HOPEX ADMINISTRATION

HOPEX V3.2

MEGA
SEE THE BIGGER PICTURE

Administrator Guide

CONTENTS

Contents 1

About HOPEX Administration 15

Presentation of this Guide 16
HOPEX Structure 17

Connecting to HOPEX Administration 19

Accessing HOPEX Administration 20
Connecting to an Environment 21

Managing Users 23

Actions to be Performed to Define a User 24
Before defining a user: profile concepts 24
Compulsory Actions to be Performed to Define a User 25
Optional Actions to be Performed to Define a User 26
Other Actions to Set or Manage a User 26
Introduction to Profile Management 27
Description of a Profile 27
Definition of the profile 27
Profile assignment 28
Connection Diagrams 28
Connection diagram (with WET) 28
Connection diagram (without WET) 30
Restricting access to the desktops of an application 31

Contents

The Administration profiles provided . . . 32
HOPEX Administrator profile . . . *32*
HOPEX Administrator - Production profile . . . *34*
User Management Web Administrator profile . . . *34*
Functional Administrator profile of a Solution . . . *34*
Profile Properties . . . 35
Name . . . *35*
Products accessible on the license (Command Line) . . . *35*
Assignable . . . *36*
Administrator profile . . . *36*
Set of UI access rights . . . *36*
Profile display . . . *37*
Profile status . . . *37*
_GUIName . . . *37*
MetaPicture . . . *37*
Working Environment Template (WET) . . . *37*
Persons . . . *37*
Available applications . . . *37*
Available desktops . . . *38*
Assignable profiles . . . *38*
Terminology . . . *38*
Available types . . . *38*
Introduction to User Management . . . 39
Users Provided . . . 40
User: Definition . . . 40
Person Properties . . . 41
Name . . . *41*
Image . . . *41*
E-mail . . . *41*
Phone number and initials . . . *41*
Data language . . . *41*
Default library . . . *41*
Person's reading access area and reading access area at creation . . . *42*
Person writing access area and writing access area at creation . . . *42*
Login . . . *42*
Belongs to a Person Group . . . *42*
Login Properties . . . 42
User code . . . *43*
Login Holder . . . *43*
Inactive user (Status) . . . *43*
Products accessible on the license (Command Line) . . . *43*
Authentication mode . . . *43*
LDAP server . . . *44*
Managing Profiles . . . 45
Viewing Profile Characteristics . . . 45
Creating a Profile . . . 46
Customizing the UI Access (Permissions) of an Existing Profile . . . 48
Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile 49
Configuring a Profile . . . 50
Configuring profile characteristics . . . *51*
Assigning a WET to a profile . . . *51*

Defining the applications accessible to the users of a profile (non WET-based configuration) 52
Defining the application desktops accessible to the users of a profile (non WET-based configuration) . . . 53
Associating a terminology with a profile . . . 53
Defining the object types available for a profile . . . 54
Assigning a Profile to a Person . . . 55
Performing a Mass Profile Assignment to Persons . . . 56
Deleting a Profile . . . 57
Access to User Management . . . 58
Accessing User Management and UI Access Management Folders . . . 58
Accessing User Management and UI Access Management Folders . . . 58
Opening the User Management window . . . 59
Opening the profile management window . . . 61
Opening the profile assignment window . . . 63
Viewing Person Characteristics . . . 64
Viewing the Login Characteristics . . . 65
Managing Users . . . 66
Creating Users . . . 66
Defining a Person . . . 67
Creating the Login of a Person . . . 69
Defining the Login of a Person . . . 70
Modifying the Properties of a User . . . 71
Preventing User Connection . . . 71
Deleting Users . . . 72
Creating or Modifying the Password of a User (Windows Front-End) . . . 72
Exporting the Repository Users . . . 73
Managing user options . . . 74
Configuring the Metamodel Access . . . 74
Authorizing Deletion of a Dispatched Object . . . 75
Making a Comment Mandatory on Dispatch . . . 75
Managing User Inactivity . . . 75
Activating/Deactivating user inactivity management . . . 75
Managing user inactivity . . . 76
Authentication in HOPEX (Windows Front-End) . . . 77
Defining Default Authentication Mode . . . 77
Viewing default authentication mode . . . 77
Defining default authentication mode to LDAP . . . 77
Modifying the user authentication mode . . . 78
Configuring Windows Authentication . . . 78
Synchronization with a company directory . . . 78
Associating a Windows user with a HOPEX user manually . . . 78
Connection in case of unique authentication . . . 79
Single sign-on precautions . . . 79
Configuring LDAP authentication . . . 79
Configuring LDAP authentication . . . 80
Accessing LDAP server management . . . 80
Creating an LDAP server . . . 80
Configuring the LDAP server . . . 81
Configuring an LDAP parameter . . . 82
Modifying LDAP directory import content . . . 83
Importing persons from an LDAP server . . . 84

Authentication and a user created on the fly (Web Front-End) 84

Managing Repositories. 87

Introduction to Repositories 88
System Repository (SystemDb) 88
HOPEX Repository 88
Repository Structure 89
Accessing Repositories 90
Creating a Repository 91
Consulting and Modifying Repository Properties 91
Accessing the Log of Repository Changes (.EMV file) 93
Repository Performance and Health 94
Consulting RDBMS Repository Performance 94
Generating a Repository Health Report 96
Performance test description 96
Health test description 96
Health report description 97
Configuring HOPEX health report emailing 97
Managing Repositories 99
Managing logfiles 99
Backup logfile 99
Backup logfile process 100
If you have a problem 100
Repository log 101
Configuring the logging for an inter repository consolidation 101
Logging 101
Modifying the log behavior 101
Viewing the Repository Update Log 102
Viewing the repository update log 102
Displaying dispatches 104
Exporting Updates 105
Enabling and Customizing Repository Indexing 105
Enabling/Disabling repository indexing for full-text search 106
Indexing a repository manually 106
Customizing the indexing scheduler 106
Deleting indexing folders 107
Converting a Repository 107
Mass converting technical data 107
Importing Libraries into a Repository 108
Repository Physical Backup 108
Backup frequencies 109
Elements to be backed up 109
Other elements to be backed up 109
Elements that could be useful to back up 110
Reorganizing an RDBMS Repository 110
Reorganizing a repository 110
Repository Logical Backup 111
Deleting a Repository 114

Updating a Repository 115
Importing command files 115
Viewing rejects 117
Viewing the import execution report file 119
Viewing the Environment Report File 120
Viewing the environment report file 120
Copying the environment report file 121
Opening the environment report file. 121
Viewing User Process Error Trace Files 122
Opening the trace file from HOPEX Administration 123
Opening the trace file from the HOPEX Server Supervisor tool. 124
Opening the trace file from HOPEX. 124
Saving the Error Zip file for Diagnostics 124
Viewing Object History 124
Optimizing Repository Access Performance 126
Managing Log Size 126
Log size management frequency 126
Deleting a log or reducing the log size 127
Backing up the repository log 128
Deleting log elements to reduce the log size 129
Modifying MetaClass loggability 131
Managing the Cache in RDBMS Environments 131
Increasing RDBMS cache size (memory). 131
Managing Status Indicators 132
Deleting RDBMS Repository Temporary and Historical Data 133
Performing Repository Regular Maintenance Tasks 134
Cleaning up a Repository 134
Configuring the Anti-Virus According to HOPEX Data 135
Referencing and Unreferencing a Repository 138
Referencing a Repository 138
Unreferencing a Repository 139
Protecting the Referencing of a Repository 139
Adding a referencing password to a repository 139
Modifying/Canceling a repository password 140

Managing workspaces 141

Private Workspace Principle. 142
Private workspace 142
Collaborative Workspace 143
Using Your Private Workspace 144
Connecting to HOPEX 144
Saving Sessions 145
Workspace Properties 146
HOPEX Repository State Changes. 147
Dispatching Your Work 147
Dispatch Conflicts. 148
Creation of duplicated objects 148
Deletion of already deleted objects or links 149

Modifying or linking a renamed object *149*
Rejects When Dispatching 149
Change in writing access values between opening and dispatching a private workspace *149*
Rename/create collisions *149*
Verifying link uniqueness *150*
Attribute uniqueness (other than name) *150*
Updating a deleted object *150*
Dispatch Report 150
Refreshing Data 151
Conflicts When Refreshing 153
Discarding Work 153
Discarding work from a private workspace *153*
Discarding work from a collaborative workspace *154*
Exiting a Session 154
Exiting a session from a private workspace *155*
Exiting a session from a collaborative workspace *156*
Workspace Administration 157
Accessing Workspace Management 157
Deleting a Workspace 159
Private Workspace Life: Example 160
Private workspace 1 . . . . . *160*
Private workspace 2 . . . . . *160*
Private workspace 3 . . . . . *161*
Private workspace 4 . . . . . *161*
Private workspace 5 . . . . . *162*
Private workspace 6 . . . . . *162*
Viewing Updates 163
Viewing Updates 163
The Updates window *164*
Displaying your current updates on the HOPEX repository *165*
Displaying your current updates on the system repository *166*
Viewing updates dispatched on the repository over a period of time *167*
Viewing the Dispatches and their Content 168
Exporting Updates 169
Exporting Your Private Workspace Log 169
Private Workspaces and Repository Size 170
Private workspace life *170*
Private workspace monitoring *171*
Modifying the maximum duration of a private workspace *171*
Managing locks 172
Principle of Locks 172
Preventing conflicts *172*
Deleting a lock or unlocking an object *172*
Clock synchronization *173*
Details on the operating method of the locks *173*
Managing Locks on Objects 174
Managing locks in HOPEX Administration *175*
Managing locks in HOPEX *175*

Managing Environments 177

Using Environments 178
Environment Structure 178
Creating an Environment 178
Moving and Referencing an Environment. 179
Moving an environment 179
Referencing an environment 179
Deleting a reference to an environment 180
Deactivating an Environment (RDBMS) 180
Customizing Environments. 181
Backing Up Environment Customizations. 181
Restoring Environment Customizations 183
Compiling an Environment 184

Managing the Scheduling (Scheduler) 187

Introduction to the Scheduler 188
Concepts 188
Job 188
Scheduler 188
Trigger 188
Option: Time Zone 188
Managing Triggers 190
Accessing Triggers 190
Accessing the Trigger management 190
Creating a Trigger 191
Managing a Trigger 192
Defining the Trigger Execution Context 192
Configuring the Trigger Scheduling 194
Defining the Execution Time Zone 194
Defining the First Execution Date of the Trigger 195
Defining the first execution date (or unique execution) 195
Defining a relative date for the first execution 195
Defining the Trigger Frequency 196
Defining the Last Execution Date 197
Defining the Execution Time 198
Defining the Trigger execution time 198
Defining a time-based recurrence on the Trigger execution 199
Managing the Scheduler 201
Accessing the Scheduler Control Panel 201
Stopping the Sheduler 201

Managing Events 203

Introduction to supervision 204
Prerequisites to Supervision 204
Supervising Events 204
Event types 204
Supervision files 205
Supervision configuration file: MegaSite.ini 205
Executable code 206
Supervision tool: HOPEX Server Supervisor 208
Starting HOPEX Server Supervisor 208
Extend HOPEX Server Supervisor functionalities 209
Modifying processes to be supervised (MegaSite.ini filter) 210
Finding the supervision file location 211
Modifying the supervision file location 212
Supervising events 213
Supervision tool 213
Supervision tool toolbar 214
Supervision tool tabs 214
Consulting a supervision event file 215
Actions from an event supervision window 217
Events to be Monitored (Production Server) 218
Events: Login and Authentication 218
Events: Configuration Management 219
Events: Workspace Activity 220
Events: Repository Connections 220
Events: Service Execution 222
Events: Infrastructure Performance and Repository Health 223
Events: Data Import/Export Tracking 229
Events: Report DataSet/GraphSet/TreeSet Generation 231
Events: Scheduled Jobs 240
Events: Reporting Datamart 240
Events: Questionnaire Generation 241

Managing objects 243

Exporting HOPEX Objects 244
Export 244
Exporting Objects 244
Exporting HOPEX objects 245
Exporting a HOPEX object from the object 247
Viewing an export file 248
Activating the export perimeter selection option 250
Viewing Objects Before Export 250
Enabling the view option 251
Viewing objects 251
Protecting Objects 254
Accessing the Object Protection Management Window 254

Assigning a Writing Access Area to an Object 255
Propagating Authorizations Between Linked Objects 256
Comparing and Aligning Objects Between Repositories 257
Compare and Align Principle 257
Compare and Align Warnings 257
Repository log 258
Users 258
Reading (confidentiality) and writing access levels 258
Comparing and Aligning Two Repositories 258
Merging Two Objects 262
Choice of the objects to be merged 262
Merging Two Objects 262
Importing a Solution Pack in HOPEX 265
Managing UI Access (Permissions) 267
Introduction to UI Access Management (Permissions) 267
Prerequisites and definitions 267
Performance 268
Accessing the UI access management (HOPEX Administration) 268
Accessing the permission management (HOPEX) 268
Object UI Access Values 269
MetaClass occurrence access permissions 269
MetaAssociationEnd access permissions 269
MetaAttribute access permissions 270
Permissions on a tool 270
Managing UI Access 270
Modifying access permissions on occurrences of a MetaClass 273
Modifying access permissions on MetaAttributes of a MetaClass 275
Modifying access permissions on tools of a MetaClass 276
Modifying access permissions of a link around a MetaClass 277
Modifying access permissions on links around a MetaClass 278
Rules on permissions while aggregating Sets of UI access rights 279
Managing Data Access Dynamically 279
Implementing a dynamic data access rule 280
Creating a permission rule (data access rule) 280
Associating a permission rule with a profile 281
Associating a permission rule with an object 281
Use case: data access rule set up 282
Generating a Report on Permissions by Profile 285
Generating the report 285
Report content 285
Managing General UI Access 285
Managing Shapes 288
Accessing Shapes 288

Managing Data Writing Access 291

Introduction to writing access management 292
Users 292
Writing Access Areas 293

Writing Access Diagram 294
Rules 294
Use 294
Link Orientation: Major and Minor Objects 294
Opening the Writing Access Diagram 296
Opening the Writing Access Diagram (Windows Front-End) 296
Compiling the Writing Access Diagram 297
Defining Writing Access Areas 298
Creating a Writing Access Area 298
Defining Writing Access Area Persons or Person Groups 298
Defining a Writing Access Area at Creation 299
Modifying Writing Access Areas of Objects 300
Modifying Writing Access Areas of an Object Group 300
Deleting Writing Access Areas 301
Propagating Object Writing Access Areas to Child Objects 302
Associating Objects with Writing Access Areas 303
Tips on Using Writing Access Areas 303
Common data 303
Tips 304
Typical example 304
Customizing Writing Access Area Management 305
Calculated Writing Access Area 305
Calculated MetaAttribute 305
Installing a Writing Access Diagram 306
Locking Validated Objects 307
Merging Two Projects 308
Splitting a Project 309
Managing Users from the Writing Access Diagram 311
Creating Persons with Writing Access Areas 311
Creating Person Group with Writing Access Areas 311
Managing Users from the Writing Access Diagram 312
Compiling the Writing Access Diagram 312
Transferring the Writing Access Diagram 312
Customizing Writing Access Diagram Display 314
Customizing Diagram Structure Representation 314
Customizing Writing Access Area Display 314

Managing Data Reading Access 317

Introduction 318
The Need to Manage Sensitive Data 318
General Concepts 318
Activating Data Reading Access Management 319
Consulting Environment Reading Access Information 319
Managing Reading Access in HOPEX 320
Compiling the Reading Access Diagram 320
Reading Access Area Matrix 322
Accessing the Reading Access Area Matrix 322

Adding an Object Reading Access Area 323
Adding a User Reading Access Area 323
Associating User Reading Access Areas with Object Reading Access Areas 324
Associating Users with User Reading Access Areas 324
Reading Access Diagram 326
Reading Access Diagram Operating 326
Activating the reading access diagram 328
Prohibiting Reading Access Diagram Modification 328
Opening the reading access diagram (Windows Front-End) 329
Organizing Reading Access Areas 329
Creating reading access areas 329
Connecting two reading access areas 330
Displaying reading access areas associated with a reading access area 330
Adding a User in the Reading Access Diagram 331
Adding a person in the reading access diagram 331
Adding a person group in the reading access diagram 331
Connecting Users to Reading Access Areas 331
Reading access area of the user 332
Reading access area at creation 332
Consulting Reading Access Diagram Information: 334
Customizing Reading Access Area Display 335
Configuring Data Reading Access 337
Associating Objects with Reading Access Areas 337
Connecting objects to object reading access areas 337
Disconnecting objects from reading access areas 338
Displaying the list of objects associated with a reading access area 338
Associating user reading access areas with objects 338
Propagating Reading Access Areas 339
Propagating a reading access area from HOPEX Administration 340
Propagating a reading access area from HOPEX 340
Managing MetaClass Sensitivity and Reading Access Areas 341
Opening the HOPEX MetaClasses reading access configuration dialog box 341
Modifying MetaClass sensitivity 342
Hiding confidential or sensitive objects in a diagram 342
Confidential or Sensitive Object Behavior 343
Displaying a confidential or sensitive object in a diagram 343
Export and Duplication 344
Generation of reports (MS Word) and Web sites 345
Macros 345
Confidential or sensitive objects and namespaces 345
Modifying Reading Access Areas 345
Modifying object reading access areas 346
Modifying user reading access areas 346

MetaClass Confidentiality Exceptions 347

Command File Syntax. 349

Command file extensions 350
Object Naming Rules. 351
Commands. 353
Basic Syntax 355
Creating an Object 355
Deleting Objects 356
Modifying an Object 356
Modifying Texts 357
Modifying a Name 359
Creating and Modifying an Object with a Single Command 359
Creating a Link Between Two Objects 360
Modifying a Link 361
Deleting a Link 361
Managing Translations 362
Validating Import 362
Displaying a Comment in the Import Dialog Box 362
Transforming an MGL File to MGR 363
Transforming an MGR File to MGL 363

Managing Options 365

Options Overview 366
Option Window Presentation 367
Accessing Options 368
Options Level 368
Modifying options at site level 368
Modifying options at environment level 368
Modifying options at profile level 368
Modifying options at user level 368
Modifying options at workstation level. 369
Option Inheritance 369
Controlling the Modification of the Options 370
Prohibiting modification of a lower level option. 370
Unlocking the modification of a lower level option. 370
Reinitializing Option Values 370
Reinitializing the values of an option 370
Reinitializing the values of an option group (Windows Front-End) 371
Generating the options Report 372
Available Option Groups 374
User options 374
Workstation Options 375

Managing Languages 376
Changing User Interface (Windows Front-End) Language 376
Defining the Data Languages Available for a User 376
Changing User Data Language (Windows Front-End) 377
Installing Additional Languages 377
Defining the Language of e-mails in Workflows 377
Managing Languages in Web Applications 378
Managing Date Format 379
Managing HOPEX Data Customization 381

Frequently Asked Questions 383

Common Operations 384
Recurrent Messages 385
Product Codes 387
Access your list of available products 387

Glossary 389

Contents

About HOPEX Administration

HOPEX products can be used on a stand-alone workstation or in a configuration including dozens of users.

This guide is for the person responsible for repository and user administration. When there are only a few users, administration is usually done by one of the users. In such cases, it mainly consists of carrying out regular backups and reorganizing repositories when required.

Chapters (Managing Users and Managing Repositories) cover most administration requirements of a structure with only a few users. In a structure with many users, the administrator must respond to more specific requirements, which are detailed in the following chapters.

Most of the functions described here can be used by the repository and user administrator, whatever the products enabled through his/her security key. However, certain functions are only available with specific technical modules (**HOPEX Power Studio** or **HOPEX Power Supervisor**). These are indicated by a note in the text.

HOPEX administration is managed from the **HOPEX Administration** application (Windows Front-End) "Administration.exe" or from the **HOPEX Administration** desktop (Web Front-End). Some actions can also be performed from the main **HOPEX** application (Windows Front-End) "Hopex.exe" or from certain **HOPEX** desktops (Web Front-End).

The **HOPEX** administration applications (Windows Front-End and Web Front-End) are designed for administrators **HOPEX**: they are used to manage environments, repositories, users, etc.

A **HOPEX** installation can contain a large number of environments, repositories and users. To facilitate their management, **HOPEX Administration** provides all the key concepts and tools required for their administration in a unified hierarchical structure.

PRESENTATION OF THIS GUIDE

This guide concerns **HOPEX** administration from the **HOPEX Administration** application (Windows Front-End).

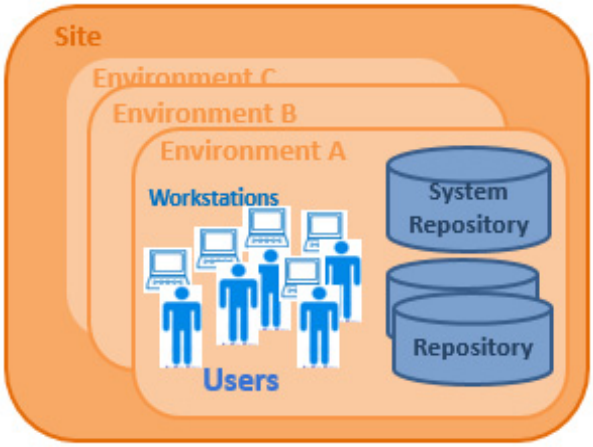
– *To perform **HOPEX** administration tasks from the **Administration HOPEX** desktop (Web Front-End), see the HOPEX Administration - Supervisor Web guide.*

It includes the following chapters:

- Connecting to HOPEX Administration: how to access **HOPEX Administration**.
- Managing Users: to create and define users, and user profiles.
  - The **HOPEX Power Supervisor** technical module is necessary to manage profiles.
  - The **HOPEX Power Studio** technical module is necessary to create profiles.
- Managing Repositories: to create, save, restore, check, reorganize, copy and move repositories.
- Managing workspaces: principle of private workspaces, dispatch and refresh private workspaces, and lock management.
- Managing Environments: to create, back up, restore, check, copy and move an environment.
- Managing Events: to supervise events with the **HOPEX Server Supervisor** tool.
- Managing objects: Advanced administration functions available with:
  - the **HOPEX Power Studio** technical module to extract objects
  - the **HOPEX Power Supervisor** technical module to manage user interface (UI) access, compare objects in two repositories.
- Managing Data Writing Access: to set up management of organized projects in the form of data writing access, from the **HOPEX Power Supervisor** technical module.
- Managing Data Reading Access: to install a confidentiality strategy using a reading access diagram and access areas.
- Command File Syntax: description of the syntax used in command files.
- Frequently Asked Questions: answers to frequently asked questions and some administration tips.
- Glossary: definition of the main terms used in this guide.

HOPEX STRUCTURE

Some basic knowledge is required to understand the architecture and operation of **HOPEX**.



HOPEX is organized in four tiers:

- ***site***
  A site groups together everything that is shared by all **HOPEX** users on the same local network: the programs, standard configuration files, online help files, standard shapes, workstation installation programs, and version upgrade programs.
- ***environment***
  An environment groups a set of users, the repositories on which they can work, and the system repository. It is where user private workspaces, users, system data, etc. are managed.
- ***workstation***
  A workstation is defined for each computer connected to the environment. A workstation contains programs and a configuration file that allow you to use **HOPEX** on that machine.
- ***user***
  A user is a person (or person group) with a login. A user:
  - has a specific workspace in each repository.
  - can connect to a repository from all workstations connected to the environment in which this repository is referenced.
  - has a specific configuration and is authorized to access specific product functions and repositories in the environment.

CONNECTING TO HOPEX ADMINISTRATION

The **Administration** application (Windows Front-End) is the **HOPEX** administration application accessible from the Windows desktop. This application contains the tools required to manage users, repositories, environments and private workspaces. It is used to manage users (individuals, business roles and profiles, access to GUIs, writing access as well confidentiality using reading access management, servers, LDAP servers) and repositories (workspaces, locks, repository snapshots, Scheduler).

The **Administration** desktop (Web Front-End) is the **HOPEX** administration application available via an internet browser. This application is used to manage users (persons, person groups, business roles, profiles, LDAP servers), repositories (workspaces, locks, repository, repository snapshots) and UI accesses. This application also provides access to tools (Excel import/export, Import/Export of command files, Scheduler, Exchange Rate) and is used to manage person skills.

– *The **Administration** desktop (Web Front-End) is described in the Web HOPEX Administration - Supervisor Guide.*

The points covered here are:

- Accessing HOPEX Administration
- Connecting to an Environment

ACCESSING HOPEX ADMINISTRATION

The **Administration** application is the **HOPEX** *administration* application accessible from the Windows desktop.

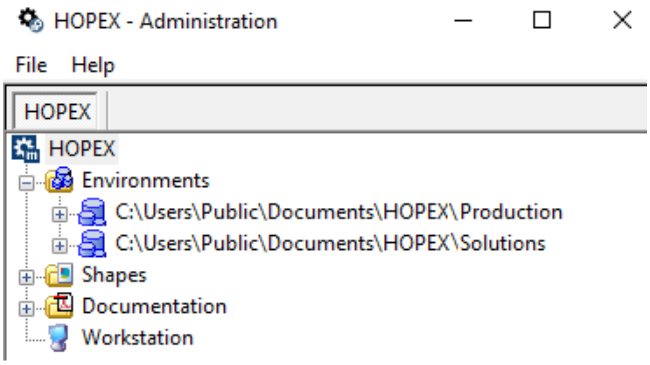
To access the **HOPEX Administration** application **(Windows Front-End)**:

› Double-click the "Administration.exe" file  .

– *The administration icon is created during setup of an administrator workstation.*

The **HOPEX Administration** main window opens.

– *Environments preceded by a red icon are not accessible.*



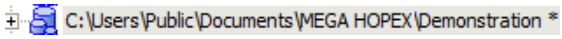
HOPEX Administration is presented in the form of a navigation tree:

- containing the elements you can manage:
  - *site*
  - *environments*
  - *repositories*
- enabling access to:
  - the list of shapes used in **HOPEX**
  - technical characteristics of the **HOPEX** installation on your workstation.

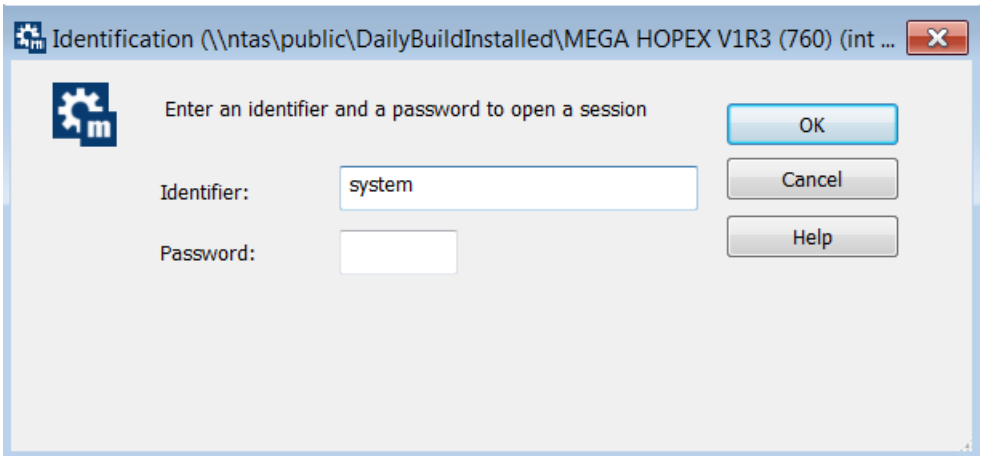
CONNECTING TO AN ENVIRONMENT

To connect to an environment:

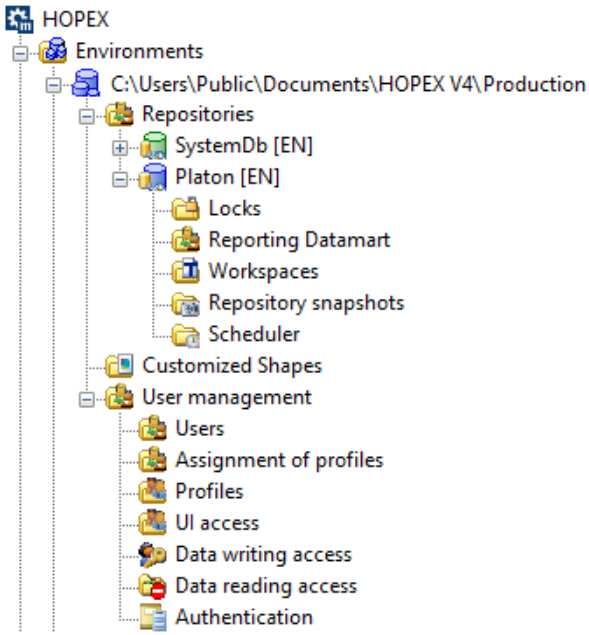
1. Access **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. Expand the **Environments** folder.
   The list of referenced environments appears.
   - *The asterisk that may appear after the environment name means that you must compile the metamodel and/or the technical data, see Compiling an Environment.*



3. Right-click the environment you want to connect to and select **Open**.
   The environment connection dialog box appears.
4. In the **Identifier** field, enter the identifier of an *Administrator*.
   - *Administrator and Mega users own administration rights. The Administrator user identifier is System. The Mega user identifier is Mega.*
5. (optional) Enter the user **Password**.
   - *By default, Administrator and Mega users do not have a password.*



6. Click **OK**.
The content of the environment folder is available.



This environment folder contains the folders:

- **Repositories** containing repositories referenced in the environment, to manage:
  - its *private workspaces*
  - *Reporting Datamarts*
  - its *locks*
  - its *repository snapshots*
  - its Triggers and Jobs (Scheduler)
- **Customized Shapes** containing **HOPEX** shapes customized by the user. They are stored in the Mega_Usr folder of the environment.
- **User Management** to manage:
  - the *users*
  - the *Assignment of profiles* of each user
  - the *profiles*
  - the *UI Access*
  - the *Data Writing Access areas*
  - the *Data Reading Access areas*
  - the authentication (*LDAP Servers*, authentication groups and parameters)

MANAGING USERS

The **Administration** application is equipped with tools required for user management.

This chapter explains how to create and manage *users* and how to define and modify their characteristics.

– *Managing person groups is a functionality available in **HOPEX** (Web Front-End). For more information on managing this functionality, see the Web HOPEX Administration - Supervisor guide.*

The following points are covered here:

- Actions to be Performed to Define a User
- Introduction to Profile Management
- Introduction to User Management
- Managing Profiles (available with HOPEX Power Supervisor)
- Access to User Management
- Managing Users
- Managing user options
- Authentication in HOPEX (Windows Front-End)

ACTIONS TO BE PERFORMED TO DEFINE A USER

To define a *user*, some actions are compulsory, while others are only necessary depending on **HOPEX** options selected, and others are optional.

) *A user is a person with a login and at least one profile assigned.*

See:

- Before defining a user: profile concepts
- Compulsory Actions to be Performed to Define a User
- Optional Actions to be Performed to Define a User
- Other Actions to Set or Manage a User

Before defining a user: profile concepts

To connect to **HOPEX** a user select the profile with which he/she wants to work.

This profile defines:

- the products accessible

  P **If a user already has restricted access rights to products (see Viewing the Login Characteristics), the products accessible to this user are at the intersection of values of the Command Line attribute of the user login and profile.**

- the desktops to which the user can access.

  - *See Connection Diagrams.*

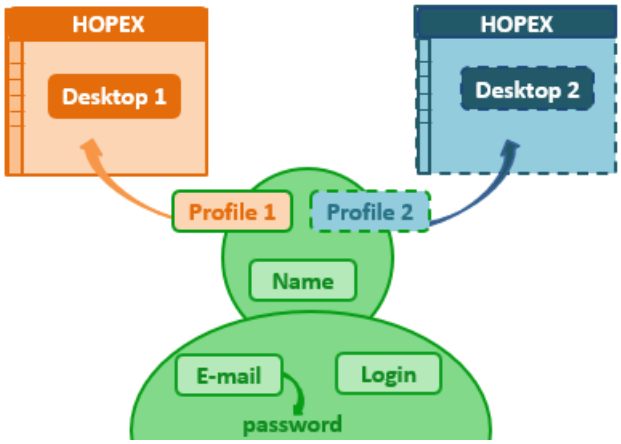
- the user's access rights to UIs (permissions)

Assigning a profile to a person defines:

- *See Assigning a Profile to a Person.*

- the repository concerned by the assignment
- the person's access rights to repositories with this profile assignment
- (optional, with read-only access to the repository) connection repository snapshot
- (optional) the validity period of the assignment

Compulsory Actions to be Performed to Define a User



To create a user who can connect to **HOPEX** you must:

- define the name of the person
  - *See Creating Users.*
- define the login of the user
  - **A person must have a login to be able to connect to HOPEX.**
  - *See Defining the Login of a Person.*
  - *The login of the user is created automatically on creation of the person, see Creating Users (If necessary, see Creating the Login of a Person).*
  - *The login status must be active so the person can connect, see Inactive user (Status).*
- (recommended) define the e-mail address of the person
  - *See Defining a Person.*
  - *The e-mail address is necessary, for example, for distributing documents, receiving notifications and questionnaires, or when a Web user lost his/her password.*
- assign a profile to a person
  - **The user must have at least one profile assigned to be able to connect to HOPEX.**
  - *See Assigning a Profile to a Person.*

Optional Actions to be Performed to Define a User

According to the selected options you must:

- (recommended) define the e-mail address of the person
  - *The e-mail address is necessary, for example, for distributing documents, receiving notifications and questionnaires, or when a Web user lost his/her password.*
- (where writing access management is activated) define the writing access area of the user
  - *See Defining a Person.*
- (where reading access management is activated) define the reading access area of the user
  - *See Defining a Person.*

Other Actions to Set or Manage a User

You can:

- define the phone number and initials of the person
  - *See Defining a Person.*
- restrict user access to certain products
  - *The products accessible to this user are at the intersection of the values of the **Command Line** attribute of the user login and profile.*
  - *See Defining the Login of a Person.*
  - *See Configuring a Profile.*
- modify user authentication mode
  - *See Defining the Login of a Person.*
- make the user inactive.
  - *See Defining the Login of a Person.*
  - *See Preventing User Connection.*

INTRODUCTION TO PROFILE MANAGEMENT

Managing users involves managing profiles. A user connects to **HOPEX** with a specific profile that determines the **HOPEX** application to which the user connects and the desktops with which it is associated.

See:

- Description of a Profile
- Connection Diagrams
- The Administration profiles provided
- Profile Properties

Description of a Profile

A profile enables definition of the same connection rights to a set of users.

- *See Viewing Profile Characteristics.*

The description of a profile includes:

- the definition of the profile
- the definition of the profile assignment to a person

Definition of the profile

A profile defines the function of a person or person group in the enterprise

Example: Application Portfolio Manager, Enterprise Architect).

The profile defines:

- the products accessible
  - *See Products accessible on the license (Command Line).*

  P **If a user already has restricted access rights to products (see Viewing the Login Characteristics), the products accessible to this user are at the intersection of values of the Command Line attribute of the user login and profile.**
- the desktops to which the user can access.
  - *See Connection Diagrams.*
  - *See Assigning a WET to a profile or Defining the applications accessible to the users of a profile (non WET-based configuration).*
- the user's access rights to UIs (permissions)
  - *See Managing UI Access.*
- the same options
  - *See Viewing Profile Characteristics.*

Profile assignment

You must assign each person at least one profile so that this person can connect to **HOPEX**.

– *By default, no profiles is assigned to a person.*

Assigning a profile to a person defines:

- the repository concerned by the assignment
- the person's access rights to repositories with this profile assignment
- (optional) the validity period of the assignment
- (optional, with read-only access to the repository) connection repository snapshot

– *See Assigning a Profile to a Person.*

Connection Diagrams

The connection diagram relies on the desktop creation, that is whether the desktop is based on a Work Environment Template (WET) or not.

Connection diagram (with WET)

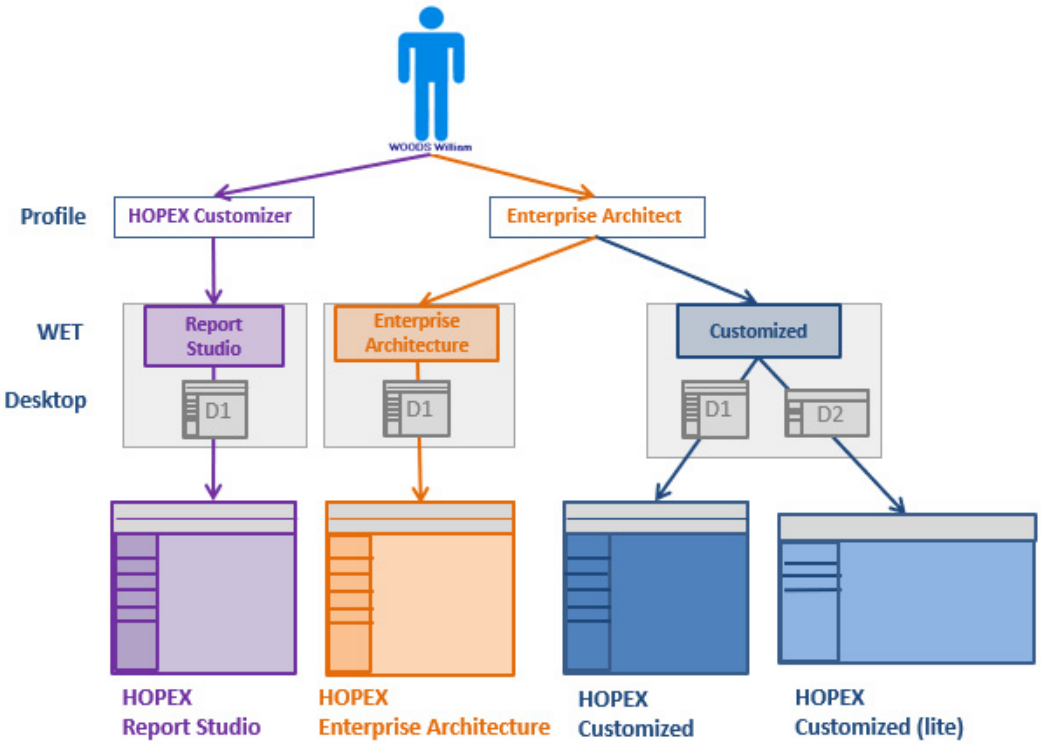
Using a Working Environment Template (WET) enables to homogenize the display of the desktops.

– *For detailed information regarding the WET creation, see HOPEX Power Studio - Versatile Desktop documentation.*

To connect to **HOPEX**, a person must have:

- a login
  - *See Creating Users.*
  - *The login status must be active so the person can connect, see Inactive user (Status).*
- at least one profile
  The profile gives access to one or several WET-based desktops.
  - *See Assigning a Profile to a Person.*

  At least one WET (with one or several associated desktops) must be assigned to the profile. A desktop manager enables to define the desktops associated with this WET-profile assignment.
  - *See Assigning a WET to a profile.*
  - *In a non WET-based configuration, applications (and their associated desktops) are connected to the profile.*



In the above example, William WOODS has an active login. He can connect to:

- **HOPEX Report Studio** with the **HOPEX Customizer** profile.
- **HOPEX Enterprise Architecture** with the **Enterprise Architect** profile.
- **HOPEX Customized** with the **Enterprise Architect** profile and choose a device (computer or tablet) adapted display.

Connection diagram (without WET)

To connect to **HOPEX**, a person must have:

- a login
  - *See Creating Users.*
  - *The login status must be active so the person can connect, see Inactive user (Status).*
- at least one profile.
  - *See Assigning a Profile to a Person.*

So that a user of a profile can connect to an application, you must connect this application to the profile concerned.

All desktops connected to the application are then accessible.

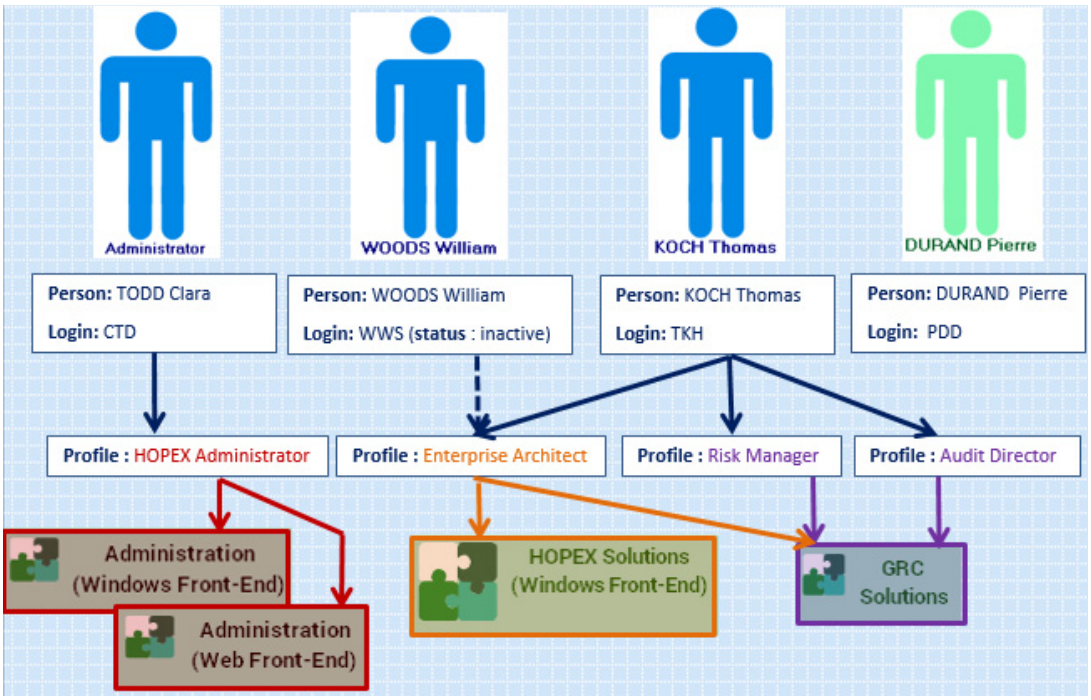
- *To modify a profile supplied by **MEGA**, **MEGA** recommends you create a new profile, see Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.*
- *To enable access to only certain desktops of the application, see Restricting access to the desktops of an application.*

Example:

A1 application is connected to **P1** profile and **A2** application is connected to **P2** profile.

None of the desktops of **A1** and **A2** applications are directly connected to **P1** and **P2** profiles.

The user U1, who is assigned the **P1** and **P2** profiles, has access to all of the desktops of **A1** and **A2** applications.



In the previous example:

- Clara TODD has a login and the **HOPEX Administrator** profile assigned: she can connect to **Administration** applications (Windows Front-Endand Web Front-End).
- William WOODS has the **Enterprise Architect** profile assigned but the status of his login is inactive:
he cannot connect to **HOPEX Solutions (Windows Front-End)** or **GRC Solutions** applications.
- Thomas KOCH has a login and the **Enterprise Architect**, **Risk Manager** and **Audit Director** profiles assigned:
he can connect to **HOPEX Solutions (Windows Front-End)** and **GRC Solutions** applications.
- Pierre DURAND has a login but does not have an assigned profile:
he cannot connect to **HOPEX**.

Restricting access to the desktops of an application

A user can connect to an application via customized desktops according to actions to be performed.

If an application contains several desktops, you can specifically define application desktops that are accessible to the concerned profile. To do this, you must connect to the profile:

- *To modify a profile provided by **MEGA**, you must have rights to modify **HOPEX** data. Alternatively, you can create a new profile, see Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.*

- the application containing the desktops.

  - *See Defining the applications accessible to the users of a profile (non WET-based configuration).*

- the desktops you want the users of the profile can connect to.
The application desktops that are not connected to the profile are not accessible to users of the profile.

  - *To enable access to only certain desktops of the application, see Defining the application desktops accessible to the users of a profile (non WET-based configuration).*

Example:

P1 profile is connected to:

- **A1** application, which particularly includes D1, D2, D3, D4, and D5 desktops.

- D2 and D5 desktops of the A1 application.

User U1 with the **P1** profile can connect only to the D2 and D5 desktops of the **A1** application. He is not allowed to access D1, D3, and D4 desktops.

The Administration profiles provided

Administration profiles are provided at installation with defined rights and access to applications.

When several users with an Administration profile connect to ***HOPEX Administration*** at the same time, certain actions, such as user management, are exclusive.

These profiles are dedicated to:

- global Administration, with exclusive access to **Administration** applications (Windows Front-End and Web Front-End):
  HOPEX Administrator
  - *See HOPEX Administrator profile.*
- Administration (Web Front-End), with exclusive access to the **Web Administration** desktop:
  - **HOPEX Administrator - Production**
    - *See HOPEX Administrator - Production profile.*
  - **User Management Web Administrator**
    - *See User Management Web Administrator profile.*
- functional Administration (Web Front-End), with access to the **Web Administration** desktop and to Solution-specific desktops:
  <Solution name> functional Administrator

  Example: **ITPM functional Administrator** gives access to Environment, ITPM, and Administration desktops.
  - *See Functional Administrator profile of a Solution.*

If needed you can modify the rights and access to applications defined on these profiles.

- *See Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile and Configuring a Profile.*

HOPEX Administrator profile

- *When several users with an Administration profile connect to **HOPEX Administration** at the same time, certain actions are exclusive (e.g.: user management).*

In the **Administration** application (Windows Front-End), the **HOPEX Administrator** profile allows, in particular, to manage :

– *For information on the HOPEX Administrator profile in the Web Administration desktop, see HOPEX Administration - Supervisor guide.*

- environments
  – *See Managing Environments.*
- repositories
  – *See Managing Repositories.*
- profiles
  – *See Managing Profiles.*
- users (Persons and Logins)
  – *See Managing Users.*
- workspaces
  – *See Managing workspaces.*
- objects
  – *See Managing objects.*
- permissions
  – *See Managing UI Access (Permissions).*
- authentication
  – *See Authentication in HOPEX (Windows Front-End).*

It also allows to perform tasks linked to:

- file import/export:
  - Excel
    – *See the **HOPEX Common Features** guide, "Exchanging Data With Excel" chapter.*
  - XMG/MGL/MGR
    – *See Importing command files.*
    – *See Exporting Objects.*
  - Visio
    – *See **HOPEX Common Features - Visio Import**.*
- Snapshot use
  – *To create a repository snapshot, see the **HOPEX Common Features - Repository Snapshots** guide.*
- Scheduler use
  – *See **HOPEX Power Studio - Scheduler** guide.*

(Web specific) It also allows to manage:

- *User groups* (Person groups and Logins)
) *(Web Front-End specific) A person group groups persons in a group. These persons share the same connection characteristics.*
  – *For information on user groups, see the **HOPEX Administration - Supervisor Web** guide.*
- business roles
  – *For more information on business roles, see the **HOPEX Administration - Supervisor Web** guide.*

HOPEX Administrator - Production profile

The **HOPEX Administrator - Production** profile is the equivalent of the **HOPEX Administrator** profile in the **Web Administration** Desktop, without permission management rights.

– *For information on the HOPEX Administrator profile in the Web Administration desktop, see HOPEX Administration - Supervisor guide.*

User Management Web Administrator profile

The **User Management Web Administrator** profile allows, in particular, to manage:

– *For information on the User Management Web Administrator, see the HOPEX Administration - Supervisor Web guide.*

- users (Persons and Logins)
- *User groups* (Person groups and Logins)

) *(Web Front-End specific) A person group groups persons in a group. These persons share the same connection characteristics.*

– *For information on user groups, see the **HOPEX Administration - Supervisor Web** guide.*

- business roles

– *For more information on business role management, see the **HOPEX Administration - Supervisor Web** guide.*

It also gives access to management of:

- authentication
- lock management.

Functional Administrator profile of a Solution

Each **<Solution name> functional Administrator** gives access to the Administration desktop and to Solution-specific desktops.

– *For information on the functional Administrator of a Solution, see the **HOPEX Administration - Supervisor Web** guide.*

From an administration point of view, the **<Solution name> Functional Administrator** profile allows, in particular, to manage:

- profiles
  A Functional Administrator of a Solution, can only assign profiles related to this Solution.
- users (Persons and Logins)
- *User groups* (Person groups and Logins)

– *For information on user group management, see the **HOPEX Administration - Supervisor Web** guide.*

- business roles

– *For more information on business role management, see the **HOPEX Administration - Supervisor Web** guide.*

It also gives access to:

- scheduler use

– *See **HOPEX Power Studio - Scheduler** guide.*

- authentication management.

Profile Properties

A profile enables definition of the same connection parameters and rights to a set of users.

- *See Introduction to Profile Management.*
- *To assign a profile to a person, see Assigning a Profile to a Person.*
- *To manage profiles, see Managing Profiles.*

Name

The **Name** of a profile can comprise letters, figures and/or special characters.

Products accessible on the license (Command Line)

The **Command Line** field enables definition of products that can be accessed by users with the current profile.

Format of the command is:

/RW'<accessible Product A code>;<accessible Product B code>;<...>'

For example: You have licenses for products **HOPEX Business Process Analysis**, **HOPEX IT Portfolio Management** and other **HOPEX products**. To authorize only **HOPEX Business Process Analysis** and **HOPEX IT Portfolio Management** modules to users that have this profile, enter:

/RW'HBPA;APM'

- *To find out the product code, see Access your list of available products or the online documentation: **Concepts > Products**.*
- *To find out the available products, see Access your list of available products.*

P **If a user already has access rights restricted by the Command Line attribute on his/her Login (see Viewing the Login Characteristics), the products accessible to this user are at the**

intersection of values of the Command Line attribute of the user login and profile.

		Profile 1	Profile 2
	Command line	**RW:/'APM'**	**none**
User A	RW:/'APM;HBPA'	user A has access to **HOPEX IT Portfolio Management**	user A has access to: **HOPEX IT Portfolio Management** and **HOPEX Business Process Analysis**
User B	RW:/'HBPA'	user B cannot access any product	user B has access to **HOPEX Business Process Analysis**
User C	none	user C has access to **HOPEX IT Portfolio Management**	user C can access all of the products for which he has the license (**HOPEX IT Portfolio Management** and **HOPEX Business Process Analysis**)

*Restrictions on products for users and profiles that have licenses for **HOPEX IT Portfolio Management** and **HOPEX Business Process Analysis**.*

Assignable

The **Assignable** attribute defines if the profile is assignable to a Login or not.

M *This attribute enables filtering of profiles and improves visibility of profiles to be assigned.*

– *The default value is "No".*

Administrator profile

Only the user whose current profile has the **Administrator Profile** attribute with value "Yes" can:

- grant administrator profile to another user.
- declare a profile as administrator.
  That is, specify value "Yes" for the **Administrator Profile** attribute of any profile.

The default value of **Administrator Profile** is "No".

Set of UI access rights

The **Set of UI Access Rights** attribute enables to define permissions associated with one or several profiles.

Profile display

A profile is provided by default at connection when it is not included in another profile.

The **Profile Display** attribute defines when the profile is provided at connection:

- "always": the profile is provided at connection even if it is included in the definition of another profile,

  – *See Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.*

- "If not included in another profile" (default value): the profile is provided at connection only if it is not included in another profile.

Profile status

The **Profile Status** attribute is used to define the profile as inactive if necessary.

_GUIName

The **_GUIName** attribute enables definition of the profile name display in the interface.

MetaPicture

The **MetaPicture** attribute enables customization of the icon representing the current profile.

Working Environment Template (WET)

The **Assign a WET** tab enables to assign a WET (Working Environment Template) to the profile. This WET defines the desktops to which the profile gives access and their display.

– *See Assigning a WET to a profile.*

– *For more details on the WET use and creation, see HOPEX Power Studio - Versatile Desktop - Using a Working Environment Template (WET).*

Persons

The **Persons** tab lists all the persons who are connected to the current profile.

Available applications

In cases where a **Working Environment Template (WET)** is not defined, the **Available Applications** tab is used to define the applications to which the current profile gives access.

– *See Defining the applications accessible to the users of a profile (non WET-based configuration).*

Available desktops

In cases where a **Working Environment Template (WET)** is not defined, the **Available Desktops** tab is used to restrict the desktops to which the current profile gives access. By default all the desktops connected to the application are accessible.

- *To restrict the desktops accessible, see Defining the application desktops accessible to the users of a profile (non WET-based configuration).*

Reporting presentation

The **Reporting** property page of an object gives access to the reports available for the accessible products. Reports of the main product are displayed at first level.

The **Reporting Presentation** page enables do define this first level.

Assignable profiles

The **Assignable Profiles** tab lists the profiles that the current profile allows to assign.

- *To assign a profile, see Assigning a Profile to a Person.*

Terminology

The **Terminology** enables associating a terminology to the profile.

- *See Associating a terminology with a profile.*

Available types

The **Available Types** tab enables definition of the specific objects available for the profile:

- Document category
- Business Document Pattern
- Report DataSet Definition
- Widget

- *See Defining the object types available for a profile.*

INTRODUCTION TO USER MANAGEMENT

– *Only a user with Administrator type profile has management rights. In particular, he/she is the only user who can modify user characteristics.*

User management in the **Administration** application involves the following concepts:

- *users*:
  -) *A user is a person with a login and at least one profile assigned.*
  - *persons*
    -) *A person is defined by his/her name and electronic mail address.*
  - *logins*
    -) *A Login uniquely defines a user or user group. It can be assigned to only one Person or Person Group.*
- *profiles*
  -) *A profile defines what a person can see or not see and do or not do in tools, and how he/she sees and can do it. The profile defines options, access rights to repositories and products, read/write and read-only rights on objects.*
- permissions:
  - *object UI access*
    -) *Object UI access defines user rights on creation, reading, update, and deletion on these objects and their tools. By default, object UI accesses have value *CRUD (C: create, R: read, U: update, D: delete, *: default value).*
  - *general UI access*
    -) *General UI access defines if tools are available or not. By default, general UI accesses have value *A (A: Available, *: default value)*

For information on:

- managing *data writing access*, see Managing Data Writing Access
- managing *data reading access*, see Managing Data Reading Access

The following points are detailed here:

- introduction:
  - Users Provided
  - User: Definition
- properties:
  - Person Properties
  - Login Properties
- access:
  - Accessing User Management and UI Access Management Folders
- characteristics:
  - Viewing Person Characteristics
  - Viewing the Login Characteristics

Users Provided

By default, at installation the following are created in the environment:

- persons indispensable to the system:
  - ***Administrator***, with Login "System" and password "Hopex"
    - *The "Administrator" user cannot be deleted. It has no profile (it has all rights).*
    - *The "Administrator" user can create a first user with the "HOPEX Administrator" profile to manage repositories and users.*
  - ***MEGA Agent***, with Login "SysMA"
    - *The "MEGA Agent" user cannot be deleted. The "MEGA Agent" user is used by the system to manage workflows. It has no profile (it has all rights).*
- a person given by way of example:
  - ***Mega***, with Login "Mega" and password "Hopex"
    - *The "Mega" user can be deleted (not recommended). The "Mega" user has the "HOPEX Administrator" profile, which allows to manage repositories and users.*

User: Definition

For each environment, a user has:

- personal characteristics defined by his/her **Person**.
  - *see Viewing Person Characteristics.*
- a **login** which defines his/her connection identifier, his/her status and his/her authentication **HOPEX** mode. The login can also restrict the accessible products.
  - *see Login Properties.*
- a **user code** which enables naming of user associated files, for example the work repository.
  - *see Login Properties.*
- at least one **profile** assigned that determines the products (restricted by the products defined for the user login), applications, desktops, and repositories to which the user has access as well the access rights to UIs (permissions).
  By default the user does not have an assigned profile.
  - *see Profile Properties.*
  - *see Managing Profiles.*
  - *see Assigning a Profile to a Person.*
- **options**
  - *see Managing Options.*

Only a user with **HOPEX Administrator** profile (or with equivalent rights) can configure and modify user properties.

- *see HOPEX Administrator profile.*

Person Properties

- *For information on a user, see User: Definition*
- *To consult properties of a person, see Viewing Person Characteristics.*
- *To define the properties of a person, see Defining a Person.*

Name

The name of the person can include name and first name. It can comprise letters, figures and/or special characters. Format of the name of the person is free. It is recommended that the same format be used for all persons.

```
Example: DURAND Pierre
```

Image

You can download an image in .ico, .bmp, .gif or .png format up to a size of 30 MB.

E-mail

The person e-mail address is useful, for distribution of reports (MS Word) for example.

It is mandatory for password change and for receipt of questionnaires for example.

```
Example: pdurand@mega.com
```

Phone number and initials

The phone number and initials of the person are optional.

```
Example: +33102030405 / DP
```

Data language

The **Data language** attribute of the person is specific to Web applications. It enables definition of a specific data language for this user. It is the language in which data names are displayed by default, in case several languages are available.

- *By default, the data language is defined in the environment options for all users at installation (Options/Installation/Web application) via the **Data language** option.*
- *To define interface language, see Managing Languages.*

Default library

The **Default Library** attribute enables attachment of objects created by the person in a library, when the creation context does not permit this.

Person's reading access area and reading access area at creation

- *Information related to the reading access area are only visible when the **Activate reading access diagram** is selected in **Options** of the **Repository** of the **Environment**.*

Certain objects or modeling projects may be confidential or contain data (costs, risks, controls) that should be visible only to authorized users.

The **HOPEX** administrator can hide objects corresponding to this confidential data.

To implement a data confidentiality policy, objects must be organized in distinct sets. Each set of objects is a ***reading access areas***.

Each user is associated with a reading access area that determines objects the user can see. A user can only see objects located in his/her own reading access area or in the lower reading access areas.

- *For more details on reading access areas, see Managing Data Reading Access.*

Person writing access area and writing access area at creation

- *Writing access management is available with the **HOPEX Power Supervisor** technical module.*

A writing access area is assigned to an object to protect it from inadvertent modifications. At creation, an object takes the writing access area of the user that creates it.

By default, a new user is connected to the only writing access area: "Administrator".

There is a hierarchical link between writing access areas: a user can only modify an object when he/she has the same writing access level as this object or a higher writing access area level.

- *See Managing Data Writing Access.*

Login

The login of a person is a unique character string uniquely identifying the person that can connect. The person without a login cannot connect to ***HOPEX***.

```
Example: pdurand, pdd
```

- *For more details, see Login Properties.*

Belongs to a Person Group

- *Managing person groups is a functionality available in **HOPEX** (Web Front-End). For more information on managing this functionality, see the Web HOPEX Administration - Supervisor guide.*

Login Properties

To:

- create a login, see Creating the Login of a Person or Creating Users.
- view login characteristics, see Viewing the Login Characteristics.
- configure a login, see Defining the Login of a Person.

User code

The **User Code** is the short identifier (upper case, maximum length 6 characters) of the user that serves as the basis for private workspace naming.

This code is defined at creation of the user. To ensure data consistency, it should not be modified.

```
Example: PDD
```

Login Holder

The login holder is the person associated with the login.

```
Example: DURAND Pierre
```

Inactive user (Status)

Login status can be used to make a user inactive (value: Inactive). The user no longer has access to repositories, but trace of his/her actions is retained. The user can be easily reactivated (value: Active).

P **When you delete a user from the repository, the commands connected to this user become orphans and you lose part of the history saved in logs. With Inactive status, the user no longer has access to repositories, but the history of commands connected to the user is kept in logs.**

Products accessible on the license (Command Line)

The **Command Line** field enables restriction of access of a user or profile to available products.

- *For more details, see Products accessible on the license (Command Line).*

P ***If* a user is connected to a profile and the user and profile each have access to products restricted by the Command Line attribute, the products accessible to the user are at the intersection of the values of the Command Line attribute of the user and profile.**

Authentication mode

Default value of the **Authentication Mode** parameter on the user login is inherited at user creation from the **Authentication Mode** option defined in the options of the environment (**Options** > **Installation** > **User Management**).

- *See Defining Default Authentication Mode.*

Authentication mode of a user is by checking the user password. Available authentication modes are:

- **MEGA**
  The HOPEX authentication service checks that the password entered matches the password stored (hashed and encrypted) in HOPEX repository.
  This is default authentication mode.
  - *For more details, see Authentication in HOPEX (Windows Front-End).*
- **LDAP**
  Passwords are managed and stored in the LDAP server of the enterprise. The directory configuration is stored in HOPEX options.
  The HOPEX user is authenticated at the LDAP server level.
  - *For more details, see Configuring LDAP authentication.*
- **Windows**
  Passwords are managed and stored in Windows. This allows the user connected to Windows to be recognized automatically when he/she is connected to **HOPEX** (Windows Front-End), not requiring entry of his/her password.
  - ***Attention**: to connect to a **HOPEX (Web Front-End)** application, the user must enter his/her password.*

  The list of users in your **HOPEX** environment is automatically synchronized with the list of users defined in your Windows network.
  This authentication mode corresponds to unique authentication (SSO).
  - *For more details, see Configuring Windows Authentication.*

LDAP server

- *This field only appears when the **Authentication Mode** is "LDAP", see Authentication mode.*

The **LDAP Server** is the server with which the **HOPEX** user is authenticated in LDAP authentication mode.

This server contains the LDAP directory in which the **HOPEX** user is registered.

MANAGING PROFILES

– *Profile management is only available with the **HOPEX Power Supervisor** technical module.*

– *Profile creation is only available with the **HOPEX Power Studio** technical module.*

Profiles are managed in the **HOPEX** administration applications: the **HOPEX Administration** application (Windows-Front-End) or the **HOPEX Administration** desktop (Web Front-End).

) *A profile defines what a person can see or not see and do or not do in tools, and how he/she sees and can do it. The profile defines options, access rights to repositories and products, read/write and read-only rights on objects.*

– *See Introduction to Profile Management.*

The following points are detailed here:

- Viewing Profile Characteristics
- Creating a Profile
- Customizing the UI Access (Permissions) of an Existing Profile
- Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile
- Configuring a Profile
- Assigning a Profile to a Person
- Performing a Mass Profile Assignment to Persons
- Deleting a Profile

To:

- modify profile options
  – *See Managing Options.*
- manage metamodel filters at profile level
  – *See Managing UI Access (Permissions).*
- implement data access rules
  – *See Managing Data Access Dynamically (HOPEX Administration documentation).*
- compare profile permissions
  – *See Generating a Report on Permissions by Profile.*

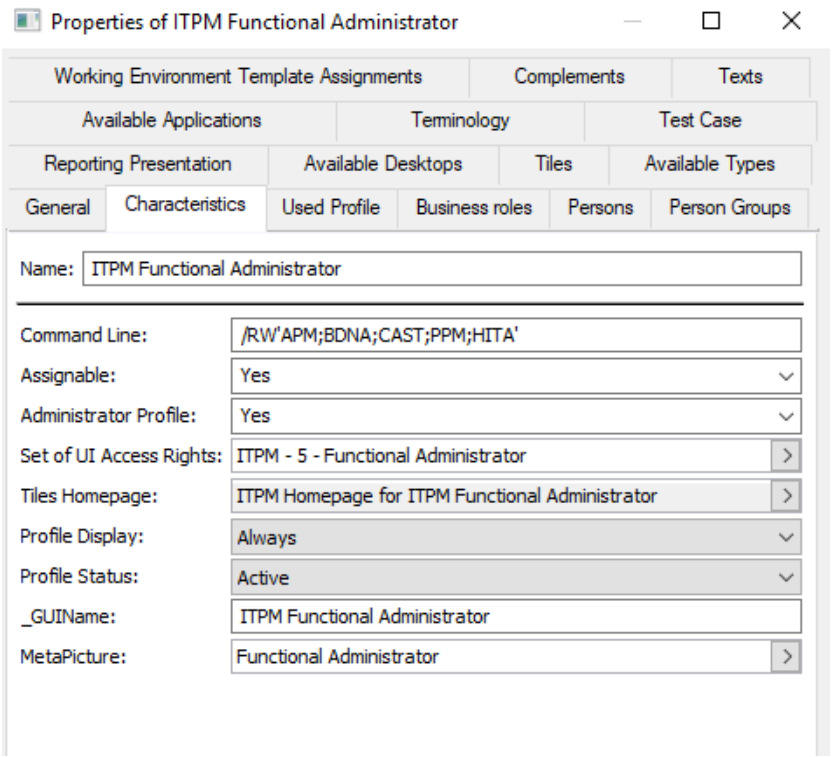
Viewing Profile Characteristics

To view profile characteristics:

1. Open the **Profiles** management window.
   – *See Opening the profile management window.*
2. In the **Profile** tab, select the profile.
3. In the toolbar, click **Properties** .
   The profile **Properties** dialog box opens.

4. Select the **Characteristics** tab.

   - *For detailed information on characteristics of a profile, see Creating a Profile.*



   - *See Configuring a Profile.*

Creating a Profile

- *Profile creation is only available with the **HOPEX Power Studio** (code MTS2) technical module.*

Users with the same profile share common characteristics (e.g.: options, authorized products, UI access rights).

To create a profile you must define:

- its name
- its set of UI access rights

  – *Defining UI access rights might be tricky. To facilitate the definition, you can use one (or several) **Set of UI access rights** already defined.*

  The set of UI access rights created inherits from all of the permissions defined on the Sets of UI access rights you have connected to it.
- its characteristics
- (WET-based desktop) its assigned WET
- (Non WET-based desktop) its accessible desktops and applications

  – *For detailed information on a WET, see **HOPEX Power Studio** - Using a Working Environment Template documentation.*

To create a profile:

1. Open the **Profiles** management window.

   – *See Opening the profile management window.*
2. Click **New** .
3. In the profile creation dialog box that appears, enter the **Name** of the profile.

   – *By default the **Name** of the profile is created in format "Profile-x" (x is a number that increases automatically).*
4. In the **Set of UI Access Rights** field, click the arrow and select **Create Set of UI Access Rights**.
5. In the **Name** field, enter a name for the Set of UI access rights of the profile.
6. (Optional, to use one or several Sets of UI access rights already defined)

   Click **Connect** :
   - (Optional) In the search field, enter the character string to be searched for.
   - Click **Find** .
   - In the list, select the Set of UI access rights on which you want to base the Set of UI access rights of your profile.

     – *To select several Sets of UI access rights, use the [Shift] key.*

     The Set of UI access rights you are creating inherits from all of the permissions defined on the Sets of UI access rights you have connected to it.
   - Click **OK**.
7. Click **OK**.
   The new profile appears in the list of profiles.
8. Keep the profile selected and click **Properties** .
   The profile properties window opens and displays the **Characteristics** tab.
9. Configure the profile characteristics.

   – *See Configuring a Profile.*

   Example: In the **Characteristics** page, set the **Assignable** parameter to "Yes", connect a **Tiles Homepage**.

10. (WET-based desktop) Assign a WET to the profile.
    - *See Assigning a WET to a profile.*
11. (Non WET-based desktop) Define:
    - the accessible desktop(s)
      - *See Defining the application desktops accessible to the users of a profile (non WET-based configuration).*
    - the available applications.
      - *See Defining the applications accessible to the users of a profile (non WET-based configuration).*

      `Example: "The Web Front-End for a Web application.`
12. (If needed) Define the Set of UI access rights of the profile.
    - *See Managing UI Access (Permissions).*

Customizing the UI Access (Permissions) of an Existing Profile

MEGA provides profiles adapted to each Solution or product. However, you might need to customize the UI access (permissions) of these profiles. For this purpose, **MEGA** recommends you to create a **Set of UI Access Rights** from the **Set of UI Access Rights** of the profile concerned, then to customize it.

To customize the UI Access (Permissions) of a profile:

1. Access the properties of the profile.
   - *See Viewing Profile Characteristics.*
2. In the **Set of UI Access Rights** field, click the arrow and access its **Properties**.
3. In the **Characteristics** tab, click the arrow of the **Customizing Set of UI Access Rights** and select **Create Set of UI Access Rights**.
   The name format of the Set of UI Access Rights s predefined as:

   `<Name of the Set of UI Access Rights of the profile concerned> (Custom)`
4. (If needed) Modify its **Name**.
5. Click **OK**.
   The set of UI access rights you created is predefined with the same UI access rights as those defined for the profile concerned.
6. Click **OK**.
7. Customize the UI Access of the set of UI Access rights you just created.
   - *See Managing UI Access (Permissions) and in the **Access Rights** field select the Set of UI access rights you just created.*

Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile

MEGA provides profiles adapted to each Solution or product. However, you may need to customize the characteristics of a profile provided by **MEGA** (for example connect a terminology).

M ***To customize a profile provided by MEGA, MEGA recommends to create a profile and base its Set of UI access rights on those of the profile you want to customize.***

To customize the characteristics of a profile provided by **MEGA**:

1. Create a profile and configure its **Set of UI Access Rights** by aggregating the Set of UI access rights of the profile on which is based your profile.
   - *See Creating a Profile.*
2. Configure the profile.
   - *See Configuring a Profile.*

Configuring a Profile

From the profile properties window you can define:

- *See Profile Properties.*

- products accessible to users with the current profile.
  - *See step 2.*
- if the profile is assignable or not.
  - *See step 3.*
- if the profile is an administrator profile or not.
  - *See step 4.*
- if the profile is provided at connection.
  - *See step 5.*
- if the profile is active or not.
  - *See step 6.*
- the profile display name in the interface.
  - *See step 7.*
- the profile icon in the interface.
  - *See step 8.*
- the Working Environment template (WET), which defines the desktops to which the users of the profile have access.
  - *See Assigning a WET to a profile.*

  Or, in a non WET-based configuration:
  - applications accessible to the users of the profile.
    - *See Defining the applications accessible to the users of a profile (non WET-based configuration).*
  - (If needed) desktops accessible to the users of the profile.
    - *See Defining the application desktops accessible to the users of a profile (non WET-based configuration).*
- the terminology associated with the profile.
  - *See Associating a terminology with a profile.*
- object types available.
  - *See Defining the object types available for a profile.*

To:

- customize the UI access rights of the profile, see:
  - Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.
- perform a mass profile assignment to persons, see:
  - Performing a Mass Profile Assignment to Persons.

Configuring profile characteristics

To configure profile characteristics:

1. Access the properties of the profile.
   - *See Viewing Profile Characteristics.*
2. (Optional) In the **Command Line** field, enter the command defining products that can be accessed by users with the current profile.
   - *See Products accessible on the license (Command Line).*
3. (Optional) In the **Assignable** field, modify the attribute value via the drop-down menu.
   - *By default, the profile is not assignable.*
   - *See Assignable.*
4. (Optional) In the **Administrator Profile** field, modify the attribute value.
   - *By default, the profile is not an administrator profile.*
   - *See Administrator profile.*
5. (Optional) In the **Profile Display** field, modify the attribute value.
   - *By default the profile is provided at connection.*
   - *See Profile display.*
6. (Optional) In the **Profile Status** field, modify the attribute value.
   - *By default, the profile is active.*
7. (Optional) In the **_GUIName** field, enter the profile name displayed in the interface.
8. (Optional) In the **MetaPicture** field, click the arrow and select **Connect MetaPicture**.
   - in the query field, enter the characters you want to find and click **Find**.
   - in the result list, select the icon and click **Connect**.

Assigning a WET to a profile

- *For more details on the WET creation and its use with profiles, see **HOPEX Power Studio - Versatile Desktop - Using a Working Environment Template (WET)**.*

With a WET-based configuration, you must assign a WET to the profile. This WET assignment to the profile enables you to define the desktop(s) associated with the profile.

- *In a non WET-based desktop configuration, you must define the applications accessible to the profile, see Defining the applications accessible to the users of a profile (non WET-based configuration).*

The desktop definition is done through a Desktop Manager. Thanks to this Desktop Manager you can, for example, define a desktop display adapted to the device (tablet or computer) used by the user.

```
E.g.: the user can connect to HOPEX Explorer application
from a tablet or a computer with an adapted desktop display.
```

For specific purposes you may need to assign several WETs to the profile.

To assign a WET to a profile:

1. Access the properties of the profile.
   - *See Viewing Profile Characteristics.*

2. Select the **Assign a WET** tab.
3. Click **New** .
4. In the **WET** field, select the Working Environment Template you want to assign to the profile.
5. Select **Create a Desktop Manager**.

   - *To reuse a Desktop Manager, keep **Reuse existing Desktop Manager** and in the drop-down list select the Desktop Manager.*

6. Click **Next**.
7. (Optional) In the **Name** field, modify the default desktop manager name.

   This can be useful if you need to reuse this desktop manager for another WET assignment.

8. Click **Connect** .
9. (Optional) In the query field, enter the characters to search for.
10. Click Find .
11. Select the desktop(s) you want to define for the profile.
12. Click **OK**.
    The desktops associated with the **Desktop Manager** are specified.
    You must define each desktop use context.
13. In the desktop list, for each desktop, in the **Device** column, select the device type adapted to the desktop.

    ```
    Example: Tablet.
    ```

14. Click **OK**.
    The selected WET is assigned to the profile and its associated desktops are defined with their use context.

Defining the applications accessible to the users of a profile (non WET-based configuration)

- *To modify a profile provided by **MEGA**, you must create a new profile; see Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.*

So that a user of a profile can connect to an application, you must connect this application to the profile concerned.

- *See Connection diagram (without WET).*

All desktops connected to the application are then accessible. To enable access to only certain desktops of the application, see Defining the application desktops accessible to the users of a profile (non WET-based configuration).

To define applications available for a profile:

1. Access the properties of the profile.

   - *See Viewing Profile Characteristics.*

2. Select the **Available Applications** tab.
3. In the toolbar, click **Connect** .
4. (Optional) In the query field, enter the characters to search for.
5. Click **Find** .
6. Select the application.

   - *You can add several applications.*

7. Click **Connect**.
8. Click **OK**.
   The applications are connected to the profile.

Defining the application desktops accessible to the users of a profile (non WET-based configuration)

A user can connect to an application via customized desktops according to actions to be performed.

If an application contains several desktops, you can specifically define application desktops that are accessible to the concerned profile.

– *See Restricting access to the desktops of an application.*

To do this, you must connect to the profile:

- the application containing the desktops.

  – *See Defining the applications accessible to the users of a profile (non WET-based configuration).*

- the desktops you want the users of the profile can connect to.

  – *The application desktops that are not connected to the profile are not accessible to users of the profile.*

  – *To modify a profile supplied by **MEGA**, **MEGA** recommends you create a new profile, see Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.*

To define application desktops available for a profile:

Prerequisite: The application accessible to users of the profile is defined.

– *See Defining the applications accessible to the users of a profile (non WET-based configuration).*

1. Access the properties of the profile.

   – *See Viewing Profile Characteristics.*

2. Select the **Available Desktops** tab.
3. In the toolbar, click **New** .
4. In the **Name** field, in the drop-down list, select the desktop you want to add.

   – *You can add several desktops.*

5. Click **OK**.
   The desktops are connected to the profile.

Associating a terminology with a profile

) *A Terminology defines a set of terms used in a specific context instead of the standard term.*

– *For information on creating and managing a Terminology, see **HOPEX Power Studio - Renaming HOPEX Concepts**.*

To associate a terminology with a profile:

1. Access the properties of the profile.

   – *See Viewing Profile Characteristics.*

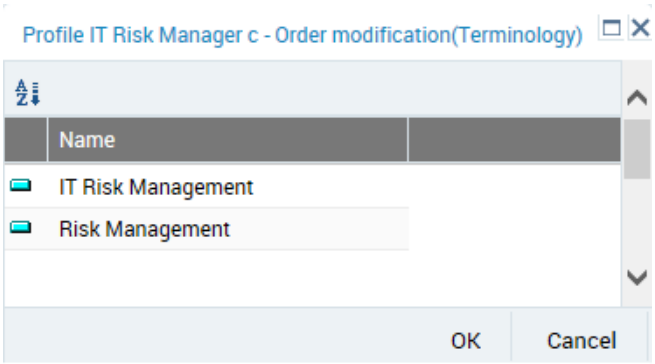
2. Select the **Terminology** tab.
3. In the toolbar, click **Add** .

4. In the **Name** field, in the drop-down list, select the terminology you want to add.
   - *You can add several terminologies.*
5. Click **OK**.
   The terminology is connected to the profile.

If you associate more than one terminology with the profile, you must define an order of priority for them.

To define the priority of the terminologies of a profile:

1. Access the properties of the profile.
   - *See Viewing Profile Characteristics.*
2. Select the **Terminology** tab.
3. In the toolbar, click **Reorganize** .
4. Drag and drop to place the priority terminology at the top.



In the example above, the terms of the Risk Management terminology are used when they are not defined in the IT Risk management terminology.

Defining the object types available for a profile

You can define which specific object types are available for a profile:

- document categories
- document models
- Report DataSet Definitions
- widgets

To define the object types available for a profile:

1. Access the properties of the profile.
   - *See Viewing Profile Characteristics.*
2. Select the **Available Types** tab.
3. Select **Available Objects**.
4. In the toolbar, click **Connect** .
5. In the list, select the object types to make available for the profile.
6. Click **Add**.
   The object types selected are made available for the profile.

Assigning a Profile to a Person

– *A person may have several profiles.*

P **A user must have at least one profile assigned to be able to connect to HOPEX.**

Assigning a profile to a person defines:

- the profile assigned
- the repository concerned by the assignment
- (optional) a validity period of an assignment.
- (optional, with read-only access to the repository) connection repository snapshot

Repository Snapshot:

) *A repository snapshot defines repository state at a given moment.*

The connection repository snapshot defines the state of the repository to which the users of a profile connect.

– *Repository snapshots are available for RDBMS repositories.*

To define a repository snapshot, a repository snapshot must have been previously created.

– *To create a repository snapshot, see the **HOPEX Common Features - Repository Snapshots** guide.*

Restrictions:

- The **Administrator profile** attribute of a profile allows to assign an administrator type profile.

  – *See Administrator profile.*

  Only administration dedicated profiles (HOPEX Administrator, HOPEX Administrator - Production, User Management Web Administrator) allow to assign any profile (including administration dedicated profiles) to persons.
- A Functional Administrator of a Solution can only assign this Solution dedicated profiles.

  E.g.: The **Application Design Functional Administrator** profile allows to assign the **Application Design Viewer**, the **Application Designer**, and the **UML Designer** profiles.

  – *In the profile property pages, **Assignable Profiles** lists the profiles it allows to assign, see Assignable profiles.*

To assign a profile to a person:

1. Access the profile assignment management window.

   – *See Opening the profile assignment window.*
2. In the **Persons** pane, select the person to whom you want to assign a profile.
3. In the **Profile Assignment** pane, click **New** .
   The profile assignment creation wizard opens.

4. In the **Profile assigned** field, click the drop-down menu and select the profile you want to assign to the person.
   - *To execute a filtered query on a profile, click the arrow and select **Connect Profile**.*
5. (optional) In the **Validity start date** field, use the calendar to define the start date of profile validity.
6. (optional) In the **Validity end date** field, use the calendar to define the end date of profile validity.
7. (optional) In the **Repository** field, click the drop-down menu and select the repository to which you want to assign the profile.
   - *By default, the profile is assigned to all repositories.*
8. (optional, with read-only access to the repository) In the **Connection Snapshot** field, select a connection repository snapshot.
9. Click **OK**.
   The profile is assigned to the person on the selected repository for the specified duration.

Performing a Mass Profile Assignment to Persons

You can perform a mass profile assignment to persons. In this case, the profile assignment to the person is defined automatically with a permanent validity on all repositories.

- *To modify the configuration of the assignment, see Assigning a Profile to a Person.*

To perform a mass profile assignment to persons:

1. Open the profile properties dialog box.
   - *See Viewing Profile Characteristics.*
2. In the **Persons** tab, click **Connect** .
3. (Optional) In the query field, enter the characters to find.
4. Click **Find** to run the query.
   The list of persons who can be connected to the profile appears.
5. In the result list, select the persons you want to connect to the profile.
   - *Use the [Ctrl] key to select more than one person at the same time.*
6. Click **OK**.
   The profile is assigned to all selected persons, on all repositories, without a validity limit.
   - *To modify the configuration of the assignment, see Assigning a Profile to a Person.*

Deleting a Profile

P **If you delete a profile that is the only profile assigned to a user, this user can no longer connect to HOPEX.**

To delete a **Profile**:

1. Open the **Profiles** management window.
   - *See Opening the profile management window.*
2. In the **Profile** tab, select the profile you want to delete.
   - *You can select more than one.*
3. Click **Delete** .
   The dialog box for deleting a profile opens.
4. Click **Delete**.
   The profile is deleted from the environment

ACCESS TO USER MANAGEMENT

See:

- Accessing User Management and UI Access Management Folders.
- Viewing Person Characteristics.
- Viewing the Login Characteristics.

Accessing User Management and UI Access Management Folders

This section describes how to access management windows for users and profiles.

See:

- Accessing User Management and UI Access Management Folders
- Opening the User Management window
- Opening the profile management window

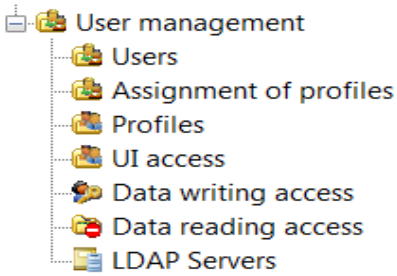
Accessing User Management and UI Access Management Folders

The **User management** folder contains all sub-folders for management of *users*, that is:

- *persons* and *logins*,
- *profiles*,
- *object UI access* and *general UI access*.

To access user management and UI access management folders:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.



From the **User management** folder of an environment, you can access management windows:

- of **Users**
  User characteristics are defined in the following tabs:
  - **Persons**
  - **Logins**

  – *See Opening the User Management window.*
- of **Profile Assignments**

  – *See Opening the profile assignment window.*
- of **Profiles**

  – *See Opening the profile management window.*
- of **UI access**
  Object UI access and general UI access are characteristics defined in tabs:
  - **Object UIs**
  - **General UIs**

  – *See Managing UI Access (Permissions).*
- of **Data reading access**

  – *See Managing Data Reading Access.*
- of **Data writing Access**

  – *See Managing Data Writing Access.*
- **Authentication**

  – *See Configuring LDAP authentication.*

Opening the User Management window

Characteristics of *users* are managed in the **Users** window.

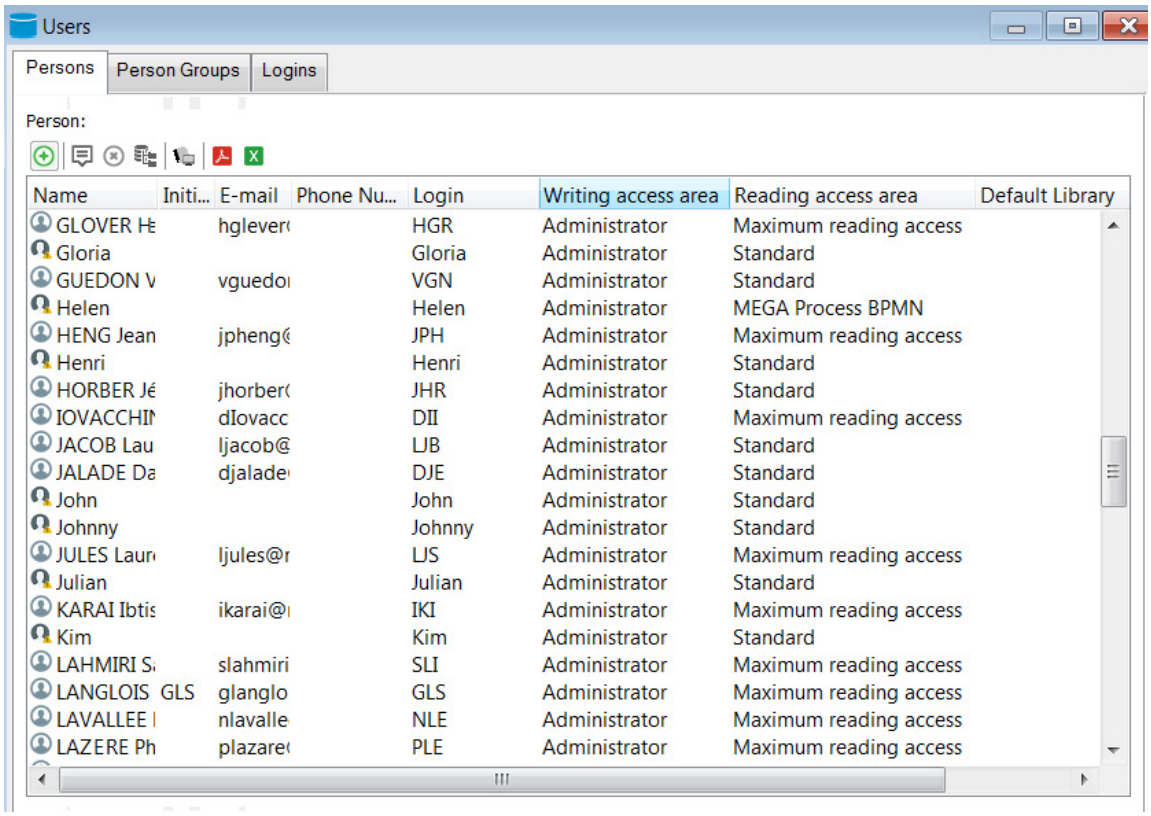
– *To manage user options, see Managing Options.*

To manage users:

1. Connect to **HOPEX Administration** and select the environment concerned.

   – *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.

3. Right-click the **Users** folder and select **Manage**.
The **Users** window opens.



The **Users** window contains tabs:

- **Persons**, which lists all of the persons in the environment and details personal characteristics of each person.
  - *See Person Properties.*
- **Person Groups**, which lists all of the person groups in the environment and details their characteristics.
  - *Managing person groups is a functionality available in **HOPEX** (Web Front-End). For more information on managing this functionality, see the Web HOPEX Administration - Supervisor guide.*
- **Logins**, which lists all the users in the environment and details their characteristics.
  - *See Login Properties.*

From the **Users** management window you can:

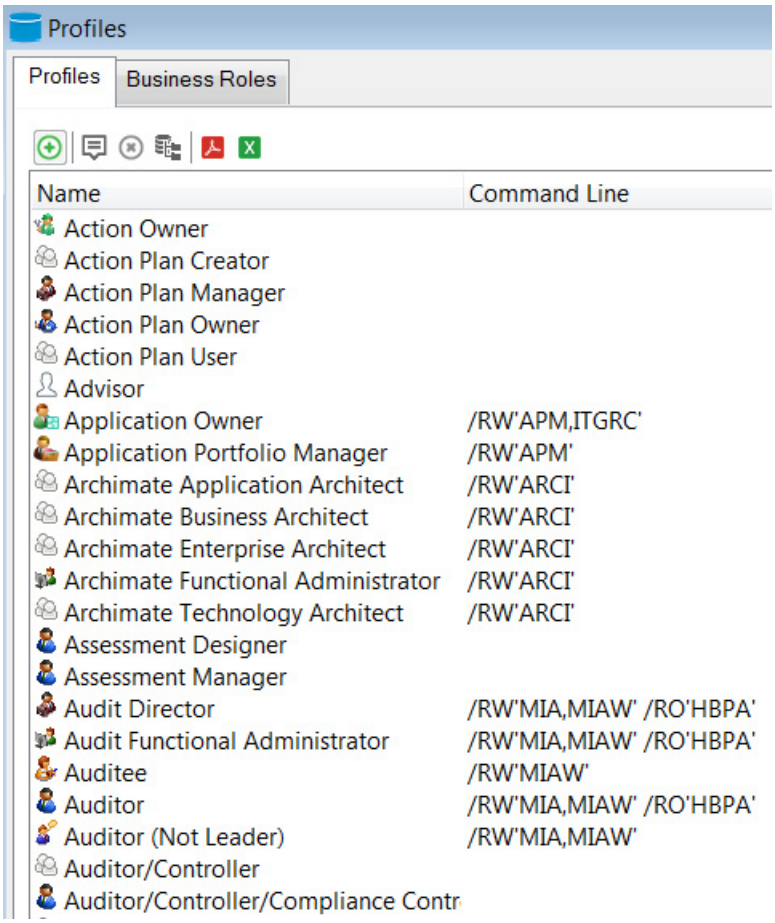
- create:
  - persons
    - *See Defining a Person.*
  - user management
    - *See Creating Users.*
- configure characteristics:
  - of a person
    - *See Defining a Person.*
  - of a login
    - *See Defining the Login of a Person.*
- delete users
    - *See Deleting Users.*
- modify:
  - user properties
    - *See Modifying the Properties of a User.*
  - user passwords
    - *See Creating or Modifying the Password of a User (Windows Front-End).*
- access user options
    - *See Managing Options.*

Opening the profile management window

To manage profiles:

1. Connect to **HOPEX Administration** and select the environment concerned.
    - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.

3. Right-click the **Profiles** folder and select **Manage**.
The **Profiles** management window opens.



The **Profiles** window contains the following tabs:

- **Profiles**, which lists all the profiles in the environment and detailing their characteristics. For each profile, it details the products accessible (optional).
  - *See Managing Profiles.*
- **Business Roles**, which lists all the business roles in the environment used by functional administrators to manage them.

From the **Profiles** management window you can:

- create profiles
  - *See Creating a Profile.*
- customize an existing profile
  - *See Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.*
- consult profile characteristics
  - *See Viewing Profile Characteristics.*
- configure profiles
  - *See Configuring a Profile.*
- perform a mass profile assignment to persons
  - *See Performing a Mass Profile Assignment to Persons.*
- delete profiles
  - *See Deleting a Profile.*

Opening the profile assignment window

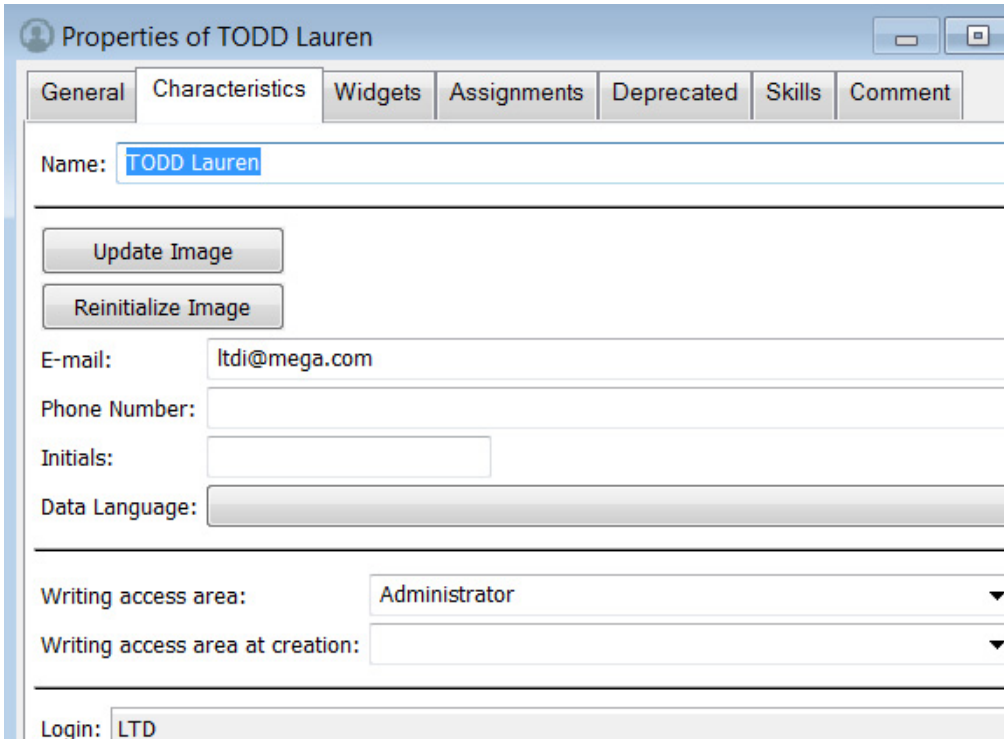
To open the assignment of profiles window

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.
3. Right-click the **Assignment of profiles** folder and select **Manage**.
   The **Assignment of profiles** window appears.

From the **Assignment of profiles** management window, you can assign profiles to persons.

- *See Assigning a Profile to a Person.*

Viewing Person Characteristics



The icon for a person is represented by:

- when the person is created (name and writing access area defined) but does not have a login.
- when the person has a login but is not fully configured (e-mail or profile assignment is not defined).
- when the person is configured as a **HOPEX** user:
  name, writing access area, login, and e-mail address are specified and a profile is assigned to the person.

  – *See Defining a Person, Creating Users and Assigning a Profile to a Person.*

To view the person characteristics:

1. Open the **Users** management window.

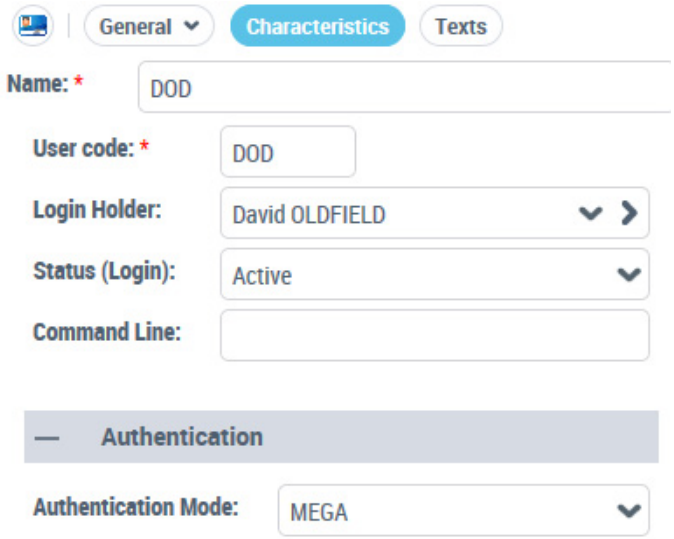
   – *See Opening the User Management window.*
2. Select the **Persons** tab.
   The list of persons appears with the characteristics of each person.

   – *You can modify these characteristics directly in this window (with a click in the corresponding field).*
3. In the **Persons** list, select the person.
4. In the toolbar, click **Properties** .

5. From the **Properties** window for the person, click:
   - the **Characteristics** tab to define or modify the person properties.
     - *See Person Properties.*
     - *See Defining a Person.*
   - the **General** tab, **History** sub-tab to display the actions performed on the person.
   - the **Assignments** tab to display and assign profiles to the person.

Viewing the Login Characteristics

- *For detailed information on characteristics of a login, see Login Properties.*
- *To configure a login, see Defining the Login of a Person.*



To view the login characteristics:

1. Open the **Users** management window.
   - *See Opening the User Management window.*
2. Select the **Logins** tab.
3. Select the **Login**.
4. In the toolbar, click **Properties** .
   The **Properties** dialog box of the selected login opens.

MANAGING USERS

For an overview of actions to be performed to create and define a user see Actions to be Performed to Define a User.

The following points are covered here:

- configuration:
  - Creating Users
  - Defining a Person
  - Creating the Login of a Person
  - Defining the Login of a Person
  - Modifying the Properties of a User
- management:
  - Managing User Inactivity
  - Preventing User Connection
  - Deleting Users
  - Creating or Modifying the Password of a User (Windows Front-End)
  - Exporting the Repository Users

Creating Users

– *Instead of creating users one by one, you can import a list of persons. This list can for example come from an LDAP server (see Synchronization with a company directory).*

A user depends on an environment. To create a user, you must connect to the environment to which the user will be attached.

A user is a person with a login. To create a user, you must create a person with its login, or create the login of a person already created.

– *For detailed information on characteristics of a person, see Person Properties.*

– *For detailed information on characteristics of a login, see Login Properties.*

– *To import users from an LDAP directory, see Configuring LDAP authentication.*

Once the user is created, he/she automatically receives an e-mail to define his/her connection password.

– *This e-mail is sent only when HOPEX SMTP settings are configured (see Specifying SMTP configuration). Otherwise, the **Password** field is available at user creation. You must then define this temporary password, that the user has to change at first connection.*

To create a user:

1. Open the **Users** management window.

   – *See Opening the User Management window.*

2. In the **Users** window, select the **Persons** tab.

3. Click **New** .
   The **Creation of Person - Characteristics** dialog box opens.
4. In the **Name** field, enter the name of the person.

   `Example: DUBOIS Guillaume`

   - *Remember to use the same format for all persons.*
5. (Optional, but recommended) In the **E-mail** field, enter the e-mail address of the person.

   - *The e-mail address is necessary, for example, for distributing documents, receiving notifications and questionnaires, or when a Web user lost his/her password.*
6. In the **Login** field, enter the login of the person.

   - *If you do not enter the Login, it will automatically take the value entered in the **Name** field.*
   - *A **Login** is unique and can be assigned to only one Person or Person Group.*
   - *A **Person** can have only one **Login**.*

   `Example: GDS`
7. (If available) In the **Password** field, enter a temporary password for the user.
8. (With the **HOPEX Power Supervisor** technical module) Using the drop-down menu in the **Writing Access Area** field, select the value of the writing access area of the user.

   - *For more details on writing access, see Managing Data Writing Access.*
9. (If required, with the **HOPEX Power Supervisor** technical module) Using the drop-down menu in the **Reading Access Area** field, select the value of the reading access area of the user.

   - *By default at creation, the user is connected to "Standard" reading access area. This field only appears if reading access management has been activated, see Managing Data Reading Access.*
10. Click **Finish**.
    The user is created (a login is associated with the person): the person appears in the list of persons and its login appears in the list of logins.

    - *You must configure the login of the user, see Defining the Login of a Person.*

    The user receives an email to define his/her password.

Defining a Person

) ***Person** represents a physical person or a system.*

- *For more information on properties of a person, see Person Properties.*

From the profile properties window of a person, you can define:

- name of the person
  - *See step 1.*
- image of the person
  - *See step 2.*
- e-mail address of the person
  - *See step 3.*
- phone number and initials of the person
  - *See step 4.*
- data language of the Web user
  - *See step 5.*
- default library to store objects created by the person
  - *See step 6.*
- writing access area of the user
  - *See step 7.*
- reading access area of the user
  - *See step 7.*
- login of the user
  - *See step 8.*
- if the user belongs to a person group
  - *See step 9.*

To configure a **Person**:

1. Access the properties of the person.
   - *See Viewing Person Characteristics.*
2. (Optional) To add or update the image of the person, click **Update Image**, select the image and click **OK**.
   - *The image is stored in binary on an attribute of the person. To delete the image, click **Reinitialize Image**.*
3. (Optional, but recommended) In the **E-mail** field, enter the e-mail address of the person.
   - *The e-mail address is required, for example, to initialize the user password, for distributing documents, receiving notifications and questionnaires, or when a Web user lost his/her password.*
4. (Optional) Enter the **Phone Number** and the **Initials** of the person.
5. (Web specific, optional) In the **Data Language** field, you can define a specific data language for this user.
   - Click the arrow and select **Query Language**.
   - In the query wizard, select the data language (objects) and click **OK**.
   - *If the field is not specified, the default data language is the interface language defined in environment options (**Options/ Installation/User Management**: **Data Language**).*
   - *See Managing Languages.*
6. (Optional) In the **Default Library** field, click the arrow and select the default library in which objects created by the user are stored if the creation context does not define one.

7. (Optional, with the **HOPEX Power Supervisor** technical module) You can modify the values at the following levels:
   - user writing access via the drop-down menu in the **Writing Access Area** field.
     - *By default, all users are connected to the only writing access area that exists: "Administrator". For more details on writing access, see Managing Data Writing Access.*
   - user writing access at creation via the drop-down menu in the **Writing Access Area** field.
   - reading access via the drop-down menu in the **Reading Access Area** field.
     - *This field only appears if reading access management has been activated, see Managing Data Reading Access.*
   - reading access at creation via the drop-down menu in the **Writing Access Area** field.
     - *This field only appears if reading access management has been activated, see Managing Data Reading Access.*
8. So that the person can connect to **HOPEX**, the person must have a **Login**.
   - *See Creating the Login of a Person.*
   - *See Defining the Login of a Person.*
9. (optional) If necessary select **Belongs to a Person Group**
10. Click **OK**.
    The person is configured.

Creating the Login of a Person

To connect to **HOPEX**, a person must have a Login. When you create a person, his/her login is automatically created.

To create the login of a person:

1. Access the properties of the person.
   - *See Viewing Person Characteristics.*
2. In the **Login** field, click the arrow and select **Create Login**.
   The **Creation of Login** dialog box opens. The name of the login is already entered with the name of the login holder.
3. (Optional) In the **Name** field, modify the login name.
   - *A login is unique; it can be assigned to one Person or one Person Group only.*
   - *A **Person** can have only one **Login**.*

   `Example: GDS`
4. In the **User Code** field, enter the user code to be associated with the login.

   `Example: GDS`
5. Click **OK**.
   The login of the user appears in the **Login** field.

Defining the Login of a Person

From the Login properties window, you can:

- *See Login Properties.*

- define the login name, the user code associated with the login and the login holder
  - *See step 1.*
- modify user status (inactive)
  - *See step 2.*
- restrict user access to certain products
  - *See step 3.*
- modify user authentication mode
  The value of this parameter is inherited, at user creation, from the value of the **Authentication Mode** option defined in the environment options (**Installation > User Management > Authentication Mode**). If you change the option value at environment level after the user creation, the user is not impacted.
  - *See step 4.*

To define the login of a person:

1. Display the **Characteristics** tab of the login properties.
   - *See Viewing the Login Characteristics.*
   - The login **Name** and **User Code** attributes are already created, but you can modify these if necessary.
) *A **login** is unique and defined for a person or person group.*
) *The **User code** is the short identifier (upper case) of the user. It serves as a basis for naming user private workspaces.*
   - The **Login Holder** is the person associated with the login.
2. (Optional) Modify the **Status (Login)** field value, which defines if the user is active or not.
   - *See Inactive user (Status).*
3. (Optional) In the **Command Line** field, define the products available to which the user has access.
   To restrict user access to products A and B, enter the command:
   /RW'<accessible Product A code>;<accessible Product B code>;<...>'

   For example: You have licenses for products **HOPEX Business Process Analysis**, **HOPEX IT Portfolio Management** and other **HOPEX products**. To authorize only the **HOPEX Business Process Analysis** and **HOPEX IT Portfolio Management** modules to a user, enter:

   /RW'HBPA;APM'

   - *To find out the product code, see Access your list of available products or the online documentation: **Concepts > Products**.*

   P **If a user is connected to a profile and the user and profile each have access to products restricted by the Command Line attribute, the products accessible to the user are at the**

intersection of the values of the Command Line attribute of the user (on his/her login) and profile.

4. (If needed) In the **Authentication Mode** field, click the arrow and modify the authentication mode. The default value is "MEGA".
   - *See Authentication mode.*
5. Click **Apply**.

Modifying the Properties of a User

You can modify user properties. For each user you can modify properties of:

- person:
  - its name
  - image
  - e-mail address
  - phone number
  - initials
  - data language
  - default library
  - writing access area
  - reading access area
  - group
  - profile assignments (connection)
  - object assignments (business roles)
    - *See Person Properties.*
    - *See Viewing Person Characteristics.*
    - *See Defining a Person.*
- login:
  - its name
  - user code

    P **To assure consistent actions history, the user code should not be modified.**
  - status
  - accessible products (Command Line)
  - authentication mode
    - *See Login Properties.*
    - *See Viewing the Login Characteristics.*
    - *See Defining the Login of a Person.*

Preventing User Connection

When you no longer want a user to connect to **HOPEX**, but want to retain trace of his/her actions, you must render the user inactive but not delete it from your repository.

To render a user inactive:

1. Open the **Characteristics** tab of the login properties dialog box.
   - *See Viewing the Login Characteristics.*
2. In the **Status (Login)** field, select "Inactive".
3. Click **Apply**.

Deleting Users

P **When you delete a user from the repository, the commands connected to this user become orphans and you lose part of the history saved in logs. To remove a user but keep its history of commands, see Preventing User Connection.**

To delete a user:

1. Access the User Management page.
   - *See Opening the User Management window.*
2. In the **Persons** tab, select the person to be deleted and click **Delete** .
   - *You can select more than one.*

   The **Delete Objects** dialog box opens.
3. (If necessary) In the **Delete** column, modify the deletion selection of a person and her/his login.
4. Click **Delete** to confirm deletion.
   The person and login are deleted from the repository.

   P **All traces of user actions are lost.**

Creating or Modifying the Password of a User (Windows Front-End)

- *To manage the password of a Web user, see the HOPEX Administration - Supervisor guide:*

The user's password is mandatory.

- *Only a user with **Administrator** type profile has management rights. He is the only user type that can create a user password. To grant administrator access rights to a user, see Defining the Login of a Person.*

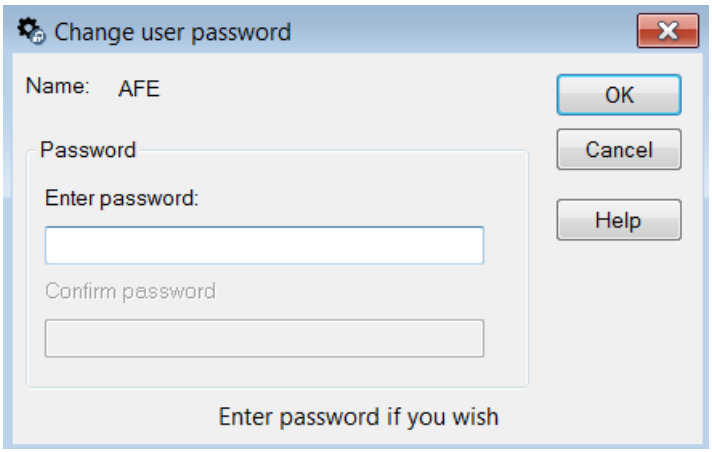
The password can be modified later:

- in **HOPEX**, by the user, from the **HOPEX** connection window.
- in **HOPEX Administration** , by the Administrator, notably when the user has forgotten his/her password.

To create or modify the temporary password of a user (Windows Front-End):

1. Open the **Users** management window.
   - *See Opening the User Management window.*
2. Select the **Logins** tab.

3. In the logins list, right-click the login concerned and select **Password**.
The **Change User Password** dialog box opens.



4. Enter the password you want to assign to the user.
   - *Do not use separation character (*"!", "$", "@", ":", ";").

   Password confirmation entry is required.
   - *If the two entries are not identical, the entry must be corrected.*
5. Click **OK**.
The user's temporary password is saved.

Exporting the Repository Users

You can export the repository users. This can be useful for example to copy users and their characteristics associated with an environment into another environment.

- *See Exporting HOPEX Objects.*

MANAGING USER OPTIONS

For specific requirements, you can modify default values of certain **Options** (see Accessing Options).

See:

- Configuring the Metamodel Access
- Authorizing Deletion of a Dispatched Object
- Making a Comment Mandatory on Dispatch
- Managing User Inactivity

Configuring the Metamodel Access

With the **Metamodel Access** option (**Options > Repository** menu) , you can restrict the view of **HOPEX** objects or functions according to user skill level.

This option can be defined at environment, profile or user level according to the requirement.

Metamodel access levels are:

- **Beginner**
  For introduction to **HOPEX**. Only basic objects are visible. This level allows very simple modeling.
- **Intermediate** (default value)
  For standard use of **HOPEX**. Almost all object types, links and non-technical attributes are visible.
- **Advanced**
  For advanced use of **HOPEX**. All objects, links and non-technical attributes are visible, including those that require advanced skills for their use. Only object types and attributes which are present only for compatibility with previous versions are filtered. Certain technical object types are visible. The user can carry out simple customizations of the **HOPEX** platform.

  For example, this level enables access to:
  - certain navigation windows, with a restricted view of objects (**MetaStudio**, **Utilities**)
  - the **Administration** navigation window, which gives access to reading and writing access areas.
  - the **Repository Activity** (Navigation window in **HOPEX,** see Viewing the Repository Update Log).

- **Expert**
  This level displays all object types, links, and attributes, as well as the abstract metamodel. All HOPEX platform customizations are available.

  P **Specify this access level only for a highly expert user or a particular profile (e.g.: HOPEX Customizer).**

  This is the level for example that offers:
  - all functions of **HOPEX Power Studio** (**MetaStudio** navigation window)
  - report template creation (**Utilities** navigation window)
  - certain advanced commands (e.g.: update logfile export at particular intervals).

Authorizing Deletion of a Dispatched Object

Users working in a public workspace can delete objects.

By default, users working in a private workspace are not authorized to delete dispatched objects, even if these have been created by the user wishing to delete.

The **Authorize dispatched object deletion from private workspace** option (**Options > Repository** folder) allows the user to delete dispatched objects from a private workspace, irrespective of the creator.

This authorization complements object deletion rights defined elsewhere for the profile or user.

Making a Comment Mandatory on Dispatch

With the **Comment on dispatch** option (**Options > Repository** folder) users must enter information in the **Dispatch comment (report)** pane when they dispatch their work.

Managing User Inactivity

You can specify for how long user session time can remain inactive before closing.

M *This option can be useful for example for security requirements, or to ensure that all sessions are closed before starting a batch program.*

By default, user inactivity management is not activated.

Activating/Deactivating user inactivity management

To activate/deactivate user inactivity management:

1. Access **Options** at the environment level.
   - *See Accessing Options.*

2. In the **Options** tree, select **Workspace**.
3. In the right pane:
   - to activate user inactivity management, select **Automatic Session Timeout**.
   - to deactivate user inactivity management, clear **Automatic Session Timeout**.

Managing user inactivity

User inactivity management is taken into account if the **Inactivity Management** option is selected.

– *See Activating/Deactivating user inactivity management.*

To manage user inactivity:

1. Access **Options** at the environment level.

   – *See Accessing Options.*

2. In the **Options** tree, select **Workspace**.
3. In the right pane, select values for options:

   – *Duration of inactivity requiring authentication must be less than that closing **HOPEX.***

   - **Period of inactivity requiring authentication**

   When this duration has been reached, the user receives a message requesting authentication.

   – *This duration warns the user before he/she is disconnected if the **Duration of inactivity before closing HOPEX** option has been specified.*

   - **Duration of inactivity before closing HOPEX**

   When this duration has been reached, the user is disconnected and **HOPEX** closes without warning.

   – *So that the user is warned of disconnection, the **Period of inactivity requiring authentication** must be specified and its value less than that of the **Duration of inactivity before closing HOPEX**.*

AUTHENTICATION IN HOPEX (WINDOWS FRONT-END)

Authentication is a process consisting of verifying that a person corresponds to his or her declared identity. In IT networks, authentication is usually based on a connection name and a password.

Defining Default Authentication Mode

In **HOPEX** (Windows Front-End) you can set authentication to:

- ***LDAP***
  If your enterprise has an LDAP authentication system, it is preferable to manage your authentication using an LDAP directory.
  - *See Defining default authentication mode to LDAP.*
- ***Standard*** (by default)
  If you have no standard authentication system in your enterprise, you can use the authentication system managed within **HOPEX** platform.
  HOPEX proposes the following authentication modes:
  - **MEGA** authentication (by default)
  - **Windows** authentication, which corresponds to Single Sign On.
  - **LDAP** authentication

Viewing default authentication mode

In the environment options, you can view and modify the default **Defined Authentication Mode**.

To view default authentication mode:

1. Access environment options.
   - *See Modifying options at environment level.*
2. In the options tree, expand the **Installation** folder and select **User Management**.
3. In the right pane, view the value of the **Authentication Mode** option.
   By default at installation, the authentication mode is "Standard", it is **MEGA** authentication mode.

Defining default authentication mode to LDAP

Users are managed in an LDAP directory and authentication is managed by the LDAP directory.

- *The authentication mode of users already created is not impacted (they keep their defined authentication mode). Only users created after the default authentication mode change are concerned.*

To define default LDAP authentication mode:

1. Access environment options.
   - *See Modifying options at environment level.*

2. In the options tree, expand the **Installation** folder and select **User Management**.
3. In the right pane, specify "LDAP" for the **Authentication Mode** option.

Modifying the user authentication mode

User authentication mode is defined on the login by the **Authentication Mode** parameter. The value of this parameter is inherited at user creation from the value of the **Authentication Mode** option defined in the environment options (see Viewing default authentication mode).

To modify the authentication mode of a user, see Defining the Login of a Person.

Configuring Windows Authentication

Synchronization with a company directory

Active Directory is a directory service designed principally for Windows environments.

Active Directory is a directory referencing persons (name, first name, phone number, etc) and objects such as servers, printers, applications, databases, etc.

Active Directory enables inventory of all information concerning the network (users, machines and applications). Active Directory is at the heart of all network architecture and its purpose is to enable users to find and access any resource identified by the service.

Active Directory is based on standards DNS, LDAP, Kerberos, etc.

Associating a Windows user with a HOPEX user manually

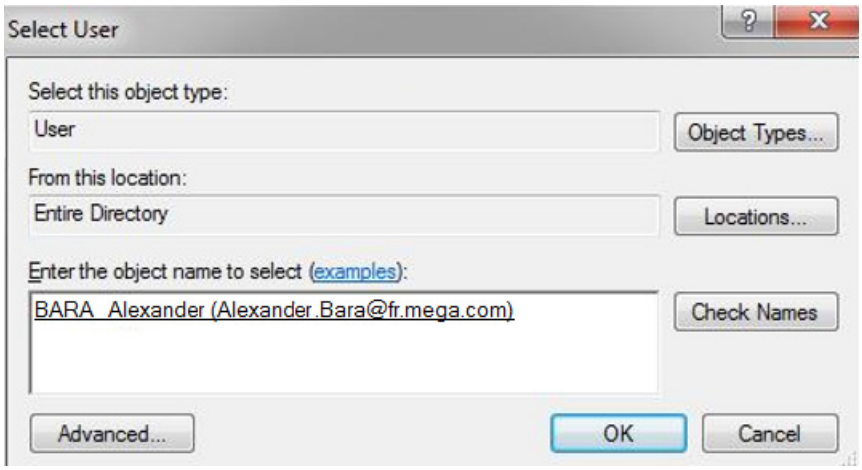
You can connect a single **HOPEX** user to a Windows user.

To indicate the Windows identifier of a **HOPEX** user in the Administration application:

1. Open the properties dialog box of the user login.
   - *See Viewing the Login Characteristics.*
2. Select the **Characteristics** tab.
3. In the **Authentication Mode** field drop-down list, select "Windows".
   The **Windows Login** field appears.
4. In the **Windows Login** drop-down list ▸, select **Select Windows User**.
   The **Select User** dialog box opens.
5. In the **Enter the object name to select** frame, enter the user name.

   *To find users with the help of a wizard, click **Advanced**.*

6. Click **Check Names**.



7. Click **OK**.
The Windows identifier of the user is displayed.

Connection in case of unique authentication

(Windows Front-End) If the **HOPEX** user is:

- different from that identified by the Windows session opened on the workstation, the user must enter his/her own Windows password.
- the same as that identified by Windows session opened on the workstation, the user does not need to enter his/her password.

(Web Front-End) The user must enter his/her Windows password.

Single sign-on precautions

A system repository in which all users have been changed to Single Sign On connection mode (Windows) can no longer be opened outside the company in which the repository was created.

If you want the repository to be opened outside your company (by the **HOPEX** technical support team for example), ensure that at least one user remains in MEGA authentication mode.

Configuring LDAP authentication

– *LDAP authentication is available only if you have **HOPEX Power Supervisor** technical module.*

An LDAP directory enables storage of user data of the enterprise.

HOPEX Administration allows you to create users authenticated at LDAP server level.

– *Only users (example: Administrator) with **HOPEX Administrator** or **User Management Administrator** profile can enter LDAP data, see The Administration profiles provided.*

Configuring LDAP authentication

To configure LDAP authentication:

1. Create an LDAP server in **HOPEX Administration**.
   – *See Creating an LDAP server.*
2. Specify parameters of your LDAP server.
   – *See Configuring the LDAP server.*
3. (Optional) You can:
   - configure LDAP parameters
     – *See Configuring an LDAP parameter.*
   - modify LDAP import parameters
     – *See Modifying LDAP directory import content.*

Once LDAP authentication is configured:

- you can import persons from the LDAP directory.
  – *See Importing persons from an LDAP server.*
- or you can manually map a **HOPEX** user group with a user group declared in your LDAP server.
  – *For more information on managing person groups, see the **HOPEX Administration - Supervisor Web** guide.*
  – *When connecting to **HOPEX**, the authentication service uses the HOPEX Login and password of the user to authenticate the user with the list of available LDAP servers.*

Accessing LDAP server management

To access LDAP server management:

› From **HOPEX Administration,** in the **User Management** folder, right-click **Authentication** and select **Manage LDAP Servers**.
– *See Accessing User Management and UI Access Management Folders.*

The **Manage LDAP Servers** window opens.

Creating an LDAP server

The LDAP server is the server on which the LDAP directory is installed.

The LDAP directory can be an Active Directory directory.

To create an LDAP server:

1. Access LDAP server management.
   – *See Accessing LDAP server management.*
2. In the **Manage LDAP server** window menu bar, click **New** .

3. In the creation of LDAP server dialog box, enter the **Name** of the LDAP server and click **OK**.
   The new LDAP server appears in the list of LDAP servers.
   You must configure the LDAP server, see Configuring the LDAP server.

Configuring the LDAP server

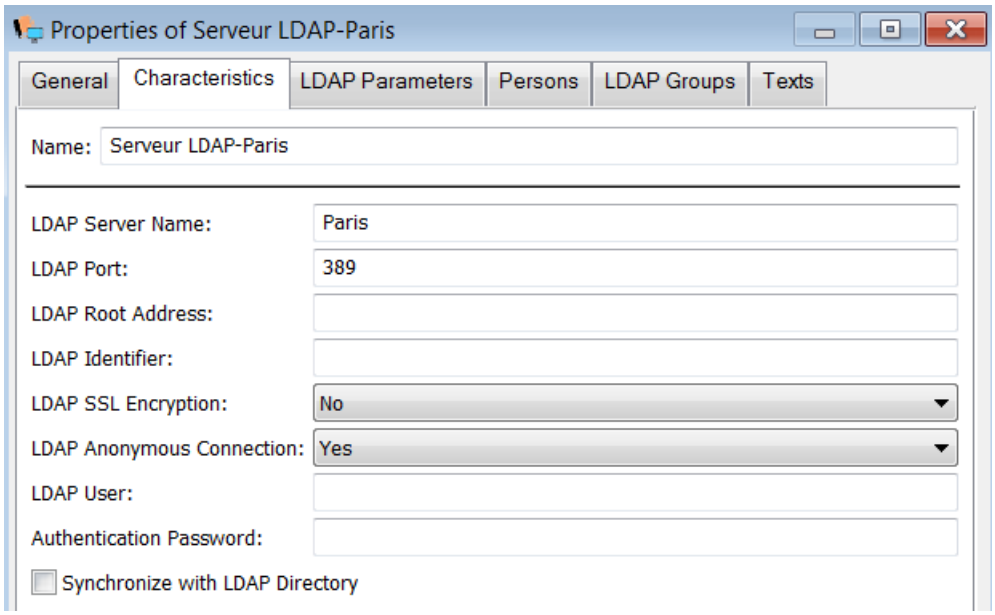
LDAP server configuration is restricted to users with HOPEX Administrator or User Management Administrator profile.

To configure an LDAP server:

Prerequisite: the LDAP server is already created.

- *See Creating an LDAP server.*

1. Access LDAP server management.
   - *See Accessing LDAP server management.*
2. Select the new LDAP server and click **Properties** .



3. In the **Characteristics** tab, complete the following fields:
   - **LDAP Server Name**: name of the server hosting the LDAP directory.
   - **LDAP Port**: LDAP communication bridge

     `Example: 389`
   - **LDAP Root Address**: root address of th LDAP server. This is an important attribute to limit query for a user in the LDAP directory or to address a particular forest.
   - **LDAP Identifier**: this is the LDAP attribute enabling unique identification of a user

     `Example: SAMAccountName, UID`
   - **LDAP SSL Encryption**: select **Yes** if you want LDAP directory connection to be SSL protocol encoded
   - **LDAP Anonymous Connection** : if you select **No**, you must specify the user via which LDAP directory connection will be made, as well as the user password
     - *Only an administrator user can connect anonymously to an LDAP server.*
   - **LDAP User**: enter the identifier of the LDAP user used for LDAP directory connection. If connection is anonymous, this field should not be completed.
     - *This user must have reading rights on data that **HOPEX** needs to access (example: LDAP person group, membership of a group in LDAP, e-mail in LDAP, etc.).*
   - **Authentication Password**: enter the password of the LDAP user used for LDAP directory connection. If connection is anonymous, this field should not be completed.
   - (If needed) Select **Synchronize with LDAP directory** to synchronize LDAP parameters defined on the LDAP server (HOPEX) with the updates of the LDAP directory parameters.
     - *See Configuring an LDAP parameter.*
4. Click **OK**.
   The LDAP server is configured.

   You can also:
   - configure an LDAP parameter, see Configuring an LDAP parameter.
   - modify content of LDAP directory import, see Modifying LDAP directory import content.

Configuring an LDAP parameter

An LDAP parameter is an authentication parameter that exists in the LDAP directory and that is associated uniquely with a **HOPEX** attribute.

Configuring an LDAP parameter is useful when importing persons from an LDAP directory. This configuration enables initialization of attributes (of the person or login created in **HOPEX**) corresponding to parameters with values stored in the LDAP directory.

`Example: the "E-mail address" MetaAttribute of the person is initialized with the "mail" LDAP parameter of the person`

in the "Active Directory" LDAP directory (if mapping has been carried out).

To configure an LDAP parameter:

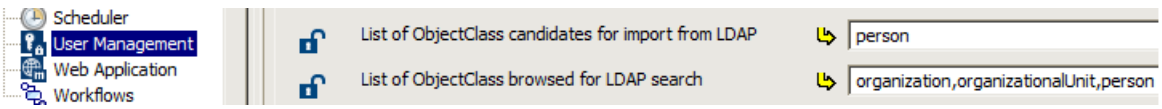
1. Access LDAP server management.
   - *See Accessing LDAP server management.*
2. Select the LDAP server for which you want to configure an LDAP parameter and click **Properties**.
3. In the **LDAP Parameters** tab, click **New**.
   - *The LDAP parameter enables pre-completion of characteristics of a person corresponding to the LDAP parameters.*
4. Enter the **Name** of the LDAP parameter (example: Mail), then click **Properties**.
5. (Optional, "expert" metamodel access) Select **Index on Persons**, so that the parameter value enables unique identification of a person. If a person in **HOPEX** has the same e-mail as a person defined in the LDAP directory, this person is reused (instead of creating a new person and risking duplicating the same person).
6. (Optional, "expert" metamodel access) Select **Is available for search** so that an e-mail can be entered in the import entry area.

   Example: if you enter ctodd@mega.com, you should find Clara TODD.
7. In the **MetaAttribute** field, click the arrow and select **Connect MetaAttribute**.
8. Execute a query on the MetaAttribute (example: E-mail).
   When importing persons from the LDAP directory, the LDAP parameter (example: mail) will initialize the MetaAttribute (example: E-mail address).

Modifying LDAP directory import content

You can modify LDAP directory import content:

- export candidate objects:
  This option enables definition of the type of objects to be imported from the LDAP directory.
  Default value: person.
- the list of objects browsed for LDAP query
  This option enables addition to the import of a particular person and/or persons of a team ("organization").
  Default value: organization,organizationalUnit,person



To define content of LDAP directory import:

1. Access the environment options management window.
   - *See Modifying options at environment level.*

2. In the options tree, expand the **Installation** folder and select **User Management**.
3. (Optional) In the right pane, modify the **List of ObjectClass candidates for import from LDAP** option.
   To import objects other than persons (default value), for example resources or org-units, specify this in this field. Objects should be separated by commas.
   Everything that is imported creates occurrences of persons with login.
4. (Optional) In the right pane, modify the **List of ObjectClass browsed for LDAP query** option.
   To add a person or organization to the import, enter the name of the person or organization (example: Quality) in the field.
   The result is the list of ObjectClass candidates for import from LDAP, that is, persons by default.

Importing persons from an LDAP server

The import of persons from an LDAP directory enables initialization of attributes (of the person or login created in **HOPEX**) corresponding to parameters with values stored in the LDAP directory.

- *See Configuring an LDAP parameter.*

Example: the "E-mail address" MetaAttribute of the person is initialized with the "mail" *LDAP parameter* of the person in the LDAP file (if mapping has been carried out).

) *An LDAP parameter is a parameter that exists in the LDAP directory and is associated uniquely with a HOPEX attribute. For example, "mail" is a parameter of the "Active Directory" LDAP directory and can be associated with the e-mail of the person.*

To import persons from an LDAP directory:

1. Open the user management window.
   - *See Opening the User Management window.*
2. In the **Persons** tab, click **Import From LDAP**.
3. The **LDAP Import Wizard** appears.
4. In the **LDAP Server** field, click the drop-down menu and select the server from which you want to import persons.
   - *The LDAP server must be created, seeCreating an LDAP server.*
5. In the **Queried Element** field, enter the queried character string.
   Example: Support service.
6. Names resulting from the query are listed.
7. Select the persons you want to import.
8. Click **OK**.

Authentication and a user created on the fly (Web Front-End)

P **Creation of a user on the fly is not available for connection to HOPEX (Windows Front-End).**

When a user has been created on the fly, the LDAP parameters can be used as indexing identifier (**Index on Person** attribute, see Configuring an LDAP

parameter) to check that a person with an attribute with the same value as the LDAP directory already exists in **HOPEX**.

Example of use:

Anne, responsible for sending questionnaires, wants to send a questionnaire. If one of the addressees does not exist:

- Anne can create the person (example: "Thomas KOCH" with e-mail "tkoch@mega.com")
- Anne cannot create the login of Thomas Koch since she is not an administrator.

When Thomas KOCH connects to **HOPEX** (Web Front-End), with tkh:

1. The authentication service authenticates tkh with the LDAP directory: the "mail" parameter exists and is indexing identifier type (**Index on Person** is selected),
2. The authentication service checks if a person already has this e-mail.
   - *Answer: Yes.*
3. The authentication service creates the login associated with the person.

If Thomas KOCH has assignments to complete the questionnaire, he can connect to the application to complete this questionnaire.

3

MANAGING REPOSITORIES

This chapter covers points relating to working in a repository and the use of repositories:

- Introduction to Repositories
- Repository Performance and Health
- Managing Repositories
- Optimizing Repository Access Performance
- Referencing and Unreferencing a Repository

INTRODUCTION TO REPOSITORIES

An environment includes:

- a system repository
- one or several **HOPEX** repositories

Repository management is carried out in the **HOPEX Administration** application (not available in the Web Administration Desktop).

The following points are covered here:

- System Repository (SystemDb)
- HOPEX Repository
- Repository Structure
- Accessing Repositories
- Creating a Repository
- Consulting and Modifying Repository Properties
- Accessing the Log of Repository Changes (.EMV file)

System Repository (SystemDb)

The system repository (SystemDb) contains the configuration required to run **HOPEX**:

- the metamodel, constituting the structure of repositories
- programmed *queries* and macros
- the elements to produce outputs: report templates, *report templates (MS Word)*, Web site templates
- users and their rights.

A system repository exists for each environment, automatically created in the **Sysdb** folder at creation of the environment.

HOPEX Repository

A **HOPEX** repository constitutes the workspace in which modeling data is stored. Several users can connect to it and work simultaneously on the same project.

A repository depends on an *environment*. Different policies of data distribution can be implemented. You can, for example, work on two repositories:

- a Development repository, which groups all projects
- a Production repository, which groups stable states of each project.

Any user of an environment can access a single repository or all of the repositories of the environment according to his/her profile assignment definition.

– *See Assigning a Profile to a Person.*

Storage type

The **HOPEX** repository is stored in the RDBMS format (Relational Database Management System): SQL Server.

– *For more details on structure of RDBMS storage, see **RDBMS Repository Installation** deployment guide.*

Repository Structure

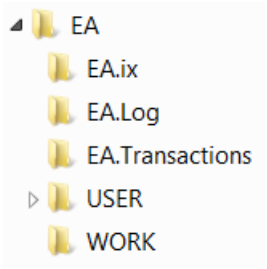
The files enabling use of a repository are all in the same folder, the location of which is stored in the *system repository* (SystDb).

) *SystDb is a particular repository containing the metamodel and technical data (descriptors, Web site templates, queries, etc.). The metamodel and technical data are common to all repositories in the same environment. Definition of users and their rights are stored in this repository, essential for operation of the software.*

A repository comprises "RepositoryName".XXX files of which XXX format depends on the information type:

- ***.EMV*** : repository evolution logfile (creation, update, etc.)
- ***.EMQ*** : pointer to data stored in the SQL Server

For each repository, folders are created dynamically while the user is working:



- ***Data***: contains all business documents created in **HOPEX**.
- ***Document***: contains all reports (MS Word) generated from **HOPEX**.
- ***<RepositoryName>.ix***: contains the result of indexing for full-text search.
  - *See Enabling/Disabling repository indexing for full-text search.*
- ***<RepositoryName>.Log*** (repository backup logfile): contains repository backup logfiles (enabled by default).
  Each private workspace is saved in logfile, .MGL format.
  - *See Backup logfile.*
- ***<RepositoryName>.Transactions*** (private workspace backup logfile: contains files linked to active private workspaces of repository users.
  - *The presence of these files depends both on the server type used and on the options enabled for this repository.*

  These files are:

  `"CCC.MGL"`

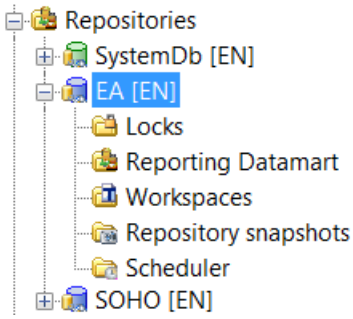
  `where CCC is the code associated with the user (the repository` **`Backup logfile`** `option is enabled).`
  - *See Backup logfile.*
- ***USER***: contains a folder for each user, in which all work files generated by the user are stored. It groups the files created by backup and extraction procedures, as well as control files.
  - *Each folder is named "CCC", where CCC is the code associated with the user.*
- ***WORK***: contains work files created by administration operations carried out using the administration application.

Accessing Repositories

To access a repository:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*

2. Expand the **Repositories** folder.
The list of the repositories in the environment appears.



Each repository is represented by:

- an icon
  - system repository
  - data repositories
- its installation language.

Creating a Repository

You can create:

- an empty repository, then you can transfer data from another repository into it by importing updates or by restoring a backup file.

  – *See **Installation and Deployment > RDBMS Repository Installation Guide**.*

- a repository already fed with data.

  – *See **Installation and Deployment > RDBMS Repository Installation Guide > HOPEX RDBMS repositories specific administration actions > Restoring a HOPEX environment from formatted data > Restoring a data repository**.*

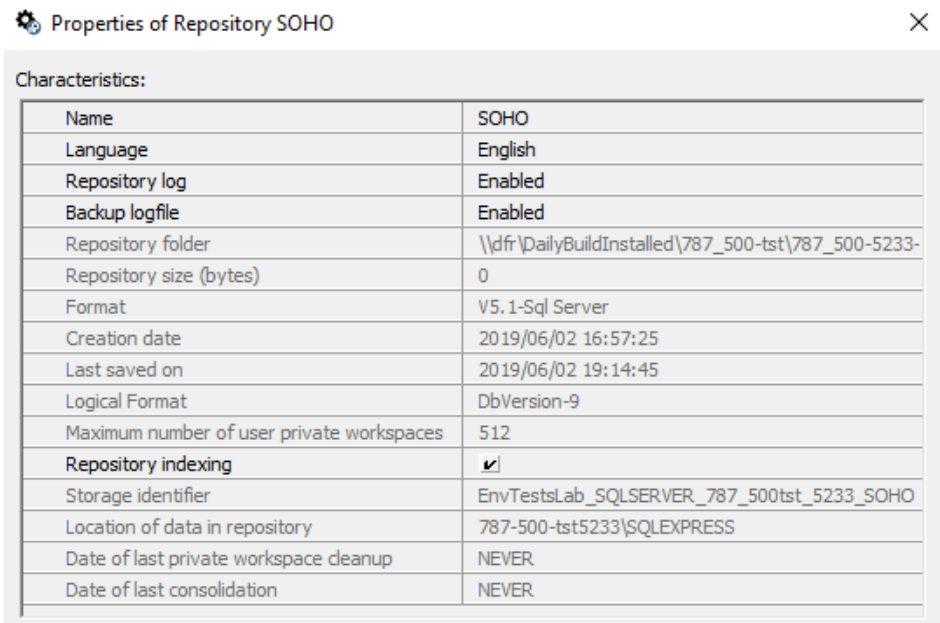
Consulting and Modifying Repository Properties

To consult and/or modify certain repository properties:

1. Connect to **HOPEX Administration** and select the repository concerned.

   – *See Accessing Repositories.*

2. Right-click the concerned repository and select **Properties**.
The **Repository Properties** dialog box opens.



From the **Repository Properties** dialog box, you can:

- click the row of one of the following characteristics to:
  - modify the **Name** of the repository
  - modify the repository **Language**

    P **Only if you need to modify the repository language right after its creation. Never change the repository language at any time, to avoid irreparable loss of object names.**

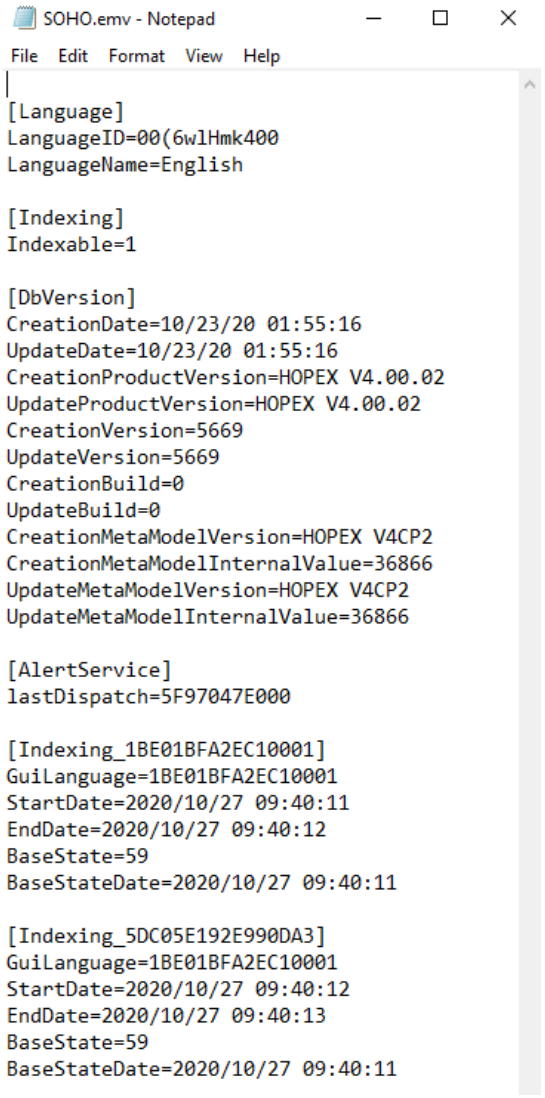
  - enable/disable the **Repository log**
    - *For more details, see Repository log.*
  - enable/disable the **Backup logfile**
- consult its following characteristics:
  - **Metamodel last compiled date** (SystemDb repository specific)
  - location of the **Folder** in which the repository is stored
  - **Dates** of repository creation and last save
  - repository **Logic Format**.
  - **Date of last private workspace cleanup** and **Date of last consolidation**, which provide useful information for the maintenance plan of an RDBMS repository.
    - *See Deleting RDBMS Repository Temporary and Historical Data.*
- enable/disable **Repository Indexing**.
  - *See Enabling and Customizing Repository Indexing.*

Accessing the Log of Repository Changes (.EMV file)

The .EMV file contains repository changes (eg: creation, update, version).

To directly access the .emv file of a repository:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **See EMV File**.



REPOSITORY PERFORMANCE AND HEALTH

See:

- Consulting RDBMS Repository Performance
- Generating a Repository Health Report

Consulting RDBMS Repository Performance

Before you start working in an RDBMS repository, **MEGA** recommends that you run the RDBMS diagnostic utility ("Measure performance of your SQL data source").

– *For more details on this utility, see the deployment guide: **Installation and Deployment > RDBMS Repository Installation Guide**.*

This utility indicates repository performance compared to optimized performance. For this purpose, you can run the utility from **HOPEX Administration**.

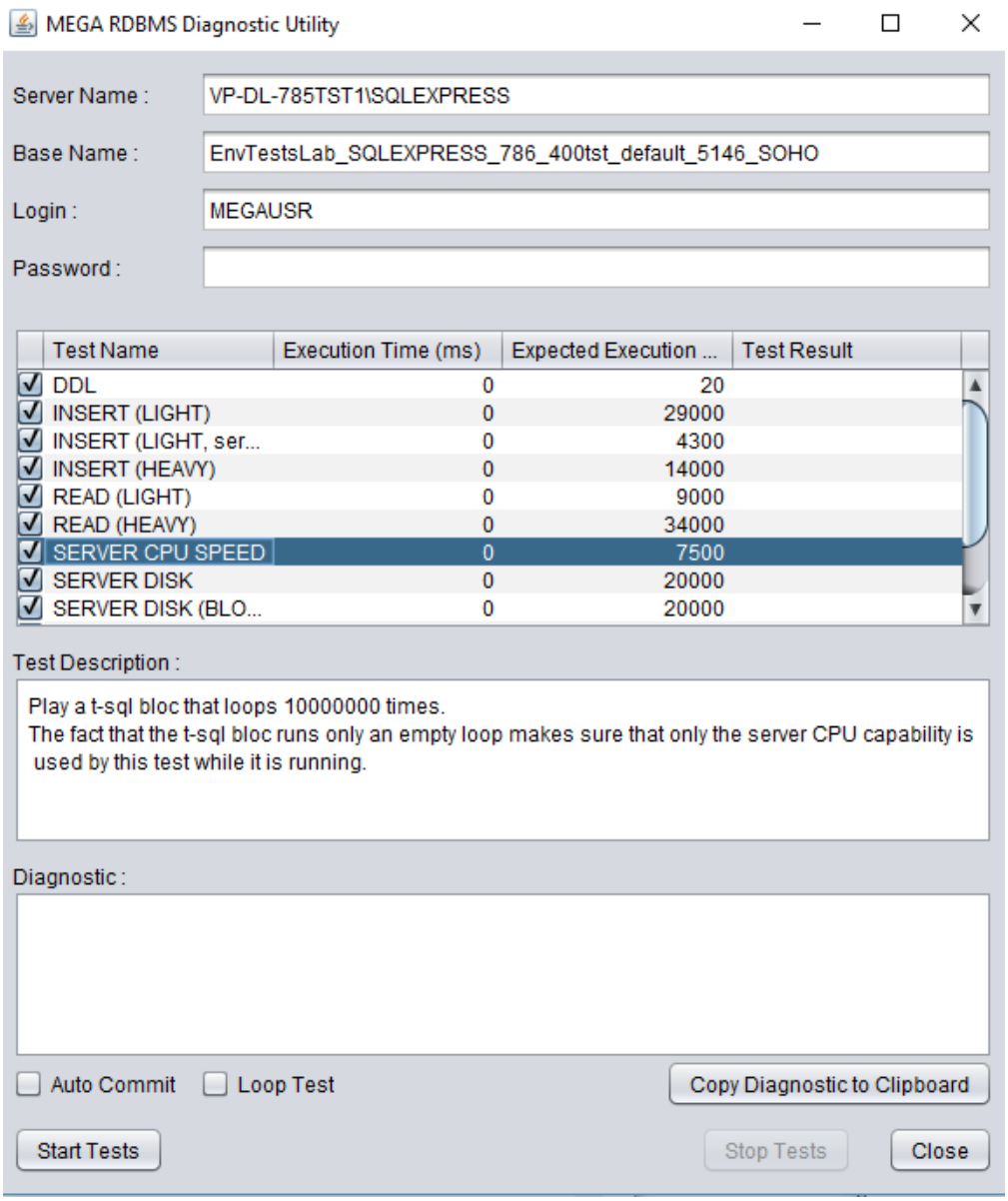
To run the RDBMS repository diagnostic utility:

1. Connect to **HOPEX Administration** and select the repository concerned.

   – *See Accessing Repositories.*

2. Right-click the RDBMS repository concerned and select **RDBMS Administration > Measure performance of your SQL data source**.

3. (If necessary) Enter the connexion parameters.



4. Click **Start Tests**.
5. Run the test twice before analyzing data.

Generating a Repository Health Report

With **HOPEX** you can generate a daily RDBMS repository health report. This report enables to detect:

- performance or usage anomalies that users can face daily.
- any significant change.

For this purpose, performance and health tests are run daily. Events are generated when anomalies are detected.

– *See Events: Infrastructure Performance and Repository Health.*

Performance test description

HOPEX standard use scenario are performed daily in the afternoon ("RepositoryHeath Daily Afternoon Trigger" job, 04:00 pm GMT):

- Reading of comments on read-only data ("Reading of").
- Read-only data loading.
- Request execution on read-only data.
- Comment writing on reading and writing data.
- Data creation.
- Data deletion.
- Search request execution on reading and writing data.

– *In a cluster-type configuration, performances are measured on all of the machines.*

Each scenario generates a result, which is stored in the repository. These results are analyzed daily in the evening ("RepositoryHeath Daily Evening Post Trigger" job, 11:05 pm GMT)

An history of 36 results are needed before generating an alert.

Health test description

It is essential to analyze certain usages to identify anything that might compromise data integrity, whether in the daily work or following a **HOPEX** update.

For all of the repositories of all of the environments, the following checks are performed every evening ("RepositoryHeath Daily Evening Trigger" job, 11:00 pm):

- Administration
  - SQL compatibility of repositories and server
  - table fragmentation
  - index fragmentation
  - SQL maintenance plan execution
- Customization
  - HOPEX data modification
  - HOPEX data volume
- Usage
  - workspace volume

– *In a cluster-type configuration, usage tests are performed randomly on a single machine only.*

Health report description

The health report includes a short description of the anomaly detected at performance or usage level.

For example, in case of an index fragmentation anomaly:

```
Sent from: <Name of the machine that ran the report formatting and emailing task>
Environment: <Environment name>
Repository: <Repository name>
Table A_BLOB
Index GBM_INDEX_A_BLOB_IDABS_BEGIN_VALIDITY
Fragmentation level 70%
```

Configuring HOPEX health report emailing

- *HOPEX health reports, automatically generated every day, are available from HOPEX Administration desktop, see Viewing the HOPEX Health Report.*

You can receive a short report of all of the anomalies detected on each repository of each environment.

The HOPEX health report is emailed daily (by default):

- You must check that the mailing configuration parameters are configured (in the site level options).
- You must define the list of recipients of the report.
- You can modify the report receiving frequency.

  E.g.: Daily, Weekly (Sunday), Daily (working days), Monthly (1st day of the month).

To configure HOPEX for emailing its health report:

1. Access the site options.
   - *See Accessing Options.*
2. In the Options tree, expand the **Installation** folder.
3. Select **Electronic Mail**.
4. In the right pane, check that the mailing configuration parameters are specified.

   Example:

   Default address of author via SMTP (FROM): admin@domain.com

   SMTP Server: mail.server.domain.com

   SMTP Port: 25

   - *See Specifying SMTP configuration.*
5. For the **Repository Health Report: recipients** option, enter the email of the recipient of the HOPEX health reports.
   - *You can enter several recipients (the separating character is the coma: ",").*

   E.g. : dan.woods@mega.com,julia.perri@mega.com

6. (If needed) For the **Repository Health Report: frequency** option, use the drop-down menu to modify the report receiving frequency.

   `E.g.: Daily (working days)`

   The recipients of the report receive an email at the frequency defined, showing a summary and including the attached detailed report.

HOPEX Health Report Summary

Infrastructure Alerts

Host: 900-002-TST5654

No abnormal performances detected on this host.

Data Alerts

Environment: EnvTestsLab_900_002_tst_5654

Repository: SOHO

Some alert have been detected on your repository:

- Data Volume Alert

Repository: SystemDb

Some alert have been detected on your repository:

- SQL Maintenance Alert

Environment: platon_900_002_5653

Repository: Platon

Some alert have been detected on your repository:

- SQL Maintenance Alert

Repository: SystemDb

Some alert have been detected on your repository:

- SQL Maintenance Alert
- Customization Alert

MANAGING REPOSITORIES

– *For management operations specific to an RDBMS repository, see deployment documentation:* ***RDBMS Repository Installation Guide****.*

The following points are covered here:

- Managing logfiles
- Configuring the logging for an inter repository consolidation
- Viewing the Repository Update Log
- Exporting Updates
- Enabling and Customizing Repository Indexing
- Converting a Repository
- Importing Libraries into a Repository
- Repository Physical Backup
- Reorganizing an RDBMS Repository
- Repository Logical Backup
- Deleting a Repository
- Updating a Repository
- Viewing the Environment Report File
- Viewing User Process Error Trace Files
- Saving the Error Zip file for Diagnostics
- Viewing Object History

Managing logfiles

At repository creation, by default:

- the backup logfile is **Enabled**
- the repository log is **Enabled**

See:

- Backup logfile
- Backup logfile process
- If you have a problem
- Repository log

Backup logfile

The repository is configured so that changes made by users are saved simultaneously in the repository and/or in a specific file called the *backup logfile*.

Backup logfile process

When opening a private workspace, the backup logfile process is as follows:

Step	User A...	Result
1	...connect to **HOPEX**.	A private workspace opens. XXX.MGL logfile is created in folder \Db\RepositoryName\RepositoryName.Private Workspaces of the environment directory.
2	...dispatches work.	XXX.MGL logfile is consolidated (*): under YYYY-MM-DD_hh.mm.ss_XXX.mgl name in "\Db\RepositoryName\RepositoryName.Log\ folder
	...discards work.	There is no copy of XXX.MGL logfile.
3	...reconnect to **HOPEX**	A new private workspace opens. XXX.MGL logfile of folder RepositoryName.Transactions is reinitialized

XXX: Code of User A

YYYY-MM-DD: dispatch date (year-month-day)

hh.mm.ss: dispatch time (hour.minute.second)

(*) : Consolidation consists of search and deletion of useless commands. For example:

```
When an object is created then deleted, the information is
not saved in the consolidated logfile.
```

```
When a comment is created, then modified several times, only
the final modification is saved in the consolidated
logfile.
```

M *To avoid overloading the* RepositoryName.Transactions directory, MEGA recommends that you regularly archive the old logfiles.

Modifications made by a user to the system repository are logged in the same way.

If you have a problem

In the event of a problem on the repository, you can restore the last repository backup and import backup logfiles saved in the RepositoryName.Log folder from this backup up to time of the problem.

- *It is highly recommended that you synchronize backup logfile archiving with repository backup.*

Dispatched modifications are logged in a new repository log.

In the event of a problem on a user private workspace, you can copy the user backup logfile saved in the RepositoryName.Log folder, delete the current private workspace and import the backup logfile in a new private workspace.

Repository log

The repository log lists all modifications made in the repository. It gives users a better understanding of actions dispatched in the repository from private workspaces.

Each time an action is executed, an occurrence of Change Item is created.

) *A **ChangeItem** is a MetaClass corresponding to a change made in a **HOPEX** repository.*

A repository log comprises **HOPEX** occurrences. These occurrences can be handled using **MEGA** APIs.

– *See Viewing the Repository Update Log.*

To keep a history of the actions performed on the repository after dispatch, the repository log must be activated.

By default the repository log is activated.

Configuring the logging for an inter repository consolidation

To improve performance you can define some MetaClasses or MetaAssociationEnds as non loggable.

– *See Modifying MetaClass loggability.*

Logging

Logging of updates enables:

- mainly to view the repository activity, i.e. actions performed on objects.

  – *See Viewing the Repository Update Log.*

  In that case, an incomplete (**Consolidate** command) or truncated (**Delete** command) log is functionally satisfactory.

  – *See Deleting a log or reducing the log size step 4.*

  This is the default configuration.
- to transfer the commands performed from a repository to another.

  Example: you want to transfer all the commands performed during the day from a development repository to a production repository.

  In that case a complete log, including all the actions performed by the users, is necessary so that the inter repository *consolidation* is performed correctly. You must modify the default log behavior.

  – *See Modifying the log behavior.*

Modifying the log behavior

To include in your log the MetaClasses and MetaAssociations defined as non loggable, you must modify the log behavior.

To modify the log behavior:

1. Access the options.
   - *See Accessing Options.*
2. In the Options tree, select **Repository**.
3. In the right pane, for the **Log behavior** option, select "Logging all updates" value.
   All of the updates are included in the logs, including MetaClasses and MetaAssociations that are defined as non loggable.
   - *See Modifying MetaClass loggability.*

Viewing the Repository Update Log

You can view the history of the actions performed in the repository after dispatch in:

- the **HOPEX Administration** application (Windows Front-End)
  - *See Viewing the repository update log.*
- **HOPEX**(**Repository Activity** navigation window)
  - *See Displaying dispatches.*
  - *See Viewing Updates.*
- the **Administration** desktop (Web Front-End)
  - *See the **HOPEX Administration - Supervisor Web** guide, section "Displaying Updates Made in the Repository".*
- the object **History**

Viewing the repository update log

To view the repository update log:

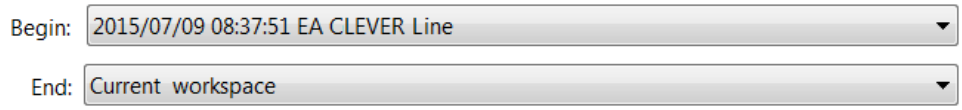
1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Repository Log > Open**.
   The **View repository updates log** appears.
   - *From **HOPEX** (with access to the **Expert** metamodel level, see Configuring the Metamodel Access), select the **File > Properties** menu. In the dialog box that appears, select the **Update** tab. The log of the current private workspace is displayed by default.*
3. Select (/Clear) **System Repository** to display the system repository (/ **HOPEX** repository) updates.
4. In the **Period** field, select the period you are interested in.

   ```
   E.g.: Today, Current week, Current month, From the beginning.
   ```

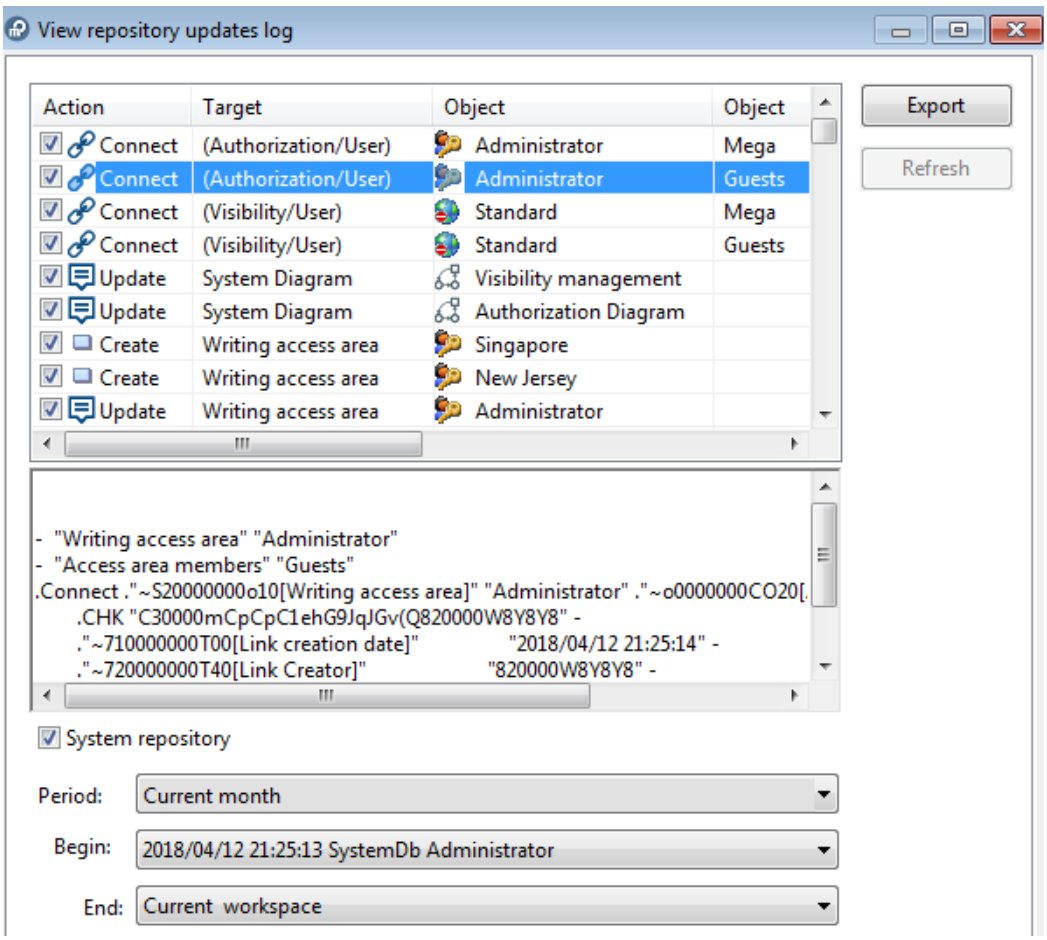
   The selected period defines the list of dispatches available in the **Begin** drop-down list.
5. In the **Begin** field, select the dispatch from which you want to display the updates.
   - *The first item corresponds to the first dispatch in the repository.*

6. In the **End** field, select the dispatch until which you want to display the updates.

   – *You can view the whole repository log, or between two given intermediate dispatches.*



7. Click **Refresh**.
   The repository log appears as a list of actions displayed in chronological order.



– *See Exporting Updates.*

Indicated for each action are:

- the **Action** type performed

  ```
  Example: "Create", "Connect", "Update".
  ```

  - *See Command File Syntax for more information on operators.*
- the type of the object concerned (**Target**)
- the name of the **Object** concerned
- the name of the second **Object** concerned in the case of a "Connect", "Disconnect" or "Change" action
- the person **Responsible** for the action
- the **Delivery date** of the action

  ```
  Format: <D/MM/YYYY h:mm:ss AM/PM>
  ```

  If you have not dispatched your work yet, the date indicated is the execution date of the action.
- The name of the **Dispatch** that contains the action, in the following format:

  ```
  <YYYY/MM/DD hh:mm:ss> <repository> <responsible>
  ```

  With:
  - <YYYY/MM/DD hh:mm:ss>: workspace date and creation time
  - <Repository>: name of the current repository
  - <responsible>: name of the person or of the collaborative workspace responsible for the action

  If you have not dispatched your work yet:

  ```
  <your name> (workspace)
  ```

  ```
  <name of your collaborative workspace> (collaborative workspace)
  ```
- (when you select the line of a command), the complete text of the update is displayed in the window lower pane.

  - *If needed, to copy the command and paste it in a text file for example, roll the mouse over the whole text, then press <CTRL> + <C> and paste the text in a text file.*

Displaying dispatches

To display in **HOPEX** private workspaces dispatched and the content of their updates:

- *To access the content of the **Repository Management** pane, you must have **Expert** metamodel access (see Configuring the Metamodel Access).*

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In the **Repository Activity** Navigation window, expand the **Repository Dispatch** sub-folder.
   All dispatches performed on the current repository and the system repository are contained in the respective **<Repository Name>** and **SystemDb** folders.
3. Expand the repository folder concerned.
   Dispatches are filed by day, week and month.

4. Expand the dispatch concerned.
   Each dispatch details the updates made to an object.
5. Expand the update concerned.
   In the properties of the object concerned, the **General > History** tab details the content of the dispatch in the form of a list of actions performed on the object concerned.

Exporting Updates

You can export updates between consecutive dispatches or not.

- *To export the repository log, see Backing up the repository log.*

To export updates:

1. Access the repository update log.
   - *See Viewing the Repository Update Log.*
2. Select the updates to be exported.
3. Click **Export**.
4. Select the export format:
   - *.mgl: text format
   - *.xmg: MEGA XML format.
5. Click **Export**.
   The file is exported and saved in the specified folder.
   The **Execution Report** appears.
6. (Optional) Click **Open result file** to view the file.
7. Click **OK** to close the window.

Enabling and Customizing Repository Indexing

To allow the user to perform full-text searches in **HOPEX** solutions, the repository must first be indexed.

Indexing runs automatically every 10 minutes (modifiable default value) with the indexing scheduler. This indexing is incremental and concerns modified objects only.

P **Some of HOPEX metamodel modifications (for example setting a MetaClass to indexable) require an entire indexing, see Deleting indexing folders).**

- *To modify the scheduler default configuration, see Customizing the indexing scheduler.*

- *The initial indexing can take some time depending on the size of your repository (eg: more than 30 hours for a 2 GB repository, of which business documents constitute a major part), and can slow the performance of **HOPEX**. Remember to run this initial indexing when other **HOPEX** users are not connected.*

P **Take care to allow sufficient disk space before enabling indexing: statistically for a large repository (eg: 2 GB) of which business documents constitute a major part, the indexing size can represent twice the repository size.**

Enabling/Disabling repository indexing for full-text search

By default, repository indexing is not enabled.

To enable/disable repository indexing:

1. Access properties of the repository concerned.
   - *See Consulting and Modifying Repository Properties.*
2. Select/Clear **Repository Indexing** to enable/disable repository indexing.
3. Click **OK**.
   The scheduler updates indexing every 10 minutes.
   - *To modify scheduler default configuration, see Customizing the indexing scheduler.*

   Indexing is carried out for all of the languages installed.

   When indexing is completed, the <Repository Name>.IX folder is created in the corresponding folder of the repository. This folder contains indexing results.

Indexing a repository manually

For initial indexing, the administrator can index the repository manually.

To index a repository manually:

- *To be able to manually index the repository, the repository indexing option must be selected, see Enabling/Disabling repository indexing for full-text search.*

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository to be indexed and select **Index for full-text search**.
   Indexing is carried out for all of the languages installed.
   - *For initial indexing, the "Repository Name".IX folder is created in the corresponding folder of the repository. This folder contains indexing results.*

Customizing the indexing scheduler

You can modify indexing scheduler default configuration.

To customize the indexing scheduler default configuration:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click **Scheduler** and select **Manage Triggers**.
3. Select the **System Triggers** tab.
4. Right-click **Indexing Automaton** and select **Update Scheduling**.
   The indexing scheduler configuration window appears.
5. Modify the configuration.
   - *For information on the scheduler, see Managing the Scheduling (Scheduler).*
6. Click **OK**.

Deleting indexing folders

Some of **HOPEX** metamodel modifications may have indexing impacts. This requires to delete old indexing folders for each repository (including SystemDb repository) so that the entire indexing is performed.

Example: when a MetaClass becomes indexable (in the MetaClass properties, its **Candidate to indexation** attribute is set to "Yes"), the indexing automatic update does not take into account the objects corresponding to this MetaClass.

Indexing folders are stored in each repository, for each language with the following format:

<Repository name>_<Language hexa IdAbs>.ix

P **Warning: as soon as you delete an indexing folder, at next indexing schedule, an entire indexing is performed. Remember to run this indexing when HOPEX users are not connected.**

To delete indexing folders:

1. In the **HOPEX** installation folder, access the repository folder.

   <HOPEX installation path>\<Environment name>\Db\<repository name>

   <HOPEX installation path>\<Environment name>\SysDb

2. In each repository folder (including SystemDb) delete all of the indexing folders:

   <Repository name>_<Language hexa IdAbs>.ix

   The entire repository indexing will be performed at next indexing schedule.

Converting a Repository

Repository conversion is only necessary:

- at migration.

  - *For more information on repository conversion, consult the technical note **How to migrate to HOPEX**.*

- on request from **HOPEX** support.

With **HOPEX** you can mass convert technical data. This possibility enables you to mass launch the procedure on all the repositories with no need to validate the execution for each repository.

Mass converting technical data

To mass convert technical data:

1. Connect to **HOPEX Administration** and select the environment concerned.

   - *See Connecting to an Environment.*

2. Expand the environment folder.

3. Right-click the **Repositories** folder and select **Manage**.
4. In the **Action list** pane, select **Perform SQL conversion on the repository**.
5. In the **Repository list** pane, select the repositories for which you want to convert technical data.
6. Click **Execute**.
   The procedure is automatically launched on each repository selected with no need to validate each execution.

   - *A dialog box enables you to stop the procedure at any time. If needed, click **Cancel** to stop the procedure execution.*

Importing Libraries into a Repository

You can import the libraries you need for your work into a repository.

To import a library into a repository:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Object Management > Import**.
   The **Import HOPEX Data** dialog box opens.
3. Alongside the **Command File** field, click **Browse** ... .
4. From the **HOPEX** installation folder, select in the **MEGA_Std** folder the desired Library (*.mol).
5. Click **Open**.
6. In the **Import HOPEX data** window, click **Import**.
7. When import is completed, click **Close**.

Repository Physical Backup

In event of a problem, you must have a valid and recoverable data backup.

A physical backup consists of copying the files of a repository from their original location to another one.

Data is stored at **HOPEX** environment level. Data to be backed up varies according to repository server type.

- *See Creating a Repository.*

The following points are covered here:

- Backup frequencies
- Elements to be backed up
- Other elements to be backed up
- Elements that could be useful to back up

Backup frequencies

For a **HOPEX** environment used by an active project, **HOPEX** recommends:

- daily backup of the environment
- backup before any major data update

  `Example: system database customization, data reprocessing, CP/RP update of HOPEX data.`
- that you keep:
  - daily backups of the last 30 days
  - monthly backups of the last 12 months

Whatever your repository server type, **MEGA** recommends cold backup (no **MEGA** user should be connected).

- *In SQL server type mode, hot backup is possible.*

Elements to be backed up

Identify environments that require regular backup.

`Example: design environment.`

From the environment folder, you must back up the complete folders:

- **Db**
- **SysDb**
- **Mega_usr**

Db and **SysDb** folders contain an .EMV file and an .EMQ file that points to other folders that you must back up: system and repository databases of SQL server databases.

Elements that can be excluded from the backup file

To save space and time it is not necessary to backup the complete content of the environment folder. You need not back up for example folders that contain:

- user work files
  These files are contained in the **SysDb\USER** and **Db\USER** folders.
- work files linked to the Administration application
  These files are contained in the **SysDb\WORK** and **Db\WORK** folders.
- lock files:
  - **Systemdb.Lock** contained in **SysDb** folder
  - **RepositoryName.Lock** contained in **Db\<RepositoryName>** folder

Other elements to be backed up

MEGA recommends that you back up folders concerning:

- configuration:
  the **Cfg** folder (in the **HOPEX** installation directory) containing the **megasite.ini** configuration file.
- licenses:
  the file containing licenses (.Must or .ELF).

(Optional) You can back up folders concerning:

- your java customization:
  lib_usr folder (in the **java** folder of the **HOPEX** installation directory)
- your installation customization:
  Mega_Usr folder
- your delivered data
  in the **HOPEX** installation directory, the following folders:
  - **Document**, which contains shared documents
  - **Intranet**, which contains the Web sites generated
  - **Approve**, which contains detached documents.

Elements that could be useful to back up

You may need to back up:

- private workspaces in progress
- technical data modifications in progress

To back up private workspaces in progress:

› copy the "RepositoryName.Transactions" folder in the repository folders tree.

– *The **USER** and **WORK** folders contain the work documents of users.*

To back up technical data modifications in progress:

› copy the **SysDb** folder and its sub-folders in the tree of the *system repository*.

Reorganizing an RDBMS Repository

You can reorganize a repository when you want to change the storage type.

`E.g.: from an Oracle repository to an SQL Server repository.`

Before reorganizing a repository, you must check that there is no other active or passive private workspace on this repository, see Workspace Administration.

The reorganization process of an RDBMS repository is automated and consists in:

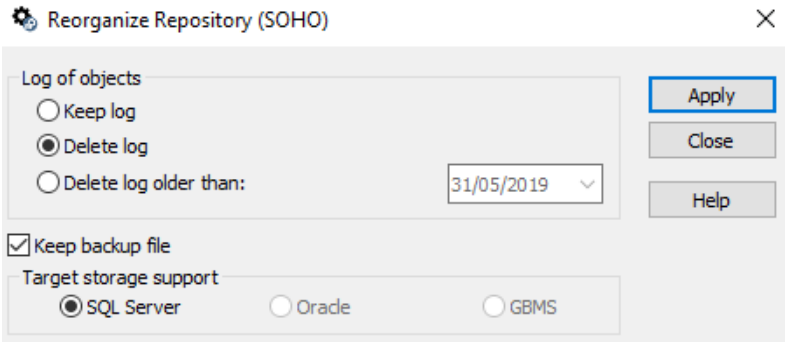
1. A logical backup of the repository, to get the *command file* which contains creation orders of repository objects and their links.
2. A repository initialization.
3. A repository restore by import of the command file in an empty repository.

Reorganizing a repository

M *To improve reorganization times and **HOPEX** performance, remember to delete old log elements before repository reorganization.*

To reorganize a repository:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Reorganize**.



3. (Optional) In the **Log of objects** frame, you can select the actions to be executed on the log at reorganization.
4. (Optional) If you do not want to keep the repository log, clear **Keep backup file**.
   - *The log is generated in the work folder WORK of the repository, in format Bkp_<YYYY-MM-dd_HH.mm.ss>_<repository name>.mgr*
   - *By default, the target server type is the same as the repository server type.*
5. Click **Apply**.
   When the repository has been reorganized, a message indicates that the reorganization is completed.

Repository Logical Backup

Before you make a repository backup, dispatch all current private workspaces so that their changes are included in the backup. To view private workspaces active on your repository, use the private workspaces administration tool, see Workspace Administration.

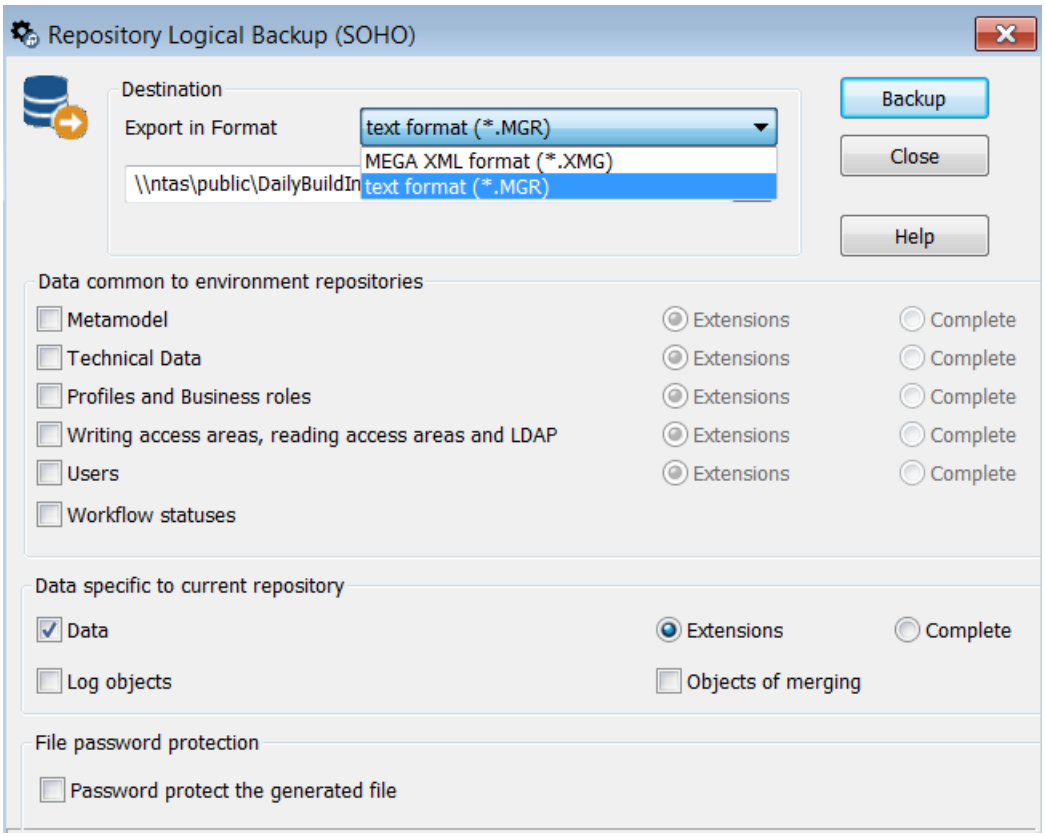
Logical *backup*:

- creates a *command file* that allows you to reconstruct the repository by update of an empty repository.
- analyzes all repository content.
- is safer than a physical backup since it checks that all data in the repository is readable.
- can be used as a long term archive or to merge repositories.

> P **You can execute updates on the repository after starting logical backup by dispatching a private workspace. However, note that these modifications are not included in the backup.**
>
> M *You can temporarily prohibit users from updating the repository by clearing the **Authorize Dispatch for the environment** option or **Authorize user dispatch** option in the environment options.*

To perform a logical backup of a repository:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Logical Backup**. The **Repository Logical Backup** dialog box opens.



3. In the **Destination** pane, select the export format of the backup file:
   - **text format** (*.MGR).

     M *For more details on .MGR file syntax, see Command File Syntax.*

   - **MEGA XML format** (*.XMG)
     This format is reserved for exchange of data between **HOPEX** and other applications. It includes commands or data (objects and links). This format cannot be used to extract the metamodel or technical data.

     M *For more details on MEGA XML data exchange format, see technical article **MEGA Data Exchange XML Format EN**.*

     - *Environment options enabling configuration of **HOPEX** data export (XMG encoding, default export format, etc.). See Managing Options.*

4. (Optional) In the **Destination** pane, click **Browse** ... to browse the folder tree and modify the name and/or location of the backup file.
   By default, the backup creates a file "RepositoryName.mgr" in the WORK *work folder*.

   - *We recommend that this backup be made to a physical device other than the device on which the repository is located. Selection of a different logical device, such as a different partition on the same disk, does not protect your backup in the event of disk failure.*

5. Select the type of data you want to save.

   - *The **Extensions** buttons correspond to data created by users.*

   P **Carry out a complete backup only if technical support asks you to do so.**

   M *You can choose what you want to save: extensions common to environment repositories and data specific to current repository. It is recommended that you save all these elements in separate specific files, with names indicating their contents.*

   - *For a standard logical backup, simply select the **Data** and its **Extensions**.*

   In the **Data common to environment repositories** pane, select the data type of the system repository to be saved:
   - **Metamodel** allows extraction of the *metamodel*, if the standard metamodel has been modified.
   - **Technical Data** allows extraction of data such as *descriptors*, *queries*, and *report templates (MS Word)*.

     P **A complete logical backup of the repository including the Technical Data can take time and occupies considerable space.**

   - **Profiles and Business roles** allows extraction of created profiles and business roles (those not provided by MEGA).
   - **Writing access areas, reading access areas and LDAP** allows extraction of created writing and reading access areas (those not

provided by MEGA) and LDAP parameters (parameters, servers, groups).
- **Users** allows extraction of created users (persons, person groups, logins).
- **Workflow statuses** enables extraction of workflows (workflow instances, transitions and statuses, tasks, validations, requests for change).

In the **Data specific to current repository** pane:
- **Data** allows extraction of repository data. This data includes assignment of business roles to persons.
- **Log objects** allows you to include object histories in the extraction of **HOPEX** data.
  - *For more information on object logs, see Viewing Object History.*
- **Objects of merging** allows you to export technical objects resulting from merging objects (_TransferredObject).
  - *For further information on merging objects, see Merging Two Objects.*

6. (If needed) In the **File password protection** pane, select **Password protect the generated file**.
   - *The password is asked at file import.*
7. Click **Backup** to start backup.
   During execution of backup, a series of messages keeps you informed of progress.
   The backup report is displayed in the **Report** area.
   - *For more information on this file, see Viewing the Environment Report File.*

To update a repository, import the repository backup file.
- *For more details on importing a command file, see Updating a Repository.*

Deleting a Repository

To delete an SQL Server repository, you need the appropriate administration rights: at least db_creator.

To delete a repository:
1. Connect to **HOPEX Administration** and unreference the repository concerned.
   - *See Unreferencing a Repository.*
2. Ask the database administrator to delete the database concerned from the SQL server.

Updating a Repository

Importing command files

You can update a repository by importing the command file produced by the repository backup tool, export of an object or any other means of command file production.

You can import two types of command file into a **HOPEX** repository:

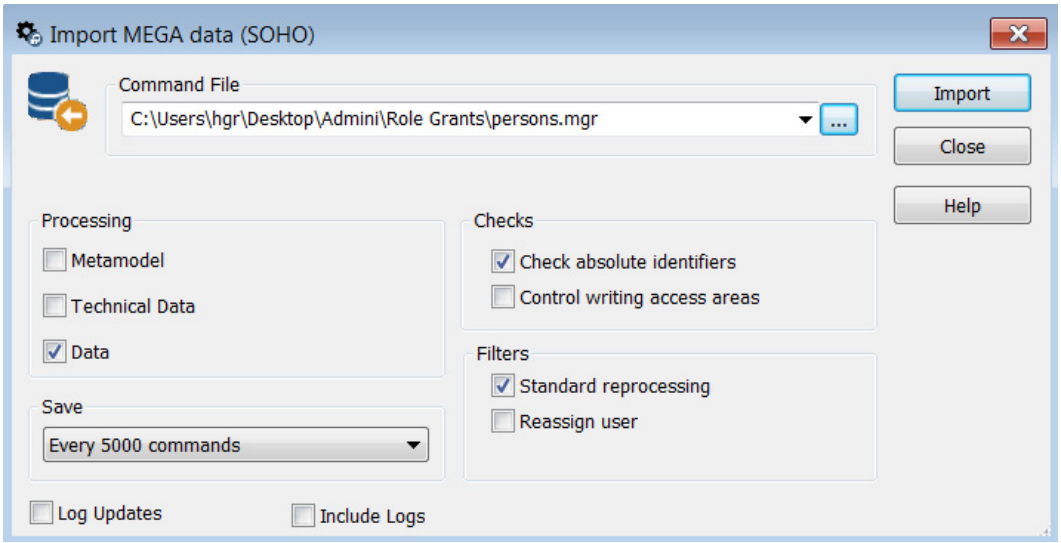
- **text format** (.MGR).
  - *For more details on .MG* file syntax, see Command File Syntax.*
- **MEGA XML format** files. These files have .XMG extension and contain commands or data (objects and links).
  - *For more details on MEGA XML data exchange format, see technical article **MEGA Data Exchange XML Format EN**.*

To import a command file:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Object Management > Import**.
   The **Import HOPEX Data** dialog box opens.
   - *To import a command file from **HOPEX**, select **File > Import > HOPEX File**.*
3. In the **Command File** pane, click **Browse** ... to browse the folders and select the backup file.

4. Select the types of **Processing** to be executed:
   You can update:
   - the **Metamodel** (repository structure)
   - the **Data** (most frequent case)
   - the **Technical Data** (*descriptions*, *requests*, as well as *users*).

     – *If the file includes commands that do not match the type you have selected, these commands are ignored.*



5. Select the **Save** frequency of the modifications.

   – *Note that there is no optimal save frequency:*

   - ***Standard*** *frequency saves at each "Validate" command in the command file and at the end of the file. This type of frequency is useful when the command file has been written by a user.*
   - ***At end*** *is generally sufficient if the file is not very large.*
   - ***At end if no reject encountered*** *saves the changes only if no rejects were encountered.*
   - ***Never*** *is used to carry out tests before the effective update, for example for syntax checking.*
   - ***Every 5000 commands****: each save is quite long. You can speed things up or slow them down by saving* ***every 100, 200, 500, 1000*** *or* ***5000 commands****.*

   – *Large files may cause memory problems when updating. To avoid such problems, you should decrease the intervals between saves.*

6. In the **Checks** pane, the checks to be carried out are selected automatically, based on the file extension:
   - **Check Absolute Identifiers** is not selected in the case of a command file that does not come from a **HOPEX** repository.
   - **Control writing access areas** is selected when the **HOPEX Power Supervisor** technical module is available on the site, ensuring that the user who executed the update has the corresponding writing access in the repository.

     – *For command files with the MGR extension (repository backup), absolute identifiers are included in the imported objects and writing access levels are maintained.*

     – *For command files with the MGL extension (log extraction or backup logfile), the absolute identifiers are included in the imported*

objects. The writing access levels are maintained if the updates are consistent with the writing access diagram for the environment.

– *These controls are not carried out if the user level is "Administrator", this enables the data restorations.*

7. In the **Filters** pane, select the import behavior to be applied:
   - **Standard Reprocessing** changes creation of an already existing object into a modification, or into creation of an object of the same name preceded by a number if their absolute identifiers are different.
   - **Reassign User** ignores the writing accesses contained in the imported file. All elements in the imported file are given the same writing access level as the user executing the import. This is useful when you have the **HOPEX Power Supervisor** technical module. The creator and modifier names are replaced with the name of the user executing the import.

     – *It is recommended that you enable this option when the import file comes from an environment where the writing access diagram is not the same as the one for the environment where this file is being imported.*
8. (Optional: **Windows Front-End**) Select the **Log Updates** option if you want to update the *repository log*, if this log will be exported to another workstation without the file being imported.

   P **This option is an advanced operation, MEGA recommends that you contact MEGA support before selecting this option.**
9. Selecting **Include Object Logs** allows you to also import object histories.

   – *For more information on object logs, see Viewing Object History.*

   The options you select are controlled by the software, based on the file extension and the standard processing to be applied. If your choices are not consistent with the file extension, a message box informs you of this fact and its possible consequences.

   – *For more details on the main causes of rejects, see Dispatch Conflicts and Rejects When Dispatching.*
10. Click **Import**.
    The report window appears showing the import progress.
    The **Processing** pane details the number of commands accepted and rejected.

When the import contains errors:

- a reject report file is generated.

  – *See Viewing rejects.*
- an execution report file is available.

  – *See Viewing the import execution report file.*

Viewing rejects

To view the rejects (or errors) recorded during the import of the command file:

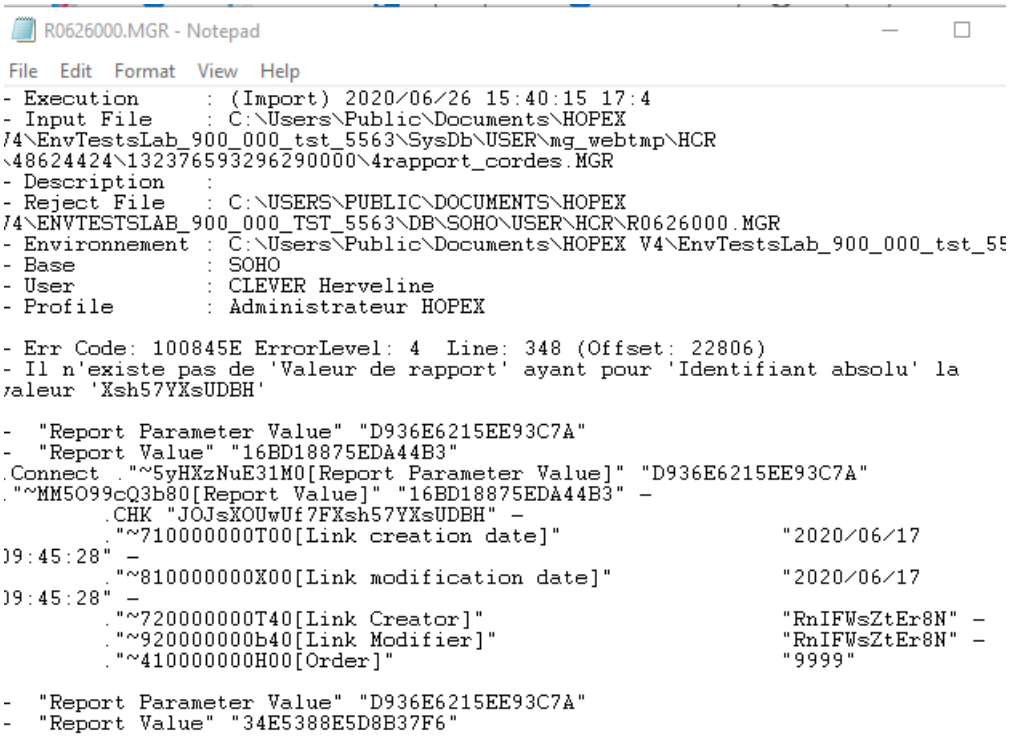
1. Import the command file

   – *See Importing command files.*
2. Click the **Report File** button.

   – *The contents of the report file depend on import options. For more details on importing a command file, see Managing Options.*

Case of a text file import (MGR, MGL)

The report file appears and details all the rejects.



R0626000.MGR - Notepad

File Edit Format View Help

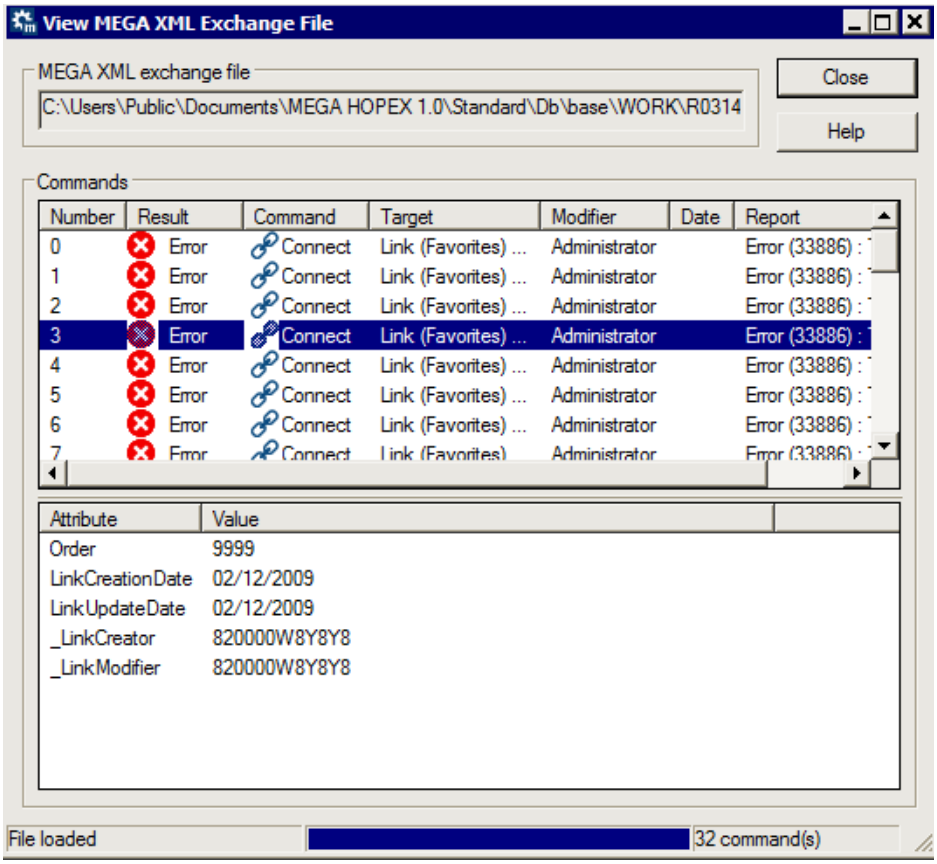
```
- Execution      : (Import) 2020/06/26 15:40:15 17:4
- Input File     : C:\Users\Public\Documents\HOPEX
/4\EnvTestsLab_900_000_tst_5563\SysDb\USER\mg_webtmp\HCR
\48624424\132376593296290000\4rapport_cordes.MGR
- Description    :
- Reject File    : C:\USERS\PUBLIC\DOCUMENTS\HOPEX
/4\ENVTESTSLAB_900_000_TST_5563\DB\SOHO\USER\HCR\R0626000.MGR
- Environnement  : C:\Users\Public\Documents\HOPEX V4\EnvTestsLab_900_000_tst_55
- Base           : SOHO
- User           : CLEVER Herveline
- Profile        : Administrateur HOPEX

- Err Code: 100845E ErrorLevel: 4  Line: 348 (Offset: 22806)
- Il n'existe pas de 'Valeur de rapport' ayant pour 'Identifiant absolu' la
/aleur 'Xsh57YXsUDBH'

-  "Report Parameter Value" "D936E6215EE93C7A"
-  "Report Value" "16BD18875EDA44B3"
.Connect  ."~5yHXzNuE31M0[Report Parameter Value]" "D936E6215EE93C7A"
."~MM5O99cQ3b80[Report Value]" "16BD18875EDA44B3" -
        .CHK "JOJsXOUwUf7FXsh57YXsUDBH" -
        ."~710000000T00[Link creation date]"                "2020/06/17
)9:45:28" -
        ."~810000000X00[Link modification date]"            "2020/06/17
)9:45:28" -
        ."~720000000T40[Link Creator]"                      "RnIFWsZtEr8N" -
        ."~920000000b40[Link Modifier]"                     "RnIFWsZtEr8N" -
        ."~410000000H00[Order]"                             "9999"

-  "Report Parameter Value" "D936E6215EE93C7A"
-  "Report Value" "34E5388E5D8B37F6"
```

Example of reject file at MGR file import

Case of a MEGA XML import (Windows Front-End)

The view window for the report file appears and details the **Commands** contained in the imported file and the **Result** of execution of each command:

- ***Accepted***: the command is accepted
- ***Error***: the command is rejected
- ***Warning***: the command is accepted, but contains anomalies



Viewing the import execution report file

To view the import execution report:

1. Import the command file
   - *See Importing command files.*
2. Click **Report**.
   The report details the number of commands analyzed, executed and rejected for each command type.

Viewing the Environment Report File

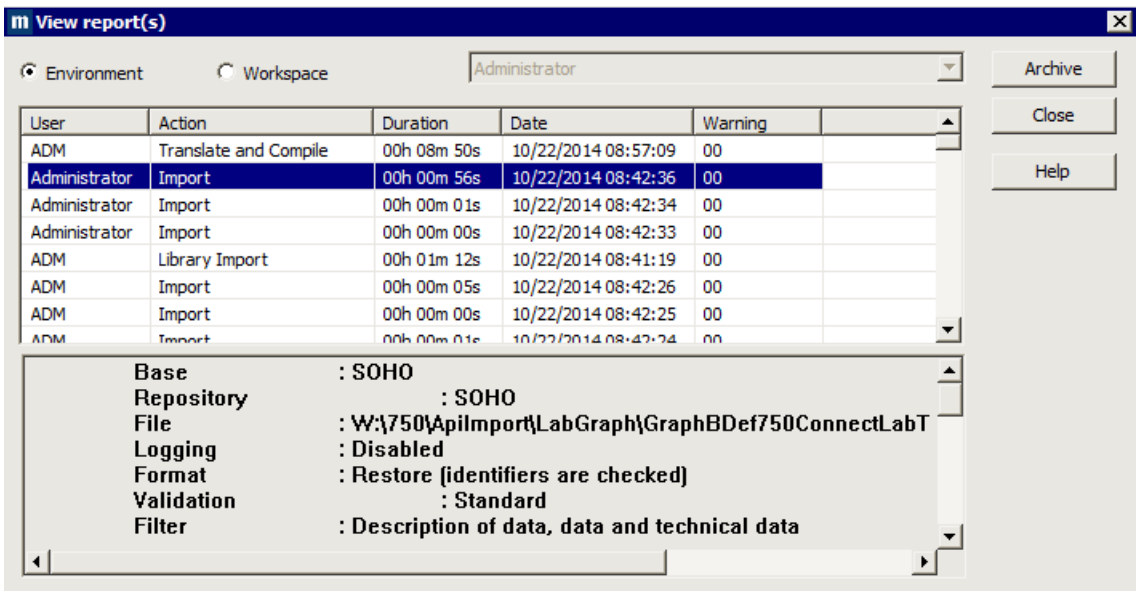
) *The environment report file, MegaCrdYYYYmm.txt (where YYYY and mm represent year and month of creation) indicates all administration operations (backup, export, restore, controls, etc.) carried out in the environment. The report file is stored in the user work folder associated with the environment system repository: SYSDB\USER\XXX\XXX.WRI where XXX is the user code.*

After you have executed backup and restore operations on a repository, you can view the environment *report file*.

Viewing the environment report file

To open the environment report view window:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Right-click the environment and select **Reports > Open**.



This dialog box also allows you to view report files of each user of the environment. Indicated for each action are:

- **User** who executed the operation.
- **Action** executed: dispatch, import, installation, translation, export, derivation, extraction, backup, update, user diagram extraction, user diagram import, repository translation, repository or environment check,

repository creation, repository deletion, log initialization, object protection, etc.
- **Date** the operation began.
- **Duration** of the operation.
- return code in the **Warning** column:
  - Error level 0: no error.
  - Error level 2: minor errors such as attempts at creating already existing objects or links.
  - Error level 4: the actions involved nonexistent objects or links.
  - Error level 8: a system error was encountered during the update.
  - Error level 16: update aborted because of a problem such as insufficient disk space.

when you select an action, the lower frame displays:
- the repository concerned
- the file used
- details of the action.

Copying the environment report file

To copy and reinitialize the environment report file:

1. In the **View Report(s)** window (see Viewing the Environment Report File), click **Archive**.
   A confirmation request message appears.
2. Select **Yes**.

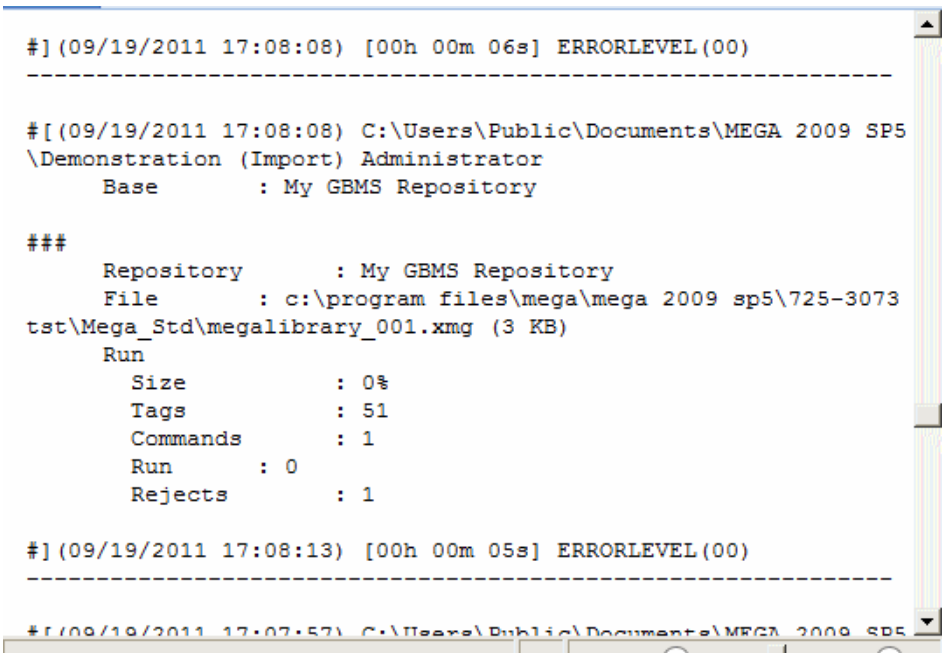
   – *You should reinitialize this file (or archive with the network version) from time to time.*

Opening the environment report file

To open the "MegaCrdYYYYmm.txt" file with Wordpad:

1. From the explorer, localize the "MegaCrdYYYYmm.txt"" file in the environment folder.

2. Right-click the file and select **Open with > Wordpad**.



Viewing User Process Error Trace Files

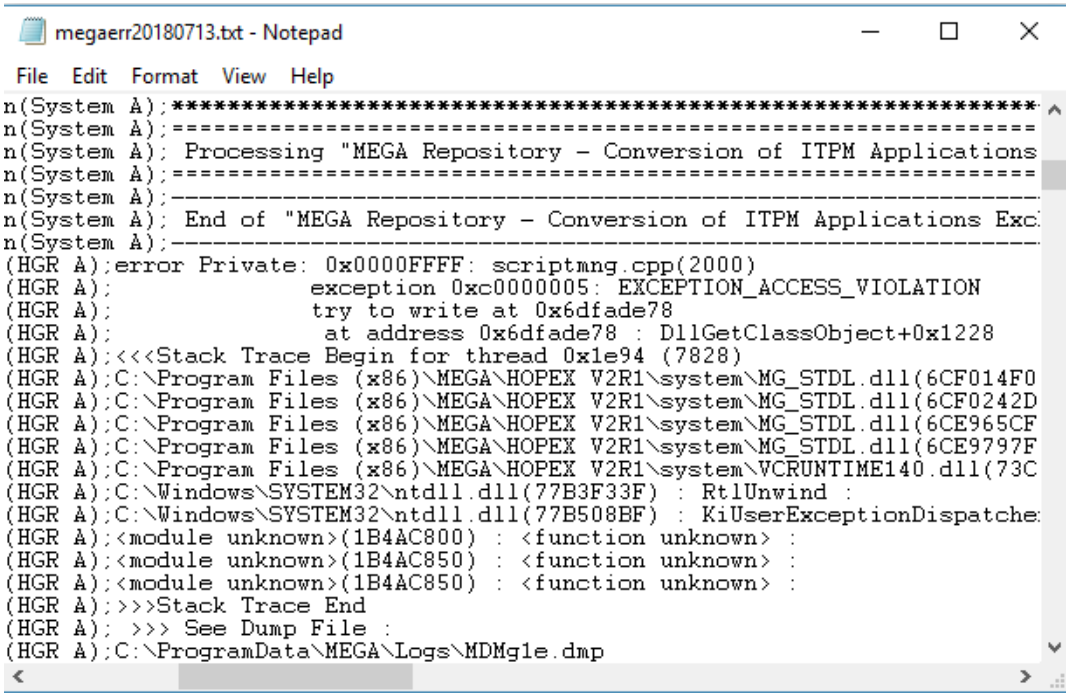
Trace files of errors in user processes contain information on operations executed and possible anomalies.

If there is a problem with a repository, this file will help the **MEGA** Research Center to analyze it.

Each *trace file* has the following characteristics:

- format: *.txt
- name: megaerrYYYYmmDD
  where YYYYmmDD represent the year, month and day when the file was generated

  Example: file Megaerr20110204.txt was generated on 04 February 2014).
- information on errors:
  - date and time of the error
  - action that produced the error
  - associated error message



You can open the trace file from:

- **HOPEX Administration**
- **HOPEX Server Supervisor**
- **HOPEX**

Opening the trace file from HOPEX Administration

To open the trace file from **HOPEX Administration**:

1. Connect to **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. In the navigation tree, right-click **Workstation** and select **Trace File > Open**.
   The file is opened in the main pane of the window.

Opening the trace file from the HOPEX Server Supervisor tool

To open the trace file from the **HOPEX Server Supervisor** tool:

1. (Prerequisite) The **HOPEX Server Supervisor** tool is started, see Starting HOPEX Server Supervisor.
2. In your workstation system tray, right-click **HOPEX Server Supervisor** and select **Mega Logs > Open Daily Logs**.

Opening the trace file from HOPEX

To open the trace file from **HOPEX**:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In the **HOPEX** menu bar, select **Help > About HOPEX**.
   The **About HOPEX** dialog box appears.
3. In the dialog box that opens, click **System Information**.
4. In the **System Information** dialog box, select **Error Log > Edit**.

Saving the Error Zip file for Diagnostics

The **HOPEX Server Supervisor** tool allows you to save a zip file containing information required by the **MEGA** Research Center to help in repository problem diagnostics.

This zip file of errors contains in particular the trace files of errors related to user processes and/or SSP (megaerrYYYYmmDD.txt and ssperrYYYYmmDD.txt).

Prerequisite: **HOPEX Server Supervisor** is started, see Starting HOPEX Server Supervisor.

To save the error zip file for diagnostics:

1. In your workstation system tray, right-click **HOPEX Server Supervisor** and select **Logs > Daily Logs manager**.
2. Click **Zip**.
3. Specify a saving location and enter the zip file name.
4. Click **Save**.

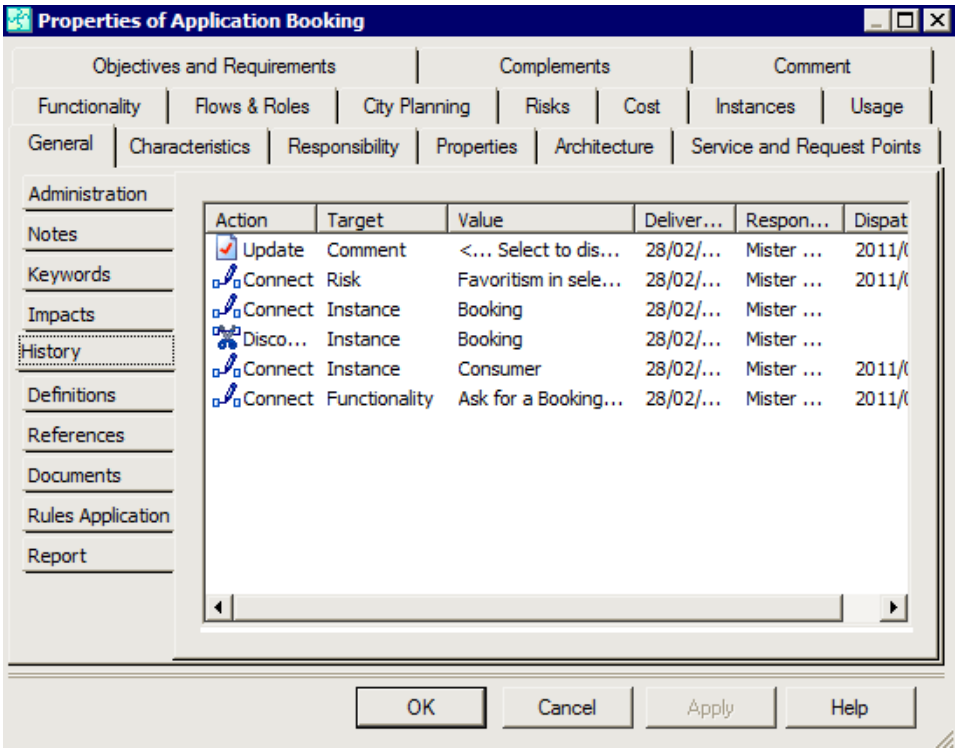
Viewing Object History

The object history is a different view of the repository log: instead of browsing actions performed by a user in a dispatch, the object history shows actions carried out from an object for all dispatches and users.

- *The repository log must be enabled so that the object history can be supplied, see Repository log.*

In the properties dialog box of an object, the **History** subtab of the **General** tab gives an overview of actions on each object in the repository. At each update

concerning the object (Create, Modify, Connect, Disconnect), the corresponding action is added to the list.



OPTIMIZING REPOSITORY ACCESS PERFORMANCE

To optimize **HOPEX** performance, you must also remember to optimize your repository size. You must:

- reduce the log size
  - *See Managing Log Size.*
- increase the cache size
  - *See Increasing RDBMS cache size (memory).*
- delete temporary data and history data (RDBMS repository) regularly
  - *See Deleting RDBMS Repository Temporary and Historical Data.*
- perform regular maintenance tasks of RDBMS repositories
  - *See Performing Repository Regular Maintenance Tasks.*
- reduce quantity of status indicators
  - *See Managing Status Indicators.*
- clean up repository
  - *See Cleaning up a Repository.*
- configure anti-virus actions
  - *See Configuring the Anti-Virus According to HOPEX Data.*
- reorganize repository
  - *See Reorganizing an RDBMS Repository.*

Managing Log Size

- *Managing the log size is only necessary if you have enabled the repository log, see Repository log.*

To reduce the log size, you can:

- delete or consolidate all the log commands earlier than a selected date
  - *See Deleting a log or reducing the log size.*
- (SQL Server) select and delete log elements, earlier than a selected date
  - *See Deleting log elements to reduce the log size.*
- modify MetaClass loggability
  - *See Modifying MetaClass loggability.*

Log size management frequency

You must reduce the log size:

- every month, for configurations of less than 50 users.
- every week, for configurations of more than 50 users.

Deleting a log or reducing the log size

Prerequisite: before deleting your log (complete or partial deletion), MEGA recommends that you back up it.

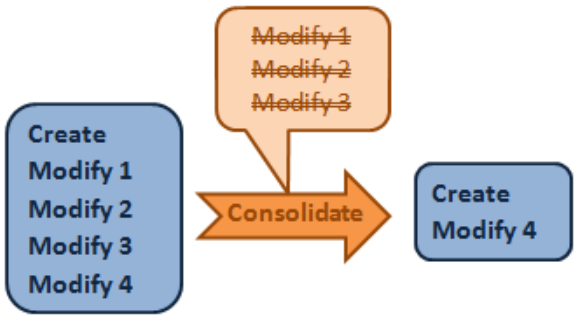
- *see Backing up the repository log.*

To delete a log or to reduce its size:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Repository Log > Manage Repository and Object Log**.
   The **Manage Log** dialog box opens.

3. In the **Manage log** window, define the commands to be deleted. Select either:
   - **All commands**
   - or **Commands earlier than** and select the date using the drop-down menu calendar.

4. Click either:
   - **Delete** to delete all the commands contained within the selected time interval.
     - *For a more specific deletion, see Deleting log elements to reduce the log size.*
   - **Consolidate** to delete only the intermediate commands contained within the selected time interval.
     - *You can consolidate the latest information.*



Backing up the repository log

To back up the repository log you have to export the repository log in a file and save it.

This file can be exported in:

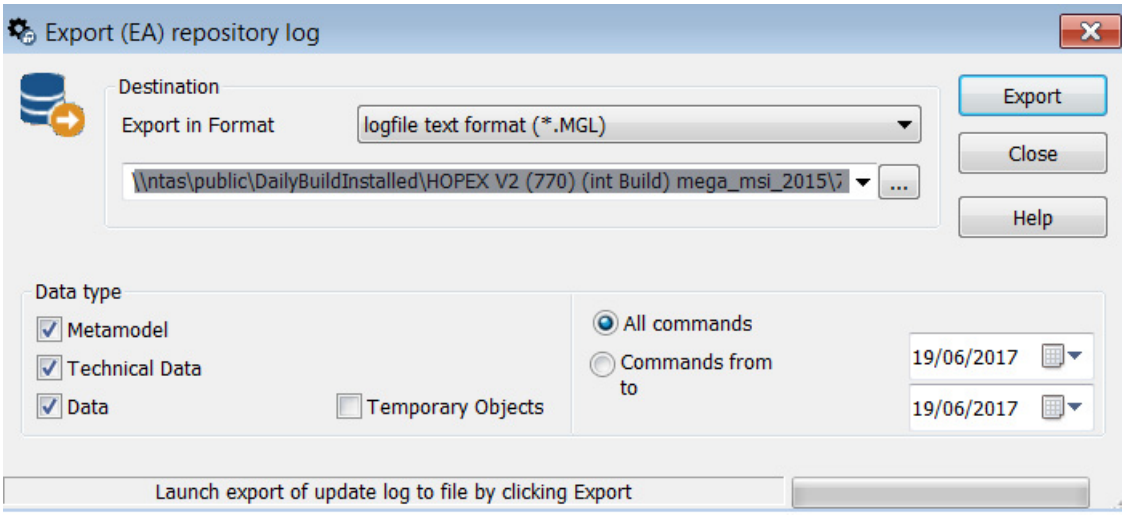
- **logfile text** (.mgl).
  Name format of the exported file is "LOGmmdd.mgl", where "mmdd" represents logfile export date month and day.
- **XML MEGA** (.xmg)
  The exported file is in the form of an XML file containing commands or data (objects and links).

To avoid backup duplications and save backup time and size, select **Commands from** "selected date" **to** "selected date" and select a starting date corresponding to the end date of your last backup, and select the ending date you require.

To backup the repository log in the form of command file:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*

2. Right-click the repository concerned and select **Repository Log > Export**.
   The **Export repository log** dialog box opens.



3. Select export format.
4. (Optional) By default the file is saved in **db\<repository name>WORK** folder. If needed, modify the name and folder proposed by default to save the export file. The **Browse** button ... allows you to browse the folder tree and select the folder in which the file will be placed.
5. In the **Data Type** frame, select the type of exported modifications:
   - **Metamodel**: to extract the metamodel from the system repository. This is useful if the standard metamodel has been modified.
   - **Technical Data**: to export, from the system repository, the changes made to data such as descriptors and queries.
   - **Data**: to export changes made to repository data, particularly the workspaces.
   - **Temporary Objects** - these objects are created when executing requests, consulting objects (stored in the history), etc. Usually you do not need to export these objects.
6. To export the commands that correspond to a defined period only, select **Commands from** / **to** and specify the relevant dates.
7. Click **Export**.
   The **Execution Report** appears.
8. Click **OK**.
   The file is exported and saved in the specified folder.

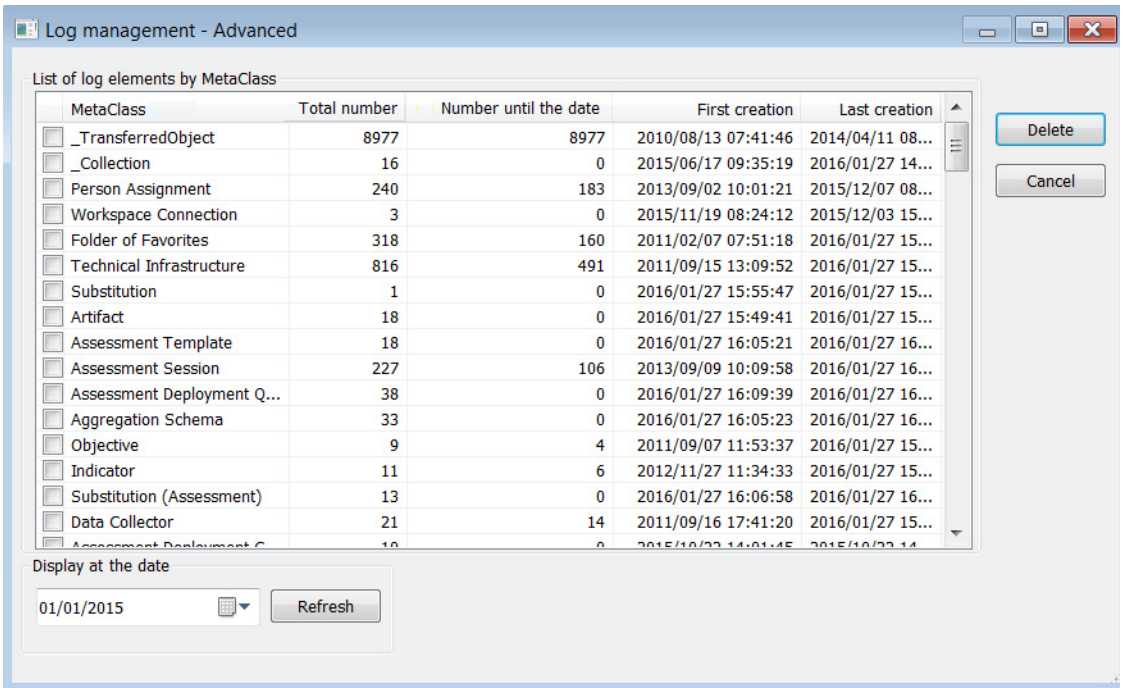
Deleting log elements to reduce the log size

With an SQL Server repository storage, you can delete log elements more specifically: you can delete only log elements relating to specific MetaClasses, earlier than a selected date.

– *If you do not want to have to delete log elements you are not interested in, see Modifying MetaClass loggability.*

To delete log elements regarding specific MetaClasses:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository and select **Repository Log > Manage Repository and Object Log**.
   The **Manage Log** dialog box opens.
3. Click **Advanced**.
   The **Log Management - Advanced** window shows the log element number by MetaClass and specifies the first and last creation date.



4. In the **Display at the date** pane, select the date until which you want to delete the log elements.
5. Click **Refresh**.
   The **Number until the date** column shows for each MetaClass the log element number until the selected date.

   M *Click the **Number until the date** column header to sort the MetaClasses, with the most populated at the top.*
6. In the **MetaClass** column, select the MetaClass for which you want to delete the log elements until the selected date.
   - *You can select several MetaClasses.*
7. Click **Delete**.
   Log elements are deleted.

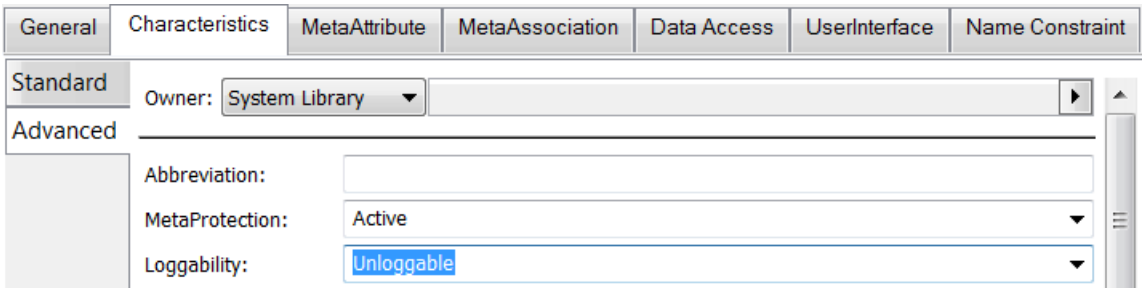
Modifying MetaClass loggability

To reduce the number of objects generated in logs, you can modify logging of MetaClasses you do not want to track.

– *In case of inter-repository consolidations, see Modifying the log behavior.*

To modify MetaClass loggability:

1. Connect to **HOPEX**.
   – *See Connecting to HOPEX.*
2. Open the **MetaStudio** navigation window and expand folders **MetaClass > MetaModel**.
   – *Alternatively click **Explore** and use explorer to find the **MetaClass** or **MetaAssociation** object.*
3. Open the properties dialog box of a **MetaClass** or **MetaAssociation**.
4. In the **Characteristics > Advanced** tab, for the loggability attribute, select **Unloggable** value.



Occurrences created, updated or deleted in a private workspace are dispatched, but certain commands are not available in object histories or in repository activity.

Managing the Cache in RDBMS Environments

The RDBMS local cache avoid multiple requests when multiple users are on the same repository view. Access for the following users is speed up.

By default the RDBMS local cache is activated on all the repositories:

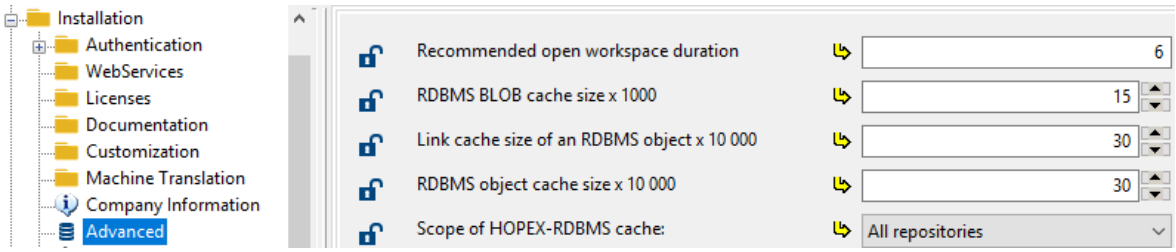
- You must configure your anti-virus accordingly.
  – *See Configuring the Anti-Virus According to HOPEX Data.*
- You can increase RDBMS cache size (memory).

Increasing RDBMS cache size (memory)

In RDBMS storage case, with repositories including a large amount of objects, we advise you to increase the size of RDBMS caches. The larger your cache space, the fewer the network exchanges and the better your **HOPEX** performance.

To increase cache size:

1. Connect to **HOPEX Administration** and select the environment in which the repository is referenced.
   - *See Connecting to an Environment.*
2. Right-click the environment and select **Options > Modify**.
3. In the environment options, select **Options > Installation > Advanced**.



4. Increase cache sizes depending on your memory space.

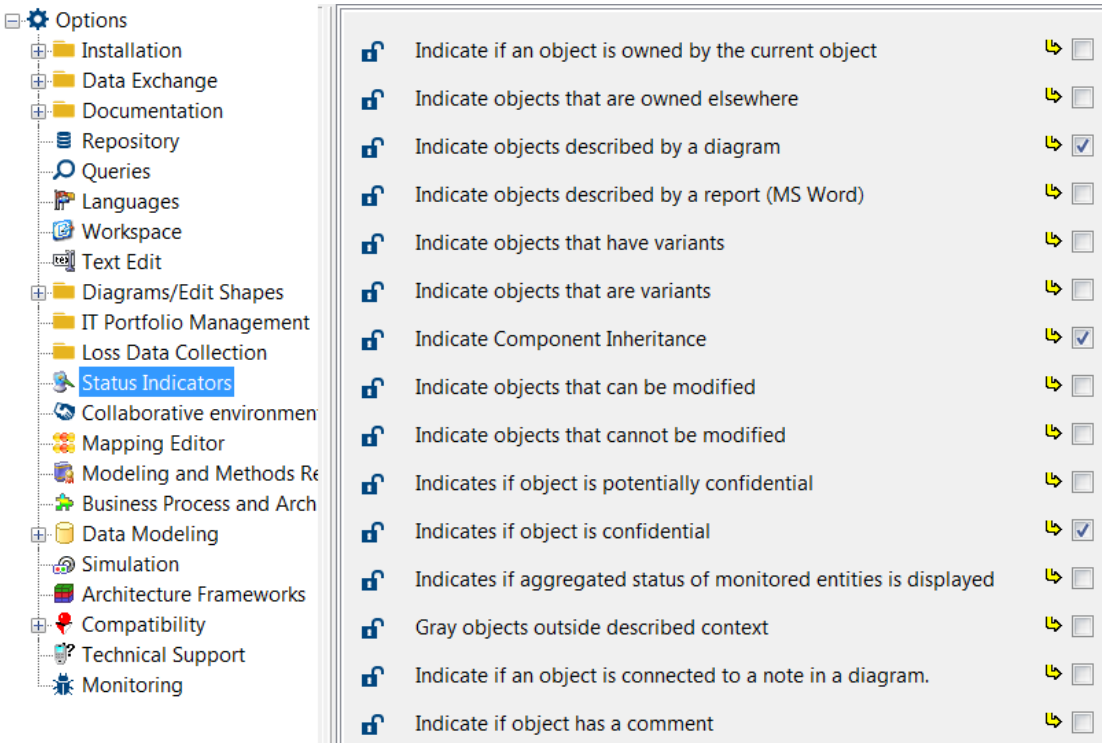
Managing Status Indicators

Use of status indicators generates large quantities of queries on repositories. Select only those indicators you require.

To modify indicator backup selection:

1. Connect to **HOPEX Administration** and select the environment in which the repository is referenced.
   - *See Connecting to an Environment.*
2. Right-click the environment and select **Options > Modify**.

3. In the **Environment Options** dialog box, select **Status Indicators**.



4. In the right pane, select only those indicators you require.
5. Click **OK**.

Deleting RDBMS Repository Temporary and Historical Data

To prevent repository size increase and keep optimized performances, you should regularly delete data of the completed private workspaces of **HOPEX** users.

MEGA recommends that you delete all RDBMS repository temporary and historical data:

- every week for less than 10 users
- every evening if you have more than 10 users.

To delete the temporary and historical data of an RDBMS repository, you (or your database administrator) must include the following procedures in your regular maintenance tasks:

– *To perform these procedures, see **RDBMS Repository Installation Guide** deployment guide.*

- SP_CLEAN_MEGA_DATABASE

  – *To consult the date of last private workspace cleanup (last execution of this procedure), see Consulting and Modifying Repository Properties.*

- SP_CONSOLIDATE_MEGA_DATABASE

  – *To consult the date of last consolidation (last execution of this procedure), see Consulting and Modifying Repository Properties.*

Performing Repository Regular Maintenance Tasks

With RDBMS repositories you must include a regular maintenance plan. You (or your database administrator) should, for each repository (SystemDb repository included), perform the following maintenance tasks regularly (at least once a week):

- rebuild indexes
- update the statistics
- (optional) shrink the logs regarding the policies
- perform the stored procedures

  – *See Deleting RDBMS Repository Temporary and Historical Data.*

  - SP_CLEAN_MEGA_DATABASE
  - SP_CONSOLIDATE_MEGA_DATABASE

Cleaning up a Repository

During modeling work, handling different objects results in creation of temporary objects (temporary queries, etc.). It is probable that these objects will subsequently be unnecessary. To avoid unnecessarily increasing repository size, you can delete these temporary objects.

To clean up a repository:

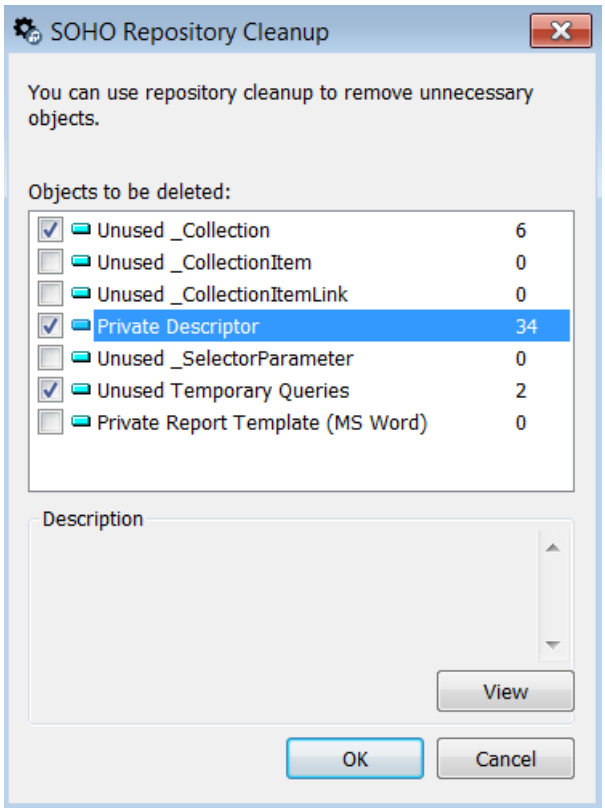
1. Connect to **HOPEX Administration** and select the repository concerned.

   – *See Accessing Repositories.*

2. Right-click the repository concerned and select **Object Management > Repository cleanup**.

   - *Alternatively, connect to **HOPEX** and in the menu bar, select **File > Properties**. In the properties of the repository to which you are connected, select **Characteristics**.*

   The **Repository Cleanup** dialog box opens.



3. (Optional) To view the content of an object group, click the object group name and then click **View**.
4. Select the object groups to be deleted and click **OK**.

   - *By default, all the object groups that include objects are selected.*

   Selected object groups are deleted from the repository.

Configuring the Anti-Virus According to HOPEX Data

To optimize anti-virus activity and to avoid unnecessary slowdown, you must configure what should be managed or not by the anti-virus.

You must exclude the following files from anti-virus scanning:

	Access mode	Location	File type
Program files external to HOPEX Files used by **HOPEX** to display user interface elements in Web or Windows interfaces	***Read***	Program Files (x86) \ MEGA\ MEGA-STD and sub-folders	Proprietary format: ***.mgs*** (vectorial shapes in diagrams) Public formats: ***.gif***, ***.ico*** and ***.bmp***
Full-text search indexes Files used by the workstation or server executing the search	***Reading*** by the workstation or node executing the search ***Writing*** by the Administration station and by the server programming index creation	<Environment folder>\Db\ <Repository name>\ <Repository name>.ix	Proprietary format: ***.ix*** (indexes) Public formats: ***.log*** and ***.dat***
Data cache Files used by **HOPEX** to improve its performance **RDBMS data** cache subfolder is automatically populated, see Managing the Cache in RDBMS Environments	***Read/Write***	<ProgramData>\MEGA \HOPEX <version HOPEX>\cache	Public format: ***.mgc***
Import/export files Files generated or read during import/export/logical backup processes	***Reading*** by the import function ***Writing*** by export and logical backup functions	Selected by the user	Proprietary formats: ***.mgr*** or ***.xmg***

	Access mode	Location	File type
Logfiles Files generated by **HOPEX** for each repository when the **Backup Logfile** option is activated. Files generated by the Export command	***Writing***	<Environment folder>\Db\<Repository name>\<Repository name>.transaction or <Repository name>.Log	Proprietary format: ***.mgl***
MUST license Note: if the anti-virus does not allow such a configuration, allow ***Reading/Writing*** for all of the files on the license oath and sub folders.	***Reading/Writing*** of the files on MUST license path and under this path	<Installation folder>\Cfg\megasite.ini gives the MUST License path: [must license] path=	Public format: ***.ini*** Proprietary formats: ***.tnk*** and *.***usr***
	Reading of the files on MUST license path and under this path		Proprietary format: ***.must***

REFERENCING AND UNREFERENCING A REPOSITORY

Referencing and unreferencing a repository is carried out via the drop-down menu of the repository.

See successively:

- Referencing a Repository
- Unreferencing a Repository
- Protecting the Referencing of a Repository

Referencing a Repository

If you have moved or copied a repository without using MEGA move or restore commands, you will have to reference the repository so that the environment recognizes it.

– *To reference a repository in another environment, the two metamodels must be identical.*

To reference a repository:

1. Connect to **HOPEX Administration** and select the environment in which you want to reference the repository.

   – *See Connecting to an Environment.*

2. Right-click the **Repositories** folder and select **Create reference**.
3. (If the repository is password-protected) Enter the repository password and click **OK**.

   – *See Protecting the Referencing of a Repository.*

4. Select **a repository from SqlServer**.

   – *For more details on **HOPEX** repository server type, see the **HOPEX** deployment guide.*

   The HOPEX **Repository Selection** dialog box opens. This dialog box allows you to create a reference for a new repository in the environment.

5. Indicate where the repository .EMQ file is located. The repository name is automatically indicated and cannot be modified.

The repository is accessible exactly as repositories created in the normal way.

– *you must save and then restore a repository to move it from one environment to another. The users and metamodel of the two environments must be defined identically so that transfer occurs without reject.*

– *To also copy these objects, import the missing part of the metamodel. One way of doing this is to upgrade the site or the environment. Then, import the rejected commands if you used the backup-restore procedure.*

– *Check that the repository is not simultaneously referenced in two different environments. Compatibility errors may occur if the environments are not identical.*

Unreferencing a Repository

You can delete a repository reference from an environment. This action does not delete the repository.

> P **If you delete a reference of a password-protected repository. you must know this password to reference the repository again.**

To delete a repository reference:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository of which you want to delete the reference and select **Delete Reference**.
   A message requests confirmation.
   > P **If the repository is password-protected, you must know this password to reference the repository again.**
3. Click **OK** to delete the repository reference.
   The repository reference is deleted, and can be created in another environment on condition that the metamodel is identical.
   - *A repository must be referenced in only one environment. It is important to check that the reference for your repository was deleted in its original environment.*

Protecting the Referencing of a Repository

You can password-protect the referencing of a repository.

See:

- Adding a referencing password to a repository
- Modifying/Canceling a repository password

Adding a referencing password to a repository

To password-protect the referencing of a repository:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository and select **Protect with a password**.
   The **Password Protect Repository** dialog box appears.
3. In the **Password** field, enter a password and confirm it.
   - *The password must contain at least eight characters.*
4. In the **Security question** pane:
   - enter a **Question**.
   - enter the **Answer** to the question.

5. Click **OK**.
   The repository password is requested on creation of a repository referencing.

Modifying/Canceling a repository password

To modify or cancel a repository password:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Password Protect**.
   The verification window for the password appears.
3. In the **Enter the password of the current repository** field, enter the repository password.
4. Click **OK**.
5. In the **Password** pane:
   - To modify the existing password, enter a password and confirm it.
     - *The password must contain at least eight characters.*
   - To cancel the password, leave the fields empty.
6. Click **OK**.

MANAGING WORKSPACES

Workspaces are managed by the administrator.

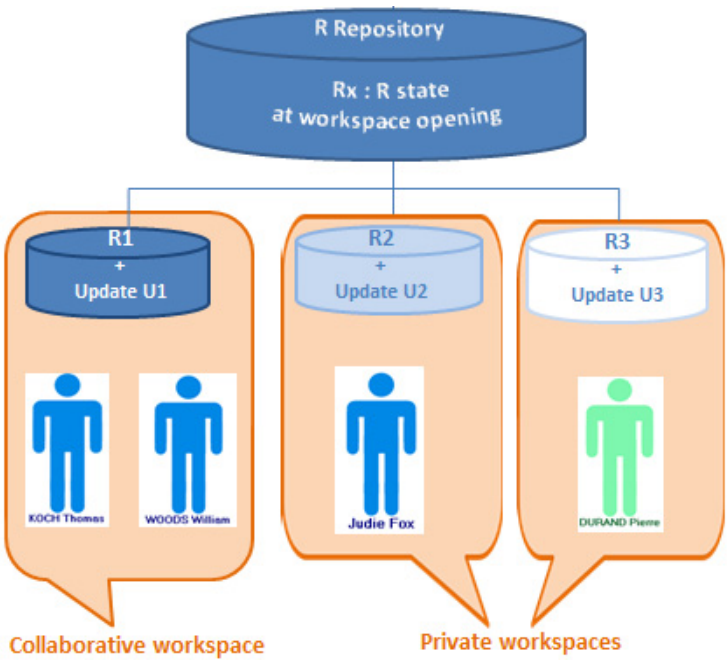
The following points are covered here:

- Private Workspace Principle
- Using Your Private Workspace
- Workspace Administration
- Private Workspace Life: Example
- Viewing Updates
- Managing locks

PRIVATE WORKSPACE PRINCIPLE

In a traditional management application, the user cannot control the opening duration of his/her workspace: the end of a data entry corresponds to a definitive save of his/her work.

With **HOPEX** the user controls management of his/her workspace: opening, closing, dispatch, refresh.



Private workspace

When a user connects to a **HOPEX** desktop (**Windows Front-End**) and to certain **Web Front-End** desktops, he/she opens a *private workspace.* This private workspace is a temporary view of the repository (repository snapshot) at the moment of user connection.

The user then sees:

- initial repository snapshot objects of his/her visibility scope
- updates he/she has executed on these objects.

The user decides when he/she wants to integrate his/her repository updates and make them visible to other users. To do this, he/she dispatches modifications.

– *See Dispatching Your Work.*

The user controls opening duration of his/her private workspace.

– *The private workspace is open for each user on both his/her repository and on the system repository. A private workspace always exists on system repository even if the user is not using any of the system repository objects.*

Several users can have private workspaces open in parallel. In his/her private workspace, the user is independent of updates carried out simultaneously by other users in their respective private workspaces.

– *Locks inform the user of objects modified by others. See Managing locks.*

The user can also update his/her private workspace with the updates of other users. To do this, the user refreshes his/her private workspace.

– *See Refreshing Data.*

HOPEX allows several users to work at the same time.

Collaborative Workspace

– *This functionality is available with an RDBMS format repository only.*

The user can also share his/her private workspace with other users before dispatching his/her modifications and making public his/her work to all other users. To do this, the user creates a **Collaborative Workspace** from his/her private workspace.

– *See the **HOPEX Common Features** guide, section "Working in a Collaborative Workspace".*

A user can, in parallel:

- have a private workspace
- be the owner of several collaborative workspaces
- be invited to participate in as many collaborative workspaces as he/she wishes.

USING YOUR PRIVATE WORKSPACE

A private workspace is a temporary view of the work repository allocated to a user before the user dispatches his/her work. This view of the repository is only changed by modifications made by the workspace user, independently of concurrent modifications made by other users. This private workspace exists until it is refreshed, dispatched or discarded.

Note that a private workspace is kept when the user disconnects from the repository, unless the user indicates otherwise.

The following points are detailed here:

- Connecting to HOPEX
- Saving Sessions
- Workspace Properties
- HOPEX Repository State Changes
- Dispatching Your Work
- Dispatch Conflicts
- Rejects When Dispatching
- Dispatch Report
- Refreshing Data
- Conflicts When Refreshing
- Discarding Work
- Exiting a Session
- Workspace Administration
- Viewing Updates
- Exporting Your Private Workspace Log

Connecting to HOPEX

When you connect to **HOPEX**, you can:

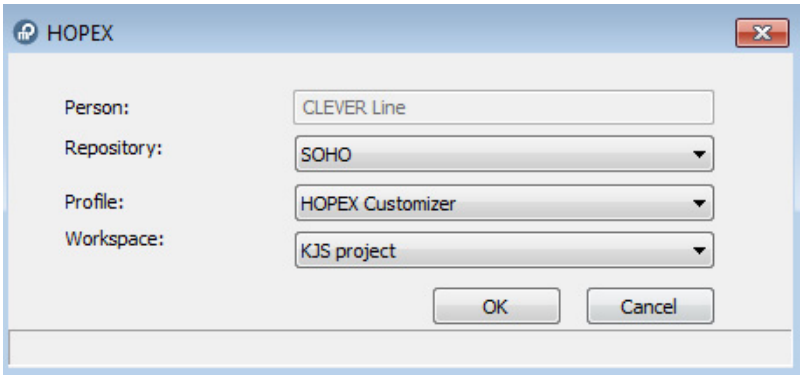
- create a private workspace (if you do not already have one).
  - *You can only have one private workspace open in the same environment.*
  - *The private workspace is open for each user on both his/her repository and on the system repository. A private workspace always exists on system repository even if the user is not using any of the system repository objects.*
- resume work in your private workspace
- resume work in a collaborative workspace

To connect to **HOPEX**:

1. Start the **HOPEX** application.
   The authentication dialog box appears.
2. In the **Login** field, enter your login.
3. (Optional) In the **Password** box, enter your password if required.
4. In the **Environment** field, select your work environment.

5. Click **OK**.
   You are authenticated, your name appears in the **Person** field.
6. In the **Repository** field, select your work repository.

   – *If you already have a private workspace open, the repository is automatically selected and this field is grayed. To change repository, you must first dispatch or discard your current private workspace.*
7. In the **Profile** field, select the profile with which you want to work.



8. If:
   - you do not have a collaborative workspace available, the **Workspace** field is not available. Click **OK**.

   A private workspace is created and your desktop opens.

   – *If you already have a private workspace open, you should connect to it. If you want to change profile or repository, you must close the private workspace that is open.*

   - you have at least one collaborative workspace available, in the **Workspace** field, select **Access Private Workspace** or select the collaborative workspace to which you want to connect, or select **Create Private Workspace** (if one has not already been created). Click **OK**.

     – *A user has at most one private workspace in progress in an environment, but can have in parallel several available collaborative workspaces.*

     Your desktop opens.

A private workspace comprises a set of files located in a sub-folder of the repository:

"<EnvironmentName>\DB\<RepositoryName>\<RepositoryName>.Transactions\xyz.*"

where "xyz" represents the user code.

Note that a private workspace cannot be separated from its repository (these files cannot be used independently).

Saving Sessions

) *A session is the period during which a user is connected to a repository. A session begins when the user authenticates and ends when he/she exits HOPEX. Sessions and private workspaces can*

overlap. When you dispatch, refresh or discard a private workspace, a new private workspace is created in the same session. Conversely, a user can keep his/her private workspace when exiting a session.

To save modifications you have made in your *session* since the last save:

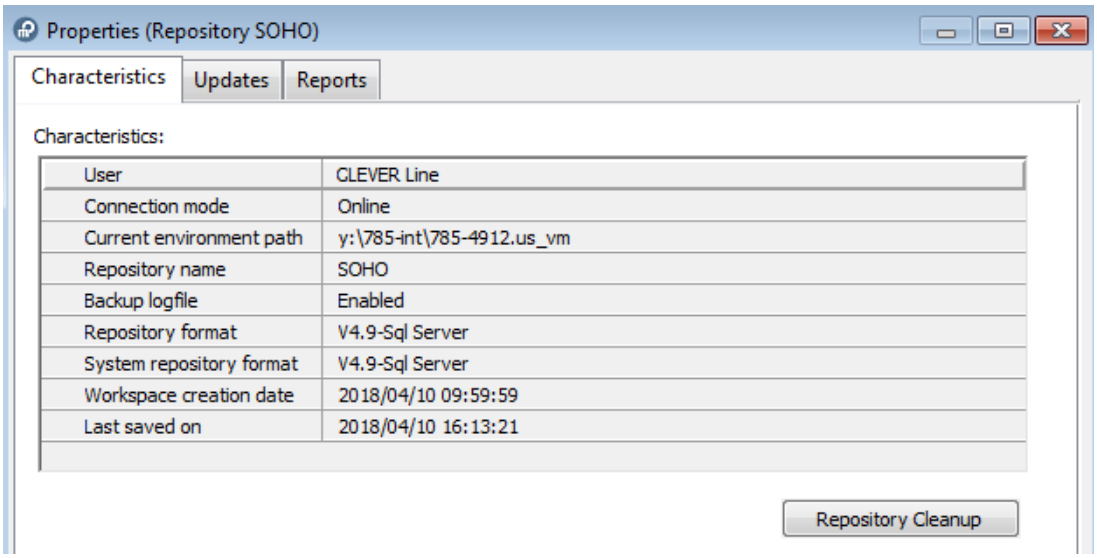
› In the **HOPEX** menu bar, click **Save** .

– *These modifications are not saved in the repository. To save your modifications in the repository, you must dispatch these modifications, see Dispatching Your Work.*

Workspace Properties

To consult your workspace properties:

1. Connect to **HOPEX**
   – *See Connecting to HOPEX.*
2. In the **HOPEX** menu bar, select **File > Properties**.
   The properties dialog box of your workspace appears.



The workspace properties dialog box provides the following information on the current workspace:

- current **User**
- information on the current repository: its **Name**, **Backup logfile**, **Format**, **Workspace creation date**, **Last saved on**.
- **Repository Cleanup** option.
  – *For more information on repositories, see Managing Repositories.*

HOPEX Repository State Changes

The integrity of the repository is assured by successive changes in its state.

– *See example Private Workspace Life: Example.*

When repository updates are executed in a private workspace, they are only visible to other users when the user dispatches his/her work.

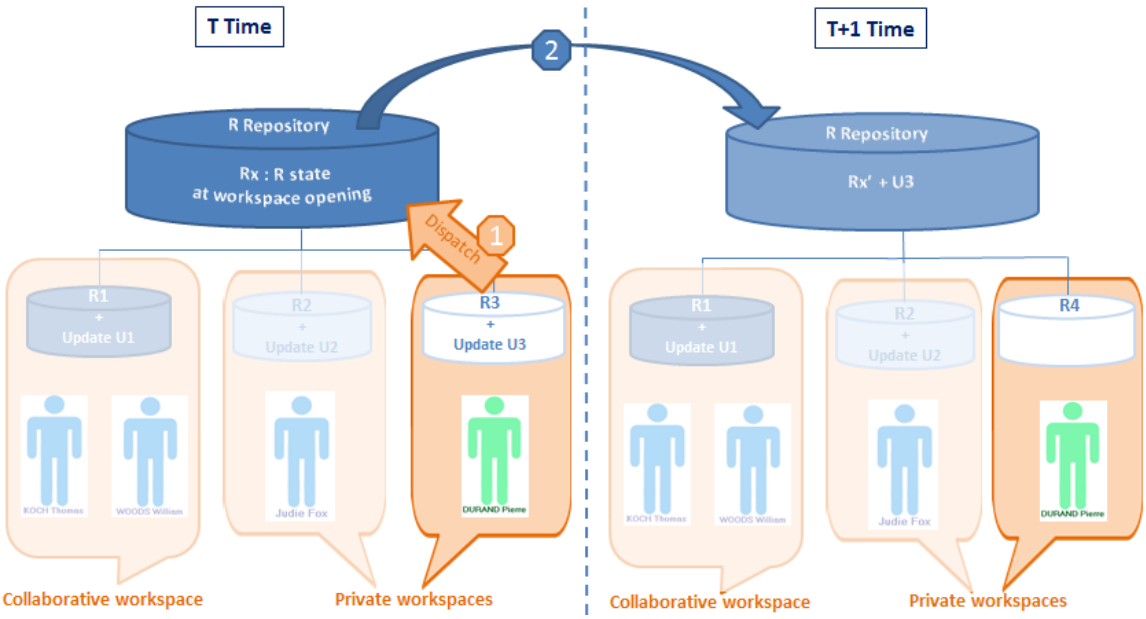
The repository changes from state N to state N+1 with memorization of new data related to the previous situation (state N).

If the user does not save updates, the changes made since the last valid state are forgotten. The repository remains in state N. Note that **HOPEX** repository state changes are managed automatically.

A workspace opens on the latest states of the repository and system repository.

Dispatching Your Work

Dispatch consists of making public the work carried out from a private workspace, or the work of participants in a collaborative workspace.



Dispatch allows:

- a user to make available to other users the modifications he/she has made to the repository.
- users of a collaborative workspace to make available to other users the modifications they have made to the repository.
- other users to have these updates available when they open a new workspace, whether this be after dispatch, refresh or discard of their current private workspace.

Dispatch:

- executes an update of the **HOPEX** repository and the system repository.
- creates a new workspace for the user containing all updates since creation of his/her previous private workspace.

Note that only one user can dispatch at a time. When several users dispatch their work at the same time, a dialog box appears asking the user if he/she wants to queue his/her dispatch. This allows the user to exit **HOPEX** without having to wait until the works from other queued private workspaces are dispatched.

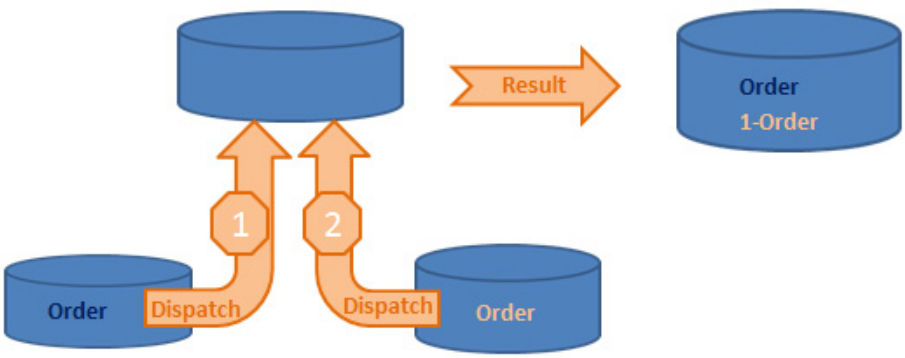
- *See Dispatch Conflicts.*

Dispatch Conflicts

The dispatch process automatically manages most conflicts that may arise when several users make updates.

Creation of duplicated objects

When duplicate objects are created, a prefix is added before the name of the second object.



Two users create an object with the same name. The first to dispatch his/her work actually creates the object. The second user to dispatch his/her work creates an object with a prefixed name.

This is indicated in the dispatch report.

The administrator or the users can then decide to rename one of these objects if they are actually different, or combine them into one object if they are in fact the same object.

– *See Merging Two Objects.*

Deletion of already deleted objects or links

As the deletion has already been performed, nothing happens. There is no mention of this in the dispatch report.

Modifying or linking a renamed object

This happens when a user modifies an object that in the interim has been renamed by another user, or tries to link something to this object. The new one is kept and the changes are executed normally. The object can be found using its *absolute identifier*. Note that this does not create a reject, but the dispatch report indicates that the object was renamed.

) *An absolute identifier is a string of characters associated with each object in the repository. This string is computed using the date and time the session was opened, the number of objects created since the session started, and the object creation date (in milliseconds). An absolute identifier provides a unique way to identify an object in the repository, so that even if the object is renamed, all the links to it are retained.*

Rejects When Dispatching

There are normally no rejects when dispatching of updates carried out in a private workspace. Most conflicts are managed automatically.

Rare cases of rejects are listed in the *rejects file*.

) *When updating a repository (importing, restoring, dispatching), a reject file is created in order to store rejected commands. Rejected commands are stored with the reason for which they were rejected. These are found in the "MegaCrd.txt" file located in the environment folder.*

Change in writing access values between opening and dispatching a private workspace

If you have access to the writing access management function, you can, for example, protect an object in the repository while a user is deleting it in his/her private workspace. When the user dispatches his/her work, he/she is no longer allowed to delete the object and the deletion is rejected.

Rename/create collisions

A user renames an object in his/her private workspace, for example, from "Customer" to "Customer***s***". Then another user dispatches his/her private workspace in which he/she created an object having the same name "Customer***s***".

When the first user dispatches his/her private workspace, since the "Customer***s***" object already exists, the object "Customer" cannot be renamed "Customer***s***". The rename command will therefore be rejected.

Verifying link uniqueness

A user creates a link for which there is a uniqueness check. For example, he/she indicates that the "Order" message is sent by the "Customer" org-unit. In the meantime, another user indicates in his/her private workspace that the "Order" message is sent by the "Customers" org-unit. When the second user dispatches his/her private workspace, the link is rejected if the uniqueness control imposes that a message can be sent by only one object.

– *See the **HOPEX Power Studio - Imposing MetaAssociation Uniqueness** Technical Article for information on MetaAssociation uniqueness check.*

Attribute uniqueness (other than name)

Other attributes besides name may also be checked for uniqueness. If two users give two different objects the same value for this attribute, the second update will be rejected.

Updating a deleted object

If a user has made changes to an object in his/her private workspace and the object has been deleted by another user, the updates are rejected. The **Connect** and **Change** commands concerning this object are also rejected. All these rejects are listed in the private workspace report file.

Dispatch Report

The report file can be accessed from **HOPEX Administration**.

To view the report file:

1. Connect to **HOPEX Administration** and select the environment concerned.

   – *See Connecting to an Environment.*

2. Right-click the environment and select **Reports > Open**.
   A new window allows you to view report contents.

   – *See Viewing the Environment Report File for more details on these dialog boxes.*

The error level is indicated in the dispatch report.

- Error level 0: no error.
- Error level 2: minor errors such as attempts at creating already existing objects or links.
- Error level 4: the actions involved nonexistent objects or links.
- Error level 8: a system error was encountered during the update.
- Error level 16: update aborted because of a problem such as insufficient disk space.

  – *These error levels are the same as those used for manual file imports.*

  – *When manually importing a file, rejects concerning the creation of already existing objects or links can be filtered out using the Reprocess option.*

Objects that were renamed are also listed in the report.

Refreshing Data

A user can see modifications dispatched by other users of this repository without dispatching his/her own modifications. To do this, the user refreshes his/her data.

A user can refresh his/her data:

- in his/her private workspace
  The system creates a new private workspace, into which the *private workspace log* of the user's previous modifications is automatically imported.

) *The private workspace log contains all modifications made by a user in his/her private workspace. It is applied to the repository at dispatch, then automatically reinitialized. This log is stored in the EMB private workspace file.*

  Refreshing allows a user to incorporate repository and system repository changes made by other users, without dispatching current work.
- in a collaborative workspace.
  The system then creates a new collaborative workspace for all participants in the collaborative workspace, into which is automatically imported the collaborative workspace log containing modifications previously made by participants.

  – ***MEGA** recommends that you warn other participants before executing refresh.*

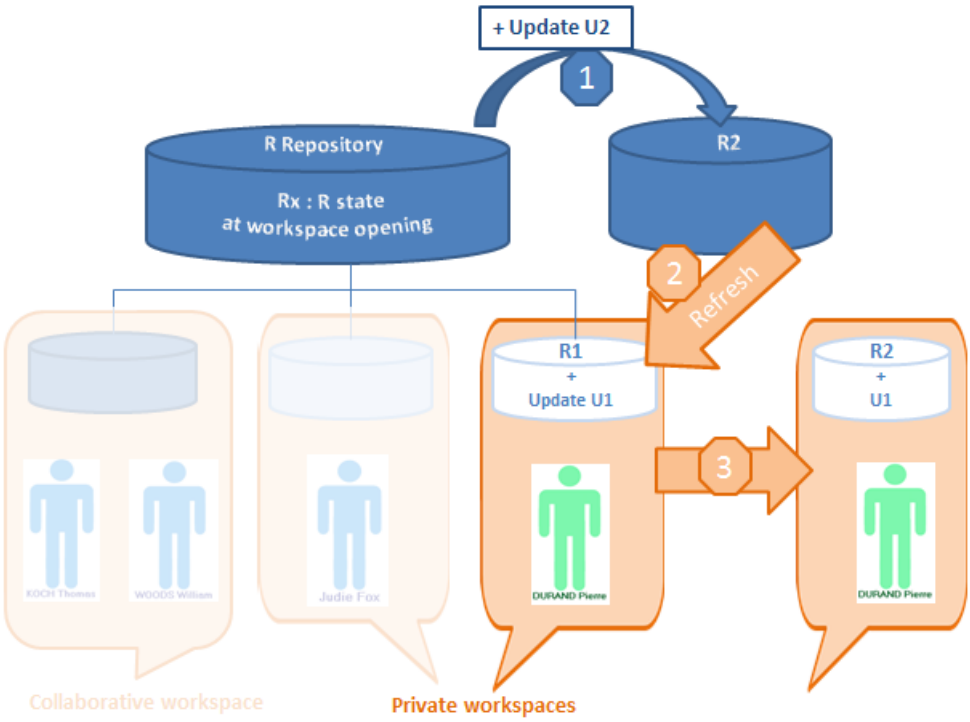
  Refreshing allows a user to incorporate repository and system repository changes made by other users, without dispatching current work.

Refreshing a private (or collaborative) workspace.

- does not update repository or system repository state.
- does not unlock objects modified in the private workspace.

  – *see Managing locks.*

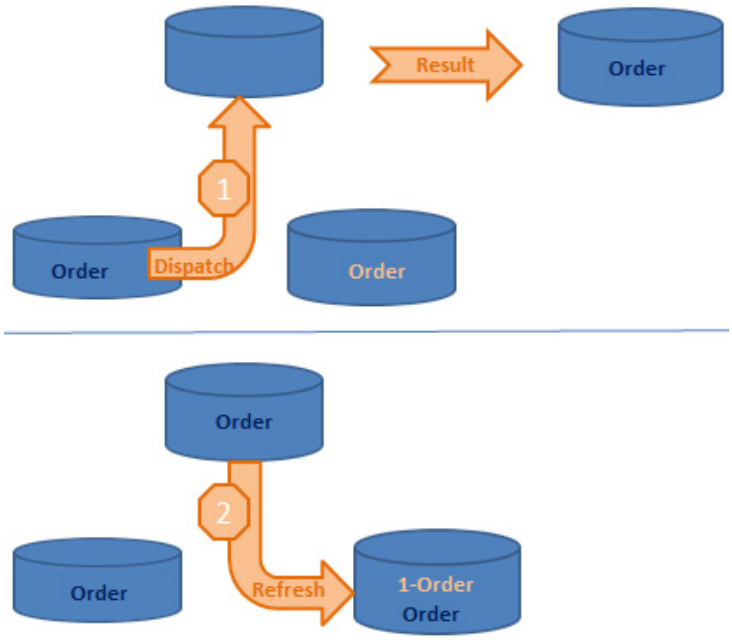
When a user resumes work on his/her private space that has lasted longer than the limit set by the administrator (the default is 6 days), **HOPEX** proposes that the user refreshes or dispatches his/her work.


+ Update U2
1
R Repository
Rx : R state
at workspace opening
R2
2
Refresh
R1
+
Update U1
3
R2
+
U1
KOCH Thomas
WOODS William
Judie Fox
DURAND Pierre
DURAND Pierre
Collaborative workspace
Private workspaces

Conflicts When Refreshing

Conflicts when refreshing are the same as when dispatching, but they apply to the private workspace only.

– *For more details on the main causes of rejects, see Dispatch Conflicts and Rejects When Dispatching.*



As is true in dispatching, if two objects are created with the same name, the second object name is prefixed:

```
The second "Order" object is renamed "1-Order".
```

Discarding Work

Discarding a workspace (from a private or collaborative workspace) cancels all modifications made since the last dispatch. *Discard* of work causes loss of work carried out since opening of the private or collaborative workspace, including modifications to the desktop. A warning message reminds the user of this. Use discard with care.

Discarding work from a private workspace

In **HOPEX**, to discard your private workspace:

1. (Optional) It is advisable to export the private workspace before confirming the discard, see Elements that could be useful to back up.

2. In the **HOPEX** menu bar, select **File > Discard**.

   – *You can also discard your private workspace at disconnection, see Exiting a Session (choose not to dispatch modifications).*

Discarding work from a collaborative workspace

Only the collaborative workspace **Owner** can discard the collaborative workspace.

– *See the **HOPEX Common Features** guide, section "Working in a Collaborative Workspace".*

In **HOPEX**, to discard your collaborative workspace:

1. (Optional) It is advisable to export the collaborative workspace before confirming discard, see Elements that could be useful to back up.
2. In the **HOPEX** menu bar, select **File > Workspace > Discard**.

Exiting a Session

When you exit **HOPEX**, you close your session. From:

- your private workspace you can:
  - save in the repository the modifications you have made in your private workspace
  - keep the modifications you have made in your private workspace

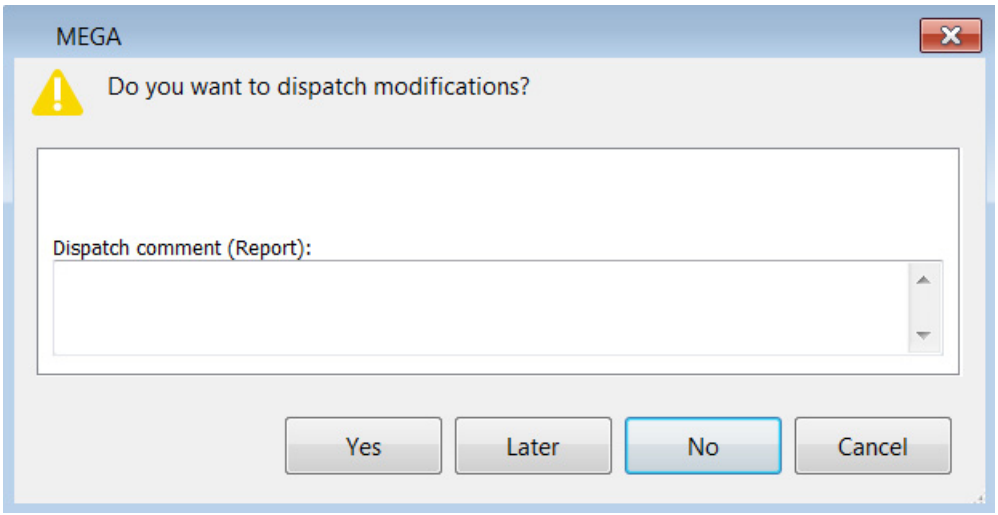
    – *These modifications will remain awaiting validation, subsequent modification, or deletion.*
  - cancel modifications you have made.
- a collaborative workspace you can:
  - keep modifications you have made

    – *These modifications are saved in the collaborative workspace. These modifications are not saved in the repository until the collaborative workspace is closed.*
  - cancel modifications you have made.

Exiting a session from a private workspace

In **HOPEX** (Windows Front-End), to exit your work *session*:

1. In the **HOPEX** menu bar, select **File > Exit**.
   The **HOPEX** exit dialog box appears.



2. (Optional) In the **Dispatch comment (Report)** frame, enter a comment to remind you of modifications made in your private workspace.

   – *This comment is added in the dispatch properties, see Viewing the Dispatches and their Content.*

3. Select your **HOPEX** exit mode.

- **Yes**

Modifications you have made in your private workspace are saved in the repository.

M *In order to work effectively as a team, it is recommended that you dispatch frequently and regularly. Other users can update their own private workspace without dispatching their work (menu **File > Refresh**).*

– *This exit mode also allows the user to select a different repository the next time he/she logs in.*

- **No**

All modifications you made since your last dispatch will be lost. You can use this option if you want to view data quickly and exit without impacting the repository.

– *Modifications to your desktop are also lost.*

- **Later**

This option allows you to keep your changes without impacting the repository. You can open your session later and continue working but other users are not yet seeing the changes you have made.

– *Click **Cancel** to not exit your private workspace.*

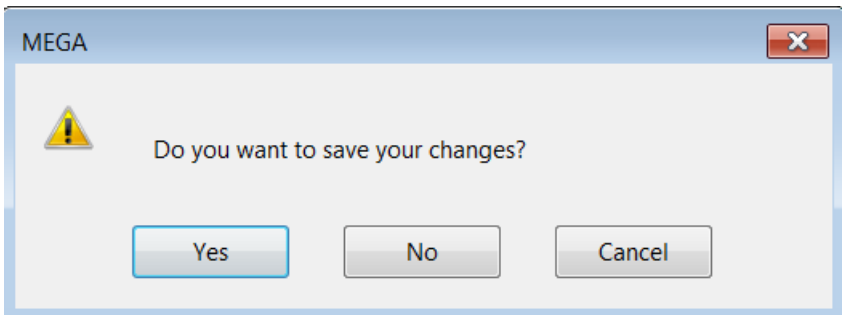
Exiting a session from a collaborative workspace

Exiting **HOPEX** from a collaborative workspace is the same whether you are its owner or not.

For as long as the collaborative workspace is not closed, participants can exit and rejoin the collaborative workspace at any time.

In **HOPEX** (Windows Front-End), to exit your work *session*:

1. In the **HOPEX** menu bar, select **File > Exit**.
   The **HOPEX** exit dialog box appears.



2. Click:
   - **Yes** to save your modifications in the collaborative workspace.
     You will be able to continue your modifications in a subsequent work session.
     These modifications are not saved in the repository. Users not participants in the collaborative workspace do not see these modifications.
   - **No** to cancel your modifications in the collaborative workspace.
     Your modifications are not saved in the collaborative workspace, but the latter remains available to carry out other updates.
     - *Click **Cancel** to remain in your collaborative workspace.*

WORKSPACE ADMINISTRATION

You can view the list of current workspaces and their characteristics (owner, delay, status).

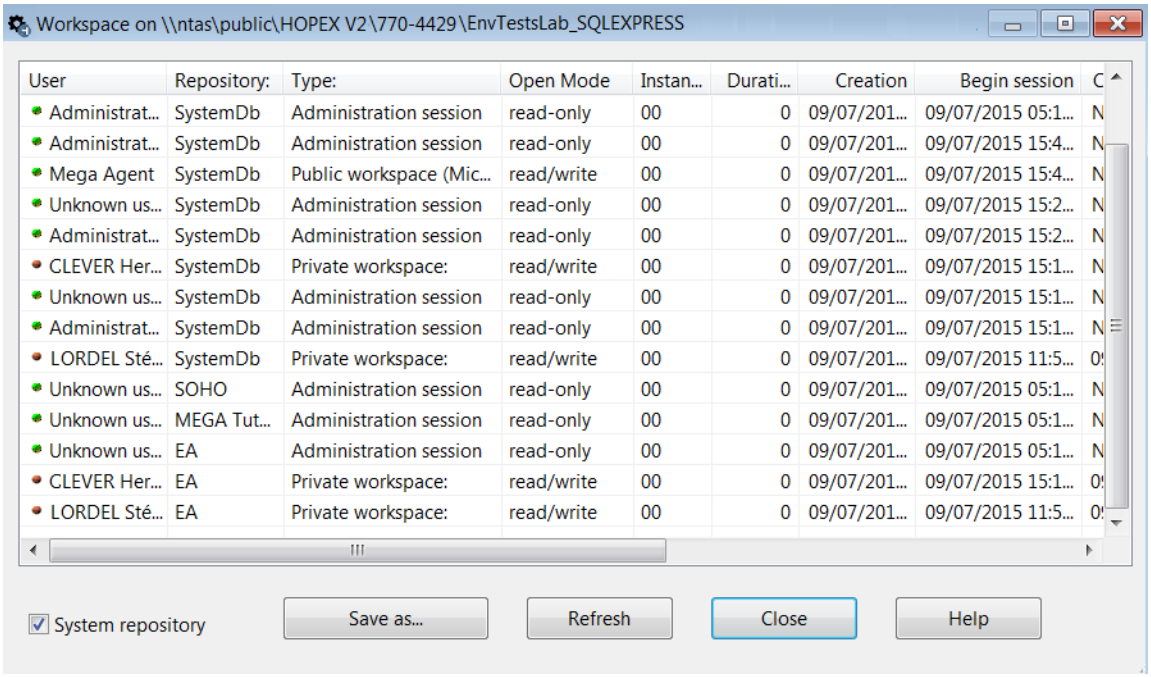
See:

- Accessing Workspace Management
- Deleting a Workspace

Accessing Workspace Management

To access the list of current workspaces in an environment:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **Repositories** folder.
3. Expand the folder of the repository concerned, right-click **Workspaces** and select **Manage**.
   The management page for workspaces currently in progress in the environment appears.



4. (Optional) Select **System Repository** to view system repository workspaces and the user connected to **HOPEX Administration**.

The private workspace dialog box details the following for each workspace:

– *To sort workspaces according to a column, click the header of the corresponding column.*

M *You can also arrange columns in the order you want. To do this, click the header of the column to be moved and, holding down the mouse button, move it to the required position.*

- its state:
  - the green button indicates that the workspace is open (private workspace) or in consultation (public workspace)
  - a red button indicates that the workspace is passive or that the user is disconnected.
- the **User** who created it and his/her **Code**.

  – *In the case of collaborative workspaces, only the collaborative workspace owner appears.*
- the **Repository** to which it relates
- the **Type** of workspace:
  - "Private Workspace":

  The user can modify data. His/her updates are kept in his/her private workspace until dispatched.
  - "Collaborative Workspace":

  The participants of a collaborative workspace can, depending on their access level, modify the data. Their updates are kept in the collaborative workspace until the owner of this workspace decides to dispatch them.
  - "Administration Session":

  The user is Administrator type. The user cannot modify data.
  - "Public Workspace":

  The user can modify data. His/her updates are immediately visible to all other users.
- the **Open Mode** of the workspace, for example:
  - "read/write" when a session is open.
  - "read-only" when the user is in consultation only.
  - no value, if the private workspace is passive (the user has saved his/her session but is not currently connected to **HOPEX**).
  - no value if the user is in offline mode
- its **Duration** in days
- its **Creation** date and time
- the **Start** and **End** dates and times of the last **Session**
- the **Location** where it is stored
- its **Dispatching Delay**, which indicates its state compared to the current state of the repository (number of saves since the private workspace began, whether manual or automatic).
  Each time the repository is updated, the difference between the private workspace view and the repository state increases. This difference is measured by the private workspace delay. Dispatching or refreshing the private workspace reduces the delay to zero.

  M *It is recommended that you refresh your workspace if its delay is greater than 50.*

Deleting a Workspace

The **HOPEX** administrator can delete a private workspace when this is passive.

The result is equivalent to discarding it.

To delete a workspace:

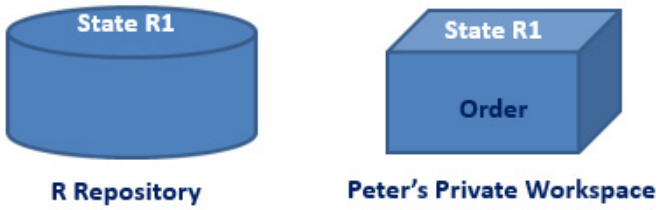
1. Open the workspace management dialog box.
   - *See Accessing Workspace Management.*
2. Right-click the workspace concerned and select **Delete**.

   When a workspace is deleted, the workspace in the work repository and that open on the system repository are deleted. Use private workspace deletion with care.

PRIVATE WORKSPACE LIFE: EXAMPLE

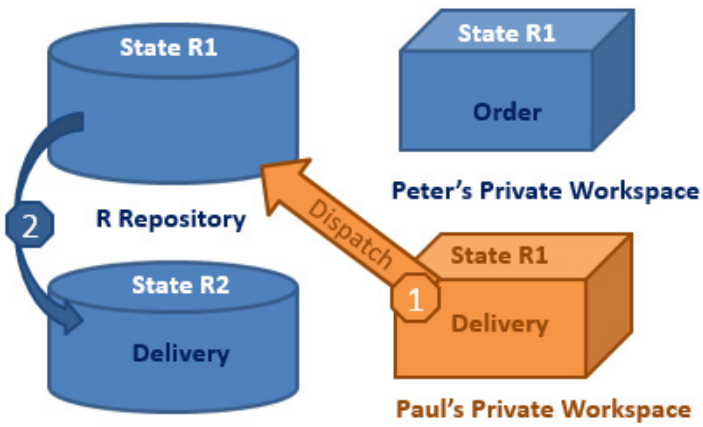
To illustrate how private workspaces work, the following is an example of some steps in the work of several users:

Private workspace 1



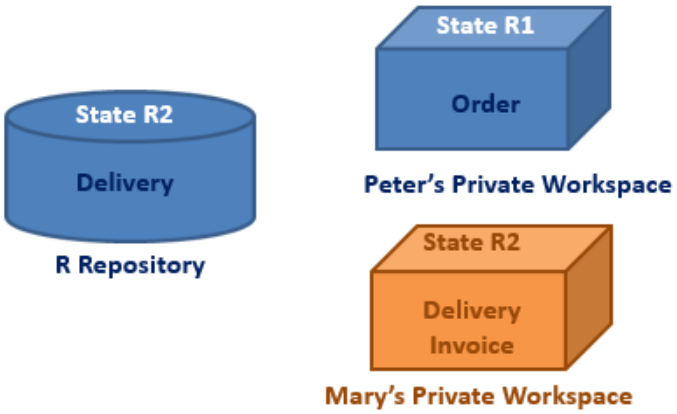
- Peter opens a private workspace, his repository view is state "n" (R1).
- He creates the "Order" message, which he links to the "Customer" org-unit.
- Simultaneously, Paul dispatches his work...

Private workspace 2



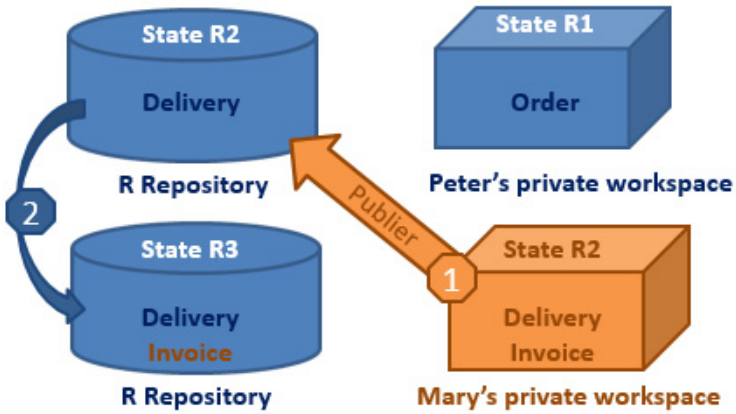
- Paul dispatched his work, which included the creation of the "Delivery" message, linked to the "Customer" org-unit.
- Paul's dispatch changes the repository to state "n+1" (R2).
- This new message is not seen from Peter's private workspace...

Private workspace 3



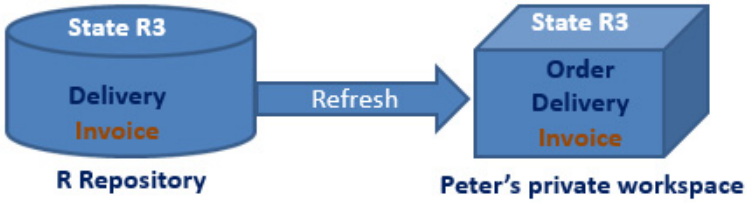
- Mary opens a new private workspace, her repository view is state "n+1" (R2).
- Mary creates the message "Invoice" connected to the "Customer" org-unit...

Private workspace 4



- Mary dispatches her work.
- The repository moves to state "n+2" (R3).
- Peter's view is still in state "n" (R1).
- Peter refreshes his private workspace...

Private workspace 5



- Peter has refreshed his private workspace.
- His view now corresponds to state "n+2" (R3).
- He can now see the "Customer" org-unit with the additions made by Paul and Mary.
- Peter dispatches his work...

Private workspace 6

- When Peter, Paul, and Mary have dispatched their work, all the modifications they have made are visible in state "n+3" (R4) of the repository.

VIEWING UPDATES

During their modeling work, users make additions to a **HOPEX** repository within their private workspace: they create objects, links between objects, diagrams, etc. Updates corresponding to user actions can be viewed in detail. You can back up all modifications made to a repository from a private workspace in a private workspace log, which can be exported in the form of a command file.

The following points are detailed here:

- Viewing Updates
- Viewing the Dispatches and their Content
- Exporting Updates
- Exporting Your Private Workspace Log
- Private Workspaces and Repository Size

Viewing Updates

The repository property windows shows the updates made in the repository.

– *You can also view the updates from the dispatch tree (**Repository Activity > Repository Dispatch** navigation window*

The repository property window shows the updates performed:

- on the current **HOPEX** repository (by default), or
- on the system repository.

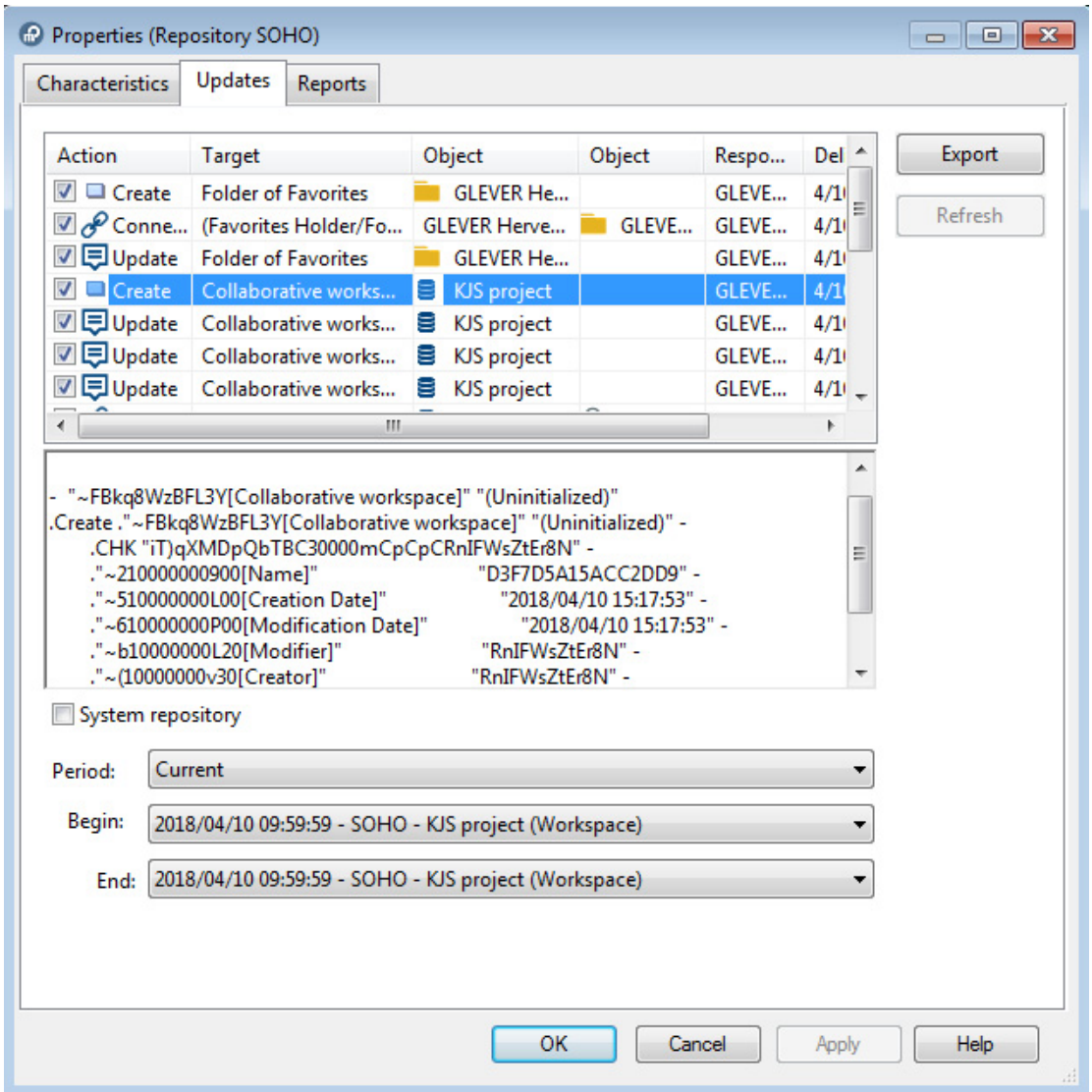
By default, the updates displayed are those you have performed:

- in your current private workspace
- on the **HOPEX** repository

You can also display the updates dispatched on the current repository within the interval defined by two dispatches.

The Updates window

– *To display the Updates window, see Displaying your current updates on the HOPEX repository, or Displaying your current updates on the system repository.*



– *To sort the updates, click the column header.*

Indicated for each action are:

- the **Action** type performed

  Example: "Create", "Connect", "Update".

  - *See Command File Syntax for more information on operators.*
- the type of the object concerned (**Target**)
- the name of the **Object** concerned
- the name of the second **Object** concerned in the case of a "Connect", "Disconnect" or "Change" action
- the person **Responsible** for the action
- the **Delivery date** of the action

  Format: <D/MM/YYYY h:mm:ss AM/PM>

  If you have not dispatched your work yet, the date indicated is the execution date of the action.
- The name of the **Dispatch** that contains the action, in the following format:

  <YYYY/MM/DD hh:mm:ss> <repository> <responsible>

  With:
  - <YYYY/MM/DD hh:mm:ss>: workspace date and creation time
  - <Repository>: name of the current repository
  - <responsible>: name of the person or of the collaborative workspace responsible for the action

  If you have not dispatched your work yet:

  <your name> (workspace)

  <name of your collaborative workspace> (collaborative workspace)
- (when you select the line of a command), the complete text of the update is displayed in the window lower pane.

  - *If needed, to copy the command and paste it in a text file for example, roll the mouse over the whole text, then press <CTRL> + <C> and paste the text in a text file.*

Displaying your current updates on the HOPEX repository

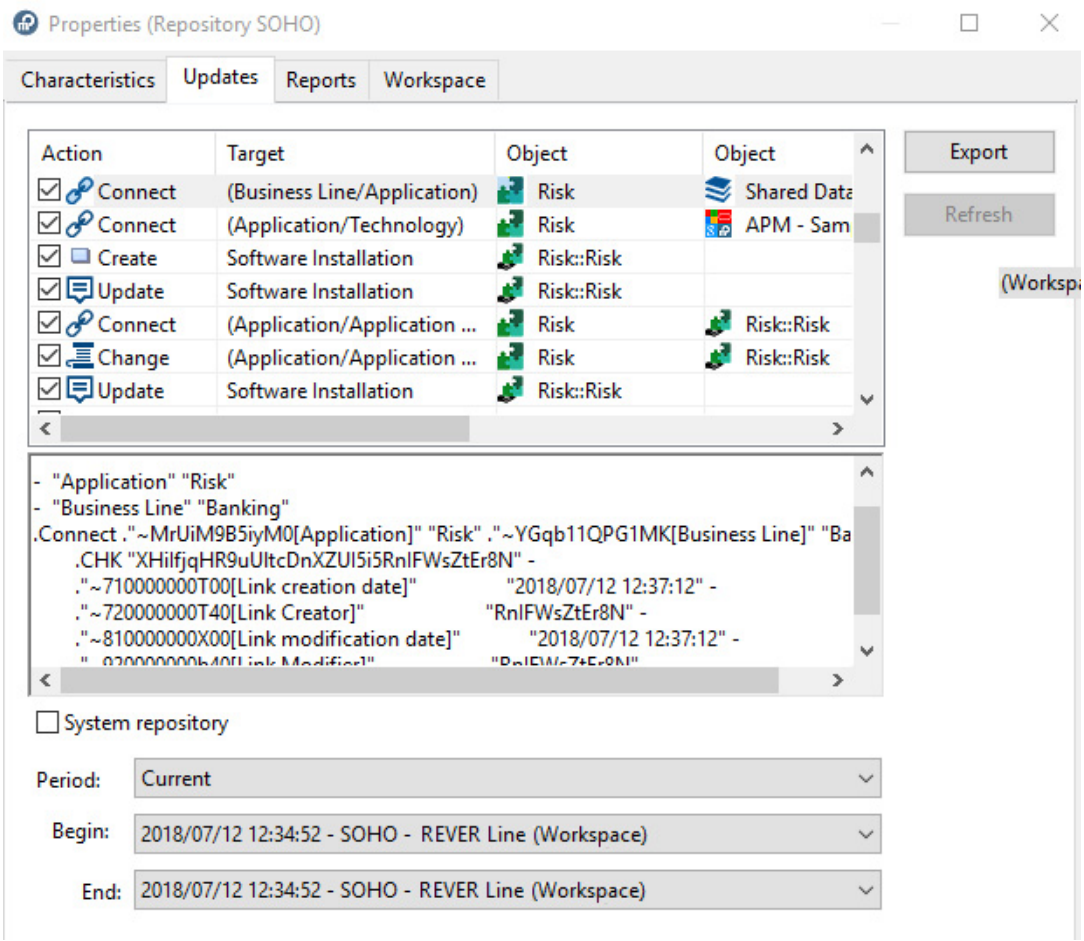
To view the updates you have made in your current workspace:

1. In the **HOPEX** menu bar, select **File > Properties**.

   - *This process may take some time if there are many updates.*

   The property window of the repository in which you are working appears.

2. Select the **Updates** tab.
   By default, the **Updates** tab shows in chronological order the actions you have performed in your private workspace on the **HOPEX** repository



– *Actions you have performed in your private workspace on the system repository are not shown. To display the updates performed on the system repository, see Displaying your current updates on the system repository.*

Displaying your current updates on the system repository

By default, the **Updates** tab shows the actions you have performed in your workspace on the **HOPEX** repository.

– *The display of updates is exclusive (**HOPEX** repository or system repository).*

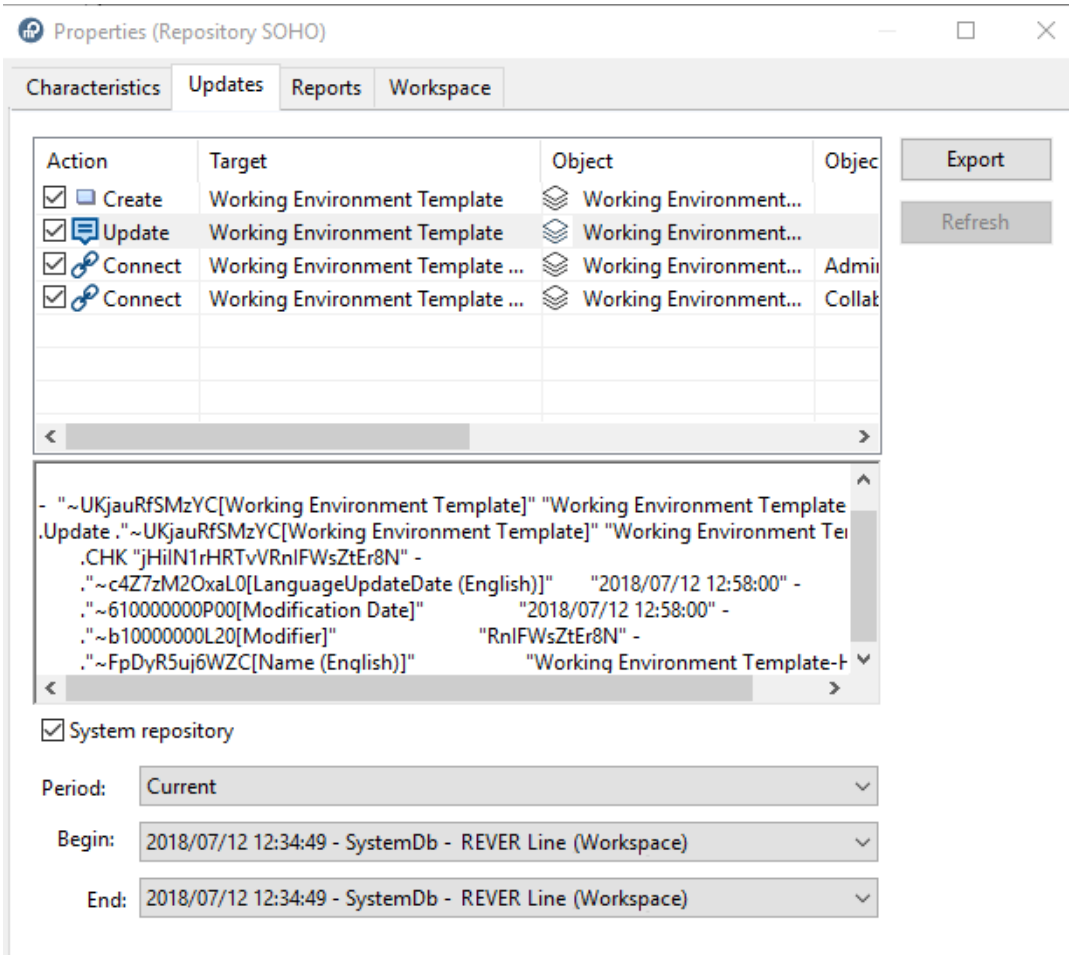
To display your current updates on the system repository:

1. In the **HOPEX** menu bar, select **File > Properties**.

   – *This process may take some time if there are many updates.*

   The property window of the repository in which you are working appears.
2. Select the **Updates** tab.

3. Select the **System repository** check box.
The **Updates** tab display in chronological order the updates performed on the objects of the system repository.



Viewing updates dispatched on the repository over a period of time

By default, the **Updates** tab shows the actions you have performed in your workspace on the **HOPEX** repository. You can display the updates made over a period defined by two dispatches.

To view the updates made over a period of time:

1. In the **HOPEX** menu bar, select **File > Properties**.
   - *This process may take some time if there are many updates.*

   The property window of the repository in which you are working appears.
2. Select the **Updates** tab.
3. Select (/Clear) **System Repository** to display the system repository (/ **HOPEX** repository) updates.

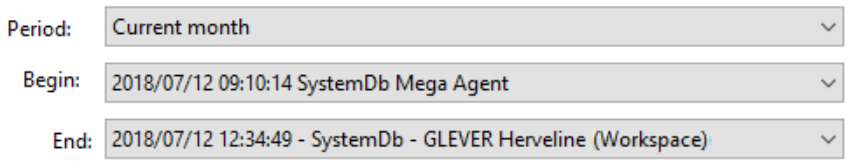
4. In the **Period** field, select the period you are interested in.

   E.g.: Today, Current week, Current month, From the beginning.

   The selected period defines the list of dispatches available in the **Begin** drop-down list.
5. In the **Begin** field, select the dispatch from which you want to display the updates.

   - *The first item corresponds to the first dispatch in the repository.*
6. In the **End** field, select the dispatch until which you want to display the updates.

   - *You can select your current workspace, your updates are included in the display.*



7. Click **Refresh**.
   The selected repository log appears as a list of actions displayed in chronological order.

Viewing the Dispatches and their Content

- *See also Viewing the Repository Update Log.*

The **Repository Dispatches** folder contains private workspaces dispatched in the current repository and in the system repository. Dispatches are listed by day, week and month.

To display private workspaces dispatched and the content of their updates:

1. From **HOPEX** (Windows Front-End), in the **Repository Activity > Repository Dispatch** navigation window, expand the repository file concerned:
   - <Current **HOPEX** repository name> or
   - System repository (**SystemDb**)
2. Right-click the dispatch concerned and select **Properties**.

   - *To access the content of the **Repository Activity** navigation window, you must have **Expert** metamodel access (see Configuring the Metamodel Access).*

   Private workspaces dispatched on the current repository and system repository are contained in the repository folder.
3. Select the **Updates** tab.

Exporting Updates

In the repository update log window, the **Export** button enables export of selected commands in MGL or XMG format.

– *To export your private workspace log, see Exporting Your Private Workspace Log.*

To export updates:

1. Access the repository update log.
2. Select the updates to be exported.
3. Click **Export**.
4. Select the export format:
   - *.mgl: text format
   - *.xmg: MEGA XML format.
5. Click **Export**.
   The file is exported and saved in the specified folder.
   The **Execution Report** appears.
6. (Optional) Click **Open result file** to view the file.
7. Click **OK** to close the window.

Exporting Your Private Workspace Log

You can create an export file (*private workspace log*) and save it in your *work folder*.

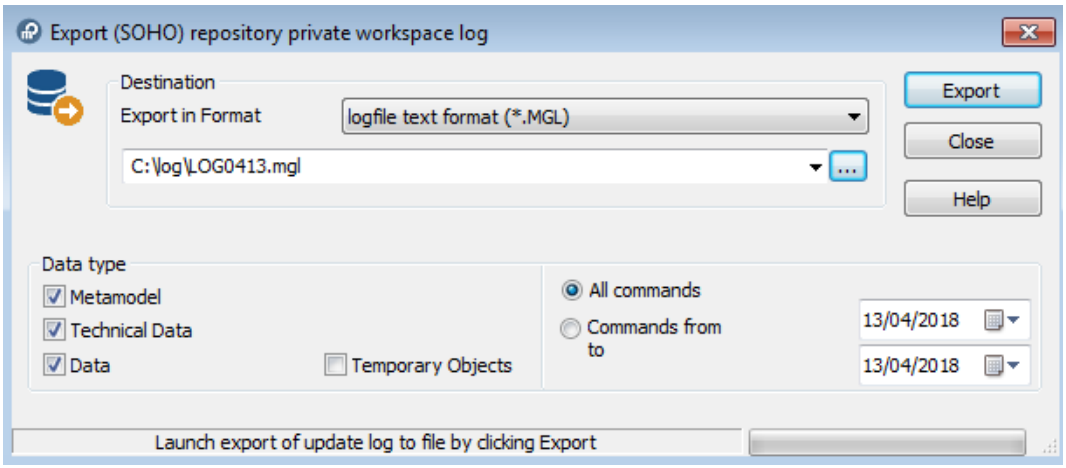
The export file can be exported in format:

- **logfile text** (.mgl).
  Name format of the exported file is "LOGmmdd.mgl", where "mmdd" represents logfile export date month and day.
- **XML MEGA** (.xmg)
  The exported file is in the form of an XML file containing commands or data (objects and links).

To export the work done in the current private workspace in the form of a command file:

1. From the **HOPEX** menu bar, select **Tools > Manage > Export Updates Log**.
   The **Export Repository Private Workspace Log** window opens.
2. Select export format.

3. (Optional) If necessary, modify the data export file name and save folder proposed as default. The **Browse** button ... allows you to browse the folder tree and select the folder in which the file will be saved.



4. In the **Data Type** frame, select the type of modifications to be exported:
   - **Metamodel** if you want to extract the metamodel from the system repository. This is useful if the standard metamodel has been modified.
   - **Technical Data** if you want to include in your file the changes made to data such as descriptors and queries.
   - **Data** if you only want to export changes made to repository data, particularly the workspace.
   - **Temporary Objects** - these objects are created when you execute requests, when you consult objects (stored in the history), etc. Generally you will not need to export these objects.
5. (Optional) If you want to export only those commands that correspond with a defined period, select **Commands from** / **to** and specify the dates.
6. Click **Export**.
   The file is exported and saved in the specified folder.
   The **Execution Report** appears.
7. (Optional) Click **Open result file** to view the file.
8. Click **OK** to close the window.

The user can subsequently import this logfile, for example into a new private workspace.

Private Workspaces and Repository Size

Private workspace life

A private workspace gives a user a frozen view of a repository.

When the repository is modified by other dispatched private workspaces, this private workspace keeps the view it had when created.

Since the data corresponding to these views is kept in the repository, its size grows, and may become disproportionate to the actual repository contents.

- *See Dispatching Your Work and Refreshing Data.*

Private workspace monitoring

More than one user can connect to a single repository via network share. The first time a user connects to the repository, he/she opens a private workspace. This private workspace ends only when the user dispatches, discards, or refreshes his/her modifications, and not when simply disconnecting from the **HOPEX** repository.

- *See Refreshing Data and Discarding Work.*

Modifications made by the user are saved in a temporary space (data) in his/her private workspace dedicated to the data of his/her private workspace. The repository is updated only when the user dispatches these changes.

- *See Dispatching Your Work.*

All data accessed by a user is "frozen" for the duration of the private workspace.

Example:

> For example, if an object is renamed in the reference repository after the private workspace is opened, the user sees the previous name unless the private workspace is refreshed. However, users connecting after the modification has been dispatched will have a view reflecting the most recent state of the repository.
>
> If other users connected before you dispatch your updates, and are accessing the same data as you, they will have an obsolete view of the data until they refresh their private workspace. Note that when you dispatch your updates, your view is refreshed automatically.

This means that data being accessed by user A and modified at the same time by user B is duplicated. It is stored in its initial state for each user private workspace; if a user makes a change, the previous state is stored along with the new one.

When the updates are dispatched and all users accessing the updated data have refreshed their private workspaces, the previous state is no longer required.

Modifying the maximum duration of a private workspace

By default the maximum duration of a private workspace is 6 days

Once this duration has elapsed, at connection, a message prompts the user to dispatch or refresh his/her private workspace.

To modify the maximum duration of a private workspace:

1. In the environment **Options**, select **Options > Installation > Advanced**.
2. Modify the **Recommended open workspace duration** option value (in day).

MANAGING LOCKS

When several users are connected to the same repository simultaneously, there may be conflicts when they modify the same object in their different private workspaces.

See:

- Principle of Locks
- Managing Locks on Objects

Principle of Locks

With the network version, concurrent accesses to objects can be checked using *locks*.

Preventing conflicts

As soon as a user modifies an object, a lock is placed on this object. Another user can thus only view this object. This user cannot access a new update until the first user dispatches his/her private workspace, and he/she refreshes his/her private workspace. This prevents conflicts between the state of the object in the repository and the obsolete view of the second user.

Deleting a lock or unlocking an object

Lock management is automatic. You do not normally need to delete locks or unlock objects.

The few exceptions are due to abnormal operations, for example when a private workspace has been deleted from the private Workspace management window, or at desynchronization of clocks.

- *For more details on clock synchronization, see Clock synchronization.*

When a lock is deleted, another user can modify the object without dispatching or refreshing his/her private workspace.

- A user can delete locks placed on his/her private workspace since its creation.

When a user dispatches his/her work, the lock (his/she had placed on an object) is unlocked but not deleted. The lock is deleted only when all other private workspaces, which were created before the lock was unlocked, are closed. A private workspace is closed when it is dispatched, discarded, or refreshed.

Clock synchronization

When lock management is active, workstation and environment clocks must be synchronized.

P **If clock times differ by more than 5 minutes when you connect to HOPEX, an error message appears and your work is saved at your workstation time. This situation can create consistency problems when dispatching your work.**

To be able to start **HOPEX**, you must synchronize the clocks:

Times relating to the network object locking system are always expressed in GMT0. For network workstations on which executables are decentralized, ensure that clocks are synchronized with a reference server. To do this, it is possible to use the "Net-Time" command or a Web clock synchronization service.

- With Windows NT and 2000, this can be done using a network command such as “Net time \\Workstation name/set” with LAN Manager or Windows NT.
- With Windows XP, setting is carried out from the Windows **Control Panel**, **Date/Time** icon.

– *For more information on clock synchronization, see Windows online Help.*

Details on the operating method of the locks

HOPEX only indicates that objects are locked when their attributes are modified (unlike links for example).

Warning on unlocking

If you attempted to use an object that was locked, a message alerts you as soon as the object is freed for you to use again.

Diagrams

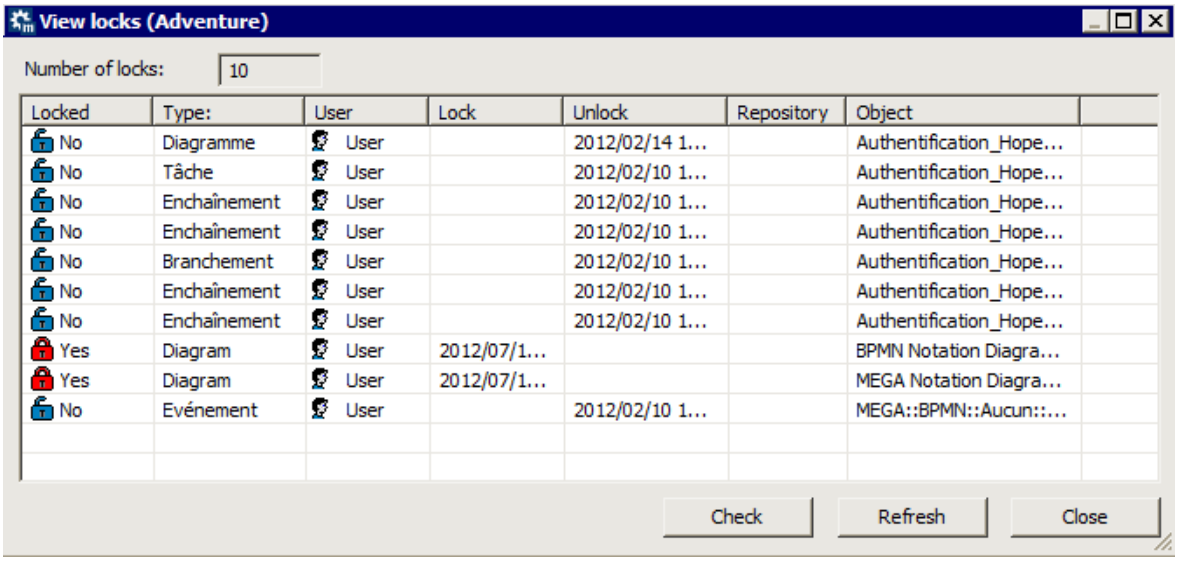
There are two types of locking applied to diagrams

- ***The diagram has simply been viewed and not modified***: as soon as the first user closes the diagram it can be opened by a second user.
- ***The diagram has been modified***: as for classical locking, the second user must wait until the diagram has been dispatched by the first user and therefore unlocked.

Managing Locks on Objects

The lock management window can be accessed from:

- **HOPEX Administration**
  - *Managing locks in HOPEX Administration*
- **HOPEX**
  - *Managing locks in HOPEX*



The **Display Locks** window displays the locks created since the creation of the oldest private workspace. The following information is provided for each lock:

- the lock state (**Locked**):
  - **Yes**: When a user locks an object, other users can only view it, even if they dispatch their work.
  - **No**: the object is locked when its modifications are dispatched. Other users must dispatch their work or refresh their private workspaces before they can modify the object.
    - *To unlock an object or delete a lock, see Managing locks in HOPEX Administration*
- the name of the **Object** concerned
- the **Type** of object concerned
- the **User** who owns the lock
- the date and time (GMT0) of the **Lock**, and, if applicable, **Unlock**.
- the name of the **Repository** to which the lock relates.

Managing locks in HOPEX Administration

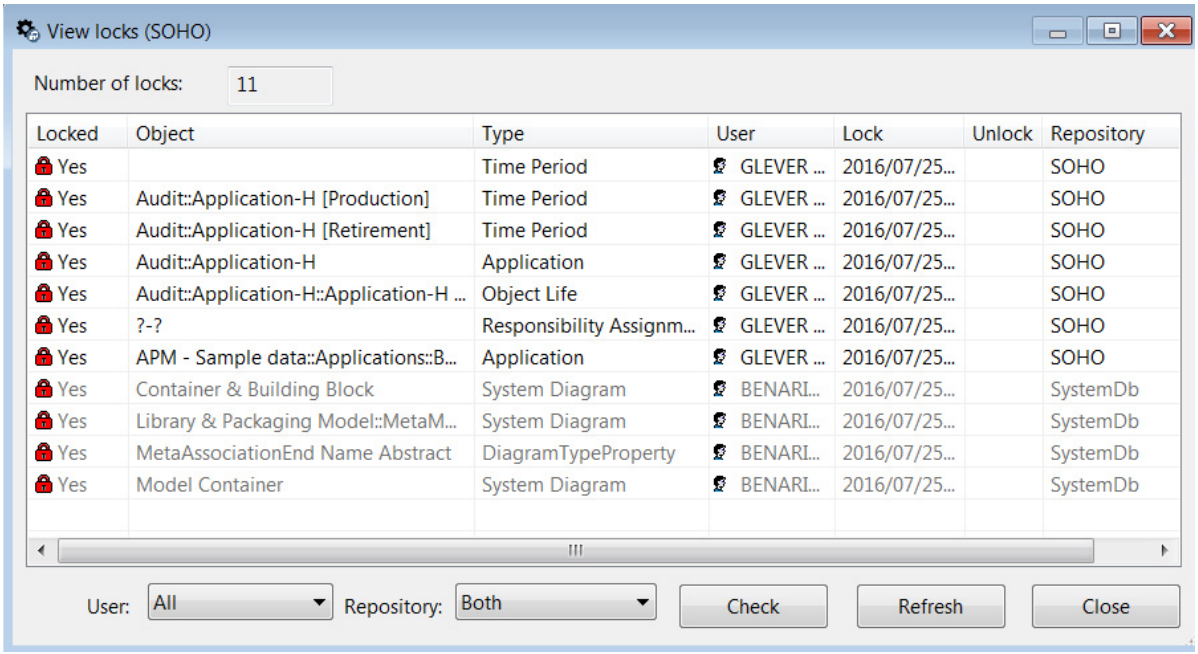
To manage locks from the **Administration** application:

1. Connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **Repositories** folder, then that of the repository concerned.
3. Right-click the **Locks** folder and select **Manage**.
   The **View locks** dialog box opens.
4. To:
   - unlock an object, right-click the lock concerned and select **Unlock**.
   - delete an object, right-click the lock concerned and select **Delete**.

Managing locks in HOPEX

To view the locks from **HOPEX**:

1. In the **HOPEX** menu bar, select **Tools > Manage > View Object Locks**.
   The **View locks** dialog box opens.



The locks appear grayed, except that of the current user.

2. To:
   - delete an object, right-click the lock concerned and select **Delete**.
   - unlock an object, right-click the lock concerned and select **Unlock**.

To:

- sort locks as a function of a column, click the column header.

  M *You can also arrange columns in the order you want. To do this, click the header of the column to be moved and, holding down the mouse button, move it to the required position.*

- refresh lock display, click **Refresh**.
- sort the display according to **User**, select:
  - **Current** to view your locks only.
  - **Other users** to view locks of other users only.
  - **All** to view your locks and those of other users.
- sort the display according to **Base**, select:
  - **Current Repository** enables display of locks on the current repository.
  - **System Repository** enables display of locks on the system repository.
  - **Both** enables display of locks on current and system repositories.

MANAGING ENVIRONMENTS

When multiple users are working together across a network, it may be useful to create several work environments. Another reason for creating a new environment is to provide the administrator with an environment where report templates (MS Word), queries, etc. can be modified and tested without interfering with users' work.

Environment management functions are used when several environments are available and they need to exchange data or repositories.

Les points suivants sont abordés ici :

- Using Environments
- Deactivating an Environment (RDBMS)
- Customizing Environments

USING ENVIRONMENTS

Environment Structure

) *An environment groups a set of users, the repositories on which they can work, and the system repository. It is where user private workspaces, users, system data, etc. are managed.*

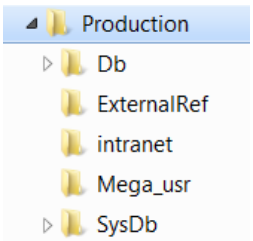
An *environment* includes:

- a configuration file
- a system repository
  - *See System Repository (SystemDb).*
- shapes, if they are different from the site standard shapes
- a "Db" folder for the repositories used in this environment.

In most cases, there is only one environment per site. It is possible to define several environments when the *site* includes repositories on which users work with different resources (such as specific report templates (MS Word) or metamodel extensions).

- *All users must have full access rights to the resources of the environments and repositories (creation and deletion of files and folders).*

The folder structure of an environment is as follows:



- ***Db***: contains the environment repositories
  - *Each repository is located in a folder carrying its name. See Creating a Repository.*
- ***ExternalRef***: contains external references of the environment
- ***intranet***: contains the Web sites generated in the environment
- ***Mega_usr***: contains shapes specific to the environment
- ***SysDb***: the system repository

Creating an Environment

To create an RDBMS environment, see **Installation and deployment > RDBMS Repository Installation Guide** documentation.

Moving and Referencing an Environment

During administration of **HOPEX**, you may need to move an environment. When an environment is moved, you must create a new reference for it to be able to use it.

When you delete an environment, it is recommended that you delete its reference. It will then no longer appear in the list of environments available in the corresponding **HOPEX** site.

For more details on these points, see:

- Moving an environment
- Referencing an environment
- Deleting a reference to an environment

Moving an environment

When you need to move an environment, such as when you have to place it on a different drive:

1. Copy the root folder of the environment in its new location.
2. Delete the reference for the old environment.
3. Create a reference for it at its new location.
4. Carry out the same operations for each of the repositories not hosted in the environment structure.
5. Check that the reports (MS Word) and Web sites of each repository point to the correct files.

Precautions to be taken:

- Verify that there is enough free space in the destination folder.
- It is not necessary to dispatch private workspaces before moving the environment. They will be moved with the environment.
- No user should be connected during movement of environments. No private workspace should be active.

Referencing an environment

When an environment has been moved by the user, it is necessary to create a reference for this environment so the site will recognize it.

To create a reference to an environment:

1. Connect to **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. In the navigation tree, right-click **Environments** and select **Create Reference**.
3. In the dialog box that opens, select the folder in which the environment to be referenced is located.
4. Click **OK** to validate.
   The new environment reference is created and appears in the list of available environments.

Deleting a reference to an environment

To delete a reference to an environment:

1. Connect to **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. In the navigation tree, right-click the desired environment and select **Delete Reference**.
   A confirmation request appears.
3. Select **Yes**.
   The environment reference is deleted and no longer appears in the list of available environments.

Deactivating an Environment (RDBMS)

In an RDBMS environment, for administration requirements, you can prevent users from connecting to **HOPEX**.

You can block all users or Web users only.

The users already connected are not disconnected and can continue to work.

To deactivate an environment:

1. Connect to **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. In the navigation tree, right-click the environment (RDBMS) that you want to make inactive and select **Environment activity** then:
   - **Inactive environment**: only the **Administration (Windows Front-End)** application is accessible. Users cannot connect to **HOPEX** (Windows Front-End and Web Front-End).
   - **Active environment with Windows-Front-End only**: access to **HOPEX (Web Front-End)** is blocked.

CUSTOMIZING ENVIRONMENTS

Administration tasks may involve you in modifying configuration of an environment so that it meets your particular requirements.

Avec **HOPEX Administration** vous pouvez :

- Backing Up Environment Customizations
- Restoring Environment Customizations
- Compiling an Environment

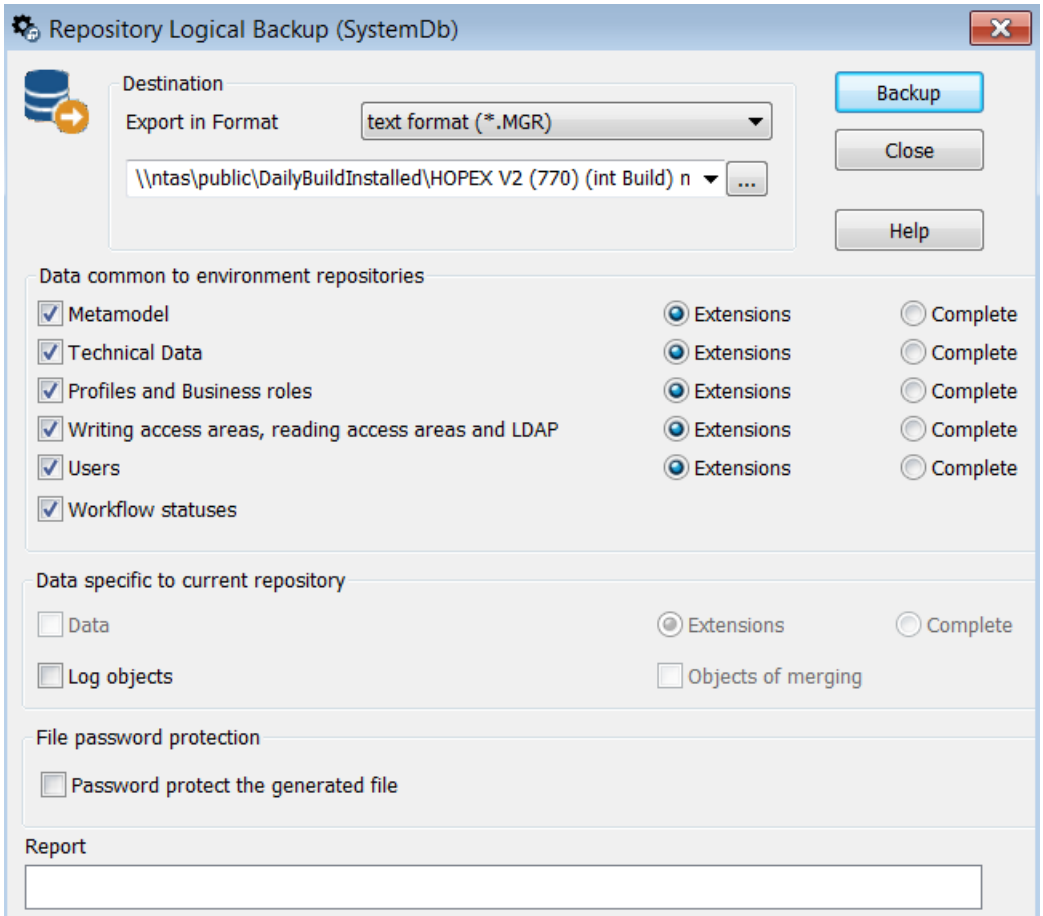
Backing Up Environment Customizations

The following operations allow you to back up modifications made to the standard environment so that these can be imported into a new environment. This backup contains modifications you have made to *report templates (MS Word)*, Web site templates, *descriptors*, *queries* etc. It also contains the list of users and their configuration if this exists. You can also save any extensions to the metamodel.

To back up customizations:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*

2. In the **Repositories** folder, right-click the system repository (SystemDb) and select **Logical Backup**.
   The **Repository Logical Backup** dialog box opens.



3. In the **Destination** pane, select the export format of the backup file:
   - **text** (.MGR)
   - **XML MEGA** (.XMG)

     – *This format cannot be used to extract the metamodel or technical data. This format is reserved for exchange of data between **HOPEX** and other applications. It includes commands or data (objects and links).*

4. (Optional) In the **Destination** frame, click **Browse** ... to browse the folder tree and modify the name and/or location of the backup file.

   – *By default, the backup is saved in the "SystemDb.mgr" file in the \SysDb\WORK\ folder of the repository.*

5. Select the type of data you want to save.

   - *The **Extensions** button corresponds to data created by users.*

   P **Carry out a complete backup only if technical support asks you to do so.**

   M *It is recommended that you save the extensions of the metamodel and technical data in different files.*

   In the **Data common to environment repositories** pane, select the data type of the system repository to be saved:

   - **Metamodel** if you want to extract the metamodel from the system repository. This is useful if the standard metamodel has been modified.
   - **Technical Data** allows extraction from the system repository of data such as *descriptors*, *queries*, and *report templates (MS Word)*.
   - **Profiles and Business roles** allows extraction of created profiles and business roles (those not provided by MEGA).
   - **Writing access areas, reading access areas and LDAP** allows extraction of created writing and reading access areas (those not provided by MEGA) and LDAP parameters (parameters, servers, groups).
   - **Users** allows extraction of created users (persons, person groups, logins).
   - **Workflow statuses** enables extraction of workflows (workflow instances, transitions and statuses, tasks, validations, requests for change).

   In the **Data specific to current repository** pane:

   - **Log objects** allows you to also save object logs.

     - *For more information on histories, see Deleting a Repository and Viewing Object History.*

6. (If needed) In the **File password protection** pane, select **Password protect the generated file**.

   - *The password is asked at file import.*

7. Click **Backup** to start backup.
   A series of messages keeps you informed of backup progress.
   The file is saved in the selected format (*.mgr or *.xmg) or in *.mgz format if you have selected the file protection option.

Restoring Environment Customizations

You can restore environment customizations that you have backed up.

- *See Backing Up Environment Customizations.*

To restore environment customizations:

1. From **HOPEX Administration**, connect to the environment concerned.

   - *See Connecting to an Environment.*

2. Under the **Repositories** folder, right-click the repository concerned and select **Object Management > Import**.

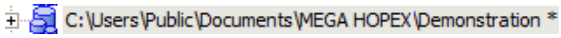
   - *See Updating a Repository.*

Compiling an Environment

The metamodel and technical data must be compiled after migration or customization. This is to check configuration of the environment concerned. When compilation has been completed, processing for all users of this environment is speeded up.

HOPEX can operate in "interpreted" (not compiled) mode but with reduced performance.

In the **HOPEX Administration** navigation tree, an asterisk after the environment name indicates that the metamodel or/and technical data (excluding permissions) of this environment is/are in "interpreted" (not compiled) mode.



Metamodel compilation includes in parallel translation in the current language. You can also translate the metamodel into another language.

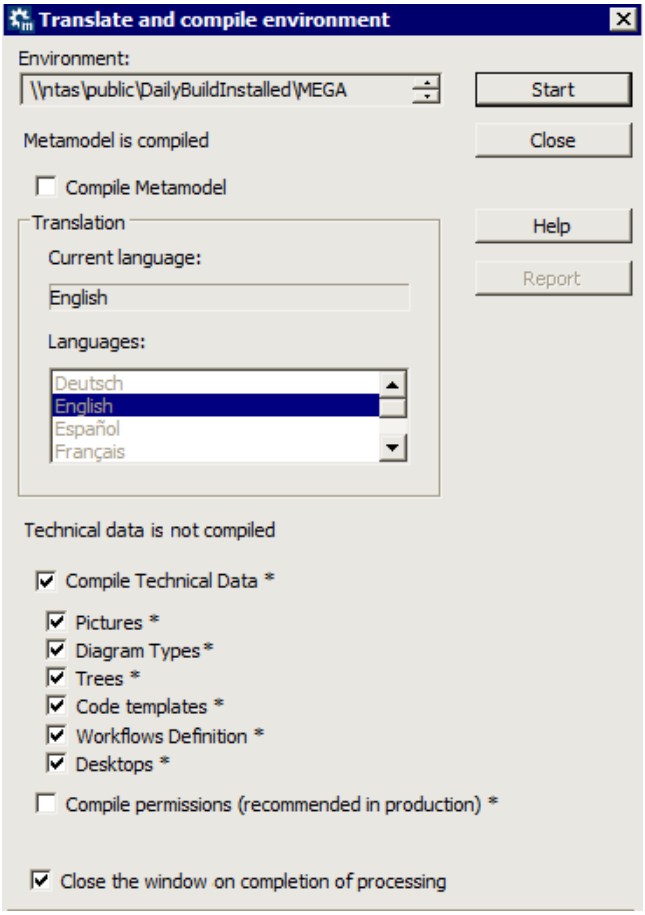
Prerequisite:

1. Ask **HOPEX** users to exit their application:
   - mandatory for **Web Front-End** users
   - recommended for **Windows Front-End** users.
2. Start the **HOPEXMega Server Supervisor** tool.
   - *See Starting HOPEX Server Supervisor.*
3. In the identification area of your workstation, right-click **HOPEX Server Supervisor** and select **System > Stop HOPEX Processes Services and Web Application**.

To translate and compile the metamodel and/or compile technical data:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*

2. In the navigation tree, right-click the desired environment and select **Metamodel > Translate and Compile**.
   The **Translate and compile environment** dialog box opens.



   * : the asterisk indicates interpreted (not compiled) mode.

   In the **Translation** frame, the **Current Language** field indicates the current language of the system repository.

3. If the metamodel is not compiled, keep **Compile Metamodel*** selected.
4. (If **Compile Metamodel*** is selected) In the **Translation** frame, in the **Languages** list of the system repository, select the target translation language.

   ```
   Example: "English"
   ```

5. If technical data is not compiled, keep **Compile Technical Data*** selected.

   P **If technical data and metamodel are not both compiled, you must also keep Compile Metamodel* selected.**

   By default, all technical data (images, diagram types, trees, code templates, workflow definitions, desktops) is selected.

6. If you are in a production environment, keep the **Compile Permissions*** option selected; otherwise, you can clear the selection. This compilation improves **HOPEX** loading times.

   – *Compilation of permissions can take some time (more than an hour) and is recommended only in a production environment. The fact of not compiling permissions (permissions interpreted mode) has no impact on correct operation of **HOPEX**.*

7. Keep **Close the window on completion of processing** option selected.

   – *This option enables automatic closing of the **Translate and Compile Environment** window when compilation is completed, allowing **HOPEX (Windows Front-End)** users to resume their work.*

8. Click **Start** to run compilation and translation.

   – *If **HOPEX (Windows Front-End)** users have remained connected, they are blocked during processing.*

   Metamodel and/or technical data compilation (excluding permissions) takes several minutes.

   If you selected metamodel compilation with a different target language, after execution the system repository is available in the new language.

9. When compilation processing is completed, **HOPEX** (Windows Front-End) users can resume their work.

   – *If at step 7 the **Close the window on completion of processing** option was not selected, click **Close** to close the **Translate and Compile Environment** window and allow **HOPEX (Windows Front-End)** users to resume their work.*

10. In the identification area of your workstation, right-click **HOPEX Server Supervisor** and select **System > Restart HOPEX Processes Services and Web Application**.

    – ***HOPEX (Web Front-End)** users can connect to Web applications.*

MANAGING THE SCHEDULING (SCHEDULER)

The **HOPEX Scheduler** feature enables to create and schedule jobs.

– *For detailed information on the scheduler, see the **HOPEX Power Studio - Scheduler** technical article.*

The following points are covered here:

- Introduction to the Scheduler
- Managing Triggers
- Configuring the Trigger Scheduling
- Managing the Scheduler

INTRODUCTION TO THE SCHEDULER

Concepts

The Scheduler feature enables to perform tasks defined by MEGA or by an HOPEX Administrator at defined dates, times, and frequencies so as to avoid overloading **HOPEX** at user working hours.

Job

A job is a process. It includes:

- a macro to be executed
- a context, which gives the information required to execute the macro: **Job Context** as a character string.

Scheduler

The Scheduler enables to schedule job execution:

- execution date and time
- frequency

Trigger

A Trigger is associated with a job to define the job execution date:

- the Trigger is based on a Trigger Definition. This definition consists of a job which includes the macro that the Trigger will execute.
- the Scheduler enables to define when (date and time) to execute the job and at which frequency.

Option: Time Zone

In the Scheduler, by default the time format is hh:mm:ss (UTC). To facilitate the configuration, you can change this UTC format for a (user or server) local time format.

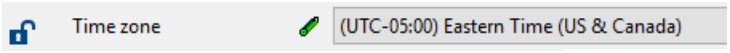
– *See Defining the Execution Time Zone.*

To define your user local time:

1. Access the site (or environment) level options.
2. Expand the **Installation** folder and select **Web Application**.

3. In the right pane, use the drop-down menu of the **Time zone** option to select your time zone.

   ```
   E.g.: select "(UTC-05:00) Eastern Time (US & Canada)" to
   configure times in New-York local time.
   ```

   When you configure your Triggers, their execution scheduling is defined in this time zone if you choose **User time zone** as execution time zone.
   If you configure your Triggers in the **UTC** or **Server time zone**, you can check their conversion in this local time zone if needed.

   - *For example, see Defining the Execution Time.*

MANAGING TRIGGERS

See:

- Accessing Triggers
- Creating a Trigger
- Managing a Trigger
- Defining the Trigger Execution Context

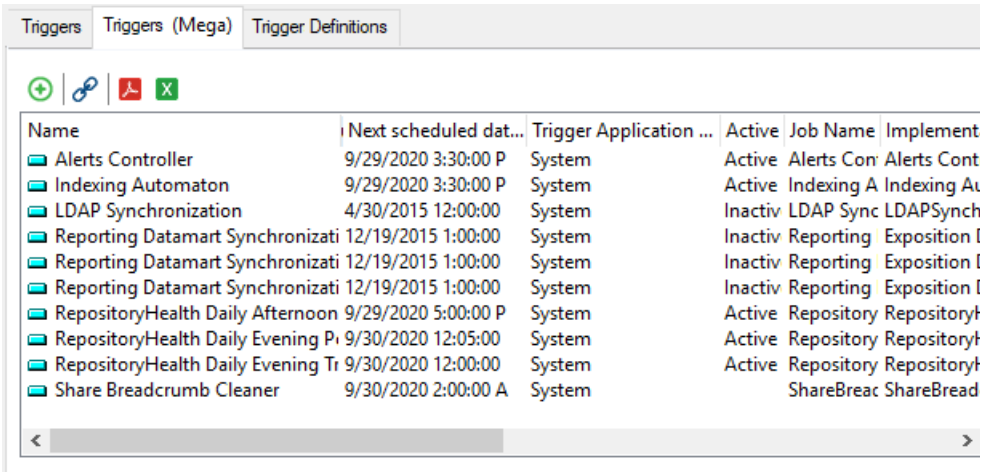
Accessing Triggers

Accessing the Trigger management

In the **Administration** application (Administration.exe), the Trigger Management window displays the following tabs:

- **Triggers**
  Triggers defined by a HOPEX administrator. These Triggers can be defined on a data repository or on the System repository.
- **Triggers (Mega)**
  Triggers provided with HOPEX and available in all the installations. These Triggers are defined on the System repository.
- **Trigger Definitions**
  Predefined Triggers available for you to create your Triggers.

  Example: "Computation of OnDemand and expired OnUpdate MetaAttributes"



For each scheduled Trigger, the list indicates in particular:

- its name
- the date and time of its next execution
- its status (active or inactive)
- the name of the executed job
- the name of the job implementing macro

To access the Trigger management:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. In the **Repositories** folder, expand the repository concerned.
3. Right-click the **Scheduler** folder and select **Manage Triggers**.

Creating a Trigger

A Trigger is based on a Trigger Definition. This definition consists of a job which includes the macro that the Trigger will execute.

Trigger Definitions are available in the **Trigger Definitions** tab.

To create a Trigger:

1. Access the Trigger management window.
2. In the **Triggers** tab, click **New** .
   - *If you create a Trigger from the **Triggers (Mega)** tab, this Trigger is automatically moved to the **Triggers** tab.*
3. Select the **Trigger Definition**.
4. Click **Next**.
   The job definition window opens.
5. Enter the **Name** of the job.
6. In the **Job Context** pane, define the job execution context.
   - *See Defining the Trigger Execution Context.*
7. Click **Finish**.
   The Trigger is created.
   By default, the Trigger is active.
   You can execute the Trigger to test it.
   - *See Managing a Trigger.*

Managing a Trigger

You can:

- update the Trigger scheduling
  To modify the job execution dates, times, and frequencies.
  - *See Configuring the Trigger Scheduling.*
- activate/deactivate a Trigger
  By default, a Trigger is active.
  To temporarily suspend the job execution, you can temporarily deactivate its Trigger.
- execute a Trigger
  To immediately execute the job associated with the Trigger (outside its scheduling).

  ```
  For example, to test a job.
  ```
- delete a Trigger
  If you want to reuse the Trigger later, instead of deleting the Trigger you can deactivate it.
- display the Trigger properties
  - The **Scheduling** tab details the scheduling definition and lists all the next executions of the trigger
  - The **System Job** tab details the job executed by the Trigger (especially the macro and execution context).

To manage a Trigger:

1. Access the Trigger management.
   - *See Accessing the Trigger management.*
2. Right-click the Trigger concerned and select:
   - **Update Scheduling**
     - *See Configuring the Trigger Scheduling.*
   - **Activate**/**Deactivate**
   - **Execute**
   - **Delete**
   - **Properties**
     - *See Defining the Trigger Execution Context.*

Defining the Trigger Execution Context

A Trigger is triggered on the objects defined in the associated job macro.

To define on which MetaClass the Trigger applies:

1. Access the Trigger management.
   - *See Accessing the Trigger management.*
2. Right-click the Trigger concerned and select **Properties**.
3. Select the **System Job** tab.

4. In the **Context** pane define the Trigger execution context, i.e. the objects on which the job applies.

   P **Attention: do not add any break line in the character string.**

CONFIGURING THE TRIGGER SCHEDULING

To configure the Trigger scheduling, you must define:

- its execution time zone for all its scheduling time definitions
  - *See Defining the Execution Time Zone.*
- the date and time of its first execution
  - *In a recurrence case, the first execution date is not mandatory.*
  - *See Defining the First Execution Date of the Trigger.*
- its frequency
  The execution can be unique or recurrent.
  - *See Defining the Trigger Frequency.*

  If the execution is recurrent:
  - its execution time.
    - *See Defining the Trigger execution time.*
  - if needed, you can define a recurrence on the execution time, i.e. execute the Trigger several times the scheduled day.
    - *See Defining a time-based recurrence on the Trigger execution.*
  - the date of its last execution (defined or with no end)
    - *See Defining the Last Execution Date.*

Defining the Execution Time Zone

To facilitate scheduling time definition, you can modify the time zone in which you define the scheduling times:

- **UTC** (default), to define times in UTC format
- **User time zone** to define times in the user time zone
- **Server time zone** (STZ), to define times in the time zone of the server executing the Trigger

Attention: if you change the time zone a posteriori, times are not automatically adapted.

To define the execution time zone:

1. Access the Triggers.
   - *See Accessing Triggers.*
2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Time zone for all the scheduling time definitions**, select the time zone.
4. If you select **User time zone**, you must define your time zone.
   - *Voir Option: Time Zone.*

Defining the First Execution Date of the Trigger

You must define the first execution date of the Trigger. It can be:

- absolute

  E.g.: on the 04/18/2020 at 18:30:15.

- relative (relative to a reference date)

  - *In a recurrence case, the first execution date is not mandatory.*
  - *In a non recurrence case, the first execution date is the unique execution date.*

Defining the first execution date (or unique execution)

To define the first execution date of a Trigger:

1. Access the Triggers.
   - *See Accessing Triggers.*
2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Start** section:
   - Use the calendar of the **Start date (absolute)** field to select the first execution date of the Trigger.

     Select **Today**, if you want to define the current day.
   - In the **Start time** field, set the triggering time of the Trigger.

     By default the time is set to 00:00:00 (hh:mm:ss format) in the time zone defined (see Defining the Execution Time Zone).

     - *If you are in the UTC time zone, to facilitate the check of your settings, see Option: Time Zone.*

Defining a relative date for the first execution

You can define a relative date for the first execution, i.e. define the first execution date as:

- (by default) immediately after the Trigger creation, or
- at a later date:
  - a specific number of days after the reference date
  - a specific day of the week after the reference date
  - a specific day of the moth after the reference date

To define the first execution date of a Trigger:

1. Access the Triggers.
   - *See Accessing Triggers.*
2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Start** section, select **Relative Date**.
   By default **Reference date/time as soon as possible** is selected: the execution is triggered right after the Trigger creation.

4. To configure a later relative date, clear **Reference date/time as soon as possible**, then in the **Start date (relative)** click ··· and define:
   The **Day of relative date**:
   - In the **Days from reference**, enter the number of days after the reference date, or
   - Select the **Day of week** and use the drop-down list to select the chosen day, or

     `E.g.: Tuesday, the Trigger is executed on the first Tuesday following the reference date.`
   - Select the **Day of month** and use the drop-down list to select the day.

     `E.g.: 15th, the Trigger is executed on the 15th of the month following the reference date.`

   The **Month of relative date**:
   - In the **Months from reference**, enter the number of months after the reference date, or

     `E.g.: 2, the Trigger is executed a couple of months after the reference date.`
   - Select the **Month of year** and use the drop-down list to select the chosen month, or

     `E.g.: June, the Trigger is executed in June following the reference date.`

Defining the Trigger Frequency

A Trigger can be executed uniquely or on a regular basis.

Whatever the frequency chosen, you can perform a first execution as defined in the **Start** section.

Frequency:

- ***daily***
  By default, the Trigger is executed every day at the time set for the first execution.
  You can execute the Trigger every N days (N to be defined)
- ***weekly***, you must define:
  - the day of the week

    `E.g.: Monday, Tuesday,..., Sunday`

    You can select several days.
  - the frequency

    `E.g.: every two weeks (N=2)`
- ***monthly***, you must define:
  - the day of the month, or the day of the week (day of the week and week of the month to be defined)

    `E.g.: 1,2,...,31, last day of the month`

    You can select several days.

    `E.g.: every Sunday of the last week of the month, i.e. the last Saturday of the month.`

    You can select several days and several weeks.
  - the frequency: every N months (N to be defined) or a specific month every year

    `E.g.: every 2 months (N=2) or in April every year.`

    You can select several months.

To define the Trigger execution frequency:

1. Access the Triggers.
   - *See Accessing Triggers.*
2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Date Recurrence** section, use the drop-down menu of the **Recurrence Type** field to select the frequency.

   `Example: Daily, Monthly, Once, Weekly.`
4. Configure the frequency.
5. (If you want to first execute the Trigger as defined in the **Start** section) Select **Execute at Start date time**.

Defining the Last Execution Date

By default, the Trigger scheduling is endless.

You can define the Trigger last execution date, via:

- an end date, or
- a defined repeat number

To define the Trigger last execution date:

1. Access the Triggers.
   - *See Accessing Triggers.*

2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Recurrence End** section, use the drop-down menu of the **Recurrence End Type** to select the end type.

   Example: End Date or Repeat Number.
4. (If you selected End Date) Define the last execution day and time.
   - Use the calendar of the **End date (absolute)** field to select the last execution date of the Trigger.
   - In the **End time** field, set the triggering time of the Trigger.

     By default the time is set to 00:00:00 (hh:mm:ss format) in the time zone define (see Defining the Execution Time Zone).

     - *If you are in the UTC time zone, to facilitate the check of your settings, see Option: Time Zone.*

Defining the Execution Time

In case the **Recurrence Type** is "Daily", "Weekly", or "Monthly", you must define the Trigger execution time:

- once: you need to define the execution time only
- several times a day, you must define:
  - the scheduling period (in hours):
  - the start time
  - the end time

Times are defined in the defined time zone (see Defining the Execution Time Zone) in hh:mm:ss format.

- *If you are in the UTC time zone, to facilitate the check of your settings, see Option: Time Zone.*

Defining the Trigger execution time

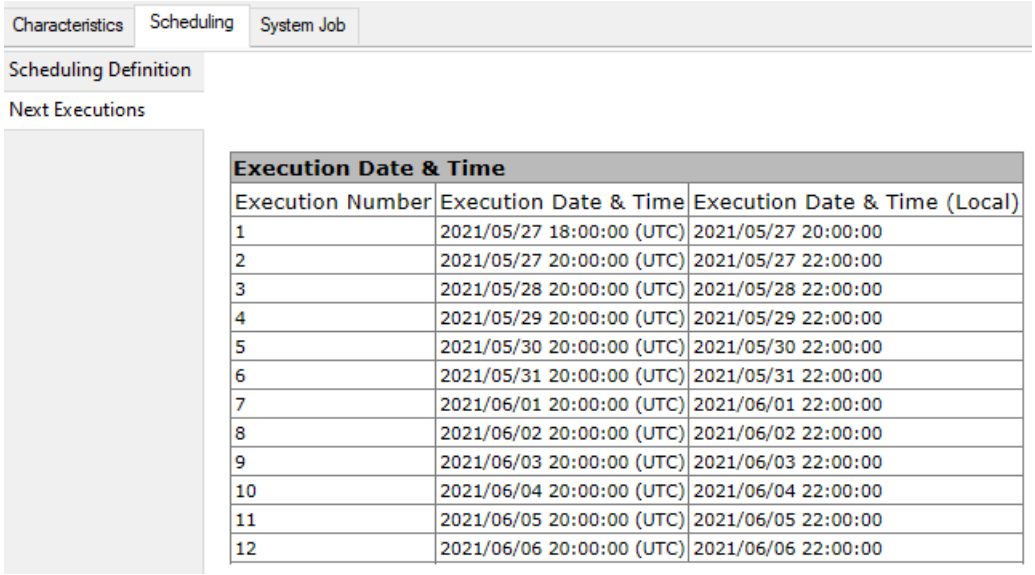
You can define a unique execution time each scheduled execution day.

To set the Trigger execution time:

1. Access the Triggers.
   - *See Accessing Triggers.*
2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Time scheduling (for date recurrence)** section:
   - in the **Scheduling type**, keep "Once".
   - in the **Single trigger time** set the time at which you want to execute the Trigger.

     E.g.: 22:00:00, by default 04:00:00 (in the time zone defined).
4. Click **OK**.

5. Check the scheduling of your execution time.
   - Close and re-open the Trigger Management window.
   - Right-click the Trigger and select **Properties**.
   - In the **Scheduling** tab, select the **Next Executions** sub-tab.



Execution Date & Time		
Execution Number	Execution Date & Time	Execution Date & Time (Local)
1	2021/05/27 18:00:00 (UTC)	2021/05/27 20:00:00
2	2021/05/27 20:00:00 (UTC)	2021/05/27 22:00:00
3	2021/05/28 20:00:00 (UTC)	2021/05/28 22:00:00
4	2021/05/29 20:00:00 (UTC)	2021/05/29 22:00:00
5	2021/05/30 20:00:00 (UTC)	2021/05/30 22:00:00
6	2021/05/31 20:00:00 (UTC)	2021/05/31 22:00:00
7	2021/06/01 20:00:00 (UTC)	2021/06/01 22:00:00
8	2021/06/02 20:00:00 (UTC)	2021/06/02 22:00:00
9	2021/06/03 20:00:00 (UTC)	2021/06/03 22:00:00
10	2021/06/04 20:00:00 (UTC)	2021/06/04 22:00:00
11	2021/06/05 20:00:00 (UTC)	2021/06/05 22:00:00
12	2021/06/06 20:00:00 (UTC)	2021/06/06 22:00:00

The **Execution Date & Time** table indicates the first execution date and time and the following ones with their corresponding local time.

- *To define your local time, see Option: Time Zone.*

E.g.: in this daily scheduling, after the first execution date (here 20:00:00 Paris local time i.e. 18:00:00 UTC), the Trigger is executed once a day at 22:00:00 Paris local time i.e. 18:00:00:00 (UTC).

Defining a time-based recurrence on the Trigger execution

You can schedule the Trigger execution several times each scheduled execution day. You then must define:

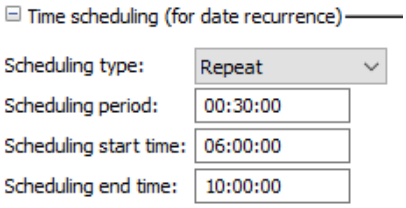
- the scheduling period (in hours):
  Default period: every 4 hours (04:00:00) each scheduled day.
- the start time of the time-based scheduling
- the end time of the time-based scheduling

To define a time-based recurrence on the Trigger execution:

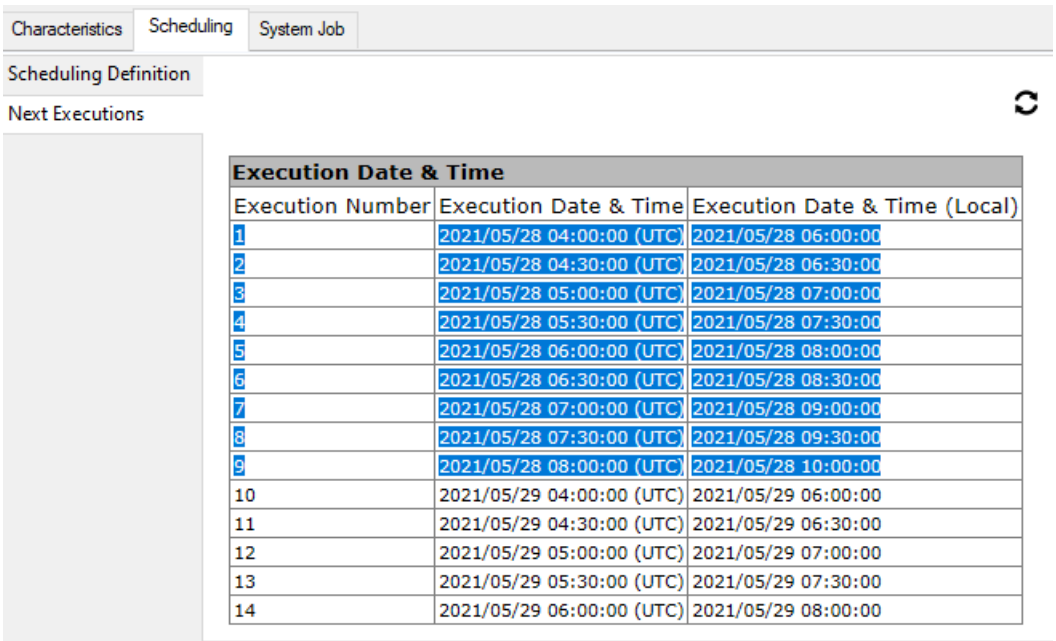
1. Access the Triggers.
   - *See Accessing Triggers.*
2. Right-click the Trigger concerned and select **Update Scheduling**.
3. In the **Time scheduling (for date recurrence)** section, use the drop-down menu of **Scheduling type** field to select "recurrent".

4. Define the recurrence (period, start time and end time of the time-based scheduling):
   - **Scheduling period** (hh:mm:ss format)
   - **Scheduling start time** (hh:mm:ss format)
   - **Scheduling end time** (hh:mm:ss format)

   E.g.: schedule the Trigger every 30 minutes from 6am to 10am in the defined time zone.



5. Click **OK**.
6. Check your scheduling configuration regarding the execution time recurrence:
   - Close and re-open the Trigger Management window.
   - Right-click the Trigger and select **Properties**.
   - In the **Scheduling** tab, select the **Next Executions** sub-tab.



Execution Date & Time		
Execution Number	Execution Date & Time	Execution Date & Time (Local)
1	2021/05/28 04:00:00 (UTC)	2021/05/28 06:00:00
2	2021/05/28 04:30:00 (UTC)	2021/05/28 06:30:00
3	2021/05/28 05:00:00 (UTC)	2021/05/28 07:00:00
4	2021/05/28 05:30:00 (UTC)	2021/05/28 07:30:00
5	2021/05/28 06:00:00 (UTC)	2021/05/28 08:00:00
6	2021/05/28 06:30:00 (UTC)	2021/05/28 08:30:00
7	2021/05/28 07:00:00 (UTC)	2021/05/28 09:00:00
8	2021/05/28 07:30:00 (UTC)	2021/05/28 09:30:00
9	2021/05/28 08:00:00 (UTC)	2021/05/28 10:00:00
10	2021/05/29 04:00:00 (UTC)	2021/05/29 06:00:00
11	2021/05/29 04:30:00 (UTC)	2021/05/29 06:30:00
12	2021/05/29 05:00:00 (UTC)	2021/05/29 07:00:00
13	2021/05/29 05:30:00 (UTC)	2021/05/29 07:30:00
14	2021/05/29 06:00:00 (UTC)	2021/05/29 08:00:00

The **Execution Date & Time** table indicates the first execution date and time and the following ones with their corresponding local time.

- *To set your local time, see Option: Time Zone.*

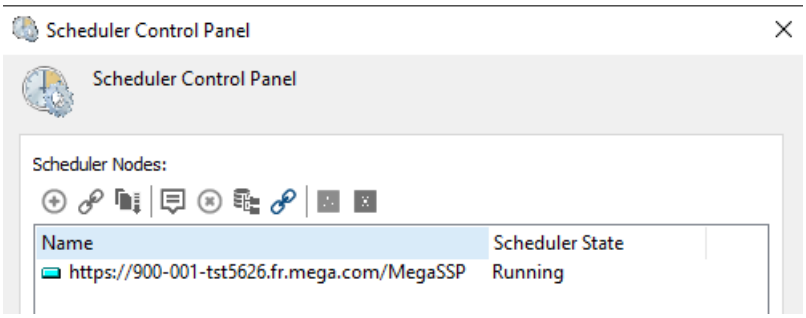
This example shows a daily scheduling, the Trigger is executed every day from 06:00:00 to 10:00:00 (Paris local time) i.e. from 04:00:00 to 08:00:00 (UTC).

MANAGING THE SCHEDULER

Accessing the Scheduler Control Panel

The **Scheduler Control Panel**:

- shows the Scheduler state on the SSP on which the Jobs are executed.
- enables to stop and restart the Scheduler.



To access the **Scheduler Control Panel**.

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **Repositories** folder, then the repository concerned.
3. Right-click the **Scheduler** folder and select **Scheduler Control Panel**.

Stopping the Sheduler

You can stop (or restart) the Scheduler.

To stop the Scheduler:

1. Access the **Scheduler Control Panel**.
2. Right-click the Scheduler concerned and select **Stop Scheduler**.
   - *To restart the Scheduler, select **Start Scheduler**.*

MANAGING EVENTS

HOPEX provides a supervision tool (**HOPEX Server Supervisor**) that enables management of events.

The following points are covered here:

- Introduction to supervision
- Supervision tool: HOPEX Server Supervisor
- Supervising events
- Events to be Monitored (Production Server)

INTRODUCTION TO SUPERVISION

The **HOPEX Server Supervisor** supervision tool is used to collect messages from **HOPEX** applications (**Windows Front-End** and **Web Front-End**). These messages include indicators to ensure that **HOPEX** is operating correctly.

```
Example: information or error messages.
```

A message corresponds to a supervision event, which has been previously coded in the executable. A client cannot create a new event.

The following points are covered here:

- Prerequisites to Supervision
- Supervising Events
- Supervision files
- Supervision configuration file: MegaSite.ini

Prerequisites to Supervision

Before starting supervision function, you must check the following points:

- MegaSSP web application is up and running.
  MegaSSP web application collects all of the messages.
- SSP service is started.
  The SSP stores messages in the supervision log file.
- The Megasite.ini file is configured.
  - *See Supervision configuration file: MegaSite.ini.*

Supervising Events

A supervision event can include about fifty pieces of information.

```
Example: "Event infos" information is a json (or a text)
that can provide details regarding the event context. The
json structure depends on each event.
```

To consult and analyze a supervision event, see Consulting a supervision event file.

Event types

Events are sorted by type

- A (Action): user action
- W (Warning): alert
- E (Error): error
- S (Snapshot): process snapshot

A type A event is characterized by:

- a start, which corresponds to its creation.
- an end, which corresponds to the moment the message is actually sent. Only one message is sent at the end. It summarizes the indicators of the process for the event duration.

Type W and type E events are immediate.

Type S events summarize the indicators of the process since the last sent snapshot.

Supervision files

Supervision files include supervision events.

Each supervision file represents a day.

Supervision file name format: sspsprvs<YYYYmmDD>.txt.

```
Example: SSPSPRVS20171219.TXT
(supervision file for the 19th of December 2017)
```

To find the supervision file location, in:

- **HOPEX Server Supervisor**, see Finding the supervision file location.
- **HOPEX Supervision console**, see Finding the Supervision File Location.

Supervision configuration file: MegaSite.ini

Configuration of some of the supervision behaviors is performed in the MegaSite.ini file (<**HOPEX** installation directory>\Cfg) in [Supervision] section.

In addition, Supervision components need to communicate with the SSP and access the [SSP] section content.

```
[Supervision]
StateInterval=<time interval>
Filter=<Filter>

[SSP]
url=<SSP url>
```

MegaSite.ini	Description
[Supervision]	Supervision parameter section
StateInterval	Time interval (in millisecond) between two supervision events of snapshot type. Minimum value: 1000 By default this parameter value corresponds to 3' (180000) (do not modify this parameter)
Filter	Enables definition of executables to be supervised. Other executable supervision is deactivated. If not specified, there is no filtering and all of **HOPEX** executables are supervised. Example: `Filter=AM` Displays only **HOPEX Windows Front-End** (code A) and its Administration (code M) events. See Executable code. To modify the filter, see Modifying processes to be supervised (MegaSite.ini filter)
[SSP]	
Url	SSP url If the url is not specified, supervision is deactivated.

Executable code

Each application is associated with a code:

Windows Front-End :

- A: Administration
  (mgwmapp.exe /DesktopAppGbm.Administration)
- M: HOPEX
  (mgwmapp.exe)
- N: Automation
  (API mgwmapp.exe/Automation)

Web Front-End :

- R: Session holder (mgwspro.exe)
- T: site (mgwmapp.exe)
- O: SSP environment holder (mgwspro.exe)
- I: SSP site (mgwmapp.exe)

SUPERVISION TOOL: HOPEX SERVER SUPERVISOR

The **HOPEX Server Supervisor** tool enables reading supervision files.

The **HOPEX Server Supervisor** tool gives access to calculated/filtered views of supervision events.

You can activate/deactivate the supervision of certain processes to filter messages. Only selected executables send messages.

– *See Supervision configuration file: MegaSite.ini.*

See:

- Starting HOPEX Server Supervisor
- Extend HOPEX Server Supervisor functionalities
- Modifying processes to be supervised (MegaSite.ini filter)
- Finding the supervision file location
- Modifying the supervision file location

Starting HOPEX Server Supervisor

To start the **HOPEX Server Supervisor** tool:

1. In the **HOPEX** installation folder, expand the **Utilities** folder, then **HOPEX Server Supervisor**.
2. Right-click **HOPEX Server Supervisor** and select **Execute as administrator**.

   The **HOPEX Server Supervisor** icon appears in the system tray of your workstation.

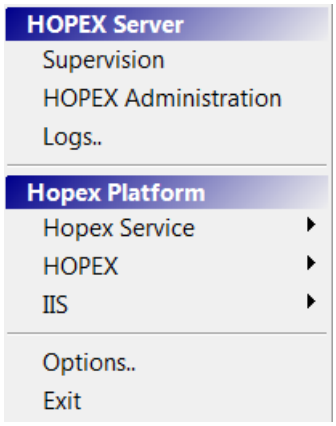
   – *The green button on the icon indicates that the SSP is ready and that IIS is started, else the button is red:* .

Extend HOPEX Server Supervisor functionalities

By default, at **HOPEX** installation, **HOPEX Server Supervisor** menus are minimum.

To extend access to **HOPEX Server Supervisor** functionalities:

1. In your workstation system tray, right-click **HOPEX Server Supervisor** .

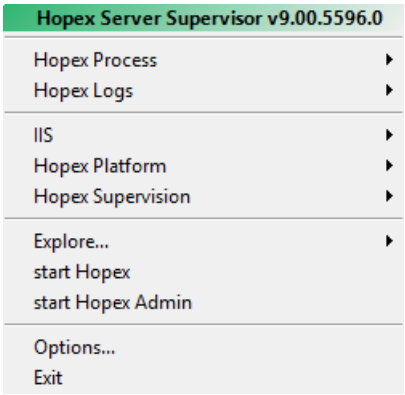


2. Click HOPEX Server .
   The **Command** field is displayed.

3. Enter "swl 1" and press "Enter".

All of **HOPEX Server Supervisor** functionalities are available.

- *To return to a minimum display, perform step 2, enter 'swm" and press "Enter".*



Modifying processes to be supervised (MegaSite.ini filter)

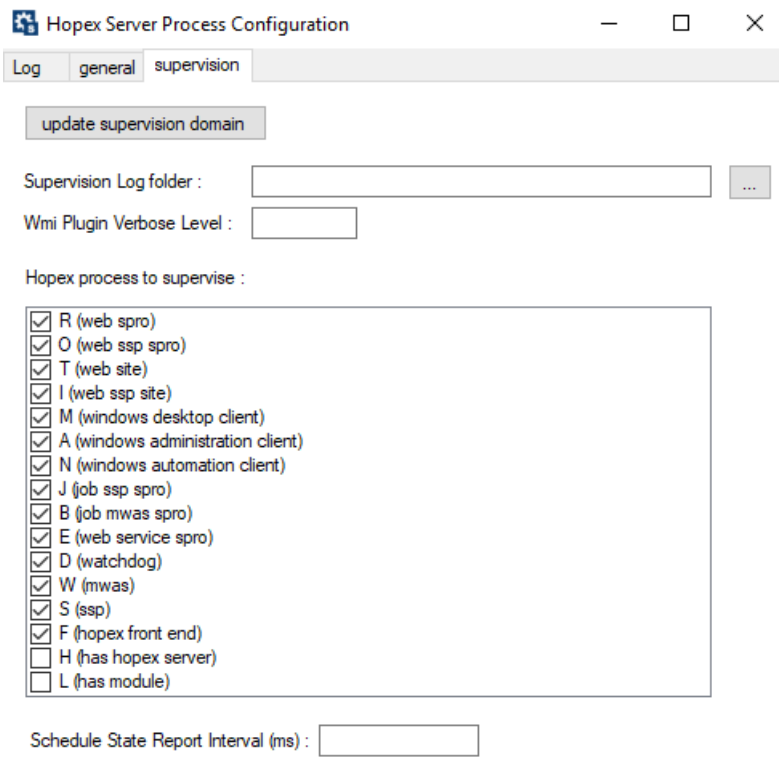
HOPEX Server Supervisor enables to configure the **Filter** section of MegaSite.ini file:

- *For information on MegaSite.ini file, see Supervision configuration file: MegaSite.ini*

To configure the **Filter** section of MegaSite.ini file:

1. In your workstation system tray, right-click **HOPEX Server Supervisor** and select **Hopex Supervision > Supervision configuration**.

2. From **HOPEX Server Process Configuration** window, **Supervision** tab, select the processes you want to supervise.



Your modifications are immediately taken into account in MegaSite.ini file.

Finding the supervision file location

To open the folder where the supervision files are stored:

1. Open **HOPEX Server Supervisor** in extended configuration.
   - *See Extend HOPEX Server Supervisor functionalities.*
2. From **HOPEX Server Supervisor**, select **Hopex Logs > Daily Logs Manager**.
3. Right-click the daily log row (sspsprvs<mm-dd-yy>.txt) and select **Open Folder**.
   - *To modify the supervision file location, see Modifying the supervision file location.*

Modifying the supervision file location

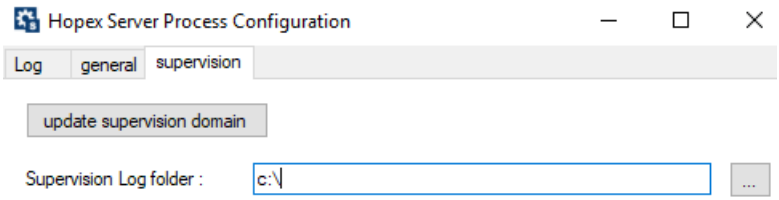
To modify the supervision file location:

1. Create the directory where you want the supervision files to be stored.

   ```
   Example: c:\log
   ```

2. From **HOPEX Server Supervisor**, select **Hopex Supervision > Supervision configuration**.
3. Select the **Supervision** tab.
4. In the **Supervision Log folder** field, use the browse button ... to define the directory path where the supervision files are stored.

   ```
   Example: c:\log
   ```



These modifications are immediately taken into account. Supervision files are stored in the specified directory (e.g.: c:\log).

SUPERVISING EVENTS

Event supervision is performed from the **Supervision** tool of **HOPEX Server Supervisor**.

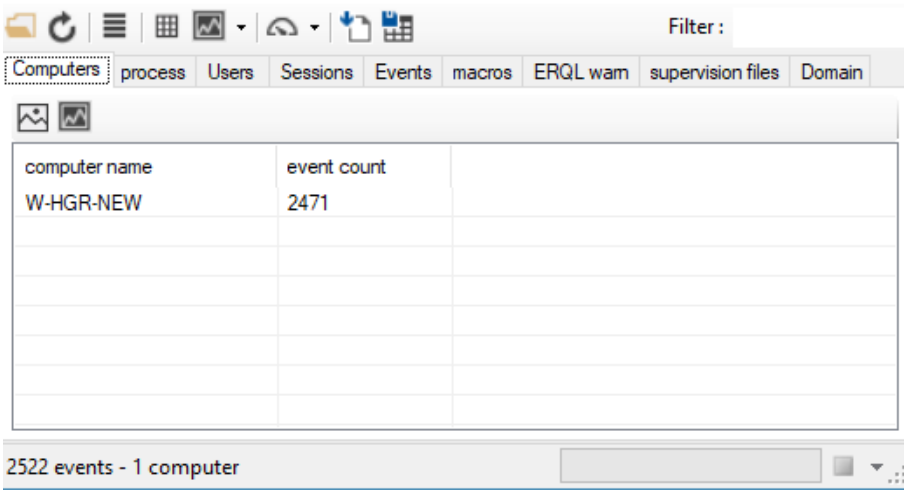
See:

- Supervision tool
- Consulting a supervision event file
- Actions from an event supervision window

Supervision tool

The **Supervision** tool of **HOPEX Server Supervisor** enables to open and analyze **HOPEX** event files.

– *To launch the **Supervision** tool, see Consulting a supervision event file.*



For detailed information on event files, see:

- Supervising Events
- Supervision files.

Supervision tool toolbar

From the **Supervision** tool toolbar, click:

-  to open one or several specific supervision files
-  to refresh calculated view data of the current supervision file
-  to open the set of snapshots that have been created on the different servers.
  A consolidated snapshot gives an application calculated view over the last three minutes.
-  to view the load state, object consumption on the set of supervised processes.
-  to load a supervision domain (reference domain - Standard) so that comparisons can be made.
-  export data in CSV format.

Supervision tool tabs

In the **Supervision** tool, events are grouped by calculated view that gives access to the corresponding list of prefiltered events. Views correspond to the following tabs:

- **Computers**
  This tab shows the list of servers used and its associated event number.
- **process**
  This tab shows the set of supervised processes of the set of servers.
- **Users**
  This tab shows the list of users who logged on the application.
- **Sessions**
  This tab shows the list of current or past sessions of all the servers or workstations supervised.
- **Events**
  This tab shows the list of HOPEX events that have been recorded.
- **macros**
  This tab enables to view macros, for which execution time is high.
- **supervision file**
  This tab shows analyzed supervision files.
- **Domain**
  This tab enables to show activity synthesis according to the supervision domain

Consulting a supervision event file

To consult an event of a supervision file:

1. From your workstation system tray, right-click **HOPEX Server Supervisor** and select **Supervision**.

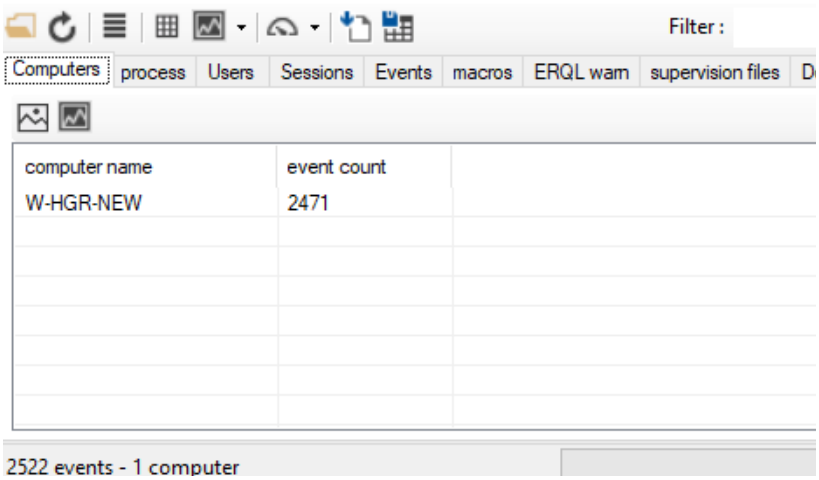
   – *If **HOPEX Server Supervisor** is in extended configuration (see Extend HOPEX Server Supervisor functionalities), select **Hopex Supervision > Supervision**.*

   The current supervision file opens.

   – *To open another (or several) supervision file , if **HOPEX Server Supervisor** is in extended configuration select **Hopex Supervision > Supervision from file** and select the files.*

   *Alternatively, from **Supervision chart** toolbar, click **Open Supervision data file** and select the files.*

   See Finding the supervision file location.



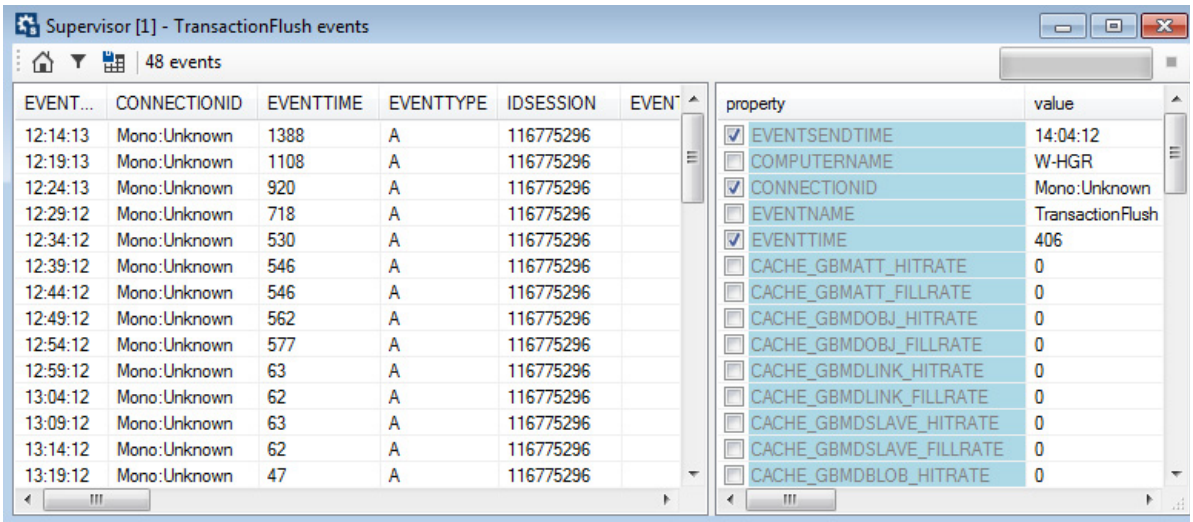
2. (optional) If you are on the current supervision file, click **Refresh** anytime you want to immediately refresh the view data.

   – *See Supervision tool tabs.*

3. Click the tab regarding the view you want to consult:

   – *See Supervision tool tabs.*

4. Double-click the line of the view you want to consult the events.
Events regarding this view are displayed in table format.



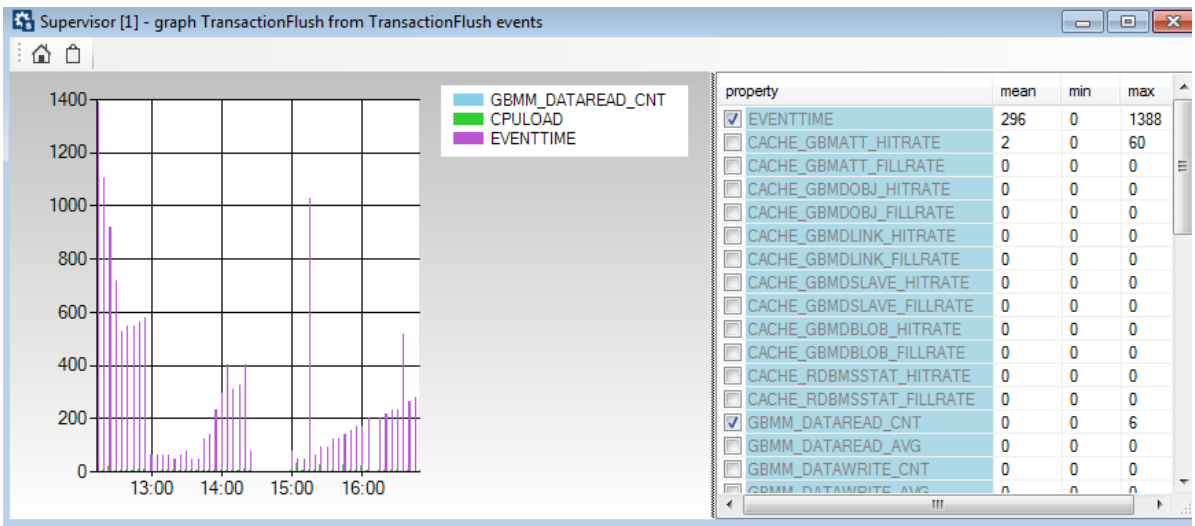
Each line represents an event of which characteristics are detailed.

- *See Actions from an event supervision window.*

5. (optional) In the **property** pane, select the indicators (columns) you want to be displayed in a table column.
6. (optional) Right-click the event for which you want to display a graph and select **show graph for <event name>**.

- *You cannot get a graph for an Error type or Warning type event.*

The graph is displayed.



7. In the **property** pane, select the indicators you want to be displayed in the graph.
   The graph is calculated on the set of the prefiltered original view, taking into account the event selected.
8. (optional) Click **copy graph image to clipboard** to copy the graph image in the clipboard.

Actions from an event supervision window

From the supervision window, which lists the events in table format, you can:

- access the window of the source **Supervision chart** tool.

  Click **Supervisor Home** .
- create filters to filter displayed events.

  Click **Filters** .
- export current events in .txt format.
  - *Example: to import events in Excel.*

  Click **Export the current filter view as supervision format** .
- interrupt data downloading.
  - *Example: when experiencing long downloading times.*

  Click **Interrupt loading** .

EVENTS TO BE MONITORED (PRODUCTION SERVER)

To keep your HOPEX production server in the best operating conditions, here are the events you have to monitor:

- Events: Login and Authentication
- Events: Configuration Management
- Events: Workspace Activity
- Events: Repository Connections
- Events: Service Execution
- Events: Infrastructure Performance and Repository Health
- Events: Data Import/Export Tracking
- Events: Report DataSet/GraphSet/TreeSet Generation
- Events: Scheduled Jobs
- Events: Reporting Datamart
- Events: Questionnaire Generation

In the following tables, each event **Type** column indicates the importance level of the event as follows:

- severity ++: requires an immediate action
- severity +: requires a rapid analysis
- severity -: requires your attention
- severity --: informative

Events: Login and Authentication

Events you have to monitor in a login and authentication context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
LoginPasswordChangeFailure	Alert (+)	***Password update***	***Error during a password update***
		"Login Name":	Name of the login of the user
LoginAuthenticationFailure	Alert (-)	***Authentication***	***Failure in the authentication process***
		"Login Name":	Name of the login of the user
		"Failure reason":	Error message
LicenseLoginFailure	Alert (+)	***License***	***Error while obtaining a license token***
		"Failure reason":	Error message

Events: Configuration Management

Events you have to monitor in a configuration management context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
GraphCompile	Action (--)	***Writing access diagram compilation***	***Should never happen on a production server*** Except after the update of a new writing access hierarchy definition
MetaDataCompile	Action (--)	***Environment compilation***	***Should never happen on a production server*** Except after the update of new customization
CompiledDataReset	Alert (+)	***Meta or technical data modification***	***Should never happen on a production server*** If this event is raised on a server, it means that a person is carrying out MetaModel extensions directly on the server
		"User":	User responsible for the update
		"CompiledDataUpdated":	Modified object
		"CompiledDataResetOrigin":	`Executed command.` This information is written during the upgrade or import process
GraphUpdate	Alert (+)	***Writing access diagram modification***	***Should never happen on a production server*** Except after the update of a new writing access hierarchy definition
		"GraphUpdated":	"Authorization" if the writing access diagram has been updated `"Visibility"` if the reading access diagram has beenupdated

Events: Workspace Activity

Events you have to monitor in a workspace activity context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
TransactionFlushFailure	Alert (++)	***Workspace flush***	***Unable to validate data update***
		`"Failure reason":`	`Error message`
TransactionDispatchFailure	Alert (++)	***Workspace dispatch***	***Failure in the authentication process***
		"Failure reason":	Error message

Events: Repository Connections

Events you have to monitor in a repository connection context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
HOPEXVersionMismatch	Action (++)	***Connection***	***Hopex services cannot start as HOPEX binary version (c:\ProgramFiles\MEGA\HOPEX Vxx) is not compatible with your environment version.***
		"HOPEXVersion":	HOPEX version number Example: 7.85.4848.0000
		"EnvironmentVer-sion":	Environment version number Example: 7.85.4843.0000

Event	Type (severity)	Context / JSON	Description / Content
RepositorySessionConnect	Action (--)	***Connection (session)***	***This event enables to monitor the connection times***
		"GBMSESSIONOPENMODE":	One of the following values: GBMSESSION_OPENMODE_READWRITE GBMSESSION_OPENMODE_READONLY_PHYSICAL GBMSESSION_OPENMODE_READONLY_LOGICAL GBMSESSION_OPENMODE_READONLY_REALTIME GBMSESSION_OPENMODE_READWRITE_REALTIME
		"Login":	IdAbs of the login
		"User":	IdAbs of the user
		"Profile":	IdAbs of the profile
		"Role":	IdAbs of the business role
		"pathDB":	Path of the SystemDb repository of the Environment
RepositorySessionConnectFailure	Alert (+)	**Connection**	**Error during the session connection phase**
		"GBMSESSIONOPENMODE":	One of the following values: GBMSESSION_OPENMODE_READWRITE GBMSESSION_OPENMODE_READONLY_PHYSICAL GBMSESSION_OPENMODE_READONLY_LOGICAL GBMSESSION_OPENMODE_READONLY_REALTIME GBMSESSION_OPENMODE_READWRITE_REALTIME
		"Login":	IdAbs of the login
		"User":	IdAbs of the user
		"Profile":	IdAbs of the profile
		"Role":	IdAbs of the business role
		"pathDB":	Path of the SysDb repository of the Environment
		"Failure reason":	Error message

Event	Type (severity)	Context / JSON	Description / Content
RepositoryOpenFailure	Alert (+)	**Repository opening phase**	**Error detected when opening a repository**
		"RepositoryName":	Name of the repository
		"Failure reason":	Error message
RepositoryCloseFailure	Alert (-)	**Repository closing phase**	**Error detected when closing a repository**
		"RepositoryName":	Name of the repository
		"Failure reason":	Error message
RepositorySessionRe-connect	Alert (++)	**The connection with the database server has been lost**	**HOPEX is trying to reconnect automatically to the server, each try generates an alert (5 by default)**
		"RepositoryServer-Name":	Name of the repository

Events: Service Execution

Events you have to monitor in a service execution context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
ScheduledJobEx-ecutionFailure	Alert (+)	**Job execution**	**Error when running a scheduled job**
		"Failure reason":	Error message
SchedulerError	Error (+)	**Scheduler service start**	**Unable to run the scheduler service due to a lack of license or other reason (see msg)**
		"Msg":	Error message
ProcessException	Error (++)	**Important exception occurred**	**Important error occurred when running HOPEX**

Event	Type (severity)	Context / JSON	Description / Content
ERQLExcessive-InvocationTime	Alert (-)	**An ERQL query executing more than 5000 GBM commands**	**Optimizing the query manually**
		"QueryId": or "QueryName":	Query identifier
		"GBM Get count":	Number of readings generated by the query
		"GBM Get count":	Number of searches generated by the query

Events: Infrastructure Performance and Repository Health

Performance and health tests are performed daily on all the repositories of all the environments.

- *See Generating a Repository Health Report.*

Repository health events to be monitored:

Event	Type (severity)	Context / JSON	Description / Content
RepositoryPerformanceROGraph-CommentReadAlert	Alert (+)	**Reading of 1000 large existing text (BLOB)**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure

Event	Type (severity)	Context / JSON	Description / Content
RepositoryPerformanceROGraph-LoadAlert	Alert (+)	**Existing graph exploration (1000 objects and 500 MetaAssociations)**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure (textual value)
RepositoryPerformanceROGraph-QueryAlert	Alert (+)	**ERQL queries on existing graph (1000 objects and 500 MetaAssociations)**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure (textual value)

Event	Type (severity)	Context / JSON	Description / Content
RepositoryPerformanceRWGraph-CommentWriteAlert	Alert (+)	**Writing of 1000 large texts (BLOB)**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure (textual value)
RepositoryPerformanceRWGraph-CreateAlert	Alert (+)	**Creation of a graph of 1000 objects and 500 MetaAssociations**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure (textual value)

Event	Type (severity)	Context / JSON	Description / Content
RepositoryPerformanceRWGraphDestroyAlert	Alert (+)	**Deletion of a graph of 1000 objects and 500 MetaAssociations**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure (textual value)
RepositoryPerformanceRWGraphQueryAlert	Alert (+)	**ERQL queries on a recently created graph (1000 objects and 500 MetaAssociations)**	**This test execution time is higher/lower than the reference time calculated for the last 30 executions. Event of this type may be due, for example, to: - the CPU load and/or the server memory available at test execution time (4 pm GMT) - an infrastructure change (e.g.: network, VM size, hardware) - the SQL repository maintenance plan execution time, which is too old - a cache configuration change (e.g.: redis, cache file) - anti-virus policy**
		Performance	Test execution time (in ms)
		Comparedtoaverage	Current measure characterization according to the standard measure (textual value)
RepositoryHealthSQLCompatibiltyLevelAlert	Alert (++)	**Compatibility check between your data SQL structure and your SQL server version**	**It is highly recommended not to work with a data structure version older than the SQL server one**
		Version	Version number of SQL server used (textual value)
		Compatibility	API compatibility type (textual value)

Event	Type (severity)	Context / JSON	Description / Content
RepositoryHealth-WorkspaceAlert	Alert (+)	**A workspace is older than 7 days**	**Keeping older workspaces reduces HOPEX performances**
		IdAbs	Workspace absolute identifier
		Type	Workspace type
		Owner	Owner name
		Device	Machine name
		Duration	Activity duration
Repository-HealthSQLTable-Alert	Alert (++)	**The table execution plan has never been performed or has been performed more than a week ago**	**An SQL maintenance plan must be performed at least once a week**
		Table	Table name
		LastExecution	Last execution time (or "never")
Repository-HealthSQLIndex-Alert	Alert (++)	**An index fragmentation exceeds 50%**	**An SQL maintenance plan must be performed at least once a week**
		Table	Table name
		Indexes	Index name
		Ratio	Fragmentation percentage
Repository-HealthSQLPrivateWorkspaceAlert	Alert (+)	**The RemovePrivateWorkspaceTempData stored procedure has never been performed or has been performed more than 7 days ago**	**The RemovePrivateWorkspaceTempData stored procedure must be performed at least once a week**
		LastExecution	Last execution time (or "never")
Repository-HealthSQLHistoricalDataAlert	Alert (+)	**The ShrinkUnusedHistoricalData stored procedure has never been performed or has been performed more than 7 days ago**	**The ShrinkUnusedHistoricalData stored procedure must be performed at least once a week**
		LastExecution	Last execution time (or "never")

Event	Type (severity)	Context / JSON	Description / Content
RepositoryHealth-Customisatio-nAlert	Alert (-)	**HOPEX MetaClass customization**	**It is forbidden to modify a HOPEX Meta-Class, as it may generate regressions at updates/migrations.**
		Metaclass	Customized MetaClass name
RepositoryHealth-MetaClassVolu-meAlert	Alert (-)	**A MetaClass for which a maximum cardinality is defined and for which the current cardinality is higher than the maximum cardinal-ity**	**Exceeding the maximum number of rec-ommended objects may degrade HOPEX performances**
		Name	Customized MetaClass name
		Volume	Current cardinality
		Volumemax	Maximum cardinality
RepositoryHealth-MetaAssociation-VolumeAlert	Alert (-)	**A MetaAssociation for which a maxi-mum cardinality is defined and for which the current cardinality is higher than the maximum cardinal-ity**	**Exceeding the maximum number of rec-ommended objects may degrade HOPEX performances**
		Name	MetaAssociation name
		Volume	Current cardinality
		Volumemax	Maximum cardinality
RepositoryHealth-MetaAssociation-ChildrenVolumeAl ert	Alert (-)	**A MetaAssociation for which a maxi-mum cardinality is defined on the chil-dren number and for which the chil-dren current cardi-nality is higher than the maximum cardinality**	**Exceeding the maximum number of rec-ommended objects may generate usability issues**
		Name	MetaAssociation name
		VolumeChildren	Children current cardinality
		Volumechildrenmax	Children maximum cardinality

Events: Data Import/Export Tracking

Events you have to monitor in a HOPEX data import/export tracking context:

Event	Type (severity)	Context / JSON	Description / Content
RepositoryLogExport	Action (-)	**Repository Log content export in a file**	
		"Repository"	HOPEX repository name
		"Login"	Name of the login of the user
		"File"	File where data is generated
		"Execution"	Succeeded or failed
RepositoryLogConsolidate	Action (-)	**Repository Log content consolidation (shrink)**	
		"Repository"	HOPEX repository name
		"Login"	Name of the login of the user
		"Execution"	Succeeded or failed
RepositoryLogDestroy	Action (++)	**Remove of all or part of the Repository Log**	
		"Repository"	HOPEX repository name
		"Login"	Name of the login of the user
		"DestroyMode"	Full or partial
		"Execution"	Succeeded or failed
RepositoryDataExport	Action (-)	**Export of a root object set to a command file (MGR or XMG)**	
		"Repository"	HOPEX repository name
		"Login"	Name of the login of the user
		"File"	File where data is generated
		"Execution"	Succeeded, aborted or failed

Event	Type (severity)	Context / JSON	Description / Content
RepositoryD-ataImport	Action (++)	**Command file import in a repository**	
		"Repository"	HOPEX repository name
		"Login"	Name of the login of the user
		"File"	File containing commands to be executed in the repository
		"RejectedCmd"	File where rejected commands are generated by HOPEX
		"NbCmdBySecond"	Number of commands imported by second during the import process
		"NbCmdAnalyzed"	Number of commands analyzed by the import process
		"NbCmdExecuted"	Number of commands executed by the import process
		"NbCmdRejected"	Number of commands rejected by the import process
		"Execution"	Succeeded, aborted or failed
RepositoryData-LogicalSave	Action (-)	**Saving the repository content in a command file**	
		"Repository"	HOPEX repository name
		"Login"	Name of the login of the user
		"File"	File containing commands to be executed in the repository
		"WARNING"	If the MGRCompare option is active a Warning is added to the event
		"Execution"	Succeeded or failed

Events: Report DataSet/GraphSet/TreeSet Generation

Events to be monitored in a Report DataSet, GraphSet, or TreeSet generation context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
Report DataSet Generate	Action (--)	***Report DataSet generation***	***Enables to find the Report DataSet, its definition and the data volume involved***
		"Report DataSet Name"	Name of the Report DataSet
		"Report DataSet Idabs"	Absolute Identifier of the Report DataSet
		"Report DataSet Definition Name"	Name of the Report DataSet Definition
		"Report DataSet Definition Idabs"	Absolute Identifier of the Report DataSet-Definition
		"Lines Count"	Number of rows (hidden or not) of the Report DataSet
		"Columns Count"	Number of columns (hidden or not) of the Report DataSet
Report DataSet Generate (Alert)	Alert (+)	***Report DataSet generation***	***The data volume in the Report DataSet exceeds the data volume recommended by MEGA***
		"Report DataSet Name"	Name of the Report DataSet
		"Report DataSet Idabs"	Absolute Identifier of the Report DataSet
		"Report DataSet Definition Name"	Name of the Report DataSet Definition
		"Report DataSet Definition Idabs"	Absolute Identifier of the Report DataSet-Definition
		"Lines Count"	Number of rows (hidden or not) of the Report DataSet
		"Lines Limit"	Maximum number (recommended) of rows allowed in the Report DataSet: 100 000
		"Columns Count"	Number of columns (hidden or not) of the Report DataSet
		"Columns Limit"	Maximum number (recommended) of columns allowed in the Report DataSet: 50

Event	Type (severity)	Context / JSON	Description / Content
Report DataSet Serialization Update	Action (--)	***Serialization of a Report DataSet***	***The Report DataSet has been regenerated, its serialization has been updated***
		"Report DataSet Name"	Name of the Report DataSet
		"Report DataSet Idabs"	Absolute Identifier of the Report DataSet
		"Report DataSet Definition Name"	Name of the Report DataSet Definition
		"Report DataSet Definition Idabs"	Absolute Identifier of the Report DataSet-Definition
		"Lines Count"	Number of rows (hidden or not) of the Report DataSet
		"Columns Count"	Number of columns (hidden or not) of the Report DataSet
Report DataSet Serialization Load	Action (--)	***Generation of a Report DataSet***	***The Report DataSet has been generated and serialized***
		"Report DataSet Name"	Name of the Report DataSet
		"Report DataSet Idabs"	Absolute Identifier of the Report DataSet
		"Report DataSet Definition Name"	Name of the Report DataSet Definition
		"Report DataSet Definition Idabs"	Absolute Identifier of the Report DataSet-Definition
		"Lines Count"	Number of rows (hidden or not) of the Report DataSet
		"Columns Count"	Number of columns (hidden or not) of the Report DataSet

Event	Type (severity)	Context / JSON	Description / Content
GraphSet Generate	Action (--)	***Generation of a GraphSet***	***Enables to find the GraphSet, its definition and the data volume involved***
		"GraphSet Name"	Name of the GraphSet
		"GraphSet Idabs"	Absolute Identifier of the GraphSet
		"GraphSet Definition Name"	Name of the GraphSet Definition
		"GraphSet Definition Idabs"	Absolute Identifier of the GraphSet Definition
		"Node count"	Number of nodes in the GraphSet
		"Internal node count limit"	Maximum number (recommended) of nodes allowed in the GraphSet: 1000
		"Internal node count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the GraphSet (Available when the recommended value has been overloaded)
		"Arc count"	Number of arcs in the GraphSet
		"Internal arc count limit"	Maximum number (recommended) of arcs allowed in the GraphSet: 10,000
		"Internal arc count limit (overloaded)"	Maximum number (overloaded) of arcs allowed in the GraphSet (Available when the recommended value has been overloaded)

Event	Type (severity)	Context / JSON	Description / Content
GraphSet Generate (alert: too many elements)	Alert (-)	***Generation of a GraphSet***	***The data volume in the GraphSet exceeds the limits recommended by MEGA.*** ***Nevertheless the GraphSet is entirely generated as either:*** ***- the "Deactivates the display limit in reports" option is activated*** ***- the recommended limits for arcs and/or nodes have been increased*** These limits are defined in **Options > Documentation > Reports**
		"GraphSet Name"	Name of the GraphSet
		"GraphSet Idabs"	Absolute Identifier of the GraphSet
		"GraphSet Definition Name"	Name of the GraphSet Definition
		"GraphSet Definition Idabs"	Absolute Identifier of the GraphSet Definition
		"Node count"	Number of nodes in the GraphSet
		"Internal node count limit"	Maximum number (recommended) of nodes allowed in the GraphSet: 1000
		"Internal node count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the GraphSet (Available when the recommended value has been overloaded)
		"Arc count"	Number of arcs in the GraphSet
		"Internal arc count limit"	Maximum number (recommended) of arcs allowed in the GraphSet: 10,000
		"Internal arc count limit (overloaded)"	Maximum number (overloaded) of arcs allowed in the GraphSet (Available when the recommended value has been overloaded)

Event	Type (severity)	Context / JSON	Description / Content
GraphSet Generate (truncated)	Error (+)	***Generation of a GraphSet***	***The GraphSet has not been entirely generated as its data volume exceeds the limits recommended by MEGA: more than 1000 nodes and/or more than 10,000 arcs***
		"GraphSet Name"	Name of the GraphSet
		"GraphSet Idabs"	Absolute Identifier of the GraphSet
		"GraphSet Definition Name"	Name of the GraphSet Definition
		"GraphSet Definition Idabs"	Absolute Identifier of the GraphSet Definition
		"Node count"	Number of nodes in the GraphSet
		"Internal node count limit"	Maximum number (recommended) of nodes allowed in the GraphSet: 1000
		"Internal node count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the GraphSet (Available when the recommended value has been overloaded)
		"Arc count"	Number of arcs in the GraphSet
		"Internal arc count limit"	Maximum number (recommended) of arcs allowed in the GraphSet: 10,000
		"Internal arc count limit (overloaded)"	Maximum number (overloaded) of arcs allowed in the GraphSet (Available when the recommended value has been overloaded)

Event	Type (severity)	Context / JSON	Description / Content
TreeSet Generate	Action (--)	***Generation of a Tree-Set***	***Enables to find the TreeSet, its definition and the data volume involved***
		"TreeSet Name"	Name of the TreeSet
		"TreeSet Idabs"	Absolute Identifier of the TreeSet
		"TreeSet Definition Name"	Name of the TreeSet Definition
		"TreeSet Definition Idabs"	Absolute Identifier of the TreeSet Definition
		"Node Count"	Number of nodes in the TreeSet
		"Internal node count limit"	Maximum number (recommended) of nodes allowed in the TreeSet: 50 000
		"Internal node Count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the TreeSet (Available when the recommended value has been overloaded)
		"Property count"	Number of properties in the TreeSet
		"Internal node property Count limit"	Maximum number (recommended) of properties allowed in the TreeSet: 50
		"Internal property count limit (overloaded)"	Maximum number (overloaded) of properties allowed in the TreeSet (Available when the recommended value has been overloaded)

Event	Type (severity)	Context / JSON	Description / Content
TreeSet Generate (aborted)	Error (+)	***Generation of a Tree-Set***	***The TreeSet has not been generated as it includes too many properties*** Limits are defined in **Options > Documentation > Reports**
		"TreeSet Name"	Name of the TreeSet
		"TreeSet Idabs"	Absolute Identifier of the TreeSet
		"TreeSet Definition Name"	Name of the TreeSet Definition
		"TreeSet Definition Idabs"	Absolute Identifier of the TreeSet Definition
		"Node Count"	Number of nodes in the TreeSet
		"Internal node count limit"	Maximum number (recommended) of nodes allowed in the TreeSet: 50 000
		"Internal node Count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the TreeSet (Available when the recommended value has been overloaded)
		"Property count"	Number of properties in the TreeSet
		"Internal node property Count limit"	Maximum number (recommended) of properties allowed in the TreeSet: 50
		"Internal property count limit (overloaded)"	Maximum number (overloaded) of properties allowed in the TreeSet (Available when the recommended value has been overloaded)

Event	Type (severity)	Context / JSON	Description / Content
TreeSet Generate (truncated)	Error (+)	***Generation of a TreeSet***	***The TreeSet has not been entirely generated as it includes too many nodes*** Limits are defined in **Options > Documentation > Reports**
		"TreeSet Name"	Name of the TreeSet
		"TreeSet Idabs"	Absolute Identifier of the TreeSet
		"TreeSet Definition Name"	Name of the TreeSet Definition
		"TreeSet Definition Idabs"	Absolute Identifier of the TreeSet Definition
		"Node Count"	Number of nodes in the TreeSet
		"Internal node count limit"	Maximum number (recommended) of nodes allowed in the TreeSet: 50 000
		"Internal node Count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the TreeSet (Available when the recommended value has been overloaded)
		"Property count"	Number of properties in the TreeSet
		"Internal node property Count limit"	Maximum number (recommended) of properties allowed in the TreeSet: 50
		"Internal property count limit (overloaded)"	Maximum number (overloaded) of properties allowed in the TreeSet (Available when the recommended value has been overloaded)

Event	Type (severity)	Context / JSON	Description / Content
TreeSet Generate (alert: too many elements)	Alert (-)	***Generation of a Tree-Set***	***The data volume in the TreeSet exceeds the limits recommended by MEGA.*** ***Nevertheless the TreeSet is entirely generated as either:*** ***- the "Deactivates the display limit in reports" option is activated*** ***- recommended limits for nodes and/ or properties have been increased*** These limits are defined in **Options > Documentation > Reports**
		"TreeSet Name"	Name of the TreeSet
		"TreeSet Idabs"	Absolute Identifier of the TreeSet
		"TreeSet Definition Name"	Name of the TreeSet Definition
		"TreeSet Definition Idabs"	Absolute Identifier of the TreeSet Definition
		"Node Count"	Number of nodes in the TreeSet
		"Interal Node Count limit"	Maximum number (recommended) of nodes allowed in the TreeSet: 50 000
		"Node Count limit (overloaded)"	Maximum number (overloaded) of nodes allowed in the TreeSet (Available when the recommended value has been overloaded)
		"Property Count"	Number of properties in the TreeSet
		"nteral Property Count limit"	Maximum number (recommended) of properties allowed in the TreeSet: 50
		"Property Count limit (overloaded)"	Maximum number (overloaded) of properties allowed in the TreeSet (Available when the recommended value has been overloaded)

Events: Scheduled Jobs

Events you have to monitor in a scheduled job context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
ScheduledJobMerge	Alert (-)	**A scheduled job will not be performed as the same scheduled job is already being performed or pending**	
		"message":"JOB <jobid> merged with the pending JOB <jobid>"	The message indicates the merged job identifier and the pending job identifier
		"message":"JOB <jobid> merged with the pending JOB <jobid>"	The message indicates the identifier of the merged job and the identifier of the job in progress
ScheduledJobStackAlert	Alert (++)	**Exceedance of the pending job number threshold from which all of the cancelable pending jobs (by default all of them) must be canceled**	
		"message":"too many jobs are waiting (<nb of stacked job>), cancel cancelable jobs"	The message indicates the number of pending jobs to be deleted
ScheduledJobCancel	Alert (+)	**The job indicated has been canceled as the pending job number threshold has been reached**	
		"message":" JOB <jobid> canceled because there is too many jobs stacked"	The message indicates the scheduled job identifier

Events: Reporting Datamart

Events you have to monitor in a Reporting Datamart creation or update context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
DatamartStructureInit	Action (-)	**Creation of all the tables and columns according to the HOPEX repository metamodel and the profile selected**	
		"RepositorySource":	Source HOPEX repository name
		"DatamartTarget"	Target Reporting Datamart name

Event	Type (severity)	Context / JSON	Description / Content
DatamartDataInit	Action (-)	**Creation and feeding of all the Reporting Datamart tables, columns and rows according to the HOPEX repository content**	
		"RepositorySource":	Source HOPEX repository name
		"DatamartTarget"	Target Reporting Datamart name
DatamartIncrementalUpdate	Action (-)	**Incremental update of the Reporting Datamart objects, based on the updates performed on the HOPEX source repository**	
		"RepositorySource":	Source HOPEX repository name
		"DatamartTarget"	Target Reporting Datamart name
DatamartCalculatedAttSynchro	Action (-)	**Alignment of the Reporting Datamart with the values of all the calculated MetaAttributes of the HOPEX source repository**	
		"RepositorySource":	Source HOPEX repository name
		"DatamartTarget"	Target Reporting Datamart name
DatamartDiagramSynchro	Action (-)	**Alignment of the Reporting Datamart with the drawings of all the diagrams of the source HOPEX repository**	
		"RepositorySource":	Source HOPEX repository name
		"DatamartTarget"	Target Reporting Datamart name

Events: Questionnaire Generation

Events you have to monitor in a questionnaire generation context are listed in the following table:

Event	Type (severity)	Context / JSON	Description / Content
Assessment Survey Generation	Action (-)	**Generation of questionnaires**	
		"Assessment Session"	Name of the assessment session
		"NbOfQuestionnaires"	Number of questionnaires
Deployment Generation	Action (-)	**Generation of deployment nodes**	
		"Assessment Session"	Name of the assessment session
		"NbOfNodes"	Number of deployment nodes generated
		"NbOfObjects"	Number of objects

Event	Type (severity)	Context / JSON	Description / Content
Aggregation Generation	Action (-)	**Generation of the aggregation**	
		"Assessment Session"	Name of the assessment session
		"NbOfNodes"	Number of aggregation nodes generated
		"NbOfObjects"	Number of objects

8

MANAGING OBJECTS

The following points are covered here:

- Exporting HOPEX Objects
- Protecting Objects (function available with **HOPEX Power Supervisor**)
- Comparing and Aligning Objects Between Repositories
- Merging Two Objects (function available with **HOPEX Power Supervisor**)
- Importing a Solution Pack in HOPEX
- Managing UI Access (Permissions) (function available with **HOPEX Power Supervisor**)
- Managing Shapes

EXPORTING HOPEX OBJECTS

Export of an object with propagation enables creation of a consistent set allowing transfer of part of the repository to another repository. For example, export of **HOPEX** objects from a library includes objects present in the library and their dependent objects.

The following points are detailed here:

- Export
- Exporting Objects
- Viewing Objects Before Export

Export

You can export common modeling objects as well as configuration and parameterization objects.

```
Examples: report templates (MS Word), queries, metamodel
extensions, users.
```

To access these objects, you must select the extended metamodel options in your configuration. This option is available in user options, from the **Repository** icon.

– *See Managing Options.*

Export uses the propagation mechanism, which can be configured using perimeters.

– *For detailed information on propagation mechanism, see the **HOPEX Power Studio - Perimeters** technical article.*

Propagation steps in organizational process (major) export:

1. For an organizational process (major) you also export its operations (minor).
2. For an operation you export the event messages or result messages (minor) of the operation (major)
3. Propagation continues step by step until all links have been explored.

   – *Export takes account of link types: for certain link types, search in depth of other links stops.*

   All of the tags are also exported.

Exporting Objects

You can export **HOPEX** objects from:

- **HOPEX Administration**
- **HOPEX**

– *To export objects from the **Administration** desktop (Web Front-End) see the HOPEX Administration - Supervisor Web guide.*

You can export objects in the following formats:

- **plain text**
  The exported file is in the form of an .MGR file.
  - *For more details on .MGR file syntax, see Command File Syntax.*
- **XML MEGA**
  The exported file is in the form of an .XMG file containing commands or data (objects and links).
  - *For more details on MEGA XML data exchange format, see technical article "MEGA Data Exchange XML Format 70".*
- **Excel**
  - *See the **HOPEX Common Features** guide, "Exchanging Data With Excel" chapter.*

You can password-protect the export file generated. The exported file has the .mgz format and can only be imported by entering the password you defined.

See:

- Exporting HOPEX objects
- Exporting a HOPEX object from the object

Exporting HOPEX objects

You can export objects from **HOPEX Administration** or from your **HOPEX** desktop.

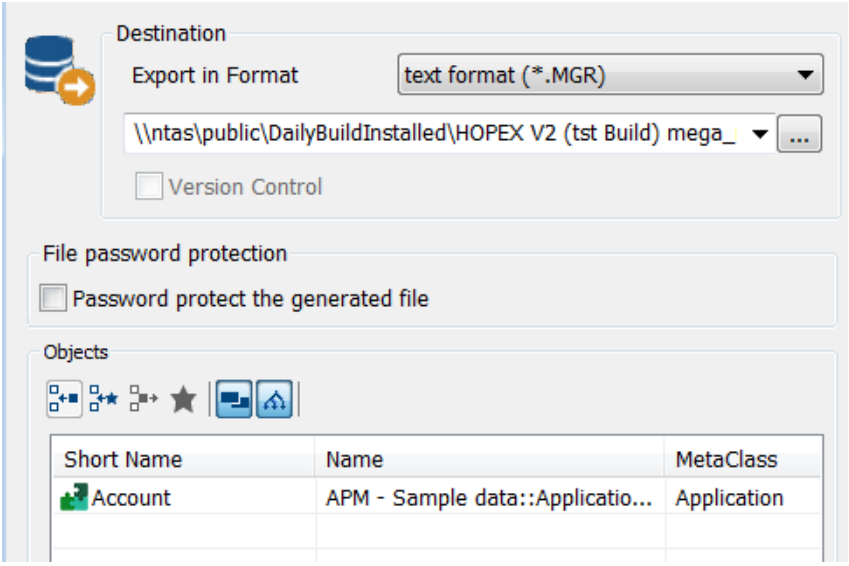
To export **HOPEX** objects:

1. Access the repository from which you want to export objects.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Object Management > Export Objects**.
   - *Alternatively, from your **HOPEX** desktop, select **File > Export > MEGA objects**.*

   The **Export HOPEX Objects** dialog box opens.
3. In the **Export in Format** field, select the export file format.
   Several options enable object export configuration.
   - *See Managing Options.*
4. (Optional) Enter a different name and folder if the default values are not suitable.
   - *Button ... enables you to browse the folder tree and select where the export file will be located.*
5. (If needed) In the **File password protection** pane, select "Password protect the generated file".
6. In the **Objects** frame, click **Add Objects to List**  .
   M *To simplify the query, click* **Add objects to list from favorites**  .

   The query dialog box appears.
7. Start the query and select the appropriate objects in the result window.

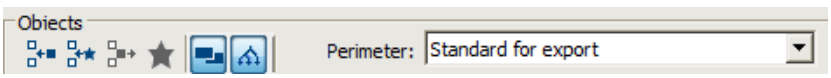
8. Click **OK**.
The selected objects are added to the **Export HOPEX Objects** dialog box list, preceded by their type.



You can carry out this procedure several times, allowing you for example to export objects of different types.

- *In the event of an error, click **Remove objects from list** to delete an object from the list.*

9. In the **Objects** group box, by default two export configuration options are proposed:
   - **Include Objects of Merging**, which allows you to export technical objects resulting from merging objects (_TransferredObject).
     - *For further information on merging objects, see Merging Two Objects.*
   - **Propagate**, which allows you to export listed objects together with their dependent objects.
     - *To view objects before export, see Viewing Objects Before Export*
10. (Optional) By default, the export perimeter is as defined in the properties of the Export tool. To modify the export default perimeter, you must have previously activated export perimeter selection, see Activating the export perimeter selection option.
In the **Objects** frame, select the **Perimeter** of export using the drop-down menu.

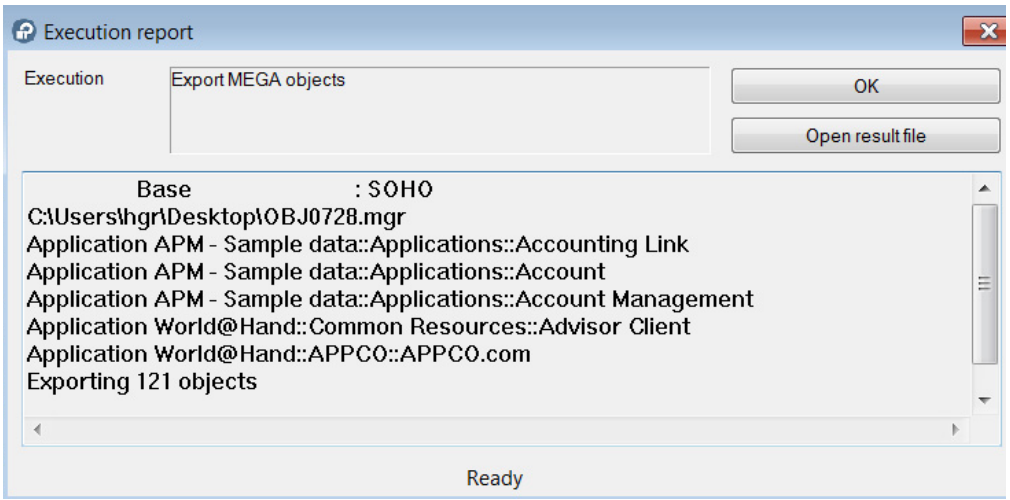


11. When selection is complete, click **Export**.

12. (If step 5 you selected the password protect file option) In the **File protection** dialog box, enter a password, confirm it and click **OK**.
The export process begins.

    - *To interrupt export during execution, click **Cancel**.*

    During export, the name and type of the objects being exported appear at the bottom of the dialog box, together with duration of export. On completion of export, the **Execution Report** displays the number of exported objects.



- *See Viewing an export file.*

Exporting a HOPEX object from the object

You can export an object from your **HOPEX** desktop.

To export a **HOPEX** object from the **HOPEX** desktop:

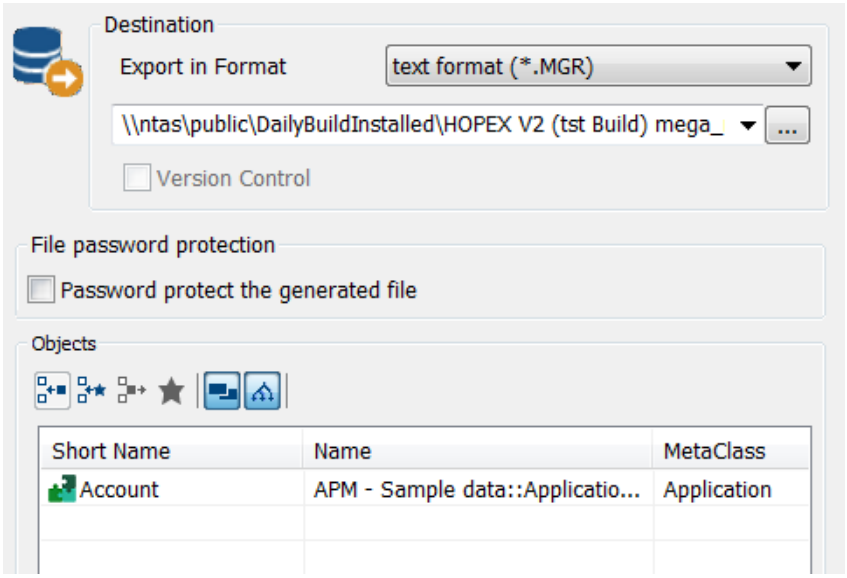
1. Connect to **HOPEX**

    - *See Connecting to HOPEX.*

2. In the **Main Objects** navigation window, right-click the object you want to export and select **Manage > Export**.

   - *You can also export **HOPEX** objects by selecting **File > Export > HOPEX Objects** in your workspace.*

   The **Export HOPEX Objects** dialog box opens.



   In the **Objects** frame, the object to be exported is already selected.

3. Refer to the procedure Exporting HOPEX objects.

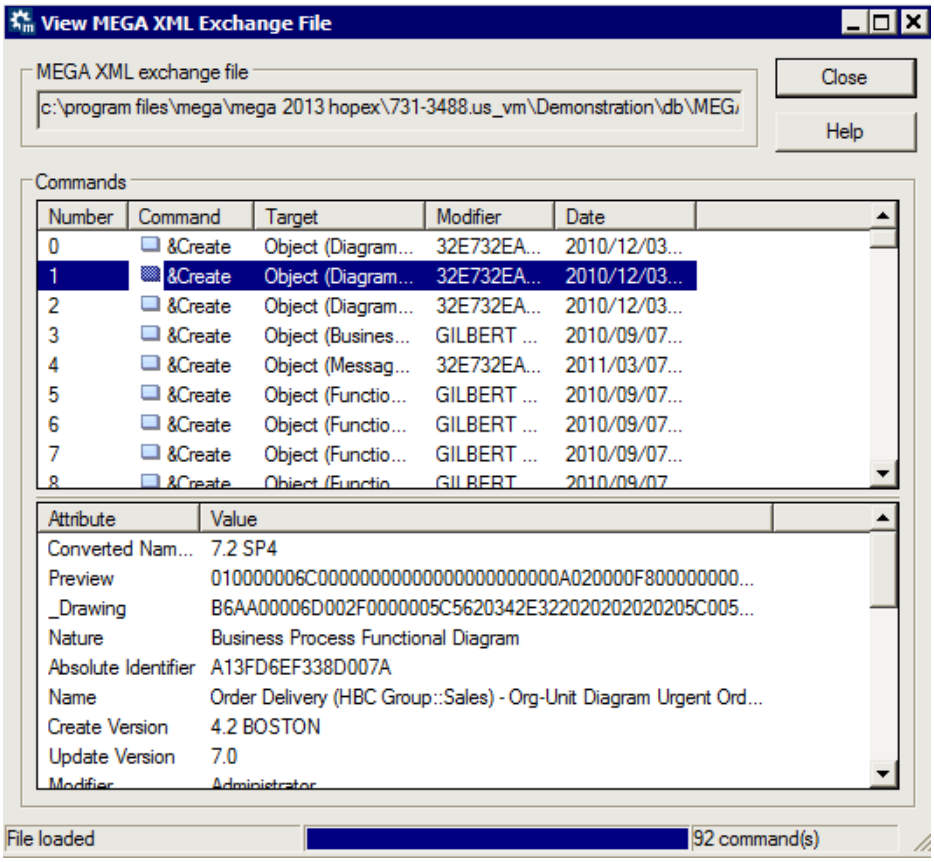
Viewing an export file

To view the exported file:

1. Export objects.

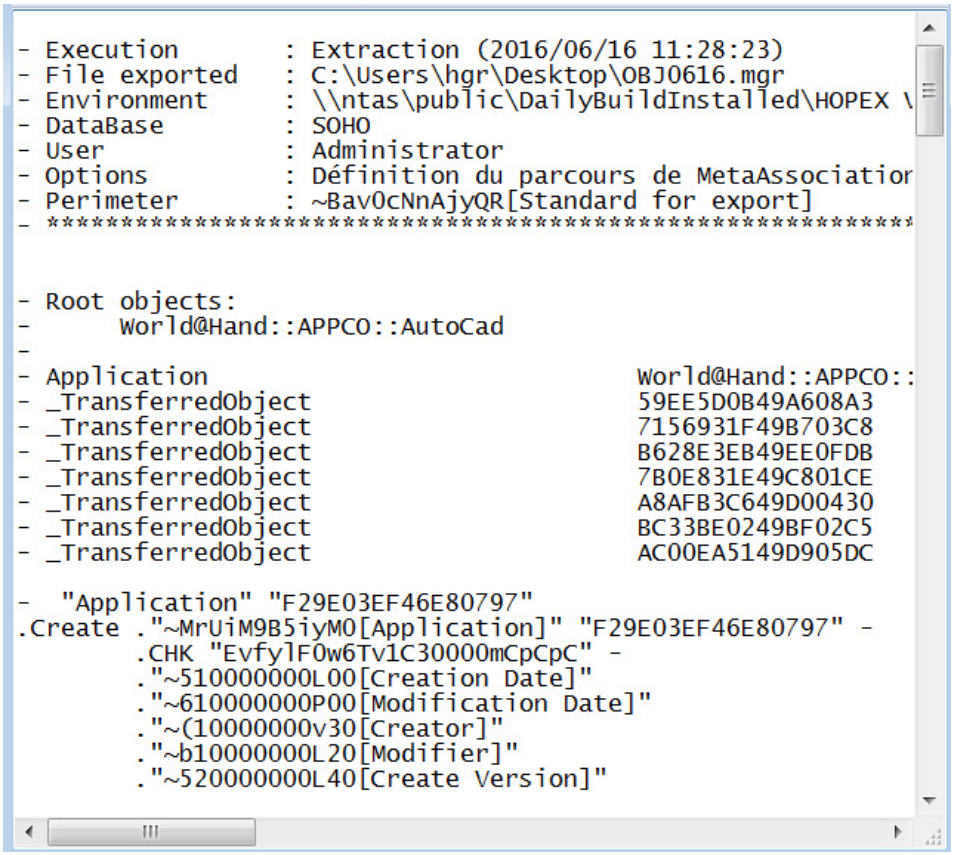
   - *See Exporting Objects.*

2. In the export **Execution Report** dialog box, click **Open result file**
In the case of:
   - an export in **MEGA** XML format, the view MEGA XML exchange file dialog box appears.



This file is presented in the form of a table showing a list of commands. Corresponding to each command ("Create", "Connect", etc.) there is a an object, object modification date and name of the last modifier.

   - export in text format, the Notepad dialog box appears:



This file lists all objects in text format.

The exported file can then be imported into another repository.

– *For more details on updating a repository by command file import, see Updating a Repository.*

Activating the export perimeter selection option

– *This option is not available in Web Front-End.*

At the time of export, to be able to select export perimeter, you must activate the **Activate export perimeter selection** option:

1. From your **HOPEX** workspace, select the **Tools > Options** menu.
2. In the HOPEX **Data Exchange > Export > Files** option group: **Generic Options** select the **Activate export perimeter selection** option.

Viewing Objects Before Export

– *Viewing of objects is available with the **HOPEX Power Supervisor** technical module.*

Viewing models with a perimeter allows preview of the operation result before its execution, and modification of the operation if required. You can therefore:

- see default impact of the applied perimeter, see Viewing objects.
- modify behavior of the perimeter according to the different types of links browsed from the root object, see the **MetaStudio** guide.

In the case of object export, other objects (connected to the root object) are also exported - they are determined by behavior of the "***Standard for export***" perimeter related to links existing around the root object.

> *Before exporting one or several **HOPEX** objects, you may find it useful to view all objects that will be exported. These can be objects you have selected as well as those deduced by the propagation mechanism.*
>
> *For detailed information on perimeters, see the **HOPEX Power Studio** - **Perimeters** technical article.*

Enabling the view option

To view objects that will be exported, you must enable the **View objects before export** option:

1. From your **HOPEX** workspace, select **Tools > Options**.
2. In the HOPEX **Data Exchange > Export > Files** option group: **Generic options** select **View objects before export** option.

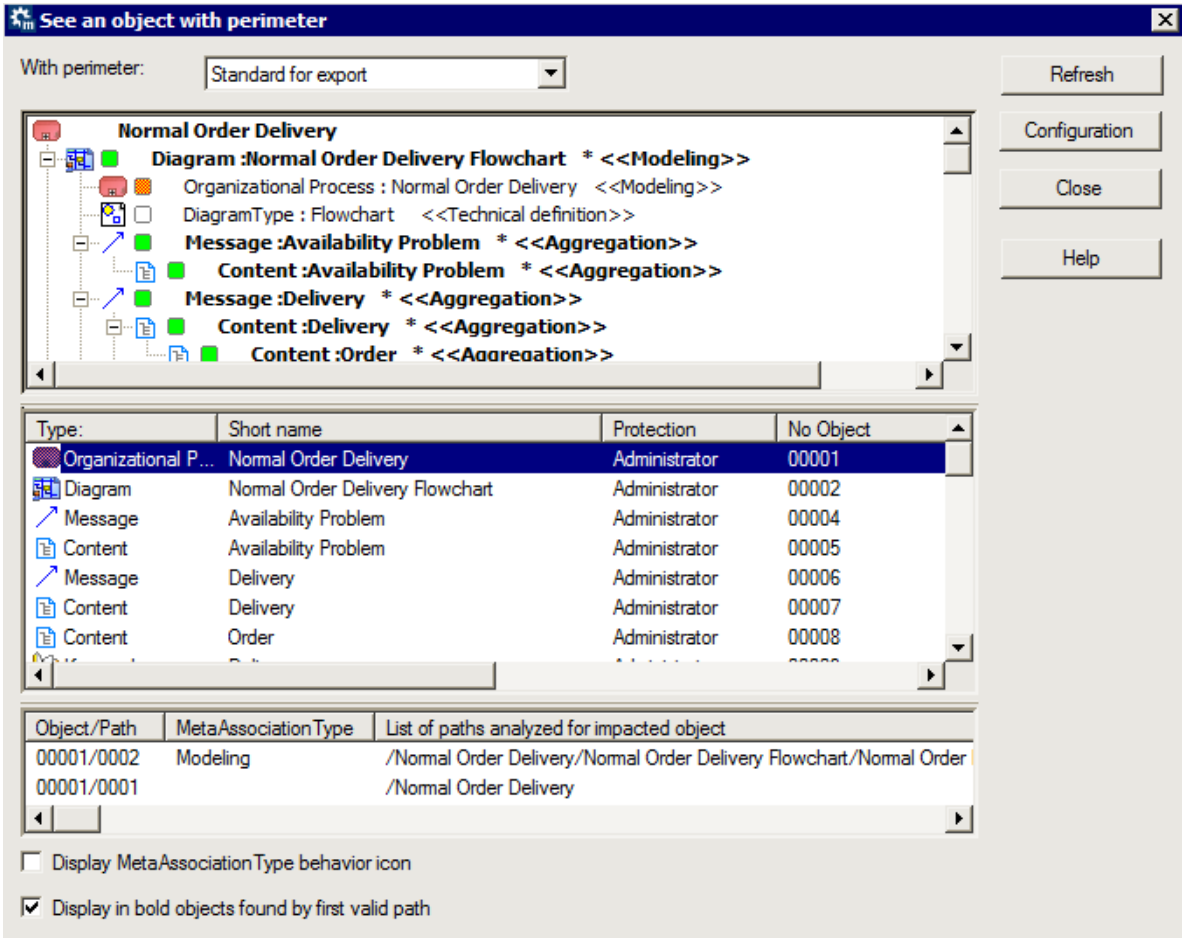
Viewing objects

To view objects that will be exported:

1. Select menu **Tools > Options**.
2. In the HOPEX **Data Exchange > Export > Files** option group: select the **View objects before export** option.

   The **View** button appears in the **Objects** frame.

3. In the **Export HOPEX Objects** dialog box, select an object in the list and click **View** .
The object detailed view window appears.



The ***See an object with perimeter*** window presents all exported objects in two ways (result of propagation applied to the root object):

- *To view the impact of other perimeters on the object concerned, select another **Perimeter** in the drop-down list.*

- The top frame presents, in tree form, the objects that will be exported with the root object. For each object it details:
  - the propagation behavior defined by the "***Standard for export***" perimeter for the link browsed.
    The behavior of this operator depends on the various link types and will determine the objects that will also be exported.
  - the corresponding link type (for example ***Modeling***).
  - the propagation type (identified by an icon) that will be executed on the object.

Table: Description of propagation behaviors

Icon	Value	Propagation description
	Deep	***Recursive complete propagation***: Takes into account this link and the opposite object only. Propagation continues.
	Standard	***Simple propagation***: Takes into account this link and the opposite object only. Propagation stops.
	Link	***Limited propagation:*** Takes into account this link but not the opposite object. Propagation stops.
	Abort	***No propagation:*** Does not take into into account this link or the opposite object. No propagation:

You can customize display of these results by selecting:

- **Display MetaAssociationType behavior icon**, which presents propagation behavior defined for the MetaAssociationType.
- **Display in bold objects found by first valid path**.

- the middle table lists objects that will be exported with the root object. For each object it details:
  - the number of paths linked to the object.
  - the comment associated with the object.
  - the bottom table details all paths by which the object selected in the middle table has been found, together with the corresponding link type (MetaAssociationType).

To locate an object/path in the tree:

› From one of the tables, right-click the object/path you want to locate and select **Find in tree**.

The **Configuration** button accesses the perimeter configuration tool.

– *For detailed information, see how to configure a perimeter in the **HOPEX Power Studio - Perimeters** guide.*

PROTECTING OBJECTS

The object protection function is available with the **HOPEX Power Supervisor** technical module.

An object has the writing access level of the user who created it. The writing access level of an object can be modified:

- directly
- by its dependence on another object (project, process, etc.) by propagating the authorization level for that object. This dependence may be indirect.
  - *See Viewing Objects Before Export for more details.*

See:

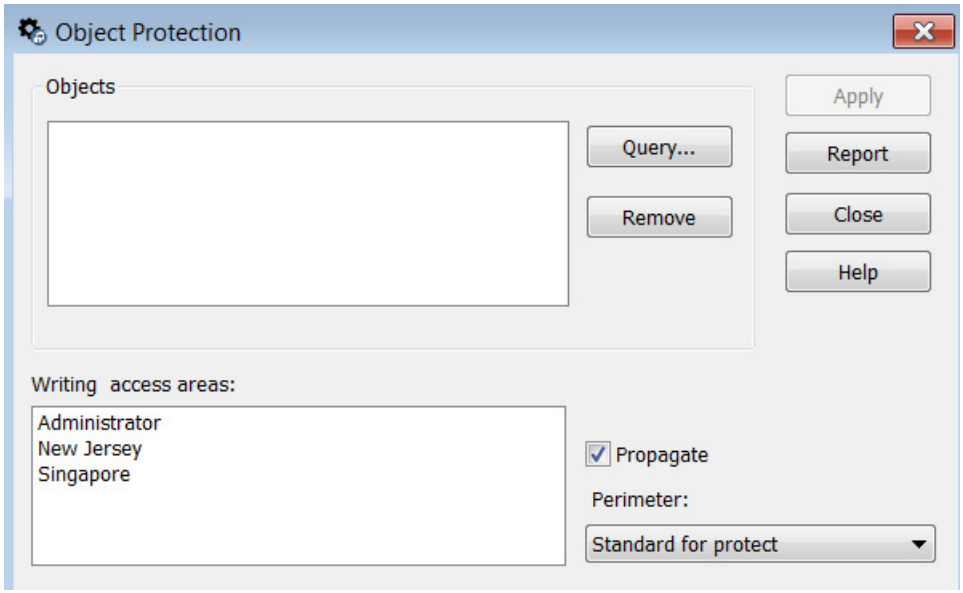
- Accessing the Object Protection Management Window
- Assigning a Writing Access Area to an Object
- Propagating Authorizations Between Linked Objects

Accessing the Object Protection Management Window

To access the object protection management window:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **Repositories** folder.

3. Right-click **SystemDb** and select **Object Management > Protect Objects**.
   The **Protect objects** dialog box opens.



   The **Writing Access Areas** list presents the writing access area of the user accessing the dialog box, and if appropriate, its dependent access areas (it is not possible to assign writing access areas higher than its own writing access area).

   Writing access areas are identified by level.

   When you open this dialog box, its **Objects** list is empty.

   - *See Assigning a Writing Access Area to an Object.*
   - *See Propagating Authorizations Between Linked Objects.*

Assigning a Writing Access Area to an Object

To assign a writing access area to an object:

1. Access the **Object Protection** window.
   - *See Accessing the Object Protection Management Window.*
2. In the **Writing Access Areas** list, select a writing access area.
3. Click **Query** to start a query.
4. Run the query, select the appropriate objects from the results, and click **OK**.
   The selected objects are added to the **Protect objects** dialog box list. Their types are also displayed.
   - *You can carry out this procedure several times, allowing you for example to protect objects of different types.*
   - *In event of error, select the unwanted object in the list and click **Delete**.*

5. Select the objects in the list.
6. When selection is complete, click **Apply**.
   The objects are assigned the selected protection level.

   – *You can select some objects in the list and assign one authorization to these, then select other objects and assign a different writing access level without executing a new query.*

Propagating Authorizations Between Linked Objects

You can automatically assign writing access level of an object to objects linked directly or indirectly to it.

1. Access the **Object Protection** window.

   – *See Accessing the Object Protection Management Window.*
2. Click **Query** to select the objects concerned.
3. In the **Writing Access Areas** list, select a writing access area.
4. Select the **Propagate** check box.
5. Click **Apply**.
   Objects dependent on those for which propagation is requested are protected with the same writing access level.

   – *For more details, see Viewing Objects Before Export.*

   – *Note that this type of propagation may modify the authorizations for objects shared by several projects or diagrams.*

COMPARING AND ALIGNING OBJECTS BETWEEN REPOSITORIES

HOPEX enables comparison and alignment of:

- two complete repositories
- objects in different repositories
- objects of the public repository with those of the current private workspace.
- a file and a repository (or repository objects)
- two repository archived states

  – *The objects compared must not be in the same private workspace.*

See:

- Compare and Align Principle
- Compare and Align Warnings
- Comparing and Aligning Two Repositories

Compare and Align Principle

The principle of comparing and aligning objects between repositories is as follows:

1. ***Extraction***
   The selected objects and any linked objects are extracted from the two repositories, browsing links according to **HOPEX** principles of object extraction.

   Comparison

   The two sets of data are compared on the basis of *absolute identifiers* of the objects they contain.
2. ***Comparison result***
   A window displays the results of the comparison. You can also generate a report and a command file in this window.

   – *The page showing differences displays a maximum of 1000 lines. If the list of differences is greater than 1000 lines, a message prompts you to either ignore this limit and display all the lines (in this case, the list may take some time to load) or not.*
3. ***Alignment***
   The upgrade command file is imported in the target repository.

Compare and Align Warnings

You must be aware of the following points before alignment and selection of the user executing alignment.

Repository log

The repository log lists all modifications made in the repository. It gives users a better understanding of actions executed in a repository in private workspaces. Each time an action is executed, an occurrence of Change Item is created.

- *For more details on repository log, see Managing logfiles.*

The repository log is not transferred from one repository to the other: a new log is created in the target repository. Object history is not therefore kept.

Users

The creator/modifier of an object in the target repository is the user executing the alignment.

The date of creation of an object is the date on which alignment was executed.

Reading (confidentiality) and writing access levels

Writing and reading access levels are taken into account during the comparison and during the alignment.

- *For more details on writing and reading access management, see Managing Data Reading Access and Managing Data Writing Access.*

To perform a comparison and an alignment, you must have reading access (if reading access management is activated) and maximum reading access for all objects in the repository.

- *Reject files are generated on completion of alignment. To delete files: in environment options **Options > Data Exchange > Import/Export Synchronization > MEGA**, select the option **Delete files produced at compare/align on completion of processing**.*

Comparing and Aligning Two Repositories

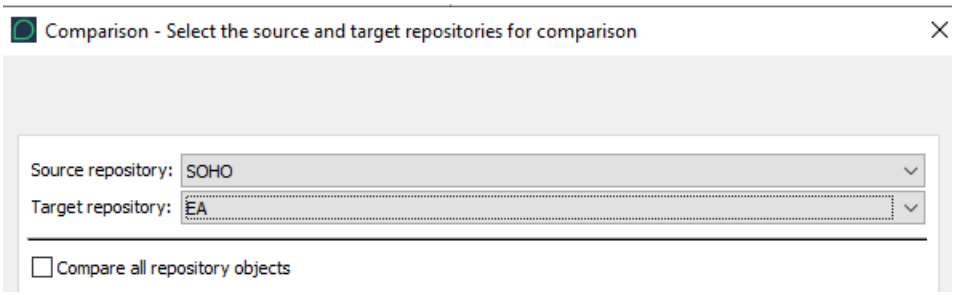
- *Before comparing and aligning, see Compare and Align Warnings*

To compare and align two repositories:

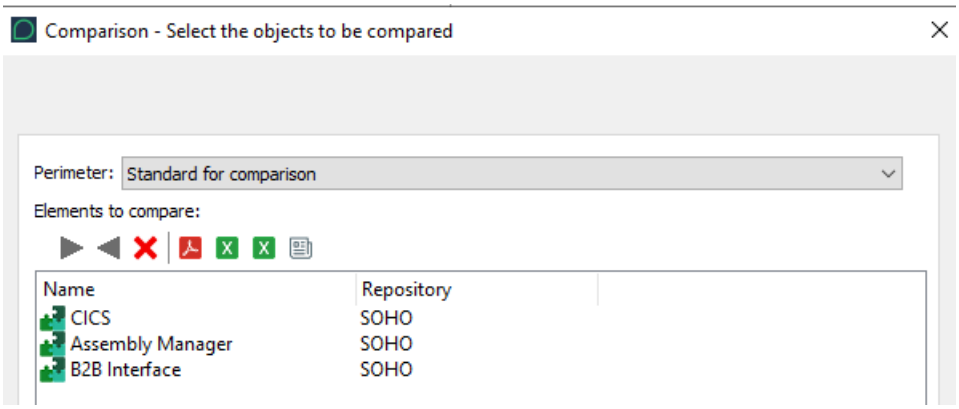
1. Connect to **HOPEX**
   - *See Connecting to HOPEX.*
   - *You can also, from **HOPEX Administration**, in the environment concerned, right-click the repository concerned and select **Compare and Align**.*
2. In the **HOPEX** menu bar, select **Tools > Manage > Compare and Align**.
   - *You can also right-click the object and select **Manage > Compare and Align**.*

   The object comparison wizard opens.
3. Indicate if you want to compare:
   - two repositories
   - two current repository archived states (RDBMS repository only)
   - a file and a repository
4. Click **Next**.

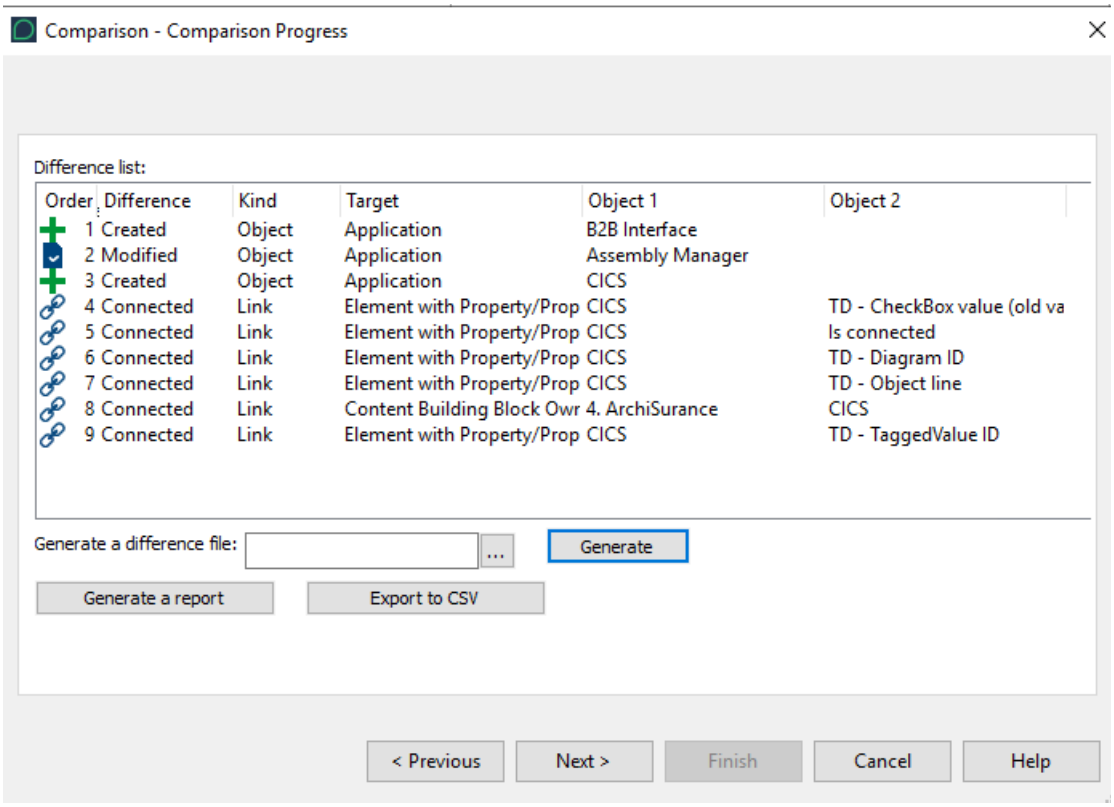
5. Select:
   - the **Source repository**
   - the **Target repository**, which is the repository to be updated.
     - *It can be a private workspace of the repository.*



6. (Optional) If required, you can choose to **Compare all repository objects**.
   - *Processing of this option can take some time.*
7. Click **Next**.
   The dialog box for selection of objects to be compared opens.
8. In the **Perimeter** field, select the perimeter type (by default **Standard for Comparison**)
   - *For detailed information on perimeters, see the **HOPEX Power Studio** - **Perimeters** technical article.*
9. In the **Elements to compare** pane, select:
   - to add objects from the source repository, or
   - to add objects from the target repository.
     - *If you have opened the comparison wizard from an object, this object is automatically added in the list of objects to be compared.*



10. Click **Next**.
The **Comparison Progress** window opens. It presents the differences between compared objects and their modifications.



The **Difference** column presents differences by update category:

- ***Created***: objects not existing in the target repository.
- ***Deleted***: objects existing in the target repository but not in the source repository.
  - *Deletion commands of compare and align can be generated in a separate file. To do this, activate the corresponding option in **Options > Data Exchange > Import/Export Synchronization > MEGA**.*
- ***Modified***: objects of which characteristics, including name, have been modified.
- ***Connected***: links, between two objects, that do not exist in the target repository.
- ***Disconnected***: links existing in the target repository but not in the source repository.
- ***Changed***: links for which a characteristic has been modified.

The **Kind** column presents differences by type.

11. In the **Generate a difference file** field (.mgr format):
    - Click ... and enter the name and location of the comparison file, then click **Save**.
    - Click **Generate** to generate the .mgr file that contains the list of differences detected.
12. (Optional) Click **Generate Report** to generate the report (.pdf format) which contains:
    - the overview of differences detected
    - the detailed list of differences detected
    - statistics
13. (Optional) Click **Export CSV** to generate a CSV file of the differences detected.

    – *The cmd-align-YYYY-MM-DD-hh-mm-ss file is stored in <environment name>\DB\<repository name>\USER\<user code>.*
14. Click **Next**.
    Differences are imported in the target repository.
    The target repository is aligned with the source repository.

    – *An alignment file with the content of differences (align-YYYY-MM-DD-hh-mm_555.mgr) is automatically saved in folder <Environment Name>\Db\<Repository Name>\USER\<User Code>.*

    If the alignment contains rejects, click **Display Rejects** to open the alignment rejects file (.mgr format).

    A rejects file is automatically saved in folder <Environment name>\Db\<Repository name>\USER\<User Code> (rejects file-reject-YYYY-MM-DD-hh-mm_555.mgr). This file is empty if alignment does not contain rejects.
15. Click **Finish**.

MERGING TWO OBJECTS

The object merge feature is available with the **HOPEX Power Supervisor** technical module.

When you merge two objects, you obtain a single object by transferring the *characteristics* and *links* from one object to the other. The source object is deleted. It is therefore recommended that merges be done in a new private workspace, so you can discard the changes if the result is not satisfactory.

You must have administration rights to merge two objects.

Choice of the objects to be merged

The **Target** object is the reference object that will be merged with the **Source** object. By default:

- its characteristics are not modified
- merging proposes addition of source object links.

The **Source** object is the object of which:

- you want to reuse certain characteristics or certain links
- characteristics and links will be transferred to the **Target** object.

When the link is to be a unique link (e.g., for sub-typing where the type is unique), the link of the target object is kept by default.

P **When merging is completed, the source object is deleted.**

M *You can **Explore** objects using the corresponding command. It can also be used to explore their links.*

Merging Two Objects

To merge two objects:

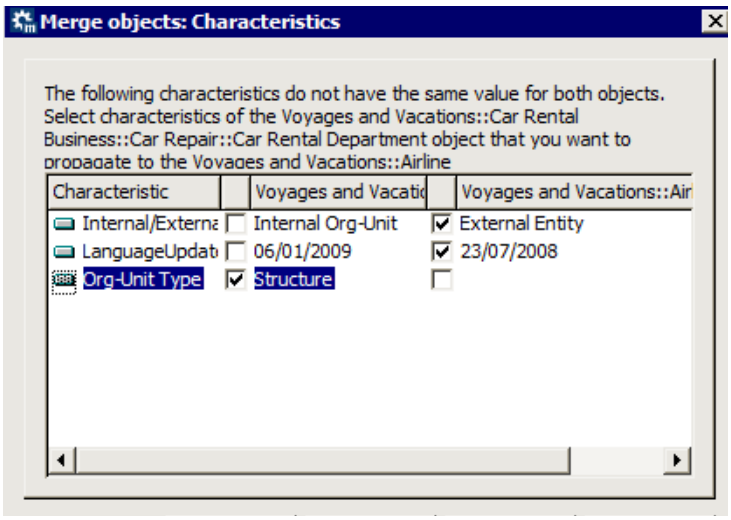
1. Connect to **HOPEX**
   - *Connecting to HOPEX.*
2. In the menu bar, select **Tools > Manage > Merge Two Objects**.
   The first step of the **Merge objects** wizard appears: **Object selection**.
   - *To have the right to merge two objects, you must have the right to delete objects.*
3. In the **Object** pane, select the object type to be merged.
   - *See Choice of the objects to be merged.*
4. In the **Source** field, click the arrow and select the source object.
5. In the **Target** field, click the arrow and select the target object.

6. Click **Next**.

   – *If the target and source objects are the same, the **Next** button is disabled.*

   The second step of the **Merge objects** wizard appears: **Characteristics**. It presents the differences found in characteristic values of both objects.

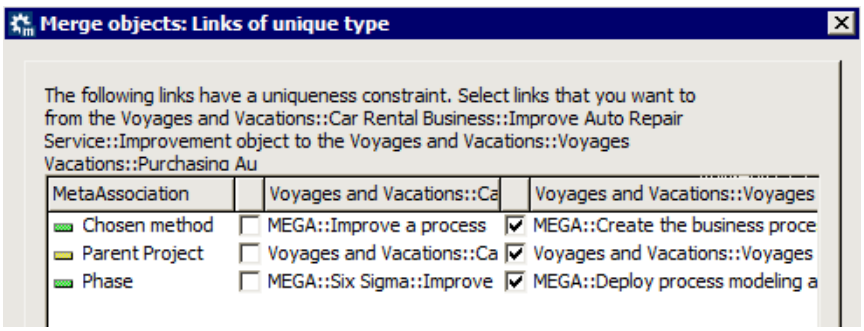
7. Select the source object characteristics you want to transfer to the target object. Characteristics that remain selected in the target object are kept.



8. Click **Next**.
   The third step of the **Merge objects** wizard appears:**Unique links** (those that can only exist once for a given object).

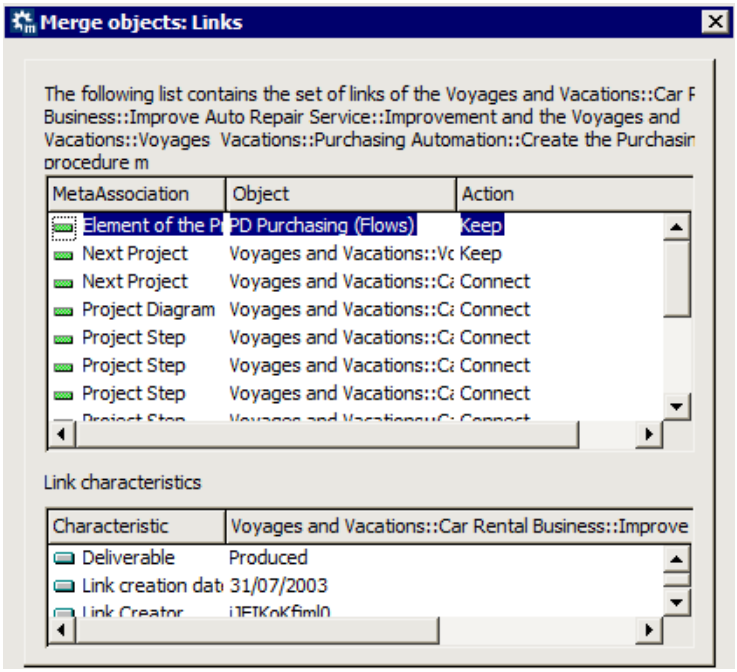
   ```
   Example: a message can have only one super-type.
   ```

   – *This step appears only if the objects to be merged have unique links connecting them to different objects.*



9. Select the links to be transferred.

10. Click **Next**.
    The **Merge objects** wizard shows: **Links**.



- When the link does not exist for the target object, the default is to connect the target object (**Action** : "Keep"). You can select the **Do not connect** command so as not to transfer the link.

You can **Keep** existing links, or **Disconnect** them.

- When the two objects are linked to the same object by the same link, it is possible to **Not change characteristics** of the link for the target object, or to **Copy characteristics** of the link for the source object. In this case, you can indicate for each characteristic whether to use the value of the source object, or keep the value of the target object.

11. Click **Next**.
12. Click **Finish** to start merging.
    The gauge indicates progress of the operation.

    When merging has been completed, the source object no longer exists and the selected *characteristics* and *links* have been transferred to the target object.

> – *"_TransferredObject" temporary merge objects are created on this occasion. Merge objects of a repository can all be exported at export of **HOPEX** objects.*

IMPORTING A SOLUTION PACK IN HOPEX

A Solution Pack includes data and/or system data.

Depending on the Solution Pack, the import must be done:

- on each HOPEX (data) repository of the environment.
- on the SystemDb repository of the environment.

Solution Pack	Import into each HOPEX repository	Import into the system repository (SystemDb)	Compile the environment
APQC.exe			
Archimate V3.exe			✓
Information Architecture.exe			
ISO.exe			
IT Infrastructure.exe			
MEGA.exe			✓
Military Terms.exe			
NAF.exe			✓
PPM.exe			
PRIVACY MANAGEMENT.exe			
SGBD SQL Type.exe			
UnifiedDesktop.exe			✓

Optional | Mandatory | Requires environment compilation

Prerequisite: the Solution Pack that you want to import is unzip in the **HOPEX** installation folder > **Utilities > Solution Pack** (double-click the Solution Pack to extract it).

– *Some of these Solutions Packs (and other add-ons) must be downloaded from the HOPEX Store (https://community.mega.com/t5/forums/filteredbylabelpage/board-id/hopex-store/label-name/content & frameworks) into the HOPEX installation folder > Utilities > Solution Pack.*

To import a Solution Pack in a repository:

1. From **HOPEX Administration**, connect to the environment concerned.
   – *See Connecting to an Environment.*
2. Expand the **Repositories** folder.

3. Right-click the repository concerned and select **Object Management > Import Solution Pack**.
   The **Solution Pack Import** dialog box appears.
4. Select the Solution Pack(s) that you want to import.
5. Click **OK**.
   The **Import MEGA Data XML** dialog box displays import progress.
   The selected Solution Packs are imported into the repository.
6. (If needed, see previous table) Repeat the steps for each HOPEX repository.
7. (If needed, see previous table) Compile the environment.
   - *See Compiling an Environment.*

MANAGING UI ACCESS (PERMISSIONS)

See :

- Introduction to UI Access Management (Permissions)
- Object UI Access Values
- Managing UI Access
- Managing Data Access Dynamically
- Generating a Report on Permissions by Profile
- Managing General UI Access

Introduction to UI Access Management (Permissions)

Prerequisites and definitions

UI access management (Permission management) is only available with the **HOPEX Power Supervisor** technical module.

You can manage:

- *object UI access*

) *Object UI access defines user rights on creation, reading, update, and deletion on these objects and their tools. By default, object UI accesses have value *CRUD (C: create, R: read, U: update, D: delete, *: default value).*

  – *For information on management of workflow UI accesses, see the HOPEX Power Studio > Customizing Workflows > Managing permissions on Workflows documentation.*

- *general UI access*

) *General UI access defines if tools are available or not. By default, general UI accesses have value *A (A: Available, *: default value).*

You can manage UI access (permissions) in:

- **HOPEX Administration**, in the **UI access** window.
- **HOPEX** with the **HOPEX Customizer** profile, in the **Permission Management** window.

  – *To manage UI access in the **Web Administration** desktop, see Managing UI Access (Permissions).*

UI access (permissions) of a profile are defined by its associated Set of UI access rights.

The **UI Access** window enables management of UI access for the complete environment and for each Set of UI access rights:

- **Object UIs** details its access to UI of objects and its access to tools specific to these objects.
  - *See Object UI Access Values.*
  - *See Managing UI Access.*
- **General UIs** details its access to general UIs.
  - *See Object UI Access Values.*
  - *See Managing General UI Access.*

Performance

For optimum performance, after modifying permissions you must compile the permissions.

- *Permission compilation is recommended in a production environment, see Compiling an Environment.*
- *Permission compilation is recommended in a production environment, see the HOPEX Administration > Managing Environments > Compiling an environment.*

Accessing the UI access management (HOPEX Administration)

To access the UI access management window:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.
3. Right-click the **UI access** folder and select **Manage**.
   The dialog box for managing **UI Access** opens.
4. Click the tab:
   - **Object UIs**
   - **General UIs**

Accessing the permission management (HOPEX)

To access the permission management window:

1. Connect to **HOPEX** (HOPEX Customization Desktop) with the **HOPEX Customizer** profile.
   - *See Connecting to HOPEX.*
2. In the **HOPEX** menu bar, select **Tools > Profile and Permission Management > Permission Management**.
   The **Permission Management** windows opens.
3. Click the tab:
   - **Object UIs**
   - **General UIs**

Object UI Access Values

Object UI access enables definition of user permissions on the selected metamodel.

- Preceding the value of a permission, the character:
  - * indicates that the value is directly inherited from the default value.
  - - indicates that the value is inherited from an element hierarchically higher in the same profile or sub-profile.
- Value empty means that the user has no permission on the element. The element is not visible to the user.
  When a MetaClass is hidden to a user, it is not available in the repository.

  For example, if the "Package" MetaClass is hidden for a user, this user cannot use packages in modeling work since this object type is not accessible in the interface.

See:

- MetaClass occurrence access permissions
- MetaAssociationEnd access permissions
- MetaAttribute access permissions
- Permissions on a tool

MetaClass occurrence access permissions

By default, the access permission on occurrences of a MetaClass takes value *CRUD:

- C: Create
- R: Read
- U: Update
- D: Delete

An access permission on occurrences of a MetaClass can take combinations of values:

- **R**: read occurrences of the MetaClass
- **CRU**: create, read and update occurrences of the MetaClass
- **CRUD**: create, read, update and delete occurrences of the MetaClass
- **RU**: read and update occurrences of the MetaClass
- **RUD**: create, read, update and delete occurrences of the MetaClass

MetaAssociationEnd access permissions

By default, the access permission on a MetaAssociationEnd takes value *CRUD :

- C: Connect
- R: Read
- U: Update
- D: Disconnect
- M: Mandatory

A permission on a MetaAssociationEnd can take combinations of values:

- R
- CRU
- CRUD
- RU
- RUD

MetaAttribute access permissions

By default, access permission on a MetaAttribute takes value: *RU.

- R: Read
- U: Update
- M: Mandatory

A permission on a MetaAttribute can take combinations of values:

- R: the MetaAttribute is visible
- RU: the MetaAttribute is visible and modifiable
- RUM: the MetaAttribute is visible, modifiable and mandatory

Permissions on a tool

A tool can be available or not.

By default, availability on a tool is: *A.

The permission on a tool can take value:

- A: the tool is available
- <empty>: the tool is not available

Managing UI Access

– *For information on management of accesses to user interface workflows, see the **HOPEX Power Studio - Workflows** guide.*

The UI access rights (permissions) of a profile are defined by its associated Set of UI access rights.

For a new Set of UI access rights, by default its access permissions on an object are:

- inherited from the access permissions defined on the Set(s) of UI access rights it uses.

  – *See:*

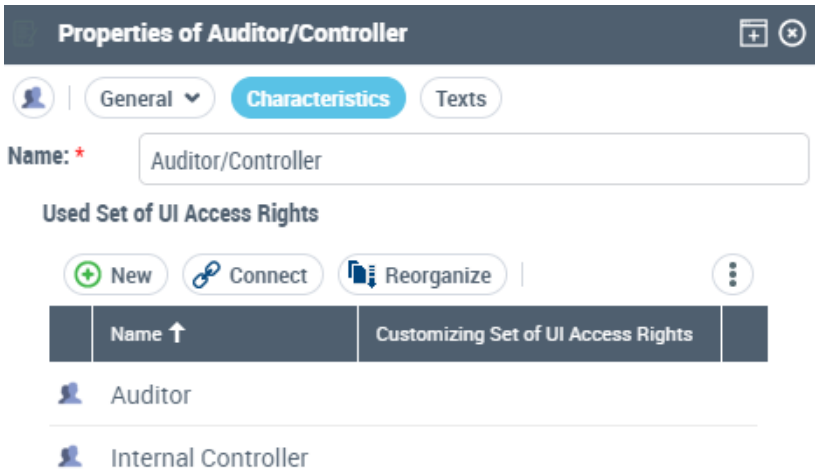
  - Customizing the UI Access (Permissions) of an Existing Profile, and

  - Customizing the Characteristics of an Existing Profile / Creating a Profile from an Existing Profile.

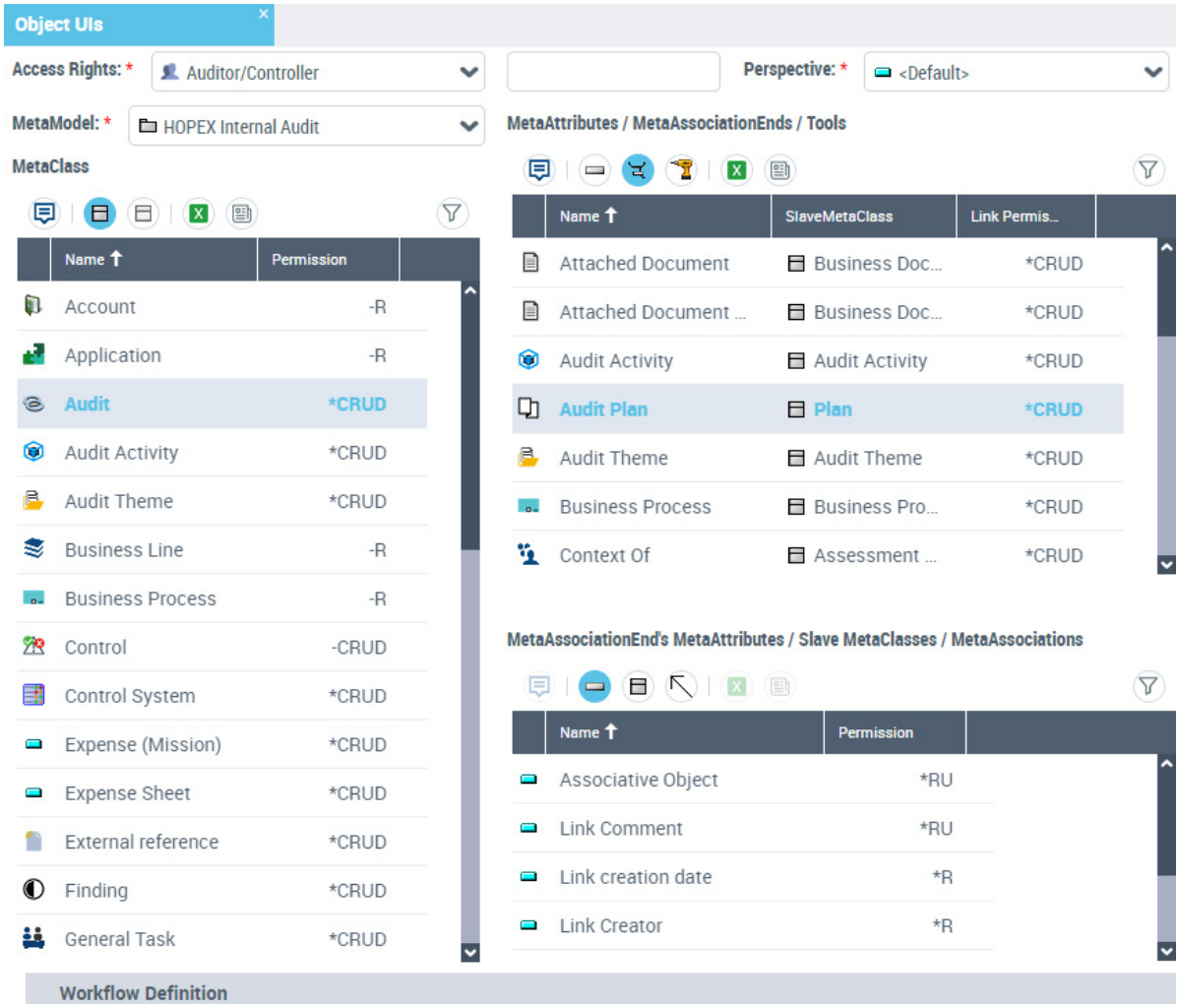
  – ***See Rules on permissions while aggregating Sets of UI access rights.***

  For example the "Auditor/Controller" Set of UI access rights (of the **Auditor/Controller** profile) inherits from

the permissions defined on the "Auditor" and "Internal Controller" Sets of UI access rights.



- Inherited from the permissions defined by default (<HOPEX default>), if it does not use any Set of UI access rights.
  - *See Creating a Profile.*



In the **Object UIs** tab:

- the **Access rights** field enables to select the Set of UI access rights for which you want to view or modify the permissions.
- the **MetaModel** field enables filtering of MetaClasses displayed in the **MetaClass** frame according to the selected MetaModel.
  - "All" value lists all existing MetaClasses.
  - <Extensions> value lists all MetaClasses that are not stored in standard MetaModels (HOPEX products)

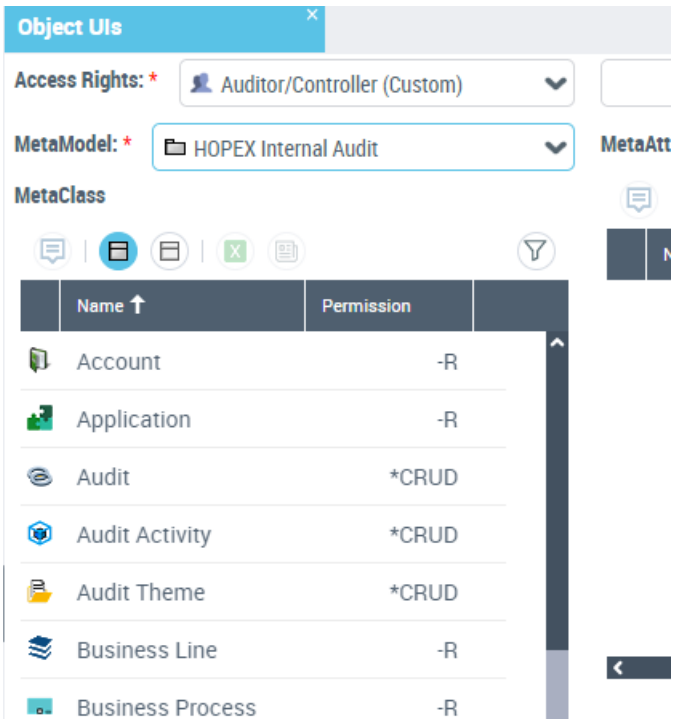
To define access permissions on objects, see:

- Modifying access permissions on occurrences of a MetaClass.
- Modifying access permissions on MetaAttributes of a MetaClass.
- Modifying access permissions on tools of a MetaClass.
- Modifying access permissions of a link around a MetaClass.
- Modifying access permissions on links around a MetaClass.

Modifying access permissions on occurrences of a MetaClass

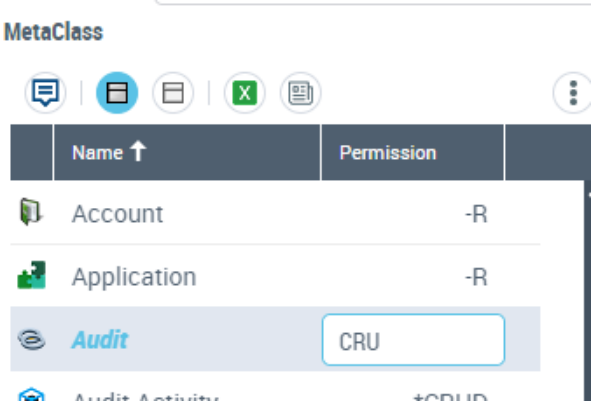
To modify access permissions on occurrences of a MetaClass:

1. Access the UI access management window and select the **Object UIs** tab.
   - *See Introduction to UI Access Management (Permissions).*
2. in the **Access Rights** field, use the drop-down menu to select the Set of UI access rights.
   - *<HOPEX Default> defines default access permissions of each MetaClass, MetaAttribute, MetaAssociationEnd and tool.*
3. In the **MetaModel** field, select the MetaModel concerned.
   In the **MetaClass** frame, the listed MetaClasses are filtered according to the selected MetaModel.



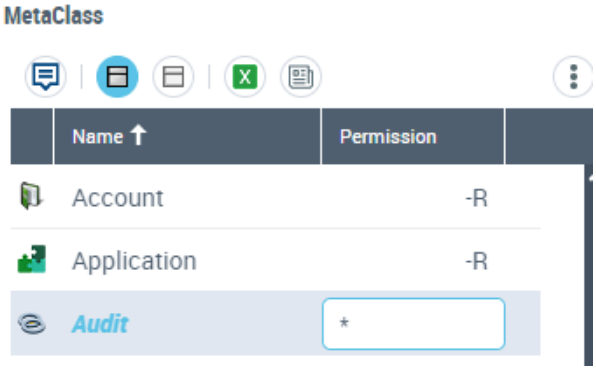
4. In the **MetaClass** frame, select the MetaClass for which you want to modify configuration of access permissions.
   - *By default, its configuration is that inherited from the <Default> profile.*

5. In the **Permission** field, enter the new value.
   - *See MetaClass occurrence access permissions.*

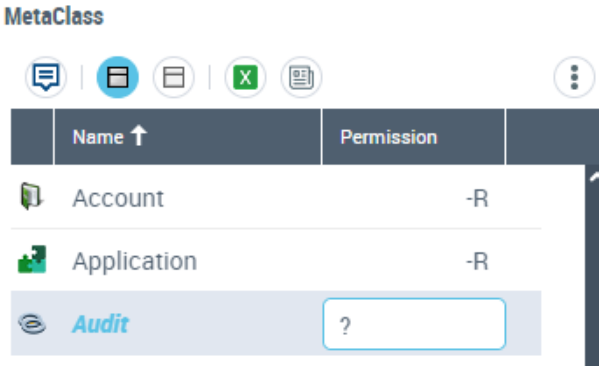


6. Press "Enter".
The value of the MetaClass permission is modified.
In the **MetaAttributes/MetaAssociationEnds/Tools** frame, the values of permissions of elements of the MetaClass are also modified.

   - *To return to the default value of the permission on the MetaClass, enter the character *.*



   - *To obtain information on inheritance of the value, enter the character ?.*



You can also modify the MetaAttributes/MetaAssociationEnds/Tools of a MetaClass, see:

- Modifying access permissions on MetaAttributes of a MetaClass.
- Modifying access permissions on tools of a MetaClass.
- Modifying access permissions of a link around a MetaClass.
- Modifying access permissions on links around a MetaClass.

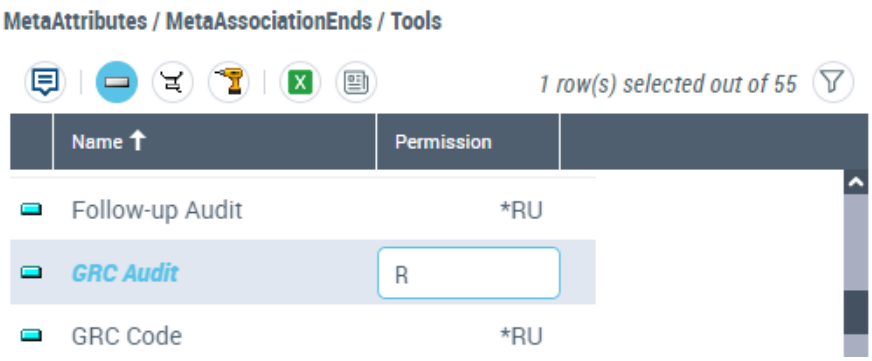
Modifying access permissions on MetaAttributes of a MetaClass

To modify access permissions of MetaAttributes of a MetaClass:

1. Access the UI access management window and select the **Object UIs** tab.

   - *See Introduction to UI Access Management (Permissions).*

2. in the **Access Rights** field, use the drop-down menu to select the Set of UI access rights.
   - *<HOPEX Default> enables to define default access permissions of each MetaClass, MetaAttribute, MetaAssociationEnd and tool.*
3. In the **MetaModel** field, select the MetaModel concerned.
   In the **MetaClass** frame, the listed MetaClasses are filtered according to the selected MetaModel.
4. In the **MetaClass** frame, select the MetaClass concerned.
5. In the toolbar of the **MetaAttributes/MetaAssociationEnds/Tools** frame, click **MetaAttribute** .
   The MetaAttributes of the MetaClass are listed.
6. Select the MetaAttribute for which you want to modify permissions.
7. In the **Permission** field, enter the new value.
   - *See MetaAttribute access permissions.*



8. Press "Enter".
   The value of the MetaAttribute permission is modified.
   - *To return to the default value, enter the character *.*
   - *To obtain information on origin of an inherited value, enter the character ?.*

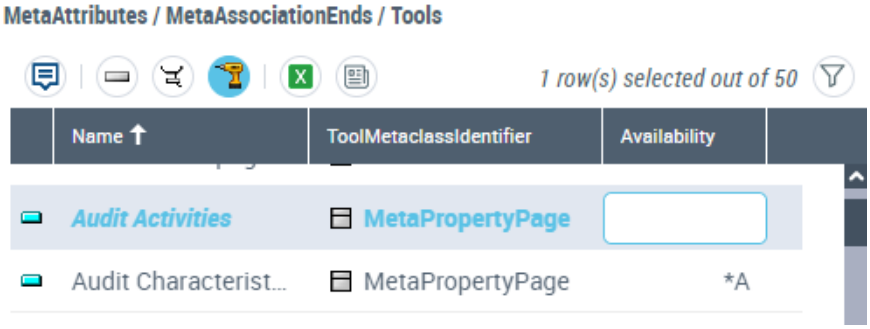
Modifying access permissions on tools of a MetaClass

A tool can be available or not.

To modify access permissions on tools of a MetaClass:

1. Access the UI access management window and select the **Object UIs** tab.
   - *See Introduction to UI Access Management (Permissions).*
2. in the **Access Rights** field, use the drop-down menu to select the Set of UI access rights.
   - *<HOPEX Default> enables to define default access permissions of each MetaClass, MetaAttribute, MetaAssociationEnd and tool.*
3. In the **MetaModel** field, select the MetaModel concerned.
   In the **MetaClass** frame, the listed MetaClasses are filtered according to the selected MetaModel.
4. In the **MetaClass** frame, select the MetaClass concerned.

5. In the toolbar of the **MetaAttributes/MetaAssociationEnds/Tools** frame, click **Tools** .
6. Select the tool for which you want to modify access permissions.
7. In the **Permission** field, enter the new value.
   - *See Permissions on a tool.*



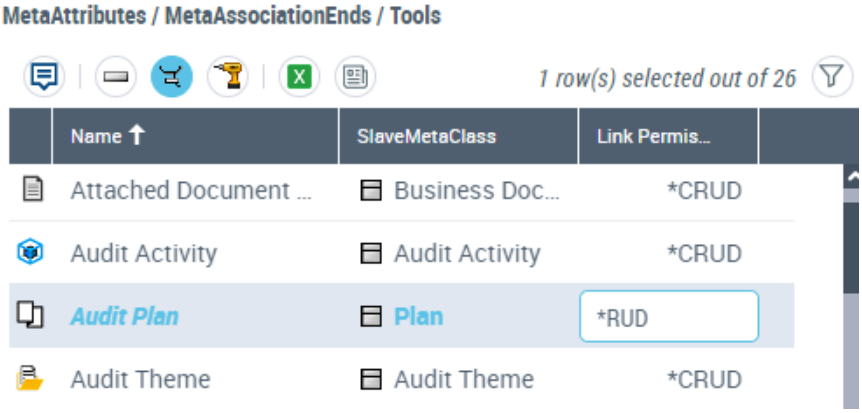
8. Press "Enter".
   The value of the tool access permission is modified.
   - *To return to the default value, enter the character *.*
   - *To obtain information on inheritance of the value, enter the character ?.*

Modifying access permissions of a link around a MetaClass

To modify access permissions of a link around a MetaClass:

1. Access the UI access management dialog box and select the **Object UIs** tab.
   - *See Introduction to UI Access Management (Permissions).*
2. in the **Access Rights** field, use the drop-down menu to select the Set of UI access rights.
   - *<HOPEX Default> enables to define default access permissions of each MetaClass, MetaAttribute, MetaAssociationEnd and tool.*
3. In the **MetaModel** field, select the MetaModel concerned.
   In the **MetaClass** frame, the listed MetaClasses are filtered according to the selected MetaModel.
4. In the **MetaClass** frame, select the MetaClass concerned.
5. In the toolbar of the **MetaAttributes/MetaAssociationEnds/Tools** frame, click **MetaAssociationEnd** .
6. Select the MetaAssociationEnd for which you want to modify link access permissions.

7. In the **Permission** field, enter the new value.
   - *See MetaAssociationEnd access permissions.*



8. Press "Enter".
   The value of the link access permission is modified.
   - *To return to the default value, enter the character *.*
   - *To obtain information on inheritance of the value, enter the character ?.*

   See also Modifying access permissions on links around a MetaClass.

Modifying access permissions on links around a MetaClass

You can modify access permissions on:

- the link according to the MetaClass accessed via the link
- one of the MetaAttributes of the link
- one of the MetaClasses accessed via the link

```
Example: You can grant rights to connect (but not to create)
an IT Service to an Application via this same link.
```

To modify access permissions on links around a MetaClass:

1. Select the MetaAssociationEnd.
   - *See Modifying access permissions of a link around a MetaClass, steps 1 to 6.*
2. In the menu bar of the **MetaAttributes of MetaAssociationEnds/ Slave MetaClasses/MetaAssociations**, click **MetaAttribute**, **MetaClass**, or **MetaAssociation**.
3. In the list, select the MetaAttribute, MetaClass or MetaAssociation concerned.
4. In the **Permission** field, enter the new value.
   - *See MetaAttribute access permissions.*
   - *See MetaClass occurrence access permissions.*

5. Press "Enter".
   The value of the access permission is modified.

   - *To return to the default value, enter the character *.*
   - *To obtain information on origin of an inherited value, enter the character ?.*

Rules on permissions while aggregating Sets of UI access rights

When a **Set of UI access rights** uses one or several Sets of UI access rights, its permissions are defined by the addition of permissions defined on the Sets of UI access rights it uses.

Example:

The Sets of UI access rights S1 and S2 are connected to the Set of UI access rights S3 of the profile P3.

If the permission value on an object A of the Set of UI access rights S1 is CR and the one of the Set of UI access rights S2 is RUD, then this permission value on object A for the Set of UI access rights S3 is CRUD.

Attention to default values

A permission value with * means that this value is the default permission value and that it has not been specifically defined. Only those values specifically defined are taken into account in aggregation.

Example:

The Sets of UI access rights S1 and S2 are connected to the Set of UI access rights S3 of the profile P3.

If the permission value on an object A of the Set of UI access rights S1 is *CRUD and the one of the Set of UI access rights S2 is R, then this permission value on object A for the Set of UI access rights S3 is R.

Managing Data Access Dynamically

Writing and reading access diagrams define data access statically. A person sees objects belonging to his/her reading access area, and can modify objects belonging to his/her writing access area.

- *See Managing Data Writing Access, Managing Data Reading Access.*

You can define dynamic access rules for reading or writing data.

A dynamic rule:

- applies to an object for given profiles
- is defined by a macro

Attention regarding confidentiality management

An object is associated with a confidentiality level and you must be careful while setting up dynamic data access rules.

- ***Static mode***:
  Confidentiality management is taken into account through reading and writing access diagrams, as they both manage data access statically.
- ***Dynamic mode***:
  Confidentiality management might not be always taken into account through data access rules, as they manage data access dynamically.
  When a user generates certain types of documentation (e.g.: Web site, report), this documentation is generated with the data access rules of the person who generates it. Once cached, this documentation might not take into account the confidentiality of the user who will read this documentation (e.g.: Web site, report), which might not follows the same data access rules.

Implementing a dynamic data access rule

– *See Use case: data access rule set up.*

A dynamic data access rule:

- defines for a person, his/her reading or writing access rights on a given object
  – *The rule can be applied to several objects.*
- can be based on characteristics of an object, a person, or an object and a person
- can be called at object creation
- can be associated with one or several profiles
  – *By default the rule is associated with all the profiles.*

To manage dynamic data access on an object, you must implement a permission rule:

1. Create the macro for the permission rule.
   For information on the macro writing, see HOPEX Power Studio > Using APIs: Optimizing the macro of a dynamic data access rule.
2. Create the permission rule.
   – *See Creating a permission rule (data access rule).*
3. (If needed) Define the profile to which the rule applies.
   By default the rule applies to all profiles.
   – *See Associating a permission rule with a profile.*
4. Associate the permission rule with the object concerned by the rule.
   The rule may apply to several objects.
   – *See Associating a permission rule with an object.*

Creating a permission rule (data access rule)

A permission rule is defined by a macro. A permission rule can define reading or writing access rights on an object.

To create a permission rule:

1. In **HOPEX** (Windows Front-End), from the **HOPEX** explorer, click **Create** .
2. Select **Data Access Rule** and click **OK**.
3. In the **Creation of Data Access Rule** dialog box, enter a **Name** for the rule and click **OK**.
4. Access the properties of the rule.
5. In the **Characteristics** tab, in the **Macro** field, click the arrow and connect the macro that manages the rule.
6. In the **Data Access Type** field, select the data access type (**Reading** or **Writing**).
   In the **User Profile** frame, if no profile is connected to the rule, the rule applies to all profiles.
   - *See Associating a permission rule with a profile.*
7. (To call the data access rule at object creation) In the **Texts > _Settings** tab enter:

```
[General]
RelaxCreationTime=0
```

Associating a permission rule with a profile

- *To associate a dynamic permission rule with a profile provided by MEGA, you must have the rights to modify **HOPEX** data, see Managing HOPEX Data Customization.*

To associate a permission rule with a profile:

1. Open the permission rule properties.
   `Example: "Action Plan - Writing"`
2. Click the **Characteristics** tab.
3. In the **User Profile** frame, click **Connect** and select the profile with which you want to associate the permission rule.
   - *You can connect several profiles.*

Associating a permission rule with an object

- *To associate a dynamic permission rule to an object, you must have rights to modify **HOPEX** data, see Managing HOPEX Data Customization.*

To associate a permission rule with an object:

1. Open the object properties.
   `Example: "Risk" MetaClass.`
2. Select the **Data Access** tab.
3. In the **Data Access Rule** frame, click **Connect** and select the rule you want to associate with the object.

Use case: data access rule set up

The same permission rules have been set up for both MetaClasses:

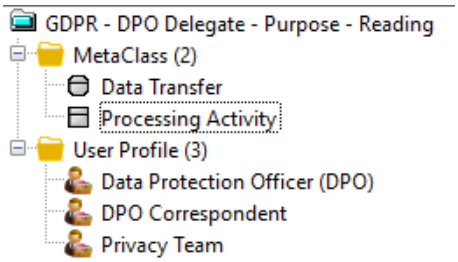
- **Data Transfer**
- **Processing Activity**

The visibility (access rights) of these MetaClasses is customized according to the user profile:

- **Data Protection Officer (DPO)**
  The Data Protection Officer (DPO) works independently to ensure that an entity is adhering to the policies and procedures set forth in the GDPR. He edits processing activities, carries out pre-assessments as well as DPIAs.
- **DPO Correspondent**
  The DPO Correspondent (Privacy) plays the same role as the DPO but his tasks are restricted to a sub-set of the organization.
- **Privacy Team**
  The Privacy Team is made of operational people who carry out the instructions of the DPO or the Chief Privacy Officer.

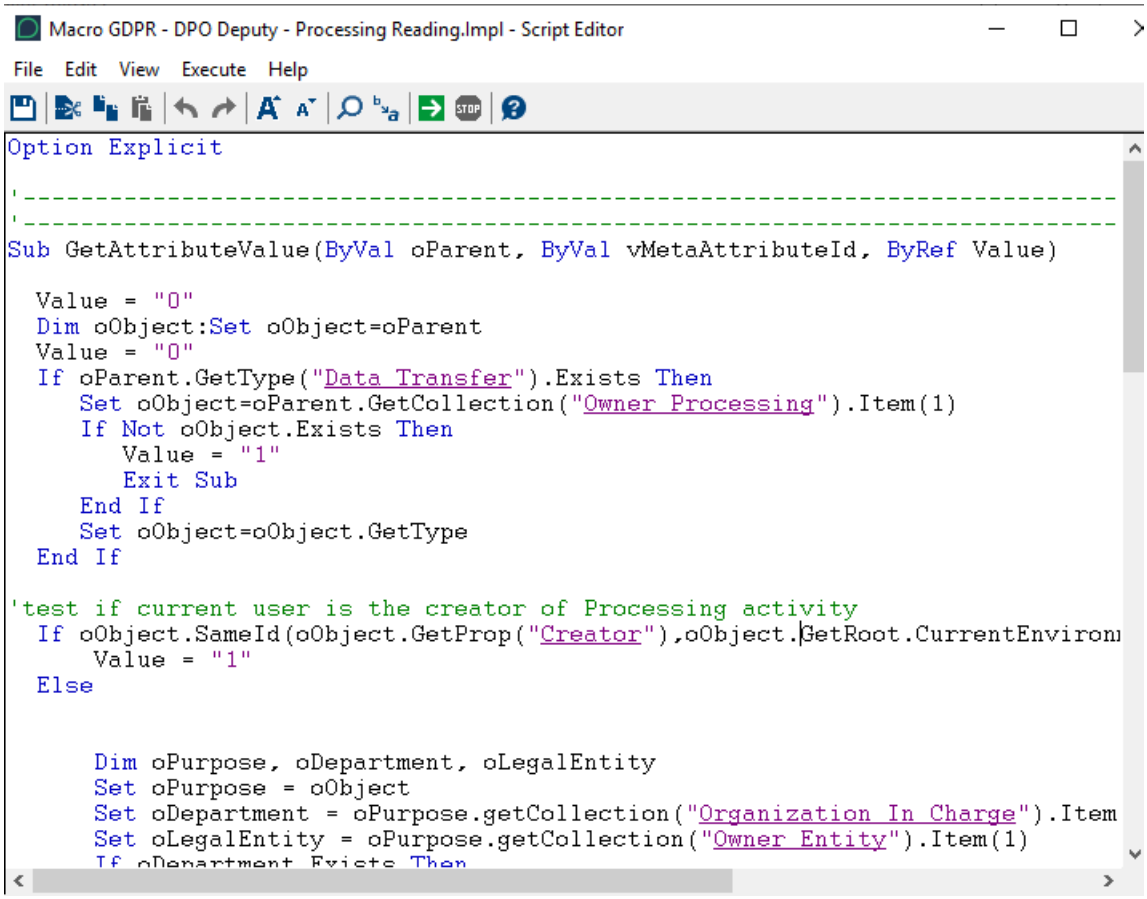
The visibility (access rights) of these MetaClasses is managed through three data access rules.

E.g.: the "GDRP - DPO Delegate - Purpose - Reading" data access rule applies to both **Data Transfer** and **Processing Activity** MetaClasses for **Data Protection Officer (DPO)**, **DPO Correspondent** and **Privacy Team** profiles.

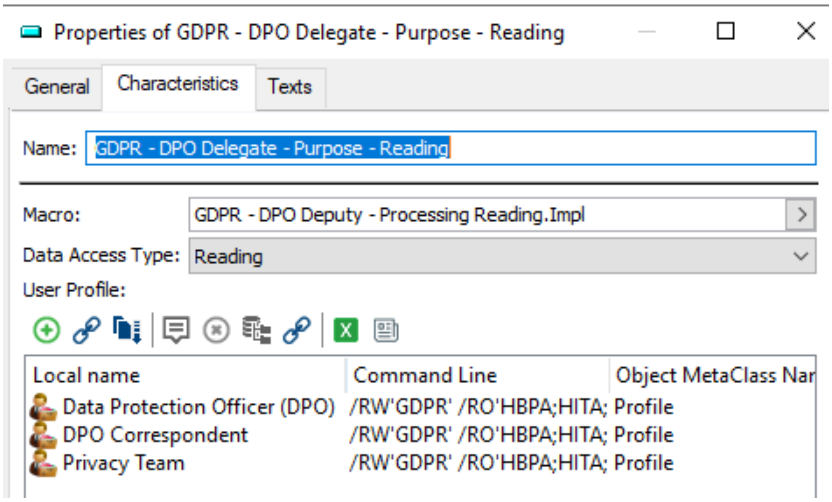


Principle of a permission rule setup on **Data Transfer** and **Processing Activity** MetaClasses:

1. Creation of the macros that manage the rules:
   - GDPR -Activity Owner PrAct - Readig.Implementation
   - GDPR - Purposes -App Owner - ReadingImplementation
   - GDPR - DPO Deputy - Processing Reading.Impl

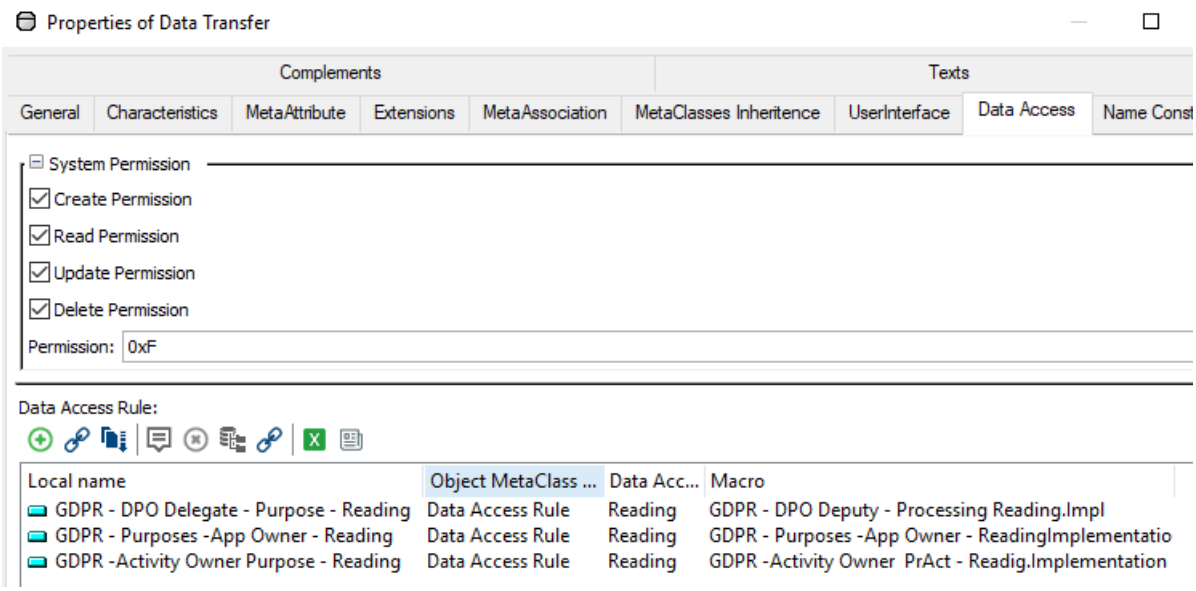


2. Creation of the data access rules associated with each macro:
   - **Data Access Type**: "Reading"
   - **Profiles** associated with the rule: **Data Protection Officer (DPO)**, **DPO Correspondent**, **Privacy Team**.



3. Connecting the data access rules with the **Data Transfer** and **Processing Activity** MetaClasses.

   E.g.: in the **Data Transfer** MetaClass properties, **Data Access** tab, the three rules are connected to the MetaClass.



Generating a Report on Permissions by Profile

A report allows you to generate the detail of permissions for a given profile.

Generating the report

To generate this report:

1. In the menu bar of **HOPEX** (Windows Front-End), select **Tools > Profile and Permission Management > Profile Permissions Report**.
   A wizard opens.
2. (Optional) In the **Name** field, modify the report default name.
3. Select report parameters:
   - In the **MetaModel** field, select the MetaModel concerned.
   - in the **Profile** frame, click the arrow and select the profile concerned.
4. Click **OK**.
   The report is generated as a table.

Report content

All MetaClasses of the selected metamodel appear in the report.

Presented for each MetaClass are:

- in rows: all MetaAttributes, Tools, MetaAssociations (and MetaAttributes of MetaAssociations) of the MetaClass.
- in columns: permissions for all selected profiles.
  - *For improved readability, missing permissions are replaced by _.*
  *For example: *RU is replaced by *_RU_.*

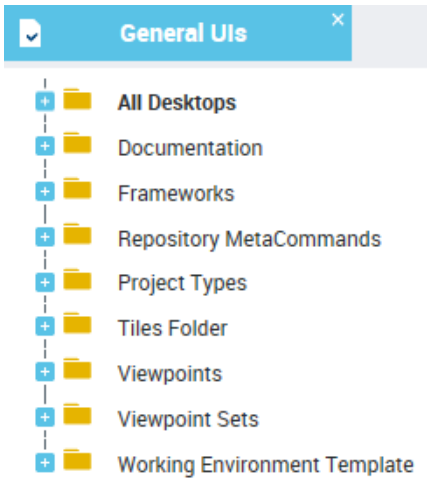
Managing General UI Access

You can manage general UI access for a profile. General UIs are classified by category:

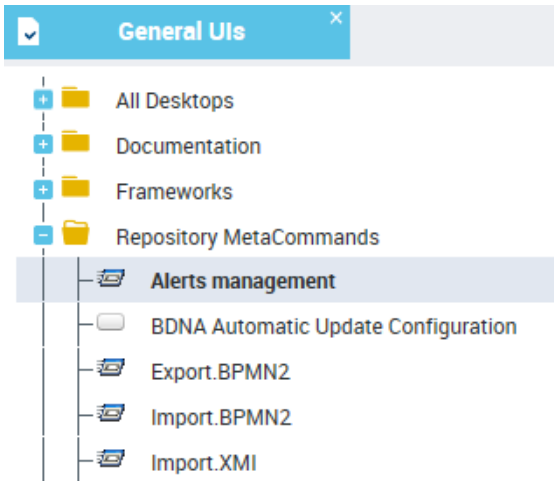
- desktop
- command category
- command group
- general command
- properties page
- tree

To manage general UI access:

1. Access the UI access management dialog box and select the **General UIs** tab.
   - *See Introduction to UI Access Management (Permissions).*

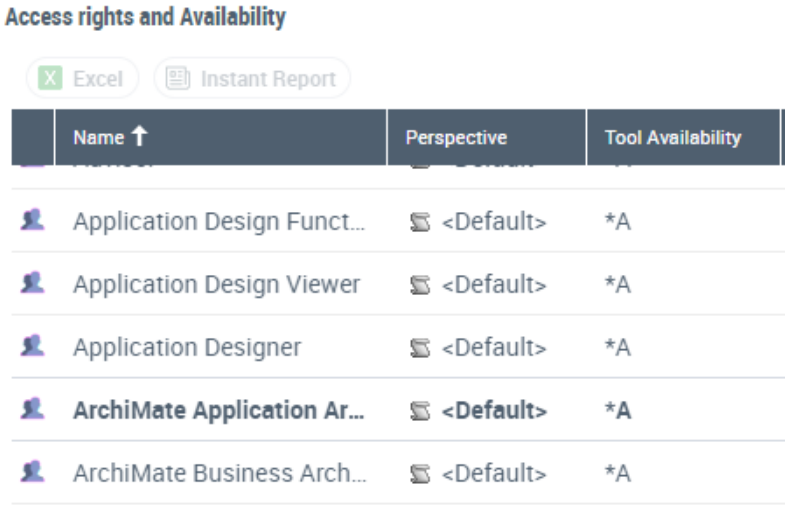


2. In the **SystemDb** tree, expand the folder of the category concerned.
3. In the list, select the tool concerned.



4. In the **Access rights and Availability** pane, select the Set of UI access rights for which you want to modify access on the tool.

5. In the **Tool Availability** field, enter the availability value.



6. Press "Enter".
The value of tool availability is modified.

   - *To return to the tool availability default value, enter the character *.*
   - *To obtain information on origin of an inherited value, enter the character ?.*

MANAGING SHAPES

HOPEX provides several sets of shapes used to represent the various objects in diagrams.

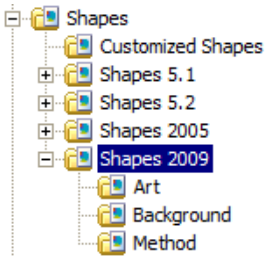
HOPEX Administration, allows you to consult the list of shapes:

- shapes customized by users
- shapes supplied by **MEGA**
  Each set of shapes (5.1 / 5.2 / 2005 / 2009) contains three shape categories:
  - Art
  - Background
  - Method

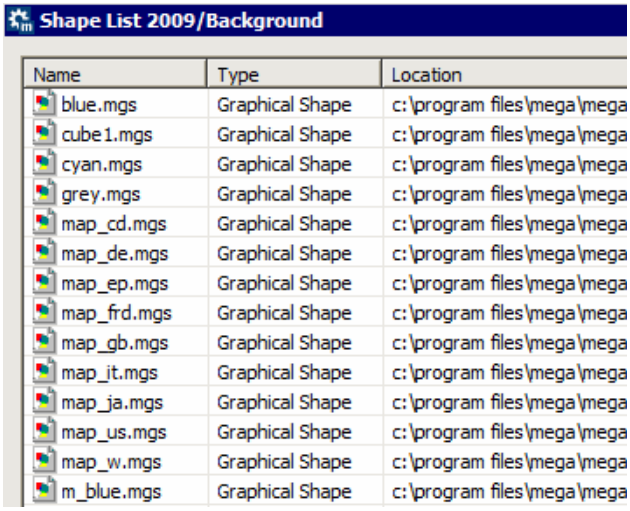
Accessing Shapes

To access shapes:

1. Connect to **HOPEX Administration**
   - *See Accessing HOPEX Administration.*
2. Expand the **Shapes** folder.
3. Expand the sub-folder of the desired shapes.



4. Right-click the category of the desired shapes and select **Manage**.
A table appears listing the images of this category and their location.



MANAGING DATA WRITING ACCESS

HOPEX Administration is provided with tools required for writing access management.

This chapter explains how to create a *writing access diagram* and how to customize its characteristics.

The following points are covered here:

- Introduction to writing access management
- Opening the Writing Access Diagram
- Compiling the Writing Access Diagram
- Defining Writing Access Areas
- Customizing Writing Access Area Management
- Managing Users from the Writing Access Diagram
- Customizing Writing Access Diagram Display

INTRODUCTION TO WRITING ACCESS MANAGEMENT

- *Managing writing access areas is available with the **HOPEX Power Supervisor** technical module only.*

The administrator declares the users and defines the writing access.

The structure of *writing access* is defined in the *writing access diagram*. There is a hierarchical link between writing access.

M *Implementation of this function does not replace a structured management of projects. To operate, the writing access diagram must model itself on organization of projects.*
Clear functional breakdown of your projects simplifies management of operational follow-up of writing accesses.

Users

user

A user is a person with a login and at least one assigned profile.

At creation of an environment, two users are declared by default. These two users have administration rights to manage repositories and users:

- the "Administrator" person, with Login "System" (without password)

  - *The "Administrator" user cannot be deleted. It has no profile (it has all rights). It is recommended that you define a password to restrict use of the "Administrator" user code.*

- The "Mega" person, with Login "mega" (without password)

You must declare other users who will access repositories.

If several environments are defined for a site, users must be defined in each of these environments. To do this, export the user diagram from the reference environment, and import it into each of the other environments.

P **Do not manually create a user with the same name in other environments. If you do, the user will have a different absolute identifier in each environment, and you would actually have created different users with the same name.**

M ***MEGA** recommends that you create the repositories before defining the users, so you can declare the user access rights when they are created.*

To access a repository, a user must identify himself/herself. Then, depending on user writing access area, he/she is able to modify the repository.

User groups (Web Front-End)

A person can belong to one or more groups. A user group is a group of persons with a login.

Persons belonging to a group:

- depend on the same environment.
- share the same connection characteristics defined by the **profile** of the group and its assignment.
- connect to the application with their **login**, but with access rights defined on the **login** of the group.
- share the assignments defined for the group.
- share the personal characteristics defined for the group.

By default at creation of an environment, a group of users is created:

- the "Guests" person group, with Login "Guest".

Writing Access Areas

Each user or group of users is connected to a writing access area. It is the person or person group that carries the writing access area.

Each object is connected to a writing access area.

- *At creation, the object inherits the writing access area of the person who created it.*

MEGA delivers by default the "Administrator" writing access; this writing access area:

- cannot be deleted.
- is the highest writing access area level; it does not depend on any other writing access area. In principle it should be reserved for repository administration.
- is the writing access area to which "Administrator", "Mega" and "Guests" are connected.
  When the writing access diagram has been installed, **MEGA** recommends that you change the writing access area levels of "Mega" and "Guests". It is not desirable that default users have such extensive rights.

All other writing access areas depend on at least one writing access area.

Writing access areas are interconnected by hierarchical links. This is a strict hierarchy, with no circular dependencies: a writing access area cannot be declared at a higher level than the writing access area on which it depends, either directly or via a succession of dependencies.

A user can modify an object connected to his/her writing access area or to a hierarchically lower writing access area.

The writing access area of an object can be modified by the administrator:

- by specifically changing the object writing access area
- when modifying the writing access area of another object (project, process, diagram, etc.) if the propagation option is enabled.

Writing Access Diagram

There is one writing access diagram per environment.

– *If several environments use the same protection configuration, the same user diagram must be used in all these environments.*

Rules

Installation of a writing access diagram diagram must respect the following rules so as to minimize user management costs:

- Any object must be modifiable by users that may need to modify it, without administrator intervention.
- The administrator should intervene only in exceptional circumstances.

Use

The writing access diagram is used similarly to a diagram. The persons, person groups and writing accesses are handled just like standard objects:

- Creation and modification of the name, etc., are done in the same way as for standard objects.
- Be careful however when you delete a writing access.

– *See Deleting Writing Access Areas.*

P **Avoid using the Cut command for a person or person group, as this can result in errors in the writing access diagram if the person or person group is not deleted from the repository or not linked to a writing access.**

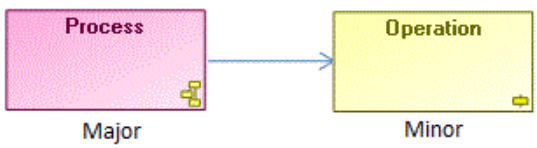
Link Orientation: Major and Minor Objects

When two objects are linked, one object is *major* and the other *minor*.

You cannot delete a minor object if you do not have writing access on the major object.

) *The major object in the link is the one whose nature changes with the presence or absence of the link. For example a process, defined as a succession of operations, is modified if you remove an operation. The process is then major for the link. If the objects are protected, you must have the correct authorization for modifying the major object in order to create or delete the link.*

) *The minor object in a link is the one whose nature is not modified or only slightly modified by presence or absence of this link. For example, removing an operation from a process does not change characteristics of this operation. Therefore the process is minor in the link.*



In the above example, you must have writing access on the org-unit (the major object) to disconnect or delete the message (minor object).

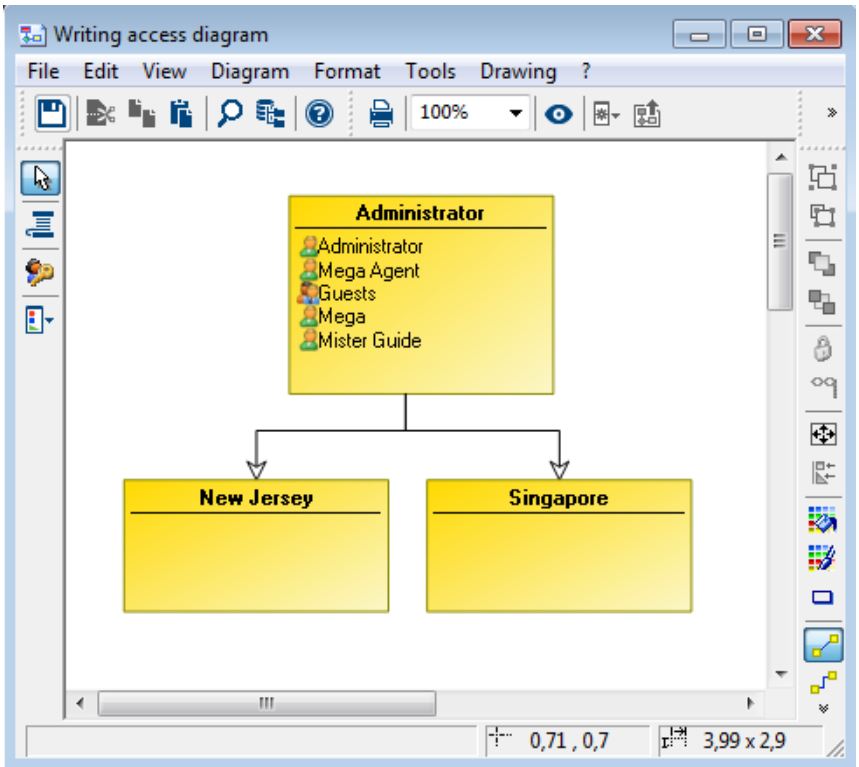
Opening the Writing Access Diagram

- *To access the writing access diagram, you must have a license for the **HOPEX Power Supervisor** technical module.*

Opening the Writing Access Diagram (Windows Front-End)

To open the writing access diagram:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. In the **User Management** folder, right-click **Data Writing Access** and select **Open Diagram**.
   The diagram opens in a new window.



COMPILING THE WRITING ACCESS DIAGRAM

Running writing access diagram compilation assures consistency of behavior of **HOPEX** with declarations of the diagram.

P **If the diagram is not compiled, there is a risk that certain users will be able to update objects that are normally protected.**

When modifying the writing access diagram, so as to warn of rejects due, for example, to writing access restrictions or deletions before compilation it is recommended that:

- all changes made on the user workstations should be uploaded to the administrator workstation, or
- all private workspaces dispatched.

To compile the writing access diagram from the **Administration** application:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.
3. Right-click the **Data Writing Access** folder and select **Compile**.
   When compilation is complete, a message indicates whether the operation was successful or whether the diagram contains errors.
   - *The most frequent errors are:*

   A writing access area (other than '"Administrator") is not attached to any other.

   A person or person group is not attached to any authorization.

DEFINING WRITING ACCESS AREAS

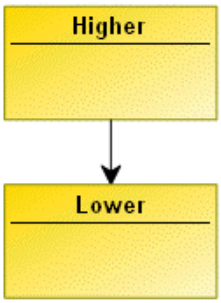
) *The writing access diagram is available if you have the HOPEX Power Supervisor technical module. With this diagram, you can create users and manage their writing access rights for repositories and product functions. By default only one writing access area is defined, named "Administrator". Attached to it are the "Administrator" and "Mega" persons. This is the highest writing access level and it should normally be reserved for repository management. You cannot delete this writing access area.*

The *writing access diagram* enables creation of writing access areas.

Creating a Writing Access Area

To create a writing access area:

1. In the diagram insert toolbar, click **Writing Access Area** then click in the diagram.
2. Enter the name of the writing access area.
   Dependency of writing access areas is determined by creation of a "lower" link which starts from the higher writing access area to the lower writing access area.



Defining Writing Access Area Persons or Person Groups

To define persons or person groups of a writing access area:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. Right-click the writing access area and select **Properties**.
3. Select the **Users** tab and click **Connect**.

4. In the query tool, click the arrow in the first field and select the target (*Access Area Member*, Person or Person Group).

) *Access area member groups all persons and person groups belonging to an access area. This area defines objects that can be accessed by the person or person group.*

5. (optional) In the second field, enter the character string to be queried.
6. Click **Find**.
7. In the results list, select the required access area member and click **Connect**.

   - *Press the [CTRL] key to select several members simultaneously.*

   The person or person group you have connected appears in the list of access area members of the selected writing access area.

Defining a Writing Access Area at Creation

To assign to all objects created by a user a writing access area different from the writing access area of this user, you must associate a writing access area at creation with the user concerned.

To define a writing access area at creation of a user:

1. Open the properties dialog box of the person.
2. Select the **Characteristics** tab.
3. In the **Writing Access Area at Creation** field, select the required writing access.

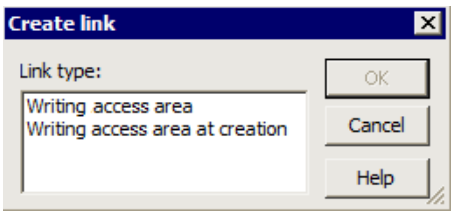
   - *See Person writing access area and writing access area at creation.*

To define a writing access area at creation of a user:

1. From **HOPEX Administration**, open the writing access diagram.

   - *See Opening the Writing Access Diagram.*

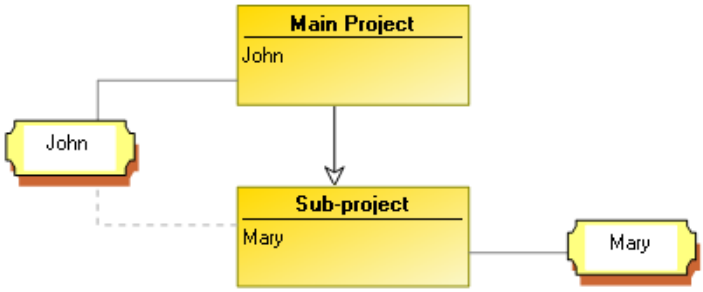
2. If this is not already done, place the person and the writing access area concerned in the diagram.
3. Draw a link between the person and the required writing access area. The **Create Link** dialog box opens.



4. Select **Writing Access Area at Creation** and click **OK**.

   - *Value "None" of **Writing Access Area at Creation** signifies that the user creates objects in the same writing access area as that to which he/she belongs.*

When the writing access area at creation has been created, it is represented by a dotted line link between the person and the writing access area in the writing access diagram.



In the above example, user John has:

- a writing access area of level "main project"
- a writing access area at creation of level "sub-project"

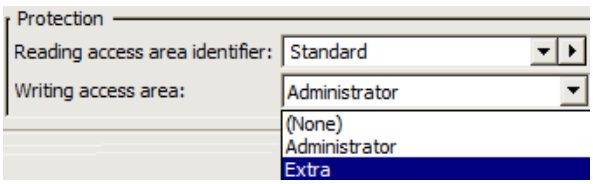
Objects created by John can therefore be modified by Mary.

Modifying Writing Access Areas of Objects

If you have a writing access area of level higher than or equal to that of an object, you can modify the writing access area of this object in the object properties dialog box.

To modify writing access area of an object:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In your **HOPEX** desktop, open the object properties window, select the **General** tab, then the **Administration** sub-tab.
3. In the **Protection** pane, in the **Writing Access Area** field, select a writing access area via the drop-down menu.



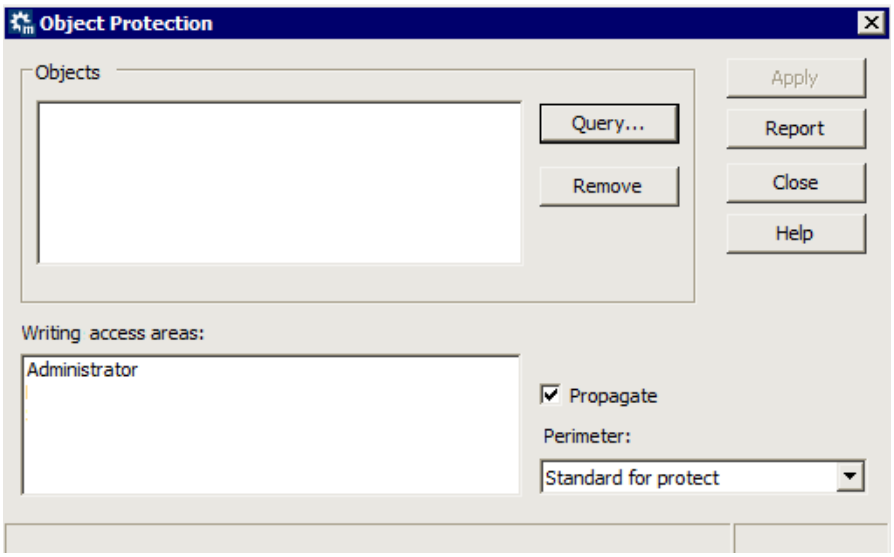
4. Click **OK**.

Modifying Writing Access Areas of an Object Group

If you have a writing access area of level higher than or equal to that of an object group, you can modify the writing access area of this object group.

To modify the writing access area of an object group:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In your **HOPEX** workspace, select **Tools > Manage > Protect Objects**. The **Protect Objects** dialog box opens.
3. In the **Objects** pane, click **Query** and select the object group.
4. In the **Writing Access Areas** pane, select the writing access area you want to assign to the object group.
5. (Optional) Select **Propagate** if you want the writing access area to be propagated to all dependent objects of the object group, as a function of the perimeter selected.



6. Click **Apply**.
   Object protection is applied.
7. Click **Report** to check if a conflict has been encountered at protection propagation in the repository.
   Conflicts may arise, especially when **Propagate** is selected and that some children objects already have a defined writing access area (through a direct link).
   To solve this conflict, on each object in conflict, you must manually delete the link between the object and the writing acces area, to define the writing access area you want to keep it.
   - *See Associating Objects with Writing Access Areas.*

Deleting Writing Access Areas

To delete a writing access area:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*

2. Open the properties dialog box of the writing access area concerned.
3. Select the **Users** tab and disconnect all **Access Area Members** of the writing access area, and connect these **Access Area Members** to another writing access area.
4. Delete the writing access area.

The writing access areas dependent on the deleted writing access area are, after updating, no longer attached to the writing access areas tree. It is therefore preferable to first delete their links with the obsolete writing access area and attach them to a writing access area that will be retained.

Objects which had this writing access area can be protected with another writing access area. Otherwise, they are considered as being protected at the highest level, with "Administrator" writing access area level.

– *For more details on protection of objects, see Protecting Objects.*

Propagating Object Writing Access Areas to Child Objects

The **Administration** navigation window of the **HOPEX** workspace allows:

- access to writing access areas
- simple automation of writing access area propagation to connected and child objects.

  – *See Associating Objects with Writing Access Areas.*
- connecting an object to a writing access area

You can propagate a writing access area from all objects connected to dependent objects, for a repository of the environment.

– *This action can take some considerable time, depending on repository size.*

To propagate a writing access area:

1. Connect to **HOPEX**.

   – *See Connecting to HOPEX.*
2. In your **HOPEX** workspace menu bar, select **View > Navigation Windows > Administration**.
3. In the **Navigation** tab, expand the **Writing Access Area** folder.

   – *If you do not see the Writing Access Areas, check that your metamodel access is at least "Advanced" level (**Options > Repository**).*
4. Right-click the writing access area to be propagated and select **Propagation of writing access area to associated occurrences**.
5. Click **Yes** to confirm.

Associating Objects with Writing Access Areas

The **Administration** navigation window of the **HOPEX** workspace allows:

- access to writing access areas
- simple automation of writing access area propagation to connected and child objects.
- connecting an object to a writing access area

  – *See Propagating Object Writing Access Areas to Child Objects.*

To connect an object to a writing access area:

1. Connect to **HOPEX**.

   – *See Connecting to HOPEX.*
2. In your **HOPEX** workspace menu bar, select **View > Navigation Windows > Administration**.
3. In the **Navigation** tab, expand the **Writing Access Area** folder.

   – *If you do not see the Writing Access Areas, check that your metamodel access is at least "Advanced" level (**Options > Repository**).*
4. Right-click the required writing access area and select **Connect > Object**.
5. In the query dialog box, find the required object and click **OK**.

To display the list of objects associated with a writing access area:

› Right-click the writing access area and select **Objects associated with writing access area**.
A dialog box displays a list of these objects.

Tips on Using Writing Access Areas

Common data

MEGA recommends that you manage data common to several projects in a specific project. This simplifies control of their evolutions.

Tips

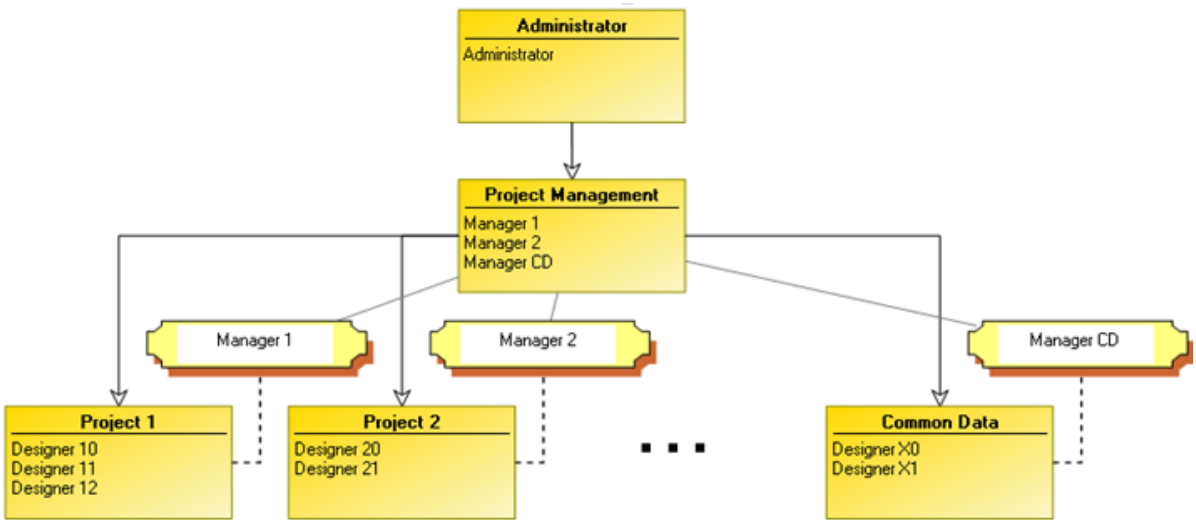
MEGA recommends:

- definition of certain users:
  - with writing access area level higher than projects so as to manage conflicts between projects, for example "Project Management".
  - with writing access area at creation at the level of the project, to avoid creation of objects that cannot be modified by the project.
    - *See Defining a Writing Access Area at Creation.*
- that only the "Administrator" user should be connected to the "Administrator" writing access area.
- that if a person produces on several projects, the person should have one **HOPEX** user per project.
  Objects are therefore directly created in the correct owner project, greatly simplifying management.

Typical example

The following example presents a typical case of writing access area use:

- Only the Administrator user has "Administrator" writing access area level.
- All managers can modify objects of all projects.
- Objects created by a manager are attached to a dedicated project.
  - *Manager 1 can modify objects of all projects, by default the objects he/she creates are in project 1.*
- Data common to different projects ("Common Data") is managed in a dedicated project with a specific writing access area.



CUSTOMIZING WRITING ACCESS AREA MANAGEMENT

This section describes how to use and customize management of writing access areas:

- Calculated Writing Access Area
- Calculated MetaAttribute
- Installing a Writing Access Diagram
- Locking Validated Objects
- Merging Two Projects
- Splitting a Project

Calculated Writing Access Area

As standard, the writing access area of an object is stored in the "_Authorization" MetaAttribute and takes the value of the writing access area absolute identifier. It is assigned at creation and you can modify it.

You can install up calculated writing access area.

```
For example, you can deduce the writing access area of an
operation from that of the process on which it depends. You
need only change the writing access area of a process, and
those of the dependent operations will automatically adapt.
```

P **In this case, watch performance.**

To customize the writing access area of an object:

› Replace the "_Authorization" MetaAttribute (which carries the object writing access area) by a calculated MetaAttribute.

Calculated MetaAttribute

A calculated MetaAttribute is a software device enabling deduction of the MetaAttribute value of an object as a function of data around the object or dependent on other sources (system, current user, etc.).

HOPEX uses a set of "conventional" MetaAttributes (including the writing access area) that do not require metamodel definition.

A substitution device is available in **HOPEX**; it enables replacement of an implicit MetaAttribute by another for a MetaClass.

This device is required when you need to alter behavior of an existing MetaAttribute by implementing a calculated MetaAttribute.

To customize writing access area of an object, you must:

1. Create a MetaAttribute with characteristics close to those of the "_Authorization" MetaAttribute.

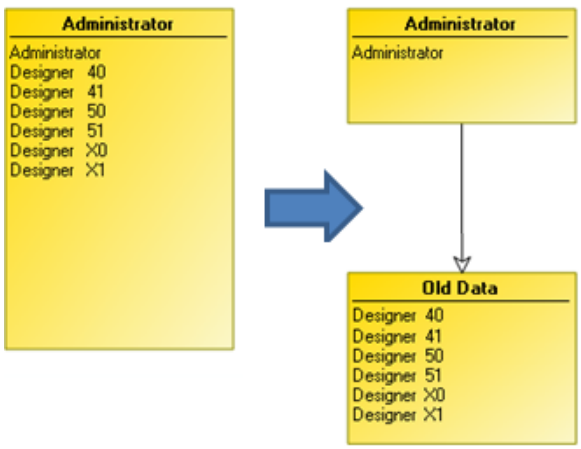
2. Substitute the "_HexaIdAbs" value of the new MetaAttribute by the "_HexaIdAbs" value of the "_Authorization" MetaAttribute.
3. Calculate the writing access area.

Installing a Writing Access Diagram

To install a writing access area diagram in an environment already in production:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. Ask users who have a private workspace in progress to close this.
3. Create an "Old Data" writing access area
   - *See Creating a Writing Access Area.*
4. Attach all users (except the "Administrator" user) and all objects of all repositories to the "Old Data" writing access area.
   - *See Defining Writing Access Area Persons or Person Groups.*
   - *See Associating Objects with Writing Access Areas.*

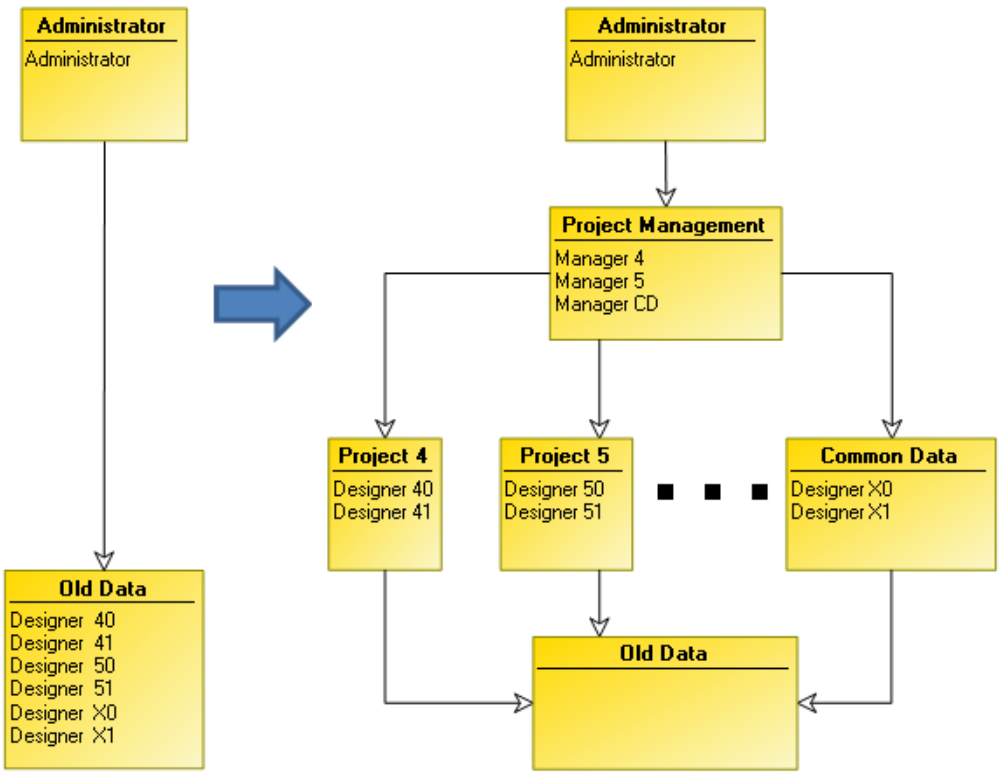
   Users may resume working.



5. Create new writing access areas according to projects.
6. Distribute users between these writing access areas.
7. Distribute objects between these projects/writing access areas for all repositories of the environment.
   - *See Associating Objects with Writing Access Areas.*

   P **Until this distribution is completed, projects can interfere with each other, since they have rights to modify objects created before distribution.**

8. (Optional) When all objects of all repositories have been distributed, you can delete the "Old Data" writing access area.



– *If the environment is new and there is no data to be distributed between the new writing access areas, you do not need to draw the diagram with the "Old Data" writing access area.*

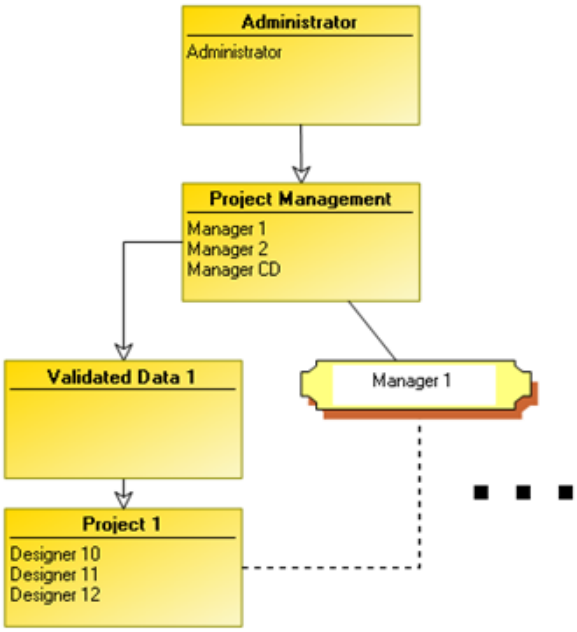
Locking Validated Objects

When objects have been validated, you can configure the writing access diagram so that these objects cannot be modified.

To lock objects:

1. From **HOPEX Administration**, open the writing access diagram.
   – *See Opening the Writing Access Diagram.*

2. Between the writing access area "Project Management" and that of the project, insert a writing access area dedicated to validated data of the project.



3. When data is validated, modify its writing access area from "Project" level to the higher level "Validated Data" writing access area.
   - *See Modifying Writing Access Areas of Objects.*
4. (Optional) If validated data must be modified:
   - it is modified by a user of "Project Management" writing access area level.
   - it is lowered to the "Project" writing access area level.

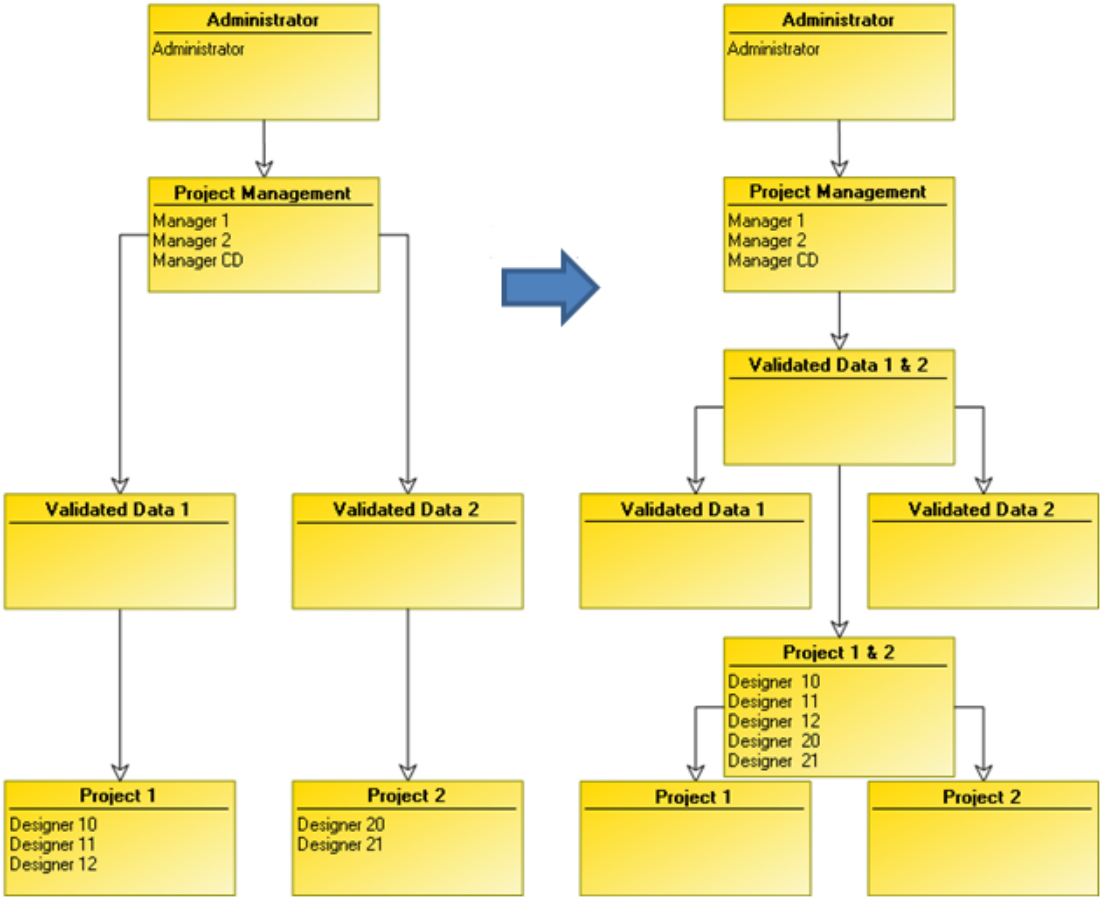
So the project perimeter is distributed between two writing access areas, but remains perfectly determined.

Merging Two Projects

To merge two projects:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. Create a new writing access area for the new project.
   - *This new writing access area must be of higher level than the writing access area it will replace.*

3. Connect users of merged projects to this new writing access area.
   - *See Defining Writing Access Area Persons or Person Groups.*



Splitting a Project

To split a project:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. Create writing access areas connected to new projects.
   These writing access areas must be of must be of higher level than the writing access areas they will replace.
   - *See Creating a Writing Access Area.*
3. Ask users who have a private workspace in progress to close this.
4. Distribute users between these writing access areas.
   - *See Defining Writing Access Area Persons or Person Groups.*
   - *Users may resume working.*

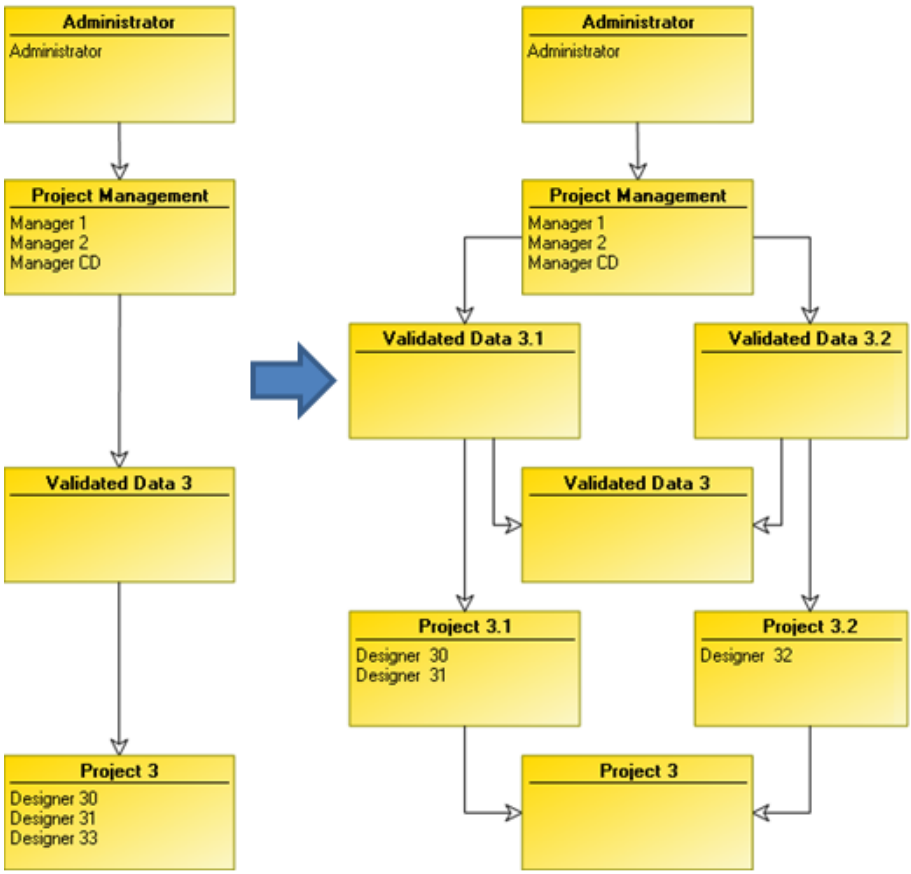
5. Distribute the objects between the new projects.

   - *See Associating Objects with Writing Access Areas.*

   P **Until this distribution is completed, projects can interfere with each other, since they do not yet have rights to modify objects created before distribution.**

6. (Optional) Delete the old writing access areas.

   - *(Optional) When all objects of all repositories have been distributed, you can delete old writing access areas.*



MANAGING USERS FROM THE WRITING ACCESS DIAGRAM

This section presents how to:

- Creating Persons with Writing Access Areas
- Creating Person Group with Writing Access Areas
- Managing Users from the Writing Access Diagram
- Compiling the Writing Access Diagram
- Transferring the Writing Access Diagram

Creating Persons with Writing Access Areas

At creation, the user is not connected to any writing access area. To implement protection, the person should be connected to a writing access area by creation of a link between person and writing access area.

To create a person with a writing access area, from the writing access diagram:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. In the writing access diagram, right-click a writing access area and select **Properties**.
3. In the **Users** tab, click the **New** button .
   A selection dialog box appears.
4. In the **MetaClass** field, select **Person**.
   The **Creation of Person** dialog box opens.
5. Follow the procedure Defining a Person.
   The person is created and connected to the selected writing access area.
   - *A person depends on a single writing access area.*

Creating Person Group with Writing Access Areas

To create a person group with a writing access area, from the writing access diagram:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. In the writing access diagram, right-click a writing access area and select **Properties**.
3. In the **Users** tab, click the **New** button .
   A selection dialog box appears.
4. In the **MetaClass** field, select **Person Group**.
   A new person group appears in the list of access area members. The new group is connected to the selected writing access area.

5. (optional) Modify the **Short Name** of the new person group.

Managing Users from the Writing Access Diagram

User access rights to repositories and functions can be restricted by the administrator. You can carry out this modification user by user, or on all users simultaneously.

– *To manage user access rights, the **HOPEX Power Supervisor** technical module is required.*

To manage users from the writing access diagram:

1. From **HOPEX Administration**, open the writing access diagram.
   – *See Opening the Writing Access Diagram.*
2. From the writing access diagram, select **Diagram > Described Object > Manage Users**.
   The **Users** administration dialog box opens.
   – *See Introduction to User Management.*

Compiling the Writing Access Diagram

Changes to the writing access diagram only take effect after it has been compiled.

To compile the diagram:

› Select **Diagram > Described Object > Compile Writing Access Diagram**.

If the diagram has been modified, it is automatically compiled at closing. This allows you to check the validity of the user diagram.

When modifying the writing access diagram, so as to warn of rejects due, for example, to authorization restrictions or deletions before compilation it is recommended that:

- all changes made on the user workstations should be uploaded to the administrator workstation, or
- all private workspaces dispatched.

When compilation is complete, a message indicates whether the operation was successful or whether the diagram contains errors. The most frequent errors are:

- A writing access area (other than '"Administrator") is not attached to any other.
- A person or person group is not attached to any authorization.

Transferring the Writing Access Diagram

On workstations where the network is not available, when the user diagram has been updated and compiled it must be exported to the administrator workstation and imported on user workstations. You must also do this if the same diagram is to be used in several different environments.

To run export:

› Select **Diagram > Described Object > Export Writing Access Diagram**.

Import on user workstations is carried out in the normal way.

CUSTOMIZING WRITING ACCESS DIAGRAM DISPLAY

You can customize writing access diagram display:

- diagram structure representation
- display of persons connected to a writing access area

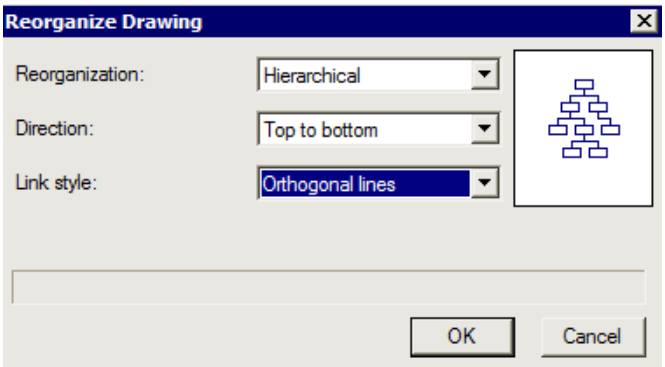
Customizing Diagram Structure Representation

You can customize representation of the structure of the writing access diagram using the drawing reorganization function.

M *The automatic drawing reorganization functionality is automatically activated on loading a diagram that does not yet include a drawing.*

To modify organization of an existing drawing:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. Select **Drawing > Reorganize Drawing**.
3. Select the desired reorganization mode, the direction and the style of links in the diagram.



M *The miniature image alongside the reorganization options gives you a view of each type of reorganization.*

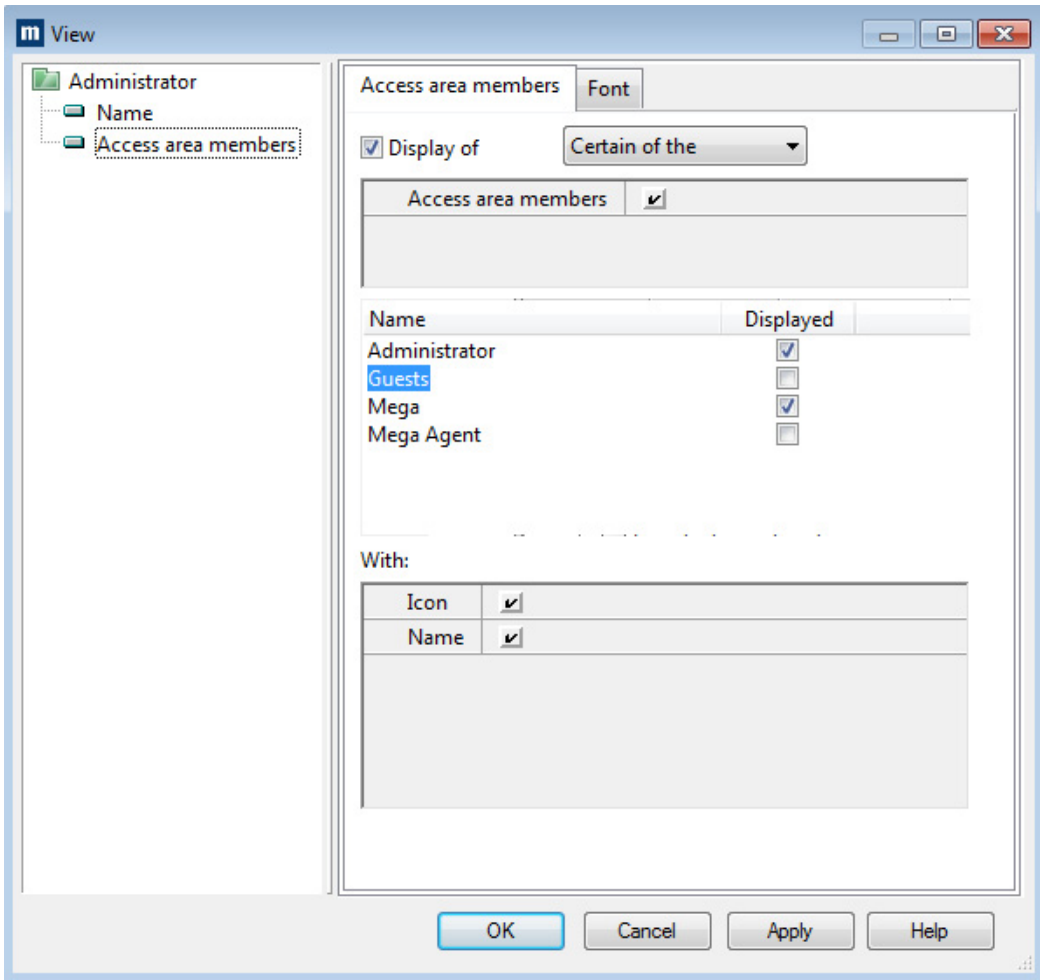
4. Click **OK** to apply the modifications.

Customizing Writing Access Area Display

You can show or hide persons, person groups, objects connected to a writing access area.

To define writing access area display:

1. From **HOPEX Administration**, open the writing access diagram.
   - *See Opening the Writing Access Diagram.*
2. Right-click the writing access area concerned and select **Shapes and Details**.
   The **View** dialog box opens.
3. In the tree on the left, select **Access Area Members**.
4. In the pane on the right, select **Access Area Members**.
5. Select the **Display of** option, then click the arrow and select **Certain of the** access area members.
   - *To display all or none of the access area members, select **All the** or **None of the**.*
6. Select the persons you want to see displayed.



7. Click **OK**.

MANAGING DATA READING ACCESS

The following points are covered here:

- Introduction
- Reading Access Area Matrix
- Reading Access Diagram
- Configuring Data Reading Access
- MetaClass Confidentiality Exceptions

INTRODUCTION

– *Managing reading access areas is only available with the **HOPEX Power Supervisor** technical module.*

The following points are detailed here:

- The Need to Manage Sensitive Data
- General Concepts
- Activating Data Reading Access Management
- Consulting Environment Reading Access Information
- Managing Reading Access in HOPEX
- Compiling the Reading Access Diagram

The Need to Manage Sensitive Data

Certain modeling projects may be confidential or contain confidential or sensitive data: costs, risks, controls.

The **HOPEX** administrator may therefore need to mask objects corresponding to confidential or sensitive data.

These objects must be visible only to authorized users.

To meet these requirements concerning data confidentiality, **HOPEX** offers functionalities for implementing consistent and effective confidentiality policies.

General Concepts

To implement a data confidentiality policy, objects must be organized in distinct sets. Each set of objects is a ***reading access areas***.

) *A user is a person with a login.*

) *A person can belong to a group. A user group is a group of persons with a login.*

Each user or group of users is associated with a reading access area. It is the person or person group that carries the reading access area.

The reading access area to which the person or person group belongs determines the objects that the user or group of users can see. A user or user group can only see objects located in his/her own or lower confidentiality areas.

P **With definition of reading access areas, hidden objects are inaccessible. This concept differs from that of the filter, which hides occurrences of MetaClasses so as not to disturb the final view of the user, see Managing UI Access (Permissions).**

Activating Data Reading Access Management

When you activate reading access management, confidential data is visible only to authorized users. Before activating reading access management, **MEGA** recommends that you familiarize yourself with reading access management using **MEGA**.

> – *For more details on confidential data, see Confidential or Sensitive Object Behavior.*

To manage confidential or sensitive data , you must first activate data reading access area management.

To activate data reading access management:

1. Connect to **HOPEX Administration** and select the environment concerned.
   > – *See Connecting to an Environment.*
2. In the **User Management** folder, right-click **Data Reading Access** and select **Activate Management**.
   > P **A message warns you that activation of reading access management is irreversible.**
3. Carefully read this warning message, then click **Yes** if you wish to activate data reading access management.

Consulting Environment Reading Access Information

When working in **HOPEX** you can check:

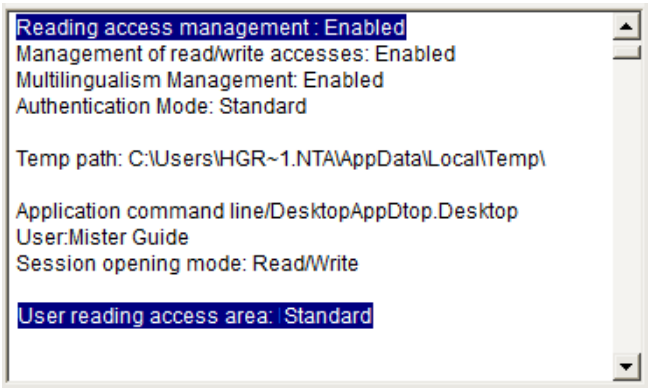
- if reading access management is activated in your environment or not.
- the reading access area to which the connected user belongs.

To consult reading access information of your current environment:

1. Connect to **HOPEX**.
   > – *See Connecting to HOPEX.*
2. In the **HOPEX** menu bar, select **Help > About HOPEX**.
   The **About HOPEX** dialog box appears.
3. Click **System Information**.
4. In the menu bar of the **System Properties** dialog box, select **System Info > Edit**.
   The Megasys.txt text file opens.

At the beginning of the file you can consult the properties of:

- ***Reading access management***
- ***Reading access area of the user***



Managing Reading Access in HOPEX

You can set up and manage data reading access in **HOPEX** in two ways:

Reading access area matrix method:

1. Create the different user reading access areas you require.
2. Distribute persons or person groups in user reading access areas.
3. Distribute objects in object reading access areas.
4. Associate user reading access areas with object reading access areas.

   - *For more details, see Reading Access Area Matrix.*

Reading access diagram method:

1. Define organization and hierarchy of the different reading access areas you require.
2. Create and organize these users in a reading access diagram.
3. Associate persons or person groups with different reading access areas. Objects created by users are then distributed in the user reading access area.

   - *For more details, see Reading Access Diagram.*

Compiling the Reading Access Diagram

Running the reading access diagram compilation ensures consistent **HOPEX** behavior with diagram declarations.

P **If the diagram is not compiled, there is a risk that certain users will be able to see objects that are normally hidden.**

To compile the reading access diagram:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.
3. Right-click the user **Data Reading Access** folder and select **Compile**.
   On completion of compilation, a message indicates the result of the operation.

READING ACCESS AREA MATRIX

The reading access area matrix enables organization of user groups with object groups. Only a user with administrator profile connected to the maximum reading access area can configure reading access areas.

In the reading access area matrix, you can create two types of reading access areas:

- an **object reading access area**  , grouping only **HOPEX** objects.
- a **user reading access area**  , grouping only persons or person groups

) *The user reading access area corresponds to the view the person or person group has of the repository: it defines objects that can be accessed by the person or person group.*

P **To ensure coherence of the reading access diagram, if you begin managing reading access of your data from the reading access area matrix, you must continue to manage reading access from this matrix.**

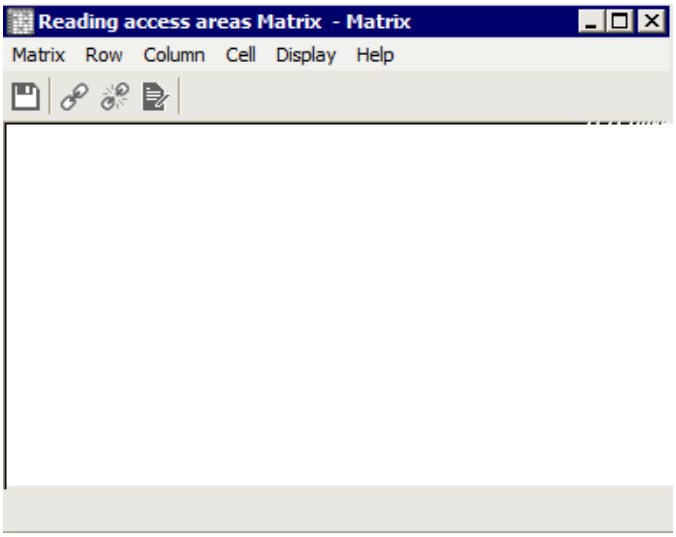
You can create links between these two types of reading access area.

Accessing the Reading Access Area Matrix

To access the reading access area matrix:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.

3. Right-click the user **Data Reading Access** folder and select **Reading Access Matrix**.
   An empty reading access area matrix appears.



Adding an Object Reading Access Area

To add an object reading access area in the matrix:

1. Open the reading access area matrix.
   - *See Accessing the Reading Access Area Matrix.*
2. Select **Row > Create**.
3. In the creation window, enter the name of the object reading access area and click **Finish**.

Adding a User Reading Access Area

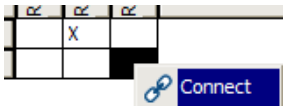
To add an user reading access area in the matrix:

1. Open the reading access area matrix.
   - *See Accessing the Reading Access Area Matrix.*
2. Select **Column > Create**.
3. In the creation window, enter the name of the user reading access area and click **Finish**.

Associating User Reading Access Areas with Object Reading Access Areas

To associate a user reading access area with an object reading access area

1. Open the reading access area matrix.
   - *See Accessing the Reading Access Area Matrix.*
2. In the reading access area matrix, right-click the cell at the intersection of the user reading access area and the object reading access area and select **Connect**.



A cross represents the association between the two selected reading access areas. The corresponding links are automatically drawn in the reading access diagram.

If you begin management of reading access of your data from the reading access area matrix, you must continue management of reading access from this matrix. Do not manually modify links created automatically in the reading access diagram; you may invalidate the diagram.

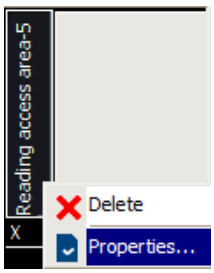
- *For more details on the reading access diagram, see Reading Access Diagram.*

Associating Users with User Reading Access Areas

In the case of the reading access area matrix, to associate a user (or user group) with a reading access area:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.
3. Right-click the user **Data Reading Access** folder and select **Reading Access Matrix**.
   The reading access area matrix appears.

4. Right-click the user reading access area concerned and select **Properties**.



The user reading access area properties window appears.

5. In the **Persons** tab, click **Connect**.

   – *If you want to create a new Person (or new person group) and associate it with the reading access area, click **New**.*

6. In the query tool, click the arrow in the first field and select **Person** (or **Person Group**).

   – *If you want to select persons and person groups, select Access Area Member.*

7. (optional) In the second field, enter the character string to be queried.
8. Click **Find** .
9. In the query result list, select the person (or person group) required and click **Connect**.

   – *Press the [CTRL] key to select several persons and/or person groups simultaneously.*

The user (or user group) you have connected appears in the list of users in the selected reading access area.

READING ACCESS DIAGRAM

The *reading access diagram* enables organization of the repository by sets of objects, unlike the writing access diagram which enables organization by work groups.

) *The reading access diagram enables definition of reading access areas and their hierarchical organization. This diagram also enables creation of users and their association with reading access areas.*

The reading access diagram can be opened only by an Administrator profile user connected to the maximum reading access area.

- *The reading access diagram feature is accessible only if you have the **HOPEX Power Supervisor** technical module.*

P **If you begin management of reading access of your data from the reading access area matrix, you must continue to manage reading access from this matrix. Do not manually modify links created automatically in the reading access diagram, you might invalidate the diagram.**

P **Warning: on exiting the reading access diagram, if a message indicates that the diagram is incorrect, the diagram is not compiled and reading access management does not operate. MEGA recommends that you correct the error that prevents diagram compilation.**

This section covers the following points:

- Reading Access Diagram Operating
- Activating the reading access diagram
- Prohibiting Reading Access Diagram Modification
- Opening the reading access diagram (Windows Front-End)
- Organizing Reading Access Areas
- Adding a User in the Reading Access Diagram
- Connecting Users to Reading Access Areas
- Consulting Reading Access Diagram Information:
- Customizing Reading Access Area Display

Reading Access Diagram Operating

The reading access diagram implements reading access areas of **General**  type. These areas can group both objects and persons and/or person groups.

Reading access areas are organized hierarchically. **MEGA** provides two extreme reading access areas:

- **Maximum Reading Access**, the highest reading access level.
- **Standard**, the lowest reading access level.

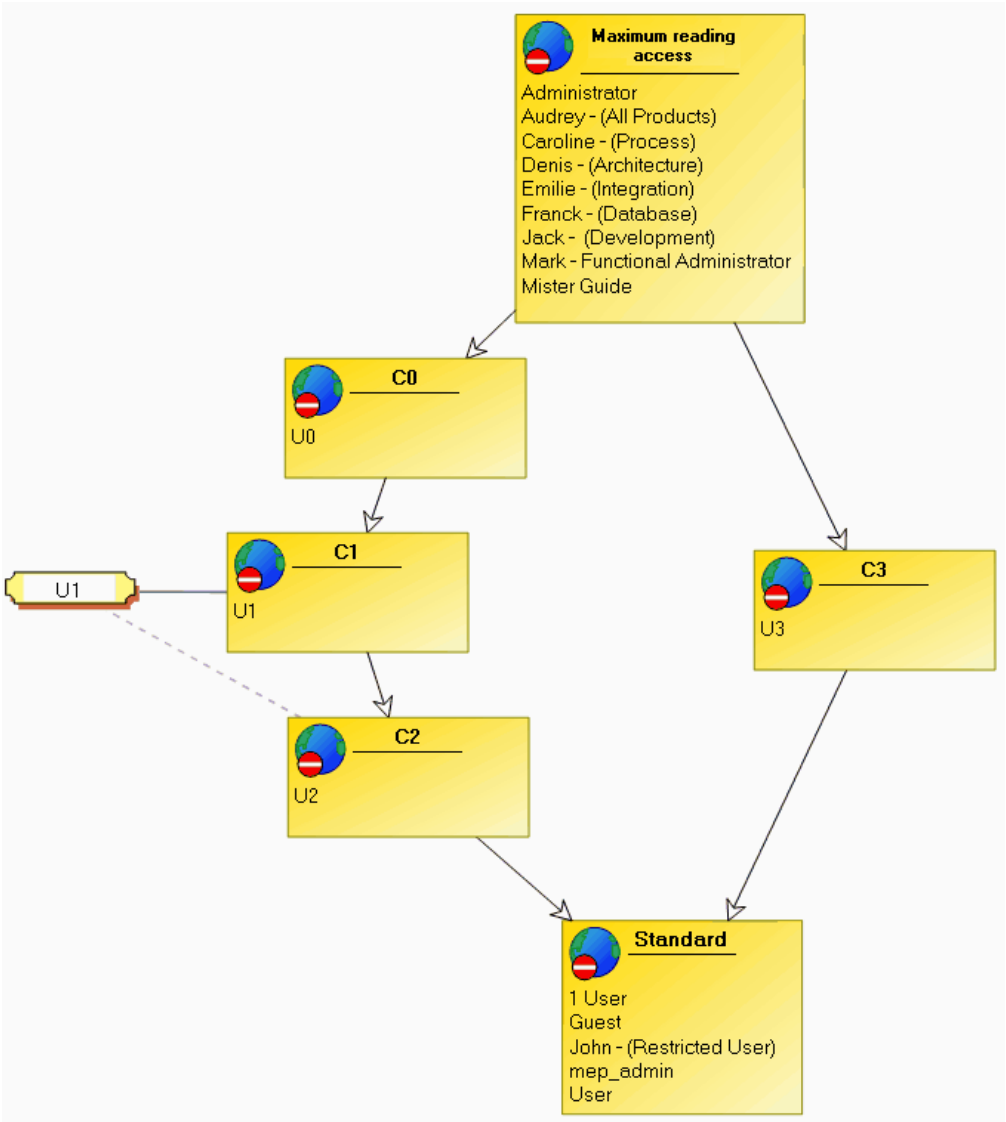
Each object belongs to a reading access area (**Standard** by default).

Each person or person group is connected to one of the reading access areas. The persons or person groups that are connected to:

- **Maximum reading access** sees all repository data.
- A created reading access area sees all data of this area, as well as that of lower level reading access areas.
- **Standard** sees only non-confidential data of the repository.

For example, a user U1 connected to a reading access area C1 sees all objects that belong to:

- his/her reading access area (C1)
- lower level reading access areas (C2 and Standard).



In the preceding reading access diagram, user U1:

- is connected to reading access area C1
- to a reading access area at creation C2.

When user U1 creates an occurrence of a MetaClass:

- if sensitive (high sensitivity), this belongs to reading access area C1.
- if non-sensitive (standard sensitivity), this belongs to reading access area C2.

> – *For more details on MetaClass sensitivity, see Managing MetaClass Sensitivity and Reading Access Areas.*

If a user does not have a reading access at creation, any occurrences of a non-sensitive MetaClass he/she creates belong to the standard reading access area.

However, Web sites and reports (MS Word) are always created at the reading access level of the user. This ensures confidentiality of the information they may contain.

Users connected to reading access area C3 cannot see objects belonging to reading access areas C0, C1, and C2, since area C3 does not belong to the same hierarchical branch as areas C0, C1, and C2.

Activating the reading access diagram

To be able to access the reading access diagram, you must first activate the reading access diagram option.

> – *By default, only the reading access area matrix is accessible.*

To activate the reading access diagram option:

1. From **HOPEX Administration**, connect to the environment concerned.
   > – *See Connecting to an Environment.*
2. Right-click the environment and select **Options > Modify**.
3. In the **Repository** group of options, select the **Activate the reading access diagram** option.
4. Click **OK**.
   The reading access diagram option is activated. The reading access diagram is accessible.

Prohibiting Reading Access Diagram Modification

By default, reading access diagram modification is authorized.

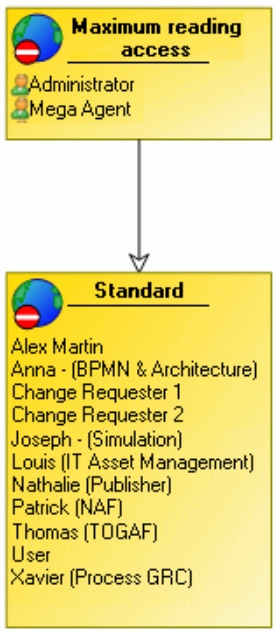
To prohibit reading access diagram modification:

1. From **HOPEX Administration**, connect to the environment concerned.
   > – *See Connecting to an Environment.*
2. Right-click the environment and select **Options > Modify**.
3. In the **Repository** options group, for option **Authorize modification of writing access and reading access diagrams** select "Prohibit".
4. Click **OK**.
   Reading diagram modification is prohibited.

Opening the reading access diagram (Windows Front-End)

To access the reading access diagram:

1. From **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Expand the **User management** folder of the environment.
3. Right-click the **Data Reading Access** folder and select **Open Diagram**.
   The reading access diagram appears.



By default, the reading access diagram contains two reading access areas:

- **Maximum Reading Access** is the highest reading access area level. Users connected to this area can see all objects in the repository.
- **Standard** is the lowest reading access area level.
  - *There can only be one maximum level and one minimum level (standard) reading access area in the diagram.*

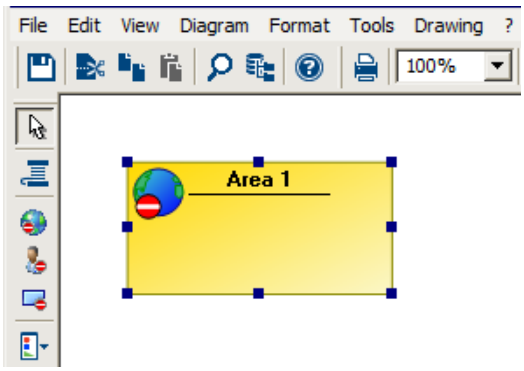
Organizing Reading Access Areas

Creating reading access areas

To create a *reading access area* in the diagram:

1. Open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*

2. In the diagram insert toolbar, click **General Reading Access Area** , then click in the diagram.
3. In the creation wizard dialog box that appears, enter the name of the reading access area and click **Create**.
   The creation wizard allows you to modify the reading access area type if necessary.
4. Click **Finish**.
   The general reading access area appears in the diagram.



Connecting two reading access areas

Reading access areas must be hierarchically interlinked. Except for **Maximum Reading Access** and **Standard** reading access areas, each reading access area must be connected to a lower level area and a higher level area.

To connect two reading access areas:

1. Open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*
2. In the diagram insert toolbar, click **Link** and draw a link between the two reading access areas (from the higher level reading access area to the lower level reading access area).

Displaying reading access areas associated with a reading access area

To display object (or user) reading access areas associated with a reading access area:

1. Open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*
2. Open the properties dialog box of the reading access area in question.
3. Select the **Matching Object reading access areas** or **Matching User reading access areas**.
   The associated object reading access areas or user reading access areas are listed.

Adding a User in the Reading Access Diagram

You can add a person or person group in the reading access diagram.

Adding a person in the reading access diagram

To add a person in the reading access diagram:

1. Open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*
2. In the diagram insert toolbar, click **Person**  , then click in the diagram.
   - *If the **Person** icon is not present in the insert toolbar, add it from the **View > Views and Details** menu.*

   The **Add Person** dialog box appears.
3. In the **Name** field, click the arrow to find the person then click **Connect**.
   - *To add a new person, in the **Name** field, enter the name of the person then click **Create**. Also create the login of the person.*

Adding a person group in the reading access diagram

To add a person group in the reading access diagram:

1. Open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*
2. In the diagram insert toolbar, click **Person Group**  , then click in the diagram.
   - *If the **Person Group** icon is not present in the insert toolbar, add it from the **View > Views and Details** menu.*

   The **Add Person Group** dialog box appears.
3. In the **Name** field, click the arrow to find the person group then click **Connect**.
   - *To add a new person group, in the **Name** field, enter the name of the person group then click **Create**. Also create the login of the person group.*

Connecting Users to Reading Access Areas

A user can:

- be connected to a ***reading access area***
  This area defines the view the user has of the repository and the objects the user can access.
- have a ***reading access area at creation***
  Occurrences created by the user belong to this reading access area at creation.
  - *If a user does not have a reading access area at creation, the occurrences he/she creates will belong to the standard reading access area.*

Reading access area of the user

To connect a user to a reading access area:

1. Open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*
2. Right-click the reading access area concerned and select **Properties**.
3. Select the **Persons** tab and click **Connect**.
   - *If you want to create a new user and associate him/her to a reading access area, select **New > Person**.*

   The access area member search dialog box appears.
4. In the first query field, select the type of access area member you wish to connect: **Person** (or **Person Group**).
   - *If you want to connect persons and person groups, select Access Area Member.*
5. (optional) In the second query field, enter the character string to be queried.
6. Click **Find**.
7. In the result list, select the person (press the [CTRL] key to select several) and click **Connect**.
   The user appears in the reading access area.

Reading access area at creation

You can assign a reading access area at creation to an existing user from:

- the reading access diagram
- the **Properties** dialog box of the person

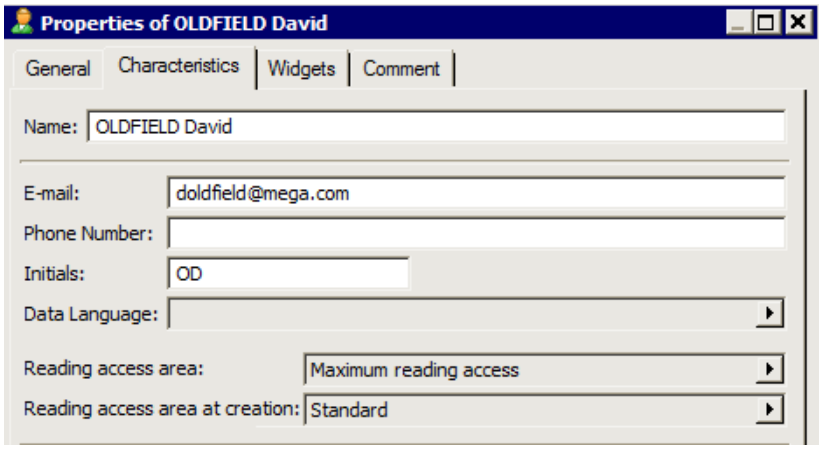
To assign a reading access area at creation to a user from the reading access diagram:

1. Place the user in the reading access diagram.
2. Connect the desired reading access area at creation to the person. This area must be at a level lower than or the same as the reading access area of the user.
   A dialog box asks you to select the type of link to be created: **Access area member** or **Access area member at creation**.
3. Select the link type **Access area at creation member**.
   This area is the reading access area at creation of the user.

To assign a reading access area at creation to a user from the user properties window:

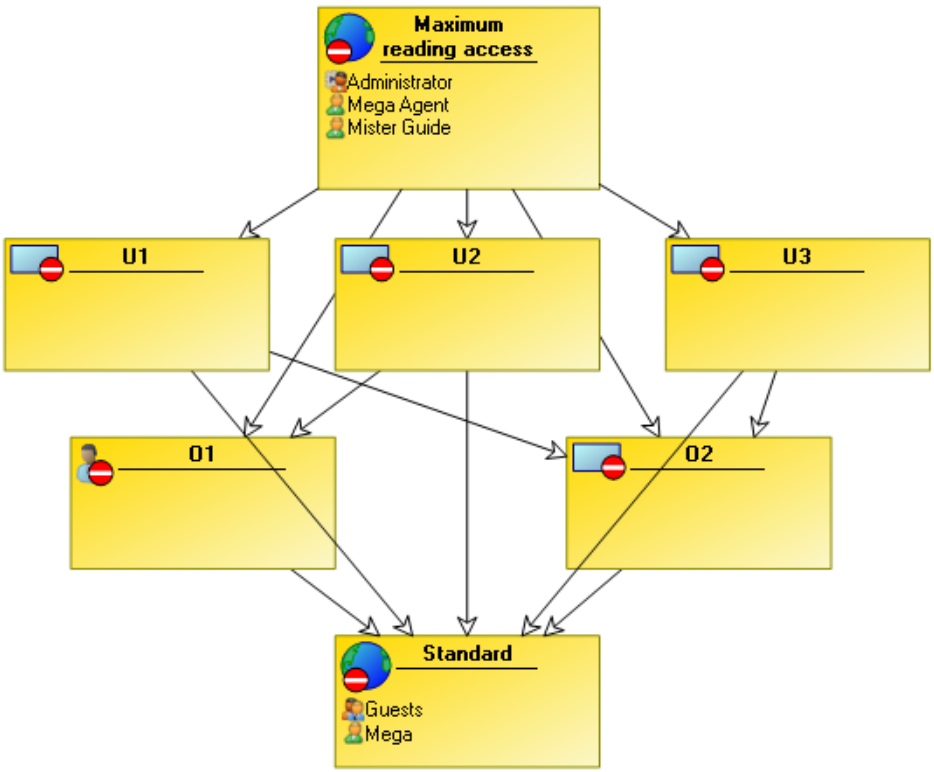
1. Open the **Properties** dialog box of the person and select the **Characteristics** tab, see *Modifying the Properties of a User*.

2. In the **Reading access area at creation** field, select the required value.



Consulting Reading Access Diagram Information:

At reading access diagram compilation, user and object are connected to the ***Maximum reading access*** reading access area and the ***Standard*** reading access area.



Open the user reading access area properties dialog box to consult:

- the list of persons connected to the area and to connect new users (**Persons** tab)
- the list of reading access areas for associated objects (**Matching Object Reading Access Areas** tab)

Open the object reading access area dialog box to consult:

› the list of user reading access areas associated with an object reading access area (**Matching User Reading Access Areas** tab)

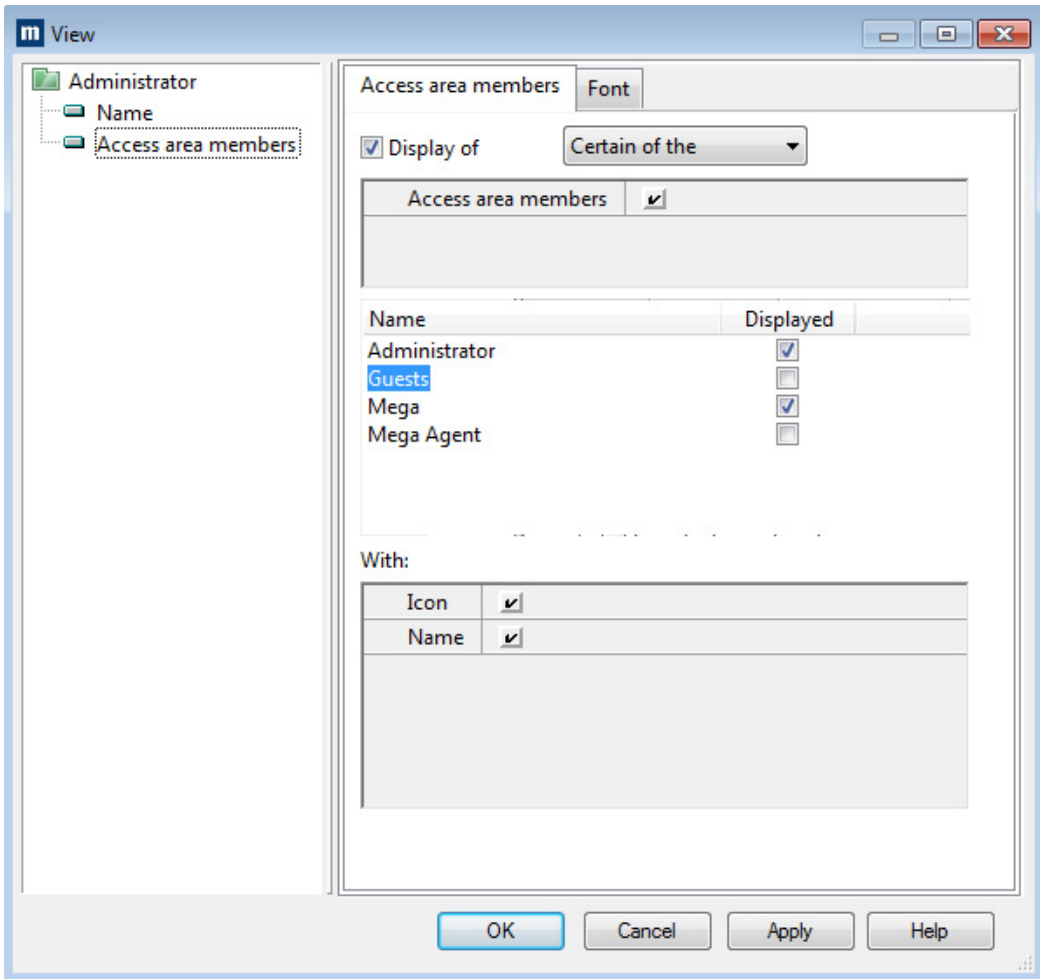
Customizing Reading Access Area Display

By default, when you create a reading access area, all users belonging to this area are displayed in the diagram reading access area. You can decide to hide certain users in this reading access area.

To define users displayed in their reading access area:

1. From **HOPEX Administration**, open the reading access diagram.
   - *See Opening the reading access diagram (Windows Front-End).*
2. Right-click the reading access area concerned and select **Shapes and Details**.
   The **View** dialog box opens.
3. In the tree on the left, select **Access Area Members**.
4. In the pane on the right, select **Access Area Members**.
5. Select the **Display of** option, then in its drop-down menu select **Certain of the** *Access area members*.
   - *To display all or none of the access area members, select **All the** or **None of the**.*

6. Select the persons you want to see displayed.



7. Click **OK**.

CONFIGURING DATA READING ACCESS

In the **HOPEX** desktop, the navigation window allows access to certain data reading access functions.

This section presents how to:

- Associating Objects with Reading Access Areas
- Associating user reading access areas with objects
- Propagating Reading Access Areas
- Managing MetaClass Sensitivity and Reading Access Areas
- Confidential or Sensitive Object Behavior
- Modifying Reading Access Areas

Associating Objects with Reading Access Areas

The **Administration** navigation window of the **HOPEX** desktop allows access to general and object reading access areas.

– *To access content of the **Administration** navigation window, you must have **Advanced** or **Expert** metamodel access (see Configuring the Metamodel Access).*

The reading access areas tree is used to:

- connect objects to a given reading access area
- quickly propagate the reading access area to child objects

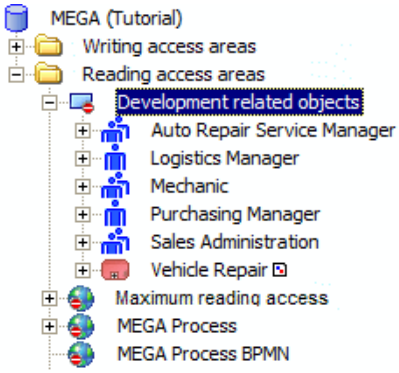
This tree simplifies management of propagation of reading access areas (see Propagating a reading access area from HOPEX). Its advantage compared with the classic propagation tool is that the propagation trace is kept thanks to the link.

Connecting objects to object reading access areas

To connect an object to an object reading access area:

1. Connect to **HOPEX**.
   – *See Connecting to HOPEX.*
2. Open the **Administration** navigation window.
3. Expand the **Data Reading Access** folder.
4. Right-click the object reading access area concerned and select **Connect Object**.

5. In the query dialog box, find and select the desired objects and click **OK**. The objects are connected to the object reading access area and appear in the tree.



Disconnecting objects from reading access areas

To disconnect an object connected to a reading access area:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. Open the **Administration** navigation window.
3. Expand the folder of the reading access area concerned.
4. Right-click the object you want to disconnect and select **Disconnect**. The object disappears from the tree.

Displaying the list of objects associated with a reading access area

To display the list of objects associated with a reading access area:

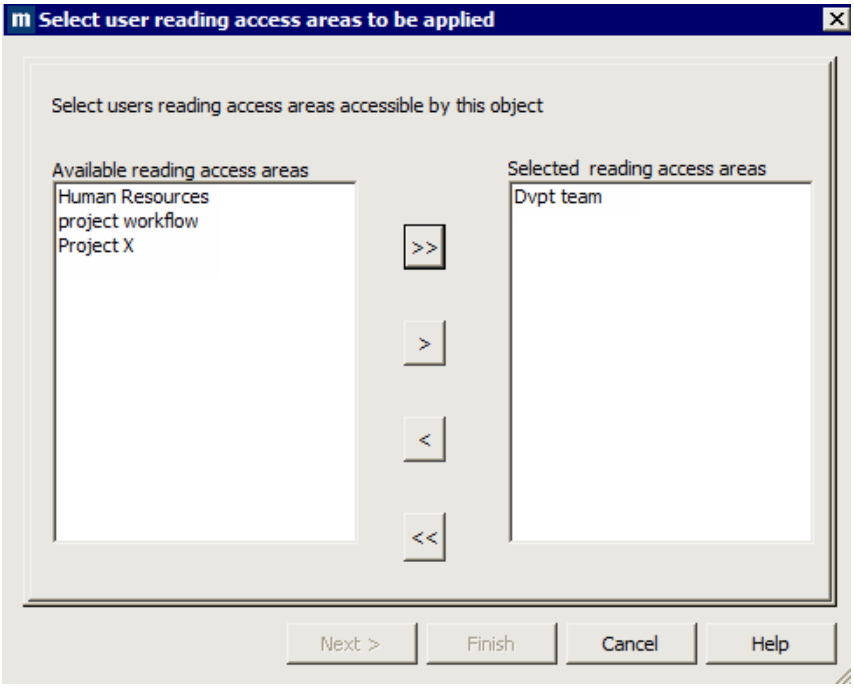
1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. Open the **Administration** navigation window.
3. Expand the **Data Reading Access** folder.
4. Right-click the reading access area concerned and select **Objects associated with reading access area**.
   A dialog box displays a list of these objects.

Associating user reading access areas with objects

To associate an object with one or several groups of users, proceed as follows:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. Select an object.
3. In the object properties dialog box, select the **General** tab, then the **Administration** subtab.

4. In the **Reading Access Area** drop-down list, click the arrow and select **Associate User Reading Access Areas**.
5. Using the arrows in the selection dialog box, move the user user reading access areas from available to selected, then click **Next**.



6. In the next dialog box, if an object reading access area corresponds to the user reading access areas, you are invited to validate this area, otherwise you must enter the name of a new user reading access area, which will be created corresponding to the previously selected user reading access areas.
7. Click **Finish**.

If you had to create a new object reading access area, this is automatically added in the reading access diagram and is connected to the corresponding user reading access area or areas, as well as to the ***Standard*** reading access area.

Propagating Reading Access Areas

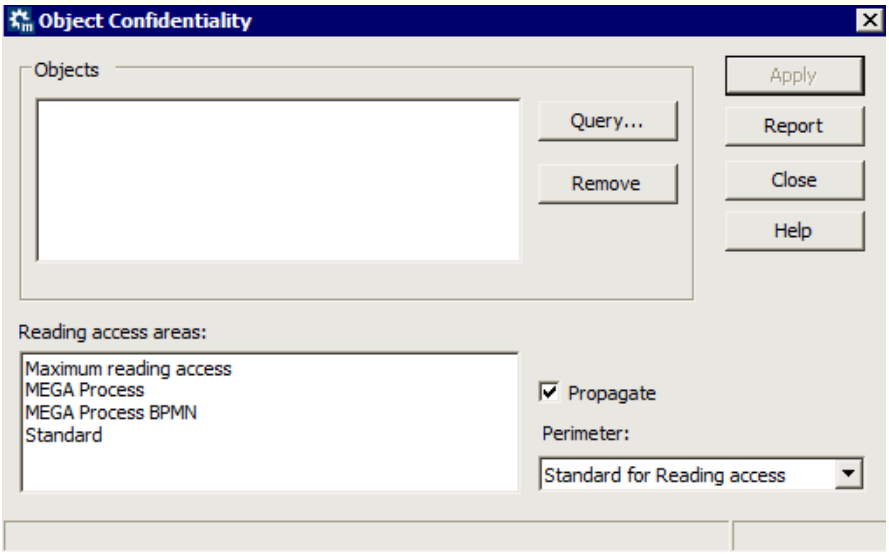
A reading access area can be propagated to objects connected to a given object. You can propagate reading access areas from:

- **HOPEX Administration**
- **HOPEX**.
  - *The propagation trace is kept when you propagate from **HOPEX**.*

Propagating a reading access area from HOPEX Administration

To propagate a reading access area:

1. Connect to **HOPEX Administration** and select the repository concerned.
   - *See Accessing Repositories.*
2. Right-click the repository concerned and select **Object Management > Object Confidentiality Setting**.



3. In the **Object Confidentiality** dialog box that opens, click **Query**.
4. Select the desired objects using the query tool, then click **OK**.
5. In the **Object Confidentiality** dialog box, in the **Reading Access Areas** frame , select the reading access area you want to apply to the selected objects.
6. (Optional) Select **Propagate** to propagate the reading access area to sub-objects.
7. Click **Apply**.
   - *The operator used to propagate reading access areas is the "Standard for reading access" operator.*

Propagating a reading access area from HOPEX

To propagate a reading access area:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. Open the **Administration** navigation window.
3. In the **Administration** navigation window, right-click the reading access area you wish to propagate and select **Propagation of reading access area to associated occurrences**.
   A dialog box may warn you of the presence of propagation conflicts.
   - *Conflicts can arise if child objects are already connected to a different reading access area. You are informed that propagation stops*

at this child object (the physical link that connects the object to a reading access area is stronger than the reading access attribute value)

– The operator used at reading access area propagation is the "Standard for reading access" operator.

*– Alternatively, you can use the confidentiality area propagation tool via the menu **Tools > Manage > Object Confidentiality Setting**.*

Managing MetaClass Sensitivity and Reading Access Areas

*– The attribute enabling configuration of MetaClass sensitivity is accessible only if you have the **HOPEX Power Supervisor** technical module.*

In the reading access management frame:

- a user is connected to a reading access area that defines all the objects he/she can see.

  – See Connecting Users to Reading Access Areas.

- an object type (MetaClass) is characterized by its sensitivity.

A MetaClass can be of sensitivity:

- **standard** (default value)
  Occurrences of the MetaClass created by a user belong to the user reading access at creation area or the Standard reading access area if the user does not have a reading access at creation area.
- **High**
  Occurrences of the MetaClass created by a user belong to the reading access area of the user that creates them.

  *– To modify the sensitivity of a MetaClass, you must have rights to modify **HOPEX** data. The option "Authorize HOPEX Data Modification" must be activated at environment level, see Managing Options.*

You can modify sensitivity value of the MetaClass.

– See Modifying MetaClass sensitivity.

*– By default, a MetaClass is **Standard** sensitivity.*

Opening the HOPEX MetaClasses reading access configuration dialog box

To open the **HOPEX** MetaClasses reading access configuration dialog box:

1. Connect to **HOPEX Administration** and select the environment concerned.

   – See Connecting to an Environment.

2. Expand the **User management** folder of the environment.
3. Right-click the **Data Reading Access** folder and select **Configure HOPEX MetaClasses for reading access**.
   The **HOPEX MetaClasses Reading Access Configuration** dialog box appears, listing the available **MetaClasses**.
   The icon alongside the name of each MetaClass indicates that default values:
   - have been modified
   - are retained.

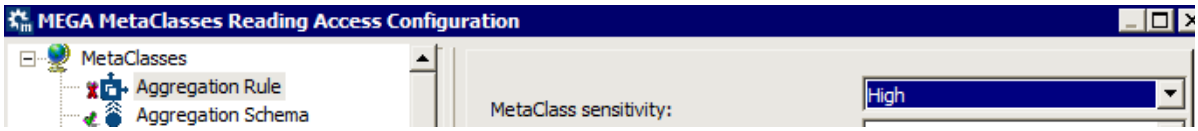
Modifying MetaClass sensitivity

To modify MetaClass sensitivity:

1. Open the **HOPEX** MetaClasses reading access configuration dialog box.

   - *See Opening the HOPEX MetaClasses reading access configuration dialog box.*

2. In the list of **MetaClasses**, select the desired MetaClass.
3. In the right pane, select the MetaClass sensitivity value:



   - *A red cross alongside the name of the MetaClass indicates that at least one attribute of the MetaClass has no longer its default value.*

4. Click **OK**.

   Modifications carried out may be canceled when your environment is updated. Remember to back up your extensions (metamodel and technical data).

Hiding confidential or sensitive objects in a diagram

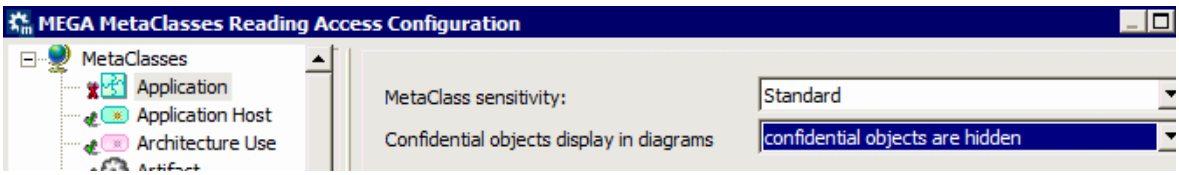
So as not to distort a diagram, confidential or sensitive objects are visible by default.

To hide confidential or sensitive objects in a diagram:

1. Open the **HOPEX** MetaClasses reading access configuration dialog box.

   - *See Opening the HOPEX MetaClasses reading access configuration dialog box.*

2. In the list of **MetaClasses**, select the desired object.
3. In the right pane, in the **Confidential objects display in diagrams** box, select "Confidential objects are hidden".



   - *A red cross alongside the name of the MetaClass indicates that the MetaClass default value has been modified.*

4. Click **OK**.

   – *These modifications are not taken into account in the metamodel until the metamodel is compiled. Compile the metamodel before closing the **HOPEX Administration** module, see Compiling an Environment.*

   Objects corresponding to this MetaClass will be hidden in the diagram.

   – *When you select "Confidential objects are visible", objects corresponding to this MetaClass appear grayed in the diagram and you cannot access information relating to these objects.*

   P **Modifications carried out may be canceled when your environment is updated. Remember to back up your extensions (metamodel and technical data).**

Confidential or Sensitive Object Behavior

A confidential object is inaccessible to a user that does not have access to the corresponding reading access area.

It is as if the object did not exist, the object:

- does not appear in lists.
- is ignored in query results.
- does not appear in reports (MS Word) or Web sites.
- is not exported, duplicated, deleted or backed up, and its possible "children" (operations of a process, for example) are considered as orphans.
- only appears when another object is created with the same name or when a higher level object with a lower reading access level is deleted.
- cannot be modified.

  – *Reading access management is not supported in **HOPEX Database Builder**.*

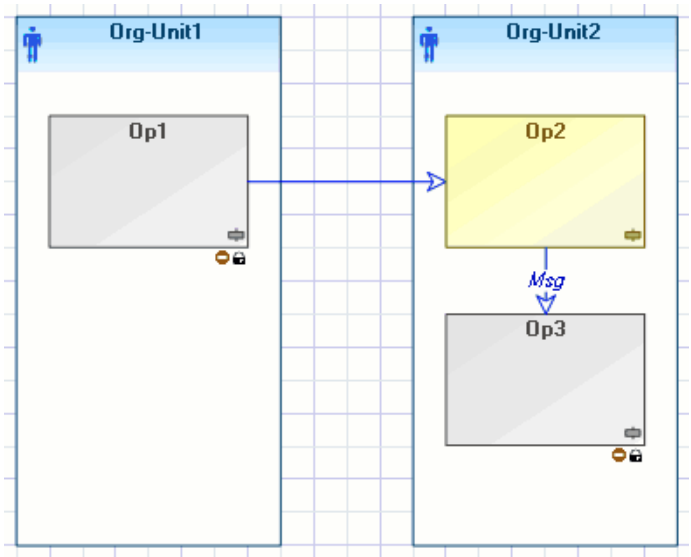
For more details on reading access areas, see Managing Reading Access in HOPEX.

Displaying a confidential or sensitive object in a diagram

By default, confidential or sensitive objects are visible in diagrams.

– *To hide confidential or sensitive objects, see Hiding confidential or sensitive objects in a diagram.*

The name of a confidential or sensitive object is visible in the diagram, but its properties are not accessible. It appears grayed and an icon at bottom right indicates that the object is confidential or sensitive.



A confidential or sensitive object can only be resized and moved.

To hide MetaClass occurrences in diagrams:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In the **HOPEX** menu bar, select **View > Navigation Windows > MetaStudio**.
3. Expand the **MetaClass** folder and its sub-folders.
4. Right-click the desired MetaClass and select **Properties**.
5. In the **Characteristics** tab (**Standard** subtab) for the **Confidential objects display in diagrams** property, select **Confidential objects are hidden**.
6. Click **OK**.
   The occurrences of the corresponding MetaClass are now hidden in diagrams.
   - *You can also configure display of objects in diagrams via the reading access diagram, see Hiding confidential or sensitive objects in a diagram.*

Export and Duplication

In the case of object export and duplication, if these operations have an impact on confidential objects, a warning message will ask you if you want to continue. If you do execute export or duplication, the confidential objects concerned will be neither copied nor exported.

Invisible objects (confidential) are not duplicated during object duplication. Only a message in the status bar indicates that the duplication result is incomplete.

Generation of reports (MS Word) and Web sites

When created, reports (MS Word) and Web sites are created with the reading access level of their creator (and not with the reading access level at creation)..

They are then always generated at their their reading access level.

A **Reading access area path** attribute exists on all reading access areas. If this attribute is specified, Web sites and reports (MS Word) are generated in this folder. Reports (MS Word) and Web sites are generated in this folder to facilitate reading access management of generated files by defining access rights to this generation folder. This task should be handled by the system administrator.

Confidential report (MS Word) and Web site generation paths can be defined in the properties dialog box of a reading access area.

To define this path:

1. In the reading access area properties dialog box, select the **Characteristics** tab.
2. In the **Reading access area path** field, specify the required path.
3. Click **OK**.

Macros

The principle of reading access management in macros is to carry out all calculations in **HOPEX** and hide confidential or sensitive objects from users that do not have sufficient reading access area access rights to view them.

By default a macro is executed at user reading access level.

A macro can also be executed at its own reading access level.

If you execute a macro with a reading access level higher than the level of the current user, the methods.

- GetProp(xxx, "display") and GetFormatted return empty,
- GetProp("xxx") returns the value.

ExecuteGlobal and ***CreateObject*** ("Mega.Application") methods are prohibited in macros.

Other properties are accessible.

So that a macro can be executed with its reading access level:

1. In the macro properties dialog box, select the **Characteristics** tab.
2. In **_ExecutionOptions**, select the **Execution at Reading Access level** option.

Confidential or sensitive objects and namespaces

If an object with a namespace is not confidential or sensitive, but its parent is confidential or sensitive, the name of the latter will be masked in **HOPEX**. It appears in **HOPEX** as:

```
" ***::Operation 1 "
```

Modifying Reading Access Areas

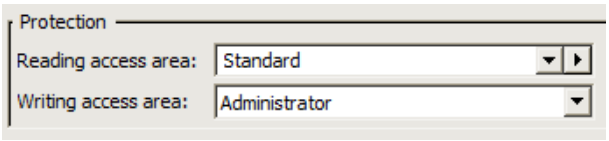
This section explains how to modify the reading access area area of an object or a user:

Modifying object reading access areas

From an object properties window, you can consult the reading access area to which it belongs.

To determine the reading access area of an object:

1. Right-click the object and select **Properties**.
2. In the object **Properties** dialog box, select the **General** tab, then the **Administration** subtab.
   In the **Protection** frame, you can consult and modify the **Reading access area**.



Modifying user reading access areas

You can modify a user reading access area from:

- the user management window (**Reading access area** column of the person)
  - *See Modifying the Properties of a User.*
  - *The reading access diagram is compiled to take account of modifications.*
- the person properties window
  - *See Defining a Person.*

  P **Warning: if you modify the reading access area in the user properties dialog box, you must recompile the reading access diagram so that the modification will be taken into account.**
- the reading access diagram
  - *See Connecting Users to Reading Access Areas or Reading access area of the user.*

You can modify a user reading access area at creation from:

- the person properties window
  - *See Defining a Person.*

  P **Warning: if you modify the reading access area in the user properties dialog box, you must recompile the reading access diagram so that the modification will be taken into account.**
- the reading access diagram
  - *See Reading access area at creation.*

METACLASS CONFIDENTIALITY EXCEPTIONS

The following MetaClasses cannot be made confidential:

_Add-ins_data
_Add-ins_meta
_Brick
_Class
_ClassCommand
_Code Template
_Command
_Dispatch
_Executable
_MappingTypeItem
_MappingTypeItemProperty
_Object
_Operator
_Property
_Proposed_Table
_Resource
_StdFile
_Style
_TagAttributeDef
_TagAttributeDefValue
_TagDef
_Template
_Text
_Transaction
_TransactionData
_TransferredObject
_Type
_UML Reserved Word
Method author
MEGA Repository
ChangeItemData
ChangeItemDataTechnical
ChangeItemSystem
Generation kinematics
Component Template
DiagramTypeLink
DiagramTypeLinkStyle
DiagramTypeObject
DiagramTypeCollection
DiagramTypeField
DiagramTypeFormat
DiagramTypeParam
DiagramTypePath
DiagramTypePathPart
DiagramTypePopulating
DiagramTypeProperty
DiagramType
DiagramTypeView
Stem Codes Folder
Web Site Templates Folder
Analysis Templates Folder
Diagram Types Folder
HTML Formatter
Generality
Generator
Programming Language
Language
Animation Mask
Matrix Template

MetaAssociation
MetaAssociationEnd
MetaAssociationType
MetaAttribute
MetaAttributeGroup
MetaAttributeValue
Metaclass
MetaClassDiagramType
MetaCommand
MetaField
MetaList
MetaListType
MetaPattern
MetaPicture
MetaPropertyPage
MetaTest
MetaTree
MetaTreeBranch
MetaTreeNode
Method

Associative Object
Default Associative Object
Generic Object
System Generic Object
Analysis Parameter
Profile
Query Parameter
Generation rule
Modeling Rule
Modeling Regulation
Query
Web Site Template
SQL Clause Type
TaggedValue
Analysis Type
user
Descriptor Setting
DBMS Version
Writing access area
Reading access area

COMMAND FILE SYNTAX

The following points are covered here:

- Command file extensions
- Object Naming Rules
- Commands
- Basic Syntax

COMMAND FILE EXTENSIONS

) *A command file is a file containing repository update commands. It can be generated by backup or object export (.MGR extension) or by logfile export (.MGL extension).*

Command files can be obtained in two ways:

- By logical backup or by object export (.MGR): the *absolute identifiers* (IdAbs) of the imported objects are used and the authorization levels are kept.

) *An absolute identifier is a string of characters associated with each object in the repository. This string is computed using the date and time the session was opened, the number of objects created since the session started, and the object creation date (in milliseconds). An absolute identifier provides a unique way to identify an object in the repository, so that even if the object is renamed, all the links to it are retained.*

- By logfile export (.MGL): the commands contain, in addition to the *absolute identifiers* (IdAbs) of objects, that of the user executing each command to check at import that the user had the necessary rights to execute this update.

Result of the import of these files is therefore different:

- ".MGR" corresponds to an image, complete or partial, of the repository at a given moment. It is therefore recommended that it be imported into an empty repository to rebuild this image.
- ".MGL" corresponds to commands to be applied to the repository to pass from initial state to final state.

At import, checks are performed automatically as a function of the file extension:

- For command files with the MGR extension, the absolute identifiers of the imported objects are used and the writing access levels are kept.
- For command files with the MGL extension (exported logfile or backup logfile), the absolute identifiers of the imported objects are used. Writing access levels are checked. The authorization levels are kept if the updates are consistent with the writing access diagram for the environment, otherwise they are rejected.

OBJECT NAMING RULES

Object naming will depend on the uniqueness rule applied to its name. This rule is important, since the name appears in command files.

An object has a unique name

An object must have a unique name throughout the repository.

For example, a report template (MS Word) has a name that appears in command files.

A name is unique in a given context

Several objects can have the same name, but the name must be unique in a particular context: therefore it has a namespace.

For example, an operation of an organizational process: its name must be unambiguous within the process, but several different process operations can carry the same name.

For these objects, two names are presented to the user in the user interface:

Con-cept	Example	Comment
Name	Hire::Call candidate	Complete object name. Unique in the repository. Calculated from the local name and the name-space name (which can itself have a name-space). Here the operation "Call candidate" belongs to the "Hire" process.
Local Name	Call candidate	Name of the object in its namespace. Unique in the namespace.

Two names are used in **HOPEX** command language:

Concept	MetaAttribute	Example of value
Internal name of the object. It contains HexaIdabs of the object.	Name	14B8162B3F3A0347
Local name of the object and Hex-aIdabs of its namespace.	Generic Local Name	Call candidate [85ED06B63EC95B6F]

These build rules ensure respect of naming rules imposed by the repository:

- The name must be unique: the IdAbs is built to be so.
- The local name must be unique in its context: specify a uniqueness constraint on the GenericLocalName.
  - *If the object is detached from its namespace, in the local name the indicated HexaIdAbs is then a string of 16 "0".*

Objects without name constraint

There is no name uniqueness constraint on certain objects such as messages: the same operation can send or receive several messages with the same name.

In this case, the object constitutes its own namespace.

Two names are used in **HOPEX** command language:

Concept	MetaAttribute	Example of value
Internal "Name" of the object. It contains HexaIdabs of the object.	Name	14B8162B3F3A0347
Local name of the object and HexaIdabs of its namespace (itself).	Message Local Name	Convocation [14B8162B3F3A0347]

COMMANDS

Commands on objects

- .Create (creation of an object)
- .Update (modification of an object MetaAttribute)
- .Delete (deletion of an object)

Commands on links

- .Connect (creation of a link between two objects)
- .Disconnect (deletion of a link between two objects)
- .Change (modification of a link MetaAttribute)

Other Commands

- .Validate (triggers intermediate save on import)
- .Description (produces display in import dialog box)

Rules to be respected

Command files must comply with the following rules:

- A command line cannot contain more than 5000 characters.
- Object names are limited to:
  - 63 characters for object types without namespace.
  - 255 characters for object types with namespace (name or local name).
- Commands begin with a verb infinitive prefixed by ".".
- The "." of the command must be in the first column.
- Use a hyphen (-) at the end of a line to indicate that it continues on the next line.
- Comment lines are indicated by a hyphen (-) at the beginning of the line.
- Use double-quotes (") around values that contain spaces or characters other than letters or digits.

Remarks

- Certain objects are functionally invalid if one of their MetaAttributes is not entered or a link is not defined. For example, a diagram type object must be connected to another object by a descriptor type link. We say that the diagram describes this object.
- To exchange data between two **HOPEX** environments, they must have identical metamodels and coherent user diagrams.

Commands as function of file type

Each command must consist of:

- a verb indicating the action to be carried out
- a list of parameters required to carry out this action (object types and names)
- a keyword ".CHK" followed by a list of the IdAbs of objects impacted by this command.

> – *The fact of repeating the object name and IdAbs in the command enables its correct execution, even if the object has been renamed.*

The difference between the command of an ".MGR" file and the same command of an ".MGL" file is in the ".CHK":

- they have the same verb
- they have the same list of parameters
- the ".CHK" of MGL contains in addition in last position, the IdAbs of the user that issued the order.

> – *A third file format (obsolete in this version) is ".MGE". In these files commands do not have a .CHK. The IdAbs are assigned as required. It is therefore not possible to process "namespaced" objects for which the namespace IdAbs cannot be assigned, since it forms part of their name.*

References to the metamodel

Each metamodel instance (MetaClass, MetaAttribute, …) can be prefixed by its IdAbs. This assures permanence of files despite renamings which may be carried out in the metamodel.

```
Example:
"~OsUiS9B5iiQ0[Operation]" is equivalent to "Operation".
```

BASIC SYNTAX

See:

- Creating an Object
- Deleting Objects
- Modifying an Object
- Modifying Texts
- Modifying a Name
- Creating and Modifying an Object with a Single Command
- Creating a Link Between Two Objects
- Modifying a Link
- Deleting a Link
- Managing Translations
- Validating Import
- Displaying a Comment in the Import Dialog Box
- Transforming an MGL File to MGR
- Transforming an MGR File to MGL

Creating an Object

Syntax	***.Create ."Object type""Object name" -*** ***.CHK "..."***
Example 1	.Create ."~IdAe93gyh020[Report template (MS Word)]" "Application documentation" - .CHK "w0e4VVXC)440e0SDsNpple00"
Example 2	.Create ."~OsUiS9B5iiQ0[Operation]" "14B8162B3F3A0347" - .CHK "GZB5hOXE)Sq0C30000mCpCpC"

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of the object
- IdAbs of object writing access
- In the case of the MGL, the IdAbs of the user that made this command.

Certain MetaAttributes, such as "Creation date" or "Creator" can only be updated at object creation. They must therefore be incorporated in this command.

Example:

```
Create ."~OsUiS9B5iiQ0[Operation]" "14B8162B3F3A0347" -
    .CHK "GZB5hOXE)Sq0C30000mCpCpC" -
    ."~510000000L00[Creation Date]"          "2003/08/13 10:42:51" -
    ."~(10000000v30[Creator]"              "OmNRasMwq400" -
    ."~520000000L40[Create Version]"          "25088"
```

The "Creator" and "Modifier" MetaAttributes contain the IdAbs of users that have created and modified the object. If they are not specified in the command, they automatically take the IdAbs of the user importing the file.

Similarly, "Link creation date" and Link modification date" are specified from the import date if they are absent.

Deleting Objects

Syntax	***.Delete ."Object type" "Object name" -*** ***.CHK "..."***
Example 1	.Delete ."~IdAe93gyh020[Report template (MS Word)]" "Application documentation" - .CHK "w0e4VVXC)440"
Example 2	.Delete ."~OsUiS9B5iiQ0[Operation]" "14B8162B3F3A0347" - .CHK "GZB5hOXE)Sq0"

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of the object
- In the case of the MGL, the IdAbs of the user that made this command.

Deletion of an object systematically results in:

- Loss of its attribute and text values.
- Deletion of all of the links around the object.

Modifying an Object

Syntax	***.Modify ."Object type" "Object name" -*** ***.CHK "..." -*** ***."metaattribute1" "Value1" -*** ***."metaattribute2" "Value2"***
Example 1	.Update ."~MrUiM9B5iyM0[Application]" "874B9C483D7828C6" - .CHK "PjqX8n9UzOCA" - ."~610000000P00[Modification Date]" "2010/09/07 10:26:30" - ."~b10000000L20[Modifier]" "xDqT)UdFwC10" - ."~2yUL4SsRp4B0[Application Code]" "GESTCAT11" - ."~ByUL4SsRpeB0[Operating Application Date]" "1995/10/04 23:00:00"
Example 2	.Update ."~gsUiU9B5iiR0[Organizational Process]" "0A496AAE407D1621" - .CHK "Vba2kgMV05Y5" - ."~pjRX1OOKne20[Process Frequency]" "Q""

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of the object
- In the case of the MGL, the IdAbs of the user that made this command.

The "Modification date" and "Modifier" MetaAttributes can be modified like standard MetaAttributes. If they are not specified in the command, they automatically take the file import date and the IdAbs of the user importing the file.

In the case of "Example 2" with a tabulated attribute, the value to be entered is the internal value (for process frequency this is "D" and not "Daily").

Modifying Texts

Syntax	***.Modify ."Object type" "Object name" -*** ***.CHK "..." -*** ***."Text name" "Text format"*** ***Text value*** ***.***
Example 1	.Update ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" - .CHK "AH(tl0UJDDxA" - ."~f10000000b20[Comment]" "g3TCfAJnyq00" 00680SbnxCMPqRc5SN6bpSsvXS6DfCZ5dN38rPcLaN31cPcLaRc5iC35dUpOpRsPST7HkUsnY 00680C6PSRcPSN6nfQ6DcSt9XC7HbKqqWQ5CWR6nbR4GWVJjd2WrzQNPSQtTbP6vfTLmqN 35Z 00602Sc5mPbnaPbmmC39pRqCWPMrj87Hk86PlS71XOsbiQNHX86vlS5mn3N9X3NqA000A .

Text format	Value
ASCII text	0 000000000000
Binary text	1 000000000001
RTF text	"MRDYO5Oe(smC"
HTML text	"LQDYO58M6tmC"
ANSI text	"G3000000W10S"

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of the object
- In the case of the MGL, the IdAbs of the user that made this command.

In a command file, each line of text is limited to 74 characters.

End of the text marked by a line containing only one point (".") in the first column.

A semi-colon in the first column inserts a blank line (the rest of the line must be left blank).

> – *When text is extracted, lines are divided after character 73 and a semicolon is inserted in position 74, indicating that the next line is to be concatenated with the previous one.*

Complementary indicators:

- Each text modification applies to its totality. It is not possible to add just one complement.
- The semicolon character (;) in first position inserts an blank line. To reinitialize a text, it is sufficient that the text value contains just a semicolon.
- The characters period and semicolon (.;) in first and second position create a line containing a period only.
- Two semicolon characters (;;) in first and second position create a line containing a semicolon only.
- Apostrophe (') and quotation marks (") are authorized as text values.
- The semicolon character (;) enables cutting of text lines exceeding 74 characters. The semicolon is therefore the last significant character in the line.

To reinitialize a text, it is sufficient that the text value contains just a semicolon.

```
Example:
Update ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" -
        .CHK "AH(tl0UJDDxA" -
        ."~f10000000b20[Comment]" "000000000000"
;
.
```

To delete a text, the text value should be left empty.

```
Example
Update ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" -
        .CHK "AH(tl0UJDDxA" -
        ."~f10000000b20[Comment]" "000000000000"
.
```

P **A reinitialized text contains nothing but it exists, while a deleted text no longer exists. For example the query *"Select Application where Comment null"* returns applications that have no comment, but not those that have a reinitialized comment.**

Modifying a Name

Syntax	***.Modify ."Object type" "Object name" -*** ***.CHK "..." -*** ***."Name or Local name" "Value "***
Example 1	.Update ."~ldAe93gyh020[Report template (MS Word)]" "Report template (MS Word)-1" - .CHK "RJ(tBUUJD5(AV(WEIeZIDT4B" - ."~210000000900[Name]" "Report template (MS Word)-New"
Example 2	.Update ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" - .CHK "AH(tl0UJDDxA" - ."~g20000000f60[Generic Local name]" "Application-1[0000000000000000]"

Creating and Modifying an Object with a Single Command

At object creation, creation of a "modification" command by MetaAttributes to be specified is of no interest: MetaAttributes (non-textual) can be directly assigned by the create command.

Syntax	***.Create ."Object type" "Object name" -*** ***.CHK "..." -*** ***."metaattribute1" "Value1" -*** ***."metaattribute2" "Value2"***
Example 1	.Create ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" - .CHK "AH(tl0UJDDxAC30000mCpCpC" - ."~510000000L00[Creation Date]" "2011/02/05 11:41:35 PM" - ."~610000000P00[Modification Date]" "2011/02/06 12:31:38 AM" - ."~(10000000v30[Creator]" "V(WEIeZIDT4B" - ."~b10000000L20[Modifier]" "V(WEIeZIDT4B" - ."~520000000L40[Create Version]" "29248" - ."~620000000P40[Update Version]" "29248" - ."~)20000000z70[Confidentiality area identifier]" "sTlVwxdH3100" - ."~2yUL4SsRp4B0[Application Code]" "AA" - ."~KyUL4SsRpCC0[Version Number]" "12" - ."~ByUL4SsRpeB0[Operating Application Date]" "2011/02/11 11:00:00" - ."~)Sgoy)ygu020[Application Type]" "M" - ."~nOU8g8IMCb30[Converted Name Version]" "0" - ."~PYq45X2wBP92[Date of C&A Completion]" "2011/02/06 11:00:00" - ."~Ol8pdE(l8r(0[Deployment Date]" "2011/02/27 11:00:00" - ."~g20000000f60[Generic Local name]" "Application-1[0000000000000000]" - ."~PZq41c2wBXP2[Security Planning]" "Operational" - ."~a20000000H60[LanguageUpdateDate]" "2011/02/05 23:41:57"

Creating a Link Between Two Objects

Syntax	***.Connect ."Object type" "Object 1 name" ."MetaAssociationEnd" "Object 2 name" -*** ***.CHK "..."***
Example 1	.Connect ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" ."~mi54NLn-HzCj0[Application within Internal Architecture]" "DFE4F2274D4D2C43" - .CHK "AH(tl0UJDDxAbJ(td8VJDD4B" - ."~710000000T00[Link creation date]" "2011/02/06 00:58:15" - ."~810000000X00[Link modification date]" "2011/02/06 00:58:15" - ."~720000000T40[Link Creator]" "V(WEIeZIDT4B" - ."~920000000b40[Link Modifier]" "V(WEIeZIDT4B" - ."~410000000H00[Order]" "9999"

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of object 1
- IdAbs of object 2
- In the case of the MGL, the IdAbs of the user that made this command.

The "Order" MetaAttribute is optional. If present, the order is numeric on maximum four positions, otherwise the order is set by default to 9999.

The "Link creator" and "Link modifier" MetaAttributes contain the IdAbs of users that have created and modified the link. If they are not specified in the command, they automatically take the IdAbs of the user importing the file.

Similarly, "Link creation date" and Link modification date" are specified from the import date if they are absent.

- *If this link is used to build a namespace, it must be completed by modification of the local name of the namespaced object to maintain repository consistency.*

Similar to object creation, it is possible to specify link MetaAttributes (except texts) directly in this command, without passing via a modification command.

Modifying a Link

With the exception of its header, this command has the same syntax as the object modification command.

– *See Modifying an Object.*

Syntax	***.Change ."Object type" "Object 1 name" ."MetaAssociationEnd" "Object 2 name" -*** ***.CHK "..." -*** ***."metaattribute 1" "Value 1" -*** ***."metaattribute 2" "Value 2" -*** ***."Text name" "Text format" -*** ***Text value*** ***.***
Example	.Change ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" ."~mi54NLnHzCj0[Application within Internal Architecture]" "DFE4F2274D4D2C43" - .CHK "AH(tl0UJDDxAbJ(td8VJDD4B" - ."~810000000X00[Link modification date]" "2011/02/06 1:22:26 AM" - ."~920000000b40[Link Modifier]" "V(WEIeZIDT4B" - ."~b20000000L60[LinkLanguageUpdateDate]" "2011/02/06 01:22:26" - ."~C3cm9FyluS20[Link Comment]" "g3TCfAJnyq00" 00680SbnxCMPqRc5SN6bpSsvXS6DfCZ5dN38rPcLaN31cPcLaRc5iC35dUpOpRsPST7HkUsnY 00680C6PSRcPSN6nfQ6DcSt9XC7HbKqqWQ5CWR6nbR4GWVJjd2WrzQNPSQtTbP6vfTLmqN35Z 00362Sc5mPbnaPbmmC39pR68WR69XS5nX3N9X3NqA000A .

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of object 1
- IdAbs of object 2
- In the case of the MGL, the IdAbs of the user that made this command.

The MetaAttributes “Link modification date”, and “Link modifier” can be modified just like standard MetaAttributes. If they are not specified in the command, they automatically take the file import date and the IdAbs of the user importing the file.

Deleting a Link

Syntax	***.Disconnect ."Object type" "Object name 1" ."MetaAssociationEnd 2" "Object name 2" -*** ***.CHK "..."***
Example	.Disconnect ."~MrUiM9B5iyM0[Application]" "DFE4E02F4D4D2BB3" ."~mi54NLnHzCj0[Application within Internal Architecture]" "DFE4F2274D4D2C43" - .CHK "AH(tl0UJDDxAbJ(td8VJDD4B"

In this command, the ".CHK" is the concatenation of the following IdAbs:

- IdAbs of object 1
- IdAbs of object 2
- In the case of the MGL, the IdAbs of the user that made this command.

Deletion of a link results in loss of link MetaAttributes values.

– *If this link is used to build a namespace, it must be completed by modification of the local name of the "namespaced" object to maintain repository consistency. Its namespace has become "[0000000000000000]".*

Managing Translations

For each language supported by **HOPEX**, two MetaAttributes indicate the last modification date of translations in a language:

- "[LanguageUpdateDate (Language)]" for an object
- "[LinkLanguageUpdateDate (Language)]" for a link

These MetaAttributes are managed following the same rules as the"Modification date" and "Link modification date" MetaAttributes:

- They can be modified in the same way as standard MetaAttributes.
- If they are not specified in the command modifying a translation, they automatically take the file import date.

Validating Import

Syntax	***.Validate***

This command does not contain a .CHK and produces an intermediate save operation at import.

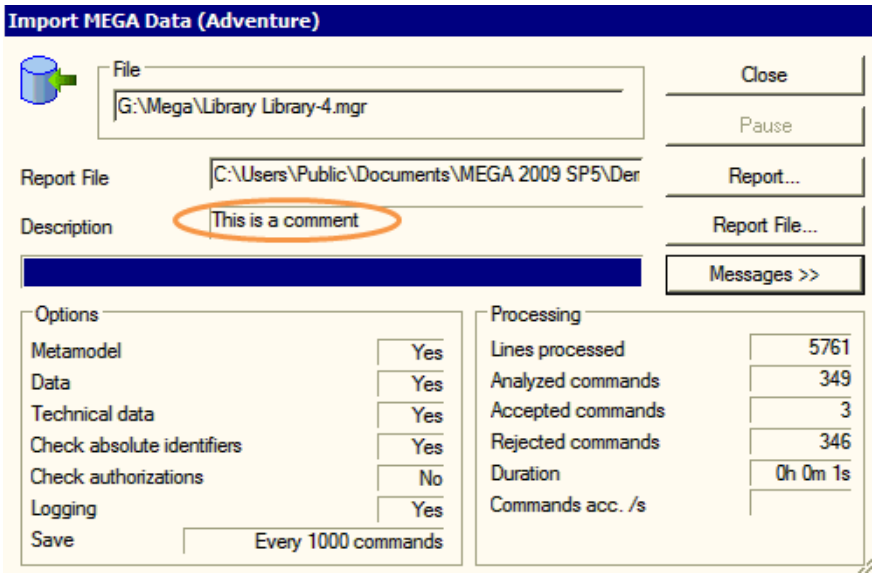
– *This command invalidates save operation selection made by the user interface. For example, to validate consistency of a command file, the user generally imports with "Save Never". Commands are then saved until the last ".Validate" of the file.*

Displaying a Comment in the Import Dialog Box

Syntax	***.Description 'Text'***
Example	.Description 'MetaClass: Acceptance Criteria'

This command does not contain a .CHK.

The text appears on the user interface at import.



Transforming an MGL File to MGR

- *See Command file extensions.*

You do not need to transform an .mgl file to .mgr.

To obtain the same result, when importing an .mgl file:

› in the data import dialog box, in the **Checks** frame, clear the **Check Writing Accesses** check box.

Transforming an MGR File to MGL

- *See Command file extensions.*

You do not need to transform an .mgr file to .mgl.

To obtain the same result, when importing an .mgr file:

› in the data import dialog box, in the **Filter** frame, select the **Reassign User** check box.
Each command is then processed as if its CHK contained the IdAbs of the importing user. Writing access checks are carried out related to its rights.

- *At import in the CHK of an MGL command, the "Reassign User" check box also allows substitution of the IdAbs of the user by that of the person importing.*

MANAGING OPTIONS

This section presents the various tools and options used to configure and customize **HOPEX**.

The following points are covered here:

- Options Overview
- Option Window Presentation
- Accessing Options
- Generating the options Report
- Available Option Groups
- Managing Languages
- Managing Date Format
- Managing HOPEX Data Customization

OPTIONS OVERVIEW

HOPEX options concern:

- site technical configuration.
- values proposed by default for each function of **HOPEX**. These values can be modified by users on each workstation.
  This configuration is described in the guides covering each function.

HOPEX options are accessible at several levels. **HOPEX** functions can be configured at the following levels:

- site

) *The site is the location where **HOPEX** is installed; it is the root of the application.*
- environment
- profile (which groups a configuration common to several users)
- user
- workstation

There is by default an inheritance mechanism between these different levels (excepting workstation level):

- the environment inherits options define at site level.
- the profile inherits options defined at environment level.
- the user inherits options defined at connection profile level.

Customizations made at user level are of highest priority, followed in order of priority by those made at profile, environment and site levels.

P **Having modified option values, it is recommended that you dispatch or save your work, close HOPEX and then reopen it. Refresh issues can occur if these precautions are not taken.**

For detailed information on these options, see the context-sensitive help in the lower part of the window.

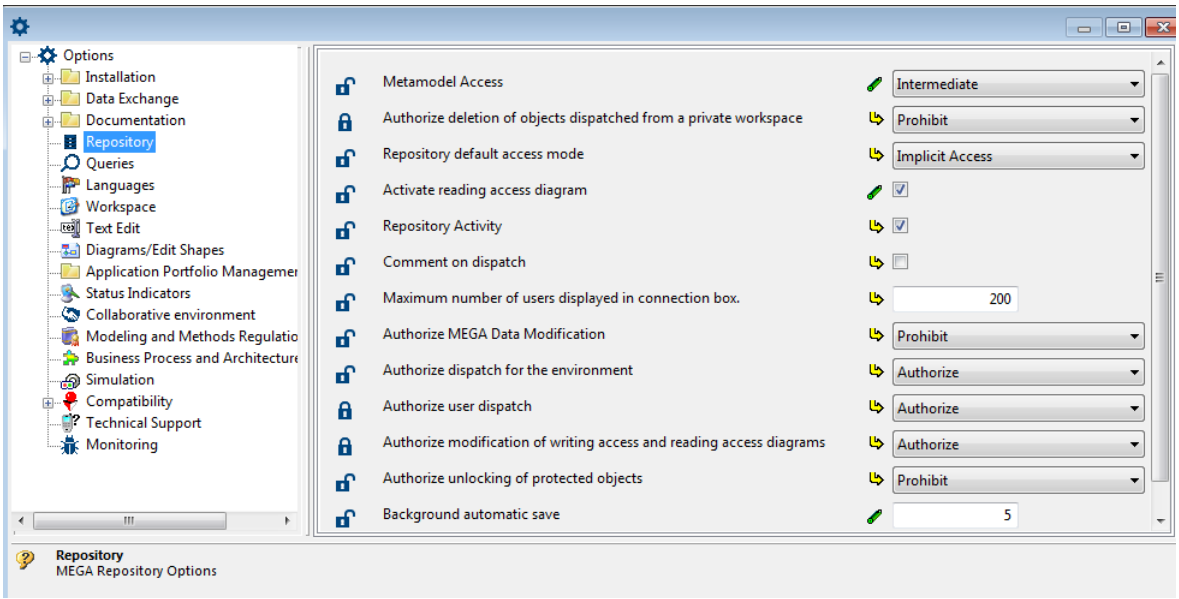
OPTION WINDOW PRESENTATION

The left pane of the window contains the various option groups. It comprises two parts:

- options available for the site, environment, profile, and user
  - *See User options.*
- options specific to the workstation
  - *See Workstation Options.*

The right pane enables configuration of the various options corresponding to the group selected in the left pane.

- *Options vary depending on products you have available.*



For more details on an option:

› Click the name of the option to display the context-sensitive help in the lower part of the window.

- *When the user has a private workspace in progress, you cannot modify its options from **HOPEX Administration**.*

ACCESSING OPTIONS

Options Level

You can modify options at the following levels:

- site
- environment
- profile
- user
- workstation

Modifying options at site level

To modify options at site level:

1. Start **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. In the navigation tree, right-click the site name and select **Options > Modify**.
   The site options window opens.

Modifying options at environment level

To modify options at the environment level from **HOPEX Administration** :

1. Using **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Right-click the environment name and select **Options > Modify**.
   The environment options window opens.

Modifying options at profile level

To modify options at profile level from **HOPEX Administration**:

1. Using **HOPEX Administration**, connect to the environment concerned.
   - *See Connecting to an Environment.*
2. Open the manage profiles dialog box.
   - *See Opening the profile management window.*
3. In the **Profile** tab , right-click the profile and select **Options**.
   The profile options window opens.

Modifying options at user level

You can modify user options from:

- **HOPEX Administration**
- **HOPEX**

To configure user options from **HOPEX Administration**:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Open the user management window.
   - *See Opening the User Management window.*
3. Select the **Persons** tab.
4. Right-click the desired person and select **Options**.
   The user options window opens.

To configure user options from **HOPEX**:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In the **HOPEX** menu bar, select **Tools > Options**.
   The user options window opens.

Modifying options at workstation level

To modify options at workstation level:

1. Connect to **HOPEX Administration**.
   - *See Accessing HOPEX Administration.*
2. Right-click **Workstation** and select **Options**.
   The workstation options window opens.

Each option can take several values.

Option Inheritance

An option inherits a value defined at a higher level:

- A user inherits options defined at the connection profile level.
- A profile inherits options defined at the environment level.
- An environment inherits options defined at the site level.

The icon located opposite the option indicates the inheritance, or not, from the higher level:

- **Default value** indicates the inheritance from the higher level.
- **Modified value** indicates that the inherited option value has been modified. The value is no longer inherited from the higher level.

To specify that an option does not inherit the value defined at higher level:

1. Open the options page.
   - *See Options Level.*
2. Click **Default value**.
   The icon changes in **Modified value**.

Controlling the Modification of the Options

You can prohibit modification of any option at a level lower than your current level.

```
Example: if you open options of the environment, you can
prohibit modification of all options at user level.
```

Prohibiting modification of a lower level option

To prohibit modification of a lower level option:

1. Access the options.
   - *See Options Level.*
2. Click icon located opposite the option concerned.

   The padlock closes : option modification by a user is now prohibited from **HOPEX**.

Unlocking the modification of a lower level option

To unlock modification of a lower level option:

1. Access the options.
   - *See Options Level.*
2. Click the closed padlock icon .

   The padlock opens : modification of the option is again possible.

Reinitializing Option Values

You can reinitialize the values for:

- an option
- an option group

Reinitializing the values of an option

To reinitialize the value of an option:

1. Access the options.
   - *See Options Level.*
2. Click:
   - (Web Front-End) **Reinitialize**
   - (Windows Front-End) **Modified Value** , the icon changes to **Default Value**

   The value of the option is reinitialized.

Reinitializing the values of an option group (Windows Front-End)

To reinitialize values of an option group from **HOPEX Administration**:

1. Access the options.
   - *See Options Level.*
2. In the options tree, right-click the option group and select **Reinitialization**.
   All the options in the group selected are reset to their default values.

GENERATING THE OPTIONS REPORT

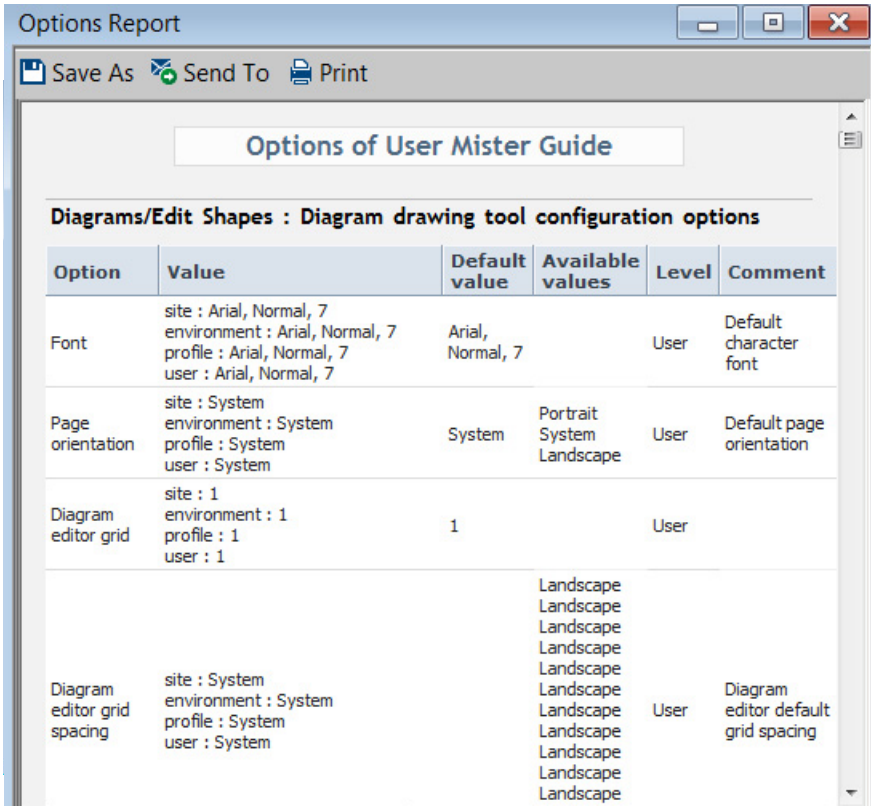
You can generate a report that lists all options classified by group, with their comments.

To generate the list of options:

1. Access the Options at the required level (site, environment or user).
   - *See Modifying options at user level.*
2. Right-click **Options** and select **Report**.
   Report generation may take some time.

The Options Report contains:

- the names of available options
- the values available for each option
- the default value
- a comment explaining the option use context
- the option level:
  - user
  - environment
  - site



To save this report in .html format:

› Click **Save As** and select *.htm format.

AVAILABLE OPTION GROUPS

User options

– *At user configuration level, certain options are grayed. They can be defined only for an environment or site and not for a user.*

Note that repository and modeling options contain important information for the functional administrator.

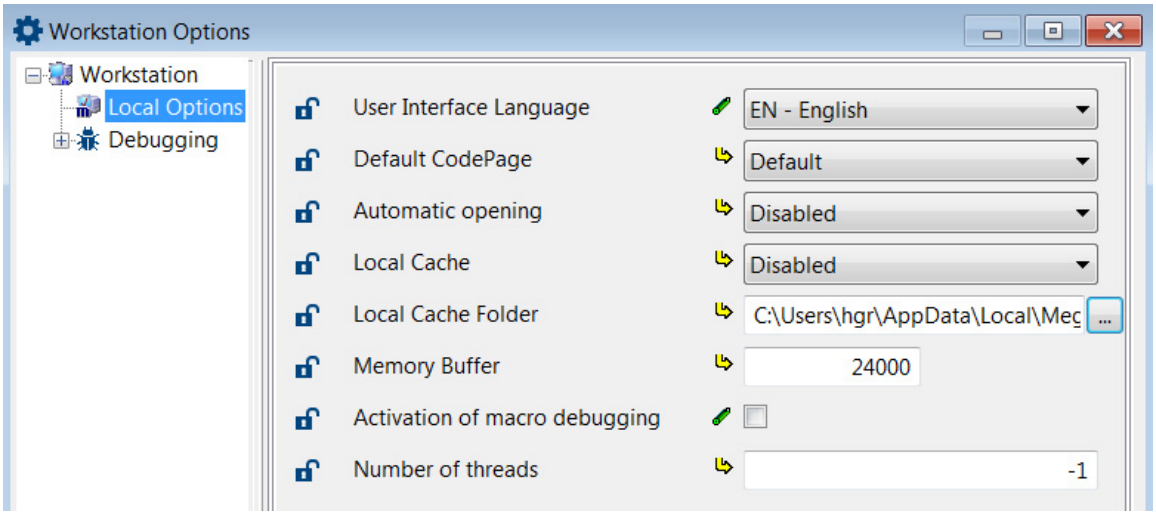
- **Installation**
  Options linked to installation: licenses, information on the company, Web application (options linked to the **HOPEX** user workspace (Web Front-End)
- **Data Exchange**
  Options linked to import/export, exchanges with third party tools.
- **Documentation**
  Options linked to documentation generated by **HOPEX** (reports (MS Word), reports (Open Office), Web sites, Description, reports, performance indicators)
- **Repository**
  Options authorizing or prohibiting access to certain repository functions.
- **Queries**
  Options linked to the query tool
- **Languages**
  Activated data languages
- **Workspace**
  Options linked to the user workspace of **HOPEX** (Windows Front-End). They enable display of a certain number of functions or not, as well as management of user inactivity or not.
- **Text Editing**
  Options concerning RTF format comment entry
- **Diagrams/Edit Shapes**
  Options of drawing tool configuration (diagrams and shapes editor)
- **Status indicators**
  Options concerning display of indicators available in workspace and diagrams
- **Collaborative Environment**
  Options concerning collaborative work in **HOPEX**
- **Mapping Editor**
  Options linked to the mapping editor, a tool enabling alignment of data models (essentially with **HOPEX Database Builder**)
- **Modeling and Methods Regulations**
  Options linked to modeling regulations and rules

- **Business Process and Architecture Modeling**
  Options linked to processes and architecture enabling display of certain functions
- **Simulation**
  Options enabling definition of **MEGA Simulation** use level
- **Compatibility**
  Options of compatibility concerning diagrams and obsolete functionalities
- **Technical Support**
  Options concerning access to Technical Support
- **Monitoring**
  Option used to supervise data access

Workstation Options

The local options group contains information specific to the workstation.

– *This information is stored in "MEGAWKS.INI" file (C:\ProgramData\MEGA\HOPEX <Version>\Cfg).*



MANAGING LANGUAGES

The following points are detailed here:

- Changing User Interface (Windows Front-End) Language
- Defining the Data Languages Available for a User
- Changing User Data Language (Windows Front-End)
- Installing Additional Languages
- Defining the Language of e-mails in Workflows
- Managing Languages in Web Applications

Changing User Interface (Windows Front-End) Language

You can change interface language of a **HOPEX** (Windows Front-End) user.

– *To modify the language of Web applications, see Managing Languages in Web Applications*

To change the interface language of a **HOPEX** (Windows Front-End) user:

1. Connect to **HOPEX Administration**.
   – *See Accessing HOPEX Administration.*
2. In the **HOPEX** tree, right-click **Workstation** and select **Options**.
3. In the **Workstation** tree, select **Local Options**.
4. In the right pane, modify the value of the **User Interface Language** option.
5. Click **OK**.
   The change is effective at next restart of **HOPEX** (Windows Front-End).

Defining the Data Languages Available for a User

You can also select the languages available in **HOPEX** and in which you can enter data.

– *See Installing Additional Languages.*

So that a **HOPEX** workstation can be used in multilingual mode, multilingual mode must be authorized for the site.

– *When you duplicate an object, it must be in the repository language so that translations in the various languages will be correctly transferred to the duplicates.*

To determine the language of your repository, consult repository properties.

– *For more information on the use of languages, see the **HOPEX Common Features** guide, "**HOPEX** in a Multilingual Context" section.*

To define the data languages available for a user:

1. Connect to **HOPEX Administration**.
   – *See Accessing HOPEX Administration.*

2. Access the options management window.
   - *See Modifying options at environment level.*
   - *See Modifying options at profile level.*
   - *See Modifying options at user level.*
3. In the options tree, select **Languages**.
4. In the right pane, select the languages available for the interface.
5. Click **OK**.

Changing User Data Language (Windows Front-End)

In **HOPEX** (Windows Front-End), the user can change his/her data language himself/herself.

- *To change user data language in Web applications, see Managing Languages in Web Applications.*

To modify the data language from **HOPEX**:

1. Connect to **HOPEX**.
   - *See Connecting to HOPEX.*
2. In the menu bar, select **Tools > Languages**.
3. Select the data language.
   - *Translated data appears in the selected language.*
   - *To install additional languages, see Installing Additional Languages:*

Installing Additional Languages

To install additional languages in **HOPEX**:

1. Connect to **HOPEX Administration** and select the environment concerned.
   - *See Connecting to an Environment.*
2. Right-click the environment (or site) and select **Metamodel > Install Additional Languages**.
3. The dialog box **Install Additional Languages** opens.
4. Select the languages you want to have available in **HOPEX**, and click **OK**.
   A window indicates progress of import of the corresponding libraries.
   The additional languages are accessible from the **Tools > Language** menu of the **HOPEX** desktop.

Defining the Language of e-mails in Workflows

To define the language of e-mails in workflows:

1. Access the options management window.
   - *See Modifying options at environment level.*

2. In the options tree, expand the **Installation** folder and select **Workflows**.
3. In the right pane, in option **Language for sending mails**, select the language to be defined in e-mails.
4. Click **OK**.

Managing Languages in Web Applications

You can modify:

- the interface language in Web applications
- the data language in Web applications.

– *To manage languages in Web applications, see the HOPEX Administration - Supervisor Web guide.*

MANAGING DATE FORMAT

In **HOPEX**, the date format depends on the data language.

This format is defined for each language in the Windows parameters of **HOPEX** installation server.

If needed you can change this format in **HOPEX**.

- *This customization might be lost at HOPEX update.*

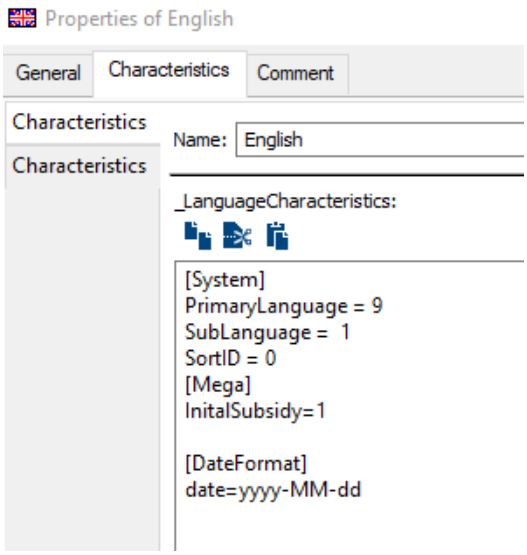
P **This modification uncompile the technical data.**

To change the date format for a language:

1. Connect to **HOPEX**.
   - *Check that you are in "Expert" metamodel access and that you are allowed to modify HOPEX data (**Options > Repository**).*
2. In **HOPEX** search toolbar, select **Language**.
3. Click **Find** .
4. In the result list, right-click the language concerned and click **Properties**.
5. In the **Characteristics**, tab, select the first **Characteristics** sub tab.
6. In the **_LanguageCharacteristics** pane, add the date format you want to be customized:

```
[DateFormat]
date=<date format>
```

You can use separating characters like: "/", ",", "-", or " ".



```
Examples:
date=yyyy/MM/dd displays 2018/04/24
date=d-MM-yy displays 4-03-18
date=dd MMMM yy displays 04 july 18
date= dddd, MMMM d, yyyy displays Tuesday, June 5, 2018
```

7. Click **OK**.
   The date format is changed in **HOPEX** for the language concerned.

   P **This modification uncompile the technical data.**

8. Compile the technical data.

   - *See Compiling an Environment.*

Date Format	Description
d	The day of the month with one or two digits 1...9, 10, 11,..31
dd	The day of the month with two digits 01...09, 10, 11,..31.
ddd	The abbreviated name of the day of the week
dddd	The full name of the day of the month
M	The numeric format month with one or two digits 1...9, 10, 11, 12
MM	The numeric format month with two digits 01...09, 10, 11, 12
MMM	The abbreviated name of the month
MMMM	The full name of the month
y	The year with one or two digits 9,18
yy	The year with two digits 09, 18
yyyy	The year with four digits 2018

MANAGING HOPEX DATA CUSTOMIZATION

To ensure a correct use of **HOPEX**, by default it is forbidden to modify **HOPEX** data. Modifying a **HOPEX** object may generate errors at **HOPEX** upgrades, import of correctives, etc.

The **Authorizing HOPEX Data Modification** option allows modifying the **HOPEX** metamodel or any other **HOPEX** technical object.

P **This option should only be selected in certain highly specific cases, at debugging operations or at MEGA request, and for a temporary period.**

This option is:

- accessible with "Extended" access to options
  - *In the **Options** tree, right-click **Options** and select **Extended**.*
- locked by default at environment level, with "Prohibit" value
  - *See Controlling the Modification of the Options.*
- accessible in the **Options > Installation > Customization** folder.

P **Specify this access level only for a highly expert user or highly advanced profile.**

FREQUENTLY ASKED QUESTIONS

The following points are covered here:

- Common Operations
- Recurrent Messages

COMMON OPERATIONS

How do I copy a repository from one environment to another?

Standard procedure: make a logical backup of the repository, then carry out a logical restore of the backup in an empty repository of the target environment.

For GBMS environments you can copy repository files (EMA, EMB, EMS, EMV) in a folder carrying the repository name, then create a reference for the repository in the second environment, but only if the metamodel is exactly the same in both environments.

Can I create a reference for an environment in another site?

No, the functional rule is that a reference for a HOPEX environment should only be referenced in an installation (site).

Can I delete a user?

Yes, you can delete a user. See Deleting Users.

P **When you delete a user from the repository, all actions linked with this user are lost.**

To delete a user but retain its actions, modify user repository access mode to **Inactive user** (see Modifying the Properties of a User). The user no longer appears in the connection dialog box, but its actions are kept.

- *Note that you cannot delete the "Administrator" user, or the "Administrator" writing access.*

Can I delete a writing access area?

Yes, you can delete a writing access area.

P **When you delete a writing access area, the objects that were attached to it pass implicitly to "Administrator" writing access level.**

MEGA recommends that before deletion, you modify the writing access of objects attached to the writing access area.

RECURRENT MESSAGES

Abnormal operation when refreshing a private workspace

Symptom: Message stating "Could not refresh your private workspace".

Reason: Rejects occurred when importing private workspace updates into the reference repository.

Solution:

1. Examine the reject file Rmmjj.MGL (eg: MGLR07150000.MGL) in the user work repository (<repository>\USER\<user code>).
2. Identify and process causes of rejects (see Rejects When Dispatching).
3. Delete reject files that are no longer needed.

   – *For as long as reject files are not deleted, a warning persists when connecting to HOPEX.*

Environment version

Symptom: Message "Your environment and site are not of the same version" when opening an environment from the HOPEX administration console" (or "Your environment and site are not of the same version. Your environment requires updating. Refer to documentation for how to carry out this action").

› Click **OK**.
Another window appears displaying a second message: "Your environment requires an update for compatibility with your version of HOPEX". Do you wish to run this procedure now? ".

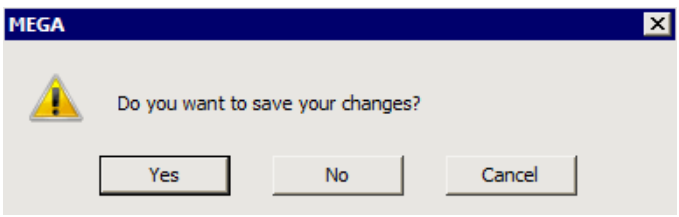
Reason: It is possible that the environment has not been created, or is not at the same version level as the site referencing it: an update is therefore proposed:

› if you have a physical backup of environment data, accept modification by clicking **Yes**.

› If this is not the case, refuse the modification by clicking **No** to exit the HOPEX data administration console. Then execute physical backup of data.

"Later" option not proposed at disconnect

Symptom: When you exit **HOPEX**, the dialog box that appears does not propose the "Later" option to save your modifications.



Reason:

- You are not connected to **HOPEX (Windows Front-End)** and the license used does not have technical module **MEGA Lan**.

  P **You prevent other users from dispatching their work performed in their private workspace.**

- You are connected to **HOPEX (Web Front-End)**, in a public workspace.

PRODUCT CODES

Access your list of available products

To view the products to which you have access:

1. Start **HOPEX Administration**.
   - *See Accessing Repositories.*
2. In the menu bar, select **Help > About HOPEX**.
3. Click **System Information**.
4. With the drop-down menu, select **Available components**.
   All the products for which you have a license are listed as well as their associated code.
   - *For information on product availability (Windows Front-End, Web Front-End) and storage, see online documentation (**Concepts > Products**).*

GLOSSARY

absolute identifier

An absolute identifier is a string of characters associated with each object in the repository. This string is computed using the date and time the session was opened, the number of objects created since the session started, and the object creation date (in milliseconds). An absolute identifier provides a unique way to identify an object in the repository, so that even if the object is renamed, all the links to it are retained.

access area member

Access area member groups all persons and person groups belonging to an access area. This area defines objects that can be accessed by the person or person group.

access path

An access path indicates which folder you can use when creating a reference for an environment database or a site environment. When all repositories are created in the same location as the environment, the path created at installation (the DB folder under the environment root) is sufficient and is given as the default. When you want to save a repository in a folder other than that of the environment, you must declare a new access path.

access rights

User access rights are the rules that manage access to software functions and databases. You can restrict the access rights of a user to the different repositories defined in his/her work environment. You can also restrict what software features a given user can run, such as the descriptor editor, modifying queries, designing report templates (MS Word), deleting objects, dispatching private workspaces, importing command files, managing environments, repositories, users, writing access, etc.

administration Administration consists of managing the work environment of repository users. This function is usually the responsibility of the administrator. Administration tasks include making backups of repositories, managing conflicts in data shared by several users and providing users with queries, descriptors, report templates (MS Word), etc. common to several projects.

Administration desktop The **HOPEX Administration** desktop (Web Front-End) is the Web version of the **Administration** (Windows Front-End) application accessible via an internet browser.

administrator The administrator is a person who has administration rights to manage sites, environments, repositories and users. In addition to Administrator (who cannot be deleted) and MEGA users, created at installation, you can grant administration rights to other users.

attribute See *Characteristic*.

backup logfile The backup logfile is an additional file stored outside the repository or private workspace. It ensures that the changes made to the repository or private workspace can be recovered even if the repository files become corrupted. Creation of this file is requested in the configuration of each repository (including the system repository). It is created when the first update is made in a private workspace or repository.

business role A business role defines a function of a person in a business sense. A person can have several business roles. A business role is specific to a repository.

characteristic A characteristic is an attribute that describes an object or a link. Example: the Flow-Type characteristic of a message allows you to specify if this message is information, or a material or financial flow. A characteristic can also be called an Attribute.

command file A command file is a file containing repository update commands. It can be generated by backup or object export (.MGR extension) or by logfile export (.MGL extension).

comparison You can compare objects in two repositories, creating a file that will modify objects in one repository to make these equivalent to objects in the other repository. This comparison also allows you to list the differences between the contents of the two different objects.

compilation

Compilation is carried out after migration or customization. Compilation checks configuration of the environment concerned. When completed, processing for all users of this environment is speeded up. Metamodel compilation includes in parallel translation in the current language. You can also translate the metamodel into another language.

consolidation

Consolidation groups the updates from stand-alone workstations or remote sites (with Lan) and merges them in a reference site. After dispatch of the private workspaces of each of the users, the repository log is exported and reinitialized. The logfiles are imported into the reference repository, then this is recopied on each of the user sites.

database

See *repository*.

description

Descriptors allow you to create reports (MS Word) containing part of the contents of the repository. Descriptor for an object includes the object characteristics, to which can be added the characteristics of objects directly or indirectly linked to it. The readable format for each of the objects encountered is entered as text in Word for Windows. You can insert descriptors into reports (MS Word) or report templates (MS Word) or use them to produce reports. Descriptors can be created or modified using the **HOPEX Power Studio** technical module.

desktop

The desktop specific to each user contains projects, diagrams, reports (MS Word), etc. handled by this user. The user has a different workspace in each of the repositories he/she accesses.

discard

Discarding the work performed in a private workspace cancels all modifications made since the last dispatch. All the work you have done since the beginning of the private workspace is lost. A warning message reminds the user of this. The user can request the discard of his/her private workspace from the **File > Discard** menu or at disconnection.

environment

An environment groups a set of *users*, the *repositories* on which they can work, and the *system repository*. It is where user private workspaces, users, system data, etc. are managed.

external reference

An external reference enables association of an object with a document from a source outside **HOPEX**. This can relate to regulations concerning safety or the environment, legal text, etc. Location of this document can be indicated as a file path or Web page address via its URL (Universal Resource Locator).

functionality A functionality is a means proposed by the software to execute certain actions (for example: the shapes editor and the descriptor editor are functionalities proposed as standard).

general UI access General UI access defines if tools are available or not. By default, general UI accesses have value *A (A: Available, *: default value)

importing Importing a command file, a backup file, or a logfile consists of applying the commands in the file to this new repository.

LDAP parameter An LDAP parameter is a parameter that exists in the LDAP directory and is associated uniquely with a **HOPEX** attribute. For example, "mail" is a parameter of the "Active Directory" LDAP directory and can be associated with the e-mail of the person.

LDAP server The LDAP server is the server on which the LDAP directory is installed. The LDAP directory can be an Active Directory directory.

link A link is an association between two types of object. There can be several possible links between two object types, for example: Source and Target between Org-Unit and Message.

link orientation The two objects connected by a link do not normally have symmetrical roles. For example, to connect an operation to an organizational process with the **HOPEX Power Supervisor** technical module, you must be authorized to modify the organizational process, since this action will modify its behavior. This action will not however modify the operation. You do not need authorization to modify the operation to create this link. The organizational process is said to be major for this link, the operation is minor. This characteristic is used by administration tools for object export, protection, object comparison and querying isolated objects.

lock A lock is a logical tag assigned to an object to indicate that it is currently being modified by a user. Simultaneous access to an object by several users can also be checked. Locks apply to all types of object. When a user accesses an object to modify it, a lock is placed on the object.

When a lock is placed on an object, another user is only able to view the object. This second user will not be able to modify the object until the first user dispatches his/her work, and the second user refreshes. This is done to avoid conflicts between the state of the object in the repository, and the obsolete view of the second user.

logfile Logfiles contain all the actions performed by one or more users over a given period. The private workspace log contains all the changes made by a user in his/her private workspace. This logfile is used to update the repository when the user dispatches his work. For additional security, it is also possible to generate another log called the backup logfile.

logfile export Export of a logfile creates a command file from the logfile of user actions in a repository. You can keep this file and import it later into a repository. You can selectively export modifications made in the work repository, or those made to technical data (descriptors, queries, etc.) in the system repository.

login A Login uniquely defines a user or user group. It can be assigned to only one Person or Person Group.

major The major object in the link is the one whose nature changes with the presence or absence of the link. For example a process, defined as a succession of operations, is modified if you remove an operation. The process is then major for the link. If the objects are protected, you must have the correct authorization for modifying the major object in order to create or delete the link.

matrix A matrix is a table comprising rows and columns containing objects from the repository. Matrices show the relationships between two sets of objects and allow you to create or delete links without having to open the diagrams themselves. For example, you can build a matrix showing the messages sent by the different org-units in a project.

MetaAssociation see "link".

Metaclass see object type

Metamodel The metamodel defines the language used for describing a model. It defines the structure used to store the data managed in a repository The metamodel contains all the MetaClasses used to model a system, as well as their MetaAttributes and the MetaAssociations available between these MetaClasses. The metamodel is saved in the system repository of the environment. You can extend the metamodel to manage new MetaClasses. Repositories that exchange data (export, import, etc.) must have the same metamodel, otherwise certain data will be rejected or inaccessible.

minor The minor object in a link is the one whose nature is not modified or only slightly modified by presence or absence of this link. For example, removing an operation from a process does not change characteristics of this operation. Therefore the process is minor in the link.

model A model is a formal structure which represents the organization of a company, or its information system. In another sense, a model can be a template for reproducing objects with similar characteristics. This is the case for report templates (MS Word) and matrix models.

object An object is an entity with an identity and clearly defined boundaries, of which status and behavior are encapsulated. In a **HOPEX** repository, an object is often examined along with the elements composing it. For example, a Diagram contains Org-Units or Messages. A Project contains Diagrams, themselves containing Org-Units, Messages etc. Database administration frequently requires consideration of consistent sets of objects. This is the case for object export, protection and object comparison.

object export The export of one or several objects enables transfer of a consistent data set from a study to another repository. For example, export of objects from a project includes the project diagrams, together with the objects these diagrams contain, such as messages and org-units, as well as dependent objects. All the links between objects in this set are also exported.

Object type An object type (or MetaClass) is that part of the database containing objects of a given type. The objects created are stored in the repository according to their type. Segments are used when searching for objects in the repository and when the metamodel is extended to include a new type of object. Example: message, org-unit, etc.

object UI access Object UI access defines user rights on creation, reading, update, and deletion on these objects and their tools. By default, object UI accesses have value *CRUD (C: create, R: read, U: update, D: delete, *: default value).

person A person is defined by his/her name and electronic mail address.

A person can access **HOPEX** once the administrator assigns him/her a login and a profile.

The list of persons can for example come from an LDAP server.

Person group (Web Front-End specific) A person group groups persons in a group. These persons share the same connection characteristics.

private workspace

A private workspace is a temporary view of the repository in which the user is working before dispatching his/her work. This view of the repository is only impacted by the changes of the user, and does not include concurrent modifications made by other users. This private workspace exists until it is refreshed, dispatched or discarded. Note that a private workspace is kept when the user disconnects from the repository, unless the user indicates otherwise. A user can see modifications dispatched by other users of this repository without dispatching his/her own modifications. To do this, the user refreshes his/her private workspace. The system then creates a new private workspace, and imports the logfile of his/her previous modifications into it.

private workspace log

The private workspace log contains all modifications made by a user in his/her private workspace. It is applied to the repository at dispatch, then automatically reinitialized. This log is stored in the EMB private workspace file.

profile

A profile defines what a person can see or not see and do or not do in tools, and how he/she sees and can do it. The profile defines options, access rights to repositories and products, read/write and read-only rights on objects.

All users with the same profile share these same options and rights. A user can have several profiles. A profile is available for all repositories in a single environment.

profile assignment

The profile assignment defines the following for each person: the repository concerned by the assignment, access rights to the repository, the validity period of the assignment, (optional, with access to the repository in read only) connection repository snapshot

protection

When several users work on the same project, it is important to provide them with the means to work on a new part of the project without inadvertently interfering with previous work. To do this, you can protect the objects concerned by assigning to them a writing access area (**HOPEX Power Supervisor** technical module). It is then possible to connect these objects to others, if the link does not modify the nature of the object. The link orientation determines whether or not the nature of the object is affected.

publish

Dispatching your work allows the other users to see the changes you have made to the repository. They will see these when they open a new workspace, either by dispatching, refreshing or discarding their work in progress

query A query allows you to select a set of objects of a given type using one or more query criteria. Most of the MEGA software functions can handle these sets. For example, you can use a query to find all enterprise org-units involved in a project.

reading access see "reading access area".

reading access area The user reading access area corresponds to the view the person or person group has of the repository: it defines objects that can be accessed by the person or person group.

reading access diagram The reading access diagram enables definition of reading access areas and their hierarchical organization. This diagram also enables creation of users and their association with reading access areas.

reflexive link A reflexive link is a link between two objects of the same type, for example: the link between projects that allows you to define sub-projects.

refresh Refreshing his/her private workspace allows the user to benefit from changes dispatched by other users since creation of his/her workspace. In this case, it keeps repository modifications carried out by the user without making these available to other users. The system creates a new private workspace and the logfile of previous changes is imported into it.

reject file When updating a repository (importing, restoring, dispatching), a reject file is created in order to store rejected commands. Rejected commands are stored with the reason for which they were rejected. These are found in the "MegaCrd.txt" file located in the environment folder.

reorganization Reorganizing a repository consists of executing a logical backup of the repository, reinitializing it and reimporting the logical backup (without log).

report (MS Word) Reports (MS Word) managed by **HOPEX** are objects allowing you to transfer written knowledge extracted from the data managed by the software.

report (MS Word) element A report (MS Word) element is the instancing of a report template (MS Word) element. It is the result, formatted in the word processing software, of execution of the query defined in the report template (MS Word) element.

report file

The environment report file, MegaCrdYYYYmm.txt (where YYYY and mm represent year and month of creation) indicates all administration operations (backup, export, restore, controls, etc.) carried out in the environment. The report file is stored in the user work folder associated with the environment system repository: SYSDB\USER\XXX\XXX.WRI where XXX is the user code.

report template (MS Word)

A report template (MS Word) is a structure with characteristics that may be reproduced when producing reports (MS Word). A report (MS Word) can be created and modified as many times as necessary; however to produce several reports (MS Word) of the same type, use of a report template (MS Word) is recommended.

A report template (MS Word) provides the framework for the report (MS Word), which is completed with data from the repository when the report (MS Word) is created. A template contains the formatting, footers, headers and text entered in MS Word. It also contains report template (MS Word) elements that allow you to format data from the repository. It allows you to produce reports (MS Word) associated with main objects of the repository.

report template (MS Word) element

A report template (MS Word) element is the basic element of a report template (MS Word). It comprises a query which enables specification of objects to be described and a descriptor for their formatting. When creating a report (MS Word) from a report template (MS Word), each report template (MS Word) element is instanced by a report (MS Word) element.

Reporting Datamart

A Reporting Datamart is a replicated RDBMS Database from an HOPEX repository content.

The Reporting Datamart is made up of data selected at creation and synchronized on regular basis, to keep the Reporting Datamart updated according to the HOPEX repository content.

The Reporting Datamart feature is to be used as a source for any usage that needs HOPEX data (for example: reporting).

repository

A repository is a storage location where HOPEX manages objects, links, and inter-repository links.

The main part is managed by a database system (SQL Server). The remainder is in a directory tree (content of Business Document versions, backup logfiles, locks.

The different users in the environment can access the repositories connected to it.

repository log

The repository log stores all the updates of users working in a repository. It is reinitialized at repository reorganization , or by selecting **Repository Log > Manage Repository and Object Log** in the repository pop-up menu. This logfile is stored in the .EMB file of the repository.

repository snapshot

A repository snapshot identifies an archived state of the repository.

Creating a repository snapshot allows you to label important states in the repository life cycle.

The repository archived states for which a snapshot exists are not deleted by repository cleanup mechanisms (Repository history data deletion).

restore

A physical restore consists of copying previously saved repository files.

saving

The work done in a session is saved when you request it, or when you exit. By default, the software executes an automatic save (the time interval between two saves is specified in the options: **Options > Repository > Background Automatic Save**). Messages appear asking you to confirm saving the changes you made in each open report template (MS Word) or report (MS Word) (except during the automatic save). It is recommended that you regularly save your session to avoid losing your work if your computer locks up or loses power.

session

A session is the period during which a user is connected to a repository. A session begins when the user authenticates and ends when he/she exits **HOPEX**. Sessions and private workspaces can overlap. When you dispatch, refresh or discard a private workspace, a new private workspace is created in the same session. Conversely, a user can keep his/her private workspace when exiting a session.

set

A set is a collection of objects with common characteristics. For example, you can build a set of messages sent or received by a certain org-unit in the enterprise. These sets are usually built using queries, and can be handled by most functions of the software.

setting

A setting is a parameter of which value is only determined when the function with which it is associated is executed. You can use variables to condition a query (**HOPEX Power Studio** technical module). When you execute this query, a dialog box asks you to enter these settings, with a box for each setting defined in the query.

site

A site groups together everything that is shared by all **HOPEX** users on the same local network: the programs, standard configuration files, online help files, standard shapes, workstation installation programs, and version upgrade programs. The site is installed on a local network resource or on each workstation if you are working without a network connection.

snapshot

See *repository snapshot*

style

A style is a particular format applied to a paragraph of text in a word processor. Styles allow you to systematically apply formats such as fonts, margins, indents, etc. Several styles are available for report (MS Word) configuration. These styles have the M- prefix and are based on the M-Normal style that is similar to the Word Normal style. Note that the M-Normal style text is in blue. All these styles are contained in the Megastyl.dot style sheet.

SystDb

SystDb is a particular repository containing the metamodel and technical data (descriptors, Web site templates, queries, etc.). The metamodel and technical data are common to all repositories in the same environment. Definition of users and their rights are stored in this repository, essential for operation of the software.

system repository

See *SystDb*.

text

You can associate text with each object found when browsing object descriptors (**HOPEX Power Studio** technical module). This text is formatted for MS Word. It presents what will be displayed for each of the objects in the generated report (MS Word). In the text you can insert the object name, its characteristics, and its comment. You can also insert the characteristics of other objects linked to it.

trace file

The trace file (Megaerr*.txt) is accessible via menu **Help** or from the **HOPEX Server Supervisor** tool (available in the **Utilities** folder). It traces all problems and errors encountered on the workstation. Technical support may ask you to check this file.

user

A user is a person with a login and at least one profile assigned.

The code associated with the user is used to generate file names as well as a specific work folder for the user.

By default at installation, Administrator (Login: System) and Mega (Login: Mega) persons enable administration of repositories and creation of new users.

work folder Each user has a work directory in each repository that he/she uses. This directory is located in sub-directory User\XXX of the repository (XXX represents the user code).

workstation A workstation is defined for each computer connected to the environment. A workstation contains programs and a configuration file that allow you to use **HOPEX** on that machine.

writing access see "writing access area".

Writing access area A writing access area is a tag attached to an object to protect it from unwanted modifications. Each user is assigned a writing access area. There is a hierarchical link between writing access areas. A user can therefore only modify objects with the same or lower authorization level. The structure of writing access areas is defined in the writing access diagram. By default there is only one writing access area, "Administrator", to which all objects and users are connected. Writing access management is available with the **HOPEX Power Supervisor** technical module.

writing access diagram The writing access diagram is available if you have the **HOPEX Power Supervisor** technical module. With this diagram, you can create users and manage their writing access rights for repositories and product functions. By default only one writing access area is defined, named "Administrator". Attached to it are the "Administrator" and "Mega" persons. This is the highest writing access level and it should normally be reserved for repository management. You cannot delete this writing access area.

Technical Articles (EN)

HOPEX ADMINISTRATION CONSOLES

HOPEX Administration Consoles are accessible through a Web browser to any user, who has been granted the rights. It gives access to:

- HOPEX Supervision console
- HOPEX Monitoring console
- HOPEX Licensing console

HOPEX ADMINISTRATION CONSOLES

Introduction to HOPEX Administration Consoles

The **HOPEX Administration Consoles** is the Administration console homepage, which gives access to the following administration consoles:

- **HOPEX Supervision** console
  - *See "HOPEX Supervision console", page 9.*
- **HOPEX Monitoring** console
  - *See "HOPEX Monitoring console", page 41.*
- **HOPEX Licensing** console
  - *See "HOPEX Licensing Console", page 55.*

Accessing the HOPEX Administration Consoles

To access all of the **HOPEX Administration Consoles** you must be granted the authorization to access C:\inetpub\wwwroot\HOPEXAdministration folder.

HOPEX Administration Consoles are independent and you access each of them directly. To access each of the **HOPEX Administration Consoles** you must be granted the authorization to access respectively:

- C:\inetpub\wwwroot\HOPEXSupervision folder.
  - *See "Accessing HOPEX Supervision Console", page 9*
- C:\inetpub\wwwroot\HOPEXMonitor folder.
  - *See "Accessing HOPEX Monitoring Console", page 41*
- C:\inetpub\wwwroot\HOPEXLicenceManager folder.
  - *See "Accessing HOPEX Licensing Console", page 55*

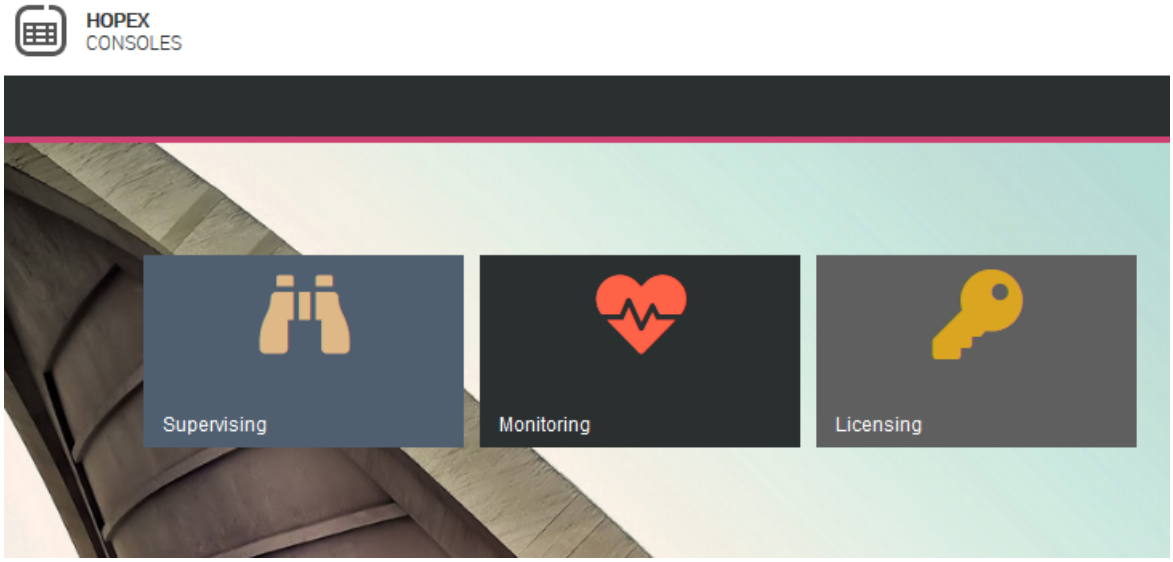
To access the **HOPEX Administration Consoles**:

1. Open your web browser.
2. In the address field, enter "/HOPEXAdministration" after the server address.

   `Example: http://<server address>/HOPEXAdministration`

   A **Windows Security** window appears.
3. Enter your Windows login and password.

4. Click **OK**.
The **HOPEX Administration Consoles** homepage appears.



5. Click:
   - **Supervising** to access the **HOPEX Supervision** console.
   - **Monitoring** to access the **HOPEX Monitoring** console.
   - **Licensing** to access the **HOPEX Licensing** console.

   The corresponding HOPEX Administration console homepage appears.

   – *To switch to another **HOPEX Administration Console**, see "Switching between HOPEX Administration Consoles", page 8.*

Directly Accessing One of the HOPEX Administration Consoles

HOPEX Administration Consoles are independent and can be accessed directly. To access each of the **HOPEX Administration Consoles** you must be granted the authorization to access on the server, respectively:

- C:\inetpub\wwwroot\HOPEXSupervision folder.
- C:\inetpub\wwwroot\HOPEXMonitor folder.
- C:\inetpub\wwwroot\HOPEXLicenceManager folder.

To directly access one the **HOPEX Administration Consoles**:

1. Open your web browser.

2. In the address field, after the server address, enter the corresponding console name:
   - "/HOPEXSupervision"

     `Example: http://<server address>/HOPEXSupervision`
   - "/HOPEXMonitor"

     `Example: http://<server address>/HOPEXMonitor`
   - "/HOPEXLicenceManager"

     `Example: http://<server address>/HOPEXLicenceManager`

   A **Windows Security** window appears.
3. Enter your Windows login and password.
4. Click **OK**.
   The corresponding **HOPEX Administration console** homepage appears.
   See:
   - "HOPEX Supervision Console Description", page 9.
   - "HOPEX Monitoring Console Description", page 42
   - "HOPEX Licensing Console", page 55

     – *To switch to another **HOPEX Administration Console**, see "Switching between HOPEX Administration Consoles", page 8*

Switching between HOPEX Administration Consoles

When connected to one of the **HOPEX Administration Consoles**, you can switch to another **HOPEX Administration Console**.

To switch between **HOPEX Administration Consoles**:

1. In the **HOPEX Supervision/Monitoring/Licensing** console menu bar, click **Go to home page**  .
   The homepage of the console appears.
2. Click another **HOPEX Administration Console**.

HOPEX SUPERVISION CONSOLE

Accessing HOPEX Supervision Console

To access **HOPEX Supervision Console**, you must be granted the authorization to access:

- C:\inetpub\wwwroot\HOPEXSupervision folder.

To access **HOPEX Supervision Console**:

1. Open your web browser.
2. In the address field, after the server address, enter:
   - "/HOPEXSupervision"

   `Example: http://<server address>/HOPEXSupervision`

   A **Windows Security** window appears.
3. Enter your Windows login and password.
4. Click **OK**.
   The **HOPEX Supervision Console** homepage appears.

   - *See "HOPEX Supervision Console Description", page 9.*
   - *To switch to another **HOPEX Administration Console**, see "Switching between HOPEX Administration Consoles", page 8*

HOPEX Supervision Console Description

The **HOPEX Supervision Console** summarizes **HOPEX** supervision information in dedicated views: **Computers**, **Processes**, **Users**, **Sessions**, **Events**, and **Macro**.

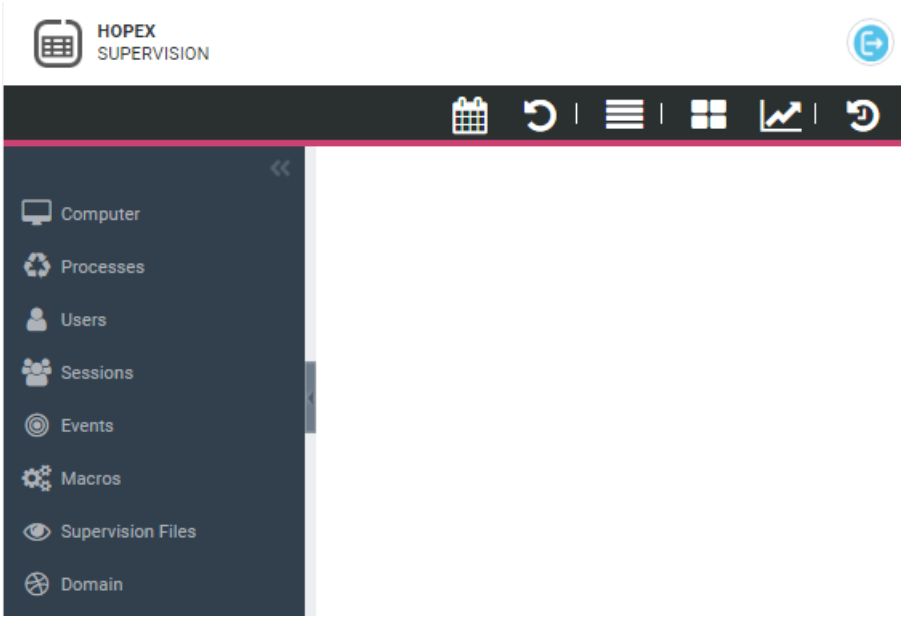
- *See "HOPEX Supervision navigation panes", page 10.*

The **HOPEX Supervision Console** toolbar enables to define the data displayed in the views.

- *See "HOPEX Supervision Console tool bar", page 12.*

The **HOPEX Supervision Console** monitors the activity within **HOPEX** and enables to understand and investigate about any anomaly at any time (e.g.: macro abnormal long execution time).

- *See "Using HOPEX Supervision data", page 29.*



HOPEX Supervision navigation panes

Events are grouped by calculated views that give access to the corresponding list of prefiltered events.

The **HOPEX Supervision Console** navigation panes give access to each dedicated view:

- **Computer**: the list of servers used by **HOPEX**
  - *See "Accessing all the Server Events", page 12.*
- **Processes**: the list of instantiated processes
  - *See "Accessing all the Server Events", page 12.*
- **Users**: the list of users who connected to **HOPEX**
  - *See "Accesssing the User Events", page 15.*
- **Sessions**: the list of **HOPEX** current or past sessions
  - *See "Accessing the Session Events", page 17.*
- **Events**: the main important events that occurred in **HOPEX**
  - *See "Accessing the Event Information", page 20.*
  - *To access all of the events, see "HOPEX Supervision Console tool bar", page 12.*
- **Macros**: the list of all the macros for which execution time is high
  - *See "Accessing the Macro Information", page 21.*
- **Supervision Files**: the supervision file location (sspsprvsYYYYmmDD.txt format)
  - *See "Finding the Supervision File Location", page 23.*
- **Domain**: a comparison of the Execution and Supervision domains
  - *See "Supervising the domain", page 24.*

HOPEX Supervision Console tool bar

The **HOPEX Supervision Console** tool bar includes:

- **Open supervision data file**  to define the supervision data files to be displayed in the supervision views.
  - *See "Defining the Supervision Data Files to be displayed", page 25.*
- **Refresh**  to refresh the information in real time (current supervision data file only).
  - *See "Refreshing the Information", page 26.*
- **All Events**  to access all of the events (not filtered) that occurred in **HOPEX**.
  - *See "Accessing all of the Events (not filtered)", page 26.*
  - *To access the main events only, see "HOPEX Supervision navigation panes", page 10.*
- **Consolidated snapshots**  to access a computed view of consolidated snapshots (list).
  - *See "Accessing a Consolidated Snapshot Event", page 27.*
- **Consolidated snapshots Graph**  to display the consolidated snapshot graph.
  - *See "Displaying consolidated snapshot graph", page 27.*
- **History**  to access your last displayed windows.
  - *See "Using the history", page 28.*

Accessing all the Server Events

The **Computer** navigation pane gives access to all the supervised events of all the servers used by **HOPEX**.

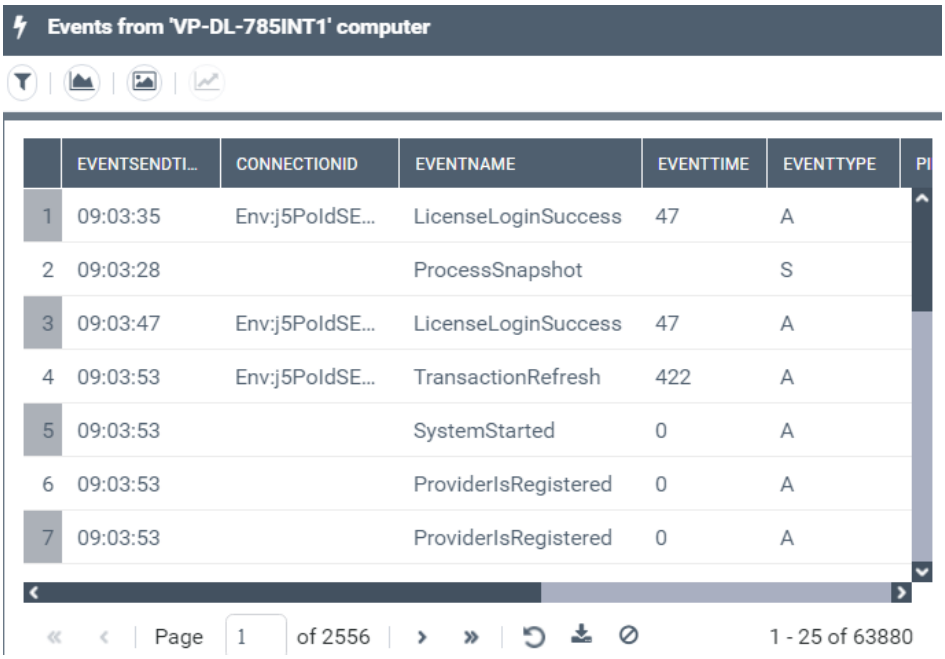
To access the server events:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** navigation panes, click **Computer.**

3. The **Computers** view is displayed.

It shows for each server its number of events.

4. Click the server row whose events you want to access.



– *See "Using HOPEX Supervision data", page 29.*

Accessing the Process Events

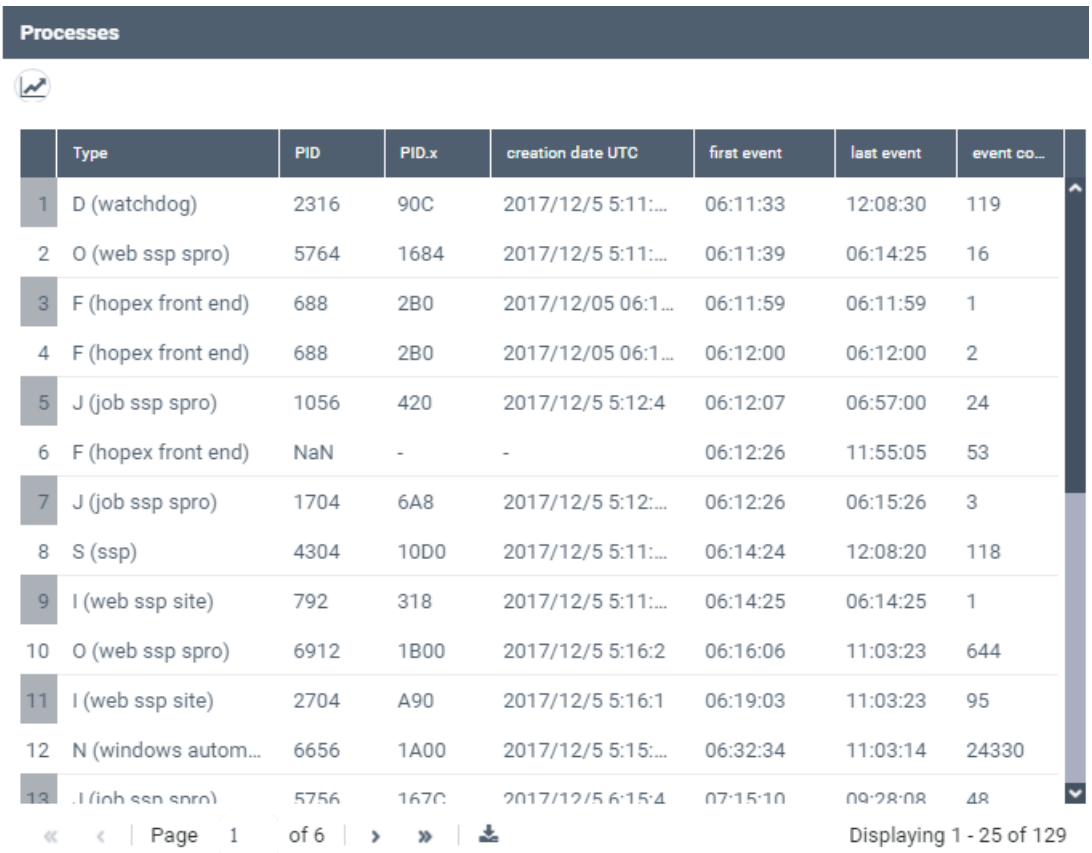
The **Processes** navigation pane gives access to all the supervised processes.

For each process it details:

- its **Type**
  - *See "Event types", page 204.*
- its identifier **PID** (decimal format), **PIDx** (hexadecimal format)
- its **Creation date** (time in UTC format)
- its **first event** time and **last event** time (both in local time)
  - *By default a snapshot is taken every 3 minutes, so that the first and last event time interval represents the process life time (more or less 3 minutes)*
- its number of events (**event count**)

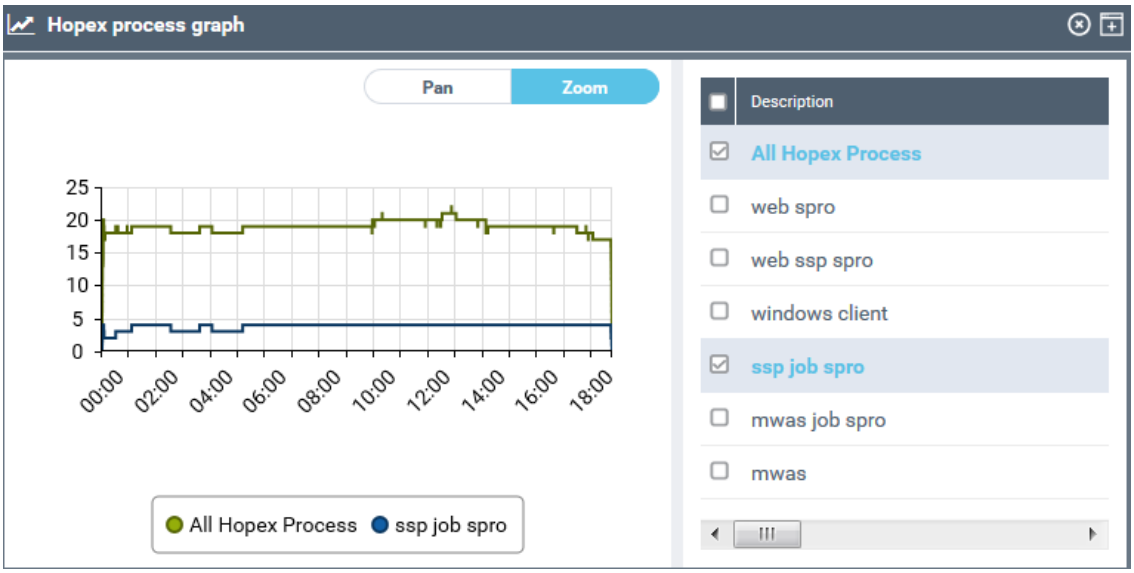
To access the process events:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** navigation panes, click **Processes**. The **Processes** view is displayed.



3. Click:
   - the process row you are interested in to get detailed information regarding all its events.
     - *See "Using HOPEX Supervision data", page 29.*
   - **Graph the number of supervised Hopex process** to display the graph of the supervised process number as a function of time.

     This graph enables to see the number and type of concurrent processes.



In the **Description** pane, select the descriptors you want to be displayed in the graph, so as to see how many processes, and of which types, are loaded simultaneously.

Accesssing the User Events

The **Users** navigation pane gives access to the list of all the users who connected to **HOPEX**.

For each user it details:

- his **Name**, in the following format:

  `<session connection mode>:<person name>`

  with <session `connection mode`>: "mono" (users do not share processes) or "multi" (users share the same processes)
- the first (**from**) and last (**to**) supervision event time regarding the user session.
  If the user connected several times, the first supervion event time corresponds to the first event of the first session and the last supervision event time to the last event of the last session.
  If several logs have been loaded, the date is also indicated.
- the process ID of the last supervision event (**Last PID**), decimal format.

  M *This information is useful to easily find out a connected user session process so as to be able to dump, kill or debug.*

  – *For more detailed inforamtion see "Accessing the Session Events", page 17*
- the number of events (**event count**)

To access the user events:

1. Access the **HOPEX Supervision console.**

   – *See "Accessing HOPEX Supervision Console", page 9.*

2. In the **HOPEX Supervision console** navigation panes, click **Users**.
The **Users** view is displayed.

3. Click the user row you are interested in to get detailed information regarding all its events.

   – *See "Using HOPEX Supervision data", page 29.*

Accessing the Session Events

The **Sessions** navigation pane gives access to the list of all the sessions of all the users.

For each session it details:

- the session identifier (**id**)
- the session **user** name, with the following format:
  - for persons:

    ```
    <session connection mode>:<person name>
    ```

    ```
    with <session connection mode>: "mono" (users do not share processes) or "multi" (users share the same processesi)
    ```
  - for the spro of the ssp:

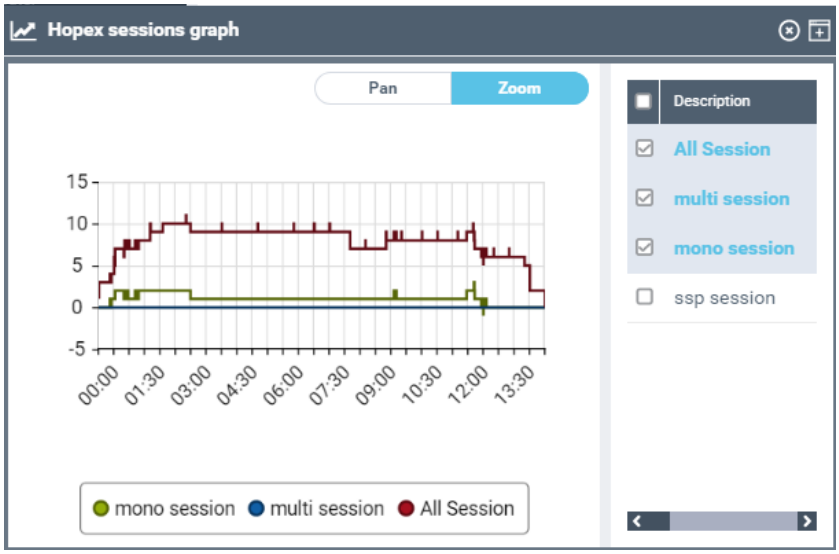
    ```
    env:<environment identifier>
    ```
- the process identifier **PID** (decimal format), **PIDx** (hexadecimal format)
- the first (**from**) and last (**to**) supervision event time regarding the user session.
  If several logs have been loaded, the date is also indicated.
- the number of events (**event count**)

To access the session events:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** navigation panes, click **Sessions**.
   The **Sessions** view is displayed.

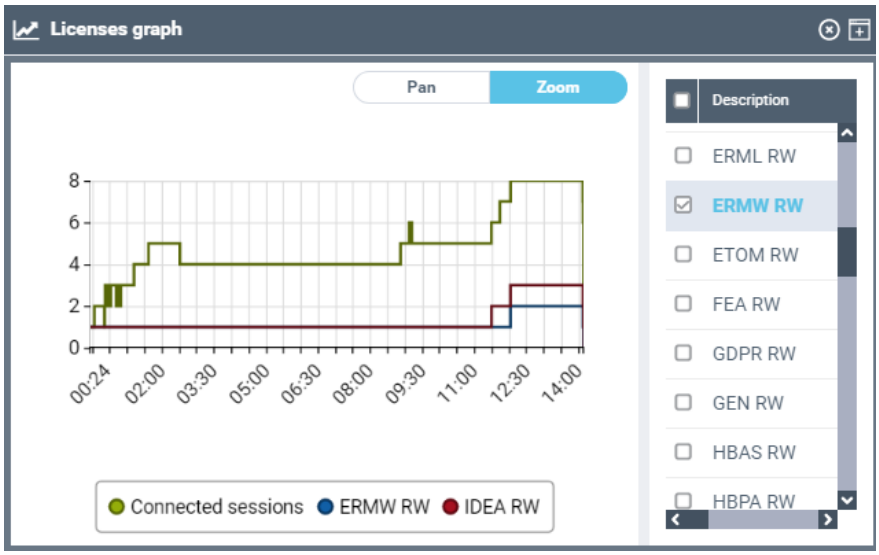
3. Click:
   - the session row you are interested in to get detailed information regarding all its events.
     - *See "Using HOPEX Supervision data", page 29.*
   - **graph the number of sessions** to get the graph of the session number as a function of time. In the **Description** pane, you can select the type of sessions (mono or multi sessions, or those open from the ssp) you want to display.

     This graph enables to see the number of concurrent sessions.



   - **graph licence token** to get the graph of the tokens used by product license, for all the sessions, as a function of time. In the

Description pane, you can select the product licenses you want to display.



- **graph profiles** to get the graph of connected pofiles as a fonction of time. In the **Description** pane, you can select the profiles you want to display.

Accessing the Event Information

The **Events** table details for each event:

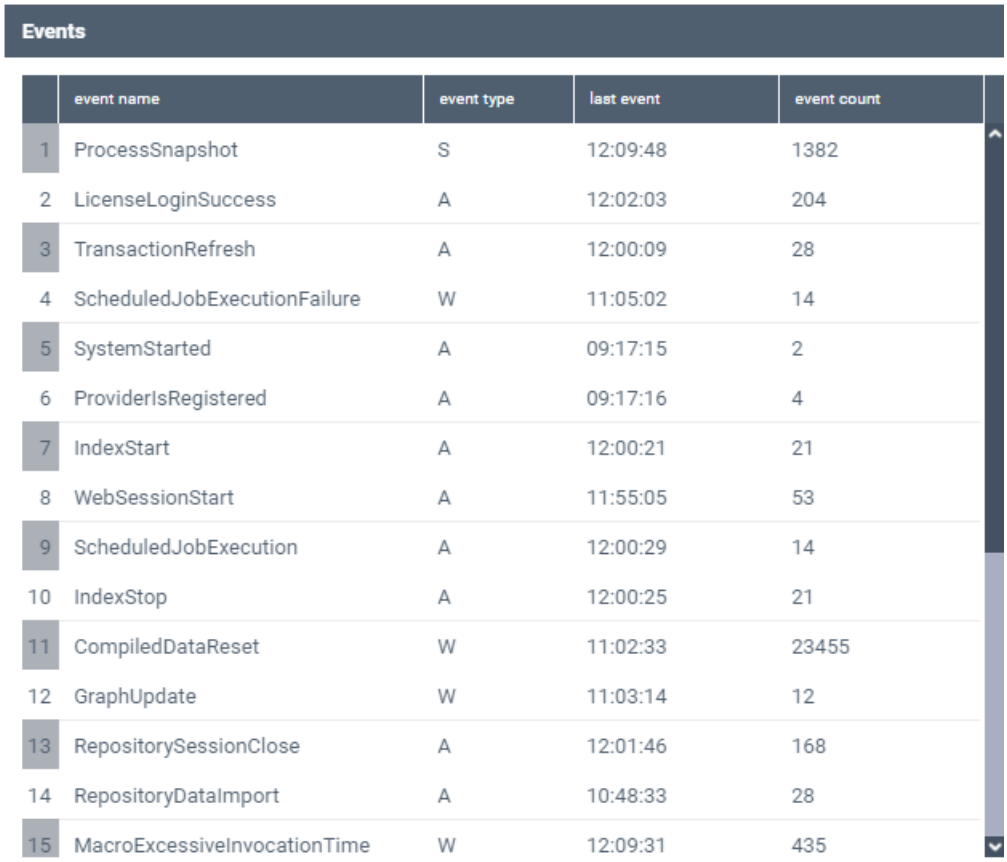
- **event name**: the name of the event
- **event type**: the type of the event
  - *See "Event types", page 204.*
- **last event**: the last time the event occurred
- **event count**: the number of time the event occurred

```
Example: The LicenseLoginSuccess action-type event has been
launched 204 times and the last time at 12:02:08.
```

To access the event information:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*

2. In the **HOPEX Supervision console** navigation panes, click **Events**.
The **Events** view is displayed and lists all the events called.



Events

	event name	event type	last event	event count
1	ProcessSnapshot	S	12:09:48	1382
2	LicenseLoginSuccess	A	12:02:03	204
3	TransactionRefresh	A	12:00:09	28
4	ScheduledJobExecutionFailure	W	11:05:02	14
5	SystemStarted	A	09:17:15	2
6	ProviderIsRegistered	A	09:17:16	4
7	IndexStart	A	12:00:21	21
8	WebSessionStart	A	11:55:05	53
9	ScheduledJobExecution	A	12:00:29	14
10	IndexStop	A	12:00:25	21
11	CompiledDataReset	W	11:02:33	23455
12	GraphUpdate	W	11:03:14	12
13	RepositorySessionClose	A	12:01:46	168
14	RepositoryDataImport	A	10:48:33	28
15	MacroExcessiveInvocationTime	W	12:09:31	435

3. Click an event row, to get more information on each event of this type.

- *See "Using HOPEX Supervision data", page 29.*

Accessing the Macro Information

The **Macros** table gives access to information regarding the macros that took a long time to be executed in **HOPEX**. In that case a supervision event is sent. For each macro, it details:

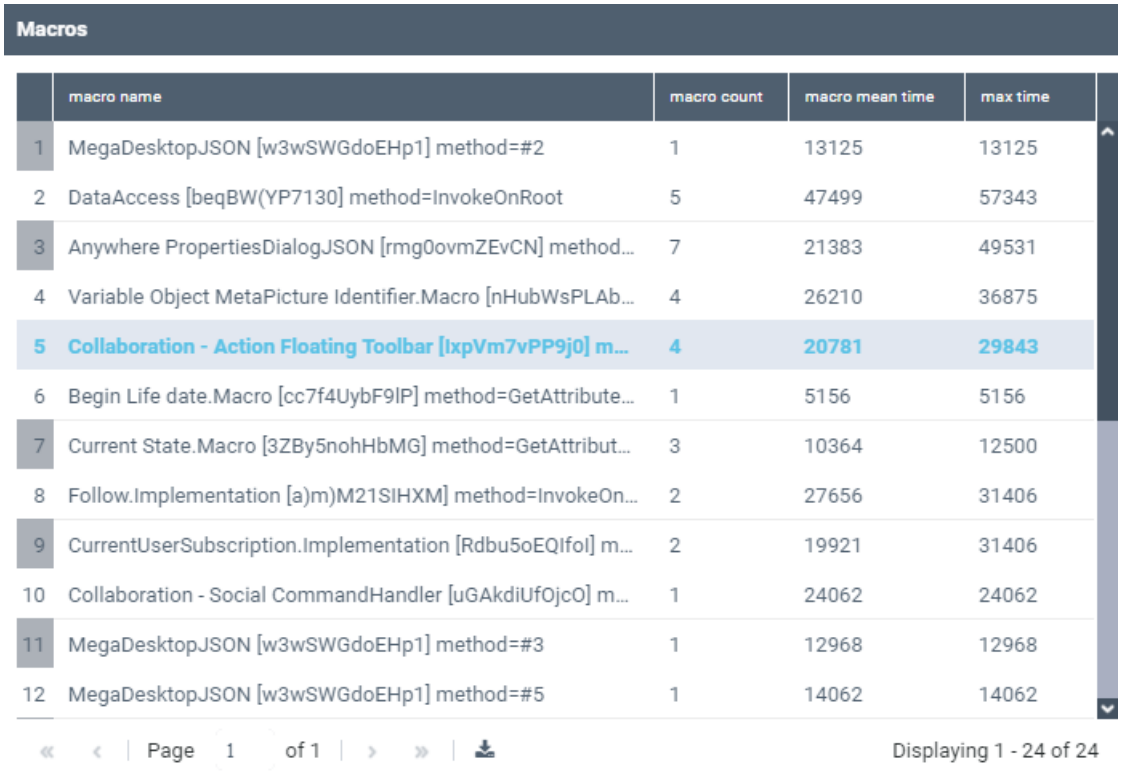
- **macro name**: the name of the macro and its use context
- **macro count**: the number of times the macro has been called
- **macro mean time**: the macro execution mean time (in ms)
- **max time**: the macro execution maximum time (in ms)

To access information regarding the macros called in **HOPEX**:

1. Access the **HOPEX Supervision console.**

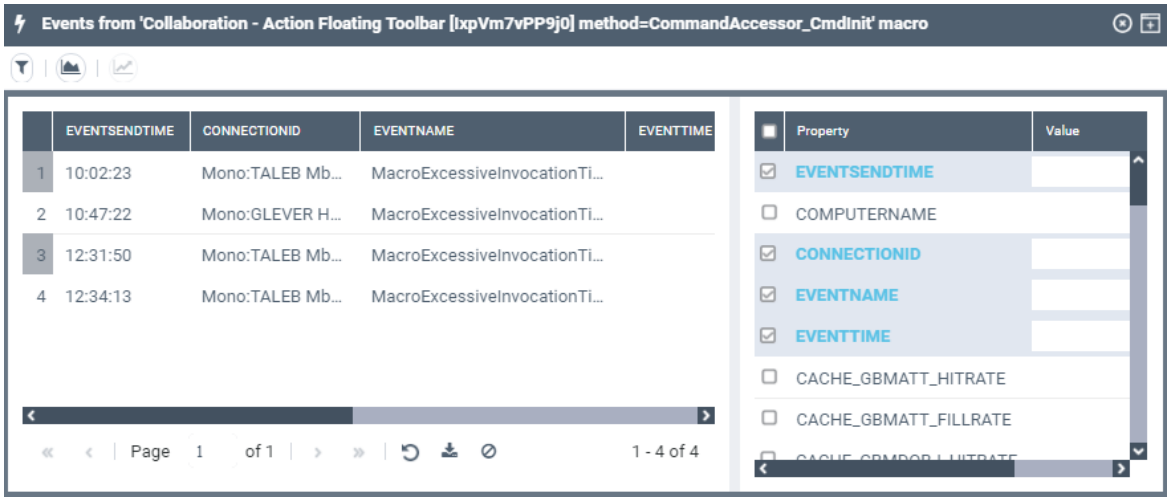
- *See "Accessing HOPEX Supervision Console", page 9.*

2. In the **HOPEX Supervision console** navigation panes, click **Macros**.
The **Macros** view is displayed and lists all the macros with a long execution time for the defined period (by default, data is reset every two weeks).



3. Click a macro row to get more information regarding this macro executions.

   Example: in the above example click the fifth row to get detailed information regarding the macro "Collaboration - Action Floating toolbar".

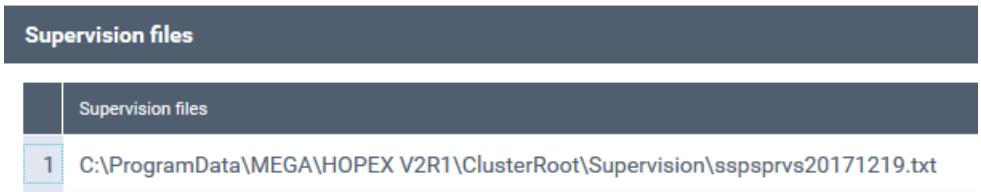


- *See "Using HOPEX Supervision data", page 29.*

Finding the Supervision File Location

To find the supervision file location:

1. Access the **HOPEX Supervision console**.
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** navigation panes, click **Supervision Files**.
   The **Supervision files** page shows the stored supervision files and their location (sspsprvsYYYYmmDD.txt) location.



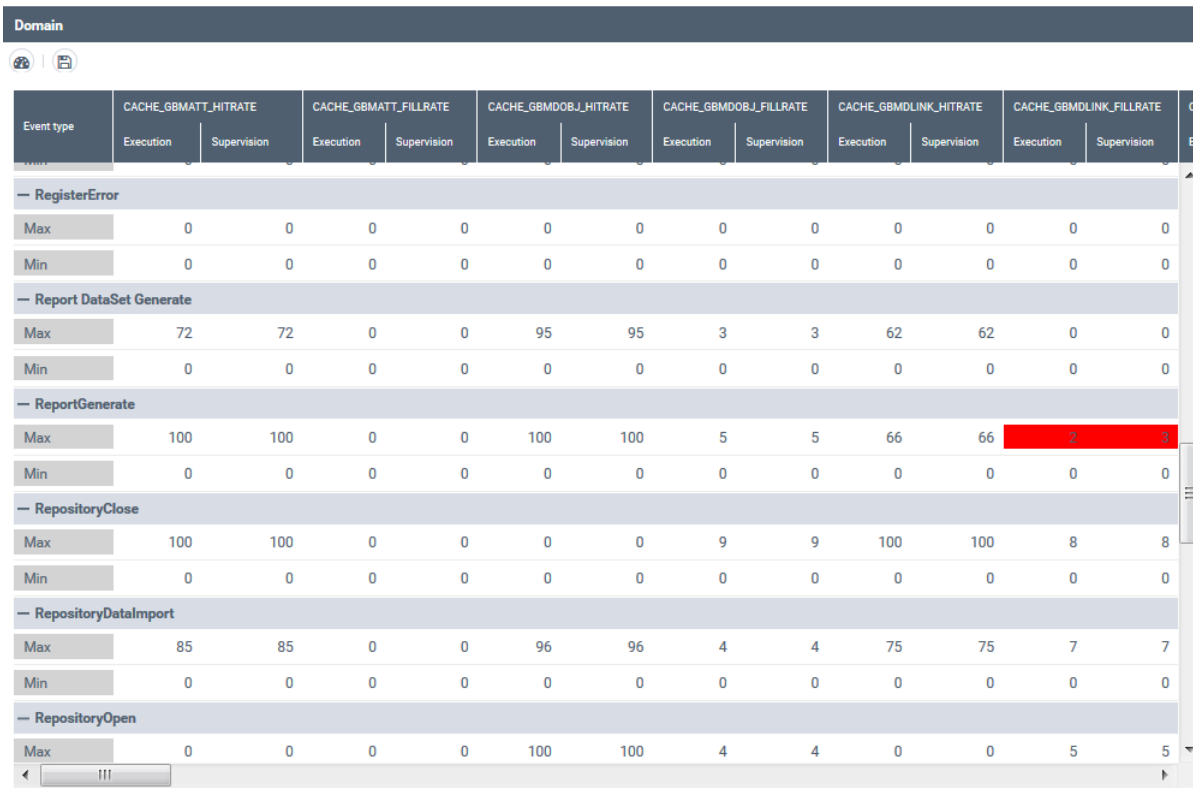
Supervising the domain

To use the **Domain** page, you must first create a Supervision domain, which corresponds to a typical log file of **HOPEX** in good operating condition. The **Domain** page compares the **Execution** domain with this **Supervision** domain stored as a reference.

To supervise the domain:

1. Create a supervision domain: in **HOPEX Server Supervisor**, select **Hopex Supervision > Supervision configuration > update supervision domain**.
   - *See "Supervision tool: HOPEX Server Supervisor", page 208.*
2. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
3. In the **HOPEX Supervision console** navigation panes, click **Domain**.

4. Click **Load domain** .
   The **Domain** page information is uploaded. For each event and each indicator, it compares the Execution domain with the Supervision domain.
   Each indicator unit depends on the indicator concerned (e.g.: %, B, MB)
   Red cells indicate that the supervised indicator (**Execution** domain) is out of the reference range (**Supervision** domain).



Event type	CACHE_GBMATT_HITRATE		CACHE_GBMATT_FILLRATE		CACHE_GBMDOBJ_HITRATE		CACHE_GBMDOBJ_FILLRATE		CACHE_GBMDLINK_HITRATE		CACHE_GBMDLINK_FILLRATE	
	Execution	Supervision	Execution	Supervision	Execution	Supervision	Execution	Supervision	Execution	Supervision	Execution	Supervision
— RegisterError												
Max	0	0	0	0	0	0	0	0	0	0	0	0
Min	0	0	0	0	0	0	0	0	0	0	0	0
— Report DataSet Generate												
Max	72	72	0	0	95	95	3	3	62	62	0	0
Min	0	0	0	0	0	0	0	0	0	0	0	0
— ReportGenerate												
Max	100	100	0	0	100	100	5	5	66	66	2	3
Min	0	0	0	0	0	0	0	0	0	0	0	0
— RepositoryClose												
Max	100	100	0	0	0	0	9	9	100	100	8	8
Min	0	0	0	0	0	0	0	0	0	0	0	0
— RepositoryDataImport												
Max	85	85	0	0	96	96	4	4	75	75	7	7
Min	0	0	0	0	0	0	0	0	0	0	0	0
— RepositoryOpen												
Max	0	0	0	0	100	100	4	4	0	0	5	5

Defining the Supervision Data Files to be displayed

To define the supervision data files to be displayed in the supervision console pages:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** tool bar, click **Open supervision data file** .

3. In the calendar select the dates you are interested in.
   By default, only the last 15 supervision data files are saved. You cannot display older data.
   - *If you want to select non consecutive dates, you are prompted to confirm your selection.*
   - *To return to the current day, click **Today**.*

   The supervision files corresponding to your selection are displayed in the **Supervision files** pane.
4. In the **Supervision files** pane, click **Ok**.
   The supervision data files corresponding to the selected dates are loaded.
   All the pages are updated with the defined dates.

Refreshing the Information

You can refresh the information displayed in all of the supervision pages.

- *For information on the supervision pages, see "HOPEX Supervision navigation panes", page 10.*

Refreshing the information is only useful if the displayed supervision data includes the log of the day.

- *See "Defining the Supervision Data Files to be displayed", page 25.*

To refresh the information:

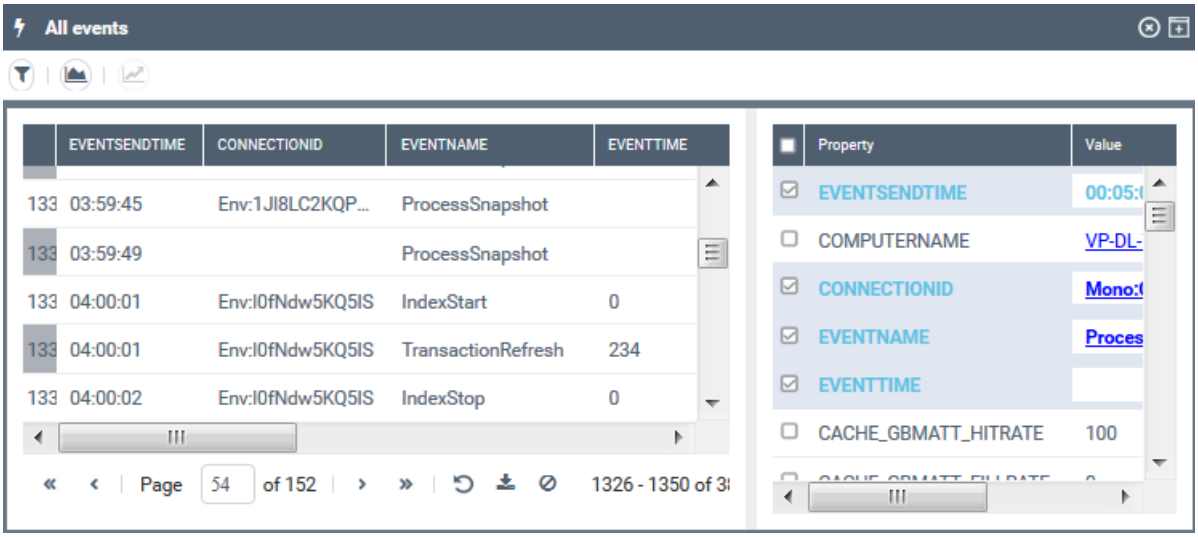
1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** tool bar, click **Refresh**  .
   The supervision data is updated.

Accessing all of the Events (not filtered)

To access all of the events that occured in **HOPEX**:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*

2. In the **HOPEX Supervision console** tool bar, click **All Events** .
The list of events not filtered is displayed.



- *See "Using HOPEX Supervision data", page 29.*

Accessing a Consolidated Snapshot Event

To access a consolidated snapshot event:

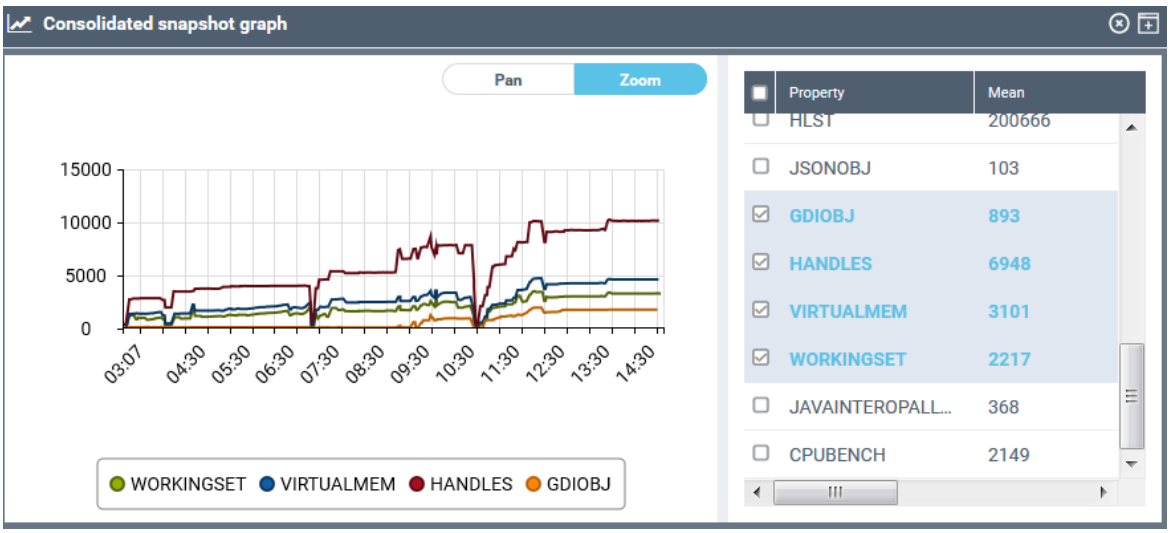
1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** tool bar, click **Consolidated snapshots** .
The Consolidated Snapshot events list is displayed.

Displaying consolidated snapshot graph

To display the graph of consolidated snapshot:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. In the **HOPEX Supervision console** tool bar, click **Consolidated snapshots Graph** .

3. In the right pane select or clear the properties you want to display or not in the graph.



Using the history

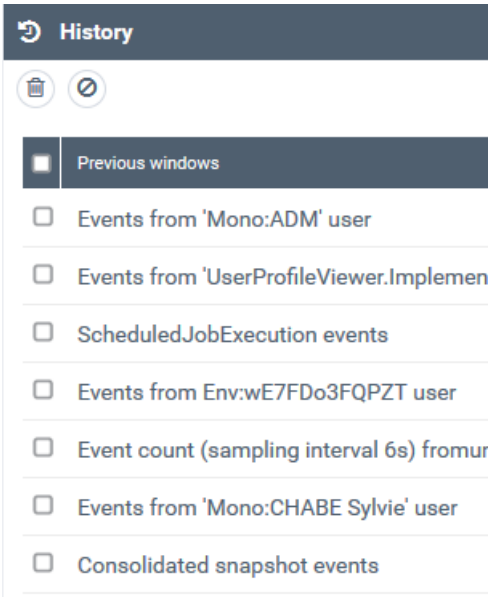
You can:

- access a page you already displayed in the **HOPEX Supervision console**.
- delete the whole content

To access a page you displayed in the **HOPEX Supervision console**:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*

2. In the **HOPEX Supervision console** tool bar, click **History** .



3. If needed:
   - click to reset the whole list.
   - select an item and click to remove the selected page from the **History** list.

Using HOPEX Supervision data

From each **HOPEX Supervision console** page you can:

- customize the data display
  - *See: "Customizing the view display", page 30,*
- access prefiltered lists
  - *See: "Accessing prefiltered lists", page 32,*

  From these prefiltered list you can:
  - add filters
    - *See: "Adding filters to the event list", page 33,*
  - create graphs
    - *See: "Graphing an event count by time interval", page 35, "Graphing an event", page 36, "Graphing the server CPU", page 38.*
    - *See also "Handling a graph", page 38.*

Customizing the view display

You can customize the data display:

- sort the data
- display only the indicators you are interested in

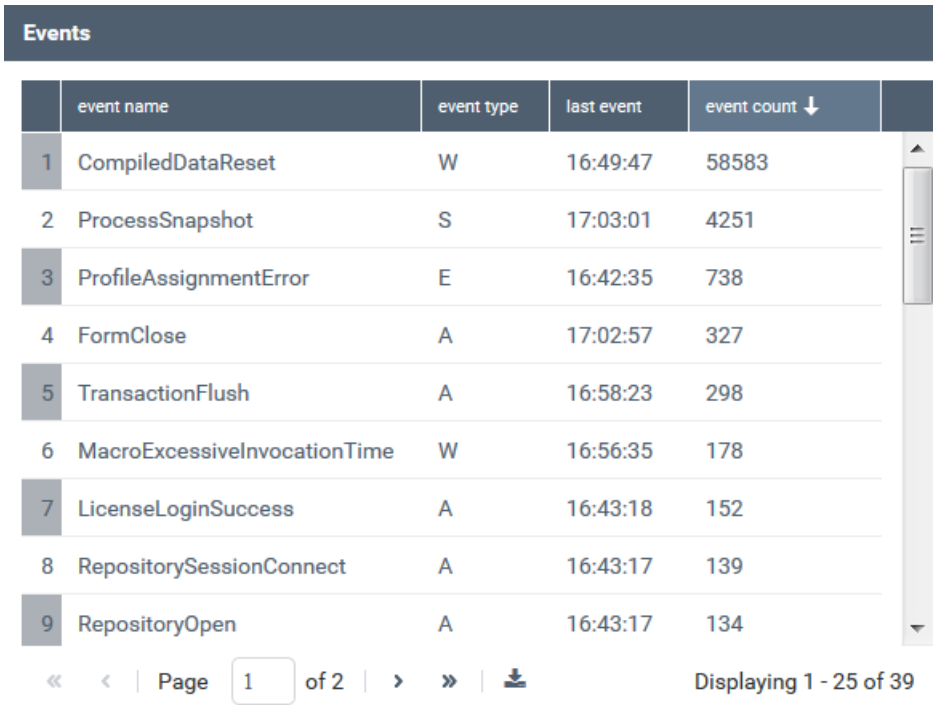
To customize the view display:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. Display the data view you want to supervise**.**
   - *See "HOPEX Supervision navigation panes", page 10.*

   Data is displayed in a table format.

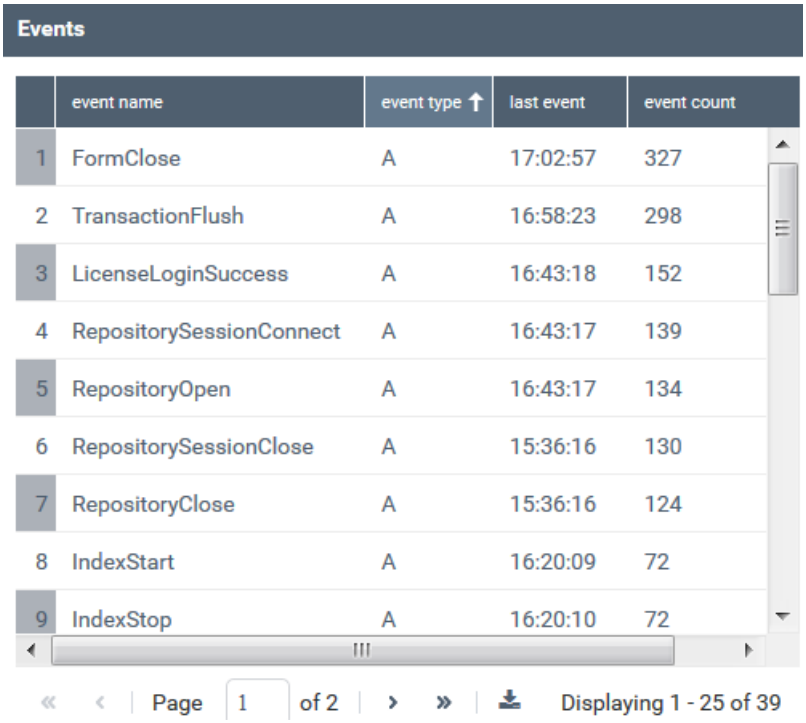
3. You can:
   - click a header arrow **> Sort Ascending** or **Sort Descending**.

     For example in the **Events** page, in the event count, select **Sort Descending** to display the most frequent events at the top.



   - click a header to sort the data according to a specific indicator.

     For example in the **Events** page, click event type to sort the events by type.



- click a header arrow **> Columns** and select the indicators you want to be displayed only.

Accessing prefiltered lists

From **Computers**, **Processes**, **Users**, **Sessions**, **Events**, **Macros** pages, you can access prefiltered lists.

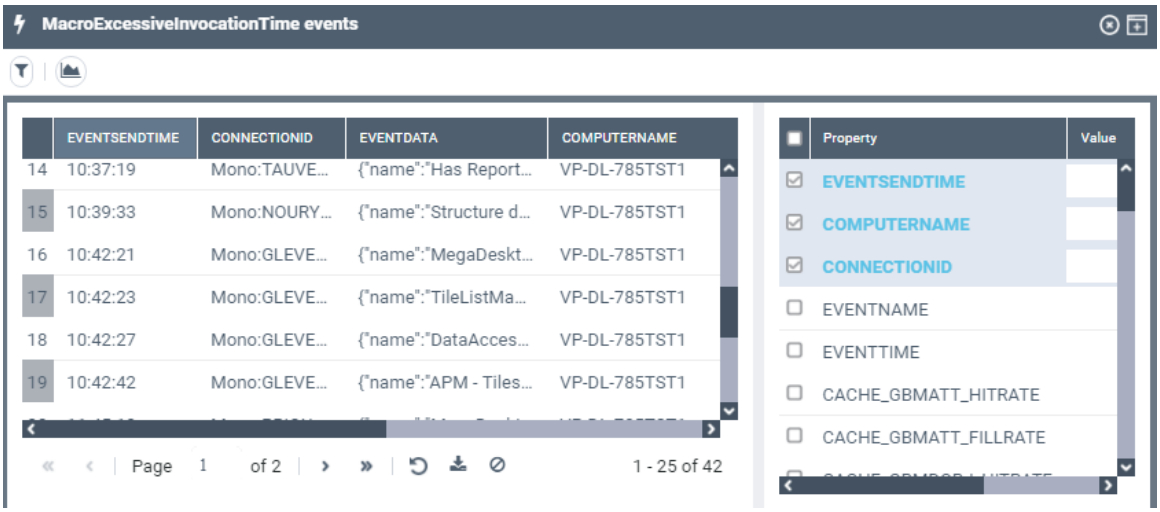
To access prefiltered lists:

1. Access the **HOPEX Supervision console.**
   - *See "Accessing HOPEX Supervision Console", page 9.*
2. Display the supervision page you are interested in**.**
   - *See "HOPEX Supervision navigation panes", page 10.*

   The supervision events are displayed in a table.
3. Click the row of the event you are interested in.
   All the events corresponding to the selected row are listed chronologically in a table.

   For example in the **Events** page, click the "MacroExcessiveInvocationTime" event. All of the

"MacroExcessiveInvocationTime" events that occured are displayed.



4. (If needed) In the right pane select the indicators you want to display as columns in the supervision event list.
5. (If needed) You can:
   - filter the event display
     - *See "Adding filters to the event list", page 33.*
   - graph the event count by time interval
     - *See "Graphing an event count by time interval", page 35.*
   - graph an event
     - *See "Graphing an event", page 36.*
   - (**Computers** view specific) graph the CPU
     - *See "Graphing the server CPU", page 38.*

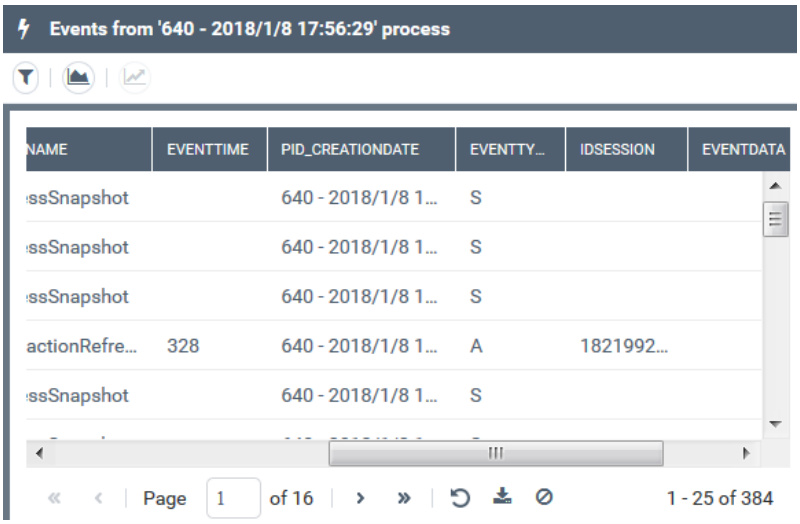
Adding filters to the event list

For a better focus on the information you can add filters to the event list. You can add filters on any of the indicators.

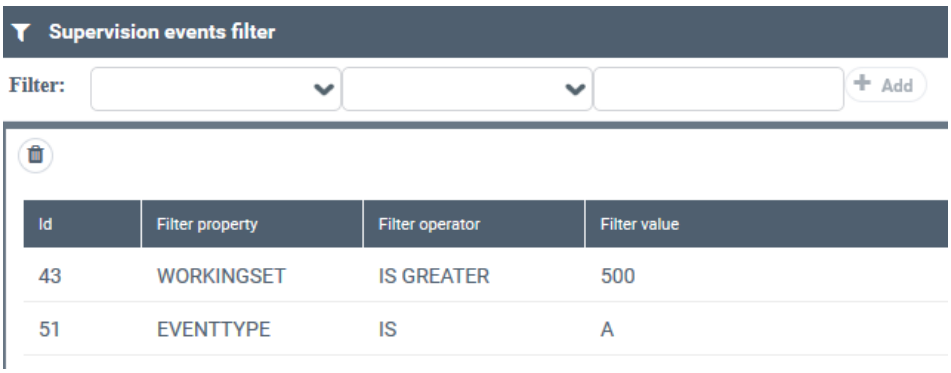
- *To delete all the filters added, see "Deleting all the filters added to the event list", page 35.*

To add filters to the event list:

1. Access the prefiltered list.
   - *See "Accessing prefiltered lists", page 32.*



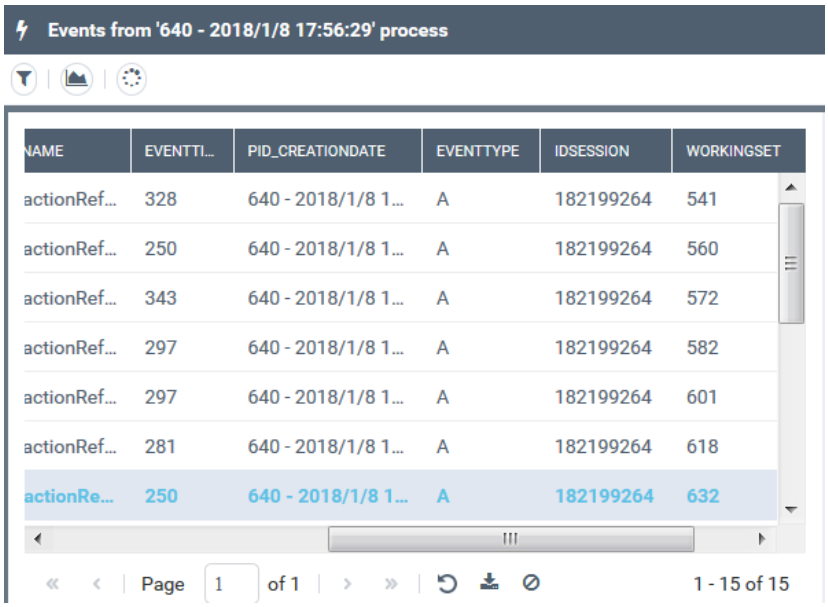
2. Click **Filters** .
   The **Supervision events filter** window is displayed.
3. Configure your filter:
   - in the first drop down list, select the indicators (**Filter property**)
   - in the second drop down list, select the operator (**Filter operator**)
   - in the field enter the value
4. Click **Add**.
   The filter is added in the filter list with a specific **Id**.
   - *You can add as many filters as you want.*



You cannot modify a filter, you can delete it only.

- *To delete a filter: select the filter row and click **Delete selected filter** .*

5. Close the window.
Events are filtered according to the added filters.



- *To delete all the added filters, see "Deleting all the filters added to the event list", page 35.*

6. (if needed) Click to export the filtered view in a .txt format.

Deleting all the filters added to the event list

To delete all the filters added to the event list:

1. Access a prefiltered event list.
   - *See "Accessing prefiltered lists", page 32.*
2. At the bottom of the list, click **Clear all filters** .

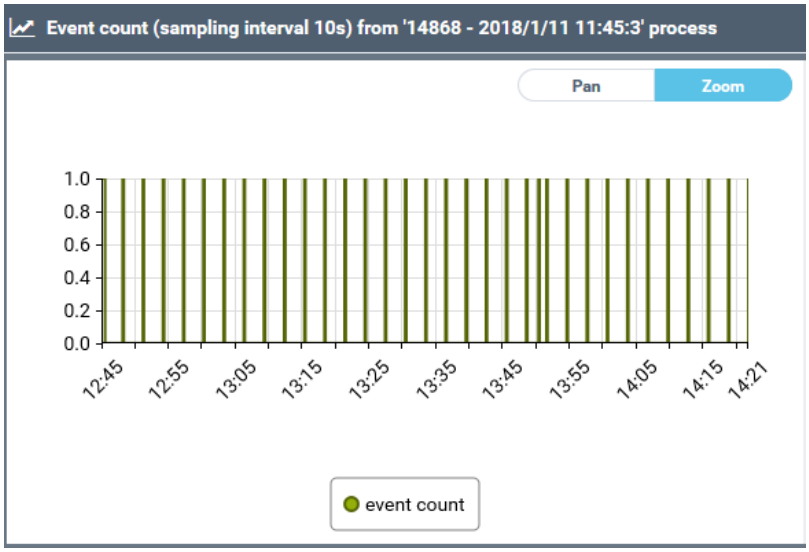
Graphing an event count by time interval

For another display of the information you can graph the event count that occurred by a defined time interval (seconds).

To graph the event count by time interval:

1. Access a prefiltered event list.
   - *See "Accessing prefiltered lists", page 32.*
2. Click **Graph events count by time** .
   The **Enter the sampling interval value** dialog box is displayed.
3. Enter a value (in seconds) to define the sampling interval value.

4. Click **Ok**.
   The event number that occurred by the defined time interval is graphed (step line format).



- *See "Handling a graph", page 38.*

Graphing an event

For another display of the information you can graph an event.

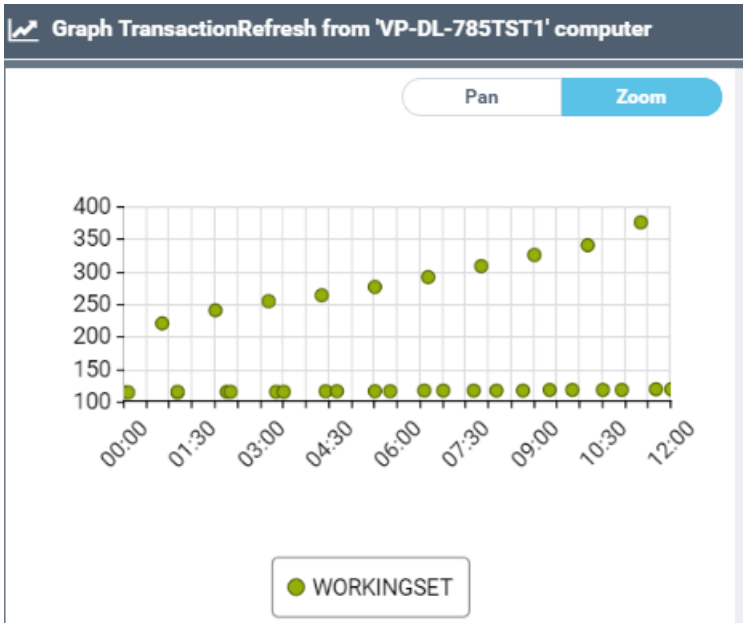
- *You can graph a warning-type (W) or an error-type (E) event.*

To graph an event:

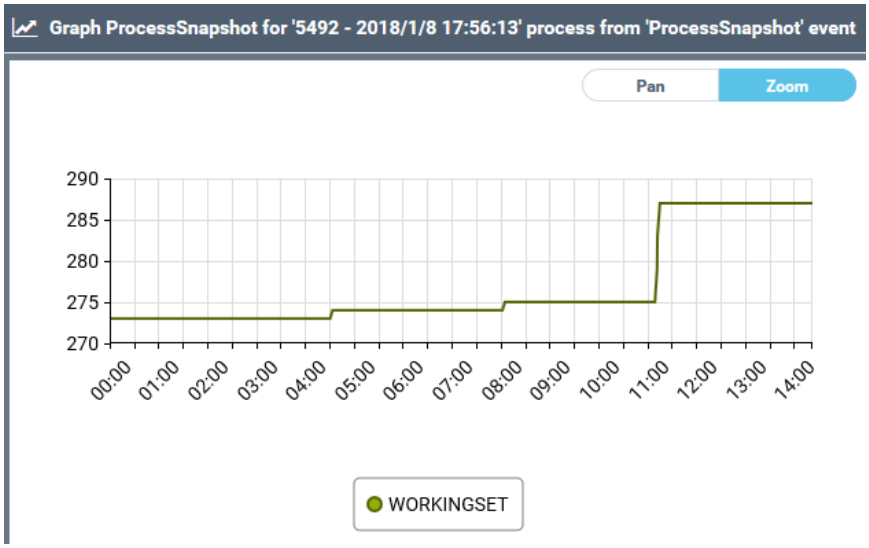
1. Access a prefiltered event list.
   - *See "Accessing prefiltered lists", page 32.*
2. Select the row corresponding to the event (Eventname) you want to graph.

3. Click **Show graph for <eventname> events** .

Example: graph for an A-type event (TransactionRefresh)



Example: graph for an S-type event (ProcessSnapshot)

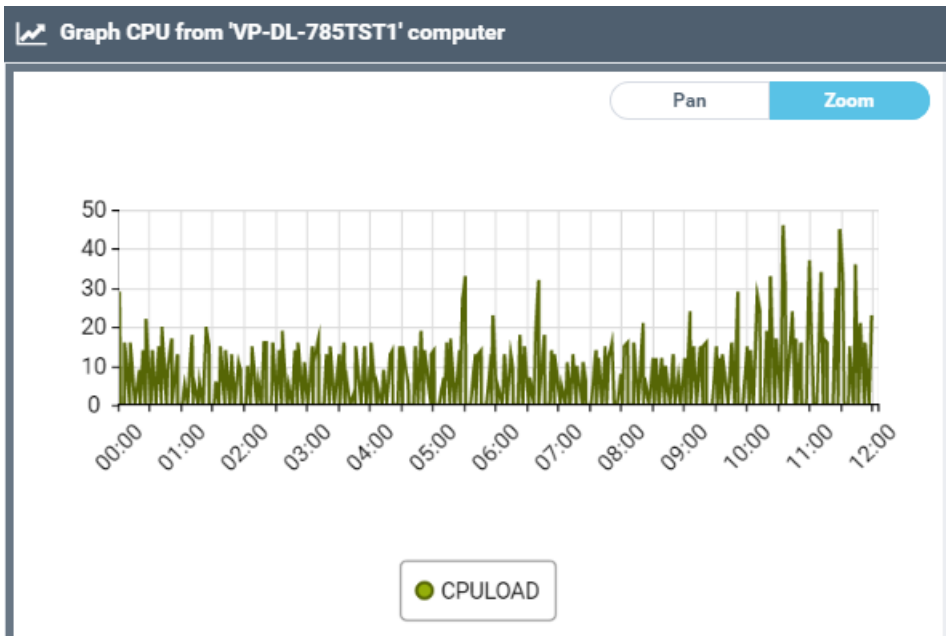


- *See "Handling a graph", page 38.*

Graphing the server CPU

To graph the server CPU:

1. From the **Computers** page, access a prefiltered event list.
   - *See "Accessing prefiltered lists", page 32.*
2. Click **Graph CPU from <server name> computer** .



- *See "Handling a graph", page 38.*

Handling a graph

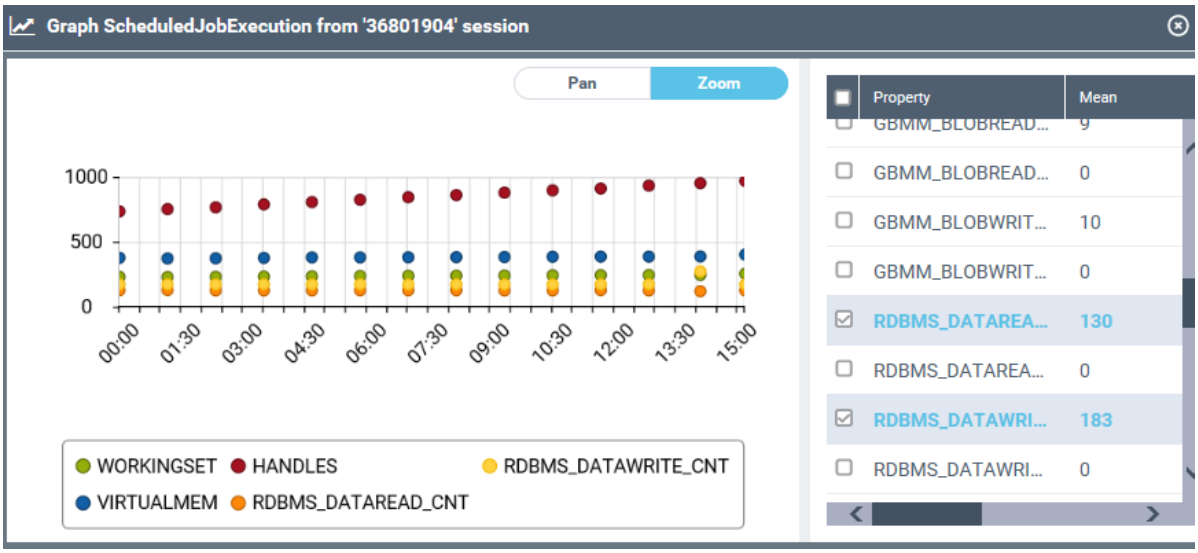
You can:

- add other indicators to the graph
- zoom in the graph
- move within the graph

To add indicators to the graph:

1. From a prefiltered list, create a graph.
   - *See "Accessing prefiltered lists", page 32,*

2. In the graph window, in the right pane, select the indicators you want to add to the graph.
   The selected indicators are:
   - added to the graph with a specific color for each indicator, and the graph scale is automatically fitted.
   - highlighted in bold blue in the right pane.



To zoom in the graph:

1. Create a graph.
   - *See "Accessing prefiltered lists", page 32,*
2. In the graph window, click **Zoom**.
3. Click in the graph, keep your finger on the mouse and drag it to the right to zoom in.

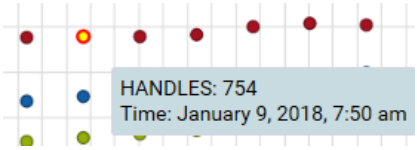
To move within the graph

1. Create a graph.
   - *See "Accessing prefiltered lists", page 32,*
2. In the graph window, click **Pan**.
3. Click in the graph, keep your finger on the mouse and drag it to the right or left. to move to the right or left

To view an indicator information:

1. Create a graph.
   - *See "Accessing prefiltered lists", page 32,*

2. Click the data in the graph.
   The indicator value is detailed and when it occurred.



HOPEX MONITORING CONSOLE

Introduction to HOPEX Monitoring Console

The **HOPEX Monitoring Console** enables Administrators to perform basic administration tasks.

These tasks include:

- web user connection management
- process monitoring (restricted to MEGA internal use)
- download of logs and **HOPEX** information
- checks regarding MEGA Web Access Server (MWAS) and Site Service Provider (SSP) operational status
- performance diagnostic test

The **HOPEX Monitoring Console** also includes Administration information regarding technical parameter values.

Accessing HOPEX Monitoring Console

To access **HOPEX Monitoring Console**, you must be granted the authorization to access:

- C:\inetpub\wwwroot\HOPEXMonitor folder.

To access **HOPEX Monitoring Console**:

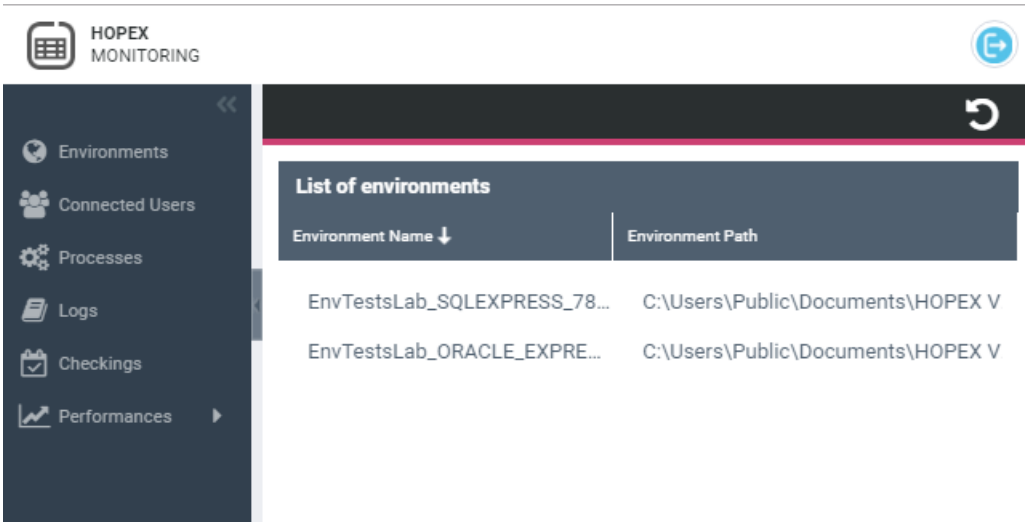
1. Open your web browser.
2. In the address field, after the server address, enter:
   - "/HOPEXMonitor"

     `Example: http://<server address>/HOPEXMonitor`

   A **Windows Security** window appears.
3. Enter your Windows login and password.
4. Click **OK**.
   The **HOPEX Monitoring Console** homepage appears.

   – *To switch to another **HOPEX Administration Console**, see* *"Switching between HOPEX Administration Consoles", page 8*

HOPEX Monitoring Console Description



The **HOPEX Monitoring Console** includes the following navigation panes:

- **Environments**
  Lists the environments accessible from HOPEX.
- **Connected Users**
  Displays information regarding the web users logged in to **HOPEX** and enables to perform actions regarding these connections.
- **Processes**
  Displays, for each server, information regarding the current processes running in **HOPEX**. This page requires high administration rights (restricted to MEGA internal use).
- **Logs**
  Enables to download log and information files regarding the operating system.
- **Checkings**
  Displays MWAS and SSP installation checks.
- **Performances**
  Enables to test performances for a specific repository (**RDBMS Test**) or a specific profile (**MEGA Test**).

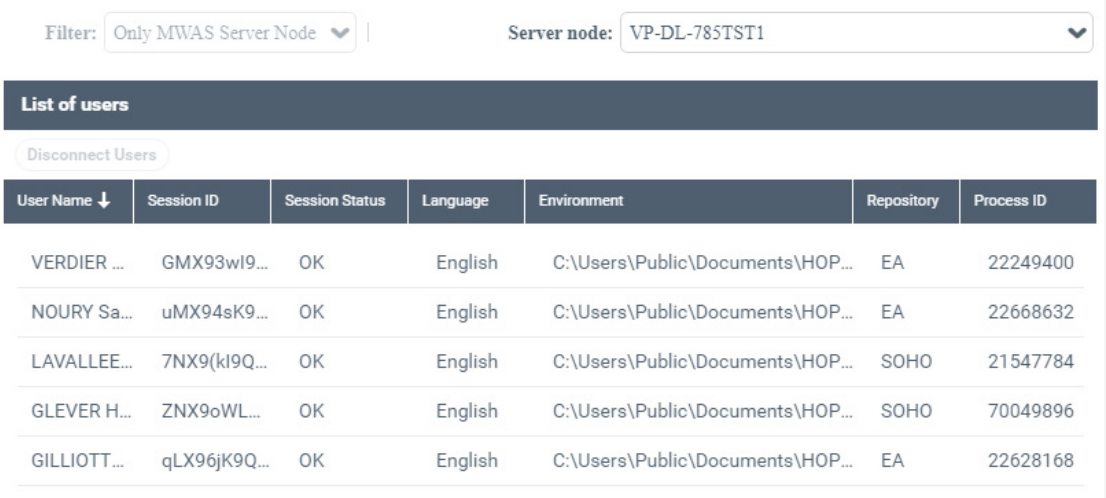
Environments

The **Environments** navigation pane details the access path of each environment that can be accessed from **HOPEX**:

Connected Users

To log in to **HOPEX**, a web user must be authenticated as a **HOPEX** user.

Upon login, a session is opened in MEGA Web Access. This session is assigned an ID by MEGA Web Access. Each Web user has his own session identifier.



The **Connected Users** navigation pane displays, by **Server node** selected, for each Web user logged in to **HOPEX**:

- his HOPEX **User name**.
- his **Session Id**, which enables to identify the Web user activity via the MWAS log.
- his **Session Status**, which can take the following values:
  - ***TimeOutISPending***: indicates that user activity has ceased and that the session will soon expire. You can disconnect the session if you want.
  - ***OK***: indicates that the Web user session is running.
  - ***TimeOut***: indicates that the session has expired.
- the **Language** in which his HOPEX application is displayed (at connection).
- the **Environment** and **Repository** on which he is working.
- his **Process ID**.

Disconnecting users

The **Disconnect Users** button enables to delete the session of the selected users, logging them out of HOPEX (Web Front-End).

Use this option when problems occur, when requested by a technical support engineer, or when you delete the cache.

To disconnect Web users:

1. Access the **HOPEX Monitoring Console**.
   - *See "Accessing HOPEX Monitoring Console", page 41.*
2. (In a cluster configuration) In the **Server node** field, select the server concerned.
3. Click **Connected Users** navigation pane.
4. Select the user(s) you want to disconnect.
   - *Use the [Ctrl] key to select several users.*
5. Click **Disconnect Users**.
   Selected users are logged out. Their work is saved for their next log in.

Disconnecting users with TimeOut sessions

You can disconnect all the user sessions that are identified as expired (TimeOut Status).

To disconnect all user expired sessions (timed out):

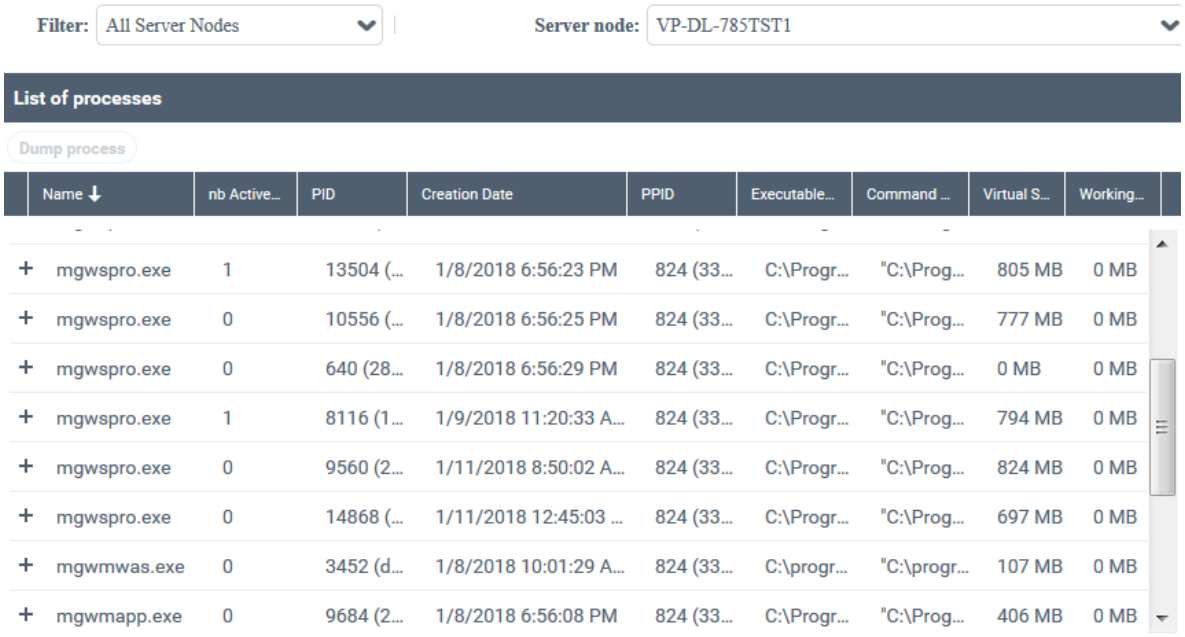
1. Access the **HOPEX Monitoring Console**.
   - *See "Accessing HOPEX Monitoring Console", page 41.*
2. (In a cluster configuration) In the **Server node** field, select the server concerned.
3. Click **Connected Users** navigation pane.
4. Sort the users by **Session Status**.
5. Select the users with "TimeOut" Session status.
6. Click **Disconnect Users**.

Processes

The **Processes** navigation pane displays, for the **Server node** selected, information regarding the current processes running in **HOPEX**. This page requires high administration rights (restricted to MEGA internal use)

It displays the following processes:

- mgwswd
- mgwssp
- mgwspro
- mgwmwas
- mgwmapp
- mgwfcgi



For each process, it details:

- the **Name** of the process
  You can expand mgwspro processes to display the names of the persons (**List of Related Active Profiles**) using the mgwspro. In monosession there is only one person.
- the number of persons using the process (**nb Active Profiles**)
- its identifier **PID** (decimal/hexadecimal formats)
- its **Creation date** (time in UTC format)
- its **PPID**
- its **Executable path**
- its **Command line**
- its **Virtual size**
- its **Working Set size**

Logs

The **Logs** navigation pane facilitates the log collection and location:

- ***uas***
  The Unified Authentication Service (UAS) is a centralized service for your login.

  `uas-YYYY-mm-DD.log`
- ***ssplog***
  The Site Service Provider (SSP) is a central component that accesses to shared information and provides internal services (e.g.: authentication, supervision, scheduler).

  `ssplogYYYYmmDD.txt`
- ***ssperr***
  The SSP error log details errors related to the environment SSP.

  `ssperrYYYYmmDD.txt`
- ***mwaslog***
  MWAS manages Web sessions when Web users login to or logout from the HOPEX (IIS) application.

  MWAS log details actions related to MWAS. MWAS includes mgwmwas and mgwmapp processes.

  `mwaslogYYYYmmDD.txt`
- ***megaerr***
  The HOPEX log details all the errors related to HOPEX kernel.

  `megaerrYYYYmmDD.txt`
- ***sspsprvs***
  The Supervision log details indicators related to the supervision.

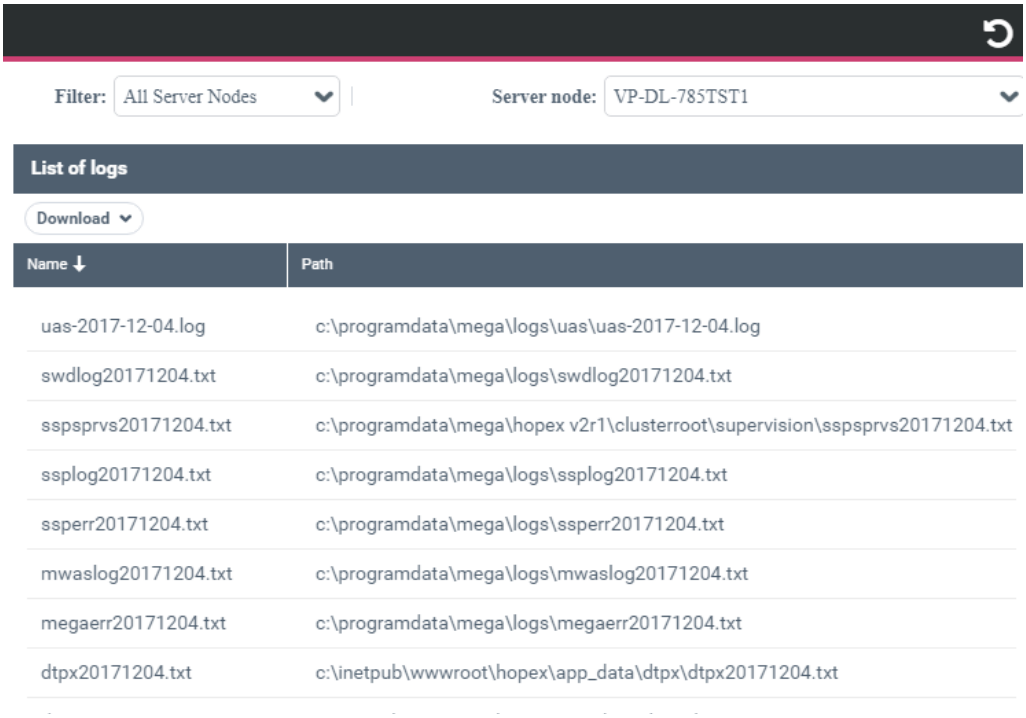
  `sspsprvsYYYYmmDD.txt`
- ***swdlog***
  The Watchdog makes sure that the SSP and MWAS components are working properly. In case one of these components is not responding to other components, the Watchdog restart it.

  The Watchdog log details actions related to the Watchdog component.

  `swdlogYYYYmmDD.txt`

You can download, by **Server node** selected, all the logs or only a specific log, in a zip file. The zip file also includes:

- megasite.ini, which is the site option configuration file.
- a zipIndex (.txt format), which details each log location.



Downloading all the logs

To download all the logs:

1. Access the **HOPEX Monitoring Console**.
   - *See "Accessing HOPEX Monitoring Console", page 41.*
2. Click **Logs** navigation pane.
3. (In a cluster configuration) In the **Server node** field, select the server concerned.
4. (if you want to restrict the result to a specific node) In the **Filter** field, select either: "Only SSP Server Nodes", "Only MWAS Server nodes", or "Only Front-End Server Nodes".
   - *By default: "All Server Nodes".*
5. In the **List of logs** section, select **Download > all**.
6. Select **Save file** and define a storage location.
7. Click **OK**.
   All the logs are saved as a zipped file.
   The zip file also includes:
   - megasite.ini
   - a zipIndex (.txt format), which details each log location:

Downloading a specific log

To download a specific log:

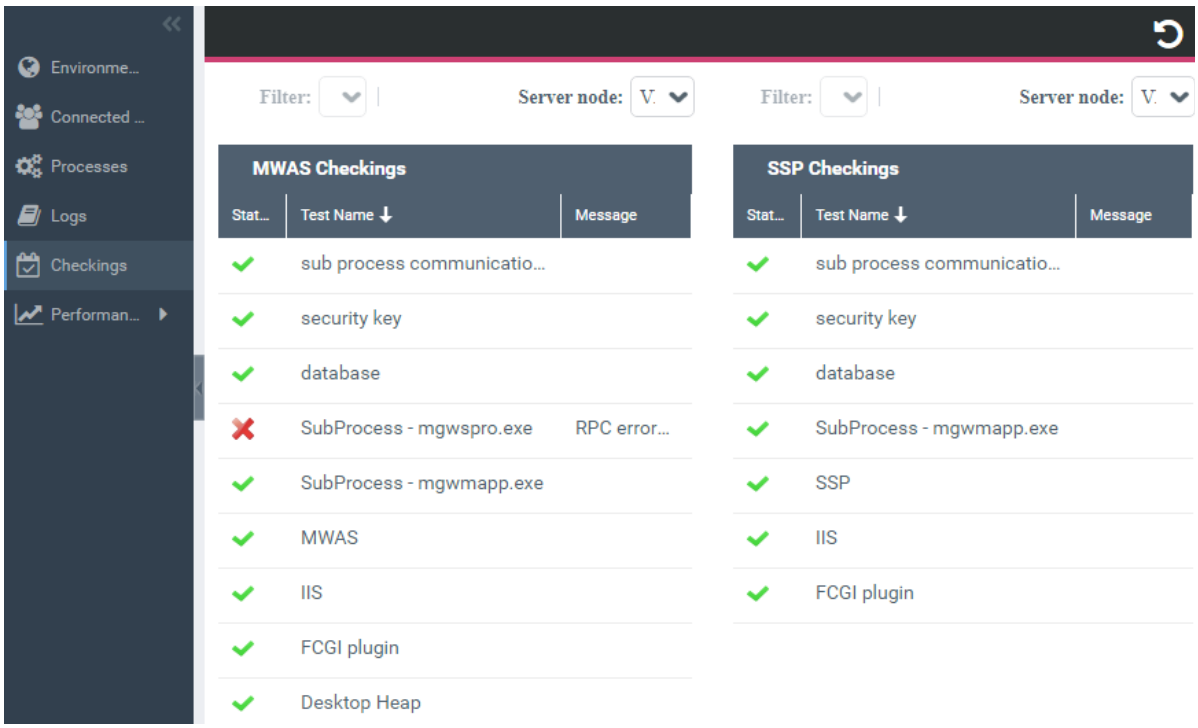
1. Access the **HOPEX Monitoring Console**.
   - *See "Accessing HOPEX Monitoring Console", page 41.*
2. Click **Logs** navigation pane.
3. (In a cluster configuration) In the **Server node** field, select the server concerned.
4. In the **List of logs** section, double-click the row corresponding to the log you want to download.
5. Select **Save File** and define a storage location.
6. Click **OK**.
   All the logs are saved as a zipped file.

Checkings

You can check, for each **Server node**, whether the MWAS and SSP are fully operational or not.

For each control test, the **Status** column indicates whether the result is:

- ✓ correct
- ✗ not correct, in that case a a **Message** indicates the invalidity reason.



Performances

You can launch performance diagnostic tests from the **Performances** navigation pane.

You can test the performances either:

- on a specific repository
  - *See "Launching a performance diagnostic test on a repository", page 49.*

  or
- on a specific profile
  - *See Launching a performance diagnostic test on a profile.*

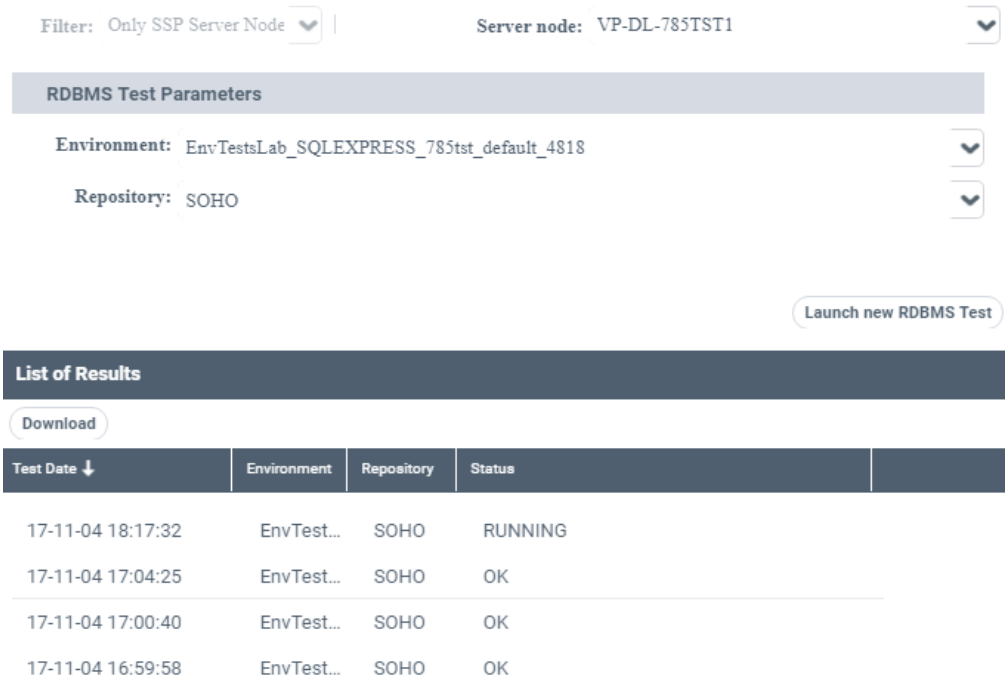
  In that case you must first define a Test Case on the specific profile.
  - *(see Defining a Test Case on a profile).*

Launching a performance diagnostic test on a repository

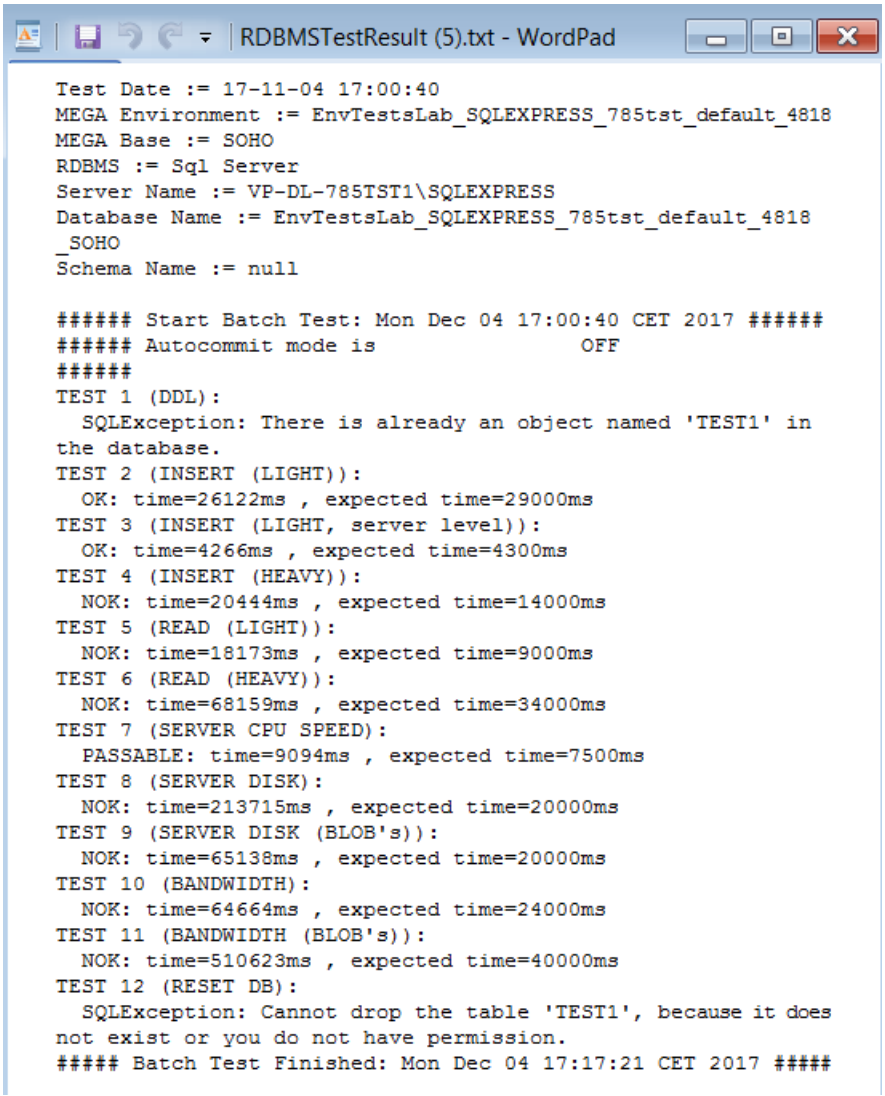
To launch a performance diagnostic test on a repository:

1. Access the **HOPEX Monitoring Console**.
   - *See "Accessing HOPEX Monitoring Console", page 41.*
2. Click **Performances** navigation pane arrow.
3. Click **RDBMS Test**.
4. (In a cluster configuration) In the **Server node** field, select the server concerned.
5. In the **Environment** field, select the environment.
6. In the **Repository** field, select the repository.
7. Click **Launch new RDBMS Test**.

8. Click **Refresh** .
   The test row appears in the result table, with the "RUNNING" status.



9. Once the test Status is "OK", double-click the test row to open or download it.



Defining a Test Case on a profile

You can define a Test Case using a macro and/or a Test Case Configuration.

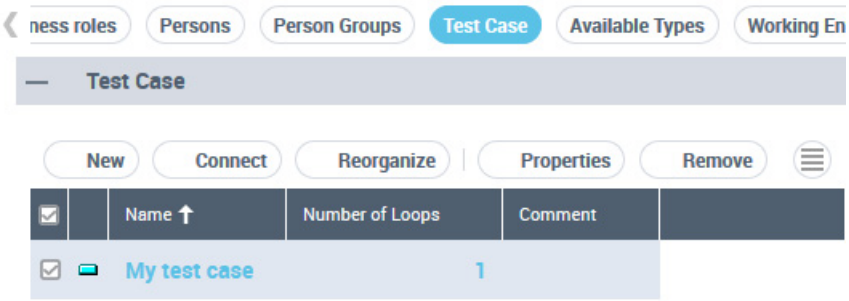
The macro describes the Test Case to be performed.

The TestCase Configuration is defined for a MetaClass, and if needed with related MetaAssociations and/or MetaAssociationEnds.

To define a Test Case on a profile:

1. Connect to HOPEX Administration (Web Front-End or Windows Front-End).

2. Access the profile properties and select **Test Case**.
3. In the **Test Case** section, click **New**.
   - *If **New** is not available, check in the Repository options that you are authorized to modify HOPEX data.*
4. Enter a **Name** for your test case and click **OK**.
5. In the **Number of Loops** cell, enter the number of times requested for the test.



6. In the T**estCase Configuration** section:
   - connect a macro (go to step 7)

   and/or
   - create a **Test Case Configuration** (go to step 8).
7. (If needed) In **Macro**, click **Connect** 🔗 and connect the macro describing the test.

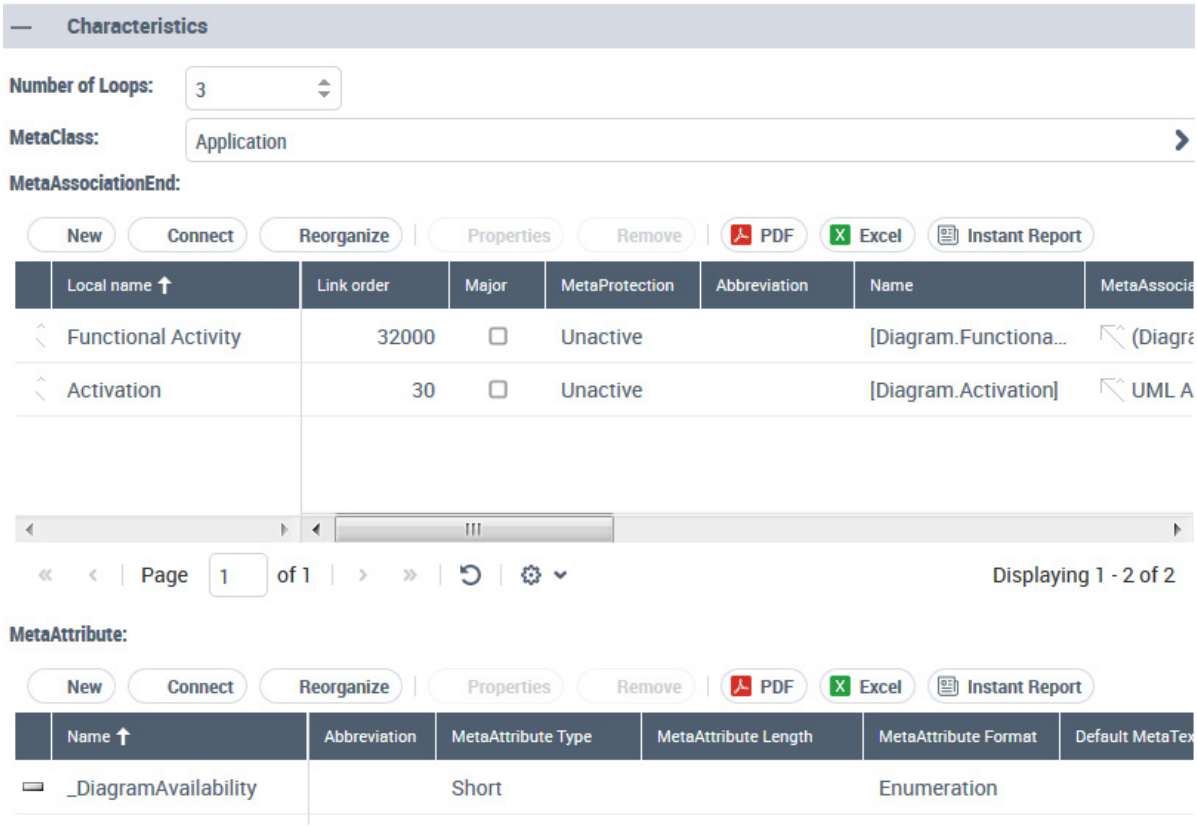
8. (If needed) In **TestCase Configuration** click **New** .
   - Enter your TestCase Configuration **Name** and click **OK**.
   - Click **Properties** to define the TestCase Configuration.
   - In the **Number of Loops** field, enter the number of time you want the testCase Configuration to be run.

     `Example: 3`
   - In the **MetaClass** field, connect the MetaClass concerned by the test case.

     `Example: Application.`
   - (If needed) In the **MetaAssociationEnd** table connect the MetaAssociationEnds concerned by the test.

     `Example: Activation and Functional Activity.`
   - (If needed) In the **MetaAttribute** table connect the MetaAttributes concerned by the test.

     `Example: _DiagramAvailability.`



9. Click **Apply**.

Launching a performance diagnostic test on a profile

Prerequisite: a Test Case is define on the profile concerned, see "Defining a Test Case on a profile", page 51.

To launch a performance diagnostic test on a profile:

1. Access the **HOPEX Monitoring Console**.
   - *See "Accessing HOPEX Monitoring Console", page 41.*
2. Click **Performances** navigation pane arrow.
3. Click **MEGA Test**.
4. (In a cluster configuration) In the **Server node** field, select the server concerned.
5. In the **Environment** field, select the environment.
6. In the **Repository** field, select the repository.
7. In the **Profile** drop-down list, select the profile on which you want to perform the diagnostic test.
8. Click **Launch new MEGA Test**.

HOPEX LICENSING CONSOLE

The **HOPEX Licensing Console** enables to supervise and manage the licenses.

Accessing HOPEX Licensing Console

To access **HOPEX Licensing Console**, you must be granted the authorization to access:

- C:\inetpub\wwwroot\HOPEXLicenceManager folder.

To access **HOPEX Licensing** console:

1. Open your web browser.
2. In the address field, after the server address, enter:
   - "/HOPEXLicenceManager"

     `Example: http://<server address>/HOPEXLicenceManager`

   A **Windows Security** window appears.
3. Enter your Windows login and password.
4. Click **OK**.
   The **HOPEX Licensing** console homepage appears.

   – *To switch to another **HOPEX Administration Console**, see "Switching between HOPEX Administration Consoles", page 8.*

Accessing a License Characteristics

The **General View** displays:

- the list of licenses
- the list of products available on each license.

For each license, it details:

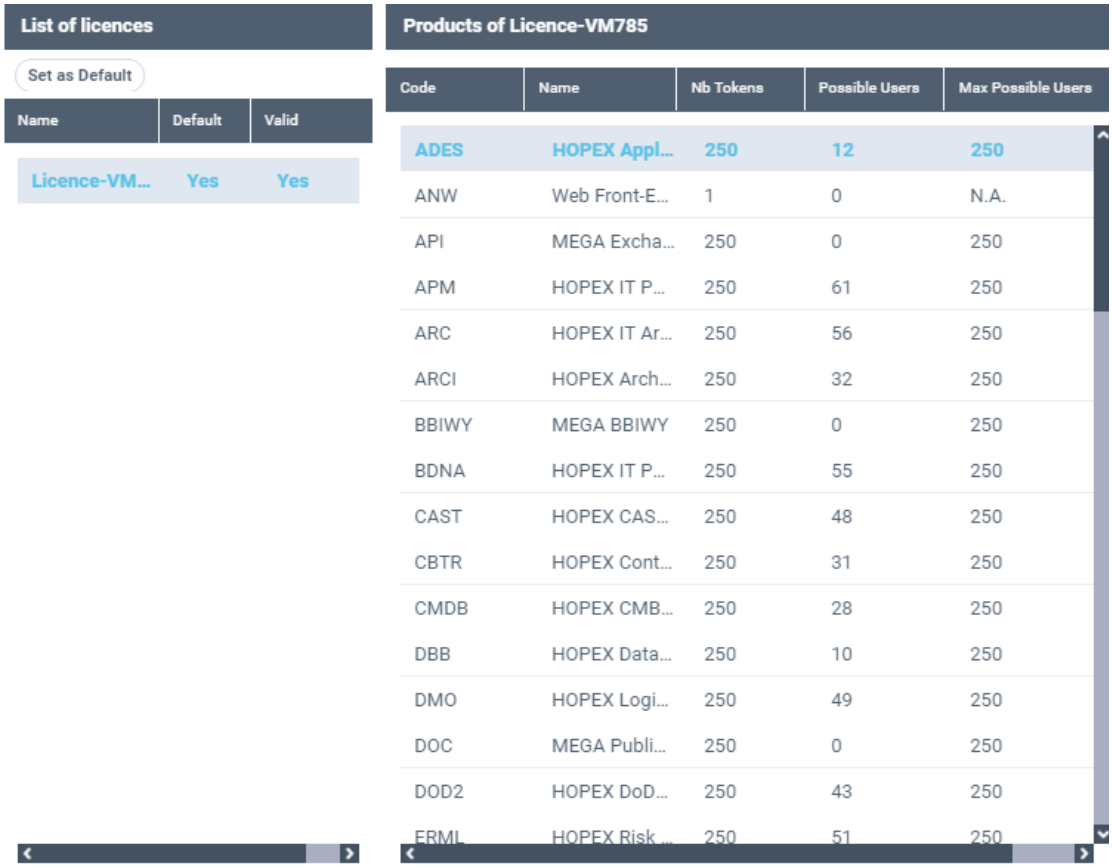
- the product **Code**
- the product **Name**
- the number of tokens, which represents the maximum number of possible concurrent users (**Nb Tokens**)
- the users who are assigned the license (**Possible users**)
- the maximum number of users who can be assigned the license (**Max Possible users**)

To access the license general view:

1. Access the **HOPEX Licensing** console.

   – *See "Accessing HOPEX Licensing Console", page 55.*

2. In the **HOPEX Licensing** navigation panes, click **General View**.
The **List of licenses** and **Products of Licence-<license name>** page is displayed.



Using the HOPEX Licensing Console

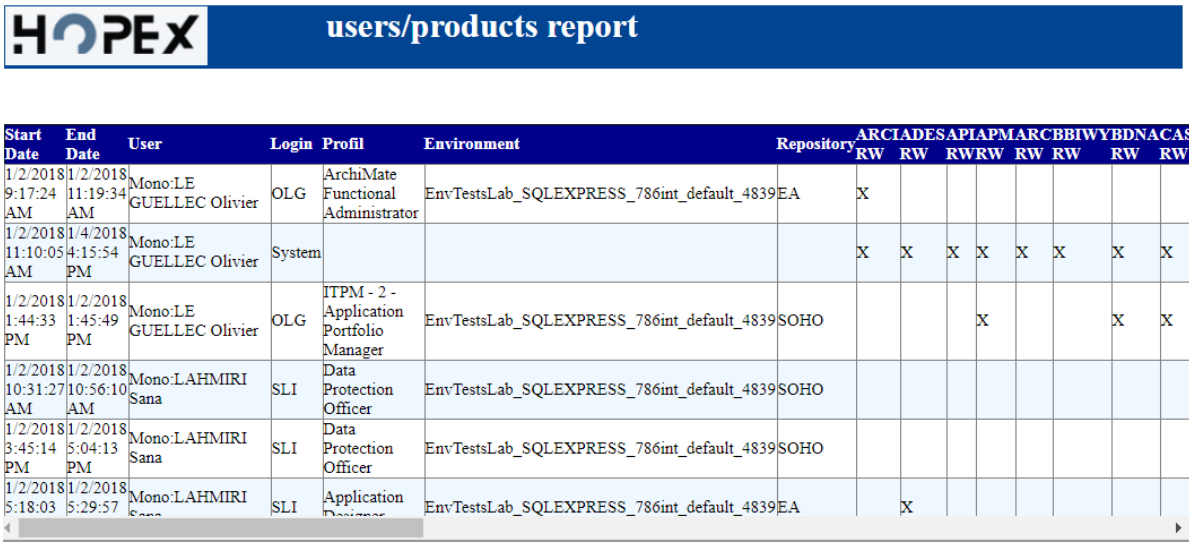
Downloading the report regarding the licenses used

The "users/products report" details the licenses used during all the sessions. It details for each session:

- its **Start Date** and **End Date** (mm/dd/yyyy hh:mm:ss format)
- the corresponding **User** name
- the corresponding user's **Login** name
- the **Profile, Environment**, and **Repository** used for connection
- each license used

To download your license report:

1. Access the **HOPEX Licensing** Console.
   - *See "Accessing HOPEX Licensing Console", page 55.*
2. In the right pane, click **Download Licenses Report**  .
   The **users/products report** is downloaded in html format.



HOPEX users/products report

Start Date	End Date	User	Login	Profil	Environment	Repository	ARC RW	IADES RW	APIAPM RW	ARCB RW	BBIWY RW	BDNA RW	CAS RW
1/2/2018 9:17:24 AM	1/2/2018 11:19:34 AM	Mono:LE GUELLEC Olivier	OLG	ArchiMate Functional Administrator	EnvTestsLab_SQLEXPRESS_786int_default_4839	EA	X						
1/2/2018 11:10:05 AM	1/4/2018 4:15:54 PM	Mono:LE GUELLEC Olivier	System				X	X	X	X	X	X	X	X
1/2/2018 1:44:33 PM	1/2/2018 1:45:49 PM	Mono:LE GUELLEC Olivier	OLG	ITPM - 2 - Application Portfolio Manager	EnvTestsLab_SQLEXPRESS_786int_default_4839	SOHO				X			X	X
1/2/2018 10:31:27 AM	1/2/2018 10:56:10 AM	Mono:LAHMIRI Sana	SLI	Data Protection Officer	EnvTestsLab_SQLEXPRESS_786int_default_4839	SOHO								
1/2/2018 3:45:14 PM	1/2/2018 5:04:13 PM	Mono:LAHMIRI Sana	SLI	Data Protection Officer	EnvTestsLab_SQLEXPRESS_786int_default_4839	SOHO								
1/2/2018 5:18:03	1/2/2018 5:29:57	Mono:LAHMIRI	SLI	Application	EnvTestsLab_SQLEXPRESS_786int_default_4839	EA		X						

Defining a default license

When several licenses are available, you can define which of the licenses is the default one.

To define a default license:

1. Access the **HOPEX Licensing** console.
   - *See "Accessing HOPEX Licensing Console", page 55.*
2. In the **HOPEX Licensing** navigation panes, select **General View > by licenses**.
3. In the **List of licenses** pane, select the license row.
4. Click **Set as Default**.
   In the **List of licenses** pane, the license **Default** value is "Yes".

Viewing which HOPEX license a user is assigned

To view which **HOPEX** license a user is assigned:

1. Access the **HOPEX Licensing** console.
   - *See "Accessing HOPEX Licensing Console", page 55.*
2. In the **HOPEX Licensing** navigation panes, select **General View** > **by users**.
3. In the **List of Users** pane, select the login of the user you are interested in.
   The user's license is indicated in the **License of <login>** pane.

Managing the user products

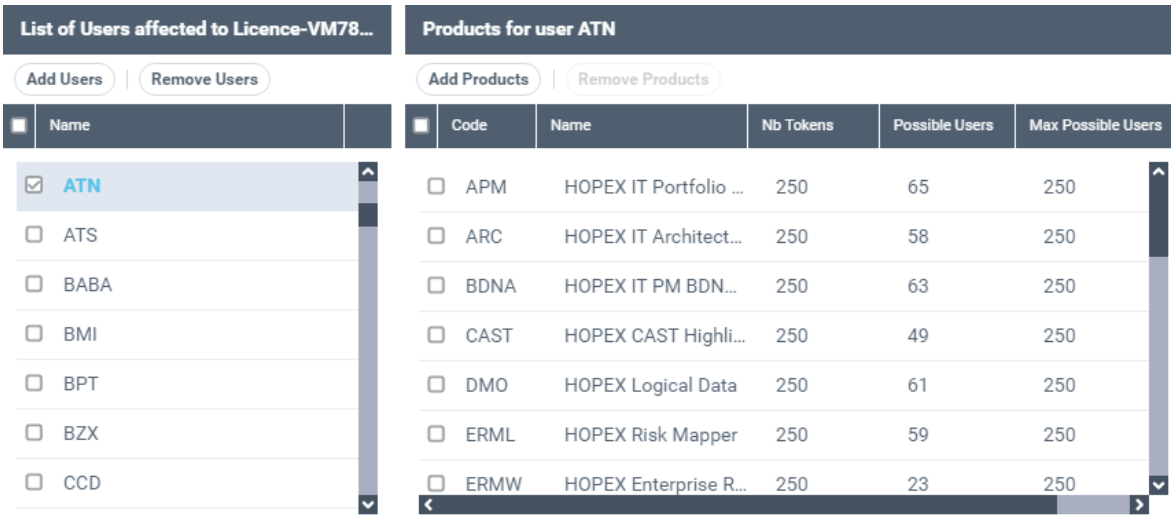
Accessing the list of products available to a user

The license description page details:

- all the users affected to the license
- for each of the users all their available products.

To access the list of products available to a user:

1. Access the **HOPEX Licensing** console.
   - *See "Accessing HOPEX Licensing Console", page 55.*
2. In the **HOPEX Licensing** navigation panes, click **Licence-<license reference>**.
3. In the right pane, in the **List of Users affected to <license name>**, select the login **Name** of the user concerned.
   The **Products for user <login name>** pane lists all the products available to the user.



It details for each product:

- its **Code**
- its **Name**

Managing the users of a license

At any time, you can add or remove users to/from the license.

To manage the users of a license:

1. Access the **HOPEX Licensing** console.
   - *See "Accessing HOPEX Licensing Console", page 55.*
2. In the **HOPEX Licensing** navigation panes, click **Licence-<license reference>**.

3. In the right pane, in the **List of Users affected to <license name>**, to:
   - add users to the license: click **Add Users**, enter each user's login names, separated by a ";" and click **Save**.
   - remove users from the license: select the login **Names** of the users concerned and click **Remove Users**.

   Users are added or removed to/from the license accordingly.

Managing the license products available to a user

You can add/remove products of a license to/from a user.

To manage the products available to a user:

1. Access the **HOPEX Licensing** console.
   - *See "Accessing HOPEX Licensing Console", page 55.*
2. In the **HOPEX Licensing** navigation panes, click **Licence-<license reference>**.
3. In the right pane, in the **List of Users affected to <license name>**, select the login **Name** of the user concerned.
4. To:
   - add products to the user: click **Add products**, select the products and click **Save**.
   - remove products from the user: in the available product list, select the products and click **Remove products**.

   Products are added or removed to/from the user accordingly.

Securing the platform

Securing the platform **1**

Introduction **4**

Securing the access to the application **5**

Activating SSL on the website 5

Disabling SSL v2, v3, TLS 1.0 and TLS 1.1, Triple DES 128, RC4 128/128 14

Block the TRACE HTTP request 19

Remove the default install page in IIS 23

Securing the RDP access (Terminal Services) 26

Configuring the firewall 29

Configuring the default error page of the application to hide application errors 29

Protection against ClickJacking 33

Securing the ASP.NET session cookies 35

Hide ASP.Net version header 37

Remove IIS Server version HTTP Response Header 38

Strict-Transport-Security HTTP header 39

Manage content type options 41

Restrict Cross Origin Ressource Sharing to Trusted Domains 41

Search engine protection 42

Securing the application **43**

Hiding the error details 43

Activating the automatic logoff 44

Hiding the information when entering the wrong credentials 46

Manage password activities 46

Documents Upload 48

Modules 48

INTRODUCTION

The goal of this document is to give detailed instructions as to how to secure the HOPEX platform. This document is valid for HOPEX V4 version.

Activating SSL on the website

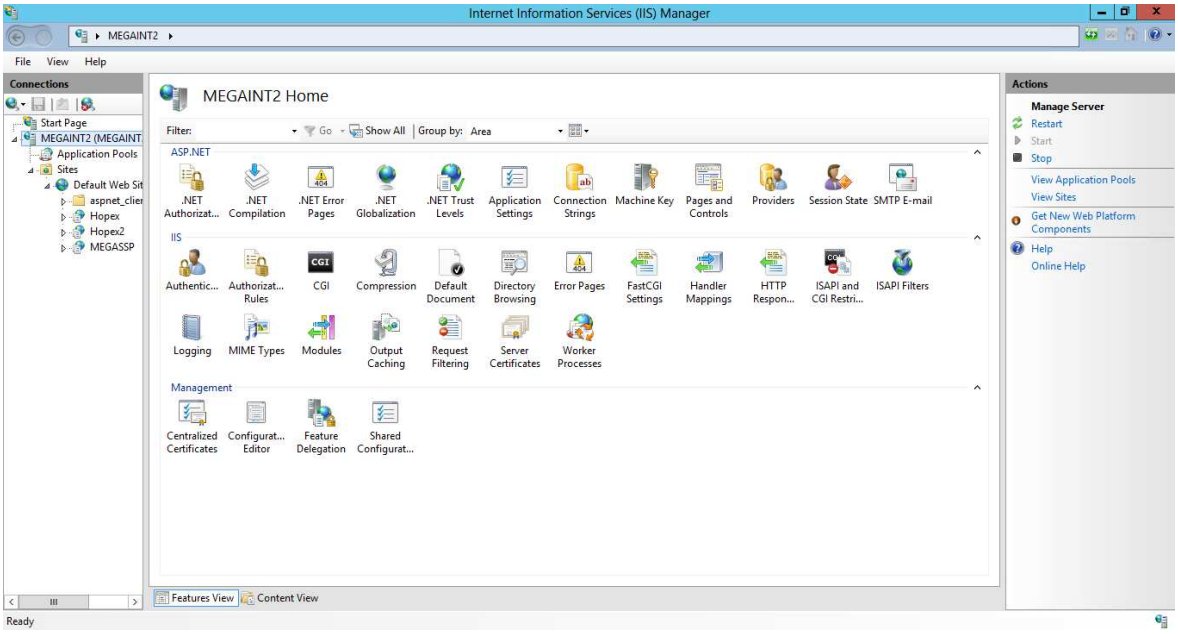
In order to encrypt the information sent between the users' workstation and the web server, the first step is to activate the SSL on the Hopex website.

The minimum level of certificate that is recommended is a certificate signed by a trusted third party. If internally, a root certificate exists and allows the applications to generate their own certificate, it is also a possibility.

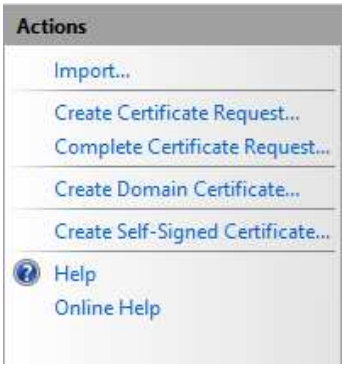
When you obtained that certificate, the steps to import it on your website are multiple. You will find hereafter one way to do it, and make sure that your website will be attainable only through HTTPS.

The following step-by-step procedure was done on a Windows Server 2012, with IIS 8.0. You need to adapt it based on your OS version.

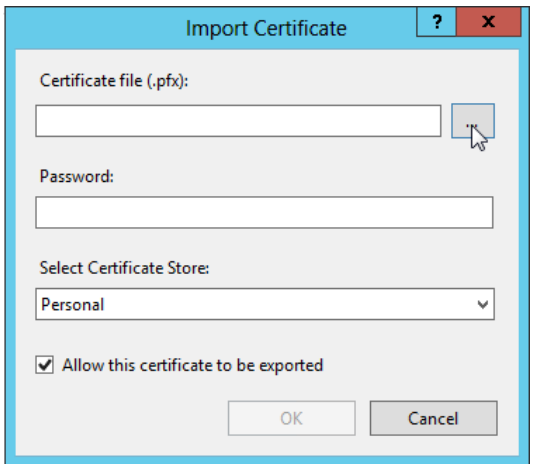
1. Copy the certificate on your server.
2. Open the "Internet Information Services (IIS) Manager" through the Administrative Tools of the server.
3. Go to the root of IIS :



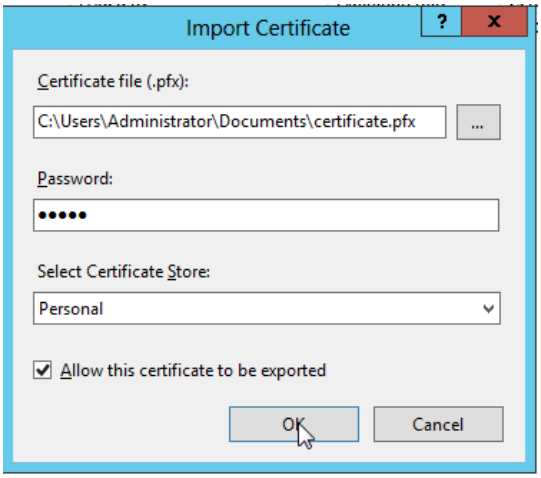
4. Double-click « Server Certificates ».
5. Once this feature is opened, on the right pane, click on the « Import » link :



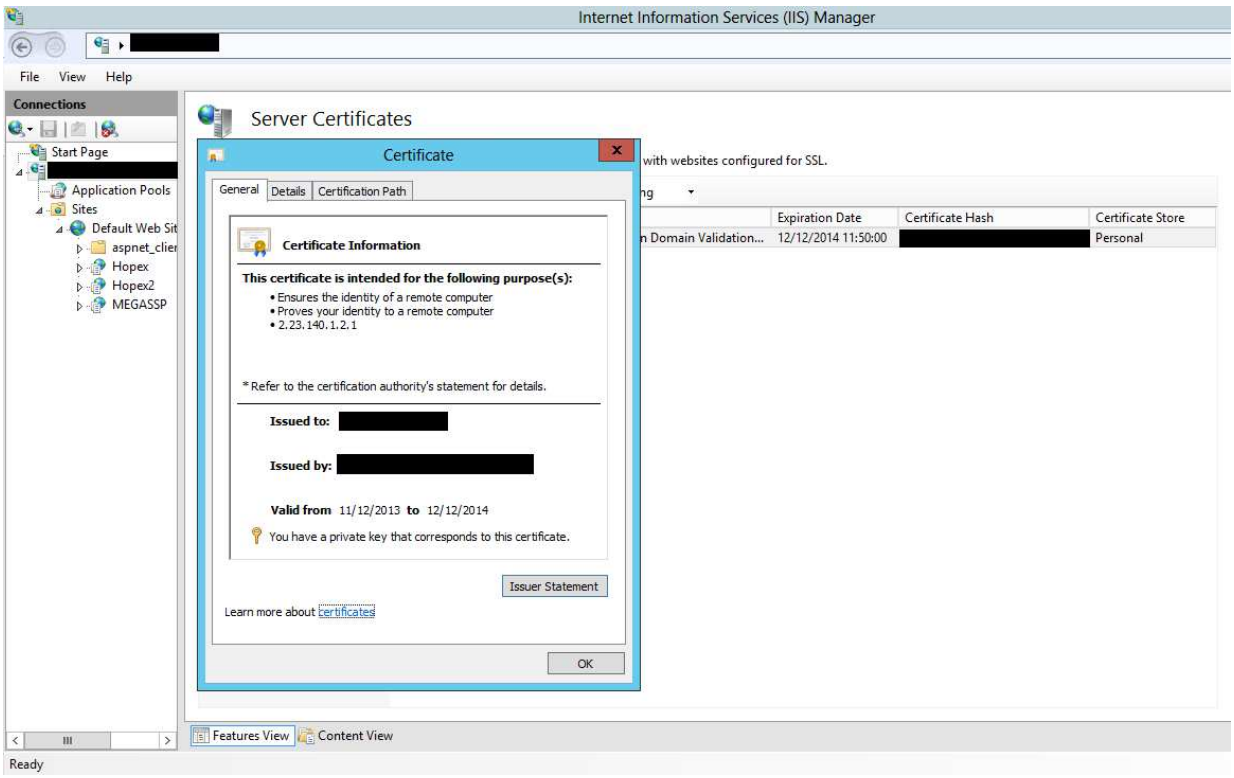
6. Click the button where the cursor is located, in order to look for the certificate :



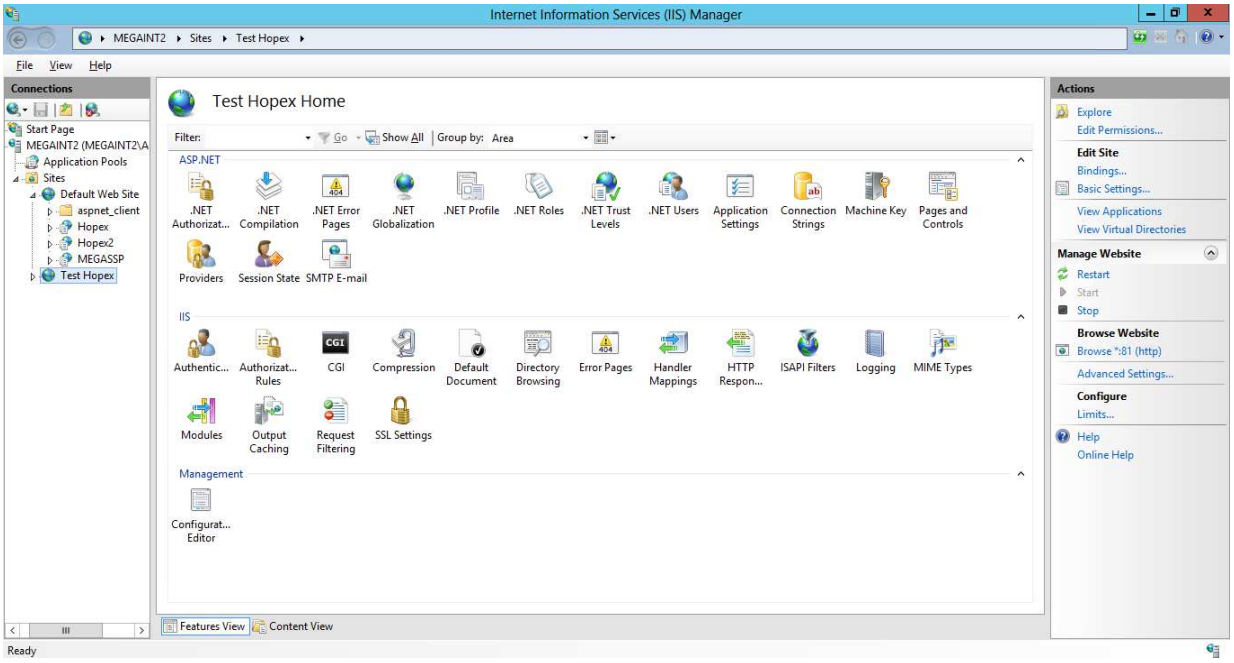
7. Select the certificate. Provide a password if necessary, and click « OK » to import the certificate :



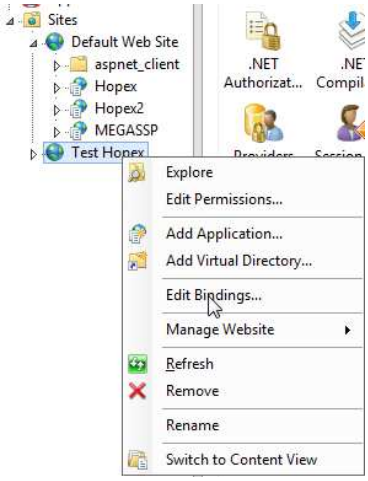
A new line will appear in the “Server Certificates” window. You can have a look at the imported certificate by double-clicking on the certificate :



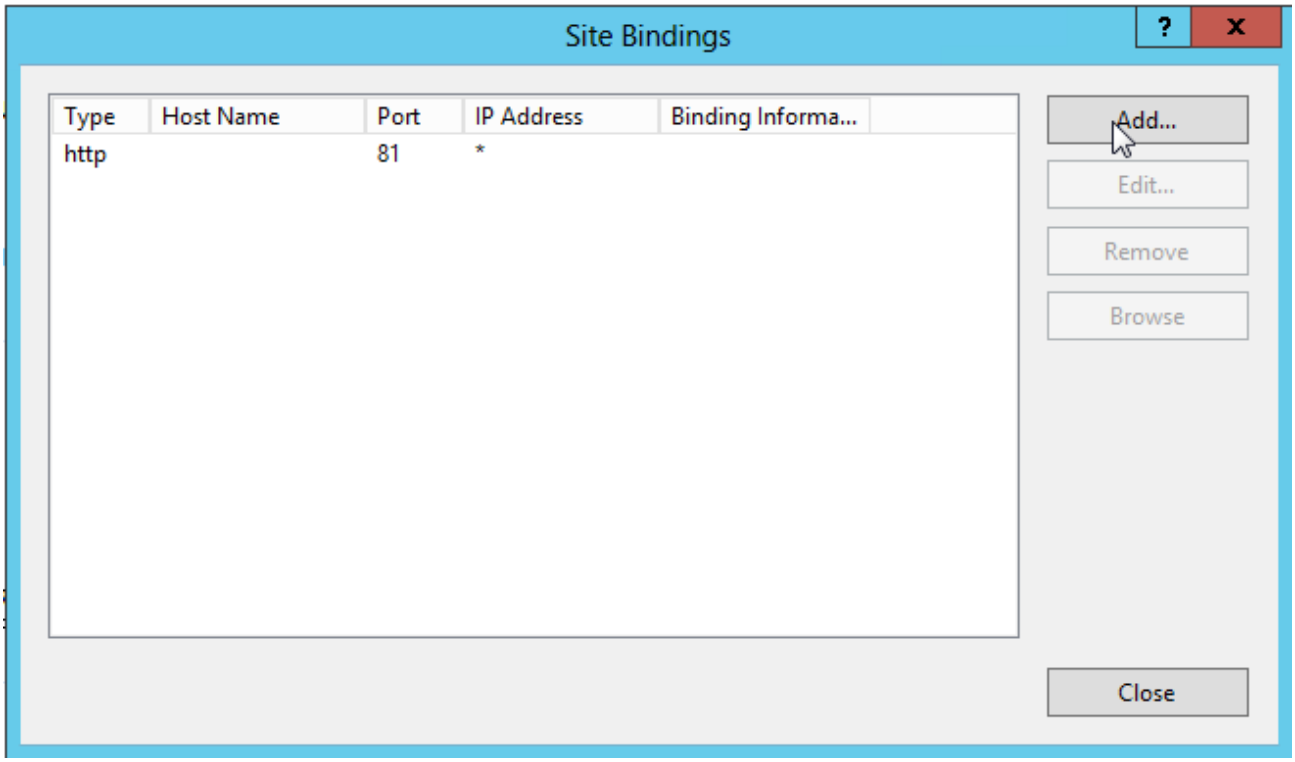
8. Locate the website where you want to create an HTTPS binding with the newly imported certificate. In this example, it is done on the "Test Hopex" website. Click the website to see its features.



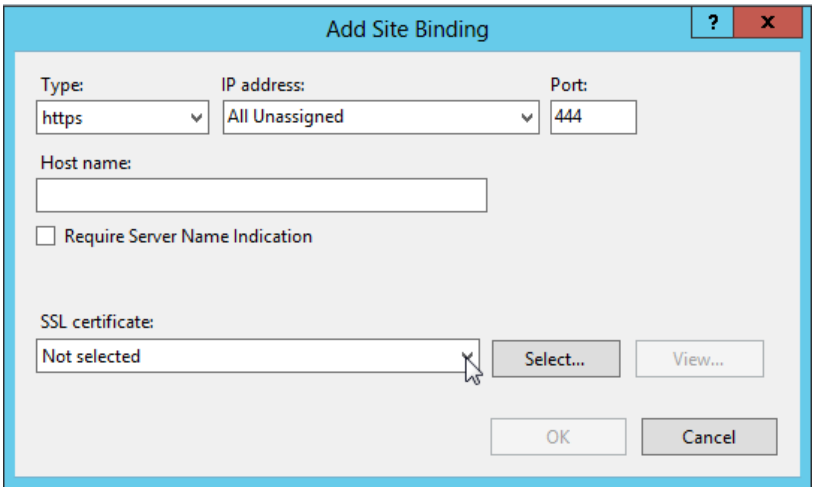
9. Right-click the website and select **Edit Bindings**.



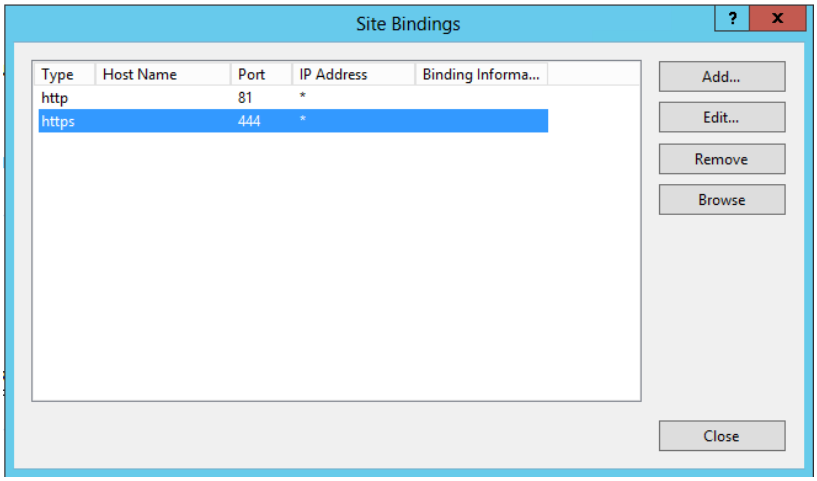
10. Add a new binding, for HTTPS, on a specific port (in this example, as the port 443 is already used, we will choose 444). Use the default port 443 whenever possible, as it permits you quite easily to make the URL of the website user-friendly :



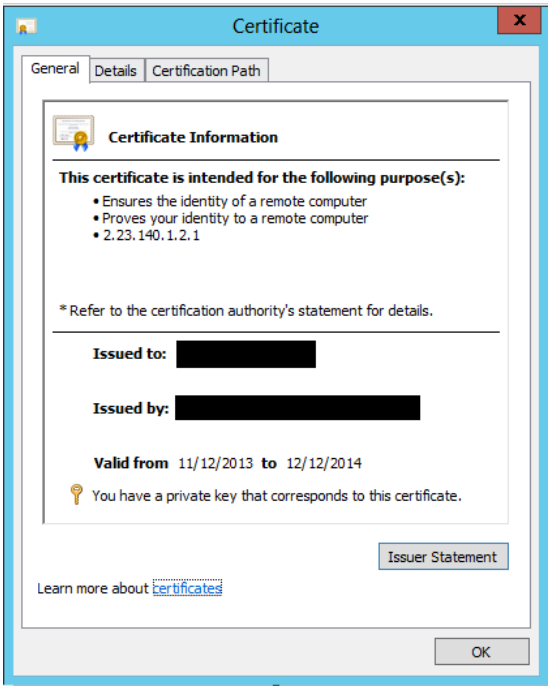
11. Use the dropdown list to select your certificate and click **OK**.



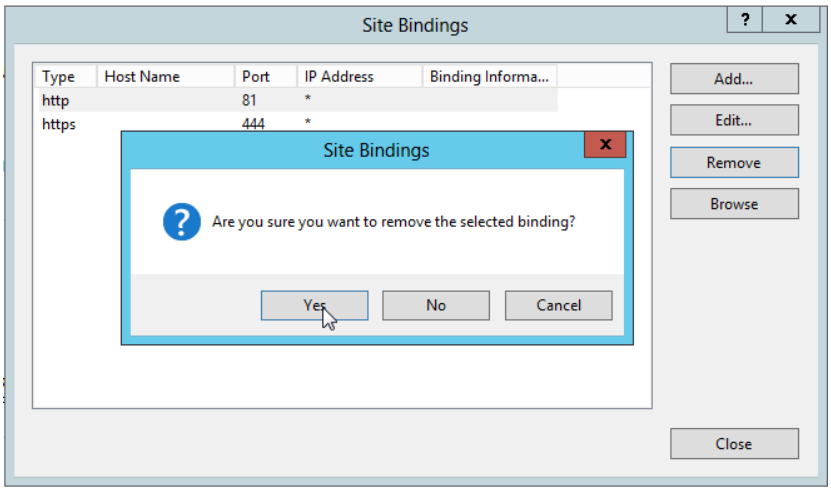
12. You see that a new binding exists. You can click **Close**.



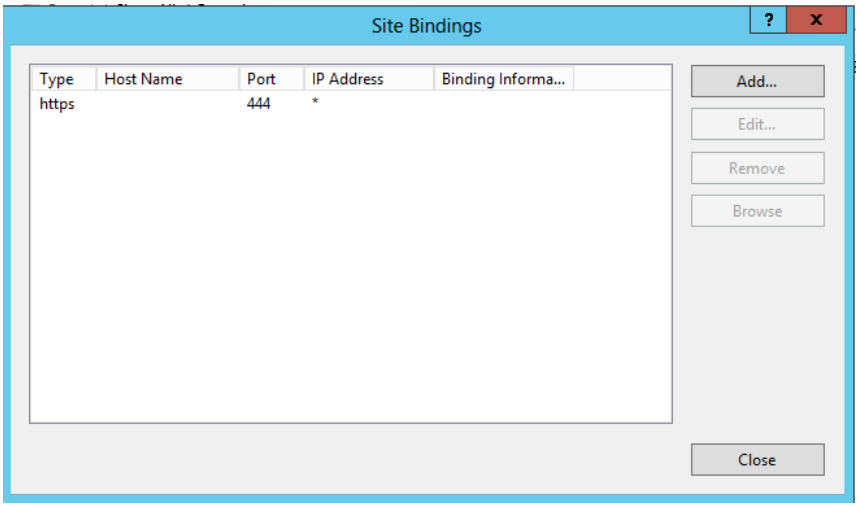
13. Install the application. Choose the appropriate website (here “test hopex”), and let the installation completes.
14. Whenever a URL is requested in the installation steps, make sure that you provide the HTTPS link, with possibly the port number, and that the address is relevant with the “Issued To” parameter when you open the certificate. Otherwise, you will receive some error messages in the application:



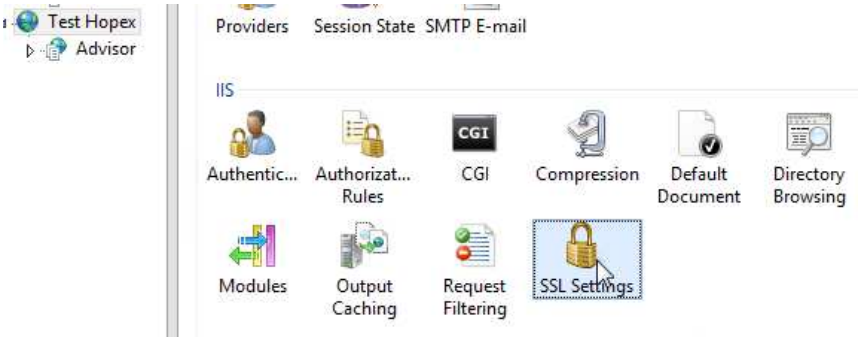
15. Once this is done, go back to IIS to force the use of SSL for your website. To do that, start by going back to the previous window to edit the bindings.
16. Select the "http" binding, and click **Remove**.



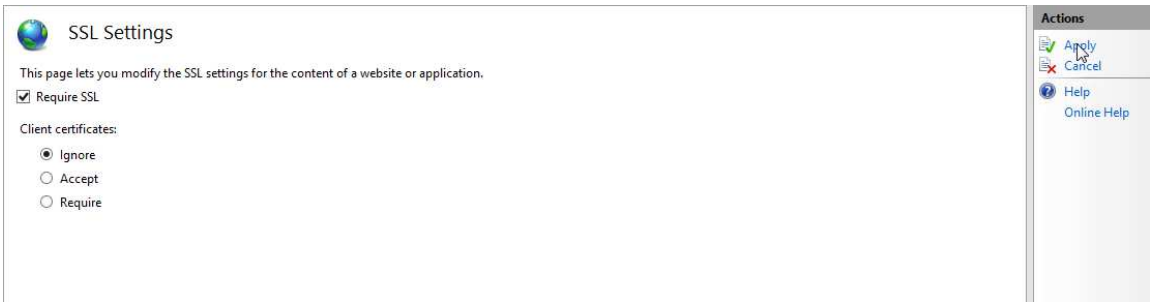
17. Check that only the https binding remains, and click **Close**.



18. Select your website, and open the « SSL Settings » feature :



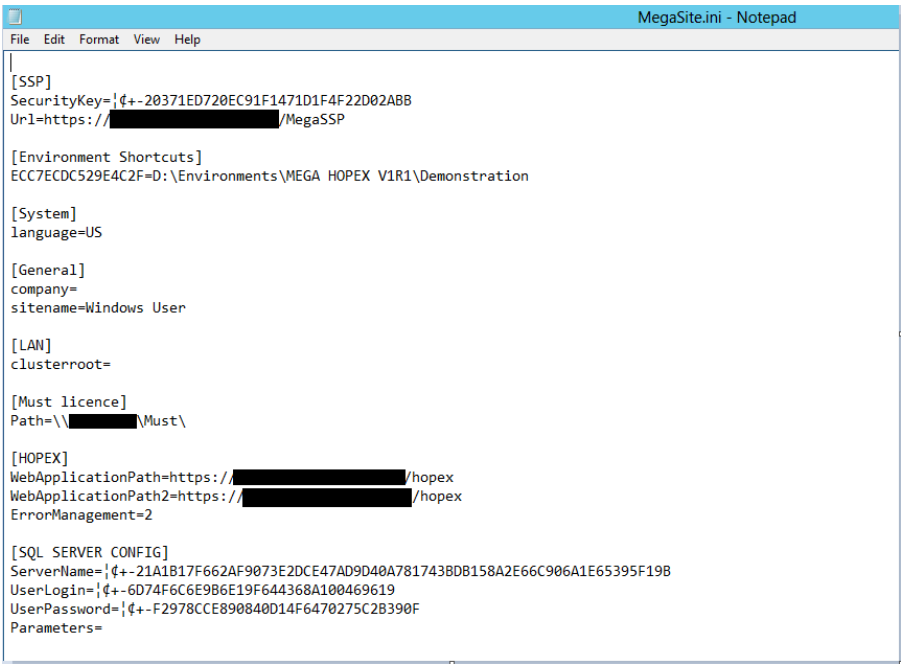
19. Click « Require SSL » and apply :



20. Install the Hopex Web front-end application on your server.

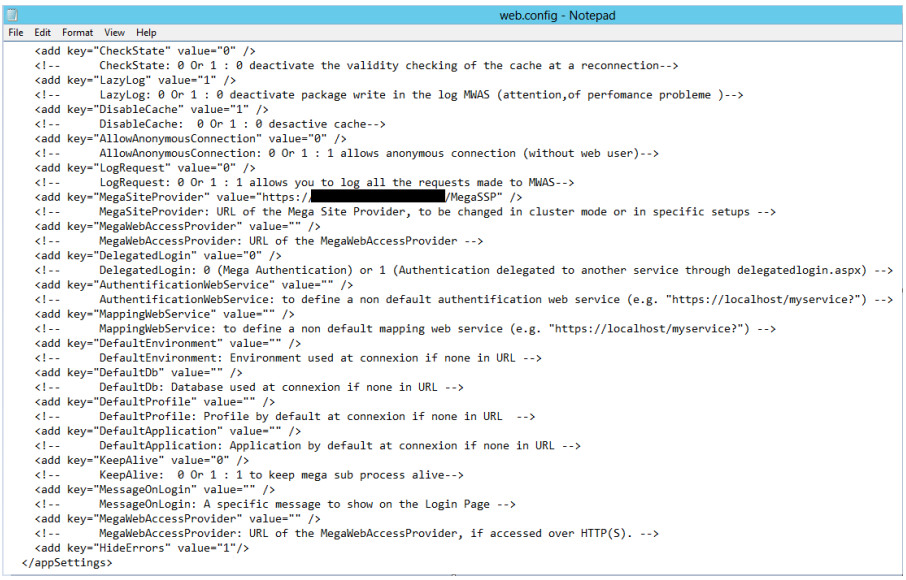
21. Last step, after the installation, is to check that the configuration files of the application all contain the proper URL. You can also install the application without SSL, and then decide to activate it. In that second scenario, you need to update the following files. If you installed with SSL activated, you just need to check the configuration with the next steps. Two locations contain such strings: the *web.config* file of the "Hopex" web application, and the *MegaSite.ini* in the "Cfg" folder of the installation folder of the application:

a. In the MegaSite.ini file, the URL, WebApplicationPath, and WebApplicationPath2 parameters need to reflect that use :

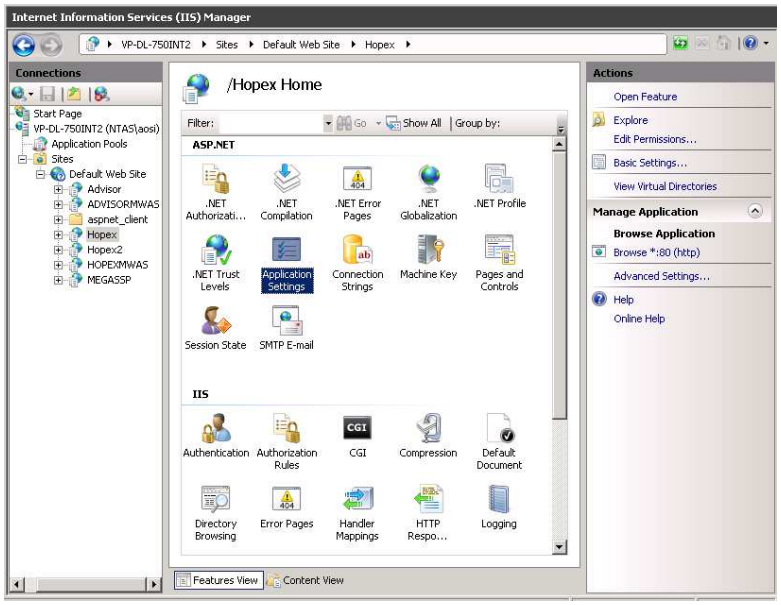


b. Application settings of the Hopex web application (two techniques), the parameters MegaSiteProvider, and MegaWebAccessProvider, also need to be switched to HTTPS with the proper port :

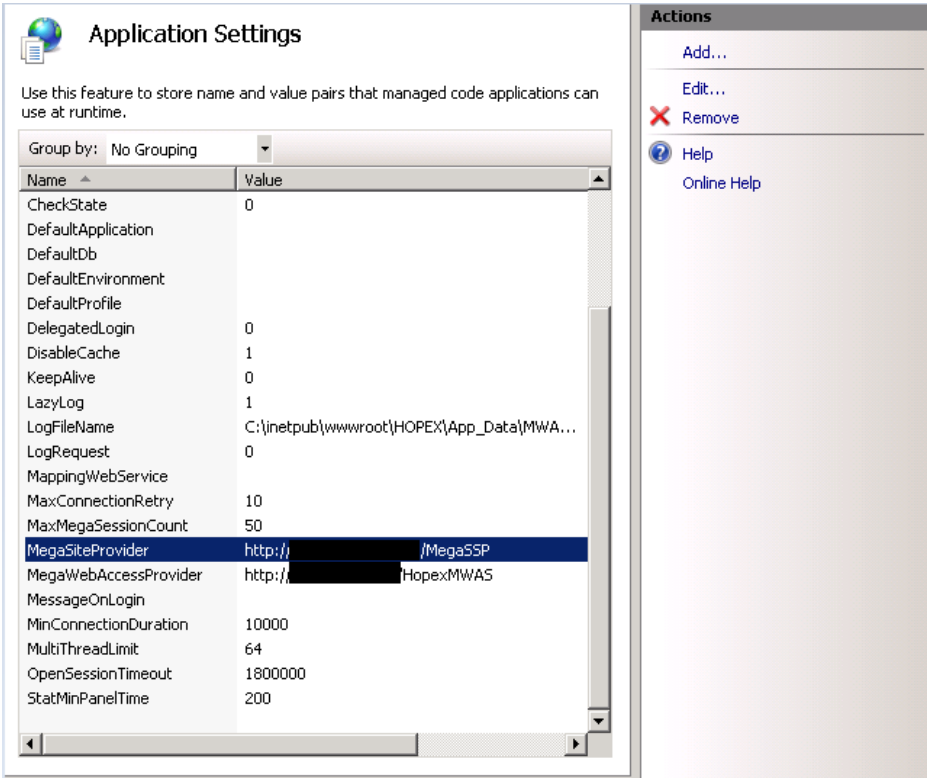
i. In the web.config file :



ii. Through the « Application settings » feature in IIS :



You can select a line and click the **Edit** link on the right pane to update a URL :



Lastly, a you need to edit the « web.config » file of the HOPEX web application.

In the <system.web> element, add the following element:

```
<httpCookies requireSSL="true" />
```

An example of the file properly configured:

Note that in the latest CPs of the different versions of HOPEX, that line might already be present in the file, commented. Remove the comment brackets if it is:

```
<!-- <httpCookies requireSSL="true" /> -->
```

To

```
<httpCookies requireSSL="true" />
```

Save the file to apply that modification. IIS will automatically restart.

Disabling SSL v2, v3, TLS 1.0 and TLS 1.1, Triple DES 128, RC4 128/128

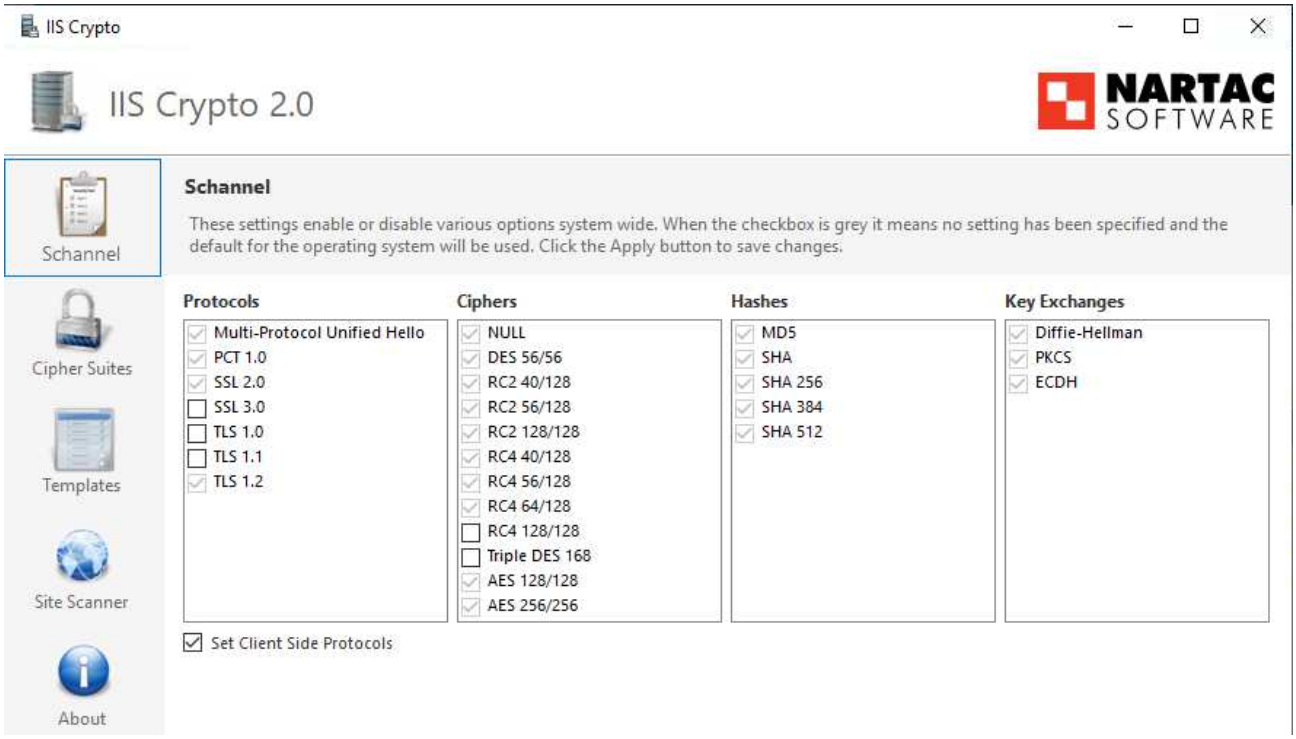
On Windows Server 2008 R2 and Windows Server 2012, the SSL V2 and V3 are activated by default. As there are known vulnerabilities with those, as well as with TLS 1.0, as well as TLS 1.1, we disable those, and then activate only TLS 1.2.

First, Download Secure Server Toolkit at the following location:

https://megaintl.sharepoint.com/:u:/r/sites/rd/dev/km/Security/Secure%20server%20toolkit.zip?csf=1&web=1&e=yPO2VI

Execute all reg files on the server.

Validate the above protocols are effectively disactivated:



On top of that, Follow the steps below:

For SSL V2:

1. Through the Startup menu, go to "Run" and type :

   regedit.exe

2. Browse through the registry until you reach the following key :

   HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL 2.0

3. In the "SSL 2.0" key, create a new one, of type DWORD (32 bit) with the following details :
   a. Name : Enabled
   b. Value : 0
4. If it does not exist, create a sub-key in "SSL 2.0" called "Client". Otherwise, go to 5. Directly.
5. In the key "CLIENT", create a DWORD (32 bit) with the following details:
   a. Name : DisabledByDefault
   b. Value : 1

For SSL V3:

1. Through the Startup menu, go to "Run" and type :

   regedit.exe

2. Browse through the registry until you reach the following key :

   HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL 3.0

3. In the "SSL 3.0" key, create a new one, of type DWORD (32 bit) with the following details :
   a. Name : Enabled
   b. Value : 0
4. If it does not exist, create a sub-key in "SSL 3.0" called "Client". Otherwise, go to 5. Directly.
5. In the key "Client", create a DWORD (32 bit) with the following details:
   a. Name : DisabledByDefault
   b. Value : 1
6. If it does not exist, create a sub-key in "SSL 3.0" called "Server". Otherwise, go to 7. Directly.
7. In the key "Server", create a DWORD (32 bit) with the following details:
   a. Name : DisabledByDefault
   b. Value : 1

For TLS 1.0:

1. Through the Startup menu, go to "Run" and type :

   regedit.exe

2. Browse through the registry until you reach the following key :

   HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols

3. Create the "TLS 1.0" key, and in that key, create a new entry of type DWORD (32 bit) with the following details :
   a. Name : Enabled
   b. Value : 0
4. Create a sub-key in "TLS 1.0" called "Server".
5. In the key "Server", create a DWORD (32 bit) with the following details:
   a. Name : DisabledByDefault
   b. Value : 1

For TLS 1.1:

1. Through the Startup menu, go to "Run" and type :

   regedit.exe

2. Browse through the registry until you reach the following key :

   HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols

3. Create the "TLS 1.1" key, and in that key, create a new entry of type DWORD (32 bit) with the following details :
   a. Name : Enabled
   b. Value : 0
4. Create a sub-key in "TLS 1.1" called "Server".
5. In the key "Server", create a DWORD (32 bit) with the following details:
   a. Name : DisabledByDefault
   b. Value : 1

To **activate** TLS 1.2 :

1. Through the Startup menu, go to "Run" and type :

   regedit.exe

2. Browse through the registry until you reach the following key :

   HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols

3. Create the "TLS 1.2" key, and in that key, create a new entry, of type DWORD (32 bit) with the following details :
   a. Name : Enabled
   b. Value : 1
4. Create a sub-key in "TLS 1.2" called "Client".
5. In the key "Client", create a DWORD (32 bit) with the following details:
   a. Name : DisabledByDefault
   b. Value : 0
6. Create a sub-key in "TLS 1.2" called "Server".
7. In the key "Server", create a DWORD (32 bit) with the following details:

a. Name : DisabledByDefault

b. Value : 0

Now we need to make sure that **the .Net layer will accept to use TLS 1.2**. To do that:

1. Through the Startup menu, go to "Run" and type :

   regedit.exe

2. Browse through the registry until you reach the following key :

   HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\.NETFramework\v4.0.30319

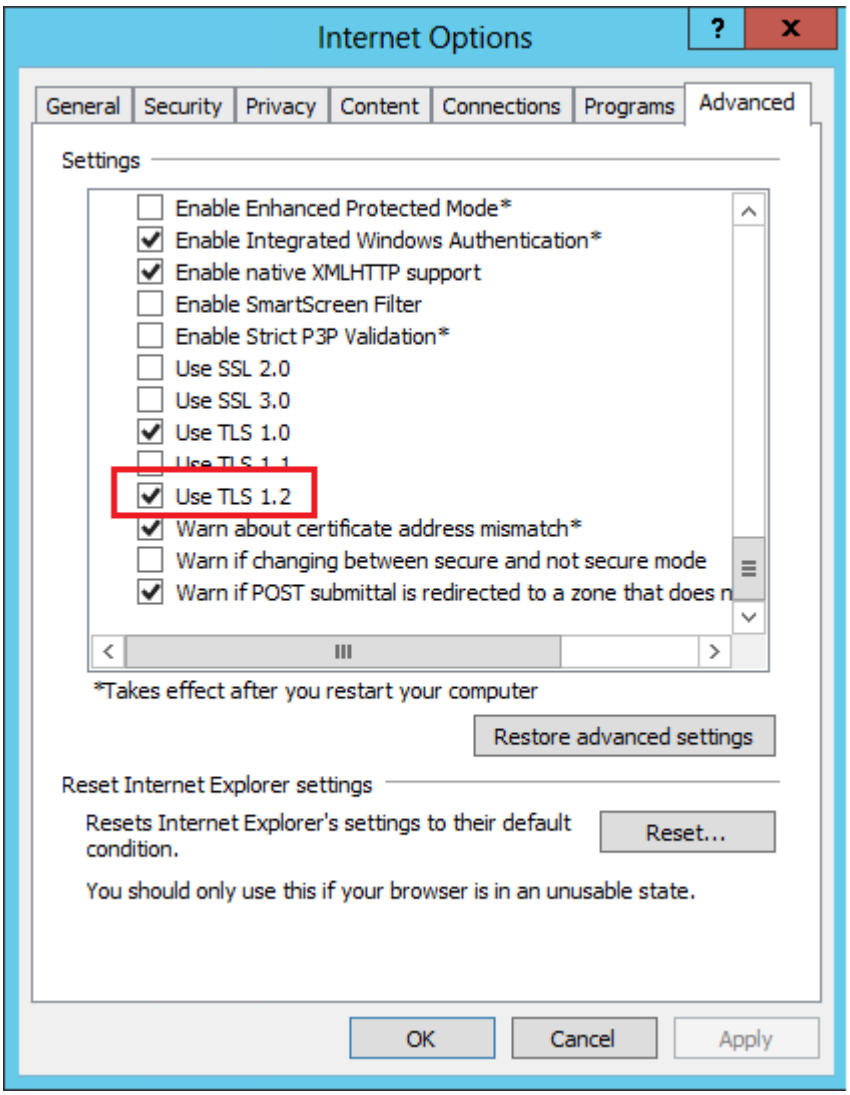
8. In that key, create a entry of type DWORD (32 bit) with the following details :

   a. Name : SchUseStrongCrypto

   b. Value : 1

Lastly, we need to make sure that Internet Explorer will allow the use of TLS 1.2. To do that, open the "Internet Options" of Internet Explorer, and go to the Advanced tab.

In the Security section, check the box "Use TLS 1.2" and uncheck TLS 1.0 and TLS 1.1:

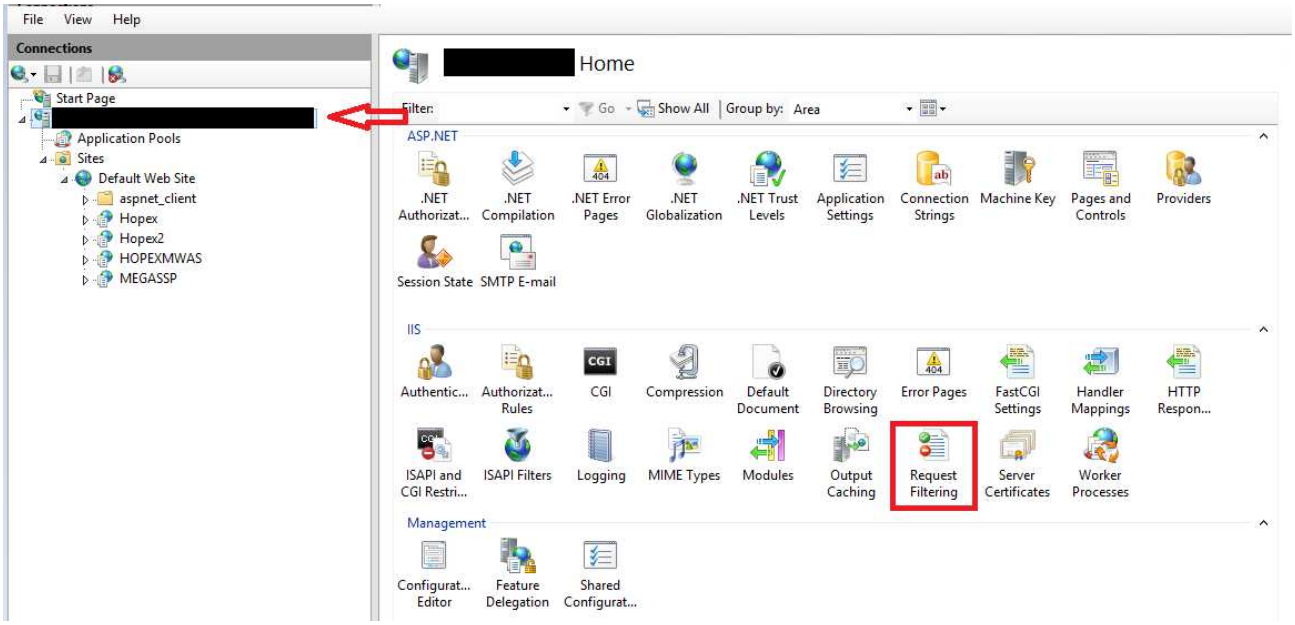


Close Internet Explorer and the registry, and reboot the server to take everything into account.

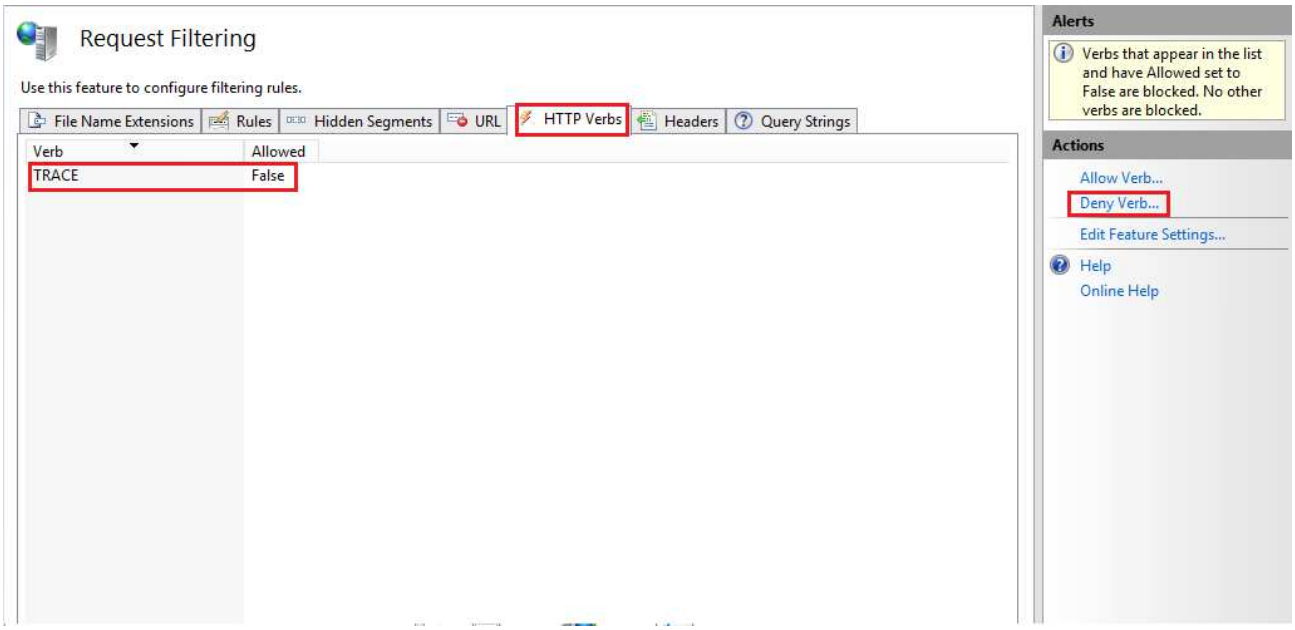
Block the TRACE HTTP request

To perform this action, three steps are needed:

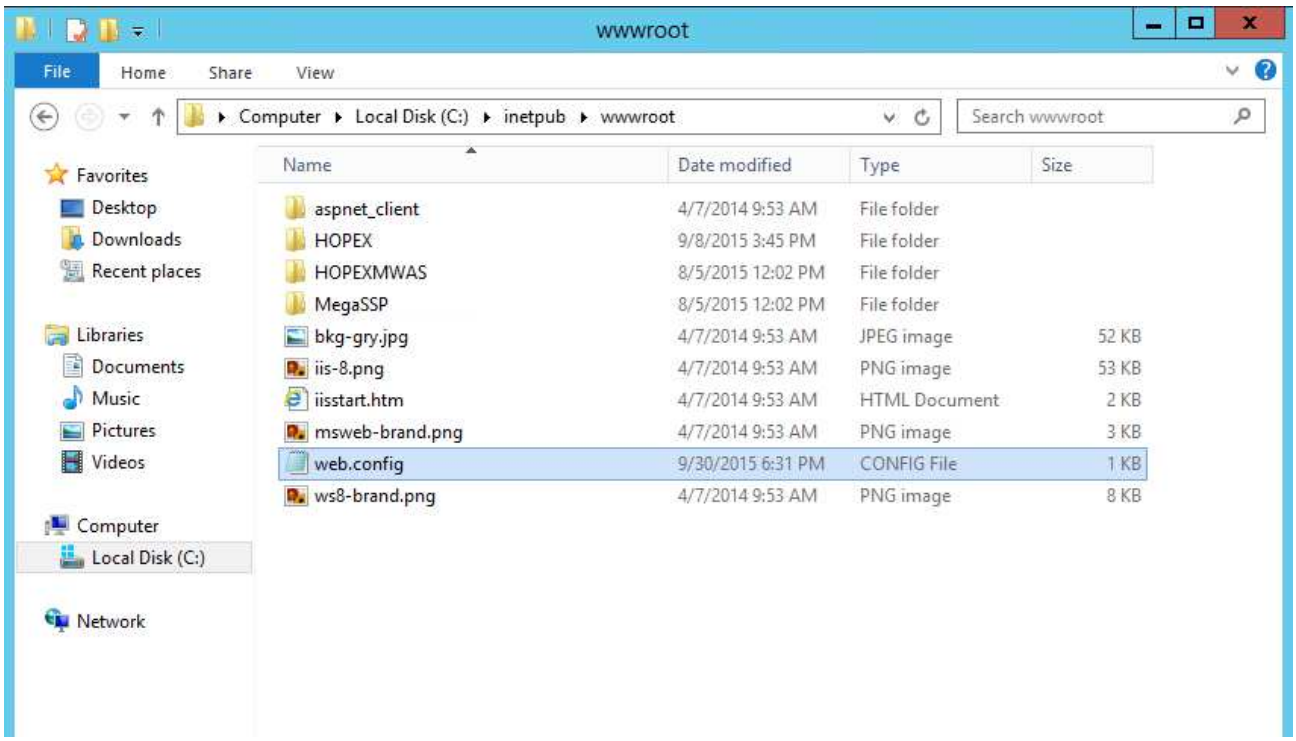
At the root of IIS, open the "Request Filtering" feature:



In the "HTTP Verbs" section, add a deny rule for the word "TRACE":



Then, open the "web.config" file at the root of your website:



And add the following lines：

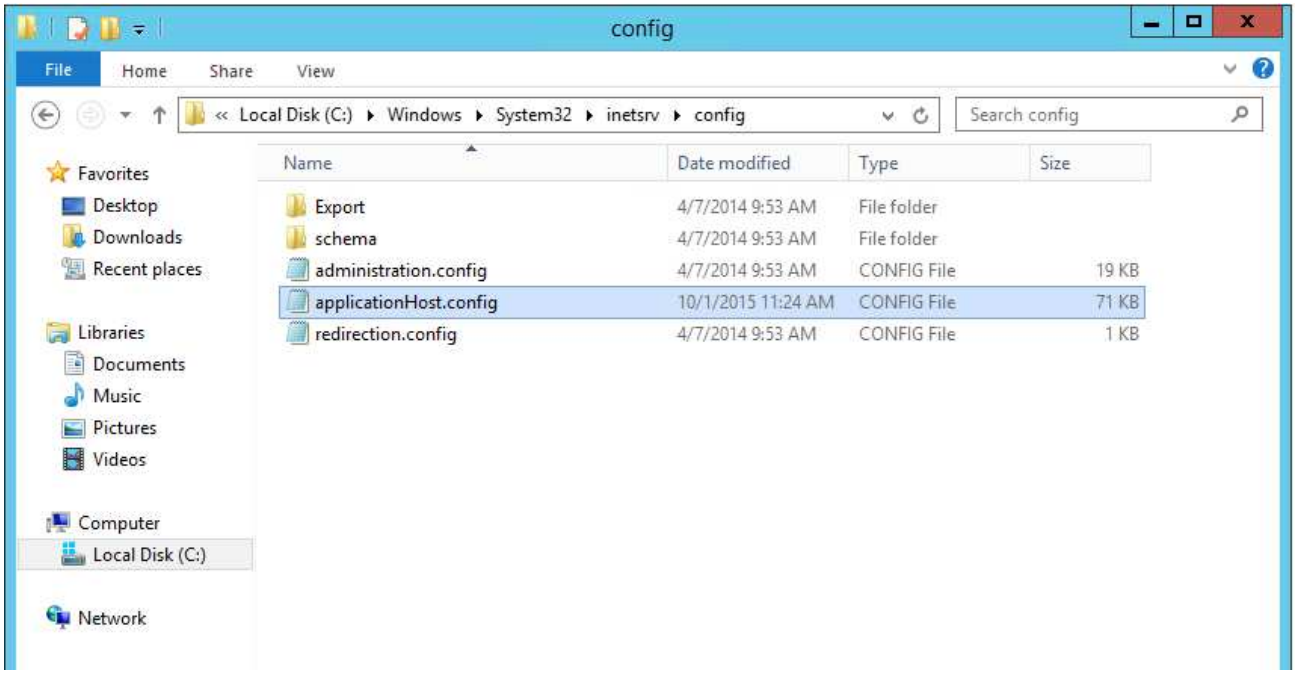
```
<authorization>
    <deny verbs="OPTIONS" users="*" />
    <deny verbs="TRACE" users="*" />
</authorization>
```

web.config - Notepad

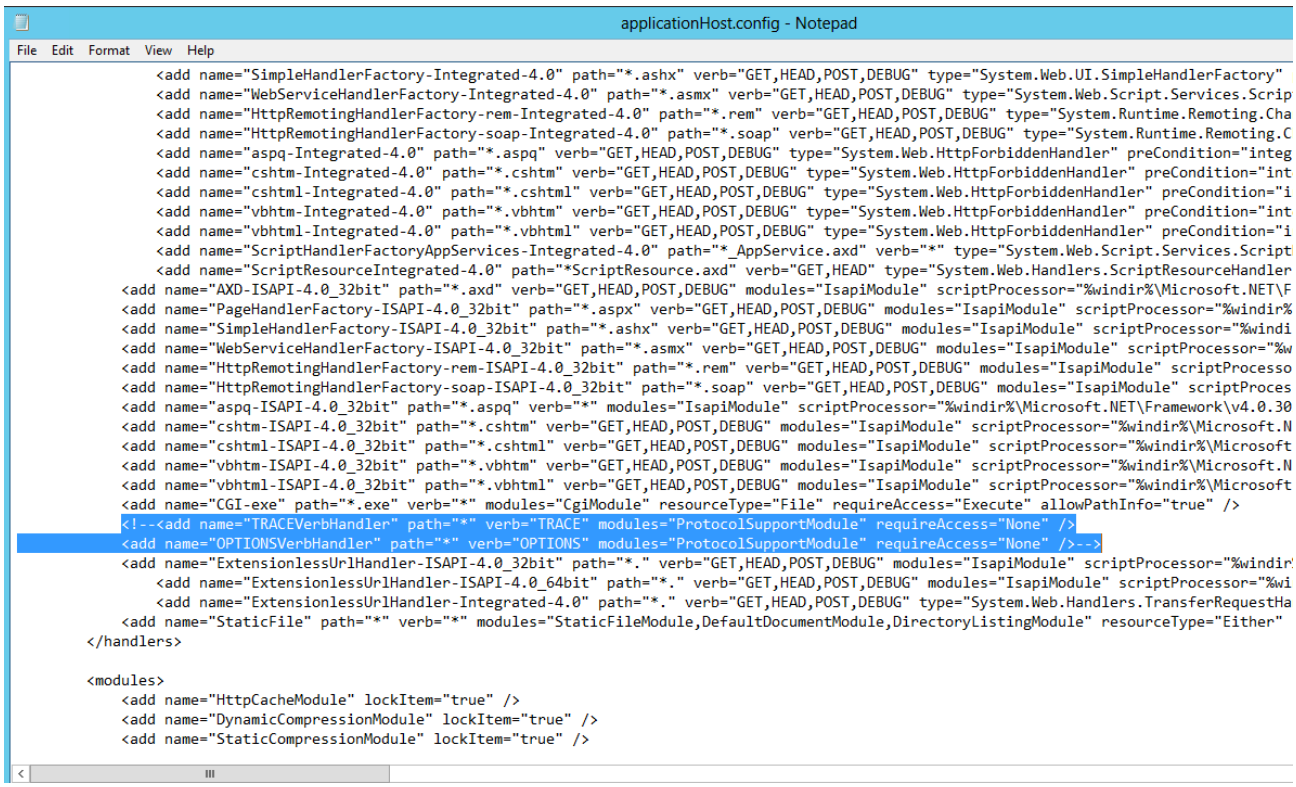
File Edit Format View Help

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <system.webServer>
        <defaultDocument enabled="false" />
        <security>
            <requestFiltering>
                <verbs>
                    <remove verb="TRACE" />
                </verbs>
            </requestFiltering>
        </security>
    </system.webServer>
    <system.web>
 <authorization>
        <deny verbs="OPTIONS" users="*" />
        <deny verbs="TRACE" users="*" />
</authorization>
        <customErrors>
            <error redirect="https://megapoc4.com/hopex/customerrors.html" statusCode="404" />
        </customErrors>
    </system.web>
</configuration>
```

Lastly, open the "applicationHost.config" file located in the "%SystemRoot%\System32\inetsrv\config" folder:



In the "handlers" section, comment the lines for "TRACEVerbHandler" and "OPTIONSVerbHandler":

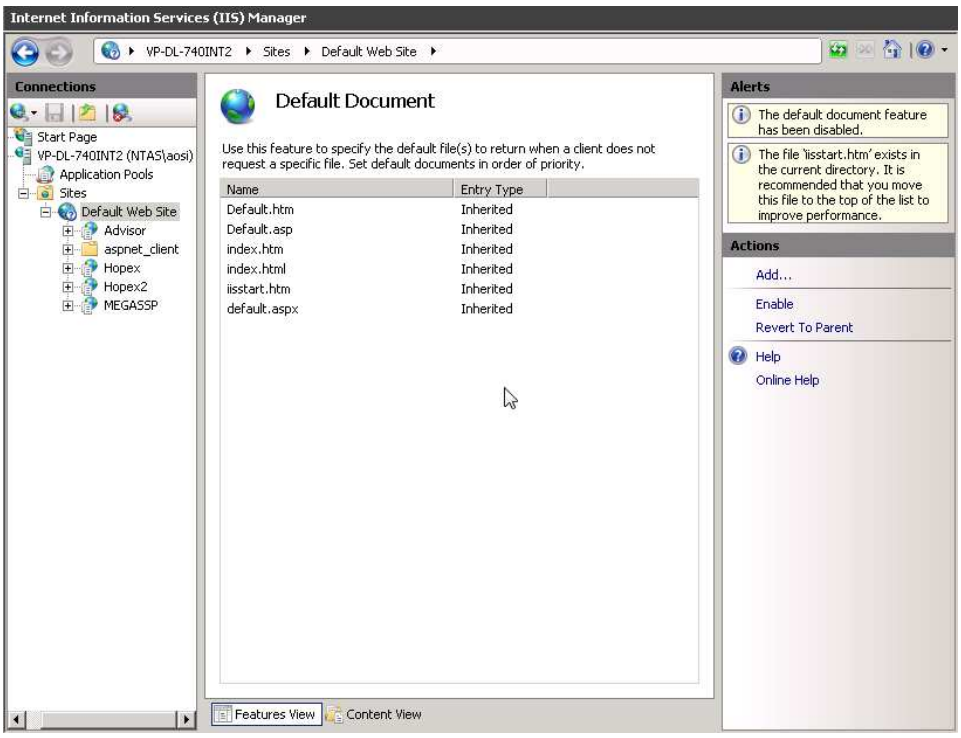


Remove the default install page in IIS

After the application is installed, you can disable the "Default Document" feature for the website on which the application is deployed.

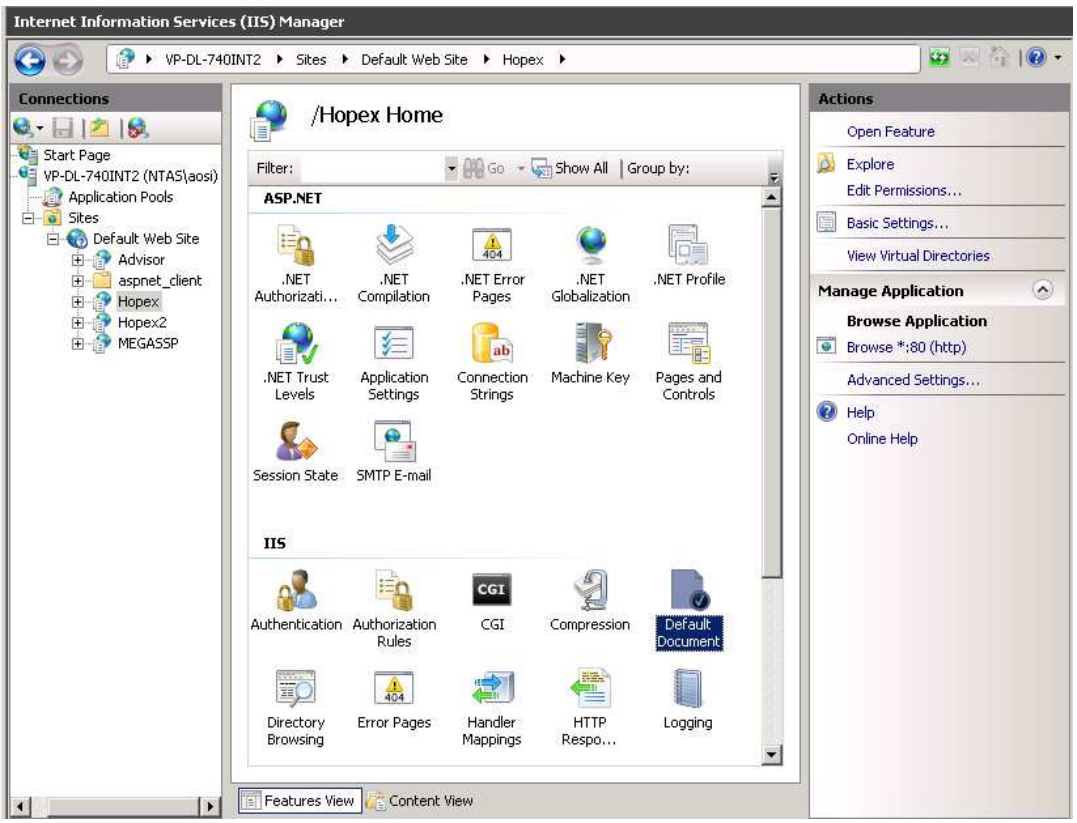
To do that :

1. Open the IIS Manager console.
2. Locate the website where the Hopex and/or Advisor web application is installed.
3. On the website, double-click **Default Document** to open this feature.
4. In the "Actions" pane on the right, click **Disable**. You should have this kind of screen after it is done :

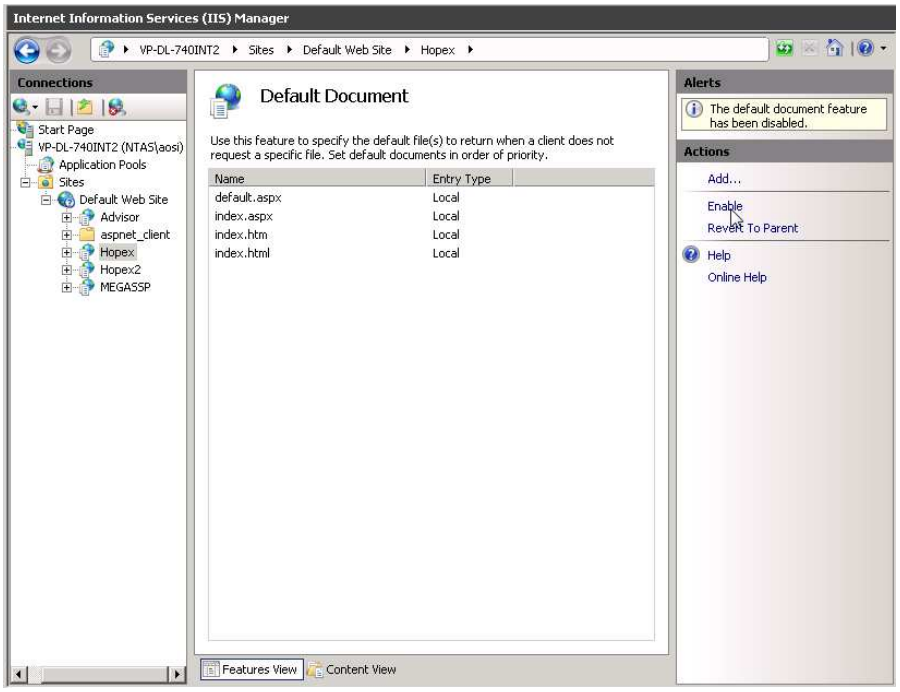


5. Make sure that the « Default Document » feature is enabled for the Hopex/Hopex2/Advisor web applications. Since it was disabled at the root level, you will have to :

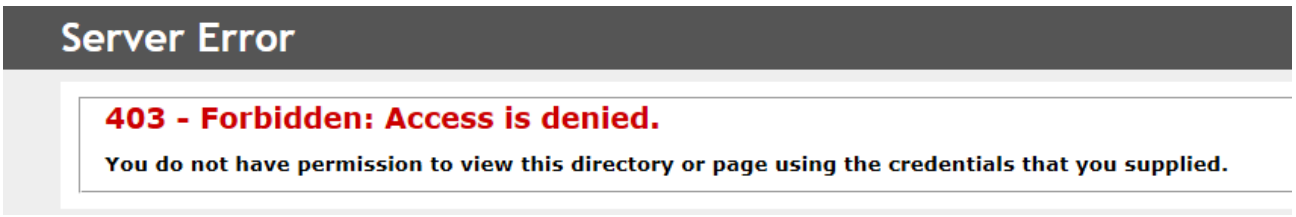
    a. Select each web application (here Hopex), and open the "Default Document" feature :



b. Click **Enable**.



c. This way, the URL https://servername won't reply,



whereas the URL https://servername/hopex will redirect the users to the login page of Mega :

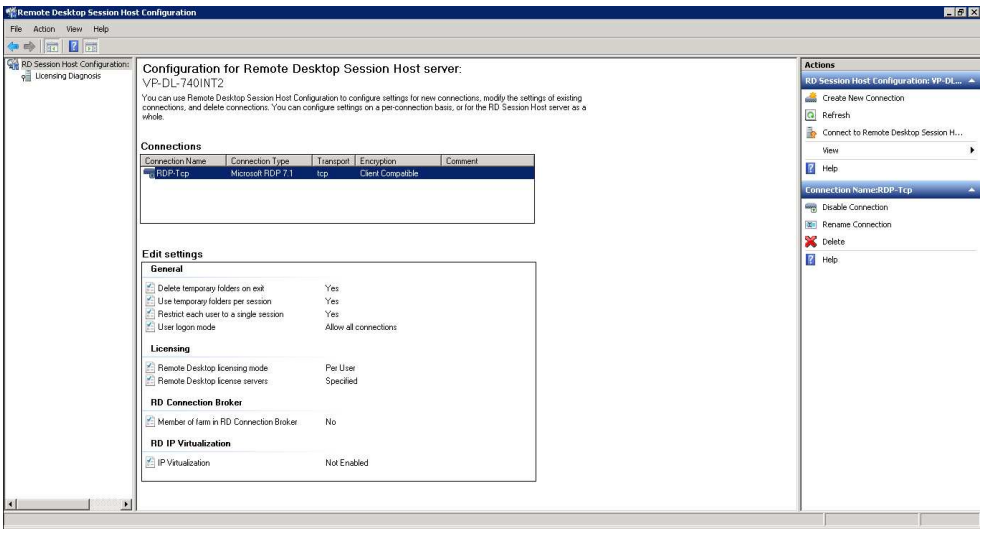
Securing the RDP access (Terminal Services)

Only valid for Windows Server 2008 R2. For later versions, we restrict the RDP access to the IPs of valid people or teams.

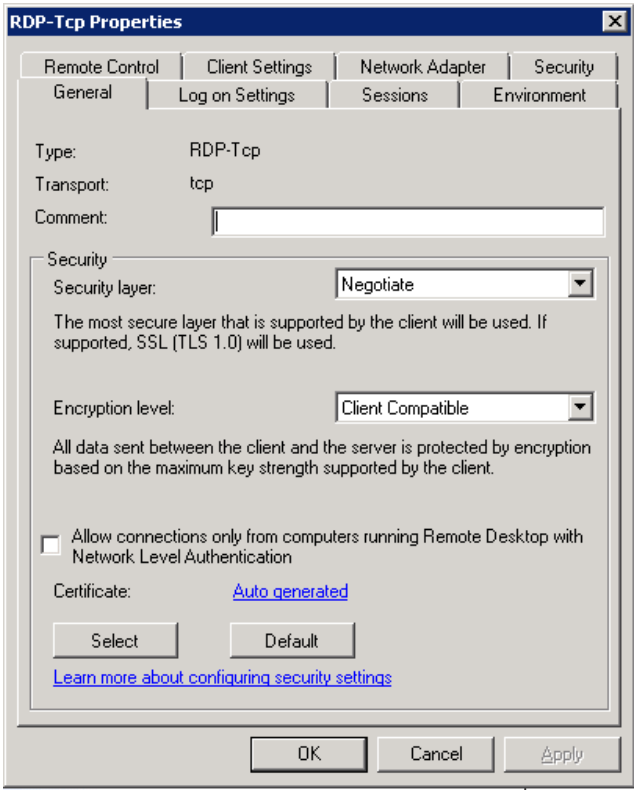
On a server hosted on a Windows 2008 R2 Operating System, you can secure the RDP layer by changing the security layer and the encryption layer of the Remote Desktop Protocol.

To do that:

1. Firstly, you have to open **Remote Desktop Session Host Configuration** the by clicking on "Start", point to "Administrative Tools", point to "Remote Desktop Services", and then click "Remote Desktop Session Host Configuration".

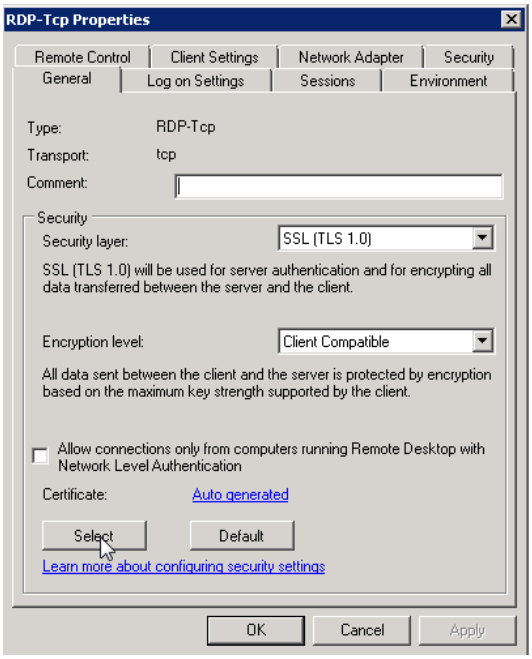


2. In the "Connections" section, do a right-lick on the "RDP-tcp" entry, and select "Properties" :

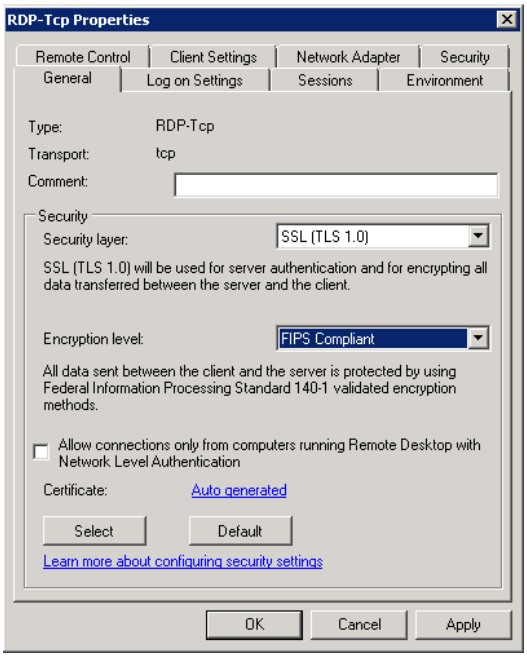


3. In the "General tab", as shown above, you can see the two security parameters that can be modified.

4. Change the "Security Layer" option to "SSL (TLS 1.0)". Be aware that you will need to have a specific certificate installed on your server to switch to that level. The "Select" button allows you to browse your installe certificates and choose the proper one:



5. Then, modify the « Encryption level » option and upgrade it to at least "High", and possibly "FIPS Compliant" :



6. Click **OK** to validate this modification and exit the configuration tool.

There are also numerous ways of securing the RDP protocol, whether it is using the Remote Desktop Gateway, filtering by IP address and user, etc.

Configuring the firewall

To avoid letting some unwanted users connect to the server or retrieve information from the server, some actions can/need to be taken on the firewall:

- If possible, restrict the access to port 3389 (RDP) to only the valid IP addresses of the System Administration, Application Administrators, and maybe the users that need to launch the Desktop client of Mega through Terminal Services.
- Restrict external access to all SMB services and ports, including TCP and UDP 135, TCP and UDP 139, and TCP and UDP 445.
- Regarding the ICMP protocol, block the following type of requests: ICMP timestamp requests (13), and ICMP timestamp replies (14).
- Limit the number of opened ports on the server to the least amount possible. Globally the server needs to be accessed through RDP, the SSL port needs to be opened, the SMB port also, and the communication port to the database server. All other ports need to be assessed in order to check that those are relevant.

Configuring the default error page of the application to hide application errors

Note that this section is necessary in the case of a remote access. While the application is only reachable through a secure network, this is not mandatory, and can make things more complex for administrators to understand what is happening on the application while errors occur on the system.

The configuration is as follows.

First, we put in the default folder of the web application a custom error file that will redirect, whenever a page doesn't respond, or a wrong page is accessed, to the default page of the application.

An example can be find below:

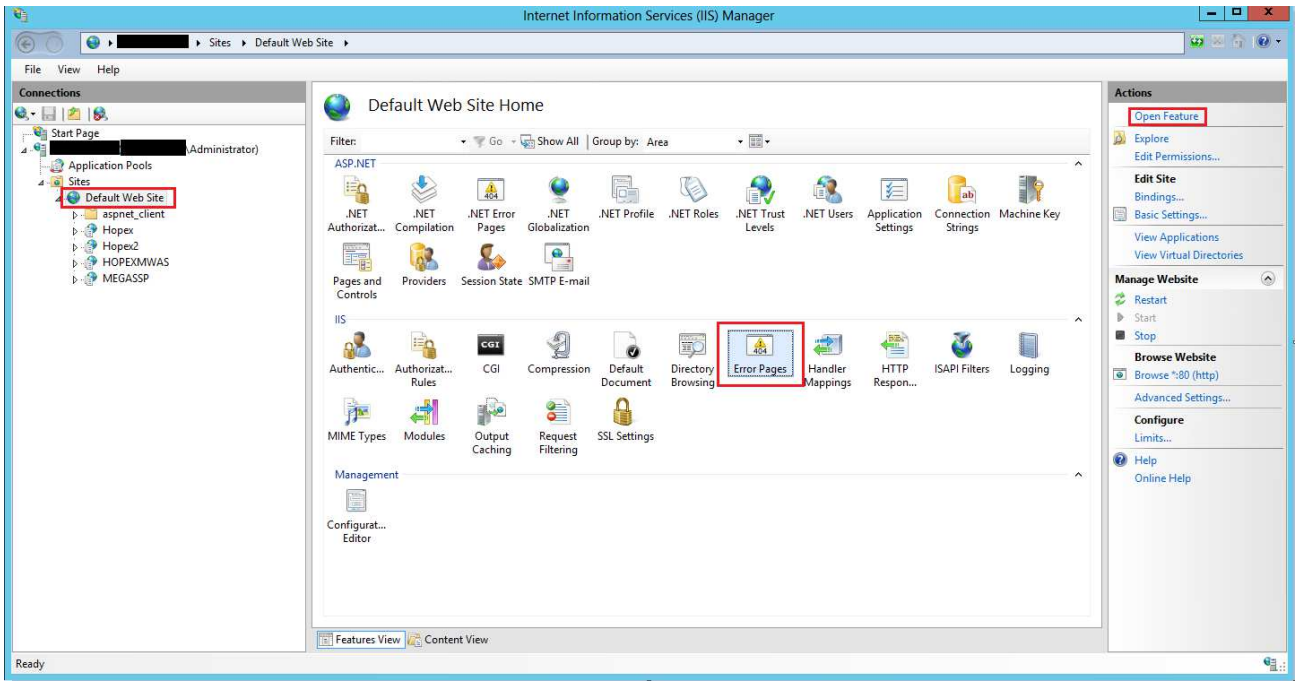
customerrors.html

Note that with the latest CPs of the HOPEX different versions, this file is most likely already in the below folder.

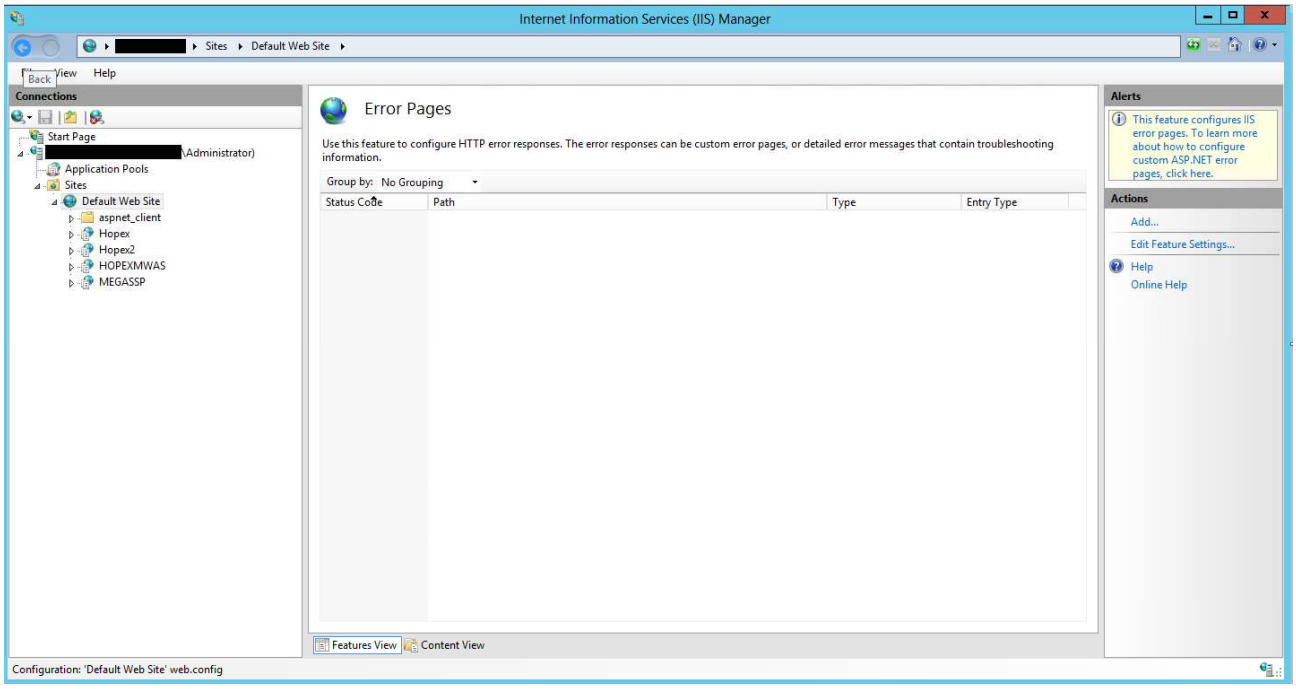
By default, it is located on the C:\inetput\wwwroot\HOPEX folder. In case you installed the "HOPEX (IIS)" feature and web application in a different location, target this path and copy the file there.

Next, we remove the default error pages of the website.

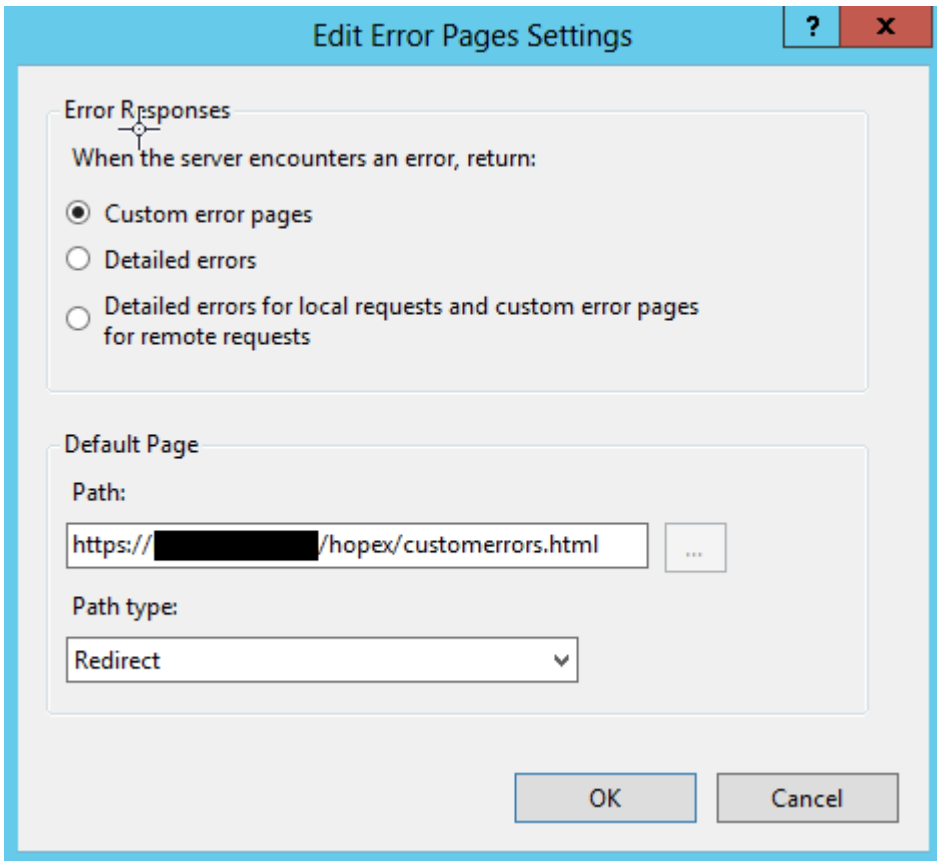
Go to IIS Manager, locate your website, and open the "Error Pages" feature:



Remove all lines to have this in the end:



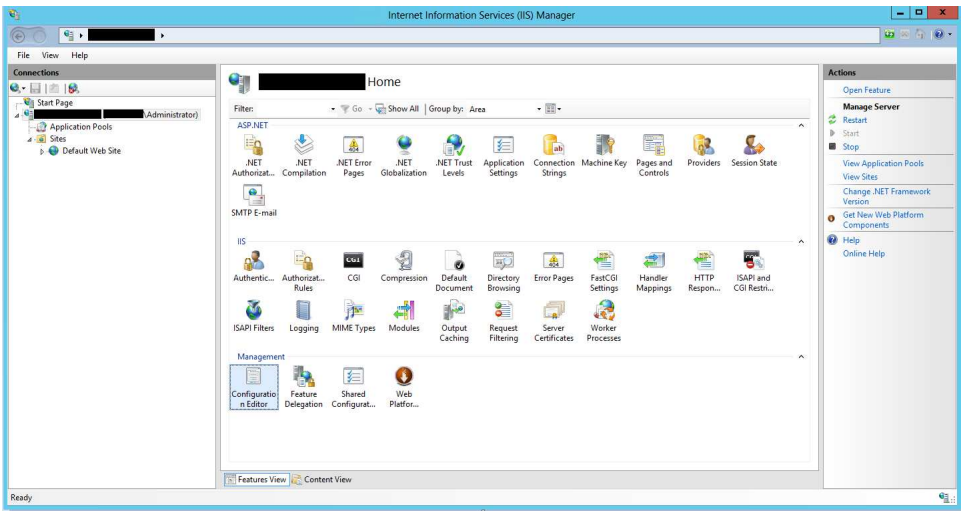
Lastly, in the same location in IIS, click on "Edit Feature Settings..." (for the "Error Pages" feature), and specify the following parameters, using the full URL of the custom error page you want to redirect to in case of error:



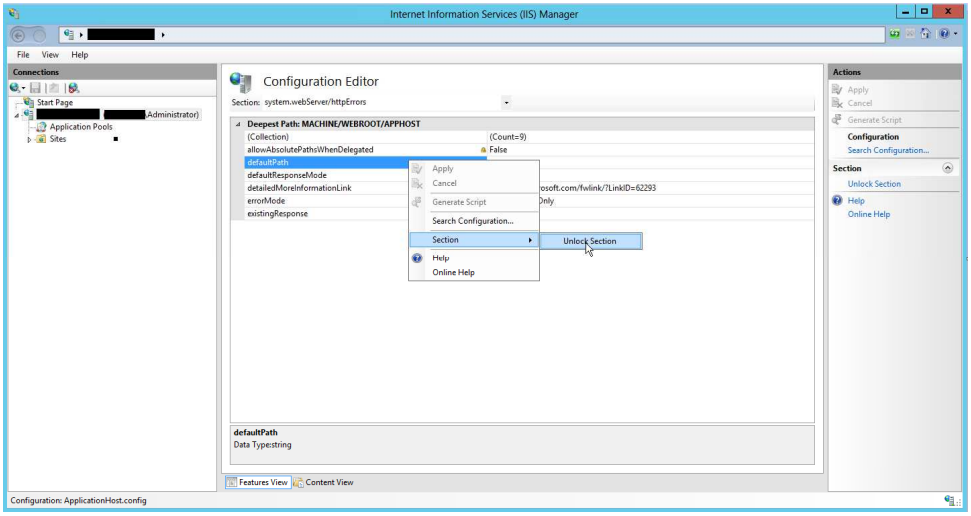
This will be reflected in the "web.config" of the website, where you will find this section ("your_url" being either the alias targeting the web server, of the name of the web server itself, depending on how the certificate was implemented) :

```
<httpErrors errorMode="Custom"
defaultPath="https://your_url/hopex/customerrors.html"
defaultResponseMode="Redirect">
            <remove statusCode="502" subStatusCode="-1" />
            <remove statusCode="501" subStatusCode="-1" />
            <remove statusCode="500" subStatusCode="-1" />
            <remove statusCode="412" subStatusCode="-1" />
            <remove statusCode="406" subStatusCode="-1" />
            <remove statusCode="405" subStatusCode="-1" />
            <remove statusCode="404" subStatusCode="-1" />
            <remove statusCode="403" subStatusCode="-1" />
            <remove statusCode="401" subStatusCode="-1" />
        </httpErrors>
```

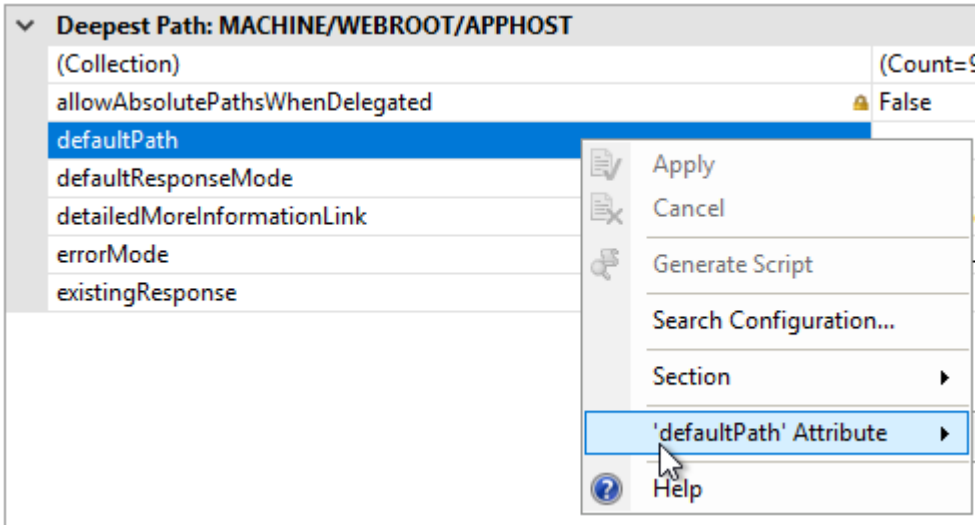
Depending on the version of IIS, you may need to first deactivate the locking of that default page. You will have to do this if you get a "lock violation" error when validating the previous setting popup.
To do this, open the "Configuration Editor" at the root level of IIS :



In “Section”, browse to “system.webServer/httpErrors”, and unlock the “defaultPath” attribute:



Apply the same configuration on the attribute.



Also, Open the c:\inetpub\wwwroot\web.config file and add the following lines in the system.web section:

```
<customErrors mode="On"
defaultRedirect="https://secured4.demomega.com/Hopex/customerrors.html">
</customErrors>
```

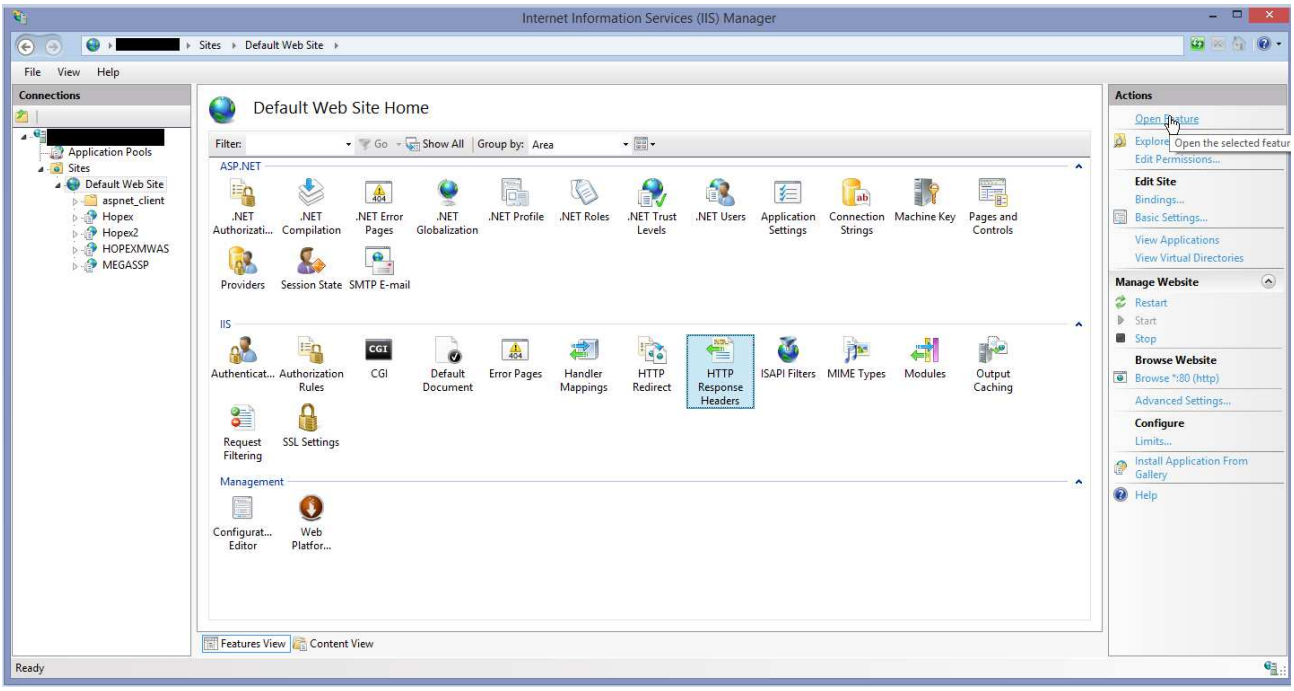
```
<system.web>
    <authorization>
        <deny verbs="OPTIONS" users="*" />
        <deny verbs="TRACE" users="*" />
    </authorization>
    <customErrors mode="On" defaultRedirect="https://secured4.demomega.com/Hopex/customerrors.html">
    </customErrors>
</system.web>
```

Protection against ClickJacking

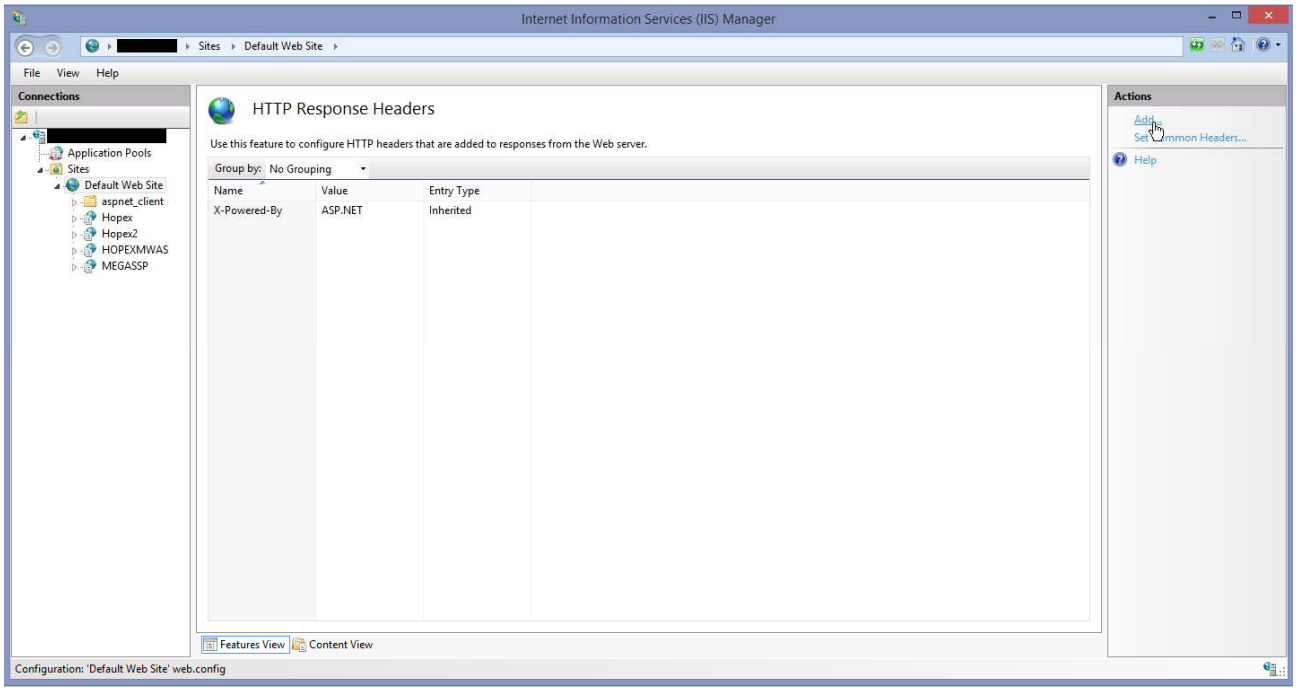
To prevent the use of ClickJacking, through Frame-Sniffing, we can configure the website.

To do so, go to IIS manager, and go to the website hosting the application.

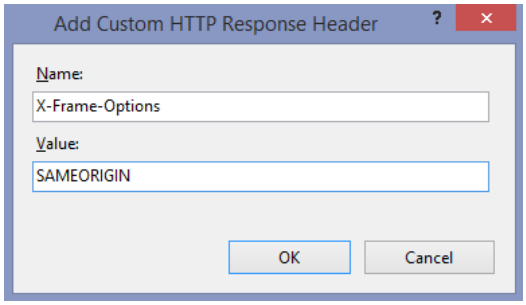
Select and open the "HTTP Response Headers" feature :

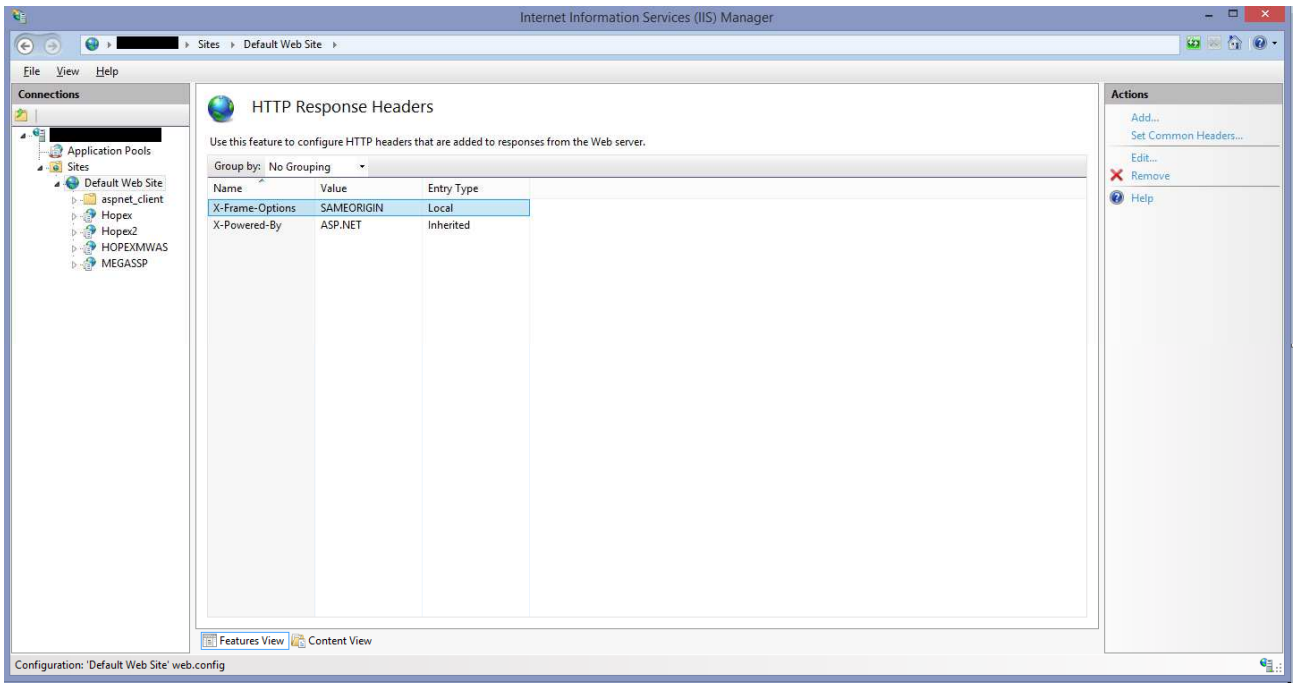


Click on « Add » in the Actions panel :



In the dialog box, type "X-Frame-Options" for the *Name* field, and "SAMEORIGIN" for the *Value* field, and validate :





Securing the ASP.NET session cookies

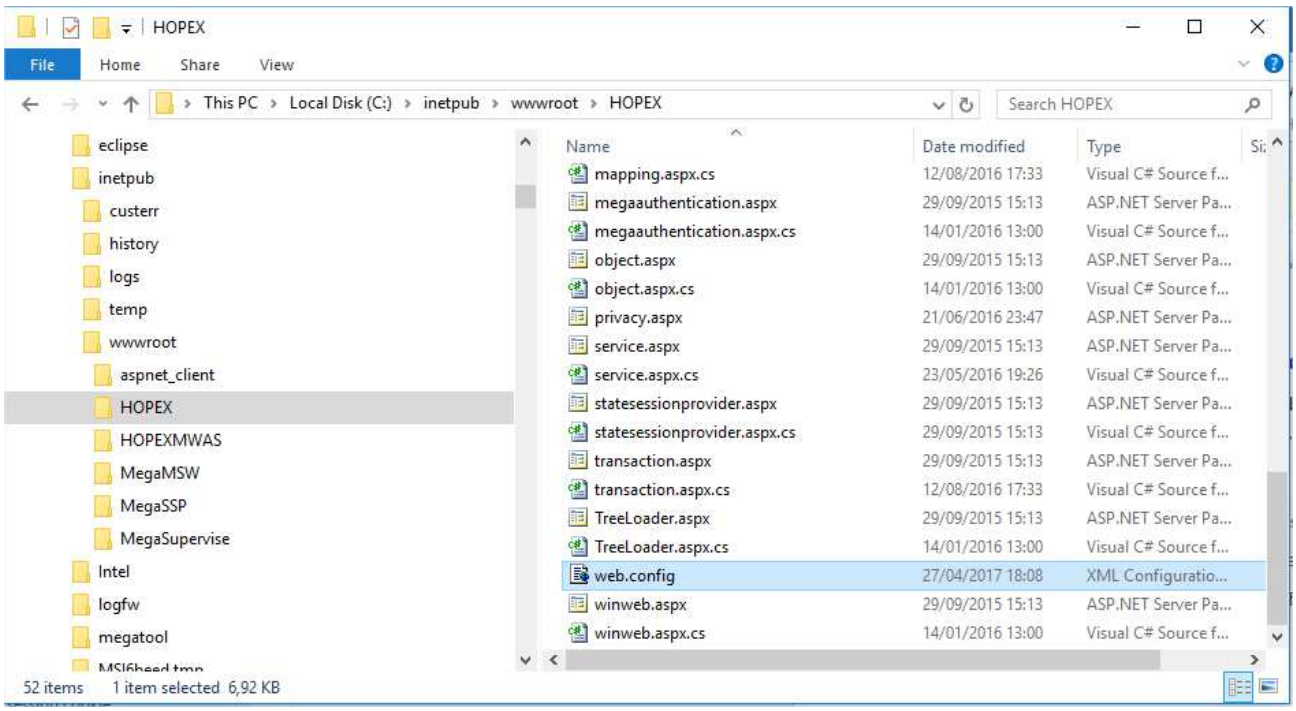
Prerequisite: requires the use of SSL on your website (see previous section in this document).

This feature is delivered by default with Hopex V1R2/R3 CP15, or Hopex V2 CP03.

For previous versions, a specific line will need to be added manually.

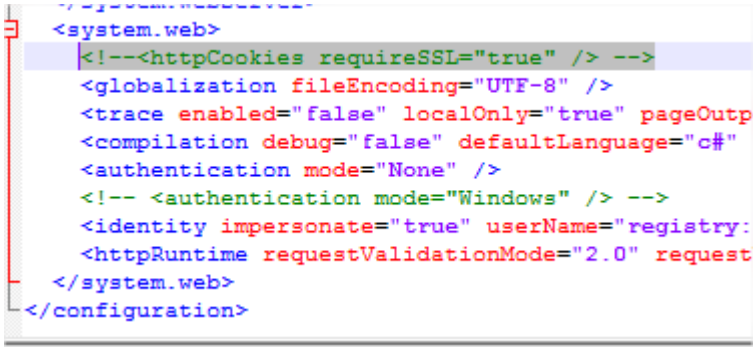
The goal is to secure the cookies of the users sessions.

1. To do that, your website requires to be configured in HTTPS.
2. Then, you need to update your web.config file of the "HOPEX" web application:



Browse through the file and go to the <system.web> section.

If you are using a version as shown at the beginning of this section, you will see the commented line hightlighted below:



Remove the comments, and save the file. It will automatically restart the application pool. You can then check if your website is still working:

```
<system.web>
  <httpCookies requireSSL="true" />
  <globalization fileEncoding="UTF-8" />
  <trace enabled="false" localOnly="true" pageOutput="true" tr
  <compilation debug="false" defaultLanguage="c#" targetFramew
  <authentication mode="None" />
  <!-- <authentication mode="Windows" /> -->
  <identity impersonate="true" userName="registry:HKEY_LOCAL_M
  <httpRuntime requestValidationMode="2.0" requestValidationTy
</system.web>
</configuration>
```

If you have previous version of Hopex, you will need to insert this line, as shown above, and save your file :

<httpCookies requireSSL="true" />

To prevent CSRF inside your application, define your cookies with the SameSite attribute set to the "Strict" value.

To do that, navigate to the <system.webServer> section of the web.config and add the following section:

```
<rewrite>
  <outboundRules>
    <rule name="Add SameSite" preCondition="No SameSite">
      <match serverVariable="RESPONSE_Set_Cookie" pattern=".*" negate="false" />
      <action type="Rewrite" value="{R:0}; SameSite=strict" />
      <conditions>
      </conditions>
    </rule>
    <preConditions>
      <preCondition name="No SameSite">
        <add input="{RESPONSE_Set_Cookie}" pattern="." />
        <add input="{RESPONSE_Set_Cookie}" pattern="; SameSite=strict" negate="true" />
      </preCondition>
    </preConditions>
  </outboundRules>
</rewrite>
```

Save the file. It will automatically restart the application pool. You can then check if your website is still working.

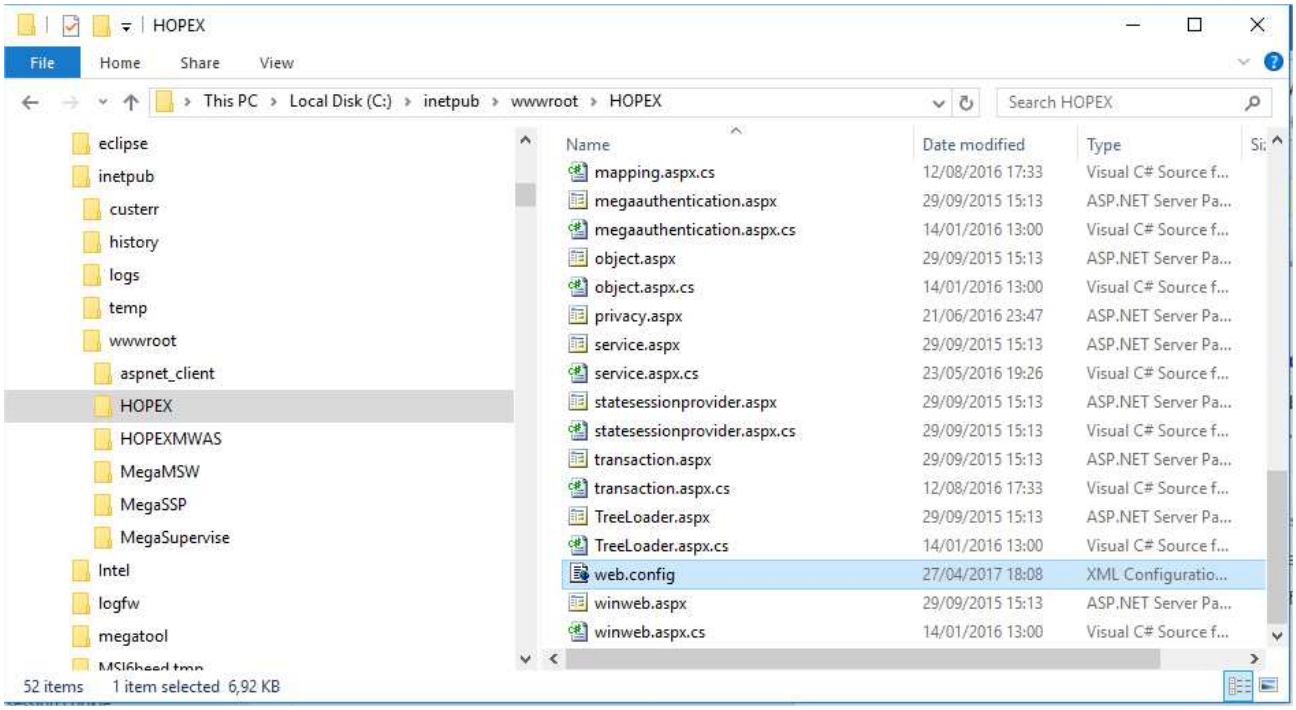
Hide ASP.Net version header

To prevent people from discovering some information about the platform and the application versions, a specific setting can be put in place to hide the ASP.Net headers.

This way, when checking the headers, people won't be able to detect that ASP.net is used, in which version. It also hides the build number of the application.

The steps to perform this setting are:

1. Update your "web.config" file of the "HOPEX" web application:



2. Browse through the file and go to the "<system.web>" section and locate the line where the "*httpRuntime*" is setup.
3. In that line, add this value:

   enableVersionHeader="false"

```
    <!-- <authentication mode="Windows" /> -->
    <identity impersonate="true" userName="registry:HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\MEGA\HOPEX\identity\ASPNET_SETREG,u
    <httpRuntime requestValidationMode="2.0" requestValidationType="MegaRequestValidator" enableVersionHeader="false"/>
  </system.web>
</configuration>
```

4. Save the file (this will restart the application pool).

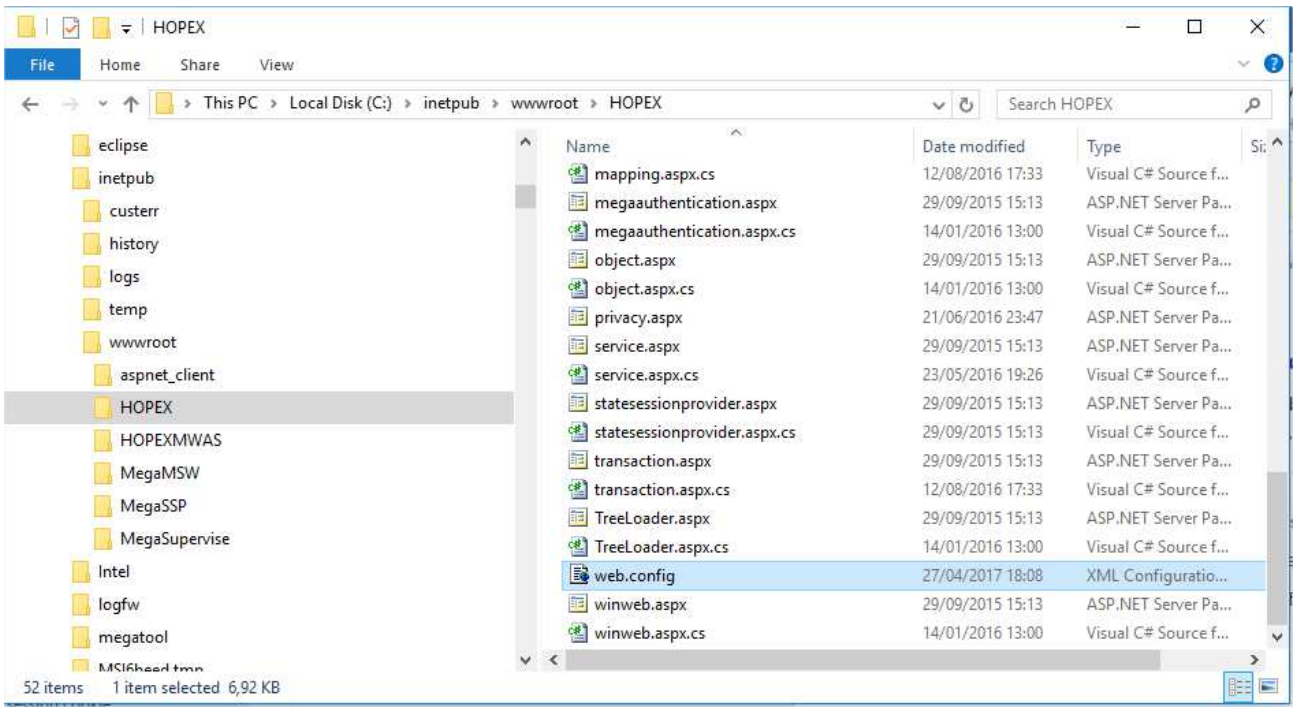
Remove IIS Server version HTTP Response Header

To prevent people from discovering some information about the platform and the IIS version, a specific setting can be put in place to hide the server headers.

This way, when checking the headers, people won't be able to detect IIS information.

The steps to perform this setting are:

1. Update your "web.config" file of the "HOPEX" web application:



2. Browse through the file and go to the "<system.webServer>" section
3. Add the following lines:

```
<security>

  <requestFiltering removeServerHeader="true" />

</security>
```

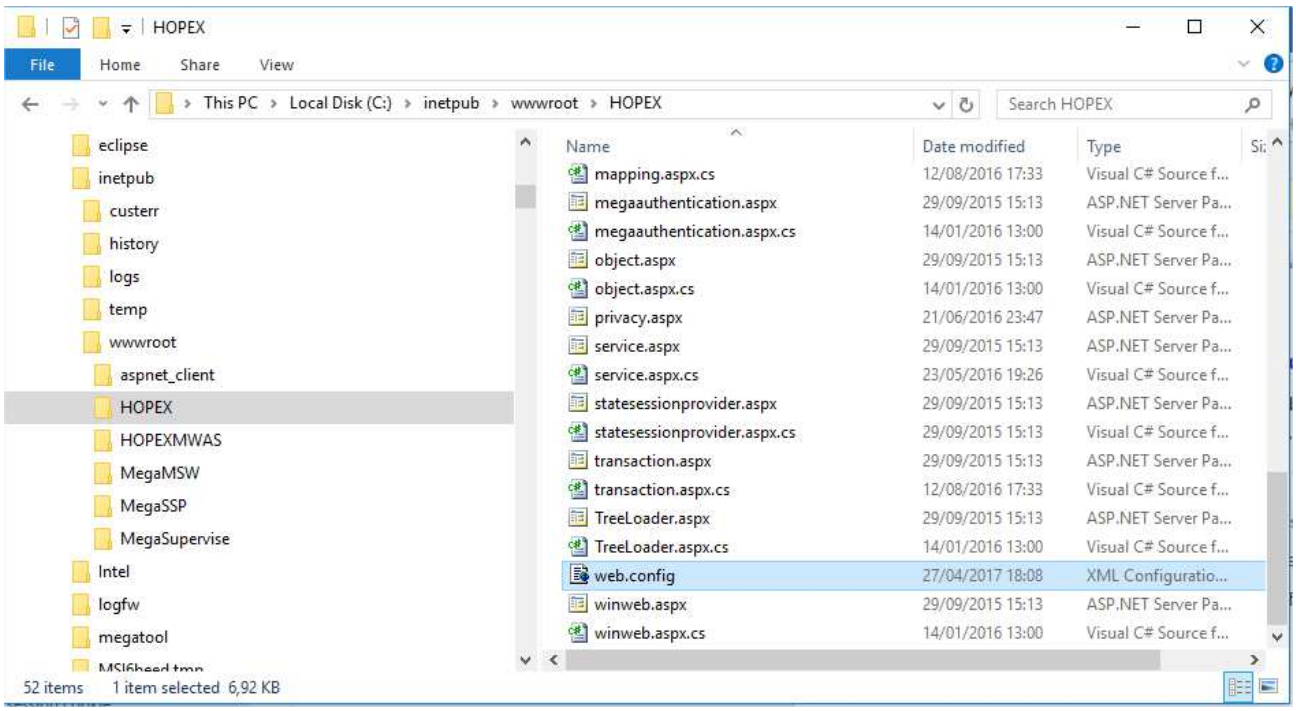
Strict-Transport-Security HTTP header

Prerequisite: only if SSL is activated on the web site.

If you are accessing your website through HTTPS, you may want to properly define the above header.

The steps to perform this setting are:

1. Update your "web.config" file of the "HOPEX" web application:



2. Browse through the file and go to the "<system.webServer>" section.
3. Add the following line in that section:

```
<httpProtocol>
    <customHeaders>
        <add name="Strict-Transport-Security" value="max-age=31536000"/>
    </customHeaders>
</httpProtocol>
```

The "max-age" value, in this example set up to **one year**, can be adapted depending on the internal policies.

```
<system.webServer>
  <httpProtocol>
    <customHeaders>
      <add name="Strict-Transport-Security" value="max-age=31536000"/>
    </customHeaders>
  </httpProtocol>
  <staticContent>
    <remove fileExtension=".woff" />
    <mimeMap fileExtension=".woff" mimeType="application/font-woff" />
```

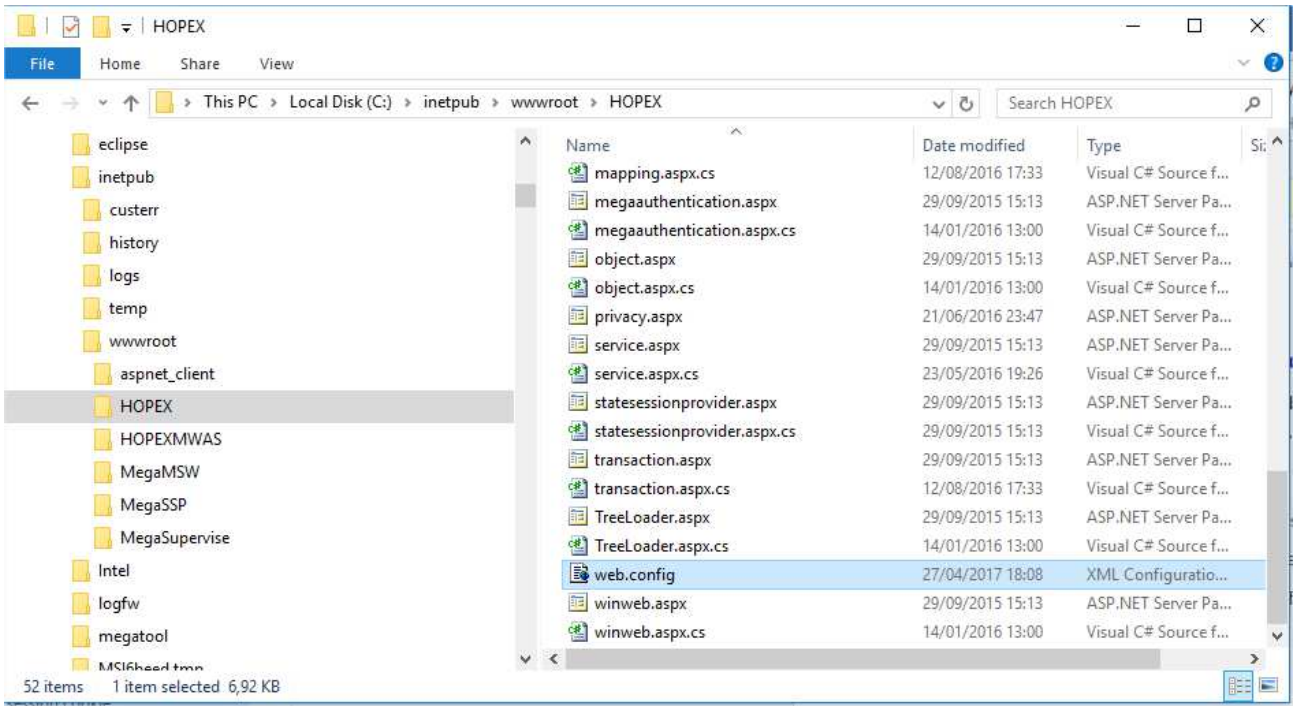
4. Save the file (this will restart the application pool).

Manage content type options

Depending on your version of the application, this section might already be activated by default. Review and check.

To prevent the user agent to render the site content in a different fashion to the MIME type:

1. Update your "web.config" file of the "HOPEX" web application:



2. Browse through the file and go to the "<httpprotocol>" section and its "<customheaders>" subsection.
3. Add the following line in that section:

```
<add name="X-XSS-Protection" value="1; mode=block"/>
<add name="X-Content-Type-Options" value="nosniff "/>
```

```
<customHeaders>
    <add name="Strict-Transport-Security" value="max-age=31536000"/>
    <add name="X-XSS-Protection" value="1; mode=block"/>
    <add name="X-Content-Type-Options" value="nosniff "/>
</customHeaders>
```

4. Save the file (this action will restart the application pool).

Restrict Cross Origin Ressource Sharing to Trusted Domains

UAS defines a default CORS policy to *.

It is important to reduce the policy to your company's trusted domains.

In UAS web.config, in the <system.webServer>\<httpProtocol>\<customHeaders> section, modify the "Access-Control-Allow-Origin" value and restrict the policy to the company's trusted domains.

```
<httpProtocol>
  <customHeaders>
    <add name="Access-Control-Allow-Origin" value="*" />
    <add name="Access-Control-Allow-Headers" value="Content-Type, Authorization, HopexContext" />
  </customHeaders>
</httpProtocol>
```

Search engine protection

HOPEX Application does not require to be indexed by an external search engine.

To prevent such action, create a robots.txt file in the application's webroot and write the following lines in the file content:

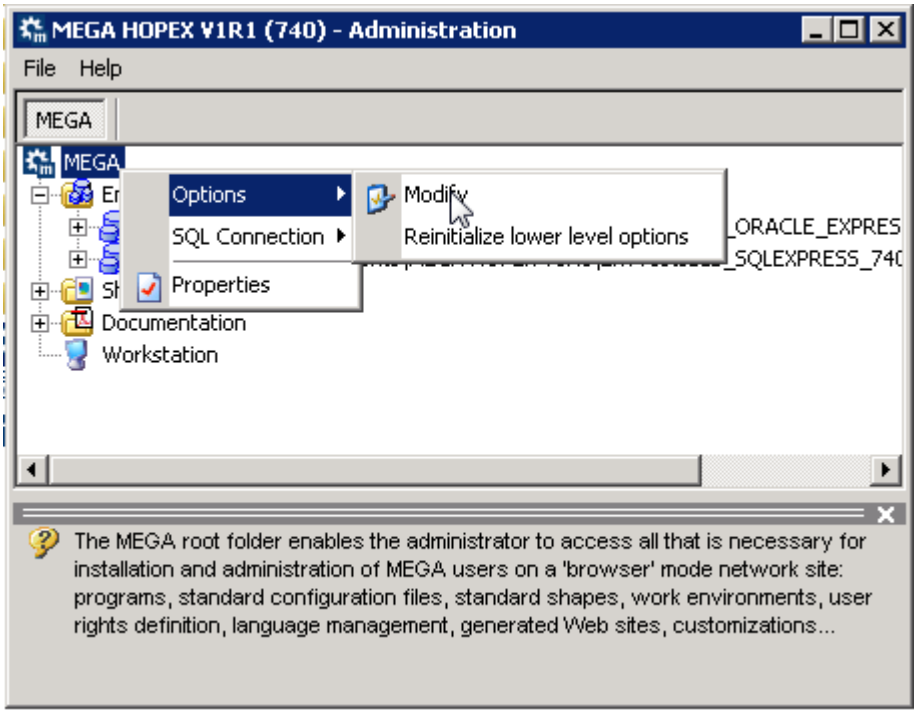
```
User-Agent: *
Disallow: /
```

Note that the following sections are normally already configured by default with HOPEX V1R2-V1R3 CP8, as well as with HOPEX V2 regardless the patch level. You can check those, and tune up the second section depending on the timeout you want to put in place.

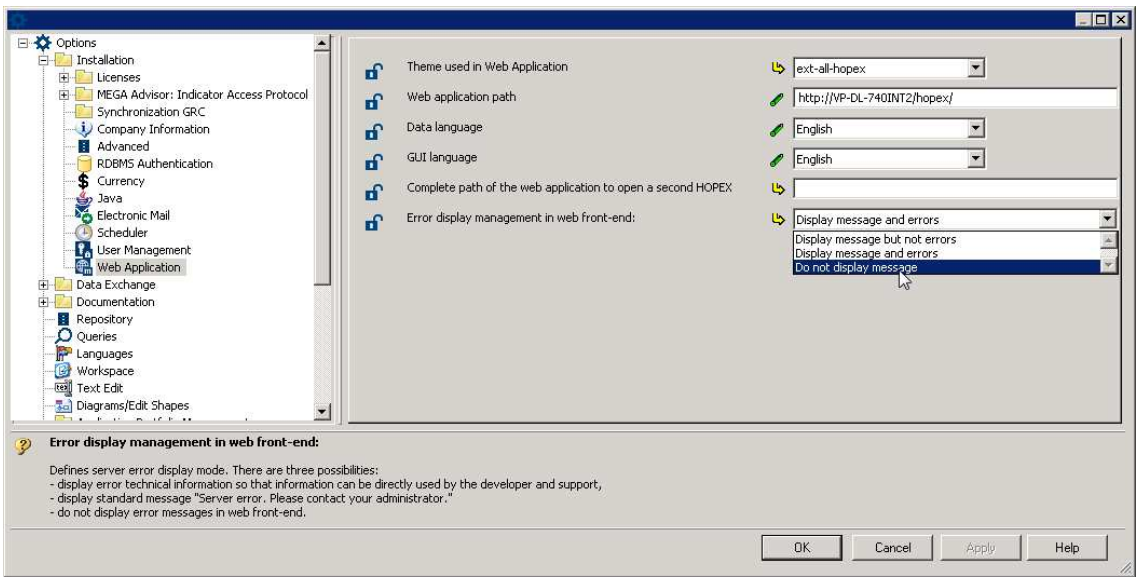
Hiding the error details

To prevent the end users from seeing the error details and get knowledge about how the application is written, some actions can be taken to hide those:

1. Open the Administration module of Mega on the web server (Administration.exe, in the installation module of Mega).
2. Open the options at the root level:



3. Go to “Installation”, and then “Web Application”, and change the option “Error display management in web front-end” to “Do not display message”:



4. Close the options and the Administration module.

5. Locate the web.config file of the “Hopex” web application (by default in “C:\inetpub\wwwroot\HOPEX”), and edit it.

6. Add the following key in the file :

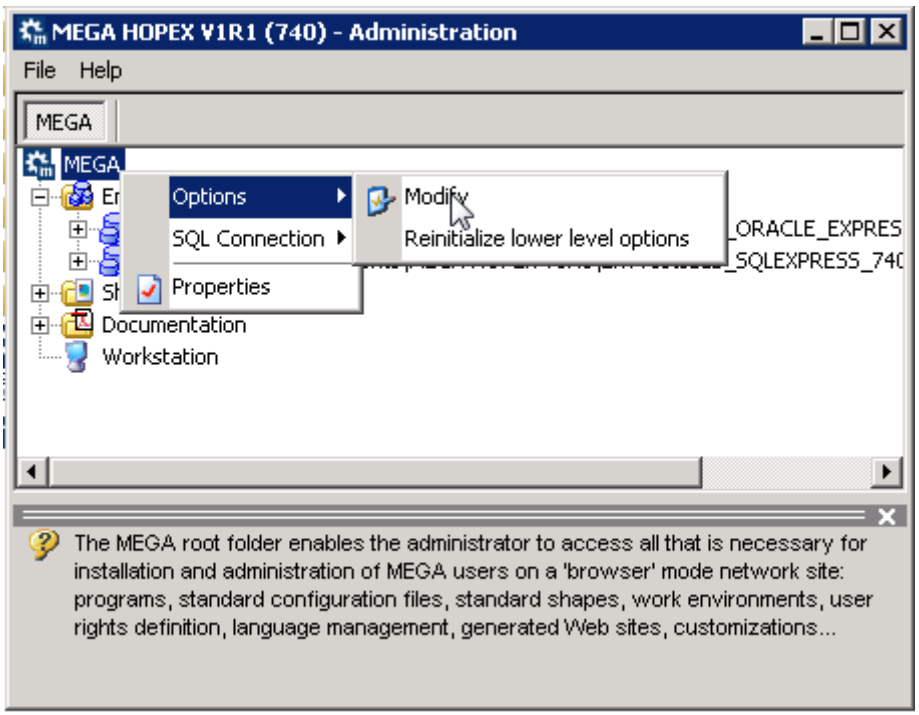
```
<add key="HideErrors" value="1"/>
```

Activating the automatic logoff

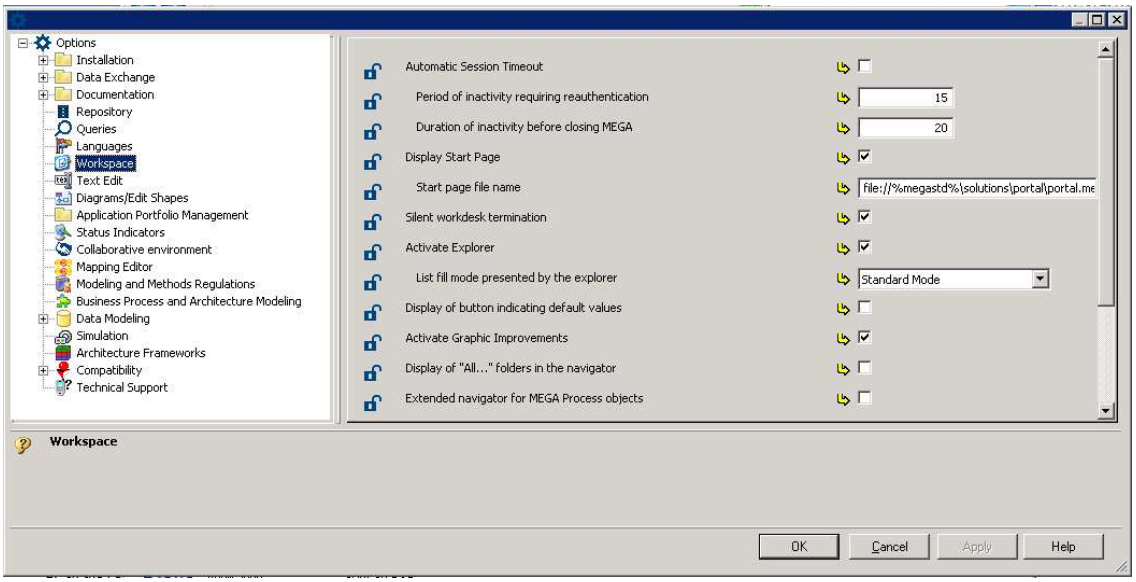
You can activate an automatic logoff of the users after a certain time of inactivity. To do so:

1. Open the Administration module of Mega on the web server (Administration.exe, in the installation module of Mega).

2. Open the options at the root level:

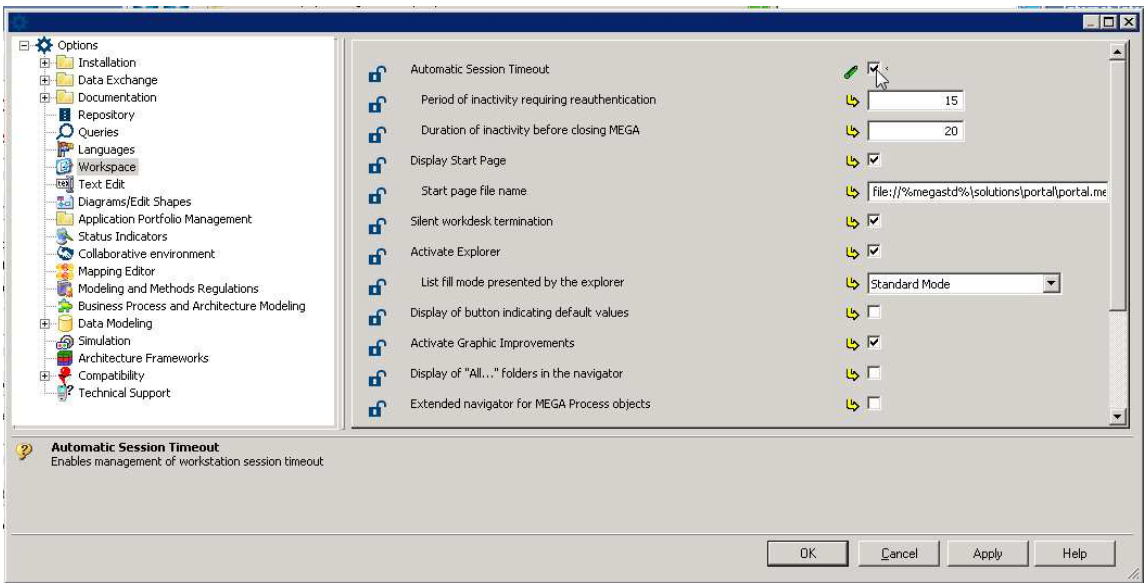


3. Go to "Workspace":



4. Select "Automatic Session Timeout" option, and tune up the parameters "Period of inactivity requiring authentication"and "Duration of inactivity before closing MEGA" to the wanted values (in minutes, by default they are set to 15 and 20, respectively) :

5.



6. Click **OK**, and close the Administration module.
7. Restart the application to validate this whole configuration.

Hiding the information when entering the wrong credentials

For versions, before V1R2 **CP15**, or V1R3 CP15, or V2 **CP03**, when someone enters a wrong password of an existing user, or tries to authenticate with a user that doesn't exist, he will get a clear message telling him what is the case.

If you want to have a generic message preventing someone from discovering the users declared in an environment, you will need to upgrade to the above-mentioned versions of the application.

The message will then be this one:

"No such login, or no security question defined for this login, or the configuration of your login does not allow you to reinitialize your password. Check the login you entered or contact your HOPEX Administrator."

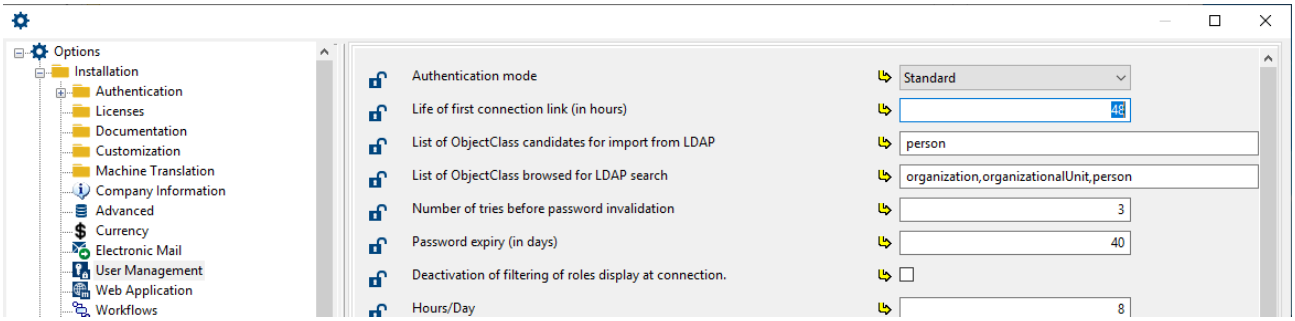
Manage password activities

If you use HOPEX authentication, the administrator has the ability to parameterize the following security rules from the environment options:

Account initialization

When the account is initialized, a mail is sent to the user enabling to create a new account.

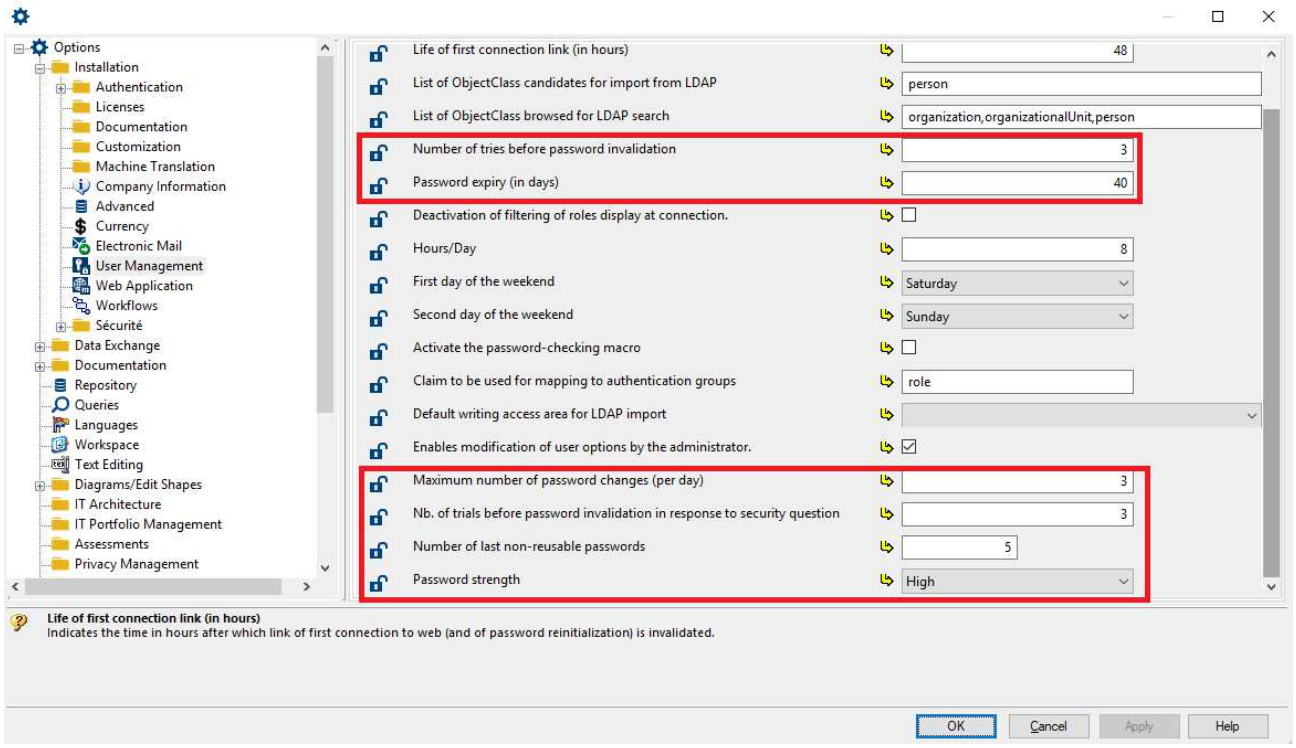
The link validity is by default set to 48 hours. If used once, the link is then obsolete.



HOPEX Password Management

The administrator has the ability to parameterize specific policies for password management:

- Set the number of passwords retries before the account is locked. The end user is notified to contact his administrator if the account is locked. By default, the value is set to 3.
- Set the validity period of a password. By default, the password expires after 40 days and the end user must change it.
- Set the maximum number of times the end user can change his password per day. The default value is set to 3.
- Set the number of security question retries before the account is locked. The end user is notified to contact his administrator if the account is locked. By default, the value is set to 3.
- Set the number of last non-reusable passwords. The default value is set to 5.
- Set the password strength. The default value is set to strong which means the end user must set a strong password when he updates his password.



Documents Upload

To prevent DOS issues, we give the ability to the administrator to limit the number of documents to be uploaded during a specific period of time.

To manage this, open Administration.exe, open your environment, navigate to installation\Security\Upload folder.

You will find 2 options:

- An option defining the number of documents the end user can upload in his session during a specific period of time.
- An option defining the period of time during which the upload of documents is limited.

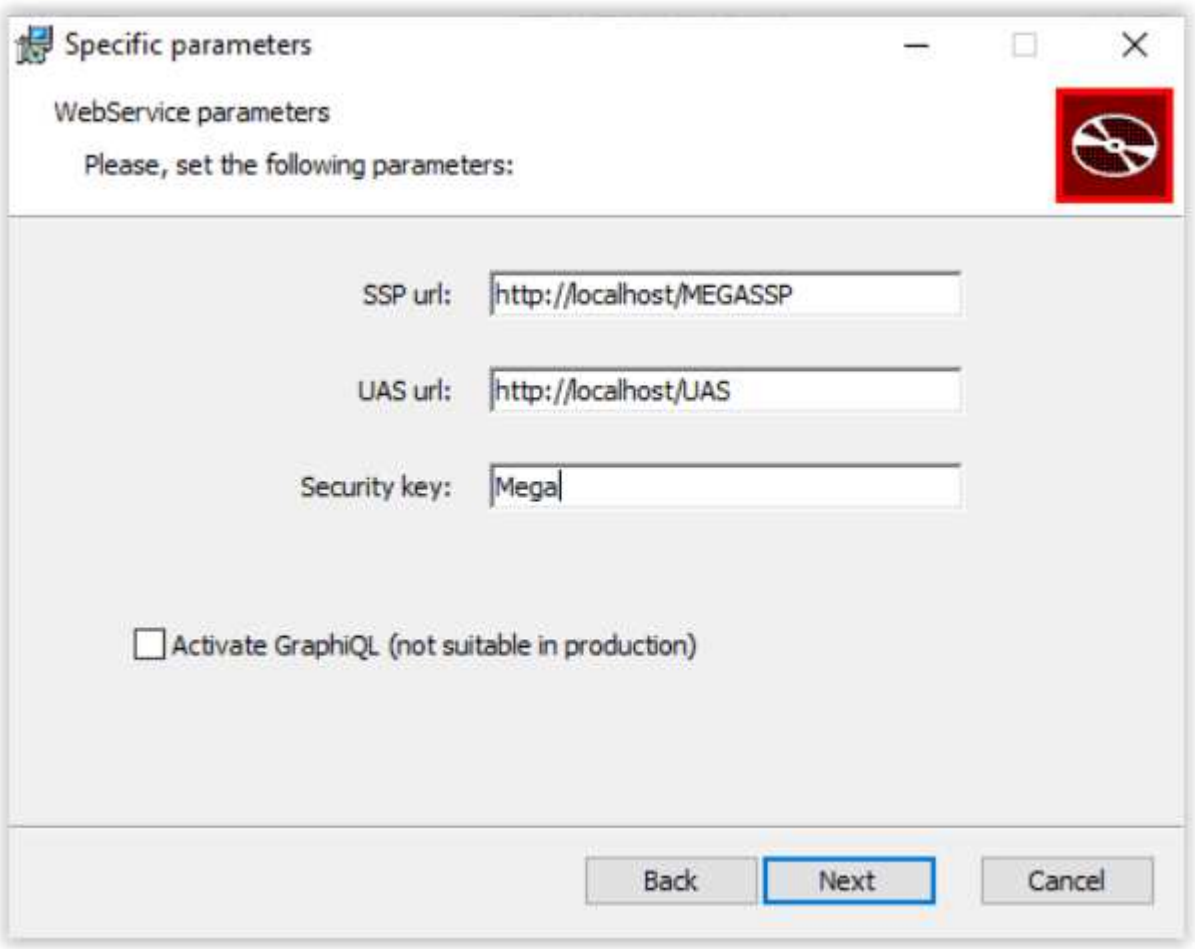
Modules

GraphQL/GraphiQL

If you install the GraphQL module, please ensure to disactivate GraphiQL.

During the installation process, you have the following screen.

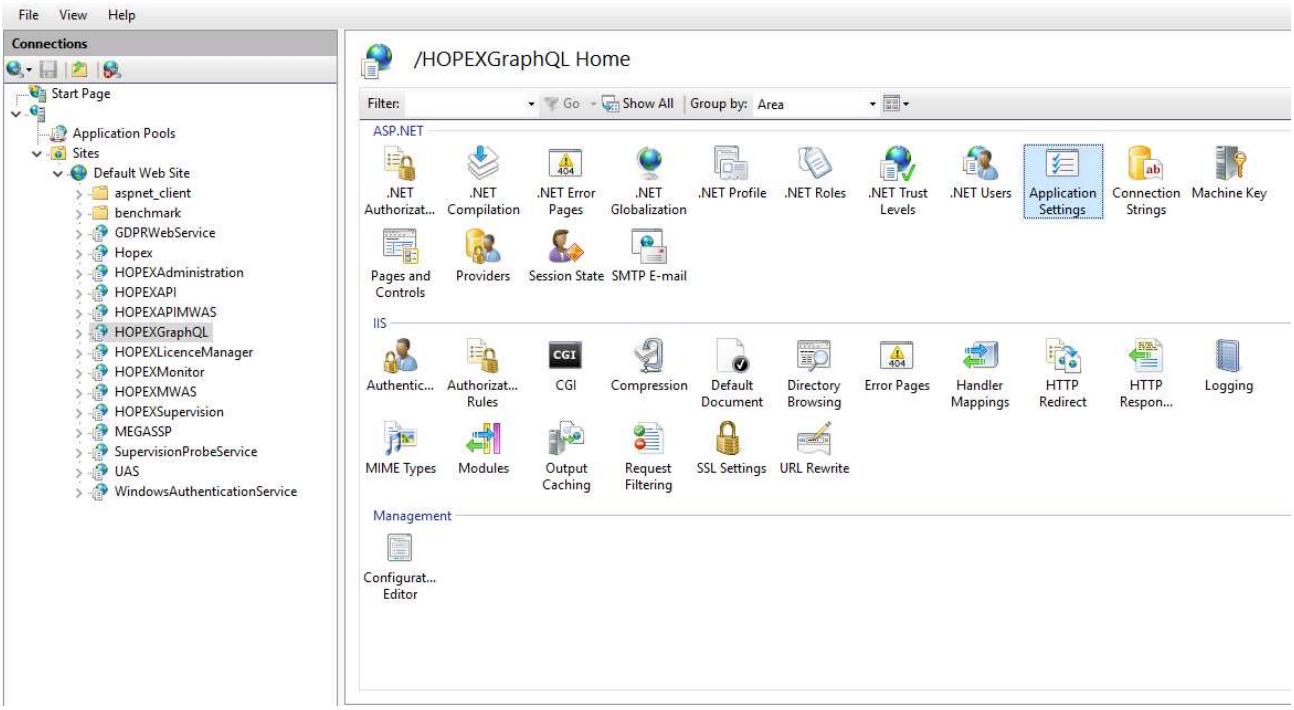
It is necessary to let the "Activate GraphiQL" unchecked.



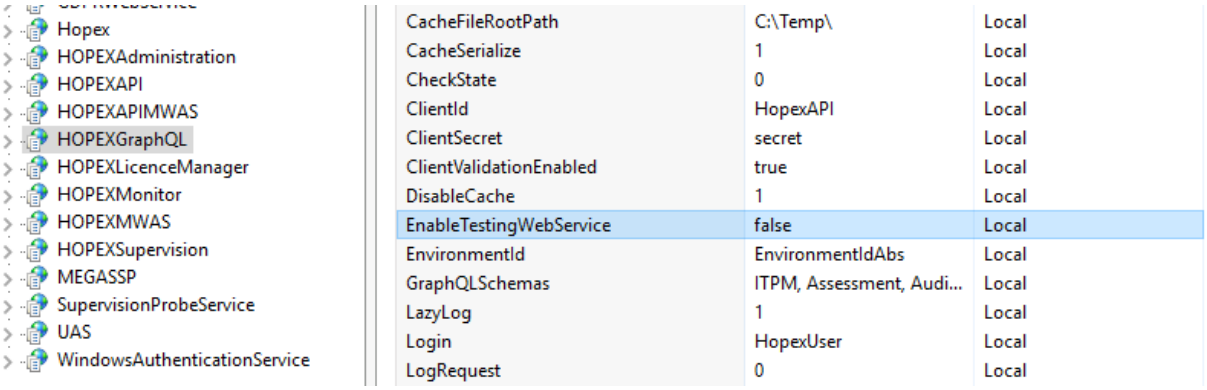
If GraphiQL is activated, the user and password defined in GraphiQL are visible in clear.

This can be used by a hacker to connect to HOPEX and retrieve data.

If GraphQL is already installed, please open IIS, navigate to HOPEXGraphQL Application and open the Application Settings.



Ensure "EnableTestingWebService" is set to false.



Description of MEGA Data Exchange XML Format

This technical article presents detailed explanation on various tags used in MEGA XML data exchange files.

INTRODUCTION TO MEGA XML DATA EXCHANGE FORMAT

MEGA allows you to import and export data in a standard XML exchange format.

Data contained in MEGA data XML documents can be described in the form of commands as for MGR, MGL or MGE documents. It can also be described in Content mode. Content mode now allows processing of sets of objects independently, free of any context (see Content exchange: <Content> tag, Content Mode, Hierarchical link in content mode).

Importing an XML document in a MEGA repository means executing or creating the commands it contains in this repository.

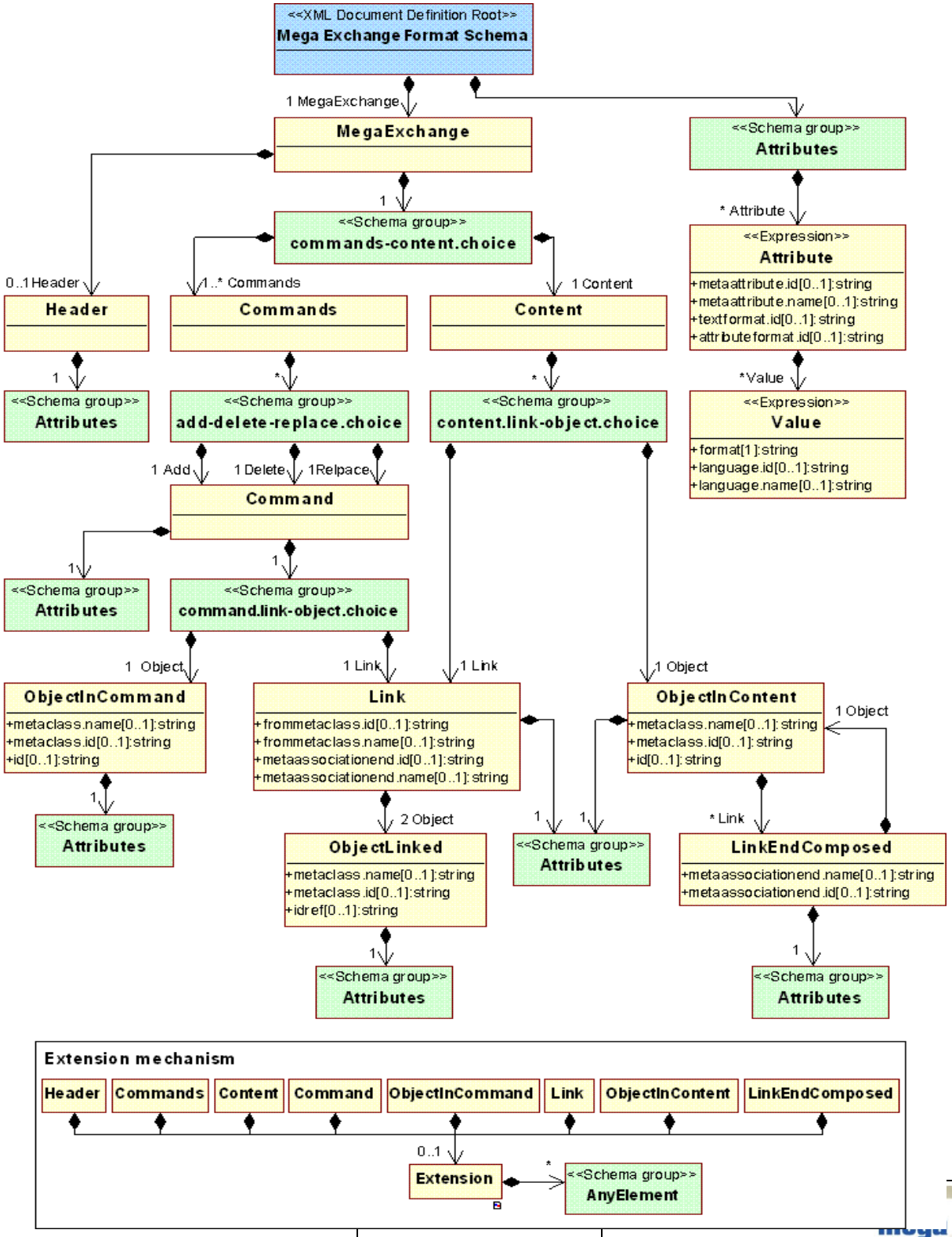
Exporting MEGA (objects, command) consists of dispatching this data in XML standard format. MEGA data exchange XML documents can also be created by software external to MEGA with a view to integrating the data they contain in a MEGA repository. Finally, they can be created manually.

You will find the XSD schema (“xmlmega.xsd”) of MEGA data exchange XML files in the MEGA installation "system" directory.

An XSLT style sheet is also provided in the MEGA installation "system" directory, which presents MEGA data exchange XML documents. This style sheet is provided only as an example.

Note: References to documentation concerning the following are given in the glossary: MEGA metamodel, XML language, importing a data file, exporting a data file.

FORMAT XML SCHEMA MODEL



Description

TAGS IN BRIEF

<MegaExchange>

This is the root element of the document. It contains all other tags.

(See Logical structure)

<Header>

This tag describes the characteristics of the document itself. We find here for example the exchange format version, the document creation date, the default exchange language, etc.

(See Logical structure)

<Attribute>

An <Attribute> tag enables expression of a command characteristic in an Add>, <Delete> or <Replace> tag, of an object in an <Object> tag, of a link in a <Link> tag, or of the exchange document itself in the <Header> tag.

(See Description of attribute values)

<Value>

This tag contains an attribute value. It appears only in <Attribute> tags. It serves to express a value in a particular format when this value can be expressed in several possible formats in an <Attribute> tag (for example, "internal" or "display" format).

(See Attribute value format)

<Commands>

This tag appears at root element level and enables expression of a set of commands.

(See Command exchange: <Commands> tag)

<Content>

This tag appears at root element level and enables inventory of a set of objects and links.

(See Content exchange: <Content> tag)

<Object>

An <Object> tag contains the description of an object: it is characterized by its type (MetaClass) and its attribute values. It can serve to describe or identify an object, for example an object to be connected at creation of a link.

(See Object Descriptions)

<Link>

A <Link> tag contains the description of a link: its type (MetaClass of the object to be connected and MetaAssociationEnd by which the second object is connected), identifications of the two objects to be connected and the link attribute values.

(See Link description)

<Add>

The <Add> tag is used in the <Commands> tag. It enables representation of an object or link creation command. Content is either an <Object> tag describing the object to be created, or a <Link> tag describing the link to be created.

(See Command exchange: <Commands> tag, Command Mode)

<Delete>

The <Delete> tag is used in the <Commands> tag. It enables representation of an object or link deletion command. Content is either an <Object> tag describing the object to be deleted, or a <Link> tag describing the link to be deleted.

(See Command exchange: <Commands> tag, Command Mode)

<Replace>

The <Replace> tag is used in the <Commands> tag. It enables representation of an object or link update command. Content is either an <Object> tag describing the object to be modified, or a <Link> tag describing the link to be modified.

(See Command exchange: <Commands> tag, Command Mode)

<Extension>

Various tags can contain the <Extension> tag.. When it is present in an element, this tag enables addition of supplementary information to the element (for example, addition of information concerning reasons for command reject). This information can be taken into account in a particular way according to tools used.

FORMAT IN DETAIL

In the remainder of this document, MEGA data exchange XML format will be referred to as "MEGA XML".

MEGA data exchange XML document structure

Physical structure

Like all XML documents, MEGA XML documents must start with XML declaration:

<?xml version="1.0"?>.

Encoding can be specified using the "encoding" attribute.

<?xml version="1.0" encoding="ISO-8859-1"?>

MEGA XML documents should be expressed in a code supported by the applications used to process them. For example, before importing a MEGA XML document it should be confirmed that MEGA can handle documents in the encoding proposed. MEGA import, like any XML analyzer, can basically handle input of Unicode encodings: UTF-8, UTF-16 little endian and big endian and ASCII. Other encodings such as ISO Latin1 FR can be used.

Note: see XML specifications for values to be specified for the "encoding" attribute (ref. Extensible Markup Language (XML) 1.0: http://www.w3.org/TR/2004/REC-xml-20040204/), and MEGA documentation to determine supported encodings.

Note: values that can be specified for the "encoding" attribute to identify the different encodings:

- "ISO-8859-1" corresponds to ISO Latin-1 FR encoding
- "UTF-8" corresponds to Unicode encoding on one or several bytes per character
- "UTF-16" corresponds to Unicode encoding on a multiple of two bytes per character

Logical structure

Like all XML documents, MEGA XML documents can have only one root tag, its name being *<MegaExchange>*.

A MEGA XML document firstly contains information relating to the document itself, such as the document creation date, the format version used or the attribute value expression language. This information is described in a *<Header>* tag.

In addition, information exchanged must be expressed either in a *<Commands>* tag, or in a *<Content>* tag

Example : MEGA data exchange XML document structure

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="format_version">Mega Xml Format Version
0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged data -->
    <Add>
      <Object metaclass.name="Procedure" id="1">
        <Attribute metaattribute.name="Name">Procedure-1</Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

Exchanged data description modes

Data exchanged via MEGA XML documents can be described in two modes:

- Either as a series of commands to be processed one after the other.
- Or as a repository content or sub-content, in other words as a set of objects.

Command exchange: <Commands> tag

Command series data expression mode is by use of the *<Commands>* tag.

It is this tag that contains command description tags in XML. The order of command description tags within the *<Commands>* tag is significant. It corresponds to the order in which commands must be processed by the

MEGA XML document analysis tools. In fact, a command may not be validated unless preceding commands have been processed.

For example, if a command for creation of a link between two objects appears before the command for creation of one (or both) of the objects themselves, the link creation command is not valid from a logical viewpoint.

Example : Command exchange

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Procedure" id="1">
        <Attribute metaattribute.name="Name">Procedure-1</Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

Content exchange: <Content> tag

Content description mode uses the *<Content>* tag. This tag contains a collection of objects and links.

These objects and links should be interpreted as independent data free of any context: they are not connected to a particular repository and do not require any other data in order to be significant (except for the metamodel describing them).

Data of an XML exchange document using description mode produces creation commands (creation of described objects and links) when the document is imported.

Order of appearance of tags describing data contained in the *<Content>* tag is significant. It corresponds to the order in which data must be processed by the MEGA XML document analysis tools.

If a link between two objects appears before one (or both) of the objects themselves, the link description is not valid from a logical viewpoint.

Example : Content description mode data exchange

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Procedure" id="1">
      <Attribute metaattribute.name="Name">Procedure-1</Attribute>
    </Object>
  </Content>
</MegaExchange>
```

In content description mode, we can explicitly show the hierarchical view of exchanged data structure. In fact, tags describing objects can themselves contain other tags describing contained objects from a logical viewpoint (for example a procedure containing operations).

This hierarchical description is valid only in content description mode.

Example : <Object> tag containing an <Object> tag

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Procedure" id="1">
      <Attribute metaattribute.name="Name">Procedure-1</Attribute>
      <!-- hierarchical link use  -->
      <Link metaassociationend.name="Contained operation">
        <!-- contained object -->
```

```
        <Object metaclass.name="Operation" id="2">
          <Attribute metaattribute.name="Name">Procedure-1</Attribute>
        </Object>
      </Link>
    </Object>
  </Content>
</MegaExchange>
```

Commands description: <Add>, <Delete>, <Replace> tags

Command tags can be used only in command mode. These tags are contained in the *<Commands>* tag explained in the chapter Command exchange: <Commands> tag.

The three available commands are:

- Creation command represented by an *<Add> tag.*
- Deletion command represented by a *<Delete> tag.*
- Modification command represented by a *<Replace> tag.*

Each of these three commands can be applied to an object or to a link: a tag representing a command can contain one tag only: *<Object>* or *<Link>*.

In command mode, *<Object>* tags cannot directly or indirectly contain other *<Object> tags.* : The hierarchical aspect of exchanged data logical structure cannot be represented by the hierarchical aspect of XML when data is exchanged in command mode.

Example : Object and link creation commands

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Class">
        <Attribute metaattribute.name="UML ClassName (Français)">Ma classe</Attribute>
```

```
      </Object>
    </Add>
    <Add>
      <Link frommetaclass.name="Package" metaassociationend.name="Contained class ">
        <Object metaclass.name="Package">
          <Attribute metaattribute.name="Name">My package</Attribute>
        </Object>
        <Object metaclass.name="Class">
          <Attribute metaattribute.name="Name">My org-unit</Attribute>
        </Object>
        <Attribute metaattribute.name="Order">9999</Attribute>
      </Link>
    </Add>
  </Commands>
</MegaExchange>
```

Object Descriptions

Objects are described by the *<Object> tag*..

The MetaClass of the object is identified by the name or MEGA absolute identifier (idabs in hexadecimal) of the MetaClass. The name of the MetaClass is specified by the value of the *"metaclass.name"* attribute of the *<Object> tag*, the idabs of the MetaClass is specified by the value of the *"metaclass.id"* attribute of the *<Object> tag*.

In addition, the objects themselves must be identified when we wish to make reference to them in the exchange document (see Object identification mechanisms).

Example : Object metaclass identification by metaclass name

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Procedure">
        <Attribute metaattribute.name="Name">Procedure-1</Attribute>
```

```
    </Object>
  </Content>
</MegaExchange>
```

Example : Object metaclass identification by metaclass idabs

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.id="B1EDB25E2C1401BB">
      <Attribute metaattribute.name="Name">Procedure-1</Attribute>
    </Object>
  </Content>
</MegaExchange>
```

Command Mode

In commands description mode, an object is described in an *<Add>*, *<Delete>* or *<Replace>* tag, depending on whether we wish to create, delete or modify the object.

In this case, the *<Object>* tag describing the object can if necessary contain an *<Extension>* tag containing information not allowed for by MEGA XML format. It also contains *<Attribute>* tags specifying attribute values characterizing this object, these tags following the *<Extension>* tag if this is present.

Example : Object creation command by attribute specification

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
```

```
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Procedure" id="1">
        <Attribute metaattribute.name="Name">Procedure-1</Attribute>
        <Attribute metaattribute.name="Type-Procedure">General</Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

An object described in a deletion command must make reference to an existing object recognized by the tool analyzing the MEGA XML document. The object description must therefore specify the value of its MEGA absolute identifier (object "_idabs" attribute) in an *<Attribute> tag*.

Example : Object deletion command

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Delete>
      <Object metaclass.name="Procedure">
        <!-- object identifier attribute -->
        <Attribute metaattribute.name="_IdAbs">MnJgyaAJ0100</Attribute>
      </Object>
    </Delete>
  </Commands>
</MegaExchange>
```

An object described in a modification command must make reference to an existing object recognized by the tool analyzing the MEGA XML document. The object description must therefore specify the value of its MEGA absolute identifier (object "_idabs" attribute) in an *<Attribute> tag*. In addition to the *<Attribute>* tag specifying

the MEGA absolute identifier value, we find *<Attribute>* tags giving the values of attributes to be changed for the object concerned.

Example : Object modification command

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Replace>
      <Object metaclass.name="Procedure">
        <!-- object identifier attribute -->
        <Attribute metaattribute.name="_IdAbs">MnJgyaAJ0100</Attribute>
        <!-- modified attribute -->
        <Attribute metaattribute.name="Code-Procedure">XYZ</Attribute>
      </Object>
    </Delete>
  </Commands>
</MegaExchange>
```

Content Mode

In content description mode, *<Object>* tags describing objects are contained in the *<Content> tag*.

The *<Object>* tag describing the object can if necessary contain an *<Extension>* tag containing information not allowed for by MEGA XML format. It also contains *<Attribute>* tags specifying attribute values characterizing this object. These tags follow the *<Extension>* tag if this is present.

In addition, in content description mode, objects can be described as containing other objects. The *<Object>* tag can therefore indirectly contain other *<Object>* tags describing contained objects (see Hierarchical link in content mode).

The same considerations apply to *<Object>* tags representing contained objects, which can themselves contain *<Object> tags*.

Example: Object description in content mode

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Procedure" id="1">
      <Attribute metaattribute.name="Name">Procedure-1</Attribute>
    </Object>
  </Content>
</MegaExchange>
```

Object identification mechanisms

We need to make reference to objects internal or external to the document within the framework of:

- An object modification command
- An object modification command
- The link between one object and another

We distinguish objects described in the document from those not described in the document but which are the object of a command:

- Objects internal to the document: these are described in the document and can be referenced in the document via use of *"id"* and *"idref"* attributes of the *<Object> tag*.
- Objects external to the document: these are not described in the document but can be the target of a command. These should be recognized by the tool processing the document and are identified by the "_idabs" attribute of the object.

Object identification therefore uses either an identifier internal to the document or a MEGA absolute identifier.

Identification internal to the document is by use of the *"id"* attribute of the *<Object>* tag. The value of the *"id"* attribute must be unique throughout the document with no other constraint on form; it is an ID type attribute (ID type is defined in XML1.0 specifications). Reference to a document object is by using the *"idref"* attribute of the *<Object>* tag, which must therefore have the same value as the *"id"* attribute of the *<Object>* tag representing the referenced object. For example, within the framework of a link, the *<Object>* tags referencing linked objects can each have an *"idref"* attribute.

External identification is by definition of an *<Attribute>* tag defining the value of its MEGA absolute identifier (object "_idabs" attribute). Objects can be created with an "_idabs" attribute by use of the *<Attribute>* tag (see Description of attribute values). Similarly, *<Object>* tags referencing objects, as in the *<Delete>*, *<Replace>* or *<Link>* tags, can identify objects by their MEGA absolute identifier, and in this case they contain an *<Attribute>* tag specifying the value of the object "_idabs" attribute.

Example : Identification by <Object> tag id attribute

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Procedure" id="1">
      <Attribute metaattribute.name="Name">Procedure-1</Attribute>
    </Object>
    <Object metaclass.name="Procedure" id="2">
      <Attribute metaattribute.name="Name">Procedure-2</Attribute>
    </Object>
    <Link frommetaclass.name="Procedure" metassociationend.name="Next procedure">
      <Object metaclass.name="Procedure" idref="1"/>
      <Object metaclass.name="Procedure" idref="2"/>
    </Link>
  </Content>
</MegaExchange>
```

Example : Identification by idabs

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Procedure">
        <Attribute metaattribute.name="Name">Procedure-1</Attribute>
        <!-- object identifier attribute -->
        <Attribute metaattribute.name="_IdAbs">MnJgyaAJ0100</Attribute>
      </Object>
    </Add>
    <Delete>
      <Object metaclass.name="Procedure">
        <!-- object identifier attribute -->
        <Attribute metaattribute.name="_IdAbs">MnJgyaAJ0100</Attribute>
      </Object>
    </Delete>
  </Commands>
</MegaExchange>
```

Link description

Links are described by the *<Link> tag.*.

The MetaAssociation of the link is identified by the name or MEGA absolute identifier (idabs in hexadecimal form) of the MetaAssociationEnd by which the objects are connected.

The name of the MetaAssociationEnd is specified by the value of the *"metaassociationend.name"* attribute of the *<Link>* tag.

The idabs of the MetaAssociationEnd is specified by the value of the *"metaassociationend.id"* attribute of the *<Link>* tag.

One of the attributes *"metaassociationend.name"* or *"metaassociationend.id"* must be present to identify the link. Both can be present simultaneously.

Hierarchical link in content mode

In content mode, a link can be represented hierarchically. The *<Object>* tag representing the first object therefore contains a *<Link>* tag which itself contains the *<Object>* tag describing the connected object.

The *<Link>* tag serves to specify the MetaAssociationEnd by which the second object is connected. To do this, we must therefore specify the *"metaassociationend.name"* attribute or the *"metaassociationend.id"* attribute.

In addition, the *<Link>* tag can contain *<Attribute>* tags describing link attribute values. These *<Attribute>* tags are placed before the *<Object>* tag.

Example: Hierarchical ink description in content mode

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Procedure" id="1">
      <Attribute metaattribute.name="Name">Procedure-1</Attribute>
      <!-- hierarchical link use  -->
      <Link metaassociationend.name="Contained operation">
        <!-- link attributes -->
        <Attribute metaattribute.name="Order">1</Attribute>
        <!-- contained object -->
        <Object metaclass.name="Operation" id="2">
          <Attribute metaattribute.name="Name">Procedure-1</Attribute>
        </Object>
      </Link>
    </Object>
  </Content>
</MegaExchange>
```

Other links

Links that are not hierarchical links can be used in both command mode command tags (<*Add*>, <*Delete*>, <*Replace*>) and in the content mode <*Content*> tag. These links make reference to two connected objects describing but not referencing these objects. The connected objects can be external to the document.

A link between two objects is described by the MetaAssociationEnd by which the second object is connected to the first. To do this, the <*Link*> tag representing the link contains either the MetaAssociationEnd idabs value in hexadecimal form in the *"metaassociationend.id"* attribute, or the MetaAssociationEnd name value in the *"metaassociationend.name"* attribute. In the latter case, the MetaClass of origin of the MetaAssociationEnd must be specified, ie. the MetaClass of the first object. This is done via the *"frommetaclass.id"* or *"frommetaclass.name"* attributes of the <*Link*> tag, specifying either the MetaClass idabs in hexadecimal form or the MetaClass name.

In addition, the <*Link*> tag contains two <*Object*> tags that reference the two connected objects. Each of these has a *"metaclass.id"* attribute specifying either the MetaClass idabs in hexadecimal form, or a *"metaclass.name"* attribute specifying its name. The object is referenced either by specifying the *"idref"* attribute value of the <*Object*> tag in the <*Link*> tag, which should be equal to the *"id"* attribute value of the <*Object*> tag describing the object referenced in the document, or by adding an <*Attribute*> tag specifying the "_idabs" attribute value of the referenced object, which in this case can be an object not described in the document.

Finally, the <*Link*> tag can contain <*Attribute*> tags describing link attribute values. These <*Attribute*> tags are placed after the two <*Object*> tags.

Example : Description of a link between two objects of the document: use of “idref” attribute

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Content>
    <!-- exchanged data -->
    <Object metaclass.name="Package" id="1">
      <Attribute metaattribute.name="Name">Package-1</Attribute>
    </Object>
    <Object metaclass.name="Package" id="2">
      <Attribute metaattribute.name="Name">Package-2</Attribute>
    </Object>
```

```
    <Link frommetaclass.name="Package" metaassociationend.name="Referenced package">
      <!-- link source object -->
      <Object metaclass.name="Package" idref="1"/>
      <!-- link destination object -->
      <Object metaclass.name="Package" idref="2"/>
      <!-- link attributes -->
      <Attribute metaattribute.name="Order">9999</Attribute>
    </Link>
  </Content>
</MegaExchange>
```

Example : Description of a link between two objects by idabs

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Link frommetaclass.name="Package" metaassociationend.name="Referenced package">
        <!-- link source object -->
        <Object metaclass.name="Package">
          <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        </Object>
        <!-- link destination object -->
        <Object metaclass.name="Package">
          <Attribute metaattribute.name="_idabs">dCyP0ZtmxSA8</Attribute>
        </Object>
        <!-- link attributes -->
        <Attribute metaattribute.name="Order">9999</Attribute>
      </Link>
    </Add>
  </Commands>
</MegaExchange>
```

Description of attribute values

Attribute values are specified in *<Attribute>* tags. The following tags can contain attribute values: *<Header>*, *<Link>*, *<Object>*, *<Add>*, *<Delete>*, *<Replace>*.

In an *<Attribute>* tag, the valuated characteristic is identified by the *"metaattribute.name"* attribute, which determines its name, or by the *"metaattribute.id"* attribute, which determines its idabs in hexadecimal form.

The value is directly contained in the *<Attribute>* tag..

Example : Description of an attribute value

```
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Package">
        <!-- object attributes -->
        <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        <Attribute metaattribute.name="nom">Packagee-1</Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

Attribute value format

Direct values in *<Attribute>* tags are expressed in the format specified by the *<Attribute>* tag representing the *"default_format"* attribute of the *<Header>* tag.

If this attribute is not specified, default format is left to the tool analyzing the document.

Example : Attribute value default format

```
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
    <Attribute metaattribute.name="default_format">internal</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Package">
        <!-- object attributes -->
        <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        <Attribute metaattribute.name="nom">Packagee-1</Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

The *<Attribute>* tag enables attribute value specification in several formats. The different forms that the attribute value can take are each contained in a *<Value>* tag, the format being specified by the *<Value>* tag *"format"* attribute.

Example : Attribute value expressed in several formats

```
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="xmg_version">XMG version 0.1</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Application">
        <!-- object attributes -->
        <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        <Attribute metaattribute.name="name">Application-1</Attribute>
        <Attribute metaattribute.name="MMI">
          <Value format="internal">W</Value>
          <Value format="display">Windowed</Value>
        </Attribute>
```

```
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

Translatable attributes and attribute value expression language

In the MEGA repository, for a given translatable attribute, there are as many attributes as there are languages into which the attribute is translatable. For example, for the "Name" attribute there are corresponding "Name (English)","Name (French)", "Name (Dutch)", etc. attributes.

In a MEGA XML document, the *"language.id"* and "language.name" attributes of the *<Value>* tag enable specification of the language in which the attribute value is expressed.

Translatable attributes are therefore identified by the name or absolute identifier of the root attribute using the *"metaattribute.name"* and *"metaattribute.id"* attributes of the *<Attribute>* tag, and the language in which the value is expressed using the *"language.name"* and *"language.id"* attributes.

For value expression language to be specified, at least one of the two attributes *"language.name"* or *"language.id"* must be present in the *<Value>* tag.

When neither the *"language.name"* nor the *"language.id"* attribute is present, the value is expressed in the language specified by the *<Attribute>* tag representing the *"model_default_language"* attribute of the *<Header>* tag.

Example : Attribute value expressed in default language

```
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="model_default_language"> Français</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Application">
        <!-- object attributes -->
        <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        <Attribute metaattribute.name="name">Application-1</Attribute>
      </Object>
    </Add>
```

```
  </Commands>
</MegaExchange>
```

Example : Attribute value expressed in several languages

```
<MegaExchange>
  <Header>
    <!-- document information -->
    <Attribute metaattribute.name="model_default_language"> Français</Attribute>
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Application">
        <!-- object attributes -->
        <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        <Attribute metaattribute.name="name">
          <Value>Invoicing</Value>
          <Value language.name="English">Invoicing</Value>
        </Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

MEGA XML format extensions

MEGA XML format includes an extension mechanism enabling addition of information to different format tags.

A format extension is created by use of the *<Extension>* tag. The extension tag can contain any information expressed in XML.

Information contained in the *<Extension>* tag is not part of the format. It cannot be taken into account by a tool using MEGA XML format.

The following tags can contain an extension tag: *<Header>*, *<Link>*, *<Object>*, *<Add>*, *<Delete>*, *<Replace>*.

Example of use of the *<Extension>* tag by MEGA

The *<Extension>* tag is used by the MEGA import function which creates a report of commands analyzed at import. The report contains commands analyzed in MEGA XML format indicating if they have been accepted, rejected or contain warnings. Therefore each MEGA XML format command in the report can contain an *<Error>* tag or *<Warning>* tags in the *<Extension>* tag of the command concerned. These tags enable appreciation of the import result, but are not to be taken into account in another context.

Example 1 : Use of the *<Extension>* tag by MEGA

```
<MegaExchange>
  <Header>
    <!-- document information -->
  </Header>
  <Commands>
    <!-- exchanged commands -->
    <Add>
      <Object metaclass.name="Application">
        <Extension>
          <Error code="XXXXXXXX">There is no « metaattribute » with « color » value as
« name »<Error>
        </Extension>
        <!-- object attributes -->
        <Attribute metaattribute.name="_idabs">ODiTpSNApa10</Attribute>
        <Attribute metaattribute.name="name">Invoicing</Attribute>
        <Attribute metaattribute.name="color">blue</Attribute>
      </Object>
    </Add>
  </Commands>
</MegaExchange>
```

Attributes of MEGA XML documents used by MEGA

A MEGA XML document exported from a MEGA repository contains a certain number of attributes that describe general elements relating to the document itself. These are described in this section.

Document attributes are located in the *<Header>* tag of the document.

Example: Document attributes

```
<?xml version="1.0" encoding="UTF-8"?>
<MegaExchange>
  <Header>
```

```
    <Attribute metaattribute.id="27C729AE3F4E004A"
metaattribute.name="exchange_format_version">
      <Value format="internal">000100</Value>
      <Value format="display">MEGA XML version 0.1</Value>
    </Attribute>
    <Attribute metaattribute.id="27C729AE3F4E0050" metaattribute.name="mega_version">
      <Value format="internal">25856</Value>
      <Value format="display">MEGA Version 2005 Pre release</Value>
    </Attribute>
    <!-- ... -->
  </Header>
  <Commands>
    <!-- ... -->
  </Commands>
</MegaExchange>
```

Document attributes list

- exchange_format_version : this attribute indicates the version of MEGA XML exchange format.
- mega_version : this attribute indicates the version of MEGA used at export.
- metamodel_language : this attribute specifies the language used to identify types in the attributes of *"metaclass.name", "metaattribute.name", "metaassociationend.name"* tags.
- model_default_language : this attribute specifies the default language used for translatable attribute values of objects and links when the language has not been specified by *"language.name"* or *"language.id"* attributes of the *<Value>* tag.
- author : the value of the *author* attribute identifies the user that executed export.
- source_database : the value of the *source_database* attribute identifies the repository from which export was executed.
- creation_date : specifies MEGA XML document creation date.

GLOSSARY

XML attribute:

An XML attribute is unstructured text data contained in an XML tag.

Example: <attribute tag="value"/>

MEGA object attribute:

Object characteristics are called attributes. For a given object, attributes can take a value of which the type is defined for the attribute.

XML tag:

XML tags are syntax elements that structure XML documents. They mark opening and closing of XML document data definition.

Example: <tag>…</tag>

MEGA command:

The various actions possible that modify a MEGA repository are called commands. In particular these include creation, deletion and modification of objects and links.

XML element :

An XML element is structured data comprising an opening tag, attributes of this tag, a closing tag and text data and tags contained between opening and closing tags.

Example: <attribute tag="value">text<sub-tag/></tag>

File export

MEGA repository data can be exported in a file. From a MEGA repository we can export : objects (exported data describing exported objects), transactions (exported data describing transaction commands) or the complete repository (exported data describing all repository objects, this procedure also being known as logical backup). The various export functions are described in the "MEGA Help" CHM file and in the "MEGA Administration" PDF user guide in the following sections: "Administration > Managing Transactions > Managing Updates > Exporting the Transaction Logfile", "Administration > Managing Objects > Exporting MEGA objects", "Administration > Managing Repositories > Reorganizing a Repository > Logical Backup of a Repository".

MEGA APIs enable file export using Automation interfaces. Data export and logical backup functions using API are covered in the "MEGA Help" CHM file in the section "API > API - Reference Guide > Administration > MEGA Database" and in the "MEGA API" PDF user guide in the similarly named section.

IdAbs

Absolute identifier of data stored in the MEGA repository, idabs is generally represented in hexadecimal or base 64 form. The idabs is for example an object attribute (attribute "_idabs").

File import

Importing an XML document in a MEGA repository means executing or creating the commands it contains in this repository. MEGA can import files of type MGR, MGL, MGE and MEGA XML. The import function is covered in the "MEGA Help" CHM file in the section "Administration > Managing Repositories > Reorganizing a Repository > Updating a Repository (Importing)" and in the "MEGA Administration" PDF user guide in the similarly named section.

MEGA APIs enable file import using Automation interfaces. The API import function is covered in the "MEGA Help" CHM file in the section "API > API - Reference Guide > Administration > MEGA Database" and in the "MEGA API" PDF user guide in the similarly named section.

MEGA links :

For a given repository object, links enable definition of connected repository objects.

MEGA MetaAssociation :

This term indicates relationships existing between repository object types. MetaAssociations define repository links.

MEGA MetaClass :

This term indicates object types stored in the repository. MetaClasses define repository objects.

MEGA metamodel :

This comprises all MetaClasses and MetaAssociations defining objects that can be created and handled in the MEGA repository. The MEGA metamodel is covered in the "MEGA Help" CHM file in section "Metamodel and Glossaries > Metamodel".

MEGA object

Data stored in a MEGA repository are called repository objects. An object comprises attributes and can be connected to other repository objects. In addition, every object is identified uniquely by an attribute called the MEGA absolute identifier or idabs.

MEGA XML

Common name for MEGA data exchange XML format.

XML

Extendable tag language (eXtensible Markup Language), XML is a metalanguage describing languages of tagged hierarchical structure. XML specifications provide syntax basics of XML languages (ref. Extensible Markup Language (XML) 1.0: http://www.w3.org/TR/2004/REC-xml-20040204/).

REPORTING DATAMART

MEGA International

9 avenue René Coty - 75014 Paris, France
tél +33 (0)1 42 75 40 00 - fax +33 (0)1 42 75 40 95 - info@mega.com - www.mega.com
S.A. au capital de 1 106 698 € - RC Paris B 385 185 806 000 28 / NAF 741 G

1. REPORTING DATAMART OVERVIEW 3
2. REPORTING DATAMART DESCRIPTION 4
2.1. Structure 4
2.2. Data 5
2.3. Content 5
2.4. Excluding MetaAttribute values from the Reporting Datamart 5
3. CREATING AND INITIALIZING A REPORTING DATAMART 6
3.1. Creating a Reporting Datamart 6
3.2. Initializing the Reporting Datamart structure 9
3.3. Initializing the Reporting Datamart data 9
4. SYNCHRONIZING THE REPORTING DATAMART WITH THE HOPEX REPOSITORY CONTENT 10
4.1. Synchronization frequency 10
4.2. Launching an incremental synchronization 10
4.3. Launching a calculated MetaAttribute synchronization 11
4.4. Launching a diagram synchronization 12
5. DELETING A REPORTING DATAMART 14
6. USING THE REPORTING DATAMART 15
6.1. Usage 15
6.2. Advantages 15
7. REPORTING DATAMART DETAILED DESCRIPTION 16
7.1. Reporting Datamart table classification 16
7.2. Reporting Datamart table name format 16
7.2.1. Technical tables 16
7.2.2. MetaClass occurrence tables 18
7.2.3. MetaAssociation tables 18
7.3. Reporting Datamart columns 20
7.3.1. MetaClass occurrence table columns 20
7.3.2. Link table columns 20
7.4. Use case: reading the Reporting Datamart through a link 21
7.4.1. Accessing your Reporting Datamart tables 21
7.4.2. Understanding a link table 21
7.4.3. Finding the MetaClass corresponding to an IdAbs 22
7.4.4. Finding attributes or tagged values belonging to a MetaClass occurrence 23
7.4.5. Finding information on an attribute 23
7.5. Use case: saving the diagram drawings 26

1. REPORTING DATAMART OVERVIEW

With the **Reporting Datamart** feature you can create a **Reporting Datamart**, which is a replicated RDBMS Database from an HOPEX repository content.

The **Reporting Datamart** is made up of data selected at creation and synchronized on regular basis, to keep the **Reporting Datamart** updated according to the HOPEX repository content.

The purpose of the **Reporting Datamart** feature is to be used as a source for any usage that needs HOPEX data (for example: reporting).

The **Reporting Datamart** feature is available with **HOPEX Datamart** license.

2. REPORTING DATAMART DESCRIPTION

2.1. Structure

The **Reporting Datamart** structure is initialized with the RDBMS database standard structure.

Tables are created and fed with the HOPEX repository and SystemDb repository data. If there is no data to be exported to a table, this table remains empty, so that the **Reporting Datamart** structure is always consistent. This ensure the user that any of his queries run properly on the **Reporting Datamart**.

Reporting Datamart table and column names are in English language.

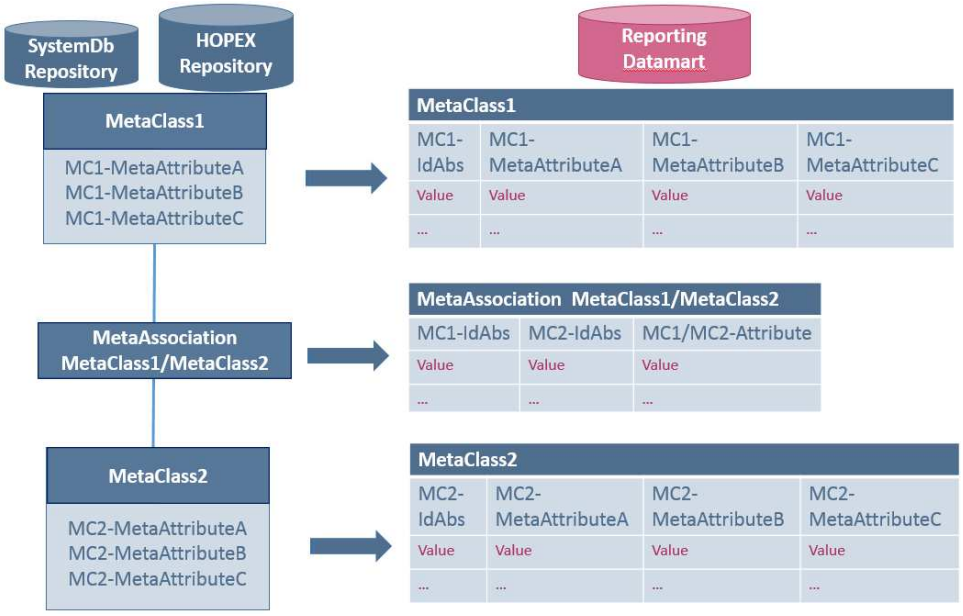
Reporting Datamart data is stored in the language selected at **Reporting Datamart** creation.

SQL database-based Reporting Datamart

For an SQL database-based **Reporting Datamart** the rule is as follows:

- an SQL Database Table is created for each HOPEX MetaClass
- an SQL Database Table is created for each HOPEX MetaAssociation
- an SQL Database Column is added in the SQL database Table for each HOPEX MetaAttribute

See Reporting Datamart Detailed description for a detailed description.



2.2. Data

HOPEX repository

Most of the HOPEX repository data is exported to the **Reporting Datamart**, including:

- MetaClasses, MetaAssociations, and MetaAttributes
- diagrams exported in BLOB column of their object
- calculated MetaAttributes
- comments, in html format (instead of RTF format)

Logs and object history are not exported.

SystemDb repository

Few data is exported from the SystemDb repository:

- Logins
- Persons (system)
- Person groups

2.3. Content

With the **Reporting Datamart,** data is filtered so that the data amount is reduced to the data needed only:

- Only the last dispatched version of each object is kept in the Reporting Datamart.
- Data is filtered at Reporting Datamart creation according to a user rights (confidentiality) and his profile permissions.
- Data name is displayed (not Idabs) for an easy reading.
- Data is in the **Reporting Datamart** language.

  Note: if you need the **Reporting Datamart** data in multiple languages, you need to create as many **Reporting Datamart** as languages needed.

2.4. Excluding MetaAttribute values from the Reporting Datamart

In some cases, you might not need to export some specific MetaAttribute values in your **Reporting Datamart**.

To exclude MetaAttribute values from the Reporting Datamart:

1) In HOPEX, access the MetaAttribute properties.
2) Click the **Characteristics** tab, then **Advanced** subtab.
3) Click the **Extended Properties** field arrow and select **Exclude from Reporting Datamart**.

3. CREATING AND INITIALIZING A REPORTING DATAMART

You can create and initialize a **Reporting Datamart** from an RDBMS HOPEX repository of your HOPEX Environment as follows:

Step	Reporting Datamart		Description	See
1	Creation		- Reporting Datamart data language definition - User and profile selection	Creating a Reporting Datamart
2	Initialization	Structure (optional)	Creation of all the tables and columns according to the HOPEX repository metamodel	Initializing the Reporting Datamart structure
		Data	Creation and feeding of all the Reporting Datamart tables, columns and rows according to the HOPEX repository content	Initializing the Reporting Datamart data

Note: you can also initialize the Reporting Datamart using APIs, see HOPEX power Studio > Initializing and synchronizing a Reporting Datamart.

3.1. Creating a Reporting Datamart

At **Reporting Datamart** creation you define:

- the **Reporting Datamart** storage
- the data language of the **Reporting Datamart**

  Note: not translated data appears in its Idabs format.
- the user and its profile

  They both define and filter the HOPEX repository data according to their access rights (confidentiality: reading access areas of object occurrences and of linked objects for link occurrences) and permissions.

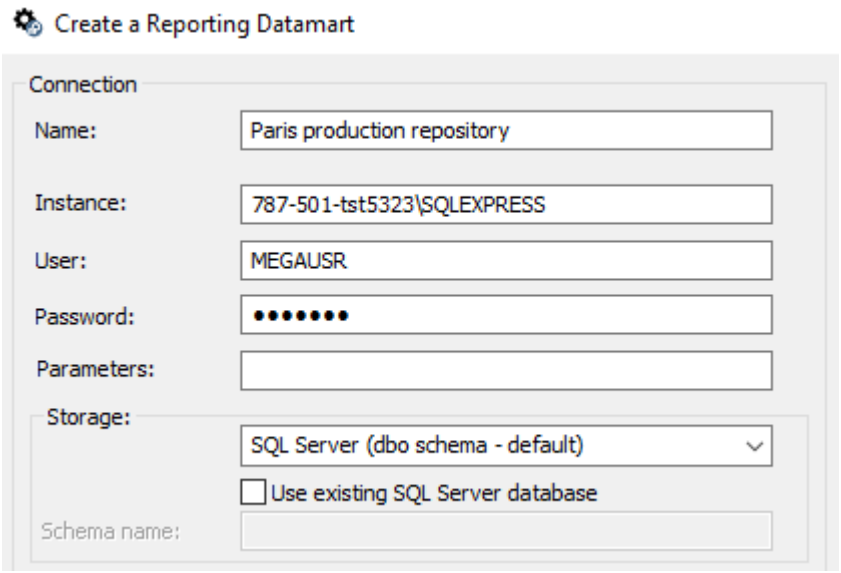
To create a Reporting Datamart (SQL Server format) from a HOPEX repository:

1) Launch **HOPEX Administration** application.
2) Connect to the environment from which you want to replicate the data.
3) In the **Repositories** folder, expand the repository folder concerned.
4) Right-click **Reporting Datamart** and select **New**.

   The Create a Reporting Datamart window opens.
5) In the **Connection** pane:
   a) In the **Name** field, enter your **Reporting Datamart** name.
   b) (If needed) Modify the HOPEX repository **Instance** set by default as well as its connection parameters (**User**, **Password**, **Parameters)**, see *RDBMS Repository Installation Guide* for detailed information.
   c) In the **Storage** pane, use the drop-down menu to select the **Reporting Datamart** storage type: “SQL Server (dbo schema - default)”.

d) (If you do not have the right to create a database) Select **Use existing SQL Server database**.

Note: if you selects **Use existing SQL Server database**, in the **Name** field you must enter the name of the existing database you want to use.



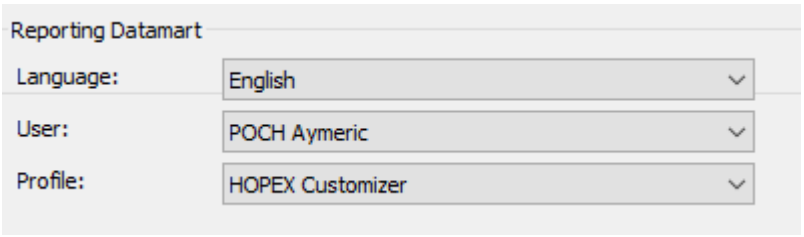
6) In the **Reporting Datamart** pane:

a) In the **Language** field, select the data language of the **Reporting Datamart**.

Note: select a language in which the data is available, so that you can read it.

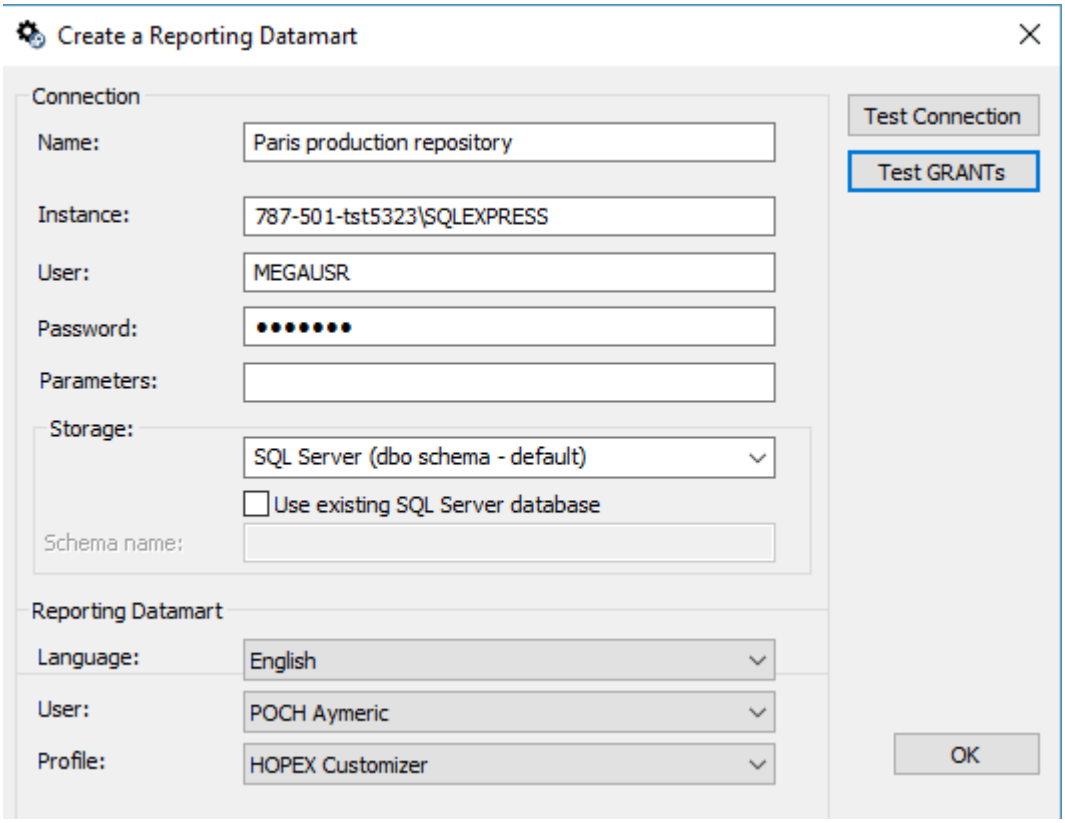
b) In the **User** field, select the user that defines the access to HOPEX repository.

c) In the **Profile** field, select the profile that defines the access to HOPEX repository.



7) Click **Test Connection**.

8) Click **Test GRANT's**.



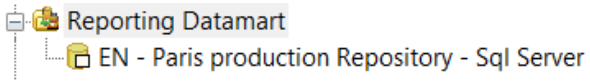
9) Click **OK**.

The Reporting Datamart is created.

The **Reporting Datamart** is added in the Reporting Datamart folder with the following format:

```
<Language> - <Reporting Datamart name> - <Storage type>
```

```
Example: EN - Paris production Repository - Sql Server
```



3.2. Initializing the Reporting Datamart structure

Once the Reporting Datamart is created you can initialize the structure with the HOPEX repository metamodel.

All the tables and columns are created even if there is no data to feed them. The table remains empty, so that the Reporting Datamart structure is always consistent. This ensures the user that any of his queries run properly on the Reporting Datamart.

To initialize the Reporting Datamart structure from the HOPEX repository:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.
3) In the **Repositories** folder, expand the repository folder concerned.
4) In the **Reporting Datamart** folder, right-click the Reporting Datamart concerned and select **Initialize > Structure**.

   The Reporting Datamart structure creation starts.

   The Reporting Datamart is initialized according to the HOPEX repository metamodel: all the tables (columns and rows) are created.

   If you have already initialized the Reporting Datamart data, only the empty tables and columns are added.

   Else initialize the Reporting Datamart data to feed the tables, rows and columns (see Initializing the Reporting Datamart data section).

3.3. Initializing the Reporting Datamart data

Once the Reporting Datamart is created you need to initialize the Reporting Datamart data with the HOPEX repository content according to the user/profile permissions and the object type selected.

To initialize the Reporting Datamart data from the HOPEX repository:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.
3) In the **Repositories** folder, expand the repository folder concerned.
4) In the **Reporting Datamart** folder, right-click the Reporting Datamart concerned and select **Initialize > Data**.
5) Click **Start**.

   The Reporting Datamart data initialization starts.

   The Reporting Datamart Data is initialized according to the HOPEX repository content.

4. SYNCHRONIZING THE REPORTING DATAMART WITH THE HOPEX REPOSITORY CONTENT

To keep your **Reporting Datamart** up-to-date with HOPEX repository updates, you can synchronize it with HOPEX Repository content thanks to the following provided triggers:

- Incremental synchronization
- Calculated MetaAttribute synchronization
- Diagram synchronization

Although, if needed, you can modify the **Reporting Datamart** data, these modifications are never synchronized with the HOPEX repository. Synchronization is only one way: from HOPEX repository to **Reporting Datamart**.

4.1. Synchronization frequency

The schedule policy of these synchronization triggers depends on the object volume in the HOPEX Repository and on the up-to-date data level you want to deliver to the Reporting Datamart users.

If you need a perfect synchronization between data, calculated MetaAttributes and Diagrams you can daily run the corresponding triggers at the same time.

Synchronization scheduling use: once scheduled the trigger launch the synchronization on all the repositories and for all their Reporting Datamarts.

To trigger the action on a specific repository for a specific Reporting Datamart, see HOPEX Power Studio > All about starting with APIs > Initializing and synchronizing a Reporting Datamart.

4.2. Launching an incremental synchronization

The HOPEX repository log must be activated (default value for an RDMS repository).

Launch an incremental synchronization to update the **Reporting Datamart** with all the dispatches performed after the last synchronization.

You can:

- at any time, manually launch an incremental synchronization
- define a scheduled trigger for the incremental synchronization

  Trigger: Reporting Datamart Synchronization

  Frequency: you can schedule the incremental synchronization with a high frequency (i.e.: every 10').

  For information regarding the scheduling, see *HOPEX Power Studio > Using the Scheduler > Scheduler*.

To manually launch an incremental synchronization:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.

3) In the **Repositories** folder, expand the repository folder concerned.

4) In the **Reporting Datamart** folder, right-click the Reporting Datamart concerned and select **Synchronize > Incremental update**.

To define the scheduled trigger for an incremental synchronization:

1) Launch **HOPEX Administration** application.

2) Connect to the environment concerned.

3) In the **Repositories** folder, expand the repository folder concerned.

4) Right-click **Scheduler** and select **Manage triggers.**

5) (If needed) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization** and select **Update Scheduling** to modify the scheduling according to your needs.

6) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization** and select **Activate**.

   The **Reporting Datamart Synchronization** is scheduled as defined (by default every 10').

4.3. Launching a calculated MetaAttribute synchronization

Launch a calculated MetaAttribute synchronization to scan all the objects and links of the HOPEX repository that can have calculated MetaAttribute values and put their values in the **Reporting Datamart**.

Do not launch or schedule a calculated MetaAttribute synchronization if you do not use the values of calculated MetaAttributes from the Reporting Datamart.

To exclude specific MetaAtribute values from this synchronization, see Excluding MetaAttribute values from the Reporting Datamart.

You can:

- manually launch a calculated MetaAttribute synchronization
- define a scheduled trigger for the diagram synchronization

  Trigger: Reporting Datamart Synchronization (Calculated MetaAttribute)

  Frequency: The synchronization may take time according to the HOPEX Repository calculated MetaAttribute volume. You should first manually launch a calculated MetaAttribute synchronization once to check the synchronization duration, so as to determine the synchronization scheduling frequency.

  When you schedule a "Reporting Datamart Synchronization (Calculated MetaAttribute)" the trigger is launched on all the repositories and for all their Reporting Datamarts.

  To trigger the action on a specific repository for a specific Reporting Datamart, see HOPEX Power Studio > All about starting with APIs > Initializing and synchronizing a Reporting Datamart.

  For information regarding the scheduling, see *HOPEX Power Studio > Using the Scheduler > Scheduler*.

To manually launch a calculated MetaAttribute synchronization:

1) Launch **HOPEX Administration** application.

2) Connect to the environment concerned.
3) In the **Repositories** folder, expand the repository folder concerned.
4) In the **Reporting Datamart** folder, right-click the Reporting Datamart concerned and select **Synchronize > Attributes calculated**.

To define the scheduled trigger for a calculated MetaAttribute synchronization:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.
3) In the **Repositories** folder, right-click **Scheduler** and select **Manage triggers.**
4) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization (Calculated MetaAttribute)** and select **Execute**

   Check the synchronization duration.
5) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization (Calculated MetaAttribute)** and select **Update Scheduling**.
6) Modify the scheduling according to your needs.
7) Click **OK**.
8) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization (Calculated MetaAttribute)** and select **Activate**.

   The **Reporting Datamart Synchronization (Calculated MetaAttribute)** is scheduled as defined.

4.4. Launching a diagram synchronization

Launch a diagram synchronization to scan all the HOPEX repository diagrams and update the **Reporting Datamart** with their drawing representation in the SVG format.

Do not launch or schedule a diagram synchronization if you do not use diagrams.

You can:

- manually launch a diagram synchronization
- define a scheduled trigger for the diagram synchronization

  Trigger: Reporting Datamart Synchronization (Diagrams)

  Frequency: The synchronization may take time according to the HOPEX Repository diagram volume. You should first manually launch a diagram synchronization once to check the synchronization duration, so as to determine the synchronization scheduling frequency.

When you schedule a "Reporting Datamart Synchronization (Diagrams)" the trigger is launched on all the repositories and for all their Reporting Datamarts.

To trigger the action on a specific repository for a specific Reporting Datamart, see HOPEX Power Studio > All about starting with APIs > Initializing and synchronizing a Reporting Datamart.

For information regarding the scheduling, see *HOPEX Power Studio > Using the Scheduler > Scheduler*.

To manually launch a diagram synchronization:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.
3) In the **Repositories** folder, expand the repository folder concerned.
4) In the **Reporting Datamart** folder, right-click the Reporting Datamart concerned and select **Synchronize > Diagrams**.

To define the scheduled trigger for a diagram synchronization:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.
3) In the **Repositories** folder, right-click **Scheduler** and select **Manage triggers.**
4) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization (Diagrams)** and select **Execute.**

   Check the synchronization duration.
5) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization (Diagrams)** and select **Update Scheduling**.
6) Modify the scheduling according to your needs.
7) Click **OK**.
8) In the **System Triggers** tab, right-click **Reporting Datamart Synchronization (Diagrams)** and select **Activate**.

   The **Reporting Datamart Synchronization (Diagrams)** is scheduled as defined.

5. DELETING A REPORTING DATAMART

You can delete a **Reporting Datamart**.

To delete a Reporting Datamart:

1) Launch **HOPEX Administration** application.
2) Connect to the environment concerned.
3) In the **Repositories** folder, expand the repository folder concerned.
4) In the **Reporting Datamart** folder, right-click the reporting Datamart concerned and select **Delete**.

6. USING THE REPORTING DATAMART

6.1. Usage

The **Reporting Datamart** can be used as:

- a data source to generate reports, through queries on a person data according to a specific profile.
- a repository to build custom static web sites with external CMS tools
- a repository to build a custom application, or an outbound interface to another tool
- a repository for mobile applications.

6.2. Advantages

Advantages of using the **Reporting Datamart** are that:

- data amount is reduced to comply with the creation parameters (user and profile rights) (see Content section)
- data access is simplified
- it contains up-to-date information (see Synchronizing the Reporting Datamart with the HOPEX Repository content section)
- it may run on a separate server
- it does not affect HOPEX repository performance while using it

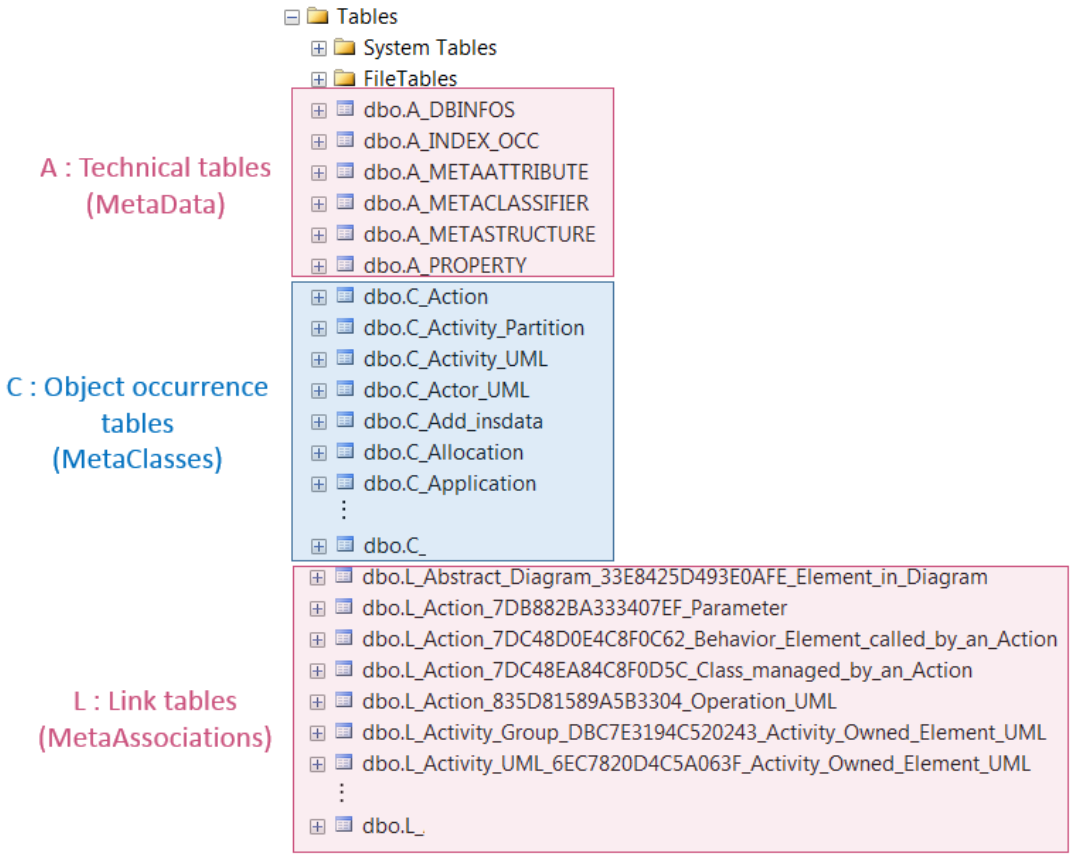
  However, see Launching a diagram synchronization and Launching a calculated MetaAttribute synchronization sections for HOPEX repository performance issues.

7. REPORTING DATAMART DETAILED DESCRIPTION

7.1. Reporting Datamart table classification

In the **Tables** folder, tables are classified according to their content:

- Technical data (MetaData)
- Object occurrences (MetaClass occurrences)
- Links (MetaAssociations)



7.2. Reporting Datamart table name format

Each table name format is as follows:

dbo.<letter>_<table name>

dbo: data base owner

<letter>: **A** for Technical tables

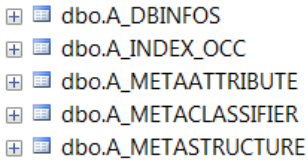
C for object occurrence tables

L for link tables

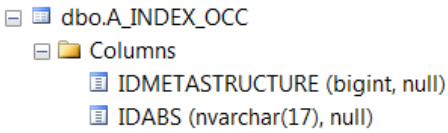
<table name>: describes the table content

7.2.1. Technical tables

Technical table name is prefixed with: **dbo.A_**.

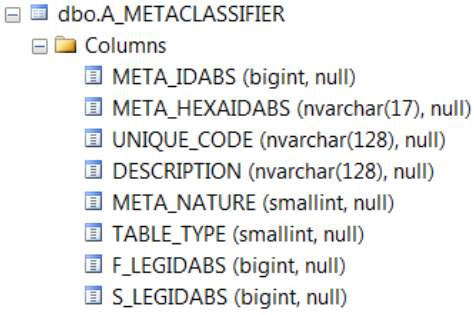


- dbo.A_**DBINFOS** table is for internal use only (source repository information).
- dbo.A_**INDEX_OCC** table details all the object IdAbs with their corresponding MetaStructure Idabs. It enables to find out to which table belongs an object.



For an example see Finding information on an attribute section.

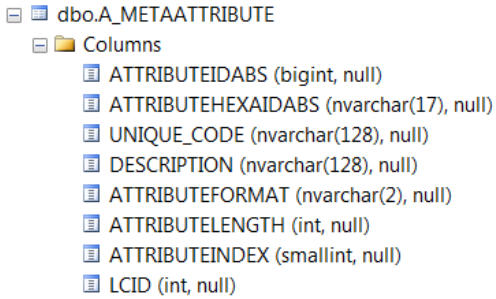
- dbo.A_**METACLASSIFIER** table describes the tables



  - META_IDABS and META_HEXAIDABS are the Metamodel object (MetaClass or MetaAssociation) IdAbs in HOPEX, in numerical (integer) and hexadecimal formats.
  - UNIQUE_CODE is the table name
  - DESCRIPTION is the Metamodel object (MetaClass or MetaAssociation) name in HOPEX
  - F_LEGIDABS and S_LEGIDABS give the MetaAssociationEnd Idabs

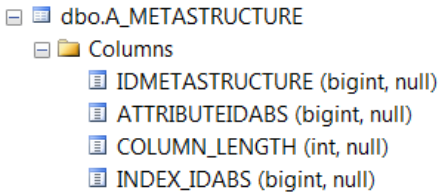
For examples see Finding the MetaClass, and Finding information on an attribute sections.

- dbo.A_**METATTRIBUTE** describes the columns



  - ATTRIBUTEIDABS and ATTRIBUTEHEXAIDABS are the MetaAttribute IdAbs in HOPEX
  - UNIQUE_CODE is the column name
  - DESCRIPTION is the MetaAttribute name in HOPEX

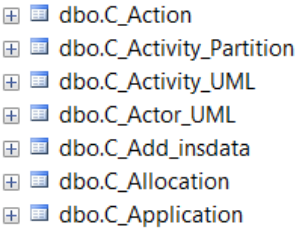
- dbo.A_**METASTRUCTURE** describes the Table x Column matrix



- IDMETASTRUCTURE gives the META_IdAbs of the dbo.A_METACLASSIFIER table
- ATTRIBUTEIDABS gives the MetaAttribute IdAbs in HOPEX.

7.2.2. MetaClass occurrence tables

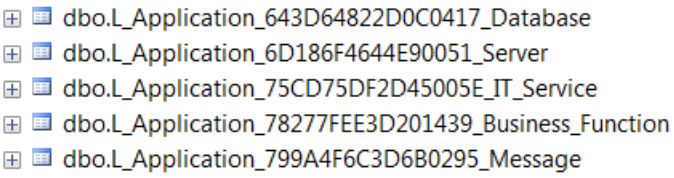
MetaClass occurrence table name is prefixed with: **dbo.C_**.



7.2.3. MetaAssociation tables

MetaAssociaton table name:

- is prefixed with: **dbo.L_**
- is made up of the MetaAssociation IdAbs between the major MetaAssociationEnd name (the first one) and the minor MetaAssociationEnd name (the second one)

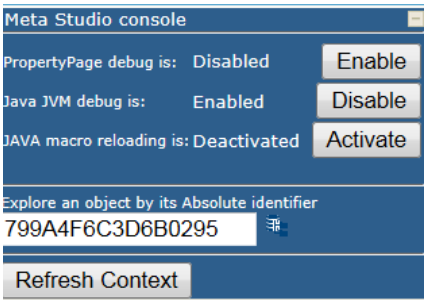


Example:

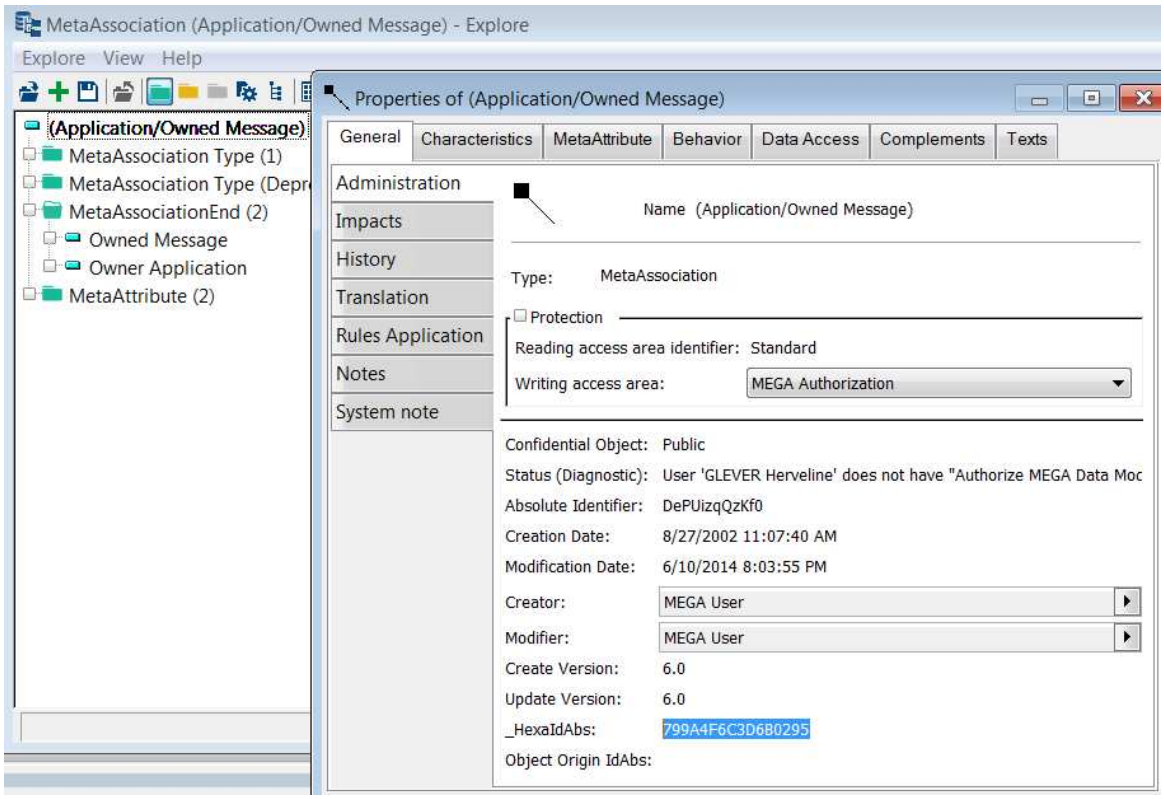
dbo.L_**Application**_799A4F6C3D6B0295_**Message**:

- dbo.L: indicates it is a link (MetaAssociation) table
- **Application**: the first MetaAssociationEnd (major)
- 799A4F6C3D6B0295: the "Application-Message" MetaAssociation IdAbs
- **Message**: the second MetaAssociationEnd (minor)

In HOPEX, with the MetaStudio console, in **Explore an object by its absolute identifier**, enter the 799A4F6C3D6B0295 IdAbs.



You get the MetaAssociation:



7.3. Reporting Datamart columns

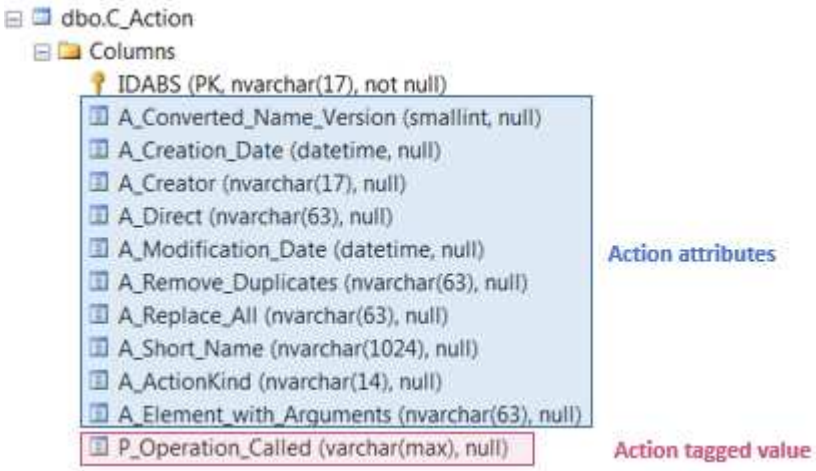
7.3.1. MetaClass occurrence table columns

In **dbo.C_**<MetaClass name>, the **Columns** folder details the attributes or tagged values belonging to the MetaClass occurrence concerned. Each column name is prefixed as follows:

- **A_** for attributes
- **P_** for tagged values

Example:

dbo.C_Action table includes in **Columns** folder all the attributes and tagged values belonging to the Actions.

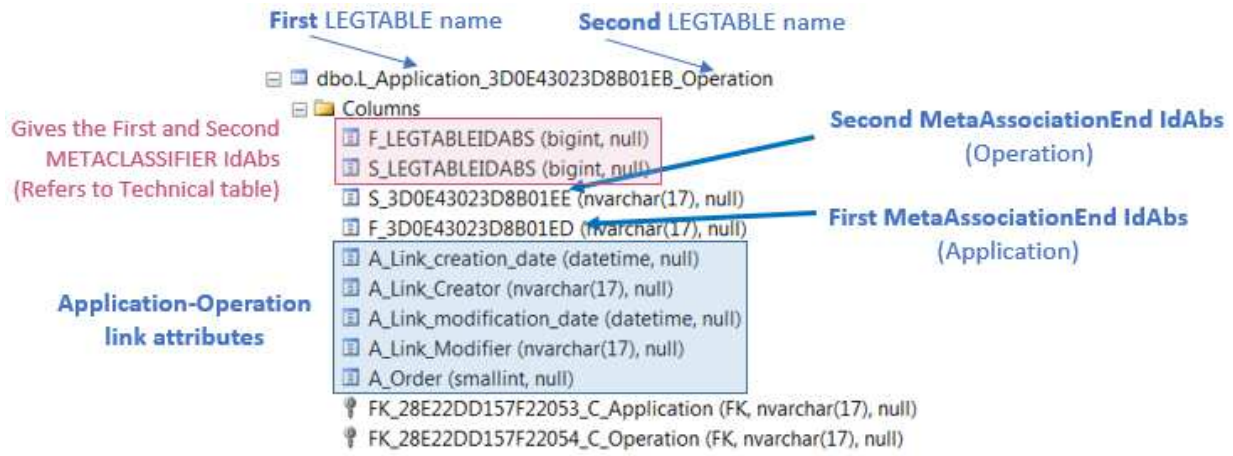


7.3.2. Link table columns

In **dbo.L_<…>**, the **Columns** folder details the attributes, and the first (major) and second (minor) MetaAssociationEnd IdAbs belonging to the MetaAssociation concerned. Each column name is prefixed as follows:

- F_LEGTABLEIDABS: indicates the first (major) table IdAbs
- S_LEGTABLEIDABS: indicates the second (minor) table IdAbs
- F_< IdAbs>: indicates the first (major) MetaAssociationEnd IdAbs
- S_< IdAbs>: indicates the second (minor) MetaAssociationEnd IdAbs
- A_<…> for attributes

Example:



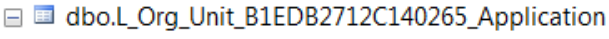
In dbo.L_**Application**_3D0E43023D8B01EB_**Operation** link table, the **Columns** folder is made up of:

- **Application** is the first MetaAssociationEnd (major)

  F_3D0E43023D8B01ED is the Application occurrence IdAbs
- **Operation** is the second MetaAssociationEnd (minor)

  S_3D0E43023D8B01EE is the Operation occurrence IdAbs

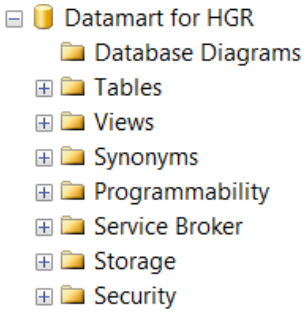
7.4. Use case: reading the Reporting Datamart through a link



7.4.1. Accessing your Reporting Datamart tables

To access your Reporting Datamart:

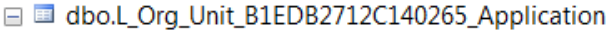
1) Launch Microsoft SQL Server Management Studio.
2) Access the server where you saved your reporting Datamart.
3) In the server tree view, expand **Databases** folder.
4) Expand your Reporting Datamart folder.



5) Expand the **Tables** folder.

7.4.2. Understanding a link table

Link example: Org_Unit - Application.



Dbo.L: “Org_Unit - Application” link

Org_Unit: major MetaAssociationEnd

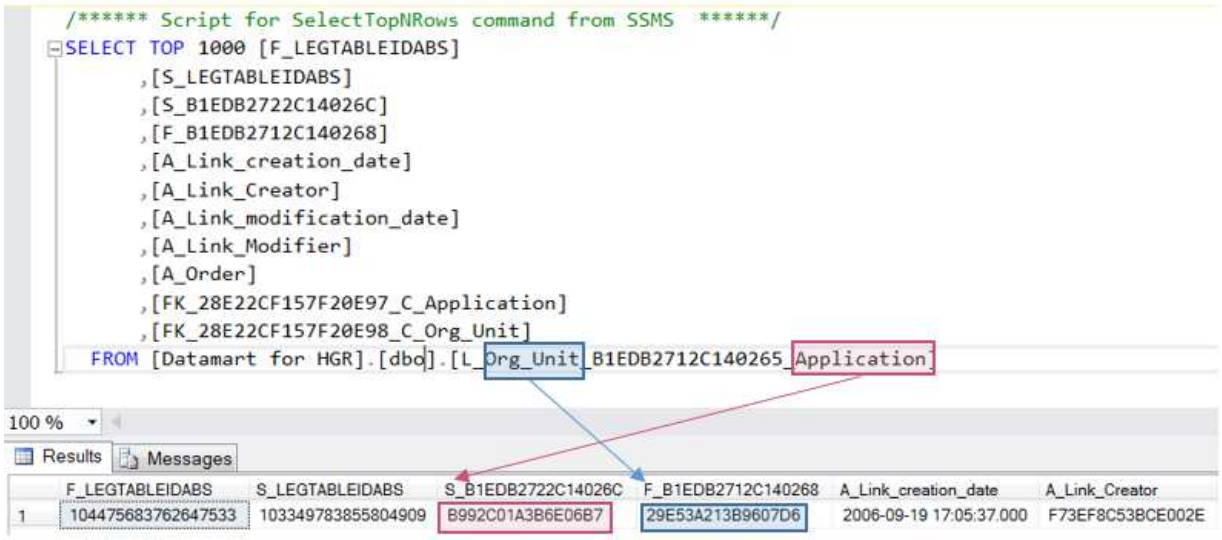
B1EDB2712C140265: MetaAssociation IdAbs

Application: minor MetaAssociationEnd

To understand the dbo.L_Org_Unit_B1EDB2712C140265_Application table information:

1) In your Reporting Datamart tree view, **Tables** folder, right-click **dbo_Org_Unit_B1EDB2712C140265_Application** table and select **Select Top 1000 rows**.

   The result is displayed in the right pane.



- 104475683762647533 is the **Org Unit** MetaClass IdAbs (for the first MetaAssociationEnd).

  See Finding the MetaClass corresponding to an IdAbs section.
- 103349783855804909 is the **Application** MetaClass IdAbs (for the second MetaAssociationEnd).
- 29E53A213B9607D6 is the **Org Unit** (first MetaAssociationEnd occurrence) IdAbs.

  See Finding attributes or tagged values belonging to a MetaClass occurrence section.
- B992C01A3B6E06B7 is the **Application** (second MetaAssociationEnd occurrence) IdAbs.

7.4.3. Finding the MetaClass corresponding to an IdAbs

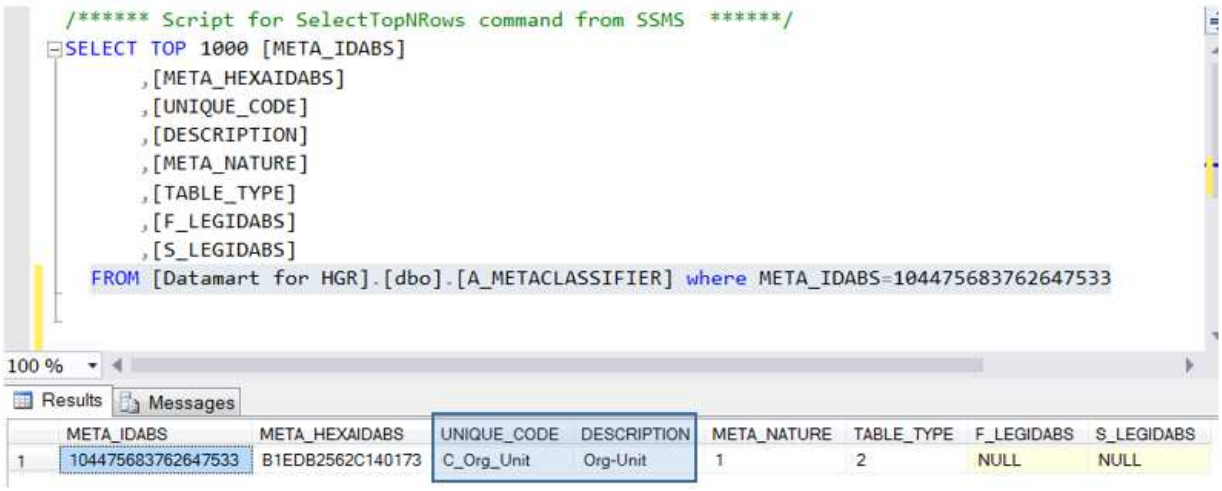
To find out the MetaClass corresponding to the 104475683762647533 IdAbs use the **dbo.A_METACLASSIFIER** table.

To find out the MetaClass corresponding to 104475683762647533 IdAbs:

1) In your Reporting Datamart tree view, right-click **dbo.A_METACLASSIFIER** and select **Select Top 1000 rows**.
2) In the right pane complete the query with : where META_IDABS=104475683762647533

```
FROM [Datamart for HGR].[dbo].[A_METACLASSIFIER] where
META_IDABS=104475683762647533
```

3) **Execute** the query.



The query result indicates that 104475683762647533 is the **Org-Unit** MetaClass IdAbs.

7.4.4. Finding attributes or tagged values belonging to a MetaClass occurrence

Use the **dbo.C_<MetaClass name>** table to find the attributes and tagged values belonging to a MetaClass occurrence.

For example 29E53A213B9607D6 is the IdAbs of an Org_Unit occurrence.

To find the Org_Unit table which IdAbs is 29E53A213B9607D6:

1) In your Reporting Datamart tree view, right-click **dbo.C_Org_Unit** and select **Select Top 1000 rows**.
2) In the right pane complete the query with: `where IDABS='29E53A213B9607D6'`

```
FROM [Datamart for HGR].[dbo].[C_Org_Unit] where IDABS='29E53A213B9607D6'
```

3) **Execute** the query.

The query result details all the Org-Unit attributes (IdAbs: 29E53A213B9607D6, Short Name: Controler). It indicates, in particular that its creator IdAbs is: 801180463B58001C.

7.4.5. Finding information on an attribute

You want information on the Creator attribute whose idabs is 801180463B58001C.

You need to:

- use the **dbo.A_INDEX_OCC** table to get the MetaClass IdAbs to which the Creator belongs to.
- use the **dbo.A_METACLASSIFIER** table to get the MetaClass name from its Idabs
- use the **dbo.C_<MetaClass name>** table to get the Creator (801180463B58001C IdAbs) information

To find the MetaClass IdAbs of the MetaClass occurrence whose IdAbs is 801180463B58001C:

1) To find out to which MetaClass belongs the occurrence with 801180463B58001C IdAbs, in your Reporting Datamart tree view, right-click **dbo.A_INDEX_OCC** and select **Select Top 1000 rows**.
2) In the right pane complete the query with: `where IDABS='801180463B58001C'`

```
FROM [Datamart for HGR].[dbo].[A_INDEX_OCC] where IDABS='801180463B58001C'
```

3) **Execute** the query.

```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP 1000 [IDMETASTRUCTURE]
      ,[IDABS]
  FROM [Datamart for HGR].[dbo].[A_INDEX_OCC] where IDABS='801180463B58001C'
```

100 %

Results | Messages

	IDMETASTRUCTURE	IDABS
1	8303511812964352	801180463B58001C

The query result indicates that the IDMETASTRUCTURE of the 801180463B58001C IdAbs is: 8303511812964352.

4) To find out which MetaClass has the 8303511812964352 IdAbs, in your Reporting Datamart tree view, right-click **dbo.A_METACLASSIFIER** and select **Select Top 1000 rows**.
5) In the right pane complete the query with: `where META_IDABS=8303511812964352`

```
FROM [Datamart for HGR].[dbo].[A_METACLASSIFIER] where
META_IDABS=8303511812964352
```

6) **Execute** the query.

```
      ,[DESCRIPTION]
      ,[META_NATURE]
      ,[TABLE_TYPE]
      ,[F_LEGIDABS]
      ,[S_LEGIDABS]
  FROM [Datamart for HGR].[dbo].[A_METACLASSIFIER] where META_IDABS=8303511812964352
```

100 %

Results | Messages

	META_IDABS	META_HEXAIDABS	UNIQUE_CODE	DESCRIPTION	META_NATURE	TABLE_TYPE	F_LEGIDABS	S_LEGID
1	8303511812964352	0000000080000001D	C_Person_System	Person	1	2	NULL	NULL

The query result indicates that 8303511812964352 IdAbs is the **Person System** MetaClass IdAbs.

7) To find out which Person has the 801180463B58001C IdAbs, in your Reporting Datamart tree view, right-click **dbo.C_Person_System** and select **Select Top 1000 rows**.
8) In the right pane complete the query with: `where IDABS='801180463B58001C'`

```
FROM [Datamart for HGR].[dbo].[C_Person_System] where IDABS='801180463B58001C'
```

```
    ,[A_Initials]
    ,[A_ParameterizationOfAlertsScheduling]
    ,[A_Report_Edition_Parameterization]
    ,[A_Data_Language]
    ,[A_Comment]
FROM [Datamart for HGR].[dbo].[C_Person_System] where IDABS='801180463B58001C'
```

100 %

Results | Messages

	IDABS	A_Creation_Date	A_Creator	A_Modification_Date	A_Short_Name	A_Number_of_Exe
1	801180463B58001C	2016-10-02 23:25:55.000	0000000022222222	2012-04-19 19:13:51.000	Patrick	0

The query result indicates that the 801180463B58001C IdAbs belongs to Patrick.

7.5. Use case: saving the diagram drawings

In **dbo.C_Diagrams** table, the diagrams drawings are in the **A_Drawing** column.

To save the diagram drawings:

1) Launch Microsoft SQL Server Management Studio.
2) Access the server where you saved your reporting Datamart.
3) Check in your configuration that the following features are unlock (= 1):

```
/*
  CONFIGURATION
  =============
*/
 ---- check the configuration for 'Ole Automation Procedures'. It must be set
to 1
EXEC sp_configure 'Ole Automation Procedures';
GO
 ---- setting 'Ole Automation Procedures' to 1
EXEC sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
EXEC sp_configure 'Ole Automation Procedures', 1;
GO
RECONFIGURE;
 /*
  EXECUTION
  =========
*/
```

4) Create a folder where you want to retrieve the diagram drawings.

   For example: C:\Pictures.

5) Enter the following code:

```
DECLARE @SQLIMG VARCHAR(MAX),
                @IDABS NVARCHAR(17),
                @NAME NVARCHAR(1024),
                @IMG_PATH VARBINARY(MAX),
                @FILENAME VARCHAR(MAX),
                @ObjectToken INT

DECLARE IMGPATH CURSOR FAST_FORWARD FOR
    SELECT [IDABS], [A_Short_Name], [A_Drawing] from [dbo].[C_Diagram]
OPEN IMGPATH

FETCH NEXT FROM IMGPATH INTO @IDABS, @NAME, @IMG_PATH
```

```
WHILE @@FETCH_STATUS = 0
  BEGIN
    SET @FILENAME = 'C:\Pictures\'+ @IDABS + '_' + @NAME + '.png'

    EXEC sp_OACreate 'ADODB.Stream', @ObjectToken OUTPUT
    EXEC sp_OASetProperty @ObjectToken, 'Type', 1
    EXEC sp_OAMethod @ObjectToken, 'Open'
    EXEC sp_OAMethod @ObjectToken, 'Write', NULL, @IMG_PATH
    EXEC sp_OAMethod @ObjectToken, 'SaveToFile', NULL, @FILENAME, 2
    EXEC sp_OAMethod @ObjectToken, 'Close'
    EXEC sp_OADestroy @ObjectToken

    FETCH NEXT FROM IMGPATH INTO @IDABS, @NAME, @IMG_PATH
 END

CLOSE IMGPATH
DEALLOCATE IMGPATH
```

HOPEX VISUAL STUDIO TEMPLATES

MEGA International

9 avenue René Coty - 75014 Paris, France
tél +33 (0)1 42 75 40 00 - fax +33 (0)1 42 75 40 95 - info@mega.com - www.mega.com
S.A. au capital de 1 106 698 € - RC Paris B 385 185 806 000 28 / NAF 741 G

1. PREREQUISITES 3
2. WEB SERVICE FACTORY (WSF) TEMPLATE 4
2.1. Installing the HOPEX WSF template 4
2.2. Creating an HOPEX Web Service 4
2.3. Controllers project 6
2.3.1. Implementing a controller 6
2.3.2. Security 7
2.3.3. Creating and using an EventSource 7
2.3.4. Log events 8
2.3.5. Using CacheComponent 8
2.4. Hosting.IIS project 8
2.5. Macro project 9
2.6. TestingConsole project 9
2.6.1. App.config 9
2.6.2. Program.cs 10
2.7. Deploying the HOPEX WSF Service 10
2.8. Solve possible Cross Origin Resource Sharing (CORS) issue 13
3. MACRO TEMPLATE 14
3.1. Installing the HOPEX macro template 14
3.2. Creating a macro 14
3.3. Deploying a macro 15
3.4. Calling a macro 16
4. IDENTITY PROVIDER TEMPLATE 17
4.1. Installing the HOPEX IdentityProvider template 17
4.2. Creating a new identity provider 17
4.3. Deploying an identity provider 18

1. PREREQUISITES

The following templates are available to help you create C# projects in:

- Visual Studio 2015
- Visual Studio 2017

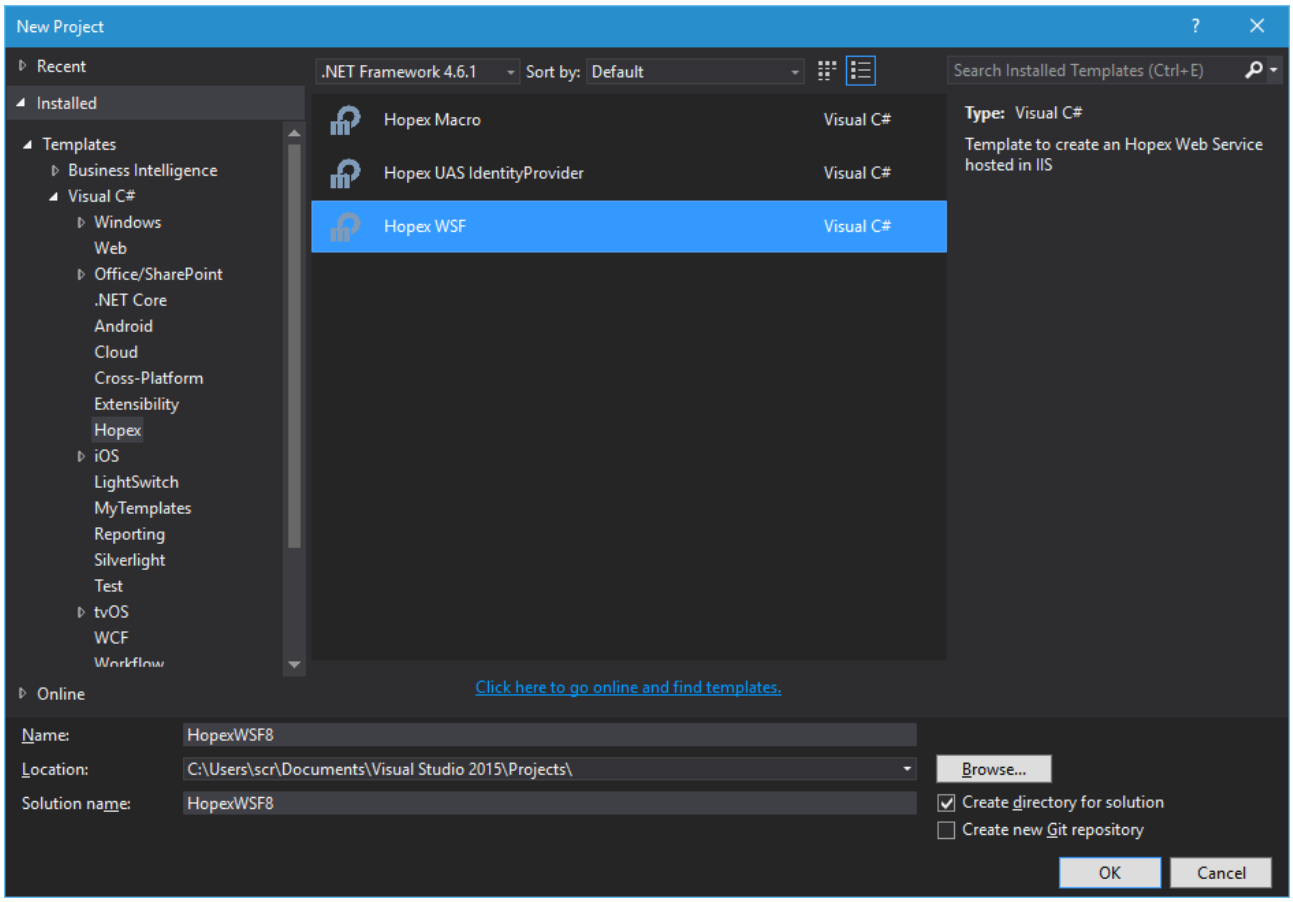
2. WEB SERVICE FACTORY (WSF) TEMPLATE

2.1. Installing the HOPEX WSF template

To install the HOPEX WSF template:

Use the VSIX file provided: Mega.WSF.Template.Installer.vsix.

This Visual Studio extension adds a new **C#** template in **Hopex** sub-section.

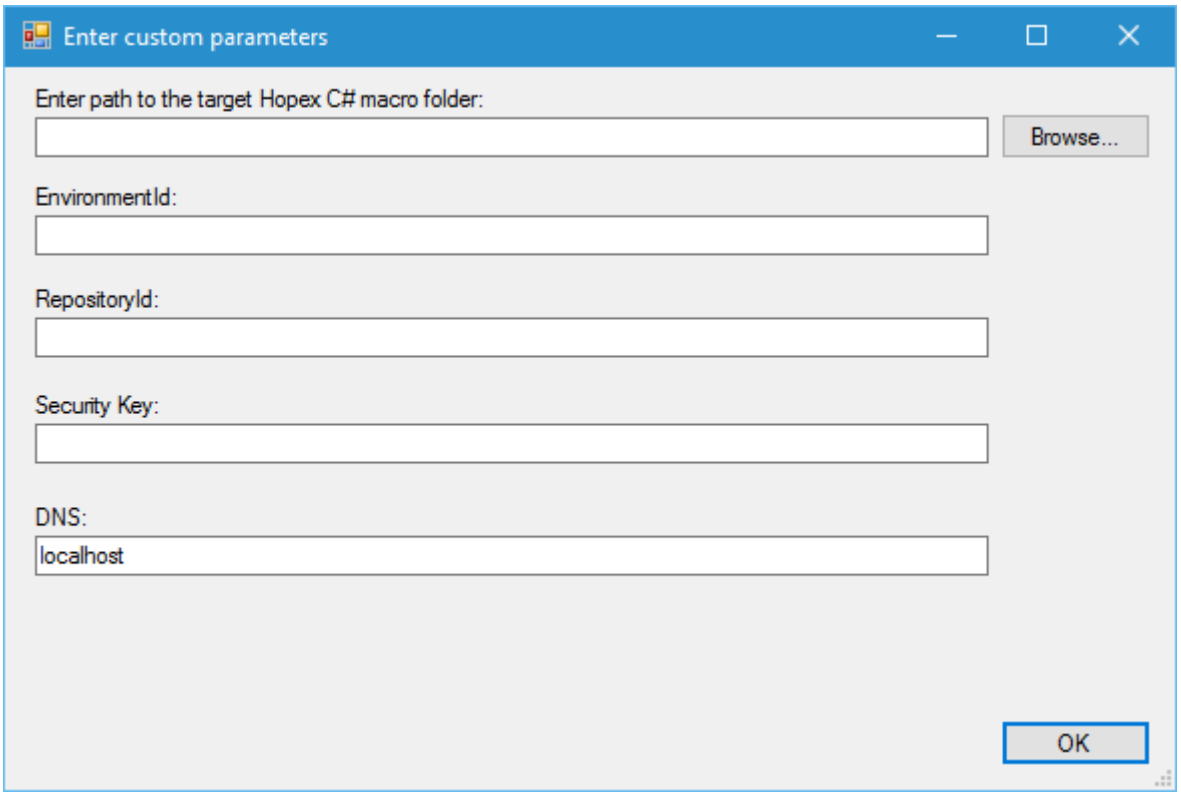


2.2. Creating an HOPEX Web Service

To create a Web service:

1) Create a new project in Visual Studio.
2) Expand **Templates/Visual C#.**
3) Select **Hopex** folder.
4) Select **Hopex WSF** type project.
5) Select **.NET Framework 4.6.2.**
6) Change the project's default name.
7) Click **OK**.

   A pop-up window asks you information to help you creating projects



8) Enter all the fields: **Path to the macro folder** (under Hopex/System folder), **EnvironmentId** and **RepositoryId** that you want to target, **Security Key** given during your installation, and machine name (**DNS**).
   - To get the environment id, you can use HOPEXAPI: http://DNSNAME/HOPEXAPI/restapi/v1/environments/ to get a list of environments.
   - To get the repository id, you can use HOPEXAPI: http://DNSNAME/HOPEXAPI/restapi/v1/environments/ENVIDABS/repositories/ to get a list of repositories
   - More info about HOPEXAPI at: Hopex Web Service API
9) Click **OK**.

The following projects are created:

- the IIS host project, which is your startup project.

See

Hosting.IIS project.

- the Controllers project, which must inherit from ApiController and contains entry points to launch and monitor your Job.

  See Controllers project.Controllers project.
- the Macro project, which is the location of your logic.

  See Macro project.
- the TestingConsole project, which is a Windows Console application that you can use to test your Web Service.

See **WARNING**: If your web services are synchronous, your macro must return its result within 90 seconds. If your treatment is longer than that you will experiment a timeout. To avoid this timeout, we strongly recommend implementing asynchronous web services and asynchronous macro if your macro exceed 20 seconds execution time.

TestingConsole project.

To run your web service:

1) Set Hosting.IIS and TestingConsole as starting projects.
2) Build the solution.
3) Run projects.
4) The macro is executed and returns “Hello World!”.

2.3. Controllers project

2.3.1. Implementing a controller

To implement a controller:

1) Inherit from ApiController.
2) Follow the SimpleCallController implementations provided in the template.

```
var macroMetadata = new MacroMetadata
{
       Id = ConfigurationManager.AppSettings["MacroId"],
       Assembly = "Mega.WSF.Template.Macro",
       Type = "Mega.WSF.Template.Macro.MyMacro",
       UseGenericMacro = true,
       Parameter = query.Parameter,
       AuthenticationToken = Request?.Headers.Authorization.Parameter,
       EnvironmentId = query.EnvironmentId,
       RepositoryId = query.RepositoryId,
       ProfileId = query.ProfileId
};

macroMetadata.HopexSessionToken = _macroBridge.BeginSession(macroMetadata);
var hopexResult = await Task.Run(() => _macroBridge.CallMacro(macroMetadata));
_macroBridge.CloseSession(macroMetadata.HopexSessionToken);
```

hopexResult variable contains the result returned by the macro.

NB: To call a macro registered in Hopex, you can set “UseGenericMacro” Boolean to false and call directly the macro by its Id. In this case, Macrometadata is as follow:

```
var macroMetadata = new MacroMetadata
{
       Id = "MacroId",
       UseGenericMacro = false,
       Parameter = query.Parameter,
       AuthenticationToken = Request?.Headers.Authorization.Parameter,
       EnvironmentId = query.EnvironmentId,
       RepositoryId = query.RepositoryId,
       ProfileId = query.ProfileId
};
```

2.3.2. Security

You can define the accessibility level (e.g.: “read”, “write”) of each method.

To define a method accessibility level:

1) In your Web service class, add the “HasScope” attribute to the method.

```
Example: [HasScope("read")]
```

2.3.3. Creating and using an EventSource

The EventSource class enables to create events for event tracing for Windows (ETW).

(https://msdn.microsoft.com/en-us/library/system.diagnostics.tracing.eventsource(v=vs.110).aspx).

To use telemetry:

1) Create a class inheriting from EventSource.

Add the EventSource attribute to your class.

For each message that you want to send to the telemetry, create a new method and call the WriteEvent method inside it.

Warning: Each method must have a distinct EventId.

Once your EventSource is created, you can start using it in your Web service.

To do so:

1) Use a private static readonly variable to load IServiceProvider interface and another one for your telemetry.

   ```
   Example:
   protected static readonly IServiceProvider Svc = PluginContainer.GetService<IServiceProvider>();
   private static readonly ExampleEventSource Telemetry = ExampleEventSource.Log.Value;
   ```

2) Enable it in your controller.

   ```
   Example:
   Svc.Telemetry.InstanceListener.EnableEvents(Telemetry, EventLevel.LogAlways);
   ```

3) Use it in your methods implementations.

   ```
   Example:
   Telemetry.GetJobResultMessage(nameof(ExampleController),
   Thread.CurrentThread.ManagedThreadId.ToString(), id);
   ```

2.3.4. Log events

To log information or errors:

1) In the ILogger interface, several methods are available.

Example:

```
Svc.Logger.Info(MyClassType, "This is an info message.");
Svc.Logger.Error(MyClassType, "This is an error message.");
```

2.3.5. Using CacheComponent

To use cache component:

1) Save any object serialized in a string in cache:

```
CacheComponent.InsertOrUpdate("{Id}", JsonConvert.SerializeObject(myObject));
```

2) Retrieve the object:

```
var myObject = JsonConvert.DeserializeObject<MyClass>(CacheComponent.Get("{Id}"));
```

2.4. Hosting.IIS project

The Hosting.IIS project hosts your controllers and uses the MegaWebServiceFactory to define how the service is hosted.

In the Hosting.IIS web.config file:

- MegaSiteProvider corresponds to the MegaSSP address
- SecurityKey is the HOPEX SecureKey.
- MacroId corresponds to the Id of the “CSharpLoaderAssemblyMacro” installed with HOPEX.

Example:

```
<appSettings>
  <add key="MegaSiteProvider" value="http://W-SCR/MEGASSP" />
  <add key="MacroId" value="E43FECDC57E37A15" />
  <add key="ConfigurationMode" value="Standalone" />
  <add key="HopexApiUrl" value="http://W-SCR/HOPEXAPI/restapi/v1/" />
  <add key="MegaWebAccessProvider" value="http://W-SCR/HOPEXMWAS" />
  <add key="AuthenticationServiceUrl" value="http://W-SCR/UAS" />
  <add key="MaxMegaSessionCount" value="2000" />
  <add key="OpenSessionTimeout" value="1800000" />
  <add key="StatMinPanelTime" value="200" />
  <add key="MultiThreadLimit" value="64" />
  <add key="MinConnectionDuration" value="10000" />
  <add key="MaxConnectionRetry" value="10" />
  <add key="CacheSerialize" value="1" />
  <add key="CacheFileDiscard" value="1" />
  <add key="CheckState" value="0" />
  <add key="LazyLog" value="1" />
  <add key="DisableCache" value="1" />
  <add key="KeepAlive" value="0" />
  <add key="AllowAnonymousConnection" value="0" />
  <add key="LogRequest" value="0" />
</appSettings>
<secureAppSettings>
  <add key="SecurityKey" value="" />
</secureAppSettings>
```

2.5. Macro project

This is where you implement your business logic. You can use all of the HOPËX APIs to retrieve, create, or modify objects in HOPEX. The ExecuteMacro method will be given a string parameter and the MegaRoot Object and will return the desired result in a string.

Example:

```
public string ExecuteMacro(string parameters, dynamic megaStartPoint)
{
    try
    {
        var megaRoot = new MegaRoot(megaStartPoint);
        var diagram = megaRoot.GetCollection("~KekPBSs3iS10[Diagram]");
        var value = "Hello " + parameters + " Number of Diagrams:" + diagram.Count;
        return JsonConvert.SerializeObject(new
        {
            Id = Guid.NewGuid(),
            Value = value
        });
    }
    finally
    {
        GC.Collect();
    }
}
```

WARNING: If your web services are synchronous, your macro must return its result within 90 seconds. If your treatment is longer than that you will experiment a timeout. To avoid this timeout, we strongly recommend implementing asynchronous web services and asynchronous macro if your macro exceed 20 seconds execution time.

2.6. TestingConsole project

2.6.1. App.config

The app.config file contains the necessary keys to get your HOPEX context:

```
<appSettings>
  <add key="JobUrl" value="http://localhost:54534" />
  <add key="UasUrl" value="http://MyServer/UAS" />
  <add key="ClientId" value="Hopex" />
  <add key="ClientSecret" value="Secret" />
  <add key="UserName" value="Me" />
  <add key="UserPassword" value="MyPassword" />
  <add key="Scopes" value="read write openid" />
  <add key="HopexEnvironmentId" value="MyEnvironmentId" />
  <add key="HopexRepositoryId" value="MyRepositoryId" />
  <add key="HopexProfileId" value="MyProfileId" />
</appSettings>
```

- JobUrl is the url of your web service (ie. the Hosting.IIS project).
- UasUrl is the url of your UAS installation.
- ClientId is the application's client ID (how the API identifies the application).
- ClientSecret is the corresponding client secret.
- UserName is the HOPEX login that executes the macro.

- UserPassword is the corresponding password.
- Scopes specifies the level of access that the application is requesting.
- HopexEnvironmentId is your HOPEX environment id, you can get it using HOPEX web API.
- HopexRepositoryId is the id of your HOPEX repository, you can get it using HOPEX web API.
- HopexProfileId is the id of the profile that runs the macro.

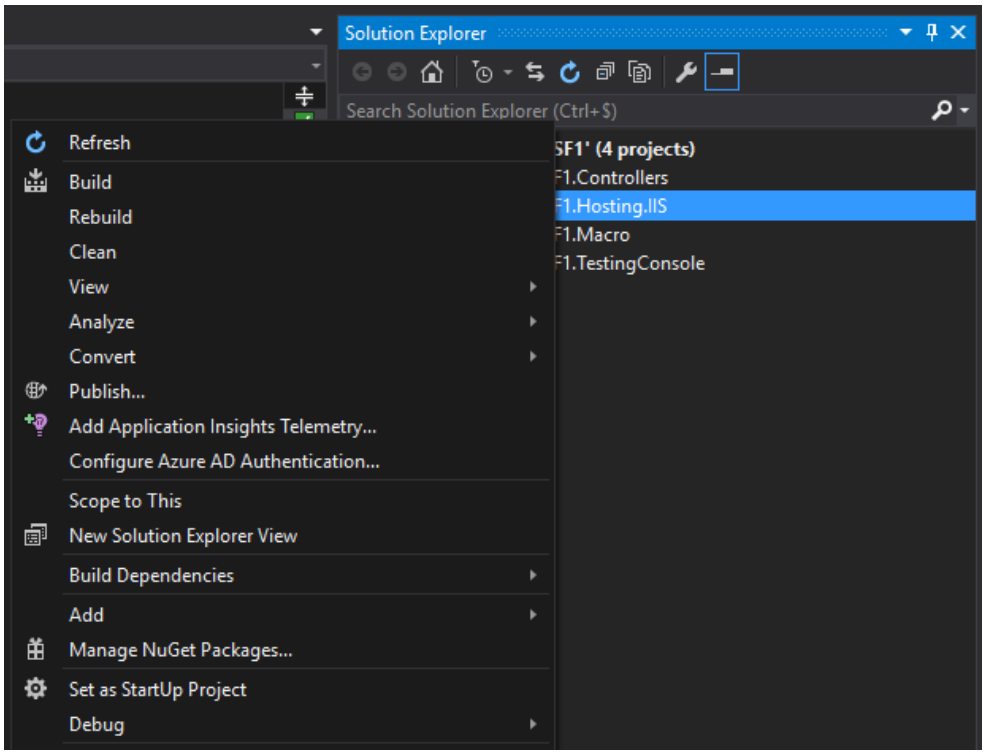
2.6.2. Program.cs

This class contains the main method and is used to call UAS and your web service for testing purposes.

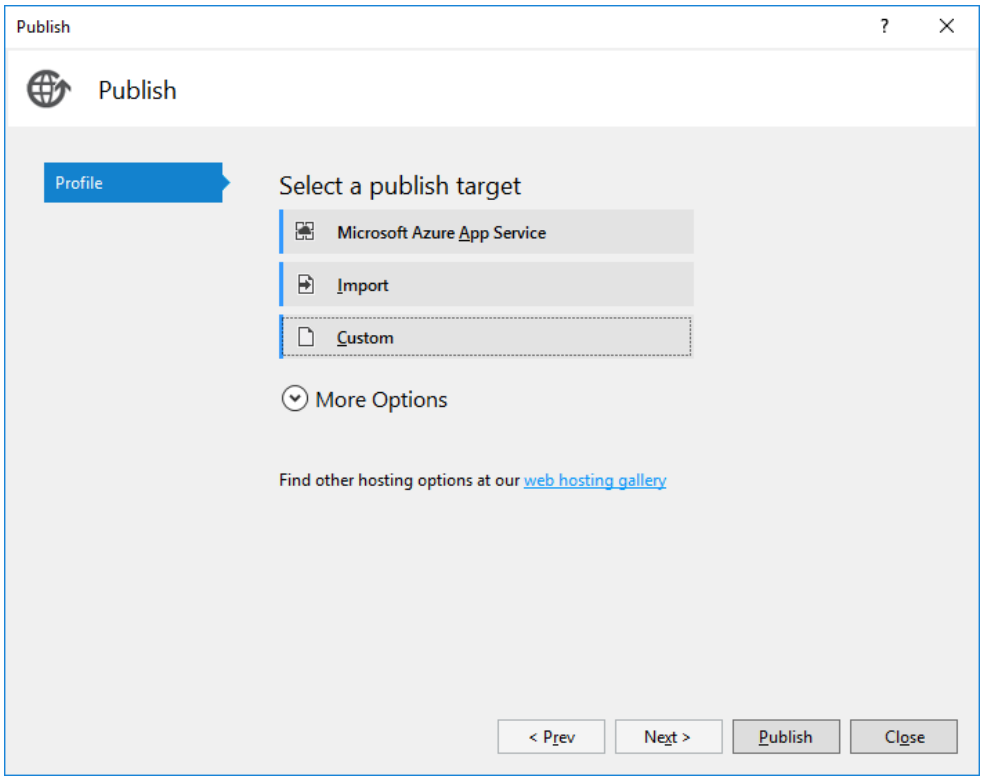
2.7. Deploying the HOPEX WSF Service

To create the deployment package:

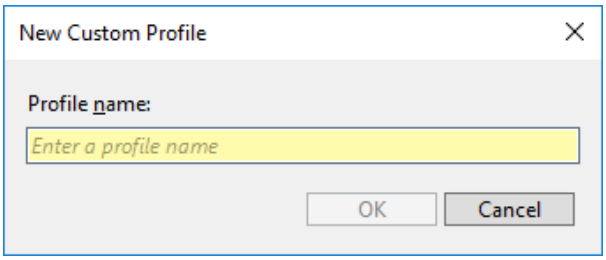
1) In Visual Studio, right-click your IIS project.



2) Click **Publish.**

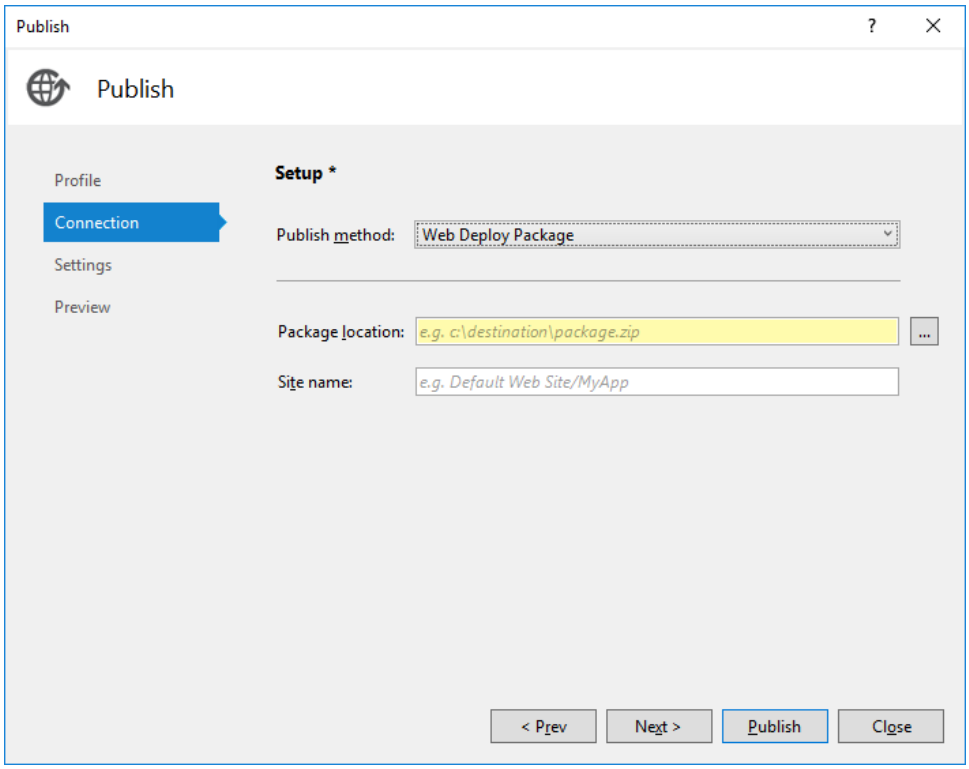


3) In **Select a publish target**, select **Custom**.



4) Enter a **Profile name**.

5) Click **OK**.



6) In the **Publish method** field, select **Web Deploy Package**.
7) Enter a **Package location** and a **Site name**.
8) Click **Publish.**

To deploy your package:

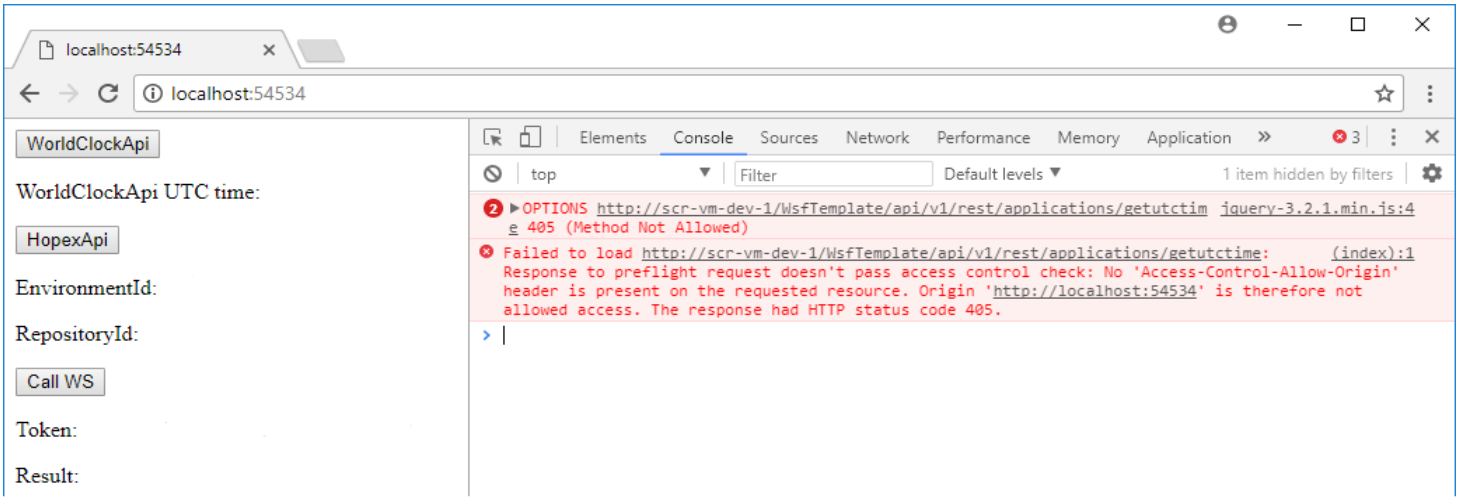
1) Open a command prompt.
2) Go to your package location.
3) Execute the following command:

```
{PackageName}.cmd /Y
```

4) Check that your package is well deployed in IIS.

2.8. Solve possible Cross Origin Resource Sharing (CORS) issue

If you call your web service from another server within an HTML page, using jQuery for instance, you will run into the following issue:



To allow CORS, you must check that the lines in yellow are present in your web.config file.

```
<system.webServer>
  <handlers>
    <remove name="ExtensionlessUrlHandler-Integrated-4.0" />
    <remove name="OPTIONSVerbHandler" />
    <remove name="TRACEVerbHandler" />
    <add name="ExtensionlessUrlHandler-Integrated-4.0" path="*." verb="*"
type="System.Web.Handlers.TransferRequestHandler" preCondition="integratedMode,runtimeVersionv4.0" />
    <add name="OPTIONSVerbHandler" path="*" verb="OPTIONS" modules="ProtocolSupportModule"
requireAccess="None" responseBufferLimit="4194304" />
  </handlers>
  <httpProtocol>
    <customHeaders>
      <add name="Access-Control-Allow-Origin" value="http://localhost:54534" />
      <add name="Access-Control-Allow-Headers" value="Content-Type, Authorization, HopexContext" />
    </customHeaders>
  </httpProtocol>
</system.webServer>
```

In **“Access-Control-Allow-Origin”**, enter the url of the web site that hosts your HTML page.

```
In this example: “http://localhost:54534”.
```

Now your service is accessible from this url but not from others.

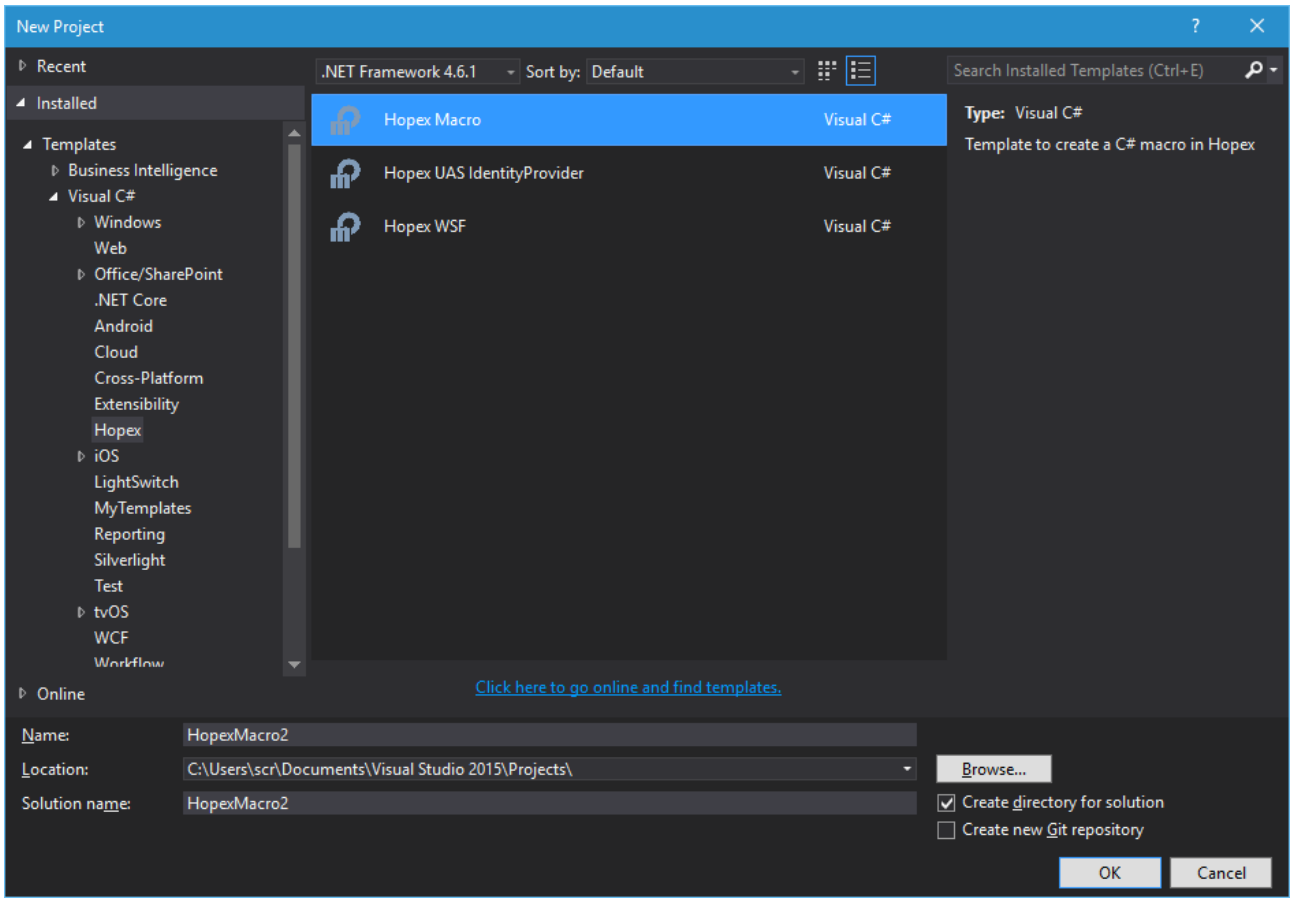
3. MACRO TEMPLATE

3.1. Installing the HOPEX macro template

To install the HOPEX macro template:

Use the VSIX file provided: Mega.Macro.Extension.vsix.

This Visual Studio extension adds a new **C#** template in **Hopex** sub-section.



3.2. Creating a macro

To create a macro:

1) Create a new project in Visual Studio.
2) Expand **Templates/Visual C#**
3) Select **Hopex** folder
4) Select **Hopex Macro** type project.
5) Select **.NET Framework 4.6.1**
6) Change the project's default name
7) Click **OK**.

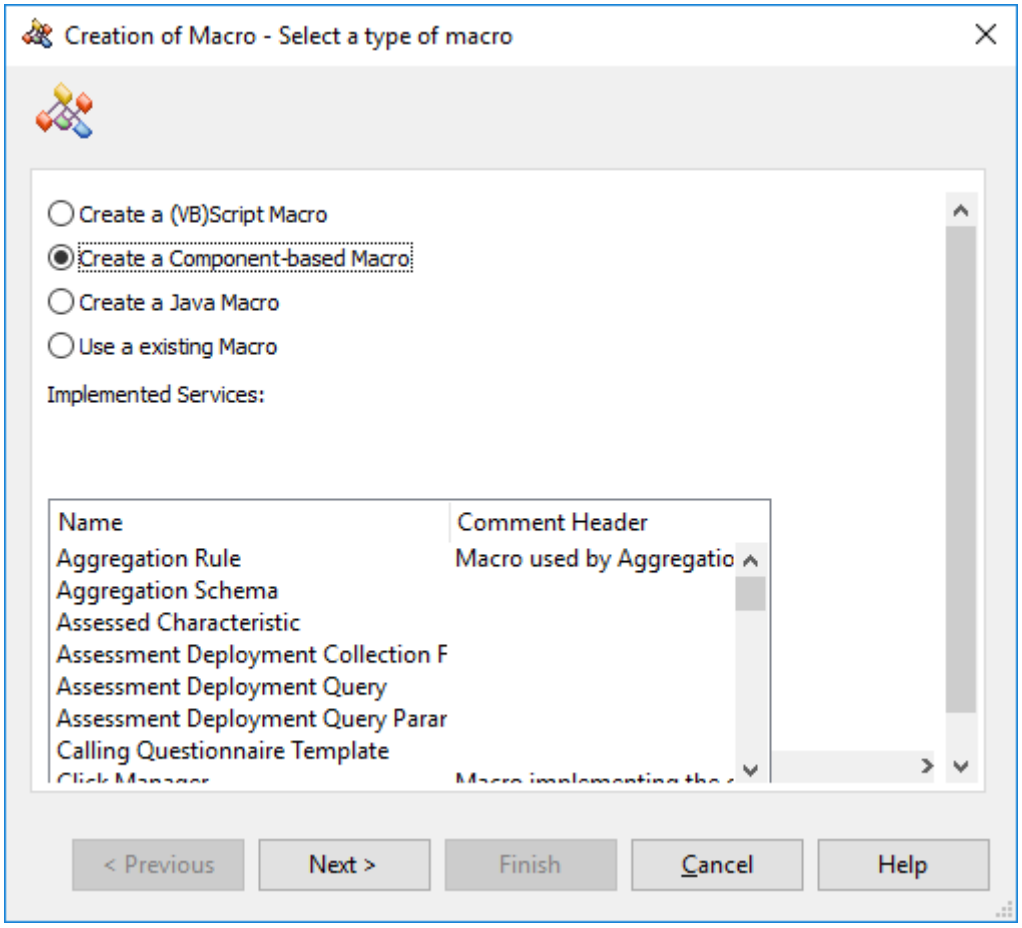
When you create this new project, a new project is created with the required packages and references, and one class (Macro) that contains your macro implementation:

```
public class Macro : Mega.Macro.Interfaces.IInvokableOnObject
{
	public string InvokeOnObject(MegaRoot myObject)
	{
		return "Invoked on c#: " + ((dynamic)myObject).Name;
	}
	public string Test(string message)
	{
		return message;
	}
}
```

3.3. Deploying a macro

To deploy a macro:

1) Compile your DLL.
2) Copy your DLL to \\HOPEXINSTALLDIR\dotnet\assemblies_usr.
3) Create a macro in HOPEX.

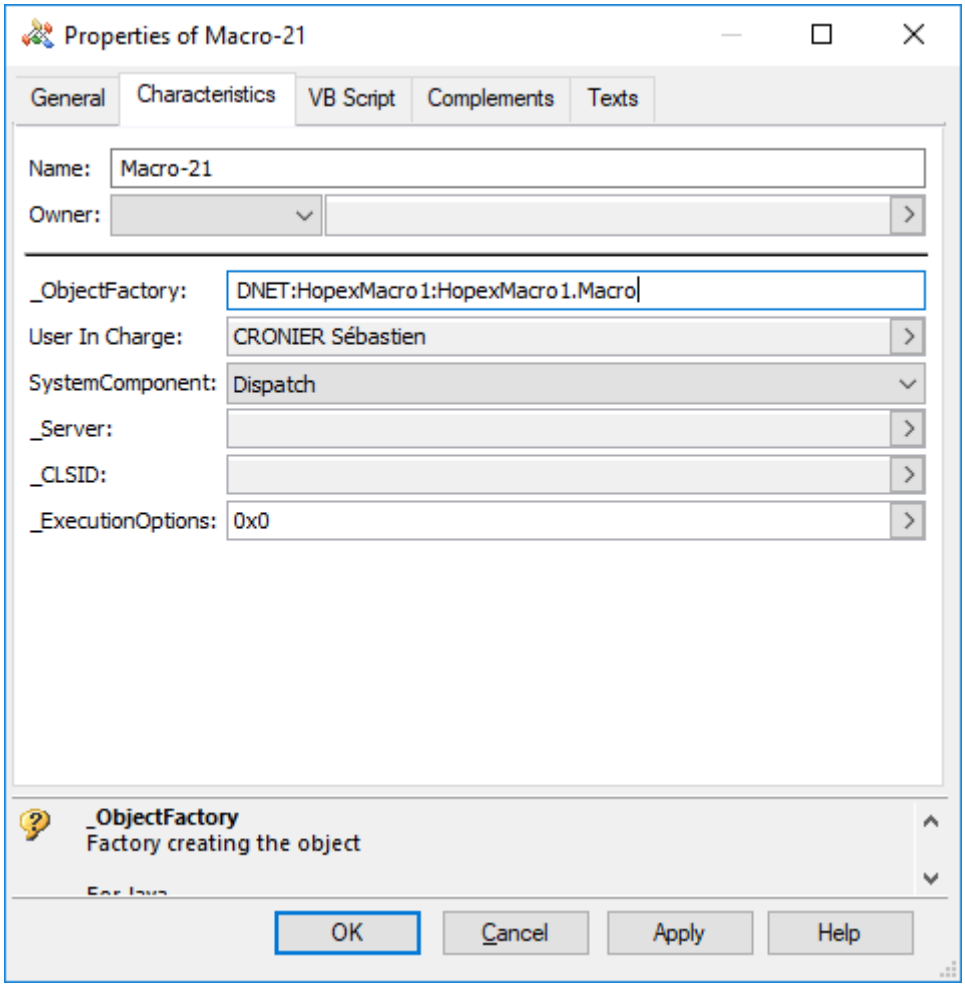


4) Click **Next**, keep default values.
5) Click **Finish**.

6) Open the macro property pages.

7) In the **_ObjectFactory** field, reference your DLL:

```
DNET:HopexMacro1:HopexMacro1.Macro
```



3.4. Calling a macro

You can execute your macro in HOPEX.

To execute your macro:

1) In HOPEX, open the script editor.

2) Copy the following script replacing macro name by your values:

```
set m = currentenvironment.getMacro("Macro-21")
print m.Add(1,2)
```

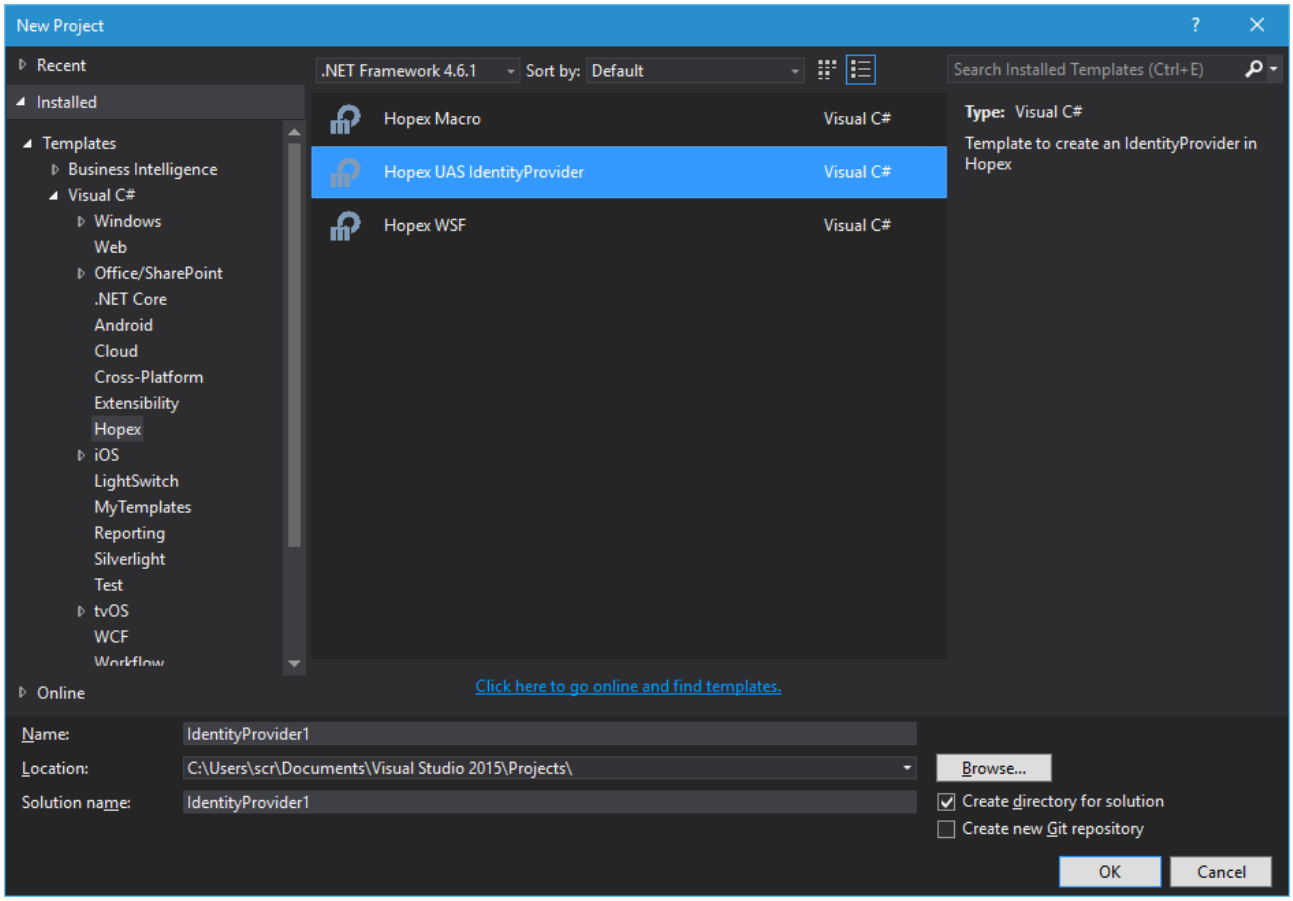
4. IDENTITY PROVIDER TEMPLATE

4.1. Installing the HOPEX IdentityProvider template

To install the HOPEX IdentityProvider template:

Use the VSIX file provided: Mega.UAS.IdentityProvider.Extension.vsix.

This Visual Studio extension adds a new **C#** template in **Hopex** sub-section.



4.2. Creating a new identity provider

To create a new provider:

3) Create a new project in Visual Studio.
4) Expand **Templates/Visual C#**
5) Select **Hopex** folder
6) Select **Hopex UAS IdentityProvider** type project.
7) Select **.NET Framework 4.6.1**
8) Change the project's default name
9) Click **OK**.

To implement a new provider:

1) Modify the Startup class that contains your identity provider implementation
2) Call your project: **MEGA.UAS.IdentityProvider.{ProviderName}** .
3) Add all your code in the Configure method. This class use Owin middleware system. Your new identity provider will be integrated into the HOPEX Owin pipeline.
4) Your class must inherit from `IIdentityProvider` to be recognized by HOPEX as an identity provider.

Example of identity provider implementation: https://github.com/TerribleDev/OwinOAuthProviders

```
public class Startup : IIdentityProvider
{
    /// <summary>
    ///  Parameters
    /// </summary>
    public string Parameters { get; set; }

    /// <summary>
    ///     Name of your namespace
    /// </summary>
    public string Namespace { get; set; }

    /// <summary>
    ///     Dictionary of your parameters - Key/Value
    /// </summary>
    public Dictionary<string, string> Properties { get; set; }


    public void Configure(IAppBuilder appBuilder, string signInAsType)
    {
        //TODO: Implement your identity logic
    }
}
```

4.3. Deploying an identity provider

To deploy the new identity provider:

1) Compile your DLL.
2) Copy the DLL and paste it in the **UAS folder bin**.

RDBMS Environment Duplication

Introduction **3**

Prerequisites **4**

The SQL Server account 4

Backup/Restore of SQL Server databases **10**

Backup and file transfer 10

Restore 18

When duplicating a repository in an environment – Expert mode 28

Post Installation Tasks **29**

Create/Attach an environment in HOPEX 29

Copy the documents from source to target 35

Get the parameters of the environment 41

INTRODUCTION

The goal of this document is to give detailed instructions as to how duplicating a HOPEX environment when it is hosted on an **SQL Server RDBMS**.

We will take the hypothesis that the new databases of the duplicated environment are hosted on a separate SQL Server instance. This way, it gives the appropriate details in case you move data from one server to another.

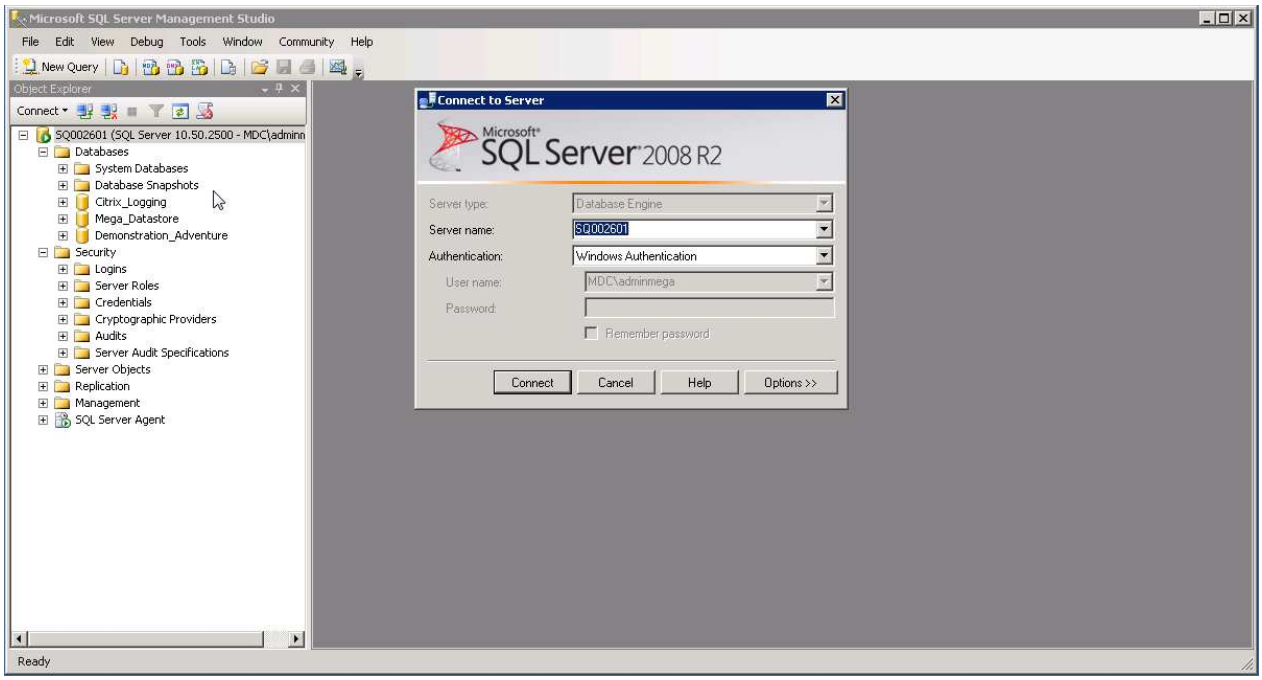
PREREQUISITES

The SQL Server account

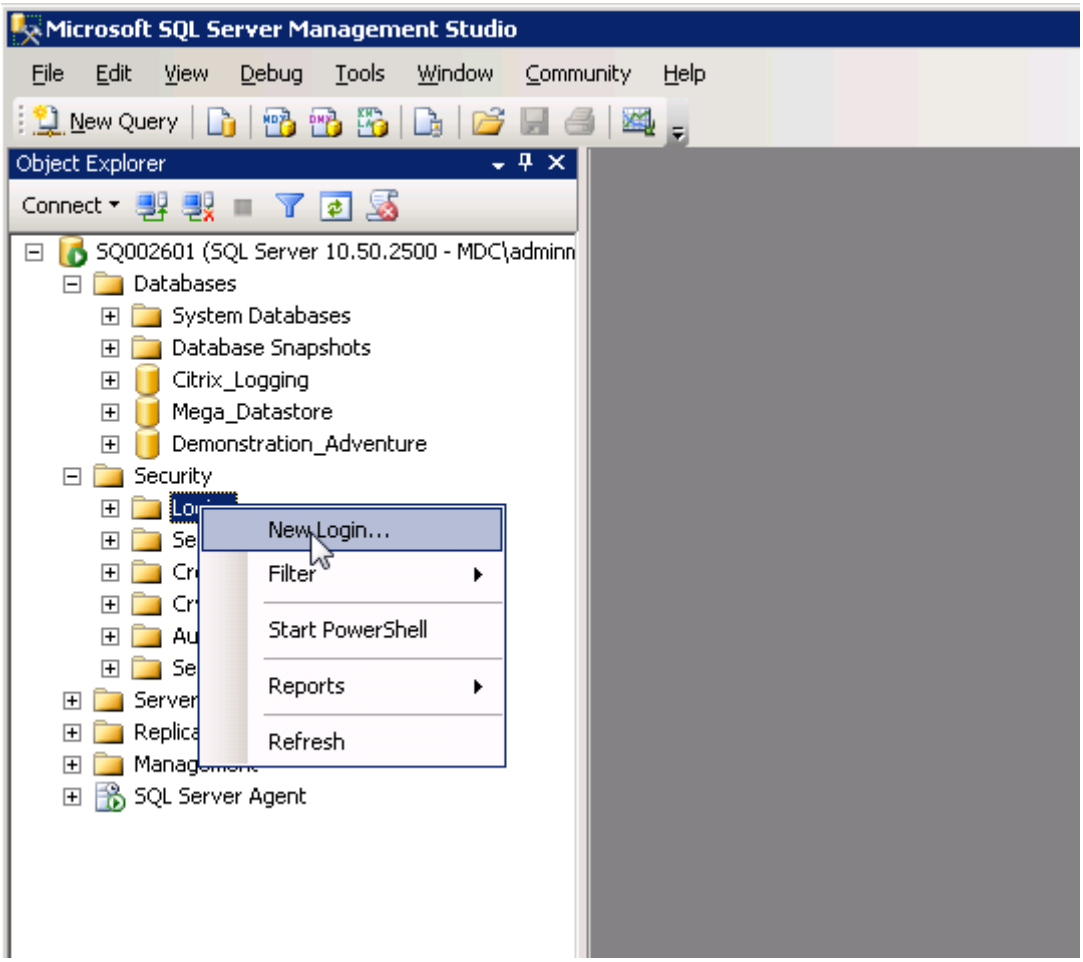
Before you start, you have to make sure that, where the duplicated databases are going to be hosted, you have a user with the proper rights, either:

- you are given the sysadmin right (the super admin) on the instance, and you perform those actions yourself, or
- you have to ask the DBA to create it and grant it.

1. Launch the "Microsoft SQL Server Management Studio" tool.
2. Connect to the SQL Server instance with the user having the sysadmin right.
3. If your Windows account is the one having the sysadmin right, stay in « Windows Authentication » mode. Otherwise, switch to "SQL Server Authentication", and provide the username and password. The field "Server Name" is the name of the instance.



4. In the left pane, expand **Security** file.
5. Right-click **Logins** and select **New Login**.

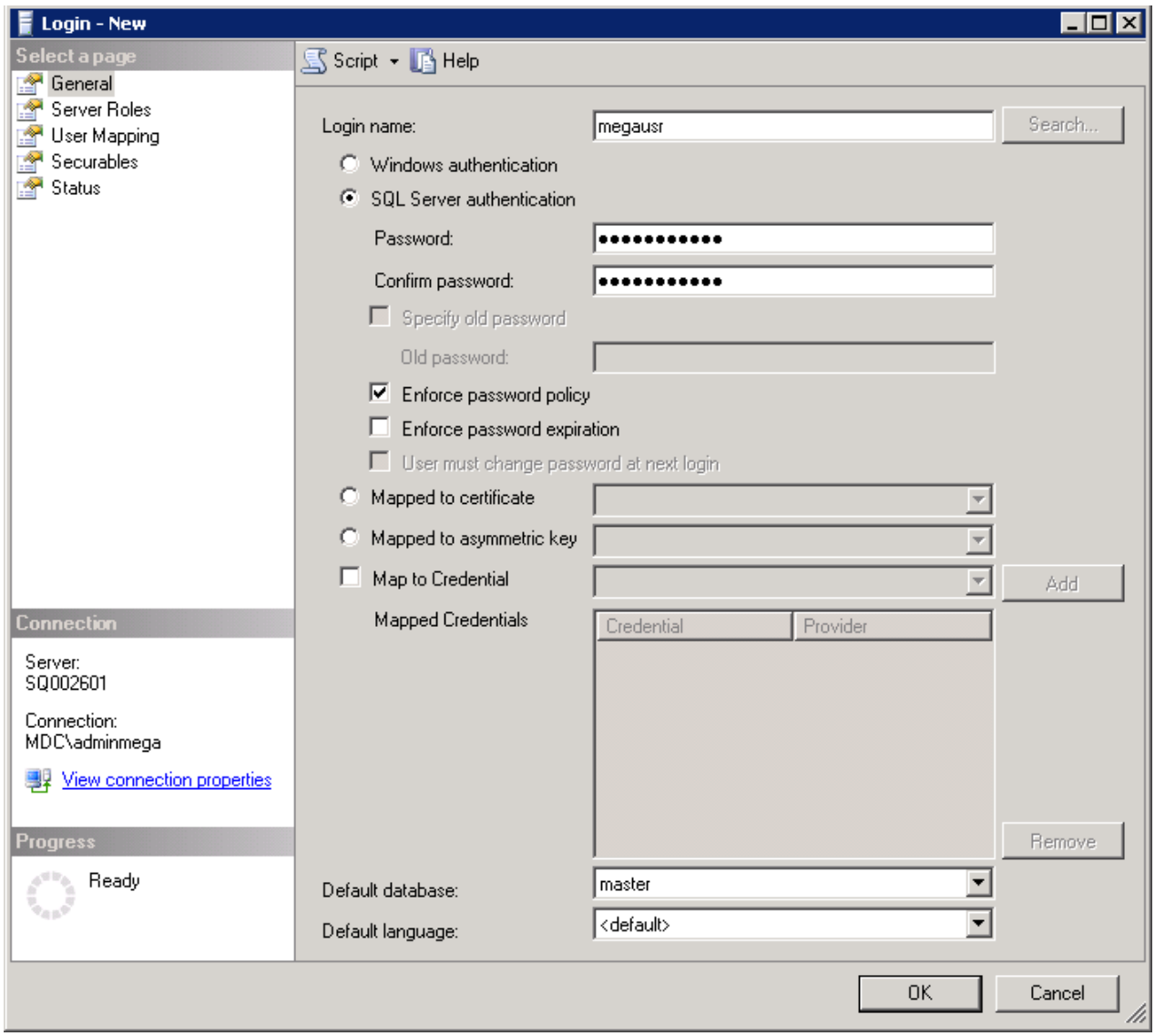


The **Login - New** window appears and displays and the **General** page.

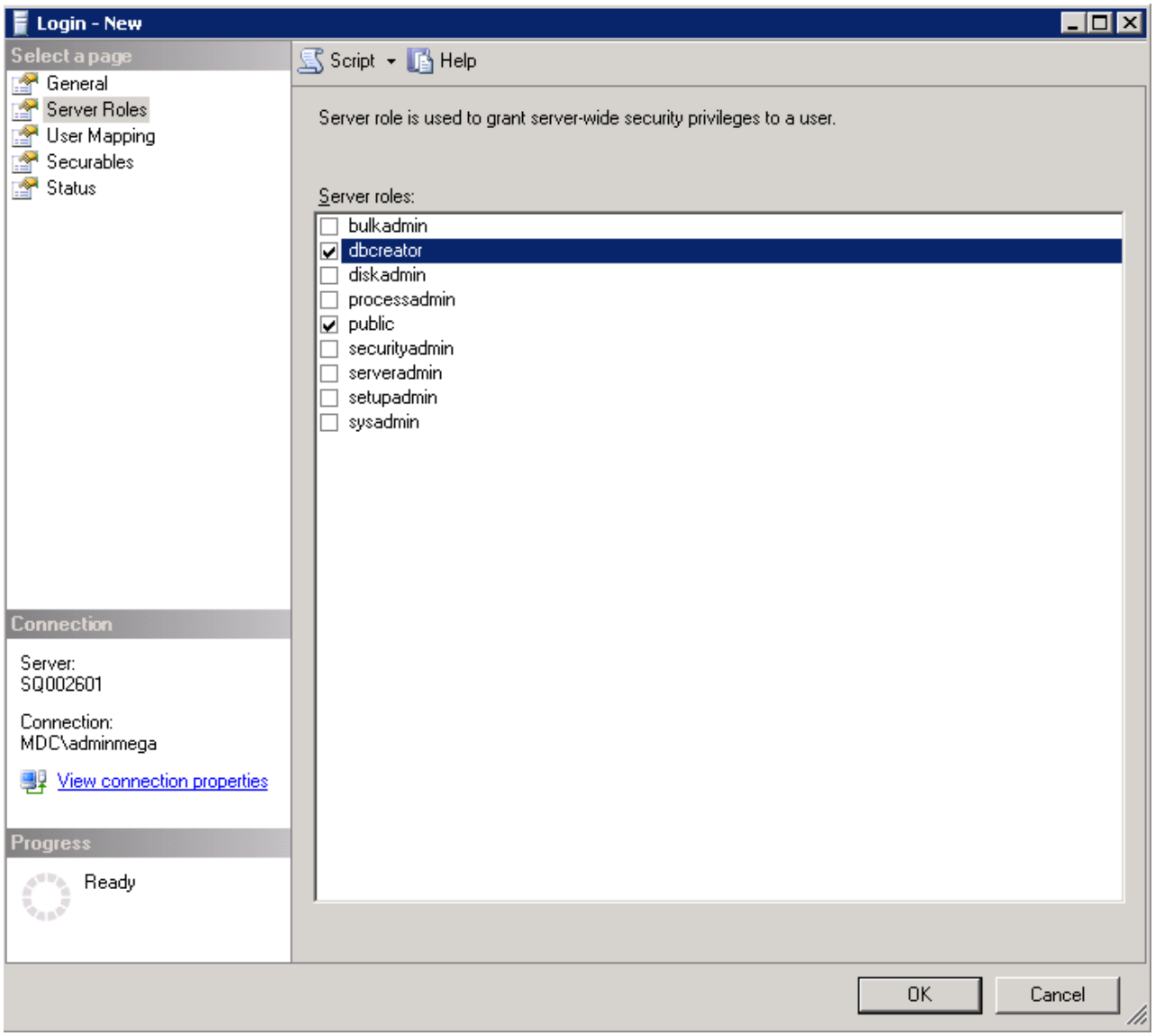
6. In the right pane:
   a. In the field **Login name** enter the login, for example « megausr ».
   b. Select **SQL Server authentication** mode.
   c. In the field **Password**, enter a password that complies with security requirements, for example :

      `Mega2k8!usr`

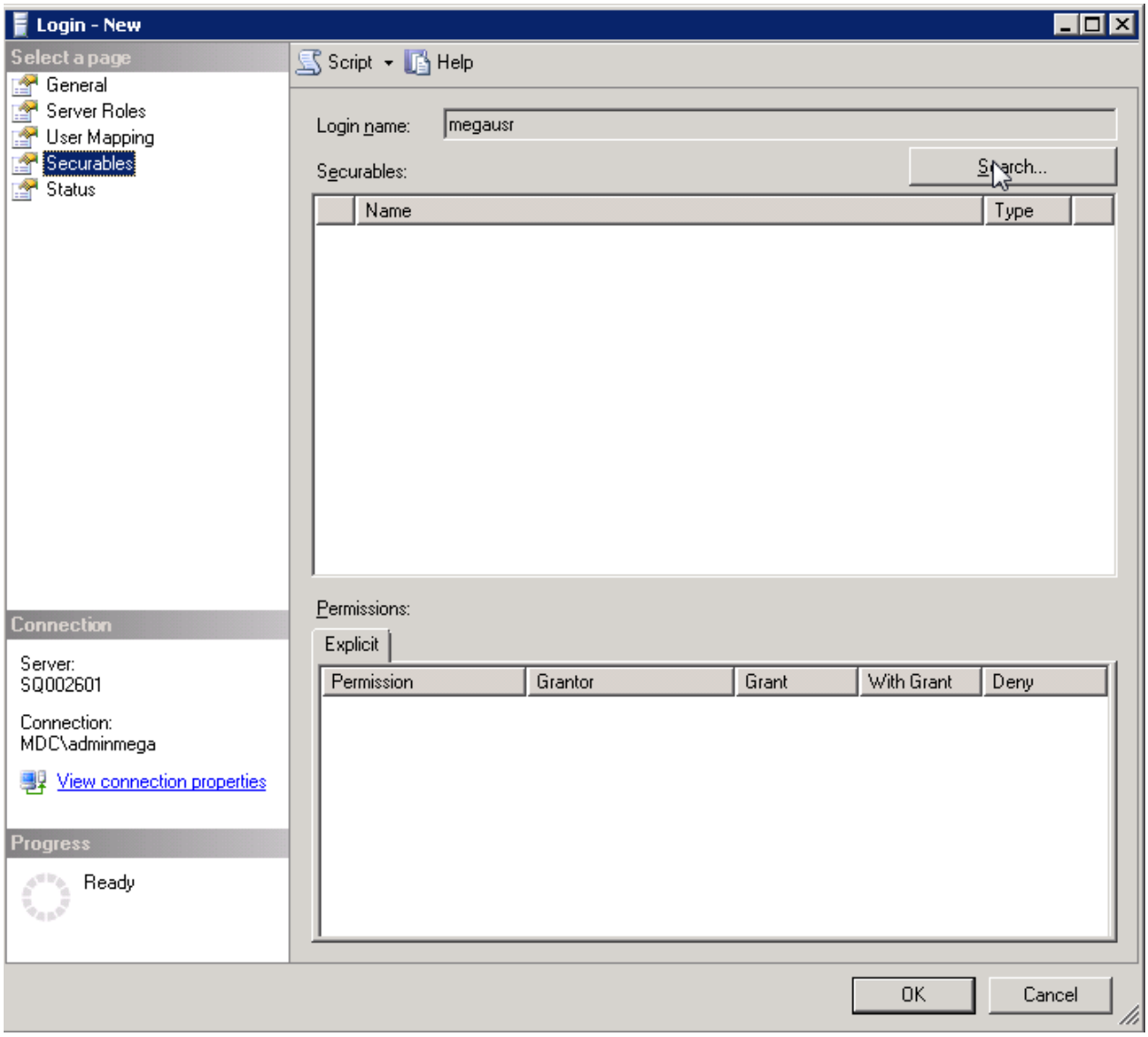
7. Unselect **Enforce password expiration** and User must change password at next login.



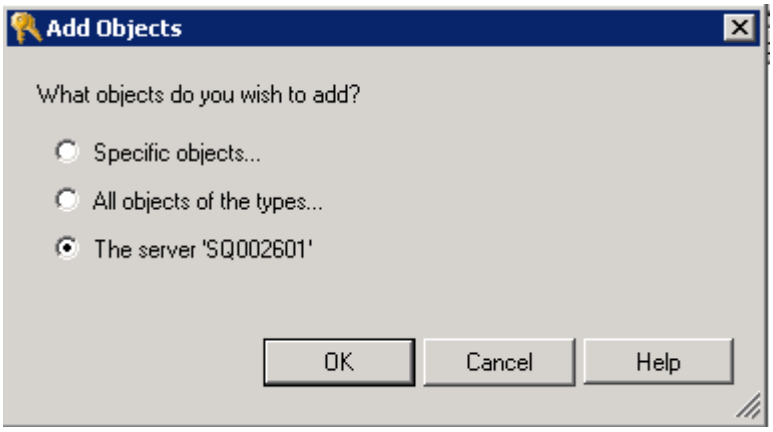
8. In the **Select a page** pane, select **Server Roles**.
9. In the right pane select **dbcreator**.



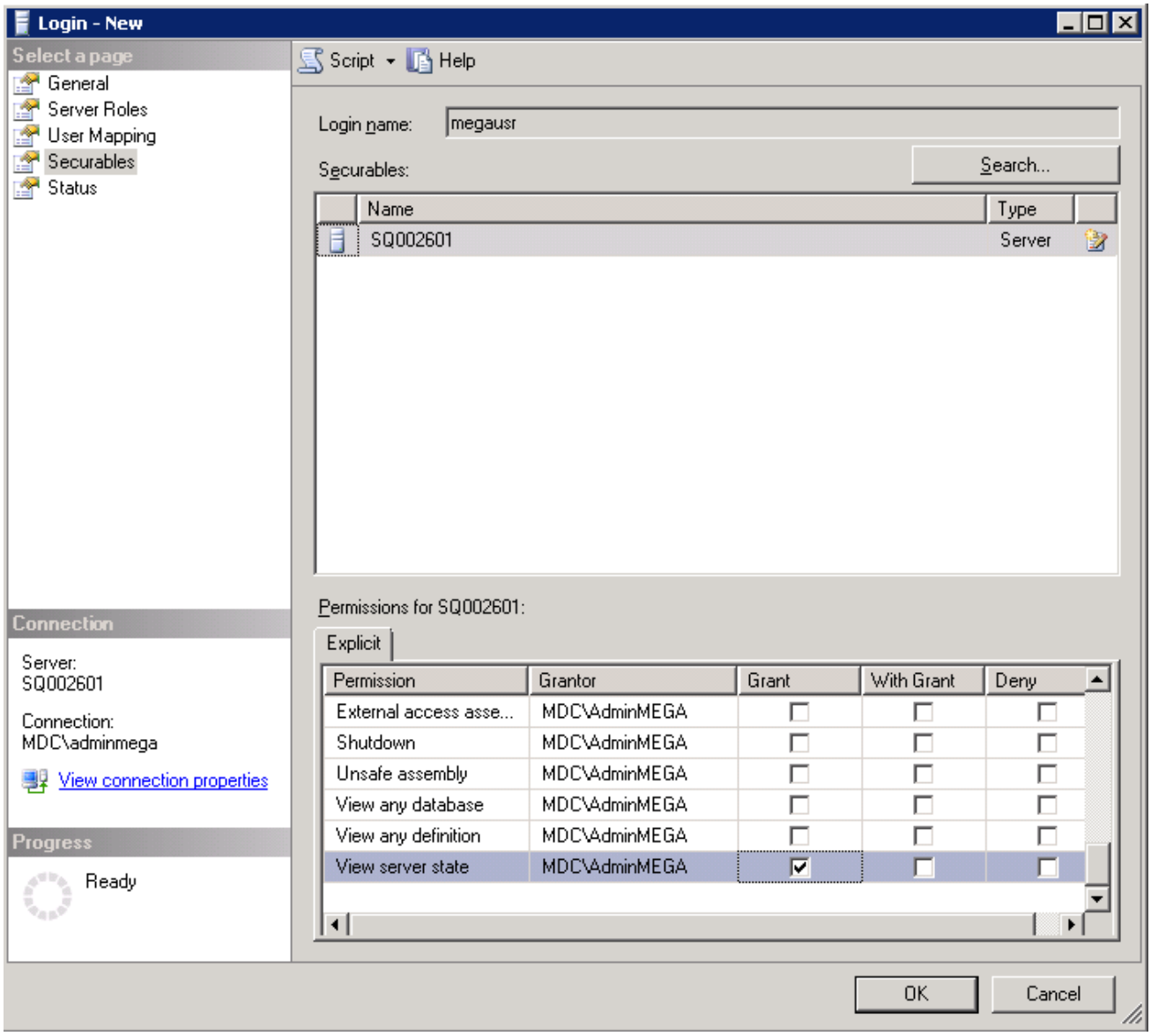
10. In the **Select a page** pane, select **Securables**.
11. In the right pane, click **Search**.



12. Select the instance to browser for its objects. Here we were on instance SQL002601 :



13. Click **OK**.
14. In the **Explicit** tab, for the **View Server state** permission select « Grant ».



15. Click **OK** to create the user.

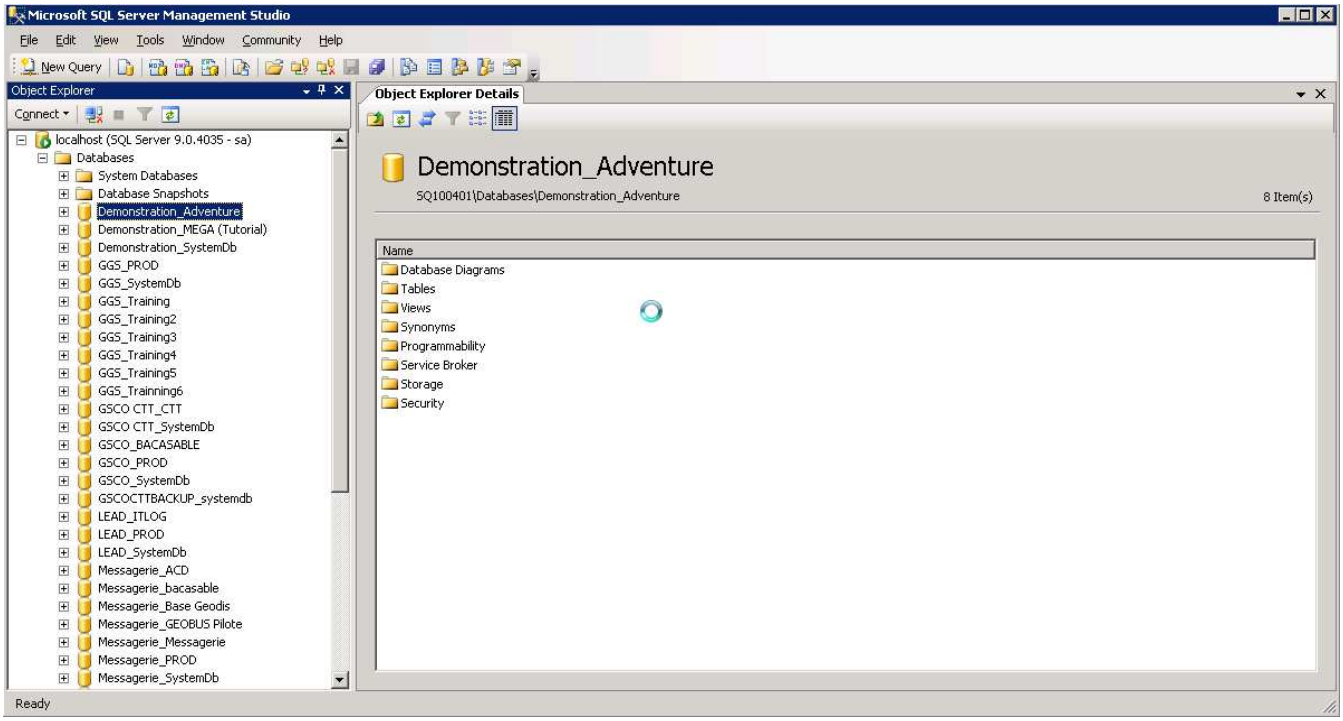
Backup and file transfer

1. Connect to the server hosting the source database.

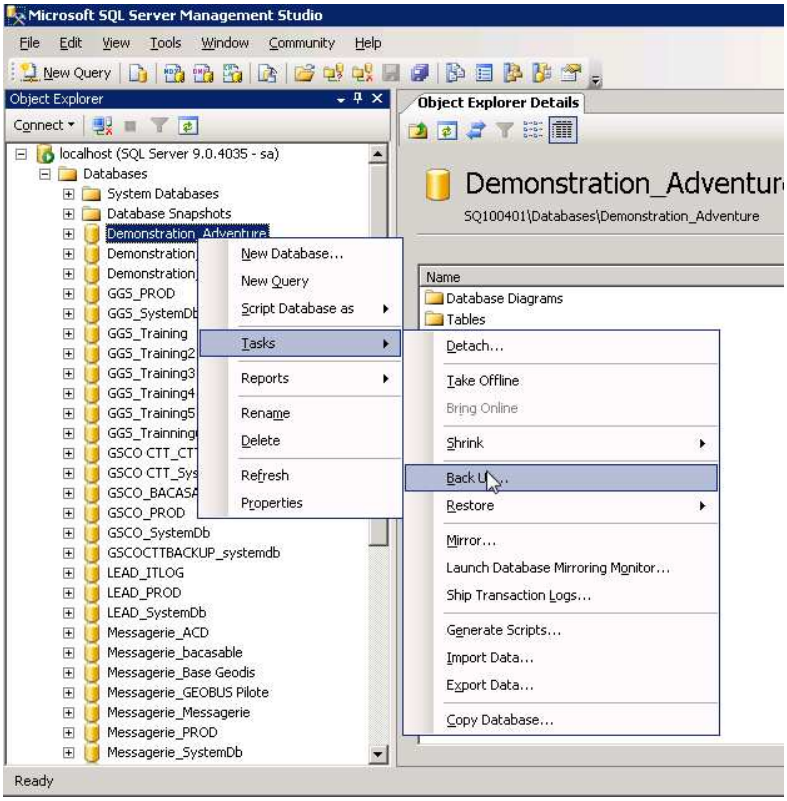
   Example: SQ100401.

2. Launch **Microsoft SQL Server Management Studio**.

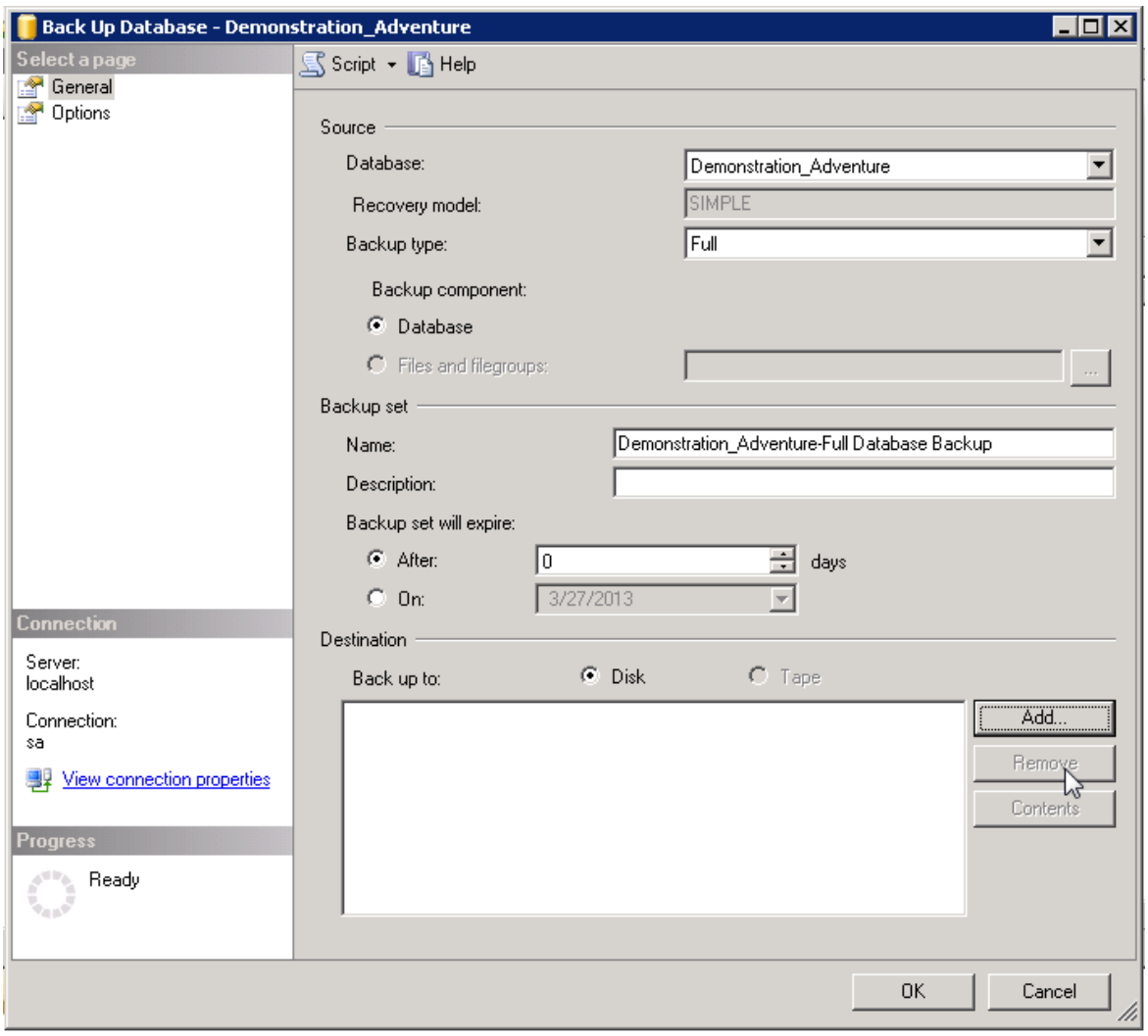
   If possible, connect with a Windows user that will have been granted sysadmin rights on the instance. Otherwise, you might encounter issues when creating the backup file to a specific location, or later, when you need to restore the database on the target instance.



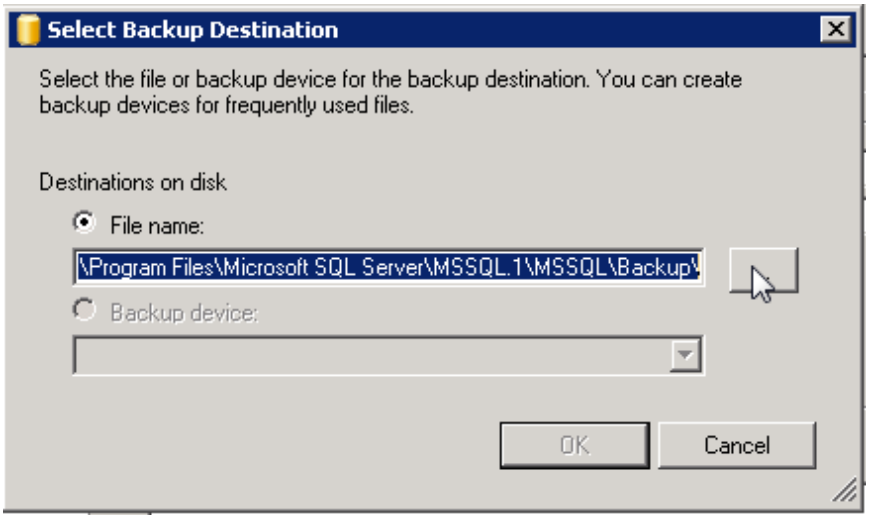
3. To make a backup of the source database (for example, Demonstration_Adventure), right-click the database and select **Tasks > Back Up**.



The **Back Up Database** window appears.

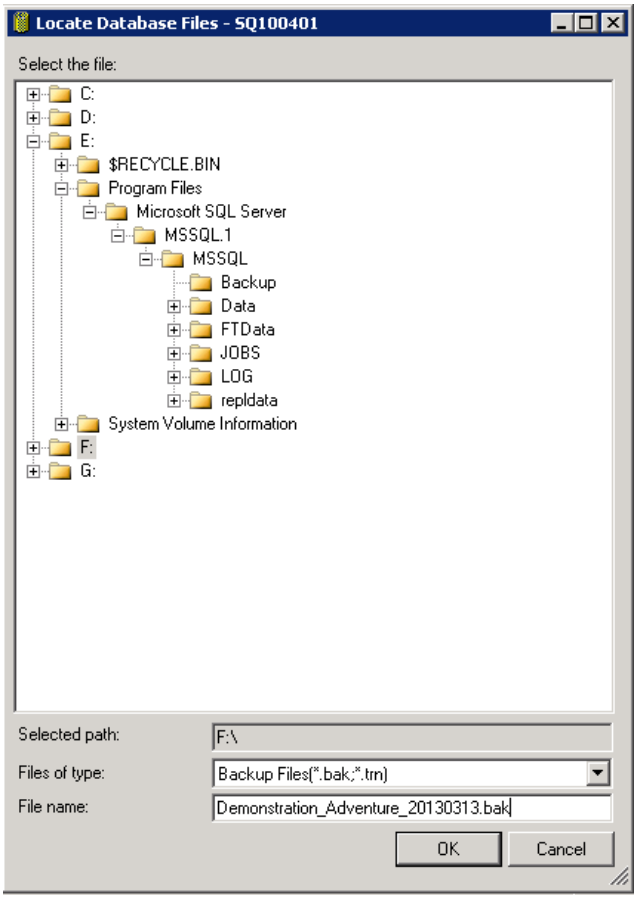


4. In the right pane, in the **Destination** pane, make sure that the destination list is empty. If it isn't, select each line and click **Remove**. Once it is empty, click **Add**.
5. In the **Select BackUp Destination**, click ....

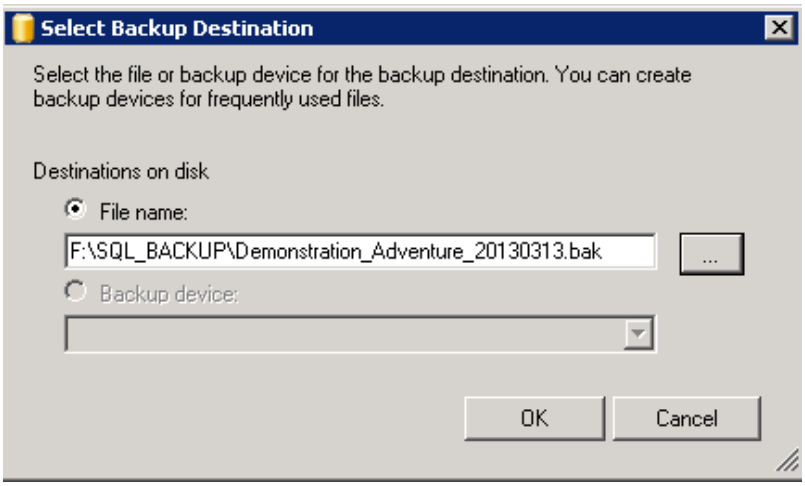


6. Choose a location where you know that the user you authenticated with, has rights to write on. In this example in « F:\SQL_BACKUP » of the F drive, and give the name of the backup file to create (here Demonstration_Adventure_20130313.bak).

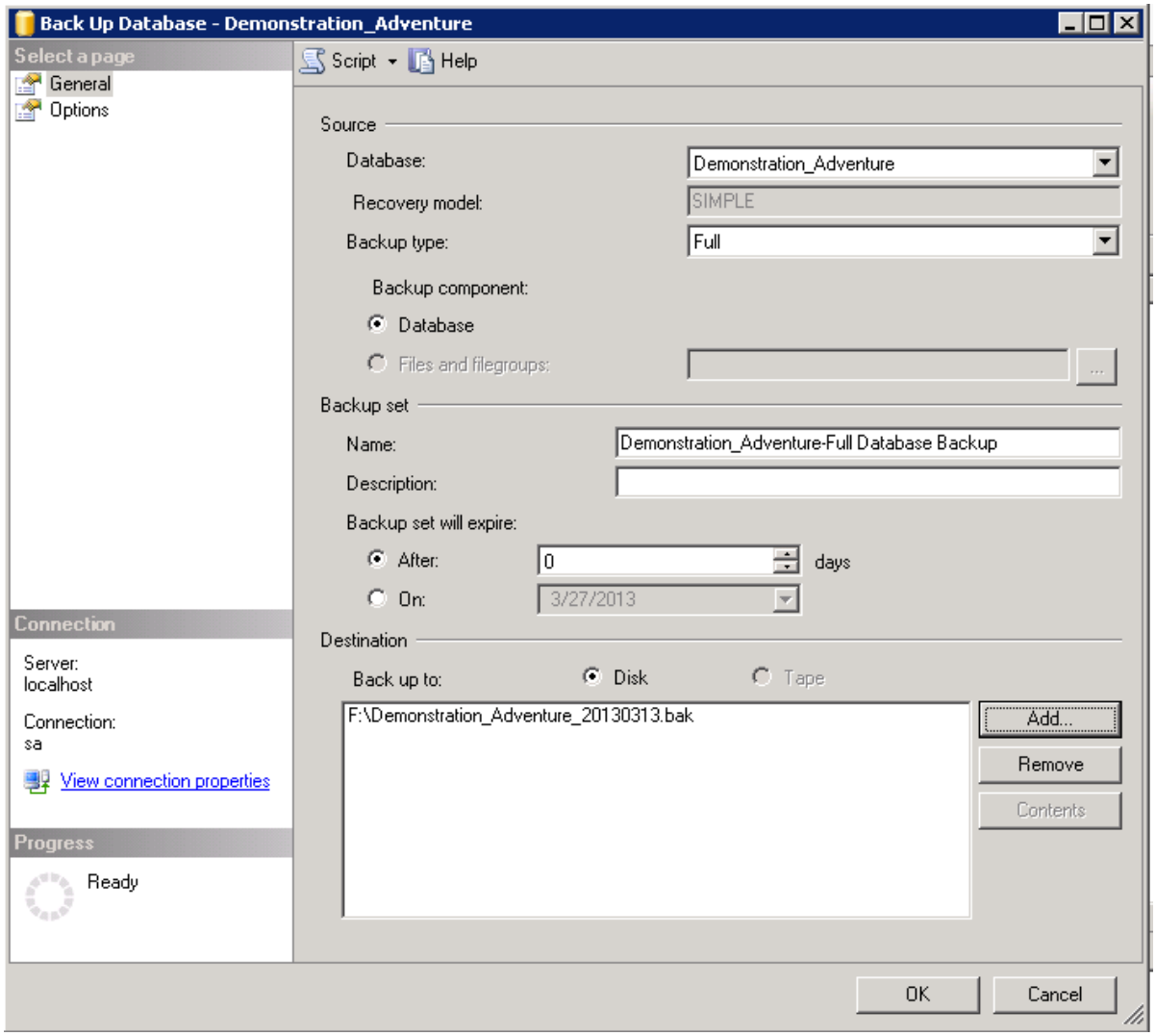
Please note that the known format of a full backup in SQL Server is **.BAK** files. You have to explicitely put it in the file name, otherwise it will not have any format.



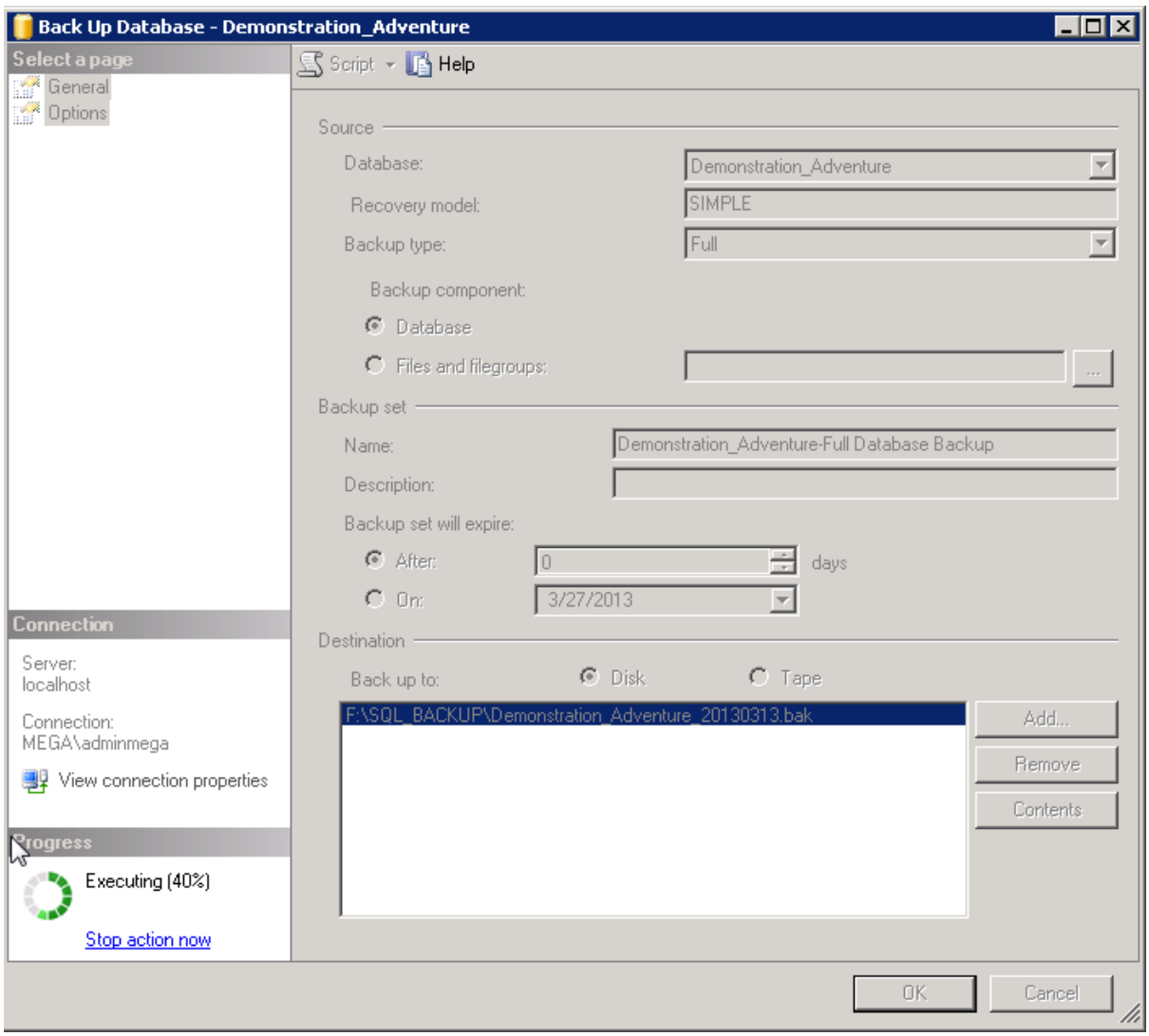
7. Click **OK**.



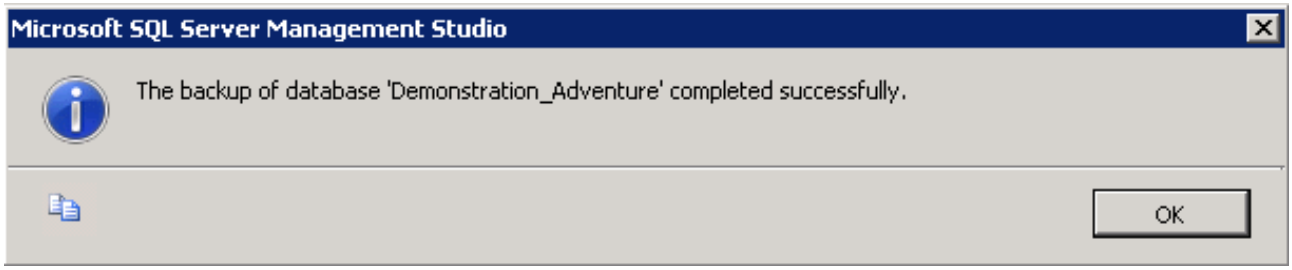
8. Click **OK**.



9. Click **OK** and check the progress of the restore by looking at the left-bottom section of this window (here, we are 40% done with the restore).



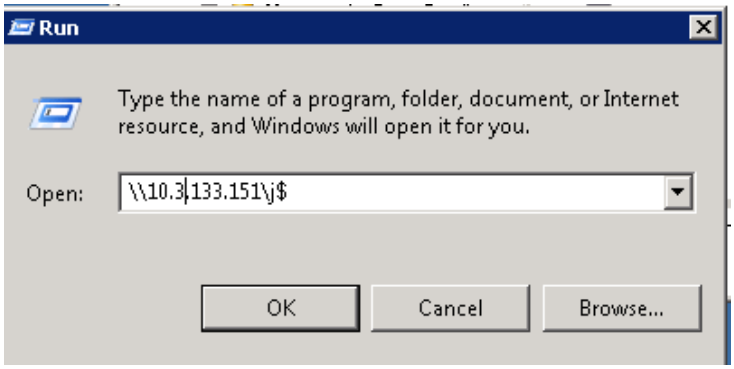
10. Check that the database was fully restored, and click **OK**.



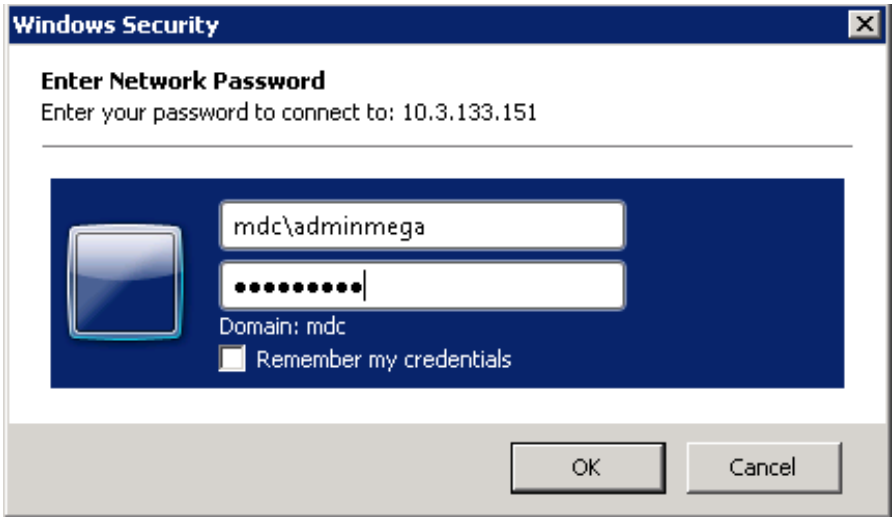
11. Transfer the backup file created on the SQL Server server, to the target server (for example, here it is SQL002601).

In this example, the drive hosting the databases on the target instance, as well as the daily backups, is the J drive.

We used its IP address instead of its name, as we were working over two different domains, that did not see each other.

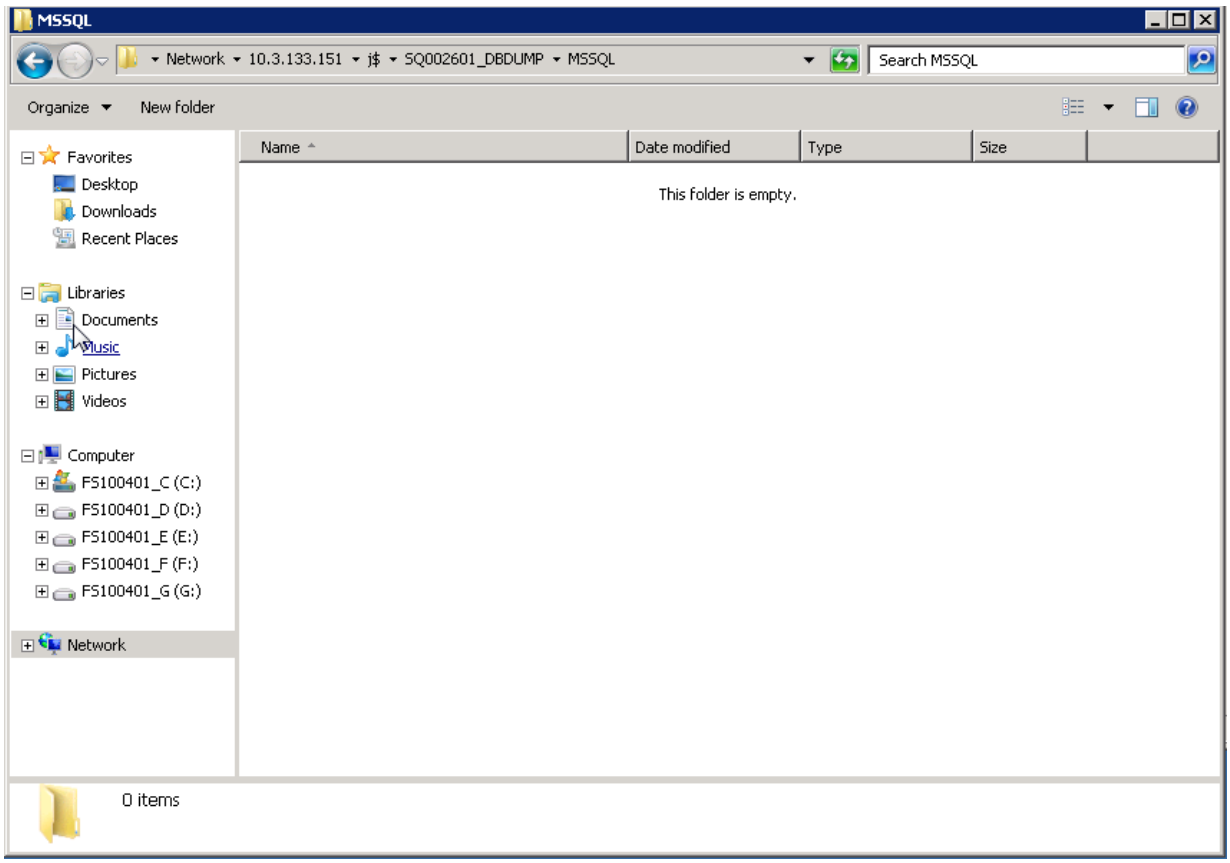


This is also the reason why we had to authenticate with another user.

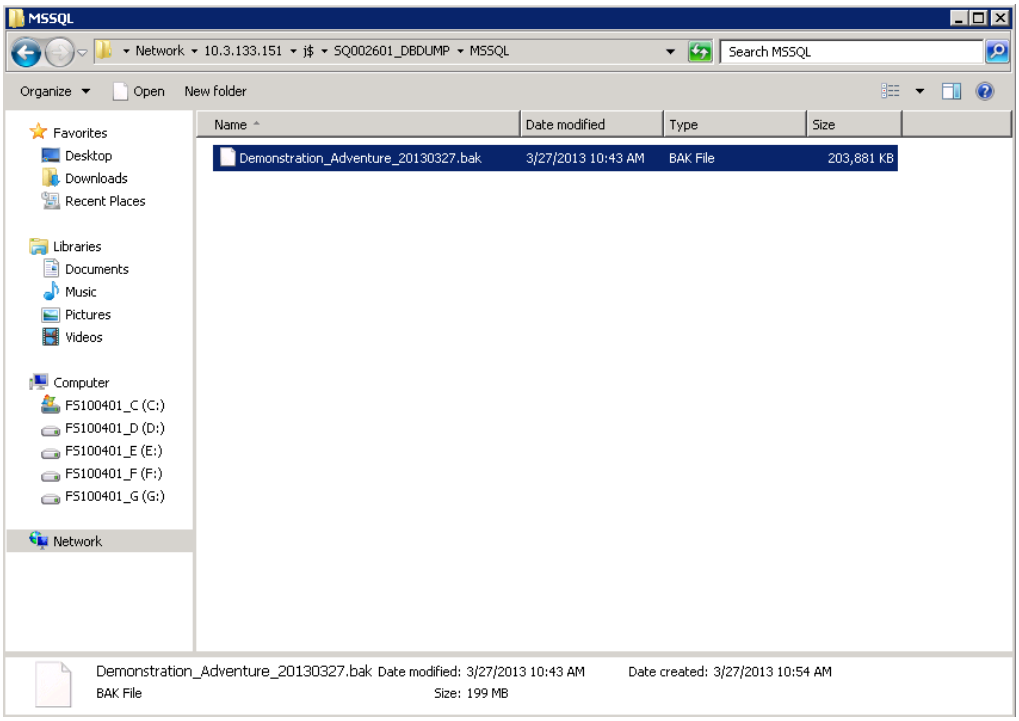


12. Go to the subfolder hosting the daily backups.

    (for example: « SQ002601_DBDUMP\MSSQL » on the J drive)

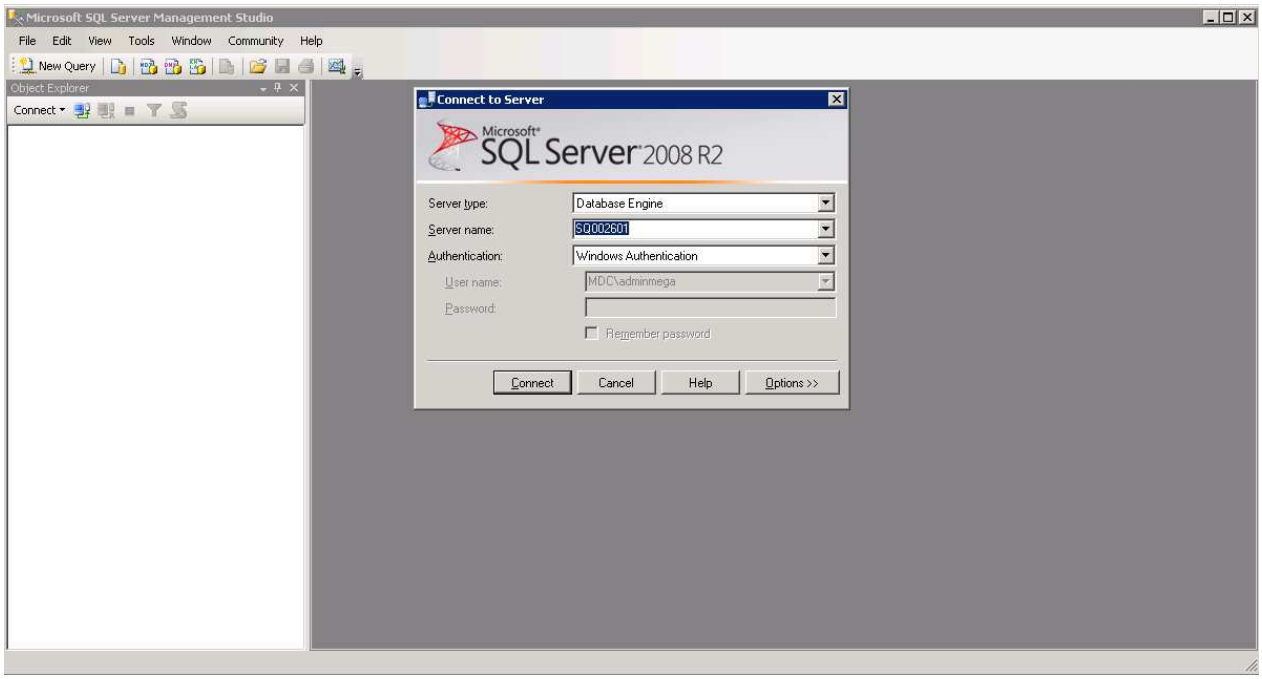


13. Copy the file from the source folder to this one.

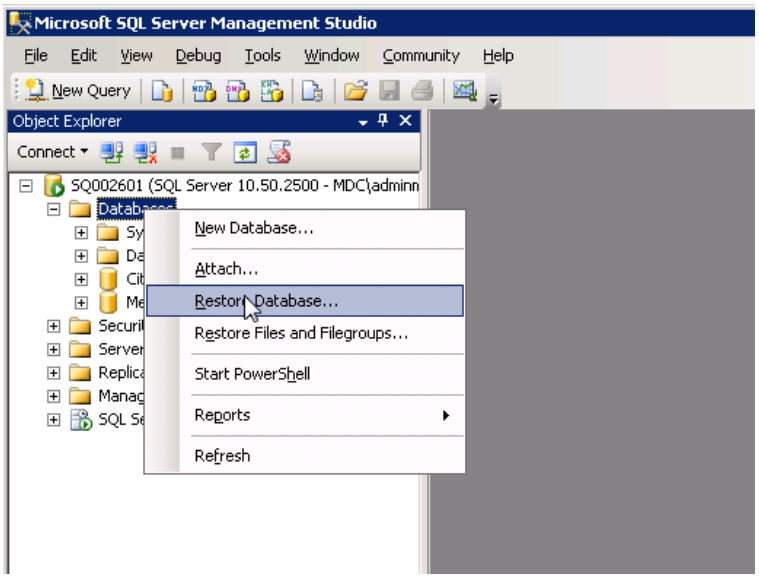


Restore

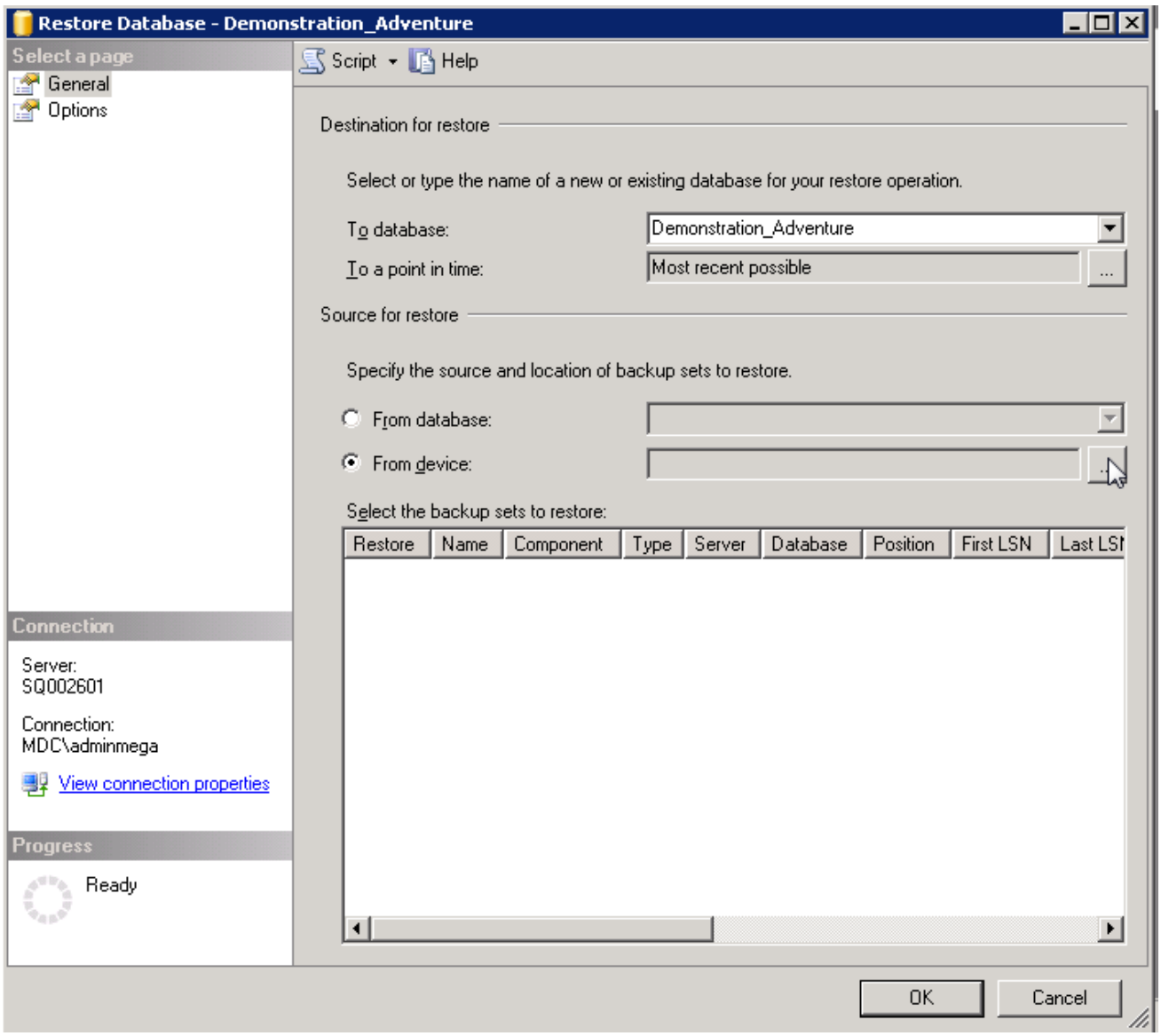
1. Connect to the target SQL Server server (example: SQL002601).
2. Launch the Management Studio, and connect to the instance using, if possible, a Windows account that is both sysadmin, and has access rights to the folder where the backup file was copied (example: mdc\adminmega).



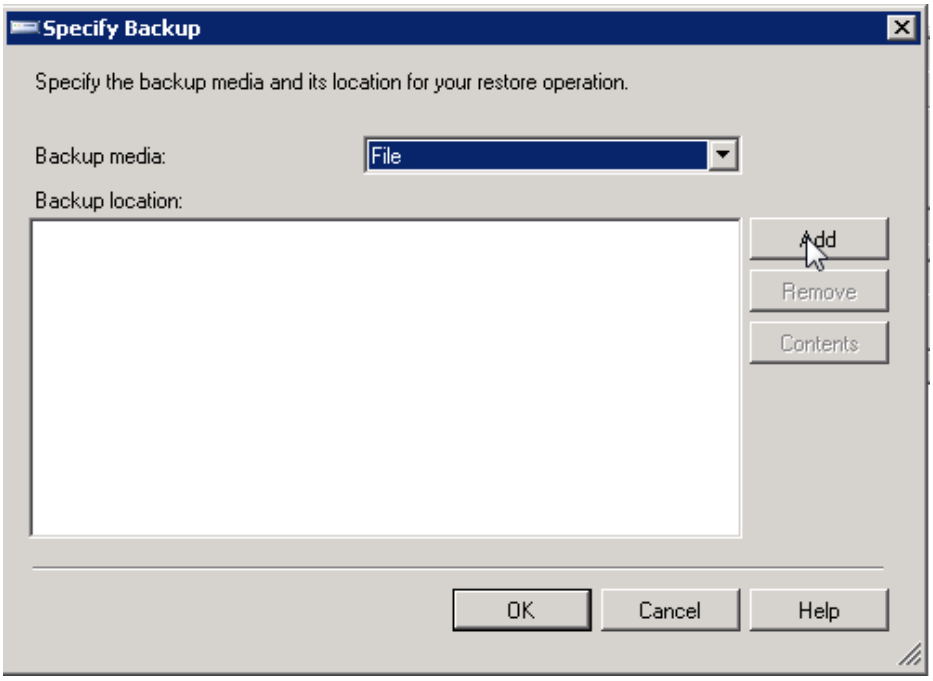
3. Right-click **Databases**, and select **Restore Database**.



4. From the **General** page, in the **Destination for restore** pane, in the **To database** field, provide the name of the database that will be created (example: Demonstration_Adventure).
5. In the **Source for restore** pane, select **From Device** option and click ....


Restore Database - Demonstration_Adventure
Select a page
General
Options
Script
Help
Destination for restore
Select or type the name of a new or existing database for your restore operation.
To database:
Demonstration_Adventure
To a point in time:
Most recent possible
Source for restore
Specify the source and location of backup sets to restore.
From database:
From device:
Select the backup sets to restore:
Restore
Name
Component
Type
Server
Database
Position
First LSN
Last LSN
Connection
Server:
SQ002601
Connection:
MDC\adminmega
View connection properties
Progress
Ready
OK
Cancel

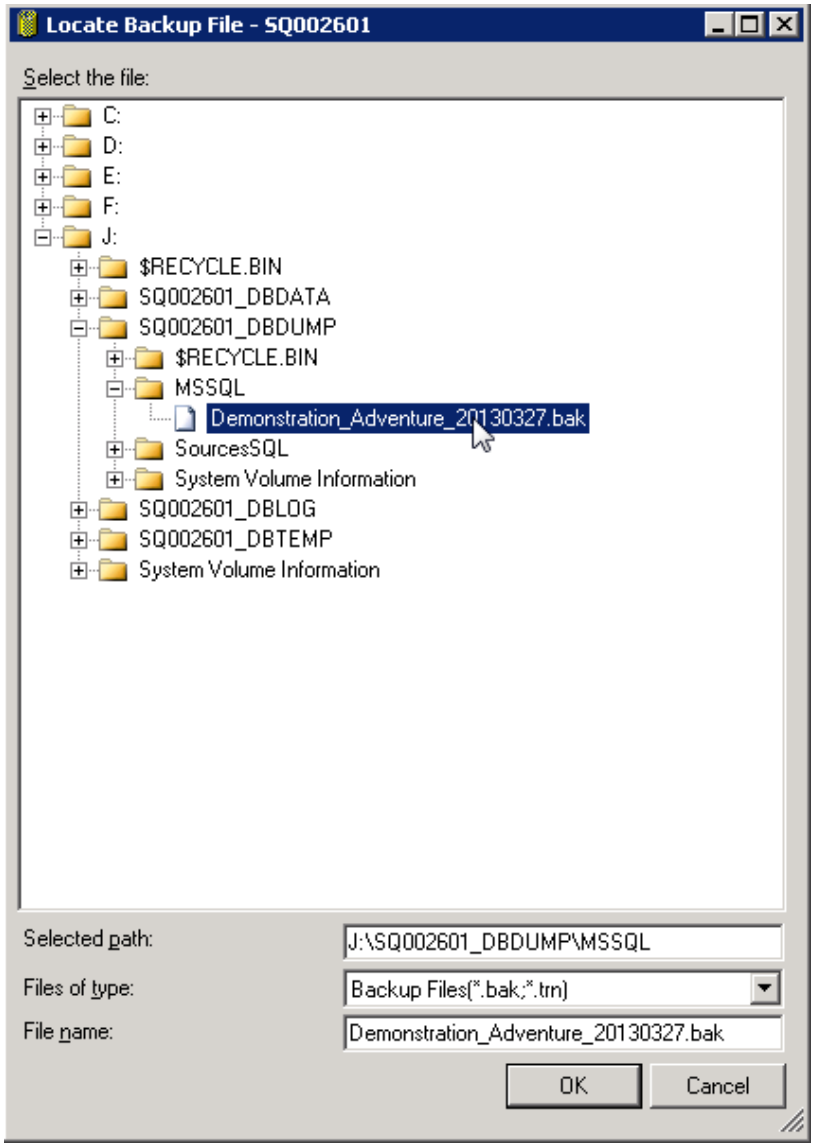
6. Click **Add**.



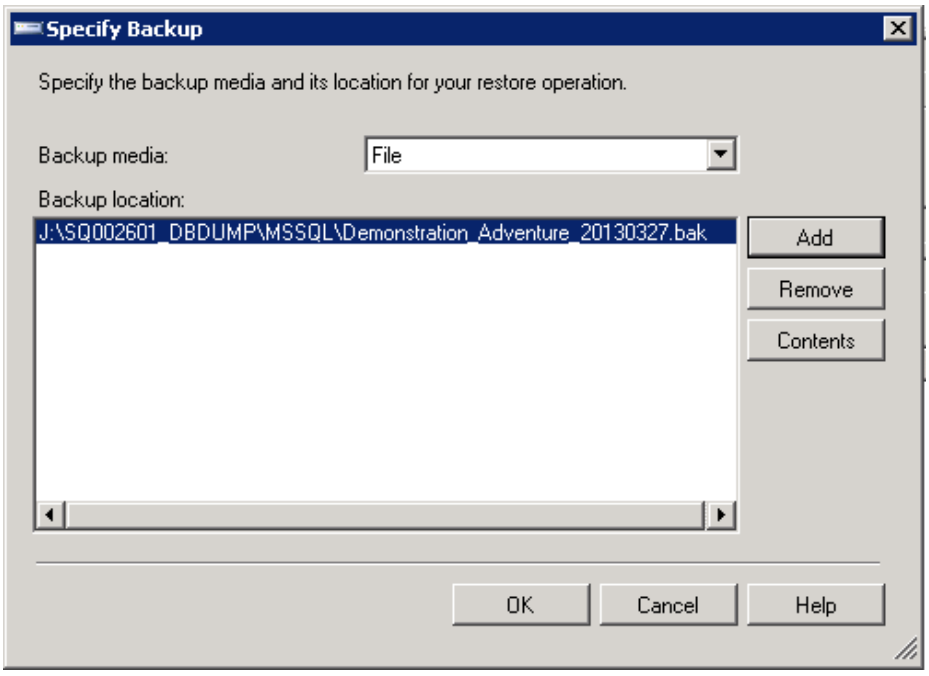
7. Check that we are correctly put in the folder « J:\SQ002601_DBDUMP\MSSQL».

   Otherwise, browse the folders to get to it. Then, click the .bak file that you just copied on the server (for example: « Demonstration_Adventure_20130327.bak »).

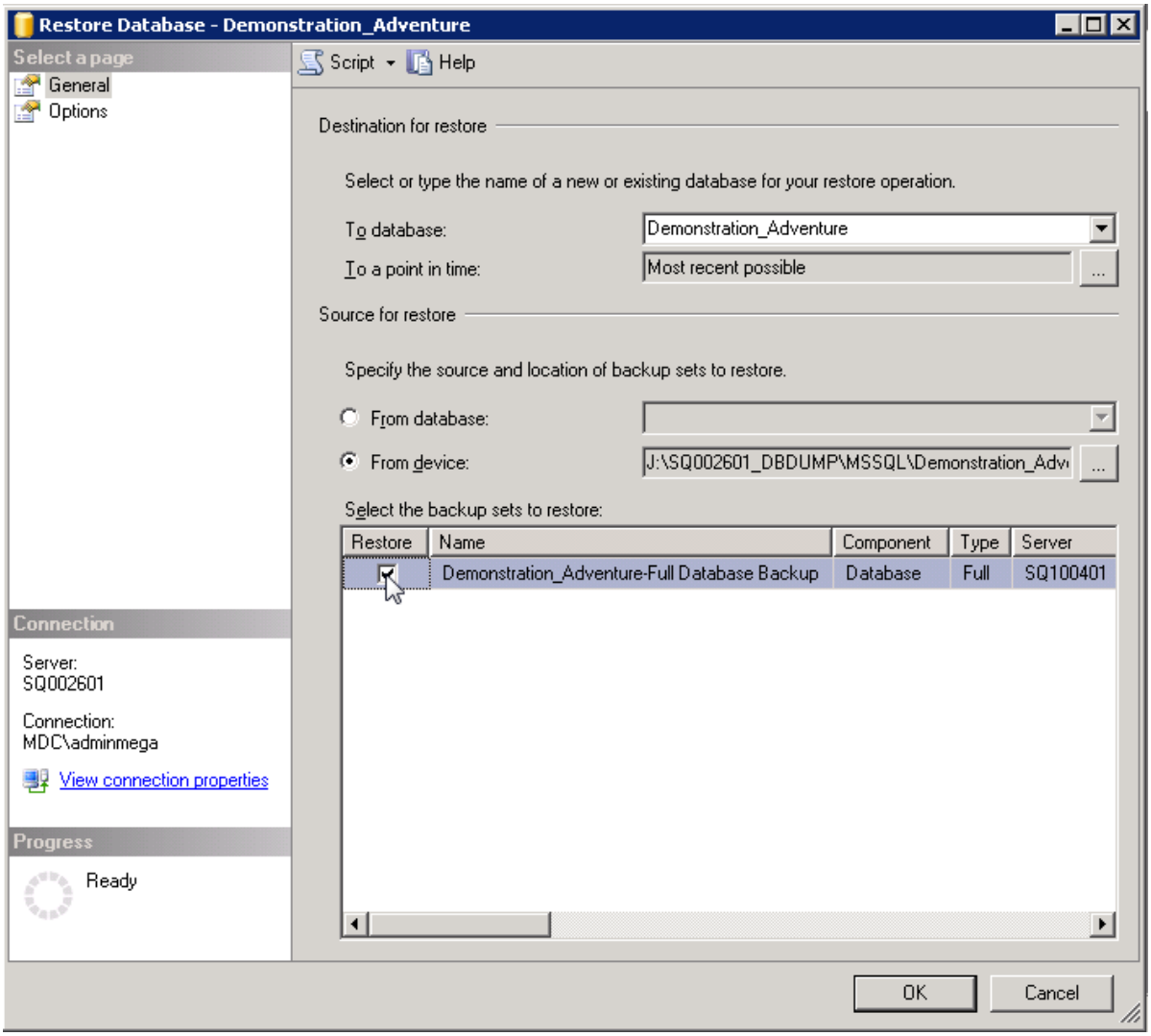
   Check that the « File Name » field is correctly filled, and click **OK**.


Locate Backup File - SQ002601
Select the file:
C:
D:
E:
F:
J:
$RECYCLE.BIN
SQ002601_DBDATA
SQ002601_DBDUMP
$RECYCLE.BIN
MSSQL
Demonstration_Adventure_20130327.bak
SourcesSQL
System Volume Information
SQ002601_DBLOG
SQ002601_DBTEMP
System Volume Information
Selected path:
J:\SQ002601_DBDUMP\MSSQL
Files of type:
Backup Files(*.bak;*.trn)
File name:
Demonstration_Adventure_20130327.bak
OK
Cancel

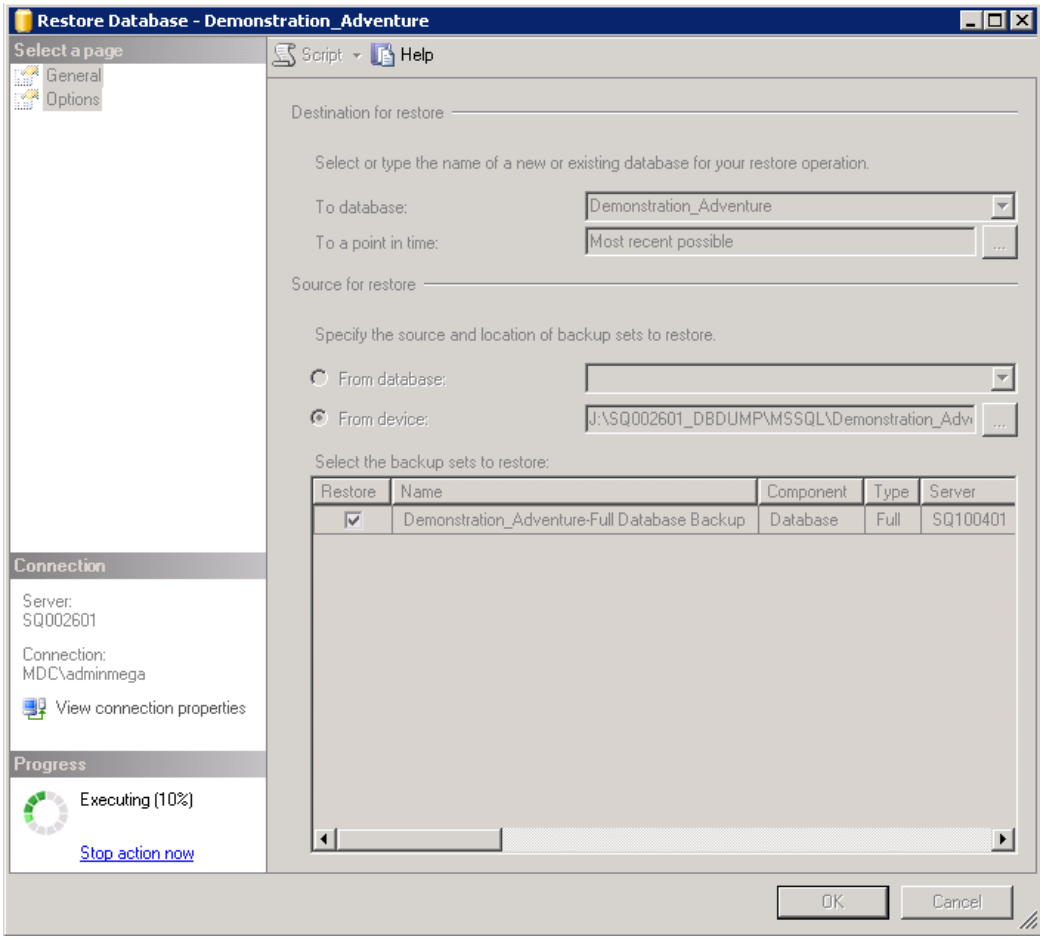
8. Click **OK**.


Specify Backup
Specify the backup media and its location for your restore operation.
Backup media:
File
Backup location:
J:\SQ002601_DBDUMP\MSSQL\Demonstration_Adventure_20130327.bak
Add
Remove
Contents
OK
Cancel
Help

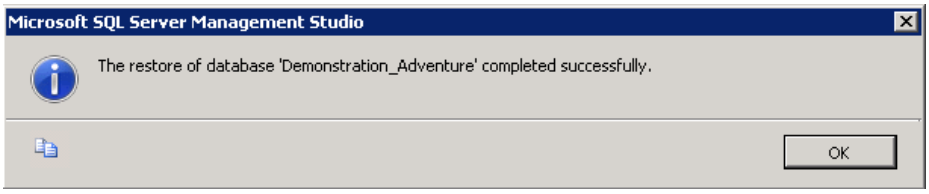
9. In the **Source for restore** pane, in the **Select the backup sets to restore** table select **Restore** at the beginning of the line.



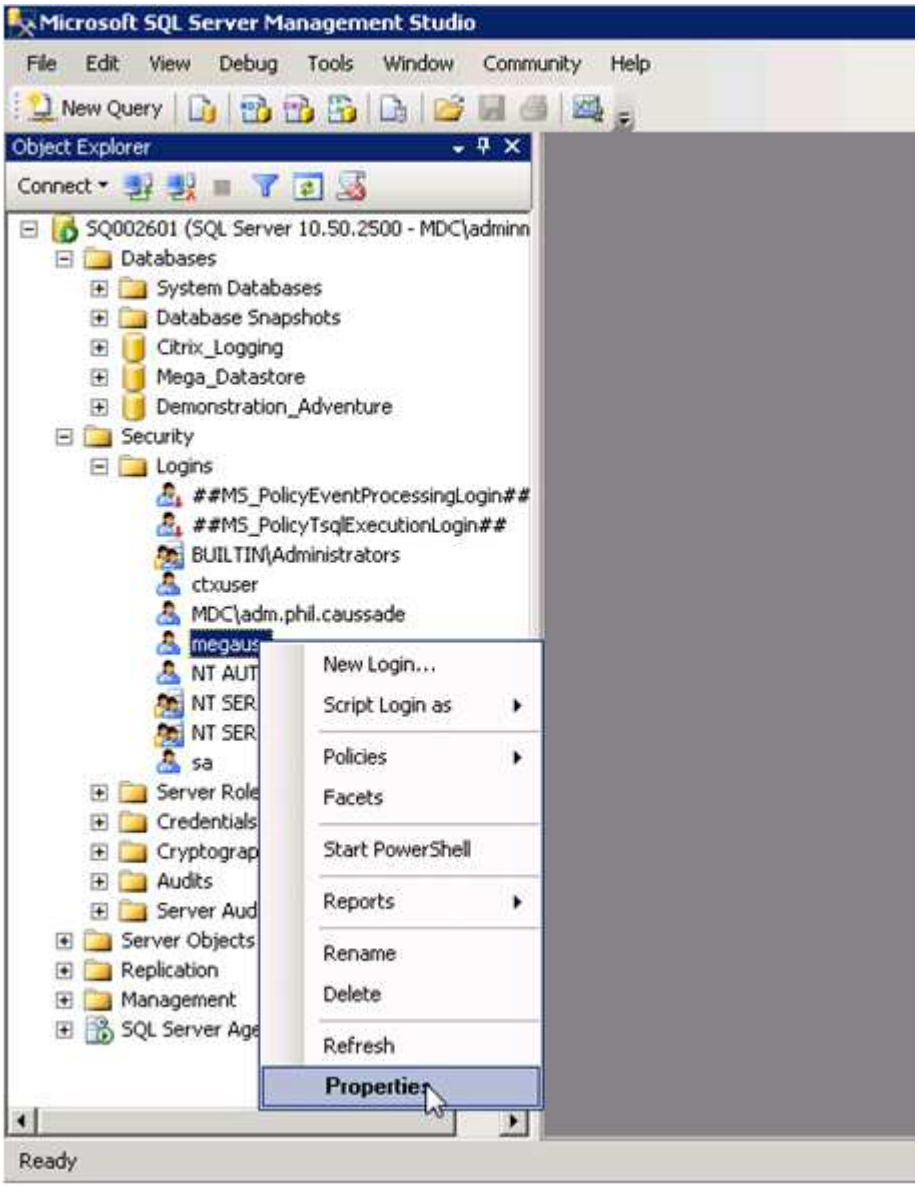
10. click **OK**.



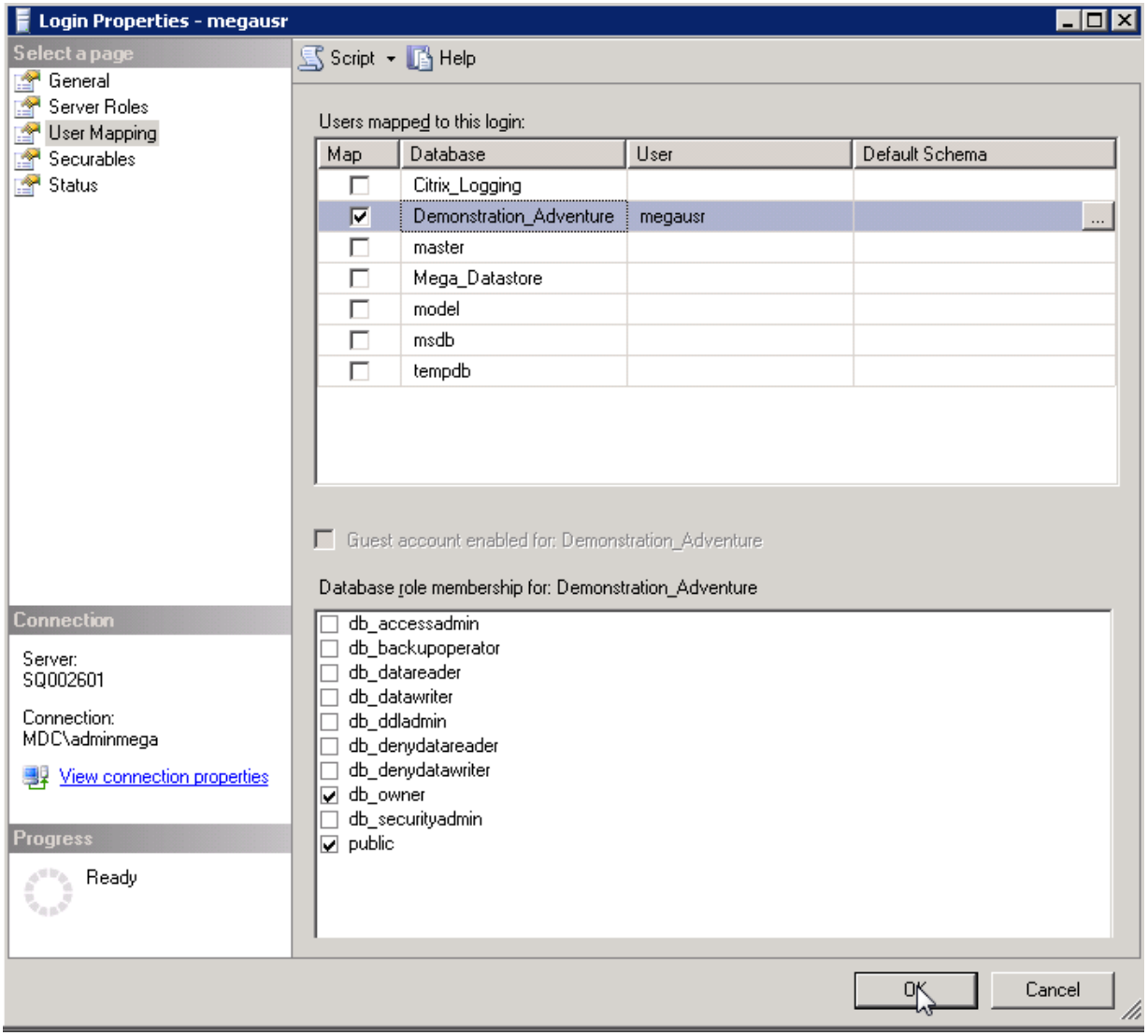
11. Once the restore completed successfully, click **OK**.



12. In ManagementStudio, expand **Security** folder, and **Logins** folder.
13. Right-click the account that you will use to connect the application to SQL Server (for example: login « megausr ») and select **Properties**.



14. In the **Login Properties - <login>** window, select **User Mapping** page, and select the **Map** corresponding to the database lin you just restored.
15. In the **Database role membering for: <database name>** pane, select « db_owner ».
16. Click **OK**.



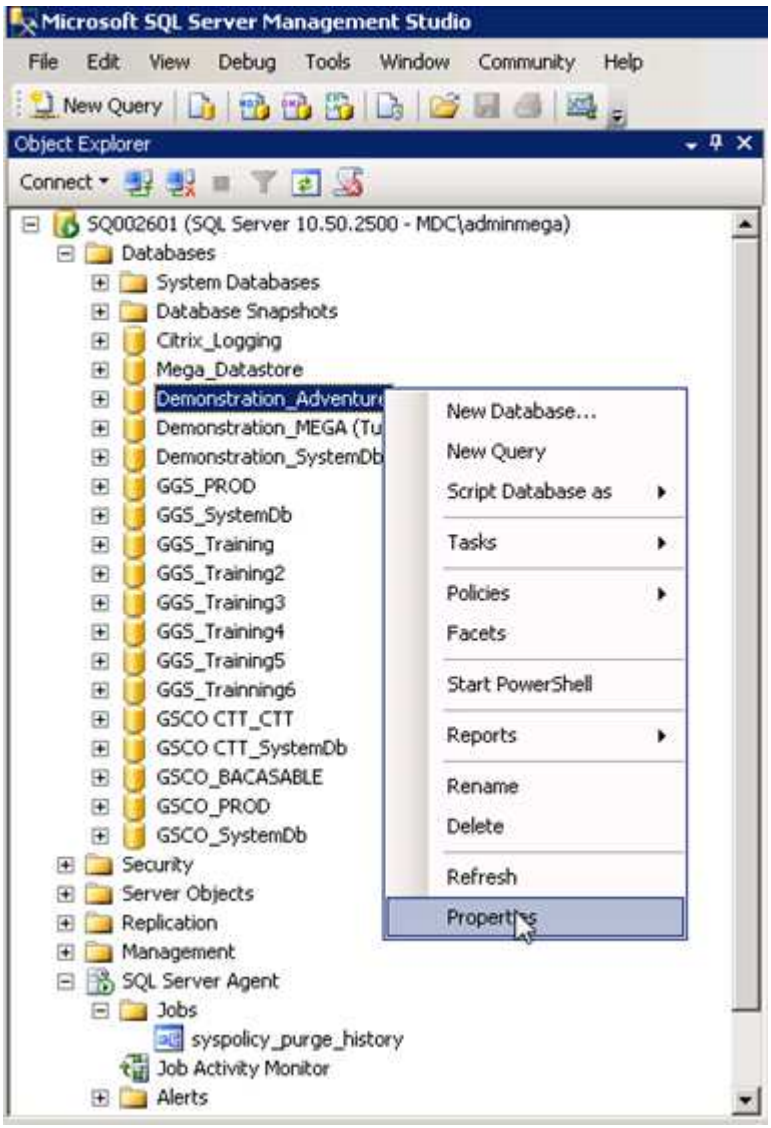
IF you moved from a version of SQL Server to a more recent one, you have to upgrade the compatibility level of your database.

> In this example, the source database came from an SQL Server 2005 instance, and was restore on an SQL Server 2008 instance.

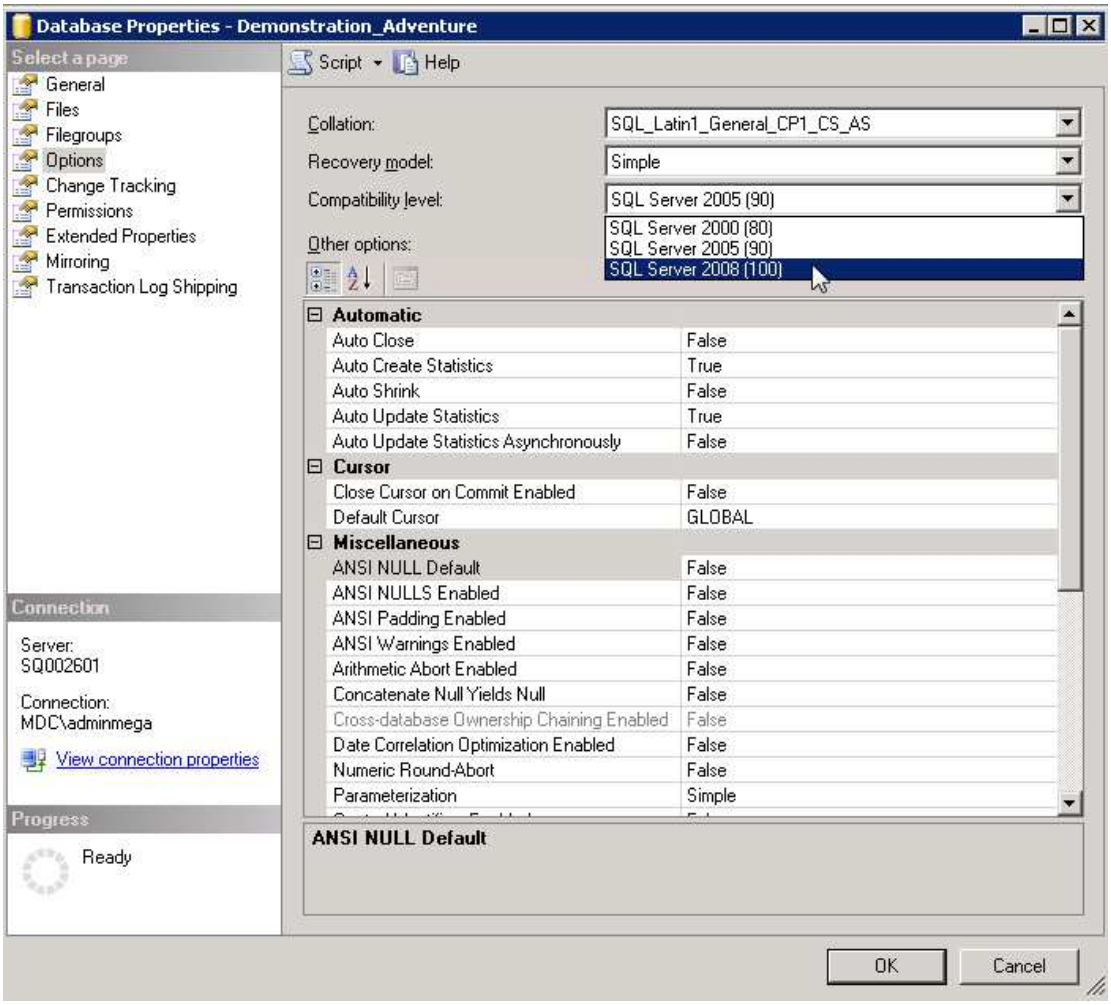
If you stay on the same version, go directly to the next section.

To upgrade:

1. Right-click the database and select **Properties**.



2. Select the **Options** page, in the **Compatibility level** drop down list select the appropriate version (for example « SQL Server 2008 (100) »).


Database Properties - Demonstration_Adventure
Select a page
General
Files
Filegroups
Options
Change Tracking
Permissions
Extended Properties
Mirroring
Transaction Log Shipping
Script
Help
Collation:
SQL_Latin1_General_CP1_CS_AS
Recovery model:
Simple
Compatibility level:
SQL Server 2005 (90)
SQL Server 2000 (80)
SQL Server 2005 (90)
SQL Server 2008 (100)
Other options:
Automatic
Auto Close
False
Auto Create Statistics
True
Auto Shrink
False
Auto Update Statistics
True
Auto Update Statistics Asynchronously
False
Cursor
Close Cursor on Commit Enabled
False
Default Cursor
GLOBAL
Miscellaneous
ANSI NULL Default
False
ANSI NULLS Enabled
False
ANSI Padding Enabled
False
ANSI Warnings Enabled
False
Arithmetic Abort Enabled
False
Concatenate Null Yields Null
False
Cross-database Ownership Chaining Enabled
False
Date Correlation Optimization Enabled
False
Numeric Round-Abort
False
Parameterization
Simple
ANSI NULL Default
Connection
Server:
SQ002601
Connection:
MDC\adminmega
View connection properties
Progress
Ready
OK
Cancel

3. Click **OK**.

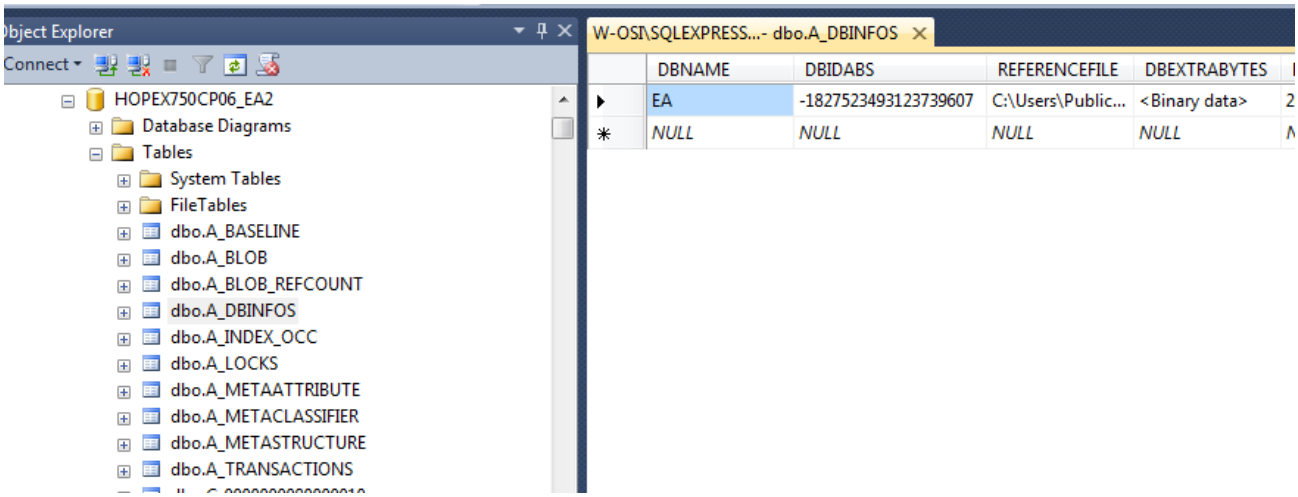
When duplicating a repository in an environment – Expert mode

This section is only if you are very confident about your HOPEX and RDBMS skills, since you have to modify data directly in some tables of your database.

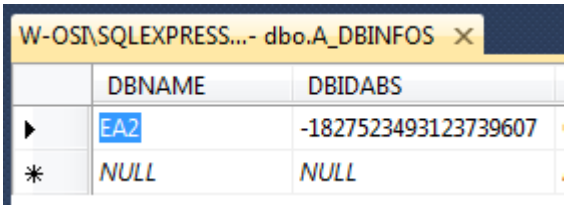
In case you want, within an environment, duplicate a specific repository, you need to tweak the data manually in two tables of your restored database.

For example: you want to duplicate repository "EA" to have a repository "EA2" in the environment "HOPEX750CP06".

1. You have restored database EA twice, in databases "HOPEX750CP06_EA" and "HOPEX750CP06_EA2".
2. Using SQL Server Management Studio, open database "HOPEX750CP06_EA2", and edit the table "dbo.A_DBINFOS":

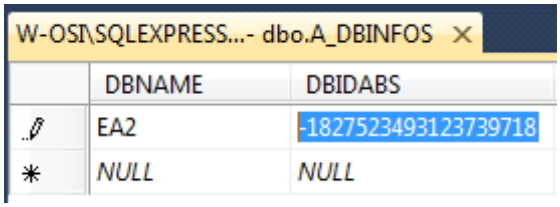


3. Modify the **DBNAME** field of that line, to enter the name of your duplicate, i.e. how you want it to appear in your environment (for example: here, "EA2").



4. Modify the **DBIDABS** field that is the unique identifier of the repository within your environment.

   You need to change its value, and make sure that it is unique for all other repositories, so you need to check in all schemas. Make sure that when you change one or two characters, it will create a string that is not used by any other (for example: here, we modified the last 3 digits from " 607" to "718").



5. Save your updates and close the table.

POST INSTALLATION TASKS

This section describes the tasks you need to carry out after duplicating the HOPEX data that were stored in the SQL Server database.

The first part details what to do to reattach to the set of duplicated data from HOPEX point of view.

As all is not stored inside the SQL Server, you need to copy from the original HOPEX environment to the new HOPEX environment pointing at the duplicate SQL Server data.

Create/Attach an environment in HOPEX

Creating an environment

To create an environment:

1. Connect to **HOPEX Administration**.
2. In the navigation tree, right-click **Environments** and select **New**.
3. Enter the **Name** of the new environment.

   The environment name must respect Windows folder naming constraints.

   It is the beginning of the name of the system database. It should be called <environment>_SystemDb, where <environment is the string that you enter in the HOPEX admin tool. In this example, the environment is called "GGS".
4. (Optional) Location of the environment is specified by default; you can modify it if necessary using the **Browse** button.

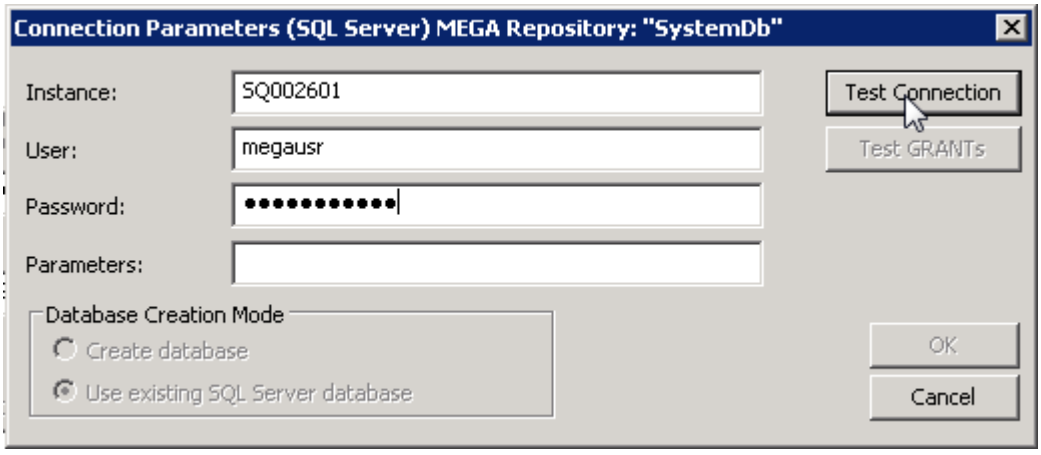
   In this example, we created a share on the server hosting the SQL Server instance, that will host the environment files, and that is called \\sq002601\EnvironnementsMega.

   The RDBMS repository server type for the new environment is already selected: SQL Server.
5. Select **Restore** which allows to connect to an existing SQL Server database on an instance.
6. Click **OK**.
7. Enter the connection parameters for the **Instance**.

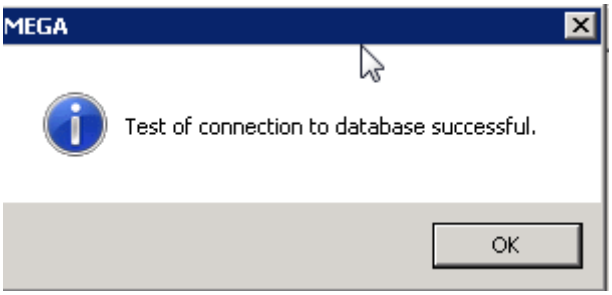
   Do not enter anything in the **Parameters** field.

   Note:

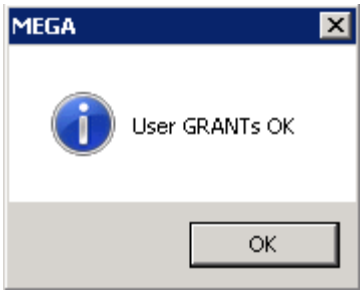
   The syntax for a named instance on **SQL Server** is: server_name\instance_name. In this example, the instance is the default one, without a specific name. That is why we only provide the name of the server where the instance is running.



8. Click **Test Connection**.

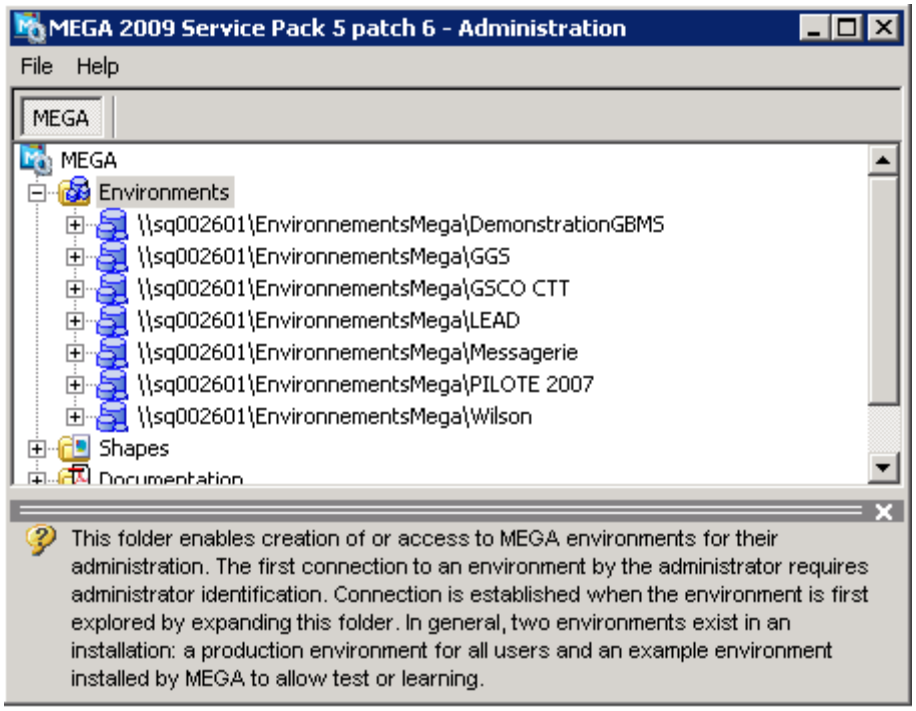


9. Click **Test GRANTS**.



10. Click **OK**.

    A popup indicating that the environment was successfully created appears, and the new environment is displayed in the list (for example here « \\sql002601\EnvironmentsMega\GGS).



Attaching the working database(s) to the environment

Once your environment is created, you still have to attach the working database(s) to this environment.

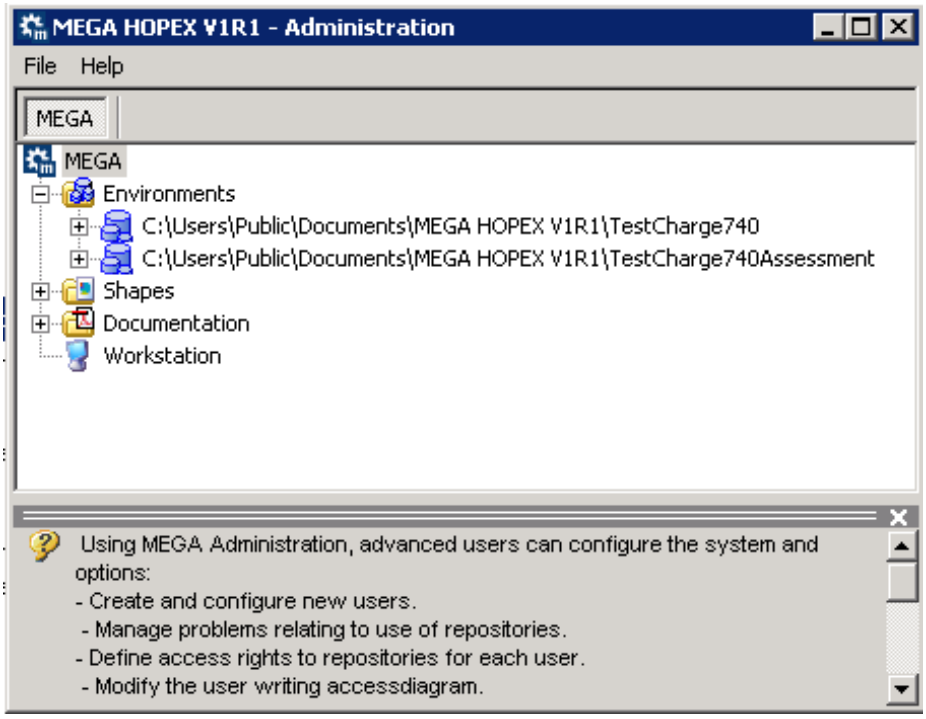
In **Repositories** you only have the SystemDb.

Now you have to connect to the environment, and add the HOPEX repositories. The first time you connect, you receive warning message(s) telling you that database "X" ("X" being the name of the repository that existed in the source environment, and most likely one of the repositories you are trying to attach) is not referenced.

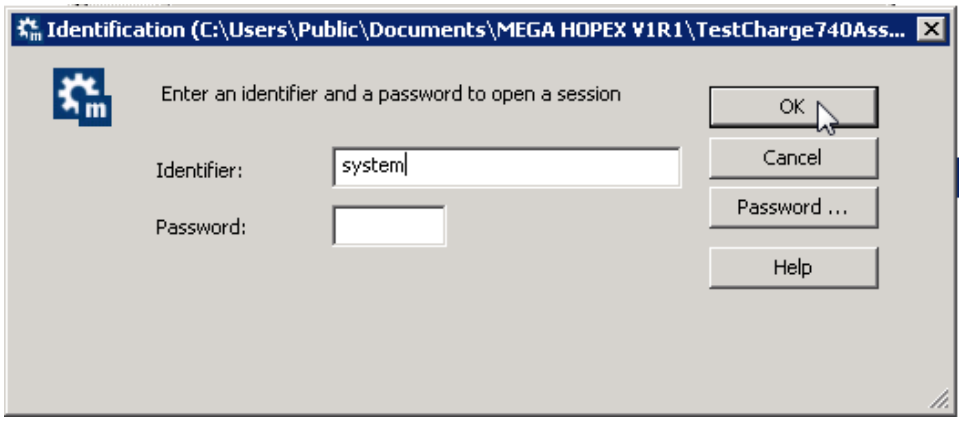
This is because the SystemDb database contains some information about the working repositories. You can discard this warning, and continue.

To attach a database, previously restored in SQL Server Management Studio:

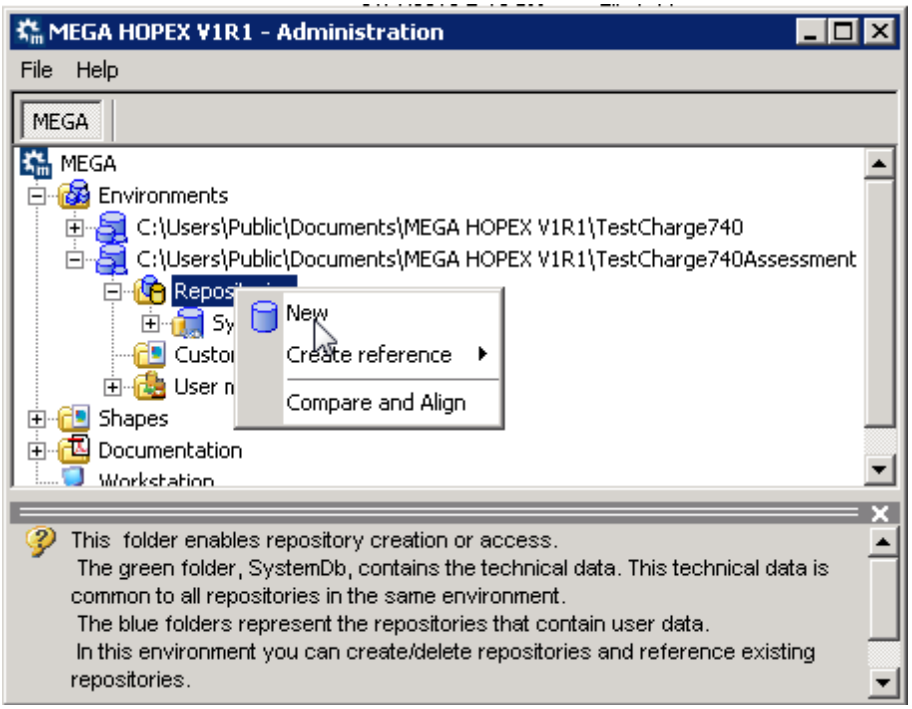
1. Connect to **HOPEX Administration**.



2. Connect to the wanted environment (for example "TestEnv", with **System** user.



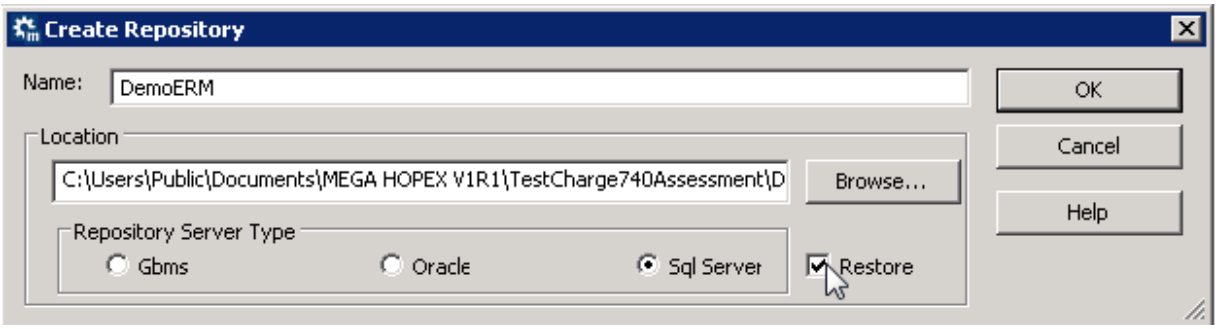
3. Right-click **Repositories** and select **New**.



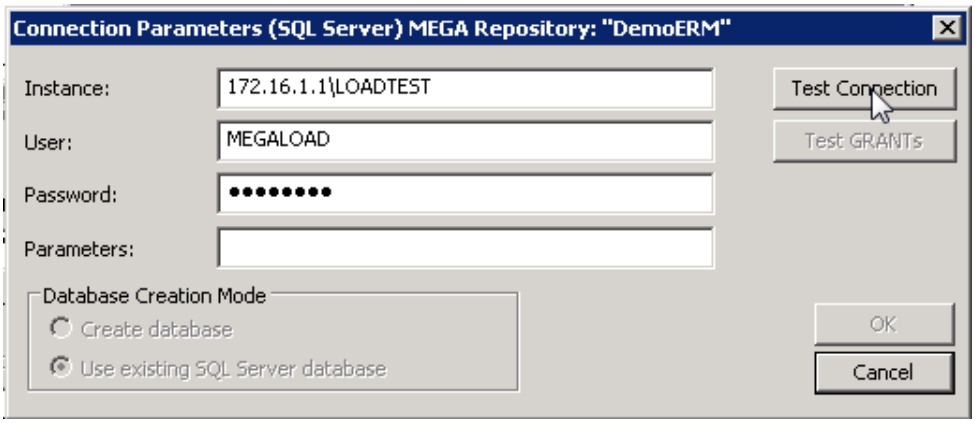
4. Enter the Name of the Repository (in this example "DemoERM")

   In **SQL Server**, you should have an existing database called "TestEnv_ DemoERM", with the native SQL Server user db_owner of the database.

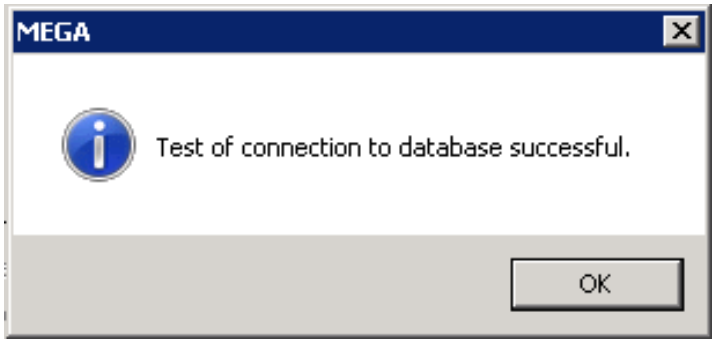
5. In the **Repository Server Type** pane, select **SQL Server**.
6. Select **Restore**.



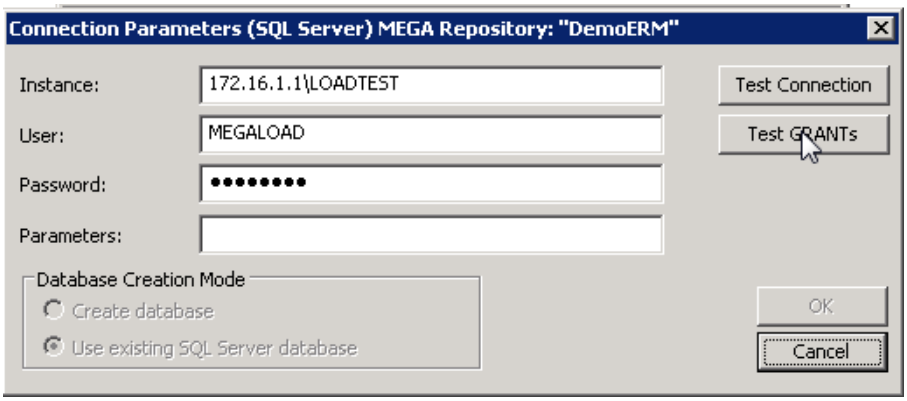
7. Click **OK**.
8. The connection parameters are already set. Check that they are correct.

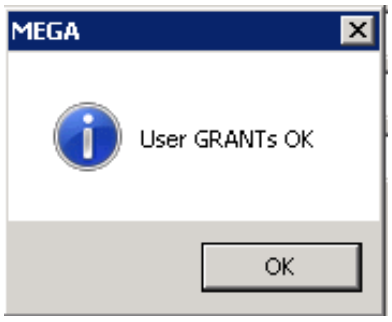


9. Click **Test Connection**.

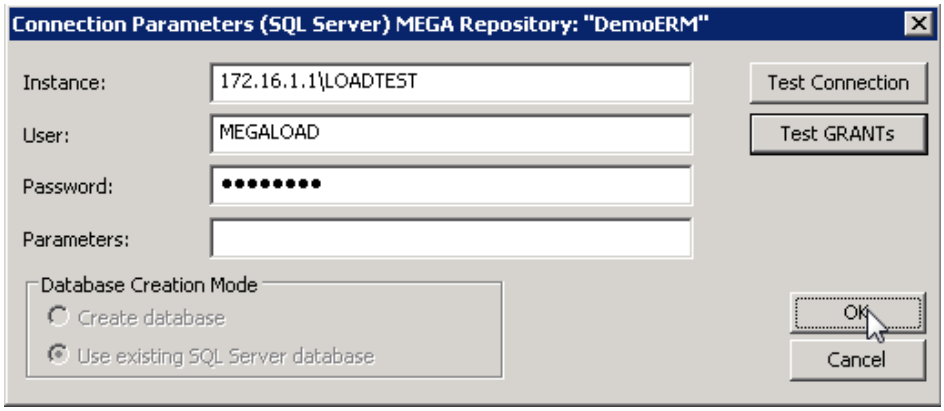


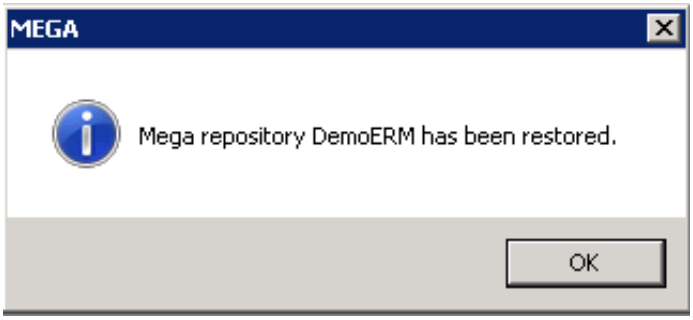
10. Click **Test GRANTs**.



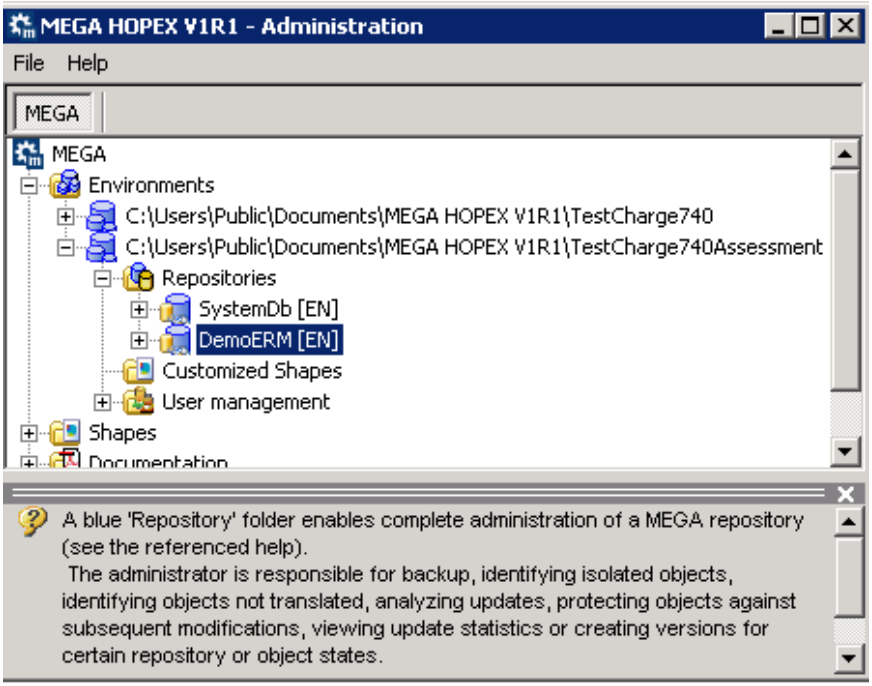


11. Click **OK** to create the repository.





The "DemoERM" repository appears in the **Repositories** list.

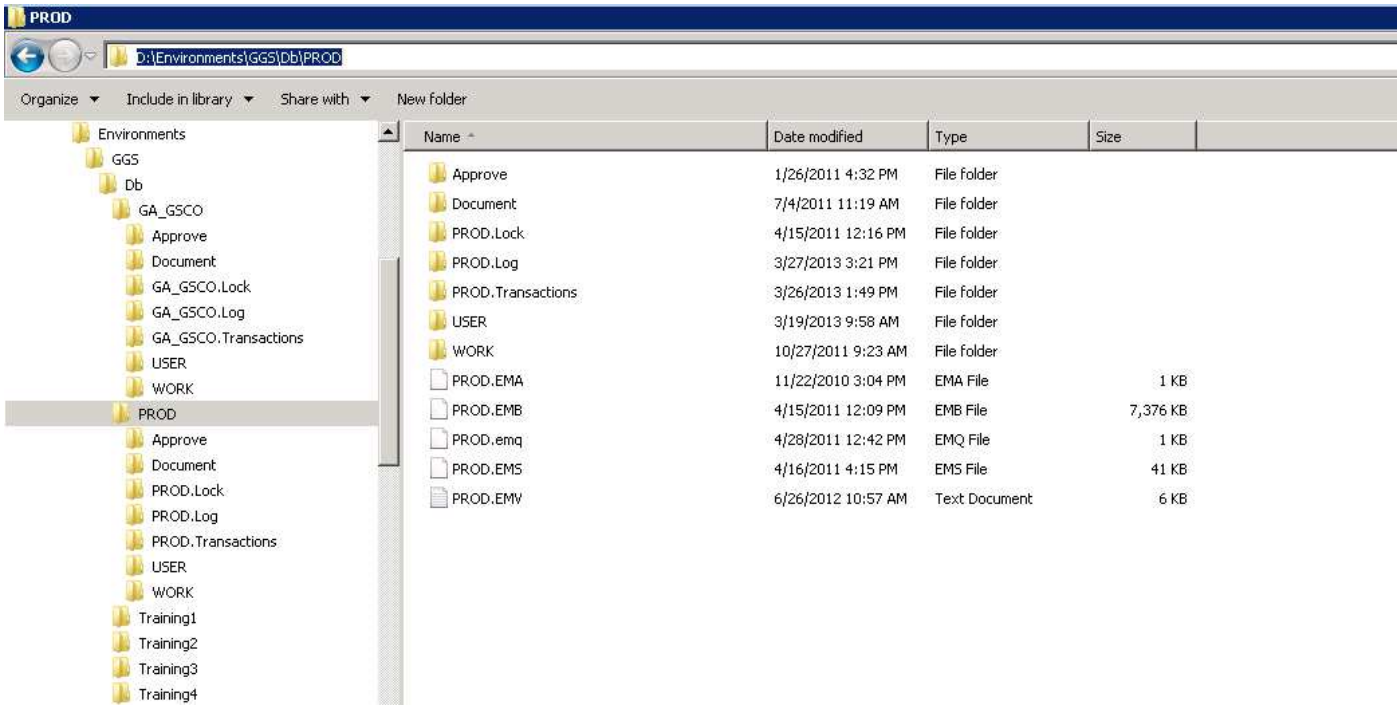


Copy the documents from source to target

You have to copy the documents from source to target for each repository.

The Word/RTF documents

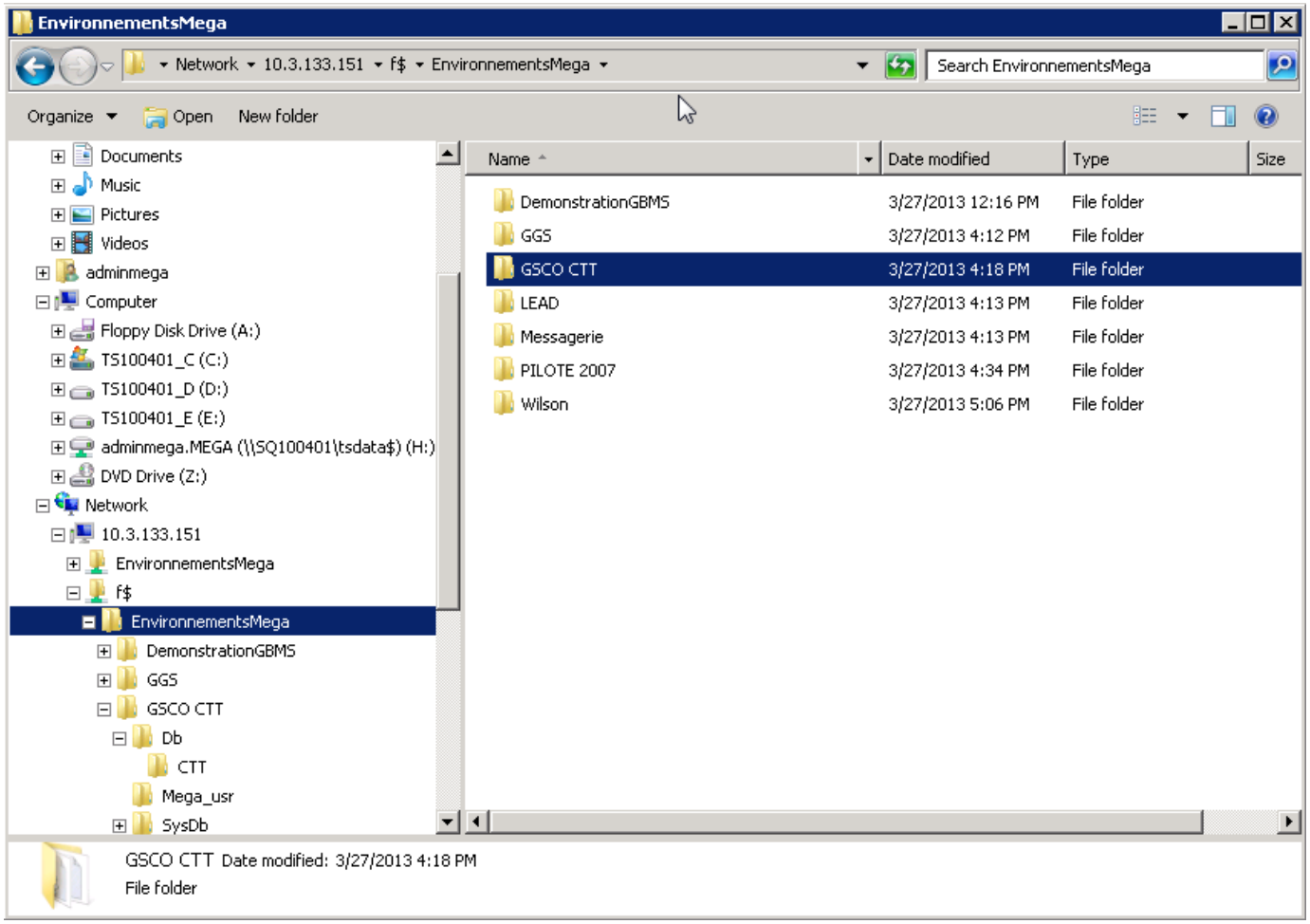
1. Connect to the source server.
2. Go to the folder hosting the environment (for example environment "GSCO CTT").
3. Go to the **Db** folder, and in the sub-folder of each repository you migrated (in this example « D:\Environments\GGS\Db\PROD » contains the data of the repository called "CTFTM_OLD").



4. From the source server, open an explorer on the target server, in the same folder.

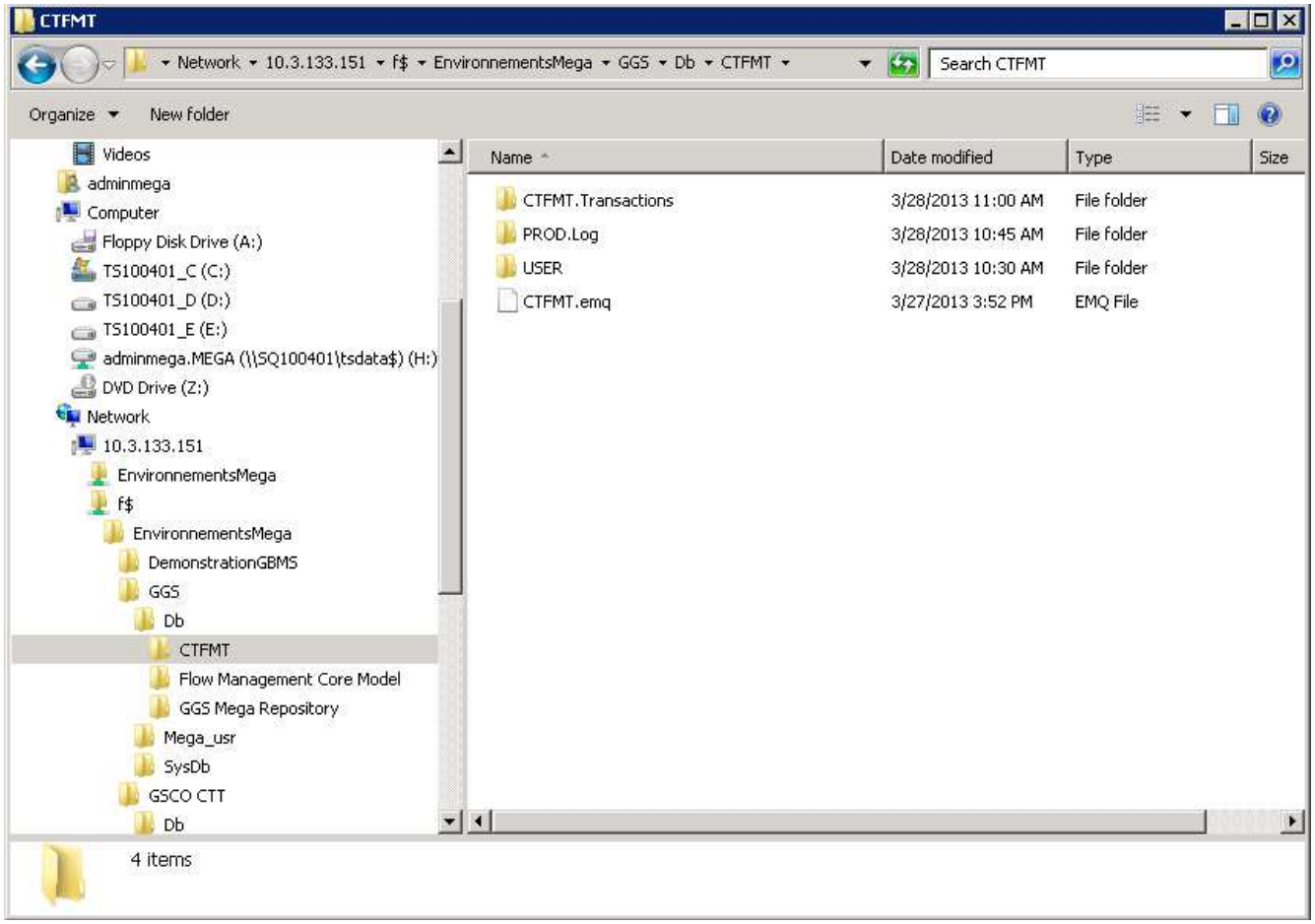
   In this example, the environments on the target server are hosted on the server with 10.3.133.151 IP address, and on that server the environments are located on the F drive in the "EnvironnementsMega" folder, so that the syntax is:

   \\10.3.133.151\f$\EnvironnementsMega\

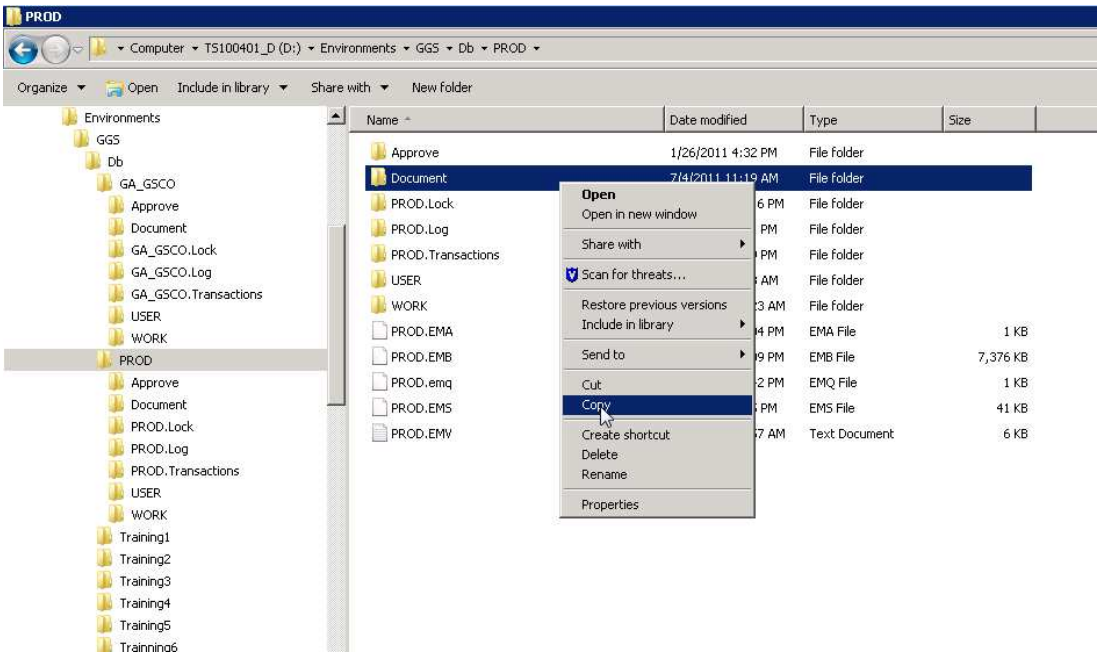


5. Go to the same sub-folder « …\GGS\Db\CTFMT ».

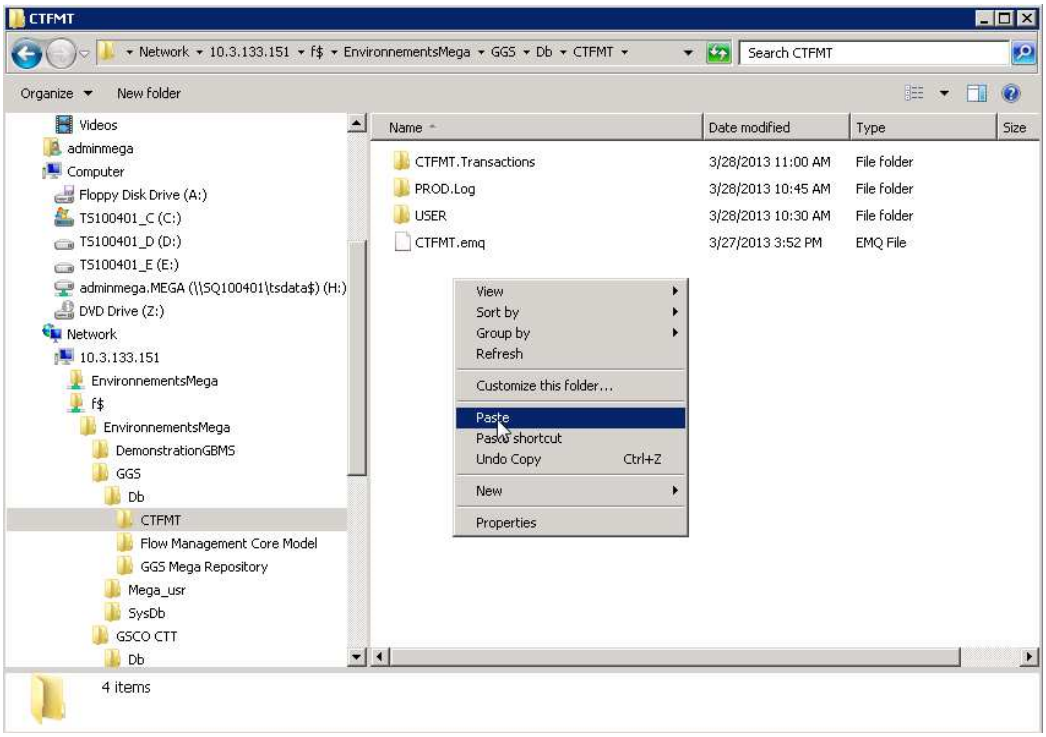
   Note: If you are wondering why it is not the same environment name and the same repository name, that is because in this example, this repository was taken from one environment to another, with a rename.



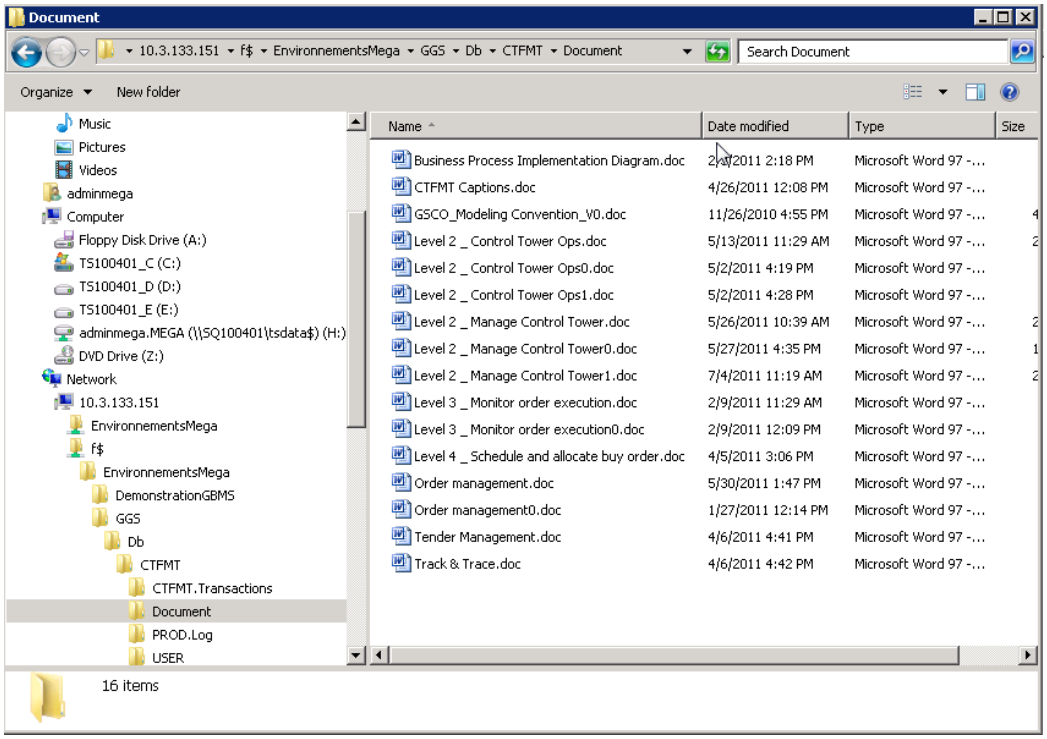
6. Right-click the **Document** folder from the source and select **Copy**.



7. Paste it on the target (future Production) :



8. Check that the documents are all in the target folder.



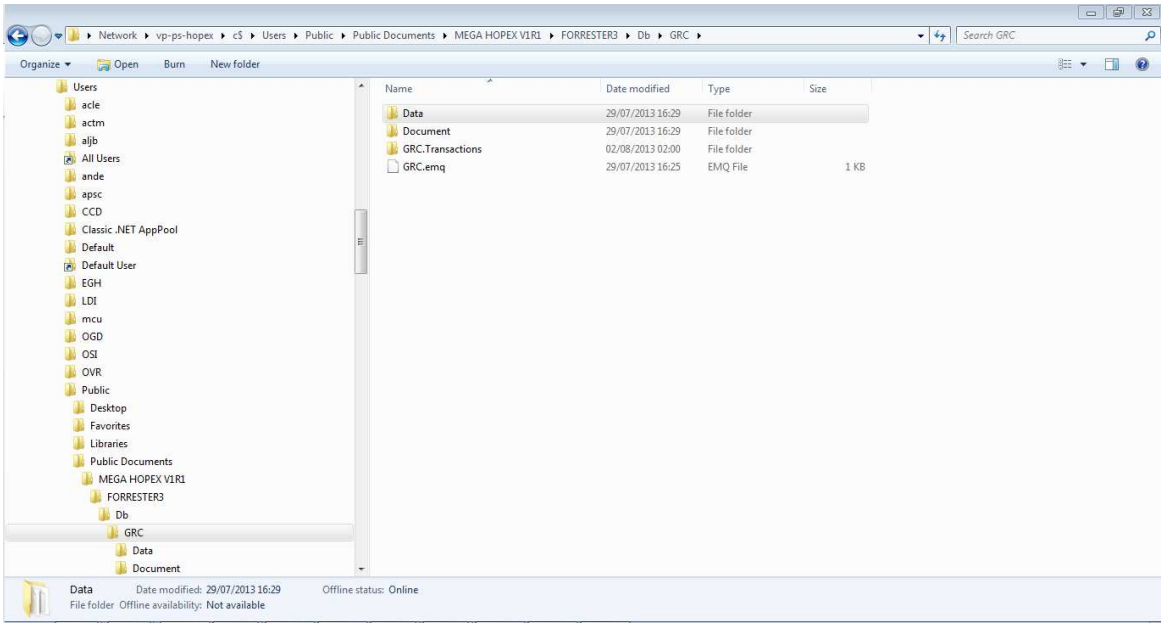
The internal documents (.DAT files)

Some documents can also be stored partially inside the SQL Server databases (as references) and partially in .DAT files (generated in the folder of each repository of the migrated environment).

By restoring the SQL Server databases, you already retrieved the references. Now you have to copy a folder from source to target.

1. From the source environment, expand the **Db** folder and the sub-folder of the database.

In this example, on server vp-ps-hopex, we are looking at environment py aRESTER3 from source to target.the migrated environment).ry name, that is because in tX V1R1\FORRESTER3\Db\GRCxa



2. Right-click the **Data** folder and select **Copy**.
3. Paste it at the same level on the target folder.

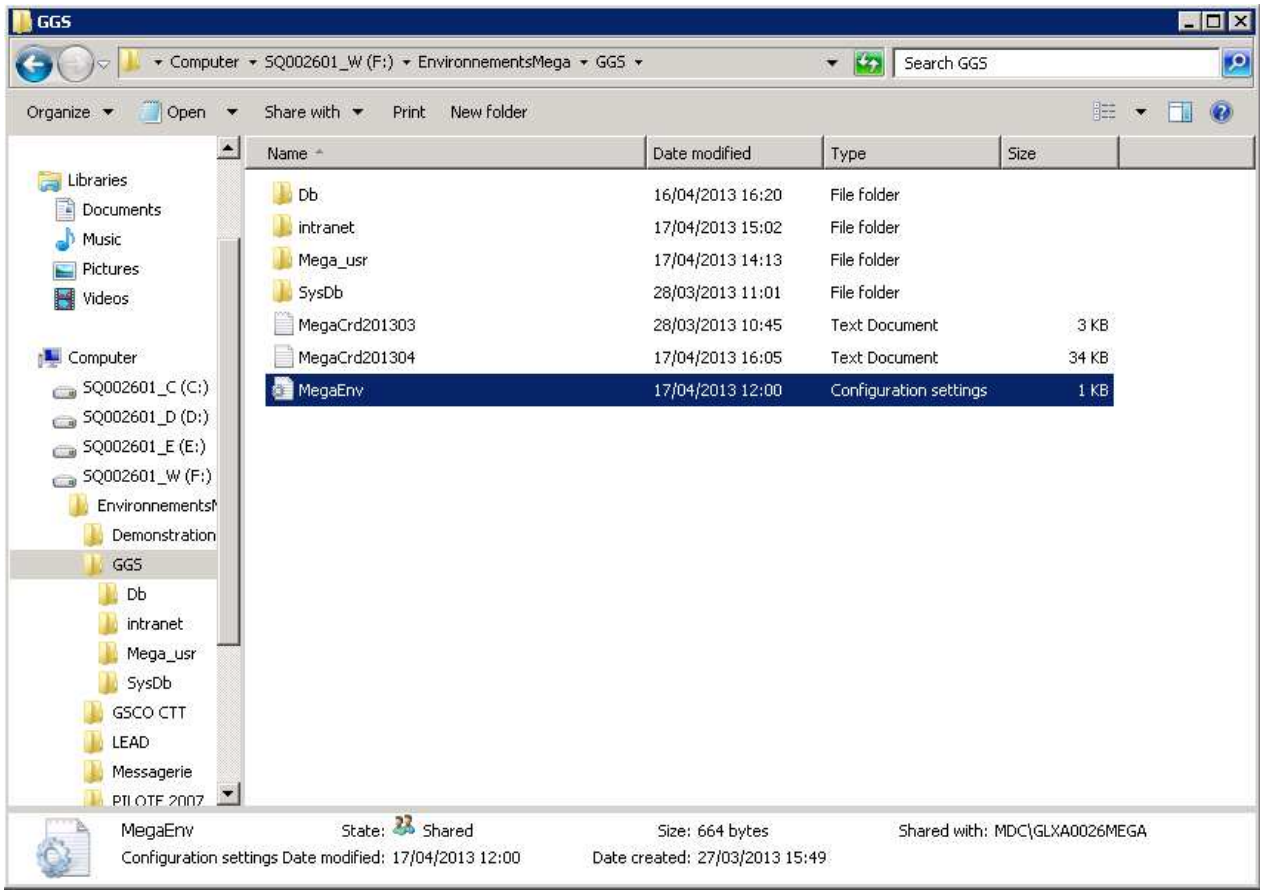
Get the parameters of the environment

For each environment, you need to retrieve certain types of configuration included in:

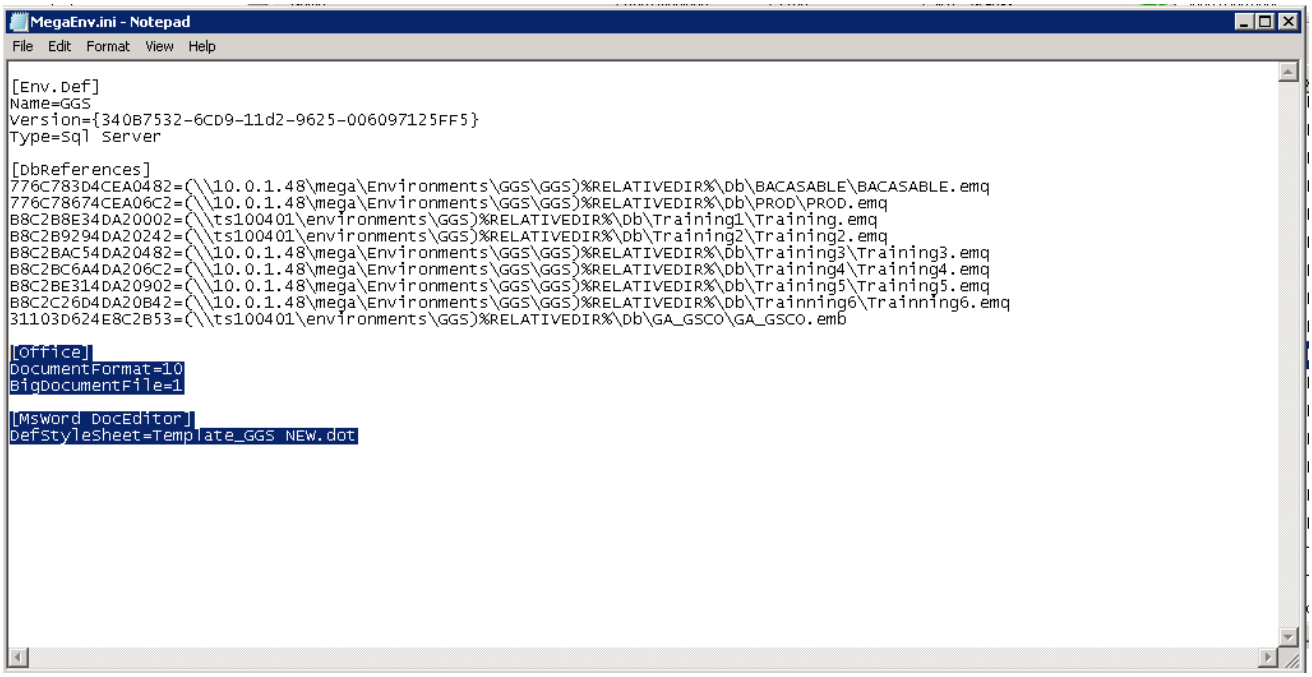
- the **MegaEnv.ini** file
- the **Mega_Usr** folder.

MegaEnv.ini file

The MegaEnv.ini file is located at the root level of each environment (in this example, in the folder tione database. In thga\GGSy afor the GGS environment) on the source server.



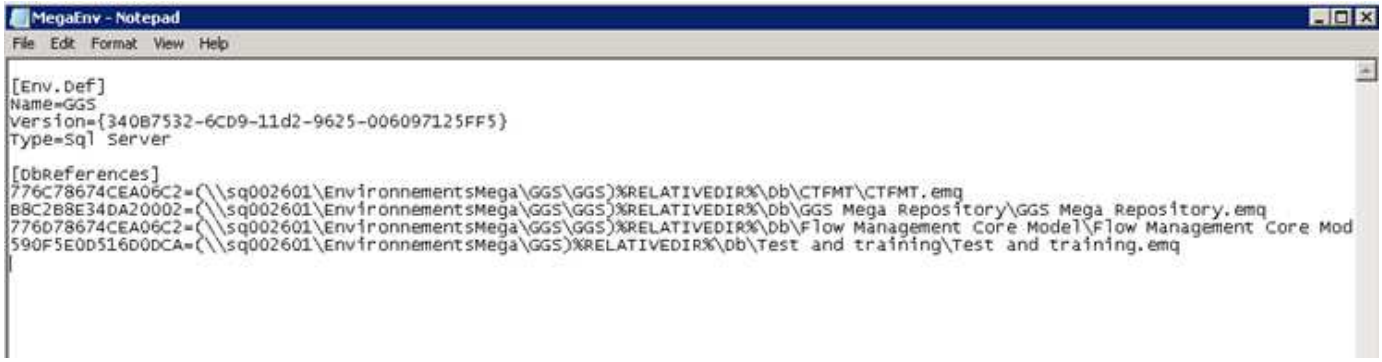
1. Open the **MegaEnv.ini** file on the source server.



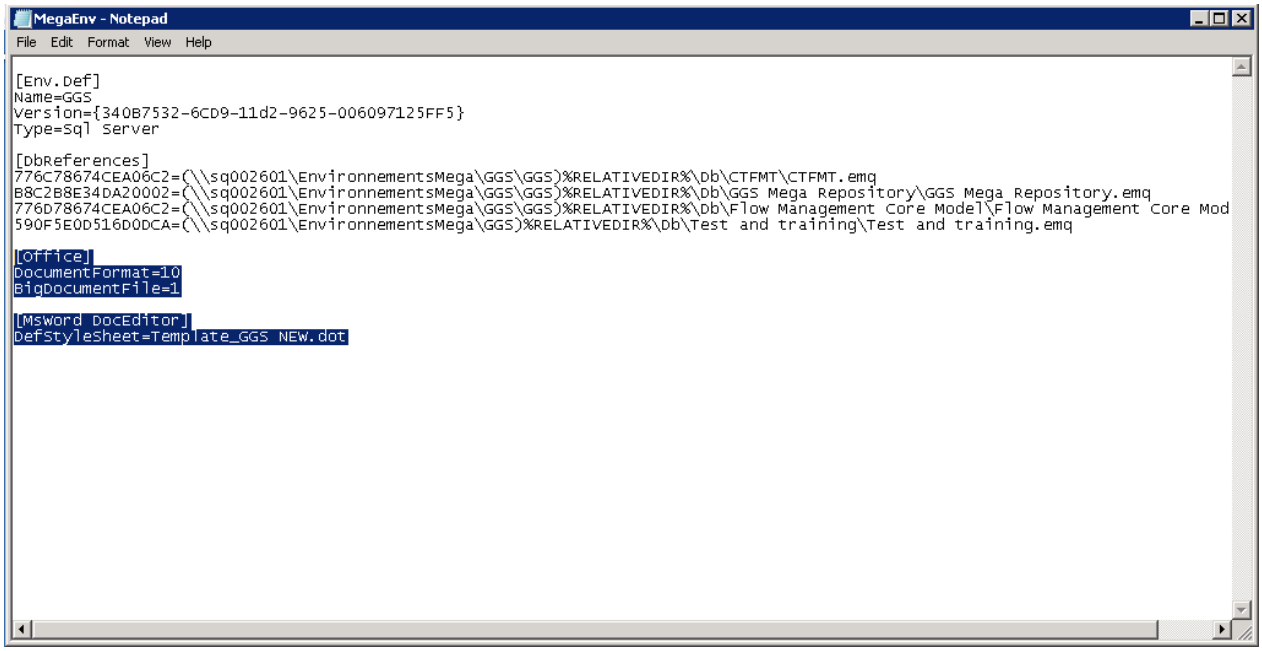
2. Below the [DbReferences] section, you find the environment-specific parameters, like the type of document (DocumentFormat=10 meaning that they were converted from .DOC to .RTF) or the templates used by default.

   Copy this specific section.

3. Open the **MegaEnv.ini** of the target server.



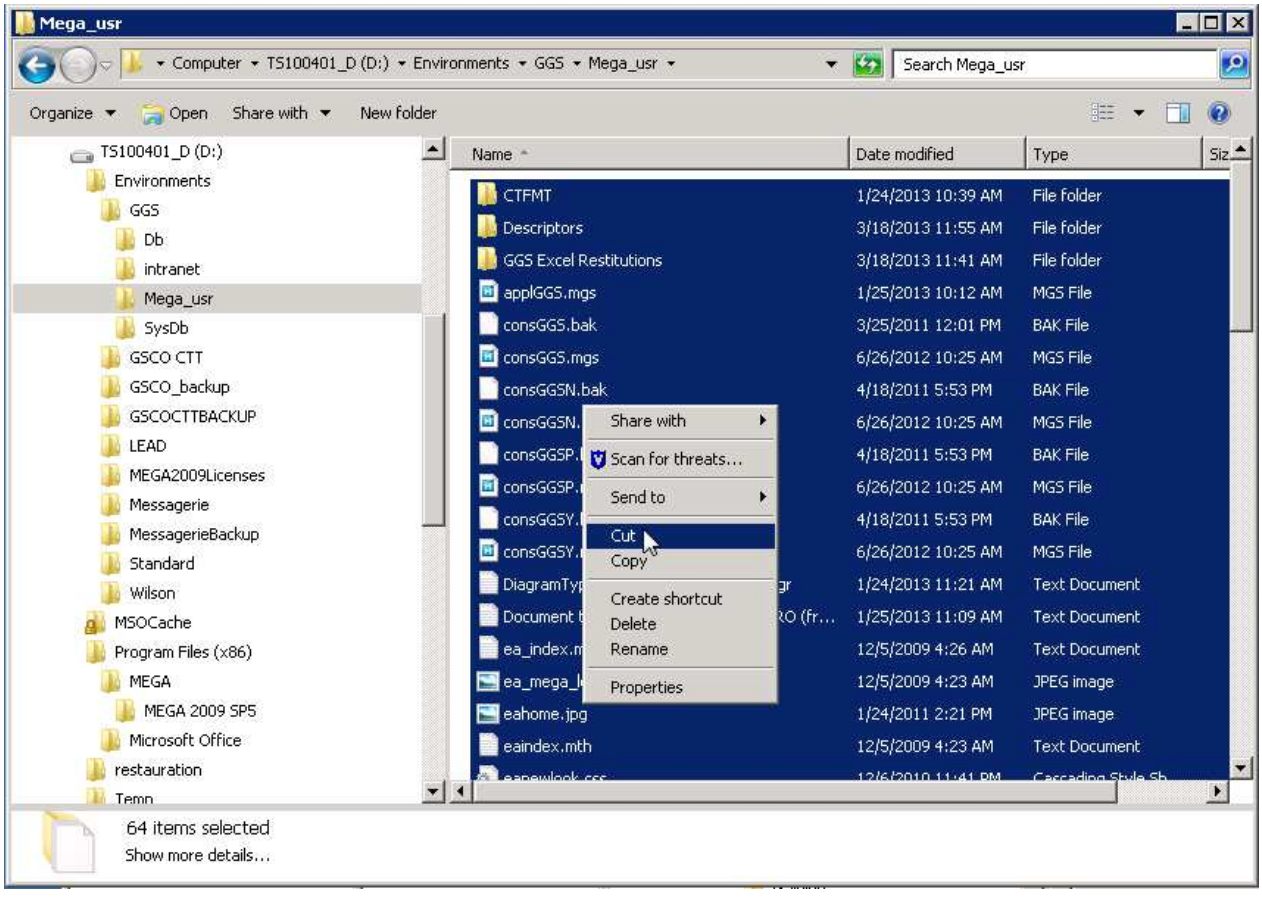
4. Paste this section at the same level in the file and save the document.



Mega_Usr folder

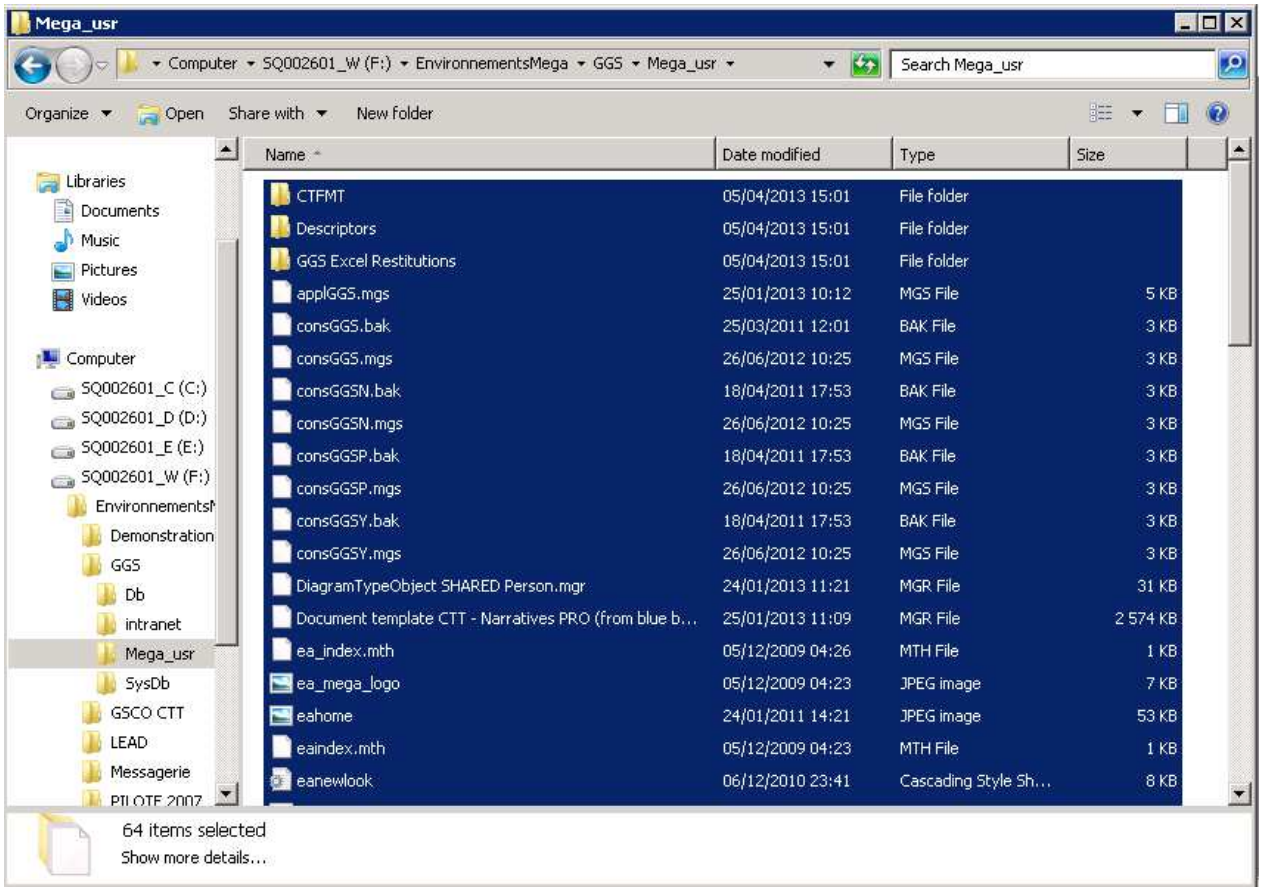
In each environment, there is a **Mega_usr** folder that includes templates, icons, web pages, etc. Those are environment-specific and correspond to objects in the HOPEX respositories. It is mandatory to transfer these if you want to keep your customizations.

1. In the source server, go to the root level of your environment.
2. Copy everything included in the **Mega_usr** folder.



3. Righ-click the selection and selecet **Cut**.

4. Go to the target server in the folder of the duplicated environment.
5. Paste everything in the **Mega_usr** folder.



AUTOMATIC TRANSLATION ADMINISTRATION

The automatic translation proposed by **HOPEX** is based on Microsoft Translator Text API.

There are two ways of using automatic translation of the data created by the users:

- interactively managed by the users, object by object.
- per batch processing by an Administrator.

This section is dedicated to the administration work of the batch processing.

PREREQUISITES FOR THE USE OF AUTOMATIC TRANSLATION

For the commands linked to the automatic translation of data to be available, you must have acquired a **Microsoft Translator Text API** license.

* *For further details about this license, see* https://azure.microsoft.com/en-us/services/cognitive-services/translator-text-api/.

M **To use this service, your IT architecture must allow the HOPEX servers to access the URL https://api.cognitive.microsoft.com/.**

The API key of your **Microsoft Translator Text API** license must be declared and, if your API key is region-dependent, the region to which this key belongs must also be declared.

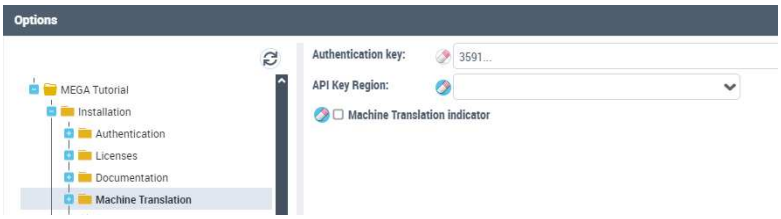
An option enables an administrator to enter the **Authentication key** number and its region, enabling activation of the automatic translation APIs.

To activate this option in **HOPEX Administration Web Front-End**:

1. You must log on as a **HOPEX Administrator**.

   * *For further information about logging in to **HOPEX Administration Web Front-End**, see the "Web Administration Desktop" chapter in the **HOPEX Administration - Supervisor** guide.*

2. At the top of the workspace, click the **Environment options** button. The **Options** window of the environment opens.
3. In the tree on the left, select **Installation > Automatic translation**.



4. Enter the **Authentication key** number.
5. If your API key is region-dependent, select the **API Key Region**.
6. Check the **Automatic translation indicator** box if you want a note to appear at the end of each translated text to specify that the translation was done automatically.

CONFIGURING THE INCREMENTAL AUTOMATIC TRANSLATION BATCH

To activate automatic translation batches, the Administrator must create a trigger in the administration tool which implements the incremental automatic translation macro. When the trigger has been created, you must then plan the automatic update of the translations. See Initiating automatic batch translation.

Before processing incremental automatic translations from a source language to a target language, these translations must be initialized via automatic batch processing. See Configuring the incremental automatic batch translation.

Initiating automatic batch translation

To create an automatic translation initialization trigger from a source language to a target language, the administrator must perform the following tasks:

1. Opening the triggers definition window.
2. Creating the automatic translation initialization trigger.
3. Creating a task specific to automatic translation initalization.

Opening the triggers definition window

To open the triggers definition window:

1. Open the **HOPEX Administration Windows Front-End** module.

   * *For further information about logging in to **HOPEX Administration Windows Front-End**, see the "Access **HOPEX Administration**" chapter in the **HOPEX Administration - Supervisor** guide.*
2. Open the environment that interests you.
3. Expand the repository folder for which you wish to translate the data.
4. Right-click the **Scheduler** folder and select **Manage Triggers**.
   The triggers definition window is opened.

Creating the automatic translation initialization trigger

To create an automatic translation initialization trigger:

1. Open the triggers definition window.
2. Select the **Triggers Definitions** tab.
3. Click the **New** button to create a **System Trigger Definition** specific to automatic translation.
4. In the creation wizard of the **System Trigger Definition**, create **System Job Definition**.
5. In the creation wizard of the **System Job Definition**, from the **Implementation** field, link the "Machine Translation Scheduler [All] Macro" and click **OK**.
6. In the creation wizard of the **System Trigger Definition**, click **Next**.
7. Specify the time and date of the batch launch.

8. Set the **Recurrence Type** at "One time".
9. Click **Finish**.

Creating a task specific to automatic translation initalization

To create an initializing automatic translation user trigger from a source language to a target language:

1. Open the triggers definition window.
2. Select the **User Triggers** tab.
3. Click the **New** button to create a task specific to automatic translation initialization.
4. In the creation wizard, select the **System Job Definition** for automatic translation initialization.
   * *For more details, see Creating the automatic translation initialization trigger.*
5. Specify the trigger name.
6. Enter the text of the **Job Context** in the following form:
   "[source language],[target language]," (a comma must be added at the end).

   ```
   For example, "it,fr," signifies that the source language is
   Italian and the target language is French.
   ```

   * *If a language is not specified, it will be replaced by the data language.*
   * *For more information about language abbreviations, see Table of abbreviations associated with languages.*
7. Click **Finish**.
   The batch will be activated at the time stipulated in the **System Job Definition** to which it is linked.

Configuring the incremental automatic batch translation

The Administrator must therefore carry out the following tasks:

1. Activate initial automatic translation, see Initiating automatic batch translation.
2. Opening the triggers definition window.
3. Creating the incremental automatic translation trigger.
4. Creating a task for incremental automatic translation.

Creating the incremental automatic translation trigger

To create an automatic translation initialization trigger:

1. Open the triggers definition window.
2. Select the **Triggers Definitions** tab.
3. Click the **New** button to create a **System Trigger Definition** specific to automatic translation.
4. In the creation wizard of the **System Trigger Definition**, create **System Job Definition**.
5. In the creation wizard of the **System Job Definition**, from the **Implementation** field, link the "Machine Translation Scheduler [Incremental] Macro" and click **OK**.

6. In the creation wizard of the **System Trigger Definition**, click **Next**.
7. Specify the time and date of the batch launch.
8. Set the **Recurrence Type** for the period best corresponding to your activity.
9. Click **Finish**.

Creating a task for incremental automatic translation

The procedure for creating a **user trigger** for incremental automatic translation from a source language to a target language is identical to the procedure for a **user trigger** for automatic translation initialization (see Creating a task specific to automatic translation initalization).

However, in the user trigger creation wizard, select the **System Job Definition** for incremental automatic translation.

* *For more details, see Creating the incremental automatic translation trigger.*

The batch will be activated at the time stipulated in the **System Job Definition** to which it is linked.

Table of abbreviations associated with languages

Language	Abbreviation
Arabic	ar
Bosnian	bs
Bulgarian	bg
Chinese (Simplified)	zh-Hans
Chinese (Traditional)	zh-Hant
Croatian	hr
Czech	cs
Danish	da
German	de
Dutch	nl
English	en
Spanish	es
Finnish	fi
French	fr
Greek	el

Language	Abbreviation
Hebrew	he
Hungarian	hu
Icelandic	is
Indonesian	id
Italian	it
Japanese	ja
Korean	ko
Malay	ms
Norwegian	nb
Polish	pl
Portuguese	pt-br
Romanian	ro
Russian	ru
Serbian	sr-Latn
Slovak	sk
Slovenian	sl
Swedish	sv
Thai	th
Turkish	tr
Vietnamese	vi