

Installation and Deployment

HOPEX V3



Information in this document is subject to change and does not represent a commitment on the part of MEGA International.

No part of this document may be reproduced, translated or transmitted in any form or by any means without the express written permission of MEGA International.

© MEGA International, Paris, 1996 - 2019

All rights reserved.

HOPEX is a registered trademarks of MEGA International.

Windows is a registered trademark of Microsoft Corporation.

The other trademarks mentioned in this document belong to their respective owners.

Web Front-End Installation Guide HOPEX V3 EN

Prerequisites	4
Operating System	4
.Net	4
Visual C++ Redistributable for Visual Studio 2015	4
.NET Core 2.1 Runtime & Hosting Bundle.....	4
Required roles	5
Desktop heap configuration	10
Configuration of SSL / TLS.....	11
Windows User(s) for MEGA HOPEX.....	11
Define Group permissions	12
Define MUST Licence Access.....	13
Define COM Access rights	13
MEGA HOPEX Setup	16
Choosing your setup type.....	16
Web Front-End Standalone Setup	16
Advanced Setup.....	21
Choice Screen	22
Advanced Parameters	27
Completing Installation	31
Define "Windows User for MEGA HOPEX" files Access Rights	31
Tune IIS	33
Configure Web Content expiration	36
What's next?	39
Testing the Installation	40
Testing MEGA HOPEX (Web Front-End).....	41
More required configuration	43
Word, Excel and PDF exports.....	43
Reports (MS Word)	43
Required options configuration	44
Allowing the use of verbose logs and activation.....	46
URL Rewrite	50
Troubleshooting	53
Check that the Site Service Provider is running.....	53
Restarting Internet Information Services	54
Referencing a New Environment	55
Disabling Data Execution Prevention	55
Loosening Internet Explorer Security Settings	56

Summary

This document describes all the steps required to install MEGA HOPEX Web Front-end on Windows Server 2008 R2 or above.

PREREQUISITES

Operating System

MEGA HOPEX Web Front-End can be installed on the following systems:

- Windows Server 2008 R2 (not recommended because of the coming end of Support by Microsoft, but supported)
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016

To install MEGA HOPEX Web Front-End, you must open a Windows session with a domain user that has administrator rights on the server machine.

This document describes installation steps for **Windows Server 2012 R2**. Some steps might need to be adapted when using Windows Server 2008 R2.

.Net

.Net 4.6.2 is required.

It is already installed by default with Windows Server 2016.

For more information on installing it on previous versions, please follow the following article:

<https://www.microsoft.com/en-us/download/details.aspx?id=53345>

Visual C++ Redistributable for Visual Studio 2015

This package is required for a good behavior of both the web client, and the Windows client of the application.

The associated libraries were previously embedded with our own, but Microsoft changed its approach, and recommends now to install this redistributable through their official installer.

The offline installer can be found at this address:

<https://www.microsoft.com/en-us/download/details.aspx?id=48145>

If you want to download it from that location, make sure to download the 32 bits' version, file "vc_redist.x86.exe".

Moreover, once the HOPEX application is installed, you will be able to find it in the folder "<installation folder>\Install\vc_dedist".

.NET Core 2.1 Runtime & Hosting Bundle

Warning: only necessary with Update 04 of Hopex V2R1, or above.

This component installs a module in IIS to allow the execution of ASP.NET Core web services.

With Update 04, a diagnostic/benchmark page is available, using this feature.

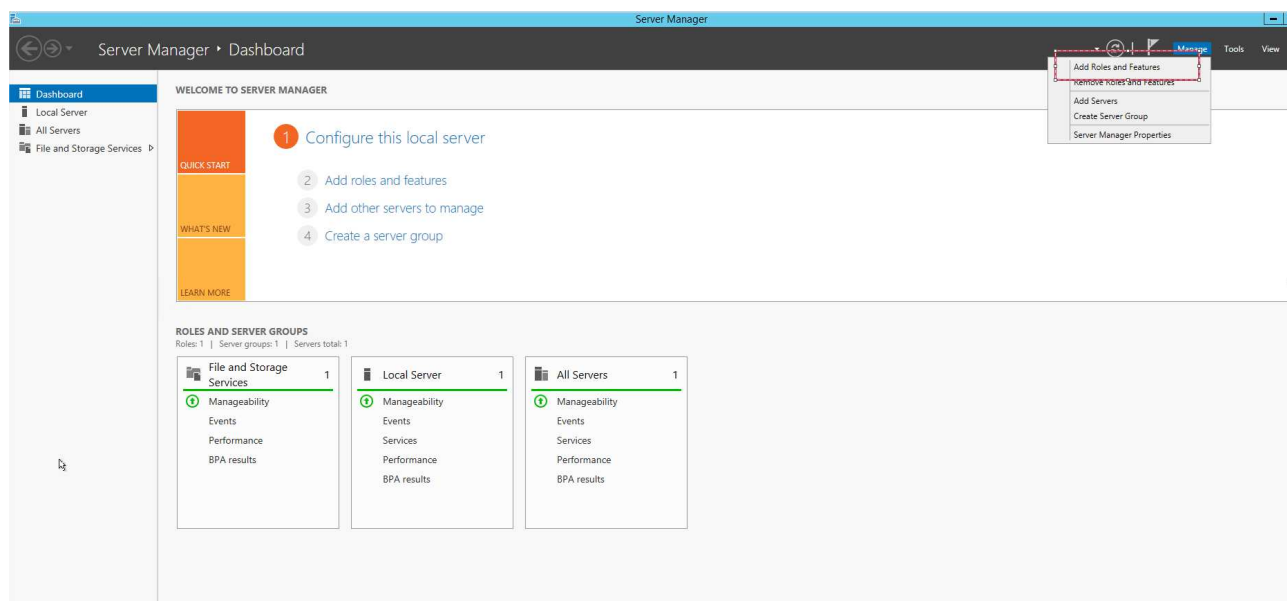
You can download the installer following this link:

<https://dotnet.microsoft.com/download/thank-you/dotnet-runtime-2.1.7-windows-hosting-bundle-installer>

Required roles

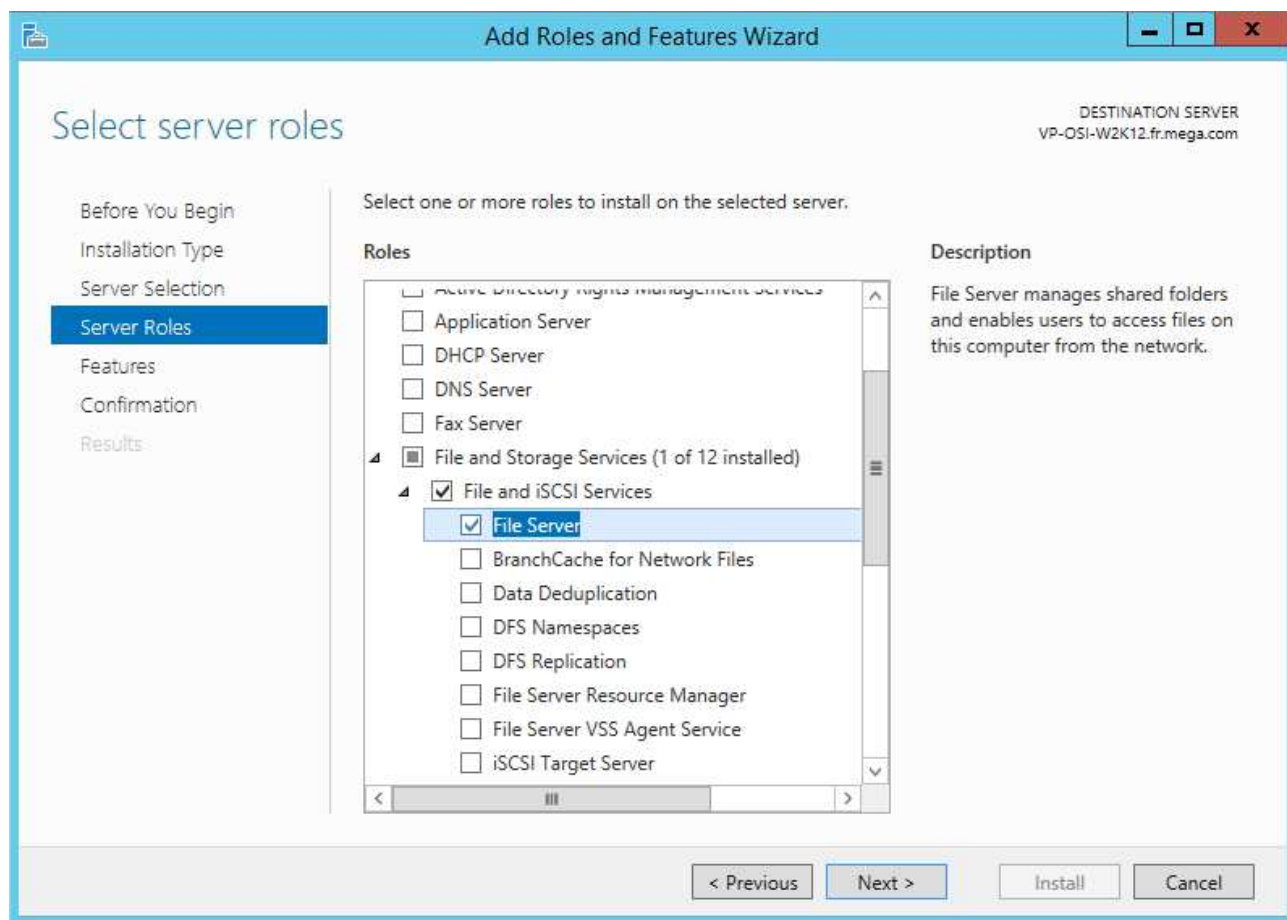
If the roles are already in place, at least check that all sub features are active.

1. Through the "Server Manager", click "Manage" and select "Add Roles and Features":

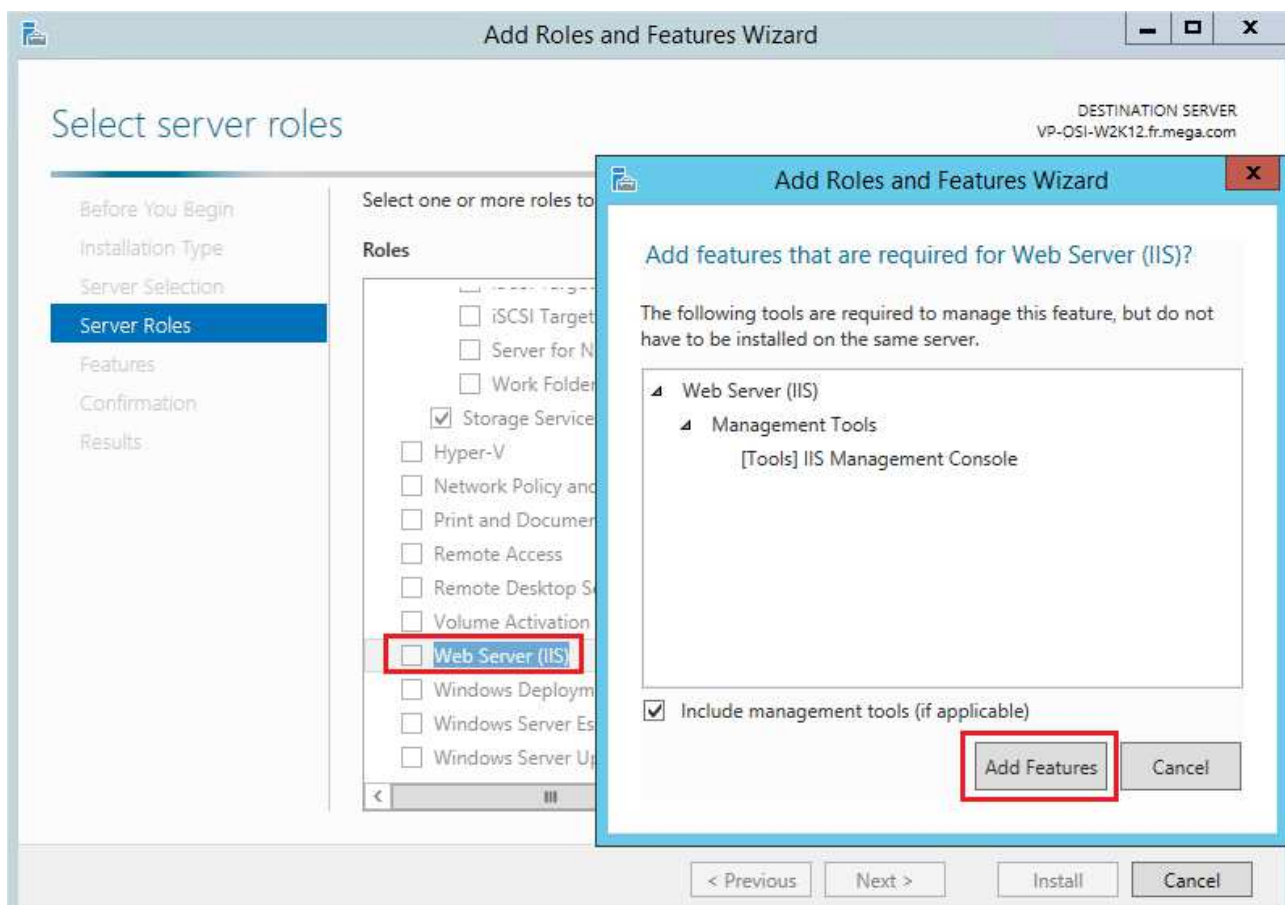


2. In the Roles, activate both:

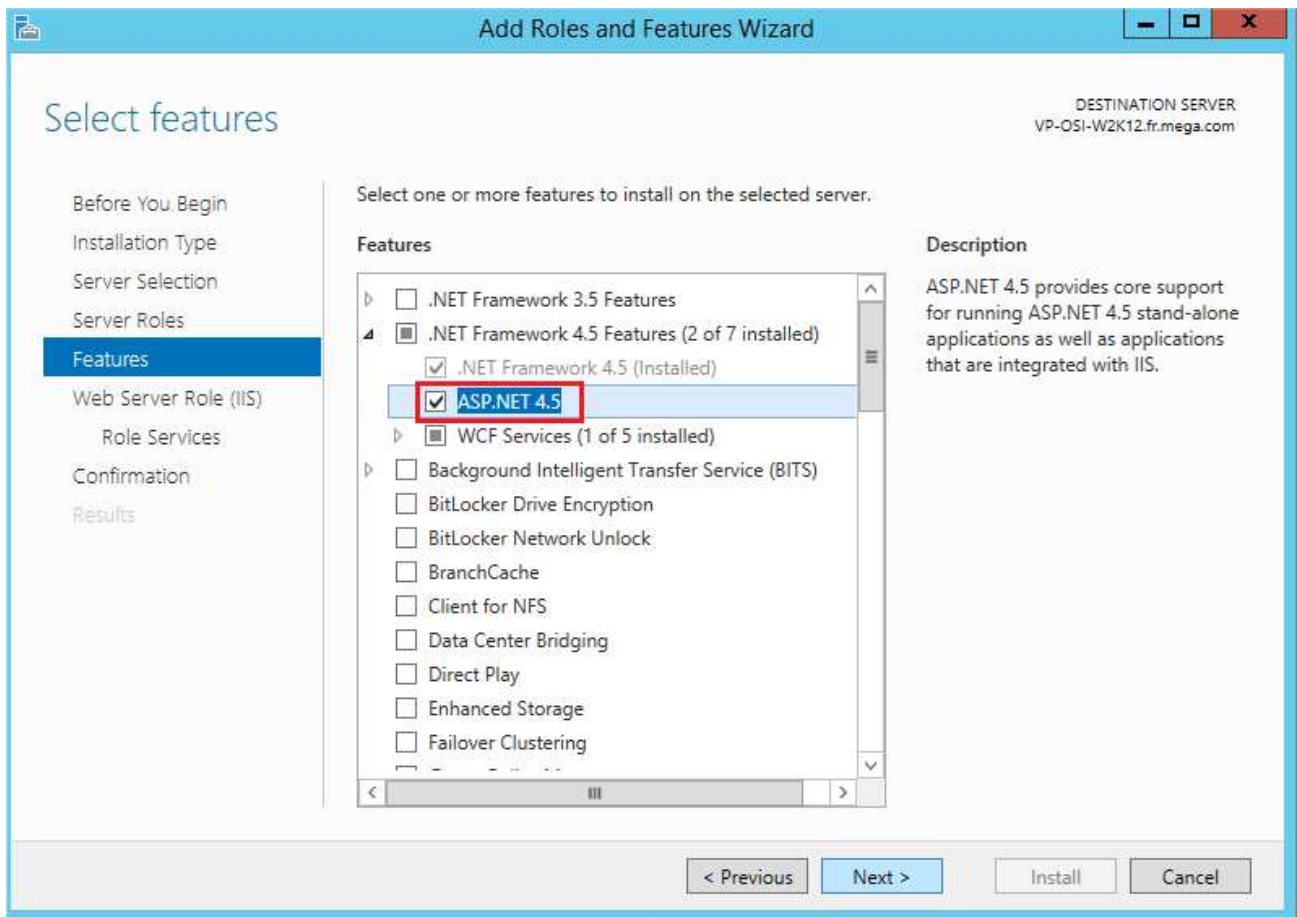
- « File Server » :



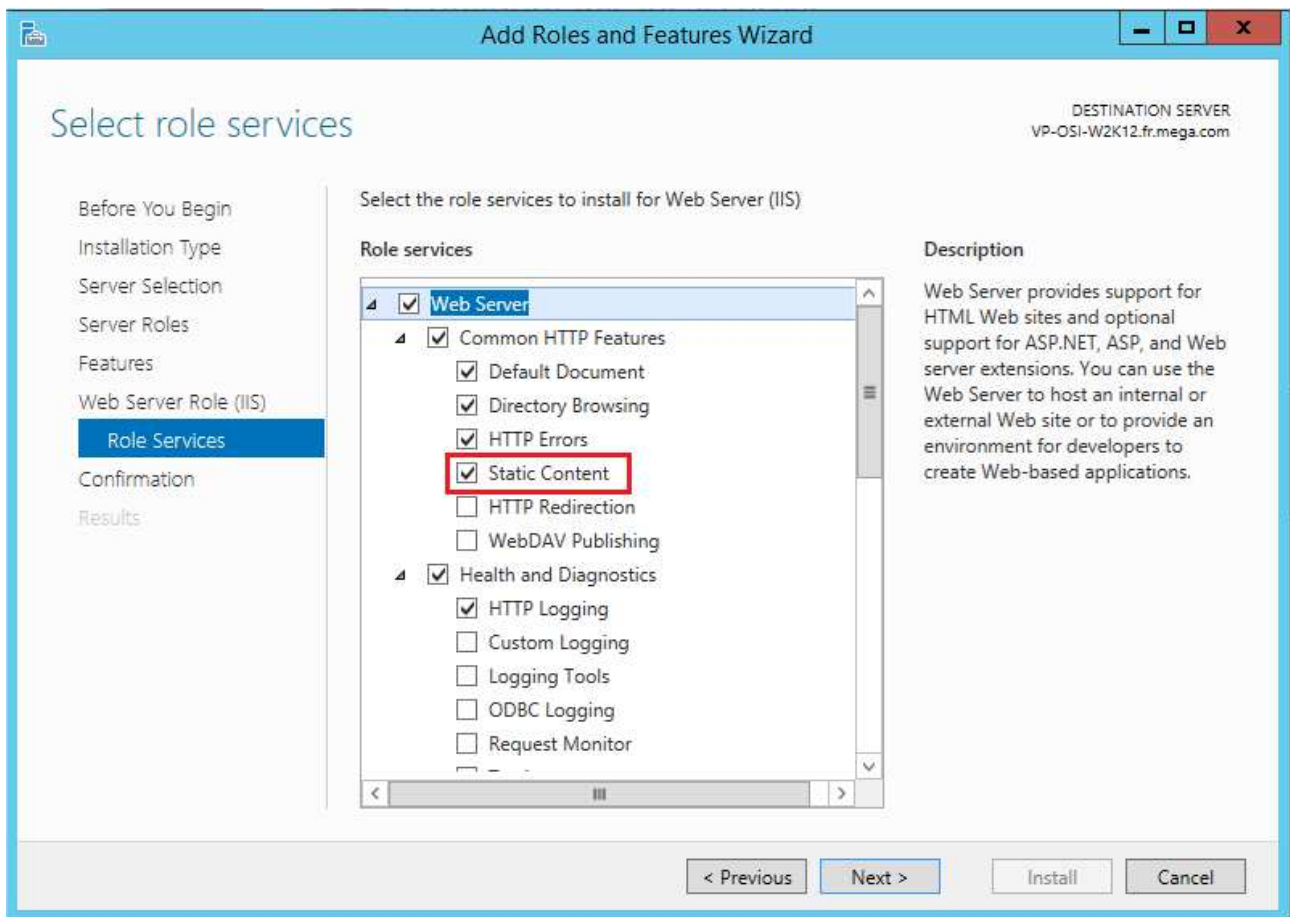
- « Web Server (IIS) » and its related features :



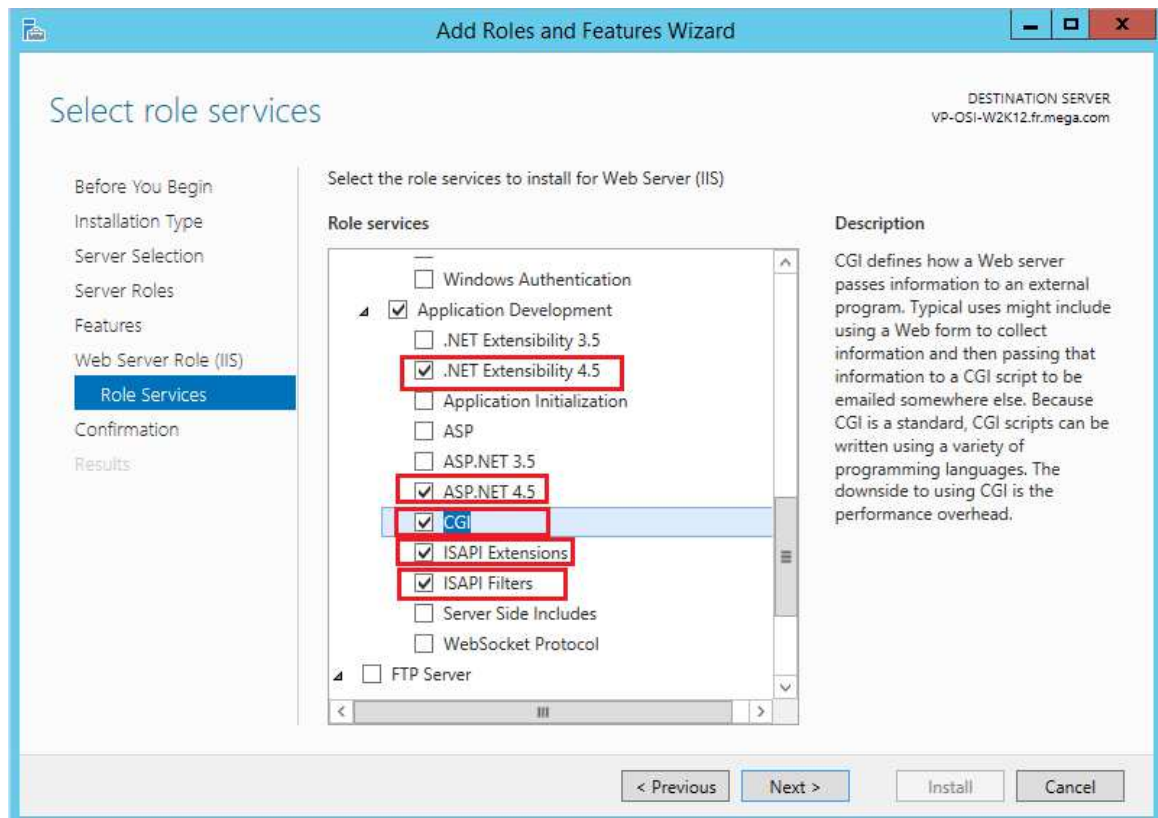
3. In the "Features", make sure to activate the "ASP.NET 4.5". Normally, if you installed 4.6.2 prior to this activation of features, it shouldn't be needed:



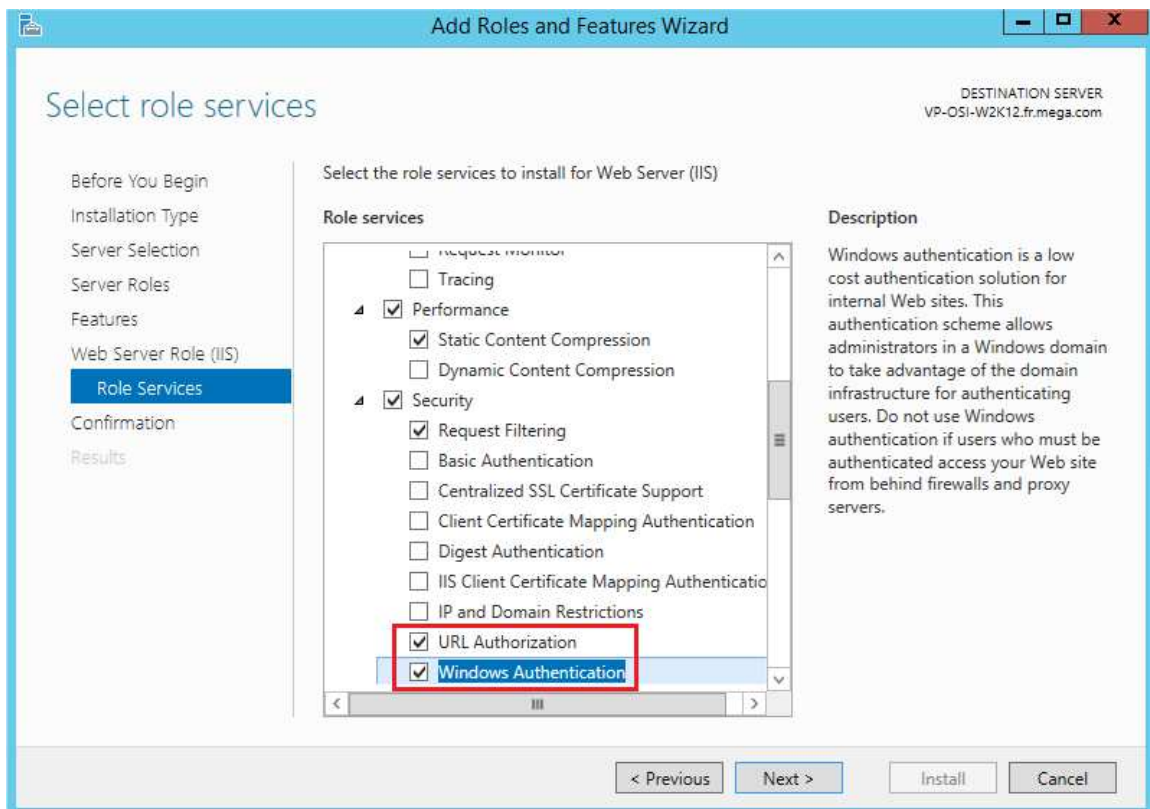
4. In the "Role Services" of IIS, make sure that "Static Content" is checked:



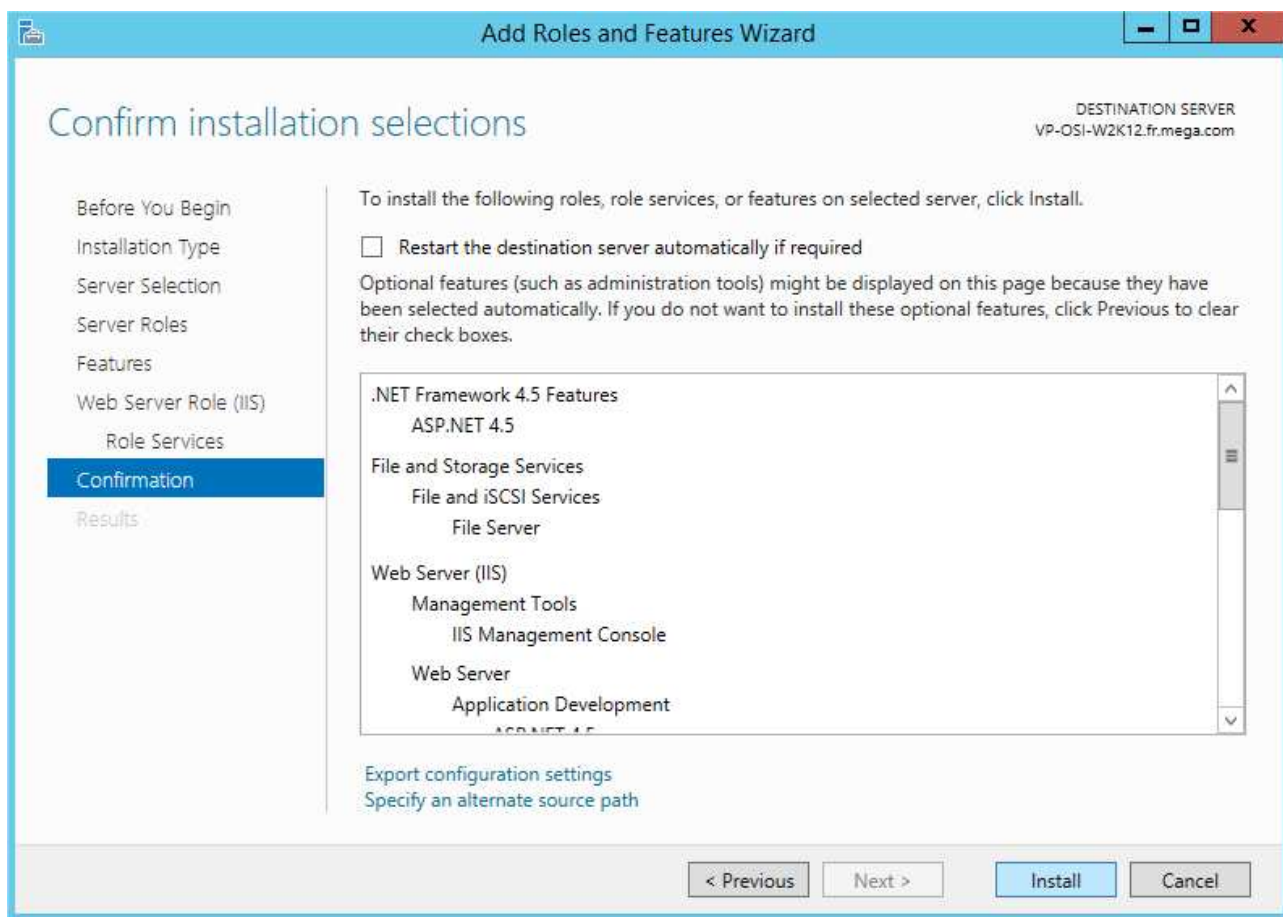
5. In addition, for the "Web Server (IIS)" role, the following "Role Services" and their dependencies must be installed:
- ASP.NET 4.5 (and related features)
 - CGI



6. To manage specific rights on the website (such as restricting access to the admin page), you can also:
 - a. activate "URL Authorization" and "Windows Authentication":



7. Install:



Desktop heap configuration

The Desktop Heap is an internal memory of Windows. It is heavily used by the web application. It is thus **mandatory** to update this value.

Thus, when running several users simultaneously on the same server, the Windows session of the impersonation user might start running out of desktop heap. This will create execution errors.

This is especially true because the impersonation user uses a non-interactive session, and the default value set for the non-interactive desktop heap for in this case is very low.

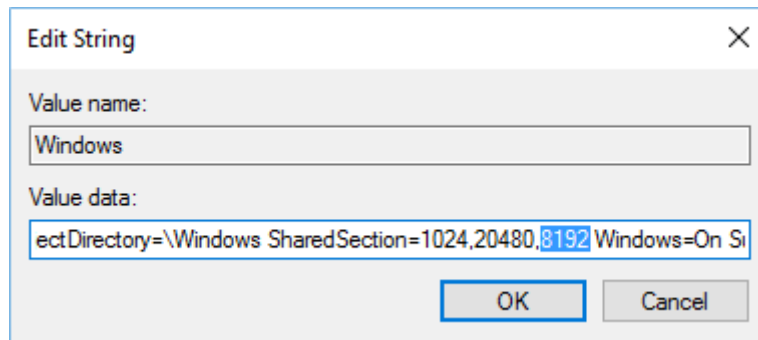
With the latest version of the application, we check that the desktop heap is set to at least 8 MB. If it isn't, anyone that will access the website will receive a warning message.

This modification needs to be made in the Windows Registry. Look for the "Windows" value in *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\SubSystems*. There's a long string for this value that will look similar to this: *%SystemRoot%\system32\csrss.exe ObjectDirectory=Windows SharedSection=1024,3072,512 Windows ...*

It is the Shared Section part that might need to be modified. The three values are, in order: the shared heap, the interactive desktop heap, and the **non-interactive desktop heap**. They are expressed in KB. Default values vary significantly between Windows versions. You will need to modify the non-interactive desktop heap.

Be careful of not using excessive values, as this could stop you from logging into your server. It is therefore recommended to change this value using small increments.

The minimum value to put is therefore: **8192**.



Value name:
Windows

Value data:
ectDirectory=\\Windows SharedSection=1024,20480,8192 Windows=On Si

OK Cancel

It works for small/medium deployments. For configuration when a large amount of concurrent users is expected on the Web Front-End server, please get in touch with your Mega contact that will ask for the assistance of appropriate people.

Configuration of SSL / TLS

To ensure data protection, it is highly recommended to use SSL/TLS.

Therefore, the installer allows to install the web application on a website where HTTPS is already deployed.

If you want to activate this feature, it is then mandatory, as a prerequisite, to configure your IIS platform to activate the SSL/TLS.

You will need to have a certificate. You can bind the HTTPS protocol to any wanted port, the installer will ask you on which website you want to install Hopex, and on which port.

Please note that if you want to do this, you will need to choose the "Custom Setup" type of install (see section "MEGA HOPEX Setup" for details). In a standalone web deployment, we deploy *without* SSL/TLS.

You can find on the following link some documentation of IIS:

<http://technet.microsoft.com/en-us/library/cc771438%28v=ws.10%29.aspx>

We also provide a technical article that explains how to secure the Hopex platform that contains some guidelines about the actions of SSL/TLS, as well as an example of configuration. Please refer to the article "Web Front-End - Securing the platform.doc".

!!Warning!! Do not choose to use the SSL/TLS when installing Hopex unless you have made the deployment on your web server, with a proper certificate on the appropriate port.

Windows User(s) for MEGA HOPEX

When installing MEGA HOPEX, a user is required to manage process authentication. It is recommended not to execute the Web Application processes with an administrator user. You will therefore need a second domain user.

This user will be used as an impersonate user in the web application. It is specifically linked to the feature called "Mega Web Access for Hopex". All the actions carried out in MEGA HOPEX will be done under the identity of this windows user.

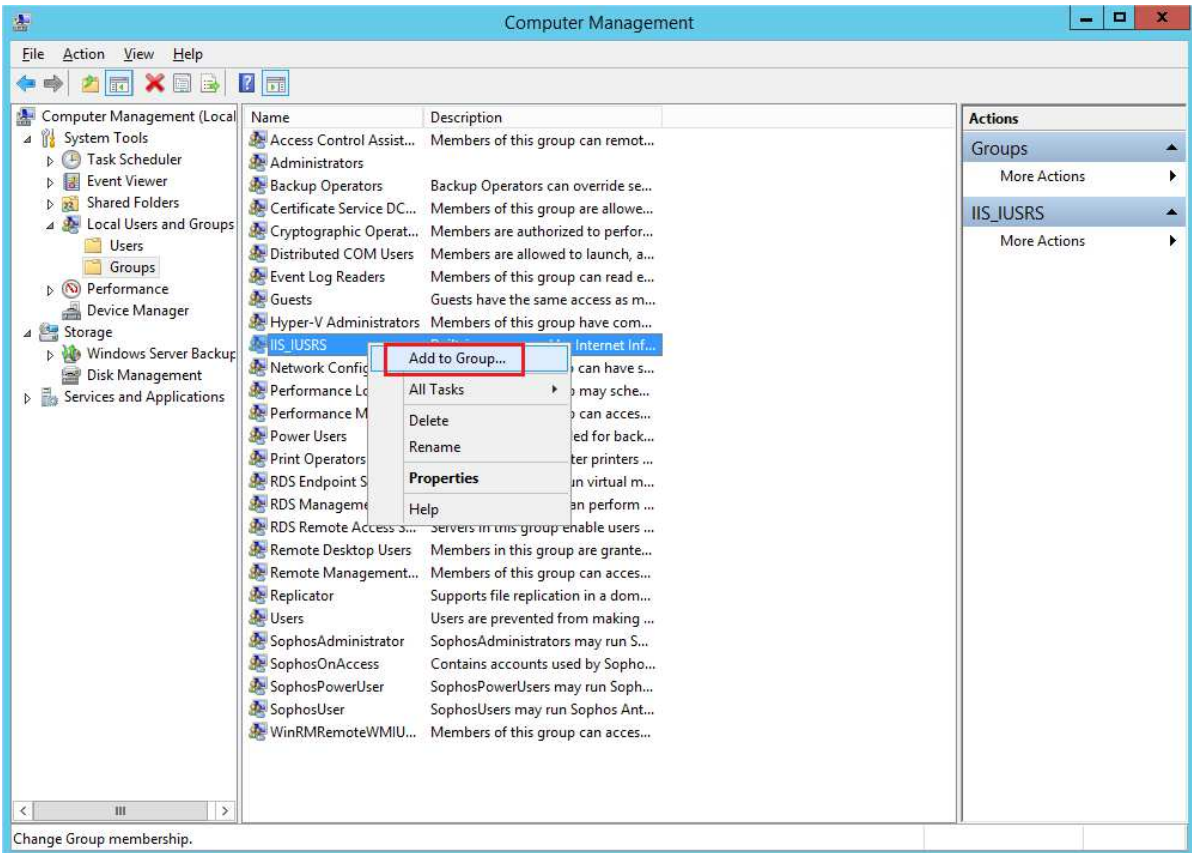
This is only the user that will run some application pools for the web front-end part of the application.

Moreover, if you need to use web services, and thus, you activate the feature called "**HOPEX API**", you will need a second Windows user. It **cannot** be the same as the first one, or it will create side-effects such as navigation issues and errors for users.

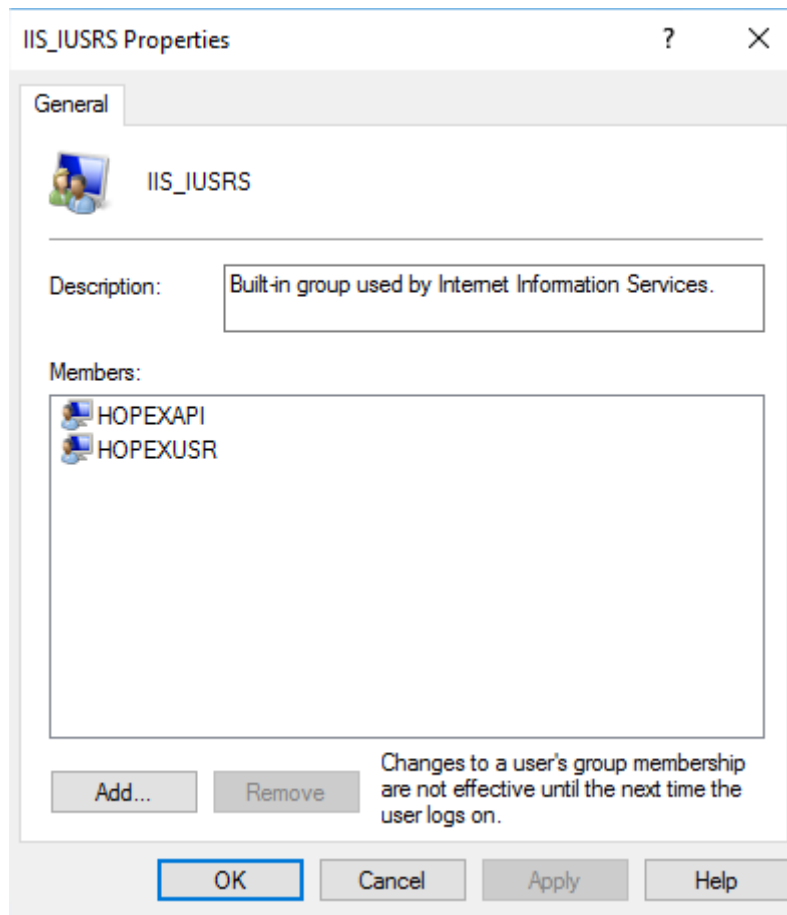
Define Group permissions

This user (or users) must belong to specific windows groups:

- He must belong to the "IIS_IUSRS" group of the server. To add the user to this group, use the "Computer Management" dialog box in the "Tools" of the "Server Manager". Browse to the "Groups" node. Right-click "IIS_IUSRS" and select "Add to Group..."



Example with local users called "...\\hopexusr" and "...\\hopexapi":

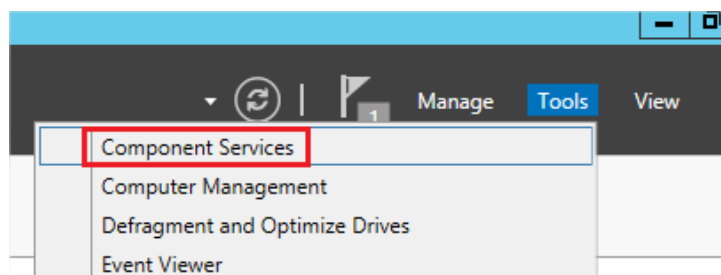


Define MUST Licence Access

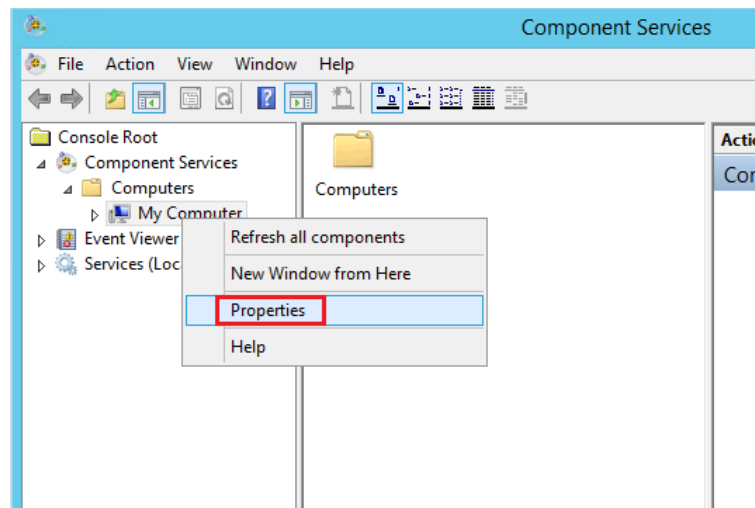
He must be registered in the MUST license tool, have the rights to read and write data in the MUST license folder and to share data. For more details, see the "Must License Installation Guide" technical article located in the Documentation folder of your MEGA installation.

Define COM Access rights

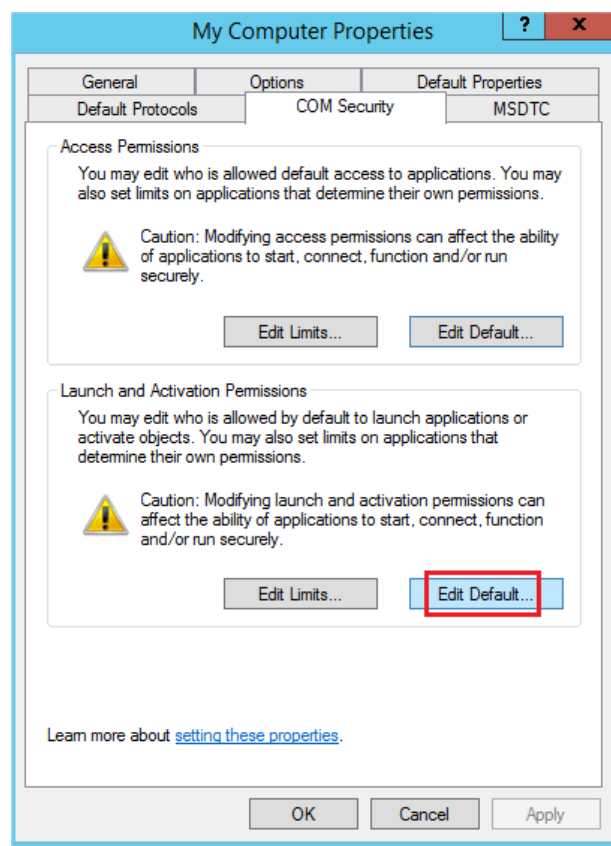
He must be able to launch COM applications by default. To assign this right, proceed from the "Component Services" dialog box through the "Tools" section of the "Server Manager":



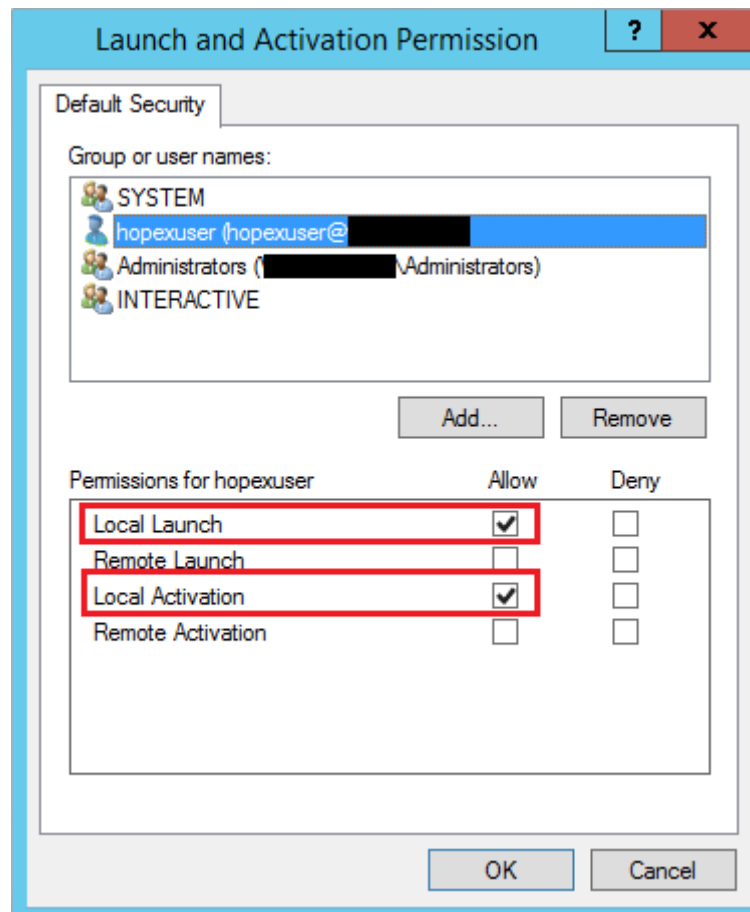
1. Expand the "Component Services" node, then Computers.
2. Right-click "My Computer" and select "Properties".



3. Select the "COM Security" tab and click "Edit Default..." on "Launch and Activation Permissions" group.



4. Add the Windows user, in this example "...\\hopexuser", and give him "Local Launch" and "Local Activation" rights.



MEGA HOPEX SETUP

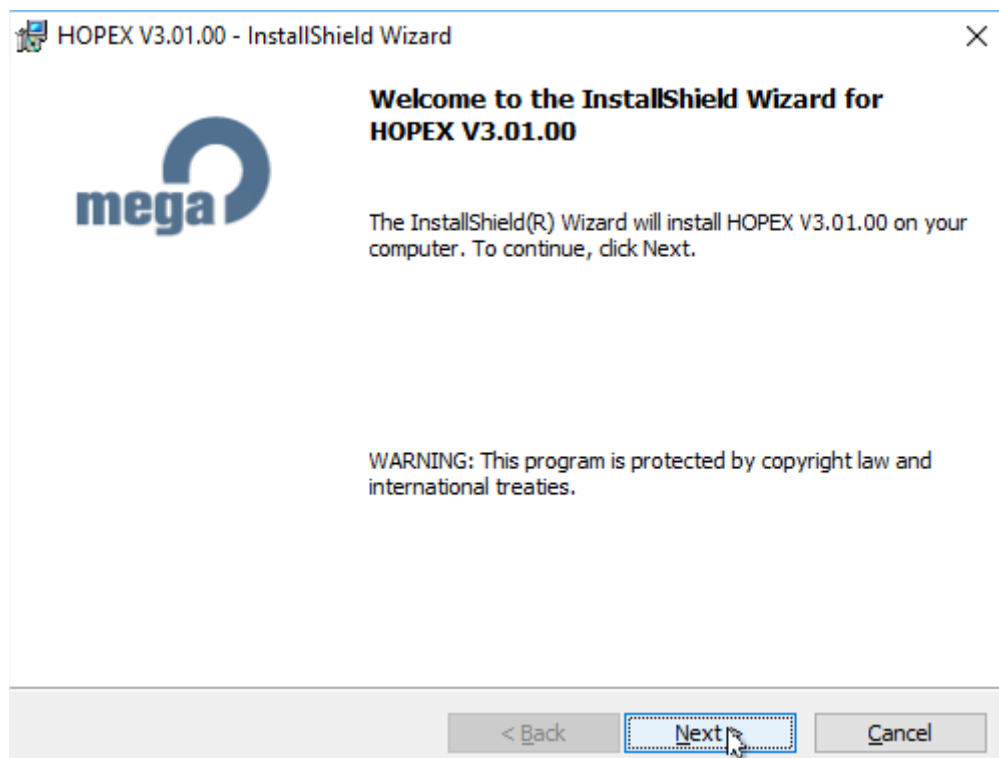
Choosing your setup type

HOPEX offers two ways to install the Web Front-End:

- **Standalone Setup:** automatically installs the Web Front-End and its dependencies (SSP,...) on a single standalone server without SSL/TLS.
- **Advanced Setup:** allows more complex installation scenarios. Use it for any multi-server installation (scale out or scale up), or if you require SSL/TLS.

Web Front-End Standalone Setup

1. Double-click **Setup** to launch the **Setup** program.
2. If prompted, answer "Yes" to "Do you want to allow the following program to make changes to this computer?"
3. Go through each of the following screens:



HOPEX V3.01.00 - InstallShield Wizard

License Agreement

Please read the following license agreement carefully.

**SOFTWARE AGREEMENT FOR END USER MEGA PRODUCTS
REDISTRIBUTION FORBIDDEN**

CAUTION : READ ATTENTIVELY BEFORE USING THIS SOFTWARE
 This agreement concerns the use of certain MEGA products. It is a legal agreement between the Customer (physical or moral person) and MEGA International, for the use of MEGA Products, named hereafter Products. When installing, copying or using the Products, you recognize the formal character of the provisions of this software agreement.
IF YOU DO NOT AGREE WITH THESE PROVISIONS, PLEASE DO NOT INSTALL THE PRODUCTS

☒ I accept the terms in the license agreement
 ☐ I do not accept the terms in the license agreement

Print

InstallShield

< Back Next > Cancel

HOPEX V3.01.00 - InstallShield Wizard

Customer Information

Please enter your information.

User Name:

Organization:

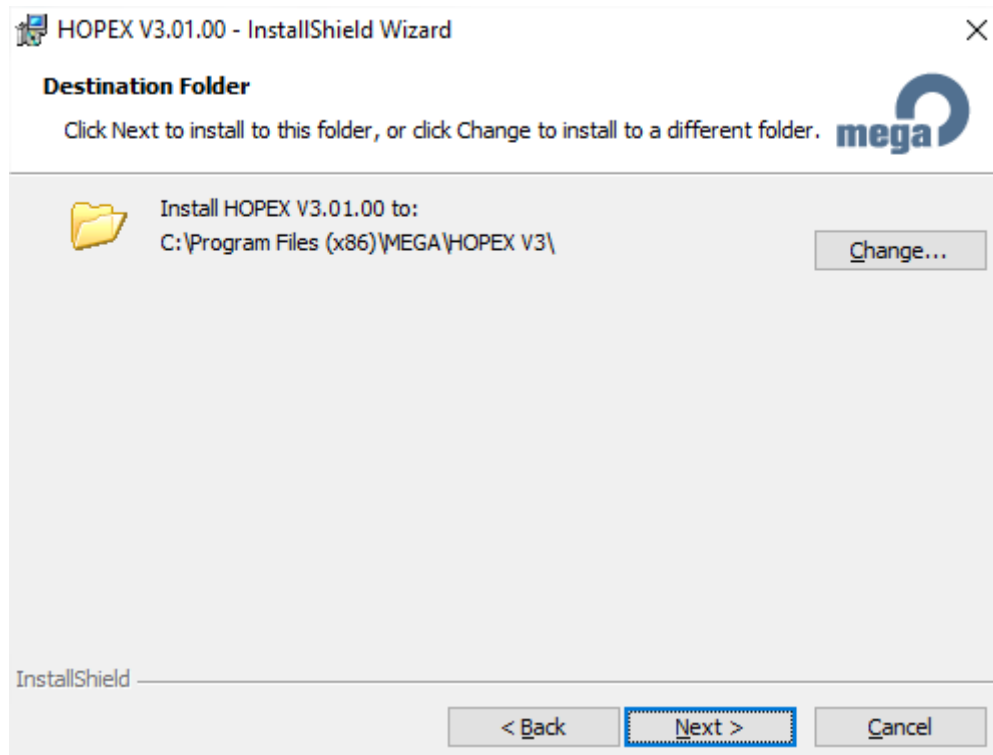
Install this application for:

☒ All system users
☐ Only for me (Windows User)

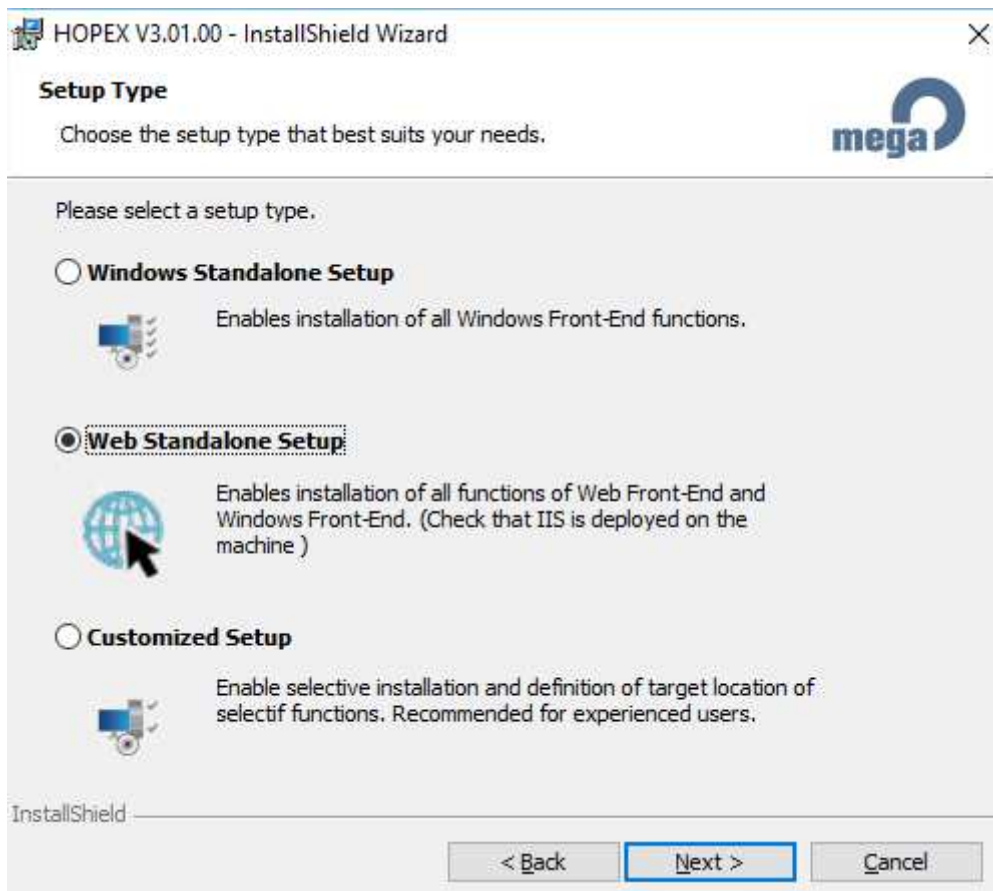
InstallShield

< Back Next > Cancel

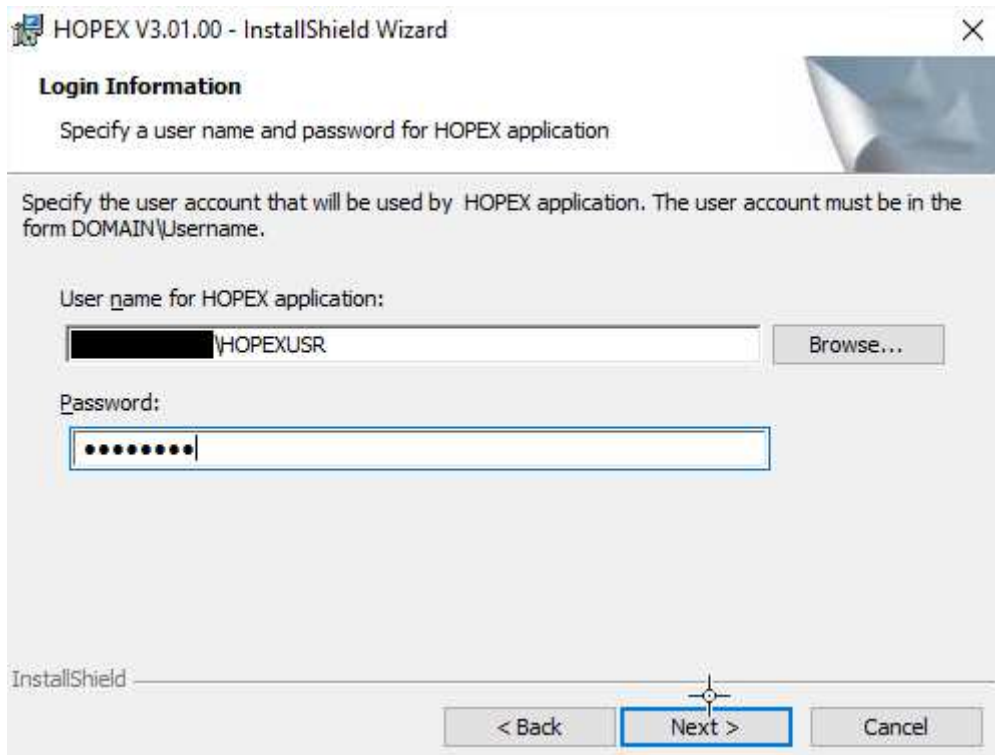
4. If needed, click **Change** to modify the installation folder for the Mega Software, else keep the default one.



5. Select **Web Standalone Setup**.



6. Enter the username and password of the **Windows User for Mega Hopex** you have chosen in the previous section of this document. It will be used for impersonation of the Web application:



HOPEX V3.01.00 - InstallShield Wizard

Login Information

Specify a user name and password for HOPEX application

Specify the user account that will be used by HOPEX application. The user account must be in the form DOMAIN\Username.

User name for HOPEX application:

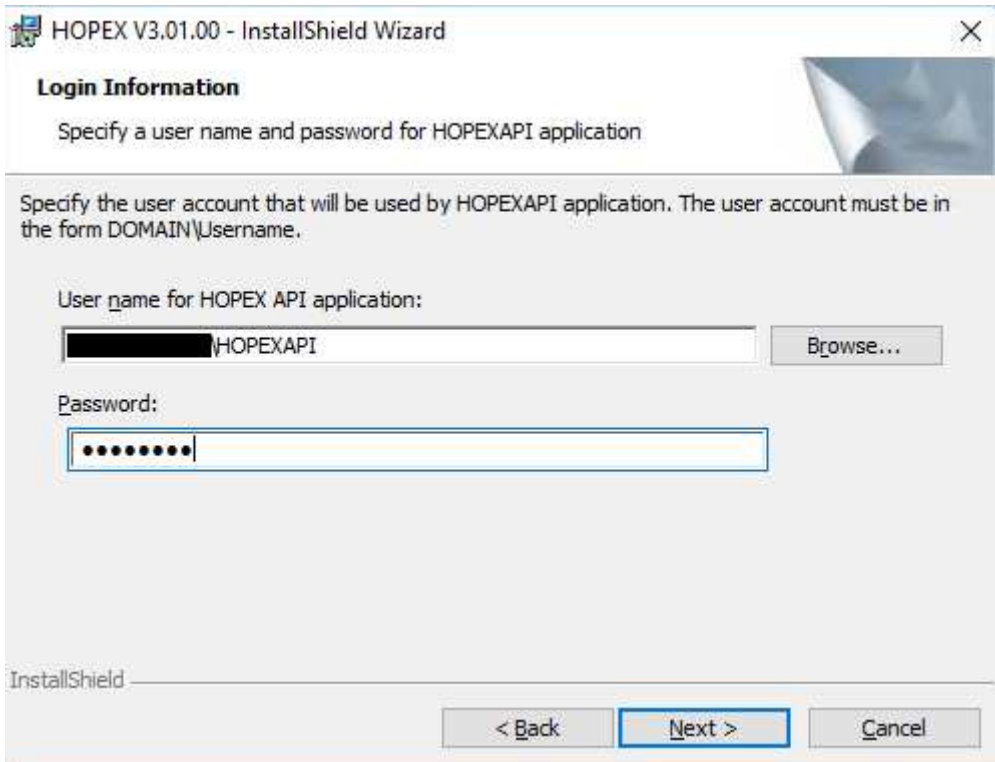
Browse...

Password:

InstallShield

< Back Next > Cancel

7. Enter the username and password of the **Windows User for Mega HOPEXAPI**, the second account that needs to be used in case you deploy the web services part:



HOPEX V3.01.00 - InstallShield Wizard

Login Information

Specify a user name and password for HOPEXAPI application

Specify the user account that will be used by HOPEXAPI application. The user account must be in the form DOMAIN\Username.

User name for HOPEX API application:

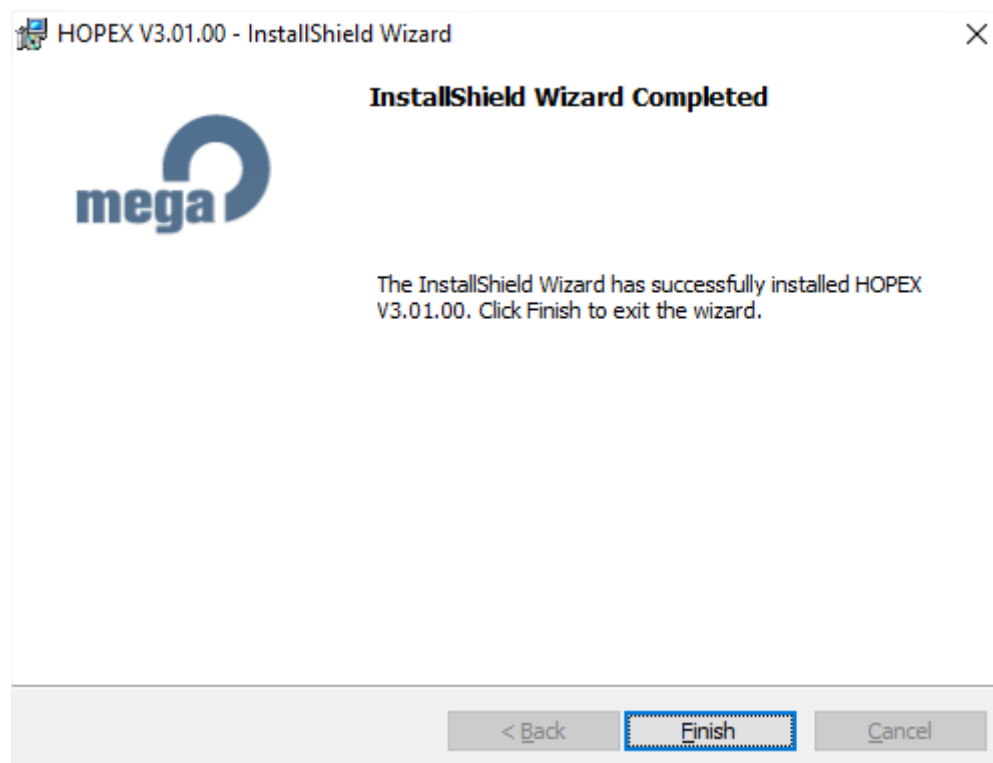
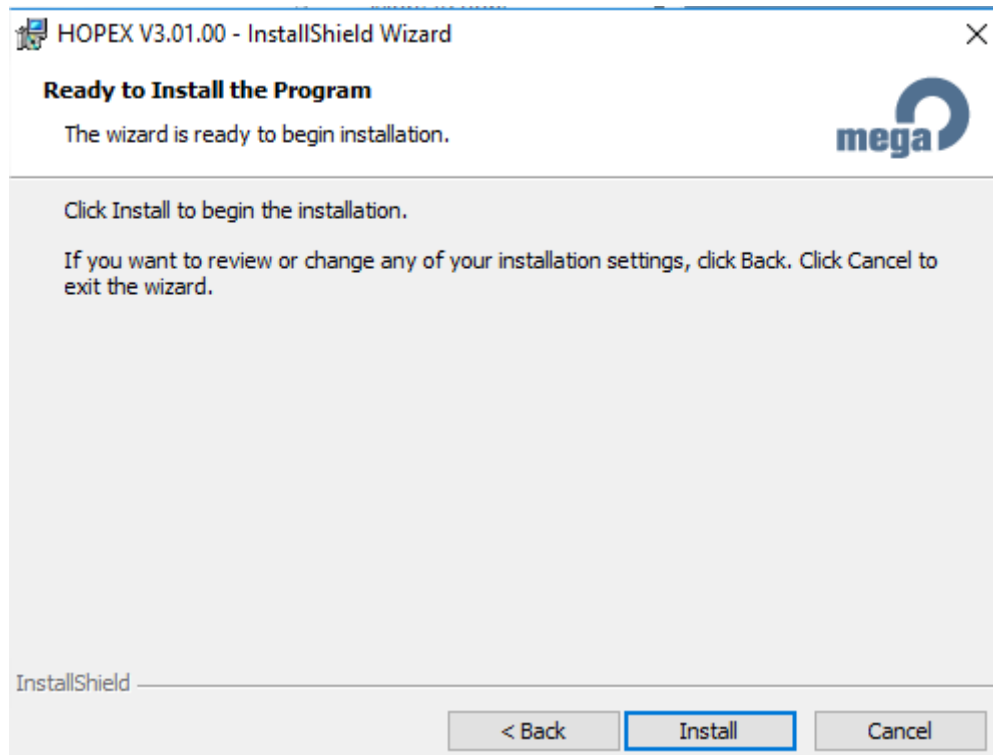
Browse...

Password:

InstallShield

< Back Next > Cancel

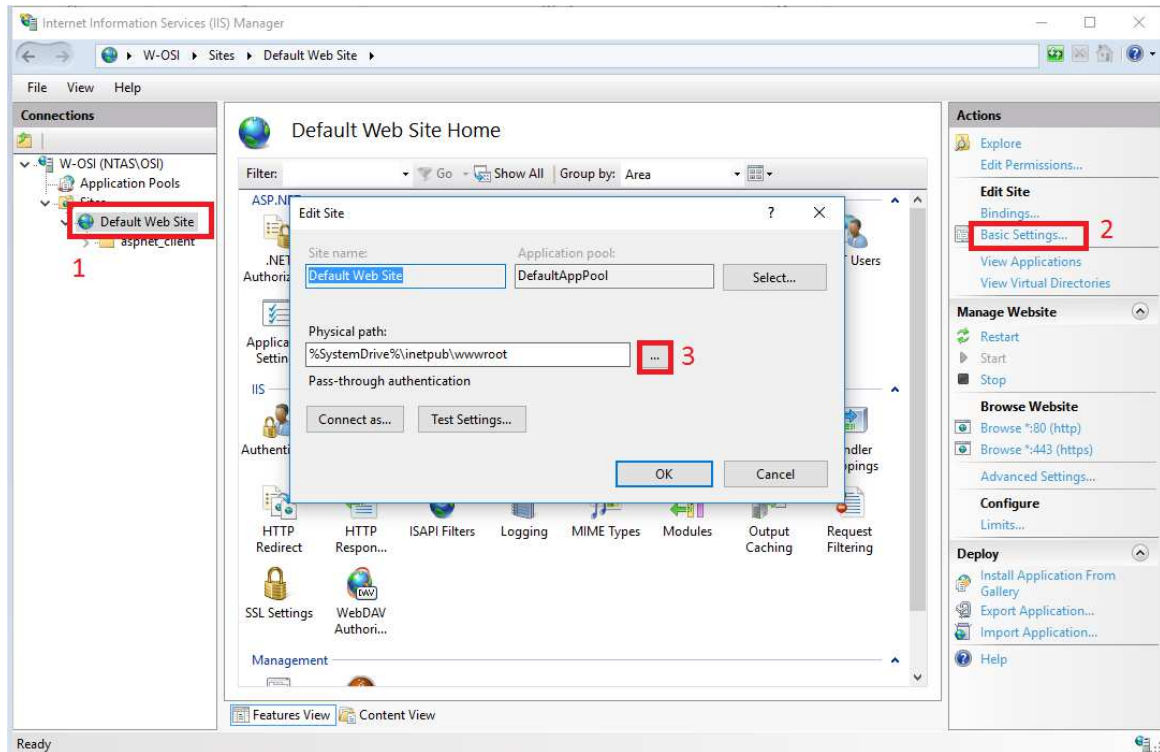
8. You are now ready to launch the installation by clicking **Install** :



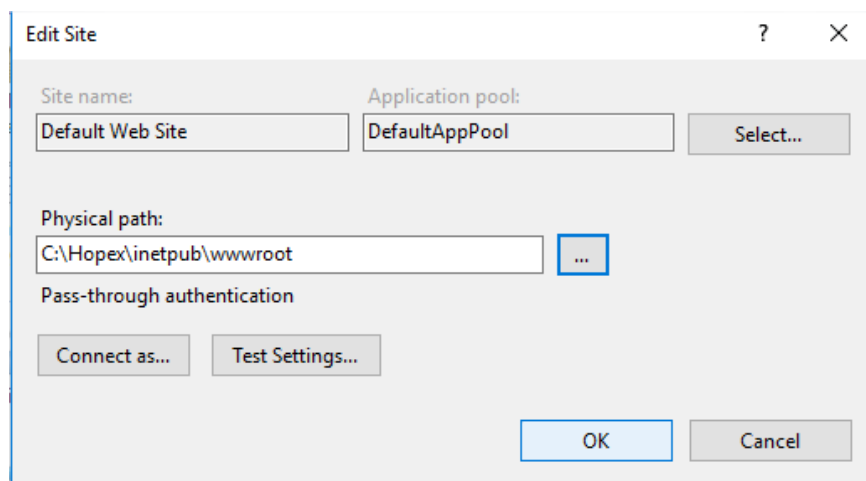
Advanced Setup

Advanced Setup is similar to the standalone setup, except for the initial choice screens and extra parameter choices.

Remark: if you plan to install the web components in a different location than “C:\inetpub\wwwroot”, you need to first configure the root directory of the website where you will install it. Here, an example for the “Default Web Site”:



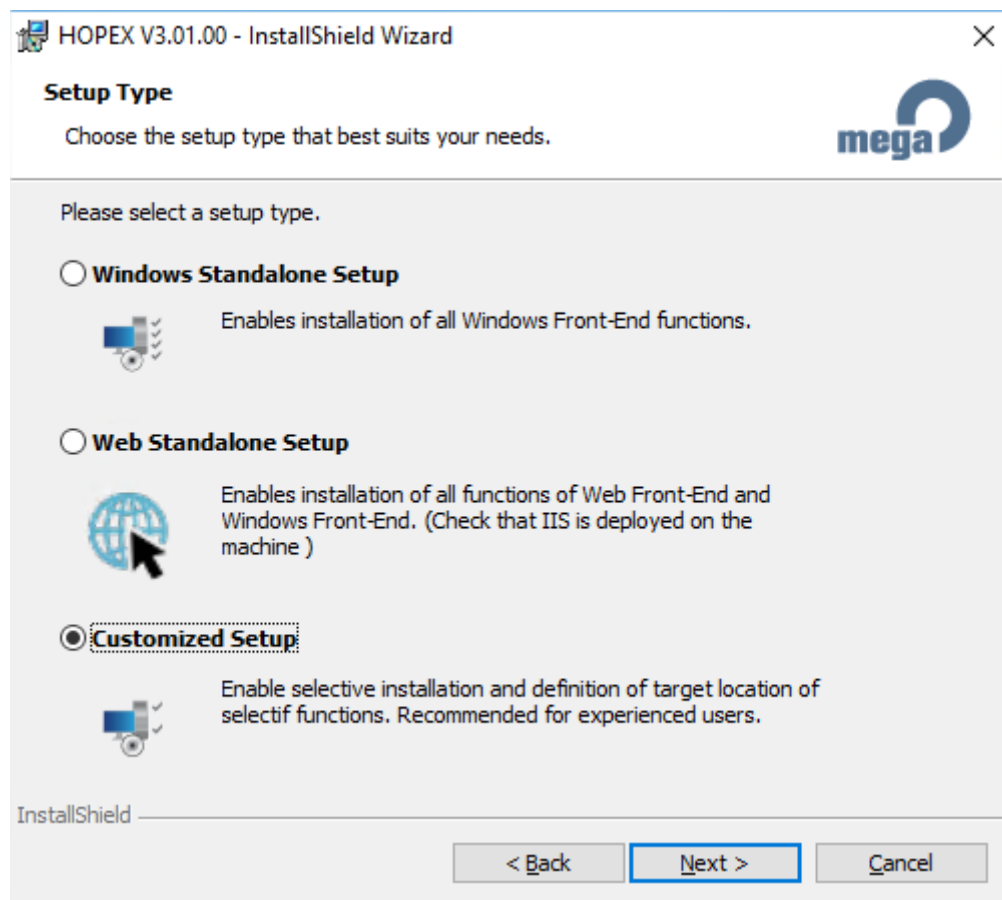
Browse to the folder where you want to install, and validate:



Do not forget that you will need to grant the same permissions for the same users on that folder, following the steps of “Define “Windows User for MEGA HOPEX” files Access Rights” in a later section of this document.

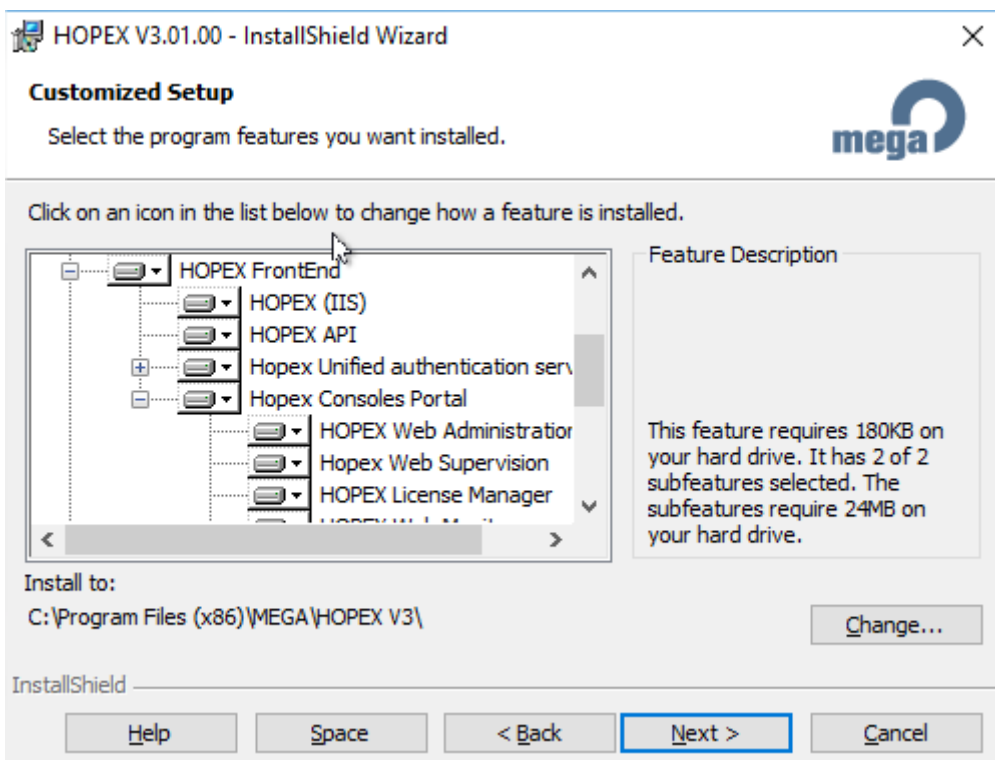
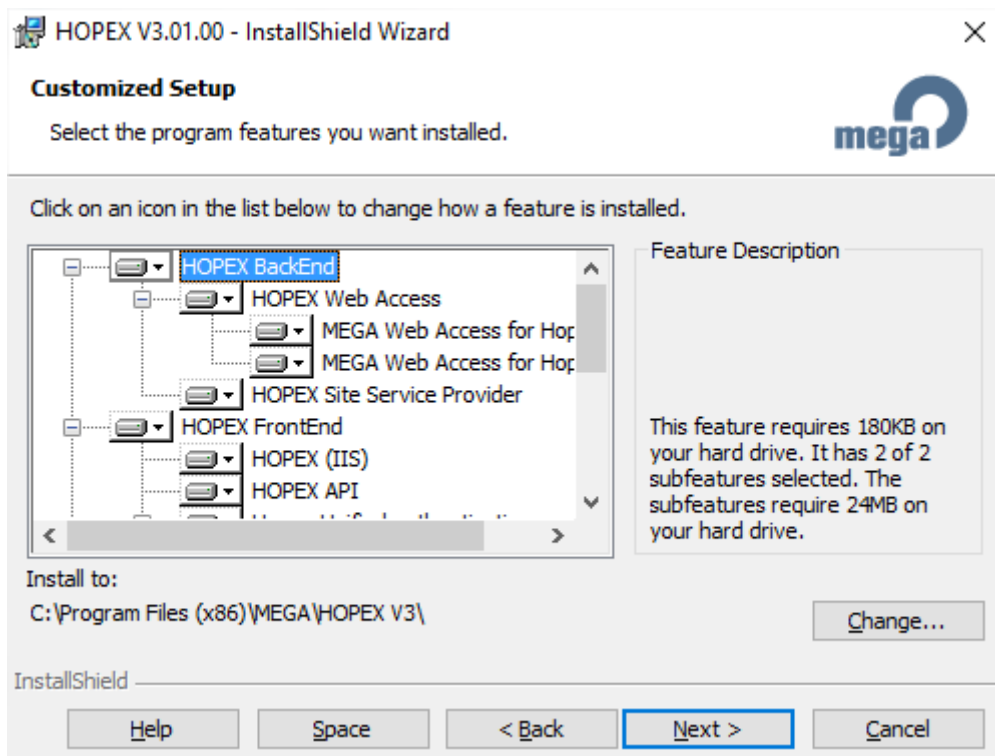
Choice Screen

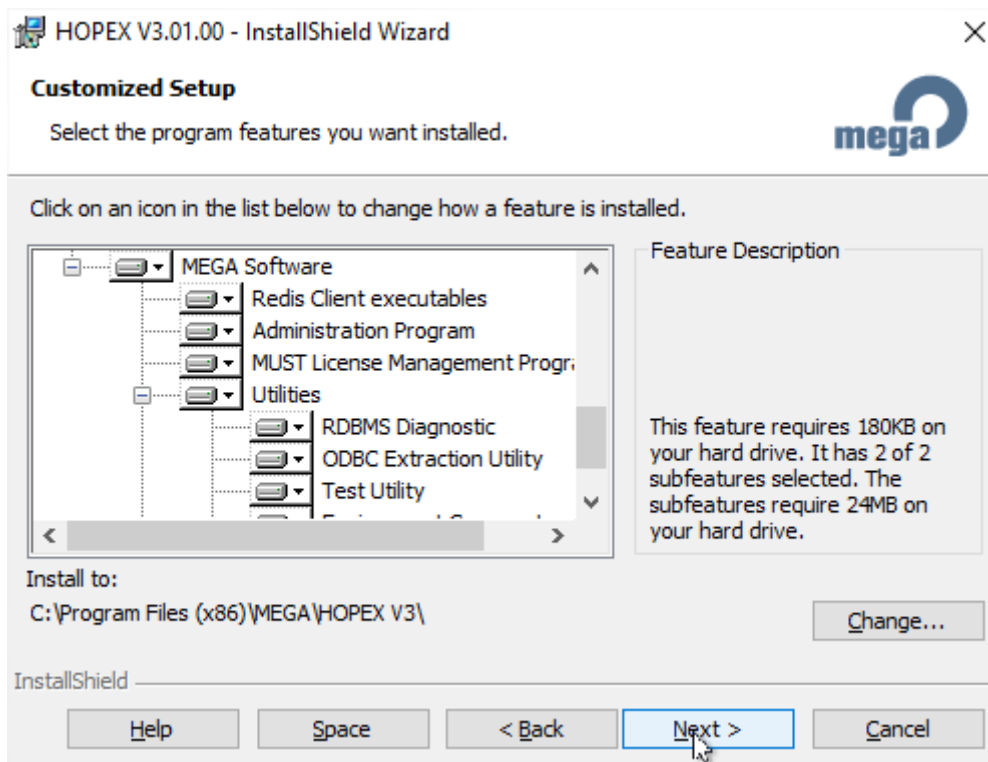
1. Choose **Customized Setup**:



2. Choose the features that you want to install. Depending on how many servers you have, and the type of deployment you choose (see the "Web Front-end Architecture Overview" document), you need at least :
 - o HOPEX FrontEnd and its subfeatures
 - o HOPEX BackEnd, wih at least:
 - "Hopex Web Access" -> "Mega Web Access for Hopex"
 - Hopex Site Service Provider
 - o The "Mega Software" suite, already activated by default

In this example we install all features on a single server, including the web services:

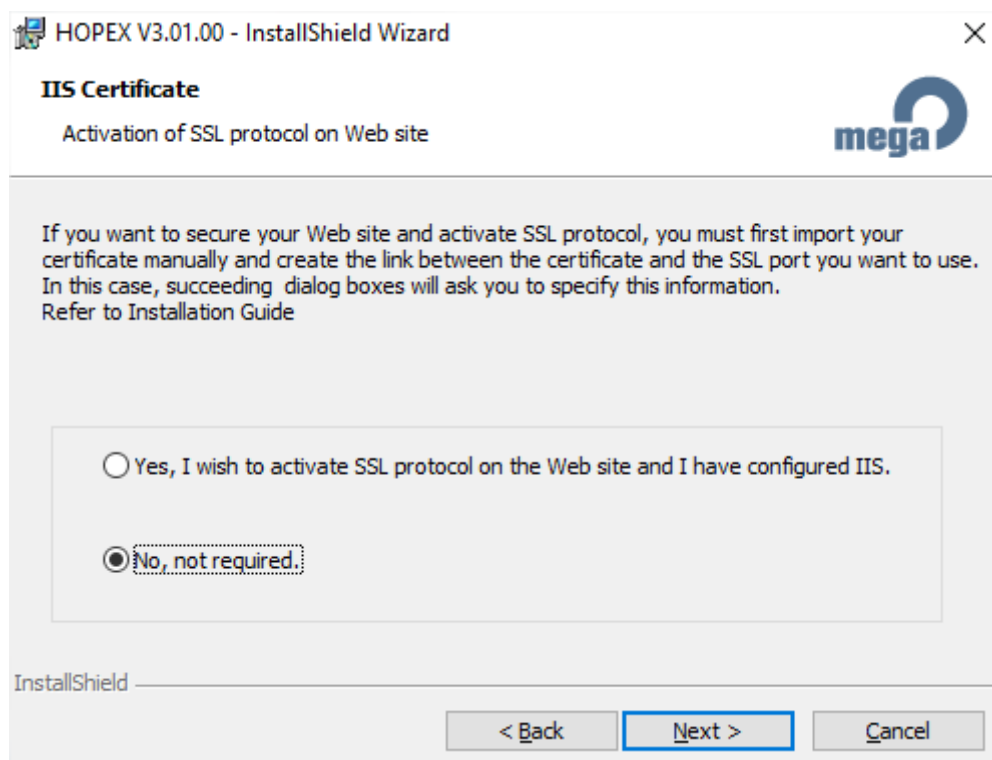




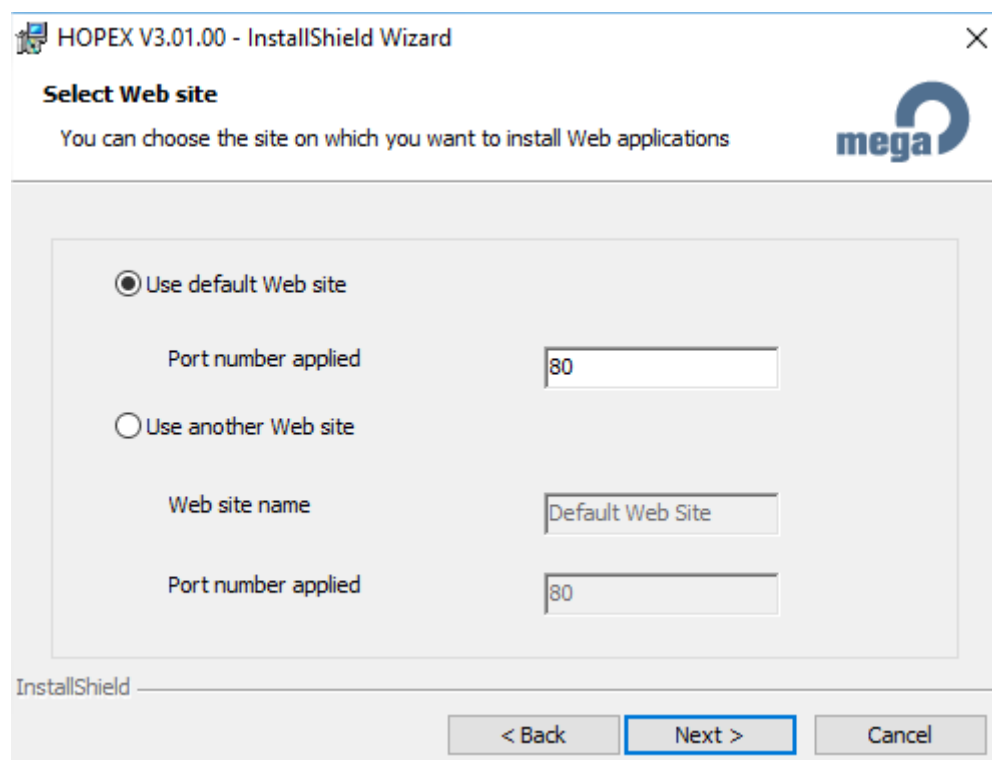
Note that by selecting each item, you can click **Change** to modify the installation location of the files that are linked to a specific feature.

3. The "HOPEX API" feature, above, is needed when you have to add web services on the platform. It requires another Windows user, different from the one used by the "Mega Web Access for Hopex" feature. You will also need to activate the "Mega Web Access for Hopen API" in the "HOPEX BackEnd" section.
4. Choose to activate use of SSL/TLS or not. SSL/TLS is highly recommended; however, it requires some prior configuration of IIS (see Prerequisites section). **Moreover, do not choose "Yes" unless the certificate is deployed in IIS. Otherwise, the installation will roll back.**

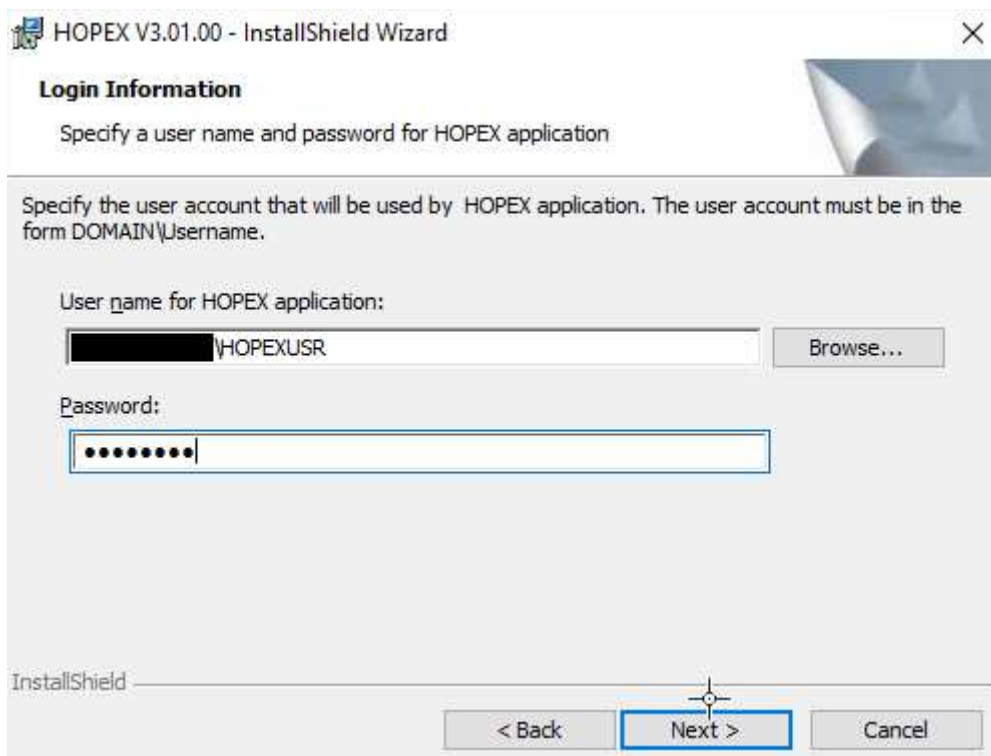
In this example, we do not have HTTPS activated on our website:



5. Choose the IIS web-site where you want to install the Hopex Web Front-End. **Please note** that the installer will check that there is a website running on the chosen port, and will install in the first site using that port that is available. So make sure to properly manage your websites and their ports before choosing one for the Hopex deployment. If you choose a port that is bound to no site, and that isn't listening during the installation, the setup will roll back :

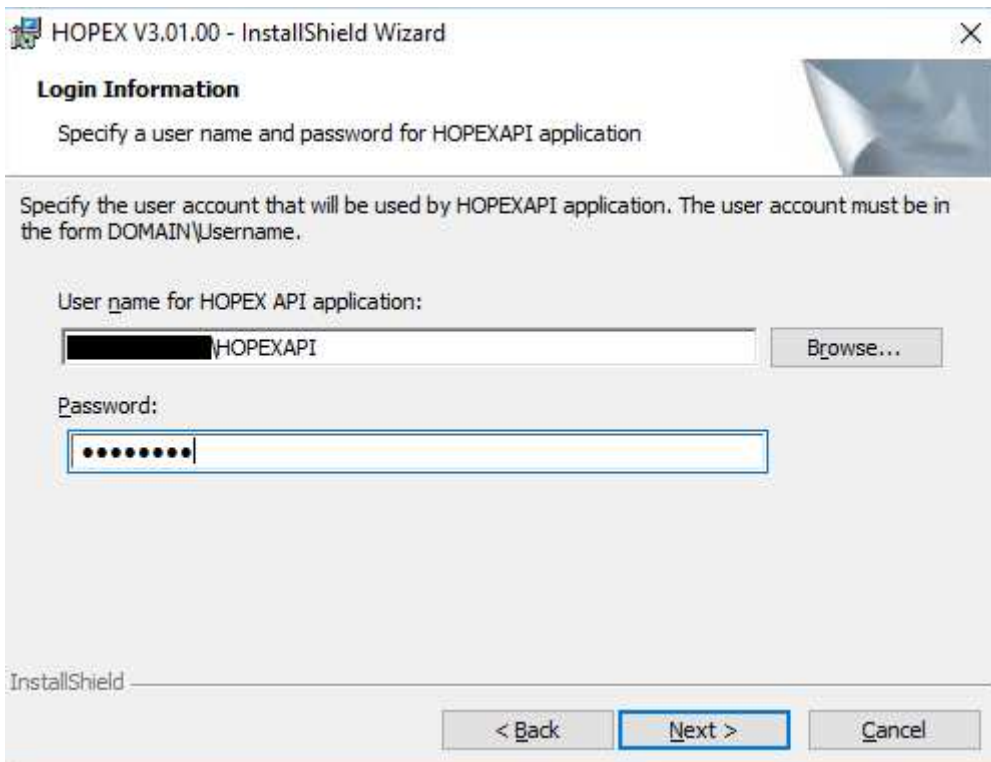


6. Enter the username and password of the **Windows User for Mega Hopex** you have chosen in the previous section of this document. It will be used for impersonation of the Web application:



The screenshot shows the 'Login Information' step of the 'HOPEX V3.01.00 - InstallShield Wizard'. The title bar includes the application icon and text, and a close button. The main heading is 'Login Information' with a subtitle 'Specify a user name and password for HOPEX application'. Below this, a note states: 'Specify the user account that will be used by HOPEX application. The user account must be in the form DOMAIN\Username.' There are two input fields: 'User name for HOPEX application:' containing 'DOMAIN\HOPEXUSR' with a 'Browse...' button to its right, and 'Password:' containing a masked password '••••••••'. At the bottom, there are three buttons: '< Back', 'Next >' (highlighted with a blue border), and 'Cancel'. The 'InstallShield' logo is visible in the bottom left corner.

7. Enter the username and password of the **Windows User for Mega HOPEXAPI**, the second account that needs to be used in case you deploy the web services part:



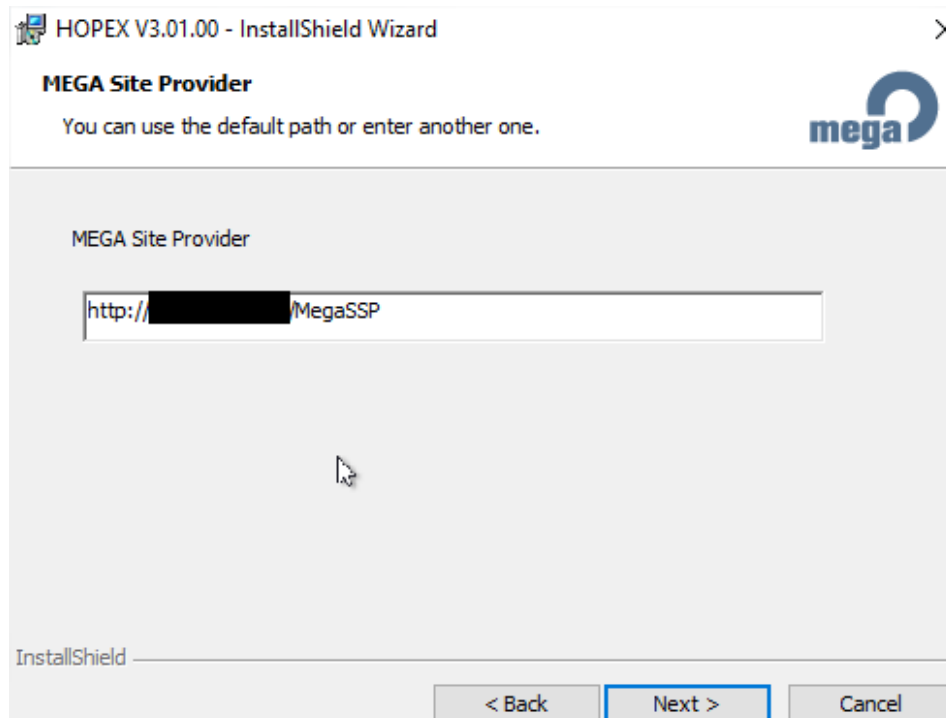
This screenshot is similar to the previous one, showing the 'Login Information' step for the 'HOPEXAPI' application. The title bar and main heading are the same. The subtitle is 'Specify a user name and password for HOPEXAPI application'. The note about the user account format is also present. The 'User name for HOPEX API application:' field contains 'DOMAIN\HOPEXAPI' with a 'Browse...' button. The 'Password:' field contains a masked password '••••••••'. The bottom buttons are '< Back', 'Next >' (highlighted with a blue border), and 'Cancel'. The 'InstallShield' logo is in the bottom left corner.

Advanced Parameters

Parameters depend of the components you have chosen to install. Here is the list of parameters that are not proposed in the standalone setup.

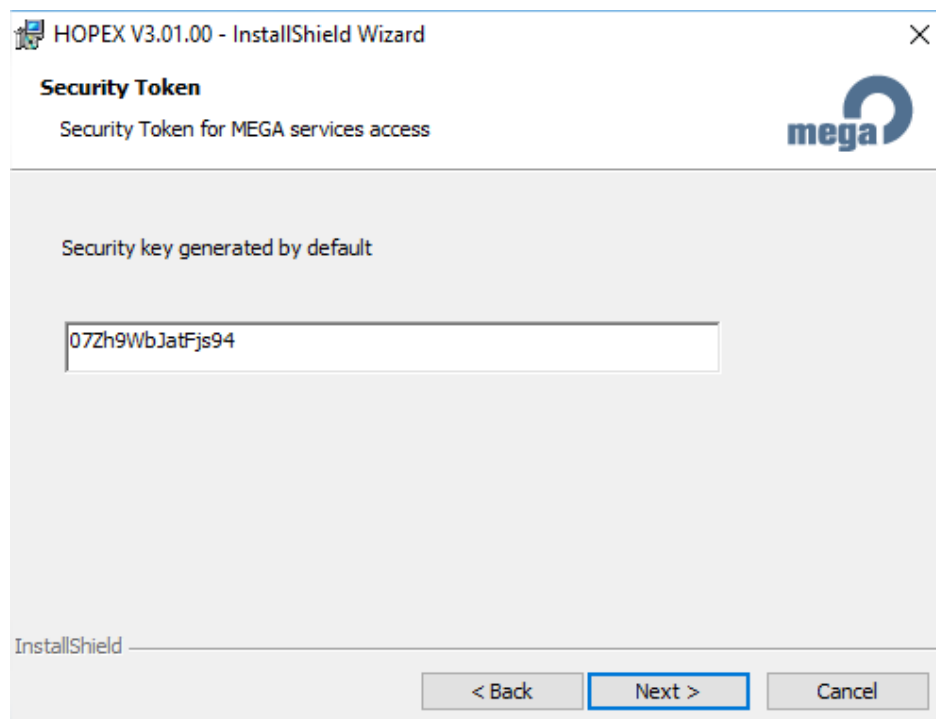
Mega Site Provider (SSP) URL

On a standalone deployment, this will contain the local name of the server. It has to be an address that can be accessed from within the server itself:



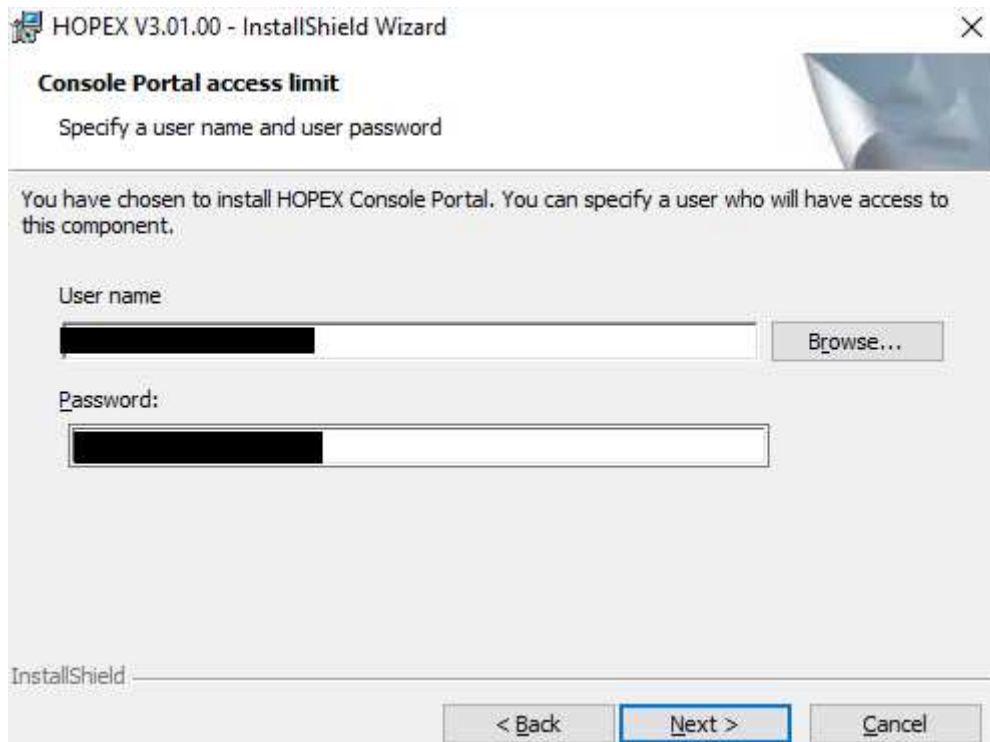
Security Token

It should be **identical on all Hopex installs** (Web, Windows, SSP,...) that work together in a scale up or scale out scenario



Control Portal access limit

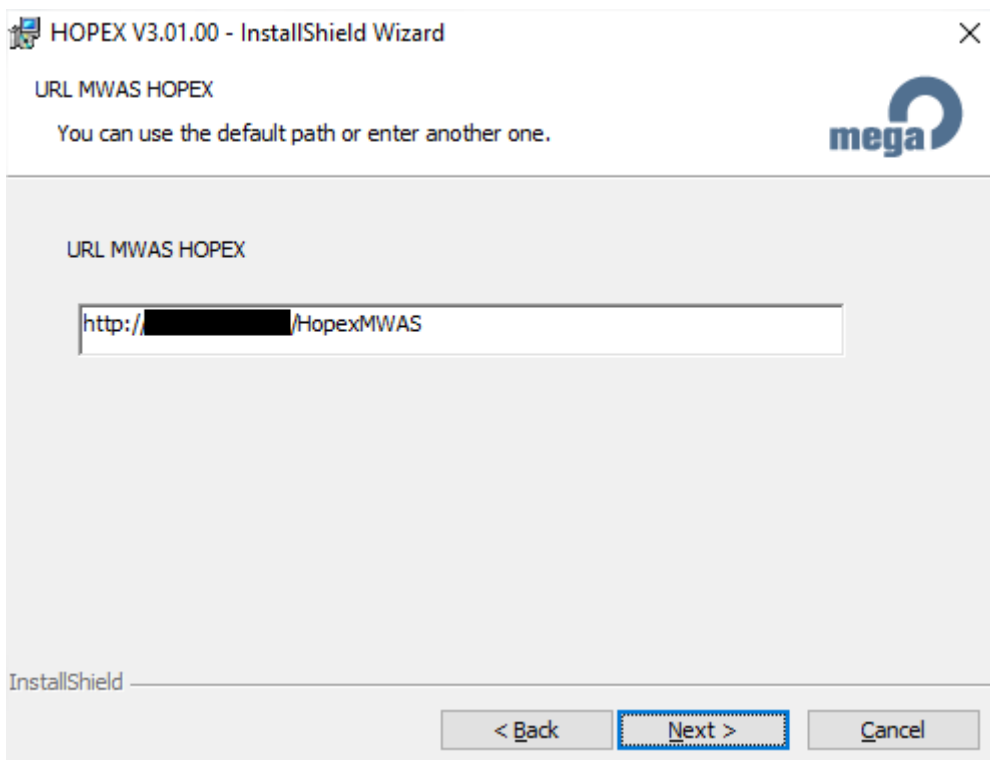
You provide the credentials of the user that can access the new Web Consoles Portal, if activated. If more than one user needs to access it, you will need to make additional IIS configuration:



The screenshot shows a Windows installer window titled "HOPEX V3.01.00 - InstallShield Wizard". The main heading is "Console Portal access limit" with the instruction "Specify a user name and user password". Below this, a message states: "You have chosen to install HOPEX Console Portal. You can specify a user who will have access to this component." There are two input fields: "User name" and "Password:". The "User name" field has a "Browse..." button to its right. At the bottom, there are three buttons: "< Back", "Next >" (which is highlighted with a blue border), and "Cancel". The "InstallShield" logo is visible in the bottom left corner.

URL of HOPEX MWAS Web Site

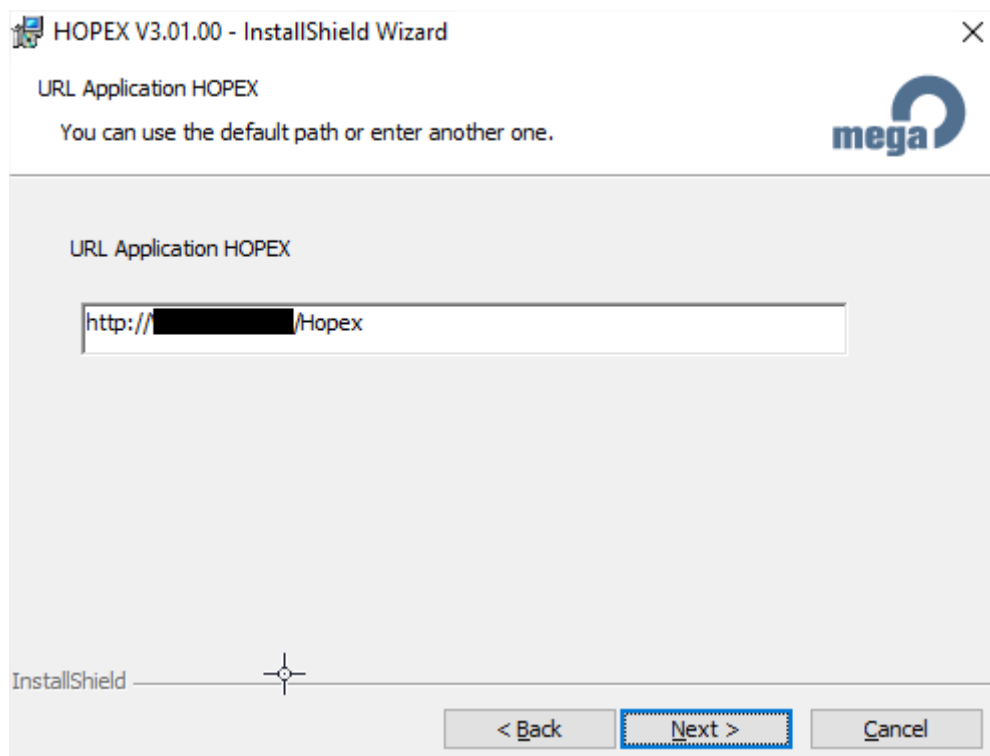
Defined on a web server to indicate the application server



The screenshot shows a Windows installer window titled "HOPEX V3.01.00 - InstallShield Wizard". The main heading is "URL MWAS HOPEX" with the instruction "You can use the default path or enter another one." The "mega" logo is in the top right corner. Below the heading, there is a text input field containing the URL "http://[redacted]/HopexMWAS". At the bottom, there are three buttons: "< Back", "Next >" (highlighted with a blue border), and "Cancel". The "InstallShield" logo is visible in the bottom left corner.

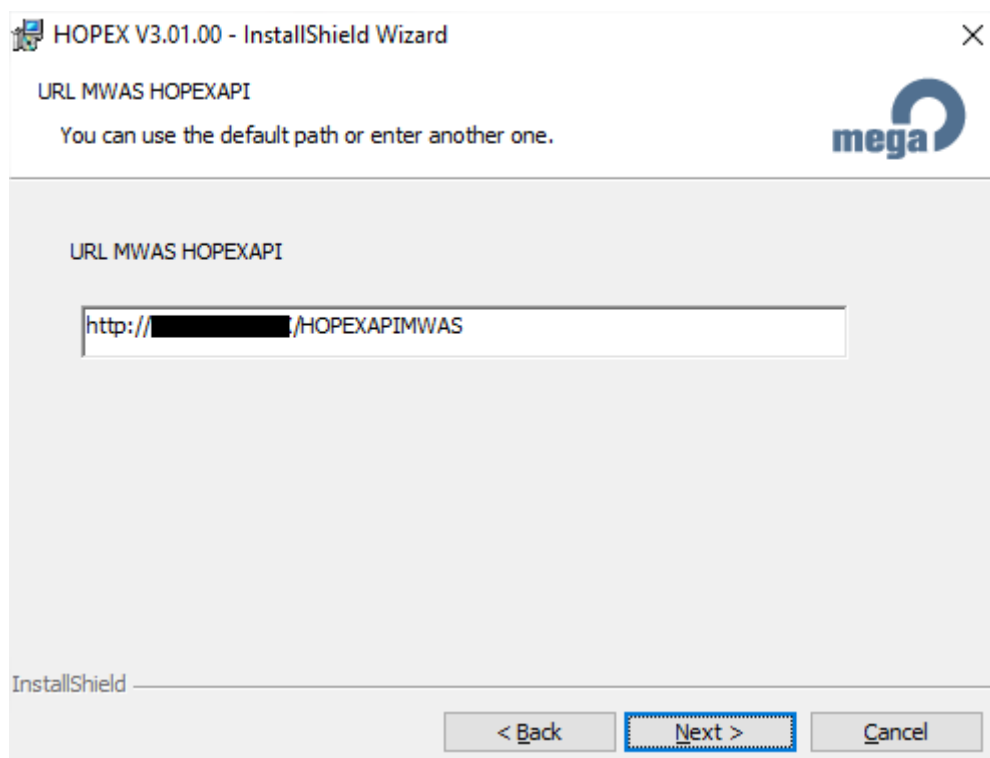
URL of HOPEX Web Site

Defined on an/the application server to target the web server:



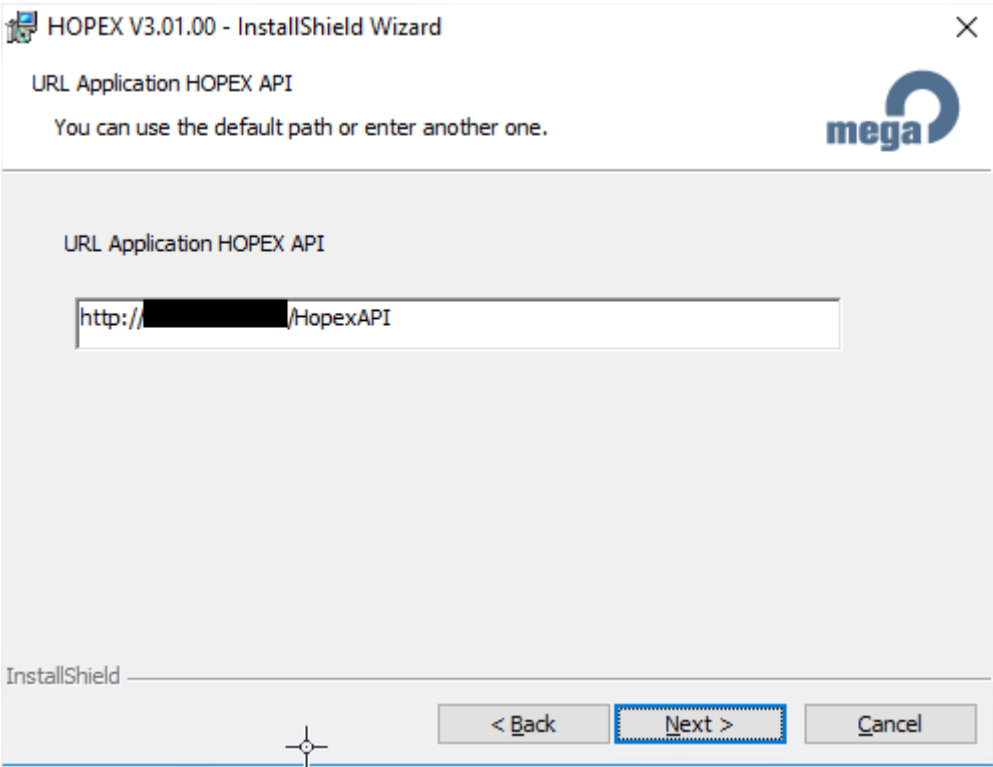
URL of HOPEXAPI MWAS Web Site

Defined on a web server to indicate the application server containing the HOPEXAPI features:



URL of HOPEXAPI Web Site

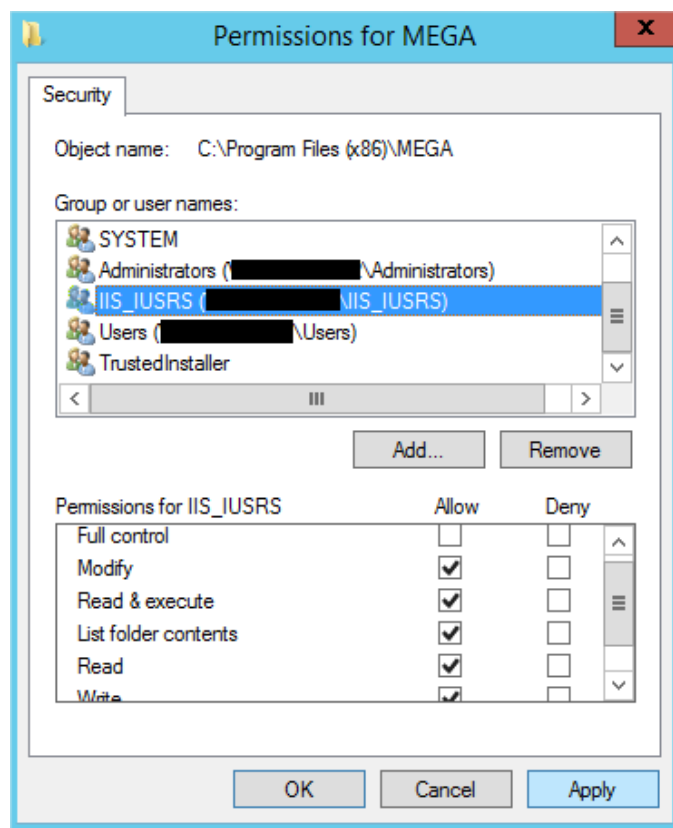
Defined on an/the application server to target the web server containing the web application of the HOPEXAPI feature:



COMPLETING INSTALLATION

Define "Windows User for MEGA HOPEX" files Access Rights

- Go to the **installation folder of MEGA HOPEX** (By default, "C:\Program Files (x86)\MEGA\HOPEX V2R1" on 64 bits systems) and give read/write access rights to the **IIS_IUSRS** group, that contains your Windows User that does the impersonation. This way, if you change that user, you won't have to change that security, but just update the above group:



- Repeat this operation for:
 - The **installation folder of MEGA HOPEX Web Front-End** (by default at C:\Inetpub\wwwroot\hopex)
 - The **environments folders** to be reached through MEGA. By default, environments are created in sub-directories of C:\Users\Public\Documents.
 - The **Temporary folder** used for Web Access. Usually, it is at C:\Windows\Temp.

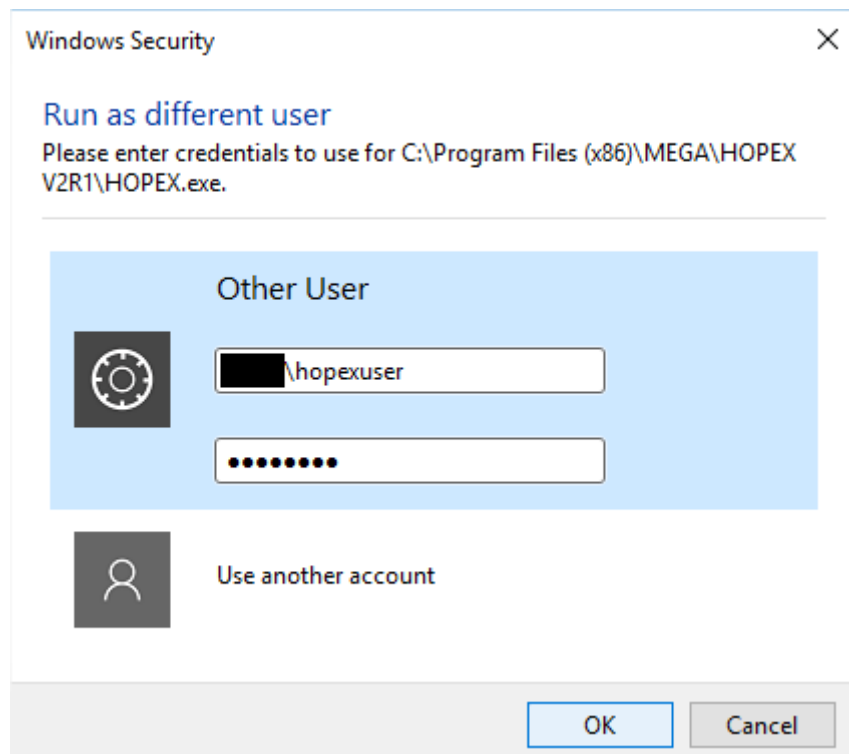
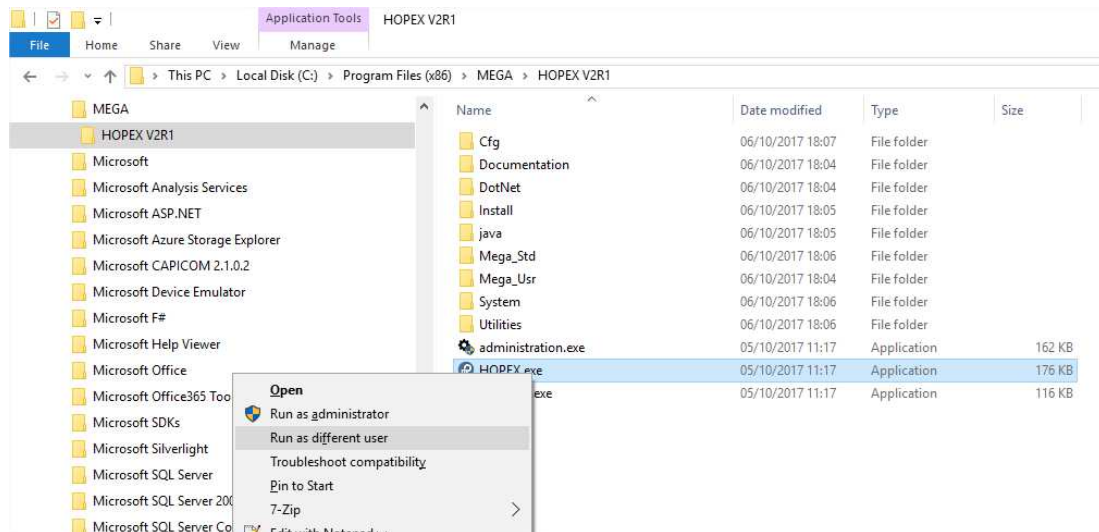
Make sure your user is correctly configured for the MUST license (for more details see the "Must License Installation Guide" technical article located in the Documentation\Articles folder of your installation).

As a first test, run HOPEX application as the "windows user for MEGA HOPEX".

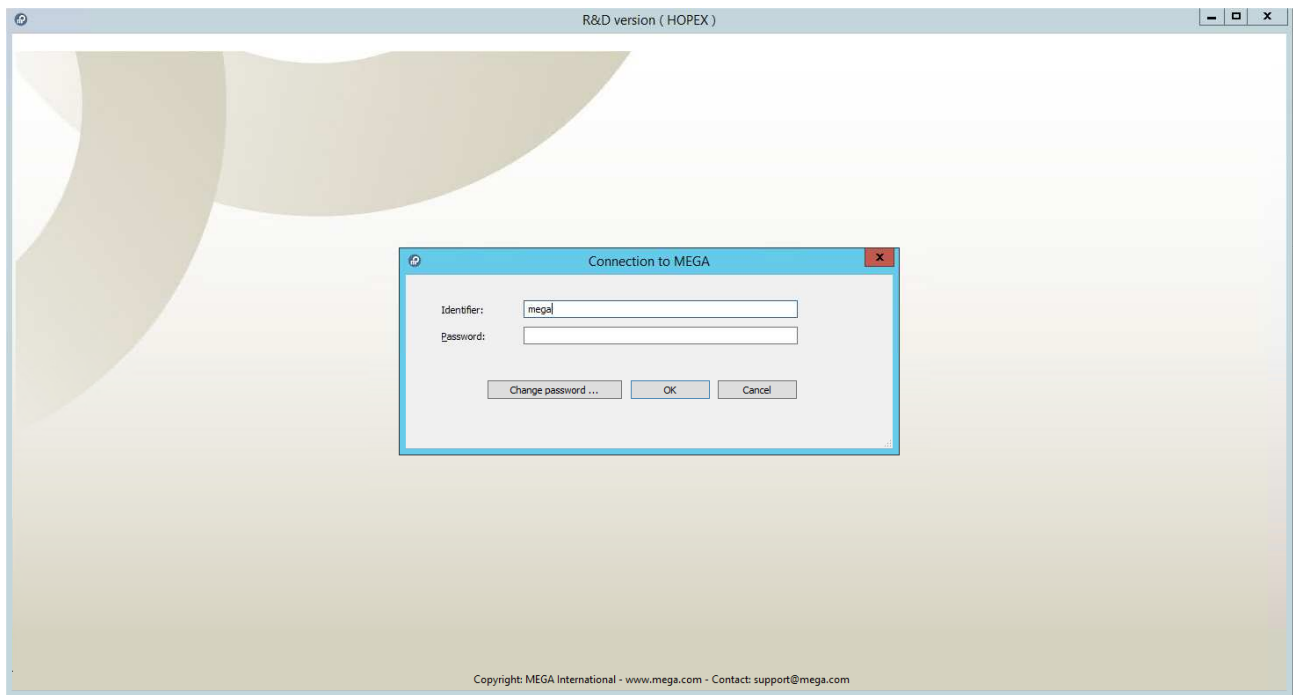
Prerequisite: a Mega HOPEX environment is referenced on your platform. An environment that you created in a separate step.

To do so, go into the Mega installation folder.

Hold down the "Shift" button, right-click Hopex.exe and select "Run as different user":



You must be able to launch MEGA.



Tune IIS

A default option of IIS makes the worker process of the HOPEX/HOPEX2 recycle every 29 hours. This is what Microsoft chose to make sure that the w3wp.exe process, that manages websites and/or web applications within websites, to be stable.

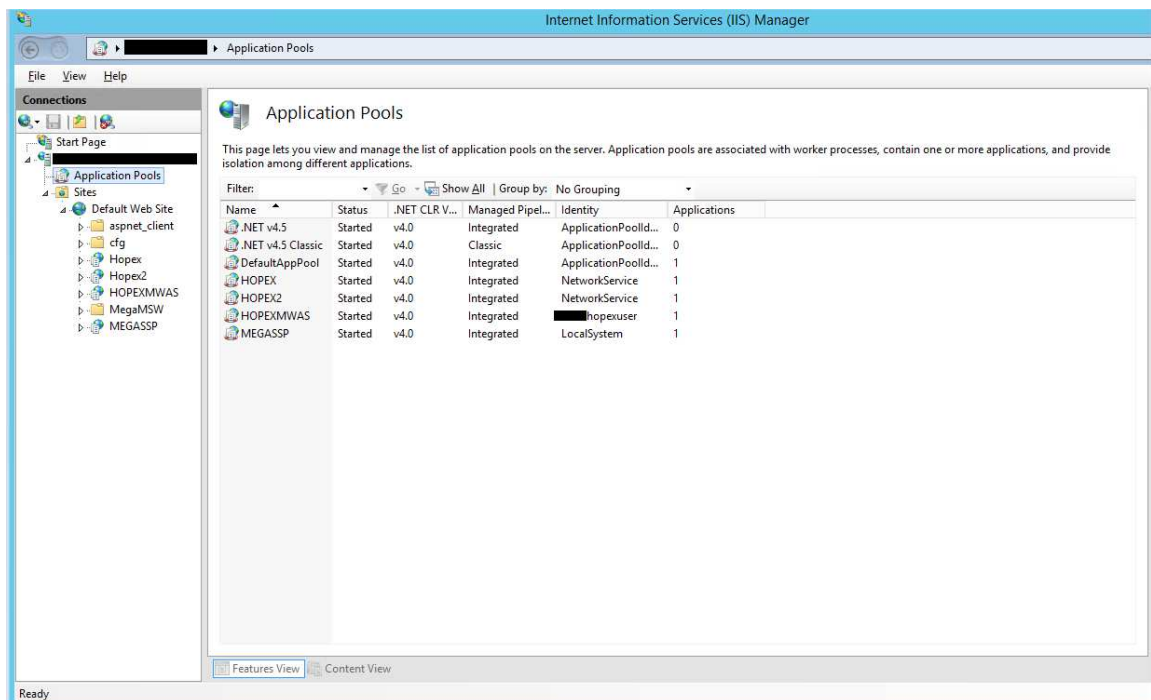
However, the fact that this process restarts, makes the browsing of connected users to fail, as they lose their browsing context.

To avoid that, the installer disables completely this recycling, and use the default idle timeout, that says that after 20 minutes of complete inactivity, the worker process will stop (it will automatically start next time someone tries to access the website).

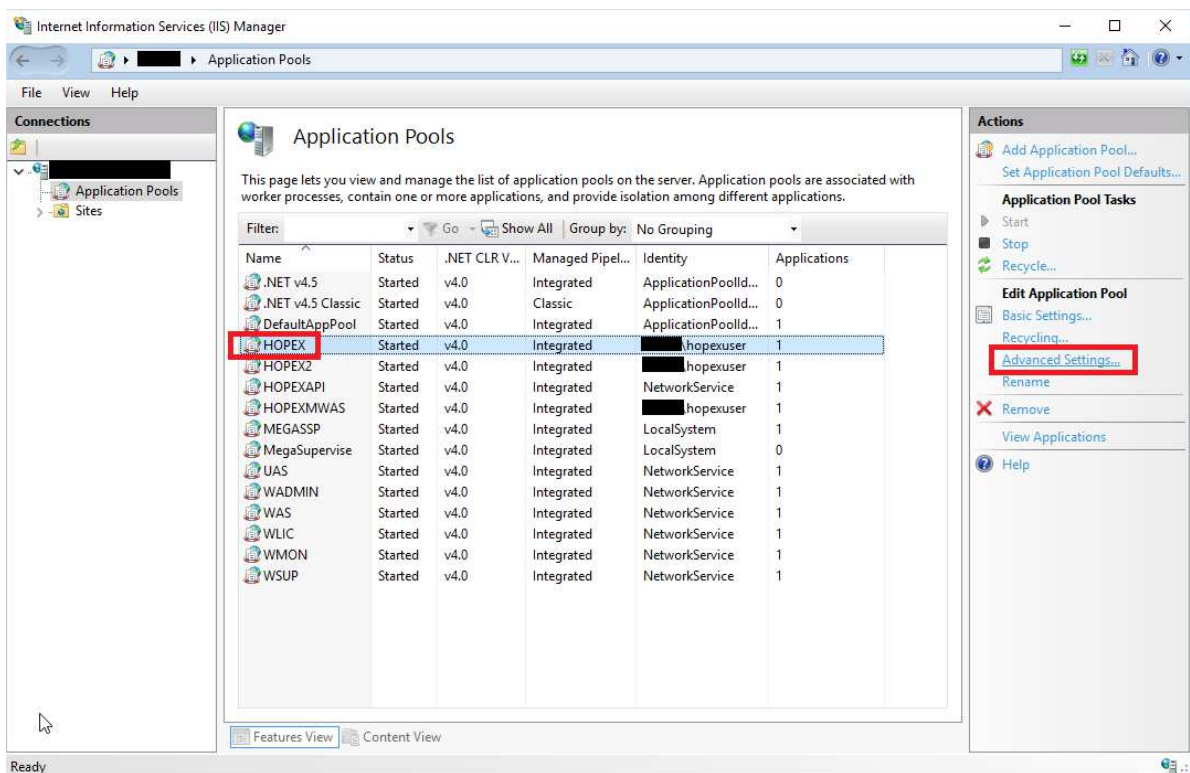
We recommend to manually add a restart at a fixed hour, when you know that no one will actually be connected on the website, to be sure it restarts at least once a day.

To do this.

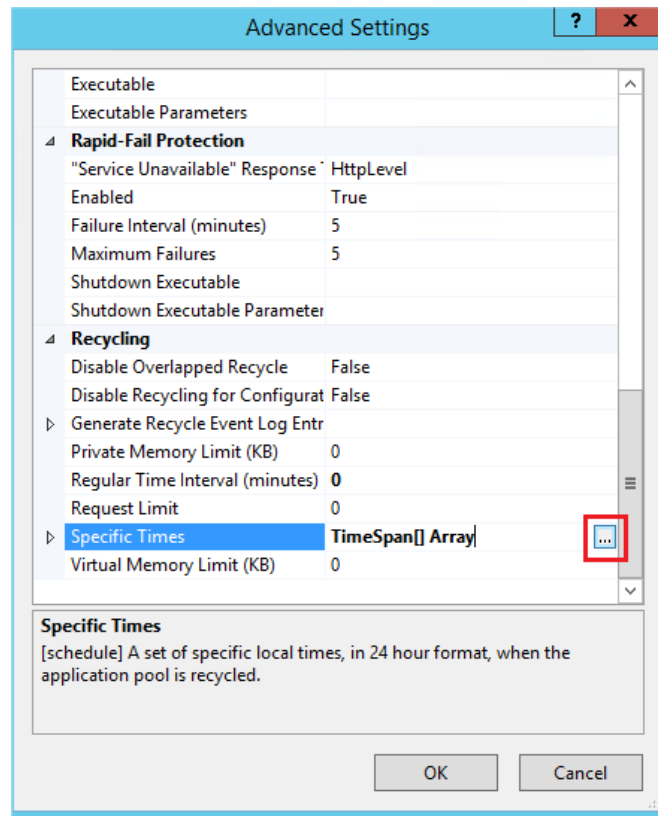
1. Open the "Internet Information Services (IIS) Manager", and go to **Application Pools**:



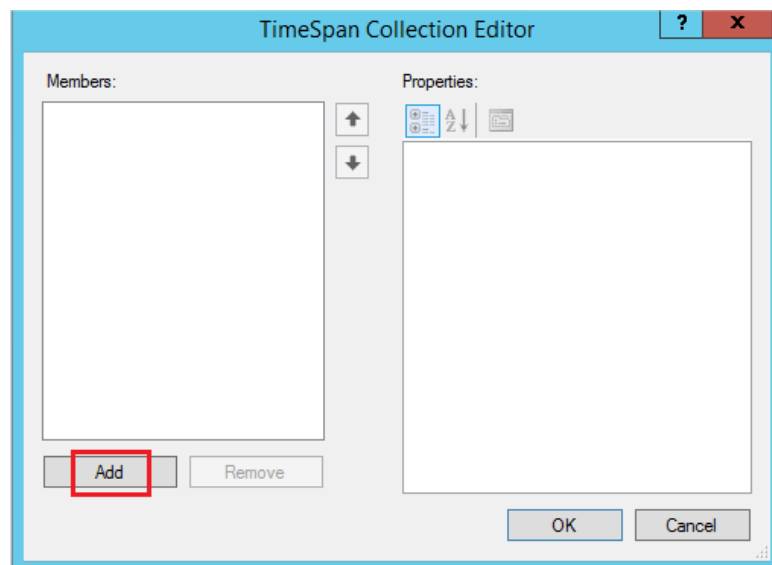
2. Select "HOPEX".
3. Click on the "Advanced Settings" option in the contextual menu:



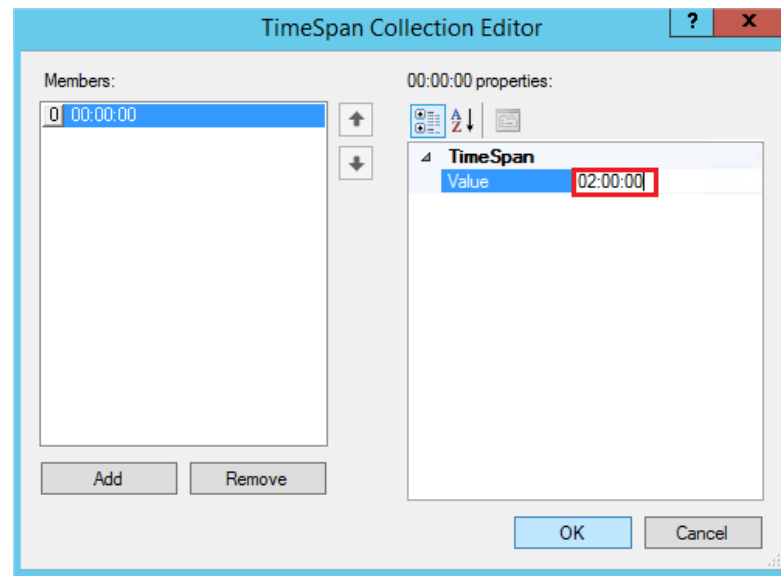
4. In the **Recycling** section, select the line Specific Times, and click on the "...” button:



5. Click on **Add**:



- Put the wanted restart time (in this example it's at 2am everyday), and click on "OK" to validate :

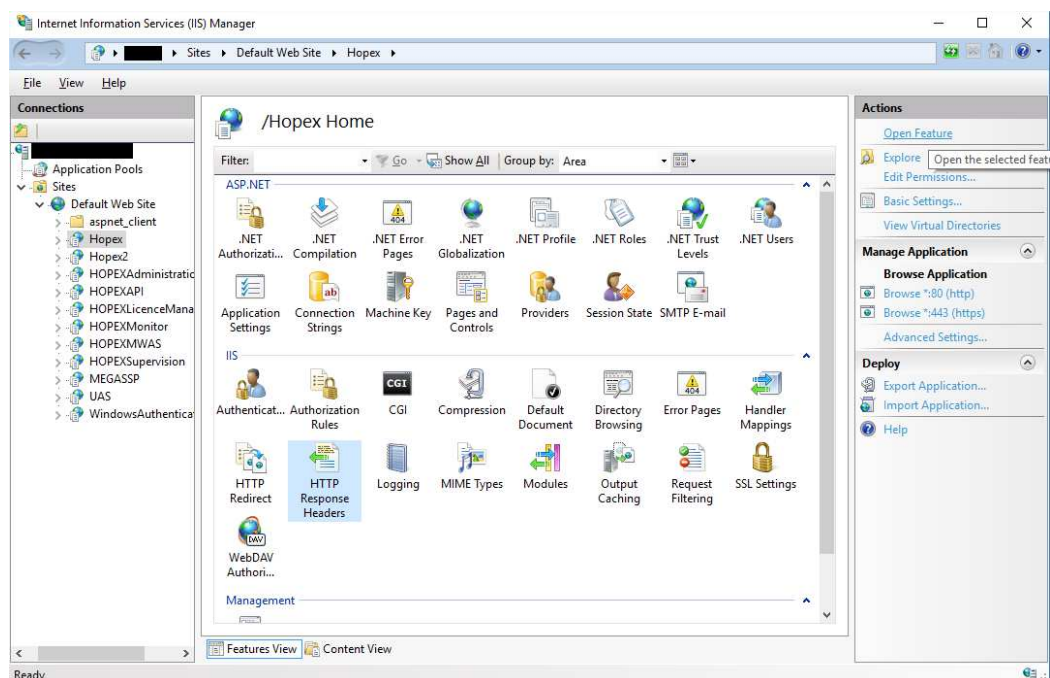


- Close the **Advanced Settings**. Reproduce this procedure for the **HOPEX2** application pool if you deployed HOPEX.

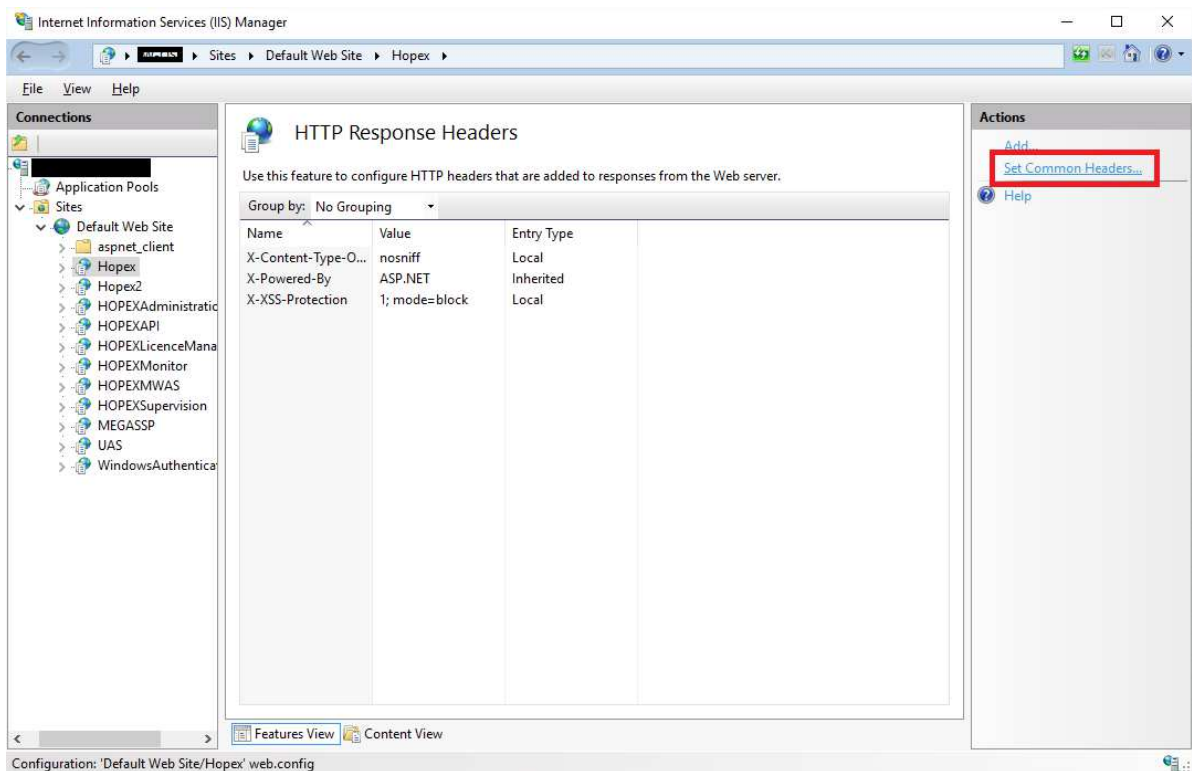
A restart of IIS will be needed to make this configuration active.

Configure Web Content expiration

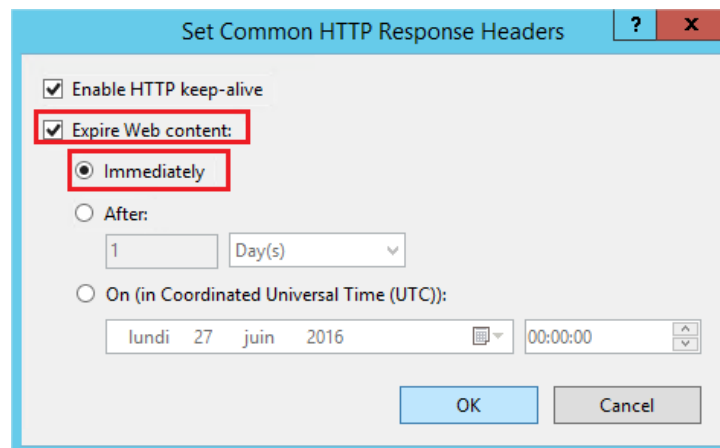
- In the "Internet Information Services (IIS) Manager", expand **Web Sites**, then **Default Web Site**.
- Select "HOPEX"
- Double-click the **HTTP Response Headers** functionality to open the feature:



4. Click **Set Common Headers** on the right panel:

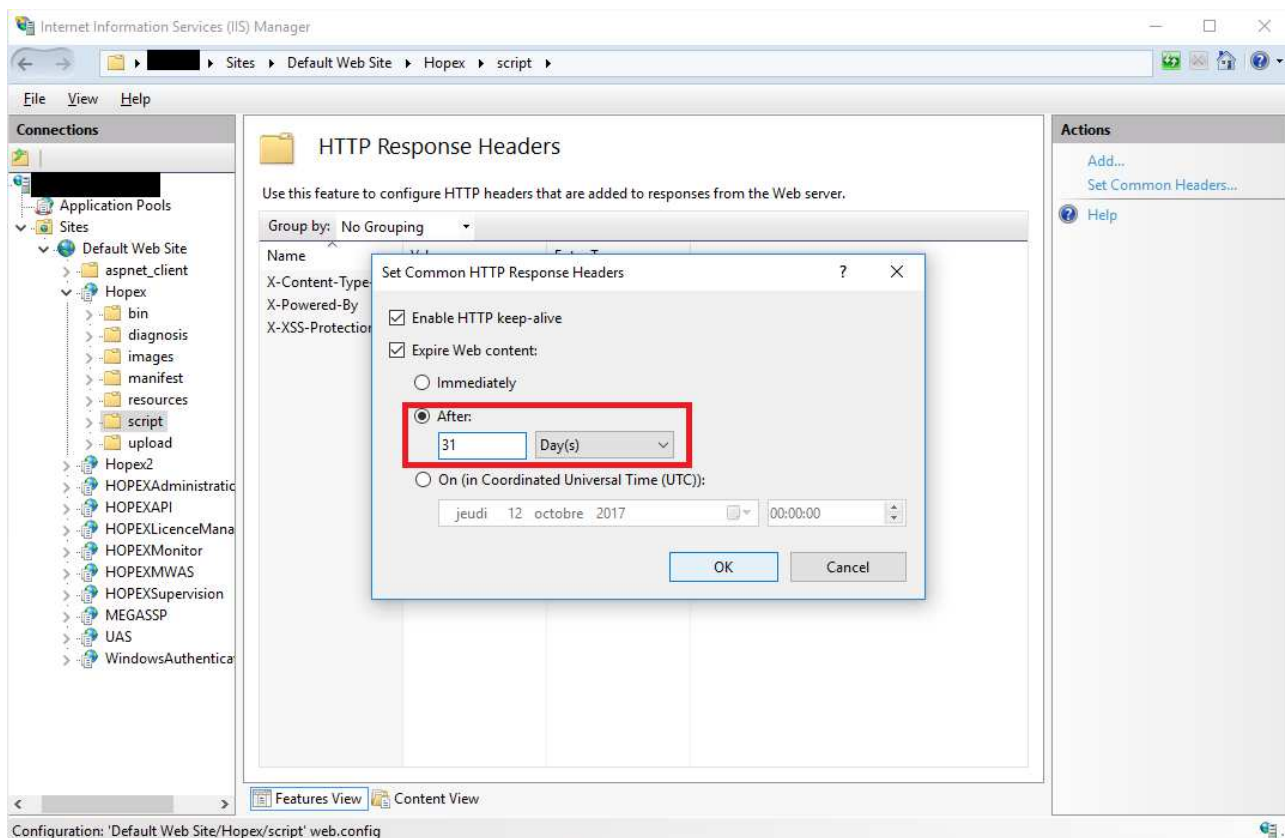


5. Set it to expire immediately :



6. In the "Internet Information Services (IIS) Manager", expand **HOPEX**
7. Select the **script** folder and choose "HTTP Response Headers" again.
8. Click **Set Common Headers** on the right panel.

9. Enable content expiration after 31 days.



10. Repeat this step on the **images** folder.

WHAT'S NEXT?

You have successfully installed MEGA HOPEX Web Front-end.

You should now personalize your setup.

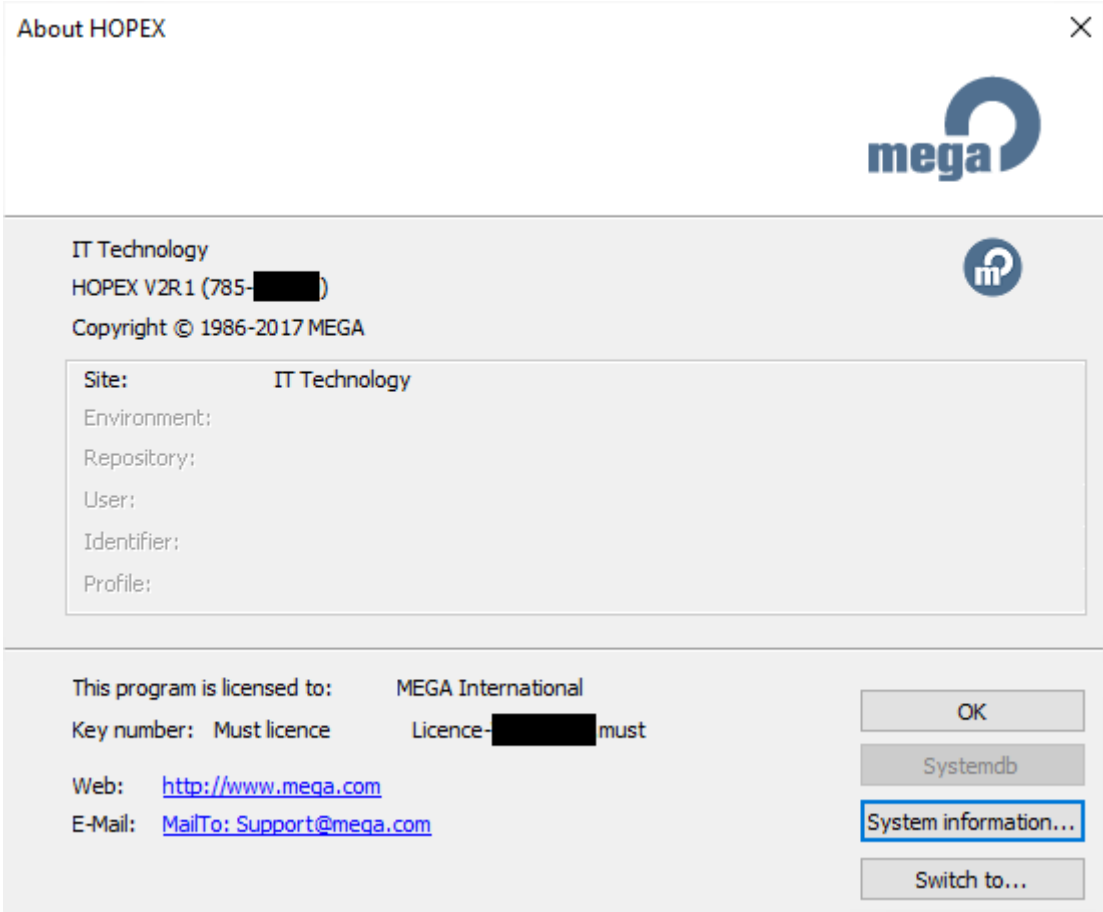
The two main steps are:

- Setting up an RDBMS environment
- Configuring authentication

Refer to Mega administration documentation.

TESTING THE INSTALLATION

A pre-requisite for the test is to install an environment.
On the server, run MEGA Administration Console (Administration.exe). Click the menu "Help" > "About MEGA" and check the license used. It must be the MUST license name generated for this installation.



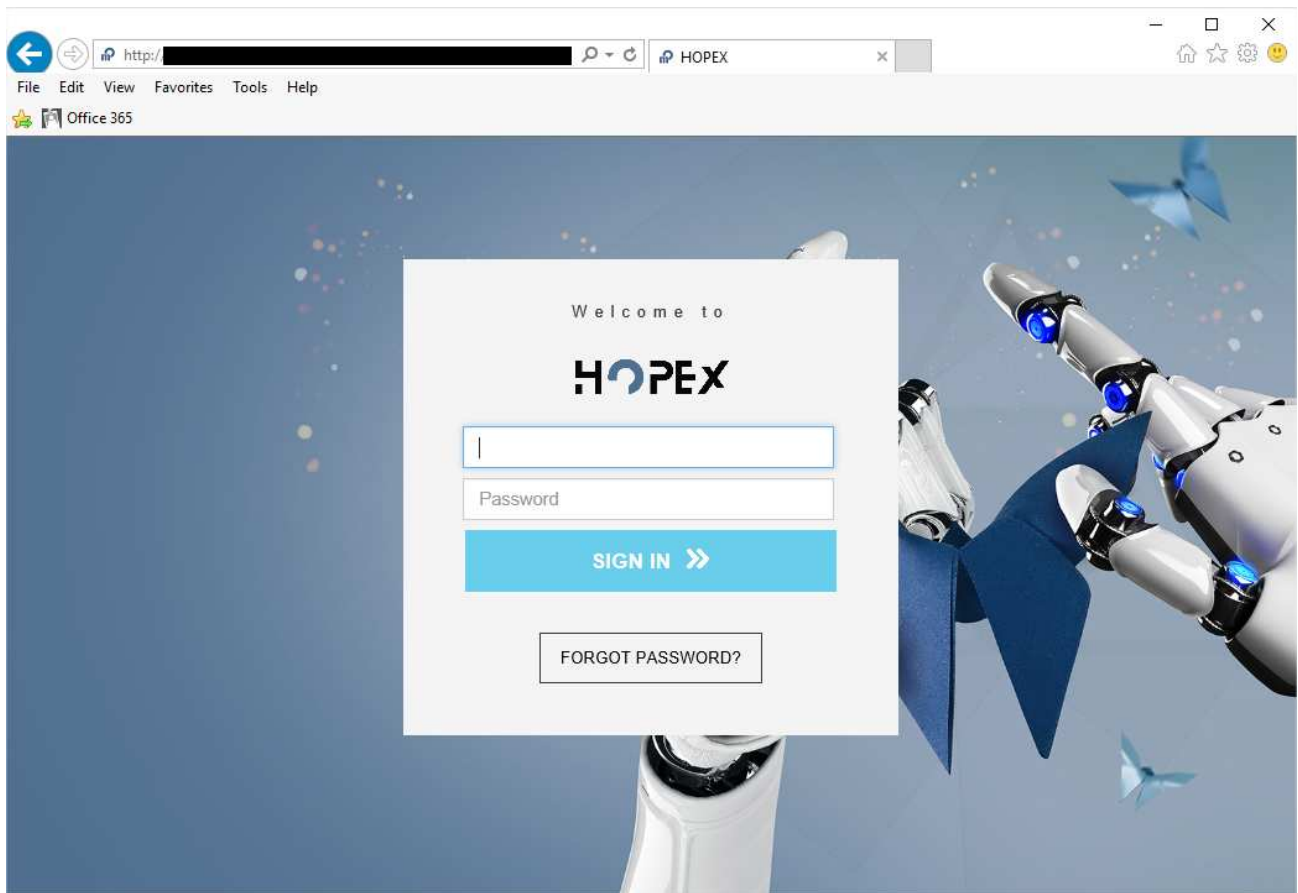
If the Administration Console does not start or if the license is not the one expected, you need to review the license configuration before going further.

Testing MEGA HOPEX (Web Front-End)

Prerequisite: a HOPEX environment is referenced on your platform.

On the server, open a supported browser and browse to <http://<servername>/hopex/>

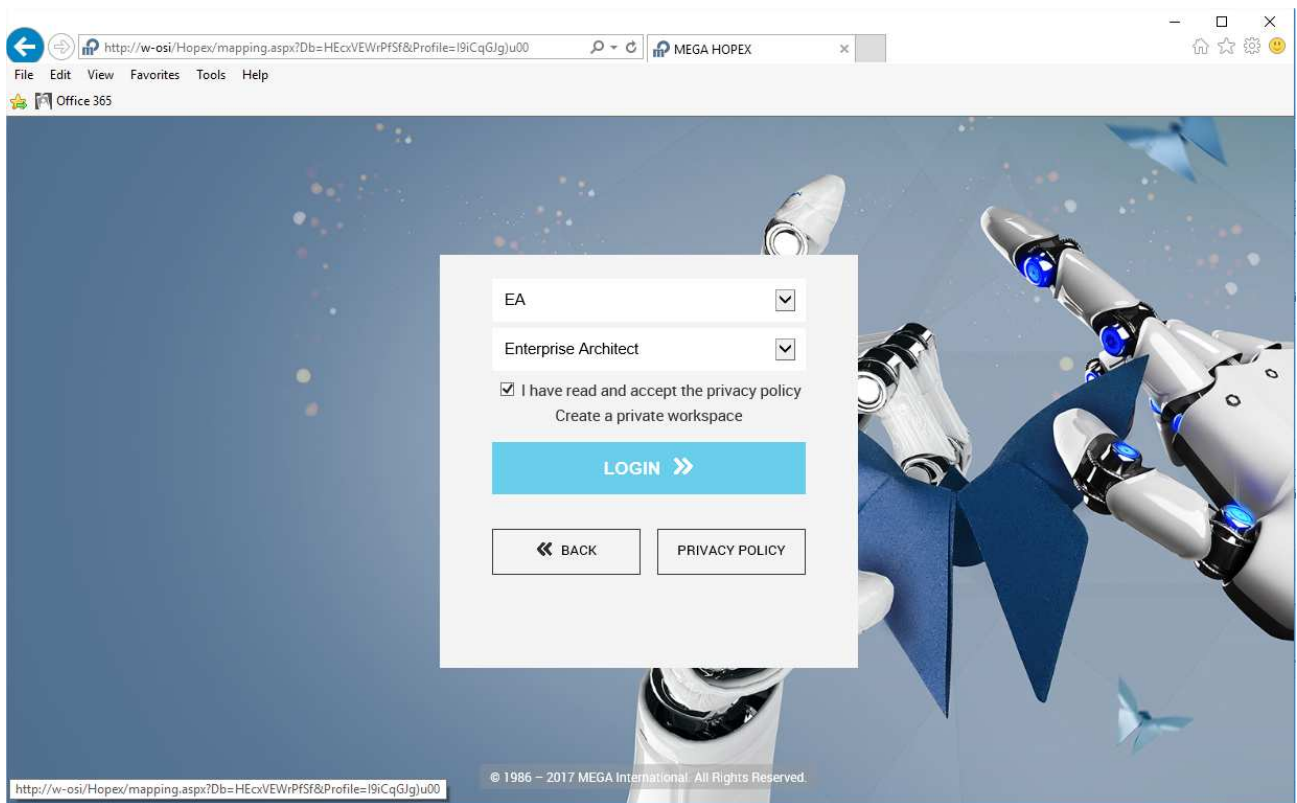
The login page should appear.



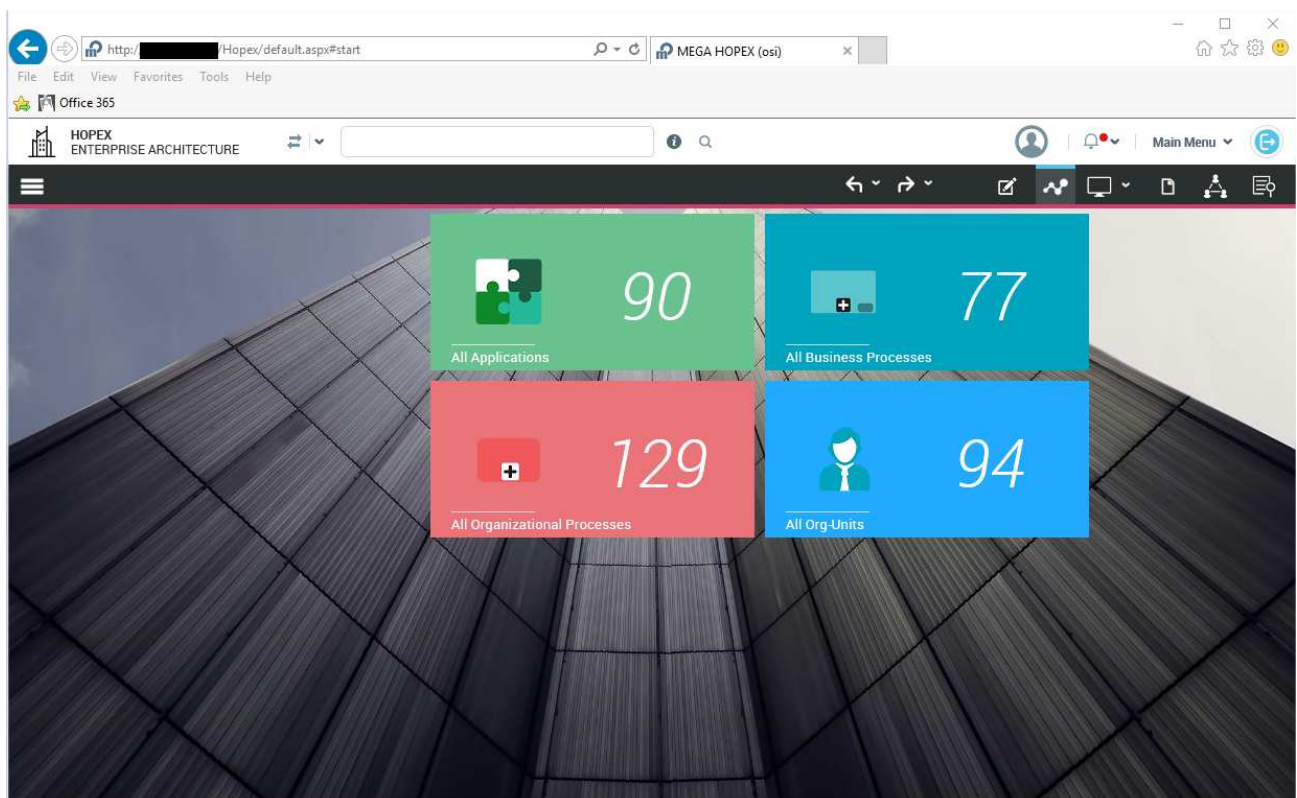
Now, use another client machine with any supported browser and browse to <http://<servername>/hopex/>. The login page should appear.

Log in to the environment with the Login "mega" and an empty password.

Then select:



The web workspace should be displayed:



Depending on available licenses, the displayed content may vary.

More required configuration

Word, Excel and PDF exports

Please, make sure your browser authorizes to download files.

IE may display the following message in the status bar preventing from opening the PDF file:

Pop-ups were blocked on this page. Press the "Ctrl" key when clicking to allow pop-ups.

Reports (MS Word)

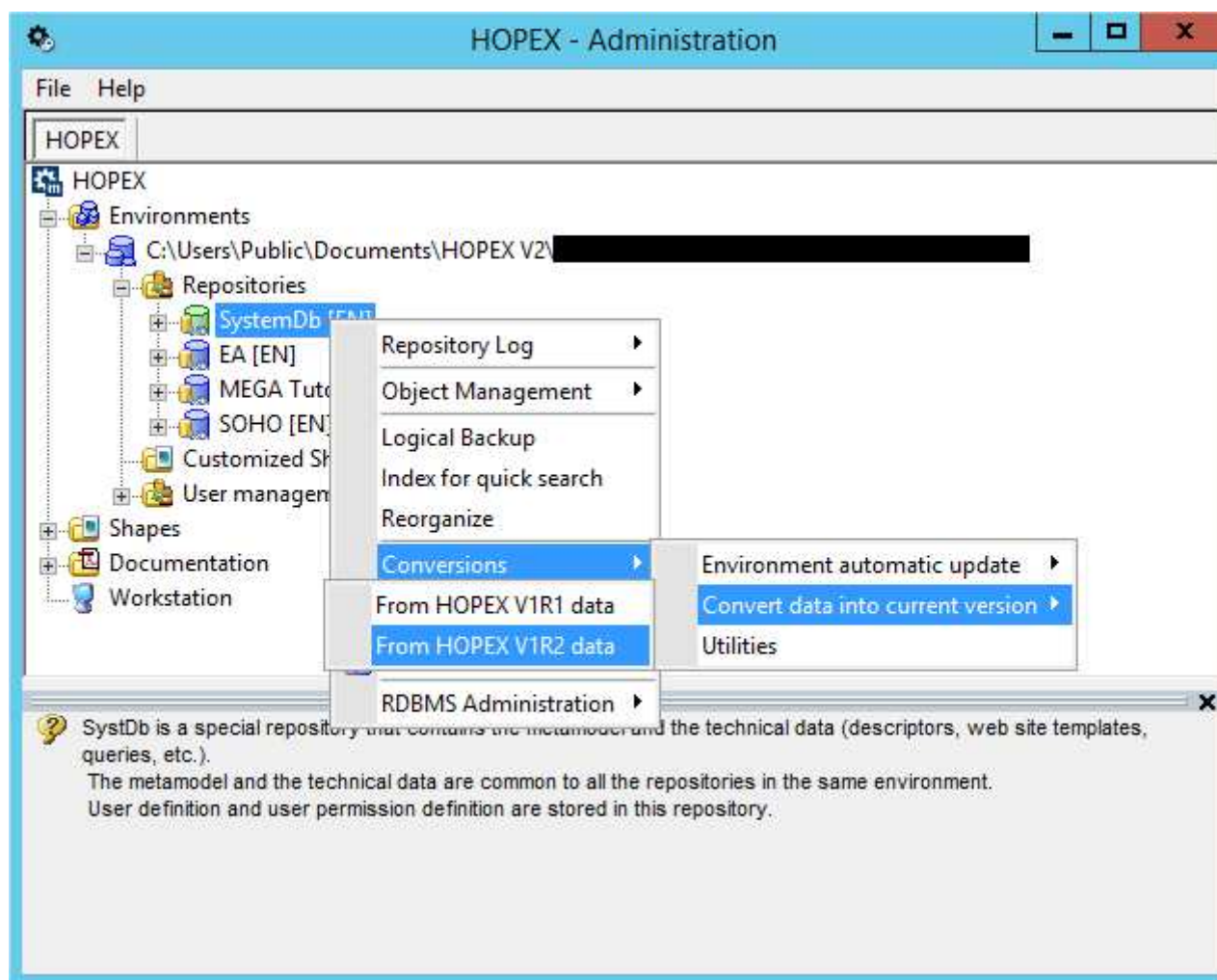
In Hopex V3, generated documents of new environments are automatically converted to the RTF format.

However, the format of Reports documents was MS Word.

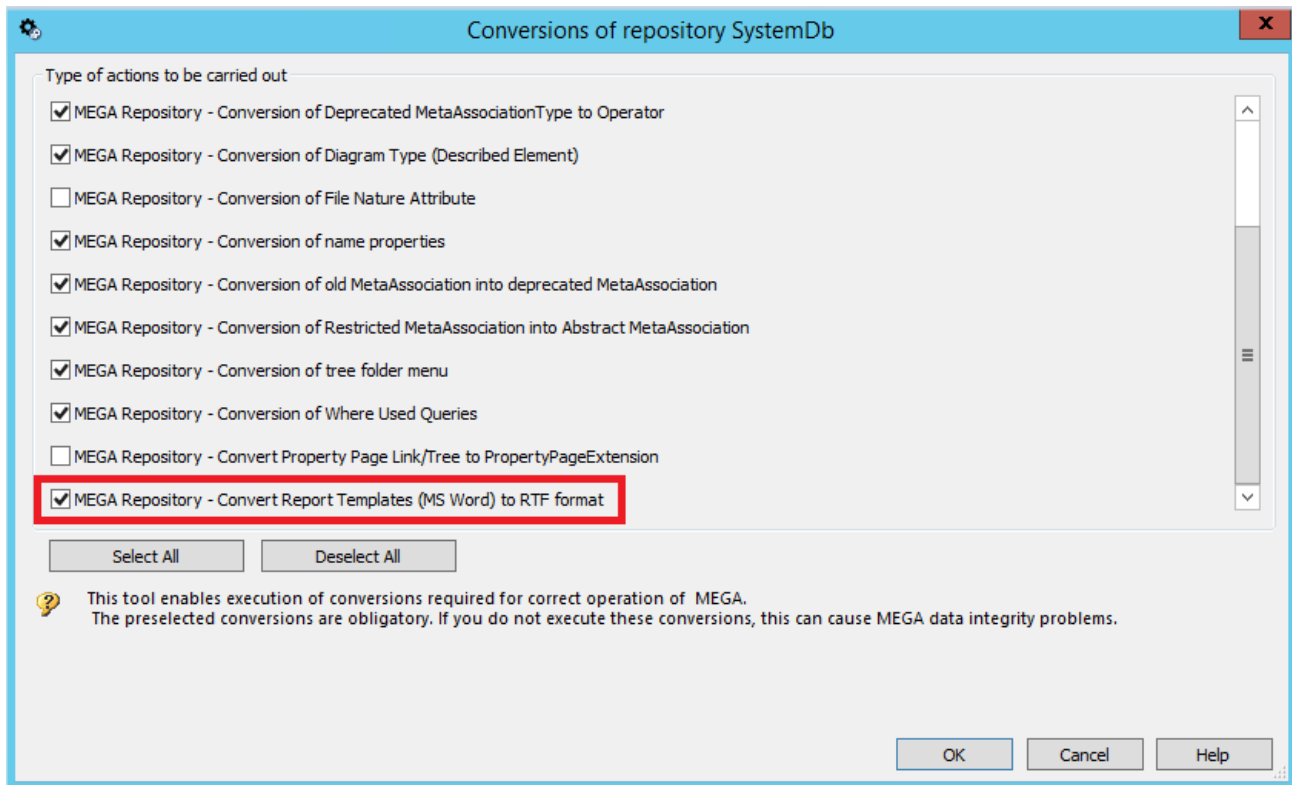
So if you are performing a migration from a source that still has this MS Word format, you must follow the below steps, in order to use documents on the Web Front-End.

You must exit any web session by dispatching or discarding your private workspace.

Then, from **a computer where Microsoft Word is installed**, go to the Administration Tool, open the environment you wish to convert with the user 'System', navigate to the systemDB and right-click "Conversions > Convert data into current version > From HOPEX V1R~~x~~ data", 'x' obviously being the version of your environment in the process of being migrated:

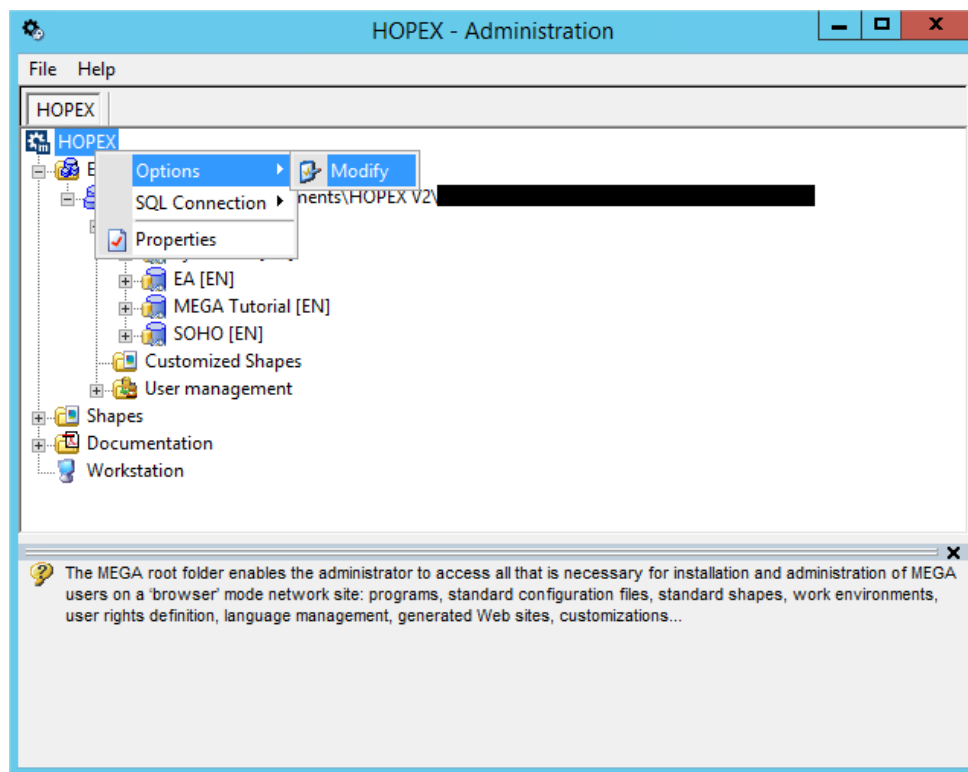


This is the last possible conversion called "MEGA Repository - Convert Report Templates (MS Word) to RTF Format", that is checked by default:



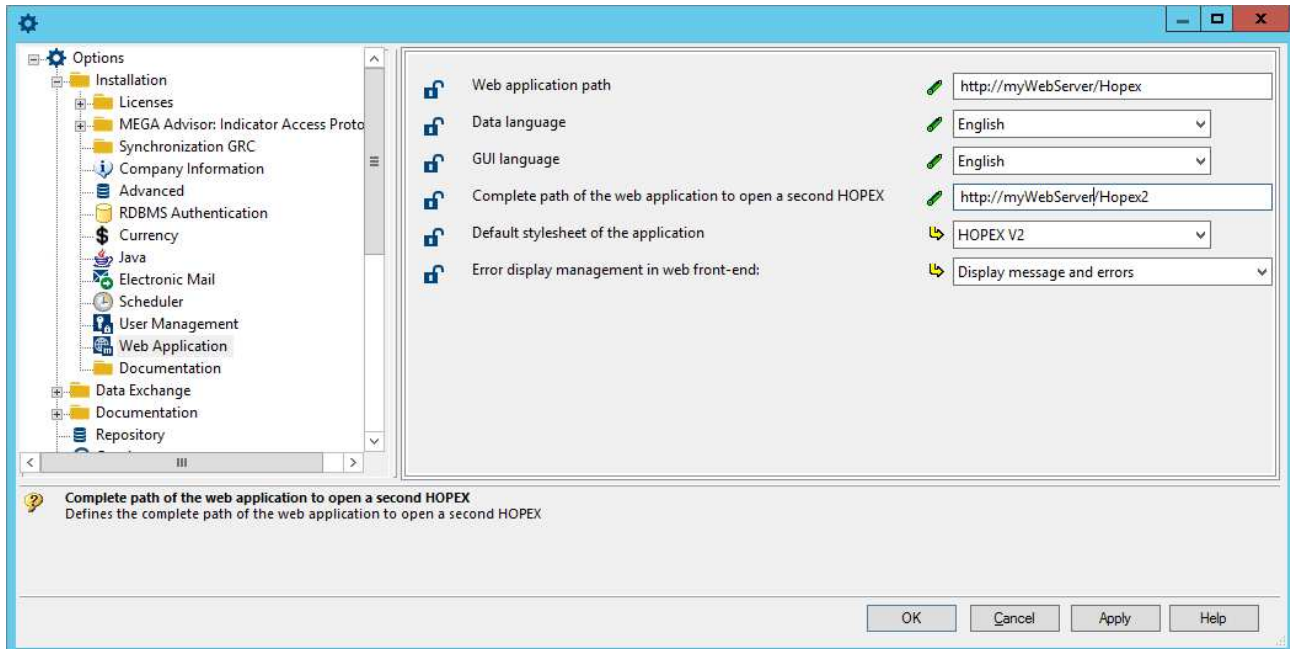
Required options configuration

You must fill-in a number of site options using Administration.exe, at the root level:

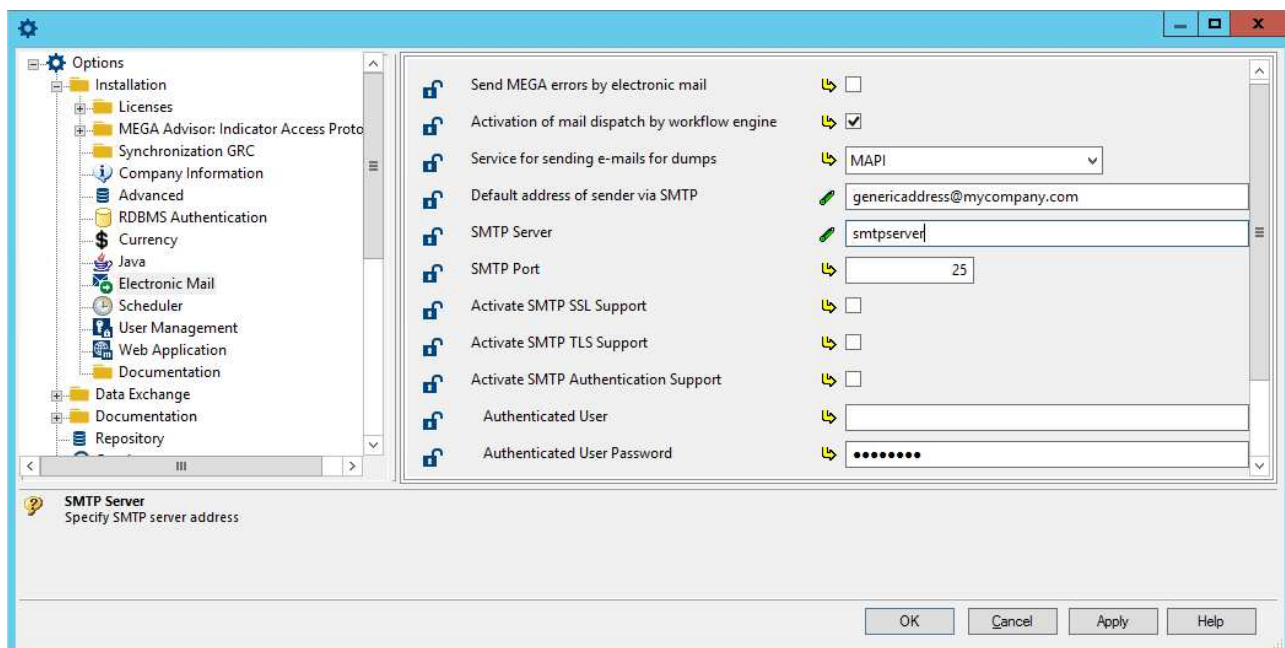


The following options are required for MEGA HOPEX Web front-end to operate:

- In the Installation > Web Application folder:
 - “Web Application path” (e.g. <http://myWebServer/HOPEX/>) and “Complete path of the web application to open a second HOPEX”, are already filled-in by the setup. However, you may have to change it if a DNS alias is put in place after the setup, or if your web servers are behind a Load Balancer or a reverse-proxy that has a different address than the server name itself:

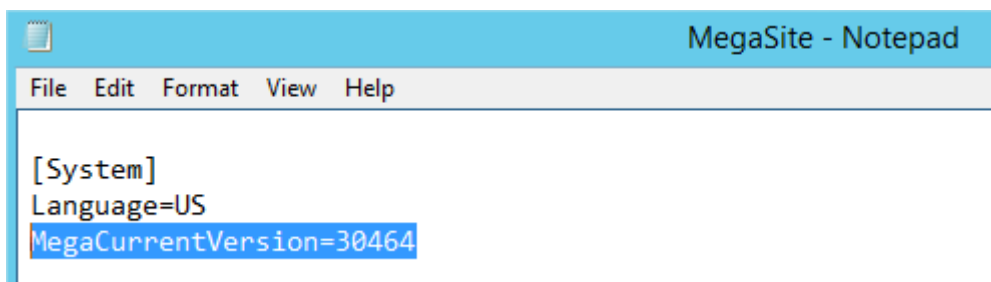


- In the Installation > Electronic mail folder:
 - Default address of sender via SMTP
 - SMTP Server
 - Any SMTP configuration (port, SSL, authentication, etc.) required by your infrastructure



In V3, please note that your MegaSite.ini will contain the version number. So if you upgraded a V1R2 version to a V2R1 version, and wanted to keep your settings, make sure that you have those information in the [System] section of your MegaSite file :

MegaCurrentVersion=30464



Allowing the use of verbose logs and activation

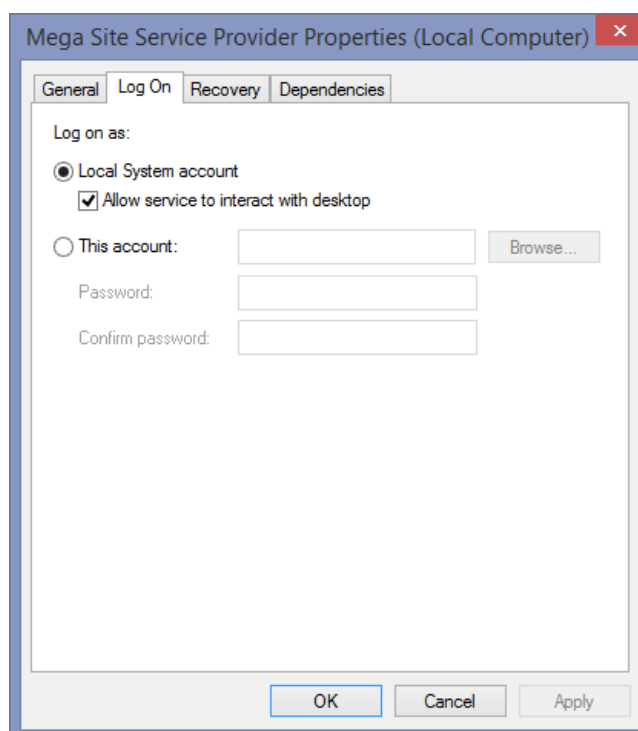
To allow Mega to perform deep analysis of the logs, it can be asked to activate the verbose mode. The verbose mode activation includes two steps:

- Registry update
- Mega Server Supervisor: « verbose mode » activation
 - ➔ To deactivate this configuration, see Disabling the verbose mode p. 50.

Registry update

To update the registry before activating the verbose mode:

1. Allow the impersonate account (see “Windows User for MEGA HOPEX” section) to have read/write access to a key (see Windows User(s) for MEGA HOPEX p. 11).
2. Make sure that the account that runs the “Mega Site Service Provider” Windows service has the same access level.
 - a. Check which account is running the service:



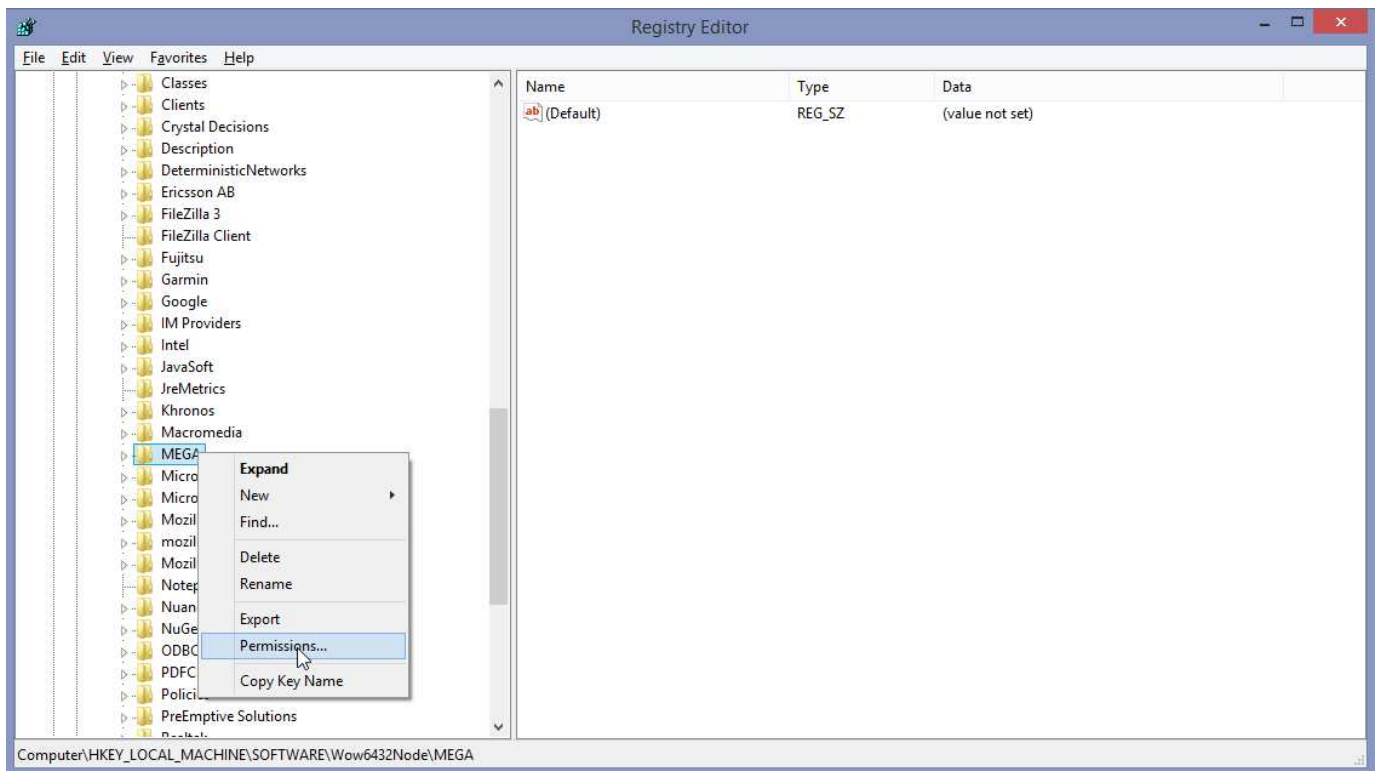
If it is « Local System », you do not need to update the registry for that service, only for the impersonate user.

Otherwise, if you run the service with a local account or domain account different than the impersonate account, we advise you to add this account in the IIS_IUSRS local group.

- b. Launch the “regedit.exe” executable to open the registry.
- c. The registry key on which you need to change the permissions is:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\MEGA

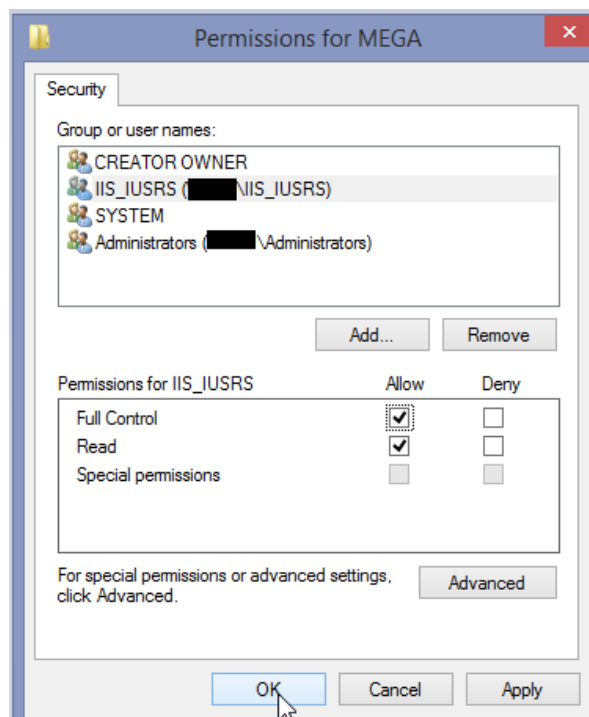
*Right-click the key and select **Permissions**.*



- d. On this key, add the "IIS_IUSRS" local group.

The impersonate account is normally included in the "IIS_IUSRS" local group, and as stated before, that can contain the account that runs the "Mega Site Service Provider".

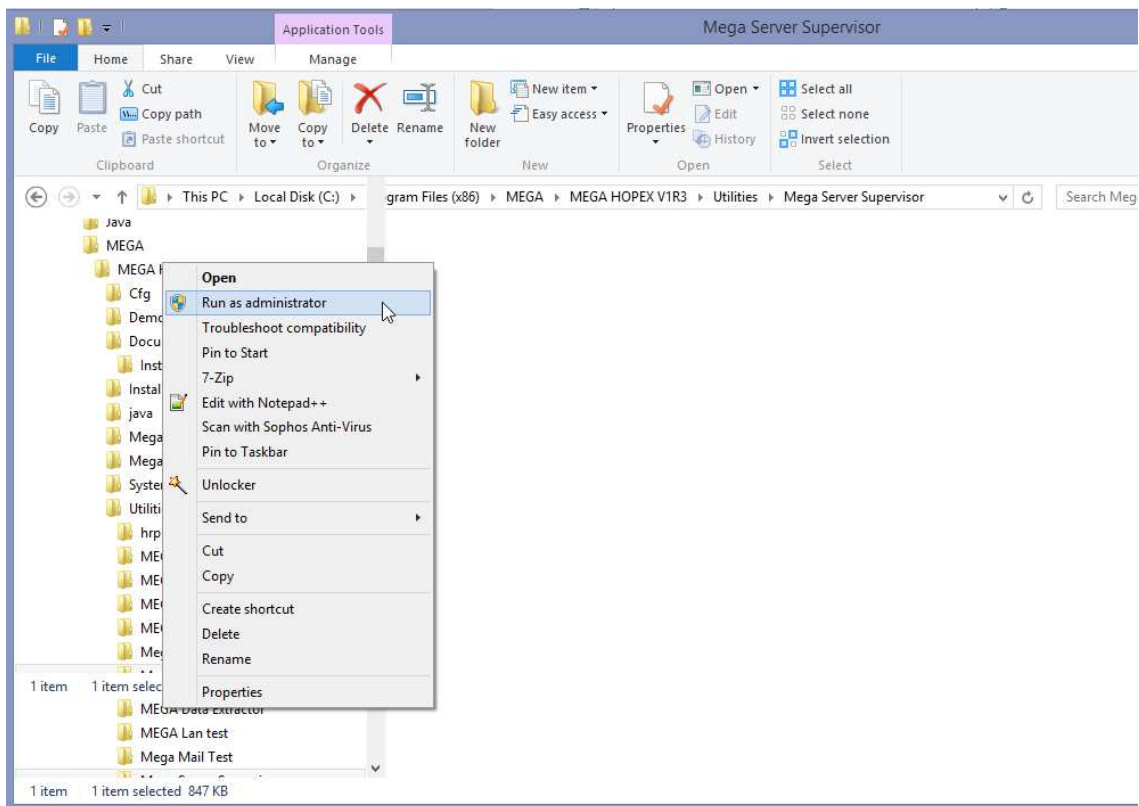
- e. Allow: "Full Control" and "Read".



- f. Apply and close the registry.

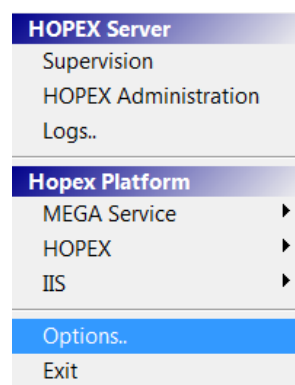
Mega Server Supervisor: « verbose mode » activation

1. In the MEGA installation folder, expand the **Utilities > Mega Server Supervisor** folder of Mega binaries.
2. Right-click « MEGA Server Supervisor.exe » tool and select **Run as administrator**.

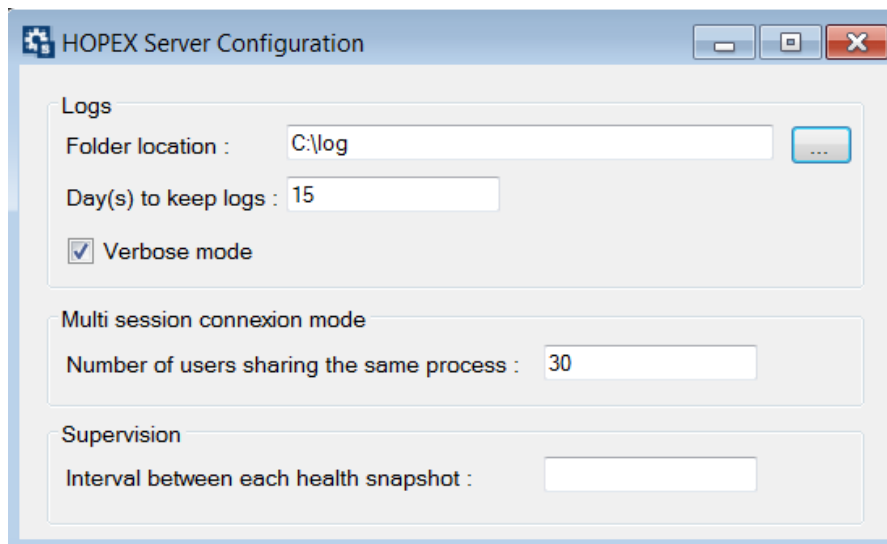



The **MEGA Server Supervisor** icon  appears in the system tray of your workstation.

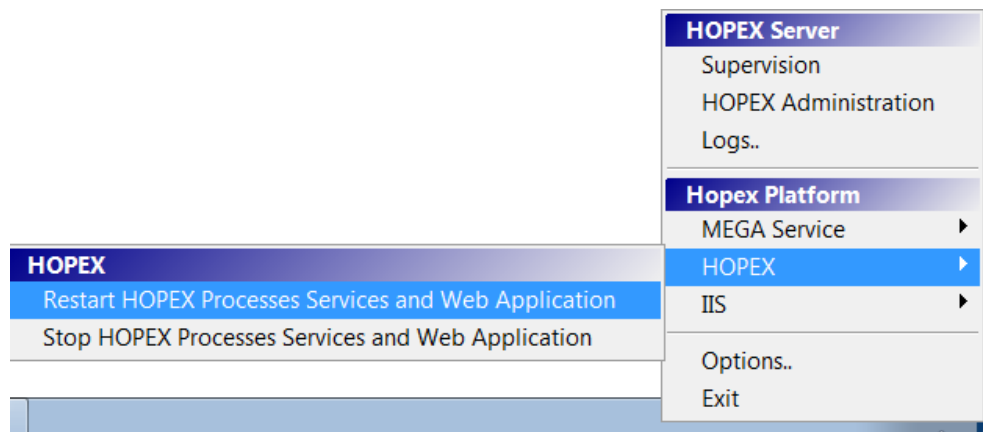
3. Right-click the icon and select **Options**.



4. Select **Verbose Mode**.



5. Close the configuration window to apply.
6. If your application was started, you need to restart it. You can use the Mega Server Supervisor to do so: right-click the **MEGA Server Supervisor** icon  and select **HOPEX > Restart HOPEX Processes Services and Web Application**.



Disabling the verbose mode

To deactivate the verbose mode:

1. Follow the Mega Server Supervisor: « verbose mode » activation procedure p. 49 and clear **Verbose mode**.
2. Restart the application to take the modification into account.


URL Rewrite

This is a component that was added with the Update 2 of Hopex V2R1, and that is needed for later releases too. A component that is added to the usual IIS features.

If you make a new install with the Update 2 master, you shouldn't concern yourself.

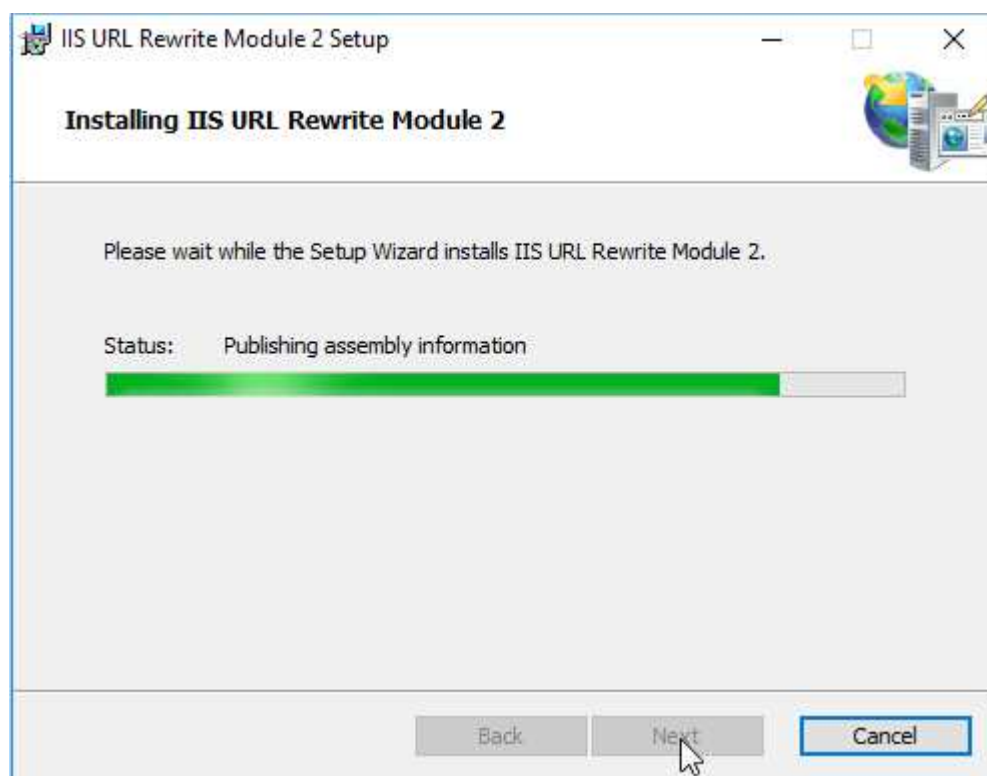
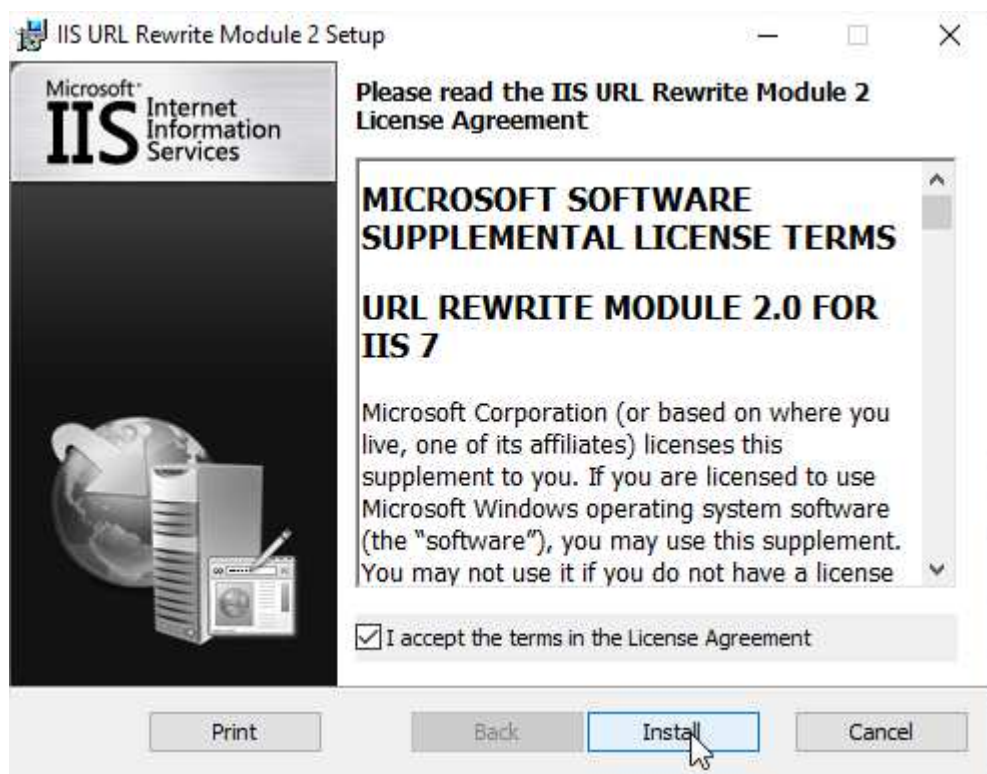
However, if you had a previous install of Hopex V2R1, and you used this document to make you installation, instead of the very specific release note linked to patching to Update 2, then you need to install that feature.

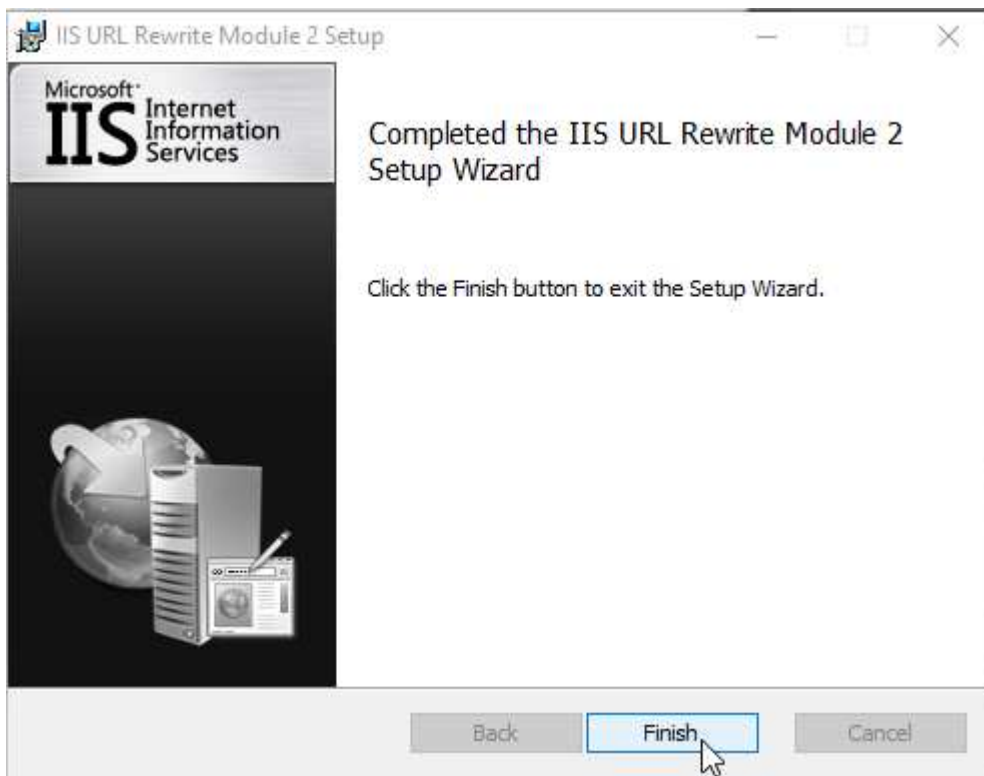
You can find the installer link at this address:

Web Front-End Installation Guide HOPEX V3 EN	page 50/62	C0 - PUBLIC	
--	------------	-------------	---

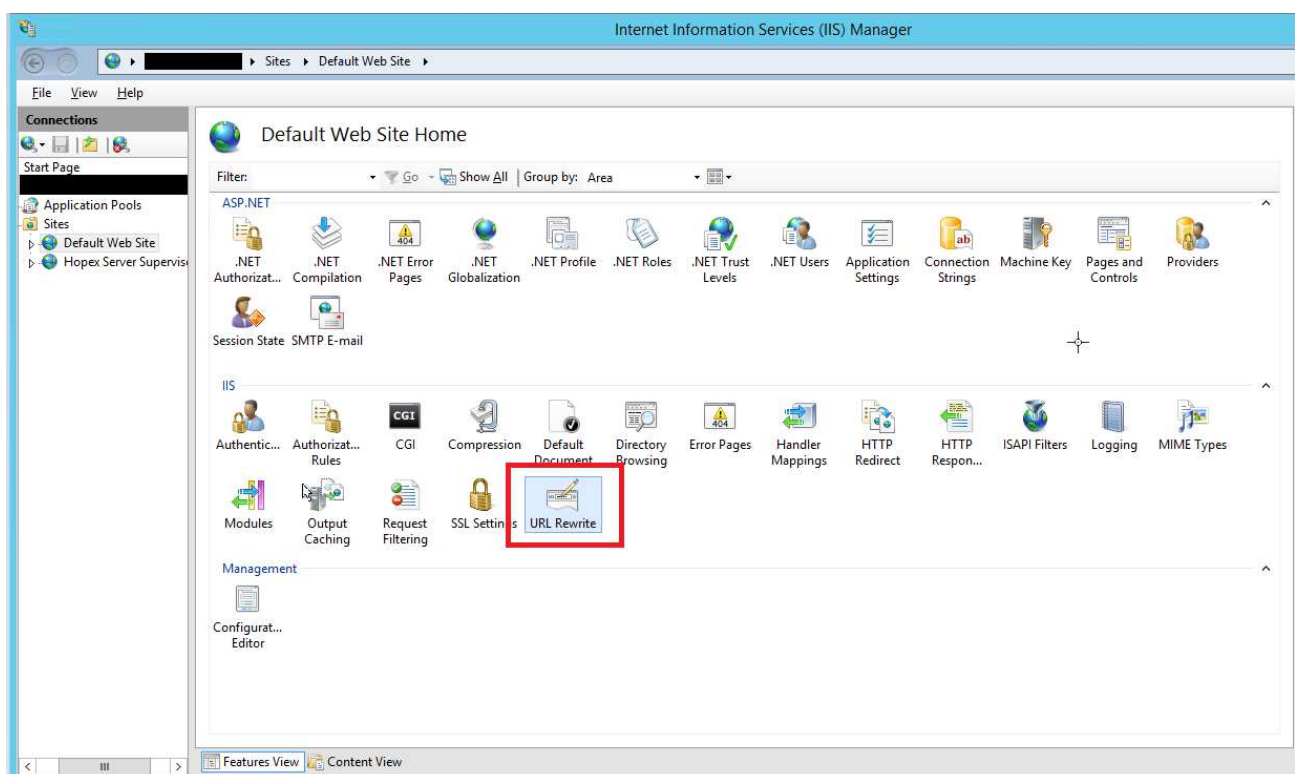
<https://www.microsoft.com/en-us/download/details.aspx?id=47337>

Click on "Install" and let the component get deployed:





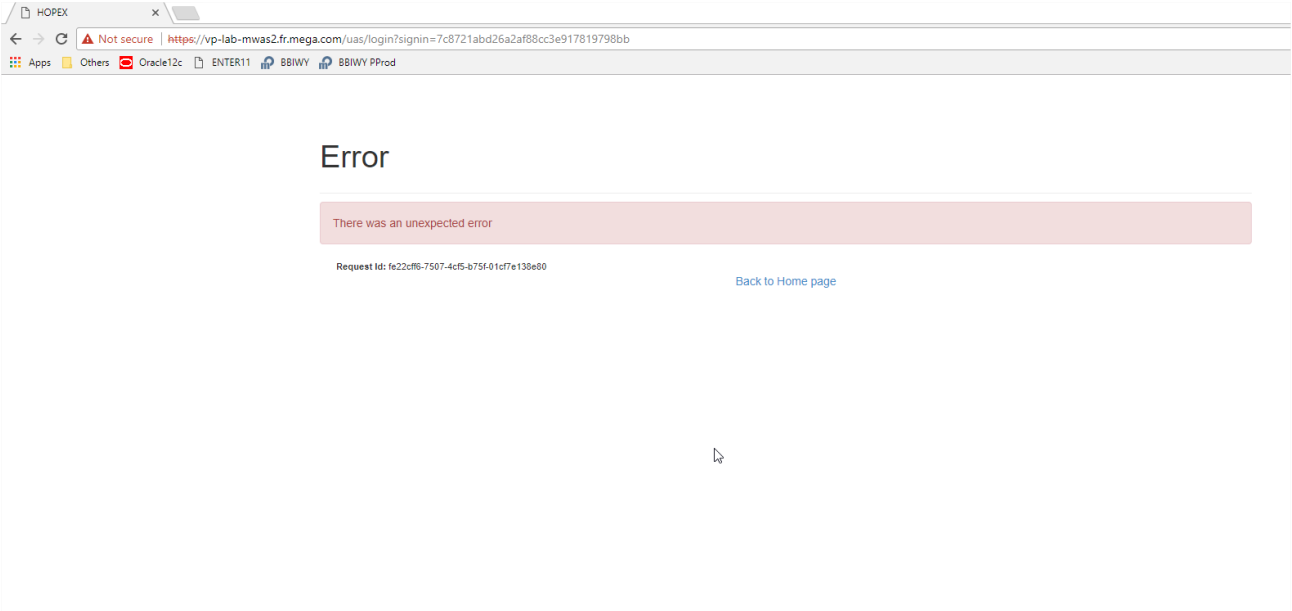
A **reboot** is required to be able to see the feature in IIS:



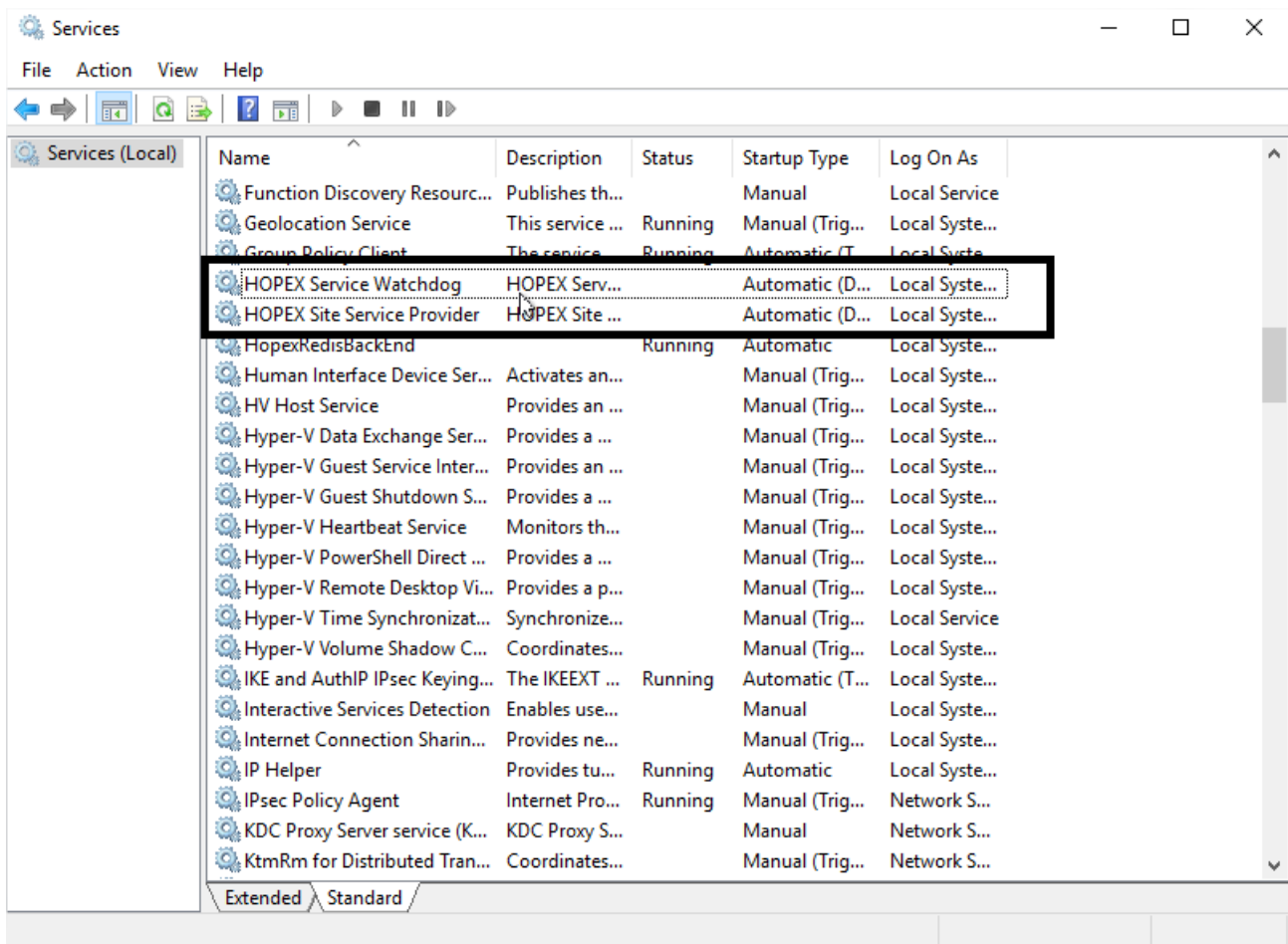
TROUBLESHOOTING

Check that the Site Service Provider is running

If you have the following message, and environments are accessible using Administration.Exe and Mega.exe:



You should first check that the “HOPEX Site Service Provider”, and the “HOPEX Service Watchdog”, are running using the Services administration tool:



If those are not running, check that its startup is « Automatic » and Start the services.

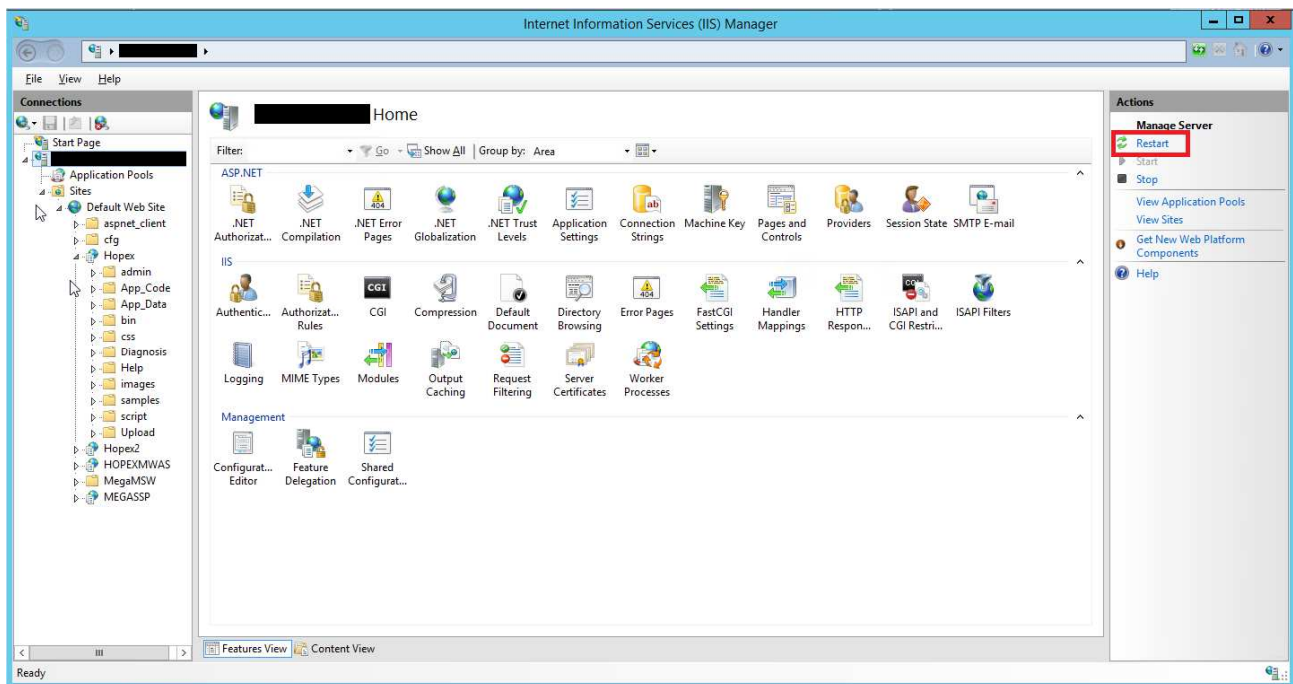
If it is running, restart the service.

Then restart IIS (see below).

Restarting Internet Information Services

If errors occur, the first step is to try to restart the Web Server.

In the "Internet Information Services (IIS) Manager", select the server name and click "Restart" in the Actions panel



Referencing a New Environment

So that a new environment is fully accessible in MEGA HOPEX, do not forget to give the "Windows user for MEGA HOPEX" full access rights to the environment folder.

If you have a multi-server deployment, you should reference it and check the rights on every server (except for pure web servers).

Disabling Data Execution Prevention

In rare cases, it might be necessary to disable "Data Execution Prevention (DEP)" for MEGA programs.

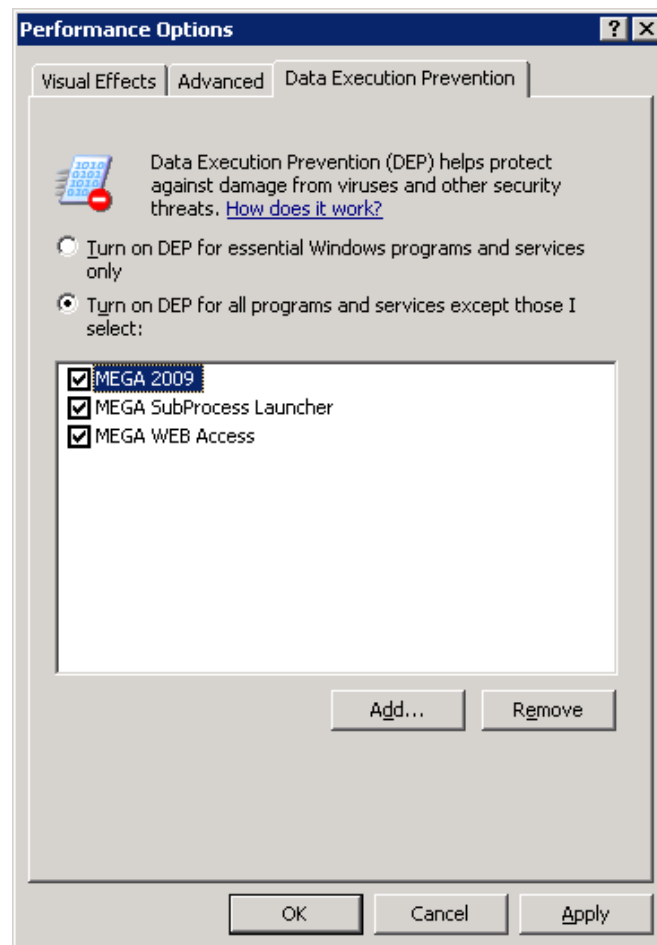
If you are able to run the Administration.exe and/or the Hopex.exe rich clients from the web server, then the following procedure is unnecessary.

To access the DEP settings:

In the "Start" menu, right-click "Computer" and select "Properties". In the next screen, click "Advanced System Settings". Go to the "Advanced" tab, click "Settings" in the "Performance" group and select the "Data Execution Prevention" tab.

You can either turn on DEP only for essential Windows programs and services, or add exceptions for the following Mega programs (default installation locations):

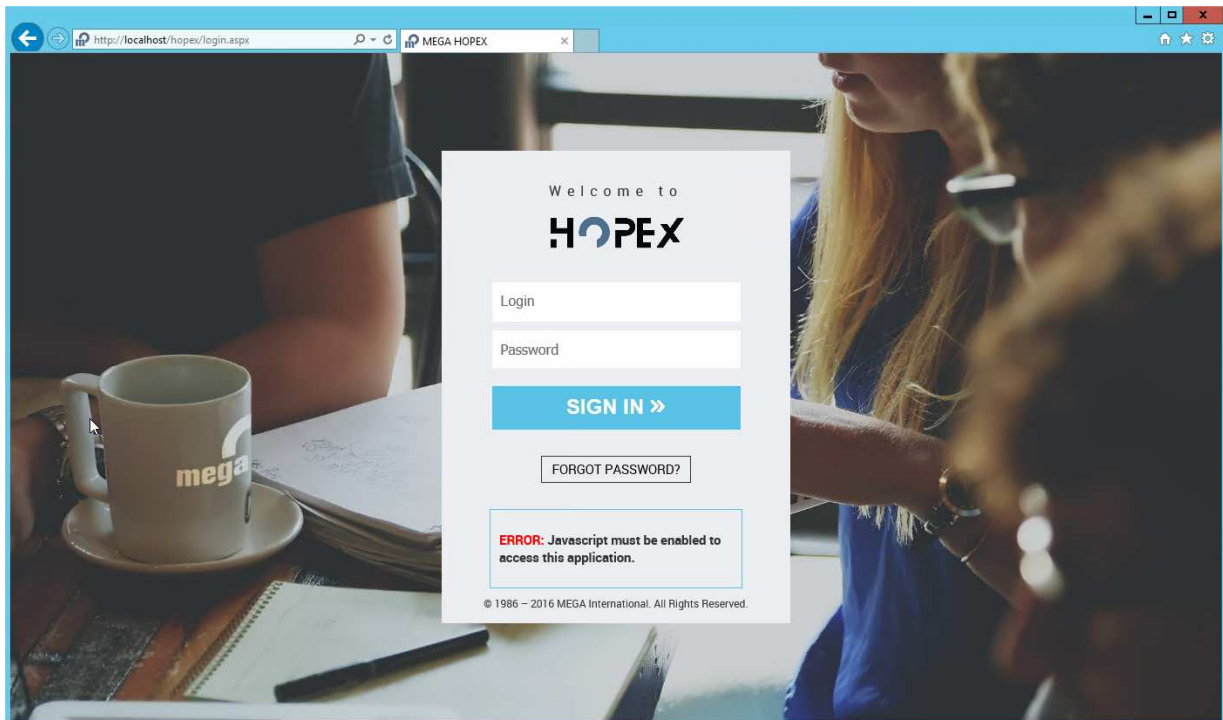
- C:\Program Files (x86)\MEGA\HOPEX V2R1\System\mgwspro.exe
- C:\Program Files (x86)\MEGA\HOPEX V2R1\System\mgwmapp.exe
- C:\Program Files (x86)\MEGA\HOPEX V2R1\System\mgwmwas.exe



Loosening Internet Explorer Security Settings

Although default browser security settings on client machines are sufficient for using the MEGA Web Application, some computers, especially servers, might have stricter security policies. For

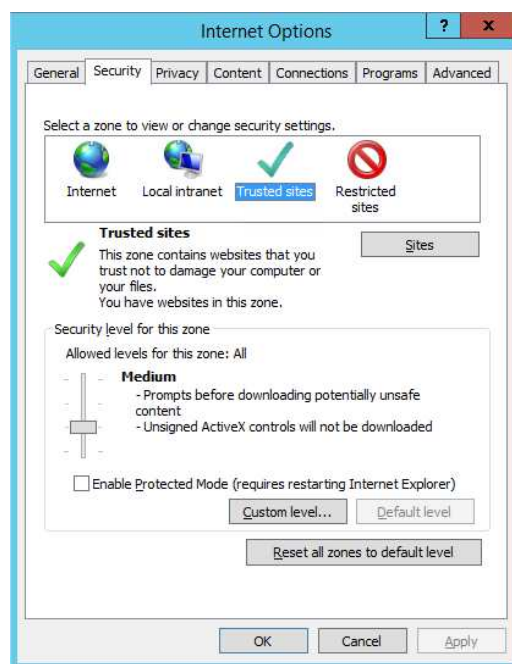
instance, they might prevent the execution of JavaScript, on which the MEGA Web Application relies.



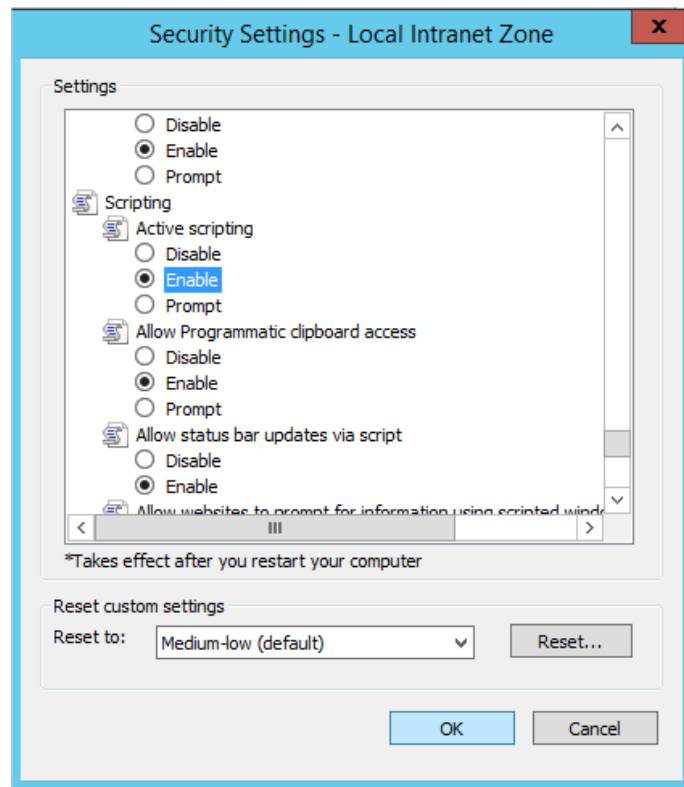
To fix this issue, follow the steps below:

Step 1: Enable Active Scripting on the trusted sites zone

1. Go to the "Tools\Internet Options..." menu
2. Select the "Security" tab
3. Click on "Trusted Sites" then "Custom Level"

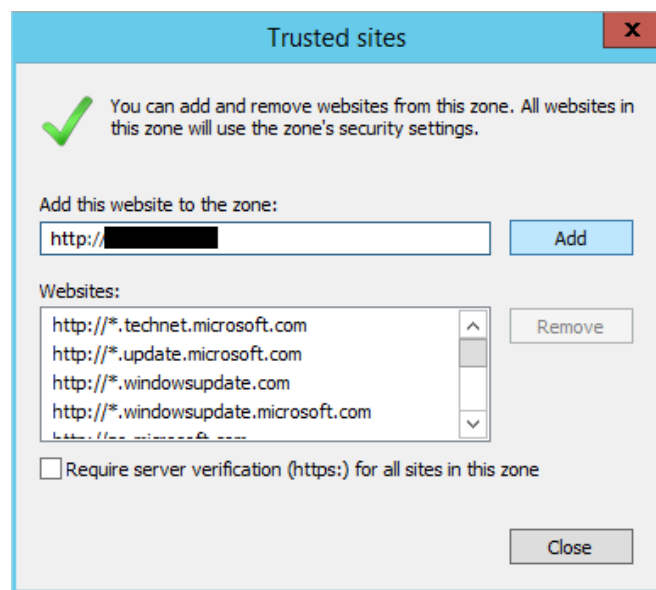


4. Enable "Active Scripting"



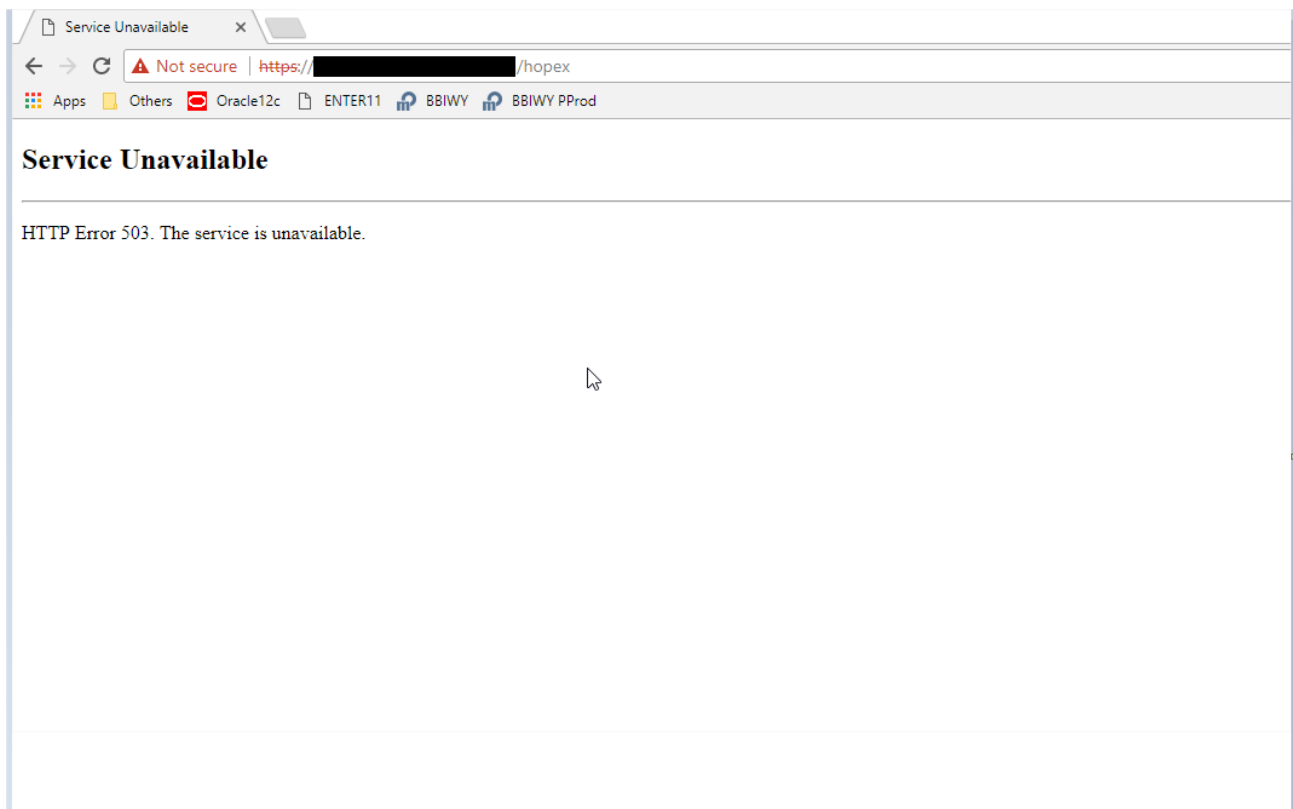
Step 2: Add MEGA Web Application to the trusted Web sites list

1. Go to the "Tools\Internet Options..." menu
2. Select the "Security" tab
3. Click "Trusted sites" then "Sites"
4. Enter the Address of your Web site, click "Add"
5. Validate



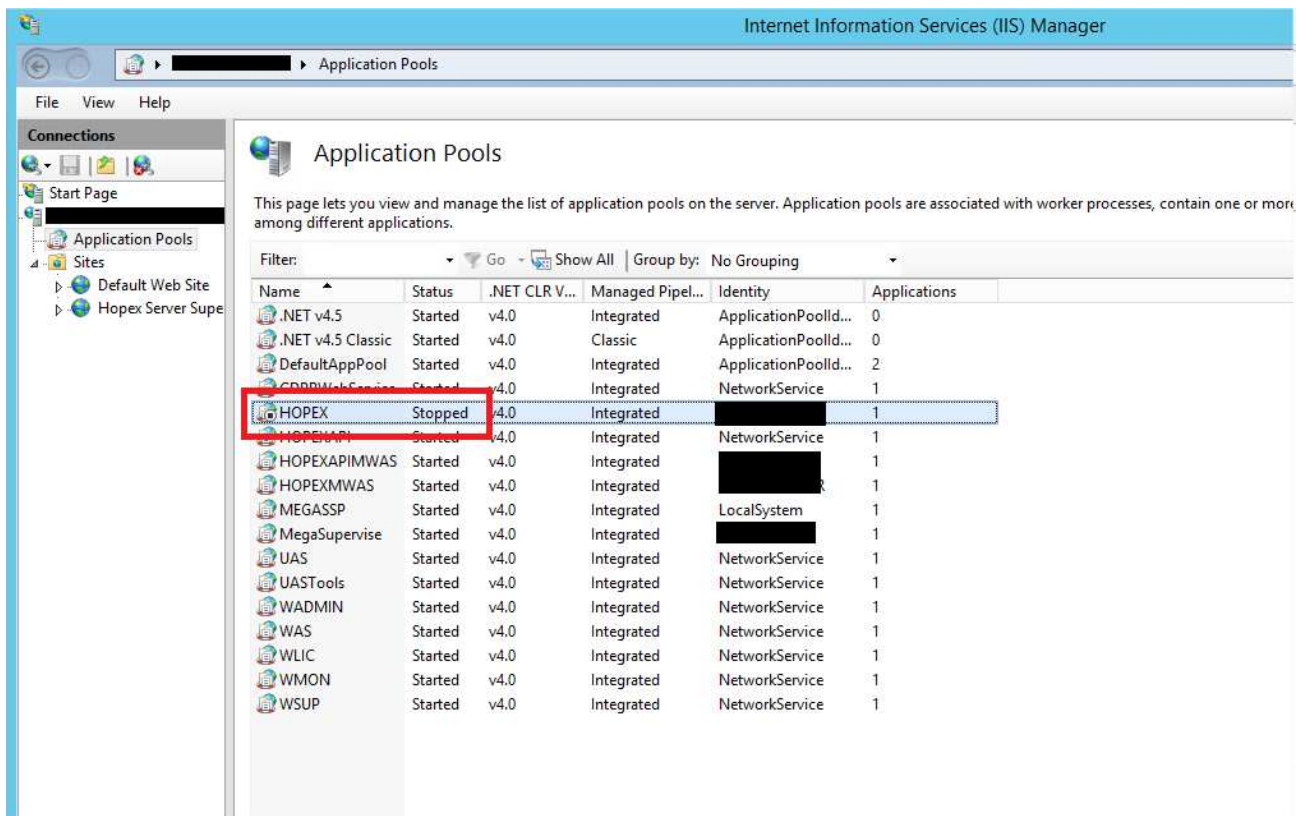
Manage http 503 error

If, when trying to access the /hopex of your web application, you end up on this kind of error page:



If this is the case, you may want to check IIS Manager, and more precisely the status of the application pool called "HOPEX".

As you can see, it is in "Stopped" status:

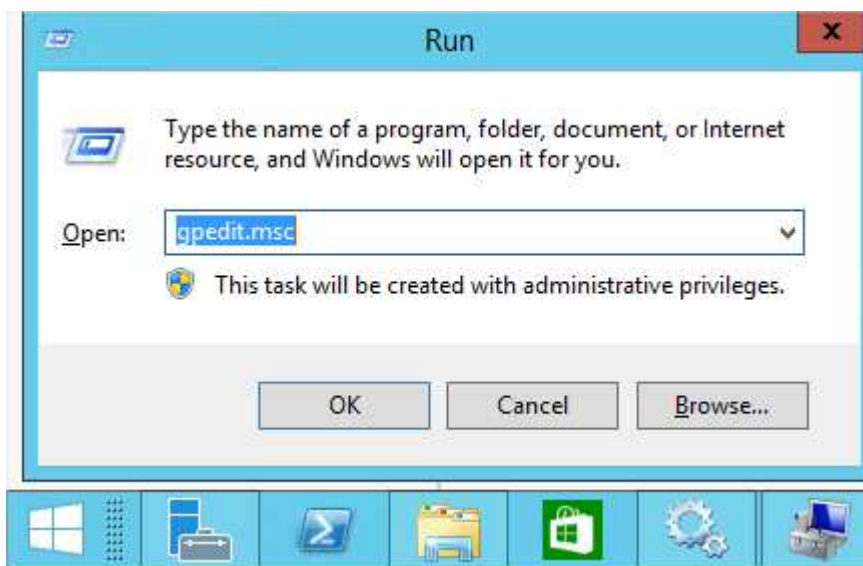


You can try to start it again, and refresh your browser. If you continue to have this error message, you can go back to IIS Manager, refresh the view of the application pools, and see if it is stopped again.

If it is still the case, you most likely have a policy issue on that account.

To check the local policies, execute the following command from the "Run" menu of your application server:

gpedit.msc



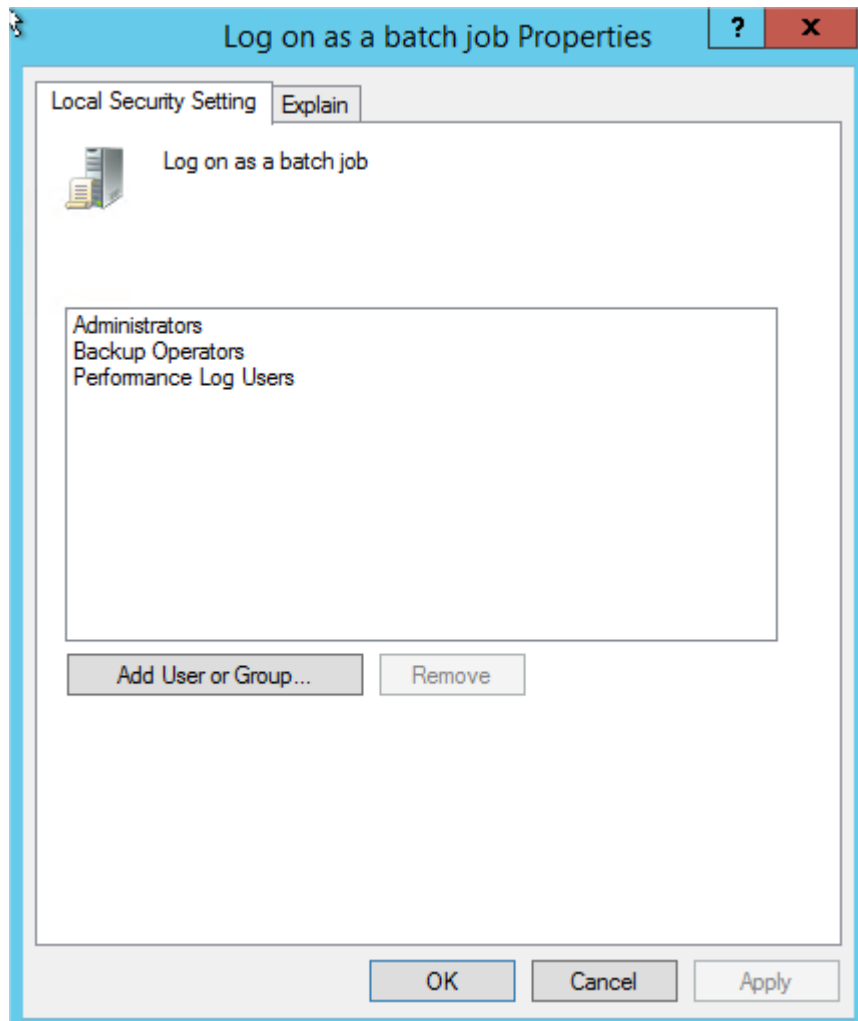
In that tool, browse to "Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> User Rights Management", and locate the rules :

- Log on as a batch job

- Impersonate a user after authentication

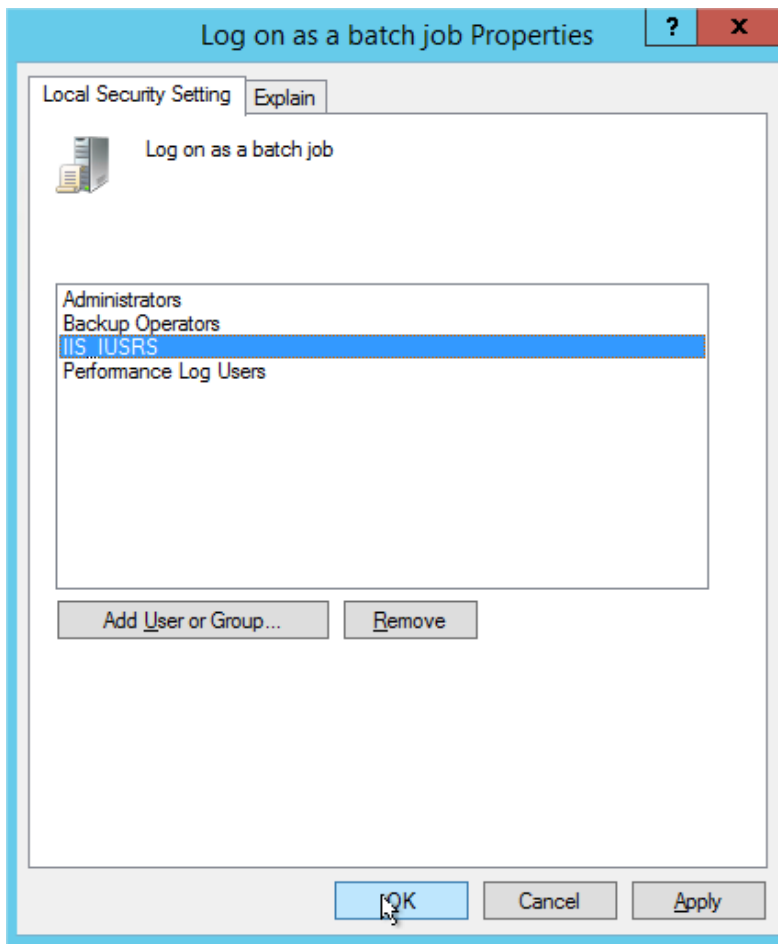
If you open the Properties of those rules, you need to check if the groups containing the Windows user you set up during the installation are part of the list of granted resources.

Here, we see that the IIS_IUSRS local group is missing:

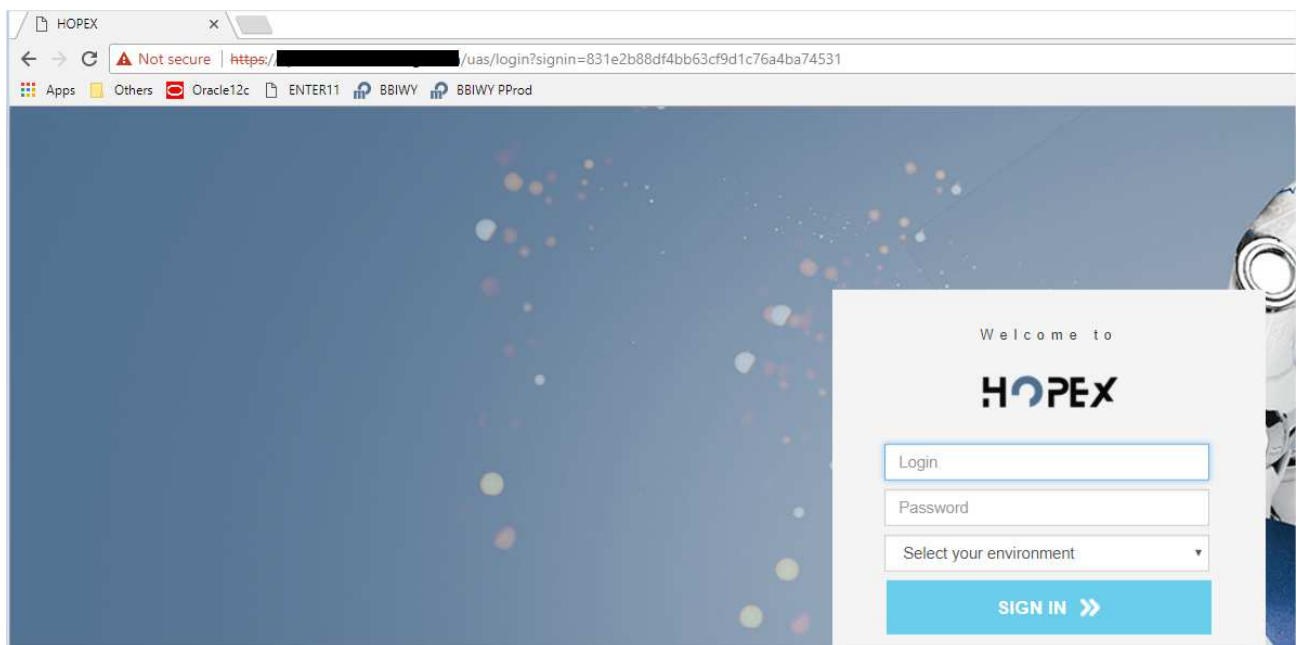


It most likely means that, at a higher level, group policies (or GPOs) are set to remove that group, which is normally granted that right.

If you can modify it yourself, click on "Add User or Group...", and add the "IIS_IUSRS" local group:



Do it for both policies. And restart the application, and IIS. Make sure that the "HOPEX" application pool is started. And test again. If this update was successful, you will see the login page:



If editing those policies is not permitted (grayed out in the interface), you need to explain this to the people in charge of setting up GPOs for the servers, so that they make this modification.

Cluster Deployment – HOPEX V3 Installation Guide



Cluster Deployment – HOPEX V3 Installation Guide	1
Prerequisites.....	5
Operating system.....	5
Tweaks.....	6
Users	6
hosts file.....	7
Server Windows 2016.....	9
Installing .Net 4.7	18
Windows users for MEGA.....	21
Define the group permissions.....	21
Define the COM rights.....	25
Create folders.....	29
Give rights on the proper folder	32
Web/Application Servers	32
RDBMS Server	35
Install MEGA on the Web Servers.....	38
Install MEGA on the MWAS Application Servers	55
Install MEGA on the SSP Application Servers	71
Finish the installation.....	80
Create share folders	80
License share	80
Environment share	84
MegaSite share	85
Configure the license	86
Update the Desktop Heap.....	91
Install SQL Server native client	92
Centralize the MegaSite	95
Managing the RDBMS setup.....	100
Instance and databases	100
Configure the default connection string	100

Creating an environment.....	103
Creating a repository	111
Configure the mega account	114
Provide a password	114
Create assignments.....	116
Configure the services	119
Declare the SSP nodes.....	123
Test the web client.....	124
IIS Tuning.....	128
Application pools recycling.....	128
Manage expiration of HTTP response headers.....	131
Diagnostic Tools.....	134
Latency test with hrping.....	134
Explaining of the command line for hrping:.....	135
First MWAS Server (V-CLUST-MW1)	136
Second MWAS Server (V-CLUST-MW2)	137
First SSP Server (V-CLUST-S1)	138
Second SSP Server (V-CLUST-S2).....	139
Conclusion	139
“RDBMS Diagnostic” Tool	139
First MWAS Server (V-CLUST-MW1)	148
Second MWAS Server (V-CLUST-MW2)	150
First SSP Server (V-CLUST-S1).....	152
Second SSP Server (V-CLUST-S2).....	154
Conclusion	156
Activation and deployment of stored procedures for SQL Server	156
Annex.....	160
The log files	160
Restart the web application	160
Automatic restart with a Mega tool	160
Configure SMTP	163

Summary

This document describes all necessary steps to install MEGA Hopex in V3, on a clustered architecture.

PREREQUISITES

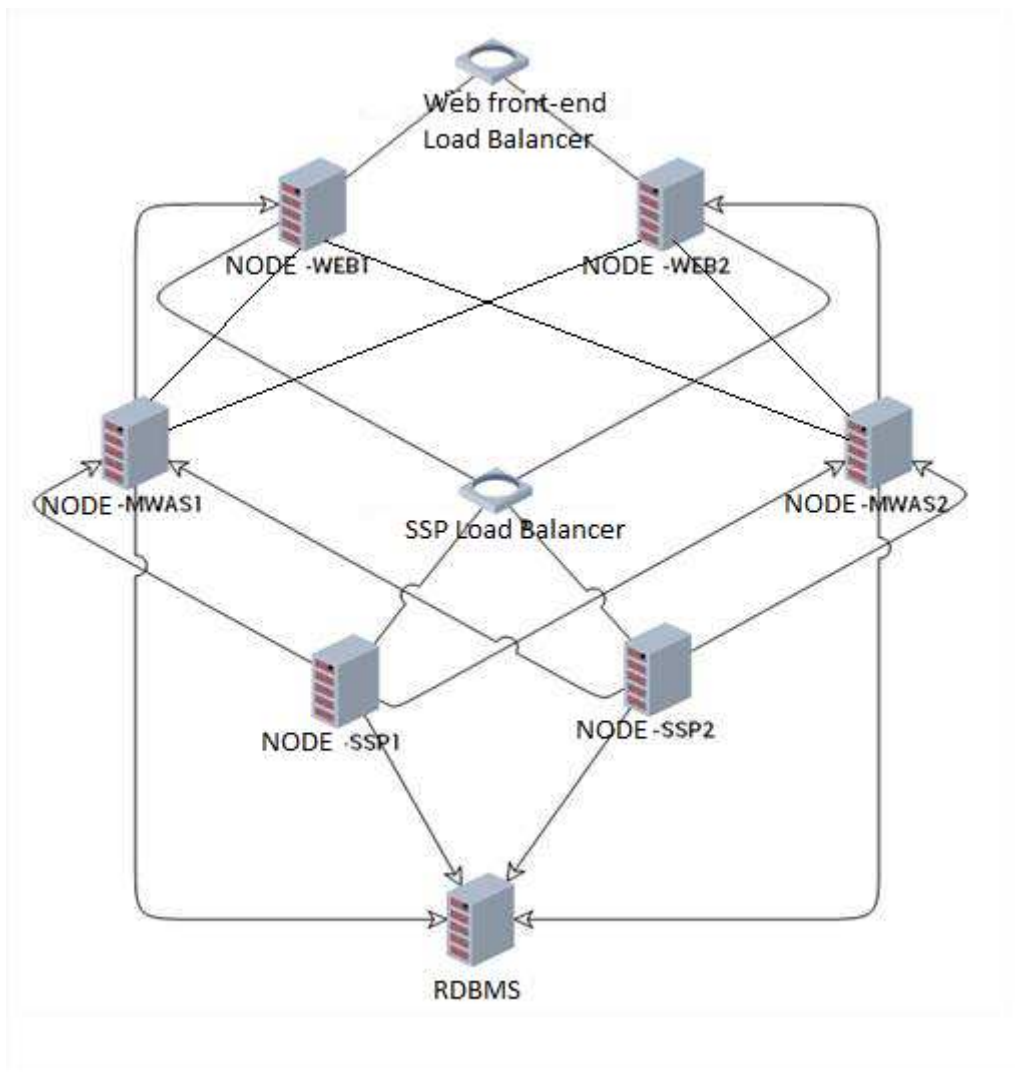
Operating system

In this deployment, **2016** was the chosen OS for all virtual machines.

The list of servers is the following:

- 137.74.87.161 (V-CLUST-W1): first web server.
- 137.74.87.162 (V-CLUST-W2): second web server.
- 137.74.87.163: IP of the Load Balancer in front of the web servers -> use this IP address when setting up URLs about Hopex front-end website.
- 137.74.87.164 (V-CLUST-MW1): first application (MWAS) server.
- 137.74.87.165 (V-CLUST-MW2): second application (MWAS) server.
- 137.74.87.167 (V-CLUST-S1): first SSP server.
- 137.74.87.168 (V-CLUST-S2): second SSP server.
- 137.74.87.169: IP of the Load Balancer in front of the SSP servers -> use this IP address when setting up URL of the SSP.
- 137.74.87.170 (V-CLUST-SQL): hosting the SQL Server instance.

This is the architecture schema:



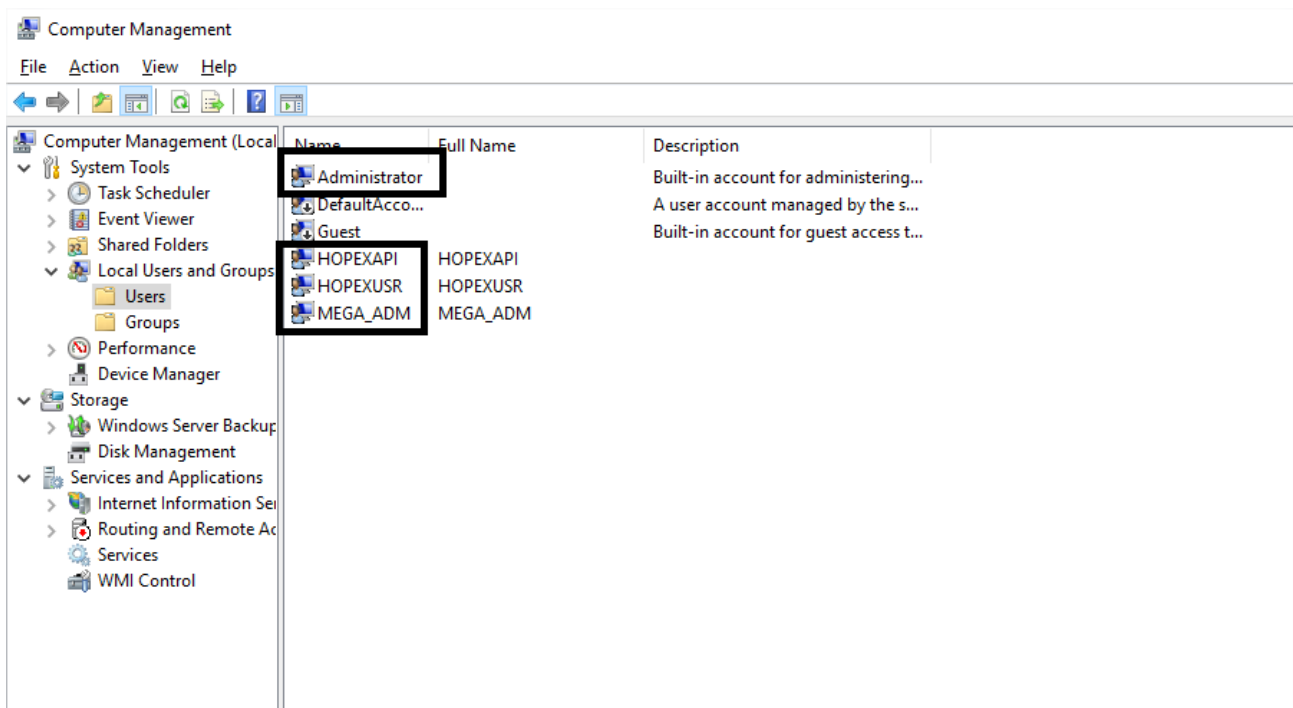
Tweaks

Because we are in a deployment with no domain, and no Active Directory, some things are not possible unless additional configuration is done.

If you deploy with this kind of scenario but with servers within a domain, you won't have to perform the two following sections. And we recommend to use domain accounts instead of local ones.

Users

First, we create the same local users on all servers:



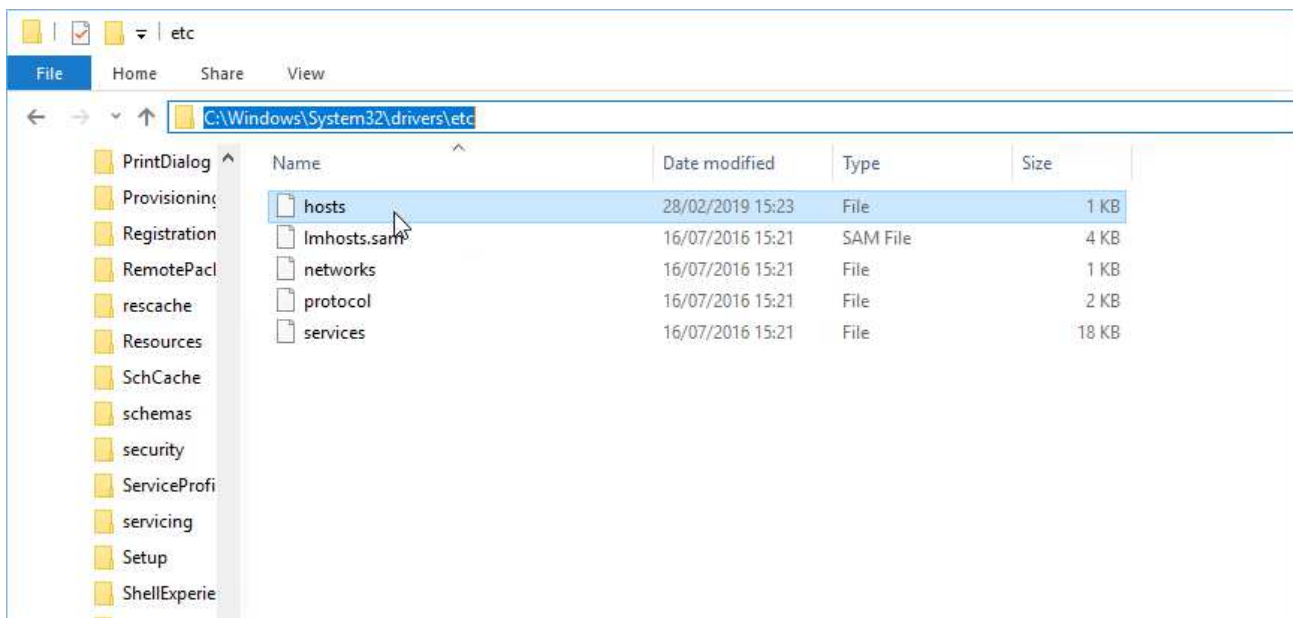
On a single server, each account can have its own password. However, **for a specific username**, you need to provide **the same password on all servers**.

This is the only way to make you access a remote share with a user that is local to your server.

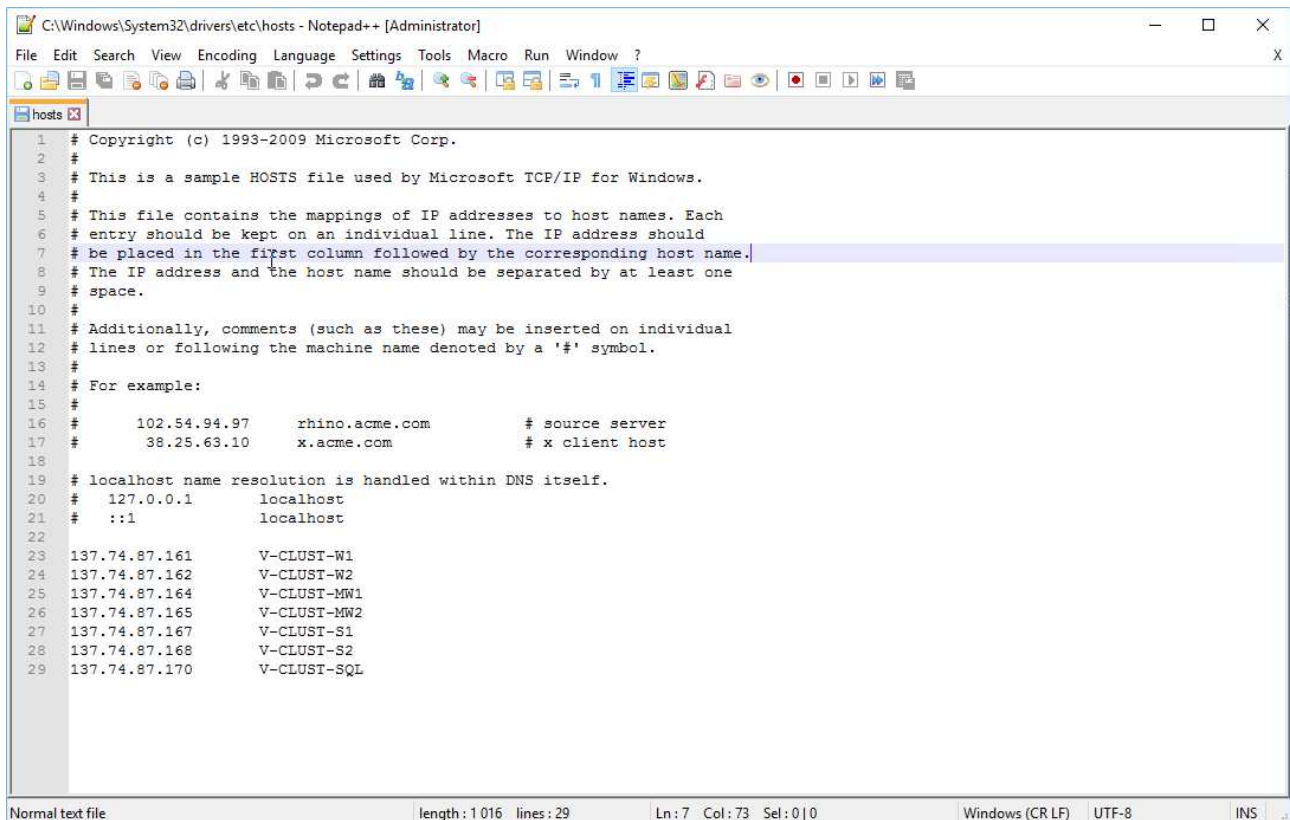
hosts file

Because the DNS server won't know the correspondence between the server name, and its IP address, we use an internal mechanism of Windows to make a link between name and IP address.

On all servers, we edit the file "hosts" in "C:\Windows\System32\drivers\etc":



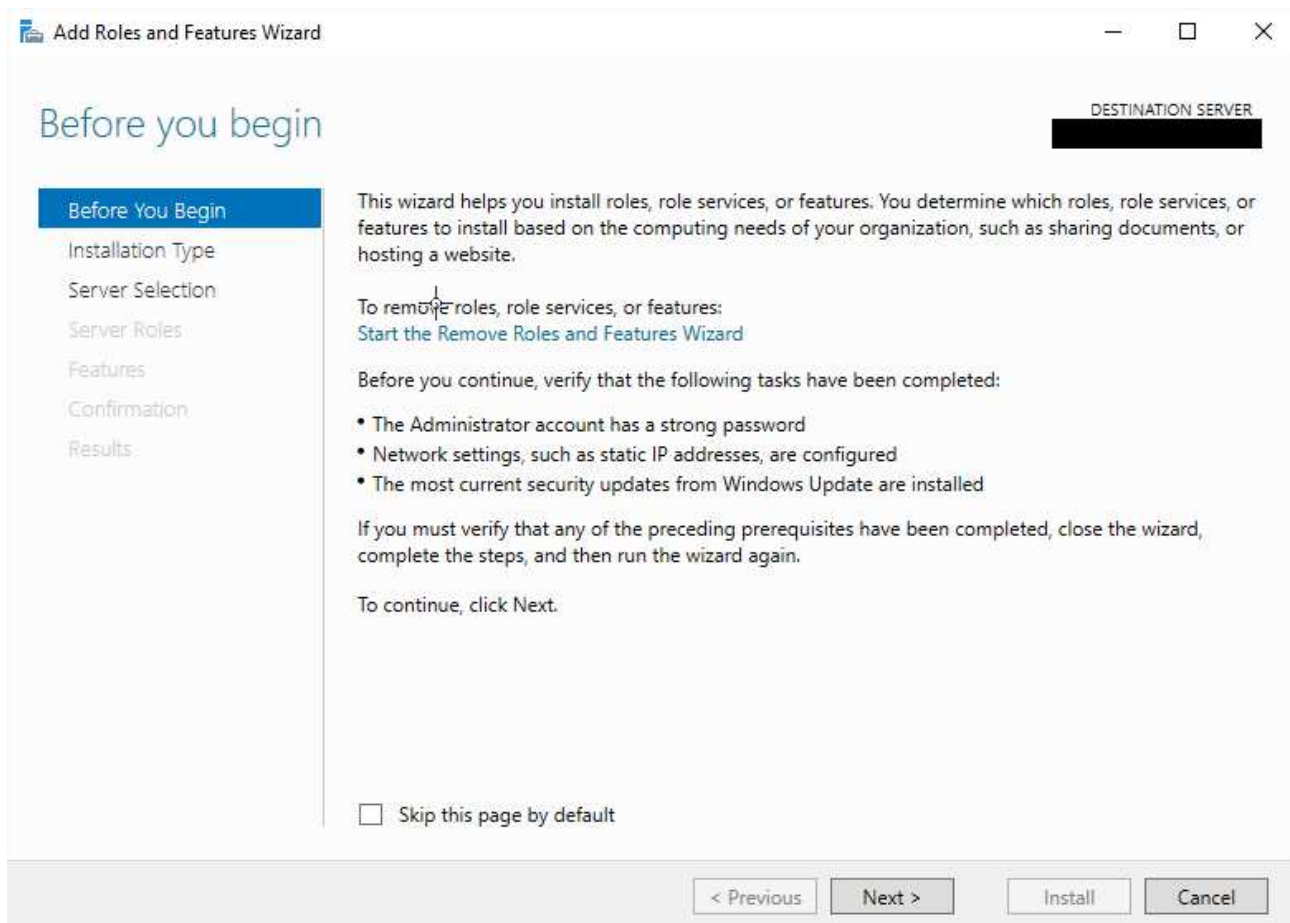
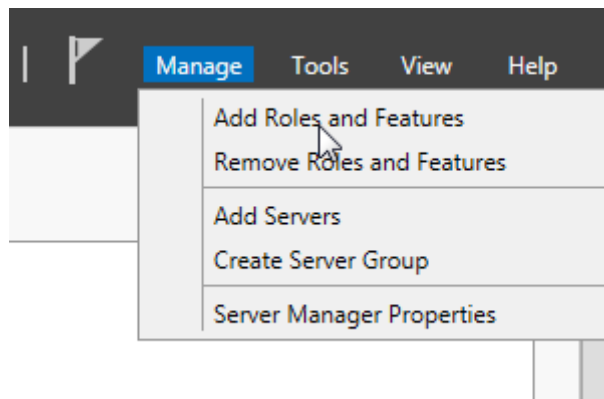
And put the lines that help the server to know which IP address to contact when using the hostname of the other servers:

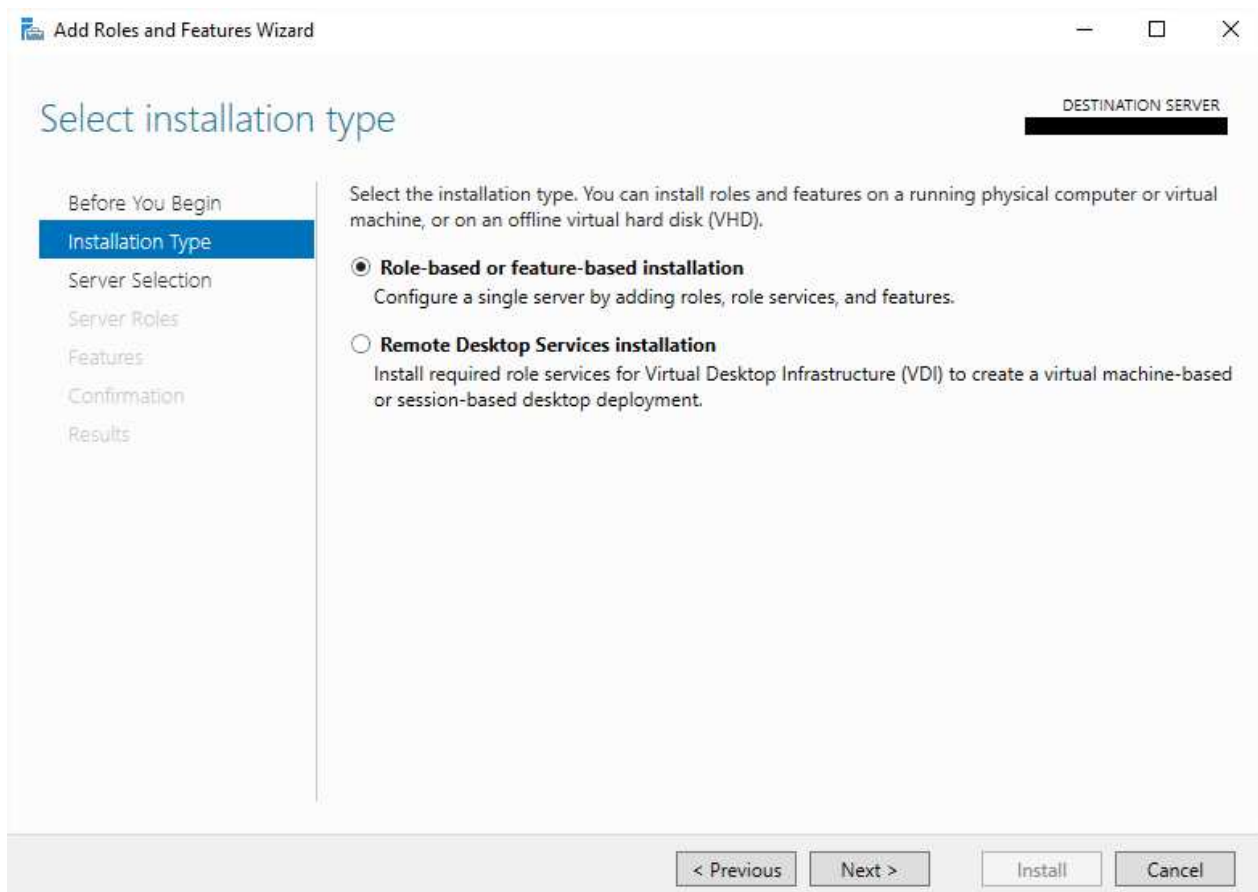


```
1 # Copyright (c) 1993-2009 Microsoft Corp.
2 #
3 # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
4 #
5 # This file contains the mappings of IP addresses to host names. Each
6 # entry should be kept on an individual line. The IP address should
7 # be placed in the first column followed by the corresponding host name.
8 # The IP address and the host name should be separated by at least one
9 # space.
10 #
11 # Additionally, comments (such as these) may be inserted on individual
12 # lines or following the machine name denoted by a '#' symbol.
13 #
14 # For example:
15 #
16 #       102.54.94.97       rhino.acme.com       # source server
17 #       38.25.63.10       x.acme.com           # x client host
18
19 # localhost name resolution is handled within DNS itself.
20 #   127.0.0.1       localhost
21 #   ::1             localhost
22
23 137.74.87.161       V-CLUST-W1
24 137.74.87.162       V-CLUST-W2
25 137.74.87.164       V-CLUST-MW1
26 137.74.87.165       V-CLUST-MW2
27 137.74.87.167       V-CLUST-S1
28 137.74.87.168       V-CLUST-S2
29 137.74.87.170       V-CLUST-SQL
```

Server Windows 2016

Install the "Web Server (IIS)" and "File Services roles" on both web front-end and application server.
From the "Server Manager", choose this option:





Add Roles and Features Wizard

Select destination server

DESTINATION SERVER

Before You Begin
Installation Type
Server Selection
Server Roles
Features
Confirmation
Results

Select a server or a virtual hard disk on which to install roles and features.

☒ Select a server from the server pool
☐ Select a virtual hard disk

Server Pool

Filter:

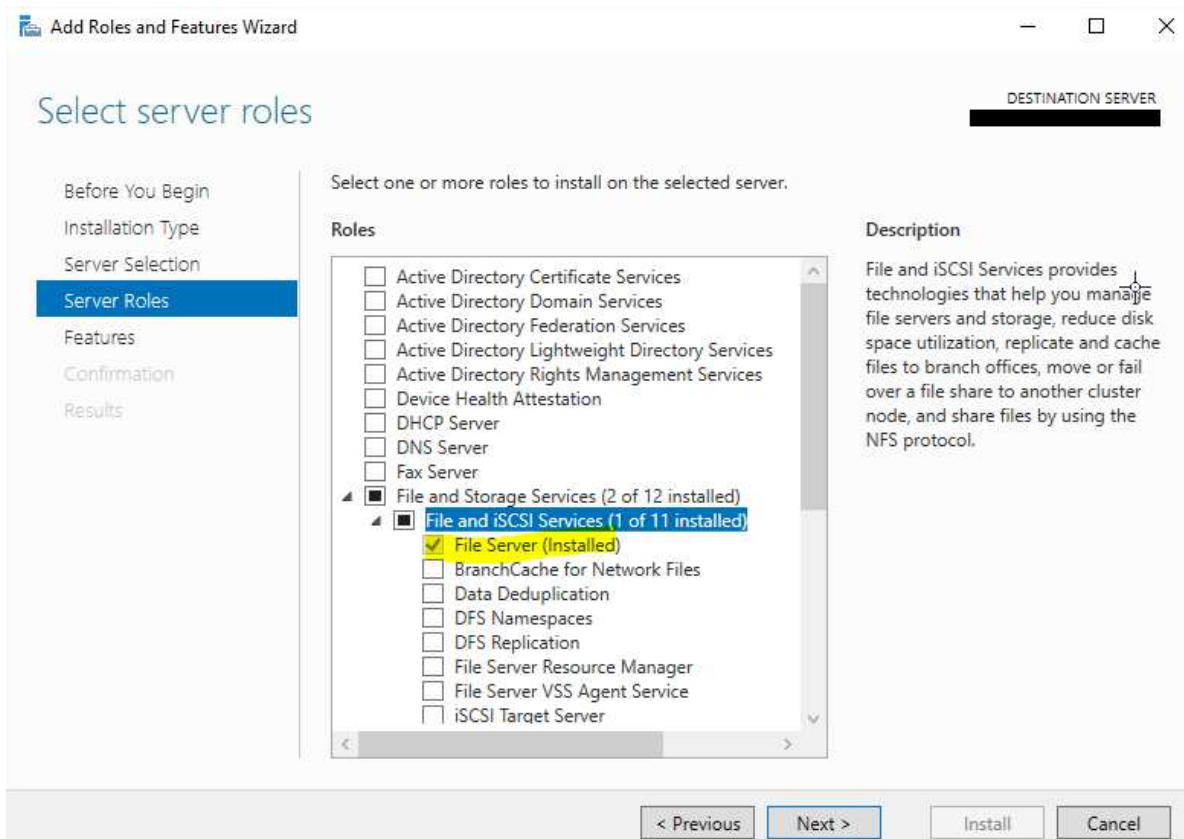
Name	IP Address	Operating System
ZZNT238V.intern.zollner....	10.99.2.238	Microsoft Windows Server 2016 Standard

1 Computer(s) found

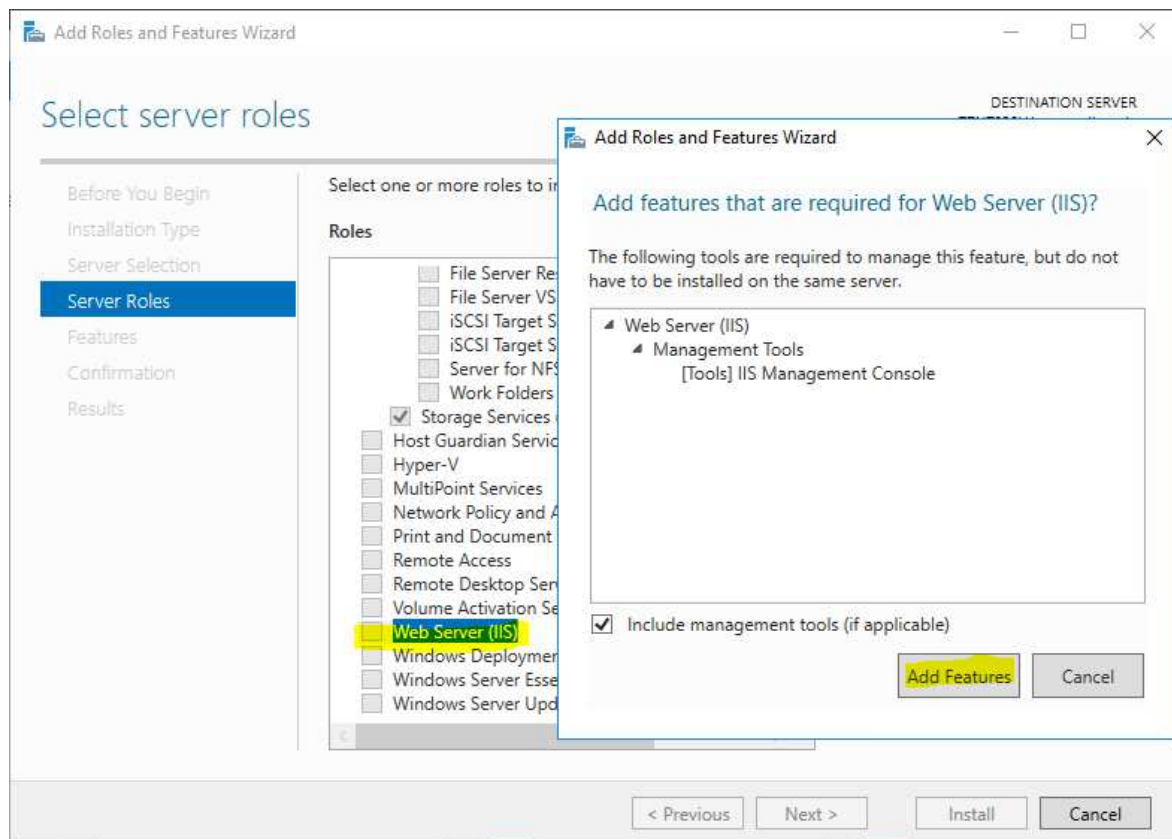
This page shows servers that are running Windows Server 2012 or a newer release of Windows Server, and that have been added by using the Add Servers command in Server Manager. Offline servers and newly-added servers from which data collection is still incomplete are not shown.

< Previous Next > Install Cancel

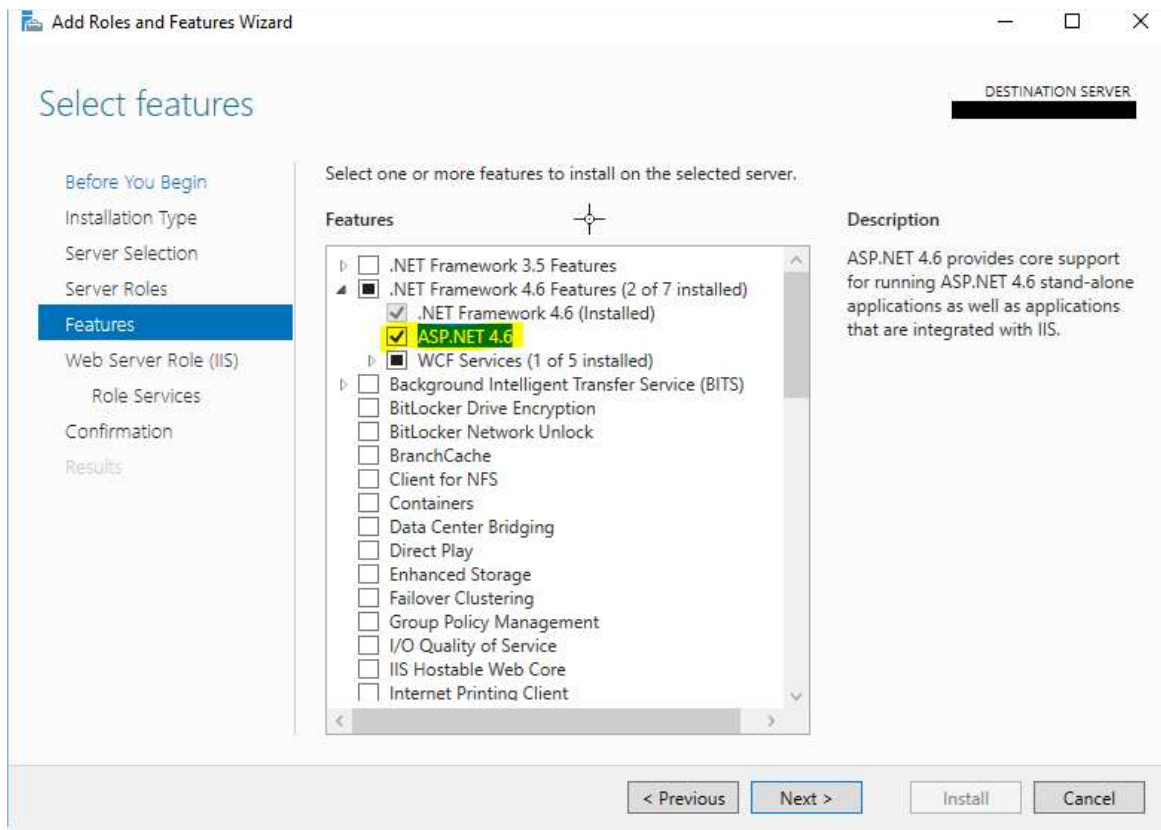
Check the "File Server" is installed. If not, activate it:



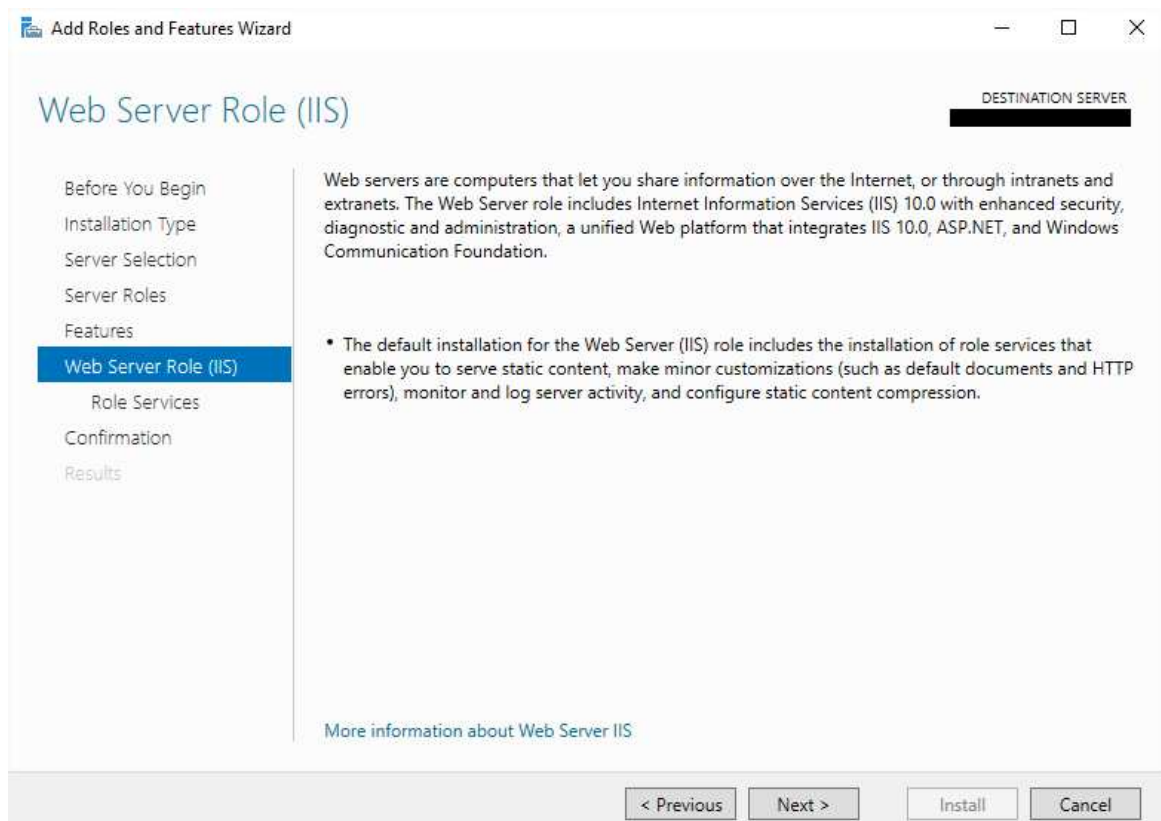
We activate the “Web Server (IIS)” and its features, and click “Next”:



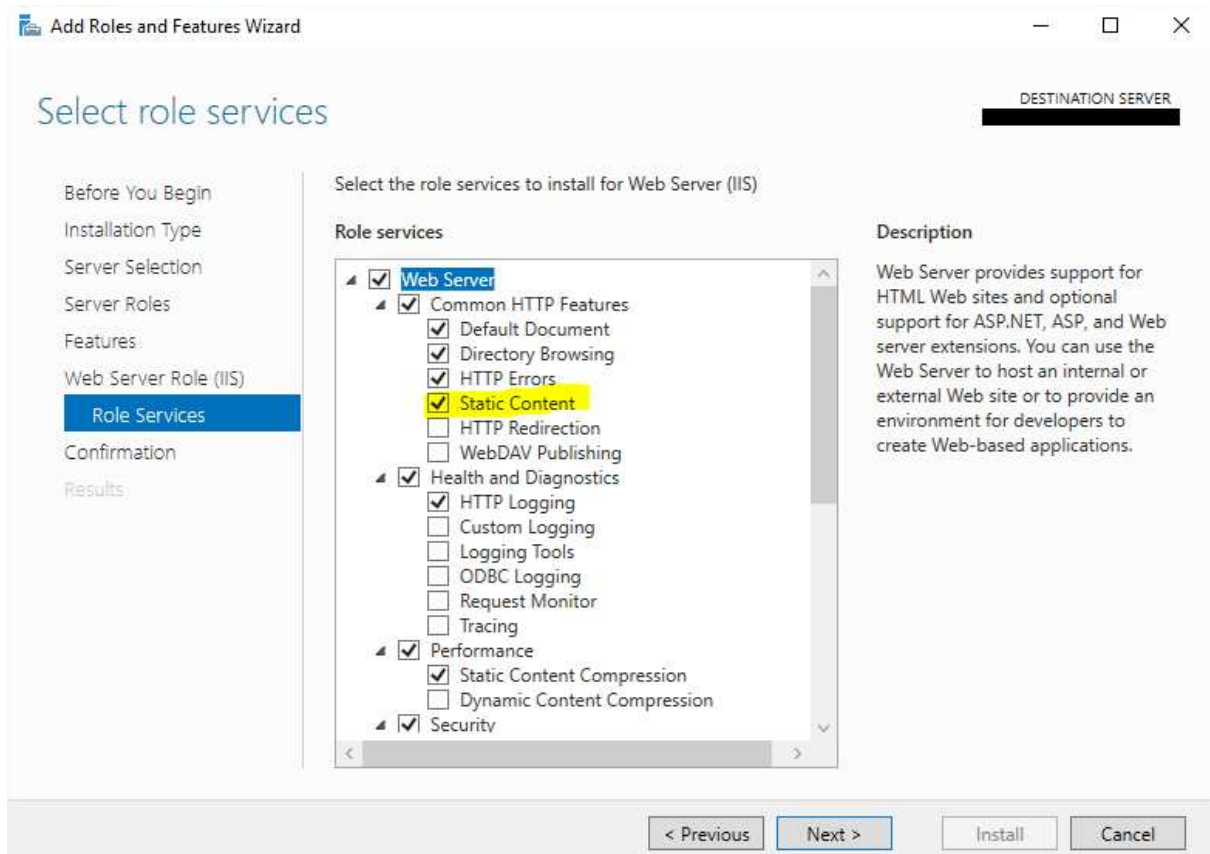
Then, in the „Features“ window, we activate the „ASP.NET 4.6“, and click „Next“:



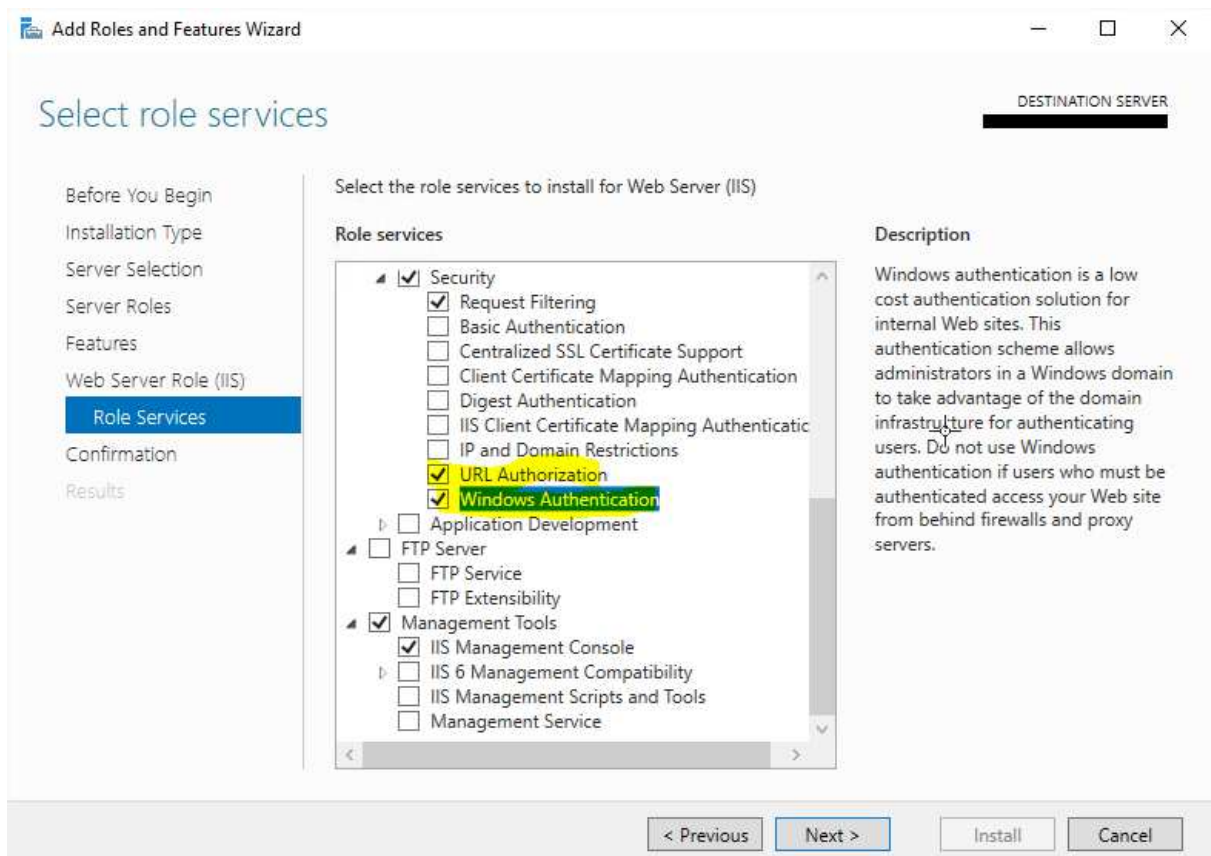
We pass this screen:



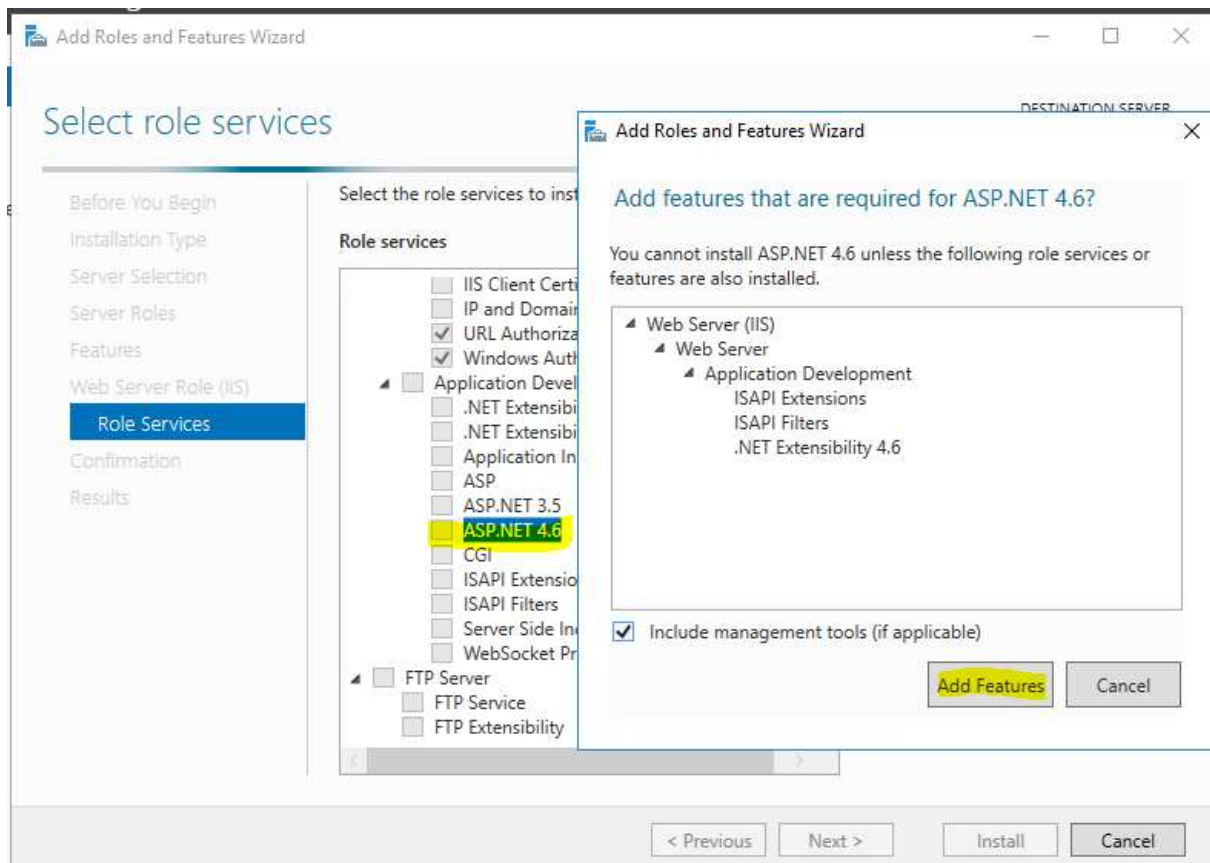
Then, in the „Role Services“ part, we check that „Static Content“ is enabled:



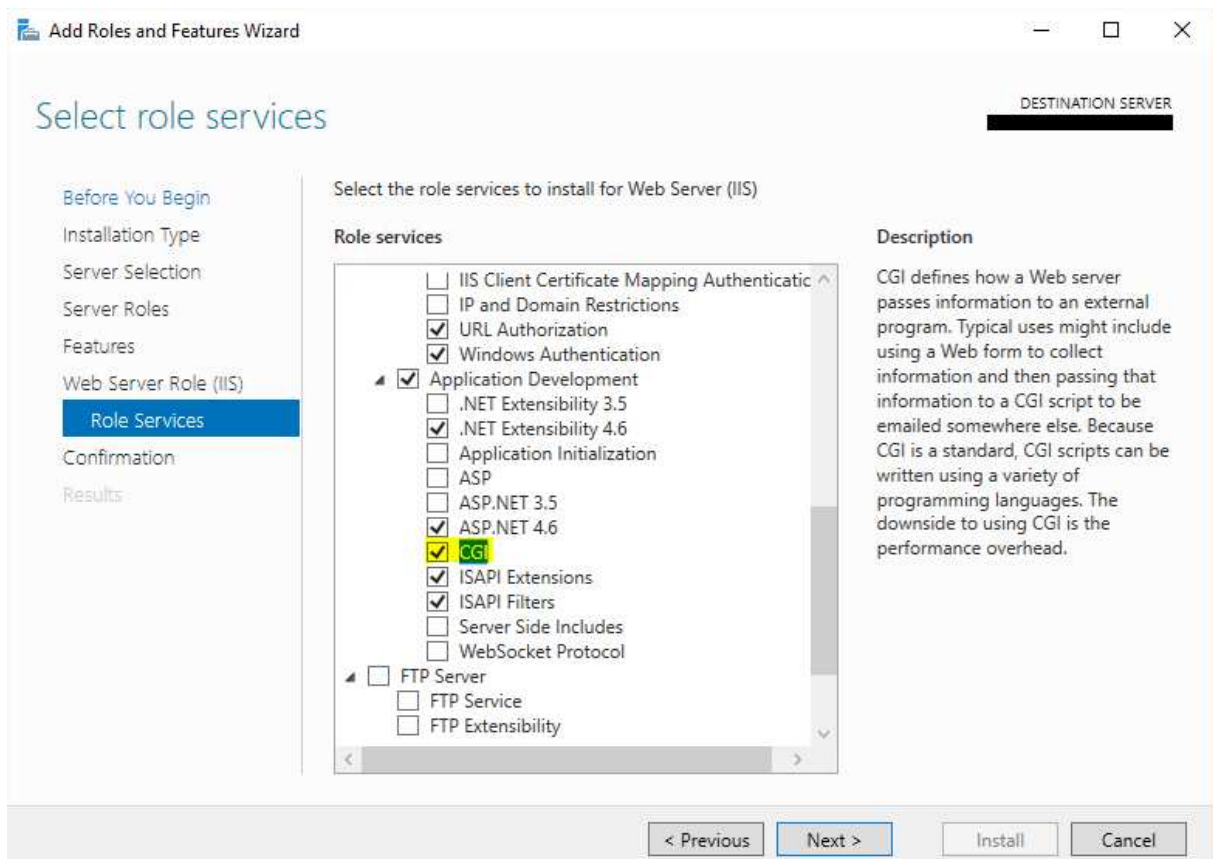
In „Security“, we manually add „URL Authorization“ and „Windows Authentication“:



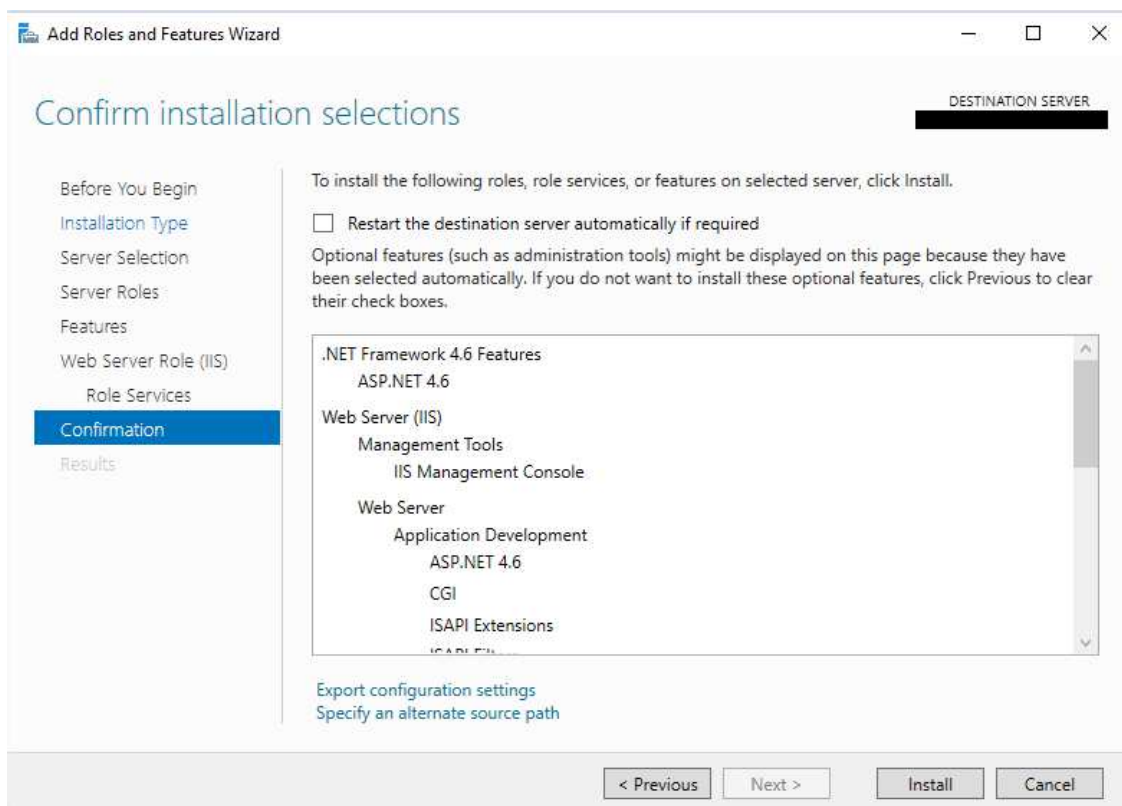
In „Application Development“, we add „ASP.NET 4.6“ and related features:



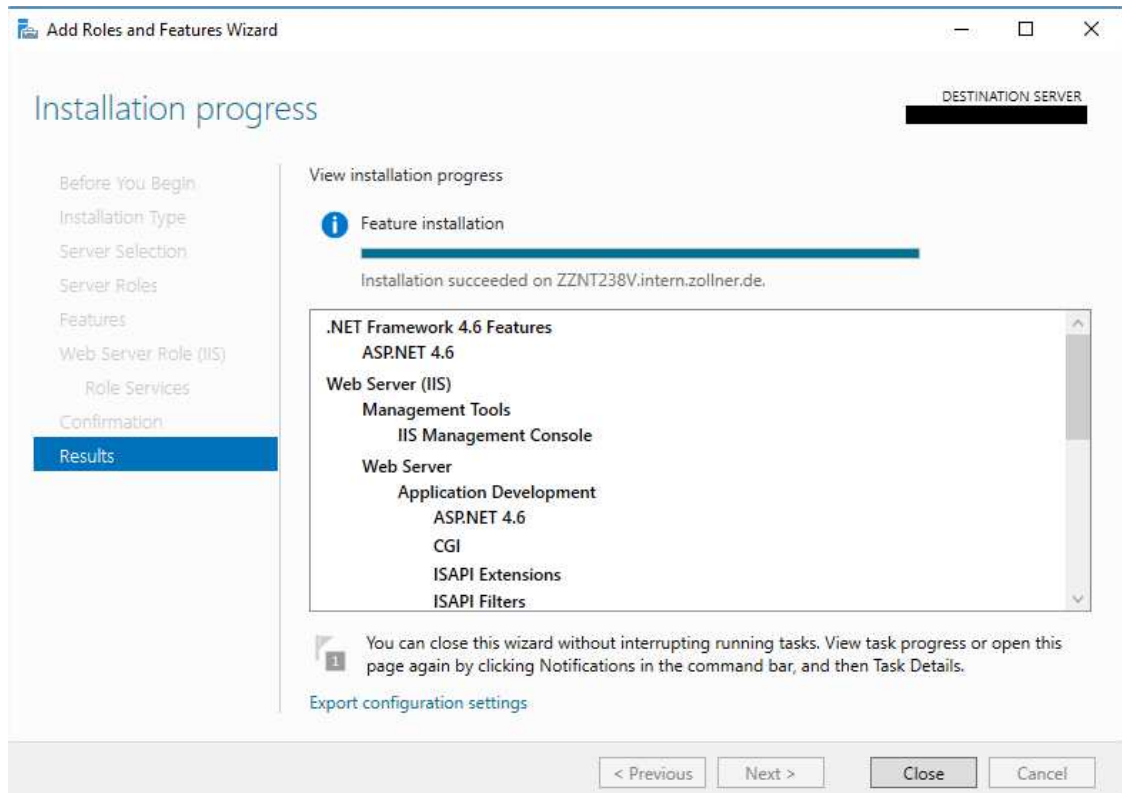
As well as „CGI“, then click „Next“:



And on „Install“:



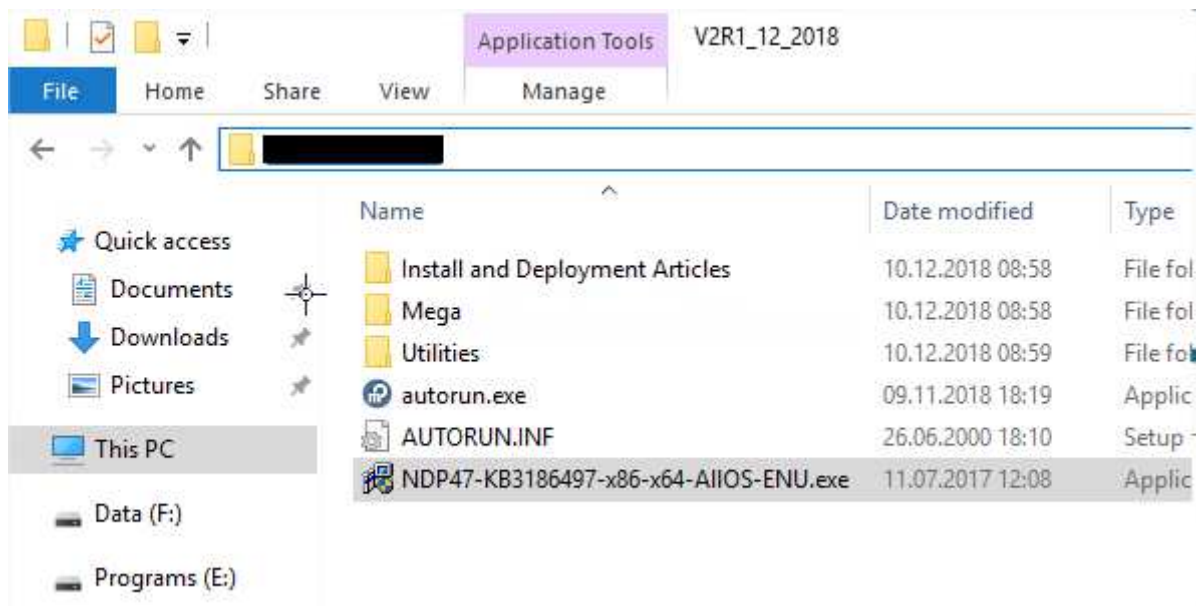
Eventually, on „Close“:



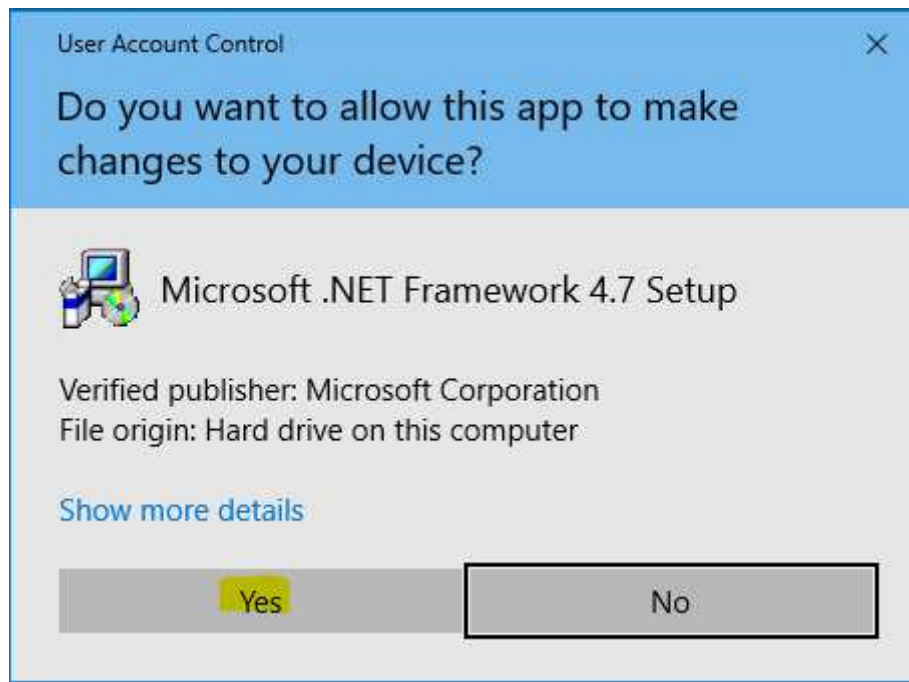
Installing .Net 4.7

With Mega Hopex V2R1, we need to use at least version **4.6.2** of the .Net Framework.

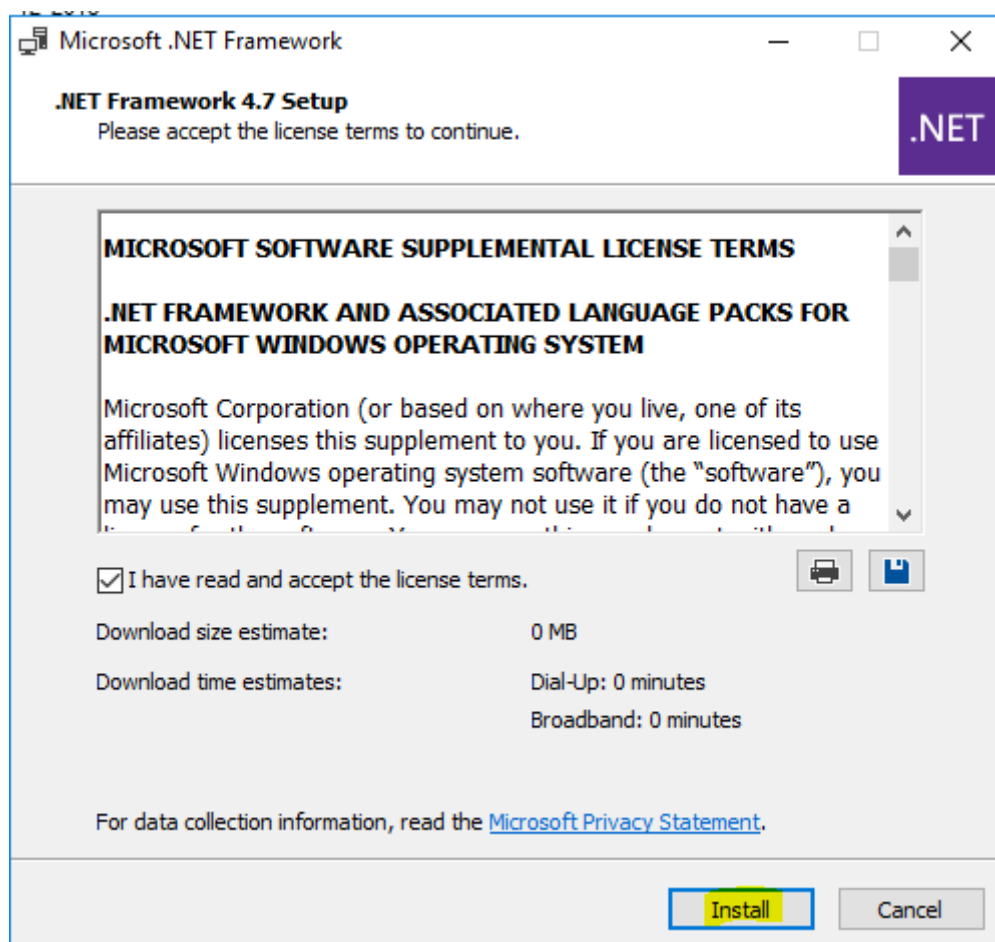
We deployed the **4.7** version that was copied on each server:



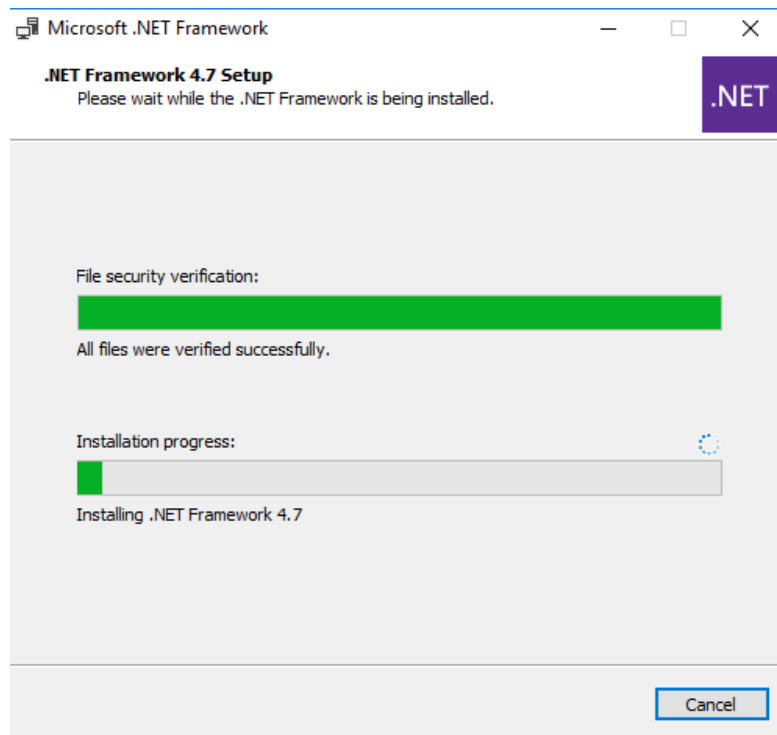
Click „Yes“:



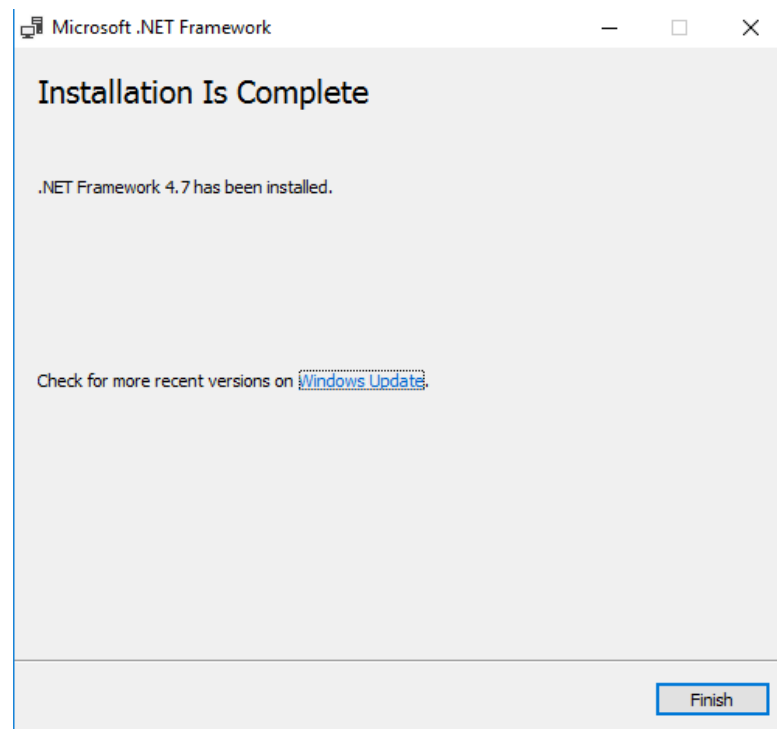
Accept the license terms and click „Install“:



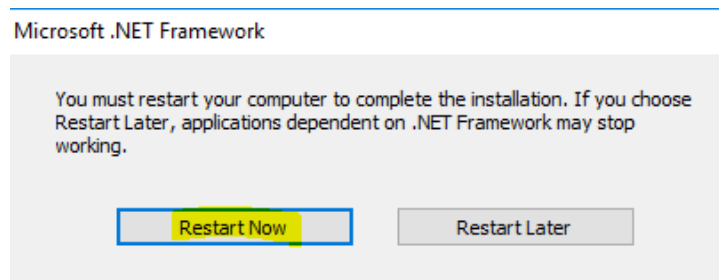
Let the installation proceed:



Then click „Finish“:



Restart the servers:



Windows users for MEGA

When you install MEGA Hopex, at least one user is necessary to manage the process authentication. It's recommended to avoid using a user that is Local Administrator of the server. You then need an additional user, preferably in the Domain.

If it is not possible, and that all resources, such as the license file and the Mega environments, are hosted locally, it is possible to have a local user.

This user is going to be impersonified for the web applications. All actions done in MEGA Hopex will be done using the identity of this user.

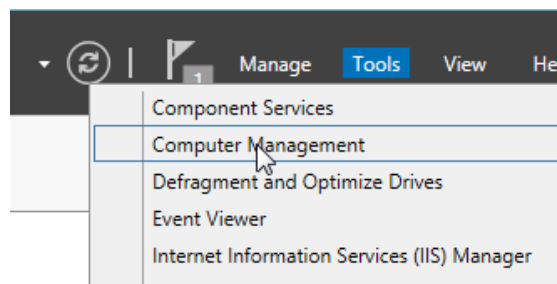
In our case, we want to deploy Mega Hopex with all its features, including the Web Services part, called "HOPEX API". Because of that, we need a second impersonate user, as you cannot use the same account for both the web front-end, and the web services.

We then have (".\\" means that those are local users):

- **.\HOPEXUSR**: for the web front-end.
- **.\HOPEXAPI**: for the web services.

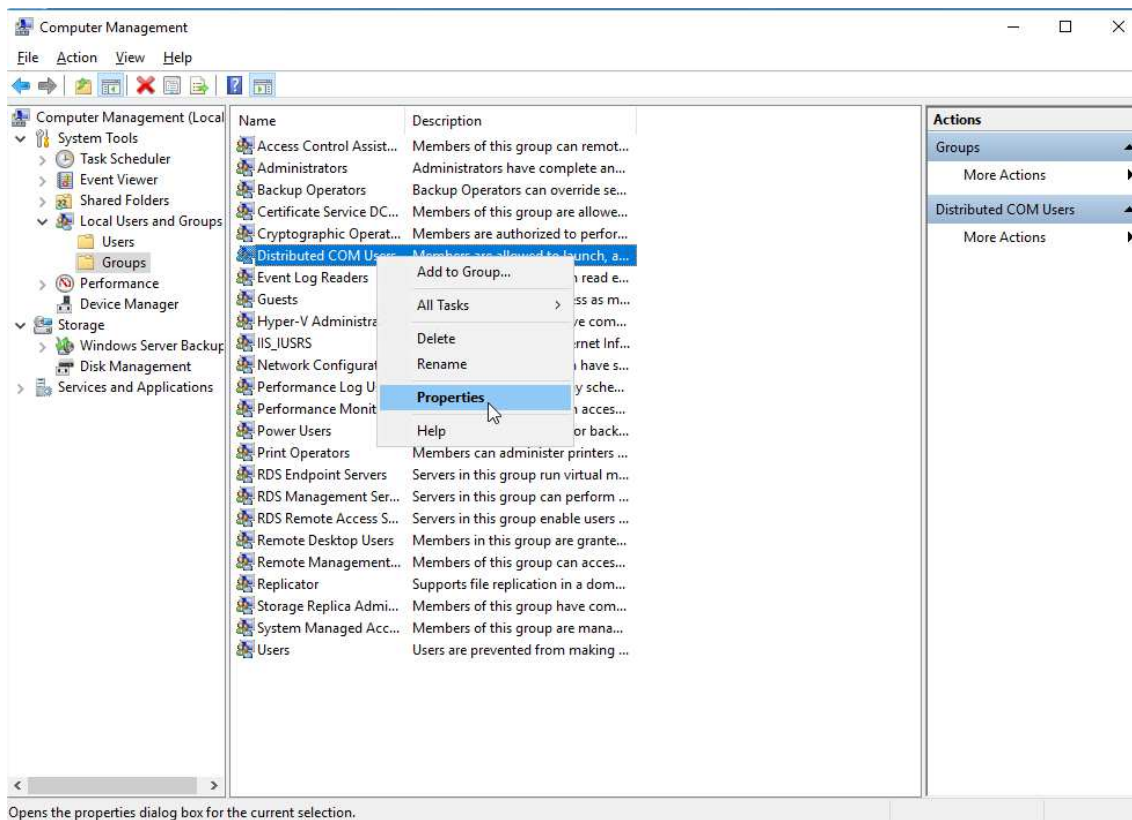
Define the group permissions

In "Computer Management":

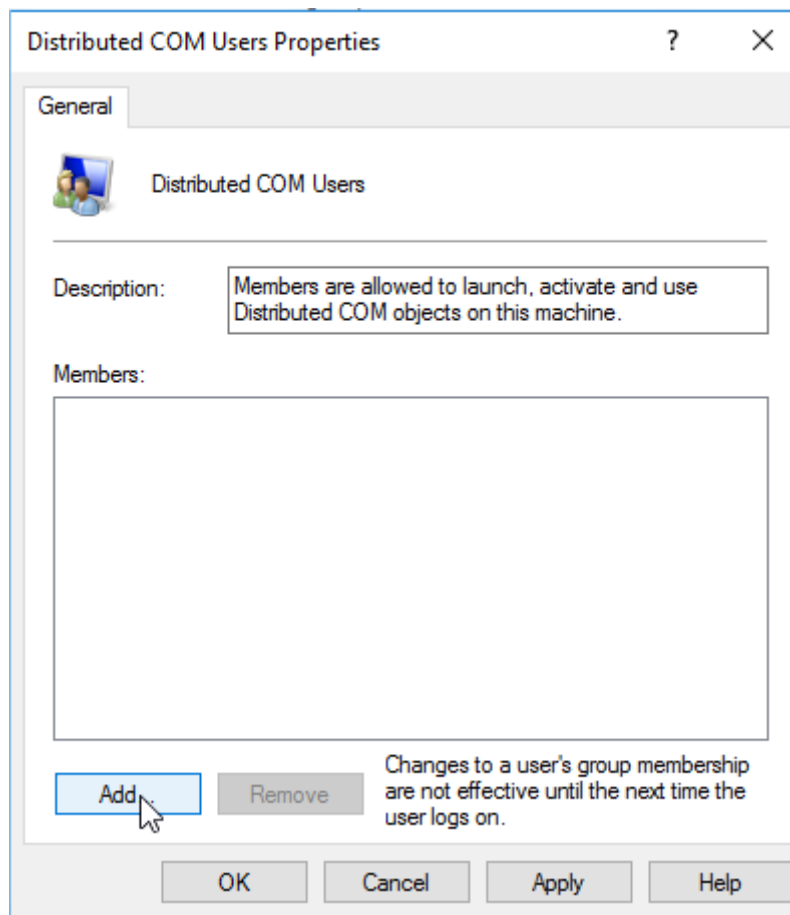


The impersonate user needs to be part of both the local groups "IIS_IUSRS", and "Distributed COM Users".

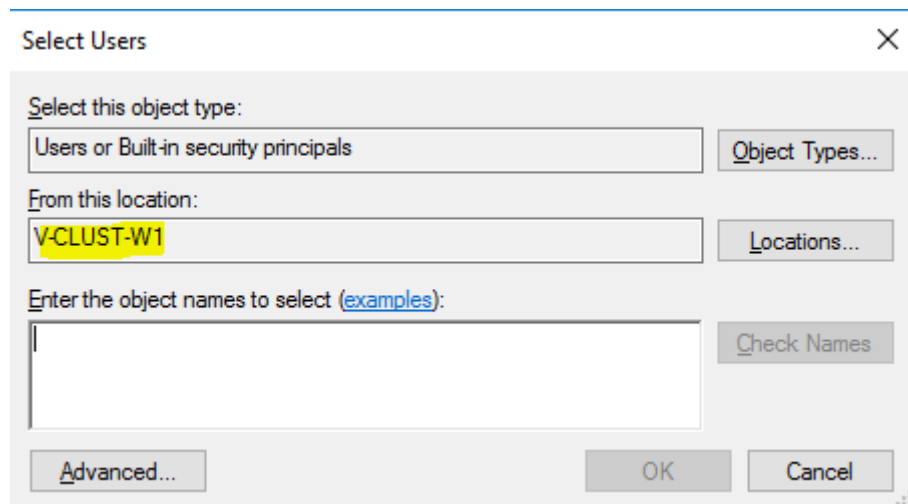
We start with "Distributed COM Users" by opening its properties:



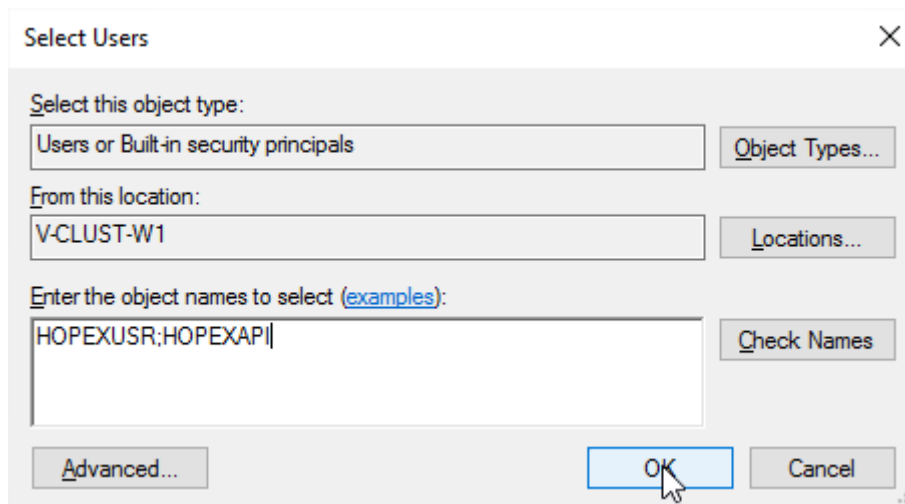
Click "Add":



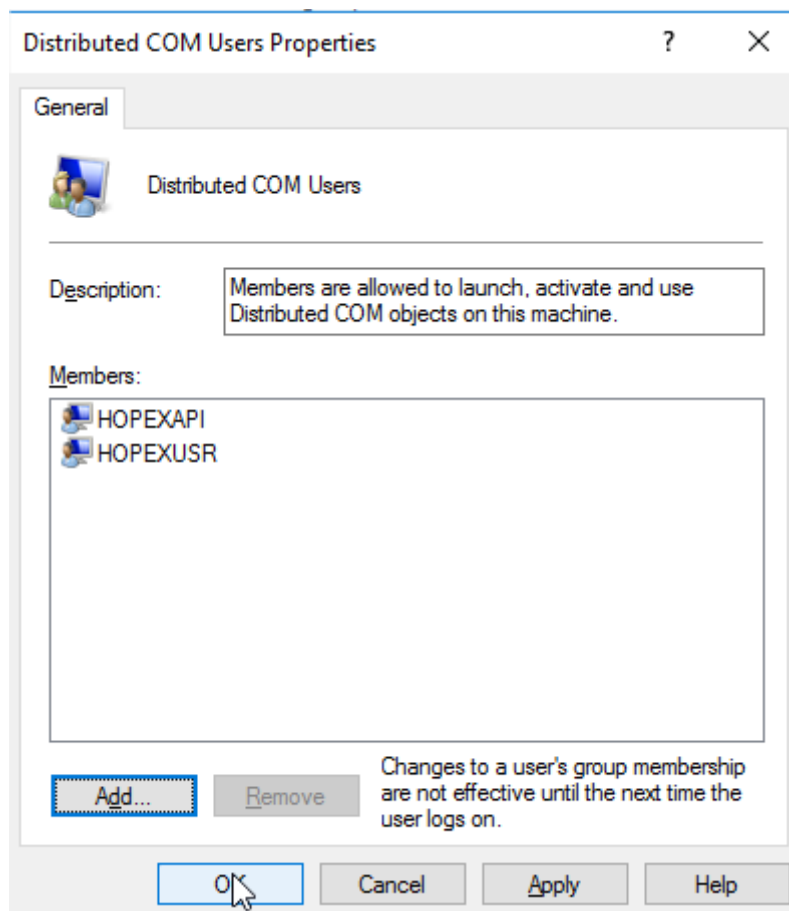
In our case, the users are local, so we need to make sure that the “Locations” are set to the server itself:



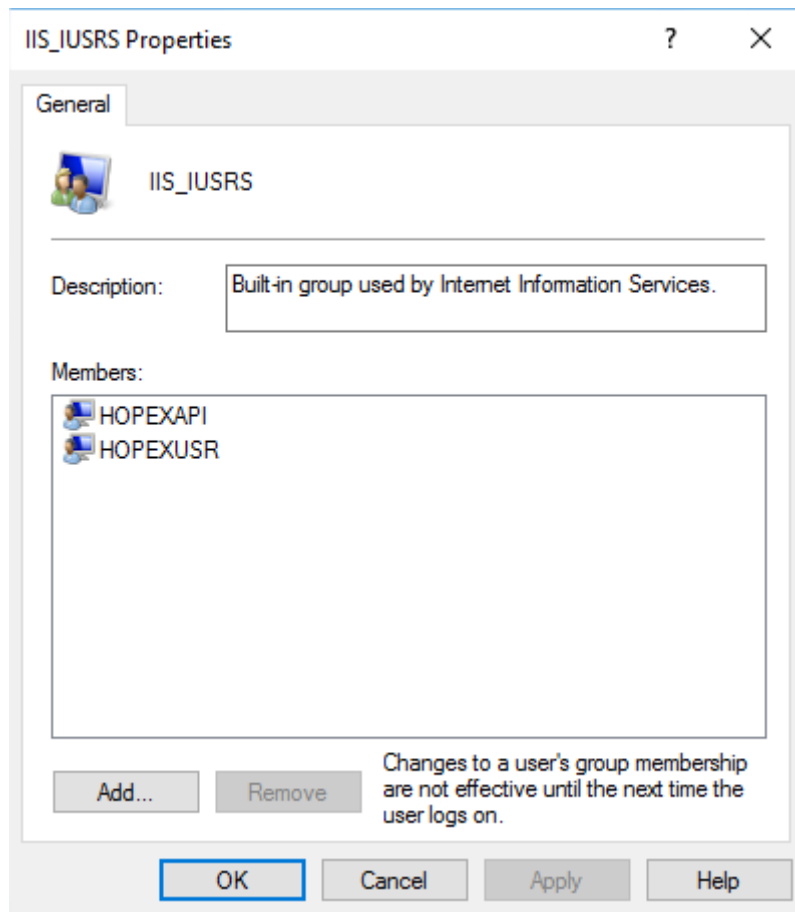
Then we type the names of the two account, with the ‘;’ separator, and click “OK”:



Click "OK" to validate:

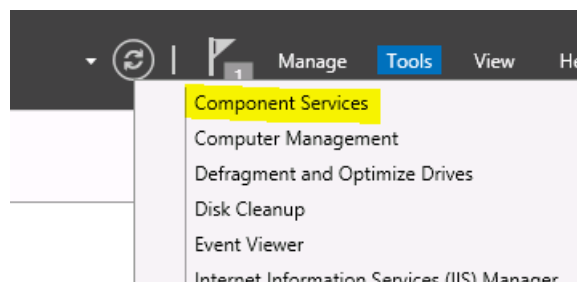


Same steps for the « IIS_IUSRS » group:

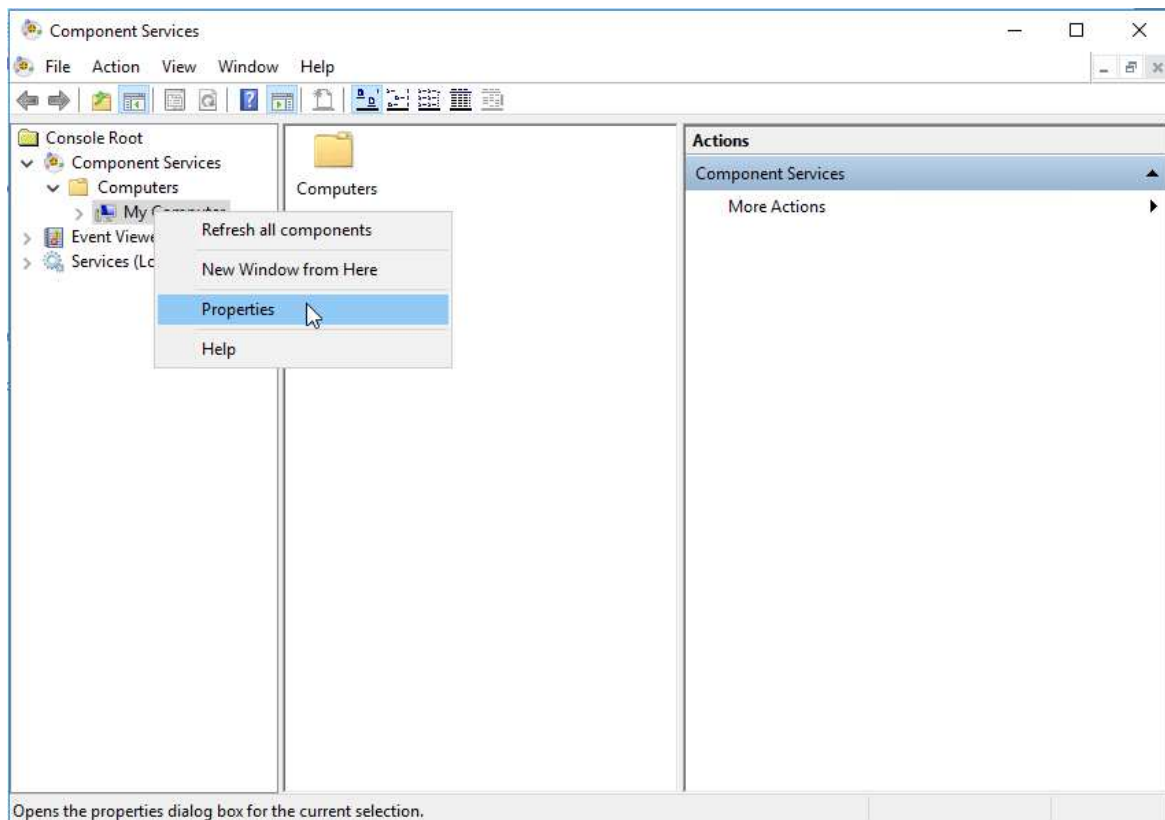


Define the COM rights

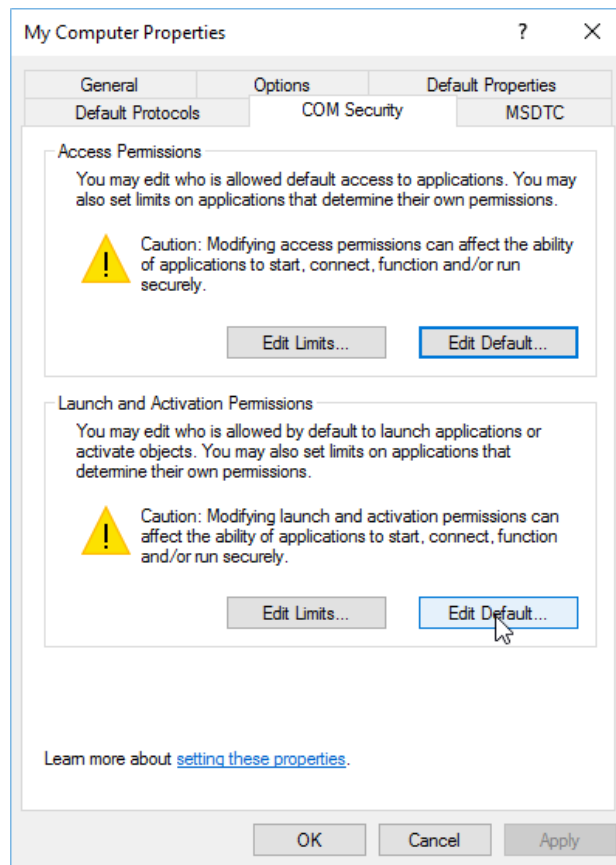
The impersonate user must be able to launch COM applications by default. To give that right, expand the node "Component Services" from the "server Manager":



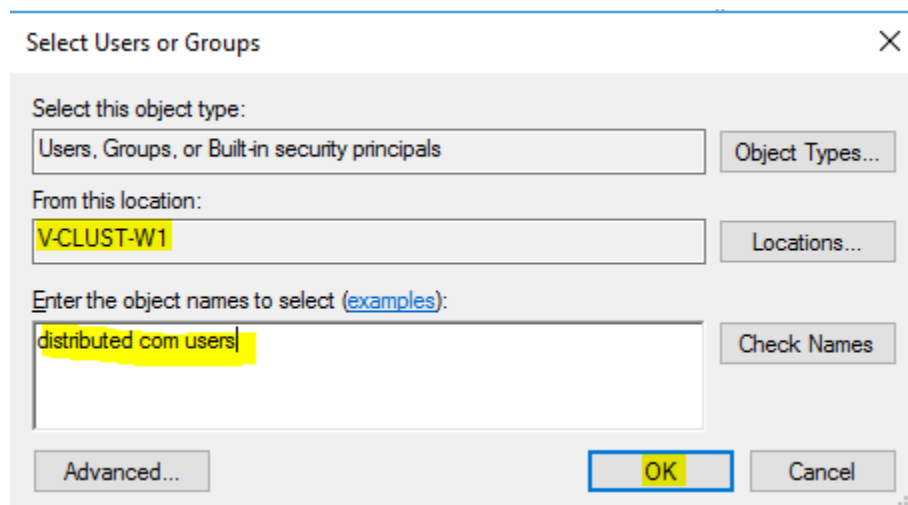
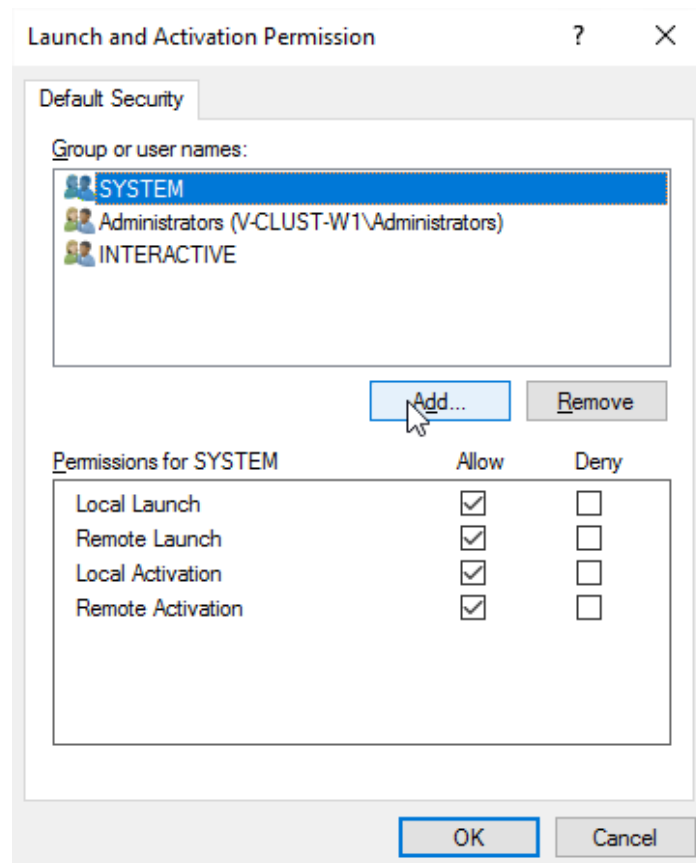
Then, expand the tree to see "Component Services -> Computers -> My computer":
Do a right-click "My Computer" and choose "Properties".



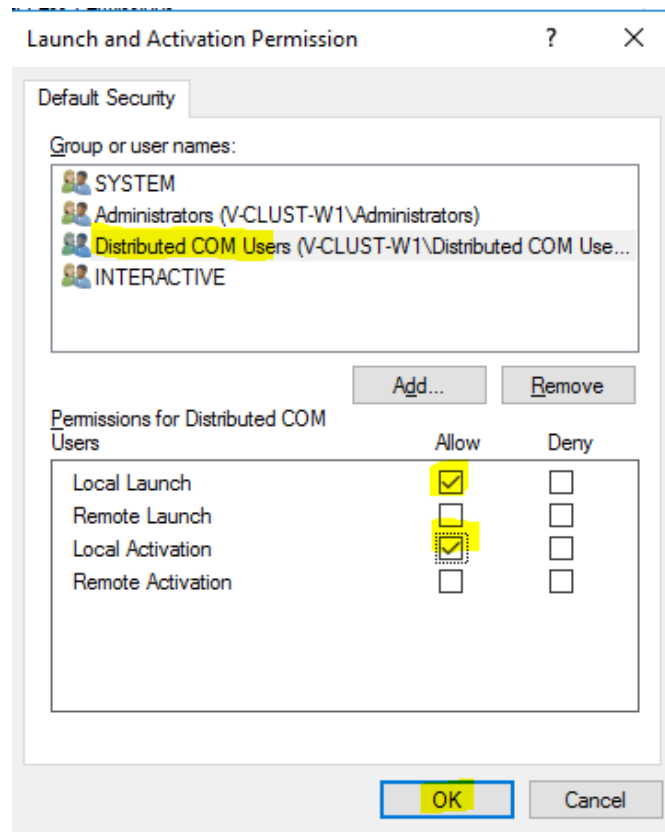
Go to the tab “COM Security” and click the “Edit Default...” button in the “Launch and Activation Permissions” zone. Add the “Distributed COM Users” group to have the “Local Launch” et “Local Activation” rights:



Add the **local** « Distributed COM Users » group, and give it « local launch » and « local activation » :



Grant it “Local Launch” and “Local Activation”, and click “OK” to validate:

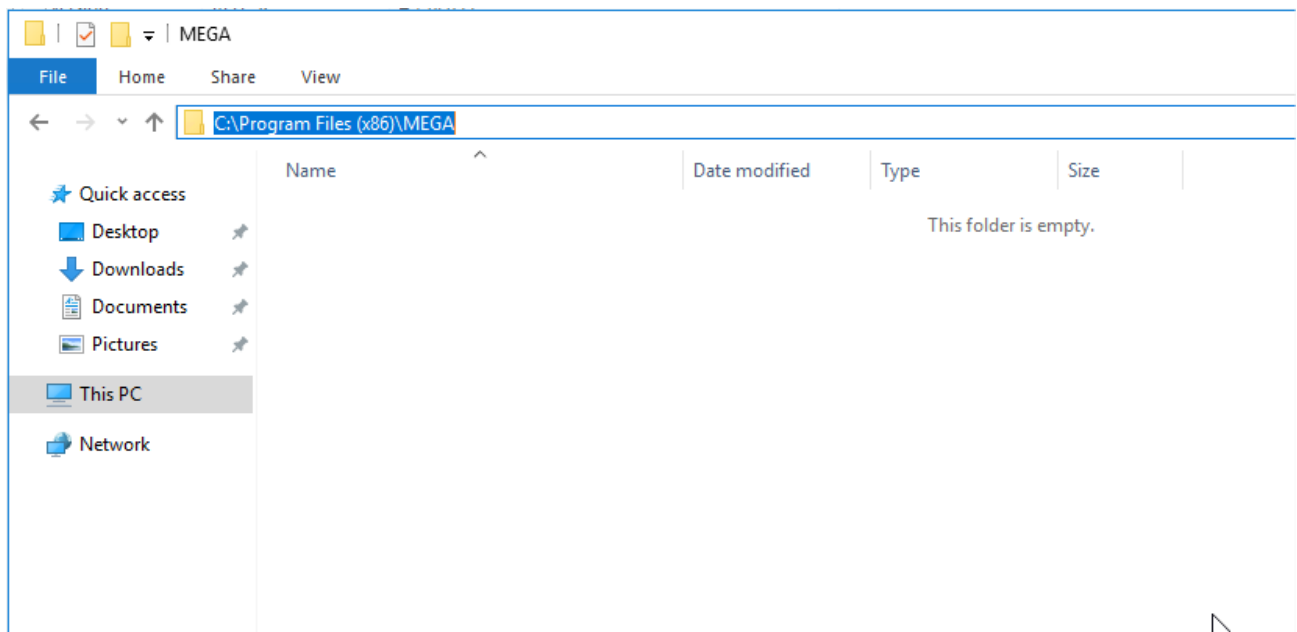


Close the properties, and the "Component Services" feature.

Create folders

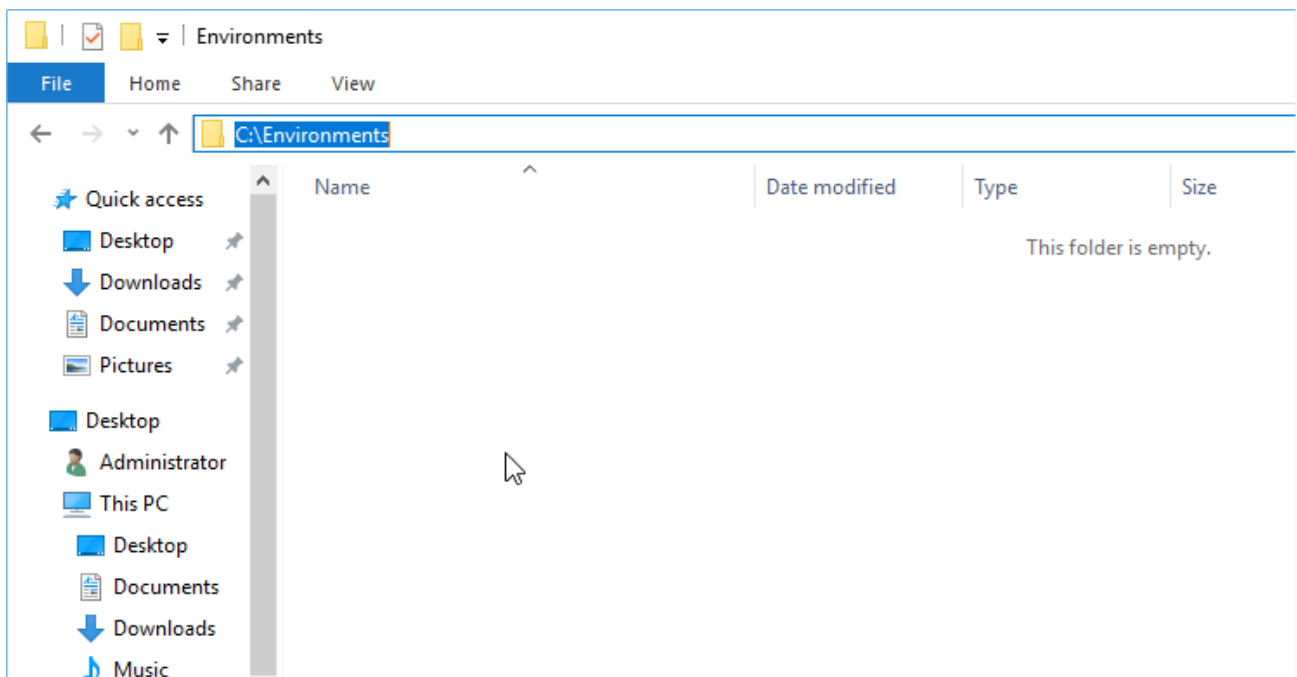
On this specific deployment, we only have a C drive available, so we will use the default installation folders for the web components, and the Mega binaries.

To gain some time later on, we create **on all servers except the RDBMS server**, the "C:\Program Files (x86)\MEGA" folder:

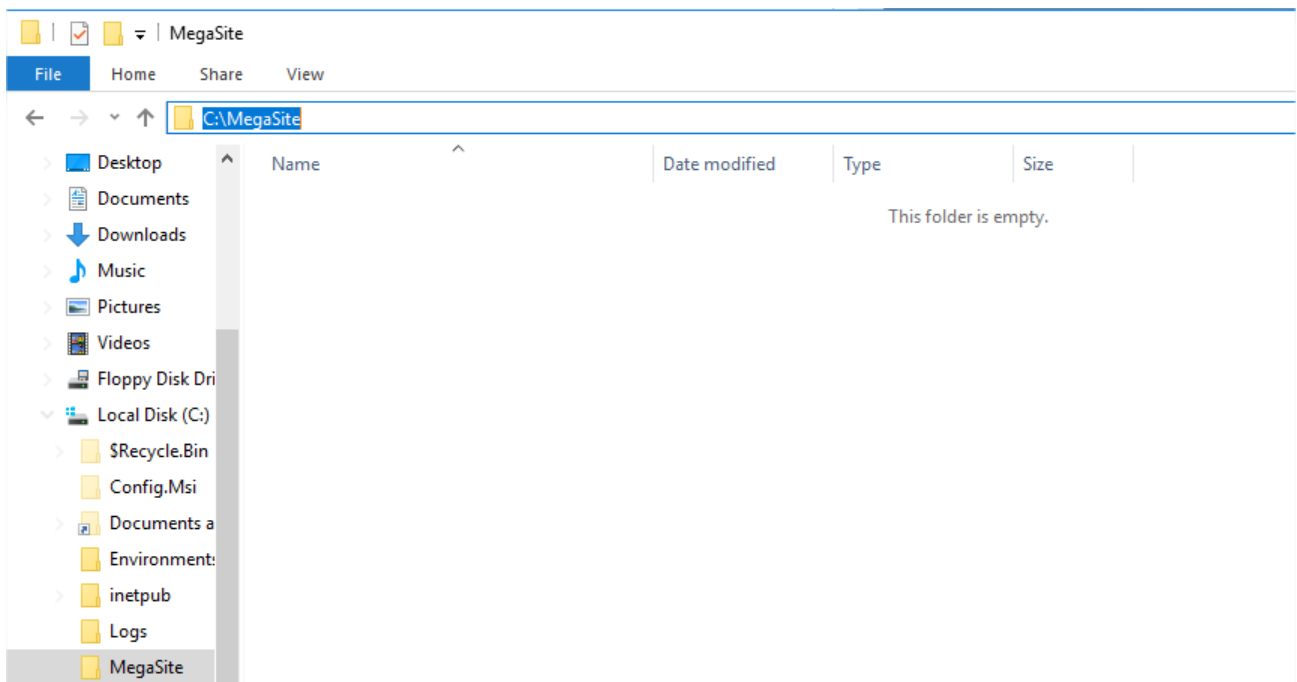


Then, **on the RDBMS server**, the folders that we need to create are those that will be transformed to shared folders,. The three folders are:

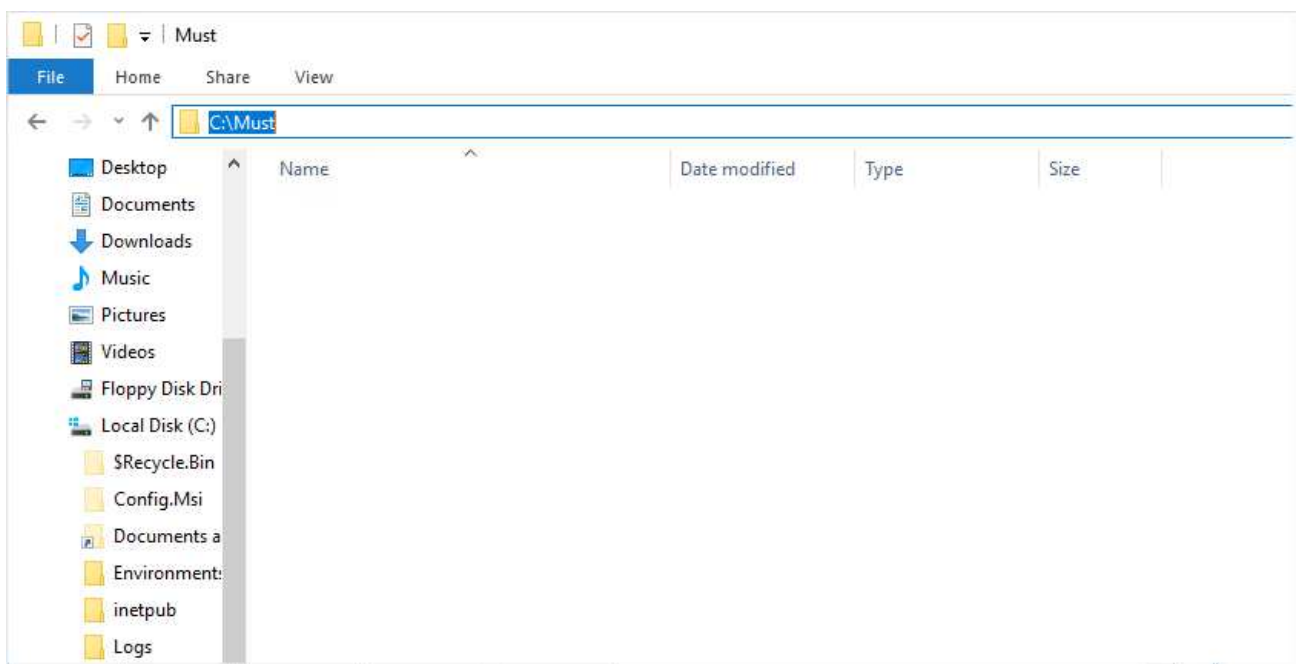
"C:\Environments" for the environments folders:



"C:\MegaSite" to host the clustered configuration file:



And “C:\Must” that will host the license file:

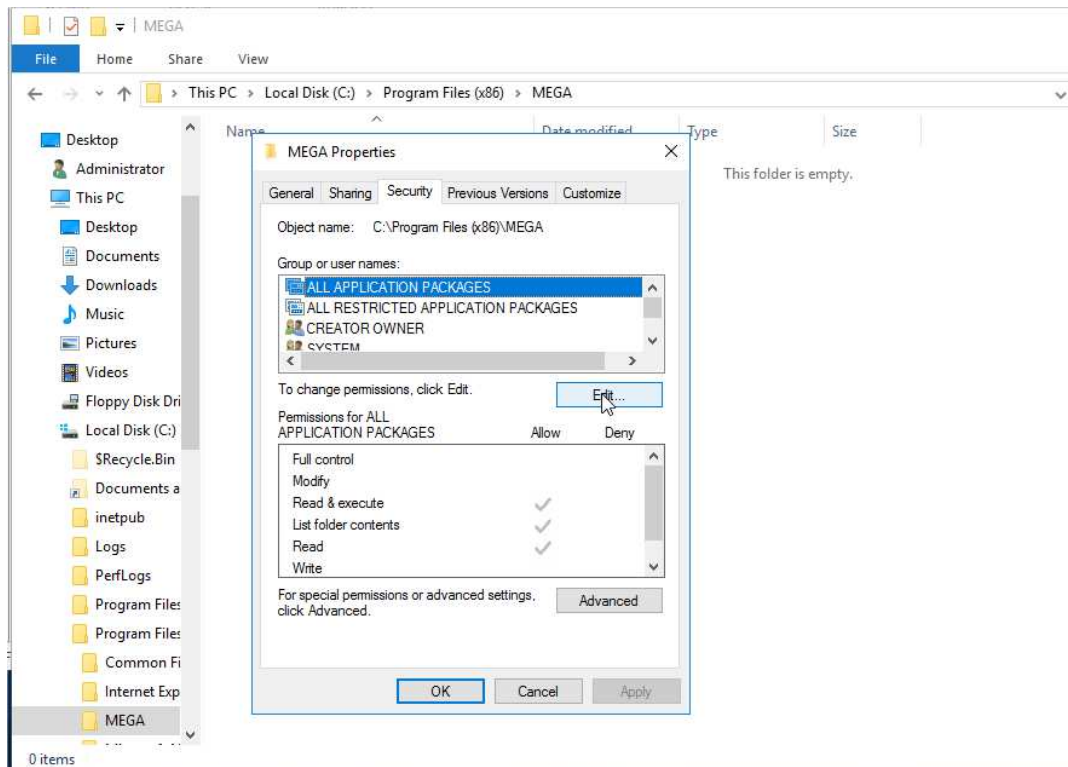


Give rights on the proper folder

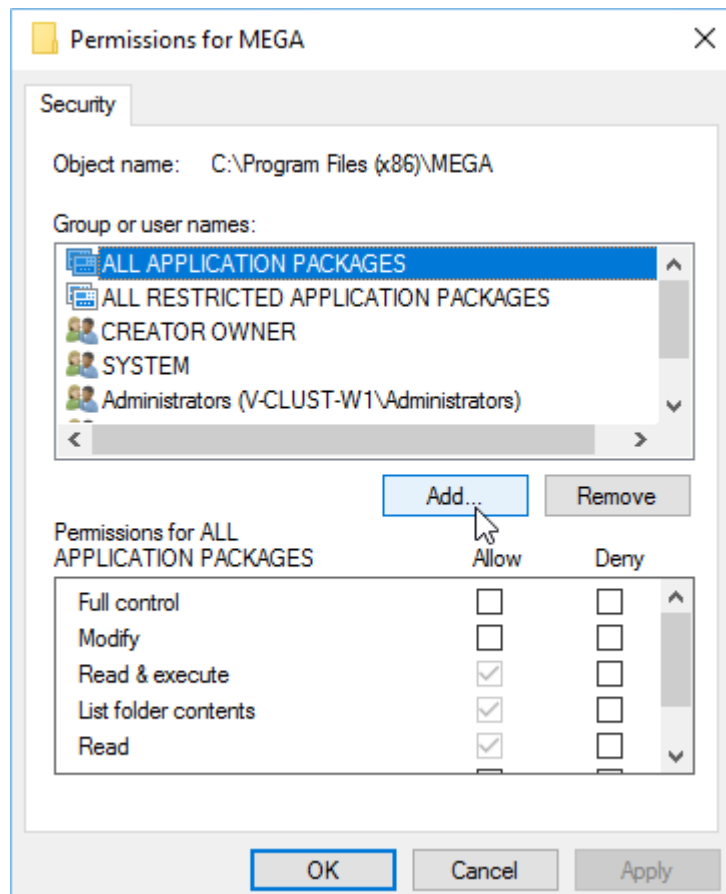
Web/Application Servers

The local "IIS_IUSRS" group needs to have read/write rights on the older that will host the Hopex website and the Mega binaries.

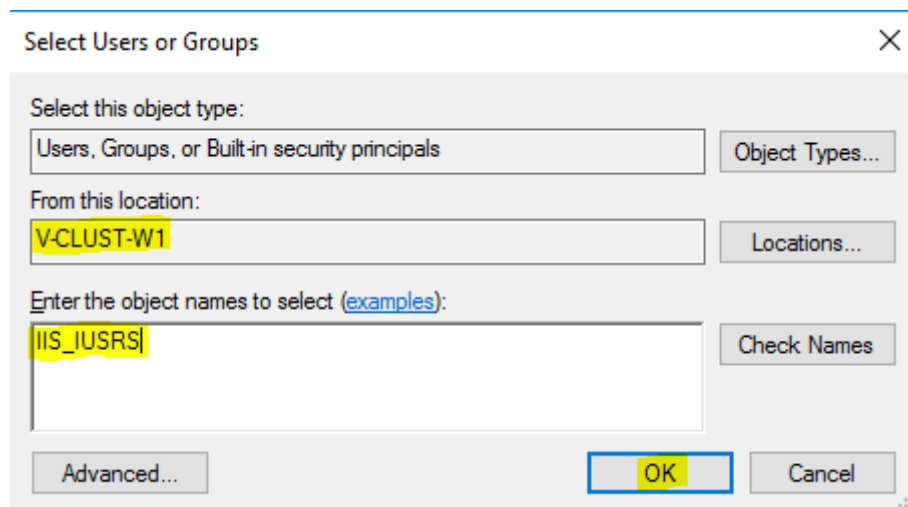
We start with the "C:\Program Files (x86)\MEGA" folder:



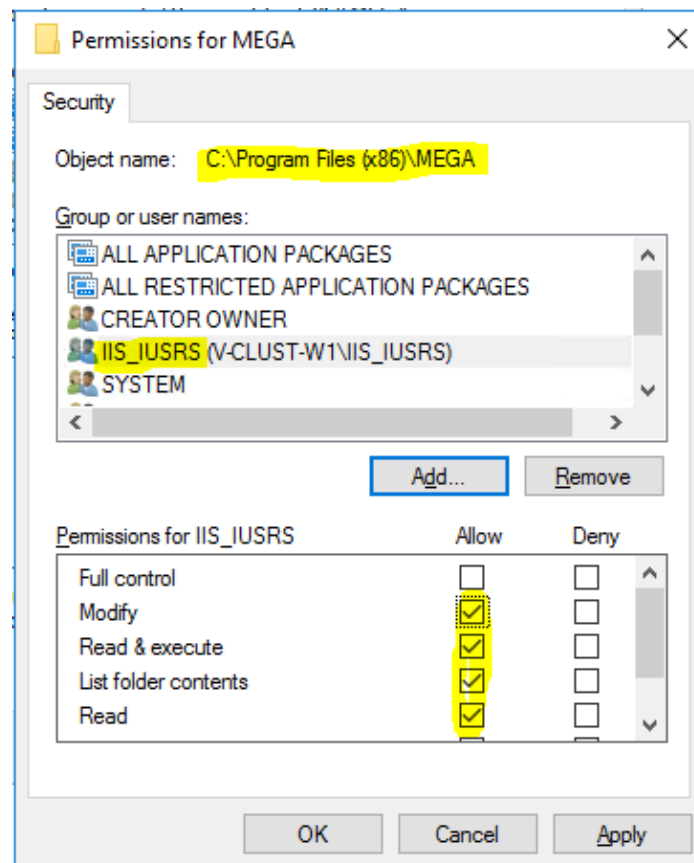
Click „Add“:



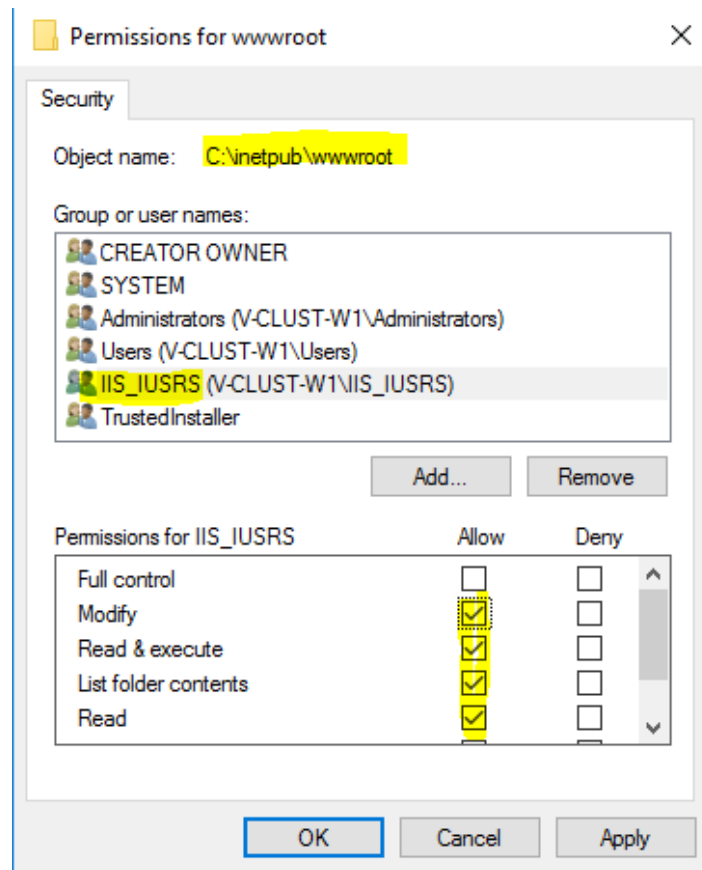
Add the local "IIS_IUSRS" group:



With read and modify rights:



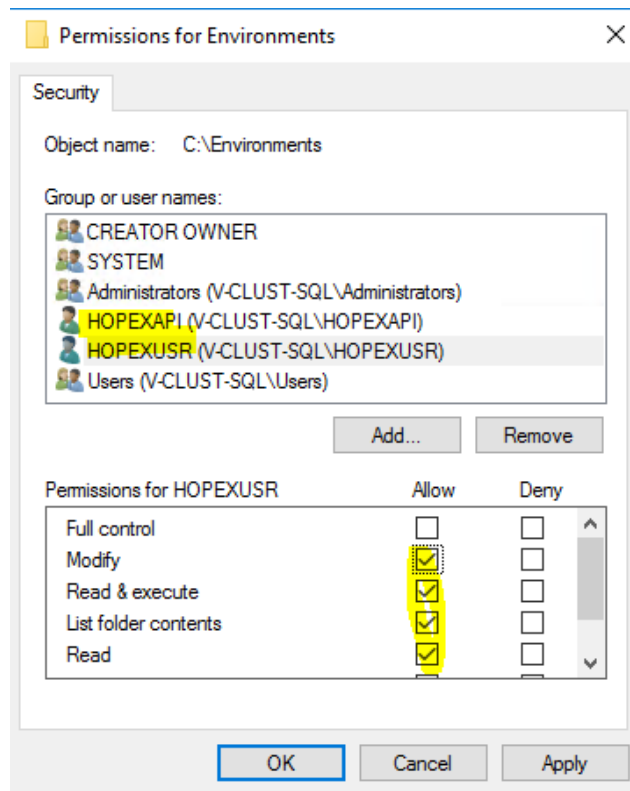
Same thing on "C:\inetpub\wwwroot". By default, the group already has read access on the folder. We add the "Modify" right:



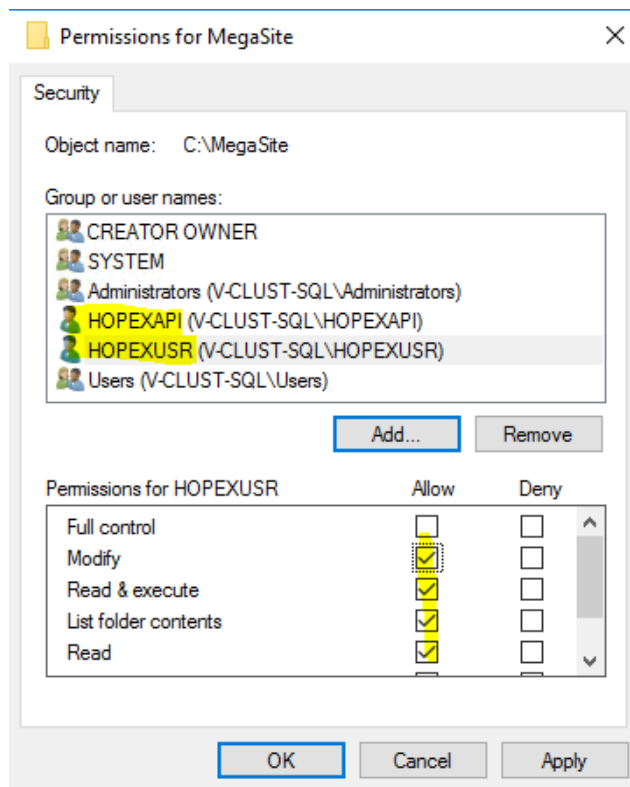
RDBMS Server

We need to grant the local users HOPEXUSR and HOPEXAPI read/write rights on three folders.

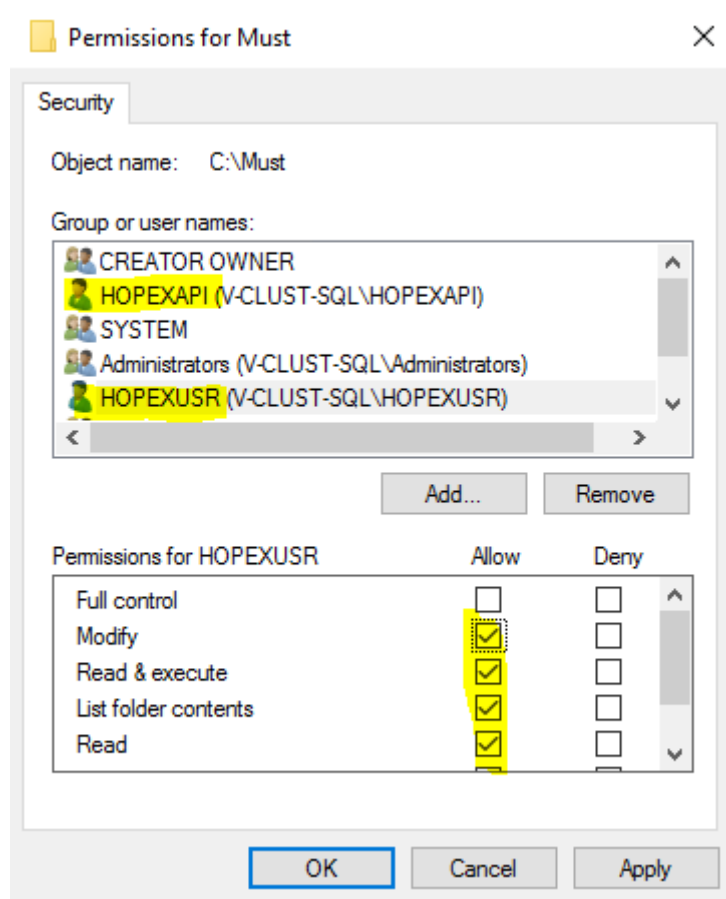
"C:\Environments":



"C:\MegaSite":

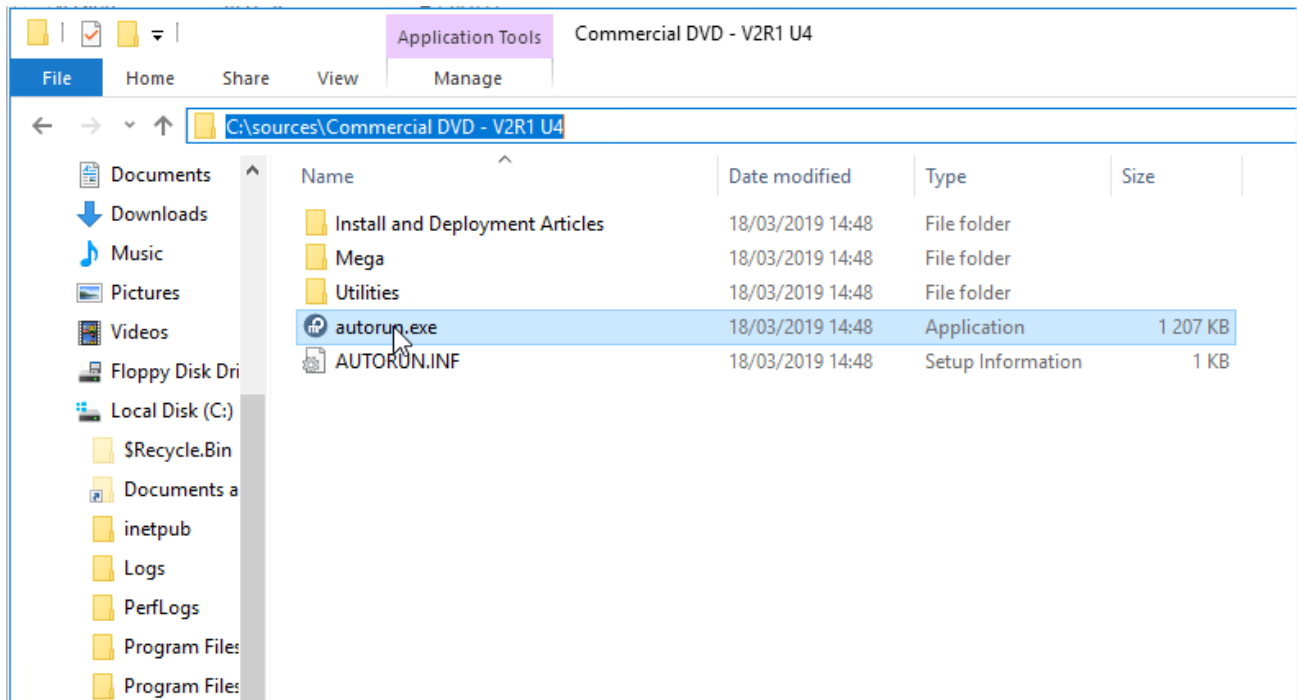


And "C:\Must":

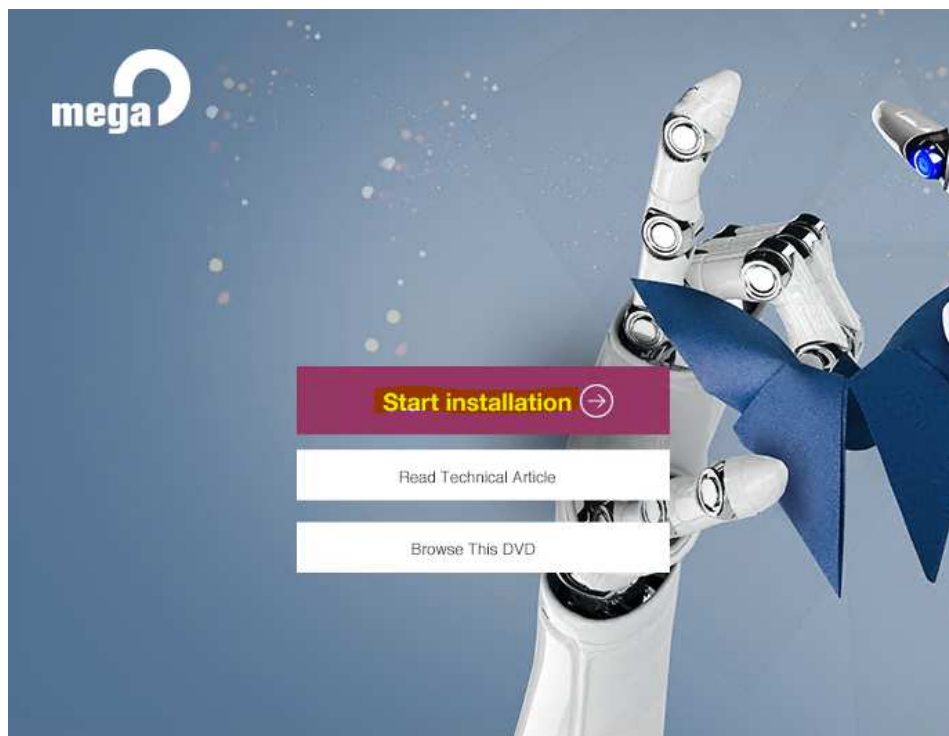


INSTALL MEGA ON THE WEB SERVERS

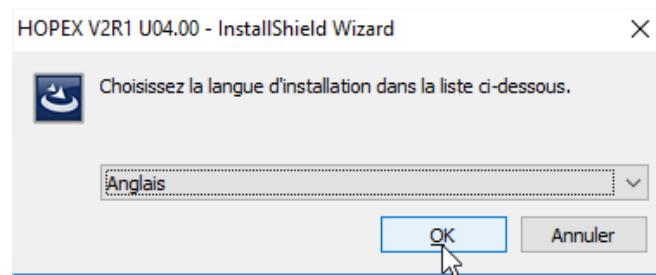
Launch the install (autorun.exe file in the install folder of Mega Hopex, in "C:\sources\Commercial DVD - V2R1 U4"):



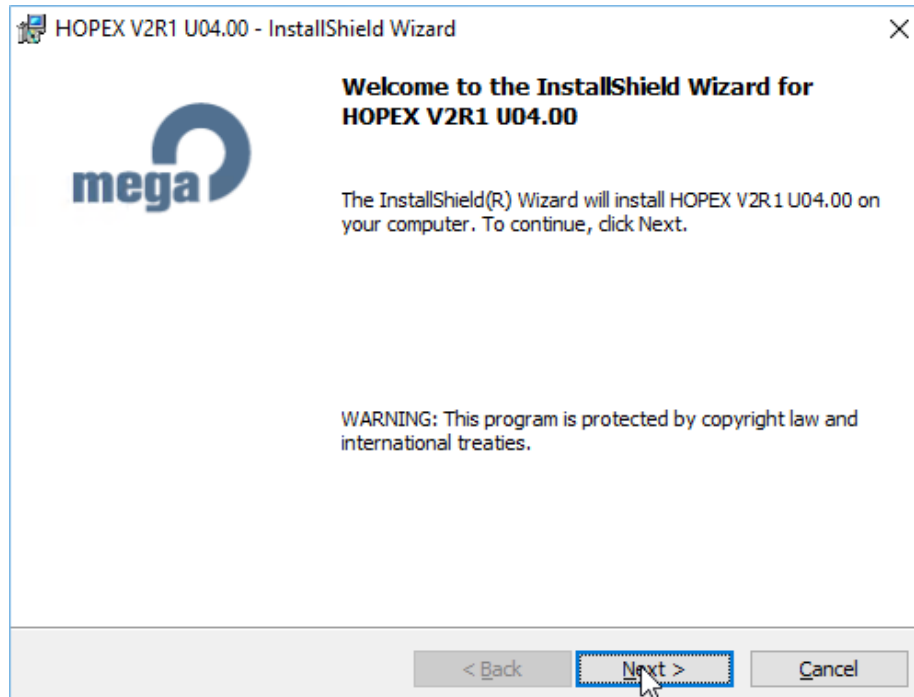
Click « Start installation »:



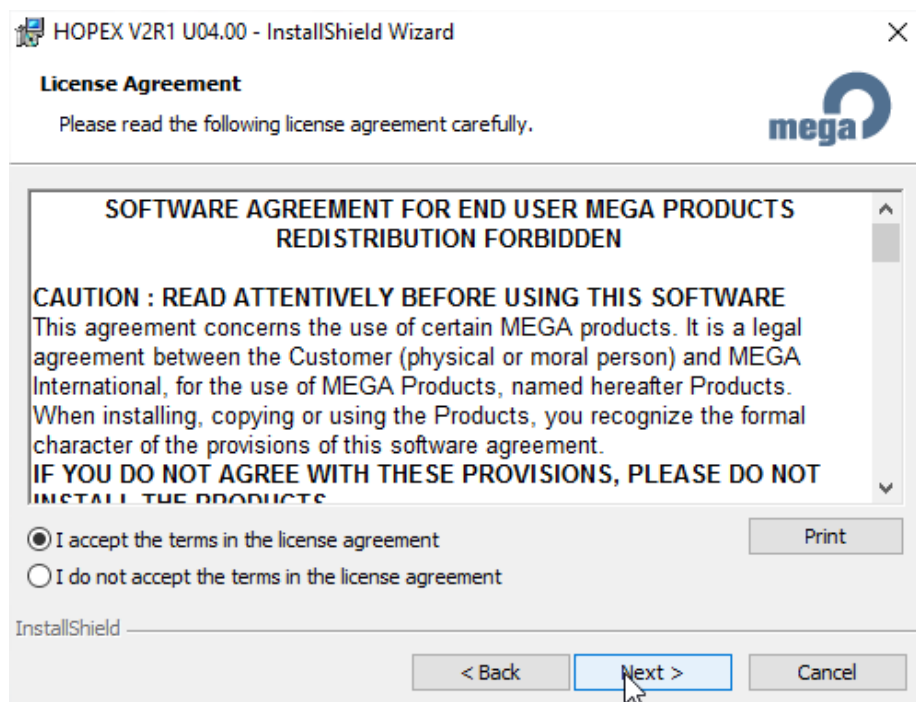
Choose "Anglais" (for "English", because the virtual machine was installed with french regional settings):



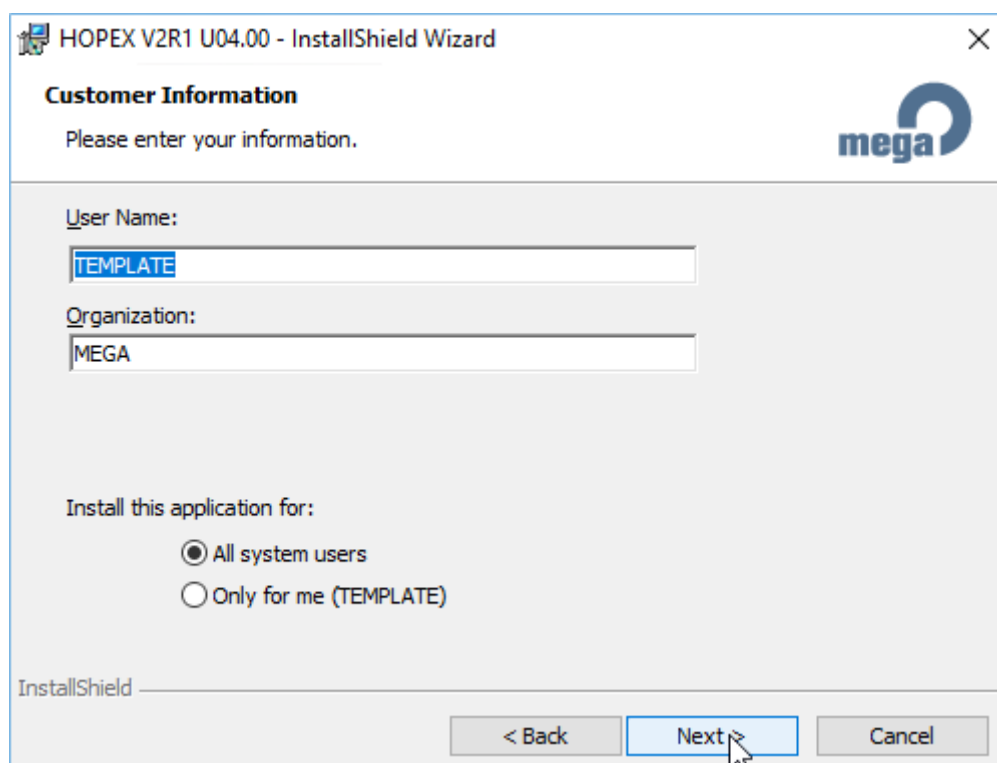
Click « Next » :



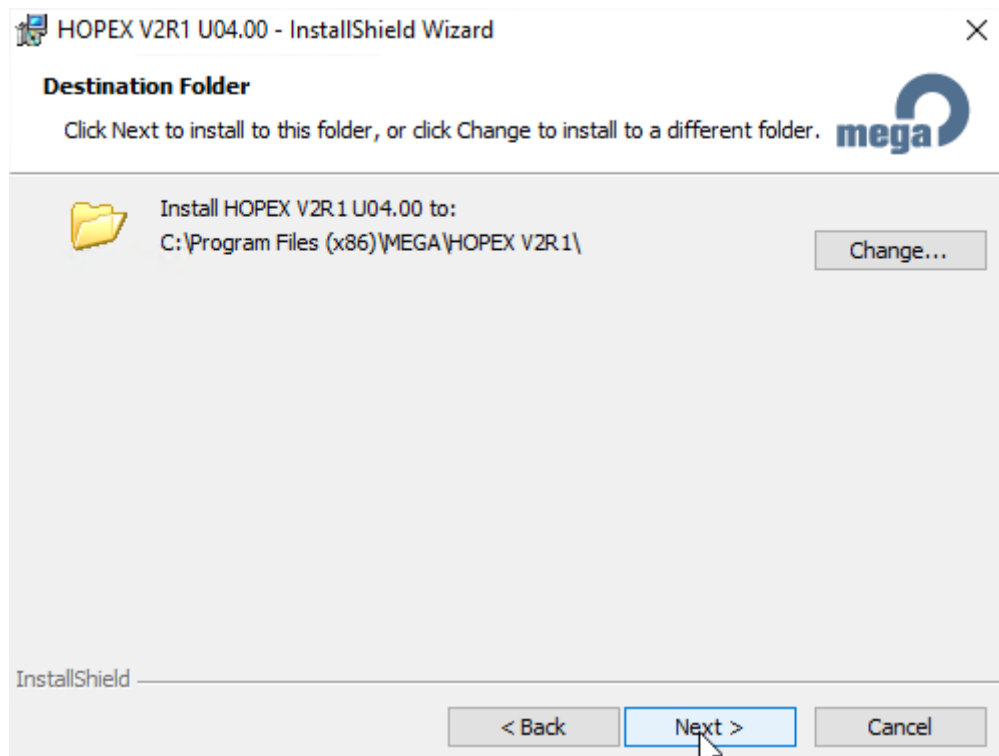
Accept, and again click « Next »:



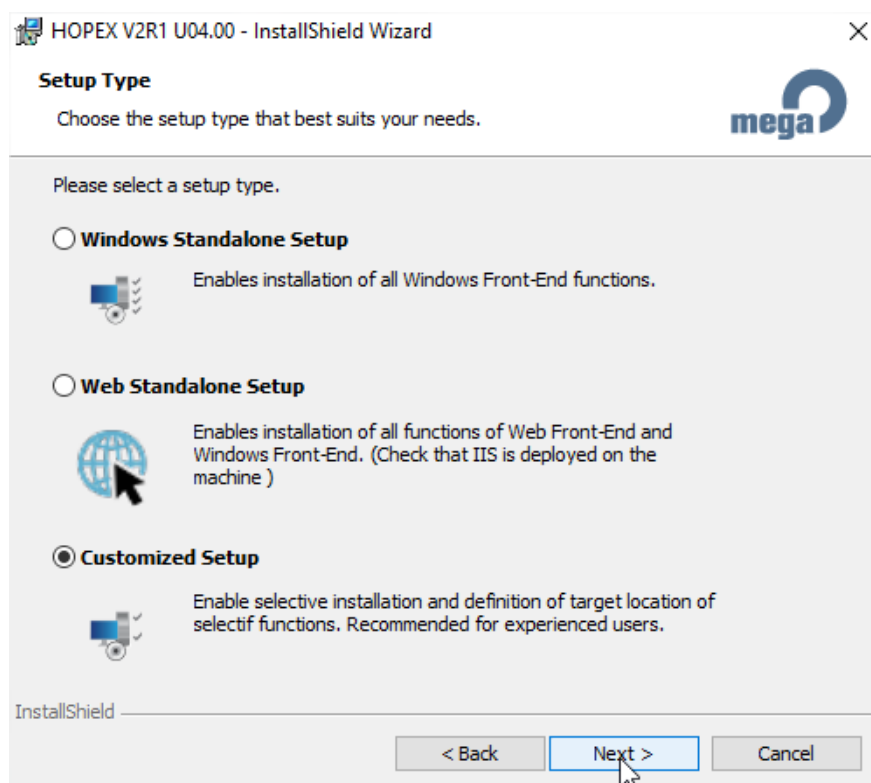
Click « Next » :



By default, the binaries of Mega are installed on the C drive in the below location. We keep that setting and click “Next”:



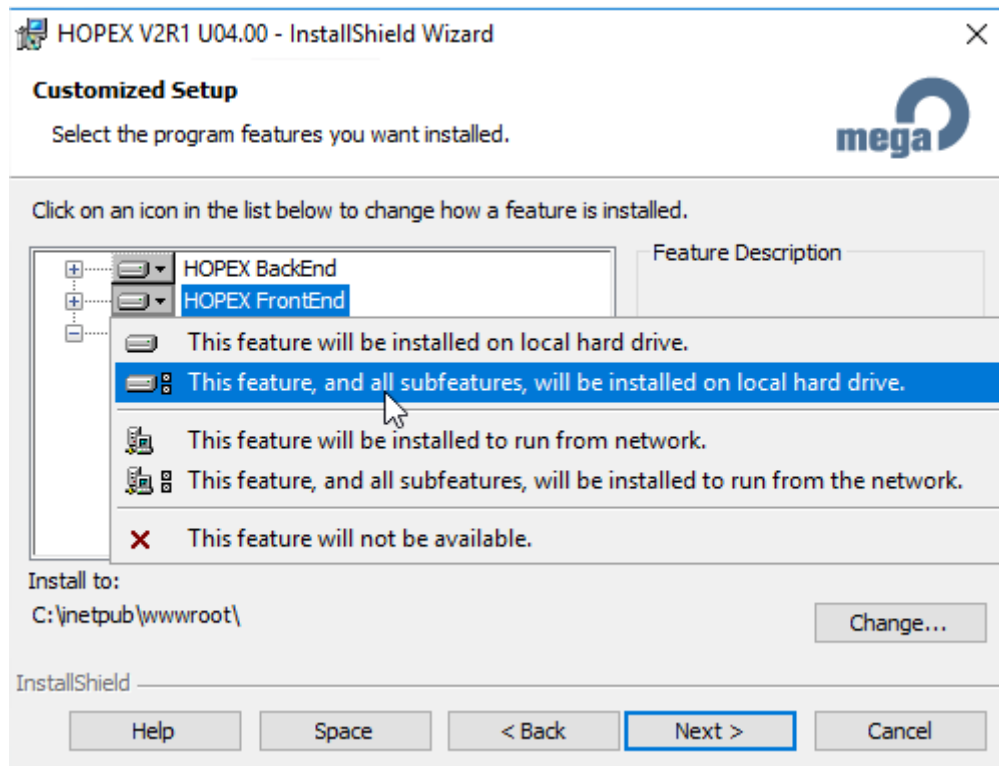
Choose “Customized setup”:



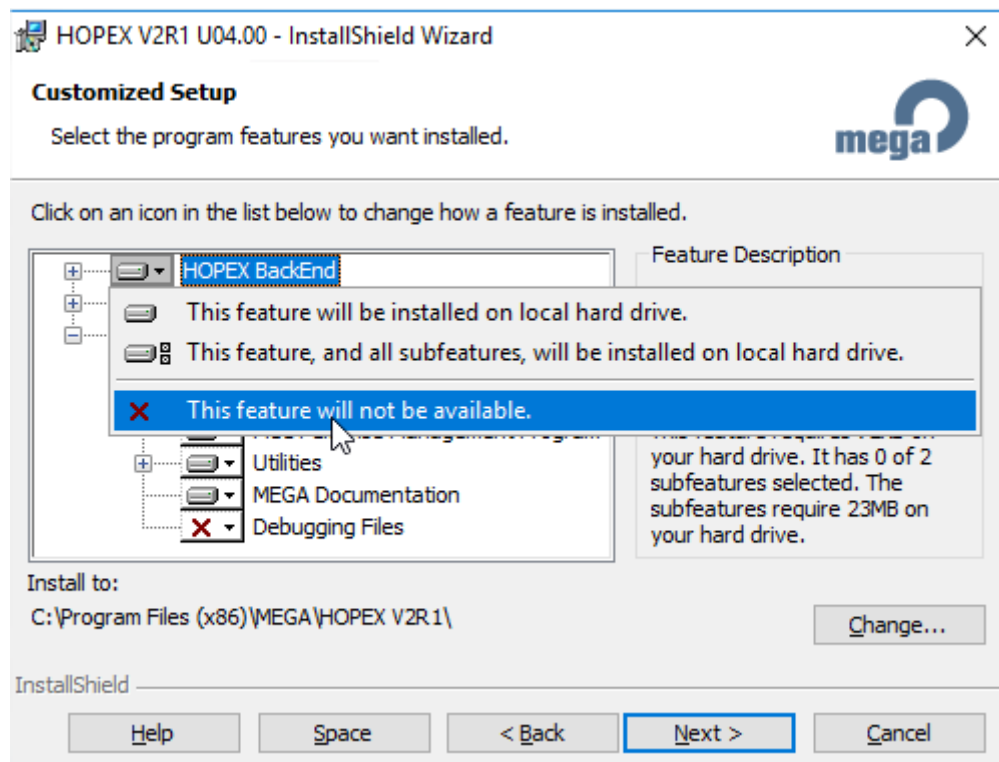
In the list of products to install, we will have:

- Everything in “Hopex FrontEnd”.
- To disable the “Software Mega”.

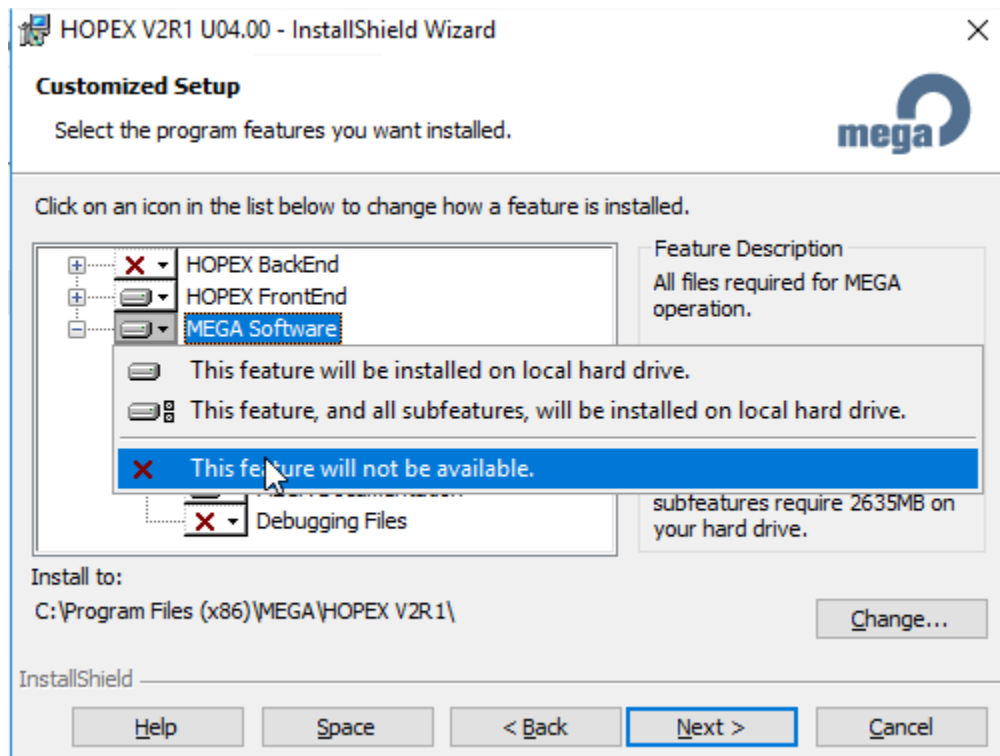
First, we activate features and subfeatures for “Hopex FrontEnd”:



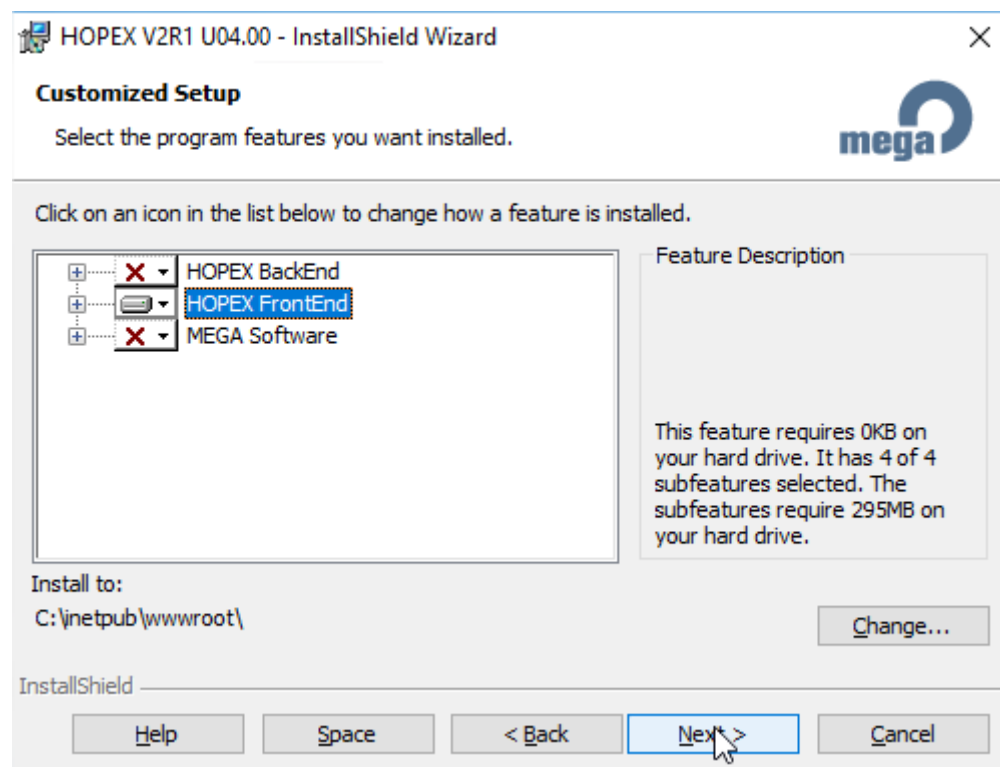
Then, we disable “Hopex BackEnd”:



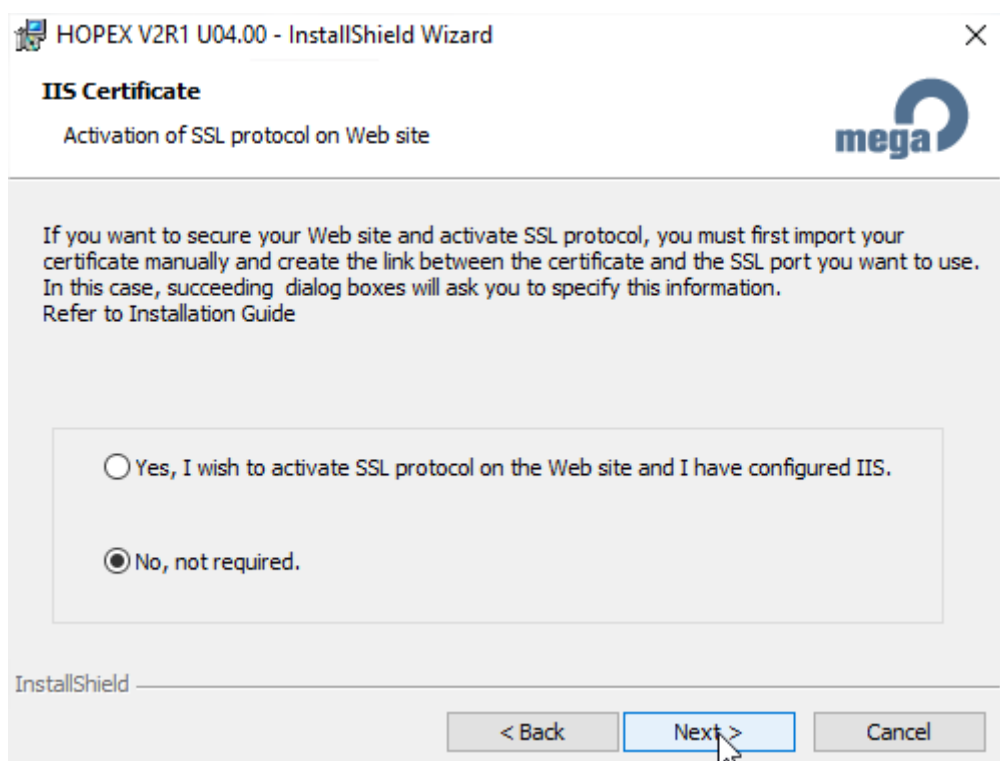
As well as “Software Mega”:



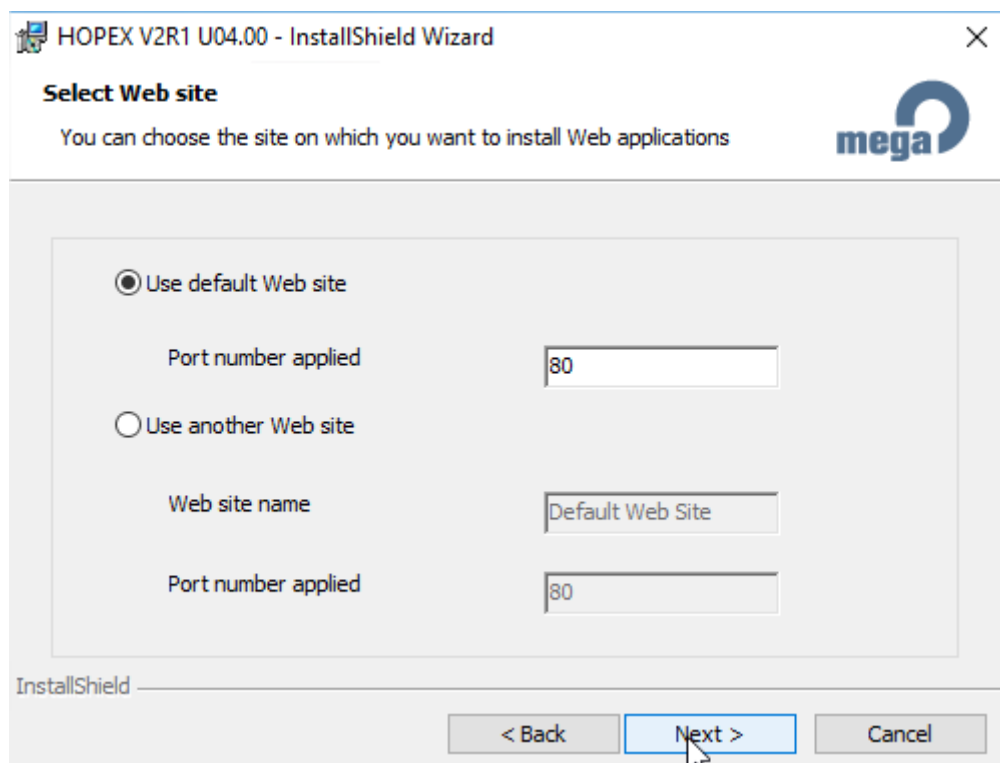
When we have this list, we click “Next”:



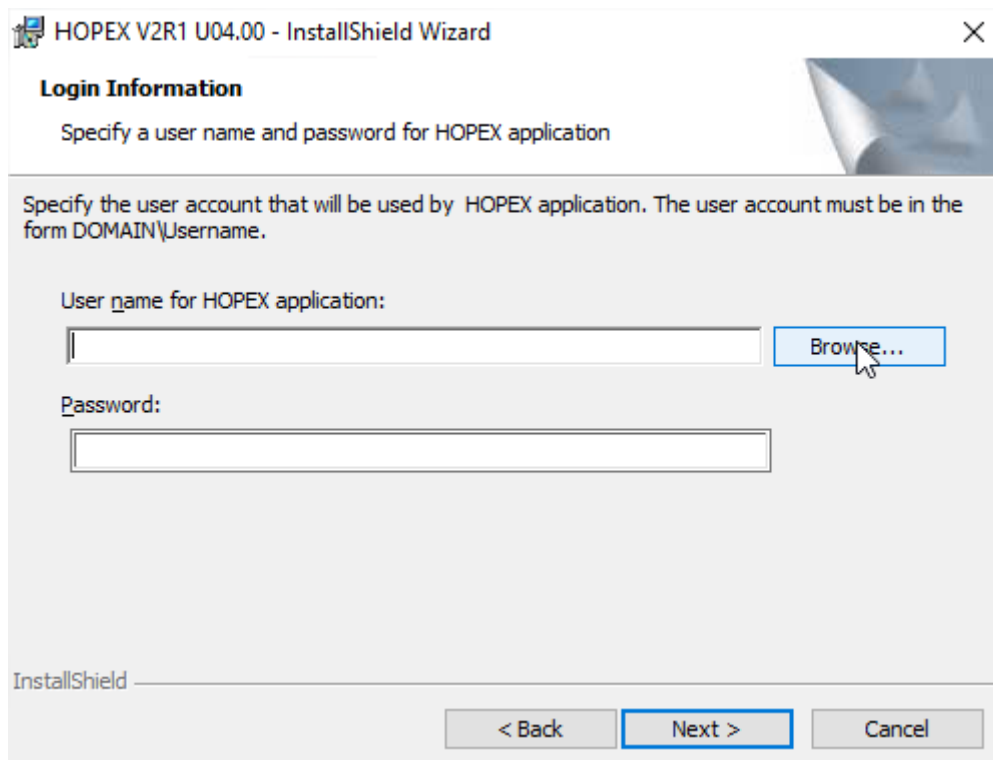
The SSL is not activated directly on the web/applications servers, so choose “No...”:



We use the default website on port 80:



We click “Browse” to search for the first impersonate user:



HOPEX V2R1 U04.00 - InstallShield Wizard

Login Information

Specify a user name and password for HOPEX application

Specify the user account that will be used by HOPEX application. The user account must be in the form DOMAIN\Username.

User name for HOPEX application:

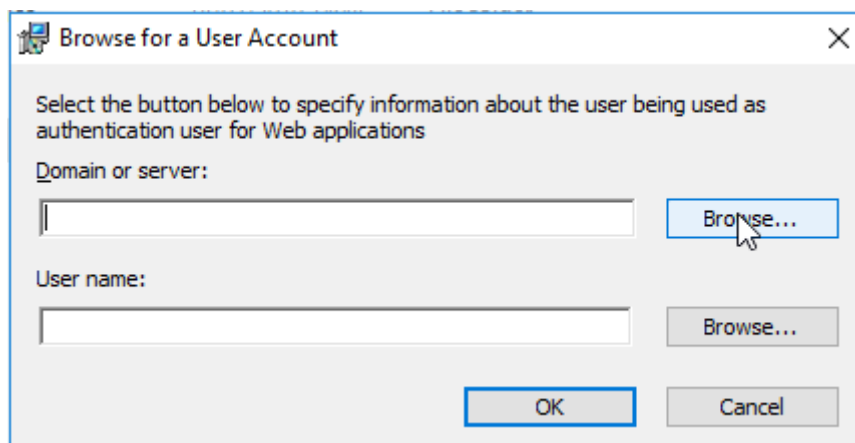
Browse...

Password:

InstallShield

< Back Next > Cancel

Again on "Browse":



Browse for a User Account

Select the button below to specify information about the user being used as authentication user for Web applications

Domain or server:

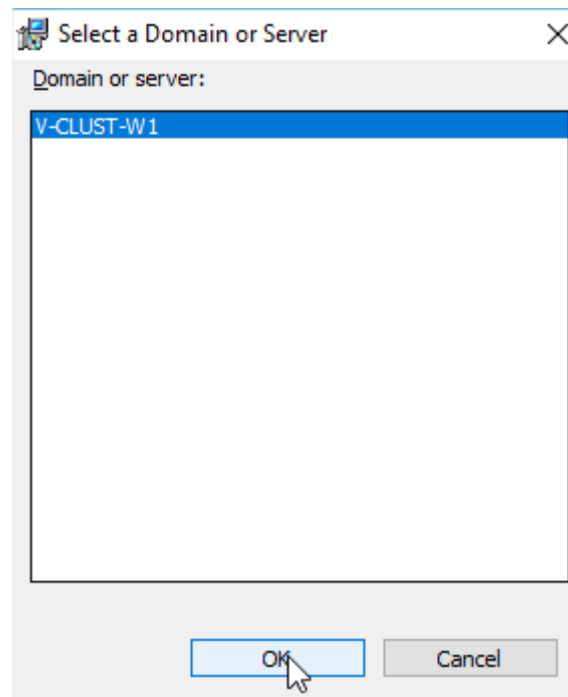
Browse...

User name:

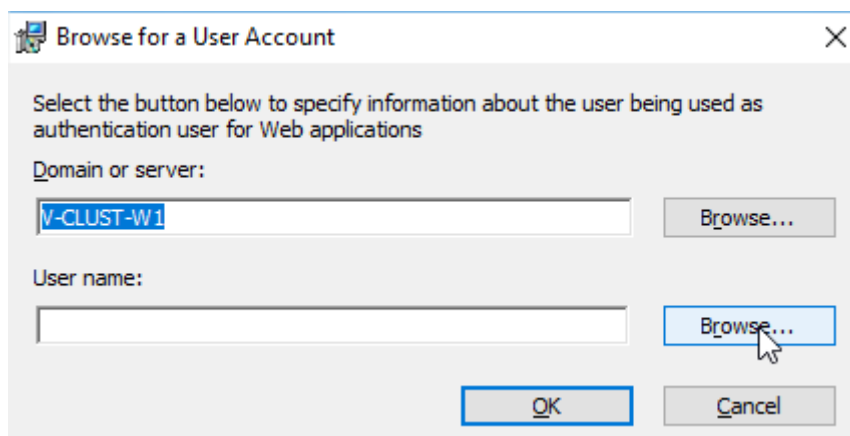
Browse...

OK Cancel

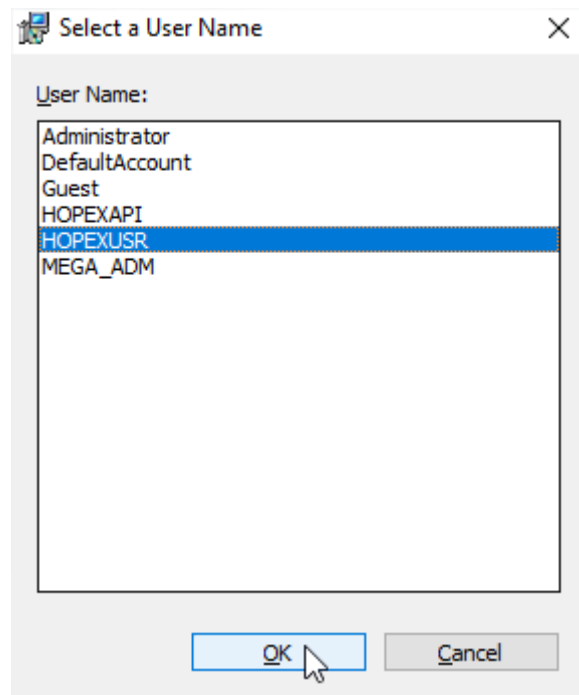
Select the local server:



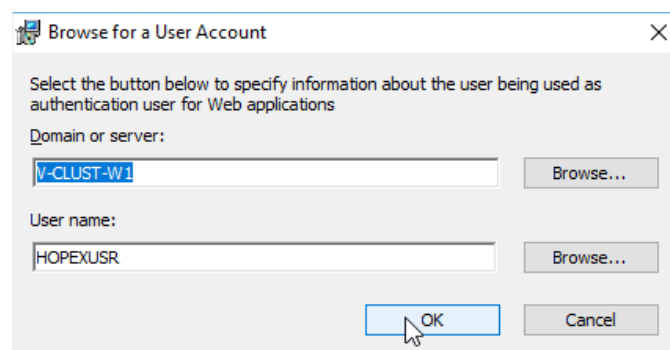
Then we click the second "Browse":



And we select the "HOPEXUSR" account:



Click « OK » :



We provide the password and click “Next”:

HOPEX V2R1 U04.00 - InstallShield Wizard

Login Information
Specify a user name and password for HOPEX application

Specify the user account that will be used by HOPEX application. The user account must be in the form DOMAIN\Username.

User name for HOPEX application:
V-CLUST-W1\HOPEXUSR Browse...

Password:
[Masked Password]

InstallShield

< Back **Next >** Cancel

Proceed with the same steps with the “HOPEXAPI” account for the “HOPEXAPI” application:

HOPEX V2R1 U04.00 - InstallShield Wizard

Login Information
Specify a user name and password for HOPEXAPI application

Specify the user account that will be used by HOPEXAPI application. The user account must be in the form DOMAIN\Username.

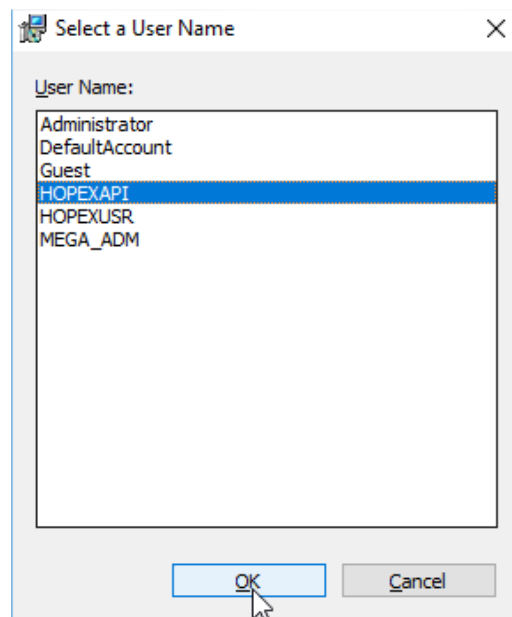
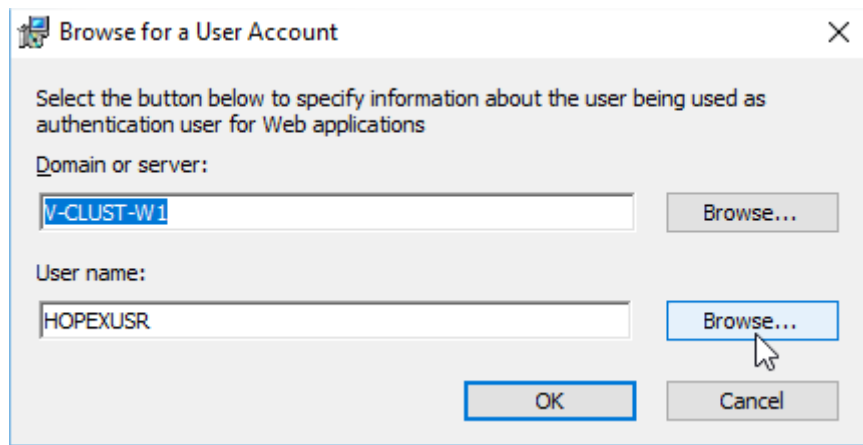
User name for HOPEX API application:
[Empty Text Box] Browse...

Password:
[Empty Password Field]

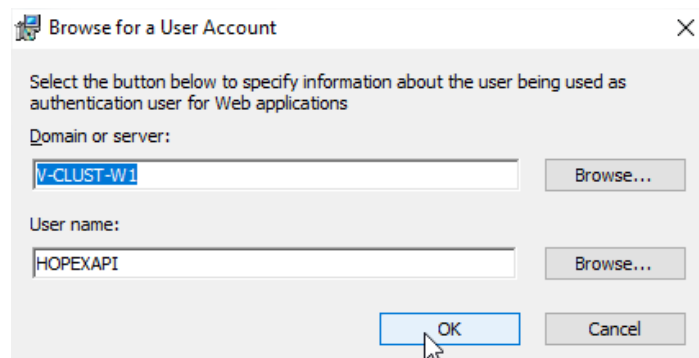
InstallShield

< Back **Next >** Cancel

It remembers previous settings, so click the second “Browse” button, to switch:



Click "OK":

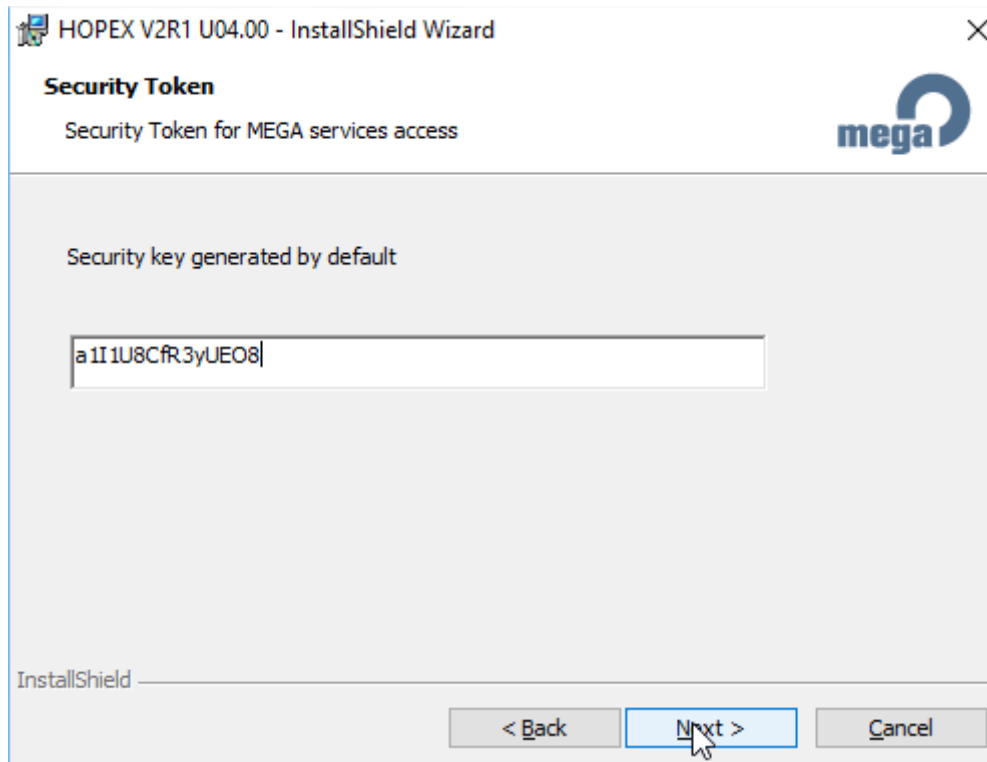


Provide the password of the user and click "Next":

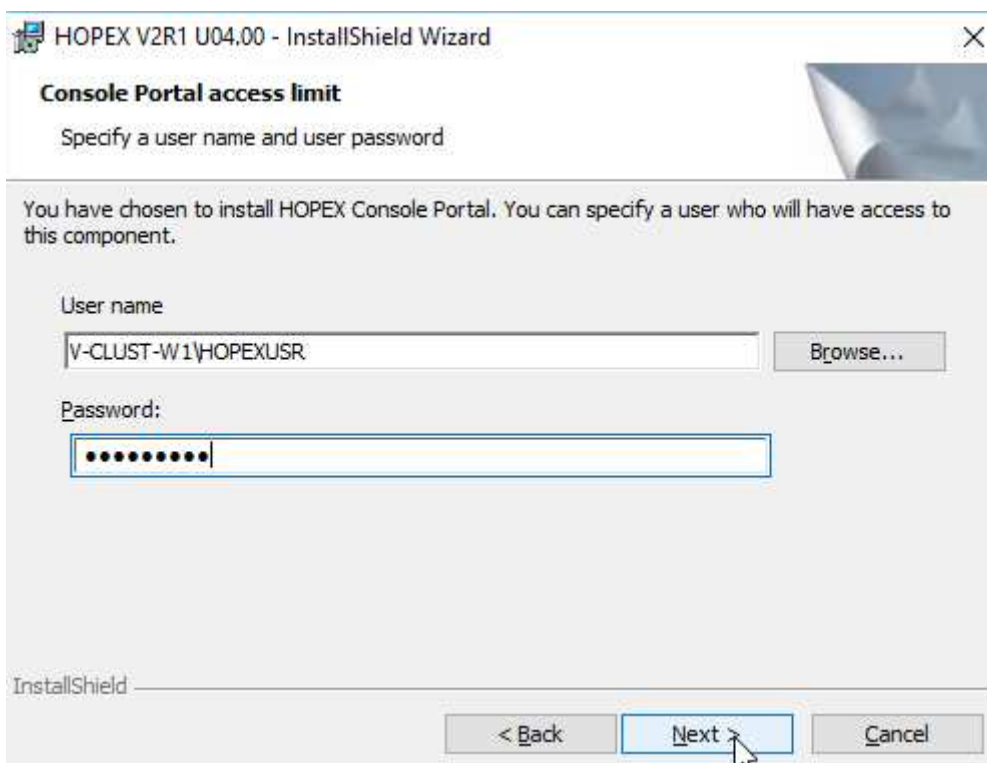
For the SSP Url, we use the IP address of the Load Balancer, so in this case “137.74.87.169”:

Security key (for reference, and because we need to use the same on all web servers). It is generated randomly. We keep the key that was created for the very first server that we install. On all other servers of the platform, we will need to reuse this key so that they all share the same one, and can work together:

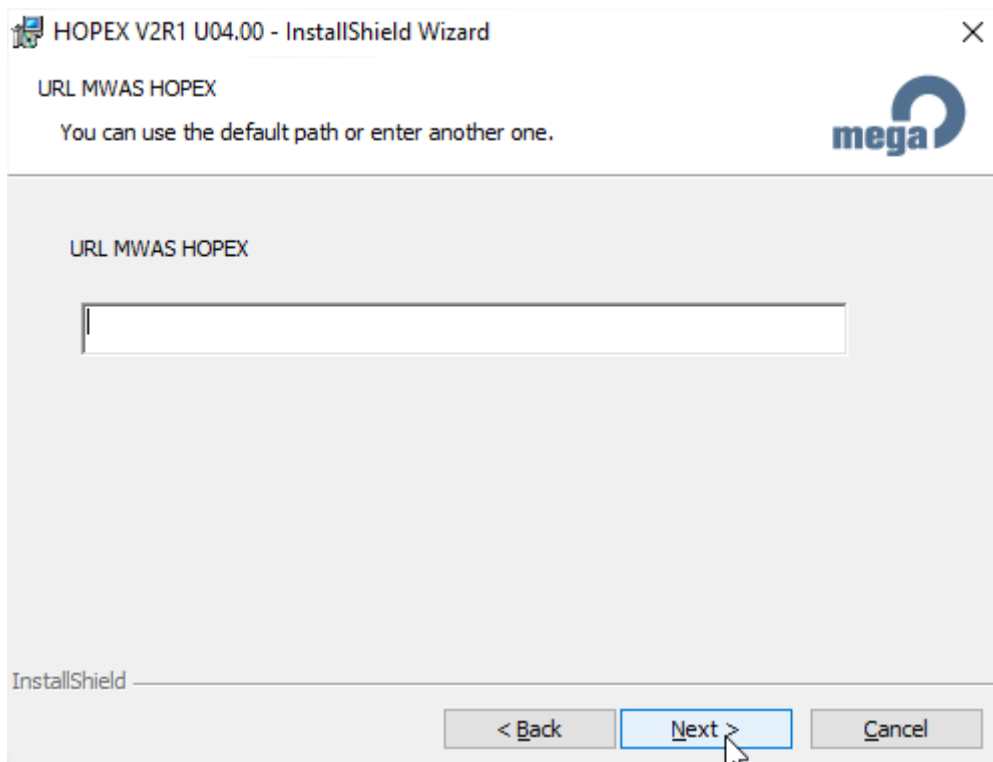
a1I1U8CfR3yUEO8



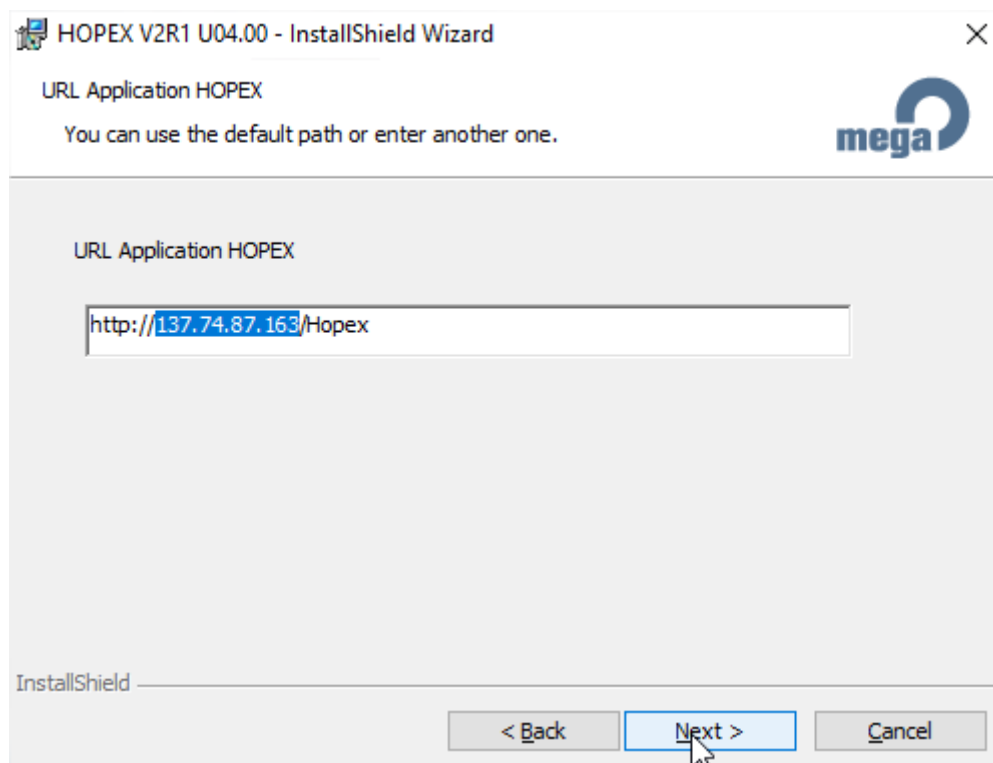
To secure the access to the consoles portal, we use the "HOPEXUSR" local account:



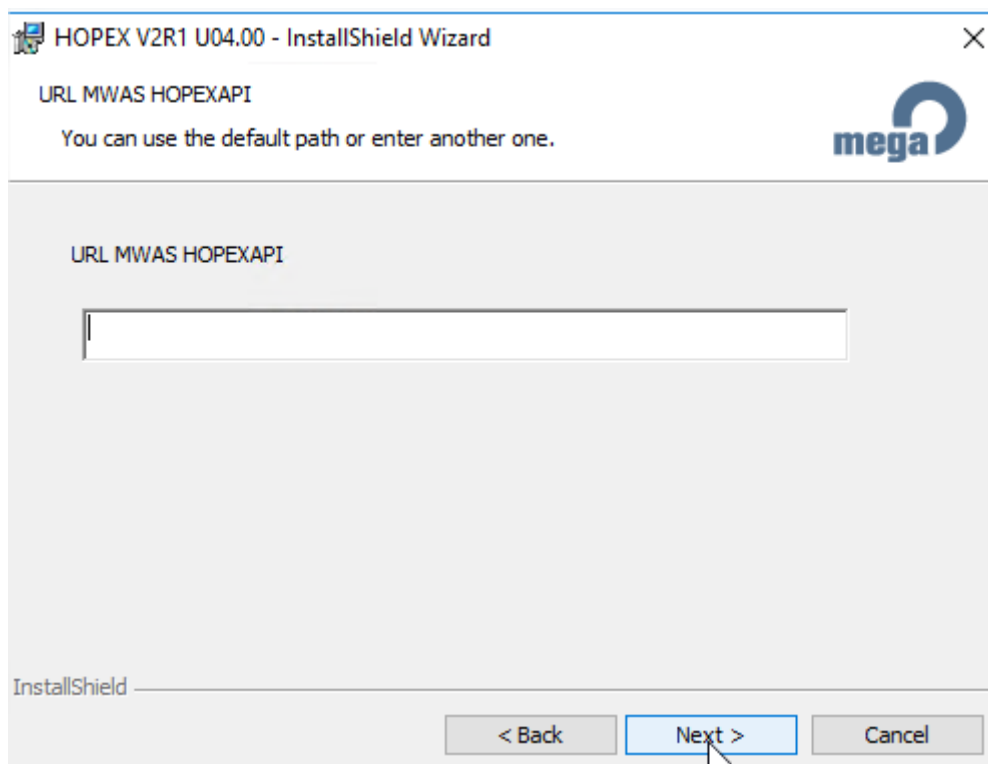
We blank the "MWAS" URL, as it is not hosted on the web servers:



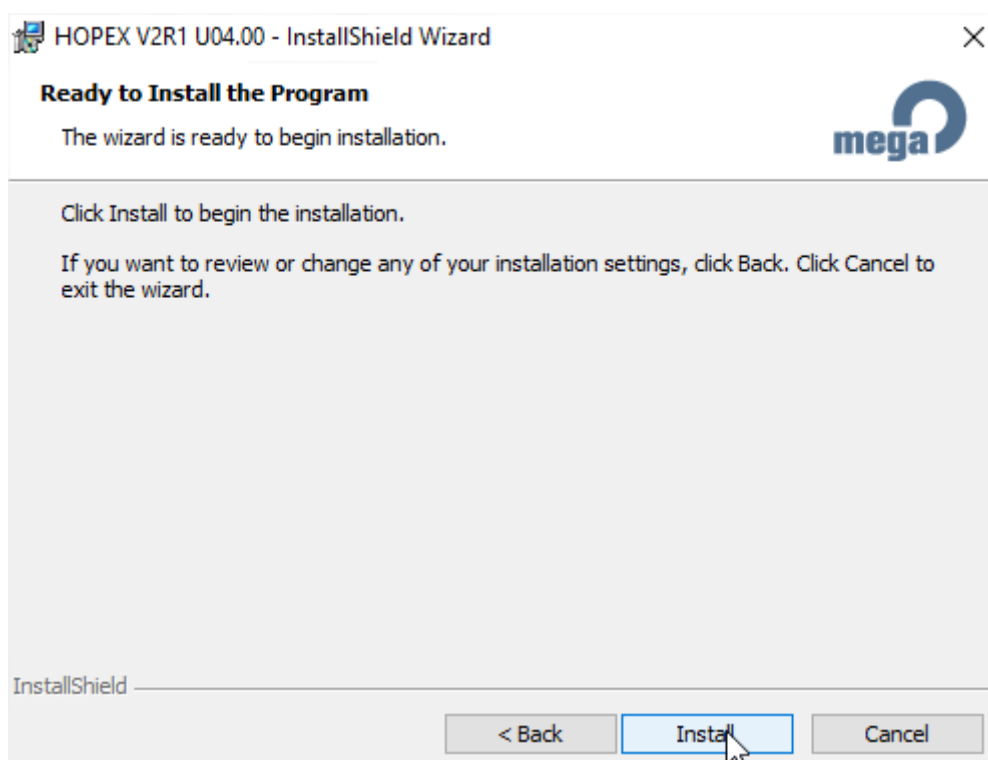
For the "HOPEX" URL, we use the IP address of the web Load Balancer, so here "137.74.87.163":



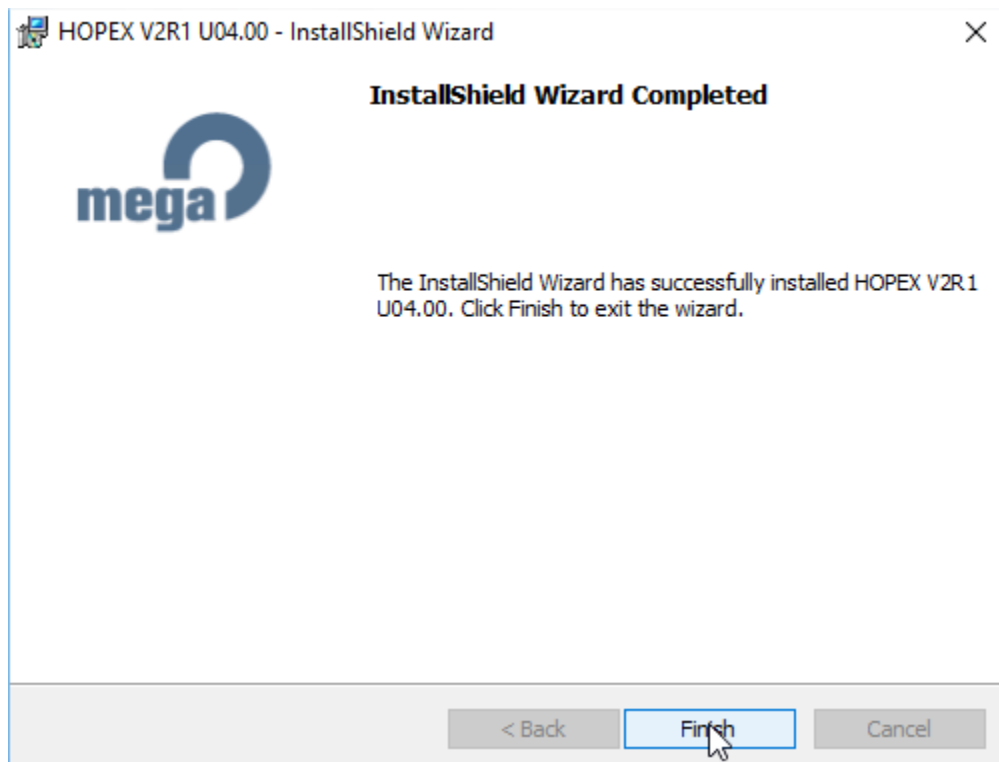
We also blank the MWAS URL of the API part:



We click "Install" to start deploying the tool:

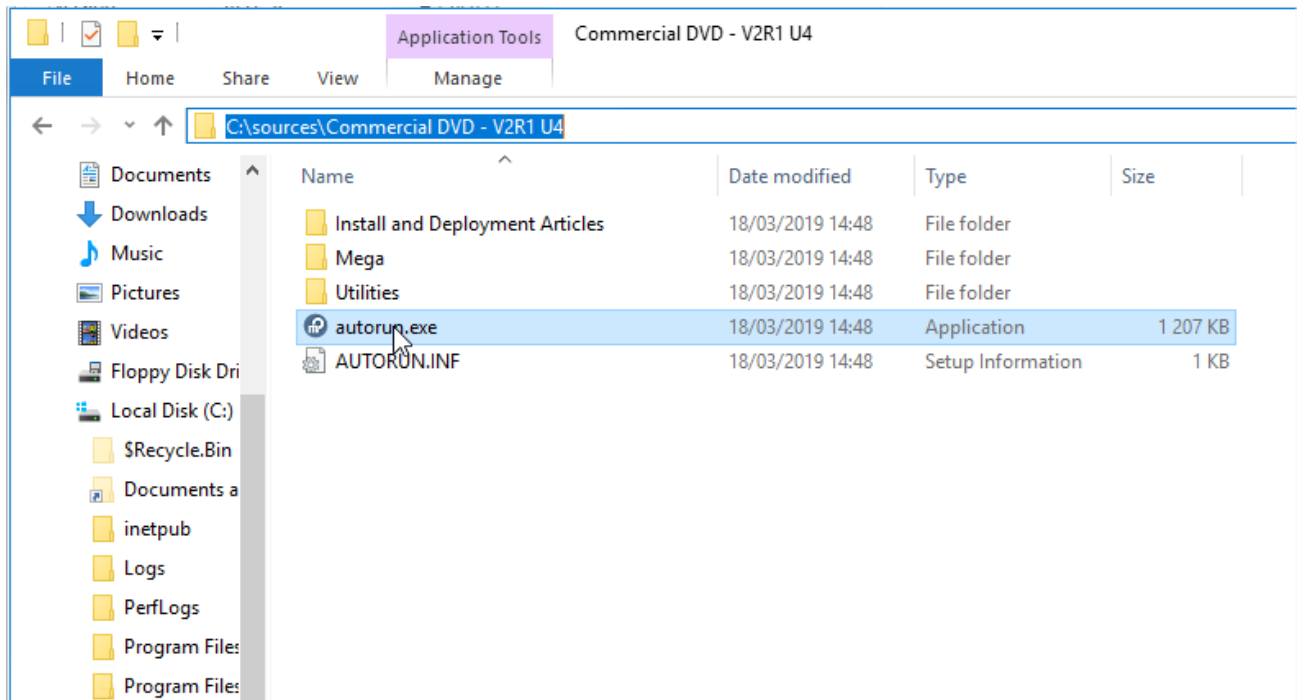


Click « Finish »:

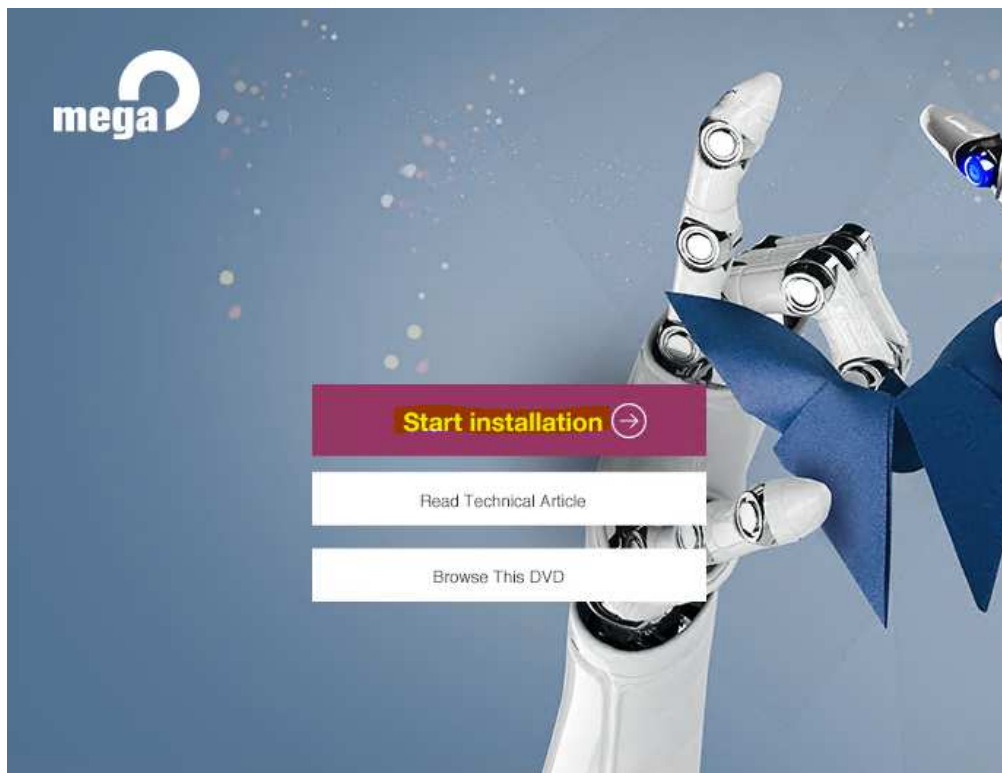


INSTALL MEGA ON THE MWAS APPLICATION SERVERS

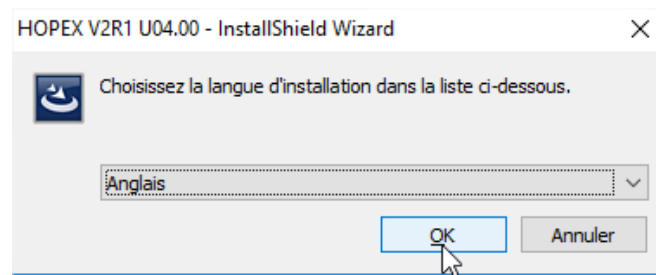
Launch the install (autorun.exe file in the install folder of Mega Hopex, in "C:\sources\Commercial DVD - V2R1 U4"):



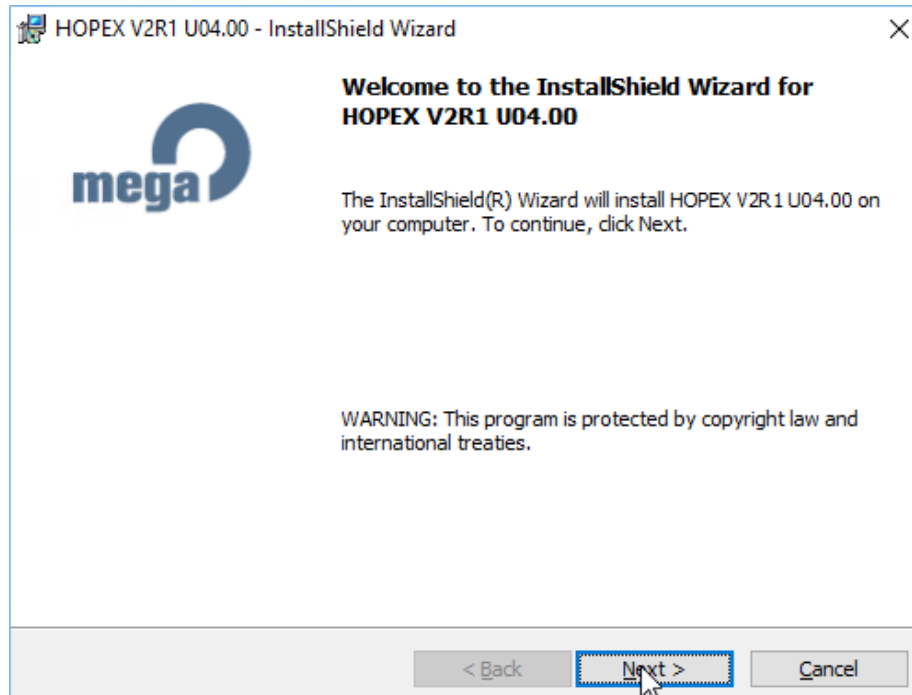
Click « Start installation »:



Choose "Anglais" (for "English", because the virtual machine was installed with french regional settings):



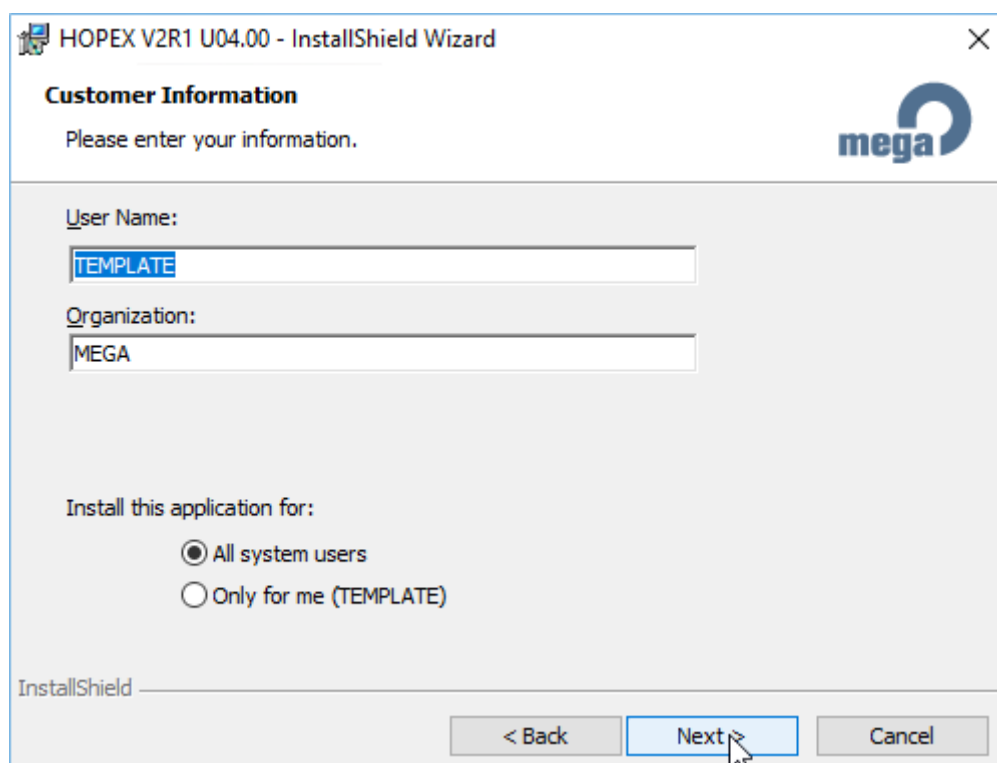
Click « Next » :



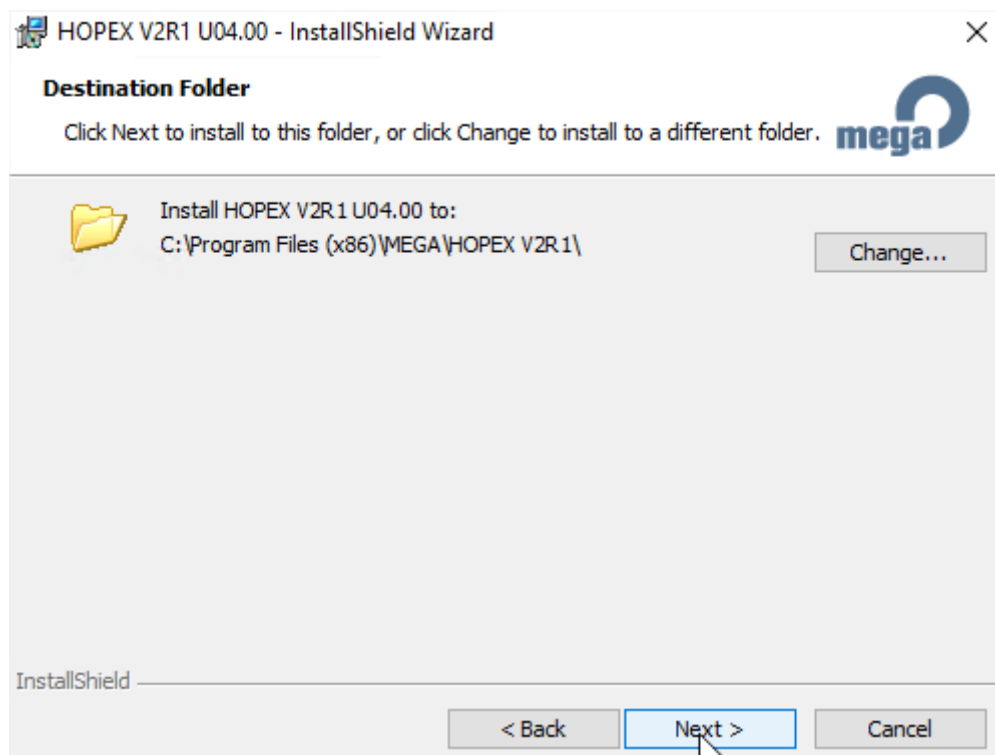
Accept, and again click « Next »:



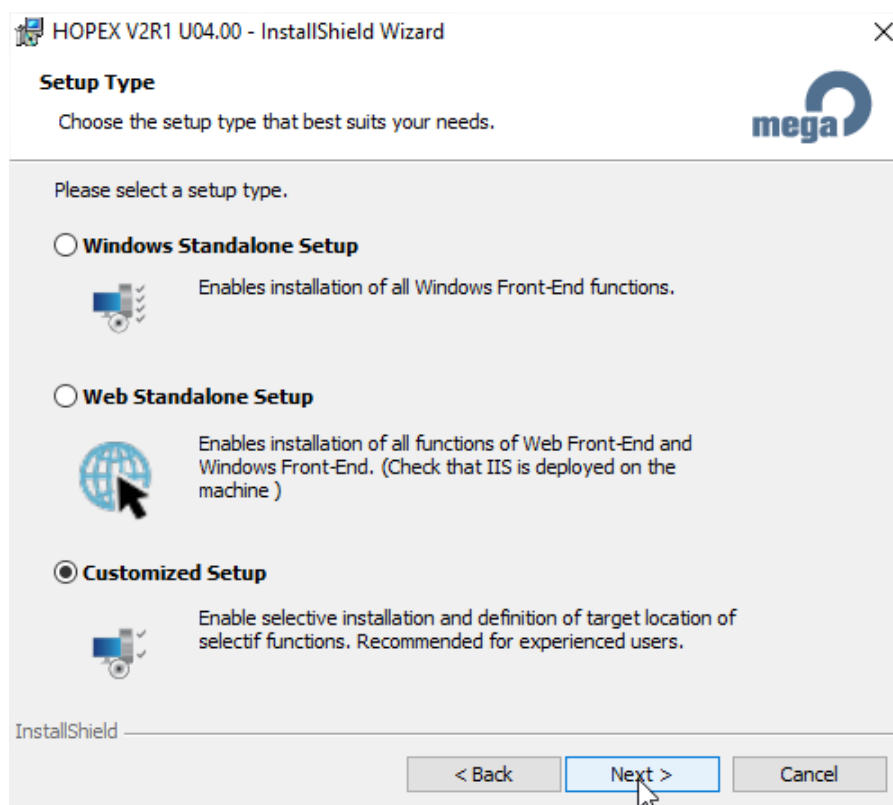
Click « Next » :



By default, the binaries of Mega are installed on the C drive in the below location. We keep that setting and click "Next":



Choose “Customized setup”:

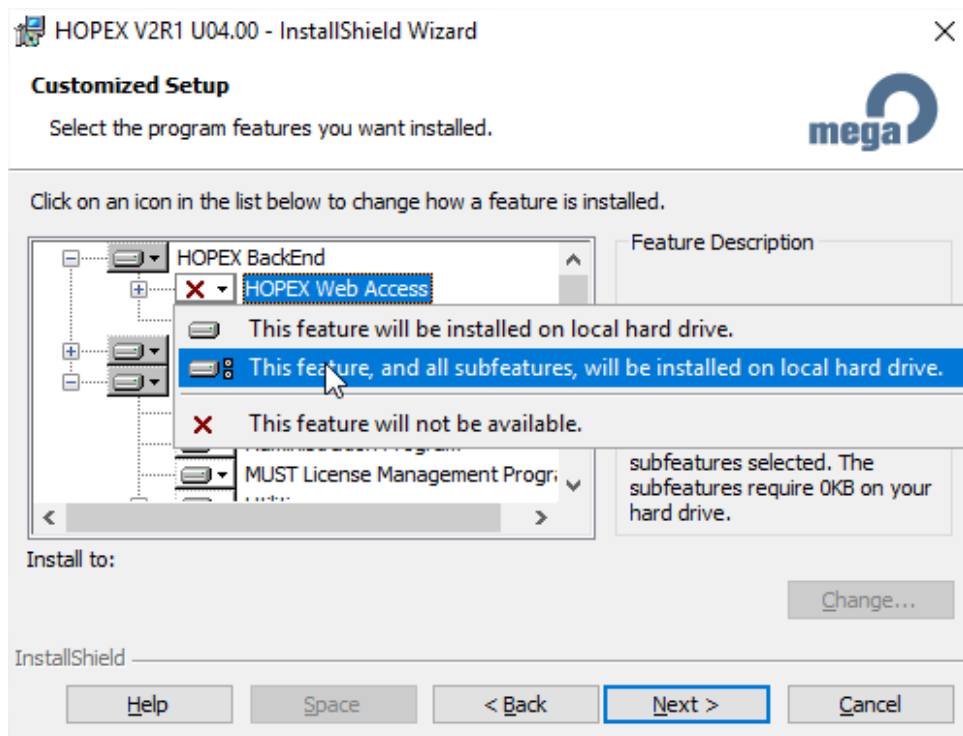


In the list of products to install, we will have:

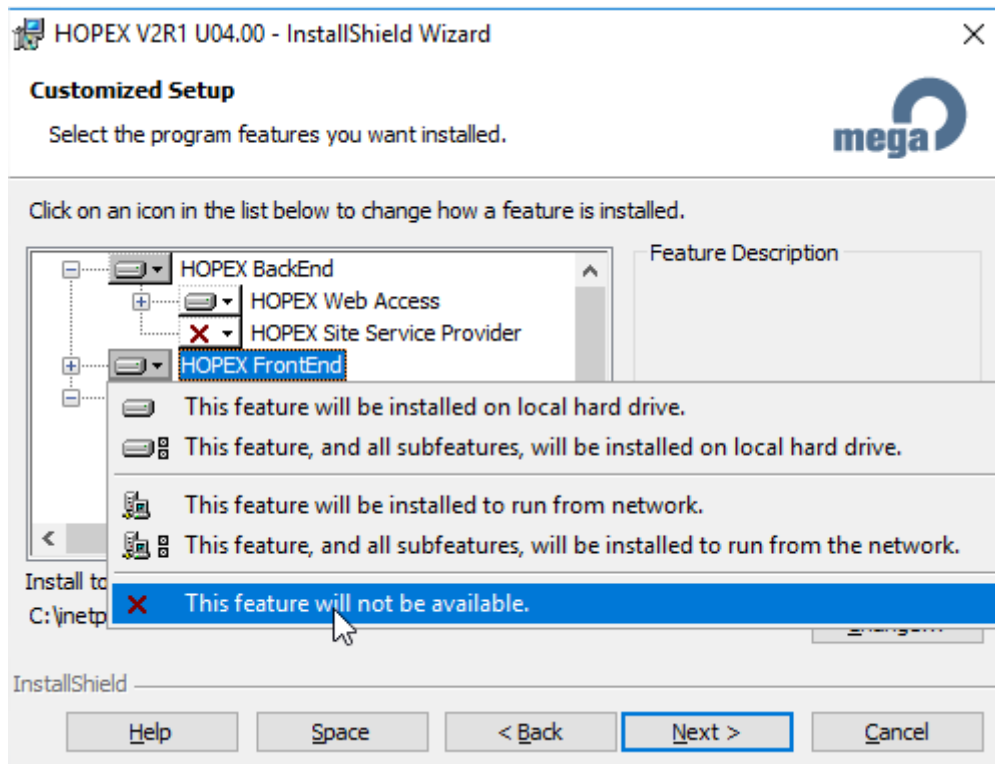
- In “Hopex BackEnd”, all the subfeatures of “Hopex Web Access”.

- The default of what is contained in “Mega Software”.

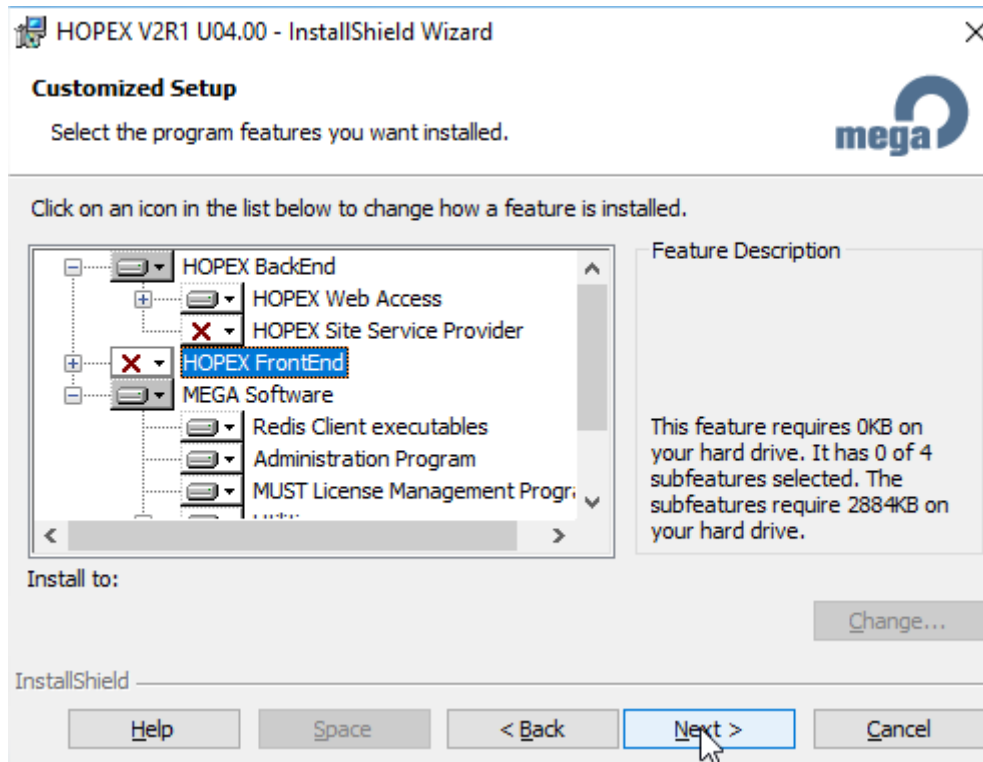
First, we activate features and subfeatures in “Hopex Web Access”:



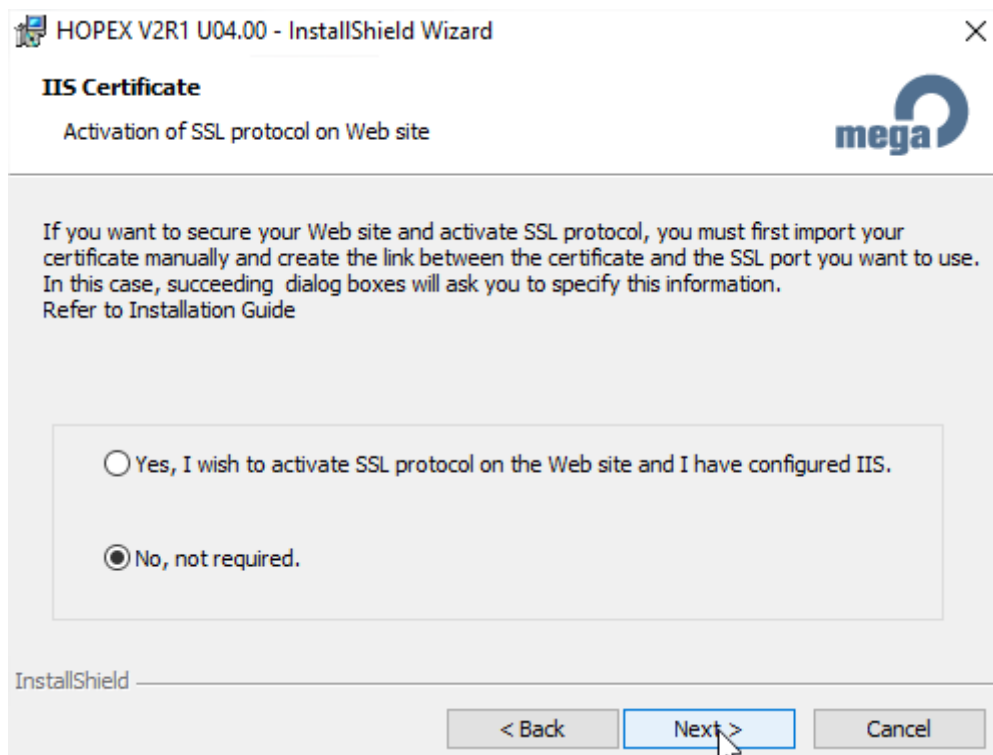
Then, we disable “Hopex FrontEnd”:



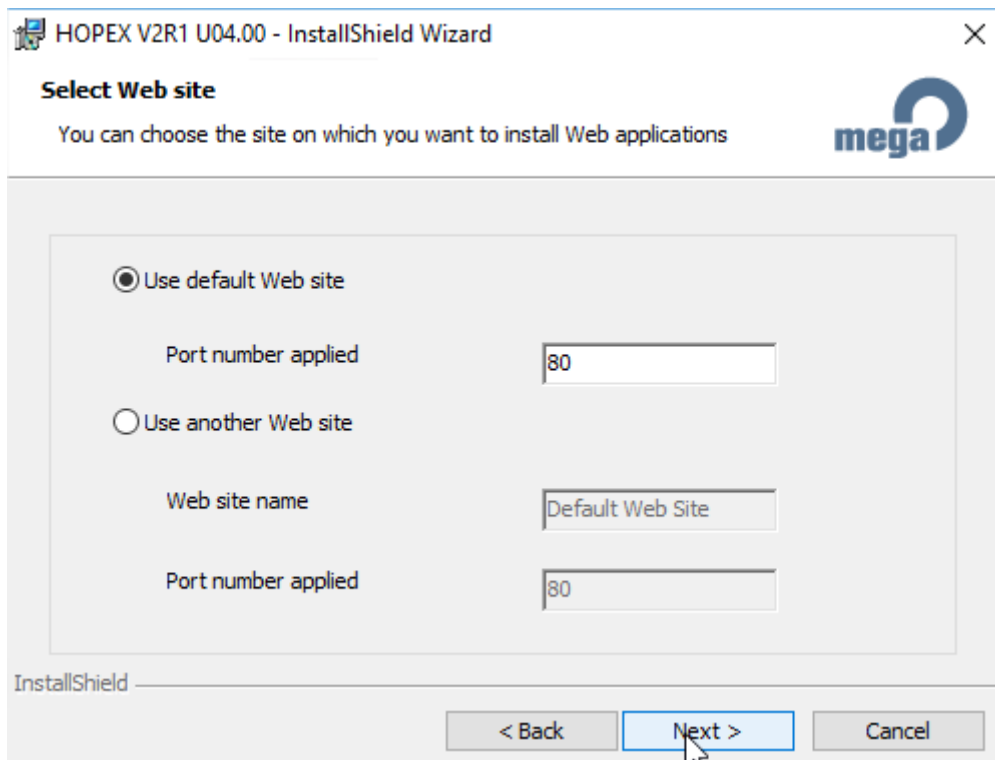
And keep “MEGA Software” as it is. Then click “Next”:



The SSL is not activated directly on the web/applications servers, so choose “No...”:



We use the default website on port 80:



HOPEX V2R1 U04.00 - InstallShield Wizard

Select Web site

You can choose the site on which you want to install Web applications

☒ Use default Web site

Port number applied: 80

☐ Use another Web site

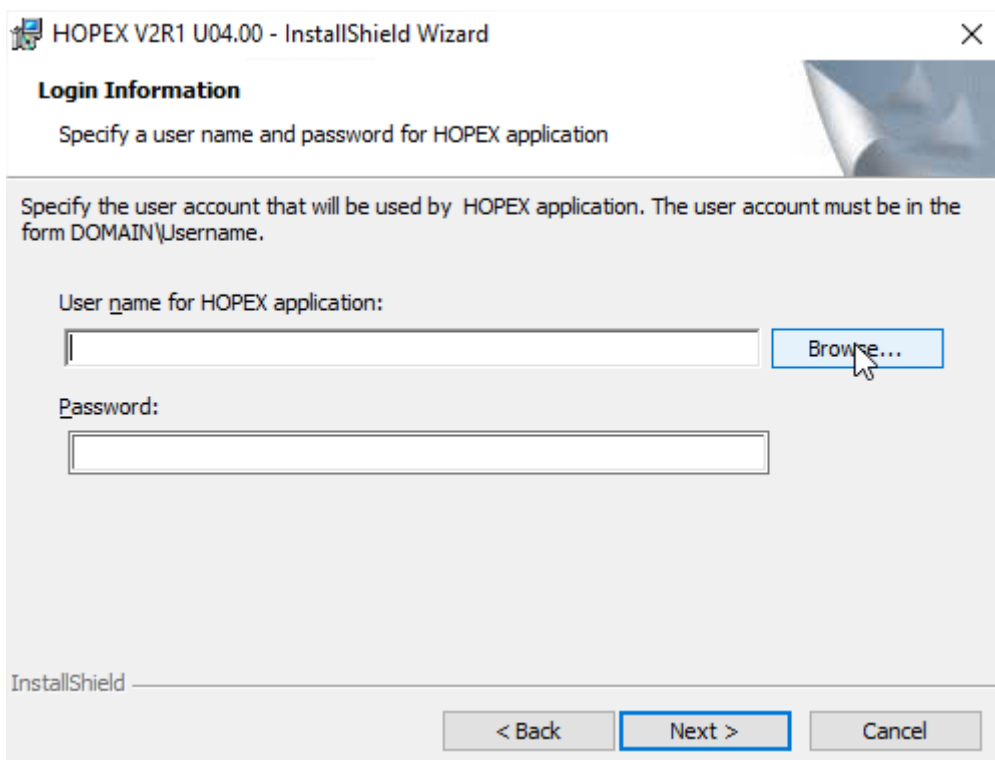
Web site name: Default Web Site

Port number applied: 80

InstallShield

< Back Next > Cancel

We click "Browse" to search for the first impersonate user:



HOPEX V2R1 U04.00 - InstallShield Wizard

Login Information

Specify a user name and password for HOPEX application

Specify the user account that will be used by HOPEX application. The user account must be in the form DOMAIN\Username.

User name for HOPEX application:

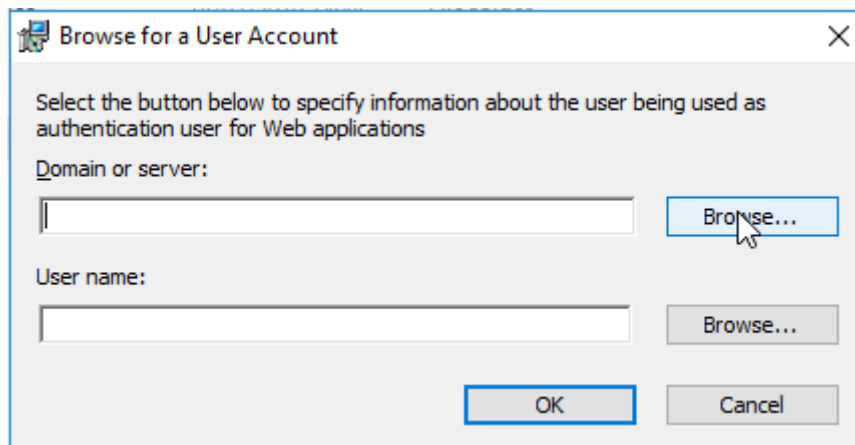
Browser...

Password:

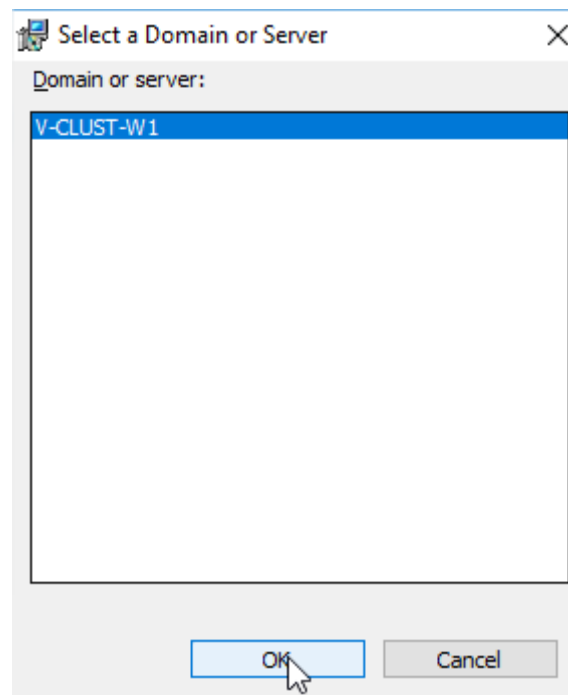
InstallShield

< Back Next > Cancel

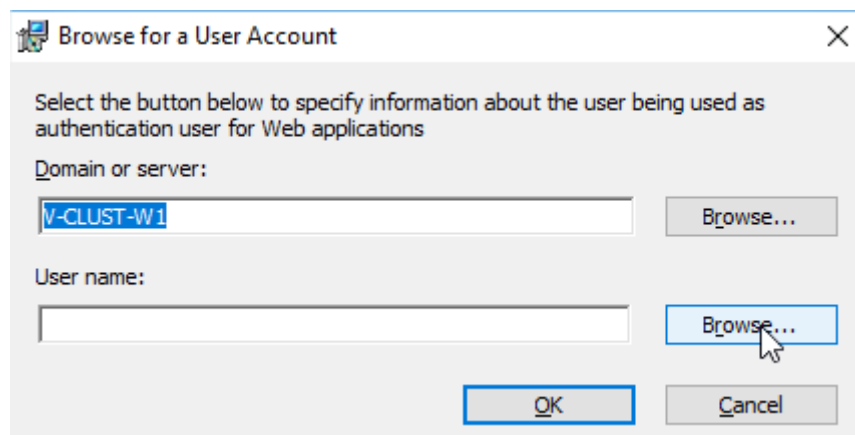
Again on "Browse":



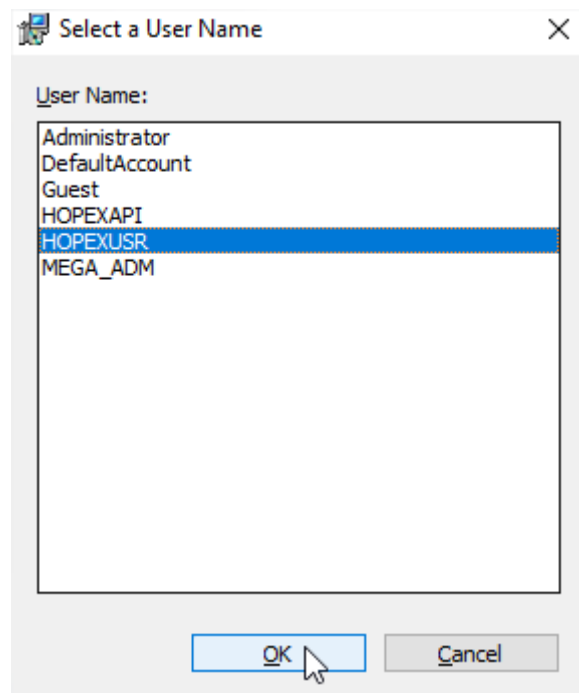
Select the local server:



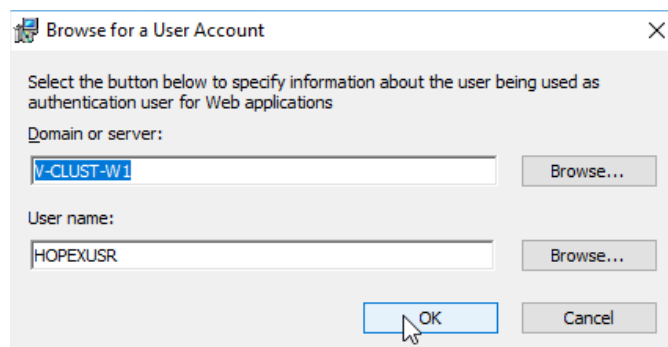
Then we click the second "Browse":



And we select the “HOPEXUSR” account:



Click « OK » :



We provide the password and click “Next”:

HOPEX V2R1 U04.00 - InstallShield Wizard

Login Information
Specify a user name and password for HOPEX application

Specify the user account that will be used by HOPEX application. The user account must be in the form DOMAIN\Username.

User name for HOPEX application:
 Browse...

Password:

InstallShield

< Back Next > Cancel

Proceed with the same steps with the “HOPEXAPI” account for the “HOPEXAPI” application:

HOPEX V2R1 U04.00 - InstallShield Wizard

Login Information
Specify a user name and password for HOPEXAPI application

Specify the user account that will be used by HOPEXAPI application. The user account must be in the form DOMAIN\Username.

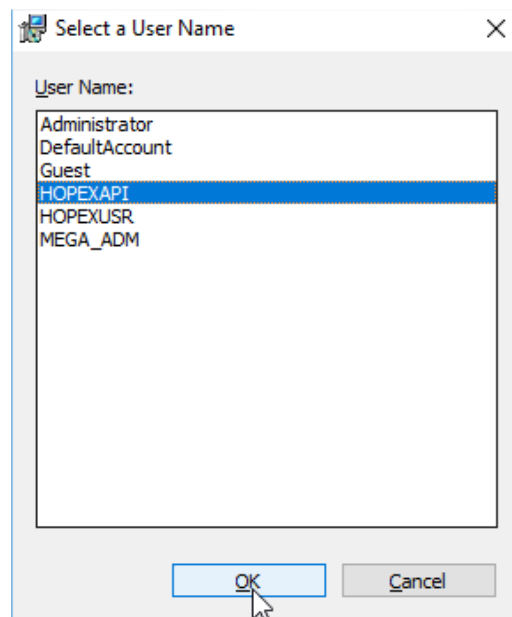
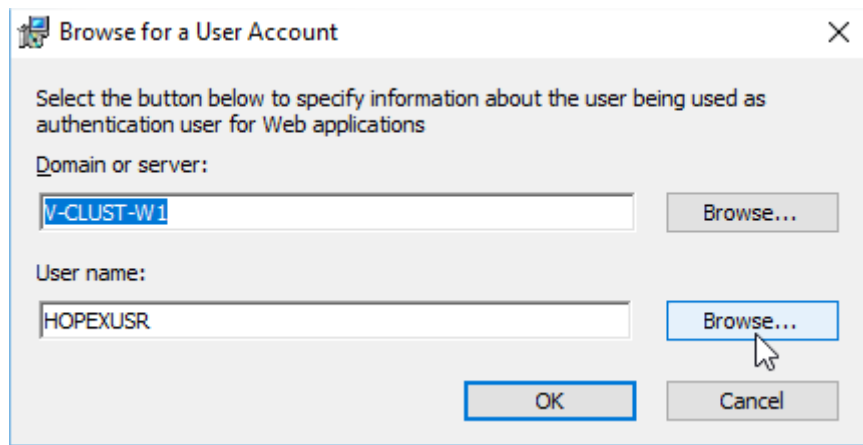
User name for HOPEX API application:
 Browse...

Password:

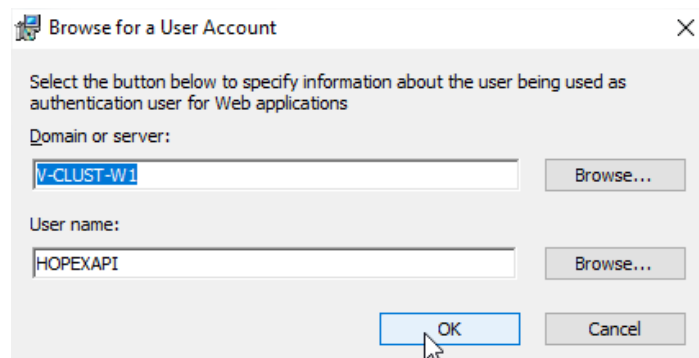
InstallShield

< Back Next > Cancel

It remembers previous settings, so click the second “Browse” button, to switch:



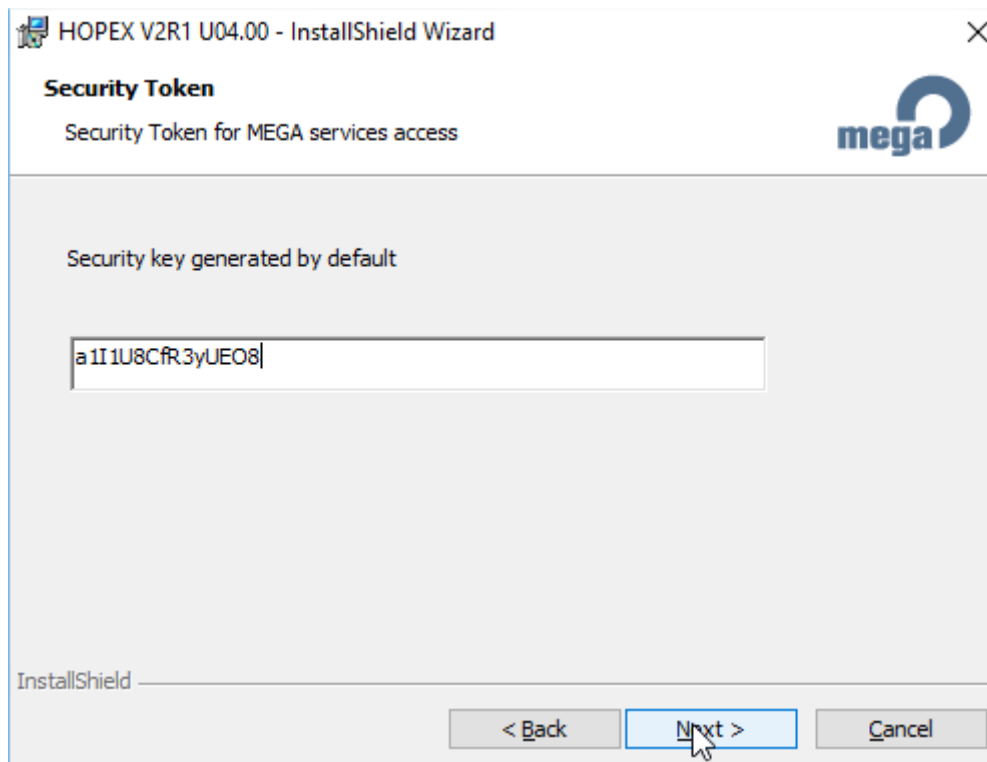
Click "OK":



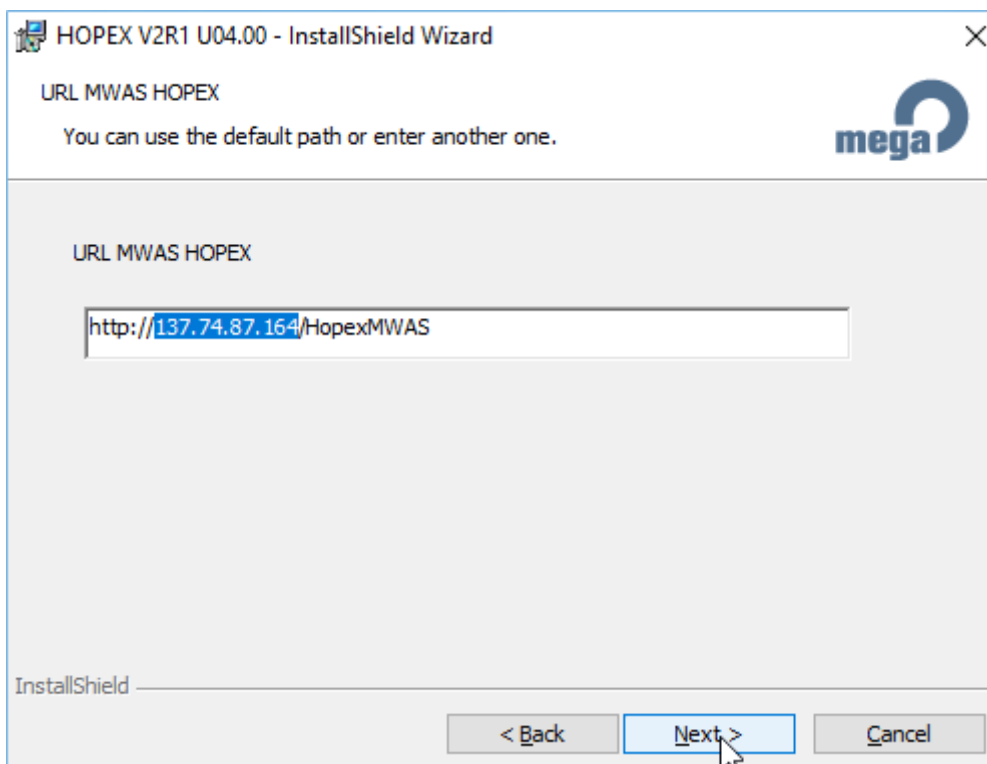
Provide the password of the user and click "Next":

For the SSP Url, we use the IP address of the Load Balancer, so in this case “137.74.87.169”:

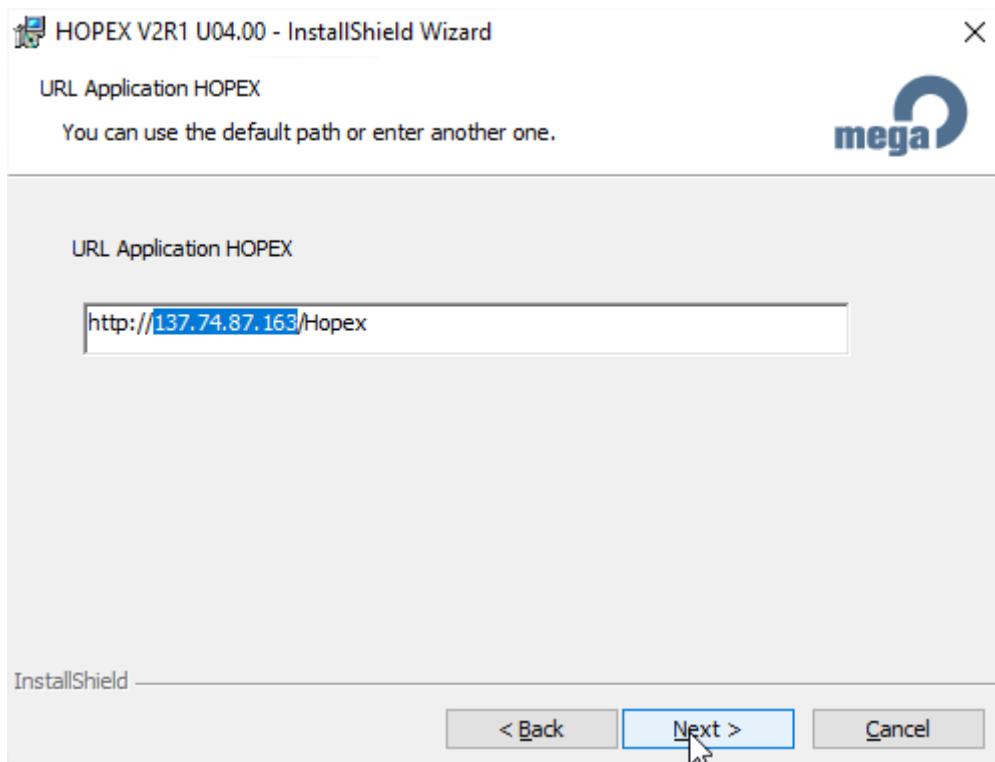
Security key (it is the one that was generated on the first web server that we installed).
a1I1U8CfR3yUEO8



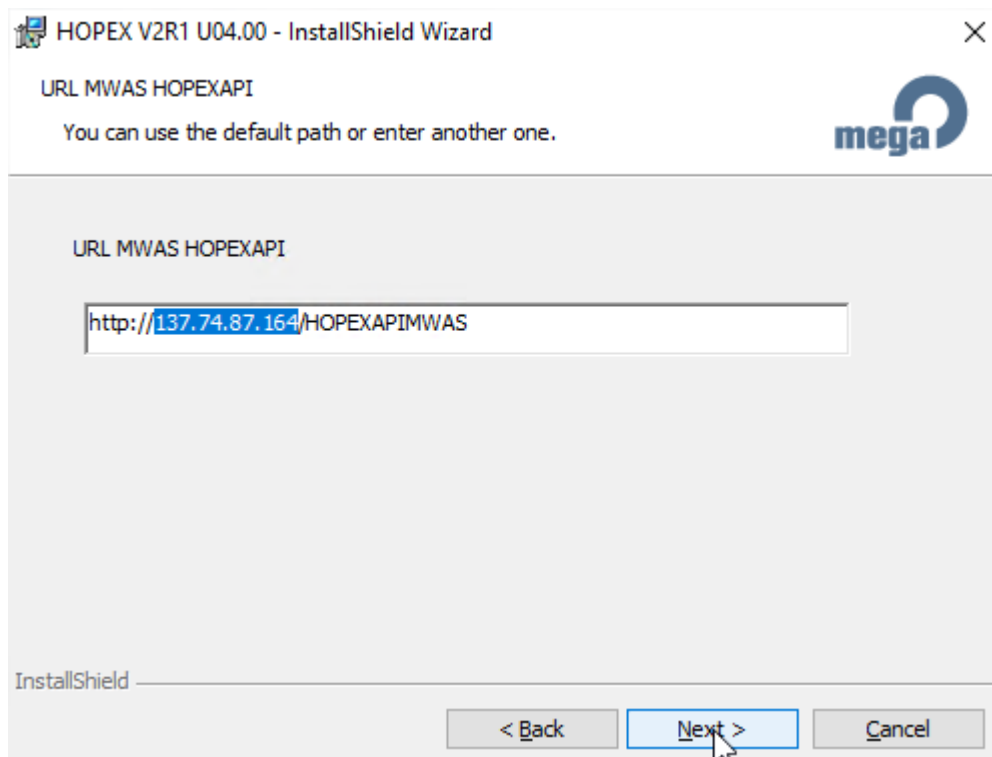
We replace the server name of the MWAS server by its IP address (as they don't belong to a domain, and are not known by the DNS server). For this example, IP address is "137.74.87.164". Do not forget to adapt depending on each server you install:



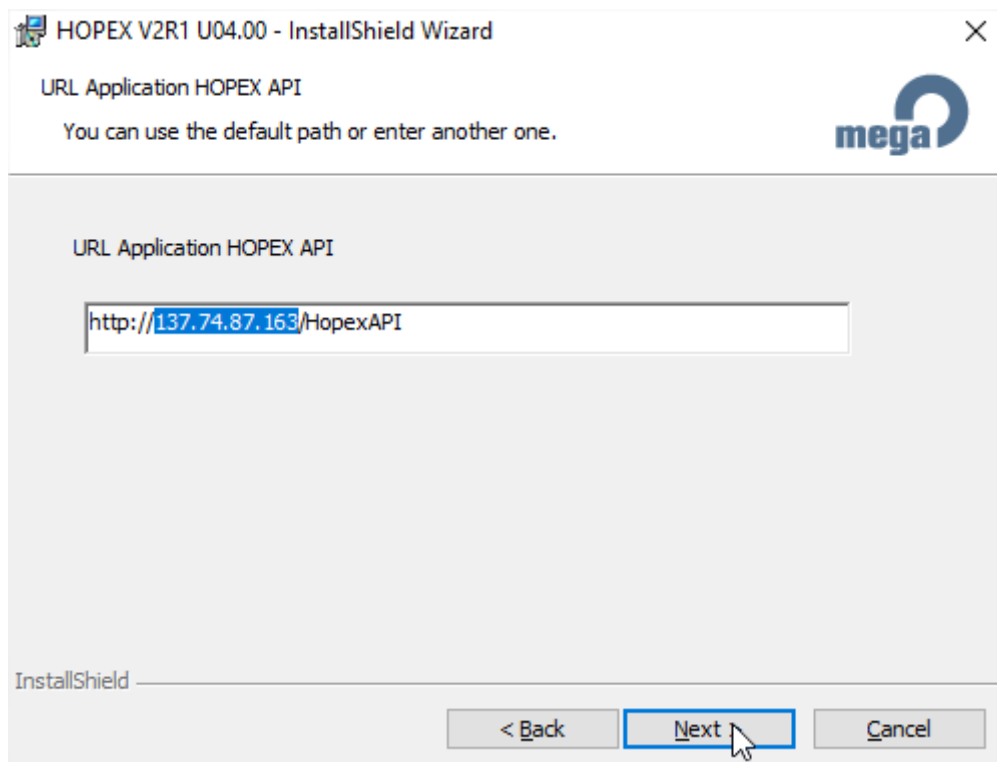
For the "HOPEX" URL, we use the IP address of the web Load Balancer, so here "137.74.87.163":



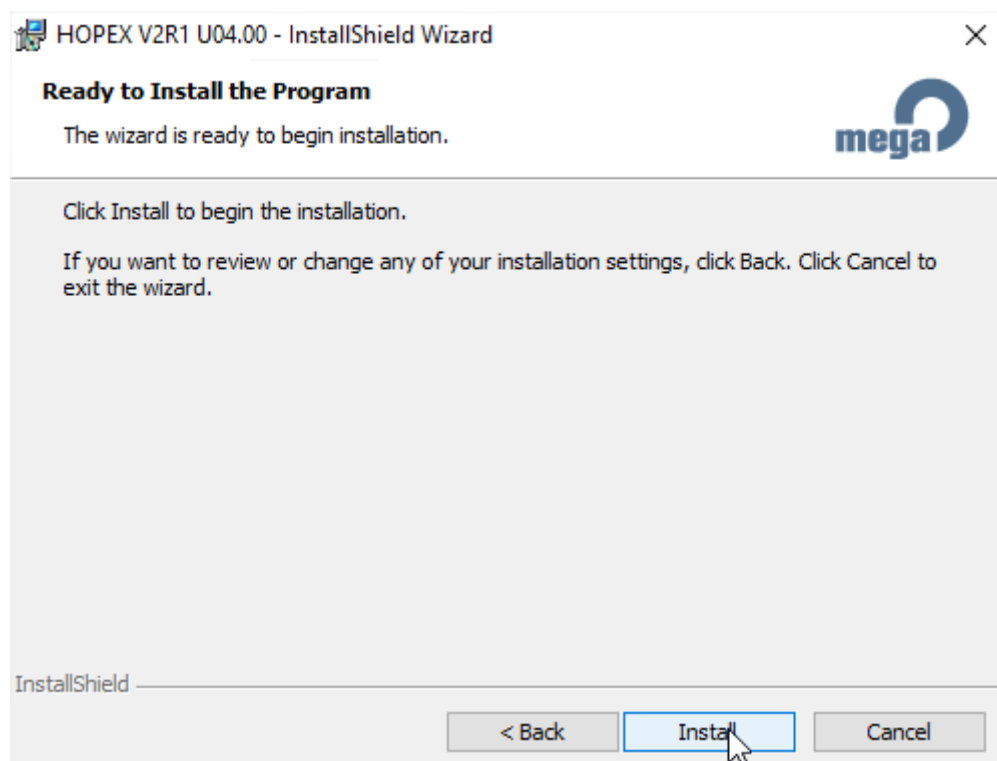
Same thing with the MWAS API URL, where we use the IP address of the server rather than its name:



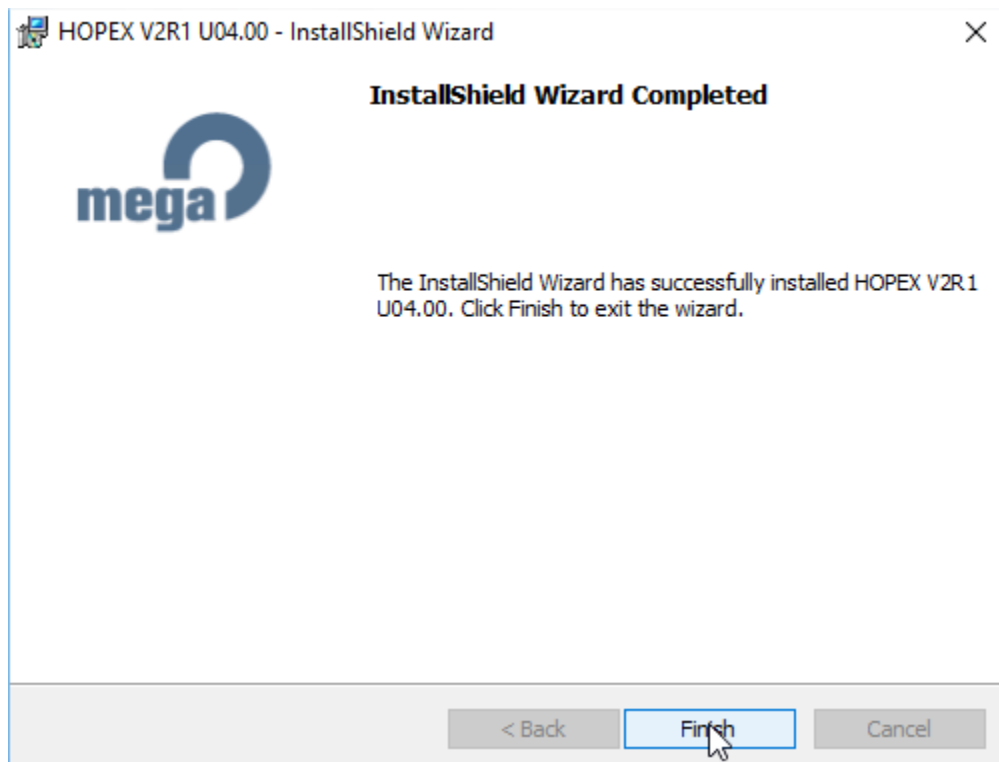
And for the "HOPEX API" URL, we also use the IP address of the web front-end Load Balancer, so "137.74.87.163":



We click "Install" to start deploying the tool:

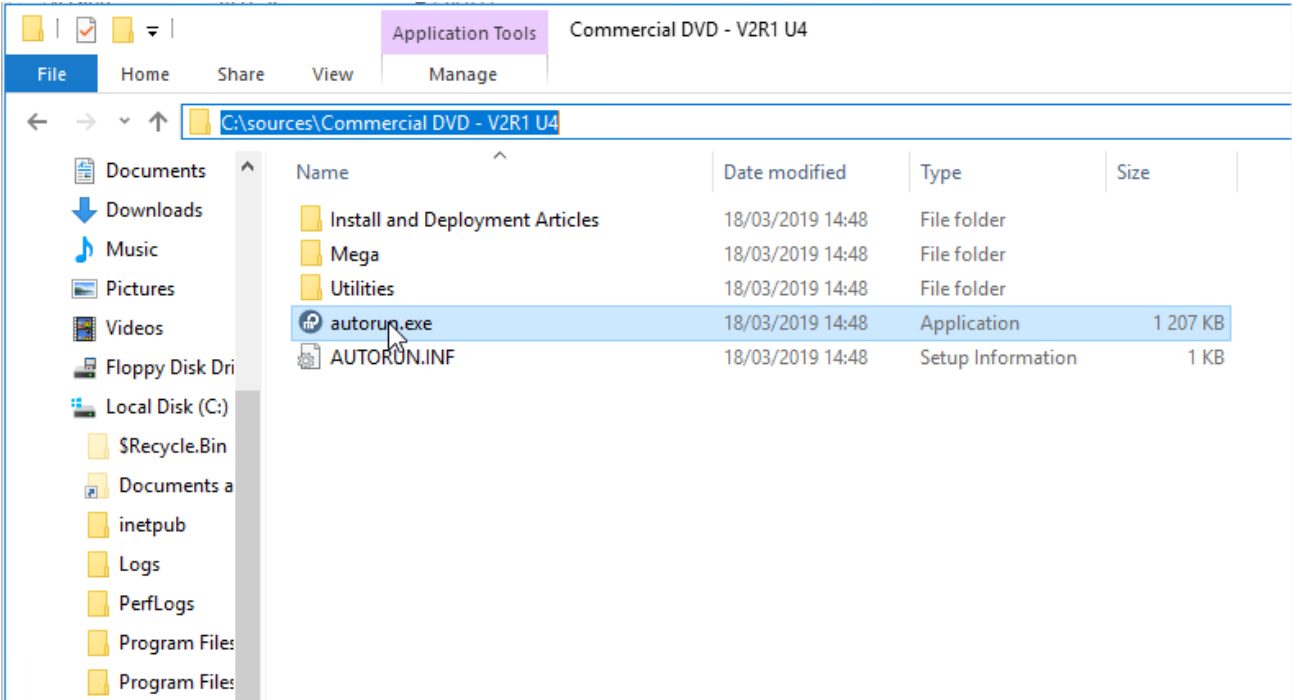


Click « Finish »:

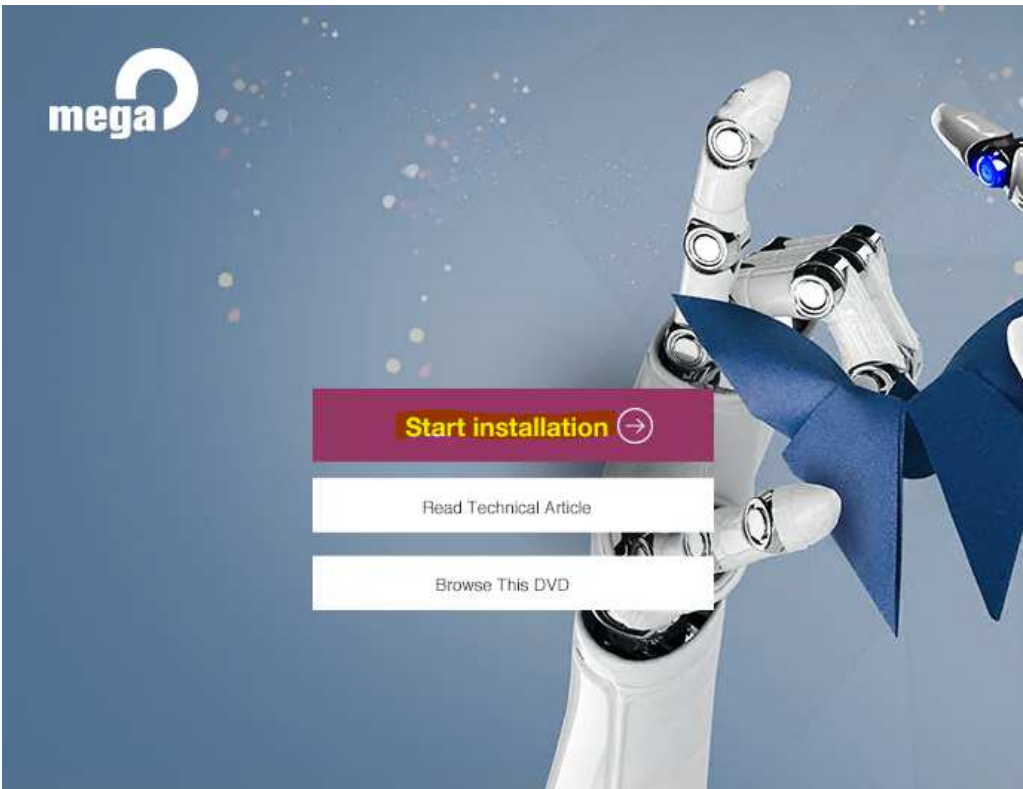


INSTALL MEGA ON THE SSP APPLICATION SERVERS

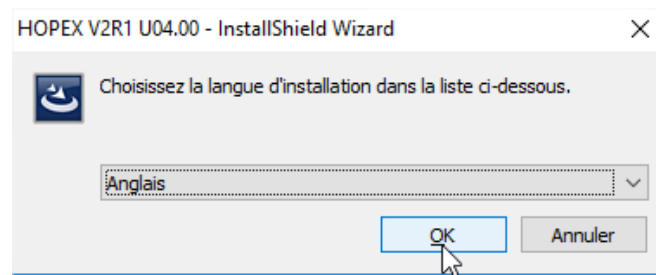
Launch the install (autorun.exe file in the install folder of Mega Hopex, in "C:\sources\Commercial DVD - V2R1 U4"):



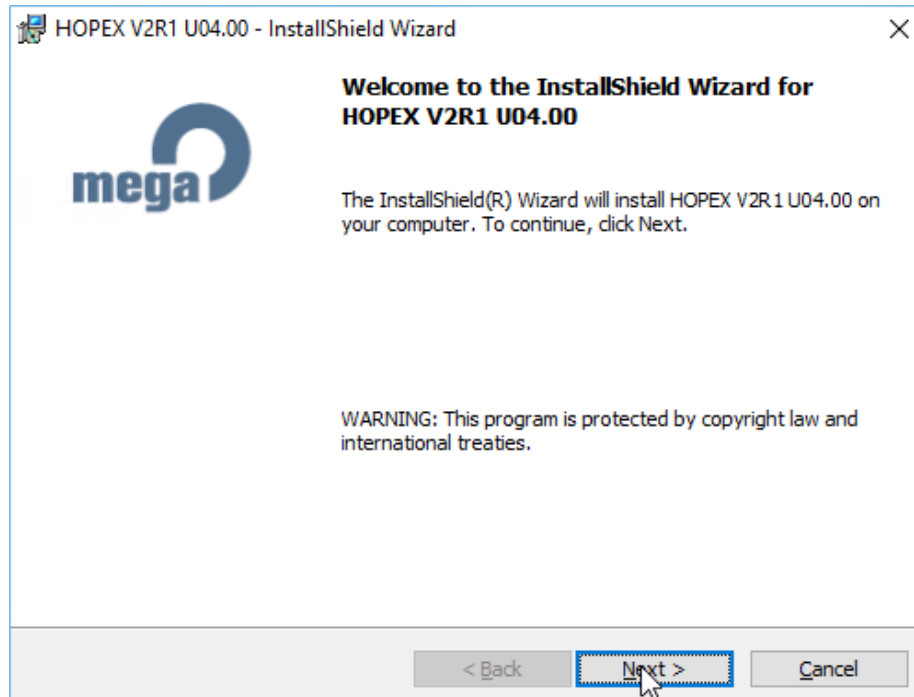
Click « Start installation »:



Choose "Anglais" (for "English", because the virtual machine was installed with french regional settings):



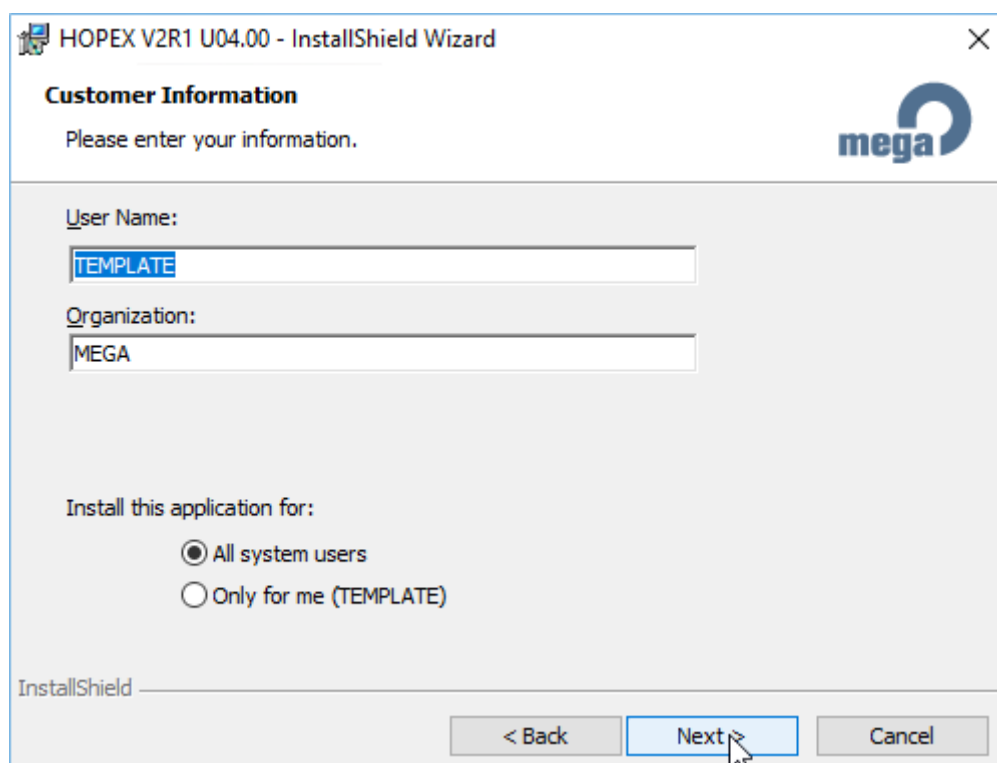
Click « Next » :



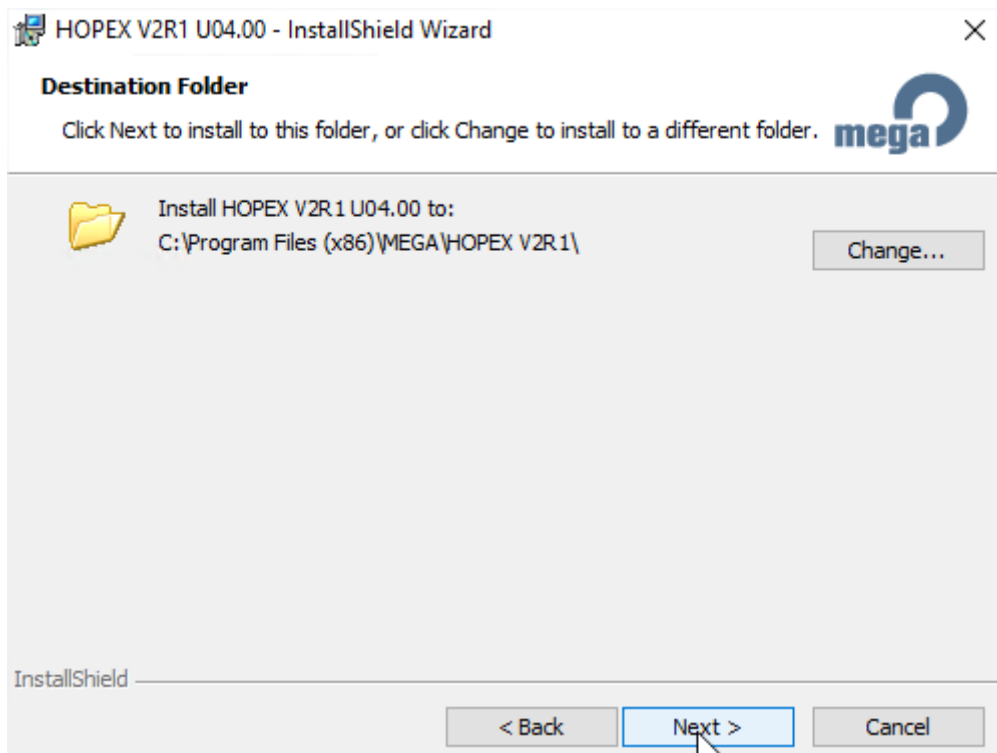
Accept, and again click « Next »:



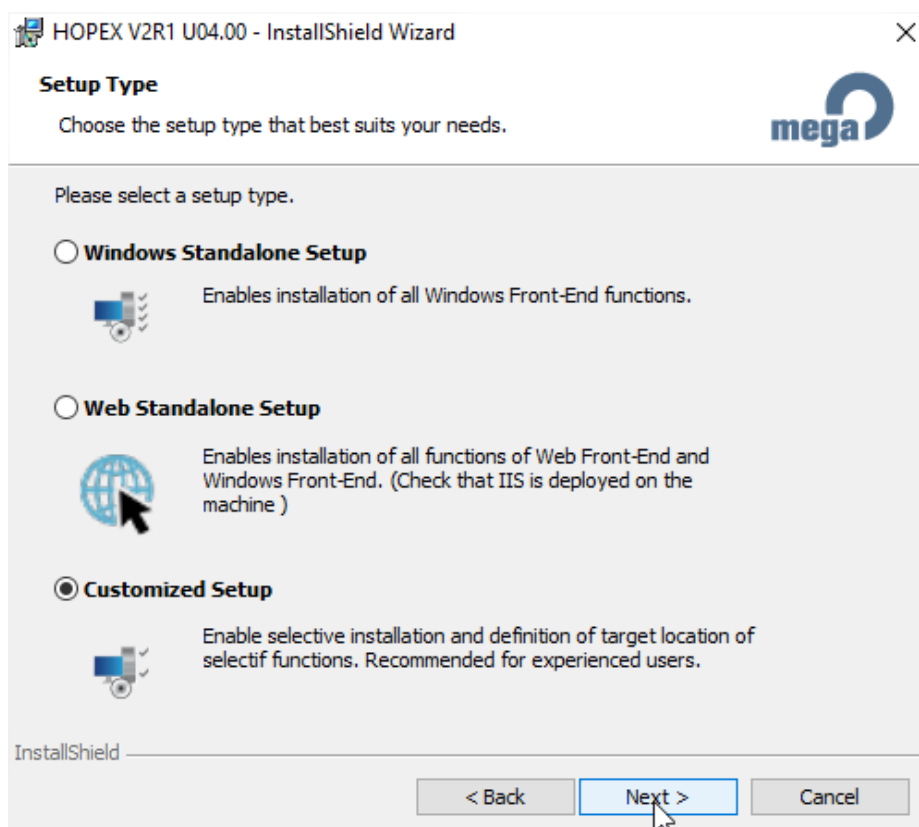
Click « Next » :



By default, the binaries of Mega are installed on the C drive in the below location. We keep that setting and click “Next”:



Choose “Customized setup”:

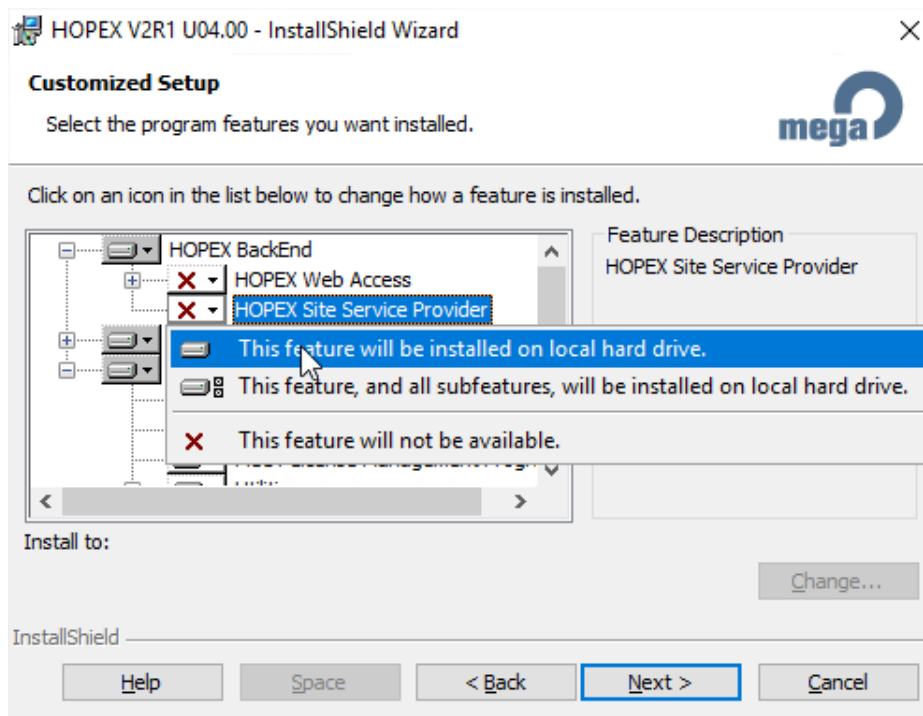


In the list of products to install, we will have:

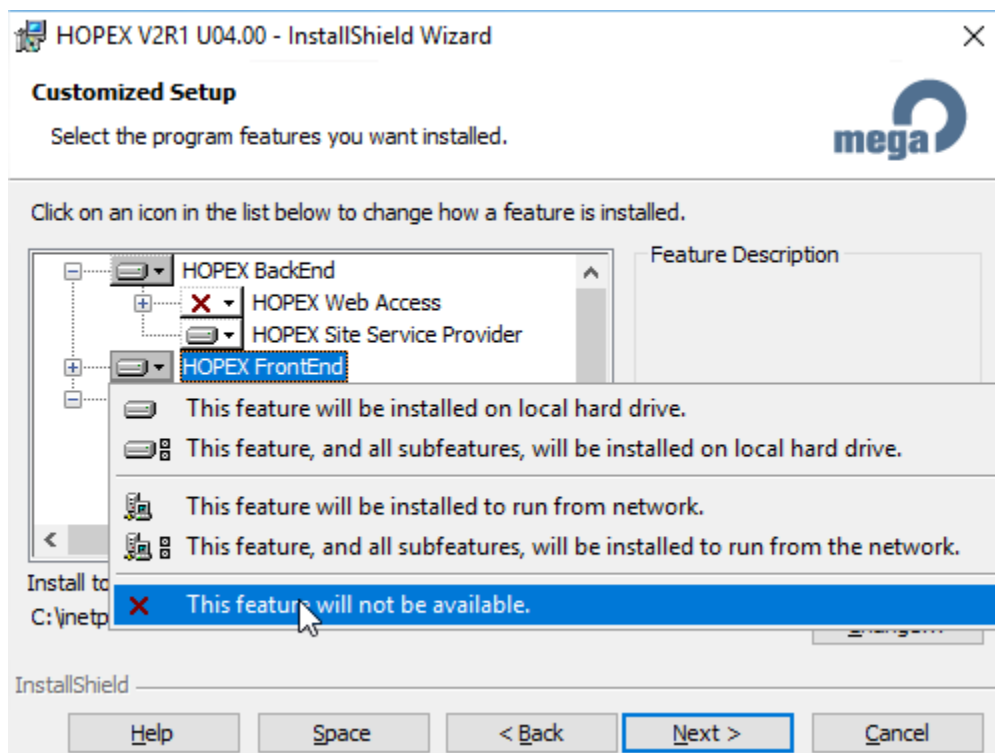
- In “Hopex BackEnd”, only “Hopex Site Service Provider”.

- The default of what is contained in “Mega Software”.

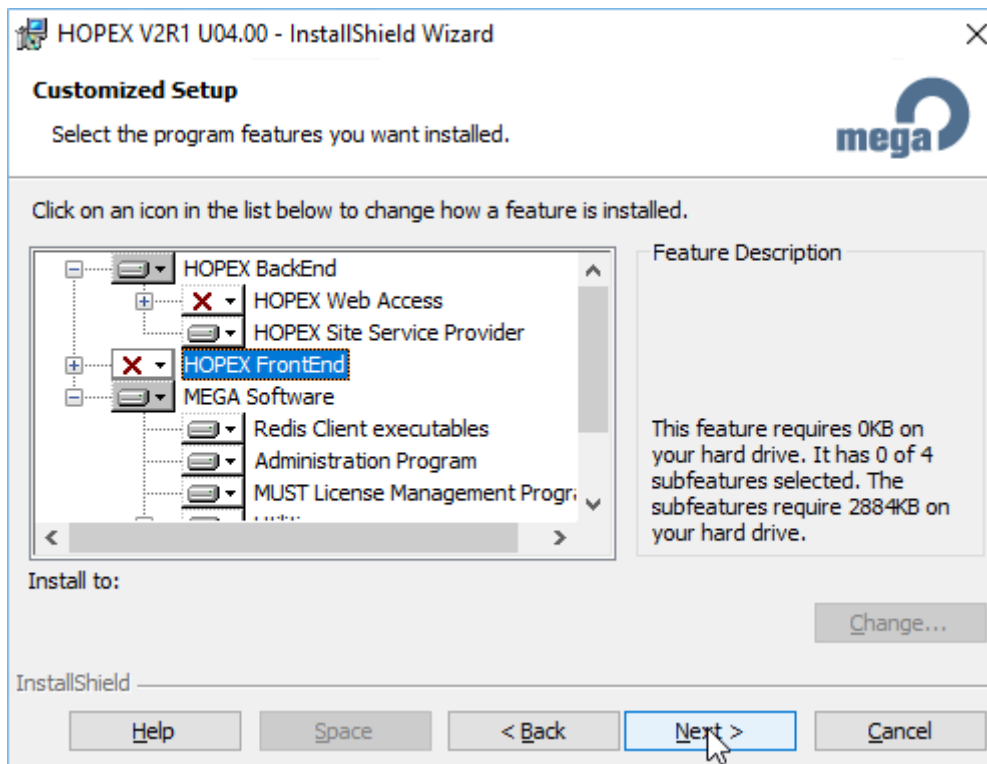
First, we activate “Hopex Site Service Provider”:



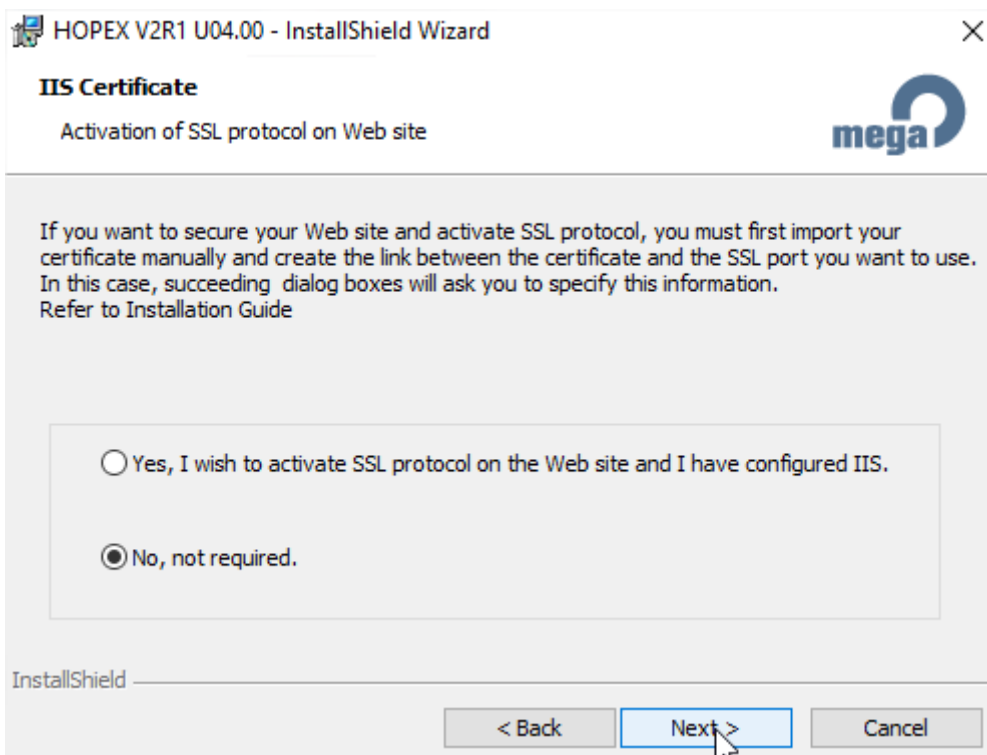
Then, we disable “Hopex FrontEnd”:



And keep “MEGA Software” as it is. Then click “Next”:



The SSL is not activated directly on the web/applications servers, so choose “No...”:



We use the default website on port 80:

HOPEX V2R1 U04.00 - InstallShield Wizard

Select Web site

You can choose the site on which you want to install Web applications

☒ Use default Web site

Port number applied: 80

☐ Use another Web site

Web site name: Default Web Site

Port number applied: 80

InstallShield

< Back Next > Cancel

For the SSP Url, we use the IP address of the Load Balancer, so in this case “137.74.87.169”:

HOPEX V2R1 U04.00 - InstallShield Wizard

MEGA Site Provider

You can use the default path or enter another one.

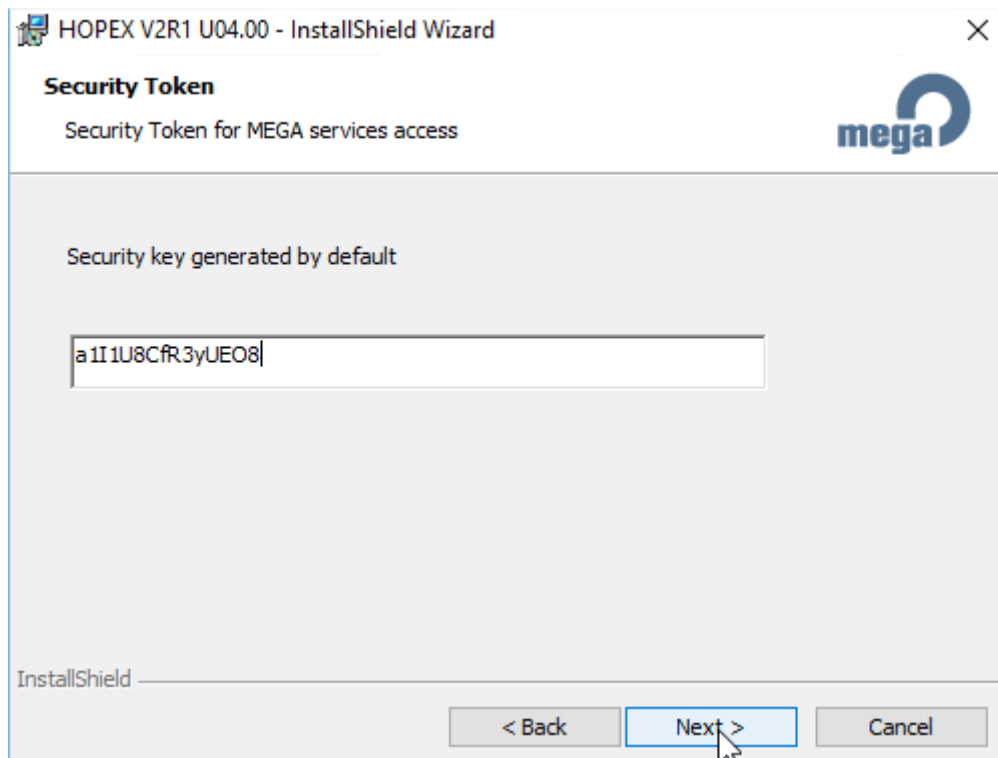
MEGA Site Provider

http://137.74.87.169/MegaSSP

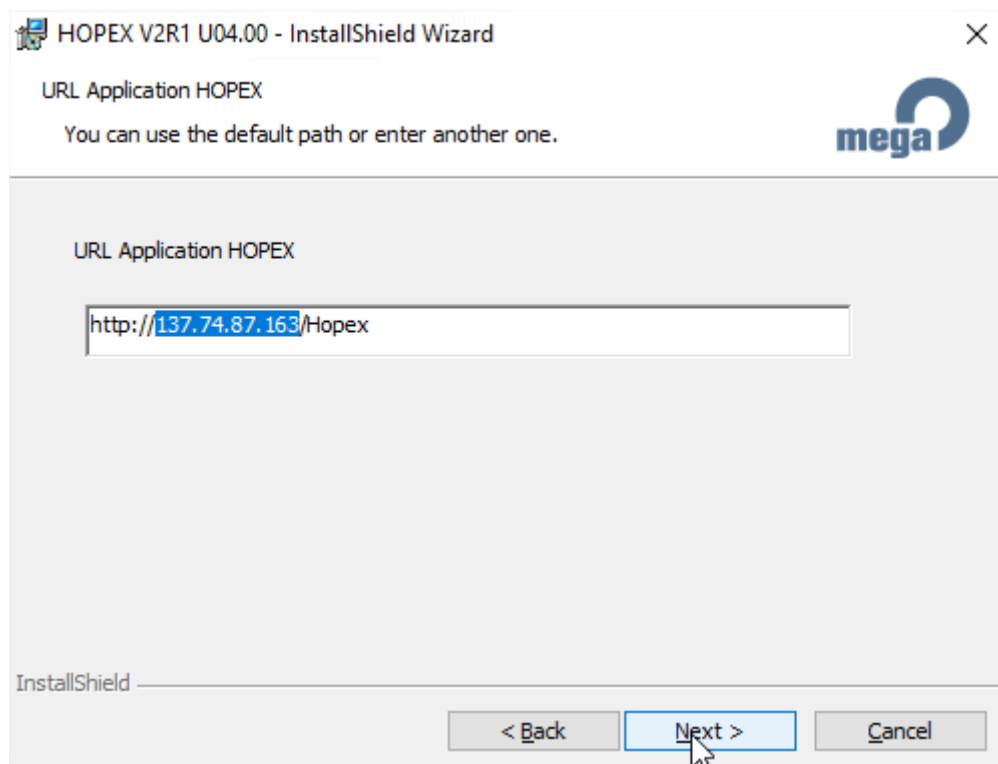
InstallShield

< Back Next > Cancel

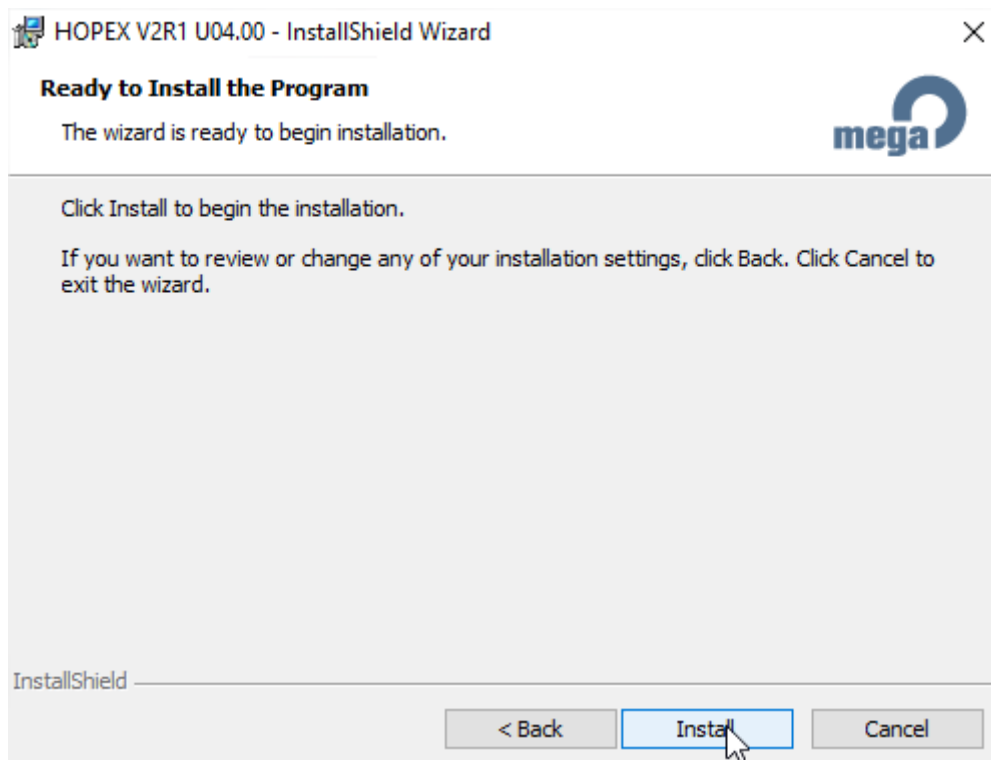
Security key (it is the one that was generated on the first web server that we installed).
a1I1U8CfR3yUEO8



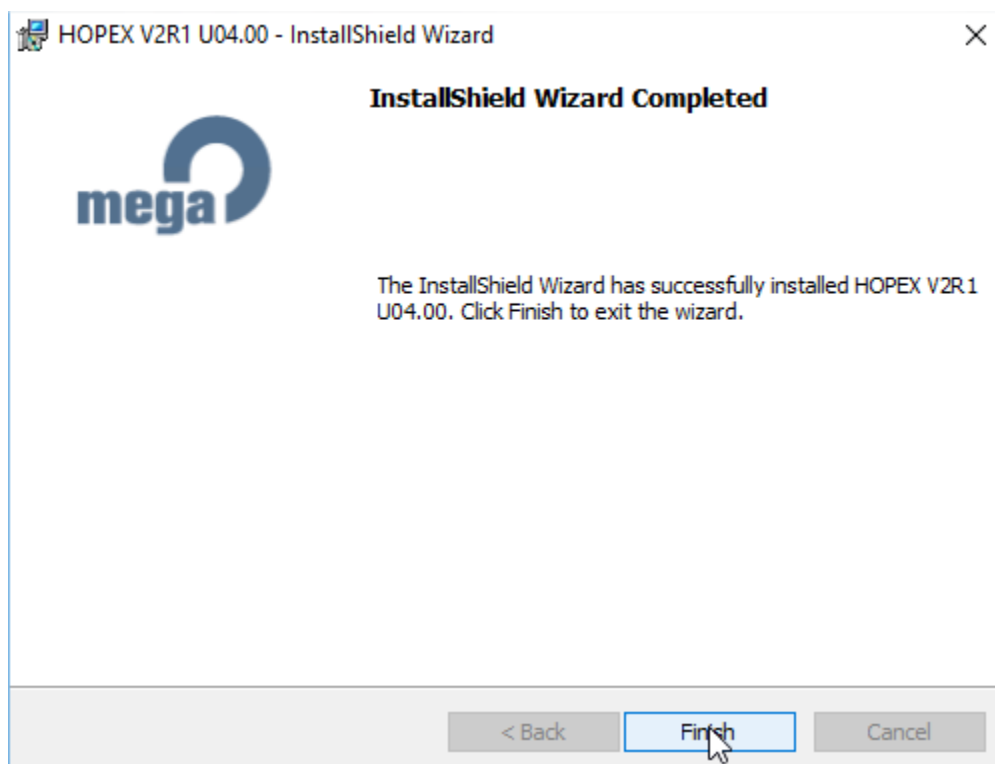
For the "HOPEX" URL, we use the IP address of the web Load Balancer, so here "137.74.87.163":



We click "Install" to start deploying the tool:



Click « Finish »:



FINISH THE INSTALLATION

Create share folders

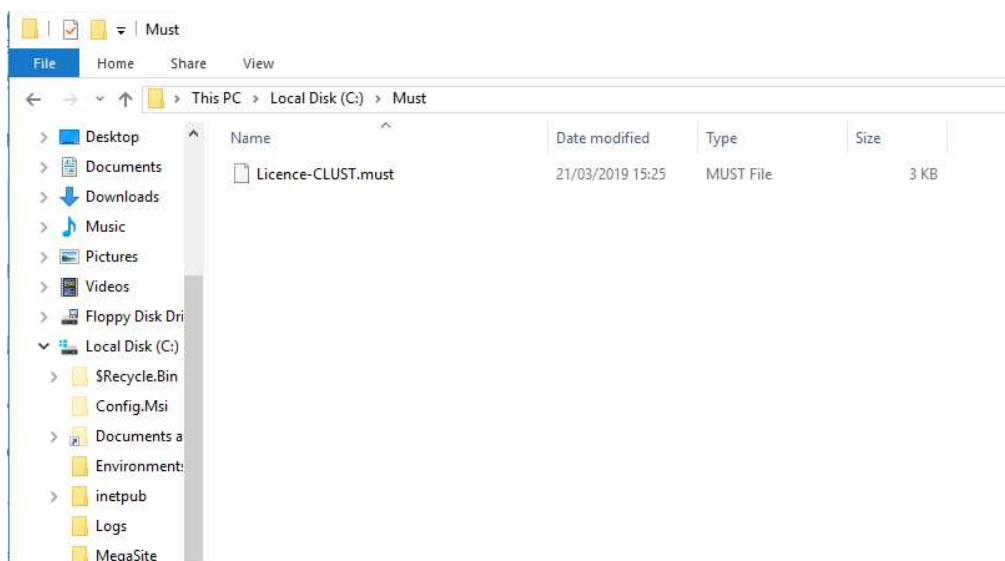
All shares will be hosted on the RDBMS Server. This is a choice that needs to be made before the deployment. Here we chose the one server that was alone and had a unique role.

License share

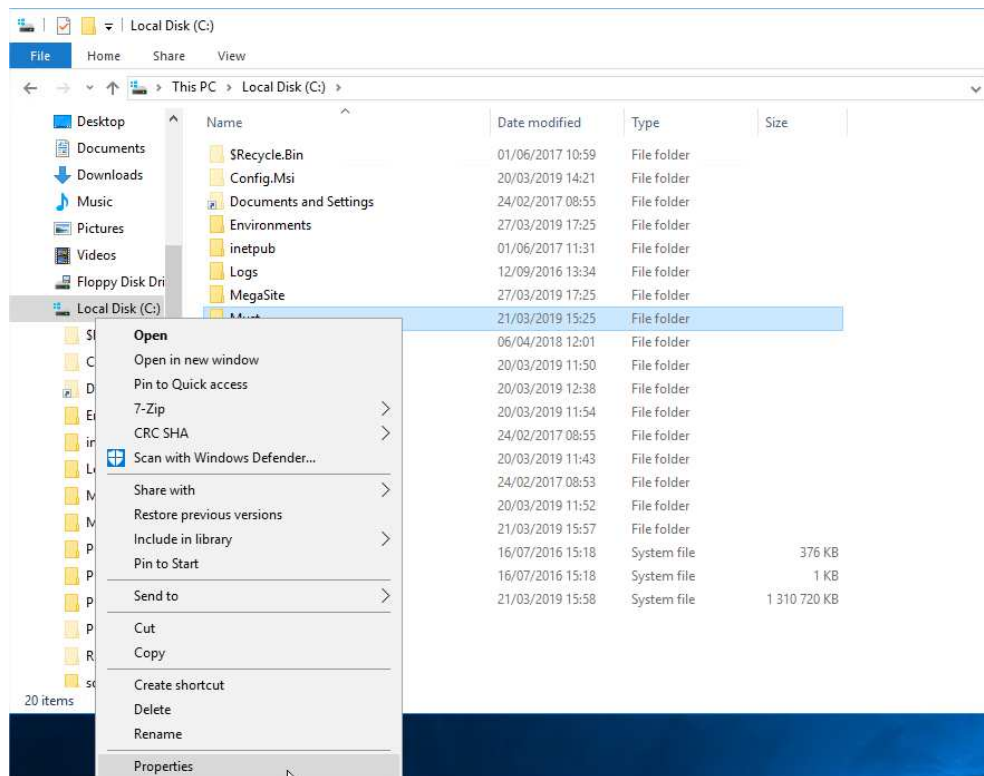
The license was generated to be hosted on the RDBMS server, and accessed on this address:

<\\V-CLUST-SQL\Must>

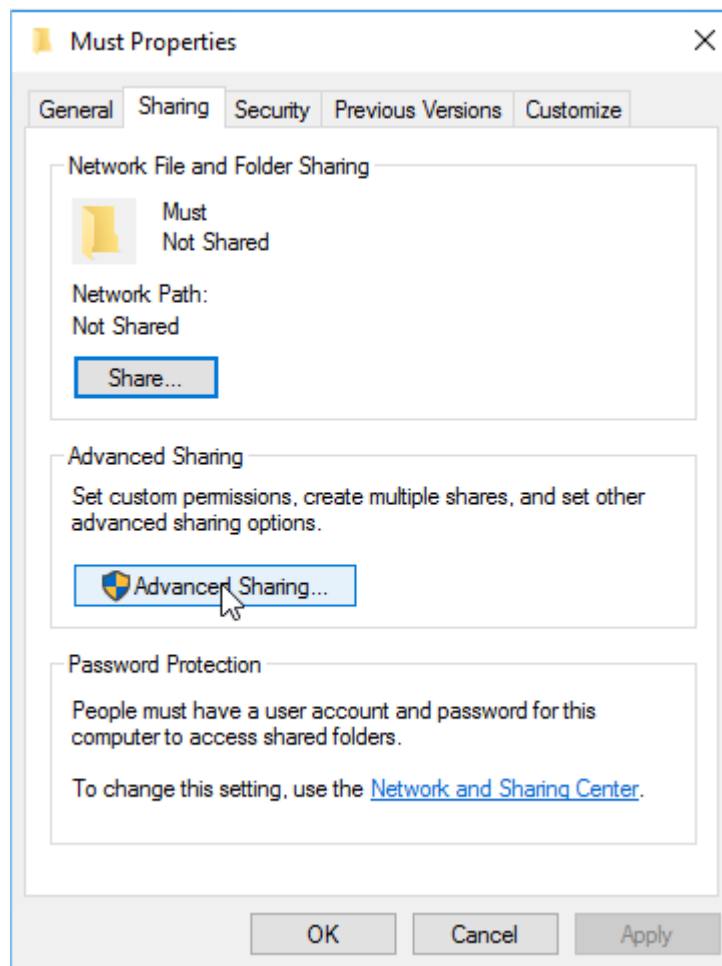
The physical folder is located on the RDBMS Server, and is "C:\Must" (note that the license file was already copied in the folder):



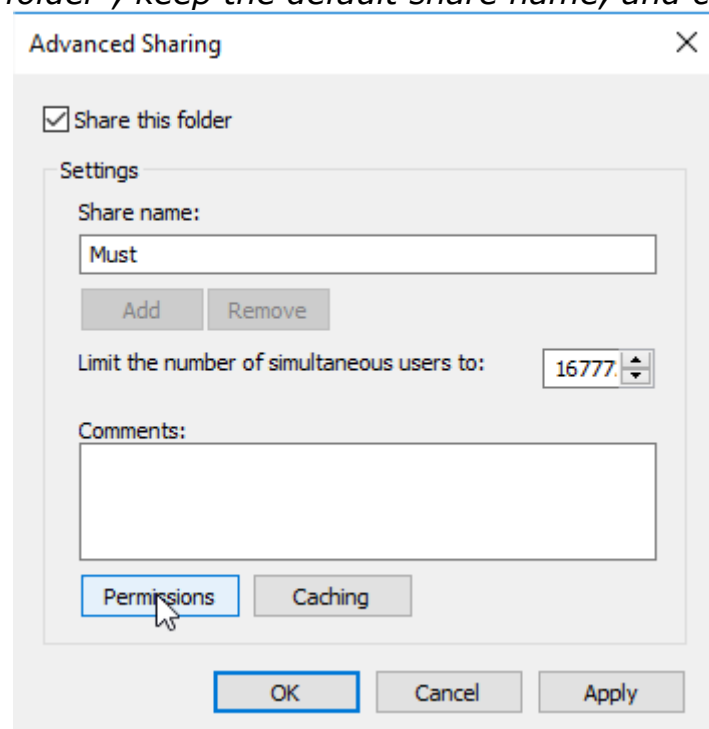
We open the properties of the folder:



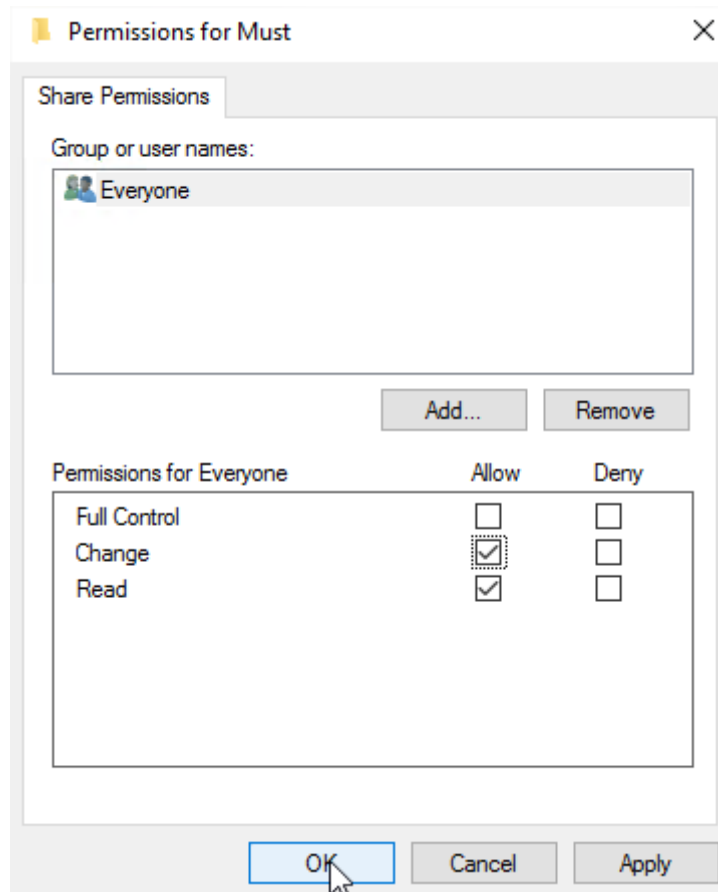
Go to the „Sharing“ tab and click „Advanced Sharing“:



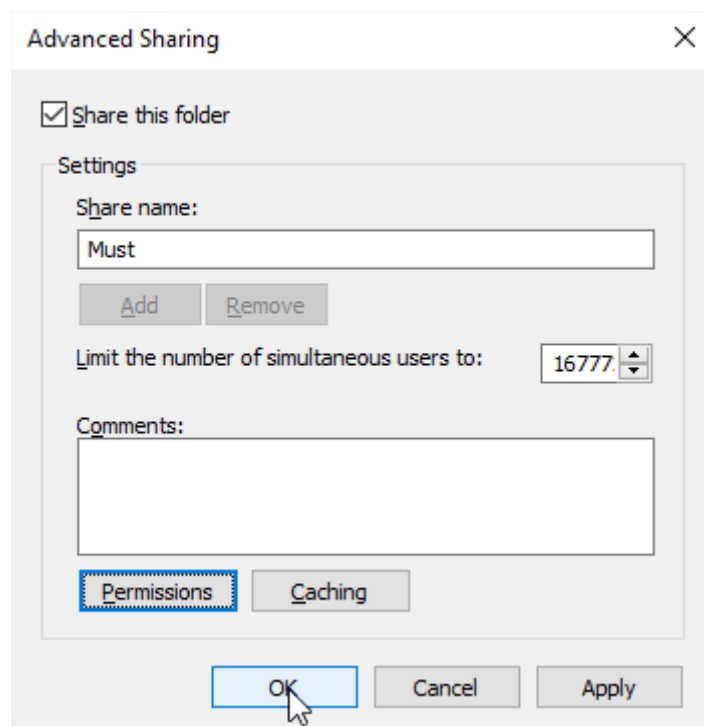
Click "Share this folder", keep the default share name, and click "Permissions":



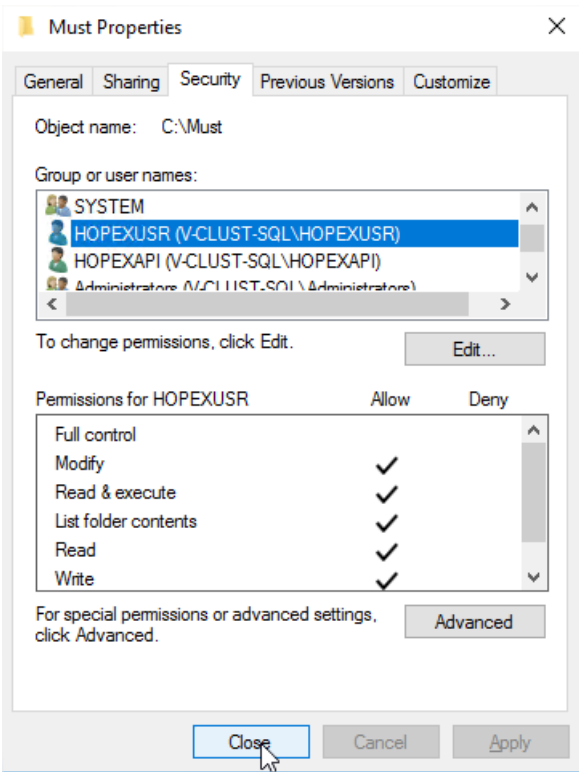
Allow Everyone the “Change” right, then click “OK”:



And again on “OK” to close the advanced sharing interface:



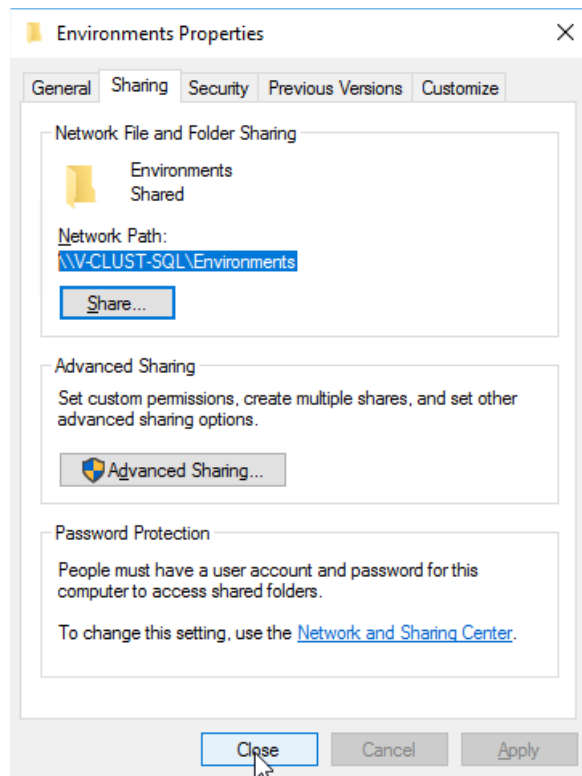
Then go to the "Security" tab, to confirm that the "HOPEXUSR" and "HOPEXAPI" users have read/write access:



Close the Properties. The share is operational.

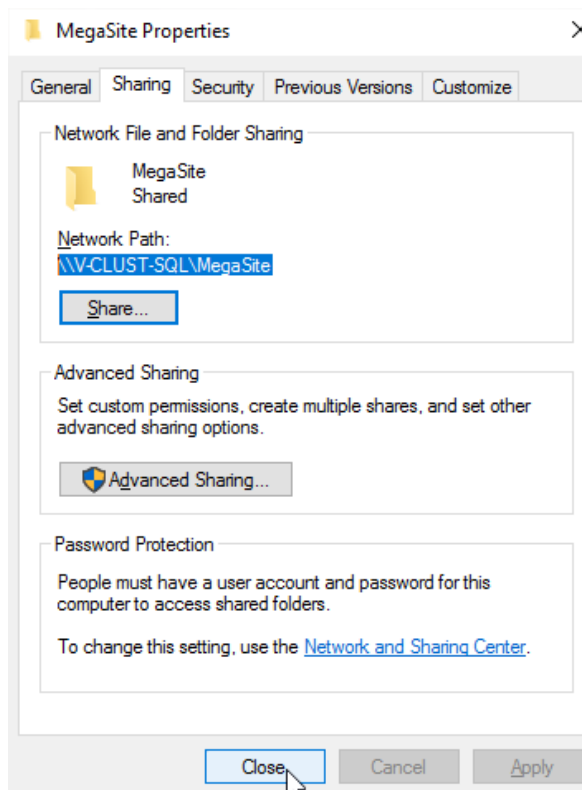
Environment share

We do the same steps on the "C:\Environments" folder, to have a share called [\\V-CLUST-SQL\Environments](#) :



MegaSite share

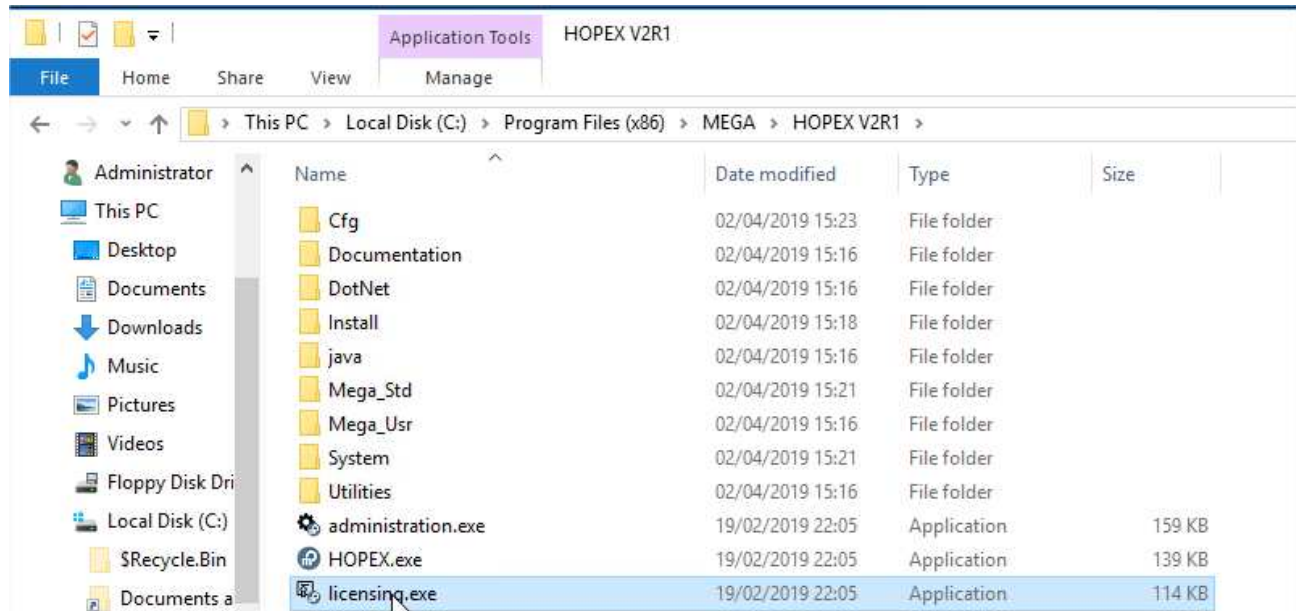
And eventually, the share that will host the clustered MegaSite file on folder “C:\MegaSite” to have the share [\\V-CLUST-SQL\MegaSite](#) :



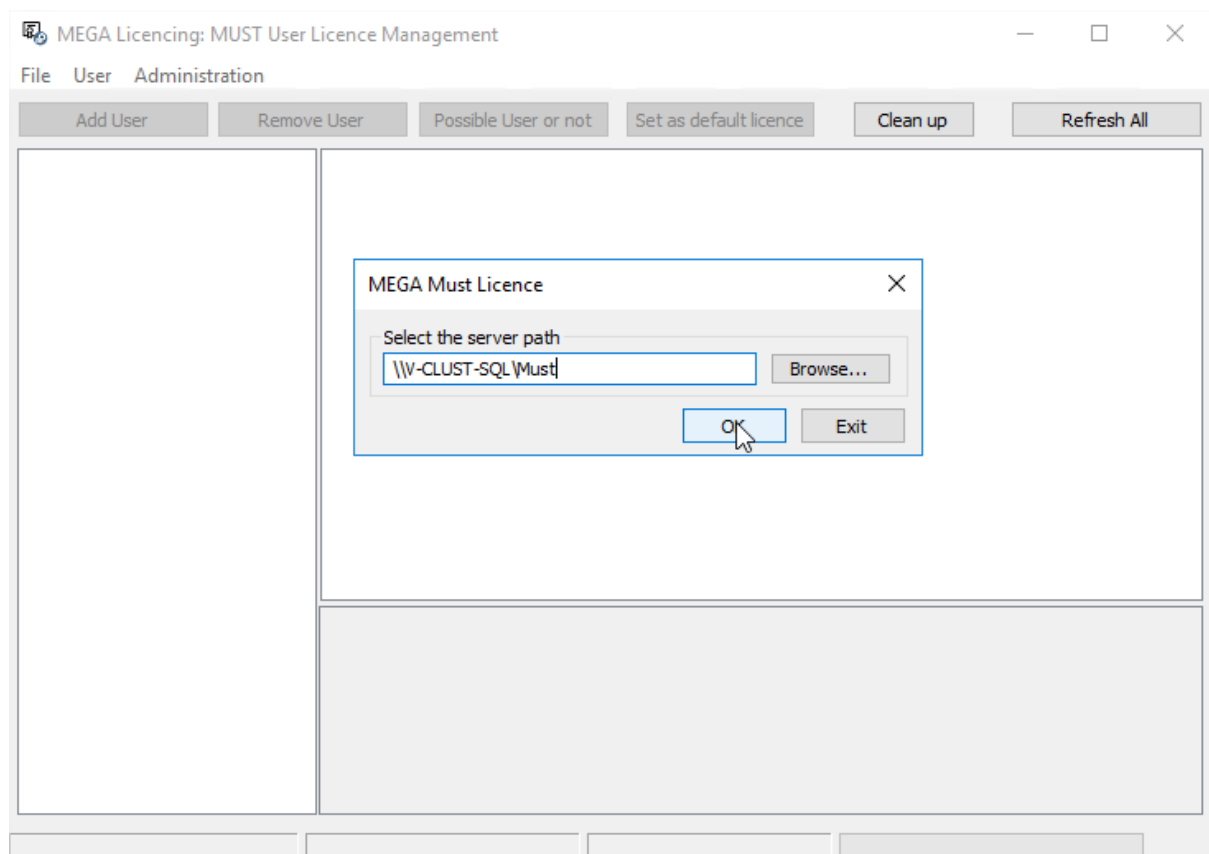
Configure the license

Connect to one of the application servers (in our case, on the first SSP Server V-CLUST-S1).

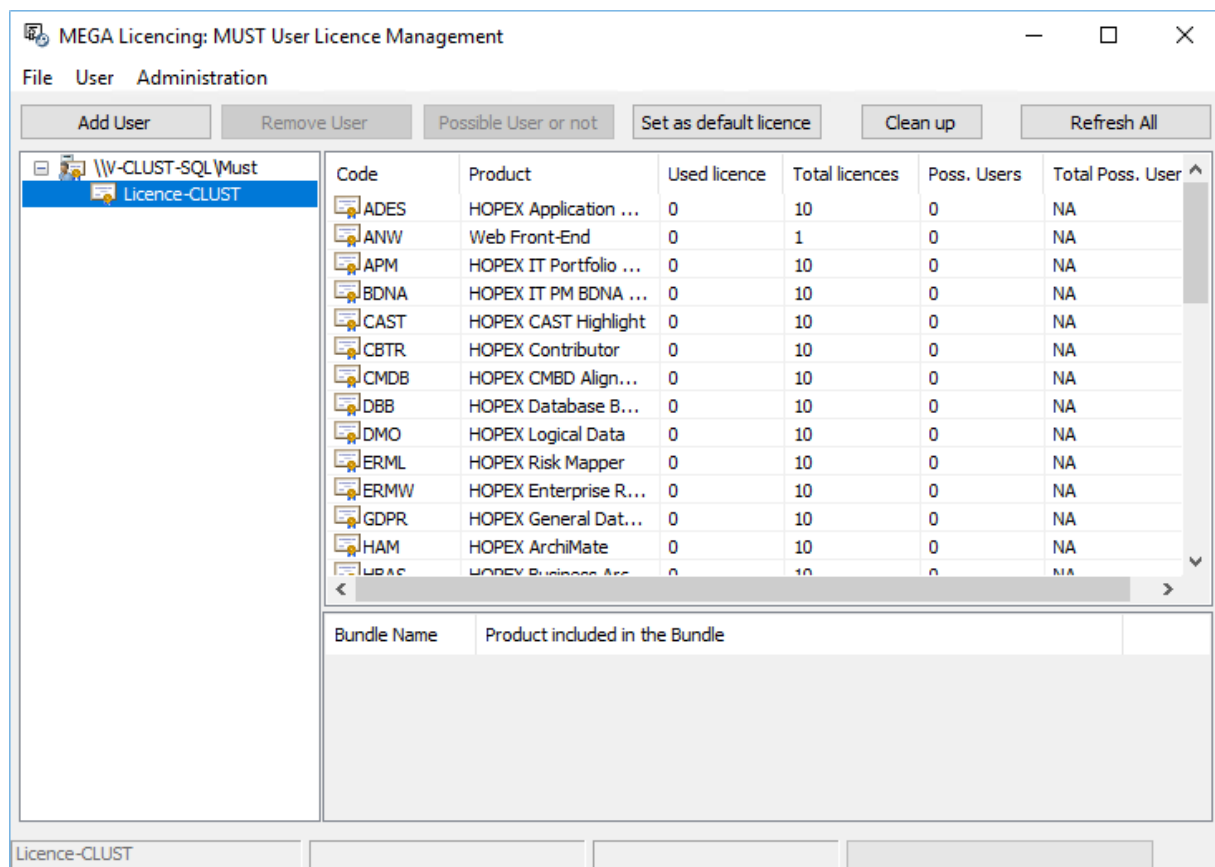
Then, go to the installation folder of Mega, and launch the « licensing.exe » tool:



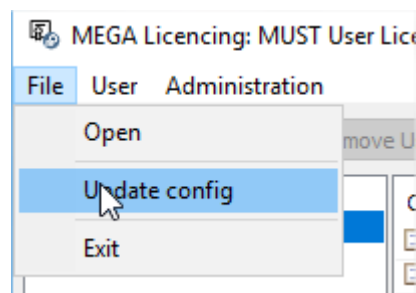
Type the UNC path and click "OK". In our case, it is [\\V-CLUST-SQL\Must](#):



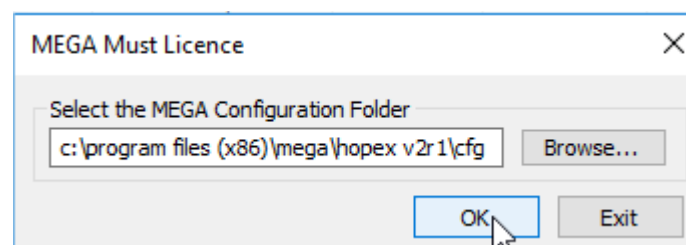
We can expand the tree to check that we have the requested products:



The click "File -> Update Config" to write this parameter in the "MegaSite.ini" file located in the "Cfg" folder of the Mega install:

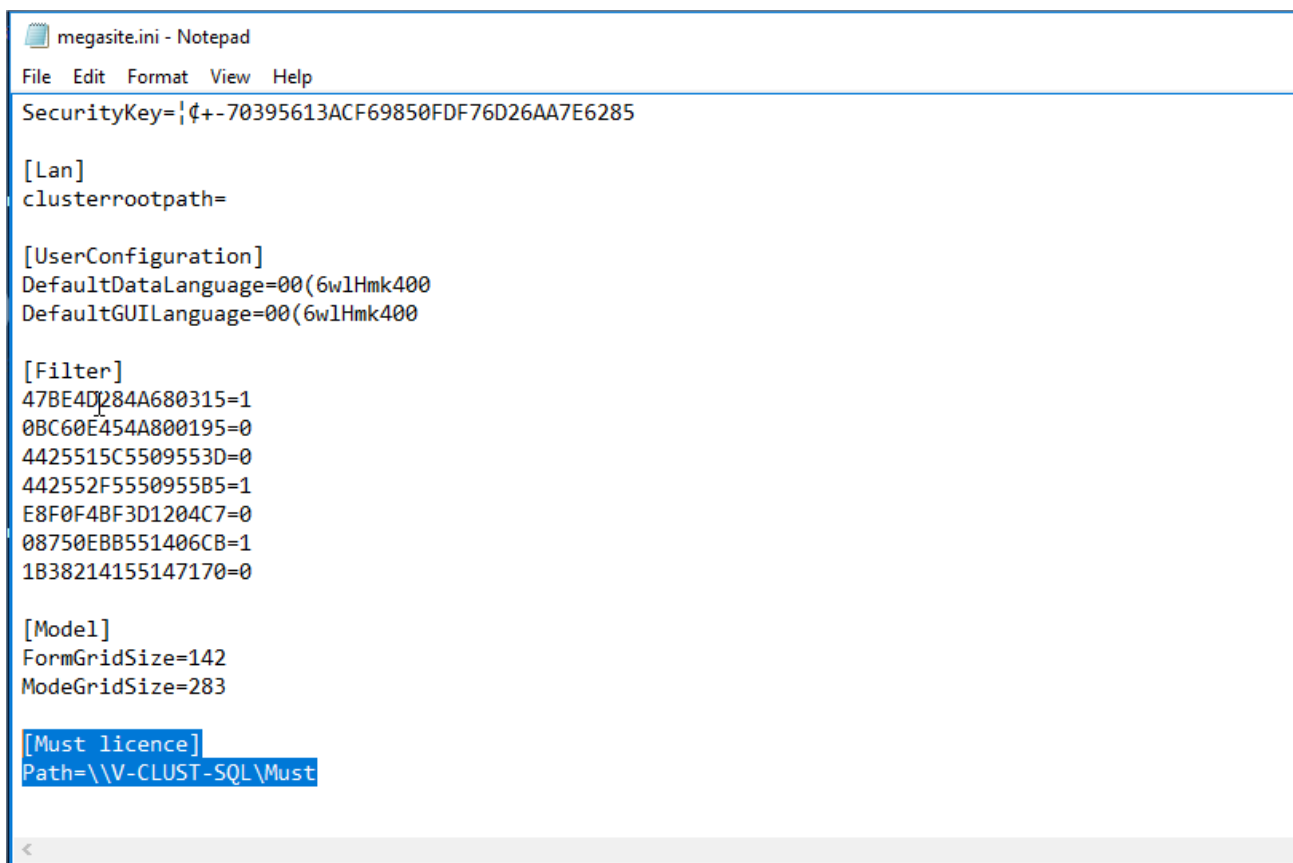
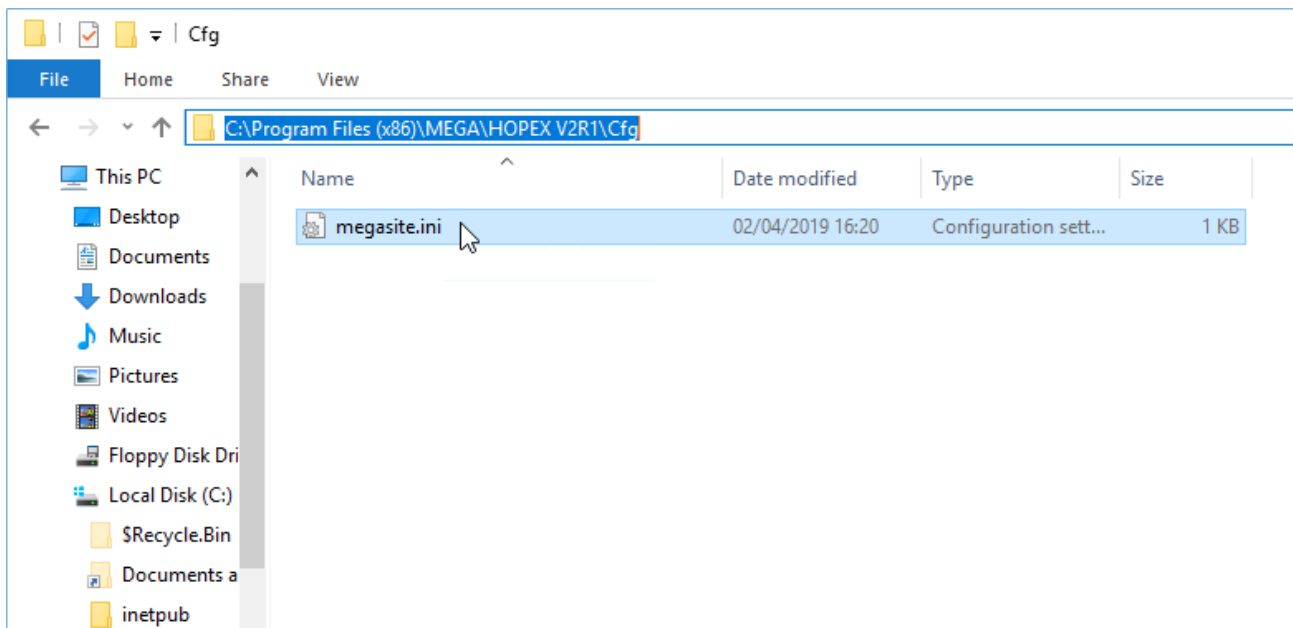


Keep the default folder, and click « OK », then exit the tool:



Close the tool.

You can check that the license information is written in the "MegaSite.ini" file in ".\Cfg" folder of the Mega binaries:



You can open "Administration.exe" to make sure that it can open:

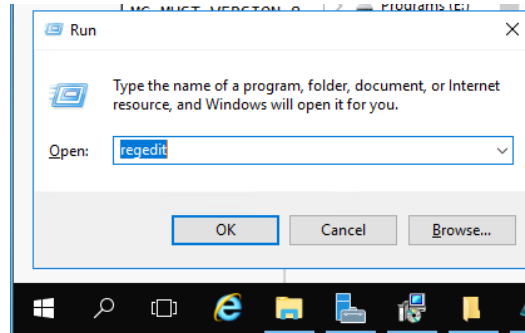
Update the Desktop Heap

To be done on the Web Servers.

This is an internal memory of Windows. It is used by the application to manage the graphical objects of the website.

We need to increase the default value of this memory size, in order for users to connect and work in the website.

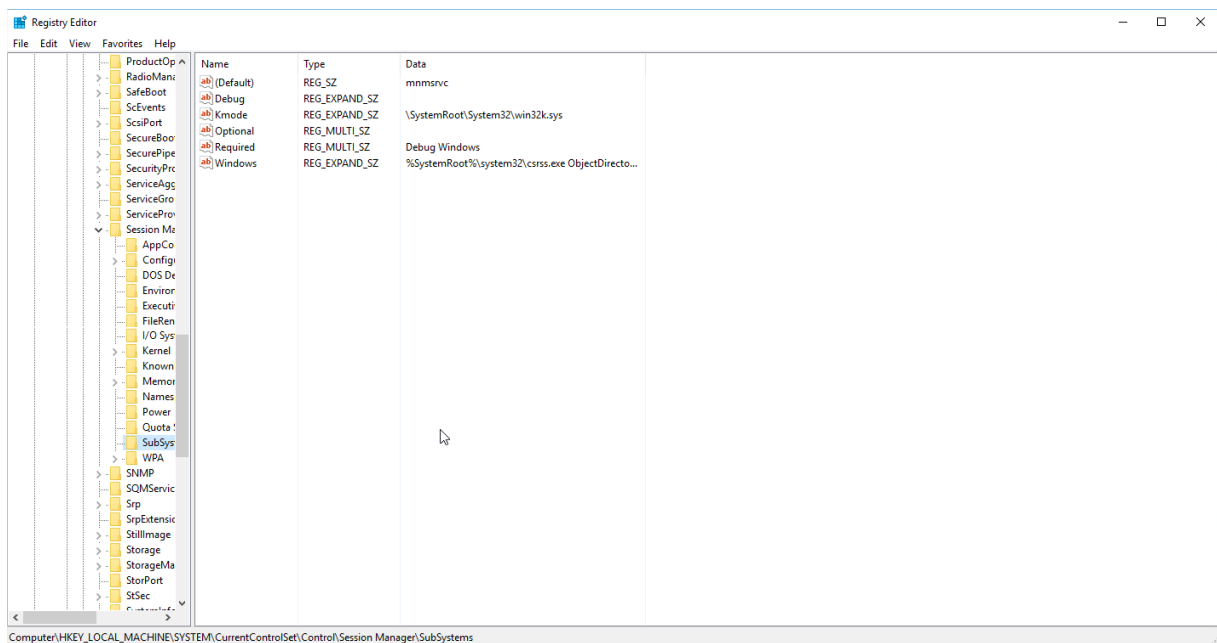
First, open the registry:



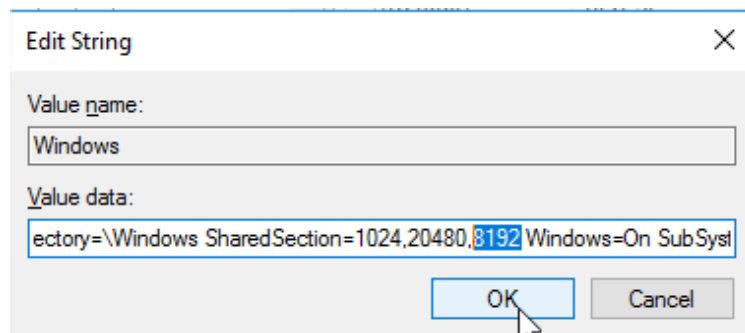
And go to the key:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\SubSystems

Then, update the "Windows" entry:



Find, in the string, the part called "SharedSection", and update the 3rd number to **8192**. The number is in KB, so that means that it will be set to 8MB:

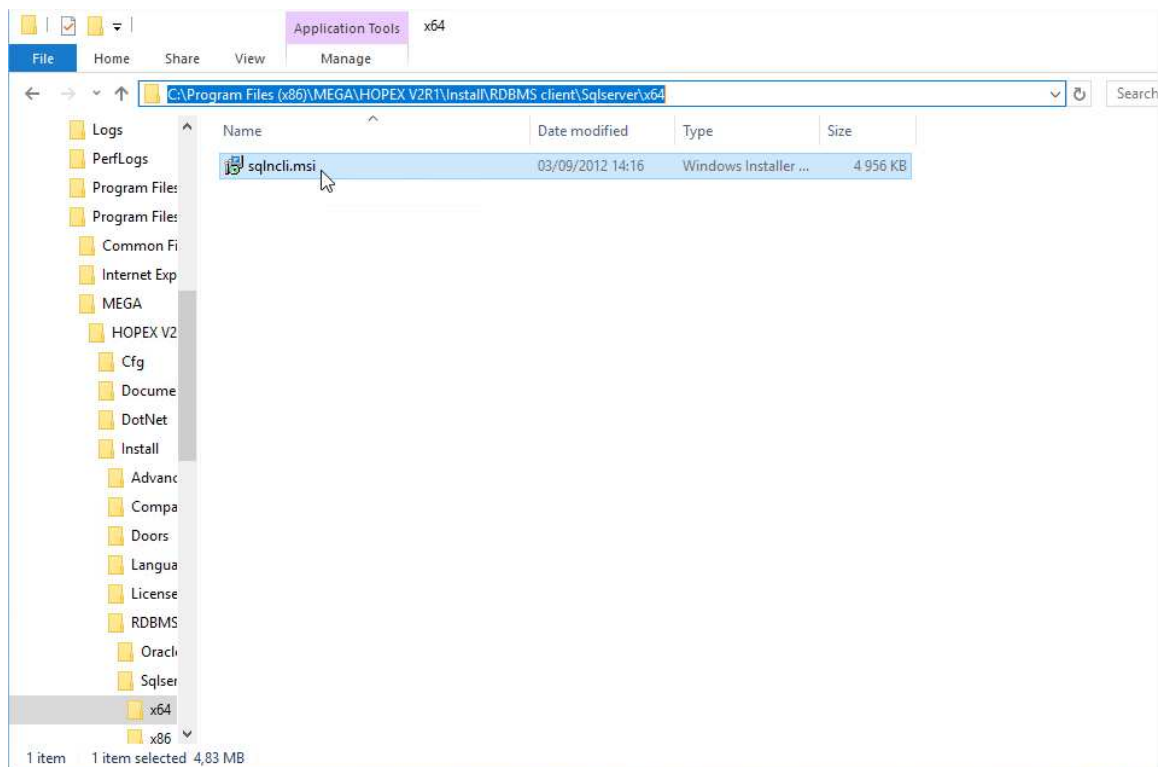


Warning ! Reboot each server you apply this configuration. Otherwise, this parameter won't be taken into account.

Install SQL Server native client

To be done on all MWAS and SSP servers. Because those are the ones that will need to make direct communication with the SQL Server databases.

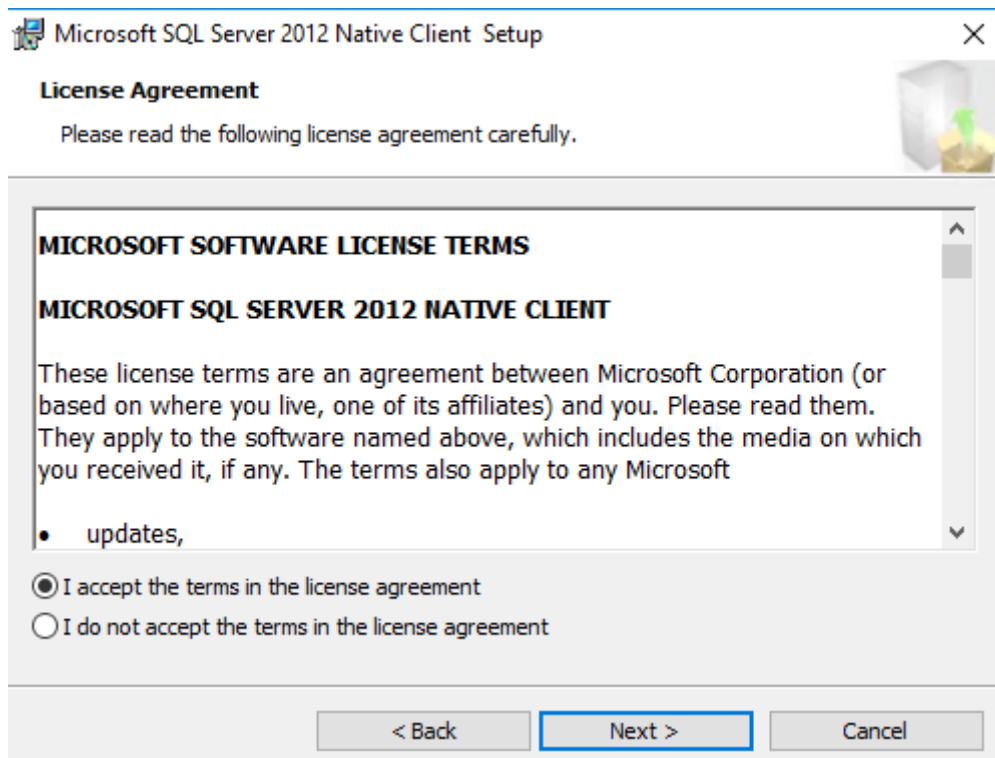
You can find the installation program in "C:\Program Files (x86)\MEGA\HOPEX V2R1\Install\RDBMS client\Sqlserver\x64". Launch it:



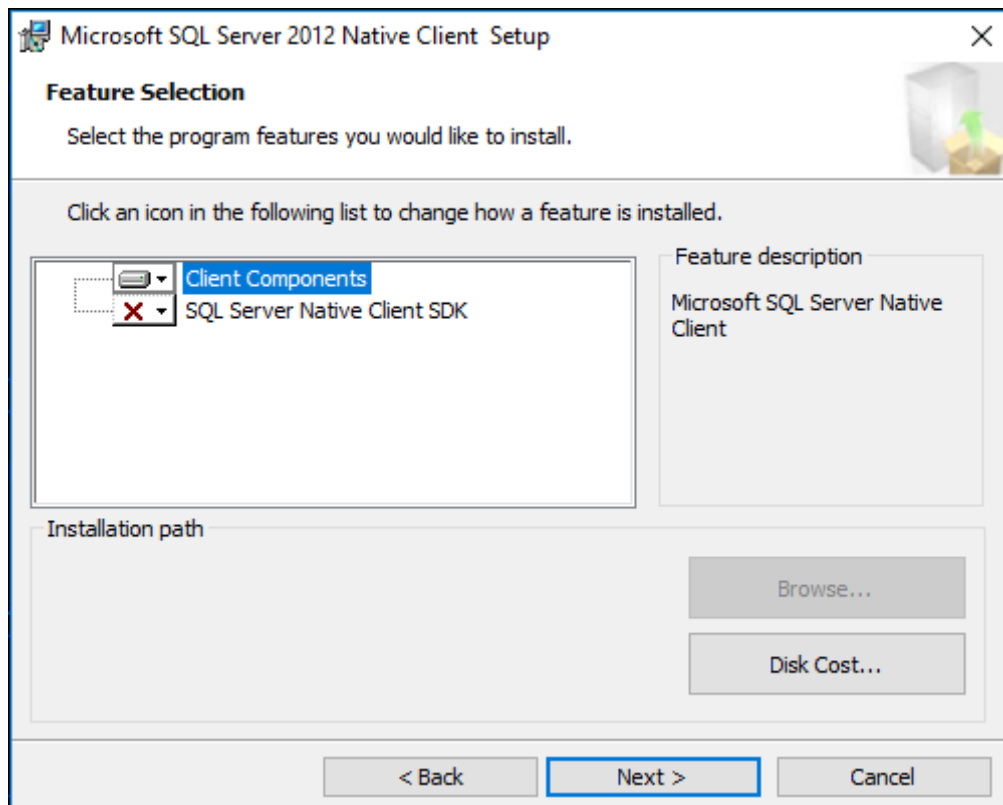
Click „Next“:



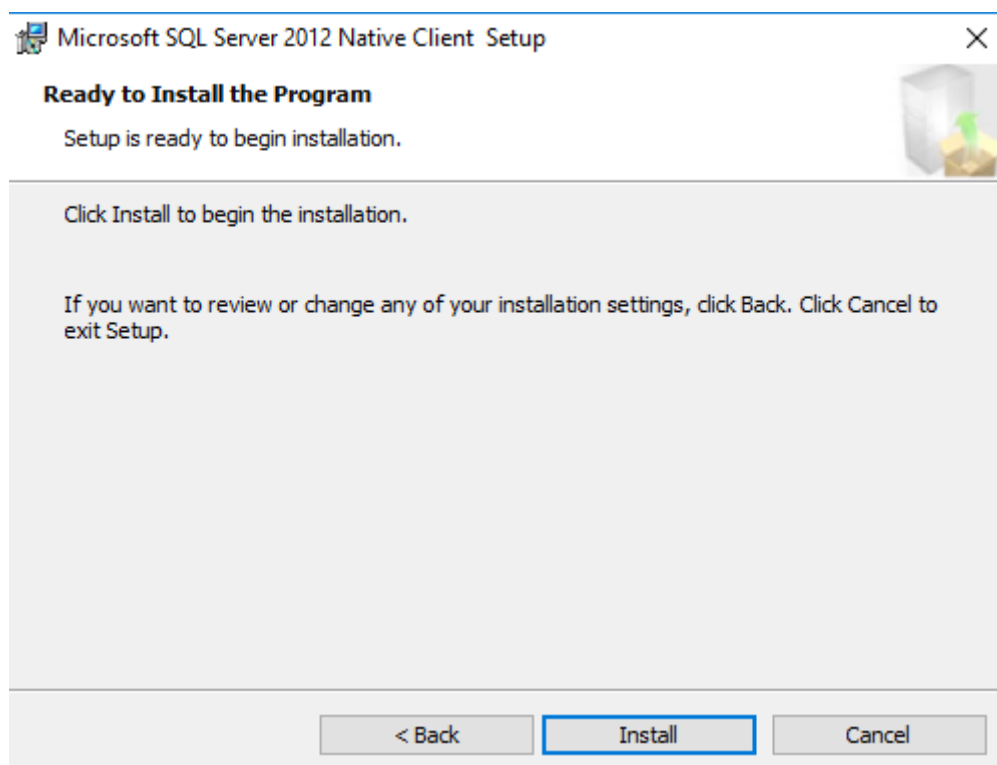
On „Next“:



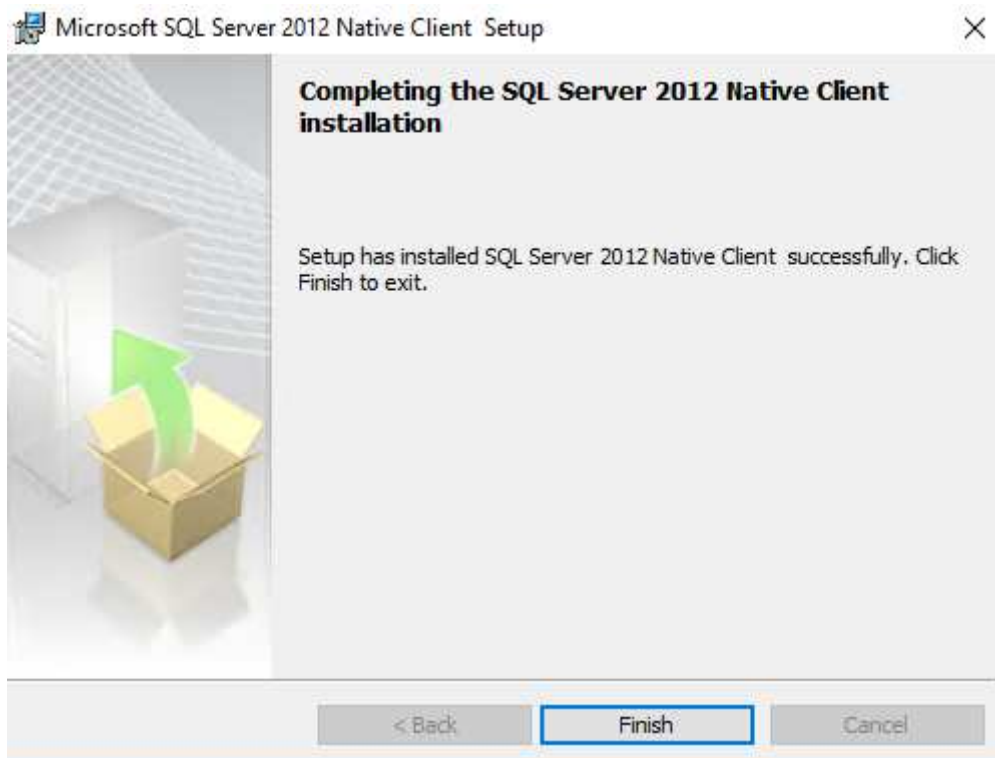
Keep the default and click „Next“:



Install:



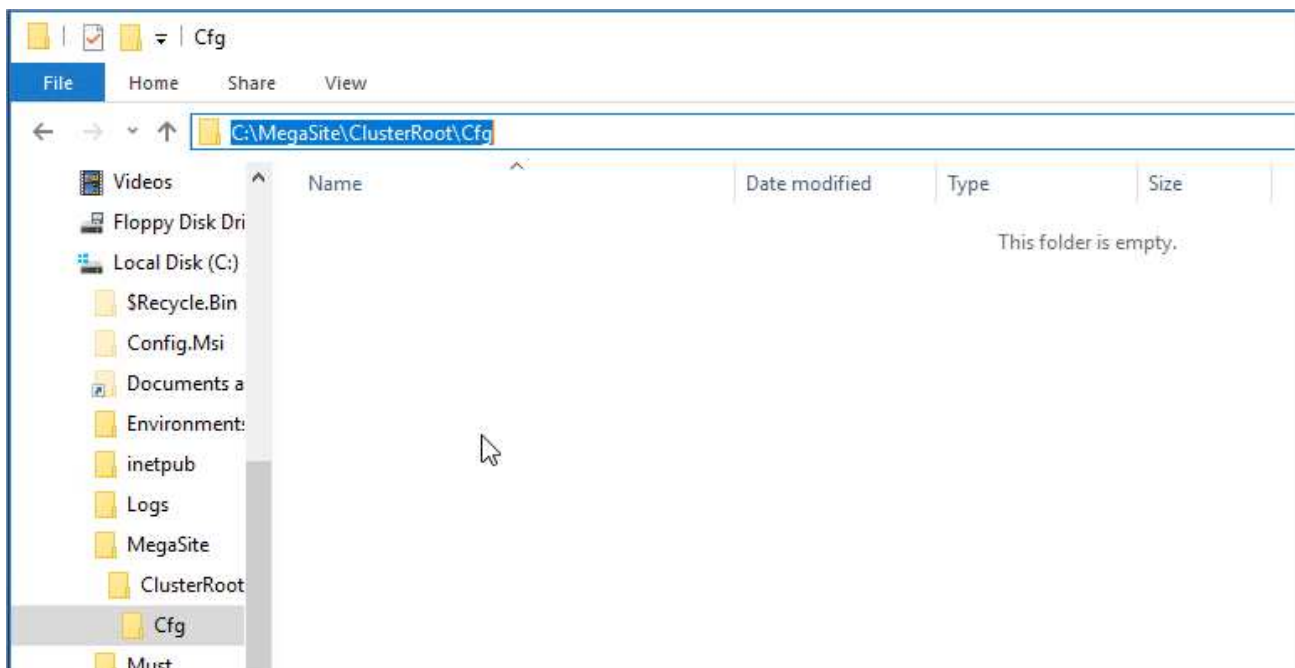
And click „Finish“:



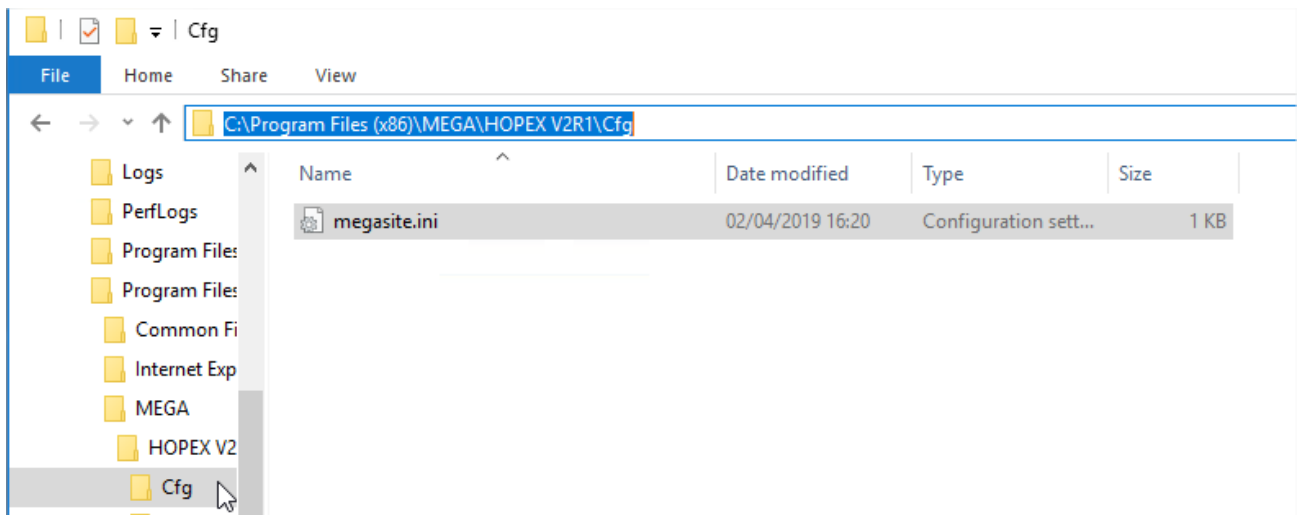
Centralize the MegaSite

Now that we have our shares, we want all servers to use the same central configuration file.

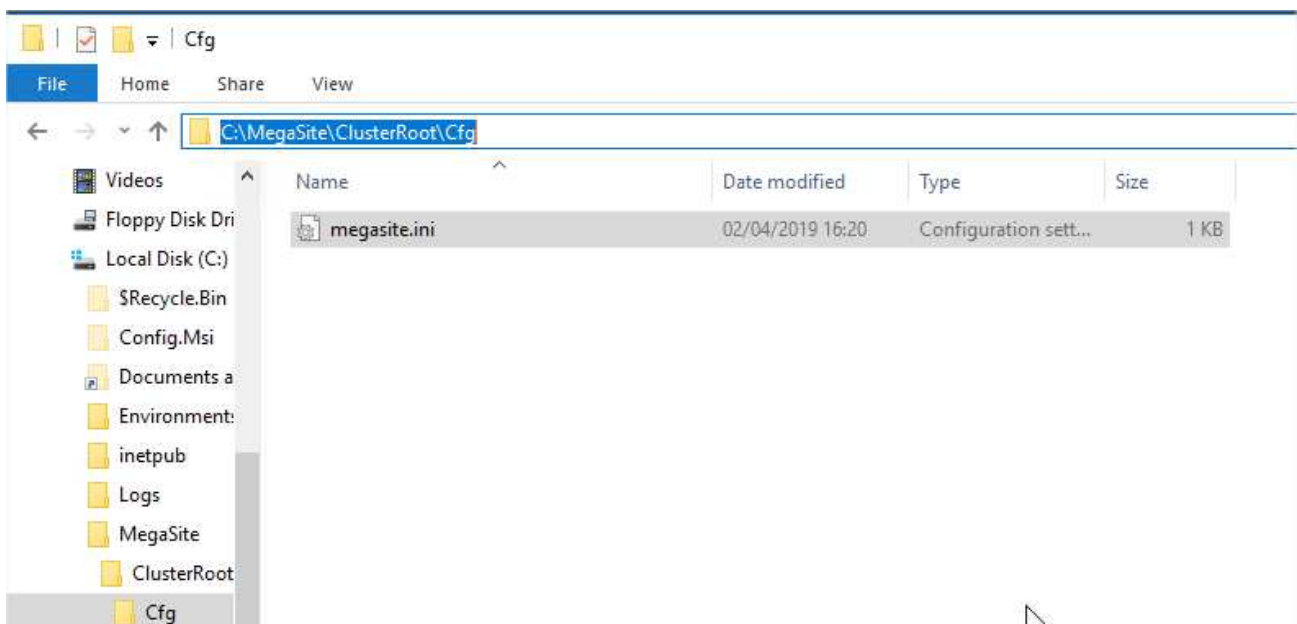
To do that, we will first create, **on the RDBMS server**, subfolders to the „C:\MegaSite“ folder. Two levels called „\ClusterRoot\Cfg“:



Then, we connect **on the first SSP server**, where we configured the licence, and copy the “MegaSite.ini” file in “C:\Program Files (x86)\MEGA\HOPEX V2R1\Cfg”:

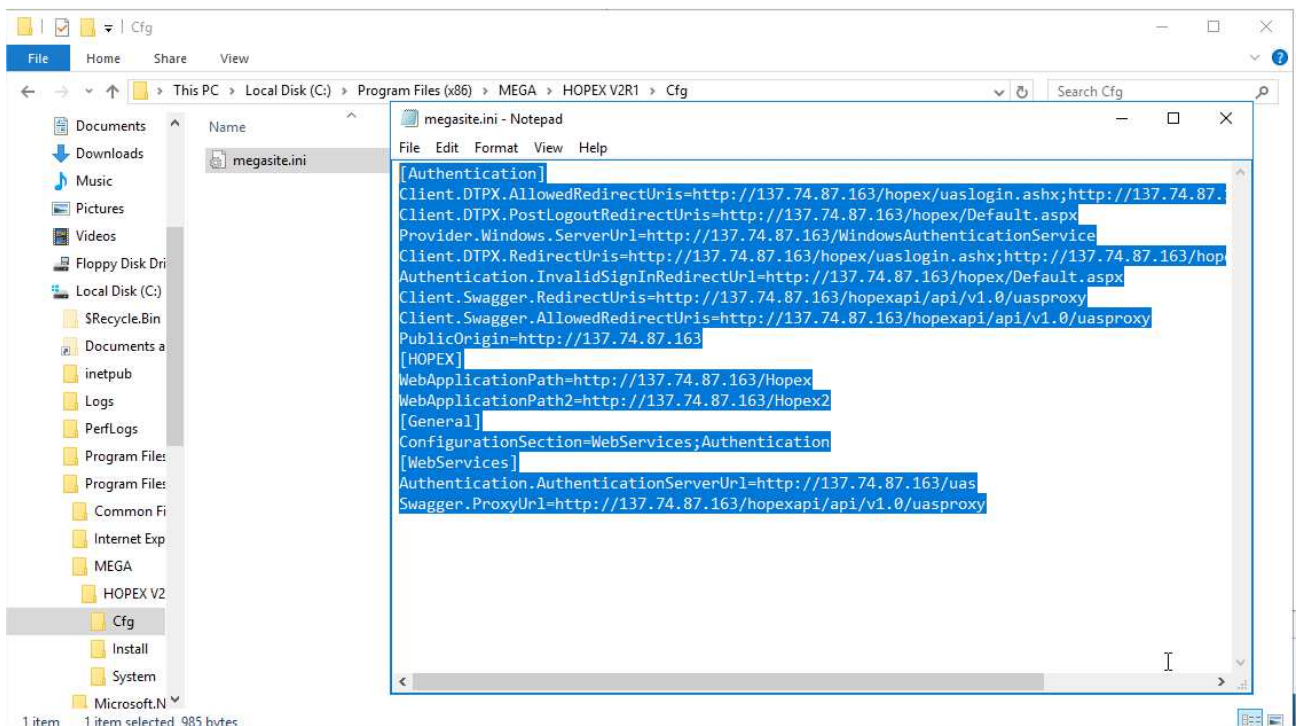


That we copy in “C:\MegaSite\ClusterRoot\Cfg” on the RDBMS Server:



After that, we need to retrieve some parts that were only written on the Web servers, linked to the UAS module to manage the authentication.

So we connect on **one of the Web servers**, and edit the “MegaSite.ini” file:



We copy all the lines.

That we paste in the file of the RDBMS Server, at the beginning of the file:



You notice that the section called [HOPEX] is duplicated. To clean this, remove the second occurrence of this section:

```
megasite.ini - Notepad
File Edit Format View Help

[Authentication]
Client.DTPX.AllowedRedirectUri=http://137.74.87.163/hopex/uaslogin.ashx;http://137.74.87.163/hopex2/uaslogin.ashx
Client.DTPX.PostLogoutRedirectUri=http://137.74.87.163/hopex/Default.aspx
Provider.Windows.ServerUrl=http://137.74.87.163/WindowsAuthenticationService
Client.DTPX.RedirectUri=http://137.74.87.163/hopex/uaslogin.ashx;http://137.74.87.163/hopex2/uaslogin.ashx
Authentication.InvalidSignInRedirectUri=http://137.74.87.163/hopex/Default.aspx
Client.Swagger.RedirectUri=http://137.74.87.163/hopexapi/api/v1.0/uasproxy
Client.Swagger.AllowedRedirectUri=http://137.74.87.163/hopexapi/api/v1.0/uasproxy
PublicOrigin=http://137.74.87.163
[HOPEX]
WebApplicationPath=http://137.74.87.163/Hopex
WebApplicationPath2=http://137.74.87.163/Hopex2
[General]
ConfigurationSection=WebServices;Authentication
[WebServices]
Authentication.AuthenticationServerUrl=http://137.74.87.163/uas
Swagger.ProxyUrl=http://137.74.87.163/hopexapi/api/v1.0/uasproxy

[System]
MegaCurrentVersion=30464
Language=US

[General]
Company=MEGA
SiteName=TEMPLATE

[SSP]
Url=http://137.74.87.169/MegaSSP
SecurityKey=|f+-70395613ACF69850FDF76D26AA7E6285

[Lan]
clusterrootpath=

[UserConfiguration]
DefaultDataLanguage=00(6w1Hmk400
DefaultGUILanguage=00(6w1Hmk400
```

The [General] section is also showing in two parts now. You can transfer one line from the first part of the file to the second section, and remove the first one:

```
megasite.ini - Notepad
File Edit Format View Help

[Authentication]
Client.DTPX.AllowedRedirectUri=http://137.74.87.163/hopex/uaslogin.ashx;http://137.74.87.163/hopex2/uaslogin.ashx
Client.DTPX.PostLogoutRedirectUri=http://137.74.87.163/hopex/Default.aspx
Provider.Windows.ServerUrl=http://137.74.87.163/WindowsAuthenticationService
Client.DTPX.RedirectUri=http://137.74.87.163/hopex/uaslogin.ashx;http://137.74.87.163/hopex2/uaslogin.ashx
Authentication.InvalidSignInRedirectUri=http://137.74.87.163/hopex/Default.aspx
Client.Swagger.RedirectUri=http://137.74.87.163/hopexapi/api/v1.0/uasproxy
Client.Swagger.AllowedRedirectUri=http://137.74.87.163/hopexapi/api/v1.0/uasproxy
PublicOrigin=http://137.74.87.163

[HOPEX]
WebApplicationPath=http://137.74.87.163/Hopex
WebApplicationPath2=http://137.74.87.163/Hopex2

[WebServices]
Authentication.AuthenticationServerUrl=http://137.74.87.163/uas
Swagger.ProxyUrl=http://137.74.87.163/hopexapi/api/v1.0/uasproxy

[System]
MegaCurrentVersion=30464
Language=US

[General]
Company=MEGA
SiteName=TEMPLATE
ConfigurationSection=WebServices;Authentication

[SSP]
Url=http://137.74.87.169/MegaSSP
SecurityKey=|f+-70395613ACF69850FDF76D26AA7E6285

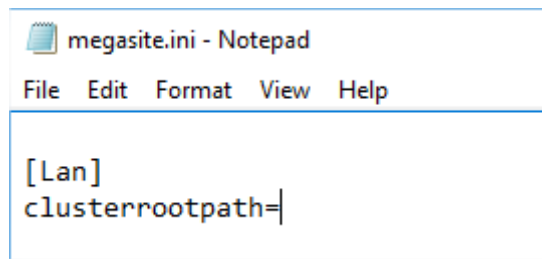
[Lan]
clusterrootpath=

[UserConfiguration]
DefaultDataLanguage=00(6w1Hmk400
DefaultGUILanguage=00(6w1Hmk400
```

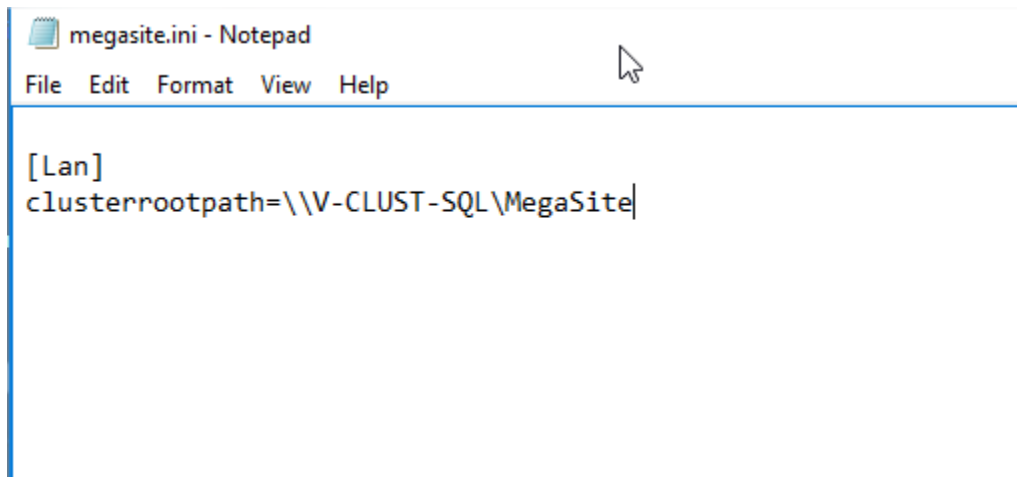
Save and close the file.

Lastly, **on all MWAS and SSP servers**, we edit the file located in “C:\Program Files (x86)\MEGA\HOPEX V2R1\Cfg”.

We remove everything except the [Lan] section:

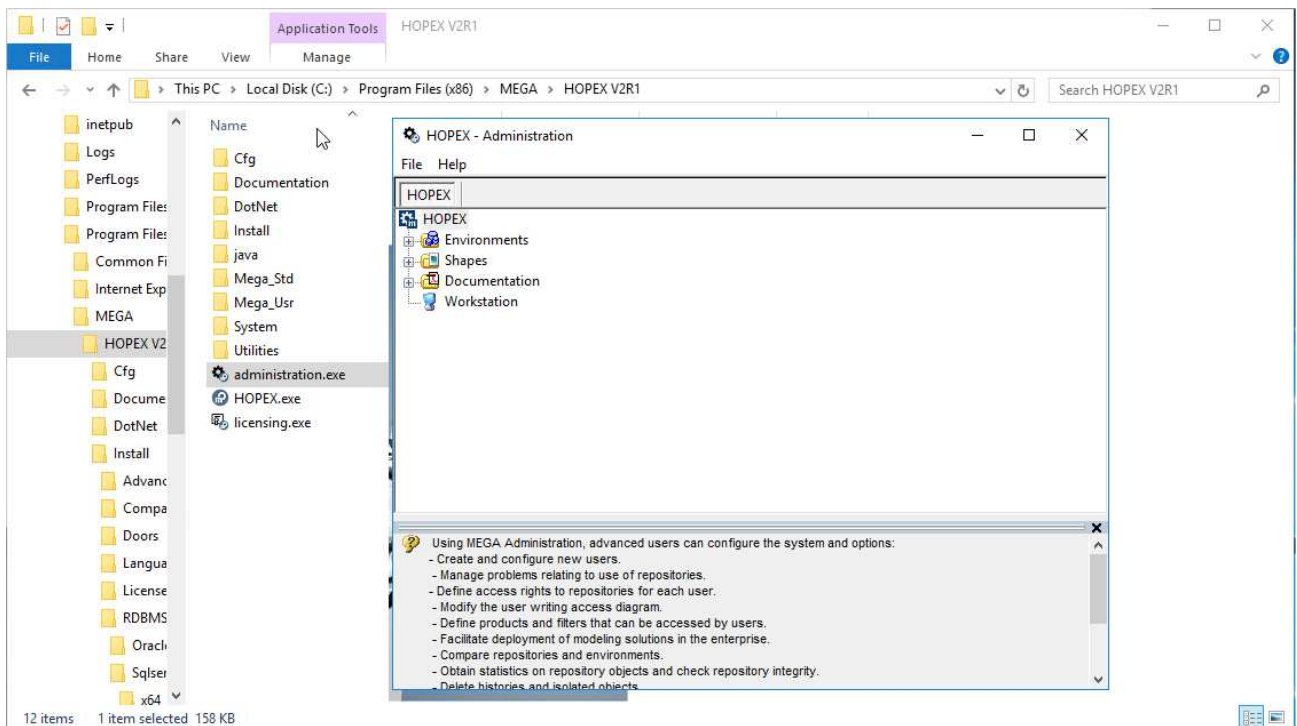


And we give the “clusterrootpath” value the path of the “MegaSite” share, so in our case [\\V-CLUST-SQL\MegaSite](#) :



Do not provide the full path. The application will look, inside that share, in “\ClusterRoot\Cfg”, to locate the MegaSite.ini file.

We save the file. And on each server, to make sure that the configuration is valid, we open “administration.exe”. If no error message appears, it means it was able to reach the centralized file and read its configuration:



Managing the RDBMS setup

Instance and databases

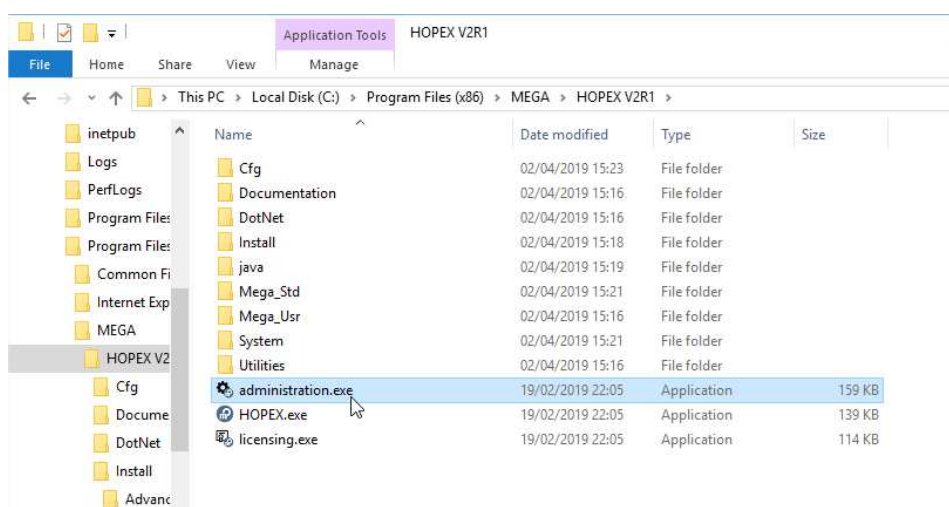
We use SQL Server as our RDBMS and data storage.

The instance name is “V-CLUST-SQL,1436”.

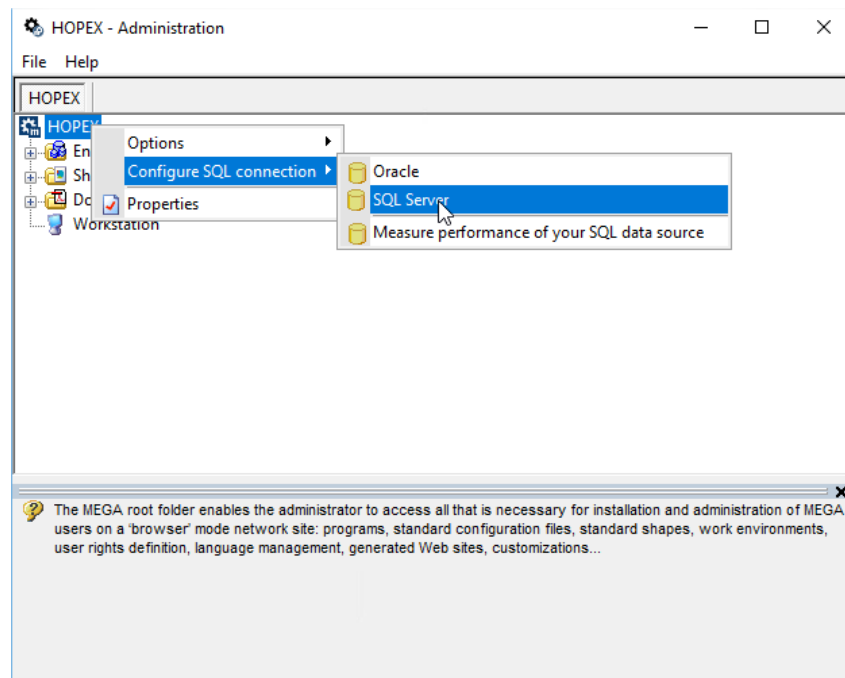
We use a SQL Server native account called “MEGAUSR”. Password is shared to the people that can use this platform to deploy the solution. It has “dbcreator” right at the instance level, as well as the securable “view server state”. Thus, it will be able to create new databases through Hopex.

Configure the default connection string

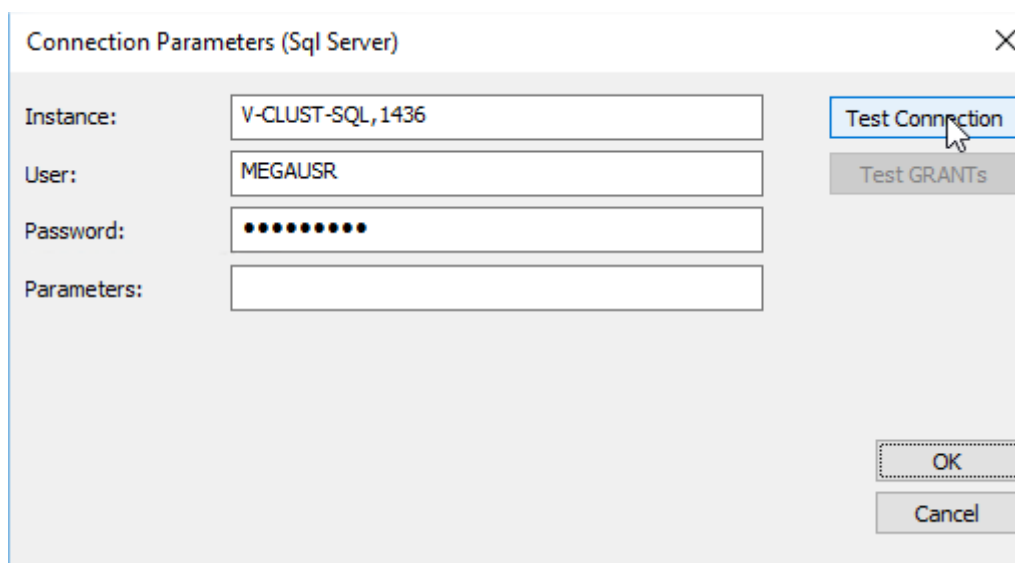
From one of the application servers (it does not matter which), open “administration.exe”:



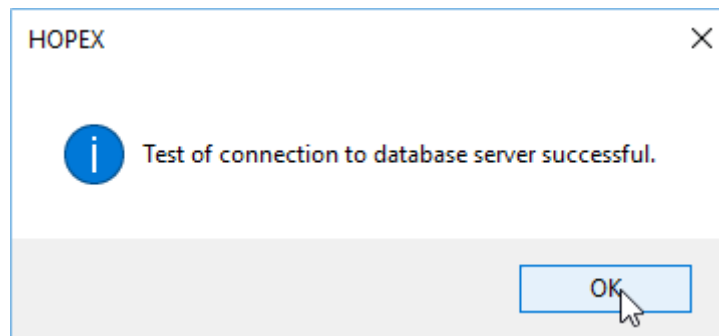
Then, open the “SQL Server” option from “HOPEX->Configure SQL Connection”:



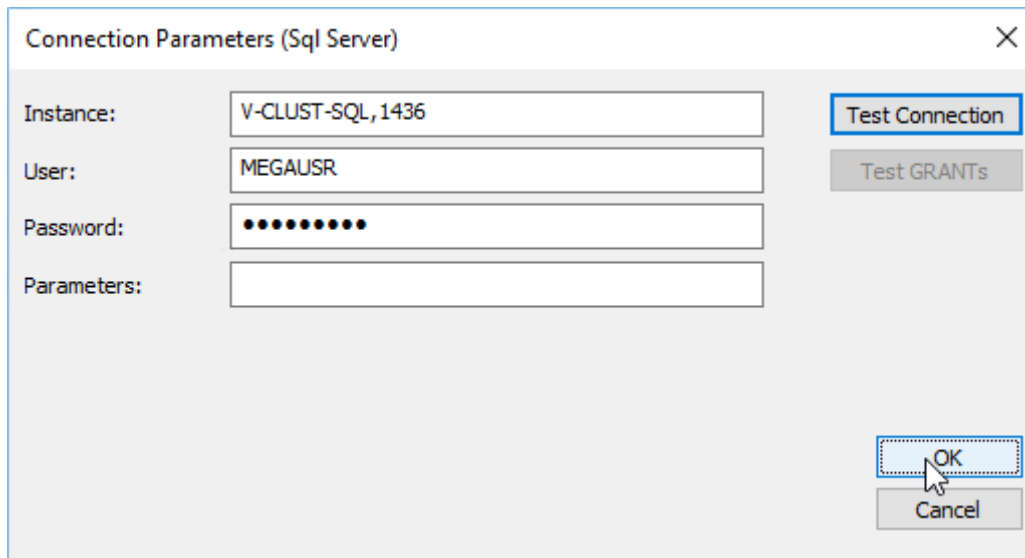
Provide the instance name, and the credentials of the „MEGAUSR“ native account. Then click „Test Connection“:



Click „OK“:

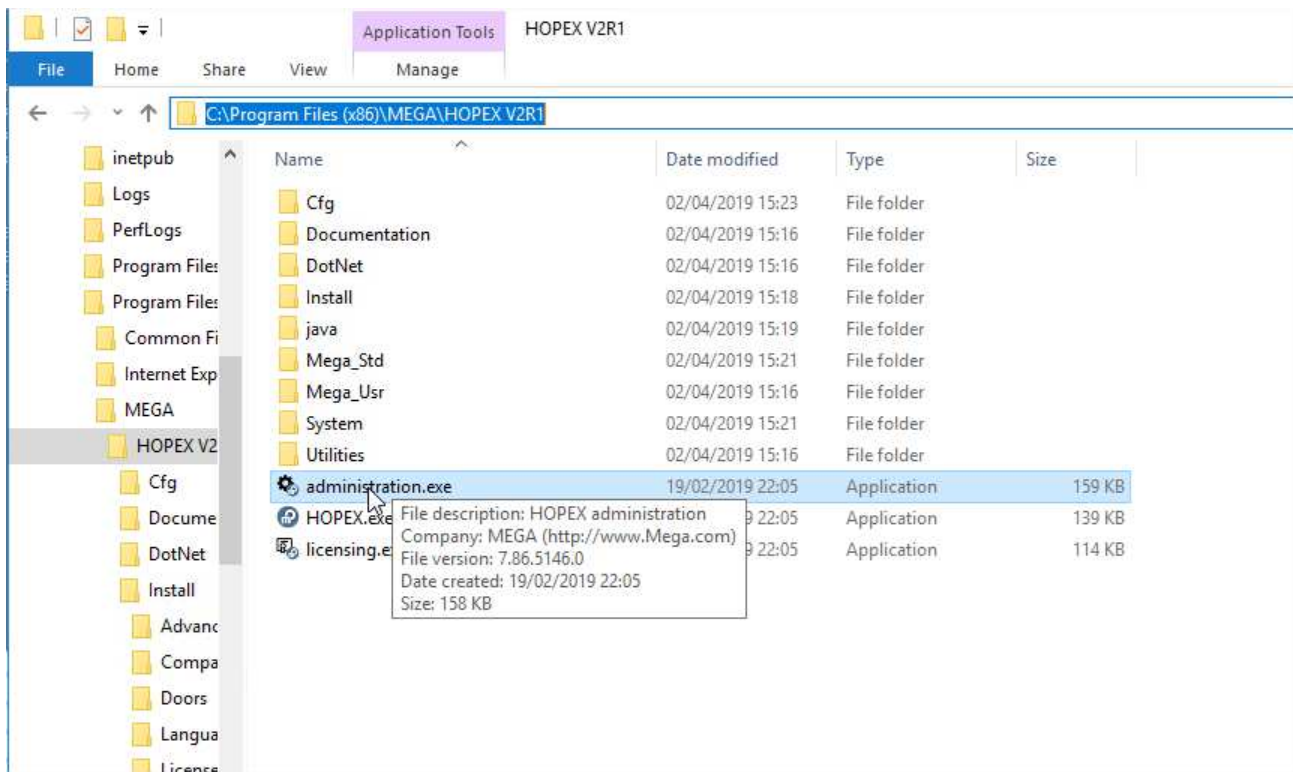


And on „OK“ to validate this configuration:

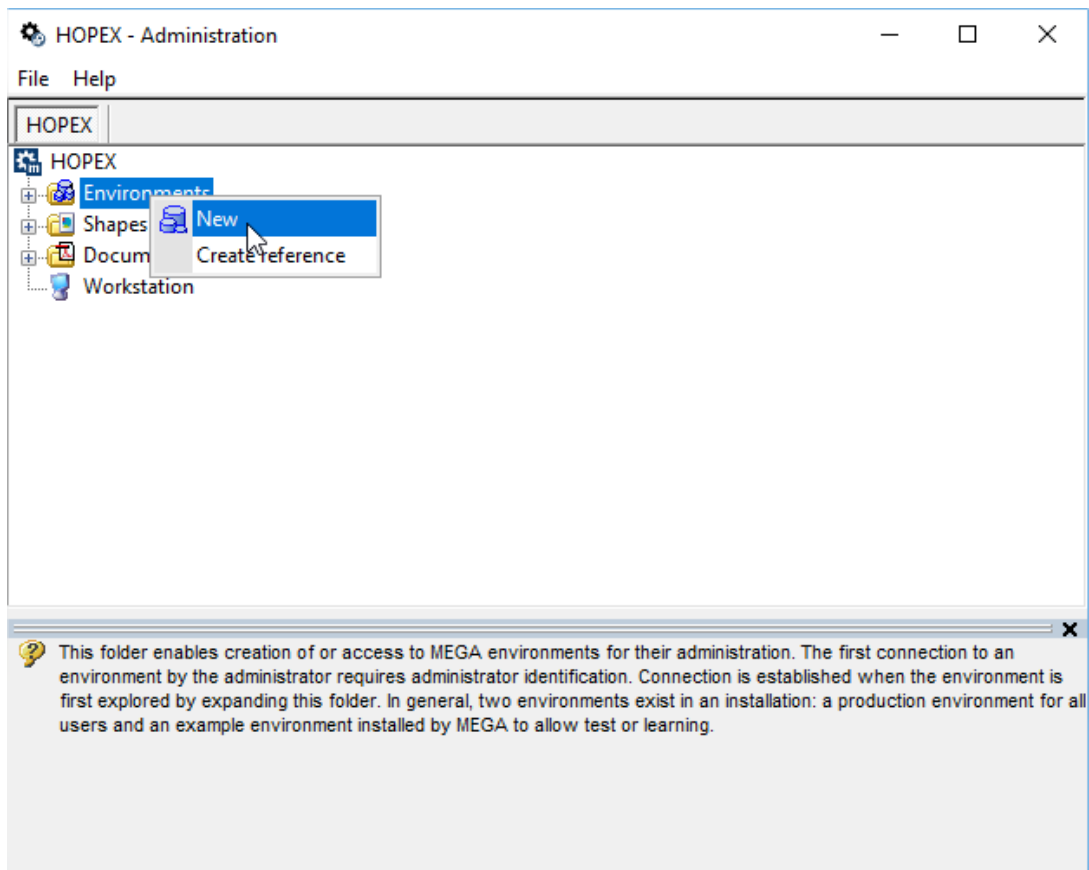


Creating an environment

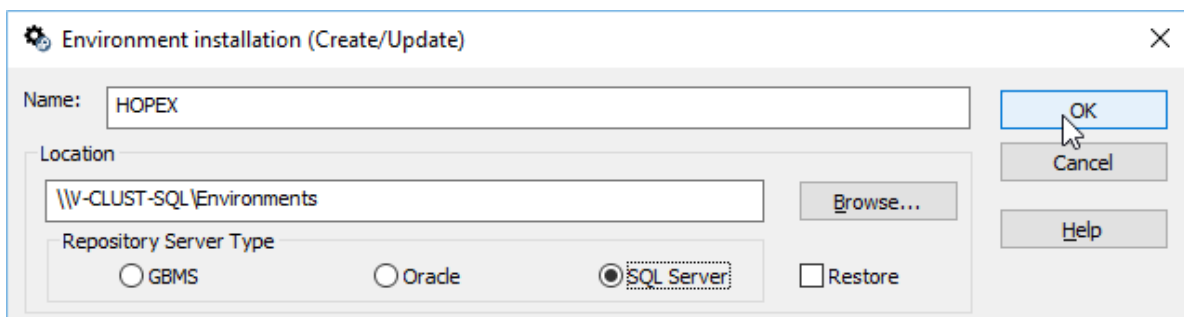
Open the “Administration.exe” tool **from one of application servers**:



Right-click “Environments” and chose “New”:



Put the name of the environment (in our case “HOPEX”), the location of the folder ([\\V-CLUST-SQL\Environments](#)) to be created, its type (SQL Server), and “OK”:



All parameters are set by default, click “Test Connection”:

Connection Parameters (SQL Server) Repository: "SystemDb"

Instance:

User:

Password:

Parameters:

Repository creation mode
Creates the SQL database ("dbo" default schema)

SQL Server:

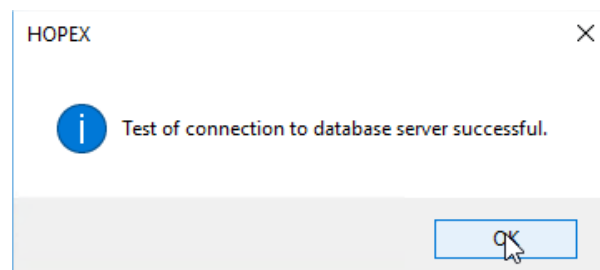
SQL Server schema:

Test Connection

Test GRANTs

OK

Cancel



Click "OK" and then click "Test GRANTs":

Connection Parameters (SQL Server) Repository: "SystemDb"

Instance:

User:

Password:

Parameters:

Repository creation mode
Creates the SQL database ("dbo" default schema)

SQL Server:

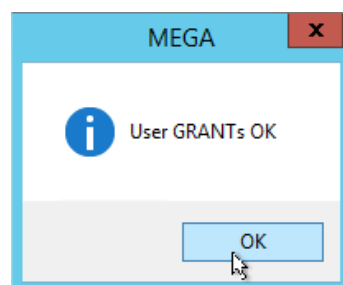
SQL Server schema:

Test Connection

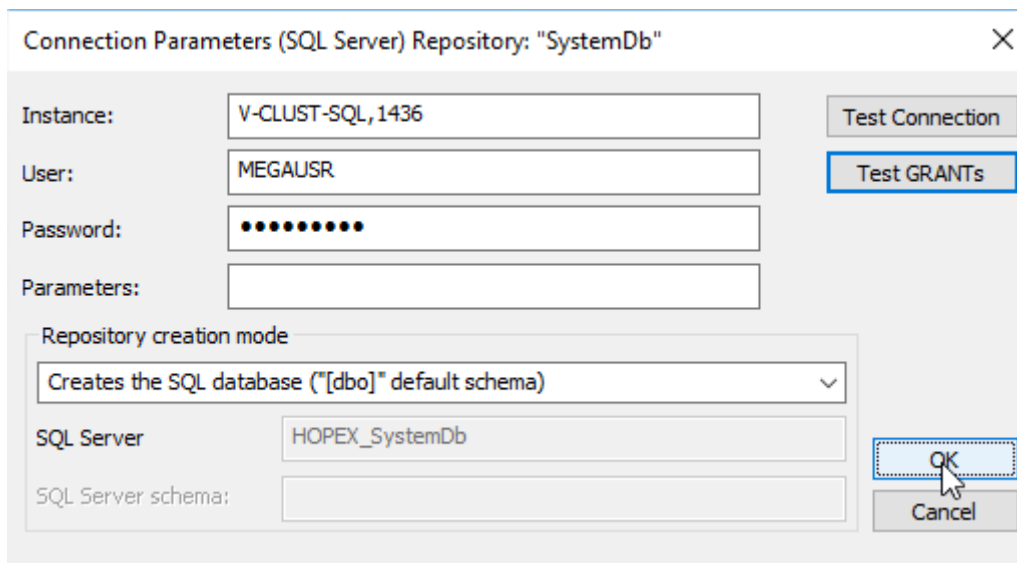
Test GRANTs

OK

Cancel



Click "OK" to start the creation:



Connection Parameters (SQL Server) Repository: "SystemDb"

Instance: V-CLUST-SQL, 1436

User: MEGAUSR

Password: ●●●●●●●●

Parameters:

Repository creation mode
Creates the SQL database ("dbo" default schema)

SQL Server: HOPEX_SystemDb

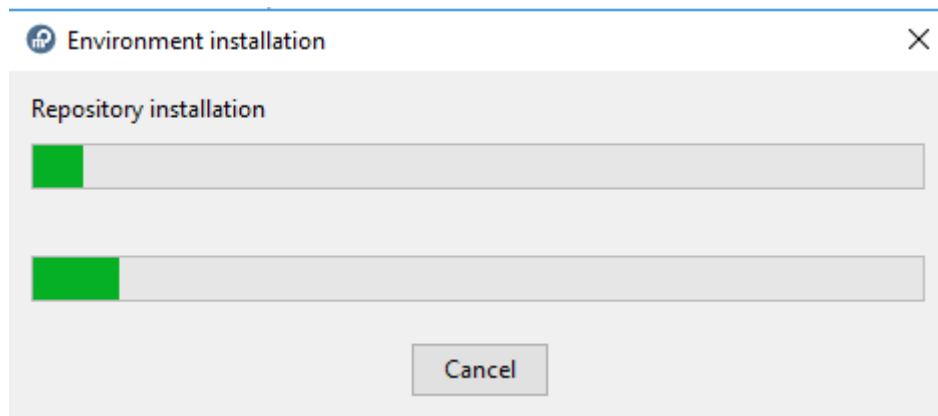
SQL Server schema:

Test Connection

Test GRANTS

OK

Cancel



Environment installation

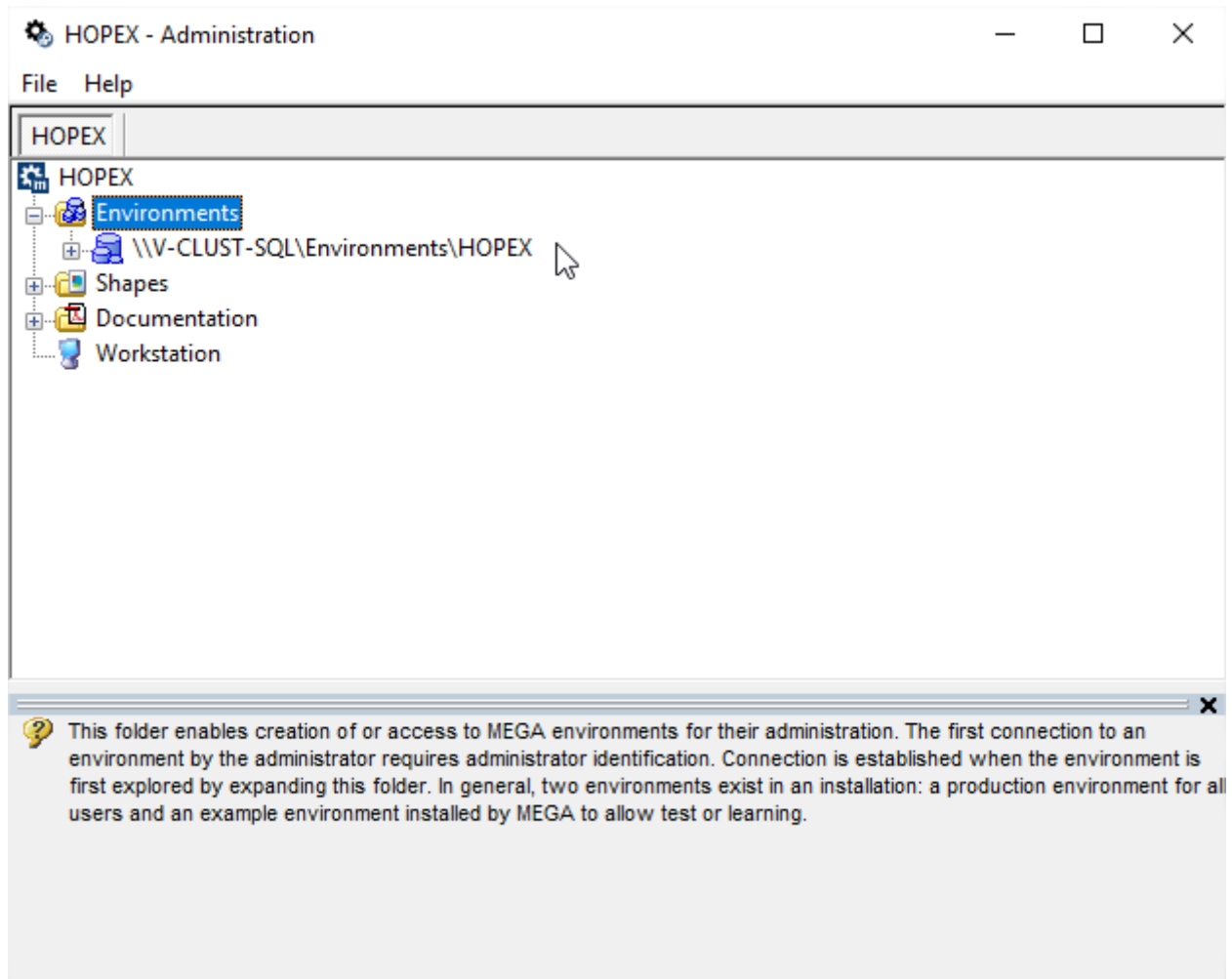
Repository installation

Progress bars showing installation progress.

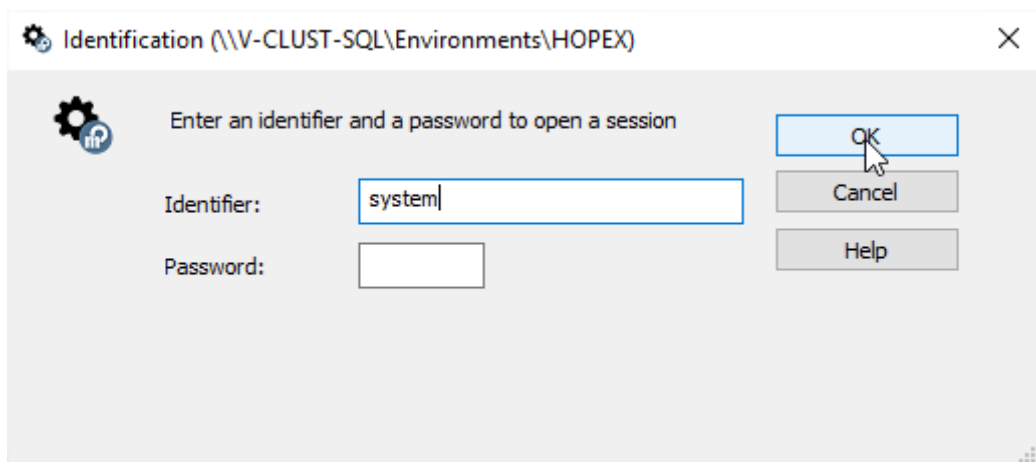
Cancel

Started around 7:15pm on the 2nd of April. Finished at 11:14pm.

The environment is created:



Connect to it by clicking on the ,+' in front of the environment name, with the « system » account :

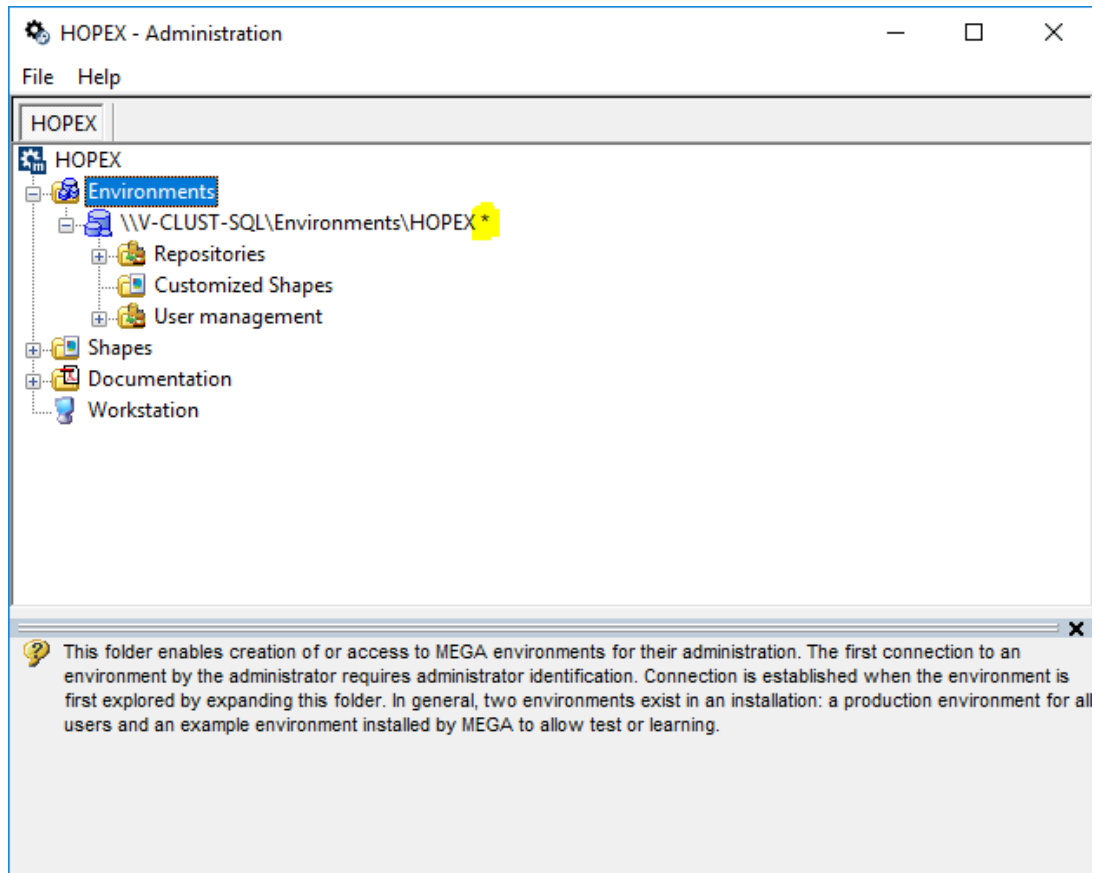


Two accounts are created by default on every new environment:

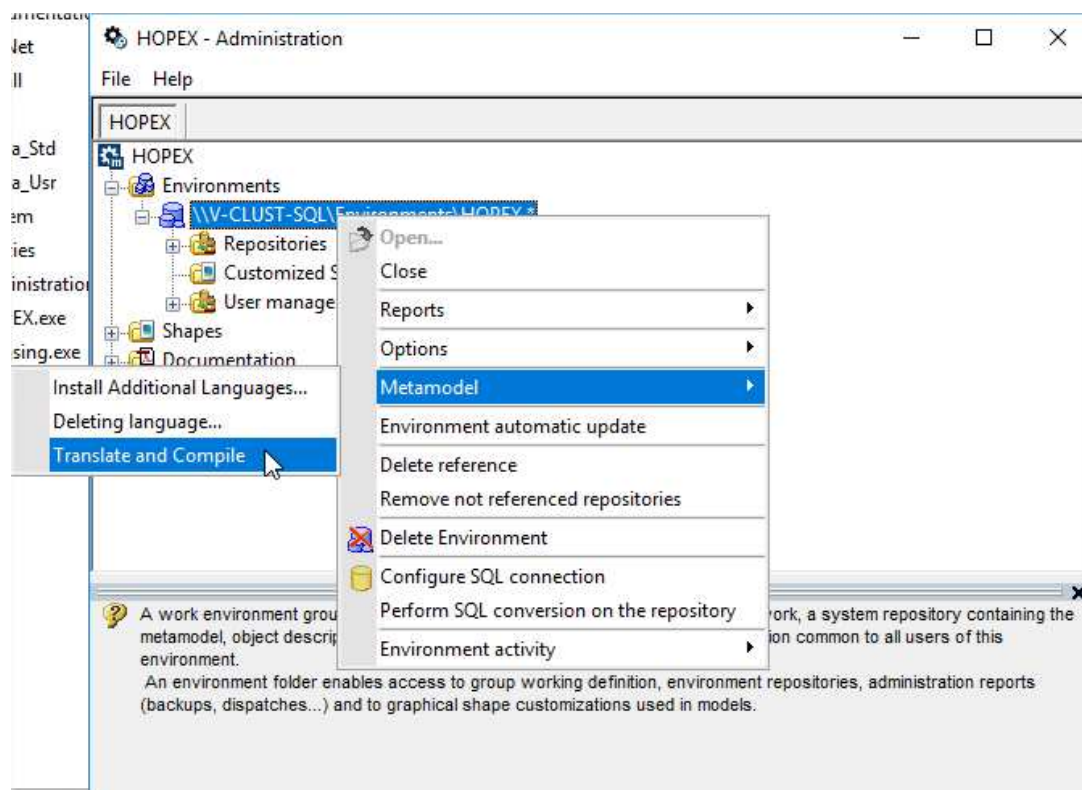
- "System" account (no password) allowing a connection to the environment in the Administration module.

- “Mega” account (no password) allowing to connect to the rich client and the web client, with basic assignments.

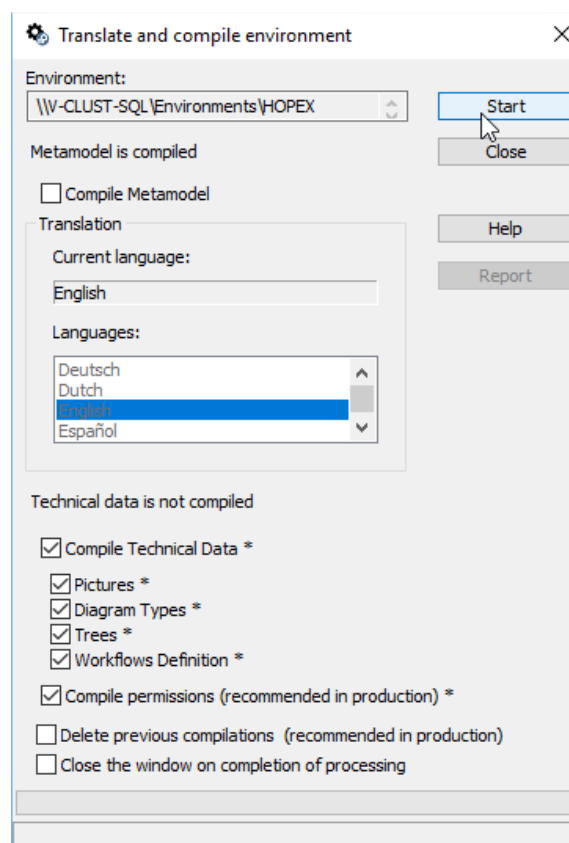
You can see a star at the end of the environment name, showing that some compilation is missing. This is normal, we don’t compile the technical data at creation:



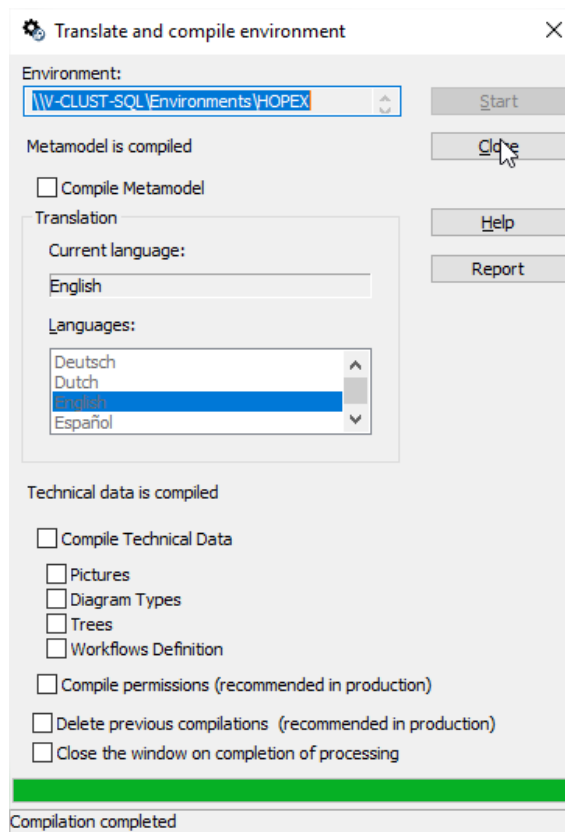
Right-click the environment, choose “Metamodel->Translate and Compile”:



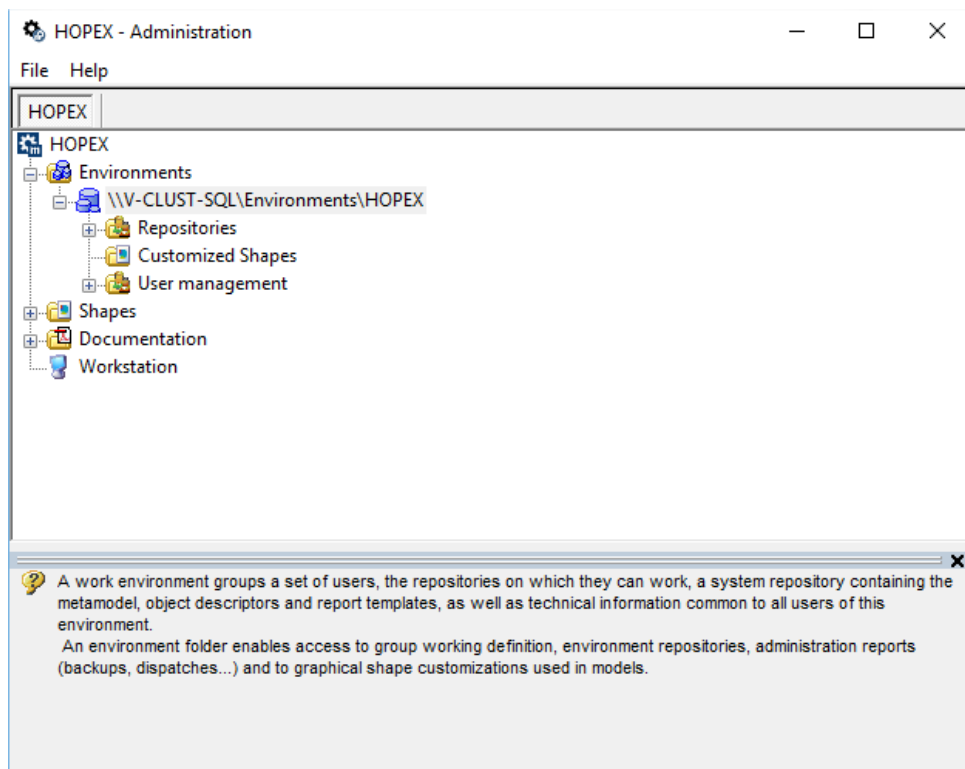
Clear the last box and click « Start » :



Let it run, and when finished, click « Close » :



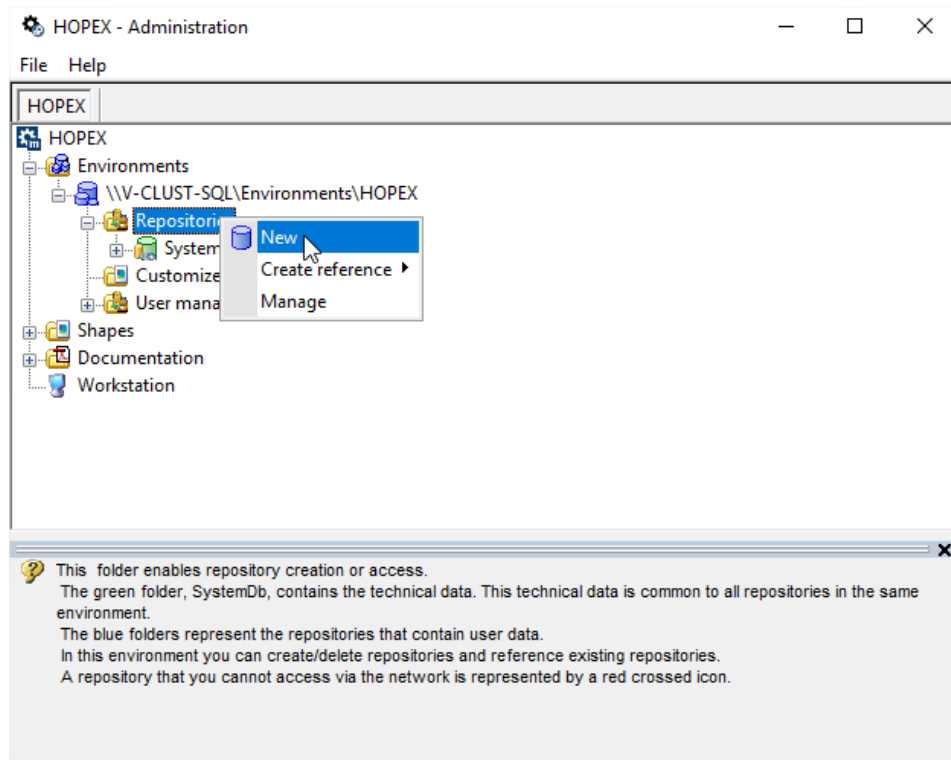
And the '*' at the end of the environment name will be gone :



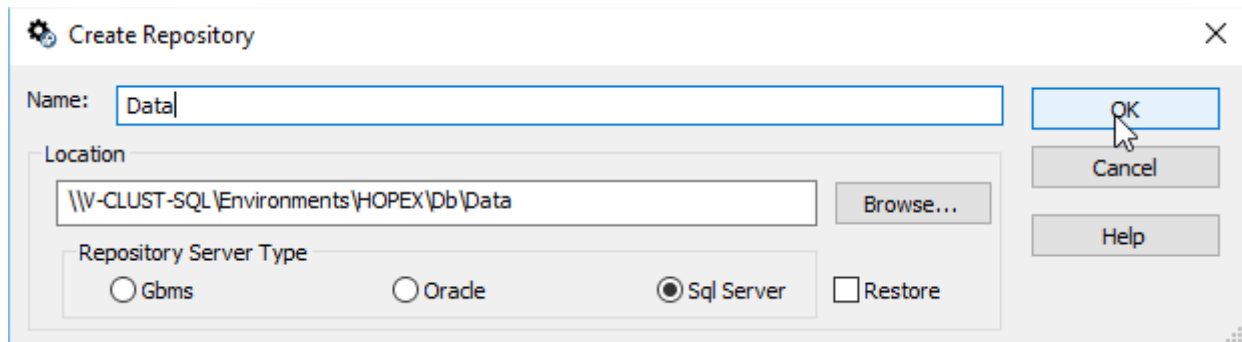
Creating a repository

We now create at least one repository in the environment.

Select “New”:



Enter the name of the repository. Here, “Data”, check that “SQL Server” is the chosen storage, and click “OK”:



Keep the default settings and pass the two tests :

Connection Parameters (SQL Server) Repository: "Data" ✕

Instance: Test Connection

User: Test GRANTS

Password:

Parameters:

Repository creation mode

▼

SQL Server

SQL Server schema:

OK Cancel

HOPEX ✕

i Test of connection to database server successful.

OK

Connection Parameters (SQL Server) Repository: "Data" ✕

Instance: Test Connection

User: Test GRANTS

Password:

Parameters:

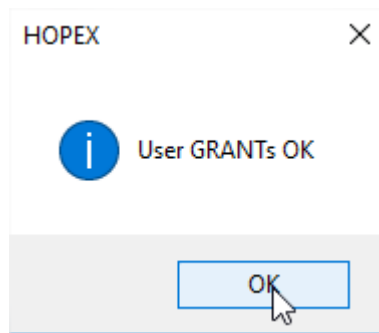
Repository creation mode

▼

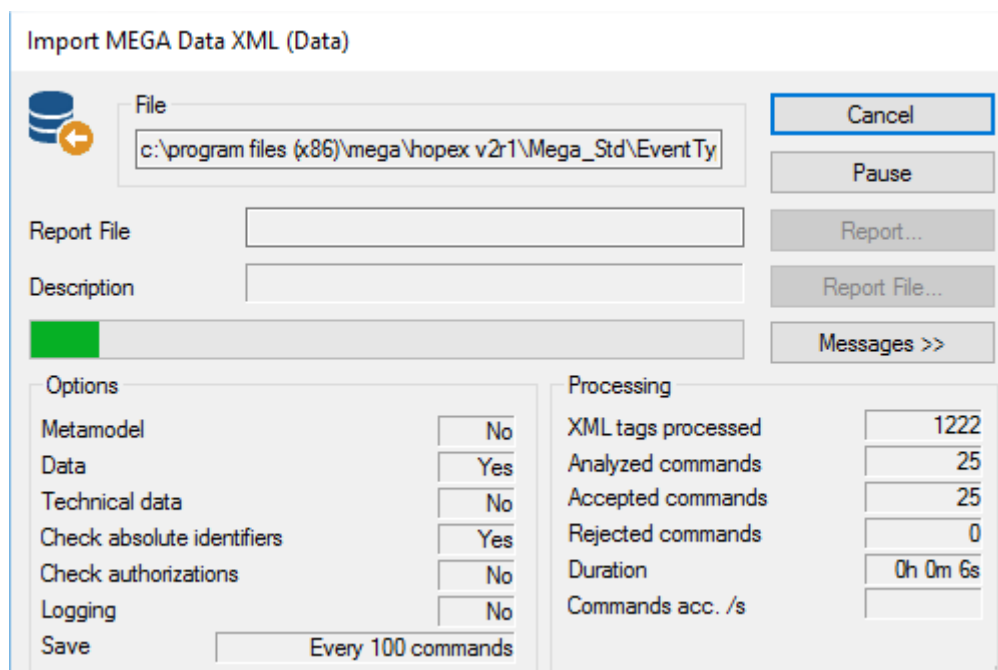
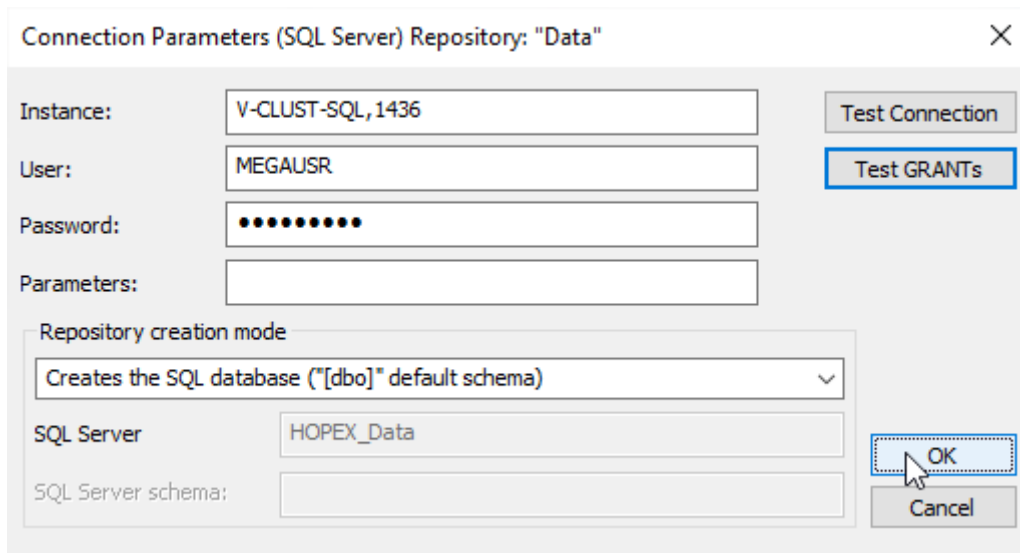
SQL Server

SQL Server schema:

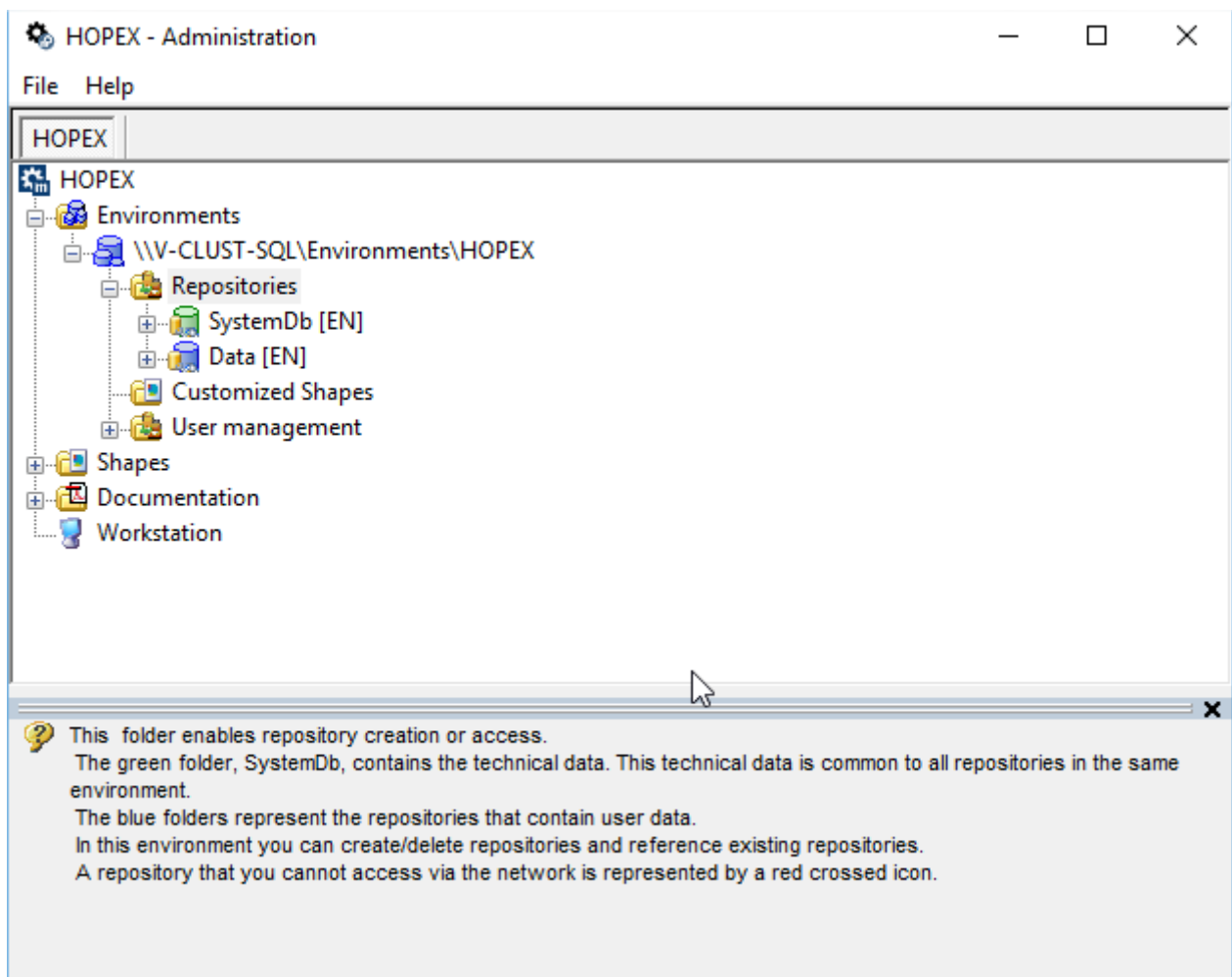
OK Cancel



Click « OK » in the previous window to start the creation, and let it run (a minute or two) :



The repository is created:



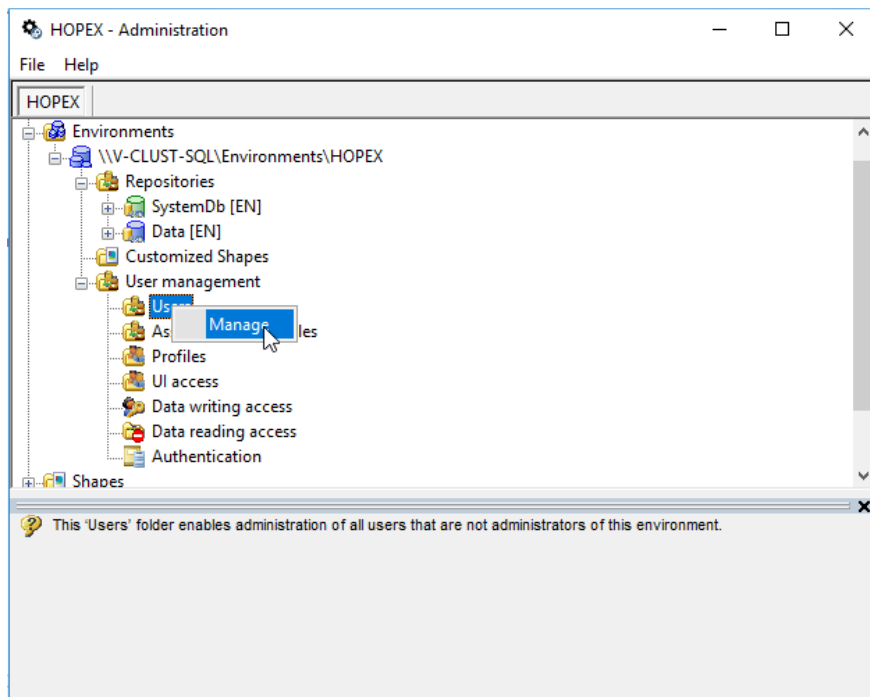
Perform the same steps to create some additional repositories, if necessary. In our example, it is not relevant.

Configure the mega account

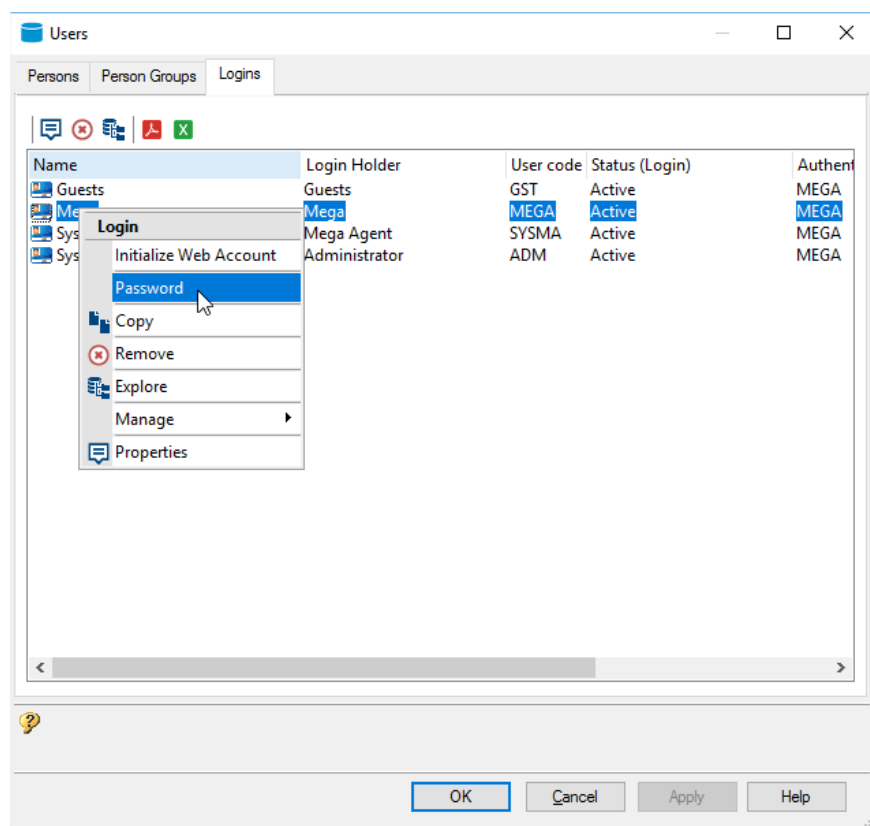
You need to be connected to the environment, using the „administration.exe“ tool.

Provide a password

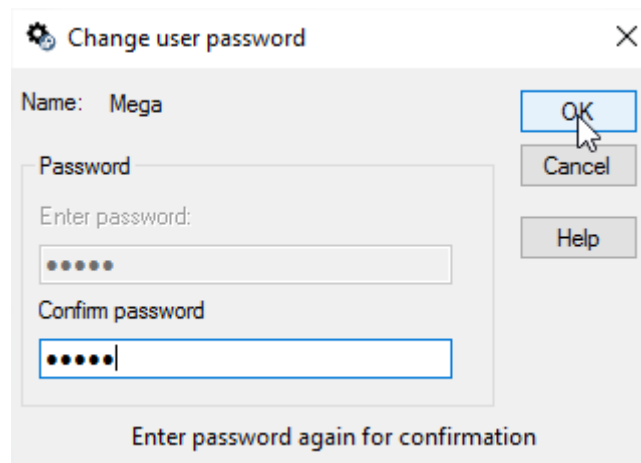
Open „Manage“ for the „Users“ in the environment:



Go to the „Logins“ tab, select the „Mega“ user, right-click and choose „Password“:



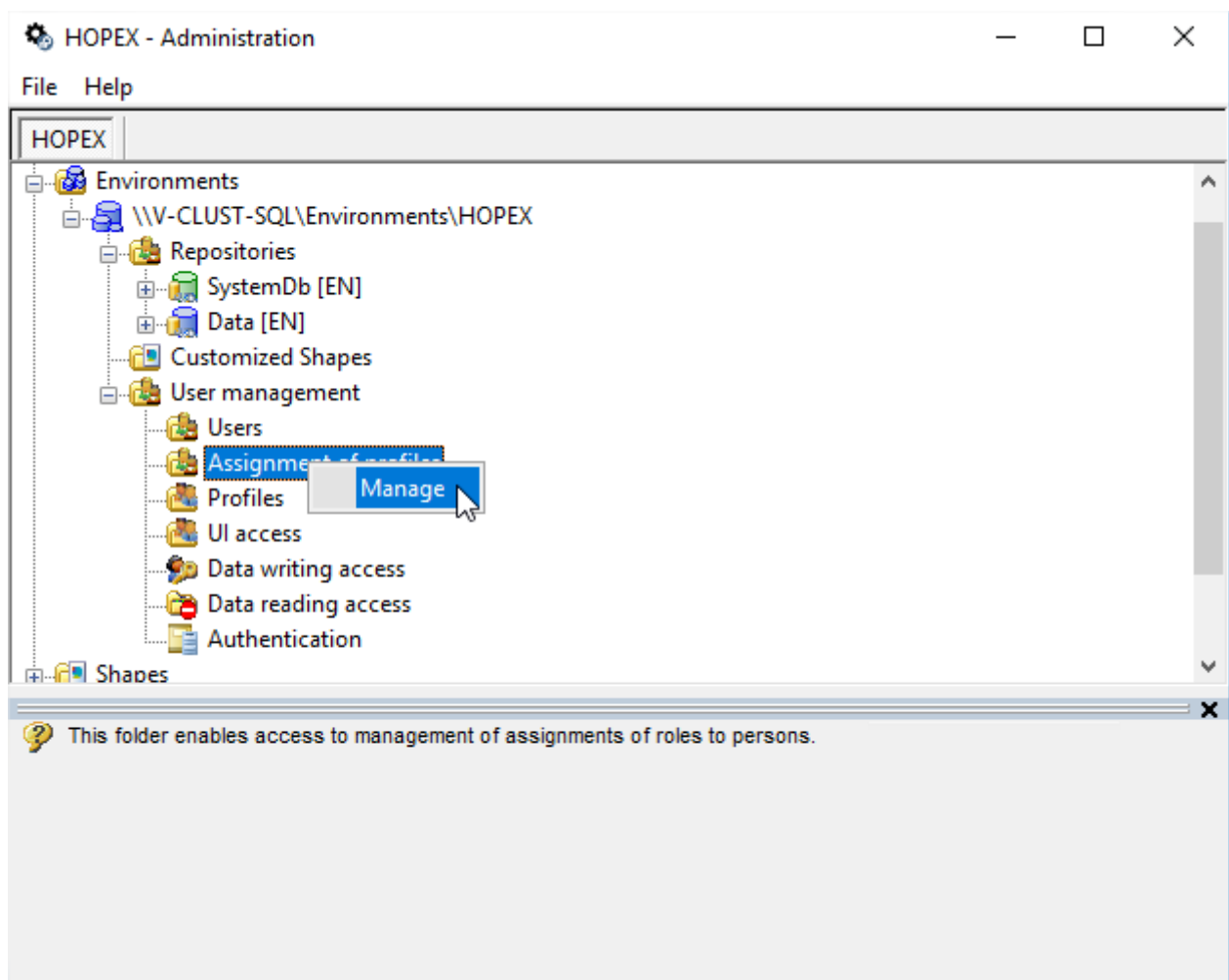
Provide an easy password. In our case we enter „Hopex“. As it will be updated later on when connecting to the web client:



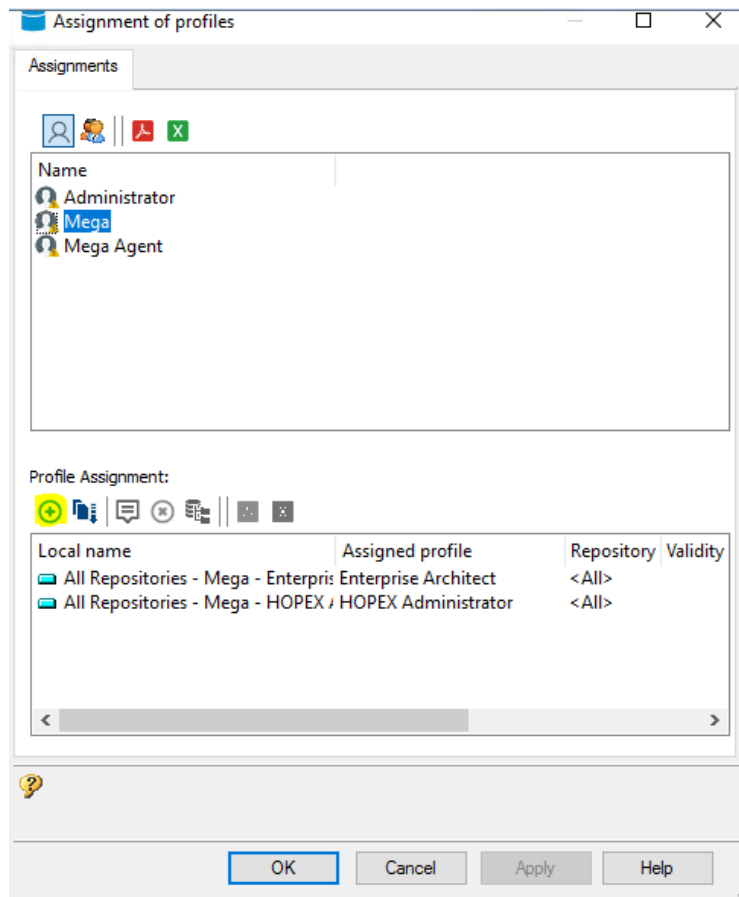
Close the Users interface after applying this password.

Create assignments

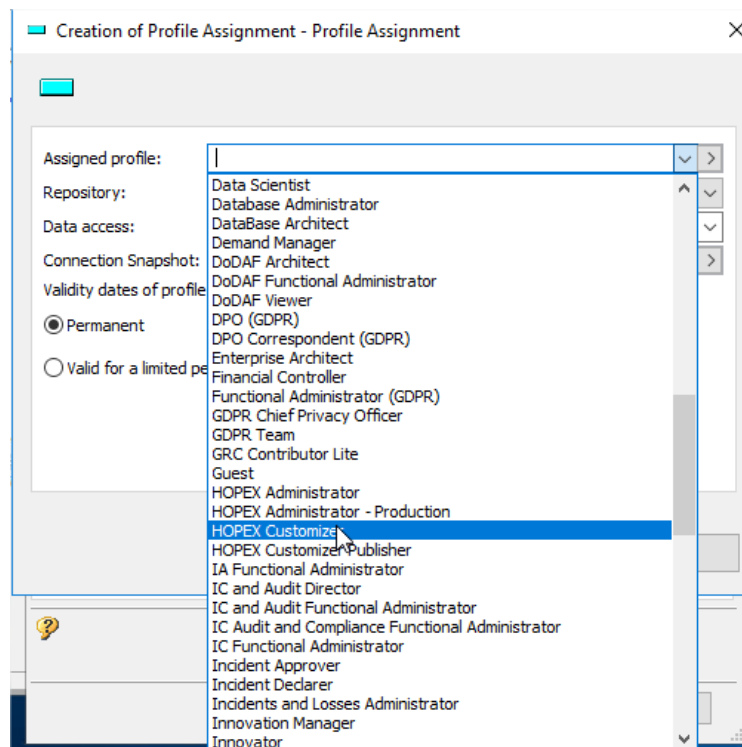
Then, in „Manage“ of „Assignment of Profiles“:



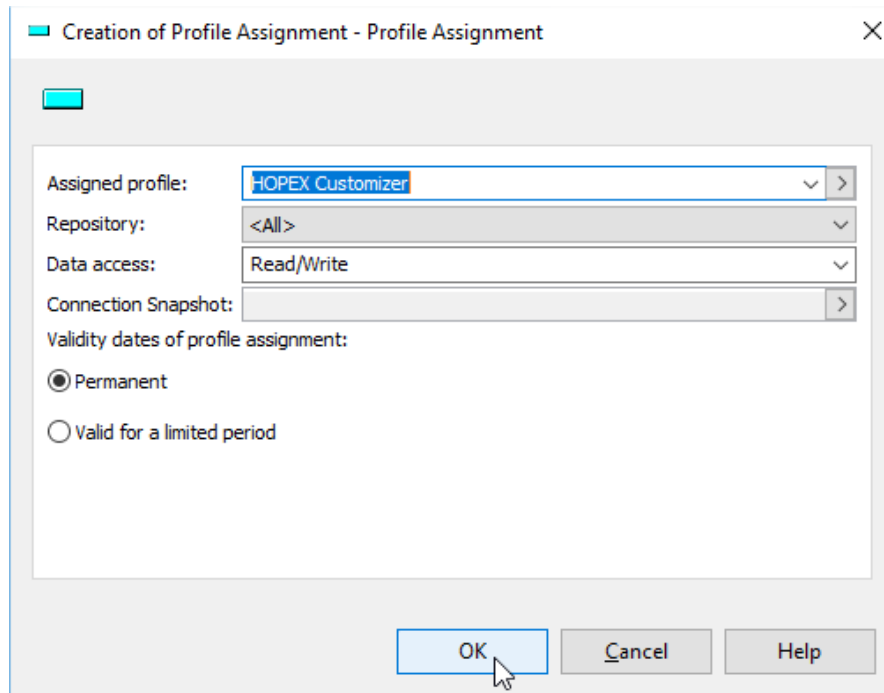
Select the „Mega“ user and click the ‚+‘ button:



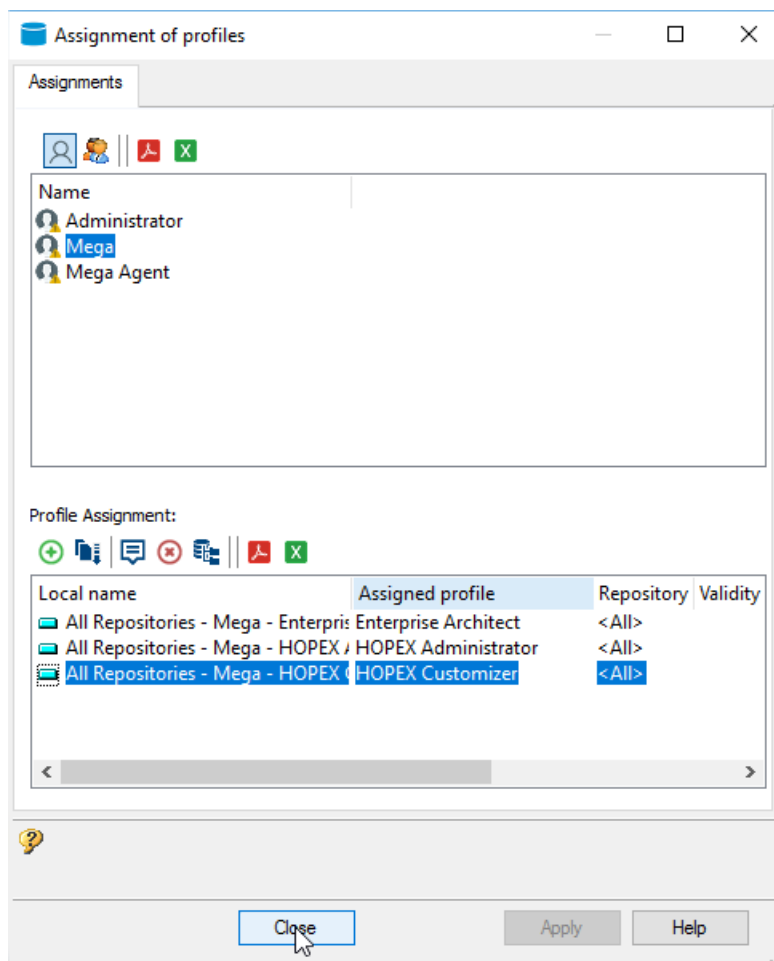
In the dropdown list, select the profiles that you are interested in, based on your license. Here, we will just add the „HOPEX Customizer“ one:



And click „OK“:



Add more assignments if necessary. Otherwise, you can close the window when the list is correct:



Configure the services

To be done on all MWAS and SSP Servers.

Because we have a cluster deployment, and have shared folders containing licenses/environments/megasite, that all need to be accessed by the different components of the application, some Windows Services are installed with the application. Two of them need to be updated, so that the account that runs them, has access to the different shares of the platform:

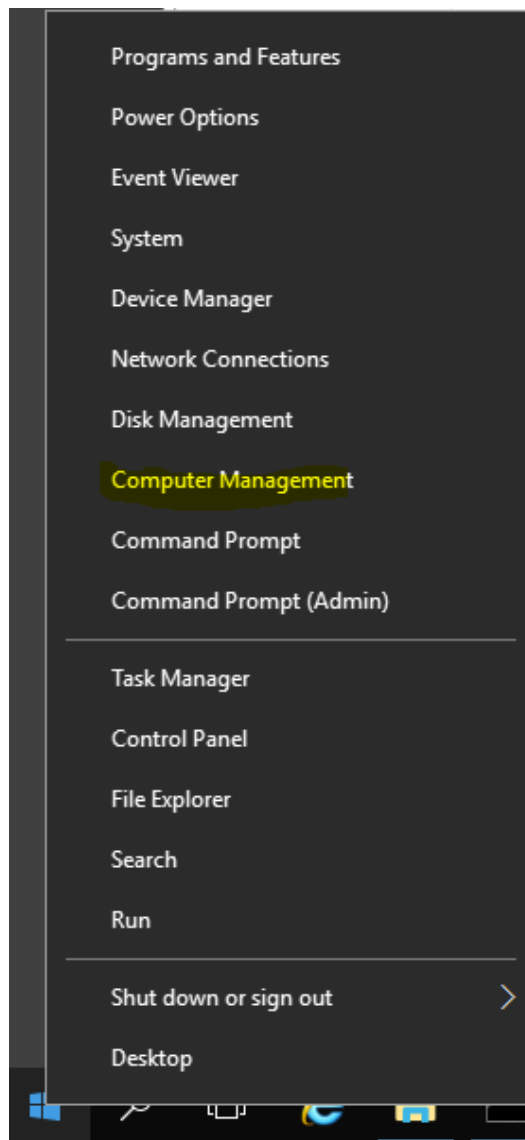
- Hopex Service Watchdog.
- Hopex Site Service Provider.

The goal is to run those services with the impersonate user ("Windows users for Mega" section).

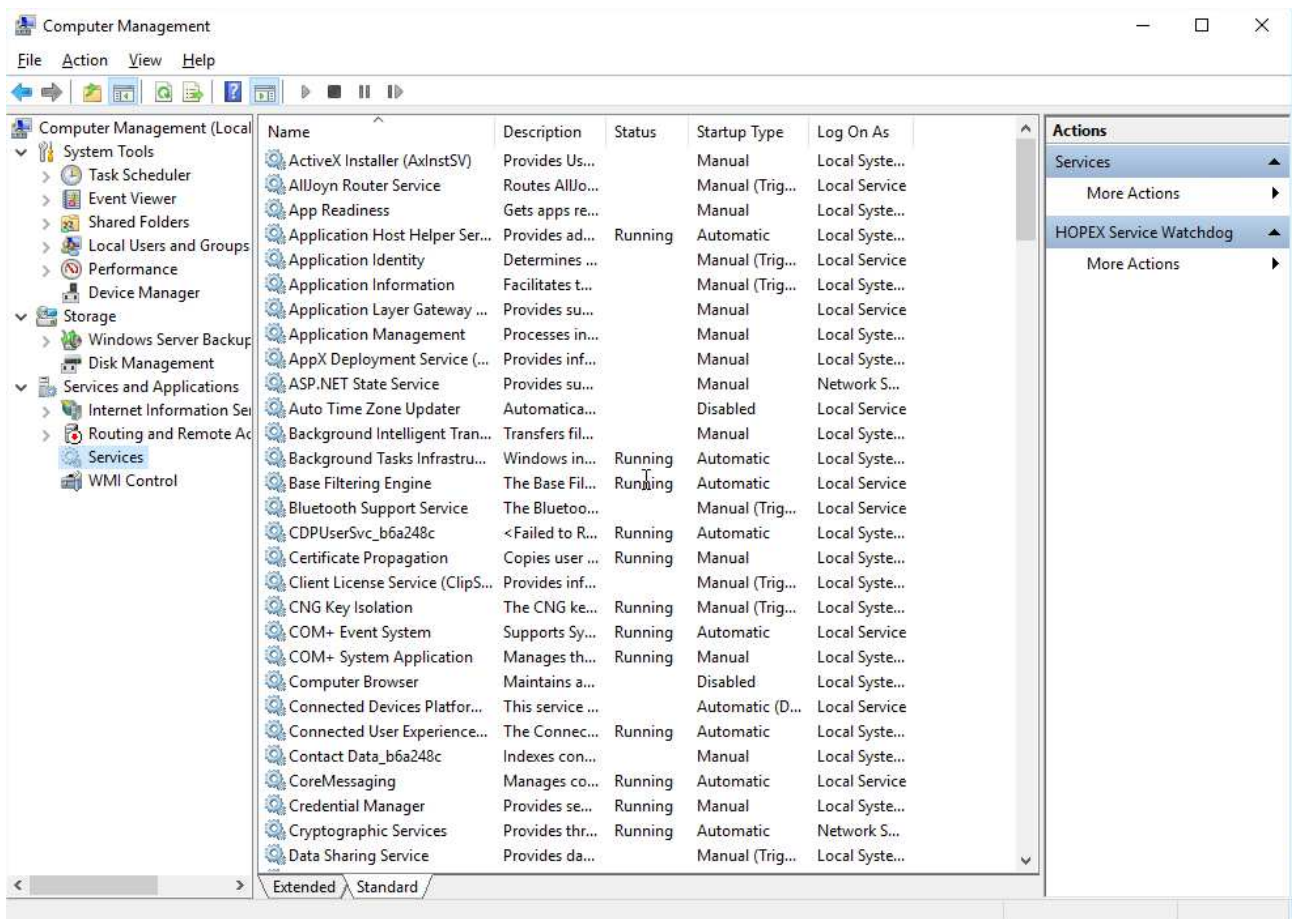
Depending on the type of server, you will find one or two of those services:

- Nothing on the Web Servers.
- Only the Watchdog service on the MWAS Servers.
- Both services on the SSP Servers.

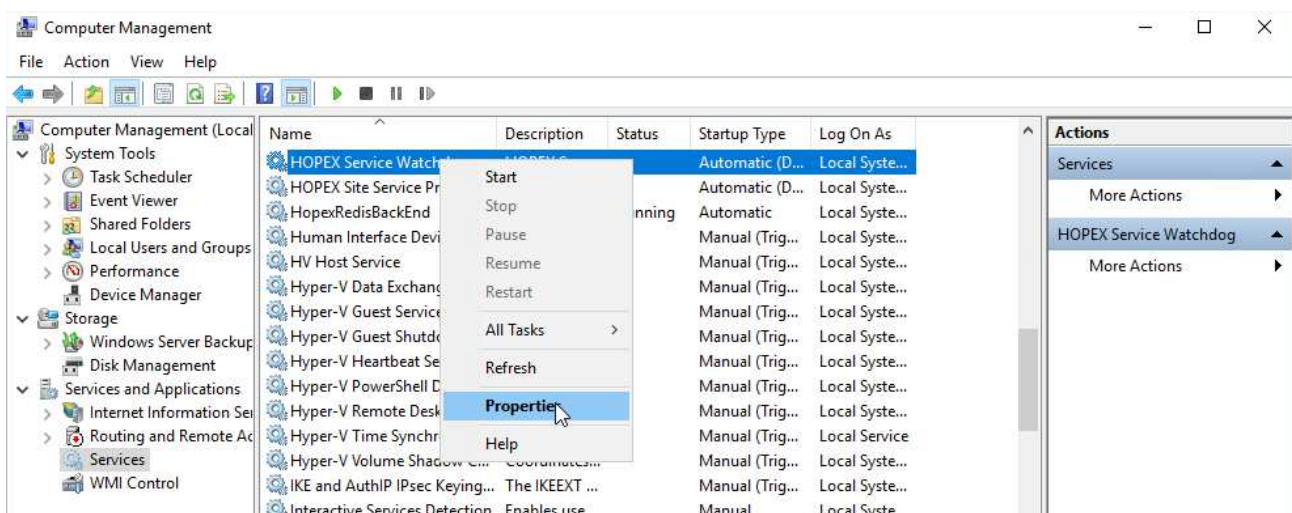
First, open the "Computer Management" interface:



And go to “Services”:

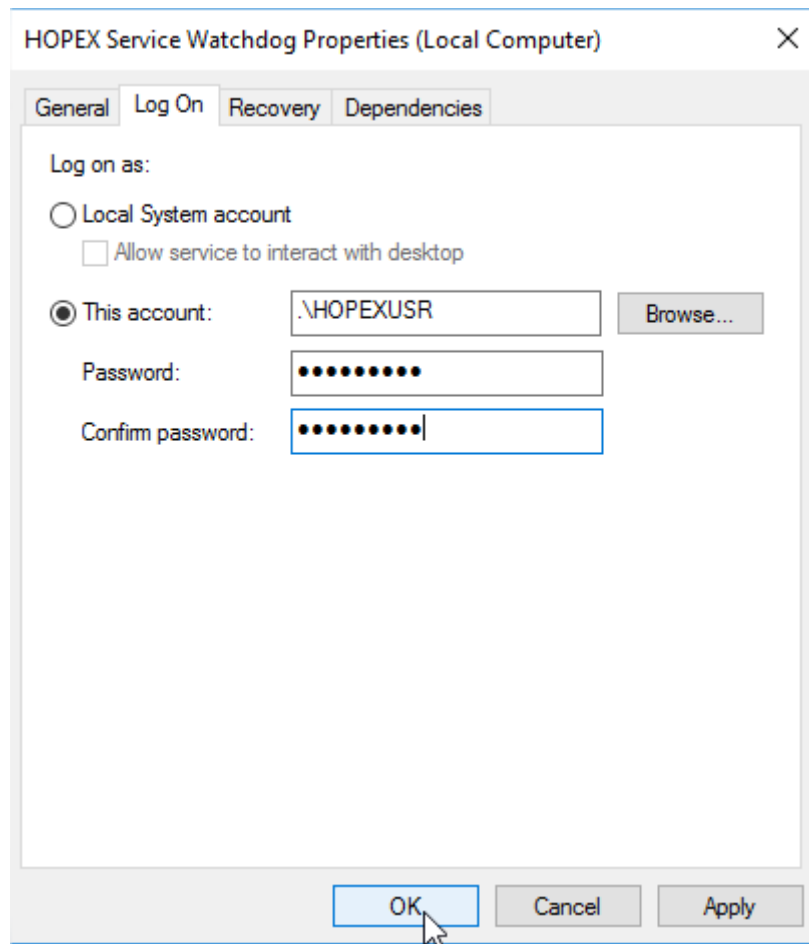


Locate the first service to update (here “Hopex Service Watchdog”) and open its properties:

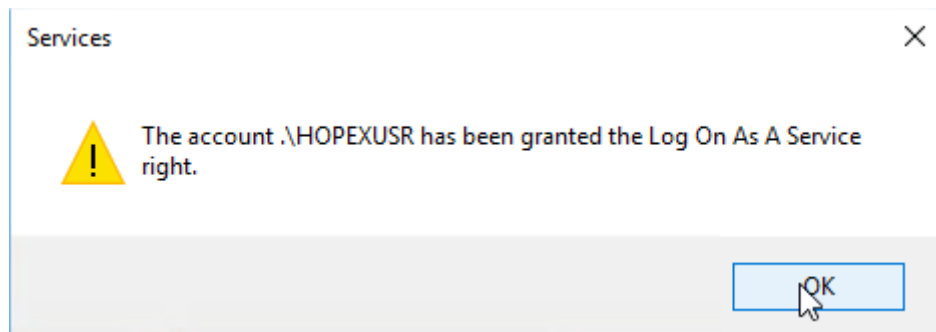


In the “Log On” tab, select “This account”, and put the proper user. We will use “.\\HOPEXUSR” (the “.\\” forcing the use of a local account).

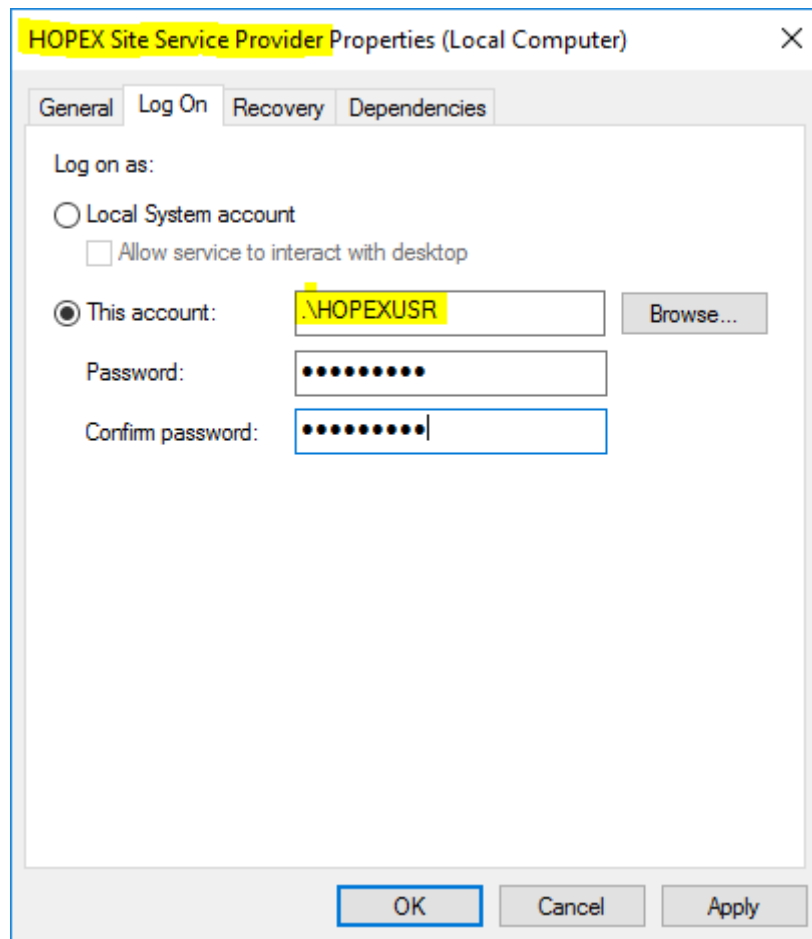
With its password, and click “OK” to validate:



And on “OK” again if the service was started:



Repeat those actions for the “Hopex Site Service Provider” on the SSP servers:

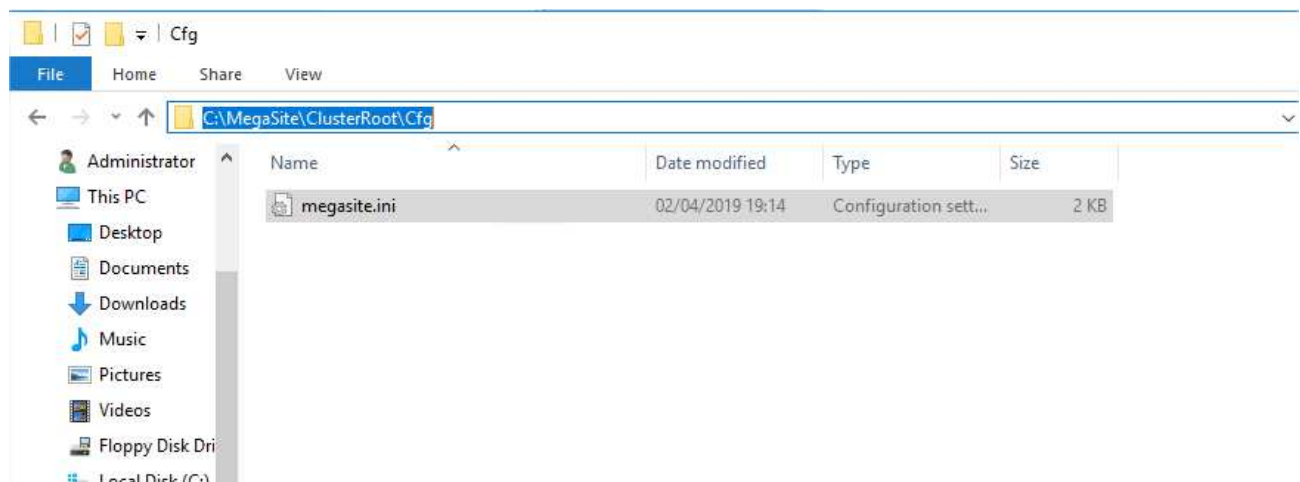


Declare the SSP nodes

When you have multiple SSP Servers, they need to be declared in the MegaSite.ini file, explicitly, so that they can talk to each other to share the information about the activity of the platform, and write in a centralized place the Supervision log.

Since it is not a standard option, you need to edit the centralized “MegaSite.ini” file.

In our case, **go to the RDBMS Server**, and open the file in “C:\MegaSite\ClusterRoot\Cfg”:



Go to the [SSP] section, and add new lines. Each node needs to be listed with its direct URL. Here we have two nodes, so we will create variables “UrlSSPNode1” and “UrlSSPNode2”.

You get the idea if you have more than two nodes, on how to name the other variables.

The URL for each server is the same as the one of the Load Balancer, but it targets directly the servers instead of going through the LB. We use IPs in this deployment, so it looks like this:

```
megasite.ini - Notepad
File Edit Format View Help

[General]
ConfigurationSection=WebServices;Authentication

[WebServices]
Authentication.AuthenticationServerUrl=http://137.74.87.163/uas
Swagger.ProxyUrl=http://137.74.87.163/hopexapi/api/v1.0/uasproxy

[System]
MegaCurrentVersion=30464
Language=US
Company=MEGA
SiteName=TEMPLATE

[SSP]
Url=http://137.74.87.169/MegaSSP
SecurityKey=!@+-70395613ACF69850FDF76D26AA7E6285
UrlSSPNode1=http://137.74.87.167/MegaSSP
UrlSSPNode2=http://137.74.87.168/MegaSSP

[Lan]
clusterrootpath=

[UserConfiguration]
DefaultDataLanguage=00(6w1Hmk400
DefaultGUILanguage=00(6w1Hmk400
```

Save the file and restart the application so it is taken into account.

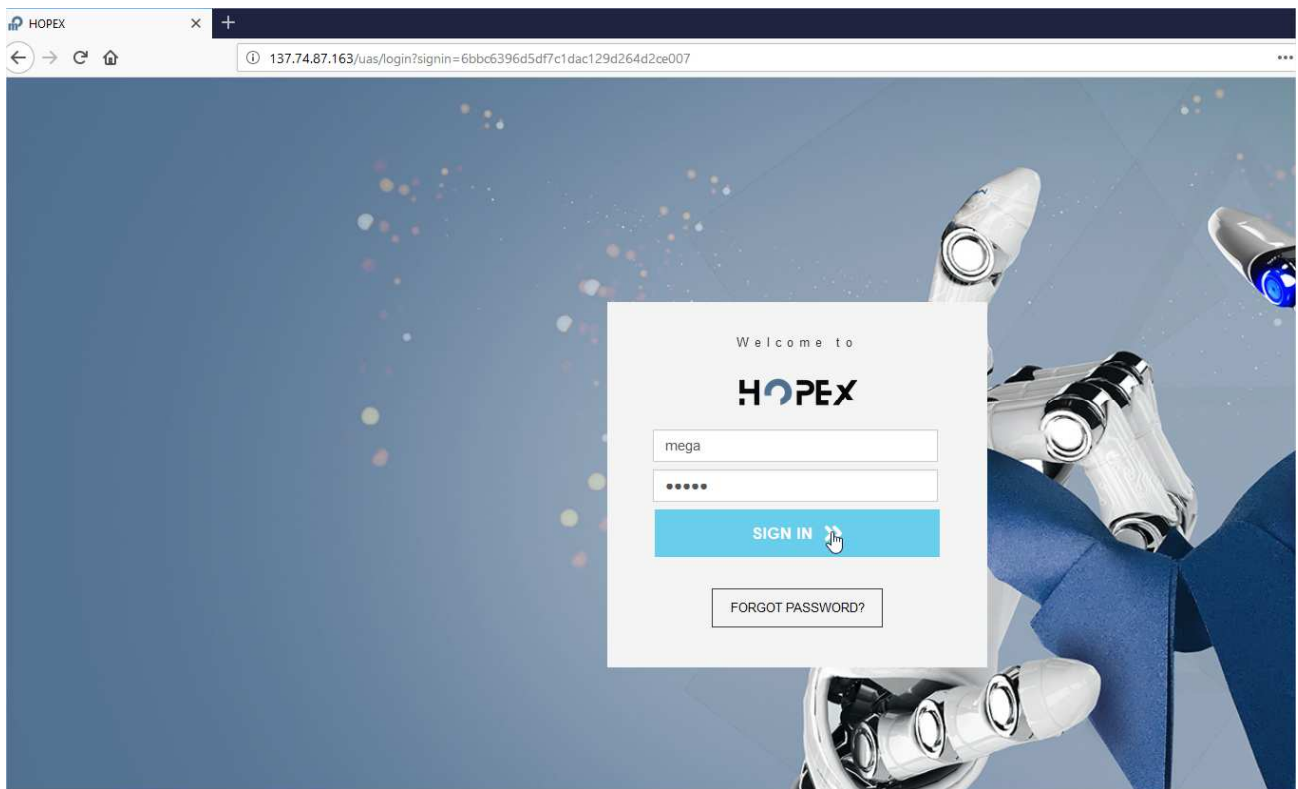
Test the web client

Prerequisite: the application must be started on all servers. See Annex for the procedure.

The URL is:

<http://137.74.87.163/hopex>

We are redirected to the UAS authentication page. When we enter the credentials of the “mega” user (at that time, password is “Hopex”):



We then choose an assignment, like „Application Architect“, check the box to agree to the privacy policy (only for the first connection), and click „Login“:

Enterprise Architect

☒ I have read and accept the privacy policy

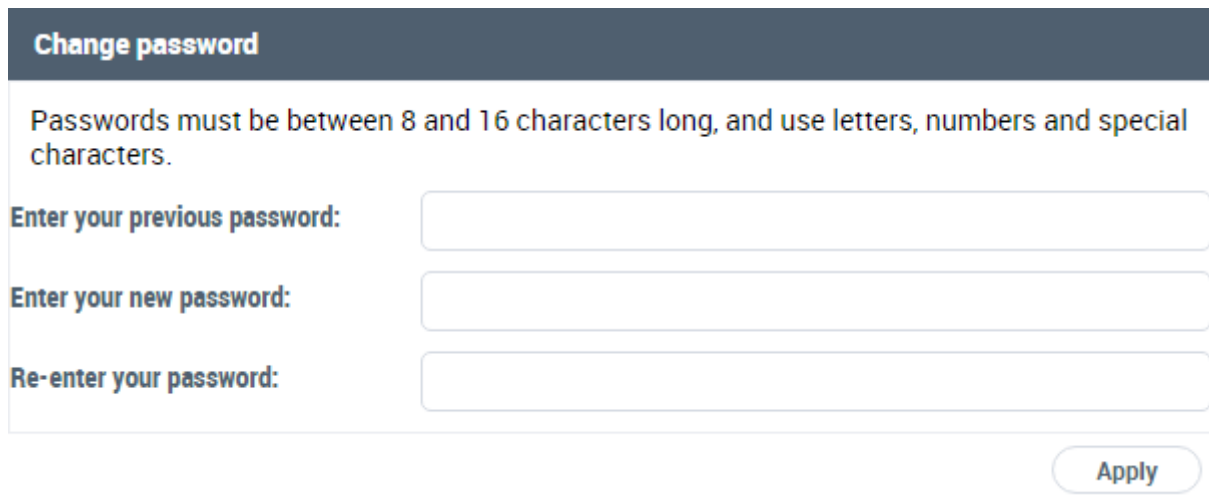
Create a private workspace

LOGIN >>

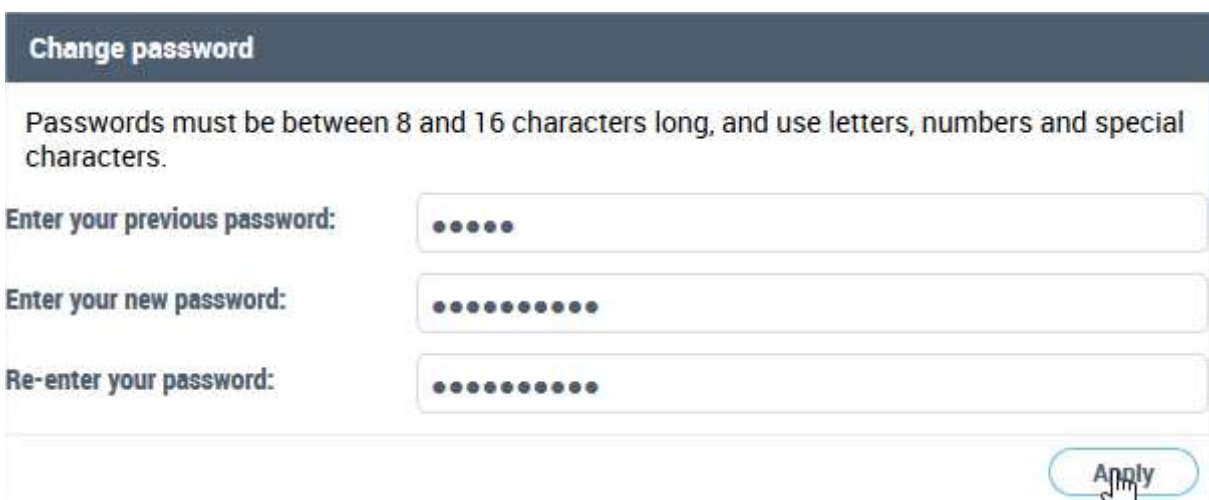
« BACK

PRIVACY POLICY

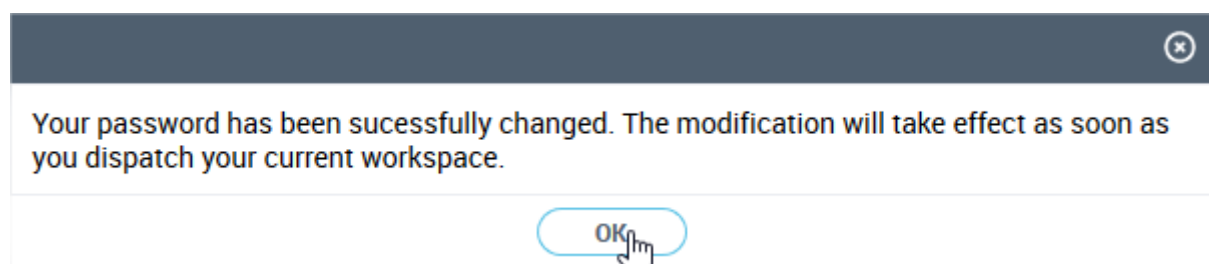
As the password was set up by the „system“ account in the administration tool, the application forces the user to set up its own password with the complexity show below:



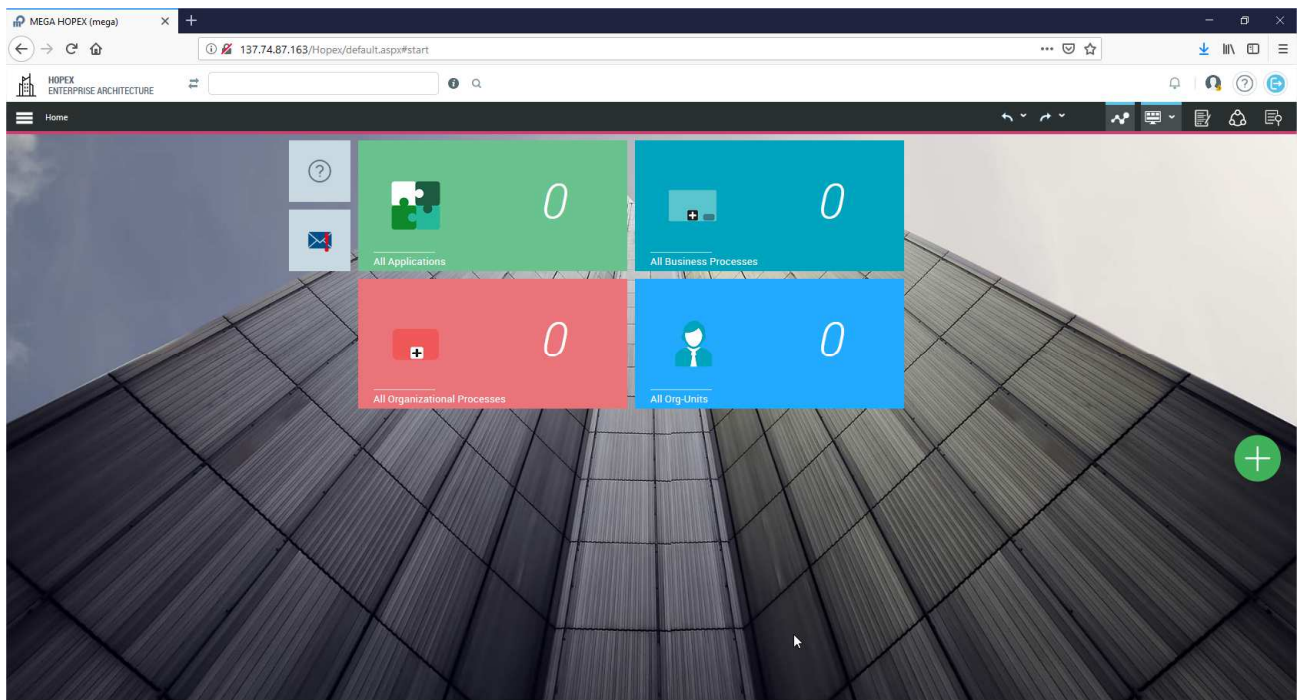
We provide a strong password and click „Apply“:



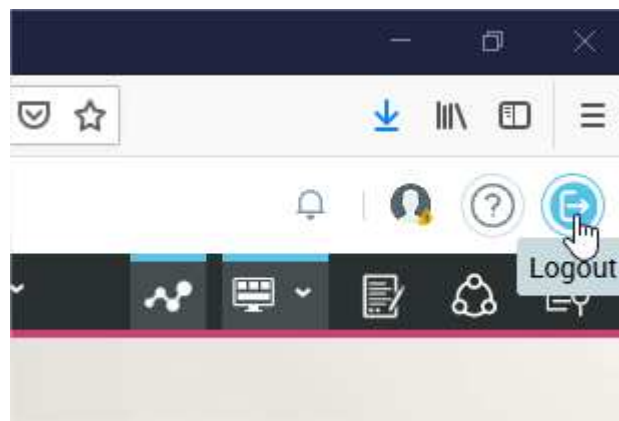
The password is updated:



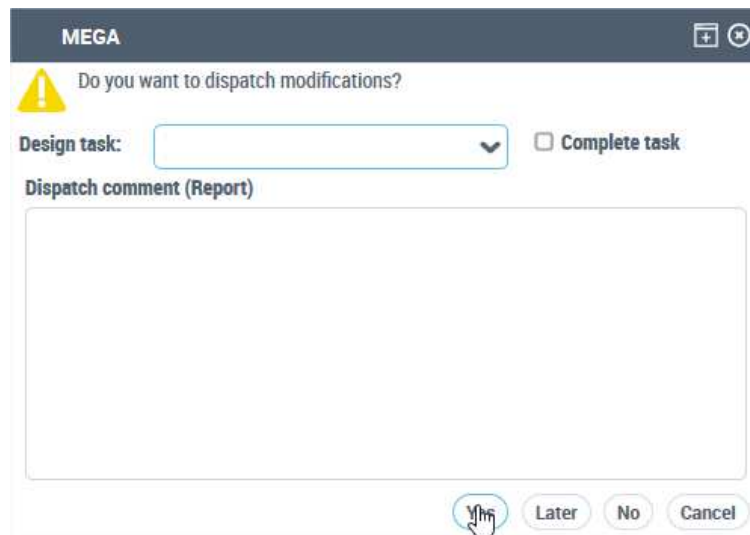
The desktop is showing properly:



Click the „Logout“ button:



Click „Yes“, otherwise the password change won't be done and you will have to do it again:



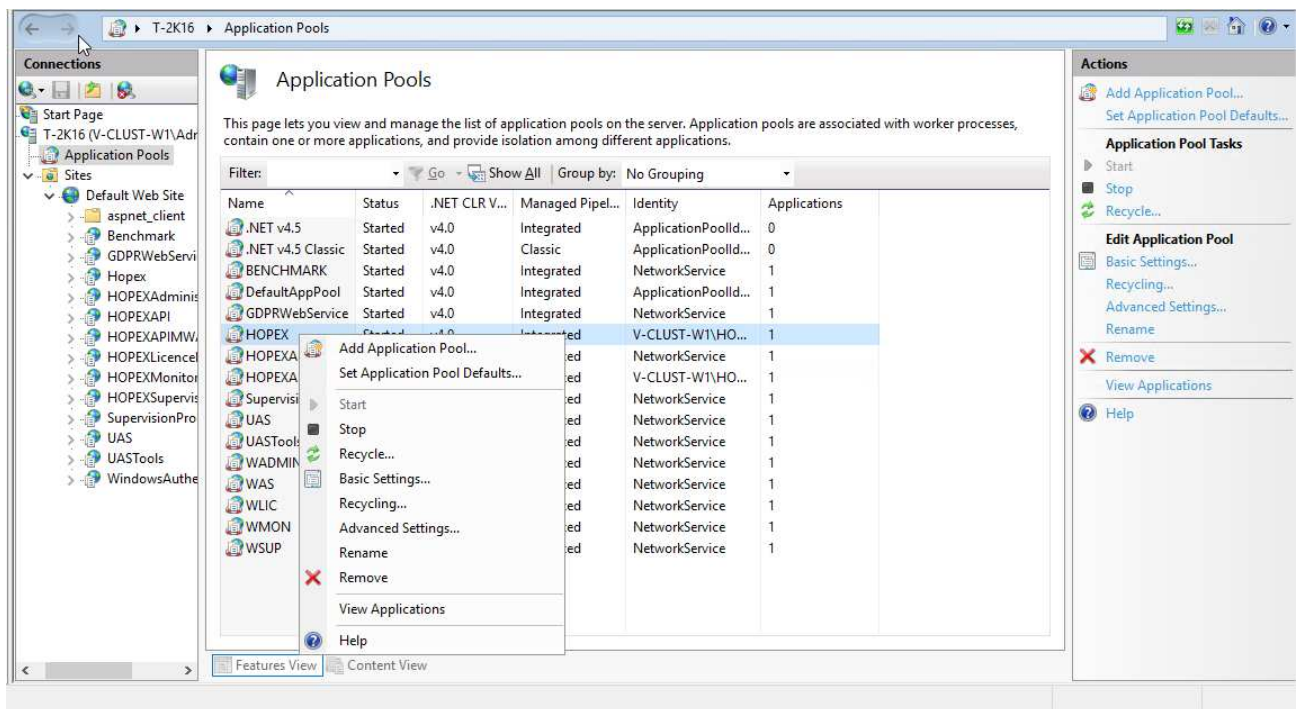
IIS Tuning

To be done on the Web Servers only.

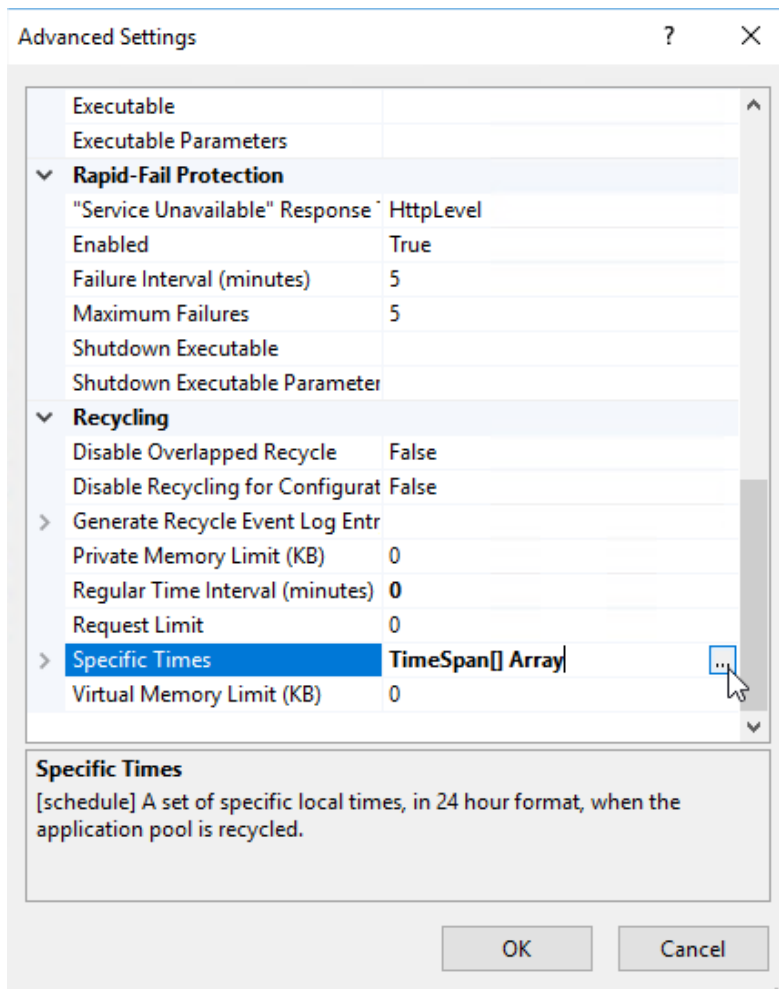
Application pools recycling

We need to configure the “HOPEX” application pool on each web server, in order to avoid an automatic recycling of the processes during users’ activity.

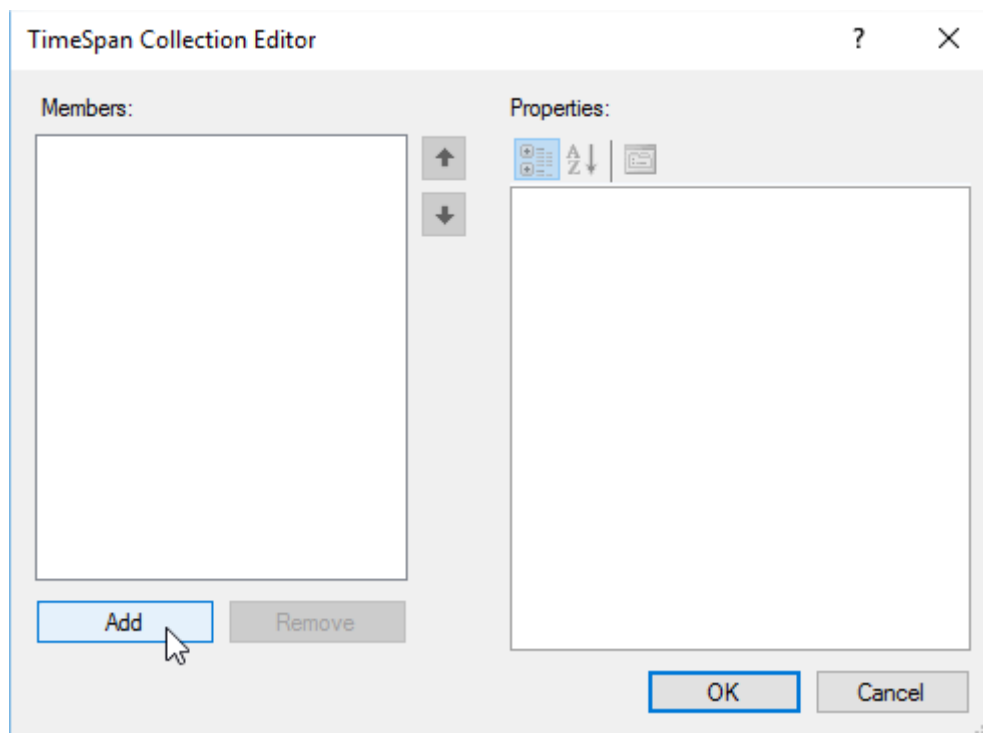
Go to the IIS Management console, then to the "Application Pools" section, select the “HOPEX” one, and open the “Advanced settings”:



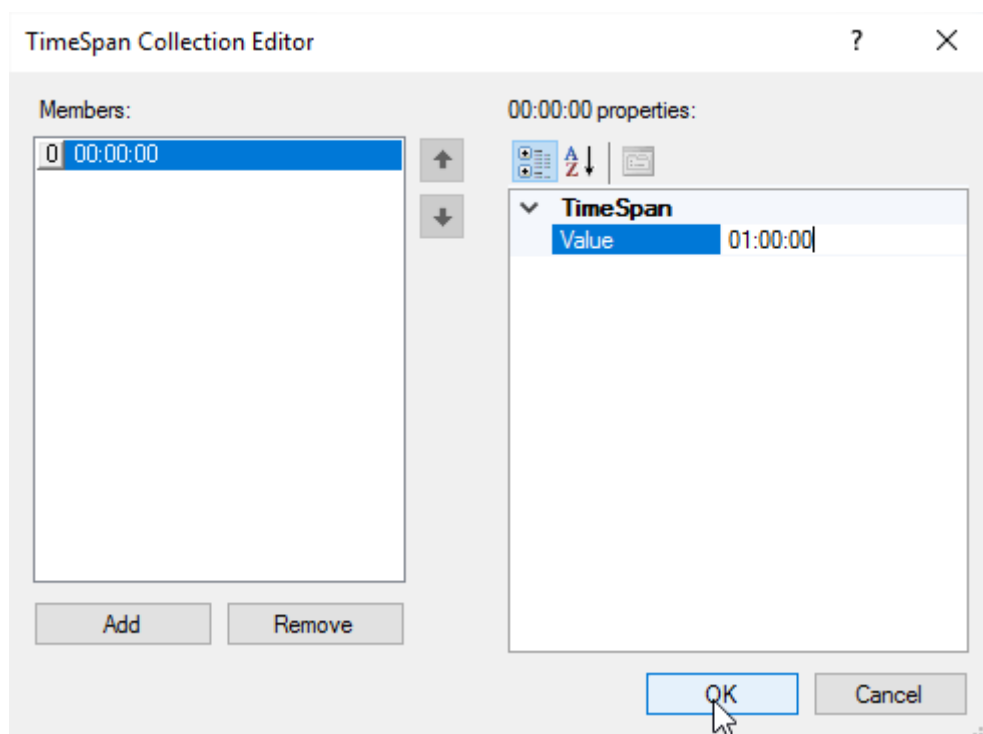
Select the “Specific Times” line, and click the “...” button:



Click "Add" :

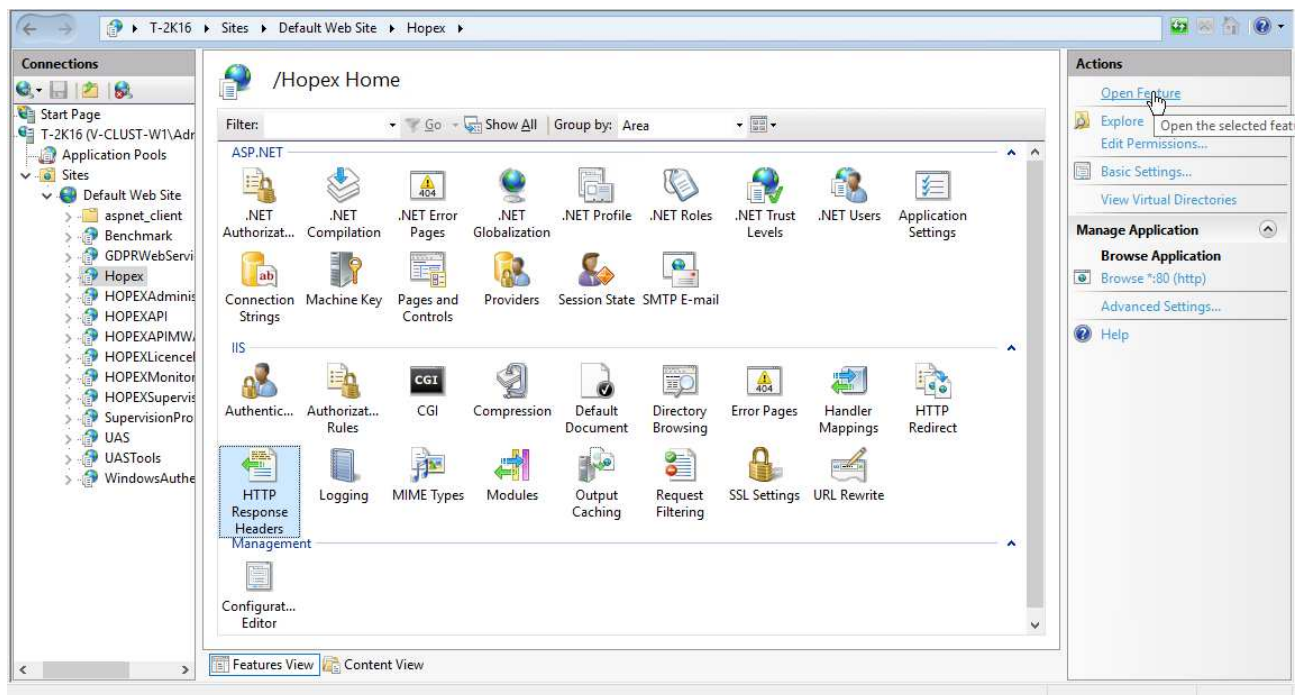


We decided to recycle every night at 1am, and validate:

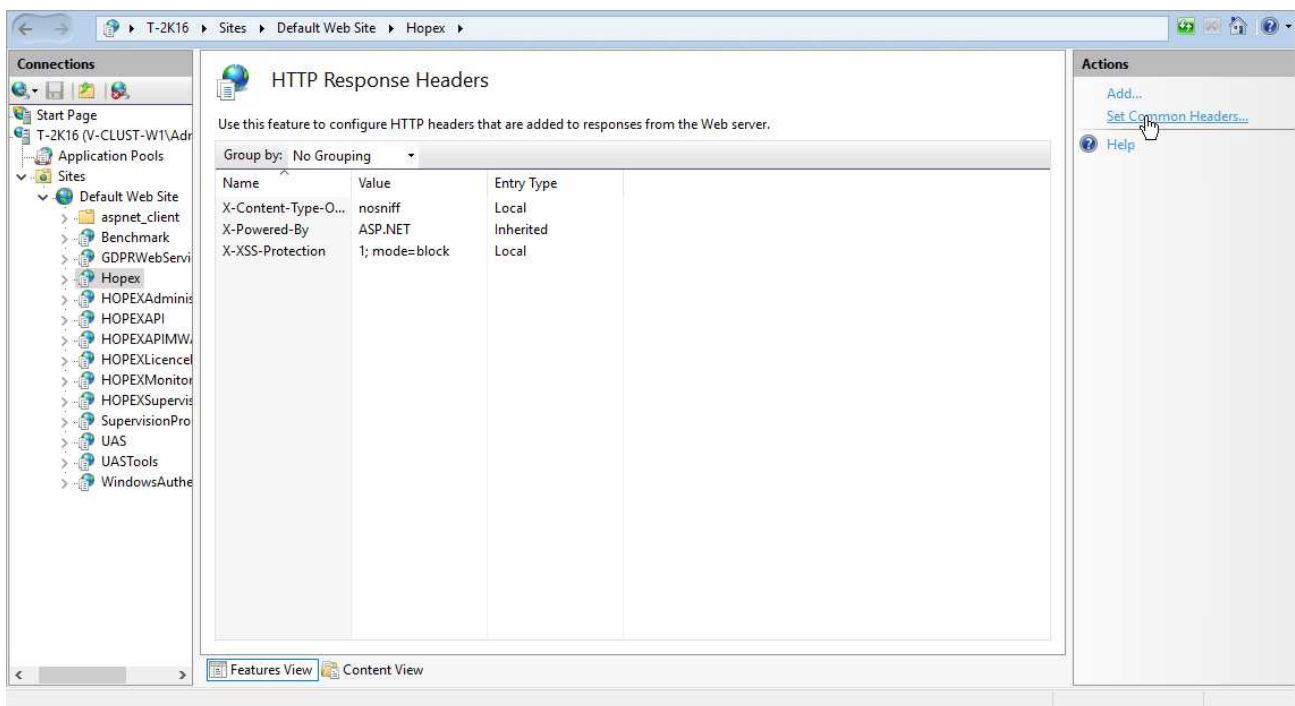


Manage expiration of HTTP response headers

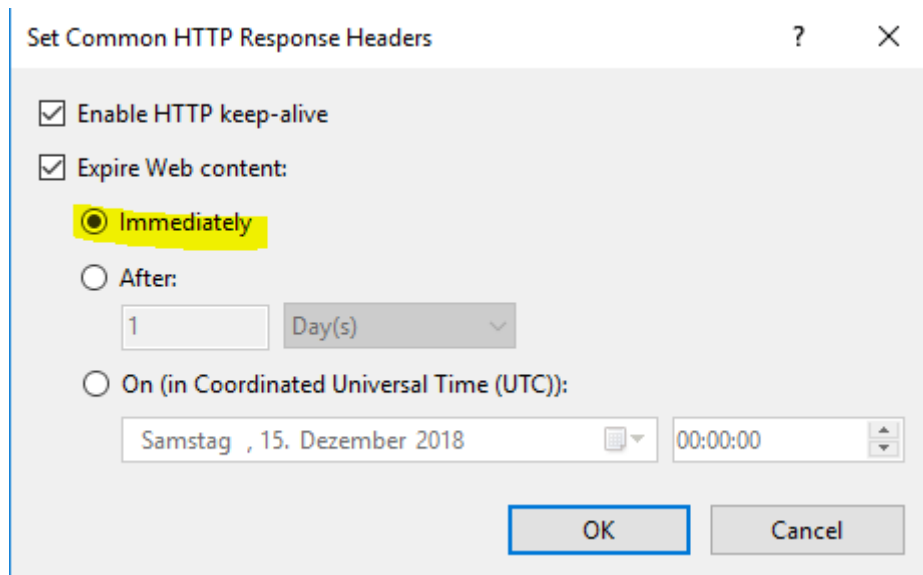
On the “HOPEX” web application within your website, we open the feature called “HTTP Response Headers”:



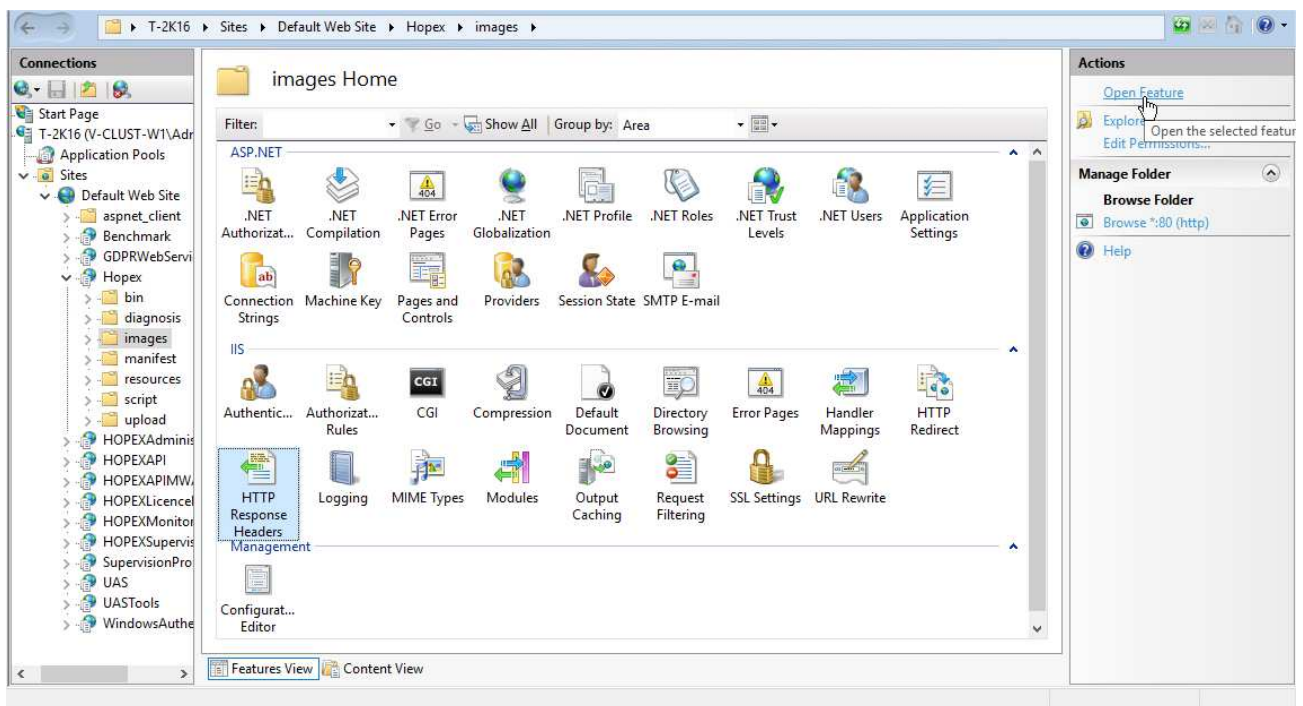
We click "Set Common Headers...":



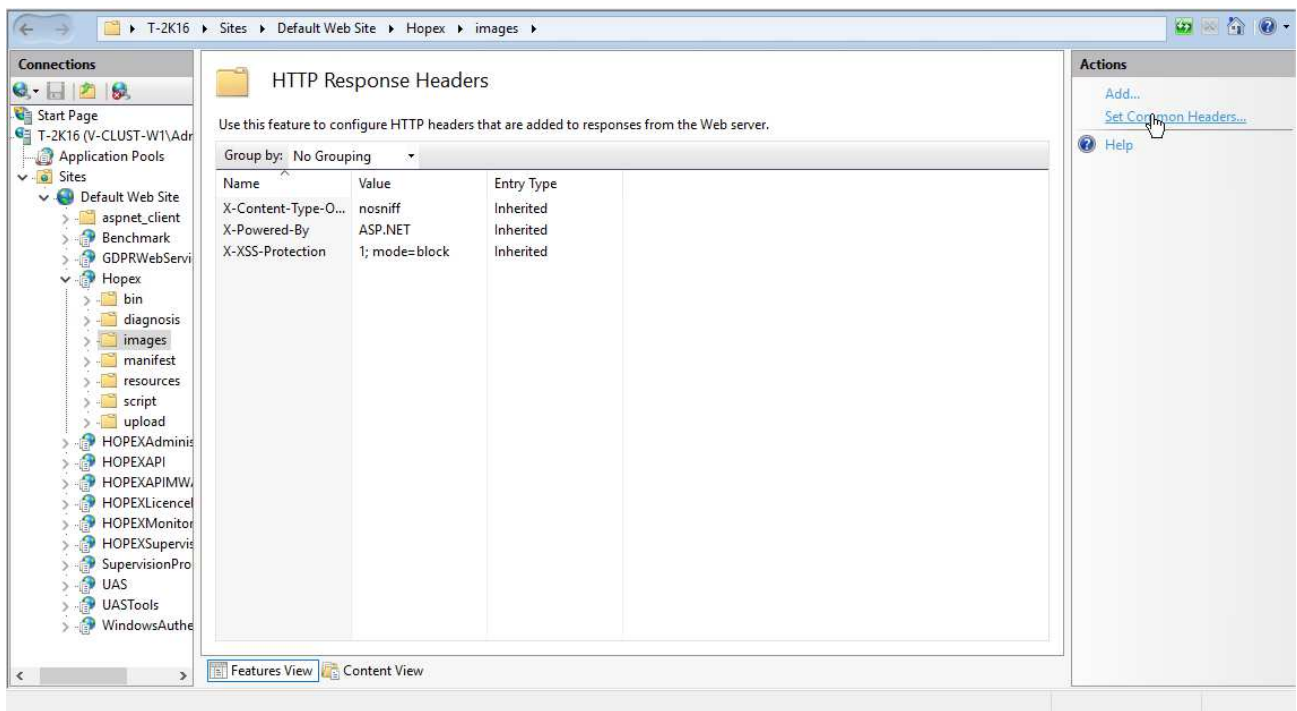
We check the box “Expire web content” and we keep the “Immediately” setting :



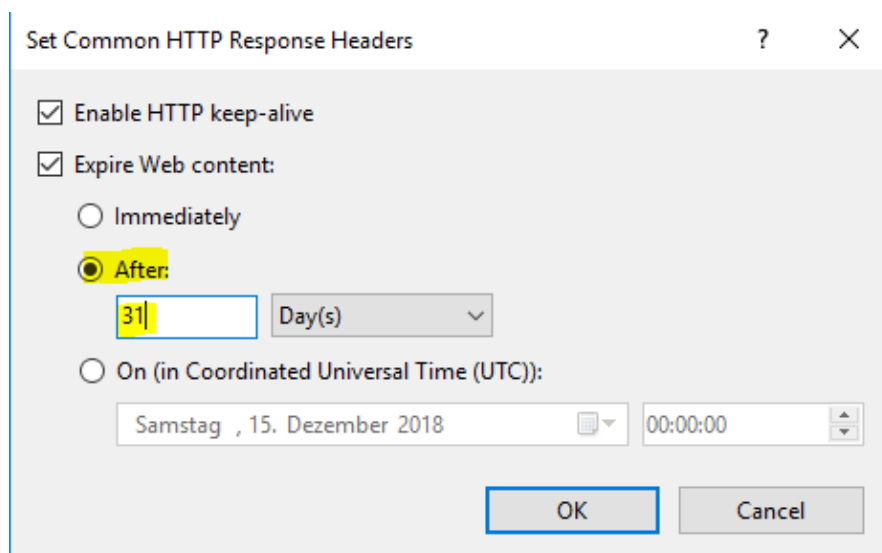
Then, we expand the web application, we select the “images” folder, and we open the same feature:



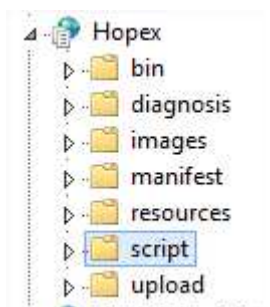
And click the same link:



This time we change the setting so that it expires after 31 days:



We do the same on the “**script**” folder:



Diagnostic Tools

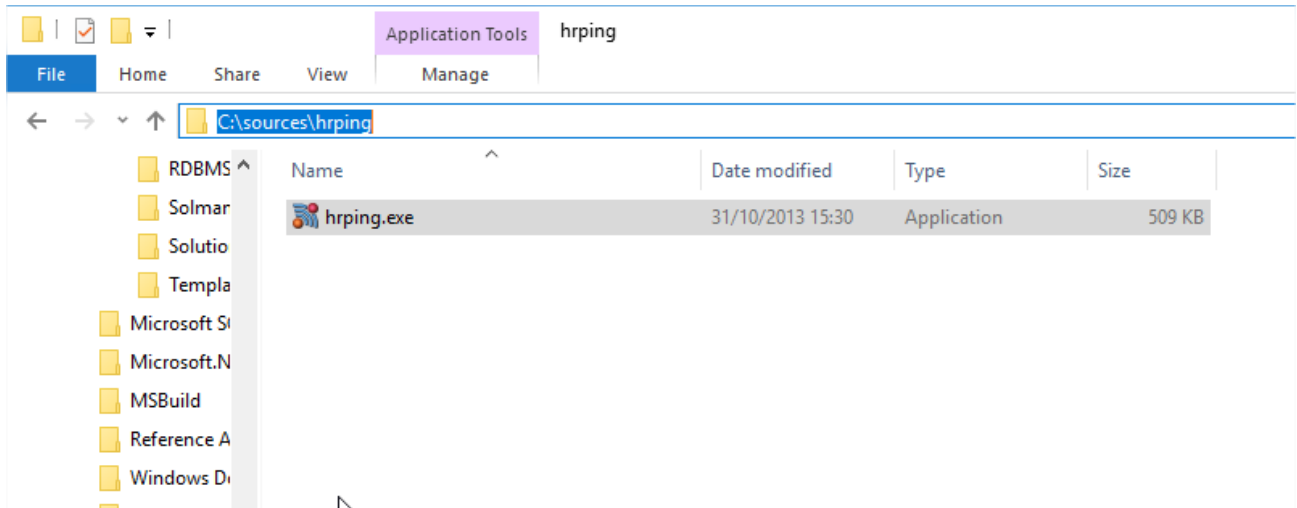
We ran a series of tests in order to estimate the performance that could be attained on the Production platform.

The tools must run on all MWAS and SSP Servers.

Latency test with hrping

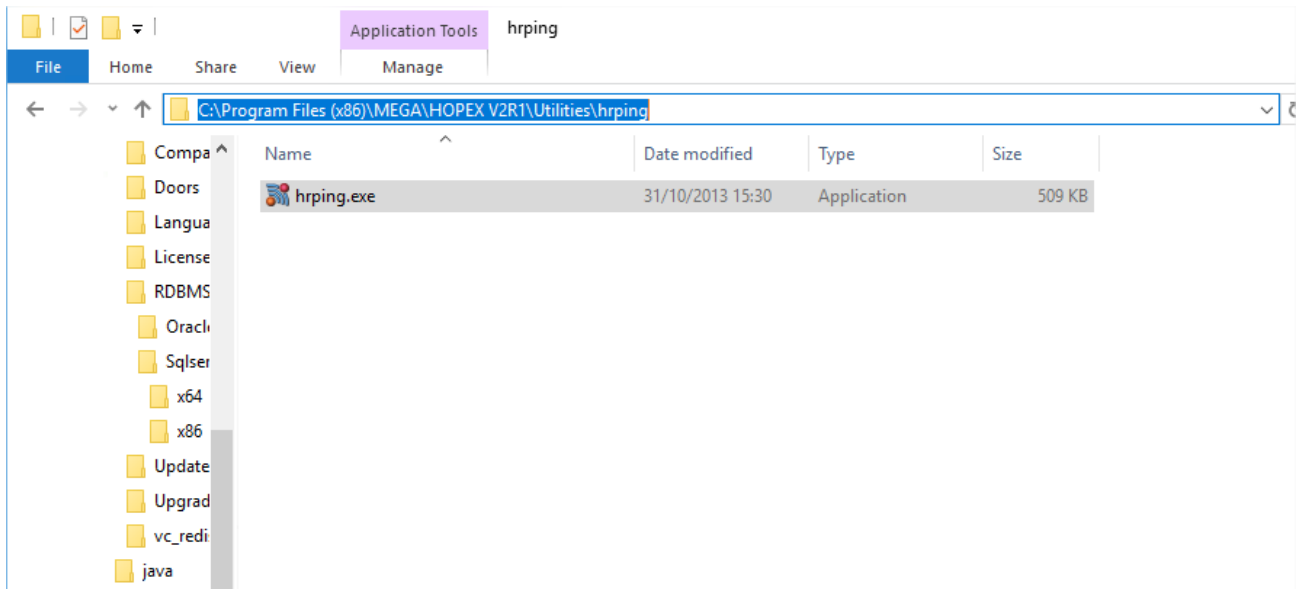
The latency between the application server and the database server is crucial. It is important for it to be small. Ideally, lower than 1ms.

On this installation, it was put in the “C:\sources\hrping” folder:



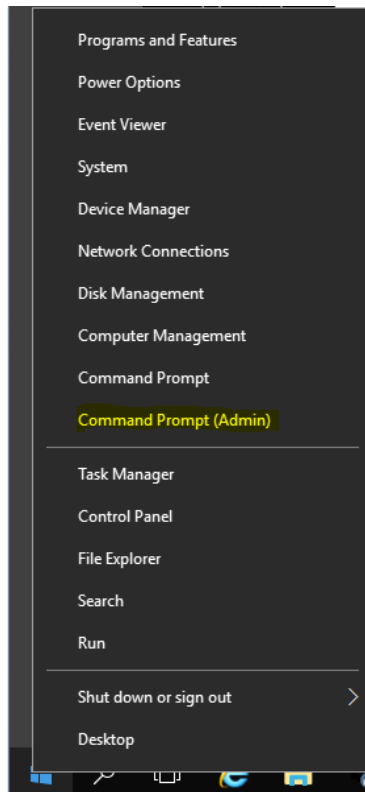
Otherwise you can get it from this link: <https://www.cfos.de/en/ping/ping.htm>. Only the “hrping.exe” is needed.

Then we copied it on all **MWAS and SSP** servers in “C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping”:



We then must execute the following command between the two servers.

Please note that the first execution on a server of the hrping tool requires to accept the disclaimer. We just must parse through them and reply “Y” for yes at the end to accept.
Also, when the UACs are activated on a server, the command invite window needs to be run with a “Run as Administrator” option:



Explaining of the command line for hrping:

hrping.exe -W -l 5000 -n 50 -y V-CLUST-SQL

The parameters:

- W : warm-up, the 1st ping is not taken into account
- l number : size of the packet, in bytes
- n number : number of sent pings
- y : regroup the results instead of writing line after line

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd "..\..\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping"
C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping>hrping.exe -W -l 5000 -n 50 -y V-CLUST-SQL
```

Press "Enter", then "Esc", to accept to read the disclaimer, and go at the end. Then press J to reply Ja to the question at the end.

First MWAS Server (V-CLUST-MW1)

```
Administrator: C:\Windows\system32\cmd.exe
aufgeführten Lizenzbedingungen vom Nutzer anerkannt und
eingehalten werden. Es kommt damit zwischen ihm, dem
Lizenznehmer, und der Lizenzgeberin der vorliegende Lizenzvertrag
zustande:

2. Installation:

Die Lizenzgeberin weist Sie ausdrücklich darauf hin, daß vor

Do you agree to the public license and warranty? (Y/N) :
Source address is 137.74.87.164; using ICMP echo-request, ID=4405
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:
  Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500143 sec
  RTTs in ms: min/avg/max/dev: 0.581 / 0.701 / 1.618 / 0.141
  Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261  4
Last 10 seconds:
  Packets: sent=20, rcvd=20, error=0, lost=0 (0.0% loss)
  RTTs in ms: min/avg/max/dev: 0.581 / 0.740 / 1.618 / 0.212
  Bandwidth in kbytes/sec: sent=10.583, rcvd=10.583  4

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500143 sec
RTTs in ms: min/avg/max/dev: 0.581 / 0.701 / 1.618 / 0.141
Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261

C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping>
```

Source address is 137.74.87.164; using ICMP echo-request, ID=4405
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:
Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500143 sec

RTTs in ms: min/avg/max/dev: 0.581 / 0.701 / 1.618 / 0.141
 Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261 4
 Last 10 seconds:
 Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)
 Packets: sent=20, rcvd=20, error=0, lost=0 (0.0% loss) in 9.501326 sec
 RTTs in ms: min/avg/max/dev: 0.581 / 0.740 / 1.618 / 0.212
 Bandwidth in kbytes/sec: sent=10.583, rcvd=10.583 4

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500143 sec
 RTTs in ms: min/**avg**/max/dev: 0.581 / **0.701** / 1.618 / 0.141
 Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261

Second MWAS Server (V-CLUST-MW2)

```
Administrator: C:\Windows\system32\cmd.exe
aufgeführten Lizenzbedingungen vom Nutzer anerkannt und
eingehalten werden. Es kommt damit zwischen ihm, dem
Lizenznehmer, und der Lizenzgeberin der vorliegende Lizenzvertrag
zustande:

2. Installation:

Die Lizenzgeberin weist Sie ausdrücklich darauf hin, daß vor

Do you agree to the public license and warranty? (Y/N) :
Source address is 137.74.87.165; using ICMP echo-request, ID=e407
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:
Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500359 sec
RTTs in ms: min/avg/max/dev: 0.399 / 0.500 / 0.655 / 0.050
Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261
Last 10 seconds:
Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)
Packets: sent=20, rcvd=20, error=0, lost=0 (0.0% loss) in 9.501167 sec
RTTs in ms: min/avg/max/dev: 0.441 / 0.522 / 0.655 / 0.054
Bandwidth in kbytes/sec: sent=10.583, rcvd=10.583

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500359 sec
RTTs in ms: min/avg/max/dev: 0.399 / 0.500 / 0.655 / 0.050
Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261

C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping>
```

Source address is 137.74.87.165; using ICMP echo-request, ID=e407
 Pinging V-CLUST-SQL [137.74.87.170]
 with 5000 bytes data (5028 bytes IP):

Total:
 Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500359 sec
 RTTs in ms: min/avg/max/dev: 0.399 / 0.500 / 0.655 / 0.050
 Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261
 Last 10 seconds:
 Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)
 Packets: sent=20, rcvd=20, error=0, lost=0 (0.0% loss) in 9.501167 sec
 RTTs in ms: min/avg/max/dev: 0.441 / 0.522 / 0.655 / 0.054
 Bandwidth in kbytes/sec: sent=10.583, rcvd=10.583

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.500359 sec
 RTTs in ms: min/**avg**/max/dev: 0.399 / **0.500** / 0.655 / 0.050

Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261

First SSP Server (V-CLUST-S1)

```
Administrator: Command Prompt
aufgeführten Lizenzbedingungen vom Nutzer anerkannt und
eingehalten werden. Es kommt damit zwischen ihm, dem
Lizenznehmer, und der Lizenzgeberin der vorliegende Lizenzvertrag
zustande:

2. Installation:

Die Lizenzgeberin weist Sie ausdrücklich darauf hin, daß vor

Do you agree to the public license and warranty? (Y/N) :
Source address is 137.74.87.167; using ICMP echo-request, ID=4808
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:
  Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.484270 sec
  RTTs in ms: min/avg/max/dev: 0.545 / 0.668 / 1.083 / 0.099
  Bandwidth in kbytes/sec: sent=10.267, rcvd=10.267  6
Last 10 seconds:
Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)
  Packets: sent=21, rcvd=21, error=0, lost=0 (0.0% loss) in 9.985041 sec
  RTTs in ms: min/avg/max/dev: 0.545 / 0.646 / 0.760 / 0.045
  Bandwidth in kbytes/sec: sent=10.574, rcvd=10.574  6

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.484270 sec
RTTs in ms: min/avg/max/dev: 0.545 / 0.668 / 1.083 / 0.099
Bandwidth in kbytes/sec: sent=10.267, rcvd=10.267

C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping>
```

Source address is 137.74.87.167; using ICMP echo-request, ID=4808
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.484270 sec
RTTs in ms: min/avg/max/dev: 0.545 / 0.668 / 1.083 / 0.099
Bandwidth in kbytes/sec: sent=10.267, rcvd=10.267 6

Last 10 seconds:

Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)

Packets: sent=21, rcvd=21, error=0, lost=0 (0.0% loss) in 9.985041 sec
RTTs in ms: min/avg/max/dev: 0.545 / 0.646 / 0.760 / 0.045
Bandwidth in kbytes/sec: sent=10.574, rcvd=10.574 6

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.484270 sec
RTTs in ms: min/avg/max/dev: 0.545 / **0.668** / 1.083 / 0.099
Bandwidth in kbytes/sec: sent=10.267, rcvd=10.267

Second SSP Server (V-CLUST-S2)

```
Administrator: Command Prompt
aufgeführten Lizenzbedingungen vom Nutzer anerkannt und
eingehalten werden. Es kommt damit zwischen ihm, dem
Lizenznehmer, und der Lizenzgeberin der vorliegende Lizenzvertrag
zustande:

2. Installation:

Die Lizenzgeberin weist Sie ausdrücklich darauf hin, daß vor

Do you agree to the public license and warranty? (Y/N) :
Source address is 137.74.87.168; using ICMP echo-request, ID=ac16
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:
  Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.499875 sec
  RTTs in ms: min/avg/max/dev: 0.536 / 0.699 / 1.070 / 0.074
  Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261  3
Last 10 seconds:
Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)
  Packets: sent=20, rcvd=20, error=0, lost=0 (0.0% loss) in 9.500592 sec
  RTTs in ms: min/avg/max/dev: 0.615 / 0.718 / 1.070 / 0.092
  Bandwidth in kbytes/sec: sent=10.584, rcvd=10.584  3
Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.499875 sec
RTTs in ms: min/avg/max/dev: 0.536 / 0.699 / 1.070 / 0.074
Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261

C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\hrping>
```

Source address is 137.74.87.168; using ICMP echo-request, ID=ac16
Pinging V-CLUST-SQL [137.74.87.170]
with 5000 bytes data (5028 bytes IP):

Total:
Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.499875 sec
RTTs in ms: min/avg/max/dev: 0.536 / 0.699 / 1.070 / 0.074
Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261 3
Last 10 seconds:
Last 10 seconds:0, rcvd=0, error=0, lost=0 (0.0% loss)
Packets: sent=20, rcvd=20, error=0, lost=0 (0.0% loss) in 9.500592 sec
RTTs in ms: min/avg/max/dev: 0.615 / 0.718 / 1.070 / 0.092
Bandwidth in kbytes/sec: sent=10.584, rcvd=10.584 3

Packets: sent=50, rcvd=50, error=0, lost=0 (0.0% loss) in 24.499875 sec
RTTs in ms: min/avg/max/dev: 0.536 / **0.699** / 1.070 / 0.074
Bandwidth in kbytes/sec: sent=10.261, rcvd=10.261

Conclusion

Network latency is below the recommended value, which is 1ms. This is what we want to have between the application servers and the database server.

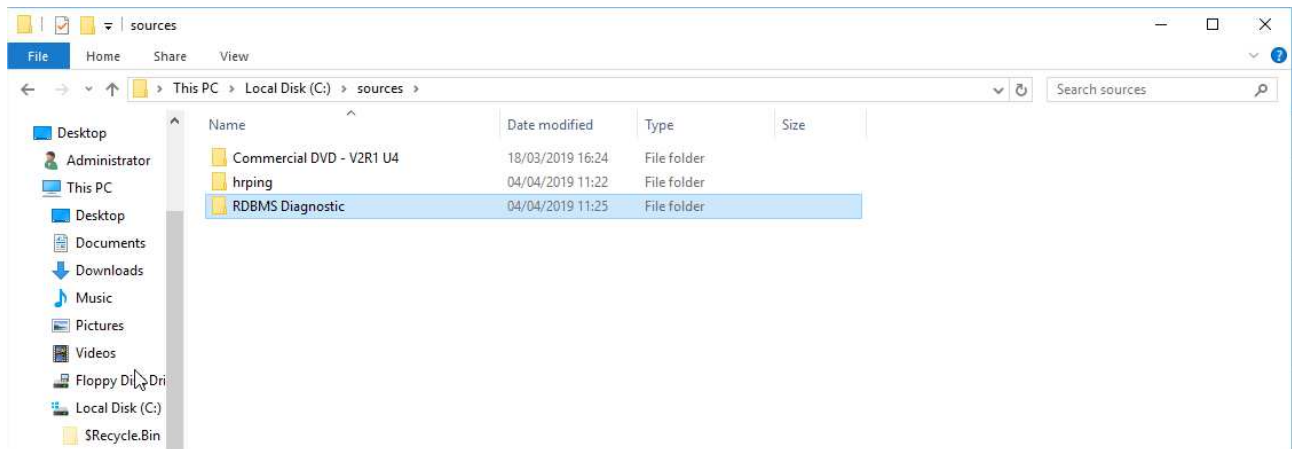
"RDBMS Diagnostic" Tool

This tool allows us to make a kind of stress test between the application and the database, by playing different types of SQL queries on a database and measure the response time.

It is split between different types of tests to check the infrastructure, and the capability of the database server to execute, as fast as possible, queries that are typical of how the application is going to trigger.

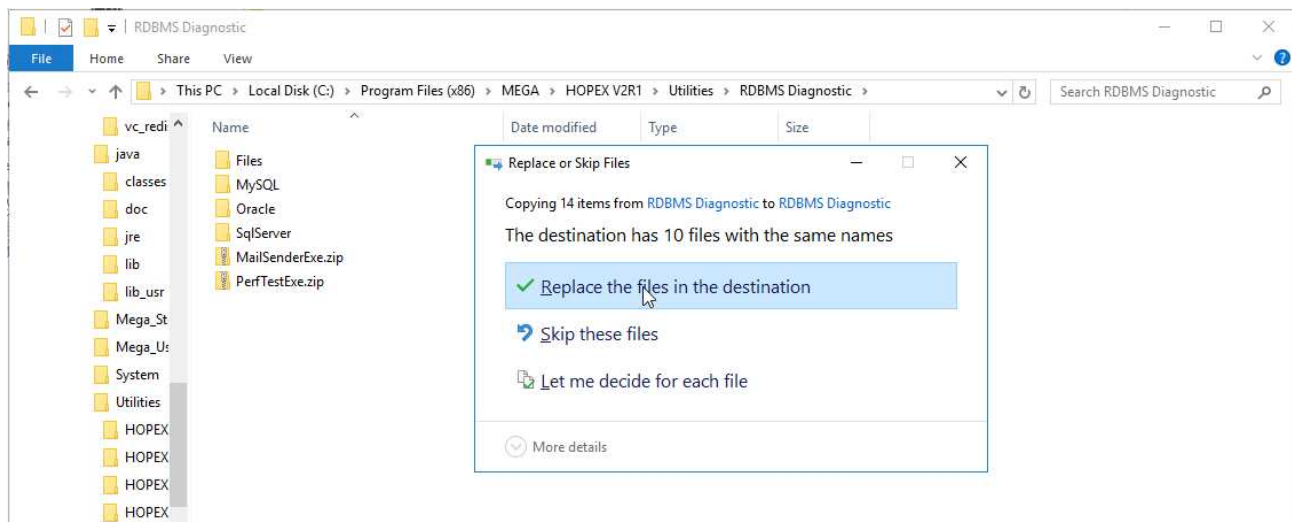
We will use the already existing “HOPEX_SystemDb” database to run the test.

Please note that the version delivered with Update 04 **doesn't work**. We got the version from Update 03 FP01 and copied it on all application servers in “C:\sources”:

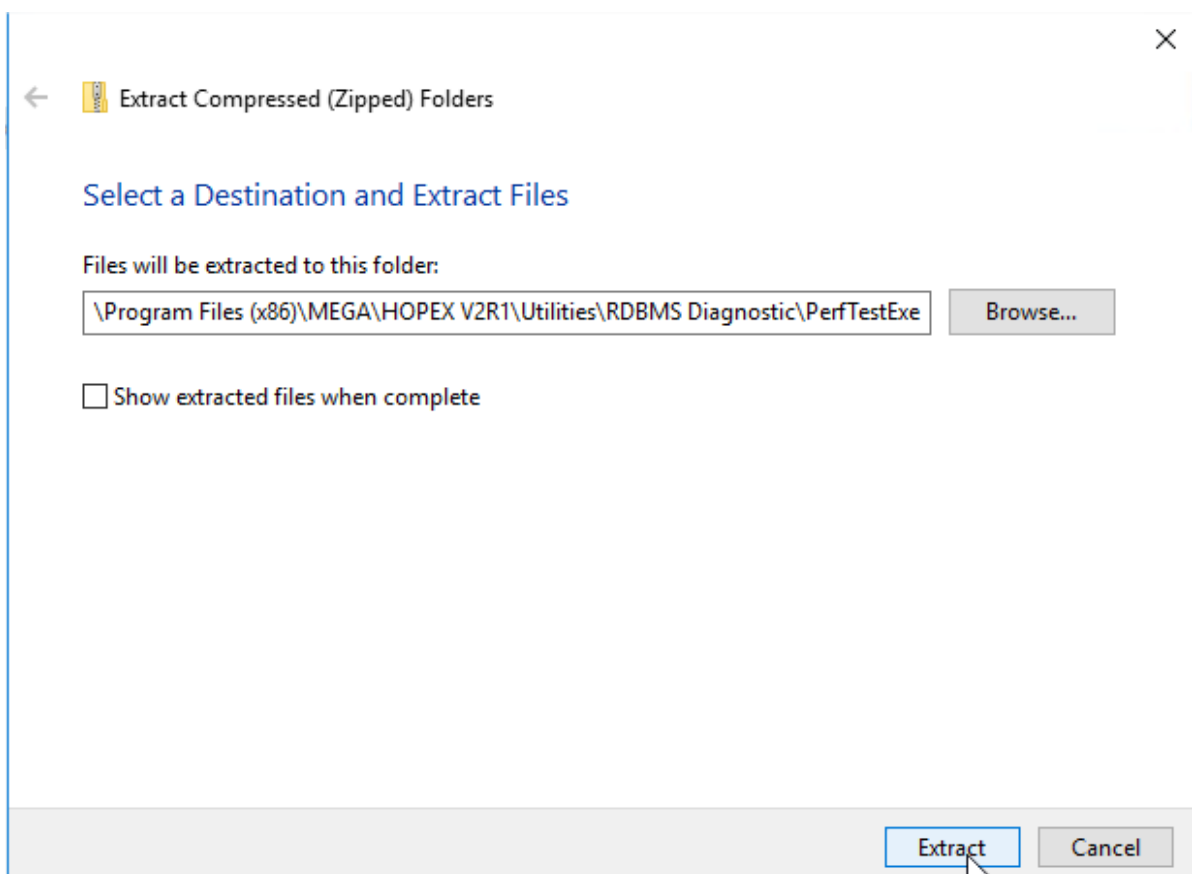
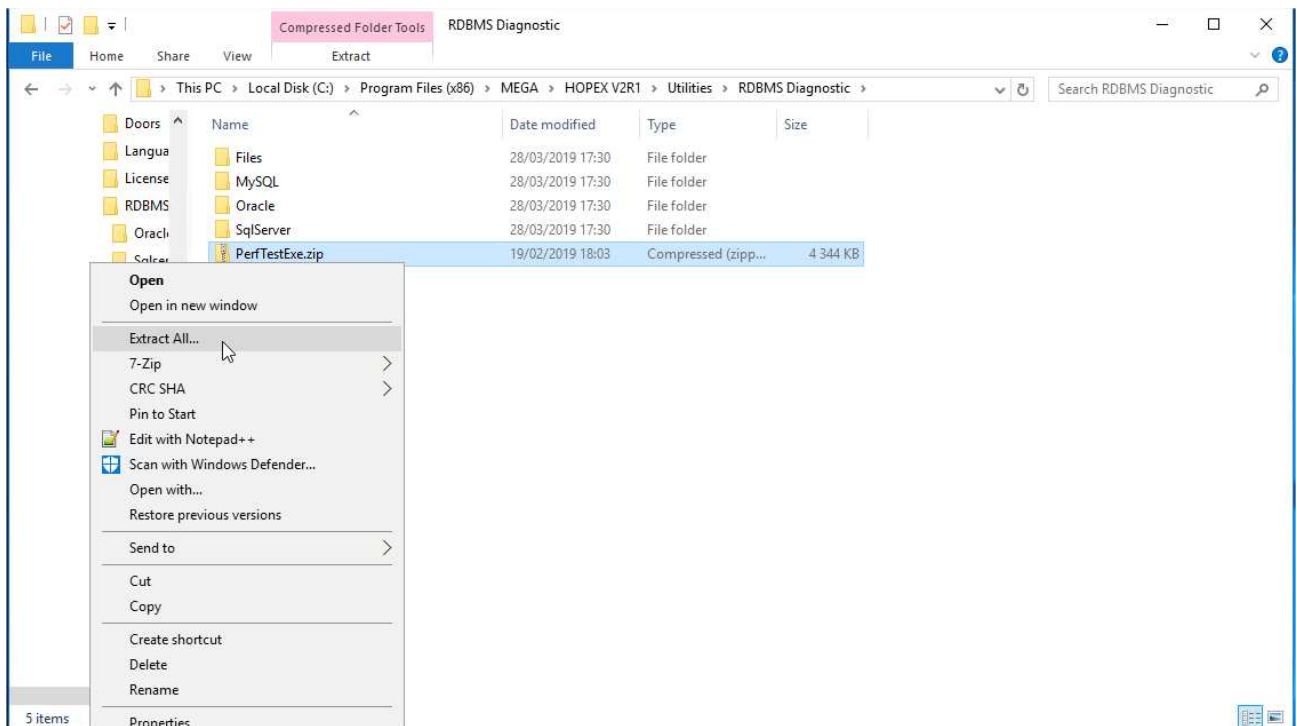


First, we need to prepare the servers so that the tool can be launched.

We go to the “C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\RDBMS Diagnostic” folder, in the Mega binaries, and replace the content of the “RDBMS Diagnostic” with what was put in the sources:

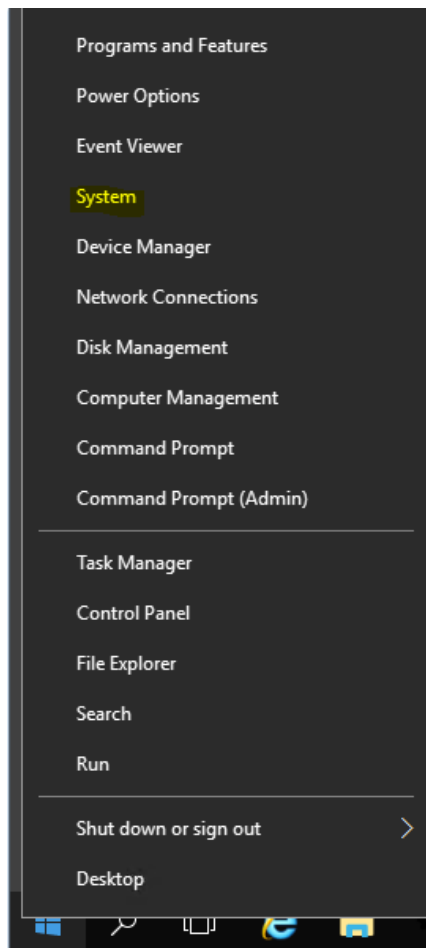


Then we extract the content of the “PerfTestExe.zip” archive file:

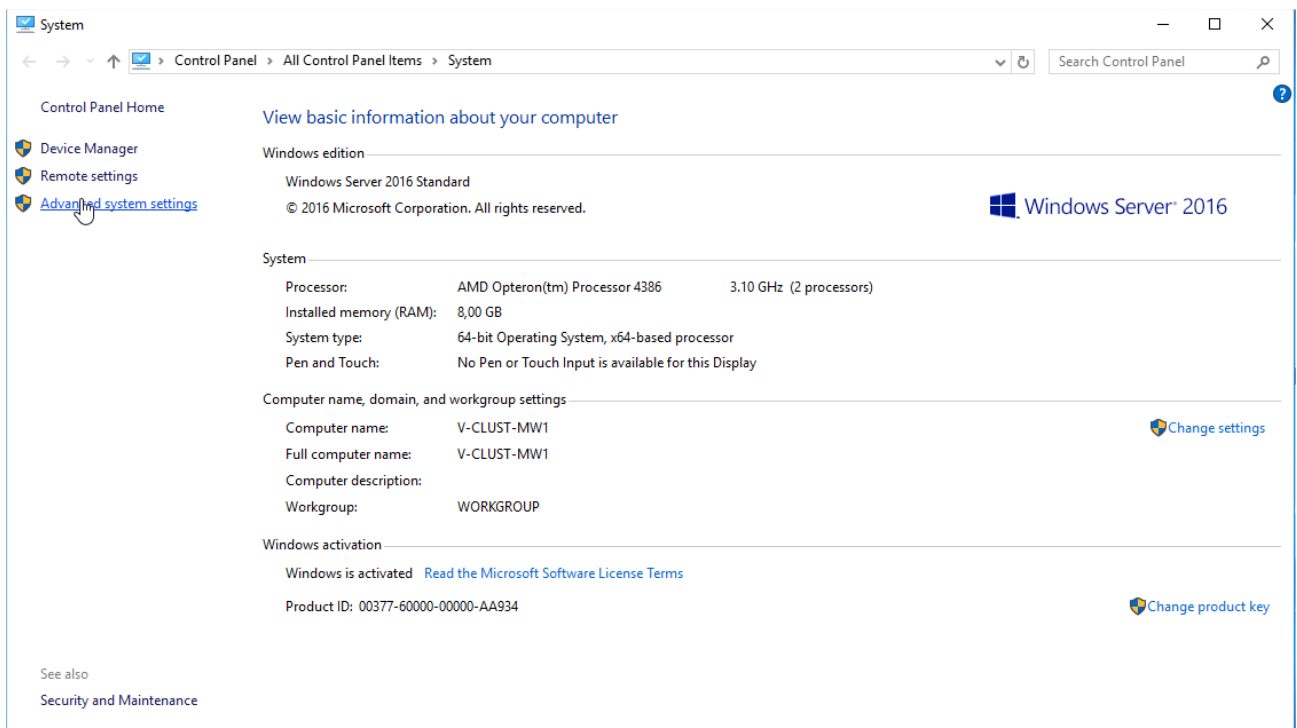


We now need to create an environment variable, so that the tool can know where the Java executable is being stored.

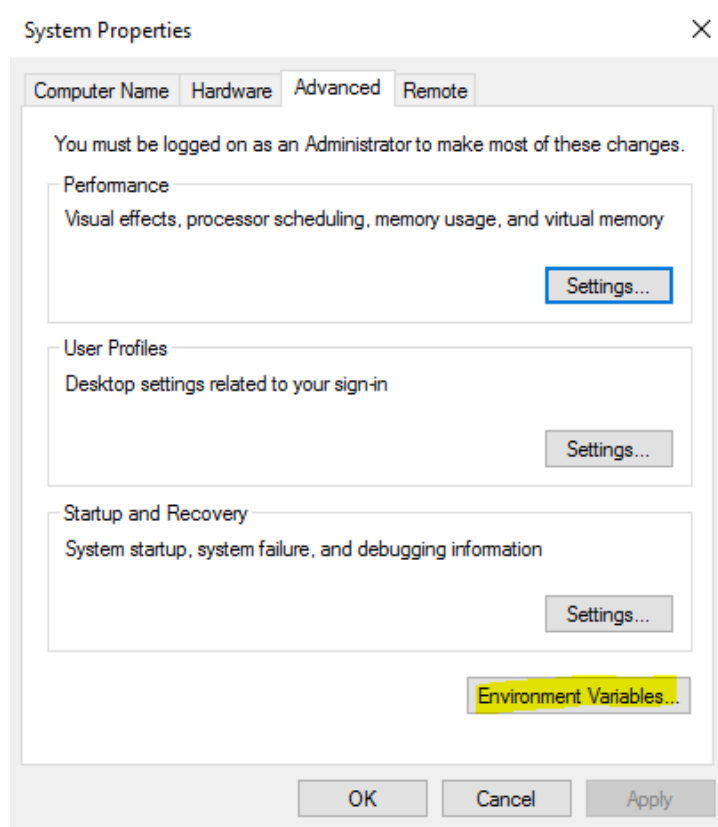
We open the "System" console (right-click the "Start" button):



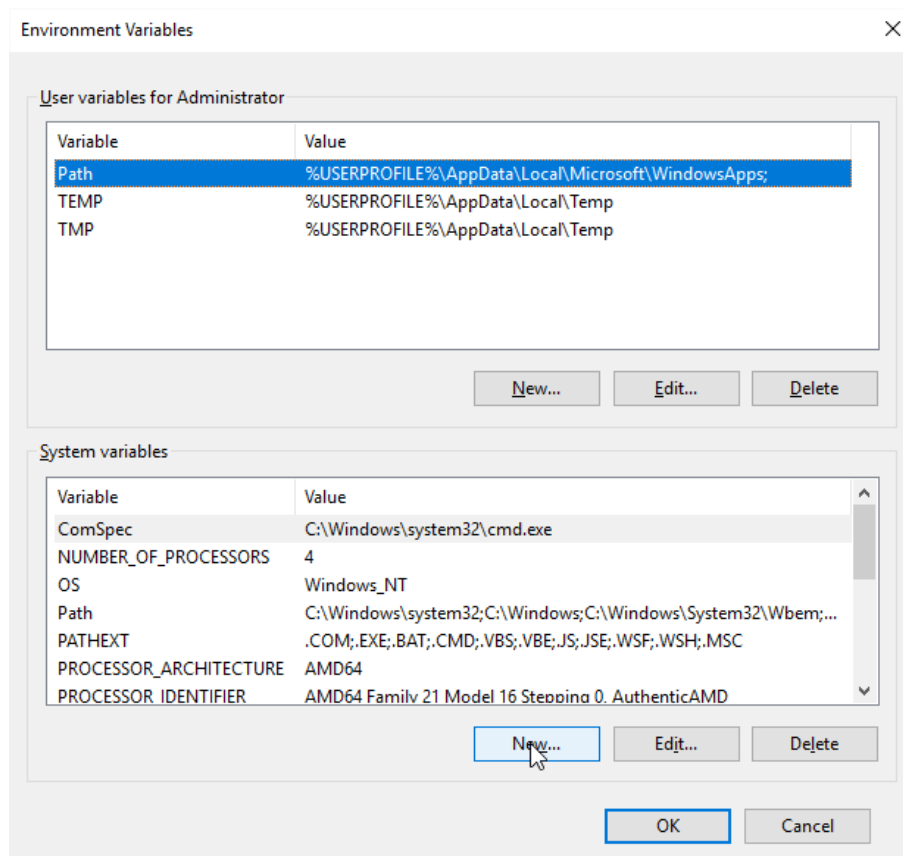
We click “Advanced system settings”:



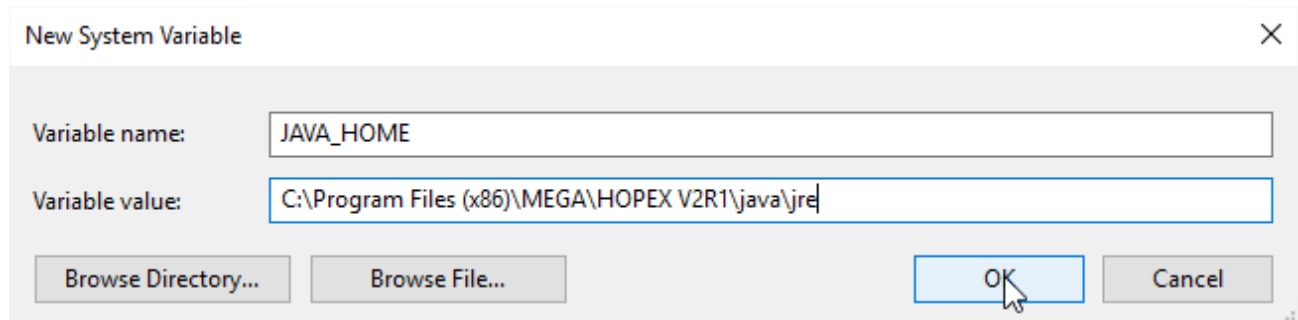
In the “Advanced” tab, we click “Environment variables”:



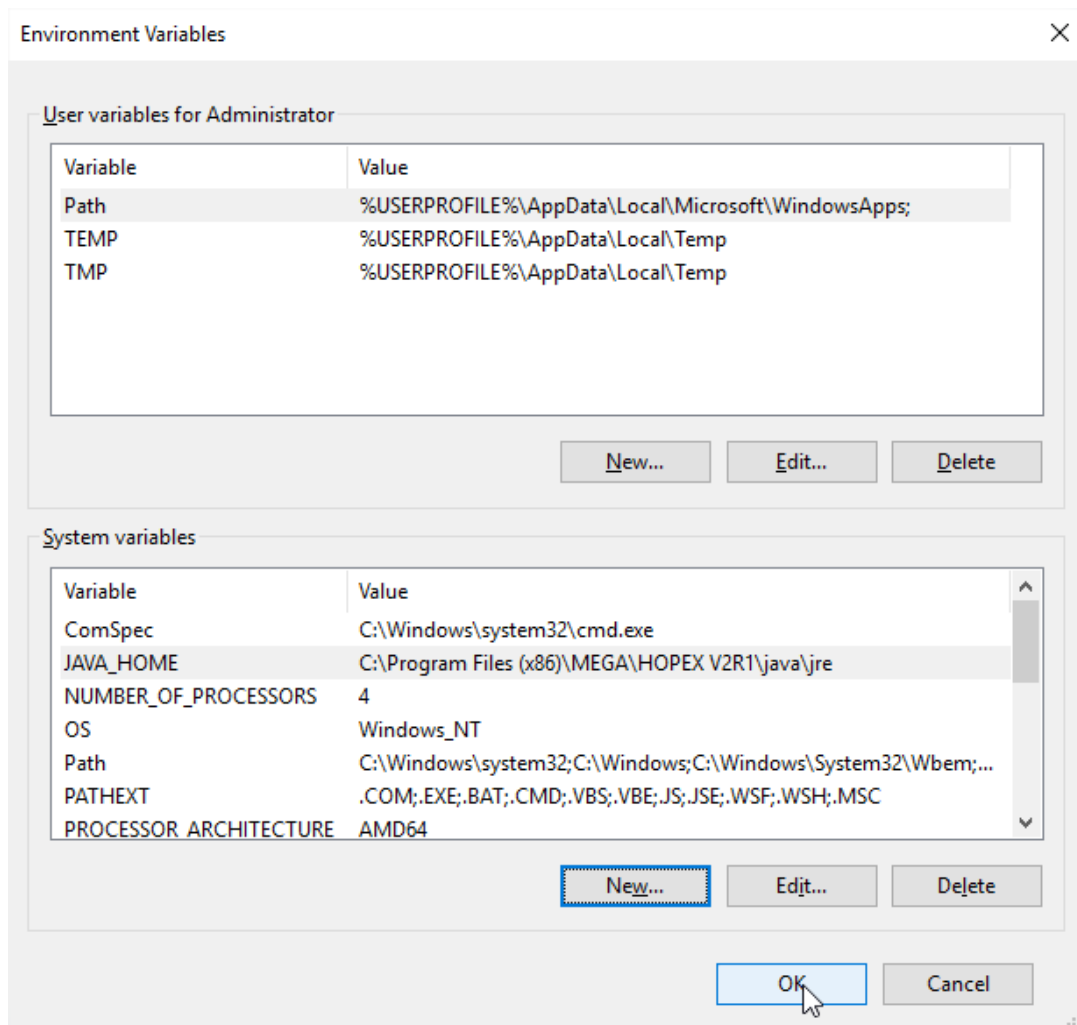
In “System variables” we click “New”:



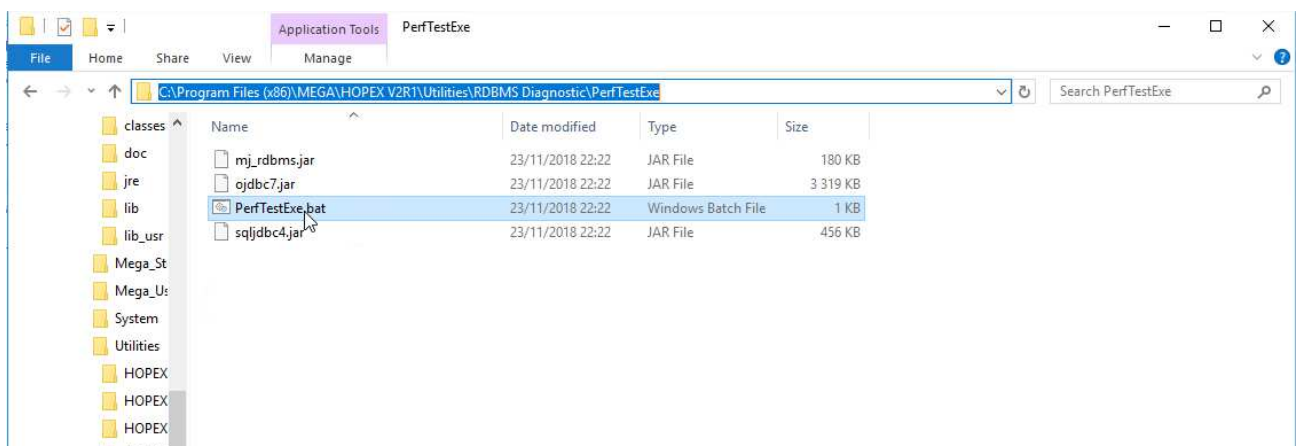
We create the variable called “JAVA_HOME” that will target the jre folder of the Mega binaries, so in this installation it is in “C:\Program Files (x86)\MEGA\HOPEX V2R1\java\jre”, on click “OK” to validate:



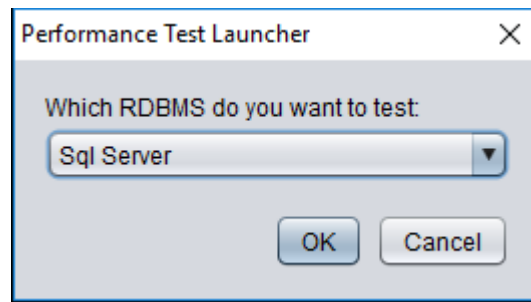
The variable is created, we can click “OK” and close this console:



Now we can go back to “C:\Program Files (x86)\MEGA\HOPEX V2R1\Utilities\RDBMS Diagnostic\PerfTestExe” and launch the “PerfTestExe.bat” file:



Choose “SQL Server” in the dropdown list:



Fill-in the instance name (be careful, in our case, the syntax is a bit different here than in Hopex, it is "V-CLUST-SQL:1436"). We chose "HOPEX_SystemDb" for the database. And "MEGAUSR" for the SQL account:

MEGA RDBMS Diagnostic Utility

Server Name : V-CLUST-SQL:1436

Base Name : HOPEX_SystemDb

Login : MEGAUSR

Password : *****

Test Name	Execution Time (ms)	Expected Execution Time ...	Test Result
<input checked="" type="checkbox"/> DDL	0	20	
<input checked="" type="checkbox"/> INSERT (LIGHT)	0	29000	
<input checked="" type="checkbox"/> INSERT (LIGHT, server le...	0	4300	
<input checked="" type="checkbox"/> INSERT (HEAVY)	0	14000	
<input checked="" type="checkbox"/> READ (LIGHT)	0	9000	
<input checked="" type="checkbox"/> READ (HEAVY)	0	34000	
<input checked="" type="checkbox"/> SERVER CPU SPEED	0	7500	
<input checked="" type="checkbox"/> SERVER DISK	0	20000	
<input checked="" type="checkbox"/> SERVER DISK (BLOB's)	0	20000	
<input checked="" type="checkbox"/> BANDWIDTH	0	24000	
<input checked="" type="checkbox"/> BANDWIDTH (BLOB's)	0	40000	
<input checked="" type="checkbox"/> RESET DB	0	100	

Test Description :

By clicking on a test, you will have a short description of it.

Diagnostic :

☐ Auto Commit ☐ Loop Test

Copy Diagnostic to Clipboard

Start Tests

Stop Tests

Close

Then click « Start Tests ». And do it twice, to have better results.

First MWAS Server (V-CLUST-MW1)

MEGA RDBMS Diagnostic Utility

Server Name : V-CLUST-SQL:1436

Base Name : HOPEX_SystemDb

Login : MEGAUSR

Password : *****

Test Name	Execution Time (ms)	Expected Execution Time ...	Test Result
<input checked="" type="checkbox"/> DDL	16	20	Ok
<input checked="" type="checkbox"/> INSERT (LIGHT)	38702	29000	too long
<input checked="" type="checkbox"/> INSERT (LIGHT, server le...	3063	4300	Ok
<input checked="" type="checkbox"/> INSERT (HEAVY)	12968	14000	Ok
<input checked="" type="checkbox"/> READ (LIGHT)	6015	9000	Ok
<input checked="" type="checkbox"/> READ (HEAVY)	23062	34000	Ok
<input checked="" type="checkbox"/> SERVER CPU SPEED	6156	7500	Ok
<input checked="" type="checkbox"/> SERVER DISK	17734	20000	Ok
<input checked="" type="checkbox"/> SERVER DISK (BLOB's)	17625	20000	Ok
<input checked="" type="checkbox"/> BANDWIDTH	15374	24000	Ok
<input checked="" type="checkbox"/> BANDWIDTH (BLOB's)	5219	40000	Ok
<input checked="" type="checkbox"/> RESET DB	0	100	Ok

Test Description :

By clicking on a test, you will have a short description of it.

Diagnostic :

```
##### Start Batch Test: Thu Apr 04 11:34:11 CEST 2019 #####
##### Autocommit mode is      OFF      #####
TEST 1 (DDL):
NOK: time=47ms , expected time=20ms
TEST 2 (INSERT (LIGHT)):
PASSABLE: time=35952ms , expected time=29000ms
```

☐ Auto Commit ☐ Loop Test

Copy Diagnostic to Clipboard

Start Tests Stop Tests Close

Results of the test in text mode:

```
##### Start Batch Test: Thu Apr 04 11:34:11 CEST 2019 #####
##### Autocommit mode is      OFF      #####
TEST 1 (DDL):
NOK: time=47ms , expected time=20ms
```


TEST 2 (INSERT (LIGHT)):
 PASSABLE: time=35952ms , expected time=29000ms

TEST 3 (INSERT (LIGHT, server level)):
 OK: time=3500ms , expected time=4300ms

TEST 4 (INSERT (HEAVY)):
 PASSABLE: time=17078ms , expected time=14000ms

TEST 5 (READ (LIGHT)):
 OK: time=6515ms , expected time=9000ms

TEST 6 (READ (HEAVY)):
 OK: time=23922ms , expected time=34000ms

TEST 7 (SERVER CPU SPEED):
 OK: time=6250ms , expected time=7500ms

TEST 8 (SERVER DISK):
 OK: time=20780ms , expected time=20000ms

TEST 9 (SERVER DISK (BLOB's)):
 OK: time=17406ms , expected time=20000ms

TEST 10 (BANDWIDTH):
 OK: time=14578ms , expected time=24000ms

TEST 11 (BANDWIDTH (BLOB's)):
 OK: time=5078ms , expected time=40000ms

TEST 12 (RESET DB):
 OK: time=16ms , expected time=100ms

Batch Test Finished: Thu Apr 04 11:36:43 CEST 2019

Start Batch Test: Thu Apr 04 11:36:49 CEST 2019

Autocommit mode is OFF

TEST 1 (DDL):
 OK: time=16ms , expected time=20ms

TEST 2 (INSERT (LIGHT)):
 NOK: time=38702ms , expected time=29000ms

TEST 3 (INSERT (LIGHT, server level)):
 OK: time=3063ms , expected time=4300ms

TEST 4 (INSERT (HEAVY)):
 OK: time=12968ms , expected time=14000ms

TEST 5 (READ (LIGHT)):
 OK: time=6015ms , expected time=9000ms

TEST 6 (READ (HEAVY)):
 OK: time=23062ms , expected time=34000ms

TEST 7 (SERVER CPU SPEED):
 OK: time=6156ms , expected time=7500ms

TEST 8 (SERVER DISK):
 OK: time=17734ms , expected time=20000ms

TEST 9 (SERVER DISK (BLOB's)):
 OK: time=17625ms , expected time=20000ms

TEST 10 (BANDWIDTH):
 OK: time=15374ms , expected time=24000ms

TEST 11 (BANDWIDTH (BLOB's)):
 OK: time=5219ms , expected time=40000ms

TEST 12 (RESET DB):
 OK: time=0ms , expected time=100ms

Batch Test Finished: Thu Apr 04 11:39:14 CEST 2019

Second MWAS Server (V-CLUST-MW2)

MEGA RDBMS Diagnostic Utility

Server Name : V-CLUST-SQL:1436

Base Name : HOPEX_SystemDb

Login : MEGAUSR

Password : *****

Test Name	Execution Time (ms)	Expected Execution Time ...	Test Result
✓ DDL	0	20	Ok
✓ INSERT (LIGHT)	27546	29000	Ok
✓ INSERT (LIGHT, server le...	3016	4300	Ok
✓ INSERT (HEAVY)	7937	14000	Ok
✓ READ (LIGHT)	6234	9000	Ok
✓ READ (HEAVY)	22202	34000	Ok
✓ SERVER CPU SPEED	6312	7500	Ok
✓ SERVER DISK	16938	20000	Ok
✓ SERVER DISK (BLOB's)	21296	20000	Ok
✓ BANDWIDTH	17702	24000	Ok
✓ BANDWIDTH (BLOB's)	6032	40000	Ok
✓ RESET DB	15	100	Ok

Test Description :

By clicking on a test, you will have a short description of it.

Diagnostic :

OK: time=17702ms , expected time=24000ms
TEST 11 (BANDWIDTH (BLOB's)):
OK: time=6032ms , expected time=40000ms
TEST 12 (RESET DB):
OK: time=15ms , expected time=100ms
Batch Test Finished: Thu Apr 04 11:45:35 CEST 2019

☐ Auto Commit ☐ Loop Test

Copy Diagnostic to Clipboard

Start Tests Stop Tests Close

Results of the test in text mode:

Start Batch Test: Thu Apr 04 11:40:31 CEST 2019

Autocommit mode is OFF

TEST 1 (DDL):
 NOK: time=31ms , expected time=20ms
 TEST 2 (INSERT (LIGHT)):
 OK: time=28843ms , expected time=29000ms
 TEST 3 (INSERT (LIGHT, server level)):
 OK: time=3093ms , expected time=4300ms
 TEST 4 (INSERT (HEAVY)):
 OK: time=11250ms , expected time=14000ms
 TEST 5 (READ (LIGHT)):
 OK: time=6094ms , expected time=9000ms
 TEST 6 (READ (HEAVY)):
 OK: time=23046ms , expected time=34000ms
 TEST 7 (SERVER CPU SPEED):
 OK: time=6312ms , expected time=7500ms
 TEST 8 (SERVER DISK):
 OK: time=16984ms , expected time=20000ms
 TEST 9 (SERVER DISK (BLOB's)):
 OK: time=16937ms , expected time=20000ms
 TEST 10 (BANDWIDTH):
 OK: time=15594ms , expected time=24000ms
 TEST 11 (BANDWIDTH (BLOB's)):
 OK: time=5015ms , expected time=40000ms
 TEST 12 (RESET DB):
 OK: time=16ms , expected time=100ms
 ##### Batch Test Finished: Thu Apr 04 11:42:45 CEST 2019 #####
 ##### Start Batch Test: Thu Apr 04 11:43:20 CEST 2019 #####
 ##### Autocommit mode is OFF #####
 TEST 1 (DDL):
 OK: time=0ms , expected time=20ms
 TEST 2 (INSERT (LIGHT)):
 OK: time=27546ms , expected time=29000ms
 TEST 3 (INSERT (LIGHT, server level)):
 OK: time=3016ms , expected time=4300ms
 TEST 4 (INSERT (HEAVY)):
 OK: time=7937ms , expected time=14000ms
 TEST 5 (READ (LIGHT)):
 OK: time=6234ms , expected time=9000ms
 TEST 6 (READ (HEAVY)):
 OK: time=22202ms , expected time=34000ms
 TEST 7 (SERVER CPU SPEED):
 OK: time=6312ms , expected time=7500ms
 TEST 8 (SERVER DISK):
 OK: time=16938ms , expected time=20000ms
 TEST 9 (SERVER DISK (BLOB's)):
 OK: time=21296ms , expected time=20000ms
 TEST 10 (BANDWIDTH):
 OK: time=17702ms , expected time=24000ms
 TEST 11 (BANDWIDTH (BLOB's)):
 OK: time=6032ms , expected time=40000ms

TEST 12 (RESET DB):

OK: time=15ms , expected time=100ms

Batch Test Finished: Thu Apr 04 11:45:35 CEST 2019

First SSP Server (V-CLUST-S1)

MEGA RDBMS Diagnostic Utility

Server Name : V-CLUST-SQL:1436

Base Name : HOPEX_SystemDb

Login : MEGAUSR

Password : *****

Test Name	Execution Time (ms)	Expected Execution Time ...	Test Result
✓ DDL	16	20	Ok
✓ INSERT (LIGHT)	34406	29000	Acceptable
✓ INSERT (LIGHT, server le...	3266	4300	Ok
✓ INSERT (HEAVY)	16203	14000	Acceptable
✓ READ (LIGHT)	6500	9000	Ok
✓ READ (HEAVY)	23469	34000	Ok
✓ SERVER CPU SPEED	5594	7500	Ok
✓ SERVER DISK	20375	20000	Ok
✓ SERVER DISK (BLOB's)	18953	20000	Ok
✓ BANDWIDTH	15344	24000	Ok
✓ BANDWIDTH (BLOB's)	4828	40000	Ok
✓ RESET DB	16	100	Ok

Test Description :

By clicking on a test, you will have a short description of it.

Diagnostic :

OK: time=15344ms , expected time=24000ms
TEST 11 (BANDWIDTH (BLOB's)):
OK: time=4828ms , expected time=40000ms
TEST 12 (RESET DB):
OK: time=16ms , expected time=100ms
Batch Test Finished: Thu Apr 04 12:09:29 CEST 2019

☐ Auto Commit ☐ Loop Test

Copy Diagnostic to Clipboard

Start Tests Stop Tests Close

Results of the test in text mode:

```

##### Start Batch Test: Thu Apr 04 11:58:43 CEST 2019 #####
##### Autocommit mode is          OFF          #####
TEST 1 (DDL):
  OK: time=15ms , expected time=20ms
TEST 2 (INSERT (LIGHT)):
  PASSABLE: time=36814ms , expected time=29000ms
TEST 3 (INSERT (LIGHT, server level)):
  OK: time=3281ms , expected time=4300ms
TEST 4 (INSERT (HEAVY)):
  OK: time=10766ms , expected time=14000ms
TEST 5 (READ (LIGHT)):
  OK: time=6547ms , expected time=9000ms
TEST 6 (READ (HEAVY)):
  OK: time=23688ms , expected time=34000ms
TEST 7 (SERVER CPU SPEED):
  OK: time=6719ms , expected time=7500ms
TEST 8 (SERVER DISK):
  PASSABLE: time=24391ms , expected time=20000ms
TEST 9 (SERVER DISK (BLOB's)):
  OK: time=21532ms , expected time=20000ms
TEST 10 (BANDWIDTH):
  OK: time=16844ms , expected time=24000ms
TEST 11 (BANDWIDTH (BLOB's)):
  OK: time=4671ms , expected time=40000ms
TEST 12 (RESET DB):
  OK: time=110ms , expected time=100ms
##### Batch Test Finished: Thu Apr 04 12:01:19 CEST 2019 #####
##### Start Batch Test: Thu Apr 04 12:07:00 CEST 2019 #####
##### Autocommit mode is          OFF          #####
TEST 1 (DDL):
  OK: time=16ms , expected time=20ms
TEST 2 (INSERT (LIGHT)):
  PASSABLE: time=34406ms , expected time=29000ms
TEST 3 (INSERT (LIGHT, server level)):
  OK: time=3266ms , expected time=4300ms
TEST 4 (INSERT (HEAVY)):
  PASSABLE: time=16203ms , expected time=14000ms
TEST 5 (READ (LIGHT)):
  OK: time=6500ms , expected time=9000ms
TEST 6 (READ (HEAVY)):
  OK: time=23469ms , expected time=34000ms
TEST 7 (SERVER CPU SPEED):
  OK: time=5594ms , expected time=7500ms
TEST 8 (SERVER DISK):
  OK: time=20375ms , expected time=20000ms
TEST 9 (SERVER DISK (BLOB's)):
  OK: time=18953ms , expected time=20000ms
TEST 10 (BANDWIDTH):
  OK: time=15344ms , expected time=24000ms

```

TEST 11 (BANDWIDTH (BLOB's)):

OK: time=4828ms , expected time=40000ms

TEST 12 (RESET DB):

OK: time=16ms , expected time=100ms

Batch Test Finished: Thu Apr 04 12:09:29 CEST 2019

Second SSP Server (V-CLUST-S2)

MEGA RDBMS Diagnostic Utility

Server Name : V-CLUST-SQL:1436

Base Name : HOPEX_SystemDb

Login : MEGAUSR

Password : *****

Test Name	Execution Time (ms)	Expected Execution Time ...	Test Result
<input checked="" type="checkbox"/> DDL	16	20	Ok
<input checked="" type="checkbox"/> INSERT (LIGHT)	42093	29000	too long
<input checked="" type="checkbox"/> INSERT (LIGHT, server le...	3421	4300	Ok
<input checked="" type="checkbox"/> INSERT (HEAVY)	14328	14000	Ok
<input checked="" type="checkbox"/> READ (LIGHT)	6000	9000	Ok
<input checked="" type="checkbox"/> READ (HEAVY)	21937	34000	Ok
<input checked="" type="checkbox"/> SERVER CPU SPEED	5640	7500	Ok
<input checked="" type="checkbox"/> SERVER DISK	18688	20000	Ok
<input checked="" type="checkbox"/> SERVER DISK (BLOB's)	17702	20000	Ok
<input checked="" type="checkbox"/> BANDWIDTH	15672	24000	Ok
<input checked="" type="checkbox"/> BANDWIDTH (BLOB's)	5093	40000	Ok
<input checked="" type="checkbox"/> RESET DB	16	100	Ok

Test Description :

By clicking on a test, you will have a short description of it.

Diagnostic :

OK: time=15672ms , expected time=24000ms
TEST 11 (BANDWIDTH (BLOB's)):
OK: time=5093ms , expected time=40000ms
TEST 12 (RESET DB):
OK: time=16ms , expected time=100ms
Batch Test Finished: Thu Apr 04 14:52:53 CEST 2019

☐ Auto Commit ☐ Loop Test

Copy Diagnostic to Clipboard

Start Tests Stop Tests Close

Results of the test in text mode:

```
##### Start Batch Test: Thu Apr 04 14:42:35 CEST 2019 #####
##### Autocommit mode is          OFF          #####
TEST 1 (DDL):
  NOK: time=31ms , expected time=20ms
TEST 2 (INSERT (LIGHT)):
  PASSABLE: time=37030ms , expected time=29000ms
TEST 3 (INSERT (LIGHT, server level)):
  OK: time=3047ms , expected time=4300ms
TEST 4 (INSERT (HEAVY)):
  OK: time=12234ms , expected time=14000ms
TEST 5 (READ (LIGHT)):
  OK: time=6187ms , expected time=9000ms
TEST 6 (READ (HEAVY)):
  OK: time=23188ms , expected time=34000ms
TEST 7 (SERVER CPU SPEED):
  OK: time=5952ms , expected time=7500ms
TEST 8 (SERVER DISK):
  OK: time=20359ms , expected time=20000ms
TEST 9 (SERVER DISK (BLOB's)):
  PASSABLE: time=24797ms , expected time=20000ms
TEST 10 (BANDWIDTH):
  OK: time=15296ms , expected time=24000ms
TEST 11 (BANDWIDTH (BLOB's)):
  OK: time=5766ms , expected time=40000ms
TEST 12 (RESET DB):
  OK: time=31ms , expected time=100ms
##### Batch Test Finished: Thu Apr 04 14:45:09 CEST 2019 #####
##### Start Batch Test: Thu Apr 04 14:50:22 CEST 2019 #####
##### Autocommit mode is          OFF          #####
TEST 1 (DDL):
  OK: time=16ms , expected time=20ms
TEST 2 (INSERT (LIGHT)):
  NOK: time=42093ms , expected time=29000ms
TEST 3 (INSERT (LIGHT, server level)):
  OK: time=3421ms , expected time=4300ms
TEST 4 (INSERT (HEAVY)):
  OK: time=14328ms , expected time=14000ms
TEST 5 (READ (LIGHT)):
  OK: time=6000ms , expected time=9000ms
TEST 6 (READ (HEAVY)):
  OK: time=21937ms , expected time=34000ms
TEST 7 (SERVER CPU SPEED):
  OK: time=5640ms , expected time=7500ms
TEST 8 (SERVER DISK):
  OK: time=18688ms , expected time=20000ms
TEST 9 (SERVER DISK (BLOB's)):
  OK: time=17702ms , expected time=20000ms
```

TEST 10 (BANDWIDTH):
OK: time=15672ms , expected time=24000ms
TEST 11 (BANDWIDTH (BLOB's)):
OK: time=5093ms , expected time=40000ms
TEST 12 (RESET DB):
OK: time=16ms , expected time=100ms
Batch Test Finished: Thu Apr 04 14:52:53 CEST 2019

Conclusion

The results are good. The performances between the application layer and the database layer should be satisfactory.

Activation and deployment of stored procedures for SQL Server

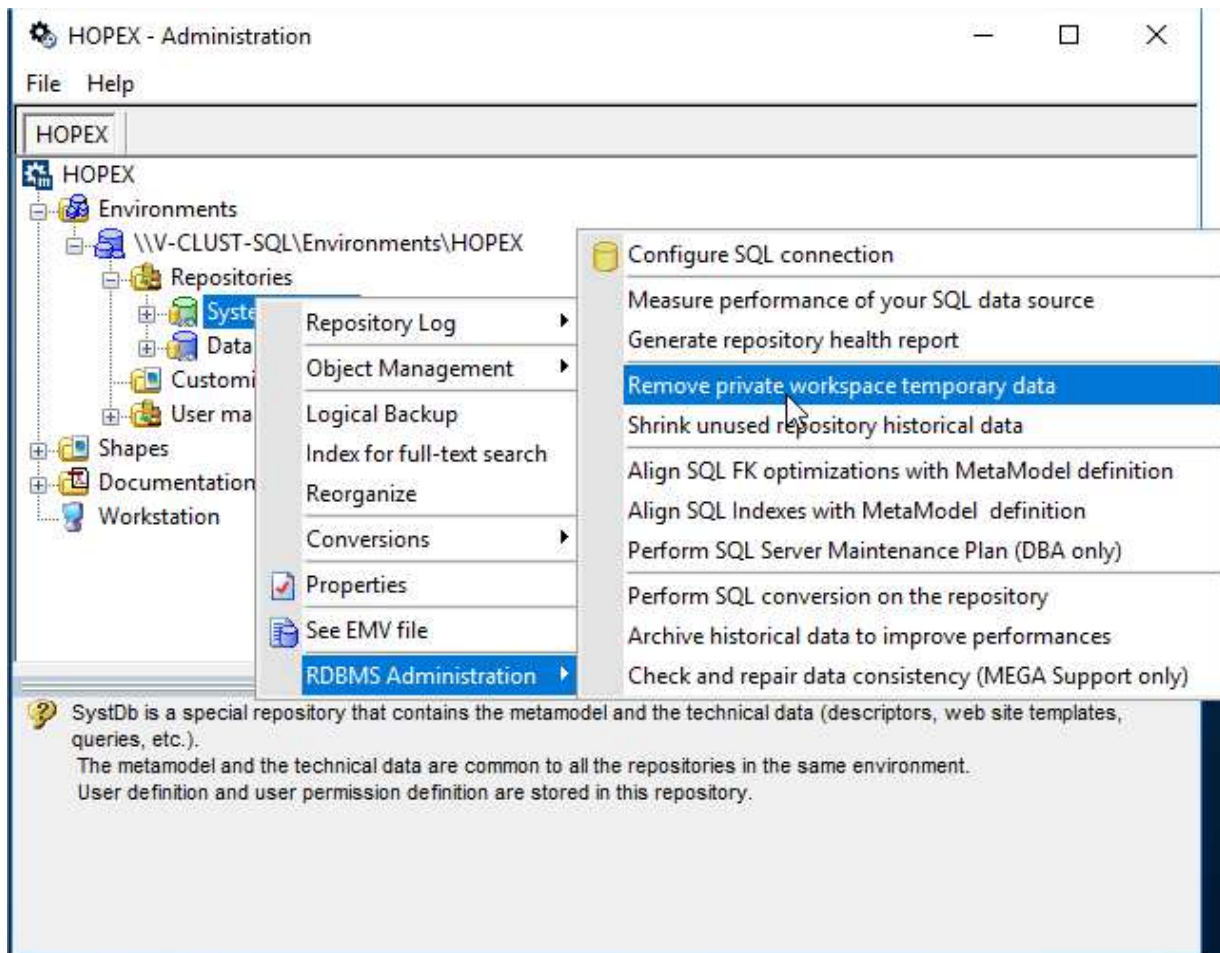
To be done once, from any of the application servers.

To avoid having the transaction tables (linked to the activity on the platform) to grow too much, so that they slow down operations such as connections or publications, it is necessary to deploy, on each Mega repository in RDBMS (in Oracle in this context), two stored procedures, and to schedule them to run regularly on the SQL Server databases.

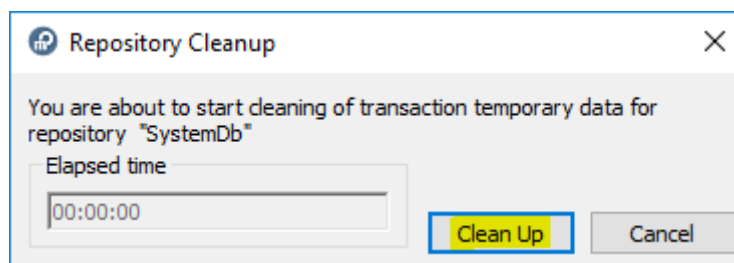
Warning ! The following steps need to be done outside of the periods of activity on the application, because it prevents new users to connect, or connected users to publish their work.

To create those, and execute those a first time, we use the Administration module:

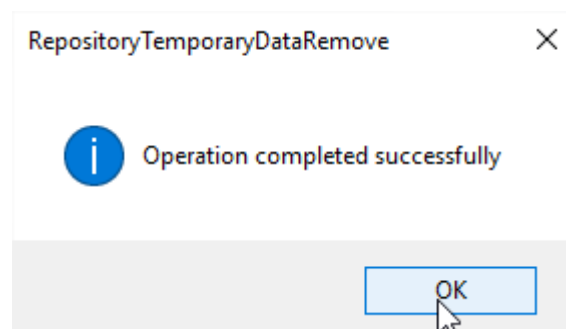
1. Start the « Administration.exe ».
2. Connect to the environnement on which we want to perform those operations.
3. Expand the repositories.
4. Right-click each repository, starting with the SystemDb : choose « RDBMS Administration », then « Remove Private workspace temporary data »



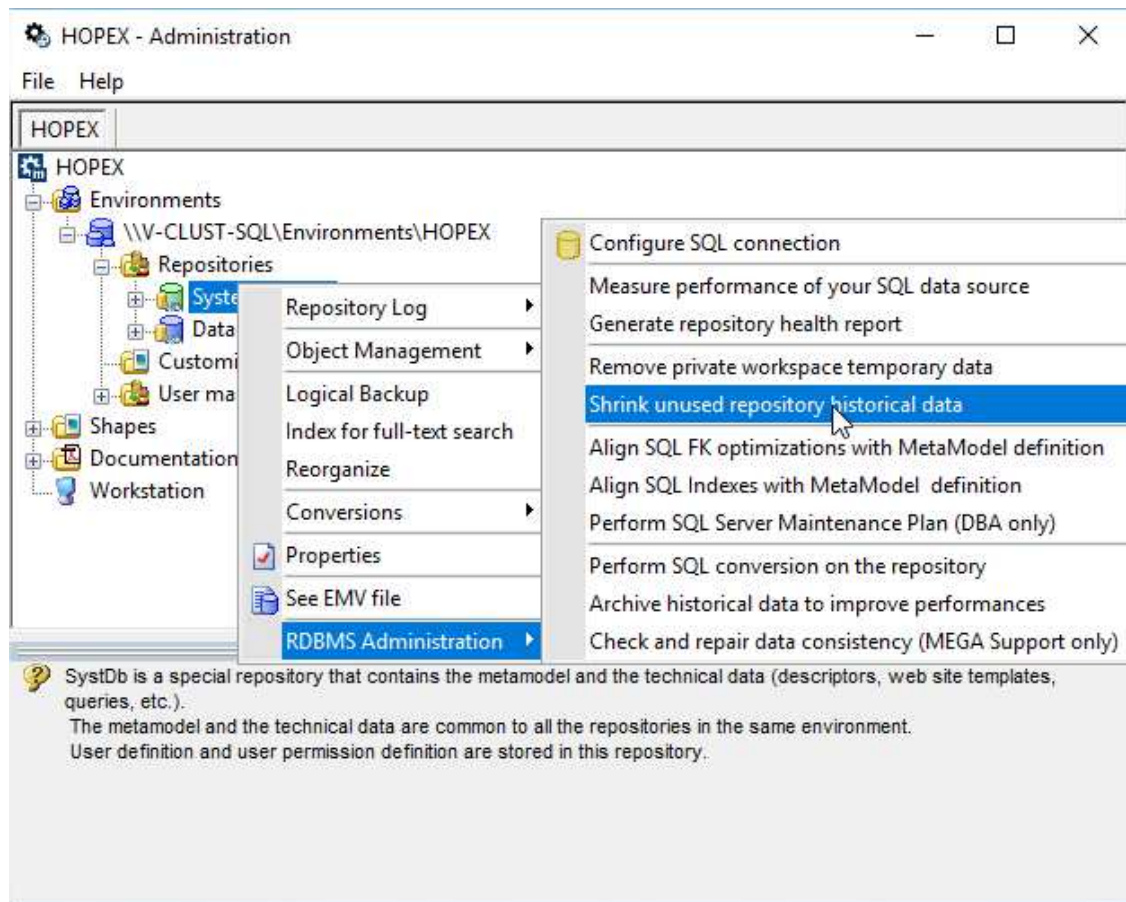
5. This opens up the following window, where you have to click « Clean Up », which will have the effect of creating a stored procedure in the SQL Server related database, and to execute it at once :



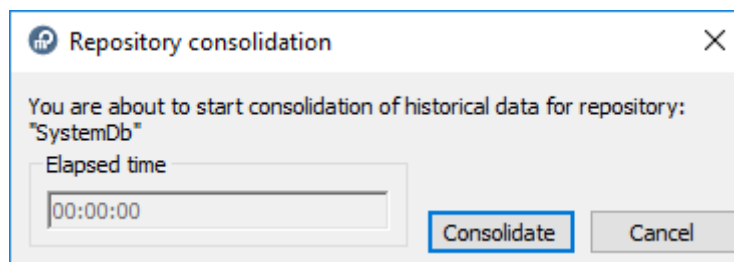
6. Click „OK“:



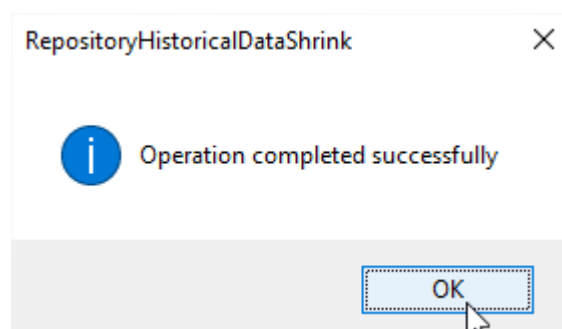
7. Do the same operation with the second option in « RDBMS Administration », called this time « Shrink unused repository historical data », that will create the second stored procedure and execute it, by pressing the “Consolidate” button :



8. Click « Consolidate »:



9. Click „OK“:



10. Reproduce all those steps **on all Mega repositories for all environments** (here we have only one).

Those must be executed on a regular basis, at least once a week, to avoid keeping too much logs on the transactions, and thus, making the application slow down during phases like login or publish. The names are:

- `dbo.SP_CLEAN_MEGA_DATABASE`
- `dbo.SP_CONSOLIDATE_MEGA_DATABASE`

ANNEX

The log files

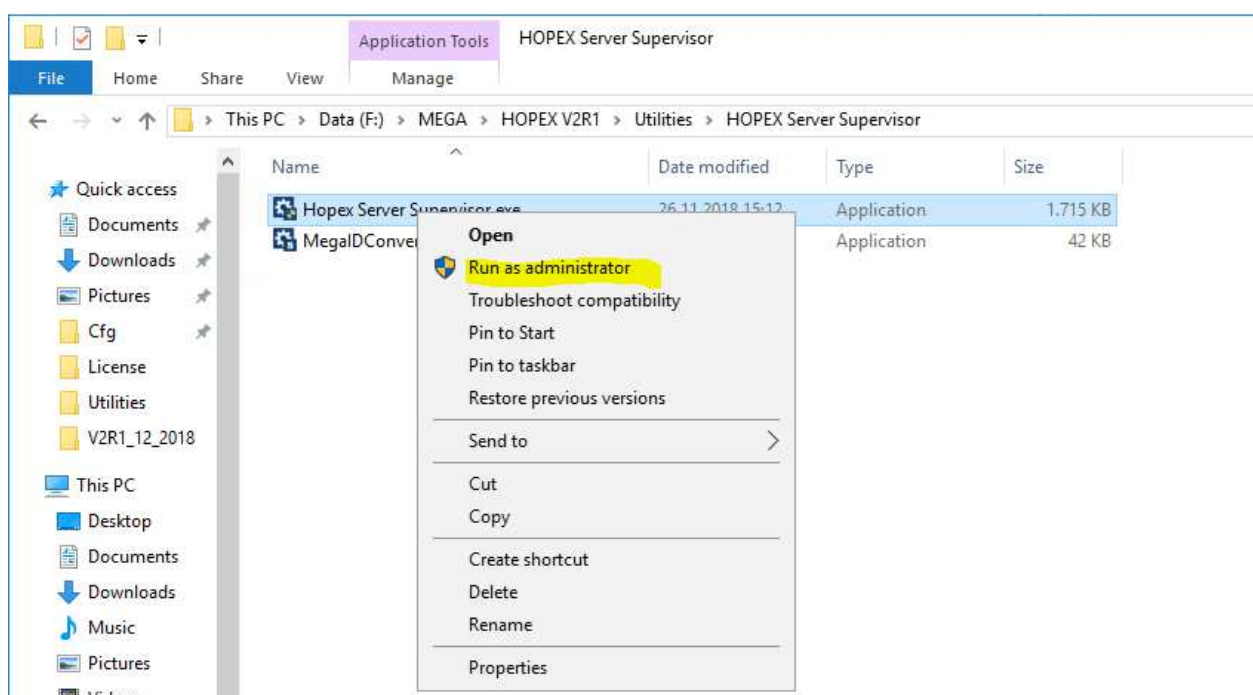
By default, all logs are located in “C:\ProgramData\Mega\Logs”, except the supervision log:

- MWAS log (thin client, to trace connection of users): MWASLOG*.TXT files.
- Rich client log: megaerr*.txt files.
- SSP error log (errors and traces when connected to the web client and working in it): SSPERR*.TXT files.
- SSP log (when launching the “Hopex Site Service Provider” and when SSP triggers actions): SSPLOG*.TXT files.
- Watchdog log (starting the service or contacting the SSP): SWDLOG*.TXT files.
- Supervision log: SSPSPRVS*.TXT files stored in the share « [\\V-CLUST-SQL\MegaSite\Supervision](#) » because we centralized the MegaSite.ini file.

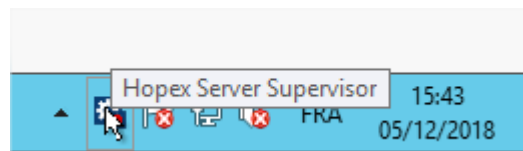
Restart the web application

Automatic restart with a Mega tool

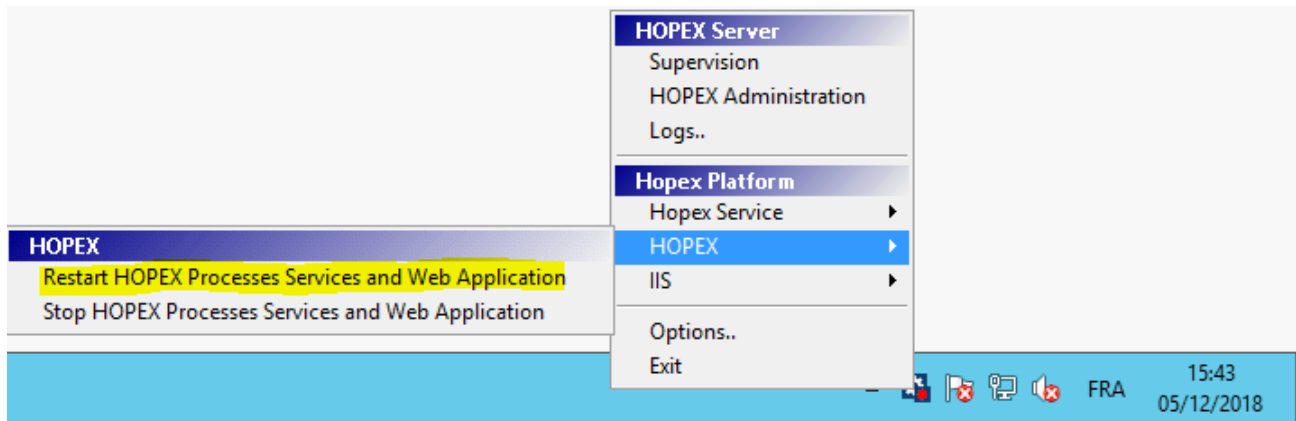
On each server hosting the Mega binaries, you can find a tool that allows you to restart the application, and open the daily logs (as well as other features). It can be found in the “Utilities\Hopex Server Supervisor” of the install folder of Mega:



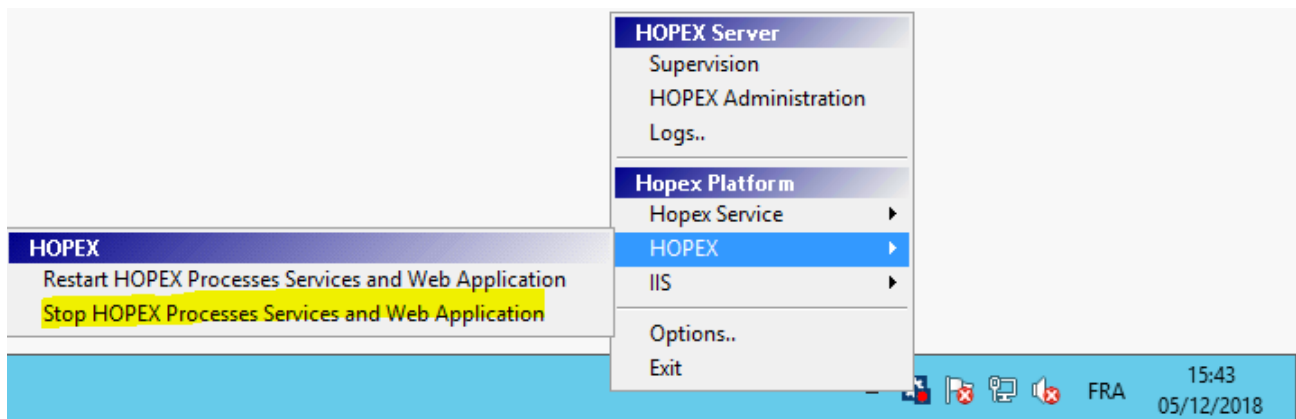
If you do a right-click the executable, and choose “Run as Administrator”, it will launch the process, and an icon will appear in the taskbar next to the clock:



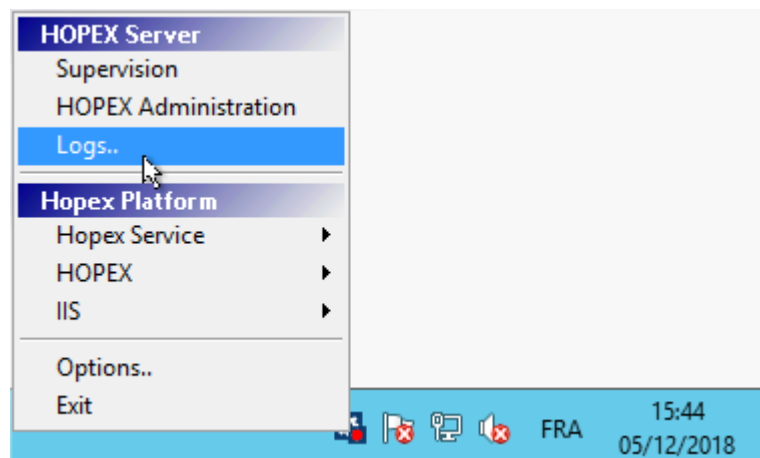
If you do a right-click the icon, you will get some options.
In the "System" section, you can restart the application (first entry), or stop the application (second entry):



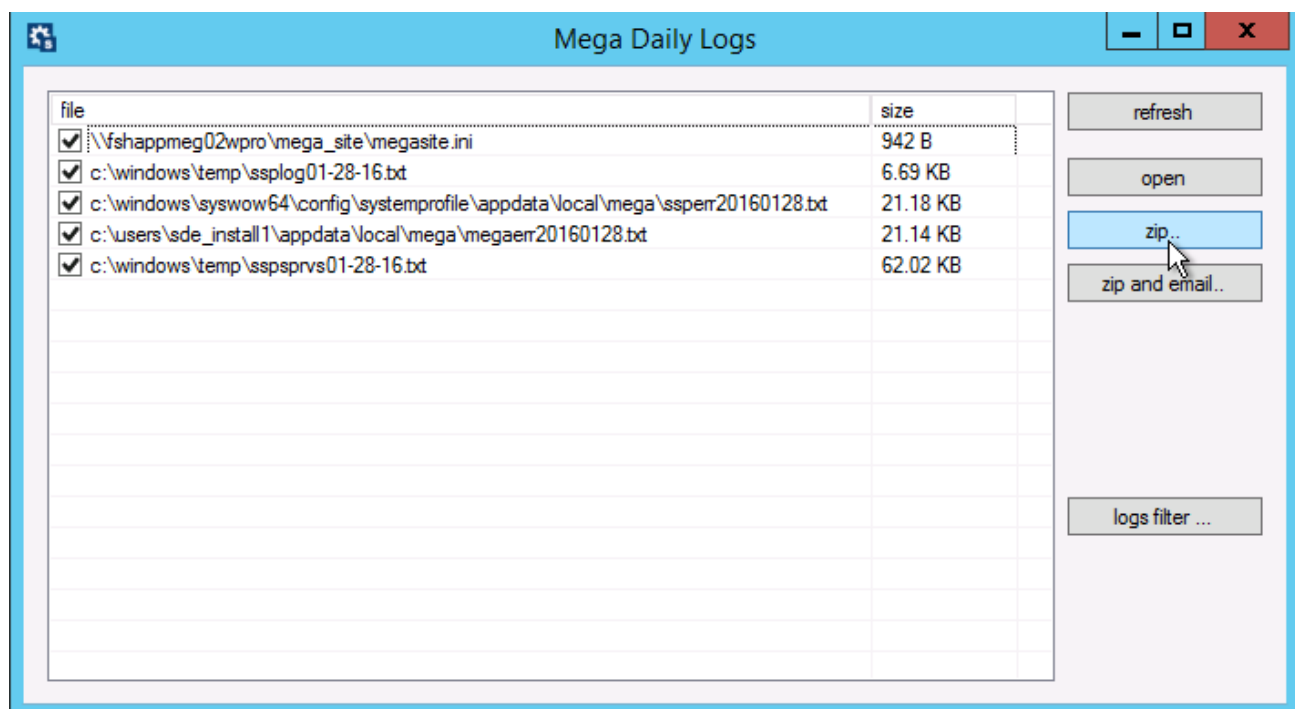
Second one will stop it, in case a maintenance is needed :



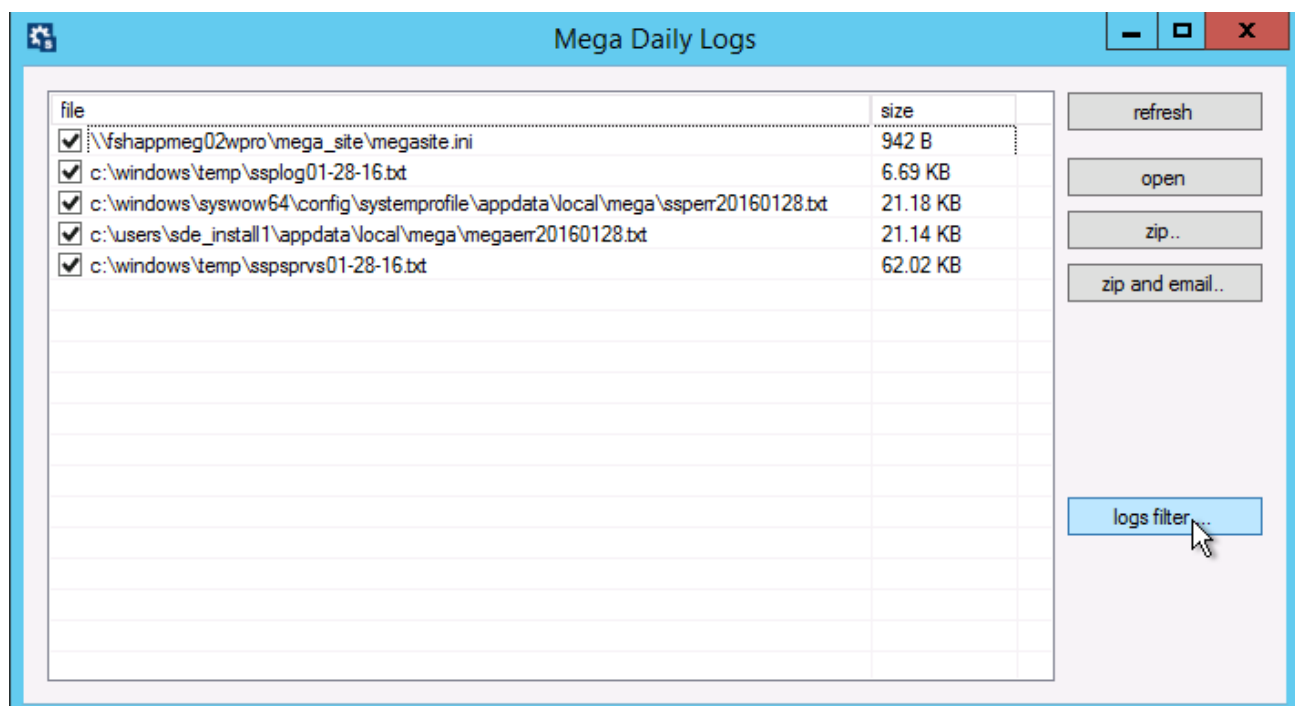
In the "Logs..." section, you can open a wizard to access the application logs:



You see by default the logs the day. You can click either “Open” to open them locally, or “Zip”, to create an archive file with all those logs.:



You can also choose « Log filter... », that will bring you another window that allows you to change to a number of days of logs, or to retrieve all logs located on the server:



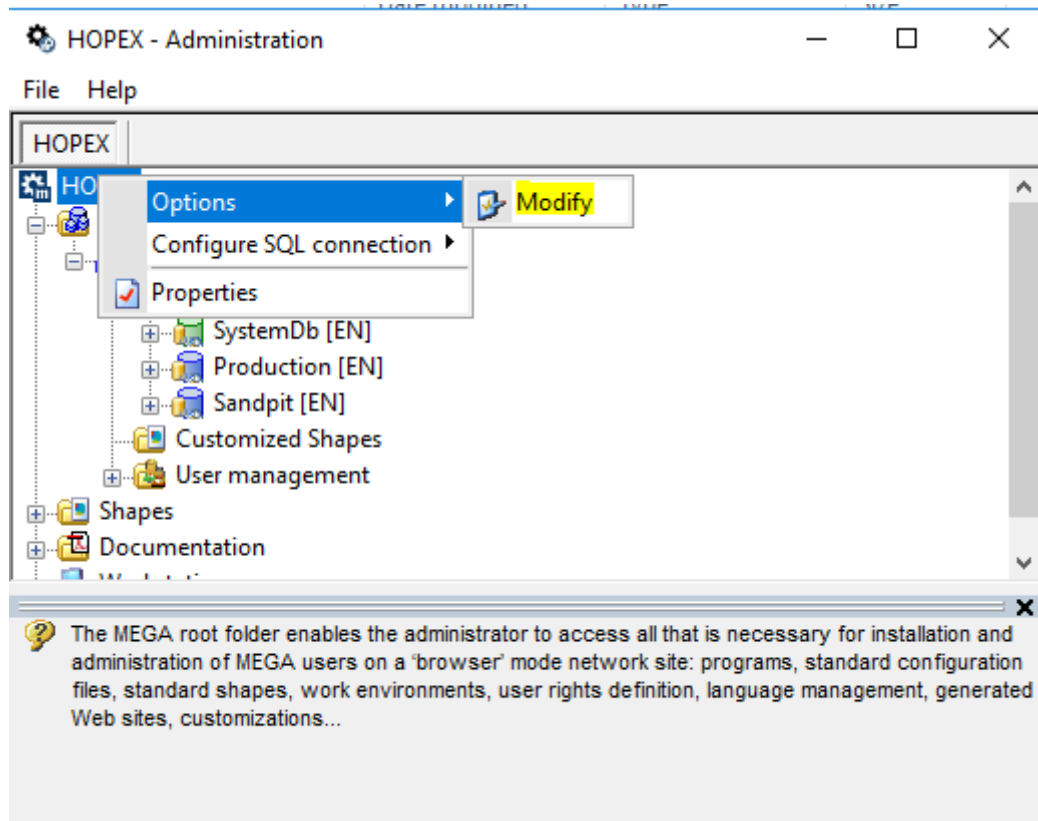
Tip : if you do a left-click the icon, it will show you the health and if it is started or not on the Application server.

It will tell you if the SSP is started (if you are on the server where the SSP is hosted), how many Mega processes are running, and if IIS is started or not.

Configure SMTP

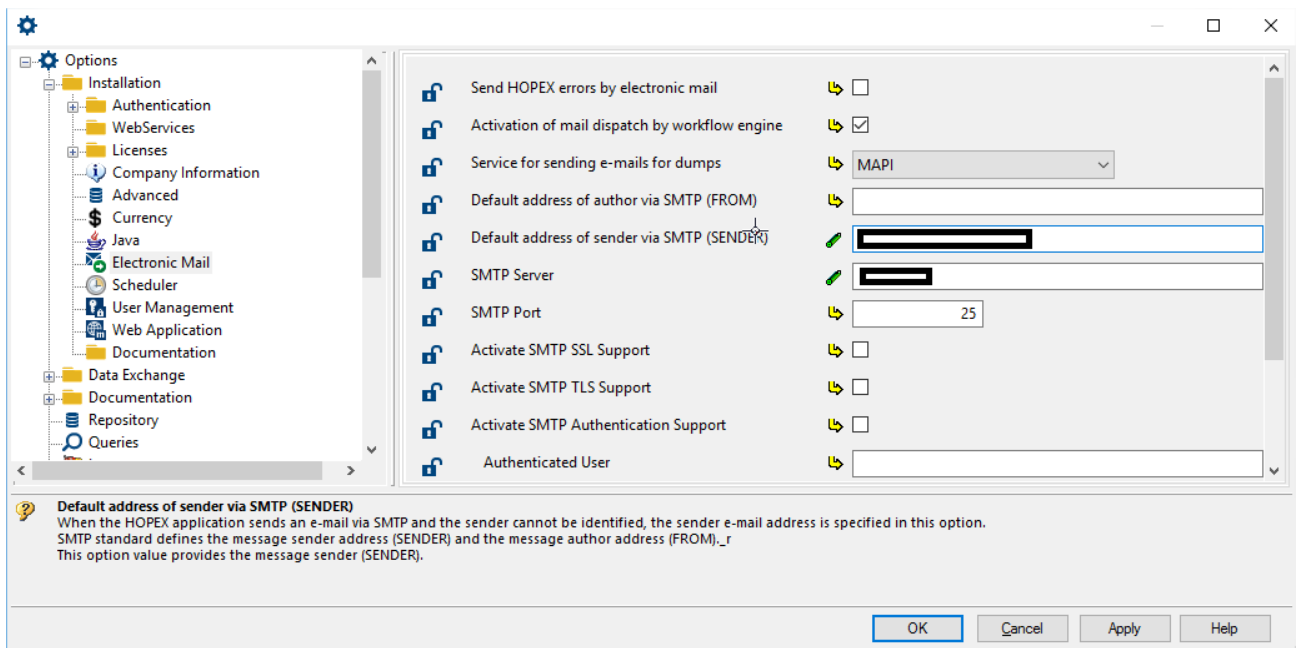
SMTP is configured as a global option.

In the Administration tool, open:

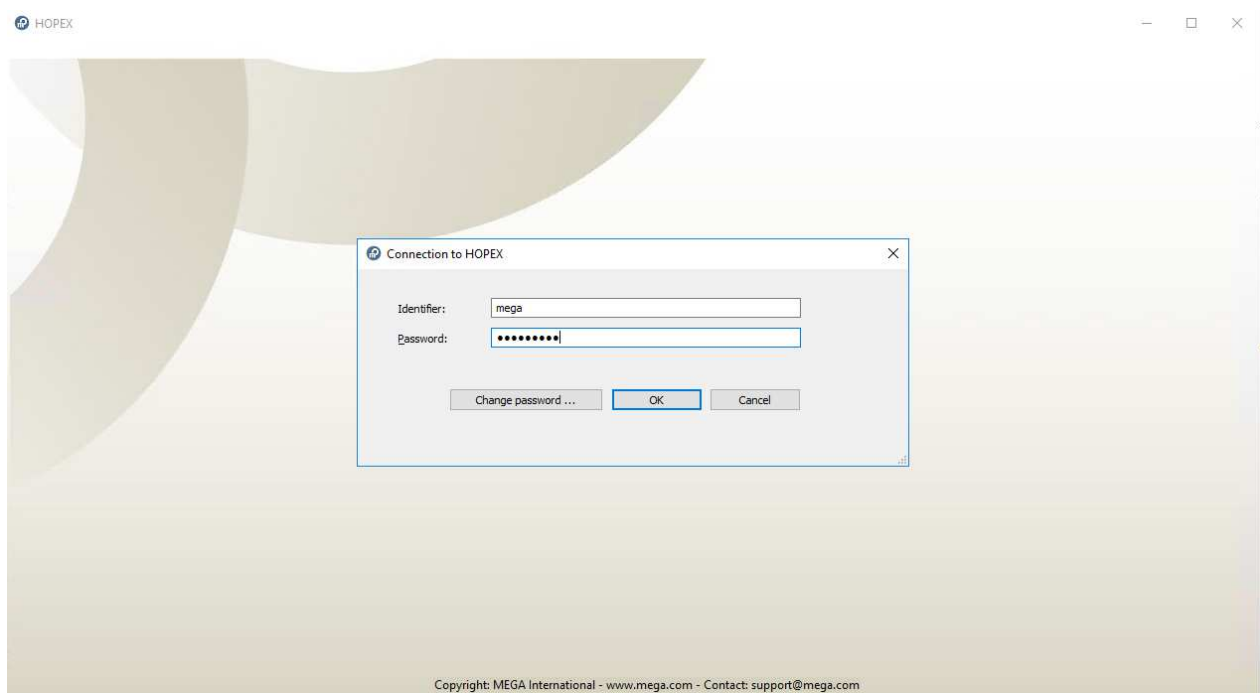


Go to « Installation-> Electronic Mail », and put the proper information for default sender, SMTP server, and communication port :

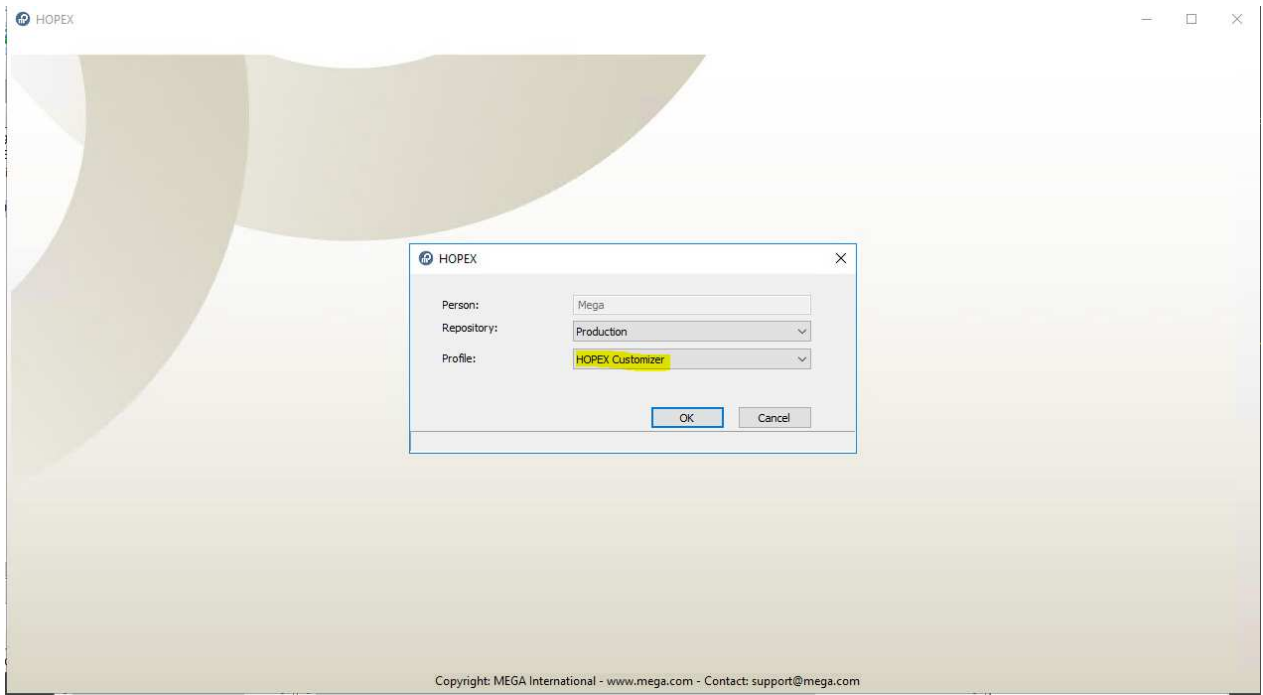
- Default sender: *hopex-noreply@yourorganization.com*
- SMTP server: *server_name*
- Port: 25 (by default)



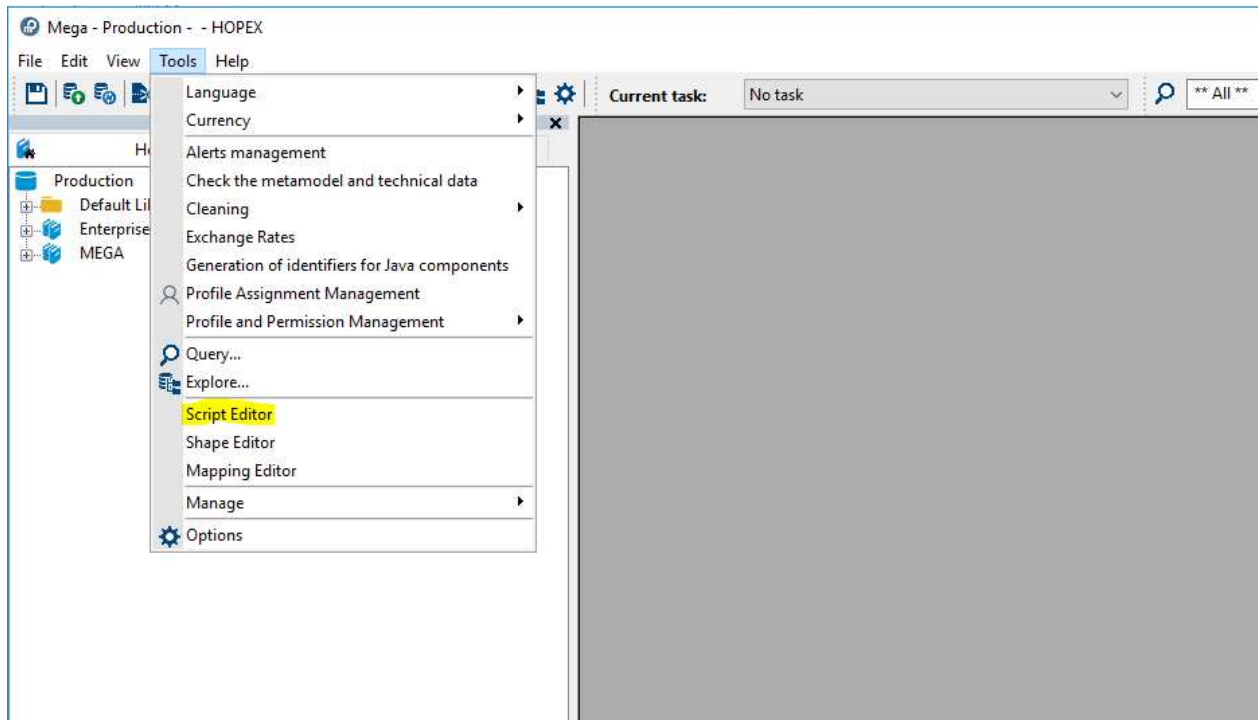
To test that the mails are working, we can use the Mega windows client ("Hopex.exe" at the root of the binaries) and we connect with the "mega" user:



Connect to any repository but with the "HOPEX Customizer" profile (you need specific tokens in your license to be able to make that test):



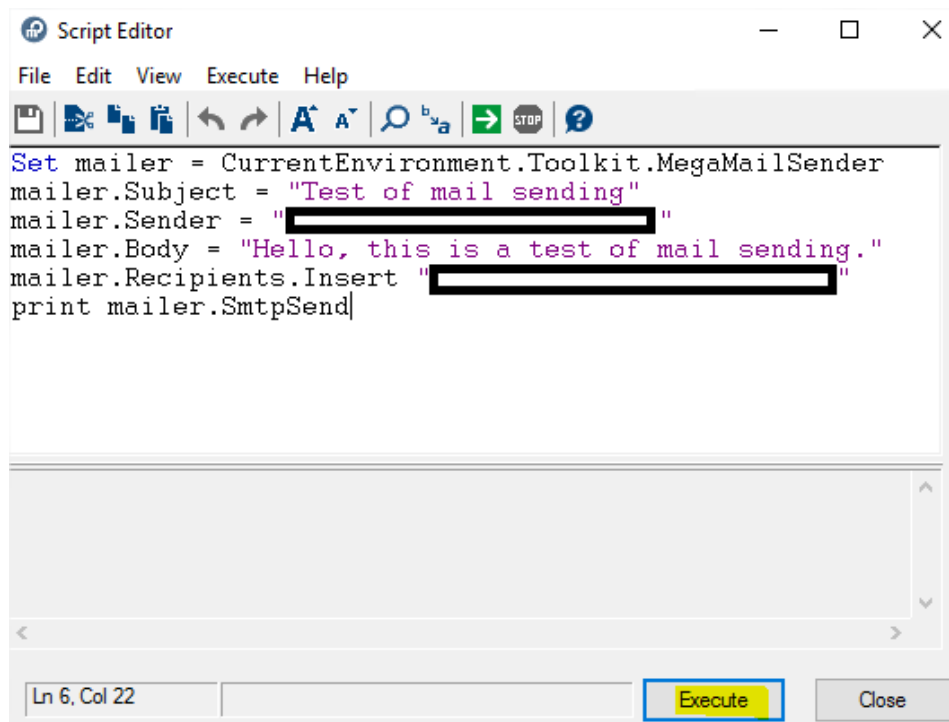
Then go to the script editor in the options:



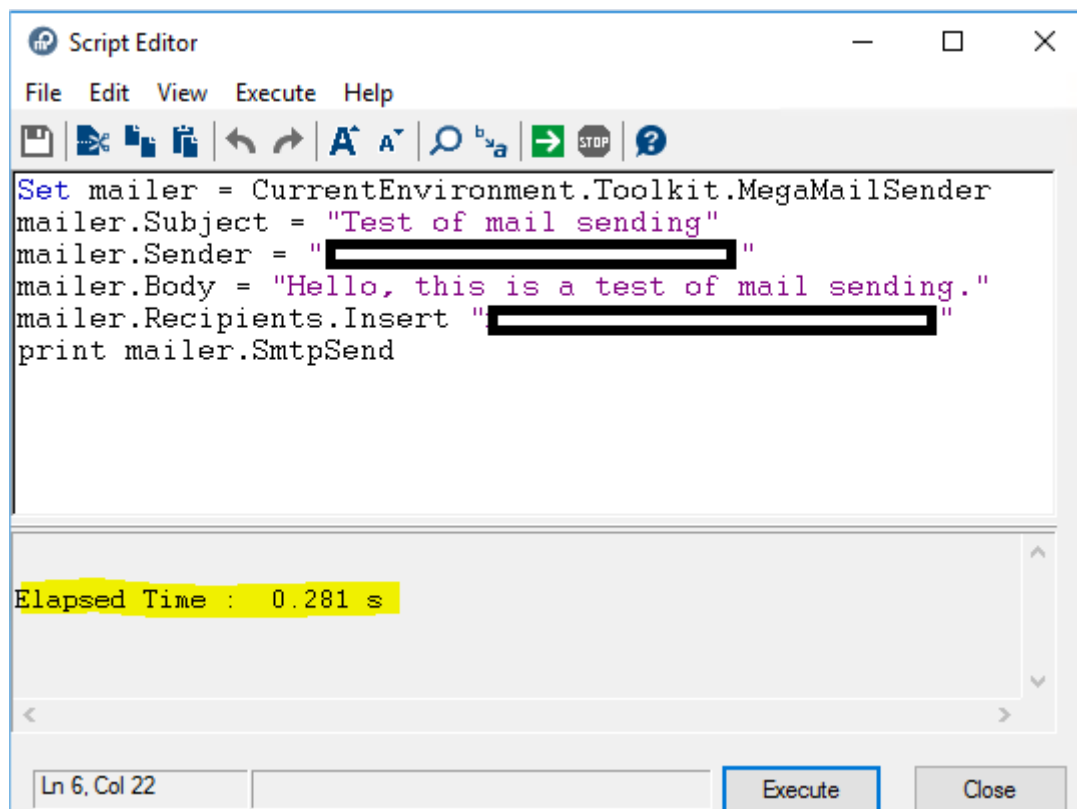
And put this kind of text:

```
Set mailer = CurrentEnvironment.Toolkit.MegaMailSender
mailer.Subject = "Test of mail sending"
mailer.Sender = "hopex-noreply@yourorganization.com"
mailer.Body = "Hello, this is a test of mail sending."
mailer.Recipients.Insert "your_target@yourorganization.com"
print mailer.SmtplibSend
```

Click "Execute":



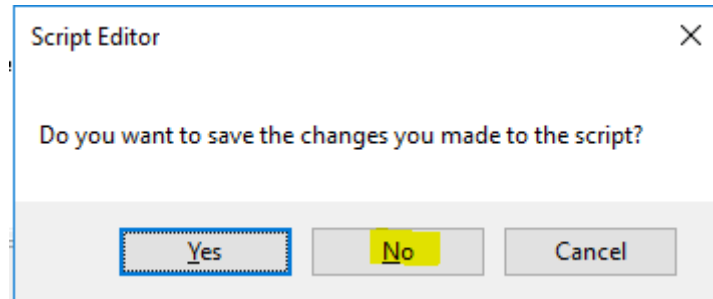
It takes a small amount of time to execute.



Check that the recipient received the test email to validate that the SMTP configuration is valid. In case of error, check the message in the popup

window, and if the message is too long, check the megaerr*.log file in the logs folder to have the full message.

You can click "Close", and then on "No", then close the "Hopex.exe" Windows client:



Web Front-End Architecture Overview

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 3 029 190 € - RC Paris B 385 185 806 000 51 / NAF 741 G

Summary

Check if a more recent version of this document is available in online documentation (MEGA Community).

This document applies to HOPEX V3.

It does not describe:

- How to perform installations (see installation documentation).
- How to manage installations (see administrator manuals).
- How products are licensed (see license installation documentation).
- How to use features (see user manuals).

The figures provided in this document are recommendations that may not apply to all contexts. In committing phases, a specific study with MEGA product management support is compulsory.

1. DEPLOYMENT TYPES	4
1.1. Standalone Deployment	5
1.2. Horizontal scaling Deployment	6
1.3. Vertical scaling Deployment	7
1.4. Vertical scaling Deployment (detailed view)	8
2. COMMON DEPLOYMENT REQUIREMENTS	9
2.1. Web Client.....	9
2.2. Application Server.....	10
2.3. File Server.....	11
2.4. Database Server	11
3. COMMUNICATION	12
3.1. Between Web Client and Web server (Web Application Server)	12
3.2. Between Environment SSP or MIK and Database server (SQL Server)	12
3.3. Between Environment SSP or MIK and mail server	12
3.4. Between Environment SSP or MIK and file server (file access, license access)	13
3.5. Between Environment SSP and LDAP Server	13
4. INSIDE.....	14
4.1. Administration tools.....	14
4.2. Anti-virus Configuration.....	14
4.3. Authentication	15
4.4. Cluster, scalability and load balancing	16
4.5. Data access.....	17
4.6. Data storage.....	17
4.7. Document management	18
4.8. Error and trace logfiles.....	18
4.9. Full search and indexing.....	21
4.10. Licensing.....	21
4.11. Mail system	21
4.12. Multi-language	22
4.13. Physical backup	22
4.14. Redo logs and activity tracking	23
4.15. Regular administration tasks.....	23
4.16. Reporting.....	24
4.17. Security.....	25
4.18. Services and running processes	25
4.19. Supervision.....	26
4.20. System caches	26
4.21. Technical documentation.....	27
5. FAQs.....	28

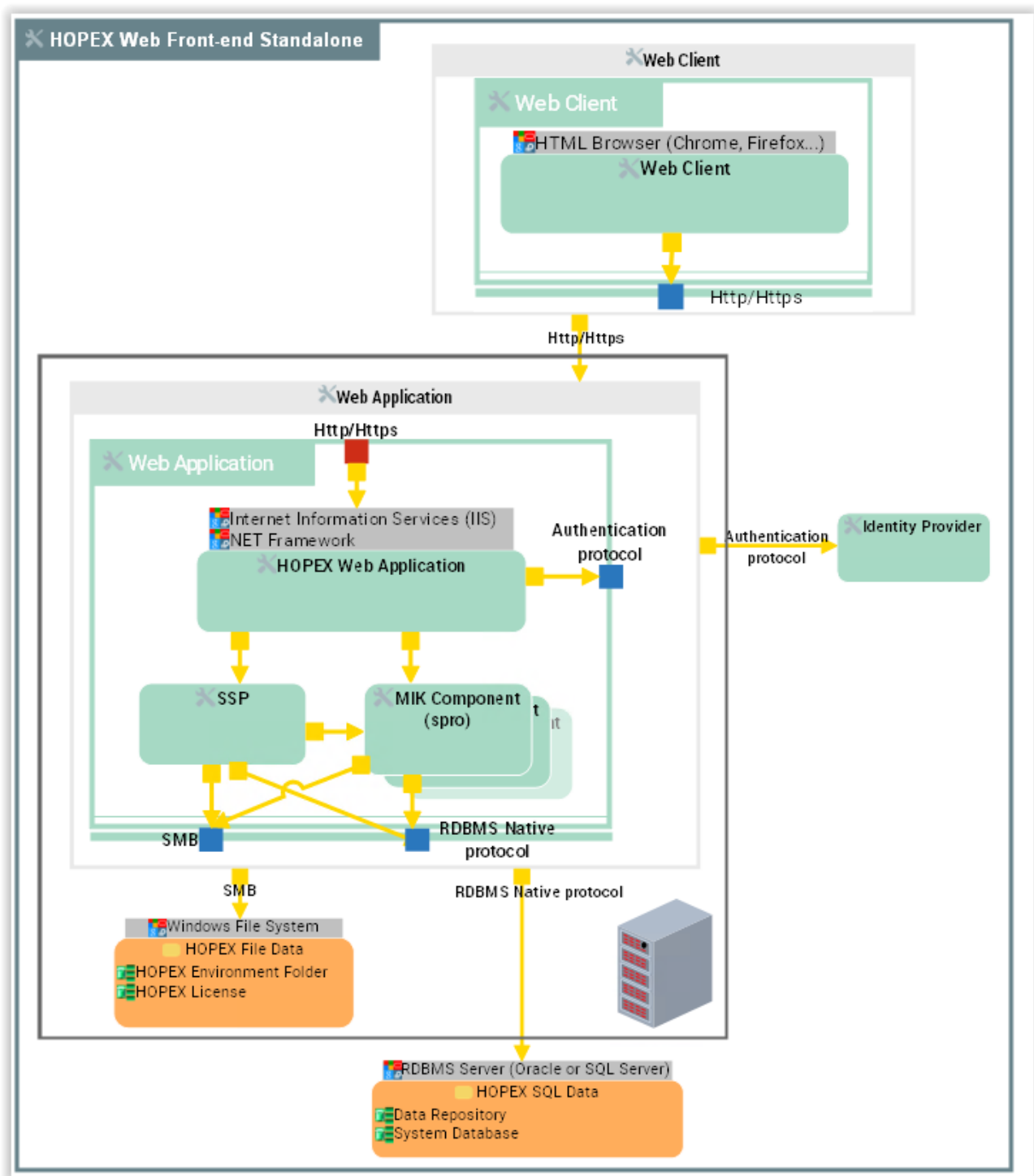
1. DEPLOYMENT TYPES

The HOPEX Web Front-End can be deployed in different typical ways:

Deployment type	Recommended for	Comment
Standalone	Small deployment	2 tiers architecture All in one server. Very easy to install.
Horizontal scaling	Large deployment	Multi-tiers architecture Also called 'Scale up'
Vertical scaling	Large deployment	Multi tiers architecture Also called 'Scale out'

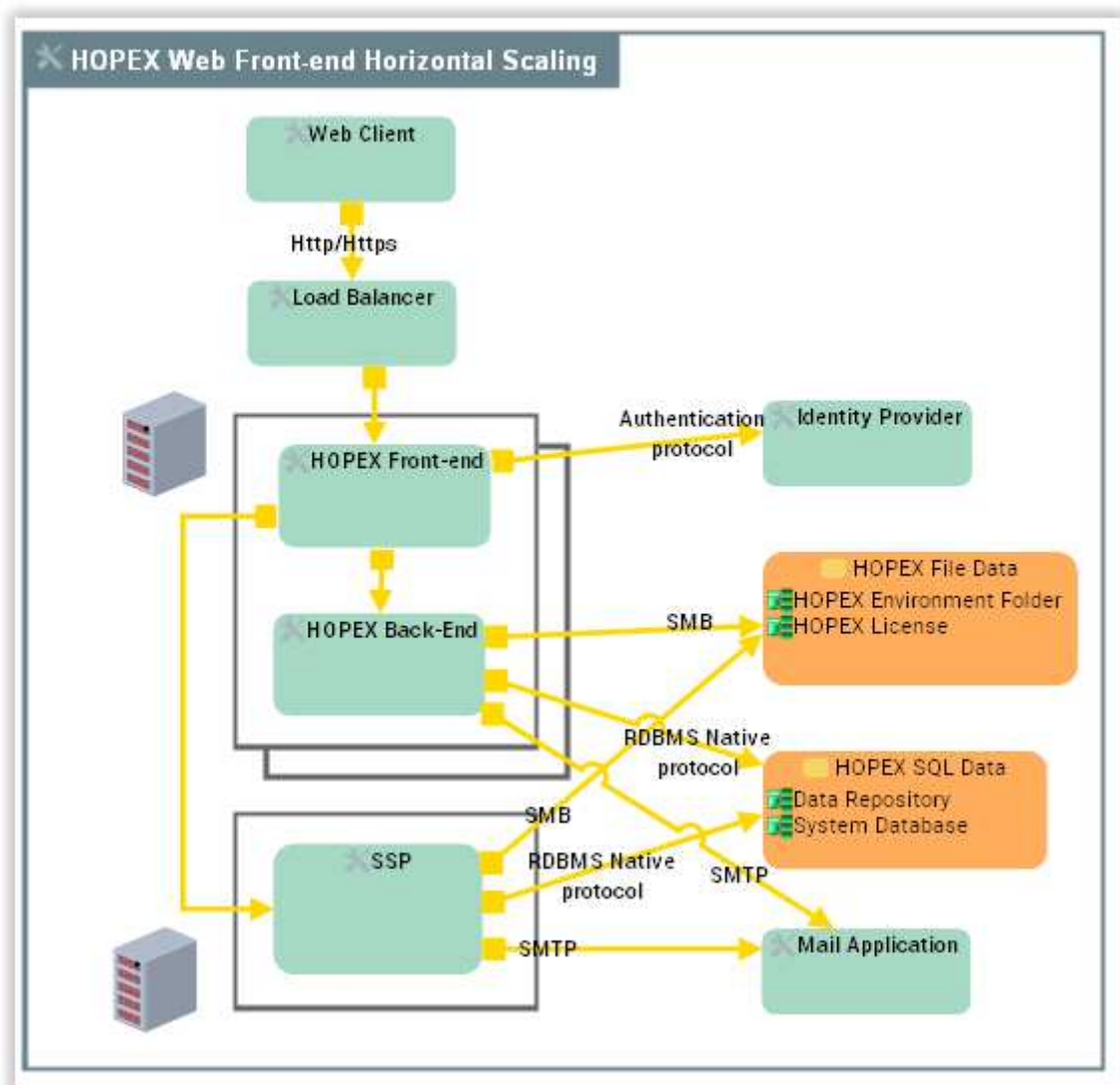
Other deployments – For specific requirements, other deployments are possible. For further information, contact your sales representative.

1.1. Standalone Deployment



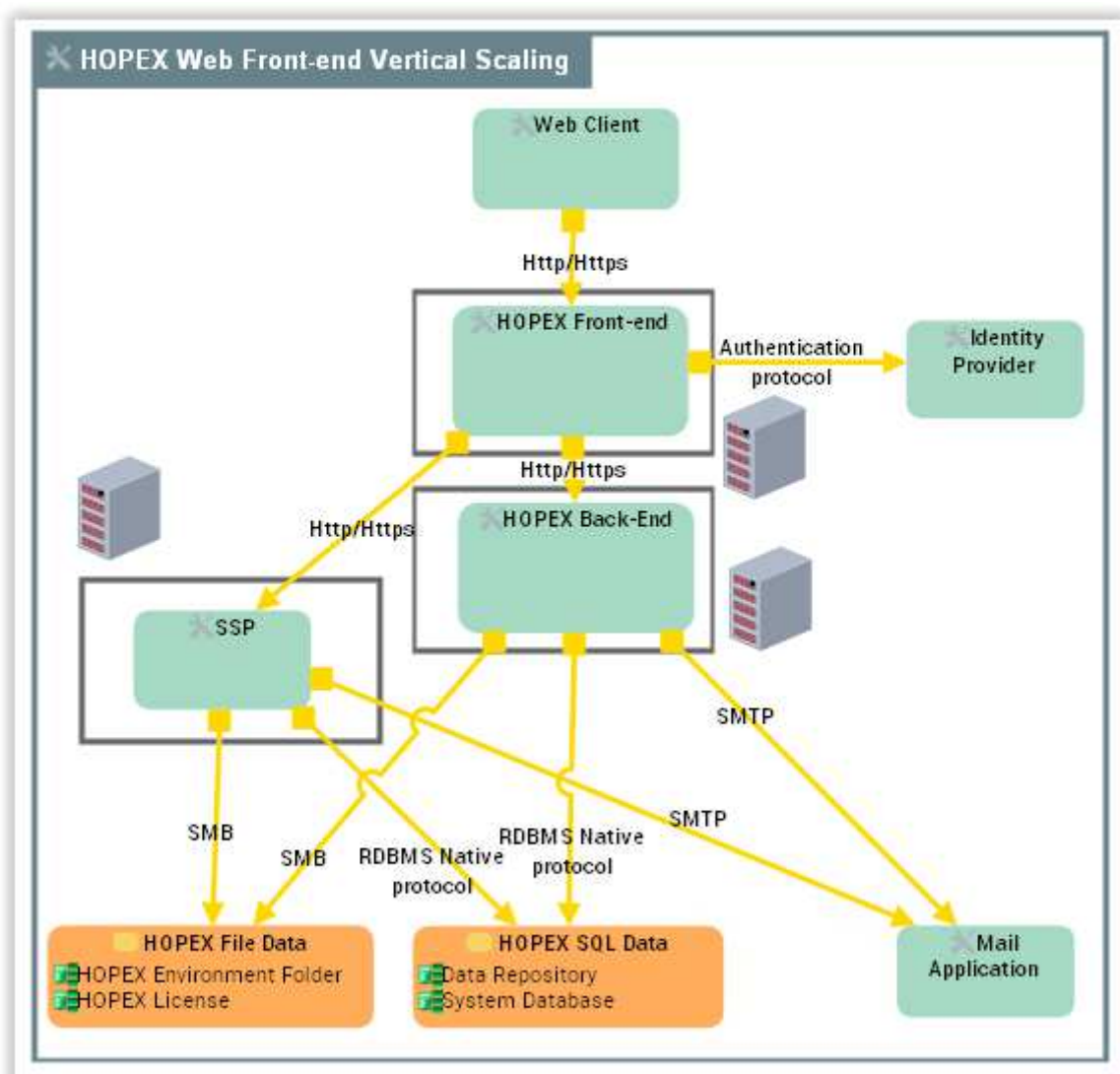
To facilitate readability, different elements have not been displayed (authentication server, mail server, SQL Server Native client required for SQL Server storage).

1.2. Horizontal scaling Deployment



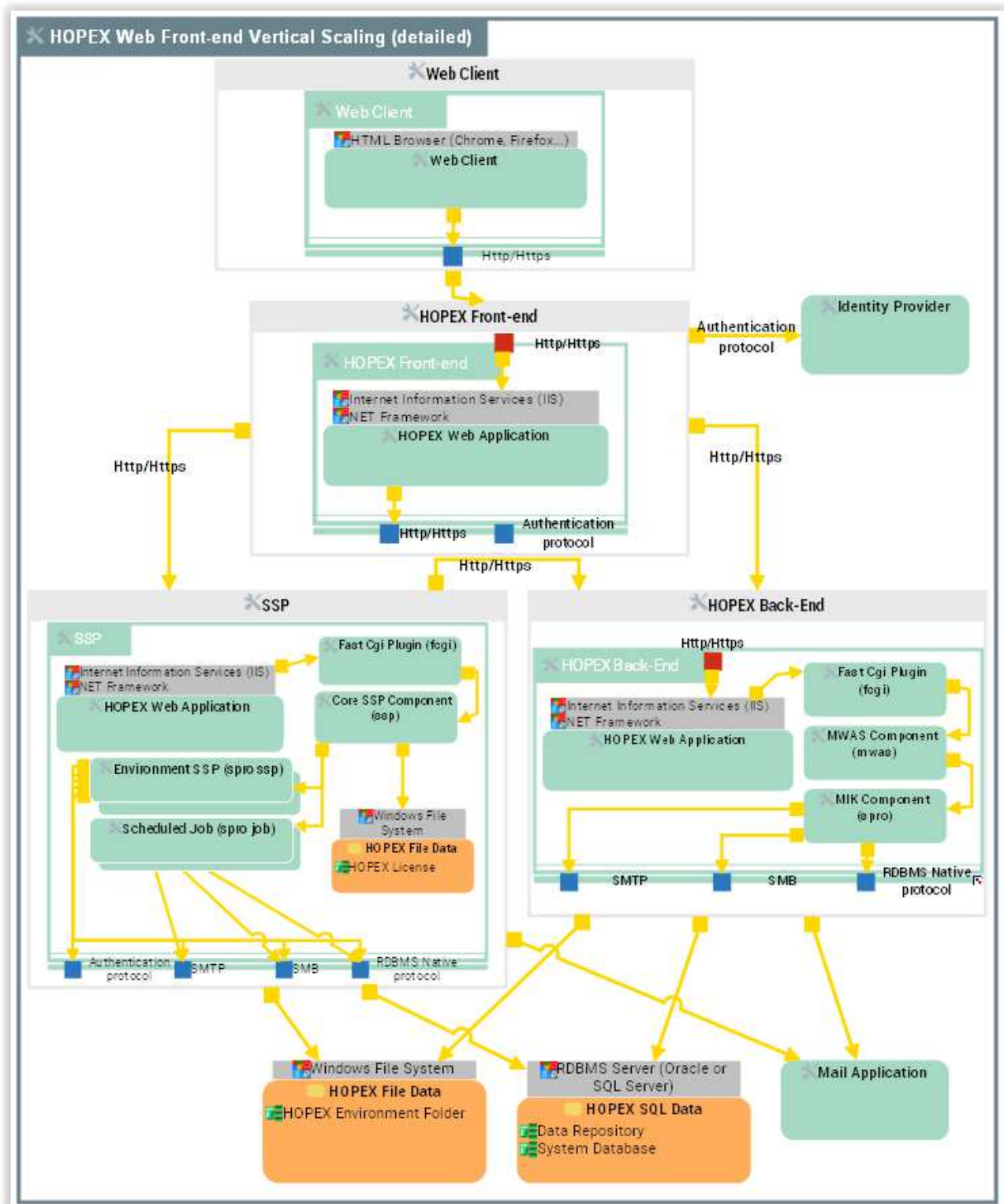
To facilitate readability, different elements have not been displayed (authentication server, document server, mail server, SQL Server Native client for SQL Server storage).

1.3. Vertical scaling Deployment



To facilitate readability, different elements have not been displayed (authentication server, mail server, SQL Server Native client for SQL Server storage).

1.4. Vertical scaling Deployment (detailed view)



To facilitate readability, SQL Server Native client (SQL Server storage) is not displayed.

2. COMMON DEPLOYMENT REQUIREMENTS

2.1. Web Client

HTML Browser 32/64 bit	Google Chrome MS Edge (1) Mozilla Firefox ESR (1)
Tablet	Chrome OS (2)
Configuration	Screen resolution 1280x800 16 M colours JavaScript enabled Cookies enabled HTML 5 enabled Download of files enabled Popup blocker disabled Web storage enabled
Additional Software	PDF reader RTF/DOC/DOCX reader XLS/XLSX reader

(1) Supported with minor restrictions. See FAQs section p. 28.

(2) For viewer users only.

2.2. Application Server

Operating system	Windows Server 2019 (3) Windows Server 2016 Windows Server 2012 R2 Windows Server 2012 Microsoft Azure deployment (1) For other systems a specific study is necessary Visual C++ Redistributable for Visual Studio 2015 (2)
Hardware	Processor Multi core RAM 6 GB minimum. 2 GB for the system 2 GB for data cache in memory Per environment SSP 400 MB Per equivalent concurrent modeller user 600 MB intensive use 300 MB low use Disk space 6 GB recommended for HOPEX Kernel 300 MB recommended for IIS applications 2 GB recommended for logs
Additional Software	SQL Server Native client 12.0 QFE (SQL Server 2012, SQL Server 2014, SQL Server 2016, SQL Server 2017) If data is stored in SQL Server
Web Server	MS Internet Information Services 8.0 MS Internet Information Services 8.5 MS Internet Information Services 10.0
Script layer	ASP .NET .NET Framework 4.6.2 or higher Microsoft URL Rewrite Module 2.0 NET Core 2.1 Runtime & Hosting Bundle for Windows (v2.1.7)

(1) With specific parameters. See Appendix.

(2) Required for each Window machine running HOPEX kernel (workstation or server).

(3) Supported from HOPEX V3 CP4.

Note that these are general indications. You should contact MEGA to discuss a more suitable sizing, especially if more than 5 users are expected.

2.3. File Server

Operating system	Windows Server 2019 (2) Windows Server 2016 Windows Server 2012 R2 Windows Server 2012 Microsoft Azure deployment (1) For other file systems a specific study is necessary
Hardware	Processor Multi core RAM 1 GB minimum. 1 GB for the system Disk space 5 GB recommended per HOPEX Environment (environment folder) 10 MB for HOPEX License

(1) With specific parameters.

(2) Supported from HOPEX V3 CP4.

2.4. Database Server

Server System	see RDBMS requirements
RDBMS	SQL Server 2019 recommended SQL Server 2017 recommended SQL Server 2016 SQL Server 2014
Disk space	Data: 2 GB minimum per system database 3 GB minimum per data repository Refer to the separate article 'RDBMS Repository Installation guide HOPEX V3'.
Hardware	RAM: a specific study is required. Refer to the separate article 'RDBMS Repository Installation guide HOPEX V3'. CPU: see hardware requirements of the RDBMS.

3. COMMUNICATION

3.1. Between Web Client and Web server (Web Application Server)

Protocol	HTTP by default
Port	80 by default
Network bandwidth	Per equivalent modeller user 60 Kbit/s average bandwidth 512 Kbit/s peak bandwidth
Network latency	100 Ms maximum (A)

(A) For a ping of 5 KB (It is recommended to use the hrping utility). Refer to the separate article 'RDBMS Repository Installation guide HOPEX V3'.

Note that a proxy configuration can be required: see section 'Security' of this document.

3.2. Between Environment SSP or MIK and Database server (SQL Server)

Protocol	SQL Server: SQL Server Protocol
Port	SQL Server: Example UDP/TCP 1433 (B)
Network bandwidth	1 Gbit/s minimum full duplex (C)
Network latency	1 Ms maximum (A)

(A) For a ping of 5 KB (It is recommended to use the hrping utility). Refer to the separate article 'RDBMS Repository Installation guide HOPEX V3'.

(B) Default port check the appropriate port with the database administrator.

(C) For 30 concurrent users.

3.3. Between Environment SSP or MIK and mail server

Protocol	SMTP
Port	25 by default, configurable
Network bandwidth	1 Gbit/s minimum full duplex (C)
Network latency	1 Ms maximum (A)

3.4. Between Environment SSP or MIK and file server (file access, license access)

Protocol	SMB/CIFS
Port	UDP/TCP 138 UDP/TCP 137 UDP/TCP 139 UDP/TCP 445
Network bandwidth	1 Gbit/s full duplex

3.5. Between Environment SSP and LDAP Server

Protocol	LDAP
Port	TCP 389 by default (B)
Direction	Bidirectional

(B) Default port check the appropriate port with the LDAP server administrator.

4. INSIDE

4.1. Administration tools

Several administration tools can be used:

Administration tool	Component	Tasks
Web Administration Desktop	Desktop of HOPEX Web Front-End	Functional administration (user, permissions, workspaces, LDAP configuration, import/export...)
Web Supervision console	.NET application	Monitoring of running processes and events...
Web Monitoring console	.NET application	Monitoring of connected user, management of logs, installation checks...
Web Licensing console	.NET application	Monitoring of license use, assignment of users to the license...
Windows Administration Console	Win32 (Administration.exe)	Data storage management (environment, repositories, stored procedures) Functional administration (user, permissions, workspaces, LDAP configuration, import/export...)
Monitoring Console	.net web page (XX.aspx)	Supervision of HOPEX (IIS) application
IIS manager	Win64 (InetMgr.exe)	Management of IIS server
Must license manager	Win32 (Licensing.exe)	Management of Must license
Windows Front-End	Win32 (HOPEX.exe)	Fix unexpected configuration issue
HOPEX Server Supervisor	Win32 (Hopex Server Supervisor.exe)	System supervision of the server

Reference:

See online documentation, HOPEX Administration ... Administrator Guide

4.2. Anti-virus Configuration

To maintain good performances, it is recommended to exclude certain file extension from antivirus scanning (on access scanning)

Machine	Location/File	Comment
Each machine running HOPEX	%programdata%\MEGA and subfolder Ex: C:\ProgramData\MEGA File extension: *.MGC	Folders of the Compiled data cache and RDBMS local cache
Each machine running HOPEX	Location: check with the HOPEX administrator Ex: C:\Program Files (x86)\MEGA\MEGA HOPEX V3 File extension: *.*	Folders of HOPEX core programs
Each machine running HOPEX IIS application	Location: see HOPEX administrator Ex: C:\inetpub\wwwroot\HOPEX File extension: *.*	Folders of HOPEX IIS application

4.3. Authentication

Basic authentication (variant MEGA) is available immediately after installation.

Other authentication models need to be configured in HOPEX or integrated with HOPEX after installation.

With HOPEX V3, an authentication framework called 'UAS (Unified Authentication Service)' is used. It enables to:

- Secure authentication requests.
- Use standard identity providers.
- Develop custom identity provider.

Several authentication models can be implemented:

Authentication models	Description	Comment
OpenID authentication	Authentication process is managed within HOPEX Platform. Users are declared in an external directory. Standard providers are available for the following identity providers: Microsoft, Salesforce, Google	This model is recommended for standard deployments where OpenID is used. No integration is required for the 3 identity providers addressed, only configuration and testing. For other identity providers, a specific integration is required.
SAML2 authentication	Authentication process is managed within HOPEX Platform. Users are declared in an external directory. A standard provider is available, implemented using AD FS (Active Directory Federation Services).	This model is recommended for standard deployments where SAML2 is used. No integration is required, only configuration and testing.
Windows Authentication	Authentication process is managed within HOPEX Platform. Users are declared in an external directory. A standard provider is available, implemented using WIF (Windows Identity Foundation)	This model is recommended for standard deployments where Windows Authentication is used. No integration is required, only configuration and testing.
Basic authentication	Authentication process is managed within HOPEX Platform. Users are declared explicitly in the HOPEX Environment and possibly mapped individually with an external directory. 3 variants: MEGA, LDAP, Windows	This model is recommended for basic deployments. No integration is required, only configuration.
Fully custom authentication	Authentication process is external to the HOPEX platform (UAS is skipped). All types of IT corporate directory can be addressed (customized identity provider)	This model is not recommended. It can be used for advanced deployments with specific requirement. It requires a specific integration.

Password values storage, encryption and update vary with the configuration chosen.

Authentication models	Storage	Encryption
OpenID authentication	According to identity provider specifications	According to identity provider specifications
SAML2 authentication	Active Directory	According to directory specifications
Windows Authentication	Active Directory	According to directory specifications
Basic authentication (MEGA)	System repository	Encrypted, hashed
Basic authentication (Windows)	Active Directory	According to directory specifications
Basic authentication (LDAP)	LDAP directory	
Fully custom authentication	According to implementation	According to implementation

Reference:

- Online documentation, HOPEX Administration ... Authentication in HOPEX
- Article 'HOPEX Unified Authentication Service Installation guide'

4.4. Cluster, scalability and load balancing

This document contains metrics for a small deployment. Sizing is a complex matter that is closely linked to infrastructure and can be impacted by security policy. As a consequence, medium or large deployments need specific studies:

- Initial sizing according to load hypothesis.
- Load tests in the final infrastructure to check that sizing is appropriate.

For large deployments, scalability and load balancing is required.

Service	Principle
Scalability	Install on a cluster/farm server. A configuration file is used to share configuration between nodes.
Load balancing	Install on a cluster/farm server. Use a load balancer mechanism to balance load between nodes. A specific integration is required.
High availability	Install on a cluster/farm server. Use server SSP nodes (multiple SSP servers) Use a load balancer mechanism to balance load between nodes. A specific study is recommended.

To implement load balancing, various solutions are available on the market. In all cases the solution must be qualified and supported by customers and/or third parties.

4.5. Data access

Access to data is mainly controlled using profiles (repository access, data permissions, and GUI permissions).

Other features are available:

- Writing access management: control of updates on existing objects.
- Reading access management: control of visibility regarding existing objects.
- Data access rules: computed control of visibility regarding existing objects.

Reference:

See online documentation.

- HOPEX Administration ... Managing Data Reading Access
- HOPEX Administration ... Managing Data Writing Access

4.6. Data storage

Each HOPEX Environment consists of one system repository and one/several data repositories.

By default, data is stored in a database server (SQL Server).

Storage	Mapping	Comment
SQL Server	<p>A data repository is an SQL Server database.</p> <p>A system repository is an SQL Server database. (1)</p>	<p>Create one SQL server user for the environment with specific privileges.</p> <p>Only SQL server authentication is supported.</p> <p>Install and schedule stored procedures by data repository or system repository.</p> <p>No dedicated instance is required.</p> <p>SQL Server native client.</p> <p>Default port can be used.</p>

Reference:

- Article 'RDBMS Repository Installation guide HOPEX V3'
- See online documentation, Products.

4.7. Document management

A document management system is available through a solution or a pack.

Object	Location	Storage
Business Document	Data repository	Database server
System Business Document	System database	Database server

If document management is enabled, web users can add, update and consult documents.

Reference:

- See online documentation, Common Features ... Using Business Documents

4.8. Error and trace logfiles

No log is generated on the client side. All errors are displayed using popup windows or via the HTML browser. An option enables to control the display of errors to end users (GUI). For advanced diagnostic, a verbose mode can be enabled to generate more detailed logfiles.

Different files can be created on server side:

File	Comment	Default location (example)
sspsprvsYYYYMMDD.txt	Supervision log (3)	%programdata%\MEGA\HOPEX V3\ClusterRoot\Supervision Ex: C:\ProgramData\MEGA\HOPEX V3\ClusterRoot\Supervision
SSPLOGYYYYMMDD.txt	Core SSP log (1)(3)	%programdata%\MEGA\Logs Ex: C:\ProgramData\MEGA\Logs
ssperrYYYYMMDD.txt	Environment SSP log (1)(3)	%programdata%\MEGA\Logs Ex: C:\ProgramData\MEGA\Logs
MWASLOGYYYYMMDD.txt	MWAS component log (1)(5)	%programdata%\MEGA\Logs Ex: C:\ProgramData\MEGA\Logs
megaerrYYYYMMDD.txt	MIK component log (1)	%programdata%\MEGA\Logs Ex: C:\ProgramData\MEGA\Logs
uas-YYYY-MM-DD.log	UAS component log (4)	%programdata%\MEGA\Logs\UAS Ex: C:\ProgramData\MEGA\Logs\UAS
SWDLOGYYYYMMDD.txt	Service Watchdog log (1)(2)	%programdata%\MEGA\Logs Ex: C:\ProgramData\MEGA\Logs
dtpxYYYYMMDD.txt	DTPX component log (4)	<iis root>\HOPEX\App_Data\DTPX Ex C:\inetpub\wwwroot\HOPEX\App_Data\DTPX
redis_server_log.txt	Redis component log (2)	%programdata%\MEGA\Logs Ex: C:\ProgramData\MEGA\Logs
HopexHealthDigestReport YYYY-MM-DD_XX-XX- XX.html	Installation health report	%programdata%\MEGA\HOPEX V3\ClusterRoot\HopexHealth Ex: C:\ProgramData\MEGA\HOPEX V3\ClusterRoot\HopexHealth

File	Comment	Default location (example)
RepositoryHealth-YYYY-MM-DD-MyEnvironment_MyRepository	repository statistics	%programdata%\MEGA\HOPEX V3\ClusterRoot\HopexHealth Ex: C:\ProgramData\MEGA\HOPEX V3\ClusterRoot\HopexHealth

Where

- DD is a number indicating the day in the month.
- MM is a number indicating the month in the year.
- YYYY is a number indicating the year.

(1) Location can be configured

(2) Generated for each server where HOPEX components are installed

(3) Generated for the server running SSP

(4) Generated for the server running HOPEX Front-end

(5) Generated for the server running HOPEX Back-end

4.9. Full search and indexing

Solutions of HOPEX platform can use full search. A parameter at data repository and/or system repository level enables to activate indexing.

There are 2 levels of indexing:

- Full indexing: the data repository/system repository is scanned, and index files are created in a subfolder of the data repository/system repository.
- Incremental indexing: the log (internal) of the data repository/system repository is scanned and index files are updated in a subfolder of the data repository/system repository.

Full search and indexing are available with RDBMS storage only.

Reference:

See online documentation

- HOPEX Administration ... Enabling and Customizing Repository Indexing
- Common Features ... Presentation of search tools

4.10. Licensing

Products and solutions of HOPEX platform are protected by Must licenses. Must licenses can be shared between multiple users.

Must licensing is not server-based (there is no Windows process for a license server). At runtime with HOPEX Web Front-end, a set of files are generated dynamically by service account.

However, a domain user (Active directory) is required for:

- Each service account running the HOPEX (IIS) application.
- Each user running the Administration Console (system administrator, functional administrator).
- Each user running the Windows Front-end (developer, functional administrator, user associated to a scheduled task).

To obtain a license, contact your sales representative. A UNC will be requested and a .must license file (locked on this UNC) will be sent with installation instructions.

Reference:

Article 'Must License Installation Guide HOPEX V3'.

4.11. Mail system

A mail server needs to be configured so that mail notifications can be used within workflows.

SMTP parameters (server, port, proxy...) can be configured for the installation using the Administration console.

4.12. Multi-language

Web Front-End enables to work with multiple languages.

Nature	List	Installation	Comment
GUI Language	Core languages (1)	Core languages are installed by default. With additional languages, it can be requested to install a language pack on the Application Server.	Controls the display of the user interface (menus, pages...) Different end users can have different GUI languages.
Data language	More than 30 languages available	Core languages are installed by default. Additional languages are installed at environment level	Enables data entry in several languages for objects. An end user can switch between several data languages within his session

(1) Core languages are English, French, Italian, Spanish and German.

4.13. Physical backup

In case you face a real disaster recovery scenario, presence of a valid and restorable backup is very important.

Element	Recommendations
Frequency	Every 24 hours (1) (2)
Retention	In the last 30 days keep daily backup In the last 12 months keep a monthly backup
Other files to backup	By default, backup folder of each HOPEX Environment

(1) For HOPEX Environment used by an active project

(2) In particular before a major update concerning data. E.g.: system repository customization, data reprocessing, CP/RP upgrade of MEGA data

Cold/warm backup are supported.

4.14. Redo logs and activity tracking

Service	Activation	Comment
Embedded log (repository log)	Enabled by default	Enables to generate a log of updates (redo log), activity tracking. Also used by specific features (full search, alert management...) This log can be partially/completely initialized and disabled using Windows Administration Console.
External log (backup logfile)	Enabled by default	Enables to generate additional command files logging the updates of a user (backup log) that can be useful to recover quickly data after an incident. This log can be disabled using Windows Administration Console.

Reference:

See online documentation

- HOPEX Administration ... Managing Repositories.
- HOPEX Administration ... Managing logfiles.
- HOPEX Administration ... Optimizing Repository Access Performance.

4.15. Regular administration tasks

A few tasks need to be run and can often be automated:

Task	Server involved	Comment
Conservation of repository performance	Database server	Stored procedure to be installed and scheduled for each data repository and system repository. Can be automated. SQL server only.
Deletion of historical data	Database server	Stored procedure to be installed and scheduled for each data repository and system repository. Can be automated.
Deletion of private workspace temporary data	Database server	Stored procedure to be installed and scheduled for each data repository and system repository. Can be automated.
Environment compilation	Application server	To build system cache. System updates are impossible during compilation. Need to stop HOPEX Services and HOPEX related processes
Full indexing	Server running SSP	Manual.
Incremental indexing	Server running SSP	Automated using HOPEX Scheduler.
Information about fragmentation and statistics	Database server	Generates a technical report regarding physical indexing (statistics gathering)
Maintenance Plan	Database server	Need to stop SSP when running maintenance plan (SQL server)
Maintenance plan (SQL Server storage)	Database server	Required with several tasks. Can be automated. Refer to the article 'RDBMS Repository Installation Guide HOPEX V1R2 EN
Physical backup of data (SQL Server)	Database server	Required. Daily backup recommended. Can be automated.
Restart HOPEX Web site	Web server	For HOPEX program upgrade (CP upgrade) Can be required in case of problem
Restart IIS server	Web application server	Can be required in case of problem For IIS programs upgrade

Task	Server involved	Comment
Restart server	Application server	Can be required in case of problem
Restart SSP service (1)	SSP server	For HOPEX program upgrade (CP upgrade) For certain changes (license, list of environments, and list of repositories...) Can also be required in case of problem

(1) Windows service 'Mega Site Service Provider'.

4.16. Reporting

There are three categories of reports:

Category	Native format	conversion format	Comment
Report	HTML	RTF, XLS, XLSX, PDF	Window or web Front-End Generated from a Report template According to the Report template considered, certain conversion formats may not be available.
Report (MS Word)	RTF	-	Window or web Front-End Generated from a list or from a Report template (MS Word).
Instant report	HTML	-	Web Front-end only Generated from a list or from a Report DataSet. A report DataSet is a table of data generated from a Report DataSet Definition

To open a report from the web client, a reader corresponding to the format should be installed.

Example: MS Excel to read .XLS documents, Adobe reader to read .PDF documents, Open Office/MS Word to read .RTF documents.

(1) Web Front-End does not enable to design Report templates (MS Word): templates must be developed on Windows Front-End with MS Word 32-bit and delivered using a specific procedure.

Execution mode	File size	Comment
.DOCX mode (by default)	Limited	Some restrictions compared to .DOC mode (no longer supported)
.RTF mode	Important (RTF format is verbose)	RTF macros are not supported MS Word fields (such as table of content) are not refreshed Minor formatting issues

Reference:

See online documentation

- HOPEX Power Studio ... Report DataSet Definition
- HOPEX Power Studio ... Report Studio
- HOPEX Power Studio ... Customizing Reports (MS Word)

4.17. Security

All ports used in the HOPEX platform are either configurable or set elsewhere. No specific port is required or hard-coded. To configure firewall ports, see the 'Communications' section earlier in this document.

MEGA strongly recommends configuring HTTPS to improve the security of flows between the Web Client and the Web Server. This requires a specific configuration of IIS and HOPEX.

If a local enterprise proxy is used, it should be configured by adding an excluding rule on the proxy. The rule refers to the IP address of the HOPEX web server involved.

File permissions should enable access to:

- Error and trace logfiles (see section 'Error and trace logfiles' in this document).
- License folder.
- Environment folder.

Reference:

Article 'Web Front-End - Securing the platform'.

4.18. Services and running processes

Several Windows services are created by the installation:

Service	Executable	Startup type	User (1)	Server
HOPEX Site Service Provider	mgwssp.exe	Automatic	Local system	SSP server
HOPEX Service Watchdog	mgwswd.exe	Automatic	Local system	Each server used to deploy Web Front-end
HopexRedisBackEnd	redis-server.exe	Automatic	Local system	Each server used to deploy Web Front-end

At runtime, several processes can be created.

Process	User	Comment	Number
mgwssp	Local system (1)	Core SSP	One/several per installation. Runs on SSP server. Started by windows service
mgwmapp			
mgwspro	Local system (1)	Environment SSP (MIK)	One per HOPEX Environment. Runs on SSP server
mgwspro	Local system	Scheduled job	According to scheduler configuration
mgwmwas	Service account	MWAS (HOPEX)	One per web application server
mgwmapp			
mgwspro	Service account	Web session (MIK)	One per end user (single session), one per group of user (multi session)
mgwswd.exe	Service account	Service Watchdog	One per server application server. Started by windows service
mgwmapp	Current user	Administration Console	One per running instance of Administration Console. Started manually.
HOPEX Server Supervisor	Current user	HOPEX Server Supervisor utility	One per running instance of the utility. Started manually.

(1) Can be configured

4.19. Supervision

The HOPEX platform enables system monitoring.

Supervision logfiles are updated by the server running the SSP when various events occur.

This information can be consulted via

- Web Supervision console
- HOPEX Server Supervisor (Windows utility)

A WMI probe can also enable to supervise HOPEX from standard tools supporting WMI (a specific integration is required).

Reference:

See online documentation, HOPEX Administration ... Managing Events

4.20. System caches

Several caches are created on the server.

Cache type	Location (disk)	Average size (disk)	Comment
Cache of systemdb and data repository (HOPEX-RDBMS cache, memory)	-	-	Process redis-server.exe in memory One process per HOPEX server Process can reach maximum 2 GB Ram
Compiled data cache	Default location: %programdata%\MEGA\<version code>\Cache\Compiled data Ex: C:\ProgramData\MEGA\HOPEX V3\Cache\Compiled data	10-30 MB (1)	One folder per HOPEX environment. Cache of systemdb configuration. Cannot be disabled. Updated by environment compilation.
Cache of MetaPicture	Default location: %programdata%\MEGA\<version code>\Cache\Compiled data Ex: C:\ProgramData\MEGA\HOPEX V3\Cache\Compiled data	1-5 MB	Cache of images. Cannot be disabled. Updated dynamically at runtime.
Cache of resources	Default location: <iis root>\wwwroot\HOPEX\App_Data\MWAS\res Ex: C:\inetpub\wwwroot\HOPEX\App_Data\MWAS\res	1-10 MB (1)	Cache of resources for MWAS. Cannot be disabled. Updated dynamically at runtime.

(1) For one HOPEX environment

4.21. Technical documentation

Category	Audience	Format	Language code
Installation and deployment guides	System administrator, functional administrator	PDF	EN
Online documentation	End user, functional administrator	web site	EN, FR, IT*, DE*
Technical articles	Developer, functional administrator	PDF	EN
Javadoc	Developer	HTML pages	EN

Installation and deployment guides and user manuals are installed in the subfolder \Documentation of HOPEX programs folder

Example: C:\Program Files (x86)\MEGA\HOPEX V3\Documentation

Language codes:

EN : English

IT: Italian

SP: Spain

FR: French

DE: German

* can be available a few months after the initial release

5. FAQs

5.1.1. What about HTML browsers other than Edge, Firefox and Chrome?

MEGA has decided to focus on Chrome, IE, Firefox. This does not mean that solutions do not run on HTML browsers. It means only that these HTML browsers are not supported.

5.1.2. What is web storage for HTML browsers?

This is a capability of HTML browsers to store data (localStorage mode)

This capability is supported by recent browsers (IE11, Edge, Firefox, Chrome)

5.1.3. What is supported for Azure?

Here are the options qualified by MEGA so far:

- Premium storage (SSD disk)
- VM DSv2
- SQL on local VM (private cloud, IaaS, SQL Server Web Edition)

5.1.4. What is Mozilla Firefox ESR?

As Firefox versions change very rapidly, MEGA has decided to focus on ESR versions.

Extended Support Release (ESR) based on an official release of Firefox for desktop is used by organizations that need extended support for mass deployments.

See also <http://www.mozilla.org/en-US/firefox/organizations/faq/>

5.1.5. What is the list of minor restrictions for Edge / IE / Firefox?

There are non-conformities to standards such as HTML browser zoom.

The list is documented in the document 'Known issues version HOPEX V3 CPX'.

5.1.6. Are IE 9.0/10/11 still supported?

Internet Explorer 9.0/10 are not supported (end of mainstream support).

See <https://support.microsoft.com/en-en/lifecycle>

Internet Explorer 11 is still supported.

MEGA recommends to use a more recent HTML Browser the IE 11.

5.1.7. Is Windows Server 2008 R2 still supported?

With HOPEX V3, Windows Server 2008 R2 SP2 is supported as application server and file server but not recommended as support end date has passed

See <https://support.microsoft.com/en-en/lifecycle>

5.1.8. Are SQL Server 2008/2008 R2/SQL Server 2012 still supported?

With HOPEX V3, SQL Server 2008/2008 R2/SQL Server 2012 are supported as database server but not recommended as support end date has passed.

See <https://support.microsoft.com/en-en/lifecycle>.

5.1.9. Are there requirements or recommendations regarding security policies (GPOs)?

It is assumed that standard policies (installed by default with the system) are available. In particular, the policy 'Impersonate a client after authentication' can be necessary for the HOPEX service account and IIS related users, based on your deployment. If certain policies are not available, a specific study is required.

5.1.10. How to configure HTTPS?

This can be done through the installation program. See the article 'Web Front-End Installation Guide MEGA HOPEX V3'. Note that a certificate should be configured before installing HOPEX: see your IIS administrator.

5.1.11. It is possible to use a Must licence that is not located on the SSP Application Server?

This is possible. An additional configuration is required.

5.1.12. Can the HOPEX web Front-End run on a web server other than IIS?

HOPEX V3 is designed for IIS only.

5.1.13. Can HOPEX solutions and products run on a mobile platform?

Most HOPEX products and solutions are designed for a web client running on a desktop or laptop computer with screen resolution 1280x800. They have not been designed for pads or smart phones.

Viewer users can use tablets running Android. Viewer users can consult data usually though a simplified desktop.

Note that technologies used by the HOPEX platform enabled to develop web application that can run on mobile platforms. Tuning of web desktops is required.

5.1.14. What are the web technologies used by HOPEX Platform?

For HOPEX Web Front-end, the HOPEX platform uses HTML5 and various JavaScript related technologies mainly: Ajax., Extjs., Dojo.

On the server side, nothing is required except the .NET Framework. All necessary execution layers are installed by default. HOPEX V3 uses an embedded JRE (version 8).

A detailed list of third-party components is available on MEGA Community
<https://community.mega.com/t5/Open-Source-in-HOPEX-Software/bg-p/legal>

5.1.15. What about other database servers?

MEGA has decided to focus on widespread and recent versions of SQL Server.

5.1.16. Are there supervision tools?

The HOPEX installation generates supervision logfiles. The standard utility HOPEX Server Supervisor provides a supervision interface. It is also possible to setup a WMI probe to communicate with supervision tools (Nagios...). For this, a specific integration is required.

How to Migrate to HOPEX V3

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 3 029 190 € - RC Paris B 385 185 806 000 51 / NAF 741 G

Summary

Check if a more recent version of this document is available in online documentation (MEGA Community).

This document describes the procedures necessary for upgrading HOPEX Data to version HOPEX V3. Migration is allowed with specific CPs for source and target versions.

Source version	Target version (direct migration path)
HOPEX V2R1 Update 3 CP5	HOPEX V3 CP2 last hotfix Then CP upgrade to last CP (1)

For previous versions or releases (HOPEX V2R1 Update 2, HOPEX V2R1 Update 1, HOPEX V2, HOPEX V1R2... MEGA 2009) it is necessary to perform an intermediate upgrade to HOPEX V2R1 Update 3 **CP5**.

Source version	Target version (indirect migration path)
HOPEX V2R1 Update 2	HOPEX V2R1 Update 2 > HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)
HOPEX V2R1 Update 1	HOPEX V2R1 Update 1 > HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)
HOPEX V2	HOPEX V2 CP07.0 > HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)
HOPEX V1R3	HOPEX V1R3 CP17.0 > HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)
HOPEX V1R2	HOPEX V1R2 CPx > HOPEX V1R3 CP 17.0 -> HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)
HOPEX V1R1	HOPEX V1R1 CPx > HOPEX V1R3 CP 17.0 -> HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)
MEGA 2009	MEGA 2009 SP5 CP11.0 > HOPEX V1R3 CP 17.0 -> HOPEX V2R1 Update 3 CP5 > HOPEX V3 CP2 (1)

(1) from HOPEX V3 CP2 you can apply CP upgrades to attain the latest CP.

It does not describe:

- System requirements and possible architectures (see architecture overview documentation).
- Change is product list (see your sales representative)
- How to perform installations (see installation documentation).
- How to install update (see how to install update documentation).
- How to manage installations (see administrator manuals).
- How products are licensed (see license installation documentation).
- How to use features (see user manuals).

1. MAIN STEPS TO MIGRATE DATA TO HOPEX V3	4
2. PREPARE UPGRADE OF DATA	5
2.1. Check metamodel, locks, workspaces and workflows	5
2.2. Verify that GBMS storage is no longer used.....	5
2.3. Verify that Windows Front-End is no longer used for runtime	6
2.4. Verify format of report templates (MS Word)	6
2.5. Verify technology of web desktops	7
2.6. Identify Solution packs used.....	7
2.7. Decide 'Definition of path of MetaAssociation'	7
2.8. Check license with your sales representative	8
2.9. Review use of the profile 'Enterprise Architect'	8
2.10. Review authentication mode	8
2.11. Decide to keep web settings	9
3. UPGRADE DATA FROM HOPEX V2R1 TO HOPEX V3	10
3.1. Check data upgrade pre-requisites	10
3.2. Upgrade environment with SQL Server storage.....	11
3.3. Update stored procedures with SQL Server storage	14
4. COMPLETE UPGRADE OF DATA.....	15
4.1. Set a value for 'Definition of path of MetaAssociation'	15
4.2. Re-import solutions packs	15
4.3. Rebuild full search indexes	16
4.4. Review command line parameters.....	17
4.5. Convert custom report templates (MS Word) to format RTF	17
4.6. Restore web settings	18
5. CHECK UPGRADED DATA	19
5.1. First control of migration.....	19
5.2. Check data modelling consistency.....	19
5.3. Other checking indications	20
6. APPENDIX.....	21
6.1. Conversion details	21
6.2. Utilities details	27
7. FAQs	30

1. MAIN STEPS TO MIGRATE DATA TO HOPEX V3

The data migration consists of several main steps:

1. Prepare data for migration

This step requires HOPEX V2R1 (source version allowed for direct migration).

This step performs a validation that the existing data is compliant with the future metamodel and that customizations associated MetaAssociation behaviors are saved.

It also checks for pre-requisites, identifies solution packs used, and helps to determine the value of important parameters (options).

Most of this work requires human intelligence and knowledge of data that has been modelled within the tool. As a consequence, it cannot be automated and should be scheduled in advance of a production migration.

2. Upgrade data

This step requires HOPEX V3.

The process upgrades the metamodel and converts data to the format required by the HOPEX platform. This is carried out via conversion tools that need to be run manually from the Administration Console (Administration.exe). The procedures vary according to the source version your existing data.

In addition, this allows important parameters to be reviewed (options).

3. Check upgraded data and customizations

This step requires HOPEX V3.

This step involves validation from the end user perspective since they are most familiar with the prior state of the data.

- Modelled data has been correctly migrated.
- Customizations have been correctly migrated.

This step also requires human intelligence and knowledge of modelled data. Therefore, it cannot be automated.

2. PREPARE UPGRADE OF DATA

2.1. Check metamodel, locks, workspaces and workflows

In the source version, for each environment:

Check	Detail
Check that the metamodel is stable	In Windows Administration Console (Administration.exe), compile the environment. If the environment compilation generates a log entry in the HOPEX error log, you should fix such errors before migrating your data
Check that no private workspace (ex-transactions) persists	In Windows Administration Console (Administration.exe), check workspaces. If a private workspace persists, dispatch or delete it.
Check that no lock persists	In Windows Administration Console (Administration.exe), check locks. With RDBMS storage (SQL Server), you need to dispatch or delete related workspace.

2.2. Verify that GBMS storage is no longer used

GBMS storage is no longer supported in HOPEX V3.

It is required to change storage to SQL storage before upgrading data.

A license Repository Storage (SQL Server) is required.

For more details, see only documentation, reorganization feature

HOPEX Administration: Administrator Guide : Managing Repositories : Managing Repositories : Reorganizing an RDBMS Repository

Pre-requisites:

- Check that you have a license with Repository Storage (SQL Server).
- Stop user activity and backup data
- Verify that a physical backup of data is available
- Dispatch or delete all pending workspaces
- Verify you have enough working space
- Create SQL Server database with appropriate permissions

Recommendations:

- Decide if you can delete repository log
- Use a server machine close to SQL Instances
- Leave the process running quietly.
- Check process (test) on a copy of production
- Loop until the processing runs without unexpected error.

2.3. Verify that Windows Front-End is no longer used for runtime

Windows Front-End (HOPEX.exe) is no longer allowed for execution (runtime) in HOPEX V3 and higher versions. It is required to move to Web Front-end. This can lead to review de deployment architecture of HOPEX.

From HOPEX V3, Windows Front-End is allowed only for:

1) **For Administration tasks** with HOPEX.exe

- Standard administration tasks
- Advanced administration features (HOPEX Power Supervisor, code SUP)
- Management of reporting DataMart (HOPEX Reporting Datamart, code HDT)

2) **For Customization** with HOPEX.exe

This requires specific profiles such as HOPEX Customizer and a licence with HOPEX Power Studio (code MTS2).

2.4. Verify format of report templates (MS Word)

With HOPEX, two formats were available for Report (MS Word) objects and RTF stylesheets.

- RTF/DOCX format: **This generation mode is the only one allowed with HOPEX V3.**
- DOC format: This generation mode is no longer supported with HOPEX V3 as Windows Front-end is no longer supported for runtime.

Format	Impacts
RTF/DOCX	Reports have the .rtf or .docx file extension according to an option (2). RTF stylesheets has the .rtf file extension. MS word is not used at runtime. No links exist in generated reports. Fields (such as table of content) are not refreshed automatically. RTF macros are not supported. Application of styles is not enforced after generation, which may cause differences in display.
DOC	Reports have the .doc file extension. RTF stylesheets has the .doc file extension. MS word is used at runtime. Links exist in generated reports (can be removed with detach). This format is now deprecated.

For each environment, check file megaenv.ini

[Office]

DocumentFormat=XX

Document format	Interpretation
[Office] DocumentFormat=10	RTF/DOCX format
[Office] DocumentFormat=20	DOC format
Not specified (default)	RTF/DOCX format

If an environment is configured with DOC format, it is required to convert report templates (MS Word). See later in this document Convert report templates (MS Word).

2.5. Verify technology of web desktops

This section is regarding GRC Solutions:

- HOPEX Enterprise Risk Management (code ERMW)
- HOPEX Internal Control (code ICM)
- HOPEX LDC (code LDC)
- HOPEX Internal Audit (code MIAW)

Each profile used with Web Front-End is associated to one or several web desktops.

Each web desktop is based on one of these technologies.

- Classic Desktops (also called V1 Desktops, legacy technology).
- Universal Desktops (latest technology).

With GRC Solutions (except for solution HOPEX Internal Audit), Universal Desktop technology is now used. This can have an impact according to project customizations. Each project needs to identify its situation.

Situation	Impacts on customization	Impact on look and feel	Comment
Full standard (standard profiles with standard desktops)	No impact	Change	Profiles are now associated to Universal Desktops instead of the Classic GRC Desktops
Full custom (custom profiles with custom desktops)	No impact	No change	By default, profiles are still associated to Classic GRC Desktops. Of course, project can decide to use Universal Desktop instead. A review of customization is required.
Mixed (standard profiles customized with standard desktops, custom profiles with standard desktops...)	Possible impact.	Possible change	A case by case review is required. A case-by-case review is required. It is highly recommended to create custom profiles. Project must decide if they are associated to Universal Desktops (recommended) but keeping working with the Classic GRC Desktop is still possible

A standard element (profile, desktop..) is a configuration provided out of the box by HOPEX. It can be customized. Customization of standard profiles (to change CRUD/Object UI permissions) is not recommended (bad practice). Keep in mind that Solution HOPEX Internal Audit is not impacted with HOPEX V3. Change of web desktop technology is planned for a coming version.

2.6. Identify Solution packs used

Solutions packs are add-ins installing data or templates. There are imported in data repositories using the Administration Console, but they can update the system database.

Example: DoDAF, NAF ...

For each HOPEX environment, identify the list of solution packs imported:

- In the system database
- In a data repository

2.7. Decide 'Definition of path of MetaAssociation'

This step requires a decision for each HOPEX environment.

In the HOPEX options, group 'Repository', an option 'Definition of path of MetaAssociation' is available at installation and environment level. This option enables to control the way MetaAssociation behaviors are interpreted according to the value chosen:

- Compatibility up to MEGA 2009: MetaAssociation behaviors are interpreted using the logic of MEGA 2009.
- From MEGA HOPEX 1.0: MetaAssociation behaviors are interpreted using a new logic.

Value	Recommended
Standard Mode	Recommended for new projects. Default value.
Compatibility Mode	Recommended for compatibility with behaviors and customizations performed in version MEGA 2009 and lower (data and system database customization). When switching to 'Standard mode', a review that may require time and expertise is necessary.

Note that 'Standard Mode' is the default value from HOPEX V1R2/V1R3. You can change the value and compile the environment without impact on data except namespace. However, the change will affect the behaviors (namespace, navigation, extraction, protection, export, comparison...).

2.8. Check license with your sales representative

The list of products/solutions changes with each version:

- Certain products/solutions are removed (not available).
- Certain products/solutions are deprecated (available and supported as is).
- Certain products/solutions are repackaged (features still available through another product/ solution)
- SQL Server storage is now required.

This document will not describe the product lists or the changes between versions. Please contact your sales representative to see if a new license needs to be programmed.

2.9. Review use of the profile 'Enterprise Architect'

For many versions, the profile 'Enterprise Architect' (ex-EA Standard) has been used for multiple purpose

- Use legacy products (MEGA Process BPMN edition, MEGA Architecture...)
- Perform customizations
- Use HOPEX Solutions

This profile is designed for use of legacy products.

- It is not designed for customization: use the profile 'HOPEX Customizer'.
- It is not designed for use HOPEX Solutions: use dedicated profiles and desktops.

From HOPEX V3, the profile 'Enterprise Architect' has a command line that filters HOPEX Studio and HOPEX Solution: /RW'NAF;ARC;HBPA;DOD2;FEA;UML;ITD;ETOM;MPL;SOIA;TOG;SAP;MBS;DMO;ERML;CMDB'

If projects need a profile and desktops that combines legacy product and HOPEX solutions or that combines HOPEX Solutions, a specific study is required.

2.10. Review authentication mode

With HOPEX V3, a new authentication framework called UAS (Unified Authentication Service) is available.

Authentication configured with previous versions will run natively in most cases (see section 'Other checking indications' in this document). Anyway, it is recommended to consider the UAS framework capabilities:

- OpenID authentication (out of the box, configuration required).
- SAML2 authentication (out of the box, configuration required).

- Windows authentication (out of the box, configuration required).
- .NET project template provided to quickly develop a custom authentication provider (UAS custom provider).

For more details, see online documentation 'Installation and Deployment : HOPEX Unified Authentication Service'.

2.11. Decide to keep web settings

Web settings are user related settings. They contain information that can be considered as useful, ex:

- List of tiles selected by user in web desktop
- List of dashboards (widgets) selected by user in web desktop

With HOPEX V3, web settings persist in different folders.

If you need to restore this information when migrating to HOPEX V3, archive (file copy) the file MegaSettings-*.ini on the server hosting the source installation (ex: HOPEX V2R1).

With HOPEX V2R1, such files are saved in the folder:

%ProgramData%\MEGA\HOPEX V2R1\ClusterRoot\UserSettings

3. UPGRADE DATA FROM HOPEX V2R1 TO HOPEX V3

For each HOPEX environment, several steps are required:

- Check data upgrade pre-requisites.
- Perform technical conversion of the system database for SQL Server.
- Upgrade of **both the system database and data repositories** using an environment update wizard. It is no longer required to convert explicitly each data repository. Of course, if a data repository is referenced after environment upgrade, the environment update wizard needs to be run again so that it is converted.

Note that environment upgrade consists in running:

- **Technical conversions.** They update SQL tables and indexes to the expected format. They apply for each data repository and for the systemdb database.
- **System database upgrade.** It upgrades the metamodel and templates stored in the system database to the format expected for the target version.
- **Functional conversions.** They update the system objects stored in system database and the data objects stored in the data repositories to the expected format.

3.1. Check data upgrade pre-requisites

Before proceeding, check the following:

Check	Detail
No private workspace persists	For each HOPEX environment, check that no private workspace persists. Each private workspace must be published or discarded (abandoned)
All HOPEX environments compile without errors	For each HOPEX environment, check that you can compile (metamodel and technical data) without error
Data is backed up	Check with the system administrator that all HOPEX environment have been backed up (physical backup). Archive key configuration file of IIS application related to HOPEX (file web.config) and HOPEX installation (Megasite.ini). Check that all customizations have been backed up (physical backup).
Password of the login 'System'	For each HOPEX environment, check that the password of the login 'System' is known or set to empty before migration. This is very important since it will be requested to login with 'System'.
All IIS web sites related to HOPEX are stopped	For the machine running HOPEX, Run 'Internet Information Services (IIS) Manager': Check that All IIS web sites related to HOPEX are stopped.
All Windows services related to HOPEX are disabled	For the machine running SSP, in Control Panel, Administrative Tools, Services: Check if that all services with name beginning with 'HOPEX' are set to 'Disabled'. Ex: HOPEX Site Service Provider HOPEX Service Watchdog
No processes related to HOPEX is running	For the machine running HOPEX, in Windows Task Manager: Check if a process mgw*.exe is running. If a process persists, end it.

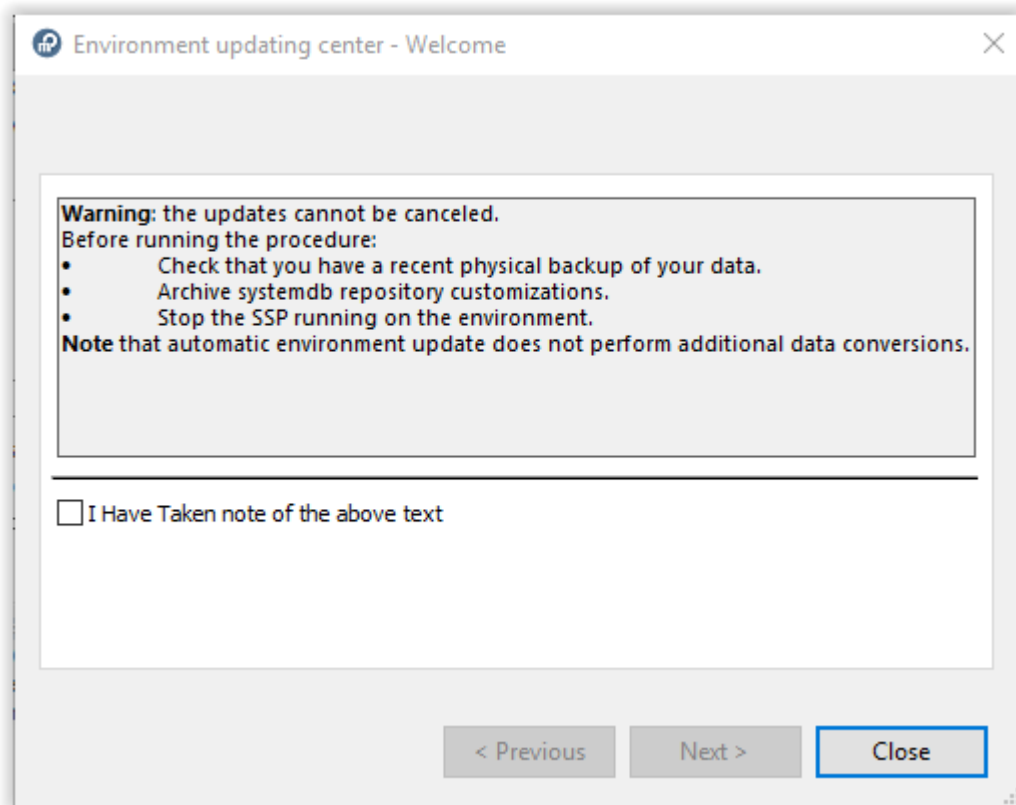
3.2. Upgrade environment with SQL Server storage

In the procedure, various warning messages will be displayed. Most of them will be ignored.
If a message is displayed that is not quoted in the procedure, see the FAQs section of this document.

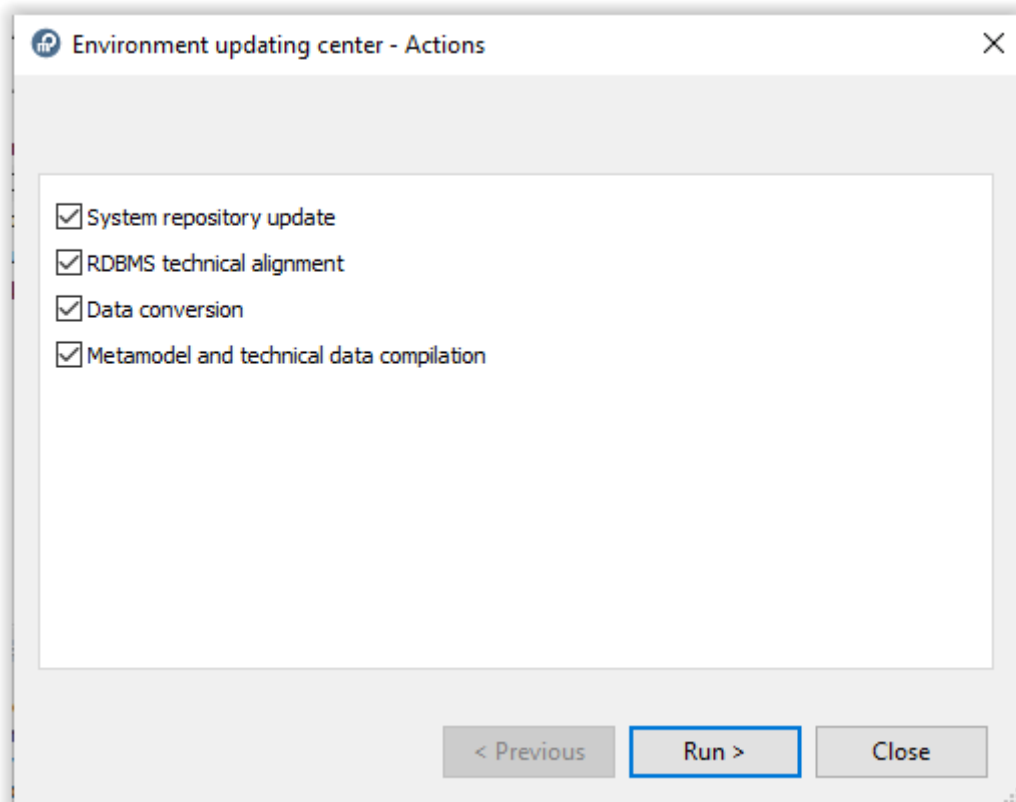
Procedure in version HOPEX V3:

1. Start the Windows Administration Console (Administration.exe).
2. Reference the environment to be converted.
3. Select the environment.
A warning can be displayed: Run the menu 'Perform SQL conversion on the repository' to perform the upgrade.
 - Click **OK** to hide the warning.
4. Select the environment and R click > **Perform SQL conversion on the repository**.
A window 'MEGA RDBMS Conversion' is displayed.
 - Click **OK** to trigger technical conversion of the SystemDb repository.
 - Wait until the conversion is over. Duration can vary according to various parameters (source and target versions, size of system database, infrastructure performances). It can last from few minutes (average time) to one hour.
A line 'Technical conversion completed' is displayed'.
 - Click **Close**.
5. Select and open the environment to be upgraded with the login **System**.
A warning can be displayed: 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** to hide the warning (it will be done later)..
A warning can be displayed for each data repository: You cannot access repository "XXX". Its internal structure is not up to date...
 - Click **OK** to hide the warning (it will be done later).
A Message is displayed: Your environment requires an update for compatibility with your version of HOPEX. Do you wish to run this procedure now?
 - Click **Yes** to trigger the environment upgrade.

A first wizard **Environment updating center - Welcome** displayed.



- Read the text, check the option **I have taken note of the above text** and click **Next**. A list of actions is displayed. Keep them checked.



- Click **Run** to start the update. Duration can vary according to various parameters (source and target versions, infrastructure performances, size and number of repositories). It usually lasts about 2 hours with one data repository.
A list of reports is displayed (one tab for each action).
- Review reports and click **Close** to exit the wizard.
- Close the environment.
- Exit the Administration Console.

3.3. Update stored procedures with SQL Server storage

This step is mandatory for each data repository or system database using RDBMS storage (SQL Server). The code of existing stored procedures (created in a previous version) needs to be initialized with the HOPEX V3.

Pre-requisite:

- Permissions to delete and create stored procedures

Procedure:

1. Start the Windows Administration Console (Administration.exe).
2. Select and open the environment with the login **System**.
3. Select the folder 'Repositories' and R click > **Manage**.
A window 'Manage repositories' is displayed
 - In the 'Repository list', check all repositories including 'SystemDb'
 - In the 'Action list', check
 - **Remove private workspace temporary data**
 - **Shrink unused repository historical data**
 - Click **Execute** and wait until the conversion is over.
 - Click 'Cancel' to exit the window 'Manage repositories'.
4. Close the environment.
5. Exit the Windows Administration Console.

Note that it is important that the execution of certain stored procedures are scheduled (batched). Refer to the document 'RDBMS Repository Installation guide HOPEX V3' to get the complete list.

4. COMPLETE UPGRADE OF DATA

4.1. Set a value for 'Definition of path of MetaAssociation'

Once a decision has been made (see section Decide 'Definition of path of MetaAssociation' sooner in the document), it must be implemented.

For each HOPEX environment:

1. Start the Windows Administration Console (Administration.exe).
2. Select and open the environment with the appropriate login (ex: system).
3. Select the environment
4. R click > **Options > Modify**
A list of options is displayed.
 - In the left tree, select the folder **Repository**
 - In the right pane, select a value for **Definition of path of MetaAssociation** according to the decision made.
5. Exit Administration Console

4.2. Re-import solutions packs

If you did not use solution packs, skip this section.

If you use solution HOPEX GDPR, skip this section. Even if you have imported a Solution Pack 'GDPR' in the past, do not import it again: You would lose update made on data.

Otherwise, solutions packs (identified before migration) need to be imported again in HOPEX V3 in particular 'Archimate'.

For each HOPEX environment, re-install each solution pack using the standard procedure.

Pre-requisite:

For each solution pack

In the HOPEX installation:

- Browse the folder \Utilities\Solution Pack
- Uncompress the .exe related to the appropriate framework
ex: PPM.exe for Solution Pack 'PPM'

Procedure:

In HOPEX installation:

- Start the Windows Administration Console (Administration.exe)
- Select and open the environment.
- Select the appropriate repository.
- R click > Object Management > **Import Solution Pack**
- For each solution pack
 - Select the appropriate framework and click 'OK'
wait until the process is completed
- Close the environment.
- Exit the Windows Administration Console.

4.3. Rebuild full search indexes

Full search required that indexes are built.

Such indexes are initialized from full indexing and completed by incremental indexing (scheduled)

The format of these indexes has changed. It is therefore required to rebuild them if full search is used. Note that full indexing can be time consuming and need resources (CPU, RAM) or large data repositories.

Procedure:

For each HOPEX environment.

For each data repository indexed.

- Browse the index folder using Windows explorer
Ex: <HOPEX environment path>\Db\<Repository>\<Repository>.ix
- Manually delete all files

Then, it is possible to rebuild them using Administration.exe

Procedure:

- Run Administration.exe
- Open each HOPEX environment.
- Select the environment
- R click > Options > Modify
- In group 'Languages', verify that the language to be indexed are checked and click OK
- For each data repository indexed.
 - Select the data repository
 - R click > Properties
 - Verify that 'Repository indexing' is checked and click OK.
 - R click > Index for full search
 - Wait for the end of the processing
- Exit

4.4. Review command line parameters

A property 'Command Line' is available at two levels:

- In properties of profile objects
- In properties of login objects (login objects are created when converting user objects)

If a string is set and contains codes that are not available for HOPEX V3 (ex: code 'PRO'), it will not be considered. No error should be displayed to screen but in the error logfile. It is therefore recommended to review command line parameters and remove codes that are not available for HOPEX V3.

Before removal	After removal
/RW'PRO,DMO'	/RW'DMO'

To identify the objects to be updated, you can run the following queries:

Object Type	Example of query for the code 'PRO'
Login	Select [Login] Where [Command Line] Like '#PRO#'
Profile	Select [Profile] Where [Command Line] Like '#PRO#'

You can get a list of codes not available for HOPEX V3 in MEGA Community, KB 00004513:

<http://community.mega.com/t5/custom/page/page-id/mega-kb-solution?sid=501D00000012hECIAY>

4.5. Convert custom report templates (MS Word) to format RTF

Standard report templates (MS Word) are natively provided in the format RTF.

Custom report templates (MS Word) need to be converted to RTF.

Pre-requisites:

Use a machine where

- HOPEX V3 is installed.
- MS Word is installed (version Office 2010/2013/2016, 32-bit version only).
- HOPEX environment to be converted are available (update file permissions).
- Verify that no process WINWORD.EXE is running.
- For each environment, manually edit file megaenv.ini, section [Office] and change DocumentFormat is needed (see below table).

Not correct	Correct
[Office] DocumentFormat=20	[Office] DocumentFormat=10

Notes:

- If the parameter DocumentFormat is not present in megaenv.ini, consider that format RTF (default format).
- DOCX is a variant of RTF format.

Procedure:

If you already use RTF/DOCX format, conversion of custom report templates (MS Word) is not required.

If you have used DOC format so far, conversion of custom report templates (MS Word) is required.

On a machine where MS Word (32 bit) is installed.

For each HOPEX environment:

6. Start the Windows Administration Console (Administration.exe).
7. Select and open the environment with the appropriate login (ex: system).
8. In the folder 'Repositories', select **Systemdb**.
9. R click > **Conversions > Utilities:**
A list of conversions is displayed.
10. Check only '**MEGA Repository - Convert Report Templates (MS Word) to RTF format**'.
Wait until the processing is finished.
11. Exit Administration Console

Result:

For the HOPEX environment

- Each report template (MS word) is saved in the RTF format in the system database.
- Each RTF style sheet used by a report template (MS Word) is duplicated and converted to the RTF format. The duplicate has the .RTF file extension and is saved in the folder 'Mega_usr' of the HOPEX environment.

It is recommended to verify in the error logfile (megaerrYYYYMMDD.txt) that each report template (MS word) has been converted without error.

4.6. Restore web settings

If you need to restore this information when migrating from to HOPEX V2R1, restore (file copy) the file MegaSettings-*.ini on the server hosting the target installation (HOPEX V3).

With HOPEX V3, such files are expected in the folder:

%ProgramData%\MEGA\HOPEX V3\ClusterRoot\UserSettings

5. CHECK UPGRADED DATA

It is highly recommended to back up each environment once it has been upgraded.

The standard installation and upgrading process takes care of all the conversions that can be automated. Technically speaking, conversion success is guaranteed by:

1. The correct execution of the environment automatic upgrade processing.
If errors are met at this step, the migration process must be stopped so that a diagnosis is made. Check carefully the Mega error log.
2. The correct execution of all mandatory conversions for the system database.
If errors are met at this step, the migration process must be stopped so that a diagnosis is made.
3. The correct execution of all mandatory conversions for each data repository.
If errors are met at this step, the migration process must be stopped so that a diagnosis is made.

After complete execution of the migration process, it is highly recommended to check data and customizations through:

- First control of migration: run a quick tour to check that data looks correct.
- Check of data consistency: run utilities to enforce rules regarding data structure.
- Other checking indications.

5.1. First control of migration

It is highly recommended to run a quick tour and check that upgraded data looks correct. Of course, this kind of check cannot be exhaustive, but it usually enables to have a first feedback and quickly identify certain migration issues.

Example of scenario:

1. Open a private workspace (ex-transaction).
2. Browse through objects using query tools, navigation trees and diagrams.
3. Perform insignificant updates (ex: change a character in a comment value, slightly move an object in a diagram...).
4. Dispatch private workspace.

5.2. Check data modelling consistency

In previous versions, many things were tolerated, although not recommended. In order to ensure better consistency, there is a need for a thorough review of the repository content and, potentially, some cleaning and tidying tasks to perform. This should be considered as a separate project.

5.3. Other checking indications

If extensions were made to the metamodel, they must be reviewed regarding the structuring rules described above. A particular attention must be paid to the orientation of MetaAssociations as it governs the behaviors of the related objects.

If customizations have been made (property pages layer, diagram configuration layer, templates, programs based on script APIs...), a specific check is required based on initial customization specifications. As customizations are often based upon standard layers, they may not be ready to use and they may have a different look and feel. This check requires functional and platform development skills.

Topic	Comment
Web Desktop related to GRC Solutions	For certain solutions (HOPEX Enterprise Risk Management, HOPEX Internal Control, HOPEX LDC), the web desktop have changed. It is required to review desktop execution if customization have been made or if projects wants to keep classic desktops for compatibility (not recommended). Review should be based on initial functional specifications. This is a customization expert work.
Profiles	It is recommended to review the profile execution for custom profiles. Review should be based on initial functional specifications.
Custom API code	It is recommended to review the customized macros and applications using API script in particular for Administration APIs. Review should be based on initial functional specifications.
Custom authentication	It is required to review authentication in case of fully customized authentication provider. Review should be based on initial functional specifications.
Web services	It is required to review web services execution. Review should be based on initial functional specifications.

The above list is not exhaustive.

6. APPENDIX

6.1. Conversion details

If mandatory conversions are not made on repositories, malfunction or loss of data can occur.
Repositories need to be converted only once.

Select a repository, right-click 'Conversions > Convert data into current version' then select the source version 'From HOPEX V2R1 data' to display conversions.

Conversions	Scope	Mandatory if upgrade from HOPEX V2R1 Update 4 to HOPEX V3
Mapping - Performances This tool deletes repository log related to mapping item. It also disables repository log for the MetaClass 'Mapping Item' This tool is implemented by a VB script macro ~ie(lCrKeJf1K[Mapping – Performances.Method])	Data	No
MEGA Repository - Add 'Substitutable' option to Report DataSet Properties This conversion updates existing Report DataSet objects to better control property display This tool is implemented by a VB script macro ~kQoLRZ50Pb87[Mega Repository - Add 'Substitutable' option to Report DataSet Properties.Method]	Data	No KB 00007470
MEGA Repository - Add 'Substitutable' option to Report DataSet Properties Update existing Report DataSet objects to better control property display This tool is implemented by a VB script macro (~kQoLRZ50Pb87[Mega Repository - Add 'Substitutable' option to Report DataSet Properties.Method])	Data	No KB 00007470
MEGA Repository - Alignment of Profile's name with Business Role's name This tools renames each profile according to the related Business Role (Ex: EA standard is renamed to Enterprise Architect). This tool is implemented in C++ and cannot be customized.	SystemDb	Optional KB 00006394
'MEGA Repository - Conversion of Assessment (Location of Assessment Deployment Query Parameter Value) Converts location of objects of MetaClass 'Assessment Deployment Query Parameter Value' from system to data It applies only if assessment templates have been customized This tool is implemented by a VB script macro MEGA Repository - Conversion of Assessment (Location of Assessment Deployment Query Parameter Value)	Data	No
MEGA Repository - Conversion of Assessment Template Definition This tool converts the objects of the MetaClass Scoring Rule to the new format.	Data	No

Conversions	Scope	Mandatory if upgrade from HOPEX V2R1 Update 4 to HOPEX V3
<p>This tool is implemented by a VB script macro ~dY(lkwG(ITj8[MEGA Repository - Conversion of Assessment Template Definition.Impl]</p>		
<p>MEGA Repository - Conversion of Assessment Template Location (Data to System) This utility converts Assessment Template to the new format. Location is transferred from data repository to system database Before HOPEX V2 Assessment templates are saved in data repository After HOPEX V2 Assessment templates are saved in systemdb This tool is implemented by a VB script macro ~OwxXvkv(M1YT[MEGA Repository - Conversion of Assessment Template Location (Data to System)]</p>	Data	No
<p>MEGA Repository - Conversion of assignments This tool creates a Profile Assignment for each Profile that were assigned to a user through a Business Role. It applies to environments using Business Role assignment in previous versions. This tool is implemented in C++ and cannot be customized</p>	Data	No
<p>MEGA Repository - Conversion of assignments (Profile mode) This tool creates a Profile Assignment for each Profile that were assigned to a user. If the access was implicit, an additional assignment is created for all repositories. The target environment must be in Profile mode (option Assignment of profiles (Management of assignment of business roles to person) checked). This tool is implemented in C++ and cannot be customized Attention: the target environment must be in Profile mode (option Management of assignment of business roles to person checked) before conversion. In megaenv.ini, the following line is expected in the section [Filter] [Filter] 47BE4D284A680315=0</p>	SystemDb	No
<p>MEGA Repository - Conversion of Business Documents or System Business Document This tool converts Business Document and System Business Document to the new format. Storage changes from disk (.DAT files) to database instance This tool is implemented in C++ and cannot be customized</p>	Data SystemDb	No
<p>MEGA Repository - Conversion of Deprecated MetaAssociation instances to Generic MetaAssociation instances This tool updates the metamodel to enable a generic management of certain MetaAssociations (Note, Document, ...). This tool is implemented in C++ and cannot be customized</p>	Data SystemDb	Yes KB 00006311
<p>MEGA Repository - Conversion of Deprecated MetaAssociationType to Operator This tool recovers behaviors associated with deprecated MetaAssociation Types</p>	SystemDb	No

Conversions	Scope	Mandatory if upgrade from HOPEX V2R1 Update 4 to HOPEX V3
The tool is implemented by a VB script macro ~W1X0aNWdMPs2[convert_deprecated_MetaAssociationType_to_Operator]		
MEGA Repository - Conversion of diagram type (described element) This tool converts Diagram Types to the new format. The generic MetaClass Described Element (or System Described Element) is used to handle the described object. This tool is implemented by a VB script macro ~piLGI9rCNfnJ[MEGA Repository - Conversion of diagram type (described element)]	SystemDb	No
MEGA Repository - Conversion of GDPR Data from Update 2 This conversion is necessary to adapt existing data to be compliant with the last version. It is mandatory if data to convert was created in Hopex U2 version. List of conversions: - Security Measures - Retention Periods - Assignments from Domain to Organization - Findings to Risks - Business Processes to Global Processing Activities - Internal Values of 'Risk Scale' Attribute This tool is implemented by a VB script macro ~ql6mQ1jyQfUQ[MEGA Repository - Conversion of GDPR Data from Update 2.Method]	Data	No
MEGA Repository - Conversion of GDPR Data from Update 3 This conversion is necessary to adapt existing data to be compliant with the last version. It is mandatory if data was created in Hopex U3 version. List of conversions: - Workflow will be available on Processing activities - Set 'Data Transfers Compliance Level' attribute value with 'Security Measures Compliance Level' attribute value - Set 'Processing Activity Purpose' attribute value with 'Processing Activity Name' attribute value This tool is implemented by a VB script macro ~FICsWwzwRXxT[MEGA Repository - Conversion of GDPR Data from Update 3.Method]	Data	Yes KB 00008599
MEGA Repository - Conversion of ITPM Assessment Updates storage of application assessment to avoid useless computation. This tool is implemented by a VB script macro (~YfgIXwaKPHD0[MEGA Repository - Conversion of ITPM Assessment])	Data	No
MEGA Repository - Conversion of ITPM Initiatives to PPM Projects Converts the former ITPM transformation portfolio initiatives into new PPM project portfolio lines.	Data	No

Conversions	Scope	Mandatory if upgrade from HOPEX V2R1 Update 4 to HOPEX V3
This tool is implemented by a VB script macro (~qrr9qL35P5NQ[MEGA Repository - Conversion of ITPM Initiatives to PPM Projects.Macro])		
MEGA Repository - Conversion of Mapping links Converts possible customization of mapping to the new format. This tool is implemented by a VB script macro (~H8rvwXEWOTqL[_MEGA Repository - Conversion of Mapping links.Method]) and can be customized	SystemDb	No
MEGA Repository - Conversion of name properties This tools aligns object names with metamodel definition Conversion may take a significant time depending on the volume of data. This tool is implemented in C++ and cannot be customized.	Data SystemDb	Optional KB 00001289
MEGA Repository - Conversion of Notification <Notification From> MetaAttribute Converts notification objects to the new format. This tool is implemented by a VB script macro (~qtTiDHWSPJ5[MEGA Repository - Conversion of Notification <Notification From> MetaAttribute.Method])	Data	No
MEGA Repository - Conversion of old MetaAssociation into deprecated MetaAssociation This tool tags old MetaAssociations as deprecated. This tool is implemented by an external script 'convert_deprecated_metaassociation.vbs'	SystemDb	No
MEGA Repository - Conversion of Persons authenticated with MEGA This conversion converts logins with Authentication mode 'MEGA' This tool is implemented by a VB script macro ~vzb5R3oFRX2I[ConvertPersonsAuthenticatedWithMEGA.Method]	SystemDb	No KB 00008006
MEGA Repository - Conversion of Posts to Review Notes Converts posts (MetaClass Post, removed) to review notes. Attention: new objects are created every time the conversion it run This tool is implemented by a VB script macro (~2b5nOh0pOrW4[Collaboration - Post Conversion])	Data	No
MEGA Repository - Conversion of Profile Permissions This conversion enables to convert profile permissions to new format This tool is implemented by a VB script macro ~sx41ZhG)RLb2[convert_profile_permission])	SystemDb	Yes KB 00008600
MEGA Repository - Conversion of report parameters This conversion sets the value of the new MetaAttribute Values Definition with the value of the deprecated MetaAttribute Kind of proposition. This tool is implemented by a VB script macro ~yCREuDrcSHcA[Conversion of report parameters.Method]	Data	Yes KB 00008598

Conversions	Scope	Mandatory if upgrade from HOPEX V2R1 Update 4 to HOPEX V3
MEGA Repository - Conversion of Restricted MetaAssociation into Abstract MetaAssociation Converts restrictive MetaAssociation to the new format When specific MetaAssociations were made restrictive, parent MetaAssociations will be declared Abstract.	SystemDb	No
MEGA Repository - Conversion of Restrictive MetaAssociation instances to Concrete MetaAssociation instances This tool updates the metamodel to move down restricted MetaAssociation instances storage on their concrete MetaAssociation. This tool is implemented in C++ and cannot be customized.	Data SystemDb	Yes KB 00006309
MEGA Repository - Conversion of tree folder menu This tool recovers menu items 'New' and/or 'Connect' on folders in tree This tool is implemented by a VB script macro ~(2H4f8nGNTOG[Convert_treefolder_menuitems])	SystemDb	No
MEGA Repository - Conversion of Where Used Queries This tool converts the format of a configuration regarding Diagrams containing object. From HOPEX V1R2 CP1.0, queries are connected directly to the MetaClass. This tool is implemented by a macro calling a VBS script file (convert_where_used_queries.vbs).	SystemDb	No
MEGA Repository - Conversion of Working Environment Template Group with desktops It is no longer possible to connect Desktop to a Working Environment Group Template This conversion converts to the new format Working Environment Group Template This tool is implemented by a VB script macro (Macro ~x2CwVr)YQDWB[ConvertWETGroupDesktopToCommands.Method])	SystemDb	No
MEGA Repository - Conversion of Working Environment Template Profile Assignments This conversion enables to convert Desktop objects into desktop manager for each concerned Working Environment Template Profile Assignment. This enables to load the appropriate Desktop for client device. This tool is implemented by a VB script macro (~53bgJlhXQH88[MEGA Repository - Conversion of Working Environment Template Profile Assignments.Method])	SystemDb	Optional To run HOPEX Explorer (V2) on a tablet device KB 00007744
MEGA Repository - Conversion of Working Environment Templates This conversion enables to define several Working Environment template with different desktops. This tool is implemented by a macro (~24cD8qJKP5qE[MEGA Repository - Conversion of Working Environment Templates.Method])	SystemDb	No
MEGA Repository - Convert Property Page Link/Tree to PropertyPageExtension	SystemDb	No

Conversions	Scope	Mandatory if upgrade from HOPEX V2R1 Update 4 to HOPEX V3
<p>This utility converts property page implementation from _PropertyPageLink to _PropertyPageExtension. It is recommended to run this utility to benefit from new customization capabilities</p> <p>This tool is implemented by a VB script macro ~zTupdajlrbB[MEGA Repository - Convert Property Page Link/Tree to PropertyPageExtension]</p>		
<p>MEGA Repository - Repair Building Block Containment</p> <p>Creates connections between building block (ex: library) and building block annotation (ex: report) for compliance with EA grid</p> <p>This tool is implemented by a VB script macro (~JkZBYfprP9EV[MEGA Repository - Repair Building Block Containment])</p>	Data	No
<p>MEGA Repository - Update initial foundation : 'MEGA Library'</p> <p>Imports mandatory data to the data repository</p> <p>This conversion is implemented by a script (convert_update_initial_foundation.vbs) and can be customized</p>	Data	No
<p>MEGA Repository - Update Setting for Building Block Role extension metaclasses</p> <p>Updates custom creation tools.</p> <p>This tool is implemented by a VB script macro by a VB script macro (MEGA Repository - Update Setting for Building Block Role extension metaclasses.Method)</p>	SystemDdb	No
<p>MEGA Teamwork - Conversion of location of workflow instance</p> <p>This tool changes the location of workflow instance from system database to data repository.</p> <p>This tool is implemented by a VB script macro (~5m5fIWg5KPfB[Mega TeamWork - Conversion of Workflow Instance System.Method])</p>	Data	No
<p>WET Home Conversion</p> <p>This conversion updates Working Environment Template object to the new format</p> <p>_Operator ~ewv(L)8R1A7[WET Home Conversion] Macro ~2hvvHN)8R5D7[WET Home Conversion.Implement]</p>	SystemDb	No KB 00008631

6.2. Utilities details

Utility	Scope	Comment
Diagram (drawings) This tool opens, saves and closes all diagrams in the repository. Enables conversion of diagrams with drawings in MGE format. Also enables to check the status all diagrams in a repository. This execution is optional for the system database and data repositories. Conversion may take a significant time depending on the volume of data. This tool is implemented in C++ and cannot be customized.	Data SystemDb	Optional KB 00001270
GRC v2 - Compute and Store Assessment KPIs This tool computes and store the following indicators: - Last Assessment Date for Risks and Controls - Aggregated Execution Rate for Controls - Aggregated Pass Control Level for Controls This tool is implemented by a VB script macro ~c9tcUnPrSPCQ[GRC v2 - Compute and Store Assessment KPIs.Method]	Data	Optional KB 00008632
HITA - Convert Software Technology Fulfillments to Technical Capabilities This conversion convertst the Business Capability Fulfillment or Functionality that could exist between Software technology objects and Business Capability objects to the new format. This tool is implemented by a macro (~10v92q21QHtG[HITA - Convert Software Technology Fulfillments to Technical Capabilities]) and can be customized	Data	Optional KB 00007770
HOPEX ITPM -Conversion of Standards This tool converts certain objects of the Metaclass 'Standard' to objects of the MetaClass 'Technology'. Selection is made using the query '~5yCf7ugklr9D[APM - Conversion - Get Standards Linked to Vendors Or Application Deployed]' This tool is implemented by a VB script macro ~2yCfd0hkITND[APM - Conversion of Standards]	Data	Optional KB 00004589
MEGA Repository - Business Function to Business Functional Area Converts a Business Functions Hierarchy from Process or BPA solution to Business Architecture's Business Functional Area composite structure models. Useful to migrate data created with MEGA Process (code PRO) and/or MEGA Process BPMN Edition (code PMN) to the solution Business Architecture (code HBAS) This tool is implemented by a macro (~18VF)i4hNf)I[MEGA Repository - City planning areas to Business Capabilities.Method]) and can be customized	Data	Optional KB 00007465
MEGA Repository - City planning areas to Business Capabilities Convert City Plans to Business Capability Maps and City Plan Areas to Business Capability objects. Useful to migrate data for the solution IT Portfolio Management (Code APM). This tool is implemented by a macro (~18VF)i4hNf)I[MEGA Repository - City planning areas to Business Capabilities.Method]) and can be customized	Data	KB 00007191 Optional
MEGA Repository - Cleanup This tool removes technical temporary data left invalid in repositories after upgrade (ex: recent queries). This tool is implemented by a VB script macro ~W7qD9X3HCT50[MEGA Repository - Cleanup.Method]	Data	Optional KB 00003321
MEGA Repository - Conversion of EA Projects to PPM projects Converts EA Project (former MetaClass Project renamed to EA Project) to PPM project (new MetaClass Project) and Project scope to Project Deliverables, when applicable	Data	Optional KB 00007466

Utility	Scope	Comment
Useful with option Product Portfolio Management (code PPM) and a compatible solution (ex IT Portfolio Management...) This tool is implemented by a macro (~6qr9AW25PntP[Mega Repository - Conversion of EA Project.Method]) and can be customized		
MEGA Repository - Conversion of ITPM Applications Exchanges (ARC -> HITA) Converts the description of application exchanges (based on message flows) to the new format used with IT Architecture (code HITA) Useful with solution IT Portfolio Management (Code APM) and new solution IT Architecture (code HITA) This tool is implemented by a macro (~ougT5TtIP1eB[HOPEX ITPM - Conversion of Applications Exchanges (ARC -> HITA)]) and can be customized	Data	Optional KB 00007461
MEGA Repository - Conversion of name properties (long name) This tool aligns object names with metamodel definition (long name) for certain MetaClasses. Conversion may take a significant time depending on the volume of data. This tool is implemented in C++ and cannot be customized.	Data	Optional KB 00001892
MEGA Repository - Conversion of Notes to Review Notes Creates review notes from notes This tool is implemented by a macro (~sa5nwd0pO5U4[Collaboration - Note Conversion]) and can be customized	Data	Optional KB 00007469
MEGA Repository - Conversion of Organizational Charts This utility converts the nature of Organizational Chart diagrams so that they can be open with MEGA Process BPMN Edition. This tool is implemented by a VB script macro ~YgaCFMJSGPv2[Organisational Chart Conversion] and can be customized	Data	Optional KB 00003984
MEGA Repository - Conversion of Specific Name of Dictionary Object to Term This conversion updates objects based on terms to the new format in case name has been customized This tool is implemented by a VB script macro (~ltqQKs6TQLoG[Mega Repository - Conversion of Specific Name of Dictionary Object to Term.Implementation]) and can be customized	Data	Optional KB 00007723
MEGA Repository - Conversion of widgets Converts dashboard widgets to a new format (independant from container of web desktop) To keep widgets used in previous versions, a prerequisite to this conversion is to restore web settings. This tool is implemented by a VB script macro ~DcdsKyj4QDAE[WidgetConversion.Method] and can be customized	SystemDb	Optional KB 00007569
MEGA Repository - Convert participants of projects This tools converts participants of projects to the new format This tool is implemented by a VB script macro ~MKy3t2XCnf7U[Convert participants of projects.Method]	Data	Optional KB 00006308
MEGA Repository - Convert Report templates (MS Word) to RTF Format This tool converts Report templates (MS Word) from Word to RTF format. This is required to generate documents with HOPEX Web Front-end. This tool is implemented in C++ and cannot be customized. Note that MS Word is required on the machine running the conversion.	SystemDb	Optional Recommended If custom template and decision to user format RTF. KB 00003499
Mega Repository – Convert Web EA Project This tool converts requirements of EA projects to new format	Data	Optional KB 00008633

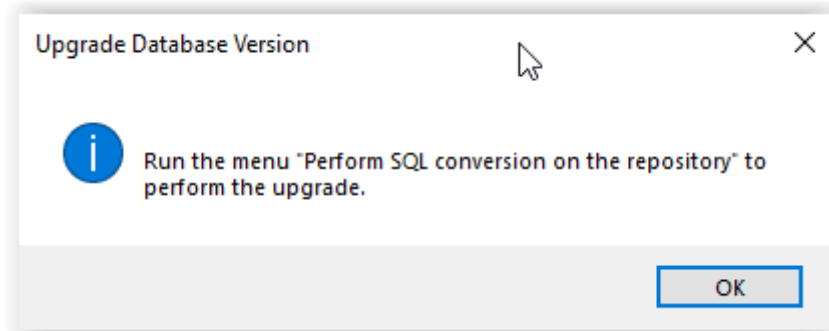
Utility	Scope	Comment
<p>This tool is implemented by a VB script macro ~\Tyap4AVOn3G[Mega Repository – Convert Web EA Project.Method]</p>		
<p>MEGA Repository - Creation of links instances from MEGA fields This tool creates impact analysis links for objects referenced by object references (MEGA fields) in texts properties. Conversion may take a significant time depending on the volume of data. This tool is implemented in C++ and cannot be customized.</p>	Data SystemDb	Optional KB 00002005
<p>MEGA Repository - Objectives -> Enterprise Objective Conversion Creates Enterprise Objective from Objectives Useful to migrate data from Business Strategy (code MBS) to Business Architecture (Code HBAS) This tool is implemented by a macro (~\ueHLXNrMP1sV[Objectives -> Enterprise Objective Conversion.Method]) and can be customized</p>	Data	Optional KB 00007463
<p>Shapes This tool updates customized shapes to the most recent format. Shapes located in the folder 'Mega_usr' or both installation and HOPEX environment are upgraded. This conversion is optional for the System repositories. This tool is implemented in C++ and cannot be customized.</p>	SystemDb	Optional KB 00000362

7. FAQs

7.1.1. Warning 'Run the menu 'Perform SQL conversion on the repository' to perform the upgrade

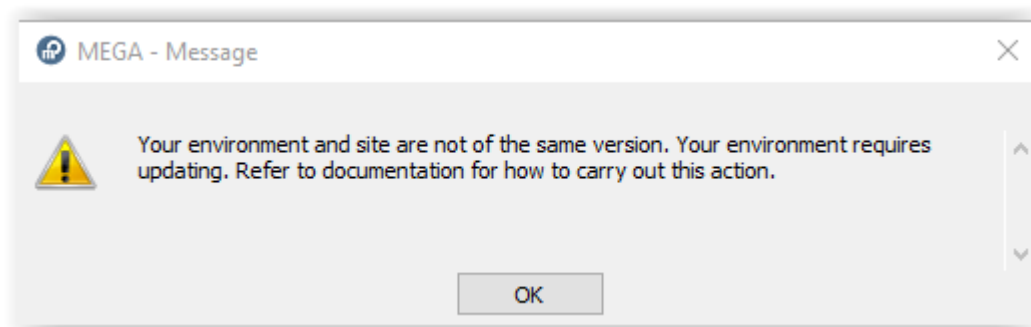
This means that the format of tables in SQL Server must be converted.

You need to run a menu (Perform SQL conversion on the repository) from the Administration Console. See earlier in this document.

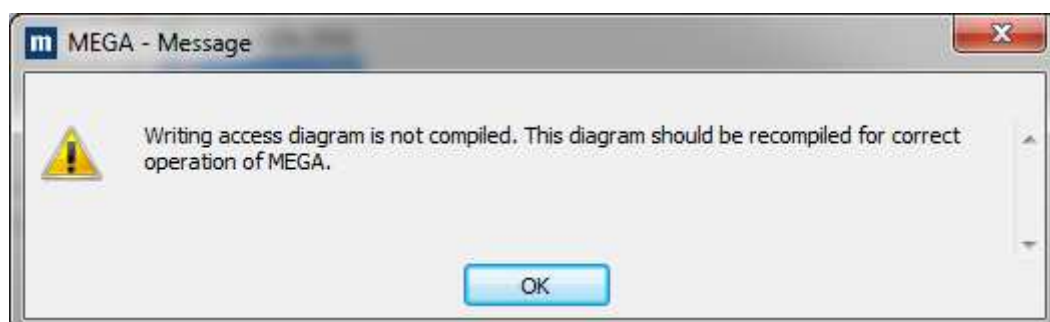


7.1.2. Warning 'Your environment and site are not of the same version. Your environment requires updating...'

This warning report that the system database is not up to date. This occurs if the programs have been updated and the environment has not/not yet been updated.



7.1.3. Warning 'Writing access diagram is not compiled...'



Certain actions can leave the writing access diagram (ex-User diagram/Authorization diagram) in a state not compiled.

To compile the metamodel of the environment:

1. Start the Windows Administration Console (Administration.exe).
2. Select and open the environment to be converted with the login **System**.
3. Select the folder 'User management'
4. R click > Compile writing access diagram
5. Click Start to trigger the compilation
Wait until the processing is complete.
6. Click 'Close'
7. Exit the Windows Administration Console

7.1.4. How to run a specific conversion or utility?

It is now required to use the update wizard instead of triggering explicitly conversions.

It is still possible to run conversions or utilities.

Procedure for conversions:

1. Start the Windows Administration Console (Administration.exe).
2. Select and open the environment to be converted with the login **System**.
In the folder 'Repositories', select **SystemDb or a data repository**
 - R click > **Conversions > Convert data into current version** and select **From HOPEX V2R1 data**.
 - Check the appropriate conversions.
See the table 'Conversion details', later in this document
 - Click 'OK' to trigger the conversion
Wait until the conversion is complete
 - Close the environment
3. Exit the Windows Administration Console.

Procedure for utilities:

1. Start the Windows Administration Console (Administration.exe).
2. Select and open the environment to be converted with the login **System**.
In the folder 'Repositories', select **SystemDb or a data repository**
 - R click > **Utilities**.
 - Check the appropriate conversions.
See the table 'Conversion details', later in this document
 - Click 'OK' to trigger the conversion
Wait until the conversion is complete
 - Close the environment
3. Exit the Windows Administration Console.

HOW TO INSTALL CP HOPEX V3

Summary

Check if a more recent version of this document is available in online documentation (MEGA Community).

This document describes the procedures necessary for installing a CP for HOPEX V3.

It applies only to HOPEX V3.

1. FOREWORD	3
1.1. Corrective Pack	3
1.2. Identification of version	3
2. UPGRADING HOPEX PROGRAMS	4
2.1. Upgrading Web Front-end	6
2.2. Upgrading HOPEX Data	7
3. FAQs	10

1. FOREWORD

1.1. Corrective Pack

With HOPEX V3 there are several levels of changes:

- **Update:** to deliver improvement and fixes. GUI can change.
Ex: Update 2, Update 3, Update 4
- **Corrective Pack (CP):** to deliver fixes only within an update. GUI should not change.
Ex: Corrective Pack 1 on HOPEX V3 (no update), corrective Pack 2 on HOPEX V3 Update 1
- **Hotfix:** to delivery urgent fixes. GUI does not change.
Ex: Hotfix 02 on HOPEX V3 Update 1 Corrective Pack 2

In concrete terms, a Corrective Pack installation program is a .MSP file embedded as a .EXE file.

Example: HOPEX_V3.00.01.exe.

Each Corrective Pack is related to an update level of a major version.

Example:

HOPEX_V3.00.01.exe: this CP1 applies to HOPEX V3 (initial release, no update).

HOPEX_V3.01.01.exe: this CP1 applies to HOPEX V3 Update 1.

Corrective Packs **are not cumulative**.

Data upgrade can be done in the final CP

Example: to upgrade from HOPEX V3 CP1 to HOPEX V3 CP4, it is required to:

1. Install CP2 on HOPEX V3 to upgrade to HOPEX V3 CP2.
2. Install CP3 on HOPEX V3 to upgrade to HOPEX V3 CP3.
3. Install CP4 on HOPEX V3 to upgrade to HOPEX V3 CP4.
4. Run the 'Environment Automatic Update' feature.

As a consequence, verify the expected update level is installed before installing a Corrective Pack.

Before proceeding, make sure that, for all the HOPEX environments to upgrade:

- Data is backed up (physical backup).
- The password of the login **System** is known.
This is very important since it will be requested to login with the login System.

1.2. Identification of version

Versions can be found through the About HOPEX menu.

HOPEX <Major version code> U<Update number>.<Corrective Pack number> (Build number)

Example: HOPEX V3.00.01 (7.87.5331.0000)

- Major version code: V3
- Update number: 00
- Corrective Pack number: 01
- Build number: 7.85.5331.0000

2. UPGRADING HOPEX PROGRAMS

The Corrective Pack installation program is an .MSP file embedded as an .EXE file.

Example: HOPEX_V3.00.01.exe.

It must be installed on each machine where the version HOPEX V3 has been previously installed. The components initially installed (HOPEX Kernel and/or IIS components) will be updated.

Front-end	Deployment	Target machine
Web Front-end	Standalone deployment	Unique Server where HOPEX programs are installed
Web Front-end	Cluster deployment	Each application server of the cluster where HOPEX programs are installed <ul style="list-style-type: none"> • HOPEX Front-end • HOPEX Back-end • SSP
Windows Front-end	Standard deployment	Each workstation where HOPEX programs are installed
Windows Front-end	Citrix/Terminal Server deployment	Each Citrix/TSE application server where HOPEX programs are installed

Deployment	Machine	Windows Service	IIS Web site (1)	HOPEX processes
Web standalone deployment	Unique server	Yes	Yes	Yes
Web cluster deployment	Server running SSP	Yes		Yes
Web cluster deployment	Server running HOPEX Back-end Server running HOPEX Front-end	Yes		Yes
Windows Citrix/Terminal Server deployment	Each workstation	No		Yes
Windows standard deployment	Each Citrix/TSE application server	No		Yes

(1) By default, an application 'HOPEX' is configured for 'Default Web Site'.

2.1. Upgrading Web Front-end

2.1.1.Pre-install

- Identify the target machine(s).
The machine varies with the chosen deployment: see the above table.
- Archive key configuration file of IIS application (web.config file) and HOPEX installation (Megasite.ini).
- Login in as administrator of the machine.
- Verify 'Control Panel > Administrative tools > Services'.
On the SSP server, the service **HOPEX Site Service Provider** must be set to 'Stopped'.
On all the servers, the service **HOPEX Service Watchdog** must be set to 'Stopped'.
- Verify 'Internet Information Services (IIS) Manager'.
The web site hosting the IIS applications (by default it is 'Default Web Site') must be stopped.
- Verify the Task Manager.
No HOPEX process (mgw*.exe, or HOPEX*) must be running.
- Verify 'Control Panel > Add or Remove programs'.
The required update level must be installed.
For example, HOPEX V3 is a requirement before installing Corrective Pack 1 on HOPEX V3 (HOPEX_V3.00.01.exe).

Note: you can also use the utility Hopex Server Supervisor and R click > HOPEX > Stop HOPEX Processes Services and Web Application.

2.1.2.Procedure

For each machine:

1. Select the .EXE file of the Corrective Pack.
Example: Select HOPEX_V3.00.01.exe.
2. R click > **Apply**
The wizard can take more than one min to load.
3. Click **Update >**.
The installation process can take one minute to initialize.
Wait until the processing is complete.
4. Click **Finish**.

Notes:

- The location of the installation folder can be found after a search is conducted in the machine registry. This location is not visible on the machine during installation or in the control panel.
- If the expected update level is not identified for the HOPEX programs registered on the machine, an error is displayed.

2.2. Upgrading HOPEX Data

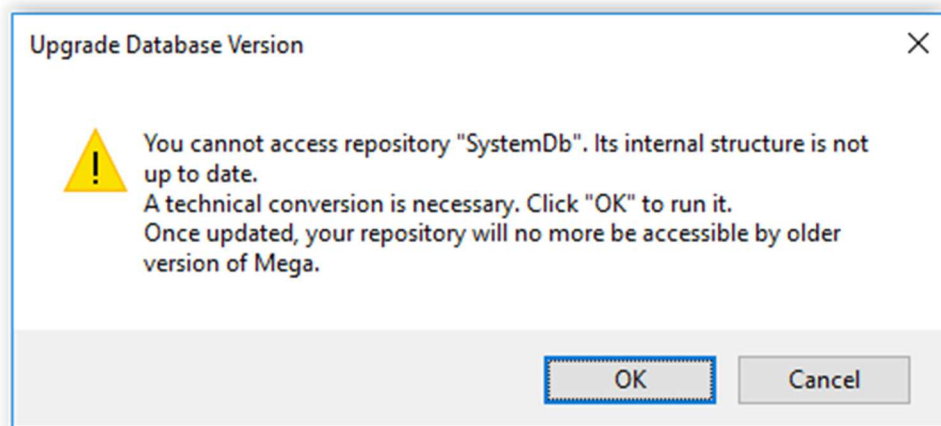
Most changes apply to HOPEX programs. However, some changes apply to the systemdb repository. It is therefore necessary to update all HOPEX environments to benefit from all the changes and fixes. The procedure varies depending on the storage.

2.2.1. Pre-upgrade

- Verify that no workspace exists in read/write mode.
- Verify 'Control Panel > Administrative tools > Services'.
On the SSP server, the service **HOPEX Site Service Provider** must be set to 'Stopped'.
On the all servers, the service **HOPEX Service Watchdog** must be set to 'Stopped'.
- Verify the Task Manager.
No HOPEX process (mgw*.exe, or HOPEX*) must be running.

2.2.2. Procedure

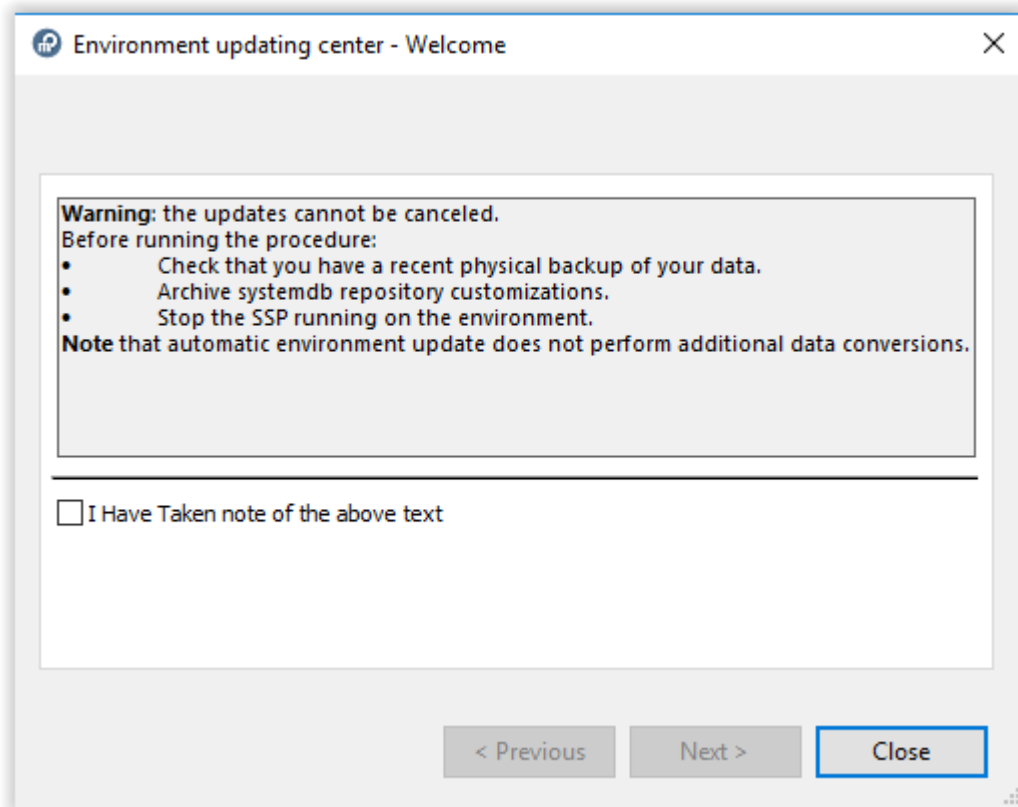
With RDBMS storage, the technical format of the system repository and data repositories may change when installing update. In this case, a warning is displayed:
You cannot access repository XXX. Its internal structure is not up to date. ...



For each HOPEX environment:

1. Start the Administration Console (Administration.exe).
2. Select the environment to be upgraded.
3. R click > **Open**.
4. If a warning is displayed (You cannot access repository "SystemDb". Its internal structure is not up to date. ...), click 'Cancel', then :
 - Select the environment in the administration tree.
 - R click > **Perform SQL conversions on the repository**
A window 'MEGA RDBMS Technical Conversion' is displayed.
 - Click 'OK'.
 - Wait until the processing is complete (it takes a few minutes) and click 'Close'.
5. Select the environment to be upgraded.
6. R click > Open.
7. Login with the **System** identifier.
8. If warnings are displayed ("You cannot access repository "XXX". Its internal structure is not up to date..."), for each data repository (ex: ProductionData):

- Select the data repository in the administration tree.
 - R click > RDBMS Administration > **Perform SQL conversions on the repository.**
A window 'MEGA RDBMS Technical Conversion' is displayed.
 - Click 'OK'.
 - Wait until the processing is complete (it usually takes a few minutes) and click 'Close'.
9. Select the environment
10. R click > **Environment automatic update**
A wizard 'Environment updating center - Welcome' is displayed:



11. Read the information, check the option 'I have taken note of the above text' then click **Next**. A list of actions is displayed. It is recommended to keep them checked.
12. Click **Run** to start the update. Duration can vary according to various parameters (source and target versions, infrastructure performances, number of data repositories). It usually lasts about 15 min. A list of reports is displayed (one tab for each action).
13. Review reports and click **Close** to exit the wizard.
14. Close the environment.
15. Exit the Administration Console.

2.2.3. Post-installation and data upgrade

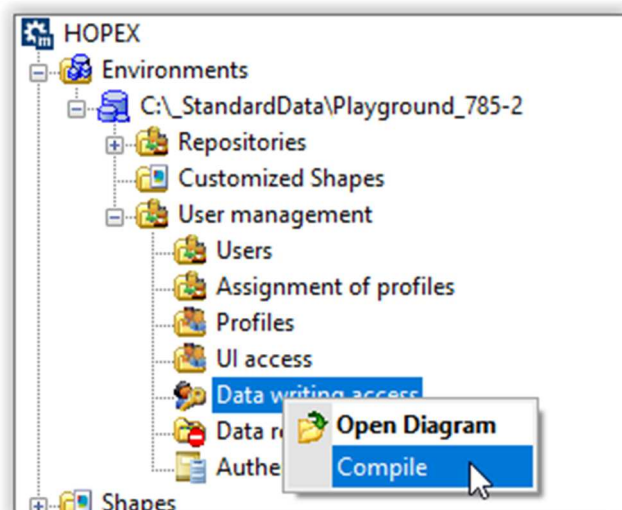
- Verify that data reading access compilation (ex-confidentiality) if necessary.
See below.
- Update HOPEX configuration if necessary.
See below.
- Verify 'Control Panel > Administrative tools > Services'.
On the SSP server, the service **HOPEX Site Service Provider** must be set to 'Automatic (Delayed Start)' and be started.
On the all servers, the service **HOPEX Service Watchdog** must be set to 'Automatic (Delayed Start)' and be started.
- Verify 'Internet Information Services (IIS) Manager'.
The web site hosting the IIS applications (by default it is 'Default Web Site') must be started.

Note: you can also use the utility Hopex Server Supervisor and R click > HOPEX > Restart HOPEX Processes Services and Web Application.

Verify data reading access compilation (ex-confidentiality) if necessary

If you use data reading access management feature (ex-confidentiality), check that data reading access is compiled

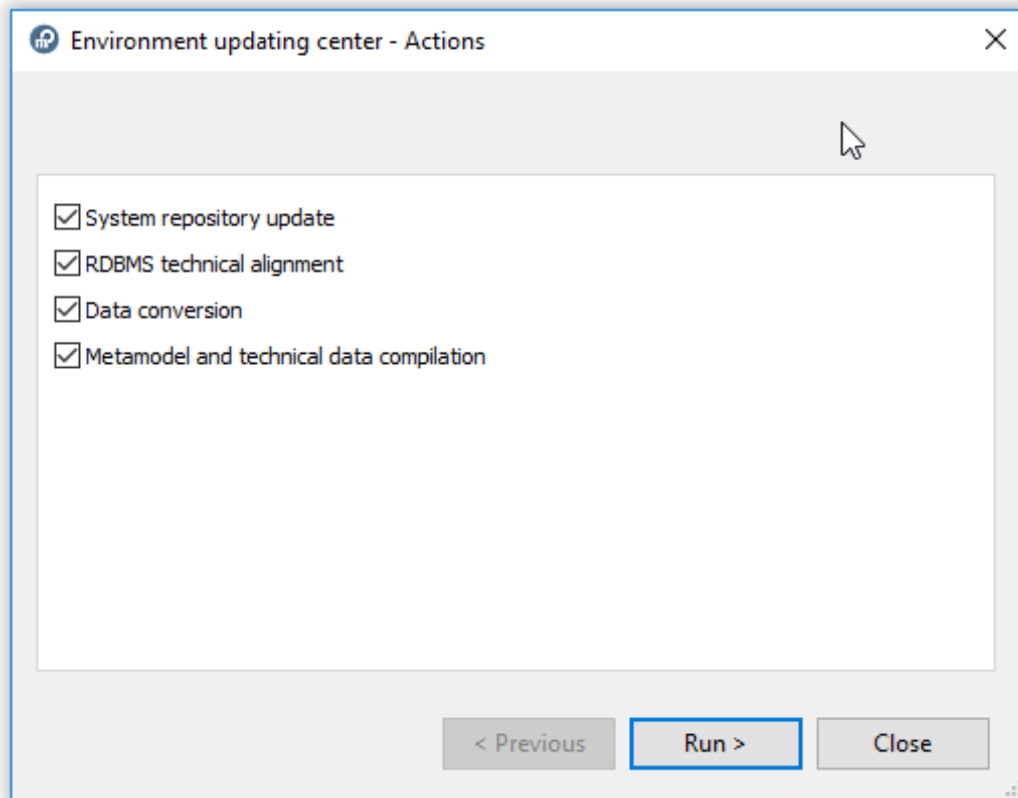
1. Start the Administration Console (Administration.exe).
2. Select the environment to be upgraded.
3. R click > **Open**.
 - In the administration tree, select the folder User management > **Data writing access**.
 - R click > **Compile**
Wait for end of processing (compilation completed)
4. Exit the Administration Console (Administration.exe).



Note this feature is not active by default.

3. FAQs

3.1.1. What is the meaning of the actions below?



System repository update: update of system repository by import of .MOL files.

RDBMS technical alignment: update of tables at SQL level (RDBMS only).

Data conversion: additional conversion for system repository and data repositories.

Metamodel and technical data compilation: compilation.

3.1.2. How can I get the MSP file of the Corrective Pack?

Procedure:

- Copy the .EXE of the Corrective Pack file in a temporary folder, ex: C:\tmp\HOPEX_V3.00.01.exe
- Open a command prompt window (CMD)
- Run the following command <path of EXE> /E, Ex: C:\tmp\HOPEX_V3.00.01.exe /E
- The installation wizard is loaded. Click 'Cancel' then 'Yes' and 'Finish' to exit the wizard.
- In the temporary folder, a .MSP file has been created, ex: MEGA HOPEX V3 Patch 501.msp

How to Migrate Data from Oracle to SQL Server HOPEX V3

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 3 029 190 € - RC Paris B 385 185 806 000 51 / NAF 741 G

Summary

Check if a more recent version of this document is available in online documentation (MEGA Community).

This document describes the procedures necessary for migrating data from Oracle storage to SQL Server storage.

It applies to HOPEX V3.

This migration is an administration task performed by HOPEX Administrator in coordination with Oracle DBA and SQL Server DBA.

Note that repository snapshot objects are tied to low level information and cannot be transferred when data repository is rebuilt logically.

This means that they will NOT be available after migration from Oracle to SQL Server.

1. PRE-REQUISITES.....	3
2. RECOMMENDATIONS	5
3. PROCEDURE	6
3.1. General process	6
3.2. Procedure.....	6
4. CHECK DATA	11

1. PRE-REQUISITES

1) Stop user activity and backup data

Follow standard procedures.

The objective is to make sure that data are not accessed not updated.

This means an interruption of service. Duration can vary according to volume of data.

Dummy run will enable to evaluate real duration.

2) Verify that a physical backup of data is available

Follow standard procedures.

3) Dispatch of delete all pending workspaces

Follow standard procedures.

The objective is that no pending update exist.

They can be exported as MGR file, cancelled or dispatched.

3) Verify you have enough working space

The process used the reorganization feature of HOPEX.

For each repository, data will be dumped to a large command file on disk using logical backup feature.

The generation folder is a subfolder of HOPEX environment folder. There is one file per repository. Ex:

<Myenvironment path>\SysDb\WORK

<Myenvironment path>\Db\MyDataRepository\WORK

4) Create SQL Server database with appropriate permissions

See SQL Server DBA.

For each repository (data repository or system repository) a SQL Server is required.

SQL Server database must be empty.

A SQL server user with appropriate permission must be created.

See HOPEX Administrator for details.

For more details, see only documentation.

Installation and Deployment : RDBMS Repository Installation Guide : SQL Server support, SQL Server Requirements.

Basic metrics:

For logical backup file: size of Oracle dump for the schema of the repository **x 5**

For logical backup file: size of Oracle dump for the schema of the repository **x 1**

Example	Size of .DMP in Oracle	Size expected for SQL Server database	Disk expected for logical backup file
Schema of MyRepository	5 Go	5 Go	25 Go
Schema of SystemDb	3 Go	3 Go	15 Go

5) Performances tuning indications

The article below contains optional indications to favor performances when performing mass import in SQL Server

<https://community.mega.com/t5/custom/page/page-id/mega-kb-solution?sid=5012p0000015TWLAA2>

2. RECOMMENDATIONS

1) Decide if you can delete repository log

Deciding to delete full/partially repository log has several benefits:

- Process is faster
- Process is more reliable
- Process requires less working space
- Final data will be less voluminous

2) Upgrade to most recent version available.

The version must be HOPEX V2R1 Update 3 or higher.

The objective is to benefit from recent fixes and to be under maintenance if ever an error occurs

- Make sur you are in a supported version
- Apply last CP/hotfix.

3) Use a server machine close to SQL Instances

The machine should be located as close as possible from Oracle instance and SQL server instance.

The objective is to have a performant and reliable access to data.

4) Leave the process running quietly.

In particular:

- Do not stop the HOPEX Process of the machine where it is running
- Do not stop Oracle instance (source data)
- Do not stop SQL Server instance (target data)
- Do not try to move the HOPEX environment folder used

5) Check process (test) on a copy of production

The more recent backup, the better vision you will have (duration, success).

a) Restore data according to standard procedures

b) Check that logical backup runs fully (dummy run)

If errors occur (related to logical backup step), analyze the errors with MEGA Technical Support.

c) Check complete process (dummy run)

If errors occur (likely related to import step), analyze the errors with MEGA Technical Support.

6) Loop until the processing runs without unexpected error.

You will have a good idea of

- Disk size required (logical backup file)
- SQL Server size required
- Global duration of operations

If an error occurs and the process fails or stops, do NOT reuse the target SQL Server database

For each repository involved:

- Delete the SQL Server database.
- Create a new SQL Server, possibility with same name.

3. PROCEDURE

3.1. General process

For each environment:

- Reorganize data repository **first**
- Reorganize system repository **last**

For each repository, it will be required to set:

- SQL Server Instance used
- SQL Server user/password used

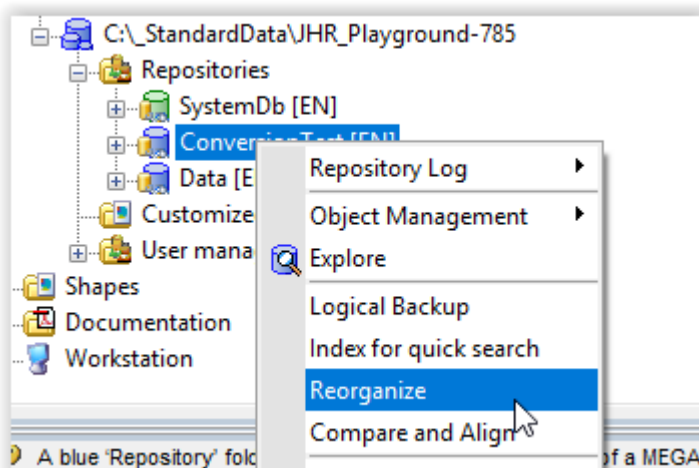
3.2. Procedure

Run Administration.exe

Open environment

Select the repository to reorganize

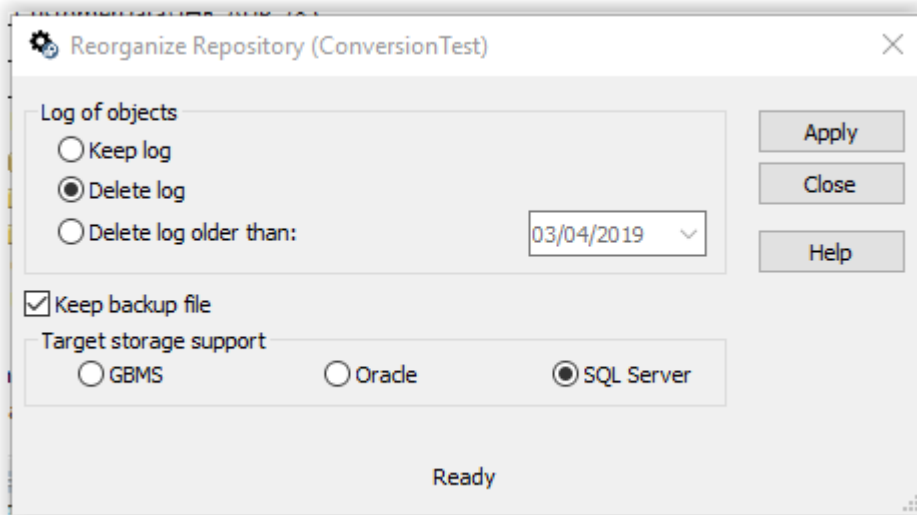
Right click > **Reorganize**



Configure window (Reorganize Repository (XX))

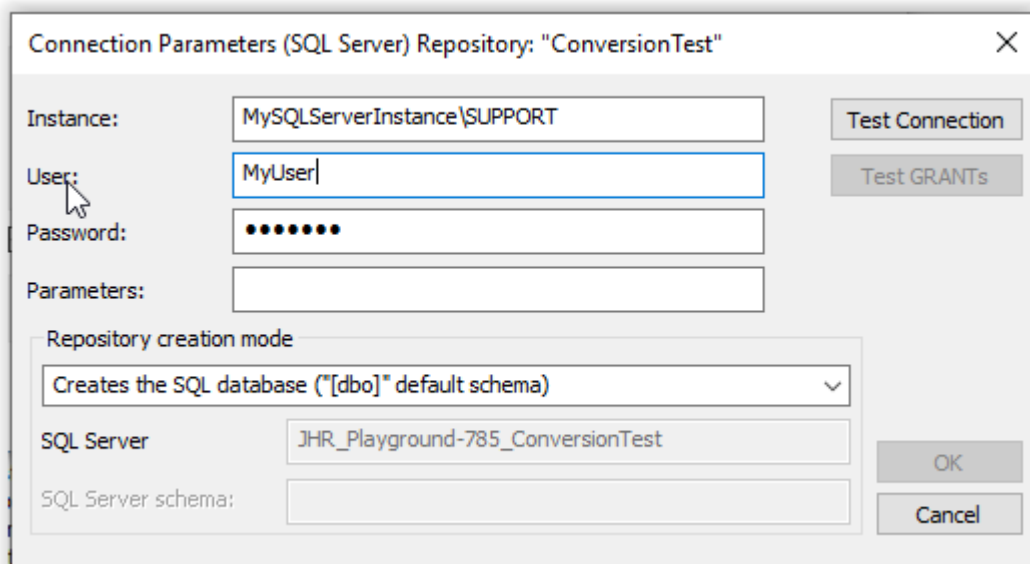
- Set **Logs of objects** (keep log, delete log, delete log order than date) according to decision
- Keep backup file** checked
- Select target storage support **SQL Server**

Example

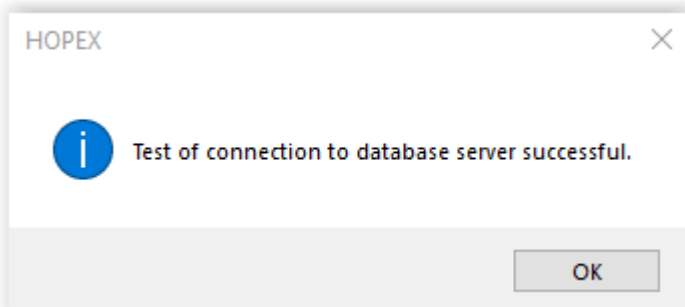


Click Apply

Set Connection parameters

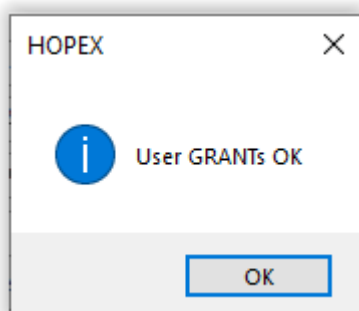


Click button 'Test Connection'

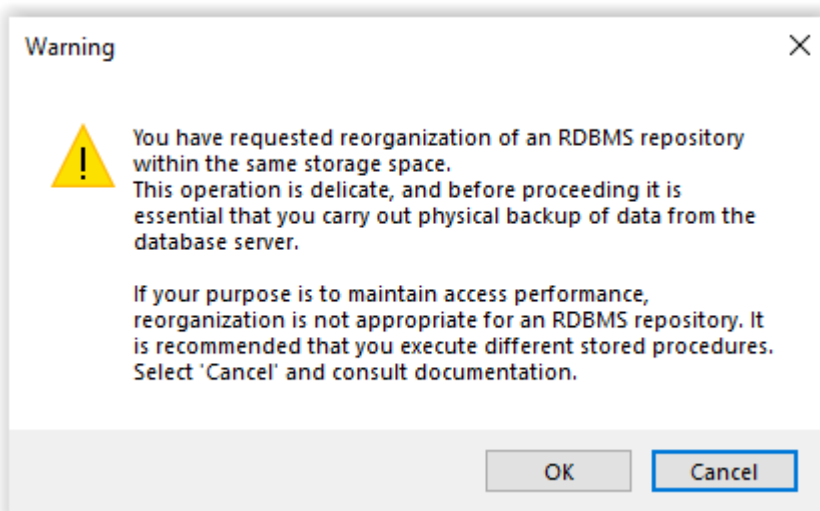


Click OK

Click button 'Test GRANTs'

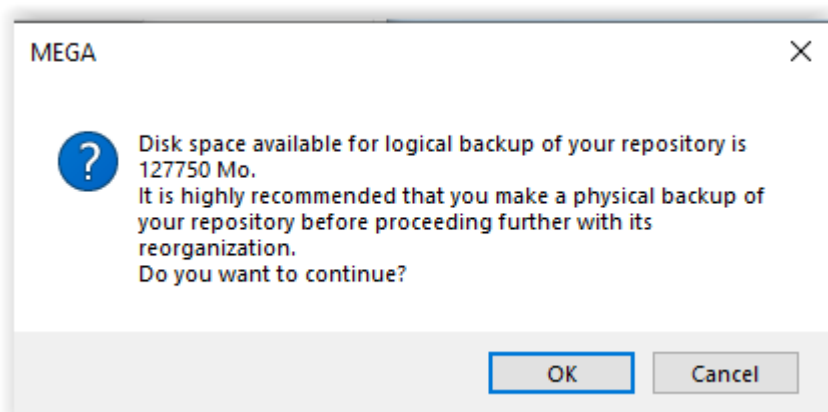


Click OK



Click OK to confirm

Note that reorganization is appropriate in this situation even if the following warning can create a doubt.
If your purpose is to maintain access performances, reorganization is not appropriate for an RDBMS repository

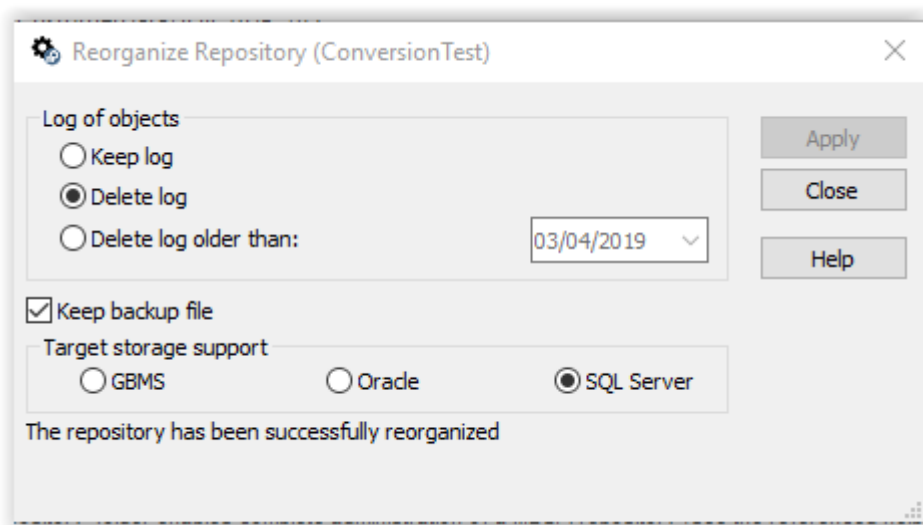


Click 'OK' to confirm

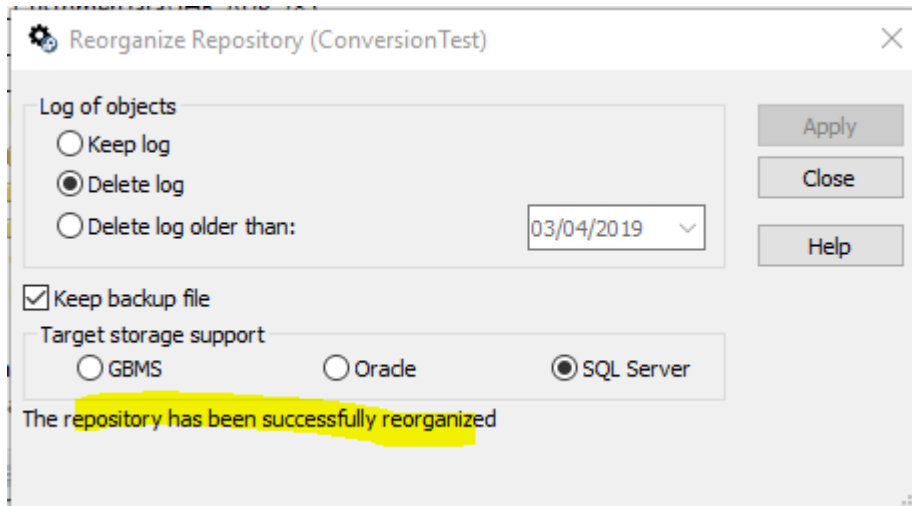
Process starts. Note that it goes through 2 mains steps:

- Logical backup
- Import

Wait until process ends. The following windows should be displayed.



Check message, ex 'The repository has been successfully reorganized'



Click button 'Close'

For more details, see only documentation, reorganization feature

HOPEX Administration: Administrator Guide : Managing Repositories : Managing Repositories :
Reorganizing an RDBMS Repository

4. CHECK DATA

The best proof is the fact that the process could run without error or that errors can be understood and accepted.

As a precautionary measure, you run a quick tour and check that upgraded data looks correct. Of course, this kind of check cannot be exhaustive, but it usually enables you to have a first feedback and confirm the migration process went well.

Example of scenario:

- Open a private workspace (ex-transaction).
- Browse through objects using query tools, navigation trees and diagrams.
- Perform insignificant updates (ex: change a character in a comment value, slightly move an object in a diagram...).
- Dispatch private workspace.

RDBMS Repository Installation Guide

HOPEX V3

Contents

Summary	3
Generalities	4
Supported Versions of RDBMS	4
Unsupported HOPEX Features in RDBMS Storage	4
Expected Advantages	4
Licensing	5
Infrastructure Requirements.....	6
RDBMS Client	6
Network Capability to Database Server	7
File Server and RDBMS local cache	7
Architecture Model	8
Database Server	8
HOPEX RDBMS Diagnostic Utility	10
Purpose	10
Running the RDBMS Diagnostic Utility	10
SQL Server support	16
SQL Server Requirements	16
Defining a HOPEX SQL Server Connection.....	18
Creating an Environment.....	21
Creating a Repository	23
HOPEX Private Workspaces Cleanup.....	25
HOPEX Historical Data Cleanup	26
Batching Cleanup procedures for SQL Server	29
Maintenance tasks	32
HOPEX RDBMS repositories specific administration actions	42
Migrating Your Data from One Storage Support to Another	42
Restoring a HOPEX environment from formatted data.....	45
Handling of HOPEX RDBMS repositories specific internal format	50
Vocabulary	53
Appendix - FAQs.....	55

Summary

This technical article describes the procedures and best practices for deploying the HOPEX application on a relational database server (SQL Server).

This deployment applies to **HOPEX V3**.

Generalities

Supported Versions of RDBMS

MEGA has qualified some versions of RDBMS for HOPEX V3. Those versions can be found in the following documents (depending on your type of deployment):

- Web Front-End Architecture Overview HOPEX V3 EN
- Windows Front-End Architecture Overview HOPEX V3 EN

Unsupported HOPEX Features in RDBMS Storage

When a HOPEX repository is stored on an RDBMS, HOPEX does not support the following features:

- MySQL RDBMS
- Oracle RDBMS
- Offline mode
- Repository protection
- Mixed environments
 - MEGA proprietary format (GBMS storage format) repository and repositories stored on an RDBMS. For example, a GBMS environment (SystemDb) and one or more repositories stored on SQL Server. The opposite is also not supported (SQL Server environment with GBMS repositories within).

Expected Advantages

The advantages expected from an RDBMS deployment are:

- Compliance with company-wide IT standards.
- Guarantee of scalability and security.
- Quicker dispatch time. In particular with “big” HOPEX private workspaces (HOPEX private workspaces with many creations/deletions/updates).

With this type of architecture, HOPEX supports global deployment on the same repository. In particular, it enables bypassing some limits related to the GBMS storage format.

- Maximum limit of 510 concurrent private workspaces per environment. No limit is identified in the HOPEX application for SQL Server storage format.
- Maximum limit of 24 GB of data per HOPEX repository. No limit is identified in the HOPEX application for SQL Server storage format.

With the RDBMS storage format, the HOPEX environment contains unshared files. All the data accessed during the execution of the HOPEX application is stored in the RDBMS. The RDBMS guarantees scalability and security.

Licensing

The “HOPEX repository storage (SQL Server)” product is required on the license to gain access to the RDBMS storage feature. The license can be dedicated to the workstation or shared by a group of users. All users connecting to HOPEX must have access to this license as well as to other products (HOPEX IT Architecture...).

Infrastructure Requirements

RDBMS Client

An RDBMS Client is necessary on each workstation that uses HOPEX with data stored on an RDBMS.

- **SQL Server**

Installation of Microsoft SQL Server 2012 Native Client is required. This client is compatible with the 2008, 2012, 2014, 2016, and 2017 versions of SQL Server. See corresponding Microsoft articles for more details:

<https://www.microsoft.com/en-us/download/details.aspx?id=50402>

The SQL Server 2012 Native Client installation program is available in a subfolder of the HOPEX installation, for:

- 64bits Windows operating systems:
Under < HOPEX installation>\Install\RDBMS client\Sqlserver\x64\sqlncli.msi
- 32bits Windows operating systems:
Under <HOPEX installation>\Install\RDBMS client\Sqlserver\x86\sqlncli.msi

Network Capability to Database Server

On a client computer running HOPEX, it is recommended to ping the RDBMS server with a filled buffer to have an evaluation of the infrastructure. To do this, download the **hrPING** freeware tool available at <https://www.cfos.de/en/ping/ping.htm>. To use this tool, you must first accept the terms of the licence. Use it with the following command in a command window from a computer that will be running HOPEX:

```
hrping.exe -W -l 5000 -n 50 -y <RDBMS Server name or IP>
```

Example for this command output:

```
Statistics for <RDBMS Server name or IP>:  
Packets: sent=50, rcvd=49, error=0, lost=1 (2% loss) in 24.500562 sec  
RTTs of replies in ms: min/avg/max/dev: 0.338 / 0.535 / 0.637 / 0.048  
Bandwidth in kb/sec: sent=10.260, rcvd=10.055
```

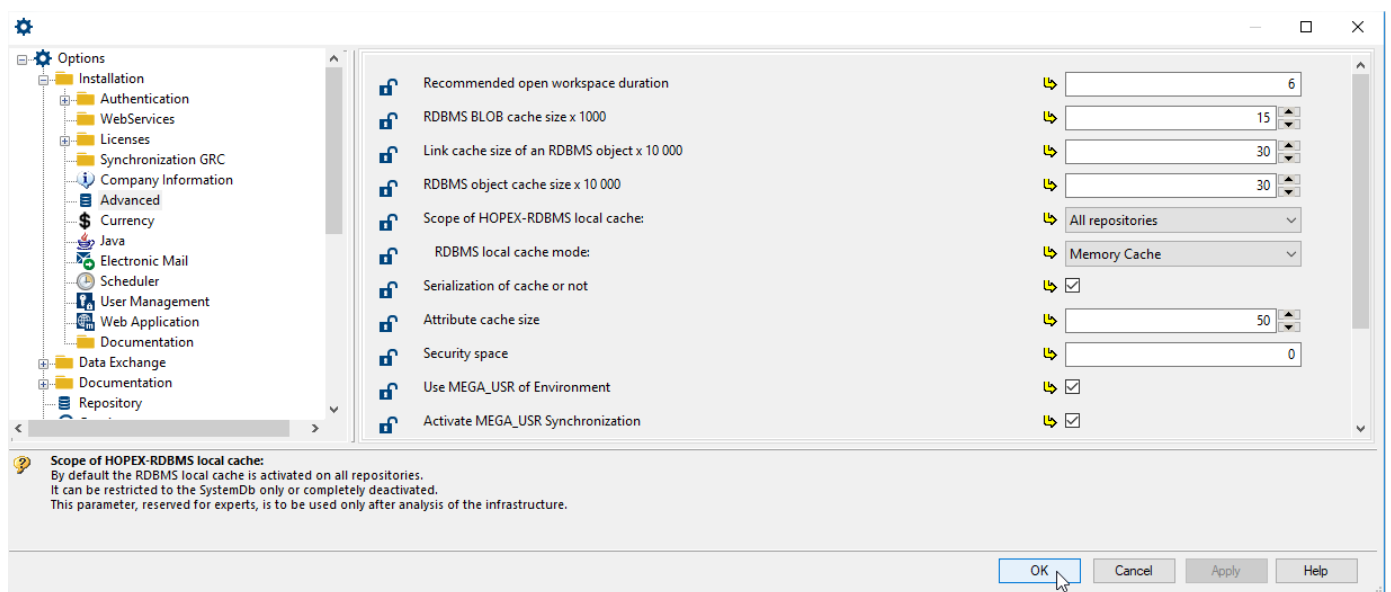
If the value returned for "RTTs of replies in ms/avg" (0.535 in the example) is higher than 1 ms, contact MEGA. See hrPING help for details on this command.

File Server and RDBMS local cache

See Deployment guide for generality.

For RDBMS deployments, there is a specific data caching option that is **enabled by default**. The purpose of this option is to improve the application response time by saving the environment data to the RAM of the server, so that fewer roundtrips to the database server are necessary. This cache is filled as the data is accessed during the use of HOPEX. The data cached that is out of date is deleted from that cache during dispatches.

The "Activate RDBMS local cache" is both accessible at the site level, or at the environment level.



To modify the RDBMS local cache globally for all environments:

1. Start Administration.exe.
2. At the top of the tree, right-click **HOPEX** and select **Option > Modify**.
3. Expand the **Installation** folder.
4. Select **Advanced**.
5. In the right pane, for the **Scope of HOPEX-RDBMS local cache** option, use the drop-down list to modify the range of that cache:
 - "SystemDb only".
 - "Disabled": if you do not want to use it, for example if your RDBMS instance is located on the same server as your application.
 - "All repositories" (default value): if for example you have a network latency with your RDBMS server.

Architecture Model

All the architecture models described in the "Web Front-End Architecture Overview HOPEX V3" document can use the RDBMS storage.

Database Server

The following sections will help your database administrator (DBA) size the Database server according to the profiles and the number of HOPEX users you plan to use.

Server disk size

Each new object takes up 30 KB on a disk (object with its attributes and links).

If you activate the HOPEX Repository Log file each action on the HOPEX repository creates an object.

You should reserve 5GB on the server disk.

Reminder:

HOPEX will stop working if the datafile is full. To avoid this, the databases can be created with the autoextend property activated. If this is not possible, the datafiles growth must be monitored carefully in order to provide more space if fullness is about to be reached.

Number of connections opened by HOPEX on the RDBMS for each HOPEX workstation

This information will help you define the amount of memory (RAM) required for the database instance used to run HOPEX on the database server

- **SQL Server**

One connection is used for each RDBMS storage. It means that, when a HOPEX User is connected to HOPEX, two connections to SQL Server are open (one for the SystemDb and one for the User repository).

An additional connection is used for each RDBMS storage when you use the HOPEX locks.

Each opened connection uses 24 KB of memory on the SQL Server.

HOPEX RDBMS Diagnostic Utility

Purpose

HOPEX provides a Java based utility that should be used before starting to use environments and repositories on an RDBMS. This utility runs several tests for which the results will be compared to some memorized values corresponding to a situation where HOPEX is likely to have close-to-optimum performances.

The **RDBMS Diagnostic** utility is stored at this path:

```
< HOPEX Installation Path>\Utilities\RDBMS Diagnostic\
```

Running the RDBMS Diagnostic Utility

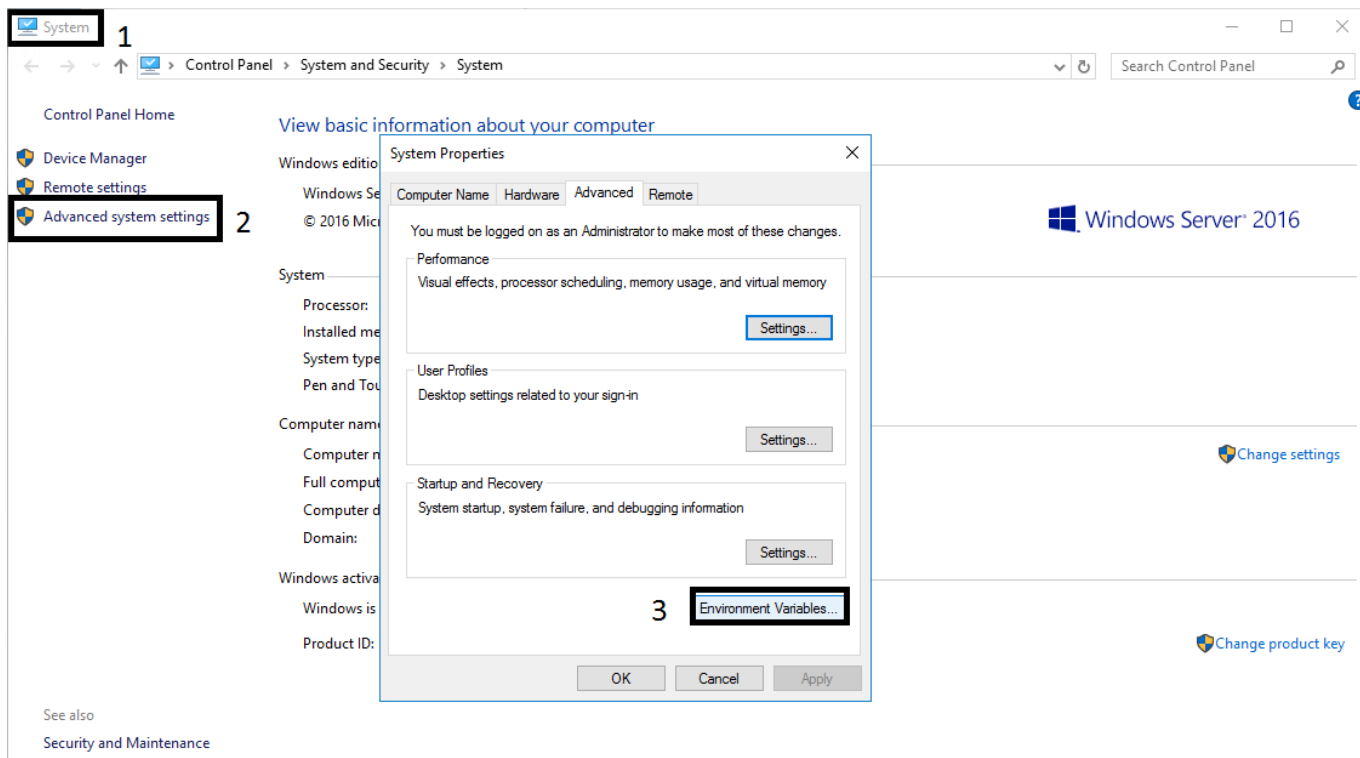
A batch file was created to run the tool.

To run the RDBMS Diagnostic Utility:

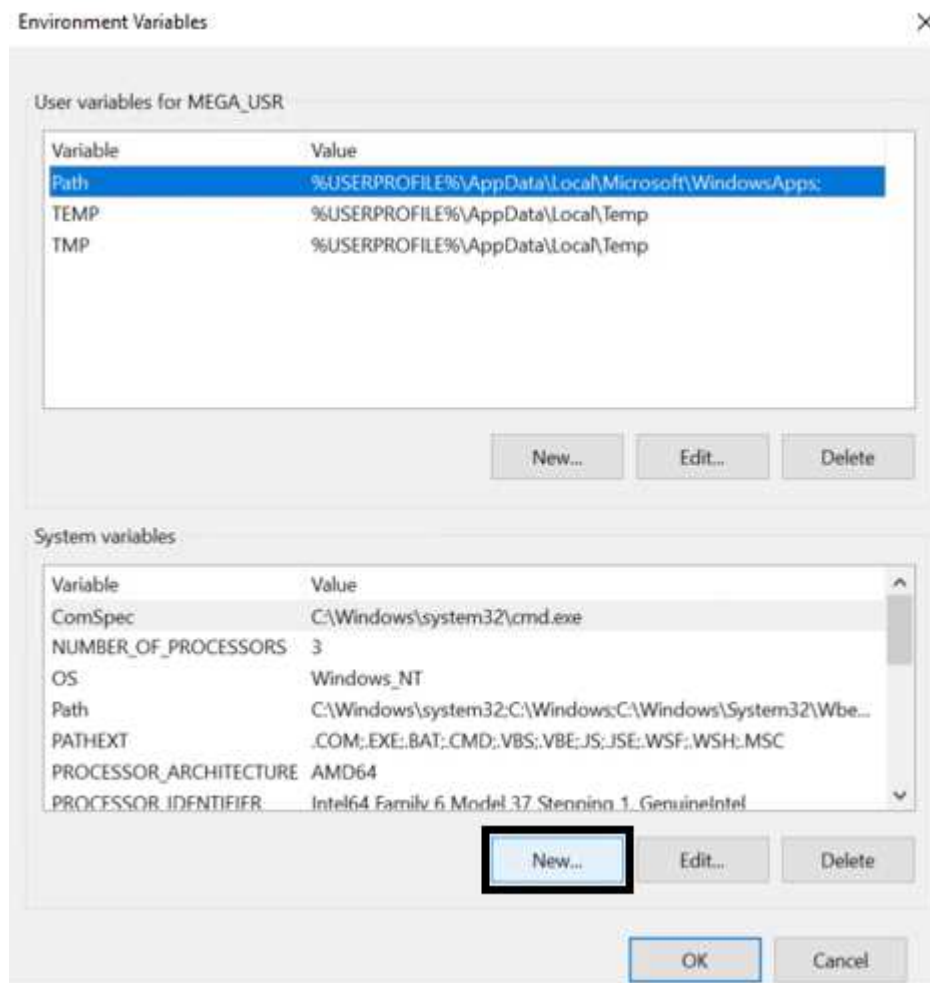
1. Extract the content of the “PerfTestExe.zip” compressed file in the same location:

This PC > Local Disk (C:) > Program Files (x86) > MEGA > HOPEX V3 > Utilities > RDBMS Diagnostic			
Name	Date modified	Type	
Files	2/25/2020 10:30 AM	File folder	
MySQL	2/25/2020 10:30 AM	File folder	
Oracle	2/25/2020 10:30 AM	File folder	
SqlServer	2/25/2020 10:30 AM	File folder	
PerfTestExe.zip	2/15/2020 8:48 PM	Compressed (zipped) Folder	
PerfTestExe	5/7/2020 7:41 PM	File folder	

2. You need to set the JAVA_HOME, so that you can run the batch that will launch the proper JAR file. One way of doing it, is to open the **Environment Variables** of the server:

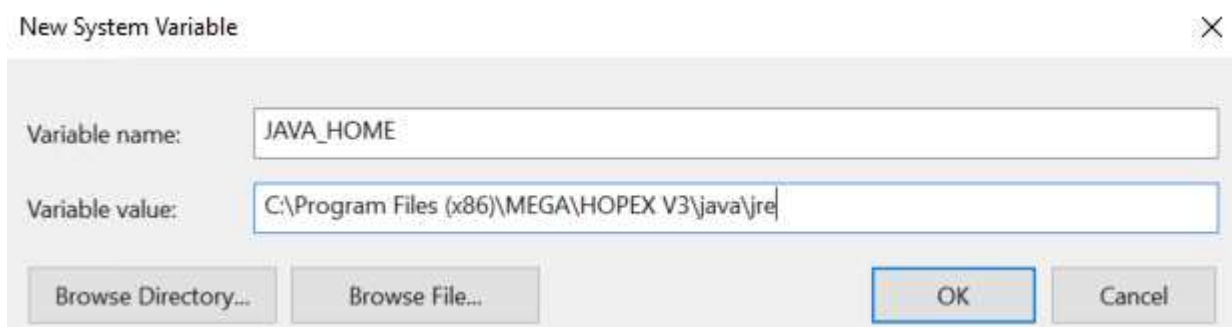


3. In the **System variables** pane, click **New**.

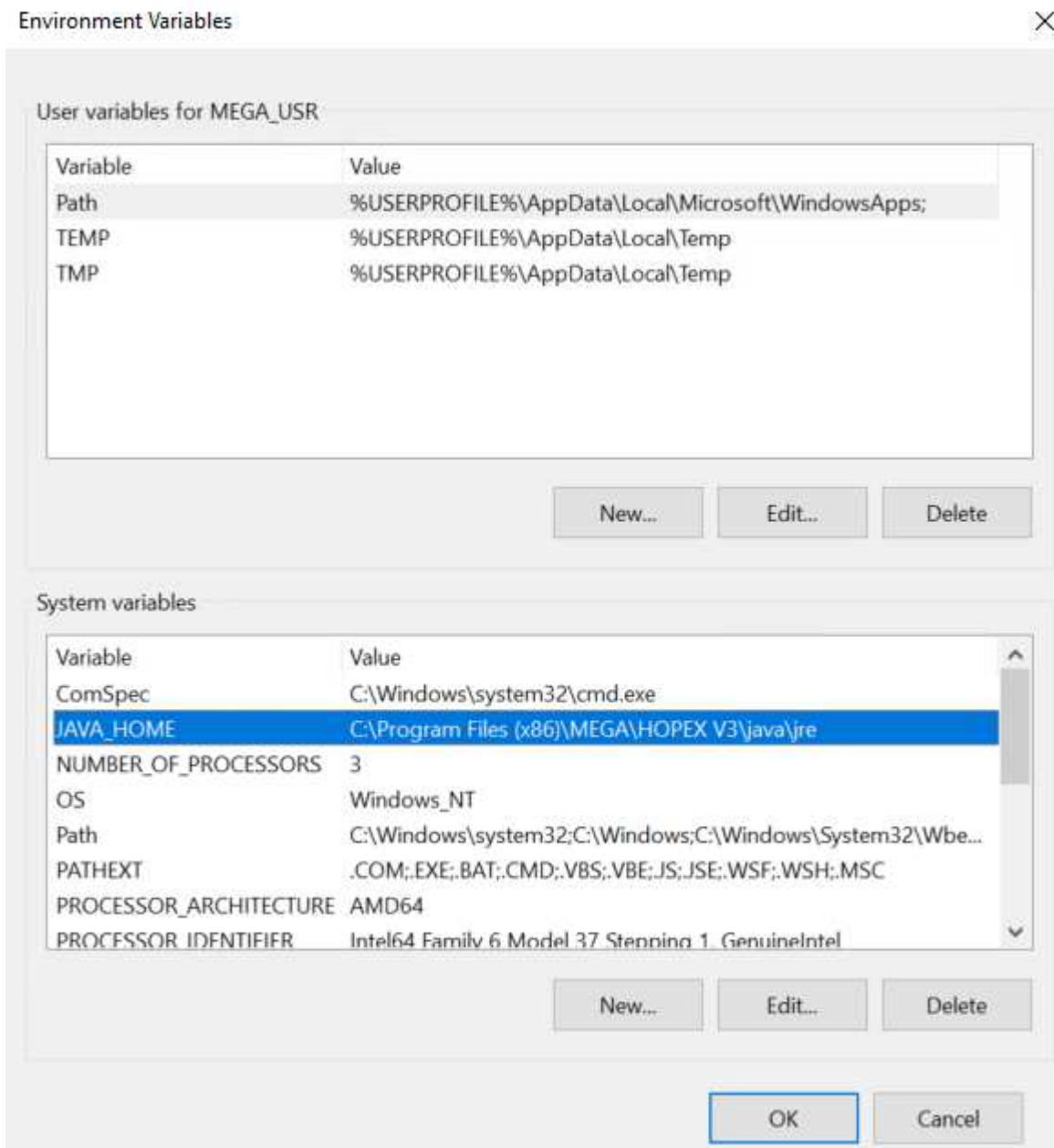


4. Create the variable "JAVA_HOME" with the value targeting the "<Hopex installation>\java\jre" folder.

For example: ``C:\Program Files (x86)\MEGA\HOPEX V3\java\jre``

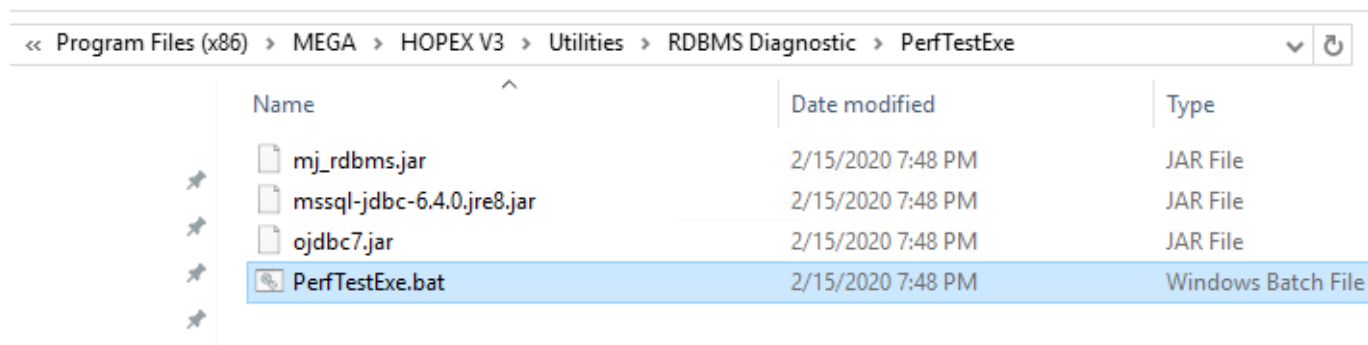


5. Click **OK**

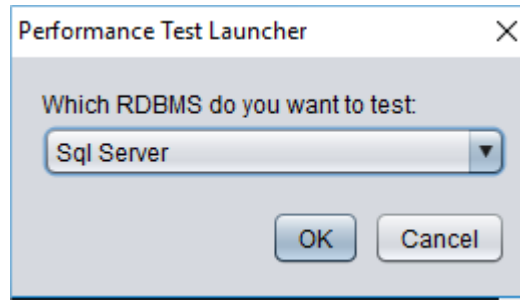


The key is created.

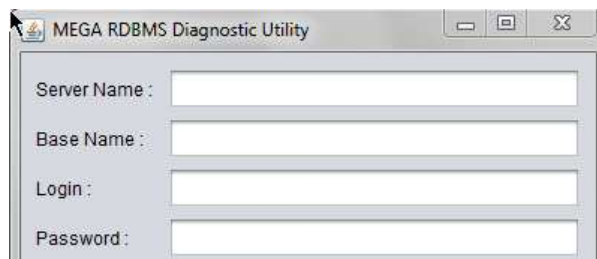
- Click **OK** to close the **Environment Variables** window.
- Go back to your file explorer, into the **PerfTestExe** folder, and execute the **PerfTestExe.bat**.



8. Select the RDBMS type (SQL Server):



9. Enter the connection information to the RDBMS storage that is the target for hosting the HOPEX data:
 - o a database name for SQL Server



10. In order to have consistent times, the **Expected Execution Time** values were recorded after running the utility more than once and noticing that the values were stable.
So to have results that can be considered valid, run the utility twice and consider the values of the 2nd run.

Here is an example of some test results:

Test Name	Execution Time (ms)	Expected Execution Tim...	Test Result
✓ DDL	13	20	Ok
✓ INSERT (LIGHT)	29402	29000	Ok
✓ INSERT (LIGHT, server I...	4532	4300	Ok
✓ INSERT (HEAVY)	10627	14000	Ok
✓ READ (LIGHT)	7925	9000	Ok
✓ READ (HEAVY)	30672	34000	Ok
✓ SERVER CPU SPEED	6084	7500	Ok
✓ SERVER DISK	20082	20000	Ok
✓ SERVER DISK (BLOB's)	19955	20000	Ok
✓ BANDWIDTH	23500	24000	Ok
✓ BANDWIDTH (BLOB's)	23418	40000	Ok
✓ RESET DB	23	100	Ok

Test Description :

By clicking on a test, you will have a short description of it.

Diagnostic :

OK: time=23500ms , expected time=24000ms
 TEST 11 (BANDWIDTH (BLOB's)):
 OK: time=23418ms , expected time=40000ms
 TEST 12 (RESET DB):
 OK: time=23ms , expected time=100ms
 ##### Batch Test Finished: Tue Mar 19 12:02:14 CET 2013 #####

☐ Auto Commit

Copy Diagnostic to Clipboard

Start Tests Stop Tests Close

SQL Server support

SQL Server Requirements

Encoding

After the database has been created, verify that "Collation" is set to "SQL_Latin1_General_CP1_CS_AS". If the database is created from the HOPEX application, the appropriate encoding is automatically configured.

User management

When the HOPEX application accesses the HOPEX data stored in the RDBMS, it uses an SQL connection string. This connection string refers to a user account that has certain privileges for the instance.

This user can either be a native account, or a Windows account :

- **Native account:**
 - **Pros:** Unique account, configured for everyone that runs the thin or thick clients.
 - **Cons:** Thought to be less secure.
- **Windows accounts:**
 - **Pros:** Don't set up any connection string in the tool.
 - **Cons:** Need to authorize several Windows accounts to have direct access to the data : the impersonate user that runs the processes of the web users, the service account that runs the SSP, every user that needs to run the thick client (either the Administration.exe or the Hopex.exe tools).

Privileges for native account

You can have several kinds of SQL server users in relation to the customer security policy:

- **Standard security policy:** the user account is enabled to manage databases. This is the easiest solution especially if the SQL Server instance is dedicated to HOPEX.

User type	Comment	Server roles	Database roles	Server permissions
User with maximum privileges	Allowed to manage any database (create database, delete database, data read access, data write access, update database structure)	dbcreator	db_owner (1)	View server state (2)

- **Advanced security policy:** only the DBA is allowed to create new databases following specific naming rules. A user is required to use the existing databases.

User type	Comment	Server roles	Database roles	Server permissions
User with limited privileges	Allowed to use an existing database (data read access, data write access, update database structure)	public	db_owner (3)	View server state (2)

(1) db_owner role is automatically assigned by the system when a database is created.

(2) To consult the view 'sys.dm_exec_sessions' for the server.

(3) db_owner role is manually assigned by the DBA after database creation.

The HOPEX application will create table, columns and index objects dynamically. The right to create Procedures is mandatory. Trigger, functions and view objects are not used.

Privileges for Windows accounts

Since this configuration requires to grant access to the different databases to several Windows accounts, and especially to accounts of people running the thick client of the application, it is recommended to limit those rights to a minimum, to reduce the risk of harming the application by directly modifying or deleting data.

- **Advanced security policy:** only the DBA is allowed to create new databases following specific naming rules. A user is required to use the existing databases.

User type	Comment	Server roles	Database roles	Server permissions
User with limited privileges	Allowed to use an existing database (data read access, data write access, update database structure)	public	db_ddladmin, db_datawriter and db_datareader (3)	View server state (2)

(2) To consult the view 'sys.dm_exec_sessions' for the server.

(3) those roles are manually assigned by the DBA after database creation.

The HOPEX application will create table, columns and index objects dynamically. The right to create Procedures is mandatory. Trigger, functions and view objects are not used.

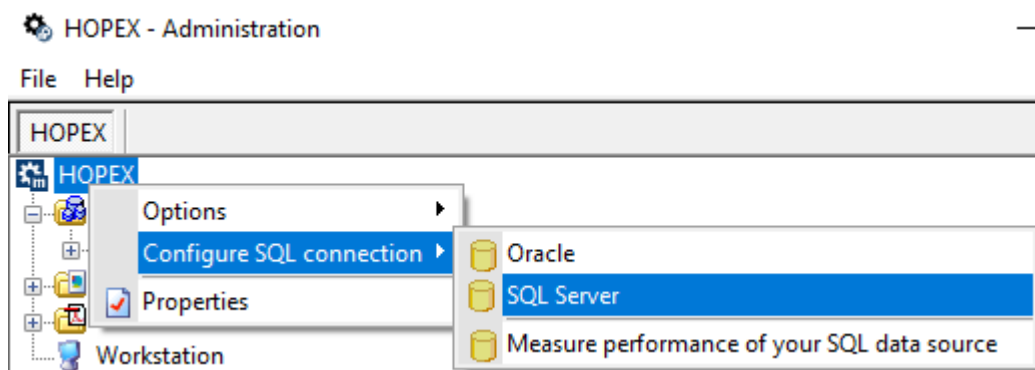
The Windows users **should not** have the "db_creator" server role.


Defining a HOPEX SQL Server Connection

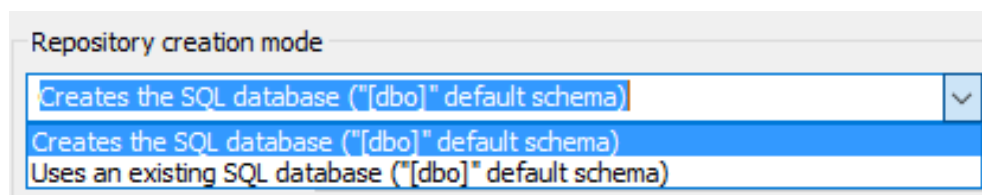
A **Configure SQL Connection** menu is available in the HOPEX Administration application at different levels (site, environment, and repository) if the license contains the Repository Storage (SQL Server) product.

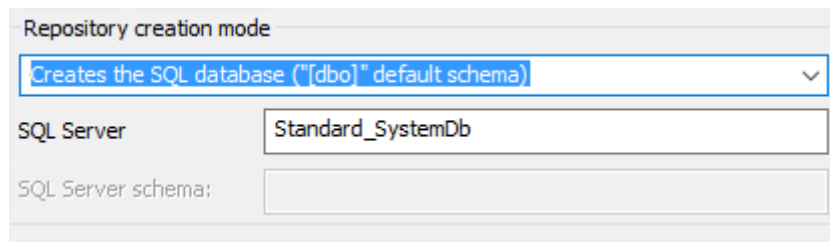
Procedure with a native SQL account

1. Start HOPEX **Administration.exe**.
2. Right-click HOPEX (the root of the administration tree) and select **Configure SQL connection > SQL Server**.

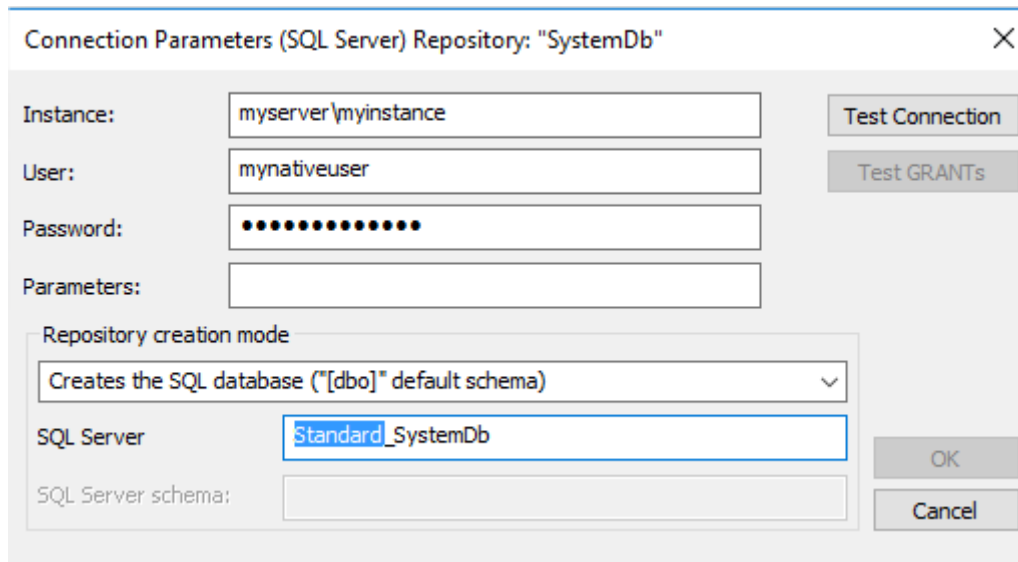


3. Enter the connection parameters.
 - o **Instance:** <machine network name>\<SQL Server instance name> (1)
Example for a standalone installation with SQL Express: MyMachine\SQLEXPRESS
 - o **User:** user enabled to access/update SQL Server
 - o **Password:** password of the user enabled to access/update SQL Server
 - o  **Warning:** Ensure this password is consistent with MS SQL rules, see MS related documentation.
 - o **Repository creation mode:** by default you create your environment in the "dbo" schema, in the database prefixed with the environment name (here, "Standard"). You can choose to target a different database using the same schema "dbo", using the dropdown list.



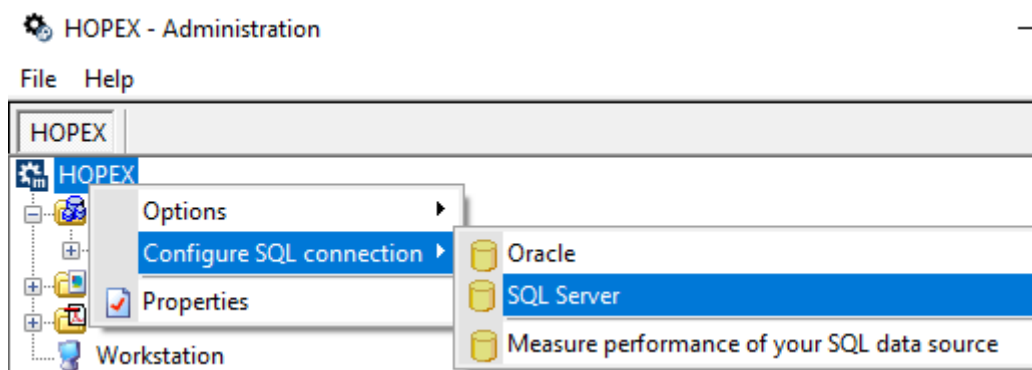


4. Click **Connection Test** to check the connection parameters.



Procedure when using Windows authentication

1. Start HOPEX **Administration.exe**.
2. Right-click HOPEX (the root of the administration tree) and select **Configure SQL connection > SQL Server**.



3. Set the connection parameters.
 - o **Instance:** <machine network name>\<SQL Server instance name> (1)
Example for a standalone installation with SQL Express: MyMachine\SQLEXPRESS
 - o **User:** leave blank
 - o **Password:** leave blank
 - o **Parameters :** set "Trusted_Connection=Yes;"

4. Click **Connection Test** to check the connection parameters.

Connection Parameters (SQL Server) ✕

Instance :	<input type="text" value="myserver\myinstance"/>	<input type="button" value="Test Connection"/>
User:	<input type="text"/>	<input type="button" value="Test GRANTS"/>
Password:	<input type="password"/>	
Parameters:	<input type="text" value="Trusted_Connection=Yes;"/>	

Creating an Environment

The environment creation mainly consists in creating a SystemDb repository. For SQL server, two creation modes are available from HOPEX:

- Create a new database on the SQL Server (standard security policy)
- Use an existing database of the SQL Server (advanced security policy)

Creating a new SystemDb database

Prerequisite:

- Identify the SQL connection parameters (RDBMS instance, user, password)
- Identify the location of the environment folder on the file server

Procedure:

1. Start HOPEX **Administration.exe**.
2. Right-click the **Environments** folder and select **New**.
3. Enter the environment **Name**.

This creates a folder on the file server.

4. (If needed) Change the **Location**.
5. Select "SQL server" Repository Storage Support.
6. Click **OK**.
7. Confirm or change SQL Connection parameters.
8. As the **Repository Creation Mode** select "Create Database" .
9. Click **Test Connection** to check that the SQL Server is reachable. This step must be successful for the process to continue.
10. Click **Test GRANTS** to check different actions (table creations, indexing columns etc.) that are necessary for HOPEX to be able to work. This step must be also successful for the process to continue.
11. Click **OK** to start the environment creation.

Result:

- A SystemDb repository stored in the selected RDBMS instance is created.
- A folder (HOPEX environment folder) is created at the selected location. This folder contains several files and subfolders (Db, Mega_Usr, SysDb).

Using an existing SystemDb database

Prerequisite:

- Identify the SQL connection parameters (RDBMS instance, user, and password).
- Identify the location of the environment folder on the file server.
- **Verify that the “Collation” property of the database is set to “SQL_Latin1_General_CP1_CS_AS”.**
- Identify the exact name of the user database in the SQL Server. It follows this naming rule:

`<EnvironmentName>_SystemDb`

Example: `MyEnvironment_SystemDb`

Note: the environment name must match the environment folder.

Procedure:

1. Start HOPEX **Administration.exe**.
2. Right-click the **Environments** folder and select **New**.
3. Enter the environment “Name” (in this example : “Name” = “MyEnvironment”) This creates a folder.
4. (If needed) Modify the **Location**.
5. Select “SQL server” Repository Storage Support.
6. Click **OK**.
7. Confirm or change the SQL Connection parameters.
8. As **Repository Creation Mode** select “Uses an existing SQL database (“[dbo]” default schema)”.
9. Click **Test connection** to check that the SQL Server is reachable.

This step must be successful for the process to continue. If “Use existing database” option was specified, this test tries to connect to the database matching the following pattern: “MyEnvironment_SystemDb”. This test must be successful for the process to continue.

10. Click **Test Grants** to check different actions (tables creations, indexing columns etc.) that are necessary for HOPEX to be able to work. This test must be also successful for the process to continue.
11. Click **OK** to start the environment creation.

Result:

- The SystemDb repository is initialized.
- A folder (HOPEX environment folder) is created at the selected location. This folder contains several files and subfolders (Db, Mega_Usr, SysDb).

Creating a Repository

For SQL Server, two creation modes are available from HOPEX:

- Create a new database on the SQL Server (standard security policy).
- Use an existing database of the SQL Server (advanced security policy).

Creating a new SQL Server database

Prerequisites:

- Identify the SQL connection parameters (RDBMS instance, user, and password).

Procedure:

1. Start HOPEX **Administration.exe**.
2. Connect to the environment concerned.
3. Right-click the **Repositories** folder and select **New**.
4. Enter the repository **Name**.
5. Keep the default **Location**.
6. Select "SQL server" Repository Storage Support.
7. Click **OK**.
8. Confirm or change the SQL Connection parameters.
9. As **Repository creation mode** select "Creates the SQL database ("[dbo]" default schema)".
10. Click **Test connection**. The test must be successful for the process to continue.
11. Click **Test GRANTS**. The test must be successful for the process to continue.
12. Click **OK** to create the new database

Result:

- A repository is created in SQL server. It follows this naming rule:

`<EnvironmentName>_<RepositoryName>`

Example: `MyEnvironment_SQLServerRepository`

- A folder is created in the specified location.
This folder contains an EMV and an EMQ file.

Using an existing SQL Server database

Prerequisites:

- Identify the SQL connection parameters (RDBMS instance, user, and password).
- **Verify that the property 'Collation' of the database is set to 'SQL_Latin1_General_CP1_CS_AS'**
- Identify the exact name of the user database in the SQL Server. It follows this naming rule:

`<EnvironmentName>_<RepositoryName>`

Example: `MyEnvironment_SQLServerRepository`

Note that the environment name must match the actual environment folder.

Procedure:

1. Start HOPEX **Administration.exe**.
2. Connect to the environment concerned.
3. Right-click the **Repositories** folder and select **New**.
4. Enter the environment **Name**.
E.g.: `SQLServerRepository`
5. Select **SQL server** Repository Storage Support.
6. Click **OK**.
7. Confirm or change the SQL Connection parameters.
8. As **Repository Creation Mode** select “Uses an existing SQL database (“[dbo]” default schema)”.
9. Click **Test** to check that the login can be performed and that the database exists.
10. Click **Test connection**. The test must be successful for the process to continue.
11. Click **Test GRANTS**. The test must be successful for the process to continue.
12. Click **OK**.

Result:

- A repository is referenced in the SQL server and initialized.

Example: `MyEnvironment_SQLServerRepository`

- A folder is created in the specified location.

`<this folder contains a .EMV and a .EMQ file.`

HOPEX Private Workspaces Cleanup

This procedure is used to delete the data of terminated private workspaces of HOPEX Users. It is necessary to clean up these data often in order to reduce database growth and preserve good performances. We recommend running this procedure every week if you have less than 10 users and every night if you have more than 10 users.

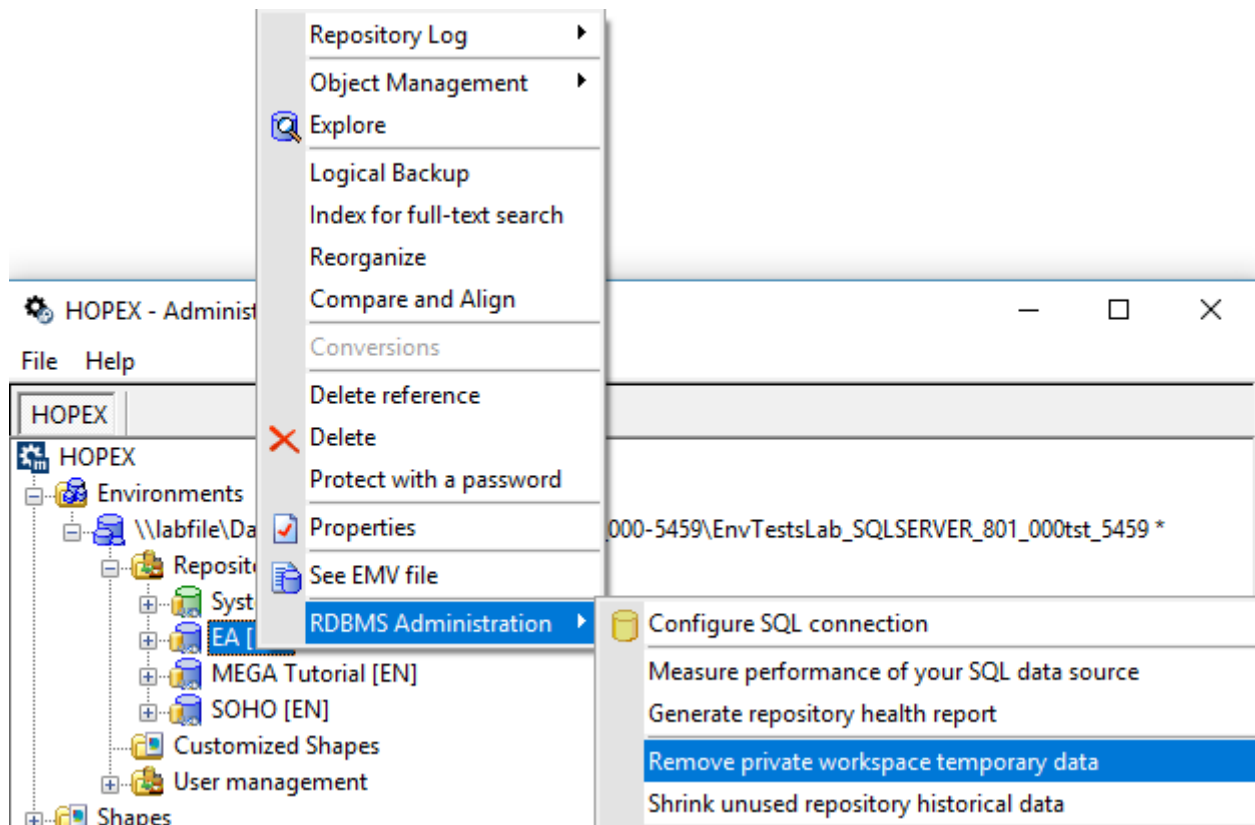
Installing the procedure

Warning: You must repeat this procedure for each HOPEX Repository and the SystemDb.

1. Right-click your HOPEX repository and select **RDBMS Administration > Remove private workspace temporary data**.

This will launch SP_CLEAN_MEGA_DATABASE and if the procedure:

- does not exist, the application will create it.
- already exists, it is overwritten by this action.



HOPEX Historical Data Cleanup

This procedure is used to delete the historical data of the HOPEX repository. Each time a HOPEX object is updated, the previous data is kept in database. That method insures a high data security even when connection to SGBD is interrupted. It is necessary to clean up these data often in order to reduce database growth and preserve good performances. This clean-up will have no impact on the repository logfile. We recommend running this procedure every week if you have less than 10 users and every night if you have more than 10 users.

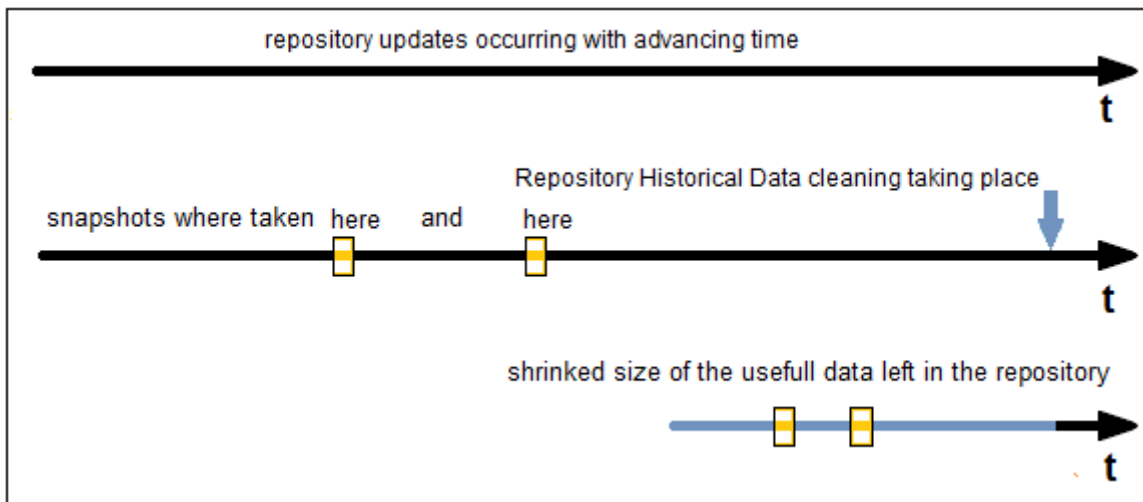
Before cleaning Historical Data

Historical data are used in the Repository Snapshot mechanism. See HOPEX Common Features > Other Features > Using Repository Snapshots: **Repository Snapshot Prerequisites** section for more details.

If you need to have Repository Snapshots taken, be aware that it will not be possible anymore for the period of time covered by the cleanings. In other words, if you need Repository Snapshots, be sure to take them before the procedure runs.



In this first illustrated case, all archived states were deleted, so all the space that these archived states were using is reclaimed physically (an actual delete in the tables was issued for every one of them).



In this second example, all archived states were also deleted except those corresponding to the state of the repository when the 2 Snapshots were taken.

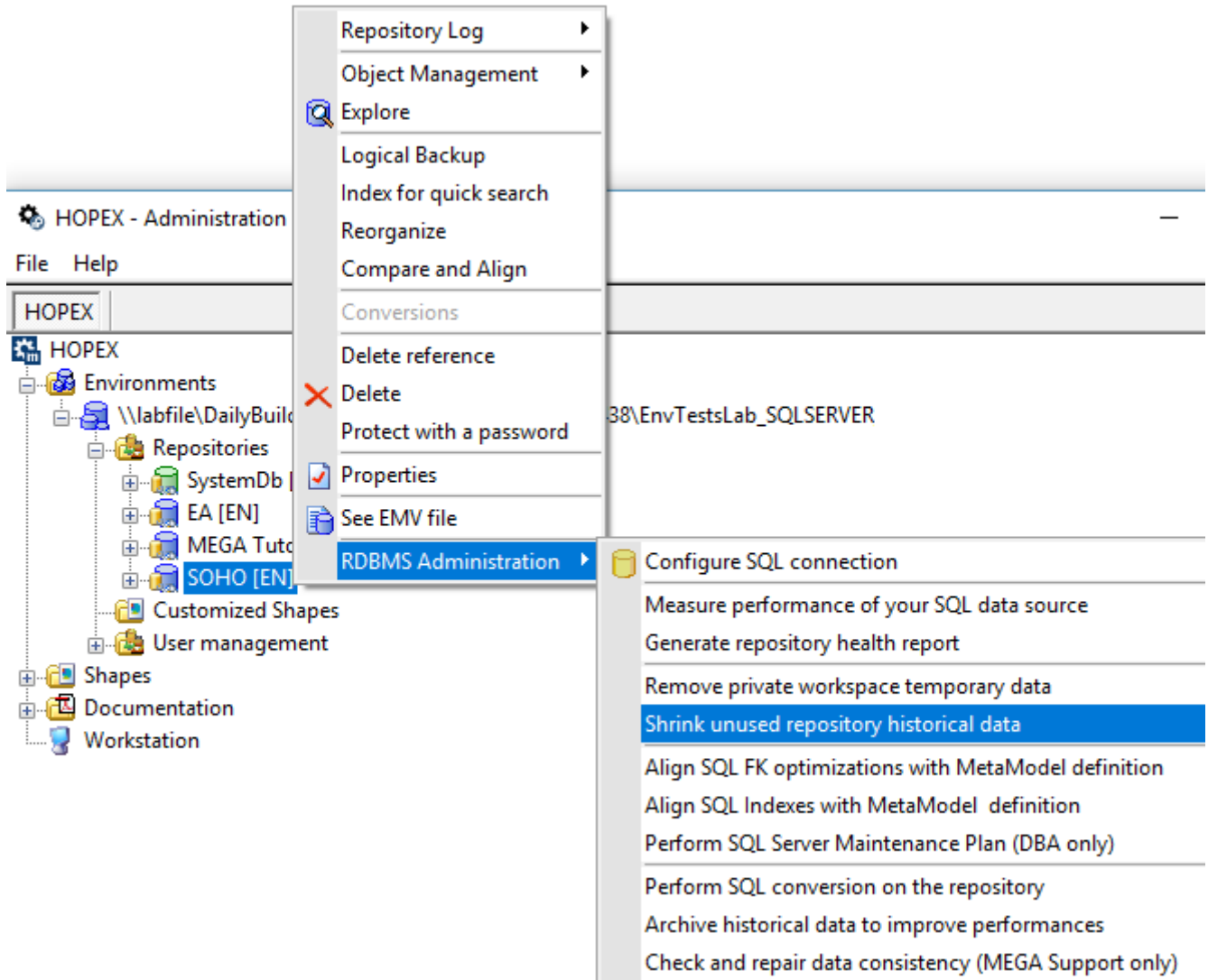
In this case, the data corresponding to the repository state for the Snapshot(s) is saved and it is thanks to this saving that special features will be available within this repository regarding this data.

Installing the procedure

Warning : You must repeat this procedure for each HOPEX Repository and the SystemDb.

1. Right-click your **HOPEX repository** and select **RDBMS Administration > Shrink unused repository historical data**.

This launches SP_CONSOLIDATE_MEGA_DATABASE and if the procedure does not exist, the application creates it. If the procedure already exists, it is overwritten by this action.



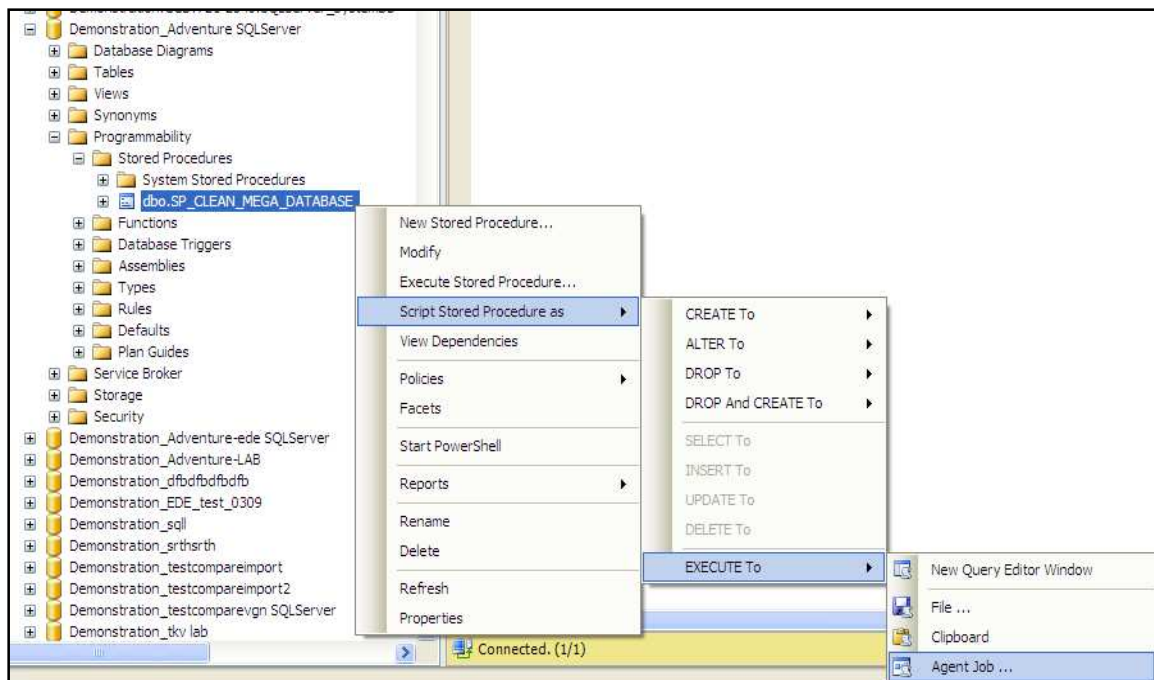
Batching Cleanup procedures for SQL Server

It is very important to run the two procedures on a regular basis. So If you do not want to have to remember to click on the corresponding menus in the Administration.exe program every time that each of the procedure should run, you can batch it using SQL Server agent job.

1. Using **SQL Server Management Studio**, find the SQL Server database that corresponds to the HOPEX repository for which you want to batch the stored procedure.

Reminder : the database will be named following this rule <EnvironmentName_RepositoryName>.

2. In **Programmability > Stored Procedures** folder, right-click this procedure and select **Script Stored Procedure as > Execute to > Agent job**.



Enter a name for the job and the schedule.

Job Schedule

Job name: Clean Database

Schedule name: Every Night

Schedule type: One time

One-time occurrence

Date: 10/21/2009 Time: 12:00:00 AM

The one-time occurrence date and time must be greater than the current date and time.

Frequency

Occurs: Daily

Recurs every: 1 day(s)

Daily frequency

Occurs once at: 5:35:21 PM

Occurs every: 1 hour(s)

Starting at: 5:35:21 PM

Ending at: 5:35:21 PM

Duration

Start date: 10/21/2009

End date: 10/21/2009

No end date:

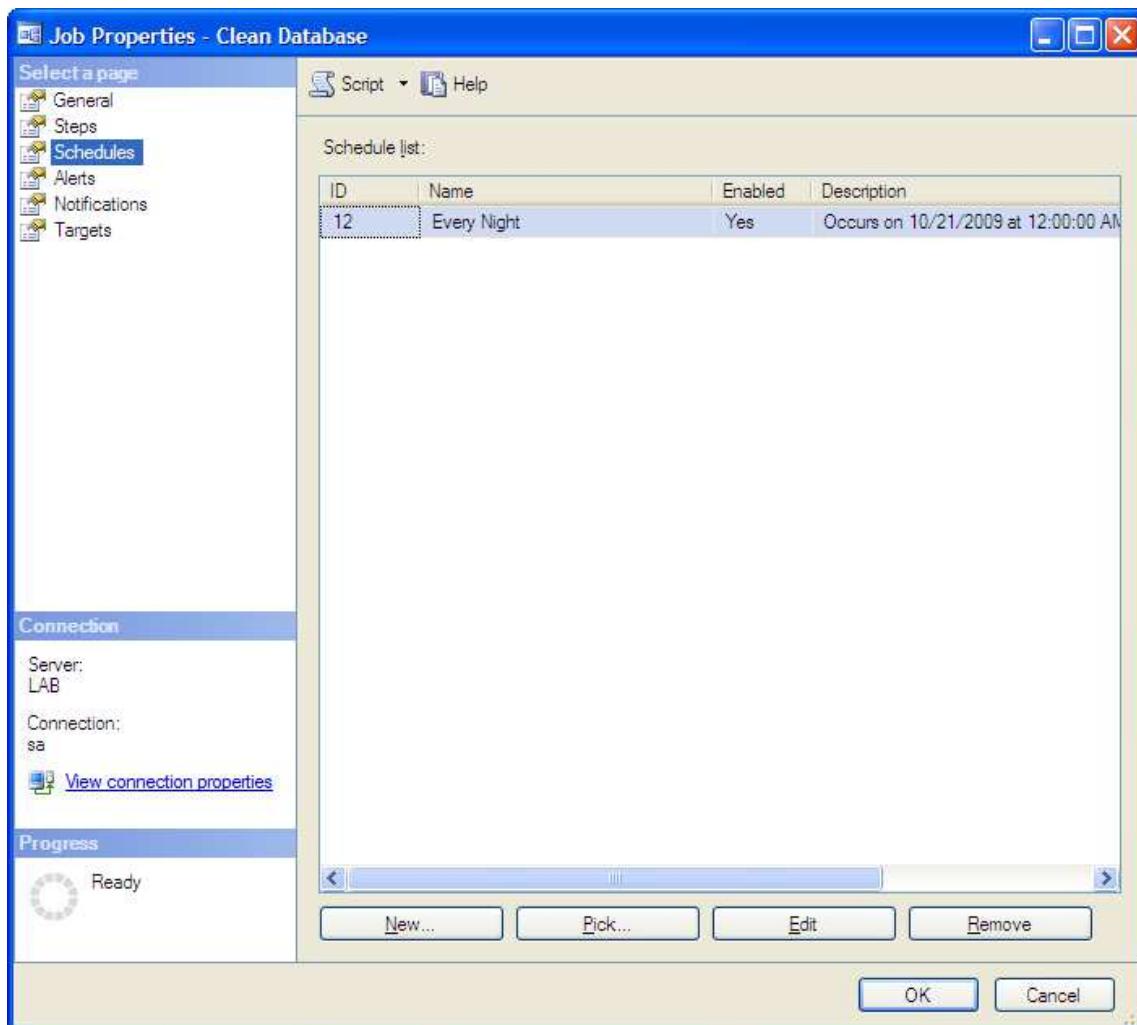
Summary

Description: Occurs on 10/21/2009 at 12:00:00 AM.

OK Cancel Help

The job is created.

3. Right-click this job and select **Properties**.
4. Select the **Schedules** tab and click **Edit**.



5. Set up the schedule to execute the job every night.

Job Schedule Properties - Every Night

Name:

Schedule type: ☒ Enabled

One-time occurrence

Date: Time:

Frequency

Occurs:

Rekurs every: day(s)

Daily frequency

☒ Occurs once at:

☐ Occurs every: hour(s)

Starting at:

Ending at:

Duration

Start date: ☐ End date:

☒ No end date:

Summary

Description:

Maintenance tasks

The SQL Server databases need to be maintained, in order to keep the best possible performances. Tasks such as "update of the statistics", "reorganize or rebuild of the indexes", "shrink of the databases", as well as backups, need to be run regularly.

We recommend set up the standard maintenance plans of SQL Server to manage those tasks. The backups can be excluded, if they are done through another channel.

Also, we can imagine to put the execution of the HOPEX cleanup procedures (see previous chapter) as the preliminary step to the SQL Server job that will run the maintenance tasks.

You can find below some screenshots of a default maintenance plan (with backups), with SQL Server 2012. It can be adapted to your version, and your rules :

1. Create a maintenance plan using the SQL Server wizard (in SQL Server Management Studio).
2. Give it a name and a schedule (click **Change**).

Maintenance Plan Wizard

Select Plan Properties
How do you want to schedule your maintenance tasks?

Name: Weekly Maintenance plan

Description:

Run as: SQL Server Agent service account

☐ Separate schedules for each task
☒ Single schedule for the entire plan or no schedule

Schedule: Occurs every week on Sunday at 12:00:00 AM. Schedule will be 1 [Change...](#)

[Help](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

3. Select the following maintenance tasks:

Maintenance Plan Wizard

Select Maintenance Tasks
Which tasks should this plan perform?

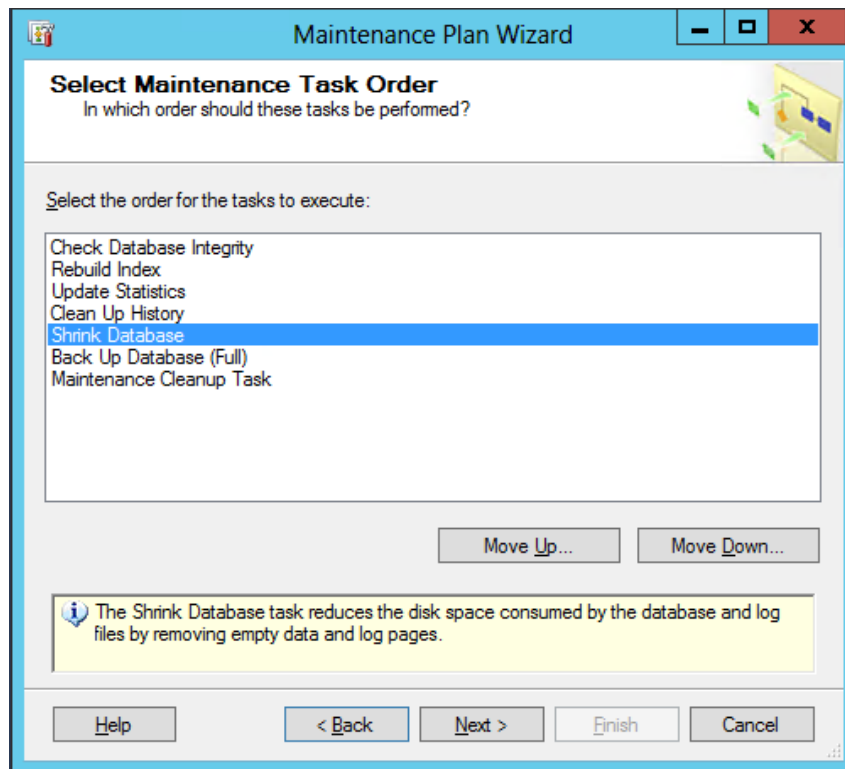
Select one or more maintenance tasks:

- ☒ Check Database Integrity
- ☒ Shrink Database
- ☐ Reorganize Index
- ☒ Rebuild Index
- ☒ Update Statistics
- ☒ Clean Up History
- ☐ Execute SQL Server Agent Job
- ☒ Back Up Database (Full)
- ☐ Back Up Database (Differential)
- ☐ Back Up Database (Transaction Log)
- ☒ Maintenance Cleanup Task

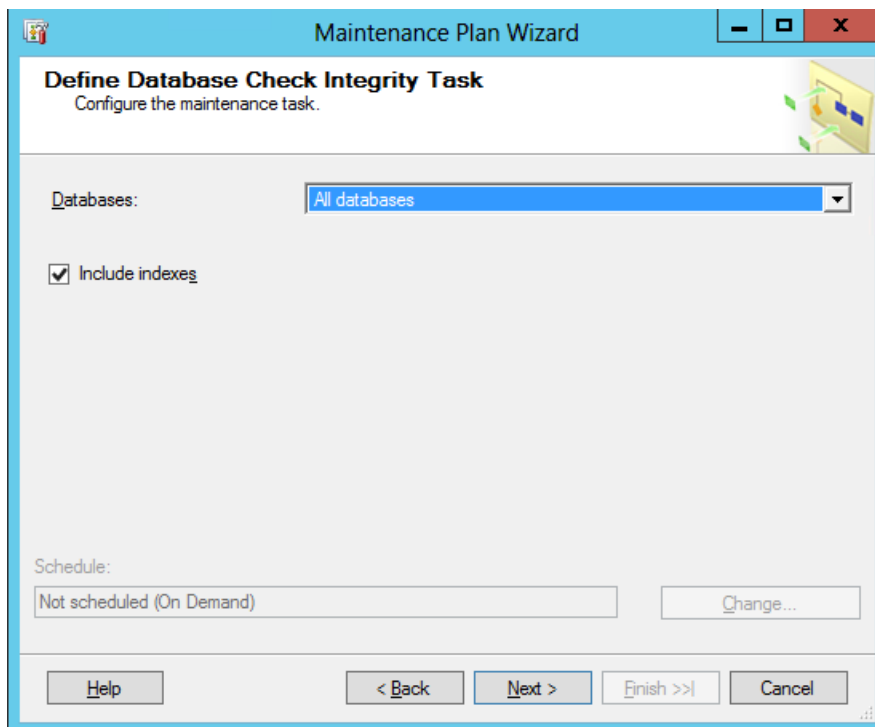
The Check Database Integrity task performs internal consistency checks of the data and index pages within the database.

[Help](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

4. Order the maintenance tasks as follows:



5. Check all databases (including the system databases):



6. Rebuild indexes for the user databases:

Maintenance Plan Wizard

Define Rebuild Index Task
Configure the maintenance task.

Databases: All user databases

Object:

Selection:

Free space options

☒ Default free space per page

☐ Change free space per page to: %

Advanced options

☐ Sort results in tempdb

☐ Keep index online while reindexing

For index types that do not support online index rebuilds

☒ Do not rebuild indexes

☐ Rebuild indexes offline

Schedule:

Not scheduled (On Demand) Change...

Help < Back Next > Finish >> Cancel

7. Same thing for the update of the statistics:

Maintenance Plan Wizard

Define Update Statistics Task
Configure the maintenance task.

Databases: All user databases

Object:

Selection:

Update:

☒ All existing statistics

☐ Column statistics only

☐ Index statistics only

Scan type:

☒ Full scan

☐ Sample by 50

Schedule:

Not scheduled (On Demand) Change...

Help < Back Next > Finish >> Cancel

8. Define how long the log files will be kept:

The screenshot shows the 'Define History Cleanup Task' dialog box within the 'Maintenance Plan Wizard'. The title bar reads 'Maintenance Plan Wizard'. The main heading is 'Define History Cleanup Task' with the subtitle 'Configure the maintenance task.' Below this, there is a section 'Select the historical data to delete:' with three checked checkboxes: 'Backup and restore history', 'SQL Server Agent job history', and 'Maintenance plan history'. Underneath is a section 'Remove historical data older than:' with a numeric input field set to '2' and a dropdown menu set to 'Week(s)'. At the bottom, there is a 'Schedule:' section with a text box containing 'Not scheduled (On Demand)' and a 'Change...' button. The bottom of the dialog features a row of buttons: 'Help', '< Back', 'Next >', 'Finish >>', and 'Cancel'.

9. Shrink all user databases, or at least the HOPEX databases:

The screenshot shows the 'Define Shrink Database Task' dialog box within the 'Maintenance Plan Wizard'. The title bar reads 'Maintenance Plan Wizard'. The main heading is 'Define Shrink Database Task' with the subtitle 'Configure the maintenance task.' Below this, there is a 'Databases:' section with a dropdown menu set to 'All user databases'. Underneath, there are two input fields: 'Shrink database when it grows beyond:' set to '50' MB, and 'Amount of free space to remain after shrink:' set to '10' %. Below these are two radio buttons: 'Retain freed space in database files' (unselected) and 'Return freed space to operating system' (selected). At the bottom, there is a 'Schedule:' section with a text box containing 'Not scheduled (On Demand)' and a 'Change...' button. The bottom of the dialog features a row of buttons: 'Help', '< Back', 'Next >', 'Finish >>', and 'Cancel'.

10. Backup all databases, choose the destination folder, and if you want to have subfolders for each database:

Define Back Up Database (Full) Task
Configure the maintenance task.

Backup type: Full

Database(s): All databases

Backup component

☒ Database

☐ Files and filegroups:

☐ Copy-only Backup

☐ For availability databases, ignore Replica Priority for Backup and Backup on Primary Settings

☐ Backup set will expire:

☒ After 14 days

☐ On 1/21/2015

Back up to: ☒ Disk ☐ Tape

☐ Back up databases across one or more files:

Add... Remove Contents

If backup files exist: Append

☒ Create a backup file for every database

☒ Create a sub-directory for each database

Folder:

Backup file extension: bak

☐ Verify backup integrity

Set backup compression: Use the default server setting

11. Provide the folder where the backups are being stored, the extension, and if you want to include subfolders, as well as how long you want to keep the files before deleting them:

Maintenance Plan Wizard

Define Maintenance Cleanup Task

Configure the maintenance task.

Delete files of the following type:

- ☒ Backup files
- ☐ Maintenance Plan text reports

File location:

- ☐ Delete specific file

File name:
- ☒ Search folder and delete files based on an extension

Folder:

File extension:

☒ Include first-level subfolders

File age:

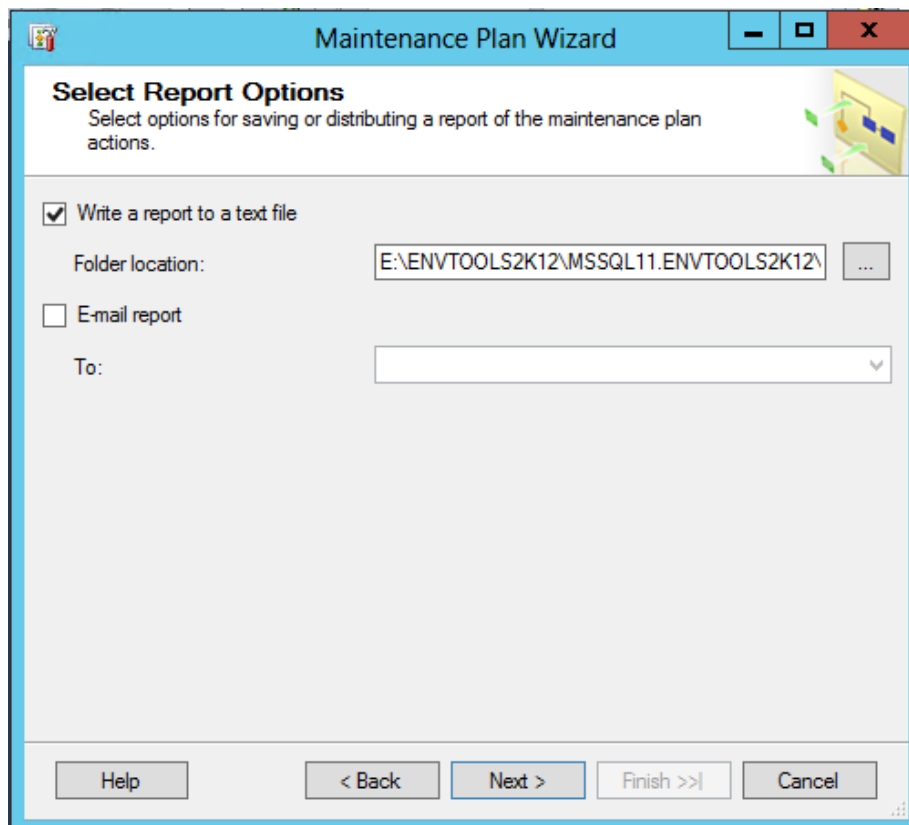
- ☒ Delete files based on the age of the file at task run time

Delete files older than the following:

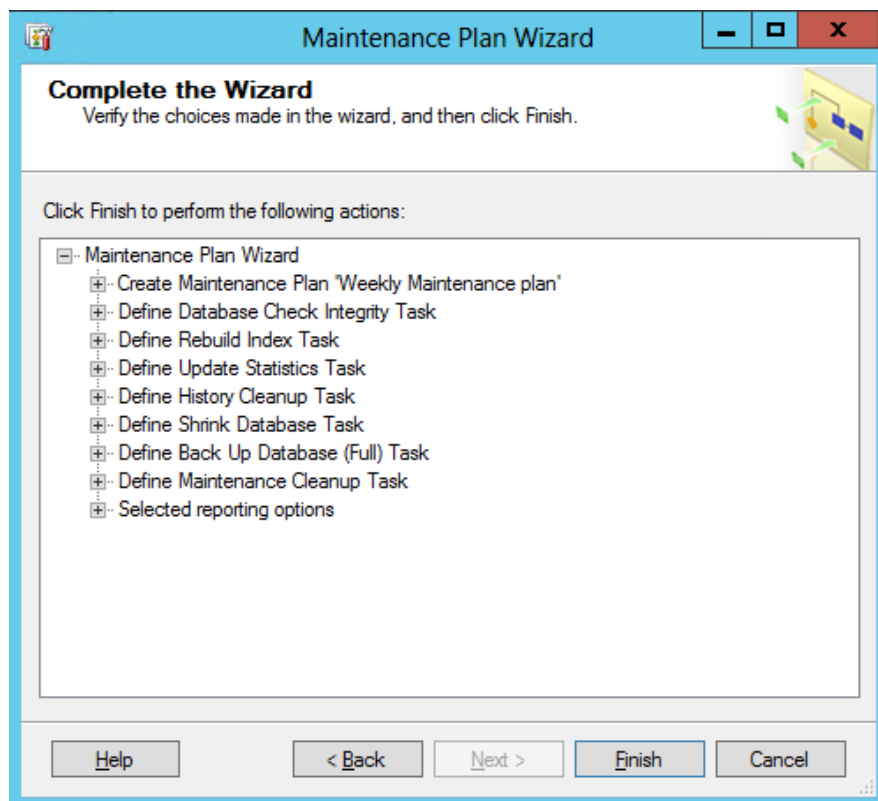
2 Week(s)

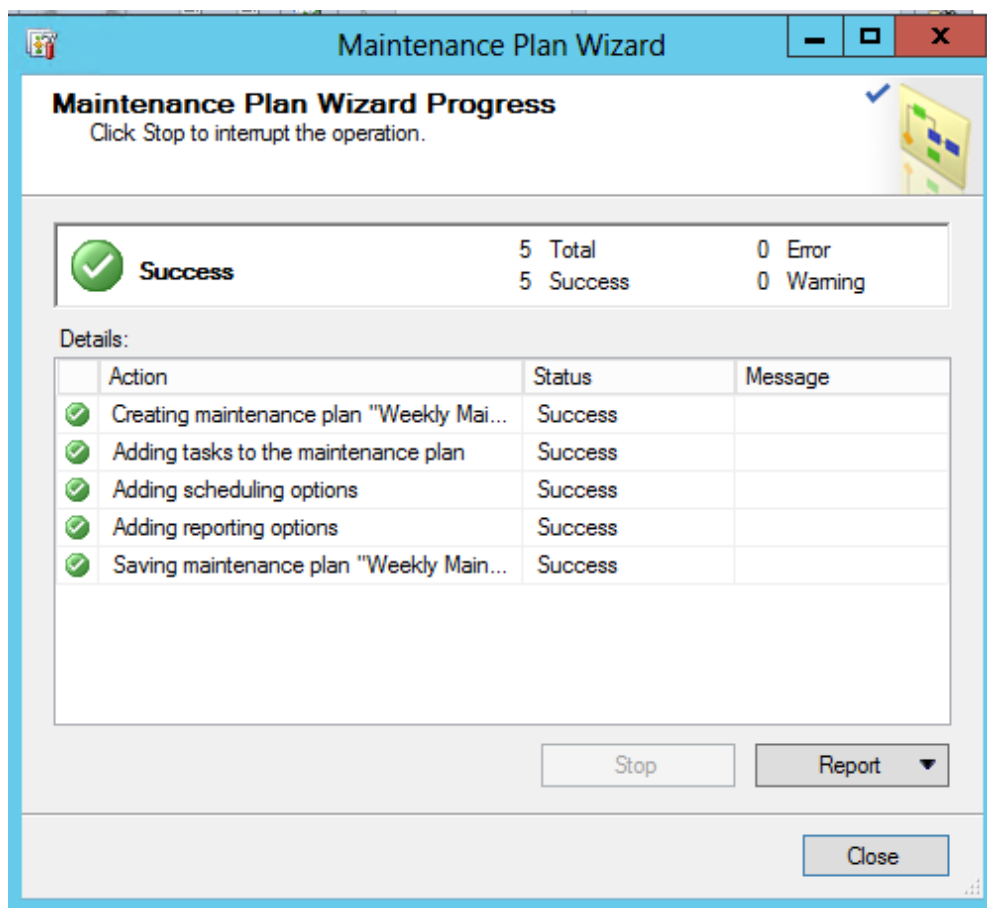
Schedule:

12. Keep the default :



13. Click **Finish** to create the maintenance plan, and the SQL Server job:





HOPEX RDBMS repositories specific administration actions

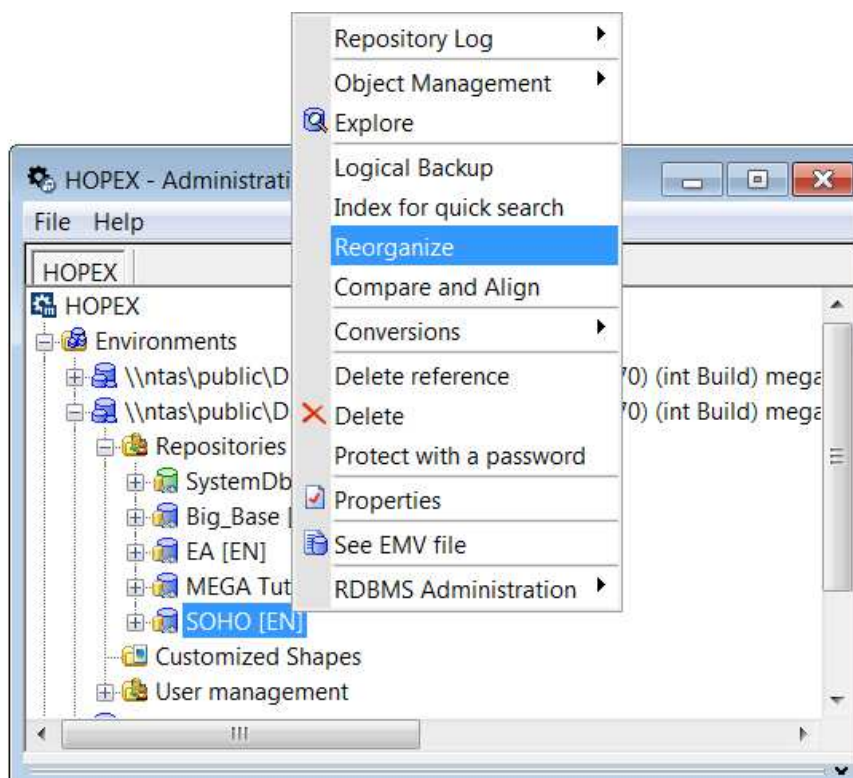
Migrating Your Data from One Storage Support to Another

Previous versions of Hopex were compatible with GBMS (proprietary Mega data format), and Oracle. This section shows how to convert data from one of those to SQL Server.

General procedure:

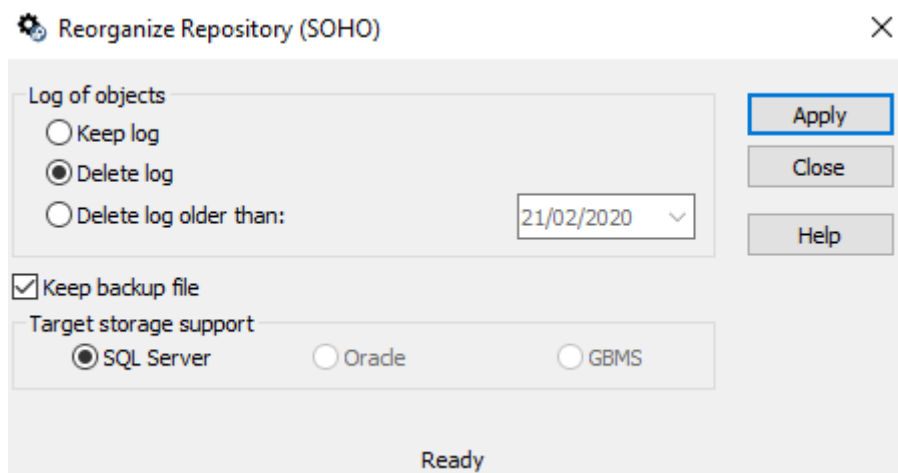
1. Start HOPEX **Administration.exe**.
2. Connect to the environment containing the repositories to be migrated.
3. Expand the **Repositories** folder.
4. Right-click a repository and select **Reorganize**.

NB: Launch a complete environment migration starting with the data repositories and finishing with the SystemDb repository.



To reorganize a repository:

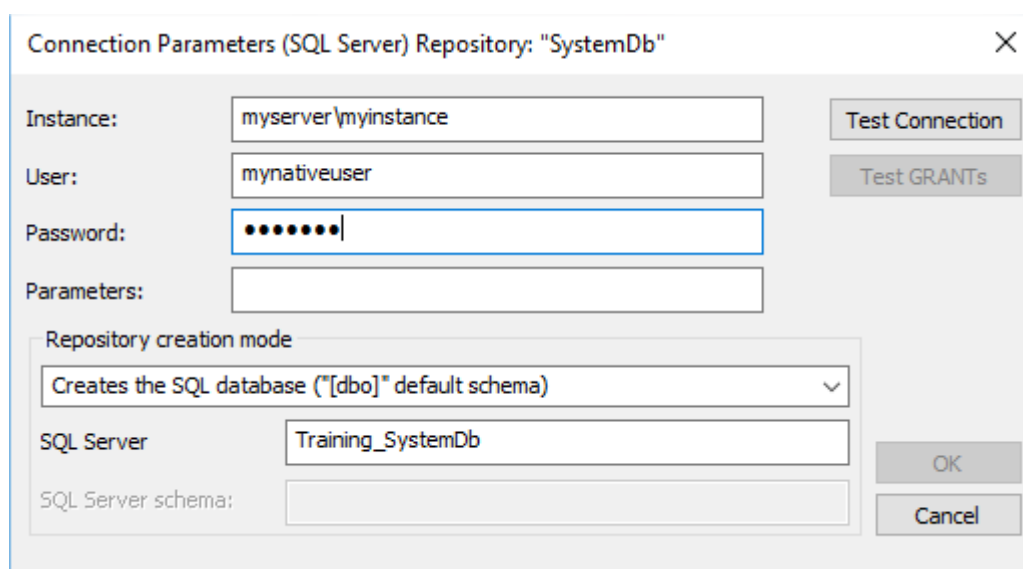
1. Select the expected **Target storage support**.



The 'Reorganize Repository (SOHO)' dialog box is shown. It has a title bar with a gear icon and a close button. The main area contains several options: 'Log of objects' with radio buttons for 'Keep log', 'Delete log' (selected), and 'Delete log older than:' with a date picker set to '21/02/2020'. There is a 'Keep backup file' checkbox which is checked. Below this is the 'Target storage support' section with radio buttons for 'SQL Server' (selected), 'Orade', and 'GBMS'. On the right side, there are three buttons: 'Apply' (highlighted with a blue border), 'Close', and 'Help'. At the bottom center, the status 'Ready' is displayed.

2. Click **Apply** to start the reorganization.

You are prompted to confirm or change the SQL Connection parameters.



The 'Connection Parameters (SQL Server) Repository: "SystemDb"' dialog box is shown. It has a title bar with a close button. The main area contains several fields: 'Instance:' with the value 'myserver\myinstance', 'User:' with the value 'mynativeuser', 'Password:' with a masked password '●●●●●●', and 'Parameters:' which is empty. To the right of these fields are two buttons: 'Test Connection' and 'Test GRANTS'. Below these fields is a section titled 'Repository creation mode' with a dropdown menu showing 'Creates the SQL database ("dbo" default schema)'. Below this is a field for 'SQL Server' with the value 'Training_SystemDb' and a field for 'SQL Server schema:' which is empty. At the bottom right, there are two buttons: 'OK' and 'Cancel'.

The **Test connection** step must be successful for the process to continue.

The **Test GRANTS** step must be successful for the process to continue.

Note: To be successful, there should be no storage on the Sql Server concerning a HOPEX repository with the same name in a same HOPEX environment.

If your Sql Server User does not have the right to create databases, you need to ask your DBA to create an Sql Server database following the naming rule: <EnvironmentName>_<RepositoryName>. You should then choose the option "Use existing Sql Server Database".

Results:

- The database is now migrated to the storage you chose.
- The .emq (SQL Server) file corresponding to the newly created repository storage is created.
- The Megaenv.ini file is updated.
- The logical backup file, used during the process, is stored in the 'work' folder of the source repository.
- This backup is named according to the following format: Bkp_Date_BaseName.mgr .

Restoring a HOPEX environment from formatted data

There are some cases when it is needed to recreate a repository in HOPEX Administration from an existing set of data (a previously HOPEX formatted repository). For example, after a physical corruption (disk crash) of the machine hosting the HOPEX repository folder tree.

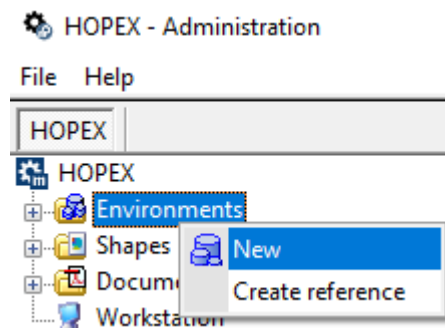
In such a situation, since the repository folder contains files indicating the way to reach the data and on which database server it can be found, the data could be considered lost from the HOPEX point of view.

It is necessary to understand that, from then on, HOPEX needs a new way to access the data inside the RDBMS. This is why this action is seen as a **Restoration** of the data: a re-creation of the repository folder structure allowing to re-save the way to access the data.

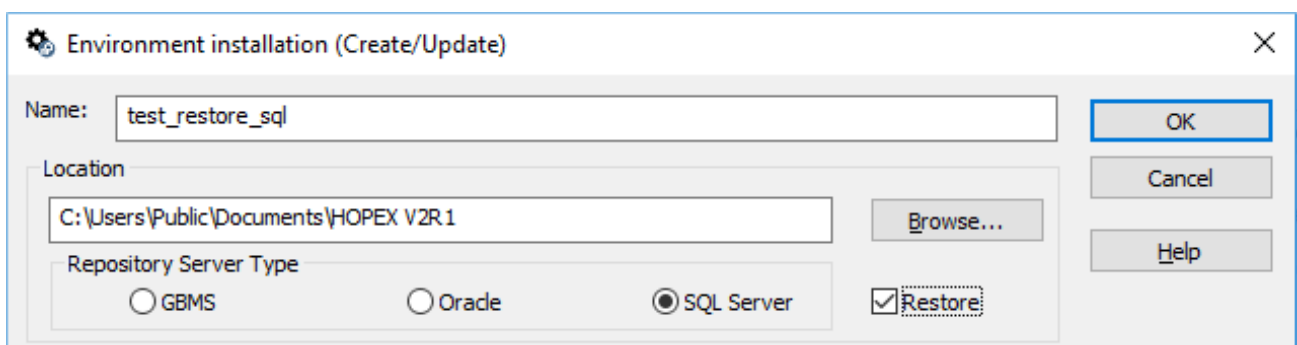
This method can also be used for duplicating an environment from a production infrastructure to a test infrastructure (or vice versa). For doing so, all the repositories (including the SystemDb) must be duplicated first in the RDBMS. The restoration can then be done on the duplicates repositories, starting with the SystemDb.

Restoring an environment (SystemDb repository)

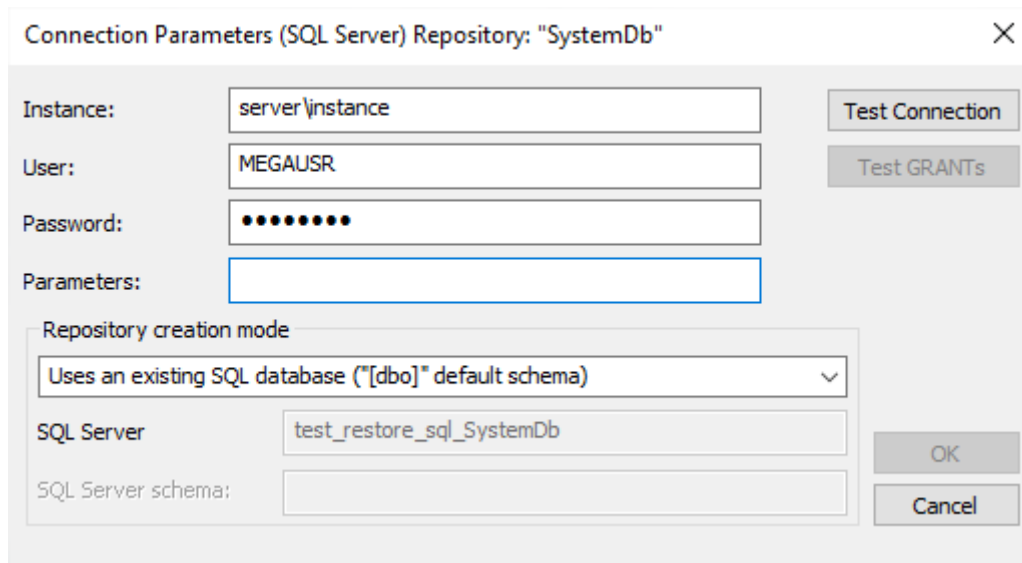
1. Start HOPEX **Administration.exe**.
2. Right-click the **Environments** folder and select **New**.



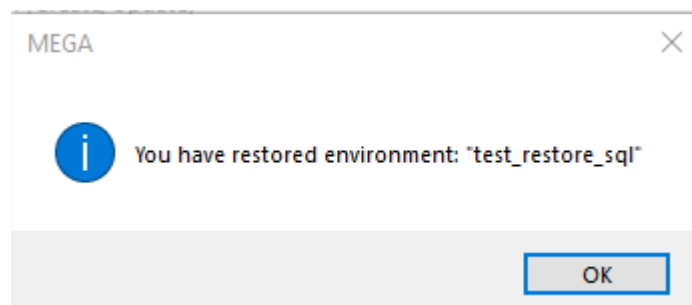
3. In **Name**, enter the name of the environment that is to be restored (the exact same name as the one used for the first creation).
4. Select **Restore**.



5. Click **OK**.
6. Specify the connection parameters for accessing the RDBMS where the HOPEX -yet-unreachable data is located.



7. Click **Test Connection**.
The test must be successful for the process to continue.
8. Click **Test GRANTS**.
The test must be successful for the process to continue.
9. Click **OK**.
The SystemDb repository is restored.



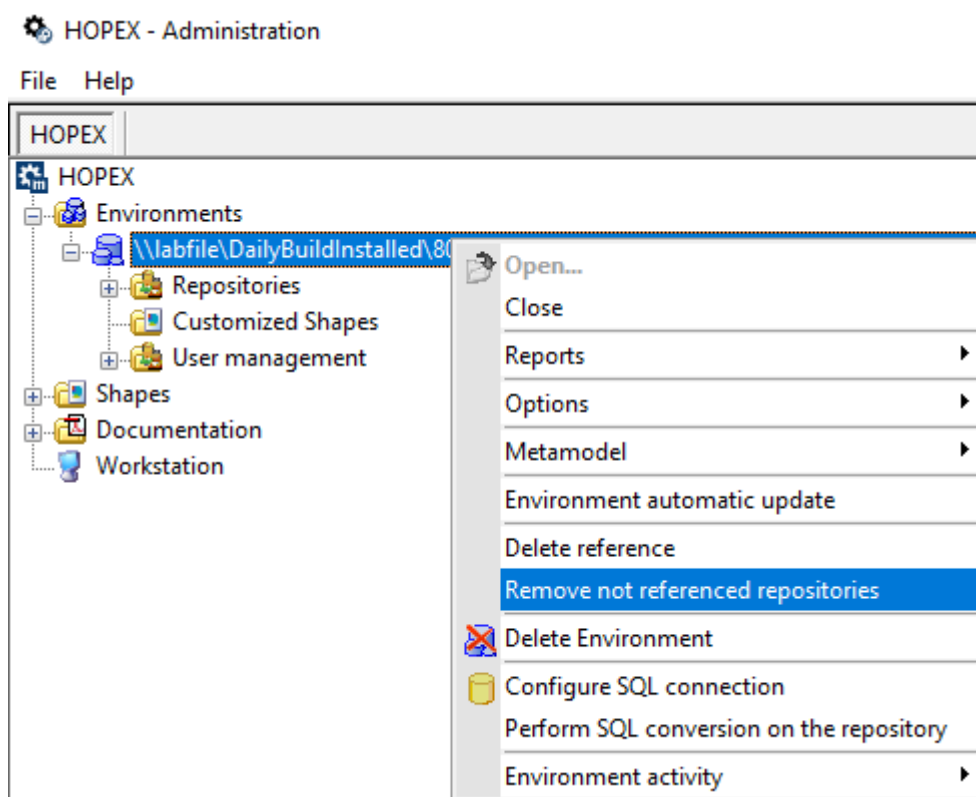
Once these actions are performed successfully, there are a few more actions to perform to be able to restore the repositories that were referenced into the newly restored environment.

At this point, if you open the environment that was just restored, you will see the following warning message: **"The <repository name> is not referenced"**).

The reason is that the environment that was just restored has "a knowledge" of the repositories that should be referenced in it but the references for those repositories do not yet exist in the folder tree structure of the newly restored environment.

To be able to re-reference the required repositories by restoration in this environment, you must first purge that “knowledge”:

1. Right-click the Environment and select **Remove not referenced repositories**:



Important notes



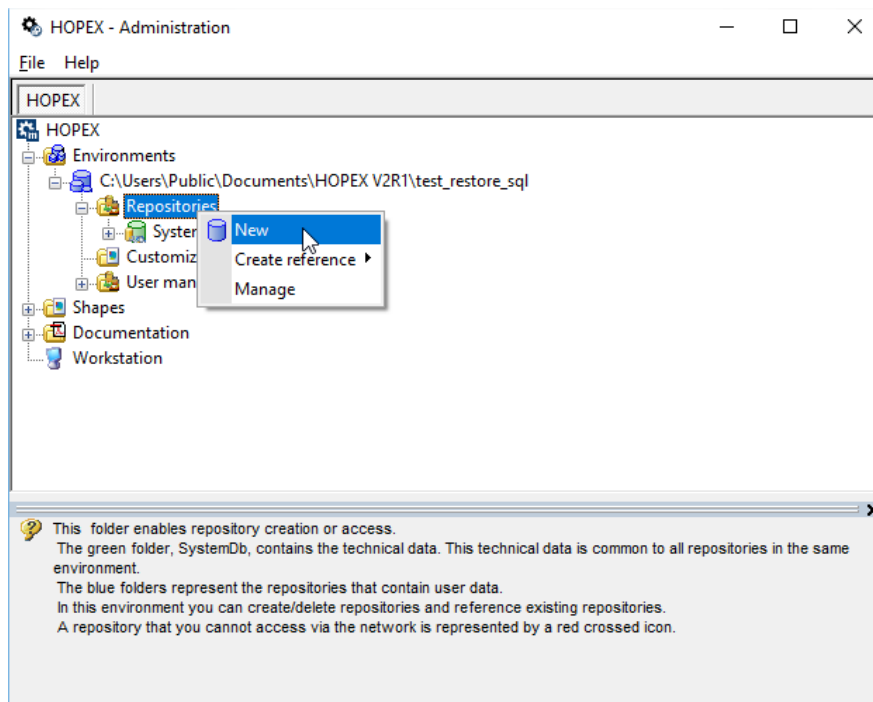
- DO NOT use **Remove not referenced repositories** if the environment is in use somewhere else as it will delete the references to the repositories there too !
- Use it only on an environment that is a physical copy on the RDBMS storage side.
- Be careful that the repositories also must be restored from a physical RDBMS copy (see next chapter for repositories restoration).
- Not taking care of this will lead to situations where users might think that they are using different sets of data when they are actually using and modifying **the same repositories** !

Restoring a data repository

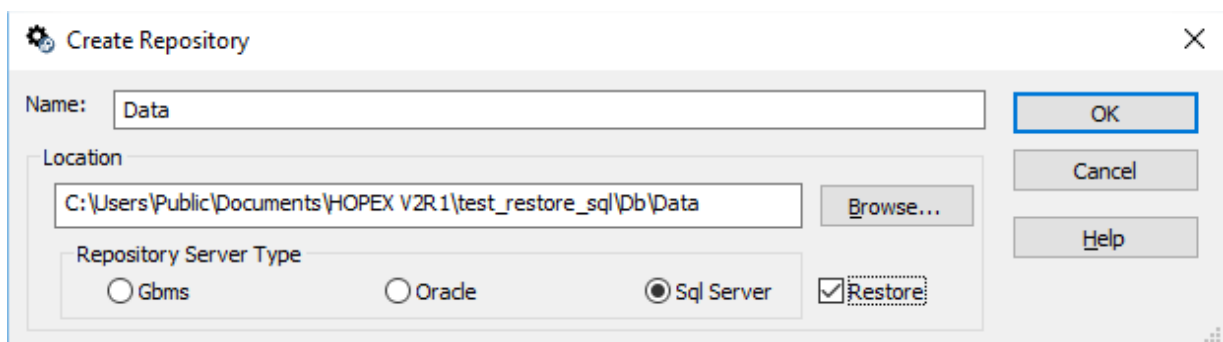
Note: A repository can only be restored within an environment that has the same name as the one in which the repository was originally created. An environment with the same name can be recreated before restoring the repository in it or the actual environment can be restored beforehand.

To restore a data repository:

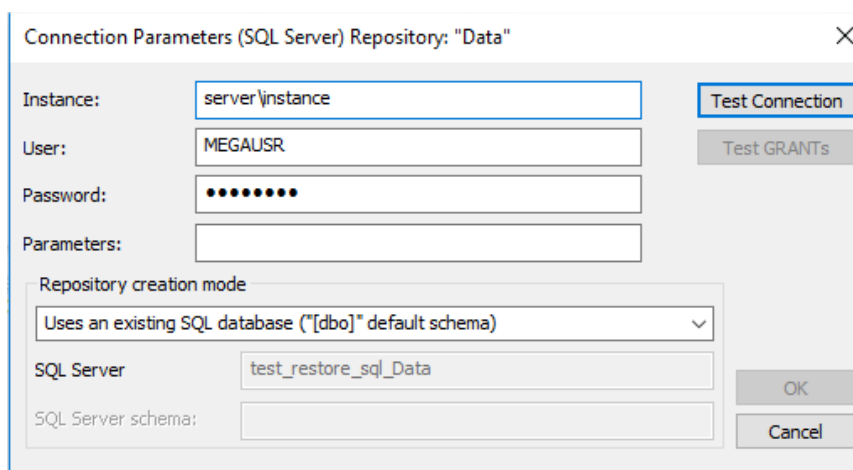
1. Start HOPEX **Administration.exe**.
2. Connect to the environment in which you want to restore the repository
3. Right-click the **Repositories** folder and select **New**.



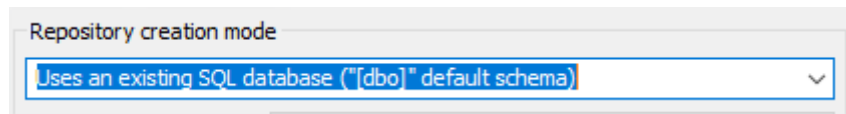
4. In **Name**, enter the name of the repository that is to be restored (the exact same name as the one used for the first creation).
5. Select **Restore**.



6. Click **OK**.
7. Specify the connection parameters for accessing the RDBMS where the HOPEX -yet-unreachable data is located.



NB: For SQL Server, the “Creation Mode” parameter is disable (the choice is not possible) when the “Restore” checkbox is checked. This is because in this situation, HOPEX is actually told to re-attach to physical data so no database creation or repository initialization will be carried out.



8. Click **Test Connection**.

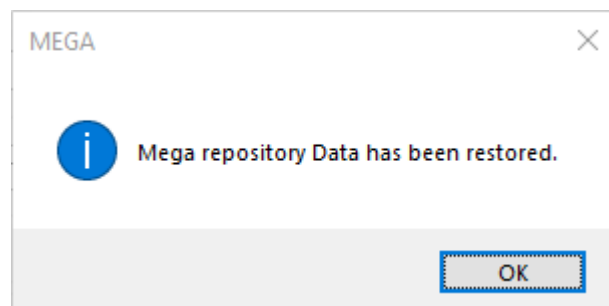
The test must be successful for the process to continue.

9. Click **Test GRANTS**.

The test must be successful for the process to continue.

10. Click **OK**.

The repository is restored.

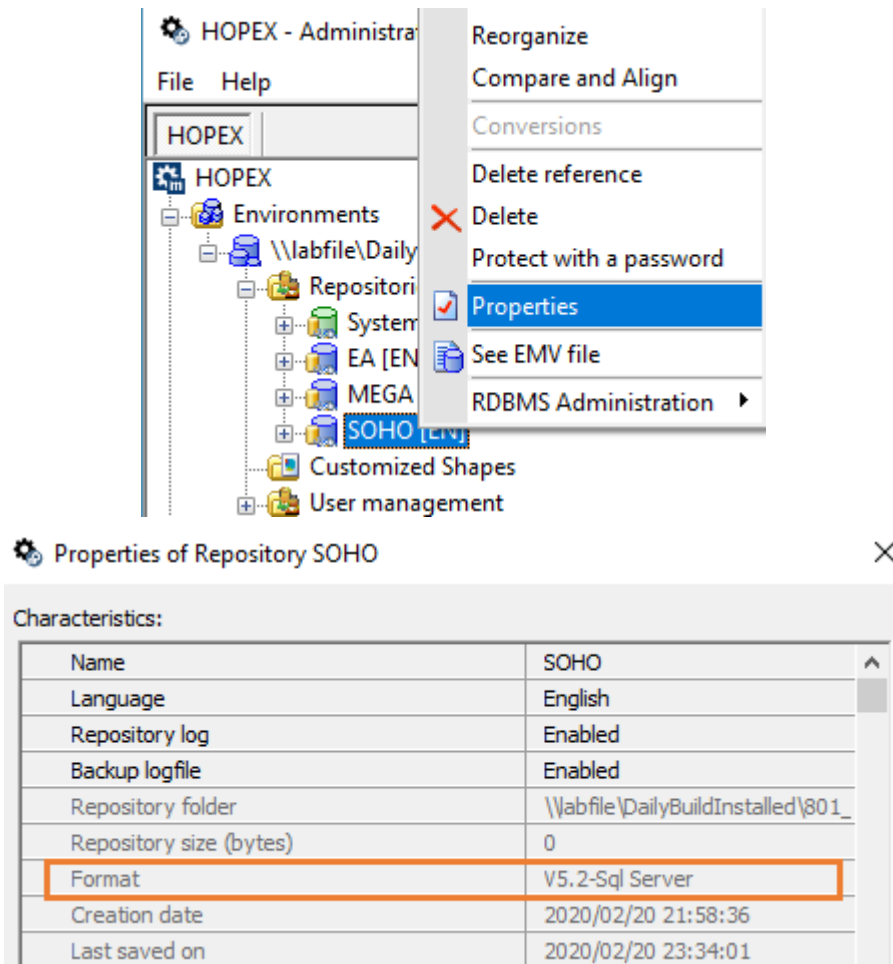


Handling of HOPEX RDBMS repositories specific internal format

There is an internal format used by HOPEX when accessing a repository that is stored on **SQL Server**.

To view this internal format version:

1. Start HOPEX **Administration.exe**.
2. Right-click the HOPEX repository (either SystemDb or data repository) and select **Properties**.

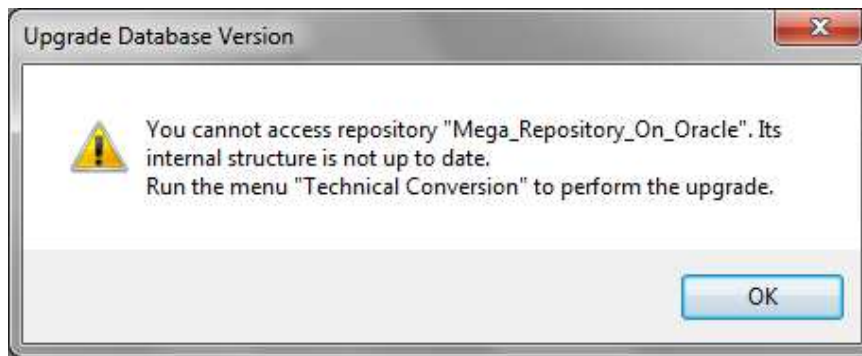


When upgrading your HOPEX installation (applying a Cumulative Patch or migrating your data from a HOPEX SP version to the next one), there might be some modifications leading to a new **internal format** version.

From Mega 2009 SP5, new menus are available to manually activate this **internal format** upgrade.

Note: Before Mega 2009 SP5, the upgrade was made "on the fly" when first accessing the Mega repository with a Mega program corresponding to a more recent **internal format** version.

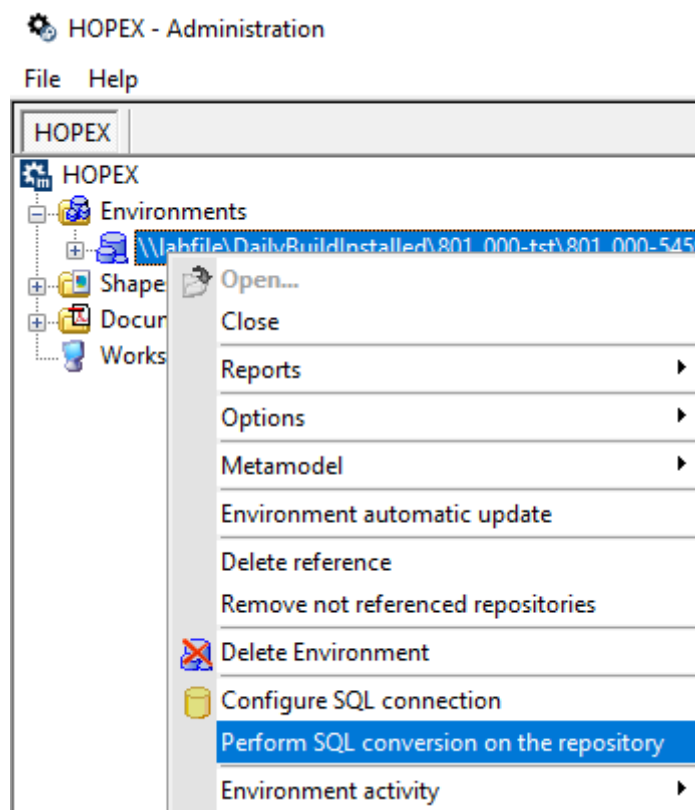
When you need to upgrade the **internal format** version, you are prompted to do it with the following window:



Note: The technical conversion of the repositories of the environment must be done before upgrading to the environment:

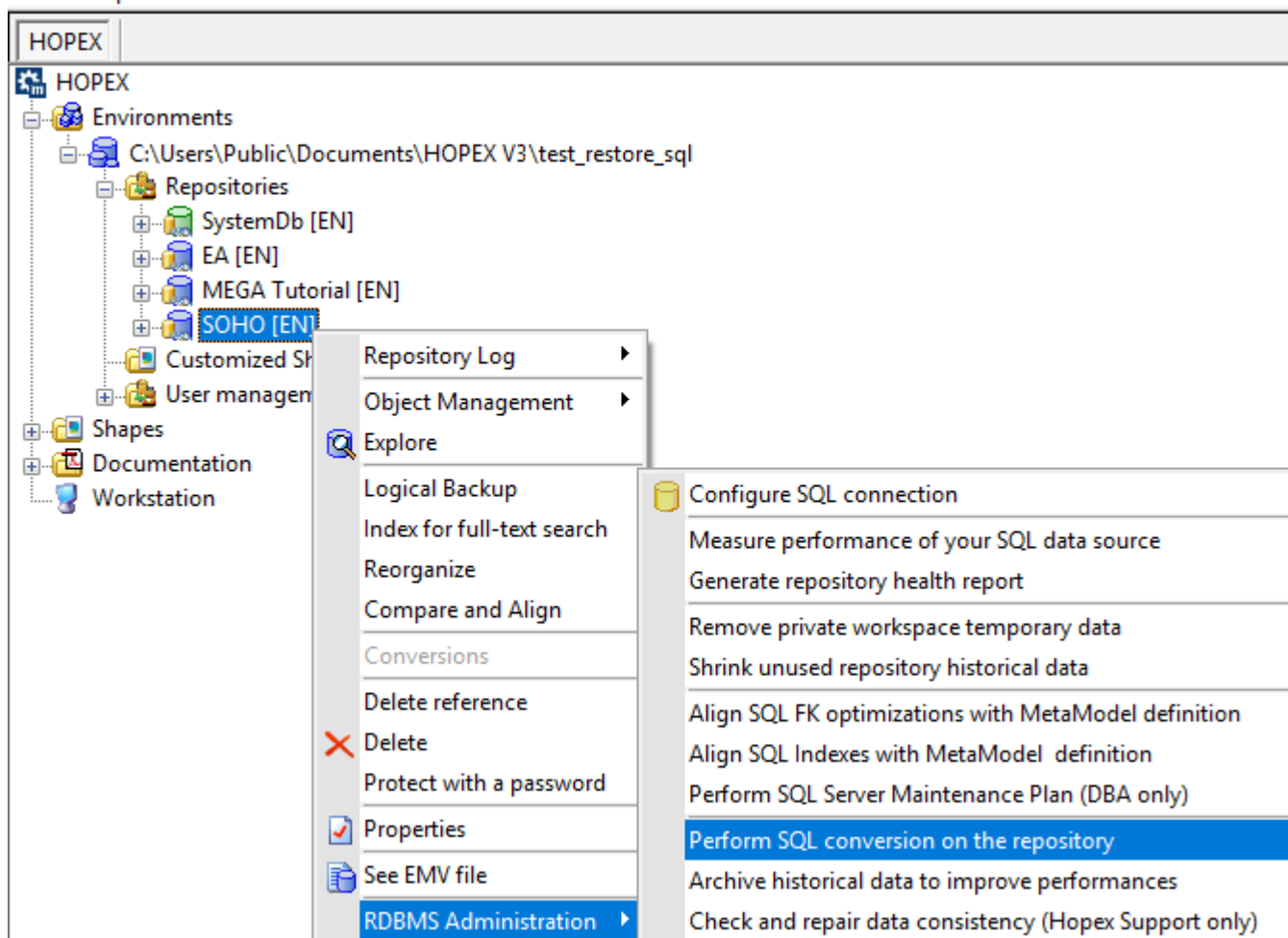
1. Apply the technical conversion on the SystemDb:

Right-click the environment and select **Perform SQL conversion on the repository**.



2. Apply the technical conversion on the other data repositories of the environment:

For each repository, right-click the repository and select **RDBMS Administration > Perform SQL conversion on the repository**.



Vocabulary

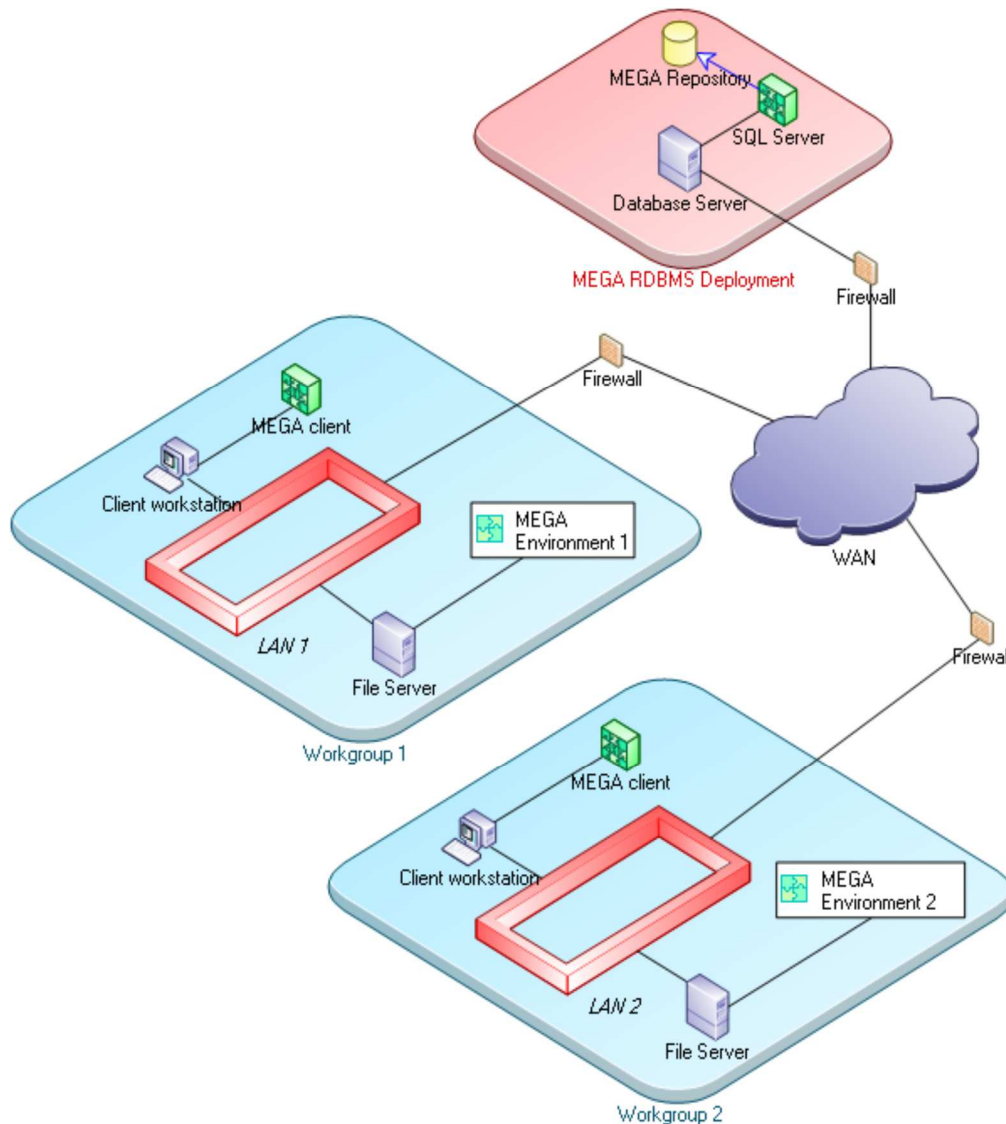
Term	Comment
Database	A database is a collection of data, usually in the form of tables or files, under the control of a database management system (DBMS).
Database server (hardware)	<p>A database server is a machine providing database services to other machines. In this document the database server is a machine running relational database management systems. A database server can host one or several instances.</p> <p>Example:</p> <ul style="list-style-type: none">• Server 'iba.company.com'• Server '192.888.777.666'• Server 'SQL02'
DBA	The DataBase Administrator is responsible for administering, monitoring, and maintaining the database.
DBMS	<p>A DataBase Management System (DBMS) is a set of software programs that controls the organization, storage, management, and retrieval of data in a database.</p> <p>Example: GBMS, Oracle...</p>
GBMS	GBMS is MEGA's historical proprietary DBMS.
HOPEX Environment	On RDBMS installations, an environment is a group of directories where HOPEX generates documents, log files, etc.
RDBMS	<p>Relational DataBase Management System.</p> <p>Examples: Oracle, SQL Server, DB2 Universal Database,...</p>
Repository	<p>A repository is a structured collection of data.</p> <p>A HOPEX repository is a collection of HOPEX data. Data is structured in relation to a metamodel. Object names are often unique within the repository or with a namespace of the repository.</p>
Schema	A schema object is a logical data storage structure.

Term	Comment
	<p>In Oracle, it is a collection of objects (example: tables, views, indexes, procedures, functions...) mapped to an Oracle user. A schema is stored in one/several tablespace objects of the database.</p> <p>It is strongly recommended to isolate each HOPEX Repository in a separate Oracle schema (User Repositories AND SystemDb repository)</p>
Storage format	<p>HOPEX term. It defines the type of DBMS storing HOPEX data.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> ○ SQL Server: storage in SQL Server DBMS ○ GBMS: storage in HOPEX historical DBMS ○ Oracle: storage in Oracle DBMS
SystemDb repository	<p>HOPEX Term. It is a HOPEX repository that stores system data, such as, user definition, metamodel definition, template definitions, queries, diagram configuration. This data can be shared by all user repositories within a HOPEX environment. A SystemDb repository is associated to one/several user repositories.</p>
User repository	<p>HOPEX Term. This is a HOPEX repository storing data, such as diagrams, org-units...</p>

Appendix - FAQs

Is it possible to share user repositories and the SystemDb repository through user's workgroups that do not share a file server?

Yes. You can duplicate HOPEX Environment on each side to obtain this kind of configuration.



Is it possible to have a user repository stored on a GBMS and a SystemDb repository stored on a SQL server?

No. Some features might work but it is not tested and not supported. Moreover many specific features will not work.

Is it possible to consult the data from a SQL Server?

It is technically possible and supported (e.g.: SELECT statement). However, this requires knowledge of the HOPEX RDBMS implementation and the HOPEX Metamodel. It is much easier to query the data from within HOPEX.

Is it possible to update the data from an SQL Server?

It is technically possible but **NOT supported** (e.g.: UPDATE or DELETE statement). This requires the knowledge of the HOPEX RDBMS implementation and of the HOPEX Metamodel. Data updates must be performed from within HOPEX. All updates from outside the HOPEX application are made at the customer's risk. Consequences of inappropriate updates will not be supported.

Must License Installation Guide HOPEX V3

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 3 029 190 € - RC Paris B 385 185 806 000 51 / NAF 741 G

1. SUMMARY

Check if a more recent version of this document is available in online documentation (MEGA Community).

This document describes the procedures necessary for installing Must licences with HOPEX V3.

It applies to all Front-ends.

It does not describe:

- System requirements and possible architectures (see architecture overview documentation).
- How to install a product release (see installation documentation).
- How to manage installations (see administrator manuals).
- How products are licenced (see licensing documentation).
- How to use features (see user manuals).

1. SUMMARY	2
2. FOREWORD	4
3. MUST LICENCE UTILITY.....	6
3.1. User Interface.....	6
4. INSTALLATION PROCEDURES	9
4.1. Communicating with MEGA Sales Administration.....	9
4.2. Choosing a machine to host the Must licence folder	9
4.3. Creating a Must licence folder	9
4.4. Sending the UNC address of the licence folder	10
4.5. Installing a Must licence file.....	10
4.6. Configuring the licence folder in the HOPEX installation (direct reference)	11
4.7. Configuring the licence folder in the shared configuration folder (indirect reference)	11
4.8. Uninstalling the Must licence.....	11
4.9. Resetting the configuration files	12
4.10. Converting licence.....	12
4.11. Configure file permissions.....	13
5. CONFIGURATION AND MONITORING PROCEDURES.....	14
5.1. Configuring the command line (/RO /RW code)	14
5.2. Specifying a default licence.....	15
5.3. Declaring users	15
5.4. Configuring possible users of products.....	16
5.5. Instant monitoring of licence connections	18
6. CONFIGURING LICENSING MODES AND USER TYPES	19
6.1. Configuration of main users with concurrent mode (floating mode).....	20
6.2. Configuration of main users with dedicated mode	21
6.3. Configuration of main users with shared mode	22
6.4. Configuration for viewer users	23
6.5. Configuration for contributor users	24
7. INSIDE.....	25
7.1. Licence deployment model	25
7.2. Licence execution.....	26
7.3. File access.....	26
8. FAQs AND TROUBLESHOOTING	27

2. FOREWORD

HOPEX Must licensing is a technology of network licences provided by MEGA.

To obtain or update your licence, contact your sales representative.

- A UNC will be requested.
- A .must licence file will be sent with installation instructions.

A Must licence:

- Is a file with a .must extension.
- Contains the definition of the licence (locking information, expiration date and list of products).
- Is locked on a shared folder (UNC address).

Must licence installation mainly consists in:

- Installing the licence.
- Configuring the licence folder in the HOPEX installation.

If you want to directly install a Must licence, go to the section 'Installation procedures' of this document.

After installation, the Must licence can be configured to better control execution:

- Configuring the command line (/RW code)
- Configuring user x licence mapping.
- Configuring user to product mapping.

A **Must licence utility** is available for the licence administrator to make these configurations and monitor licence use.

The Administration Console is still used to create and configure HOPEX users.

A **Web licensing console** is also available to configure licence.

It does not enable to monitor use of token at a given moment.

The list of available services varies with the front-end:

Service	HOPEX Web Front-End	HOPEX Windows Front-End
Locking	•	•
Shared licences	•	•
Dedicated licence	•	•
Concurrent licence	•	•
Multiple licences	• (NR)	•
Cluster licence	•	•

NR: not recommended because execution warning regarding are not displayed in HOPEX Web Front-End.

Definition of services:

- **Locking:**
 - The licence is programmed for a specific UNC address. The availability of this address is checked at runtime.
- **Dedicated licences:**
 - It is possible to program a licence when a token for a product will be dedicated to a user. The number of tokens equals the number of users. The product is said to be programmed in dedicated mode.
- **Shared licences:**
 - It is possible to program a licence when a token for a product will be assigned to a list of possible users. The number of tokens is lower than the number of possible users. The product is said to be programmed in shared mode.
- **Concurrent licences:**
 - It is possible to program a licence when a token for a product will not be assigned to any user. The number of possible users is set to 0. The product is said to be programmed in concurrent mode.
- **Multiple licences:**
 - For the same HOPEX installation, it is possible to use different licences to enable different access policies for different populations of end-users.
- **Cluster licence:**
 - The same licence can be used for a set of HOPEX installations on different machines (cluster).

Independently from the products/solutions, there are three exclusive types of users:

- **Viewer users:** consult data, search, use collaboration features
- **Contributor users:** consult data, search, perform limited updates, use collaboration features
- **Main users:** consult data, search, perform all updates in particular with diagram editor, use collaboration features

3. MUST LICENCE UTILITY

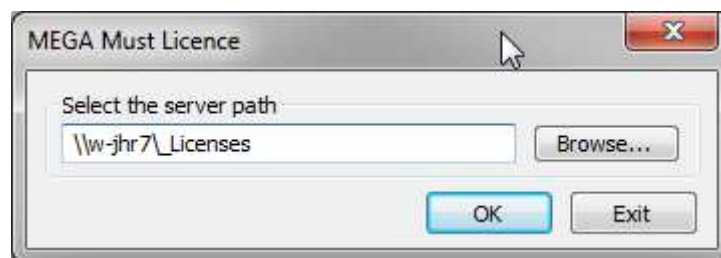
A launcher **licensing.exe** is installed in the root folder of the HOPEX installation. It is a shortcut to the program **mgwusrmng.exe** located in the 'System' folder of the HOPEX installation.

3.1. User Interface

Several Windows are available:

- Select server path window: to locate a folder containing the Must licence.
- User management window: to configure Must licences located in this folder.
- Select HOPEX installation window: to locate the configuration folder of the HOPEX installation.

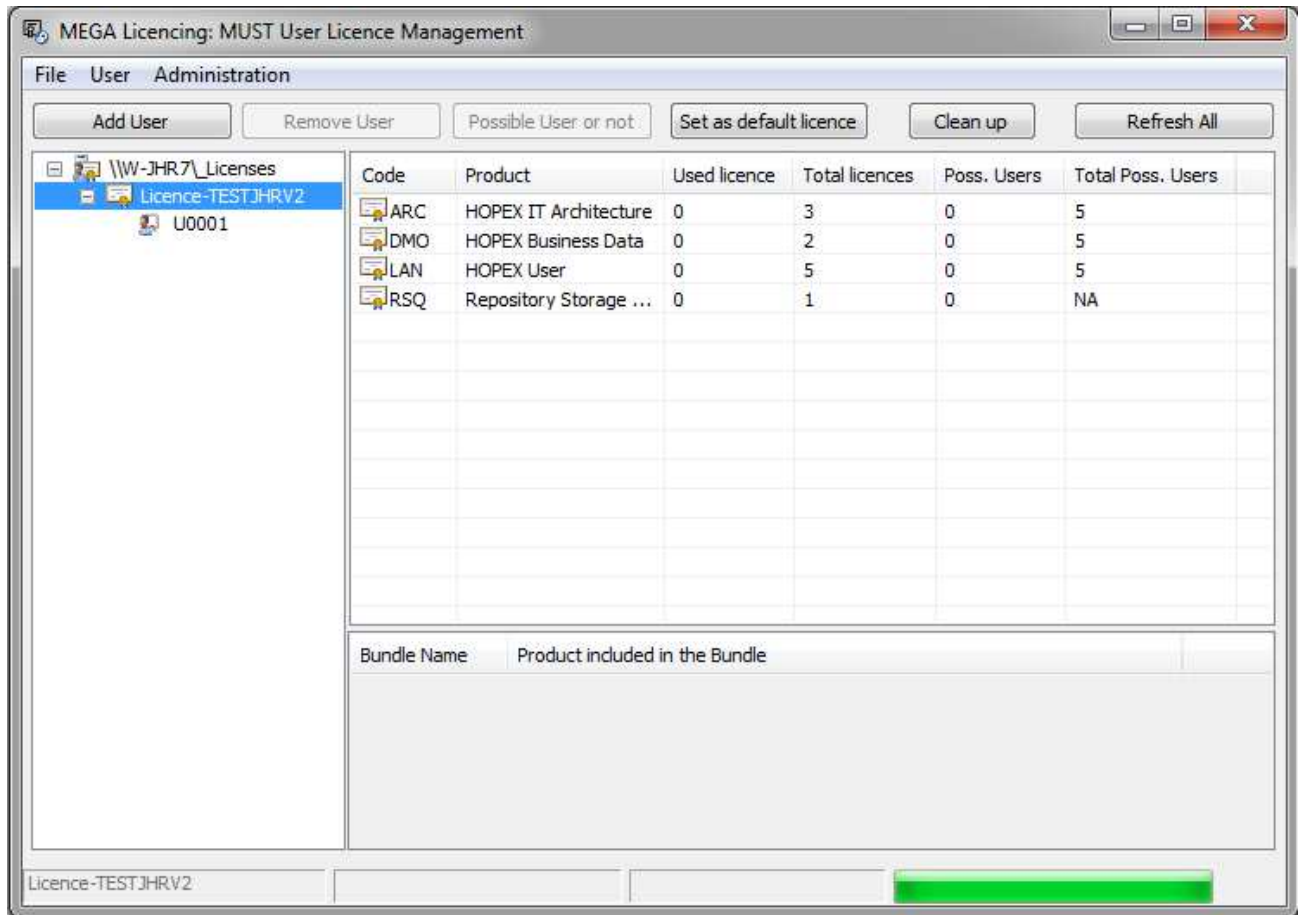
Select server path window



Click 'Browse' and select the folder containing the Must licence to be configured. The 'OK' button is enabled as soon as a file with the .must extension is identified.

Verify that the server path is the same as the UNC address chosen for the licence.

User management window





This window displays several elements:

- A top menu (File, User, Administration) and a toolbar 'Add User, Remove User..)
- The left pane displays the Must licence available in the selected folder.
- The top right pane displays the products available for the selected licence.
- The bottom right pane displays the bundle definition, if any.



The top right pane has several columns. The list is different if a user or a licence is selected:

- **Code:** the code of the technical product.
- **Product:** the name of the technical product.
- **Connected:** the number of users currently logged in to the product (this figure changes over time).
- **Used licences:** the number of licence tokens currently used for the product (this figure changes over time).
- **Remaining licences:** the number of licence tokens currently available for the product (this figure changes over time).
- **Total licences:** the number of licence tokens programmed for the product (this figure does not changes over time).
- **Poss. User:** the number of users that are set as possible users of the product (this figure changes over time).
- **Remaining Poss. Users:** the number possible users currently available for the product (this figure changes over time).
- **Total Poss. Users:** the number possible users programmed for the product (this figure does not changes over time).

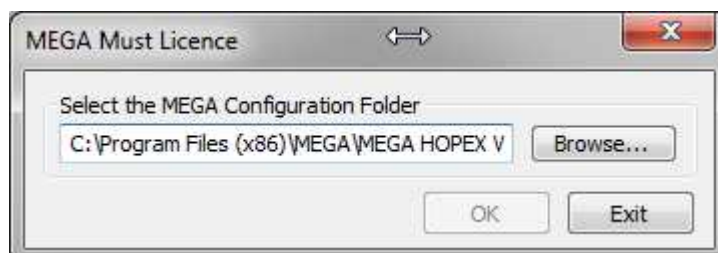
The licence status is displayed in the left pane:

Display	Status	Possible causes
 Licence-T0001	Valid	-
 Licence-T0001	Invalid	Licence has expired Locking failed: the folder address containing the licence file does not match the expected UNC

The user status is displayed:

Display	Status
 U0001	Connected
 U0001	Not connected

Select HOPEX installation window



Click 'Browse' and select the folder containing the 'CFG' folder of the HOPEX installation to be configured. The 'OK' button is enabled as soon as the Megasite.ini file is identified.

4. INSTALLATION PROCEDURES

The following procedures apply to all front-ends. They enable to install the Must licence and protect a HOPEX installation with this licence.

4.1. Communicating with MEGA Sales Administration

There are times where you will need a new Must licence:

- When you purchase a new HOPEX product or solution.
- When you purchase additional licences or users of HOPEX products or solutions.
- When you relocate Must licence folder.

To obtain or update your licence, contact your sales representative.

- A UNC will be requested.
- A .must licence file will be sent with installation instructions.

If a problem occurs during licence installation, see the 'FAQs and troubleshooting' section of this document. For additional assistance, contact the appropriate Support Center.

Must licence installation consists in:

- Installing the Must licence file.
- Configuring the licence folder in the HOPEX installation.

Must licence update consists in:

- Verifying that no user is connected to the former licence.
- Removing the former licence.
- Resetting the configuration files
- Installing the updated licence.

4.2. Choosing a machine to host the Must licence folder

List of requirements:

- No specific hardware requirements (CPU, Ram). However, the machine hosting the Must licence must be an efficient file server.
- The machine hosting the Must licence must be available for all users running the HOPEX Kernel.

List of recommendations:

- The machine must be an efficient file server:
 - Select top quality components for disks and disk controller cards.
 - During installation and configuration, choose all options that favor file service performance.
- Choose an NTFS disk.
- Choose a DFS-based folder for the licence folder

4.3. Creating a Must licence folder

If you do not have the technical skills or the authorization required for this step, contact you system administrator.

Steps:

- Choose a machine to host the Must licences.
See above.

- Create a shared folder on this machine.
This will be the licence folder. See requirements below.
- Configure this share folder.
See requirements below.

Folder sharing requirements:

- The licence folder must be accessible as a UNC address, meaning a shared folder with one unique address on the network.
- Examples of authorized sharing:
\\Server001\Apps\Licences
\\Domain01\Applications\HOPEX\Licences (DFS)
[\\Server001.Domain01.com\Licences](#) (FQDN)
- Examples of unauthorized sharing:
\\Server002\c\$\ HOPEX\Licences (administrative share)
M:\Licences (network letter)

Licence folder requirements:

- The licence folder must be accessible as a UNC with **full control** to all Windows users that are allowed:
 - To configure a Must licence.
 - To run HOPEX Kernel programs with a Must licence.
- If you want to configure smarter permissions, consult the 'FAQs and troubleshooting' section of this document.

4.4. Sending the UNC address of the licence folder

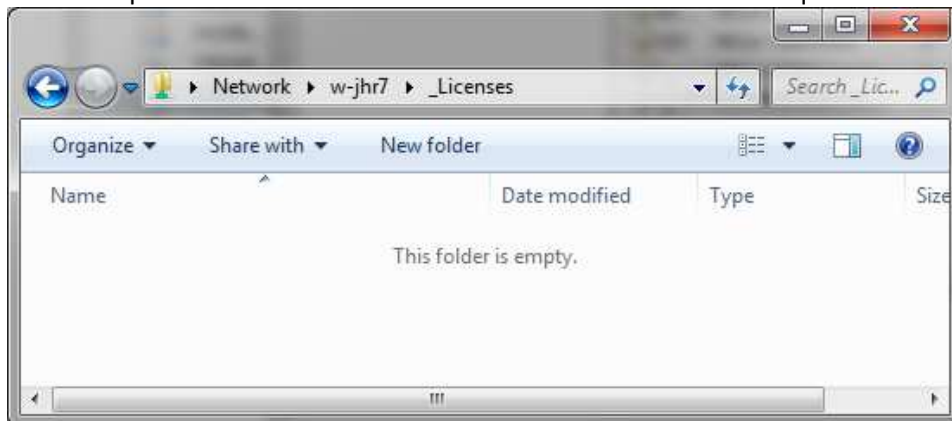
Prerequisite:

- Create a licence folder.

No specific utility is provided for this step: you can use Windows explorer.

Example:

- Navigate to the licence folder.
- Select the folder path in the address bar.
- Copy the folder path and send it to MEGA Sales Administration when requested.



4.5. Installing a Must licence file

Prerequisites:

- Get a Must licence file.
- Identify the licence folder. This folder must match the UNC on which the HOPEX licence file is locked.

- Verify that this folder exists and is shared (Windows permissions) for every Windows user that can run HOPEX Kernel through a Front-end.

Procedure:

1. With Windows explorer, select the folder matching the UNC.
2. Copy the .must file sent by MEGA Sales Administration to this folder.

Results:

- The Must licence is installed.

4.6. Configuring the licence folder in the HOPEX installation (direct reference)

Prerequisites:

- A Must licence is installed in the licence folder.
- A target HOPEX installation is available.

Procedure:

1. Run the Must licence utility.
2. In the menu, select File > 'Update config'.
3. Click the 'Browse' button.
4. Select the 'CFG' folder of the HOPEX installation.

Results:

- The Megasite.ini configuration file is updated. A section [Must licence] is created or updated.
Example:
[Must licence]
Path=\\server001\Apps\Licences
- It is possible to run the HOPEX installation on behalf of this Must licence.

4.7. Configuring the licence folder in the shared configuration folder (indirect reference)

Prerequisites:

- A Must licence is installed in the licence folder.
- A shared configuration folder contains a file Megasite.ini.
- A HOPEX installation is part of a cluster.

Procedure:

1. Browse the shared configuration folder.
2. Edit the file Megasite.ini.
3. Add a section [Must licence] and a variable 'Path'. Example:
[Must licence]
Path=\\server001\Apps\Licences

Results:

- In the shared configuration folder, the Megasite.ini configuration file is updated. A section [Must licence] is created or updated.
- It is possible that each workstation of the cluster shares the same licence.

4.8. Uninstalling the Must licence

Prerequisites:

- Identify the Must licence file to be uninstalled.
- Identify the licence folder. This folder must match the UNC on which the HOPEX licence file is locked.
- Verify with the Must licence utility that no user is currently logged on to the licence to be uninstalled.

Procedure:

1. With Windows explorer, select the folder matching the UNC.
2. Remove the .must file from this folder.

Results:

- The Must licence is uninstalled.

4.9. Resetting the configuration files

When replacing a Must licence with a licence having the same name and UNC, it is recommended to reset the Must licence configuration.

Otherwise, the licence may not run correctly in particular if the number of token has become lower for a product. A consequence it will be necessary to specify again the list of possible users.

Prerequisites:

- Identify the Must licence file to be uninstalled.
- Decided whether you reset the configuration
- Verify with the Must licence utility that no user is currently logged on to the licence to be uninstalled.

Procedure:

1. Run the **licensing.exe** utility as **Administrator**.
2. Select the server path where the licence is saved.
3. Select the licence in the left tree.
4. Right-click > **Reset Licence** configuration.
5. Confirm reset.

This will:

- Delete the possible user configuration.
- Delete the token files.
- Delete the file Router.ini

4.10. Converting licence

When upgrading from HOPEX V1R2-V1R3 CP8.0 or lower CP, it is required to convert the file Router.ini to a new format. Otherwise, various issues can occur.

Pre-requisites:

- Stop all activity regarding HOPEX Windows Front-End and HOPEX Web Front-End.

Procedure:

1. Run the **licensing.exe** utility as **Administrator**.
2. Click on the menu Administration > **Convert**.
3. Click the button **Refresh All**.

This will:

- Archive the file configuration file 'Router.ini as 'Router.bak'.
- Update the file 'Router.ini' to the new format:
 - A version tag is added (section [Router], version=x).
 - The reference to the domain (ex: @Domain01) is removed.
 - Duplicate line are removed.
- Technical files are renamed.
 - The reference to the domain (ex: @Domain01) is removed.

Example:

Router.ini (before conversion)	Router.ini (after conversion)
[User/Licence] U001@Domain01=Licence-T0001	[User/Licence] U001=Licence-T0001

U001@Domain02=Licence-T0002 U002@Domain01=Licence-T0001	U002=Licence-T0001 [Router] Version=2
--	---

4.11. Configure file permissions

At runtime, files will be created dynamically in a hidden subfolder in the licence folder.

It is necessary to configure file permissions so that execution is correct.

It is recommended to grant the permission 'Modify' for the licence folder (ex: \\Server001\Apps\Licences and its subfolders).

The list of windows users varies with the front-end:

Front-end	Users to be configured
Web Front-end	Only the service account for the HOPEX (IIS) web application should be configured (ex: D01\hopex). Contact the person in charge of installing HOPEX Web Front-end
Windows Front-end	Each end-user can be configured (D01\u0001, D01\u0002...). It is therefore recommended that a group is created for users of the Windows Front-end.

5. CONFIGURATION AND MONITORING PROCEDURES

The following procedures apply to all front-ends.

5.1. Configuring the command line (/RO /RW code)

Each product is associated to a product code.

Ex: HOPEX Business Process Analysis code 'HBPA'

A property 'Command line' can be configured at several levels:

Level	Comment
Profile level	Configuration at this level is recommended. As there are less profiles than users, configuration is easier to maintain.
User level (Login)	Configuration at this level is NOT recommended. It is mainly available for compatibility with previous versions.

At each level, it is possible to specify a command line with the following syntax:

`/RW'<list of product codes>' /RO'< list of product codes>'`

Example:

`/RW'DMO;HBPA' /RO'DBB'`

Where:

- /RW: defines a list of product code accessed in read/write mode.
Note that /K (previous specification) is equivalent to '/RW'
- /RO: defines a list of product code accessed in read/only mode.

Prerequisites:

- Identify the HOPEX environment containing the users to be configured.
- Get the table of product codes that you have bought.
- Get a company specification of user/profile x product assignment. The level of configuration (user level or profile level) must be specified for each user for the company.

Example of procedure to set /RW /RO for a profile:

- Run the Administration Console.
- Open the environment.
- Select the folder 'User Accounts > Profiles and Permissions'.
- Right-click > Manage.
- In the tab 'Profile', select the expected profile.
- Right-click > Properties.
- In the tab 'Characteristics', set the property 'Command line'.

Example of procedure to set /RW /RO for a user:

1. Run the Windows Administration Console.
2. Open the environment.
3. Select the folder 'User Accounts > Users'.
4. Right-click > Manage.
5. In the tab 'Logins', select the login of the user requested.

- Ex: select the login 'Mega' for the login holder 'Mega'
6. Right-click > Properties.
 7. In the tab 'Characteristics', set the property 'Command line'.

If a value is set at both level, the intersection will be considered for /RW.

If a value is set at both level, the concatenation will be considered for /RO.

Example:

Command line value set for the user (Login Level)	Command line value set for profile	Command line value considered
	/RW'HBPA'	/RW'HBPA'
/RW'DMO;HBPA'	/RW'HBPA'	/RW'HBPA'
/RW'DMO'	/RW'HBPA'	-
/RW'DMO;HBPA' /RO'DBB'	/RW'HBPA' /RO'MTS2'	/RW'HBPA' /RO'DBB;MTS2'

Results:

- The profile is configured to run certain products.
- The user is configured to run certain products.

5.2. Specifying a default licence

If several licences exist in the licence folder, users must be configured explicitly. Otherwise, they cannot login. It is however possible to specify a default licence.

Procedure:

In the file in router.ini, manually add a section [Config] such as:

[Config]

DefaultLicence=<licence name>

Where 'licence name' is the name of the licence file without the .must extention.

Ex: if the file is 'Licence-T0002.must', the licence name is 'Licence-T0002'.

5.3. Declaring users

This is important in several situations:

- Several Must licences exist: users should be allocated in the different licences unless a default licence is specified.
- Shared licence: possible users should be specified beforehand.
- Dedicated licence: named users should be specified beforehand.

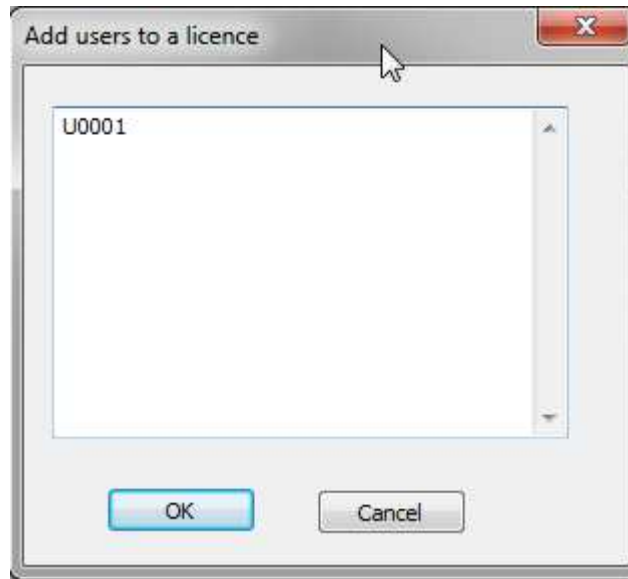
Adding a user to a licence

Prerequisites:

- A Must licence is installed.
- A HOPEX installation is available and configured for Must licences.
- Get user x licence mapping specification. Each user must be declared with its HOPEX login. The users must be able to know the login of each user. Ex: the HOPEX login of John Smith is 'U0001'.

Procedure:

1. Run the Must licence utility.
2. Select the licence folder.
3. Select the licence to be configured. Ex: Licence-T0001.
4. Click the 'Add user' button: enter the login name (Ex: enter 'U0001' for the user 'John Smith is 'U0001') and click 'OK'.


Results:

- The user is displayed in the left pane below the licence (<user login>). Ex: U0001
- The 'Router.ini' configuration file is created in the licence folder for saving this specification. A section [User/Licence] is created or updated.

Example:

```
[User/Licence]
U0001=Licence-T0001
```

Note that you can also enter several login names separated with semicolon or line break (example: U0001;U0002)

Removing a user from a licence

Prerequisites:

- A Must licence is installed.
- A HOPEX installation is available and configured for Must licences.

Procedure:

1. Run the Must licence utility.
2. Select the licence folder.
3. Select the licence to be configured. Example: Licence-T0001.
4. Select the login of the user to be removed.
5. Click the 'Remove user' button.

Results:

- The user is no longer displayed in the left pane below the licence.

5.4. Configuring possible users of products

Setting a user as a possible user of a product

Prerequisites:

- A Must licence is installed in the licence folder.
- A HOPEX installation is available and configured for Must licences.
- Users are declared.
- Get user x product mapping specification.

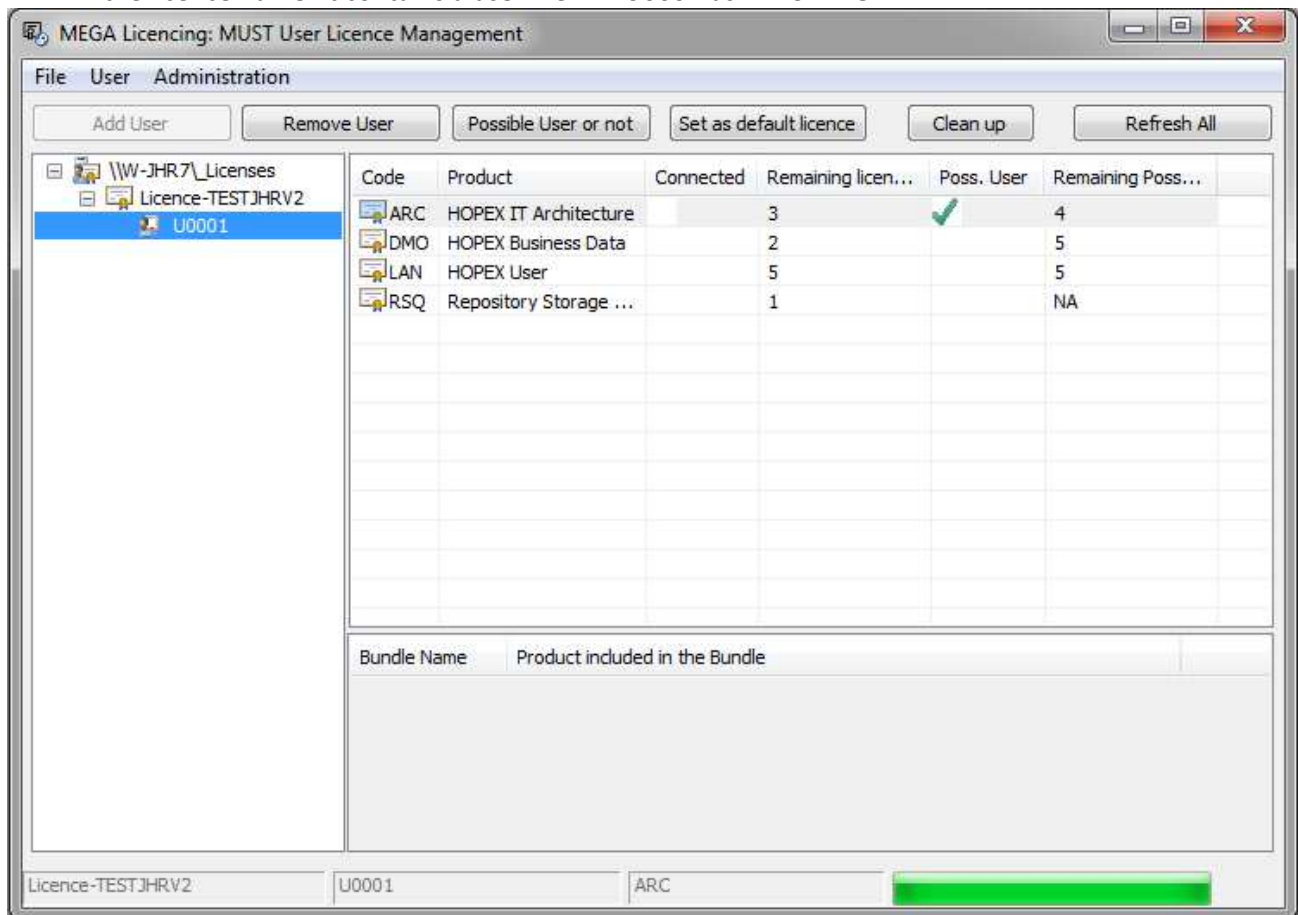
Procedure:

1. Run the Must licence utility.
2. Select the licence folder.

3. Select the licence to be configured.
4. Select the product to be configured.
5. Select the user to be set as a possible user of the product.
6. Click the 'Possible user or not' button.

Results:

- When both user and product are selected, a checkbox is displayed in the 'Poss. User' column of the top right pane. It shows that the current user is a possible user for the product.
- Files are created in the licence folder for saving this specification. A subfolder 'USERS' exists with the licence name. It contains a user file. Ex: U0001.usr-ARC-MEGA.



Removing a user as a possible user of a product

Prerequisites:

- A Must licence is installed in the licence folder.
- A HOPEX installation is available and configured for Must licences.
- Users are mapped to licences.

Procedure:

1. Run the Must licence utility.
2. Select the licence folder.
3. Select the licence to be configured.
4. Select the product to be configured.
5. Select the user to be removed as a possible user of the product.
6. Click the 'Possible user or not' Button.

Results:

- When both user and product are selected, no checkbox is displayed in the 'Poss. User' column of the top right pane.

- Files are updated in the licence folder for saving this specification.

Cleaning up licence tokens

Prerequisites:

- A Must licence is installed in the licence folder.
- A HOPEX installation is available and configured for Must licences.
- Verify with the Must licence utility that no user is currently logged on to the licence to be cleaned.

Procedure:

1. Run the Must licence utility **as Administrator**.
2. Select the licence folder.
3. Select the licence to be configured.
4. Click the 'Clean up' Button.

Results:

- Unexpected token files are purged.

Note that, if a lot of token files exist when the button 'Clean up' is first clicked, the processing can take several minutes according to the number of token files, the number of licences and the file access performances. The token files are purged for each licence displayed in the left pane. The processing will run faster the next times button 'Clean up' is clicked (as most token files will have been purged).

5.5. Instant monitoring of licence connections

The Must licence utility can be used to monitor connections even though it was not designed for this purpose. Display may be slow.

Prerequisites:

- A Must licence is installed in the licence folder.
- A HOPEX installation is available and configured for Must licences.

Procedure:

1. Run the Must licence utility.
2. Select the licence folder.
3. Select the licence to be monitored.
4. Select the user or the product to be monitored.
5. Read the top right pane, column 'Connected'.

6. CONFIGURING LICENSING MODES AND USER TYPES

In certain situation, the same product can be programmed in several licensing modes.

Ex:

For HOPEX IT Architecture:

- 5 access in **shared** mode
- 10 access in **concurrent** mode (floating mode)
- 5 access in **dedidated** mode

Command line parameters will be used to control user tokens (LAN_X tokens) delivery.

By default the product named LAN is used in shared mode.

There can also be a product named LAN_F for concurrent mode.

There can aslo be a product named LAN_D for dedicated mode.

The general steps will be:

- Check that .Must licence is programmed in the expected mode.
- Enable extended mode in megasite.ini.
- Configure command line.

Independantly from the products/solutions, there are tree exclusive types of users:

- **Viewer users:** consult data, search, use collaboration features. VIEW counter is used.
- **Contributor users:** consult data, search, perform limited updates, use collaboration features. CBTR counter is used.
- **Main users:** consult data, search, perform all updates in particular with diagram editor, use collaboration features. LAN_X counters are used.

Command line parameters will be used to control user tokens delivery.

- By default, LAN counter is used if no possible user is configured.
- To user viewer users, a specific command line should be used.
- To user contributor users, a specific command line should be used.

6.1. Configuration of main users with concurrent mode (floating mode)

Check .Must licence

By convention, a product programmed in concurrent mode will use the LAN_F counter for main users. Other products must be programmed in concurrent mode.

This can be checked in the licence file description: the second digit equals 0 (except for LAN_X counter).

Extract of licence description	comment
[MEGAComponentInfo]	
(LAN) HOPEX MainUser=3 ; 0	Counter of main users (shared mode)
(RSO) Repository Storage (ORACLE)=NO	-
(RSQ) Repository Storage (SQL Server)=YES	-
(DMO) HOPEX Logical Data=3 ; 5	Programmed in shared mode
(SUP) HOPEX Power Supervisor=1 ; 1	Programmed in dedicated mode
(APM) HOPEX IT Portfolio Management=1 ; 1	Programmed in dedicated mode
(ANW) Web Front-End=NO	-
(HPP) HOPEX Productivity Pack=NO	-
(HBPA) HOPEX Business Process Analysis=3 ; 3	Programmed in dedicated mode
(CBTR) HOPEX Contributor=1 ; 0	Counter of contributor users
(VIEW) HOPEX Viewer=1 ; 0	Counter of view users
APM_F=5 ; 0	Programmed in concurrent mode
LAN_D=5 ; 0	Counter of main users (dedicated mode)
LAN_F=3 ; 0	Counter of main users (concurrent mode)
[MEGABundleInfo]	
APM_F=APM	
LAN_D=LAN	
LAN_F=LAN	

Note that the extension _F is conventional (except for LAN_F). Although it is not called APM_F, APM is programmed in concurrent mode.

Configure command line

This property 'Command line' exists at login level and profile level.

Use the /RW syntax and quote product codes programmed in concurrent mode.

Ex: /RW'LAN_F,APM_F'

Reminders:

- It is recommended to configure command line for profiles rather than for logins
- Standard profiles are protected, it is recommended to create a custom profiles that inherits from a standard profile.
- Changing command lines property will reset technical data cache. A warning 'The technical data are not compiled...' will be displayed as long as technical data are not recompiled.

6.2. Configuration of main users with dedicated mode

Check .Must licence

By convention, a product programmed in dedicated mode will use the LAN_D counter for main users.

Other products must be programmed in dedicated mode.

This can be checked in the licence file description: the second digit equals the first one (except for LAN_X counter).

Extract of licence description	comment
[MEGAComponentInfo]	
(LAN) HOPEX MainUser=3 ; 0	Counter of main users (shared mode)
(RSO) Repository Storage (ORACLE)=NO	-
(RSQ) Repository Storage (SQL Server)=YES	-
(DMO) HOPEX Logical Data=3 ; 5	Programmed in shared mode
(SUP) HOPEX Power Supervisor=1 ; 1	Programmed in dedicated mode
(APM) HOPEX IT Portfolio Management=1 ; 1	Programmed in dedicated mode
(ANW) Web Front-End=NO	-
(HPP) HOPEX Productivity Pack=NO	-
(HBPA) HOPEX Business Process Analysis=3 ; 3	Programmed in dedicated mode
(CBTR) HOPEX Contributor=1 ; 0	Counter of contributor users
(VIEW) HOPEX Viewer=1 ; 0	Counter of view users
APM_F=5 ; 0	Programmed in concurrent mode
LAN_D=5 ; 0	Counter of main users (dedicated mode)
LAN_F=3 ; 0	Counter of main users (concurrent mode)
[MEGABundleInfo]	
APM_F=APM	
LAN_D=LAN	
LAN_F=LAN	

Note that the extension _D is conventional (except for LAN_D). Although it is not called HBPA_D, HBPA is programmed in dedicated mode.

Configure command line

This property 'Command line' exists at login level and profile level.

Use the /RW syntax and quote product codes programmed in concurrent mode.

Ex: /RW'LAN_D,HBPA,APM'

Note that it is not required to quote all product codes programmed in dedicated mode in the command line. Here SUP, is not quoted because it is assigned to a specific login/profile.

Reminders:

- It is recommended to configure command line for profiles rather than for logins
- Standard profiles are protected, it is recommended to create a custom profiles that inherits from a standard profile.
- Changing command lines property will reset technical data cache. A warning 'The technical data are not compiled...' will be displayed as long as technical data are not recompiled.

6.3. Configuration of main users with shared mode

Check .Must licence

By convention, a product programmed in dedicated mode will use the LAN counter for main users.

Other products must be programmed in dedicated mode.

This can be checked in the licence file description: the second digit is greater than the first one (except for LAN_X counter).

Extract of licence description	comment
[MEGAComponentInfo] (LAN) HOPEX MainUser=3 ; 0 (RSO) Repository Storage (ORACLE)=NO (RSQ) Repository Storage (SQL Server)=YES (DMO) HOPEX Logical Data=3 ; 5 (SUP) HOPEX Power Supervisor=1 ; 1 (APM) HOPEX IT Portfolio Management=1 ; 1 (ANW) Web Front-End=NO (HPP) HOPEX Productivity Pack=NO (HBPA) HOPEX Business Process Analysis=3 ; 3 (CBTR) HOPEX Contributor=1 ; 0 (VIEW) HOPEX Viewer=1 ; 0 APM_F=5 ; 0 LAN_D=5 ; 0 LAN_F=3 ; 0 [MEGABundleInfo] APM_F=APM LAN_D=LAN LAN_F=LAN	Counter of main users (shared mode) - - Programmed in shared mode Programmed in dedicated mode Programmed in dedicated mode - - Programmed in dedicated mode Counter of contributor users Counter of view users Programmed in concurrent mode Counter of main users (dedicated mode) Counter of main users (concurrent mode)

Configure command line

This property 'Command line' exists at login level and profile level.

Use the /RW syntax and quote product codes programmed in concurrent mode.

Ex: /RW'LAN,DMO'

Reminders:

- It is recommended to configure command line for profiles rather than for logins
- Standard profiles are protected, it is recommended to create a custom profiles that inherits from a standard profile.
- Changing command lines property will reset technical data cache. A warning 'The technical data are not compiled...' will be displayed as long as technical data are not recompiled.

6.4. Configuration for viewer users

By convention, a product programmed in dedicated mode will use the VIEW counter for main users. Check the licence.

Extract of licence description	comment
[MEGAComponentInfo]	
(LAN) HOPEX MainUser=3 ; 0	Counter of main users (shared mode)
(RSO) Repository Storage (ORACLE)=NO	-
(RSQ) Repository Storage (SQL Server)=YES	-
(DMO) HOPEX Logical Data=3 ; 5	Programmed in shared mode
(SUP) HOPEX Power Supervisor=1 ; 1	Programmed in dedicated mode
(APM) HOPEX IT Portfolio Management=1 ; 1	Programmed in dedicated mode
(ANW) Web Front-End=NO	-
(HPP) HOPEX Productivity Pack=NO	-
(HBPA) HOPEX Business Process Analysis=3 ; 3	Programmed in dedicated mode
(CBTR) HOPEX Contributor=1 ; 0	Counter of contributor users
(VIEW) HOPEX Viewer=1 ; 0	Counter of view users
APM_F=5 ; 0	Programmed in concurrent mode
LAN_D=5 ; 0	Counter of main users (dedicated mode)
LAN_F=3 ; 0	Counter of main users (concurrent mode)
[MEGABundleInfo]	
APM_F=APM	
LAN_D=LAN	
LAN_F=LAN	

Configure command line

This property 'Command line' exists at login level and profile level.

Use the /HV syntax and quote product codes programmed in concurrent mode.

Ex: /HV'APM'

6.5. Configuration for contributor users

By convention, a product programmed in dedicated mode will use the VIEW counter for main users. Check the licence.

Extract of licence description	comment
[MEGAComponentInfo]	
(LAN) HOPEX MainUser=3 ; 0	Counter of main users (shared mode)
(RSO) Repository Storage (ORACLE)=NO	-
(RSQ) Repository Storage (SQL Server)=YES	-
(DMO) HOPEX Logical Data=3 ; 5	Programmed in shared mode
(SUP) HOPEX Power Supervisor=1 ; 1	Programmed in dedicated mode
(APM) HOPEX IT Portfolio Management=1 ; 1	Programmed in dedicated mode
(ANW) Web Front-End=NO	-
(HPP) HOPEX Productivity Pack=NO	-
(HBPA) HOPEX Business Process Analysis=3 ; 3	Programmed in dedicated mode
(CBTR) HOPEX Contributor=1 ; 0	Counter of contributor users
(VIEW) HOPEX Viewer=1 ; 0	Counter of view users
APM_F=5 ; 0	Programmed in concurrent mode
LAN_D=5 ; 0	Counter of main users (dedicated mode)
LAN_F=3 ; 0	Counter of main users (concurrent mode)
[MEGABundleInfo]	
APM_F=APM	
LAN_D=LAN	
LAN_F=LAN	

Configure command line

This property 'Command line' exists at login level and profile level.

Use the /HC syntax and quote product codes programmed in concurrent mode.

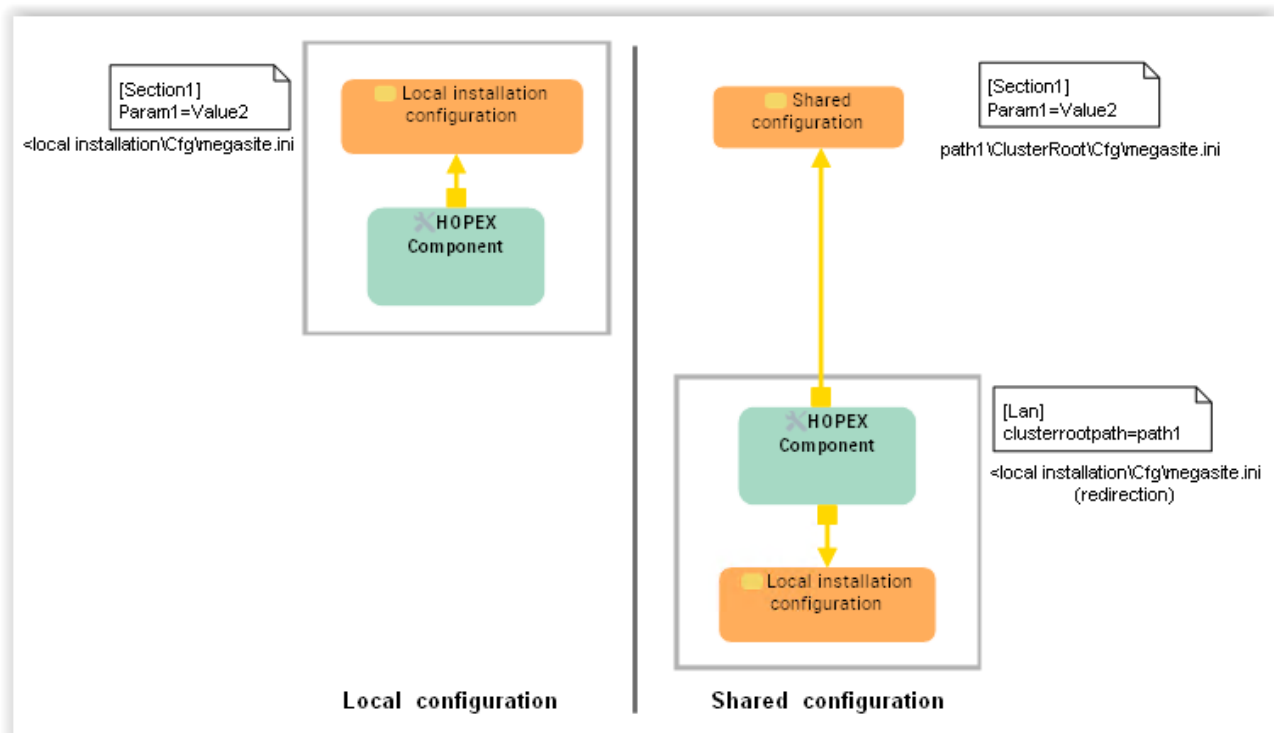
Ex: /HC'APM'

7. INSIDE

7.1. Licence deployment model

Two types of references can be used:

Configuration	Example of Megasite.ini	Comment
Local configuration	[Must licence] Path=\\server001\Apps\Licences	The Must licence folder (\\server001\Apps\Licences) is configured in the file Megasite.ini of the HOPEX Installation
Shared configuration	[Lan] clusterrootpath=\\mega\data	The file Megasite.ini of the HOPEX installation refers to the root of a shared configuration folder (ex: \\mega\data\ClusterRoot\Cfg) containing another file Megasite.ini. This file contains a direct reference to the Must licence folder (\\server001\Apps\Licences). This type of reference must be used for Citrix/TSE deployment



When HOPEX is run by user U0001:

1. A configuration file (megsite.ini) is read to identify the licence folder.
2. The licence folder can be referenced directly (local configuration) or indirectly (shared configuration).
3. The licence folder is read to identify the authorized licence file for this user.
4. The Must licence configuration is read to identify the products authorized for this user.
5. Connection is made if tokens are available for the authorized products.

7.2. Licence execution

Licence execution is homogenous through Front-Ends or Windows Administration Console.

Windows Front-End

Context	Must licence checked	Storage product checked (1)	Tokens requested	Command line considered
HOPEX.exe (main user)	Yes	Yes (2)	One token per Product One token LAN_X	Yes
Administration.exe	Yes	Yes (2)	One token SUP One token LAN_X	No
HOPEX.exe with HOPEX Power Studio (MTS2)	Yes	Yes (2)	One token MTS2 One token LAN_X	Yes
API component (3)	Yes	Yes (2)	One token per Product One token LAN_X	Yes

(1) RSQ or RSO

(2) Unless GBMS storage is used.

(3) Administration component creating a running instance of HOPEX (mgwmapp.exe)

Web Front-End

Context	Must licence checked	Counter used	Tokens requested	Command line considered
HOPEX Product (multi front-end)	Yes (1)	LAN_X	One token per Product One token LAN_X	Yes
HOPEX Product (controlled multi front-end)	Yes (1)(2)	LAN_X	One token per Product One token LAN_X	Yes
HOPEX Solution	Yes (1)	LAN_X	One token per Solution One token LAN_X	Yes
SSP component	Yes (1)	-	No token (3)	No
HOPEX Explorer	Yes (1)	LAN_X	One token for HEXP One token LAN_X	Yes
HOPEX Contributor	Yes	CBTR_X	One token for CBTR_X	Yes
HOPEX Viewer	Yes	VIEW_X	One token for VIEW_X	Yes

(1) RSQ or RSO product should be programmed.

(2) ANW product should be programmed.

When running the SSP component, must licence is checked but no token is requested. SSP is used systematically with HOPEX Web Front-End. It can also be used by Windows Front-End.

7.3. File access

File access in the licence folder should be similar to that of accessing a HOPEX repository data file. Contact MEGA Support if you encounter problems.

8. FAQs AND TROUBLESHOOTING

8.1.1. How can I use the Web licensing console?

See online documentation.

Note that the Web licensing console

- Needs to be installed when installing HOPEX Web Front-End.
- Needs to be allowed to the administrator in charge (security configuration).
- Does not allow to see user of tokens by login.

8.1.2. Do I have to configure possible users?

This is not required. If products are programmed in shared mode and the command line is configured, you do not need to explicitly configure possible users.

When user U0001 logs in, a token is requested for each product mentioned in the command line. If possible, user seats are available, U0001 is automatically configured as a possible user of the requested products. If tokens are available, U0001 can log in to these products.

8.1.3. How can I secure configuration of the HOPEX Must licence?

If you do not want to configure systematically full control, you may configure advanced file permissions:

File	Location	Administrator rights	User rights
*.must	Example: \\server001\Apps\Licences\Licence-001.must	Modify	Read & execute
.	Licence subfolders containing the user files and token files. Example for Licence-001.must: \\server001\Apps\Licences\Licence-001 and subfolders	Modify	Modify ¹

The users considered vary with the front-end.

Front-end	Comment
HOPEX Web Front-end	Only the service account for the HOPEX (IIS) web application should be configured
Windows Front-end	Each end-user and administrators should be configured (ex: D01\U0001)

8.1.4. How can I prevent the dynamic declaration of possible users?

There is no way of preventing a user who is not explicitly configured from logging in.

If a possible user seat is available, the system will set a user requesting a token as a possible user. For this reason, it is recommended you configure possible users beforehand.

8.1.5. How can I get a log of licence connections?

Supervision logs contain information regarding connection and disconnection. However, this is technical information and MEGA does not provide a report to consolidate and display this information.

8.1.6. What is the Router.ini file?

This file contains the mapping of users to licences.

It is updated when configuration is made using the Must licence utility ('Add user' and 'Remove user' actions). Deletion of this file will not remove possible user configurations.

¹ By default, user files and token files are set as 'Not visible'.

8.1.7. How can I get the assignment of users to licences?

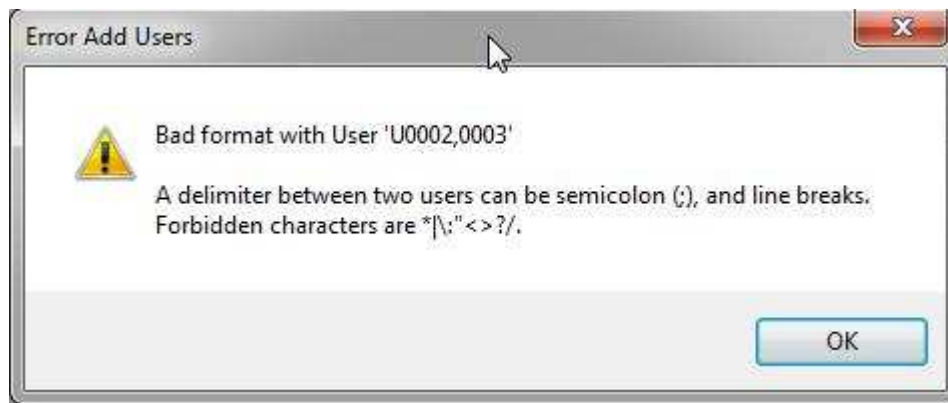
No report is available. You can consult the Must licence utility (left pane) or read the Router.ini file.

8.1.8. How can I get the assignment of possible users to product?

No report is available. Consult the Must licence.

8.1.9. When adding a user, I get an error 'Bad format with user 'XX'!

This is because the data entered does not match the format expected.



8.1.10. How can I get the list of logins of users?

No report is available. Consult the Windows Administration Console (Administration.exe).

8.1.11. I do not know the names of the logins. Why can't the utility provide a list of existing login?

This is a design option. The list of existing logins is related to a HOPEX environment

This would require to login to a HOPEX environment which is not in the scope of the licensing utility.

8.1.12. How can I set possible users for a selection of users?

It is not possible to select multiple users in the left pane of the Must licence utility. However, a specific operating mode enables the administrator to replicate the possible user configurations of products on a licence for other users.

Procedure:

1. Run the Must licence utility.
2. Select the licence folder.
3. Select the licence to be monitored in the left pane.
4. Select a user for this licence.
5. Configure possible users of the different products in the top right pane:
 - o Select the products to be configured.
 - o Click the 'Possible User or not' button.
6. Select another user in the left pane: the same list of products is selected.
7. Click 'Possible User or not': the same configurations are replicated on the products selected.

8.1.13. Can I mix shared and dedicated modes?

Yes. Note that modes are set at the product level.

8.1.14. Is my licence shared or dedicated?

To check the licensing mode:

1. Open the .Must licence file with a text editor such as Notepad
2. Analyze configuration: the mode depends on the combination of 2 digits

<Licence Product>=T ; U

Where:

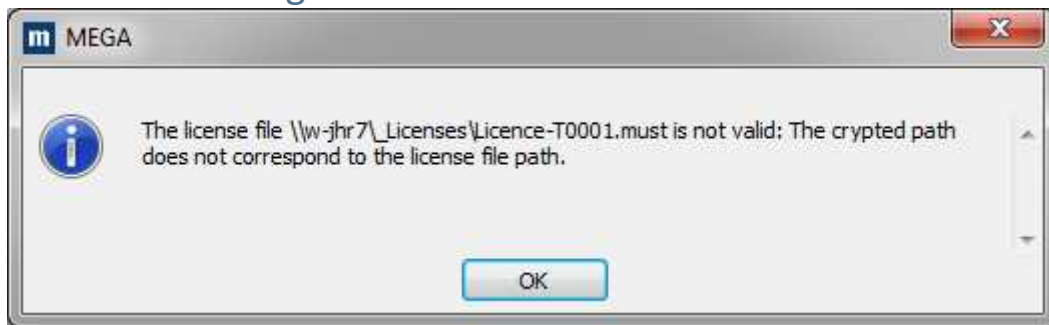
T: tokens

U: users

Licensing mode	Example
Dedicated mode (T=U)	(HITA) HOPEX IT Architecture=20 ; 20
Shared mode (T< U)	(HITA) HOPEX IT Architecture=20 ; 25
Concurrent mode/floating mode (T>U, U=0)	(HITA) HOPEX IT Architecture=20 ; 0

Note that modes are set at the product level.

8.1.15. Error message 1



Possible reasons:

- The path of the folder containing the Must licence file does not match the path programmed in the licence.
- The licence file name does not match the file name programmed in the licence (licence file was renamed).

8.1.16. Where is the latest licence folder used saved?

With Windows Front-End, it is saved in the user settings file (MEGASETTINGS.INI) in a section [MGWUSRMNG].

Example:

```
[MGWUSRMNG]
```

```
LastServerPath=\\server001\Apps\Licences
```

8.1.17. How can I check that an HOPEX installation is part of a cluster?

Check the file Megasite.ini of the installation. It must contain a section [Lan] and a variable clusterrootpath.

Example:

```
[Lan]
```

```
clusterrootpath=\\mega\data
```

8.1.18. A message is displayed like You are not allowed to launch HOPEX V3 with your licence file “xxxx.must”. It must be upgraded for this version. Please contact your sales representative to obtain a valid licence file.

This is a standard warning with HOPEX V3.

Licences generated for lower versions (MEGA 2009, HOPEX V1R1) are not compatible with higher versions. Please contact your sales representative.

HOPEX UNIFIED AUTHENTICATION SERVICE

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

Contents

1. INTRODUCTION	5
1.1. Unified Authentication Service Overview	5
1.2. What is SAML2 ?	5
1.2.1. How SAML Works ?	5
1.2.2. Profiles	6
1.2.3. Security	6
1.3. What is Oauth2 and OpenID?	7
1.3.1. Description of the OAuth2 protocol	7
1.3.2. The notion of token	7
1.3.3. Client registration	7
1.3.4. OpenID Connect protocol description	9
1.3.5. The notion of Token ID	9
1.3.6. Sample Token ID:	10
1.3.7. The notion of Authorization Flow	11
1.3.8. Focus on JWT	12
2. UAS OPTIONS CONFIGURATION	14
2.1. Configuring authentication options	14
2.2. Server Option Description	15
2.2.1. Server global options	15
2.2.2. Authentication options	16
2.2.3. Cookie options	17
2.2.4. Events options	17
2.2.5. Logging trace options	18
2.2.6. Token Signature options	18
2.3. Identity Provider Option Description	18
2.3.1. Google, see Open ID Connect (OIDC) provider	19
2.3.2. HOPEX provider	19
2.3.3. IIS Windows provider	19
2.3.4. SAML2 provider	19
2.3.5. Open ID Connect (OIDC) provider	19
2.3.6. Google provider	20
2.3.7. Microsoft provider	20
2.3.8. Salesforce provider	20
2.3.9. Custom provider	21
2.4. Cross-Origin Resource Sharing Option Description	21
2.4.1. Simple requests	22
2.4.2. Preflighted requests	23
2.4.3. Use within browsers	23
Error detection	26
2.5. Client Option Description	29
2.5.1. HOPEX Custom (options in Extended view only)	29
3. UAS API ENDPOINTS	31
3.1. Authorization/Authentication	31
3.2. Token	32

3.3.	UserInfo	33
3.4.	Discovery Endpoint.....	33
3.5.	Logout Endpoint	33
3.6.	Token Revocation	34
3.7.	Introspection Endpoint.....	34
3.8.	Access token validation endpoint.....	35
3.9.	Identity Token Validation Endpoint.....	35
3.10.	CSP Endpoint	36
4.	ESTABLISH AN SSL CONNECTION.....	37
4.1.	Creating a certificate request from IIS	37
4.2.	Completing the certificate request.....	38
4.3.	Binding IIS with SSL certificate.....	38
4.4.	Exporting certificate to the local disk.....	39
5.	INSTALL HOPEX SIGNING CERTIFICATE (MANUALLY).....	40
6.	CONFIGURE UAS HOPEX BY OPTIONS	41
6.1.	Local Configuration.....	41
6.1.1.	Defining authentication options.....	41
6.2.	Cluster Configuration.....	41
6.2.1.	Configuring your data component type	41
6.2.2.	Configuring your SQL Server Data component type.....	41
6.2.3.	Configuring Hopex Web options	42
6.2.4.	Generating Machine Key	42
7.	CONFIGURE CLIENT USING UAS.....	43
8.	STANDALONE MODE.....	44
9.	ANONYMOUS ENVIRONMENT MODE	45
10.	SAML2 ADFS SERVER CONFIGURATION.....	46
11.	WINDOWS AUTHENTICATION IN CLUSTER MODE WITH UAS	53
12.	OKTA CONFIGURATION	54
12.1.	Configuring OKTA	54
12.2.	Configure UAS with OKTA.....	54
13.	TERMINOLOGY.....	56
13.1.	Client.....	56
13.2.	User.....	56
13.3.	Scope	56
13.3.1.	Identity scopes.....	56
13.3.2.	Resource scopes	56
13.4.	Authentication/Token Request	56
13.4.1.	Identity Token.....	56
13.4.2.	Access Token	57
14.	PROTOCOL SPECIFICATIONS.....	58
15.	TROUBLESHOOTING.....	59

15.1. General	59
15.2. Get information about configuration	59
15.3. Redirect Server name to Full Qualified Domain name.....	59
15.4. Client configuration in Windows Authentication mode.....	59
15.5. Filtering Windows group	60
15.5.1. Filtering Windows group by number.....	60
15.5.2. Filtering Windows group by name	60

1. INTRODUCTION

1.1. Unified Authentication Service Overview

Authentication is a common basic requirement for modern web-based applications as more and more customized and access controlled services move it online.

Cloud applications have become popular among organizations that share content on the Internet.

Most of the organizations have started using a centralized authentication source for their web portal, web and mobile applications.

Most of the web and mobile applications have their own authentication system to provide a better user experience. To support that need, one of our goals is to provide you with a clear framework that allows you to develop secure web based authentication system easily.

We want to provide the most secure and scalable platform on our market.

With the **Unified Authentication Service (UAS)**, you have a centralized service for your login logic and workflows in a single and well secured place.

It allows managing single sign-on (SSO) (and out) over multiple application types like web or mobile, access control for APIs and federation (support for external identity providers like Google and enterprise identity management systems via SAML2.)

We manage two standard authentication protocols: SAML2 and Open ID

1.2. What is SAML2 ?

Security Assertion Markup Language 2.0 (SAML 2.0) is a version of the SAML standard for exchanging authentication and authorization data between security domains.

SAML 2.0 is an XML-based protocol that uses security tokens containing assertions to pass information about a principal (usually an end user) between a SAML authority, named an Identity Provider, and a SAML consumer, named a Service Provider.

SAML 2.0 enables web-based, cross-domain single sign-on (SSO), which helps reduce the administrative overhead of distributing multiple authentication tokens to the user.

SAML 2.0 was ratified as an OASIS Standard in March 2005, replacing SAML 1.1.

1.2.1. How SAML Works ?

SAML SSO works by transferring the user's identity from one place (the identity provider) to another (the service provider). This is done through an exchange of digitally signed XML documents.

Consider the following scenario: A user is logged into a system that acts as an identity provider. The user wants to log in to a remote application, such as a support or accounting application (the service provider). The following happens:

- The user accesses the remote application using a link on an intranet, a bookmark, or similar and the application loads.

- The application identifies the user's origin (by application subdomain, user IP address, or similar) and redirects the user back to the identity provider, asking for authentication. This is the authentication request.
- The user either has an existing active browser session with the identity provider or establishes one by logging into the identity provider.
- The identity provider builds the authentication response in the form of an XML-document containing the user's username or email address, signs it using an X.509 certificate, and posts this information to the service provider.
- The service provider, which already knows the identity provider and has a certificate fingerprint, retrieves the authentication response and validates it using the certificate fingerprint.
- The identity of the user is established, and the user is provided with app access.

1.2.2. Profiles

The most common profile, called the "Web Browser SSO", describes, among other things, the steps of authenticating a user and going back and forth between the service provider (SP) and the Identity Provider (IdP).

The user tries to access his protected resource by the SP.

The SP verifies that the user is authenticated and if he is not authenticated, redirects him to his IdP.

The IdP asks the user to authenticate (identifier and password for example) and returns a SAML assertion to the SP containing the identity of the user and the guarantee that it is authenticated. The SP then allows the user to access the resource initially requested.

This authentication mechanism is based on Internet browser redirects. This profile also makes it possible to recover a set of additional attributes related to the identity of the user and requested by the resource.

A second artifact-based profile decorrelates the authentication of the retrieval of the user's identity information. The SP receives from the IdP, through the user's Internet browser, a SAML assertion containing an artifact. The SP must then directly query the IdP for information related to the identity of the user.

Other profiles describe how to implement the DS, the notions of logout and the ability to dispense with the user's browser to transmit SAML assertions between services.

1.2.3. Security

SAML assertions are based on SOAP, XML Encryption, and XML Signature layers.

SOAP is the standard encapsulation protocol for XML messages, used primarily by Web services. XML Encryption is the standard protocol for encrypting XML messages.

It has the distinction of being able to encrypt the whole message or just a specific subset. This makes it possible, for example, to have an XML document in clear with encrypted attribute values. XML Signature is the standard protocol for signing XML messages. Just like XML Encryption it allows to target the element to sign. This allows multiple stakeholders to sign each different part of the XML document.

The SP and the IdP are two entities that are each aware of each other in terms of identifier and certificate. The XML messages that pass through the network are therefore encrypted by the public key of the recipient, only able to decrypt the message with his private key. The issuer signs his assertions with his private key allowing the recipient to verify his provenance.

1.3. What is OAuth2 and OpenID?

Today OpenID Connect has been adopted by all major players in the web such as Google, Facebook, Salesforce, or Microsoft, as well as by any organization wishing to implement a centralized identity federation and respond to SSO issues. Let's look in more detail at the principles and concepts on which this protocol is based, how it is implemented, and what technologies are used. OpenID Connect (OIDC) specifies an HTTP Restful authentication interface and relies on the OAuth2 protocol to do delegation authorization, ie in the vast majority of cases, the end user will no longer have need to directly provide credentials to a third-party application. OIDC also uses the JSON Web Token (JWT) exchange formalism to convey user identities to applications, as well as their roles / entitlements. In this article we will focus on how to manage and secure this relationship of trust between users and applications, while ensuring data integrity.

1.3.1. Description of the OAuth2 protocol

OAuth2 specifies an access delegation protocol (<https://tools.ietf.org/html/rfc6749>). Its main purpose is to describe how access to secure APIs of an application or a website (provider) will be delegated to another application (consumer). The protocol distinguishes 4 main roles:

- Resource Owner: the one who holds the resources
- Resource Server: server that hosts protected resources
- Client: client application (front, back or mobile) that requests access to resources
- Authorization Server: A server that generates tokens for the client and will be passed on when queries are made to the resource server.

1.3.2. The notion of token

- **Access Token:**
Token to validate access to a secure service (an authorization) with a defined lifetime. It is essential.
- **Refresh Token:**
A long-term "renewal" token to request the creation of a new Access Token if it has expired. It is issued at the same time as the Access Token but is not sent to each request. The Refresh Token must be stored by the client application in a secure manner.

1.3.3. Client registration

It should be noted that as part of the OAuth2 protocol, each client application that wants to access protected resources must first register with the authorization server (usually via a form).

The OAuth2 specification specifies the standard parameters that clients must fill in during the registration process:

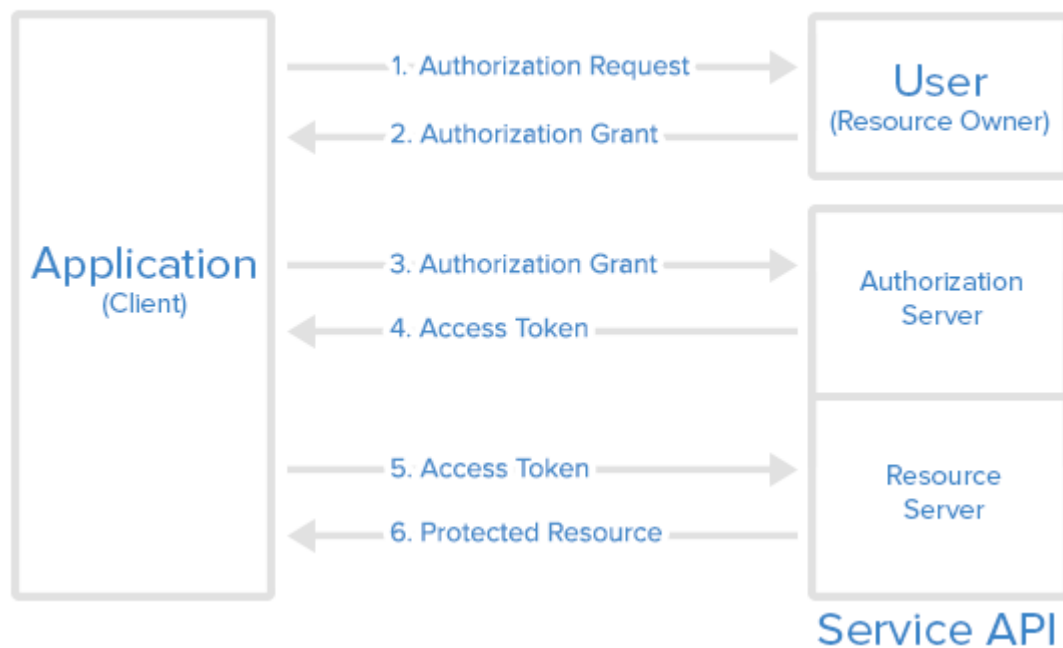
- Application Name: name of the application.
- Redirect URI (or Callback URL): The URI (or URL) of the client application to which redirections will be made by the authorization server, once access to the authorized resources (or when access is denied).
- Grant Type (s): Types of authorization that can be used by the client when requesting protected resources.

The authorization server delivers a couple `client_id` / `client_secret`

- `client_id`: randomly generated character string that uniquely identifies a client application.
- `client_secret`: string representing the secret client key that will be used when calling certain APIs that require an HTTP Authorization header.

The request flow for access to secure resources can therefore be represented in the following generic manner:

Abstract Protocol Flow



1. The client application sends a request for access to the protected resources of the user, specifying in particular his identity (`client_id`), the type of authorization and the scopes desired. The scopes are determined by the authorization server beforehand. The more the API is cut into small scopes, the more the profile of each client is accurate (and therefore limited). Each client application knows only the scopes it can use.
2. If the user approves the request, an access right is returned to the client.
3. The client then requests an Access Token from the token server by providing its identity (eg client credentials), as well as its previously received access right.
4. If the identity is validated (the client is authenticated) and the access right is valid, the token server issues an Access Token.

5. The client can then request access to the protected resources to the resource server by presenting its Access Token.
6. If the provided Access Token is valid, the requested resources are returned to the client.

Note that the specification OAuth2 requires that all these exchanges take place in HTTPS. Depending on the type of authorization specified by the client, this schema will be implemented in different ways. In the OAuth2 specification, there are 4 types of permissions:

- **Authorization Code:** Used if the client application is located on the server side. Most implemented case.
- **Implicit:** used if the client application is located on the client side (eg a Javascript application or a mobile application) and no other type of authorization is available. This mode is less secure because the token does not stay on the server side but is exposed on the client side and can be intercepted.
- **Resource Owner Credentials:** Login credentials are sent to the client and then to the authorization server. This implies that there is absolute trust between the two. Often used when the client has been developed by the same entity that provides the authorization server (eg access to secure resources of a subdomain), this type of authorization is strongly discouraged because OAuth has been thought just so that the login credentials are no longer transmitted to third-party applications. In addition, there is no verification of the callback URL.
- **Client Credentials:** Used when the client is the owner of the data. There is therefore no specific authorization to obtain from the user. The exchanges begin directly in step 3. It is important to remember that OAuth2 does not handle authentication or even authorization. At no time is there any user information, roles or entitlements. In order to have a complete identity solution, it is necessary to use the OpenID Connect protocol.

1.3.4. OpenID Connect protocol description

OpenID Connect is a protocol that is gaining popularity because it is an overlay to OAuth2 (it is able to respond to all of its use cases), and adds new features that were missing from OAuth2:

- Support for authentication,
- The notion of Token ID,
- Management of the SSO session (eg the Single Logout),
- A new API to retrieve user information (User Info endpoint),
- Standardization of user information,
- An OpenID server discovery system to allow customers to register on their own.

1.3.5. The notion of Token ID

A Token ID is a self-bearing token that contains the identity of a user. It is conveyed in JWT format and consists mainly of:

- Authentication settings:
 - o expiration date,

- creation date,
- date of authentication,
- control means for validating the Token ID and the Access Token.
- Accreditations (roles, authorizations) of the user:
 - formalism to be determined by the identity provider.
 - attributes (claims) of the user.

Attributes are associated with "scopes":

- standard attributes:
 - scope profile: name, surname, nickname, date of birth, ...
 - scope email: email, verified email
 - scope address: address
 - scope phone: phone number, verified phone number
- private attributes
 - attributes offered by the identity provider. It is necessary to specify them in order to avoid any collision with existing claims.

1.3.6. Sample Token ID:

```
{
  "iss": "http://server.meritis.fr",
  "sub": "24400320",
  "aud": "s6BhdRkqt3",
  "nonce": "n-0S6_WzA2Mj",
  "exp": 1515604697,
  "iat": 1515593897,
  "name": "Matthieu Mabyre",
  "given_name": "Matthieu",
  "family_name": "Mabyre",
  "gender": "male",
  "email": "matthieu.mabyre@meritis.fr",
  "acr": ["role1", "role2", "role3"]
}
```

The meaning of the different fields can be found on the official documentation (http://openid.net/specs/openid-connect-core-1_0.html#IDToken).

OpenID Connect offers several interfaces (endpoints):

- authorization: to authenticate a user
- token: to request a token (access / refresh / ID)
- user info: to retrieve information about the user (his identity, his rights)
- revocation: to delete a token (access / refresh)
- introspection: to validate a token (access / refresh).
- Other optional interfaces are available for client registration, discovery of OpenID Connect providers, and more.

1.3.7. The notion of Authorization Flow

OpenID Connect offers three algorithms to determine how to return tokens:

Authorization Code Flow:

The algorithm returns an authorization code and then retrieves tokens:

- tokens are returned only through the token interface,
- the recovery of an access token is done in two steps:
 - o a code is returned by the authorization interface,
 - o this code is sent by the client to the token interface.
- The customer must be registered with the OpenID provider (via an identifier and a secret),
- applies very well to mobile, web and back-end applications,
- most implemented algorithm.

Implicit Flow:

The algorithm directly returns the tokens.

- Tokens are returned directly by the authorization interface (the token interface is no longer used),
- There is no customer registration,
- There is no notion of Refresh Token,
- Long life tokens are not allowed,
- Algorithm for Javascript type applications (without back-end).

Hybrid Flow:

Hybrid Flow is a mix between Authorization Code Flow and Implicit Flow.

- the sequencing is identical to the Authorization Code Flow except that the authorization interface can return the code, the Token ID and the Access Token,
- Refresh Token is obtained by a call to the token interface,
- Algorithm very little used.

1.3.8. Focus on JWT

JWT (JSON Web Token) is a standard, secure information exchange format (<https://tools.ietf.org/html/rfc7519>).

A JWT token is broken down into three parts, separated by the character.

The header

(JOSE header) describes the algorithm used to sign or encrypt the token. For example, for a signature using the HS256 algorithm (HMAC with SHA-256), we would have: {"alg": "HS256", "typ": "JWT"} The header must then be encoded in base64.

The Payload

This is the content of the token. For example our Token ID previously described:

```
{
  "jti": "f232b54cb285452db02770c9d16f8f212151",
  "iss": "http://server.meritis.fr",
  "sub": "24400320",
  "aud": "s6BhdRkqt3",
  "nonce": "n-0S6_WzA2Mj",
  "exp": 1515604697,
  "iat": 1515593897,
  "name": "Matthieu Mabyre",
  "given_name": "Matthieu",
  "family_name": "Mabyre",
  "gender": "male",
  "email": "matthieu.mabyre@meritis.fr",
  "acr": ["role1", "role2", "role3"]
}
```

The field jti (JWT id) corresponds to the identifier of the token. For example, a UUID generated randomly. The payload must also be encoded in base64 thereafter.

Token signature

It is performed using the signature algorithm (defined in the header) from:

- the base64 encoded url header
- encoded base64 encoded payload
- the private key of the authorization server.

The signing of the tokens aims to ensure that the self-supporting tokens generated by the authorization server have been generated by this server, and that they have not been tampered with by a third party.

Example:

```
HMACSHA256 (
  Base64UrlEncode(header) + "." +
```

In the end, we get a JWS (S for Signed) token encoded in base64 (its size will depend on the size of the payload):

PASTE A TOKEN HERE

EDIT THE PAYLOAD AND SECRET (ONLY HS256 SUPPORTED)

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

```
{
  "jti": "f232b54cb285452db02770c9d16f8f212151",
  "iss": "http://server.meritis.fr",
  "sub": "24400320",
  "aud": "s6BhdRkqt3",
  "nonce": "n-0S6_WzA2Mj",
  "exp": 1515604697,
  "iat": 1515593897,
  "name": "Matthieu Mabyre",
  "given_name": "Matthieu",
  "family_name": "Mabyre",
  "gender": "male",
  "email": "matthieu.mabyre@meritis.fr",
  "acr": ["role1", "role2", "role3"]
}
```

```
HMACSHA256(
    base64UrlEncode(header) + "." +
    base64UrlEncode(payload),
    secret
)
```

Page: 13 / 61

2. UAS Options configuration

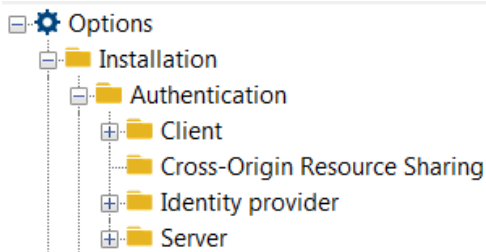
UAS is configurable by HOPEX Administration through option module.

2.1. Configuring authentication options

To configure authentication options, you must access to HOPEX Options in Extended view.

To configure authentication options:

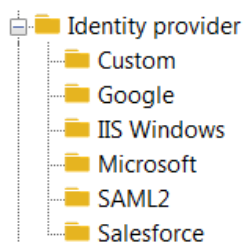
- 1) Go to HOPEX Administration.
- 2) Right-click **HOPEX** and select **Options > Modify**.
- 3) Right-click **Options** and select **Extended**.
- 4) Expand **Installation > Authentication** folders.



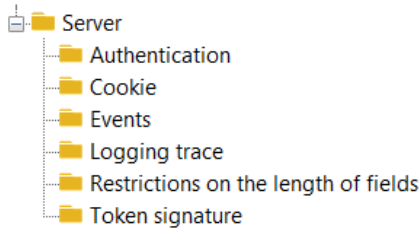
- 5) Click:
 - **Client**, and configure the client options.
 - See Client Option Description.



- **Cross-origin resource sharing**, and configure the cross origin resource sharing options.
 - See Cross-Origin Resource Sharing Option Description .
- **Identity provider**, and configure the identity provider options.
 - See Identity Provider Option Description



- **Server**, and configure the server global options and specific options.
 - See Server Option Description.



2.2. Server Option Description

2.2.1. Server global options

The server global options (**HOPEX > Options (Extended view) > Installation > Authentication > Server**) are the following:

- **Name of the authentication site** (available in Standard view)
Displays name of the site used in standard views. Default to HOPEX.
- **SSL Activation**
Indicates if SSL is required for UAS. You need to activate this option if you want to secure your authentication service and work with some identity providers. Defaults to false.
- **Public origin of the site**
By default, UAS uses the host, protocol, and port from the HTTP request when creating links. This might not be accurate in reverse proxy or load-balancing situations. You can override the origin used for link generation using this property.
- **Type of information storage**
UAS allows to use the following data component types:
 - **InMemory**
This storage type allows to store data in memory of your server. It's the simplest storage type. You cannot use it with a cluster deployment and you lose your token when you reboot your server.
 - **Redis (Cluster Compatible)**
This storage type allows to store data in Redis, a nosql database storage, used for caching. You can use it with a cluster deployment. No data loss.
 - **NCache (Cluster Compatible)**
This storage type allows to store data in NCache Enterprise 4.6.2 minimum, used for caching. You can use it with a cluster deployment. No data loss.

For legal reason (License redistribution), MEGA cannot provide the assemblies to work with. You need to put the following assemblies in the UAS binary folder (by default wwwroot/uas/bin):
 - Alachisoft.NCache.Web.dll
 - Alachisoft.NCache.Runtime.dll
 - **SqlServer (Cluster Compatible)**
This storage type allows to store data in SQL Server 2008 R2 minimum, used for caching. You can use it with a cluster deployment. No data loss.

You must launch SQL scripts which can be found in:

```
{HOPEX installation folder}\Utilities\Hopex Cluster Tools\UAS_Scripts.sql
```

- **Parameters of the UAS data component**

This parameters allows to complete the data component type. Several data components needs more informations like connectionstring or key.

- **InMemory**

No need to fill this parameter.

- **Redis (Cluster Compatible)**

Fill the Redis connection string into this parameter.

- **NCache (Cluster Compatible)**

Fill the NCache name into this parameter.

- **SqlServer (Cluster Compatible)**

Fill the SQLServer connection string into this parameter.

```
Data Source={ServerName}\{Instance};Initial Catalog={DataBaseName};Integrated
Security=False;User ID={UserId};Password={UserPassword}
```

2.2.2. Authentication options

The server authentication options (**HOPEX > Options (Extended view) > Installation > Authentication > Server > Authentication**) are the following:

- **Activation of user name backup**

Indicates whether UAS will remember the last username entered on the login page. Defaults to true

- **Activation of the local connection**

Indicates if UAS will allow users to authenticate with a local account. Disabling this setting will not display the username/password form on the login page. This also will disable the resource owner password flow. Defaults to true.

- **Activation of the login hint**

Indicates whether the *login_hint* parameter is used to prepopulate the username field. Defaults to true.

- **Limit to the number of consecutive messages**

Gets or sets the limit after which old sign in messages (cookies) are purged. Defaults to 3.

- **Activation of the capacity to redirect directly to an URL after disconnection**

Gets or sets a value indicating whether UAS automatically redirects back to a validated *post_logout_redirect_uri* passed to the sign out endpoint. Defaults to false.

- **Activation of confirmation during connection**

Indicates whether UAS will show a confirmation page for sign-in. When a client initiates a sign-in, by default UAS will not ask the user for confirmation but you can activate it. This is a mitigation technique against “logout spam”. Defaults to false.

- **The redirection timeout of the automatic post-disconnection**

Gets or sets the delay (in seconds) before redirecting to a *post_logout_redirect_uri*. Defaults to 0.

- **Activation of display of the disconnection confirmation page**

Indicates whether UAS will show a confirmation page for sign-out. When a client initiates a sign-out, by default UAS will not ask the user for confirmation.

- **The invalid sign in redirect URL**

Gets or sets the invalid sign in redirect URL. If the user arrives at the login page without a valid sign-in request, then they will be redirected to this URL. The URL can be absolute or relative (starting with “~/”).

2.2.3. Cookie options

The server cookie options (**HOPEX > Options > Installation > Authentication > Server > Cookie**) are the following:

- **Display the “Remember me” checkbox in the login page or not**

Indicates whether the “remember me” option is presented to users on the login page. If selected this option will issue a persistent authentication cookie. Defaults to false.

- **Sliding cookie Expiration**

Indicates if the authentication cookie is sliding, which means it auto renews as the user is active. Defaults to false.

2.2.4. Events options

UAS raises many events at runtime.

The server event options (**HOPEX > Options (Extended view) > Installation > Authentication > Server > Events**) are the following:

- Successful/failed authentication (resource owner flow, pre, partial, local and external)
- Token issued (identity, access, refresh tokens)
- Token handle related events (authorization code, refresh token issued/redeemed/refreshed)
- Permission revoked
- Endpoint success/failures
- Expired/invalid/no signing certificate
- Unhandled exceptions and internal errors
- CSP errors reported by the browser

By default, these events are forwarded to the configured log provider - a custom event service can process or forward them in any way suitable for the environment.

The Events options are the following (all default to false):

- **Generation of all events**

Activates all the events below

- **Generation of success events**

Activates refresh token refreshed or authentication success.

- **Generation of failure events**

Activates authentication failure, authorization code redeems failure events.

- **Generation of error events**

Activates unhandled exceptions events.

- **Generation of information events**

Activates token issued or certificate valid events.

2.2.5. Logging trace options

The server logging trace options (**HOPEX > Options (Extended view) > Installation > Authentication > Server > Logging trace**) are the following:

- **Activation of all logging**

Activates all the logging traces (http flow, katana Flow, programming web interface, programming web interface in Verbose mode).

- **Activation of Http flow logging**

When enabled, HTTP requests and responses are logged.

- **Activation of Katana Flow Logging**

When enabled, the Katana log output is logged (this is often useful to troubleshoot problems with external identity providers).

- **Activation of logging of the programming web interface**

When enabled, Web API internal diagnostic logging is forwarded to the log provider.

- **Activation of logging of the programming web interface in Verbose mode**

When enabled, the Web API diagnostics logging is set to verbose.

2.2.6. Token Signature options

The server token signature options (**HOPEX > Options (Extended view) > Installation > Authentication > Server > Token Signature**) are the following:

- **Name of the certificate to be used by the authentication server**

X.509 certificate (and corresponding private key) name for signing security tokens.

- **Password for the signature certificate of the authentication server**

X.509 certificate (and corresponding private key) password for signing security tokens.

2.3. Identity Provider Option Description

The identity provider options (**HOPEX > Options > Installation > Authentication > Identity provider**) are the following:

- HOPEX, see HOPEX provider.
- IIS windows, see IIS Windows provider
- SAML2, see SAML2 provider.

2.3.1. Google, see Open ID Connect (OIDC) provider

- Google provider.
- Microsoft, see Microsoft provider.
- Salesforce, see Salesforce provider.
- Custom, see Custom provider.

2.3.2. HOPEX provider

The HOPEX provider is the HOPEX default provider, which displays a login page with username and password.

To authenticate HOPEX users, use either:

- HOPEX User Native Authentication
See HOPEX Administration documentation: Authentication in HOPEX section.
- LDAP Authentication

See HOPEX Administration documentation: Authentication in HOPEX section.

2.3.3. IIS Windows provider

With the IIS Windows provider HOPEX users are authenticated by Windows Authentication.

To use IIS Windows provider, you must configure the following mandatory options:

- Enabled (Default to false)
- Name displayed for the Windows IIS access provider (Windows by Default)
- URL of the Windows IIS authentication server.

2.3.4. SAML2 provider

SAML 2.0 is an XML based framework, used to describe and exchange security information. It can be used for Single Sign On (SSO), Identity Management and Federation.

To use SAML2 provider, you must set UAS in SSL Mode.

UAS manages only Service Provider (SP) initiated SSO and not Identity Provider(IDP) initiated SSO.

We have implemented two SAML2 Identity Provider and you can find the server configuration example:

- SAML2 ADFS Server Configuration section
- OKTA Configuration section

2.3.5. Open ID Connect (OIDC) provider

2.3.6. Google provider

Use the Google provider to authenticate HOPEX users with a google account by OAUTH2.

To use Google provider, you must:

- set UAS in SSL Mode (Options > Installation > Authentication > server)
- Have your website accessible by Google server
- enter the following fields:
 - Enabled (Activated or not)
 - Display name (Text the user can display in the user interface, by Default Google)
 - Google Client ID (Identifier of the Google access provider client)
 - Google Secret code (Password of the Google access provider client)
 - Google Scope (Scope of the authenticated client)

If you do not know this information, go to the Google console (<https://console.developers.google.com>) to create your API keys.

Do not forget to activate Google + API at least, otherwise you will not be able to authenticate HOPEX users.

2.3.7. Microsoft provider

Use the Microsoft provider to authenticate HOPEX users with a Microsoft account by OAUTH2.

To use Microsoft provider, you must set UAS in SSL Mode and ensure the address is accessible by Microsoft server and fill the following fields:

- Enabled (Activated or not)
- Display name (Text the user can display in the user interface)
- Microsoft Client ID (Identifier of the Microsoft access provider client)
- Microsoft Password (Password of the Microsoft access provider client)
- Microsoft Scope (Scope of the authenticated client)

If you do not know this information, go to the Microsoft website (<https://apps.dev.microsoft.com>) to create your API keys.

2.3.8. Salesforce provider

Use the Salesforce provider to authenticate HOPEX users with a Salesforce account by OAUTH2.

To use Salesforce provider, you must set UAS in SSL Mode and ensure the address is accessible by Salesforce server and fill the following fields:

- Enabled (Activated or not)
- Display name (Text the user can display in the user interface)
- Salesforce Client ID (Identifier of the Salesforce access provider client)
- Salesforce Password (Password of the Salesforce access provider client)
- Salesforce Scope (Scope of the authenticated client)

If you do not know this information, go to the website:

https://help.salesforce.com/articleView?id=connected_app_create.htm for information on how to create a connected app.

To connect Salesforce with UAS through Salesforce Admin, fill:

- **Application Url** field: `http(s)://<servername>/HOPEX/DEFAULT.ASPX`
- **CallBack Url** field: `http(s)://<servername>/UAS/signin-salesforce`

2.3.9. Custom provider

Use the Custom provider to authenticate HOPEX users with your own implementation. You must fill the following fields:

- **Enabled** (Boolean)
Activate or not your Custom Provider
- **Parameters** (string)
Parameters used by your custom provider.

To create your custom provider, see **Error! Reference source not found.** section.

Do not forget to:

- set **Namespace** properties to "Custom"
- call your C# project: **MEGA.UAS.IdentityProvider.Custom**

2.4. Cross-Origin Resource Sharing Option Description

(available with options in Extended view only)

Cross Origin Resource Sharing (CORS) allows us to use Web applications within browsers when domains aren't the same. For example, a site with domain `test.org` wants to execute AJAX requests to a Web application with domain `mydomain.org` using HTTP.

Using CORS isn't so simple especially when you face debugging difficulties. As a matter of fact, CORS can imply an additional OPTIONS request and error messages aren't so explicit. Most of the time, errors correspond to a lack of required headers from the server. For such reasons, a good understanding of how this feature works is essential.

CORS is used in a lot of places and use cases. In Web development, it's often necessary to split the front application from the server application for development reasons or to interact with a remote service.

The CORS mechanism is mainly implemented with the Web server but this has an impact on the client side if some headers are missing in responses.

With CORS, the remote Web application (here the one with domain `mydomain.org`) chooses if the request can be served. The CORS specification distinguishes two distinct use cases:

Simple requests. This use case applies if we use HTTP `GET`, `HEAD` and `POST` methods. In the case of POST methods, only content types with the following values are supported: `text/plain`, `application/x-www-form-urlencoded` and `multipart/form-data`.

Preflighted requests. When the ‘simple requests’ use case doesn’t apply, a first request (with the HTTP `OPTIONS` method) is made to check what can be done in the context of cross-domain requests.

Notice that if you add authentication to the request using the `Authentication` header, simple requests automatically become preflighted ones.

Client and server exchange a set of headers to specify behaviors regarding cross-domain requests. Let’s have a look at them now. We will then describe how they are used in both use cases.

`Origin`: this header is used by the client to specify which domain the request is executed from. The server uses this hint to authorize, or not, the cross-domain request.

`Access-Control-Request-Method`: with in the context of preflighted requests, the `OPTIONS` request sends this header to check if the target method is allowed in the context of cross-domain requests.

`Access-Control-Request-Headers`: with in the context of preflighted requests, the `OPTIONS` request sends this header to check if headers are allowed for the target method in the context of cross-domain requests.

`Access-Control-Allow-Credentials`: this specifies if credentials are supported for cross-domain requests.

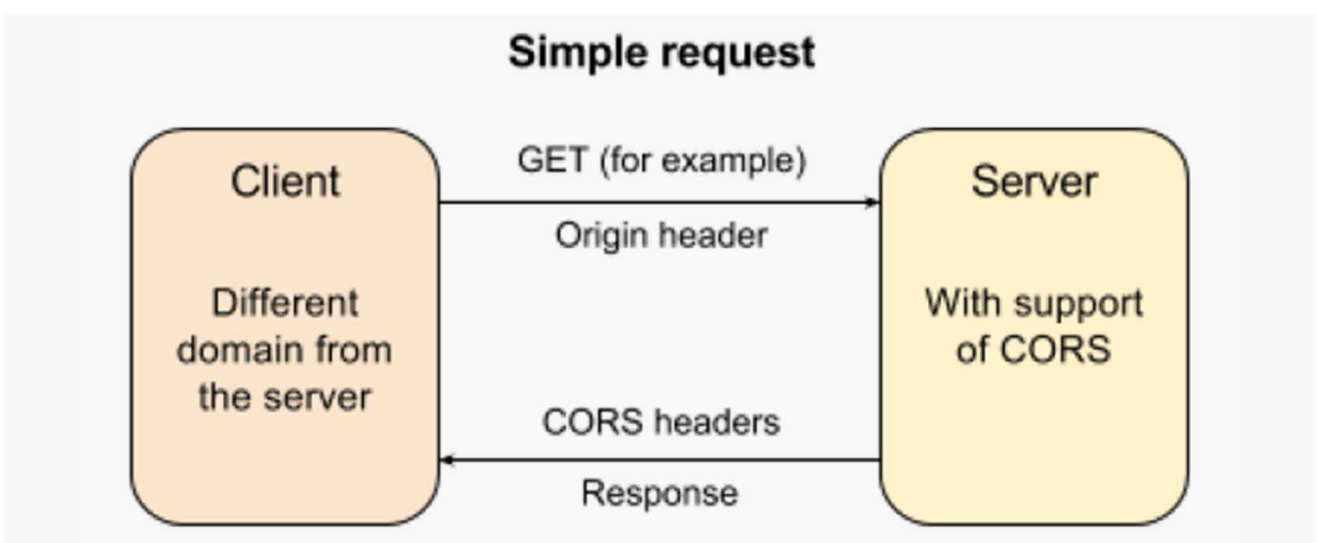
`Access-Control-Allow-Methods`: the server uses this header to tell which headers are authorized in the context of the request. This is typically used in the context of preflighted requests.

`Access-Control-Allow-Origin`: the server uses this header to tell which domains are authorized for the request.

`Access-Control-Allow-Headers`: the server uses this header to tell which headers are authorized in the context of the request. This is typically used in the context of preflighted requests.

2.4.1. Simple requests

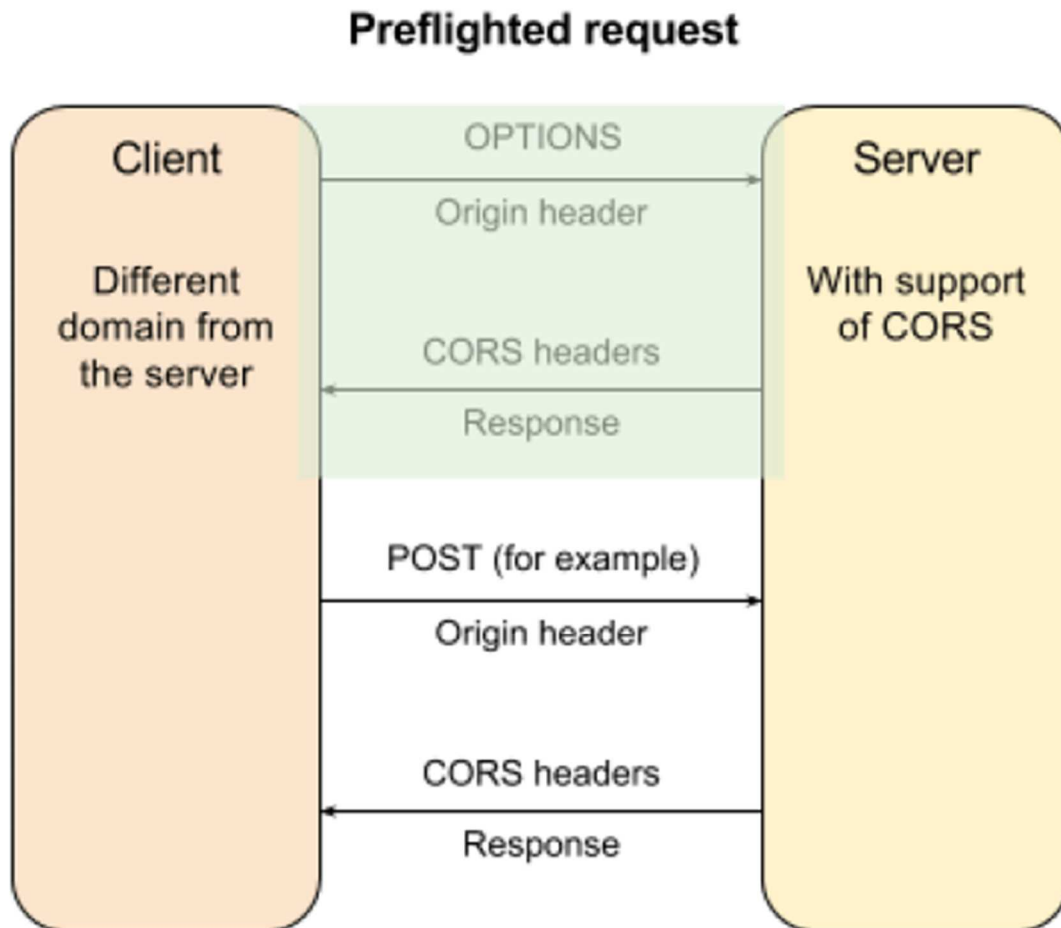
Where simple requests are concerned, the request is executed against the other domain. If the remote resource supports cross domains, the response is directly sent back. Otherwise an error occurs.



2.4.2. Preflighted requests

In the case of preflighted requests, this corresponds to a negotiation between the caller and the Web application based on HTTP headers. It consists of two phases:

The browser first executes an `OPTIONS` request with the same URL as the target request to check that it has the rights to execute the request. This `OPTIONS` request returns headers that identify what is possible to do for the URL. If rights match, the browser executes the request.



This processing is completely transparent for the caller but we can have hints of what actually happens using the development tools of the browser, for example Firebug within Firefox. With the latter, we can use the Network tab to check which calls are executed and which CORS headers are exchanged.

An important comment. You must take into account that, when executing CORS request containing security, i.e. an `Authorization` header, the `OPTIONS` request won't contain it. So you need to be careful regarding security when handling the first `OPTIONS` requests of preflighted ones. As a matter of fact, no authentication check can be done at this level.

2.4.3. Use within browsers

When implementing applications within browsers using JavaScript, it's common to interact with services that aren't hosted on the same domain. Several solutions are usable like JSONP, but also CORS. In this section, we will describe how to use this feature within commonly used tools.

Low-level API

XHR corresponds to the low-level API to implement AJAX calls within JavaScript applications in browsers. Regarding CORS, there is nothing to do unless you want to use credentials with the `withCredentials` property. It's mainly if you deal with cookies.

In fact, if the browser detects that the domain of the page is different from the AJAX request that is about to be sent, it automatically adds the Origin header. The consequence is to enable CORS support within the response if the Web service supports this technology.

Here is a sample of a simple CORS request:

```
var req = new XMLHttpRequest();

req.open('GET', 'https://maptestapi.apispark.net/v1/maps/', true);

req.onreadystatechange = function() {
    if (req.readyState === 4) {
        console.log(req.responseText);
    }
};

req.setRequestHeader('Accept', 'application/json');
req.send();
```

This will result in the following request in the browser:

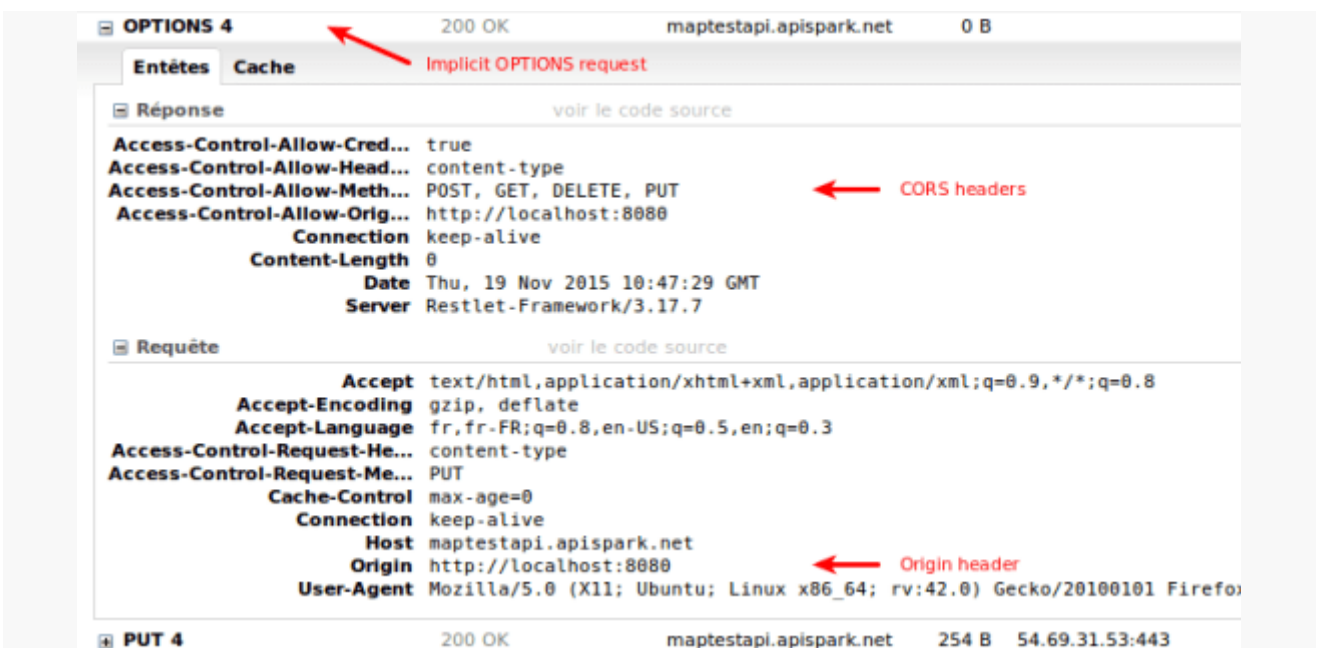
The screenshot shows the browser's developer tools with the network tab selected. The request is a GET to `/v1/maps/` with a status of 200 OK. The response headers are expanded, showing CORS headers: `Access-Control-Allow-Credentials: true` and `Access-Control-Allow-Origin: http://localhost:8080`. A red arrow points to these headers with the text "CORS headers of the response". The request headers are also expanded, showing the `Origin: http://localhost:8080` header, with a red arrow pointing to it and the text "Origin header automatically added".

Here is a sample of a preflighted CORS request:

```
var req = new XMLHttpRequest();
```

```
req.open('PUT', 'https://maptestapi.apispark.net/v1/maps/4', true);
req.onreadystatechange = function() {
    if (req.readyState === 4) {
        console.log(req.responseText);
    }
};
req.setRequestHeader('Content-type', 'application/json');
req.send('{ "id": "4", "name": "Meteorites", "type": "d3js", (...) }');
```

This will result in the following request in the browser:



As described above, CORS is natively supported by the JavaScript XMLHttpRequest object. We can notice that old Internet Explorer versions (7 and 8) use a dedicated object for XDomainRequest requests.

That being said, if you want to send credentials managed by the browser (Access Control) within CORS requests, you need to enable the credentials support of XHR using the `withCredentials` attribute. By default, the feature is disabled.

```
var req = new XMLHttpRequest();
(...)
```

```
req.withCredentials = true;
```

Notice that you don't need to use this attribute if you build the `Authentication` header by yourself.

Libraries and frameworks

The aim of JavaScript libraries and frameworks regarding AJAX is to provide an abstraction level upon browser specificities, an API that is common and works in every browser. That means that using CORS is transparent.

JQuery provides a way to configure the `withCredentials` attribute previously discussed, as described below:

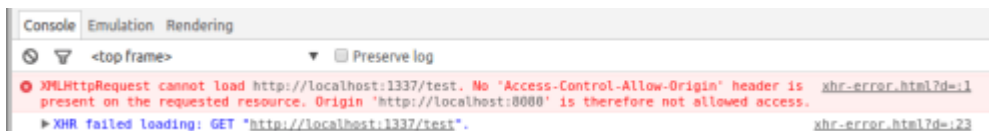
```
$.ajax({  
  type: "POST",  
  data: {},  
  dataType: 'json',  
  xhrFields: {  
    withCredentials: true  
  }  
});
```

With AngularJS, there is nothing to do.

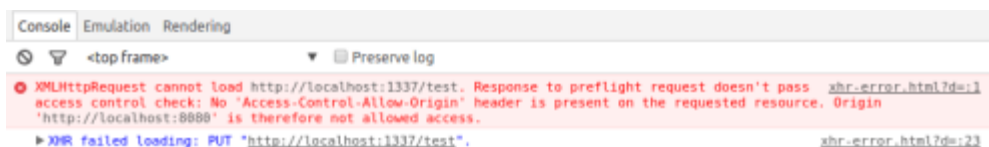
Note that on the web you can find content telling you that you need to set the `defaults.useXDomain` property of the `$httpProvider` object to true or to delete the `X-Requested-With` header. This may have been true in the past but it's not the case anymore.

Error detection

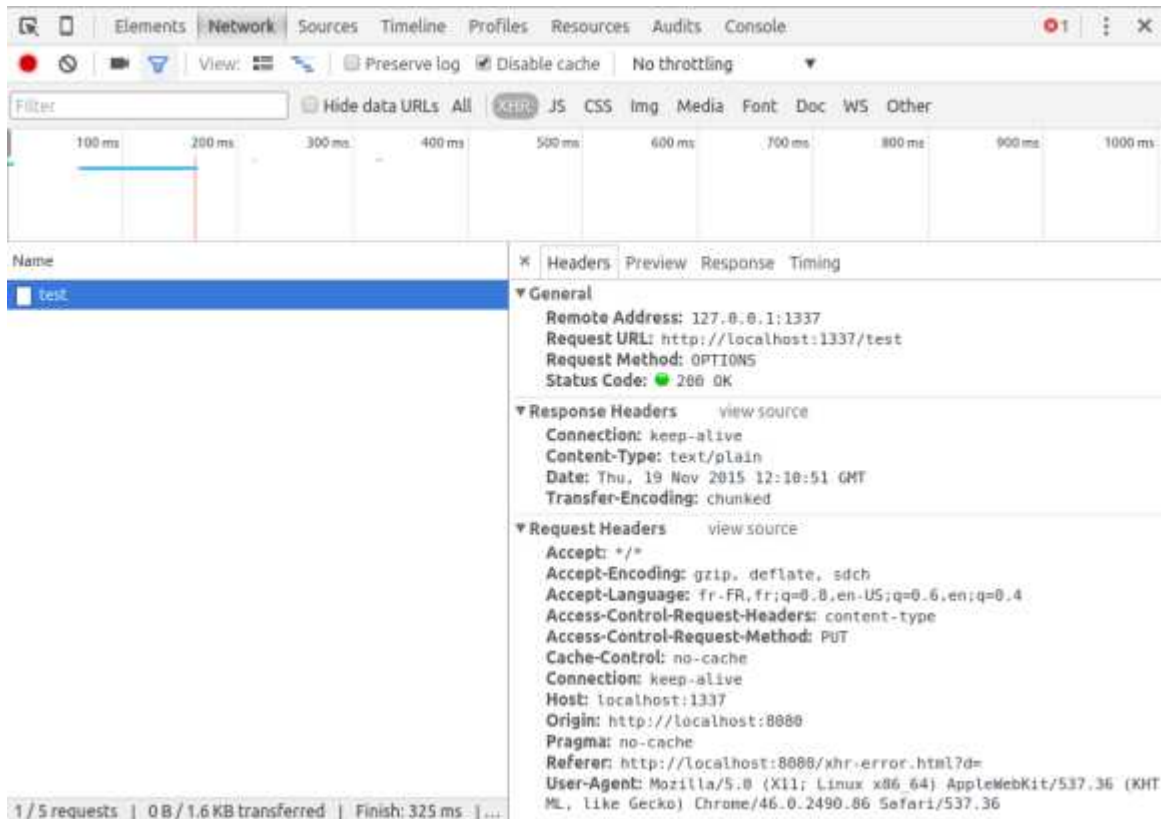
To detect CORS problems, you obviously need to enable debugging tools in your browser and especially the JavaScript console. For example, in Chrome, if the server doesn't include CORS headers in the response, you will see this error for a simple request:



And this one for a preflighted one:



Using the Network tab, you will see that CORS headers aren't present in the response of the OPTIONS request:



Errors can be more subtle since the content of CORS headers could possibly be incorrect. For example, if you try to call a HTTP method that didn't return `Access-Control-Allow-Methods`.

Another source of errors is that credentials aren't sent within the `OPTIONS` request (`Authorization` header). When implementing CORS on the server side, checking credentials musn't apply for this request but only for target ones.

OPTIONS 4

200 OK

maptestapi.apispark.net

0 B

Entêtes

Cache

Réponse

voir le code source

Access-Control-Allow-Cred...

true

Access-Control-Allow-Head...

authorization, content-type

Access-Control-Allow-Meth...

POST, GET, DELETE, PUT

Access-Control-Allow-Orig...

http://localhost:8080

Connection

keep-alive

Content-Length

0

Date

Thu, 19 Nov 2015 12:53:24 GMT

Server

Restlet-Framework/3.17.7

Requête

voir le code source

Accept

text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Encoding

gzip, deflate

Accept-Language

fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3

Access-Control-Request-He...

authorization, content-type

Access-Control-Request-Me...

PUT

Cache-Control

max-age=0

Connection

keep-alive

Host

maptestapi.apispark.net

Origin

http://localhost:8080

User-Agent

Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:42.0) Gecko/20100101 Firefox/42.0

PUT 4

200 OK

maptestapi.apispark.net

254 B

54.68.76.111:443

Entêtes

Put

Réponse

XML

Cache

Réponse

voir le code source

Accept-Ranges

bytes

Access-Control-Allow-Cred...

true

Access-Control-Allow-Orig...

http://localhost:8080

Connection

keep-alive

Content-Type

application/xml; charset=UTF-8

Date

Thu, 19 Nov 2015 12:53:25 GMT

Server

Restlet-Framework/3.17.7

Transfer-Encoding

chunked

Vary

Accept-Charset, Accept-Encoding, Accept-Language, Accept

Requête

voir le code source

Accept

text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Encoding

gzip, deflate

Accept-Language

fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3

Authorization

Basic MTY3ZWJmZjYtN2VnYy00MjYtNDItZWZmODU0ZDg2ZWZk0jlkODRmM2U4LTc5MGQ0tNGMSMS05Y2UyLTkwOGI

Connection

keep-alive



With UAS you can add resources from another domain. It can be used if you use several domains in your HOPEX installation. You can have up to 5 different CORS.

2.5. Client Option Description

2.5.1. HOPEX Custom (options in Extended view only)

HOPEX allows you to authenticate an external client with UAS and enable UAS SSO Features.

You must fill the following fields:

- Enabled (Activated or not)
- Client ID: HopexCustom (default)
- Client Name: Hopex Custom Client (default)
- Client secret: secret (default)
- Client Scopes
- Client Flow

- Client Redirect uri
- Client Allowed Redirect uri
- Client Post Logout Redirect uri

3. UAS API ENDPOINTS

3.1. Authorization/Authentication

The authorization endpoint can be used to request either access tokens or authorization codes (implicit and authorization code flow). You either use a web browser or a web view to start the process.

- **client_id** (required)
Identifier of the client
- **scope** (required)
One or more registered scopes
- **redirect_uri** (required)
must match exactly one of the allowed redirect URIs for that client
- **response_type** (required)
 - **code** requests an authorization code
 - **token** requests an access token (only resource scopes are allowed)
 - **id_token** token requests an identity token and an access token (both resource and identity scopes are allowed)
- **response_mode** (optional)
form_post sends the token response as a form post instead of a fragment encoded redirect
- **state** (recommended)
Unified Authentication Service will echo back the state value on the token response, this is for correlating request and response.
- **nonce** (required for identity tokens using implicit flow)
Unified Authentication Service will echo back the nonce value in the identity token, this is for correlating the token to the request).
- **prompt** (optional)
 - **none** no UI will be shown during the request. If this is not possible (e.g. because the user has to sign in or consent) an error is returned
 - **login** the login UI will be shown, even if the user is already signed-in and has a valid session
- **code_challenge** (required when using proof keys)
Sends the code challenge for proof key flows.
- **code_challenge_method** (optional - default to plain when using proof key)
 - **plain** indicates that the challenge is using plain text (not recommended)
 - **S256** indicates the challenge is hashed with SHA256
- **login_hint** (optional)
Can be used to pre-fill the username field on the login page.
- **ui_locales** (optional)
Gives a hint about the desired display language of the login UI

- **max_age** (optional)
If the user's logon session exceeds the max age (in seconds), the login UI will be shown
- **acr_values** (optional)
Allows to pass additional authentication related information to the user service - there are also values with special meaning:
 - **idp:name_of_idp** bypasses the login/home realm screen and forwards the user directly to the selected identity provider (if allowed per client configuration)
 - **tenant:name_of_tenant** can be used to pass a tenant name to the user service

Example (URL encoding removed for readability)

```
GET /connect/authorize?client_id=client1&scope=openid email
api1&response_type=id_token token
```

3.2. Token

The token endpoint can be used to programmatically request or refresh tokens (resource owner password credential flow, authorization code flow, client credentials flow and custom grant types).

- **grant_type** (required)
 - authorization_code
 - client_credentials
 - Password
 - refresh_token
 - custom
- **scope** (required for all grant types besides refresh_token and code)
- **redirect_uri** (required for code grant type)
- **code** (required for code grant)
- **code_verifier** (required when using proof keys - added in v2.5)
- **username** (required for password grant type)
- **password** (required for password grant_type)
- **acr_values** (allowed for password grant type to pass additional information to user service)
Values with special meaning:
 - **idp:name_of_idp** bypasses the login/home realm screen and forwards the user directly to the selected identity provider (if allowed per client configuration)
 - **tenant:name_of_tenant** can be used to pass extra information to the user service
- **refresh_token** (required for refresh token grant)
- **client_id** (either in the post body, or as a basic authentication header)
- **client_secret** (either in the post body, or as a basic authentication header)

Authentication

All requests to the token endpoint must be authenticated - either pass client id and secret via Basic Authentication or add client_id and client_secret fields to the POST body.

Example: (Form-encoding removed and line breaks added for readability)

```
POST /connect/token
Authorization: Basic abcxyz
grant_type=authorization_code&code=hdh922&redirect_uri=https://myapp.com/callback
```

3.3. UserInfo

The UserInfo endpoint can be used to retrieve identity information about a subject. It requires a valid access token with at least the “openid” scope.

Example:

```
GET /connect/userinfo
Authorization: Bearer <access_token>
HTTP/1.1 200 OK
Content-Type: application/json
{
  "sub": "248289761001",
  "name": "Bob Smith",
  "given_name": "Bob",
  "family_name": "Smith",
  "role": [
    "user",
    "admin"
  ]
}
```

3.4. Discovery Endpoint

The discovery endpoint can be used to retrieve metadata about **Unified Authentication Service** - it returns information like the issuer name, key material, supported scopes etc.

Example:

```
GET /.well-known/openid-configuration
```

3.5. Logout Endpoint

Redirecting to the logout endpoint clears the authentication session and cookie.

You can pass the following optional parameters to the endpoint:

- id_token_hint
The id_token that the client retrieved during authentication. This allows bypassing the logout confirmation screen as well as providing a post logout redirect URL
- post_logout_redirect_uri

A URI that **Unified Authentication Service** can redirect to after logout (by default a link is displayed). The URI must be in the list of allowed post logout URIs for the client.

```
/connect/endsession?id_token_hint=...&post_logout_redirect_uri=https://myapp.com
```

See the Authentication options section to configure the behavior of logout endpoint and logout page.

3.6. Token Revocation

This endpoint allows revoking access tokens (reference tokens only) and refresh token. It implements the token revocation specification.

Supported parameters:

- **token** (required)
The token to revoke
client_id (required)
client_secret (required)
- **token_type_hint**
Either access_token or refresh_token

3.7. Introspection Endpoint

The introspection endpoint is an implementation of RFC 7662.

It can be used to validate reference tokens (or JWTs if the consumer does not have support for appropriate JWT or cryptographic libraries).

The introspection endpoint requires authentication using a scope credential (only scopes that are contained in the access token are allowed to introspect the token).

Example:

```
POST /connect/introspect
Authorization: Basic xxxyyy
token=<token>
```

A successful response returns a status code of 200 and either an active or inactive token:

```
{
  "active": true,
  "sub": "123"
}
```

Unknown or expired tokens are marked as inactive:

```
{
  "active": false,
}
```

An invalid request returns a 400 or a 401 if the scope is not authorized.

Note:

The introspection endpoint replaces the old access token validation endpoint. Since the introspection endpoint requires authentication, it adds privacy features to reference tokens, which were not available previously. The access token validation endpoint still exists, but it is recommended to disable it and use the introspection endpoint instead.

3.8. Access token validation endpoint

The access token validation endpoint can be used to validate reference tokens. It can be also used to validate self-contained JWTs if the consumer does not have support for appropriate JWT or cryptographic libraries.

You can either GET or POST to the validation endpoint. Due to query string size restrictions, POST is recommended.

Example:

```
POST /connect/accesstokenvalidation
token=<token>
```

or

```
GET /connect/accesstokenvalidation?token=<token>
```

A successful response returns a status code of 200 and the associated claims for the token.

An unsuccessful response returns a 400 with an error message.

It is also possible to pass a scope that is expected to be inside the token:

```
POST /connect/accesstokenvalidation
token=<token>&expectedScope=calendar
```

Note:

The access token validation endpoint does not enforce client authentication.

Do not use reference tokens for confidentiality purposes.

3.9. Identity Token Validation Endpoint

The identity token validation endpoint can be used to validate identity tokens. This is useful for clients that do not have access to the appropriate JWT or crypto libraries (e.g. JavaScript).

You can either GET or POST to the validation endpoint. Due to query string size restrictions, POST is recommended.

Example:

```
POST /connect/identitytokenvalidation
token=<token>&client_id=<expected_client_id>
GET /connect/identitytokenvalidation?token=<token>&client_id=<expected_client_id>
```

A successful response returns a status code of 200 and the associated claims for the token.

```
{
  "nonce": "nonce",
  "iat": "1413203421",
```



```
"sub": "88421113",  
"amr": "password",  
"auth_time": "1413203419",  
"idp": "idsrv",  
"iss": "https://idsrv3.com",  
"aud": "implicitclient",  
"exp": "1413203781",  
"nbf": "1413203421"  
}
```

An unsuccessful response will return a 400 with an error message.

3.10. CSP Endpoint

Unified Authentication Service provides an endpoint to record CSP errors that the browser reports. These CSP errors are raised as events in the system event

4. ESTABLISH AN SSL CONNECTION

UAS must establish an SSL connection to communicate with SAML2 or OPENID Provider .

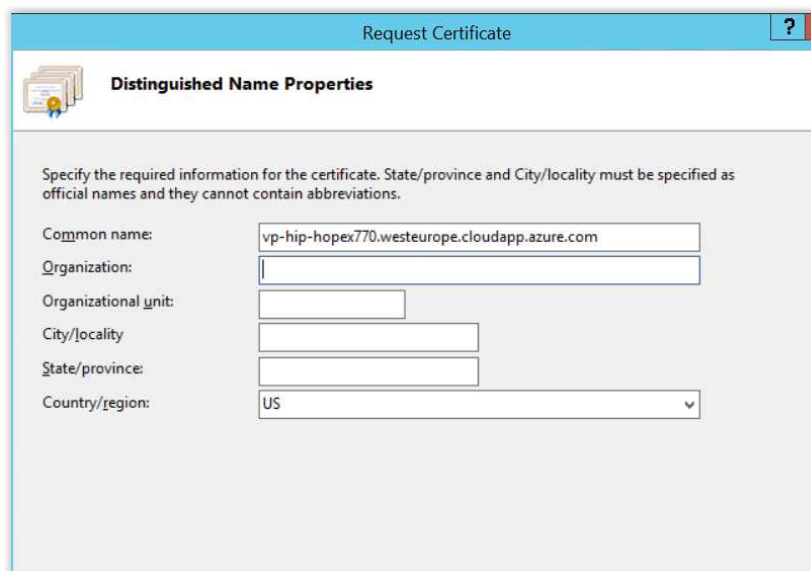
To establish an SSL connection:

Action	See
1 Create a certificate request	Creating a certificate request from IIS
2 Complete the certificate request	Completing the certificate request
3 Bind IIS with SSL certificate	Binding IIS with SSL certificate
4 Export the certificate to the local disk	Exporting certificate to the local disk

4.1. Creating a certificate request from IIS

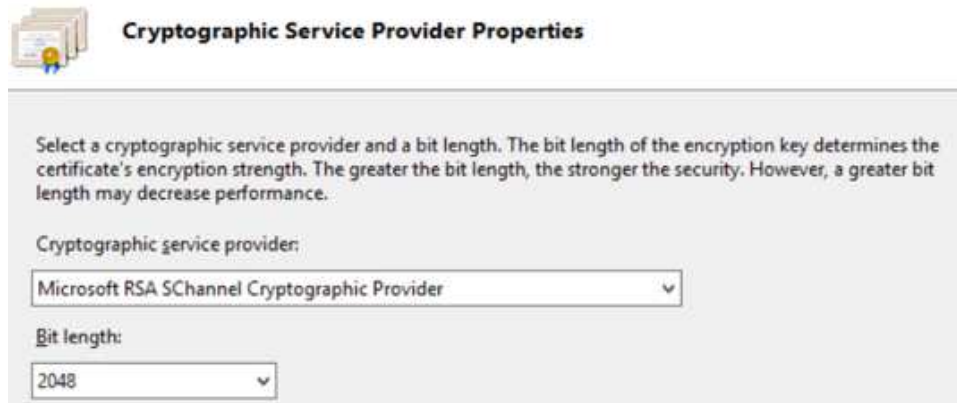
To create a certificate request from IIS:

1. Open **IIS**.
2. In **IIS** section, double-click **Server certificates**.
3. In the **Actions** pane, click **Create certificate request**.
4. In the **Common name** field, enter the server name (Fully Qualified domain name).



5. Enter all the requested fields.
6. Click **Next**.
The **Cryptographic Service Provider Properties** window appears.
7. In the **Cryptographic Service Provider** field, select "Microsoft RSA Channel Cryptographic Provider".

8. In the **Bit length** field, select “2048”.



9. Click **Next**.
10. Enter the name and saving location for the output request file
11. Click **Finish**.
12. Send this request to your CA Authority.
When you receive the certificate, complete the certificate request, see Completing the certificate request.

4.2. Completing the certificate request

To complete the certificate request:

1. Access **IIS > Server Certificates**.
2. In the **Actions** pane, click **Complete certificate request**.
3. In the **Friendly name** field, enter any name.
4. In the **Select a certificate store for the new certificate**, select “Personal”.
5. Click **OK**.

4.3. Binding IIS with SSL certificate

To bind IIS with SSL certificate:

1. In **IIS, Connections** pane, expand **Sites** folder and select **Default Web Site**.
2. In the **Actions** pane, click **Binding**.
3. Click **Add**.
4. In the **Type** field, select “https”.
5. In the **SSL certificate** field, select the certificate previously imported.
6. Click **OK**.

4.4. Exporting certificate to the local disk

To export the certificate to the local disk:

1. Open Manage certificate computer.
2. Expand **Personal > Certificates** folders.
3. Right-click your previously imported certificate and click **All Task > Export**.
4. Select “Yes, export the private key”.
5. Check **Export all extended properties**.
6. Select **Password**.
7. Enter and confirm your password.
8. Select your file location to export the certificate.
9. Copy this exported certificate file to the SAML2/ADFS server.

5. INSTALL HOPEX SIGNING CERTIFICATE (MANUALLY)

By default, the installer installs the UAS signing certificate, but the certificate could also be updated with the following procedure;

To install HOPEX signing certificate (manually):

1. Double-click your certificate.
2. Select **Local Machine** and click **Next** twice.
3. Fill the password field.
4. Click **Next** twice and click **Finish**.
5. Go to your certificate through "Computer Certificate Manager".
6. Go to **Personal Folder > Certificates**.
7. Right-click your certificate and select **All Tasks > Manage private keys**.
8. Add IIS_USRS as user and click **OK**.
9. Right-click your certificate and select **Properties**.
10. Enter a **Friendly Names**

The Friendly name and the password are important for UAS configuration: do not forget to change it if it changes in the HOPEX options.

6. CONFIGURE UAS HOPEX BY OPTIONS

6.1. Local Configuration

6.1.1. Defining authentication options

To define authentication options:

- 1) Go to HOPEX Administration.
- 2) Right-click **HOPEX** and select **Options > Modify**.
- 3) Check that you are in “Extended” mode (right-click **Options** and select **Extended**).
- 4) Expand **Installation**, **Authentication** and **Client** folder.
Note: **Client** folder is only available in “Extended” mode.
- 5) Select **Hopex Web**, and define:
 - Redirection URL when the user is authenticated to `http://<servername>/hopex/uaslogin.ashx`
 - List of authorized redirection URLs to `http://<servername>/hopex/uaslogin.ashx`
 - Redirection URL after disconnection to `http://<servername>/hopex/Default.aspx`

6.2. Cluster Configuration

In a cluster configuration, do not forget to specify the same MachineKey in the web.config of each UAS node. You also need to fix the urls on the loadbalancer urls, and not on several nodes.

6.2.1. Configuring your data component type

Hopex manages several data component for Cluster configuration.

To configure your data component type:

- 1) Go to HOPEX Administration.
- 2) Right-click **HOPEX** and select **Options > Modify**.
- 3) Check that you are in “Extended” mode (right-click **Options** and select **Extended**).
- 4) Expand **Installation > Authentication** folders.
- 5) Select **Server** and in the right pane define (see Authentication options section):
 - **Type of information storage**
 - **Parameters of the UAS data component**

6.2.2. Configuring your SQL Server Data component type

To configure your SQL Server Data component type:

- 1) Create an SQL Database named “UAS”.

- 2) Create the table with the script "UAS_Scripts.sql" located in {Mega installation folder}\Utilities\HOPEX Cluster Tools.
- 3) Go to HOPEX Administration.
- 4) Right-click **HOPEX** and select **Options > Modify**.
- 5) Check that you are in "Extended" mode (right-click **Options** and select **Extended**).
- 6) Expand **Installation > Authentication** folders.
- 7) Select **Server** and in the right pane define (see Authentication options section):
 - **Type of information storage : SQL Server**
 - **Parameters of the UAS data component : Your connection string**

6.2.3. Configuring Hopex Web options

To configure Hopex Web options:

- 1) Go to HOPEX Administration.
- 2) Right-click **HOPEX** and select **Options > Modify**.
- 3) Check that you are in "Extended" mode (right-click **Options** and select **Extended**).
- 4) Expand **Installation, Authentication** and **Client** folder.
Note: **Client** folder is only available in "Extended" mode.
- 5) Select **Hopex Web** and define:
 - **Redirection URL when the user is authenticated** to front load balancer server name:
`http(s)://<servername>/hopex/uaslogin.ashx`
 - **List of authorized redirection URLs** to front load balancer server name:
`http(s)://<servername>/hopex/uaslogin.ashx`
 - **Redirection URL after disconnection** to front load balancer server name:
`http(s)://<servername>/hopex/Default.aspx`

6.2.4. Generating Machine Key

To generate the machine key:

- 1) Go to your IIS Manager
- 2) Expand
- 3) Click **UAS** application.
- 4) Double-click **Machine Key**.
- 5) Select your validation method (SHA1 by default).
- 6) Select your Encryption method (AES by default).
- 7) Clear "Automatically generate at runtime" validation key.
- 8) Clear "Automatically generate at runtime" decryption key.
- 9) In **Actions** pane, click **Generate Keys**.
- 10) Do the same with **Windows Authentication Service** if you use UAS Windows Authentication mode.

7. CONFIGURE CLIENT USING UAS

All the clients that want to communicate with UAS can use UAS SDK.

To configure the client with UAS SDK:

1. Go to <HOPEX installation folder>\DotNet\packages.
2. Download **MEGA.UAS.Client.Owin** package to retrieve UAS SDK.
MEGA.UAS.Client.Owin is an OWIN middleware.
3. Use the **ConfigureClient** method to configure the client.

The following options must be the same (case sensitive) in client configuration as in HOPEX Options configuration:

- **AuthenticationUrl** (mandatory, only in Client configuration)
Unified Authentication Service URL
- **ClientId** (mandatory)
Identifier of your client
- **ClientSecret** (mandatory)
Secret of your client
- **ClientRedirectUri** (mandatory)
Redirect Url of your client
- **ClientPostLogoutRedirectUri** (not mandatory)
Post logout redirect url of your client if you want to manage it.
- **ClientScope** (mandatory)
Scope of your client
Usually, this information is stored in web.config website.

8. STANDALONE MODE

In its nominal version, UAS is designed to work with SSP. But it is possible to operate it in an environment without SSP: it is the standalone mode. This allows it to be used by earlier version of Hopex, or even independently of Hopex.

To do so, you must declare in the appsettings section of UAS web.config file :

```
<add key="ConfigurationMode" value="Standalone" />
```

In this mode, the configuration necessary for the proper operation of UAS is retrieved from a config.json file which must be located in the UAS bin folder.

In case the Hopex provider is activated in the standalone mode, you must also add two additional keys:

- the URL to access the API for recovering old version environments:

```
<add key="HopexApiUrl" value="http://my-server/hopexapi/restapi/v1" />
```
- the URL to call in case of a forgotten password, if the mega authentication provider is used

```
<add key="ForgetPasswordUrl" value="http://my-server/hopex/account.aspx" />
```

In addition, the options that are normally retrieved from the SSP are read from the configuration file. These are not used by UAS but can be used by custom providers.

9. ANONYMOUS ENVIRONMENT MODE

In its nominal version, UAS is designed to work with an environment list on the login page but you can replace it with a textbox.

It can be used when you have multiple environments and you do not want that your user can see them.

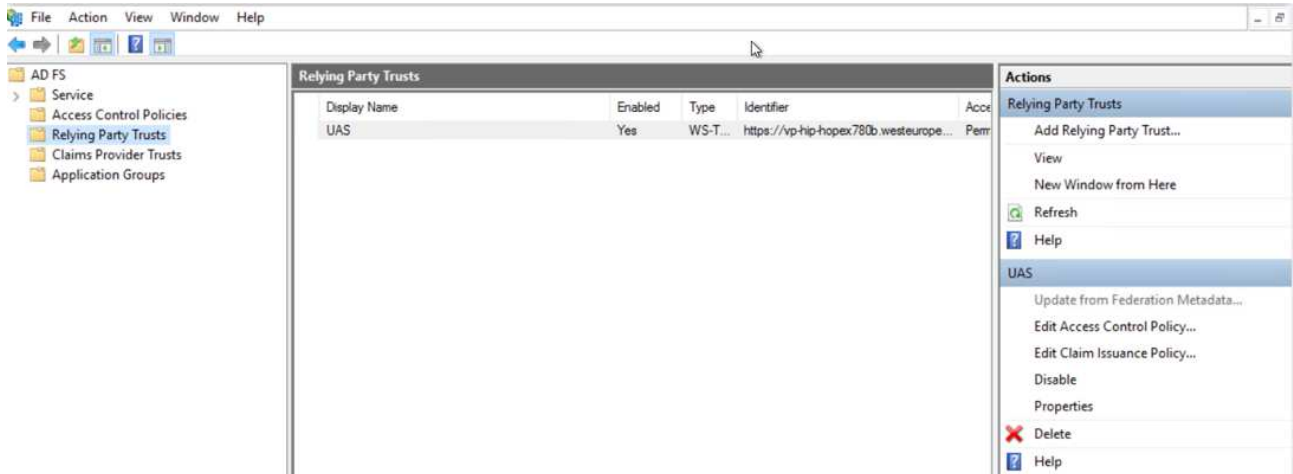
To do so, you must declare in the appsettings section of UAS web.config file:

```
<add key="IsAnonymousEnvironment" value="1"/>
```

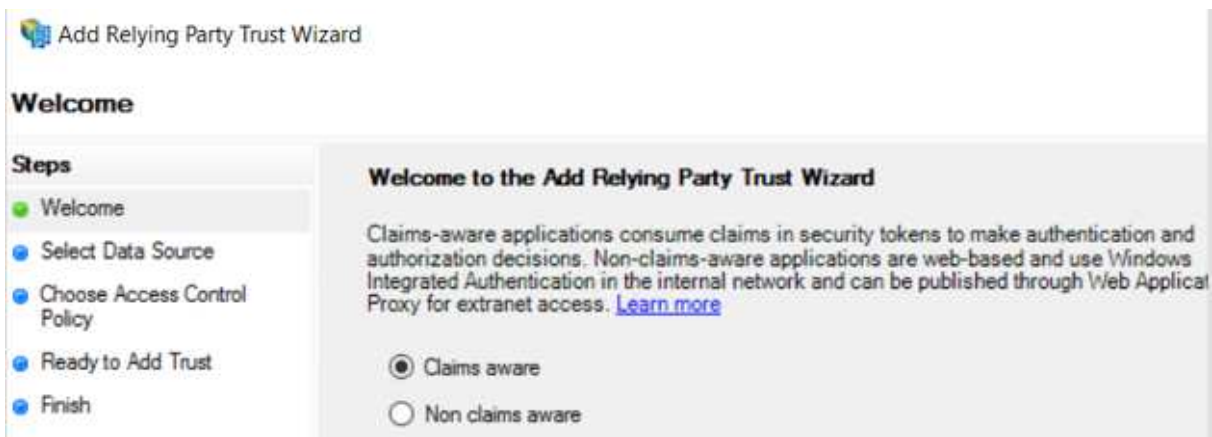
10. SAML2 ADFS SERVER CONFIGURATION

To create the Relay Party:

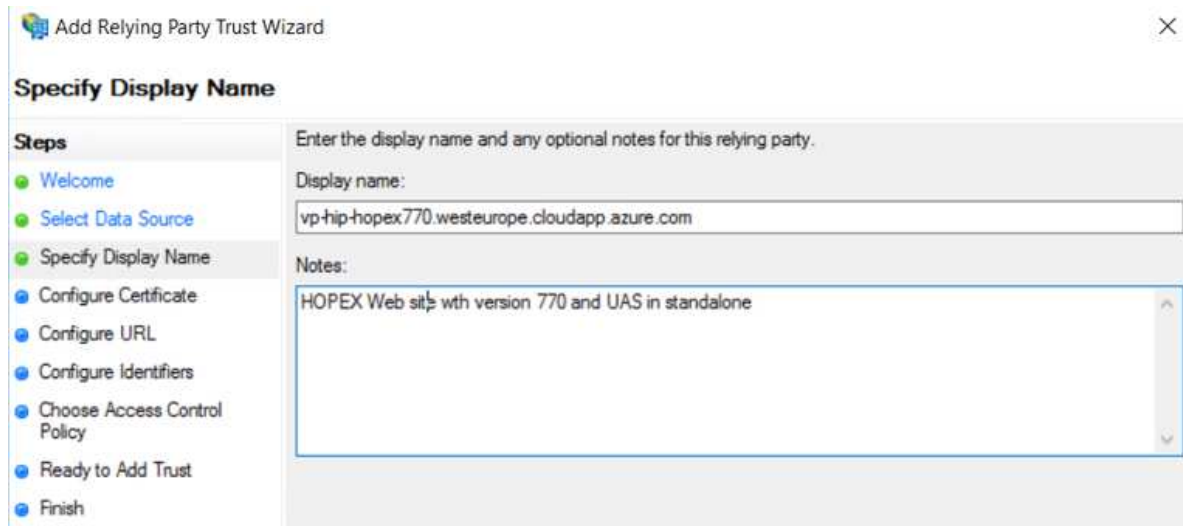
1. Start **AD FS Management** console.
2. In the **AD FS** folder, select **Relying Party Trust** folder.



3. In the **Actions** pane, in the **Relying Party Trusts** section, click **Add Relying Party Trust**.
4. In the **Welcome** page, select **Claims aware**.



5. In the **Specify Display Name** page, select Enter data about the relying party manually
6. Enter the Web Front-End server name.



Add Relying Party Trust Wizard

Specify Display Name

Enter the display name and any optional notes for this relying party.

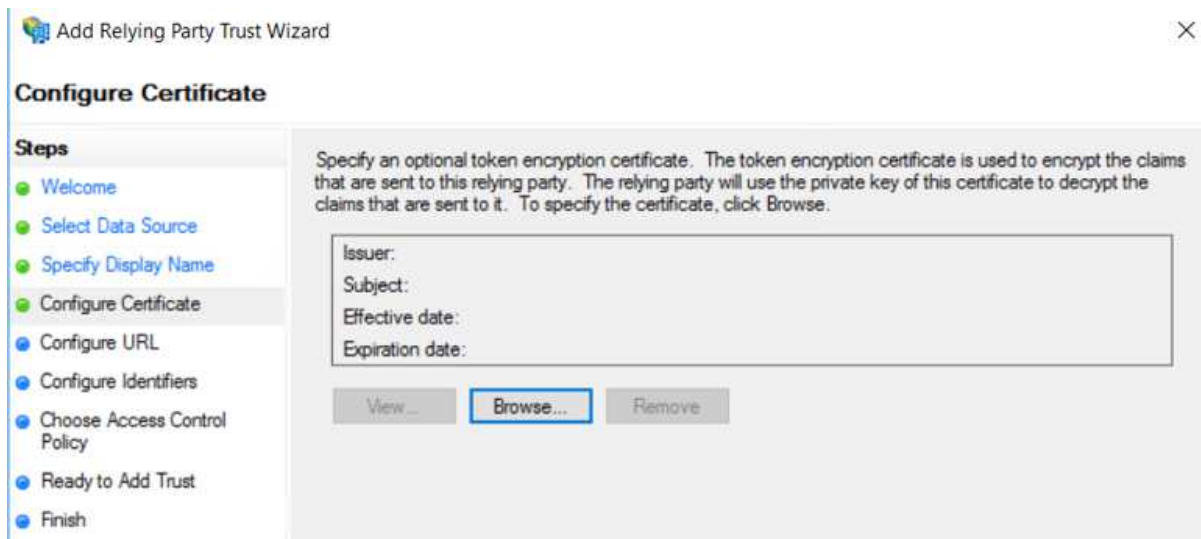
Display name:
vp-hip-hopex770.westeurope.cloudapp.azure.com

Notes:
HOPEX Web site with version 770 and UAS in standalone

Steps

- Welcome
- Select Data Source
- Specify Display Name
- Configure Certificate
- Configure URL
- Configure Identifiers
- Choose Access Control Policy
- Ready to Add Trust
- Finish

7. Click **Next**.



Add Relying Party Trust Wizard

Configure Certificate

Specify an optional token encryption certificate. The token encryption certificate is used to encrypt the claims that are sent to this relying party. The relying party will use the private key of this certificate to decrypt the claims that are sent to it. To specify the certificate, click Browse.

Issuer:
Subject:
Effective date:
Expiration date:

View ... **Browse...** Remove

Steps

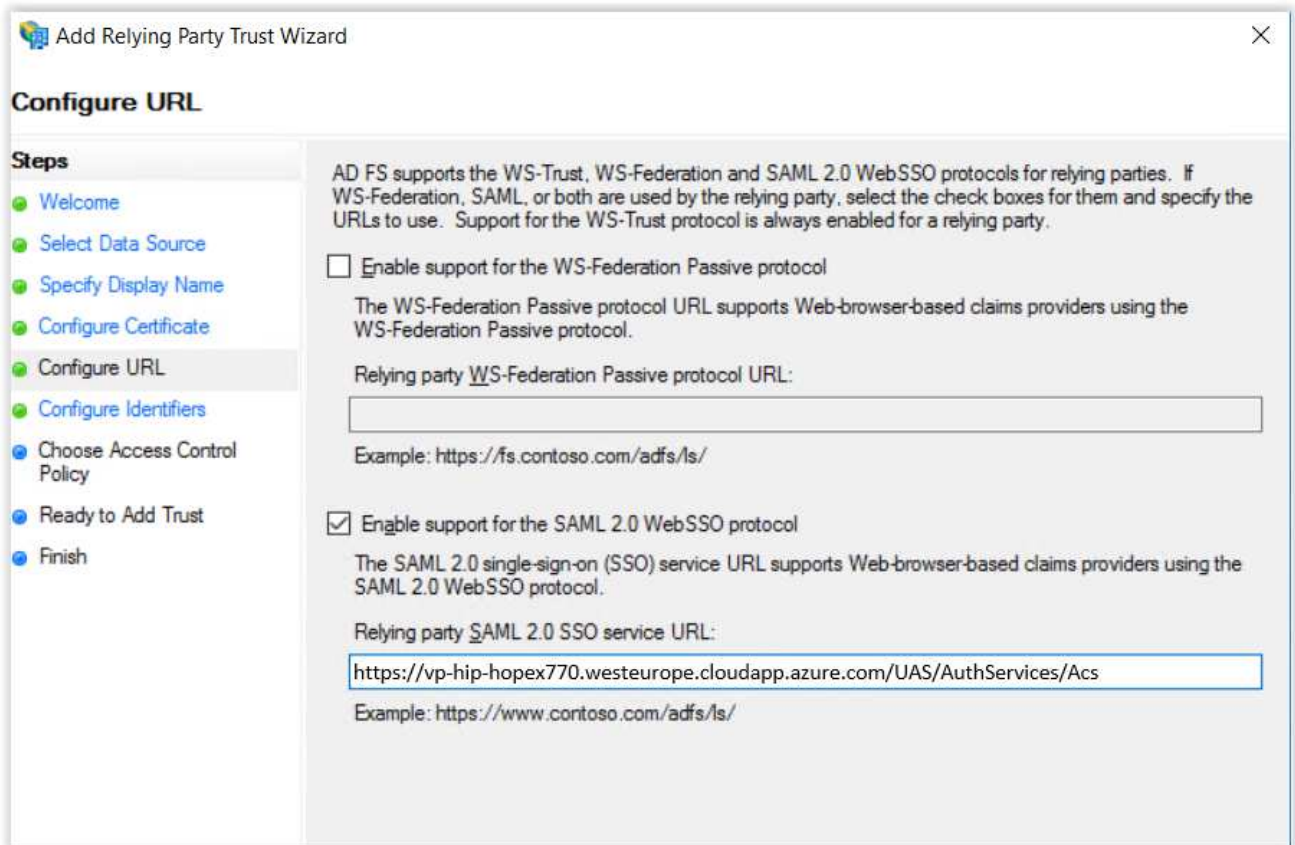
- Welcome
- Select Data Source
- Specify Display Name
- Configure Certificate
- Configure URL
- Configure Identifiers
- Choose Access Control Policy
- Ready to Add Trust
- Finish

8. Click **Browse**.

9. In the **Configure URL** page, select **Enable support for the SAML2 Web SSO protocol**.

10. Enter the **Relying party SAML2 SSO service URL**:

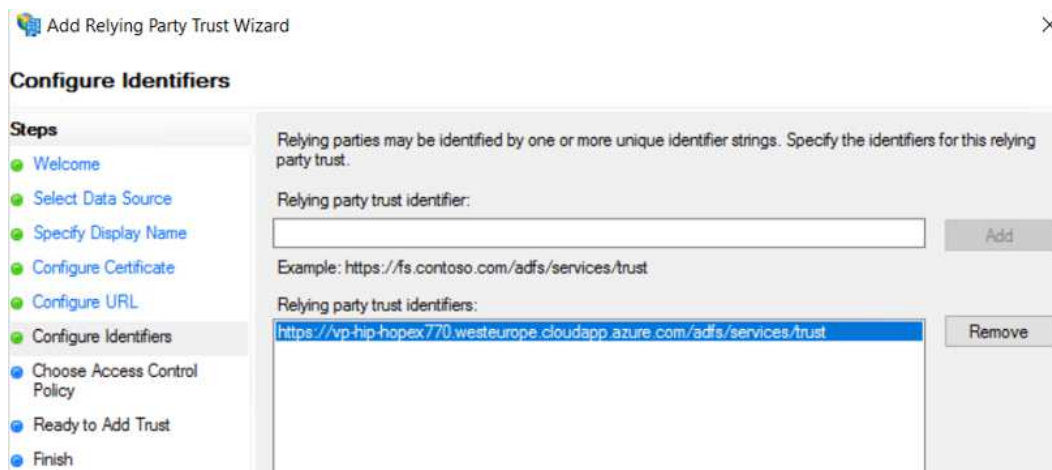
`http://{servername}/UAS/AuthServices/Acs`



11. In the **Configure Identifiers** page, enter the **Relying party trust identifier**:

`http://{servername}/UAS`

12. Click **Add**.



13. Click **Next**.

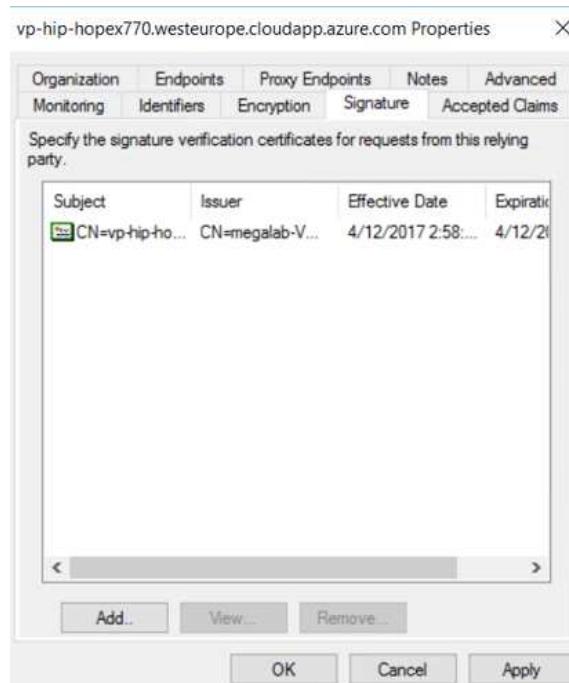
To add Signature certificate imported from the front server:

1. Import the certificate to the local storage.
2. Export the certificate without the private key.

To assign the certificate in ADFS:

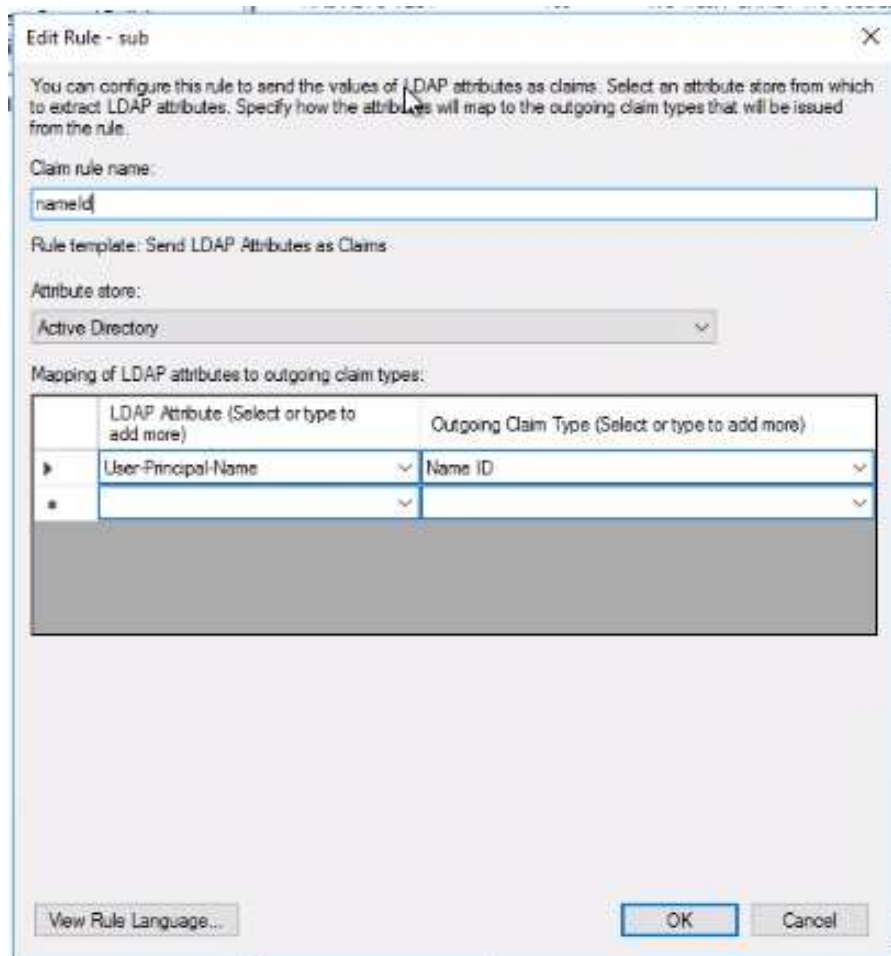
1. Right-click the Relay party.

2. In the **Signature** tab, select the imported certificate file
3. Click **OK**.



To configure claims to be returned to UAS:

1. Select the Relay and click **Edit Claim Issuance Policy**.
2. To send the user ID to UAS, click **Add Rule**.
3. Select "Send LDAP Attributes as Claims".
4. In the **Claim rule name** field, enter the name "nameid".
5. In the **Attribute store** drop-down list, select "Active Directory".
6. In the **Mapping of LDAP attributes to outgoing claim types**, select:
 - a. In **LDAP Attribute**: "User-Principal-Name" (you can choose another value, depending on which AD attribute is used as an identifier for the employees)
 - b. In **Outgoing Claim Type**: "Name ID"



Edit Rule - sub

You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:
nameid

Rule template: Send LDAP Attributes as Claims

Attribute store:
Active Directory

Mapping of LDAP attributes to outgoing claim types:

	LDAP Attribute (Select or type to add more)	Outgoing Claim Type (Select or type to add more)
▶	User-Principal-Name	Name ID
*		

View Rule Language... OK Cancel

7. Click **OK**.
8. To send additional claims to UAS, click **Add rule**.
9. Select "Send LDAP Attributes as Claims".
10. In the **Claim rule name** field, enter the name "Get LDAP Attributes".
11. In the **Attribute store** drop-down list, select "Active Directory".
12. Add attributes and transformation rules.

Edit Rule - Get LDAP Attributes

You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:

Rule template: Send LDAP Attributes as Claims

Attribute store:

Mapping of LDAP attributes to outgoing claim types:

	LDAP Attribute (Select or type to add more)	Outgoing Claim Type (Select or type to add more)
▶	E-Mail-Addresses	E-Mail Address
	Company	Company
	Department	Department
	Display-Name	Display-Name
	Employee-ID	Employee-ID

13. Click **OK**.
14. To send the groups to which the user belongs, click **Add rule**.
15. Select "Send LDAP Attributes as Claims".
16. Enter the name "Get Groups".
17. In the **Attribute store** drop-down list, select "Active Directory".
18. In the **Mapping of LDAP attributes to outgoing claim types**, select:
 - a. In **LDAP Attribute**: "Token-Groups-Qualified by Domain" (you can choose another value, depending on which AD attribute is used to store groups of users)
 - b. In **Outgoing Claim Type**: "Group"

Add Transform Claim Rule Wizard

Configure Rule

Steps
 Choose Rule Type
 Configure Claim Rule

You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:

Rule template: Send LDAP Attributes as Claims

Attribute store:

Mapping of LDAP attributes to outgoing claim types:

	LDAP Attribute (Select or type to add more)	Outgoing Claim Type (Select or type to add more)
▶	Token-Groups - Qualified by Doma...	Group
*		

< Previous Finish Cancel

19. Click **OK**.

To configure UAS to enable SAML2 Provider and to communicate with ADFS:

1. Open **HOPEX Administration**.
2. Right-click **HOPEX** and select **Options > Modify**.
3. Expand **Installation > Authentication > Identity Provider** folders.
4. Select **Identity provider** folder, and in the right pane, select **Activation of the SAML2 identity provider**.
5. Select **SAML2** folder and in the right pane, in:
 - the **Location of the metadata file**: enter your federationmetadata.xml URL.
 - the **Identifier of the SAML2 identity provider**: enter the ADFS URL.

11. WINDOWS AUTHENTICATION IN CLUSTER MODE WITH UAS

If you want to use windows authentication with UAS in an intranet network with a Network Load Balancer, you need some prerequisites:

1. Create an account service as follows:
 - a. This account must belong to **IIS_WPG** group.
 - b. Define the delegation level on "Trust the user for delegation to any service (Kerberos Only)".
 - c. Add this account service to **IIS_USRS** group on all the nodes of your cluster.
2. In the **appHost.config** file (c:\windows\system32\inetsrv\config), deactivate the kernel mode.
3. Activate the account service use: **useAppPoolCredentials="true"**.
4. Add read rights with account service on:
 - c:\inetpub\wwwroot
 - c:\inetpub\wwwroot\uas
 - c:\inetpub\wwwroot\hopex
 - c:\inetpub\wwwroot\windowsAuthenticationService
5. Add full control with account service on:
 - C:\Windows\Microsoft.NET\Framework64\v4.0.30319\Temporary ASP.NET Files
 - C:\Windows\Microsoft.NET\Framework32\v4.0.30319\Temporary ASP.NET Files
6. At DNS level:
 - a. Define a host name for the Network Load Balancer(NLB).
 - b. Define an alias host name by machine in the same Network Load Balancer domain.
 - c. In your port rules, put single affinity.
7. Configure the Service Principal Name (SPN):
 - a. Set SPN on the NLB by FQDN and Short Name with your account service
 - b. Set SPN on each node of your cluster by FQDN and Short name with your account service

Example:

```
setspn -S HTTP/NLBName.domain.com domain\webapplicationaccount (NLB)
setspn -S HTTP/NLBName domain\webapplicationaccount
setspn -S HTTP/ServerName1.domain.com domain\webapplicationaccount
setspn -S HTTP/ServerName1 domain\webapplicationaccount
setspn -S HTTP/ServerName2.domain.com domain\webapplicationaccount
setspn -S HTTP/ServerName2 domain\webapplicationaccount
```

12. OKTA CONFIGURATION

12.1. Configuring OKTA

To configure OKTA:

- 1) Connect to your OKTA account.
- 2) Go to **Admin Portal > Applications**.
- 3) Click **Add application**.
- 4) Click **Create New App**.
- 5) Select **Web platform and SAML2 sign on method**.
- 6) Click **Create**.
- 7) Enter the **General Settings** as you want.
- 8) Click **Next**.
- 9) Enter **Single Sign on URL** with the following URL syntax:
`http://<server name>/UAS/AuthServices/Acs`
- 10) Enter **Audience URI** and **Default Relay state** with the following URL syntax:
`https://<server name>/UAS`
- 11) In **Attribute statements**, do not forget to add an attribute named “sub” and its value will be your UAS login so you can choose user login or email.
- 12) Retrieve the SAML metadata and store it in UAS folder or secured folder in your network accessible by HTTP.

12.2. Configure UAS with OKTA

To configure UAS with OKTA:

- 1) Go to HOPEX Administration.
- 2) Right-click **HOPEX** and select **Options > Modify**.
(Check that you are in Extended view)
- 3) Expand **Installation > Authentication** folders.
- 4) Select **Identity Providers** and in the right pane, select “Activation of the SAML2 identity provider”.
- 5) Expand **Identity Providers** and select **SAML2**.
 - In the “**Contact email**” field: enter OKTA administrator email.
 - In the “**Location of the metadata file**” field: enter the UAS URL where you store the metadata retrieved before.
 - In the “**Identifier of the SAML2 identity provider**” field: enter the Url in the following format:
`http://www.okta.com/<youroktaid>`
 - In the “**Return URL**” field: enter the Hopex URL in the following format:
`https://<server name>/hopex`
 - In the “**Sign on URL**” field: Enter your Sign on URL:

https://<name of your organization>.okta.com/app/<appname>/<oktaid>/sso/saml

- Set the certificate friendly name and password if it is necessary.

OKTA	HOPEX Options
Identity Provider Issuer	Identifier of the SAML2 identity provider
Identity Provider Single Sign-On URL	URL for connection to the SAML2 identity provider
Identity Provider metadata	Url of the metadata file

13. TERMINOLOGY

13.1. Client

A client is a piece of software that requests tokens from UAS - either for authenticating a user or for accessing a resource (also often called a relying party or RP). A client must be registered with the OP.

Examples: Web applications, native mobile or desktop applications, Single Page Applications (SPA), server processes etc.

13.2. User

A user is a person who is using a registered client to access his or her data.

13.3. Scope

Scopes are identifiers for resources that a client wants to access. This identifier is sent to the OP during an authentication or token request.

By default, every client can request tokens for every scope, but you can restrict that.

They come in two flavors.

13.3.1. Identity scopes

Requesting identity information (aka claims) about a user, e.g. his name or email address, is modeled as a scope in OpenID Connect.

There is e.g. a scope called profile that includes first name, last name, preferred username, gender, profile picture and more. You can read about the standard scopes [here](#) and you can create your own scopes in UAS to model your own requirements.

13.3.2. Resource scopes

Resource scopes identify web APIs (also called resource servers) - you could have e.g. a scope named calendar that represents your calendar API.

13.4. Authentication/Token Request

Clients request tokens from the OP. Depending on the scopes requested, the OP will return an identity token, an access token, or both.

13.4.1. Identity Token

An identity token represents the outcome of an authentication process. It contains at a bare minimum an identifier for the user (called the sub aka subject claim). It can contain additional information about the user and details on how the user authenticated at the OP.

13.4.2. Access Token

An access token allows access to a resource. Clients request access tokens and forward them to an API. Access tokens contain information about the client and the user (if present). APIs use that information to authorize access to their data.

14. PROTOCOL SPECIFICATIONS

Open ID Specifications: <http://openid.net/connect/>

OAUTH2 Specifications: <https://tools.ietf.org/html/rfc6749>

SAML2 Specifications: <https://tools.ietf.org/html/rfc7522>

15. TROUBLESHOOTING

15.1. General

If you have problems with UAS, launch HOPEX Daily Logs, you need to activate “Hopex Web App Logs” to see UAS logs.

You can also use Hopex Monitor Console.

15.2. Get information about configuration

UAS use an SSP webservices to retrieve HOPEX options.

You can request it to know if your configuration is well defined.

You need to type this URL below if you want to retrieve a specific authentication configuration :

`http://<servername>/megassp/sitecfg.mgsp?skey=<serialkey>§ion=<sectionname>&key=<keyname>`

15.3. Redirect Server name to Full Qualified Domain name

To redirect Server name to Full Qualified Domain name:

1. Download IIS rewrite module 2 at:

http://download.microsoft.com/download/D/D/E/DDE57C26-C62C-4C59-A1BB-31D58B36ADA2/rewrite_amd64_en-US.msi

2. Install it.

3. Into system.webServer in the web.config file, add the following:

```
<rewrite>
  <rules>
    <rule name="Redirect2FQDN" stopProcessing="true">
      <match url="(.*)" />
      <conditions>
        <add input="{HTTP_HOST}" pattern="^(^\.]+$" />
      </conditions>
      <action type="Redirect" url="http<s>://{HTTP_HOST}.mega.com/hopex" />
    </rule>
  </rules>
</rewrite>
```

4. Change the domain **.mega.com** by yours.

15.4. Client configuration in Windows Authentication mode

To avoid the basic login popup, you must configure this website like a Local intranet website.

To configure the website like a local intranet website:

- 1) Open your "Internet browser".
- 2) Go to **Internet Options**.
- 3) Select **Security** tab.
- 4) Click **Local Intranet**.
- 5) Click **Sites**.
- 6) Click **Advanced**.
- 7) In the **Add this website to the zone** field, enter: "http://<domain name>".
- 8) Click **Add**.
- 9) Click **Close**.

15.5. Filtering Windows group

In Windows authentication mode, when you have too many Windows groups, you can filter them.

You can filter either by group number or by group name.

15.5.1. Filtering Windows group by number

MaxWindowsRoles enables to filter the Windows groups by number.

By default, the filter is disabled (value="0").

You can set this value to the number of Windows groups you want.

To filter Windows group by number:

- 1) In HOPEX installation folder, expand **wwwroot** > **WindowsAuthenticationService** folders.
- 2) Open the **Web.config** file.
- 3) In the **appSettings** section, set the **MaxWindowsRoles** value to the number of Windows group you want:

```
<appSettings>
  <add key="MaxWindowsRoles" value="<number of Windows groups>" />
  <!-- <add key="WindowsRoles" value="" />-->
</appSettings>
```

15.5.2. Filtering Windows group by name

Windows Roles enables to filter the Windows groups by name.

By default, the filter is disabled.

Values are group names separated with a " , ".

To filter Windows group:

- 1) In HOPEX installation folder, expand **wwwroot** > **WindowsAuthenticationService** folders.
- 2) Open the **Web.config** file.

- 3) In the **appSettings** section, remove the comment characters (`<!-- -->`) and add the group names separated with a `“;”`:

```
<appSettings>
  <add key="MaxWindowsRoles" value="0"/>
  <add key="WindowsRoles" value="GroupName1;GroupName2;GroupName3"/>
</appSettings>
```

ERROR: undefined
OFFENDING COMMAND: 84,75

STACK:

-mark-
/Rect
22
/SrcPg
-mark-

UAS Tools

1. PURPOSE	3
2. CONFIGURATION	3
2.1. Configuration page.....	3
2.1.1. <i>Getting Started</i>	4
2.1.2. <i>Identity Providers</i>	5
2.1.3. <i>CORS</i>	6
2.2. Result page	7
3. DIAGNOSTIC	8

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. PURPOSE

UAS Tools serves as an easy means to configure and diagnose UAS. It is made of two entities: the Configuration section and the Diagnostic section. Its purpose is thus twofold: prevent UAS configuration errors by validating options beforehand and provide insight on existing UAS errors by analyzing the log files.

You may access it via you web browser, at the following address:

`http(s)://<hopex_server_or_cluster_url>/uastools`

2. CONFIGURATION

The **Configuration** tab of UAS Tools displays a wizard that allows you to configure UAS options. It aims at showing and asking you the least information possible, as several options can be inferred from others.

You will input information in the home page (the configuration page) and if everything is valid, you will then be presented with the result page that will give you information on how to include the generated configuration files in UAS.

2.1. Configuration page

The configuration tab is split in three categories presented as a wizard:

- Getting Started
- Identity Provider
- CORS (Cross-Origin Resource Sharing)

You can go from step to step as you wish if there are no client-side validation errors.

If there are any errors at this point, they appear either below the corresponding field, or at the top of the related section.

The **CORS** section includes a publish button ("**Generate configuration**"), which submits all the form data to the server where they will be subject to more advanced validation. As with the client-side errors, server-side errors will be shown at their correct location, and the wizard will switch back to the first section.

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

2.1.1. Getting Started

Configuration Diagnostic
Documentation

Getting Started
Basic Settings
Identity Provider
Identity Provider Settings
CORS
Cross-Origin Resource Sharing Management

General Information (everything is required)

★Topology
Standalone

Front server

☐ Use HTTPS

★Front URL (load balancer or server name)
http://vp-dl-785int2

★Orchestrator server (SSP) Url
http://vp-dl-785int2/megassp

★Security key
.....

Signing certificate

★Certificate name
mega.com

★Certificate password
.....

Next

In the **General Information** section, you can select the topology of the server, which defaults to Standalone. In case of Cluster mode, some additional options pertaining to cache management are displayed.

The **Front server** section lets you input Hopex and SSP URLs, as well as the related Security Key.

- Front Server URL should indicate only the root address, i.e., do not add */hopex* or */uas*
- SSP URL is usually the same address as above, appended with */megassp*
- The Security Key is found in clear text during HOPEX install time, or encrypted in the UAS **web.config** file

The certificate section lets you input information concerning the certificate. It must be installed on each server UAS is installed on. This is especially important in Cluster mode.

2.1.2. Identity Providers

Configuration
Diagnostic
Documentation

Getting Started
Basic Settings
Identity Provider
Identity Provider Settings
CORS
Cross-Origin Resource Sharing Management

Please select at least one provider.

HOPEX
☒ Use HOPEX
[More information](#)

SAML2
☐ Use SAML2
[More information](#)

Microsoft
☐ Use Microsoft
[More information](#)

SalesForce
☐ Use SalesForce
[More information](#)

Windows
☐ Use IIS Windows
[More information](#)

Google
☐ Use Google
[More information](#)

Custom OpenID
☐ Use Custom provider
[More information](#)

Previous
Next

In the providers sections, you can add the authentication providers UAS will use. You need to select at least one authentication provider. Unlike almost all other sections in UAS Configuration tool, the following providers information cannot be validated on the server-side, so make sure they are correct:

- Microsoft
- Google
- Salesforce

Note: for now, the client secret for these providers will be written in plain text in megasite.ini for Hopex V2R1.

If you need help for a provider, click the “More information” link at the bottom of every section, which will open the UAS documentation directly on the correct section.

2.1.3. CORS

Configuration Diagnostic
Documentation

Getting Started
Basic Settings
Identity Provider
Identity Provider Settings
CORS
Cross-Origin Resource Sharing Management

Add or remove CORS as needed. Order is not important.

More information

Add Remove

URL
http://www.mega.com

Client
Hopex Web Site

Add Remove

URL
http://www.microsoft.com

Client
API

Add Remove

URL
http://www.apple.com

Client
Windows

Previous
Generate configuration

In the **CORS** section, you can add or remove CORS as needed, up to a maximum of 5, by using the Add and Remove buttons.

The order in which you add them is not important.

If you need help concerning CORS, click the **More information** link pointing to UAS documentation that will give you more details.

When you are done with everything, click **Generate configuration** to validate your changes and create a configuration file that will be used to configure UAS.

2.2. Result page

Configuration Diagnostic
Documentation

Configuration results

1 Replace or add this settings

```
<add key="AuthenticationUrl" value="http://vp-dl-785int2/uas" />
<add key="DelegatedLogin" value="2" />
```

Copy this settings and paste it in your appsettings section in your HOPEX web.config

2 Download the following file and update your megasite.ini contents

[Click here to download](#)

Copy this file content and paste it in your megasite.ini

After successful validation of the configuration, the **Result page** is displayed.

Depending on the Topology, you have specific actions to perform. You have to:

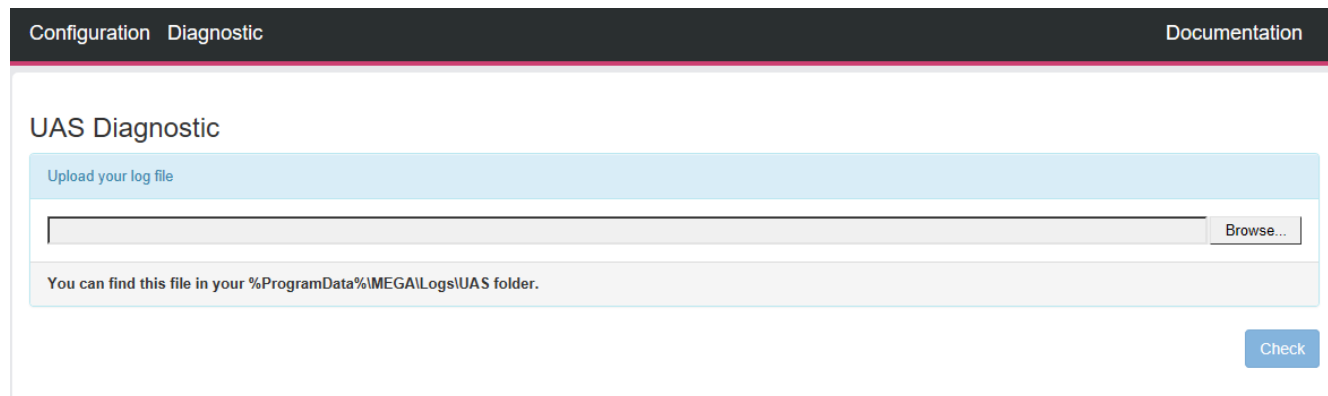
- Edit the HOPEX web.config and perform the changes displayed in the first section of this page (*Replace or add these settings*)
- For HOPEX V2R1
 - Download the megasite.ini file, populated with entries concerning UAS, that need to be injected in HOPEX megasite.ini
- For HOPEX V2
 - Download and overwrite UAS config.json file
- In cluster mode, install the same signing certificate on each node.

3. DIAGNOSTIC

As well as helping you configure UAS, UAS Tools also allows you to diagnose existing issues.

To diagnose existing issues:

1. In the UAS Tools menus, click the **Diagnostic** tab.



1. In the **Upload your log file** section, upload the log file you want to diagnose:
 - a. Click **Browse**.
 - b. In the dialog box: enter %ProgramData%\MEGA\Logs\UAS to reach the log file location.
 - c. Select the log file.

Your file is uploaded.

2. Click **Check** to analyze the log file.

UAS Diagnostic tool automatically loads the configuration from the log file (the last one present in the file), and it analyzes it too.

When UAS Diagnostic is done, the **Diagnostic result** page is displayed.

Diagnostic result

General

⚠ SSP Url
Impossible to contact SSP Website

[Edit Configuration](#)

Provider

Logging

i Your log file doesn't reveal any serious anomalies. If there is a real problem, please send the logs to Mega Support for investigation.

Log details :

Time	Source	Message
09:49:36,123	/LM/W3SVC/1/ROOT/UAS	Unhandled exception System.NullReferenceException: Object reference not set to an instance of an object. at Mega.UAS.DataComponent.CustomViewService.SanitizeEnvironments(HopexEnvironments environments) at Mega.UAS.DataComponent.CustomViewService.Login(LoginViewModel model, SignInMessage message) at IdentityServer3.Core.Results.LoginActionResult.<>c__DisplayClass2.<<.ctor>b_0>d_4.MoveNext() --- End of stack trace from previous location where exception was thrown --- at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task) at System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotification(Task task) at IdentityServer3.Core.Results.HtmlStreamActionResult.<Render>d__0.MoveNext() --- End of stack trace from previous location where exception was thrown --- at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task) at System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotification(Task task) at System.Owin.C...>

It is split in three main sections:

- **General** and **Provider** sections give you either the errors it encountered while analyzing the logs, or a message saying that the configuration is ok. You can edit the current configuration with the **Edit Configuration** link.
- **Logging** section shows a dump of the actual errors from the log file, with a message giving you advice on how to solve these issues, if necessary.