HOPEX System Blueprint

User Guide

HOPEX V2

INTRODUCTION

The aim of **HOPEX System Blueprint** is to assist architects in specification of applications by describing their behavior within enterprise IT architecture. Advanced product functions are illustrated by the example of a purchase request processing application.

HOPEX System Blueprint is designed to allow you to:

- Apply service-oriented architecture principles.
- Implement business process automation tools in the creation of applications.
- Prepare a specification.
- Integrate user interface design in analysis of services.

Specifying IS evolution projects with **HOPEX System Blueprint** allows you to:

- Describe a project and its application environment.
- Create project component use cases.
- Build a specification of services for each application and their interfaces as seen by users.
- Have this specification validated by future users as it is developed.
- Describe organization of communications between services using interaction scenarios
- Describe detailed sequence of tasks completed on execution of the application.
- Describe the data model associated with an application system.
- Generate reports enabling analysis of the different system design phases.

UML 2.3 Compatibility

HOPEX System Blueprint Includes UML 2.3 diagrams. This evolution responds to questions of compatibility but is also aimed at benefiting from the extended capabilities of UML 2.3 models.

In version MEGA 2009 SP5, is based on the specification of IT projects on UML 1.3. diagrams.

The majority of UML1.3 diagrams are constructions valid under UML2.3.

HOPEX System Blueprint methodology differs little with this new version. Changes intervene on two diagram types:

- the interaction scenario diagram is replaced by the UML 2.3 sequence diagram.
- the state diagram is replaced by the state machine diagram.

Old UML 1.3 diagrams can still be used thanks to an option enabling working with UML1.3 standard. To activate this option:

1. Select **Tools** > **Options**.
2. In the left pane of the options window, expand the **Compatibility** folder then double-click the **Diagram** sub-older.
3. In the right pane, select the option **Display interaction scenario diagrams and state diagrams**.

An advanced description of UML 2.3 diagrams is available in the **HOPEX UML** guide.

CONNECTING TO HOPEX SYSTEM BLUEPRINT

☛ ***HOPEX System Blueprint*** *This function is available with the* ***Windows Front-End*** *technical module only.*

To connect to **HOPEX System Blueprint**, see in **HOPEX Common Features** guide, chapter"The MEGA Desktop" > "Accessing to MEGA".

About This Guide

This user guide complements the quick start guide and covers the various functions offered by **HOPEX System Blueprint** to prepare specifications of an IT project.

Guide Structure

It contains the following chapters:

- "Defining Project Objectives and Requirements", page 23 explains how to describe objectives and requirements covered by a project and how to connect these to the enterprise context: organizational and IT.
- "Creating use cases", page 37 explains how to describe use cases of IT services concerned by a project.
- "Identifying Project Objects and Scope", page 47 presents modeling of a summary diagram representing the main components impacted by the project, as well as the nature of these impacts.
- "Modeling Use Case System Processes", page 57: explains how to model IT processes implemented in the different use cases of an application or service..
- "Describing Sequences of Applications and Services", page 69 presents interactions, enabling representation of the services architecture to supplement or validate it.
- "Describing User Interfaces of Services", page 81 describes interfaces connecting services or operations with the exterior.
- "Analyzing Project Data", page 85 describes use of analysis reports in the framework of IT system design.

Additional Resources

This guide is supplemented by:

- **HOPEX Common Features** guide describes the basic functions common to **HOPEX** products and solutions.

  ☛ *It can be useful to consult this guide for a general presentation of the interface.*
- the **HOPEX Data** describes data models.
- the **HOPEX Power Supervisor** administration guide.
- More advanced technical functions are described in the **HOPEX Power Studio** guide.

Conventions Used in the Guide

Styles and formatting

Remark on the preceding points.

Definition of terms used.

A tip that may simplify things.

Compatibility with previous versions.

Things you must not do.

Very important remark to avoid errors during an operation.

Commands are presented as seen here: **File > Open**.

Names of products and technical modules are presented in bold as seen here: **HOPEX**.

CREATING A SYSTEM PROCESS DIAGRAM

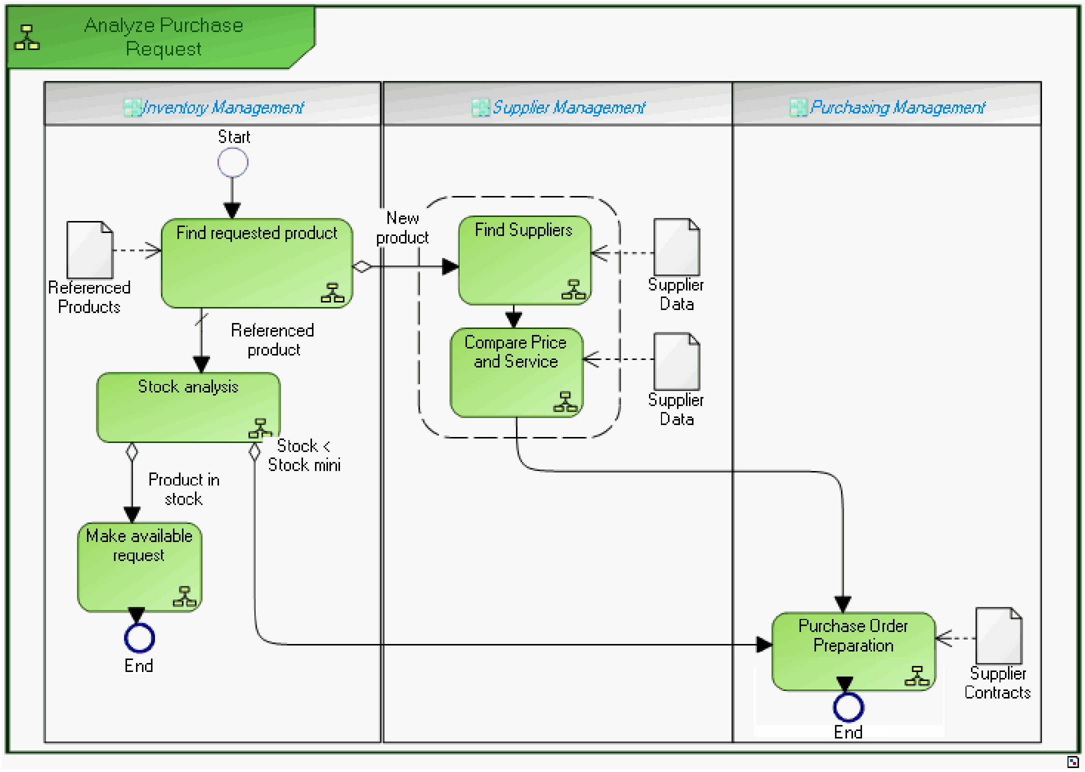
This chapter presents the basic functions offered by **MEGA** to model a system process.

You will learn how to:

- ✓ define your work context
- ✓ create a system process and its diagram
- ✓ define system process participants and the tasks they execute
- ✓ add comments to objects present in the diagram

This presentation is based on the example of purchase requests represented below.

- A product search is carried out from the referenced products repository.
- If the product is referenced, stock is analyzed.
- If stock is sufficient, a "Make available" request is activated and the process ends.
- If stock is less than minimum stock, a purchase order is sent to the supplier.
- If the product is new, search for a supplier and comparative study of prices and services is carried out. An order is then sent and the process ends.



CREATING A NEW SYSTEM PROCESS

You will learn to define your work context and create your first process.

Work Context

The context in which you will work is defined by a library.

In the framework of this Quick Start guide you will create a specific library.

To create a library:

1. In the left part of the **MEGA** workspace, select the tab of the **Home** navigation window.
2. In the navigator, right-click the **Default Library** folder.
3. Select **New** > **Library**.
   The **Creation of Library** dialog box appears.
4. Enter the name of your library.
5. Click **OK** to close this dialog box.
   The library appears in the **Home** navigation window and becomes the default library.

   ☛ *Under normal circumstances, a work library in an enterprise is defined by the project manager.*

Creating a System Process

A system process is the executable representation of a process. the events of the workflow, the tasks to be carried out during the processing, the algorithmic elements used to specify the way in which the tasks follow each other, the information flows exchanged with the participants.

The system process diagram uses notation proposed by BPMN standard.

To create the "Analyze Purchase Request" process from a library:

1. Right-click the library you have just created.
2. Select **New** > **System Process**.
   The **Creation of System Process** dialog box appears.
3. Enter the name of the system process. "Analyze Purchase Request".
4. Click **Finish** to close the dialog box.
5. Expand the tree of your library to show the new system process.

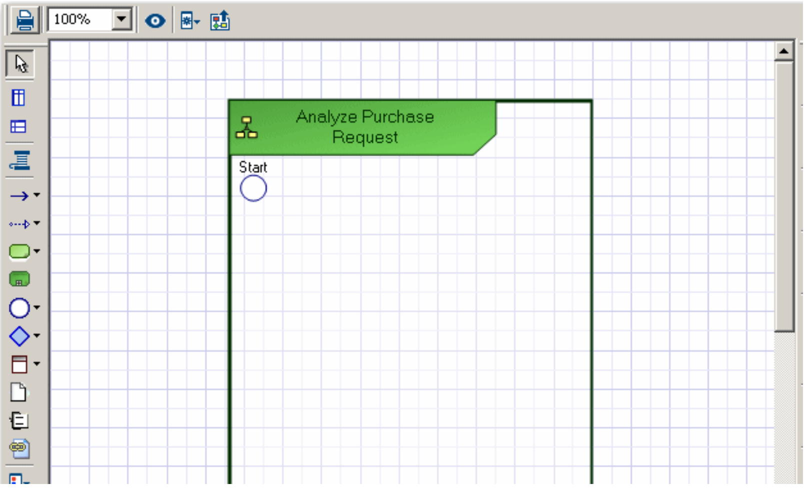
DRAWING A SYSTEM PROCESS DIAGRAM

In **MEGA**, system processes are described by diagrams.

Creating a System Process Diagram

To create a system process diagram:

1. Right-click the system process name and select **New > Diagram**.
2. In the window that opens, select **System Process Diagram**, confirm that the **Diagram initialization** check box is selected and click the **Create** button.
   The diagram window opens. You are now in the **MEGA** graphic editor.



The diagram is initialized. The current process, represented by a frame, is placed in the diagram.

Creating Objects

In this section you will create the objects to be placed in your diagram: participants in the process and tasks executed with their sequence flows and message flows exchanged.

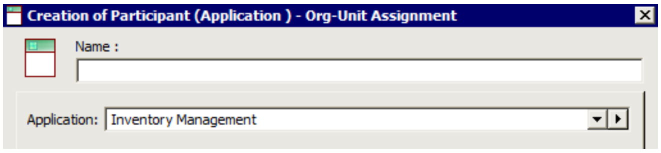
Creating participants

You will define the participants that carry out the process steps.

☛ *Column presentation simplifies diagram reading, though other presentations are possible.*

To create an actor participating in the process:

1. Click the arrow on the right of the **Participant** button in the objects toolbar, and in the drop-down list select **Participant (Application)**.
2. Click in the diagram workspace within the system process frame.
   The **Creation of Participant (Application)** dialog box opens.



3. To assign an application to this participant, enter the application name in the field **Application**, for example "Inventory Management".

   ☛ *Given that you are working on an empty library, you must create an application.*
4. Click the **Close** button.
   The participant is positioned in the diagram and carries the name of the assigned application.

In the same way, create the other applications participating in the process: "Supplier Management" and "Purchasing Management".

To position the participant in the frame of the described process:

- Click the title bar of the participant and, while holding down the mouse button, move the object.
  You can, for example, position it at top left of the organizational process frame.

To enlarge frames of participants so that their height is the same as that of the process:

1. Select the three participants.
2. Click the point at the center of the bottom line of the frame, and holding the mouse button down, drag the frame line downwards.

Creating Tasks

A task is a step in a process.

You will create tasks and connect each of these to the application responsible for its execution.

To create a task:

1. In the diagram insert toolbar, click the **Task** button
2. Click in the diagram within the "Inventory Management" actor frame.
   The task appears in the diagram.

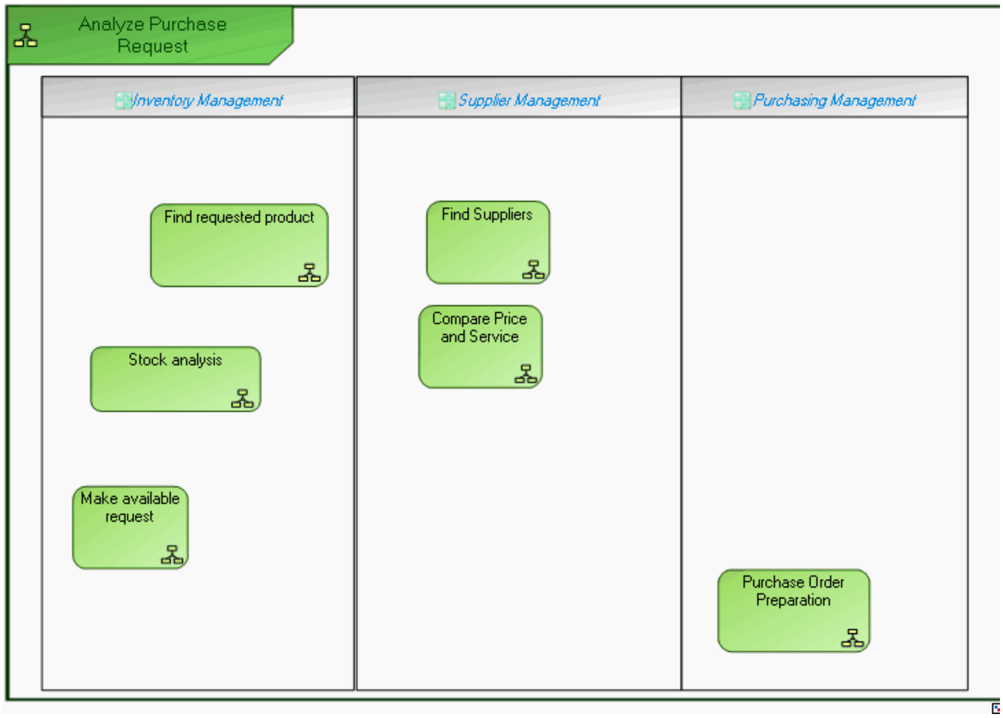
To assign a task to another participant:

1. Move the task from one participant to another, for example from "Inventory Management" to "Supplier Management".
   The frame of the destination participant is highlighted.

   ☛ *When placed, the task is disconnected from "Inventory Management" and connected to "Supplier Management", indicating that "Supplier Management" will execute the task.*

2. Now move the task back to the "Inventory Management" frame to restore the original link.

In the same way, create the "Stock Analysis", "Make Available Request", "Find Suppliers", "Compare Price and Service" and "Purchase Order Preparation" tasks as shown in the diagram below.



Creating events

You will create a start event and the two end events.

To create the process start event:

1. In the objects toolbar, click the **Event** button.
2. Click in the frame of the described process.
   The **Creation of Event (GRC)** dialog box opens.
3. Select **None** in the list of proposed types and click **Next**.
4. Select **Start** in the list on the next page and click **Finish**.
   An event named "Start" appears in the diagram.

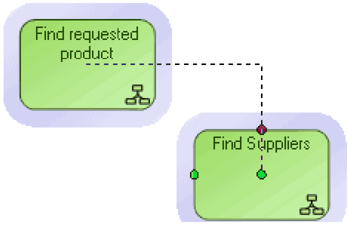
Proceed in the same way to create the end events.

Describing task sequence flow

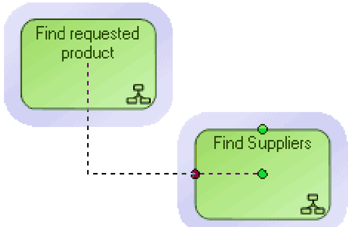
You will create links representing the order of execution of tasks.

To create a sequence flow between two tasks:

1. Click on **Sequence Flow** button.
2. Click the sender task and draw a link to the destination task.
   The two tasks are highlighted and a dotted line indicates the path that will be taken by the graphic link.



☛ *Several paths are possible, you must simply move the cursor to within the frame of the destination task.*



Similarly create all sequence flows connecting tasks of the process, as well as the flows between events and tasks.

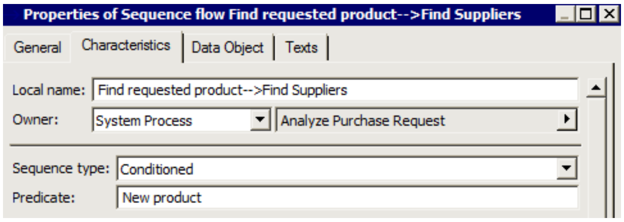
Defining a condition on a sequence flow

Depending on whether it is a new product or a product already referenced, the "Find Requested Product" task can result in either a search for a supplier, or a stock analysis. You will indicate these alternatives in the diagram.

To define a condition on the sequence flow between the "Find Requested Product" task and the "Find Suppliers" task:

1. Right-click the sequence flow and select **Properties**.
2. In the dialog box that appears, select the **Characteristics** tab.
3. Click the arrow at the right of the **Sequence Type** box.
4. Select "Conditioned" in the drop-down list.

5. Enter the condition expression in the **Predicate** text box, for example "New Product".



6. Click **OK**.
   The text associated with the condition appears on the condition which then takes form .

To specify that the sequence flow between "Find Requested Product" and "Stock Analysis" is used by default:

1. Right-click the sequence flow and select **Properties**.
2. In the dialog box that appears, select the **Characteristics** tab.
3. Click the arrow at the right of the **Sequence Type** box and select "Default" in the drop-down list.
4. Click **OK**.
   The sequence flow then takes form .

Creating data objects

You can add data objects to the diagram. A data object is used to explain how documents, data, and other objects are used and updated during the system process.

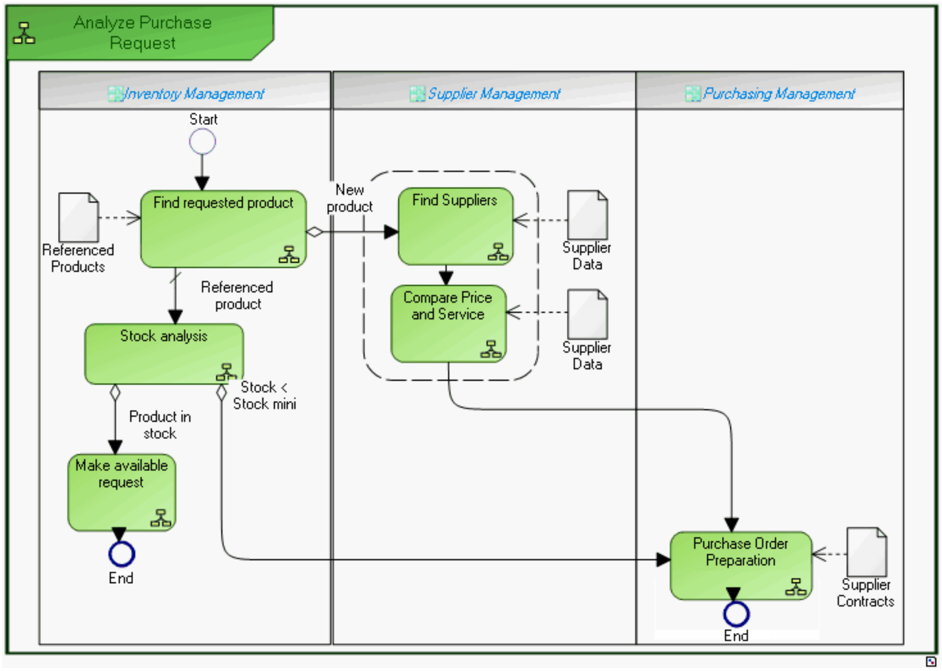
Data objects are created using the **Data Object** button in the diagram objects toolbar.

Once created, you must connect them to the tasks that use them.

To connect a data object to a task:

1. Click the **Link** button in the diagram objects toolbar.
2. Click the data object and, holding the mouse button down, draw a link to the destination task.
3. Release the mouse button.

Create the model below.



Saving a diagram

Save your drawing using the **Save** button.

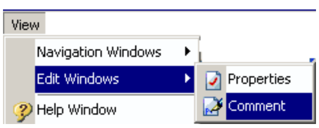
You can now print your diagram by selecting **File > Print**.

Entering Object Comments

Comments provide more information on the flowchart when such useful details do not appear in the drawing. These comments are generated in the document associated with the diagram.

You should enter a comment for each object in the diagram.

To avoid having to open and close the properties dialog box for each object in the diagram, select **View** > **Edit Windows** > **Comment**.



The Comment dialog box appears at the bottom of the screen.

It shows the comment of an object when it is selected.

Each time you click on an object in your diagram, the Comment window displays the comment for the object. You can also modify the comment in this window.

☛ *You can also enter a comment from the **Texts** tab of the object properties dialog box. In the dialog box that opens, select **Comment** in the drop-down list and enter your comment.*

DEFINING PROJECT OBJECTIVES AND REQUIREMENTS

HOPEX System Blueprint is based on a methodological approach, the first step of which is to identify and represent objectives and requirements leading to revision of part of the company information system.

Requirements are then related to components concerned, whether these be technical or organizational.

This chapter describes how to use the functionalities of **HOPEX System Blueprint** to clearly define the scope of your project.

The following points are covered here ::

Example Overview

This guide is based on the example of purchase request processing introduced in the **HOPEX Business Process Analysis**guide.

Organization of this process is modified to take account of urgent purchase requests.

HOPEX System Blueprint enables specification of information system evolutions necessary for implementation of the new version of this process.

Business context presentation

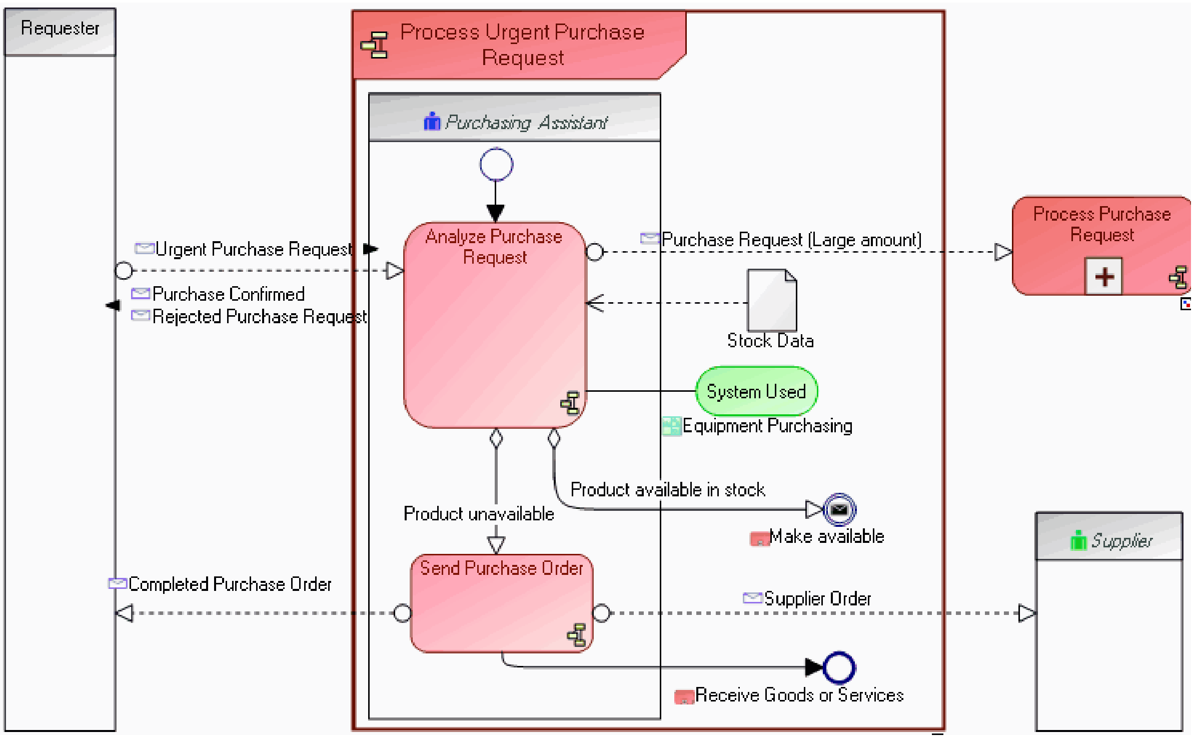
The example used is based on the purchasing process of a company whose activity is boat rental and cruise sales.

Company purchasing processes are split between operational purchase processes and investment purchase processes. Operational purchasing processes include purchase of spare parts and raw materials required for boat maintenance and repair.

Availability of boats being a vital element of company activity, a specific organizational process is defined to process urgent purchases for boat maintenance and repair.

Organizational Context

The purpose of the new organizational process for processing urgent purchase requests is to increase responsibilities of the purchasing assistant, so as to be able to send purchase orders at very short notice.



"Process Urgent Purchase Request" Organizational Process

The purchasing assistant begins by analyzing the purchase request. If the amount is large, the normal process purchase request process is implemented.

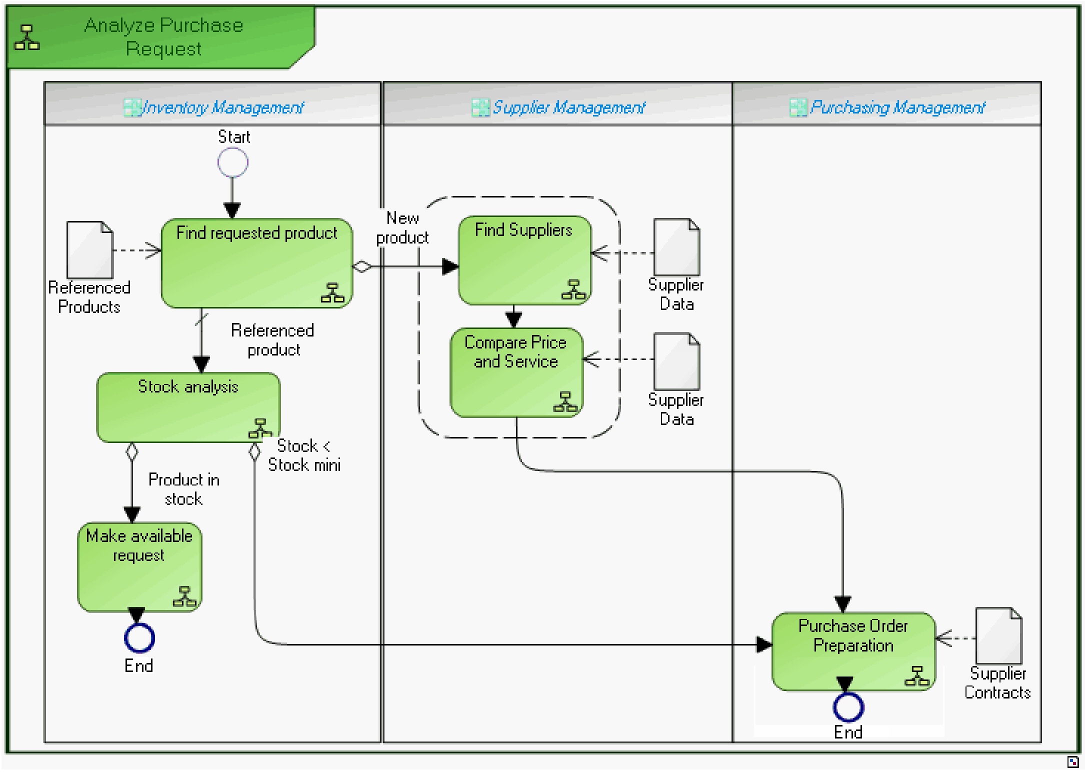
If the product is available, the assistant sends the availability request.

If not, the assistant sends a purchase order to the supplier. The rest of the processing is carried out within the framework of the normal "Process Purchase Request" organizational process.

To allow the purchasing assistant to rapidly analyze the purchase request and send the order, he/she is provided with a new application enabling access to data concerning stock and suppliers.

System process presentation

The following system process diagram proposes a first draft of operation expected of the new application.



A product search is carried out from the referenced products repository.

If the product is referenced, stock is analyzed.

If stock is sufficient, a "Make available" request is activated and the process ends.

If stock is less than minimum stock, a purchase order is sent to the supplier.

If the product is new, search for a supplier and comparative study of prices is carried out. An order is then sent and the process ends.

CREATING PROJECTS

HOPEX System Blueprint allows you to automatically build the package structure recommended for projects.

Creating a Project from a Library

You can use libraries to organize your IT projects in the HOPEX repository.

> *Libraries are groupings of objects enabling organization of contents of a HOPEX repository.*

> ☛ *For more details on libraries, see the **HOPEX Common Features** guide, "Libraries" chapter.*

To create an object from a library:

1. Select the **Home** navigation window.
2. Right-click the portfolio and select **New > Others**.
   The object type selection dialog box opens.
3. In the **Metaclass** box, select Project.
4. Click **OK**.
   The **Create Project** dialog box appears.
5. Enter the name of the project.
6. Click **Finish**.
   The project is created and added to the list of projects in the library.

> ☛ *For more details on Projects, see the **HOPEX Common Features** guide, "Projects" chapter.*

Initializing the project HOPEX System Blueprint

To clearly organize information relating to an IT infrastructure evolution project, **HOPEX System Blueprint** relies on *package* mechanisms.

> *A package partitions the domain studied and the associated work. It enables grouping of various elements, in particular use cases and classes. A package can also contain other packages. Packages are interconnected through contractual reports defining their interface.*

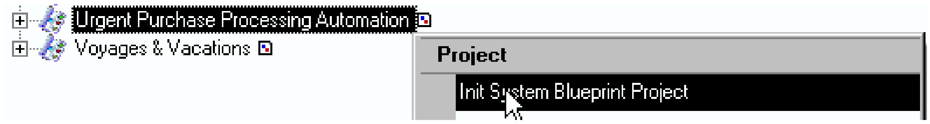
The package associated with the project allows you to store referenced elements in sub-packages such as one for UML actors and another for use cases.

These packages determine project scope; at generation of deliverables on a project, only the elements contained in these packages are analyzed.

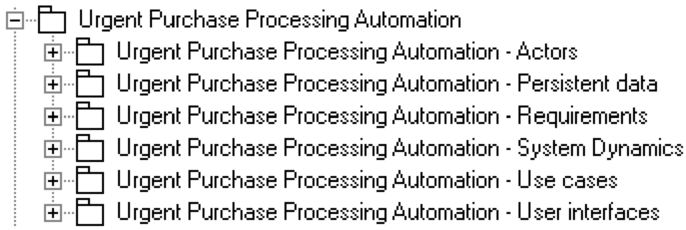
To simplify use of packages, **HOPEX System Blueprint** enables automatic creation of all packages recommended for projects.

To initialize your project:

1. Right-click the project and select **Initialize System Blueprint Project**.

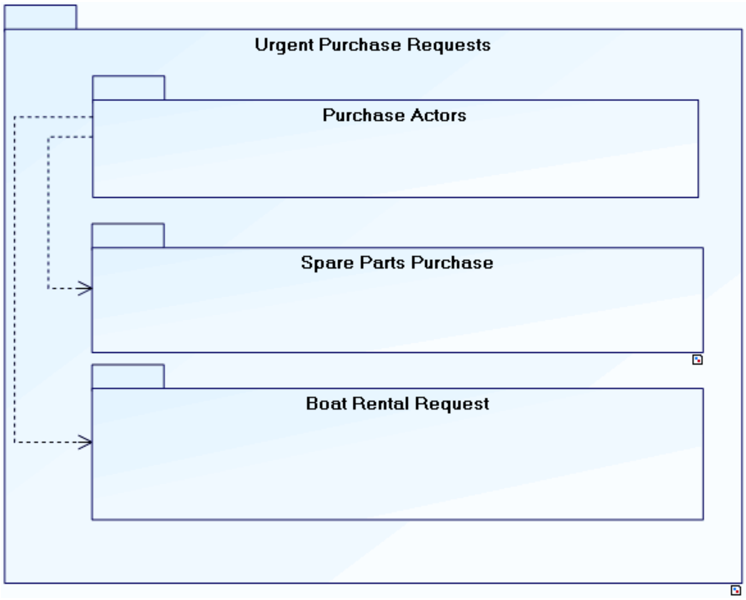


2. Expand the project folder, then the **Content** folder to view the package folder that carries the project name, and the sub-packages created by default.



Creating sub-packages

You can create sub-packages in a package to classify objects in finer detail, for example actors of a project



Urgent purchase requests are provided to process purchase of spare parts and boat rental requests. In both of these cases, users are actors of the purchasing domain.

☛ *For more details on the use of packages, see the **HOPEX UML** guide.*

You can create sub-packages:

- from the pop-up menu of the package
- or from a package diagram.

To create a sub-package from a package pop-up menu:

1. Right-click the package to open its pop-up menu and select **New > Package**.
   The create package dialog box appears.
2. Specify the package name and click **OK**.
   Expand the package folder to view the sub-package you have just created.

Establishing Objectives and Requirements

To ensure that evolutions specified with **HOPEX System Blueprint** meet project expectations, the main concepts used can be linked to requirements.

The first task in creation of a *project* is to inventory objectives involved and requirements imposed, and to connect these to the main organizational and technical components.

A project is a part of a system whose study is entrusted to the same team.

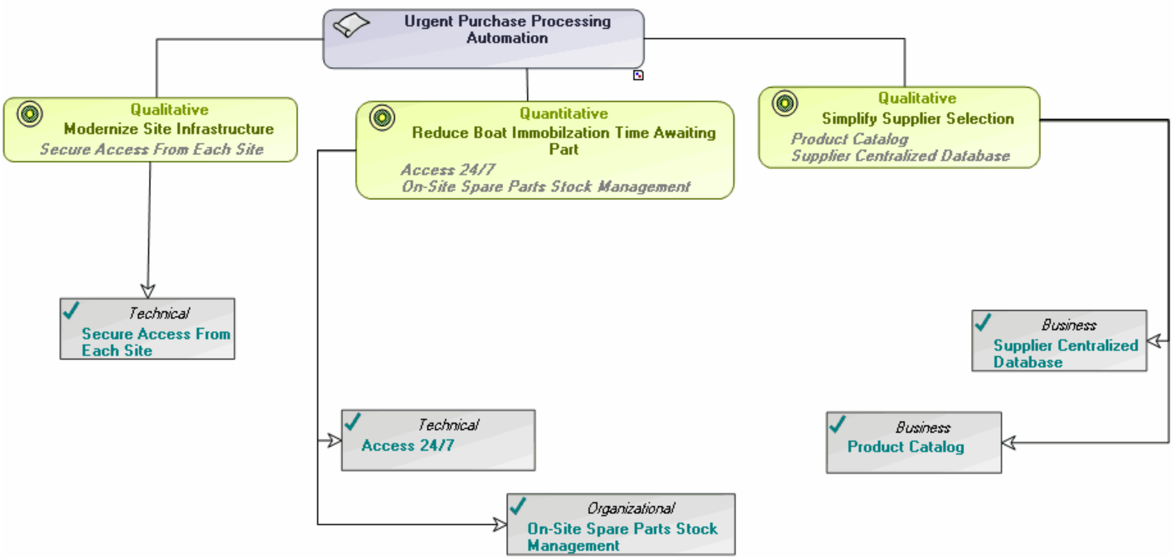
Objectives and Requirements Example

Our example is based on the purchasing process of a company whose activity is boat rental and cruise sales. We wish to develop the system to take into account urgent purchase requests.

The project of evolution of an application supporting the processing of urgent purchase requests has several objectives:

- From a functional viewpoint, this application must ensure reduction in boat immobilization time due to a spare part requirement. Purchase request processing must therefore be accessible 24 hours a day, since company representation offices are spread over several continents, and 7 days a week since the company activity is leisure and tourism.

- To reduce delays in the search for suppliers of spare parts already referenced or not, the purchasing department has decided to set up a product catalog and a centralized supplier database.
- The project also meets a technical objective, which is to modernize and improve security of communications between company headquarters and the representation offices.




```
In this example, the project meets the quantitative
objective of reduction in boat immobilzation time, and the
qualitative objectives of improvement in the communication
system on the one hand and the management of suppliers on
the other.
```

```
These objectives are linked to technical or system
requirements, such as 24/24 and 7/7 access to the process
urgent purchase request application.
```

Creating an Objective and Requirement Diagram

With **HOPEX System Blueprint**, you can also create your *objective* and *requirement* diagram from a new project integrated in the library of components concerned.

An objective is a goal that a company/organization wants to achieve, or is the target set by a process or an operation. An objective allows you to highlight the features in a process or operation that require improvement.

A requirement is a need or expectation explicitly expressed, imposed as a constraint to be respected within the context of a project.

This project can be a certification project or an organizational project or an information system project.

Creating a project objective and requirement diagram

Objectives and requirements associated with a project for automating the processing of urgent purchase requests can be represented by a diagram.

To create a new project objective and requirement diagram:

1. Right-click a project icon and select **New > Diagram**.
2. In the wizard that opens, select **Project Objective and Requirement Diagram** and click the **Create** button.
   An empty diagram appears.

> ☛ *For more details on Projects, see the* ***HOPEX Common Features*** *guide, "Projects" chapter.*

> ☛ *For more details on objective and requirement diagrams, see the* ***HOPEX Common Features*** *guide, chapter "Objectives and Requirements".*

To introduce the project representation in the diagram.

1. Right-click the icon of the project in the navigation tree and, holding the mouse button down, drag the object into the diagram.
2. Release the mouse button.
   The project appears in the diagram.

Creating project objectives from the objective and requirement diagram

To create an *objective*:

1. Click the **Objective** button in the diagram insert toolbar, then click in the diagram.
   The add objective dialog box appears.
2. Enter the name of the objective and click **Create**.
   The objective appears in the diagram and in the project navigation tree.

> ☛ *If the described project is positioned in the diagram, a link between the project and the objective automatically appears.*

Creating project requirements from the objective and requirement diagram

To create a *requirement*:

1. In the diagram insert toolbar, click the **Requirement** button then click in the diagram.
   The add requirement dialog box appears.
2. Enter the name of the requirement and click **Create**.
   The requirement appears in the diagram.

> ☛ *If the described project is positioned in the diagram, a link between the project and the requirement automatically appears.*

To specify the objective associated with a requirement:

1. Click the Link button.

2. Click the objective to position the link and, holding the left mouse button down, draw a link to the requirement.

   ☛ *The link can also be created from the requirement to the objective.*

3. Release the mouse button.
   A link from the objective to the requirement appears in the diagram. The name of the requirement appears within the shape of the objective.

Connecting Technical and Organizational Components to Requirements

Reminder of principle

The technical and organizational components to which the requirements relate can be specified.

For example, the "Purchasing Management" application must respect requirements related to setting up a product catalog and a centralized supplier database.

These links can be created graphically in the objective and requirement diagram.

Adding an application in the objective and requirement diagram

To connect an application to a requirement in an objective and requirement diagram, we must first show the applications in the diagram since they are not displayed by default.

To view applications in a project objective and requirement diagram:

1. Click the **Views and Details** button in the diagram toolbar.
2. Select the **Applications, IT Services, Databases** check box.
3. Click **OK**.
   The corresponding buttons appear in the insert toolbar.

To insert an application in the objective and requirement diagram:

1. In the objects toolbar, click the **Application** button.
2. Click on the diagram.
   The add application dialog box appears.
3. Click the arrow at the right of the **Name** box and select **Propose** in the drop-down list.
   The list of applications accessible from the current library appears.
4. Select the required application.
5. Click **OK**.
   The application appears in the diagram.

Connecting an application to a requirement

Connecting an application to a requirement can have two non-exclusive meanings:

- Either the application ***contributes*** to satisfaction of a requirement,
- Or the application is ***constrained*** by the requirement.

To graphically specify that an application contributes to satisfaction of a requirement :

1. Click the reel button in the insert toolbar.
2. Click the application and, holding the mouse button down, draw a link to the requirement.
3. Release the button.
   The directional link appears as a dotted line from the application to the requirement.

To graphically specify that an application is constrained by satisfaction of a requirement:

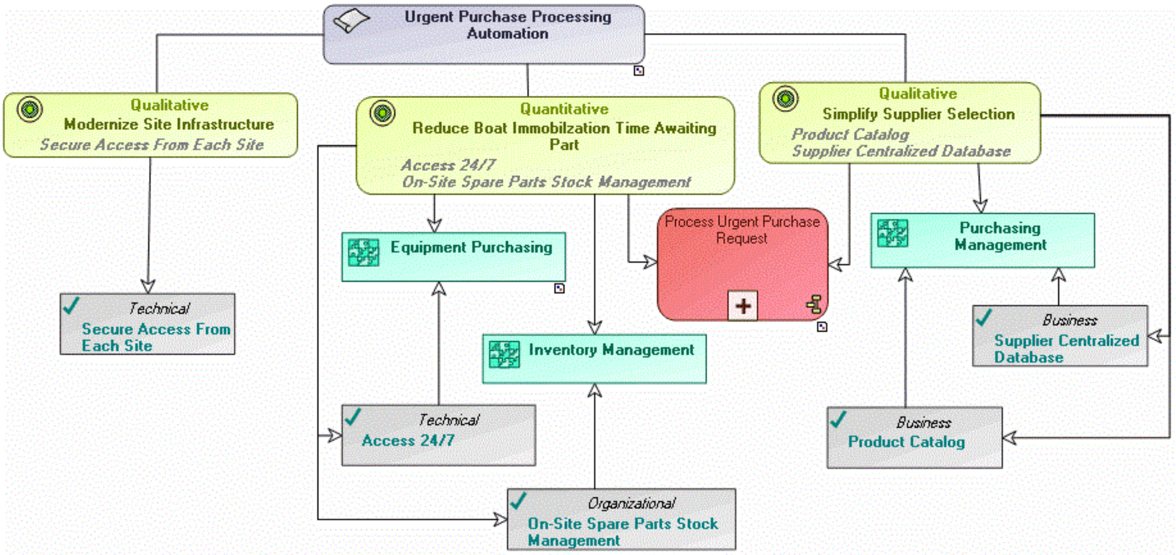
1. Click the reel button in the insert toolbar.
2. Click the requirement and, holding the mouse button down, draw a link to the application.
3. Release the button.
   The directional link appears from the requirement to the application.

Opening an Objective and Requirement Diagram

To open an objective and requirement diagram, for example, the diagram describing the "Urgent Purchase Processing Automation" project from the "Urgent purchases" Library:

1. Right-click the project to open its pop-up menu.

2. Double-click **Project Objective and Requirement Diagram**.
The diagram opens.



The urgent purchase processing process must respect company objectives related to reducing boat immobilization time and simplifying access to suppliers and products.

In addition, the equipment purchasing application must be accessible 24/7. The inventory management application must integrate the requirements for setting up product catalog and supplier database. Finally, the inventory management application must integrate the on-site spare parts management requirement.

☛ *To show application and organizational components in an objective and requirement diagram, you must have activated the corresponding view from* ***View > Views and Details****.*

☛ *Organizational processes are available with the* ***HOPEX Business Process Analysis*** *product.*

3

CREATING USE CASES

Creating use cases of project components enables representation of the different use contexts of the system. From a use case, you can describe how a functionality or set of functionalities is implemented within an application or service.

By means of use cases, **HOPEX System Blueprint** enables addition to description of an application architecture of the detailed specifications of its different sub-systems.

To simplify the identification of an application system use case, you can base your approach on the definition of requirements. Requirements are constraints to be respected within the framework of an application evolution project.

The following points are covered here:

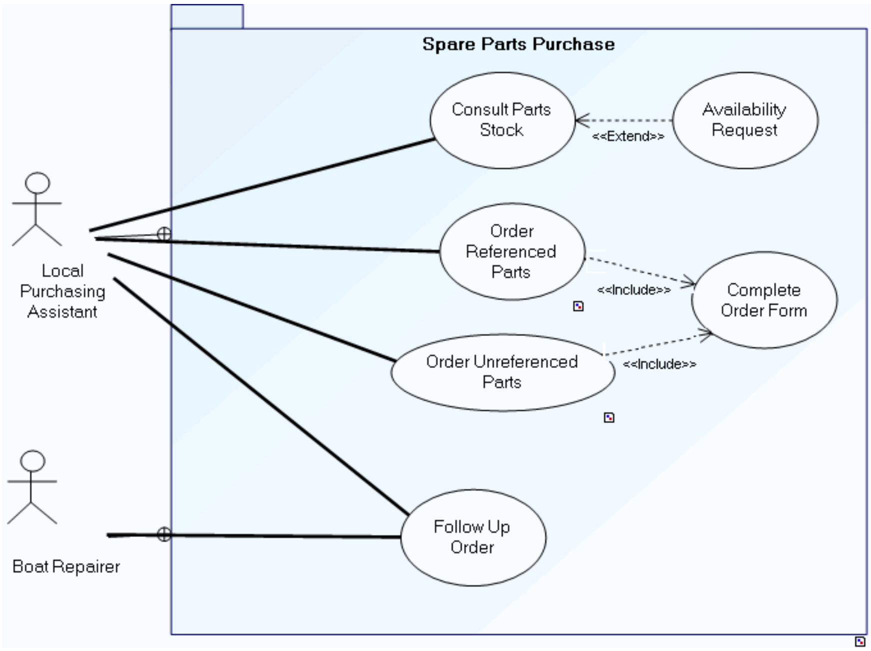
Opening a Use Case Diagram

A *use case* diagram enables description of interactions between a system and actors of the organization.

A use case is a series of actions leading to an observable result for a given actor. Scenarios illustrate use cases for example.

To open an existing use case diagram, that of the "Spare Parts Purchase" package, for example :

1. In the pop-up menu of the package, select **Use Case Diagram**.
2. The diagram opens.



The system is used to consult parts in stock and to order new spare parts.

Consultation of parts in stock is carried out by the local on-site purchasing assistant. Following consultation, the assistant can make an availability request.

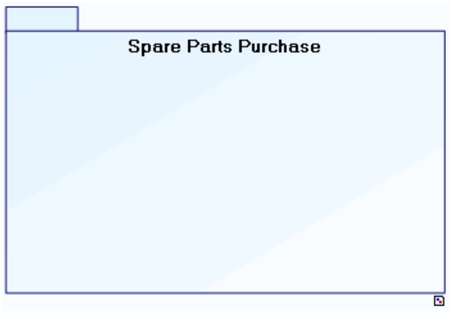
Two order types are possible, one for parts already referenced, the other for parts as yet unreferenced. In both cases, an order form should be completed.

Order follow-up is assured by the local purchasing assistant and the boat repairer.

Creating a Use Case Diagram

To create a use case diagram from a project package:

1. Right-click the package concerned and in its pop-up menu select **New** > **Diagram**.
2. Select **Use Case Diagram**, confirm that the **Diagram initialization** check box is selected, and click **Create**.
   The diagram opens in the **HOPEX** Edit window. The frame of the package is positioned within the drawing.



Use Case Actors

An actor represents the role played by something or someone within the enterprise environment of the modeled system. It is related to the business activities of the enterprise, and interacts with the system in different use cases. It can be an element in the enterprise structure such as a division, a department, or a workstation.

Creating org-units

To create an *actor* in a use case diagram:

1. In the use case diagram objects toolbar, click the **Actor (UML)** .
2. Then click in the diagram work area.
   The **Add Actor (UML)** dialog box opens.

3. Enter the actor name and click the **Create** button.

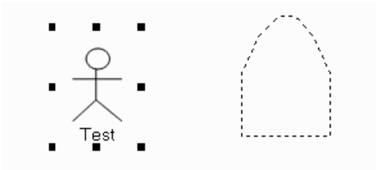
   ☛ *If the name you enter corresponds to an actor that already exists in the repository, the* ***Create*** *button becomes* ***Connect****.*

   The actor then appears in the diagram.

   ☺ *You can create several elements successively without clicking in the toolbar each time by double-clicking the button. To return to normal mode, press the Esc key or click on a different button in the toolbar, such as the arrow.*

To move an actor in the diagram:

1. Click the actor to select it.
   Black points delimit the selected actor, a frame appears around the object and the form of the cursor changes. This frame moves with the mouse to indicate where the object will be inserted when the button is released.
2. Hold the left mouse down to move the object.
3. Release the mouse button when the actor is positioned.



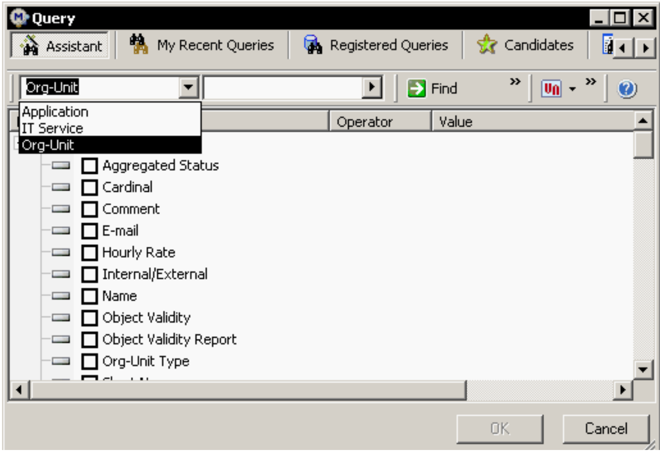
To assign a participant to an actor:

Having created an actor, you can assign it a participant such as an actor of the enterprise, a service or an application.

To assign a participant to an actor:

1. Right-click the actor concerned.
2. In the pop-up menu that appears, select **Properties**.

3. In the **Actor Player** tab of the actor properties dialog box, you can for example connect an existing object by clicking the **Connect** button . The query dialog box appears.



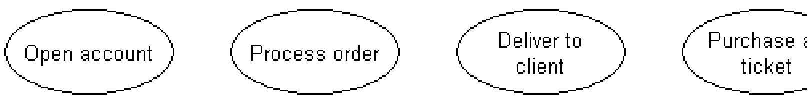
4. In the list of actors proposed, select the actors that interest you and click **OK**.
   Names of selected actors appear in the diagram below the UML actor.

   ☛ *If the object you wish to assign to the actor does not yet exist in the repository, you can create it using the **New** button*

Use Cases

A use case is a series of actions leading to an observable result for a given actor. Scenarios illustrate use cases for example.

Examples of *use cases*: processing an order, delivering to a client, opening an account, sending an invoice, establishing credit, purchasing an airline ticket, etc.



To create a use case in a diagram:

1. In the diagram objects toolbar, click the **Use Case** button
2. In the diagram, click within the package frame in the position where you wish to place the use case.
   The **Add Use Case** dialog box appears.
3. Enter the use case name and click the **Create** button.
   The use case appears in the diagram.

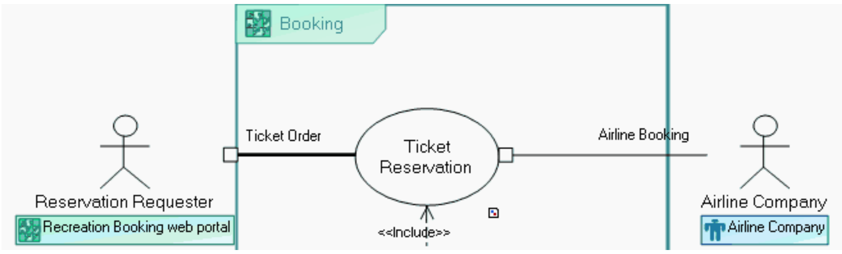
Participations

Actors are connected to use cases with which they are involved by means of *participations*.

A participation indicates that an actor plays a role in a use case.

You can indicate that an actor participates in a use case as initiator or contributor by using a participation.

Example:

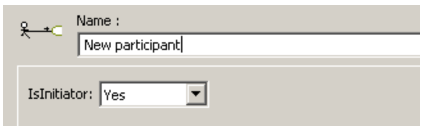


- The "Booking Requester" actor intervenes in "Ticket Booking" as Initiator.

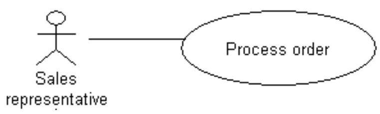
  The participation link appears in bold when the actor is initiator of the participation.
- The "Airline Company" actor intervenes in "Ticket Booking" as Contributor.

To create a participation:

1. In the diagram objects toolbar, click the button.
2. Click the actor concerned, and drag the mouse to the use case before releasing the button.
   The **Create Participation** dialog box appears.
3. Enter the name of the new participation if required.
4. Specify the actor participation type in the **IsInitiator** box.



5. Click **OK**.
   The link representing the participation appears in the diagram.



USE CASE ASSOCIATIONS: EXTENSIONS AND USES

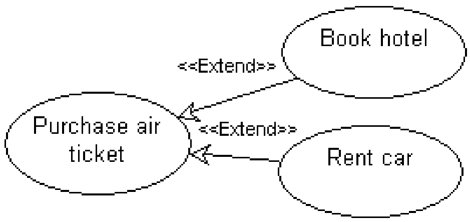
When the system to be described is large, it is useful to have modeling mechanisms that can be adapted to the desired level of detail.

Definitions

When a use case includes several alternatives or exceptions, these are represented separately as relationships that extend the standard use case.

Examples of *extension*:

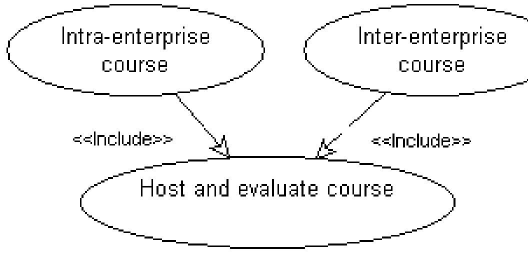
- The purchase of an airline ticket can also include booking a hotel room or renting a car.



An inclusion association enables insertion into a particular use case of behavior common to several use cases. This use case only has meaning when used as part of another use case.

Examples of *inclusion*:

- In a training organization, the "Host and Evaluate Course" use case is common to use cases:
  - Inter-enterprise course (where the participants come from several different companies)
  - Intra-enterprise course (where the participants all come from the same company)



- In a commercial enterprise, the "Issue Order" use case can reuse the following use cases:
  - Provide client information
  - Place a manufacturing order
  - Propose a payment method

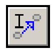
Connecting Use Cases

To indicate that one use case is an extension of another:

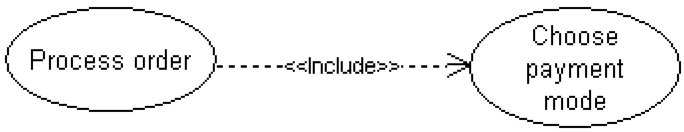
1. In the diagram insert toolbar, click button 
2. Click the use case, for example "Consult Catalog" and drag the mouse to the extension case, for example "Process Order" before releasing the mouse button.

The link appears in the diagram, labeled “Extend”.

To indicate that one use case includes another:

1. In the diagram insert toolbar, click button 
2. Click the use case, for example "Process Order" and drag the mouse to the extension case, for example "Choose Payment Mode" before releasing the mouse button.

The link appears in the diagram, labeled “Include”.



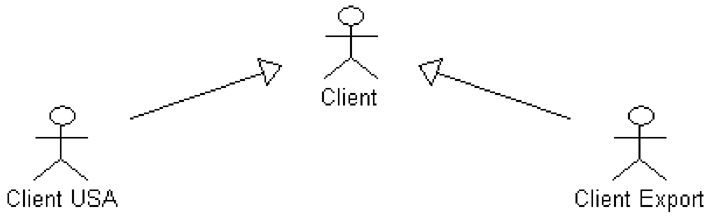
GENERALIZATIONS

A generalization represents an inheritance relationship between a general class and a more specific class. The more specific class is fully consistent with the more general class and inherits its characteristics and behavior. It can however include additional attributes or associations. Any instance of the more specific class is also an instance of the general class.

The notion of *generalization*, initially used for classes, has been extended to actors and use cases.

Examples of generalizations between actors:

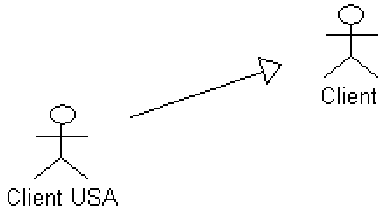
The "Client" actor can have the USA or Export specialization.



To create a generalization between actors:

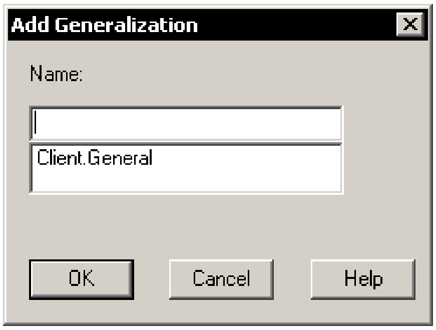
- Click the button and drag the link from the specialized actor (eg: USA client) to the more general actor (eg: Client).

The generalization then appears in the drawing.



☛ *In the same way you can create a generalization between two use cases.*

When creating a second generalization, a dialog box allows you to reuse the existing generalization if it involves the same subject.



IDENTIFYING PROJECT OBJECTS AND SCOPE

Having defined system objectives and use cases, you can define the main impacts of the project on the organization and its IT system.

To do this, the project impact diagram enables creation of a summary of the main applications, databases and entities impacted by the project.

This summary is based on existing system architecture on the one hand, and analysis of data used on the other.

At the end of this step, company management can decide whether or not to proceed with the project.

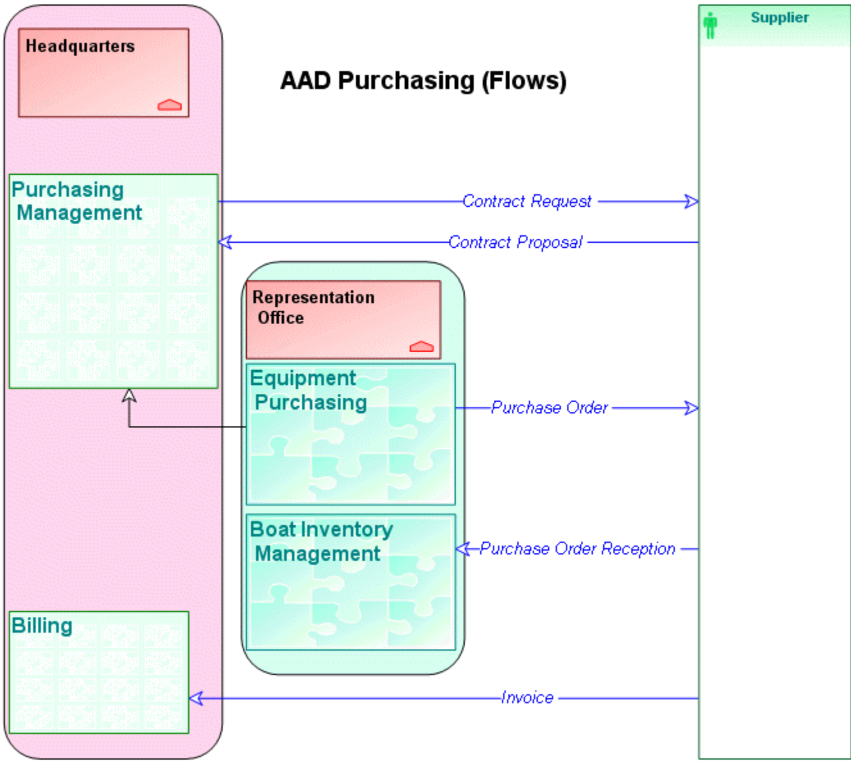
The following points are covered here:

PRESENTING EXISTING SYSTEM ARCHITECTURE

Existing application architecture can be represented by a diagram grouping the main applications and flows exchanged in the system under study.

An application architecture diagram is created from an application via command **New** > **Diagram**.

The following example presents the internal architecture application diagram l"ADD Purchasing (Flows)" of the "Purchasing Management System" application.



Equipment purchases for boat maintenance are made directly by the Representation Office, using the "Equipment Purchasing" application. This application calls a "Purchasing Management" application located centrally at Headquarters.

If the purchase involves a new type of equipment or a new supplier, a contract request is sent to the supplier using the application at Headquarters. The supplier returns a contract proposal.

In the other cases, a representative from the Representation Office sends a purchase order to the supplier using the "Equipment Purchasing" application.

Reception of purchases delivered from the supplier is handled by the "Boat Inventory Management" application at the Representation Office.

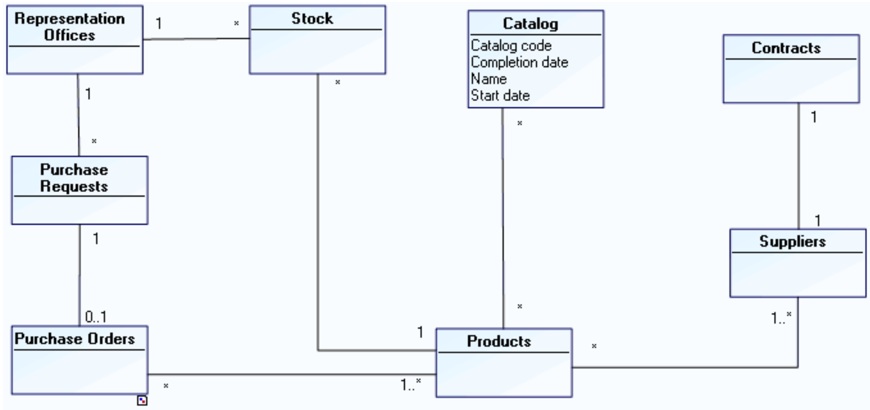
The invoice sent by the supplier is managed in the "Billing" application at Headquarters.

PRESENTING DATA

Description of the architecture can be supplemented by that of data. The project data model, built in preliminary study phase, presents the main entities involved.

The representation with **HOPEX System Blueprint** of new or existing data for a project is carried out from a data model.

The data model of the "Purchase Request Automation" project is presented below.




```
The application manages purchase requests, orders and
product stock levels in each of the representation offices.
A centralized catalog of products and suppliers is
installed.
Contracts with referenced suppliers are also accessible
from the application.
```

☛ *For more details on creation and updating of a data model, see chapter "Modeling Data" in the* ***HOPEX Data*** *guide.*

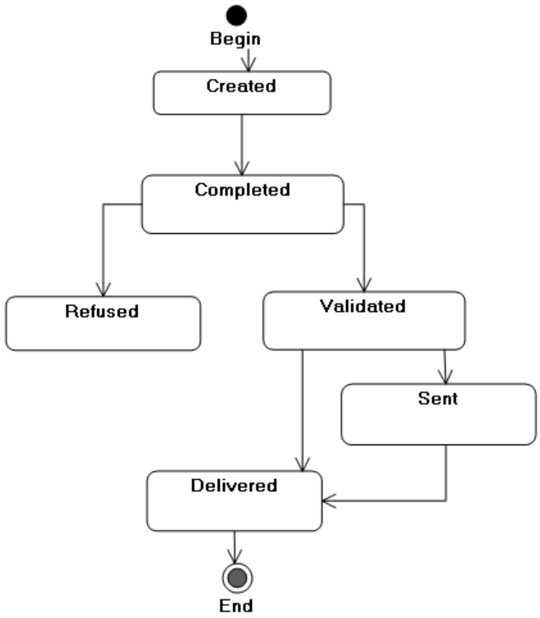
Specifying Entity Behavior

Behavior of a particular entity can be represented by a state diagram.

A state diagram enables representation of the different states of an entity, as well as possible transitions from one state to another.

You can create the state diagram of an entity from its pop-up menu, via command **New** > **Diagram**.

Here for example is the diagram for the "Purchase Order" entity.




```
A purchase order is first created, then completed by the
user. It is then either refused by the purchasing
representative who analyzes it, or validated. Having been
validated, the order is sent and goods are delivered. If the
product is available from stock, it is delivered directly.
```

☛ *For more details on the use of state diagrams, see the guide **HOPEX UML**.*

UML 2.3 Compatibility

Standard UML 2.3 replaces the state diagram by the state machine diagram.

A state machine is the set of states and state transitions which define behavior of an object.

To create the state machine of an entity and the corresponding diagram:

- Right-click the entity and select **New** > **State Machine**.

☛ *For more details on the creation of a state machine diagram, see the **HOPEX UML** guide.*

Presenting Project Impact

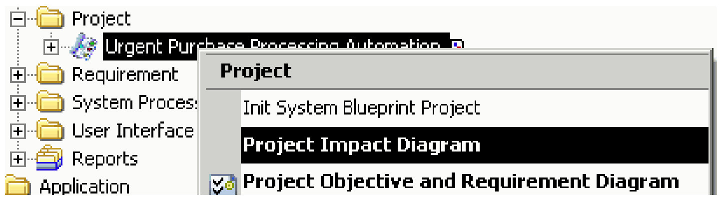
The impact diagram enables summary representation of information gathered on applications, databases and *entities* impacted by the *project*.

A project is a part of a system whose study is entrusted to the same team.

Accessing the Project Impact Diagram

To access the project impact diagram:

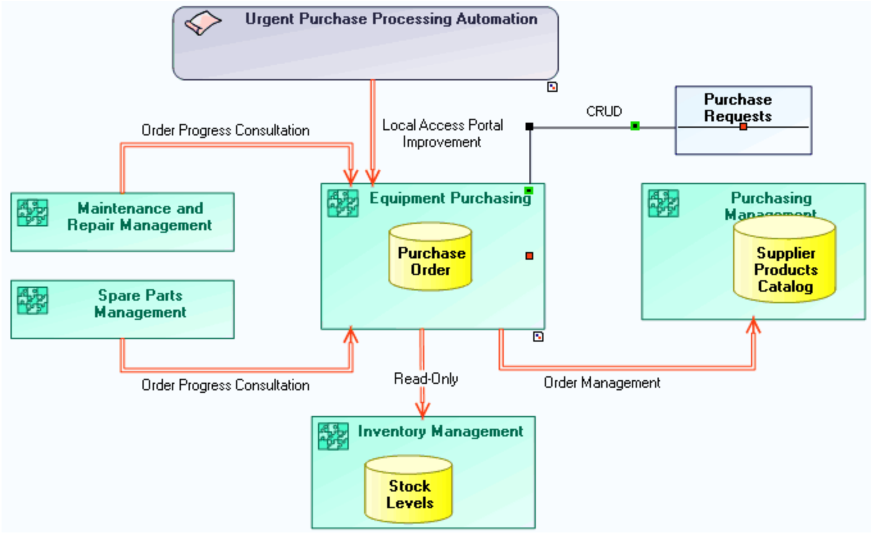
1. Right-click the project, for example "Urgent Purchase Processing Automation", to open its pop-up menu.
2. Select **Project Impact Diagram**.



The diagram opens in a new window.

If the process does not have a diagram, you can create it. For more details, see *"Building an Impact Diagram", page 53**.*

A project impact diagram can be presented as below.



The "Equipment Purchasing" application is the main application impacted by the project, in particular for improvement of its access portal.

The "Equipment Purchasing" application used the "Inventory Management" application to consult and update information on spare parts stock levels. It also uses the "Purchasing Management" application with the "Catalog" database to send a purchase order.

The "Maintenance and Repair Management" and the "Spare Parts Management" applications will also use the "Equipment Purchasing" application portal.

Finally, the "Purchase Request" entity is accessed on creation, read-only, read/write and deletion by this application.

Building an Impact Diagram

Creating an impact diagram

To access an impact impact diagram of a project:

1. Right-click the project to open its pop-up menu.
2. Select **New > Diagram**.
   The diagram creation dialog box appears.
3. Select **Project Impact Diagram** and confirm that the **Diagram Initialization** check box is selected before clicking **Create**.
   The impact diagram opens. The current project is positioned in the diagram.

Adding an application in an impact diagram

The main components impacted by the project are placed in the diagram. These can be applications, services, databases or actors.

To add an application in a project impact diagram

1. Select the **Application** button and click in the diagram.
   The **add application** dialog box appears:
2. Click the arrow at the right of the **Name** box and select **Propose** in the drop-down list.
   The list of applications accessible from the current library appears.
3. Select the required application.
4. Click **OK**.
   The application appears in the diagram.

Defining project impact on an application

To specify that a project impacts an application:

1. Click the **System Impact** button .
2. Click the project to position the impact and, holding the left mouse button down, draw a link to the application.
3. Release the mouse button.
   The link representing project impact on the application appears in the diagram.

To associate a summary text describing nature of the impact:

1. Right click the impact to open its pop-up menu and select **Properties**.
   The properties window opens.
2. Specify the name of the impact in the **Local Name** box.
3. Click **OK**.
   The name of the impact appears in the diagram.

☛ *In an impact diagram, you can create impact links from a project, application, service, actor, database or entity.*

☛ *A database or entity can only have an impact on a project, application or service.*

You can similarly add databases or entities in your diagram.

☛ *To show the corresponding buttons, click the "Views and Details" button in the diagram toolbar and select views "Entities" and "Databases".*

Specifying entity access mode

The entity is a stereotype enabling characterization of passive classes that do not themselves generate interactions. They can participate in in several use cases and generally outlive any single interaction. They represent objects shared between the different actors that handle them.

You can specify the type of access of the different impact diagram components to an *entity* (CRUD).

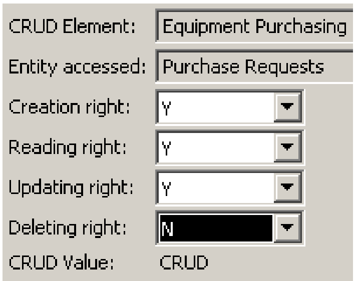
☛ *CRUD is an abbreviation of Create, Read, Update, Delete.*

To specify that an entity is used by an application:

1. Click the reel button  in the insert toolbar.
2. Click the entity to position the link and, holding the left mouse button down, draw a link to the application.
3. Release the mouse button.
   The link representing access to the entity by the application appears in the diagram.

To define the type of access of the application to the entity:

1. Right click the link to open its pop-up menu and select **Properties**.
   The properties window opens.
2. Select the **Characteristics** tab and enter access types by selecting "Y" or "N" in the drop-down menus. The **CRUD Value** box is completed from letters defining the access type.



3. Click **OK**.
   The CRUD value is displayed on the link in the diagram.

MODELING USE CASE SYSTEM PROCESSES

Operation of each use case can be described by a system process modeling sequence flow of screens presented to the user.

In detailed specification phase, the progress of tasks implemented in a service can also be modeled by a system process.

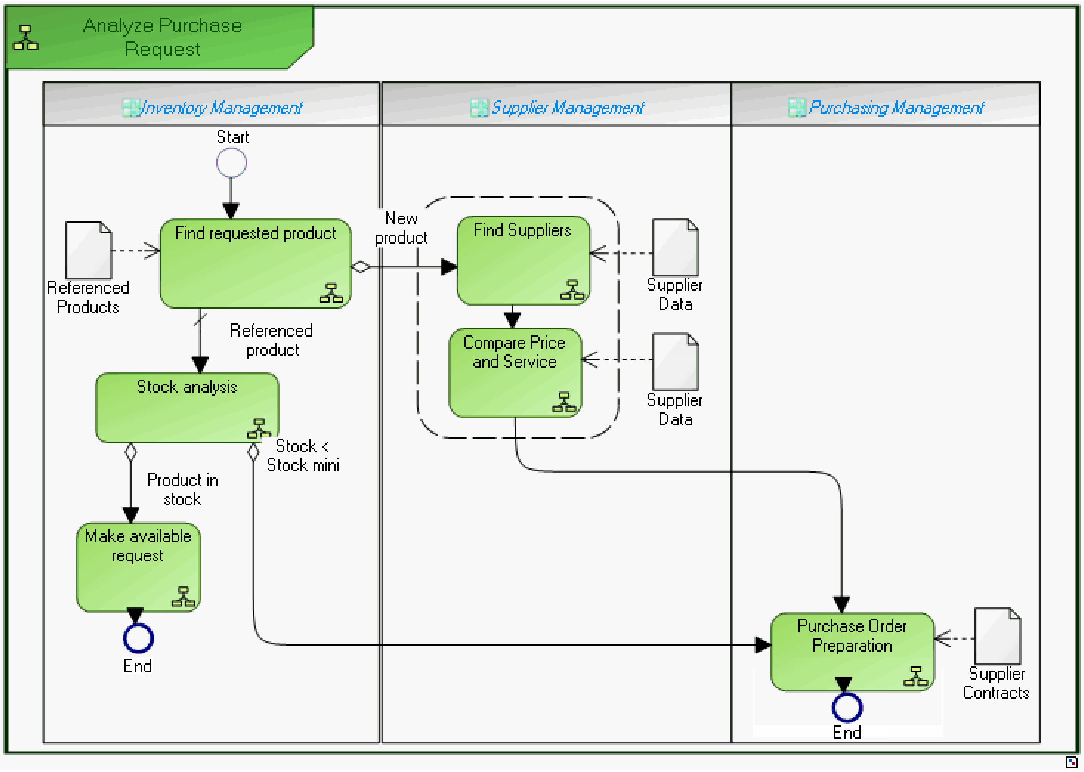
System processes are introduced in this chapter and presented in more detail in the **HOPEX Business Process Analysis** guide.

Drawing a System Process Diagram

System process diagram example

This presentation is based on the example of purchase requests represented below.

- A product search is carried out from the referenced products repository.
- If the product is referenced, stock is analyzed.
- If stock is sufficient, a "Make available" request is activated and the process ends.
- If stock is less than minimum stock, a purchase order is sent to the supplier.
- If the product is new, search for a supplier and comparative study of prices and services is carried out. An order is then sent and the process ends.

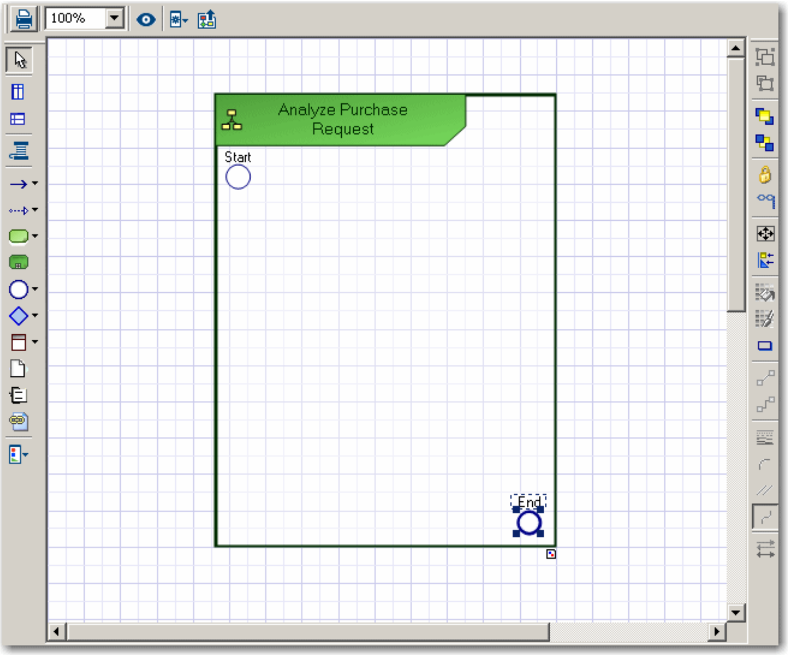


Creating a System Process Diagram

To create a system process diagram:

1. Right-click the business process name and select **New Diagram**.
2. Select the **System Process** diagram type.
3. Keep the option **Diagram initialization** selected (by default) to initialize the diagram.

   ☛ *Diagram initialization automatically positions the frame of the described process and the main events.*
4. Click **Create**.
   The diagram is initialized. The current process, represented by a frame, is placed in the diagram.



Creating Objects Within the System Process

In this section you will create the objects to be placed in your diagram: participants in the process and tasks executed with their sequence flows and message flows exchanged.

Creating participants

You will define the participants that carry out the process steps.

☛ *Column presentation simplifies diagram reading, though other presentations are possible.*

To create an actor participating in the process:

1. In the diagram objects toolbar of the system process, click on the arrow to the right of the **Participant** button and, in the scroll-down list, select **Participant (Application)**.
2. Click in the diagram workspace within the system process frame.
   The **Creation of Participant (Application)** dialog box opens.
3. To assign an application to this participant, enter the application name in the **Application** field.

   ☛ *If you are working on an empty library, you must create an application.*
4. Click **Finish**.
   The participant is positioned in the diagram; by default it has the name of the assigned application.

To position the participant in the frame of the described process:

- Click the title bar of the participant and, while holding down the mouse button, move the object.
  You can, for example, position it at top left of the organizational process frame.

To enlarge frames of participants so that their height is the same as that of the process:

1. Select the participants.
2. Click the point at the center of the bottom line of the frame, and holding the mouse button down, drag the frame line downwards.

Creating Tasks

A task is a step in a process.

You will create tasks and connect each of these to the application responsible for its execution.

To create a task:

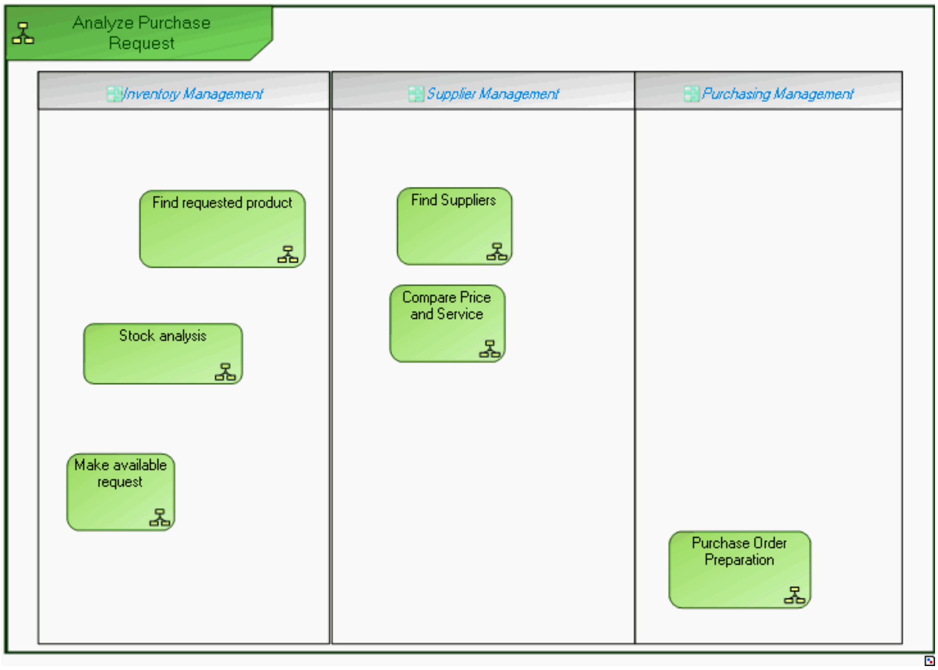
1. In the diagram insert toolbar, click the **Task** button
2. Click in the diagram within the shape of the participant concerned.
   The task appears in the diagram.

To assign a task to another participant:

1. Move the task from one participant to another.
   The frame of the destination participant is highlighted.

   ☛ *When placed, the task is disconnected from the source participant and connected to the target participant, which means that the target participant will execute the task.*

Example of participants and tasks



Creating events

To create the process start event:

1. In the diagram insert toolbar, click the **Event** button .
2. Click in the frame of the described process.
   The **Creation of Event (GRC)** dialog box opens.
3. Select the "Start" event nature, then click **Next**.
4. In the list of types offered, select "None" then click **Next**.
5. Click **OK** (Web Front-End) or **Finish**(Windows Front-End).
   An event named "Start" appears in the diagram.

Proceed in the same way to create the end events.

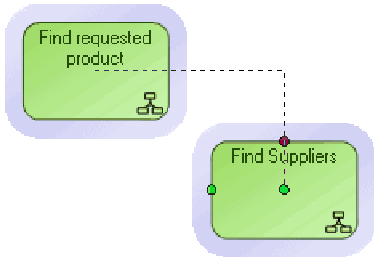
Describing task sequence flow

The sequence flows specify the order of task execution.

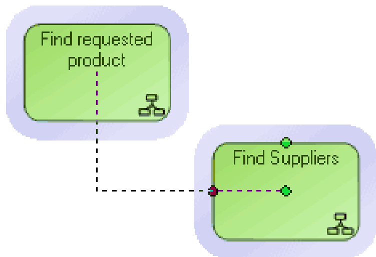
To create a sequence flow between two tasks:

1. In the diagram insert toolbar, click the **Sequence Flow** button .

2. Click the sender task and draw a link to the destination task.
The two tasks are highlighted and a dotted line indicates the path that will be taken by the graphic link.



☛ *Several paths are possible, you must simply move the cursor to within the frame of the destination task.*



Similarly create all sequence flows connecting tasks of the process, as well as the flows between events and tasks.

Defining a Condition on a Sequence Flow

```
Example: Depending on whether it is a new product or a
product already referenced, the "Find Requested Product"
task can result in either a search for a supplier, or a
stock analysis. This alternative requires the creation of a
condition.
```

To define a condition on the sequence flow between the "Find Requested Product" task and the "Find Suppliers" task:

1. Right-click the sequence flow and select **Properties**.
2. In the dialog box that opens, select the **Characteristics** tab.
3. Click the arrow at the right of the **Sequence Flow Type** field and select "Conditioned".
4. In the **Predicate** field, specify the conditioning expression, for example "New Product".
5. Click **OK**.
The text associated with the condition appears on the condition which then takes form .

To specify that the sequence flow between "Find Requested Product" and "Stock Analysis" is used by default:

1. Right-click the sequence flow then select **Properties**.
2. In the dialog box that opens, select the **Characteristics** tab.
3. Click the arrow at the right of the **Sequence Type** box and select "Default" in the drop-down list.
4. Click **OK**.
   The sequence flow then takes form .

Creating data objects

You can add data objects to the diagram. A data object is used to explain how documents, data, and other objects are used and updated during the system process.

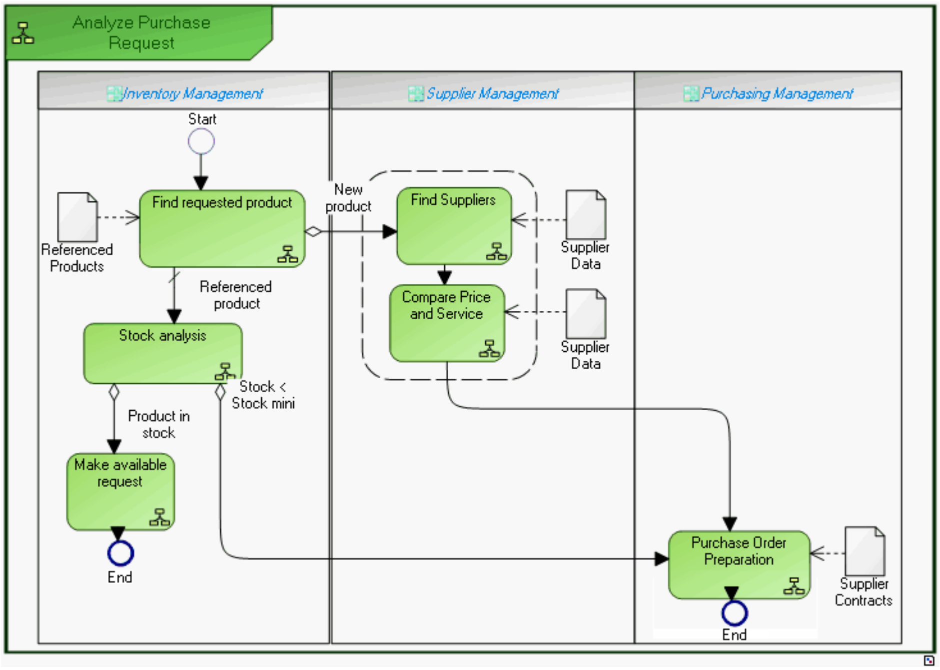
Data objects are created using the **Data Object** button in the diagram objects toolbar.

Once created, you must connect them to the tasks that use them.

To connect a data object to a task:

1. Click the **Link** button in the diagram objects toolbar.
2. Click the data object and, holding the mouse button down, draw a link to the destination task.
3. Release the mouse button.

Shared Artifacts Example



Saving a diagram

Save your drawing using the **Save** button.

You can now print your diagram by selecting **File > Print**.

Improving the Presentation

Various functionalities allow you to improve presentation of objects in diagrams: annotation of objects, modification of shapes and presentation of names, etc. For more details, see chapter "Handling Objects in Diagrams" in the **HOPEX Common Features**. guide.

Modeling Tasks of a System Process

The functional analysis phase of the project is based on use cases and on the data model, obtained in preliminary study phase, to represent the functional architecture of the future system.

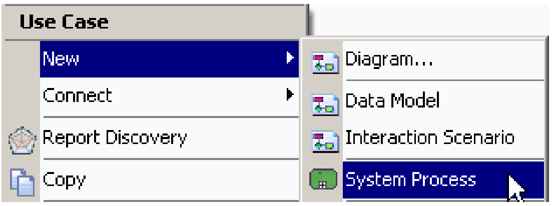
The functional analysis phase describes the system processes implemented in the different use cases of an application or service.

A system process diagram specifies the sequence flow of tasks to be executed so that the user can check that the application satisfies its requirement.

Describing a Use Case by a System Process

To create a system process from a use case:

1. Right-click the use case to open its pop-up menu.
2. Select **New > System Process**.



The diagram of a new system process opens. The system process associated with the use case is already positioned in the diagram.

To change the name of the system process created:

1. Click the name of the process.
2. Press key <F2>
3. Enter the new name of the process.

☛ *For more details on creating system process diagrams, see the **HOPEX Business Process Analysis** guide, paragraph "Managing System Processes".*

Functional Modeling Example

The *system processes* used for a project functional analysis are stored in a package.

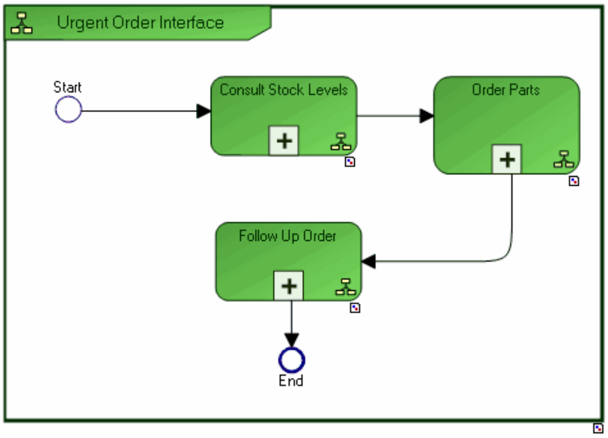
In the example of the purchase request processing automation project, system processes are stored in the "Urgent Purchase Requests" package .

A system process is the executable representation of a process. the events of the workflow, the tasks to be carried out during the *processing, the algorithmic elements used to specify the way in which the tasks follow each other, the information flows exchanged with the participants.*

Access the system process diagram from a package

To access the system process diagram from a package:

1. In the **Home** navigation window, expand the tree associated with the package, for example the "Urgent Purchase Requests" package.
   The list of system processes stored in the package appears.
2. Right-click the system process, for example "Urgent Order Interface" to open its pop-up menu, and select **System Process Diagram**.
   The diagram opens.



The main steps of this system process are:

consult local stock levels so that a spare part can be made available

order parts when stock level has reached critical threshold or if a part is unreferenced

follow-up an order

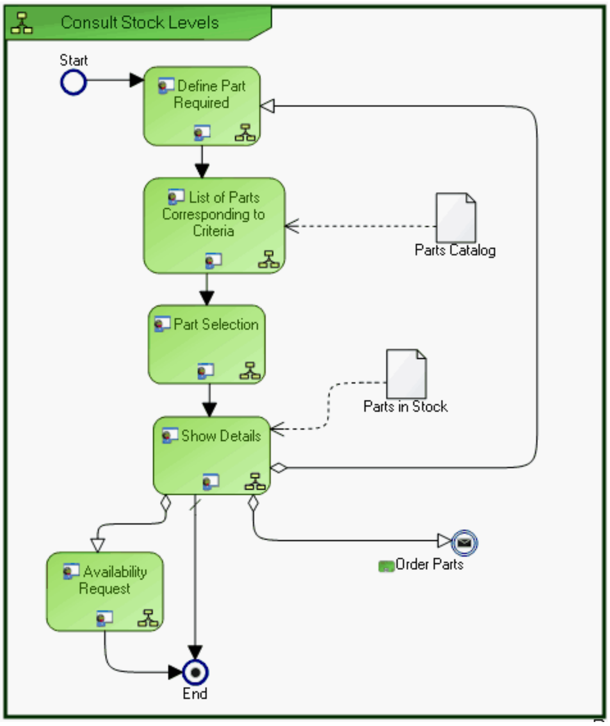
☛ *You can also access a system process from a use case. For more details, see* *"Describing a Use Case by a System Process", page 65.*

Display the diagram describing a step in the system process in detail:

To open the diagram describing a step in the system process in detail:

1. Right-click the system process, for example "Consult Stock Levels" to open its pop-up menu.

2. Select **System Process Diagram**.
   The diagram associated with the process opens.



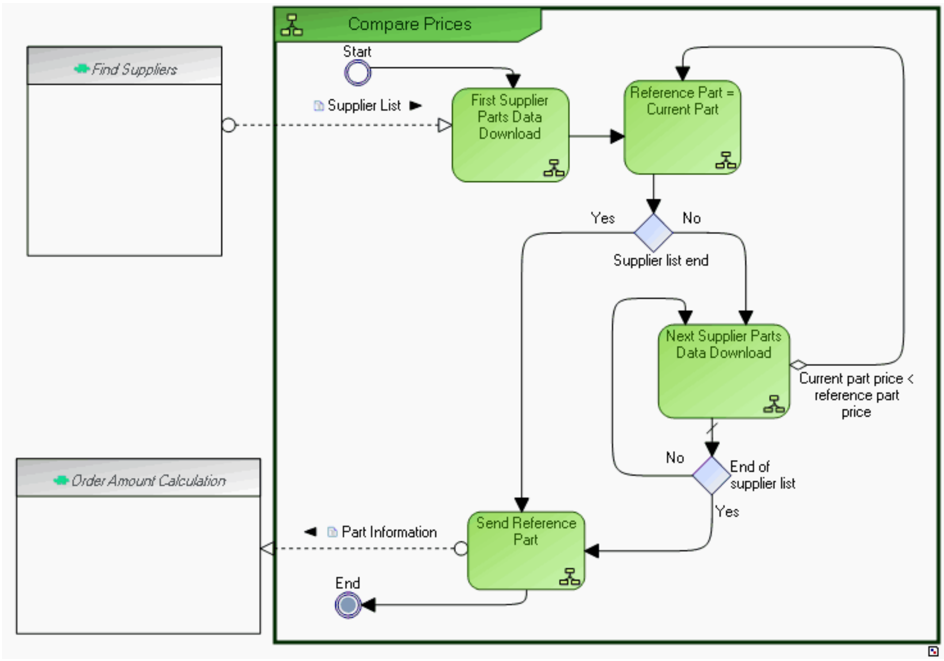
Consulting stock levels begins by display of a screen enabling identification of the required part. The list of parts found in the catalog is presented in the next screen.

When the user has selected the required part, information on details is displayed. From this screen, it is possible to obtain information on another part, make an availability request for the part, or indeed order the part.

Modeling Tasks of a Service

The phase of detailed analysis of system components impacted by the project consists of detailed modeling of the operation of IT services.

In the context of the urgent order request processing automation example, the service for comparing prices is represented by a system process.



This diagram describes the algorithm of the "Compare Prices" service, which should return the reference of the lowest-priced part.

The list of suppliers of the required part is given at input. The part proposed by the first supplier in this list becomes the reference part. Assuming the supplier list is not empty, data concerning the required part is then analyzed. If the price of the current part is lower than the price of the reference part, the reference part becomes the current part.

When the complete list of suppliers has been analyzed, information concerning the reference part is sent to the "Order Amount Calculation" service.

6

DESCRIBING SEQUENCES OF APPLICATIONS AND SERVICES

For each use case, you can create sequence diagrams describing the messages exchanged between elements of the system in a particular context. Depending on which UML diagrams you use (UML 1.3 or 2.3), you will use the interaction scenario diagram or the sequence diagram to describe this operation. Interaction diagrams focus on cooperation of a set of objects.

The execution of services can be described by a batch planning structure diagram. This diagram is used to represent the execution schedule for batches, batch programs and batch organization.

The points covered here are:

Sequence Diagram

The sequence diagram replaces the interaction scenario diagram in the context of UML 2.3 compatibility.

The sequence diagram presents the same exchanges between system elements, highlighting their chronology. These elements are represented in the diagram by their lifelines.

Thanks to new features offered by UML2, it is now possible to break sequence diagrams down into sequence sub-diagrams and use control structures within sequence diagrams.

Creating a Sequence Diagram

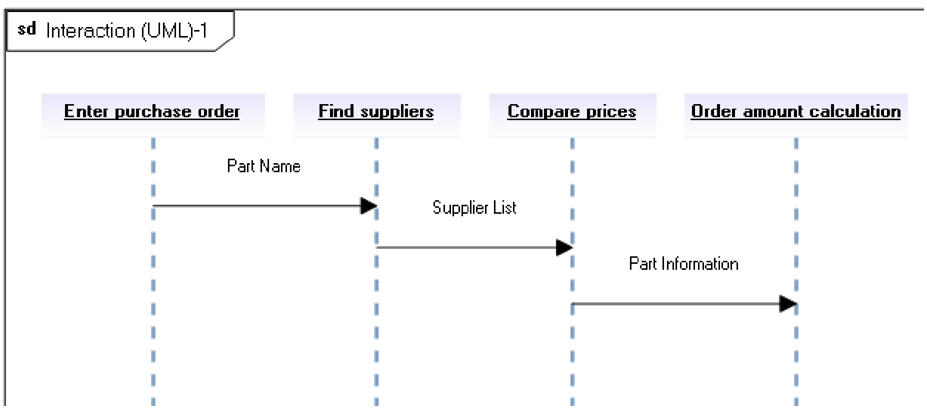
To create a sequence diagram of a use case:

1. Right-click the use case and select **New** > **Interaction (UML)**.
   The diagram creation dialog box opens: sequence diagram (SBP) is selected.
2. Click **Create**.
   The diagram opens.

Sequence Diagram Objects

The sequence diagram contains:

- Lifelines which define interaction participants: instances of applications, services or interfaces.
- Different types of messages exchanged between participants.
- Advanced functions that enable concise description of several execution sequences.



For more information on sequence diagrams, see the "**HOPEX UML**" guide.

Interaction Scenario Diagram

If you work with version UML 2.3 of diagrams, interaction diagrams are replaced by sequence diagrams. For more details, see "Sequence Diagram", page 70.

You can specify use cases of an application or service by means of interaction scenarios. An interaction scenario details message instances exchanged between application, *service* or *User interface* instances in an application architecture.

An IT service is a component of an application made available to the end user of the application in the context of his/her work.

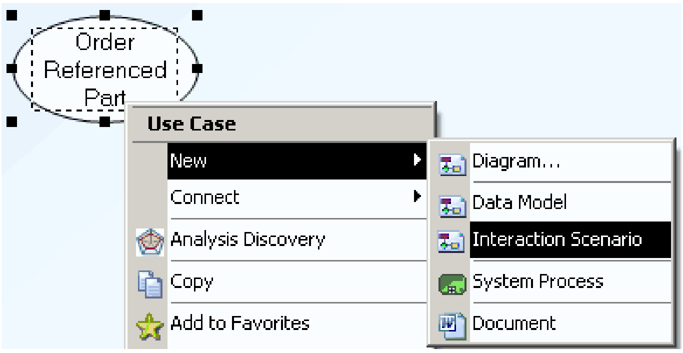
A Human-Machine Interface enables definition of a screen of an application or an IT service.

Creating an Interaction Scenario Diagram

An interaction scenario diagram is created from a use case. It can describe an application or service by presenting a succession of message instances and their interaction with application or service instances.

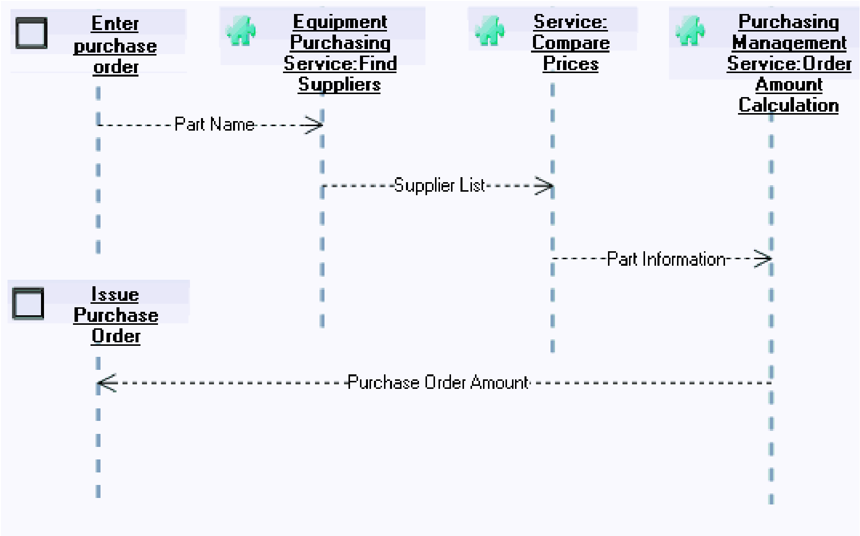
To create an interaction scenario diagram of a use case:

1. In the use case diagram, right-click the use case you wish to detail.
2. In the pop-up menu select **New > Interaction Scenario**.



3. In the dialog box that appears, select the "Interaction Scenario Diagram" diagram type.
4. Click **Create**.

Interaction Scenario Diagram Example



This diagram describes the operation of the "Order Unreferenced Parts" use case :

- When a purchase request is entered in the user interface, the name of the part is received by the "Find Suppliers" service, which draws up the list of suppliers offering the requested part.
- The "Compare Prices" service looks for the lowest-priced product and sends information to the "Order Amount Calculation" service.
- When the order amount has been established, a final "Issue Purchase Order" service sends the order via the interface.

Creating an Interaction Scenario Diagram

The interaction scenario diagram contains:

- Instances of applications, services or interfaces
- Instances of messages

Instances of applications, services or interfaces

Depending on whether the diagram describes a user interface, an application or a service, the interaction scenario diagram describes messages exchanged between application instances, *service* instances and *user interface* instances.

A Human-Machine Interface enables definition of a screen of an application or an IT service.

An IT service is a component of an application made available to the end user of the application in the context of his/her work.

To create a service instance for example:

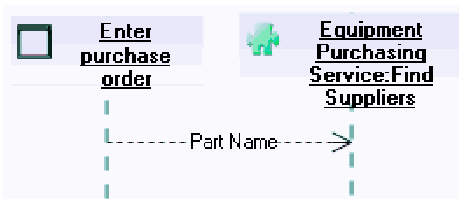
1. Click the "Service" button in the diagram toolbar:
2. Click on the diagram.
   The **Creation of Service** dialog box appears.
3. Click the arrow at the right of the **Represented Service** box and select **Propose IT Service** in the drop-down list.
   The list of services accessible from the current library appears.
4. Select the IT service you require.
5. Click **OK**.
   The service instance appears in the diagram.

Message instance

Message instances define the data exchanged between application, service or user interface instances. The sequence described in an interaction scenario diagram indicates message sending order.

Message instances displayed in an interaction scenario diagram correspond with messages owned by the application that have been previously defined in another diagram.

The message instance sent by "Enter Purchase Order" to "Find Suppliers Service" carries the "Part Name" message instance.



To create a message instance:

1. Click the "Message Instance" button in the toolbar:
2. Click the dotted line under the first object and, holding the mouse button down, draw a line to the second object.
3. Release the mouse button.

The message instance exchanged between the two objects is drawn.

Batch Processing

HOPEX System Blueprint is used to describe, in the **batch planning structure diagrams**, the sequencing of automated processing.

Describing batch processing

A batch processing is a set of IT processing operations executed by a computer without human intervention, generally overnight or at the weekend.

A **batch process** is defined by a **batch plan** or by a **program**.

A batch planning defines all the IT processing operations to be executed on one or several machines over a given time period.

A program is an elementary stage in execution of a batch processing that consists of running execution of a program using the appropriate parameters.

A **batch plan** is a set of **batch processes**. Each is associated either with a **program** or with another **batch plan**. A **batch plan** can be described by a **batch planning structure diagram**.

☛ *For more details, see* *"Building a batch planning structure diagram", page 75*.

A **program** is a set of **batch processes**. Each of these can be associated with a single **program**. A **program** is described by a **batch program structure diagram**.

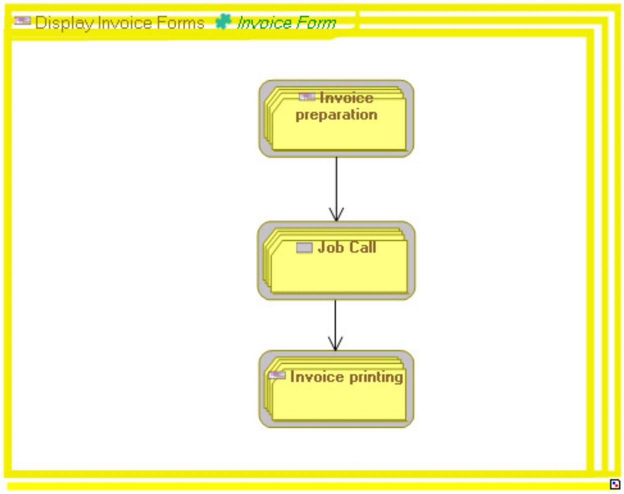
☛ *For more details, see* *"Creating a batch program structure diagram", page 77*.

Building a batch planning structure diagram

A batch planning defines all the IT processing operations to be executed on one or several machines over a given time period.

Creating a batch planning structure diagram

The sequencing of automated processes can be described in a **batch planning structure diagram**.



To create a batch planning structure diagram:

1. Right-click the batch plan and select **New** > **Diagram**.
   The diagram creation dialog box opens.
2. Creating **Batch Planning Structure Diagram**. Check that you have selected the **Initialize the diagram** option before clicking **Create**.
   The diagram opens.

Adding a call for batch processing in the diagram

The components of a **batch plan** are defined with **batch processing calls** that are positioned in the diagram. This can be applied to batch plans or programs.

To add an operating type component to the string structure diagram for batch process:

1. Select the **Batch Processing Call** button and click in the diagram.
   The **Add a Batch Processing Call** dialog box opens.
2. Click the arrow at the right of the **Batch Process Called** box and select **Connect** in the drop-down list.
   A connection dialog box opens.
3. Select **Batch Planning** in the left part of the window and click the search arrow.
   The list of batch plans appears.
4. Select the report that interests you and click **Connect**.
5. Click **OK**.
   The call for batch processing appears in the diagram with the batch planning icon.

Defining batch sequencing

To specify the execution order of processes:

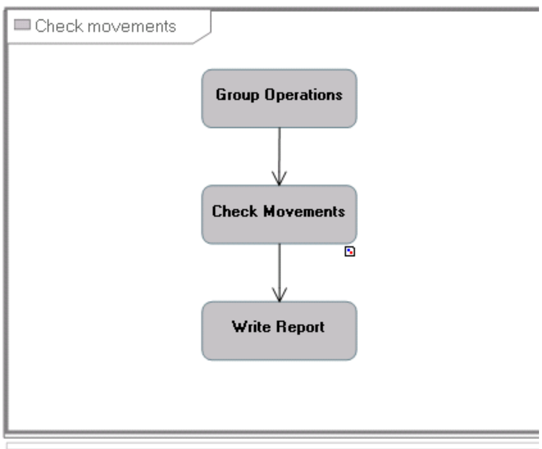
1. Click **Batch Sequence**.
2. Click the initial batch processing call and, holding the left mouse button down, draw a link to the batch processing call.
3. Release the mouse button.
   The link representing the sequencing of the processes appears in the diagram.

Creating a batch program structure diagram

A program is an elementary stage in execution of a batch processing that consists of running execution of a program using the appropriate parameters.

Creating a batch program structure diagram

The sequencing of the processes of a program can be described in a **batch program structure diagram**.



To create the batch program structure diagram:

1. Right-click the program in question and select **New Diagram**.
   The diagram creation dialog box opens.
2. Select **batch program structure diagram**. Check that you have selected the **Initialize the diagram** option before clicking **Create**.
   The diagram opens.

Adding a programming call to the diagram

The components of an **program** are defined with **programming calls** that are positioned in the diagram.

To add a component to a diagram:

1. Select **Programming Call** and click in the diagram.
   The **Add a Programming Call** dialog box opens.
2. Click the arrow at the right of the **Batch Process Called** box and select **Connect** in the drop-down list.
   A connection dialog box opens.
3. Select **Program** in the left part of the window and click the search arrow.
   The list of batch plans appears.
4. Select the program in question and click **Connect**.
5. Click **OK**.
   The program call appears in the diagram.

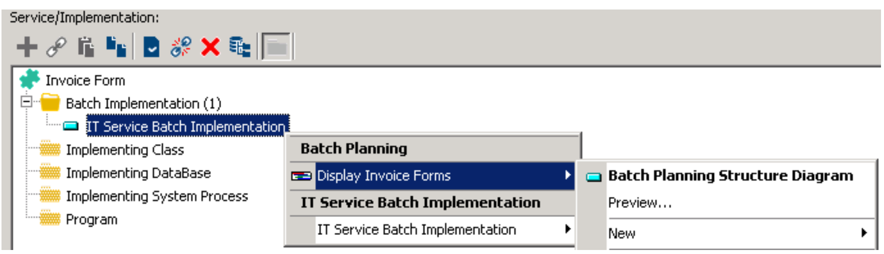
The of program execution is defined by links. For more details, see "Defining batch sequencing", page 77.

Describing IT Service Implementation

The execution of application services can be described by a batch planning structure diagram. For more details, see "Creating a batch planning structure diagram", page 76.

To describe that an application service is implemented by a batch plan, for example:

1. Open the properties window of the application service in question.
2. Select the **Implementation** tab.
3. In the **Service/Implementation** section, right-click on the **Implementation per Batch Process** folder and select **New**.
   The window for creating an application service implemented by a batch process appears.
4. Click the arrow at the right of the **Implementing Batch Process** box and select **Connect** in the drop-down list.
   A connection dialog box opens.
5. Select **Batch Planning** in the left part of the window and click the search arrow.
   The list of batch plans appears.
6. Select the report that interests you and click **Connect**.
7. Click **OK**.
   The application service implemented by a batch process appears in the list. It is connected to the batch plan selected.



Using realizations

A realization mechanism is provided to associate an application process to an **organization schedule** or via a **Program**.

A realization describes the relationship between a logical entity and a physical entity that implements it. The physical entity gives the list of logical entities that it implements.

To describe that an batch plan is associated with an application process:

1. Open the properties pane for the batch plan in question.
2. Select the **Characteristics** tab and the **Realization** subtab.
3. In the **Composite Realization** section, click **New**.
   The window for creating a system process batch realization appears.
4. Opposite the **Base Type** field, select **Connect**.
   The query dialog box appears.
5. Select the application process that interests you and click **Connect**.
6. Click **OK.**
   The system process batch realization appears in the properties page of the batch plan.

Describing User Interfaces of Services

It is possible to describe interfaces connecting services or operations with the exterior. This description is carried out in a user interface diagram.

Creating a User Interface Diagram

To create an interface diagram of a service:

1. Right-click the service in question.
2. Select **New** > **User Interface**.
   The MMI diagram opens in the Edit window.

Opening a User Interface Diagram

You can open an interface diagram:

- From the pop-up menu of the described service or operation.
- From the **HOPEX** workspace **Diagrams** window.

Interface example

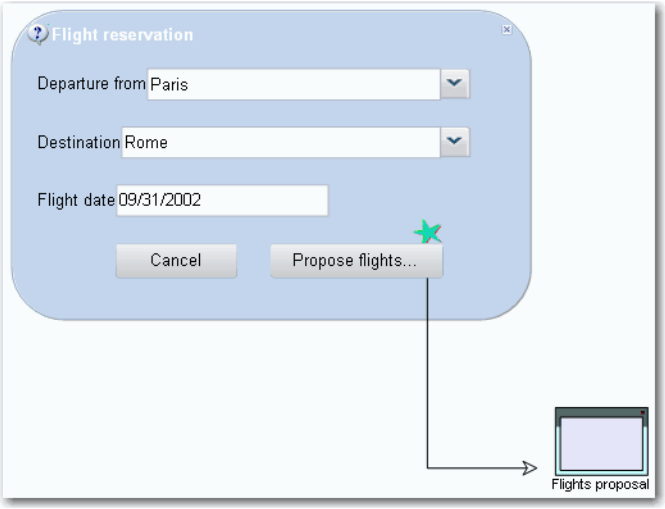
Take, for example, the "Flight Reservation" service.

To access its user interface diagram:

1. From the **Home** navigation window, expand the folder "Voyage and Vacations", then "Travel and Vacation Business" and "Reservation".
2. Expand the "Application" folder and the "Online Reservation V&V" application.

3. Right-click the "Flight reservation" service and select **User Interface**.

The MMI diagram opens in the Edit window.



The interface is presented in the form of a dialog box, in which various fields must be completed.

- Departure city
- Destination
- Flight date

A button cancels the request, another button opens a second interface.

Drawing the Interface Diagram

The interface diagram allows you to draw the interface of the operation or service.

User interface element

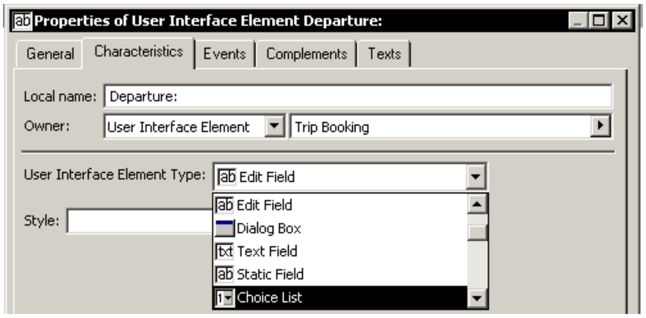
Buttons allow you to modify the appearance of the interface:

- Text Field
- List
- Radio Button
- Check Box
- etc.

To create an element:

1. In the diagram objects bar, select the button corresponding to the element required, then click in the diagram.
2. In the dialog box that appears, enter the name of the element.
3. Click **OK**.

You can also click the **User interface element**  button and indicate the element type in its properties dialog box.



User interface event

You can connect an *event* to a user interface element. In our example, the "Propose flights" button is connected to an event, which when actuated opens another interface.

To create an event:

1. Click the **Interface event**  button, then click in the diagram.
2. Enter the name of the event and click **OK**.

Event type

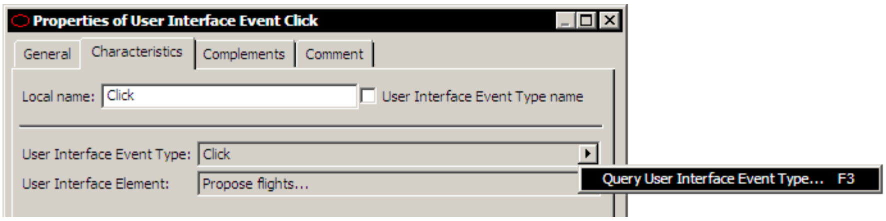
There are various types of event. An event can be:

- Click on a button
- Entry in a text field
- etc.

To specify the type of event:

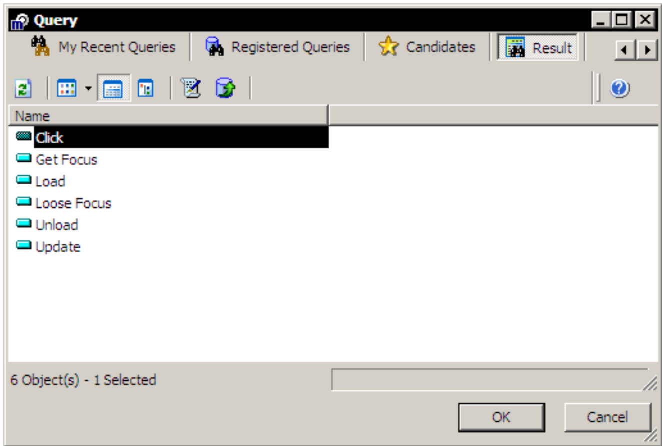
1. Open the properties dialog box of the event.
2. Click the **Characteristics** tab.

3. In the **User Interface Event Type** text box, click the arrow and select **Query User Interface Event Type**.



The selection dialog box appears.

4. Click **Find**.
The list of event types appears.



5. Select the type required and click **OK**.

Connecting the event to an element

To connect the event to an element, there are two possibilities:

- Select the event in the diagram and drag it onto the element.
- Or open the properties dialog box of the event and complete the **User interface element** text box.

ANALYZING PROJECT DATA

HOPEX System Blueprint provides different report templates used to analyze each design phase of an IT system.

Reports are generated automatically from your repository data, and their purpose is to provide a clearer view of information according to the aspect under study (project requirements, functional analysis, project impacts, etc.).

The points covered here are:

HOPEX System Blueprint Reports

A report template is a report model. **HOPEX System Blueprint**provides by default different report templates enabling analysis of elements contained in a project.

Predefined Report Template Operation

Two main factors are necessary for correct use of **HOPEX System Blueprint** analysis reports: project scope and project deliverables state.

Project scope

Objects displayed in **HOPEX System Blueprint** analysis reports are objects within the scope of the project, in other words contained in the project packages. These packages group the different project elements: requirements, actors, use cases, etc. They can be created automatically at initialization of a project (see "Initializing the project HOPEX System Blueprint", page 27).

Each package generated automatically is stereotyped according to the objects it contains:

Package default name	Package object type	Package stereotype
Project name - Actors	Org-Units	Actors profiles group
Project name - Use cases	Use case	Use case group
Project name - Persistent data	Data models	Data model group
Project name - System dynamics	System process	System process group
Project name - Requirements	Requirement	Requirement group
Project name - User interfaces	MMI	User interface group

You can then create new packages from the project, attaching these to one of the required stereotypes.

A stereotype is a type of modeling element that extends the semantics of the metamodel. Stereotypes must be based on existing types or classes whose structure they use. Other stereotypes can be created by the user.

To define the stereotype of a package:

1. Open the properties dialog box of the package.
2. Click the **Characteristics** tab.
3. In the **Stereotype** box, select the required stereotype.

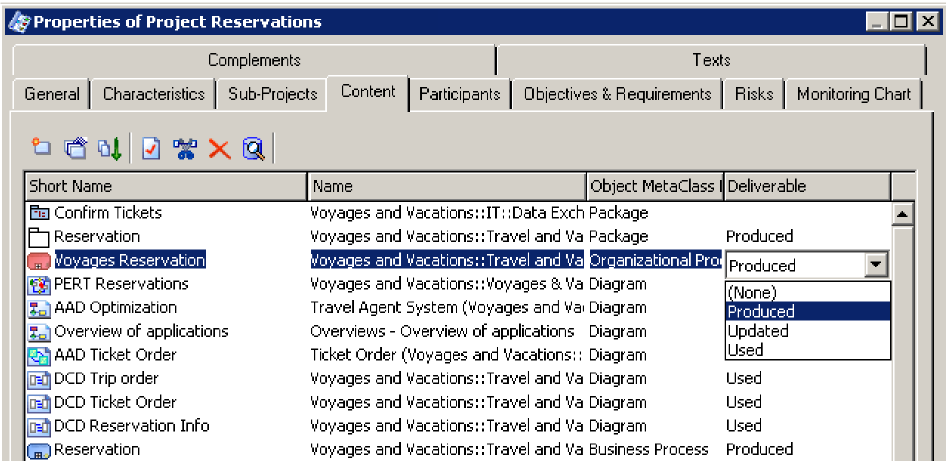
State of deliverables

Certain reports require definition of the state of deliverables contained in the project, corresponding to the state of progress of your IT project.

A deliverable is information on an element contained in the IT project, which can be: "produced", "updated" or "used".

To indicate if an element is produced, updated or used in the project:

1. Open the project properties dialog box.
2. Select the **Content** tab.
3. Select the element name.
4. In the **Deliverable** column, select one of the options.



HOPEX System Blueprint Report Templates

Here is the list of report templates useful within the framework of designing an IT project with **HOPEX System Blueprint**.

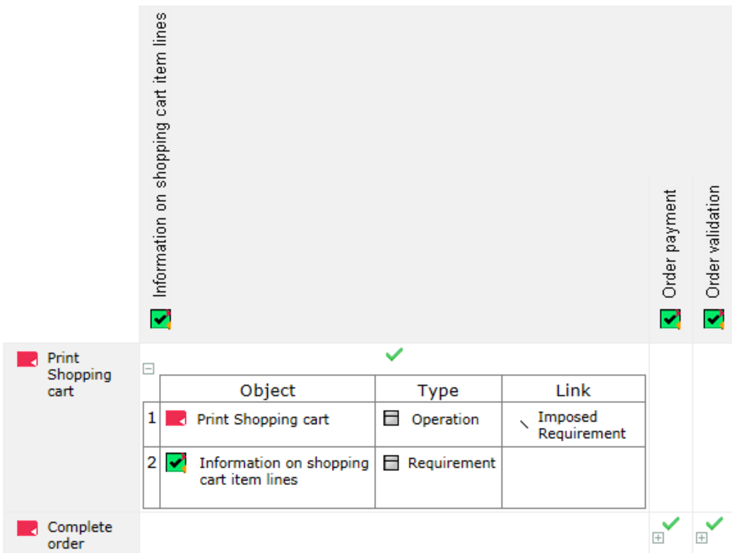
System Blueprint - 1 Requirement Analysis

This report template describes the project, its objectives and requirements as well as the list of business, organizational and functional processes contained in the project and linked to the requirements.

☛ *For more details on the descriptions of a project's objectives and requirements, see:* *"Establishing Objectives and Requirements", page 30.*

The generated report presents:

- the project objective and requirement diagram
- the diagram of processes owned by the project.
- for each process owned by the project, the detail of requirements linked to these components.



System Blueprint - 2 Functional Analysis

This report template concerns the functional aspect of the project; it describes the functionalities of applications by highlighting application and service behaviors. It focuses on modeling a functional viewpoint of your IT project: use cases, actors and system processes.

To be visible in the report, modeled elements must be within the scope of functional analysis; they must therefore be attached to packages of stereotype "Use case group", "Data model group", "User interface group" or "Actors profile group".

When information exists, the generated report presents:

- actors of the system (system actors or users). These are actors contained in the "*Project name* - Actors" package, or any other package of "Actors profile group" stereotype attached to the project.
- use case diagrams supplemented by description of each use case. These are use cases contained in the "*Project name* - Use cases" package, or any other package of "Use case group" stereotype attached to the project.
- dynamics of user interfaces represented by system process diagrams contained in the "*Project name* - User interfaces" package, or any other package of "User interface group" stereotype attached to the project.
- business data models contained in a package of "Data Model" stereotype, under the package of project use cases.
- state machine diagrams contained in a package of "Data model" stereotype, under the package of project data models.

System Blueprint - 3 System Analysis

This report template focuses more specifically on impacts of the system under study.

The generated report presents:

- all impact diagrams contained in project packages, with detail of elements affected.
- interaction scenario diagrams and sequence diagrams contained in project use case packages. Use cases must be "used" by the project (see “State of deliverables", page 87).
- detail of applications and services "used" in the project.
- data models "used" in the project.
- matrix of requirement traceability.

System Blueprint - 4 System Design

This report template enables description of the internal structure of a project application, the IT services and the databases of this application.

The description of services concerns services "produced" or "updated" in the project (see “State of deliverables", page 87). These are services that belong to the internal architecture of the analyzed application, and that are linked to the application by the “defined-service” link.

Each report relates to a specific application. Using this report template you can generate as many reports as there are applications.

A "System Blueprint - System Design" report template presents:

- the application tree of the described application.
- internal architecture diagrams of the application.
- data models describing databases of the application.
- system processes of services.
- interaction scenario diagrams and sequence diagrams of use cases implementing services of the application (see below "Specifying services implemented in a use case").
- matrix of requirement traceability.

Specifying services implemented in a use case

To define services implemented in a use case:

1. Open the properties dialog box of the use case.
2. Select the **Implementation** tab.
3. Select the **IT Service** folder and click the **New** or **Connect** button, depending on whether it is a new or existing service.