

|  |
| --- |
| **D5.1.2Process Management Tool Use Case Definition**  Galaxy use case definition |

logo-galaxy_barre_points

|  |  |  |  |
| --- | --- | --- | --- |
|  | NamE | partner | Date |
| Written by | L. BARESSE | AKKA | 23/04/2012 |
|  | IRIT |  |
|  |  |  |
| Reviewed by |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Record of Revisions

| Issue | Date | Effect on | | Reasons For Revision |
| --- | --- | --- | --- | --- |
| Page | Para |
| 01A | 23/04/2012 |  |  | Document creation |
|  |  |  |  |  |

Table of contents

[1. Introduction 6](#_Toc322614323)

[1.1 Goal of this document 6](#_Toc322614324)

[1.2 Document organization 6](#_Toc322614325)

[2. SCope 6](#_Toc322614326)

[2.1 Description of the Process Management Tool system 6](#_Toc322614327)

[2.1.1 Activity performed by project managers and quality engineers 7](#_Toc322614328)

[2.1.2 Activity performed by the galaxy users during process enactment 8](#_Toc322614329)

[2.2 Scope of the study 8](#_Toc322614330)

[3. Validation method 8](#_Toc322614331)

[4. Involved PARTNERS 9](#_Toc322614332)

[5. Validation ScenariOS 9](#_Toc322614333)

[5.1 Technical validation scenarios 9](#_Toc322614334)

[5.1.1 Create Generic Process Model 9](#_Toc322614335)

[5.1.2 Publish Generic Process Model into the GPM repository 9](#_Toc322614336)

[5.1.3 Retrieve Generic Process Model from the GPM repository 9](#_Toc322614337)

[5.1.4 Initialize Enactable Process Model on the Galaxy Server 10](#_Toc322614338)

[5.1.1 Open an Enactable Process Model from the local machine 10](#_Toc322614339)

[5.1.2 Receive the list of all available Enactable Process Model from the Galaxy Server 10](#_Toc322614340)

[5.1.3 Open an Enactable Process Model from the Galaxy Server 11](#_Toc322614341)

[5.1.4 Open several Enactable Process Models in parallel 11](#_Toc322614342)

[5.1.5 Request lock for an Enactable Process Model 11](#_Toc322614343)

[5.1.6 Work offline on a locked for an Enactable Process Model 11](#_Toc322614344)

[5.1.7 Request unlock for an Enactable Process Model 11](#_Toc322614345)

[5.1.8 Publish updated Enactable Process Model on the Galaxy Server 12](#_Toc322614346)

[5.1.9 Receive updated Enactable Process Model from the Galaxy Server 12](#_Toc322614347)

[5.1.10 Receive current actives tasks from the Galaxy Server 12](#_Toc322614348)

[5.1.11 Notify a event for a given task to the Galaxy Server 12](#_Toc322614349)

[5.2 Functional validation scenarios (IRIT) 12](#_Toc322614350)

[5.2.1 To be defined 13](#_Toc322614351)

[6. Involved models 13](#_Toc322614352)

[7. tools used 13](#_Toc322614353)

Table of APPLICABLE DOCUMENTS

| N° | title | Reference | Issue | Date | Source | |
| --- | --- | --- | --- | --- | --- | --- |
| Siglum | Name |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Table of ReferenceD DOCUMENTS

| N° | title | Reference | Issue |
| --- | --- | --- | --- |
|
|  | Galaxy glossary | D1.2.2 |  |
|  | Goal and metrics document | D1.1 |  |
|  | Architecture specification | D4.1 |  |
|  |  |  |  |

ACRONYMS AND DEFINITIONS

Except if explicitly stated otherwise the definition of all terms and acronyms provided in [R1] is applicable in this document. If any, additional and/or specific definitions applicable only in this document are listed in the two tables below.

Acronymes

|  |  |
| --- | --- |
| Acronym | DESCRIPTION |
| epm | Enactable process model |
| gpm | Generic process model |
| PM Tool | Process Management Tool |
|  |  |

Definitions

|  |  |
| --- | --- |
| TERMS | DESCRIPTION |
|  |  |
|  |  |
|  |  |
|  |  |

# Introduction

The goal of the Galaxy project is to work on the technical hard points related to the fragmentation and to the distributiveness of huge models, and to their synchronization in regards of the communication means classically used by development teams. Galaxy partners believe that a set of technical solution integrated in a common platform of service (called: “the Galaxy platform”) may greatly help in dealing with these hard points.

Based on a selected subset of candidate technologies, the architecture of the platform has been specified. As planned in the project proposal this platform is assessed using use cases where scalability issues can be identified and characterized.

## Goal of this document

A specific task of the Galaxy project (T5.1) is dedicated to the definition of the use cases. This document is a product of the task which describes the Process Management Tool study case.

## Document organization

The chapter 2 describes the context of the test case, including an overview of the domain and a focus on the scope on which of particular interest for the study.

Chapter 3 explains how the approach we have selected to verify and validate the added value of the services provided by the Galaxy platform.

Chapter 4 presents the partners involved on this study case and the way they have contributed to it.

Chapter 5specifiesthe scenarios which are used to assess the performance offered by the Galaxy platform.

Chapter 6 describes the models involved in validation scenarios that are played: viewpoints, views, sizes and organizations.

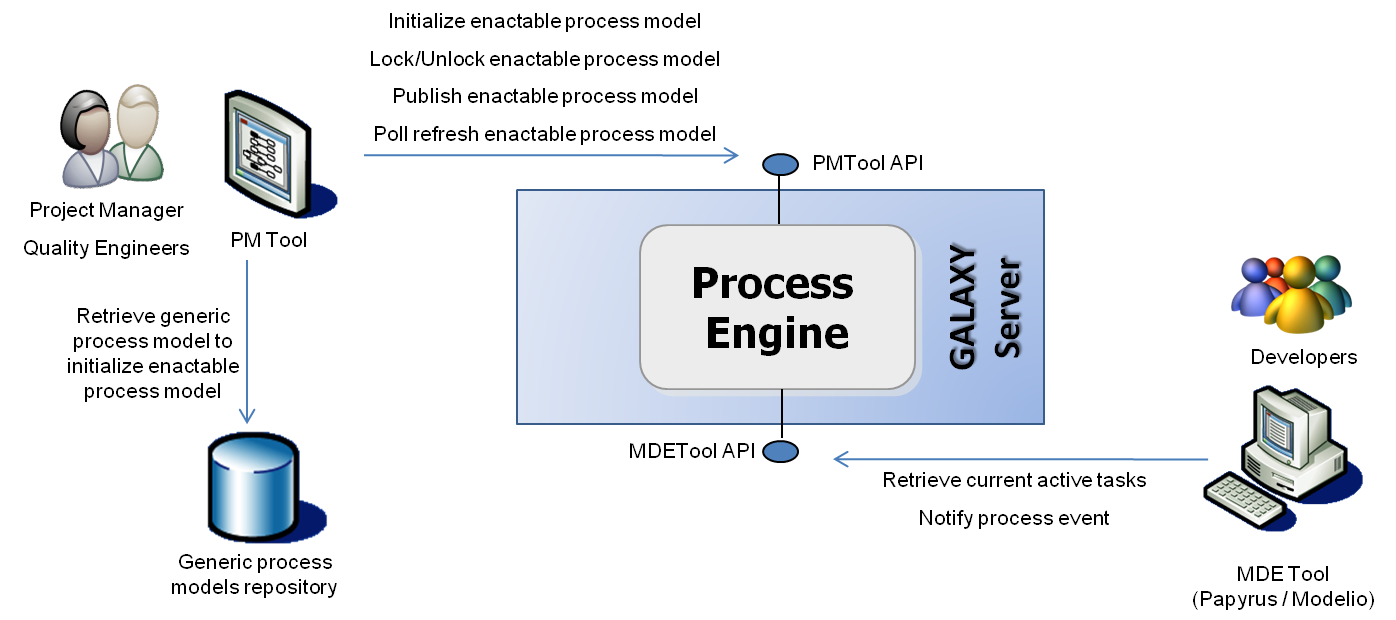
Last, Chapter 7is about the tools software tools used for the validation scenarios.

# SCope

## Description of the Process Management Tool system

Describe here the context of the domain of the study case and specific concepts of interest which helps in getting a better understanding of it.

Summarize the added values granted to the Galaxy project by the process management tool and the process enactment client tool. (IRIT)



This use case contains two general activities. The first activity involve project manager and quality engineers which use the PM Tool in order to initialize, create and update enactable process models (epm) for Galaxy projects. This activity will be available with 1st prototype.The second activity involves galaxy users (such as designers, developers, testers, etc.) during the process enactment. This 2nd activity will be available with the 2nd prototype.

### Activity performed by project managers and quality engineers

During the first activity, actors will usually customize a process model based on a generic process model (gpm) retrieved from the generic process models repository. The result will be an enactable process model which will be published on the Galaxy server thanks to the initialize enactable process model service. At this point, the epm will be completely managed by the Galaxy server and will available (read/write) to allowed galaxy users having the project manager role or quality engineer role from their personal PM Tool installed on their client machine.

An authorized actor having the project manager role or quality engineer role from their personal PM Tool installed on their client machine will be able to retrieve all the Galaxy epm available for him on the Galaxy server.

Until an authorized actor gets the lock on a given epm, this actor only has a read access on this enactable process model. Any authorized actor can request the lock on a given epm; the lock is granted to him if nobody else already own the lock on the epm. Once the lock has been granted to a given actor, this actor and only this will be able to edit the epm locally in his PM Tool.

An authorized actor with a lock granted on a given epm will be able to publish his modified epm on the Galaxy server as many times as needed until the lock is released. This publication will be delivered to all listening PM Tool clients.

An authorized actor with a lock granted on a given epm will be able to release the lock for this specific epm.

An automatic polling process will be launch by the PM Tool if an authorized actor has an epm managed the galaxy server (i.e. an epm previously published by someone on the galaxy server) opened and this actor doesn’t has the lock on this given epm. This polling process will download from the Galaxy server any update published of the managed epm.

### Activity performed by the galaxy users during process enactment

During the first activity (see chapter 2.1.1), project manager will assign defined tasks to Galaxy users involved in their project. This action will require the project manager to get the lock on the epm and modify it in order to affect tasks to users and then to publish the new version of the epm.

From the Galaxy users’ point of view, they will have access to their current affected tasks directly on the MDE Tool thanks to the Process Enactment client tool.

Galaxy users will be able to notify the Process Engine located on the Galaxy server of any event related to their affected tasks. Note, the available events for a given task will be dynamic and given by the Process Engine along with the tasks information it-self.

Event notifications sent by the Galaxy users will be handled by the Process Engine. As a result the related task’s state machine will be updated on the server. Eventually some other task’s state machine could be updated depending on pre-condition mechanism. For instance, the coding task for the user Laurent has the following pre-condition: Laurent’s design task must be finished. When Laurent will notify to the Process Engine that his design task is finished, it will have as consequence, the design task’s state machine update but also an update of the coding task’s state machine.

Once a Galaxy enabled MDE Tool is launched, the user must authenticate it-self to system. The Process Enactment Tool will then start an automatic polling mechanism which will retrieve from the Galaxy server, the current tasks affected and opened for the authenticated user. This mechanism will remain active until the MDE Tool is closed.

## Scope of the study

Focus here on the perimeter directly involved in the study case. Describe the scalability issues which are used to assess the added value of the platform

This use case will focus on the process management contribution in the Galaxy architecture. This contribution to the project doesn’t address directly scalability issues but provides added values on the organizational plan for large projects which involve a large number of contributors (projects managers, quality engineers, designers, developers, testers, etc…)

# Validation method

Explain the approach and the process applied on this study case.

The validation will be carried out in a two-phase process. First, the validation scenarios will focus of the 1st prototype of Galaxy architecture. Then the validation scenarios will address additional features provided by the 2nd prototype.

During each phase two sub-processes will be played in parallel. The first sub-process will be performed by AKKA and will focus on technical validation of the client/server architecture. The second sub-process will be performed by IRIT and will focus on the functional validation of the tools (i.e. diagrams authoring, enactment, etc.).

The advantage of this method is to allow an early start of the validation activities since the 1st prototype release (mid of June 2012) and will decouple the activities performed by the involved partners.

Softeam will be involved during the 2nd prototype validation in order to validate the technical part of the client/server architecture between the Galaxy server and the Process Enactment Client included in Modelio.

# Involved PARTNERS

All partner directly involved in this study case are listed here with their respective contribution.

The involved partners are AKKA, IRIT and Softeam.

# Validation Scenarios

Describe the validation scenarios here:

* Involved service of the Galaxy platform
* procedure,
* assessed metrics and corresponding objectives

In this chapter we will describe the validation scenarios.

## Create Generic Process Model

This scenario aims to assess:

* Ability to create and save locally Generic Process Models which are not managed by Galaxy server.

PM Tool must be able to create and save locally CM\_SPEM diagrams which could be used as Generic Process Models by project managers and/or quality engineers.

## Publish Generic Process Model into the GPM repository

This scenario aims to assess:

* Ability to publish previously saved Generic Process Models into the GPM repository.

PM Tool must be able to publish Generic Process Models into the GPM repository. The GPM repository is a shared directory accessible by all the authorized actors. This scenario aims to copy a given GPM file into the shared directory.

## Retrieve Generic Process Model from the GPM repository

This scenario aims to assess:

* Ability to retrieve locally from the GPM repository a previously published GPM in order to initialize the project manger and/or quality engineer work.

PM Tool must be able to retrieve locally from the GPM repository a previously published GPM in order to initialize the project manger and/or quality engineer work. The GPM repository is a shared directory accessible by all the authorized actors. This scenario aims to copy a given GPM file from the shared directory to a local directory with a given name. The new copied file is then open in the diagram editor in order to let the authorized actor to start to customize the process. The objective is then to update it in order to make it transform into a publishable enactable process model.

## Initialize Enactable Process Model on the Galaxy Server

This scenario aims to assess:

* Ability to publish a locally initialized EPM on the Galaxy Server.

PM Tool must be able to publish a non managed EPM file on the Galaxy Server. As a consequence, the published EPM will be managed by the Galaxy Server since that moment. The EPM will be accessible from the Galaxy Server for all authorized project managers and/or quality engineers. After a successful EPM publication on the Galaxy Server, the PM Tool shall automatically start the polling process in order to retrieve eventual EPM update coming from the server. Since the current user doesn’t have request the lock on this file, the PM Tool will not permit write action on this EPM. Read only access are granted until the lock is retrieve from the Galaxy Server.

## Open an Enactable Process Model from the local machine

This scenario aims to assess:

* Ability open a local EPM in the diagram editor and to start (if needed) the polling mechanism upon the Galaxy Server.

An EPM is a Galaxy managed process model which has been previously initialized by the current user or which has been downloaded from the Galaxy server (see chapter 5.1.3).

PM Tool must be able to open a local EPM in the diagram editor and to start (if needed ⬄ the current user doesn’t own the lock on this EPM) the polling mechanism upon the Galaxy Server.

## Receive the list of all available Enactable Process Model from the Galaxy Server

This scenario aim to assess:

* Ability to retrieve the list of all available Enactable Process Model from the Galaxy Server for a given authenticated used.

PM Tool must be able to present the list of available EPM accessible from the Galaxy Server for a given authorized project managers and/or quality. Thanks to this feature, a project manager and/or a quality engineer will be able to download from the Galaxy Server an EPM initialized by another user.

## Open an Enactable Process Model from the Galaxy Server

This scenario aims to assess:

* Ability to download from the Galaxy Server an EPM which is not already present locally and to open it in the diagram editor.

PM Tool must be able to download from the Galaxy Server an EPM initialized by another user and to open it locally once the EPM has been downloaded.

## Open several Enactable Process Models in parallel

This scenario aims to assess:

* Ability have several EPM opened in the diagram editor.

PM Tool must be able to handle that several EPM are opened simultaneously. Automatic polling mechanism must be set in order to guaranty the user is viewing an up-to-date version of the EPM.

## Request lock for an Enactable Process Model

This scenario aims to assess:

* Ability to request the lock on a given EPM.

PM Tool must be able request the lock on a given EPM on the Galaxy Server. Should this EPM be already locked by someone else, an error message should be presented to the user. Should this EPM not locked on the Galaxy Server, the lock will be set for the given user on the Galaxy Server and the automatic polling mechanism will be stop on the user client for this specific EPM. PM Tool will then let the user modify the EPM locally: read/write access.

## Work offline on a locked for an Enactable Process Model

This scenario aims to assess:

* Ability to work on and modify a locked EPM while no network connection is available.

PM Tool must be able to let the user work on and modify a locked EPM while no network connection is available. In this scenario, the login process will not challenge the Galaxy Server since no network connection is available. PM Tool will just ask the user for his name and check matching with lock owner present locally with the EPM.

## Request unlock for an Enactable Process Model

This scenario aims to assess:

* Ability to request unlock on a given EPM.

PM Tool must be able request unlock on a given EPM on the Galaxy Server. Should this EPM be already locked by someone else or not locked, an error message should be presented to the user. Should this EPM locked on the Galaxy Server for the current user, the lock will be reset on the Galaxy Server and the automatic polling mechanism will be restart on the user client for this specific EPM. PM Tool will then restrict the user access locally to read only.

## Publish updated Enactable Process Model on the Galaxy Server

This scenario aims to assess:

* Ability to publish on the Galaxy Server updated version of a locked EPM.

PM Tool must be able to publish on the Galaxy Server an updated version of a locked EPM. The publish service will not release the lock on the Galaxy Server; as a consequence, the current user still have read/write access on his local EPM. It must be possible to perform several publish before requesting unlock on a given EPM.

Once the Galaxy Server receives an updated version of an EPM, this new version becomes the new reference version. Should another PM Tool requests any update on this EPM via its polling mechanism, the Galaxy Server will send the new version just updated to this PM Tool.

## Receive updated Enactable Process Model from the Galaxy Server

This scenario aims to assess:

* Ability to receive on all listening PM Tools the up-to-date version of a given EPM.

Thanks to its automatic polling mechanism, the PM Tool must be able to request the Galaxy Server the last version of a given EPM if this EPM is not locked by the current user. In that case, the last version is obviously the local one.

Once the Galaxy Server receives an updated version of an EPM, this new version becomes the new reference version. Should another PM Tool requests any update on this EPM via its polling mechanism, the Galaxy Server will send the new version just updated to this PM Tool.

## Receive current active tasks from the Galaxy Server

This scenario aims to assess:

* Ability to receive current active tasks from the Galaxy Server for a given authenticated Galaxy user inside his MDE Tool (Papyrus/Modelio).

When a Galaxy user launches its MDE Tool, he has to authenticate him-self on the Galaxy Server. After a successful authentication, the Process Enactment Client inside the MDE Tool will request from the Galaxy server all current active tasks for the user. An automatic poll mechanism is started in order to be notified if new active tasks are affected to the current user. This mechanism is stopped when the MDE Tool is closed.

## Notify a event for a given task to the Galaxy Server

This scenario aims to assess:

* Ability for a given authenticated Galaxy user from his MDE Tool (Papyrus/Modelio) to notify the Galaxy Server of a task related event.

When a Galaxy user launches its MDE Tool, he has to authenticate himself on the Galaxy Server. After a successful authentication, the Process Enactment Client inside the MDE Tool will permit the user to notify the Galaxy Server for any current local active task.

## validation scenarios related to editing process models

In this section, we describe validation scenarios corresponding to a subset of the uses cases described in the Galaxy deliverable D4.1 Architecture Specification, section 4.1 6 Process Modeling and Enactment. The subset of selected D4.1 uses cases is related only to editing process models, not to enactment. As the PM Tool is mainly a Process Editor, we call it “Process Modeling Editor (PME)” in the following of this section. Process models produced by PME conform to the CM\_SPEM meta-model. We describe below scenarios which correspond to PME prototype.

### Scenario related to editing Generic process models

“Generic Development Process Model (GPM)” means the description of the know-how of development within an organization independently from particular projects.

This scenario aims to assess:

* Ability to create/describe/modify a GPM

PME allows a Process Designer to create a new GPM. It offers two ways to achieve that.

A first way is a hierarchical description provided by a tree editor.

The second way is a graphical one. PME provides diagrams which allow process designers to describe several views of the GPM. More precisely, the process designer uses a graphical interface to produce:

* An Activity structure diagram; this diagram describes an activity with the following process elements: name, roles, guidances, input and output products.
* An Activity flow diagram; this diagram represents the sub-activities of the current activity, and their sequencement.
* A State machine diagram; this diagram shows the lifecycle associated to a process element, typically a workproduct or an activity, as a UML-like state machine.
* Ability to check a GPM.

The PME ensures that a GPM is conform to the CM\_SPEM meta-model. So at any time during the edition, the process designer can ask for this conformity checking.

* Ability to store/retrieve/delete a GPM.

Whenever a GPM has been built, the process designer can store it into the GPM repository. The output format is XMI. Conversely, the process designer may retrieve an existing GPM from the GPM repository and load it into his current workspace. He may also delete a GPM from the GPM repository if he is the owner.

### Scenario related to Editing Enactable Process Models

“Enactable Process Model (EPM)” means the customization of a GPM to a particular development project.

This scenario aims to assess:

* Ability to create/describe/modify an EPM

PME allows a Process Designer to create a new EPM. It offers two ways to achieve that.

A first way is a textual description provided by a tree editor.

The second way is a graphical one. PME provided diagrams which allow process designers to describe several views of the GPM. More precisely, the process designer uses a graphical interface to produce:

* An Activity structure diagram; this diagram describes an activity with the following process elements: name, roles, guidances, input and output products.
* An Activity flow diagram; this diagram represents the sub-activities of the current activity, and their sequencement.
* A State machine diagram; this diagram shows the lifecycle associated to a process element, typically a workproduct or an activity, as a UML-like state machine.
* Ability to check a GPM.

The PME ensures that a GPM is conform to the CM\_SPEM meta-model. So at any time during the edition, the process designer can ask for this conformity checking.

* Ability to store/retrieve/delete a GPM.

Whenever a GPM has been built, the process designer can store it into the GPM repository. The output format is XMI. Conversely, the process designer may retrieve an existing GPM from the GPM repository and load it into his current workspace. He may also delete a GPM from the GPM repository if he is the owner.

# Involved models

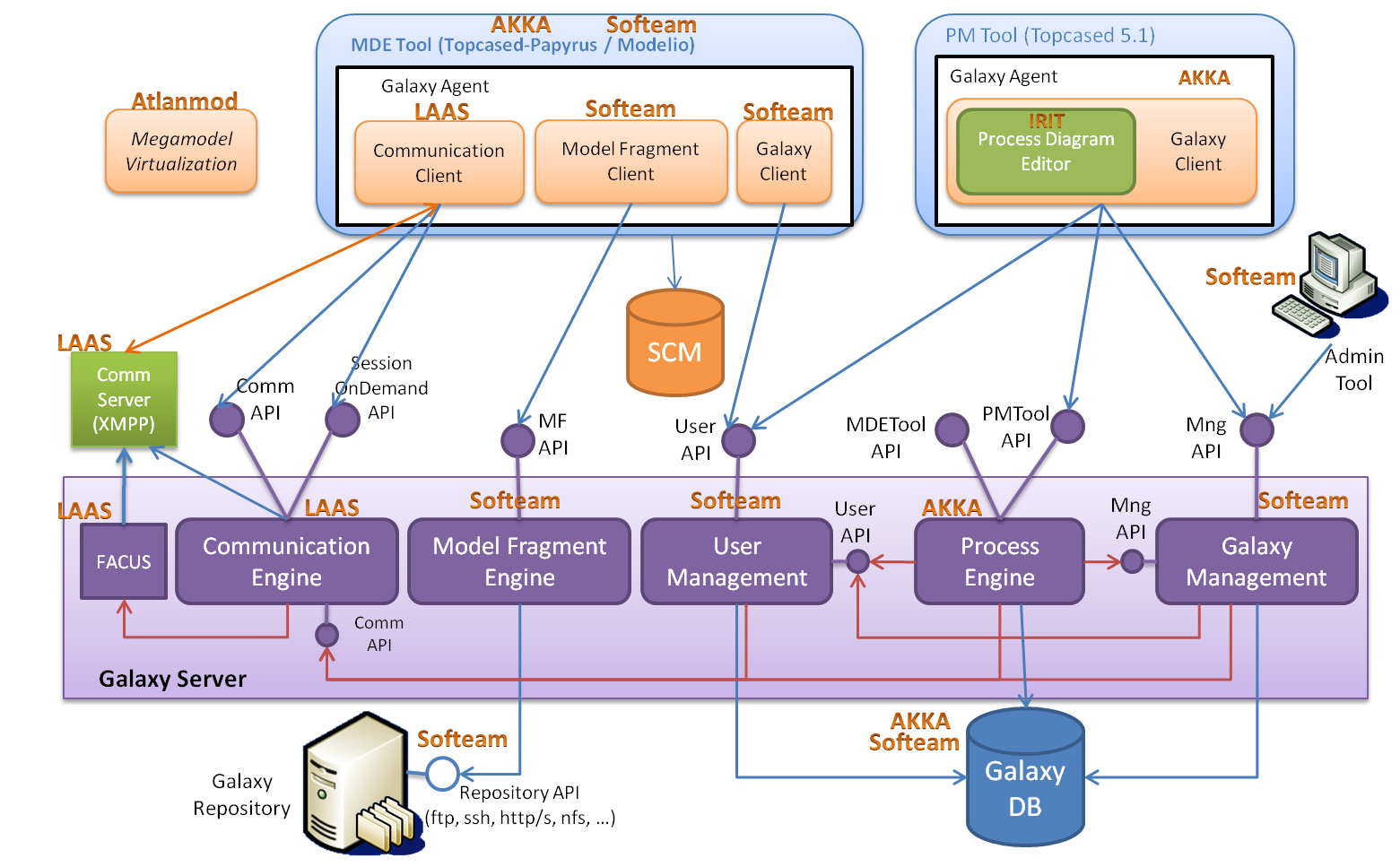
List and describe the kind of models (view, viewpoint) involved in the study case and the way they are organized (relationships)

N/A

# tools used

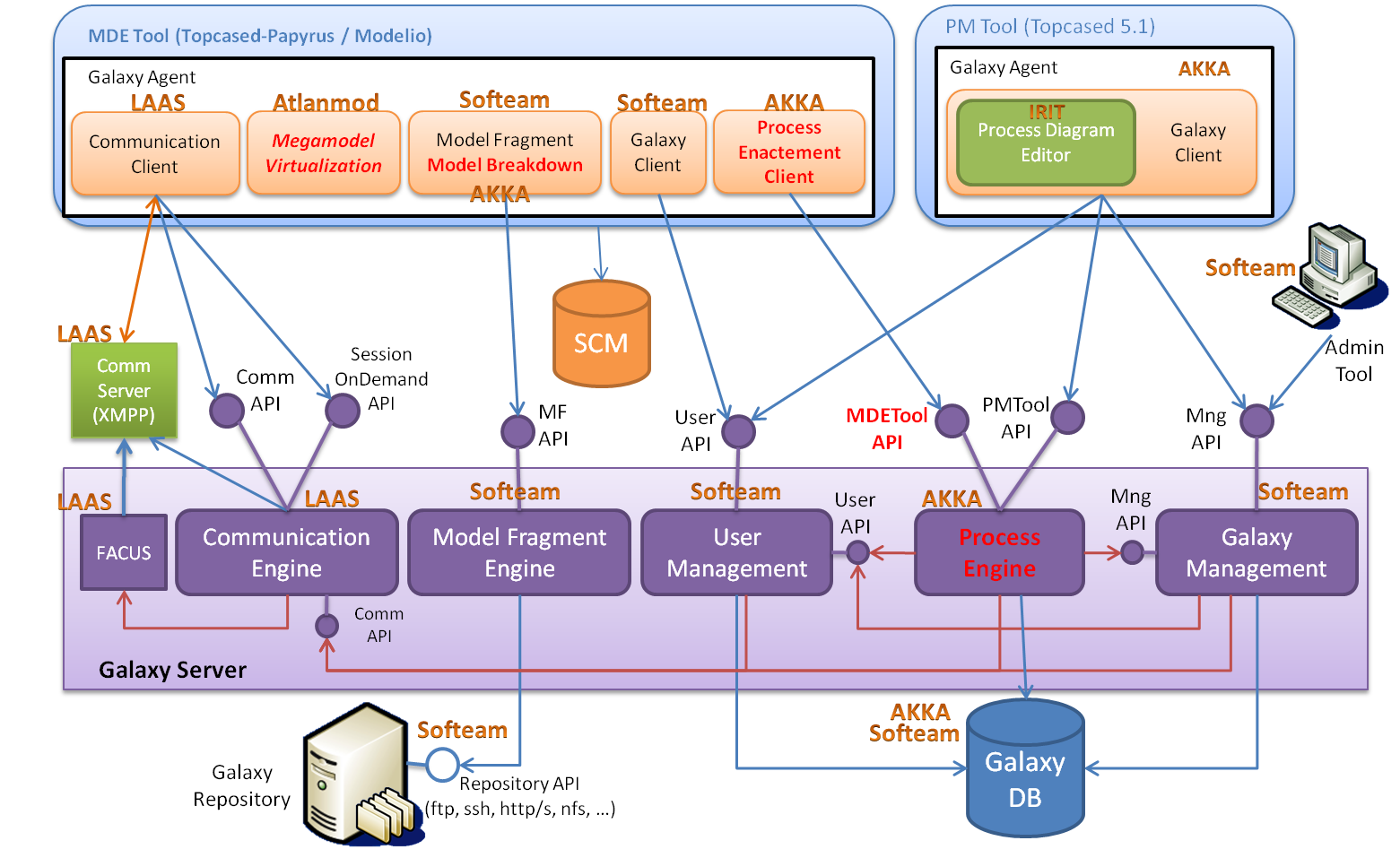
List the tools used in this study case with their corresponding version and purpose.

During the 1st validation phase, the following tools will be used:



* Process Management Tool
  + Based on Topcased 5.1
    - CM\_SPEM diagram editor
    - Galaxy client glue (technical middleware in order to communicate with the Galaxy server)
* Galaxy Server
  + Process Engine
    - PMTool API

During the 2nd validation phase, the following tools will be used:



* Process Management Tool
  + Based on Topcased 5.1
    - CM\_SPEM diagram editor
    - Galaxy client glue (technical middleware in order to communicate with the Galaxy server)
* Galaxy Server
  + Process Engine
    - PMTool API
    - MDETool API
* Topcased/Papyrus based MDE Tool
  + Galaxy Agent
    - Process Enactment Client
* Modelio based MDE Tool
  + Galaxy Agent
    - Process Enactment Client