# D1.2.2 Galaxy Glossary
v1.0

| | NAME | PARTNER | DATE |
|---|---|---|---|
| WRITTEN BY | R. LBATH, B. COULETTE, E. KEDJI, H.N. TRAN, S. EBERSOLD | IRIT | |
| | Y.BERNARD | Airbus | |
| | F. RACARU | AKKA | |
| | W. KLING | ATLANMOD | |
| | J. ROBIN | LIP6 | |
| REVIEWED BY | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

# RECORD OF REVISIONS

| ISSUE | DATE | EFFECT ON | | REASONS FOR REVISION |
|---|---|---|---|---|
| | | PAGE | PARA | |
| 1.0 | 03/29/2011 | | | Création du document |
| | | | | |

# TABLE OF CONTENTS

# TABLE OF APPLICABLE DOCUMENTS

| N° | TITLE | REFERENCE | ISSUE | DATE | SOURCE | |
|---|---|---|---|---|---|---|
| | | | | | SIGLUM | NAME |
| A1 | | | | | | |
| A2 | | | | | | |
| A3 | | | | | | |
| A4 | | | | | | |

# TABLE OF REFERENCED DOCUMENTS

| N° | TITLE | REFERENCE | ISSUE |
|---|---|---|---|
| R1 | Galaxy glossary | | |
| R2 | | | |
| R3 | | | |
| R4 | | | |

# 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 |
|---|---|
| | |
| | |
| | |
| | |

**Definitions**

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

# 1. INTRODUCTION

## 1.1 GOAL OF THIS DOCUMENT

This document does not aim to constitute a general-purpose dictionary in the domain of collaborative model-driven software development, but to constitute rather a reference glossary shared by all the partners of the project Galaxy.

It provides a definition of technical concepts that are to be used all along the project Galaxy, taking into account the definitions found in the literature and in comparable projects. Each concept definition is organized into the following sections:

- Definition(s) from Literature: This section gives a selection of some definitions found in academic research works and projects comparable to Galaxy. The objective here is not to provide an exhaustive list of the existing definitions but only to select those that are very close to the project Galaxy.
- Galaxy Definition: This section gives the definition commonly adopted and shared by all the project Galaxy's partners.
- See Also: This section refers to the concomitant Galaxy concepts.
- Example(s): This section gives examples that illustrate the concept definition.
- Comment: This section adds complementary explanations and precisions.
- Galaxy Usefulness: This section indicates how the defined concept is useful to and how it is indented to be used in the project Galaxy.

The sections Galaxy Concept, Galaxy Definition, and Galaxy Usefulness are provided for each concept definition. The others sections are provided whenever they may help the reader better understand the concept definition.

This version 1.0 of the glossary provides a first set of the concepts identified in the Galaxy's work packages WP2 (Conceptual Model for Model Driven Collaborative Development), and WP3 (Transformations with Large and/or Numerous Models), at the end of the first year of the project Galaxy. It is intended to be modified and updated whenever it is needed during the remaining two years of the project.

## 1.2 DOCUMENT ORGANIZATION

The rest of the document consists of two chapters: chapter 2 where concepts are defined, and chapter 3 which lists references to documents cited in chapter 2. Each concept constitutes a section of chapter 2.

Template pour copier/coller (§ provisoire)

- ***Definition(s) from literature***

**[Ref.]**

- ***Galaxy definition***

- ***See also***

- ***Example(s)***

- ***Comment***

- ***Galaxy Usefulness***

# 2. GALAXY CONCEPTS

## 2.1 ARTIFACT

- ***Definition(s) from literature***

**[Lambda]** Any information item produced or consumed during the development, the deployment or the maintenance of a system

**[UML]** An artifact is the specification of a physical piece of information that is used or produced by a software development process, or by deployment and operation of a system.

- ***Galaxy definition***

A versioned container of model elements and/or model views. Persistently stored in a Collaborative Unit (CU). Galaxy defines three main artifacts meta-classes: Reuse Unit (RU), a model element container, Method Units (MU), a model view container and Product Unit, a recursive containment tree which leaves are RUs and MUs.

- ***See also***

- ***Example(s)***

  (Source) code file, specification document, user manual, test data set, execution trace, model files, scripts, binary executable files, a table in a database system, a development deliverable, a word-processing document, a mail message.

- ***Comment***

In a Galaxy Collaborative Unit (CU), artifacts are stored with their revision history. This information includes a pointer to their preceding version. This preceding version can be stored explicitly and/or implicitly by storing the structural diff from the current version and/or the behavioral diff from the current version (*i.e.,* the trace of the artifact revision actions that changed the preceding version into the current one). The revision history information also includes the date, time, author and comment of the commit action that wrote it to the CU.

- ***Galaxy Usefulness***

The concept of artefact is linked to the physical storage of information which is one of the major issues regarding the scalability.

## 2.2 COLLABORATION

- ***Definition(s) from literature***

**[Wikipedia]** Collaboration is the act of working jointly with others, especially in an intellectual endeavor.

**[Wordnet]** Collaboration is the act of working together on a common enterprise of project.

**[Roschelle]** Collaboration is a coordinated, synchronous activity that is the result of a continued attempt to construct and maintain a shared conception of a problem.

- ***Galaxy definition***

Collaboration is a collective process where participants agree to contribute, each in a non-predetermined way, towards a common goal. It is based on permanent interactions, while maintaining a shared and evolving understanding of the process and the goal.

- ***See also***

Cooperation, Coordination

- ***Example(s)***

The engineering of any reasonably complex product is an inherently collaborative activity, with at least two dimensions. On the one hand, specialists from different domains have to continuously make sure their efforts not only improve the aspect of the system they are responsible for, but also does not conflict with other aspects of the systems; the latter requiring continuous coordination with specialists from other domains. On the other hand, within the same speciality, participants strive everyday to known what others are doing, and communicate about their own ongoing efforts and intentions, so as to maintain the shared understanding needed for everyday decision making.

- ***Comment***

Cooperation is focused on the objective, while collaboration is much more about the process. The participants in a collaborative endeavor are more or less interchangeable (they have flexible roles). There can be some sort of work division and role assigning, but it is ad-hoc and changing.

- ***Galaxy Usefulness***

Galaxy is interested in the engineering of complex systems, which necessarily requires large teams. As the understanding one can have of such systems continually evolves until project completion, it is critical to make sure that participants have the appropriate tools to make collaboration frictionless.

## 2.3 COLLABORATIVE MDE PROCESS

- ***Definition(s) from literature***

- ***Galaxy definition***

A collaborative MDE process is a collaborative process that takes into account the peculiarities of MDE development.

- ***See also***

  Collaborative process

- ***Example(s)***

  Any project where models are first-class artifacts can be described by a collaborative MDE process.

- ***Comment***

- ***Galaxy Usefulness***

Galaxy focuses on purpose on the use of MDE in the development of complex systems. It is therefore necessary to tailor collaborative processes to this particular engineering style.

## 2.4 COLLABORATIVE PROCESS

- ***Definition(s) from literature***

**[Lonchamp & Seguin, 1996]** A collaborative process is a process in which several participants perform together some creative work. This creative work progresses mainly through collective decisions (individual decisions being a special case), asynchronism is the basic interaction mode, and collaboration takes places mainly within a shared workspace containing the product being constructed, and the process history, including the process design rationale.

**[Kolfschoten et. Al., 2006]** A collaborative process is a process in which people join forces to accomplish goals through collaboration that they could not achieve as individuals.

**[Giensen, 2007]** A collaborative process is a process that requires parity among participants, is based on mutual goals, depends on shared responsibility for participation and decision making, implies shared resources, requires shared accountability for outcomes, and relies on mutual trust.

- ***Galaxy definition***

A collaborative process is a process with at least one collaborative activity. A collaborative activity is an activity carried out by two or more human participants, with shared accountability.

- ***See also***

  Collaboration

- ***Example(s)***

  Fixing a bug is a collaborative activity, which involves at least the user who reported the bug, the developer who reproduced the problem, the developer in charge of fixing the bug, and possibly the developer who introduced the bug.

- ***Comment***

The most important issue with collaborative activities is how to make collaboration seamless by providing sufficient and adequate communication bandwidth.

- ***Galaxy Usefulness***

Collaboration can be a productivity multiplier when properly handled, as well as slow everyone down when it induces too much friction. The engineering of complex systems being necessarily

collaborative (see Collaboration), capturing and optimizing the critical aspects of collaboration in a process model can help making the most out of it.

## 2.5 COLLABORATIVE PROCESS PATTERN

- ***Definition(s) from literature***

- ***Galaxy definition***

Pattern describing typical scenario of a collective work in a model-driven software process. It reflects process knowledge encompassing not only the tasks but also the involved actors, products, and resources. It captures a recurrent situation of collaborative work (on models) specified by a goal of the collaboration, a context (scenario) of collaboration together with potential problems of such a situation and proposes a solution for organizing and coordinating the involved factors (task, roles, and artifacts) in order to accomplish the objective of the collaborative work.

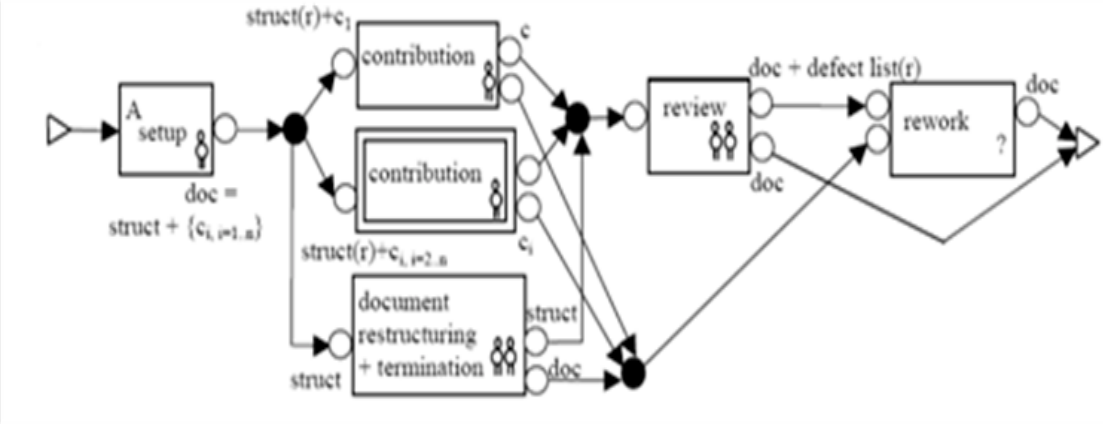
- ***Example(s)***

Pattern: Division of labor.

Goal: Collaborative production of a document.

Context: Several actors perform in parallel and independently some partial work for building the document. Their work is mainly independent because it concerns different subparts of the document ($c_i$, in the figure below). However, their work can sometimes interfere, in particular when a modification of the document structuring (struct in the figure below) is required. In this case, a collective document restructuring task creates a new decomposition that will be reworked. This task is also responsible for the collective decision to terminate the parallel work. When termination has occurred, a collective review of the whole document is performed for ensuring its global consistency.

Solution:




- ***Comment***

Why do we need to identify collaborative process patterns for model-driven software process?
- To identify how we divide works on models (i.e. how to fragment models)
- To identify operations/manipulation on models, and model fragments relationships (in order to clarify the concept of collaborative unit and the influence of model-driven process on it).
- To identify coordinating tasks, and roles in model-driven software process (i.e. requirements of collaborative model-driven development assistance and management).

- ***Galaxy Usefulness***

Galaxy deals with the complexity and the heterogeneity of collaborative developments. It aims at providing solutions to resolve the scalability and the synchronizing problem in complex MDE projects. Besides the technical solutions, the complexity of a collaborative process can be reduced if we use the process semantics to specify which process aspects are important to be described and controlled. Capturing proven solutions for collaborative development, collaborative process patterns can be usefule in Galaxy to define the efficient strategies for organizing collaborative process in specific contexts. Concreteley, for a given situation, they can help identify just neccesary process tasks; roles and artifacts; they suggest how to organize, coordinate these tasks, roles and artifcats in order to minimize the dependencies.

## 2.6 COLLABORATIVE UNIT

- ***Definition(s) from literature***
  None. This concept is an innovative conceptual contribution of the project. The entire project deliverable D2.1. is fully dedicated to define this concept.
- ***Galaxy definition***

The complex object-oriented data structure that supports large scale collaborative multi-view model edition. A workspace to store the history of project artifacts and perform collaborative revision control actions on them. A general concept that subsumes local main memory workspaces as well as persistent local and global repositories.

- ***Examples***
- A Subversion repository containing the XMI files representing various versions of project model fragments together with their revision history metadata;
- A Mercury repository containing Eclipse Modeling Framework (EMF) objects representing various versions of project model fragment, together with their revision history metadata objects, all persisted on a computing cloud infra-structure.
- ***Galaxy Usefulness***

Central to the project since most deliverables of project tasks 2 and 4 must reuse the concept in their own defintions.

- ***Comment***

Was defined in a sufficiently versatile and generic way to support:

- both the sequential *(i.e.,* lock-change-unlock) and concurrent (*i.e.,* copy-change-merge) collaborative revision approaches;
- both centralized and distributed revision control;
- revision control using any modeling language or software process.

## 2.7 CONFIGURATION

- ***Definition(s) from literature***

**[Lambda]** (Of a composite artefact) the list all the components of an artefact with their version

**[TOPCASED]** Consistent set of version of component which is selected for configuration management purposes. Configuration Managements is the activity for establishing and maintaining consistent record of functional and physical characteristics compared to the Configuration. The Configuration Management is applied through the entire life-cycle of the product.

**[TechTarget]** Generally, a configuration is the arrangement - or the process of making the arrangement - of the parts that make up a whole. Configuration management is also used in

software development in order to keep track of the source code, documentation, problems, changes requested, and changes made.

**[Wikipedia] Configuration** is an arrangement of functional units according to their nature, number, and chief characteristics. Often, configuration pertains to the choice of hardware, software, firmware, and documentation**. Configuration management** focuses on establishing and maintaining consistency of a system's or product's performance and its functional and physical attributes with its requirements, design, and operational information throughout its life. **Software configuration management (SCM)** process identifies the functional and physical attributes of software at various points in time, and performs systematic control of changes to the identified attributes for the purpose of maintaining software integrity and traceability throughout the software development life cycle.

- ***Galaxy definition***

In the context of Galaxy project, we use the term of Configuration with the meaning of a system configuration managelent.

- ***See also***
    - Version

- ***Example(s)***
    - tracking and controlling changes in the source code
    - creating and maintaining an up-to-date record of all the components of the infrastructure, including related documentation

- ***Comment***

- ***Galaxy Usefulness***

## 2.1 CONFLICT

- ***Definition(s) from literature***

- ***Galaxy definition***
  A conflict is a possible state for a difference between two versions of an element in the context of a Diff/Merge operation. This state signifies that there is no single solution consisting of merging two concurrent modifications.

- ***See also***
  Incoherence

- ***Example(s)***
  If two people change the name of a class, for example, then there are two solutions and therefore a conflict.
  If a person deletes an element and another person adds a link to this element, then there is no solution to merge the two modifications and therefore a conflict.

- ***Comment***
  A conflict is not to be confused with an “incoherence”.

- ***Galaxy usefulness***
  In view of carrying out Diff/Merge operations on models, we need to talk about conflicts.

## 2.2 COOPERATION

- ***Definition(s) from literature***

**[Wikipedia]** Cooperation is the act of working or acting together, and can be accomplished by intentional and non intentional agents. It contrasts with working separately in competition.

**[Heutte]** Cooperation results from an a priori negotiated (planned) division of a task into actions, which will be distributed (attributed) between individuals working autonomously. Interactions are limited to organisation, coordination, and follow-up (usually under the responsibility of someone who should guaranty the individual performance of everyone). Everyone's responsibility is limited to making sure the actions assigned to him are done: it is the progressive and coordinated concatenation of the results of everyone's action that makes the goal a reality.

- ***Galaxy definition***

Cooperation is a collective effort where roles and responsibility are a priori delimited towards a clear goal. Each participant in a cooperative effort has a distinct and specialized role, a clear responsibility, and does not need to care what others are doing.

- ***See also***

Collaboration, Coordination

- ***Example(s)***

The network, the OS network stack, the OS scheduler, a web browser, and a display technology can be described as cooperating towards the goal of helping a computer user navigate the web.

- ***Comment***

Cooperation is focused on the goal, while collaboration is much more about the process.

- ***Galaxy Usefulness***

Cooperation can be seen as a minimal form of collaboration, and have acquired, in technology, a distinct meaning as it is commonly used to describe interactions between automated agents (these interactions are governed by cooperation protocols). Galaxy will focus on the term “collaboration” which better captures the collective effort involved in building complex systems.

## 2.3 COORDINATION

- ***Definition(s) from literature***

**[Wikipedia]** Coordination is the act of making different people or things work together for a goal or effect.

**[Wordnet]** Coordination is the regulation of diverse elements into an integrated and harmonious operation.

- ***Galaxy definition***

Coordination is the regulation of the efforts of diverse contributors into a harmonious whole, towards a certain goal or effect.

- ***See also***

Collaboration, Cooperation

- ***Example(s)***

The work done by a manager in a team is a coordination work.

- ***Comment***

Coordination implies a role distinct from working on the main task at hand, a role made necessary only by the plurality of participants.

- ***Galaxy Usefulness***

Coordination can be described as the paperwork of collaboration, a sort of necessary evil, as it is does not directly participate to the main goal. As such, one strives to automate most aspects of it, so as to make it “invisible”. Therefore, one of the goals of the tools to be developed in Galaxy is to take care of most of the coordination work so that participants can focus on their main endeavours.

## 2.4 COMPLEXITY

- ***Definition(s) from literature***

**[Lambda]** Resistance to the reductionist approach. Something is said “complex” when the understanding of its components is not sufficient to grasp the whole.

- ***Galaxy definition***

From the Galaxy point of view a system is complex when the conventional development tools and model-based technologies cannot be used with the required level of quality (especially performance and reliability) required for an industrial project.

- ***See also***

- ***Example(s)***

The IMA embedded computers are complex systems.

- ***Comment***

Several factors may impacts the system complexity. The mains of them are: the size of the model, the size of the development team and the number and diversity of viewpoints to be considered and synchronized.

- ***Galaxy Usefulness***

Ability for MDE technologies to deal with complex system is the main target of the Galaxy project.

## 2.5 CONTAINER

- ***Definition(s) from literature***

**[Wiktionary]** An item in which objects or materials can be stored or transported

- ***Galaxy definition***

An artifact with no specific semantics intended to be use as a pure organizational unit.

- ***See also***

Artifacts

- ***Example(s)***

  UML package.

- ***Comment***

- ***Galaxy Usefulness***

Containers provide a means to organize information in arbitrary tree-like structures. They can be useful to carry out miscellaneous strategies for the organization of engineering data.

## 2.6 INFORMATION

- ***Definition(s) from literature***

**[Wiktionary]**Things that are or can be known about a given topic; communicable knowledge of something

- ***Galaxy definition***

Any data describing the specification of the system to be developed, how those data are organized, or how the system is developed.

- ***See also***

- ***Example(s)***
- Engineering data, metamodel and model structure, development process.***Comment***

- ***Galaxy Usefulness***

Generic concept to designate “things that are to be known” about the development of a system whatever their domain, level of abstraction or finality.

## 2.7 INCOHERENCE

- ***Definition(s) from literature***

- ***Galaxy definition***

  Incoherence is a possible state for an element in its model. This state signifies that the element does not respect certain syntactical rules.

- ***See also***

  Conflict

- ***Example(s)***

  An inheritance cycle is a syntactical incoherence in the context of a model whose objective is to generate Java code, for example.

- ***Comment***

  Incoherence can appear after a Diff/Merge operation, but must not be taken into account during the operation itself.

- ***Galaxy usefulness***

  "Incoherence" must be defined in order to be able to properly distinguish between an incoherence and a conflict during a Diff/Merge operation.

## 2.8 INFORMATION ITEM

- ***Definition(s) from literature***

**[TOPCASED]** Similar to "configuration item" in TOPCASED which represents a component which is selected for configuration management purposes.

- ***Galaxy definition***

An element of information that is individually managed and exchangeable. Depends on the revision strategy.

- ***See also***

- ***Example(s)***

- ***Comment***

- ***Galaxy Usefulness***

An information unit identifies a unit of information that can be share by replication.

## 2.9 LIBRARY (OF MODELS)

- ***Definition(s) from literature***

**[Lambda]** Organization structured to manage the reuse of model elements

- ***Galaxy definition***

Same as above

- ***See also***

- ***Example(s)***

  A model repository.

- ***Comment***

- ***Galaxy Usefulness***

Industrial MDE will make an intensive use of reusable model components. Those components will have to be managed in consistent library, just like those existing for the Java or C programming languages.

## 2.10 MEGAMODEL

- ***Definition(s) from literature***

  [Bezivin 05] A megamodel ia a model which elements represent models, metamodels and other global entities.

- ***Galaxy definition***

  A Megamodel is a model such that all its elements represent models (like models, metamodels, metametamodels etc). The Megamodel captures the metadata that describes the models of a system and the relations between them. It captures relations like weaving models, transformations, model traces, and conformance relationships among others.

- ***See also***

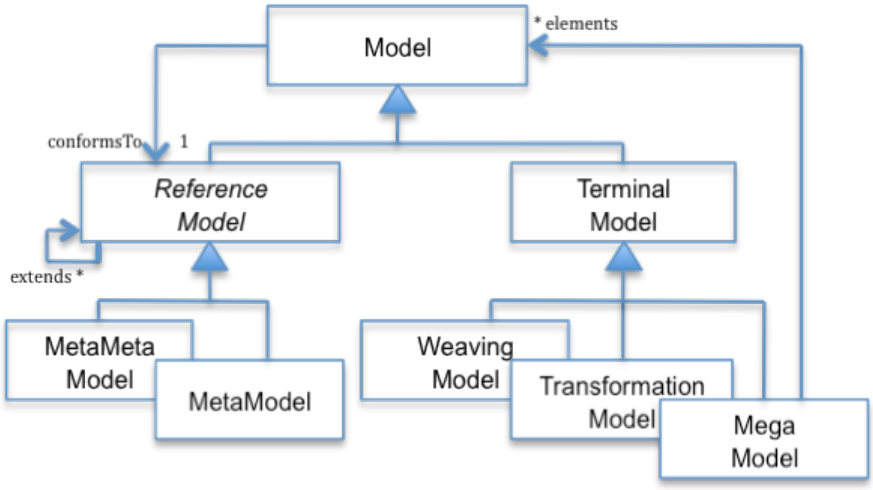
  *View*

  ***Example(s)***

  A model which elements represent each and every artifact produced during a MDE oriented software development process e.g., requirement specifications models, analysis specifications models, code, data files, transformations etc.

- ***Comment***

  A Megamodel relies on the concepts of Model, Reference Model and Terminal Model. A reference model is a model to which other models may conform, thus a reference model in turn may be a Metametamodel or a Metamodel. A terminal model is a model to which no other model conforms. A Megamodel and a Transformation Model are examples of terminal models. A Megamodel is a model, which represents a system it contains (or represents) models, thus it may also contain other megamodels.




- ***Galaxy Usefulness***

  The Megamodel acts as an index of all of the models of the system and the relations between them. This kind of registry of models facilitates its manipulation in automated ways, like e.g., the navigation between models, the search of models by specific properties,

their inspection and transformation, which is very useful when handling with large amounts of models.

The Megamodel serves also as a global view of different modelling resources by abstracting the global elements and operations for enabling interoperability between them and allowing them to be handled through common interfaces.

## 2.11 MODEL DRIVEN DEVELOPMENT PROCESS

- ***Definition(s) from literature***

  **[Kleppe]** The process of developing (software) using different models on different levels of abstraction with (automated) transformations between models.

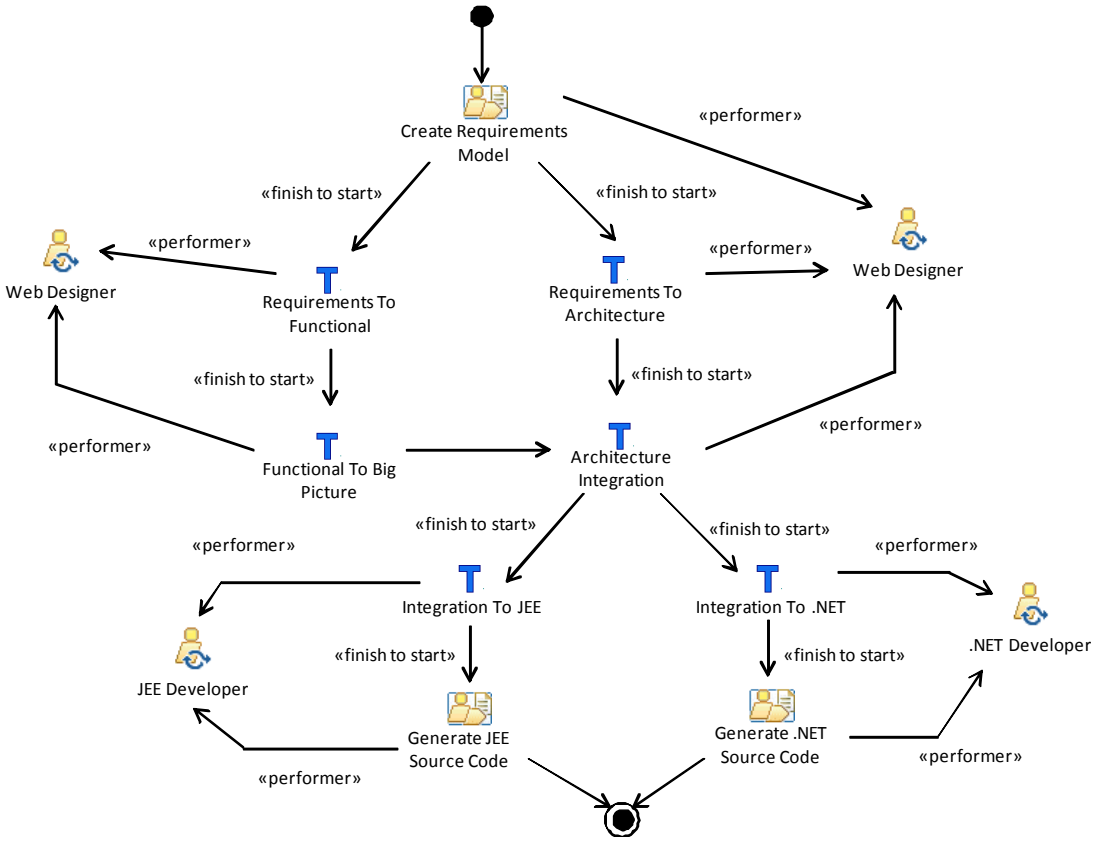
- ***Galaxy definition***

  The process of developing software and systems based on models and models transformations, and on other "traditional" activities and artifacts as well.

- ***See also***

- ***Example(s)***

The following figure shows an example of model-driven development process: an extract of the UML-based Web Engineering (UWE) process [Koch 06]. It starts the activity of creating of a Requirements Model. Then, a series of models transformations are to be performed before the activities of generating JEE Source Code and/or .NET Source Code can be achieved.



- ***Comment***
- ***Galaxy Usefulness***

  An explicit model of the model-driven development process will be used for offering process-based support to collaborative model-driven development.

## 2.12 MODEL

- ***Definition(s) from literature***

  [UML] A model captures a view of a system. It is an abstraction of the system, with a certain purpose. This purpose determines what is to be included in the model and what is irrelevant. Thus the model completely describes those aspects of the system that are relevant to the purpose of the model, at the appropriate level of detail.

- ***Galaxy definition***

  A useful abstraction of a system. Consists of model elements, model references between them and model views pointing to selected model element subsets. Also includes the model diagrams, which are visual rendering of model views.

- ***See also***
- ***Examples***

  - The containment tree of a UML model or a Domain-Specific Modeling Language (DSML) model including all the model elements and diagrams

- ***Comment***
- ***Galaxy Usefulness***

## 2.13 MODEL ELEMENT

- ***Definition(s) from literature***
- ***Galaxy definition***

A *model element* is a part of a model that corresponds to an instance of a concept (metaclass) defined within the model's metamodel.

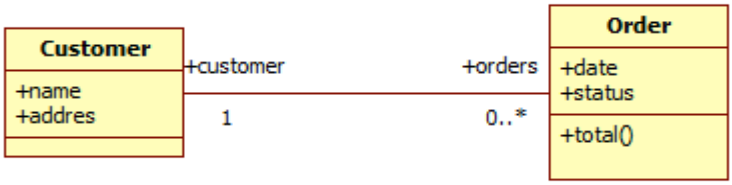
- ***See also***

Model, Metamodel, Model Fragment

- ***Example(s)***

The class diagram below models a customer order from a catalogue. The model conforms to the UML metamodel and consists of the following model elements:
- The classes Customer, and Order, which correspond to instances of the UML concept Class;

- The attributes name, address, date, and status, which correspond to instances of the UML concept : Attribute;
- The associations Customer-Order, which corresponds to an instance of the UML concept Association;
- The operation total(),which corresponds to an instance of the UML concept Operation;
- The association roles customer, and orders, which correspond to instances of the UML concept association's Role;
- The multiplicities 1, and 0..*, which corresponds to instances of the UML concept MultiplicityElement.




- ***Comment***

- ***Galaxy Usefulness***

## 2.14 MODEL FRAGMENT

- ***Definition(s) from literature***

**[Siikarla]** A copy of a piece of a model. It contains instances of metaclasses, meta-associations, and meta-attributes that are (possibly partial) copies of instances in the source model. A model fragment is a concrete and distinct entity in itself and can, e.g. be modified independently.
A model fragment can be based on, for example, model hierarchy (e.g. a package), syntactical aspects (e.g. all stereotypes), or semantic aspects (e.g. user identification related).

- ***Galaxy definition***

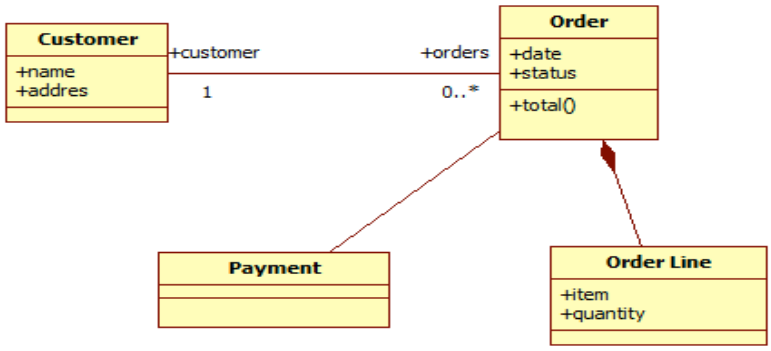
A *model fragment* is a part of a model, called the *source model*, that satisfies two conditions: (1) all its elements belong to the source model; (2) it forms a model that conforms to the source model's metamodel.

- ***See also***

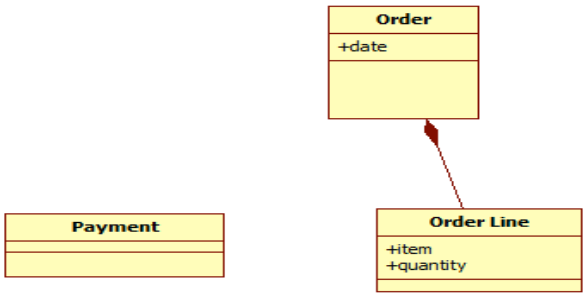
Model, Model Element, Metamodel

- ***Example(s)***

The class diagram below models a *customer* order from a catalogue. An *order* is composed of *order* lines and is associated with a *payment*. The model conforms to the UML metamodel.




The following class diagram is an example of model fragment. It consists of extracting from the source model, a part of the class *Order* (only the attribute *date*), the class *Payment*, the class *Order Line*, and the composition association *Order* / *Order Line*.




- ***Comment***
- ***Galaxy Usefulness***

Model fragments will be used for decomposition and composition of models in the context of collaborative model-driven development.

## 2.15 MODEL VIRTUALIZATION

- ***Definition(s) from literature***

- ***Galaxy definition***

  Model virtualization is the abstraction of model elements and its relations contained within a variety of models so that they may be accessed without regard to their physical storage or heterogeneous structure.

- ***See also***

  **Erreur ! Source du renvoi introuvable.**, virtual Model

- ***Example(s)***

- ***Comment***

- ***Galaxy Usefulness***

  Model virtualization is the technique used to compose several models into single models, which serve as views of the system. Model virtualization is also used for simplifying the work with large models by presenting to the user only the model content subset relevant to him.

## 2.16 MODEL WEAVING

- ***Definition(s) from literature***
- ***Galaxy definition***

  Model weaving is the operation for capturing different types of relationships, such as match relationships, between elements of different models.

- ***See also***
- ***Example(s)***
- ***Comment***
- ***Galaxy Usefulness***

  Model weaving is used as mechanism for storing the virtual links of a virtual model, which in turn act as a view of one or several concrete models.

## 2.17 PARTICIPANT

- ***Definition(s) from literature***
- ***Galaxy definition***

  A developer or an institution that participates to a collaborative MDE project using the Galaxy framework. It corresponds to an actual, specific developer or institution. It is not merely a generic role of a software process.

- ***See also***
- ***Example(s)***

  Redouane Lbath and IRIT are participants of the collaborative ANR Galaxy project.

- ***Comment***

  In a given project, each participant is allocated a collaborative unit to cooperate with the other participants.

## 2.18 PROCESS ENACTMENT

- ***Definition(s) from literature***

- ***Galaxy definition***

  Process enactment is the action of carrying out a process according to the process model, i.e. a computer-based execution of a process model that involves both automatic execution of activities and human execution.

- ***See also***

- ***Example(s)***

- ***Comment***

  Rather than “process execution”, the term “enactment” is used to be neutral between interpretation and compiled execution and to indicate that both human and computerised agents are involved in carrying out the process.

  The conventional approach to process enactment consists in providing a Process-centred System Engineering Environment (PSEE) that is meant to support the development process. The process model specifies how people should interact and work, how and when automated tools are activated or invoked. The PSEE takes then as input a Process Model and "behaves" according to what it is defined within the model. The PSEE offers a variety of services, such automation of routine tasks, invocation and control of development tools, enforcement of mandatory rules and practices, and management of artefacts.

  The PSEE is characterized by the process modeling language that defines its input process models and may have different user support goals (e.g., passive guidance, active guidance, or process enforcement).

- ***Galaxy Usefulness***

  Process enactment will be used to provide computer-based assistance and control for model-driven collaborative development of complex systems.

## 2.19 PROCESS MODEL

- ***Definition(s) from literature***

- ***Galaxy definition***

  A process model is a description of system development process, in terms of a process modelling language or a process metamodel.

- ***See also***

- ***Example(s)***

- ***Comment***

  A process modelling language, depending on the domain, should support the description of several concepts that characterize system development processes. In the context of Software Process Modelling, early classifications of the constituents of software process models have been proposed in the literature. We give here an essential summary of each element:
  - Activity: A concurrent process step, operating on artefacts and coupled to a human agent or a production tool. It can be at different abstraction levels i.e., activities can be decomposed. They can be at almost any level of granularity.
  - Artefacts: A product created or modified during a process either as a required result or to facilitate the process. They are the input and output of activities. An artifact can be simple or composite and may have relationships with other artifacts.
  - Role: Defines rights (i.e., permissions) obligations and responsibilities of the human agent involved in the software activity. A Role is a static concept while the binding between a role and an agent can be dynamic. A role can be played by several agents and inversely, an agent can play several roles.
  - Human: Human are process agents who may be organized in teams. They have skills and authority and can fulfil a set of roles. They are in charge of executing certain activities that compose the process.
  - Tool: Relates to any tool used by the software process, may be batch (e.g. compilers) or interactive (e.g. graphical editor).

- ***Galaxy Usefulness***

  An explicit model of the development process will be used for offering process-based support to collaborative development.

## 2.20 PRODUCT

- ***Definition(s) from literature***

  **[TOPCASED]** A part of system which is identified and delivered to accomplish a specified function

  **[TechTarget]** In information technology, a product is something (for example, a software application) that is created and then made available to customers, usually with a distinct name or order number.

  **[WordNet]** An artifact that has been created by someone or some process

- ***Galaxy definition***

  Galaxy project will use the term “poduct” with the same meaning as in the TOPCASED project.

- ***See also***

- ***Example(s)***

  - For Galaxy project, the product is represented by the project deliverables (and associated software) that make up or contribute to delivering the objectives of the project

- ***Comment***
- ***Galaxy Usefulness***

## 2.21 PROJECT

- ***Definition(s) from literature***

  **[PMBOK]** A project is a temporary endeavor undertaken to create a unique product, service or result.

  **[CMMI]** A "project" is a managed set of interrelated resources that delivers one ormore products to a customer or end user. This set of resources has a definite beginning and end and typically operates according to a plan. Such a plan is frequently documented and specifies the product to be delivered or implemented, the resources and funds used, the work to be done, and a schedule for doing the work. A project can be composed of projects."

- ***Galaxy definition***

  A collaborative endeavor aiming at producing and maintaining an independently usable system.

- ***See also***

  Participant, Collaborative Unit.

- ***Example(s)***
  - An Eclipse.org project.
  - A Google code project.
  - A source forge project.
- ***Comment***

  A project involves participants

- ***Galaxy Usefulness***

## 2.22 REPOSITORY

- ***Definition(s) from literature***

**[Lambda]** (Model repository): Electronic archive to save and retrieve models. Some kind of model repositories can be more or less restrictive according to the nature of model (e.g.. imply that the model they store conforms to a given metamodel, or to a given meta-metamodel, etc...)

- ***Galaxy definition***

A repository is a reference area used to by the developers to publish the new version of the artefacts they have modified or to update their local version according to a given already published one.

- ***See also***

- ***Example(s)***

  SVN repository.

- ***Comment***

- ***Galaxy Usefulness***

The concept of repository is commonly used to synchronize the works of several developers in industrial development projects.

## 2.23 SCALABILITY

- ***Definition(s) from literature***

**[TechTarget]**: the ability of an application or product (hardware or software) to continue to function well when it (or its context) is changed in size or volume in order to meet a user need.

**[Wikipedia]*:*** A property of a system, a network, or a process, which indicates its ability to either handle growing amounts of work in a graceful manner or to be readily enlarged.

**[LINFO]** Scalable refers to the situation in which the throughput changes roughly in proportion to the change in the number of units of or size of the inputs. It can also be looked at as the cost per unit of output remaining relatively constant with proportional changes in the number of units of or size of the inputs. Scalability refers to the extent to which some system, component or process is scalable.

- ***Galaxy definition***

The aptitude of a system (software) to work correctly when its size is changed (the number of artefacts, volume of data, number of users, etc).

- ***See also***

- ***Example(s)***

  - Scalability of databases: ability to grow to very large size while supporting an ever-increasing rate of transactions per second.
  - A routing protocol is considered scalable with respect to network size, if the size of the necessary routing table on each node grows as O(log N), where N is the number of nodes in the network.

- ***Comment***

- ***Galaxy Usefulness***

  The scalability of the Galaxy software is one of the main objective to be achieved by the project.

## 2.24 SYSTEM

- ***Definition(s) from literature***

**[WordNet]**

- Instrumentality that combines interrelated interacting artifacts designed to work as a coherent entity
- a procedure or process for obtaining an objective. an ordered manner; orderliness by virtue of being methodical and well organized

**[Techtarget]** A system is a collection of elements or components that are organized for a common purpose.

**[Wikipedia]** Systems engineering is an interdisciplinary field of engineering that focuses on how complex engineering projects should be designed and managed. Systems engineering deals with work-processes and tools to handle such projects, and it overlaps with both technical and human-centered disciplines such as control engineering and project management.

**[TOPCASED]** Set of end products and enabling products (See EIA-632).

- ***Galaxy definition***

Set of end products and enabling products (See EIA-632).

- ***See also***

  - See EIA-632

- ***Example(s)***

- ***Comment***

- ***Galaxy Usefulness***

## 2.25 SYSTEM DEVELOPMENT PROCESS

- ***Definition(s) from literature***

**[Humphrey 89]** (A software engineering process is) The total set of software engineering activities needed to transform a user's requirements into software.

**[IEE 90]** (The software development process is) the process by which user needs are translated into a software product. The process involves translating user needs into software requirements, transforming the software requirements into design, implementing the design in code, testing the code, and sometimes, installing and checking out the software for operational use. Note: These activities may overlap or be performed iteratively.

**[Fuggetta 00]** (A software process is) a coherent set of policies, organizational structures, technologies, procedures and artifacts that are needed to conceive, develop, deploy, and maintain a software product.

**[Sommerville 07]** (A software process is) the set of activities and associated results that produce a software product.

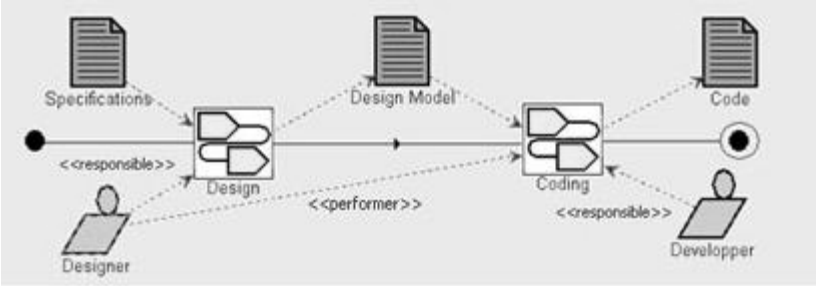
- ***Galaxy definition***

A system development process is the set of sequential or parallel engineering activities, with their associated results, guidance, and resources (human and tools) needed to produce a system.

- ***See also***

- ***Example(s)***

An example of a development process is shown by the figure below. It is a very simplistic process for developing small-size software. The process consists of two sequential activities: Design which produces a design model from specifications, and Coding which produces software code that implements the design model. The process involves roles (human resources) of designer, and developer. Specific guidance might be added depending on particular contexts. For example, Java Programming Guidelines might be associated with the activity Coding when Java is used.




- ***Comment***

Depending on how large and complex the development process is, multiple views on the same process might be envisaged. Typical examples of views are:

- The activity view, also called the workflow view, which focuses on the types, structure and properties of the activities in the process and their relations, sequencing. This view may be used for instance for scheduling purposes and monitoring.

- The artefact view which focuses on the types, structure and interdependencies of the artefacts produced or used within the process. This view may be used to see the transformation perspective of the artefacts form one state to another. It may be of interest also to highlight impact dependencies between artefacts.

- The resource view which describes the resources needed by the process. This view may be relevant from a resource management perspective.

Notice, that one view can refer to some elements defined in other views. One consideration then to take into account is to ensure the global consistency of the different views of the process.

- ***Galaxy Usefulness***

The elicitation of the development process is useful for management and control of collaborative MDE development of complex system. It will also constitute a basis for offering efficient assistance to developers.

## 2.26 TESTBED

- ***Definition(s) from literature***

**[Wordnet]:** A place equipped with instruments for testing (e.g. engines or machinery or computer programs etc.) under working conditions

**[Wikipedia]:** A platform for experimentation of large development projects. Testbeds allow for rigorous, transparent, and replicable testing of scientific theories, computational tools, and new technologies.

- ***Galaxy definition***

- ***See also***

- ***Example(s)***

  - For testing and education purpose, a test network (or testbed) was configured and connected to RENATER network.
  - PlanetLab testbed (www.planet-lab.org): One of PlanetLab's main purposes is to serve as a testbed for overlay networks. Research groups are able to request a PlanetLab slice in which they can experiment with a variety of planetary-scale services, including file sharing and network-embedded storage, content distribution networks, routing and multicast overlays, QoS overlays, scalable object location, scalable event propagation, anomaly detection mechanisms, and network measurement tools

- ***Comment***

- ***Galaxy Usefulness***

In order to validate Galaxy concepts, WP5 proposes to set up a test-bed environment, (a platform to conduct different types of tests: functional, performance, etc) which has a twofold purpose:

- to be able to run in realistic conditions the prototypes realized in WP4

- to evaluate these prototypes

## 2.27 TRACEABILITY

- ***Definition(s) from literature***

**[Lambda]** 1. Organization, recording and usage of chaining links between miscellaneous development or maintenance artifacts. 2. Traceability of a need: ability to retrieve the need which leads to create an artefact (upstream traceability) or the way a specification is realized (downstream traceability)

- ***Galaxy definition***

Same as above

- ***See also***

- ***Example(s)***

- ***Comment***

Traceability is an important concept in System and Software Engineering which deals with the semantic links between the requirements and some elements of the solution definition.

- ***Galaxy Usefulness***

The need to insure the traceability all along the development process has to be taken into account. Especially it implies that the consistency between the versions of two information items cannot be restricted to syntactical aspects.

## 2.28 VERSION

- ***Definition(s) from literature***

**[Lambda]** One state in the history of the evolution of an artefact.

**[TOPCASED]** Mean to identify a modification of a component which is selected for configuration management purposes

**[LearnThat]** One of a sequence of copies of a program, each incorporating new modifications.

**[WIkipedia]** A given state of the evolution of a software.

- ***Galaxy definition***

Same as Lambda's definition

- ***See also***

- ***Example(s)***

- ***Comment***

- ***Galaxy Usefulness***

A version identifies a consistent set of information gathered in an artifact that can be published or retrieved for the history of that artifact.

## 2.29 VIEW

- ***Definition(s) from literature***

**[Finkelstein]** A view is a partial or incomplete description of a system that fits responsibilities or roles assigned to a system's stakeholder. It represents an instance of a viewpoint applied to a system.

**[VUML]** At execution time, a view represents an active viewpoint on a system.

**[MOVIDA]** Views are the entities that belong to the universe of discourse world and that describe a system.

**[IEEE00]** A representation of a whole system from the perspective of a related set of concerns.

- ***Galaxy definition***

A methodologically defined partial model fragment showing the model elements, references and attributes relevant to a single concern.

- ***See also***

Viewpoint

- ***Example(s)***

The development of an avionics product involves several disciplines from several domains. Each of them uses its own set of views. All those views conform to viewpoints that can be specific to a domain or a discipline but they are related since they have the same product for subject. The thermician's view on an electronic system provides him an access to this system, according to thermal aspects of the system (the thermal perspective).

- ***Comment***

- ***Galaxy Usefulness***

Galaxy focuses on the concern of defining model fragments of intermediary grains between, on the one hand, the macro-grain of the whole project megamodel that integrates all the software project models, and, on the other hand, the micro-grain of views and diagrams individual model elements. Views are intended to be used in order to overcome the scalability problem.

## 2.30 VIEWPOINT

- ***Definition(s) from literature***

**[Finkelstein]** The combination of the agent and the view that the agent holds.

**[VUML]** Perspective of one stakeholder or category of user on the system.

**[MOVIDA]** The viewpoints specify the different views that have to be elaborated to completely describe the system.

**[IEEE00]** A specification of the conventions for constructing and using a view. A pattern or template from which to develop individual views by establishing the purpose and the audience for a view and the technique for its creation and analysis.

- ***Galaxy definition***

A stakeholder's vision of a system. It defines a set of constraints which specify which model elements, and model element references are allowed to appear in a stakeholder's view.

- ***See also***

View

- ***Example(s)***

A thermician, expert on a satellite for example, has his own vision of this complex system, with the ad-hoc vocabulary and concepts, his own models, data models, thermal models, dynamic or static models.

- ***Comment***

- ***Galaxy Usefulness***

The development of complex systems requires the involvement of several disciplines like: design engineering, dependability, thermal dissipation, electromagnetic compatibility (EMC), etc... Each of these disciplines defines a viewpoint or a set of viewpoints that specifies the concepts used and then the model artifacts which represent them. Independently of the viewpoint they belong to, all those model artifacts represent one or more aspect of the same product. In that sense they are all related together through more or less complex relationships.

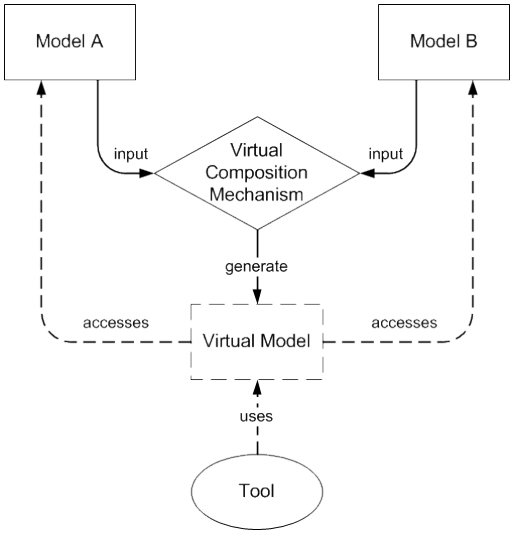
Galaxy aims to offer concepts for managing models in a collaborative way. Views and viewpoints will provide support for efficient fragmentations of models.

## 2.31 VIRTUAL MODEL

- ***Definition(s) from literature***

- ***Galaxy definition***

A virtual model is a model whose (virtual) model elements are proxies to elements contained in other models, i.e., a virtual model delegates the access to its elements to the models it references. The figure below shows how a virtual model delegates access to concrete models when its elements are accessed.




- ***See also***

**Erreur ! Source du renvoi introuvable.**

- ***Example(s)***

  Suppose there exist two models of a system: model A that represents the system component view and the model B that corresponds to the system class diagram. A virtual model C may be created to show how the classes are deployed in the components.

- ***Comment***

- ***Galaxy Usefulness***

  Virtual models are used as underlying mechanism for composing models, which may act as views.

## 2.32 WEAVING MODEL

- ***Definition(s) from literature***

  **[Del Fabro 08]** A weaving model captures the relationships (i.e., links) between model elements. A weaving model conforms to a weaving metamodel. The weaving metamodel defines the kinds of links that may be created.

- ***Galaxy definition***

  A weaving model is the resulting model of a model weaving operation.

- ***See also***

- ***Example(s)***

  Examples of weaving models are model transformation traces, matching models, aspects weaving models etc.

- ***Comment***

- ***Galaxy Usefulness***

  Weaving Models are used as storing mechanism for virtual links between the models, which make part of a view. They are interpreted by the a linking API and presented to the user according to the type of link (inheritance, references, merged elements etc..).

## 3. REFERENCES

[Bezivin 05] J. Bézivin, "Model Engineering: From Principles to Platform", WIT-Kolloquium, Technical University of Vienna, March, 15, 2005, http://wit.tuwien.ac.at/events/bezivin/index.html

[CMMI] Chrissis, M.B., Konrad, M. and Shrum, S. CMMI for Development®: Guidelines for Process Integration and Product Improvement (3rd Ed). Addison-Wesley. 2011.

[Del Fabro 08] Towards the efficient development of model transformations using model weaving and matching transformations. Software and Systems Modeling (July 2008) Special section paper. Springer, Berlin/Heidelberg. doi:10.1007/s10270-008-0094-z.

[Finkelstein] A. Finkelstein, J. Kramer, B. Nuseibeh, L. Finkelstein and M. Goedicke. Viewpoints: A framework for integrating multiple perspectives in system development. International Journal on Software Engineering and Knowledge Engineering, 2(1), 31-57, March 1992, World Scientific Publishing Company.

[Heutte] Jean Heutte. Apprentissage collaboratif : vers l'Intelligence Collective. http://jean.heutte.free.fr/article.php3?id_article=55

[Fuggetta 00] A. Fuggetta. "Software Process: A Roadmap". 22nd International Conference on Software Engineering (ICSE'2000), Future of Software Engineering Track, June 4–11, Limerick (Irlanda), ACM, 2000

[Humphrey 89] W.S. Humphrey. "The Software Engineering Process: Definition and Scope", in Proceedings of the 4th International Software Process Workshop on Representing and Enacting the Software Process, Devon, United Kingdom, 1989

[IEEE 00] IEEE Standard 1471-2000.

[IEEE 90] IEEE Standard Glossary of Software Engineering Terminology, IEEE St. 610.121990 (Revision and redesignation of ZEEE Std792-1933), September 1990. http://www.idi.ntnu.no/grupper/su/publ/ese/ieee-se-glossary-610.12-1990.pdf

[Kleppe] Kleppe, A., Warmer, J., Bast, W. The Model Driven Architecture: Practice and Promise, Addison-Wesley (2003)

[Koch 06] Koch, N.: Transformations Techniques in the Model-Driven Development Process of UWE. In: 6th International Conference on Web Engineering (ICWE), Volume 155 Article N° 3. ACM, California (2006)

[Lambda] F2-4 : Analyse des exigences pour le passage à l'échelle, v1.3, Lambda project, Y.Bernard, J.Bezivin, S.Madelenat, P.Tessier

[LearnThat] http://www.learnthat.com

[MDA] http://www.omg.org/docs/omg/03-06-01.pdf

[MOVIDA] http://movida.gforge.inria.fr/uploads/Main/D1.1.pdf

[OMG] http://www.omg.org/

[PMBOK] Project Management Institute A Guide to the Project Management Body of Knowledge (PMBOK® Guide) - Fourth Edition. 2008.

[Roschelle] Roschelle, J. and Teasley, S.D. The construction of shared knowledge in collaborative problem solving. Journal NATO ASI Series F, Computer and Systems Sciences, vol. 128, Springer Verlag, 1994.

[Sommerville 07] I.Sommerville, "Software Engineering 8", 2007, eighth edition, Addison-Wesley, ISBN 10:0-321-31379-8

[SysML] http://www.sysml.org/

[Techtarget] http://www.whatis.techtarget.com

[UML] www.omg.org/**spec/UML/2.3/**

[VUML] Anwar A. Formalisation par une approche IDM de la composition de modèles dans le profil VUML. Thèse de doctorat, Université de Toulouse-le-Mirail, décembre 2009

[Wikipedia] http://fr.wikipedia.org/wiki/Wiki

[Wiktionary] http://en.wiktionary.org/wiki/Wiktionary:Main_Page

[Wordnet] http://wordnetweb.princeton.edu/perl/webwn