Eclipse Modeling Platform

Creation Review
March 2006

Richard Gronback, Borland (co-lead)
Ed Merks, IBM (co-lead)

Agenda

- Proposal Overview
- Mission
- Scope
- Projects
  - Impact
  - Placeholders
- Timeline
- PMC

Proposal Overview

- Fundamentally, it is a reorganization of modeling-related projects
  - Currently, found in Tool and Technology projects
  - Goal is to improve communication and interoperability
- More importantly, it provides a vision and plan for comprehensive modeling capabilities within the Eclipse community
  - Described in Scope

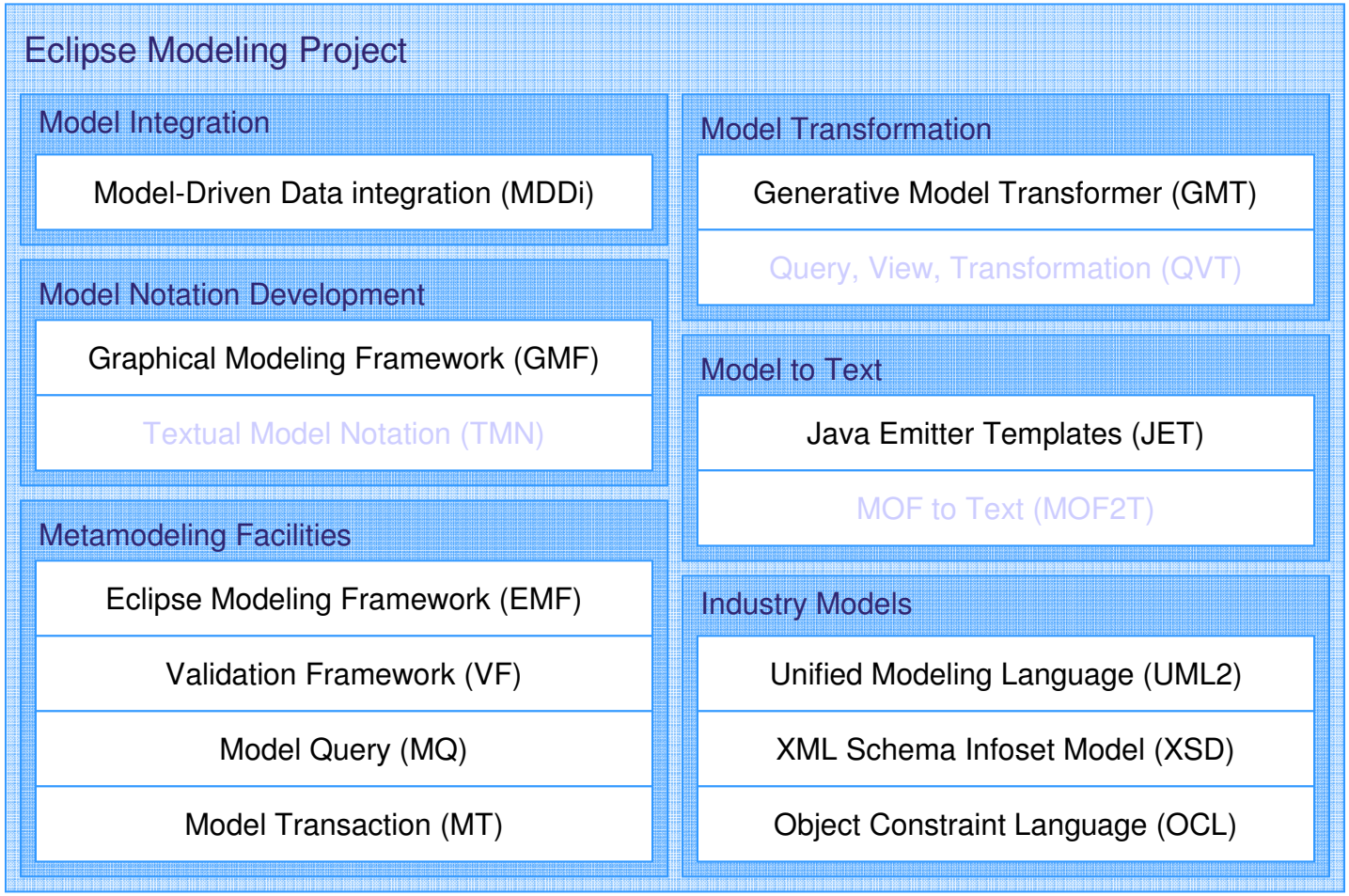
Mission

"The Eclipse Modeling Project will focus on the evolution and promotion of model-based development technologies within the Eclipse community. It will unite projects falling into this classification to bring holistic model-based development capabilities to the Eclipse."

Scope

- Details in Charter, but at a high-level:
  - Abstract Syntax Development
  - Concrete Syntax Development
  - Model Transformation
  - Model to Text Generation
  - Support for Industry Standards
  - Domain-Specific Modeling
  - *Out-of-scope items listed as well*

Modeling Projects



Projects in gray are placeholders

Impacts

- EMF
  - Moves from Tools
  - XSD and JET become independent projects
- EMFT
  - Moves from Technology, remains Incubator
  - OCL, VF, Query become independent projects
- GMF
  - Moves from Technology
- UML2
  - Moves from Tools
- OCL
  - Promoted to its own project (was part of EMFT)
- GMT
  - Moves from Technology, remains Research
- MDDi
  - Moves from Technology

Placeholder Projects

- As part of overall Modeling project vision, placeholder projects are specified for PMC recruitment:
  - QVT
    - Query, View, Transformation OMG standard
    - Part of MDA, used to specify model-to-model transformations
  - Textual Model Notation
    - For the generation of textual notation editors for DSLs
    - Generalization of popular Emfatic editor for ECore
  - MOF2T
    - MOF to Text
    - Currently, and RFP at the OMG

Timeline

- Creation Review
  - In progress
- Board Approval
  - Anticipate vote during EclipseCon (March)
- Provisioning
  - Following approval, anticipated early April
- Migration
  - Post Callisto release

PMC

- Groups
  - Requirements
  - Architecture
  - Planning
- Proposed PMC
  - Ed Merks (IBM) : co-lead
  - Richard Gronback (Borland): co-lead
  - Fred Plante (IBM)
  - Kenn Hussey (IBM)

Community

- Response to Proposal
  - Minimal interest, as anticipated given its nature
  - Interested Parties
    - SAP, Unisys, ILOG, etc.
- Consumers
  - Most of the existing projects enjoy a strong community today
  - EMF and related projects already have tremendous usage with the community and industry
- Contribution
  - To promote contribution, “Help Wanted” campaign to be initiated
  - Active recruitment for placeholder projects

Eclipse Modeling Project

Charter Appendix

Overview

- This charter was developed in accordance with the Eclipse Development Process and will outline the mission, scope, organization, and development process for the Eclipse Modeling Project. This document extends the Eclipse Standard Top-Level Charter v1.0, and includes the required content and overrides which follow. It is anticipated that as the standard charter is updated, this charter will incorporate the changes and make adjustments as seen fit by the PMC, and with approval from the EMO and board of directors.

Mission

- The Eclipse Modeling Project will focus on the evolution and promotion of model-based development technologies within the Eclipse community. It will unite projects falling into this classification to bring holistic model-based development capabilities to Eclipse.

Scope

- The items below delineate the scope of the Modeling project. They do not do so uniquely, as other roughly equivalent terms and concepts could have been chosen to define the domain of modeling in the context of software development.

Scope: Abstract Syntax Development

- Included in the scope of the project is a framework to support the definition of abstract syntax for modeling languages that support business, system, and software modeling, using an industry standard modeling facility or language. The framework for developing abstract syntax will support editing, validating, testing, querying, and refactoring models created with the modeling facility. This includes the production of general-purpose modeling languages in addition to application domain specific models.
- The underlying modeling facility must support the production of a wide range of models, not all of which will themselves be suited for inclusion within the Modeling project. Only those models and modeling languages of broad cross-domain utility fall within the scope of the Modeling project itself. For example, implementations of the Object Constraint Language (OCL) and Unified Modeling Language (UML) specifications are appropriately maintained within the Modeling project. In contrast, it is expected that users will create and maintain languages for narrowly defined problem domains using the project facilities, but that these languages and models will not be of interest to the general modeling community. A section below includes a more complete list of standard languages to be supported by the Modeling project.

Scope: Concrete Syntax Development

- Support for the production of textual and graphical concrete syntax for an abstract syntax is within the scope of the project. Both manual and generative approaches to the production of these are to be supported. As examples, graphical editing for the Unified Modeling Language (UML) as well as editing of UML models using textual notation fall within the scope of the project. Furthermore, the production of editors for any Domain-Specific Language (DSL) is to be supported.

- With respect to support of textual notation in the form of Eclipse editors, the Modeling project will focus on the generative aspect to producing these editors, targeting the facilities provided by the platform.

Scope: Model Transformation

- The transformation of models using a transformation definition and associated technologies falls within the scope of the project. The support of industry standards is expected in this area, specifically the OMG's Query, View, Transformation (QVT) specification.

Scope: Model to Text Generation

- Text generation from a model, typically source code of some programming language, including the merger of user changes to generated output, is within the scope of the project. Alternative mechanisms have been requested by the community and along with support for patterns, falls within the scope of the project.

Scope: Industry Standards

- The importance of supporting industry standards is critical to the success of the Modeling project, and to Eclipse in general. The role of the Modeling project in the support of industry standards is to enable their creation and maintenance within the Eclipse community. Furthermore, as standards bodies such as the OMG have a strong modeling focus, the Modeling project needs to facilitate communication and outreach through its PMC and project contributors to foster a good working relationship with external organizations.
- The following industry standards are either supported by current modeling projects, or are anticipated to be supported in the future:
  - Object Management Group (OMG) standards
    - Meta-Object Facility (MOF)
    - Unified Modeling Language (UML) and UML Profiles not falling within the scope of other projects
    - Model-Driven Architecture (MDA) related specifications
    - Query, View, Transformation (QVT)
    - MOF to Text (MOF2T)
    - Diagram Interchange Specification (DIS)
    - XML Metadata Interchange (XMI)
  - Business Process Modeling Notation (BPMN)
  - Business Process Definition Metamodel (BPDM)
  - XML Schema Infoset Model (XSD)
  - etc...

Scope: Domain Specific Modeling

- The support of industry standards and specifications are an important aspect to the scope of the Modeling project, but not to the exclusion of the emerging trend of Domain-Specific Languages (DSLs). The Eclipse Modeling Project will provide leadership in delivering these capabilities through its projects and in working with others within the Eclipse and external communities.

- The generative production of editors for textual notations is an essential component of DSL support within Eclipse, and required if Eclipse is to be used as a "language workbench." The Modeling project will provide, within its scope, the generative aspect of producing these editors to complement graphical editors for a modeled domain.

Out of Scope

- Clearly, there are some things that are **not** in the scope of the Modeling Project. In particular, there are certain DSLs and industry-based models that the Modeling Project should support creation of at a fundamental level, but that are not approporiate to be housed within the project. Specific examples are listed below:
  - A home for DSLs based on EMF which do not pertain to the modeling domain, e.g. the Java EMF Model (JEM) created for the Visual Editor (VE) project and used by the WebTools project.
  - UML Profiles falling within the scope of other projects.
  - A number of OMG standards that fall into the modeling realm are not considered to be within the scope of this project:
    - Software Process Engineering Metamodel (SPEM), as it is part of the proposed Beacon project.
    - Common Warehouse Metamodel (CWM), as it is likely to be included in DTP.
    - UML Testing Profile, as it is included in the TPTP project.
    - Reusable Asset Specification (RAS)
    - etc...
  - As mentioned above, in support of textual notation editor generation, the Modeling project will work with the Platform or other teams, and not itself focus on the development of general purpose text editor frameworks.
  - In the area of model transformation, XSL does not fall within the scope of the Modeling project and is provided within the context of the WebTools Project (WTP).

Project Management Committee

- The content found in this section of the standard charter is sufficient for the Modeling project, with the exception of those subsections found here related to the project's Requirements, Architecture, and Planning Groups.
  - **Requirements Group**
    - The Requirements Group is formed at the discretion of the PMC. The Requirements Group gathers requirements for the project and communicates them to all members of the Project, including the PMC. The Requirements Group will accomplish its objectives by working closely with the development teams and the PMC.
  - **Architecture Group**
    - The Architecture Group is formed at the discretion of the PMC. The Architecture Group is responsible for development, articulation and maintenance of the Project Architecture, as well as for providing an explicit description of the architecture and communicating this description to all members of the Project, and for releasing it as part of the project deliverables. The Architecture Group will accomplish its objectives by working closely with the development teams and the PMC.
  - **Planning Group**
    - The Planning Group is formed at the discretion of the PMC. The Planning Group assists the PMC in establishing the Project plan in conjunction with available Project resources, coordinating relationships between Project participants and with other Eclipse projects. The Planning Group helps to ensure that projects have enough contributors, filling vacancies in roles and facilitating code or other donations by individuals or companies. The Planning Group will accomplish its objectives by working closely with the development teams and the PMC.

Roles

- The content found in this section of the standard charter is sufficient for the Modeling project, with the exception of those items listed below.
  - **Committers**
    - It is the responsibility of the Committers to maintain the accuracy of the version numbers of their plug-ins in accordance with the published guidelines.

Projects

- The content found in this section of the standard charter is sufficient for the Modeling project. Below is a diagram [see previous slide] representing the projects found in the Modeling project, as well as those under recruitment by the PMC.

Infrastructure

- The content found in this section of the standard charter is sufficient for the Modeling project, with the exception of those items listed below.
- To the greatest extend possible, the PMC will facilitate the use of common infrastructure and process by all projects within the Modeling project, e.g. build, IP tracking, etc... This should include a continuous build mechanism, cascading builds, and common reporting infrastructure.

Development Process

- The content found in this section of the standard charter is sufficient for the Modeling project, with the exception of those items listed below.
  - **Tests**
    - It is expected that contributors to the project provide a test with each contribution. It will fall to the responsibility of the Committers to ensure that adequate tests are provided for each commit to the source repository.
  - **Release Cycle**
    - Each project will coordinate their release cycles and be respectful of their dependent projects, particularly in the context of a larger Eclipse coordinated release cycle. Projects are expected to publish their milestone plans.

Legal

- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.