

EMF Index – Creation Review

Contents

Introduction	1
Scope	1
Out Of Scope	2
Relations To Other Eclipse Projects	2
Community Involvement	3
Code Contribution	3
Mentors	3
Initial Participants	3
Roadmap	4
Copyright Statement	4

Introduction

The goal of EMF Index is to provide an index for modeling tools based on the Eclipse Modeling Framework (EMF). That index will automatically collect and synchronize index data on Ecore models and instances. It will allow fast queries without the need to have EMF resources instantiated in memory, thereby enabling scalable modeling tools based on EMF with a JDT-like user experience.

Comments should be made to the EMFT newsgroup:
<news://news.eclipse.org:119/eclipse.technology.emft>

Scope

EMF follows an open-world paradigm, i.e. it is possible to load every resource that is accessible, locally or from the network. While that is fine for on-demand loading of referenced elements, a search operation needs a confined scope. Clients must have a way define the scope of indexed elements, e.g. as the resources in their workspace.

- **Collection of index data**

Retrieve index data from a given EMF resource. The index data will consist of generic data, e.g. the type of an element, as well as model-specific data, e.g. a collection of attributes that constitute a fully-qualified name. Any model-specific data can be stored in the index. There will be a way to specify which data will be retrieved from the EMF model. This also includes the definition of model-specific scopes for queries.

- **Persistence**

Store the collected index data independently from the indexed resources in such a way that it can be queried easily later.

- **Synchronization of index data**

Whenever it is possible, detect model changes and update the model index automati-

11th of March 2009

cally.

- **Queries**

Queries should consist of a set of criteria and a query scope. The results of a query must be descriptors (e.g. EMF proxies or fragment URIs) of model elements rather than the elements themselves, such that the client decides whether to load them or not. The descriptors can carry any additional information on the element available in the index.

- **Exchangeable storage backend**

The backend to actually store the index data will be exchangeable, allowing customized implementations for specific storage technologies, such as pure in-memory or database storage of index data.

Out Of Scope

- **Non-EMF models**

EMF Index will not deal with models coming from other modeling frameworks.

- **Implement a model repository**

EMF Index will not store the model elements themselves, but only specific, pre-selected properties of them. Queries will be restricted to these properties.

Relations To Other Eclipse Projects

- **EMF**

EMF Index will be built on top of EMF, such that it will be usable by any EMF-based framework, not depending on the specific concrete syntax.

- **TMF/Xtext**

Xtext is expected to make strong use of EMF Index. As it describes models textually, a common problem is to resolve/find a model element by some domain-specific textual description. An index taking these descriptions into account will be a great benefit

The main distinction to other EMF based projects that provide searching and querying services is that the queries of EMF Index will be reduced to a subset of properties, thereby allowing to query elements without loading resources.

- **CDO**

CDO implements database persistence for EMF models offering powerful generic queries. Nevertheless, EMF Index will provide a more abstract interface to common queries needed in modeling tools. These can be mapped to the CDO queries by a specialized CDO storage backend for EMF Index.

- **EMF Search**

EMF Search provides a user interface for querying Ecore elements generically. Models have to be loaded to be accessible. EMF Index could be used to reduce the list of candidates to be instantiated, thereby improving scalability and performance.

- **EMF Query**

EMF Query's API is also too generic to be used for EMF Index.

Community Involvement

- In a discussion on scalable EMF tooling at Eclipse Summit Europe 2008 with representatives from several companies involved in modeling tools lead to the decision to start a new Eclipse Project for a model indexer.
- The proposal has had very positive feedback in the newsgroup, that also helped clarifying the scope of the proposed project. The proposal has been modified taking these comments into account.
- In a meeting with representatives from Eclipse CDO, SAP and itemis, the requirements for indexing have been confirmed and the relation to CDO has been clarified.

Code Contribution

EMF Index will have an initial contribution of a bunch of plug-ins currently hosted in the TMF/Xtext project. This should not raise any IP issues.

Mentors

- **Sven Efftinge** (itemis AG, Germany)
Project lead of Modeling/TMF, PMC member, Committer for M2T and EMFT
- **Ed Merks** (Macro Modeling, Canada)
PMC lead of Eclipse Modeling, Project lead of EMF

Initial Participants

- **Jan Köhnlein** (itemis AG, Germany) – Project Lead
Jan has earned several years of experience in the development of modeling tools and model-driven software development. He is a committer of TMF/Xtext, M2T and TMF. Before joining itemis, he worked as an architect for Gentleware AG. As product manager for Apollo for Eclipse, he designed a model index for storing ID to element mappings needed during reverse engineering Java code to UML.
- **Sven Efftinge** (itemis AG, Germany) – Committer
Sven works as consultant, coach and developer and leads a branch of itemis AG in northern germany. His focus is on domain-specific modeling, generator development and Eclipse technologies. Sven is the project lead of TMF Xtext, member of the Eclipse Modeling PMC and a committer to EMFT and M2T.
- **Dennis Hübner** (itemis AG, Germany) – Committer, Release Engineer
Dennis is an expert on model driven technologies and Eclipse plug-in development. He is actively developing TMF/Xtext, M2T and MWE and has also set up the continuous build and testing infrastructure for these projects.
- **Bernd Kolb** (Independent, Germany) – Committer
Bernd focuses on model-driven software development and Eclipse technologies. He is a

11th of March 2009

committer at the Eclipse Modeling Project (TMF-Xtext, M2T-Xpand and EMF-MWE) as well as on openArchitectureWare and a regular speaker at various conferences.

- **Jos Warmer** (Ordina, Netherlands) – Committer

Jos Warmer is project lead of the Mo4j open source project

(<http://www.mod4j.org/projectsite>) which uses the Eclipse Modeling Project technology extensively. As part of Mod4j Jos has developed CrossX, a technology that allows different EMF models for different Ecore metamodels to refer to each others model elements while remaining loosely coupled. The concepts in CrossX are similar to the concepts in the EMF Index proposal.

Roadmap

- First milestone release of EMF Index is planned synchronously with the Eclipse Galileo release, i.e. late June 2009. This initial pre-release will present the high-level architecture and feature an in-memory implementation of the main services.
- It is planned to join the release train of the Eclipse 3.6 simultaneous release. As EMF Index depends on EMF, we target a +2 slip.

Copyright Statement

Copyright 2009 by itemis AG (www.itemis.de).

This document is made available under the terms of the Eclipse Public License v1.0

(<http://www.eclipse.org/legal/epl-v10.html>).