D1.7.1-1 Report: Annual Product Owner Review

ModelWriter

Text & Model-Synchronized Document Engineering Platform

Work Package: WP1
Task: T1.7 - Annual Product Review

Edited by:

Anne MONCEAUX <anne.monceaux@airbus.com> (AIRBUS GROUP)
Ferhat Erata
Moharram Challenger
Etienne Juliot

Date: 31-08-2016
Version: 1.0.0

Document History

Version	Author(s)	Date	Remarks
0.1.0	Ferhat Erata Moharram Challenger	30-Apr-2015	Draft
1.0.0	Anne Monceaux	<date>	Initial Release

Table of Contents

1. Introduction

Role of the deliverable

This document is the first version of the ModelWriter product review due at M15. The document is the output of the task 1.7 Annual Product Review, coordinated by AIRBUS and all Country Coordinators, which constitute altogether the "Product Review Committee". It will be up-dated yearly.

Since the project doesn't produce a real product at M15, the first annual product review task has had objectives very similar to those of the ITEA first annual review and the French DGE first annual review. Thus we refer to the documents realized for these reviews. They are accessible at the following address: https://github.com/ModelWriter/WP5/tree/master/Reviews/2015

Structure of the document

This document is organized as follows:

- Chapter 1 introduces the document.
- Chapter 2 is a short synthesis of the review.
- Chapter 3 is a description of the technical progress at M15.

Terms, abbreviations and definitions

Abbreviation	Definition
WP	Work Package
UC	Use Case

2. Review synthesis

At milestone M15, the technical integration activities are just starting. Therefore, there is no real ModelWriter product available yet. Instead, we have several prototype software that can be discussed and partially validated using the use cases data and requirements.

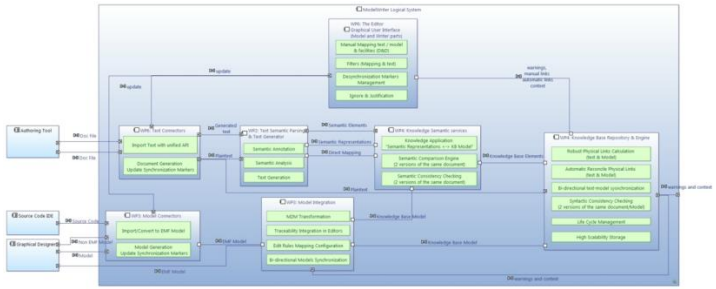
The reasons for the late start of integration are explained in the project review minutes. Partners in this project have many different backgrounds and competencies: ontology and natural language processing, Microsoft technologies, Eclipse Modeling technologies, etc. Thus, a large part of the first months of the project was used to agree and define the complete solution architecture, expected capabilities (based on the understanding of various uses cases) and skill improvement from each engineer to better understand their peers' expertise. Second, the exit of the Belgian partners has had both management and technical consequences we had to deal with from the beginning: early change request, delayed starting of the development tasks, scope of WP2 impacted (technical efforts regarding text parsing has been passed over the LORIA with no increase of budget, and consequently Natural language parsing and generation techniques have less time left). Concrete development of first prototypes started beginning of summer 2015.

All partners agree that the product integration must be achieved in the next period and this require a sustained long term effort.

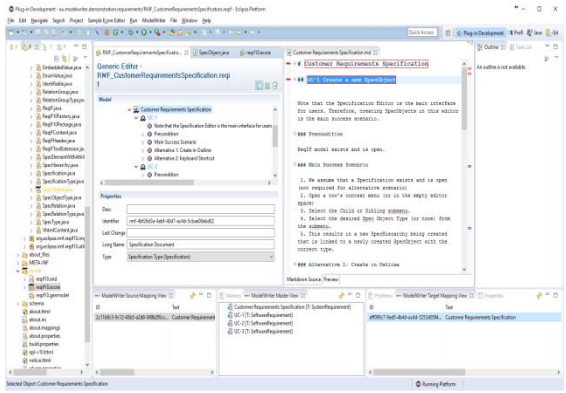
The project benefits from efficient collaboration and steering tools that allow interaction between developers on a daily basis and a very good cooperation on technical issues, including the sharing of Open Source code.

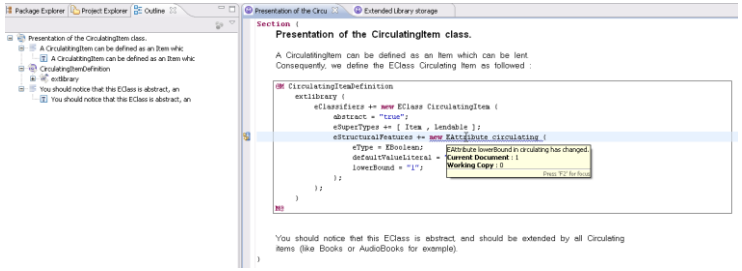
3. Main technical progress at M15 - Ongoing prototypes

Considering the status at M15 we list in this section the several available pieces of software.

Contributors	OTP	Achievement	Comment
All	Requirements	Version 1 achieved	User requirements and software requirements have been defined and validated during project meetings. They are shared and managed through the GitHub platform in order to provide access to up-to date data to all partners.
All	Technical Architecture Design	architecture v1 defined L'objectif principal de ce sous-projet est d'assembler et d'évaluer les versions annuelles successives du ModelWriter intégré. 3 composants sont en cours de développement : Annotation sémantique, Knowledge Base, UI. Ils ont permis de valider d'une manière indépendante l'approche de chaque composant. Il faut maintenant faire une intégration pour poursuivre l'expérimentation. Le plan d'implémentation suit les principes d'architecture définis dans la tâche 6.2, avec comme priorité les composants qui forment la chaine minimale d'intégration et la fourniture des API pour brancher les extensions des partenaires du WP2 et WP1.	The global architecture is defined and the ongoing development are kept aligned with it.  Light blue: systems outside of ModelWriter. Dark blue: ModelWriter logical component scope Green: concrete capabilities.
HISBIM	QDMS system integration via plug-in	ModelWriter system communicates with QDMS by XML files	
LORIA; Airbus		Automatic Text Annotation: Create links automatically; annotates text using terminology/ontology concepts labels	Use in combination with a structured "knowledge base" (this knowledge base is different from the ModelWriter synchronization KB; here it refers to a populated ontology)
LORIA	Create links automatically	Automatic Text annotation: annotate text using model element labels	Used for raw extracts from text or from EMF models (Flat structures)

LORIA	Text parsing	Semantic analysis: translate text into RDF	Translate a text in natural language into RDF triples. Used in combination with a knowledge base on Airbus data; enable easier access/processing of textual elements
LORIA	Text generation	NOT STARTED at M15	
OBEO	ModelWriter knowledge base	ModelWriter knowledge base: store and manage synchronization links 1st prototype: ongoing development. Three components of the MW_KB are being developed. One handle the storage of links. The secnd one handle the storage of the data used for semantic analysis. The KB is fed with the annotation tools provided by the WP2 ; or it can be fed with manual annotations performed by the user. Selection of a reliable library for document parsing is a functionality to be implemented in the 1st scenario. The last function is not yet implemented at M15. The User Interface is the last component. Sketches have been proposed. This will be developed together with the integration tasks autumn 2015.	Used formalisms: - Links: EMF (storage), available API to ease evolution - Knowledge: API for inputs in EMF, OWL, RDF
OBEO	UI	Ces travaux n'ont pas commencé mais sont prévus dans le prochain jalon. Une première intégration des 3 composants en cours de développement est prévue d'ici fin 2015. Un workshop dédié est prévu début novembre pour lancer cette tâche.	Le développement est en cours. Un mockup a été réalisé pour valider l'ergonomie et le développement de l'interface correspondance a été réalisé. Cette interface est considérée comme un prototype car les comportements des boutons ne sont pas réellement fonctionnels et sont simulés. L'intégration de ce composant avec le moteur est prévue pour démarrer fin 2015. **Voici une capture d'écran du résultat du prototype :** Certains composants disposent déjà de tests unitaires : le composant de gestion des liens et la partie stockage sémantique. Il faut étendre cette pratique à tous les composants. Des outils ont été sélectionnés pour garantir l'intégration continue, l'exécution des tests unitaires et d'intégration. Ils

			seront déployés courant 2016. Pour les tests fonctionnels, l'outil RCPTT a été pré-sélectionné. L'écriture des scenarios fonctionnels est prévue pour démarrer en 2016. Cette tâche a du retard au niveau européen car les partenaires turques qui devaient travailler dessus ont demandé à avoir l'ensemble des livrables des uses cases en amont. 
OBEO	Connect to external text editors	Connector to external editor: ongoing development (1 component). Initial integration planned end 2015.	Integration between Intent and EMF Compare engine. Use the Intent document meta model as a formalism usable by EMF Compare, and provide integration into the edition tools. This will allow using a posteriori text edited with MSWord or other edition tools, without being intrusive. Au montage du projet ModelWriter, nous pensions nous appuyer sur le moteur existant du projet Intent réalisé par Obeo. Dès le début du projet, nous avons fait le constat que 80% du code du projet Intent concernait la partie édition de texte. Or, cette partie est complètement écartée du projet car nous préférons laisser à l'utilisateur le choix de rester dans son outil habituel de rédaction de documentation. Cette complexité était donc devenue inutile et freinait le développement du cœur du vrai objectif

			du projet. Obeo a ainsi fait le choix de recréer un moteur, en reprenant certaines parties existantes, et en ne garantissant aucune compatibilité ascendante. Par soucis de simplicité vis-à-vis des process de la fondation Eclipse, nous avons choisi de faire ce travail au sein du même projet Intent. L'impact sur le périmètre de l'annexe technique est nul, et permet même d'avoir un projet Intent totalement aligné sur les objectifs de ModelWriter. L'impact le plus important est sur le planning car il a fallu reposer les bases du nouveau moteur. 
UNIT	Connect to external text editors	Connector to external editor: parsing of .docx files	
UNIT	Create links manually		
UNIT	Manage links	Markers and their classification	
UNIT	Manage links	Mapping of the markers: The first version of Mapping is realized with regarding type of the relation for that mapping.	
UNIT	Model transformation	Model Transformation towards and from MW_KnowledgeBase	