INL/EXT-21-65503| NRIC-22-ENG-0002

Summary of Work on a Software Adapter for Project Requirements and MBSE

Brennan Harris
Full Stack Developer
B711 Model-based Design

NRIC | INL Idaho National Laboratory

INL is a U.S. Department of Energy National Laboratory operated by Battelle Energy Alliance, LLC

DISCLAIMER

REVISION LOG

Revision No.	Date	Affected Pages	Description
0	12/06/21	All	New Document
1	12/10/21	All	Responding to reviewer edits
2	12/10/21	Title	Accepting new title changes

Page intentionally left blank

CONTENTS

FIGURES

Page intentionally left blank

ACRONYMS

API	Application Programming Interface
DOORS	Dynamic Object Oriented Requirements System
IDEAS	Ion Digital Engineering Application Suite
MBSE	Model-based system engineering
NRIC	National Reactor Innovation Center

Page intentionally left blank

Summary of Work on a Software Adapter for Project Requirements and MBSE

1. ABSTRACT

The National Reactor Innovation Center (NRIC) has funded the development of a digital pipeline to pull project requirements from IBM Rational Dynamic Object Oriented Requirements System (DOORS), save them to Deep Lynx, and then push them into the corresponding project in Innoslate. These utilities are accessible from a web browser and allow a project's functional and business requirements to be tied to its tree of model-based systems engineering (MBSE) data.

2. SIGNIFICANCE

Well-formed requirements are key to successfully carrying out a large project. Project requirements ensure that (1) key stakeholder interests in a project can be satisfied by the given design, (2) the resources (e.g., financing, material, time, and labor) are sufficient to accomplish the project goals, and (3) changes to an initial design can be carried out with greater efficiency, thanks to enabling the assessment of the changes' impacts to all the requirements.

DOORS is an industry standard for requirements management. The requirements pipeline was created with DOORS as the data source, since many NRIC stakeholders already use it in their project management activities. Digital requirements management tools such as DOORS provide four key advantages over traditional document-based solutions—(1) *live editing*: team members who are defining, updating, and consulting requirements are granted live access to the latest versions; (2) *reusability*: any requirements that apply in multiple instances across a project can be defined once and then referenced every time they appear, instead of being copied over and over again throughout the documents, with any changes automatically being propagated to each reference; (3) *traceability*: the history of changes made to the requirements is saved, along with the identity of the manager(s) or engineer(s) who made them; and (4) *uniformity*: requirements formatting and standards can be automatically enforced, and requirements can be searched and sorted by risk, category, and status.

Many NRIC stakeholders are also applying the model-based approach to the development and management of their projects. MBSE is a methodology that goes beyond just requirements and "is used to support the…design, analysis, verification, and validation associated with the development of complex systems. In contrast to document-centric engineering, MBSE puts models at the center of system design."[1] Innoslate is the MBSE software package used in this pipeline.

A method of linking the corpus of requirements data to the structured tree of MBSE data without replicating data entry is a capability that expands the benefits of both requirements management and MBSE. Requirements centering on modular equipment, for example, can be enhanced via direct links to the models that represent those assets.

3. FUNCTIONAL DESCRIPTION

The application's architectural base is its connection to the Deep Lynx data lake. Deep Lynx is an open-source project developed primarily in the Digital Engineering department (B710) of the Energy and Environment Science and Technology organization. It provides a standard set of utilities to import and export structured data from a common ontology called DIAMOND. This standardized ontology also allows developers to create applications that can read, consume, and manipulate data from many sources, provided it has been adapted to the data lake's ontology.

An adapter has been developed for extracting requirements and their associated metadata from DOORS projects. Project managers with access to DOORS can create and edit requirements there, using the standard DOORS workflow. The DOORS adapter is configured in the Deep Lynx admin panel to pull the latest requirements at a set interval (any interval is permitted, but 10 minutes is set as the default). The adapter configuration requires the user to supply the project name and valid DOORS user credentials in order to target and access the correct requirements. On the first ingestion, records of each requirement will be written to the Deep Lynx database, prompting the user to map the requirements data and metadata. At subsequent intervals, any change in a requirement or its metadata will cause an automatic update to the corresponding data in Deep Lynx. As an innate Deep Lynx data source, the DOORS adapter is configured, managed, and scheduled using the Deep Lynx administrative interface.

A paired adapter was also developed to push the requirements from Deep Lynx into Innoslate. The Innoslate adapter is an application programming interface (API) that targets both an Innoslate project and a specific Deep Lynx container. The "pushing" endpoint requires four parameters in order to extract the requirements from a Deep Lynx container and push them into an Innoslate project: (1) Deep Lynx credentials, which can be generated from Deep Lynx in the form of a key/secret pair of strings; (2) a Deep Lynx container reference, which can also be extracted from Deep Lynx as an integer identifier; (3) Innoslate credentials, which can be generated from Innoslate in the form of a single authentication string; and (4) an Innoslate project reference, comprised of an organization string and a project identifying integer. The Innoslate adapter also provides some supplementary endpoints for extracting the Innoslate organization and project identifiers in an appropriate format for use at the operative endpoint. The Innoslate adapter is a software utility that is independent from Deep Lynx and can be accessed via the Ion Digital Engineering Application Suite (IDEAS).[2]

When a web request is launched to the Innoslate requirements endpoint, it first uses the Deep Lynx key/secret pair to obtain a temporary access token from Deep Lynx, then uses the provided token and the container identifier to access all the database entries labeled "requirement." Each requirement in the list is transformed so as to comply with the required format for publishing to Innoslate. For each requirement, a local and global identifier is allocated and attached. The Innoslate token, project destination, and transformed list of requirements are then pushed to the Innoslate API endpoint and saved there as MBSE elements.

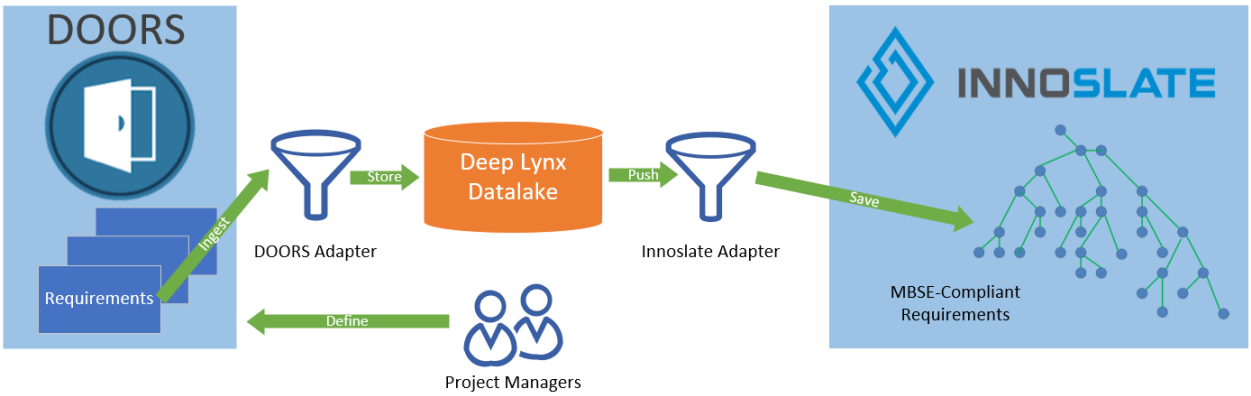

Figure 1. Functional diagram and data flow of the requirements pipeline.

4. SUMMARY

This software pipeline demonstrates the capability to take requirements defined in IBM Rational DOORS and automatically register them to an Innoslate project's MBSE data tree. This allows direct application of requirements management insights to the model-based design and analysis of complex projects, without duplicating any efforts made in the requirements drafting and editing phase. It also allows requirements managers to engage with technology and processes already familiar to them, and no type of action or curation is required on their part before the data are sent to an MBSE environment.

5. REFERENCES

1. Shevchenko, N., 2020. An Introduction to Model-Based Systems Engineering (MBSE). [online] SEI Blog. Available at: https://insights.sei.cmu.edu/blog/introduction-model-based-systems-engineering-mbse/ [Last accessed 30 November 2021].
2. *Ion Digital Engineering Application Suite*, Idaho National Laboratory, September 2021. https://denric.azureacc.inl.gov/.