The Development of a Modelling Language for Rich Internet Applications

Jevon M. Wright, Jens B. Dietrich

School of Engineering and Advanced Technology
Massey University, Palmerston North, New Zealand
{j.m.wright,j.b.dietrich}@massey.ac.nz

MASSEY UNIVERSITY

Problem

Rich Internet Applications (RIAs) are a recent innovation that improve the usability and performance of web applications. This is enabled through AJAX, offline toolkits, and other new technologies.

Existing modelling languages for web applications (WebML, UWE) cannot model RIAs. These languages lack support for modelling the fundamental requirements of RIAs – such as events, lifecycles, users, and security – and are often not platform independent nor supported by CASE tools, hindering their evaluation by academia and industry [1].

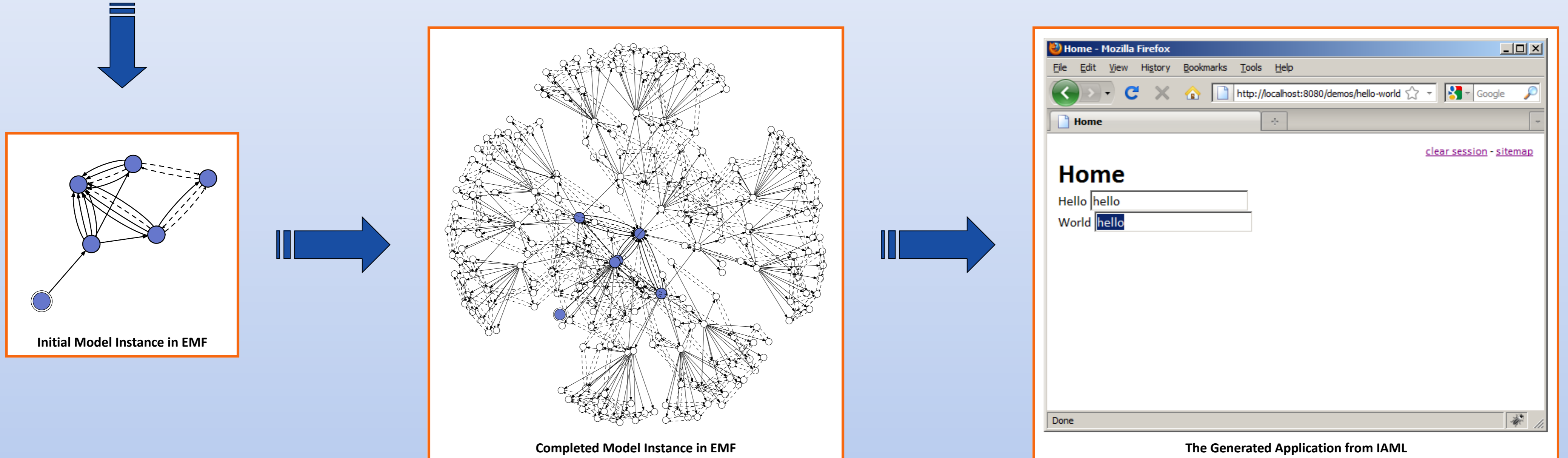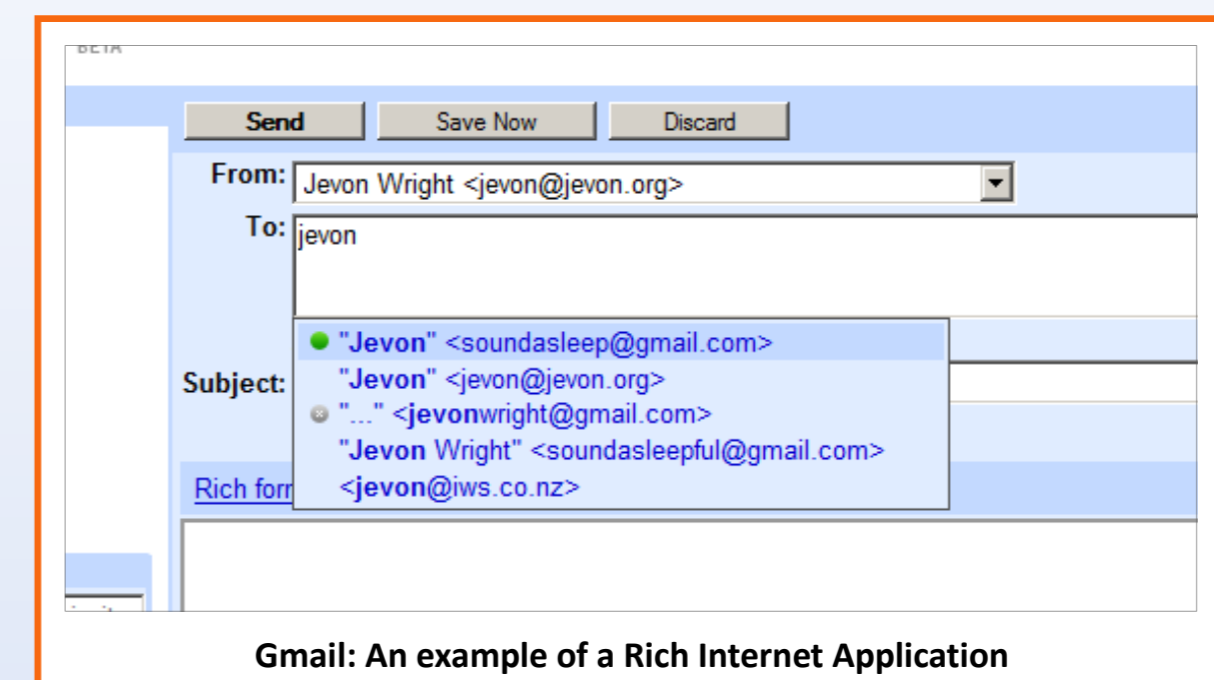**The Eclipse Plugin for the Internet Application Modelling Language**
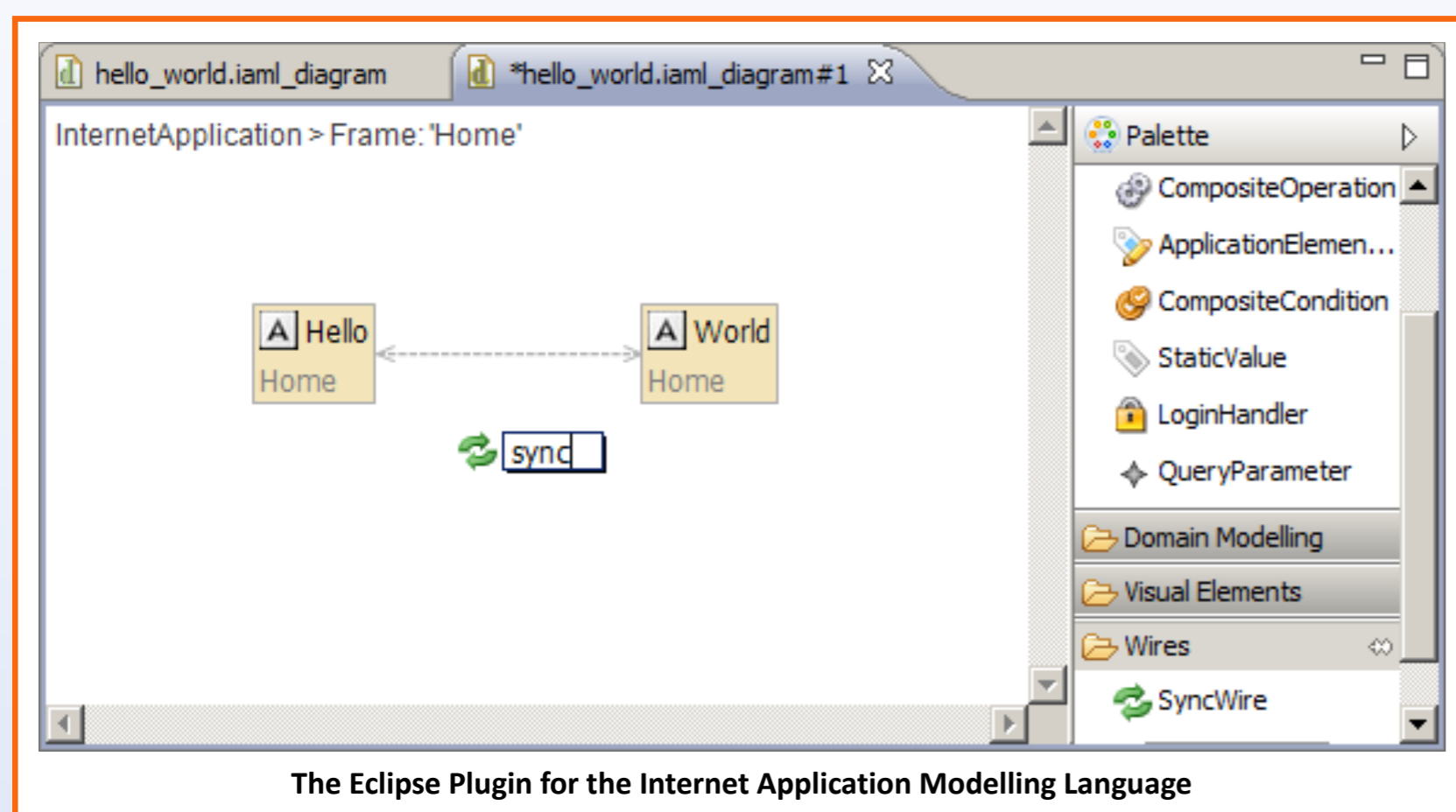
Solution

The development of a new language, called the *Internet Application Modelling Language* (IAML), aims to provide modelling support for all of the fundamental concepts of RIAs.

Along with operations and domain objects, we propose to model events and conditions as first-class citizens; likewise, we also promote users and security as first-class.

After reviewing and evaluating existing languages, we decided to develop a new language for modelling RIAs, but use concepts from existing languages where appropriate – such as ECA rules, ER diagrams, and UML Activity and Class diagrams.

Gmail: An example of a Rich Internet Application

Initial Model Instance in EMF

Completed Model Instance in EMF

The Generated Application from IAML

Implementation

IAML is implemented in a proof-of-concept Eclipse plugin, using the following modelling technologies:

- Eclipse Modeling Framework (EMF) for meta-modelling.
- Graphical Modeling Framework (GMF) for a graphical editor.
- JBoss Rules (Drools) for model completion [2].
- openArchitectureWare (OAW) for code generation.

This plugin provides a rich model-driven development environment, transforming model instances made by a model developer, into web applications of PHP, Javascript, HTML, CSS and SQLite.

Future Work

There are many unanswered questions to address with further research. For example:

- Completing the IAML modelling language to implement the *Ticket 2.0* benchmarking application [3].
- Using model checkers such as Crocopat or NuSMV to verify complex properties of models, such as detecting infinite loops.
- Providing an authoritative and concise documentation source.
- Extending the existing code generation templates to generate JSP, Python, Ruby or .Net applications.
- The unification of existing scripting languages (PHP, Javascript, ...).

References

[1]: J. Wright and J. Dietrich, "Survey of Existing Languages to Model Interactive Web Applications," in Proceedings of the 5th Asia-Pacific Conference on Conceptual Modelling (APCCM 2008), Wollongong, NSW, Australia, 2008.
[2]: J. Wright and J. Dietrich, "Non-Monotonic Model Completion in Web Engineering," in Proceedings of the 21st Australian Software Engineering Conference (ASWEC 2010), Auckland, New Zealand, 2010.
[3]: J. Wright and J. Dietrich, "Requirements for Rich Internet Application Design Methodologies," in Proceedings of the 9th International Conference on Web Information Systems Engineering (WISE 2008), Auckland, New Zealand, 2008.

The IAML proof-of-concept Eclipse plugin can be downloaded from: **http://openiaml.org**