WEB RATIO®
You think You get

Interaction Flow Modeling Language (IFML) Request For Proposal status

Marco Brambilla
marco.brambilla@webratio.com

Ad/11-09-11
OMG Technical Meeting, ADTF, September 21, 2011

Status - Recap

- WebML: Web Modeling Language
  - Presented at DC meeting, March 2011
- IFML: intention for RFP
  - Discussed at SLC meeting, June 2011
- IFML RFP: Interaction Flow Modeling Language
  - Mailing list: ifml@omg.org
  - Several versions since June
  - Comments received and addressed
  - Reviews from AB members
  - Presented last Monday to AB
  - Now completely revised for
  - Current document number for the RFP: ad/11-09-07

Objectives

Expressing

- **content**
- **user interaction**
- **control behavior**

of the front-end of applications belonging to the following domains:

- Traditional, HTML+HTTP based Web applications.
- Rich Internet Applications and HTML 5 apps.
- Mobile applications.
- Client-server applications.
- Desktop applications.
- Embedded Human Machine Interfaces for control applications.
- Multichannel and context-aware applications.

Advantages

- permit the formal specification of the different perspectives of the front-end
- isolate the specification of the front-end from its implementation-specific issues.
- improve the development process, by fostering the separation of concerns in the user interaction design
- enable the communication of interaction design to non-technical stakeholders, permitting validation of requirements complement existing practices that use PIM and PSM separation for other architectural layers
- automatic generation of code also for the application front-end part, increasing the level of automation of the process

State of the Art

- **WYSIWYG form editors**
- **Interface specification languages**
  - UIML, XUL and some existing profiles for Uis (WAE)
- **Multimedia presentation languages**
  - SMIL, VOICP
- **Web engineering methodologies, notations and tools**
  - WebML, OO-HDM, WAE, WebDSL, Hera, UWE, …
- **Research projects**
  - FAST, ServFace
- **Ongoing tandardization activities**
  - W3C

No specific focus on user interaction modeling (except for WebEng)

Terminology

The VIEW part of a software application (vs. the business logic)

- the part of the application responsible of displaying the status of the application to the use and the commands enabling the user interaction

View component

- piece of software which belongs to the view and supports the publication of dynamic content, the interaction with the user, and the binding to the business logics (namely business components), to data objects and to events

View module

- modularization construct used to represent the nested composition of the interface. Examples of modules in the Web application domain may include Web pages and frames; in desktop applications may include panels.

Focus

The VIEW part of a software application

- view components
- view modules
- events
- interaction between components
- Interaction between the user and the components (events)
- the distribution of view components and referenced data and business logic at the different tiers of the architecture

Mandatory requirements

- A **MOF-compliant metamodel** which describes the user interaction within front-end interfaces, their constituents, and their connections with back-end business components, data storage and Web Services.
- A **concrete syntax for the interaction flow notation** which provides an intuitive representation of the user interface composition, interaction and control logic for the front-end designer.
  - Specified through Diagram Definition
- A **UML Profile** consistent to the metamodel.
- A **complete mapping** between the UML Profile and the metamodel.
- An **interchange format** between tools using XMI.

Mandatory application requirements

- Multiple views for the same application
- Support for mobile and multi-device applications
- view components: the visualization and input of data, and the production of events.
- view components shall be independent of
  - the concrete widgets at PS level
  - the presentation aspects
- interaction flow, initiated by the user or by external events, in terms of:
  1. a source component;
  2. possibly, a reference to an action flow model specifying the actions triggered by the event; 3) one or more target view modules and components that are presented as the result of the interaction.
- Extensibility for components
- User context: the user status in the current instant of the interaction (position, history, machine, platform,…)

Mandatory application requirements

- Reference to content model items
- declarative specification of a data binding between a view component and a sub-schema of the Content Model
- Reference to atomic actions and actions orchestrations triggered by an event. Actions consist of business components and/or services that are invoked upon the event
- Outcome of actions and orchestrations and for describing how the view reacts to these outcomes
- Parameter passing dependencies
- Modeling capabilities for expressing the architectural tier where the business component triggered by an event is executed define
- Reference mechanism to role-based access control (RBAC) for describing access control rules to view components or modules
- mapping of the view components and modules at the PIM level to PSM artifacts

Issues to be discussed

- non-normative example models and accompanying explanatory text to illustrate the usage to define a typical front-end system for an application. Examples shall cover the metamodel, profile, concrete syntax and interchange format
- platform-independent semantics defining how the view is computed as the result of an event occurrence and of a possible action and/or orchestration invocation
- mapping of the view components and modules at the PIM level to PSM artifacts

Optional requirements

- Modularization of the model (design-time containers for reuse purposes of model fragments)
- User input validation, according to OCL or other existing constraint languages
- inference rules that make model specification simpler and more concise

Evaluation

- How compact and usable the proposed modeling language is
- How general and platform-independent the language and its concepts are
- The availability of a proof of concept implementation
- How compatible the proposal is with respect to existing, well-established practices in user interaction modeling

Agenda

Event or Activity	Actual Date
Preparation of RFP by TF	November 14, 2011
RFP placed on OMG document server	November 14, 2011
Approval of RFP by Architecture Board Review by TC	December 16, 2011 December 2011
TC votes to issue RFP	December 16, 2011
LOI to submit to RFP due	May 20, 2012
Initial Submissions due and placed on server (4 week)	August 20, 2012
Voter registration closes	August 20, 2012
Initial Submission presentations	August 20, 2012
Preliminary evaluation by TF	September, 2012
Revised Submissions due and placed on svr. (4 week)	February, 2013
Revised Submission presentations	March 20, 2013
Final evaluation and selection by TF Recommendation to AB and TC	June, 2013
Approval by Architecture Board Review by TC	June, 2013
TC votes to recommend specification	June, 2013
BoD votes to adopt specification	June, 2013

Open issues

- Clarify the position of the expected proposals wrt the sw architecture
- Refine the definition of “component”
- Clarify the positioning wrt the newborn W3C WG on MBUI
- Get feedback and requirements from various actors
  - IBM
  - SAP
  - Lockheed

WEBRATIO®

You think You get

Thanks

www.webratio.com
marco.brambilla@webratio.com