

Aspects of Building Web Application Systems and of Using the MBASE Approach

Anca-Juliana Stoica
Royal Institute of Technology
KTH IS Kista
Electrum 213 SE-164 40 Kista, Sweden
anca@isk.kth.se

Abstract

The paper focuses on essential aspects of building Web Application Systems which can be analysed and modelled using the MBASE approach.

Model Based (System) Architecting and Software Engineering (MBASE) [Boehm, Port, 1998] is an approach aimed at developing and integrating software system success, product, process, and property models as well as the interactions between them. Other similar approaches are: Model Based Software Engineering [Fisher et al., 1998], Software Engineering Institute's Model Based Software Engineering [Gargaro-Peterson, 1996], Honeywell's Model Based Software Development, [Honeywell 1998], Rational Unified Software Development Process [Jacobson, Booch, Rumbaugh, 1999], most of them being oriented towards product and process models (sometimes also on their associated property models).

In a previous forum presentation [Stoica, 1999], we showed how different facets of the software development are related in different approaches such as the MBASE model system and the RUP generic process framework and presented an enhancement of MBASE with a reasoning and decision framework (DF) in which optimal decisions in software development are based on formal dynamic models related to cost, schedule, process, and risk. DF is applicable especially in the early stages of the software development process when major decisions have to be taken and major risks have to be managed.

This presentation identifies the challenges of building Web Application Systems (WAS) and of using the MBASE approach based on interconnected models associated to software development. MBASE was selected because of at least the following main reasons:

- sound theoretical foundation
- the most complete system of interconnected models to date
- offers support for analysing model clashes
- flexibility, adaptability, ease of extension.

Models in MBASE are defined as patterns of something to be made, descriptions or analogies used to visualize and reason about the system to be developed and its likely effects. WAS is a Web System (WS) that allows its users to execute business logic with a browser. A WAS:

- is dynamic, in the sense that it changes the state of the business as a result of user interaction
- is used for e-business, collaborative contents, and distributed communities on the web
- deals with many different stakeholders, such as: users, domain experts, developers (architects, analysts, designers, implementers, integrators), project managers, graphic artists, lawyers, etc.
- has to be architected for continuous change
- has to cope with periods of peak interaction
- uses specific Web technologies such as: HTML, XML, EJB, SSL, CGI, TCIP/IP, ASP, JSP, ...

These are infrastructures that transform simple Web sites into business logic execution systems.

- from an architectural point of view, the basic components are: client browser, network connection, web server, and an application server, the addition of which enables the system to manage business logic and state.

Building WAS in a predictable and repeatable fashion is a team work which requires use of proven development practices such as: object-oriented, iterative development, focus on architecture, visual modeling. WAS are characterized by using specific artifacts such as: web pages, page relationships, navigation routes,

2

client-server scripts, server-page generators, modelled at the appropriate level of abstraction and detail. These have to enable the Web-specific elements of the model to interact with the rest of the system's elements. We found that the MBASE approach is very useful for representation of the success, product, process, and property aspects of WAS and that there is a synergy between developing WAS and developing MBASE, in the sense that:

- MBASE offers the framework for analysing and modelling WAS to meet the critical success factors for software projects
- WAS offers the information to enhance MBASE models (or create new ones) for this specific software market share.

Finally, an example is given to round out the paper.

REFERENCES

- [1] *Model-Based Software Development*. Honeywell Technology Center, Course Announcement, Minneapolis, MN, 1998.
- [2] A.Gargaro and A.S.Peterson, *Transitioning a Model-Based Software Engineering Architectural Style to Ada 95*. SEI Technical Report CMU/SEI 96-TR-017, 1996.
- [3] A.J.Stoica, *Facets of the Software Development Represented by Model Systems: Analysis and Enhancement*. 14th International Forum on COCOMO/SCM, 1999.
- [4] B.Boehm and D.Port, *Conceptual Modeling Challenges for Model-Based Architecting and Software Engineering*. USC Center for Software Engineering, 1998.
- [5] B.Boehm and D.Port, *Escaping the Software Tar Pit: Model Clashes and How to Avoid Them*. ACM Software Engineering Notes, Jan., 1999.
- [6] H.Maruyama, K.Tamura, and N.Uramoto, *XML and Java. Developing Web Applications*. Addison-Wesley, 1999.
- [7] I.Jacobson, G.Booch, and J.Rumbaugh, *The Unified Software Development Process*. Addison-Wesley, 1999.
- [8] J.Fisher and et al., *Model-Based Systems Engineering: A New Paradigm*. INCOSE INSIGHT, pp. 3-16, October, 1998.
- [9] R.Kalakota and M.Robinson, *e-Business*. Addison-Wesley Information Technology Series, 1999.
- [10] T.Quatrani, *Visual Modeling with Rational Rose 2000 and UML*. Addison-Wesley Object Technology Series, 2000.