

## Chapter 2.6

# Conceptual Model Driven Software Development (CMDSD) as a Catalyst Methodology for Building Sound Semantic Web Frameworks

**Thomas Biskup**

*Carl von Ossietzky University, Oldenburg, Germany*

**Nils Heyer**

*Carl von Ossietzky University, Oldenburg, Germany*

**Jorge Marx Gómez**

*Carl von Ossietzky University, Oldenburg, Germany*

### **ABSTRACT**

This Chapter introduces Hyperservices as a unified application model for Semantic Web frameworks, and proposes Conceptual Model-Driven Software Development as a means of easy adoption to them. Hyperservices are based on agent societies, provided with structured information by the Semantic Web, and using Web services as a collaboration and communication interface. Afterwards, the WASP model is proposed as a

framework for implementing Hyperservices, also adding personalization rules to modify the agents' perception as well as the HIVE Architecture as Semantic Information Server infrastructure within the WASP framework. For easier adoption of these new models, Conceptual Model-Driven Software Development is proposed. It separates the conceptual aspects from the technical details by automatically generating executable code from models while the implementation details are hidden to the end user, the service developer.

## OVERVIEW

The Semantic Web and its effects are a mainstream catalyst for current Web development. Its influence is felt across many areas of research and business development: Agent systems, knowledge management frameworks, ontology definitions, and other areas are all refined by new ideas from Semantic Web research (and vice versa). Since many complex topics are now combined with the goal of building the “Next Generation Internet”, it becomes more and more important to build sound and flexible frameworks to abstract the implementation details of the underlying technologies.

As underlying technologies are still in a state of flux as far as their implementation details are concerned, it seems to be very important to find a simple yet appropriate meta-model for the overall architecture which can be used to follow a kind of model-driven approach: Model the required system in a meta-level and then derive the actual implementation by transforming the model into executable code (or even directly executing the model). This approach allows both the early adoption of Semantic Web technologies and a continuing evolution of the implementation details.

Research shows that the underlying methodology for defining Semantic Web-oriented frameworks can be defined very well. This chapter will explain the main streams which will be integrated towards the Semantic Web and more importantly show, based on a thorough requirements analysis, how Semantic Web-oriented systems might be structured in a simple meta-model, allowing more detailed specification as research progresses. A new software development methodology, named Conceptual Model-Driven Software Development or CMDSD for short, which is currently under development in our research team, is used to provide a notion of the appropriate meta-models which will allow the early adoption of Semantic Web technologies in standard industrial projects.

The following steps in this chapter will lead to an early-stage meta-model which might be used

to connect Semantic Web frameworks in an easy and non-intrusive way with standard projects:

- The main research streams and technologies making up the Semantic Web are identified. Their interrelations and resulting requirements for frameworks and systems are shown.
- An in-depth requirements analysis concerning the architecture of Semantic Web systems and the must-have features of such features provides the groundwork for the definition of the cornerstones of future Semantic Web systems. It will be shown that the basic cornerstones are limited in scope, thus making it possible to define a very simple high-level meta-model for a model-driven strategy.
- An approach to build multi-platform Semantic Web frameworks based on the core technologies of Agents, Ontologies, Web Services, and Personalization frameworks is explained. This approach is generic enough to encompass most currently-existing frameworks and lends itself towards the integration of emerging standards. A new type of service, a Hyperservice, is derived from integrating these core technologies into a new type of service infrastructure.
- An overview of Model-Driven Architecture (MDA) and Model-Driven Software Development (MDSD) will be given. It provides the infrastructure for our extension of CMDSD (Conceptual Model-Driven Software Development) which strives to close the gap between technology expertise and conceptual requirements by building meta-models focused on the conceptual task and defining a transformation path to build complex systems from simple meta-models.

22 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/conceptual-model-driven-software-development/29412](http://www.igi-global.com/chapter/conceptual-model-driven-software-development/29412)

## Related Content

---

### Formal Specification and Implementation of Priority Queue using Stream Functions

Gongzhu Hu, Jin Zhang and Roger Lee (2013). *International Journal of Software Innovation* (pp. 66-75).  
[www.irma-international.org/article/formal-specification-and-implementation-of-priority-queue-using-stream-functions/103282](http://www.irma-international.org/article/formal-specification-and-implementation-of-priority-queue-using-stream-functions/103282)

### Sentiment Analysis of Hybrid Network Model Based on Attention

Wenqian Shang, Hongzhan Zhen and Wanyu Zhang (2023). *International Journal of Software Innovation* (pp. 1-17).  
[www.irma-international.org/article/sentiment-analysis-of-hybrid-network-model-based-on-attention/327364](http://www.irma-international.org/article/sentiment-analysis-of-hybrid-network-model-based-on-attention/327364)

### Critical Issues in Requirements Engineering Education

Rafia Naz Memon, Rodina Ahmad and Siti Salwah Salim (2014). *Handbook of Research on Emerging Advancements and Technologies in Software Engineering* (pp. 19-40).  
[www.irma-international.org/chapter/critical-issues-in-requirements-engineering-education/108608](http://www.irma-international.org/chapter/critical-issues-in-requirements-engineering-education/108608)

### Free Access to Law and Open Source Software

Daniel Poulin, Andrew Mowbray and Pierre-Paul Lemyre (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 2803-2811).  
[www.irma-international.org/chapter/free-access-law-open-source/29536](http://www.irma-international.org/chapter/free-access-law-open-source/29536)

### EA Anamnesis: An Approach for Decision Making Analysis in Enterprise Architecture

Georgios Plataniotis, Sybren de Kinderen and Henderik A. Proper (2014). *International Journal of Information System Modeling and Design* (pp. 75-95).  
[www.irma-international.org/article/ea-anamnesis/119077](http://www.irma-international.org/article/ea-anamnesis/119077)