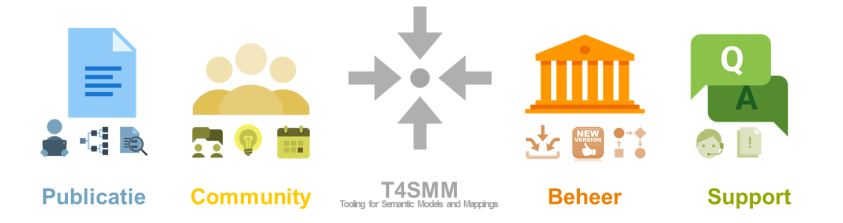
# Specification and Development of T4SMM – a Community Application for Semantic Standards

Michiel Stornebrink, Rieks Joosten (TNO)

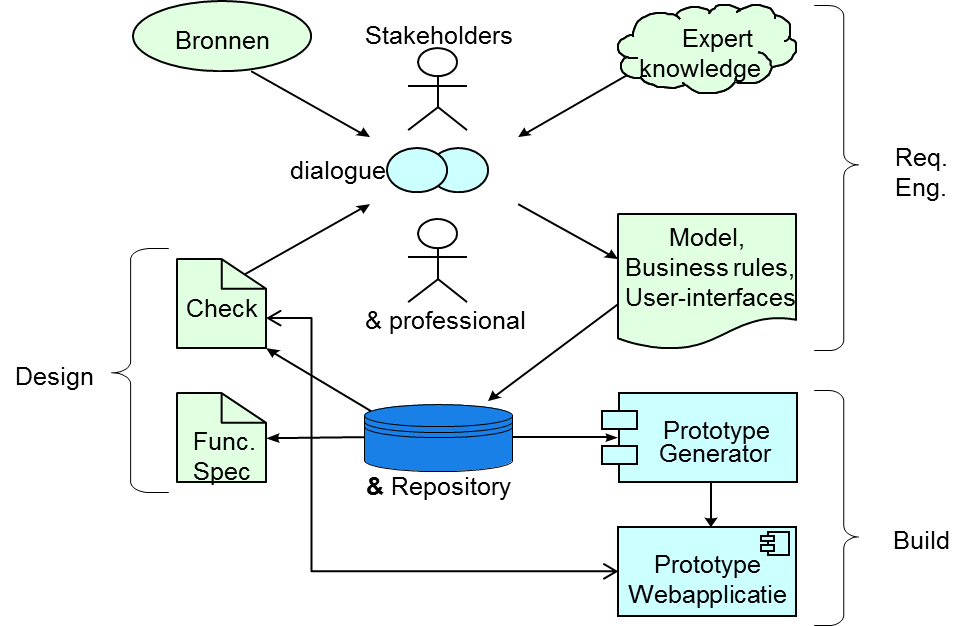
In 2015, TNO developed a first prototype of an application that supports the development and maintenance of semantic standards by a community. The application is called T4SMM (Tooling for Semantic Models and Mappings). Five potential end-user organizations formulated some functional requirements. This inspired us to draft some formal models that should capture these requirements and allow us to generate working prototypes using the Ampersand method and tools. The prototypes enabled both us and the end-user organizations to experience and demonstrate the effects of the models. We found that the quality of requirements and consensus about them is reached more easily when people experience the effects (i.e. using the prototype) rather than reverting to e.g. MOSCOW[[1]](#footnote-1) statements that are commonly used for requirements engineering.



**Purpose of the T4SMM project**

TNO’s research group Data Science developed semantic standards for many industries over the past couple of years. We see more and more industries join forces and improve the efficiency of their day-to-day electronic information exchange by means of industry specific semantic models. Many tools exist to engineer such models, but almost no tooling is available to develop, discuss, maintain and support these models. Together with a group of participants in the Dutch Staffing Industry we set out to develop an application for this purpose with the following goals:

1. Be able to publish semantic models in an online, user-friendly, clickable application (i.e. get rid of the pdf documents that are commonly used to formalize standards).
2. Support the maintenance processes including functionality for RFC (request for change), review, community building and Q&A.
3. Support knowledge management. Standards change over time by people discussing and making decisions about features/aspects of the standards.

**The Ampersand approach**

In the project we have performed four requirement and design iterations with the potential end users (stakeholders in figure below). These iterations followed the PDCA approach:

* Plan: In a dialog with the stakeholders we captured the requirements in formal Ampersand models[[2]](#footnote-2).
* Do: The models are stored in a repository and used to generate a prototype, that is subsequently put online.
* Check: Stakeholders can experience the functionality of the generated prototype.
* Act: The functionality is evaluated and findings are used in the next iteration to adjust the design.

It is characteristic for Ampersand that it does not require stakeholders to precisely formulate and validate requirements as many other approaches do. Instead, it asks stakeholders to validate the the formalized model by comparing their actual (and possibly non-explicit) needs and wishes with the functionality provided by the generated prototype. The figure below is a screenshot of what this looked like:

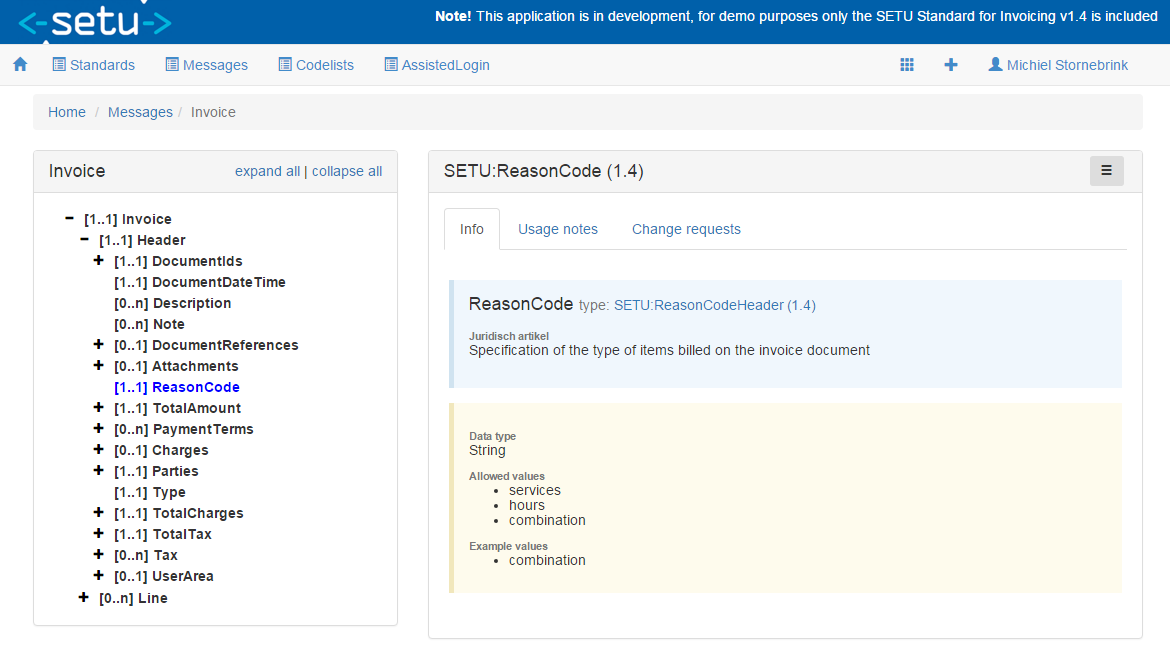
****

Figure 1: Impression of prototype

**Characteristics and Benefits of Ampersand Use for T4SMM**

By using the Ampersand method and prototyping tooling we were able to create a functional design for the application (prototype), which in the end was converted to a working web-service. Here are some characteristics and benefits of using Ampersand for T4SMM:

* The design resulted in 300+ function points (NESMA v2 method)
* Within 3 months
* In a multi stakeholder setting
* Actively involving the end users
* With validated functionality through prototypes

1. MoSCoW - Must haves,should haves, could haves and won’t haves. [↑](#footnote-ref-1)
2. Ampersand models use a typed relation algebra for its formalizations. [↑](#footnote-ref-2)