Generic system functions in OSAS

R. Bosman, J. Lukkien

OSAS (Open Service Architecture for Sensors) is an event-based programming system for sensor networks designed by the System Architecture and Networking group in TU/e. It contains a limited number of system functions, like handling flooding and code upload. Any system function could, in principle, be written in the language as well. The goal of this project is to come up with a minimal set of built-in system in two categories: essential (like the code upload) and pragmatic (which could be expressed in the WasCol language but have a more efficient direct implementation). Of the latter category we want to first have a WasCol implementation.

In the current language/system, putting a handler into a message is implicit; it is done by the functions send_message/send_to_subscribers. The task includes a way to describe how to compose a message, by putting handlers in, which means proposing consistent syntax for this.

The generic functions/handlers that must at least be addressed are the following:
1. routing & flooding
2. acknowledgement (to be used by any service that wants to acknowledge completion of something)
3. subscription refreshment
4. performing an operation on all data elements.

This assignment is done in close collaboration with researchers in SAN. The expected outcome is a demonstration prototype documented in a research paper. This project is less clear in what can be achieved and is very open. The final goal is to be able to describe detailed low-level functionality in the language as well and to decide later to optimize it by inclusion into the Virtual Machine implementation.