FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Diplomarbeitspräsentation

computer languages

Semantic XVSM
Design and Implementation

Masterstudium:
Software Engineering & Internet Computing

Lukas Klausner

Technische Universität Wien
Institut für Computersprachen
Arbeitsbereich: Verteilte Systeme
BetreuerIn: A.o. Univ. Prof. Dr. Dipl.-Ing. eva Kühn

Context

XVSM is a tuple spaces framework for coordination and communication in distributed systems:

- entries are the communication data and have a tree structure (e.g. Java object)
- operations read, take and write entries from/into containers
- a conatiner is hosted by one XVSM runtime (commonly on a server)
- support for concurrent transactions

Semantic Web technologies (platform independent W3C standards):

RDF a language to describe data in a graph structure
OWL a language for ontologies that can be used for reasoning (infer new data, validate consistency)
SPARQL query and update language for RDF graphs

Motivation

How can XVSM profit from the features of Semantic Web technologies?
Because a tree is a special form of a graph, the mapping of entries to RDF graphs should be easy.
Is a simple solution for problems of distributed task allocation possible with Semantic XVSM?

Scientific Approach

1. experimental prototyping
2. specification that includes alternative options and decision statements
3. evaluation prototype (reference implementation of the specification)
4. implementation of a use case application (task allocation)
5. benchmarks of the prototype and the use case implementation

Semantic Web Technologies in XVSM (Contribution)

basic (also supported by reviewed projects, but with improvements):

- platform independent RDF data format for storing and exchanging semantic entries
- Data models of applications can be described in OWL.
- Entries can be selected by defining SPARQL queries (template matching like in original tuple spaces, but much more powerful, and with addition filtering functions).
- reasoning
  - to enrich communication data (entries) with implicit data from models, knowledge bases, and coordination meta-data
  - to validate consistency of these communication data against data models

new functions and properties:

- The complete coordination logic of an application can be formulated with SPARQL and OWL. This logic can be dynamically managed by writing and removing entries during runtime. Special entries for:
  - managing the ontology that is used for reasoning
  - storing reusable queries
  - subscribing for notifications
  - sub-queries that enable distributed joins
  - mapping of Java objects to semantic entries
- The interactions with the data store are specified by exclusive use of SPARQL.

Example (Figure)

The figure shows simplified interactions of agents with an XVSM runtime in a task allocation application.
An operator agent writes an entry that describes a task requiring the skills (cutting and spraying) in an XVSM container.
A task executor agent reads destructively (takes) such an entry by defining a template (query) to get only the tasks that she/he is able to execute.
The tasks and the query template are simplified in this figure. The ontology ”t” belongs to the example application.

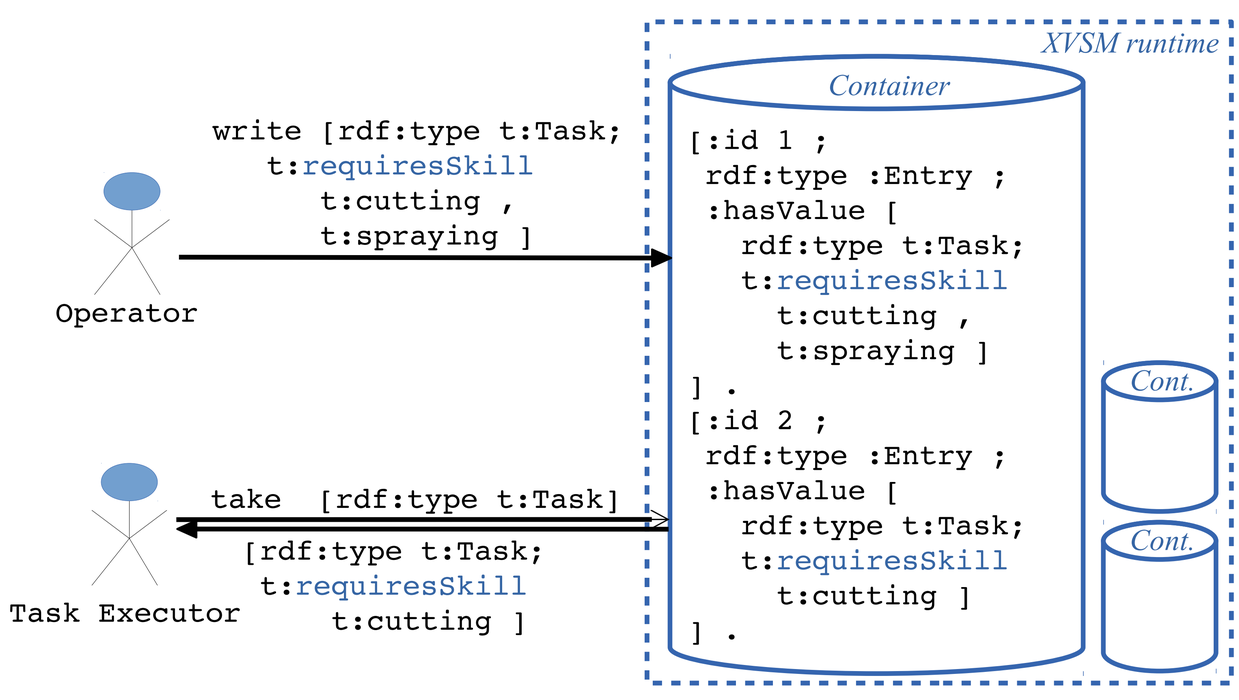

Figure : Users accessing Semantic XVSM

Kontakt: lukas@palu.at