ReApp Store – a semantic AppStore for applications in the robotics domain

Ana Sasa Bastinos[1], Peter Haase[1], Georg Heppner[2], Stefan Zander[2], Nadia Ahmed[2]

[1]fluidOperations, Altrottstraße 31, 69190 Walldorf, Germany
(ana.sasa, peter.haase)@fluidops.com
[2]FZI Research Center for Information Technology, Haid-und-Neu-Straße 10-14, 76131 Karlsruhe, Germany
(heppner, zander, ahmed)@fzi.de

Motivation

Many companies have a high demand for flexible and economical automation solutions. As the development of such is time-consuming and requires specialized knowledge this usually demands significant investments. The more or less isolated development also prevents sharing and reuse of applications. Due to this, using robot-based automation solutions is often not sensible for medium-sized companies.

In order to enable a wider use of robot-based automation solutions in medium-sized companies, we present a common repository of robotics applications to facilitate their reuse. The repository provides a way to find the application one needs, for example based on the requirements of a specific use case and a given situation. The goal of our work is to develop a semantic AppStore with robotic applications (apps) that not only contains a catalog of robotic apps, but also comprises their semantic descriptions and provides semantic search for apps based on their capabilities and properties. This semantic repository of robotic apps is developed in the scope of the ReApp project [3], hence its name ReApp Store. The ReApp Store is implemented based on real-world requirements of the end-user partners of the project consortium with the purpose to offer real-world apps from the domain of robotics.

Users and Purpose of the ReApp Store

The ReApp Store is intended for three types of users: general public, customers of robotics apps, and developers of robotics apps. The main purpose of the ReApp Store is to enable users to find information about the apps that app providers have to offer, to browse through the apps, to search for the apps, and to download the apps. By facilitating the semantic properties, the ReApp Store assists users to find the right application for their purpose, especially if they do not possess the specialized knowledge to decide what software is best suited for the task. Furthermore, developers can also upload their apps to the ReApp Store from their development environment, which includes semantic descriptions of apps as well app artifacts, such as installation and support files.

Architecture

The ReApp Store is developed as a Linked Data application on the fluidOps Information Workbench platform [1] using W3C standards like OWL, RDF and SPARQL (Fig. 1). The Information Workbench is a Web-based open platform for Linked Data and Big Data solutions in the enterprise. In the ReApp Store, we use the OpenRDF Sesame triple store [5] and the Hermit OWL reasoner [4] in the backend, in order to enable semantic wiki pages and to incorporate the ability for semantic querying.

ReApp Ontology

An important part of the ReApp Store is the underlying ReApp ontology. It is composed of the base ReApp ontology, a software taxonomy and a hardware taxonomy. The base ontology defines base classes from the domain of robotic apps, such as types of apps, their properties, messaging used by apps, and relationships between apps and software and hardware taxonomy concepts.

The software taxonomy consists of a hierarchy of software capabilities as well as of algorithms that can be used to implement them. It is based on the sensing, planning and acting paradigm [2]. In the ReApp Store, it is used to relate apps with their capabilities and with algorithms they implement.

The hardware taxonomy contains classification schemes for different hardware types such as sensors, output formats generated by hardware components, and applications scenarios a component can be used in, e.g. hazardous area protection. In the ReApp Store, it is used to relate apps with the hardware components they can access.

Moreover, the hardware and software taxonomies contain axioms that allow for the computation of capabilities that hardware components and apps exhibit. This enables a reasoner to infer the capabilities needed, e.g. for building a Pick&Place application, or to retrieve a list of components that satisfy a specific capability requirement.

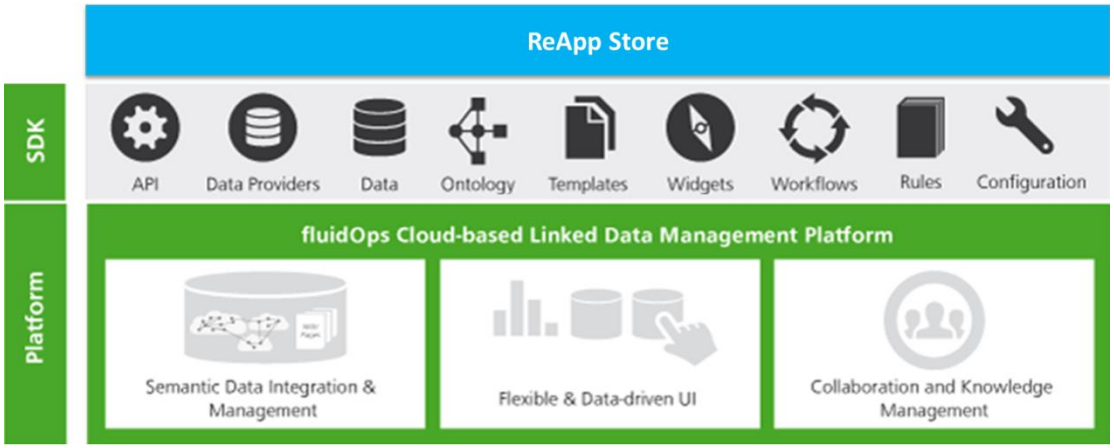

Fig. 1. ReApp Store Architecture

Use case

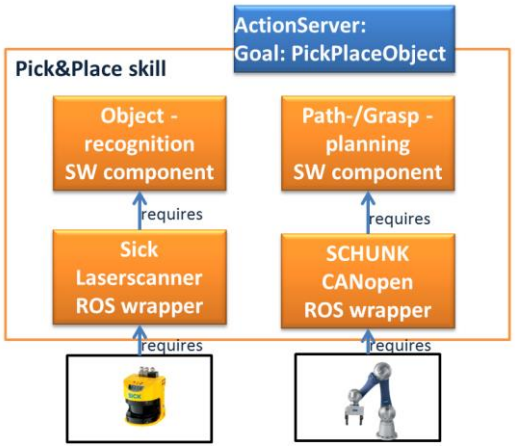

Fig. 2. Pick&Place Skill

In order to present the real-world deployment of the ReApp Store, we will present a use case of a Pick&Place skill application as it is widely used in industrial packaging applications (Fig. 2). It acts as a server for the action of picking and placing an object for other applications that may use it. It uses a laser scanner and a robot arm with a gripper as hardware components. We will show how the Pick&Place skill, as well as its composite applications, are represented in the ReApp Store, together with different types of semantic search for the apps, such as searching for applications compatible with the individual components of a composed app, and searching for apps with specific capabilities. The use case presentation will highlight how a fitting software component can be found by satisfying the requirements of the application without specific domain knowledge.

References

[1] Haase, P., Eberhart, A., Godelet, S., Mathäß, T., Tran, D.T., Ladwig, G., Wagner, A.: The Information Workbench. Interacting with the Web of Data. In: Proceedings of the 3rd Future Internet Symposium. Berling, Germany (2010).

[2] Murphy R., An Introduction to AI Robotics (Intelligent Robotics and Autonomous Agents). MIT Press, (2000).

[3] ReApp project, http://www.reapp-projekt.de.

[4] HermiT OWL Reasoner, http://hermit-reasoner.com.

[5] OpenRDF Sesame, http://www.openrdf.org.