A Proof-of-Concept Implementation of a Semantic Container Management System for Air Traffic Management

Eduard Gringinger[1], Christoph Fabianek[1],
Christoph G. Schuetz[2][0000−0002−0955−8647],
Johann Stöbich[1,2], Bernd Neumayr[2][0000−0003−1551−172X], and
Michael Schrefl[2][0000−0003−1741−0252]

[1] Frequentis AG, Vienna, Austria
firstname.lastname@frequentis.com
[2] Johannes Kepler University Linz, Linz, Austria
{schuetz,stoebich,neumayr,schrefl}@dke.uni-linz.ac.at

Abstract. As part of the BEST project – a collaborative research effort under the Single European Sky ATM Research (SESAR) initiative – we developed the concept of the semantic container, aiming to facilitate information sharing and reuse in air traffic management (ATM) and air traffic operations, enabled by semantic technologies. In this context, a semantic container is a package of ATM information with an ontology-based description of the contents. The description serves for the management of semantic containers: The formal definition of the contents of a semantic container allows to automatically match existing semantic containers with an application's information need. This demonstration showcases a proof-of-concept implementation of a system for the distributed and versioned storage of semantic containers.

Keywords: Semantic Technologies · Distributed Databases · Version Control · System Wide Information Management.

1 Introduction

Air traffic management (ATM) aims to ensure safe and efficient flight operations through the timely provisioning of relevant information to stakeholders such as pilots, air traffic controllers and others. Among the most common types of information exchanged in ATM, are messages such as Digital Notices to Airmen (DNOTAMs), which notify stakeholders of important events like runway closures and malfunctioning airport equipment, and Meteorological Terminal Air Reports (METARs), which report current weather conditions at airports. Other types of messages report on weather in regions of airspace, weather forecasts at airports or in regions of airspace, and significant meteorological events.

Different tasks in ATM and air traffic operations have different information needs. For example, planning of air traffic flow requires different types of information than piloting a flight. Furthermore, piloting a flight from Vienna (VIE) to Frankfurt (FRA) requires different information content-wise than a flight from Frankfurt (FRA) to Rio de Janeiro (GIG). While pilot briefings for both routes contain, among other message types, DNOTAMs and METARs, the briefing for the VIE-FRA flight contains DNOTAMs and METARs relevant for VIE and FRA airports as well as messages relevant for en-route segments of the flight; messages for GIG airport will not be part of the VIE-FRA briefing.

The System Wide Information Management (SWIM) concept, which is currently being developed as part of the Single European Sky ATM Research (SESAR) initiative, stipulates a service-oriented architecture and standard exchange models for sharing ATM information (see [1] for further information). Hence, information services, which are indexed in the SWIM registry, provide stakeholders with relevant ATM information in a standard format. In order to facilitate the packaging and sharing of ATM information inside the SWIM environment, the BEST project[3] developed the semantic container approach. A semantic container is a package of data items such as DNOTAMs or METARs with an ontology-based, semantic description of the contents which allows to match an application's information need to existing semantic containers. SWIM information services will then receive semantic containers as input and return semantic containers as output.

In this demonstration[4], we present a proof-of-concept implementation of a semantic container management system (SCMS) for the distributed, versioned storage of semantic containers in ATM. We refer to Deliverables 3.1 [3] and 3.2 [2] of the BEST project for further information on use case scenarios and the implementation, respectively. The remainder of this paper is organized as follows. In Sect. 2, we sketch the semantic container approach. In Sect. 3, we present an overview of the SCMS and sketch integration with the SWIM concept.

2 Semantic Containers

A semantic container has a semantic description, which defines the membership condition for data items [5]: Every data item, e.g., DNOTAM or METAR, that fulfills the membership condition belongs to the container. Such a membership condition has multiple facets – dimensions of content description. With each facet, a membership condition associates a concept from an ontology. For example, a container's membership condition may associate a concept LOVV with a spatial facet to indicate that the container incorporates all DNOTAMs relevant for a particular region. Using subsumption reasoning, an information need expressed as an ontology class can be matched to existing semantic containers. For example, a membership condition indicating relevance for the LOVV-TS1 route segment is subsumed by a membership condition indicating relevance for the LOVV region as a whole, which contains that specific route segment.

[3] "Achieving the benefits of SWIM by making smart use of semantic technologies". For more information, visit the project website: https://project-best.eu/

[4] A demonstration video of the implementation is available on the website of the BEST project: http://project-best.eu/video/best_180604_final.mp4

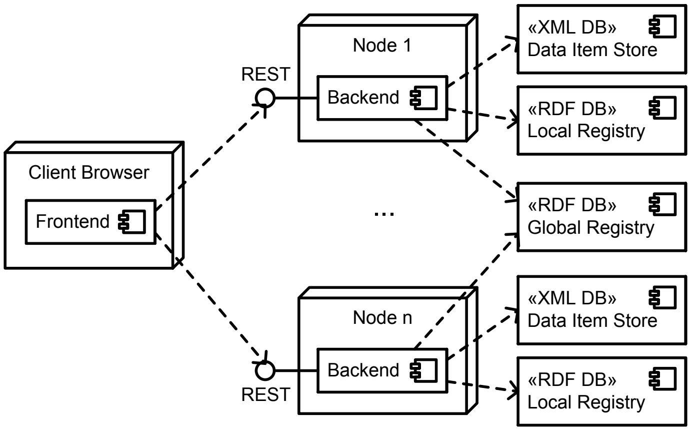

Fig. 1. An overview of the semantic container management system

In order to separate the semantics from the storage concerns, i.e., distribution and versioning, we distinguish between logical and physical semantic containers [4]. Each logical container can be allocated at multiple server nodes, resulting in multiple physical containers – one for each server where the logical container is allocated. Furthermore, a physical container maintains multiple data sets to account for different versions of the physical container when updates occur. When looking for a semantic container that matches a certain information need, the logical containers are relevant.

3 System Overview and SWIM Integration

Figure 1 illustrates the architecture of the distributed semantic container management system (SCMS). For demonstration purposes, we provide a web frontend that allows to create and allocate containers at multiple server nodes. The frontend accesses via REST API multiple backend nodes at different locations, e.g., in Vienna, Frankfurt or even on an aircraft. Each backend node maintains an XML database for the storage of the actual data and an RDF database that serves as a local registry of physical containers. The different backend nodes share a global registry of logical containers, which may be (fully) replicated for increased availability. Thus, the SCMS comprises both a knowledge base for metadata and a database that contains the actual data items.

Using the presented proof-of-concept prototype, we demonstrate potential integration of semantic containers from the BEST project into the SWIM network – a *BEST-enabled SWIM.* In this scenario, an adapted Frequentis SWIM Registry provides not only information about SWIM services but also about semantic containers. The SCMS is used to define and create containers that are then visible in the SWIM registry. On an organizational level, the Frequentis

SWIM Integration platform (MosaiX) is used to configure the SWIM information for the specific SWIM applications. Finally, the information is accessed by the SWIM applications. To show BEST integration on an application level, we used an existing prototype – the Integrated Digital Briefing – which was developed in SESAR 1.

In a BEST-enabled SWIM, semantic containers would facilitate reuse and caching of information as well as redundant allocation of safety-critical information, in addition to tracking changes of the relevant information over time. Consider, for example, that there are several main routes of air traffic. For each of these most frequent routes, a semantic container could package the static information that does not change very often and many flights on that route require. When collecting the relevant information for a specific flight on such a route, rather than starting from scratch, the application could start with the already existing container. An application could then customize the container by creating a new container, based on the existing container, that adds additional information relevant for the specific flight. Redundant allocation of semantic containers is important for safety reasons. Important information could be stored in multiple locations, including the aircraft. Tracking changes in the actual data set of a physical container over time is important to ensure auditability in case of accidents. That way, the version of a physical container in a particular location at a particular point in time can be reconstructed.

Acknowledgments. This project has received funding from the SESAR Joint Undertaking under grant agreement No 699298 under the European Union's Horizon 2020 research and innovation program. The views expressed in this paper are those of the authors.

References

1. EUROCONTROL: System wide information management (2018), `https://www.eurocontrol.int/swim` (Accessed: 26 October 2018)
2. Gringinger, E., Fabianek, C., Schuetz, C.G.: D3.2 – Prototype SWIM-Enabled Applications. Tech. rep., BEST Consortium (2018), http://project-best.eu/downloads/
3. Gringinger, E., Neumayr, B., Fabianek, C., Savulov, A.: D3.1 – Use Case Scenarios. Tech. rep., BEST Consortium (2018), http://project-best.eu/downloads/
4. Gringinger, E., Schuetz, C., Neumayr, B., Schrefl, M., Wilson, S.: Towards a value-added information layer for SWIM: The semantic container approach. In: Proc. 18th ICNS Conference. pp. 3G1–1–3G1–14 (2018)
5. Neumayr, B., Gringinger, E., Schuetz, C.G., Schrefl, M., Wilson, S., Vennesland, A.: Semantic data containers for realizing the full potential of system wide information management. In: Proc. 36th DASC (2017)