D5.3 – Initial DSF Connectors for external systems and services

Deliverable ID	**D5.3**
Deliverable Title	**Initial DSF Connectors for external systems and services**
Work Package	**WP5**
Dissemination Level	**PUBLIC**
Version	**1.0**
Date	**04/09/2017**
Status	**final**
Type	**Prototype**
Lead Editor	**FIT**
Main Contributors	**ISMB**

Published by the Storage4Grid Consortium

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731155.

Document History

Version	Date	Author(s)	Description
0.1	2016-06-20	FIT	First TOC draft
0.2	2016-06-28	FIT	Second TOC draft
0.3	2017-07-10	FIT	Third TOC draft
0.4	2017-07-30	ISMB	First draft
0.5	2017-08-21	ISMB	Content chapter 2, 3, 4
0.6	2017-08-30	FIT	Chapters 1 and 2, placeholder for executive summary, some proof-reading, restructuring of chapter 2, harmonization of components names
0.7	2017-08-31	ISMB	Enhancements
0.8	2017-09-01	FIT	Ready for internal review
1.0	2017-09-04	FIT	Ready for msubmission to the EC

Internal Review History

Review Date	Reviewer	Summary of Comments
2017-09-01	Vasco Delgado-Gomes (UNINOVA)	Approved: • Minor suggestions for better readability • Minor corrections • Missing acronyms
2017-09-01	Mihaela Albu (UPB)	• editorial changes • extended references

Table of Contents

Executive Summary

This deliverable presents the first results of the task 5.2 whose main goal is to develop a set of connectors to allow the Decision Support Framework to operate with 3rd party and DSO systems.

It provides first an overview of the S4G Data Warehouse implementation with its component set-up which enables the data collection of all S4G test sites. Secondly, it describes the two DSF connectors developed in the project's first phase: the Data Dispatcher and the Fronius Cloud Connector. Both connectors feed data into S4G's Data Warehouse. The former runs on the SMX box and handles data from the SMX, uploading local data periodically as well as resending data whenever failures occur, e.g. network connection downtime. The Fronius Cloud Connector runs in turn in the cloud and fetches information from Fronius hybrid systems installed in the Bolzano and Fur test sites.

Information on installation and deployment of the two connectors as well as their software dependencies and requirements is given. A preliminary API reference is presented.

This is a prototype document which reports on initial results. The number of connectors is to grow much bigger, as Figure 4 of D3.1 shows. D5.4 and D5.5, update and final, respectively, will report on further implementation results.

1 Introduction

The deliverable D5.3 Initial DSF Connectors for external systems and services documents the first results of T5.2 whose main goal is to develop a set of connectors to allow the Decision Support Framework (DSF) to operate with 3rd party and DSO systems.
Chapter 2 presents the structure of the DSF Data Warehouse before describing the prototype of the two first DSF connectors and their structure. The installation and deployment of the connectors are presented in chapter 3. Chapter 4 shows the software dependencies and requirements, whereas chapter 5 presents a preliminary API reference. Finally, chapter 6 discusses the conclusions and next steps.

1.1 Scope

Following the project work-plan, the effort of WP5 in the first phase has been focused on the development of the first components necessary to gather data from the test sites in Fur and Bolzano. In its first version, therefore, this deliverable describes mainly the initial outcomes of Task T5.2 DSF Interoperability with 3rd party and DSO systems.
Future issues of this deliverable, will be refined according to results of T5.2 as well as of T5.3 – DSF Hybrid Simulation Support, resulting in D5.4 "Updated DSF Connectors for external systems and services" at M21 and "D5.5 Final DSF Connectors for external systems and services" at M33. The work in T5.2 and T5.3 is largely intertwined with the work in WP3.

1.2 Related documents

ID	Title	Reference	Version	Date
D2.1	Initial Storage Scenarios and Use Cases		V1.1	2017-06-08
D3.1	Initial S4G Components, Interfaces and Architecture Specification		1.0	2017-08-31
D4.8	Initial USM Extensions for Storage Systems		1.0	2017-08-31
D6.1	Test Site Plans		1.0	2017-08-31

2 Connectors Prototype Overview

This chapter provides a high-level overview of the implementation of the S4G Data Warehouse, i.e. the main component set-up to collect all the test sites data in Storage4Grid, together with a description of the two first DSF connectors developed by S4G in the first phase: the Data Dispatcher and the Fronius Cloud Connector.
The Data Dispatcher is an instantiation of a North-bound SMX Connector, as seen in USM's high-level architecture depicted in Figure 4 of D3.1 as well as in one of the USM configurations in Figure 5 of D3.1. The Fronius Cloud Connector in turn resides in the cloud.

The purpose of both connectors is to enable the feeding of data into the data warehouse.

2.1 Basic Information – Data Warehouse

The DSF Data Warehouse (DSF-DWH) is the central system used to store data from all test-sites in the Storage4Grid project. Its structure is depicted in
Figure 1.

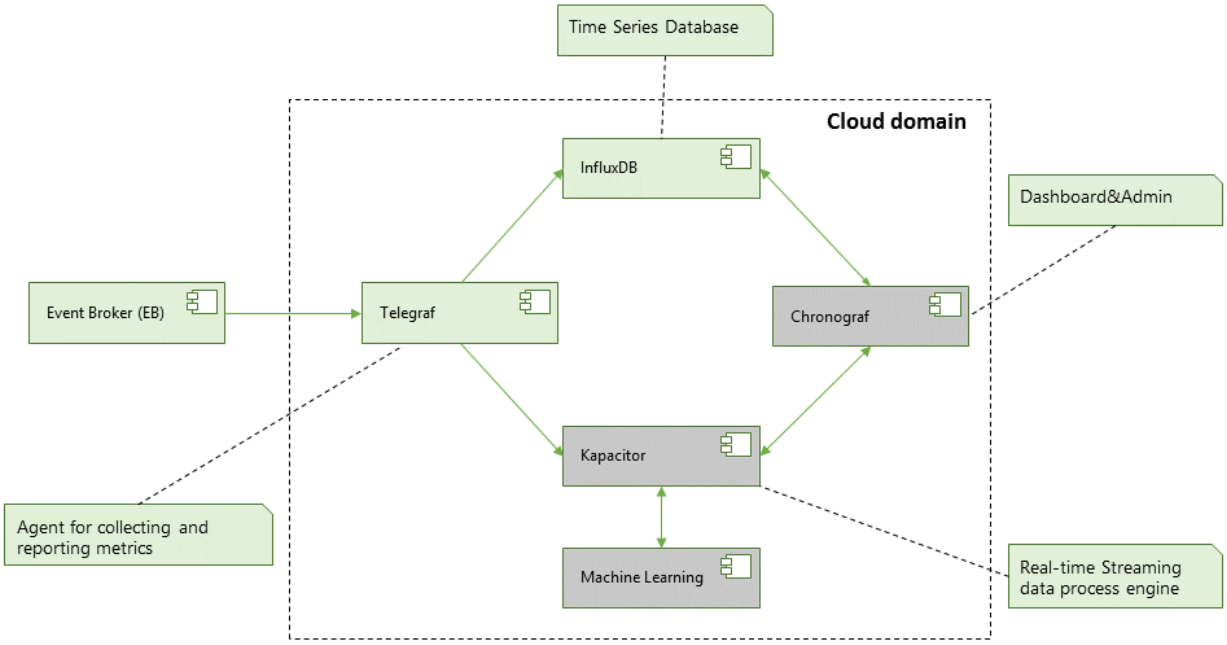

Figure 1 – Data Warehouse (DWH) Structure

The development of a Data Warehouse is not within the core objectives of the project. Nevertheless, this is a key component supporting the S4G test site, as well as a key test-bench where all data and information model developed by S4G must be reflected.
In order to match the project requirements, the DWH is implemented using existing open-source solutions for industrial-scale real-time data processing and storage, namely the TICK suite[1]. In the first phase of the project, only the basic components (Telegraf and Influxdb) are adopted. Further components (e.g. Kapacitor, Chronograf) may be adopted in subsequent phases, e.g. to support development of more advanced processing and visualization features.

[1] https://www.influxdata.com/

2.2 Data Dispatcher

The Data Dispatcher component resides in the SMX box. It feeds data from the SMX to the DSF-DWH, i.e. it synchronizes the SMX local database to the remote DSF-DWH in bi-directional fashion. The Data Dispatcher is expected to upload local data batches periodically. The interval should be configurable. At the same time, it should be able to resend local data batches whenever sending failures occurred due to e.g. network connection downtime.

The structure of the Data Dispatcher is shown in Figure 2.

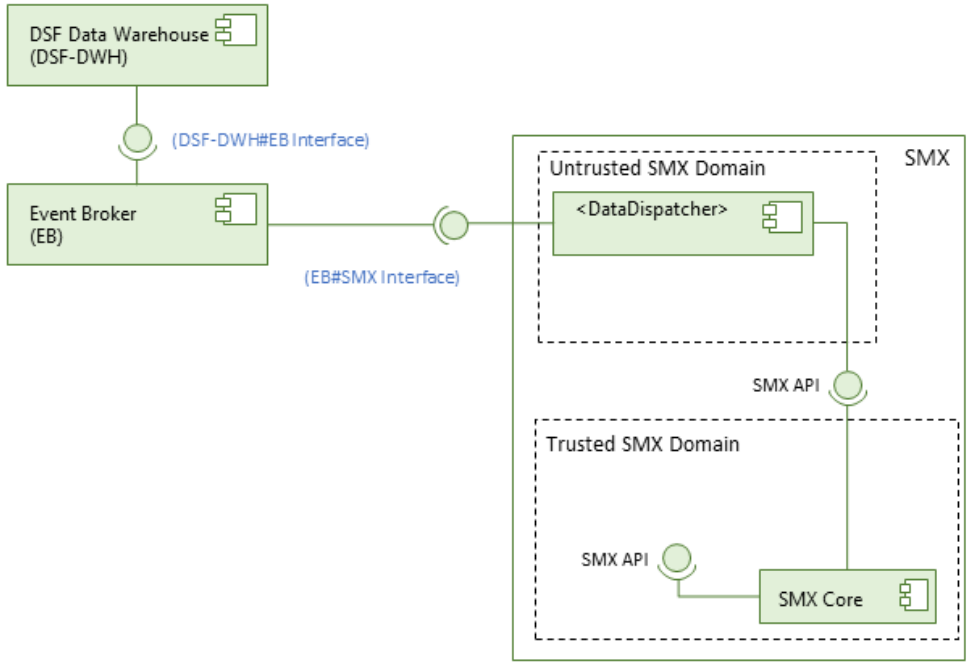

Figure 2 – Data Dispatcher Structure

2.3 Fronius Cloud Connector

The Fronius Cloud Connector component runs in the cloud and is used for fetching information from Fronius hybrid systems installed in the Bolzano and Fur test-sites. It can access data from the whole population of Fronius systems from all test-sites and make it available to the DSF-DWH via the EB. The structure of the Fronius Cloud Connector is shown in Figure 3.

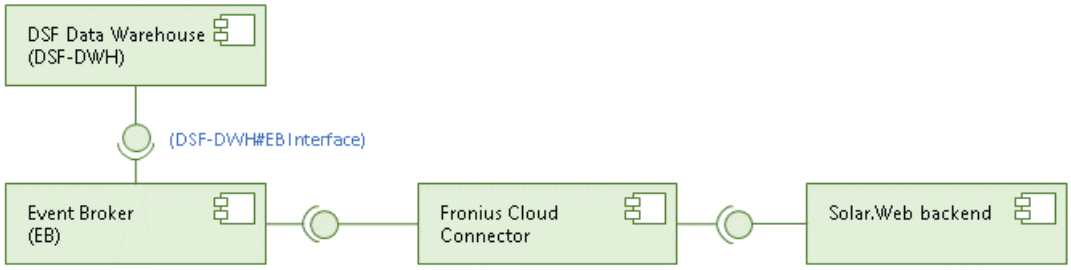

Figure 3 – Fronius Cloud Connector Structure

The functionalities of the Fronius Cloud Connector partially overlap with the functionalities of the local SMX adapter for Fronius ESS described in deliverable D4.8. The main difference is that the Fronius Cloud

Connector relies on the internet-based data infrastructure provided by Fronius, and it is therefore not indicated for applications where local control loop must be established.
While S4G use cases normally require more direct interaction with ESS, and therefore the local SMX adapter is the preferred solution, the key advantage of the Fronius Cloud Connector component is the possibility to quickly interconnect to the existing Fronius-based installations – so that operational data can be collected without deploying additional infrastructure, which is a key requirement in the early stage of the project, so that more data can be collected.

3 Installation/Deployment instructions

3.1 Data Dispatcher

This extension is designed to be deployed as part of a standard USM installation, and therefore no specific installation or configuration instructions are needed specifically to use this extension stand-alone.

3.2 Fronius Cloud Connector

The Fronius Cloud Connector can be deployed as a standard Linux system service and/or as a docker[2] component.

4 Software dependencies and requirements

Both DSF connectors are designed to be lightweight, and therefore do not have special requirements in terms of RAM or processing power.

Table 1 – Software Dependencies

Dependency	License	Role
influxdb[3]	MIT License	Part of Data warehouse: Scalable time-series database for metrics, events, and real-time analytics
telegraf[4]	MIT License	Part of Data warehouse: The plugin-driven server agent for collecting & reporting metrics.

[2] https://www.docker.com/
[3] https://www.influxdata.com/time-series-platform/influxdb/
[4] https://www.influxdata.com/time-series-platform/telegraf/

5 API Reference

5.1 Data Dispatcher

In the cloud domain, a MQTT broker is running which collects all data sources from distributed environment. For DSF-DWH, Telegraf is used to connect to the MQTT broker by using the MQTT consumer plugin[5] in Telegraf. The MQTT consumer plugin reads data in JSON format[6] from specified MQTT topics and adds messages to InfluxDB.

5.2 Fronius Cloud Connector

The connector wraps the internal API provided by Solar.Web [1]. It works as a client, fetching data from Solar.Web.

- GetData(): This function will return a JSON message with following parameters
  - P_Grid : Active power transferred from the grid to the local customer (house). A negative value means that power is generated into the grid.
  - P_Load : Home load, i.e. the active power transferred into the local loads. A positive value means that the house is in load mode, i.e. consuming active energy.
  - P_Akku : Active power transferred from the battery to the local loads (home). A negative value means that the battery is charging, i.e. in load mode, positive means discharging, i.e. in generation mode.
  - P_PV: Active power being generated by the PV (always positive or zero)
  - SOC: State of Charge, i.e. energy percentage of the rated capacity which is available in the battery

In principal, developers could integrate this client easily by setting intervals and credentials for Solar.Web.

6 Conclusions and next steps

This deliverable presents the first results of T5.2 whose main goal is to develop a set of connectors to allow the Decision Support Framework to operate with 3rd party and DSO systems.

It describes the first two DSF connectors which enable the feeding of data into the DSF-DWH, one being the Data Dispatcher, the other the Fronius Cloud Connector. The Data Dispatcher makes data available from the SMX into the DSF-DWH, whereas the Fronius Cloud Connector fetches information from Fronius hybrid systems at a global scale.

The number of connectors will increase throughout the project. A preliminary idea about the amount of connectors foreseen in the course of the project can be gotten from D3.1 *Initial Components, Interfaces and Architecture Specification* as in Figure 4 which shows USM's high-level architecture. The results will be further described in D5.4 *Updated DSF Connectors for external systems and services* at M21 as well as in D5.5 *Final DSF Connectors for external systems and services* at M33.

[5] https://github.com/influxdata/telegraf/tree/master/plugins/inputs/mqtt_consumer
[6] https://github.com/influxdata/telegraf/blob/master/docs/DATA_FORMATS_INPUT.md

Acronyms

Acronym	Explanation
DSF	Decision Support Framework
DSF-DWH	DSF- Data Warehouse
DSO	Distribution System Operator
EB	Event Broker
JSON	JavaScript Object Notation
MQTT	MQ Telemetry Transport or Message Queue Telemetry Transport
S4G	Storage4Grid
SMX	Smart Meter eXtensions
USM	Unbundled Smart Meter

List of figures

List of tables

References

[1] Fronius International GmbH, „Fronius Solar.web," Fronius International GmbH, 2017. [Online]. Available: https://www.solarweb.com/. [Accessed on August 2017].