

ZDMP: Zero Defects Manufacturing Platform



WP5: ZDMP Core Services and Middleware

EU ID: D065: Data Harmonisation and Interoperability - Vs: 1.0.0A

ZDMP ID: D5.3a

Deliverable Lead and Editor: Alexander McDonald, ICE

Contributing Partners: ICE, UNIN

Date: 2020-06

Dissemination: Public

Status: EU Approved

Abstract

The deliverables for this task, and all WP5-8 tasks, are software and are of EU type "OTHER". The software and accompanying material (eg description, instructions) is available on the ZDMP software repository which is updated dynamically. However, for EU formal reporting purposes, this brief cover document provides a formalised pointer to the downloadable software and related content. This deliverable should read in conjunction with the D006-D020 deliverables which document the software process/status for each WP/Task. This deliverable represents the status as at M18 with further living editions at M18 and M48

Grant Agreement:
825631



Document Status

Deliverable Lead	Alexander McDonald, ICE
Internal Reviewer 1	Fernando Sousa, CEI
Internal Reviewer 2	Mircea Vasile, SIVECO
Internal Reviewer 3	Stuart Campbell, ICE
Type	Deliverable
Work Package	ZDMP Core Services and Middleware
ID	D065: Data Harmonisation and Interoperability
Due Date	2020-06
Delivery Date	2020-06
Status	EU Approved

History

See Annex A.

Status

This deliverable is subject to final acceptance by the European Commission.

Further Information

www.zdmp.eu and <mailto:info@zdmp.eu>

Disclaimer

The views represented in this document only reflect the views of the authors and not the views of the European Union. The European Union is not liable for any use that may be made of the information contained in this document.

Furthermore, the information is provided “as is” and no guarantee or warranty is given that the information is fit for any particular purpose. The user of the information uses it at its sole risk and liability.

Project Partners:



Executive Summary

The main objective of “WP5: Core Services and Middleware” is to deliver the core Industrial IoT/Network support of data acquisition, interoperability, and AI/analytics supported by orchestration, monitoring and autonomous computing.

The deliverables of this work package and the WP1 Management work package are divided into software packages and document/reports. In terms of reporting:

- **Process/Status:** This report corresponds to D009 Technical Management: WP5 Report of WP1 Management: Procedures, Metrics, Coordination, and Reporting and, as identified in the DOA, focuses on the process/status of the work accomplished in WP5
- **Software:** All WP5 software deliverables of T5.1-T5.6 (type “OTHER”) are available in the ZDMP public repository with access details and install instructions further described in this report which is a ‘current’ extract of the repository

“WP5: ZDMP Core Services and Middleware” consists of the following main parts: Data Acquisition, Network Support, Data Harmonisation, Orchestration and Monitoring, Distributed and Autonomous Computing, and AI and Analytics. The tasks of WP5 are the following:

- T5.1: Data Acquisition and IIoT
- T5.2: Robust Industrial Network Support
- T5.3: Data Harmonisation and Interoperability
- T5.4: Orchestration, Monitoring, and Alerting
- T5.5: Distributed and Autonomous Computing
- T5.6: AI and Analytics

This deliverable represents Task T5.3 Data Harmonisation and Interoperability which in turn is components of the following components:

- Data Harmonisation Design Time
- Data Harmonisation Runtime

As reported in the architecture deliverable the purpose of these components is: “To define data maps which take data from existing software systems and restructure it for the needs of other systems, ie transforming data from its source format to its destination format.

These data maps aide in the transition from one file format to another - changing the data in form/schema fit for consumption by other components. This component works in conjunction with the run-time component, T5.3 Data Harmonisation Run Time. “and “To change data from one format to another using the data maps created by the Design-time component and processed in the Run-time components. This component harmonises data from a variety sources into the required formats for ZDMP applications”.

Each of the components is structured into the following sections:

- General Description
- Architecture Diagram
- Features
- Requirements
- Installation
- How to Use
- Functional Requirements Implementation Status (M18)

This report covers the period from the project start until M18 with most activity in the M13-M18 period. Further formal deliverables are due M30 and M48 as well as an informal iteration at 24.

Table of Contents

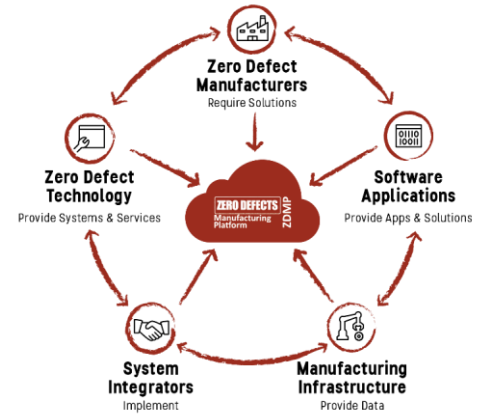
0	Introduction.....	1
1	Component: Data Harmonisation Designer and Runtime.....	3
1.1	General Description	3
1.2	Architecture Diagram	3
1.3	Features.....	4
1.3.1	Maps Designer	4
1.3.2	Semantic Reasoner.....	5
1.3.3	Ontology Storage	5
1.3.4	Data Map Parser	5
1.3.5	Data Enrichment and Feature Extraction API	5
1.3.6	Map Designer UI	5
1.3.7	Publish Map API.....	5
1.3.8	Load and Save API	5
1.3.9	Schema Data Analyser	6
1.4	System Requirements.....	6
1.5	How to use	6
1.5.1	The Welcome Page.....	6
1.5.2	Creating a New Job.....	7
1.5.3	Creating Inputs and Outputs	8
1.5.4	Semantic Reasoner.....	9
1.5.5	Building the Job	11
1.6	Functional Requirements Implementation Status (M18)	11
2	Conclusions.....	13

0 Introduction

Due to the cover nature of this deliverable; this introduction is presented in short-form only. For further information please consults D006 - Technical Management: Overview Report.

0.1 ZDMP Project Overview

ZDMP – Zero Defects Manufacturing Platform – is a project funded by the H2020 Framework Programme of the European Commission under Grant Agreement 825631 and conducted from January 2019 until December 2022. It engages 30 partners (Users, Technology Providers, Consultants and Research Institutes) from 11 countries with a total budget of circa 16.2M€. Further information can be found at www.zdmp.eu.



ZDMP aims at providing such an extendable platform for supporting factories with a high interoperability level, to cope with the concept of connected factories to reach the goal of zero-defect production. For this, the platform provides the tools to allow following each step of production, using data acquisition to automatically determine the functioning of each step regarding the quality of the process and product.

0.2 Deliverable Purpose and Scope

The deliverables for this task, and all WP5-8 tasks, are software and are of EU type “OTHER”. The software and accompanying material (eg description, instructions) is available on the ZDMP software repository which is updated dynamically. However, for EU formal reporting purposes, this brief cover document provides a formalised pointer to the downloadable software and related content. This deliverable should read in conjunction with the D006-D020 deliverables which document the software process/status for each WP/Task. This deliverable represents the status as at M18 with further living editions at M18 and M48. Specifically, the DOA states the following regarding this Deliverable:

T5.3	Data Harmonisation and Interoperability			ICE	Six Monthly
D065 D066 D067	Data Harmonisation and Interoperability	OTHER (Prototype)	PU	As above	As above
<p>This task aims to (a) extract more features from incoming data and (b) prepare the data in a form/schema suitable for other tasks in the subsequent work packages/applications. It will enable semantic enrichment with background knowledge (via the Data Model of T5.1) and data mining on real-time streams received through the data bus. The data enrichment objective is to generate, from the observed data, additional derived attributes/features, either using external background knowledge or internal relationships within data. This could include the use of external ontologies, statistical properties, data models, and/or temporal characteristics of data. The key aim is to encode functional transformations of data to help analytic techniques of T5.6 to detect non-linear and other nontrivial patterns within the data (otherwise non-detectable by traditional analytic techniques). This will be built using background technology from partner ICE, who is constructing a semantic enablement platform driving syntactical data transformation to the next level such that transformation maps and element data maps can be quickly (even automatically) generated. This will speed up the interconnections between different sources and interactions. The deliverable represents the functionalities of the semantic-based data harmonisation services.</p>					

0.3 Target Audience

The primary target audience for this document are the partners and WPs of the project, as well as the EU and reviewers.

0.4 Deliverable Context

The deliverable context is as per Section 0.2:

Primary Preceding documents:

- **D006: Technical Management Overview Report:** Represents the general software status of the project including information on commits and WP5-8 Risks and mitigations
- **D009: Technical Management: WP5 Report:** Represents the process/status and future actions of this work package, including this task. It also includes related KPIs and their status
- **D055: Technical Specification and Update:** Describes the different APIs of the components

0.5 Document Structure

This deliverable is broken down into the following sections:

- **Section 1:** Component: Data Harmonisation Designer and Runtime

0.6 Document Status

This document is listed in the Description of Action as “public” since it represents the open nature of the project’s software deliverables.

0.7 Document Dependencies

- None

0.8 Glossary and Abbreviations

A definition of common terms related to ZDMP, as well as a list of abbreviations, is available at <http://www.zdmp.eu/glossary>.

0.9 External Annexes and Supporting Documents

- See the ‘Resources’ grid within the General Description Section of each component

0.10 Reading Notes

- None

0.11 Document Updates

- This is the first version of this document

1 Component: Data Harmonisation Designer and Runtime

1.1 General Description

This component has access to raw data and ensures that data can be integrated using unified and standardised formats or the formats needed by the data recipient. It also provides basic functionalities for semantic homogenisation in a context of heterogeneous data meaning that the mapping is facilitated with reference to semantic through a ZDMP ontology.

The developed application enables a business analyst driven approach for the automatic linking of organisations' data and meta data to the reference data model.

These data Maps are available in the ZDMP Data Storage and deployed and encapsulated as services to be finally exposed as software mini-packages, ie Docker containers. These mini-packages, containing the transformation routines, are uploaded, and published in the Marketplace to advertise and commercialise them.

One of their uses is, for example, as part of the execution of a process model in which they are rendered as services that can be called.

Resource	Location
Source Code	Link
Latest Release	Vs 1.0.0
X Open API Spec	Link
Video	Coming soon

The date of generation of this component content is: 2020-06-25

1.2 Architecture Diagram

The following diagram shows the position of this component in the ZDMP architecture

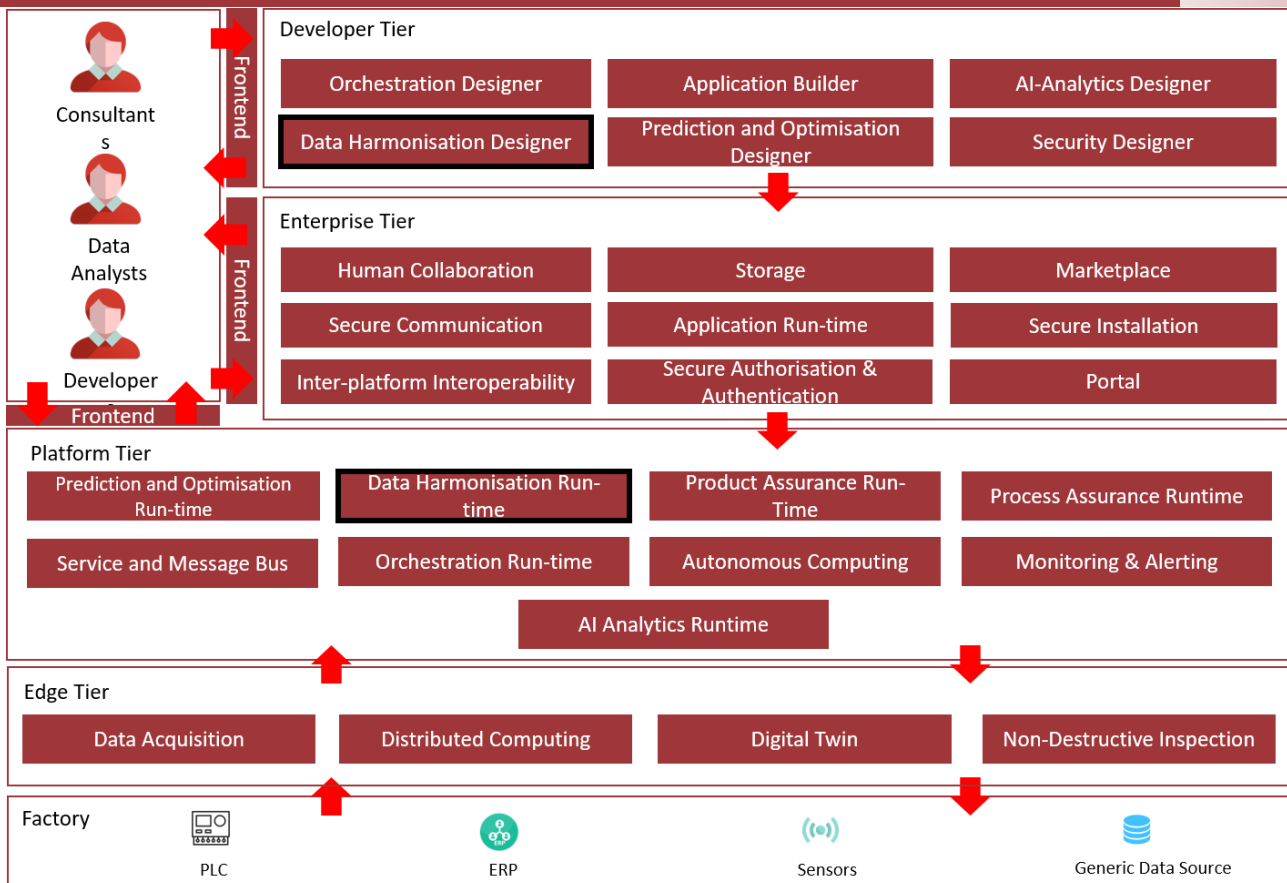


Figure 1: Position of Component in ZDMP Architecture

1.3 Features

This component offers the following features:

- Maps Designer
- Semantic Reasoner
- Ontology Storage
- Data Map Parser
- Data Enrichment and Feature Extraction API
- Map Designer UI
- Publish Map API
- Load and Save API
- Schema Analyser API

1.3.1 Maps Designer

The Maps Designer module allows a person to generate Manufacturing Maps. A Manufacturing Map file describes the rules to be executed to transform a specific syntax format A into format B which could then, for example, be used as part of a process.

This map is a java archive file (with the .jar extension) wrapped in a docker container for execution. This container executes as a transformation engine.

It offers the developer the possibility to annotate these maps with additional descriptions.

This description eases the search and selection of the appropriate mapping rules. Manufacturing Maps can be stored in T6.2 Storage and may also be sent to the T6.2

Marketplace to use them as a service in T5.4 Orchestration Designer / Run-time, or via API requests from the T6.4 Service and Message Bus.

1.3.2 Semantic Reasoner

Provides the user with helpful suggestions when performing mappings between distinct concept knowledge bases. The component relies on user input to improve suggestion quality, as it incorporates the effective mappings elected by users into a growing internal knowledge base.

It supports mapping restrictions for specific subsets of source and target mappings and provides data versatility by supporting different conceptual types of knowledge base transparently – triples, graph, ontology, or tabular.

1.3.3 Ontology Storage

Enables persistent data storage in support of the Semantic Reasoner. It stores the relevant ontology knowledge bases in graph form as well as the overarching ZDMP ontology.

The ZDMP ontology contains ZDMP specific vocabulary and ontology alignment.

1.3.4 Data Map Parser

It is the interoperability layer between the Semantic Reasoner core functionality and the Map Designer. It transforms generic requests into Semantic Reasoner queries.

1.3.5 Data Enrichment and Feature Extraction API

Derives attributes and features from the data. Use statistical properties, data models, and temporal data characteristics to discover internal relationships within the data.

1.3.6 Map Designer UI

This UI is based upon the current ICE Data Platform (IDP). Developers can access the various Mapping features in a friendly way, eg generate Manufacturing Maps or annotate them with keywords and metadata.

It also allows the connection with the Data Service components to retrieve Data from querying the data model, ie the ontology.

1.3.7 Publish Map API

This interface encapsulates the access to the T6.2 Marketplace, where the deployed Maps are stored for re-use to sell and share.

The maps may also be published to T6.1 Application Builder for integration into zApps and directly to the T5.3 Data Harmonisation Run-time.

1.3.8 Load and Save API

This interface encapsulates the access to the Storage, where data from this component is stored.

1.3.9 Schema Data Analyser

This interface manages the actual access to the external data sources and the objects where the data is generated and/or stored.

These formats can range from a text-file to a database schema passing by a CSV-type file or an XML and JSON types.

This helps to facilitate inter-platform interoperability by taking example data and meta-data from external platforms.

1.4 System Requirements

Minimum requirements needed:

- Docker

1.5 How to use

This section will show how to use the Data Harmonisation Component, with the following sections:

- The Welcome Page
- Creating a New Job
- Creating Inputs and Outputs
- Semantic Reasoner
- Building the Job

1.5.1 The Welcome Page

When the user runs the file TOS_DI-winx86_64.exe, the ICE Data Platform opens. The user is prompted to either run an existing project or create a new project. After they name a new project they will be taken to same screen as in the screenshot below.

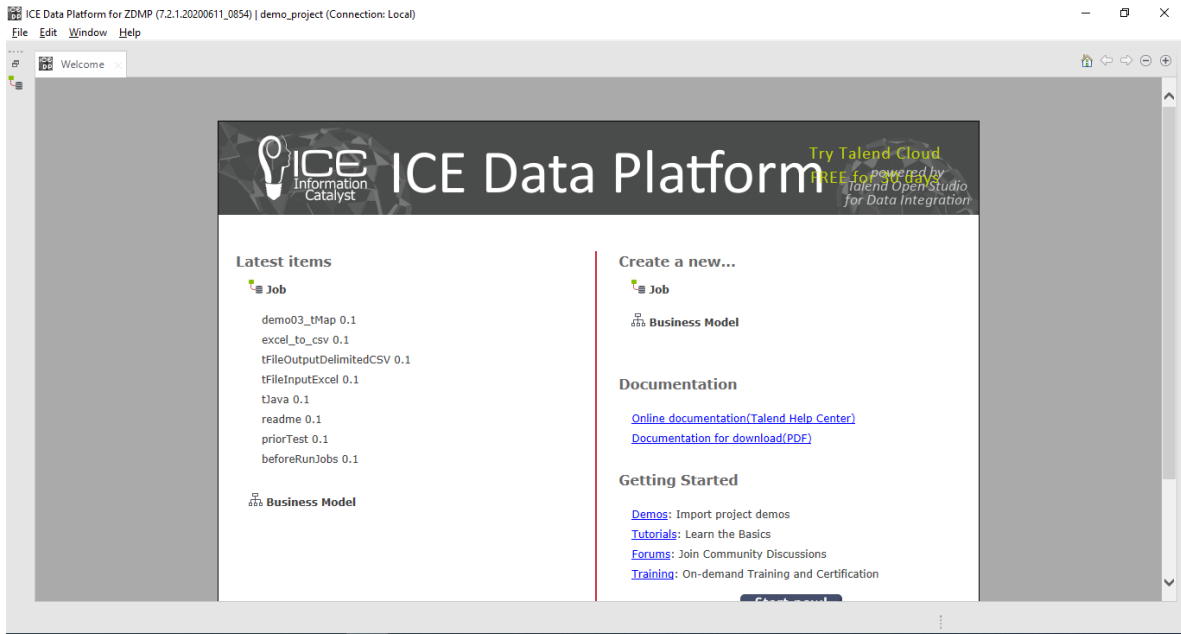


Figure 2: Screenshot of the Ice Data Platform Welcome Page

From here, the user can open a job they have previously worked on, or they can create a new job. They can also be directed towards various tutorials where they can learn more about the intricacies of the Ice Data Platform or view more documentation.

1.5.2 Creating a New Job

When the user chooses to create a job, they are greeted with a popup box that requires the user to make entries. Whilst the Name field is required, there are warnings about leaving the Purpose and Description fields blank. The user cannot continue until they enter a name, but the other details can be changed through the properties menu.

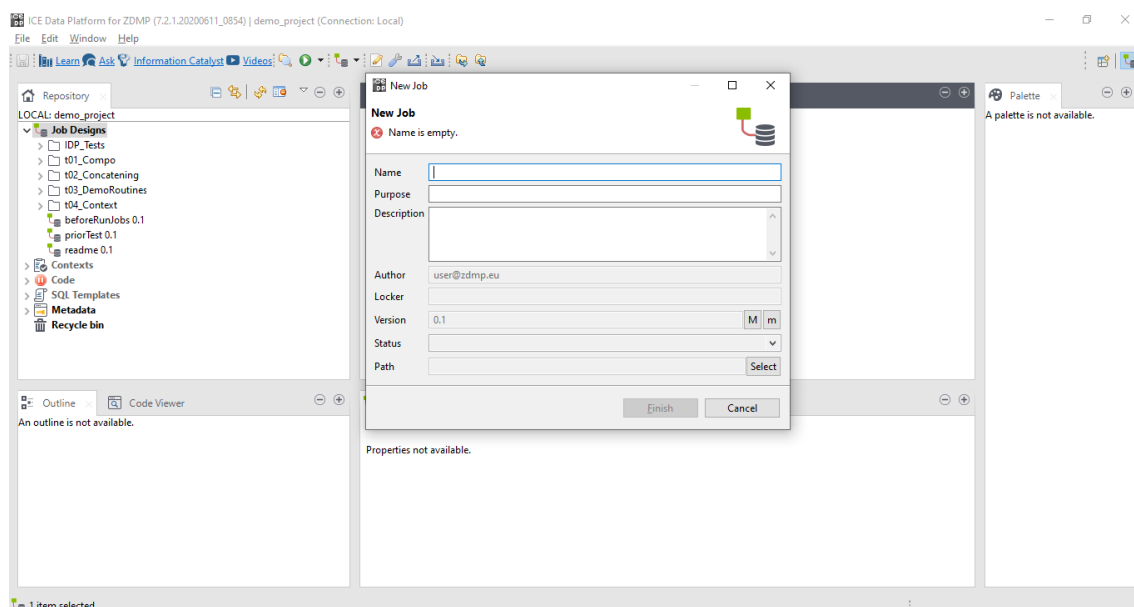


Figure 3: A Screenshot Showing the New Job Pop-up Box

When the user does select the finish button, the pop-up box disappears, and the user views the main canvas of the ICE Data Platform. On the left side of the screen, they can see the list of jobs, as well as the code base that makes up the different jobs. On the right, they can see the palette of tasks that they can use. In Figure 3, the palette has not loaded, and this process can take a few seconds before it becomes populated with elements.

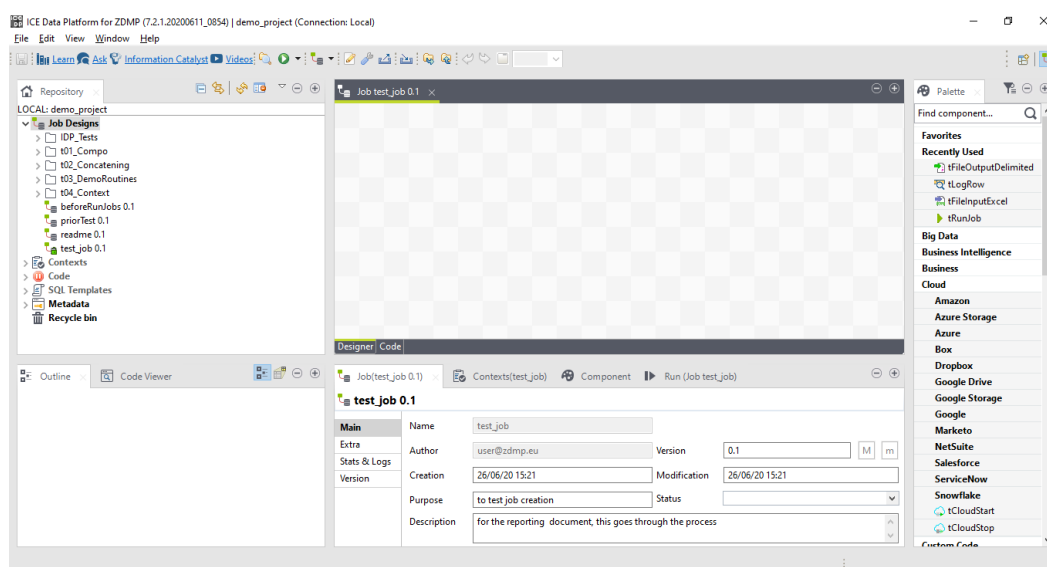


Figure 4: A Screenshot of the Ice Data Platform with a Populated Palette

1.5.3 Creating Inputs and Outputs

From the open version of the ICE Data Platform, users can click and drag elements from the palette into the canvas to create jobs. Eventually, these jobs will be exported to become executable services.

From the Palette, the user can search for specific tasks, such as importing an Excel file (tFileInputExcel), exporting to a text file (tFileOutputDelimited) or use the Semantic Reasoner (tSemanticMap).

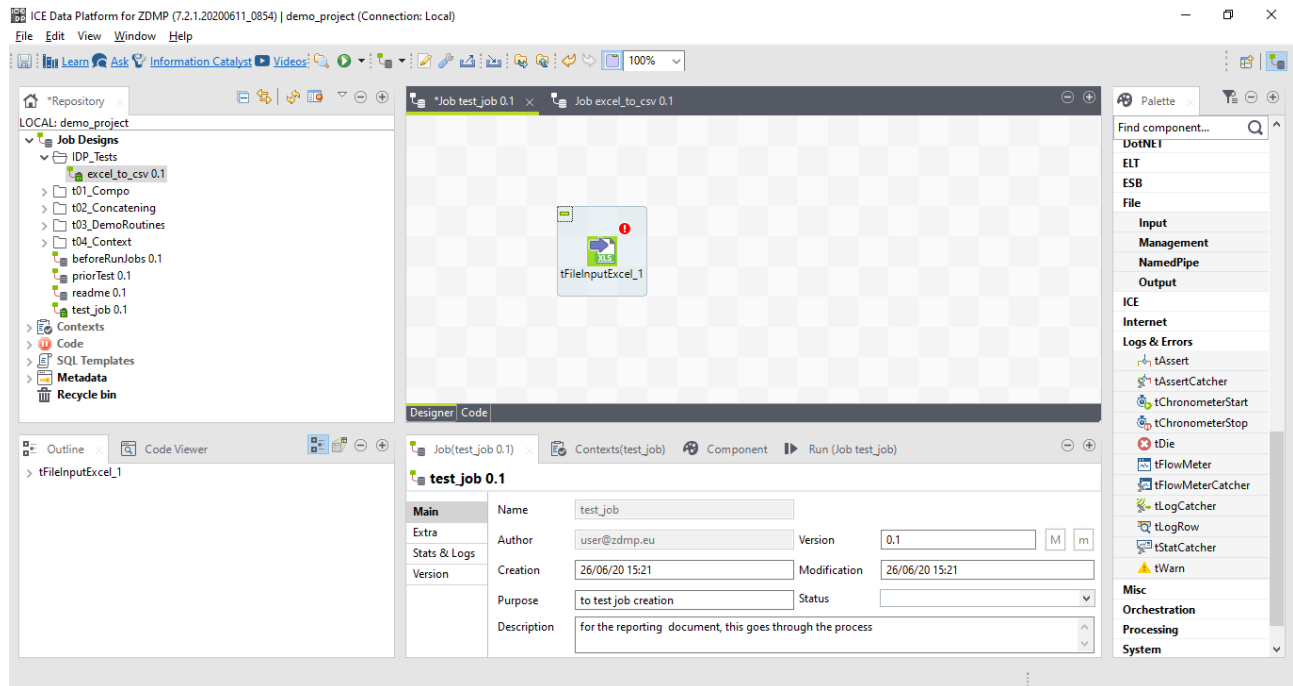


Figure 5: A Screenshot Showing the User Using the Canvas

As shown in Figure 5, the tFileInputExcel_1 task has a red exclamation symbol on it. This shows the user that they have actions to do on the task. For this task specifically, it is defining the schema of the input, and whilst this is for an Excel input, the process is similar for most inputs.

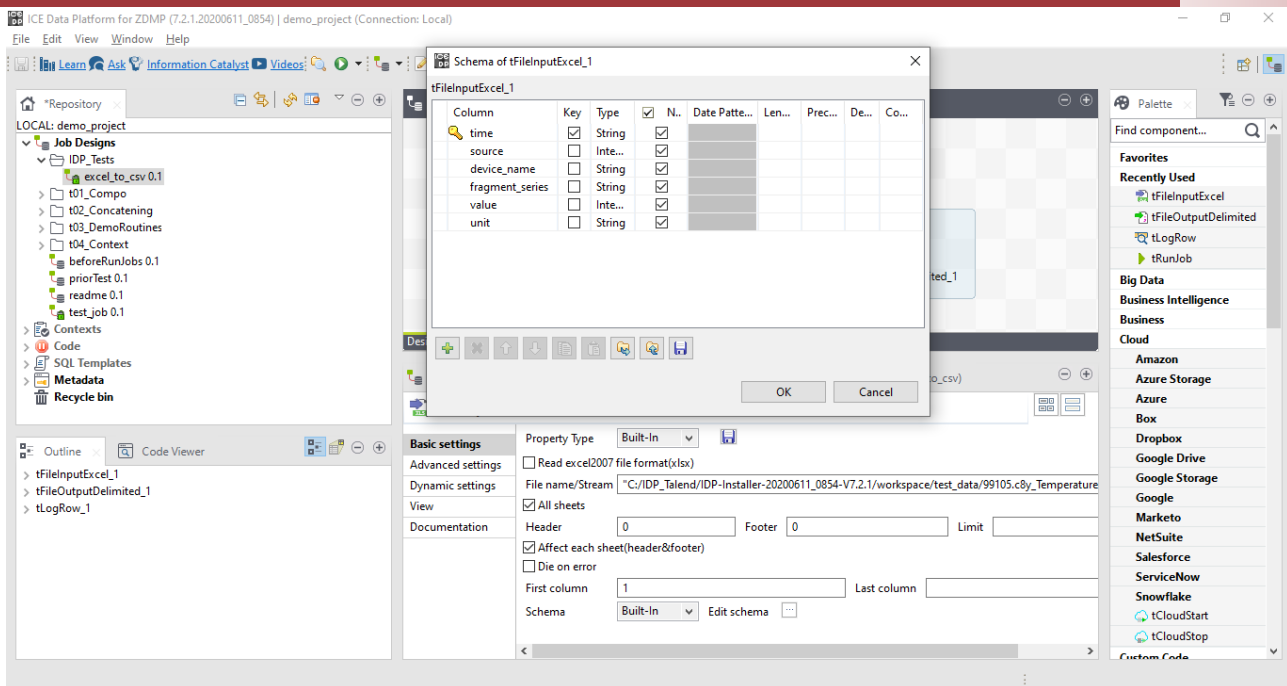


Figure 6: A Screenshot showing the File Input Properties

As shown in Figure 6, the user will be required to define the files that they want the Ice Data Platform to use. This process will be the same for both input and output processes. The user defines what they are putting into the process, and they define what they expect to get out of it.

1.5.4 Semantic Reasoner

The functionality that the ICE Data Platform can offer includes the ability to transform files between formats. The way that it does this is by using the Semantic Reasoner component, or in the palette, the tSemanticMap task.

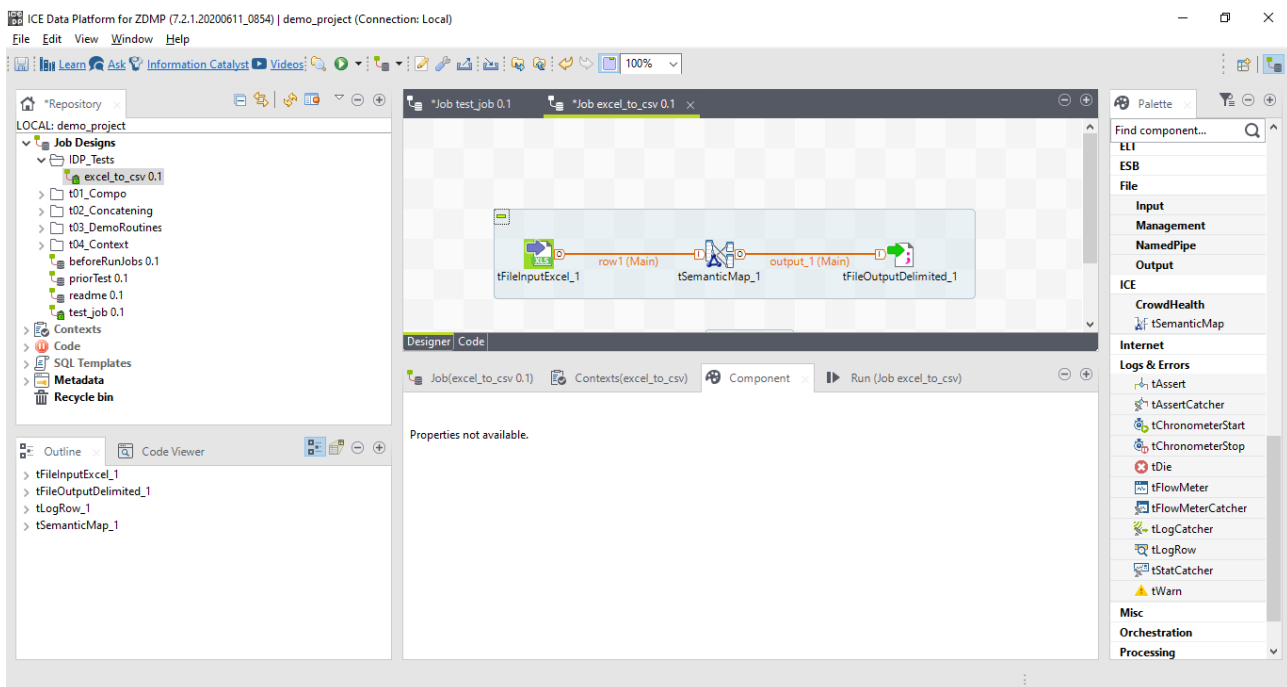


Figure 7: A Screenshot of the Ice Data Platform Canvas using the Semantic Reasoner

By double clicking on the tSemanticMap_1, they are taken to the Semantic Reasoner interface.

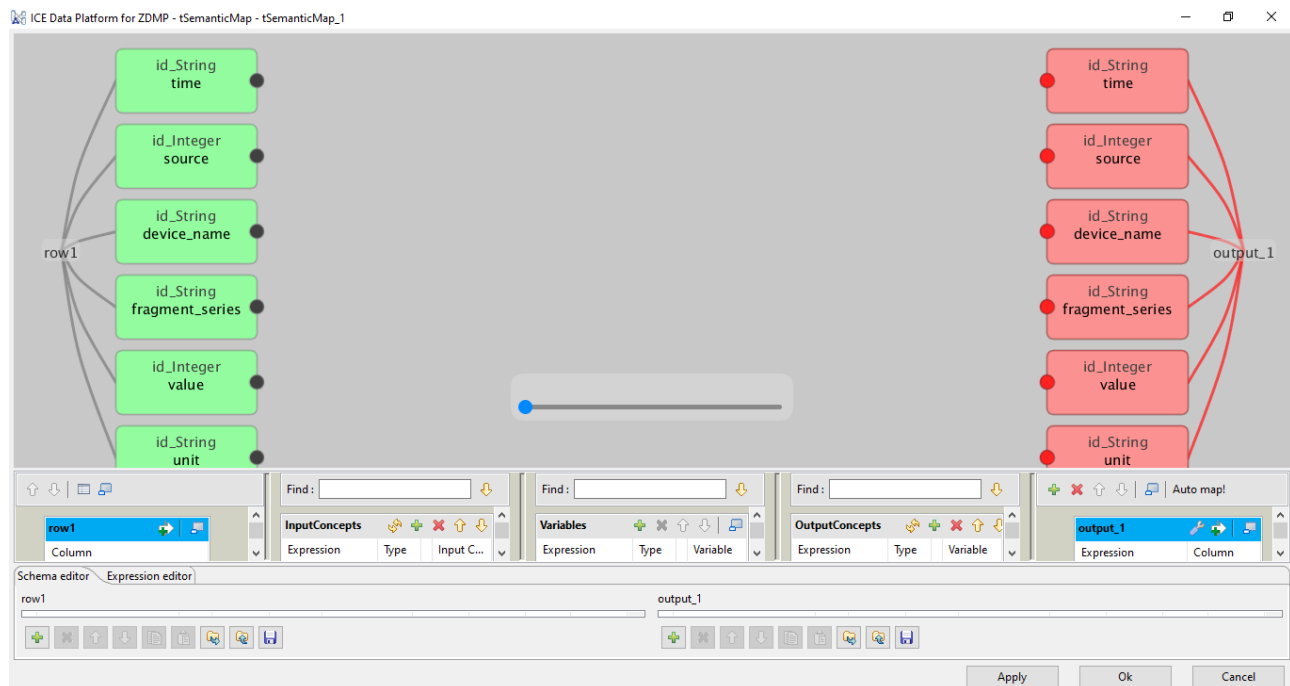


Figure 8: The Ice Data Platform Semantic Reasoner Interface

From this screen, the user can see the schema of both input and outputs to the semantic mapping task. As in Figure 8, the bottom right, there is the “Auto map!” which will use the Semantic Reasoner component to make the best guesses at which of the input data will match to the same or similar output data.

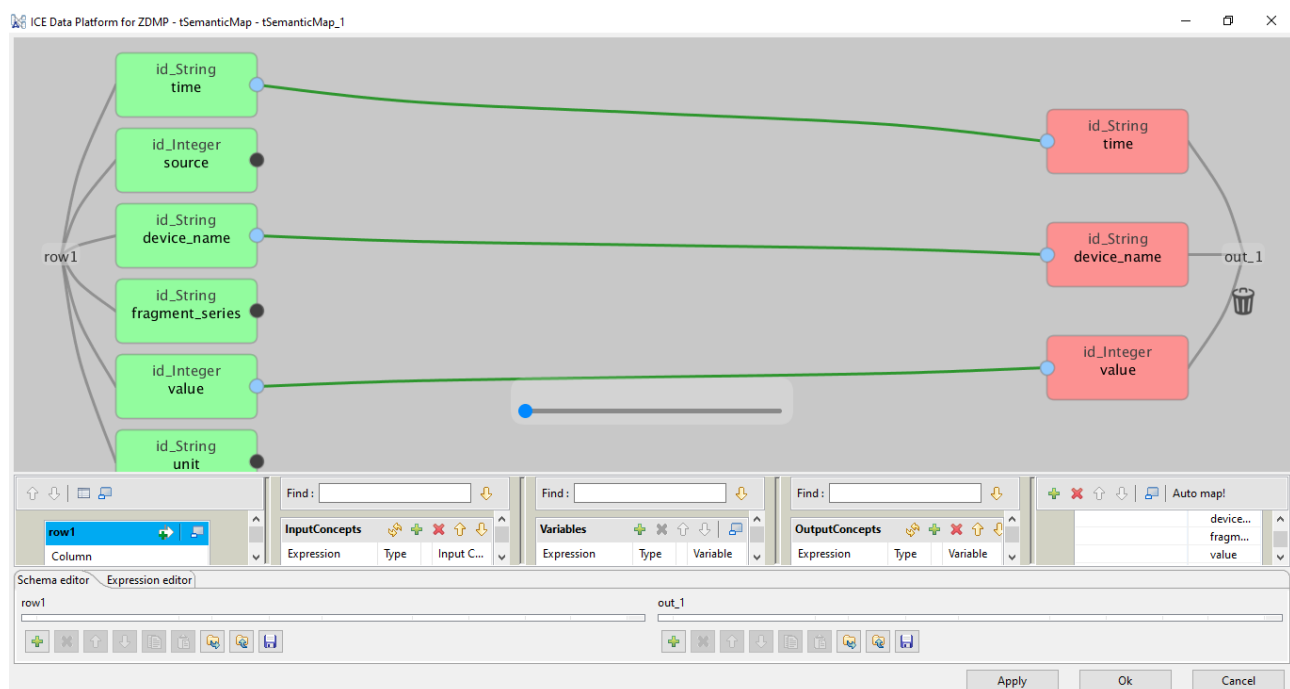


Figure 9: The Semantic Reasoner Mapping to a Smaller Output

The user can also make modifications here, and this will send data back to the Semantic Reasoner to help improve the semantic mapping in future.

When they select the apply button, these changes will be applied, and when they click the ok button, they will be taken back to the canvas.

1.5.5 Building the Job

When the user has created a job, they can build the job into an executable process. This process can be used in conjunction with other executable processes and services.

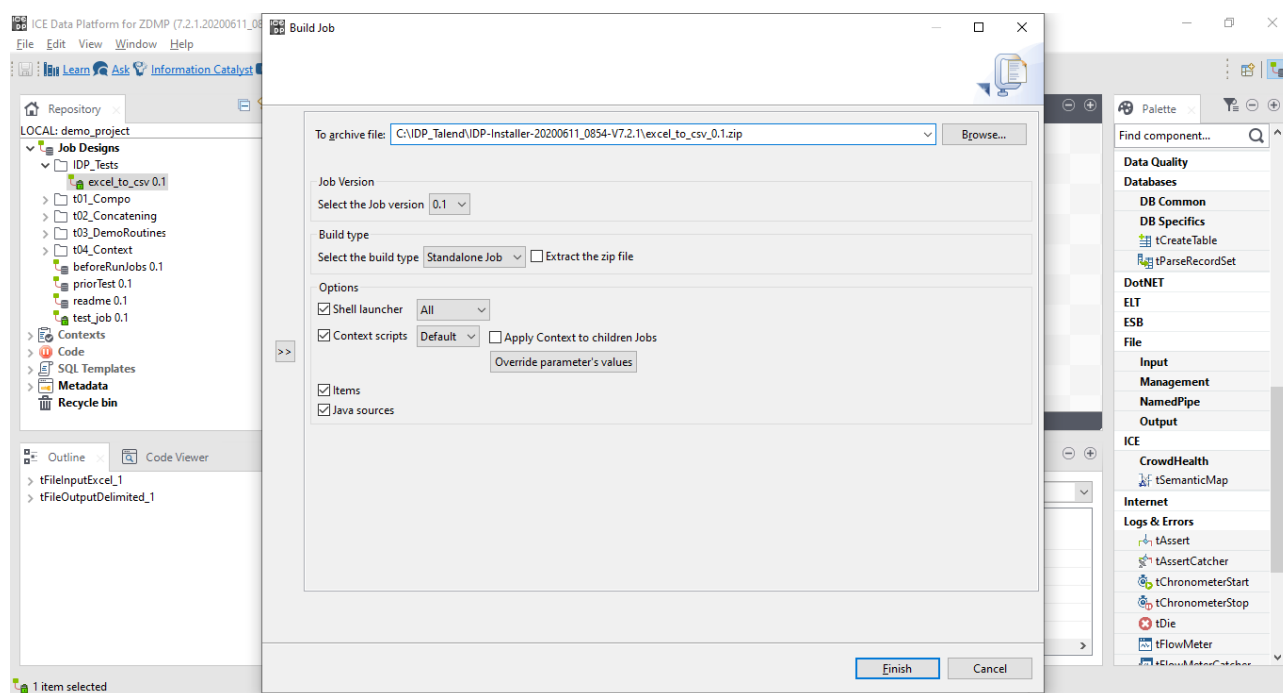


Figure 10: A Screenshot of the Build Job Pop-up Box

As shown in Figure 10, the user is given various options. From where to save built job, to more specific options, such as different shell options. One of the more important options, is the Build Type selection, where the user can get choose the build type, choosing between a standalone job, or a ZDMP Docker instance.

This Docker instance can be utilised by any of the components that use Docker, and if they do not, or the user doesn't use the Docker instances, they can choose the standalone executable.

1.6 Functional Requirements Implementation Status (M18)

The actual implementation status vis-à-vis the functional requirements implementation at M18 is provided in the annex of the D006 Technical Management Overview Report. This represents the general software status of the project and this WP/Task including information on commits and WP5-8 Risks and mitigations. Below is shown a dummy example for a security component.

Functional requirement	Description	Status	Progress	Comments
T52A013 - Issue New certificates	New client certificates are created. These certificates include the details that permit the identification of the subject (physical device, gateway or server).	Working	90%	Beta version, requires integration API with security command centre for credentials tokenization

Figure 15: Functional Requirements Progress Status

2 Conclusions

This deliverable is the first deliverable in the reporting series for T5.3 Data Harmonisation and Interoperability. The deliverables for this task, and all WP5-8 tasks, are software and are of EU type “OTHER”. The software and accompanying material (eg description, instructions) is available on the ZDMP software repository which is updated dynamically. However, for EU formal reporting purposes, this brief cover document provides a formalised pointer to the downloadable software and related content.

This deliverable should read in conjunction with the D006-D020 deliverables which document the software process/status for each WP/Task vs its content. This deliverable represents the status as at M18 with further living editions at M18 and M48 and an informal iteration at M24.



www.zdmp.eu