TechInvestLab.ru

.15926 Platform Compliance Report

30.09.2012

This compliance report is prepared according to the guidelines provided by
JORD ISO 15926 Compliance Specification (version 1.0 published 26.07.2012 at
https://www.posccaesar.org/attachment/wiki/FiatechJord/JORDComplianceSpecification.doc).

Preparation and distribution of this compliance report is sole responsibility of TechInvestLab.ru. Report reflects our own understanding and interpretation of JORD Compliance Specification. It was in no way reviewed, endorsed or approved by POSC Caesar Association in its capacity as JORD operating entity or by any other entity or person empowered by it.
Methods by which compliance was tested were determined by TechInvestLab.ru and were not in any way suggested or approved in JORD project.

1. Introduction

.15926 Platform is a name for an architecture and a set of specific interfaces and libraries to work with ISO 15926 data. It is developed to facilitate creation of semantic applications to work with ISO 15926 data in all possible ways – read, visualize, explore, search, reason, map, write, exchange, etc.

The .15926 Platform (hereafter – the Platform) is not an application for a specific business interface, but a toolset for development of applications to support different interfaces. As a development framework, the Platform contains application programming interfaces (APIs) for work with ISO 15926 data from Python programming language environment. Support of various business interfaces is realized through adapter modules (readers and writers) developed using these APIs. It is possible to access and use external mapping information in adapters.

Compliance report below is organized along Compliance Categories and Compliance Levels defined by JORD Compliance Specification. In each Compliance Category Platform's functionality and capabilities to support business interfaces are described for all Compliance Levels. These descriptions are made with particular focus on the following grades of implementation difficulty:

- functionality is supported out-of-the box;
- functionality requires development of a specific mapping adapter or expansion of a functionality of an existing adapter using existing APIs;
- functionality requires additions to .15926 Platform core and API set.

Short descriptions are also provided for capabilities of freely distributed **.15926 Editor (v1.00)** (hereafter – the Editor) – the software application built on the Platform with three major goals in mind: explore existing sources of reference data in as many formats as possible, verify reference data and engineer new reference data, including automated reference data creation through mapping from external sources.

Please refer to the original **JORD ISO 15926 Compliance Specification (**https://www.posccaesar.org/attachment/wiki/FiatechJord/JORDComplianceSpecification.doc) for full definitions of business interfaces, Compliance Categories and Compliance Levels.

2. Compliance Categories

2.1. Technical Categories

2.1.1. Semantic Modeling Category

Dictionary & Typing Level
Short-Cut Relations Level
Full Ontology Level

Business interface support is possible at any of these three Levels, depending only on the selection of business-specific set of patterns. The set of patterns have to be programmed into an adapter using Platform's software API. Manual or automated (by adapter programming in Python) selection of patterns and mapping of business information at interoperability interface are possible.

The Platform can operate with patterns defined as template signature patterns or with patterns defined by explicit Part 2 entity representations.

Template signature expansion and template lifting requires additions to the Platform's core (planned for future releases).

Full set of registration patterns (identification, typing, classification, specialization), some short-cut patterns and some full ontology patterns are supported out-of-the-box by the Editor. Pattern definitions include both template signature patterns and patterns defined by explicit Part 2 entity representations. The set can be expanded by adapter functionality expansion using existing API.

2.1.2. Referencing Technology Category

Local Naming Level

Not supported.

URI Referencing Level

Fully supported by the Platform and by the Editor. All applications developed on the Platform use URI's for entity referencing and work with on-line or local reference data library(ies) to resolve reference data items. URI generation for new entities is supported with UUIDs compliant with RFC 4122 / ITU-T X.667 / ISO/IEC 9834-8.

2.1.3. Representation Technology Category

No Explicit XML Schema Level
Explicit XML Schema Level

Development of a specific mapping adapter using existing APIs is required to work with data represented in any regular format (tabular, mark-up, XML-Shema, JSON, database schema, etc.). Sample adapters for .xlsx tables and JSON are included with the Editor.

RDF/OWL Schema Level

RDF/OWL compliant XML Schema defined in Part 8 is supported by the Platform and by the Editor out-of-the-box. Some legacy OWL representation schemas are also supported (Part 2 and Part 4 data representations, iRING template format).

2.1.4. Interface Technology Category

File Exchange Level.

Work with files with reference and project data (creation, viewing, editing) is supported by the Platform and by the Editor out-of-the-box. Files produced by the Editor are tested for seamless upload to triple store.

API or Query Level:

Work with API or support of querying other than SPARQL requires adapter development for the platform and is not supported by the Editor.

SPARQL Query Level:

SPARQL querying is supported by the Platform and by the Editor out-of-the-box. The set of queries is predefined for extraction of all data items defined by ISO 15926 (templates, Part 2 instances, template instances, annotations and metadata).

The set of constructs extracted can be expanded by existing adapter programming in Python.

2.2. Business Categories

2.2.1. Industrial Standardization Category

Local Sandbox Level
Global Industrial Level
PCA / JORD Level
ISO Level

The Category is not directly applicable to the Platform. Applications built on the Platform can work with reference data managed at any Standardization Level provided their compliance with corresponding semantic, referencing, representation and interface levels in categories listed above.

2.2.2. Payload Content Category

Generic Level
Explicit Scope Level

The Category is not directly applicable to the Platform. The Platform's modular architecture can support development of generic tools or tools with explicit scope in the particular business data environment.

2.2.3. Change-Management Meta-Data Category

Identity Only Level
Version Level
Status Level

Interoperability interface developed with the Platform's tools can facilitate exchange of change-management information on any Level, depending on the exact specification of identification data and its possible representations. Version and status information can be represented by annotations, special template instances or even by explicit Part 2 entity representation if required. A mapping for a particular method has to be developed and programmed into an adapter. The Platform's software APIs allows manual or automated change-management data recognition and delivery of version information at interoperability interface.

2.2.4. Change-Management Functionality Category

Export Level
Import Level

Application built on the Platform can have an adapter developed to extract, deliver and process change-management information independent of data scope and interface technology (subject to the compliance with interface category level above).

Seeding Level

Technologies to facilitate this Level are not part of the Platform.

Consolidation Level
Reconciliation Level

Complex specialized reasoning algorithms required to achieve these two Levels can be developed using the Platform set of APIs and Python programming language environment. Depending on the algorithms additions to the Platform core may be required.