Deliverable D1.2:
AIRM Compliance Validator

What is the contribution of this deliverable to the overall goals of BEST?

→ *Please refer to the document "BEST Results: Overview" to get an overall picture of the relationship between BEST deliverables and project objectives.*

Deliverable D1.2 is responsible for developing a prototype software application called the AIRM Compliance Validator. This application provides tool-support for automatically verifying compliance between ontological representations of the ATM Information Reference Model (AIRM) and the set of ontology modules developed in the BEST project. The AIRM Compliance Validator consists of a set of matching algorithms based on principles from ontology matching research.

Compliance verification is important both in the model development process and in the compliance assessment process. During model development, re-use of existing ATM information elements is desired. By using the AIRM Compliance Validator the modeller can compare model elements in development with the AIRM model to see if equivalent or similar elements already exist and re-use these instead of developing new and duplicate information elements. In the assessment process the "compliance assessor" can use the AIRM Compliance Validator to compare an information model claimed for compliance with the AIRM model and receive as output a compliance report that lists compliant information elements as well as non-compliant elements.

Current Status of the Deliverable

Completed and approved by funding authority (SJU).

What items does the deliverable contain?

When we talk about a "Deliverable" in BEST, we mean not only the formal document describing the work done, but also any associated technical artefacts such as software, models, ontologies, diagrams etc.

See also: "Explanatory Notes" following the table.

Item#	Brief Description	What it can be used for
Provided separately (i.e. not in the formal deliverable document)		

Item#	Brief Description	What it can be used for
1	AIRM Compliance Validator prototype application	A software application that automatically compares an input ontology (monolithic ontology or ontology module) with AIRM and produces a report of semantic correspondences identified by different types of matching algorithms.
Provided in the formal deliverable document		
2	Background knowledge (chapter 2)	Understanding what it means to be compliant with AIRM and learning the basics principles of ontology matching.
3	Technical framework applied in the development of the AIRM Compliance Validator (chapter 3.3)	Getting a deeper understanding of how the AIRM Compliance Validator can support the compliance validation process. In chapter 3.3 the three sequential processes of the AIRM Compliance Validator are explained in detail.
4	Experimental evaluation of the AIRM Compliance Validator (chapter 4)	Understanding how the AIRM Compliance Validator is evaluated in seven datasets composed of the monolithic AIRM ontology and ontology modules from AIXM and IWXXM. A comparative analysis with two state of the art ontology matching systems is conducted.
5	Interaction with the AIRM Compliance Validator (chapter 5)	Learning how to interact with the AIRM Compliance through a set of command line user interfaces.

Explanatory notes:

1. The AIRM Compliance Validator is to be considered as a prototype application demonstrating feasibility and used for experimentation, not at all an application ready for a production environment (item 1).
2. A "monolithic" ontology is one describing a *complete domain* and contained within a single model. There are usually no external references to other ontologies. Ontology "modules" are self-contained but coherent knowledge models, each responsible for describing a single, narrower knowledge domain, and typically taking part in a network of interdependent modules in order to represent a larger knowledge domain (item 1).

What details can I find in the deliverable document?

Details about what?	Reference
Definition of compliance with the AIRM	Chapter 2.1
Definition of ontologies and basic principles of ontology matching	Chapter 2.2
Overall process of the automated compliance validation process and requirements mapping	Chapter 3.1.1 (figure 2)
The three processes performed by the AIRM Compliance Validator	Chapter 3.2 (figure 3)
An overview of the matching algorithms used by the AIRM Compliance Validator	Chapter 3.2.2 (table 5)
Chart showing average quality scores for the different matching algorithms applied	Chapter 4.2.8 (figure 21) for equivalence relations. Chapter 4.2.9 (figure 22) for other semantic relations. Chapter 4.2.10 (figure 23) for combination strategies.

How can I access parts of the deliverable that are not part of the formal document?

The source code of the AIRM Compliance Validator prototype application is available from GitHub: https://github.com/sju-best-project/compliancevalidator