Semantic Interoperability for Technology-Enhanced Learning Platforms

Abels, S., Chepegin, V., Campbell, S.
TIE Holding B.V., Amsterdam, the Netherlands
{sven.abels | vadim.chepign | stuart.campbell}@tieGlobal.com

Abstract
This paper presents the results of the STASIS (www.stasis-project.net) project and applies it to the area of semantic interoperability within technology-enhanced learning platforms. Within the paper an innovative approach for creating a comprehensive application suite is introduced which allows partners to simplify the mapping process between data schemas of different e-Learning platforms. This mapping allows users to easily transfer content from one e-Learning platform to another and hence making it much easier to reuse e-Learning objects and to update to new e-Learning environments without spending serious resources on manual migration of content and user models. However, in real world situations the import and export of information into one particular standard is not always possible. On the other hand, mapping internal data formats is often very complicated because of the required deep technical understanding of formats and the syntactical structures.

The following sections describe the STASIS approach, which aims on allowing people to ease a creation of mappings between their e-Learning systems.

1. Problem Description
One of the most time consuming and common tasks in the e-Learning domain is to enter information into a system and to make it available for other people. Once, this task has been achieved, the information is available and may be used within a reasonable amount of time by new learners. However, what is a rather complicated process is to share learning objects and student models with other educational systems. In order to minimize the workload, it is desired to automate this process as much as possible. Currently there are two different approaches that allow content managers to do this:

i. An e-Learning standard may be used across different systems (e.g. Sharable Content Object Reference Model - SCORM [1]), which allows the export of data into a SCORM compliant file and the import of this data into the new system.

ii. A mapping between the data formats of the original system and those of the target system can be performed by either mapping their database information or by mapping their exported data files.

2. Existing Solutions: Standards
Within the last 10 years, some standards or de-facto standards have been established in the e-Learning domain. Among them, SCORM [1], IEEE LOM [2] and IMS Learning Resource Meta-data Specification [3] can be highlighted.

The general idea of those standards is to increase the interoperability between different e-Learning systems by providing a common baseline for describing content. This allows users to move content between multiple heterogeneous e-Learning environments by exporting and importing their data.

In some cases this works out well and users are able to seamlessly import and export data. However, there are some drawbacks of this method:
- “The good thing about standards is that there are so many of them to choose from” [4]. What makes the usage of standards problematic is that the same standard needs to be available on both e-Learning systems (source and destination) and in most cases, systems need to be able to support even the same version of a standard in order to work properly. The reality is such that there are many systems have been rolled out in different time all over a globe without knowing about each other and without sharing the same view on the problem of annotating learning objects. Consequently, their deployers and users chose incompatible formats for representing metadata for their content.
• Another problem with this approach is that e-Learning systems are interpreting the data in a different way. This leads to the problem that the content is sometimes not imported correctly or displayed in a wrong way.
• Last, but not least, the import and export functionality does not support a usage of the same versions of e-Learning objects in real time. Different e-Learning systems need to be maintained separately meaning that the data needs to be synchronized manually in case of changes.

3. Existing Solutions: Mappings

Another approach is the mapping of data models of systems that want to establish a mutual bridge. This way the syntactical structures of the systems are mapped to each other. This allows users to either import native data into the other system or even to develop a bridge between different e-Learning systems that may be used to synchronize data.

Fig 1. Syntactical Mapping between two XSD schemas

Mapping processes between schemas today focus on a syntactical approach. Tools such as Altova Mapforce\(^1\) or TIE Integrator\(^2\) are used to successively map different attributes of two different schemas to each other (see e.g. [6], [8], [10]). Figure 1 shows an example, mapping the syntactical elements of a source XML Schema file (XSD) to a destination format.

Attributes that are mapped in syntax based approaches are in most cases either mapped in a 1:1 relationship or they are mapped to each other by using various connectors such as “Methlets” for creating complex mappings. “Methlets” (sometimes called “Functions”) are elemental connectors that allow users to, for example, concatenate two attributes or to perform simple calculations such as calculating the sum of two values. A detailed description for “Methlets” and their use in syntax based schemas mappings is provided in [5].

The syntax based approach has been successfully used for years in industry and is indeed ‘the’ approach used today. It allows relatively fast mapping creation if the schema is not too complicated and the syntaxes and structures are known by those performing the mapping, which is why most mapping exercise are conducted by experts. However, limiting the mapping creation to a syntactical mapping process has some disadvantages:

**Experts needed:** First of all, real-world mappings are far from being well structured, logically named and fully linkable on a 1:1 basis. Figure 1 therefore shows a very simplified view that is almost never found in real world situations. Schemas may be expressed in many different formats and their data may be transported in XML/RDF/OWL files, database entries or other bespoke structures. What is found in real world cases is that the description of those elements is rather chaotic even when studying available documentation. For example, attributes may not be called “courseType” but “c_typ” or “CT02”. Mappings therefore have to be performed by a technical person who has schema and syntax knowledge and practical experience whereas what is required is to allow the mapping to be created by knowledge workers that are administrating or maintaining e-Learning systems. Of course easy-to-use GUIs may ease this task but nevertheless the process is rather technical involving skilled people that know both schemas and information structures.

**Error prone:** Next, mappings are often hard to understand and error-prone because people normally think in terms of the actual semantics (the concepts) instead of the syntax. For examples, educational content providers such as universities usually want to map “Java language course: Basics” to “Introduction to Java for beginners” instead of mapping element “JC_EL_4711” to “ELT0815”.

**No reuse:** Because of the complexity and the deep understanding that is necessary, mappings are typically created once ‘on demand’. They are then stored for use in the current mapping project but this knowledge is not available for reuse in the creation of future mappings especially those of other parties. The lack of reuse forces users to start from scratch with every new mapping.

**No Semantics:** Concepts, such as inheritance and logical constraints, are usually not supported by syntax based mapping tools. All semantic information is

\(^1\)http://www.altova.com
\(^2\)http://www.tieglobal.com
completely ignored although it might be a precious help when creating mappings.

4. The STASIS Approach
The STASIS project aims in addressing many of the problems outlined above and also both introduces the concept of user-driven semantics and to promote a neutral standard mapping format which can be taken advantage with existing transformation tools and technologies. This way STASIS can be classified as one of the modern user-centered ontology-based semi-automatic annotation tools. The main difference of a STASIS approach lies in a fact that it offers a usage of “folksonomy” (end-user defined categorization system) without a necessity of establishing a real common ontology prior to be able to do any mapping. Due to the shortage of space for this paper and different focus, for the comprehensive review of (semi)automatic annotation approaches and their comparison please see, e.g. [13].

In order to achieve a semantic mapping, STASIS is adopting the following process as identified in [6]:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Import Schema Information</td>
</tr>
<tr>
<td>2</td>
<td>Identify Semantic Entities</td>
</tr>
<tr>
<td>3</td>
<td>Share Information in a Semantic Space</td>
</tr>
<tr>
<td>4</td>
<td>Map Semantic Entities to each other; Generate Mapping File</td>
</tr>
</tbody>
</table>

The following sections will give more details of each step where a general description will be given (Conceptual View), the activity will be specified and the Technological Background will be summarized.

Step 1: Import Schema Information
STASIS provides implementations for importing and automatically converting syntactical schema files into a content neutral format (called STASIS Neutral Format - SNF), see fig. 3. For example, these inputs can be an XML schema specification (XSD), or a relational database schema (SQL), or even Microsoft Excel files marked-up in a specific way. The first prototype implementation supports all these formats. Other formats are planned for the next versions of the STASIS prototype implementation.

Importing schemas into STASIS is performed by a simple Import Wizard module that allows users to specify the original schema files (e.g. XSD files or SQL files) or to directly connect to a system that provides schema information (JDBC connection).

Figure 3 shows the SCORM format being imported and annotated into the STASIS prototype.

Fig 3. Representation of the SCORM format in STASIS Neutral format

Technological Background
STASIS has created a completely content neutral representation for schemas that internally is based on OWL and is referred to as the ‘STASIS Neutral Format’ (SNF) in the project. A detailed specification of the STASIS Neutral Format has been published and discussed in [7].

Step 2: Identify Semantic Entities
Instead of focusing on syntactical mappings, STASIS concentrates on identifying semantic entities and mapping those semantic pieces. For this purpose, STASIS has defined the idea of allowing users to define so-called “STASIS Semantic Entities” (SSEs). These SSEs are representing specific elements with a semantic meaning. For example, two syntactical elements coming from a SCORM file and being called SCORM::OBJECTIVE and SCORM::OBJECTIVEALIAS may be linked to one semantic entity called IMS:PURPOSE.

These semantic entities are the essence of the STASIS approach. They allow users to define anything that they believe should be a semantic element and then they connect their syntax elements (being derived from their schema files) to those semantic entities using a graphical editor. SSEs are not limited in size, for example they can be as ‘small’ as a course number value or as large as a whole learning object such as quiz although most will be more elemental.

While the example that has been given is a rather simple, STASIS supports the specification of more complex relationships between elements. For example, STASIS reuses the concept of Methlets/Functions as described in section 2.1, which allows users to connect
their syntax elements to semantic entities by using logical expressions, concatenations, etc. Those links are referred to as 'complex links' in the STASIS approach and are fully supported in the STASIS specification and partially in the prototype implementation.

All semantic elements are expressed in OWL within STASIS in order to benefit from existing libraries for ontology processing. Obviously, technological details are hidden in the background without involving users.

Step 3: Share Information in a Semantic Space

All elements that have been created by STASIS are managed in a distributed registry and repository network called the SRRN (Semantic Registry and Repository Network; see [8]). The SRRN is a cutting-edge pure Peer-to-Peer network based on webServices that allows the storage and retrieval of semantic elements in OWL format. The SRRN has been developed within two European Research projects as described in [8]. It utilizes the semantic query language SparQL, [9]. The SRRN provides a common semantic space for OWL based elements and allows users to share semantic information with each other in a scalable, fully non-centralized network.

The SRRN gives STASIS another significant advantage over traditional mapping creation tools as STASIS participants may reuse semantic entities, mappings or schema definitions. It allows STASIS to make mapping suggestions by reusing mapping information from earlier semantic links. For example, imagine two universities A and B that would like to map their e-Learning schemas in order to exchange e-Learning objects. For our knowledge this feature makes STASIS one of the few existing systems that can operate and offer an integrated view on the defragmented user models and distributed learning objects.

Similar functionality is achieved by the I-Help [11] system that employs multi-agent approach to user modeling in a peer to peer environment. Let us assume that both have conducted sharing of their learning objects with university C already in the past. In this situation, STASIS can map the semantic entities of A and B automatically because it knows about their mapping to a common schema from university C. This will become increasingly beneficial as more educational content providers begin to use STASIS and the repository turns into a significant mass of mappings. In addition to this, links of semantic entities to global concepts - as described in the last section – can be used to generate additional links automatically.

Step 4: Map Semantic Entities to each other & Generate Mapping File

Once users have performed all other steps, they may create the actual mapping between their data schema and the data schema of a partner. In order to do so, they will search and select two schemas (SNF entries in STASIS terminology). This results in a display of all semantic elements that have been defined in each SNF. STASIS allows users to map their e-Learning schema to the schema of another e-Learning system by simply connecting their Semantic Entities. This allows users to create mappings in a more natural way by considering the meaning of elements rather than their syntactical structure. Some elements might even be connected to a common third party, e.g. element from a merging ontology, or a chain of them. STASIS automatically detects if such a third party exists as discussed in step 3 of the process. STASIS performs an automatic guessing of links helping the users to perform the mapping in a semi-automatic way.

To justify the usability of STASIS approach in the e-Learning domain, two of the most popular formats for learning objects metadata, namely SCORM and IMS, were mapped. Following the steps of approach, as described above, we imported IMS and SCORM schemas into STASIS desktop application (fig. 4). Then asked STASIS to propose us links between elements, and then manually correcting the wrong links and added the missing ones. After that we could export a resulting mapping into a format that understandable by mapping engines. One of the options is to export a mapping file into OWL and let one of the existing reasoners to do the job. Currently we are working on a more production oriented way of capturing that mapping information, i.e. stream oriented processing of data for better performance and ability to map the most popular industry standards.
6. Next Steps and Conclusion
STASIS is currently available in a first beta version [12] and is based on the Eclipse RCP framework. To ensure a wide applicability of the approach, STASIS is evaluated by users with a different level of technical understanding. So far, the evaluation process is performed by companies coming from the Automotive and Furniture sectors from Spain, Italy and China. In order to do this, a total number of 15 Test Cases have been defined and evaluated. A detailed evaluation report is currently under construction and will be published at the project website in the next months. In addition to the use case based evaluation, STASIS currently provides a third beta version and which will be published for an open testing phase.

STASIS provides a new way of defining mappings between different schemas by providing an easy means of creating mappings that are based on the semantics instead of syntax. Compared to existing approaches, STASIS also fosters the reuse of information, which is usually neglected in existing mapping approaches.

STASIS implements a set of services that allow people and applications to share e-Learning schema information in a webService based network. The STASIS Desktop application allows people to actually use those services for creating semantic based mappings between schemas.

In the next phase, STASIS is focused on four main tasks:
- Improving the usability in cooperation with users,
- Providing additional import functionalities,
- Providing support for third party mapping formats (e.g. XSLT)
- Improving the process for automatic link suggestions via semantic annotations.

7. References