Dear COMMIUS Community member,

You are receiving this newsletter because you subscribed to the COMMIUS Community, an active interest group created with the aim of supporting and driving COMMIUS research and activities, composed by subjects for different reasons interested in the project, such as persons in charge of SMEs or involved in SME associations, ICT companies who could be interested in exploiting commercially COMMIUS results or researchers and experts in interoperability and collaboration issues, mainly interested in the project scientific results to support their research activities.

The goal of this newsletter is to keep you informed with the latest project advancements and results, and to elicit your collaboration in order to constantly improve our output and provide us with your useful feedback on our results.

If you wish to contact us, please feel free to write to the COMMIUS community e-mail address community@commius.eu.

This issue’s contents:

• Project News
• The COMMIUS Semantic Layer
• COMMIUS At ICT 2010
• Agenda.
What’s new

Here are some of the project achievements of the last period:

- COMMIUS has successfully undergone its third project review, held on Bruxelles on May 5\textsuperscript{th}, 2010.
- The project is approaching the end of the software integration phase; a first pilot application based on the integrated platform is already available for piloting activities in order to collect end users impressions and feedback for the final prototype development and final integration round.
- A first round of Commius Piloting events has already been held at the sites of the three project partners acting as end users partners in the three selected piloting scenarios, respectively Aitek in Savona, Italy, Fedit in Madrid and SingularLogic in Athens.
- Commius has participated with a stand in the ICT 2010 event, which took place in Brussels, 27\textsuperscript{th}-29\textsuperscript{th} Sept. 2010 (see at page 11 of this newsletter).

The COMMIUS Semantic Layer

After having illustrated COMMIUS System Interoperability concepts in the previous issue of this newsletter, we are now going to the next layer in COMMIUS architecture: The COMMIUS Semantic Interoperability layer. Thanks to this layer’s functionality, a document received by an SME via email is automatically mapped onto one of a set of user specified document types,. If this is impossible, automatic negotiations investigate the possibility of using a substitute document type, or record the information required to further align the documents received in the future.

Semantic Interoperability

In general terms Semantic interoperability signifies meaningful communication among two or more parties. Generally, it refers to a shared understanding of data, information, process models, etc., within and among organisations. As such, it underpins other kinds of interoperability between collaborating enterprises and thereby constitutes an essential prerequisite for enterprises to be truly interoperable. It is typically achieved by fixing a unique meaning (semantics) through a commitment to a common ontology.

Currently, semantic interoperability is typically achieved “off-line”, chiefly through adherence to standards governing both the structure of the information being exchanged and the manner in which this information should be processed (for example RosettaNet and ebXML); or in particular, through commitment to a shared ontology, which is then used as metadata.
Indeed, many proponents of the Semantic Web seek a universal medium for information exchange based upon XML syntax. This has given rise to such standards as the Resource Description Framework (RDF) and its elaboration in RDF Schema or the Web Ontology Language (OWL); and is also reflected in the vision of Semantic Web Services and related standards, namely, the Web Service Modelling Ontology (WSMO), the Web Service Modelling Language (WSML) or OWL-S. The predominant use of ontology to foster semantic interoperability is reflected by the numerous research efforts, and software tool development and support in this area.

In response to approaches to ontological modelling, such as those cited above, e.g. RDF, a number of tools for ontology editing, storage, querying and reasoning are now available. These include several semantic frameworks for accessing and manipulating documents in OWL, RDF and RDFS. The most prominent semantic framework is Jena. It provides a wide range of functionality through its APIs. Ontologies can be created using ontology editors, such as Protégé. This is an open source development environment for ontologies and knowledge-based systems which was developed at Stanford University (USA). OWL plug-ins for Protégé supports the editing of OWL ontologies and the ongoing CO-ODE project provides valuable support for user modelling in OWL.

Automated annotation of the Web documents is a key challenge for the realisation of the Semantic Web. Web documents are structured, but this structure is typically understandable only for humans, while the semantic web requires them to be machine understandable. This is typically achieved by annotating them with concepts from a formal ontology. Several annotation protocols exist, including Annotea, Rubby and RDF annotation. Annotation solutions can be manual, such as CREAM and Magpie, or semi-automatic based on natural language processing (NLP), a document structure analysis or a learning approach (which requires training sets or supervision). Moreover, there exist pattern-based semi-automatic solutions, such as PANKOW and C-PANKOW, SemTag or pattern based approach Ontea.

In COMMIUS we will exploit the pattern-based solutions supplied by the COMMIUS System Layer, which uses metadata to semantically enrich messages, communications and also documents exchanged (See the previous issue of this newsletter for a more detailed description of the COMMIUS System Interoperability Layer).

The creation of domain specific ontologies and in particular as extensions of agreed, "upper" ontologies achieves semantic interoperability through semantic agreement. Developing domain specific ontologies is clearly not "zero-cost". Moreover, this agreement on a common semantic model means that existing approaches to semantic interoperability are typically centralised. Naturally, many have recognised that the use of a single ontology is untenable in a distributed environment and this has led to research in "ontology alignment", for example, by mapping from one to another or by creating some "meta-structure" to relate the ontologies. In either case, one must “learn” the local structures and create appropriate mapping rules or "meta-structures". Again, this is clearly not "zero cost".

An alternative approach to this problem has been investigated as part of the MaBE and Crosswork European research projects. It advocates abstracting from matters of syntax to focus on the information passed within a concept. When a concept is passed to a recipient with a local ontology, the approach allows a range of options, starting from only retaining the subset of the concepts overall information which that recipient will understand; to a decision by the recipient to learn the complete new concept because of the future utility of this concept. This permits actors/trading partners to meaningfully communicate without the need for potentially expensive ontology agreement or alignment.
COMMIUS Semantic Interoperability Basic Approach

Traditional approaches to interoperability have adopted the upper ontology approaches discussed in the previous section – the set of documents which the companies will use to communicate and their detailed structure are both agreed upon and fixed. Such approaches are very accurate and work very well for large companies in stable partnerships but also incur considerable costs in their initial configuration and do not work well for more dynamic groupings.

The aim of the semantic interoperability layer within COMMIUS is to provide interoperability for SMEs. Such companies are typically engaged in small projects with dynamically changing partners, and indeed cannot rely on the documents they receive being in any kind of formal feedback. Indeed a crucial overall goal of the COMMIUS project is to produce a piece of software which a single company can easily install and use, and so the system can make no assumptions regarding the structure of incoming documents.

The principal challenges faced by the layer are therefore to:

1) Allow users of the system to specify a set of internal documents which they find significant enough to wish to detect and treat automatically,

2) Interpret incoming, free text, documents and identify which, if any, of these document types they most closely correspond to.

In order to do this the semantic interoperability layer uses the following features:

- The software comes with a set of ‘core components’ – these represent the basic information tokens that are expected to be found within business documents.
- When a user first installs COMMIUS they define both the set of internal document types they consider of interest and the set of core components they would expect to find in a ‘typical’ example of each document type.
- When an e-mail is received text mining techniques are used in order to detect the set of core components present within that e-mail.
- The set of core components detected is compared to the definitions of the internal document types and a set of scores produced, together with a recommendation concerning which document type the e-mail most closely matches. The user is free to override this recommendation.

Classification algorithms which, over time, learn from the sets of core components in the e-mails assigned to each class are also under development. In addition to providing a basic idea of how to interpret each e-mail this data is crucial for the operation of the process layer. Finally the detection of the core components present within each e-mail enables the COMMIUS system to automatically fetch additional information of use to people interpreting e-mails.

A concrete example of this can be seen within the ‘catch a contact’ use case where the text mining package within COMMIUS detects the presence of otherwise of new contacts within incoming e-mails, the classification algorithms inform the system of whether the incoming document contains such a contact and finally the system uses this information to decide whether or not to insert a new contact.
COMMIUS Architecture’s Semantic Interoperability Layer

The primary aim of COMMIUS Semantic Interoperability Layer is to establish a framework to allow collaborating SMEs to quickly and painlessly achieve semantic alignment. As a first step, we compared the UN CEFACT “Core Components” standard with the document types used by our case study partners, and also with important business ontology standards. The findings were used to identify a core of semantic descriptors which will be used as “building blocks” to characterise relevant business document types. A first characterisation of documents from our case study partners provided the initial core ontology of business documents. Together, these two aspects constitute an extensible “Semantic Core”.

The second step has been the design of mechanisms to support each new partner in applying this Semantic Core to its own documents, extending and customising it where appropriate. This mechanism includes a “feedback loop” to enable an automatic enrichment of the core ontology with those specific concepts and instances which are found to be common among a number of partners. This is linked with (semi-) automatic semantic annotation and appropriate results from the System Interoperability Layer, for example, the automatic analysis of email messages and existing partner documents. Work done also include the development of a visualisation tool to support users in mapping the structure of their existing documents and databases to elements of the core ontology. This is meant to support both the classification of existing documents onto an existing core and inform appropriate re-categorisations (of the core) based on specific domain concepts and instances.

The third step consisted in developing semantic alignment protocols between trading partners. Upon receiving a message containing an unknown document type, the ontology alignment mechanism first attempts to automatically find a match for this in the ontology of the receiving partner. “Close enough” matches are handled by adding a “note” containing the additional information. If this fails, the mechanism initiates a “hidden” discussion with the sender about either using a more abstract document (one from the core ontology, for example), provided the two parties are happy to tolerate a certain loss of information for this particular transaction. However, if this partnership is of a particular importance for one of the parties, they may be prepared to “learn” or internalise a new document type, automatically adjusting their document repositories for the new document.
COMMIUS architecture’s Semantic Interoperability Layer contains several basic software components, briefly listed below. A detailed discussion of these components and their operation can be found within the project public architecture deliverable D3.1.2, available on the project web site (http://www.commius.eu/).

- The **Semantic Manager** functions as the Semantic Interoperability Layer interface. Its basic functionality consists of managing all the semantic interoperability, from receiving a Document for initial semantic annotation to initiating any needed semantic negotiation. It provides the layer output as well consisting of Mapped Documents.

- The **Semantic Annotator** parses Documents to identify valuable information in terms of CC for further mapping to internal Document Types. It uses a combined approach of machine-learning and pattern matching.

- The **Document Interpreter** reads the set of Core Components found in an Annotated Document and finds an appropriate correspondent to a Document Type. This is done with the aid of the Semantic Core and the Document Type Repository which are managed by the Document Type Manager.

- The **Semantic Negotiator** is in charge of reaching common agreements with any other Semantic Negotiator in another COMMIUS system when the Document Interpreter cannot find a close mapping to a Document Type. Such agreements consist on finding a new common set of Core Components to represent the same information.

- The **Document Type Manager** functions as an interface to manipulate both the Semantic Core and the Document Type Repository. It provides operations for the creation, addition, updating, and removal of Document Types. It updates the Semantic Core by allowing the addition of new Document Instances.

- The **Semantic Core** consists of an FCA lattice representing the Core Components relationships existing among Document Instances.

- The **Document Types Repository** is a collection of Document Types available to a COMMIUS system.

The figure in the following page shows the Semantic Interoperability Layer components along with their dependencies and interactions. As can be appreciated, the Semantic Manager functions as a layer interface receiving Incoming Documents from the System Interoperability Layer and providing Mapped Documents (and possibly additional information yet to be defined) to the Process Interoperability Layer for supporting business processes (which will be described in the next issue of this newsletter). Moreover, it controls the execution order of the subordinate components. First the Semantic Annotator, then the Document Interpreter and finally, if needed, the Semantic Negotiator. The first two components modify the initial input which is an Incoming Document transforming it into an Annotated Document and a Mapped Document, respectively, which is the final output of the Semantic Interoperability Layer. The Semantic Negotiator works on a partial Mapped Document only when a concept negotiation is required.

The Document Type Manager provides an interface to manipulate both the Semantic Core and the Document Type Repository. It supports the Document Interpreter for mapping Annotated Documents into Document Types as well as the Semantic Negotiator for updating both the Semantic Core and existing
Document Types according to the negotiation results. Finally, the Visual Mapping Tool can also operate on the Semantic Annotator, Document Interpreter, and Document Type Manager for assisting the user in manual Local Document classification.

Semantic Annotation and the Extensible Semantic Core

The purpose of the Semantic Interoperability Layer within COMMIUS is to attempt to correctly interpret the information contained within documents received by an SME. Semantic annotation is a way of assigning a “meaning” to the set of terms extracted from the emails. Semantic annotations are essential in the process of understanding email content. Thus it is important to correctly annotate as many entities as possible in the email text. With semantic annotation we understand metadata assigned to a text entity. Such metadata semantically describe entities and thus semantic annotation provides meaning to them.

Manual annotation is a kind of semantic annotation created by a human. However the use of manual annotation is sensitive to errors due to factors such as familiarity with the domain, amount of training, motivation and complexity. Manual annotation is also quite expensive process and rarely considers different perspectives of the same text by means of different taxonomies. Another problem with manual
annotation is the volume of information, which for emails is quite extensive. Therefore the manual semantic annotation has led to a knowledge acquisition bottleneck.

To overcome such a bottleneck, semi-automatic annotation of emails is implemented in COMMIUS. Fully automated annotation methods are not yet possible with the current technologies, and thus all existing annotation systems rely on human intervention. In contrast, semi-automatic annotation provides the scalability needed to address large volume of emails and reduces the burden of manual annotation. The currently available semantic annotation platforms can be classified based on the type of annotation methods used. The main two categories are pattern-based annotation and machine learning-based annotation.

**Pattern-based annotation** platforms perform pattern discovery or have patterns manually defined. An initial set of entities is defined and the corpus is scanned to find the patterns with existing entities. New entities are discovered, along with new patterns. This process continues recursively until no more entities are discovered, or the user stops the process. Annotations can also be generated by manual rules to find entities in the text. In the domain of email annotation, the regularity of the patterns is as obvious as it is to frequently find email addresses, dates and time patterns, etc., and therefore the initial set of annotations can be easily accomplished by such approaches. However, pattern-based methods as such are not sufficient as there is significant amount of further information to be extracted and used.

**Machine learning-based** platforms use two methods: probability and induction. Probabilistic platforms use statistical models to predict the locations of entities within text. Induction methods use the machine learning induction algorithms to induce the initial set of models and then refine them based on further samples. Machine-learning based platforms usually require learning and testing corpus in order to train the model. In the email domain this has shown to be quite complicated due to the existence of only a few corpora suitable for annotation.

Semantic annotation in COMMIUS is based on a multi-strategy platform, i.e. a combination of pattern and machine learning methods. The COMMIUS approach focuses on a simple approach based on regular expression patterns with possibility to integrate existing advanced semantic annotation approaches based on NLP or machine learning techniques.

Within this context, the purpose of COMMIUS Semantic Core is to provide a language which is suitable for describing the basic units of information contained within both the documents used internally by SMEs and those documents that they typically receive by email. Such basic units are represented by a set of identified semantic building blocks which are called Core Components.

The COMMIUS Semantic Core is built upon the UN/CEFACT CC (Core Components) standard. This standard describes and specifies a new approach to the well-understood problem of the lack of information interoperability between applications in the e-business arena. A Core Component (CC) is defined as “a building block for the creation of a semantically correct and meaningful information exchange package. It contains only the information pieces necessary to describe a specific concept”. The CC standard presents a methodology for developing a common set of semantic building blocks that represent the general types of business data in use today and provides for the creation of new business vocabularies as well as restructuring the existing ones. This standard also provides a way to identify, capture and maximise the re-use of business information to support and enhance information interoperability across multiple business situations. For example, the CC technical specification can be employed whenever business information is being shared or exchanged among and between enterprises, governmental agencies, and/or other organisations in an open and worldwide environment.

A key aspect of the approach to semantic interoperability involves the removal of document structure and the translation of meta-data (e.g. key-value pairs) into descriptors deriving from the UN/CEFACT CC,
essentially standardised semantic building blocks to promote information interoperability. The removal of structure and translation into standardised semantic building blocks produces an abstract version of the original incoming document. With it, it is possible to align or map the document to a structure specific to the company, which in COMMIUS terminology is called a Document Type. In the end, such an alignment is the output of the Semantic Interoperability Layer which provides support to the Process Interoperability Layer or any other module that requires semantic matching.

From an implementation point of view, the Semantic Core can be roughly defined as a lattice structure representing how the semantic building blocks included in documents are interrelated one another providing an idea of similarity among documents. This functions as a repository where the internal semantic building blocks represent documents are interrelated emphasising their similarities between one document to another. The extension mechanisms on the other hand aims to find document categories common to a set of collaborating partners using COMMIUS. The current status of this mechanism is the definition of the approach, i.e. to use emergent semantics over a network.

A running COMMIUS system may receive different types of Documents e.g. special requests, purchase order, invoices, quotations, etc., and the Semantic Interoperability Layer should be capable of correctly interpreting those Incoming Documents. Starting from the CC standard, the COMMIUS Semantic Core is built from real documents from our case study partners. Such documents were used to construct the lattice of concepts by means of Formal Concept Analysis techniques.

In the following an example of a Document interpreted using the Core Components is shown: The left part of the figure below shows an example of an Incoming Document which is annotated upon reception and some Core Component related information is highlighted. The figure shows a yellow text highlighter only for illustration purposes, the annotation output is not expected to be delivered in this way between the Semantic Interoperability Layer components. However, the Annotated Document is expected to be represented in ebXML using the Core Components tags. The right part of the figure shows the Core Components extracted from the email in ebXML syntax.

The ebXML document is then used in the document alignment process by the Document Interpreter aided by the Document Type Manager. After the completion of the document alignment process the system stores a record of the set of core components found within this instance together with a note of which internal document type it was recognised as. Some of the classification algorithms under consideration within COMMIUS use this information to improve their classifications over time.
One of the goals of COMMIUS in relation to Semantic Interoperability was the development of a Visual Data Mapping Tool (VDMT), or just Visual Mapping Tool for short, with which target users can easily work with documents of different Document Types, manipulating Document Types and instances, and mapping information to properties of documents. This tool is of particular importance in allowing users to easily define the set of internal document types which they wish to treat as significant internally.

The provided visual mapping tool comprises of four discrete areas (also called panes); the left pane, the central pane, the right pane and finally the bottom pane. Each pane visualises different kind of information. The following figure depicts all panes together. The left pane is the Core Component visualisation pane. When the visual mapping tool is started an XML file that contains a document instance is visualised. This document instance contains an email that has already been processed by the Semantic Annotation. The tool “consumes” the Annotation output and automatically selects the proper Core Component that this document instance refers to. The selected Core Components are represented in a hierarchical tree structure. The then user can (de-)select items of the hierarchy. Any selected leaf Component automatically selects the parent Component.

All selections are accompanied by a visualised representation in the initial email. In this email, different identified Core Components are highlighted with different colours. The tool user is also able to edit/remove or add new annotations accomplished in the central pane in which the user highlights a sentence (for example) then clicks on the related Core Component in the hierarchy.

Additionally, the right pane shows the current classification of the document instance in the central pane. Next to each Document Type, a percentage is shown representing the similarity the document instance has to each of the Document Types. Moreover, this pane will eventually allow the user to browse Document Instances which will be shown in the central pane.

Finally, Document Types can be easily created with the help of the bottom pane. Document Type creation uses a Document Instance as the “starting template” which, as explained above, already contains assigned Core Components. These are the Components to be included in the Document Type. If the user is happy with those Components, then a simply click on the button “Save New Document Type” will show
a small field for entering the name of the new Document Type. Then after saving it, a new Document Type will appear in the right pane.

However, if the user prefers to use additional Core Component for the Document Type, then by a simple GUI Drag & Drop functionality the user can choose those Components and drop them to the middle box in the bottom pane. By clicking on “Save New Document Type” button the Document Type will be saved as explained above.

A more detailed description of the visual mapping tool can be found in the project public deliverable D5.3, available on the project web site (http://www.commius.eu/)
COMMIUS has participated with a stand in the SME Village at the ICT 2010 event, which took place in Brussels from 27 to 29, September, 2010.

The biennial ICT event is Europe's premier forum on research and innovation through information and communication technologies - "ICT". The event consists of conference and networking programmes and an accompanying exhibition.

Key theme for ICT 2010 was "ICT for sustainable growth in a low-carbon economy", encompassing digital solutions for a sustainable recovery and improved energy efficiency. Another central topic was "ICT for and by the citizen", the benefits of involving ICT users in the innovation process.

The ICT 2010 exhibition was an opportunity to explore what is happening in current areas of research and to find potential future collaborators. Researchers from all parts of Europe and beyond present their latest work in upstream ICT innovation as well as close-to-market projects. The response was overwhelming from research institutes, ICT companies, universities, public authorities and non-profit organisations.

for more information on the event: http://ec.europa.eu/information_society/events/ict/2010/index_en.htm

---

**Agenda**

Here are some of the events that are foreseen in the next period:

- The project is approaching is final phase. COMMIUS will terminate at the end of January, 2011. The final project review has been scheduled for February, 16th, 2011.
- After the completion of the first piloting phase, development and integration are now aimed at the completion of the final COMMIUS prototype, which will be used as basis for the final round of piloting events in the last months of the project.
COMMIUS is a Specific Targeted Research Project (STREP) part-funded by the EU under the 7th Framework Program, ICT priority, Grant Agreement No. 2138