

Project no. FP6-028038

Palette

Pedagogically sustained Adaptive LEarning Through the exploitation of Tacit and
Explicit knowledge

Integrated Project

Technology-enhanced learning

D.IMP.01 – First Guidelines for Development

Due date of deliverable: 31 July 2006
Actual submission date: 30 October 2006

Start date of project: February 01, 2006

Duration: 36 months

Organisation name of lead contractor for this deliverable: **EPFL**

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
R	Public	PU

Keywords List: Guidelines for PALETTE Implementation, integration of services,
Interoperability, Usability, Acceptability

Responsible partner: Christine Vanoirbeek, EPFL

MODIFICATION CONTROL			
Version	Date	Status	Modifications made by
1	18-07-2006	Draft	Christine Vanoirbeek, Bernadette Charlier
2	29-08-2006	Draft	Vincent Quint, Olivier Corby, Christina E. Evangelou, George Gkotsis, Nikos Karousos, Manolis Tzagarakis, Yannick Naudet, Alain Vagner, Manfred Kuenzel
3	29-09-2006	Ready for internal evaluation	Yannick Naudet, Alain Vagner, Adil El Ghali, Christina E. Evangelou, George Gkotsis, Nikos Karousos, Manolis Tzagarakis
4	26-10-2006	Ready for final evaluation	Bernadette Charlier, Amaury Daele, France Henri, Christine Vanoirbeek, Yannick Naudet, Alain Vagner, Adil El Ghali, Christina E. Evangelou, George Gkotsis, Nikos Karousos, Manolis Tzagarakis
5	30-10-2006	Final	Adil El Ghali, Christine Vanoirbeek

Deliverable manager

- Christine Vanoirbeek (EPFL)

List of Contributors

- Bernadette Charlier (UNIFR), Olivier Corby (INRIA), Amaury Daele (UNIFR), Adil El Ghali (INRIA), Christina E. Evangelou (CTI), George Gkotsis (CTI), France Henri (TELUQ), Nikos Karousos (CTI), Manfred Kuenzel (UNIFR), Yannick Naudet (CRP-HT), Vincent Quint (INRIA), Manolis Tzagarakis (CTI), Alain Vagner (CRP-HT), Christine Vanoirbeek (EPFL)

List of Evaluators

- Vincent Quint (INRIA), Etienne Vandeput (ULG)

Summary

This document presents the initial agreement between partners on the adoption of relevant standards, the choice of main technical options and, the specification of procedures to be followed for the setting up of the PALETTE services repository.

A classification of standards is proposed in order to share understanding on the role of these standards and, provide a reference framework for anchoring the further adoption of additional standards during the project life. A set of initial technical recommendations is provided to set up the PALETTE services repository; it aims at guiding a consistent development and evolution of tools to be provided in the framework of the project. A methodological approach is described, currently based on the building up of three cross cutting teams aiming at defining scenarios of use and precisising the required interoperability level between services.

1. Introduction

The effectiveness of activities performed in a Community of Practice (CoP) relies on its possibilities to deal with different types of information and knowledge, make use of various applications and, accommodate several organizational and technical environments; an underlying goal being the enhancement of CoPs activities by learning on achievement of their own goals.

The PALETTE project contributes to this goal by providing the CoPs with a set of *extensible* and *interoperable* services that will enhance (i) the exchange and reuse of information - data and documents- , (ii) the sharing of skills and knowledge and, (iii) the collaboration between CoPs members.

The purpose of this document is the establishment of first guidelines for the setting-up, implementation and evolution of the PALETTE services. It addresses two main goals:

1. Heighten the project members awareness of the benefits to be taken out from the use of standards;
2. Provide first guidelines to sustain the development of interoperable and extensible tools that suit scenario of use within CoPs.

This document presents the initial agreement between partners on the adoption of relevant standards, the choice of main technical options and, the specification of procedures to be followed for achieving the goals of the project. It is a “living” document that will be updated incrementally all along the project duration, in order to progressively integrate the evolving requirements of Communities of Practice (CoPs) and, anticipate the impact of technology insertion.

It is organized in the following way: section 2 reminds the objectives of the workpackage 5 and, emphasizes the associated issues to be addressed, section 3 proposes a classification of standards and justify their role in the project , section 4 summarizes the first technical recommendations for development of services, section 5 prefixes our methodological approach and, finally, a conclusion prepares the ground for the further revisions of this document.

2. Global considerations

The workpackage 5 objective is the implementation of the framework that will integrate and, make available the palette of services to CoPs. Two main challenging issues are to be faced in this respect:

- In order to enhance the sharing of information and knowledge, the set of services proposed by Palette have to be potentially connected through scenarios of use; *interoperability* issues need to be carefully taken into consideration when designing and implementing the services;
- Services will be used by CoPs that may differ considerably in terms of culture and organization; the acceptability of proposed tools will bigly depend on their *usability* by CoPs members, i.e. in which manner the services accommodate the practices in use and further ones.

2.1 Interoperability issues

Agreeing on the targeted level of interoperability of services is a crucial issue addressed in the PALETTE project; it significantly affects the design and implementation of services.

Let us consider, as an introductory example, a number of CoP members debating about a specific problem (using an argumentation tool). A typical behaviour, in such situation, is to search additional information in relation with the discussed issue (using, for instance, search engines or refereeing to classified bookmarks) in order to sustain an argumentation in the discussion. We can say that a very basic interoperability level would be achieved by the fact that the final user is able to use several tools through an unified interface (such as a web browser), is able to cut&paste information he estimates relevant, is able to update its bookmarks list accordingly, etc. However, supporting the user in such a task could be improved by

1. anchoring functionalities in the argumentation tool that help her/him to identify more efficiently additional information (avoiding noise and potentially unreliable sources of information because such functionalities would be based on the use of ontologies in the specific domain of the CoP)
2. providing the user with enhanced cut&paste facilities (avoiding a tedious work for reformatting reused information in a target framework)
3. keeping a trace of the user's actions, to face further situations (memorizing awareness of the process, capitalizing related assets)
4. facilitating the task of the user in context (avoiding the overload of bookmarking, downloading and classifying documents, etc.)

Such a level of interoperability could be achieved if a number of services may, indeed, be connected. Point (1) requires interoperability between the argumentation tool and services provided to deal with ontologies, point (2) may be addressed by using functionalities offered by a tool that facilitates reuse of information and, finally, point (3) and (4) may be tackled by functionalities provided by awareness tools.

In this sense, interoperability of services is a notion that encompasses both technological and operational capabilities.

Technological capabilities mainly concern the development of services that may be connected to other ones and, may exchange pieces of information. Interoperability issues are thoroughly explored in the software engineering domain and, it is commonly accepted that the adoption of a number of standards as well as the explicit specification of APIs (Application Program Interfaces) bring a significant contribution to the technical issues.

The difficulties of the task are, on one hand, the identification and agreement on standards useful to attain the required level of interoperability and, on the other hand, the specification of end users' requirements in order to precise the functionalities of services and subsequently deduce the APIs specification.

Operational capabilities address the issue of enabling services to operate *effectively* together. This issue is much more complex because the targeted operational environments are "moving grounds" whose characteristics highly depend on several factors such as the domain of activity of a CoP, its members' profile, specific organizational aspects, technical constraints, etc.

The PALETTE project aims at providing services to be adopted by a wide range of CoPs; in this purpose, the palette of services to be provided need to be generic and, accommodate features, that may drastically differ from one Cop to another.

The activity domain of a CoP has an influence on the nature of information to be dealt with (for instance, CoPs in the engineering domain clearly require appropriate management of mathematical data).

The CoPs members' profile may be related to CoP categories (the potential adoption of new tools by CoPs members depends on the maturity of the CoP ; a well established CoP, for instance, making use of common set of tools for a while, is likely to be more reluctant to the use of new tools if the effort to combine them with existing ones is too important)

CoPs are potentially organized in very different ways, ranging from a network of quite unstructured collectivities to organized entities that are issuing guidelines to sustain activities and communication framework within the CoP (constraints to achieve a given interoperability level will be different from case to case).

Finally, technical constraints may be indirectly imposed by CoPs and, affect the implementation of services (members of a CoP, for instance, may be particularly mobile and, thus requiring adapted functionalities that covers features such as the use of mobile devices, the use of a limited bandwidth, etc.)

Taking into account the targeted level of interoperability, we may summarize the concept of PALETTE services in the following way:

"PALETTE services to be provided to Communities of Practice (CoPs) are dedicated to a common global objective: enhance the effectiveness of collaborative learning process of CoPs. They have to be connected in such a way as maximizing exchange, reuse and sharing of information and knowledge between members and outside the CoP. They have to sustain collaborative interactions in accordance with the evolution of CoPs members practices and environments."

2.2 Usability issues

CoPs have been existing for many years; they are already using a wide variety of heterogeneous tools (ranging from basic email communications, forum, chats to the use of sophisticated content management systems or learning web-based platforms) and, they are rapidly integrating the use of emerging ones (such as blogs, wiki, etc.). It is absolutely clear that new tools will only be adopted by CoPs members if they satisfy two fundamental conditions: first, they have to bring evidence that they will significantly facilitate the tasks performed in a CoP - providing a real added value to an existing situation - secondly, they are to be usable by CoPs' members, i.e. accommodate the culture of users by proposing appropriate user interfaces and making transparent any technological features for the end users.

Requirements also relate to the work organization of CoPs; they deal with different levels, for example:

- At the level of the individual members: what are his or her own competence and routines related with the realization of a specific task (i.e. the management of own mails or documents) ? ; Which changes in his or her work routines would be needed? Is the activity of the COPs a priority for him or her? ; What are his or her individual stakes ?
- At the level of the animators: What is the added value brought by the introduction of new services in his/her CoP environment? Which power does the animator have to intervene with the CoP practice?
- At the level of the organization: which support the organization would be ready to offer to support the adoption of new practices in a CoP ? Which training ? Which human resources? What would be, for an organization, the added value of the adoption of new practices?

The *process of ensuring integration of services is not linear*; it has to be supported by a communication framework that allows re-assessment and corrective actions to sustain the targeted level of interoperability. Achieving a mutual understanding of interoperability and usability issues relies on a participative approach involving all the partners, to be adopted during the overall duration of the project (this point is described in section 5). Additionally, jointly addressing technical and operational issues is supported through the *training activities* provided via WP8 in order to enhance communication and share a common vision.

3. Standards and their role

It has been unanimously recognized by the PALETTE partners that the use of XML based technologies may significantly help for reaching the goals of the project. However, the family range of XML based standards is wide; this section proposes a classification whose goal is twofold: (i) to provide a synthetic overview of standards classes in order to achieve a global understanding, between partners, about the roles of standards in the scope of the project and (ii) to establish a reference classification framework for identifying the role of further adoption of standards.

Figure 1 illustrates the proposed classification, that distinguishes 6 levels:

1. Core technologies; they gathers the standards that participate in the foundation of the XML approach
2. Standards addressing non textual information

(6)	WSDL (Web Services)	
(5)	LOM	IMS
		SCORM
(4)	RDF	RDFS
(3)	Domain specific vocabularies	
(2)	SVG	SMIL
		MathML
(1)	XHTML XML (DTD) – XML Schema CONTENT	CSS XSLT PRESENTATION

Figure 1: Classification of standards

3. Standards addressing the definition of domain specific vocabularies
4. Standards addressing knowledge representation
5. Standards addressing the definition of domain specific metadata
6. Standards addressing the implementation of web services

Section 3.1 to 3.6 shortly describe each level of the classification, emphasizing the major benefits to be taken out of the standards it includes. Section 3.7 summarizes the expected advantages of standards adoption in the scope of the PALETTE project.

3.1 Core technologies

Extended Markup Language (XML)

The XML language (<http://www.w3.org/XML/>) is the foundation of the approach. Issuing from its ancestor SGML (Standard Generalized Markup Language - ISO 8879), XML proposes a character based document interchange format that promotes two fundamental concepts:

1. A clear dissociation between *content* and *presentation*

This allows the rendering of a same document on various devices (not limited to printers or screens) in multiple layouts, without affecting the document content.

2. The rigorous specification of a document type through the DTD (Document Type Definition) mechanism

This capability allows the checking - by a program - that a document conforms to a precise definition (it helps authors to write documents that respect guidelines but also enhance the processing operations to be performed on a document).

A feature of XML (in comparison with SGML) is its evolution towards the representation and management of "data-centric" information in addition to "document-centric" information. The associated XML Schema language (<http://www.w3.org/XML/Schema>), an alternative to DTD, participates in this trend.

eXtensible Stylesheet Language - Transformations (XSLT)

The XSLT language (<http://www.w3.org/TR/xslt>) provides a mean to generate, from an XML representation, documents and/or data that can be dealt with by various applications, even in proprietary formats (such as Web browsers, text processing systems, spreadsheets, etc.). Despite its name, that includes the word "Stylesheet", XSLT offers much more than this; it includes powerful transformation capabilities - such as sorting and computing facilities - that allows the generation of customized information to be used in multiple purposes.

eXtensible HyperText Markup Language (XHTML)

The XHTML language (<http://www.w3.org/MarkUp/>) represents the new generation of the popular HTML language; as stated by the W3C "expected benefits include: reduced authoring costs, an improved match to database & workflow applications, a modular solution to the increasingly disparate capabilities of browsers, and the ability to cleanly integrate HTML with other XML applications".

XHTML documents, contrarily to most of HTML documents, may be easily processed by applications because they are valid XML documents. The "class" attribute introduced since the version 4 of HTML encourages the users (or developers of applications) to produce documents integrating a structural markup that facilitates the separation between content and presentations well as the reusability of documents in several contexts.

Cascading Style Sheets (CSS)

The CSS language (<http://www.w3.org/Style/CSS/>), essentially used in combination with the structural markup approach proposed by XHTML, provides a powerful mechanism for defining several stylesheets associated to a same content. This capability may be used in various purposes: to factorize a style reflecting the corporate identity of an organization, to accommodate the technical users' environment (for example, the screen resolution), to render documents on specific devices (either for making information accessible to everyone - notably those with disabilities - or for addressing specific needs, such as those of mobile workers).

3.2 Standards addressing non textual information

The "Markup" approach promoted by SGML was at the origin mostly dedicated to textual documents. Nowadays, however, a lot of documents or data available on the Internet include graphics and multimedia components. A number of standards, based on XML, came into being to

address the representation and management of such elements. Taking into account the CoPs' domain of activities, the following ones have been considered of interest in the framework of the PALETTE project.

Scalable Vector Graphics (SVG)

SVG (<http://www.w3.org/Graphics/SVG/About>) is a language for describing two-dimensional graphics and graphical applications in XML. Like MathML for formulas, it allows drawings to be included in Web pages. As opposed to pictures, SVG drawings are represented as an organized collection of geometric shapes with a number of graphical attributes (line style, line width, color, etc.). This allows SVG documents to be displayed with the best quality at very different scales. In addition, each shape, including characters, can be animated. This is especially helpful when it comes to explain for instance how mechanical devices work or how various phenomena evolve over time.

Synchronized Multimedia Integration Language (SMIL)

SMIL (<http://www.w3.org/AudioVideo/>) enables simple authoring of interactive audiovisual presentations. The SMIL language is typically used for "rich media"/multimedia presentations which integrate streaming audio and video with images, text or any other media type. Its main role is to express how these media objects are displayed on a screen and how they are synchronized during presentation. Media objects are Web resources that are referred from a SMIL document, like images in a HTML page. It is then possible to build new multimedia documents that combine existing material available on the Web. SMIL is flexible enough to represent many different types of interactive multimedia documents and has a key role to play in education.

Mathematical Markup Language (MathML)

MathML (<http://www.w3.org/Math/>) is an XML markup language for describing mathematics. It provides a foundation for the inclusion of mathematical expressions in Web pages and other XML documents. MathML is intended to facilitate the use and re-use of mathematical and scientific content, and for other applications such as computer algebra systems, print typesetting, and voice synthesis. MathML can be used to encode both the presentation of mathematical notation for high-quality visual display, and mathematical content, for applications where the semantics plays more of a key role such as scientific software or voice synthesis.

3.3 Standards addressing the definition of domain specific vocabularies

The adoption a XML technologies by communities of common interest lead sometimes to the establishment of common vocabularies to facilitate sharing of information between their members. Typical examples are : in the publishing domain, the ISO standard 12083; "This International Standard presents a reference document type definition which facilitates the authoring, interchange and archiving of a variety of publications" or, more recently, initiatives under way to agree on a common representation of patients' records in the medicine area.

For the time being, no similar standard has been identified in the scope of the PALETTE project in relation with CoPs activities. However, with a view of being consistent and, provide an exhaustive panorama of standardization levels, it seems important to us to mention this level explicitly.

3.4 Standards addressing knowledge representation

The use of XML and, related core technologies, have also been adopted by stakeholders of knowledge management to represent and manage metadata. The following standards are currently considered of interest in the PALETTE project.

Resource Description Language (RDF)

RDF (<http://www.w3.org/RDF/>) is a language for representing information about resources in the World Wide Web. It is particularly intended for representing metadata about Web resources. RDF is intended for situations in which this information needs to be processed by applications, rather than being only displayed to people.

RDF is based on the idea of identifying things using Web identifiers (called URIs: Uniform Resource Identifiers - <http://www.w3.org/Addressing/>), and describing resources in terms of simple properties and property values. This enables RDF to represent simple statements about resources as a graph of nodes and arcs representing the resources, and their properties and values. (from W3C RDF Primer).

Hence RDF relies on a simple triple model: resource, property, value. An RDF base is a graph of triples. The language is based on URI: property name is an URI. Resources are also denoted by URI.

RDF also has an XML syntax as exchange format. There is a W3C working draft in preparation for designing conventions, called RDFa, to integrate RDF statements within XHTML.

RDF Vocabulary Description Language 1.0: RDF Schema (RDFS)

RDFS (<http://www.w3.org/TR/rdf-schema/>) is a simple ontology language which provides mechanisms for describing groups of related resources and the relationships between these resources. RDF Schema vocabulary descriptions are written in RDF.

RDFS enables to define classes of resources and properties relating resources. Classes are related by the subsumption relation (here called subClassOf). A class may be subClassOf several classes.

RDFS enables to define a hierarchy of properties, organized with the subPropertyOf relation, where a property may have several subproperties. Properties exist on their own, they are autonomous entities. There can be properties without classes as in the Dublin Core.

Properties have a signature made of domain and range. A property can have several domains (respectively ranges) which are considered in conjunction.

Type inference is special in RDF/S. If a resource has a property then the resource has as type the domain of the property (i.e. It is the converse of object-oriented languages where an object has the properties of its class). Furthermore, a resource that is instance of a class is not required to possess all properties that are legitimate for this class.

RDFS proposes properties to document RDF/S resources with natural language, multilingual labels and definitions. The properties are: label, comment, seeAlso, isDefinedBy.

3.5 Standards addressing the definition of domain specific metadata

The importance of sharing and reusing metadata of common interest through communities is a preoccupation that has been identified for a while. As an example, a significant initiative in this domain is the Dublin Core Metadata Initiative (<http://dublincore.org/>). It addresses the needs of digital libraries and proposes, in this purpose, the definition of specialized metadata vocabularies based on the use of RDF.

In the learning domain, a number of standards exist in order to specify metadata about learning objects.

Learning Object Metadata (LOM)

LOM (<http://ltsc.ieee.org/wg12/>) is a data model encoded in XML used to describe a learning object and similar digital resources used to support learning. The purpose of learning object metadata is to support the reusability of learning objects, to aid discoverability, and to facilitate their interoperability, usually in the context of online learning management systems (LMS).

Instructional Management Systems (IMS)

IMS (<http://www.imsglobal.org/>) formal name is IMS Global Learning Consortium, Inc., also referred to as IMS/GLC. IMS is concerned with establishing interoperability for learning systems and learning content and the enterprise integration of these capabilities. IMS specifications address requirements in a wide range of learning contexts, including of course K-12 schools and corporate and government training. Digital Repositories Specification, IMS ePortfolio Specification, IMS Learner Information Package Specification and Learning Design Specification, are among the IMS specifications.

Sharable Courseware Object Reference Model (SCORM)

SCORM (<http://www.adlnet.gov/Scorm/index.cfm>) is a suite of technical standards that enable web-based learning systems to find, import, share, reuse, and export learning content in a standardized way. It is written for toolmakers who build LMS and learning content authoring tools.

For the time being, this category of standards is not considered as essential in the framework of the PALETTE project. However, they are mentioned and, will be taken into consideration if some CoPs explicitly express a willingness to be provided with services compliant to this category of standards.

3.6 Standards addressing the implementation of Palette services

Developing networked applications, relying on XML based technologies, is currently promoted through the use of Web Services. Roughly speaking, a Web Service may be considered as a software component (in the sense of object oriented applications development). In comparison with approaches that do not make use of XML technologies, a Web Service has the following features:

- The service is available on the Internet (accessible via an URI);
- It is published on a repository; it may be searched (typically through the use of UDDI services (Universal Description Discovery & Integration - <http://www.uddi.org/> -)

- Its description is provided through the use of WSDL (Web Service Description Language - <http://www.w3.org/TR/2006/CR-wsdl20-primer-20060327/>), an XML based language that specifies in a formal way:
 - the generic model of data manipulated by the service (based on the use of XML Schema language);
 - an abstract description of the functionalities implemented by the service (signature of the methods)
 - a concrete technical description about the way to invoke the methods (the supported protocols, etc.)

3.7 Summary

As a summary, the major reasons of using XML based standards and related technologies in the framework of the PALETTE project are the following.

XML provides a standardized exchange format which is adapted for representing information - either document or data oriented - , as well as associated metadata. Adopting an XML representation clearly facilitates the interchange, reuse and management of information between services; it permits more elaborate processing on information and, contributes to achieve the targeted level of interoperability.

Using an homogeneous approach allows us to take benefit from a number of existing standard-based and open source software to deal with information manipulated by PALETTE services, in multiple purposes (validation, styling, transformations, etc.); it lessens system design and development time.

Specifying explicitly the format of XML information (on the basis of XML schema) is helpful for many reasons: it provides a formal, human understandable, documentation about information models - a way to agree on information - and, facilitates the development of tools based on the use of web services.

Finally, we may anticipate an increasing adoption of XML, by existing and upcoming applications, to import/export data. This is an important point, addressing third party programs and, their ability to process and understand data. An XML based approach may take advantage of converters that permit versatile use of data and information, also allowing backward and forward compatibility.

4. Initial technical recommendations

The development of PALETTE services - the tools intended to be used by the CoPs final users - rely on an extensive use of XML based standards (for the reasons mentioned in section 3). However, it appears essential to precise the notion of service in terms of implementation, to guide the development of tools and guarantee an homogeneous use of relevant standards to reach the targeted level of interoperability.

Globally, we may distinguish between "web services-based applications" and "web services-compliant applications". The first one correspond to applications making intensive use of web services; they are based on a typical Service Oriented Application (SOA) approach, while the second one correspond to applications that have a web services layer allowing to call external web services, or to publish some functionalities as web services.

"Services" and "Tools" are both software components, which can be defined in the following way:

- A service provides at least either an interface for machine access (e.g. WSDL description), or an interface for human access, a Graphical User Interface (GUI)
- A service answers or contributes to a CoP's need
- A service can be composed with other services to form more complex services or tools. Two distinct composition can be performed:
 - a composition at the processing level: interactions, communications between services
 - a composition at the GUI level
- A tool may have the possibility to integrate other services (tools or not), as add-ons or plug-ins
- A service without a GUI can integrate other services without a GUI
- A (complex) tool may propose some of its functionalities as services (tools or not)

In this perspective and, at the current state of the project, the following set of technical recommendations have been identified in order to facilitate the implementation and integration of PALETTE services.

Use of namespaces

The namespaces mechanism (<http://www.w3.org/TR/2006/REC-xml-names-20060816/>), standardized by the W3C, provides a way to achieve modularity when combining the use of various XML vocabularies. The PALETTE partners need to define and agree on a common strategy to specify the use of namespaces in the framework of the project.

Use of XML schema for the specification of information manipulated by Web Services

The objective of the PALETTE project is to develop tools making use of Web Services. It implies that the information managed by Web Services will be described through the use of XML schema. The purpose of this guideline is twofold: (i) provide a mutual understanding of the information to all the partners and (ii) facilitate the development of tools (such as facilitating the validation of information exchanged by Web Services).

RDF/S related guidelines

Since RDF/S permit many equivalent XML syntax (see Example 1), we recommend to agree on the same syntax, to facilitate the development of XSLT stylesheets.

Example 1: Two equivalent syntax for an RDF/S class

```
<rdfs:Class rdf:about="Actor" rdfs:label="Actor">
  <rdfs:comment>relates to the actors</rdfs:comment>
</rdfs:Class>
<rdfs:Class rdf:ID="Actor">
  <rdfs:label xml:lang="en">Actor</rdfs:label>
  <rdfs:comment>relates to the actors</rdfs:comment>
</rdfs:Class>
```

It is also recommended to define a root class (resp. properties) for each ontology module as well as a PALETTE root class (resp. properties) of all ontologies, in order to facilitate the development of requests.

Finally we recommend to assign a label and a comment (a definition) at least in English for each class and property, and to use XML Schema datatypes.

Conformance to standards

The information produced by the Palette tools have to be in accordance with the standards mentioned in section 3; as an example, an XHTML document and an associated CSS will be validated through the tools provided by W3C in this purpose.

Establishment of a GUI chart

A graphical chart need to be defined by partners, specifying guidelines for graphical style and ergonomic; agreement on a systematic use of XHTML and associated CSS will contribute to the establishment of consistent interfaces for users.

5. Methodological approach

The PALETTE R&D methodology aims namely to enable incrementally the objectives of interoperability as they have been defined in section 2. Through the participative design process, almost two services and their integration into the actual virtual environment and practices of each COP will be considered.

According to this methodology described in the deliverable D.PAR.01, steps 2 and 3 developers, observers and delegates of CoPs will work together to establish "a first version of joint and negotiated scenarios" and prepare the functional specifications related to these scenarios.

To achieve these goals, it has been agreed to use the modeling language MOT (Modélisation par Objets Typés - Typed-Objects Modeling - <http://www.licef.teluq.quebec.ca/gp/fr/productions/mot.htm>). MOT is a graphical knowledge-modeling methodology allowing to represent different types of general knowledge with links between them: concepts, procedures and principles. The MOT+ software, developed by the LICEF lab at the Têluq in Quebec, is able to graphically represent these knowledge and links as well as to export the graphics in different standardized formats: such as XML or IMS-LD. More information about MOT can be found in the deliverable D.PAR.01.

This task (see figure 2) will contribute to:

The specification and development tasks in WP2, 3 and 4.

- The design of services and customizable scenarios and their validation in WP1.3 and WP.1.4
- The functional specification of services and scenarios and their compliance with standards in WP.5.1 and WP.5.3

The lead, the participating members, the output and the internal organization of each PALETTE WP tasks as described in the description of work of the project (DoW) are maintained. The

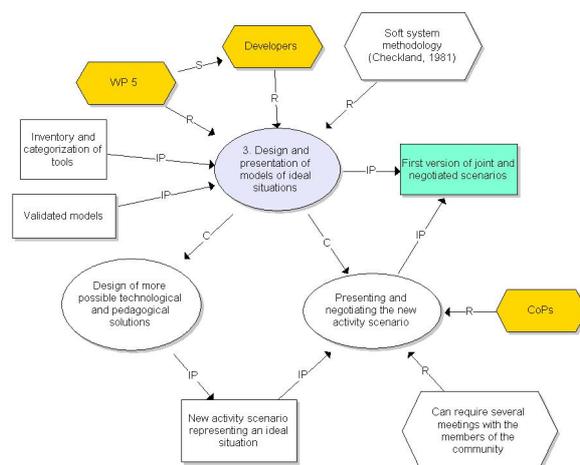


Figure 2: participative design methodology

developer-Cop working groups are cross cutting. Three cross cutting working groups (named Team-A, Team-B and Team-C) have been set-up to facilitate the flow of crucial information and exchange between the R and D WP (1,2,3,4 and 5) and the coordination of activities.

Two main issues are addressed through this approach:

- Interoperability: this approach will allow to address interoperability between services. In each group two service-developing teams approach the Cops together. Surely all pairs of services are possible, but some profit especially well from each other in the initial state of scenario.
- Specific and more general scenarios: each group should address two Cops and negotiate two specific scenarios. With the integration of both a more general scenario can also be created.

The developer-Cop working groups will produce documents enabling them to communicate with the CoPs, i.e. :

1. Models of the COP's actual activities (called validated models in fig 1. as they should be validated by the COP) expressed with the MOT language.
2. Models of "ideal" situations integrating into the "validated model" of the actual practice of a COP the added value of one or two PALETTE services. This model suggests an adaptation of the actual practices towards a more suitable practice ("ideal"). Again this model will be validated with the COP
3. Detailed use cases with screen prototypes
4. Prototypes and mock ups
5. CoP based scenarios describing the cumulated activities of a CoP (i) as a general scenario depicting all the services – including the PALETTE services - used within the CoP's activities and (ii) as a specific scenario explaining all main activity individually such as the creation of documents.
6. Functional specification, written in a Cop readable manner.
7. A description of the interoperability (i) of at minimum two services of Palette and (ii) interoperability with actual COP virtual environment.
8. A concept for each cop describing how the services are introduced:

- technical requirements (practicability)
- training and information needs (ability)
- the value of the services (willingness, commitment)

6. Conclusion

The current guidelines for the development of the PALETTE services mainly aimed at giving a comprehensive overview, for all the partners, of relevant standards to be used for achieving the goals of the project. A classification has been proposed that emphasizes the role of each category of standard, taken into account the targeted level of interoperability, in order to both agree on the use of these standards and, provide a reference framework for discussing and anchoring the further adoption of additional standards during the project life. As underlined in section 2, the guidelines for development are not limited to technical preoccupations but, also address the methodological approach to be adopted through a participative design methodology, described in section 5.

These initial considerations constitute a first step for the establishment of the PALETTE services repository; the following points have been agreed between the partners:

1. The tools to be ultimately provided to CoPs members are partially built on existing pieces of software; a definition of PALETTE service is provided at section 4, that precises the technical approaches to be adopted for accommodating the variety of developments under way;
2. The targeted level of interoperability between services is an important issue; it is directed by the work performed by the cross cutting teams whose goal is to sketch scenario of use and, precise the role of standards in this respect;
3. An initial repository will be set up that integrates the current state of developments; services being described in a synthetic way (short description, keywords, useful links, etc.)

The next steps will be to guide further developments and refine guidelines about the evolution of the PALETTE services repository; they address several layers of implementation ranging from the physical level (the transport layer) to an abstract level dealing with applications functionalities (addressing the orchestration of web services).

In this respect, the following issues are currently under discussion and will be precised in further revisions of this document.

Composition of services

Services will be connected for the purpose of scenarios of use within CoPs; the use of standards for composition of Web Services like WS-CDL (Web Services Choreography Description Language - <http://www.w3.org/TR/2004/WD-ws-cdl-10-20041217/>) and WS-BPEL (Web Services Business Process Execution Language - http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel) will be discussed.

If automatic composition is required, semantic services descriptions and ontologies could be used, such as OWL-S (Semantic Markup for Web Services - <http://www.w3.org/Submission/OWL-S/>)

Discovery mechanism

As a starting point, the services will be described in a very simple way (basically relying on the use of keywords and simple UDDI services). However, the PALETTE goal is to propose generic and more sophisticated methods for advanced services discovery. It requires semantic description of services, as well as an adequate query language, SPARQL (Query language for RDF - <http://www.w3.org/TR/rdf-sparql-query/>) could be used in this purpose.