CRITICAL SYSTEM ENGINEERING ACCELERATION

CRYSTAL Global Glossary
D102.050
# DOCUMENT INFORMATION

<table>
<thead>
<tr>
<th>Project</th>
<th>CRYSTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grant Agreement No.</td>
<td>ARTEMIS-2012-1-332830</td>
</tr>
<tr>
<td>Deliverable Title</td>
<td>CRYSTAL public aerospace use case Development Report – V2</td>
</tr>
<tr>
<td>Deliverable No.</td>
<td>D102.050</td>
</tr>
<tr>
<td>Dissemination Level</td>
<td>PU</td>
</tr>
<tr>
<td>Nature</td>
<td>R</td>
</tr>
<tr>
<td>Document Version</td>
<td>V0.6</td>
</tr>
<tr>
<td>Date</td>
<td>2015-05-08</td>
</tr>
<tr>
<td>Contact</td>
<td>Eugenio Brusa</td>
</tr>
<tr>
<td>Organization</td>
<td>POLITO – Politecnico di Torino, Italy</td>
</tr>
<tr>
<td>Phone</td>
<td>+39 011 090 6900</td>
</tr>
<tr>
<td>E-Mail</td>
<td><a href="mailto:eugenio.brusa@polito.it">eugenio.brusa@polito.it</a></td>
</tr>
</tbody>
</table>
### AUTHORS TABLE

<table>
<thead>
<tr>
<th>Name</th>
<th>Company</th>
<th>E-Mail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eugenio Brusa</td>
<td>POLITO</td>
<td><a href="mailto:eugenio.brusa@polito.it">eugenio.brusa@polito.it</a></td>
</tr>
<tr>
<td>Maurizio Morisio</td>
<td>POLITO</td>
<td><a href="mailto:maurizio.morisio@polito.it">maurizio.morisio@polito.it</a></td>
</tr>
<tr>
<td>Francesco De Vita</td>
<td>POLITO</td>
<td><a href="mailto:francesco.devita@polito.it">francesco.devita@polito.it</a></td>
</tr>
<tr>
<td>Ambra Cala</td>
<td>POLITO</td>
<td><a href="mailto:ambra.cala@polito.it">ambra.cala@polito.it</a></td>
</tr>
<tr>
<td>Davide Ferretto</td>
<td>POLITO</td>
<td><a href="mailto:davide.ferretto@polito.it">davide.ferretto@polito.it</a></td>
</tr>
<tr>
<td>Oscar Rodríguez Rocha</td>
<td>POLITO</td>
<td><a href="mailto:oscar.rodriguezrocha@polito.it">oscar.rodriguezrocha@polito.it</a></td>
</tr>
<tr>
<td>Federico Della Croce</td>
<td>POLITO</td>
<td><a href="mailto:federico.dellacroce@polito.it">federico.dellacroce@polito.it</a></td>
</tr>
<tr>
<td>Christian El Salloum</td>
<td>AVL</td>
<td><a href="mailto:christian.elsalloum@avl.com">christian.elsalloum@avl.com</a></td>
</tr>
<tr>
<td>Annemarie Hamedler</td>
<td>AVL</td>
<td><a href="mailto:annemarie.hamedler@avl.com">annemarie.hamedler@avl.com</a></td>
</tr>
<tr>
<td>Sytze Kalisvaart (contributor)</td>
<td>TNO</td>
<td><a href="mailto:sytze.kalisvaart@tno.nl">sytze.kalisvaart@tno.nl</a></td>
</tr>
<tr>
<td>Anna Todino (contributor)</td>
<td>ALA</td>
<td><a href="mailto:anna.todino@alenia.it">anna.todino@alenia.it</a></td>
</tr>
<tr>
<td>Frédéric Loiret (contributor)</td>
<td>OFFIS</td>
<td><a href="mailto:frederic.loiret@offis.de">frederic.loiret@offis.de</a></td>
</tr>
</tbody>
</table>

### REVIEW TABLE

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Reviewer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Version | Nature | Date       | Page |
--------|--------|------------|------|
V0.6    | R      | 2015-05-14 | 3    |
CHANGE HISTORY

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Reason for Change</th>
<th>Pages Affected</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2015-03-27</td>
<td>First draft</td>
<td>all</td>
</tr>
<tr>
<td>0.2</td>
<td>2015-04-12</td>
<td>Update the Glossary in Ch.6</td>
<td>29-44</td>
</tr>
<tr>
<td>0.3</td>
<td>2015-04-16</td>
<td>Add IOS related entries in Ch.5 and Ch.6</td>
<td>26-48</td>
</tr>
<tr>
<td>0.4</td>
<td>2015-04-28</td>
<td>General update of the document</td>
<td>all</td>
</tr>
<tr>
<td>0.5</td>
<td>2015-05-04</td>
<td>Add contributors in the authors table</td>
<td>3</td>
</tr>
<tr>
<td>0.6</td>
<td>2015-05-14</td>
<td>Merge feedbacks and suggestions</td>
<td>all</td>
</tr>
</tbody>
</table>
Contents

1 TASK DESCRIPTION AND GOAL ............................................................................................................ 6
  1.1 GLOSSARY OF THE CRYSTAL PROJECT ............................................................................. 6
  1.2 RELATIONSHIP TO OTHER CRYSTAL DOCUMENTS ......................................................... 7
2 THE GLOSSARY COMPOSITION WORKFLOW ............................................................................. 8
  2.1 PRELIMINARY PROPOSAL .................................................................................................. 8
  2.2 THE NEED OF A TOOL FOR DISCUSSION ....................................................................... 10
3 FIRST VERSION OF THE GLOSSARY ..................................................................................... 11
  3.1 GLOSSARY ON LINE ......................................................................................................... 11
  3.2 REVIEW PROCESS ........................................................................................................... 11
  3.3 CONTENTS OF THE GLOSSARY .................................................................................... 12
4 THE COLLABORATIVE WORK TOOL: SEMANTIC MEDIAWIKI ........................................... 18
  4.1 TOOLS AND WORKFLOWS ................................................................................................ 18
  4.2 COLLABORATIVE EDITING ............................................................................................... 18
  4.3 SEMANTIC WEB AND LINKED DATA ................................................................................ 21
  4.4 PROPOSED SOLUTION ..................................................................................................... 22
  LAMP ARCHITECTURE .................................................................................................................. 22
  LAMP COMPONENTS .................................................................................................................. 22
  MEDIAWIKI .................................................................................................................................. 22
  ADVANTAGES OF MEDIAWIKI ................................................................................................. 23
  MEDIAWIKI OPEN SOURCE EXTENSIONS ............................................................................. 24
  SEMANTIC MEDIAWIKI ............................................................................................................. 24
  4.5 IMPLEMENTATION ............................................................................................................ 26
5 ASSESSMENT OF THE GLOSSARY COMPOSITION WORKFLOW ........................................ 28
  5.1 CRITICAL ISSUES OF THE FIRST PROPOSAL .................................................................. 28
  5.2 GLOSSARY DIRECTORY ON THE CRYSTAL SHARE POINT ......................................... 28
  5.3 REVIEW PROCESS ............................................................................................................. 29
1 Task description and goal

1.1 Glossary of the CRYSTAL project

The CRYSTAL project and its activities are going to be widely disseminated on national and European level towards all identified stakeholders and public authorities, therefore a strong focus is set on exploitation of results to ensure industrial take-up. In this context the objective is to propose a cross-domain ontology in order to build a common language which represents the reference within the project. Thus a critical element is the CRYSTAL Global Glossary, being originally conceived as an update from the existing Global Glossaries assessed in some previous projects like CESAR and MBAT. In the CRYSTAL Glossary a specific novelty is appreciated as the Health Care domain is included.

The CRYSTAL Global Glossary looks like a product of the whole activity of the CRYSTAL Project and needs the contribution of all the CRYSTAL Partners, organized in Teams, Units, Work packages (in the following simply WPs).

It is worthy noticing that the CRYSTAL Global Glossary fits the requirements of:

- providing a list of terms with their definitions to create a common language among the Partners, being suitably defined, accepted and shared. This result should be significantly documented to be a reference within the project frame and for future activities;
- proposing a cross-domain ontology being focused on the artefacts produced by the application of different engineering methods;
- updating and suitably applying some global glossaries already composed within the frame of previous projects (as in CESAR and MBAT) or being worldwide recognized as a product of a preliminary standardization like in case of Glossaries proposed by INCOSE, IEEE, or other already available in the technical and scientific Community.

During the CRYSTAL project meeting held in Paris on October 2014 the Polito was formally designed as a coordinator of activities related to the CRYSTAL Global Glossary composition, aiming to collect the contributions coming from all CRYSTAL partners. Main duties to carry out on behalf of the Technical Board were:

- organizing the submission of proposed Glossary terms from all the WPs teams and more in general from the CRYSTAL Partners;
- a preliminary classification of the submitted terms;
- a management of the Glossary review, through a collaborative work among the WPs teams and together with the Technical Board.

This approach suggested to investigate two main issues about the methodology applied to perform these tasks:

- creation of a collaborative tool to open the discussion among the Partners online;
- management of the review process to define whether:
  - each term should be included into the Glossary,
  - it could be actually used across the domains,
  - definitions related to the proposed terms were complete and were accepted by all the domains.
The workflow of the review process was therefore tailored to have a suitable feedback from the WPs and the lightest protocol to be implemented. A hierarchy of the acceptance operation was identified preliminarily by assuming that the collaborative work among the WPs leaders, being coordinated by the Polito team, could provide the Technical Board of the CRYSTAL Project a list of proposed terms, to be evaluated and accepted, through an assessment of the Glossary during the CRYSTAL Project development.

To perform this task a suitable tool for a collaborative work should be used. This need motivated a preliminary screening of some tools already available for a remote discussion of the Glossary contents, through the Share Point of the CRYSTAL Project. A preliminary analysis of the Semantic MediaWiki environment and of its properties was performed, to investigate the benefits of an eventual use during the proposal and review process.

### 1.2 Relationship to other CRYSTAL Documents

The CRYSTAL Global Glossary will be a reference for the development of contents of the WPs deliverables, within all the domains foreseen by the project. A preliminary composition of the Glossary requires to be performed by using the definitions and the terms identified by each domain within each WP. Necessarily the composition of the CRYSTAL Global Glossary will take advantage from an iterative approach which starts from the daily operations of each WP, since teams will propose some typical terms and definitions to be either included or left according to a review process. After that partners of the CRYSTAL project shared some proposed definitions and terms, the review will start by a preliminary evaluation of the Technical Board being based on the results of the activities of some specialized teams working within the frame of the project like those dealing with IOS, RTP and PB as it will be described in the following sections.

To compose a standard Glossary, a reference to interoperability specifications is necessary and a connection with WP 6.1 and WP 6.2 is strictly required.

According to those remarks, this deliverable refers to some documents, being related to the above-mentioned work-packages, as well as any already existing CRYSTAL domain glossary or ontology concept lists.

In this document are presented:

- the assessment of the approach followed for the preliminary composition of the Glossary
- the collection of proposed terms with related attributes
- the results about the investigation upon the Semantic MediaWiki tools and the structure of the Glossary section of the Share Point
2 The Glossary composition workflow

2.1 Preliminary proposal

A possible workflow to perform the aforementioned task was preliminary proposed as it is described in this section. However, some critical issues for starting a fruitful discussion among all the CRYSTAL partners immediately arose, thus suggesting to resort to the solution discussed in Section 5.1. In practice two steps are foreseen, a preliminary review process is performed to define terms typically related to the CRYSTAL project as this deliverable shows, then an open discussion among the CRYSTAL partners will start to complete the Glossary and to assess the definitions.

According to the first workflow proposed to compose the Glossary, as is shown in Error! Reference source not found., a collection of some glossaries already present in the literature was foreseen to have:

- a recognized set of references;
- a preliminary list of terms;
- an example of classification;
- a preliminary indication about the interpretation of different domains about the glossary contents.

Some examples of glossaries are depicted in the above figure, coming from the InCoSE, the joint SEBok (SERC-INCOSE-IEEE), from the CESAR project, but even more than those above cited could be found in the literature. In particular, the Glossary proposed by AFIS (French Association of Systems Engineering) was considered as an example for some interesting contributions concerning the domains covered and the

Figure 2-1: Possible workflow for the Glossary composition (Phase 1)
proposed classification, but other and similar glossaries could be added. All those should compose the direct reference package for the activity of Glossary composition.

The main goal identified within the CRYSTAL project was never to create a new Glossary as a result of a selection of terms included inside those references, being eventually merged with some newly proposed words. In fact the CRYSTAL Glossary was conceived as a lighter and shorter list of concepts, aimed to introduce some typical new terms and concepts being frequently used within the frame of CRYSTAL by all the domains, to cope the need of agreeing their meaning among all the Partners. In addition the Glossary should include some concepts already present in previous ones, being associated within the CRYSTAL documents to an updated or additional interpretation that could fit the needs of some domain. In this case the Glossary is aimed to highlight any eventual difference among domains about the use and the interpretation of the concepts as they appear inside the delivered documents of the WPs.

From this point of view instead of a top–down approach, which could provide the terminology from the references down to each team working inside the WPs a bottom–up approach was highly preferred. Each WP could realize during the CRYSTAL project development what are the most frequently used and significant concepts and terms which could be applied to all the domains and propose their inclusion inside the Glossary. Obviously this structure made extremely important the information coming from all the Partners, being organized in working groups for each WP and the review of the Technical Board, while the POLITO provides a contribution as a Partner of some WPs, but in the Glossary composition is a sort of collector of the proposed terms to be analyzed. That's why in Figure 2-1 the POLITO Unit is cited as a hub the knowledge, at least for the activity above described.

As Figure 2-2 shows a second step (phase 2) was foreseen. Once that terms were received from the working teams and groups, a preliminary glossary could be assembled. It looks crucial associating some attributes and contents to open a public discussion about their interpretation, coherence with the aims of the Glossary. A preliminary proposal included the term, the class (word, method, artefact...), the domain in which it was proposed, the definition and a recognized reference. After that the discussion should follow, among the CRYSTAL Partners, as, for instance, it is depicted in Figure 2-3.
2.2 The need of a tool for discussion

Due to the number of Partners, the workflow above identified could be easily implemented by resorting to some dedicated tool tailored for an open debate in which resources are accessible to the whole Community (in this case the CRYSTAL teams), versions of contributions are traceable and are easily shared among the participants. This motivation required the preliminary investigation described in Section 4 about the available tools which in addition may provide a semantic approach.

The tool should fit some requirements.
- It should be compatible with the CRYSTAL Project sharepoint, being user friendly and somehow compatible with a fast implementation by each Partner
- Remote connection should be fast, easily accessible and open to several users simultaneously
- A common database, being controllable by the CRYSTAL chair, should be used.

In the following sections a first version of the Glossary as it came out from the first review of definitions proposed by the CRYSTAL Partners is presented, then the tool to be eventually used for the collaborative work for the assessment of the CRYSTAL Global Glossary is described.
3 First version of the Glossary

3.1 Glossary on line

Several CRYSTAL Partners proposed a list of terms, which were collected in a dedicated webspace on the share point. This source represents the database used for the fulfilment of a first review. During the latter it was realized that the first collection included many terms (up to 170). Some looked either too generic or too specific for the purpose of the CRYSTAL Glossary, whose aim is to be project–oriented. Due to this reason some of proposed terms did not satisfy the requirements of the CRYSTAL Global Glossary.

The Glossary on the website looks like in Fig.3–1. Nevertheless, it is open to the entries of all of CRYSTAL Partners, therefore it might include more/less terms than those herein considered for a preliminary step of the assessment. For practical reasons this deliverable does not include the whole collection of terms stored on the AVL SharePoint dedicated space “Glossary”.

3.2 Review process

The first review process was accomplished regrouping the submitted terms in different status categories (defined in Section 5.2 of this document). It was evident that several terms were too generic and poorly tailored for the CRYSTAL Project, although they are currently and widely used within the Systems Engineering. After a first review, a screening was performed, in agreement with the Technical Board, to select those more suitable for the CRYSTAL Glossary as it looks in Section 3.3.

In the meanwhile many terms coming from the activity of the IOS, RTP and Platform Builder teams and related specifications were proposed for a straight inclusion inside the CRYSTAL Glossary.

As a consequence of this review process a first outcome is proposed in Section 3.3. In practice collection of proposed terms was filtered to avoid terms either too specific or too generic, which don’t satisfy the Glossary requirements. Only terms tailored for the CRYSTAL Project contents were assumed to be suitable and truly required by Glossary. A particular attention was given to the entries related to IOS, RTP and Platform Builder, being main tasks of the CRYSTAL Project. List proposed in Section 3.3 does not represent the final version of the CRYSTAL Global Glossary, since a final assessment and acceptance of terms will be further performed, in agreement with the Technical Board. Status declared in the contents of Section 3.3 is either “Proposed” or “Relevant for Glossary”, generally speaking all the terms included were considered suitable for this goal, since they are typically valid cross domains and very often used. Status “Proposed” indicates that definition should be further discussed and completed or a selection among multiple entries has to be performed. Status “Relevant for Glossary” indicates that the term is widely used by different partners within documents, thus requiring a full agreement about its definition.
3.3 Contents of the Glossary

A collection of terms tailored for the CRYSTAL Project contents are listed in present section as they came out from the first review process. The following items are part of the collection available on the AVL SharePoint “Glossary”.

**Activities Catalogue**

**Definition**: It is a container, to collect in a formal way all identified Activities within an engineering domain.

**Source**: WP-602 Technical Team

**Status**: Proposed

**Activity**

**Definition**: A set of actions that consume time and resources and whose performance is necessary to achieve, or contribute to, the realization of one or more outcomes

**Source**: ISO/IEC 15288

**Status**: Proposed

**Brick**

**Definition**: A technology brick as defined in CRYSTAL is a software tool/product, a software component to build a software tool/product, a systems engineering methodology, an interface, a standard, or means for establishing interoperability that is needed for the efficient development of safety-critical embedded systems.

**Source**: DOW amendment 1 page 15

**Status**: Proposed

**CRYSTAL IOS Lifecycle Extension**

**Definition**: The set of extensions to existing OSLC specifications proposed by the CRYSTAL consortium for supporting interoperability scenarios from the CRYSTAL Use Cases.

**Source**: CRYSTAL deliverable D601.022

**Status**: Proposed

**CRYSTAL Ontology**

**Definition**: The activities covered by the CRYSTAL Ontology workgroup (encompassing the Ontology Work Packages on Aerospace, Automotive, Healthcare and Rail). This workgroup assesses:

- domain-agnostic interoperability concerns (e.g., on Safety/Risk Management, Simulation, Digital Mockup, Variability Management, User Management),
- and domain-specific standards and glossaries (e.g., coming from the Aerospace industrial domain: ISO DIS 10303-233, ISO 10303-239 PLCS, Behavioural Digital Aircraft, from the Automotive domain: ASAM-ODS, AUTOSAR, EAST-ADL, FMI, from the Healthcare domain: FDA. Part 820, DICOM GSDF, GMDN, ICPS, IEC 60601-1, IEC 60601-2, IEC 62304, IEC 62366, and from the Rail domain (IOP) that can be used as inputs of the CRYSTAL IOS Lifecycle Extension.

**Source**: CRYSTAL deliverable D601.022

**Status**: Proposed

Version | Nature | Date          | Page  
--------|--------|---------------|-------
V0.6    | R      | 2015-05-14    | 12 of 30
**EM Catalogue**

**Definition:** It is a container to collect in a formal way all identified Engineering Methods within an engineering domain.

**Source:** WP-602 Technical Team

**Status:** Proposed

**Engineering Method**

**Definition:** Technical description of approach and capabilities needed for a specific repeatable activity in a use case from an end user perspective. This describes the general task, preferred workflow and artefacts involved, not limited to interoperability.

**Source:** Sytze Kalisvaart

**Status:** Proposed

**(IOS) Engineering Standard**

**Definition:** Existing Engineering Standards defined by Standardization Bodies which have been adopted by the CRYSTAL consortium for supporting Non-Lifecycle Systems Engineering Activities, and which are already widely used by CRYSTAL partners.

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed

**Engineering Tool Function (ETF)**

**Definition:** Engineering Tool Function is a detailed function of an Activity and it refers to an action performed by means of a Tool Brick. ETF's can be of two types: IOS Service or Internal ETF.

**Source:** WP-602 Technical Team

**Status:** Relevant for Glossary

**ETF Catalogue**

**Definition:** It is a container of Engineering Tool Functions identified in CRYSTAL.

**Source:** WP-602 Technical Team

**Status:** Proposed

**Extended OSLC Vocabulary**

**Definition:** The vocabularies related to Lifecycle Interoperability defined by the CRYSTAL consortium as extensions of the existing ones defined within the OSLC Domains or as new IOS Domains (e.g., for Risk/Safety Management, Ontology Management, Formal Requirement Management, etc).

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed

**IOS**

**Definition:** Acronym for InterOperability Specification, the Crystal standard for interoperability between system development software tools.

**Source:** DOW

**Status:** Proposed
IOS (2)
Definition: The CRYSTAL Interoperability Specification.
Source: ARTEMIS projects CESAR and MBAT
Status: Proposed

IOS Extended Service
Definition: Extended (or added-value) Services related to Lifecycle Interoperability defined by the CRYSTAL consortium, e.g., related to Change Impact Analysis, Testing/Requirement Coverage, Automation support between Simulation & Design Activities, Advanced Services related to Ontology Management (e.g., pattern matching algorithms), etc.
Source: CRYSTAL deliverable D601.022
Status: Proposed

IOS NLC Domain
Definition: Non-Lifecycle Domains as part of the CRYSTAL Non-Lifecycle IOS for supporting In Depth Systems Engineering Activities based on existing Engineering Standards identified as already widely used among CRYSTAL partners and European developing organizations. Indeed, CRYSTAL aims at supporting engineering activities focused on end-users’ businesses, and which require detailed, specific and “bespoke” semantics and methodologies besides Lifecycle Interoperability. Such engineering activities encompass heterogeneous co-simulation, combination of dynamic testing and formal static analysis, variability management, design space exploration, just to name a few.
Source: CRYSTAL deliverable D601.022
Status: Proposed

IOS NLC service
Definition: Non-Lifecycle Interoperability Services defined in the context of Non-Lifecycle Domains.
Source: CRYSTAL deliverable D601.022
Status: Proposed

IOS service
Definition: Also called Engineering IOS function, an engineering function provided by a Tool Brick that supports interoperability between different Tools.
Source: WP-602 and WP 209 Technical Teams
Status: Relevant for Glossary

IOS Service (2)
Definition: A generic term for characterizing OSLC Core and Extended Services (as part of the Lifecycle IOS) and the NLC Services (as part of the Non-Lifecycle IOS).
Source: CRYSTAL deliverable D601.022
Status: Proposed
**Lifecycle IOS**

**Definition:** The part of the CRYSTAL IOS focused on Lifecycle Interoperability, the latter being based on Artefacts used in Systems Engineering Development Environments (“Tool Chains”): (i) to support collaboration between stakeholders of different roles, different engineering disciplines, and different industrial domains, (ii) to enable status reporting, traceability, dashboarding, analysis, prediction, data collection for reports, automation support, etc., between Tools and Data Repositories.

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed

**Non-Lifecycle IOS**

**Definition:** The part of the CRYSTAL IOS focused on Non-Lifecycle Interoperability for supporting In Depth Systems Engineering Activities. This part is based on existing Engineering Standards identified as already widely used among CRYSTAL partners and European developing organizations. CRYSTAL aims at supporting engineering activities focused on end-users' businesses, and which require detailed, specific and “bespoke” semantics and methodologies besides lifecycle interoperability. Such engineering activities encompass heterogeneous co-simulation, combination of dynamic testing and formal static analysis, variability management, design space exploration, just to name a few.

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed

**OSLC Based Specification**

**Definition:** The Interoperability Specifications defined by OSLC (Open Services for Lifecycle Collaboration) [5]. The OSLC initiative is creating a family of web services specifications for products, services and other tools that support all phases of the software and product lifecycle. The purpose of these specifications is to enable integration between products that support Application Life-cycle Management (ALM) and Product Life-cycle Management (PLM).

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed

**OSLC Core**

**Definition:** The OSLC Core specifications [6]. It sets out the common features that every OSLC Service can be expected to support using terminology from the World Wide Web Consortium (W3C). This specification is mostly about OSLC Services, it specifies what OSLC Services MUST, SHOULD and MAY do. It also contains some required behaviors for OSLC clients and rules for OSLC domain specifications that extend this specification.

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed

**OSLC Core Service**

**Definition:** The Core Services defined by OSLC for handling OSLC Resources. Typical services are:

- C.R.U.D services (for Creating a resource using HTTP POST and content being resource format of choice, Reading a resource using HTTP GET and standard HTTP content negotiation, Updating a resource using HTTP PUT to send updated resource, Linking a resource using properties where values are just URIs, Deleting a resource using HTTP DELETE),
- Querying for resources, and
- UI Preview & Delegation.

**Source:** CRYSTAL deliverable D601.022

**Status:** Proposed
OSLC Domain

**Definition:** The OSLC Domain Specifications [7]. Each domain (or part of the lifecycle) has its own group and specification, for example there are Change Management, Quality Management, Estimation & Measurement and more.

**Source:** CRystal deliverable D601.022

**Status:** Proposed

OSLC Domain Service

**Definition:** OSLC Services defined for an OSLC Domain-specific purpose, e.g., related to extensions of OSLC Query Capabilities via new query parameters.

**Source:** CRystal deliverable D601.022

**Status:** Proposed

OSLC Resource Vocabulary

**Definition:** OSLC relies on the concept of Resource, the latter being managed by OSLC Services. An OSLC Resource is typically something like a Change Request, a Requirement or some other ALM or PLM artifact or record. Within each OSLC Domain is specified a vocabulary, consisting of a set of resource types, each of them defined by a URI – Unified Resource Identifiers, and a set of properties and relationships.

**Source:** CRystal deliverable D601.022

**Status:** Proposed

PB Modeler

**Definition:** A software application, supported by catalogues, to define a SEE descriptor relevant to a specific development process applying identified Engineering Methods. It is a component of the Platform Builder.

**Source:** WP-602 Technical Team

**Status:** Proposed

Platform Builder (PB)

**Definition:** A software solution based on development process definition, and on the availability of an engineering methodology and of related support tools, to support the configuration of an instance of the SEE for use by company engineers to manufacture the product

**Source:** WP-602 Technical Team

**Status:** Proposed

Reference Technology Platform

**Definition:** The Crystal collection of interoperable system engineering bricks that can be used to set up a system engineering environment in a company.

**Source:** RTP team Sept 2014

**Status:** Proposed

SEE descriptor and meta model

**Definition:** SEE descriptor is an instance of SEE meta-model, containing information for: describing the Tool set to deploy in terms of Tools and their properties, interoperability aspects, and IT infrastructure constraints to be applied in order to have a correct and interoperable deployment of selected tools.

**Source:** WP-602 Technical Team

**Status:** Proposed
**Tool**  
**Definition:** A software application or component – described in terms of the offered functions – meant to support Engineering Methods adopted in a system development process.  
**Source:** alternative for tool brick  
**Status:** Proposed

**Tool Brick descriptor and meta model**  
**Definition:** Tool Brick descriptor is an instance of Tool meta-model, describing Tool in a formal way including engineering functions and technology requirements/constraints to deploy the tool itself.  
**Source:** WP-602 Technical Team  
**Status:** Proposed

**Tool Catalogue**  
**Definition:** It is a container of Tool Brick descriptors, defined on the basis of provided ETF. Two different Tool Catalogue are defined in the PB approach:  
- RTP Tools Catalogue contains the Tool Descriptors of the Available Tools in the market that support the IOS  
- Company Available Tools Catalogue contains the Tool Descriptors of the tools that are available for use within a given company and which provide IOS support.  
**Source:** WP-602 Technical Team  
**Status:** Proposed

**Use Case**  
**Definition:** It is a CRYSTAL Use Case and describes information relevant to a system to be developed in the company context.  
**Source:** WP-602 Technical Team  
**Status:** Proposed
4 The collaborative work tool: Semantic MediaWiki

4.1 Tools and workflows

The Crystal project creates and maintains a glossary of terminology inherent to the project itself and to the field of application. Currently, to achieve this purpose, a tool based on Microsoft Share Point is being used. It provides the necessary means to the glossary editors (spreaded around the world) to add and edit glossary terms in a collaborative way.

As the number of terms is constantly increasing, it is important to give a classification to them through the use of labels to understand easily in which state of approval they are and what is their level of importance inside the project.

Similarly, the description of the terms contains important information that many times makes internal reference to other elements of the glossary. This information, sometimes, doesn't contain any semantic meaning, so it can be ambiguous.

These limits of the current solution should be handled. In general terms, the management of the Crystal glossary can be seen as a collaborative editing problem, or a knowledge management problem. In the following sections existing approaches and tools for this issue are described. Next it is described the proposed solution.

4.2 Collaborative editing

The composition of the Glossary basically needs a collaborative editing. This has been defined in many ways. According to [1], it is “the practice of groups producing works together through individual contributions. Effective choices in group awareness, participation, and coordination are critical to successful collaborative writing outcomes”.

In different environments, the need for collaborative editing tools has become a necessity. This is where the IT through a series of computer tools and software, proposes a number of different solutions [2].

To facilitate their study and analysis, they can be classified as follows:

- **Wikis**

  According to [3], a wiki is an application, typically a web application, which allows a collaborative modification, extension, or deletion of its content and structure. In a typical wiki, text is written using a simplified markup language (known as “wiki markup”) or a rich-text editor. While a wiki is a type of content management system, it differs from a blog or most other such systems in that the content is created without any defined owner or leader, and wikis have little implicit structure, allowing structure to emerge according to the needs of the users.

  The encyclopedia project Wikipedia is the most popular wiki on the public web in terms of page views, but there are many sites running many different kinds of wiki software. Wikis can serve many different purposes both public and private, including knowledge management, notetaking, community websites and intranets. Some permmit control over different functions (levels of access). For example, editing rights may permit changing, adding or removing material. Others may permit access without enforcing access control. Other rules may also be imposed to organize content.

  There are a large number wikis available that can run on different platforms and environments. Each of these wikis, has particular characteristics, which can be adapted to different requirements. The web site WikiMatrix [4] makes available to users a graphical matrix with the characteristics of each wiki, thus allowing to compare their different features and characteristics.
Of these wikis, the most used are [5]:

- **MediaWiki**: Used by the WikiPedia project, which is the most popular wiki (PHP and MySql).
- **MoinMoin**: A PythonLanguage wiki engine, features flexibility and modular design.
- **PhpWiki**: A very popular PhpLanguage Wiki based on UseModWiki, with many features added.
- **XWiki**: A sophisticated Wiki platform build with Enterprise technology and also declared as the 'Next Generation Wiki'
- **OddMuseWiki**: Really popular descendant of UseModWiki ("one big Perl script").
- **TikiWiki CMS**: A full-featured, open source, multilingual, all-in-one Wiki + CMS + Groupware written in PHP.
- **PmWiki**: A popular PHP Language Wiki, easy installation, simple design, nice feature list.
- **DokuWiki**: Standards compliant, simple WikiEngine written in PHP, accepted into Debian and Fedora linux distro's
- **FoswikiEngine**: The free open source fork from TWiki, a powerful PerlLanguage structured wiki with numerous plugins, aimed at large corporate Intranets
- **MojoMojo**: The modern PerlLanguage wiki, powered by the CatalystFramework and the DBIC ORM.
  Hierarchical structure, AJAX live previews, 3-way merge edit conflict resolution, attachment gallery etc.

**Version control systems**

This category refers more to collaborative editing systems focused on the management of software assets (notably source code). In this area, the management of versions of code provided by the collaborators is a crucial task.

According to [6], a Version Control (also known as version control (system) (VCS), source control or (source) code management (SCM)) is the management of multiple revisions of the same unit of information. It is most commonly used in engineering and software development to manage ongoing development of digital documents like application source code, art resources such as blueprints or electronic models, and other projects that may be worked on by a team of people. Changes to these documents are usually identified by incrementing an associated number or letter code, termed the "revision number", "revision level", or simply "revision" and associated historically with the person making the change. A simple form of revision control, for example, has the initial issue of a drawing assigned the revision number "1". When the first change is made, the revision number is incremented to "2" and so on.

Software solutions belonging to this category can be grouped according to their architecture model and license type (open source or proprietary). In [7], a community created matrix is presented to compare the most common revision control software solutions available. Software solutions are compared by considering different criteria like:

- General information
- Technical information
- Features
- Advanced features

---

1http://en.wikipedia.org/wiki/Content_management_system

2http://en.wikipedia.org/wiki/Collaborative_software

3http://www.catalystframework.org

4http://www.dbix-class.org
- **Word processors (change recording)**

According to [8], a word processor is an electronic device, or computer software application, that, as directed by the user, performs word processing: the composition, editing, formatting, and sometimes printing of any sort of written material.

This category of software is included because through their "change tracking" of "change recording" features, many people mainly in organizations and universities is able to implement a collaborative workflow for producing and editing documents.

Within the most used ones, there are:
- Microsoft Word[^5]
- WordPerfect[^6]
- OpenOffice Writer[^7]
- LibreOffice Writer[^8]
- AbiWord[^9]

- **Cloud-based solutions**

This category includes those collaborative software applications that are accessed through the web (since they are somewhere in the cloud). Users access these solutions through a browser and perform the process of creation and collaborative editing therefrom. Some of these solutions, in addition to providing an environment for collaborative editing, include features such as real-time editing, chat and access permissions list.

Among the most popular:
- Google Docs[^10]
- EtherPad[^12]
- Zoho[^13]

[^6]: http://www.wordperfect.com/
[^7]: https://www.openoffice.org/product/writer.html
[^8]: http://www.libreoffice.org/discover/writer/
[^9]: http://www.abiword.org
[^10]: http://docs.google.com/
[^12]: http://etherpad.com/
[^13]: http://zoho.com/
The Semantic Web is a set of tools and technologies for the publishing and processing of structured, and semi-structured data on the Web. The idea of extending the capabilities of the Web in order to publish structured data on it, exists from its own creation. It’s in 1994 when Tim Berners-Lee highlights the need of introducing Semantics in the Web to achieve this idea. Later, this idea became the well-known “Semantic Web”.

According to the W3C, the Semantic Web is the Web of Linked Data which comprise a set of technologies that enable people to create data stores on the Web, build vocabularies and write rules for manipulate such data (The World Wide Web Consortium, 2013). It is an evolution of the traditional Web by giving structure to the meaningful content of Web data and acting as a layer of resource descriptions to combine web resources with machine-understandable data. In this way it enables people and computers to work in cooperation to simplify the sharing and handling of information [9]. Nevertheless, as the same Berners-Lee stated, “this simple idea, however remains largely unrealized”[10].

While the Semantic Web is the goal, Linked Data provides the means to make it reality [11], it refers to a set of best practices for publishing and connecting structured data on the Web in order to increase the number of data providers and consequently accomplish the goals of the Web of Data. In this way Linked Data makes it possible to semantically interlink and connect different resources at data level regardless the structure, authoring, location etc.

The set of principles are:
- Use URIs as names for things.
- Use HTTP URIs, so that people can look up those names.
- When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).
- Include links to other URIs, so that they can discover more things.

Data published on the Web using Linked Data has resulted in a global data space called the Web of Data. Mainly through the efforts of the scientific community and the W3C LOD project, more and more data have been published on the Web of Data, helping their growth and evolution.

---

14 http://thinkfree.com/
15 https://www.apple.com/iwork-for-icloud/
16 https://owncloud.org/
17 https://www.sharelatex.com/
18 http://www.w3.org/Talks/WWW94Tim
19 http://www.w3.org
20 http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData
## 4.4 Proposed solution

A possible solution for the Glossary composition could be found by considering the critical issues above described, concerning the number of Partners, the traceability of versions and the need for an open discussion. Using a wiki might encourage the collaborative creation and editing of the Glossary. Considering also the benefits that can be obtained from the Web of Data, it was found also the possibility of using a plugin that will allow associating semantic elements glossary, with Semantic Web concepts. The solution and its components are detailed here below.

### LAMP Architecture

According to Wikipedia [12], LAMP is an acronym for an archetypal model of web service solution stacks, originally consisting of largely interchangeable components: Linux, the Apache HTTP Server, the MySQL relational database management system, and the PHP programming language. As a solution stack, LAMP is suitable for building dynamic web sites and web applications.

The LAMP model has since been adapted to other componentry, though typically consisting of free and open-source software. As an example, the equivalent installation on a Microsoft Windows operating system is known as WAMP.

### LAMP Components

- **Linux**

  Linux is a Unix-like computer operating system assembled under the model of free and open source software development and distribution. Most Linux distributions, as collections of software based around the Linux kernel and often around a package management system, provide complete LAMP setups through their packages. According to W3Techs in October 2013, 58.5% of web server market share was shared between Debian and Ubuntu, while RHEL, Fedora and CentOS together shared 37.3%.

- **Apache**

  The role of LAMP's web server has been traditionally supplied by Apache, and has since included other web servers such as Nginx.

  The Apache HTTP Server has been the most popular web server on the public Internet. In June 2013, Netcraft estimated that Apache served 54.2% of all active websites and 53.3% of the top servers across all domains. In June 2014, Apache was estimated to serve 52.27% of all active websites, followed by nginx with 14.36%.

  Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Released under the Apache License, Apache is open-source software. A wide variety of features are supported, and many of them are implemented as compiled modules which extend the core functionality of Apache. These can range from server-side programming language support to authentication schemes.

- **MySQL**

  MySQL's original role as the LAMP's relational database management system (RDBMS) has since been alternately provisioned by other RDBMSs such as MariaDB or PostgreSQL, or even NoSQL databases such as MongoDB.

  MySQL is a multi-threaded, multi-user, SQL database management system (DBMS), acquired by Sun Microsystems in 2008, which was then acquired by Oracle Corporation in 2010. Since its early years, the MySQL team has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements.
MariaDB is a community-developed fork of MySQL, led by its original developers. PostgreSQL is also an ACID-compliant relational database, unrelated to MySQL.

MongoDB is a widely used open-source NoSQL database that eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas (calling the format BSON), making the integration of data in certain types of applications easier and faster.

- **PHP**
  PHP's role as the LAMP's application programming language has also been provisioned by other languages such as Perl and Python.

  PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP code is interpreted by a web server via a PHP processor module, which generates the resulting web page. PHP commands can optionally be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications.

  PHP is free software released under the PHP License, which is incompatible with the GNU General Public License (GPL) due to the PHP License's restrictions on the usage of the term PHP.

  Perl is a family of high-level, general-purpose, interpreted, dynamic programming languages. The languages in this family include Perl 5 and Perl 6. They provide advanced text processing facilities without the arbitrary data-length limits of many contemporary Unix command-line tools, facilitating manipulation of text files. Perl 5 gained widespread popularity in the late 1990's as a CGI scripting language for the Web, in part due to its parsing abilities.

  Python is a widely used general-purpose, high-level programming language. Python supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. It features a dynamic type system, automatic memory management, and a standard library. Like other dynamic languages, Python is often used as a scripting language, but is also used in a wide range of non-scripting contexts.

**MediaWiki**

Among the many existing wikis, thanks to its features and especially its extensibility, MediaWiki[13] was chosen as the base for mounting the wiki solution.

MediaWiki is free server-based software which is licensed under the GNU General Public License (GPL). It's designed to be run on a large server farm for a website that gets millions of hits per day.

MediaWiki is an extremely powerful, scalable software and a feature-rich wiki implementation that uses PHP to process and display data stored in a database, such as MySQL. Pages use MediaWiki's wikitext format, so that users without knowledge of XHTML or CSS can edit them easily.

When a user submits an edit to a page, MediaWiki writes it to the database, but without deleting the previous versions of the page, thus allowing easy reverts in case of vandalism or spamming. MediaWiki can manage image and multimedia files, too, which are stored in the file system. For large wikis with lots of users, MediaWiki supports caching and can be easily coupled with Squid proxy server software.

**Advantages of MediaWiki**

- Long-term maintenance and support: MediaWiki development is backed by an organization with an annual budget in excess of $27 million. One need not worry that in the foreseeable future the codebase will be completely abandoned by its developers and become unmaintained.
• Suitability for huge, highly active wikis: MediaWiki is used by the English Wikipedia, the largest wiki in the world, with more than 4 million pages, 600 million edits since the project's inception, and 470 million unique visitors monthly. MediaWiki has been designed with scalability in mind for high-usage, high-profile sites that are prone to vandalism, spam, and other attacks.

• Lots of content to borrow: Whatever sorts of articles or templates (e.g. info-boxes) you are used to seeing on Wikipedia and think would be useful on your site, you can import. Often the starting place for a new wiki is to borrow content from Wikipedia (subject to the CC-by-SA license). If you use wiki software other than MediaWiki, the Wikipedia content might have to be reformatted or you might have to start from scratch.

MediaWiki Open Source Extensions

A very important feature that MediaWiki offers, is the extensibility through the Open Source Extensions. They can be used for different purposes like:

• extend the wiki markup used to write articles
• add new reporting and administrative capabilities
• change the look and feel of MediaWiki
• enhance security via custom authentication mechanisms

While some of this extensions are maintained by MediaWiki's developers, others were written by third-party developers.

Semantic MediaWiki

Semantic MediaWiki (SMW)[14] is an extension of MediaWiki – the wiki application best known for powering Wikipedia – that helps to search, organize, tag, browse, evaluate, and share the wiki's content. While traditional wikis contain only text which computers can neither understand nor evaluate, SMW adds semantic annotations that allow a wiki to function as a collaborative database. Semantic MediaWiki was first released in 2005, and currently has over ten developers, and is in use on hundreds of sites. In addition, a large number of related extensions have been created that extend the ability to edit, display and browse through the data stored by SMW: the term "Semantic MediaWiki" is sometimes used to refer to this entire family of extensions.

Wikis are a great tool for collecting and sharing knowledge in communities and organizations. This knowledge is mostly contained within texts and multimedia files, and is thus easily accessible for human readers. But though wikis are very good for storing and retrieving individual facts, they are less useful for getting queried or aggregated information. As a simple example, let's say you use a wiki that stores information about projects related to your organization. You have the following simple question:

"What are the active projects that were started in 2012?"

This should be an easy question to answer, but in fact it's not - you would have to read through all of the pages about projects every time you wanted to answer the question. Text searches won't necessarily help. Categories could help to some extent, although they're not an ideal tool and maintaining them can become complex (see below). And there doesn't exist an artificial intelligence tool that could help with this task either. Semantic MediaWiki enables wikis to make their knowledge computer-processable, so that you can find and display the answer to this question - and to many more.

Semantic MediaWiki introduces some additional markup into the wiki-text which allows users to add "semantic annotations" to the wiki. While this at first appears to make things more complex, it can also greatly simplify the structure of the wiki, help users to find more information in less time, and improve the overall quality and consistency of the wiki. Here are some of the benefits of using SMW:

• Automatically-generated lists. Lists and tables are a natural way to view information at a glance. In some cases, non-semantic wikis contain human-generated lists; Wikipedia itself has thousands, like "List of metropolitan areas in Spain by population". These lists are prone to errors, since they have to be updated manually. Furthermore,
the number of potentially interesting lists is huge, and it is impossible to provide all of them in acceptable quality. In SMW, lists are generated automatically like this. They are always up-to-date and can easily be customized to obtain further information.

- Visual display of information.

  The various display formats defined by additional extensions, such as Semantic Result Formats and Semantic Maps, allow for displaying of information in calendars, time-lines, graphs and maps, among others, providing a much richer view of the data than simple lists and tables would.

- Improved data structure.

  MediaWiki wikis tend to make heavy use of categories for structuring data. While these are generally helpful, consider the category on Wikipedia called “1620's births”; if the information in these pages were stored using SMW, these categories could be replaced by simple semantic values, reducing the need for a complex classification system. In addition, if semantic markup within the wiki is stored within templates, otherwise known as semantic templates, a wiki can easily gain a solid data structure. And the Semantic Forms extension lets administrators create forms for adding and editing the data within semantic templates, thus making the addition of semantic information even easier and more straightforward than using regular wiki text.

- Searching information.

  Individual users can search for specific information by creating their own queries, supported via extensions like Semantic Drilldown and Semantic Forms.

- External reuse.

  Data, once it is created in an SMW wiki, does not have to remain within the wiki; it can easily be exported via formats like CSV and JSON. This enables an SMW wiki to serve as a data source for other applications, or, in the case of enterprise usages, to take over the role that a relational database would normally play. Through the use of the External Data extension, SPARQL, and other tools, one SMW-based wiki can even use the data from another, eliminating the need for redundancy between wikis. You can also query SMW's data from outside the wiki, via the API or an RDF triple-store.

- Integrate and mash-up data.

  Data contained in an SMW installation does not have to be an isolated store of information. Extensions such as Data Transfer and External Data empower you to integrate external data (coming e.g. from legacy systems, web services or linked data sources) and interrelate it with existing semantic data in the wiki. Thus, an SMW-powered wiki can serve as a central information hub in an IT landscape.

Cited references

4.5 Implementation

Solution described in previous sections has been implemented to test its functionalities and performance. Similarly, a semantic glossary of terms was modelled, such as that currently used in the Crystal project.

Technically speaking, an installation of Ubuntu 14.04 has been used as a base. On it is installed LAMP platform and its components. Likewise the MediaWiki platform and the Semantic MediaWiki plug in have been also installed. To bridge the semantic data (triplets) generated by Semantic MediaWiki, the Apache Jena\textsuperscript{21} was used as triple store. To model the glossary in a better way, has been necessary to use the Semantic Glossary\textsuperscript{22} plugin, which has allowed us the creation and editing of the elements of the glossary in the form of triplets that are saved as RDF in the triple store.

Figure 4-1, shows the web form that allows entering new elements to glossary and edit existing ones. Even if the creation and editing of the glossary entries is "textual", the generation of RDF triples is carried out by the system automatically.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure4-1.png}
\caption{Insertion and editing of terms}
\end{figure}

The general view that contains all the glossary elements present in the system is shown in Figure 4-2. When the system is in the definition of an element glossary, a reference to other elements in the same glossary, this generates the reference and link to automatic.

\textsuperscript{21}https://jena.apache.org/index.html

\textsuperscript{22}http://www.mediawiki.org/wiki/Extension:Semantic_Glossary
From the previous interface view that shows the glossary items, it is possible to access the detailed information of each element of the glossary by clicking on its name. After clicking, the system will display an HTML view (see Figure 4-3) with the detailed information of the item. Additionally, the view provides a link that allows the user to see the same information rendered in RDF format (see Figure 4-4), as it is originally stored in the semantic triple store.

![Figure 4-2: Main view of the glossary](image)

![Figure 4-3: HTML view of a glossary item](image)

```xml
<swinv:wikiPageModificationDate rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2015-02-04T01:29:56Z</swinv:wikiPageModificationDate>
<property:Modification_date-23aux rdf:datatype="http://www.w3.org/2001/XMLSchema#double">2437057.5624537</property:Modification_date-23aux>
<property:Glossary-2DDefinition-23aux rdf:datatype="http://www.w3.org/2001/XMLSchema#string">A combination of interacting elements organized to achieve one or more stated purposes.</property:Glossary-2DDefinition-23aux>
<property:Glossary-2DTerm-23aux rdf:datatype="http://www.w3.org/2001/XMLSchema#string">system</property:Glossary-2DTerm-23aux>
```

![Figure 4-4: RDF view of a glossary item](image)
5 Assessment of the Glossary composition workflow

5.1 Critical issues of the first proposal

Use of the collaborative tool of the Semantic Media Wiki might introduce some initial difficulties.

- If a requirement of the CRYSTAL Project is keeping all the information inside the environment of the CRYSTAL Share Point, opening a session dedicated to Semantic Media Wiki for discussion of the Glossary contents may need to have a separated environment, out of the main website of the CRYSTAL Project.
- Collaborative work seems to be more effective after that at least a preliminary collection of terms was included inside the shared area, thus making preferable to select a suitable number of terms before starting a real integrated and collaborative discussion.
- The above described difficulty actually was related to a preliminary effect of a call for proposal of terms sent to the WPs Leaders, together with a summarized list of outlines and goals of the Glossary. As soon as Partners of the CRYSTAL Project addressed the problem of collecting the terms from the WPs to be proposed as candidates for the CRYSTAL Glossary, it was immediately realized that feeling about the type of terms to be included inside the Glossary was extremely different within the domains and the WPs, thus making rather difficult opening a discussion through a collaborative tool like the Semantic Media Wiki, being accessible to the whole CRYSTAL Community. To assure that each definition could be interesting and valid for all the domains in some cases proposed terms were fairly basic, like “system”, “engineering method”, “requirement”, in other cases fairly detailed like “interoperability”, or too detailed as “Multiprocessor System-on-Chip”.
- That test pointed out a possible risk of proceeding in practice to a selection of typical terms already present inside the other glossaries available in the literature, thus completely changing the goal of this Glossary whose aim is to create a light, useful and agreed reference of terms to be used cross-domains and within the frame of this project, never to repeat an assessment of the vocabulary of terms being already performed by previous projects.

To assure a good homogenization of the Glossary contents, being related to both the detail of concepts applied to the work of teams and to the type of terms included, a first screening was performed by collecting the terms from each WP through the Share Point. Then for each one the pertinence with this Glossary was evaluated, once that some criteria of pertinence and a dedicated board of reviewers were preliminary selected by the CRYSTAL Project Management.

5.2 Glossary directory on the CRYSTAL Share Point

Among the sections of the CRYSTAL Share Point a dedicated area referred to as “Glossary” was introduced to upload the preliminary selection of terms performed by each WP, through the Leader. A very basic information was required for this uploading:

- Item (acronym, concept or term)
- Definition (proposed description)
- Source (reference recognized in the literature or within the Standards)
- Related Concept (connection to other terms of the Glossary, eventually used inside the definition of the word)

Actually, in addition to the fields above described the List of Glossary includes the author of each item, thus indicating the team or even the domain who proposed the term. A first period of time for the submission of terms was closed at the end of December 2014. At that time up to 135 items were uploaded. A second period of time was foreseen from January to March 2015. In this second step a number of terms was directly included by the specialised Teams of the CRYSTAL Project, being identified by the Management to collect the new ontologies across the domains of the CRYSTAL Project and used within some WP. Last contributions came in May 2015.
To prepare the review process each extract of the Glossary is a document which collects all the submitted terms, enriched with a set of status tags, being useful for the identification of issues and a field dedicated to cross-references. A preliminary work was performed to apply the status tags to all the terms included in the Glossary available on the related AVL SharePoint space, to highlight some critical issues related to each items. The status tag system was composed by the following tags:

- **NEW**: new term proposed with the project to be accepted
- **OK**: everything is ok
- **OL**: definition overlapped, possible duplicate
- **S**: source is missing or too vague
- **IMP**: definition can be improved
- **NC**: not clear, definition is too vague or too specific
- **NV**: not valid, definition doesn't fit requirements

The abovementioned status tags are compatible with the structure that should be reproduced in the Semantic MediaWiki, when activated.

This preliminary action provided a number of tables including the terms and the status tags as the reviewers identified by the CRYSTAL Project Management could evaluate during the first review process. These status tags are no more associated to the terms of the Glossary, as it appears in Section 3.3. According to the review only the two status tags, “Proposed” and “Relavant for Glossary”, were maintained for further assessments.

### 5.3 Review process

According to the ‘bottom–up’ approach applied to the Glossary composition an assessment of the workflow was preliminary achieved and it is herein summarized to clarify the steps already performed until this deliverable and the following activities foreseen to complete the composition of the CRYSTAL Global Glossary.

**Step 1**: All the WP Leaders collected the relevant terms used within the workpackage deliverables and agreed with all the Partners involved in the WP to submit a selection of terms which might be used cross–domain through the Glossary share point.

**Step 2**: List of terms was uploaded inside the Glossary on the Share Point, and monitored by POLITO who detected multiple submissions and eventual overlaps in definitions, by checking also the completeness of the provided information. Authors were required to complete the submission in some case.

**Step 3**: The Glossary was submitted to a board of Reviewers selected by the Technical Board. A preliminary check concerned the pertinence of each term with the main criterion of acceptance, which was defined by the following statement:

**Glossary Content:**

1. The glossary should only contain concepts where agreement on a project-wide level is absolutely required for project execution and for communication on project-level.

2. In particular those items include the concepts of the RTP, IOS, and the Platform Builder and general concepts like safety, variant management, configuration management, etc.
Step 4: A second call was sent the teams involved within RTP, IOS and Platform Builder to complete the list of terms and assuring that all the partial glossaries already composed within the frame of the teams were added to the main one. Teams uploaded some other terms and related definitions.

Step 5: A first version of the Glossary was consolidated and sent the Technical Board for a preliminary agreement. (until this deliverable)

Step 6: Contents will be shared with all the CRYSTAL Community for assessing the definitions and adding few other terms.

Step 7: Extract of Glossary will be sent to the WPs Leaders for a review, aimed at collecting emendaments, suggestions, refinement proposals. A open discussion will be arranged thorough some conference call.

Step 8: Glossary will be consolidated and submitted to the Technical Board for the final agreement.

Step 9: Dissemination activity. The Glossary will be made public eventually through a Public Semantic Media Wiki interlinked with the CRYSTAL homepage to test a possible enrichment.

Step 10: Results of dissemination after a suitable period of time will be collected, consolidated and discussed.