

# Project Markup Language (PML) Schema Proposal

PAULO GOMES<sup>1</sup>, JORGE MACHADO<sup>2</sup>

Superior School of Technology and Management,  
Polytechnic Institute of Portalegre,  
Lugar da Abadessa, Apartado 148,  
Portalegre – Portugal

<sup>1</sup>paulogomes@estgp.pt

<sup>2</sup>jmachado@estgp.pt

**Abstract.** In this paper we present the steps followed to make a proposal for a Project Markup Language (PML). PML is to use in project management solutions, like GPRM (Global Project for Research Management) [1]. PML (Project Markup Language) is a markup language for Project Management Servers (like Microsoft Project Server/EPMS Servers [10], Global Project Management/GPM Servers or GPRM Server [1]). PML has the main purpose to establish a standard model to Project information, to use it through the various Project Management Applications and Servers. With that we can use search and retrieval index engines (like SPEAK) to have a free communication between different Project Servers and Applications. This paper focuses on the language features and presentation scheme designed for Project Management.

## 1 Introduction

This document defines the PML. This document also revises the PML grammar, using the XML concept, defined by W3C Recommendation.

Since March, 2000, some of the industry software leaders join Pacific Edge Software [23] to define the Project Management XML Schema. Creating a flexible Business-to-Business schema, that enables project data exchange between Information Systems. Around this, Pacific Edge Software [23] had worked by developing the concept of project knowledge management (PKM) solutions for project-driven organizations. Companies like eProject.com, Great Plains, Onyx Software, PlanView, Primavera Systems [19] already start to use PMXML model [21].

The coalition of collaboration technology, e-business solutions, customer relationship management (CRM), project management, workforce management and PKM companies will work together to forge an open industry standard by making modifications and extensions to PMXML schema [21].

The schema and its source already can be viewed at BizTalk web site [24].

The goal of the XML schema design is to enable project management tools 'talk' and thus 'understand' each other; as a result (is it maintained) they can exchange in-

formation regarding task and project status, resource assignments, additional project attributes, and the work involved to complete these projects.

In 2002, organizations like NASA, Oracle, PM Boulevard LLC, and PM Solutions have recently joined the PMXML initiative [19]. Also the Yahoo Group (pm\_xml - Project Management XML) group are developing is Project Management XML (PMXML) schema [21].

Historically, however, project management has been treated as ancillary to a normal business practice. Successful enterprises need to share project management information such as status, resource allocations, scheduling, and costing. But, for innovative projects, including R&D projects, as more innovative the tasks are, more difficult is to foresee with some severity the time and necessary resources do to it. In the actual processes of project management, such as Enterprise Project Management (EPM) [10] or Global Project Management (GPM), these characteristics, would become (planning and execution) practically impossible to control. In Microsoft Project Server 2003 [9], we solve with easiness the problem of the resources contribution in geographically distant places, but does not solve the management problem when necessary knowledge are need for the solution on emergent problems, because these management systems are based on the local control of the workmanship evolution determined by the empirical knowledge of the Project Manager and not based on a network knowledge management. In these cases, Project Manager cannot enter the time, the resources and costs of tasks with techniques that still are not known. Normally projects with these particularities are developed without an end "on the sight". Project Manager know when project pulls out, but do not make the minim idea at least if it goes to be concluded. This type of situation creates a feeling of lack of control on all the elements, generating inevitably the loss of motivation.

To solve that, Superior School of Technology and Management from Polytechnic Institute of Portalegre create a workgroup to research and develop a solution to implement in projects with this kind of characteristics. To this concept, we name it "GPRM – Global Project for Research Management" [1].

The main goal to GPRM [1] is to provide a sufficient intelligence to help the project manager determining the correct project plan. Instead of Project Manager determine the duration, cost or even how many resources is necessary to each task, GPRM [1] will provide all of this information automatically.

To create an Intelligent Project Framework we need to do much more than PMXML. We already assume PMXML [21] schema as the beginning of our work. And to communicate the project information from a project information system to another, PMXML [21] schema it's sufficient. But PMXML [21] will never give us the possibility to achieve our goal. That's why we propose "PML – Project Markup Language" schema, as an extension of PMXML [21] for innovate and research projects. Also, we are including almost the entire PMXML [21] schema inside of PML.

So, the main goal for this paper is to propose an effective schema that provides software solutions like GPRM [1] to use a real Project Knowledge Management (PKM).

## 2 The PMXML solution

The PMXML [21] standard is a data definition for project management systems. The PMXML Consortium [21] maintains the standard. PMXML [21] was originally created by Pacific Edge Software Inc. in 2000 and published at BizTalk.org [24].

PMXML [21] is an important step for the Project Management industry, following initiatives in other industries to provide integration mechanisms on application-level. Project management systems that are PMXML-enabled allow distributing data easily inside organizations and between different organizations and their respective management systems. The standard can help vendors, who don't have to implement so many different import/export filters, and it can help the users, who can transfer PM data to other systems (e.g., HR, ERP) or share it with partners more easily.

A standard that allows easy sharing and transfer of project management data could help to bring project management out of its relative isolation in comparison to the line management.

## 3 The PMXML Data Definition

The PMXML data definition is already on the version 2. It contains definitions for four major project management data types and some minor ones. The four major data types are: project, resource, task, and assignment. The definition starts with a ProjectManagementSchema as a root element. It contains InstanceData, a collection of user- and application-specific data; PoolResources, a collection of resource definitions; Projects, a collection of project definitions.

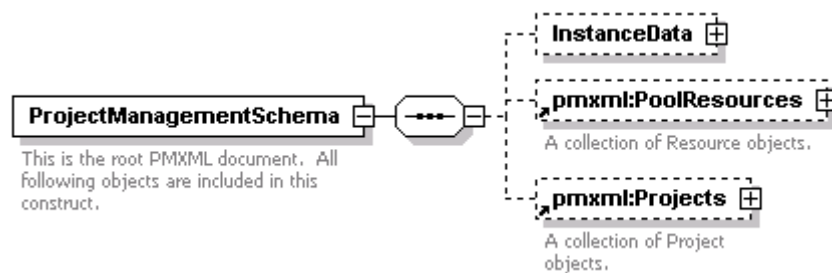


Fig. 1. The ProjectManagementSchema.

The InstanceData element is collection application-specific data that helps to interpret the project management data. It contains fields such as user ID and name, application name and version, and used schema version.

PoolResources is a collection of resource objects. These resource objects define material or working resources. They contain:

- Name and address data
- Cost data (rates, overtime cost, etc)

- Availability data
- Plan and baseline data
- Actuals
- Metrics (ACWP, BCWP, etc) and variances
- Customization and extension data

Projects are a collection of project objects. A project comprises:

- General data (name, company, status date, etc)
- Definitions for calculations (hours per day, days per month, etc)
- Plan and baseline data
- Actuals
- Metrics (ACWP, BCWP, etc) and variances
- Customization and extension data
- Resources (actually two collections, one for all resources, one for all local resources)
- Tasks
- Assignments

Task and assignment are structured similarly. Besides the usual groups for plan, baseline, actual data etc. the task definition contains attributes to define a milestone, a summary task and completion status. The assignment repeats also the plan, baseline etc.

```
<?xml version = "1.0" encoding="UTF-8"?>

<Schema name = "ProjectManagementSchema" xmlns =
"urn:schemas-microsoft-com:xml-data" xmlns:dt =
"urn:schemas-microsoft-com:datatypes">

  <description>The Project Management schema is de-
signed to provide an accurate summary of detailed pro-
ject data such as cost, schedule, and resource informa-
tion for a particular project. The schema represents
the data most commonly used by project management soft-
ware which is essential for accurate communication of
project status.</description>

  <AttributeType name="ProjectID" dt:type="int" />
  <AttributeType name="Name" dt:type="string" />
  <AttributeType name="Title" dt:type="string" />
  <AttributeType name="Description" dt:type="string" />

  <!-- here must include all of project attributes -->
</Schema>
```

**Schema Source 1.** Part of a ProjectManagement Schema Source (PMXML)

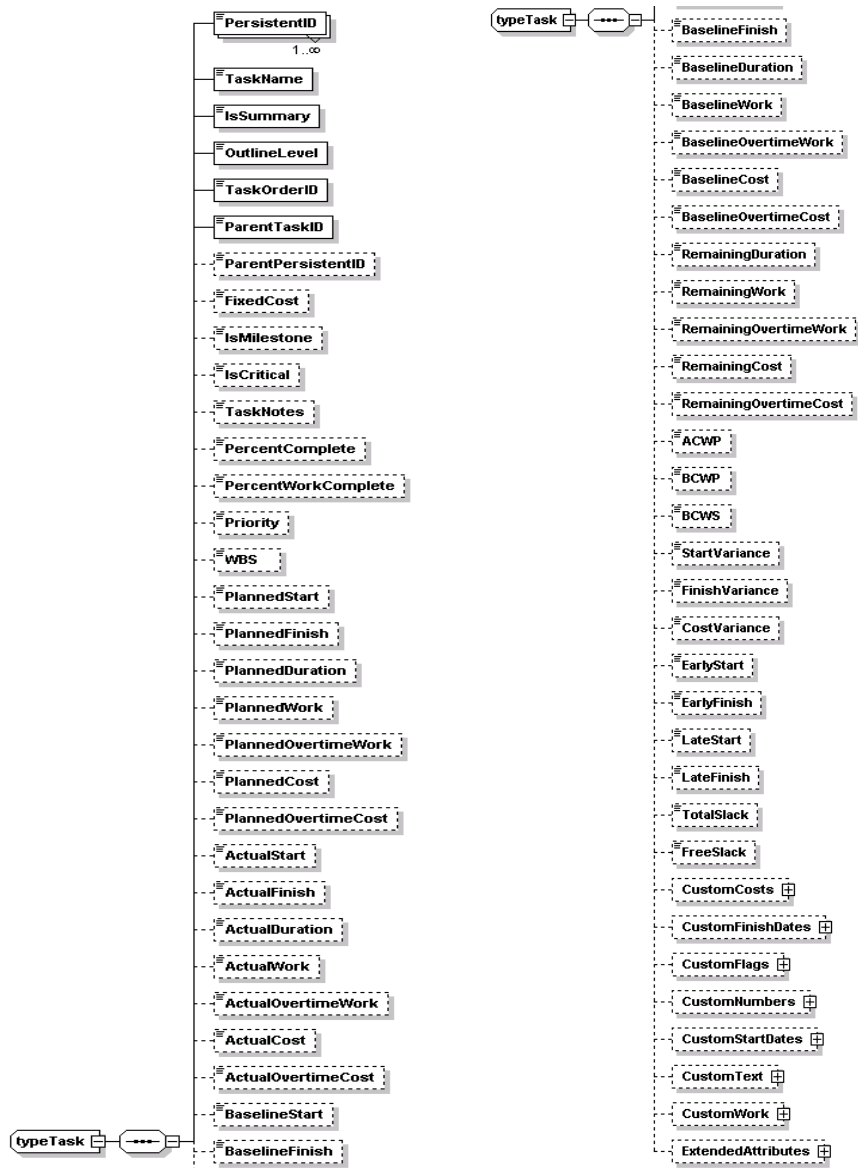


Fig. 2. PMXML version 2, task definition.

## 4 The PML solution

Despite PMXML [1] solve on a sufficiently and satisfactory way, the allotment of information in projects, tasks and resources between project systems and easy communication from/to any point of the world, continues to represent only a global and open solution.

In relation to the Project Knowledge Management (PKM), we have already a significant evolution with the introduction of a goal-giving system, a research of knowledge engine named as SPEAK (Search Process of Engineering for Assimilation of knowledge) [25].

SPEAK [25] is able to understand where it will find the information that it needs. But unhappily, the SPEAK [25] is not one human being. Therefore, we know that SPEAK [25], to have more success, has to know the semantic and grammar equivalent to the one of the Project Manager. When the Project Manager or the Project Team Leader write task description like:

*“Michael must develop video capture software with Visual Studio.Net 2005, in the University”*

We cannot assure that the same phrase is not written in hundreds of different forms, with the same felt. But most serious problem is that the description of a task could have a completely different interpretation from who it wrote to who reads it. This problem of communication exists between the human beings. Therefore, between one human being and a machine we have millions of more potential errors.

These possibilities inhibit for complete the use this kind of artificial intelligence search engines. In project management, if does not exist reasons to trust on the calculated information, then never will be taken in consideration.

Therefore, as the PMXML [21] do not solve the main problem of the GPRM [1]. For the GPRM [1], it is not important in which provider the specific task information came. What matters is to know, with the biggest severity, which is the esteem average time for the duration of one determined task. And the lesser time? And the greater? Which is the amount and quality of the sampling?

All of this information is important to determine:

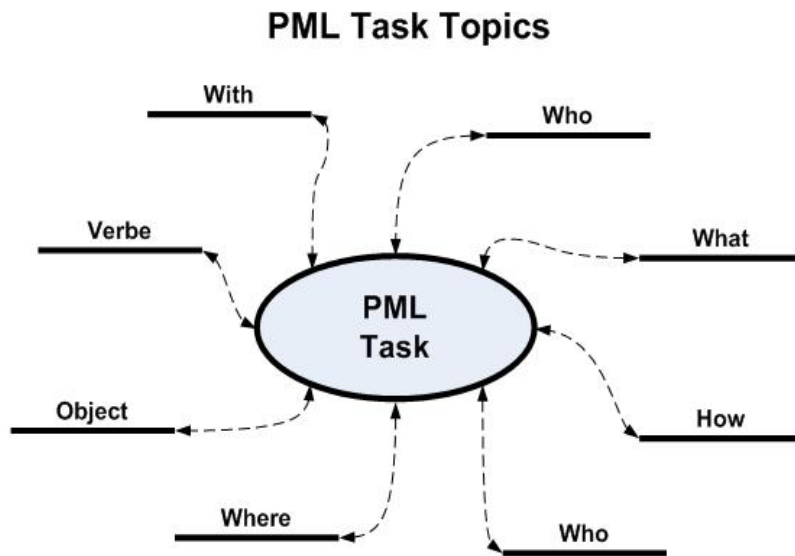
- Calculate the relative search
- Rigorous information
- Security and quality of information
- Quality of the supplier of information
- Other pertinent questions...

For such we have to look for correct information. The goal of this project is not to place one human being reading many pages of information, to take later one decision based in what it read. Also we do not intend that the Project Manager be the only detainer of the knowledge, placing in risk the proper project.

What it is really intended is to conceive a proper grammar for information treatment of the project management. This proper grammar gives the possibility to have a correct understanding between machines and humans.

After collecting on a database thousands of names of tasks typed by humans and used in many projects with Microsoft Project, we analyze this data and we verify that, of one everything forms generality has a similar semantic structure.

For example, when we intend to define our task, we use:

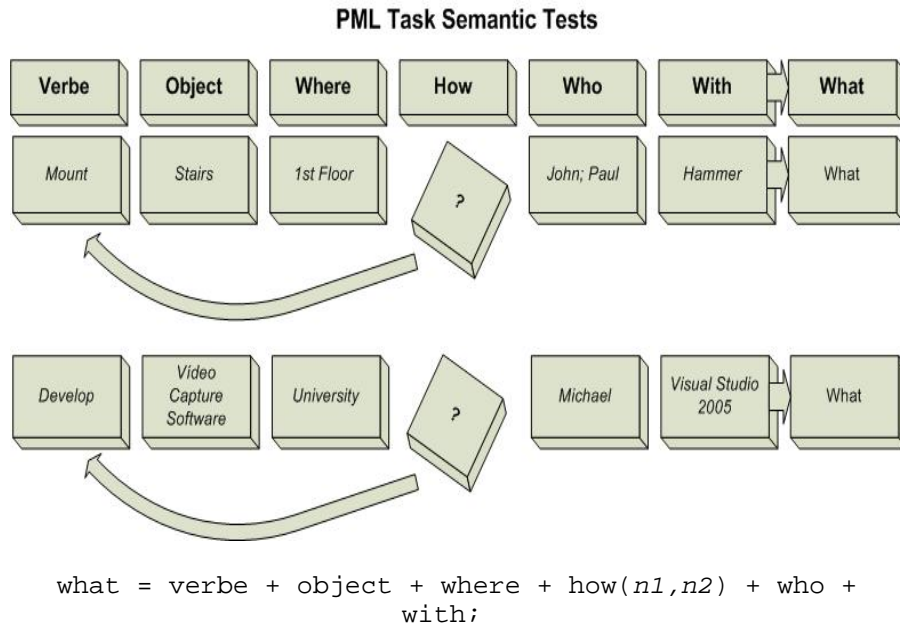


**Fig. 3.** PML Task Topic Elements

Any meta-information on a task, it will answer to:

- Verb (defines what it is really gone to do)
- Object (what it goes to be made through the verb)
- Where?
- How? In this case we can eventually come back to sub-tasks to explain his composition.
- With who?
- Where?

This defines the “What” to do Task to any task project.



**Fig. 4.** PML Task algorithmic graphic representation

```

<?xml version = "1.0" encoding="UTF-8"?>

<Schema name = "PML" xmlns =
"http://www.estgp.pt/gprm/pml" xmlns:pmxml =
"urn:schemas-microsoft-com:xml-data" xmlns:dt =
"urn:schemas-microsoft-com:datatypes"> <description>
The PML schema is designed to provide a better IA
search. PML is designed to be used by SPEAK or similar
systems, to provide a realy Project Knowledge Manage-
ment (PKM) </description>

<!-- ////////////////////////////////////// -->

<element name="Task" dt:type="int">

<sequence>
<element name="How"> <!-- /// ... /// --> </element>
<element name="Who"> <!-- /// ... /// --> </element>
<element name="With"> <!-- /// ... /// --> </element>
</sequence>

<AttributeType name="Verbe" dt:type="string" minOc-
curs=1 maxOccurs="unbounded" />
<AttributeType name="Object" dt:type="string" minOc-

```



```

curs=1 maxOccurs="unbounded" />
<AttributeType name="Where" dt:type="string" minOccurs=1 maxOccurs="unbounded" />

<!-- //////////////////////////////////////// -->

</Schema>

```

## Schema Source 2. Part of a Project Markup Language Schema Source (PML)

With our approach we want to make PML an extension to PMXML. This extension will be a way to conquer the WEB distributed projects including search engines. To make something, and to keep it useful and ship, we need to keep it simple. What we need is an engine to give us statistical values about projects. That engine only will be useful with simple data structures. If we want to search in very complicated structures we will get lost. PML is for us like DUBLINCORE [26] is for bibliographic descriptions. We want to find a set of primitives to describe an entire project with simplicity. These six primitives are based on the six primitives of Zachman Framework [27], redimensioned to Project Management reality, being the PM primitives.

```

<?xml version="1.0" encoding="UTF-8" ?>
- <pml:PML xmlns:pml="http://schemas.estgp.pt/xsd/grpm/pml/1.0/"
  xmlns="http://www.pacificedge.com/PMXML" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
+ <ProjectManagementSchema>
- <pml:PMLCore TaskID="1">
  <pml:Verb>Develop</pml:Verb>
  <pml:Object>Video Capture Software</pml:Object>
  <pml:Where>University</pml:Where>
- <pml:How>
  - <pml:PMLCore>
    <pml:Verb>Develop</pml:Verb>
    <pml:Object>Camera Driver</pml:Object>
  </pml:PMLCore>
  - <pml:PMLCore>
    <pml:Verb>Develop</pml:Verb>
    <pml:Object>MPEG Movie Builder From Driver</pml:Object>
  </pml:PMLCore>
  </pml:How>
  <pml:Who>Michael</pml:Who>
  <pml:With>Visual Studio 2005</pml:With>
</pml:PMLCore>
</pml:PML>

```

Fig. 5. A PML Sample

## 5 Conclusions and future work

PML it was drawn as an element of a complete framework solution (GPRM) to achieve the real Project Knowledge Management (PKM). To do that we are integrating PMXML inside PML and testing the communication between SPEAK Servers and GPRM Application.

We search for the capability to solve the problem of the knowledge management and new problems evaluation.

Soon we will develop better versions and extensions for PML. This first framework solution (GPRM+SPEAK+SUSI+PML) should be allow documents we will identify individually each one of the problems and also the correspondent solutions.

In this project we will follow the Tim Berners-Lee vision, working in web semantic solution for PML/SPEAK. We want to test an implementation of the RDF (Resource Description Framework).

## References

1. Paulo Gomes, "**GPRM (Global Project Research Management) Solution Overview**", ICEEEM - International Congress of Energy and Environment Engineering and Management, ISBN: 84-934089-9-9.
2. Paulo Gomes, Jorge Machado, "**GPRM: General Platform Presentation**", Superior School of Technology and Management, Polytechnic Institute of Portalegre.
3. Paulo Gomes, Jorge Machado, "**Client Interface and connection to other applications**" - Superior School of Technology and Management – Polytechnic Institute of Portalegre.
4. Paulo Gomes, Jorge Machado, "**GPRM Dictionary definition and inter-connection of their entities**" - Superior School of Technology and Management, Polytechnic Institute of Portalegre.
5. Paulo Gomes, Jorge Machado, "**Instantiation, evolution, behavior e learning in GPRM**". - Superior School of Technology and Management, Polytechnic Institute of Portalegre.
6. Paulo Gomes, Jorge Machado, "**Normalization and Internationalization of technical terms in GPRM**". - Superior School of Technology and Management, Polytechnic Institute of Portalegre.
7. José Borbinha, Jorge Machado, "**Mitra: A semantic web indexer**", Instituto Superior Técnico, Universidade Técnica de Lisboa, 2005.
8. José Borbinha, Jorge Machado, "**DEPTAL: A Framework for Institutional Repositories**", DELLOS workshop, Crete, Greece, 2005 <http://deptal.estgp.pt/4>
9. Microsoft. "**Planning, Implementing and Mantaining a Microsoft Windows Server 2003 Active Directory**". Microsoft MOC 2279, 2003
10. Microsoft. "**Planning, Deploying and Managing an Enterprise Project Management Solution**". Microsoft MOC 2732A, 2003
11. Shtub, A. / Bard, J. / Globerson, S. "**Project Management Engineering, Technology and Implementation**", Ed. Prentice Hall, 1994.
12. <http://www.gprm.org>, Global Project for Research Management official web site
13. <http://mitra.bn.pt>, Semantic Indexer of described web contents, National Library of Portugal
14. <http://www.openarchives.org>, OAI-PMH. Open Archives Initiative - Protocol for Metadata Harvesting
15. <http://www.adlnet.org/index.cfm?fuseaction=scormabt>, SCORM. The Sharable Content Object Reference Model
16. <http://www.loc.gov/z3950/agency>, Z39.50. Z39.50 Maintenance Agency
17. <http://www.loc.gov/z3950/agency/zing/srw>, SRU Search and Retrieve URL
18. <http://www.udcc.org/mrf.htm>, Unitary Decimal Classification

19. <http://xml.coverpages.org/MembersJoin-PMXML20020603.html>,  
Announcement 2002-06-03: "*Pacific Edge Software Adds Industry Leaders to PMXML Consortium. NASA, Oracle among New Members of PMXML Standards Organization.*"
20. <http://www.pacificedge.com/xml/>, PMXML website
21. [http://groups.yahoo.com/group/pm\\_xml](http://groups.yahoo.com/group/pm_xml), Project Management XML Discussion Group
22. Andrew Goloboy and David A. Shiang, "*Pacific Edge's Project Office: Improving Enterprise Initiatives.*", IDC White Paper, 2000.
23. <http://www.pacificedge.com/>, Pacific Edge Software web site
24. <http://www.biztalk.org>, Microsoft BizTalk web site
25. Paulo Gomes, Jorge Machado, "*SPEAK (Search Process of Engineering for Assimilation of knowledge), A generic framework to search in Metadata*", Superior School of Technology and Management, Polytechnic Institute of Portalegre.
26. <http://dublincore.org>, The Dublin Core Metadata Initiative.
27. Zachman, John, "Enterprise Artifacts vs. Application Artifacts".