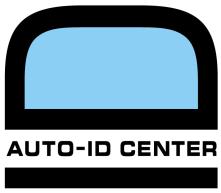
Published June 1, 2002. Distribution restricted to Sponsors until September 1, 2002.



TECHNICAL MEMO

Physical Mark-Up Language Update

Christian Floerkemeier & Robin Koh

AUTO-ID CENTER MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 77 MASSACHUSETTS AVENUE, BLDG 3-449, CAMBRIDGE, MA 02139-4307, USA

ABSTRACT

The objective of this briefing is to give an overview of the efforts to develop a Physical Mark-up Language (PML). The main goal of the Physical Mark-up Language is to provide a common, standardized vocabulary to represent and distribute information related to Auto-ID enabled objects. In this document, the types of information modeled in this vocabulary and their main usage scenarios are discussed. This brief also describes the division of the development effort into a PML Core component and a PML Extension component. The former focuses on developing a vocabulary to model the data directly generated by the Auto-ID infrastructure – such as location and telemetry information. The work related to the PML Extensions leverages existing developments in the world of e-commerce standards to combine the low-level, instance-specific Auto-ID generated data with high-level product- and process-related information.

TECHNICAL MEMO

Physical Mark-Up Language Update

Biography

by Christian Floerkemeier
Doctoral Candidate, M-Lab ETH Zurich

Christian Floerkemeier is currently a doctoral candidate at the M-Lab (www.m-lab.ch), a joint initiative of ETH Zurich and the University of St. Gallen to promote the application of pervasive computing in business environments. He holds a Bachelor and Masters degree in Electrical and Information Science from Cambridge University in the UK. His work at the Auto-ID Center focuses on the development of the physical mark-up language and the underlying information models.

by Robin Koh
Associate Director

Robin Koh has worked in the field of logistics and supply chain management for the last 11 years. He has held Manager and Director positions at Pepsi-Cola and Arrow Electronics. He has a Masters in Engineering from the Massachusetts Institute of Technology, a Bachelors in Industrial Engineering and Operations Research from the University of Massachusetts at Amherst and an MBA from the Tuck School of Business at Dartmouth College.

TECHNICAL MEMO

Physical Mark-Up Language Update

Contents

1. BACKGROUND

The Auto-ID Center's proposed Electronic Product Code (EPC) system brings new functionality to the traditional role of Automatic Identification and Data Capture (AIDC) technology. It will be possible to uniquely identify items, trace and locate objects through the supply chain and access telemetric information related to the object such as pressure and temperature. This helps build a comprehensive picture about a particular object and the processes it has gone through.

The Physical Mark-up Language (PML) plays a major role in enabling this new functionality. It will provide a standardized method to describe physical objects, processes and environments that can be used for software development, data storage and analytic tools in industry and commerce. It will enable a dynamic environment where static, temporal, dynamic and algorithmic data about objects can be exchanged.

2. OBJECTIVES OF THE PHYSICAL MARK-UP LANGUAGE

The main goal of the physical mark-up language is to provide a common, standardized vocabulary to represent and distribute information related to Auto-ID–enabled objects.

2.1. Type of Information described in PML

Information directly captured from the Auto-ID infrastructure is modeled as part of the physical mark-up language.

Examples include:
- Location information e.g. tag X was detected by reader Y, which is located at loading dock Z
- Telemetry information
    - Physical properties of an object e.g. its mass
    - Physical properties of the environment, in which a group of objects is located e.g. ambient temperature
- Composition information e.g. the composition of an individual logistical unit made up of a pallet, cases and items

The information model will also include the history of the various information elements listed above e.g. a collection of the various single location readings will result in a location trace.

On the other hand, the objective of PML development is to provide complete information about the object and to enable transactions. This requires information not captured by Auto-ID infrastructure directly, but aggregated from other sources.

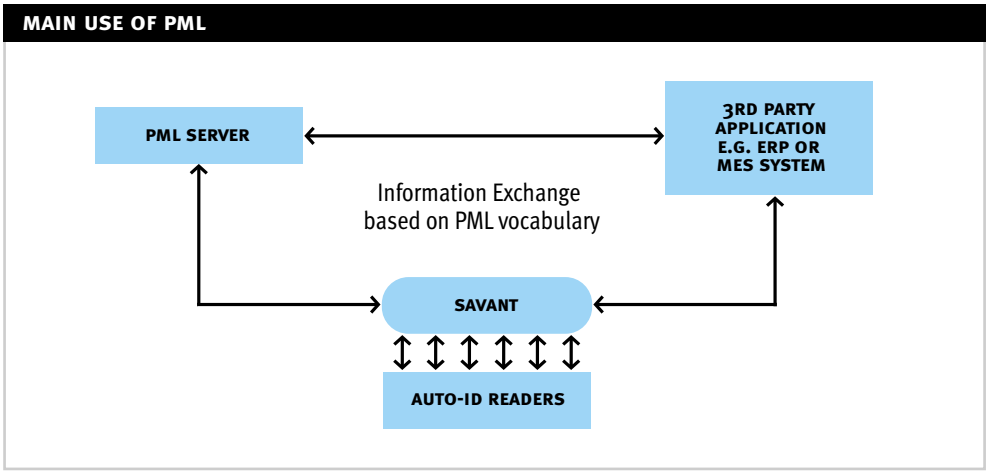
Among others this could include:
- Product related information e.g. item information in the retail industry or technical datasheets in the high-tech industry
- Process related information e.g. a link between an Auto-ID enabled object and an appropriate transaction like an advanced shipping notice

The rationale to include both information which does and does not originate from the Auto-ID infrastructure is that only the knowledge of both can trigger the appropriate actions. For example temperature monitoring of a set of objects, combined with the product storage information allows for effective monitoring and exception handling of temperature-controlled goods.

2.2. Use of PML in relation to other components of the Auto-ID infrastructure

The main use of the PML language is to act as a common interface between the various components in the Auto-ID infrastructure. Figure 1 illustrates an example showing the Savant, a third party application such as an Enterprise Resource Planning (ERP) or Manufacturing Execution System (MES) and the PML Server acting as storage for the Auto-ID related data.

Figure 1: PML as the interface between the various Auto-ID components



Note that the PML vocabulary makes no recommendation on how the information is actually stored, since it only specifies how the information is exchanged between the various components.

3. DESIGN

To facilitate the orderly development of the Physical Mark-Up Language (PML), research has been divided initially into two primary sections – PML Core and PML Extensions (see Figure 2).

PML Core provides the common standardized vocabulary to distribute information directly captured from the Auto-ID infrastructure e.g. location, composition and other telemetric information. As this level of data was not readily available before Auto-ID, PML Core has to be developed to represent it.

PML Extensions are used to integrate information that is not generated by the Auto-ID infrastructure and is aggregated from other sources. The first extension that will be implemented is the PML Commerce Extension. The PML Commerce Extension involves the rich choreography and process standards that enable transactions within and between organizations to take place. Many organizations are already working on these standards and Auto-ID will evaluate and integrate the ones that best fit its users' requirements.

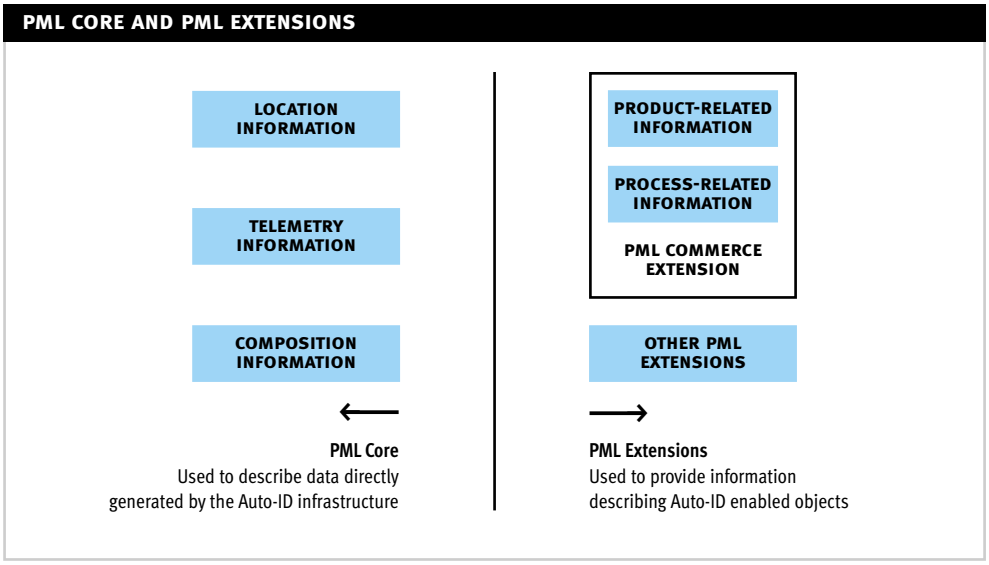


Figure 2: Division of PML developments into PML Core and PML Extensions

The PML Core effort, which focuses on the data directly generated by the Auto-ID infrastructure, can be characterized as containing largely instance-specific and domain-independent information. The term instance-specific relates to the fact that (e.g. a location) it is specific to an individual Auto-ID enabled object as opposed to a whole class of objects. The term domain-independent indicates that the way data is modeled is not dependent on the industry or business process the specific object is involved in.

For the PML Commerce Extension component, most of the information provided is applicable to a whole class of objects. Most of the information content is also highly dependent on the actual industry. In the high-tech industry for example the technical data sheet of a component is far more common than in other industries. The PML Commerce Extension is, to a large extent, class-specific and domain-dependent.

So far the focus has been on integrating existing e-commerce standards in the PML Extension framework. One could of course also imagine other PML extensions covering different domains.

4. IMPLEMENTATION APPROACH

The World Wide Web Consortium's (W3C) XML Schema specification received a mixed response from the target audience, when it was first released. While some people like the richer structure and semantics that can be defined with these new schemas, others complained that the resulting schemas make it difficult to share the information with a wider audience of users and business partners.

The PML development team decided to regard XML Schemas as the implementation syntax for the PML and rely on the Unified Modeling Language (UML) to represent the model and share the PML definitions with its users. UML was chosen since it was perceived as a widely adopted standard for system specification and design. This approach will allow us to benefit from the advantages of XML Schemas over DTDs and at the same time still enable us to easily share the definitions and underlying models with a wider audience.

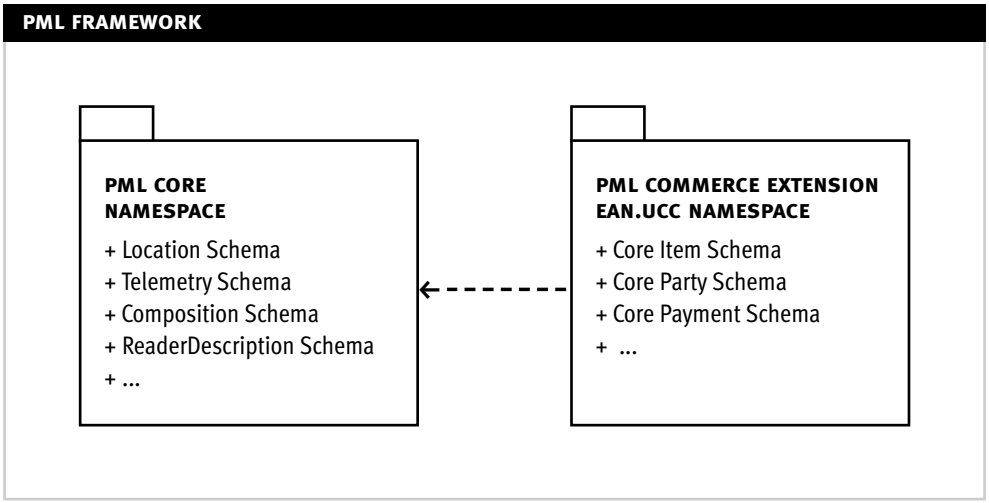
4.1. Framework for PML Core and PML Extensions

The separation into PML Core and PML Extensions requires a framework in which the various extensions can be integrated. This framework needs to be compatible with the existing standards that are targeted. It should also be designed in such a way that extensions can be included at a later stage without requiring a redesign of the existing components.

The original XML specification did not include a way to combine several vocabularies when composing a document. This capability is however essential if the reuse of industry standard vocabularies is to be promoted rather than forcing each application to reinvent the same definitions. The XML Namespace specification was written to address this requirement.

The framework to support PML Core and PML Extensions will be based on a combination of XML Schemas and Namespaces (see Figure 3). The XML Schemas define and document the vocabulary. They also intend to allow for a straightforward validation of structural and semantic accuracy. The XML Namespaces enable the reuse of existing XML-based e-commerce standards within the framework.

Figure 3: Example of a XML Schema and Namespace based PML organization in a UML package notation



5. PROPOSED NEXT STEPS

- Finalize XML Schemas for PML Core elements
- Test and review XML schemas including the documentation provided
- Continue evaluating e-commerce standards
- Integrate revised PML in Field Test

APPENDIX

Overview of E-Business Standards

In order to position the various E-Business standardization efforts it is of considerable value to use a conceptual model. The main components of a conceptual model for E-Business standards were identified as being a:

- Messaging Service (e.g. Transfer Routing Protocols like SOAP, RNF2.0, and ebMS)
- Registry/Repository (e.g. UDDI, ebXML)
- Technical Dictionary (e.g. RosettaNet Technical Dictionary)
- Business Dictionary (e.g. EAN.UCC Business Message Standard)
- Business Processes (e.g. the various PIPs in RosettaNet, Global Master Data Synchronization in GCI and UCCNet)

The following questions provide further guidance when evaluating today's E-Business standards:

- What is the current status (development, pilot or deployed)?
- Are the components supply-chain specific or domain-independent?
- To which extent are the various components present (e.g. does the standard build on a single business process or on a whole cluster of processes)?

Figure 4 positions the various E-Business standards that were investigated in the conceptual model. Although the PML Development Team attempted to objectively reflect current standards in this simplified model, other interpretations are certainly possible. Figure 4 also shows how PML will be leveraging on the common components of today E-Business standards indicated by the arrow. The figure also indicates how they get augmented with an Auto-ID data dictionary called PML Core.

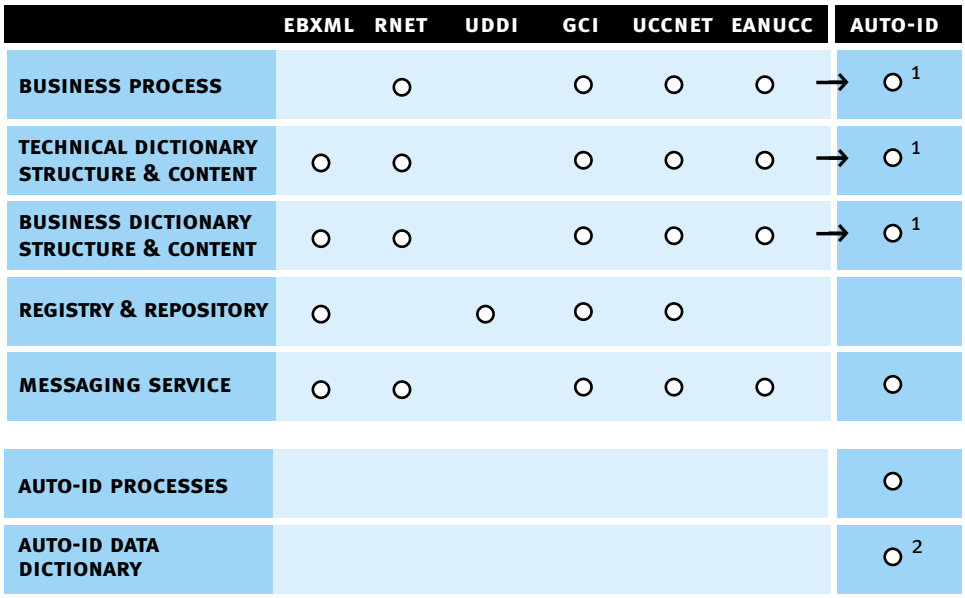


Figure 4: Various E-Business standards in a conceptual model

[1] PML Extensions

[2] PML Core

Designed and produced by Foxner. www.foxner.com