Physical Markup Language
Version 1.0

David Brock, Director
Auto-ID Center
Massachusetts Institute of Technology
University of Cambridge
EPC - electronic product code

PML - physical markup language

Electronic tags, readers and sensors

ONS - Object Name Service
Objective of PML

Common language for describing physical objects, systems, processes and environments.
Background

Generality

- Addresses largest numbers of industries
- Encourages software development
- Describes common characteristics of physical objects
- Encourages inter-industry cooperation and information transfer.
Background

Simplicity

- Simple standards encourage adoption, reduce learning curves and increase audience.
Pathway

- Rather than a single standard, PML will proceed through a series of planned iterations.

- A complex language will limit learning and a simple language would be insufficient.

- A series of increasing sophisticated releases will allow familiarity to grow with capability.
Background

**Timestamps**

- Static, temporal and dynamic are time dependent views of generic data.

- Time stamps, durations and frequencies will be provided for all data elements in the Physical Markup Language.
Nomenclature

- PML will avoid verbose names for data types.
- Efficient mnemonics designed by and for software developers.
Background

Robust

• PML will operate effectively with incomplete or intermittent information.

• The language will have to support alternatives and approximations to inaccessible data.
Background

Units

- PML will adopt a single standard for weights and measures.

- Translation software in both the PML editors and viewers will provide the necessary conversion to familiar standards.
PML will use the Extensible Markup Language (XML) as the method to store and transmit data.

General XML utilities, viewers and validating software exist to parse, modify and access XML files.
# Background

## PML Version 0.1 beta Field Trial

### Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Measurements</td>
<td>6</td>
<td>Image</td>
</tr>
<tr>
<td>1.1</td>
<td>VAL Element</td>
<td>6.1</td>
<td>IMG Element</td>
</tr>
<tr>
<td>2</td>
<td>Date</td>
<td>7</td>
<td>Datum</td>
</tr>
<tr>
<td>2.1</td>
<td>DATE Element</td>
<td>7.1</td>
<td>DATUM Element</td>
</tr>
<tr>
<td>3</td>
<td>Location</td>
<td>8</td>
<td>Reading</td>
</tr>
<tr>
<td>3.1</td>
<td>LOC Element</td>
<td>8.1</td>
<td>READ Element</td>
</tr>
<tr>
<td>4</td>
<td>Ownership</td>
<td>9</td>
<td>Node</td>
</tr>
<tr>
<td>4.1</td>
<td>OWNER</td>
<td>9.1</td>
<td>NODE Element</td>
</tr>
<tr>
<td>5</td>
<td>ePC</td>
<td>10</td>
<td>PML</td>
</tr>
<tr>
<td>5.1</td>
<td>EPC Element</td>
<td>10.1</td>
<td>PML Element</td>
</tr>
</tbody>
</table>
Single representation

- PML uses a single representation for information elements
Exact not abstract representation

• PML specifies an exact representation of data types.
Developers Tool

- PML is designed and implemented primarily as a tool for software developers.
Modular building blocks

- PML contains modular elements that can be mixed and matched in higher-level data structures.