Collaboration on Planning and Scheduling

Yasuyuki Nishioka, Prof. Dr.
Hosei University,
PSLX Consortium Japan
nishioka@k.hosei.ac.jp
Outline

• Scope and Objectives
• Introduction of PSLX Consortium Japan
• PSLX Technical Specifications
• APS Agent models
• Domain Object models
• XML Schema Specification
• Experimental Applications
• Summary
Scope and Objectives
Hierarchical Levels

1. **Level 4**: Business planning and logistics
   - Target of this specification

2. **Level 3**: Manufacturing operations and control

3. **Level 2**: Continuous control, Batch control, Discrete control

   - Level 2
   - Level 1
   - Level 0
Scope of APS

- Supplier’s Design
- Supplier’s APS
- Supplier’s Manufacturing
- Design
- APS
- Manufacturing
- Customer’s APS
- Customer’s Design
- Customer’s Manufacturing

Supplier
Maker
Customer
External interfaces

- Supplier’s Design
- Supplier’s APS
- Supplier’s Manufacturing
- APS
- Design
- Customer’s APS
- Customer’s Design
- Customer’s Manufacturing

Interfaces (external)
Internal interfaces

APS

Interfaces (internal)

APS agent
Terms and definitions

• Planning
  To find suitable goals and operations to fulfill the gap between desires and facts, by clarifying their well-formed structures and parameters

• Scheduling
  To clarify relationship among production items and manufacturing resources in the time horizon, considering various constraints and objectives

• APS (Advanced Planning and Scheduling)
  A system architecture of planning and scheduling integration to dynamically and synchronously achieve the goal of each manufacturing enterprise
Introduction of PSLX Consortium Japan
What is PSLX

- Reference Architecture for APS software development
- Communication Protocols and Ontology among APS software
- XML Schema for Planning and Scheduling problems
- Common Terminology of Planning and Scheduling problems
PSLX Consortium Japan

- Founded in July 2001
- Chairman Prof. Kazuhiko Yasuda
- Board members (51)
  - IT vendor (16)
  - System integrator (18)
  - Consulting firm (5)
  - Manufacturer (4)
  - Research Institute (8)
- Recommendation specification will be published in June 2003
The Objective of the consortium is to establish APS standard for Japanese Manufacturing and support world-wide manufacturers to implement our recommending APS systems. As results of our activities, we hope that Japanese manufacturing knowledge and IT based management are combined to Integrate for the next generation’s manufacturing industry.
Board Members

Asprova Corporation
ILOG Co., Ltd
KOZO KEIKAKU Engineering Inc.
Toyo Engineering Corporation
M-2-M Inc.
ISAC, Inc.
e-Manufacturing Co., Ltd.
Process Chain Management Research Institute
NK-EXA Corporation
JT Engineering inc.
CIMTOPS Corporation
Sumitomo Metal System Solutions Co., Ltd.
NS Solutions Corporation
System Plaza Inc.
NEC Nexsolutions Ltd
APPLIED TECHNOLOGY Co., Ltd
LOGIX JAPAN Co., Ltd.
Hitachi Ltd.
NEXTECH CORPORATION
REEDREX Corp.
Fujitsu Limited

Hitachi Engineering Co., Ltd
Hitachi Tohoku Software, Ltd.
Kawatetsu Systems, Inc.
Future Knowledge Consulting Co., Ltd.
AlphaPurchase Co., Ltd.
IWAI KiKAI-KOGYO Co., Ltd
MASP Association
Lightwell Co., Ltd
TOYO Business Engineering Corporation
Hitachi Information & Control Systems, Inc.
FFC Limited
CIMX Co., Ltd
Yokogawa Electric Corporation
Hosei University
Hiroshima Prefectural University
Nagoya Institute of Technology
Ashikaga Institute of Technology
Tohoku University
MUSASHI University
Japan Society for the Promotion of Machine Industry
Sofix Co., Ltd.
Scheduling software provided by the PSLX consortium members

ACCROAD/Logics Japan, Co.,Ltd.

ASPROVA / ASPROVA Corporation

DIRECTOR / CIMTOPS Corporation

JoyScheduler / JT Engineering Inc.
PSLX Technical Specifications
Draft Recommendation

• PSLX-01 : Grand Design for Manufacturing Enterprises
• PSLX-02 : APS Agent models
• PSLX-03 : PSLX Domain Objects
• PSLX-04 : XML Specification and Data Exchange
• PSLX-05 : PSLX Common Dictionary

(These will be approved in June 2003)
APS Architecture

Level 5  Business model layer
Level 4  Collaboration layer
Level 3  APS agent layer
Level 2  Domain object layer
Level 1  Data implement layer

<table>
<thead>
<tr>
<th></th>
<th>PSLX-01</th>
<th>02</th>
<th>03</th>
<th>04</th>
<th>05</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business model layer</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Collaboration layer</td>
<td>✓ ✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>APS agent layer</td>
<td>✓ ✓ ✓ ✓ ✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Domain object layer</td>
<td>✓ ✓ ✓ ✓ ✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Data implement layer</td>
<td>✓ ✓ ✓ ✓ ✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
APS Agent models
APS agent models

External interfaces

Internal interfaces

(external)

(active)

(passive)

(external)

(external)

(active)

(active)

(active)

(active)

(active)

APS
Agent models (example)

- Strategic agent
- Scheduling agent
- Planning agent
- Product design agent
- Process design agent
- Purchasing agent
- Sales agent
- Accounting agent
- SCM agent
- Transportation agent
- Capacity ctrl agent
- Inventory ctrl agent
- MES agent
- Maintenance agent
- BOM agent
- Option ctrl agent
- Pegging ctrl agent
- Federation mgmt agent
## External interface (passive)

<table>
<thead>
<tr>
<th>Interface name</th>
<th>Customer</th>
<th>Supplier</th>
<th>Design</th>
<th>Mfg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>setPlan</td>
<td>□</td>
<td>□</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getPlan</td>
<td>□</td>
<td>□</td>
<td></td>
<td></td>
</tr>
<tr>
<td>setSchedule</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>getSchedule</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>setProduct</td>
<td></td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>getProduct</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>setProcess</td>
<td></td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>getProduct</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>setOrder</td>
<td>□</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>getOrder</td>
<td>□</td>
<td></td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>setOption</td>
<td>□</td>
<td>□</td>
<td></td>
<td></td>
</tr>
<tr>
<td>getOption</td>
<td>□</td>
<td>□</td>
<td>□</td>
<td>□</td>
</tr>
<tr>
<td>Interface name</td>
<td>Customer</td>
<td>Supplier</td>
<td>Design</td>
<td>Mfg.</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------</td>
<td>----------</td>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>setEstimation</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>getEstimation</td>
<td>○</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>setPromise</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>getPromise</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>setClaim</td>
<td>○</td>
<td></td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>getClaim</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>setProgress</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>getProgress</td>
<td>○</td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SetEmergency</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>getEmergency</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>setStock</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>getStock</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
</tbody>
</table>
# External interface (cont.)

<table>
<thead>
<tr>
<th>Interface name</th>
<th>Customer</th>
<th>Supplier</th>
<th>Design</th>
<th>Mfg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>setLoad</td>
<td>○</td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>getLoad</td>
<td>○</td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>setCapacity</td>
<td></td>
<td>○</td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>getCapacity</td>
<td>○</td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>setLot</td>
<td></td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>getLot</td>
<td></td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>setTask</td>
<td></td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>getTask</td>
<td></td>
<td></td>
<td></td>
<td>○</td>
</tr>
</tbody>
</table>
### Internal Interface (Scheduling agent)

<table>
<thead>
<tr>
<th>Interface name</th>
<th>Interface name</th>
<th>Interface name</th>
</tr>
</thead>
<tbody>
<tr>
<td>initSchedule</td>
<td>setOrder</td>
<td>setCapacity</td>
</tr>
<tr>
<td>makeSchedule</td>
<td>getOrder</td>
<td>getCapacity</td>
</tr>
<tr>
<td>setSchedule</td>
<td>setOption</td>
<td>setLot</td>
</tr>
<tr>
<td>getSchedule</td>
<td>getOption</td>
<td>getLot</td>
</tr>
<tr>
<td>setParty</td>
<td>setProgress</td>
<td>setTask</td>
</tr>
<tr>
<td>getParty</td>
<td>getProgress</td>
<td>getTask</td>
</tr>
<tr>
<td>setProduct</td>
<td>setStock</td>
<td></td>
</tr>
<tr>
<td>getProduct</td>
<td>getStock</td>
<td></td>
</tr>
<tr>
<td>setProcess</td>
<td>setLoad</td>
<td></td>
</tr>
<tr>
<td>getProcess</td>
<td>getLoad</td>
<td></td>
</tr>
</tbody>
</table>

Interface arguments are defined using PSLX domain objects.
Domain Object model
## Top level objects

<table>
<thead>
<tr>
<th>Object name</th>
<th>Object name</th>
<th>Object name</th>
</tr>
</thead>
<tbody>
<tr>
<td>item</td>
<td>operation</td>
<td>order</td>
</tr>
<tr>
<td>substance</td>
<td>mode</td>
<td>party</td>
</tr>
<tr>
<td>feature</td>
<td>schedule</td>
<td>rule</td>
</tr>
<tr>
<td>state</td>
<td>progress</td>
<td>sd</td>
</tr>
<tr>
<td>subject</td>
<td>precedence</td>
<td>pegging</td>
</tr>
<tr>
<td>calculation</td>
<td>interval (switch)</td>
<td>tracking</td>
</tr>
<tr>
<td>plan</td>
<td>action</td>
<td></td>
</tr>
<tr>
<td>constraint</td>
<td>condition</td>
<td></td>
</tr>
<tr>
<td>event</td>
<td>produce (consume, assign)</td>
<td></td>
</tr>
<tr>
<td>occurrence</td>
<td>lot (task)</td>
<td></td>
</tr>
</tbody>
</table>

( ) is a special case
Ontology model (PSLX-04)
Top level objects

- PSLX domain objects are ontology in representing interface arguments
- Each data schema of practical systems can be generated by this object model
- Generating operators are:
  - create subclass, divide class, merge class, create attribute class, create relation class, add/delete attribute, move attribute through class relation, add/delete relation, restrict multiplicity, add constraints, change name according to PSLX common dictionary
Sub classes

- item -- resource, product, material, wip, subassy
- resource – equipment, tool, labor, ws, shop, site
- feature – stock, load, capacity, location
- event – start, end, halt, resume
- operation – fabrication, transportation, storage, inspection, setup, maintenance
- party – customer, supplier
- calculation – cost, profit
XML Schema for PSLX
Schema mapping

Domain objects

- schedule
- operation
- precedence
- occurrence
- item
- stock
- resource
- event
- interval
- order

XML schema

- operation
- interval
- predecessor
- successor
- event
- item
- order
- resource
- stock
Sample XML specification
Experimental Applications
PSLX interface module
(copy free software)

Client module

Local PC

internet

Apache
Tomcat

Site server

HTTP/SOAP

RMI

Server module

Remote PC

beta version in Jan 2003

C, C++, VB, Java, C#
Scheduling software provided by the PSLX consortium members

ACCROAD/Logics Japan, Co.,Ltd.

ASPROVA / ASPROVA Corporation

DIRECTOR / CIMTOPS Corporation

JoyScheduler / JT Engineering Inc.
Show Dispatching Order

Rescheduling/Display Gantt Chart

Send Progress Data
Industrial Applications

• Project 1-A (started in Aug 2002)
  – Resource booking system for a mechanical fabrication manufacture

• Project 1-B (started in Aug 2002)
  – Stock information sharing with a 3rd tire automotive parts supplier

• Project 3 (started in Sep 2002)
  – Web based supply chain planning for one of the kind production

• Project 2 (started in Nov 2002)
  – Remote maintenance using MES and scheduler integration
Summary
How to use PSLX

• Package vendors develop their software product for shop floors, e.g. scheduler, according to PSLX standard
• ERP vendors make their database schema using the PSLX domain object model, and clarify the mapping processes.
• System Integrators develop mapping modules to integrate PSLX software and legacy systems.
Benefit of PSLX for APS

- Schedule can be viewed graphically from different places outside the factory.
- Planning and Scheduling Problems of different business components can be federated.
- Manufacturing execution systems can be developed with a common interface for schedulers or shop floor software.
- Many kind of integration with ERP will be dynamically available for floor level management.
- APS system has more design flexibility by using scheduler as a connectable component.
Web-based App Integration by PSLX

Transport Product/Process/Service data on Internet using XML specifications
Thank you

more information is on
http://www.pslx.org
(English pages will be available by July 2003)