DCML: The Missing Link in IT Service Management

J. Darrel Thomas
Chief Technologist, Hosting
Electronic Data Systems Corporation
The only open, XML-based standard designed to achieve interoperability by providing a systematic, vendor-neutral way to describe an IT Service environment.

Universal language that describes elemental, process, and service-oriented relationships between IT service entities and policies governing the management of such environments.

Handles heterogeneous and semantic information required to manage at the service level.
The Problem DCML Helps Solve

- IT Services Management requires a multi-dimensional approach
- Mapping of interdependencies and inter-relationships at the elemental, process, and services level does not currently exist
- Complex, heterogeneous environments require interchange standardization between management systems
- DCML helps bridge this disparity using ontological and “like” meta-relationships
IT Services Management Standardization Has Varying Domain-Based Initiatives to Address Its Dimensions...

Many standards have been or are being developed to address the INDIVIDUAL DOMAIN issues...

Service Desk
Incident
Problem
Release
Change
Configuration
Service Level
Financial
Capacity
Continuity
Availability

Service-Oriented

Business Process (BPEL, ebXML)
Application (.NET, Java, SOA, WS, UDDI, etc.)
Industry MDA, HIPPA, etc
IT Foundation (LDAP, SMTP, etc)
Etc.

Servers (CIM-SMASH)
Storage (CIM-SMIS)
Purposeful Infrastructure (WSDM2.0, WS-*)
Operating Systems (CIM-SMASH)
Network (CIM)
Applications (CIM)

©2003-2005 DCML Member Section, All Rights Reserved
But, There is STILL Confusion with Possibly The Most Important IT Services Dimension – Interrelationships, Configurations, and Dependencies…

This area still looms as the enterprise/service provider’s most important inhibitor for standardizing IT Services Management.
Why Vocabularies, Semantics and Ontologies are Important to DCML

- Provide a solution that facilitates automated reasoning about data - DCML documents do not merely describe static information about the simple state of an environment
- Flexible enough to be applied at the framework specification level of DCML
- The best choice for managing data center interdependency complexity, which DCML is intended to help solve
The Semantic Web
Tim Berners-Lee, Director
W3C World Wide Web Consortium

Ontology Layer

From http://www.w3.org/2002/Talks/04-sweb/Overview.html
Example: Server Provisioning Script

A service is being deployed and part of the deployment requires a web-server to be created. A service architect creates a service blueprint that specifies how the service is to be created – what servers it needs to construct the service and what software installed on these servers. This blueprint is codified in DCML. An automation system or tool orchestrates the construction of the service by passing along the blueprint (either in whole or in part) to various provisioning systems.

```xml
<owl:Class rdf:ID="LogicalServerBlueprint">
  <rdfs:subClassOf rdf:resource="&dcml;Configuration"/>
  <!-- Restrict os to cardinality of 1 -->
  <rdfs:subClassOf><owl:Restriction>
    <owl:onProperty rdf:resource="#os"/>
    <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
  </owl:Restriction></rdfs:subClassOf>
</owl:Class>
<owl:ObjectProperty rdf:ID="os">
  <rdfs:range rdf:resource="#OperatingSystem"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="package">
  <rdfs:range rdf:resource="#SoftwarePackage"/>
</owl:ObjectProperty>
```
Why OWL and RDF are a Good Fit for DCML

- Conceptual Model – OWL captures knowledge of both current and future managed environments, and the policies and procedures governing the management of those environments (i.e. two layers, a core schema and extension schemas)

- Encoding Scheme – OWL allows DCML to be encoded into a concrete representation before it is exchanged with other management systems, written to a file, or transported by a protocol

- Instance Document – RDF/XML provides an extensible and flexible structure to capture large data populations
DCML Enables IT As A Service
DCML could create dynamic instrumentation and interface relationship mapping for the ITIL Configuration Management process, Datacenter/Infrastructure CMDB, and standardization/modeling and specifications for Configuration Items (CIs), both discovered during runtime as well as created during development pre-production integration.

Would enable ITIL to be electronically and standards-based within a ITIL-framed operational model.

Could define a standards-based data format that allows disparate IT systems to implement ITIL Service Support and ICTIM processes and exchange information with each other.

Could give IT a concrete step for implementing ITIL and automating IT processes.
Goals of DCML

- Create standards-based information exchange data for disparate management systems
- Instrument interrelationships and mappings to dynamically store and process configurations (CMDB, configuration entities/items)
- Instrument best practices models and IT process frameworks, such as ITIL, in a form that can be used by customers
- Provide a solution in pre-production CMDB (optimistic integration) and existing environments (existing configuration discovery) to help solve complex, real-world problems: incident management, CMDB population, disaster recovery/business continuity
- Reduce cost, speed implementation and improve learning curve through iteration and reuse
- Enable standards-based automation using meta-mapped aggregated relationships as exchange between disparate management systems
- Further the vision of utility and grid computing
Next Steps

- DCML Member Section currently testing and validating the usage of semantics and vocabularies
- Evaluating different approaches to describe the interchange relationships between the dimensions of IT Services management
- Validating usage of DCML within installed base through end user reference implementations
- Continuing to work with like-minded standards bodies
QUESTIONS AND ANSWERS

Adobe
©at 7.0 Docu