OOPSLA Workshop
“Designing for the Cloud”

Stuart Charlton, CTO
October, 2009
## The World of Design & Operations

<table>
<thead>
<tr>
<th>Category</th>
<th>HP</th>
<th>IBM</th>
<th>Microsoft</th>
<th>Oracle</th>
<th>BMC</th>
<th>VMware</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Architecture, Development</strong></td>
<td><strong>Rational</strong></td>
<td><strong>Visual Studio</strong></td>
<td><strong>Jdeveloper</strong></td>
<td></td>
<td></td>
<td><strong>Spring</strong></td>
</tr>
<tr>
<td><strong>Application Lifecycle Management</strong></td>
<td><strong>Mercury</strong></td>
<td><strong>Rational</strong></td>
<td><strong>Visual Studio Team System</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>IT Automation</strong></td>
<td><strong>OO</strong></td>
<td><strong>TPM</strong></td>
<td><strong>Systems Center CM</strong></td>
<td><strong>Oracle EM</strong></td>
<td><strong>Atrium</strong></td>
<td><strong>vCenter, vApps</strong></td>
</tr>
<tr>
<td><strong>Operations and System Management</strong></td>
<td><strong>SA, NA, CMDB</strong></td>
<td><strong>Tivoli</strong></td>
<td><strong>Systems Center</strong></td>
<td><strong>Oracle EM</strong></td>
<td><strong>Patrol Remedy Blade Logic</strong></td>
<td><strong>vCenter</strong></td>
</tr>
<tr>
<td><strong>Virtualization, Operating Systems</strong></td>
<td><strong>HPUX NonStop</strong>, <strong>z/OS, LPAR AIX</strong></td>
<td><strong>Windows, Azure, Hyper-V</strong></td>
<td><strong>Oracle VM, Linux, Solaris</strong></td>
<td></td>
<td></td>
<td><strong>vSphere</strong></td>
</tr>
<tr>
<td><strong>Hardware, Network, Storage</strong></td>
<td><strong>Proliant Integrity ProCurve</strong>, <strong>System x, 1, p, z</strong></td>
<td><strong>Sun</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# A Major Cultural Split

<table>
<thead>
<tr>
<th></th>
<th>HP</th>
<th>IBM</th>
<th>MSFT</th>
<th>ORCL</th>
<th>BMC</th>
<th>VMW</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Architecture, Development</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Application Lifecycle Management</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>IT Automation</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Operations and System Management</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Virtualization, Operating Systems</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Hardware, Network, Storage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Delivery Orientation**
- **Culture and Tool Gaps!**
- **Operations Orientation**
The Realities

- Organizationally & Geographically Distributed Design and Operations (The Cloud)

- Performance, Scale, and Availability are due to a **complex combination** of design and operational decisions

- Application and infrastructure management is **complex** and **inter-disciplinary**
Suggested Design Goals for Cloud Computing

• Separate Applications from Infrastructure
  » How far can Black-Box PaaS really go?

• Enabling Computer-Assisted Design and Operations
  » IT complexity is getting overwhelming
  » Can machine reasoning and planning help?

• Explicit Collaboration
  » Both design and operations suggest highly collaborative work
  » Not traditionally supported by most tooling
Characterizing an Integrated Approach to Integrated Cloud App Design & Operations

• Distributed, Autonomous Control
  » Ownership & stewardship of artifacts and systems are normally decentralized

• Open Document-Exchange
  » The trouble with APIs
  » Today’s attempts: model marts, CMDBs
  » Contrast to the success of the Web

• Hyperlinked Web Architecture
  » No monolithic documents
Characterizing an Integrated Approach to Integrated Cloud App Design & Operations

• Model-Driven
  » Make documents conform to a logical framework and visual notation

• Goal and Policy Driven
  » “What, not How”: Declarative specifications
  » Allow for automated planning of operational steps

• Viewpoint-Based
  » Extensible modeling languages & constraints
Characterizing an Integrated Approach to Integrated Cloud App Design & Operations

• Collaborative
  » Leveraging social computing
  » Faster decision making to enact changes to a system

• Governable
  » Access control & entitlement enforcement
End-to-End Collaboration & Change Management

Model-Driven Collaborative Application Design

Enterprise Architects
- Service Designs

Dev & QA
- Software Settings & Configuration

Application Architects
- System Designs

IT Operations
- Resource Models

IT Management
- Auditing, Metering and Planning

System Admins
- Change & Configuration Management

Automated Planning

Provisioning & Configuration

Test System
Staging System
Production System

Private & Public Virtual Infrastructure
Vision: A Distributed Model-Driven Cloud

Application Infrastructure: Databases, BPM, Integration, App Servers, Web Servers

Business Services Models Processes System Architecture

Operations and Automation
Scale In/Out Recover Deploy Configure Change Migrate

Software Configuration Host Storage Network

Management Plane

Cloud Control Plane

Interoperability via Open Web Protocols

Virtual Resources: Multi-Cloud, Multi-Organization, Geographically Distributed
A Foundation for Interoperability

The Elastic Modeling Languages

**EMML**
- Deployment State
- Historical State
- Versions
- Change Lists

**ECML**
- Extensible Views & Policies
- Resource Allocation View
- Structure View (Components & Connectors)
- Lifecycle
- Dependencies

**EDML**
- Categories & Capabilities
- Resources
- Configuration
Modeling Applications, not Appliances

Example System Design with ECML

- **Application Server**
  - Component
  - Lifecycle
  - Settings
  - Requirements

- **Database Connection Pool**
  - Connector
  - Lifecycle
  - Settings
  - Requirements

- **Relational Database**
  - Component
  - Lifecycle
  - Settings
  - Requirements