OPEN SOURCE CLOUD-BASED ENTERPRISE INTEGRATION

Tony Shan

May 16, 2012
# Table of Contents

- Concept
- Purpose
- Value Prop
- Pros and Cons
- Core Functions
- Usage Scenarios
- Integration Approaches
- Integration Solutions
- Market Landscape: 5-tier
- Open Source Solutions for Integration Functions
- Key Industry Leaders
- Challenges
- Trending
- Best Practices
- Case Study
What is Enterprise Integration?

- The use of software or hardware to integrate a set of internal and external enterprise applications and services by applying the architectural principles and patterns for computer systems.
Purpose

Data consistence
• Ensures that information in multiple systems is kept consistent with data integrity.

Service interactions
• Enable inter-system or inter-application communications

Product independence
• Extracts business policies or rules from applications and implements them in the integration system, so that even if one of the business applications is replaced with a different vendor's application, the business rules do not have to be re-implemented.

Common facade
• An integration system can front-end a cluster of applications, providing a single consistent access interface to these applications and shielding users from having to learn to use different software packages.
Value Prop

Simplify and automate business processes

Avoid having to make sweeping changes to the existing applications or data structures.

Unrestricted sharing of data and business processes among any connected application or data sources in the enterprise.

Reduce the inter-dependency and tight coupling among applications
## Pros and Cons

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Real time information access among systems</td>
<td>• High initial development costs, especially for small and mid-sized businesses (SMBs)</td>
</tr>
<tr>
<td>• Streamlines business processes and helps improve organizational efficiency</td>
<td>• Require a fair amount of up-front business design, which many managers are not able to envision or not willing to invest in.</td>
</tr>
<tr>
<td>• Maintains information integrity across multiple applications</td>
<td>• Fairly steep learning curve for inexperienced teams to get on board</td>
</tr>
<tr>
<td>• Ease of development and maintenance</td>
<td></td>
</tr>
</tbody>
</table>
## Core Functions

<table>
<thead>
<tr>
<th>Core Functions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connection</strong></td>
<td>Messaging formats, communications protocols, connectivity alternatives</td>
</tr>
<tr>
<td><strong>Mediation</strong></td>
<td>Dynamic provisioning, transformation, mapping, transaction management, policy enforcement, registry, SLA</td>
</tr>
<tr>
<td><strong>Orchestration</strong></td>
<td>BPMN, BPEL, long-running process</td>
</tr>
<tr>
<td><strong>Quality of services</strong></td>
<td>Fault tolerance, HA, scalability, throughput, topology, and extensibility</td>
</tr>
<tr>
<td><strong>Change and control</strong></td>
<td>Lifecycle management, monitoring, design tools, security</td>
</tr>
</tbody>
</table>
Integration Approach

- Vertical
- Star
- Horizontal
- On-demand
# Comparison of Integration Approaches

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Strength</th>
<th>Weakness</th>
<th>Implementation</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Vertical</strong></td>
<td>Integrate subsystems</td>
<td>Fast integration -&gt; cheaper cost in short term</td>
<td>Poor maintainability</td>
<td>OOP, RMI</td>
<td>Silos inhibiting reuse</td>
</tr>
<tr>
<td><strong>Star</strong></td>
<td>interconnecting</td>
<td>Flexibility of reuse of functionality</td>
<td>Rising cost for heterogeneous platforms and proprietary interfaces</td>
<td>MQ, MB, Pub/Sub, WS SOAP/REST, JSON</td>
<td>Point to point connectivity</td>
</tr>
<tr>
<td><strong>Horizontal</strong></td>
<td>Specialized engine</td>
<td>Significantly cut the number of interfaces</td>
<td>Overheads and platform dependency</td>
<td>ESB, integration engine, EII</td>
<td>Loosely coupled</td>
</tr>
<tr>
<td><strong>On-demand</strong></td>
<td>Dynamically bound to available services in the pool or discovered on the fly</td>
<td>Choose best-of-breed services with best cost-effectiveness</td>
<td>Unreliable SLA and security concerns</td>
<td>Integration Platform as a Service (IPaaS)</td>
<td>Emerging and maturing</td>
</tr>
</tbody>
</table>
## Integration Solutions

<table>
<thead>
<tr>
<th>Patterns</th>
<th>Topology</th>
<th>Techniques</th>
<th>Communication Styles</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Mediation</td>
<td>• 1 to 1</td>
<td>• Pipe</td>
<td>• Central broker</td>
</tr>
<tr>
<td>(intra-communication)</td>
<td>• Hub and spoke</td>
<td>• Adapter</td>
<td>• Canonical data model</td>
</tr>
<tr>
<td>• Federation</td>
<td>• Bus</td>
<td>• Data format transformation</td>
<td>• Connector</td>
</tr>
<tr>
<td>(inter-communication)</td>
<td></td>
<td>• Pub/Sub</td>
<td>• API</td>
</tr>
<tr>
<td>• Access (sync vs async)</td>
<td></td>
<td>• Transaction (1-phase and 2-phase commit)</td>
<td>• Driver</td>
</tr>
<tr>
<td>• Duration (short-lived vs long-running)</td>
<td></td>
<td>• Share-nothing</td>
<td>• Shared memory</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Copyright © 2012. Contact TonyShan@live.com

Reproduction, redistribution or disclosure is prohibited without prior written consent from the author.
Market Landscape: 5-Tier

**Commercial Enterprise-grade**
- Software AG
- Tibco
- Oracle
- IBM

**Vendor-backed Open Source**
- Red Hat JBoss
- Mulesoft
- WSO2
- FuseSource

**Open Source Community**
- Camel
- ServiceMix
- ActiveMQ
- Spring Integration

**Cloud-based Middleware**
- Amazon SQS
- Azure message queue
- OpenShift
- Cloud Foundry

**Cloud-oriented for SaaS/PaaS**
- Cast Iron
- Boomi
- Fiorano
- Mulesoft
Open Source Solutions for Integration Functions

- ActiveMQ
- Camel
- CFX
- ServiceMix
- Karaf
- ODE
- JQuery
# Key Industry Leaders

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Solution</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software AG</td>
<td>webMethods product has long history. Bundle with CentraSite registry/repository</td>
<td>Largest number of ESB installations. Integrate with ARIS BPM.</td>
</tr>
<tr>
<td>Tibco</td>
<td>ActiveMatrix widely used – over 3000 installations</td>
<td>Integrate with BusinessWorks</td>
</tr>
<tr>
<td>Oracle</td>
<td>Rooted from BEA AquaLogic product</td>
<td>More than 2600 implementations</td>
</tr>
<tr>
<td>Progress</td>
<td>Based on SonicMQ and Sonic ESB</td>
<td>Integrate with CIS and BPM solutions</td>
</tr>
<tr>
<td>FuseSource</td>
<td>Spin-out from Progress, based on Iona</td>
<td>Combine Apache integration solutions</td>
</tr>
<tr>
<td>WSO2</td>
<td>Lean middleware</td>
<td>Used in eBay for 1B transactions per day</td>
</tr>
<tr>
<td>IBM</td>
<td>Websphere Message Broker, ESB, ESB Registry, Datapower Appliance</td>
<td>3 ESB products – 1 from Datapower and 2 from traditional middleware</td>
</tr>
<tr>
<td>MuleSoft</td>
<td>2500 deployments in production</td>
<td>Tcat for Enterprise Tomcat</td>
</tr>
<tr>
<td>Red Hat</td>
<td>JBoss ESB and SOA Platform</td>
<td>Data virtualization – aggregate data sources into a single view</td>
</tr>
</tbody>
</table>
Challenges

- Standardization
- Heterogeneous platforms
- Orchestration and choreography
- Collaborative computing
- Scalability
- Interoperability
- Federation
- Security
- Massive data
- Soft lock-in

Copyright © 2012. Contact TonyShan@live.com

Reproduction, redistribution or disclosure is prohibited without prior written consent from the author.
Trending

- Federated ESB to link with cloud applications
  - e.g. Mule ESB Cloud Connector to eCommerce Platform Magento

- Cloud middleware platform with federated SSO
  - e.g. WSO2 Stratos

- Integration Platform as a Service (IPaaS)
  - e.g. Mule iON for cloud-to-cloud and cloud-to-premise integration solutions

- Cloud-bound ID management patterns (trusted, external, interoperable)
  - e.g. Ping Identity

- High-level platform-independent integration design representation emerges
  - e.g. Guarana Domain-Specific Language (DSL)

- ESB as a Service is still in the infancy stage
  - e.g. Fiorano hybrid Cloud Platform for SaaS and PaaS with on-premise in P2P mode

Copyright © 2012. Contact TonyShan@live.com

Reproduction, redistribution or disclosure is prohibited without prior written consent from the author.
Best Practices

1. Identify commonalities of business and technology integration needs
2. Decouple application linkage to reduce mutual awareness
3. Consider asynchronous communications (MOM and pub/sub)
4. Leverage integration patterns and styles
5. Start with a lower-cost and out-of-box integration solution
6. Explore the potential of open source ESB
7. Upgrade to a more robust solution as needs grow
8. Look into cloud alternatives
9. Drive two-prong cloud integration strategies
10. Take a hybrid deployment approach
11. Treat security as a cross-cutting concern
12. Scale end-to-end
Case Study: Integration for Big Data Ingestion

- Small batch
- Near real-time
- Streaming
- Javabeans, ActiveX, .Net
- Open source
- SDK

- Staging
- Queue
- Management
- Secure comm
- ESB
- ServiceMix/FuseSource

- Monitoring
- Security
- Error handling
- Notification
- Talend
- Hyperic HQ
Summary

- **Foundation**
  - Concept
  - Purpose
  - Value Prop
  - Pros and Cons
  - Core Functions

- **Solutioning**
  - Usage Scenarios
  - Integration Approaches
  - Integration Building Blocks

- **Platform Selection**
  - Market Landscape: 5-tier
  - Open Source Solutions for Integration Functions
  - Key Industry Leaders

- **Direction and Outlook**
  - Challenges
  - Trending
  - Best Practices

- **Case Study**
  - Real-world example
  - Lessons learned
ACCORDING TO THIS, THE PLANET EARTH WAS ONCE POPULATED BY HUMANS, THEN IN 2012...

...THEY ALL MOVED TO THE CLOUD.