Towards
SaaS (Software as a Service)
Evaluation Model

Paul Sorenson & Xian Chen
Department of Computing Science
Agenda

- IT Business Services Models
- SaaS - what is it & how is it different?
- Quality-based Model
- Future Directions and Summary
An Important Definition (thanks to IBM sponsored white paper)*

A **service system** is a dynamic configuration of resources (people, technology, organizations, and shared information) that creates and delivers value between the provider and customer through service.

*“Succeeding through service innovation”, Cambridge Service Science, July 2008*
One more definition (thanks to Wikipedia)

- **SaaS (Software-as-a-Service)** is a model of software deployment where an application is hosted as a service provided to customers across the Internet.

- By eliminating the need to install and run the application on the customer's own computer, SaaS alleviates the customer's burden of software maintenance, ongoing operation, and support. Conversely, customers relinquish control over software versions or changing requirements; moreover, costs to use the service become a continuous expense, rather than a single expense at time of purchase.
**IT Business Service (Conceptual) Models**

- **In-house services** - traditional, very challenging and hard to justify

- **Business framework** - growing in popularity, challenging, change organization to fit business model, not always easily delivered or accepted (e.g., ERP’s and IBM Common Business Model)

- **Outsource services** - strategic, promising, brings a new set of problems (dealing with the service provider through contracts)

- **SaaS approach** - strategic, deployment of “best practices” in a business area, limited integration and recognition of special requirements of an organization being serviced
Existing SaaS Maturity Models

- Saas Simple Maturity Model by Microsoft Corporation (2006)
- Model by Forrester Research (2008)
- From SaaS Maturity Model to SaaS Evaluation Model
SaaS Simple Maturity Model (Microsoft, 2006)

- Model on Single Packaged Application
- Focused on SaaS Application Architecture
- Three Key Attributes of an Architecture:
  - **Configurability**: Metadata used to configure the way the application behaves for customers
  - **Multi-tenant Efficiency**: Maximizing the sharing of resources across tenants
  - **Scalability**: Maximizing concurrency, resource efficiency
SaaS Simple Maturity Model: Four Levels

- **Level 1: Ad Hoc/Custom**
- **Level 2: Configurable**
- **Level 3: Configurable & Multi-Tenant-Efficient**
- **Level 4: Scalable, Configurable, Multi-Tenant-Efficient**
## Summary

<table>
<thead>
<tr>
<th>Maturity Level</th>
<th>Instances</th>
<th>Configurability</th>
<th>Multi-tenant Efficiency</th>
<th>Scalability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>Multiple different instances - ASP Model</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 2</td>
<td>Multiple identical instances - code sharing</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 3</td>
<td>Single instance - configurable metadata</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Level 4</td>
<td>Multiple identical instances with tenant load balancer</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
SaaS Maturity Model (Forrester, 2008)

- Evolution Model on SaaS Applications: guidance on realistic strategy transformation for software vendors and services providers considering an SaaS business model
- Focused on SaaS Application Domain
SaaS Maturity Model: Six Levels

1. Manual ASP
   Provide similar apps to many clients.

2. Industrial ASP
   Provide configured apps to many clients.

3. Single-app SaaS
   Provide one packaged business app out of one multi-tenant app to many clients.

4. Business-domain SaaS
   Provide tenant-specific configuration of multiple packaged apps and custom extensions on a multipurpose, multitenant platform.

5. Dynamic Business Apps-as-a-service
   Compose user-specific apps based on packaged and custom business apps. Dynamic orchestration and provisioning in a multitenant environment.

Outsourcing
Delegate operations of existing app.

Time
Level 0: Outsourcing

- A service provider operates a major application or a unique application landscape for a large enterprise customer
- The outsourcing company can't leverage the application for a second customer
- Does not really qualify as SaaS
Level 1: Manual ASP

- A service provider runs packaged applications for multiple midsize enterprises
- Each client usually has a dedicated server running its instance of the application and is able to customize the installation in the same way as self-hosted applications
- Target midsize companies
- Similar to level 1 (Ad-hoc/custom) of Microsoft’s model
Level 2: Industrial ASP

- A service provider runs identical packaged applications with customer-specific configurations to many customers
- Cut the operating costs of applications to a minimum
- Usually applicable to small and midsize business customers
- Similar to level 2 (Configurable) of Microsoft’s model
Level 3: Single-app SaaS

- An SaaS provider provides one packaged business application with Web-based user interface to many customers (e.g. Salesforce’s initial CRM application)
- Customization is restricted to configuration
- Focus on small and midsize business customers
- Similar to level 3 (Configurable & Multi-Tenant-Efficient) of Microsoft’s model
Level 4: Business-domain SaaS

• An advanced SaaS vendor provides not only a well-defined business application but also a platform for additional business logic

• Single application of level 3 complemented with third-party packaged SaaS solutions and custom extensions

• Satisfy the requirements of large enterprises

• Similar to level 4 of Microsoft’s model, yet with extension of composition with other applications
Level 5: Dynamic Business-domain Apps-as-a-Service

- An advanced SaaS vendor coming provides a comprehensive application and integration platform on demand, which they will pre-populate with business applications or services.

- New paradigm: design for people, build for change.

- Composition of user-specific business applications on various levels in a multi-tenant environment.

- The resulting process agility will attract everyone, including large enterprise customers.
### Summary

<table>
<thead>
<tr>
<th>Maturity Level</th>
<th>Definition</th>
<th>Single/Multiple Application</th>
<th>Configurability</th>
<th>Multi-tenant Efficiency</th>
<th>Scalability</th>
<th>Equivalent Level in MS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 0</td>
<td>Outsourcing</td>
<td>Single app to one client</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 1</td>
<td>Manual ASP</td>
<td>Similar apps to multiple clients</td>
<td></td>
<td></td>
<td></td>
<td>Level 1</td>
</tr>
<tr>
<td>Level 2</td>
<td>Industrial ASP</td>
<td>Configured apps to many clients</td>
<td>X</td>
<td></td>
<td></td>
<td>Level 2</td>
</tr>
<tr>
<td>Level 3</td>
<td>Single-app SaaS</td>
<td>One packaged app to many clients</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Level 3</td>
</tr>
<tr>
<td>Level 4</td>
<td>Business-domain SaaS</td>
<td>Configuration of multiple packaged apps and custom extension</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Level 4 + custom extension</td>
</tr>
<tr>
<td>Level 5</td>
<td>Dynamic Business Apps-as-a-Service</td>
<td>Dynamic composition of user-specific apps based on packaged and custom business apps</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>Level 4 + dynamic composition</td>
</tr>
</tbody>
</table>
Both perspectives of service provider and service customer are needed in SaaS maturity model (not just the service provider)

Also need a focus on the business relationship between provider and customer
SaaS Relationships - Provider View

Conformance Quality

Marketing

Product

Gap Quality

Clients

Customer Organization

Provider Organization
SaaS Relationships - **Customer View**

- **Conformance Quality**
- **Value Quality**
  - **Business Units**
  - **Users**
  - **Functiona l“Quality”**

- **Marketin g**
- **Producti on**
- **Gap Quality**

- **SaaS Relationships**
- **Customer View**
- **Provider Organization**
- **Gap Quality**
- **Value Quality**
- **Business Units**
- **Users**
- **Functional “Quality”**
Relationship Management and Improvement

1. **Internal Provider Organization Relationship** - (Conformance Quality):
   Marketing Maturity Models (see refs) dealing with business relationships between marketing and production in a producer organization to meet or succeed market performance expectations

2. **Provider to Customer Relationship** - (Gap Quality): SERVQUAL/Gap Models used to determine and assess customer QoS requirements as a basis for service improvement

3. **Internal Customer Organization Relationship** - (Functional “Quality”):
   Functional Roadmaps, Business Modeling tools, frameworks and approaches

4. **Customer to Provider Relationship** - (Value Quality) ROI, Business case analysis, risk-based analysis
# Maturity Levels of Business Relationship in SaaS

<table>
<thead>
<tr>
<th>Maturity Level</th>
<th>Business Relationship</th>
<th>Service Customer</th>
<th>Service Provider</th>
<th>Quality Approaches</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>Ad-hoc transaction</td>
<td>Functionality achieved</td>
<td>Static service delivery</td>
<td></td>
</tr>
<tr>
<td>Level 2</td>
<td>Repeatable transaction</td>
<td>Reliability and other quality requirements guaranteed</td>
<td>Service delivered with stable capability</td>
<td>QoS requirements/basic SLAs</td>
</tr>
<tr>
<td>Level 3</td>
<td>Configurable transaction</td>
<td>Monitoring on service quality</td>
<td>Service delivered with configurable capability</td>
<td>SLAs, Survey</td>
</tr>
<tr>
<td>Level 4</td>
<td>Long-term relationship</td>
<td>Continuous evaluation on service quality</td>
<td>Integrated delivery with customer extension</td>
<td>Survey, Balance Scorecard</td>
</tr>
<tr>
<td>Level 5</td>
<td>Strategic partnership</td>
<td>Governance of service (e.g. risk prevention)</td>
<td>Dynamic delivery with change management</td>
<td>ROI, Value analysis</td>
</tr>
</tbody>
</table>
Questions?!?

- Discuss the factors (forces at play) that will shape SaaS solutions and markets in the future? (hints: What are SaaS shortcomings? What about IT infrastructure?)

- What, if any, is the role of business models in SaaS deployment?

- Is a SaaS Maturity Model useful?
## IBM’s Component Business Model

### Business competencies

<table>
<thead>
<tr>
<th>Customers</th>
<th>Products/ services</th>
<th>Channels</th>
<th>Logistics</th>
<th>Business administration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Market strategy</td>
<td>Merchandise planning</td>
<td>Channel strategy</td>
<td>Network design</td>
<td>Corporate strategy</td>
</tr>
<tr>
<td>Customer service strategy</td>
<td>Assortment planning</td>
<td>Store design</td>
<td>Corporate planning</td>
<td>Corporate governance</td>
</tr>
<tr>
<td>Marketing strategy</td>
<td>Space planning</td>
<td>Real estate strategy</td>
<td>Warehouse design</td>
<td>Financial planning</td>
</tr>
<tr>
<td></td>
<td>Promotion planning</td>
<td>Internet design</td>
<td>Demand/flow planning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Product development</td>
<td>Catalog/call center</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sourcing</td>
<td>design</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Accountability level

#### Direct

- Campaign management
  - Product flow
  - Planogramming
  - Allocation
  - Inventory mgt/OTB
- Service management
  - Demand forecasting
  - Price management
  - Content management
  - Vendor management
- Customer service
  - Item management
  - Product management
  - PO management
  - Vendor management
  - Replenishment
  - Revenue/clearance management

#### Control

- Channel management
  - Channel management
  - Labor management
  - Order management
- Real estate, construction and facilities management
- Loss prevention

#### Execute

- Carrier management
  - Inbound/routing
  - Receipt scheduling
  - Delivery scheduling
- Business performance management
  - Business performance management
  - Treasury and risk management
- Legal and regulatory compliance
- Inventory control
- Cash and banking

---

Source: IBM Business Consulting Services.
References


Thank you!