Using integration maturity as a guideline when deciding to implement SOA
Key objectives

- The logistics integration environment
  - Provide an overview of the integration environment and high-level architectural landscape of our logistics client to illustrate the complexity of the integration space and the obstacles encountered.

- Rating the integration maturity
  - Provide some guidelines on how to rate the integration maturity of an organization

- TOGAF and SOA
  - Using the TOGAF methodology and SOA to decide upon an integration approach.
Overview of the client architecture landscape

Our client has an international principle base that it needs to deliver world-class services to and requires a sound information management approach.
Overview of the client architecture landscape

Our client has an international principle base that it needs to deliver world-class services to and requires a sound information management approach.

External Principle Systems

- SAP BC
- MFGPro
- Gentran
- MQ Series

Internal Configuration – South Africa

- Integration Hub
- Webmethods Integration Solution
- PRIDE Warehouse Management System
- WMOS Warehouse Management System
- SAP BC
- MFGPro
- Gentran
- MQ Series
- PRIDE Warehouse Management System
- WMOS Warehouse Management System
Our client has a complex internal systems design that requires not only transactional, but product, financial, logistical and tracking information.
The complex technical integration design that facilitates the integration paths, with hardware and fail-over components.
The complex technical integration design that facilitates the integration paths, with hardware and fail-over components.
The Data Architecture depicts the business information flow relevant to our project within the B2B logistics environment.
Different File Complexities

- Flat file
- GS1 XML
- XML
- AS400
- Delimited
- Custom defined
The Business Order Cycle depicts some of the integration services that we support within the B2B logistics environment.

The order process depicts 8 processes (services) that will be orchestrated to link into the supporting services via the orchestration layer.
This webMethods Service Architecture depicts the orchestration of the services within the integration layer environment.

The outbound message flow diagram depicts the sequence flow (orchestration) of the services that support the client business process in the integration solution.
The following typical questions will be asked to determine the level of integration maturity

<table>
<thead>
<tr>
<th>Business Architecture</th>
<th>Application Architecture</th>
<th>Data Architecture</th>
<th>Technology Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>What requirements have been set by business?</td>
<td>What integration applications exist?</td>
<td>What protocols are proposed for use?</td>
<td>Who is responsible for the communication medium?</td>
</tr>
<tr>
<td>Is there a project plan driven by a PM?</td>
<td>What current integrations are in place and what have been done?</td>
<td>What level of data security is required?</td>
<td>How advanced is the hardware used in the integration?</td>
</tr>
<tr>
<td>Is there involvement by business and how do they measure success?</td>
<td>What requirements are set and how flexible is the integration layer?</td>
<td>What mechanisms are used to validate data sent / received?</td>
<td>Is there consideration towards DR and fail-over?</td>
</tr>
<tr>
<td></td>
<td></td>
<td>What alerting mechanisms are required for message control?</td>
<td>What SLA’s are in place for the physical integration?</td>
</tr>
</tbody>
</table>
## Determining the level of integration maturity of the business layer

<table>
<thead>
<tr>
<th>Level 1: Ad-hoc</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
<th>Level 5: Optimised</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chaotic</td>
<td>Need-based</td>
<td>Structured</td>
<td>Managed</td>
<td>Seamless</td>
</tr>
</tbody>
</table>

- **Level 1: Ad-hoc**
  - Undocumented processes
  - Manual processes
  - Limited standardisation, re-use and predictability of processes
  - Individual dependant processes
  - Non-standard Business Architecture

- **Level 2: Foundational**
  - Automatic processes
  - Inconsistent processes
  - Basic change/configuration management
  - Quality assurance
  - Organisational definition
  - Project planning & tracking
  - Estimating

- **Level 3: Disciplined**
  - Business process orientation
  - Best practises, process standardisation & documented
  - Business process modeling
  - Model managed and stored in a central repository
  - Architectural Governance
  - Advanced change/configuration management

- **Level 4: Managed**
  - High degree of business process re-use
  - Operational excellence
  - Operational cost tracking
  - Predictable business results
  - Performance Orientation

- **Level 5: Optimised**
  - Seamless business process integration
  - Environmental risk management

Source: Global Integration Summit - Integration Maturity Model
### Determining the level of integration maturity of the application layer

<table>
<thead>
<tr>
<th>Level 1: Ad-hoc</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
<th>Level 5: Optimised</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chaotic</td>
<td>Need-based</td>
<td>Structured</td>
<td>Managed</td>
<td>Seamless</td>
</tr>
</tbody>
</table>

- **Level 1: Ad-hoc**
  - Chaotic
  - Undocumented applications
  - Unpredictable systems behaviour
  - Stove-pipe custom interfaces
  - Non-standard Applications Architecture
  - Limited application re-use

- **Level 2: Foundational**
  - Need-based
  - Point to point interfaces
  - Application embedded business rules
  - Loosely coupled
  - Error handling and recovery
  - Basic integration testing

- **Level 3: Disciplined**
  - Structured
  - Architectural Governance
  - Hub-spoke, rules engine
  - Training & vendor management
  - Business objects design standards
  - Error handling and recovery
  - Systems monitoring and alerting
  - Managed Integration testing

- **Level 4: Managed**
  - Managed
  - High degree of application re-use
  -Applications built to integrate
  - Predictable results
  - Standards-based integrations
  - Integration Reuse
  - Managed Integration testing
  - Standards-based integrations
  - Integration Reuse

- **Level 5: Optimised**
  - Seamless
  - SOA
  - Use of BPM & BAM tools
  - Repository of business-level services
  - Integration maintenance tracking

Source: Global Integration Summit - Integration Maturity Model
### Determining the level of integration maturity of the data layer

<table>
<thead>
<tr>
<th>Level 1: Ad-hoc</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
<th>Level 5: Optimised</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Chaotic</strong></td>
<td><strong>Need-based</strong></td>
<td><strong>Structured</strong></td>
<td><strong>Managed</strong></td>
<td><strong>Seamless</strong></td>
</tr>
<tr>
<td>- Data structures undocumented</td>
<td>- Data automation</td>
<td>- Data model managed and stored in a central repository</td>
<td>- Metadata based strategy</td>
<td>- Seamless integration</td>
</tr>
<tr>
<td>- Data behaviour unpredictable</td>
<td></td>
<td>- Architectural Governance</td>
<td>- Repository management</td>
<td></td>
</tr>
<tr>
<td>- Manual data</td>
<td></td>
<td></td>
<td>- High degree of data re-use</td>
<td></td>
</tr>
<tr>
<td>- Individual dependant</td>
<td></td>
<td></td>
<td>- Integration Competency Centre</td>
<td></td>
</tr>
<tr>
<td>- Non-standard data architecture</td>
<td></td>
<td></td>
<td>- Predictable results</td>
<td></td>
</tr>
<tr>
<td>- Limited re-use</td>
<td></td>
<td></td>
<td>- Information visibility</td>
<td></td>
</tr>
</tbody>
</table>

Source: Global Integration Summit - Integration Maturity Model
### Determining the level of integration maturity of the technology layer

<table>
<thead>
<tr>
<th>Level 1: Ad-hoc</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
<th>Level 5: Optimised</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chaotic</td>
<td>Need-based</td>
<td>Structured</td>
<td>Managed</td>
<td>Seamless</td>
</tr>
</tbody>
</table>

- Undocumented technology
- Unpredictable
- Non-standard Architecture
- Message-orientated middleware
- Defined Technical Architecture
- Security practices
- Defined systems environments
- Managed Technical Architecture
- Architectural Governance
- Repository management
- High degree of reuse
- Integration Competency Centre
- Web-services standards based infrastructure

Source: Global Integration Summit - Integration Maturity Model
Determining the level of integration maturity requires the assessment of several key attributes.

<table>
<thead>
<tr>
<th>Level 1: Ad-hoc</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
<th>Level 5: Optimised</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chaotic</td>
<td>Need-based</td>
<td>Structured</td>
<td>Managed</td>
<td>Seamless</td>
</tr>
</tbody>
</table>

- Undocumented
- Unpredictable
- Manual
- Individual dependant
- Stove-pipe custom interfaces
- Non-standard Architecture
- Limited Reuse
- Point to point interfaces
- Loosely coupled
- Automatic
- Message-orientated middleware
- Application embedded business rules
- Inconsistent business processes
- Best practices, process standardisation & documented
- Business process modeling
- Model managed and stored in a central repository
- Governance
- Hub-spoke, rules engine
- Training & vendor management
- Metadata based strategy
- Repository management
- High degree of reuse
- Integration Competency Centre
- Hub-spoke, rules engine
- Training & vendor management
- Use of BPM & BAM tools
- Repository of business-level services
- Web-services standards based infrastructure
- Seamless integration

**SOA enabled...**

- Simple FTP
- Controlled FTP
- Un-secure web service
- Secure AS2
- Secure web service

Source: Global Integration Summit - Integration Maturity Model
### Example for Categorizing a client

#### Level of Maturity

<table>
<thead>
<tr>
<th>Integration Architecture and Environment - (Application &amp; Data)</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Defined Technical Architecture</td>
<td>• Managed Technical Architecture</td>
<td>• Standards-based integrations</td>
<td></td>
</tr>
<tr>
<td>• Security practices</td>
<td>• Error Handling and Recovery</td>
<td>• Metadata</td>
<td></td>
</tr>
<tr>
<td>• Defined system environments</td>
<td>• Business Object design standards</td>
<td>• Information visibility</td>
<td></td>
</tr>
<tr>
<td>• Error handling and recovery</td>
<td>• Architectural Governance</td>
<td>• High degree of application re-use</td>
<td></td>
</tr>
<tr>
<td>• Point to point interfaces</td>
<td>• Systems monitoring and alerting</td>
<td>• Repository management</td>
<td></td>
</tr>
<tr>
<td>• Error handling and recovery</td>
<td>• Managed Integration testing</td>
<td>• Integration Competency Centre</td>
<td></td>
</tr>
<tr>
<td>• Basic integration testing</td>
<td>• Architectural Governance</td>
<td>• Predictable results</td>
<td></td>
</tr>
<tr>
<td>• Data automation</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ability to implement Business</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Estimating</td>
<td>• Architectural Governance</td>
<td>• Performance Orientation</td>
<td></td>
</tr>
<tr>
<td>• Project planning</td>
<td>• Business Process Orientation</td>
<td>• Integration Reuse</td>
<td></td>
</tr>
<tr>
<td>• Project tracking</td>
<td>• Managed Integration testing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Basic integration testing</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Organisational Business</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Quality assurance</td>
<td>• Training &amp; education</td>
<td>• High degree of business process re-use</td>
<td></td>
</tr>
<tr>
<td>• Organizational definition</td>
<td>• Integration strategy</td>
<td>• Operational excellence</td>
<td></td>
</tr>
<tr>
<td>• Automatic processes</td>
<td>• Vendor management</td>
<td>• Operational cost tracking</td>
<td></td>
</tr>
<tr>
<td>• Inconsistent processes</td>
<td>• Knowledge management</td>
<td>• Predictable business results</td>
<td></td>
</tr>
<tr>
<td>• Quality assurance</td>
<td>• Business process orientation</td>
<td>• Performance Orientation</td>
<td></td>
</tr>
<tr>
<td>• Organizational definition</td>
<td>• Architectural Governance</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Technology / Infra</th>
<th>Level 2: Foundational</th>
<th>Level 3: Disciplined</th>
<th>Level 4: Managed</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Basic change/ configuration management</td>
<td>• Systems monitoring and alerting</td>
<td>• Integration maintenance tracking</td>
<td></td>
</tr>
<tr>
<td>• Environmental risk management</td>
<td>• Operations standards and processes</td>
<td>• Operational cost tracking</td>
<td></td>
</tr>
<tr>
<td>• Message-orientated middleware</td>
<td>• Managed Technical Architecture</td>
<td>• Repository management</td>
<td></td>
</tr>
<tr>
<td>• Defined Technical Architecture</td>
<td>• Architectural Governance</td>
<td>• High degree of reuse</td>
<td></td>
</tr>
<tr>
<td>• Security practices</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Defined systems environments</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Level 1 occurs on an ad-hoc basis**

Source: Global Integration Summit - Integration Maturity Model
Overview of the client integration maturity landscape

An illustration of the different levels of integration maturity in different sectors.
TOGAF Architecture Development Methodology (ADM) provides a valid framework for SOA as a whole.

- Enterprise Architecture is the key to bringing coherence to an organisation and building a Service-Oriented Enterprise.
- Enterprise architecture removes barriers and creates a more open system.
- It is characterised by an extensive reuse of common data and processes, a process-centric approach with supervisory process management, and an assumption of continuous change.

The iterative phases of the Architecture Design Methodology (ADM) provides a solid framework for approaching a Service Oriented Architecture Integration Solution.
Architecture Change Management is reduced because of the agility of SOA

“Modification of the architecture itself is addressed in the Architecture Change Management phase.”

SOA delivers agility. Changes to services and development of new services does not require changes to the architecture, when they are carried out using methods that are part of that architecture.

The dividing line between projects that implement the architecture and projects that change it remains; but SOA can move projects across that line by including methods for change within the architecture.

Reference Architecture: Services orchestration

Source: The Open Group SOA Working Group White Paper - July 2007,
Challenges faced when using an EA approach to integrate

- Key questions that need to be addressed to be able to implement according to the EA approach.