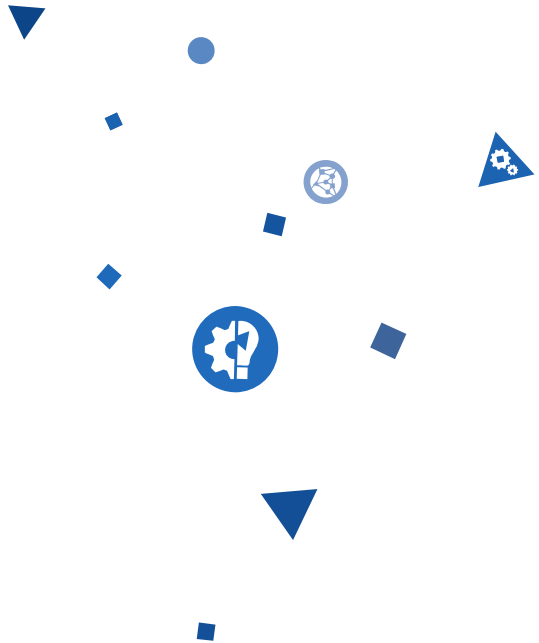
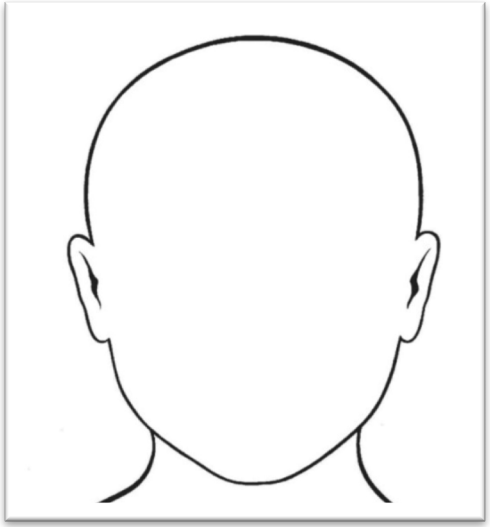


ENTERPRISE APPLICATION INTEGRATION

ARC-013

INTRODUCTION

- Alexander Breyman
- Expert, Luxoft Professional Training Center
- Ph.D. (1998), Associate Professor (2001),
  Software Engineering Department,
  Higher School of Economics



- abreiman@luxoft.com

TRAINING ROADMAP: OVERVIEW

- Introduction
- Overview
- Requirements
- Tools
- Standards
- Patterns

- This training covers major aspects of Enterprise Integration.
- The goal of this training is to learn about Enterprise Application Integration architectural patterns, tools and techniques, and get hands-on experience with Application Integration tool.
- This training is targeted to developers, architects, team leads.

Pre-requisites:

- Enterprise application development experience
- Basic knowledge of the application integration tasks and techniques: Messaging, XML, SOAP

TRAINING ROADMAP: STRUCTURE

- 8 Hour sessions
- 15-30 mins breaks every 1.5 – 2 hours
- Lunches

INTRODUCTION – PRESENT YOURSELF

Name

Company and experience

- Company
- Development experience
- Technologies stack
- Projects

Why you’re here

Your expectations from the training

1:
OVERVIEW

LXFT
LISTED
NYSE

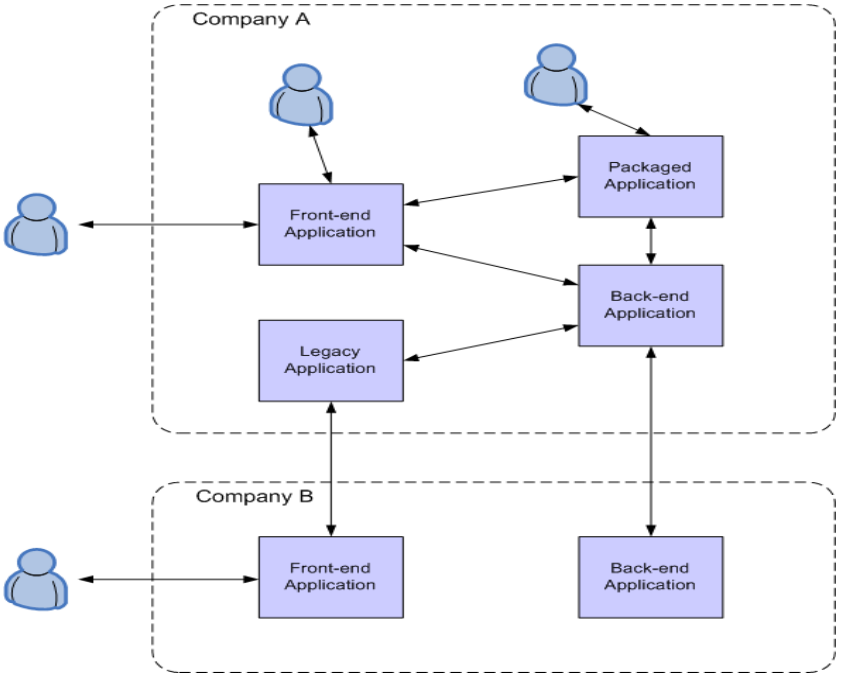
LUXOFT

ENTERPRISE INTEGRATION: DEFINITION

Enterprise integration is a technical field of Enterprise Architecture, which focused on the study of topics such as system interconnection, electronic data interchange, product data exchange and distributed computing environments (source: “Enterprise application integration”, Wikipedia)

ENTERPRISE INTEGRATION 2

Enterprise Integration may involve processes, systems, people and external organizations



ENTERPRISE INTEGRATION TYPES

Enterprise integration areas can be divided on the following types by the layer and involving components:

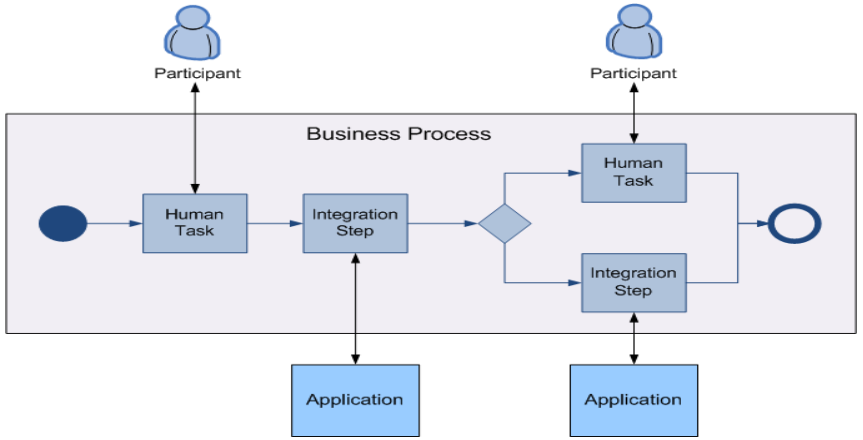
- **Process level integration**. Includes integration of people, IT system to automate fulfillment of the business processes within organization.
- **Presentation level integration**. Integration between user interface components of the separate applications.
- **Application level integration**. Integration between applications which enables the execution of the particular application scenarios.
- **Data level integration**. Integration on a data source level.

PROCESS LEVEL INTEGRATION 1

Business process is a set of related activities and human-assigned tasks which are performed to fulfill specific business function.

Business process activities may implement the following logic:

- Integration with another business processes and applications;
- Performing of human tasks;
- Execution of the business rules;
- Processing of incoming events;
- Scheduling and generation of the events;
- Execution of the custom logic.



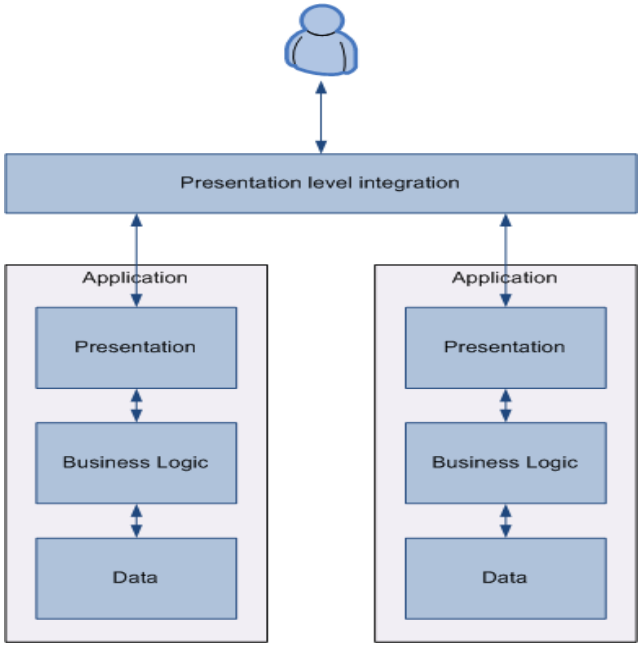
PROCESS LEVEL INTEGRATION 2

Types of business processes (not a complete classification):

- Human-centric business processes. Mainly focused on user to user interaction.
- Integration-centric business processes. These business processes are implemented mostly as a orchestration of the operations of separate enterprise applications.
- Document-centric business processes. This category contains business processes which automate the lifecycle of the documents.

PRESENTATION LEVEL INTEGRATION 1

Presentation level integration uses existing application user interface to implement integration between application.



PRESENTATION LEVEL INTEGRATION 2

Presentation level integration may be useful in the following cases:

- Integration of the legacy or third-party applications which cannot be modified to provide another integration capabilities;
- Presentation capabilities provided by several separate applications should be presented to end-user as one singe application with consistent user interface.

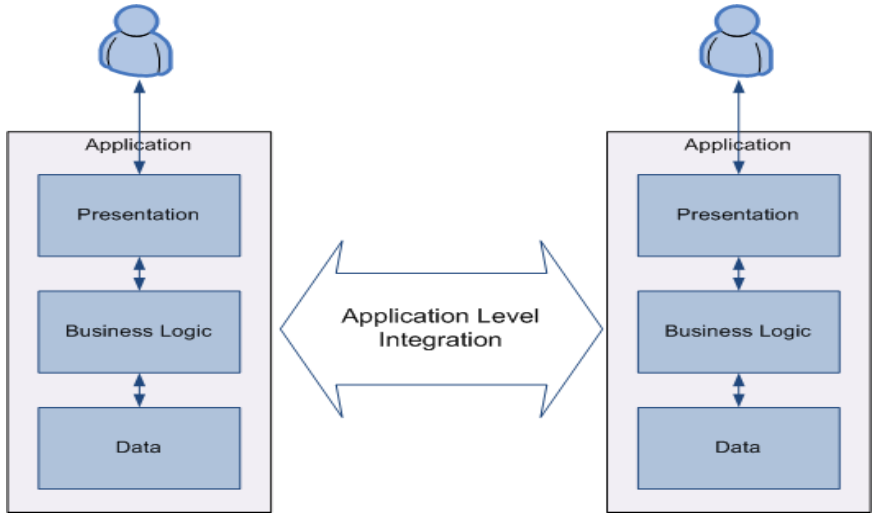
Presentation level integration may be based on the following technologies:

- Terminal emulation, data scraping;
- Usage of special tools for combining application user interfaces into single composite rich client application;
- Integration of web applications into portal or web mashup;

APPLICATION LEVEL INTEGRATION 1

Application level integration is the integration between applications which enables the execution of the particular application scenarios.

Usually this kind of integration is implemented based on integration interface and protocol (API, service interface, etc.).

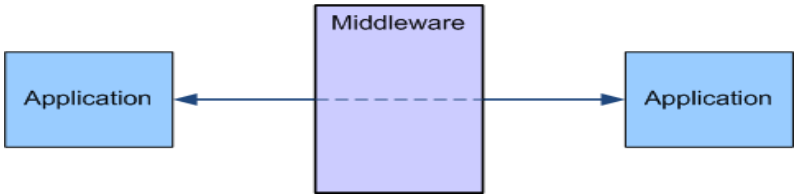


APPLICATION LEVEL INTEGRATION 2

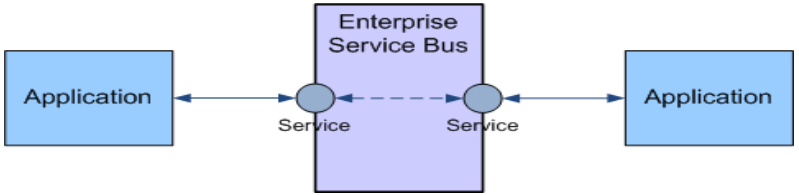
Application level integration implementation types:

- Direct integration

- Middleware integration

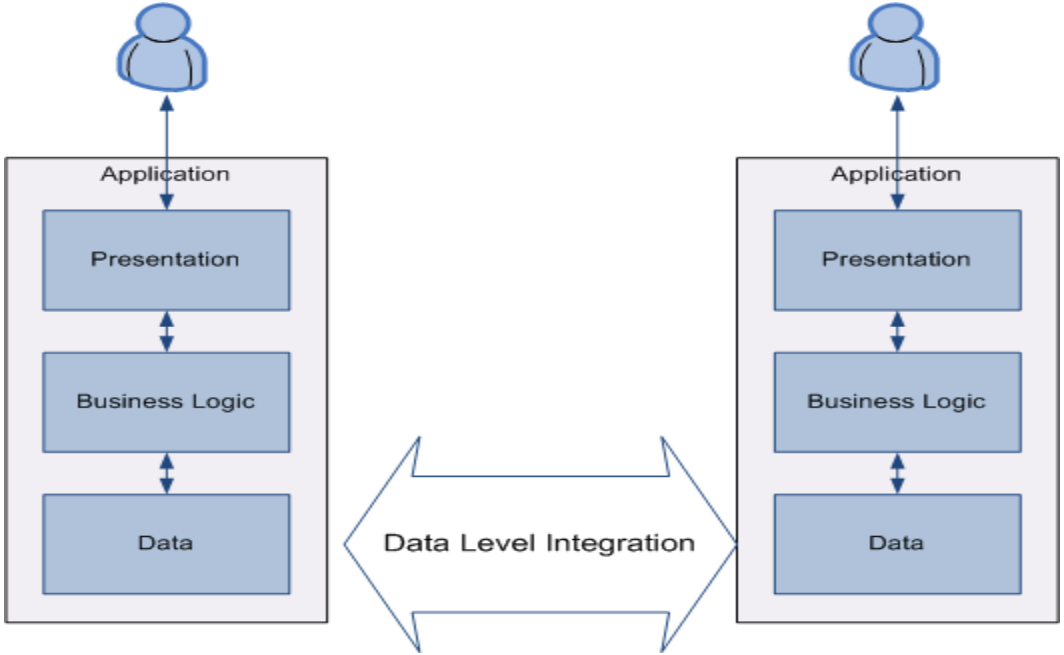


- Service Oriented Integration



DATA LEVEL INTEGRATION 1

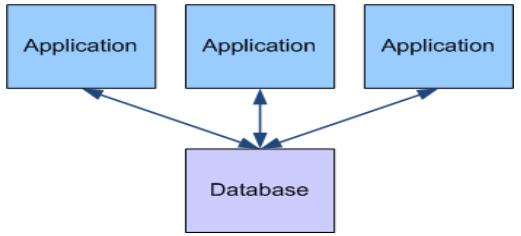
Data Level Integration combines data from different data sources into one view available for business processing.



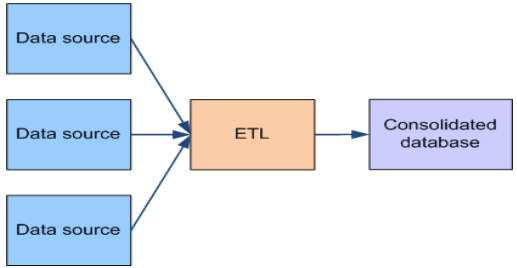
DATA LEVEL INTEGRATION 2

Main types of Data Level Integration:

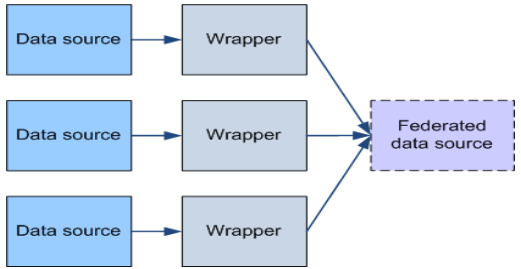
- Shared data source



- Extract, Transform, Load (ETL)



- Federated data source



SERVICE ORIENTED ARCHITECTURE 1

Service-Oriented Architecture (SOA) is an architectural style that supports service-orientation.

Service-orientation is a way of thinking in terms of services and service-based development and the outcomes of services.

Service:

- Is a logical representation of a repeatable business activity that has a specified outcome (e.g., check customer credit, provide weather data, consolidate drilling reports)
- Is self-contained
- May be composed of other services
- Is a “black box” to consumers of the service

Source: “Service Oriented Architecture : What Is SOA?”, The Open Group

SERVICE ORIENTED ARCHITECTURE 2

Principle	Description
Standardized Service Contract	Services adhere to a service-description.
Loose Coupling	Services minimize dependencies on each other
Service Abstraction	Services hide the logic they encapsulate from the outside world
Service Reusability	Logic is divided into services with the intent of maximizing reuse
Service Autonomy	Services should have control over the logic they encapsulate.
Service Statelessness	Ideally, services should be stateless.
Service Discoverability	Services can be discovered (usually in a service registry).
Service Composability	Services break big problems into little problems.

SOA PRINCIPLES TO QUALITY ATTRIBUTES: DISCUSSION

SOA principles:

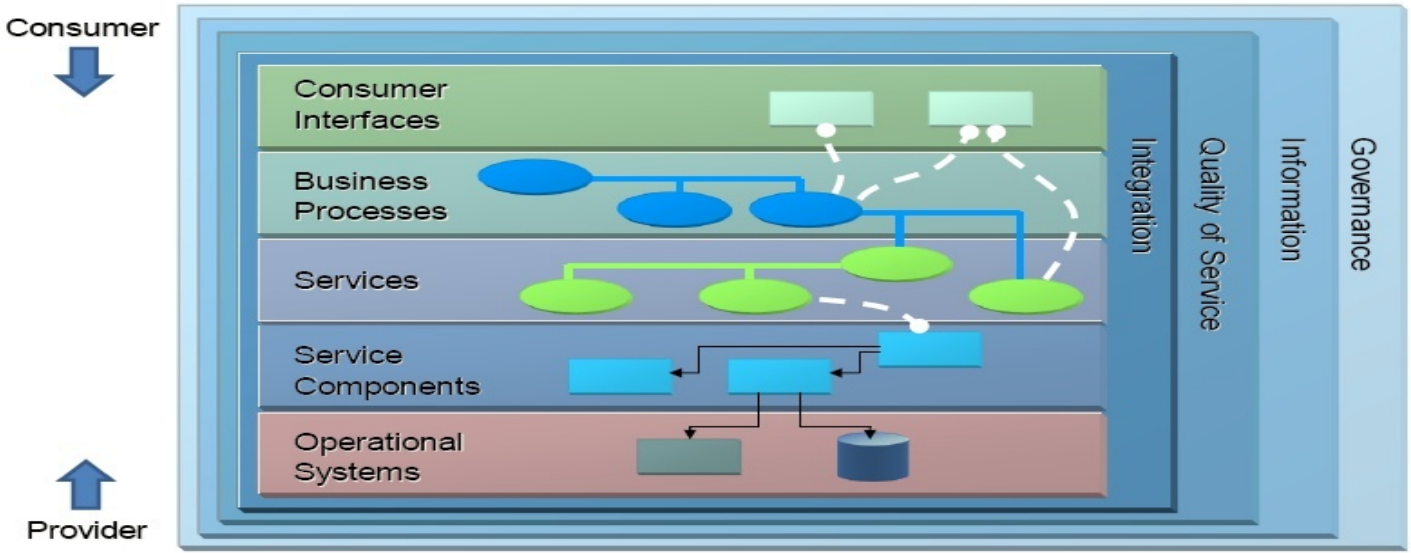
- Standardization
- Loose coupling
- Reusability
- Composability
- Discoverability

Quality attributes:

- Interoperability
- Modifiability/Maintanability
- Performance
- Reusability
- Security
- Testability
- Scalability
- Reliability

SERVICE ORIENTED ARCHITECTURE 3

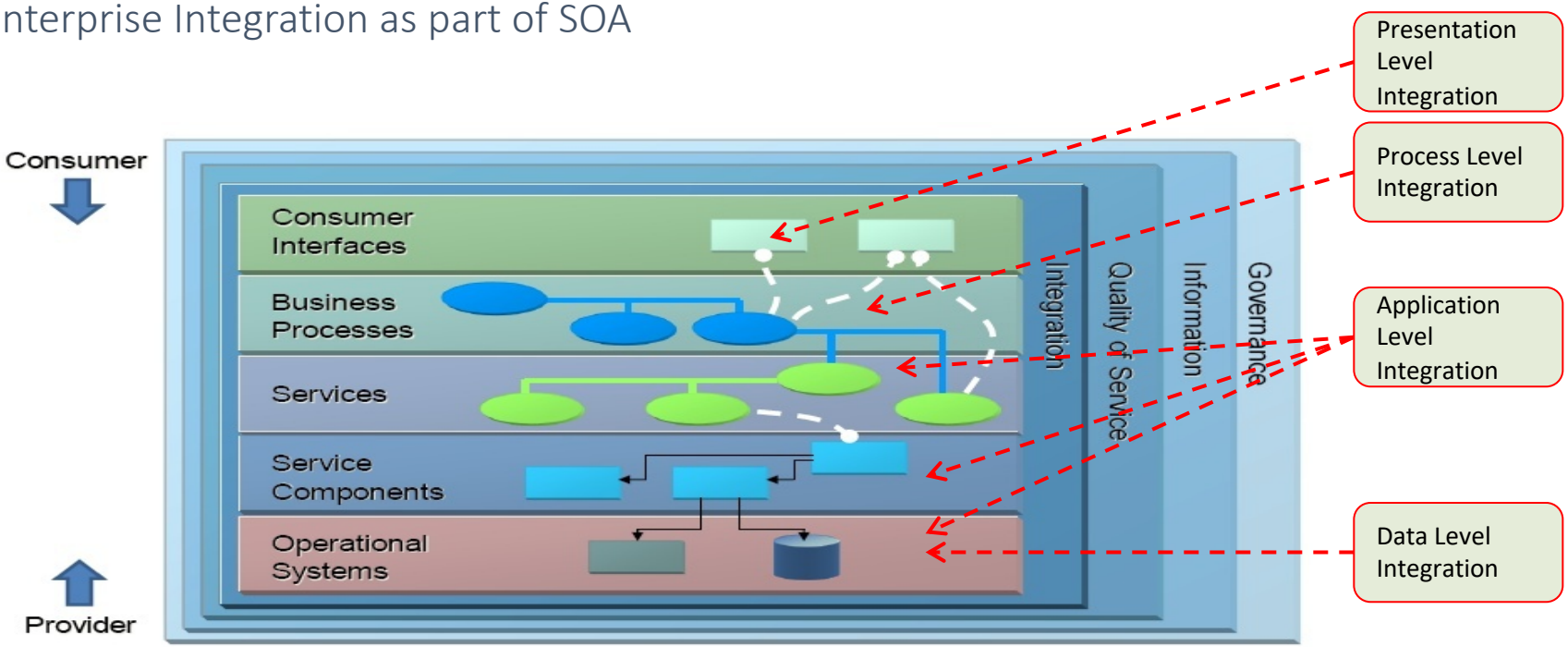
Logical Solution View of SOA Reference Architecture



Source: SOA Reference Architecture Technical Standard, The Open Group

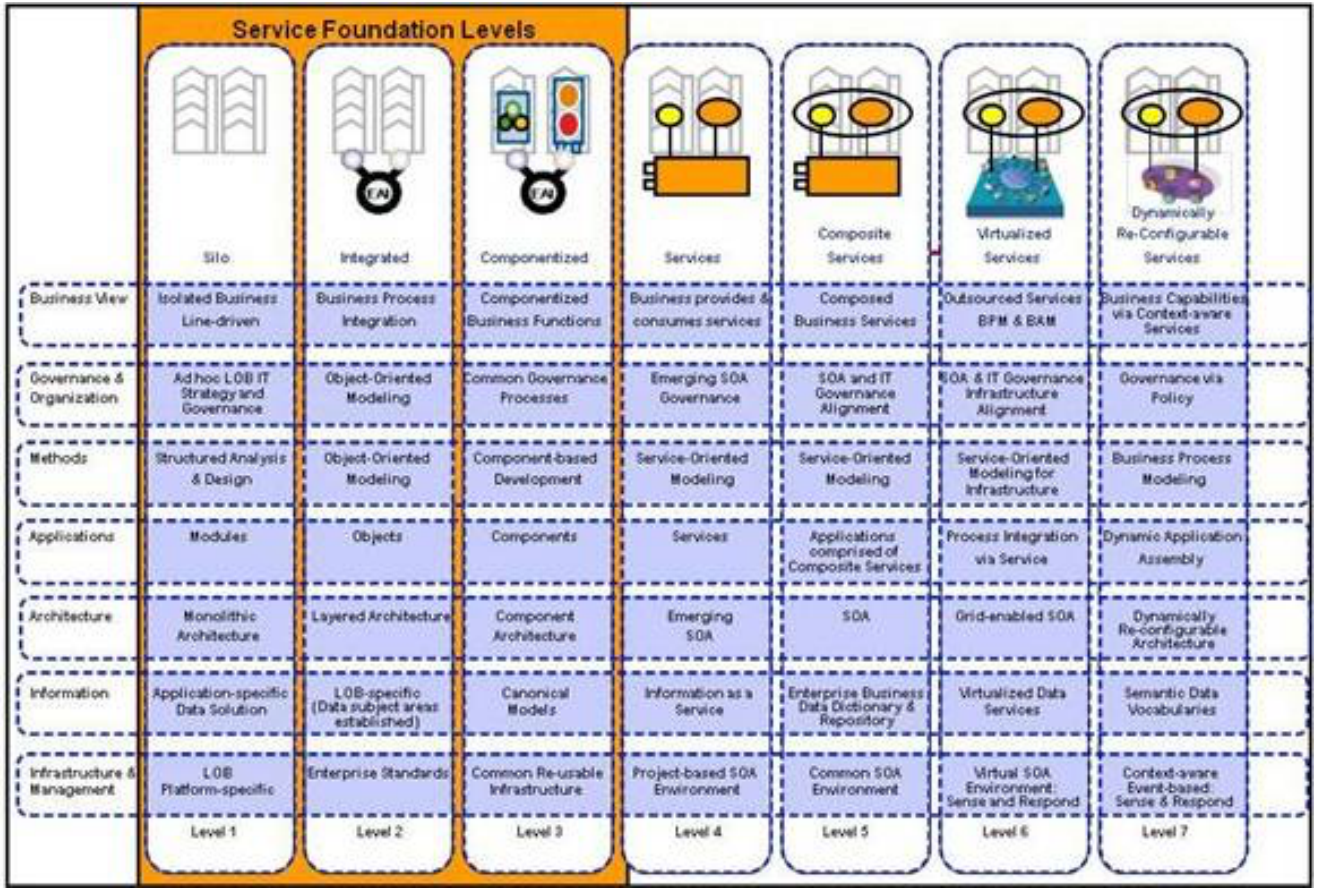
SERVICE ORIENTED ARCHITECTURE 4

Enterprise Integration as part of SOA



Source: SOA Reference Architecture Technical Standard, The Open Group

THE OPEN GROUP SERVICE INTEGRATION MATURITY MODEL V2



IBM'S OSIMM MAPPING