# Business Process Digitalization and Cloud Computing

3. Realizing the promise of SOA

Andrea Morichetta, Phd

October 16, 2017

Computer Science Division

## A definition of SOA

- SOA is concerned with the **independent construction of services** which can be combined to realize **meaningful, higher level** business processes within the context of the enterprise.

- A Service Oriented Architecture describes several **aspects** of services within an enterprise:
  - The granularity and types of services
  - How services are constructed
  - How the services communicate at a technical level
  - How the services are combined together (i.e. orchestrated)
  - How the services interoperate at a semantic level (i.e. how they share common meanings)

- How services contribute to IT and Business Strategy

## Key of success

The usage of SOA permits to improve:

- customer retention and **satisfaction**
- **reduce costs and time** to market
- take advantage from disruptive technologies

From a more technical point of view:

- **integrate** multiple systems
- **support** multiple channels and devices
- **scale** horizontally to support very large-scale and highly reliable requirements
- add new **functionality**
- **converge** on a common service infrastructure
- manage **reuse**

## What went wrong?
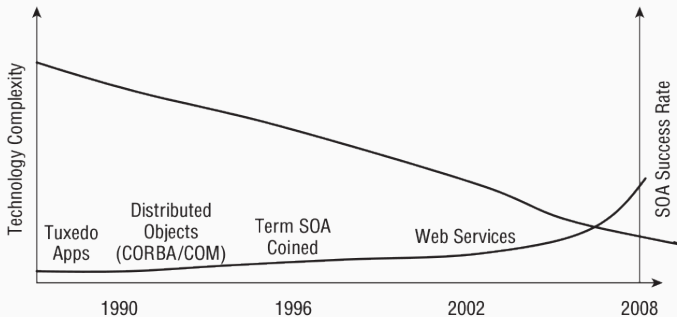


1. Old technology like CORBA, RPC and DCOM was just **too difficult** for the average programmer to master
2. Industries had not figured out **what a good service was** (characteristics, interface, interaction)

## What about today?

1. Web Services are much **easier** to use than previous technologies (tools and environments have been devolved)
2. Services are built into **platforms based** on Java, .NET, or something known.
3. Service **abstraction** layer is built into the Web Service technologies.
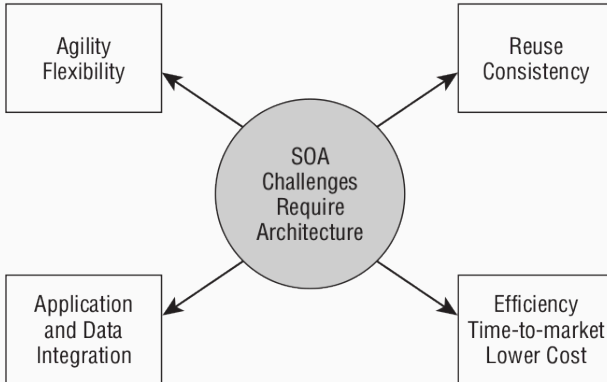
Technology can cause you to fail, but it doesn't make you succeed.

- You need to know **how to use the technologies** to build enterprise applications, not just isolated services

## Reuse
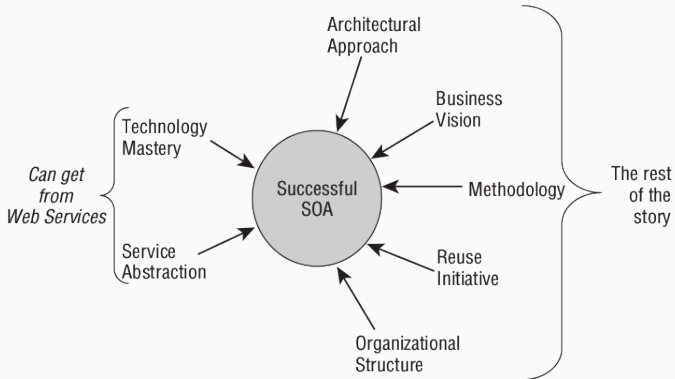
Reusability is based on:

- Ability to publish, search for, evaluate, and **register as a consumer of a service**.
- Capabilities for **managing and maintaining a service** life cycle across organizational boundaries.
- Ability to **guarantee availability** and lifetime of a service version.
- Mechanisms for **decoupling** the consumer's life cycle from the provider's.

To reuse services, you have to be able to **find the services that exist**, and you have to be able to examine them

### Run-time registry
A registry is used at run time to identify a service endpoint for a requested service interface

### Design-time repository
A repository is used at design time to find existing services for inclusion in processes during the design of that process

## Efficiency in development

Making development more efficient means building more functionality, in less time, at less cost.

- Have a **reference architecture** that guides the development of services.
- Use Business Process Management (BPM) to **define business processes**, based on service composition and a layered set of services.
- Have efficient processes that **manage the integrity** of the total set of services for both providers and consumers in accordance with the overall vision and the business and information models.

## Integration of Applications and Data

Difficult integration due to a variety of **different technologies and protocols**

- Have an enterprise, **common semantic model** for the shared information.

- Have a **reference architecture** that differentiates between business services and integration services.

- Have a reference architecture that describes common **patterns for integration**.

- Have infrastructure capabilities that enable **semantic transformation** between existing systems and the enterprise model.

Agility and flexibility occur when new processes can quickly and efficiently be created from the existing set of services.

- Have a reference architecture that defines the **business** and **information** aspects of SOA and their **relationship** to the enterprise.

- Have an enterprise, common **semantic model** that is used to inform the service interface design.

- Use **model-based development** techniques to ensure the traceability between the business models and the implemented systems.

- Have processes that **enable** and **validate conformance**.

## Business process modeling towards SOA

The flexibility of business process comes from the the ability to quickly construct new business processes form business services. So business processes should:

- Be **specified** using Business Process Models and **executed** in a business process management system
- Be **composed of activities** that are implemented by business services (provided by the SOA)
- **Pass information** into, out of, and within the processes in the form of documents, which are built on top of the common information model

**Questions?**