1.4 Características fundamentales de SOA

Raúl Estrada

Noviembre 2020

Overview

- Service-oriented computing represents a new generation distributed computing platform.
- As such, it encompasses many things, including its own design paradigm and design principles, design pattern catalogs, pattern languages, a distinct architectural model, and related concepts, technologies, and frameworks.

Overview

- To better understand the fundamental complexion of a typical service-oriented computing platform we need to describe each of its primary parts, which we'll refer to as *elements*:
- Service-Oriented Architecture (SOA)
- Services and Service-Orientation
- Service Compositions
- Service Inventory
- Conceptual View of Service-Oriented Computing
- Physical View of Service-Oriented Computing

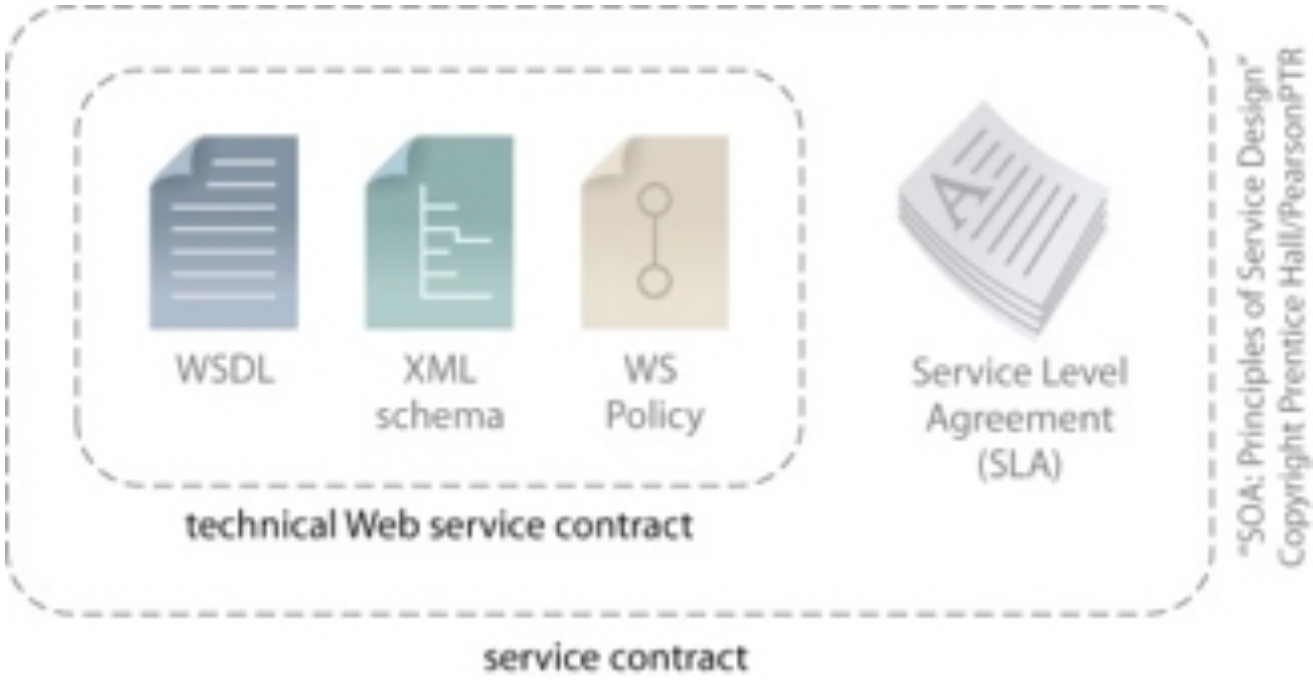
Service-Oriented Architecture

SOA establishes an architectural model that aims to enhance the efficiency, agility, and productivity of an enterprise by positioning services as the primary means through which solution logic is represented in support of the realization of the strategic goals associated with service-oriented computing

Services and Service-Orientation

- *Service-orientation* is a design paradigm comprised of a specific set of design principles.
- The application of these principles to the design of solution logic results in *service-oriented solution logic*.
- The most fundamental unit of service-oriented solution logic is the *service*.
- Services exist as physically independent software programs with specific design characteristics that support the attainment of the strategic goals associated with service-oriented computing.

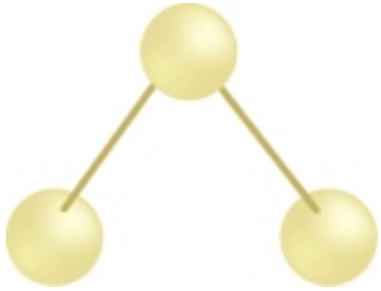
The yellow sphere symbol (left) is used to represent a whole service and the chorded circle symbol (right) is used to express a service and its capabilities.

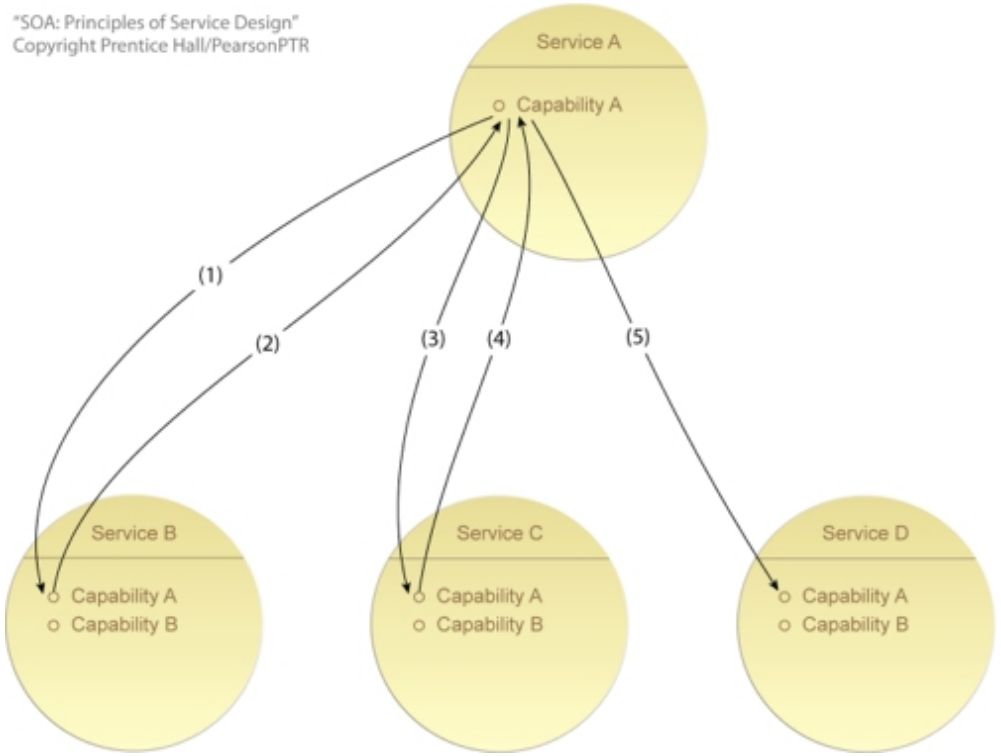


The individual description documents that can comprise a service contract for a Web service.

Service Compositions

- A service composition is a coordinated aggregate of services.
- The consistent application of service-orientation design principles leads to the creation of services with functional contexts that are agnostic to any one business process.
- These agnostic services are therefore capable of participating in multiple service compositions.





The symbol comprised of three connected spheres represents a service composition. Other, more detailed representations are based on the use of chorded circle symbols (right) to specifically identify which service capabilities are actually being composed

Service Inventory

A service inventory is an independently standardized and governed collection of complementary services within a boundary that represents an enterprise or a meaningful segment of an enterprise.

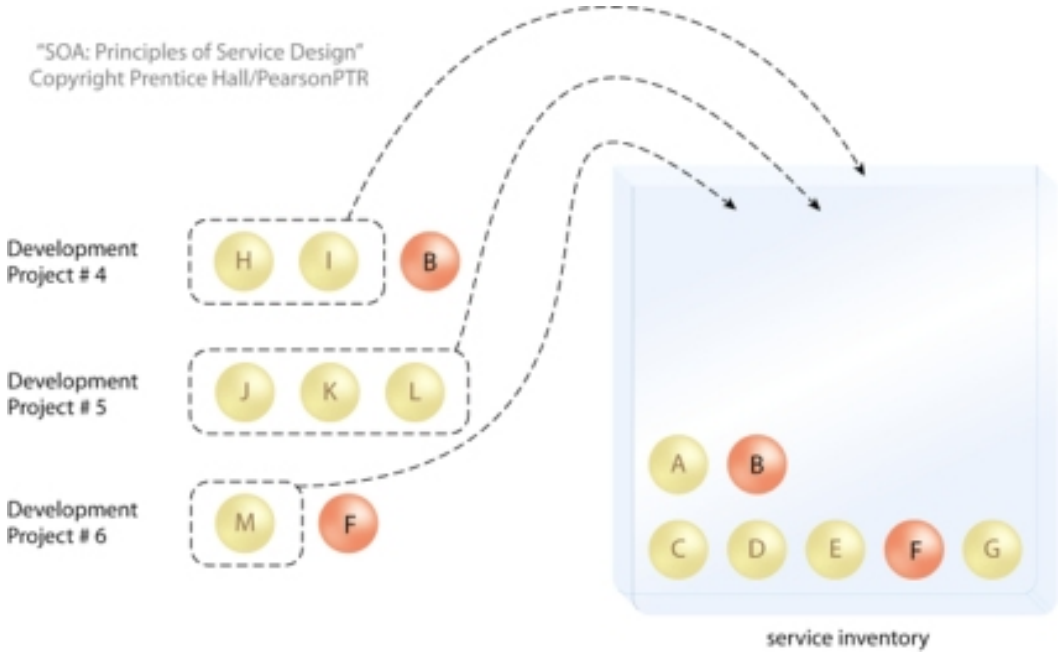
Service Inventory (2)

- An IT enterprise may include a service inventory that represents the extent to which SOA has been adopted.
- Larger initiatives may even result in the enterprise in its entirety being comprised of an enterprise-wide service inventory.
- Alternatively, an enterprise environment can contain multiple service inventories, each of which can be individually standardized, governed, and supported by its own service-oriented technology architecture.

Service Inventory (3)

- Service inventories are typically created through top-down delivery processes that result in the definition of service inventory blueprints.
- The subsequent application of service-orientation design principles and custom design standards throughout a service inventory is of paramount importance so as to establish a high degree of native inter-service interoperability.
- This supports the repeated creation of effective service compositions.

Service Inventory (4)



The service inventory grows as projects deliver new services. Plus, opportunities to reuse existing services increase with each subsequent project.

Conceptual View of Service-Oriented Computing

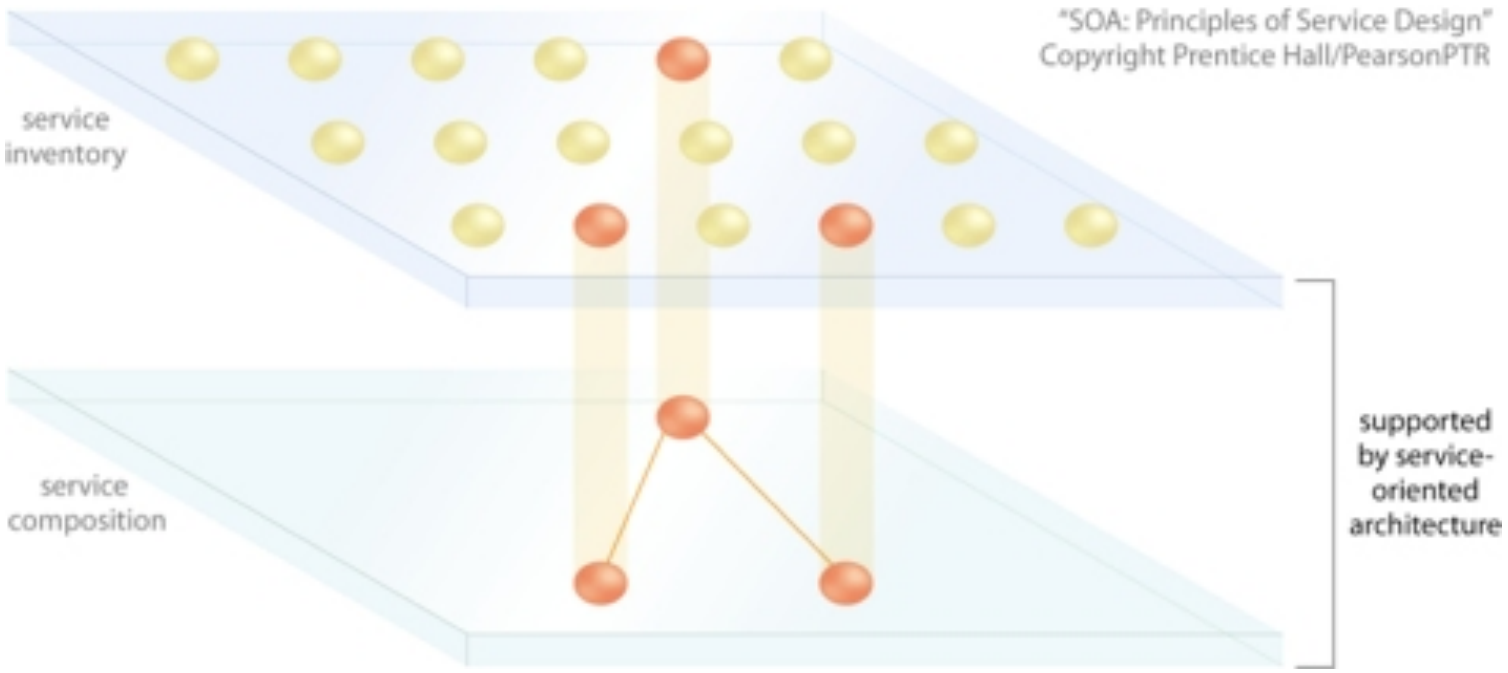
- *Service-oriented architecture* represents a distinct form of technology architecture designed in support of *service-oriented solution logic* which is comprised of *services* and *service compositions* shaped by and designed in accordance with *service-orientation*.
- *Service-orientation* is a design paradigm comprised of *service-orientation design principles*. When applied to units of solution logic, these principles create *services* with distinct design characteristics that support the overall goals and vision of *service-oriented computing*.
- *Service-oriented computing* represents a new generation computing platform that encompasses the *service-orientation paradigm* and *service-oriented architecture* with the ultimate goal of creating and assembling one or more *service inventories*.

Physical View of Service-Oriented Computing

- To fully appreciate how service-oriented computing elements are ultimately used we need to explore how they translate into the real world. To do so, we need to clearly distinguish the role and position of each element within a physical implementation perspective, as follows:
- *Service-oriented solution logic* is implemented as *services* and *service compositions* designed in accordance with *service-orientation design principles*.
- A *service composition* is comprised of *services* that have been assembled to provide the functionality required to automate a specific business task or process.

Physical View of Service-Oriented Computing

- Because *service-orientation* shapes many *services* as agnostic enterprise resources, one *service* may be invoked by multiple consumer programs, each of which can involve that same *service* in a different *service composition*.
- A collection of standardized *services* can form the basis of a *service inventory* that can be independently administered within its own physical deployment environment.
- Multiple business processes can be automated by the creation of *service compositions* that draw from a pool of existing agnostic *services* that reside within a *service inventory*.
- Service-oriented architecture is a form of technology architecture optimized in support of *services*, *service compositions*, and *service inventories*.



A service inventory establishes a pool of services, many of which will be deliberately designed to be reused within multiple service compositions.