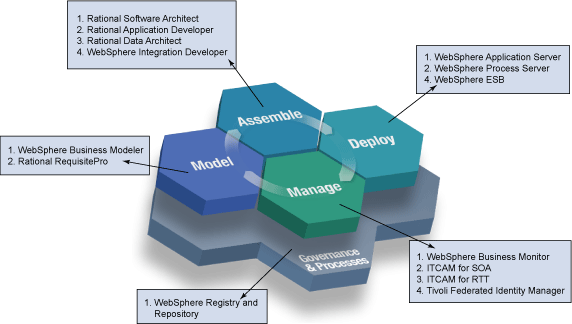
* **SOA Lifecycle and Importance of service choreography.**

**Ans: -**

The SOA governance framework must span the entire SOA lifecycle, bridging and connecting the unique stages. Each lifecycle stage encompasses a distinct yet interconnected set of assets, actors, and activities, from the planning, development, and quality assurance testing of services to their deployment and consumption. Gateways, checks and balances, controls, and other processes and practices can be establish

ed throughout the SOA lifecycle. This helps ensure that SOA stays aligned with business needs and delivers ongoing value.



* **Service choreography**

Service choreography is better understood through the comparison with another paradigm of service composition: [service orchestration](https://en.wikipedia.org/wiki/Service_orchestration). On one hand, in service choreographies the logic of the message-based interactions among the participants are specified from a global perspective. In service orchestration, on the other hand, the logic is specified from the local point of view of one single participant, called the *orchestrator*. In the service orchestration language [BPEL](https://en.wikipedia.org/wiki/BPEL), for example, the specification of the service orchestration (e.g. the BPEL process file) can be deployed on the service infrastructure (for example a BPEL execution engine like [Apache ODE](https://en.wikipedia.org/wiki/Apache_ODE)). The deployment of the service orchestration specification creates the composed service.

|  |  |
| --- | --- |
|  | Basic technologies such as (XML, SOAP, WSDL) provide means to describe, locate, and invoke services as an entity in its own right. However, these technologies do not give a rich behavioral detail about the role of the service in more complex collaboration. This collaboration includes a sequence of activities and relationships between activities, which build the business process. There are two ways to build this process: service orchestration and service choreography.  Service choreography is a global description of the participating services, which is defined by exchange of messages, rules of interaction and agreements between two or more endpoints. Choreography employs a decentralized approach for service composition.  Choreography  The choreography describes the interactions between multiple services, where as orchestration represents control from one party's perspective. This means that a **choreography** differs from an **orchestration** with respect to where the logic that controls the interactions between the services involved should reside. |

# Service orchestration

Service orchestration represents a single centralized executable business process (the orchestrator) that coordinates the interaction among different services. The orchestrator is responsible for invoking and combining the services.

The relationship between all the participating services are described by a single endpoint (i.e., the composite service). The orchestration includes the management of transactions between individual services. Orchestration employs a centralized approach for service composition.

