Service Identification Criteria

This section discusses on the key aspects for a service identification, definition and specification. What are the characteristics to call it out explictly as a service rather than a component. The idea is to have a standard practice and governance model to identify services based on the contexts of usage, who are going to use, busines needs addressed and finally architecture fitment with respect to SOA core principles. This will help reduce another accidental architecture or hairball architecture in future. It takes some of the inputs from component vs services model.

Follow Core SOA Principles

- Consider Quality of Service across the entirety of the service platform and not at component level (Micro service)
- Understand clearly the business processes
- When identifying services, use related behavior as the guideline to define the scope of the service
- Follow SOA Guidelines:
    - Design the contract first
    - Design for service reusability and composition
    - Design for loose coupling
    - Design for orchestration
    - Design for data representation
    - Use schemas for data representation
- Ensure design for “enterprise consistency”: services that can effectively be reused through out the enterprise
- Standardization of enterprise business processes and creation of a canonical data model within the Information architecture can facilitate "Enterprise Service" design

Understand Business Granularity and Reusability

Granularity is the balance of reusability versus scalability and performance, as we move from lower tier e.g. data tier to top tier e.g services tier, the granularity increases from finer grained to coarse grained.

Check if the services planned has the right granularity to accomplish a business unit of work in a single interaction. A service would be regarded too coarse-grain if the size of exchanged messages grows and sometimes might carry more data than needed, or presents a complex interface which is prone for regular changes. On the other hand if the service is too fine grained multiple round trips may be required which would introduce quality concerns and outflow of services. A balance is hence required between level of abstraction, likelihood of change, complexity of the service, and the desired level of cohesion and coupling. Though in some cases a tradeoff needs to be made while taking into account non-functional requirements particularly performance. Its recommended to have a discussion with architecture and platform soari team to understand what is the best approach for your service model. Some of the below factors need to considered while deciding the granularity.

- **Coarse Grained**—Follows from business process decomposition, reduced flexibility and scalability, reduced overhead per service call, decoupled from underlying logic, limited no of services needed to orchestrate business process, sequencing of operations handled within service and less transaction between consumer and provider.
- **Functional Reusability**—Instead of developing a services for a specific consumer, first check if there is any overlapping of functionality of your service offering is available in any other service across organization, if yes, try to understand why those can not be reused or merged by appropriate adjustments to contract, message exchange or additional operations / resources which internally may either use your service component or their service component. Secondly check how many consumers are there at present or will be there in future (immediate 1-6 months, 6-12 months, 12-24 months time frame), its recommended unless there are atleast 3 or more consumers to consume your service in near term, it does not make sense to host it as a service for the sake of it.

Understand Service Usage Type

Process Service—A service whose operations are guided by the process definition. These are reactive services which need business events that would trigger various activities that would be using business and information services.

Business Service—A service encapsulating transactional nature of functionalities that would build business context over other informational service.

Composite Service—A service with either composition or aggregation of multiple other services. The internal invocations are abstracted from the consumer providing a unified view. An orchestration would help composite service to be synchronous in nature and choreography would help composite service to be asynchronous.

Informational Service—Services that focusing on providing processed data and whose operations are atomic, executed and realized by one provider on a particular type of runtime environment/platform.

Data Services—Services that provide normalized and aggregated view of critical data entities (or master data) such as Customer, Order, Claim and so on. These services are often realized along with Master Data Management strategies.

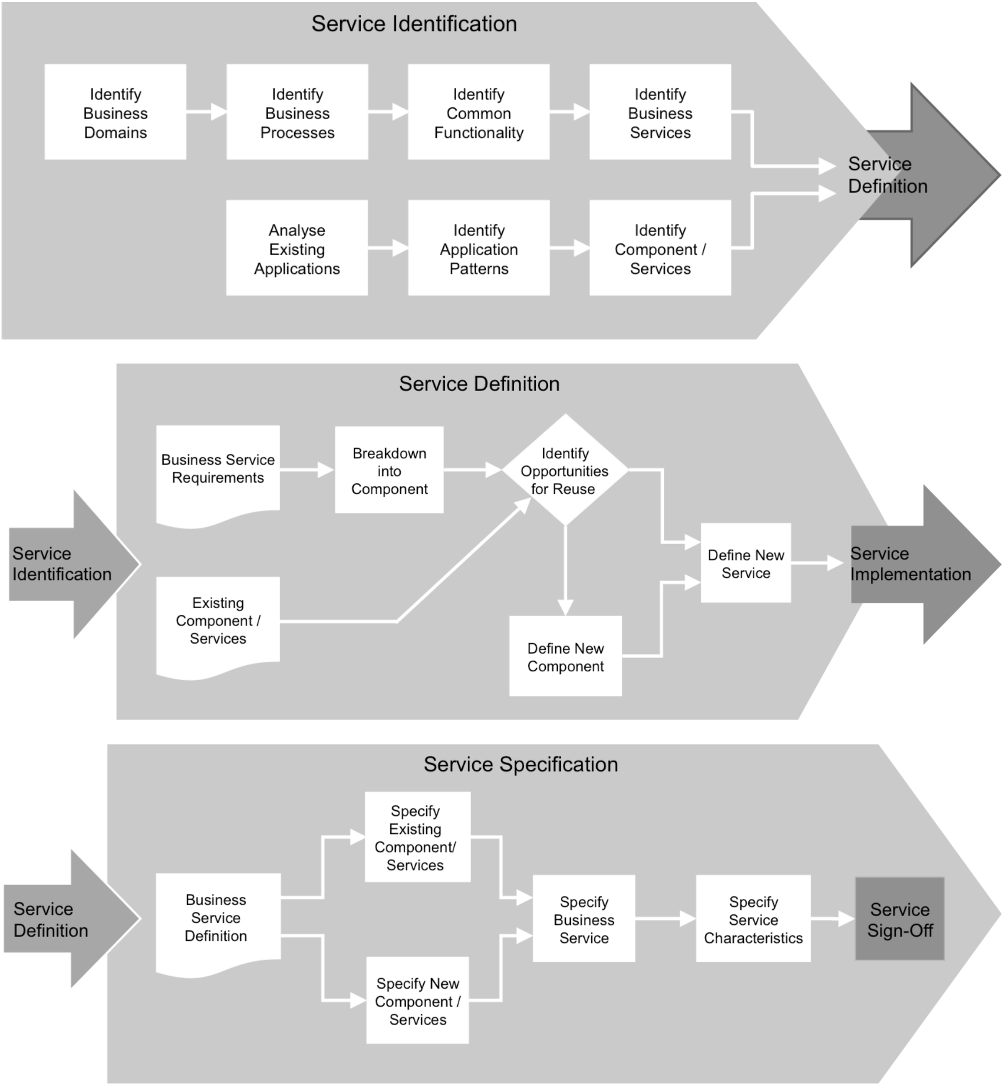
Utility Service—A service, whose operations are, shared among various services due to the commonly accepted practices or standardization such as payments, credit card transactions etc. Due to the utility or commodity nature of these services, business might often like to use the best possible provider may be from external sources too.

Infrastructure Service—A specialized technical automation service that provide essential infrastructural capabilities to other services. For example - Cache, Queue, Log, AlertEscalation, AuthN/AuthZ etc.

Partner Services—A manifestation of Business, Informational or Data Service offered to external business partners based on agreed terms.

Follow a Model/Methodology

Follow a model to derive and identify a service and finally define the specification for implementation.


Service Identification
Identify Business Domains
Identify Business Processes
Identify Common Functionality
Identify Business Services
Analyse Existing Applications
Identify Application Patterns
Identify Component / Services
Service Definition
Service Definition
Service Identification
Business Service Requirements
Breakdown into Component
Identify Opportunities for Reuse
Existing Component / Services
Define New Service
Define New Component
Service Implementation
Service Specification
Service Definition
Business Service Definition
Specify Existing Component/ Services
Specify New Component / Services
Specify Business Service
Specify Service Characteristics
Service Sign-Off