

### Feature Oriented Software Re-engineering:

The feature analysis approach discussed in [1] presents a reengineering process in uses domain features to identify the candidate services. It involves three basic steps.

1. Feature Identification and Feature Model creation: The approach interprets feature as any recognizable deliverable to the customer or any coherent group of functionalities visible to users. The technique looks for features in problem domain and transfer into web services in solution domain. Moreover, in order to have a clear notion to recognize any functionality as a feature, the following requirements are provided.

- a. the functionality is used to specify the capability of the system
- b. the functionality is visible and identifiable from the user's perspective
- c. the functionality is an instance of a domain feature
- d. the granularity of the functionality should be coarse grained without addressing the computational detail
- e. if the functionality is identified as a feature and is indecomposable based on users' perspective then the feature is an atomic feature
- f. two or more atomic features can compose a composite feature based on business rules

The feature model is constructed based on the relationship between the features following the approach given by FODA (Feature Oriented Domain Analysis). The intent of feature model is to give a clear understanding about the system under consideration. A feature model is achieved through feature identification, feature classification and coherent organization and by consulting various domain information source such as current system, domain experts etc.

2. Feature location: It is the process of locating the implementation of feature in solution domain and filtering the implementation to achieve coherency. There are various studies done in order to achieve this. A few examples are test case technique and program slicing.

3. Feature aggregation: The interaction between the components are studied and aggregated into feature-oriented components.

4. Service Identification: Finally, the service can be realized to satisfy one or more features. A service should be as coarse grained as possible, complete, coherent and should follow consistency. The criteria of grouping features into a service depend upon modeling approach taken. [1]

### M4SOD (Method For Software Oriented Development):

The services can be derived from use case model as use case can explain business workflow in a better way. There are following phases for the process.

1. Service Development Planning
2. Business Domain Modeling
3. Legacy System Detail Analysis
4. Service Identification and Modeling
5. Realization Strategy Decision
6. Service Development and Integration

The business domain analysis for new system or the legacy system can be done using use case model.

The use case model thus obtained will have various levels of use cases. Each use case provides a workflow and is composed of various task sets. The objective is to provide some rules to utilize these tasks sets as individual services.

Each use case can be a good candidate for a service. However utilizing the use cases directly will not help us in simplification of governance, possibility of independent scalability, performance as well as management optimization.

Guidelines:

1. One to one correspondence: if task sets of a use case is independent of the task sets of other use cases, the use case can be utilized as a service.
2. Decomposition: If a use case can be divided into two or more functionally independent partition, then each partition can be realized as a service
3. Composition: If two or more small-grained use cases have similar tasks, then they can be merged and realized as a single use case.
4. Derivation from Common Tasks: Optionally if the common task refers to an important business workflow then the common task can be realized as an independent service.
5. External Legacy: If there is a use case working with external system where the external system is presented as an actor, the use case is also considered important to business workflow so can be realized as an independent service. [2]

References:

1. [http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1607153&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs\\_all.jsp%3Farnumber%3D1607153](http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1607153&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D1607153)
2. <http://ieeexplore.ieee.org.eaccess.ub.tum.de/stamp/stamp.jsp?tp=&arnumber=4026934>