services interoperable. Two systems are component and services interoperable if they agree on the set of services offered by the components of the two systems and the interfaces to these components. By defining standardized interfaces to these components, the components of one system will be able to request services from components in another system.

The computational viewpoint clause provides the following:

— Defines and discusses the concepts of component, interface and service and the relations between these concepts

— Provides an approach to physical distribution of services using an n-tier architecture.

— Defines a model for combining services in a dependent series to achieve larger tasks – service chaining

— Defines a service metadata model to support service chaining through a service catalogue

6.2 Services, operations and interfaces

The relationship of three terms is provided in this clause. The three terms are used extensively in this International Standard:

— **Service**: A collection of operations, accessible through an interface, that allows a user to evoke a behaviour of value to the user.

— **Operation**: specification of an interaction that can be requested from an object to effect behaviour

— **Interface**: an implementation of operations including the syntax of the interaction for a given distributed computing technology.

The terms are related to each other as depicted in Figure 2. Abstract services and operations provide the definition of a behaviour of value to an intelligent user, most typically a human. The human uses a service to create a value-added product, where the value is judged by the human, i.e., semantic value. Operations are implemented by interfaces that can be accessed by a client implementation. The interface implements the syntax that is defined in a specification of an operation.

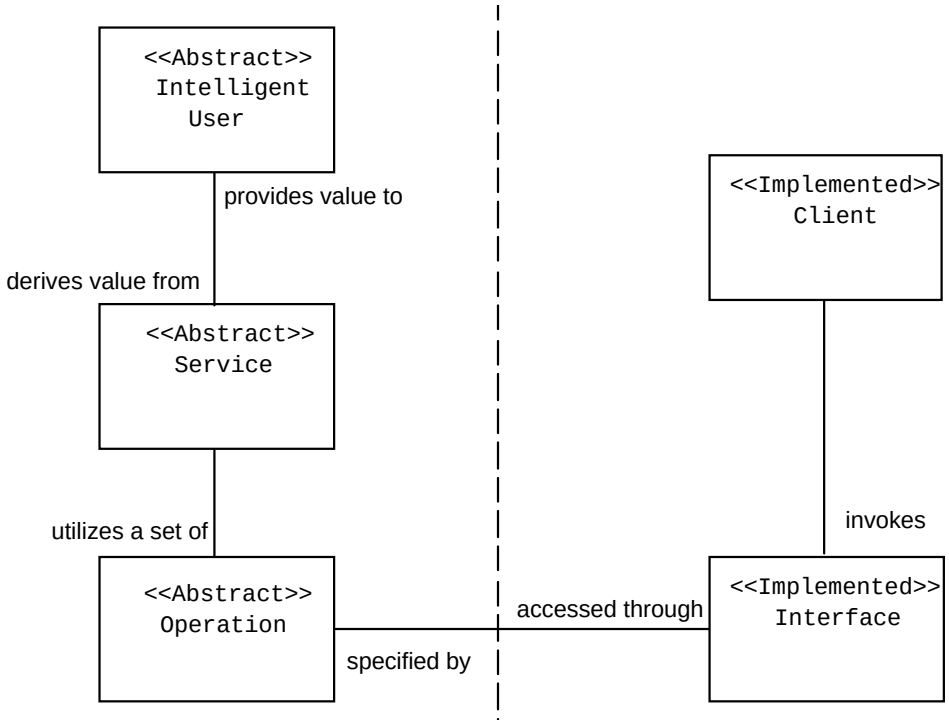

Figure 2 — Service definition relationships

The following provides further discussion of the elements of the definition of Service.

- "A collection of operations"

Multiple operations allow for an activity to be accomplished in multiple ways defined by the context of the specific interaction.

- "Provides a behaviour of value to a user"

Accomplishes a task that adds value that was not present before the service was invoked. The value is apparent to the user who invoked the service. The added value is possible through the aggregation of individual operations.

- "Implemented as an interface"

A service and its operations may be implemented in a variety of ways depending upon technology and situation specific variations.

Discussion:

A service provides access to a logically interesting and related set of operations accessible through one or more interfaces, implemented in one or more interfaces. A service has a static and a dynamic part. The static part describes the signatures of the operations that are implemented in the interfaces. The dynamic part consists of one or more interactions that show message flows over time in the service to perform computations.

A service may be expressed at various levels of granularity. A coarse-grained collaboration may be refined to produce a service that has a finer granularity. This is accomplished by expanding one or more operations from a high level collaboration into distinct lower level services, one for each operation.

A service may be implemented in terms of subordinate services. Each subordinate service implements a part of the overall functionality and has its own set of roles. Each role of the overall service may be bound to one or more realization of the nested service.

An instance of a service may be associated with a specific instance of a dataset or it may be a service that can be used to operate on multiple, unspecified datasets. The first case is referred to as a tightly coupled data and service. The second case is referred to as loosely coupled service. Service operations can be associated with data classes (data type) or with instances (data set).

An interface is a grouping of the externally visible operations of a implementation, without specification of internal structure. Each interface often specifies only a limited part of the behaviour. More than one component may provide an implementation behind the same interface.

6.3 Distributing components using a multi-tier architecture model

To support flexible deployment, IT architectures are structured as multi-tiered distributed architectures. As a reference model, a logical 4-tier architecture is presented with discussion on variations in different physical architectures. The logical architecture is the arrangement of services and associated interfaces that are present in the system (see Figure 3). The physical architecture is the arrangement of components and associated interfaces that implement the services. The components are hosted on hardware computing resources or nodes.

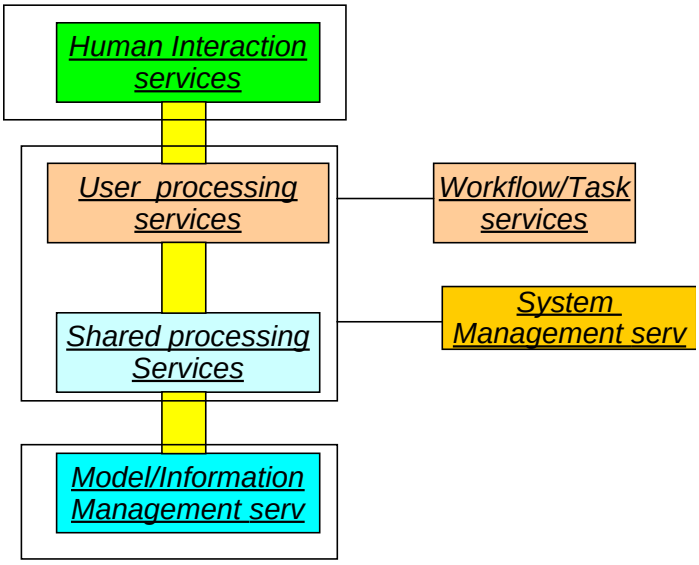

Figure 3 — Logical multi-tiered architecture

OSE model, as defined in ISO 19101, structures the types of services of an IT system. Each tier can contain both IT-general services and GIS-extended services for that tier.

— The *Human Interaction service* tier is responsible for physical interaction with the user, through display and input media, and an appropriate dialogue.

— The *Communication services* are responsible for connecting the various tiers together (although not labelled in Figure 2, the communication services are present as connections between the other service tiers).

— The *User processing service* tier is a part of the processing services responsible for the functionality required by the user

— The *Business processing service* tier is part of the processing services responsible for common services (both domain specific and general) that can be used by multiple users.

— The *Model/Information Management service* tier is responsible for physical data storage and data management.

— The *Workflow/Task services* is a set of services that can be viewed as a specialized processing service