**Capability Management**

Table of Contents

[Capability Management 2](#_Toc447014899)

[Capability Management Framework Model 2](#_Toc447014900)

[Key Terms for Capability Management 3](#_Toc447014901)

[Capability: 3](#_Toc447014902)

[Capability Management: 3](#_Toc447014903)

[Types of Capabilities 4](#_Toc447014904)

[Why do enterprises perform Capability Modeling? 5](#_Toc447014905)

[Clearly Defined Capability Management Ownership is Critical for Success 6](#_Toc447014906)

[Understanding Capability Classifications or Categorizations 7](#_Toc447014907)

[Capability Management Ontology as a Means for Language Standardization 8](#_Toc447014908)

[Lifecycle Phases for Capability Management 8](#_Toc447014909)

[Capability Management Rule Sets / Constraints 10](#_Toc447014910)

[Common Principles and Best Practices for Capability Management 11](#_Toc447014911)

[How to Create a Capability Model 14](#_Toc447014912)

[Level 1 Enterprise Area 14](#_Toc447014913)

[Level 2 Business Areas 14](#_Toc447014914)

[Level 3 Business Department Level 14](#_Toc447014915)

# Capability Management

## Capability Management Framework Model

To elaborate, this artifact is intended to:

1. Organize different areas of the discipline known as Capability Management into clear and compartmentalized areas that allow the Foundation to more effectively and productively collect, document and publish information that pertains to this discipline.
2. Decompose each area of Capability into smaller and, therefore, more digestible units for more efficient learning and understanding.
3. Document common industry wisdom about each area, piece or subcomponent of Capabilities
4. Act as a set of Capability Management related *best practices* and *guidelines* that have been collected, documented, and published for the benefit of colleagues of the IT Organization, industry, line of business, or area of expertise.
5. Act as a consistent and repeatable pattern for documenting, publishing and learning, both, within this Discipline and across "all" Disciplines.

From the Foundation's perspective, if done correctly, all of the above will allow the Foundation to properly decompose, document and publish content related to each sub-area or sub-topic for each Discipline.

As this artifact evolves and progresses, the reader will see it address key areas of the professional Discipline "Capability Management" that range from its detailed definition through closely related terms, phrases and their definitions, to its detailed specification of a Capability, and all the way through to defining, delivering, operating and supporting a Capability Management.

## Key Terms for Capability Management

Before moving on to learn more about the rest of the Capability Management framework, we suggest that you take some time to familiarize yourself with the following very basic term(s)...

### Capability:

*A manageable feature, faculty, function, process, service or discipline that represents an ability to perform something which yields an expected set of results and is capable of further advancement or development.*

### Capability Management:

*The professional discipline that involves working with, in or on any aspect of planning, delivering, operating or supporting for one or more Capability Items or any and all solutions put in place to deal with such Items.  
  
The solution set that a person or organization puts in place to manage one or more Capability Items.  
  
The process or processes put in place by a person or organization to assist in the management, coordination, control, delivery, or support of one or more Capability Items.*

## Types of Capabilities

When it comes to Capabilities, there are multiple types that an enterprise needs to be aware of. Examples include but are not limited to:

* **Business Capabilities**

A set of Capabilities needed and/or offered by a Business whereby Business means also the Business Unit IT.

* **Application Capabilities**

A set of Capabilities needed and/or offered by a System. In development also defined as Use case, or Application Function

* **Solution Capabilities**

A set of Capabilities needed and/or offered by a Solution. Like: the Business Solution Repository (BSR)

* **Technical Capabilities**

A set of Capabilities needed and/or offered by a technology

This now brings us to a very obvious problem that surrounds Capabilities, which is the fact that there are simply too many "granular" or "specific" Capabilities to document and publish in any single Capability Model. The end result is that a Capability Model may become unwieldy because of trying to incorporate so many different specific Capabilities. Also, Capability Modeling "Purists," who all have their own (and very differing) opinions about how Capability Models should or should not be represented, almost always refuse to get into the details. To address this we will have different types of capabilities in different Models as described above.

As you can see from the above, we now have a very limited, controlled and manageable set of Discipline-specific Capabilities for the Discipline Capability Management.

## Why do enterprises perform Capability Modeling?

Enterprises most often build Capability Models for the following reasons...

* To create a common enterprise language for the Discipline known as Capability Management.
* To create a common and reusable Capability Taxonomy Such Taxonomies can be used to help layout or structure an Business Process, Structure Application Features and Software Architectures.
* To help categorize and structure Requirements during Requirements collection and analysis. With those Requirements (Scope) we can visualize the scope of Solutions exactly and have structured discussions before implementations.
* To help map things like Applications, Software, and Hardware to the Capability areas.
* To help define enterprise Services, along with helping to understand who provides and consumes such services.
* To help understand what activities go on, in an enterprise, and who performs such activities.
* And much more...

**Capability Modeling Recommendations:** Some things to consider and keep in mind when working on or creating your Capability Models...

* Try and keep "context" or "contexts" out of your core model. For example, your Capability Stakeholders may work hard to get what's important to them into the model (e.g. their "organizations" or organizational contexts), as opposed to what might actually be the enterprise. It's always a good idea to keep the model as pure as possible and keep such contexts out.
* While it's a good idea to keep contexts out of your core Capability Model, it will eventually become clear that Capability Contexts are, in fact, very important to your enterprise. These type of Context is the Business Architecture Model (Business Processes from a Value type of view and the detailed BPMN Process Models which describe the implementation of value Flows from a Business perspective. In addition the organizational perspective can be stored here as well.
* One of the quickest ways to lose the interest of your enterprise is to keep your model secluded or silo-ed in a spreadsheet or database and not publish it or use it for solving other problems in the enterprise. So therefore store the capability Model in the central Repository and let others participate.
* You will constantly be under pressure to help define and get your stakeholders to understand what a Capability is. It's for this reason that you may want to use terminology that might be more common with Businesses, such as "Services," "Functions," and/or "Features."
* It's better to start with a model you can go into your Business Stakeholder meetings with, with the intent of improving upon what you have, than it is to go in with nothing at all and build your models from scratch. Therefore use those capability Models which are in the repository already. Folder : Reference Models
* Good Capability Models will have many layers but start small. Start to define first the highest layer and use one Area of interest (one Business line and look into high level Business Process) to define Business Capabilities. If you have a hierarchy level defines decompose these and define the detailed capabilities.
* Keep your model published and fresh so that the enterprise can see it and use it, immediately. The more eyes you have on your model, the faster it will improve. To not confuse others post only those versions which are “approved” between all stakeholders.

## Clearly Defined Capability Management Ownership is Critical for Success

Here's a very simple fact... If an enterprise does not establish and enforce clearly defined Ownership (i.e. a Resources and his or her Organization are assigned as accountable ownership) for Capabilities, the enterprise has automatically set itself up for failure in its implementation of that discipline. Therefore, if you and your enterprise want to implement and maintain a successful solution for the creation of Capabilities, there must be a clearly defined Owner that can and will be held accountable for getting work done and providing transparency.

Having clearly defined Ownership should not be confused with having fully dedicated Resources that spend one hundred percent of their time working on Capabilities.

In summary, if you and your enterprise truly want to be successful with your implementation of Capability Modeling, ensure that a clear and highly accountable owner is identified and assigned to the Discipline. Publish those ownership details, preferably in a Repository Portal, and socialize it so everyone knows whom to go to for answers and for help with the creation of Capabilities related work.

## Understanding Capability Classifications or Categorizations

A **Taxonomy**, in its noun form, is defined as:

*...a documented and orderly set of types, classifications, categorizations and/or principles that are often achieved through mechanisms including but not limited to naming, defining and/or the grouping of attributes, and which ultimately help to describe, differentiate, identify, arrange and provide contextual relationships between the entities for which the Taxonomy exists.*

In short, what this means all means is that a Taxonomy is nothing more than a classification or typing mechanism and that a Capability Taxonomy is nothing more than a classification or typing mechanism that helps people and systems distinguish between different Capability Items, Entities, Types, Records or any other Capability Management element you can think of.

It's important to understand that Taxonomies can be as simple as a list of relevant terms or phrases with respective meanings or definitions or they can take on more complex forms, such as hierarchical and graphical model structures that can be homogeneous and heterogeneous in nature.

Ex. If we link a Capability to an organization then we can get a report: Which kind of Capabilities are for a special Organization (LoB, Industry) and which kind of Capabilities are shared.

## Capability Management Ontology as a Means for Language Standardization

While Taxonomies represent organized classifications or types, you can think of Ontologies as the design and representation of entire languages, with the specific intent to control things like structure, behavior, representation, and meaning. Without getting into a theoretical conversations about Ontologies, you can view this entire article as a foundation for the ontology of Capability Management. Or, in other words, a Capability Management Ontology.

Throughout this artifact/framework, you will find things like Capability Management related terms, phrases, definitions, roles, responsibilities, nouns, verbs, classifications, and so on, all as a means of defining a standard representation for and interpretation of the language of Capability Management.

It is only through the definition, communication, and establishment of such Ontologies that we can standardize language and communication associated with Capability Management, whether it be between humans and/or systems.

## Lifecycle Phases for Capability Management

When we talk about *Life Cycle* (or *lifecycle*) for Capabilities, it's important to keep in mind that there are two different types of Life Cycles that apply. The first is a **Data Life Cycle**, which addresses linkage of Capabilities and data entities. As we typically do not have the understanding in the Business of Entities or Entity Relationship Models. Let us use the nomenclature of the business like Information or Information Views. These Information View can be directly linked to a Capability and give us the understanding of linked Business Objects. The second is associated with delivering Capabilities like Software Solutions or IT-Practices. As IT Practices will have typically more than one capability. The linkage is from the IT-Practice to the Capability Model.

**Capability Management Data Life Cycle Phases:**

Data Lifecycle (or Life Cycle) for any and all data is the period from the "inception" of data through to its ultimately being "purged" from existence. This is no different for Capability related data.

Like the data associated Capability related data adheres to the following common Data Lifecycle Phases:



**Figure: Capability Management Lifecycle Phases**

* **Inception:** Data is in its raw idea-like form and is not ready for consumption by the general population because it has not been final documented and is in the status DRAFT.
* **Creation and Registration:** Data is formally put into existence for day-to-day use by appropriate stakeholders. The Status is Production.
* **Iterative Maintenance:** Data is in a mode of constant use and is updated and modified, as needed, to meet the needs of daily use by various stakeholders.
* **Decommission and Deletion:** Data is prepared for deletion and eventually deleted from daily operational use but still exists for administrative or organizational purposes, such as historical auditing. It can be restored to any one of its relevant last states and, therefore, can be brought back into existence for day-to-day use.
* **Purged From Existence:** Data is completely removed from the Capability Model or Entity Relationship Model with no means to restore or reconstruct it, without recreating it from scratch and with no guarantees that it will match it's previous state.

The above Life Cycle Phases represent the high level transitions that occur from the inception of Capabilities or Entities all the way through to their complete elimination from existence. A more detailed breakdown of these transitions or phases represents what are referred to as "Capability States."

## Capability Management Rule Sets / Constraints

The successful implementation of Capability as a set of Constraints for your enterprise usually implies that a number of key components have been established to support it. These components are:

1. *A clearly documented and socialized Capability Owner that is held accountable for a Capability or a set of Capabilities.*
2. *For every Capability there should be a clearly defined data relation. (Information View) To help to understand the relation of a capability to the used Information context.*
3. *It's important to understand who your Capability Stakeholders are, this includes but is not limited to your Customers, Consumers, Clients, Sponsers, etc. are, as well as the types of problems it is that they're trying to solve or interests that they will have in the Capability.*

**Capability Ownership:** The most important thing to understand about a Capability is that, there must be a clear and accountable Owner for it. That is, there needs to be a very clear and accountable named person or organization that owns and is fully responsible for the Capability. Without clear ownership, Capabilities are getting outdated and useless

## Common Principles and Best Practices for Capability Management

A "**Principle**" is defined as being: *"A professed assumption, basis, tenet, and doctrine, plan of action or code of conduct for activities, work or behavior."* Therefore, we can deduce the definition of "a Capability Principle" to be:

**Capability Principle:** *"1. A professed assumption, basis, tenet, doctrine, plan of action or code of conduct for any activities, work or behavior associated with the Discipline known as Capability Management."*

A "**Best Practice**" is defined as being: *"One or more Activities, Actions, Tasks or Functions that often do not conform with strict Standards and that have evolved, over time, to be considered as conventional wisdom for consistently and repeated achieving Outcomes or Results that can be measured as being equal to or above acceptable norms."* Therefore, we can deduce the definition of "a Capability Best Practice" to be:

**Capability Best Practice:** *"One or more Capability related Activities, Actions, Tasks or Functions that often do not conform with strict standards and that have evolved, over time, to be considered as conventional wisdom for consistently and repeatedly achieving Outcomes or Results that can be measured as being equal to or above acceptable norms."*

Common Capability related principles and best practices exist to help achieve higher than average expectations of quality and to ease in the implementation, support, operations, and future change associated with the solutions industry professionals put in place to address the needs of this Discipline and all its related stakeholders.

While this entire document is meant to represent and serve as a set of common principles and best practices for Capability Management, the following list represents a summary of some very basic examples of what implementers, supporters, and operators of Capability Management should constantly be working toward:

|  |  |
| --- | --- |
| **Principle or Best Practice** | **Description** |
| Establish and always have very clear Ownership for Capability Management. | Establishing, publishing and socializing clear Ownership for Capability Management allows an enterprise and all its Resources, regardless of their geographic location, to assign accountability for all aspects of the Discipline. It also ensures that there's always at least one person that everyone can go to for transparency into the Discipline as well as for handling work that is associated with the Discipline. |
| Always use standard terminology for Capability Management, in order to standardize communications between stakeholders. | It is often argued that the biggest mistake you can make is to create your own words and/or your own definitions, when communicating with others. There is no place where this is more accurate than in the field of Information Technology. IT Stakeholders make up their own words and definitions far too often, or let their business constituents do so. When you make up words or definitions, or you let others do so, you're creating a grave injustice for your organization. Self-invented terminology and grammar often leads to poor communications, which in turn leads to redundancy of solutions, higher complexity of environments, slower delivery times, and much higher costs. Therefore, the IF4IT always recommends that you leverage standard terminology for Capability Management, whenever possible. |
| Centralization of Capability related data. | While often impossible to centralize and collocate all Capability related data and information, especially in a geographically dispersed environment, Capability Management related stakeholders should always strive to centralize all data and information. The goals are to eliminate data fragmentation, improve source of truth for data, reduce the number of systems needed to support stakeholders, reduce the complexity of solutions, improve usability, and to ultimately reduce the costs associated with Capability Management. |
| Clearly define, implement, track, and analyze Capability Management Metrics. | In order to successfully set up the discipline of Capability Management and its related Services, it is critical to clearly define, track, and constantly analyze Capability Management metrics. Such metrics include but are not limited to Supply and Demand Metrics (i.e. Operational Metrics), Performance Metrics, Quality Metrics, and Financial Metrics. |
| Transparency of Capability related data. | Stakeholders should always strive to make any and all Capability Management data transparent to all other appropriate stakeholders, at a minimum, and often to the entire enterprises. The exception when private user data must be protected. Many stakeholders often make the mistake of treating internal operational data as private or protected. This often creates a data silo and will often lead to internally silo-ed organizations that revolve around such data silos. |
| Do not let "perfection" of Capability Management solutions stand in the way of "good enough solutions". | Often, Capability Management stakeholders "overthink" solutions, leading to the impression that best-of-breed or perfect solutions are more effective than "good enough" solutions. Experience tells us that "good enough" is, almost always, the better path to follow. We live in an age where technologies grow old in the blink of an eye. Even the implementation of something that looks perfect, today, will look antiquated, tomorrow. This is especially true if your enterprise doesn't have a long term funding plan and commitment to improvements and upgrades of the solution(s) put in place. |
| Follow industry Standards, Best Practices, and Guiding Principles for Capability Management, whenever possible". | One of the most common errors many enterprises make is to create solutions from scratch or without the guidance, assistance and/or experience of others who have created such solutions, before them. Whenever possible, the IF4IT recommends that you research existing Standards, Best Practices, and Guiding Principles to avoid the mistakes of others, while also gaining from their successes. Remember, we live in a vast world. Chances are very high that someone else has already experienced the pain you're about to create for yourself. Wise people will always look to learn from such people's experiences before they go down the road of implementing their own solutions. |
| Work toward and maintain a Single Source of Truth (SSoT), whenever possible. | While it may be impossible to truly maintain a Single Source of Truth (SSoT) for all data items at all times, especially in the case where the same data entity or instance enters an enterprise through unique data channels, it is an accepted, industry-wide best practice to always work toward such a goal. |

## How to Create a Capability Model

Definition of the Hierarchy:

### Level 1 Enterprise Area

A group of business capabilities and functions which reflects typical high level responsibilities in a company. It corresponds to “Enterprise” which provide a global view of the activities performed by a value System to achieve a particular goal. These Business Functions do not show any boundary and typically span multiple businesses (for instance “make a plane”).

A starting Point is typically the Organization. For Example an Enterprise will have a Sales and Marketing Organization which hast the Responsibility of getting Customer who buying the products of the Enterprise. May be in Addition the Company will have a Supply Chain Organization which includes the Manufacturing of Products, delivering these via the logistics and do the Storage managements. In addition a Company will have Support functions like Finance, Controlling and Human Resources. Therefore we will start on Level 1 with the following Capabilities

Sales, Marketing, Supply Chain, Finance, Controlling and Human Resources.

### [Level 2 Business Areas](http://wiki.scn.sap.com/wiki/display/ModHandbook/Level+1+Business+Areas)

A group of business capabilities and functions which reflects typical high level responsibilities in a company. A Business Function define “what” a business does.

This level corresponds to “Functions in a Business area” which coordinate the actions of employees and the business transactions initiated by systems and business partners within the boundary of a single department to achieve a particular business goal.

In our Example We can now drill into Sales. Sales will have at minimum two capabilities. Sales management and Sales Execution. Whereby The Idea is that we have to Plan some activities in Sales and then execute these. A discussion could be to include capabilities like Self Sales or Cross Channel and or Channel Specific capabilities. This need to be discussed with the stakeholders in which kind of structure they like to implement a capability.

### [Level 3 Business](http://wiki.scn.sap.com/wiki/display/ModHandbook/Level+2+Process+Groups) Department Level

Capabilities grouped within a Business Area belong to the same area of responsibility dealing with similar tasks and activities. These Capabilities are directly related to the Process Level of Value Streams and Value Flows.