Institute of Architecture of Application Systems
University of Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Master's Thesis No. MCS-0003

Resource-centric Modeling of Organizations

Archana Kalidoss

Course of Study: Computer Science M.Sc

Examiner: Prof. Dr. Dr. h. c. Frank Leymann
Supervisor: M.Sc. C. Timurhan Sungur
Commenced: 2nd November 2015
Completed:

CR-Classification:

Abstract

In every field, starting from professional organization to education, human intelligence is one of the most important resource of that field. Despite its importance, it is difficult to systemize them as the human intelligence is something that perceives and acquires differently from human to human. Thus decision making for a similar problem is not going to be the same for every human or for every situations. Informal Process Essentials (IPE) is an approach, proposed to support and automate such unstructured processes. This approach provides neccessary modeling elements to create resource-centric process models. However, an editor to create descriptive models for such processes are still missing.

Every organization thrives to achieve its intentions, these intentions can be in any levels of organization like technical intentions that focus to satisfy technical level requirements, management intentions that focus to satisfy the management level requirements, and financial intentions to achieve financial level requirements. Intentions play critical role in many organizations because they motivate organizations towards the overall development. Therefore supporting and automating organizational intentions and associated components are absolute necessary for any organization. In our context, intentions are realized through strategies, which are associated with organizational capabilities, from which resources are created. As a result IPE models are realized as strategy that is associated with capabilities, resources and intentions. The reason for selecting resource-centric organizational modeling is, a process result may be executed by selecting same set of resources and engaging them towards the intentions of that informal process.

This Master thesis aims at providing means to design and realize the Resource-centric modeling of organizations. We propose a motivating scenario that helps the reader in easily acquiring the concepts and to validate the usability of developed web editor. The purpose of the web editor is to create/view/update intentions, strategies, capabilities and informal process descriptive models.

Key words: Informal Processes, Intentions, Capabilities, Strategies and Resources.

Contents

1 Introduction 11
- 1.1 Motivation . . . 12
- 1.2 Problem Statement . . . 13
- 1.3 Research Objectives . . . 14
- 1.4 Outline . . . 14

2 Fundamentals and Related Work 17
- 2.1 Definitions of Terms . . . 17
- 2.2 Organizational Modeling Notations . . . 19
- 2.3 Entity Types Representation . . . 20
- 2.4 Overview of Informal Process Essentials . . . 22
- 2.5 Human Centric Process . . . 23
- 2.6 Second Phase of InProcXec . . . 25

3 Motivating Scenario 31
- 3.1 Resource-centric Organizational Modeling Example . . . 31
- 3.2 An Abstract View of Entity Types . . . 33

4 Analysis of Resource-centric Organizational Modeling 39
- 4.1 Requirement Analysis . . . 39
- 4.2 Literature Review . . . 41

5 An Approach to Resource-centric Organizational Modeling 47
- 5.1 Overview of the Modeling Process . . . 47
- 5.2 Technologies and Frameworks . . . 48
- 5.3 A Top-Down Modeling Approach . . . 51
- 5.4 Design Methodology . . . 52
- 5.5 Characteristics of the Entity Types . . . 54

6 Case Study on Resource-centric Organizational Modeling 59
- 6.1 Architecture of the Functioning System . . . 59
- 6.2 A Concrete View of Entity Types . . . 62
- 6.3 Realization of Motivating Scenario . . . 62

6.4 Validation 70
7 Conclusion and Future Work 73
Bibliography 75

List of Figures

2.1 Organizational Modeling Entity Types Representation 21
2.2 Steps of InProXec method [SBLW15] 25
2.3 Steps of the Informal Process Modeling 27

3.1 Motivating Scenario 32
3.2 Relation of Strategy with IPE 36

5.1 MVC architecture components 49
5.2 MVC Pattern of adding new entity 50
5.3 Top Down Modeling Approach 51
5.4 User Interface Design of the Editor 53
5.5 Context Intention Relationship 55
5.6 Capabilities Resources Relationship 55
5.7 Acquirable Instances 56
5.8 Acquirable Entities Hierarchy 56

6.1 Architecture of the Functioning System 60
6.2 User interface URL navigation of the functioning system 61
6.3 Realization of Motivating Scenario 63
6.4 Screenshot of Resource Model 68
6.5 Screenshot of Instances(2) 69
6.6 Screenshot of Instances(1) 69

7.1 Contribution of the thesis work 73

List of Tables

1.1 Research Objectives 14

2.1 Organizational Modeling Notations 20

4.1 Requirements Satisfaction Phase 42
4.2 Evaluation of the Approach 45

List of Listings

5.1 XML Schema Definition of Acquirable Entity 57

6.1 XML Schema Definition of Entity Type 64
6.2 XML Schema Definition of Node Type 67
6.3 XML Schema Definition of Instances 70

1 Introduction

Creating a better world requires teamwork, partnerships, and collaboration, as we need an entire army of companies to work together to build a better world within the next few decades. This means corporations must embrace the benefits of cooperating with one another - Simon Mainwaring

Every organization knows the benefits of co-acting a process in order to achieve its desired intention. Resources of an organization play an important role to collaborate and accomplish those tasks. Though organizations re-use data resources and tool resources during this collaboration work, business logics and decisions cannot be reused in certain types of processes. These type of processes are not structured like traditional processes because the process execution steps cannot be pre-defined due to its dynamic nature e.g processes that require involvement of human knowledge in deciding the execution steps[SKL14]. Such type of processes are called *Informal Processes*.

Humans play an important role in informal processes which makes the informal processes collaborative in nature. The participants of an informal processes collaborate to accomplish a task. These participants are the resources that drives towards the accomplishment of the task. Developing an editor to create models for such *resource-centric informal processes* is a part of realizing the automated execution of informal processes. In this document, we explain how we realized developing an editor that creates models for resource-centric informal processes. Along with this we also validate the developed prototype using a case study. This case study has been taken as an example scenario throughout this document that helps for better understanding of the concepts.

In this Chapter, the first section provides a detailed motivational reasons about why this master thesis work is relevant and what about this thesis work is new. The second section contains an overview about the problems in existing approaches and how this approach serves as an *complemenatry approach* to the existing work. The third section discusses about the contributions done in this work and the research objectives satisfied by this approach. The final section provides an overview about the following chapters. In our functioning system each informal process model is a strategy that has capabilities, resources that are created out of capabilities, an intention that has a specific strategy.

1.1 Motivation

Any task today has both well defined, predictable elements and less defined, more ambiguous elements where knowledge workers has power to implement their decisions[1]. For example, research and development projects are of these type where, *what to do next* cannot be decided much in advance. These type of processes are highly unpredictable in nature and this makes it quite challenging to support and automate these type of processes. This master thesis is a part in realizing the automation of such processes. These *unstructured/informal/human-centric processes* are called as *informal processes*[SKL14]. Any approach that supports informal process is required to be more autonomous because of their dynamic behavior of enacting a process, so the existing approaches available for traditional processes are not helpful in realizing the execution of informal processes.

Though the execution steps of informal processes cannot be determined beforehand, *intentions* of informal processes are known before their enactment [SBLW15]. Achieving these intentions requires another important driving force *resources*, which can be anything from human actors, development environment, material resources etc. These resources has to posses certain *capabilities* to qualify for executing the process. So we need an approach that supports informal processes along with the support of intentions, required resources, capabilities etc. This can be achieved by associating intentions with strategies, strategies with capabilities and informal processes with resources. In extension to the work of Sungur et. al. [SBBL14] where a descriptive model and an overview of solution architecture has been presented, this work focuses to provide web based editor to create resource-centric models of organizations. The reason for selecting meta-model approach is to preserve the essential information associated with informal processes such as intentions, context information, resource definitions etc. This work also provides means to initialize and acquire instances which can be further extended during enactment of resource-centric informal processes.

The developed editor serves as an *descriptive* web based editor tool, where the business experts can create models for informal processes, intentions, strategies, capabilities etc and do not comprise any functionality for acquiring the runnables. Instead this editor provides facility to plug-in the functionality for transforming the descriptive information about the entities into deployable information.

[1] White, Michael. "Case management: Combining knowledge with process." BPTrends, July (2009).

1.2 Problem Statement

Every organization contains multiple entities like resources e.g humans, tools etc., intentions e.g revenue based intentions,quarterly intentions etc., strategies e.g the process to achieve the intention and capabilities e.g a resource that can provide a particular capability. Thus an organization needs efficient mechanisms to handle and manage these different types of entities. Though there are some existing tools which provide facility to manage resources in an organizations, they act either as a *Retrieval Service* i.e they are used only to view or retrieve resource engagers for resources or as an *Control Service* i.e they are used to run resource engagers [SBLW15]. But there is not a service which provides both mechanism to retrieve and initiate the instances of each entities.

Informal processes are collaborative in nature, which means that participants of informal process collaborate with each other to accomplish its intentions[SBLW15]. Designing these collaborations and assigning participants their respective privileges, plays an important role during execution of the respective informal processes. The research work by Matthews et. al [MWMY11] clearly points out below as the major problems in adopting to a workspace collaboration tools.

1. Lack of Methods
2. Methods that focus on individuals
3. Not well targeted groups
4. Not well supported editors for executing abstract descriptions

Though there are *activity-centric* modeling and reusing of business processes such as Business Process Execution Language (BPEL) [2] and Business Process Model and Notation (BPMN) [3] are available, they are not suitable for certain type processes whose execution steps cannot be predicted in advance [SBBL14]. Also complementary concepts such as automatic initialization and acquiring of interrelated resources are still missing in the existing work [SBLW15]. Another key thing to remember is informal processes are volatile in nature which is one of the important reason for challenges in developing an environment that supports informal processes.

[2] http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf
[3] http://www.omg.org/spec/BPMN/2.0/PDF/

1.3 Research Objectives

The main focus of this master thesis, is to realize the phase *Informal Process Modeling* (P2) described in approach *Executing Informal Processes* (InProXec) mentioned by Sungur et al. [SBLW15]. Coupled with the main focus, of developing web based editor the following requirements which are research objectives of this work, has been provided in the Table 1.1 are also satisfied by the developed prototype.

Requirements	**Description**
R1	*Organizational intentions transparency*
R2	*Organizational intention resource-based cost estimation*
R3	*Organizational intention achievability estimation*
R4	*Intention oriented working style*
R5	*Participative organizational modeling*
R6	*Re-use of organizational knowledge*

Table 1.1: Research Objectives

1.4 Outline

The remainder of this document is organized into following chapters:

Chapter 2 – Fundamentals and Related Work: In this chapter, basic fundamental concepts that are essential to understand this thesis work has been described.

Chapter 3 – Motivating Scenario: In this chapter, a motivating scenario has been taken and detailed explanation of each phases of the scenario has been provided. This aids reader to understand clearly the concepts of organizational modeling throughout the document.

Chapter 4 – Analysis of Resource-centric Organizational Modeling: This chapter provides detailed requirement analysis based on scientific facts published in existing work. This chapter also provides literature review of existing work.

Chapter 5 – An Approach to Resource-centric Organizational Modeling: This chapter discusses about the methodology followed in realizing the concepts of resource-centric organizational modeling has been provided.

Chapter 6 – Case Study on Resource-centric Organizational Modeling: This chapter validates the approach presented in Chapter 5. This chapter also discusses detailed system architecture and also presents the experimental results. The abstract concepts motivating scenario discussed in 3 has been explained in a concrete way.

Chapter 7 – Conclusion and Future Work: This chapter summarizes the results of this thesis work and draws conclusion. This chapter also throws some light on the future work to be carried out in the field of organizational modeling.

2 Fundamentals and Related Work

This chapter provides the fundamental concepts and related work that are required to understand the approach to be discussed in following Chapter 5. The first section introduces definitions of terms that are used throughout this document. The second section provides a brief introduction about the basic concepts of this thesis work. The third and fourth sections present description about works related to the presented approach. The last section focuses on concepts of Resource-centric Organizational Modeling and its notations. Though modeling notations are not part of implementation, it has been introduced to assist the reader in better understanding. The last section also discusses about the process and entity representation of the organizational modeling.

2.1 Definitions of Terms

In this section, the definitions of terminologies that are used throughout this document are provided.

Business Process - A business process has been defined as the set of activities and tasks whose final output is accomplishment of a goal. These activities are performed in an organizational and technical environment [Wes12]. Based on the type of input and the operation of tasks, these processes can be categorized as management process, support process, research process, development process, etc., [SBLW15].

Business Logic - Business logic refers to the activities that need to be done to execute the corresponding process.

Business Process Models - Business process models are models to capture recurring procedures during a business process execution and enact them in a automated fashion for re-using those stored knowledge. A model can be in any form of representation such as graphical, descriptive etc. In our context, model refers to descriptive information of a process or an entity.

Business Process Model and Notation (BPMN) - BPMN is the standard notation used for business process modeling [Rec10] and business experts model their business processes mostly using BPMN [SAP09]. Models developed using BPMN can also be executed using

BPMN engines. BPMN bridges the gap between developers and business experts. BPMN uses graphical representation to model the business processes.

Business Process Engines - Business process engines can enact the business process models automatically once the configuration of necessary infrastructure has been carried out. Also in the research article [Ley10], it has been mentioned that BPMN users wanted to execute the models with BPMN2.0 engine due to its operational semantics.

Business Process Management - Business process management (BPM) includes concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes [Wes12].

Business Process Management Life Cycle - Business process management life cycle is the series of phases such as modeling, configuring, executing, and improving business process. These series phases are conducted as a cycle [Wes12].

Informal Process - The processes that human participate and create knowledge are called unstructured/informal/human-centric processes. In informal process, execution steps cannot be modeled or are not feasible to model before their enactments. This is because due to the dynamic changing behavior of execution steps of the informal processes. For Example, software development process is an informal process, where required activities and order of their execution cannot be determined beforehand [SBLW15]. The four characteristic properties are: implicit business logic, varying relationships among resources, resource participation in multiple informal processes and changing resources [SBBL14].

Informal Process Essentials - Informal Process Essentials (IPE) is an intention-based approach that enables describing process declartively, i.e., without describing how the intention is achieved, and providing only information about what has to be achieved [SBBL14].

OASIS Topology and Orchestration Specification for Cloud Applications(TOSCA) - TOSCA is a new OASIS(Organization for the Advancement of Structured Information Standards) standard to describe composite applications and their management [KBBL13].

Winery - Winery is a modeling tool offering an HTML5-based environment for graph-based modeling of application topologies and defining reusable component and relationship types. It uses TOSCA as an internal storage, import, and export format [KBBL13].

Intention - Intentions are defined hierarchically, which can contain and extend sub-intentions.It is depicted by a double circle in organizational notations. The sub-intentions are refined starting from main intentions. Intentions are associated with capabilities or resources. An accomplishment of an intention changes state. An intention can extend another intention.

Resource - A resource can be a people or tool those/that drive towards the successful execution of the process. It is key for achieving specified process intentions. In the context of our work, the definition of organizational resources refers not only the entities that are capable of doing work but also entities that have an impact on the outcome of the processes, e.g., software tools, human performers, data etc.

Capability - A capability is the ability to provide business values like software applications, resources, and potential of the actor to make decisions even in changing situations [SGHZ12].Describes a capability provided by a resource or required by an intention. The performers of an informal process have certain skills and roles to achieve the intention.

Strategy - A strategy is a method or plan chosen to bring desired results, such as achievement of an intention or solution to a problem [1].

ENntity - Throughout our document we use the term *entity*, which denotes every individual model e.g., an intention model, strategy model, capability model etc. Entity type means the type which an entity belongs to e.g., intentions, strategies, capabilities etc. For example, when we call an intention definition as an entity then entity type of this entity is intention.

2.2 Organizational Modeling Notations

The Organizational Modeling element notation has been selected as per the guidelines mentioned in the literature [Moo09] and these notations are taken from the thesis work by the author Sierra[Sie15]. Though these notations modeling are not part of this master thesis, this has been mentioned in this section for the sole purpose of aiding the reader to understand the concepts much better through pictorial representations. Also by observing the fact that business process modelers are already well-known with the present process modeling notations such as Business Process Modeling Notation 2.0 (BPMN) [Gro11] and ArchiMate notation[Gro13], the shape depiction of organizational model elements has been designed in the previous work [Sie15] similar to those existing process notations.

Due to the importance of shapes in expressing the information visually , the notations are chosen in such a way that each element of Organizational Modeling differ by shape. Also a legend will be always shown in the modeling notation to denote the meaning of each shape [Moo09]. The description of each element in the Organizational Model Notation is shown in the Table 2.1.

[1] http://www.businessdictionary.com/definition/strategy.html

Element	Definition	Notation
Intentions	Intentions are purposeful concrete steps taken by organizations or individuals to achieve an expected outcome.	
Capabilities	Capability is an ability that should be possessed by a resource that work towards achievement of one or several intentions.	
Context	The environment that forms the setting for an event, statement, or idea, and in terms of which it can be fully understood. There are two Contexts: Initial and Final. The Initial Context is the situation which describes the driving forces that trigger the process to start. The Final Context is the expected situation once the process has finished. Both initial and final context are represented by an hexagonal shape except the final context has thick edges than initial context.	
Strategy	A method or plan chosen to bring about a desired future, such as accomplishment of an intention.	
Resources	The people or tools those/that needed to fulfill the middle objectives or work towards the achievement of intentions .	
Relationship	A relationship is used specify the fixed links between the elements of the model.	

Table 2.1: Organizational Modeling Notations

2.3 Entity Types Representation

The conceptual model of entity types in organizational modeling are shown in the Figure 2.1. This model shows that among all the entity types, intentions are in the top level of hierarchy which can be further divided into *sub-intentions* and/or *strategies*.

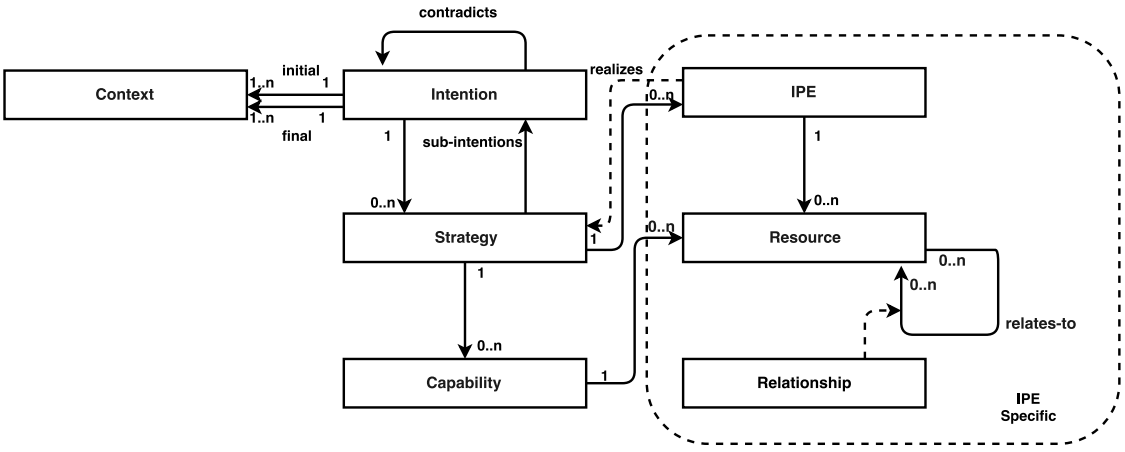

Figure 2.1: Organizational Modeling Entity Types Representation

An intention can either contradict or be a sub intention of another intention. These type of sub-intention and contradicting intention has been explained in detail with a suitable example in Section 3.2.2. An intention can be achieved through a strategy, which is a plan of action designed to meet the intention. An intention can be achieved through none or many strategies. Strategies also describe none or many capabilities and processes required to achieve intention. The capabilities and processes can be further resolved into resources or resource models. Thus starting from defining intentions, we define strategies then required capabilities and process models. The capabilities and process models define the required resources.

As reported by Sungur et al. [SBBL14], the concept of IPE provides an agent-based approach i.e., human performers are considered as agents who execute the processes autonomously. Organizational Process Modeling is *Resource-centric* approach as they support processes by providing required resources and thrives to successfully execute the processes by using qualified autonomous agents, i.e., actors under certain *context definitions*. As we mentioned before, in our context resources can be anything like people, IT tools, data that are used to accomplish the objectives. Emerging intentions can result in the requirement of new capabilities, i.e., resources. Resource models are also provided in the developed prototype to make precise definitions of resources needed.

In Sungur et al [SBBL14] work, the concept of *Informal Process Support Model* (IPSM) has been introduced which is to make use of existing knowledge of human performers. Here the initial creator of the model is experienced human performers. Based on their experience, they add relevant resources of an informal process. The models are generated at runtime based on the interactions and activities of corresponding human performers. An informal process targets for accomplishment of an intention. The

intentions can be refined by defining sub-intentions and/or strategies, which can then be further refined recursively as independent informal processes. The intention-based approach enables describing processes declaratively, i.e., without describing *how* the intention is achieved, and providing only information about *what* is achieved. As the author [SBBL14] suggests that this avoids need of predefined business logic in the representations of informal processes. Each resource can be related to another resource in the context of an informal process using predefined or custom *Relationships*. Informal Process Essentials are realized through strategies. Each informal process starts from an initial context, i.e., *initial context* and aims to achieve an intention. After accomplishing the intention, there is a resulting context called as *final context*. The beginning state before achieving intention is called as initial context and the end state after achieving intention is called as final context. On completion of intention execution, the process state changes from one state to another.

2.4 Overview of Informal Process Essentials

In this section, we provide an overview about the concepts introduced in the approach Informal Process Essentials (IPE) [SBBL14]. Models are used in various fields like manufacturing, scientific, IT, etc. These models are mainly useful in re-using the predefined regular, intelligible and field-tested solutions. Such models has numerous benefits [2] like performance improvement, reduced cost of operation and design, etc. Besides these processes there are processes which requires participation of human and performance of these processes depend on human knowledge, i.e., they are subject to change and carried out based on experience of previous knowledge. These processes are called *Informal Processes* and they do not have formal structured execution of steps for the enactment of processes.

The work by Sungur et. al [SBBL14] gives a comprehensive account of challenges in defining the business logic of informal processes as below:

- The structure of informal processes are not known before enactment of the processes
- Results in less flexible and less efficient solutions
- The cost of creation of well-defined business logic is too high

[2] http://www.nomagic.com/getting-started/modeling-benefits.html

This thesis work realizes the concept of *resource-centric modeling of informal processes*, specified in the Informal Process Essentials(IPE) approach by Sungur et al [SBBL14]. As mentioned in the Section 2.1, resources are drivers to achieve intentions in the informal processes. In this approach author states that, when the desired process result is repeated the same set of resources can be selected and engaged towards collective intention of the informal processes. Also it has been mentioned that each IPE model contains the list of necessary resources to accomplish the main intention of the respective informal process.

In IPE approach, the author differentiates the resources based on the time. The resources that are needed in the informal processes are below :

- *Initial resources* which are required during the start of informal processes.
- *On-demand resources* that are required based on intentions during process enactment.
- *Actors* are the resources in IPE meta-model, that drive process execution autonomously.
- *Knowledge resources* resources that contain important information required for the enactment of a process. These are critical for guiding actors.

It has been mentioned in the approach [SKL14] that Informal Process Essentials (IPE) describes the following about informal process, 1. Describes the constituents informal process such as performers, data and software tools 2. Describes how to make core element ready for the enactment of the informal process i.e resource providers. Predefining business logic would results in higher cost compared to making decisions by human performers during enactment [SKL14]. Sometimes, a process team may require participation of new resources with different roles and relationships from a different team [MWMY11; MWM+12]. For example, in our motivating scenario we have two teams software development team and help desk team. To improve the user feedback portal, help desk team may require resources from software development team with a role of user interface web developer. Thus to satisfy requirement changes, resources are also changeable during process execution.

2.5 Human Centric Process

The role of humans in organizations has been evolving over time. The shift from "personnel" to "human resources" acknowledges the importance of humans as organizational

resources. There are incredible number of pressure on today's organizations [3] due to varying dynamic nature of organizations. For example, organizational changes like addition of new organizational alliances, new structures and hierarchies, new ways of assigning work, and a very high rate of changes like changes in the workforce, including employees' priorities, capabilities, and demographic characteristics. Thus it is impossible to do one hundred percent perfect forecasting of dynamically changing activities or processes in an organization.

In order to manage such a dynamic environment, organizations need skilled human resources with previous knowledge of handling unforeseen scenarios. Thus human resources are vital part of any organizations as they have skills of acute future orientation to understand changing organizational environment. Humans in an organizations carry out many important activities. Managers and Human Resource (HR) professionals organizes jobs of each and every human in the organization so that they can effectively perform these jobs. Thus humans in any organization are viewed as resources of the organization which is a contemporary part of Human Resource Management [4].

When there are multiple human resources working for a process, then there should be some sort of co-ordination and understanding between the humans which is called *collaboration* at an organizational level. Collaboration exists in every levels of an organization. For example at management levels of an organization, managers and HR professionals work together to assign employees their roles and task in the organization. This helps the employees of the organization to adapt to its environment. In a flexible organization, employees roles and responsiblities changes dynamically based on the requirements and business priorities. Thus the need for network of representations between the human resources which sets up an environment to support collaborative work of business related process has been realized in the work by the author Canko[Can15]. The concept of *virtual human representation* is an extension of actor-concept described in *Informal Process Essentials* [SBBL14]. The developed prototype *Human Resource Representation* in the work by the author Canko[Can15] saves the information such as capabilities, roles, responsibilities etc. as a virtual human web ontology instance which can be re-used in web based environments.

These kind of human representation are highly helpful to organizations with dynamically changing processes. These representations can describe and match resources with their capabilities based on the requirements. As we have mentioned in Chapter 1, in our context of resource-centric modeling humans are also considered as resources and we associate *capabilities* with every resources. Moreover, associating capabilities, with resources is helpful in following situation. There can be a situation where resources

[3] http://www.siop.org/tip/backissues/tipjan98/may.aspx
[4] http://smallbusiness.chron.com/role-human-resource-management-organizations-21077.html

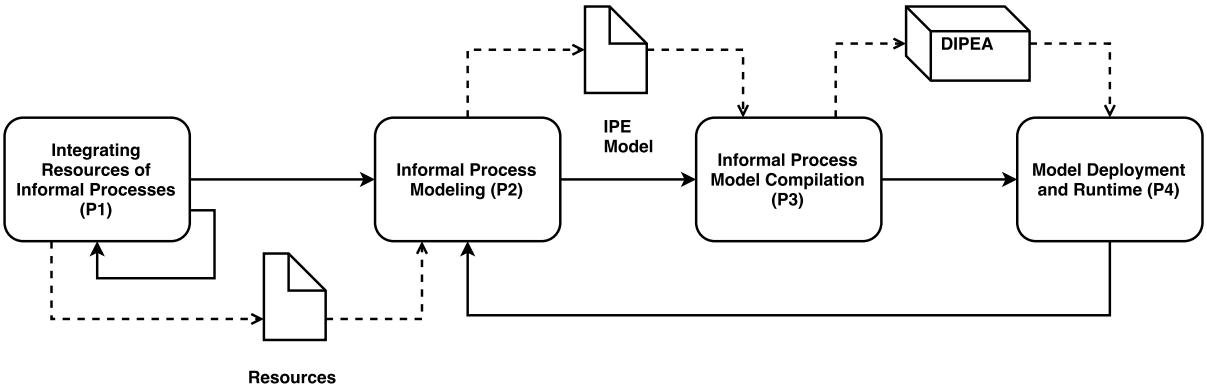

Figure 2.2: Steps of InProXec method [SBLW15]

producing more accurate results for a processing task are preferred than resources which can produce higher throughput for a processing task. Thus we need to associate capabilities with each resources and need to automate the process of discovering and matching the resources with their capabilities based on their process.

2.6 Second Phase of InProcXec

In this section, we present an overview about the *Executing Informal Processes* (InProXec) method, proposed by Sungur et al. [SBLW15]. Since this thesis work is realizing resource-centric modeling of organizations, the main focus of this section is on the second phase of InProXec which is *Informal Process Modeling*(P2). The method described in Figure 2.2, initializes informal process models in an automated fashion. In the following paragraphs, we present short overview about different phases of the InProXec method which has been decribed in detail in the article [SBLW15] and with a detailed description about the second phase of the *InProXec* method.

As shown in the Figure 2.2, the InProcXec method consists of four different phases:

Integrating Resources of Informal Processes (P1) - In order to model an informal process, we need information about the resources, these information are collected beforehand during process execution. There exist many services to acquire information about informal processes resources automatically. The final output of this phase is integrated resources which are required as an input to next modeling phase. Thus this phase sets up an environment required for modeling and execution of informal processes.

Informal Process Modeling (P2) - This phase receives the resource definitions made available in the first phase P1 as an input. Based on this, business experts model informal processes and associated entities like strategies, intentions, capabilities etc.,

using our developed web editor. This phase has been explained in detail in the following sub section 2.6.1

Informal Process Compilation (P3) - The previous phase P2, describes only the intentions required to be achieved, corresponding required resources etc. But in phase P2, the functionality to instantiate acquirable entities are not included. Thus in third phase P3, the output of phase P2 is taken i.e IPE models and are transformed into intializable self-contained *Deployable Informal Process Essentials Archives(DIPEA)* [SBLW15] takes place. This results in DIPEAs enacting required informal process. To realize, phase P3 an *IPE Model Compiler* also been introduced.

Informal Process Model Deployment and Runtime (P4) This phase employs *IPE Runtime* which parses DIPEAs and runs the executables contained in those archives. During this phase, the autonomous actors work towards intentions of informal processes using acquired resources and other involved resources.

2.6.1 Informal Process Modeling (P2)

This approach of Informal Process Modeling is directed towards modeling the informal process based on their intentions rather than their activities. The developed prototype serves as an holistic web based editor to create, view and update all the associated entities of informal process like intentions, capabilities, strategies etc., along with informal process. Also from our detailed explanation in previous sections about the importance of resources in organizational modeling and along with the fact that phase P2 receives resource defintions as input from phase P1 of InProXec method we can apprehend that resource definitions are the lowest level in the hierarchy of resource-centric organizational modeling approach. The sequence of steps to be carried out using the developed editor has been shown in the Figure 2.3. It is important to note that in the figure, only solid round edged rectangles are part of the developed editor. The tasks to be carried out in each of the steps in developed editor is described as below:

Model Context Definitions (M1)

The first step is to model context definitions, where we can model both basic properties like name and namespace of a context definition and entity specific properties like contained contexts, entity definitions etc., of a context definition.

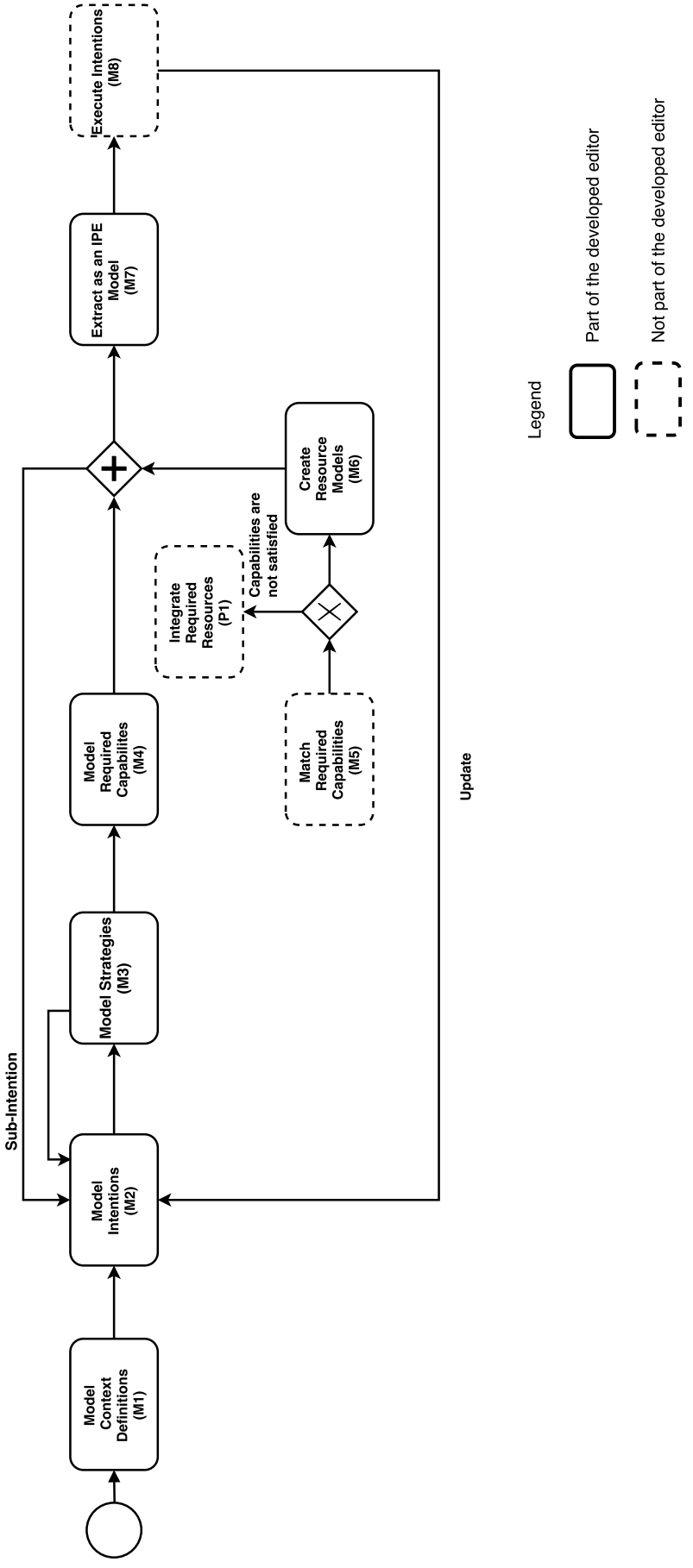

Figure 2.3: Steps of the Informal Process Modeling

Model Intentions (M2)

Similar to context definition modeling (M1), the second step (M2) is to model intentions. The context definitions created in step M1 can be used to specify initial and final contexts of an intention. Intentions can contain sub-intentions and contradicting intentions. These type of sub intentions and contradicting intentions are also modeled as intentions in this step and their type of relation to specific intention are mentioned.

Model Strategies (M3)

Once intentions are identified and modeled, the third step is modeling of strategy to achieve a specific intention. As mentioned earlier in Section 2.3, an intention can have multiple strategies.

Model Required Capabilities (M4)

After modeling of strategies, capabilities required to achieve an intention in a specific strategy is modeled. A strategy can require multiple capabilities which has been explained in detail with a suitable example in the following Chapter 3.

Create Resource Models (M6)

After matching the resources and capabilities i.e after finding the correct resource that has the capability to carry out the process, the resource models are created. The need for modeling a new intention may arise in parallel this has been explained with a suitable example in the following Chapter 3.

Extract as an IPE Model (M7)

After the completion of above mentioned steps, the modeled entities can be extracted as an IPE model which can be reused.

The other steps denoted in dashed round edged rectangle which are not part of developed web editor includes matching of required organizational capabilities that are satisfied by resource models to enable the achievement of organizational intentions in certain context definitions through a strategy. If there is no suitable matching capability then phase P1 of InProXec can be carried out again until a matching capability is found. If Capabilites are satisfied resource models can be created. The created resource models(M6) along with

modeled capabilities can be extracted as an IPE Model(M7) which will be provided as input for the next step execution of intentions (M8). After the execution of an intention, the status of an intention is updated inside the specific intentions's property.

3 Motivating Scenario

In order to help in understanding the concepts of organizational modeling, a motivating scenario has been taken and explained through the notations mentioned in Section 2.2. This scenario also helps in validating the developed web editor in the following Chapter 6. The motivating scenario has been chosen based on the collected real life scenarios provided in the thesis work of the author Sierra[Sie15]. The scenario of this example was taken from the context of manufacturing sector.

In this chapter, the first section provides an overview about the motivating scenario followed by second section that explains in detail about the motivating scenario. The third section provides an abstract overview about the entity types in the motivating scenario which will be explained in a concrete way, in the following Chapter 6.

3.1 Resource-centric Organizational Modeling Example

The concept of resource centric organizational modeling can be explained with the following scenario in a manufacturing organization. Consider a budding manufacturing company which designs, develops, manufactures and sells personal computers, tablets and laptops. The CEO's main intention of the quarter is to increase the revenue and number of unit sales. The initial context describes the situation before starting the execution of intention. The initial context also provides description that motivates to start the process. The final context describes the situation that is achieved once the intention executed successfully. Intentions connect initial context definitions with final context definitions [SBBL14]. The sub-intentions are the intermediate intentions which describes the expected outcome in a measurable form. Intentions reach strategy implementations through achieving strategies which are plans of action designed to meet a specific intention.

The example scenario follows top-down approach of organizational modeling i.e.,organization's higher level intentions can be achieved by amalgamation of specific, measurable and realistic sub-intentions, strategies etc. The figure3.1 provides the details of intention and its associated strategies, sub-intentions. There can be multiple strategies followed to achieve an intention. In this type of process modeling, strategies

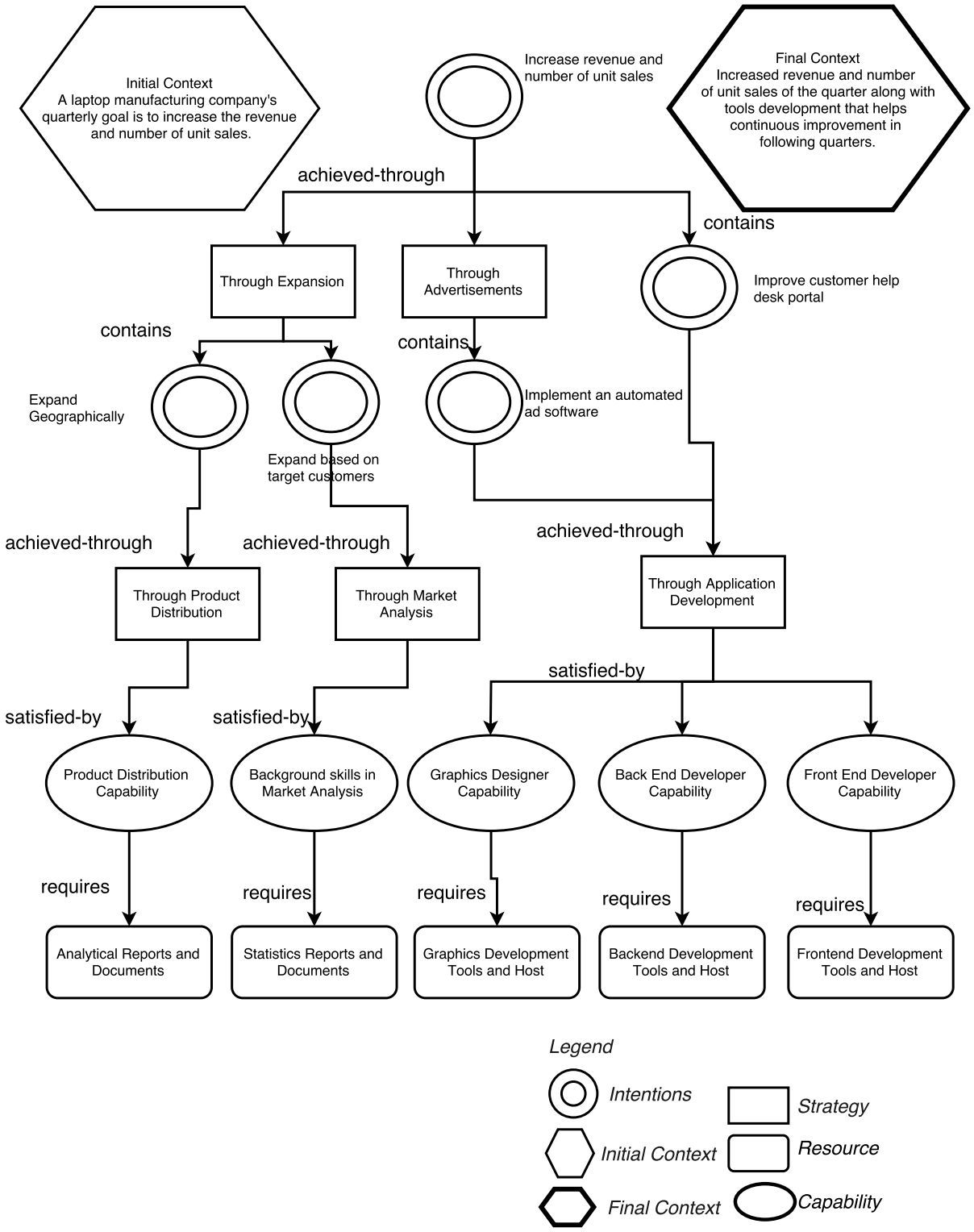

Figure 3.1: Motivating Scenario

are self-contained and loosely coupled. This is the reason when we extract only the strategies from organizational process modeling it would be similar to informal process essential modeling.

In order to achieve this main intention through our organizational modeling approach, as a first step we need to break the intention into concrete levels like strategies, sub-intentions, process definitions, resource definitions etc. This intention can be achieved by following all of the below mentioned strategies and sub-intentions, which requires resources with matching capabilities.

1. Increasing the revenue through expanding the market sales.
2. Through increasing the advertisement which helps in customer to know about the product.
3. Improving the existing customer help desk portal, as it helps to maintain good customer relationship.

3.2 An Abstract View of Entity Types

This section discusses in details, about each entity types of the motivating scenario in an abstract way, which is further detailed using concrete steps in the following Chapter 6. The participating resources work towards one *main intention* and certain *sub-intentions*. Sub-intentions are part of main intention, which helps the resources to modularize and achieve the main intention. Also each sub-intention has certain type of relationship with main-intention. For example in our below described motivating scenario in Section 3.1 one of the sub-intention is to *expand sales geographically*. Before executing this sub-intention, few ground works like collection of laptop usage statistics such as average buying capacity of the consumers, average computer knowledge in the new area has to be done. Thus the execution of main intention i.e *increase revenue and number of unit sales*, requires collaboration of people with different skills and expertise. People who has skills to collect and study statistics can serve as external resources. As new intentions may emerge dynamically the team working towards the achievement of main intention should also be ready to accommodate new resources with new capabilities and skills. For example, there is a software development team, which work towards achievement of one of the sub-intention *improve help desk portal*, i.e this team develops software that automatically attends and records user queries. The management of the project is done through the support of project management software called Redmine [1]. The

[1] http://www.redmine.org/

participating human resources are members of business oriented social network called XING [2].

3.2.1 Contexts

The execution of manufacturing processes such as the one provided in Figure 3.1, are not similar to execution of typical business processes. This is because, the execution of manufacturing processes mostly depends on the information collected from the real world, i.e., the execution context [SBLW16]. A context definition provides a mechanism to act adaptively based on the current situation in the production environment by describing each process for a specific context definition [SBLW16]. For example, in our motivating scenario the initial context provides details about status before achievement of the main intention i.e it specifies the situation of the organization which triggers the execution of main-intention. The initial context *quarterly goal of increasing the revenue and number of unit sales*, helps to decide the main intention and its related low level associates. On successful achievement of main-intention the organization reaches it desired final context of *increased revenue and number of unit sales*. This also provides tools such as web-based help desk portals, automated ad software etc., that are developed as part of this contexts can also be re-used in future executions. When one of the final context definitions has been reached the process completion starts. Models that contain these knowledge are also stored and re-used during next enactments. Final state of the model instance is also saved on successful reaching of the final context[3].

3.2.2 Intentions

Intentions are defined hierarchically, in our approach intentions are in top level of the hierarchy, which are refined until lower level of the hierarchy is reached. In our context, intentions are not associated with capabilities, instead intentions are associated with strategies which are then associated with capabilities and processes. Thus a strategy realizes informal processes. For example, in our motivating scenario the main intention is to increase revenue and number of unit sales which also has sub-intention of improving the customer help desk portal and strategies such as 1. through expanding sales and 2. through advertisements. The relation between strategies and intentions are denoted by the term *achieved-through* in Figure 3.1 as strategies are methods through which intentions can be achieved. The relation between an intention ans its sub-intentions are

[2] http://www.xing.com/

[3] C.Timurhan Sungur, An Approach to Supporting and Automating Informal Processes, May 2015.

denoted as *contains* as intentions can contain and contradict themselves. For example in our motivating scenario IT system managers are not willing to give up the systems they are working for a long time even if it is a better solution for organization as whole. This real life example scenario has been provided in the thesis work [Sie15] and also it has been suggested that such contradicting intentions has to be handled in some way. Thus our developed web editor has provision to associate both sub-intentions and contradicting intentions for any intention. There are also sub-intentions that emerge through strategies which are also denoted by the term *contains*. For example, in our motivating scenario, one of the strategy to increase the revenue and number of unit sales is through expanding the sales which further has sub-intentions such as *expand geographically* and *expand based on target customers*.

3.2.3 Strategies

Strategies are used to identify the most appropriate method of utilizing the capabilities through which an intention can be achieved. Strategies are associated with both intentions and capabilities. Capabilities are related to resources. Each strategy needs certain capability to successfully execute an intention. Resources are the potential holder of the capability i.e., to satisfy a capability we need resources. The capability and its associated resources are also shown in the Figure 3.1. In our motivating scenario, the main intention can be achieved through two strategies *through expansion* and *through advertisements*. These two strategy further contains intentions such as *expand geographically*, *expand based on target customers* and *implement an automated ad software*. Since strategies contain intentions they are related through the term *contains* in the Figure 3.1.

As mentioned before in the Chapter 2, informal processes are realized through strategies. This achieved through strategy containing capabilities and process models. For example, consider a small part in our motivating scenario, where to achieve an intention expand geographically, through strategy product distribution we need resources with *product distribution capability*. This results in informal process as a strategy that has capabilities, resources that are created out of capabilities and an intention for that specific strategy which has been shown in the Figure .

3.2.4 Capabilities

Resources tends to posses certain capabilities that allow them to do something that they want or need to do something. Each organizational capability must be provided by a

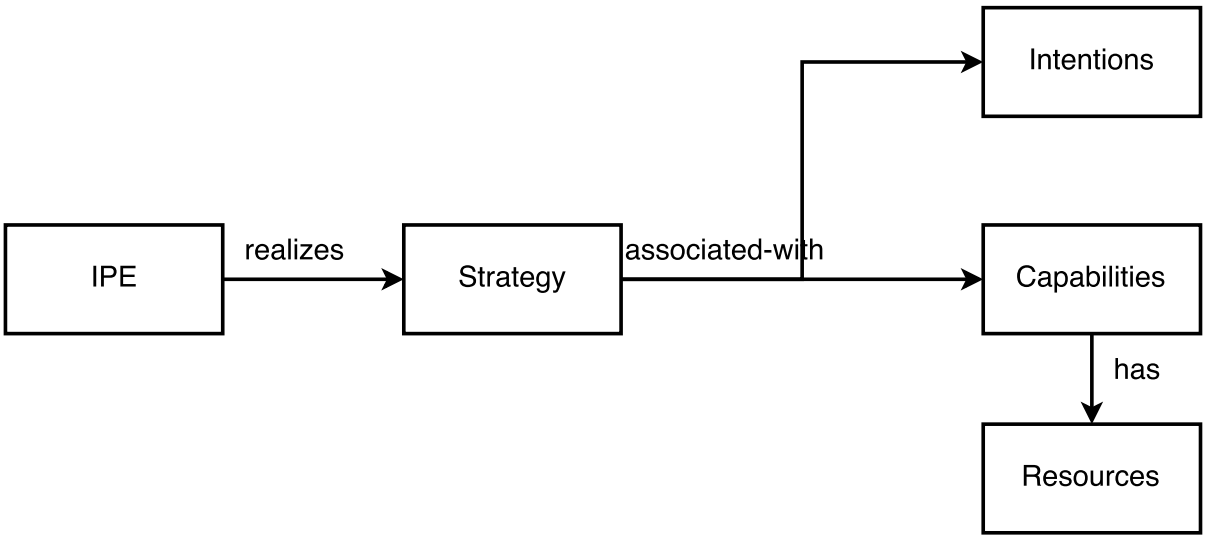

Figure 3.2: Relation of Strategy with IPE

resource in the organization. Resource models are optional [4] to make precise definitions of resources needed. In our context, capabilities that are associated with resources are called as *functional capabilities*. The type of capability that contains functional capabilities are called as *cross functional capabilities*. Strategies are associated with cross functional capabilities, which contains functional capabilities out of which resources are created. In our motivating scenario to achieve the main intention, we need several capabilities such as product distribution capability, graphics designer capability etc. Thus in the Figure 3.1, strategies and associated capabilities are related through the term *satisfied-by*.

3.2.5 Resources

Each resources has different types of relationship with other resources based on how they communicate with other resources [SBLW15]. For example in our motivating scenario described in Section 3.1 has sub-intention of *improve customer help desk portal*. This sub-intention can be achieved by providing skills improvement training to the employees or by recuriting newly skilled employee. Here the manager has permissions to decide whether to improve skills of existing employee or recruit new employee. But the team lead has restricted permission like what type of skills are required for the project based on decision of manager. The Informal Process Essentials (IPE) approach proposed by Sungur et al. [SBLW15], paves the way to create models with definitions of key actors

[4] C.Timurhan Sungur, An Approach to Supporting and Automating Informal Processes

e.g manager, team lead and definitions of suppoting resources such as Mediawiki [5]. A *resource organizer* is responsible for gathering definitions about the resources which are required by business experts for modeling [SBBL14].

[5] http://www.mediawiki.org/

4 Analysis of Resource-centric Organizational Modeling

This chapter positions the thesis work with respect to the other existing approach. The first section provides detailed requirement analysis about the research objectives described in Chapter 1. The final section provides a detailed literature review about the existing approaches. This literature review is used to evaluate the proposed approach in the following Chapter 5.

4.1 Requirement Analysis

This section provides a detailed requirement analysis of research objectives mentioned in Chapter 1. The below mentioned requirements are part of the functioning system proposed in the following Chapter 5. Also the exact phase when each of the requirements get satisfied are provided in Table 4.1.

4.1.1 Organizational Intention Transparency (R1)

An intention can be broken down into definitive actionable components, or sub-intentions, upon which individual resources can act. When these lower level sub-intentions are made achievable for individual resources, they can be combined to provide successful execution of higher level intention. Different organizational members can observe lower level and higher level intentions in their organizations. Intentions are traceable in the different levels of the organizational hierarchy. This means that the status of each intention can be accessed by members in different levels of the organizations. This kind of transparency within an organization reduces inefficiencies in intention execution, and is a key factor in attracting and retaining high performers in the labor market [MHL+07]. Requirement R1 has to be satisfied in the modeling phase itself as the designing of intentions, sub-intentions, strategies are done during the modeling time.

4.1.2 Organizational Intention Resource-based Cost Estimation (R2)

Linking intentions with strategies enable us a cost estimation for each intention. This is because intentions are realized through some strategies, strategies are associated with organizational capabilities which in turn has been associated with organizational resources. Cost is estimated in a recursive manner which has been explained in detail with an example in the following Chapter 5. To incorporate the cost estimation of intentions, we have to understand the recursive structure of the intentions associated with strategies. Since intentions are defined hierarchically, they can contain and extend intentions. Here strategy represents a means for achieving the intention. Further on, the cost of a strategy can be analyzed using the costs of derived sub-intentions, process definitions and so on. Including resources cost in intention cost calculation is important. This is achieved by associating resource models' cost with process models' cost. The recursion is stopped when the intention derivation process reaches the operational level. At the moment a intention is achieved, some resources should be allocated to maintain the desired state (intention maintenance costs)[MBH+10]. Allocation of resources is mainly done at the operational level,hence requirement R2 has to be satisfied during the deployment phase.

4.1.3 Organizational Intention Achievability Estimation (R3)

The sub-intentions are projections of their super intentions, and satisfaction of the sub-intentions ensures satisfaction of the super intentions. Hence validity of an organizational intention is achievable when the intentions can be refined by defining sub-intentions, which can then be defined recursively as strategy and then to independent informal process models. Lower-level requirements can be validated against higher-level intentions, thus enabling validation of strategic alignment of higher level intentions. The objectives of business strategy are found in the highest levels of the intention model.[BCV06].Requirement R3 can be found during the modeling time of the process models as intention achieveability estimations are done before starting the execution of the intention based on the related intentions.

4.1.4 Intention Oriented Working Style (R4)

As each member of the organization is aware of the higher level and lower level intentions and he can engage for these explicit intentions. Intention orientation is the degree to which a person or organization focuses on tasks and the end results of those tasks. Strong intention orientation advocates a focus on the ends that the tasks are

made for instead of the tasks themselves and how those ends will affect either the person or the entire company. Those with strong intention orientation will be able to accurately judge the effects of reaching the intention as well as the ability to fulfill that particular intention with current resources and skills [Lac16]. The distinction between explicit knowledge of each sub intentions should not be seen as a division but rather as a continuum which aligns towards achieving the higher level intention . Though Requirement R4 itself has sub-requirement of R1, R4 has to be done at the run time which makes it distinct from the Requirement 1.

4.1.5 Participative Organizational Modeling (R5)

Different members of an organization participate to create organizational intentions, as a result intentions are shaped based on all members but directed by the executives. The social extension of a business process can be regarded as a process optimization phase, where the organization seeks efficiency by extending the reach of a business process to a broader class of stakeholders[BFV12].Requirement R5 would be done at the run time as the input from different members of the organization provided during the process execution. But the list of participants who can have the priviliges such as own/edit/follow/view access to the models can be determined beforehand.

4.1.6 Re-use of Organizational Knowledge (R6)

Intentions specific solutions can be extracted as abstract re-usable entities, organizational strategy patterns and can be re-used in multiple context definitions. These field tested solutions are made as descriptive model informations which can be re-used. Re-using the informations as models from the previous executions trims out the model designing time [YHP00].

4.2 Literature Review

In the literature, several work has been done in order to support and automate the business processes such as strategy-driven [**bider2005strategy**], activity-centric[YMMS09], activity-oriented [Rei06], artifact-centric [CH09], capability-driven [SGHZ12], archimate [AIvH+15] and subject-oriented [FKSS13]. A detailed description about these approaches and their degree of satisfying the requirements mentioned in Section 4.1 has also been provided.

Requirements	Requirement Satisfaction Phase	Pre-requisites
R1	Modeling phase	1. Main intention can be refinable into sub-intentions. 2. Organizational members can view the intentions at different levels.
R2	Deployment phase	1. Intention cost estimation that includes all recursive sub-intentions and resources. 2. Cost estimation including the strategy.
R3	Modeling phase	1. Each sub-intention should be achievable and valid.
R4	Deployment phase	1. Satisfaction of R1. 2. Understanding of the intentions and how they can be reached.
R5	Both Modeling phase and Deployment phase	1. Satisfaction of R1. 2. The output of intention is based on the inputs provided by different members of the organization.
R6	Modeling phase	1. Satisfaction of R1,R2,R3,R4 and R5

Table 4.1: Requirements Satisfaction Phase

4.2.1 Strategy-Driven

This approach [**bider2005strategy**] defines business process in terms of goals and strategies in order to achieve the goals. It also uses map representation system that conforms goals and strategies. In this approach the details regarding visibility of goals has not been addressed, hence requirement R1 is not satisfied. Also details about cost of achieving a goal through a strategy has also been not addressed, hence requirement R2 is also not satisfied. The requirement R3 contradicts with the process rule of this approach which states that "There is no goal/strategy in the map that can be considered as the subset of another one". Requirement R4 is also not satisified due to the fact that this approach follows strategy driven modeling. Information about participation of different members has also not been metioned, hence requirement R5 is also not met. This approach addresses the requirement R6, as it supports reusability of the map components in different maps.

4.2.2 Activity-Centric

The activity-centric approach [YMMS09] also supports knowledge workers by providing shared activity constructs as a computational unit for organizing the work. This approach provides team level view of past and ongoing work and also supports propogation of completed activities to the existing activities. Hence requirement R1 is partially met. The information about cost of achieving a goal or activity has not been mentioned. Thus requirement R2 is not satisfied. In this approach, activities support objectives at various levels of granularity and thus requirement R4 is met. Since the main focus of this activity, it has not provided any information regarding working style based on goals instead working style is based on activities. Hence requirement R5 is also met. Cross activity overview pattern is one of the pattern described in this approach which does unifying work across the team members, hence reequirement R5 is addressed. Since this approach also supports reusable activity patterns requirement R6 is also met.

4.2.3 Activity-Oriented

The activity-oriented approach [Rei06] is traditional workflow management approach where the main focus unit is business process' activity rather than strategy. In traditional workflows the concept of "process view" from different levels of an organisation not addressed, thus requirement R1 is not addressed. The details about cost calculation is not mentioned, hence requirement R2 is also not satisfied. Though this approach does not support sub-processes directly, it provides support for plugging in sub process extensions, this satisfies requirement R3. From goal oriented to activity oriented working style, so the working style is based on activities and not on goals. Thus requirement R4 is not addressed. Traditional workflow models like BPMN do not support participative modeling, but extensions like social-BPM supports social interactions. Hence requirement R5 is addressed. Re-using of existing activities is not addressed, thus requirement R6 is not addressed.

4.2.4 Artifact-Centric

The artifact-centric approach [CH09] combines business data as artifacts and business process in a holistic way. This approach clearly states that artifacts "views" is not addressed, thus requirement R1 is not satisfied. The requirement R2 which is about cost calculation is not addressed. This approach allows modularity and componentization of business operations at various levels, hence requirement R3 is satisfied. Requirement R4

is partially met as the process evolves through a series of intermediate goals. Requirement R5 is not met due to the fact that concept of social organizational modeling is not addressed. Requirement R6 is partially met because, only the concepts of modularization, componentization at various levels of abstraction has been discussed but reuse of components has not been addressed.

4.2.5 Capability-Driven

The capability driven approach [SGHZ12] also proposes to support the changing environment of organizations. But in this approach there is no information about the visibility of goals has been addressed. Hence requirement R1 is not met. This approach claims that, it overcomes the challenge of high cost in developing applications but there is no clear details about how cost calculation is done, hence requirement R2 is partially addressed. In this approach, the top goal is refined into a number of sub-goals, then each sub-goal is lined to one or several KPIs. Thus requirement R3 is met. Since visibility of goals is not addressed, the details about explicit goals which is requirement R4 is not addressed. Requirement R5 is not addressed as the concept of multiple resources working together is not described. Reuse and execution of capability delivery pattern has been addressed, this meets requirement R6.

4.2.6 ArchiMate

This approach [AIvH+15], investigates if ArchiMate modeling language tool can be used to model strategies and also addresses the properties of Archimate. This apporach provides visibility of whole process, supports "viewpoints" in different levels of modeling. Thus requirement R1 is addressed. Requirement R2 which is cost of achieving each goal is not addressed. The approach provides three levels of modeling i.e., business, application, and technology, thus requirement R3 is partially met. This approach provides visibility of whole process which supports explicit goals but ArchiMate modeling language is not very easy to use to model multiple strategies with goal concept, thus requirement R4 is partially met. Requirement R5 is addressed as it supports different resources participation at different levels. Requirement R6 is also addressed as re-use of existing models is supported.

4.2.7 Subject-Oriented

This approach [FKSS13] supports multi-agent business process models that improves efficiency of the business logics. In this approach, requirement R1 is partially addressed

as S-BPM shows process view of who communicates with whom but not how the process advances or if it terminates at all. Requirement R2 which is cost of achieving each goal is not addressed. Requirement R3 is addressed as the main process net can be divided into sub-nets. Requirement R4 is not addressed, as the concept of explicit goals is not described. S-BPM emphasises "subjects" in a process as a decenrtalised, interacting entities thus requirement R4 is addressed. Requirement R6 which is about re-using of existing knowledge is not addressed.

4.2.8 Evaluation of the Approach

The approach *Adaptive Case Management*, proposed by Hermann et. al [HK11] bridges the gap between business processes management and flexibility in adapting knowledge intensive processes by defining activities and re-using created activity structure. When the required activities changes dynamically, capturing them for re-use are not helpful [SBLW15]. Though the approach *Ad-hoc and Collaborative Processes* proposed by Dustdar et. al. overcomes the challenges in process aware collaborations, defining activities in a ad-hoc fashion does not support human actor in various cases [SBLW15]. Also the approach proposed in Chapter 5 serves as a complementary to the above discussed approaches. This is because every existing approach satisfies one or few of the requirements even though not all requirements are satisfied. The Table 4.2, shows the number and extent of requirements satisfied by the existing approaches.

Approach	**R1**	**R2**	**R3**	**R4**	**R5**	**R6**
Strategy-Driven	No	No	No	Partial	No	Yes
Activity-Centric	Partial	No	Yes	Partial	Yes	Yes
Activity-Oriented	No	No	Yes	Partial	Partial	No
Artifact-Centric	No	No	Yes	Partial	No	No
Capability-Driven	No	No	Yes	No	No	Yes
ArchiMate	Yes	No	Partial	Partial	Yes	Yes
Subject-Oriented	Partial	No	Yes	No	Yes	No

Table 4.2: Evaluation of the Approach

Legend :

Yes Requirement is addressed

No Requirement is not addressed

Partial Requirement is partially addressed

5 An Approach to Resource-centric Organizational Modeling

This section describes in detail about the technical approach that has been taken to solve the problem mentioned in problem statement section of Chapter 1. This chapter also provides an outline of the design, the methodology and overall structure of the approach. The first section of this chapter provides an overview of modeling process approach. The second section provides a brief description about the frameworks and libraries used. The third section discusses in detail about the *top-down approach*, which has been used to realize the resource-centric organizational modeling. The fourth section discusses the design methodology followed to realist this approach of developing a descriptive modeling web based editor. The final section discusses in detail about the relationship between each entity types of this approach. The main contribution of this approach is to explain the concepts in a concrete way, whose abstract concepts are discussed in earlier chapters.

5.1 Overview of the Modeling Process

The main focus of this approach is to develop a web-based editor which can be used by business experts to model the informal processes. Also in this thesis work, the scope of modeling is limited only to the descriptive type of modeling. As we mentioned before, the resource definitions required for the editor is made available from the first phase P1 of the InProXec approach. Business experts develop descriptive models through the editor using these resources to achieve main intention that contains sub-intentions, strategies etc. The reason for following descriptive modeling approach is due to the fact that models reusable descriptive data and these stored models provides means of execution for the phases P3 and P4 of InProXec. The model provides necessary concepts and relations for modeling the core elements of resource centric organizational modeling. Resources are abstract description which are made concrete during initialization of an instance. There are also resource specific views based on the participating resources' role. For example, based on the privilege provided to a participant he can view/edit/own/follow the instances. Initializing resource-centric models requires *acquiring* and engaging

interrelated resources which is explained in detail in the following sections of this chapter.

5.2 Technologies and Frameworks

In order to realize the web-based editor of Intention-centric Organizational Modeling, a formal inquiry has been done to choose suitable technologies and frameworks required. The below specifications were finalized and *client-side scripting*[1] has been chosen due to the fact that our developed editor is web-based.

1. *Clojure*[2] as the programming language
2. *IntelliJIDEA*[3] as the development environment
3. *MVC* as the architecture pattern
4. *Re-frame*[4] as the pattern for writing SPAs in ClojureScript, using Reagent

Other than the above listed frameworks and technologies, frameworks like *react-bootstrap*[5], jquery[6] were also used to provide more optimal view of the editor. Along with this we have also used libraries like bidi[7] and pushy[8], to handle page navigation fom current location to the desired location in the URL[9] of the browser.*Clojure* is a dynamic, general-purpose programming language, combining the approachability and interactive development of a scripting language with an efficient and robust infrastructure for multithreaded programming. *ClojureScript*[10] is a compiler for Clojure that targets JavaScript which has been designed to emit JavaScript code. In our implementation, we have used both Clojure and Clojurescript. We also used *Reagent*[11] which provides a minimalistic interface between ClojureScript and React[12]. A *Re-frame*[13] is a pattern for writing applications in ClojureScript, using Reagent.

[1] https://en.wikipedia.org/wiki/Client-side_scripting
[2] https://clojure.org/
[3] https://www.jetbrains.com/idea/
[4] https://github.com/Day8/re-frame
[5] https://react-bootstrap.github.io/
[6] https://jquery.com/
[7] https://github.com/juxt/bidi
[8] https://github.com/kibu-australia/pushy
[9] URL- Uniform Resource Locator
[10] http://clojure.org/about/clojurescript
[11] http://reagent-project.github.io/
[12] https://facebook.github.io/react/
[13] https://github.com/Day8/re-frame

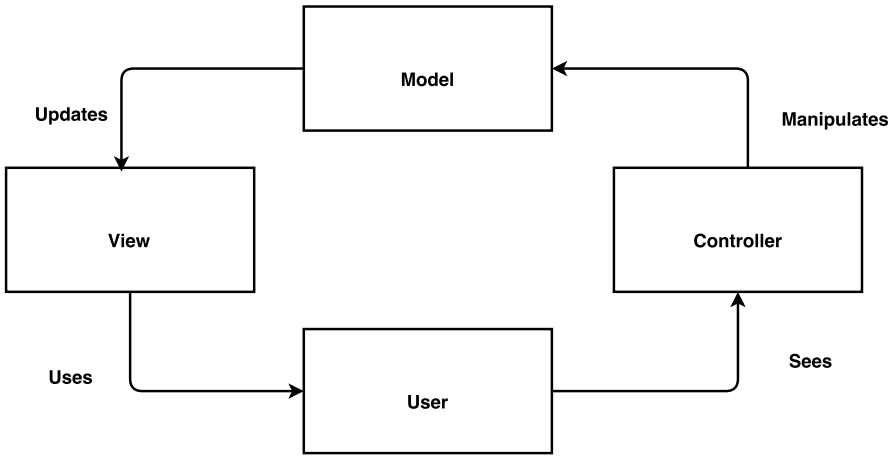

Figure 5.1: MVC architecture components

5.2.1 MVC Architecture

The architecture of the developed user interface of the editor is based on the *Model-View-Control (MVC)* design pattern. The MVC paradigm allows to separate business logic from the code that controls presentation and event handling [Ora16].Each entity view in the web page is made up of combination of at least on Model and View, and one or more Controls. The functionalities of individual files which acts an Model, View and Controller has been shown in the Figure 5.1

Model artifact stores the required data structure for web-editor. In the developed model artifact, the four main types of data stored inside the artifact are intentions, strategies, capabilities and informal process instances.

View artifact contains HTML elements and HTML constructs that describe the way of displaying the data from Model to the user. Most of the common functionalities that render user interface components are re-used.

Control artifact contains the handler functions which can only change the model. Even the initial values of the model are put inside the control. This artifact has functions causes default database to change, which then causes a re-render of view and then the user sees new view.

Apart from the above artifacts, there is another important artifact that registers subscription functions i.e., query layer of the data. As view components never source data directly from default model, we use *subscription* functions. Subscription functions returns a value that changes over time i.e based on a user events.

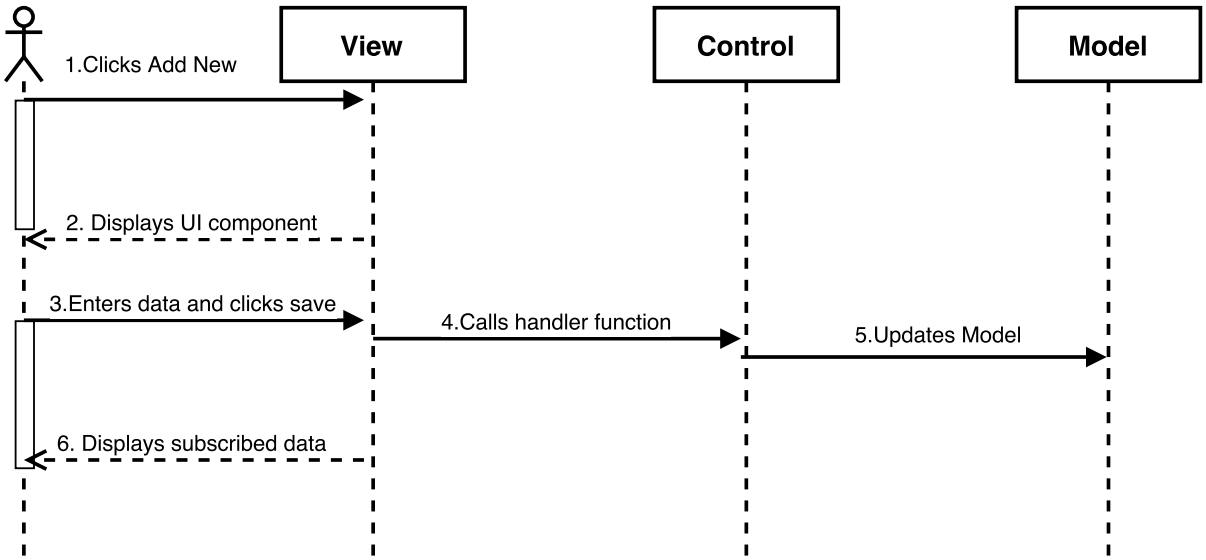

Figure 5.2: MVC Pattern of adding new entity

Example: Component using MVC Pattern

The Figure 5.1 below shows the simplifed version of how the components interact with each other using the Model-View-Control (MVC) pattern, for the functionality of adding new entity data. This functionality is same for all the types such as intentions, strategies, capabilities and informal processes and below is the detailed explanation of each interaction.

1. User clicks the tab *Add New* button in the developed editor.
2. In response to the user click, the view displays the respective user interface component for entering the new entity data details.
3. User enters the required basic details for adding new entity data and clicks save button.
4. The view dispatches the data to control, which can only modify the model.
5. Control inserts/updates data into the model.
6. View displays the updated model as it has been subscribed to the model.

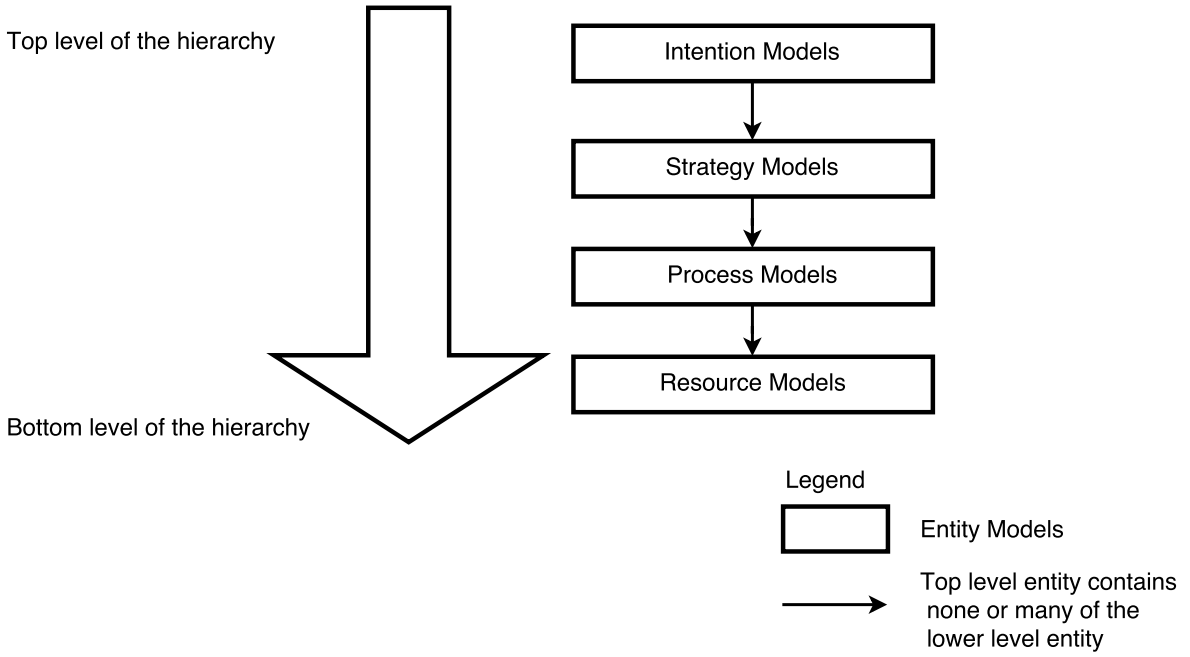

Figure 5.3: Top Down Modeling Approach

5.3 A Top-Down Modeling Approach

Intentions are defined hierarchically, intentions can contain and extend intentions which are called as sub-intentions. Intentions can contradict to itself as well. Intentions are associated with strategies, thus intentions can be realized through strategies. Strategies are associated with capabilities. These capabilities are of two types *functional capabilities* and *cross functional capabilities*. Functional capabilities are associated with resources and cross functional capabilities are associated with functional capabilities. Each informal process model is a strategy that has capabilities, strategies, resources that are created out of capabilities and intentions. In the Figure 5.3, it has been shown that how this modeling approach starts modeling from top level of the hierarchy and does modeling until the lower level is reached.

Bider et al [**bider2005strategy**] propose a strategy-driven modeling approach of processes. Processes are defined based on the goals and refinement continues until meaningful operation level is reached. Consequently, created models are easily changeable as they are decoupled from their operational terms. Such declarative approaches provide more flexibility and enable easier change of the business process models [SBLW16]. As we mentioned before, the modeling approach in our context is descriptive modeling approach which starts from the top level and refines modeling until the bottom level is reached.

5.4 Design Methodology

When designing the user interface components and functionalities required to develop the tool, most of the similar functionalities are designed as common functionalities and re-used. This reduces the unnecessary functional redundancies and overhead. The common functionality methodology are implemented for both model functions and view functions. Some of important methodologies followed with respect to user interface components design are 1. multiple items to be selected from multiple items are displayed as *list* 2. selecting single item from multiple items are displayed as drop down. For example, to select multiple strategies from a list of strategies, available strategies are displayed as a list from which the user can select desired number of strategies. Another important methodology followed during user interface design is, for every entity their properties should be displayed only under their respective properties tab. For example in the Figure 5.4, the basic properties such as name, target namespace and process type of an informal process model should be displayed only under the respective basic properties tab. This methodology is followed uniformly throughout the design of all the entity types such as intention definitions, strategy definitions, capability defintions, context definitions, instance definitions and informal process definitions and for all of their property types.

All data are stored only under the data artifact. This applies to the labels and text fields of all user interface elements and this data can be updated only through the handler function. Through *settings* option, the user also has option to add new namespace and intention relation type. From the Figure 5.4, it is clear that a standard design methodology has been followed to display the list of available entity types such as intentions, strategies, capabilities etc., and to display their respective properties such as basic, entity specific, instance data, etc., properties. Though the top-down modeling approach 5.3, shows that definition of each entity type is contained within another entity type, as per the design, separate entities references each other but does not contain each other. For instance, a strategy containing an intention should contain only the intention's unique reference identifier but not the actual intention itself. Later, in the view of strategy, actual intention properties are fetched and displayed based on the unique reference identifier.

The research objectives mentioned in Chapter 1 are also met during the development of the editor. The validity of the research objectives are discussed in Chapter 6 using the motivating scenario discussed in Chapter 3.

Organizational intentions transparency (R1): In the current functioning system, users are stored in database artifact and these users can login through their valid credentials.

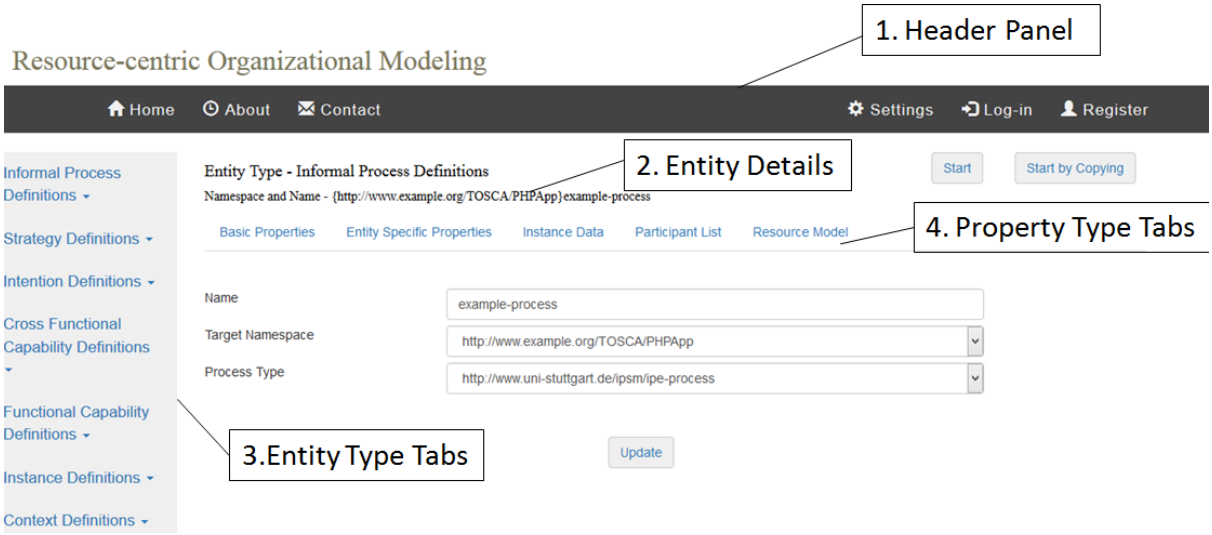

Figure 5.4: User Interface Design of the Editor

Organizational intention resource-based cost estimation (R2): Intentions are associated with strategies, which are associated with capabilities and hence with resources. Cost is calculated in a recursive manner. For example, consider we need to calculate cost of an instance whose entity type is intention. To calculate the cost we go recursively to the lower levels starting from the required level. Since our instance is of type intention, we start iterating through every associated strategy, and for each associated strategy, we iterate through their instance descriptors as well. In case the cost of an instance descriptor of a strategy has not been specified, we specify it by calculating the cost of instance descriptors of associated informal process definitions. For informal process definitions, we use the cost resource definitions. This ends the recursion and returns the total sum as the cost of an instance of type intention

Organizational intention achievability estimation (R3): Similar to resource-based cost estimation for an intention, the achievability of an intention also depends on its instance state. For example, if an instance of type intention is associated with a strategy which also has an instance that is completed. Then the total number instances remaining to be completed to achieve an intention is calculated as one out two instances.

Intention oriented working style (R4): The users can login and create intention models, strategy models, informal process models etc., through the developed editor.

Participative organizational modeling (R5): Each entity type that can be acquired or instantiated has list of participants with their corresponding privileges.

Re-use of organizational knowledge (R6): The descriptive information about each models can be stored and re-used for next enactments.

5.5 Characteristics of the Entity Types

As mentioned earlier, the entity types are modeled as descriptive informations, this is because models can be initialized and can be made runnable elements which are required for subsequent phases such as P3 and P4 of InProXec. An IPE model describes the main intention that reflects the informal process' main goal. Each intention may be refined into sub-intentions. The IPE model's initial context specified the triggers that signal when model's corresponding resources should be initialized and subsequently work towards the informal process' main intention[SDDL15]. The final context specifies conditions for determining the processes' main intention as successfully achieved. As disucssed in Section 5.3, entity types are dependent on one or another. For example, for successful execution of an intention we need strategies or sub-intention and resources that has capability to achieve an intention through a specified strategy.

5.5.1 Context Intention Relationship

Intentions connect initial context definitions with final context definitions Figure [14]. From the Figure 5.5, it is clear that only on successfully achieving an intention the context reaches desired final context. Consider in the Figure 5.5, from our motivating scenario that C1 as inital context of the motivating scenario, I1 as the main intention of the motivating scenario and C2 as the desired final context of the motivating on successful completion of intention I1. Only when the main intention of increased revenue and number of unit sales (I1) is achieved for the quarter, the desired final context(C2) is achieved else the execution may end up in a state(C3) other than the desired final context.

5.5.2 Capabilities Resource Relationship

Each organizational capability must be provided by a resource in the organization. Resource models are optional to make precise definitions of resources needed. The relationship between organizational capabilities and organizational intentions has been provided in the Figure 5.6

Each *intention* can require certain *capabilities* which are provided by *organizational resources*. As a result each informal process model is a strategy that has cross-functional capabilities, resources which are created out of capabilities and an intention specific to

[14] C Timurhan Sungur, An approach to supporting and automating informal processes, May 2015

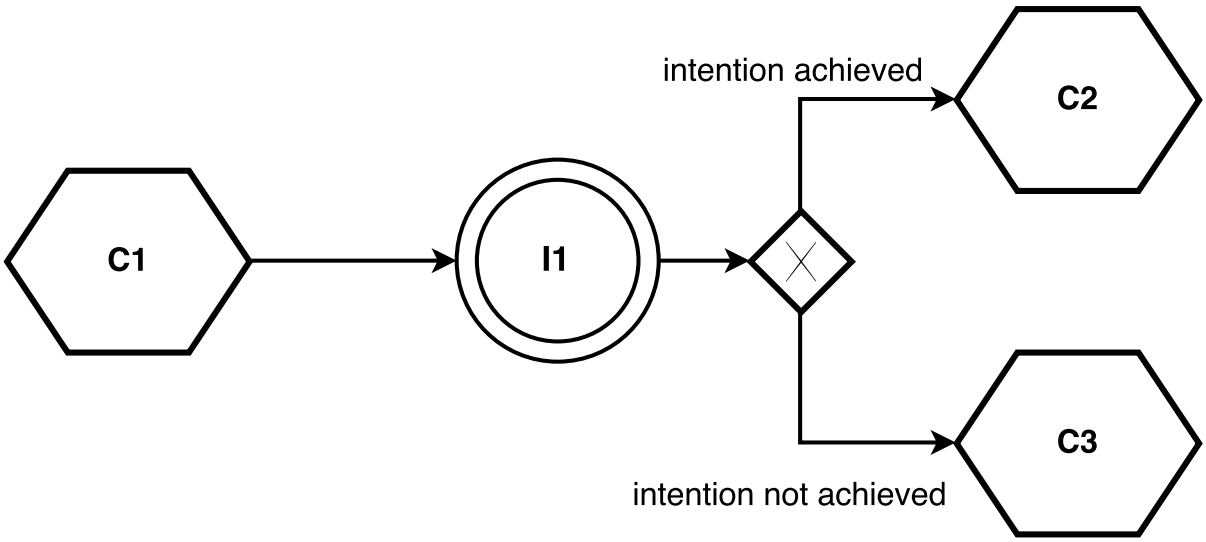

Figure 5.5: Context Intention Relationship

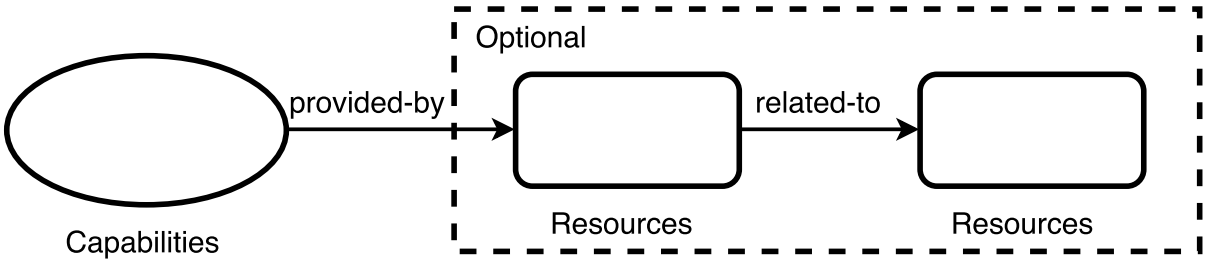

Figure 5.6: Capabilities Resources Relationship

that strategy. In our context of resource-centric organizational modeling capabilities are of two types such as *functional capabilities* and *cross functional capabilities*. Functional capabilities are the capabilities that are associated with resources that satsifies required capabilities. Cross functional capabilities are the capabilities that contains functional capabilities. The required resources are typically provided by some services which are called as *resource organizers*. Resource organizers are responsible for preparing the resources for the process execution and releasing them upon process completion [SBBL14]. In our functioning system, resource models can be created for each informal process models.

5.5.3 Acquirable Entity Types

This thesis work scope is limited only till creating descriptive models. Creating runnable or executable components are not part of the implementation. A short overview about

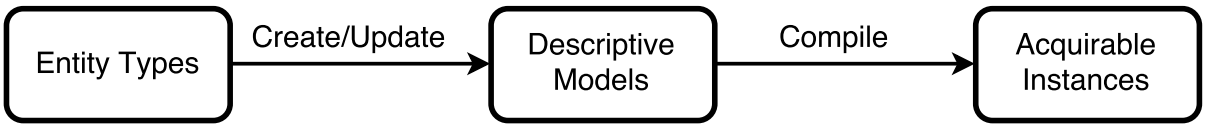

Figure 5.7: Acquirable Instances

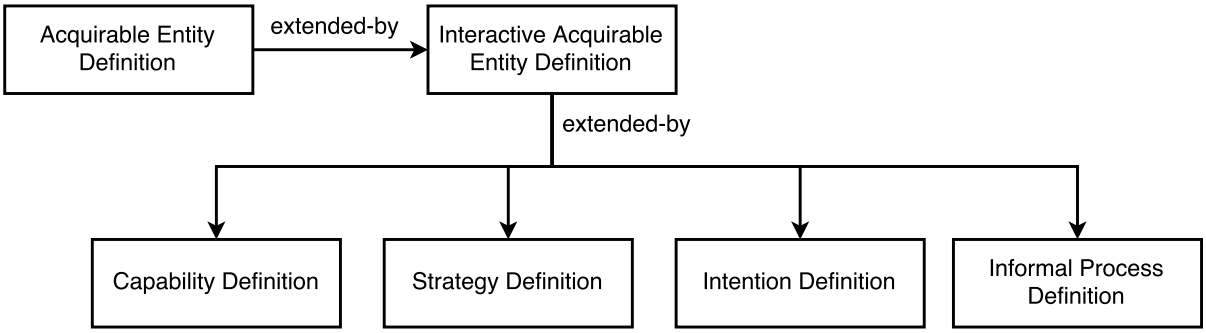

Figure 5.8: Acquirable Entities Hierarchy

acquirable entities are provided as the editor provides support to add functionality that are required to instantiate a process. The models can be created using the developed editor, these models when compiled they are acquirable i.e., these are instantiate-able which is shown in the Figure 5.7. Final state of the model instance is saved as they are required for subsequent phases during the execution of informal processes [SBLW15].

As shown in the Figure 5.8, acquirable entity models are extended by interactive acquirable entities, which are further extended by capability models, strategy models, intention models and informal process models. Here the term capability model refers to functional capability model. As mentioned earlier, acquirable entities are entities of entity types which are instantiate-able i.e., user can create instance data out of these entity types' models. Interactive acquirable entities are acquirable entities which has interactive participant list with each participant assigned with their respective privileges such as owning an entity, viewing and entity, editing an entity and following an entity. During the user interface design of interactive acquirable entities two additional tabs to display instance details and participant list details has been included along with the tabs to display basic and entity specific properties. Along with this editor also provides functionalities such as extracting only the required instances from an entity and viewing basic properties of a particular participant. A sample Extensible Markup Language (XML) schema definition code 5.1, has been provided to understand the realization of implementing instance data of each entity type. In the listing 5.1, the element of name *InstanceDescriptors* hold details of each of the instance data i.e instantiate-able. Each identifiable entity definition is identified using a combination name and namespace pair, they can also specify multiple entity definitions.

Listing 5.1 XML Schema Definition of Acquirable Entity

```
<complexType name="tAcquireableEntityDefinition" abstract="true">
<annotation>
<documentation>
Each acquirable / initializable entity definition.
</documentation>
</annotation>
<sequence>
<element name="InstanceDescriptors" type="ipsm:tInstanceDescriptors"
minOccurs="0" maxOccurs="1"></element>
</sequence>
</complexType>
```

6 Case Study on Resource-centric Organizational Modeling

In this chapter, we provide architecture of the functioning system as a first section. This section provides, implementation details along with the reason for making certain decisions regarding the implementation. The second section explain how motivating scenario has been realized using the proposed modeling approach. Successful modeling of the motivating scenario using the developed editor serves as a proof for usability of the web editor. The final section validates the system by validating it with the proposed approach. This section also has some requirement evaluation with the state of the art approaches.

6.1 Architecture of the Functioning System

As discussed in the Chapter 5, informal process targets for accomplishment of an intention. Thus in the Figure 6.1, we associate intentions with both process definitions and strategy definitions as intention definitions are used by both process definitions and strategy definitions. Intentions are associated through resources either through strategies or through informal processes. Intentions can be refined by defining sub-intentions, which can be defined recursively as independent informal process. For example, in our motivating scenario the main intention increase revenue and number of unit sales can be refined into sub-intention of improve customer help desk portal. This sub-intention can be associated with process models. This *intention-based* approach enables describing process declaratively, i.e., without describing *how* the intention is achieved, and providing information about *what* has to be achieved. This avoids the need for predefined business logic in the representations of informal process [SBBL14].

Also from the Figure 6.1, it is clear that we followed the MVC architecture to design the user interface. Business experts can use the editor to view/update the descriptive entity details. Whenever a change in the model data is detected respective handler function is *dispatched* and the corresponding handler can only *update* the model. Since we associate every entity type with another entity type, model data of an entity type is

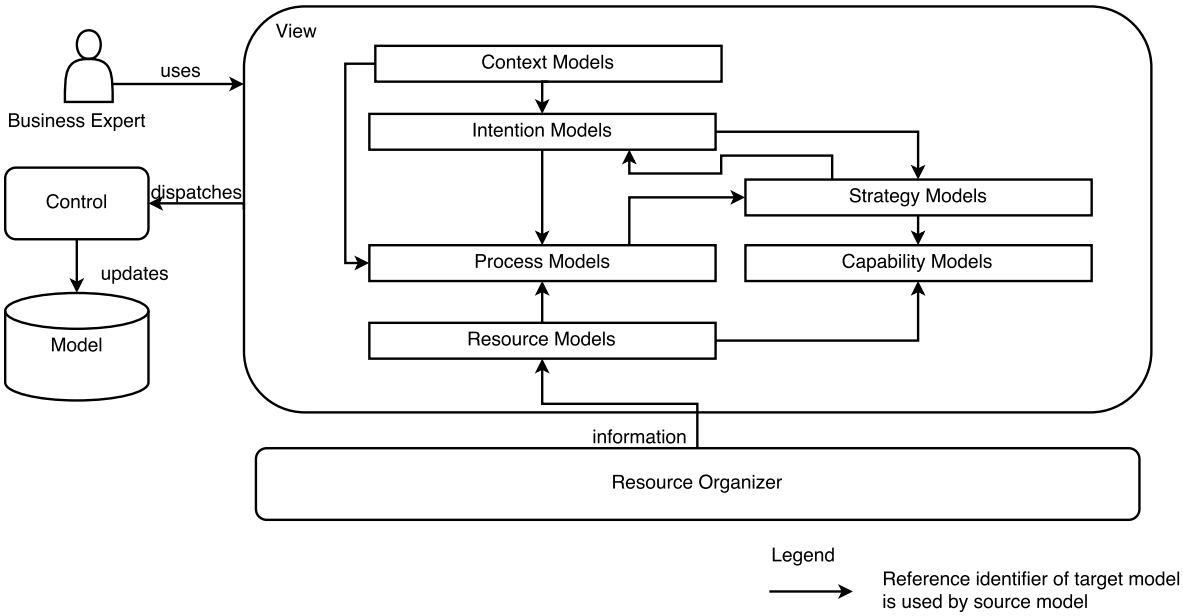

Figure 6.1: Architecture of the Functioning System

required by another entity type which are resolved using the unique reference identifier. For example, intention model's unique reference identifier of intention *improve help customer help portal* is required by the strategy *through application development*. This is because for the strategy "through application development" , intention "improve help customer help portal" is the target intention.

6.1.1 Application Flow

In this sub section we provide an overview about how page navigation from current location to the desired location happen in URL[1] of the browser. The external libraries used for route navigation, parses URLs into data structures and also generates URLs from data structure defined as required routes. We call a function to dispatch route, with the matched route. Then we also have fucntion that parses the URL, to turn a URL into a data structure representing it. From the Figure 6.2, it is clear that route navigation for each entity items happens based on their entity type and its own unique reference identifier.

Each entity item has basic properties such as *name* and *target namespace*. The entities are identified using their unique id which is generated using the combination of name and

[1]URL- Uniform Resource Locator

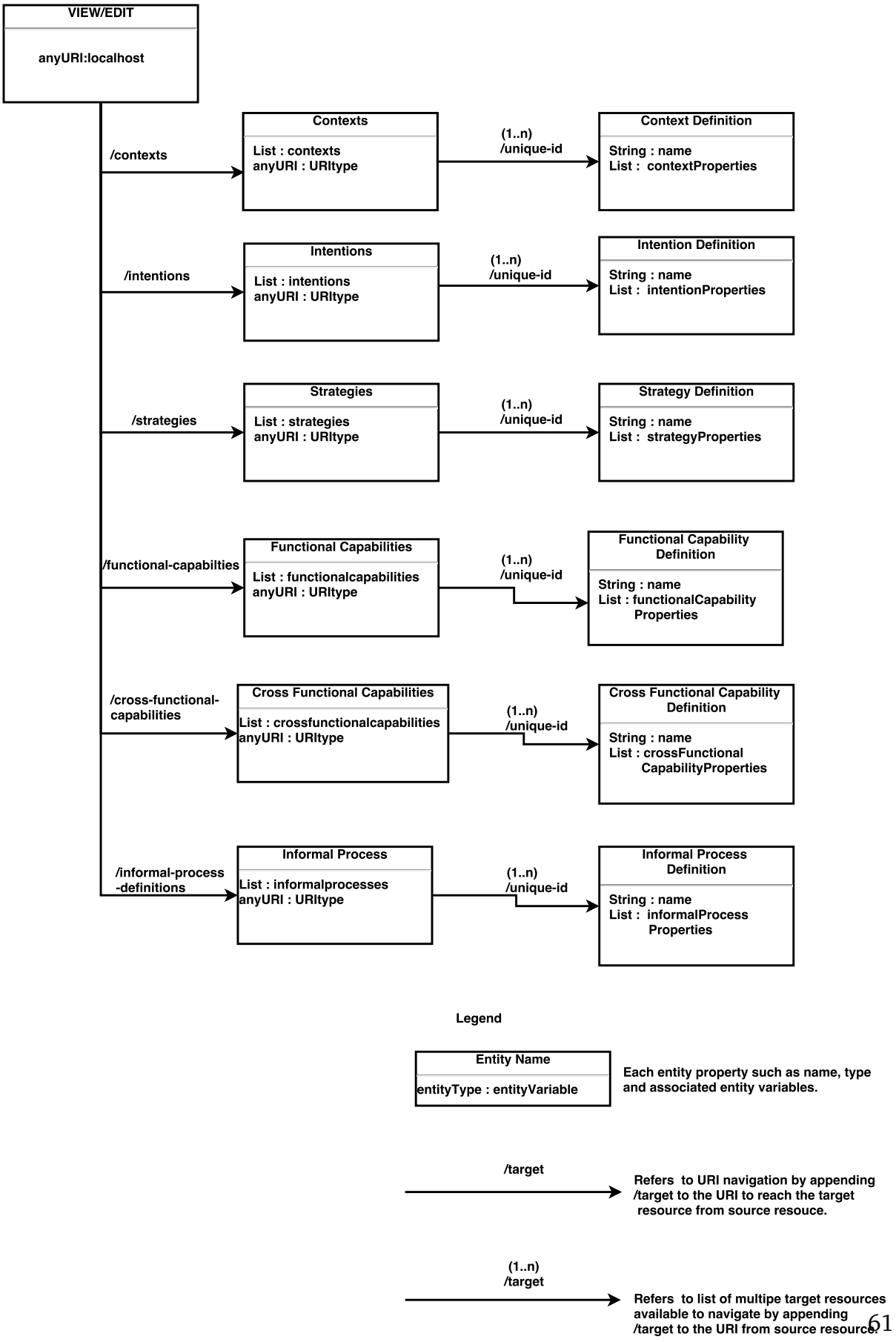

Figure 6.2: User interface URL navigation of the functioning system

target namespace. Other entities that are associated with a particular entity are resolved through this unique identifier. For example, in our motivating scenario consider the intention *improve the customer help desk portal* when creating model for this intention, business expert provide name and namespace for this intention and add it to the database. A unique identifier is generated for the intention model using the combination of name and namespace by the system. The strategy (in our scenario *through application development*) that is associated with this intention, just contains only this unique identifer for the reference.

6.2 A Concrete View of Entity Types

It is important to discuss the concrete concepts of informal process from an organizational aspect, because they have a direct effect on the outcome of the informal process [SKL14]. This section discusses about how Intention-centric organizational modeling realized the user interface editor from an organizational aspect by taking the motivating scenario discussed in Chapter 3. Though developing schema definitions are not part of the thesis implementation, it has been provided because the editor has a view that is capable of adding, viewing, deleting and updating model data aligned with the schema definition. A typical XML Schema Definition of entity type has been provided in the listing 6.1.

6.3 Realization of Motivating Scenario

The realization of motivating scenario is explained by integrating the concepts discussed in Chapter 3 and the informal process modeling approach discussed in Chapter 2. From the Figure 6.3, it is clear that to realize the motivating scenario using the proposed approach it is important to model them step by step as mentioned in the Informal Process Modeling approach. The developed editor also supports dynamic changes in the models whenever there is a need to add new models. As each models are designed in each modeling steps, we have detailed them individually in the following sub sections. The reason for selecting motivating sc

6.3.1 Realization of Context Definitions

In the informal process modeling approach, the first modeling step is to model the context definitions(M1). Each informal process start from an initial context, i.e., IPE Context and aims to achieve an intention, i.e., an IPE Intention [SBBL14]. After reaching

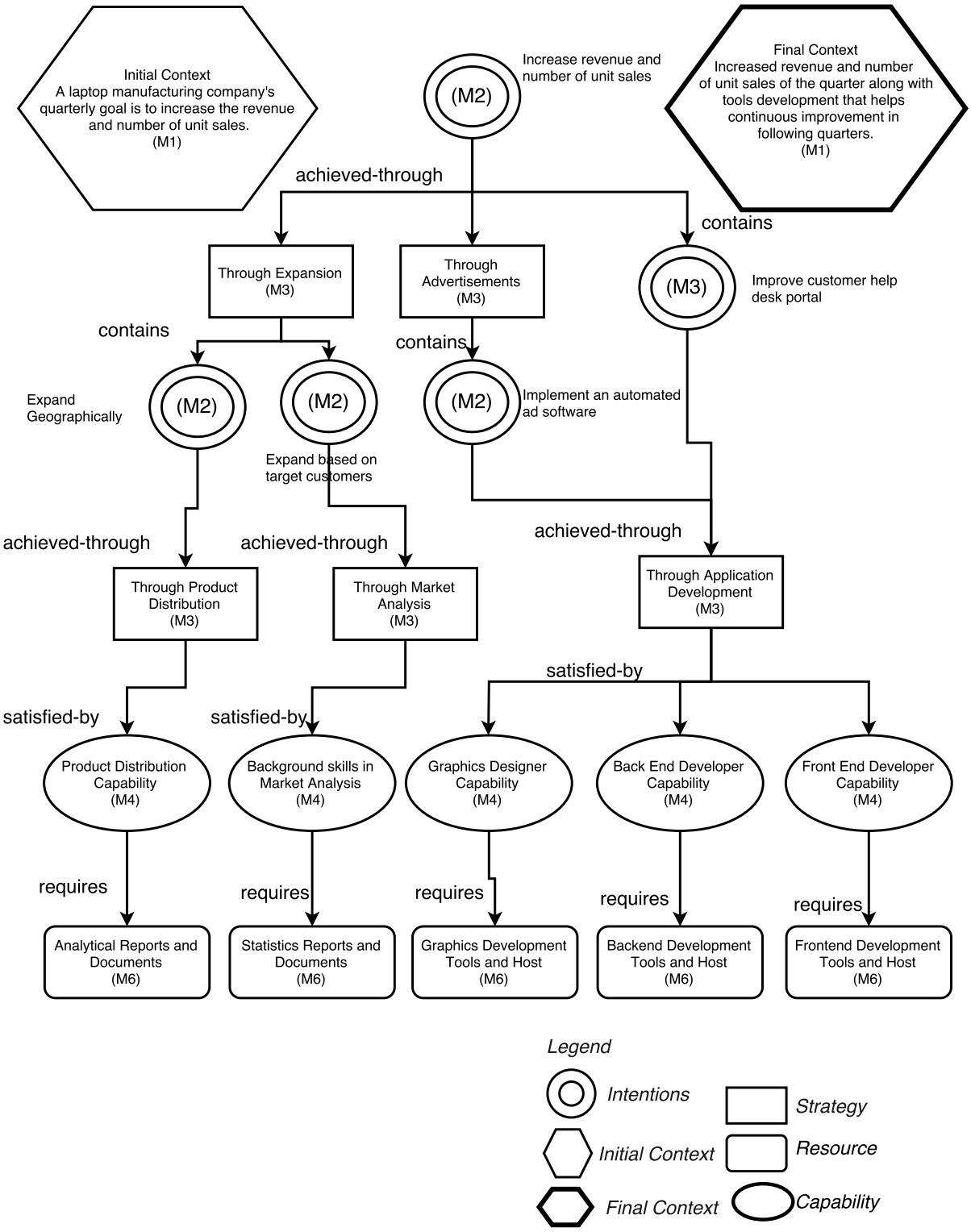

Figure 6.3: Realization of Motivating Scenario

Listing 6.1 XML Schema Definition of Entity Type

```
<xs:complexType name="tEntityType" abstract="true">
<xs:complexContent>
<xs:extension base="tExtensibleElements">
<xs:sequence>
<xs:element name="Tags" type="tTags" minOccurs="0"/>
<xs:element name="DerivedFrom" minOccurs="0">
<xs:complexType>
<xs:attribute name="typeRef" type="xs:QName" use="required"/>
</xs:complexType>
</xs:element>
<xs:element name="PropertiesDefinition" minOccurs="0">
<xs:complexType>
<xs:attribute name="element" type="xs:QName"/>
<xs:attribute name="type" type="xs:QName"/>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" type="xs:NCName" use="required"/>
<xs:attribute name="abstract" type="tBoolean" default="no"/>
<xs:attribute name="final" type="tBoolean" default="no"/>
<xs:attribute name="targetNamespace" type="xs:anyURI" use="optional"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>>
```

an intention, there is resulting IPE Context. In the motivating scenario, the user can add new contexts by providing basic properties such as name of the context and target namespace of the context as they serves as unique reference identifier for these contexts. After successfully adding the basic properties, user can provide entity specific properties such as contained contexts inside the main context, entity definition details about the contexts and participant list such as which user has what type of privileges. The required context definitions are modeled first because these definition are required for modeling intention definitions and process definitions.

6.3.2 Realization of Intention Definitions

After modeling context definitions(M1), the second step of the modeling is to model the intentions(M2). For example, in our motivating scenario we have main intention of "increase revenue and number of unit sales" and other sub-intentions that are emerged out of main intentions and strategies of the main intention. The user can provide descriptive information about particular intention as intention definition. Similar to context modeling, the user has to provide basic properties such as name and target namespace

required for unique identification of this entity. After providing basic properties, the user has to provide entity specific details of the intention such as due date and time for intention completion, priority of the intention, cost of the intention, sub intentions that are contained under this particular intention and how the sub-intentions are related to this intention. The strategies to achieve this intention and contexts of the intention are also provided as entity specific properties. The participant list with respective privileges for each participant are also provided.

6.3.3 Realization of Strategy Definitions

After modeling context definitions(M1) and intention definitions(M2) user can proceed to model the strategies through which an intention can be achieved which is third step of the modeling process. For example, in our motivating scenario user can model the strategies such as *through expansion*,*through advertisements* and other required strategies as third step of the modeling process. Similar to earlier modeling steps, during the modeling of strategy also user required to provide basic properties such as name and target namespace. After providing the basic properties, entity specific properties such as target intention of the strategy, intention, capability and process definitions associated with strategy are also provided. Since strategy is also an interactive acquirable entity similar to intention, participant list details are also provided during modeling of strategies

6.3.4 Realization of Capabilitiy Definitions

There are two types of capabilities. Functional capabilities and cross-functional capabilities. Functional capabilities are the capabilites that associated with other entity types. Cross-functional capabilities contains multiple functional capabilities. Similar to earlier entity types basic properties such as name and target namespace are added to get the unique reference identifier and entity specific properties for both capabilities are added. Since cross functional capability contains functional capabilities it holds the identifiers of the functional capabilities contained in it. Functional capability definitions also has participant list details similar to intention definitions and strategy definitions.

6.3.5 Realization of Process Definitions

By modeling the business processes based on the resources that work towards certain intentions, informal processes are modeled without predefining their business logic

[SBBL14]. Also as mentioned earlier each informal process starts from an initial context and aims to achieve an intention that results in a final context. Thus we require context definitions and intention definitions before modeling process definitions. Similar to earlier modeling of entity types, process modeling also require basic properties such as name and namespace and entity specific properties such as associated intentions, contexts and resources. Process definition also has participant list similar to other entity types.

6.3.6 Realization of Resource Definitions

As discussed earlier each resource can be related to another resource which are defined using predefined or custom *relationships* [SBBL14]. These resources are managed through *Resource Organizers,* this is because resource organizers are used to bring together the relevant interrelated resources that work towards to achieve the corresponding intentions. TOSCA [BBKL14] can be used to model all nodes and relationship among them. In our context, we can consider resources as nodes to make use of the TOSCA's service. The schema definition of considering each resource as node is provided in the listing 6.2. In the developed editor, the resource models are managed by embedding the open source modeling tool Winery web page [KBBL13] in our editor's web page. This is because, it creates a new service template that contains an application topology by using the topology modeler. Winery also offers all available node types in a palette. From there, the user drags the desired node type and drops it into the editing area. There, the node type becomes a node template i.e., a node in the topology graph. Node templates can be annotated with requirements and capabilities, property values, and policies. The screen shot of modeling sample resource has been provided in the Figure 6.4.

In order to achieve this we use tosca repository url referring to winery and the other one referring to topology modeler of the winery. Using these values we create corresponding url required for our modeling based on the name and namespace properties of an entity. The functionality to generate resource model page, using tosca repository url and topology modeler url is provided below.

```
{topology-modeler-url}?repositoryURL={encoded-tosca-repository-
url}&ns={encoded-target-namepsace}&id={encoded-id}#
```

6.3.7 Realization of Instance creation

Initializing resource-centric processes requires acquiring and engaging interrelated resources [SBLW15]. As mentioned earlier, the phases of compiling and initializing of

Listing 6.2 XML Schema Definition of Node Type

```
<xs:complexType name="tNodeTemplate">
<xs:complexContent>
<xs:extension base="tEntityTemplate">
<xs:sequence>
<xs:element name="Requirements" minOccurs="0">
<xs:complexType>
<xs:sequence>
<xs:element name="Requirement" type="tRequirement" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Capabilities" minOccurs="0">
<xs:complexType>
<xs:sequence>
<xs:element name="Capability" type="tCapability" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Policies" minOccurs="0">
<xs:complexType>
<xs:sequence>
<xs:element name="Policy" type="tPolicy" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="DeploymentArtifacts" type="tDeploymentArtifacts" minOccurs="0"/>
</xs:sequence>
<xs:attribute name="name" type="xs:string" use="optional"/>
<xs:attribute name="minInstances" type="xs:int" use="optional" default="1"/>
<xs:attribute name="maxInstances" use="optional" default="1">
<xs:simpleType>
<xs:union>
<xs:simpleType>
<xs:restriction base="xs:nonNegativeInteger">
<xs:pattern value="([1-9]+[0-9]*)"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="unbounded"/>
</xs:restriction>
</xs:simpleType>
</xs:union>
</xs:simpleType>
</xs:attribute>
</xs:extension>
</xs:complexContent>
</xs:complexType>
```

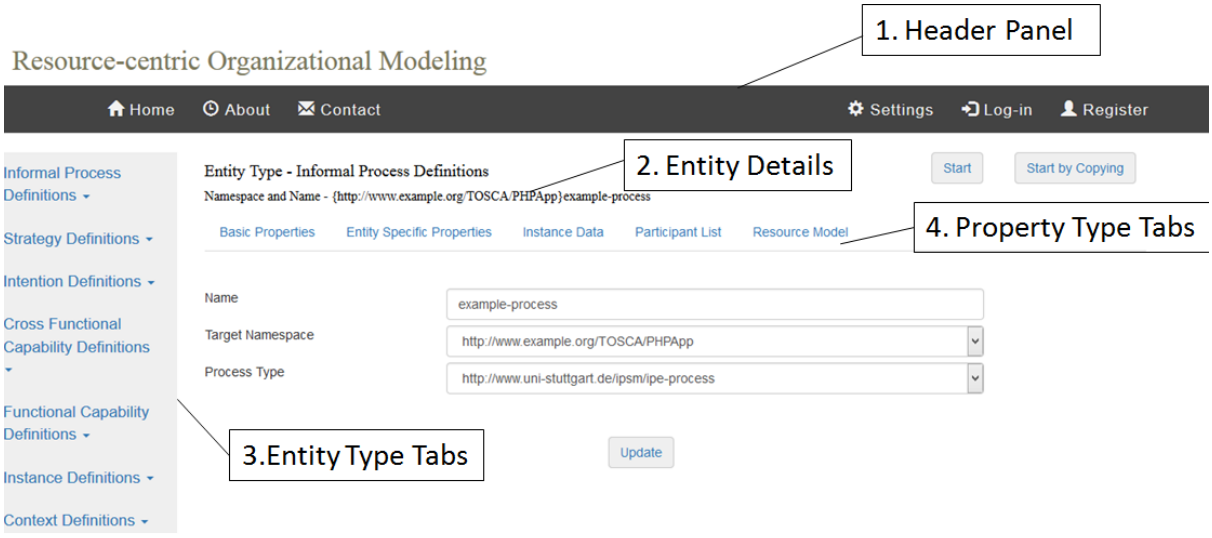

Figure 6.4: Screenshot of Resource Model

informal process models are out of scope of this master thesis. Only the functionalities such as creating instances, extracting instances and editing instances are part of the developed editor. This is because initializing informal process models starts after the initial context defined in an IPE model [SBLW15]. Thus it is important to discuss realization of instance creation which are required for subsequent phases P3 and P4 of Executing Informal Processes. Acquirable entity types' models can be converted into instances. For example, resource definition is converted into *resource instance*. A model instance contains additional meta-data about the executed processes such as the information about the start date and time, end date and time, instance status, cost, source model etc. From the screenshot image 6.5 it is clear that these properties of an instance can be edited through the developed editor. Only when a acquirable model is successfully initialized it can be enagaged to adapt the process execution of emerging requirements [SBLW15]. The properties that describe each instance is provided in the listing 6.3

The developed editor supports creation and updation of descriptive information about instances. Each instance belong to any one of the acquirable entity type such strategies, intentions and informal processes. Any entity that has instances are also listed inside the *Instance data* tab of each entity. From the screenshot image Figure 6.6, it is clear that the editor has ability to add, remove and extract instance descriptors for any entity type. An instance descriptor of a functional capability refers to a resource definition meaning that a capability is provided by a resource definition. So an instance descriptor of a capability refers to a resource definition.

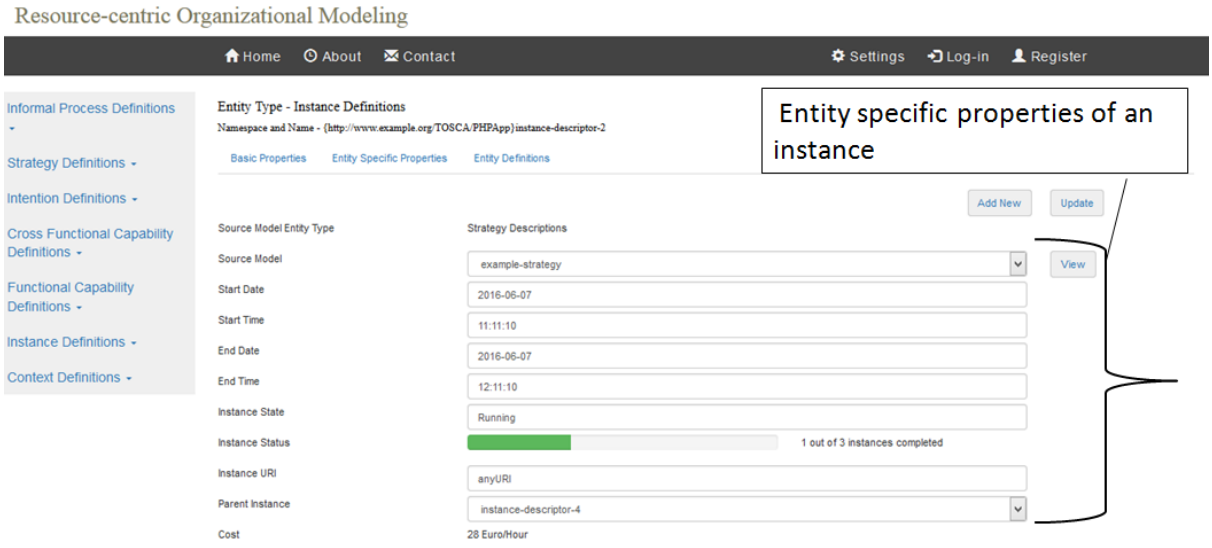

Figure 6.5: Screenshot of Instances(2)

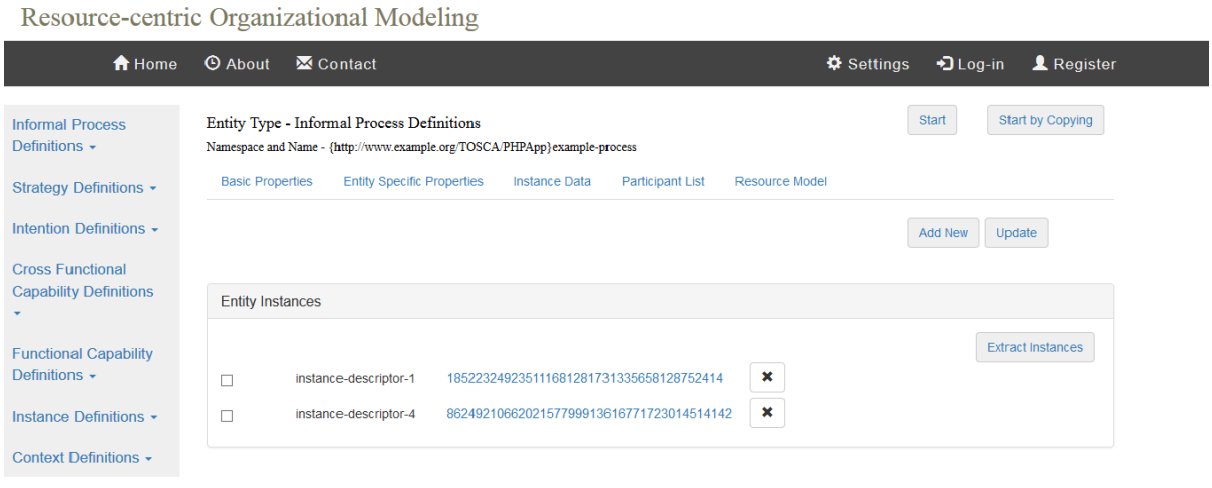

Figure 6.6: Screenshot of Instances(1)

Listing 6.3 XML Schema Definition of Instances

```
<complexType name="tInstanceDescriptor">
<complexContent>
<extension base="ipsm:tIdentifiableEntityDefinition">
<attribute name="startTime" type="time" use="optional"></attribute>
<attribute name="endTime" type="time" use="optional"></attribute>
<attribute name="instanceState" type="string" use="required"></attribute>
<attribute name="instanceURI" type="anyURI" use="optional"></attribute>
<attribute name="sourceModel" type="tAcquireableentitydefinition"
    use="required"></attribute>
<attribute name="parentInstance" type="string" use="optional"></attribute>
<attribute name="id" type="string" use="required"></attribute>
<anyAttribute></anyAttribute>
</extension>
</complexContent>
</complexType>
```

6.4 Validation

This section validates the degree of satisfaction of the research objectives discussed in Chapter 1 against the developed prototype through. Also, we claim that this master thesis is a part of creating models that are required for supporting and automating informal processes, it is important to evaluate the developed prototype along with the requirements that are discussed in the approach *Informal Process Essentials* [SBBL14]. In this section, examples are provided from motivating scenario which is discussed in the Chapter 3. The concept of *resource-centric* modeling approach has also been validated in the approach *Informal Process Essentials* [SBBL14], where the author describes that the approach is right one since the focus is not on business logic rather on other dimensions like resources. The author also states that non-existence of business logic facilitates more autonomy for human performers and enables establishment of best practices. Since the above arguments justifies to the fact of providing more autonomous informal process modeling, one can claim that the approach of *resource-centric modeling* is a valid one. Not stopping with these arguments, we also provide a detailed validation of research objectives discussed in Chapter 1 and validation of developed prototype with suitable examples.

6.4.1 Validation of Research Objectives

As discussed in Chapter 5, the research objectives are satisfied at the design level but their validity can be confirmed only by evaluating the research objectives with some sample scenarios provided in Chapter 3.

Organizational intentions transparency (R1): A valid user whose credentials are stored in database is able to login successfully and view the intentions and its associated entities. Hence the research objective R1 is met.

Organizational intention resource-based cost estimation (R2): An intention whose cost is unspecified for a sample intention, is calculated by the developed system recursively as mentioned in the Chapter 5. Thus the research objective R2 is also met.

Organizational intention achievability estimation (R3): Similar to cost calculation, an intention instance whose achievability is not in prior is also estimated by the current functioning system. Hence research objective R3 is satisifed.

Intention oriented working style (R4): The users can login and create intention models, strategy models, informal process models etc., through the developed editor. Hence research objective R4 is also met.

Participative organizational modeling (R5): Each entity type that can be interactively acquirable has list of participants with their corresponding privileges. Thus this satisfies the requirements of research objective R5.

Re-use of organizational knowledge (R6): The descriptive information about each models can be stored and re-used for next enactments. Hence research objective R6 is also met.

6.4.2 Validation of Prototype

In the approach of *Supporting Informal Processes* [SKL14], the author has categorized generic requirements that supports enactment of informal processes under three dimensions such as business logic, IT infrastructure and organization. In order to make the validation procedure simple, we have taken the concrete requirements discussed in the approach of *Informal Process Essentials* [SBBL14]. This is because the latter approach itself is an extended work of former approach.

Enactable Informal Process Representation: In this requirement, the core elements are performers, IT tools, data etc., and the requirement gets satisfied only when we are able to provide textual descriptions of how to make these ready. In the functioning web editor, the user can create textual information i.e models required for resources, contexts, strategies etc. Hence the developed prototype satisfies the requirement of providing enactable informal process representation. For example, using our motivating scenario we only provide definitions of intentions, contexts, strategies, capabilities and resources inside the editor but there is no functionality to predefine business logic of these informal processes.

Resource Relationships Definition: In an informal process, each resource can have a relation with other resource. For example in our motivating scenario, a resource with front-end developer capability has a "requires" relationship with front-end developer tools. In the functioning system, Winery modeling tool's repository web page has been included to edit the resource models. Thus the functioning system also satisfies the requirement of defining relationship between the resources.

Resource Visibility Definition: Informal processes contains resources that work towards the process' specific intention. These resources can participate in more than one informal process. For example, in the same organization as in our motivating scenario there can be another process working towards achieving an intention say *improving skills of all the employees*. This can make an employee with a developer capability to participate in both the processes. Thus all the resources of an informal process has to be visible. In or functioning system we have septate user interface tab that details associated resources of an informal process. This satisfies the requirement of making the resource definitions visible.

Support for Dynamically Changing Resources: Due to dynamic nature of informal processes, the developed editor provides facility to add or remove resources. For example, in our motivating scenario consider the sub-intention of *improving the help desk portal* where a new requirement of *extending the help desk portal support in mobiles* may arise dynamically. In this situation, the editor must provide means to add new resource with new capability of *mobile application developer capability*. The functioning system provides facility to add or remove resources associated with capabilities. Thus the requirement of providing support for dynamically changing resources is also satisfied by the editor.

7 Conclusion and Future Work

The place of this thesis in the execution steps of informal process is shown in the Figure 7.1. The developed editor serves the purpose of creating descriptive models for informal process and its associated entities. This created models are taken to the next phases of compilation (P3) and execution (P4).

There exists an overhead to configure, coordinate and engage different resources without automation during initialization [SBLW15]. For example in our motivating scenario 3, one of the sub-intention is to improve the help desk for answering consumer queries. One of the strategy to achieve this sub-intention is providing facility to automatically record and answer some of the known basic queries from the consumer. To develop such an automatic help desk software we need different IT services and software developers need to be assigned and tasks has to be initiated. Though existing automation standards such as BPEL suggest to avoid such overhead by acquiring interrelated resources in priori, such complementary concepts of automatic initialization are still missing in this work. Due to high cost of automating in modeling execution steps, in comparison to its less benefits [SBLW15] this work has not provided details of formal definitions like which execution steps has to be taken by which actors. This work of resource-centric

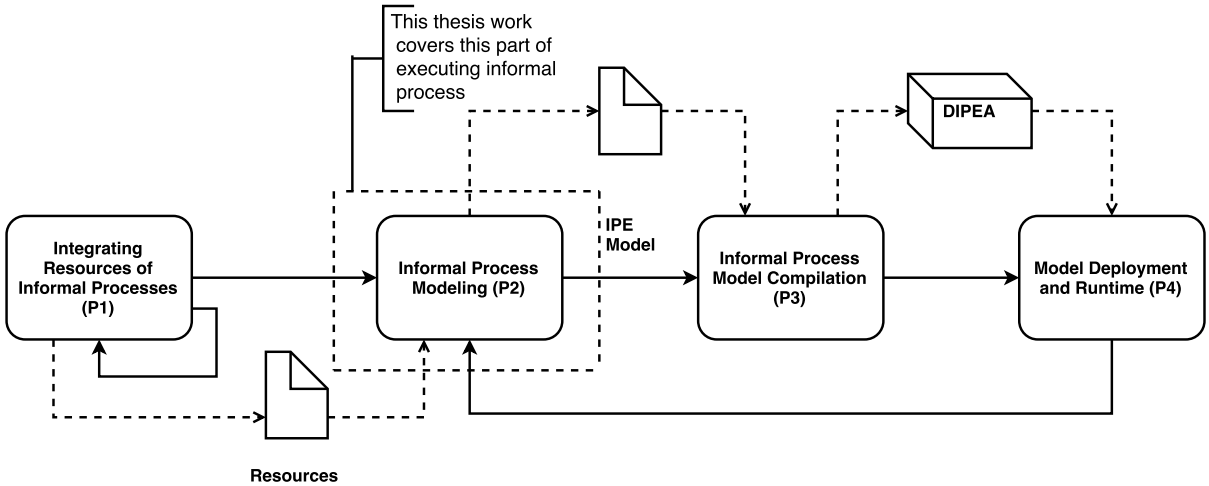

Figure 7.1: Contribution of the thesis work

informal process modeling provides complementary *informal* guides and definitions of intentions of the respective processes.

In this thesis work, we first started Chapter 1 with motivational and problem statement followed by research objectives of this work. In Chapter 2, the fundamental concepts from existing literature has been provided in a detailed way. In Chapter 3, a motivating scenario has been taken and explained based on the guidelines and real life scenarios discussed in some previous work. In Chapter 4, a detailed requirements analysis and literature review from the existing approaches has been provided. This is followed by Chapter 5, which provides detailed description about the methodology and characteristics of the modeling process. A detailed case study has been provided in Chapter 6, which helps to explain the abstract concepts discussed in the earlier chapter in a concrete way. This chapter also validates the developed web –based editor by providing examples that satisfies the research objectives discussed in Chapter 1 and also conformance of the motivating scenario discussed in Chapter 3 with the developed prototype.

Future Work

As discussed in Chapter 1, the developed web based editor developed as part of this master thesis, will be further extended such that it can generate deployable entities from the current descriptive information. These deployable entities will be further developed as compilable and executable entities in phases P3 and P4 of the InProXec[SBLW15]. Also extension of providing mobile support to this web editor are also part of future work.

Each resources can be related with other resources through *relationships*. This helps business experts to create models with logical resource structures. In this thesis work, we have addressed resource models without relationships and left the ones contain relationships as future work. This is due to the fact that relationships are optional entities in each model and also due to the broad context of this work [SBBL14].

Bibliography

[AIvH+15] A. Aldea, M.-E. Iacob, J. van Hillegersberg, D. Quartel, L. Bodenstaff, and H. Franken. “Modelling strategy with ArchiMate.” In: *Proceedings of the 30th Annual ACM Symposium on Applied Computing*. 2015, pp. 1211–1218 (cit. on pp. 41, 44).

[BBKL14] T. Binz, U. Breitenbücher, O. Kopp, and F. Leymann. “TOSCA: portable automated deployment and management of cloud applications.” In: *Advanced Web Services*. Springer, 2014, pp. 527–549 (cit. on p. 66).

[BCV06] S. J. Bleistein, K. Cox, and J. Verner. “Validating strategic alignment of organizational IT requirements using goal modeling and problem diagrams.” In: *Journal of Systems and Software* 79 (2006), pp. 362–378 (cit. on p. 40).

[BFV12] M. Brambilla, P. Fraternali, and C. K. Vaca Ruiz. “Combining Social Web and BPM for Improving Enterprise Performances: The BPM4People Approach to Social BPM.” In: *Proceedings of the 21st International Conference on World Wide Web*. 2012, pp. 223–226 (cit. on p. 41).

[Can15] M. Canko. “Investigating the Virtual Representation of Human Resources.” Diploma Thesis. University of Stuttgart, Faculty of Computer Science, Electrical Engineering, and Information Technology, Germany, July 2015, p. 83 (cit. on p. 24).

[CH09] D. Cohn and R. Hull. “Business artifacts: A data-centric approach to modeling business operations and processes.” In: *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering* 32 (2009), pp. 3–9 (cit. on pp. 41, 43).

[FKSS13] A. Fleischmann, U. Kannengiesser, W. Schmidt, and C. Stary. “Subject-oriented modeling and execution of multi-agent business processes.” In: *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 02*. 2013, pp. 138–145 (cit. on pp. 41, 44).

[Gro11] O. M. Group. *Business Process Model and Notation (BPMN) 2.0*. [Online; accessed 25-November-2015]. 2011. URL: http://www.omg.org/spec/BPMN/2.0/PDF/ (cit. on p. 19).

[Gro13] T. O. Group. *Archimate 2.1*. [Online; accessed 25-November-2015]. 2013. URL: http://pubs.opengroup.org/architecture/archimate-doc/ts_archimate/ (cit. on p. 19).

[HK11] C. Herrmann and M. Kurz. "Adaptive case management: supporting knowledge intensive processes with IT systems." In: *S-BPM ONE-Learning by Doing-Doing by Learning*. Springer, 2011, pp. 80–97 (cit. on p. 45).

[KBBL13] O. Kopp, T. Binz, U. Breitenbücher, and F. Leymann. "Winery–a modeling tool for TOSCA-based cloud applications." In: *Service-Oriented Computing*. Springer, 2013, pp. 700–704 (cit. on pp. 18, 66).

[Lac16] T. Lacoma. *What Is Goal Orientation?* [Online; accessed 18-November-2015]. 2016. URL: http://smallbusiness.chron.com/goal-orientation-20360.html (cit. on p. 41).

[Ley10] F. Leymann. "BPEL vs. BPMN 2.0: Should you care?" In: *International Workshop on Business Process Modeling Notation*. 2010, pp. 8–13 (cit. on p. 18).

[MBH+10] V. Mandić, V. Basili, L. Harjumaa, M. Oivo, and J. Markkula. "Utilizing GQM+ Strategies for business value analysis: An approach for evaluating business goals." In: *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*. 2010, p. 20 (cit. on p. 40).

[MHL+07] T. McManus, Y. Holtzman, H. Lazarus, J. Anderberg, E. Berggren, and R. Bernshteyn. "Organizational transparency drives company performance." In: *Journal of Management Development* 26 (2007), pp. 411–417 (cit. on p. 39).

[Moo09] D. L. Moody. "The "physics" of notations: toward a scientific basis for constructing visual notations in software engineering." In: *Software Engineering, IEEE Transactions on* 35 (2009), pp. 756–779 (cit. on p. 19).

[MWM+12] T. Matthews, S. Whittaker, T. P. Moran, S. Y. Helsley, and T. K. Judge. "Productive interrelationships between collaborative groups ease the challenges of dynamic and multi-teaming." In: *Computer Supported Cooperative Work (CSCW)* 21 (2012), pp. 371–396 (cit. on p. 23).

[MWMY11] T. Matthews, S. Whittaker, T. Moran, and S. Yuen. "Collaboration Personas: A New Approach to Designing Workplace Collaboration Tools." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2011, pp. 2247–2256 (cit. on pp. 13, 23).

[Ora16] Oracle. *MVC Architecture*. [Online; accessed 2016-05-03]. 2016. URL: https://docs.oracle.com/cd/E13174_01/alui/devdoc/docs60/Overview_of_the_Portal_Architecture/Portal_UI/PlumtreeDevDoc_Overview_MVCArchitecture.htm (cit. on p. 49).

[Rec10] J. Recker. "Opportunities and constraints: the current struggle with BPMN." In: *Business Process Management Journal* 16 (2010), pp. 181–201 (cit. on p. 17).

[Rei06] H. A. Reijers. "Implementing BPM systems: the role of process orientation." In: *Business Process Management Journal* 12 (2006), pp. 389–409 (cit. on pp. 41, 43).

[SAP09] A. SAP. *BPM Technology Taxonomy: A Guided Tour to the Application of BPM*. [Online; accessed 2016-06-28]. 2009. URL: http://www.evolvedmedia.com/wp-content/uploads/2014/10/Evolved-Technologist_SAP-and-Accenture_BPM-Technology-Taxonomy-A-Guided-Tour-to-the-Application-of-BPM_White-Paper_2009.pdf (cit. on p. 17).

[SBBL14] C. T. Sungur, T. Binz, U. Breitenbücher, and F. Leymann. "Informal Process Essentials." In: *Proceedings of the 18th IEEE Enterprise Distributed Object Conference (EDOC 2014)*. 2014, pp. 200–209 (cit. on pp. 12, 13, 18, 21–24, 31, 37, 55, 59, 62, 66, 70, 71, 74).

[SBLW15] C. T. Sungur, U. Breitenbücher, F. Leymann, and J. Wettinger. "Executing informal processes." In: *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services*. 2015, p. 54 (cit. on pp. 12–14, 17, 18, 25, 26, 36, 45, 56, 66, 68, 73, 74).

[SBLW16] C. T. Sungur, U. Breitenbücher, F. Leymann, and M. Wieland. "Context-sensitive Adaptive Production Processes." In: *Procedia CIRP* 41 (2016), pp. 147–152 (cit. on pp. 34, 51).

[SDDL15] C. T. Sungur, C. Dorn, S. Dustdar, and F. Leymann. "Transforming Collaboration Structures into Deployable Informal Processes." In: *Proceedings of the 15th International Conference, ICWE 2015, Rotterdam, The Netherlands, June 23-26, 2015*. 2015, pp. 231–250 (cit. on p. 54).

[SGHZ12] J. Stirna, J. Grabis, M. Henkel, and J. Zdravkovic. "Capability driven development–an approach to support evolving organizations." In: *The Practice of Enterprise Modeling*. Springer, 2012, pp. 117–131 (cit. on pp. 19, 41, 44).

[Sie15] S. C. P. Sierra. "Investigating Informal Processes." Diploma Thesis. Tilburg University, University of Stuttgart, University of Crete, June 2015, p. 85 (cit. on pp. 19, 31, 35).

[SKL14] C. T. Sungur, O. Kopp, and F. Leymann. “Supporting Informal Processes.” In: *The 6th Central European Workshop on Services and their Composition (ZEUS 2014)*. 2014, pp. 49–56 (cit. on pp. 11, 12, 23, 62, 71).

[Wes12] M. Weske. *Business process management: concepts, languages, architectures*. Springer Science & Business Media, 2012 (cit. on pp. 17, 18).

[YHP00] B. Yu, J. Harding, and K. Popplewell. “A reusable enterprise model.” In: *International Journal of Operations & Production Management* 20 (2000), pp. 50–69 (cit. on p. 41).

[YMMS09] S. Yarosh, T. Matthews, T. P. Moran, and B. Smith. “What is an activity? Appropriating an activity-centric system.” In: *Human-Computer Interaction–INTERACT 2009*. Springer, 2009, pp. 582–595 (cit. on pp. 41, 43).

All links were last followed on June 29, 2016.

Declaration

I hereby declare that the work presented in this thesis is entirely my own. I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

_____

place,date,signature