Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Telecom Engineering Telecom Information Systems stream

Research thesis

On

An Integration Pattern Selection Framework for ethio telecom Enterprise Systems

By:

Thomas Abebe (GSR/3025/08)

Advisor:

Mesfin Kifle (PhD.)

November, 2018

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Telecom Engineering Telecom Information Systems stream

Research thesis

On

An Integration Pattern Selection Framework for ethio telecom Enterprise Systems

By: **Thomas Abebe Gulilat**

Approval by Board of Examiners

---------- ----------

Chairman, School of Graduate Committee

Mesfin Kifle (**PhD**.) ----------

Advisor Signature

Dr. Yihenew Wonde ----------

Internal Examiner Signature

Dr. Surafel Lemma ----------

Internal Examiner Signature

Abstract

Enterprise Application Integration (EAI) as a solution combines processes, standards, software and hardware for the seamless integration of different enterprise systems in order to operate and function as one. The success of EAI deployment is highly dependent on a proper selection of EAI solution patterns. ethio telecom's enterprise systems are integrated and interoperated using EAI techniques. However, many organizations including ethio telecom, have the difficulty of selecting proper EAI solution patterns. Thus, several EAI solution pattern projects fail to deliver the service that the business is expecting.

The objective of this thesis is to develop an integration pattern selection framework, which is used for the selection of an appropriate EAI solution pattern from the available real world's design decisions. The research therefore studied the company's end-to-end enterprise systems' business scenario, by conducting interviews and group discussions with selected staffs, and also referred world's scientific related works and literatures. Case study for validating the framework is also implemented.

The proposed framework has the basic activities like: describing business process requirements for integration, studding the environments' development context and integration purpose, examining the integration data and interfaces, modeling of data, mapping of those models, defining of EAI requirements specification, and finally selection of optimal EAI solution pattern, from the available EAI solutions' patterns based on studied evaluation criteria.

The framework has been evaluated using three different case studies; by taking ethio telecom's existing integrated enterprise systems and also the forthcoming integration technology parameters. It is then reviewed and evaluated by the internal experts of the company.

The main contribution of this research is: to support the company in choosing a suitable EAI solution pattern accommodating both technical and organizational measures, and to increase its' level of success by implementing appropriate EAI solution, which avoids some drawbacks between packaged and third party systems in relation to interoperability, and flexibility issues.

Keywords: - EAI, EAI Integration, Enterprise Systems, EAI Solution Patterns

Acknowledgement

First and foremost, I give thanks to God for his absolute protection and ability to do this work.

I am grateful to my company ethio telecom, and Addis Ababa University's School of Electrical and Computer Engineering, Graduate Program in Telecom Engineering, for making it possible for me to study here.

I would also like to express my sincere gratitude to my thesis advisor Dr. Mesfin Kifle, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of my research and writing of this thesis.

Besides my advisor, I would also like to thank my thesis examiners: Dr Yihenew Wonde, and Dr. Surafel Lemma, for their encouragement, insightful comments, and hard questions.

In addition, I like to thank my company experts who were involved in giving me valuable comments, feedbacks, and participated on the validation survey of this thesis; especially: Mr. Hassen, W/ro Etagegn, and Ato Yonas. Without their participation and input, the validation survey could not have been successfully conducted.

Finally, I must express my very profound gratitude to my parents: Ato Abebe Gulilat, W/ro Yimegnushal Abebe, and my sister Frehiwot; for providing me with unfailing support and affection, and also continuous encouragement throughout my years of study. This accomplishment would not have been possible without them.

May the almighty God richly bless all of you!

Table of Contents

List of Tables

List of Figures

Acronyms

API	Application Programming Interface
BSS	Business Support Systems
BSC	Base Station Controller
BTS	Base Transceiver Stations
CBS	Convergent Billing Systems
CDR	Call Detail Records
CRM	Customer Relationship Management
DT-IMC	Dynamic Tariff-Inter Media Center
EAI	Enterprise Application Integration
EAM	Enterprise Architecture Models
eCAF	Electronic Customer Acquisition Form
EFD	Enterprise Function Diagram
ERP	Enterprise Resource Planning
ESB	Enterprise Service Bus
ETL	Extract Transform and Load
GTP	Great Transformation Project
HLD	High Level Design
HTTP	Hypertext Transfer Protocol
ICD	Interface Control Documents
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IPCC	Internet Protocol Contact Center
ISP	Internet Service Provider
IT	Information Technology
IVR	Interactive Voice Recorder
JVM	Java Virtual Machine
LLD	Low Level Design
MMS	Multimedia Messaging Service

mVAS	Multi Value Added Services
NGBSS	Next generation Business Support systems
NMS	Network Management Systems
OCG	Online Charging Gateway
PDD	Processes Deliverable Diagram
RFQ	Request for Quotation
RPI	Remote Process Invocation
SAP	Systems Applications Products
SCM	Supply Chain Management
SDN	Software Defined Networks
SFTP	Secure File Transfer Protocol
SME	Situational Method Engineering
SMS	Short Message Service
SOA	Service-Oriented Architecture
SoS	System-of-Systems
TEP	Telecom Expansion Project
UML	Unified Modeling Languages
UVC	Universal Voucher Center
USSD	Unstructured Supplementary Service Data

Chapter One: - Introduction

1.1 Background

Nowadays, most companies heavily rely on technologies that are of different type and on enterprise systems which are running in different business units and technologies (i.e. platform, infrastructure, language, etc.).

In this competitive business environment, to shorten the applications development life cycle and reduce the financial cost, the reuse of existing products (i.e. database and applications) is a first choice instead of renovating or doing them from scratch.

In order for companies to provide different and diversified services with high quality and gain maximum profit out of it, its business processes should also be supported by enterprise systems that are integrated and interoperated using Enterprise Application Integration (EAI) techniques.

The company, ethio telecom, is a public enterprise providing different types of services such as: voice, Short Message Service (SMS), and data services, for more than fifty million mobile service subscribers and more than fourteen million data service subscribers using its diversified and heterogeneous enterprise systems [1]. For that its enterprise systems needs to be integrated using a method of EAI techniques and EAI solution patterns.

EAI is a solution, which is a combination of processes, software, standards, and hardware resulting in a seamless integration of two or more enterprise systems allowing them to operate as one [2]. It is a technology-business philosophy where all enterprise systems can share information and business logic in an easiest way [3].

EAI applications function by coordinating standalone and independent applications that can run by themselves in the enterprise landscape in a loosely coupled manner.

EAI solution patterns are collections of design decisions that supports the integration of different enterprise systems [4]. EAI patterns also reflect the experience of the broad community of software engineers over decades of building complex systems.

However, the selection of EAI patterns as a solution is a very critical decision for an integration architect that could make some of the systems' data, control of resources and business process objectives easier or harder to achieve. Also, the selected and implemented EAI solution patterns

have a negative or positive impact (i.e. related to maintenance cost and/or integration time) on the environmental context of the company.

One of the basic reason for the failure of most EAI projects, is not knowing **how** to approach the application integration problems and not selecting and adopting proper EAI patterns based on a framework used for the evaluation of those technical solutions [4].

Therefore, selecting an optimal EAI solution patterns based on a predesigned selection mechanism, by considering the existing environmental context of the company, and also in view of the forthcoming systems is very mandatory.

1.1.1 Enterprise Systems

Enterprise systems are commercial software products or packages that enables the integration of transactions-oriented data and business processes throughout an organization and perhaps eventually throughout an entire inter-organizational chain [7]. Also, an enterprise system is a packaged application that supports and automates business processes and manages business data [5]. They also come with customizable modules that reflect best practice for common business operations.

Some of the world's enterprise systems includes one or more of the applications, such as: Enterprise Resource Planning (ERP), Supply Chain Management (SCM), Customer Relationship Management (CRM) systems, etc.

Basic characteristics of Enterprise systems are:

- **Integration**: Allows a seamless integration of information flow throughout a company, such as: financial and accounting information, human resource information, supply chain information, and customer information.
- **Packages**: Are commercial software packages or sub-modules that are purchased or leased from software vendors rather than being developed in-house from scratch.
- **Best Practices:** They are designed to fit the needs of many organizations. Enterprise systems are built to support generic business processes that may differ quite substantially from the way any particular organization does business [7].

- **Evolving**: Like all of Information Technology (IT), enterprise systems are rapidly changing. They are changing architecturally from the mainframe to todays designed for client-server architectures. The functionality of enterprise systems is also evolving [7].

1.2 Motivations for the Research

ethio telecom currently has numerous enterprise systems. It provides services to its customers by attaining the business process logics of the company, using EAI solutions deployed by vendors [11]. It could also add many upcoming systems when the need arises and based on the evolution of the technology.

Thus, not deploying EAI solution patterns by considering the current environmental context of the company and the forthcoming systems and technologies, would ultimately lead to project failure and high loss of cost.

It is also noticed that there is also a knowledge gap on the area of EAI solution patterns [11].

The motive of this research is therefore to close or minimize this knowledge gap on the area of EAI solution patterns selection mechanisms.

Knowledge gap: The Company has already deployed EAI solution. But, it does not have sufficient integration architects and professionals that could analyze the current and upcoming environmental situations and propose a mechanism to bring a suitable EAI solution pattern.

1.2.1 Enterprise Systems in ethio telecom

In ethio telecom, there are around 180 applications that are categorized into business, support, corporate and office tools [11]. These applications shall also be grouped into a form of some similar behavior or use class. One of the grouping of applications would be Convergent Billing System (CBS), Customer Relationship Management (CRM), Multi Value Added Services (mVAS), Internet Service Provider (ISP), Enterprise Resource Planning (ERP), Revenue Assurance (RA), Electronic Customer Acquisition Form (e-CAF), Internet Protocol Contact Center (IPCC), Security, and Others.

Some of the enterprise systems in the company with their few descriptions are stated below:

CBS: Is a new generation convergent charging solution developed from online charging system. CBS inherits large capacity, high performance, and highly reliable intelligent network and provides a unified data model that incorporates customers, subscribers, and accounts.

CRM: Is also another Next generation Business Support systems (NGBSS) which is used to carry out customer management and sales related tasks.

ERP: It is a business process management software that allows the company to use a system of integrated applications to manage the business and automate many back office functions related to technology, services and human resources. Some modules of the ERP system are Oracle General Ledger, Accounts Payables, Account Receivables, Fixed Assets, Cash Management, Human Resources (HR) and Facility modules, etc. It has also integration with other external NGBSS systems.

Unified Voucher Card (UVC): used to store voucher cards related data and handles the recharge request from subscriber.

1.3 Statement of the Problem

The EAI Consortium states that approximately 70% of all EAI projects fail in some way; where failure is rated as missing deadlines, blowing budgets or failing to deliver the service that the business was expecting [30].

The most common reason for failed EAI projects and solutions is related to management, such as: inappropriate decision by managerial staff, gap in knowledge, and lack of project managers. Most of this management issues could be avoided when organizations know **how** to approach such application-integration problems properly.

In addition, many of the organizations' integration landscape are also in need of the looming over a new forces of enterprises' IT, which are: cloud, mobile, big data and social [29].

Hence, a framework that could assist organizations in the selection and evaluation of technical solutions could positively influence the adoption of the EAI solutions within the company [30].

ethio telecom has deployed Service Oriented Architecture (SOA) based EAI solution pattern by a vendor, not considering the forthcoming technology integration needs, and not selecting out of other EAI solution patterns by means of selection mechanisms [11].

Some observed effects due to this are:-

- Inability to easily orchestrate among enterprise systems of different vendors.
- Increase in maintenance service cost and response time on integrated enterprise systems.

- Lack of considering the parameters of the upcoming systems like: cloud-computing, multimedia streams, big data, Internet of Things (IoT), etc.
- Generally, failure of many EAI projects because of not applying a suitable EAI solution pattern [6].

Thus, not having context specific framework that recognizes and considers the company's current systems and the forthcoming technologies and services, in selecting an optimal EAI solution patterns is the basic problem.

Since, the company's environment is different from other world's scenario; this research develops a selection framework by observing its existing and upcoming enterprise systems. So, this framework minimizes the gap in knowledge and ultimately contributes in getting an optimal EAI solution pattern by exploiting orchestration among systems.

1.4 Research Questions

Having the above stated problem in mind, the below research question is formulated:

How could ethio telecom systematically select an optimal EAI solution pattern?

In a process the following sub-questions would also been answered.

Q1: What is the current practice of ethio telecom in selecting systems integration pattern?

Q2: What selection-criteria and characteristics for EAI solution pattern are identified in literatures and how can the existing EAI integration pattern selection methods be used and adopted for the company's selection framework?

Q3: What method engineering approach should be used in order to construct the selection of EAI solutions' pattern framework?

1.5 Objective of the Research

1.5.1 General Objective

This research is intended to develop a framework for the selection of EAI integration solution pattern, accommodating the existing and upcoming enterprise systems from the available real world design decisions.

1.5.2 Specific Objective

The following specific objectives are formulated to realize the general objective.

- To understand the data, business processes, and needed infrastructure of the enterprise systems' for integration.
- To identify best practices, tools, methods and techniques used in the area for the selection of EAI solution patterns.
- To design a way in defining requirements for the selection of application integration design decisions.
- To develop a framework for selecting an integration solution pattern among different EAI solutions. And,
- To evaluate the proposed EAI integration solution pattern selection framework.

1.6 Research Method

To achieve the general and specific objectives of the research, different tools, methods and techniques are used to gather and analyze data, develop, and evaluate the framework.

1.6.1 Method of Research's Design Process

Based on the research's objective, the research's design process is prepared and shown in Figure 1.It is a conceptual model displaying the processes performed throughout the study.

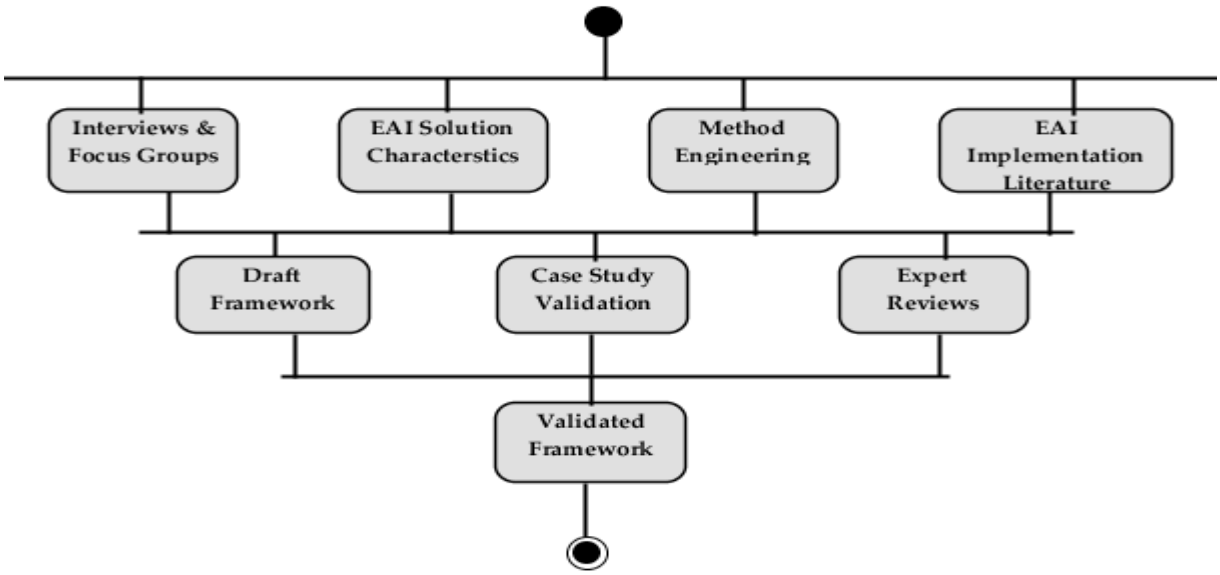


Figure -1- Research's Conceptual Model

1.6.2 Method of Data Collection

To gather information on company's current EAI solution's practice, and to gain knowledge of the real world's current trend on EAI solution selection methods, the below data gathering techniques have been used.

A- Literature review:

Various literatures, such as: Institute of Electrical and Electronics Engineers (IEEE) papers, journals, conference papers, white papers and books on the area are studied and reviewed to understand the state-of-the-art and gain knowledge on how others have seen the problem and gone through it .

B- Document review

Company documents that are accessible and non-confidential, are reviewed to know the techniques used for selecting the existing EAI solution. In addition, Interface Control Documents (ICD), high and low level design documents are also reviewed to have an understanding of the requirements and parameters on existing systems integration.

C- Interviews and Questioners

Some officers, project managers, technical and integration design staffs are interviewed and questioners are also done for gathering information on EAI requirements, to get feedbacks on the framework.

1.6.3 Method of Developing Techniques

The company's different enterprise systems' data, business processes, and applications for integration solution including the infrastructure are analyzed, modeled and mapped to describe the existing systems scenario and identify the gap on integration using different modeling tools and techniques such as: Unified Modeling Languages (UML), flowcharts, data flows, business process modeling.

Also, situational method engineering technique, meta-modeling technique and Enterprise Architecture Models (EAM) are used for designing the framework and business processes. Software tools such as: Rational rose and/or Visual Paradigm, EAM tools, etc. are also used.

1.6.4 Method of Validating the Framework

The selection framework is validated using a case study done at ethio telecom and also using internal expert reviews by integration design professionals, project and/or application officers of the company.

As per [18], the following quality criteria for validating the framework should be taken into consideration: *Completeness*, *Consistency*, *Efficiency*, *Reliability*, and *Applicability*.

A- Case Study

When designing the case study validations, the below design tests are described, which are also adopted for this research [19]:

- **Construct validity**: To validate as per the context of this research. A way of establishing that the correct operational measures are taken for the concepts being studied.
- **Completeness**: validates that the activities found on the framework are complete in bringing the intended EAI solution patterns.
- **Reliability**: Describes that the operations of a study such as data collection procedures can be repeated with the same results. Meaning, the case study should be performed exactly to the framework's specification [19].

B- Expert Review

Expert review is done by the company's integration design professionals and also selected design and application officers. The number of experts that are involved in the expert review are three individuals.

1.7 Scope and Limitation

1.7.1 Scope of the Research

The scope of this research is to develop an integration pattern selection framework for ethio telecom; considering the company's current enterprise systems, legacy or custom applications, and basic requirements of the upcoming and future enterprise technologies.

But, this research does not cover studying and analyzing the evolution of the world's existing EAI solution patterns and integration frameworks.

1.7.2 Limitation of the Research

Due to time constraints, the framework's validation using case study is done only on the company's systems. Expert review is also performed by the company's internal design experts and related managerial staff. Hence, the framework has not been reviewed by external experts.

1.8 Contribution of the Research

This thesis provides the company with an explicitly specified approach to select an optimal enterprise systems application integration solution pattern. Besides, gaining an understanding in the process of selecting new application integration solution, it would also help the company in avoiding some of the pitfalls that are involved in the selection of these EAI solutions patterns.

1.9 Document Layout

The remaining of this research paper is organized in the following manner:

Chapter **Two** describes the concepts on the definition and overview of EAI (i.e. what an EAI solution and EAI solution patterns means), the general category measures taken for EAI, the technical category characteristics and selection criteria for EAI solution patterns, and the EAI's organizational category measures for the selection framework.

Chapter **Three** discusses the related works and literatures reviewed in order for understanding the state-of-the-art on EAI solution patterns and selection parameters, the company's determining factors (i.e. basic guiding principles, chosen parameters) for the framework, the type of method engineering and modeling technique chosen to develop the framework and the validity measures or quality to be achieved by the framework.

Chapter **Four** elaborates the new developed EAI solution patterns selection framework. Also, the selection framework parts or fragments (i.e. framework's sub-processes, activities, and deliveries) in detail.

Chapter **Five** describes the validation activities done using the framework by taking selected case studies and presents the results of findings through discussions, and finally explains evaluations done and reviewed by internal experts.

Lastly, Chapter **Six** clarifies the research's conclusion, the research's limitations, and future research areas.

Chapter Two: - Theoretical Background

2.1 Overview

This chapter describes different types of concepts on EAI. It defines what EAI means, it discusses on the solutions and patterns for EAI, and it illustrates the general category measures and technical category characteristics and selection criteria for EAI solution patterns. Finally, it describes the considerations on organizational category measures for EAI solution patterns.

2.2 Definition and Overview of EAI

Several definitions for EAI are found during the literature study. Five of the referred definitions from different literatures and publications are listed below:

- EAI is the use of hardware and software to integrate a set of enterprise computer applications. It also says that EAI is a collection of technologies and approaches to provide short-term and cost-effective solution [3].
- EAI is the combination of processes, software, standards and hardware, resulting in a seamless integration of two or more enterprise systems allowing them to operate as one [2].
- EAI is a collection of methods, tools and services that work together to bring heterogeneous applications into communication as part of the traditional, distributed or extended enterprise. Here, the important thing to note is that not only data is shared by applications but it can also share some of their functionalities with other applications [8].
- The book written by Linthicum D. describes, EAI is the unrestricted sharing of information between two or more enterprise applications. A set of technologies that allows the movement and exchange of information between different applications and business processes within and between organizations [9].
- EAI works as a connecting interface between different applications running in an enterprise. It is a solution which is responsible to enhance, combine and communicate among numbers of applications running in an enterprise [10].
- EAI, is an Intra-organizational systems integration, which is the interoperation of packaged and custom (legacy) applications [16].

Therefore, realizing the above definitions of EAI and based on the context of this research, the below EAI definition is chosen.

"The unrestricted sharing of information between two or more enterprise applications. A set of technologies that allows the movement and exchange of information between different applications and business processes within and between organizations [9].**"**

This definition is chosen as it accommodates all the basic pillars for integration, which are: technology, applications, information, business processes, sharing and organizations.

Figure 2 shows the combination and sharing of information between systems (i.e. custom and e-commerce applications, legacy applications, and web objects), databases and client/server applications using EAI method as a means.

Figure -2- Enterprise Application Integration (EAI) - [8]

2.3 EAI Solution Patterns and Integration Frameworks

Most organizational systems are based on new and old technologies and use a broad range of platforms, databases and applications, and also are based on different programming languages. According to [3], Forester Research estimates that up to 35 % of development time and 30 % of development cost, are spent on creating interfaces and points of integration for applications and data sources.

Thus, EAI is a solution, which is a collection of technologies and approaches to provide short-term and cost-effective solution. It is a technology business philosophy where all enterprise systems can share information and logic in an easiest way.

System-of-Systems are set of systems that are cooperating and interoperating while also simultaneously working as independent entities (i.e. not only as parts of the integrated system). They are ways of interoperability and optimization of systems that enhances workability and performance in future cooperative works [4].

Therefore, one of the major aspects of EAI, is to make integration among heterogeneous systems that would ultimately constitute SoS. In other ways round, it also implies that there is no SoS formation without having individual systems. In general, SoS is formulated by creating a framework to integrate different constituent systems.

EAI solution patterns reflect the experience of the broad community of software engineers over decades of building complex systems [4]. The motivations for creating them and the forces involved in using them are seldom clearly explained. Hence, EAI patterns are collections of design decisions. They are generally used for integrating various enterprise systems.

EAI solution patterns could also be identified based on four categories of properties that form the parameters and describe those patterns in a more specific way [28]. Those categories of properties representing the parameters of the EAI solution patterns are shown in Figure 3.

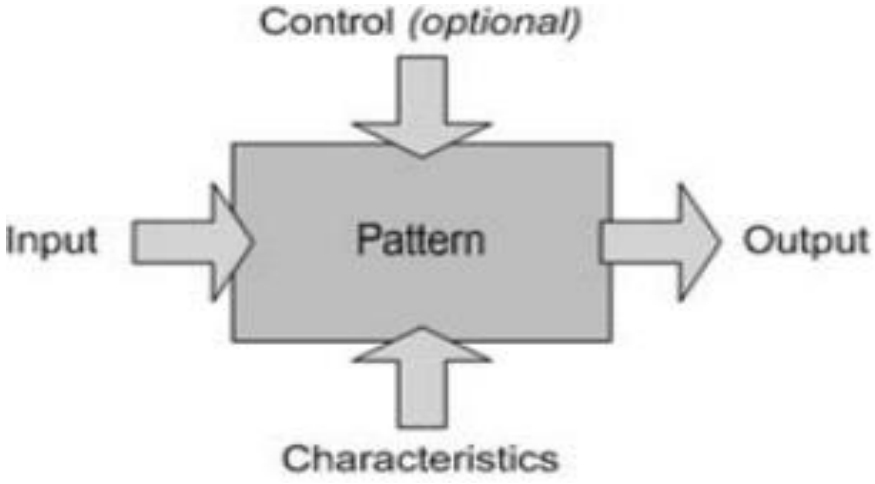


Figure -3- General representation for EAI Patterns Parameters - [28]

Where:-

- ***Input***: Describes the amount of incoming messages and/or request data.
- ***Output***: Tells the amount of outgoing messages and/or data.
- ***Characteristics***: It is the main kind of category. It describes the detailed behavior of EAI patterns; since different choices and settings would lead to different characteristics of the platform specific code. And,
- ***Control***: Tells the format of control messages and data that would dynamically lead to a new behavior of patterns.

Different collections of EAI solution patterns are described by various literature papers. However, they are not meant to be complete. Based in [4], some of the main integration patterns are: Systems Applications Products (SAP3) integration patterns, Pattern-Oriented Software Architecture, Messaging, Salesforce, Patterns from enterprise application architecture, etc. The Salesforce Integration Pattern includes patterns like: Remote Process Invocation (RPI), User Interface (UI) Update based, and Batch Data Synchronization. The SAP3 includes patterns such as: Service-Oriented Architecture (SOA), Peer-to-Peer (P2P), Publish-Subscribe, etc. [12]. The Messaging integration pattern, includes patterns like: Pipes and Filters, Dynamic Router and Canonical Data Model [4]. Figure 4 shows one out of the different types of the EAI solution patterns (i.e. Pipes and Filters integration pattern), categorized in the Messaging integration pattern. This style divides very large processing tasks in to smaller sequence of process steps (filters) which are connected by channels (pipes).

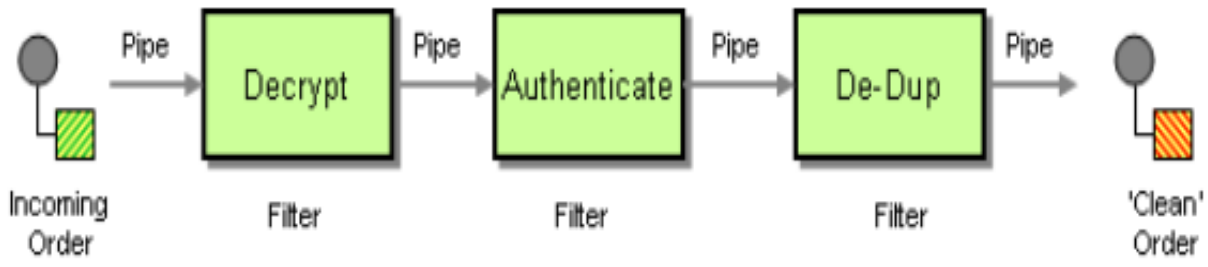


Figure -4- EAI Messaging's Pipe and Filter Integration Pattern - [17]

The SAP3's of SOA solution pattern is a natural and very popular solution for many modern SoS integration. It is organized in a loosely coupled of distributed services which is implemented by

service providers [4]. These different services could be given by different languages, in different platforms, and by different organizations. The interaction of services could be mediated by ESB routing services among service consumers and providers. SOA, provides registry so that services can learn of each other's existence.

In addition, service consumers and producers could also be unaware of each other's existence which make this pattern well-suited to support interoperability and systems evolution. It also includes an orchestration engine supporting the creation and execution of complex workflows and services.

Currently, SOA solution pattern based on web-services, is also a solution implemented in the company's integration of systems; but is highly customized by the solution's provider.

The Salesforce integration patterns, having different pattern types (like: RPI's Request-Replay and Batch Data Synchronization), if implemented properly, enables to get to production as fast as possible [12]. Based on the paper, Salesforce patterns are also most stable, scalable and having maintenance free set of applications. Salesforce's consulting architects have also been actively engaged in maintaining and improving the patterns frequently. Thus, very recent and also up-to-date pattern update releases are available for the industry.

Also, from the open source communities, a couple of integration frameworks have been released that can be placed in the EAI domain which help build EAI solutions [32]. Consequently, open sources' integration frameworks are on one side and the EAI solution patterns are on the other side.

The currently available open source integration frameworks such as: Mule, Apache Camel, Spring, and Guarana , supports the enterprise integration architects with a very broad variety of features in implementing even complex integration patterns [32]. These integration frameworks could also be used in almost every integration projects within the Java Virtual Machine (JVM) environment; no matter which technologies, transport protocols, and/or data formats are used. Hence, when choosing SoS wide integration solution pattern or integration framework, an integration architect must be clear on what form of integration is going to be achieved. That means, on top of describing the general categories like: data/control points, development

context and purpose of integration, a more detailed technical characteristics and classifications of enterprise systems should also be examined and studied.

2.4 General Category measures of EAI

Category measures are ways or central and focal points, used for the study and selection of SoS solution patterns and integration frameworks; when integrating enterprise systems and/or applications. This section defines the general category measures which are needed for EAI solution patterns selection. These general category measures are: the integration scope, understanding the SoS development context, and purpose of integration [4].

2.4.1 SoS Scope

Scope is a fundamental decision that determines whether preexisting decisions regarding the SoS design principles must be taken into account or whether the company is in a situation to actively make such decisions [4]. For instance, a new SoS is usually not developed without first establishing systems in this context, so the decisions are actually about both the SoS and individual systems.

When creating and having individual systems there were decisions or pillars for the formation. That means, how and what decisions should be adopted from the individual systems or what new decision choices for the SoS is taken, and does those choices have impact on individual systems, etc. should be considered.

2.4.2 Development Context

Development context is an important step in knowing the company's systems environmental settings in which it is developed [4]. These three categories of development contexts are:

- **Greenfield**: Implies that there are no preexisting implementations that restrict the design space. By this implementation it means that a complete EAI solution needs to be newly constructed and thus, there are no architectural constraints besides those that are a consequence of the SoS purpose. For example, creating a new SoS without the need to take legacy or custom applications in to account.
- **Brownfield**: Indicates that implementations or systems exist, but in principle their realization could be modified. All existing EAI solutions could be adopted. In particular a

common backbone (middleware) could be introduced or replaced [4]. For instance, adapting an existing system such that it can be integrated into SoS. And,

- **Closed Source**: Shows that an implementation already exists, but not having an access to change it. This implies that external adapters can be used, if the existing implementations do not already provide the required integration facilities [4].

2.4.3 Purpose of Integration

The purpose of integration among systems should be clarified [4]. The identified ranges of potential purposes are described below:

- **One-Directional Information Exchange (Inform)**: One system needs to provide information to one or more systems about what is going on, so that others have the appropriate interpretation of the operational context.
- **Two-Directional Information Exchange (Sync)**: Two systems exchange information in order to keep each other in sync. Here, there is no clear provider-consumer relation among systems.
- **Control**: The way that one system controls the other (again the direction may change over time or different aspects might be controlled in different directions). Unlike information exchange, here the sending system already determines how the receiving system should react [4].
- **Negotiation**: In here, multiple systems should negotiate to achieve their particular purpose. (e.g. systems negotiate to determine which instance may exert controlling power) [4].

2.5 Technical Category and Selection Criteria for EAI

This section describes the concepts for the technical categories, which are the characteristics and selection criteria used for EAI solutions, found and taken from different related reviewed papers.

From literature papers in [6] [9] [14] [15], most of the existing approaches to implement EAI solution patterns, addresses both the technical and organizational features. In addition, the factors that influence EAI adoption also consists of both technical and non-technical factors [13].

These technical characteristics which are categorized under the technical parts, also applies for the selection of EAI solution patterns [4] [14] [16] [17]. These technical categories used for the purpose of understanding and applying the technical aspects of the framework are:

- Category in EAI taxonomy,
- Integration style,
- Integration level,
- Layers of integration,
- Data Abstraction Level,
- Degree of integration (Coupling),
- Target applications, and
- Quality of integration.

2.5.1 Category in EAI Taxonomy

This categorization can be used by organizations to avoid the confusion that exists in different ranges of EAI solutions. The EAI solution patterns, could be categorized in the following EAI taxonomy, shown and summarized in Table 1 [16].

Table -1- Characteristics and Categories of the EAI Taxonomy- [16]

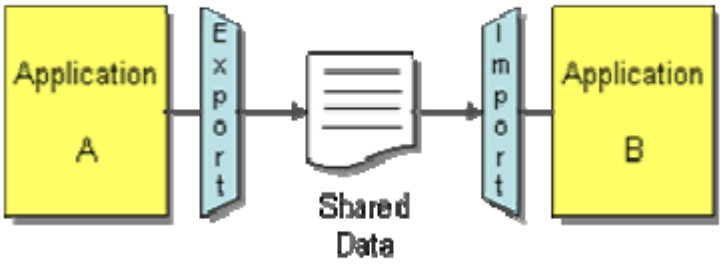
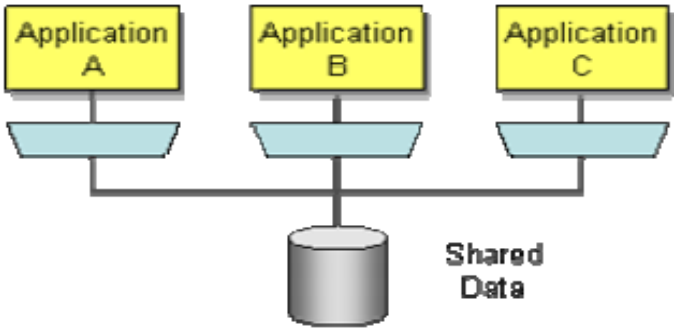
Category	Characteristics
Intra-organizational	Integrates enterprise applications.
	Integrates packaged and custom systems.
	No transactions with external users or partners.
Hybrid	Integrates Business-to-Consumer (B2C) applications with IT infrastructure.
	Hybrid application-integration supports transactions by integrating internal systems and/or external partners.
Inter-organizational	Integrates cross-enterprise applications with IT infrastructure.
	Integrates Business-to-Business (B2B) applications.
	Based on the degree of integration, this can be categorized into: extended enterprises and virtual enterprises.

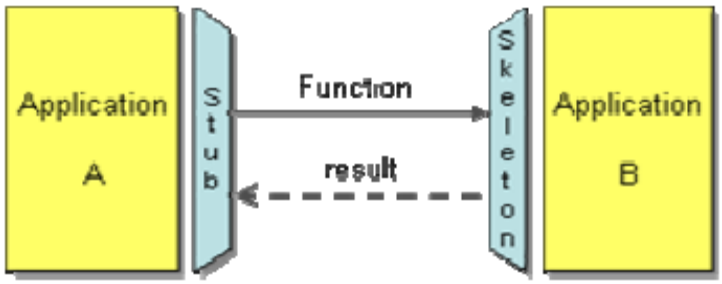
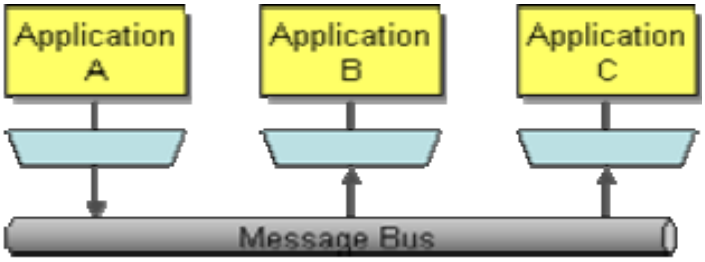
2.5.2 Integration Style

Application-integration solutions can be divided into a set of integration styles. Multiple styles of integration can be used so that each point of integration takes the advantages of the style that is best suited for the problem [17].

Table 2 describes the main four integration styles (which are: the file transfer, shared database, remote procedure invocation, and messaging) with their detail description.

Table -2- Basic Application Integration Styles with their Descriptions

Integration Style	Description
File Transfer	A way that one application writes a file that another later reads. The applications need to agree on the file name and location, the format of the file, the timing of when it will be written and read, and who will delete the file. Figure 5 shows the style of file transfer.  **Figure -5-** Style of File Transfer -[17]
Shared Database	A mechanism that multiple applications share the same database schema, located in a single physical database. Since, there is no duplicate data storage, no data has to be transferred from one application to the other. Figure 6 displays the shared database style.  **Figure -6-** Style of Shared Database -[17]

Integration Style	Description
Remote Procedure Invocation (RMI)	A way that one application exposes some of its functionality; so that it can be accessed remotely by other applications as a remote procedure. The communication occurs in real-time and synchronously [17]. Figure 7 indicates the applications' remote procedure invocation style triggered by functions.  **Figure -7-** Style of RMI -[17]
Messaging	A mechanism that one application publishes a message to a common message channel. Other applications then read the message from the channel at a later time. These applications must agree on a channel as well as the format of the message. The communication type is asynchronous. Figure 8 shows the messaging style of applications using message bus.  **Figure -8-** Style of Messaging -[17]

2.5.3 Integration Level

The integration levels aims to describe how deeply the different applications are integrated with each other [4]. Table 3 displays the four levels of systems integration with their description.

Table -3- The EAI's Integration Level with their Descriptions

Integration Level	Description
Information Exchange (**Data Level**)	A mechanism that one system provides information that is used in another system as part of its normal way of processing.
Basic Behavior Interaction (**Service Level**)	The way in which one system makes use of capabilities of another one. For instance, simple service request or a request with a reply. It also needs to address information exchange as part of interaction for communication.
Complex Behavior Interaction (**Business Process Level**):	Describes not simple but complex interaction among different systems. Complex behavior interaction differs from basic behavior interaction in that it typically spans across a number of individual interactions and later ones may depend on earlier ones [4].
User Interface Sharing (**UI** Level):	The way that multiple systems might need to share (parts of) the user interface. These might be different portals in a web-based interface or it may be that even within a single user interface some regions belong to different systems.

Figure 9 shows the four logical EAI levels of integration among different systems and in between data.

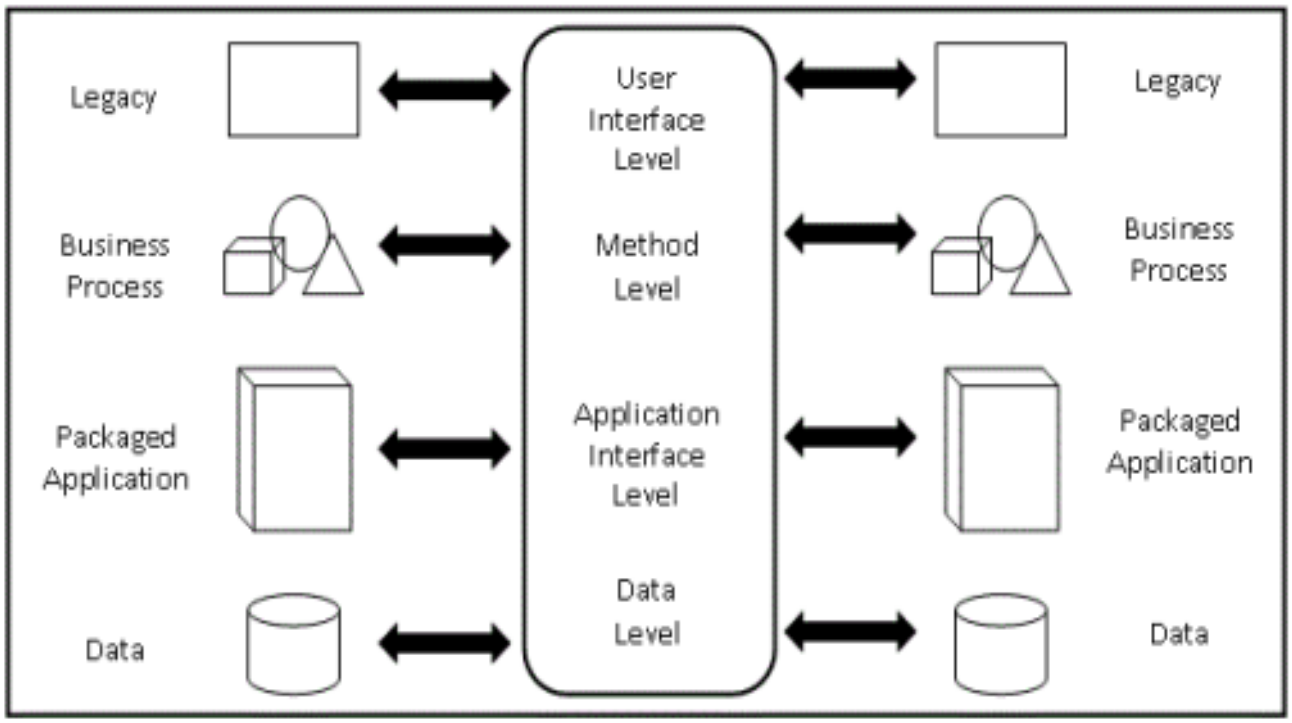


Figure -9- Different levels of EAI - [3]

2.5.4 Layers of Integration

Application integrations could also take place through four layers of integration, these are [14]:

- **Connectivity**: Plays role in interfacing with source systems in order to retrieve the data. The applications' elements: data and objects, are extracted from a source system using this connectivity layer.
- **Transportation**: Takes care in transferring data to the centralized integration infrastructure (i.e. the Enterprise Service Bus (ESB) bus or middleware).
- **Translation**: Translates and reformats these elements to the format of the target applications.
- **Process-automation**: automates business processes or routes application elements to the target system(s) and triggers new events.

2.5.5 Data Abstraction Level

For the purpose of communication and integration among systems, a mutual understanding of any data that is going to communicate is important. This requires a common basis on different levels [4]. Those levels that are distinguished are:

- **Structural**: Attaining relevant standards in describing the data exchange. It could be a low-level standard that does not necessarily determine the details of the data format.
- **Synthetic**: A way that the data exchange occurs with the appropriate format. This is used to denote that higher-level data types are appropriately mapped.
- **Semantic**: A way in which a correct data is provided as part of the data exchange. For instance, it implies that the described customer address is a real known customer address and correct one [4].

2.5.6 Degree of Integration (Coupling)

As per [16], the degree of coupling indicates the dependency of applications with each other. Where a tight level of coupling indicates a high dependency and a loose level of coupling indicates a lesser dependency between the different applications. It is also believed that EAI products which have a tight coupling supports synchronous data integration. Loose coupling is believed to be associated best with asynchronous data-integration. Table 4 shows the type of coupling and its detailed characteristics.

Table -4- Degree of Integration with its characteristics

Coupling Type	Characteristics
Loose	Focuses on exchange of data among systems.
	Low degree of integration and process dependency.
	Asynchronous communication.
Tight	Integrates cross enterprise business processes and systems.
	Highest degree of dependency and integration.
	Synchronous communication.

2.5.7 Target Applications

This criteria designates whether an EAI solution pattern integrates a packaged application or custom build systems [14]. For instance, many vendors provide products which only have interfaces with standard business applications such as SAP, Microsoft Dynamics and Oracle. Other vendors aim specifically in integrating legacy and non-standard applications.

2.5.8 Quality of Integration

The last important criteria that needs to be ensured is the quality of integration [4]. The following qualities are particularly important.

- **Reliability**: Ensuring that at least the system initiating the communication is informed if the integration breaks down.
- **Performance**: Requires that the integration supports temporal properties and the constraints of integration.
- **Security**: Assures the integration is secure. Thus, no alteration of the exchanged information may happen and it must be possible to assure the source of the data [4].
- **Availability**: Guarantees that the integration source/destination remains available despite component faults or failures.
- **Scalability**: Requires that the integration will work correctly if many different systems or instances of systems are integrated, or much information is exchanged [4].
- **Manageability**: Means that it is easy to manage the integration.
- **Consistency**: Ensures the validity and integrity of the data shared between systems when integrated.

2.6 Organizational Measures of EAI

On top of the general category measures in section 2.4 and the technical characteristics described in section 2.5, EAI solution patterns could also have the organizational measures. The organizational measures, and summarized on Table 5, points out the analysis of the existing EAI implementation methods [13]. More details of it, is found on the framework construction and development chapter of the research.

Table -5- EAI Organizational Measures and their Values - [13]

Type of Organizational Measures	Value
Reduce redundancy in:	Data
	Processes
	Applications
Reengineering:	Business Processes

Type of Organizational Measures	Value
	Flow/Exchange of Data
Define Ownership for:	Data
	Processes
	Applications

Hence, from Table 5, it is understood that before implementing the integration solution pattern, it is recommended to first reduce redundancy. That means, duplicate data and business processes, and applications should have to be eliminated and/or removed. In addition, inefficiencies on the existing processes of individual systems and the ways that those systems exchange data among each other should also have to be improved or reengineered. And then, ownership and responsibility of data, processes, and integrated applications will have to be identified. This will prevent confusion about responsibilities, quality and costs.

The secret behind these steps is that these organizational category measures of EAI, would minimize the complexity of the technology that is needed to address the remaining integration problems [13].

2.7 Chapter Summary

This chapter starts by defining what EAI means based on different reviewed literatures and then selects one among them; which says **"**The unrestricted sharing of information between two or more enterprise applications. And, a set of technologies that allows the movement and exchange of information between different applications and business processes within and between organizations [9].**"** It also briefly explains what EAI solution patterns and Integration frameworks are, using different instances.

Finally, it defines what is meant by category measures, by saying "are ways or central and focal points, for the study and selection of solution patterns and integration frameworks, when integrating enterprise systems and/or applications". And, it then clearly describes the three basic category measures and their criteria for the selection of EAI solution patterns; which are: the general category measures (having the SoS scope, development context, and purpose of integration), the technical category measures (that are: the EAI taxonomy category, integration

style and level, layers of integration, data abstraction level, degree of integration, target applications, and quality on integration), and the organizational category measures (which have descriptions to first reduce the redundancy of data, business processes, and applications, reengineer them, and then define ownership for the reengineered data, business processes and applications).

Chapter Three: - Works and Methods for the Framework

3.1 Overview

This chapter first reviews different literatures on the matter, to understand the state-of-the-art on the area, to point out lessons on how it is done on the subject matter and also to show some gaps and limitations observed. It then illustrates the basic determining factors defined for the proposed framework, which are: the basic guiding principles, the chosen parameters and selected modeling techniques. The fourth section illustrates the type of engineering adopted for the framework development, which is: *method engineering*. It specifically describes the *situational method engineering* of an *assembly based* approach. Finally, it explains the qualities that the framework should accomplish which are also used as a validation metrics.

3.2 Related Works

Five elected literatures have been reviewed in order to understand the state-of-the-art on the area and take lessons like: selection parameters, design science, and methodologies used for the framework. In addition, some limitations and gaps, have also been stated after reviewing each paper.

The paper in [4], first pointed out the categories of integration in order to determine and select a suitable patterns for integration solution. It stresses that before selecting SoS wide integration patterns, the two central basic types: data and control, should first be studied. Hence, how data would be shared and how control of elements among systems would be managed needs to be answered. It then emphasizes that to decide relevant patterns for integration problems, a study should be made classified in to two broad categories (i.e. general type and technical type of category). The general type of category focuses on understanding the SoS development context, integration purpose, and data/control choices. In the context part, clarification should be done on three classes of the development contexts, which are: *Greenfield, Brownfield* and *Closed form*. Also, the purpose of integration among systems, which are: one-directional information exchange and/or two-directional information exchange (sync) should be studied. The technical type of category, provides a more fine-grained grouping that can be used to determine relevant patterns. This includes: the integration level, data abstraction, data level

integration, interaction style, quality of integration, Levels of Information System Interoperability (LISI) and "PAID" attributes of interoperability: Procedures, Applications, Infrastructure, and Data.

The main lessons learned from this paper are, the two briefly described determining categories and pillars for EAI solution patterns selection; these are: the General type categories and technical type categories, and also the consideration of the EAI characteristics.

However, it does not clearly state the techniques used for outlining those categories of integration and type of mechanism used for preparing the EAI solution patterns selection.

In paper [25], it is because integration of enterprise systems is not an easy task to accomplish, patterns are possible ways of helping to find the best solution. The paper first proposed five different patterns for EAI. Two of the patterns are based on direct access to the database and three of them are based on Service Oriented Architecture (SOA): specifically based on web services. For all the proposed patterns, the subsequent factors and needs have to be studied before selection; considering their application's context. These factors are: how much is it difficult to share information, how much is performance critical with a big volume of accesses to data, how much is essential to keep data synchronized between systems being integrated, is there a likelihood for the access of data to be read-only and is there information that is local to one of the systems. Hence, the five proposed patterns for selection are: the direct access via database (DB), the direct access via Web Services (WS), the intermediate duplication with access via DB (in which performance is critical when there is a high volume of access to data), the intermediate duplication with access via WS and the duplication via monitor and WS pattern (which is used by applications that belong to different organizations and quite difficult to share resources).

The lessons taken from the paper in selecting EAI solutions are: first it defines basic factors for integration (i.e. sharing capacity, performance, security, and synchronization degree), and then it compares each patterns with each other based on a given scenario and takes the best one having high cumulated positive registered values.

Nevertheless, the paper has its own drawbacks. It only considers two of the integration mechanisms or styles, which are: shared database, and web service's messaging. It also does not

clearly describe the modeling science and/or techniques used for preparing the selection method.

As per [6], many organizations have difficulties in selecting a suitable application integration solution pattern. It implies that their remains the need of a well-founded method for selecting appropriate application integration solution patterns by considering both technical and organizational issues. This paper, using an *assembly-based method engineering* approach, has proposed a method for selecting application-integration solutions that supports organizations in choosing a suitable EAI solution pattern that positively influences the company's level of EAI solution. The proposed method executes different activities based on five major categories. These categories are: modeling, problem analysis, restructuring, requirements analysis and advisory. The modeling part includes sub-activities which are: modeling the application integration problem of an enterprise, modeling the data structure of involved applications by following the steps (i.e. identify the data, catalogue the data and prepare an enterprise metadata), modeling an end-to-end business processes, modeling of application interfaces and needed Information Technology (IT) infrastructure.

The problem analysis part includes activities which are: mapping the above specified models and analyze the integration gaps (i.e. identify missed data, procedures, and then after reanalyze the integration gaps based on mappings). Then, restructuring of activities is then done to avoid over-complex and costly application solutions. Analysis is then performed by identifying restrictions and conditions followed by outlining of requirements for EAI solutions. Finally, in the advisory part, an optimal EAI solution is selected including preparation of an advisory report with recommended solutions and remaining unaddressed integration issues.

The good lessons taken from the paper are: the development method, *method engineering,* used for the EAI solution patterns selection, and the type of categories and modeling techniques applied, and also the mechanisms used for presenting the selection method are the major ones. However, the paper has its own limitations in not considering the future and/or looming technologies' (like: Cloud computing, Mobile, and Big data), characteristics and parameters in its proposed selection method.

In [27], most of the world's available integration frameworks provides the realization of the well-known integration solution patterns. Decision in a way that which integration framework must be used is critical, since costs for the adaptation are not minimal. This paper has provided a method that helps integration architects to decide which integration framework should be selected. Amongst the integration frameworks available, the paper has taken four integration frameworks, which are: Camel, Spring, Mule, and Guarana, in which most of them are recent and open-source integration frameworks based on the integration patterns. The paper used and selected the basic maintainability measures categorized as: size measures, coupling measures, complexity measures and inheritance measures. All these categories have a total of twenty five different detailed measures, (some are: Number of Classes (NOC) used, Number of Interfaces (NOI) it has, Afferent coupling (AFC), Degree of abstractness of a framework (ABS), etc.), in order to decide and forecast how much maintainable the selected integration framework is.

The paper then applied a method to compute those measures using two free software tools, computing their empirical rank, and checking their rank for differences. This methodology evaluates maintainability focusing only on the core implementation of the selected frameworks. The method does not consider the code required to implement the adapters used to interact with the applications being integrated. As a result of the above four different statistical sample tests done, the Guarana framework outperforms the other integration frameworks regarding maintainability on core implementation.

The important thing found and taken from this paper is that the real world's most open source integration frameworks, should also be considered in addition to EAI integration solution patterns. It also defines basic measures used for selection and uses simulation tools for testing and measuring each integration frameworks based on those measurement values.

However, the paper has also its own drawbacks. It only considers and compares open source integration frameworks; without considering and comparing it with the EAI solution patterns. Also, the analysis is done only on the integration frameworks' core implementation. It does not even take the code required to implement the adaptors.

In paper [34], future and new technologies like: Cloud and Software as a Service (SaaS), Mobile, and Internet of Things (IoT), demand a new level of connectivity that cannot be achieved with

yesterday's integration techniques. It says that today's digital transformation is also driving companies to reframe their relationships with their customers, suppliers, and employees by influencing those upcoming new technologies to engage in ways that were not possible before. These technologies, particularly SaaS, mobile, and IoT, have dramatically increased the number of integration endpoints to connect to. In addition, it clarifies that the time dynamics in which these new technologies change have also been increasing. So, this paper emphasizes that existing connectivity and integration approaches or techniques are not fit for these new type of challenges. Hence, it underlines to consider new integration approaches for leveraging existing investments in order to drive transformational change, to enable agility and maintain visibility and control too.

The paper therefore points out that to meet todays and future needs, Application Programming Interface (API) connectivity that builds on the central beliefs of SOA, is emerged by defining methods for connecting and exposing different enterprise, legacy and upcoming systems.

Hence, the main contribution of this paper are: suggestions in which future and/or upcoming technologies and systems should also be taken in to account by the EAI's solution patterns selection framework, and also points out API connectivity as a solution for EAI integration.

3.3 Framework's Determining Factors, and Proposed System Model

This section describes the basic guiding principles that are taken in to consideration for the preparation of the selection framework. It also illustrates the main parameters chosen for the framework and proposed solution's system model. It finally describes the kind of modeling techniques taken and used in performing this framework.

3.3.1 Basic Guiding Principles

As per [11], ethio telecom has more than 180 functional applications, categorized in to business, corporate, and office applications. Out of them, nearly 55 applications (i.e. approximately 30% of applications) are enterprise systems which are critical and backbone for the company's day-to-day business related activities in providing services to subscribers. In addition, some of this enterprise systems are useful in controlling and monitoring the financial transactions of the company.

The integration landscape is also looming over a storm that is driven by the new forces of enterprise IT [29]. These basic four disruptive and driven forces of an enterprise IT are: Mobile, Big data, Social Networking and Cloud. This has promoted organizations to rethink and redesign enterprise IT solutions to cater to modern enterprise needs. Hence, the SOA/ESB approach cannot facilitate all of these new enterprise IT requirements; as SOA is primarily designed for internal interactions [29].

ethio telecom also has future expansion plans in bringing and providing new technologies (i.e. Software Defined Networks (SDN), Cloud Computing environments, IoT), using Telecom Expansion Projects (TEP) and other programs.

Therefore, on top of the criteria and category for EAI solution patterns found and learned in [6] [4] [25] [27], the current environmental context and the upcoming enterprise system features of the company, should also be incorporated when developing the selection framework.

Therefore, the below basic principles have been considered for the development of this framework:

- The company's current environmental context on existing enterprise systems and EAI solution patterns, and
- EAI integration considerations and parameters of the company's future or forthcoming technologies and systems.

3.3.2 Parameters Chosen for the Framework

Considering the basic guiding principles discussed on the above sub section, parameters and EAI selection categories are chosen for the development of the framework.

Some of these parameters are adopted from reviewed literatures and examined white papers on technical background. In addition, some new parameters are also added and some are modified based on the new added processes and deliverables.

Based on the basic concepts discussed in Sections 2.4, 2.5 and 2.6, the basic parameters taken for the proposed framework are classified as: general, technical, and organizational category measures.

The general category includes: the SoS scope, the development context and integration purpose, which are reviewed and expressed on related paper [4], and whose concepts are also discussed in section 2.4.

On top of the technical characteristics and categories reviewed and adopted from related works in [6] [4], additional technical category factors found and observed in [14] [16] [17], whose concepts are discussed on Section 2.5 (i.e. through Sub-section 2.5.1 to 2.5.8), are also considered.

The organizational category measures in [13], whose concepts are discussed in Section 2.6, is the third parameter chosen for the framework.

Finally, in [34] [37], future and new technologies such as: Cloud and SaaS, mobile, and IoT, demand a new level of connectivity that cannot be achieved with today's integration approaches. So, future and/or forthcoming technologies criteria and parameters are also considered and taken for this proposed framework.

3.3.3 Proposed Solution's System Model

Figure 10 shows a solutions'' system model designed for the proposed EAI solution patterns selection framework.

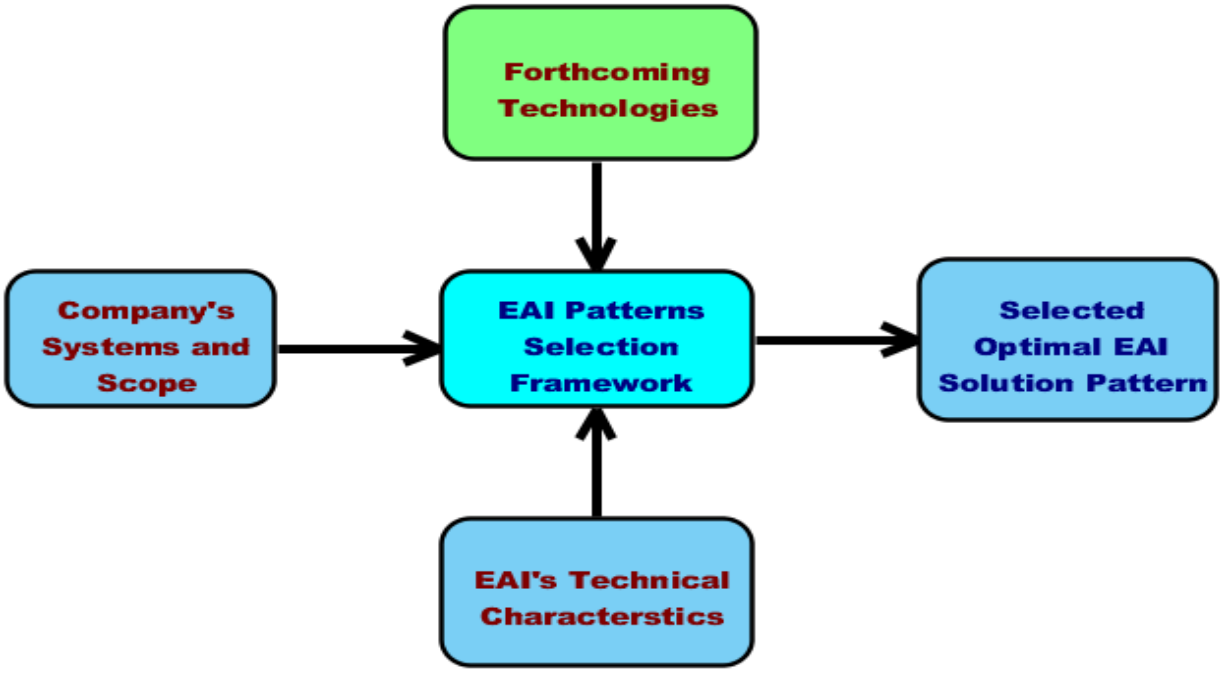


Figure -10- Proposed Solution's System Model

From Figure 10 solution system's model, the framework Inputs are: Company's existing context, Scope, technical characteristics of packaged and legacy systems, and forthcoming technologies. The framework's Process is: Studding and analyzing the inputs and develop EAI Solution requirements or specification. Finally, the framework's Output is: an optimally selected EAI Solutions' Pattern.

3.3.4 Modeling Techniques Selected

Modeling techniques are significant in order to attain the purpose of developing the framework. Hence, the modeling technique selected and adopted, is *method engineering* [6].

Method engineering is a widely used method; especially in the field of Information Systems (IS) development. Also, in *method engineering,* specifically *situational type method engineering* technique, is applied for this research.

In [22], out of the different approaches in *situational method engineering,* that are: *assembly based, roadmap based, paradigm based* and *extension based approaches*; the *assembly based approach (*which defines the projects situation and prepares a framework from candidate framework fragments*)* is taken.

In the meantime, for capturing and describing the parameters identified in the selection framework, the modeling technique described in [24], *meta-modeling*, is used in supporting the situational method analysis. And lastly, the *meta-modeling* technique results are presented using a method called *Process-Deliverable Diagram* (PDD).

The selected *method engineering,* which is a *situational method engineering* technique of an *assembly-based approach,* and the *PDD meta-modeling* techniques, are elaborated in the next Sections, (i.e. Sections 3.4 and 3.5).

3.4 Method Engineering

Method Engineering, for information systems development, is the discipline used to construct new advanced development methods or frameworks from parts of existing methods [21].

Many developed methods and frameworks, could also be considered to be the results of applying this *method assembly* [18]. For instance, Object Modeling Technique (OMT) has been developed from the existing method fragments called Object Class Diagram, State Transition

Diagram, Message Sequence Chart and Data Flow Diagram, all originating from other method sources.

Hence, the *method assembly technique* is a significant technique to construct both situational type methods and other powerful methods and frameworks with multiple viewpoints (i.e. behavioral view, functional view and/or object view [18]

Therefore, for the purpose of developing this EAI patterns selection framework, *method engineering* discipline techniques is chosen and adopted.

3.4.1 Situational Method Engineering

Method engineering, specifically a *situational method engineering* technique is taken [21]. This technique is a discipline to build project specific methods, called *situational methods*, from parts of the existing methods, called *method fragments*.

Also, the *situational method engineering* (*SME*), focuses on the creation of new technologies and tools allowing to construct project specific methods and/or frameworks instead of looking for universally applicable ones [22].

And most of the *SME* approaches promote the construction and adaptation of new methods by assembling reusable method fragments or method chunks. These approaches lead to the construction of new modular methods or frameworks. A modular method means a collection of interconnected method fragments/chunks [22].

Different approaches are found for *situational method engineering, based on* [22] [38]. These are:

- Assembly-based approach
- Roadmap-based approach
- Extension-based approach
- Paradigm-based approach

In [23], *assembly-based method engineering* approach, creates methods or frameworks from the present or existing method fragments, based on the needs of the specific projects or situations. The *roadmap-based method engineering* approach, uses roadmaps to configure or tailor an existing method to the specific project or situation. *Extension-based method engineering* is used to extend the existing methods in order to fulfill any missing needs. In [38], *paradigm-based*

method engineering, is the most generic approach to construct a method. By this approach, a new method is constricted by abstracting from an existing model or by instantiating a model.

Out of these four approaches, an *assembly based method engineering* approach, which mostly focuses on IS related project specific methods construction, is adopted for this framework. In addition, using this method makes possible to add relevant and established method fragments out of other existing methods.

So, in developing the framework, it becomes better to construct the framework's fragments from the existing EAI selection criteria; categorized and adopted from related works and literatures, by considering the basic guiding principles and factors defined in the previous sections.

3.4.2 Assembly based method Engineering

An *assembly based method engineering* approach, has four steps to follow for developing the framework [23][24]. These four steps are:

- Analyze the project's situation and identify the needs: It is a way of studying and analyzing the project area's condition (i.e. existing and/or present environment's context), and identify the requirements and determining factors for the selection framework.
- Select candidate frameworks that meets one or more aspects of the identified needs: It is a way of selecting nominated frameworks from different reviewed literatures of EAI solutions selection methods that meets the requirements and parameters of the project's situation.
- Analyze candidate frameworks and store relevant framework fragments in a framework base: It is a way of analyzing the proposed frameworks from candidate frameworks using different views and put the framework fragments in a repository.
- Select useful framework fragments by using route-maps configuration to obtain situational selection framework: Finally, choose the important selection framework fragments using route-map techniques and come up with a proposed selection framework.

In [23], Route-maps are ways in which framework fragments are combined to form a new structured framework. They are used to tune the framework in to a situational type of method/framework. In [24] also, there are basically two types of route-maps: static and dynamic. In [23], static route-map: is the way in which the implementation situation is chosen and the route followed is static. Whereas, dynamic route-map: if the usage of the method indicating that parts of the method are not useful, the map can be adopted or approved.

3.5 Framework Modeling Technique

For the purpose of capturing and describing the newly created EAI selection framework, a *meta-modeling* technique is used. As per [24], the *meta-modeling* supports the analysis and storage of the framework fragments and also the selection and assembly of the new framework. The *meta-modeling* technique also supports the process of a *situational method engineering* (i.e. especially the third and fourth steps of the assembly based method engineering), described in the previous sub-section.

In [24], the *meta-modeling* technique is used for analyzing, storing, selecting and assembling the framework fragments, and placing the results by using the diagram called the *Process Deliverable Diagram* (*PDD*). *PDD* is used for modeling the artifacts of the proposed selection framework's activities and results.

As per [24], *PDD* consists of two integrated diagrams.

- Process View (On the left hand side of the diagram).
- Deliverable View (On the right hand side of the diagram).

To model the processes view on the left side of the diagram, an adapted Unified Modeling Language (*UML*) *activity diagram* is used [24]. This diagram shows the schematic overview of the *activities* that are being used in the framework. Also, to model the deliverables view on the right side of the diagram, an adjusted UML *class diagram* is applied, offering the schematic overview of the *concepts* being used in the framework.

In [24], *activities* are the dynamic view of the system. *Activities* can be decomposed into *sub-activities* when necessary, and thereby creating a hierarchical activity decomposition. The deliverable side of a diagram is called a *concept diagram*.

A *concept* is also a simple version of a UML *class diagram*. *Class* definitions are set of objects that share the same attributes, operations, relations, and semantics.

An instance of the PDD diagram is shown in Figure 11, having the activity and concept views.

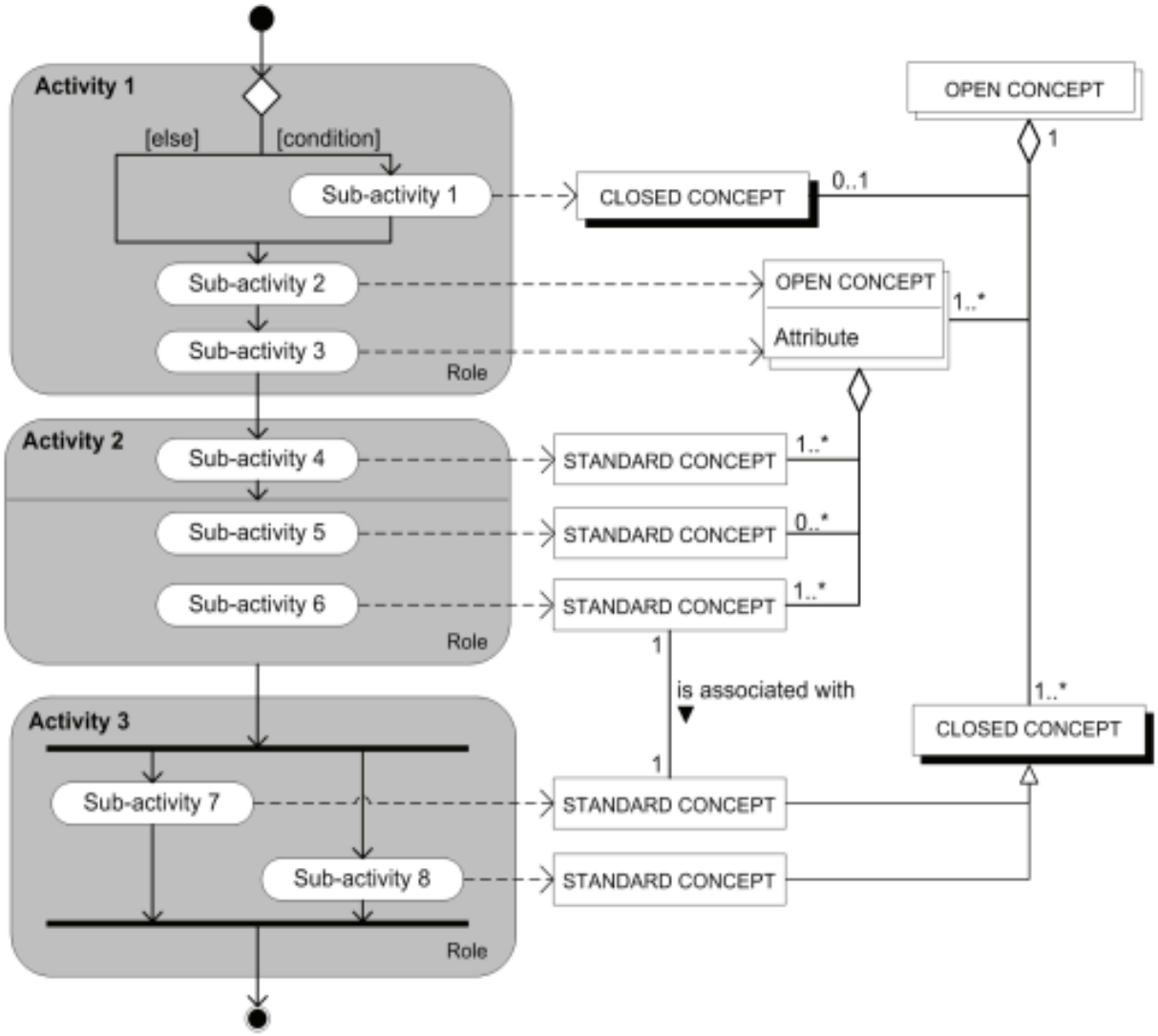


Figure -11- Processes and Deliverables in PDD - [24]

The PDD diagram shown in Figure 11, which is taken from literature papers in [23] [24], recommends to use two types of descriptive tables: Activity table and Concept table.

- Activity Table- Is used to provide a description for the *activities* shown in the process view.
- Concept Table- Is used to provide a description for the *concepts* shown in the deliverable view.

In [23], the *concept* types are: *Standard, Open* and *Closed Concepts*. In which, *Standard concept,* is a concept containing no further concepts and visualized with a rectangle. *Open concept,* is a concept whose concepts are expanded and visualized by a white shadow border. *Closed concept,* is a concept whose concepts are not expanded and visualized by a dark shadow border.

Again, in [24], different types of *activities* are found and used for different reasons. These are *Standard, Open, Closed, Sequential, Concurrent, Unordered* and *Conditional* activities. *Standard activities* contain no further activity and illustrated by round rectangle. *Open activities* are complex activities, whose sub-activities are expanded and illustrated by either round rectangle containing two or more activities or round rectangle with a white shadow indicating that the *activities* are depicted elsewhere. *Closed activities* are complex ones whose sub-activities are not expanded. *Sequential activities* are ones that needs to be done in a predefined order and are connected by an arrow. *Unordered activities* are used when sub-activities of an activity can be executed in any order. And, *concurrent activities* can occur concurrently, illustrated by drawing activities parallel in the diagram and connected with a synchronization bar.

3.6 Framework's Quality and Validation Metrics

Based in [18], for assessing and evaluating the quality of the proposed selection framework, the below proposed quality criteria are applied as metrics for validation. These are:

- **Completeness**: Assesses that the selection framework contains all the framework fragments needed and necessary, and
- **Consistency**: Evaluates that all activities, products, tools, and people plus their mutual relationships in the selection framework do not contain any contradiction and are thus mutually consistent,

Furthermore, the internal quality criteria below should also be attained.

- **Efficiency**: Evaluating that the framework can be performed at minimal cost and/or effort or time,
- **Reliability**: Measuring that the framework is semantically correct and meaningful, and
- **Applicability**: Demonstrating that the consultants and/or company design professionals are able to apply the framework.

For the purpose of evaluating the proposed framework, three case studies are done, for the selection of EAI solution patterns; in order to integrate the current enterprise and custom systems, and also for integrating the forthcoming technologies. Finally, reviews and validation is performed by internal experts of the company, based on this validation metrics.

3.7 Chapter Summary

This chapter basically has four main parts; which are: the related works part, and the proposed framework's determining factors and proposed system model part, the method engineering and modeling technique part, and the framework's quality and validation metrics part.

In the related works part, five elected literature papers have been reviewed for the purpose of understanding the state-of-the-art on the area and also take lessons like: selection parameters, design science, and methodologies used for the framework. Also, some limitations and gaps, have also been stated after reviewing each papers.

The framework's determining factors and proposed framework's part, then chooses and defines three basic parameters or category measures, which are: the general, technical and organizational category measures, used for the preparation of the framework, and whose concepts are briefly discussed through Sections 2.4, 2.5 and 2.6. It also clearly defines the proposed framework's system model.

The paper then describes what *method engineering* is, and proposes to use an *assembly-based situational engineering,* out of the different engineering approaches and techniques, for the development and assembly of the framework's fragments. It also explains and chooses the PDD Meta modeling technique, for modeling the framework diagrammatically.

Eventually, the five basic quality criteria, which are: completeness, consistency, efficiency, reliability, and applicability, are defined and proposed as a validation metrics.

Chapter Four: - EAI Selection Framework Development

4.1 Overview

This chapter describes the overall process of constructing the framework and their activities and deliverables found in the selection framework. Thus, the second section describes the requirements gathered from the company using interviews and questioners and the method engineering applied for selecting framework's fragments. The sections from third to fifth discusses on the prepared *PDD* for the framework fragments, and explains each and every phases of the *PDD's* framework processes.

4.2 Framework Processes Assembly

In order to construct this selection framework, the method assembly process is applied. For further details on this approach, look at Section 3.4 of chapter three.

4.2.1 Requirements for the Framework

Before starting construction of the framework, a series of preliminary interviews and discussions with company's stakeholders, has been performed. Interview questions have been prepared in three categories. This categories are: Managerial related interview questions, Business and service related interview questions, and pure technical type interview questions.

These interview questions were set up in a semi-structured way. The purpose of the questioners were to uncover the wishes and needs that those stakeholders had on the preparation of the framework, and also fetch requirements and selection criteria for the framework. The below sub sections illustrate the requirements gathered from interviewees based on categories.

A- Requirements from Managerial Staff

Three selected individuals with managerial positions, where: one is an IT Service Design Project Manager (**M3**), the second one is an Information Architecture Manager and Acting IT Service Design Officer (**M2**), and the last one is an IT Service Transition Officer (**M1**) have been interviewed. All of them are from the company's Information Systems Division, and have direct involvement with EAI solutions. Table 6, illustrates the requirements gathered and pointed-out by those managerial staff.

Table -6- Requirements description gathered from Interviewed Managerial Staffs

Item No.	Requirements Description	M1	M2	M3
R1	The framework should examine the current context of the company.	*	*	*
R2	Should choose a solution based on the company's prepared Roadmap and strategic vision.	*	*	
R3	Should give an advice on the remaining issues after integration.	*	*	*
R4	Consider the Gap on the company's systems' integration.	*	*	*
R5	Should consider the financial capability of the company for selecting the EAI solutions.	*		*
R6	Consider and achieve integration's quality needs.	*	*	*
R7	Should incorporate a mechanism for the solution using Phase based approach.	*		

B- Requirements from Business and Service related Staffs

Two selected individuals having IT business services and managerial position, in which: one is an IS Application and Services Officer (**B1**), the second one is a Data-Center and Application Technical Manager (**B2**), in which two of them are from the company's Information Systems Division have been interviewed.

Table 7, illustrates the aggregated requirements gathered for the framework pointed out by the above two IT service professionals.

Table -7- Requirements description for technological and Service related types

Item No.	Requirements Description	B1	B2
R8	Should consider the forthcoming technologies solution for integration.	*	*
R9	Should define the problem area for the need of technology in integrating with the solution.	*	*
R10	Should define the criteria and scope for integration of the upcoming technology.	*	*
R11	Should consider the service requirements based on predefined business rules.	*	

Item No.	Requirements Description	B1	B2
R12	Should incorporate testing of the solutions based on defined requirements before deployment.	*	

C- Requirements from Design and Integration Professionals

Lastly, two selected design and application integration professionals (i.e. **T1,** and **T2**) working on enterprise and 3rd party systems integration design, in the company's Information System's Division, have been interviewed.

Table 8, illustrates the requirements gathered for the framework pointed-out by the above two application and integration professionals.

Table -8- Requirements description for Technical characteristics of EAI Solutions

Item No.	Requirements Description	T1	T2
R13	It should incorporate the flow of data from sender to receiver applications.	*	*
R14	It has to minimize the flow of data and process redundancy.	*	*
R15	It needs to define the interfaces and needed infrastructures for integration.	*	*
R16	Should consider and prioritize the basic integration **qualities**.	*	*
R17	Define the ownership and responsibilities for the activities.	*	*
R18	Perform mapping of activities for integrated systems.	*	*
R19	Perform restructuring of activities to minimize expensive and redundant activities.	*	
R20	Should consider the data transformation and connectivity protocols between systems.	*	
R21	Consider external triggers and factors which could change the structure of the processes, data or applications.	*	*

D- Aggregated list of Requirements

Table 9, summarizes the requirements found and gathered based on the interviews done on three categories. (I.e. Managerial related requirements, Technological and Service related requirements, and purely technical type describing the technical characteristics of the EAI solutions.)

Table -9- Aggregated List of Requirements for the Selection Framework for EAI Solutions

Item No.	Requirement's Description
R1	The framework should examine the current context of the company.
R2	Should choose a solution based on the company's prepared Roadmap and strategic vision.
R3	Should give an advice on the remaining issues after integration.
R4	Consider the Gap on the company's systems' integration.
R5	Should consider the financial capability for selecting the EAI solutions.
R6	Consider and achieve the integration quality needs.
R7	Should incorporate a mechanism for the solution using phase based approach.
R8	Should entertain the forthcoming technologies by the solution for integration.
R9	Should define the problem area for technology need in integrating with the solution.
R10	Should define a criteria and scope for integration with the upcoming technology.
R11	Should consider the service requirements based on predefined business rules.
R12	Should incorporate testing of the solutions based on predefined requirements before deployment.
R13	It should incorporate the flow of data from sender to receiver applications.
R14	It has to minimize the flow of data and process redundancy.
R15	It needs to highlight the interfaces and needed infrastructures for integration.
R16	Should consider and prioritize the basic integration **qualities**.
R17	Define the ownership and responsibilities for the activities.
R18	Perform mapping of activities for integrated systems.
R19	Perform restructuring of activities to minimize expensive and redundant activities.
R20	Should consider the data transformation and connectivity protocols between systems.
R21	Consider external triggers and factors which could change the structure of the processes, data or applications.

4.2.2 Analyzing the Existing Methods

During the study and revision of related works, the below papers are found, studied, and analyzed; containing the proposed selection framework's fragments, which are: New, Adopted,

and/or Modified; fulfilling one or more of the aggregated requirements, summarized in the above Table 9. These literatures are:

1. Understanding Patterns for SoS Integration [4],
2. Patterns for Enterprise Application Integration [25],
3. A Selection-Method for EAI Solutions [6],
4. A Methodology to evaluate maintainability of EAI Frameworks [27], and
5. API-Led Connectivity- The next Step in the evolution of SOA [34]

The method on the first paper [4], specifically focuses on defining the general category measures which are required for EAI solution patterns selection. These general category measures are: understanding the SoS development context, integration scope and purpose of integration. This is an important step in the engineering of SoS in order to understand the development setting in which it is developed.

The second literature in [25], basically works only for five elected EAI solution patterns. The selection out of those patterns depends on studying their subsequent factors, like: Performance criticality, difficulty in sharing information, and etc.

The third related work in [6], supports organizations in choosing EAI solutions consisting of both the technical and organizational category measures. These basic measures and activities of the method are: Modeling (i.e. describe enterprise problem domain, describe data and process structure, describe infrastructure and interface), Problem Analysis (i.e. Mapping models and analyze integration Gaps), Restructuring activities, Requirements Analysis (i.e. Identify Restrictions, Define EAI Requirements (i.e. having the technical characteristics of the solution), and Advisory.

The forth paper in [27], provides the realization and selection of a well-known integration solutions called integration frameworks (i.e. Apache Camel, Spring, Mule, and Guarana), using the selected maintainability measures categorized as: size measures, coupling measures, complexity measures and inheritance measures. Hence, integration frameworks, should also be considered as one solution for integrating systems.

And the last literature in [34], focuses on future and new technologies that demands new level of connectivity for integration which cannot be achieved by yesterday's integration techniques.

4.2.3 Framework Fragments Selection

Based on the previously analyzed methods from different literatures and related works, the proposed framework's fragments or activities are selected and assembled fulfilling the company's aggregated gathered requirements.

Table 10, illustrates these framework fragments found on Section 4.2.2 with activities listed in vertical rows. The identified aggregated list of requirements in Section 4.2.1, are also listed in the horizon columns. Thus, for each activity (or framework fragment), the requirement that fulfills it, is marked by this (*) sign.

Note: - From the list of requirements found on Table 9, which are: R7, R20, and R21 are not directly incorporated on the list found on the vertical rows. However, R20 and R21 could be accommodated when working the data description and integration mapping activities (i.e. analysis of integration gaps and restructuring of activities).

Based on Table 10, the level of detail of each of the framework fragments varies and overlap exists, some customization is therefore made for a proper fit on the new proposed framework. Hence, in the next section, the results of the assembled framework fragments, is shown using the *PDD Meta modeling technique* on Figure 12.

Table -10- Selection of Framework Fragments

Framework's activities	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16	R17	R18	R19	R20	R21
1. Requirement Analysis (Rafael Z. Frantz, Rafael Corchuelo, 2015)																					
1.1 Define technology's' Business process Requirements and Rules	*		*		*	*		*			*			*		*			*		
1.2 Outline Integration criteria and Scope				*	*	*				*						*					
2. Study Current Environmental Context (R. Kazman, K.Schmid, J.Klein, 2013)																					
2.1 Study development Context	*																				
2.2 Integration Purpose	*			*						*	*										
3. Integration Modeling (van den Bosch, E. van Steenbergen, 2010)																					
3.1 Understand the enterprise and get Problem Domain				*				*	*					*			*				
3.2 Describe the Data								*					*	*			*				
3.3 Define the Interfaces & IT Infrastructure					*			*		*					*						
4. Integration Mapping (van den Bosch, E. van Steenbergen, 2010)																					
4.1 Mapping Models								*					*					*			
4.2 Analyze Integration Gaps			*	*						*			*					*			
5. Restructuring (van den Bosch, E. van Steenbergen, 2010)																					
5.1 Restructuring of Activities					*														*		
6. Integration Solution Requirements (R. Kazman, K.Schmid, J.Klein, 2013)																					
6.1 Define Restrictions and Conditions		*			*																
6.2 Describe Integration Quality						*		*		*						*					
6.3 Define EAI Requirements Specification						*		*			*				*	*					
7. Consider Future Technology for EAI Solutions (API Led Connnectivity, Mulesoft Company, 2016)																					
7.1 Define Future EAI Principles and Criteria								*	*	*											
8. Advisory and Consult (van den Bosch, E. van Steenbergen, 2010)																					
8.1 Select EAI Solution Patterns		*										*									
8.2 Produce advisory and Consult report			*												*						

4.3 Process Deliverable Diagram

Figure 12, shows a schematic overview of the proposed selection framework modeled according to the *meta-modeling* technique described in detail in Chapter three Section 3.5.

The proposed selection framework shown on Figure 12, has a total of sixteen (16) sub-activities and deliverables categorized by eight (8) distinct phases. Out of them, eight of the sub-processes and deliverables are directly adopted from similar related works. However, six of the sub-processes and deliverables are new and added on this framework, the rest two of the sub-processes and deliverables are modified because of the assembly of the new process fragments to this framework.

Table 11 summarizes the framework's fragments, displaying which of the activities (Sub-Processes) are new, which are adopted, and which are modified.

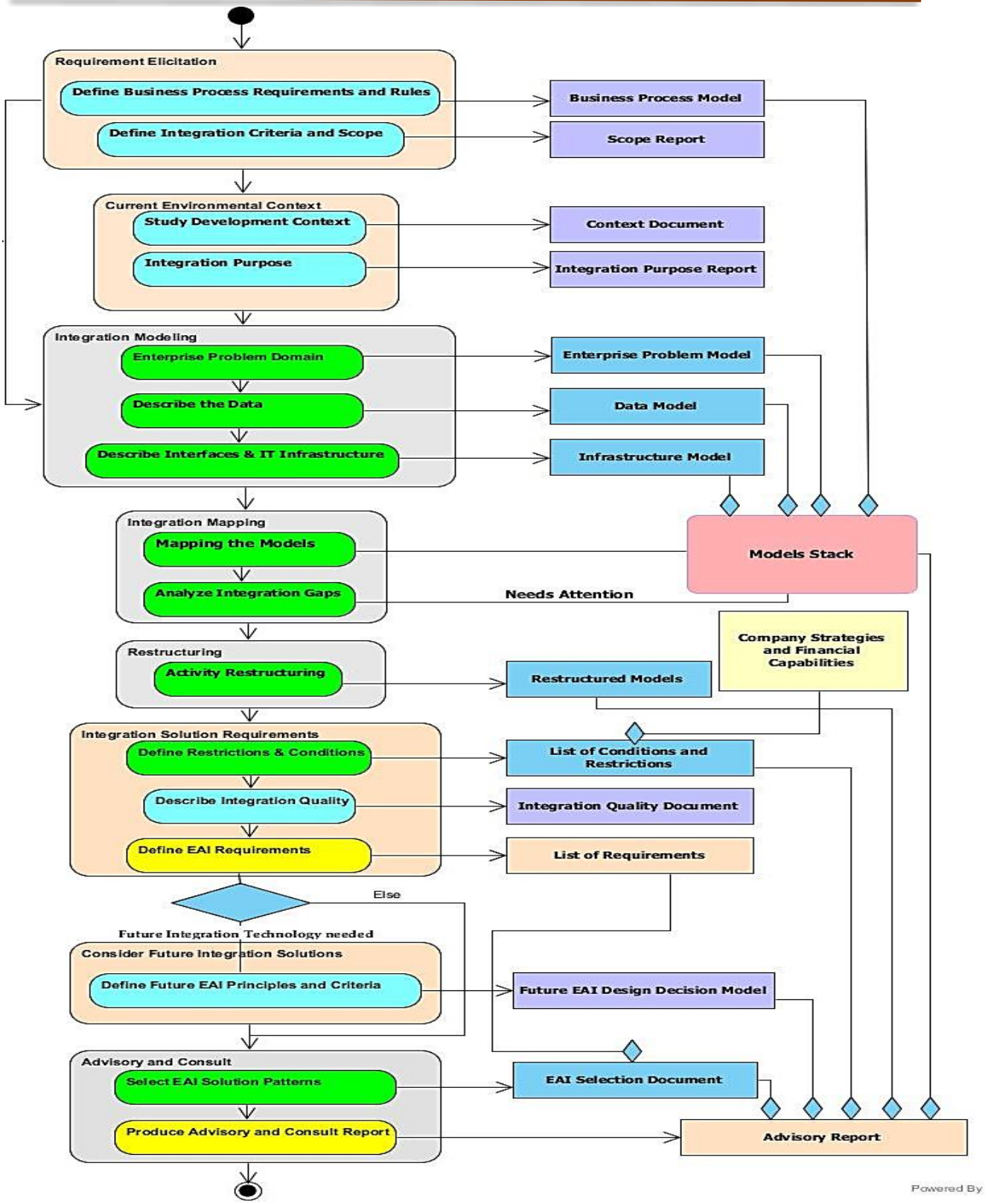


Figure -12- Process Deliverable Diagram of the assembled Selection Framework

Table -11- Summery of Proposed Selection Framework Fragments

No.	Basic Framework Phases	Sub-Processes	Activity Type	Deliverables
1	Requirements Elicitation	Define Business Process Requirements	New	Business Process Model
2		Define Integration Criteria and Scope	New	Scope Report
3	Current Environemental Context	Study Development Context	New	Context Document
4		Integration Purpose	New	Purpose Report
5	Integration Modeling	Enterprise Problem Domain	Adopted	Ent. Problem model
6		Describe Data	Adopted	Data Model
7		Describe In terface & Infrastructure	Adopted	Interface & Infrastructure Model
8	Integration Mapping	Mapping Models	Adopted	Models Stack
9		Analyze Integration Gaps	Adopted	Models Stack
10	Restructuring	Activity Restructuring	Adopted	Restructured Model
11	Integration Solution Requirements	Define Restrictions & Conditions	Adopted	List Conditions and Restrictions
12		Define Integration Quality	New	Integration Quality Document
13		Define EAI Requirements	**Modified**	List EAI Requirements
14	Consider Future Technologies	Define Future EAI Parameters & Criteria	New	Future EAI Decision Model
15	Advisory & Consult	Select EAI Solution Patterns	Adopted	EAI Selection Document
16		Produce Advisory & Consult	**Modified**	Advisory Report

Based on the *PDD's modeling* technique shown on Figure 12 and also by referring Table 11, the activities or sub process numbers: 1, 2, 3, and 4 are *unordered activities*, whereas activity numbers: starting from 5 through 13, 15, and 16 are *sequential activities*, but only activity number 14 is a *standard activity*.

4.4 Phases of the Framework

From the *PDD diagram* shown in Figure 12, the below are the distinct phases of the proposed selection framework. These are:

- Requirement Elicitation,
- Current Environmental Context,
- Integration Modeling,
- Integration Mapping,
- Restructuring,
- Integration Solution Requirements,
- Future Technologies Integration Criteria, and
- Advisory and Consult.

In the Requirement Elicitation phase, requirements for the integration of packaged and upcoming technologies and systems would be studied. In the Current Environmental Context phase, the current or existing situation of the company's systems integration, is analyzed and checked. In the Integration Modeling phase, a set of diagrams and/or models of data and processes are made for defining the EAI's situation. In Integration Mapping phase, the models would be mapped using techniques to minimize the gaps for integration. In the Restructuring phase, all points of attention that should be solved using organizational measures are addressed. In the EAI Solution Requirements phase, next to describing company's restrictions and conditions that apply in the integration scenario, non-functional integration requirements are also defined and list of requirements for EAI solutions would be illustrated, in the Future Integration Technologies phase, parameters of the forthcoming integration technologies are defined. Eventually, the Advisory phase contains a report for the selection of technical solutions and recommended organizational measures.

4.5 Framework Activities Description

In this section, activities that are used in the framework are described. This activities are listed in a consecutive manner. But, depending on real project's specific situation some of this activities could be exempted or reconsidered in real scenario.

4.5.1 Define Business Process Requirements and Rules

In order for the organization to move beyond simple application-to-application integration, an understanding of the end-to-end business processes is required [15]. Without insight of the involved business processes, it is difficult to determine that the organization is being integrated accurately.

This refers to the ultimate business process or systems' requirements. It defines the business processes and rules of systems for integration. These systems could be the existing company systems and/or upcoming (looming) systems which could be incorporated to the integrations' solution. Different interviews with concerned stakeholders would be done and company documents on this area could also be referred to find the available information.

Output

Business processes and basic rules of systems to integrate would be found after completing this activity. This could be done using the process flow diagram modeling techniques (e.g. Sequence or Process flow diagrams) and detailed activity descriptions could be described using tabulated processes activity descriptions.

Also, when having a higher level of process detail, the process structure can be described in the form of an enterprise function diagram with scenario overlays.

Preconditions

None. But, assumed that the company has the understanding and future strategic plan and/or roadmap of what is going to happen with the existing systems and upcoming systems and technologies considering the systems' integration.

Risks

When this activity is not done properly, some systems for integration could not be considered in to account that could ultimately lead us to a wrong decision. And also parameters of some upcoming systems could be overlooked. In general, if no clear understanding of the business processes exists, the need for integration cannot be defined clearly. So, lack of business process understanding will result in an integration solution that does not reflect the desired business process.

But, too much effort in describing processes in great detail, could be very time consuming. Therefore, it is recommended to get an overall understanding of the business processes.

Responsibilities

- Process analyst

4.5.2 Describe Integration criteria and Scope

This states the criteria, basic preconditions and parameters for integration. It tells the main considerations to take when integrating systems to systems or systems to the existing EAI solution. Scope determines whether preexisting decisions regarding the SoS design principles must be taken into account or whether the company is in a situation to actively make such decisions [4]. It means that to have SoS, how and what decisions should be adopted from individual systems or what new decision choices for the SoS is taken.

Output

An integration criteria and considerations or pillars and scope document is the result of this process.

Preconditions

First, describing the business process requirements and rules for integration is the priority condition before performing this activity. There is no integration criteria and scope without having business processes and rules of integration between systems.

Risks

Not doing this activity could create a space; since what is done and what is out of this project is defined only by the scope document.

Responsibilities

- Process analyst

4.5.3 Study the Development Context

This describes the EAI environmental settings of the company systems in which it is developed. It is an activity defining the current environmental category or situation of the company for systems to integrate. The company could therefore be categorized as Greenfield, Brownfield or Closed form [4].

Output

A study report describing the company's current systems' environmental context is found as an output.

Preconditions

None

Risks

Not performing this activity could ultimately lead us to significantly reduce the design options available to the system integrators.

Responsibilities

- Systems Analyst

4.5.4 Describe Integration Purpose

This activity describes the potential purpose of integration among systems that needs to integrate. There are various kinds of integration among systems, Such as: one directional, two-directional, control, and negotiation [4]. This clearly answers the purpose of integration among systems.

Output

A report document defining the integration purpose among systems is the result of this activity.

Preconditions

None

Risks

Not knowing the integration purpose could ultimately lead us not to tell the integration solutions pattern adequately, as patterns specifically supports the integration purpose.

Roles

- Integration analyst

4.5.5 Describe Enterprise Problem Domain

Integration problem domains mostly overpass departments and sometimes the company they are defined for. A series of interviews or workshop sessions needs to be done in order to understand the integrations problems area of the company.

Output

A scoping of the problem domain results after completing this activity. This is done by creating Supply Chain and Enterprise Function Diagram modeling techniques [31]. If the problem exists between multiple enterprises, corporate Enterprise Function Diagram should be made for all relevant parts of the affected enterprises.

Preconditions

Before doing this modeling, the company has at least a general understanding of the problem and integration opportunities. For this refer Sub-sections 4.6.1 and 4.6.2.

Risks

This activity is important. It dictates the modeling efforts in the next two activities. When this activity is not done properly, certain parts of the company systems would not be included in the modeling efforts of the next three activities. In a worst case scenario, the integration architect is not able to get a clear overview of the actual integration problems that exists within the organization.

Responsibilities

- Integration architect

4.5.6 Describe the Data

In every types of integration, it is important to understand the data that is used by integrated applications. Since integration technology usually involves some sort of data movement, modeling where and what data exists, is considered to be essential [6]. Besides the schematic representation, the use of a data dictionary is also recommended.

For describing the data structure, the following steps must be followed according to [9].These are:

- Identify the data
- Catalog the data
- Build an enterprise metadata-model

First the data has to be identified. This could be done by reviewing both the technical (i.e. High Level Design (HLD) and Low Level Design (LLD) documents of integrated systems, and business process documents). After the data has been identified, the data will have to be cataloged. This

can be done using a data-dictionary [9]. This should contain aspects like: 1) Reason for existence, 2) ownership, 3) format, 4) security parameters, and 5) Role. Finally, when all data is cataloged, a metadata-model will have to be created. This model contains information about the location of the data and how it interacts.

Output

The result of this activity is a data dictionary and data flow diagram of the data available in the organization. This should also describe the location of the data and how it interacts.

Preconditions

In order to start this activity, a clear understanding of the problem domain is required.

Risks

When there is no clear understanding of the business processes, the need for integration cannot be defined clearly. And, lack of business process understanding will finally result in an integration solution that does not reveal the desired business processes.

Responsibilities

- Process and system Analyst

4.5.7 Describe Application Interface and IT Infrastructure

Because application interfaces and IT infrastructure are an important determinant for the selection of a new application-integration solution, an overview of the infrastructure and interface landscape is needed [31]. During the description of these interfaces, any transformation or translation will also have to be incorporated into the description.

Output

To do this activity, a clear understanding of the problem domain is required.

Deliverables

After describing the application interfaces and related IT infrastructure, a diagram or table containing the interfaces is produced. This diagram can be created in a similar fashion using System Infrastructure Diagram, and Application Overlays [31].

Risks

This activity is important, since the results are later being used to determine the integration gaps. When this is done inaccurately, integration gaps can be overlooked or false-positives can be found.

Responsibilities

- Application administrator or system analyst

4.5.8 Mapping the Models

After completing the previously created data, processes and interface/infrastructure models, a mapping between those models is created. During the process of relating these models, integration issues and gaps could already be identified.

The mapping can be performed either in top-down (seen from a process and ownership perspective) or bottom-up (seen from data or interface/infrastructure perspective) [6]. Ideally, these approaches should result in the same model. Differences between these perspectives, can be considered points of attention and should be resolved before selecting a new integration solutions' pattern.

Mapping these models is also needed in order to make sure that the integration solution is properly aligned with all of the perspectives (e.g. data, business processes, and interfaces/infrastructure) [15].

Output

A set of mappings is created by this activity.

Bottom-Up Perspective

In this perspective the earlier described data structure is used as a starting point in the mapping process. Based on the data structure, the other models are linked to it.

Top-Down Perspective

The business processes are used as a starting point in the mapping process. Based on the processes, the other models are related. These diagrams can be modeled using the enterprise function diagram or system infrastructure diagram, combined with application overlays.

Preconditions

The mapping activity primarily requires the completion of the models that need to be mapped (i.e. data structure, process structure, application interfaces and IT infrastructure).

Risks

Without a mapping from the business process, an integration solution can be selected that does not align well with how the business works. In addition, it will not be possible to fully leverage the company's legacy systems and custom applications.

Responsibilities

- Integration Consultant

4.5.9 Analyze the Integration Gaps

The purpose of this analysis, is to discover integration 'gaps' based on the mapping that is created in the previous activity [6].

The following types of integration problems can be found using this analysis:

- Missing in data or functionality;
- Redundancy in data, application and systems;
- Inefficiency in the integration of data and processes;
- Governance issues (e.g. ownership, responsibility) in data, processes and applications/systems;

The gap analysis process in [6], as shown in Figure 13, consists of: 1) Input, 2) Output, 3) Objectives, and 4) Techniques.

The Input contains the mappings from the previous activity. In the Output, points of attention are specified. The Objectives indicate that this is done for the current situation (i.e. based on models and mapping results that are used as an input), and could also be done for the different change scenario.

In order to achieve this, techniques like: whiteboard brainstorming, or workshops can be done in close collaboration with stakeholders.

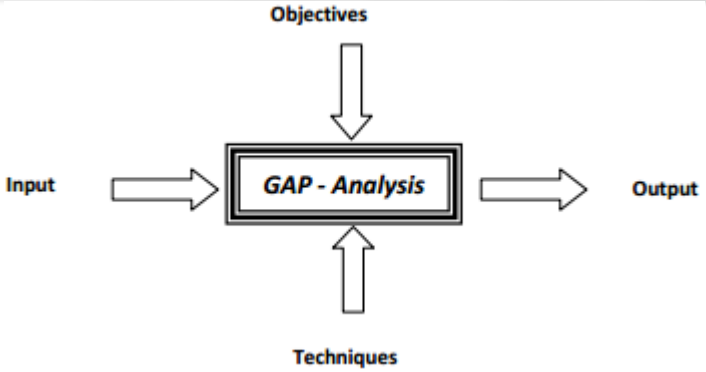


Figure -13- A Schematic Overview of the Gap analysis Process

Output

A list with the points of attention are defined for integration problems, after studying the mapped models. A consensus between the involved stakeholders on these attention points should be reached during this activity.

Preconditions

In order to identify gaps in the integration, it is required that a mapping exists between the previously mentioned models.

Risks

The gap analysis should be done in close cooperation with the involved stakeholders. It is preferred to do this in the form of a workshop. When this activity is done incorrectly, the above stated issues that are causing the integration problem are uncovered.

Responsibilities

- Integration consultants.
- And, all other involved parties/ stakeholders.

4.5.10 Restructuring the Activities

In order to avoid the selection of over-complex, high-risk and often costly application integration solutions, it is recommended to create a solid organizational basis before a technical solution is implemented [6].

The below sub-activities, shown in Figure 14, can be done during restructuring of activities.

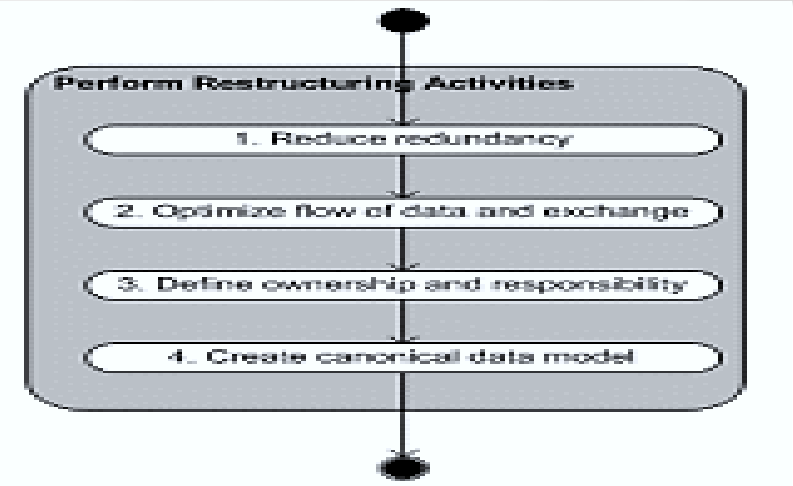


Figure -14- Process Diagram showing sub-activities for restructuring [6]

Output

This activity results in a restructured set of models. These models advise on changes that need to be taken before implementing an EAI solution.

Preconditions

In order to begin with restructuring of activities, a list with the points of attention is needed. In addition, the models created in the previous activities are also used in this activity.

Risks

Although, it is still strongly advised to solve these organizational issues before implementing additional technology, the practitioner of this framework should also remain realistic about the feasibility of these measures.

Responsibilities

- Integration Consultant

4.5.11 Identify Restrictions and Conditions

Before defining the requirement specifications of the EAI solution and the integration quality, the restrictions and conditions will have to be identified. Performing a distinction between project specific, strategic restrictions and financial capability is needed. That way, long term goals in the organization can also be taken into account when selecting an application-integration solution.

Output

A list of restrictions and conditions would be produced by this activity. The kind of restrictions and conditions could be divided in to the form of two categories:

- Company Strategies and project specific restrictions, and
- Financial restrictions and conditions

Preconditions

None

Risks

When this activity is skipped or done incorrectly, the solution that is selected at the end of this method can become unrealistic in project terms (i.e. time, money, quality). It can also be in conflict with the company's IT policy or architecture and ultimately be in conflict with the organization's long term goals and strategies.

Responsibilities

- Integration Consultant

4.5.12 Define Integration Quality

This activity is an important criteria in order to select an EAI solutions for the company [4]. The qualities of integration as defined in the previous chapter are: reliability, performance, security, maintainability, serviceability, and manageability, are included [18].

Output

A document expressing the quality criteria and type of quality having higher priority would be delivered.

Preconditions

All the above activities in the framework should first be done. Then, define the quality criteria for the EAI solution that is used for selecting an optimal EAI solution by doing prototype or during testing phase.

Risks

Not defining the quality criteria on integrations solution, could ultimately lead us not to measure the degree of integration and/or interoperability between systems. It would also create

additional time and money investment to ensure other qualities of integration that are not seen before.

Responsibilities

- Systems Analyst and Integration Consultant

4.5.13 Define EAI Requirements

After defining the quality criteria needed, and the restrictions and conditions, the requirements specification for the application-integration solution can now be defined more precisely. In section 2.5, the technical characteristics of EAI solutions have been presented. These characteristics would also be used as a guide in defining these EAI requirements. Then, verifying of those defined requirements with the stakeholders is necessary. This would be done by using workshop sessions.

Output

List of requirements for EAI solutions.

Preconditions

Before the requirements are identified, a clear understanding of the points of attention, the restrictions and conditions that apply to the specific project are also required.

Risks

Without a clear and complete set of requirements, the integration solution that is selected will be inadequate.

Responsibilities

- Integration Consultant

4.5.14 Define Future Integration Technologies' EAI Principles and Criteria

It is very advisable to define future integration technologies' (i.e. like: IoT, Mobile, Big data, Cloud and SaaS, and others) criteria and principles for the integration purpose with the current or existing systems [35].

In [35] also, the other main reasons for considering these future integration technology solutions into account, are:

- ESB integration solutions like: SOA based web services, helps to clear out the spaghetti mess created by masses of point-to-point integrations; by providing a simple, well

defined, and "pluggable" system that scales well. However, when times change, technology evolves, and for today's complex enterprise technological architecture with its hybrid infrastructure and rapidly exploding number of endpoints; ESB integration is no longer adequate.

- With a widespread adoption of cloud and SaaS applications, integration solutions needs to support on premise, hybrid, and cloud/SaaS applications. This requires integration capabilities to be cloud-friendly, scale horizontally, and spin up/down many instances rapidly. But, ESB is not designed to deliver for that.
- Generally, ESBs were designed before cloud, before mobile based service explosion, before social media, and before IoT. Hence, the product is not designed for agility and innovation.

Output

Future design decision model and parameters considering selected upcoming integration technology solutions.

Preconditions

The EAI requirements specification should be taken in to account.

Risks

Not considering future technologies would be costly, time taking, minimizes flexibility, and hinders agility and innovation.

Responsibilities

- Integration architect or professionals.

4.5.15 Select EAI Solution Patterns

After defining EAI requirements or integration considerations for future technologies, an EAI solution pattern can be selected. Based on [26], selecting an application integration solution includes the following activities:

- Find available application integration solution patterns.
- Filter results based on 'must-have' criteria and create shortlist.
- Evaluate shortlist solutions.
- Analyze results and select optimal EAI solution pattern.

Hence, the steps shown in Figure 15, results in the selection of an EAI solution pattern that is most suitable for the given set of requirements. A proof of concept could also be done for the selected solution.

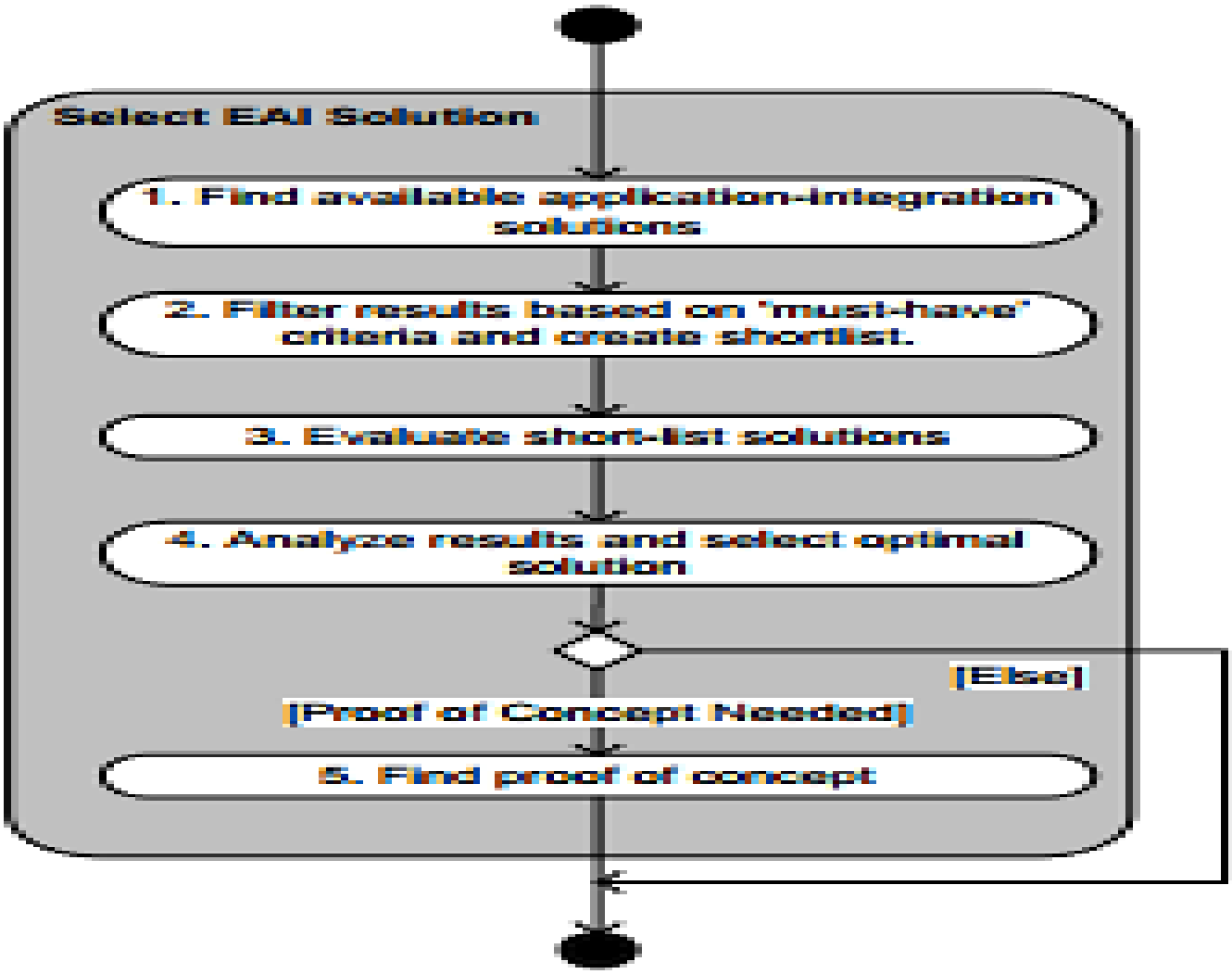


Figure -15- Process Diagram for the Selection of EAI Solutions [26]

Output

A report document of the EAI solution pattern selection process is made.

Preconditions

To select an EAI solutions pattern, EAI requirements specification should have to be defined.

Risks

When this activity is done incorrectly or skipped, no correct solution pattern will be selected.

Responsibilities

- Integration architect

4.5.16 Produce Advisory and Consult Report

According to the activities done above, final results of the modeling and problem analysis are described [6]. Based on these results, requirements for the EAI solution are listed and optimal EAI solution pattern is selected. In addition, the restructuring measures are also included. Furthermore, the remaining unaddressed integration issues (leftovers) and follow-up actions are mentioned here.

Hence, the advisory report generally provides project reports used for future EAI implementation task.

Output

The advisory report consists the following elements:

- Models,
- Points of attention,
- Conditions and restrictions,
- Restructuring measures and models,
- Report of EAI solution patterns selection process, and
- Unaddressed integration issues (leftovers) and follow-up actions.

Preconditions

An EAI solutions pattern should have to be selected first.

Risks

None

Responsibilities

Integration Consultant or Architect

4.6 Chapter Summary

This chapter first applies the method assembly technique described in Sections 3.4 and 3.5, for the assembly of the proposed framework's sub-activities, by collecting and then aggregating the requirements gathered from the company's managerial, technical, and business and service related staffs.

Secondly, the framework's fragments or activities are selected and re-assembled fulfilling the aggregated list of requirements based on the studied literatures. Then, a schematic overview of the proposed framework's fragments are prepared, using a Meta modeling technique known as PDD.

Eventually, eight of the main phases of the proposed framework found in the PDD, and also having sixteen sub-activities, in which each of the activity's deliveries, preconditions, risks, and responsibilities, are clearly explained.

Chapter Five: - Validation and Testing

5.1 Overview

This chapter illustrates the detailed validation process results found by applying the framework on three selected case studies, discussion reports for the case study's validation, and expert review results and findings done by using semi-structured questions. In each case studies, the first part defines the objectives for the case study, the next part explains the settings for the validation's selection, and the third part illustrates the sequence of process activities done by applying the proposed framework. The discussion sub-section then recaps the reports and validation results of the three evaluated scenarios. Finally, the third section summarizes the validation results based on quality metrics, evaluated by the internal experts of the company.

5.2 Case Study: ethio telecom's Enterprise and 3rd Party Systems

5.2.1 Case One- Voucher Card Provisioning and Activation

General description:

UVC system is mainly used to store voucher cards related data, and handles recharge requests from subscribers. UVC will first create Voucher Card (VC) file, the VC file then sent to VC dealer/supplier for the real voucher card generation. Subscribers finally use the VC card using Interactive Voice Recorder (IVR), Short message Service (SMS), and Unstructured Supplementary Service Data (USSD) functions to recharge their account.

In addition to the IVR, SMS, and USSD recharge modes that Online Charging Gateway (OCG) system provides, UVC offers interfaces for the business center, eCare, and charging system to access and perform recharge operations.

A subscriber also uses this VC card to recharge the account of any service, including mobile Service (i.e. GSM/CDMA (Mobile & Data)/WCDMA/3G/4G), fixed Network, and Data service. An encryption algorithm is used to ensure the security of voucher cards.

I- Validation Objective

The main objective of this case study is:

- To check the completeness and consistency of the proposed EAI solution patterns selection framework, and

- To assess and come up with another EAI solution pattern than the currently implemented vendor based and customized solution.

II- Validation Selection Settings

AS IS Scenario:

Currently, UVC works integrated with different packaged and other third party systems. The main integrated systems with UVC are: CRM, CBS's OCG, and Revenue Assurance (RA). ERP and VC Supplier's systems are third party systems that works interoperated with it. UVC, CRM, CBS, and RA systems have been deployed equally with OCG by the China's vendor company. Whereas, ERP system, is deployed by other third party next to them. Even if, the ERP and NGBSS systems almost started equally through Telecom Expansion Projects (TEP); NGBSS systems had been given more emphasis and implemented before ERP systems.

The systems' EAI solution is also done by TEP's vendor based projects, in which vendors become the only source in providing and implementing the integration solution by following their own selected integration pattern.

Hence, the implemented EAI solution pattern is SAP3's of SOA web based service solution, which had highly been customized by the vendor.

III- Activities Using the Proposed Framework

A-Defined Business Process requirements and rules

For this specific case of voucher card provisioning and activation, a model is done using a sequence diagram shown in Appendix B. Also business process flows for VC making file, activation, and provisioning is placed in Appendix C.

As seen in the sequence diagram and business process flows on Appendix B and C, the basic high-level processes that are involved on UVC making of file, activation, provisioning, and issuing of VC cards are summarized on Table 12.

Table -12- UVC's Basic High level Processes

Integrated Systems	Basic High-level Processes
ERP	1. Accepts voucher making file, 2. Inspects VC making file based on audit result, 3. sends to load VC loading file, and 4. Sends to issue the VC card for activation.
UVC	1. Generates voucher making file, 2. Generates audit file for reconciliation, and 3. Loads approved VC file into the DB, 4. Activates the VC file.
RA	Audits and reconciles the VC file with UVC and VC Suppliers.
OCG	Informs and dictates the VC usage.
CRM	1. Requests for VC activation, and 2. Sells VC cards based on activation.

B- Integration criteria and Scope

Basic considerations for UVC's integration:

The company uses VC cards used by subscribers for recharging purpose of voice, data, and SMS services. For that VC card numbers should first be generated by the company based on request and proposal by Marketing and Enterprise Division.

Thus, UVC first cross checks with Marketing Division for VC request through ERP system and prepares VC making file in order to be manufactured by VC Suppliers. This implies that integration should be made among those related enterprises and third party systems.

Hence, a study of subscribers' VC usage and request made by Marketing and Enterprise Division should be considered and followed.

Integration Scope:

The main systems' preexisting decision for integration to consider is ERP (as it handles the VC request data made by Marketing and Enterprise Division).

C- The Development Context

The Business Support Systems (BSS) solution includes systems integration of CRM, CBS, IPCC, Reporting, Provisioning, Mediation, and other enterprise systems components. Also, the solution includes integration of 3rd party systems like: Electronic Customers' Acquisition Form (eCAF), ERP and others. As discussed in the above sub-activities, UVC works integrated with other BSS systems (i.e. CRM, CBS's OCG and RA), 3rd party system like: ERP and external systems like: VC suppliers system.

From BSS point of view, there is no preexisting condition which restricts the design space of UVC. But in the case of ERP system, restrictions are found (i.e. platform, language of development, existing infrastructure); and should be taken into account. That means, Oracle's ERP uses different language and platform, since deployed by other vendor.

The VC supplier's module uses mediation service through Simple Mail Transfer Protocol (SMTP) and email service in order to transfer and accept VC making or manufacturing files and notifies BSS's RA, using security encryption and decryption mechanisms.

Hence, from BSS side, the development context is Greenfield; in which there is no implementation mechanism restricting the development of the solution. But, the integration of UVC considers the IT infrastructure of ERP. That means, ERP should be adopted to integrate with UVC system using a middleware as a backbone indicating that it's in Brownfield context.

D- Purpose of Integration

For the UVC's integration with other systems, both one and two-directional information exchange between systems are found. Table 13 shows the connected systems and their integration purpose with their description.

Table -13- UVC's Integration Purpose and Activation

Connection	Integration Purpose	Description of Integration
UVC <- CRM	One-Directional information exchange	CRM (Customer Care) query voucher cards and UVC activates voucher cards before selling to end customers and dealers. So, CRM (Dealer Management/Customer Care) invokes UVC interface to activate voucher card. When VC activation fails, UVC sends notification to CRM.

Connection	Integration Purpose	Description of Integration
UVC <-> OCG	Two-Directional exchange	OCG gets information about the voucher (value) from UVC, and it then informs UVC that the voucher has been used.
UVC->VC Supplier	One-Directional exchange	UVC provides VC file to VC Supplier for manufacturing.
ERP -> UVC	One-Directional exchange	ERP uploads VC to UVC for issuance and UVC issues and informs the result to ERP.
ERP <-> VC Supplier	Two-Directional exchange	UVC first sends manufactured VC card based on reconciliation result to ERP and ERP then sends acceptance or reject report for manufactured VC to UVC.

E- Problem domain of the scenario

UVC is generally used to store and issue VC related data and also handles the recharge request from subscribers. UVC also gives an interface for CBS's OCG for different type of recharging modes. The ERP system is needed for VC procurement, inspection, and issuance. The VC suppliers are needed to manufacture, deliver and reconcile the VC file with ERP and UVC systems.

The basic issues/problems to consider regarding the usage of data and the need for integration among those systems are,

- Complexity of integration among BSS systems (i.e. UVC, CRM, and CBS), and other third party systems like: ERP, and
- Inconsistency of data in the exchange of information among those systems.

Appendix A, is an Enterprise Function Diagram (EFD) diagram that shows the relationships among the involved enterprise entities and functionalities between systems in the integration's problem domain.

F- Describing Data

After defining the problem domain, the data and process structures were analyzed. This was done by looking at the LLD, HLD, and business process documents of the area. A data dictionary

of the VC process are made. The dictionary includes, source application, responsible person, data type description.

The data dictionary made was presented and verified by the stakeholders. Table 14, 15, 16 and 17 shows the data dictionaries with all attributes and descriptions of all the process interfaces with the UVC.

Table -14- UVC Making file attributes

Name	Attribute Type	Description
PO No.	Varchar	Purchase order number
Batch	Varchar	Containing the batch No. and prefix
Quantity	Long	Voucher card quantity
Voucher Type	Varchar	Specifies subscribers of vouchers and types of recharge rewards.
Serial No.	Varchar	Having a sequence no as prefix and serial no.
Voucher Pin	Varchar	Used by subscribers during recharge.
Face Value	Varchar	It's the value of the voucher.
Validity Date	Date	It's the validity period of the voucher Card.

Table -15- ERP-CRM-VC attributes

Attribute Name	Type	Description
Serial No.	Varchar	An ID given for the manufacturer.
Face Value	Varchar	Name of the manufacturer.
Vendor Code	Varchar	Uploaded manufacturers' public key to the UVC system.
Expiry Date	Date	Date of expiration.

Table -16- VC Audit file attributes

Attribute Name	Type	Description
PO Number	Varchar	It is provide by ERP, operator will input PO number manually
Batch	Varchar	Value given in set.
Quantity	Long	Total number of vouchers in a batch
Voucher type	Varchar	Type of voucher (i.e. hybrid,

Attribute Name	Type	Description
Face Value	Varchar	Amount it has.
Expiry Date	Date	Date of expiration
Start Sequence	Varchar	serial no which format is prefix and sequence

Table -17- VC Reconciliation result file attributes

Attribute Name	Attribute Type	Description
Manufacture Id	Varchar	VC suppliers Id
Operator Id	Varchar	ID number
Serial No.	Varchar	Serial Number
PO No.	Varchar	Provided by ERP
Batch No.	Varchar	Value given in set.
Start Serial No.	Varchar	Start serial no for the batch of voucher card.
End Serial No.	Varchar	End serial no for the batch of voucher card.
Qty. in UVC	Long	Number of vouchers in a batch which is come from UVC audit file
Qty. in Supplier	Long	Almighty
Qty. consistent Records	Long	Consistent records quantity.
Qty. inconsistent records	Long	Inconsistent records quantity.

G- Application interfaces and IT Infrastructures

Based on the business processes and the data structures defined above their application interfaces between activities are described in Table 18 below. These interfaces are found by revising the ICD documents, and also by interviewing the belonged professionals.

A detailed diagram of their relations between applications and their interfaces displaying the optimum integration is shown in Appendix D.

Table -18- VC and Integrated system Interfaces

Identified Applications	Interfaces between Applications
Between ERP and RA	VC reconcile result file
Between ERP and CRM	ERP-CRM-VC
Between UVC and RA	VC Audit file
	VC Reconcile Result File
Between UVC and VC Supplier	Voucher Making File
Between UVC and OCG	VC Supply audit file
Between VC Supplier and RA	VC Reconcile result file
	VC activation (Not found)
Between CRM and UVC	CRM and UVC

From Table 18, the basic interfaces between involved applications are defined and scoped. These are: between UVC and ERP, between VC Supplier and UVC, between UVC and CRM, and between UVC and RA.

H- Mapping Models

After completing the modeling activities, a mapping between the models is performed. The two models (i.e. top-down perspective and bottom-up perspective shown in Appendix E and F) is applied. The combined mapping displayed in Figure 16 below, shows the differences between the two models.

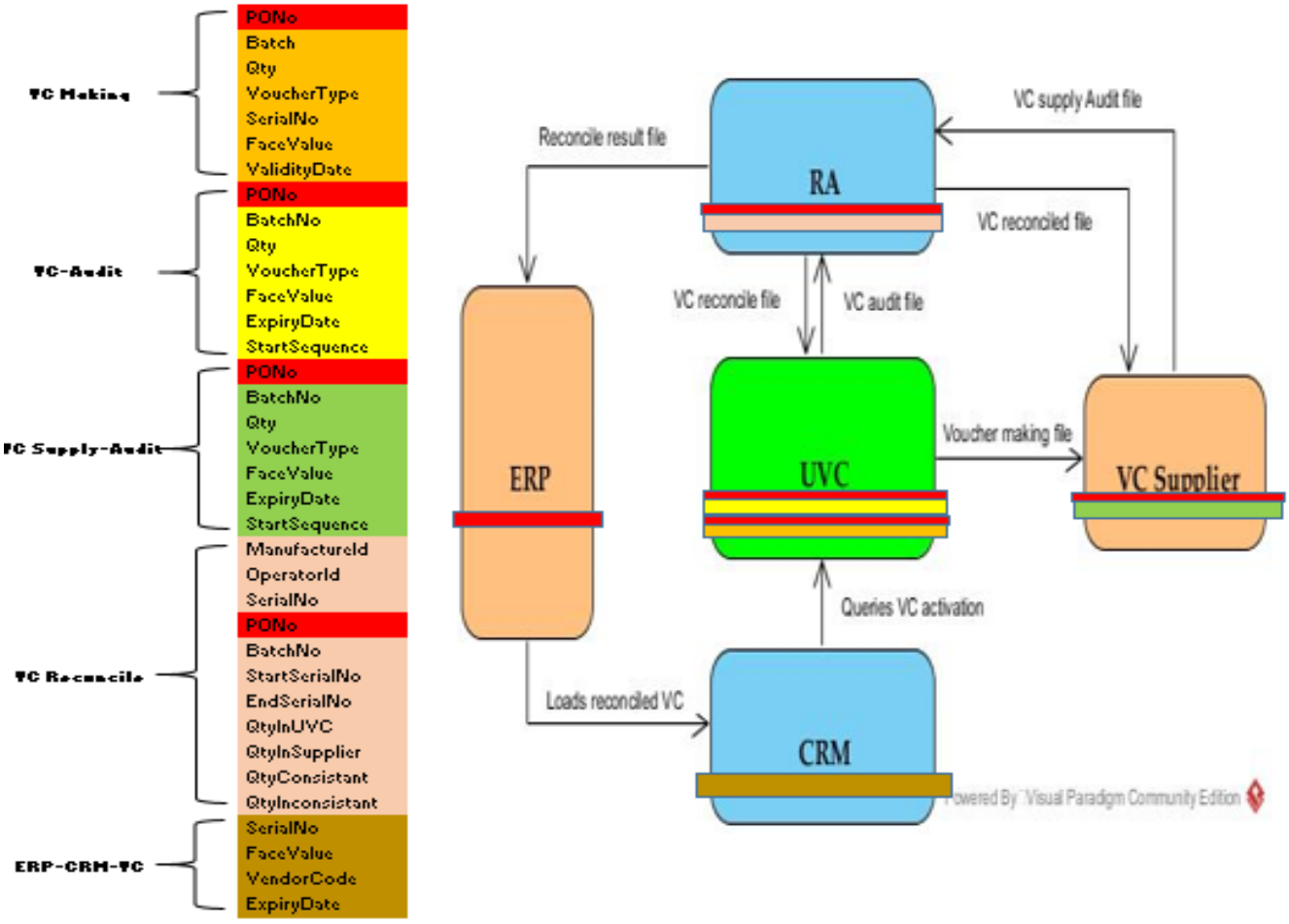


Figure -16- Combined Mapping model of UVC Integration

In the figure above, discrepancies between the data and processes is illustrated. This is indicated using the red overlays drawn on data and multiple processes (i.e. on UVC, VC Suppliers, ERP, and RA applications). Hence, mismatches like the one shown in the above figure provides an input for the gap analysis done in the next sub-activity.

I-Gap Analysis

The mappings done on the above activity dictates that the key point of attentions are illustrated based on discussions with the stakeholders. This results with the attention point that should be addressed before extending any new world of work related to this mater.

PONo is a main attribute on ERP system and also needed in UVC and VC Supplier's system for manufacturing purpose. Here, it needs attention on minimizing the gap and reducing redundancy, which is also elaborated in the next activity.

J- Restructuring of activities

When doing the gap analysis of the ERP system, and other enterprise applications (i.e. UVC, VC Supplier and RA), data overlaps on these processes have been observed. If this type of issue remains unaddressed or solved, it would create additional burden on the selected EAI integration solution pattern.

In this specific case, the ownership of the data (i.e. PONo.) is overlapped on different processes and application interfaces. Based on discussion with the involved stakeholders, the data ownership of PONo directly belongs to the ERP application. Other enterprise systems' application interfaces and databases use this data as a foreign key. Therefore, deletion, updating and removal of this data in relation to this scenario's sub-activity belongs to ERP sub-module.

Thus, to reduce or minimize redundancy, it is possible to remove PONo from RA. Because, the ERP's PONo would be sent to UVC using the voucher making process, whereas UVC's SerialNo and/or BatchNo are beyond enough for VC manufacturing process based on reconciliation result from RA. This indicates that redundant storing of PONo on RA would be minimized; which also optimizes the flow and exchange of data among ERP and other enterprise applications, and finally minimizes storage space.

K- Restrictions and Conditions

Before expressing the requirements specification for the EAI solution patterns, restrictions and conditions in relation to this or other project and company's strategies and financial capacity needs to be defined. For this purpose, the company's strategic goals, missions and visions has been referred. In addition, project specific restrictions has also been reviewed against what is going to be done in this specific UVC manufacturing scenario.

As a strategy, ethio telecom has an ambition to reach 100% mobile coverage and 10% fixed line service connection respectively. This is where TEP's project II is derived by the company from the launch of Great Transformation Project II (GTP II) by the Ethiopian government.

For that purpose, these applications are one service orchestrators that needs to be integrated smoothly, highly secured and reliably, with a high performance, and easily maintainable.

Thus, these basic strategical and project specific restrictions (i.e. related to time and quality) should be considered in to account. But, the financial capability document and budget cannot be found and included here due to confidentiality.

L- EAI Quality

The other criteria that needs to be considered is the quality attained by EAI solutions. Here, the basic quality criteria to be attained is Security.

An enhanced reversible encryption and decryption mechanism should be implemented when making the voucher card and passing the VC card to the VC suppliers too. Also, the VC supplier should use a decryption algorithm for VC making file and transfer the VC file before manufacturing it for reconciliation to RA.

The other quality characters to be fulfilled are systems' reliability and performance when transferring the VC making files and VC audit files among applications (i.e. ERP, RA, UVC and VC Supplier).

Hence for this case, Security, Reliability and performance are the main EAI quality requirements to be fulfilled, when selecting the EAI solution patterns from the available real world solutions.

M- EAI Requirements Specification

The EAI requirements specification is therefore shown and summarized in Table 19, in tabulation and report format.

Table -19- UVC's EAI Requirements Specification

EAI Requirements Specification	Description	Remarks
Development Context and Scope	Brownfield, the ERP legacy system needs adoption to the BSS system using adopters to integrate with packaged applications using middleware.	
Integration Purpose	Both One and Two-directional	
EAI Taxonomy	Intra and Inter (for C2C)	
Integration Style	File-transfer and Messaging	
Integration Level	Data level, Service level and UI Level	

EAI Requirements Specification	Description	Remarks
Layers of Integration	All that are: Connectivity, Transport, Translate, and Processes automation.	
Data Abstraction Level	Both Loose and tight coupling.	
Degree of Integration	Both	
Target Applications	Both standard and custom applications.	
Quality of Integration	Reliability, Security and Performance	

N- EAI Solution Patterns

Note that the selection of an EAI solution pattern does not include the "Proof-of-concept" due to time constraints. Whereas, an overview of different technologies and EAI solution patterns have been done from white papers and official publication of vendors' solutions.

Typically three different EAI solution patterns have been selected and analyzed for this case study. The first is SAP3's of SOA integration solution pattern which has already been deployed and implemented in the company. The second one is Salesforce integration solution pattern (Specifically: Remote Call-In for synchronous communication and Batch Data Synchronization for batch type of data file transfer). A solution pattern that is used in the industry with best practices parallel to SOA. The third choice is an Apache Camel Integration framework which is an open source integration framework which works like other integration solution patterns too.

1. **The SOA Integration solution pattern:**

From [39], Table 20 shows SAP3's SOA integration pattern with the type of technology, protocol type fulfilling the scenario's criteria.

Table -20- SOA Pattern for UVC Integration

Type Characteristics	Type achieved by the pattern	Scenario Criteria
Development Context and Scope	Brownfield	
Integration Purpose	One-Directional, Two-directional,	Both one and two-directional
EAI Taxonomy	Intra-Organizational, Hybrid, Inter-Organization	Intra and Hybrid
Integration Style	File-transfer, shared DB, RMI, Messaging	File-transfer and Messaging
Integration Level	Data Level, Service level, Business process level and UI Level	Data level, Service level and UI Level
Layers of Integration	Connectivity, Transport, Translate and Automate Processes.	All from Connectivity, Transport, Translate, and Processes automation
Data Abstraction Level	Synthetic and Semantics	Both needs to be attained.
Degree of Integration	Loose and Tight	Both
Target Applications	Standard (Enterprise) and Custom (Legacy)	Both standard and custom.
Quality of Integration	Availability, Security, Performance and Interoperability	Reliability, Security and Performance

SOA Integration solution Technology	
Technology, Protocol and Middleware type	Description
Web service of SOAP based XML, JMS, EDI and/or JDBC, uses WSDL language	Service provider binds to requestor or publishes to service broker.
ESB middleware technology.	➢ Service enabler (Router, Adaptor, Mediator and Transformer). ➢ Data and message service. ➢ Service monitoring. ➢ Service Orchestration functionality
EJB, SFTP, SMTP, RMI and EJB protocols	Service Bus protocols.

2. The Salesforce Integration solution pattern:-

Selected Salesforce pattern's basic criteria:-

The below tables are the solution type and criteria that should be fulfilled for the given scenario. Based on [12], The below tables, Table 21, 22 and 23 shows the two types of Salesforce patterns, which are selected for synchronous and asynchronous type of integration for the case study.

Table -21- Salesforce Pattern for UVC

Source/Target	Type	Timing	Pattern Type
Systems-> Salesforce	Process/Data Integration	Synchronous	Remote Call In
Systems-> Salesforce	Data (file type) Integration	Asynchronous	Batch Data Synchronization

2.1 Remote Call-In Pattern

Table -22- Salesforce's Remote Call-In for UVC

Technology, Protocol and Middleware type	Description
SOAP API (supporting SSL and TLS) technology	➢ Accessibility is high. ➢ It's Synchronous API for real time. ➢ Uses WSDL for systems.
Protocol	WSDL using XML
Middleware Capability (ESB based)	➢ Protocol conversion and translation. ➢ Queuing and buffering. ➢ Synchronous communication. ➢ Encryption, routing, and extract transform and load (ETL).

2.2 Batch Data Synchronization

Table -23- Salesforce Batch Data for UVC

Technology, Protocol and Middleware type	Description
SOAP API	Uses WSDL for systems.
Hypertext Transfer Protocol (HTTPS) Protocol	For confidentiality, integrity, and availability

Technology, Protocol and Middleware type	Description
Middleware Capability	➢ Protocol conversion and transformation. ➢ Queuing and buffering. ➢ Mediation routing. ➢ Reliable delivery. ➢ Extract, transform and Load.

3. Apache Camel Integration Framework

The Camel integration framework characteristics shown in Table 24, fulfills the scenario's specification. Camel integration framework offers a pattern implementation which can be used easily and has lots of transport types where data can be received from or written [40]. Also, it is easy to read files from the database and write files to the database or HTTP endpoints. Thus, Table 24 defines the technology and protocols of the framework.

Table -24- Apache Camel Integration Framework for UVC

Technology, protocol and Middleware	Description
Apache DSL, or XML technology	For defining business rules in the framework.
Apache ServiceMix	JBI Based enterprise service bus.
Apache CXF	For a web service framework.
Apache ActiveMQ message broker	Used for routing and mediation purpose.

N- Advisory

The advisory report would be presented to decision makers and stakeholders in order to support their final decision on the selection of the EAI solutions. This selection result should also include restrictions and conditions. Significant restrictions that are related to: company's budget, project and implementation time, and complexity; considering the project's EAI solution patterns security and performance are incorporated on the above activities.

Based on the case study done; Sap3's of SOA based on web service and Salesforce Batch data and Remote Callin are selected from the solution patterns and Apache Camel with ServiceMix and ActiveMQ from the integration framework are selected for recommendation.

But, this report should include all the above activities and also the budget and time needed starting from development till implementation with comparison of different vendors' solution patterns.

Note: The amount of budget and time needed for completion is not done here, as it requires invitation and offer from different solution providers. Also, the current investment on the existing solution cannot be found due to confidentiality.

Therefore, this advice could not include the optimum recommended solution pattern for this case study.

5.2.2 Case Two- New DT Service Integration with Current EAI systems

General description:

DT (Dynamic Tariff) is a new proposed value added service with a complete dynamic tariff solution; providing different discounts for voice (i.e. for: prepaid/hybrid/postpaid/2G/3G/ and 4G), and data (for: prepaid/hybrid/postpaid/2G/3G and 4G) at different locations and at different discount periods, when subscribers make a call or initiate a data session. It is sometimes known as Home Zone. It enables subscribers to define a zone as home zone and enjoy a preferential tariff or discount when the subscriber originates or receives calls in that home zone.

Zone and Home Zone:

A zone consists of a set of closed cells (closed cells having basic GSM, CDMA, LTE location area). A carrier or an operator could define multiple zones.

Figure 17 shows partial zone formation of different closed network cells.

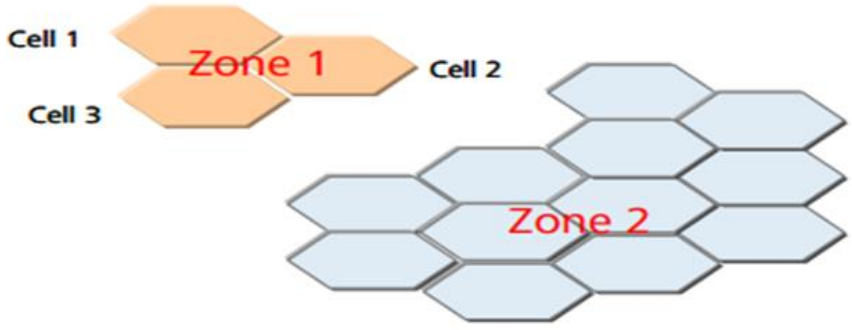


Figure -17- Network's Closed Cells for Zoning

Some benefits of DT service are:

- **Network Optimization**: Encourages the usage of network resources at times during the day where there is spare capacity, and alleviates the traffic to provide better network quality where there is network congestion rate.
- **Increases Network Usage**: Increases revenue from an existing subscriber's base (i.e. it stimulates dormant subscribers to call more or access data).
- **Improves Subscribers Experience**: Provides call price attraction and good calling quality, as a result improving subscribers experience and loyalty.

From the diagram shown in Appendix G, DT system (having different sub modules like: DT analysis, DT Management, DT Inter Media Center (IMC)), works integrated with existing enterprise systems; minimizes a preferential discount for subscribers.

I- Validation Objective

This scenario is done in a way to show the proposed framework in selecting an integration solution patterns accommodating the integration of existing enterprise systems with the new service application. Thus, it also checks the completeness of the framework in selecting an optimum EAI solution pattern.

II- Validation Selection Settings

AS IS Scenario:

Presently, NGBSS systems are integrated with third party systems like ERP system and others. The implemented EAI solution's integration pattern is SAP3's of SOA web based service solution.

The current pattern had been highly customized by the solution's provider. Hence, the intended solution pattern which accommodates the new DT service integration, should also work with the existing integration solution.

III- Activities Using the Proposed Framework

A- Define Business process Requirements

The integration of DT system with the existing packaged systems (that are: CRM, CBS, Network elements (i.e. network management system (NMS), SMSC, base station controller (BSC) and

base transceiver stations (BTS)) is shown in Figure 18. The red lines drawn in the figure, indicates one Scenario Overlay Diagram on EFD, displaying the business flows for integration.)

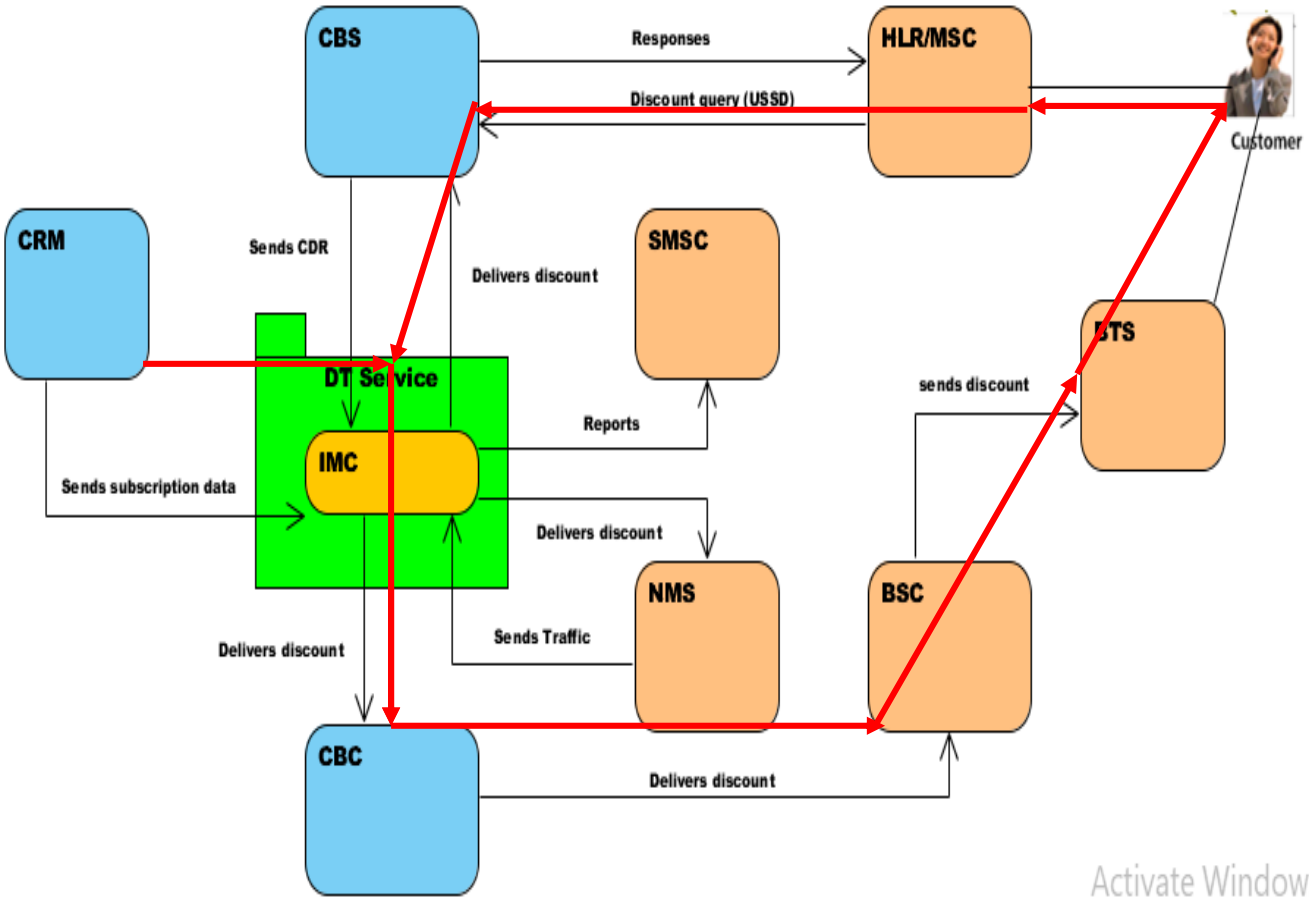


Figure -18- Scenario Overlay Diagram on DT's EFD

Thus, the Scenario Overlay Diagram indicated on Figure 18 shows the business processes for the DT service. Its detail business process flows is found in Appendix H.

B- Integration criteria and Scope

Considerations for DT's integration with existing systems:

The company basically needs to provide a new DT service to its subscribers in order to maximize its revenue and utilize its network resources at its optimum level. This would also increase the subscribers' usage experience by encouraging the usage of network resources because of attractive and reduced network tariff for voice and data service.

So, the main consideration for this integration is existing network's idealness and decrease in usage of subscribers' at an optimum level.

Integration Scope:

The major preexisting decision considered in this aspect is the repetitive study of subscribers usage and experience for voice, sms, and data services on different time basis (i.e. hourly, daily, monthly, and yearly), and also the company's tariff and revenue for those services. The new DT system should therefore accommodate the existing BSS systems integration.

C- Development Context

In integrating the new DT system with the existing integrated packaged applications (Such as: BSS, Operational Support Systems (OSS), and some Network elements), the current enterprise applications should be adopted to the new service's system.

This implies that adopters or middleware would be used on the existing side to interoperate with the new DT system. In that scenario the DT system should consider the basic criteria of the current or present integrated systems for integration. This indicates that the context for development of integration is brownfield.

D- Integration Purpose

The main integration purpose of the new DT system with the existing applications is to generate different discounts for registered subscribers using: voice, sms, multimedia messaging Service (mms), and data services. Table 25 shows the new DT service's integration purpose with its basic description.

Table -25- DT's Integration purpose with Current Systems

Connection	Integration purpose	Description
CRM -> DT	One-directional	CRM system sends the customers and subscribers data to DT.
CBS <-> DT	One-directional	CBS application sends the subscribers Call Detail Records (CDR) to DT for predicting the traffic of subscribers. And, DT finally sends discount rates for CBS to be used by subscribers.
DT <-> NMS	One-directional	NMS sends the network traffic of cells to DT on different time intervals for discount prediction. DT lastly sends discount tariff to NMS to be used by subscribers using OSS services (like USSD, SMS).

E- Enterprise Problem Domain

The new DT service should reconsider the existing integrated systems. The main problem in this regard is a way to think the integration technology that should attain the basic criteria of the DT service application and also the current integrated systems functionalities.

So, the main concerns with this regard are: how is the data transfer between these applications, what kind of security issues are there and how to alleviate the security hole without affecting the existing system, and how to expand or make ready this service's infrastructure for other upcoming services that needs integration.

F- Describing Data

The main DT system modules are: DT Management, DT Engine, DT Analysis, and DT Inter Media Center (IMC).

DT Management, focuses on cells management, configuration of discount policies, and discount management, etc. DT Engine, is responsible in segmenting cells based on segmentation rules, decides discount ranges and generates discount tariff using discount algorithms, studies user behaviors (i.e. CDR detail), and discount policies. DT Analysis, helps in network resource monitoring, business monitoring, DT effectiveness evaluation, etc. And, DT-IMC, provides integration interfaces to collect required data from external existing systems and then delivers DT discounts to those integrated external systems. Table 26, 27, 28, and 29 shows the data description of the main integrated systems and DT's IMC file. A high level processes defining the integration and interoperability of DT system with the current packaged applications is shown in Appendix H.

Table -26- CRM's data Subscription file

Attribute Name	Type	Description
CustomerID	Number	Customer ID
MSISDN	Varchar2	MSISDN
TypeID(Optional)	Varchar2	Type ID (service type ID, i.e. Prepaid, Postpaid, hybrid)
ProductID	Varchar2	Customer's Product ID
EffectiveDate	Date	Effective date
ExpiredDate	Date	Expired date

Attribute Name	Type	Description
UpdateDate	Date	Update date
OrderStatus	Varchar2	Activated or deactivated
CustomerID	Number	Customer ID

Table -27- CBS's CDR data source file

Attribute Name	Type	Description
Event Type	Varchar	Voice/data/sms/mms
CallingPartyNumber	Varchar	Calling party number
CallingCellID	Varchar	Calling cell Id
CallStartTime	Date	Call start time
CallDuration	Number	Call duration
ProductID	Varchar	This attribute help DT to identify whether this CDR is charged using DT discount.
ChargeAmount	Number	The fee of the call
CallType	Varchar	e.g. on-net local, off-net local, National, international
Discount	Number	DT Discount of this call
ServiceType (optional)	Number	Postpaid, Prepaid, Hybrid
CalledPartyNumber	Varchar	MSISDN of the called party
CalledCellID (Optional)	Varchar	Cell ID of called party
Charge Duration	Number	Charge call duration
Roam State (Optional)	Number	Roam state of the calling party

Table -28- NMS's Cell Performance Statistics (i.e.: 2G's)

Attribute Name	Type	Description
CellID	Varchar	Cell Global ID. This field is unique in System.
StartTime	Date and time	Start Time of this statistics
EndTime	Date and time	End Time of this statistics
AvailableTCHs	Number	The number of available TCHs, this also can be offered TCH Traffic (including half and full rate)

Attribute Name	Type	Description
TCH Traffic	Number	Carried TCH Traffic, including full rate and half rate traffic Unit: Erlang

Table -29- Discount delivery file from DT System

Attribute Name	Type	Description
CellId	Varchar	Id of the cell
Date	Date	Date applicable for discount.
StartTime	Date	Time discount would started.
EndTime	Date	Time discount would end.
Discount	Number	On net, and Off net, etc.
Event type	Varchar	Voice/SMS/Data, etc.
Subscriber Type	Varchar	Prepaid, Postpaid, hybrid

G- Define Interfaces and infrastructure

Based on the defined business processes in Appendix H, the proposed interfaces and presumed applications are shown in Table 30. The DT service needs to integrate with packaged applications which are: CBS and CRM, and also Network elements that are: SMSC, USSD, and MMSC. In addition it integrates with NMS.

Thus, the main interfaces and relationships between the DT system and external packaged applications considering the major relationships is shown in Appendix I's EFD.

Table -30- Main Interfaces with DT System

Identified applications	Interfaces between applications
Between CBS-and-DT	CBS-to-DT-IMC interface
Between CRM- and-DT	CRM-to-DT-IMC interface
Between NMS- and-DT	NMS-to-DT-IMC interface
Between DT- and-CBS	DT-IMC-to-CBS interface
Between DT- and-NMS	DT-IMC-to-NMS interface
Between DT- and-SMSC	DTIMC-to-SMSC interface
Between DT- and-USSD	DT-IMC-to-USSD interface

From the interface and application on Table 30, the basic and important interfaces are: interfaces between DT and CBS, between DT and CRM, and between DT and NMS.

H- Mapping models

After completing the modeling activities, a mapping between the models is done. This comes up with two diagrams (i.e. top-down and bottom-up models) and a combined diagram of these two diagrams. These two models shown in Appendix J and K are: top-down and bottom-up mapping diagrams. The top-down diagram is done from the business processes and requirements point of view, whereas the bottom-up diagram is done from the data perspective point of view. The diagram shown in Figure 19 displays a combination of these two diagrams. Due to time constrains and complexity, the NMS-IMC data model for this case study took only the 2G's data cell performance attributes.

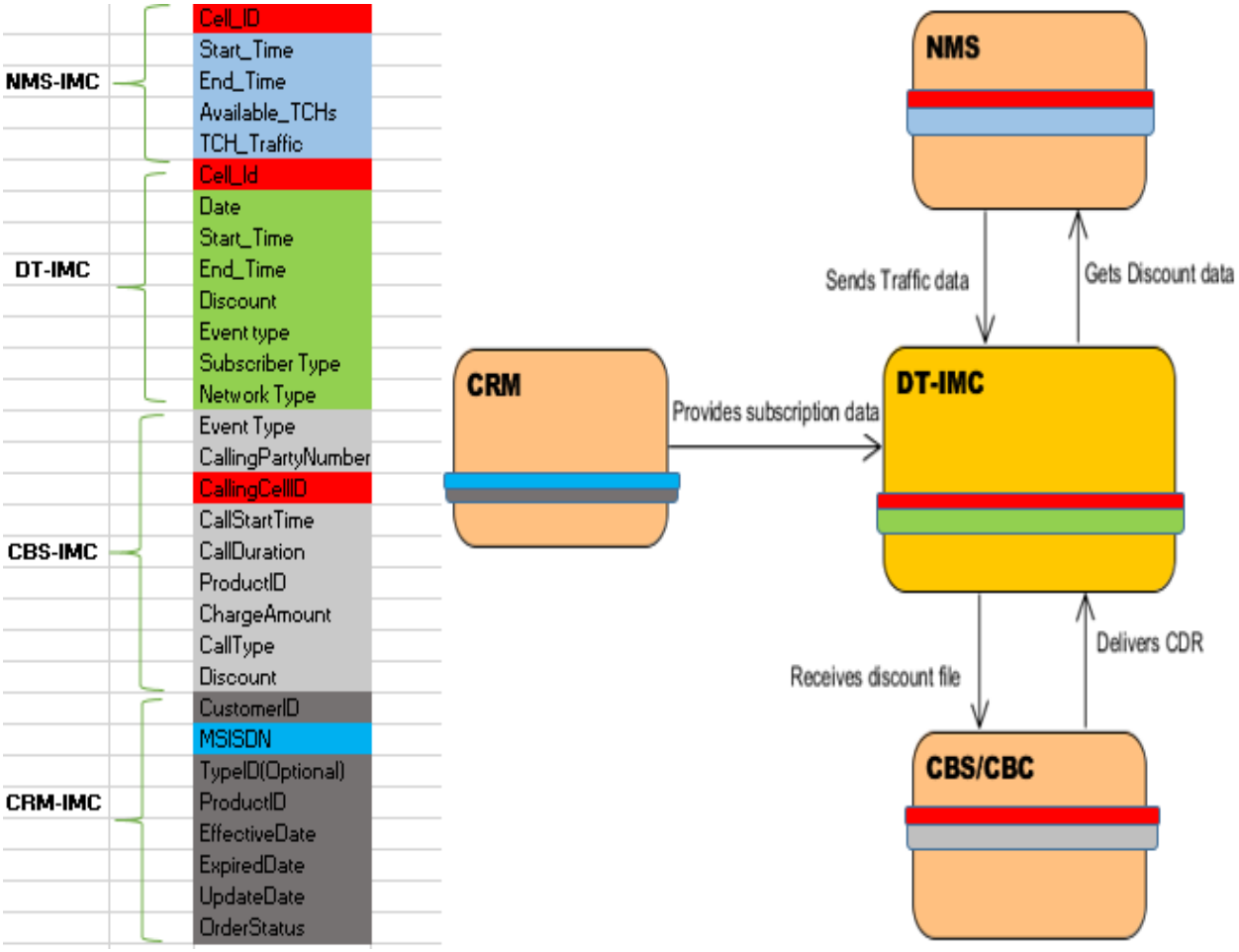


Figure -19- Combined Mapping Model of DT System's Integration

I- Integration Gaps

The mapping found on Figure 19 is explained to the stakeholders of the company. Informal discussion is done with the concerned technical staff and architects of the company. During that discussion with stakeholders, some key points of attention have been defined and presented in the mappings diagram.

MSISDN of CRM system indicates the subscribers' number, the CallingCellId of CBS stands for the subscribers' cell information which are ultimately used by the DT and NMS. So, CellId is the focus point.

J. Activity Restructuring

During the gap analysis, it is observed that the integration between CRM system and DT and the integration among DT, NMC, and CBS needs to have attention. Subscribers' basic information mainly goes from CRM to DT, DT also accepts the cell performance statistical information from NMS and the CDR details from CBS. Further analysis is done by DT using regression algorithm and a discount result is come up as an output. Since, DT is a new system that functional overlap would not be encountered. This is because, the flow of data between processes have been optimized.

But, for some observed processes the data ownership should be established and defined. Therefore, in CRM to DT-IMC integration, the MSISDN ownership belongs to CRM. CBS systems uses CRM's MSISDN (i.e. CallingpartyNumber), for integration. The CellId attribute ownership basically belongs to NMS system. So, CBS and DT services uses CellId for interoperability with the NMS system.

K- Restrictions and Conditions

The company's business, financial strategy and the organization's TEP2 reference architecture; is to make smooth integration with existing systems based on purpose, and deploy the new systems, integrated with existing ones to provide new services and maximize its profit.

For that there is no separate restriction for integration. The new integration therefore follows the company's business strategy and TEP2's reference architecture.

Here, the main condition and restriction is, the DT-IMC's discount calculation of the cells, which is basically dependent on the marketing and business strategies of the company (i.e. on Regions, Zones, and Cities marketing strategy).

L- Integration Quality

The basic criteria to consider is the quality for EAI solutions. For this specific scenario, the basic quality criteria is Security.

Very sensitive information and data is transferring from the current packaged applications like: CRM (customers' data), CBS (subscribers' CDR), and NMS (i.e. subscribers' traffic data) through secure file transfer protocol (sftp) and web service. This implies that high security techniques should be applied.

After the DT system calculates the discount amount of cells, the data should be transferred to NMS, CBS and also to CRM for usage by subscribers, on daily and/or hourly basis. This also implies that the system should work reliably with high performance.

It should also be maintainable or serviceable with minimum delay in order to get maximum profit from the service and to provide high satisfaction to subscribers.

M- EAI Requirements

Table 31 shows the EAI requirements specification shown in tabulation form.

Table -31- EAI Requirements for DT Integration

EAI Requirements Specification	Description	Remarks
Development Context and Scope	**Brownfield**, the current/existing enterprise systems such as: CRM, CBS, and NMS needs adoption to the new DT system using adopters or middleware for integration.	
Integration Purpose	One-directional information exchange.	
EAI Taxonomy	Intra (between sub-modules of DT systems).	
Integration Style	File-transfer (using SFTP) and Messaging (using web service)	
Integration Level	Data level (i.e. plain data transfer among DT modules and existing packaged applications) Service level, and	

EAI Requirements Specification	Description	Remarks
	UI Level (DT service management console used by operators.)	
Layers of Integration	All, which are: Connectivity, Transport, Translate, and Processes automation.	
Data Abstraction Level	All. Structural, synthetic and semantics needs to be attained.	
Degree of Integration	Loose coupling (for asynchronous transmission)	
Target Applications	The current existing system's packaged applications, which are: CRM, CBS, and NMS.	
Quality of Integration	Reliability, Security and Performance are the basic integration qualities needed.	

N- EAI solution patterns

Again, the selection of an optimal EAI solution pattern does not include the "Proof-of-concept" for this scenario too. However, like the first scenario, an overview of different technologies and EAI solution patterns have been done from different white papers and official publication of vendors' solutions.

Based on the study done in the above activities, the major integration type among existing and the new DT system is a secure transfer of data files on daily basis. This signifies that batch amount of file would be transferred and/or received among integrated systems. Hence, the EAI solution patterns would be selected that accommodates batch files delivery.

Typically three different EAI solution patterns have been selected and analyzed for this scenario. The first is the SOA solution pattern which has already been deployed and implemented in the company. The second one is Salesforce pattern (i.e. Batch Data Synchronization for batch type of data file transfer). It is a solution pattern that is used in the industry with best practices parallel to SOA.

The third one is from the open source community. Different types of integration frameworks have been released which could replace or work in parallel with other EAI solution patterns for EAI implementation. Some are: Apache Camel, Mule, Spring, Guarana, ServiceMix, and

ActiveMQ, etc. ActiveMQ and ServiceMix are related to Camel or used to enhance it and are produced and supported by the same community. For this scenario Mule framework is selected.

1. SAP3's SOA Integration solution pattern

In [4], the SOA pattern, is more natural and popular choice for many modern system to systems integration. It is organized in the concept of loosely coupled. This implies that the new DT service if offered with different languages, with different computing platforms and by different organizations. The SOA has a service registry so that weather new or existing services can learn of each other's existence. Table 32 shows the type characteristics and type achieved by the SOA solution pattern.

Table -32- EAI DT's Requirements and its Criteria achieved by SOA

Type Characteristics	Type achieved by the pattern	Scenario's Criteria
Development Context and Scope	Greenfield, Brownfield, Closed Source	Brownfield
Integration Purpose	One-Directional, Two-directional,	One-directional information exchange.
EAI Taxonomy	Intra-Organizational, Hybrid, Inter-Organization	Intra and Inter.
Integration Style	File-transfer, shared DB, RMI, Messaging	File-transfer (using FTP, SFTP) and Messaging (using web service for USSD.)
Integration Level	Data Level, Service level, Business process level and UI Level	Data level
Layers of Integration	Connectivity, Transport, Translate and Automate processes.	All, which are: Connectivity, Transport, Translate, and Processes automation.
Data Abstraction Level	Synthetic and Semantics	All. Structural, synthetic and semantics needs to be attained.
Degree of Integration	Loose and Tight	Loose coupling (for asynchronous transmission.)

Type Characteristics	Type achieved by the pattern	Scenario's Criteria
Target Applications	Standard (Enterprise) and Custom (Legacy)	Packaged applications, which are: CRM, CBS, and NMS.
Quality of Integration	Availability, Security, Performance and Interoperability	Reliability, Security and Performance

Table 33 shows the SOA integration technology, protocol and middleware type used to accommodate the type characteristics [39].

Table -33- SOA Pattern for DT Integration

SOA Integration solution Technology	
Technology, Protocol and Middleware type needed	Description
Web service of SOAP based on XML, JMS, EDI and/or JDBC, using WSDL language.	Service provider binds to requestor or publishes to service broker.
ESB middleware technology.	➢ Service enabler (Router, Adaptor, Mediator and Transformer). ➢ Data and message service. ➢ Service monitoring. ➢ Service Orchestration functionality
EJB, SFTP, SMTP, and EJB protocols	Service Bus protocols.

2. Salesforce Integration Solution Pattern

Table 34 shows the salesforce patterns that are selected for asynchronous for integration of bulk data files transfer.

Table -34- Salesforce's Batch Data Synchronization for DT

Source/Target	Type	Timing	Pattern Type
Systems-> Salesforce	Data (file type) Integration	Asynchronous	Batch Data Synchronization

Type Characteristics	Type achieved by the pattern	Scenario Criteria
Development Context and Scope	Greenfield, Brownfield, closed	Brownfield
Integration Purpose	Batch Information exchange	One-directional information exchange.
EAI Taxonomy	Intra, hybrid, inter-organizational	Intra and Inter.
Integration Style	File transfer	File-transfer (using FTP, SFTP) and Messaging (using web service for USSD.)
Integration Level	Information exchange	Data level
Layers of Integration	Connectivity, Transport, Translate	All, which are: Connectivity, Transport, Translate, and Processes automation.
Data Abstraction Level	Structural, synthetic	All. Structural, synthetic and semantics needs to be attained.
Degree of Integration	Loose	Loose coupling (for asynchronous transmission.)
Target Applications	Packaged, custom	Packaged applications, which are: CRM, CBS, and NMS.
Quality of Integration	Performance	Reliability, Security and Performance

Table 35 shows the selected solution, which is salesforce batch data's technology and protocol type [12]. But, it is also noticed that for batch data files transfer even if the performance of batch data synchronization is so high, the integration quality requirements such as; security and reliability could be compromised.

Table -35- Batch Data Synchronization's Technology for DT

Technology, Protocol and Middleware type	Description
SOAP Application Programming Interface (API) based technology.	For service communication protocol
HTTPS Protocol, and DMZ. Using company's firewall or proxy server.	For confidentiality, integrity, and availability
Middleware Capability (using enterprise ESB)	➢ Protocol conversion and transformation. ➢ Queuing and buffering. ➢ Mediation routing. ➢ Reliable delivery. ➢ Extract, transform and Load.
ESB pushes messages into SOA infrastructure and vice versa.	

3. Mule Integration framework

Mule ESB is a widely used integration platform with many built-in features that can be combined to create a very powerful ETL application. With its variety of endpoints including HTTP/HTTPS, SFTP, File, Database, and others along with its built-in transformers and robust custom component capabilities, Mule has the building blocks that can provide many, if not all, of the enterprise's ETL needs.

This implies the performance and security of this solution very attractive, which makes it selected than other integration frameworks.

Table 36 shows the needed building blocks based on the scenario's EAI requirements specification of mule's batch data transformation.

Table -36- Mule Integration Framework for DT

Technology, protocols and others	Description
Dataweave language	MuleSoft's graphical design environment in mule ESB for mapping and transformation.
Anypoint Connectors	Used to extract and load data into popular sources and endpoints.
XML, JSON, CSV, Excel, POJOs, and much more,	File type supported for processing by the pattern.

Technology, protocols and others	Description
Mule's Batch processor	Splits messages into individual records, performs actions upon each record, reports on the results and potentially pushes the processed output to other systems.
High performance, and High security (PGP encryption,	PGP encryption ensures unauthorized access than SFTP.

N- Advisory

This report is prepared and presented to decision makers of the company for last decision. Based on the above list of detailed activities performed, the case study has come up with EAI solution patterns solution (i.e. Sap3's of SOA based on web service and also Salesforce batch data synchronization). In addition it proposes Mule's Batch data processing solution.

However, the financial capability has not been performed; as it needs an RFQ preparation and invitation of solution providers. And that needs lots of time which cannot be accomplished on this study period. This made not select an optimum solution out of the three proposed solutions.

5.2.3 Case Three- Integrating Future Technologies

I- Validation Objective

The main objectives of this scenario are:

- To show the trends on integration techniques of the forthcoming technologies,
- To define basic considerations and criteria for future technologies integration.
- And, ultimately to fulfill the completeness on the validation of the proposed selection framework.

II- Validation Selection Settings

Based on the proposed solution's model, future technologies are one of the inputs that have been considered by the framework. Future technologies have an impact on the selection of integration solution patterns.

General Context:

The new technologies which are: Cloud and SaaS, Mobile, Big Data, and Internet of Things (IoT), demand a new level of connectivity that cannot be achieved with the past integration approaches. These technologies have dramatically increased the number of integration endpoints to connect to. For instance, payment transactions previously carried out by checks, are now transacted by an expanded set of channels including: telephone, online and mobile banking. The frequency with which this new and upcoming technology systems changed have also increased dramatically.

III- Activities Using the Proposed Framework

A- Integration Considerations and Criteria:

Future technologies are becoming the critical enablers of digital transformation. For instance, Hospitals are extending care beyond the hospital ward; non-bank players are driving innovation in the payments space; media companies are distributing the content across multiple channels and partners.

This implies that Mobile and cloud, for years viewed as trends on the horizon, are now proven to be drivers for IT-enabled business disruption, both inside and outside the enterprise. Therefore, digital transformation could only be achieved when organizations are able to bring multiple technologies together to create a truly distinctive and differentiated offerings.

According to [34], traditional methods for integration of applications and enterprise systems do not work for digital transformation. This is because, the former integration approaches are designed at time when there were fewer endpoints and slower delivery expectations. And, it often cannot move at the pace of today's business requirements.

Therefore, the criteria of using new API connectivity are:

- Point-to-point application integration is inflexible and expensive to maintain.
- The principles of SOA based integration are sound: having well-defined services that are easily discoverable and easily reusable. However, these goals were rarely achieved in practice.

- Using SOAP-based Web Services technology to implement SOA proved to be a heavyweight approach that was ill-suited then and even more ill-suited now for today's mobile use cases.

B- Proposed future Integration solution:

Based on a recent whitepaper by Mulesoft, MuleSoft's anypoint Platform is the main solution that allows enterprises to truly achieve digital transformation with API-led connectivity for service orchestration, application integration and APIs through a single platform [37]. Also in [36], Anypoint Platform is trusted by 35% of the Fortune's 500 enterprises and MuleSoft is the only integration provider to be named a leader across all of the three of Gartner's connectivity-focused Magic Quadrant reports.

Mulesoft's API Led Connectivity:

From [34], API-led connectivity, is a way to connect data to applications through reusable and purposeful APIs. These APIs are developed to play a specific role by unlocking data from systems, composing data into processes. Nowadays, API-led connectivity is becoming an important integration strategy. This is because the technologies that enterprises are using to engage with their customers, employees, and partners have changed dramatically.

In the paper, the convergence of the enterprise technologies like: IoT, SaaS, Big data, Social, Mobile, and APIs are providing powerful new tools to allow businesses to do more, unlock new revenue streams, understand their customers better, and innovate faster than ever before. But to do so, they need to integrate these new technologies using APIs.

As per [34], the main APIs that enables this API Led Connectivity are:

- ***System Layer:*** Here, the underlying IT architectures are the core systems of record (e.g. ERP, CBS, proprietary databases etc.). These mostly accesses the core systems of record and protects the user from the complexity or any changes in the subsequent layer.
- ***Process Layer:*** These APIs interact with and shape data within a single system or across systems and are created here without any dependence neither on source system (i.e., from where data originates) nor on target system (i.e., through which the data is to be delivered). The process APIs should be held privately and should not be revealed to the public.

- ***Experience Layer:*** Data can be reconfigured easily in this layer according to specific needs. They are the means by which data can be rearranged so that it is easily consumed by its intended audience, all from a common data source, rather than setting up separate point-to-point integrations for each channel.

Figure 20 is a diagram showing the three layers of API's connectivity described before.

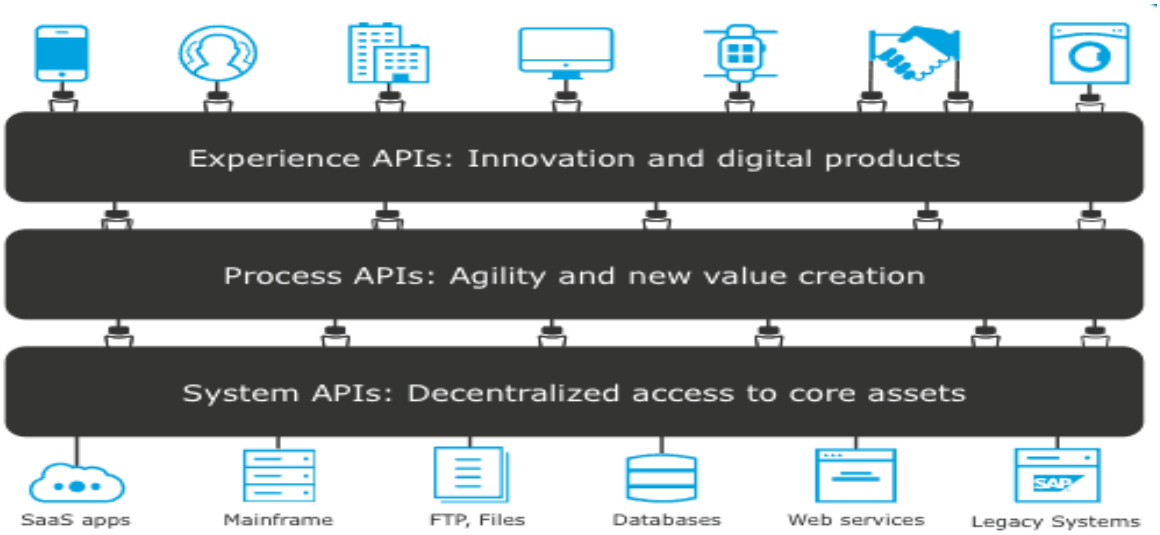


Figure -20- Mulesoft's API Led Connectivity API Layers [34]

Note:-

Further details of the above integration solution have not been included here and also comparisons with other future integration solutions have not been covered due to time constraints.

5.2.4 Discussion

This discussion is done based on the validation findings of the proposed solution using three case studies' and their suggested or proposed EAI solution patterns; after going through the detailed activities and sub-processes of the proposed selection framework.

It is a reflection of the validations' results, which is presented by classifying the discussion points in to two parts. These are: the technology matters (i.e. the selected EAI solution patterns' advantages and their importance), and varied results of EAI solution patterns (i.e. the reason for the variable results found on the first two scenarios and their advantages or disadvantages).

A- Technology Matters:

For the first two scenarios (i.e. for the integration of the current UVC system with other systems and integration of the upcoming DT service with the existing integrated EAI systems), the two proposed EAI solution patterns are: SAP3's of SOA pattern, and Salesforce integration pattern (which are: Remote Call-in and Batch Data Synchronization). From the open source integration frameworks, Mule's Anypoint Framework and Apache Camel integration frameworks, have been suggested.

For the third scenario (i.e. for the integration which also accommodates future integration technology parameters and conditions), Mulesoft's API Led Connectivity solution has been proposed.

For the first two validated case studies, the integration frameworks especially, Mule and Apache Camel have been chosen as a solution. Based on the technological reason observed on the literature reviewed in Section 3.2 and then after observing and understanding from their official web sites, both integration frameworks (i.e. Mule and Apache Camel) have some advantageous similarities. In which both are: Open-source (provides easily available source codes obeying global standards), Lightweight (are known for their small footprint and interoperable with other open source technologies and development tools), and also having active-communities (i.e. active online community of developers are most of the time present in both products).

In addition, Apache Camel is standardized, and highly available integration framework, available in the industry. Apache Camel, provides and works fully. But, it is mostly paired with Apache's ServiceMix or other 3rd party service containers. By [27], ActiveMQ and ServiceMix , in relation to Camel are used to enhance Apache Camel, and are produced and supported by the same Camel community.

Mule's Anypoint platform as an ESB, is also a complete integration framework addressing both non-functional requirements (i.e. reliability, high availability, scalability, and enterprise security), and functional requirements.

Table 37 shows and summarizes Mule and Apache Camel integration frameworks, discussed and important technical characteristics, which makes them suitable for integration.

Table -37- Integration Framework types and their Characteristics

Integration Framework Types	Characteristics
Camel and Mule	Are Easy to Learn. Both Camel and Mule are easy to learn from programmers' point of view.
Mule and Camel also,	Have Graphical user interface: ➢ Mule- using Anypoint studio, and ➢ Camel—by adding Fuse IDE.

On top of that, when looking the maintainability capability of these observed and selected integration frameworks, on selected enterprise systems and applications, and also based in [27]. Table 38 is prepared as a summary for showing the comparison and analysis of the integration frameworks, by adding spring framework.

Table -38- Integration Framework types Comparison [27]

Description	Integration Framework type		
	Apache camel	Spring	Mule
Number of Packages*	54	32	124
Number of Classes*	96	58	51
Total number of lines of Codes	62439	14929	67090
Number of methods in classes and interfaces*	7015	5158	1431
Number of Methods per Classes/Interfaces*	192	39	129
Maximum number of parameters per method	11	9	19

In Table 38, the column chart prepared and shown in Figure 21 is depicted displaying the other technological reason and importance of selecting Mule and Apache camel integration frameworks for the first two scenarios than that of spring framework.

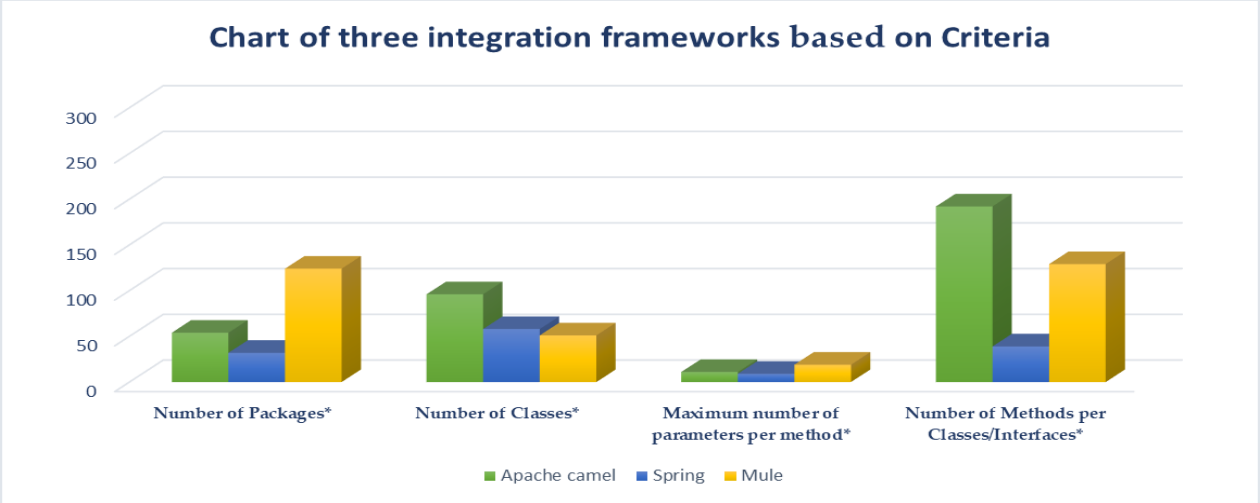


Figure -21- Comparison of three Integration frameworks

Likewise, SOAP3's of SOA based on web service solution pattern and Salesforce solution patterns (i.e. Remote Call-In and Batch Data Synchronization), have been proposed as a solution. Again, based on the technological characteristics referred in Section 3.2 and detailed technical importance and conditions discussed in Sub-sections 5.2.1 and 5.2.2; both solution patterns could accommodate the first two case studies and even could also interoperate with each other.

Generally, for the first two case studies, the proposed EAI solution patterns (that are: SAP3's of SOA based on web services and Salesforce patterns (i.e. Remote Call-in and Batch Data Synchronization), and also the elected integration frameworks (which are: Mule's and Apache Camel frameworks), are technologically fit and could also interoperable.

For the third scenario (that considers the integration of future technologies), the Mulesoft's API-Led Connectivity has been proposed as a solution. It is illustrated in Sub-section 5.2.3 that large enterprises having complex connectivity needs, requires multiple API based connectivity building blocks. This implies that agility and flexibility would only come from a multi-tier architecture containing different distinct layers.

Again, some additional technical benefits of using API based connectivity are [34]:

- Distributed and tailored approach: API based connectivity, a one-size-fits-all architecture, allows the connectivity to be addressed in small pieces and for that capability to be exposed through the API.

- Greater agility through loose coupling of systems: Separate API tiers allows a different level of governance and control to exist at each layer, making possible simultaneous loose-tight coupling.

Also, by referring paper [35], other significant benefits of using API based Connectivity are:

- By exposing data assets as a services to a broader audience, IT can start to become a platform that allows lines of business to self-serve.
- Increases developers' productivity through re-use across different applications.
- Deeper operational visibility.
- Highly scalable, global presence, highly productive, agile, real-time solution, etc.

Therefore, future technologies (like: SaaS, Mobile, IoT), where enterprises like ethio telecom, could use to engage with their customers or partners on a daily basis; denotes that the role of using the API-Led connectivity seems to be a crucial integration strategy.

Applying API-Led connectivity for integration of services and applications, also uses some of the principles of SOA based integration solution. It gives freedom for developers throughout the business to compose and recompose reusable services, and to build projects rather than the traditional top-down and heavyweight SOA based approach.

B- Varied results of EAI Solution Patterns:

This part discusses the reflections on the varied results found for the first two case studies and their advantages or disadvantages of finding different results.

Table 39, shows the summaries of the selected EAI solution patterns and integration frameworks for the tested and validated three case studies done in Section 5.2.

Table -39- Summery of Optimally selected EAI Solution Patterns

Type	Case Description	Selected EAI Solution	
		Solution Patterns	**Integration Frameworks**
Case 1	Current-UVC's Integration	Salesforce **Remote Call-In** and **Batch Data** Sychronzation	**Apache Camel** with ServiceMix, CXF & ActiveMQ Framework
		SAP 3's SOA based Webservice Solution	
Case 2	New Service DT Integration	Saleforce **Batch Data** Sychronization	**Mule** Anypoint Framework using Mule ESB.
		SAP 3's **SOA based** Webservice Solution	
Case 3	Future Technologies Integration	Mulesoft's API Led Connectivity	

Each of the first two cases shown in Table 39 (i.e. Case 1 and 2), have two selected EAI solution patterns and one integration framework choices. However, the third case has only one selected EAI solution pattern (which is an API-Led Connectivity).

For the first case study, after passing through the sub-processes (i.e. starting from business process requirements through modeling and mapping of data and interfaces till EAI solution patterns selection) done by the proposed selection framework, both synchronous and asynchronous type of data transfer is found and needed. And, the EAI pattern selection activity performed in Sub-sections 5.2.1 and 5.2.2, signifies that both SOA web based service and Salesforce solution patterns fulfills the requirements. At present, the SOA based web service is already implemented by the vendor. Whereas, the Salesforce solution also fulfills the requirements' specification.

Also, when observing the second case scenario, after going through the framework's sub processes or activities, an asynchronous or batch type of data transfer becomes important. For that both SOA based web service and Salesforce batch data synchronization solution pattern accommodates it.

In each of the first two case studies, two EAI solution patterns have been proposed.

One of the reason for this is that each of the various proposed EAI solution patterns (i.e. SOA, and Salesforce), fully complies the scenarios' EAI requirements studied and discussed in Sub-sections 5.2.1 and 5.2.2, autonomous, developed and also deployed by different solution

providers. In the meantime, they mostly could work integrated and interoperated with each other.

The second reason is that the financial strategies and capabilities of the company in relation to the EAI solutions' providers, have not been examined and then used as a selection mechanism. This is because of the confidentiality in getting some company documents related to the matter, and also due to time constraints.

In order to do the financial capability and strategy, it first needs floating of Request for Quotation (RFQ), for collecting the investment needed for deploying the proposed solutions and then compare and select based on their offer both technically and financially. This needs lots of time, first for RFQ preparation to invite world's solution providers, and then collect their proposed solutions' offer; and finally examine and select one optimal EAI solution pattern among competitors based on their quotation.

Whereas, for the third scenario, an API-Led Connectivity is chosen as a solution. As been observed from the EAI's specification and proposed solution and even from the technological matters discussion and argument, the API based connectivity is the future trend for the looming technologies integration. These connectivity could also attain and works with the existing integration solutions.

5.3 Expert Review

This Section describes the validation results of experts. First the approach including the summary of questioners with validation quality measures and the participated experts are described. Secondly, the results and findings are discussed.

5.3.1 Approach

On top of evaluating the framework using case studies, an expert review validation is performed as an additional validation technique. The expert panel is done by the company's selected internal professionals and managerial staffs. This qualitative form of expert review is performed in order to assess the validity of the scientific artifacts of the framework. A chance is also given to suggest or come up with new ideas or possible improvements.

Participated Experts:

Table 40 shows the internal experts position and profession that are participated in the validation process.

Table -40- Experts and their Position in the Company

Professional or Managerial Experts (Coded)	Position in the Company
E1	IT Services Design Officer
E2	IT Services design Manager
E3	IT Architect Specialist

Semi-Structured Interviews:

The expert reviews are done using semi-structured questioners. The respondents first received the framework's solution having detailed activities and deliverables. A series of questions used for validating the framework based on quality metrics has been forwarded. Table 41 summarizes the basic requirements and quality criteria that has been used for validation purpose.

Table -41- Framework's basic Requirements & Validation Metrics

Framework's Basic Requirements	Framework's Quality Metrics
Assists in defining requirements for EAI integration solution patterns,	➢ Completeness ➢ Consistency ➢ Efficiency ➢ Reliability ➢ Applicability
Chooses a technological solution and organizational measures suitable for specific integration scenario,	
Identifies the flow of data and processes from sender to receiver,	
Reduces the data and process redundancy,	
Chooses EAI solutions pattern based on company's project and financial strategic capabilities,	
Chooses an EAI solution pattern, based on the organization's Strategic vision, and	
Gives advice on issues after selecting integration solution.	

5.3.2 Results and Findings:

The feedback obtained from the experts' validation are summarized as follows. However, detailed overview of the questions and answers is placed in Appendix **L**.

A- Framework's Quality Metrics:

According to consulted experts, the framework appeared to be cohered and the framework fragments looks well integrated. Almost all of the framework fragments appeared to be available and no contradictions/inconstancies have been found by the experts' responses.

But, somehow the modeling activities looks like to be time consuming which would cause problems on practical applications. The experts argued that doing all the complete data and process modeling might not be very efficient.

But, as a counter argument, first it is possible to say that doing only partial modeling activities of data, business processes and application would make some integration activities to be overlooked or exempted.

Secondly, the modeling activities are one time job activities. That means, the modeling of data and processes are done just once when there is an EAI integration selection project.

B- Framework's Fragments:

The experts generally agreed upon the identified set of processes and their deliverables or outcomes. The overall idea beginning from elicitation of requirements, studying on the environment's context, defining integration quality, modeling, mapping, and then defining EAI specification and selection are good suggestions.

Nevertheless, the experts also recommended to include testing as one last activity. As it is believed that on top of having the selected solution patterns, testing the solution with some prototype is very significant and used to clarify things not seen.

C- Framework Requirements and Environmental Context:

The framework activities started by defining integration requirements and by considering the current status or context of the company, is suggested to be good.

D- Modeling and Mapping:

All the modeling activities included in the framework are found to be important. Moreover, the mappings done among the models provides all the necessary information. Therefore, no further modeling activity is proposed by the experts.

E- Restructuring:

The experts believe that the restructuring of activities are necessary in order to avoid very complex solution patterns by optimizing the modeling activities based on the study of gaps on integration.

F- Integration quality and Future Technology Considerations:

Defining quality criteria for integration and parameters for future integration technologies, is found very important by experts. This confirms the generic nature and flexibility of the framework; accommodating not only the present but also the forthcoming integration technologies.

G- EAI Solution's Pattern Selection and Advisory:

The steps found in the selection phase based on EAI requirements specification, are considered to be "complete" by the experts. It is known that every solution pattern has its own distinctiveness and added values. Thus, specifying requirements for selection would give space for trade-offs and debates, and then finally make search for optimal solution's pattern. This would be archived as a record and be used for advisory and related purposes.

H- Practical Applicability:

All of the experts agreed that this proposed framework is practically applicable. But, depending on situations, the modeling and restructuring of activities would take much time and effort to accomplish.

As a counter argument it has already been discussed that the mapping, modeling and restructuring of activities are one-time jobs and have great benefits in order to have a deep understanding of the integrated enterprise systems data and business processes for integration. Hence, doing all the framework's activities would also make things not to be overlooked or exempted.

I- Summary of Framework's Validation based on Metrics:

Table 42 summarizes the experts' validation result based on metrics, for the proposed EAI solutions' pattern selection framework.

Table -42- Experts' Framework Validation Summery using Metrics

Validation Metrics	Evaluation by Experts
Completeness	**Complied**. But, add "Proof-of-Concept" of solution providers.
Consistent	**Complied**. It has cohered and sequentially organized set of processes and deliverables.
Efficient	**Not fully Complied**. Since, the internal experts assumed to be time taking when doing full modeling and restructuring activities. This is based on the experts judgment in sub section 5.3.2, part A.
Applicable	**Complied**. But suggested to add "Integration Testing" at different stages.
Reliable	**Complied**.

Recommendation:

The experts assumed that doing the full modeling and mapping of activities is time taking and not fully efficient. But, it is counter argued in sub-section 5.3.2 part H that doing the modeling and restructuring of activities have so many advantages.

However, this paper also recommends to automate the mapping and modeling activities using software techniques, for lessoning the efforts it takes in doing the activities and then minimize the time it takes.

5.4 Chapter Summary

This chapter first tests and validates the proposed EAI solution patterns selection framework using three different case studies; in which: the first case study is the integration of the existing UVC service, the second case study is the integration of the upcoming service (i.e. the DT service) with the current enterprise systems and the last case study is the integration of the future or forthcoming integration technologies (like: IoT, Big data, SaaS, Mobile, and others).

Each of the case studies are done and their EAI solution patterns have been proposed, by following the framework's sequential sub activities.

Secondly, the test results' (i.e. the selected EAI solution patterns) reflections, based on the validation process of the three case studies, have been discussed categorizing it in two parts. These are: in relation to the technology matters and the varied results of the EAI solution patterns.

Eventually, expert review done by the company's selected design experts, service, and managerial staffs, using semi-structured interview questions, have been incorporated. The review incorporates their reflections and feedbacks combined with the researcher's responses, counter arguments, and recommendations.

Chapter Six: - Conclusion, Research's Limitation, and Future Works

6.1 Conclusion

The conclusion of this research is organized and done in a way by responding to the research questions found in Section 1.4. It is noted that before starting of the research project, a main research question is formulated. However, in order to respond the main research question, the below sub-questions have been formulated and answered primarily. This conclusion is written based on a style adopted in [41], and modified for attaining the research questions.

Sub-question 1: What is the current practice of ethio telecom in selecting systems integration pattern?

Based on the requirements studied and gathered from the company, it is confirmed that requirements for proposal (RFP) is first prepared and then based on that the EAI solution's pattern would be proposed by the solutions' providers or vendors. Thus, at present a traditional way of solution proposition is entertained.

Sub-question 2: What selection criteria and characteristics for EAI solutions pattern are identified in literatures and how can the existing EAI integration pattern selection methods be used and adopted for the company's selection framework?

The answer for this question is provided through Sections 2.4, 2.5 and 2.6. According to the reviewed literatures in Section 2.5, the main technical selection criteria for the EAI solution patterns are: 1) Category in EAI Taxonomy, 2) Integration style, 3) Integration Level, 4) Layers of Integration, 5) Data abstraction level, 6) Degree of Integration, 7) Target Applications, and 8) Quality of Integration. In addition, the general category measures, which are: the SoS scope, Current development context, and integration purpose needs to be studied.

The parameters for the framework are then defined considering the above selection criteria and characteristics of the EAI solution patterns. These characteristics are also proven to be relevant based on performed case studies and also by the internal experts' reviews. Hence, defined list of selection characteristics are applied when outlining the EAI solutions' pattern requirements sub-activity of the framework.

Sub-question 3: What method engineering approach should be used in order to construct the selection of EAI solutions' pattern framework?

Section 3.4 of the paper, briefly explains the kind of method engineering used to construct the framework. First, an overview of the *method engineering* is elaborated, then the selected type used for preparing this framework is clearly stated. Thus, by reviewing different literatures which considers the situation of the company, an *assembly based situational method engineering* approach is selected and implemented for developing this framework.

Research question: How could ethio telecom systematically select an optimal EAI solution pattern?

This main research question is answered by preparing and then validating the proposed framework which is used for selecting an optimal EAI solutions' pattern for enterprise systems' integration.

The paper first answers and obtains insights on what EAI and EAI solution pattern means. Further understanding is then performed by studding the characteristics for the EAI solution patterns. A *method engineering*, specifically an *assembly-based*, is selected and studied for the development of the framework; followed by the study of the framework's determining factors. Then, the important framework fragments are identified, selected, and assembled; considering the company's requirements and development context.

This new framework which begins with the framework's processes assembly, followed by the phases and activity descriptions, are elaborated through Sections 4.2 to 4.5 in detail.

For the purpose of validation, three case studies have been conducted, and then internal expert reviews are also done. The findings of the case studies and the results of the reviews based on validation metrics, are described in detail in chapter five.

6.2 Research's Limitation

This research is done in a form of qualitative research approach. Although, this gives ability in gaining detailed information starting from finding of the requirements for the framework fragments, to that of reviewing the solution by the internal experts; it has some hole for interpretation by researchers.

External expert validation is one point of attention that is not done for this research solution. Likewise, multiple case studies could have been done in order to increase the validity of the solution.

However, due to the limited time for the research, this was not possible during the validation process.

6.3 Future Research Area

This research has worked with effort to achieve the objectives of the study. But, the below are some of the potential future work areas to the continuation of this work.

A- "Examining specific details on how to execute each activity of the selection framework". This would enrich the proposed framework with best practices, professional experience, and common pitfalls.

B- "Investigating on distinct characteristics and selection criteria of future technologies for integration". This would fully complete the selection of EAI solution pattern in relation to future technologies considerations.

C- And, "Studding the long-term effects of implementing the selected EAI solutions pattern". This would determine the wide-ranging successfulness of the selection framework and its solution patterns.

Bibliography

[1] H. R. I. Communications, "e-Magazine Vol-3, No. 1-2017," ethio telecom, Addis Abeba, 2017.

[2] J. Fenner, "Enterprise Application Integration Techniques," 2011.

[3] A. H. A. Tariq Rahim Soomro, "Challenges and Future of Enterprise Application," *International Journal of Computer Applications,* vol. 42–No.7, March, 2012.

[4] K. S. C. B. N. J. K. Rick Kazman, "Understanding Patterns for System of Systems Integration," Carnegie Mellon University and IEEE, Pittsburgh, 2014.

[5] J. A. Gulla, "Introduction to Enterprise Systems," Norwegian University of Science and Technology, Norway, 2005.

[6] M. E. v. S. M. L. a. R. B. Marcel A.P.M. van den Bosch, "A Selection-Method for Enterprise Application Integration Solutions," in *Springer-Verlag Berlin Heidelberg*, The Netherlands, 2010.

[7] M. L. M. a. C. Tanis, "The Enterprise System Experience from Adoption to Success," in *The Enterprise System Experience from Adoption to Success*, Netherlands, 2000, pp. 173-205.

[8] K. Röppänen, "Requirements for an Enterprise Application Integration Tool," Tampere, May 2013.

[9] D. S. Linthicum, Next Generation Application Integration: From Simple Information to Web Services, Boston: Addison Wesley, August 2003.

[10] F. S. Abdullah AL-Malaise AL-Ghamdi, "Enterprise Application Integration as a Middleware: Modification in Data & Process Layer," in *Science and Information Conference 2014*, London, August, 2014.

[11] T. A. w. M. a. Z. M. Alemtsehay Ke., "Assessment of Enterprise Systems Integration Problems in ethio telecom," Seminar Final Report, Addis Abeba, 2018.

[12] salesforce, "Integration Patterns and Practices," *Version 42.0, Spring '18,* 11 January 2018.

[13] M. Themistocleous, "Justifying the Decisions for EAI Implimetations: a validated proposition of influential factors," *Journal of Enterprise Information management,* vol. Vol 17, no. 2, pp. 85-104, 2005.

[14] Z. I. Marinos Themistocleous, "Towards a Novel Framework for the Assessment of Enterprise Application Integration Packages," in *36th Hawaii International Conference on System Sciences*, Brunel, 2003.

[15] W. L. a. V. Shankararaman, "An Enterprise Integration Methodology," IEEE Computer Society, Singapore, April, 2004.

[16] M. T. a. Z. Irani, "Novel taxonomy for application integration," *Benchmarking: An International Journal,* vol. 9, no. 2, pp. 154-165, 2004.

[17] G. H. a. B. Woolf, Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions, Bethesda, MD: Addison Wesley, August, 2003.

[18] M. S. a. F. H. Sjaak Brinkkemper, "Assembly Techniques for Method Engineering," in *CAiSE '98 Conference*, Berlin-Heidelberg, 1998.

[19] R. K. Yin, "Case Study Research Design and Methods," SAGE Publications, New Delhi, 2003.

[20] T. H. C. B. M. C. A. R. Sarah Beecham, "Using an Expert Panel to Validate a Requirements Process Improvement Model," University of Hertfordshire, UK, 2005.

[21] M. S. A. F. H. SJAAK BRINKKEMPER, "Meta-Modeling Based Assebly Techniques For Situational Method Engineering," *Information Systems Journal,* vol. 24, pp. 209-228, 1999.

[22] I. M. a. J. Ralyte, "Situational method engineering: combining assembly-based and Roadmap-driven approaches," *Springer-Verlag,* vol. 11, pp. 58-78, 2006.

[23] S. B. J. S. a. J. V. Inge van de Weerd, "A Situational Implimentation method for Web-based Content Management System Applications: Method Engineering and Validation in Practice," in *Software Process Improvement Practice 2006; 11: 521–538*, Nijmegen, Netherlands, July, 2006.

[24] I. v. d. W. a. S. Brinkkemper, "Meta-Modeling for Situational Analysis and Design Methods," in *5th international conference on Global Perspectives on Design Science Research*, The Netherlands, 2009.

[25] E. K. a. R. T. V. BRAGA, "Patterns for Enterprise Application Integration," in *9th Latin-American Conference on Pattern Languages of Programming*, Brazil, September, 2012.

[26] G. R. a. A. E. Abdallah Mohamed, "COTS Selection: Past, Present, and Future," in *14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems*, 2007.

[27] R. C. a. F. R.-F. Rafael Z. Frantz, "A methodology to evaluate the Maintainability of Enterprise Application Integration Frameworks," *Interantional Journal on Web Engineering and Technology*, vol. 10, no. 4, pp. 334-354, 2015.

[28] T. S. a. F. Leymann, "A Framework for Executable Enterprise Application Integration Patterns," in *New Challenges and Industrial Approaches Conference*, Berlin, Germany, 2008.

[29] K. Indrasiri, "The Evolution of Integration: A Comprehensive Platform for a Connected Business," WSO2 Con., New York, 2017.

[30] e. Consortium, "Integration Methodology Workshop," EAI Industry Consortium, Feb., 2004.

[31] R. B. a. S. B. Henk Koning, "A Lightweight Method for the Modeling of Enterprise Architectures: Introduction and Empirical Validation," Utrecht University of Information and Computing Systems, Netherlands, 2008.

[32] A. S. a. J. S. Robert Thullner, "Implementing Enterprise Integration Patterns Using Open Source Frameworks," in *3rd IFIP Central and East Europe Conference on Software Engineering*, Vienna, 2010.

[33] F. S. A. H. T. P. Danny Merkel, "Cloud Integration Patterns," in *4th European Conference on Service-Oriented and Cloud Computing (ESOCC)*, Taormina, Italy, Sept, 2015.

[34] M. T. Company, "API-Led Connectivity - The next Step in the evolution of SOA," Deloite Tech Trends, UK, 2016.

[35] M. Daly, "Reasons why ESB Led Integration out & API Led Integration In," Mulesoft Company, UK, 2017.

[36] M. T. Company, "Anypoint Platform the Agility Layer for your Business," Mulesoft, UK, 2018.

[37] R. M. a. J. McKendrick, "The Rising values of APIs: MuleSoft's Digital Transformation Predictions," Mulesoft, UK, 2018.

[38] R. D. a. C. R. Jolita Ralyté, "Towards a Generic Model for Situational Method Engineering," in *International Conference on Advanced Information Systems Engineering*, Geneva, Switherland, 2003.

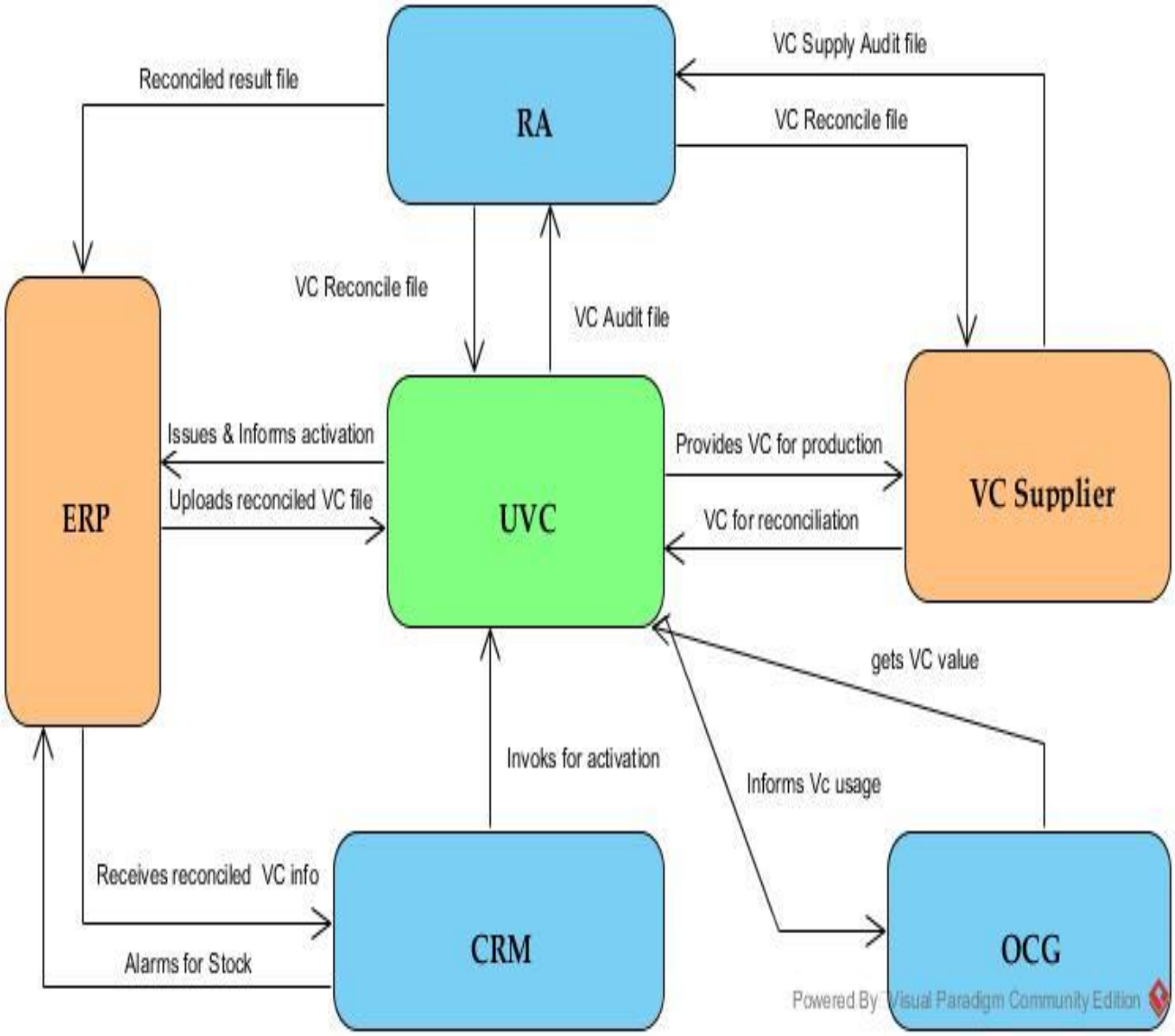
[39] J. A. A. A. S. C. P. C. T. N. M. L. P. K. Mark Endrei, Patterns: Service Oriented Architecture and Web Services, USA: IBM Redbooks, April 2004.

[40] C. Ibsen, "The Top Twelve Integration Patterns for Apache Camel," DZone Communities, New York, 2012.

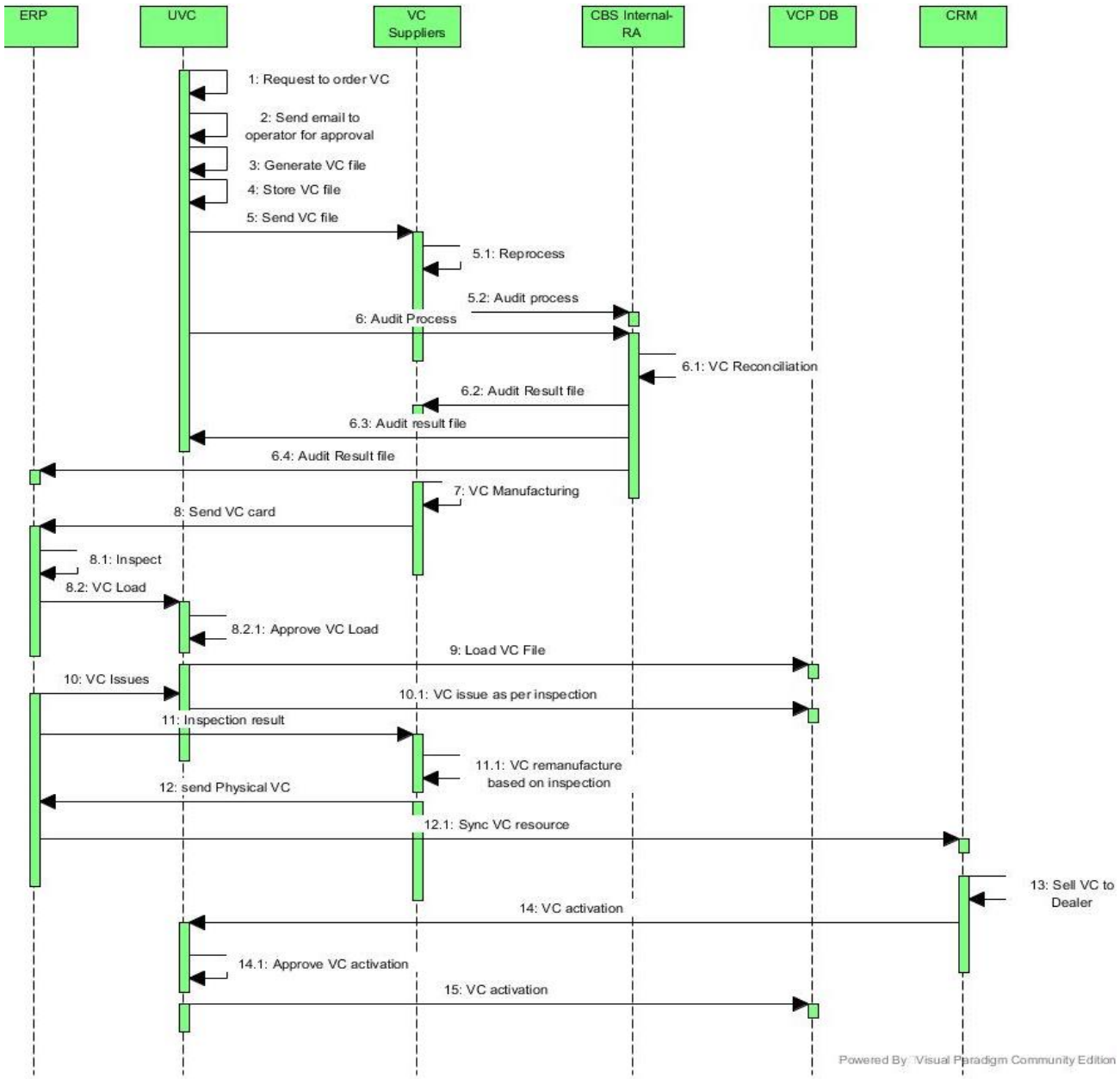
[41] R. P. RADHAKRISHNAN, "TOWARDS A NEW APPROACH FOR ENTERPRISE INTEGRATION: THE SEMANTIC MODELING APPROACH," University of Saskatchewan, Saskatchewan, Canada, May, 2003.

Appendixes

Appendix A: UVC's Enterprise Function Diagram



Appendix B: Sequence Diagram for the UVC' Provisioning and Activation processes.



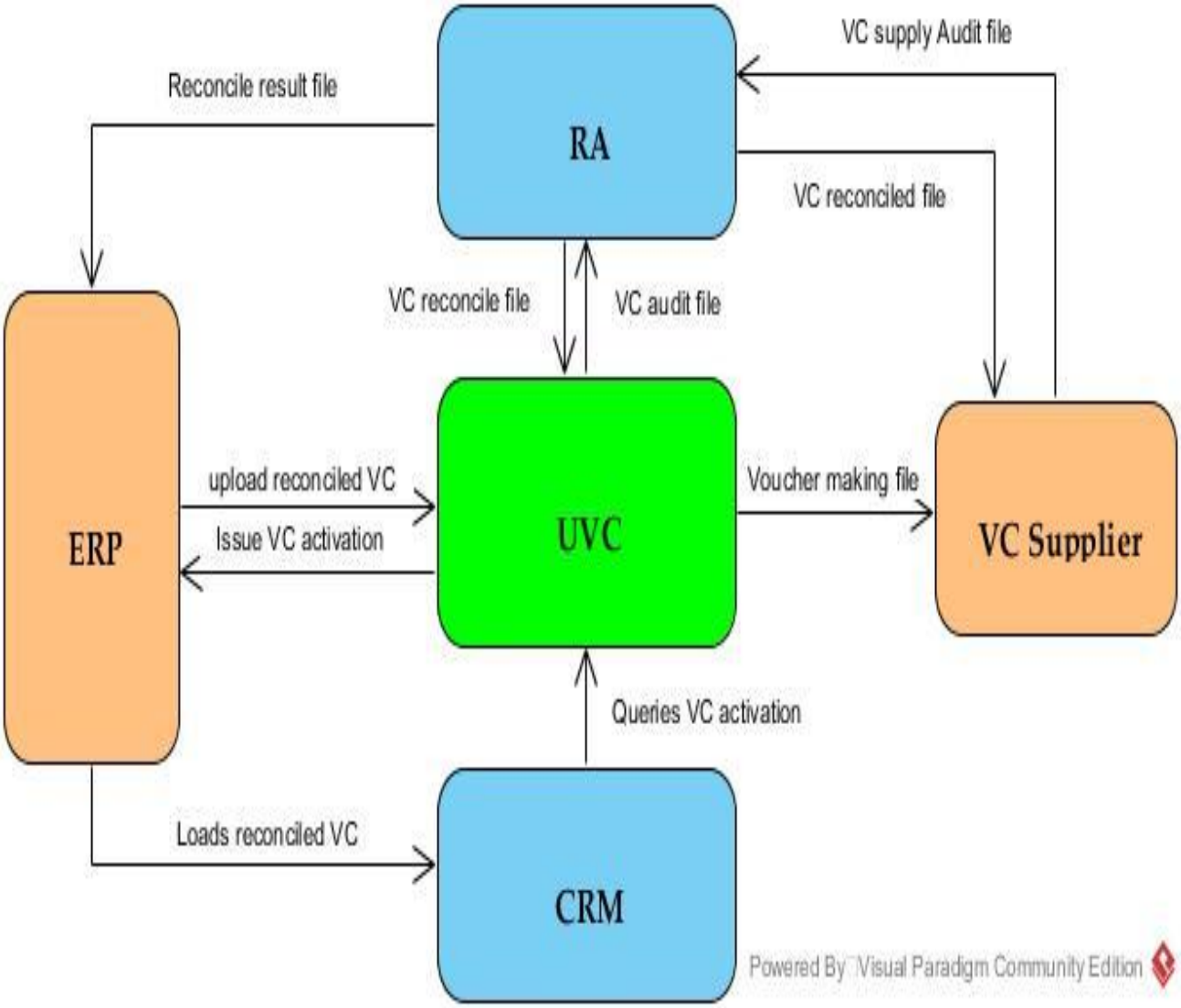
Appendix-C: Business Processes flows for UVC making file, activation, and provisioning

Seq. No.	Flow Name	Description	Related Module
1	Request to order voucher card	Operator logins to UVC and creates a request to voucher card generation.	UVC
2	email for Approval	When operator creates a generation task. The system will send email to related operator to prompt him/her for approval. The administrator then approves the order request.	VMP
3	Generate voucher making file	VMP generates voucher making file and encrypts the file.	UVC
4	Store voucher making file in the hard disk.	After the VMP generates voucher making file, the files will be stored in hard disk.	UVC
5	Send VC file	UVC sends to VC supplier for voucher card manufacturing. At the Same time the UVC sends **batch no** information and the **serial no's** to ERP.	UVC,VC-Supplier
5.1	Re-Process	After received voucher making file, the VC supplier will decrypt the file to get voucher card information.	VC-Supplier
5.2	Audit file	VC-Supplier generates audit file and sends to FTP Server, RA system will collect it from FTP server for reconciliation.	VC-Supplier, CBS internal RA
6	Audit Process	UVC generates audit file and sends to FTP Server, RA system will collect it from FTP server for reconciliation	UVC, RA
6.1	CBS internal RA	CBS internal RA will compare these two audit files to ensure the VC making file is consistent for each other.	CBS internal RA
6.2	Audit result file	After reconciliation, the RA sends audit result file to VC-Supplier. If the audit result is successful, the VC- supplier starts voucher card manufacturing. If the audit result is failed, the operator will fix the error manually.	VC-Supplier, CBS internal RA

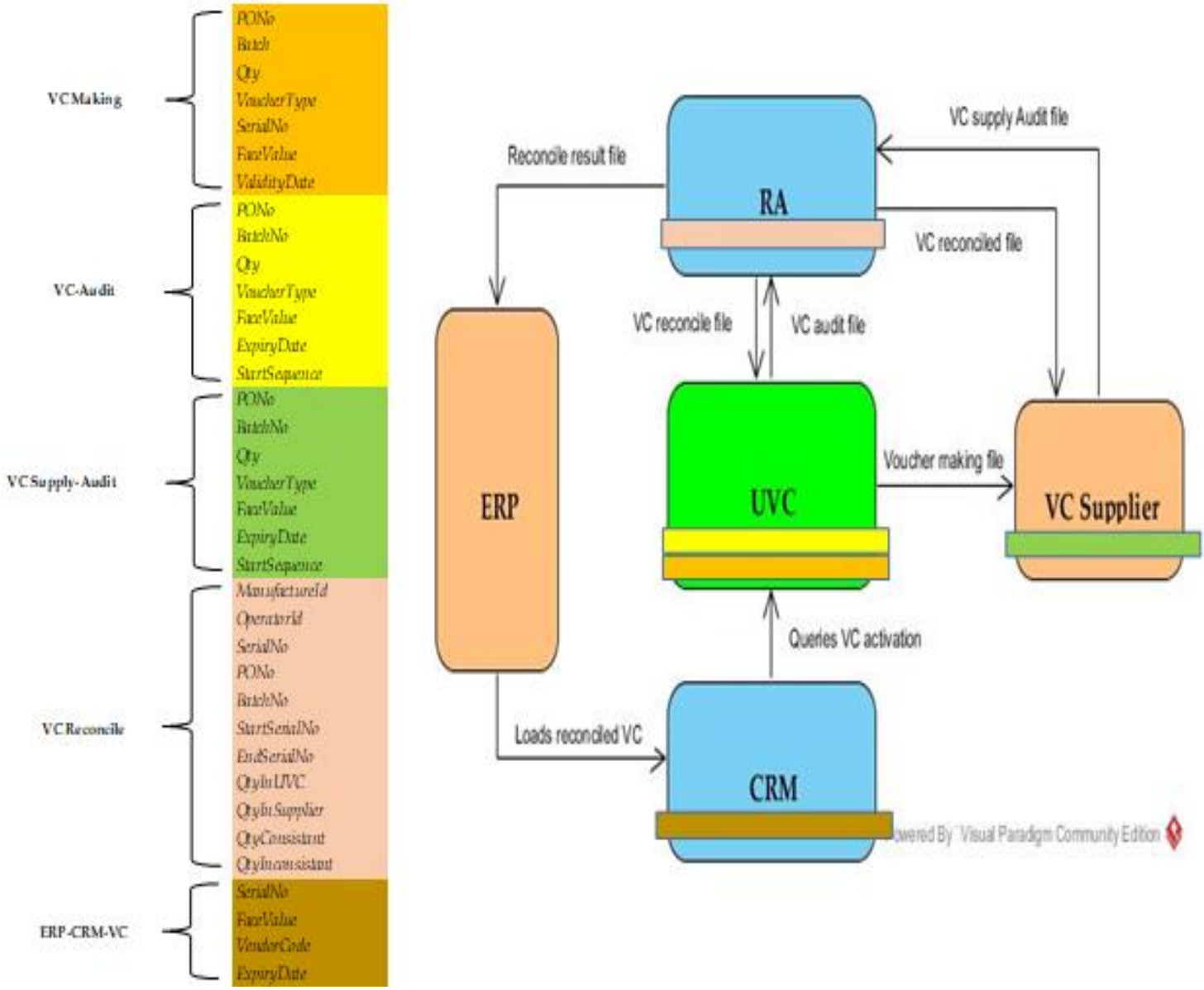
Seq. No.	Flow Name	Description	Related Module
6.3	Audit result file	After the reconciliation, the RA sends audit result file to UVC through FTP. If the audit result is successful, the order request continues. If the audit result is failed, the operator will fix the error manually.	CBS internal RA, UVC
6.4	Audit result file	After the reconciliation, the RA sends audit result file to ERP. If the audit result is successful, the order request continues. If the audit result is failed, the operator will fix the error manually.	CBS internal RA, ERP
7	Voucher card manufacturing	Lastly, the VC supplier starts voucher card manufacturing if audit result is successful.	VC-Supplier
8	Send Physically VC	The VC-supplier then sends manufactured voucher card to ERP for inspection.	VC-Supplier, ERP.
8.1	Inspection	ERP will inspect voucher card based on audit result file and physical voucher card.	ERP
8.2	VC Load	ERP sends to VMP of UVC to load voucher file. The UVC will reject the loading request if voucher card already loaded in UVC.	ERP,UVC
8.3	Approve VC loading	System sends email to related person to prompt login to VMP for approval of voucher file loading	VMP of UVC
9	Load VC file	After approval, UVC loads all voucher cards into database based on batch no.	VMP,VCPDB
10	VC file issuing	ERP sends request to VMP to issuing voucher file, at the same time ERP will upload available voucher card list to VMP, VMP will then issue voucher file accordingly.	ERP,VMP of UVC
10.1	VC issue as per inspection	When receive voucher card issuing request, UVC issues voucher card file based on inspection result file. The available voucher card will be issued. The rejected voucher card will be skipped automatically.	VMP,VCPDB

Seq. No.	Flow Name	Description	Related Module
11	Inspection result	After finishing inspection, ERP informs the result to VC-Supplier with rejected voucher card list. The ERP will generate the unavailable voucher card list, and send to VC-supplier for re-manufacturing.	ERP, VC-Supplier.
11.1	VC manufacturing based on inspection result	The VC supplier re-manufacture voucher card based on inspection result file.	VC-Supplier
12	Sends VC cards physically	The VC Supplier sends the manufactured VC cards physically to ERP for processing.	VC-supplier, ERP
12.1	Synchronize Card resource	ERP synchronizes the available voucher card list to CRM for sale.	ERP, CRM
13	Sells voucher card to dealer	The CRM operator sells voucher card to dealer.	CRM
14	VC activation	After dealer payment in CRM, CRM will send message to VMP for voucher card activation. In this case the voucher card activation does not need approval. If UVC active successes, the CRM can sell these voucher card. If UVC active failed, the UVC should response the result with reason to CRM, and the CRM can't sell these voucher cards.	CRM,VMP of UVC
14.1	Approval for Voucher Card activation	If UVC operator wants to activate voucher manually, system will send email to related person to prompt for approval.	VMP,VCP DB
15-	Voucher Card activation	The order requests approval, the operator activates the voucher card through VMP GUI.	VMP, VCP DB

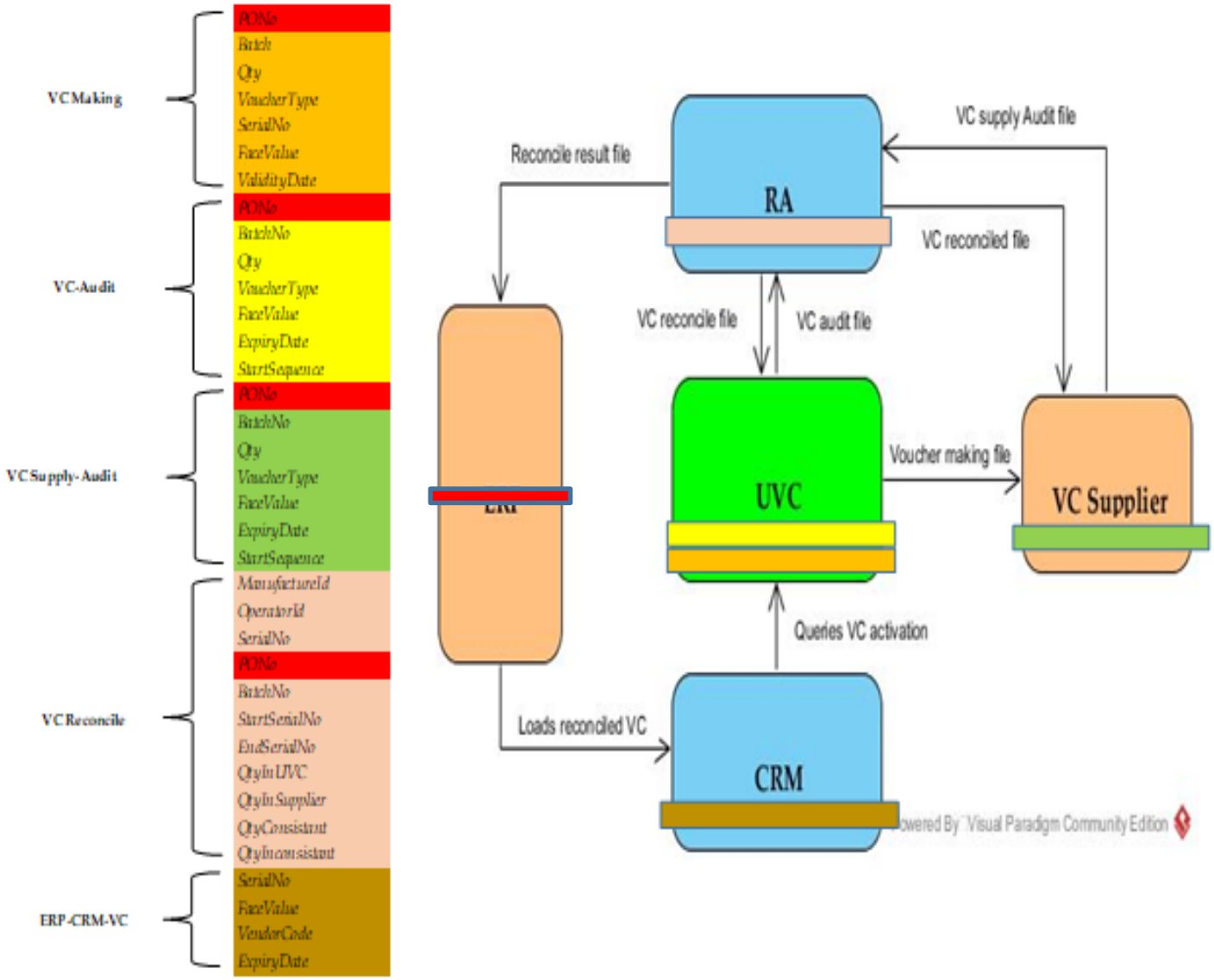
Appendix-D: **VC card making and reconciliation sub-activity from the VC scenario**



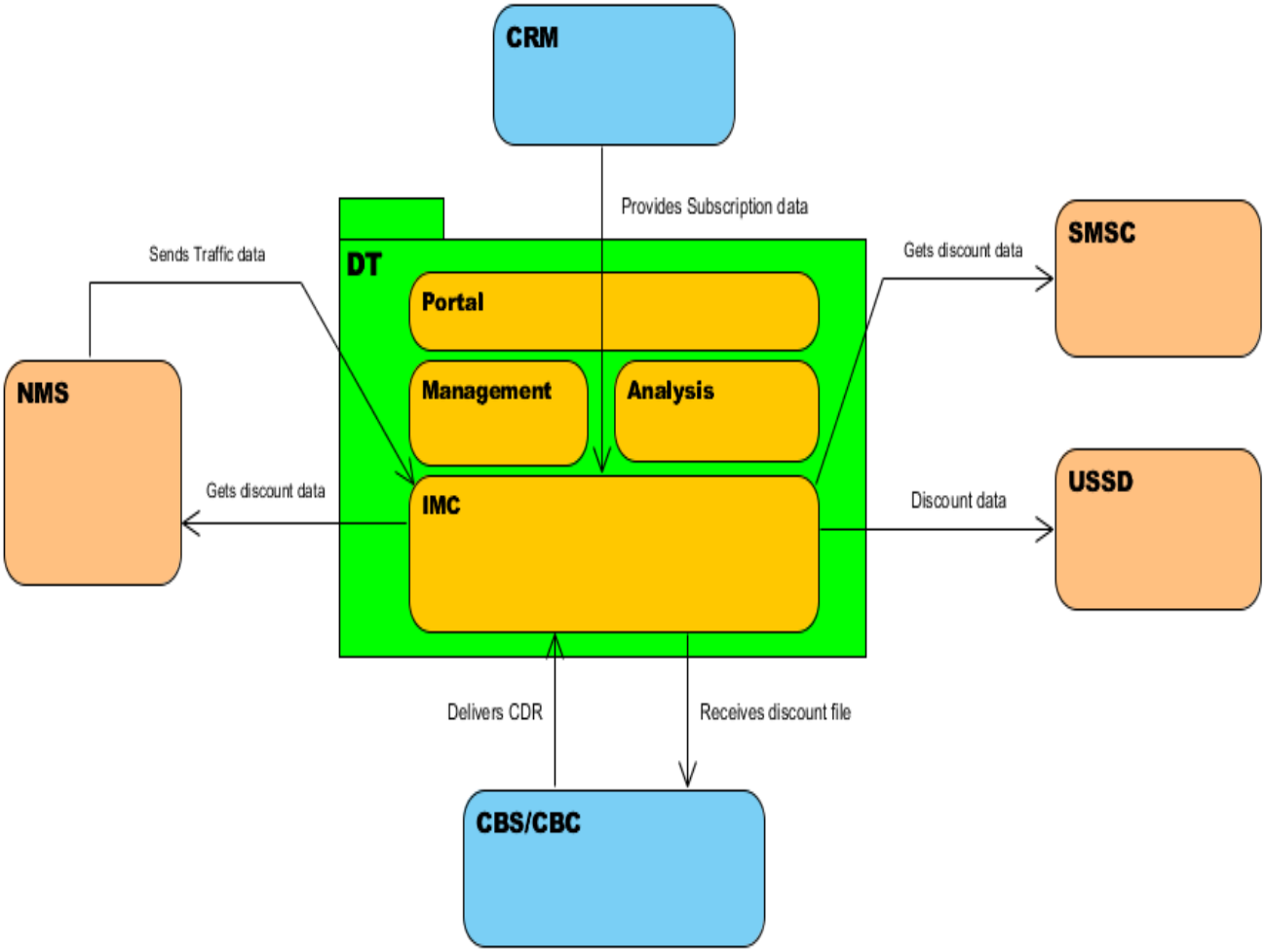
Appendix E: Top-Down Mapping of the UVC making to reconciliation Scenario



Appendix-F- Bottom-Up Mapping of the UVC making to reconciliation Scenario



Appendix G: EFD diagram of DT's discount service integration with external systems

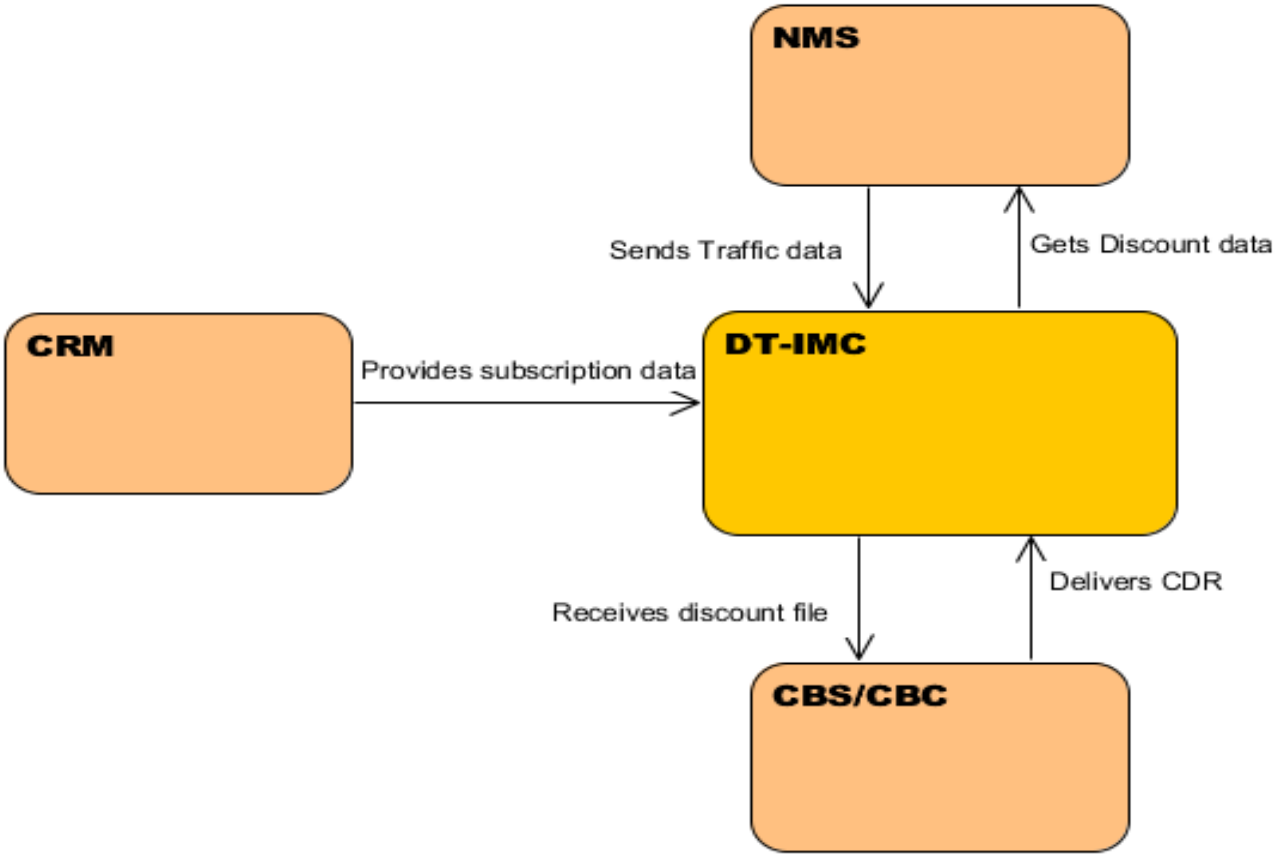


Appendix H: DT Business process flows with integrated systems

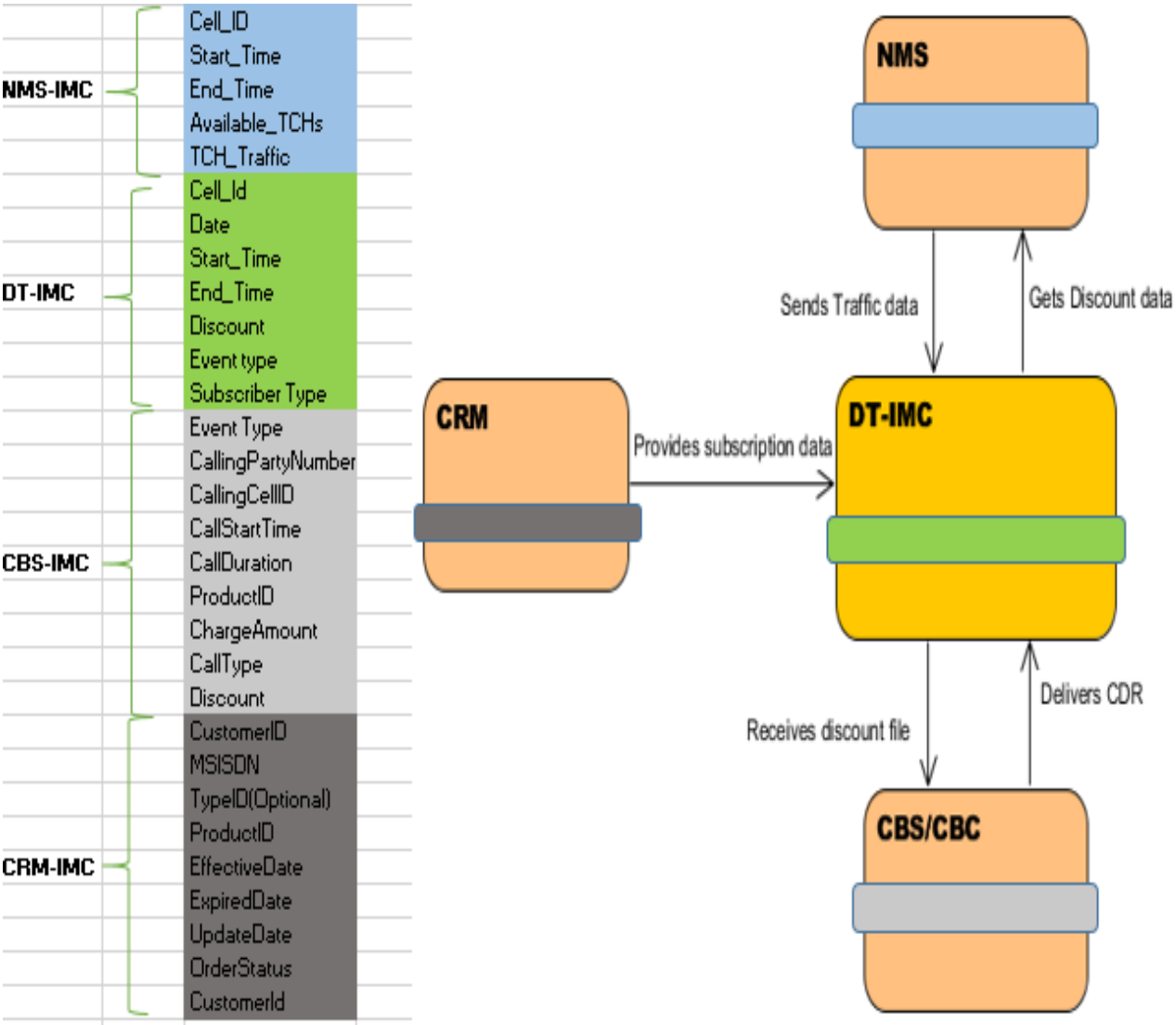
Basic Process Flows	Processes Description	Targeted modules
1	Existing subscribers would be registered for the DT service via available channels: (i.e. USSD, SMS, and or IVR).	CRM, CBS
2	DT system collects voice and data CDRs from CBS or Mediation Server every hour using ftp/ sftp interface.	CBS
3	DT system will collect (voice and data) cell traffic statistics from NMS/EMS/OSS systems every hour using ftp/sftp interface to calculate the optimal discount.	NMS and OSS (USSD, SMS)
4	DT also collects DT service subscription information from provisioning system or CRM system using ftp/sftp interface on daily basis to do some subscriber trend related reports.	CRM, Provisioning system
5	DT system generates voice discounts (on-net/off-net) for prepaid/postpaid subscribers, data discounts (price, volume, duration) for prepaid subscribers based on history traffic analysis, CDR statistics and configured discount policies respectively and delivers discounts to CBS daily or every hour using standard protocols preferably ftp/sftp protocol.	DT, CBS
6	DT system delivers discounts to CBS or NMS/OSS/BSC/RNC for DT discount broadcasting to subscribers every hour or daily.	CBS, Network elements
7	Subscriber could also send a USSD service code to query current DT discount based on current location of the DT subscriber. USSD system will trigger DT system to get the DT discount information whenever discount query requests will be received by USSD system.	DT, USSD,
8	When DT subscriber originates a voice call, during the call setup phase, CBS can send a USSD notification message to DT subscriber informing the applicable discount of the current call.	CBS, DT

Basic Process Flows	Processes Description	Targeted modules
9	DT system generates DT subscribers saving report on daily/weekly/monthly (these parameter can be configured) basis and send the saving report to the DT subscribers via SMSC as per the configuration.	DT, SMSC
10	DT system provides the web based DT discount query interface to Customer care to get the discount information for a particular Cell ID real time. No future discount will be shared with Customer care.	DT,CBC
11	When a DT subscriber initiates a data session, the discount applied will be based on subscriber's current location (Cell ID) and current time. The discount will remain the same for entire data session.	DT

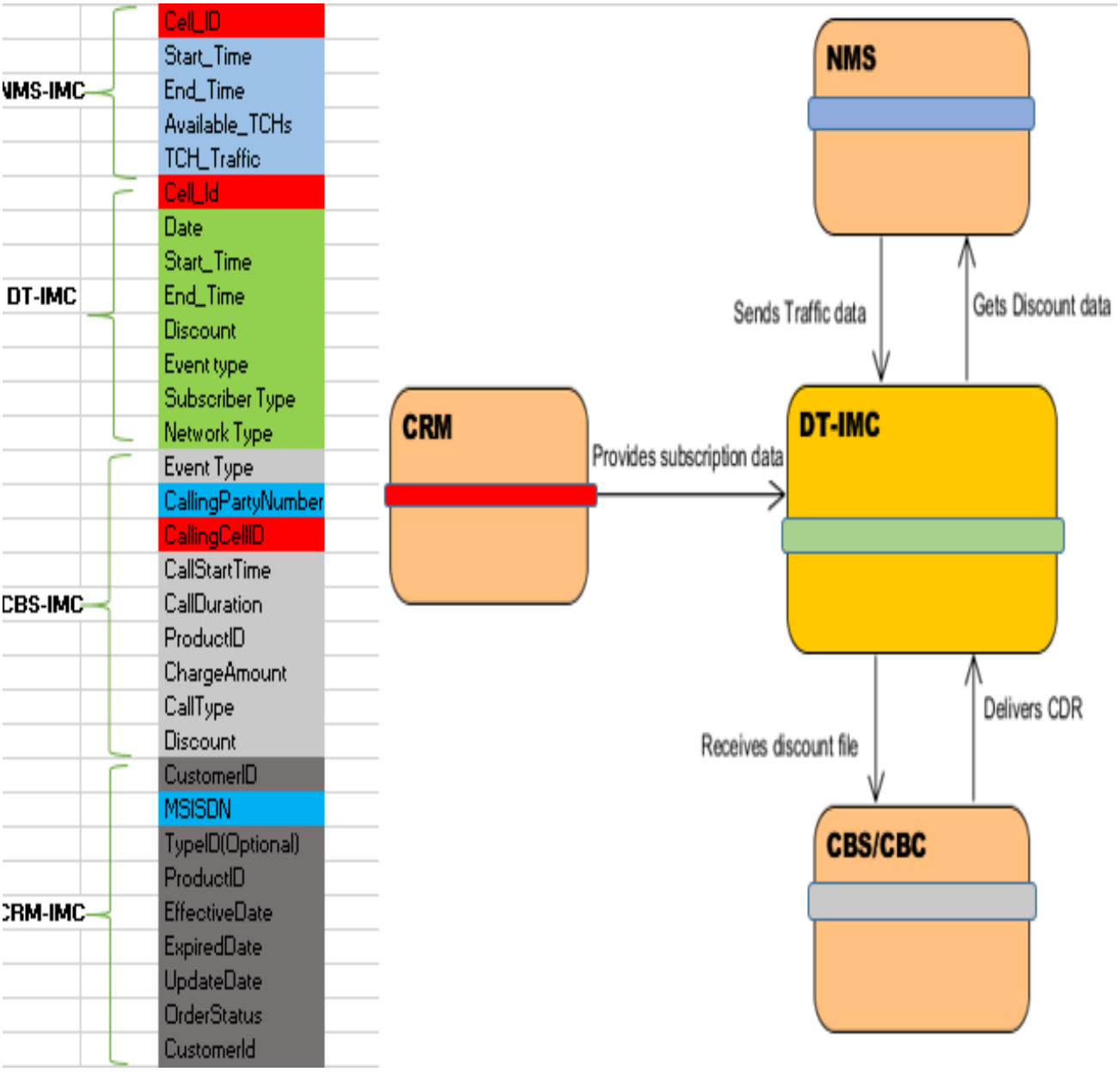
Appendix I: EFD diagram of DT-IMC integrated with selected external packaged applications.



Appendix J: Top-Down Mapping of DT service interoperated with external systems.



Appendix K: Bottom-Up Mapping of DT service interoperated with external systems.



Appendix L: Expert Review Questions and Answers

In this appendix, the questions prepared in semi-structured format, for receiving responses of internal experts are listed. Also, the feedbacks, remarks and suggestions of the experts for each questions are included. However, similar types of remarks are grouped and summarized together.

Q1.	**Which type of integration problems do you encounter in your daily practice?**
	- Missing of parts of systems that needs to be integrated. *- Knowledge issues on areas of application integration.* *- Problem on compatibility of protocols.*
Q2.	**Can the proposed selection framework provide some advice for the problems mentioned in Q1?**
	- The framework works, as it makes most of the steps more explicit. *- It helps to manage and control the flow of data between different systems.*
Q3.	**Do you believe that the described activities in the proposed framework solution are relevant to application-integration problems?**
	- Overall, most of the framework activities are relevant & taken as best practice. *- Granular points will also be addressed during deployment when properly applied.* *- In general, helps the company to use standardized framework.*
Q4.	**Does the framework missed any activities? If so, what is missing?**
	- Generally, no parts were found to be missing.
Q5.	**In your experience with application-integration, what other models than data, process and application interfaces/IT infrastructure do you use?**
	-We mostly follow the traditional approach for integration. *-Security (i.e. Application's security) models.*
Q6.	**Do you believe that all the currently used models in this selection framework, are important to select an optimal EAI solution?**
	- It depends, Some models can be done more thoroughly than others based on specific situation.
Q7.	**Is this selection framework suitable for the future (i.e. now unpredictable or forthcoming) application integration technology problems?**
	- The framework does not depend on specific technologies, it uses requirements identification approach, so be used for future integration. *-The generic nature of this framework is assumed flexible in order to apply for the future integration scenario.*

	-Yes, it is suitable, but with adjustment on each project.
Q8.	**In addition to mapping between data, processes, and interfaces & infrastructure, do you think additional mappings are necessary in understanding application-integration problems correctly?**
	- None is recommended.
Q9.	**The selection framework finally defines EAI requirements specification that would remain after performing restructuring activities. Do you think this approach is best suited for application integration problems?**
	- Yes, the approach of the framework is valid.
Q10.	**Based on the defined requirements, a series of generic software-selection activities will be performed. Do you think these generic software-selection activities are relevant to application patterns integration selection? If not, what kind of changes would you suggest?**
	- Yes, but with addition of "Proof of Concept" activity.
Q11.	**Based on your practical experience with application-integration problems, what do you think is in general missed in this selection-framework?**
	- Go/No-Go decision after each activity has to be considered. *- Risk of each activity (i.e. what happens when one activity of the framework is skipped or done improperly) should be considered.* *- Also, "Integration testing" at different stages is necessary.* *- Experts involvement on each solution selection.*
Q12.	**What is your opinion on the coherence and sequence of the activities of this method? Is there anything to adjust or change?**
	- In general, Coherence and sequence of activities is found to be good. All steps seem to have a logical dependency with each other. *- The modeling activities should first look at the processes, then the applications involved and then the data.*
Q13.	**Would you consider using this framework in practice for the future? If not, what limitations do you expect to encounter when using this selection framework?**
	- Yes, the framework helps defining requirements for selecting a specific integration topology. *- "Proof of Concept" should be selected in the end based on the selected EAI patterns solution.* *- Also, limitation on understanding and involvements of higher managerial staffs.*

Declaration

I, the undersigned, declare that this MSc thesis is my original work, and has not been presented for the fulfillment of a degree in this or any other university. And all sources and materials used for this thesis have been acknowledged.

Declared by:

Name: Thomas Abebe

Signature: _____

Date: November-14, 2018

This thesis has been submitted for examination with my approval as a university advisor.

Confirmed by advisor:

Name: Mesfin kifle (PhD)

Signature: _____

Date: November-14, 2018