

An Information Systems Reference Architecture for the CRM domain

André Gonçalo Batista Correia da Cruz

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisor(s): Prof. André Ferreira Ferrão Couto e Vasconcelos

Examination Committee

Chairperson: Prof. Miguel Nuno Dias Alves Pupo Correia
Supervisor: Prof. André Ferreira Ferrão Couto e Vasconcelos
Member of the Committee: Prof. Artur Miguel Pereira Alves Caetano

June 2015

Acknowledgments

In this section I would like to thank some people that in different ways contributed to this work. First of all I would like to thank my mother for along the years provide me the best education, for giving me the opportunity to be studying in this institution and for all the support that gave me.

I would also want to thank the supervisor of my master thesis, Prof. André Vasconcelos, by the orientation and guidance given that helped in completing this master thesis and the publication of the paper in the ICEIS conference.

This work wasn't possible without the help by AMA, more concretely the Eng. Maria João Marques, that provided the case studies used in this master thesis.

By last I would like to thank all my colleagues that helped with suggestions, with the reading of my work and with the encourage to finish this master thesis.

Resumo

Nos últimos anos, o foco das empresas mudou de foco no produto para foco nos clientes, resultando numa adesão por parte das empresas a soluções de Gestão da Relação com os Clientes.

Com esta adesão a soluções de CRM, espera-se que uma Arquitetura de Referência para este domínio forneça uma maneira de abordar os problemas que ocorrem habitualmente através de uma documentação das boas práticas arquiteturais. O objetivo neste trabalho é fornecer uma Arquitetura Aplicacional de Referência para o domínio CRM, de forma a assegurar a agilidade das organizações, e o contínuo alinhamento entre o negócio e os sistemas de informação. A Arquitetura de Referência final alcançada contém seis módulos de CRM e cinco sistemas, que interagem com o sistema de CRM: Módulo de Conta, Módulo de Vendas, Módulo de Marketing, Módulo de Atendimento, Módulo de Agendamento, Módulo de Administração, Portal, Contact Center, sistema de Gestão de Documentos e da Base de Conhecimento, sistema de Gestão do Fluxo de Trabalho e sistema de Relatórios e Análises.

Após a Arquitetura de Referência estar definida, esta é avaliada em dois casos de estudo da Administração Pública Portuguesa: o caso do Alto Comissariado da Migração e o caso do Espaços Cidadão.

O objetivo da avaliação realizada em cada caso, é validar a adequação da Arquitectura de Referência face a atributos de qualidade específicos. Os atributos de qualidade escolhidos na avaliação são: a facilidade de mudança, a facilidade de teste e do alinhamento.

A avaliação efectuada nos dois casos de estudo permitiram também, a identificação de um padrão arquitetural na Administração Pública Portuguesa no domínio de CRM, o padrão SIGA.

Palavras-chave: CRM, arquitetura de sistemas de informação referência, sistemas, módulos, avaliação.

Abstract

In recent years, the focus of the business changed from product focus to customer focus, resulting adherence to Customer Relationship Management solutions by companies.

With this adherence to CRM solutions, a Reference Architecture for this domain is expected to provide a way to approach usual occurring problems by documenting good architectural design practices. The goal in this work is to provide a Reference Application Architecture for the CRM domain, to ensure the agility of organizations, and the continuous alignment between the business and information systems. The final Reference Architecture reached contains six CRM modules and five systems, which interact with the CRM system: Account module, Sales module, Marketing module, Service module, Scheduler module, Administration module, Portal, Contact Center, Document and Knowledge Base Management system, Workflow Management system and Reporting and Analytics system.

After defining the Reference Architecture, we evaluated it in two case studies from Portuguese Public Administration: the High Commissioner for Migration and the Citizen Spaces.

The objective of the evaluation done in each case, is to validate the adequacy of the Reference Architecture in specific quality attributes. The quality attributes chosen to be measured in the evaluation are: change facility, test facility and the alignment.

The evaluation done allowed the identification of an EA pattern in CRM domain from the Portuguese Public Administration, the SIGA pattern.

Keywords: CRM, information systems reference architecture, systems, modules, evaluation.

Contents

- Acknowledgments iii
- Resumo v
- Abstract vii
- List of Tables xii
- List of Figures xv

- 1 Introduction 1**

 - 1.1 Business Environment 1
 - 1.2 Thesis Problem 2
 - 1.3 Research Methodology 2
 - 1.4 Objectives 3
 - 1.5 Document Structure 4

- 2 Related Work 5**

 - 2.1 Enterprise Architecture 5
 - 2.1.1 Enterprise Architecture Framework 6
 - 2.1.2 Reference Enterprise Architecture 7
 - 2.1.3 Methodology for Specific Architectures Generation 8
 - 2.1.4 Enterprise Architecture Principles 9
 - 2.1.5 Information Systems Quality Attributes 10
 - 2.1.6 Metrics and Heuristics for Information Systems Evaluation 12
 - 2.1.7 CRUD Matrix 15
 - 2.1.8 Critical Analysis 16
 - 2.2 Customer Relationship Management 16
 - 2.2.1 CRM Architecture Models 17
 - 2.2.2 CRM Features 19
 - 2.2.3 CRM Business Processes 27
 - 2.2.4 CRM Data Models 29
 - 2.2.5 Critical Analysis 32

- 3 Architecture Solution 33**

 - 3.1 Methodology followed for defining the Reference Architecture 33

3.2	Reference Architecture - Vision, Mission and Strategy (Step 1)	34
3.3	Reference Architecture (Step 2)	35
3.4	Reference Architecture Complete (Step 3)	40
3.4.1	Comparison of the two Reference Architectures	46
3.5	Principles Identified in the Reference Architecture	46
3.6	Critical Analysis	47
4	Evaluation	49
4.1	Evaluation Methodology	49
4.2	Case Studies	51
4.2.1	High Commissioner for Migration (ACM)	51
4.2.2	Citizen Spaces (CS)	58
4.3	Results Analysis	66
5	Conclusions	69
5.1	Research Questions Analysis	69
5.2	Major Contributions	70
5.3	Accessory Contributions	71
5.4	Limitations	72
5.5	Future Work	73
	References	78
A	ACM Business Processes	79
B	Information Entities Clustering	82
C	Case Studies Services Realization	83
D	Customer Relationship Management Data Models	87
D.1	Buttle Components	88
D.2	CRM Solutions Information Entities Mapped	89
E	Ref. Appl. Architecture CRUD Matrix	103

List of Tables

2.1	Sales Features Table from Cruz and Vasconcelos [2015]	20
2.2	Marketing Features Table from Cruz and Vasconcelos [2015]	22
2.3	Customer Service Features Table from Cruz and Vasconcelos [2015]	23
2.4	Reporting and Analytics Features Table from Cruz and Vasconcelos [2015]	24
2.5	Integration Features Table from Cruz and Vasconcelos [2015]	25
2.6	Security Features Table from Cruz and Vasconcelos [2015]	25
2.7	Other Important Features Table from Cruz and Vasconcelos [2015]	26
3.1	Reference Architecture Principles Information	47
3.2	How Reference Architecture satisfies the identified Enterprise Architecture Principles	47
4.1	Case Studies and their objectives	51
4.2	Fulfilment of the Alignment Heuristics by Architecture from ACM	57
4.3	Fulfilment of the Alignment Heuristics by solution from Reference Architecture	57
4.4	Evaluation Results on ACM CRM	58
4.5	Fulfilment of the Alignment Heuristics by Architecture from CS	65
4.6	Fulfilment of the Alignment Heuristics by solution from Reference Architecture	65
4.7	Evaluation Results on CS CRM	66
D.1	Salesforce Information Entities Mapped - Part 1	89
D.2	Salesforce Information Entities Mapped - Part 2	90
D.3	Salesforce Information Entities Mapped - Part 3	91
D.4	Salesforce Information Entities Mapped - Part 4	92
D.5	Salesforce Information Entities Mapped - Part 5	93
D.6	Salesforce Information Entities Mapped - Part 6	94
D.7	Microsoft Information Entities Mapped - Part 1	95
D.8	Microsoft Information Entities Mapped - Part 2	96
D.9	Microsoft Information Entities Mapped - Part 3	97
D.10	SugarCRM Information Entities Mapped - Part 1	98
D.11	SugarCRM Information Entities Mapped - Part 2	99
D.12	SugarCRM Information Entities Mapped - Part 3	100
D.13	Sage CRM Information Entities Mapped	101

D.14 Buttle Components Mapped 102

List of Figures

1.1	Action Research Methodology from Baskerville [1999]	3
2.1	ArchiMate Framework from Haren [2009]	6
2.2	Phases to transform Reference Architecture in a actual system from Muller and Hole [2007]	7
2.3	Inputs of Reference Architecture from Muller and Hole [2007]	8
2.4	Process for the generation of specific architectures from Bauer [2012]	9
2.5	EA design principle meta-model from Winter and Aier [2011]	10
2.6	ISO 9126 Standard model from ISO [2001]	10
2.7	Proposed Metrics mapped with qualities and architecture levels (1) from Vasconcelos [2007]	13
2.8	Proposed Metrics mapped with qualities and architecture levels (2) from Vasconcelos [2007]	14
2.9	CRUD Matrix Example	15
2.10	Oracle(Siebel) CRM architecture model from Fardoie and Monfared [2008]	18
2.11	HP CRM architecture model from Fardoie and Monfared [2008]	18
2.12	Microsoft Dynamics CRM architecture model from ¹	18
2.13	Sage CRM architecture model from ²	18
2.14	Sugar CRM architecture model from ³	18
2.15	Surado CRM architecture model from Fardoie and Monfared [2008]	18
2.16	Aspect CRM architecture model from Fardoie and Monfared [2008]	19
2.17	Mid-market CRM Suites from Buttle [2009]	20
2.18	MIT Sales Processes [MIT, 2001]	27
2.19	MIT Marketing Processes [MIT, 2001]	27
2.20	Microsoft Sales Processes [Microsoft, 2006]	28
2.21	Microsoft Marketing Processes [Microsoft, 2006]	28
2.22	Microsoft Customer Service Processes [Microsoft, 2006]	29
2.23	Salesforce Sales Data Model from ⁴	30
2.24	SugarCRM Account Data Model from ⁵	30
2.25	Microsoft Dynamics Marketing Data Model from ⁶	30
2.26	Sage CRM Data Model from ⁷	30
2.27	Siebel Account Data Model from ⁸	30
3.1	Methodology for defining the Reference Architecture - step 1	33

3.2	Methodology for defining the Reference Architecture - step 2	34
3.3	Methodology for defining the Reference Architecture - step 3	34
3.4	A Reference Architecture elaborates mission, vision and strategy to provide guidance to multiple organizations from Muller and Hole [2007]	35
3.5	CRM Reference Application Architecture [Cruz and Vasconcelos, 2015]	35
3.6	Sales Module [Cruz and Vasconcelos, 2015]	37
3.7	Marketing Module [Cruz and Vasconcelos, 2015]	37
3.8	Service Module [Cruz and Vasconcelos, 2015]	37
3.9	Reporting and Analytics Module [Cruz and Vasconcelos, 2015]	38
3.10	Mobile Module [Cruz and Vasconcelos, 2015]	38
3.11	Document Module [Cruz and Vasconcelos, 2015]	38
3.12	Integration Module [Cruz and Vasconcelos, 2015]	39
3.13	Security Module [Cruz and Vasconcelos, 2015]	39
3.14	Calendar Module [Cruz and Vasconcelos, 2015]	39
3.15	Workflow Module [Cruz and Vasconcelos, 2015]	39
3.16	Reference Information Architecture for the CRM domain	40
3.17	Reference Architecture CRUD Matrix	41
3.18	Reference Application Architecture proposed for the CRM domain	42
3.19	Account/Contact Module	42
3.20	Sales Module	43
3.21	Service Module	43
3.22	Marketing Module	43
3.23	Security Module	44
3.24	Scheduler Module	44
3.25	Portal System	44
3.26	Contact Center System	45
3.27	Document and Knowledge Management System.	45
3.28	Workflow Management System	45
3.29	Reporting and Analytics System	45
4.1	Evaluation Step 1 Workflow	50
4.2	Evaluation Step 2 Workflow	50
4.3	ACM CRM Information Entities	53
4.4	ACM CRM CRUD Matrix	53
4.5	ACM CRM Application Architecture	54
4.6	ACM CRM CRUD Matrix by Reference Architecture	55
4.7	ACM CRM Application Architecture by Reference Architecture	56
4.8	CS CRM Service Management Processes	59
4.9	CS CRM Cashier Management Processes	59

4.10 CS CRM Citizen Spaces Management Processes	59
4.11 CS Data Model	60
4.12 CS CRUD Matrix	61
4.13 CS Application Architecture	61
4.14 CS CRUD Matrix from Reference Architecture	63
4.15 CS Application Architecture from Reference Architecture	64
4.16 SIGA Pattern	67
4.17 Reference Application Architecture with SIGA Pattern	67
A.1 ACM CRM Customer Management Processes	80
A.2 ACM CRM Processes Management Processes	80
A.3 ACM CRM Task Management Processes	80
A.4 ACM CRM Follow Up Management Processes	80
A.5 ACM CRM Service Management Processes	80
A.6 ACM CRM Service Management Processes	80
A.7 ACM CRM Document Management Processes	81
A.8 ACM CRM Administration Management Processes	81
A.9 ACM CRM Report Management Processes	81
A.10 ACM CRM User Management Processes	81
B.1 Campaign Activity information entity clustering	82
B.2 Opportunity information entity clustering	82
B.3 Order information entity clustering	82
B.4 Person information entity clustering	82
B.5 User information entity clustering	82
B.6 Contract information entity division	82
C.1 ACM Services Realization in Current State	83
C.2 ACM Services Realization by architecture from Ref. Architecture	84
C.3 EdC Services Realization in Current State	85
C.4 EdC Services Realization by architecture from Ref. Architecture	86
D.1 CRM customer components from Buttle [2009]	88
D.2 CRM products components from Buttle [2009]	88
D.3 CRM marketing automation components from Buttle [2009]	88
D.4 CRM sales-force automation components from Buttle [2009]	88
D.5 CRM service automation components from Buttle [2009]	88
D.6 CRM partner relationship management components from Buttle [2009]	88

List of Acronyms

ACM High Commissioner for the Migration

AMA Agency for Administrative Modernization

CRM Customer Relationship Management

CRUD Create, Read, Update and Delete

CS Citizen Spaces

EA Enterprise Architecture

ICT Information and Communication Technology

IEEE Institute of Electrical and Electronics Engineers

ISO International Organization for Standardization

IT Information Technology

ITIL Information Technology Infrastructure Library

LCOISF Lack of Cohesion in «IS Block» Factor

NOISF Average Number of Operations in «IS Blocks»

RSF Response for a Service Factor

SIGA Integrated Service Management System

Chapter 1

Introduction

A great change happened in the business world in the recent years, the focus of businesses changed from product focus to customer focus. [Fardoie and Monfared, 2008] Nowadays, more and more companies adhere to Customer Relationship Management (CRM) solutions in order to gain more loyal customers. [Fardoie and Monfared, 2008] However to implement a true CRM system, proper architecture is required.

The companies, due the complexity of integrating the CRM with their business processes and IT, need to know and analyse their actual state and define the strategic direction they want to follow. [op't Land et al., 2009] Enterprise Architecture helps to solve these requirements, since they are part of its objective as stated by Mark Lankhorst: *"An enterprise architecture tries to describe and control an organisation's structure, processes, applications, systems and techniques in an integrated way."* [Lankhorst, 2005]

With this adherence to CRM solutions, a Reference Architecture for this domain is expected to provide a way to approach usual occurring problems by documenting good architectural design practices. [Cloutier et al., 2010] In this work we present a Reference Application Architecture for the CRM domain, which is applied in cases of the Portuguese Public Administration provided by the Agency for Administrative Modernization (AMA). In order to reach the Reference Architecture it is necessary to gather the industries best practices. We extracted the best practices from five CRM very known commercial solutions: SugarCRM, Microsoft Dynamics CRM, Sage CRM, Oracle Siebel and Salesforce.

1.1 Business Environment

In this section we explain the context where the work done it's applied. This thesis is done in cooperation with the AMA. The main purpose in this cooperation is to verify if the constructed Reference Application Architecture in this work can be adapted to be the Reference Architecture for the CRM domain for the Portuguese Public Administration.

To accomplish that goal, the Reference Architecture reached will be applied in several case studies. In this thesis we apply the Reference Architecture in two public organizations of the Portuguese Public

Administration, which are: the High Commissioner for the Migration (ACM) and the Citizen Spaces (CS) (managed by AMA). These two cases differ in their dimension and in the systems that they use. The application of the Reference Architecture in these two cases consists in verifying if the architecture proposed covers all the requirements of the cases. If some of these requirements aren't covered by the Reference Architecture, we do an analysis to verify if there is a pattern. If a pattern is found, that pattern can be added to the Reference Architecture in order to make it more adapted to the Portuguese Public Administration.

1.2 Thesis Problem

In this section we present the thesis problems that we propose to solve with this work. The main problems that we propose to solve in this thesis, consists in answering the following three questions:

- **Is it relevant to define a Reference Architecture for the Customer Relationship Management considering industry best practices extracted from commercial solutions?**
- **Is a Reference Architecture for Customer Relationship Management useful for defining specific Enterprise Architectures for the CRM domain?**
- **Is the Reference Architecture adequate to be the Reference Architecture for the CRM domain for the Public Portuguese Administration?**

The three main questions declared above, raise other important questions regarding how to reach the Reference Architecture, that is the core of this thesis:

- **What are the main features of the Customer Relationship Management systems?**
- **What are the main information entities of the Customer Relationship Management systems?**
- **Which patterns compose a Reference Architecture for Customer Relationship Management domain?**
- **What are the main Enterprise Architecture Principles in Customer Relationship Management architectures?**

After presenting the thesis problems that we propose to solve, we explain in the next section the research methodology followed for the development of the work.

1.3 Research Methodology

For the development of this work we needed to follow a research methodology. We followed the Action Research methodology. Baskerville [1999] This methodology is organized in five steps, illustrated in Figure 1.1:

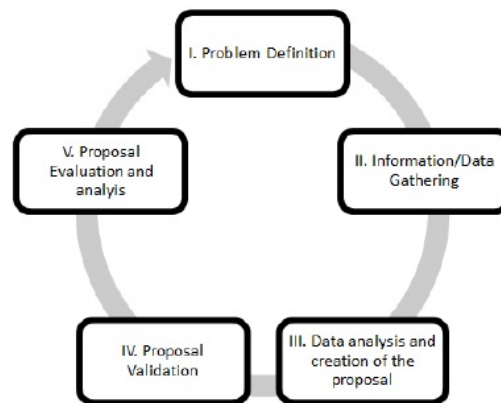


Figure 1.1: Action Research Methodology from Baskerville [1999]

We explain next the basis of each step:

- **First Step:** define the problem, to draw a scenario of what to be done;
- **Second Step:** gather and organize information about the problem, to create a theoretical and practical basis;
- **Third Step:** create the solution based on the previous information;
- **Fourth Step:** validate the solution with a case study;
- **Fifth Step:** evaluate and analyze the previous solution, to draw a conclusion about the proposal;
[Baskerville, 1999]

The document is structured in accordance to the Action Research Methodology, as can be seen in section 1.5.

1.4 Objectives

The objectives of our work are the following ones:

- Search related work about Enterprise Architecture, in particular the Reference Enterprise Architecture theme, how to reach a specific architecture from a Reference Architecture and how to evaluate information systems;
- Search related work about CRM domain, in particular what is a CRM system and what composes it;
- Identify the common features of the five CRM commercial solutions chosen;
- Identify the common information entities of the five CRM commercial solutions chosen;
- Propose a Reference Information Architecture for CRM domain;

- Propose a Reference Application Architecture for CRM domain;
- Model the patterns that compose the Reference Application Architecture;
- Identify the Enterprise Architecture Principles that the Reference Architecture solution proposed satisfies;
- Evaluate the benefits and pitfalls of the proposed Reference Architecture in real cases studies provided by AMA;
- Verify if the Reference Architecture proposed is adequate to the Portuguese Public Administration;
- Analyze and take conclusions on the results obtained in the case studies.

1.5 Document Structure

This document is structured into six chapters. In chapter 1, we do the introduction. The introduction includes the contextualization of the work, the definition of the thesis problem, the work goals and the scientific methodology followed. In chapter 2, we present the related work. In this chapter we present all the information that we gathered from our research, which was necessary to create the solution to the problem and for its evaluation. In chapter 3, we explain the steps of the methodology that we used to get to the architectural solution, the architecture solution and its specifications. In chapter 4, we present the evaluation methodology for the architecture proposed and the evaluation done in two case studies in accordance to that methodology. In chapter 5, we take the conclusions from the work done, the main contributions, the limitations and the future work. Chapter 6, is the final chapter, is where we present the bibliography that supports the content of the work.

Chapter 2

Related Work

In this chapter are presented the theoretical concepts required for the development of the solution and its evaluation. The chapter begins with an explanation of the Enterprise Architecture (EA) theme, and the most important concepts in that field for this work. In this section are also referred methodologies and metrics that are going to be used in evaluation chapter. Following is a section focused in the CRM domain, explaining what is a CRM, its specifications and the analysis that we done regarding CRM commercial solutions required for the development of the architecture solution.

2.1 Enterprise Architecture

The EA can be interpreted as an instrument to define the future direction of the enterprise, and also the mechanism that coordinates the actual transformation of the enterprise. EA handles the requirements that business performance needs, which are an integrated design of the enterprise and all that is related with it, e.g.: people and their competencies, organizational structures, business processes, IT, finances, products and services and its environment. [Greefhorst and Proper, 2011] So EA can be considered as a connector of the business strategy and the IT strategy, and also the essence of enterprise information planning. [Jin et al., 2010] We now present some EA definitions to help get a clearer view of this theme.

The Institute of Electrical and Electronics Engineers (IEEE) Standard ISO/IEC 42010 states that architecture is: *"The fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution."* [IEEE, 2000]

Mark Lankhorst defines EA objective by stating: *"Enterprise architecture tries to describe and control an organization's structure, processes, applications, systems and techniques in an integrated way."* [Lankhorst, 2005]

The Gartner Group defined EA concept as: *"Enterprise architecture (EA) is the process of translating business vision and strategy into effective enterprise change by creating, communicating, and improving the key principles and models that describe the enterprise's future state and enable its evolution."* [Lapkin, 2008]

Following the context we introduce important definitions of key concepts for EA and which we use during this work:

Design Pattern: *"A design pattern systematically names, motivates, and explains a general design that addresses a recurring design problem in a system. It describes the problem, the solution, when to apply the solution, and its consequences. It also gives implementation hints and examples. The solution is customized and implemented to solve the problem in a particular context". [Gamma et al., 1995]*

Models: *"a purposely abstracted and unambiguous conception of a domain". [Lankhorst, 2005]*

View: *"A representation of a whole system from the perspective of a related set of concerns". [IEEE, 2000]*

Viewpoint: *"A specification of the conventions for constructing and using a view. A pattern or template from which to develop individual views by establishing the purposes and audience for a view and the techniques for its creation and analysis". [IEEE, 2000]*

2.1.1 Enterprise Architecture Framework

An Enterprise Architecture Framework is as stated by Lankhorst [2005]: *"a conceptual structure of what an EA should contain and how to create it, i.e. models, principles, approaches, standards that guide the development of enterprise architectures".* For the representation of the EA, there are several numbers of different EA frameworks, which distinguish several architecture layers and views. [Winter and Fischer, 2010]

In this work, for the representation of our architecture and to represent the case studies, we use the ArchiMate notation, represented in Figure 2.1. We choose the ArchiMate, because this offers in a detailed and comprehensive way, the representation of the application domain and its relation with the business layer and the data domain.

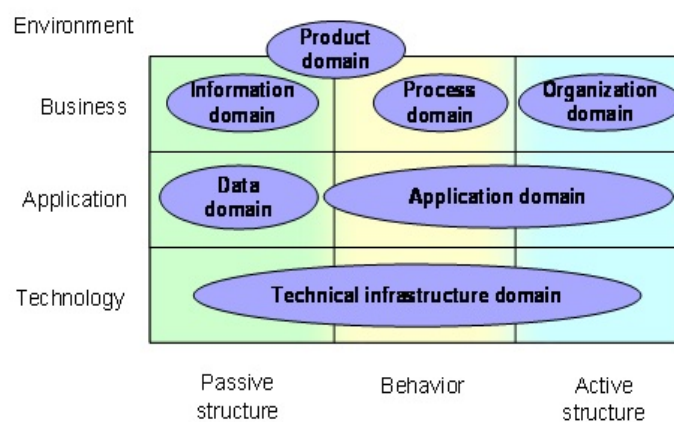


Figure 2.1: ArchiMate Framework from Haren [2009]

Following is the explanation of what consists each layer present in the Figure 2.1:

- **Business Layer:** "offers products and services to external customers, which are realized in the organization by business processes performed by business actors."
- **Application Layer:** "supports the business layer with application services which are realized by (software) applications."
- **Technology Layer:** "offers infrastructure services (e.g., processing, storage, and communication services) needed to run applications, realized by computer and communication hardware and system software." [Haren, 2012]

Along the work done in this thesis, we use the terminology Application Architecture because we use the ArchiMate notation in the representation of the views. But in the title we used Information Systems terminology because is a more understandable and global terminology.

2.1.2 Reference Enterprise Architecture

A Reference Enterprise Architecture is a way to approach usual occurring problems by documenting good architectural design practices. [Cloutier et al., 2010]

The Reference Enterprise Architecture primary objective is to direct and constrain the instantiations of solution architectures. To get a more clear view of the Reference Enterprise Architecture theme we have to answer three questions:

- **What is a Reference Architecture?** The answer to this question is already given above by the definition from Cloutier et al. [2010], but to complement that we present Figure 2.2, which illustrates the role of a Reference Architecture:

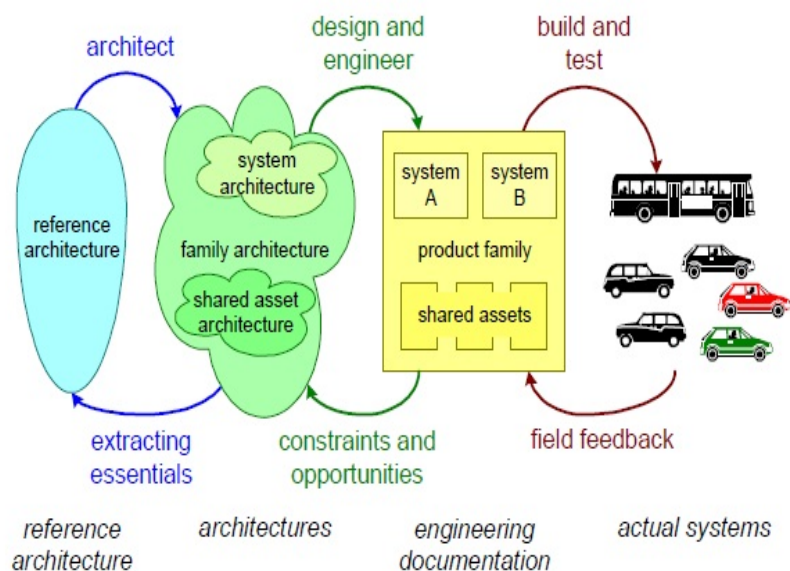


Figure 2.2: Phases to transform Reference Architecture in a actual system from Muller and Hole [2007]

- Why do we need Reference Architectures?** We need Reference Architectures because they improve effectiveness through: managing synergy, providing guidance (best practices, architectural principles), providing an architectural baseline and blueprint and by capturing and sharing architectural patterns. [Muller and Hole, 2007]
- How do you create a Reference Architecture?** A Reference Architecture captures previous experience, for instance by mining, or by generalizing existing architectures. To be of value for future architectures, a Reference Architecture is based on proven concepts. The validation of concepts in Reference Architectures is often derived from preceding architectures. The Figure 2.3, illustrates the inputs necessary to create a Reference Architecture. [Muller and Hole, 2007]

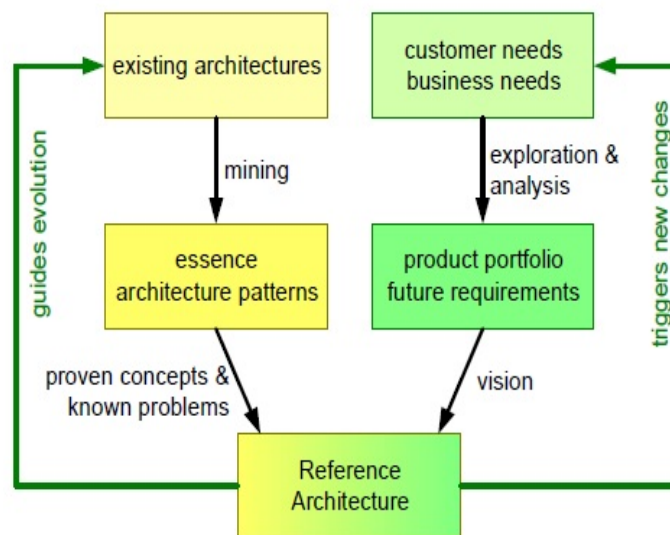


Figure 2.3: Inputs of Reference Architecture from Muller and Hole [2007]

2.1.3 Methodology for Specific Architectures Generation

In order to evaluate our Reference Architecture solution in various case studies, we need to reach a specific architecture for each case study based on the Reference Architecture. To reach a specific architecture we use the methodology from Bauer [2012], where to reach a specific architecture is needed the requirements from the properties of the desired system. Taking into account the requirements, the Reference Architecture is used to guide the architect by providing the best practices, architectural blueprints and patterns. Also in the definition of the specific architecture, at the same time that is used the guidance by the Reference Architecture, are also used engineering strategies in the designing of the system. In our case the engineering strategies are the ArchiMate and CRUD matrix. This methodology is exemplified in Figure 2.4.

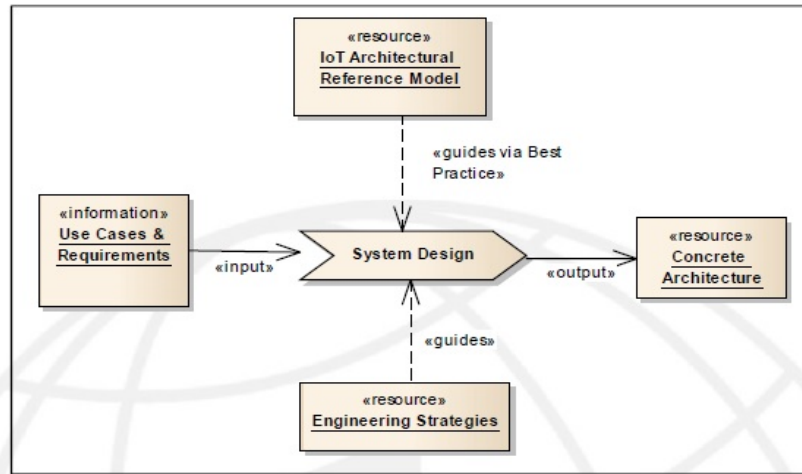


Figure 2.4: Process for the generation of specific architectures from Bauer [2012]

2.1.4 Enterprise Architecture Principles

TOGAF defines Enterprise Architecture Principles as: *“general rules and guidelines, intended to be enduring and seldom amended, that inform and support the way in which an organization sets about fulfilling its mission.”* [The Open Group, 2009]

A system fundamental organization is usually represented as his as-is state model or his to-be state model. This idea poses a problem, since the principles that guide the design and evolution of an architecture from the as-is state to the to-be state are often ignored, and most of the literature, doesn’t cover this aspect. The disregard of these principles is unanticipated, since these principles are considered the core of architecture design by e.g. Hoogervorst [2004] or Dietz [2007]. [Winter and Aier, 2011]

In this neglecting and lack of literature of principles, we could find Stelzer [2009] review of EA literature, that identified six publications that addressed EA Principles design. From those six publications, we present the three of them that contribute more for an EA design principle meta-model (illustrated in Figure 2.5). We start by the three definitions of EA Principles given by the authors of the three publication selected.

- Richardson et al. [1990]: *“Principles are an organizations basic philosophies that guide the development of the architecture. . . . Principles provide guidelines and rationales for the constant examination and re-evaluation of technology plans.”*
- Hoogervorst [2004]: *“collectively the design principles are identified as enterprise architecture”*
- Lindström [2006]: *“Architectural principles define the underlying general rules and guidelines for the use and deployment of all IT resources and assets across the enterprise . . . ”*

Following, we present the concepts that the authors give to the creation of EA design principle meta-model. Two important aspects were considered by Richardson et al. [1990]:

- a rational explanation on how the principle is meant to work and why the principle is defined;

- the implications that the principle brings to the enterprise. Implications that display how relevant system stakeholders are affected by the principle.

The aspects defined by Richardson et al. [1990] were re-used by Hoogervorst [2009, 2004] who added to them some key actions. A new aspect is also introduced, the principle statement, by Hoogervorst [2009, 2004] and Lindström [2006]. Lindström [2006] introduced another important concept of EA Principles design, the measurement, that can allow to evaluate the efficacy of principle as well as the fulfilment of the statement and the support of managing the EA Principles. [Winter and Aier, 2011] Figure 2.5 illustrates the core components of an EA design principle in a meta-model.

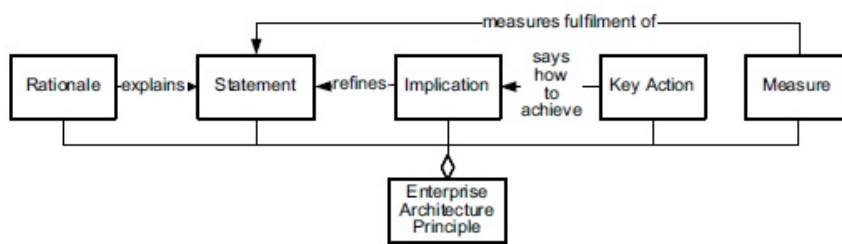


Figure 2.5: EA design principle meta-model from Winter and Aier [2011]

We use to make our EA Principles selection, the principles defined by Greefhorst and Proper [2011]. We analyse the list of principles and choose the ones that are satisfied by our Reference Architecture. Those principles are going to be considered the main EA Principles to follow when developing an Enterprise Architecture for the CRM domain. The principles chosen are presented in table 3.1 and table 3.2 in section 3.5.

2.1.5 Information Systems Quality Attributes

In the software engineering domain there is a standard for a set of Quality Attributes purposed by ISO [2001], illustrated in Figure 2.6. The objective of these Quality Attributes is to be a component that can be used to evaluate the suitability of certain architectures. The quality attributes are divided into groups and classified in a hierarchical structure. Two levels are identified in that hierarchy: the upper level, that represents quality attributes and the lower level, representing software quality criteria. [ISO, 2001]

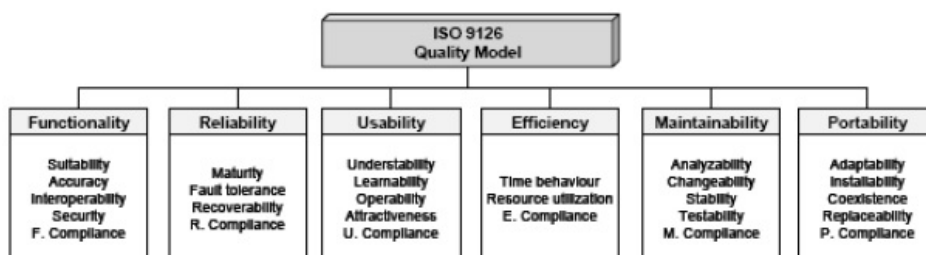


Figure 2.6: ISO 9126 Standard model from ISO [2001]

Following, we give a brief explanation on what each quality attribute means:

- **Functionality:** *"The capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions."*
 - **Reability:** *"The capability of the software product to maintain a specified level of performance when used under specified conditions."*
 - **Usability:** *"The capability of the software product to be understood, learned, used and attractive to the user, when used under specified conditions."*
 - **Efficiency:** *"The capability of the software product to provide appropriate performance, relative to the amount of resources used, under stated conditions."*
 - **Maintainability:** *"The capability of the software product to be modified."*
 - **Portability:** *"The capability of the software product to be transferred from one environment to another."*
- [ISO, 2001]

An extension to this model was proposed by Vasconcelos [2007], where is proposed the application of these Quality Attributes to the Information Systems domain. This model has a new element in relation to the original model: the alignment. The alignment is the capacity of the Information Systems work to contribute for an improvement of performance of the organization over time, in accordance to the requirements available in other architectural levels. There are four types of alignments between architectural layers:

- **Business / System Alignment:** capacity of the Information System Architecture to operate in accordance with the requirements requested by Business Architecture components, to improve the performance of the organization over time.
- **Information / Application Alignment:** capacity of the Application Architecture to operate in accordance with the requirements requested by Information Architecture components, to improve the performance of the organization over time.
- **Information / Technology Alignment:** capacity of the Technology Architecture to operate in accordance with the requirements requested by Information Architecture components, to improve the performance of the organization over time.
- **Application / Technology Alignment:** capacity of the Technology Architecture to operate in accordance with the requirements requested by Application Architecture components, to improve the performance of the organization over time. [Vasconcelos, 2007]

The alignment qualities that are measured in this work are the alignments between business architecture and system architecture and between information architecture and application architecture. The purpose of presenting these information systems quality attributes is because these are the aspects we want to measure when evaluating an information system.

2.1.6 Metrics and Heuristics for Information Systems Evaluation

In the proposed solution one part of the work is to demonstrate the results that our architecture has in real case studies. For that demonstration is necessary to evaluate the architecture proposed with some metrics to evaluate information systems. We decided to use two methods from investigations done in our university, that provided a way to measure and evaluate information systems. The first method of Pereira and Sousa [2003, 2005]; Vasconcelos et al. [2005], provides heuristics and metrics to quantify the level of alignment in three vectors: the alignment between Business Architecture and Information Architecture, the alignment between Business Architecture and Application Architecture and the alignment between Information Architecture and Application Architecture. These heuristics and metrics are presented below:

- **Alignment between Business Architecture and Information Architecture:**

H1.1 - All entities are created (C) by at least one process.

H1.2 - All processes create, update or delete (CUD) at least one entity.

H1.3 - All entities are read (R), at least by one process.

$$AlinAN_AI = \frac{H1.1 + H1.2 + H1.3}{\#HeurAlinAN_AI} \quad (2.1)$$

$$H1.x, \exists x, H1.x \mathbb{N} x \subset [1, 3] \wedge H2.x \subset [0, 1] \quad (2.2)$$

- **Alignment between Business Architecture and Information Systems Architecture:**

H2.1 - Each business process must be supported by at least one information system.

H2.2 - Each functionality of a information system must support at least one activity of the business process.

H2.3 - All the activities of a business process are rather supported by a single system or application.

$$AlinAN_ASI = \frac{H2.1 + H2.2 + H2.3}{\#HeurAlinAN_ASI} \quad (2.3)$$

$$H2.x, \exists x, H2.x \mathbb{N} x \subset [1, 3] \wedge H2.x \subset [0, 1] \quad (2.4)$$

- **Alignment between Information Architecture and Information Systems Architecture:**

H3.1 - Each entity is managed by a single system. Manage means create and identify.

H3.2 - Each attribute of an entity must not be updated by more than a system (different attributes of the same entity can be updated by different systems).

H3.3 - One system must access to the information through the system that manages it, but so that the computational independence can be preserved.

H3.4 - The systems must be independent computational.

H3.5 - The characteristics of the information should be in accordance with the characteristics of system that manages.

H3.6 - One transaction must involve only one system.

H3.7 - The data management must be automatic between systems.

$$AlinAI_{ASI} = \frac{H3.1 + H3.2 + H3.3 + H3.4 + H3.5 + H3.6 + H3.7}{\#HeurAlinAI_{ASI}} \quad (2.5)$$

$$H2.x, \exists x, H3.x \mathbb{N} x \subset [1, 7] \wedge H2.x \subset [0, 1] \quad (2.6)$$

Now that we have presented the metrics and heuristics from Pereira and Sousa [2003, 2005]; Vasconcelos et al. [2005], we explain next the second identified method to evaluate information systems from Vasconcelos et al. [2008]. The list of the metrics is illustrated in Figures 2.7 and 2.8:

Qualidade	Sub-qualidade	Métrica Proposta	Nível Arq.
Funcionalidade	Adequação	BSRPF - Factor de Serviço de Negócio Requerido e Disponibilizado	Negócio Aplicacional
	Interoperabilidade (semântica e técnica)	DIEF - Factor de Diferentes Implementações de Entidade Informacional	Informacional
		DTISSF - Factor de Tecnologias em que os «IS Service» são disponibilizados	Aplicacional Tecnológico
	Segurança	SCBITABF - Factor de componentes de segurança entre «IT Application Blocks»	Tecnológico
IASF - Factor de Segurança Informacional - Aplicacional		Informacional Aplicacional	
Fiabilidade	Tolerância a falhas	ITRF - Factor de Redundância Tecnológica	Tecnológico
Eficiência	Comportamento face a recursos	SITPLBF - Factor de <i>stapful</i> «IT Presentation Block» e «IT Logic Block»	Tecnológico
Facilidade de Manutenção	Facilidade de Análise	SCCF - Factor de Complexidade Ciclométrica dos Serviços	Negócio Aplicacional
	Facilidade de Alteração	LCOISF - Factor de Falta de Coesão em «IS Block»	Aplicacional Informacional
		NOISF - Factor do Número de operações em «IS Block»	Aplicacional
	Facilidade de Ensaio	RSEF - Factor de Resposta para um Serviço	Aplicacional
Portabilidade	Facilidade de adaptação	POSF - Factor de Possíveis Sistemas Operativos	Tecnológico

Figure 2.7: Proposed Metrics mapped with qualities and architecture levels (1) from Vasconcelos [2007]

Qualidade	Sub-qualidade	Métrica Proposta	Nível Arq.
Alinhamento	Alinhamento Negócio/Sistemas de Informação	CPSMF - Factor de Desalinhamento Processo Critico - Sistema	Negócio Aplicacional
	Alinhamento Informacional/Aplicacional	NAIEF - Factor do Número de aplicações por entidade informacional	Aplicacional Informacional
	Alinhamento Informacional/Tecnológico	LLIEITBDTMF - Factor de Desalinhamento de Tipo de Dados Entidade Informacional de Baixo Nível - Bloco Tecnológico	Informacional Tecnológico
	Alinhamento Aplicacional/Tecnológico	CSTMf - Factor de Desalinhamento Sistema Critico - Tecnologia	Aplicacional Tecnológico
Dimensão		NE - Número de Entidades	Informacional
		NA - Número de Aplicações	Aplicacional
		NITB - Número de Blocos Tecnológicos	Tecnológico

Figure 2.8: Proposed Metrics mapped with qualities and architecture levels (2) from Vasconcelos [2007]

From these list of metrics presented, we chose the following three metrics:

- **NOISF - Average Number of Operations in «IS Blocks»:** this metric is calculated by counting the number of operations in each «IS Block» and dividing the product of the number of «IS Blocks» and «IS Operations». The adaptability and changeability of Information Systems Architecture (ISA) increased with the value of this metric; [Vasconcelos et al., 2008]

$$NOISF = \frac{\# \ll ISBlock \gg}{\sum_{i=1}^{\# \ll ISBlock \gg} \# \ll ISoperation \gg_{\ll ISBlock \gg_i}} \quad (2.7)$$

- **RSF - Response for a Service Factor:** metric is calculated by the average by service of the IS components that are summoned to support the services. The facility of test increases with the value of the metric; [Vasconcelos et al., 2008]

$$RSF = \frac{\# \ll BusinessService \gg + \# \ll ISService \gg}{\sum_{i=1}^{\# \ll BusinessService \gg + \# \ll ISService \gg} \# \ll ISoperation \gg_i} \quad (2.8)$$

- **LCOISF - Lack of Cohesion in «IS Block» Factor:** is calculated by counting the number of sets of informational entities that are used by different functionalities in the same application. The ease of modification of an ISA grow with the value of this metric; [Vasconcelos et al., 2008]

$$LCOISF = 1 - \frac{\sum_{i=1}^{\# \ll ISBlock \gg} \# LCOISF}{\# \ll ISBlock \gg \times \# \ll ISoperation \gg \times \ll InformationEntity \gg} \quad (2.9)$$

We chose these metrics, because they allow us to evaluate our architecture in terms of the number of systems and modules used, as well as the relationship that the information systems have with the information and business processes.

2.1.7 CRUD Matrix

The CRUD Matrix goal is to present the actions of the businesses processes that act over the information entities. The type of actions are related to the name CRUD, which means "Create", "Read", "Update" and "Delete":

- **Create:** implies the creation of the id of the information entity;
- **Read:** means an access to the information entity from the business process;
- **Update:** implies a change of the state associated with the entity identifier;
- **Delete:** implies the invalidation of the entity identifier and after the delete the entity can't be nevermore manipulated; [Spewak and Hill, 1993]

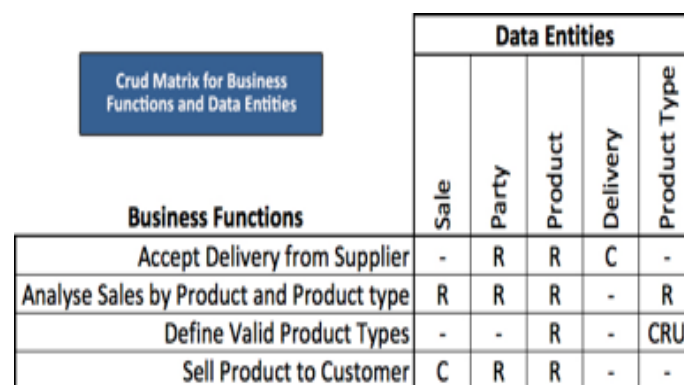
Through an analysis of the CRUD Matrix, we can divide and allocate the businesses processes to systems that support them. In the matrix are identified clusters where are processes grouped by functional area, and this is done by mapping the business processes with the information entities. There cases when the matrix is more complex, this means with a great number of lines and columns, which makes it more difficult to manage. For these cases there are some rules to follow for simplification and these rules result in systems definition:

- **Rule 1:** eliminate irrelevant columns and lines, combine lines and columns which are equal;
- **Rule 2:** aggregating processes by the entities created by these;
- **Rule 3:** aggregate processes that update entities with the processes that create them; [Spewak and Hill, 1993]

Then is time to identify the clusters resulted from the rules. The clusters can represent:

- **Domains:** if macro processes and macro entities are combined;
- **Solutions:** if processes and entities are combined;
- **Applications:** if activities and entities or attributes are combined; [Spewak and Hill, 1993]

The systems which result from the CRUD matrix must be the more simple it can be. In the Figure 2.9 is illustrated an example of a CRUD matrix.



Business Functions	Data Entities				
	Sale	Party	Product	Delivery	Product Type
Accept Delivery from Supplier	-	R	R	C	-
Analyse Sales by Product and Product type	R	R	R	-	R
Define Valid Product Types	-	-	R	-	CRU
Sell Product to Customer	C	R	R	-	-

Figure 2.9: CRUD Matrix Example

This section purpose was to demonstrate the importance of the CRUD matrix on the definition of an Information System Architecture.

2.1.8 Critical Analysis

In this chapter, we presented and explained all concepts related to Enterprise Architecture, needed for this work. We started by introducing the context of what is Enterprise Architecture, and then explained what a EA Framework is and which one we use in this work. Next we clarify what is a Reference Architecture, which is the core of our work. The next step was to explain a methodology for the generation of specific architectures, more specifically, for the definition of an architecture based on the Reference Architecture for the case study. It didn't make sense to apply directly the Reference Architecture in a specific case, because the Reference Architecture contains a lot of information that a specific case may not need. Also very related to Reference Architecture, we explained what are Enterprise Architecture Principles and Information Systems Quality Attributes, which are the qualities that we are going to evaluate. After explaining the Information Systems Quality Attributes, we presented metrics and heuristics that allow us to evaluate those quality attributes. In the end, we have detailed what was a CRUD matrix, because it is a very important method to define Application architectures and, which we used in the solution and evaluation chapters.

2.2 Customer Relationship Management

In recent years, companies have acquired Customer Relationship Management (CRM) technology to expand their markets clearly. The CRM technology brings with it the creation of marketing opportunities, the rise of customer value and customer satisfaction in order to achieve business excellence, with the main purpose of gaining loyal customers. [Fardoie and Monfared, 2008]

The main question was what type of information is needed to create a strong, reliable, and lasting relationship with the customers. The companies wanted to know who were they customers, what were they interests, how to contact them. For example, in large businesses it is impossible to know the customers in a close way, in contrast to neighbourhood small businesses. There are various ways to interact with the company, which turned difficult to integrate all the information. This is where CRM systems are useful. [Laudon and Laudon, 2012]

Kenneth C. Laudon and Jane P. Laudon defined CRM systems as a way to: *"capture and integrate customer data from all over the organization, consolidate the data, analyse the data, and then distribute the results to various systems and customer touch points across the enterprise."* [Laudon and Laudon, 2012]

The CRM software companies provide solutions for three major areas:

- **Sales force automation:** *"is the application of computerized technologies to support salespeople and sales management in the achievement of their work-related objectives."* [Buttle, 2009]

- **Marketing automation:** *"is the application of computerized technologies to support marketers and marketing management in the achievement of their work-related objectives."* [Buttle, 2009]
- **Customer Service:** *"provide information and tools to increase the efficiency of call centers, help desks, and customer support staff. They have capabilities for assigning and managing customer service requests."* [Laudon and Laudon, 2012]

The CRMs also have three major technologies components:

- **Collaborative technologies:** can be interpreted as the customer touch points. By other words, the collaborative technologies are the different channels that the customers use to interact, such as email, phone call, fax, website pages, and so on.
- **Operational technologies:** are all the processes and functions related to the three major areas: sales (account management, territory management and others), marketing (campaign management, email marketing and others) and customer support (case management, contact center and other).
- **Analytical technologies:** corresponds to the processing of the information of the sales, marketing and customer support and its transformation in information for reports and analytics. This can be used, for example, a diagnosis of customer relationship management. [Fardoie and Monfared, 2008]

In the Reference Enterprise Application Architecture proposed these three technologies are taken into account, but the Operational technologies are the most present and detailed in the proposed architecture. Now we provide an overview on what a CRM system is, by presenting the concept proposed by Buttle [2009], that states: *"CRM is the core business strategy that integrates internal processes and functions, and external networks, to create and deliver value to targeted customers at a profit. It is grounded on high quality customer related data and enabled by information technology."* In the following section, we present some CRM architecture models.

2.2.1 CRM Architecture Models

In this section, we present some models of CRM architectures, which give us an insight into what most of CRM models have. We consider the following CRM models: the Oracle (Siebel) model [Fardoie and Monfared, 2008], HP model [Janjicek, 2005], Microsoft Dynamics model ¹, Sage model ², SugarCRM model ³, Surado model [Fardoie and Monfared, 2008] and Aspect model [Fardoie and Monfared, 2008], illustrated in Figures 2.10-2.16.

¹<http://msdn.microsoft.com/en-us/library/bb928229.aspx>

²<http://www.kastechco.com/documents/CRM/TheBenefitsofCRMInternetArchitecture.pdf>

³<http://www.slideshare.net/Loadedtech/an-overview-of-sugarcrm>

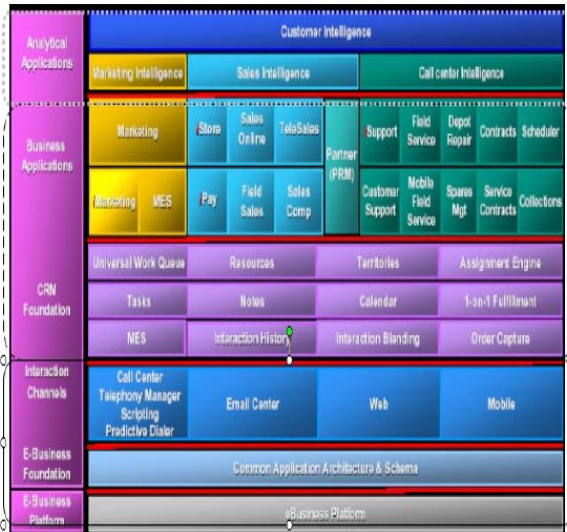


Figure 2.10: Oracle (Siebel) CRM architecture model from Fardoie and Monfared [2008]

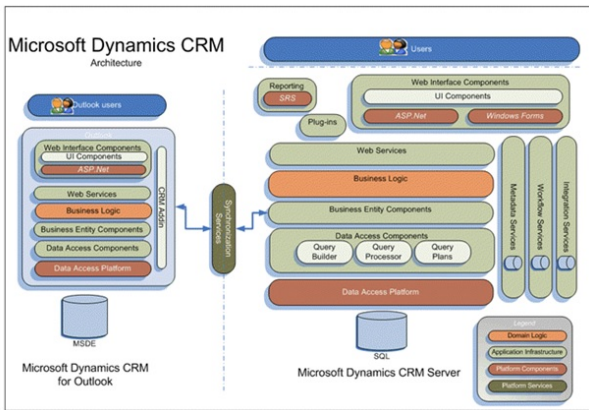


Figure 2.12: Microsoft Dynamics CRM architecture model from¹

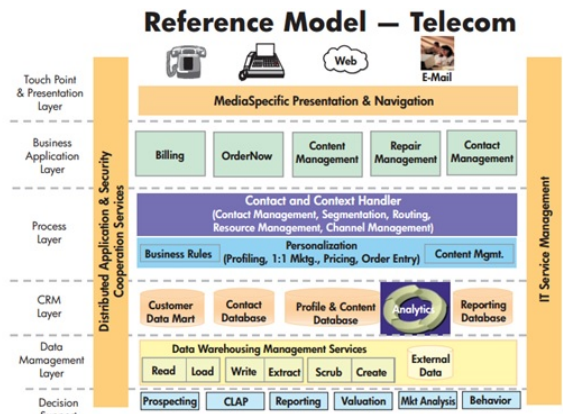


Figure 2.11: HP CRM architecture model from Fardoie and Monfared [2008]

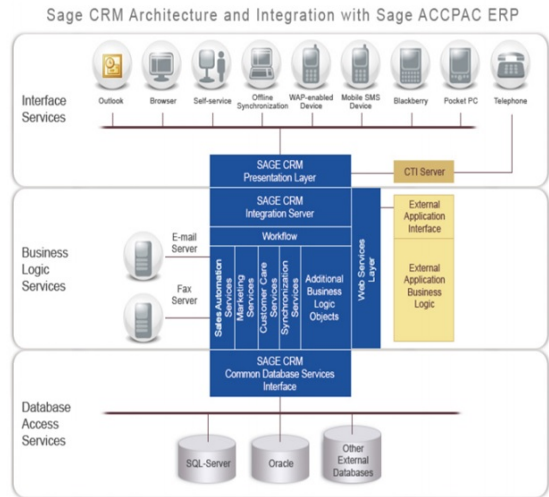


Figure 2.13: Sage CRM architecture model from²

SugarCRM offering



Figure 2.14: Sugar CRM architecture model from³

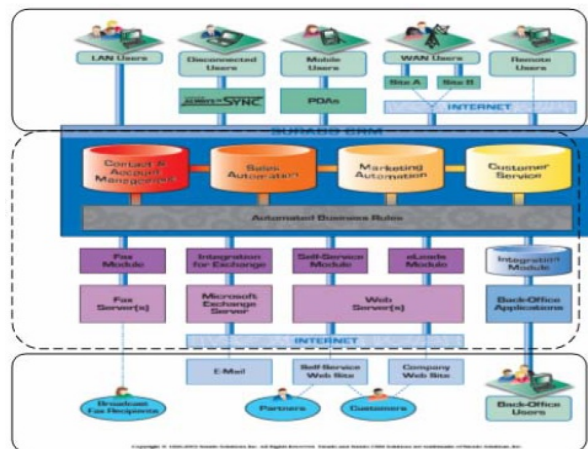


Figure 2.15: Surado CRM architecture model from Fardoie and Monfared [2008]



Figure 2.16: Aspect CRM architecture model from Fardoie and Monfared [2008]

In these models, we identify common components between them, that give us an idea on what composes a CRM:

- Interface/Channel: present in Oracle(Siebel) CRM, HP CRM, Microsoft CRM, Sage CRM, Sugar CRM, Surado CRM and Aspect CRM.
- Presentation layer: present in Oracle(Siebel) CRM, HP CRM, Microsoft CRM and Sage CRM.
- Web services: present in Microsoft CRM, Sage CRM, Surado CRM.
- Business logic: present in Oracle(Siebel) CRM, HP CRM, Microsoft CRM, Sage CRM, Surado CRM and Aspect CRM.
- Reporting and Analysis: present in Oracle(Siebel) CRM, HP CRM, Microsoft CRM and Sugar CRM.
- Data layer: present in HP CRM, Microsoft CRM and Sage CRM.
- Integration services: present in Microsoft CRM, Sage CRM and Surado CRM.
- Workflow automation: present in Microsoft CRM, Sage CRM, Sugar CRM and Surado CRM.

In the next section, we present the analysis and identification of the functionalities of some chosen CRM market solutions, to get the best practices of the industry.

2.2.2 CRM Features

To specify the Application functions of our CRM Reference Architecture, we extracted the features of five known CRM solutions and compared what features were common between them. We based the choice of these CRM solutions on: the list of mid-market CRM suites from Buttle [2009], illustrated in Figure 2.17, and the list of the Top CRM Software from Barrish [2014].

Enterprise CRM suites	Mid-market CRM suites
Amdocs CRM	Entellium
Chordiant Cx	GoldMine Corporate Edition
Onyx CRM	Maximizer Enterprise
Oracle's E-Business Suite CRM	Microsoft Dynamics CRM
Oracle's Siebel CRM	NetSuite
Oracle's PeopleSoft CRM	Oracle's Siebel CRM Professional Edition
Infor CRM E.piphary	Oracle's Siebel CRM On Demand
Pegasystems Customer Process Manager	Pivotal CRM
mySAP CRM	RightNow
	Sugar Enterprise
	SageCRM
	salesforce.com
	Soffront CRM Suite

Figure 2.17: Mid-market CRM Suites from Buttle [2009]

Merging the two references referred above we arrived to the five CRM solutions chosen: SugarCRM, Microsoft Dynamics, Sage, Oracle(Siebel) and Salesforce. The common features between the were the features chosen to be the functions of our Reference Application Architecture.

We introduce below the tables 2.1-2.7 with the features of each CRM, to verify what features were common between them and also the explanation of each feature. For the identification of the features that are present in the tables, we made a clustering of the features from the features present in the datasheets of the chosen CRM solutions. To give an example of the clustering made, in the SugarCRM datasheet there were two features: lead capture and lead scoring, routing and assignment, we clustered these two features in a feature by the name of lead management, because in the other CRM solutions existed a feature by the name of lead management, which covered these features as one, in other words, lead management is a more comprehensive feature. The features are presented in groups, in accordance with clustering of these features by the datasheets of the CRM solutions. Only in the case of the "Other important features", where we cluster the common features that weren't specific of an area, as were the features from sales, marketing for example.

Table 2.1: Sales Features Table from Cruz and Vasconcelos [2015]

Sales Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Account Management	X	X	X	X	X
Activity Management	X	X		X	X
Approvals	X				X
Competitor tracking	X	X	X	X	X
Contact Management	X	X	X	X	X
Contract Management	X	X		X	X
Sales Literature	X	X			
Lead Management	X	X	X	X	
Opportunity Management	X	X	X	X	X
Product Management	X	X	X	X	X
Quote Management	X	X	X	X	
Sales Forecasting	X	X	X	X	X
Territory Management	X	X	X	X	X
Order management	X	X	X	X	
Quota Management	X	X		X	X
Sales Pipeline	X	X		X	

Description of the Sales features[Microsoft, 2008d; Oracle, 2007c; Sage, 2012; Salesforce, 2012;

SugarCRM, 2014], illustrated in Table 1:

Account Management: offers sales representatives and managers a complete view of the customer relationship including contacts, contact history, completed transactions, current orders, shipments, enquiries, service history, opportunities and quotations.

Activity Management: keeps sales representatives and managers aware of all activities, whether complete or pending, related to an account, contact or opportunity, by establishing to-do lists, setting priorities, monitoring progress and programming alerts.

Approvals: manage success with flexible approvals processes for deal discounts, expenses, and more.

Competitor Tracking: maintain detailed information on competitors associated with opportunities.

Contacts Management: includes tools for building, sharing and updating contact lists, making appointments, time setting, and task, event and contact tracking.

Contract Management: enables representatives and managers to create, track, progress, accelerate, monitor and control contracts with customers.

Sales Literature: create, manage, and distribute a searchable library of sales and marketing materials, including brochures, white papers, and competitor information.

Lead Management: allows companies to create, assign and track sales leads. Leads either expire or convert into qualified opportunities.

Opportunity Management: enables representatives and managers to create an opportunity record in the database and monitor progress against a predefined selling methodology.

Product Catalog and Management: enables work with a full-featured product catalog that includes support for complex pricing levels, units of measure, discounts, and pricing options.

Quote Management: allows representatives and managers to quote for opportunities. This may be part of a broader order management capability.

Sales Forecasting: offer sales representatives and managers a number of qualitative and quantitative processes to help forecast sales revenues and close rates.

Territory Management: allows sales managers to create, adjust and balance sales territories, so that sales representatives have equivalent workloads and/or opportunities.

Order Management: allows representatives to convert quotations and estimates into correctly priced orders once a customer has agreed to buy.

Quota Management: design quota plans that motivate your sales team while supporting your company revenue goals.

Sales Pipeline: is the process of managing the entire sales cycle, from identifying prospects, estimating sales potential, managing leads, forecasting sales, initiating and maintaining customer relationships, right through to closure.[Buttle, 2009; Microsoft, 2008d; Oracle, 2007c; Salesforce, 2000; SugarCRM, 2004]

By analysing Table 1, we conclude that the most important sales features are: account management, activity management, competitor tracking, contact management, lead management, opportunity management, order management, product catalog and management, quote management, quota management, sales forecasting, sales analytics and territory management, because they are common to four or five CRM solutions and the result from that, is that we can infer that these are the most used sales features by the companies, given the fact that, all the manufacturers provide these features.

Table 2.2: Marketing Features Table from Cruz and Vasconcelos [2015]

Marketing Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Campaign Management	X	X	X	X	X
Campaign Execution	X	X	X	X	X
Email Marketing	X	X	X	X	X
Newsletter Management	X		X		
Marketing Campaigns	X	X	X		
List Management	X	X		X	X
Lead Management		X		X	X
Web To Lead Capture	X	X			X

Description of the Marketing features[Microsoft, 2008c; Oracle, 2007b; Sage, 2012; Salesforce, 2012; SugarCRM, 2014], illustrated in Table 2:

Campaign Management: define tasks, activities, and marketing materials for the entire campaign life cycle. Create budgets and define follow-up activities. Track responses to every campaign activity, monitor campaign results.

Campaign Execution: includes use of predefined system templates for future re-use in campaigns, or create new campaigns from scratch, schedule campaign activities to be performed immediately or at specific times in the future, and launch campaigns anywhere in the world with strong multi-lingual and multi-currency capabilities.

Email Marketing: Send email campaigns, merge customer data into personalized emails, insert conditional messaging based on recipient attributes, track delivery and response for each recipient automatically.

Newsletter Management: track responses to every campaign activity and convert email responses to leads or opportunities, qualify leads, and do much more.

Marketing Campaigns: marketing campaigns like Telemarketing, Internet marketing, Event-based marketing and Direct mail marketing, all except Email Marketing.

List Management: automatically create static or dynamic lists based on accounts, contacts, or leads.

Lead Management: track marketing campaign results across a variety of channels, from on-line ads to social media, to when leads come in, automated scoring and lead routing ensure that leads never fall through the cracks and always get to the right sales representative fast.

Web To Lead Capture: a way to allow visitors to your website or other on-line location to become leads.[Buttle, 2009; Microsoft, 2008c; Oracle, 2007b; Salesforce, 2000; SugarCRM, 2004]

By analysing Table 2, we conclude that the most important marketing features are: campaign management, campaign execution, list management, and email marketing, because they are common to four or five CRM solutions and the result from that, is that we can infer that these are most used marketing features by the companies, given the fact that, all the manufacturers provide these features. Note: the Lead Management is repeated in Sales features and Marketing features the reason for this, is due the fact that the Lead Management is the function that is responsible for a possible client approached in marketing passing to the sales phase. So is present in both areas.

Table 2.3: Customer Service Features Table from Cruz and Vasconcelos [2015]

Customer Service Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Case Escalation and Notification	X		X		X
Case Routing and Queuing	X	X		X	X
Contact Center	X	X	X	X	X
Case Management	X	X	X	X	X
Customer Self Service Portal	X	X	X	X	X
Email Management	X	X		X	X
Knowledge Base	X	X	X	X	X
Customer Information	X	X	X	X	X
Service Contracts				X	X

Description of the Service features[Microsoft, 2008b; Oracle, 2007d; Sage, 2012; Salesforce, 2012; SugarCRM, 2014], illustrated in Table 3:

Case Escalation and Notification: ensures that issues get escalated according to internally determined rules. Higher levels of authority typically have greater discretion to resolve issues.

Case Routing and Queuing: Queuing and routing applications allow issues to be routed to agents with particular expertise and positioned in that agents queue according to some criterion.

Contact Center: enables users to understand each customer as an individual, obtain all relevant customer information in a single view, and access that information when it matters from an incredibly fast, multi-channel agent desktop application. Teams can understand their accounts inside and out with personalized 360-degree business, on-line, and social customer intelligence.

Case Management: create, assign, and manage customer service requests across multiple channels, including phone, email, Web, in-person and emerging channels. Manage cases from initial contact through resolution and automatically associate incoming support inquiries with the appropriate case.

Customer Self Service Portal:allows companies to provide self-service capabilities to customers and prospects for key marketing, sales and support activities. Also allows non-technical users to create

and deploy web-to-lead forms, enables users to log and manage support cases on-line, allows customers to update account, contact, billing and shipping address and gives users the ability to manage subscriptions to company communications in an automated fashion.

Email Management: maintain accurate account, contact and service history with automated tracking and response for customer email messages.

Knowledge Base: resolve common support issues quickly using a searchable knowledge base. Ensure that published information is complete, correct, and properly tagged using built-in review processes. Build and maintain a solution database that makes it easy for people to find appropriate solutions quickly.

Customer Information: manage accounts, contacts, calls, products, territory, activity and contracts.

Service Contracts : Service contracts are agreements between you and your customers for a type of customer support. Service contracts can represent different kinds of customer support, such as warranties, subscriptions, or service level agreements (SLAs).[Buttle, 2009; Microsoft, 2008b; Oracle, 2007d; Salesforce, 2000; SugarCRM, 2004]

By analysing Table 3, we conclude that the most important service features are: case routing and queuing, contact center, case management, knowledge base, customer self-service portal and email management, because they are common to four or five CRM solutions and the result from that, is that we can infer that these are most used service features by the companies, given the fact that, all the manufacturers provide these features.

Table 2.4: Reporting and Analytics Features Table from Cruz and Vasconcelos [2015]

Reporting Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Custom reports	X	X	X	X	X
Dashboards	X	X	X	X	X
Sales Analytics	X	X		X	X
Marketing Analytics		X		X	X
Service Analytics		X		X	X

Description of the Reporting features[Microsoft, 2008d; Oracle, 2007a; Sage, 2012; Salesforce, 2012; SugarCRM, 2014], illustrated in Table 4:

Custom Reports: easily build customized reports with wizard-based tools that do not require technical resources from IT.

Dashboards: insightful and focused dashboards for executives and top constituents that adeptly highlight key marketing metrics, key sales metrics and for service analytics.

Sales Analytics: generate and use reports, make data relevant and track pipelines to transform information into Sales Intelligence.

Marketing Analytics: is the application of mathematical and statistical processes to marketing problems. Marketing analytics can be used to explore, describe and explain. Exploratory applications of marketing analytics provide insights into, and understanding about, issues and problems.

Service Analytics: provides in-depth knowledge into service request activity, resolution trends, service revenue, costs, and customer satisfaction. [Microsoft, 2008a; Oracle, 2007a; Salesforce, 2000; SugarCRM, 2004]

By analysing Table 4, we conclude that all the reporting features are important, because all of them are common to at least three CRM solutions and its not strange that this happened, because the base of CRM is to analyse information of the customers, which corresponds to reporting and analytics, so all the manufacturers provide these features.

Table 2.5: Integration Features Table from Cruz and Vasconcelos [2015]

Integration Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Email Integration	X	X	X	X	X
Social Networks	X		X	X	
Integrated third-party apps	X	X			X
Web service API - SOAP	X	X	X	X	X
Web service API - REST	X				
Computer Telephone Integration	X	X	X	X	X
Automatic Call Distributor	X	X	X	X	X
Microsoft Office Integration	X	X			X
Cloud Connectors	X	X			X

Description of the Integration features[Microsoft, 2008d; Oracle, 2007c; Sage, 2012; Salesforce, 2012; SugarCRM, 2014], illustrated in Table 5:

Integration Features: all most common components that are integrated with CRM.

By analysing Table 5, we conclude that the most important integration features are: email integration, web services api - SOAP integration, computer telephone integration and automatic call distributor , because they are common to four or five CRM solutions and the result from that, is that we can infer that these are most used integration features by the companies, given the fact that, all the manufacturers provide these features.

Table 2.6: Security Features Table from Cruz and Vasconcelos [2015]

Security Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Role Based Security	X	X	X		X
Advanced Password Management	X	X		X	X
Audit Trail		X		X	X
Field Level Security		X	X	X	X
User Based Security	X	X	X	X	X
Team Based Security	X	X			X

Description of the Security features[Gilchrist and Hariharan, 2009; Oracle, 2011; Sage, 2012; Salesforce, 2012; SugarCRM, 2014], illustrated in Table 6:

Role Based Security: privileges are assigned to defined categories of users (known as roles) rather than to individual users.

Advanced Password Management: allows administrators to set up system generated passwords versus manually created passwords for new users, failed login lockout attempts, and configure the email templates used to send password information to users.

Audit Trail: automatically records changes made to fields within the application, ensuring data security and integrity across the organization.

Field Level Security: restrict access to high business impact fields to specific users and teams.

User Based Security: authentication of users for security access.

Team Based Security: Control what your users can access. Lock down sensitive data to specific teams (groups). [Gilchrist and Hariharan, 2009; Oracle, 2011]

By analysing Table 6, we conclude that the most important security feature is: role based security, team based security, field level security, advanced password management and user based security feature, because they are common to four or five CRM solutions and the result from that, is that we can infer that these are most used security features by the companies, given the fact that, all the manufacturers provide these features.

Table 2.7: Other Important Features Table from Cruz and Vasconcelos [2015]

Other Important Features	Sugar CRM	Microsoft CRM	Sage CRM	Siebel CRM	Salesforce CRM
Workflow Processes Automation	X	X	X	X	X
Document Management	X	X	X		X
Mobile Access	X	X	X	X	X
Offline Access	X	X		X	X
Data Deduplication	X	X	X	X	X
Calendar Management	X	X	X	X	X

Description of the other important CRM features[Microsoft, 2008a; Oracle, 2007a; Sage, 2012; Salesforce, 2012; SugarCRM, 2014], illustrated in Table 7:

Workflow Processes Automation: design and run any business process with point and click simplicity using Workflow.

Document Management: provides the creation and management of files to share with users and contacts.

Mobile Access: access customer data instantly on a mobile device.

Offline Access: allows access a subset of records using the same browser-based interface as the online system but without an Internet connection.

Data De-duplication: detect and remove duplicate records.

Calendar Management: allows users to easily schedule, view, and manage their activities (e.g. calls, meetings, tasks) in one place. [Microsoft, 2008a; Oracle, 2007a; Salesforce, 2000; SugarCRM, 2004]

In Table 7, are presented features that for themselves are a specific module, and which we decided to group in a unique table. Note the Mobile access and Offline Access are part of the same module, that in the CRM solutions data-sheets goes by the name of Mobile CRM. All the features presented in the this table are important, because all are common to at least three CRM solutions.

2.2.3 CRM Business Processes

In this section we present common business processes in CRM domain. We decided to search for the most common business processes in the CRM domain, with the goal of using these business processes when making a CRUD matrix. We identified two references in this field of the business processes related to CRM domain: the MIT Process Handbook [MIT, 2001] and the Microsoft Dynamics Customer Model Departments Work [Microsoft, 2006]. The MIT Process Handbook had only processes related to the sales and marketing fields not covering the customer service. The business processes are represented in Figure 2.18-2.19:

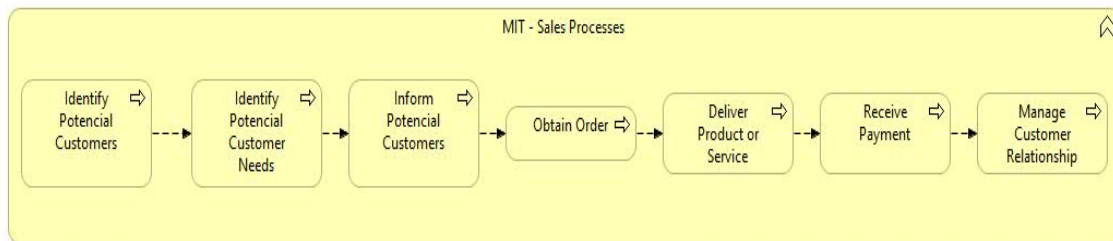


Figure 2.18: MIT Sales Processes [MIT, 2001]

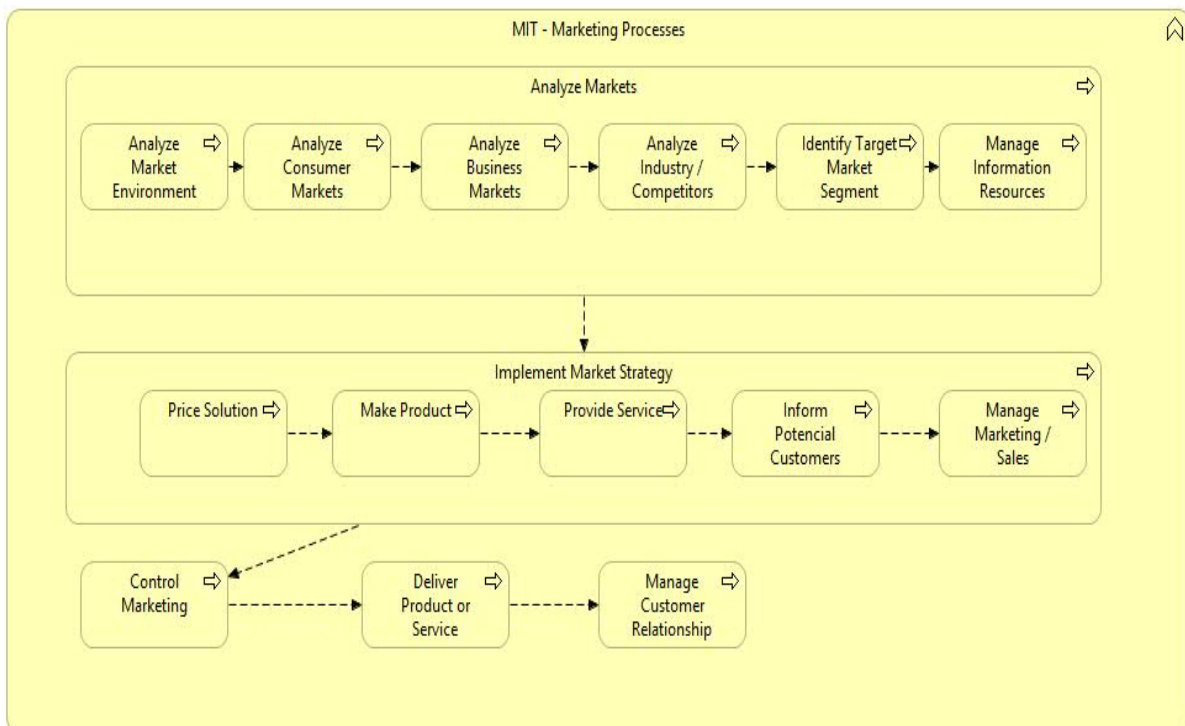


Figure 2.19: MIT Marketing Processes [MIT, 2001]

The business processes from the Microsoft Dynamics Customer Model Departments Work, are illustrated in Figures 2.20-2.22:

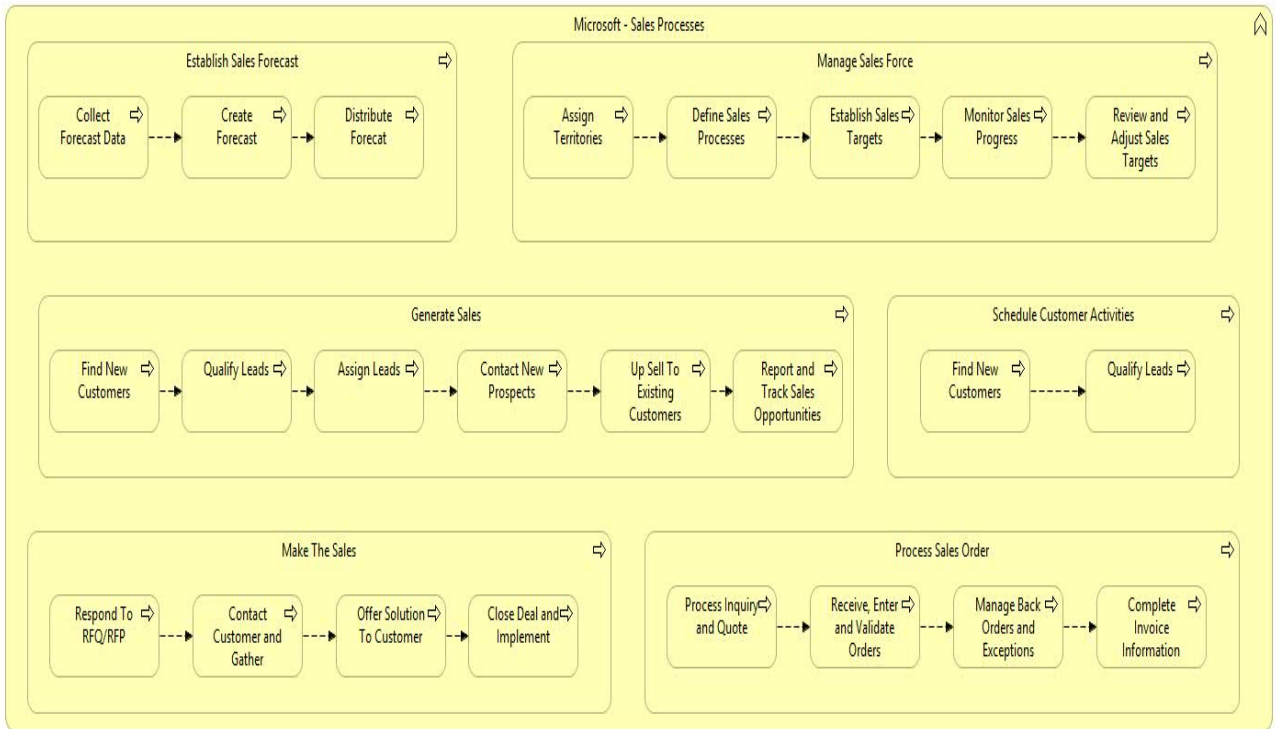


Figure 2.20: Microsoft Sales Processes [Microsoft, 2006]

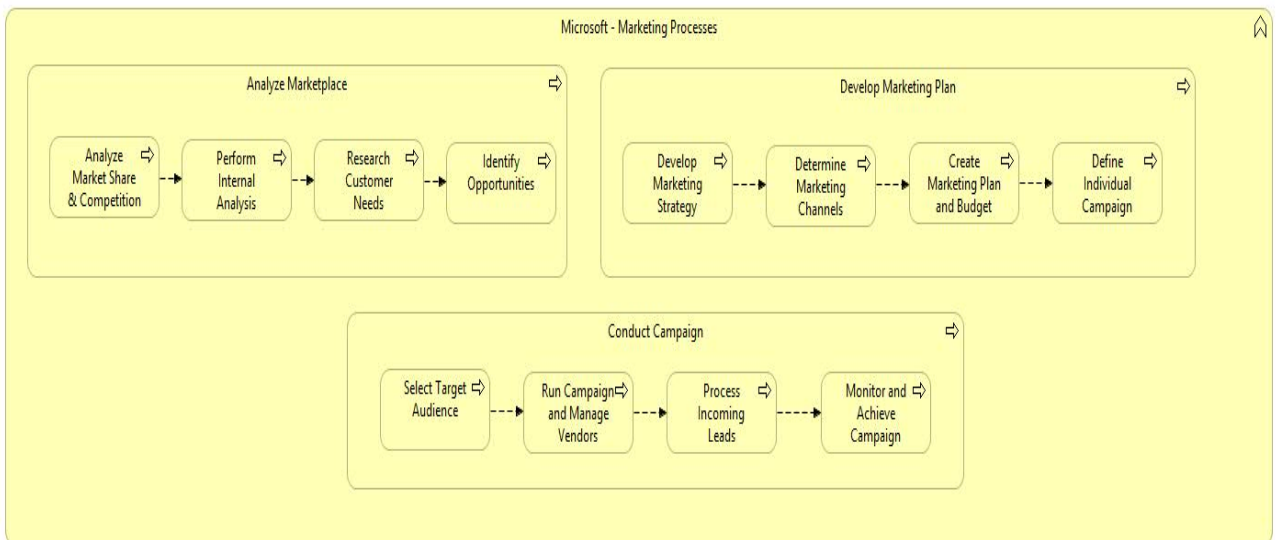


Figure 2.21: Microsoft Marketing Processes [Microsoft, 2006]

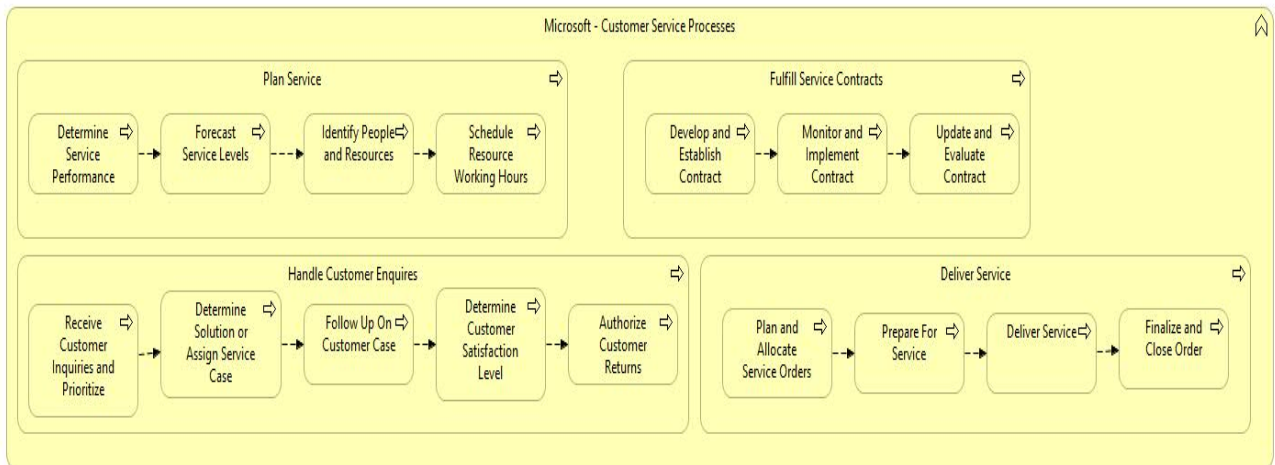


Figure 2.22: Microsoft Customer Service Processes [Microsoft, 2006]

Through an analysis of these figures, we can conclude that the business processes extracted from MIT are covered by Microsoft Dynamics business processes and that Microsoft Dynamics has in addition, business processes related to the customer service and for that reason we assert that the Microsoft Dynamics is more complete. But despite being the best reference between the two presented references, the Microsoft Dynamics reference wasn't complete enough, because it doesn't cover all the areas that we require to the Reference Architecture, and for this reason, when defining the architecture solution, we decided to use the functionalities extracted in the section 2.2.2 in the CRUD matrix.

2.2.4 CRM Data Models

After the identification of the features and business processes, we extract the common information entities from the data models of the five CRM solutions chosen: Salesforce⁴, SugarCRM⁵, Microsoft Dynamics⁶, Sage CRM⁷ and Siebel⁸. Following we present examples of the data models of each solution:

⁴http://www.salesforce.com/us/developer/docs/api/Content/data_model.htm

⁵http://dl.sugarforge.org/sugarcrm/SugarCE6.0GA/SugarCE6.0.4/Sugar6.0.0_CE_Schema_Diagrams.pdf

⁶<https://community.dynamics.com/>

⁷<http://creately.com/diagram/example/hhbgyd91/New+Sage+CRM+Data+Model>

⁸<http://pt.slideshare.net/PhilipJung/siebel-data-model-reference-82>

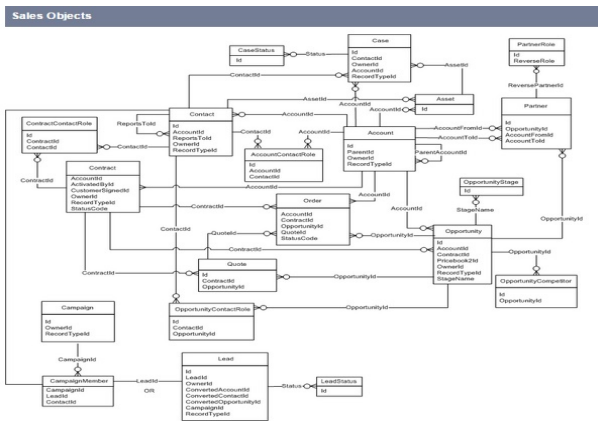


Figure 2.23: Salesforce Sales Data Model from 4

Accounts

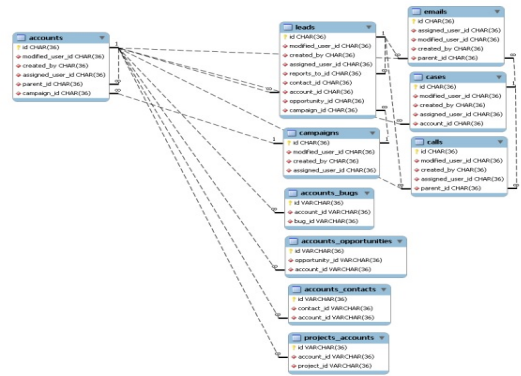


Figure 2.24: SugarCRM Account Data Model from 5

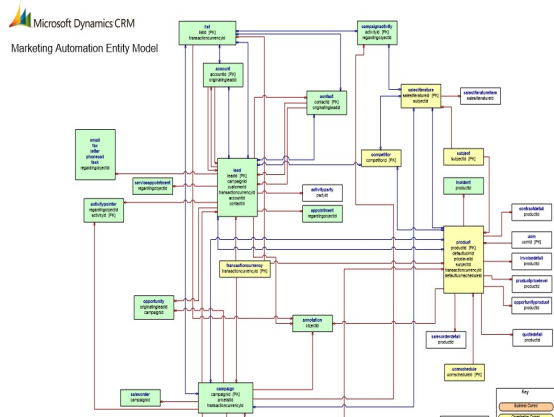


Figure 2.25: Microsoft Dynamics Marketing Data Model from 6

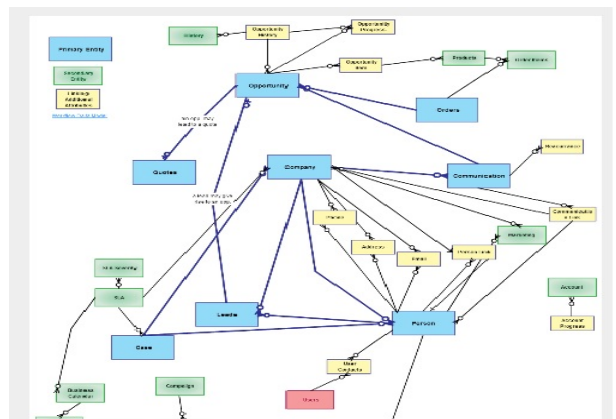


Figure 2.26: Sage CRM Data Model from 7

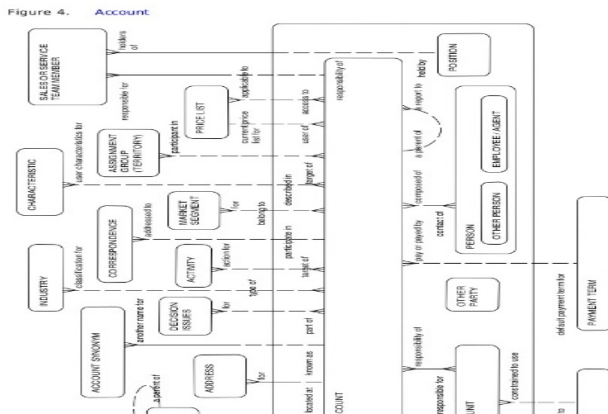


Figure 2.27: Siebel Account Data Model from 8

For the identification, we did a mapping between the identified information entities of each CRM solution, to reach the common information entities⁴⁵⁷⁸. This is done in tables, like in the section for the CRM features. The tables are present in Appendix D. Note that we begin by comparing a commercial solution with the other four solutions, and in the following table that commercial solution that has already been compared ceases to appear. This is done until all the solutions have been compared. Other important aspect in our work is that, to reach the common information entities we also use the [Buttle,

2009] reference. The data models from the five solutions were all different from each other, which would result in very confusing Reference Information Architecture, if we tried to see all the connections between the information entities in each data model and tried to merge all of them in one data model. So we decided to use the [Buttle, 2009], which has an example of CRM components and the connections between them, illustrated in Figures D.1-D.6 in Appendix D, to help in the creation of the Reference Information Architecture in chapter 3.

The tables D.1-D.14 in Appendix D present the mapping of information entities between the solutions, and also the Buttle [2009] components mapped with the five CRM solutions entities. [Cruz, 2015]

Through an analysis of these tables[Cruz, 2015] we reach the common information entities:

- Customer, Account, Organization, Person, Partner, Contact, Sales Activity, Competitor, Competitor Product, Contract, Lead, Prospect, Opportunity, Opportunity Item, Quote, Forecast, Quota, Territory, Customer Product, Order, Invoice, Product, Price, Product Catalog, Campaign Activity, Campaign, Campaign Wave, Marketing Funds, Marketing Segment, Marketing Budget, Marketing Plan, Campaign Response, Marketing List, Service Activity, Case, Case Solution, Email, Call, Communication, Portal, User, Group, Sales Team, Service Team, Scheduler, Calendar, Report, Dashboard, Workflow, Document, Knowledge Article, Person Address, Order Item, User Role, User Login, User Contact.⁴⁵⁷⁸

In order to simplify some of the common information entities identified we clustered and divided the ones we considered too detailed and could be simplified:

- Campaign Activity: aggregates Campaign Activity and Campaign Wave, illustrated in Figure B.1 (Appendix B);
- Opportunity: aggregates Opportunity and Opportunity Item, illustrated in Figure B.2 (Appendix B);
- Order: aggregates Order and Order Item, illustrated in Figure B.3 (Appendix B);
- Person: aggregates Person and Person Address, illustrated in Figure B.4 (Appendix B);
- User: aggregates User, User Role, User Login and User Contact, illustrated in Figure B.5 (Appendix B);
- Contract: is a specific case, that we divided in two specialized information entities specifically for the CRUD matrix, the Sales Contract and the Service Contract. We did this separation because in ? components we identified a difference between the contracts in service domain, which are more regarding agreements and warranties, and the contracts in sales domain. Is illustrated in Figure B.6 (Appendix B);

The information entities identified in this section allowed us to define a Reference Information Architecture in the Architecture Solution chapter.

2.2.5 Critical Analysis

In this section we describe the domain in which we will hold the reference architecture, the CRM domain. We explained what a CRM is and its goals, and next presented examples of CRM architecture models. Following we detailed the work done in the analysis of the five CRM commercial solutions: SugarCRM, Microsoft Dynamics CRM, Sage CRM, Oracle Siebel and Salesforce. We extracted from these five CRM solutions the best practices in the industry, which in this case are the common features and the common information entities. We also identified two references related to the common business processes in CRM domain. The information gathered in this chapter allowed us to define an architectural solution in the next chapter.

Chapter 3

Architecture Solution

This chapter starts by the presentation of the steps of the methodology defined for the development of the CRM Reference Architecture. This methodology fits in the third step of the Action Research Methodology followed for the whole work. After the presentation of the methodology, each step is explained and in the last step is given the final Reference Architecture solution reached. In the end of the chapter is exposed an identification of the Enterprise Architecture Principles present in the Reference Architecture solution.

3.1 Methodology followed for defining the Reference Architecture

We use a methodology to reach the Reference Application Architecture for the CRM domain based in Figure 2.3 from section 2.1.2, that illustrates the inputs of a Reference Architecture. The Figure 2.3 demonstrates that the creation of a Reference Architecture is a work of continuous improvement, a cyclical work. In our research we have three steps to reach the final Reference Architecture. The first step corresponds to the vision input in the referred figure, and the second and third steps corresponds to the mining into essence architecture patterns, that result in an input of proven concepts and known problems to the Reference Architecture. The steps are the following:

- **Step 1:** definition of the mission, vision and strategy of the Reference Architecture, because the Reference Architecture must facilitate an understanding of the current architectures and the vision of the future of the architecture. [Cloutier et al., 2010]

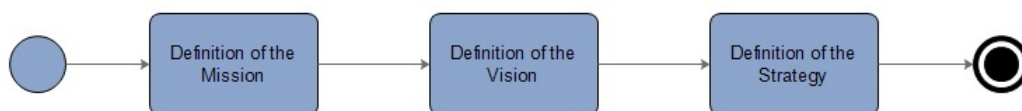


Figure 3.1: Methodology for defining the Reference Architecture - step 1

- **Step 2:** definition of a Reference Application Architecture based on the analysis and identification of the CRM solutions features from section 2.2.2, where we reached the most common modules and

functionalities of the CRM commercial solutions.

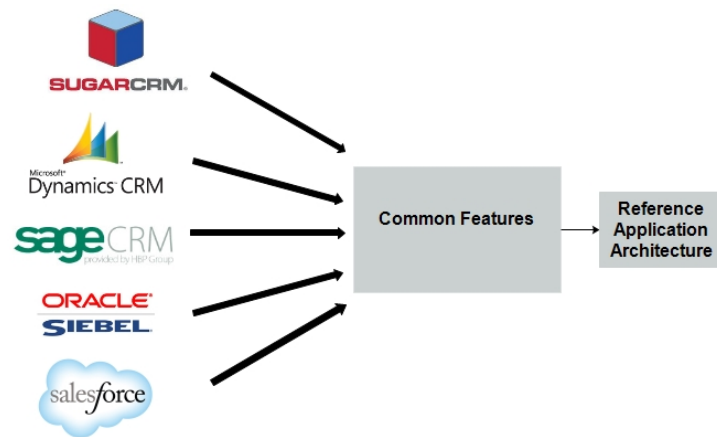


Figure 3.2: Methodology for defining the Reference Architecture - step 2

- **Step 3:** definition of a Reference Application Architecture based on a CRUD matrix, where are mapped the processes functions identified in section 2.2.2 and the most common information entities identified in section 2.2.4. With the common information entities we also define a Reference Information Architecture.

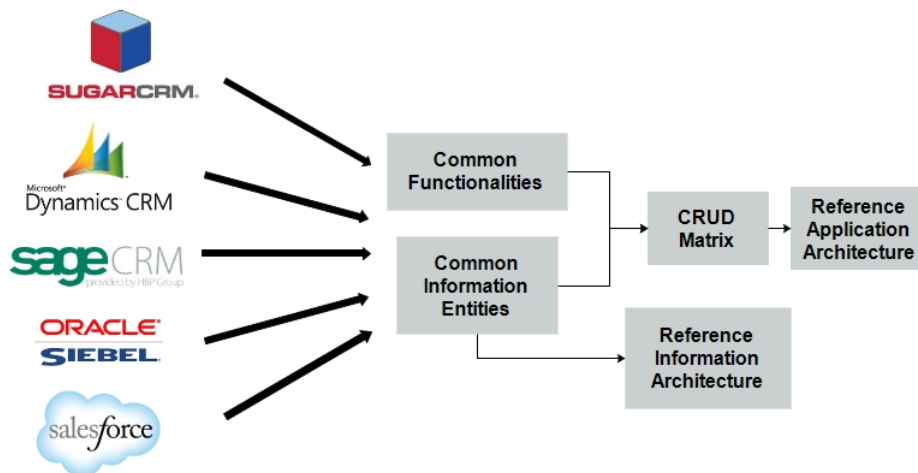


Figure 3.3: Methodology for defining the Reference Architecture - step 3

3.2 Reference Architecture - Vision, Mission and Strategy (Step 1)

As referred in Cloutier et al. [2010], the first principle of a Reference Architecture is: *“Reference Architecture is an elaboration of company (or consortium) mission, vision, and strategy. Such Reference Architecture facilitates a shared understanding across multiple products, organizations, and disciplines about the current architecture and the vision on the future direction”*. This is the reason for the first step of creating our Reference Architecture, being the definition of the vision, mission and strategy:

- **Mission:** provide guidance, knowledge, an architectural blueprint and architectural improvement in CRM domain;

- **Vision:** be the most complete Reference Architecture for the CRM domain;
- **Strategy:** extract best practices regarding the CRM domain, define the Reference Architecture based on those best practices and evaluate the Reference Architecture in case studies, in order to improve it;

The Figure 3.4 illustrates this relation between Reference Architecture and vision, mission and strategy concepts.

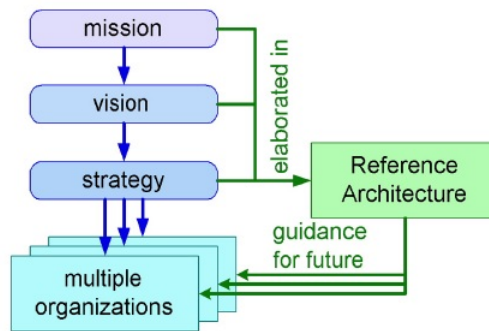


Figure 3.4: A Reference Architecture elaborates mission, vision and strategy to provide guidance to multiple organizations from Muller and Hole [2007]

3.3 Reference Architecture (Step 2)

In this section, we present a first view of the Reference Application Architecture based on the features of the CRM commercial solutions. The view is illustrated in Figure 3.5.

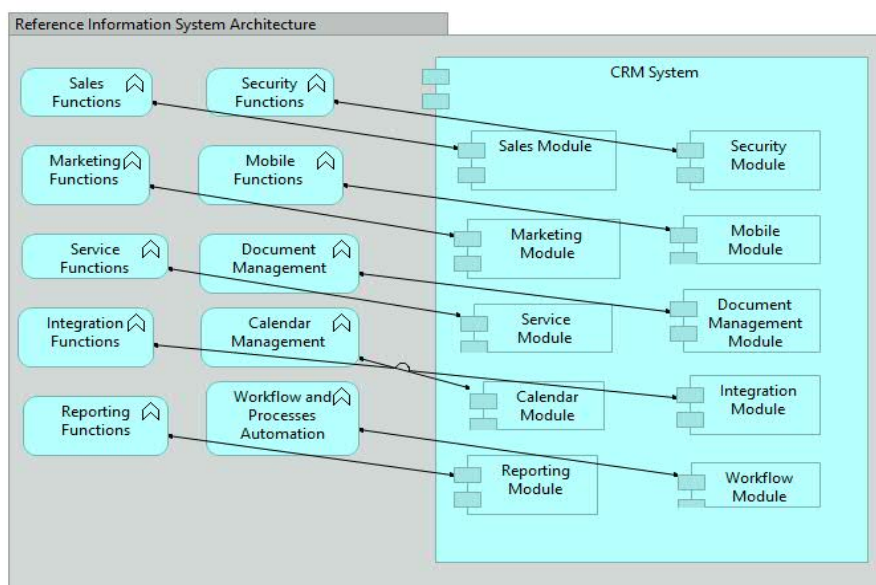


Figure 3.5: CRM Reference Application Architecture [Cruz and Vasconcelos, 2015]

The goal of this view is to present the Application layer with the reached modules and their functionalities. We arrived to this model through the features, which we identified and are presented in the tables

2.1-2.7 of section 2.2.2. For the selection of the features from those tables, we chose the features that are common to, at least, three of the five CRM solutions. The common features are:

- **Sales features:** account management, activity management, competitor tracking, contact management, contract management, lead management, opportunity management, product catalog and management, quote management, territory management, quota management, order management, sales pipeline and sales forecasting.
- **Marketing features:** campaign management, campaign execution, email marketing, marketing campaigns, list management, web lead to capture and lead management.
- **Service features:** case escalation and notification, case routing and queuing, contact center, case management, customer self service portal, email management, knowledge base, customer view and service contracts.
- **Report and Analytics features:** custom reports, dashboards, sales analytics, marketing analytics and service analytics.
- **Integration features:** social networks, email integration, web-services api - soap integration, microsoft office integration, automatic call distributor, computer telephone integration, cloud connectors and integrated third-party apps.
- **Security features:** role based security, advanced password management, control data access, user based security, team based security and audit trail.
- **Other important CRM features:** workflow and processes automation, document management, mobile and offline access and calendar management.

These features were grouped into 10 modules, in accordance with the areas that the features were presented in the CRM solutions datasheets:

- **Sales module** composed by the common sales features, Figure 3.6;

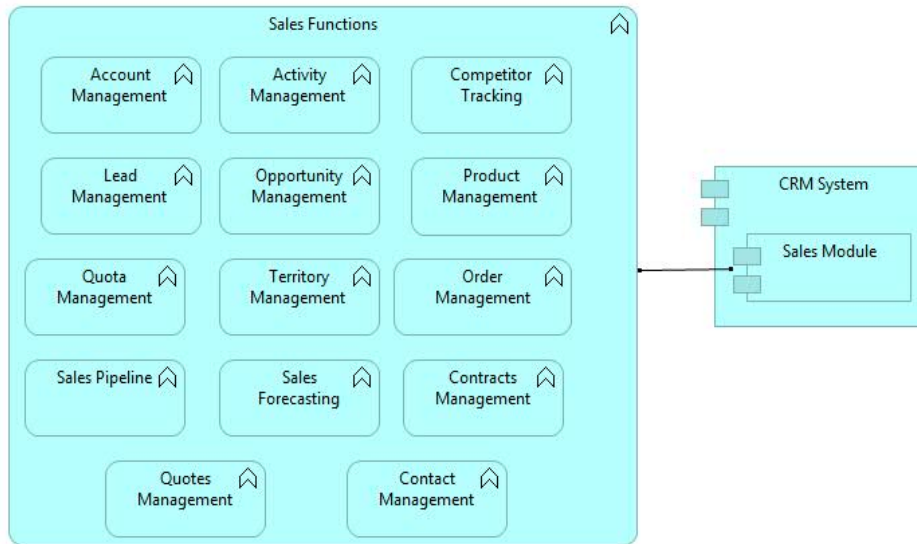


Figure 3.6: Sales Module [Cruz and Vasconcelos, 2015]

- **Marketing module** composed by the marketing features, Figure 3.7;

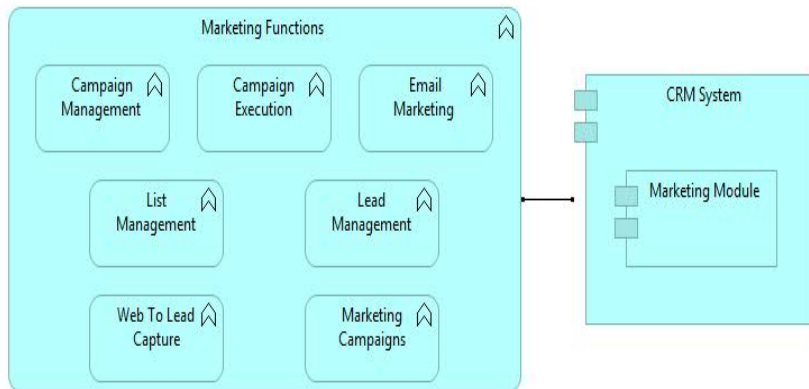


Figure 3.7: Marketing Module [Cruz and Vasconcelos, 2015]

- **Service module** composed by the common service features, Figure 3.8;

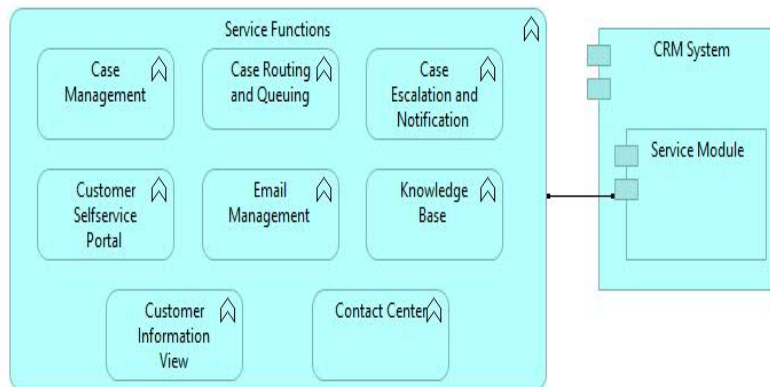


Figure 3.8: Service Module [Cruz and Vasconcelos, 2015]

- **Reporting module** composed by the common reporting features, Figure 3.9;

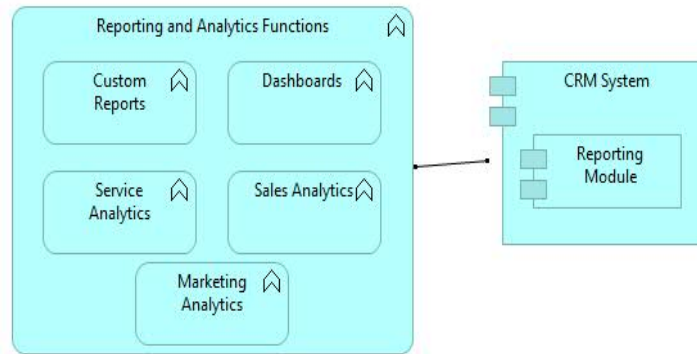


Figure 3.9: Reporting and Analytics Module [Cruz and Vasconcelos, 2015]

- **Mobile module** composed by mobile and offline features, Figure 3.10;

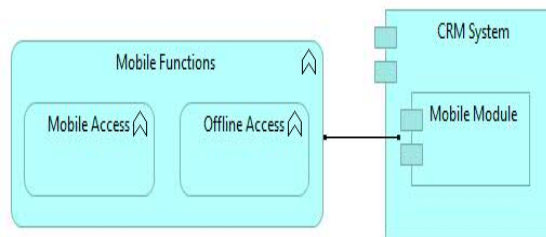


Figure 3.10: Mobile Module [Cruz and Vasconcelos, 2015]

- **Document module** composed by document management feature, Figure 3.11;

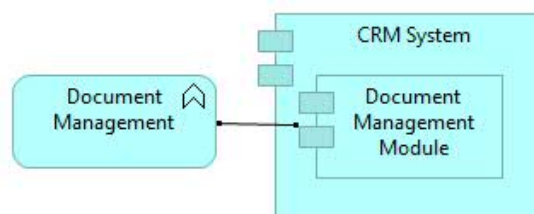


Figure 3.11: Document Module [Cruz and Vasconcelos, 2015]

- **Integration module** composed by the common integration features, Figure 3.12;

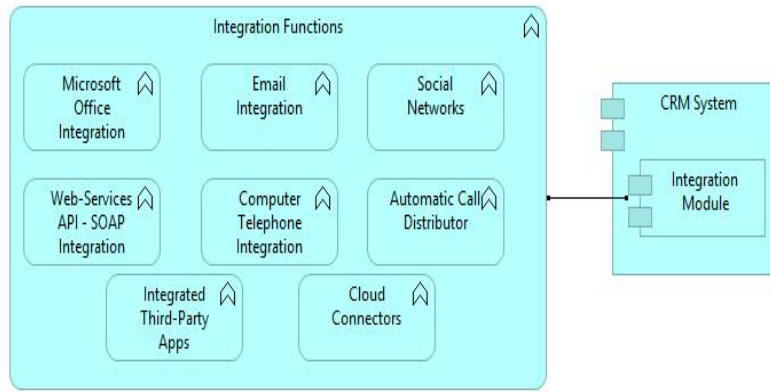


Figure 3.12: Integration Module [Cruz and Vasconcelos, 2015]

- **Security module** composed by the common security features, Figure 3.13;

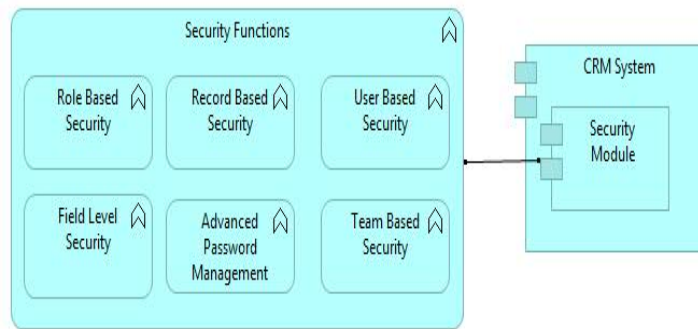


Figure 3.13: Security Module [Cruz and Vasconcelos, 2015]

- **Calendar module** composed by the calendar management feature, Figure 3.14;

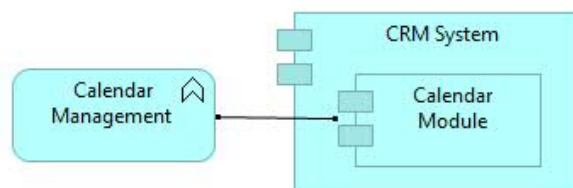


Figure 3.14: Calendar Module [Cruz and Vasconcelos, 2015]

- **Workflow module** composed by workflow process automation feature, Figure 3.15;

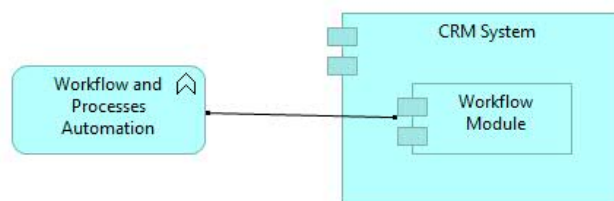


Figure 3.15: Workflow Module [Cruz and Vasconcelos, 2015]

3.4 Reference Architecture Complete (Step 3)

In this section we present the final Reference Architecture based on the third step of our methodology to reach a Reference Architecture. In this step we use the identification of the functionalities of the CRM solutions, done in the previous step, as the processes activities on the CRUD Matrix. It is important to refer that we choose not to use the business processes from Microsoft or MIT identified in section 2.2.3, because they don't cover all the CRM areas like administration, reporting, document management and others. The functionalities that we identified and, which we use in the CRUD Matrix, cover all the CRM areas required. We also use in this step of the methodology the most common Information Entities that we identify in section 2.2.4. In the Figure 3.16 is illustrated a view of a proposed Reference Information Architecture that we defined using those information entities.

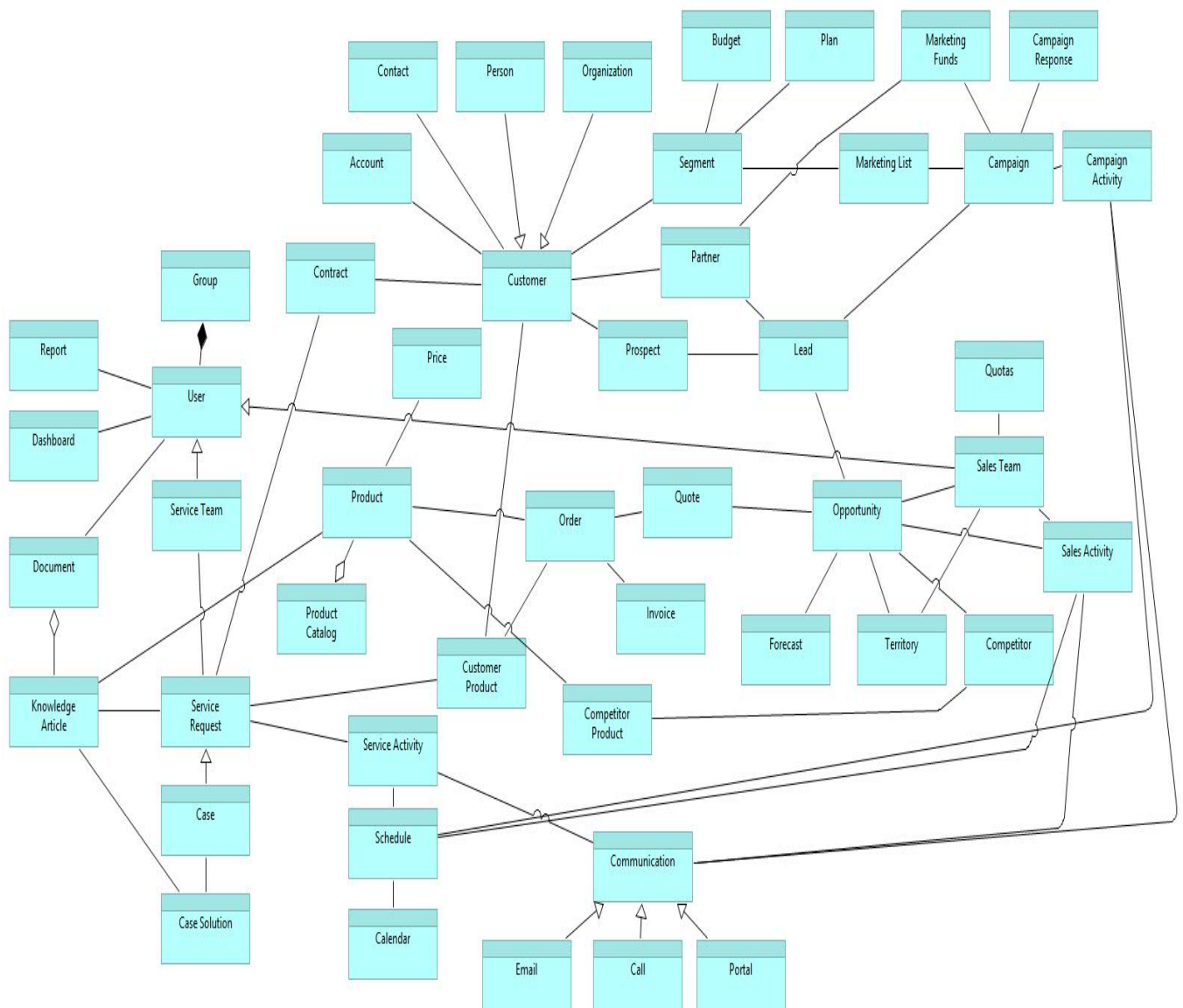


Figure 3.16: Reference Information Architecture for the CRM domain

With the processes functions and information entities, we reach the Reference Application Archi-

By analyzing the CRUD matrix, we model the Reference Application Architecture view in Figure 3.18:

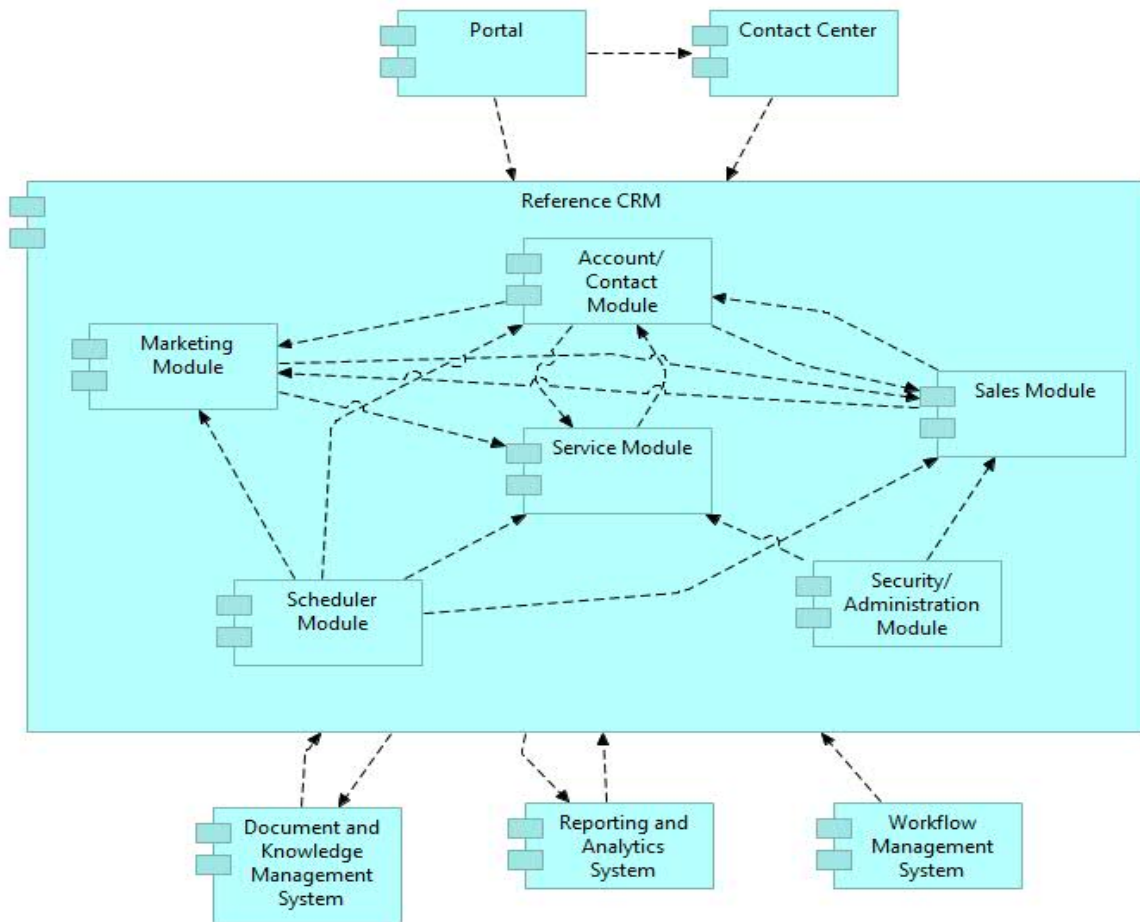


Figure 3.18: Reference Application Architecture proposed for the CRM domain

The Reference Application Architecture is composed by 6 modules in the CRM system and five systems that interact with the CRM system. These are the architectural patterns of our Reference Architecture:

- **Account/Contact module** satisfies account management, contact management and customer information (360° customer view) functions, Figure 3.19;

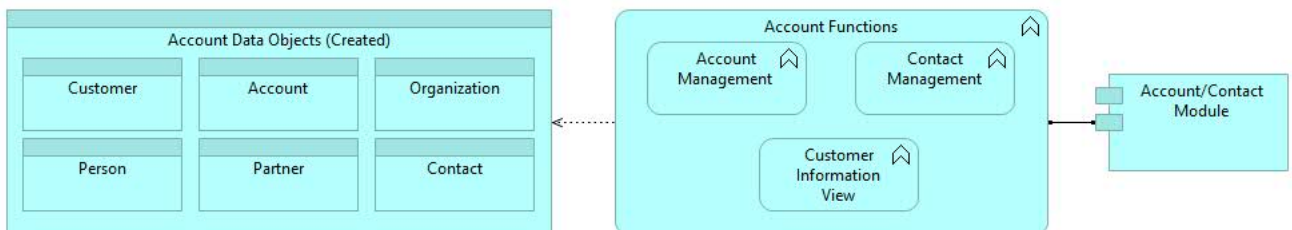


Figure 3.19: Account/Contact Module

- **Sales module** contains activity management, sales pipeline management, competitor tracking, contracts management, lead management, web to lead capture, opportunity management, quote

management, sales forecasting, quota management, territory management, order management and product and catalog management functions, Figure 3.20;

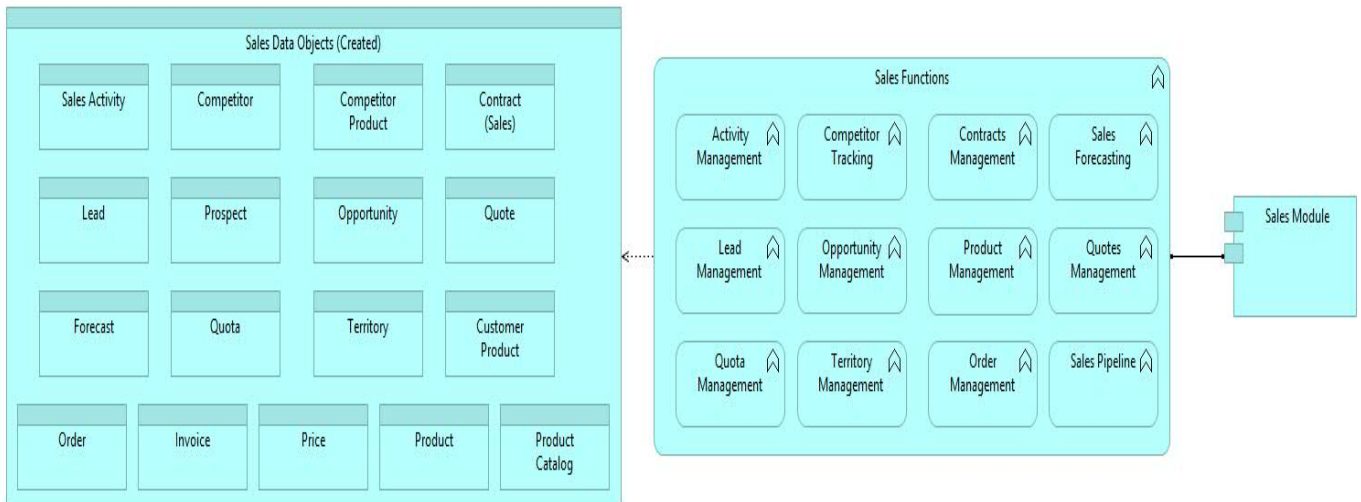


Figure 3.20: Sales Module

- **Service module** contains service contracts, case escalation and notification, case routing and queuing, case management, email management functions, Figure 3.21;

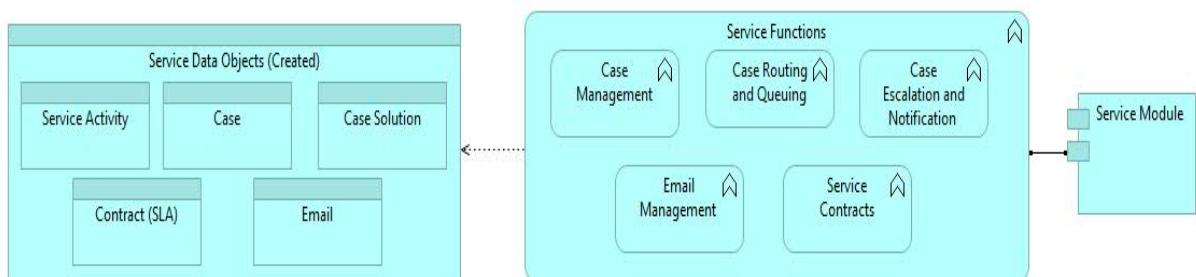


Figure 3.21: Service Module

- **Marketing module** contains campaign management, campaign execution, email marketing, marketing campaigns, newsletter management and marketing list management functions, Figure 3.22;

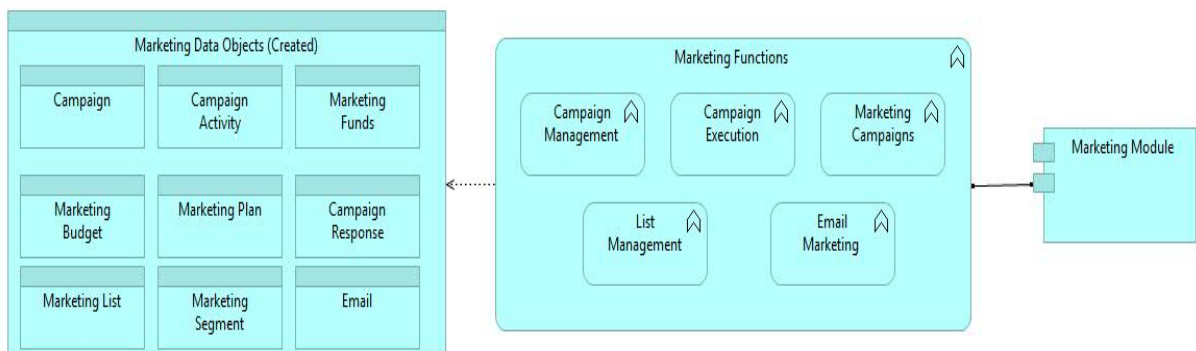


Figure 3.22: Marketing Module

- **Security/Administration module** contains user authentication, team authentication, role authentication and field permissions functions, Figure 3.23;

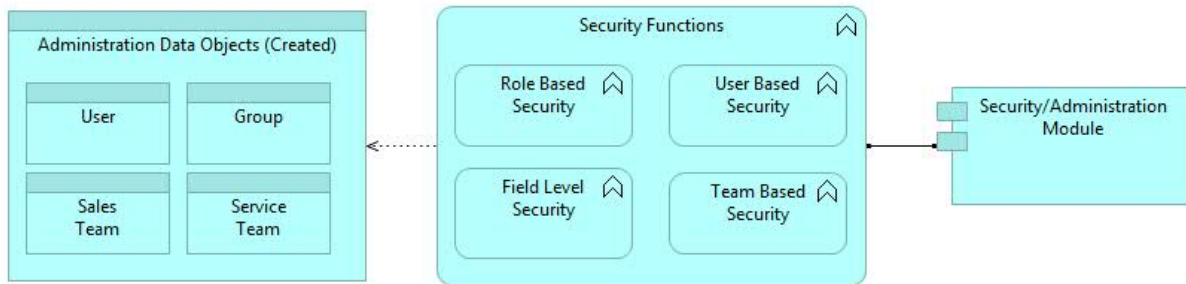


Figure 3.23: Security Module

- **Scheduler module** contains calendar management, Figure 3.24;

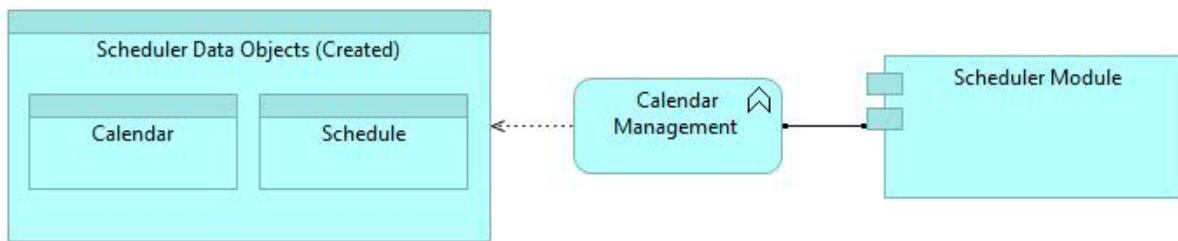


Figure 3.24: Scheduler Module

- **Portal** contains customer self-service portal functions and mobile CRM functions, Figure 3.25;

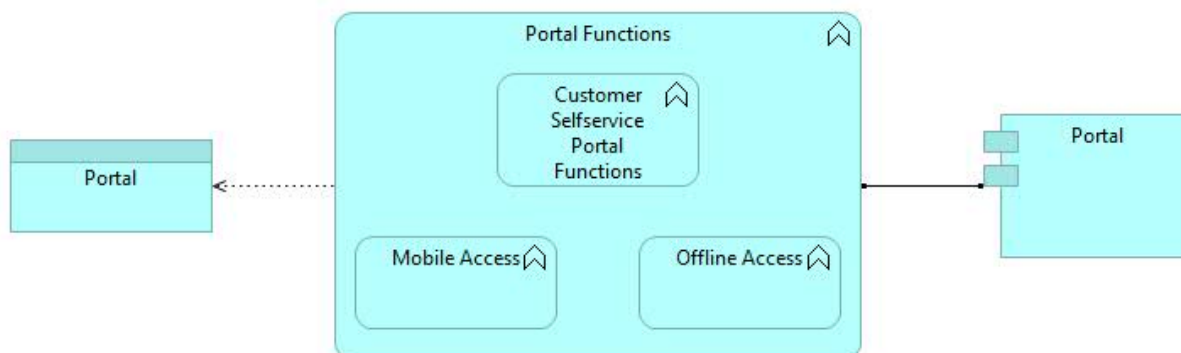


Figure 3.25: Portal System

- **Contact Center System** contains contact center function, Figure 3.26;

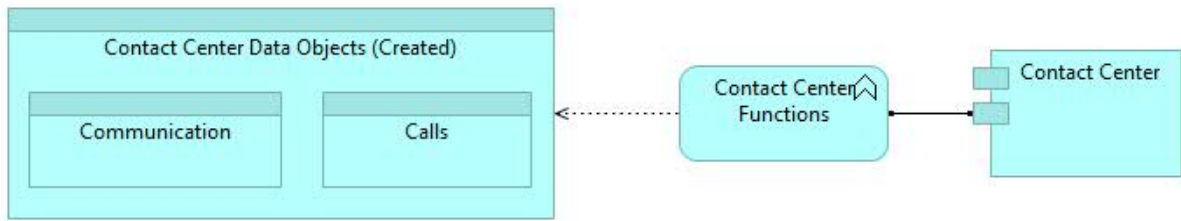


Figure 3.26: Contact Center System

- **Document and Knowledge Management System** contains sales literature, document management and knowledge management functions, Figure 3.27;

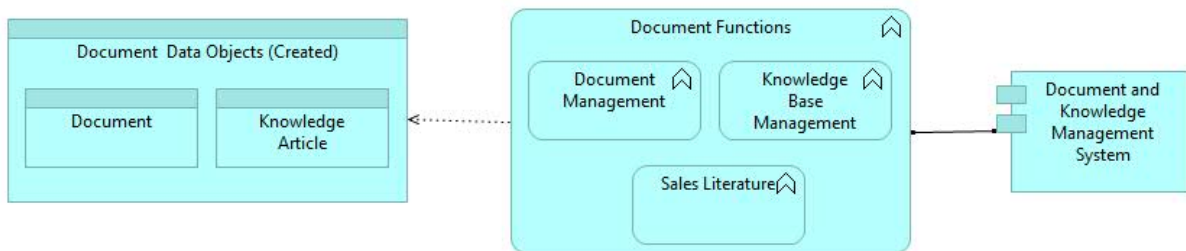


Figure 3.27: Document and Knowledge Management System.

- **Workflow Management System** contains workflow processes automation management function, Figure 3.28;

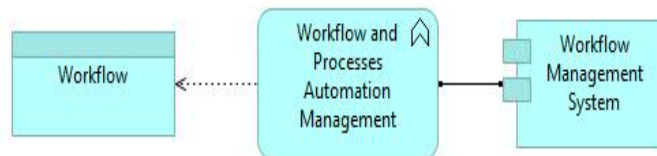


Figure 3.28: Workflow Management System

- **Reporting and Analytics System** contains dashboards, reports, sales analytics, service analytics and marketing analytics functions, Figure 3.29;

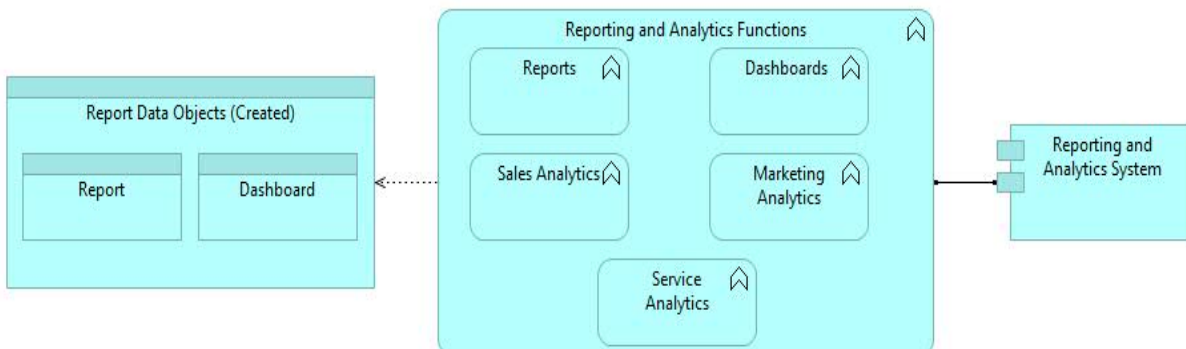


Figure 3.29: Reporting and Analytics System

In the next section we present a comparison between the two Reference Architectures reached in each step.

3.4.1 Comparison of the two Reference Architectures

Comparing the Reference Application Architecture from step 2 and the Reference Application Architecture from step 3, we identify some differences between them. The main differences regarding the CRM system are the fact that the Reference Architecture from step 3 has a new module, the Account module, and the Integration module doesn't exist. These differences occur because in the CRUD matrix the mapping between process functions and information entities resulted in a module in CRM only for the customer information, the Account module. In the Reference Architecture from step 2, the functions regarding customers information were scattered between the Sales module and Service module.

Regarding Integration module the reason for not being present in Reference Application Architecture in step 3, is due the fact that we considered that the Integration domain is positioned in the Technological domain. So we assumed the Integration module in step 2 as a guide and knowledge required for a CRM integration, and we decided not to put the integration functions in the CRUD matrix in the step 3.

There are other five important differences:

- The Document module in Reference Architecture in step 2 is now in a system that interacts with CRM and has also the knowledge base management function that in the Reference Architecture in step 2 was in Service module;
- The Workflow module and Reporting module in step 2, are two different systems that interact with CRM system in step 3;
- Two systems appear in step 3, the Portal system and Contact Center system, which were only referred in step 2 inside the Service module;

These were all the differences between the Reference Architectures in the two steps of the methodology.

3.5 Principles Identified in the Reference Architecture

In this section, we present in the Table 3.1 the principles identified in our Reference Architecture, and in the Table 3.2, we demonstrate which part of the Reference Architecture satisfies those principles, and also what Quality Attributes those principles bring with them.

Table 3.1: Reference Architecture Principles Information

Enterprise Architecture Principles Identified	Type of Information	Implications
A.2 Customers Have a Single Point of Contact	Business	Existence of one access point for customers, with capacity to handle customers requests completely.
A.5 Processes Are Standardized	Business	Standardized processes based on best practices.
A.11 Front-Office Processes Are Separated from Back-Office Processes	Business,Data,Application	Partition between Front-Office and Back-Office processes is defined.
A.13 The Status of Customer Requests Is Readily Available Inside and Outside the Organization	Data,Application	Customer Request is always available and update when a changes in its status occurs.
A.23 Documents Are Stored in the Document Management System	Data	Existence of a Document Management system where all the incoming or outgoing documents are stored.
A.24 Reporting and Analytical Applications Do Not Use the Operational Environment	Data,Application	Existence of a Data Warehouse environment and the reports being based on data loaded some time ago.
A.28 Applications Are Modular	Application	Applications are decomposed into component that have limited and acyclical dependencies between them.
A.29 Application Functionality is Available Through an Enterprise Portal	Application	Existence of a Enterprise Portal that provides access to the applications functionalities.
A.41 Processes Are Supported by a Business Process Management System	Application,Technology	Existence of a Business Process Management system that automates business processes.
A.52 Authorizations Are Role-Based	Application,Technology	Existence of an administration of roles which is the basis for authorizations and roles are related to responsibilities.
A.55 Access to IT Systems Is Authenticated and Authorized	Application,Technology	Users authenticate before using na IT system and the user identity determinates the access rights.

Table 3.2: How Reference Architecture satisfies the identified Enterprise Architecture Principles

Enterprise Architecture Principles Identified	Reference Architecture Satisfies EA Principle with	Quality Attributes
A.2 Customers Have a Single Point of Contact	Portal and Contact Center	Usability, Efficiency
A.5 Processes Are Standardized	Processes and functions are based on CRM industry best practices	Reliability, Efficiency, Maintainability, Portability
A.11 Front-Office Processes Are Separated from Back-Office Processes	CRM normally have Front-Office Processes supported by Back Office Systems.	Maintainability
A.13 The Status of Customer Requests Is Readily Available Inside and Outside the Organization	Account Module	Usability
A.23 Documents Are Stored in the Document Management System	Document and Knowledge Management System	Functionality, Reliability, Usability
A.24 Reporting and Analytical Applications Do Not Use the Operational Environment	Reporting and Analytics System	Reliability, Efficiency, Maintainability
A.28 Applications Are Modular	CRM system composed by 6 modules	Reliability, Maintainability, Portability
A.29 Application Functionality is Available Through an Enterprise Portal	Portal System	Usability
A.41 Processes Are Supported by a Business Process Management System	Workflow Management System	Maintainability,Efficiency
A.52 Authorizations Are Role-Based	User/Administration Module	Maintainability
A.55 Access to IT Systems Is Authenticated and Authorized	User/Administration Module	Functionality

3.6 Critical Analysis

In this chapter, we presented the solution that we propose to solve the thesis problems. We began by explaining the method that we used to define the Reference Architecture solution. The starting point for developing a Reference Architecture is the definition of its Vision, Mission and Strategy. After defining the focus of the Reference Architecture, we decided to identify the best practices in the domain of the

Reference Architecture. So we analysed five CRM commercial solutions and extracted the common features between them and create a Reference Application Architecture from those features. Then we extracted the common information entities from those CRM solutions and with the functionalities previously extracted, we used the CRUD matrix to define a improved Reference Application Architecture. The extracted common information entities allowed us also to define a Reference Information Architecture for the CRM domain.

The second architecture is more complete than the first one, because it identifies the principal systems that interact with CRM system and the common modules belonging to the CRM system are more in accordance in what must be implemented. The reason for this is that the first Reference Architecture is only a identification of what is related to CRM system. For example, in the first Architecture we identified an Integration module, and that Integration domain is more related to the Technological domain, so we only treated this module as a guidance to what is necessary in terms of integration in a CRM.

In the end of this chapter, are identified a list of EA Principles satisfied by the Reference Architecture. This identification is important, because the EA Principles are guides to follow in the definition of an EA, which in this case is for the CRM domain.

Chapter 4

Evaluation

In this chapter, we present the methodology followed for the evaluation of the Reference Architecture in case studies. This methodology corresponds to the fourth and fifth steps of the action research methodology followed for all the work. After detailing the methodology we explain each of the case studies, and for each one, we apply the evaluation methodology. For each case we also make an observation of which requirements that are not covered by our Reference Architecture and that can be added to improve our solution, if these observations are considered a pattern. In the last section of the chapter, we make an analysis of the results obtained from the assessment made.

4.1 Evaluation Methodology

The evaluation methodology that we propose is a merge of two of the topics presented in the related work chapter: the methodology for specific architectures generation and the metrics and heuristics for information systems. Before detailing the evaluation methodology, is important to refer that the evaluation that we do, focuses in the modules of a CRM system and the systems that interact with it. Next we detail the evaluation methodology steps:

- **Step 1:** we start by examining the provided case study documents. In those documents we identify the information required to define the business processes, the information entities, the CRM modules and the systems that interact with the CRM system. From this information we define a CRUD matrix, that illustrates the current state of the case study, and through that we define the Application Architecture of the case study. Then with the same information extracted, we use the methodology for specific architectures generation, to reach a specific architecture using our Reference Architecture. We analyse how our Reference Architecture satisfies the requirements of the case study, and with that analysis we define a CRUD matrix. With the CRUD matrix defined, we represent the specific architecture view for the case study. This step of the evaluation methodology is illustrated in Figure 4.1.

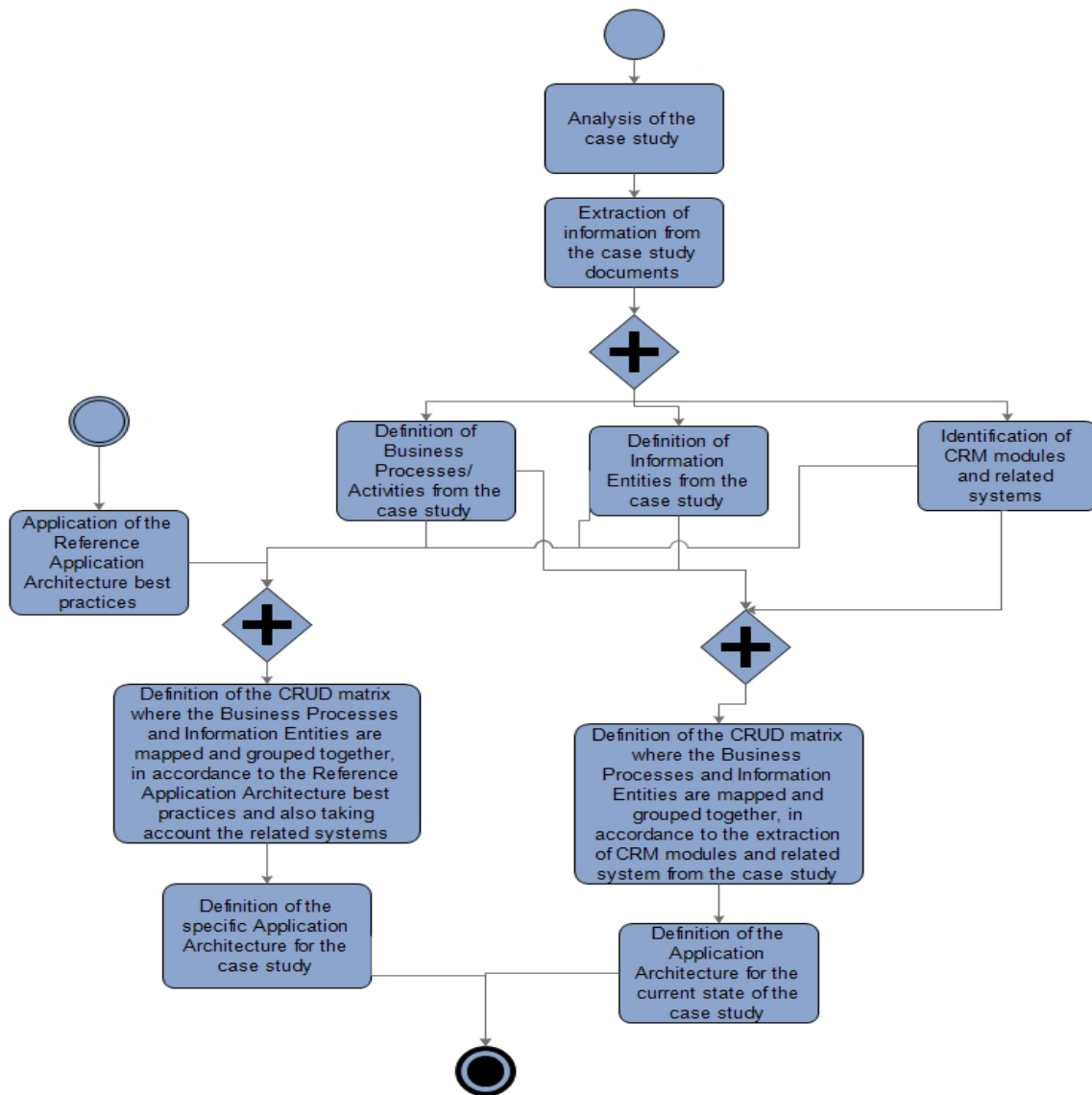


Figure 4.1: Evaluation Step 1 Workflow

- **Step 2:** we apply the the metrics and heuristics for information systems evaluation, detailed in section 2.1.3, in the current architecture of the case study and in the specific architecture reached for the same case with the Reference Architecture. Next we compare the results to assess the benefits and pitfalls of the purposed architecture. This step of the evaluation methodology is illustrated in Figure 4.2.

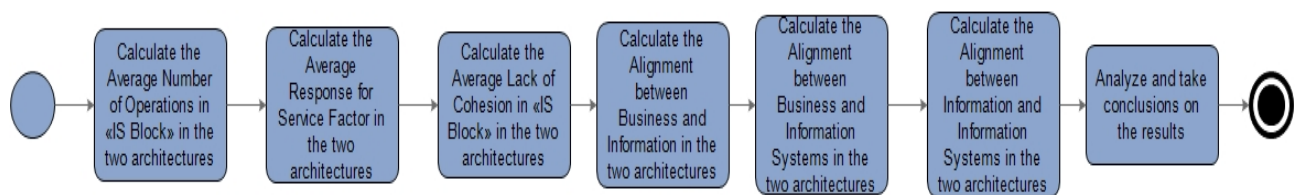


Figure 4.2: Evaluation Step 2 Workflow

4.2 Case Studies

In this section we present each of the case studies used for evaluation and the results obtained in those evaluations. All the cases were provided by the Agency for Administrative Modernization (AMA), after some meetings in order to get access to the documents of those cases. Is important to refer that the cases which we evaluate are all in the public sector domain, despite our Reference Architecture being based on CRM solutions for mid-market companies. This aspect results that during the evaluation that some information systems, CRM modules and information entities present in the cases will not be satisfied by the Reference Architecture, because they are very focused on public sector. If in the case studies are identified some patterns that aren't covered by the Reference Architecture, those patterns may be added to the Reference Architecture in the end of the evaluation, in order to improve the Reference Architecture in terms of the public sector domain.

The case studies, that we are going to present in the following sections, differ in their dimensions and their purpose. All the organizations referred in the case studies, have the goal of providing services to the citizens, but they differ in the coverage and complexity of those services. The Portuguese High Commissioner for Migration offers simpler services and doesn't have many systems integrating with the CRM. The Citizen Spaces has a higher coverage, much more citizens covered, more information and have more systems integrating with the CRM. It's important to have these differences between the cases, because they provide an opportunity to verify, if the Reference Architecture proposed can be adequate in the Portuguese Public Administration. The following table presents the main goals of the next case studies described.

Table 4.1: Case Studies and their objectives

Case Studies	Objectives
High Commissioner for Migration:	Identify the business processes related to the CRM domain; Identify the information entities related to the CRM domain; Identify the CRM modules and the systems that interact with the CRM; Represent the current state of the Application Architecture; Propose an Application Architecture based on the Ref. Architecture guidance; Measure in both Application Architectures the Change Facility; Measure in both Application Architectures the Test Facility; Measure in both Application Architecture the fulfilment of the Alignment Heuristics; Compare the results;
Citizen Spaces:	Identify the business processes related to the CRM domain; Identify the information entities related to the CRM domain; Identify the CRM modules and the systems that interact with the CRM; Represent the current state of the Application Architecture; Propose an Application Architecture based on the Ref. Architecture guidance; Measure in both Application Architectures the Change Facility; Measure in both Application Architectures the Test Facility; Measure in both Application Architecture the fulfilment of the Alignment Heuristics; Compare the results;

4.2.1 High Commissioner for Migration (ACM)

The ACM is a public institute with the goal of cooperating in the definition, execution and evaluation of public politics, transverse and sectorial policies on migration, relevant to the attraction of migrants in national and international contexts. The ACM is also responsible for the integration of immigrants and

ethnic groups, in particular Roma communities, and for managing and valuing diversity among cultures, ethnicities and religions. Being the focus of our work the CRM domain, we had the opportunity of trying out the CRM of ACM.[Ruivo, 2012] It is an open source CRM different from the ones we used to define the Reference Architecture. While we experienced the CRM from ACM, we extracted the businesses processes that could be done in this CRM as well as the information entities. The business processes extracted from ACM CRM are:

- **Customer Management Processes:** Create Customer Form, Associate Customer, Edit Customer, List Customers, illustrated in Figure A.1;
- **Processes Management Processes:** Create Process, Edit Process, List Processes, Close Process, Write a Note, illustrated in Figure A.2;
- **Tasks Management Processes:** Create Task, Edit Task, List Tasks, Write a Note, illustrated in Figure A.3;
- **Follow-up Management Processes:** Edit Follow-up, List Follow-ups, Do a Follow-up on a Process, illustrated in Figure A.4;
- **Service Management Processes:** Begin Service, Close Service, Resume Service, List Services, illustrated in Figure A.5;
- **Scheduler Management Processes:** Create Schedule, Edit Schedule, Assign Ticket, List Schedules, Call Customer, illustrated in Figure A.6;
- **Document Management Processes:** Upload Document, Disassociate Document, Edit Document, List Documents, Visualize Procedural Document, Visualize Personal Document, illustrated in Figure A.7;
- **Administration Management Processes:** List Users, Create New User, User Profile, Edit User, List Teams, Create Team, Team Profile, Edit Team, List Profiles, Create New Profile, Profile of Profile, Edit Profile, List Modules, Module Profile, Edit Module, illustrated in Figure A.8;
- **Report Management Processes:** Generate Offices Call Report, Generate Service Users Report, Generate Report Office Processes, Generate Report Processes Users, illustrated in Figure A.9;
- **User Management Processes:** User Sign Up, User Login, User Profile, Password Recovery, Visualize Notifications, illustrated in Figure A.10;

The information entities identified are illustrated in Figure 4.3:

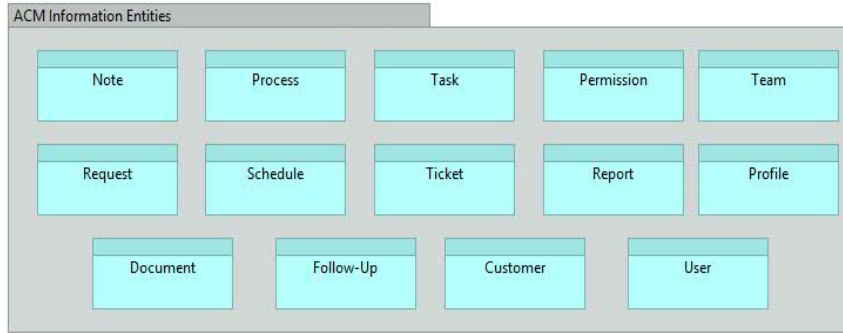


Figure 4.3: ACM CRM Information Entities

We also identified the main modules of this CRM and the interacting systems, which are going to be represented in the CRUD matrix and in the Application Architecture. With all these information we had obtained the requirements of the ACM CRM. The next part of the work is to define a CRUD matrix that exemplifies the current ACM CRM state, with the information extracted referred above. The CRUD matrix is illustrated in Figure 4.4.

ACM CRM Application Architecture

Processes/ Information Entities	Client	FollowUp	Process	Task	Note	Scheduler	Ticket	Request	Report	Document	User	Permission	Team	Profile
Create Customer Form	CRUD													
Associate Customer	RU													
Edit Customer	RU													
Do a Follow Up on a Process		CRUD	RU											
Edit Follow Up		RU												
Create Process	R		CRUD											
Write a Note on a Process			RU		CRUD									
Edit Process			RU											
Close Process			RU											
Create Task			RU	CRUD										
Write a Note on a Task				RU	CRUD									
Edit Task				RU										
Create Schedule						CRUD								
Edit Schedule						RU								
Call Customer	R					RU	R							
Assign Ticket						R	CRUD							
Begin Service	R	R	R	R	R			CRUD		R				
Close Service								RU						
Resume Service								RU						
Generate Offices Call Report							R		CRUD			R		
Generate Report Office Processes							R		CRUD		R			
Generate Service Users Report			R						CRUD			R		
Generate Report Processes Users			R						CRUD		R			
Upload Document	R		R							CRUD				
Disassociate Document										RU				
Edit Document										RU				
User Sign Up	R										CRUD			
User Login											RU			
Recover Password											RU			
Create New User											CRUD	CRUD	CRUD	
Create New Team												CRUD		CRUD
Create New Profile														CRUD
Edit User											RU			
Edit Team												RU		
Edit Profile													RU	
Edit Module												RU		
List Customers	R													
List Follow Up		R												
List Processes			R											
List Tasks				R										
List Schedules					R									
List Services								R						
List Documents									R					
User Profile											R			
User Alerts											R			
List Users											R			
User Profile											R			
List Teams												R		
Teams Profile													R	
List Profiles														R
Profile of Profile														R
List Modules												R		
Module Profile												R		

Figure 4.4: ACM CRM CRUD Matrix

By analysing this CRUD Matrix, we define the Application Architecture view present in Figure 4.5.

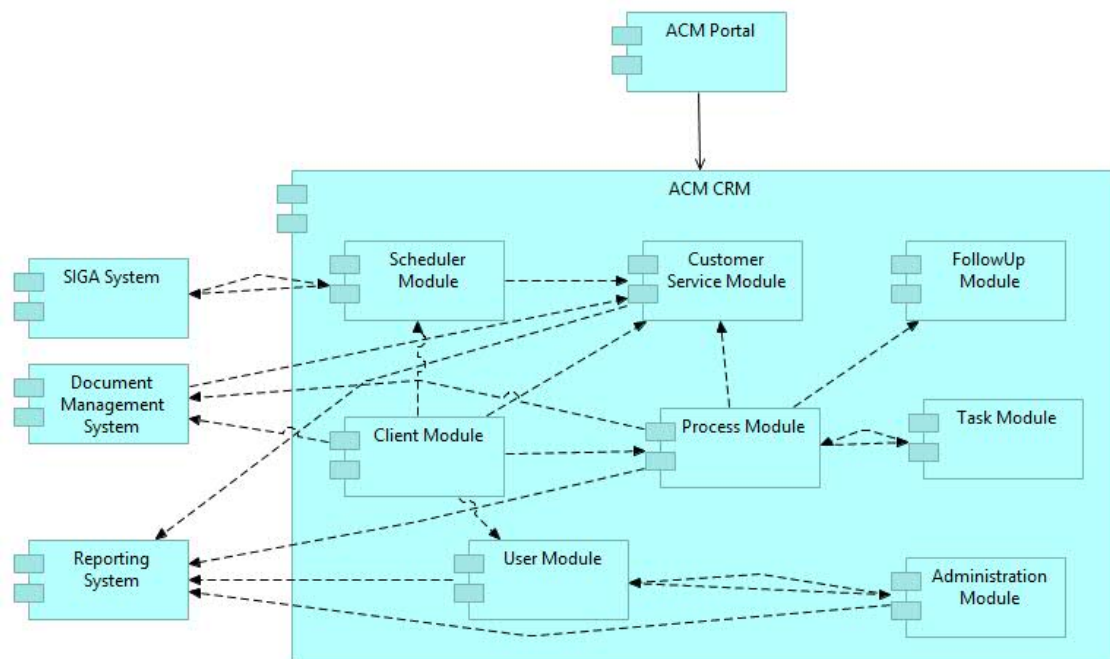


Figure 4.5: ACM CRM Application Architecture

In the Application Architecture of the ACM CRM, we identified eight modules and four systems which interacted with the CRM:

- **Processes Module:** provides processes management functionalities;
- **Follow-Up Module:** provides follow-up management functionalities;
- **Task Module:** provides tasks management functionalities;
- **Client Module:** provides client account management functionalities;
- **Customer Service Module:** provides customer service management functionalities;
- **Scheduler Module:** provides scheduler management functionalities;
- **User Module:** provides users management functionalities;
- **Administration Module:** provides administration management functionalities;
- **Document Management System:** provides document management functionalities;
- **Reporting System:** provides reporting functionalities;
- **Integrated Service Management System (SIGA) System:** provides ticket management functionalities;
- **Portal System:** front-end platform that communicates with CRM API's and provides an easier interface. This system wasn't identified in the CRUD matrix, instead it was identified through the contact with AMA responsible;

Application Architecture Solution for ACM CRM by Ref. Architecture

We arrived at the Application Architecture of the ACM CRM, so is time to reach a specific architecture based on our Reference Architecture. We know the requirements of the ACM CRM so we apply the best practices of our Reference Architecture to reach a solution architecture for this case. We reach a CRUD matrix with the Reference Architecture guidance. The changes made because of the best practices are:

- We grouped the Request, Follow Up, Process and Task modules into one module, the Service module, because the Service module in our architecture has the case management functions, which includes the functions of these four module;
- The User and Administration modules correspond to a single module in our architecture, the User/Administration module;
- The Client module in our architecture goes by the name of Account module;
- The SIGA system that wasn't covered by our Reference Architecture stayed the same;

The CRUD matrix reached is illustrated in Figure 4.6.

	Client	FollowUp	Process	Task	Note	Request	Ticket	Scheduler	Report	Document	User	Permission	Team	Profile
Processes/ Information Entities														
Create Customer Form	CRUD													
Associate Customer	RU													
Edit Customer	RU													
Do a Follow Up on a Process		CRUD	RU											
Create Process	R		CRUD											
Create Task			RU	CRUD										
Write a Note on a Process			RU		CRUD									
Write a Note on a Task				RU	CRUD									
Begin Service	R					CRUD				R				
Edit Follow Up		RU												
Edit Process			RU											
Close Process			RU											
Edit Task				RU										
Close Service						RU								
Resume Service						RU								
Assign Ticket							CRUD	R						
Create Schedule								CRUD						
Edit Schedule								RU						
Call Customer	R						R	RU						
Generate Offices Call Report						R			CRUD				R	
Generate Report Office Processes						R			CRUD		R			
Generate Service Users Report			R						CRUD				R	
Generate Report Processes Users			R						CRUD		R			
Upload Document	R		R							CRUD				
Disassociate Document										RU				
Edit Document										RU				
User Sign Up	R										CRUD			
Create New User											CRUD			
Create New Team											CRUD			
Create New Profile											CRUD			
User Login											RU			CRUD
Recover Password											RU			
Edit User											RU			
Edit Team											RU			
Edit Profile											RU			
Edit Module											RU			RU
List Customers	R													
List Follow Up			R											
List Processes		R												
List Tasks				R										
List Schedules								R						
List Services														
List Documents						R				R				
User Profile											R			
User Alerts											R			
List Users											R			
User Profile											R			
List Teams												R		
Teams Profile												R		
List Profiles													R	R
Profile of Profile														R
List Modules											R			
Module Profile											R			

Figure 4.6: ACM CRM CRUD Matrix by Reference Architecture

By analyzing this CRUD Matrix, we define the Application Architecture view present in Figure 4.7.

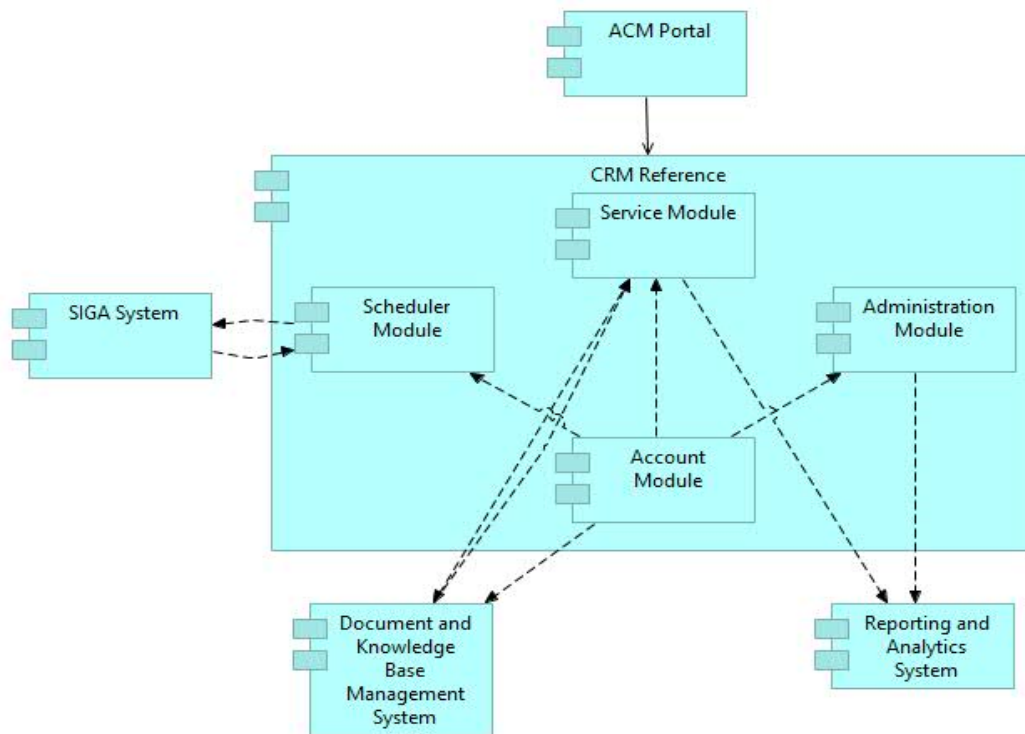


Figure 4.7: ACM CRM Application Architecture by Reference Architecture

In the Application Architecture for the ACM CRM case based on our Reference Architecture, we identify four modules of the CRM System and four systems that interact with CRM System:

- **Account Module:** provides client account management functionalities;
- **Service Module:** provides customer service management, processes management, follow-up management and the tasks management functionalities;
- **Scheduler Module:** provides scheduler management functionalities;
- **Reporting and Analytics System:** provides reporting functionalities;
- **User/Administration Module:** provides administration and user management functionalities;
- **Document and Knowledge Management System:** provides document management functionalities;
- **SIGA System:** stays the same;
- **Portal System:** stays the same;

ACM Evaluation Results

The final step is to evaluate both application architectures with the metrics and heuristics to evaluate information systems chosen in section 2.1.6.

Average Number of Operations in «IS Blocks»:

$$NOIS\ ACM\ Architecture = \frac{11}{36} = 0,305 \quad (4.1)$$

$$NOIS\ ACM\ Ref.\ Architecture\ solution = \frac{7}{36} = 0,194 \quad (4.2)$$

Response for a Service Factor: For this metric we had to generalize Services that are realized by the Information Systems. In Appendix C in Figure C1-C2 are illustrated those services and the systems that are used in their realization.

$$RSF\ ACM\ Arch. = \frac{12}{2+6+3+5+7+3+8+3+7+4+5+5} = \frac{12}{58} = 0,207 \quad (4.3)$$

$$RSF\ ACM\ Ref.\ Arch.\ solution = \frac{12}{2+4+3+4+6+5+5+5+5+4+4+4} = \frac{12}{51} = 0,235 \quad (4.4)$$

Lack of Cohesion in «IS Block» Factor:

$$LCOISF\ ACM\ Architecture = 1 - \frac{36}{6 \times 37 \times 15} = 1 - \frac{36}{5544} = 0,9935 \quad (4.5)$$

$$LCOISF\ ACM\ Ref.\ Architecture\ solution = 1 - \frac{36}{7 \times 36 \times 14} = 1 - \frac{36}{3528} = 0,9897 \quad (4.6)$$

Table 4.2: Fulfilment of the Alignment Heuristics by Architecture from ACM

	H1.1	H1.2	H1.3	H2.1	H2.2	H2.3	H3.1	H3.2	H3.3	H3.4	H3.5	H3.6	H3.7
Meets	X		X	X	X	X			X	X	X	X	X
Fails		X					X	X					

Table 4.3: Fulfilment of the Alignment Heuristics by solution from Reference Architecture

	H1.1	H1.2	H1.3	H2.1	H2.2	H2.3	H3.1	H3.2	H3.3	H3.4	H3.5	H3.6	H3.7
Meets	X		X	X	X	X	X		X	X	X	X	X
Fails		X						X					

We reach the following results in these metrics evaluation:

Table 4.4: Evaluation Results on ACM CRM

Evaluated Qualities Criteria	ACM Case Study Current State	ACM Case Study by Reference Architecture
Change Facility (Application)	0,305	0,194
Test Facility	0,207	0,235
Change Facility (Informational / Application)	0,9935	0,9897
Alignment between Business Architecture and Information Architecture	0,666(66,6%)	0,666(66,6%)
Alignment between Business Architecture and Information Systems Architecture	1(100%)	1(100%)
Alignment between Information Architecture and Information Systems Architecture	0,714(71,4%)	0,857(85,7%)

From the Table 4.4, that presents the results of the evaluation done on ACM case, we can conclude that the solution from the Reference Architecture has a better alignment between Information Architecture and Information Systems Architecture and a better test facility, but it has worst change facility. The test facility result is better because the solution from Reference Architecture uses less CRM modules. The change facility is worse because of that same factor, the current state of ACM architecture has more CRM modules, so is easier to change than an architecture with less modules. Regarding the better alignment between Information Architecture and Information Systems Architecture, this happens because in the ACM current architecture there are various modules that manage the same entities, and in the solution from the Reference Architecture this doesn't happen.

Observation

In this case study two information entities weren't covered by our Reference Architecture, the note and ticket information entities. The process and follow up entities correspond to the case information entity in our Reference Architecture.

In relation to the interacting systems our Reference Architecture also didn't took into account the SIGA system for ticket management. Regarding ticket management our Reference Architecture only took in account that the CRM manage the cases and route those same cases.

4.2.2 Citizen Spaces (CS)

Citizen Space is an instrument of social inclusion and territorial cohesion under public service rules regarding the politic of digitalization. Through a fine mesh network of on-site care services disseminated by national territory, all citizens and economic operators, will have access to the benefits of the electronic services of the Portuguese State. We analyzed this case through a set of documents [AMA, 2014a,b,c,d] provided by AMA about the Citizen Space, and by meeting with AMA responsible for SugarCRM in Citizen Space, the Eng. José António Rodrigues.

We started by identifying the business processes and the information entities related to the Citizen Spaces. The business processes identified are the following ones:

- **Customer Service Management Processes:** begin service, identify citizen, execute request, close request, close service;

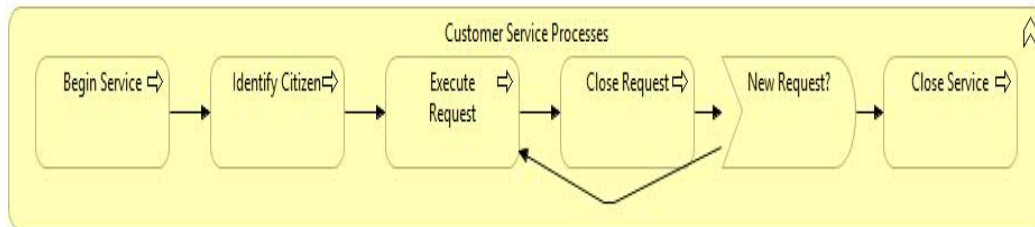


Figure 4.8: CS CRM Service Management Processes

- **Cashier Management Processes:** open cashier, close cashier, validate cashier;

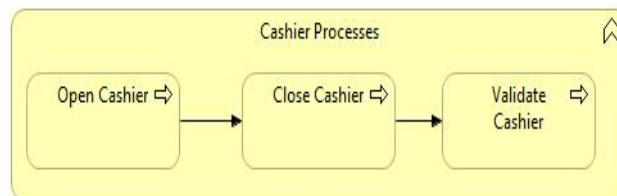


Figure 4.9: CS CRM Cashier Management Processes

- **List and Search Requests Processes;**
- **Citizen Spaces Management Processes:** create entity hostess, create citizen space and associate to entity hostess, create entity service provider, create type of request and associate to entity service provider, create user, associate user to citizen space, associate type of request to citizen space;

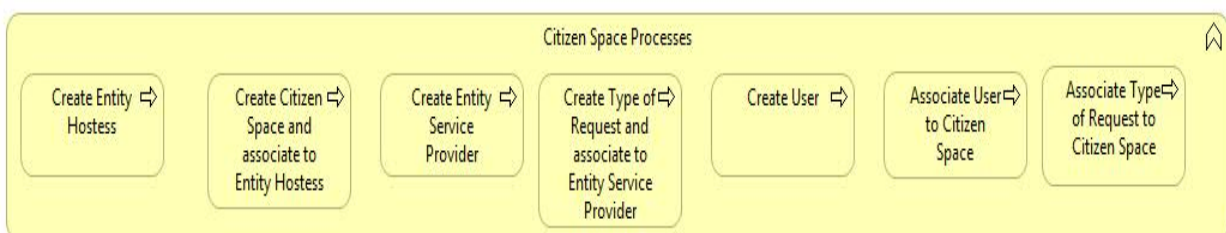


Figure 4.10: CS CRM Citizen Spaces Management Processes

- **Entity Management Process;**
- **Services Management Process;**
- **User Management Process;**
- **Payment Methods Management Process;**

- **Tax Rate Management Process;**

- **Access Records Processes on:** Service, Requests, Cashiers, Citizens, Cashier Items, Billing;

The information entities were already identified in the documents and are illustrated in Figure 4.11:

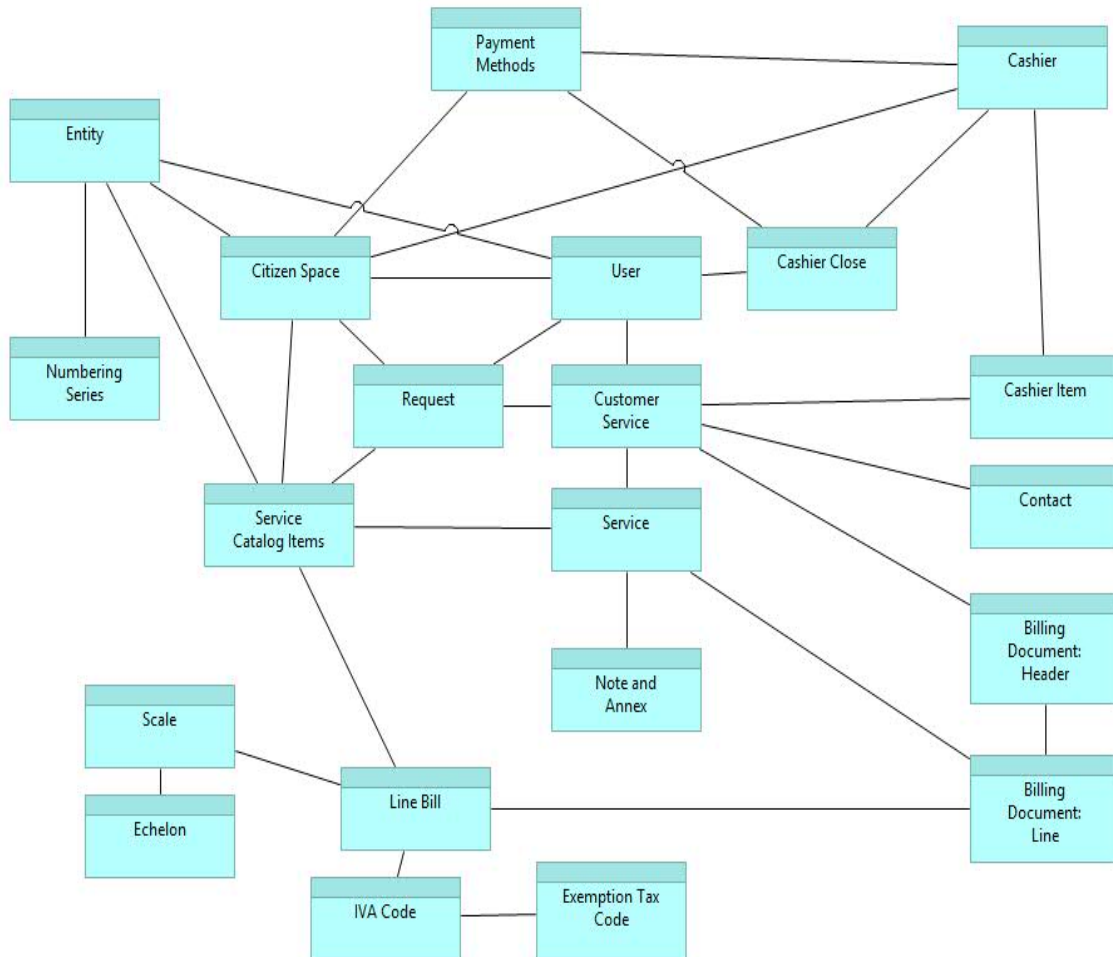


Figure 4.11: CS Data Model

CS CRM Application Architecture

With the information extracted, we map the business processes with the information entities in a CRUD matrix. We need for this part of the work, to identify in the documents the modules of CRM used in the CS, which support the business processes and the information entities. The CRM used in CS is the SugarCRM, one of the CRM solutions that we used to define our Reference Architecture, but customized for this specific case. We identified six CRM modules, and with all this data we defined the CRUD matrix, illustrated in Figure 4.12.

Processes/ Information Entities	Customer Service	Contact	Request	Cashier	Cashier Items	Cashier Close	Note/Anex	Document Billing: Header	Billing Document: Line	Billing Line	Scale	Echelon	IVA Code	Code Tax Exemption	Entities	Series Numbering	Citizen Space	Payment Method	User	Service	Services Catalog Items
Begin Service	CRUD			R															R		
Identify Citizen	RU	CRUD																			
Execute Request	RU		CRUD																	R	R
Close Service	RU																				
Open Cashier	RU			CRUD	CRUD														R		
Close Cashier	RU			RU		CRUD													R		
Validate Cashier				RU															R		
Close Request	RU		RU				CRUD	CRUD	CRUD												
Tax Rate Management								R	CRUD	CRUD	CRUD	CRUD	CRUD								R
Entities Management															CRUD	CRUD					
Create Entity Hostess															CRUD	CRUD					
Create Entity Service Provider															CRUD	CRUD					
Create Citizen Space and Associate with Entity Service Provider															RU		CRUD				
Payment Methods Management																	R	CRUD			
Users Management																			CRUD		
Create User																			CRUD		
Associate User with Citizen Space																	RU		RU		
Services Management																				CRUD	CRUD
Create Request Type and Associate with Entity Service Provider															RU					CRUD	CRUD
Associate Request Type with Citizen Space																	RU			RU	
List and Search Requests			R																		
Access Service Records	R																				
Access Requests Records			R																		
Access Citizen Records		R																			
Access Cashier Records				R																	
Access Cashier Items Records					R																
Access Billing Records							R	R													

Figure 4.12: CS CRUD Matrix

By analysing the CRUD matrix and also, with the identification in those same provided documents of the systems that interact with the CRM system, we reach the following Application Architecture, in Figure 4.13.

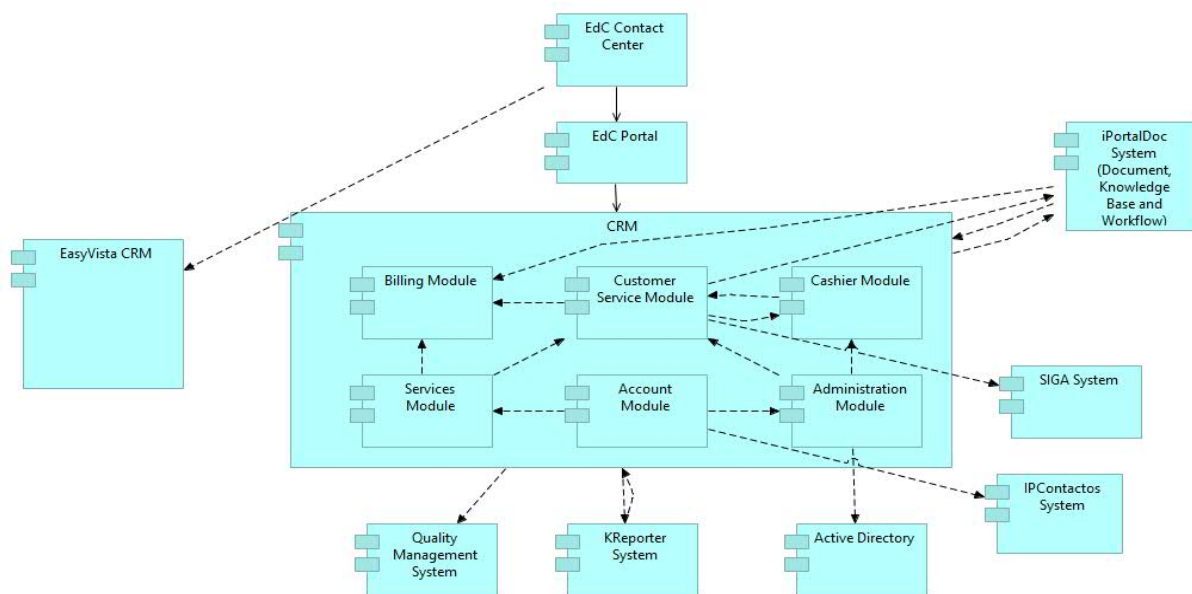


Figure 4.13: CS Application Architecture

The six modules identified are:

- **Customer Service Module:** provides customer service management functionalities;
- **Cashier Module:** provides cashier management functionalities;
- **Billing Module:** provides billing management functionalities;
- **Administration Module:** provides users management functionalities;
- **Services Module:** provides services, services catalogues and the taxes applied management functionalities;
- **Entities Module:** provides entities management functionalities;

The systems that interact with the CRM system are:

- **Contact Centre System:** system that interacts with the CRM to provide customer service through calling services;
- **CS Portal:** front-end platform that communicates with CRM APIs and provides an easier interface;
- **SIGA System:** system responsible for ticket management;
- **Quality Management System:** system responsible for all complaints, suggestions and quality surveys;
- **IPContactos System:** system where the contacts from SugarCRM are synchronized to daily;
- **iPortalDoc System:** system with document, knowledge base and workflow management functionalities;
- **KReporter System:** plug-in system for SugarCRM, which provides reporting functionalities;
- **Active Directory:** system where are located all the users and their activity;
- **EasyVista CRM System:** CRM system specialized in Information and Communication Technology (ICT) services and that is compliant with the Information Technology Infrastructure Library (ITIL) framework;

Application Architecture Solution for CS CRM by Ref. Architecture

We arrived at the Application Architecture of the Citizen Spaces, so is time to reach a concrete architecture based on our Reference Architecture. We know the requirements of the Citizen Spaces, so we apply the best practices of our Reference Architecture to reach a solution architecture for this case. We define a CRUD matrix with the Reference Architecture guidance. The changes made because of the best practices are:

- The Billing module and Services module functions in this case, are satisfied by the Sales module in ours Reference Architecture;

- The functionalities of the Quality Management system in this case, are satisfied in the Service module of the CRM system in our architecture through service contracts functionalities (service level agreements, SLAs);
- The iPortalDoc system functionalities in this case, is accomplished in our architecture by two systems: the Document and Knowledge Base Management system and the Workflow Management system;
- The Entities module corresponds to the Account module in our architecture;
- The systems that weren't covered by our Reference Architecture stayed the same. In this case those systems are: the EasyVista CRM, the SIGA system, the iPContactos system, the Active Directory system and the Cashier CRM module;

The CRUD matrix reached is present in Figure 4.14.

Processes / Information Entities	Cashier	Cashier Items	Cashier Close	Customer Service	Contact	Request	Note/Anex	Document Billing: Header	Billing Document: Line	Billing Line	Scale	Echelon	IVA Code	Code Tax Exemption	Service	Services Catalog Items	Entities	Series Numbering	Citizen Space	Payment Method	User	
Open Cashier	CRUD	CRUD		RU																		R
Close Cashier	RU		CRUD	RU																		R
Validate Cashier	RU																					R
Begin Service	R			CRUD																		R
Identify Citizen				RU	CRUD																	
Execute Request				RU		CRUD								R	R							
Close Service				RU																		
Close Request				RU		RU	CRUD	CRUD	CRUD													
Tax Rate Management								R		CRUD	CRUD	CRUD	CRUD	CRUD	R							
Services Management															CRUD	CRUD						
Create Request Type and Associate with Entity Service Provider														CRUD	CRUD	RU						
Associate Request Type with Citizen Space														RU					RU			
Entities Management																	CRUD	CRUD				
Create Entity Hostess																	CRUD	CRUD				
Create Entity Service Provider																	CRUD	CRUD				
Create Citizen Space and Associate with Entity Service Provider																	RU		CRUD			
Payment Methods Management																			R	CRUD		
Users Management																						CRUD
Create User																						CRUD
Associate User with Citizen Space																			RU			RU
List and Search Requests																						
Access Service Records				R																		
Access Requests Records					R																	
Access Citizen Records					R																	
Access Cashier Records	R																					
Access Cashier Items Records		R																				
Access Billing Records							R	R														

Figure 4.14: CS CRUD Matrix from Reference Architecture

By analysing it and with the identified systems and changes from the Reference Architecture best practices, we define the Application Architecture for this case illustrated in Figure 4.15.

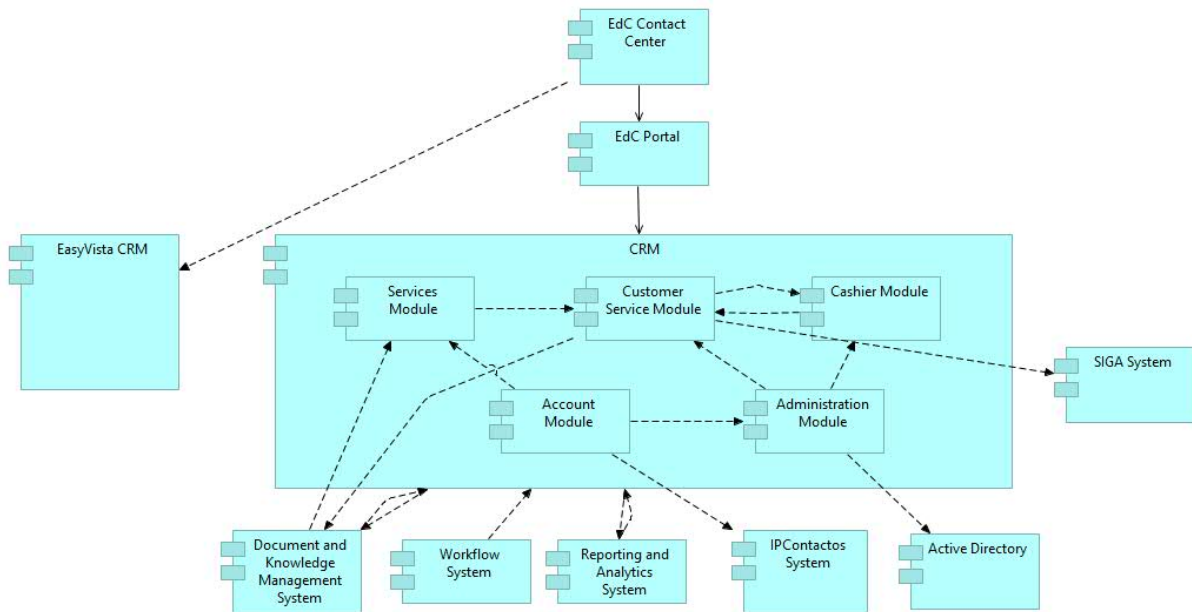


Figure 4.15: CS Application Architecture from Reference Architecture

In our Application Architecture we used five modules and nine systems:

- **Service Module:** provides customer service management functionalities;
- **Sales Module:** provides invoice management, services, services catalogue and taxes management functionalities. We used the Sales module, because in Public Sector the services provided can be seen as the products that a company sells, so we used our sales module that cover these aspects;
- **Administration Module:** provides user management functionalities;
- **Account Module:** provides entities management functionalities;
- **Cashier Module:** provides cashier functionalities;
- **Document Management and Knowledge Base Management System:** provides document management and knowledge base functionalities;
- **Workflow Management System:** provides workflow management functionalities;
- **Reporting and Analytics System:** provide reporting functionalities;
- **IPContactos:** stays the same;
- **Active Directory:** stays the same;
- **CS Contact Centre system:** stays the same;
- **CS Portal:** stays the same;
- **EasyVista CRM:** stays the same;
- **SIGA system:** stays the same;

CS Evaluation Results

The final step is to evaluate both Application architectures with the metrics and heuristics to evaluate information systems chosen in section 2.1.6.

Average Number of Operations in «IS Blocks»:

$$NOIS\ CS\ Architecture = \frac{7}{20} = 0,35 \quad (4.7)$$

$$NOIS\ CS\ Ref.\ Architecture\ solution = \frac{6}{20} = 0,3 \quad (4.8)$$

Response for a Service Factor: For this metric we had to generalize Services that are realized by the Information Systems. In Appendix C in Figure C3-C4 are illustrated those services and the systems that are used in their realization.

$$RSF\ CS\ Arch. = \frac{16}{2 + 4 + 3 + 5 + 8 + 5 + 3 + 5 + 5 + 3 + 4 + 6 + 5 + 5 + 5 + 3} = \frac{16}{71} = 0,225 \quad (4.9)$$

$$RSF\ CS\ Ref.\ Arch.\ sol. = \frac{16}{2 + 3 + 3 + 5 + 7 + 5 + 3 + 5 + 5 + 3 + 4 + 5 + 5 + 5 + 2 + 3} = \frac{16}{65} = 0,246 \quad (4.10)$$

Lack of Cohesion in «IS Block» Factor:

$$LCOISF\ CS\ Architecture = 1 - \frac{20}{7 * 20 * 21} = 1 - \frac{20}{2940} = 0,993 \quad (4.11)$$

$$LCOISF\ CS\ Ref.\ Architecture\ solution = 1 - \frac{20}{6 * 20 * 21} = 1 - \frac{20}{2520} = 0,992 \quad (4.12)$$

Table 4.5: Fulfilment of the Alignment Heuristics by Architecture from CS

	H1.1	H1.2	H1.3	H2.1	H2.2	H2.3	H3.1	H3.2	H3.3	H3.4	H3.5	H3.6	H3.7
Meets	X		X	X	X		X		X	X	X		X
Fails		X				X		X				X	

Table 4.6: Fulfilment of the Alignment Heuristics by solution from Reference Architecture

	H1.1	H1.2	H1.3	H2.1	H2.2	H2.3	H3.1	H3.2	H3.3	H3.4	H3.5	H3.6	H3.7
Meets	X		X	X	X		X		X	X	X		X
Fails		X				X		X				X	

Table 4.7: Evaluation Results on CS CRM

Evaluated Qualities Criteria	CS Case Study Current State	CS Case Study by Reference Architecture
Change Facility (Application)	0,35	0,3
Test Facility	0,225	0,246
Change Facility (Informational / Application)	0,993	0,992
Alignment between Business Architecture and Information Architecture	0,666(66,6%)	0,666 (66,6%)
Alignment between Business Architecture and Information Systems Architecture	0,666(66,6%)	0,666(66,6%)
Alignment between Information Architecture and Information Systems Architecture	0,714(71,4%)	0,714(71,4%)

From the Table 4.7, that presents the results of the evaluation on CS, we can conclude that the solution from the Reference Architecture presents the same alignment as the architecture from CS, a better test facility and worst change facility. The test facility result is better because the solution from the Reference Architecture provides less CRM modules and doesn't use the Quality Management system. The change facility is worse because of that same factor, the current state of CS architecture has more CRM modules and interacting systems, so is easier to change than an architecture with less systems and modules.

Observation

In this case study we identified several systems which interacted with the CRM system that weren't present in our architecture. The systems are: EasyVista CRM, SIGA system, IPContactos and Active Directory System. Also in the CRM system we identified a module customized for this case that wasn't covered in our Reference Architecture, the Cashier Module.

Regarding the information entities, in this case there were some information entities that weren't also covered by our architecture. The information entities related to the Cashier module weren't present and some information entities regarding the billing module, like IVA Code, Line Bill, Exemption Tax Code, Scale and Echelon weren't also covered, because in our architecture, the only information entity related to billing area, is the invoice information entity.

4.3 Results Analysis

After the evaluation done, we identified a pattern from Portuguese Public Administration from the CRM domain present in both cases evaluated that wasn't covered by our Reference Architecture. The pattern is regarding the SIGA system, a system responsible for ticket management, which appears in both case studies interacting with CRM. The interaction between that system and the CRM, involves in the two case studies two different modules. In the first case study the SIGA system interacts with the Scheduler module and in the second case study interacts with the Service

module. From this observation we can represent a pattern regarding CRM domain in Portuguese Public Administration that wasn't satisfied by our Reference Architecture and that can be added to it for improvement in this sector. The pattern is illustrated in Figure 4.16, and we nominated it the SIGA pattern.

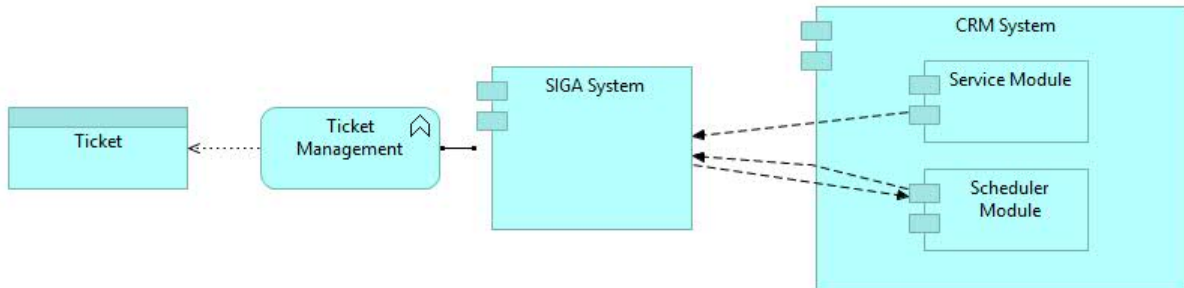


Figure 4.16: SIGA Pattern

Following we present a view with the SIGA pattern added to our Reference Architecture.

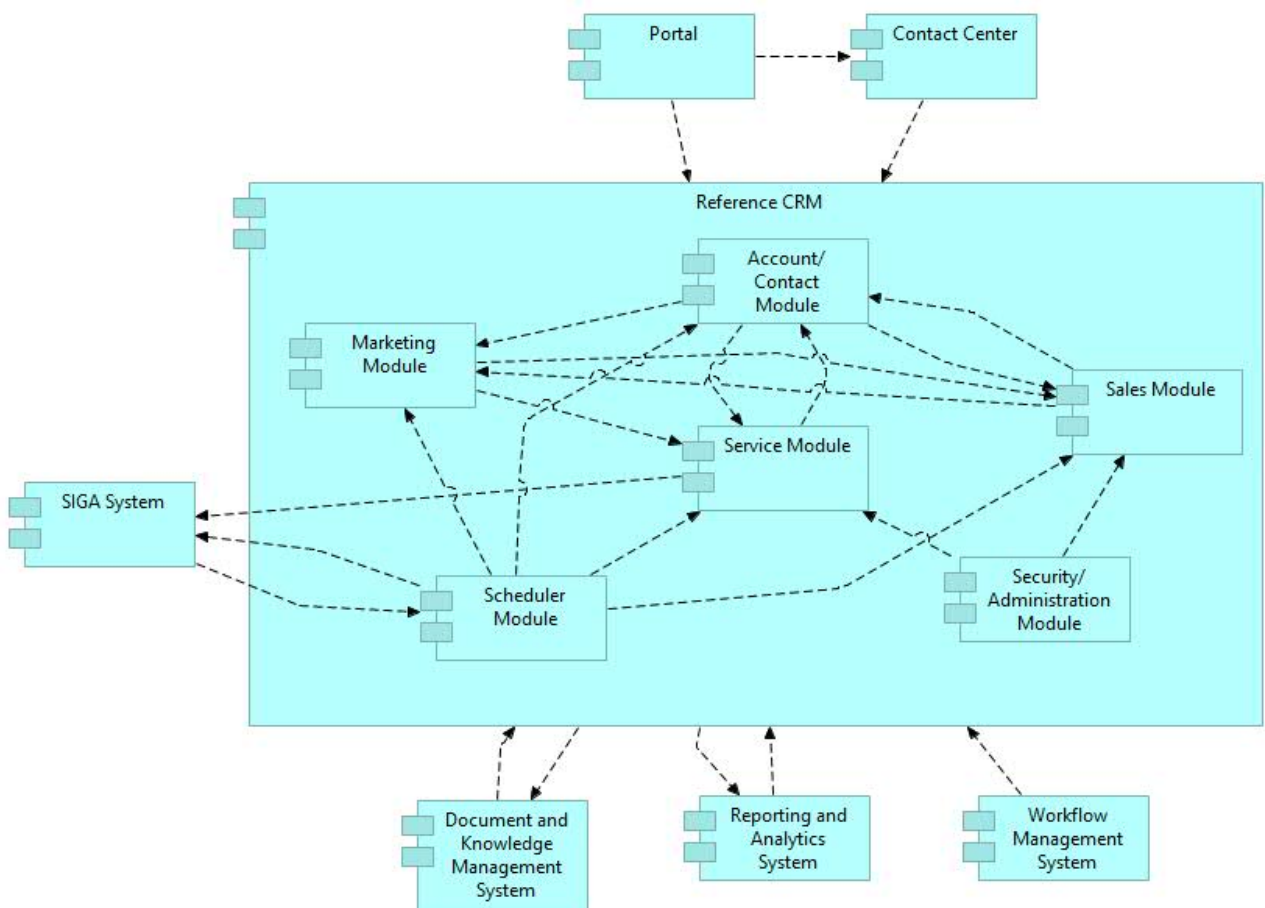


Figure 4.17: Reference Application Architecture with SIGA Pattern

The goal of the view in Figure 4.17 is to present a first view of the Reference Application Architecture adaptation to the Portuguese Public Administration, through the addition of the SIGA pattern.

With the evaluation done in the two case studies provided, we can now answer the questions from Section 1.2, regarding the problems that we proposed to solve with this work. There were three main investigation questions. The first one was:

- **Is it relevant to define a Reference Architecture for Customer Relationship Management considering industry best practices extracted from commercial solutions?**

This evaluation enabled us to verify that our Reference Architecture, defined from the best practices extracted from commercial solutions, satisfied most of the requirements in both case studies evaluated and presented some benefits and pitfalls in relation to what is the current Enterprise Architectures in CRM domain in these case studies.

The second main question was:

- **Are Reference Architecture for Customer Relationship Management useful for defining specific Enterprise Architectures for Customer Relationship Management domain?**

In the evaluation done one of the steps of the methodology for the evaluation of the case studies was to define a specific EA for CRM domain through the Reference Architecture guidance. With the methodology used for reaching a specific architecture and with the best practices from our Reference Architecture we demonstrated that our Reference Architecture was useful for the definition of specific architectures for specific cases, but has to take into account all the requirements of the case study. By last the third main question was:

- **Is the Reference Architecture adequate to be the Reference Architecture for the CRM domain for the Public Portuguese Administration?**

With the evaluation done we demonstrated that our Reference Architecture covered a big part of the requirements of each case, but it had always some components that weren't covered. With a continuous evaluation of more case studies, the Reference Architecture will be more adapted to the Public Portuguese Administration, because more patterns, like the identified SIGA pattern would be extracted and added to the Reference Architecture. So we can assert that the Reference Architecture showed promising results of being able to be adequate to the Public Portuguese Administration, but it requires more case studies to reach a level of being considered the Reference Architecture for CRM domain for the Public Portuguese Administration.

An aspect that we didn't consider, for each of the case studies evaluated, was the time and cost to apply the specific architecture based on the Reference Architecture. We assume that the changes that have to be done in order to change the current architecture to the architecture from the Reference Architecture would be very costly and have a high time-consuming.

Chapter 5

Conclusions

In this chapter, we present the conclusions taken from the work done. We explain all the contributions, the limitations of the solution and thoughts regarding what can be done for future work.

5.1 Research Questions Analysis

In this section, we make an analysis on the research issues of section 1.2. The research problems consisted of answering the three main questions and a set of four questions related to the definition of the Reference Architecture. We start by analyzing and answering to the set of four questions about the Reference Architecture.

The first question of the set of questions was: what are the principal features of the CRM systems? In this work we did an analysis on the datasheets from five CRM commercial solutions and through that analysis we identified fifty-eight features. From those fifty-eight features we considered fifty-two of them to be principal features of the CRM commercial solutions, because they were common to at least three of the five CRM solutions chosen to be analysed.

The second question was: what are the main information entities of the CRM systems? We identified through an analysis of the data models of the CRM commercial solutions and with the help of Buttle [2009] reference, fifty information entities which we considered as the main information entities of the CRM domain. With the main information entities we defined a Reference Information Architecture.

The third question regarding the Reference Architecture was: which patterns compose a Reference Architecture for the CRM domain? When we reached the Reference Architecture through the identification of the features and information entities, we defined eleven design patterns that together composed our Reference Architecture.

The fourth question was: what are the main Enterprise Architecture Principles in CRM architectures? After the definition of the Reference Architecture we used the Greefhorst and Proper [2011]

EA Principles list, and we verified which ones of them were satisfied by our Reference Architecture. We identified eleven Enterprise Architecture Principles satisfied by the Reference Architecture, and concluded that those were main Enterprise Architecture Principles for CRM Enterprise Architectures. Then we analyze and answer to the main questions of the research.

The first main question was: is it relevant to define a Reference Architecture for CRM considering industry best practices extracted from commercial solutions? This was the main goal of our work, we reached the Reference Architecture and applied it in two case studies. When applied to those case studies we verified that it was relevant, because our Reference Architecture satisfied most of the requirements of the case studies and provided a better test facility and in one of the case studies a better alignment. So in conclusion we consider that we have shown in this work that is relevant to define a Reference Architecture for the CRM domain considering industry best practices.

The second main question was: are Reference Architectures for CRM useful for defining specific Enterprise Architectures for CRM domain? In the evaluation made, one of the steps was to reach a specific architecture for the case study, based on the Reference Architecture. With a methodology for reaching specific architectures and with the best practices of our Reference Architecture we could always reach a specific architecture for each case study. So we can conclude that the Reference Architecture for CRM are useful defining specific Enterprise Architectures.

The third and final main question was: is the Reference Architecture adequate to be the Reference Architecture for the CRM domain for the Public Portuguese Administration? With the evaluation made, we have shown that our Reference Architecture fulfilled most of the requirements of each case, but there was always some components that were not satisfied. With a continuous evaluation of more case studies, the Reference Architecture will be more adapted to the Public Portuguese Administration, because more patterns, like the SIGA pattern, would be extracted and added to the Reference Architecture. So we can assert that the Reference Architecture showed promising results to be able to be adequate to the Portuguese Public Administration, but it required more case studies to achieve a level of being considered the Reference Architecture for the CRM domain for the Portuguese Public Administration. After this analysis on the questions of the research, in the next section we detail the major contributions of this work.

5.2 Major Contributions

The whole work done in this thesis was in order to provide a Reference Architecture for the CRM domain. Therefore the main contribution is the proposal of a Reference Application Architecture for the CRM domain. This Reference Architecture was defined based on the analysis done on five CRM commercial solutions. Through that analysis, we identified six CRM modules and five systems that interact with the CRM system. The six CRM modules are: Account module, Sales module, Marketing module, Service module, Administration module and Scheduler module. The

five systems that interact with the CRM are: Portal, Contact Center, Document and Knowledge Base Management system, Workflow Management system and Reporting and Analysis system. In order to construct this Reference Architecture, we used a methodology to reach this same Reference Architecture. This methodology is a cyclical work, with three steps but more steps can be added to it in future work. Those three steps are: the definition of the mission, vision and strategy, the definition of a first view of the Reference Architecture using only the CRM features and the definition of the final view of the Reference Architecture using the features and information entities mapped in a CRUD matrix.

Regarding the evaluation, we provide other contribution on the benefits and pitfalls of the proposed solution architecture. We evaluated our Reference Architecture in two case studies from Portuguese Public Administration: the High Commissioner for Migration and the Citizen Spaces. For each of these cases we applied a methodology for the evaluation of the specific quality attributes. This evaluation methodology, which we defined consists in two steps: in the first step we analyse the documents from the case study and with the information extracted from these documents we can define the current state Application Architecture and also, define an Application Architecture specific for the case study based on the best practices from our Reference Architecture. In the second step we apply the chosen metrics for information systems evaluation in both architectures and we make an analysis on the results obtained.

For each case we obtained a better test facility, a worst change facility and for the High Commissioner for Migration case we obtained also a better alignment between information and systems information. The reason for these results is due the fact that the solutions derived from our Reference Architecture have fewer CRM modules and also less systems that interact with the CRM, in relation to the current architectures of the provided case studies. These aspects result in a better test facility, because it requires less modules and systems to test, and a worst change facility because the more the processes are divided in various systems and modules, the easier is to make a change.

Also in the evaluation done we provide other contribution. We have identified a pattern from the CRM domain from the Portuguese Public Administration. The pattern consisted of the interaction between the CRM system and a system responsible for ticket management, the SIGA system. We named this pattern the SIGA pattern that is present in the two case studies used in this work. In conclusion, these were the major contributions in this thesis.

5.3 Accessory Contributions

In all the steps of the work, some minor contributions were accomplished, and its those contributions that we are going to detail next.

The analysis of the CRM solutions has allowed us to draw conclusions, about the offer of each solution in the different areas. We concluded that the SugarCRM is the most complete in terms of

Sales domain, to the Marketing domain the Microsoft Dynamics CRM stands out as the CRM solution which provides more features, in terms of Customer Service the most suitable CRM solution is the Salesforce CRM, in the Reporting and Analytics field all the CRM solutions are equivalent, in the Integration domain the SugarCRM solution is the solution with the most integration capacity and, for the last, in terms of Security the Salesforce CRM and Microsoft Dynamics CRM are the solutions with the most features.

Also regarding the analysis of the CRM systems we done an identification of the most common information entities for the CRM domain. We have analysed the data models from the five CRM systems and in that analysis we identified fifty common information entities. The identification of fifty common information entities allowed the definition of a Reference Information Architecture for CRM domain.

The defined Reference Architecture enabled the identification of eleven Enterprise Architecture Principles, which can be considered the main principles to follow to reach an architecture for the CRM domain.

The last contribution of our work was the analysis done in each case study. For each case study we did an identification and representation of the business processes, of the information entities and of the current state application architecture, which is relevant for the evaluated entities, because is knowledge and architectural models regarding their CRM domain.

5.4 Limitations

In this section we present the limitations of the architectural solution. One limitation identified at the evaluation was the fact that our Reference Architecture was developed from five CRM mid-market suites and not CRM commercial solutions specialized for public sector. This limitation implied that during the evaluation some aspects of the case studies weren't covered by our Reference Architecture.

We identified that our Reference Architecture didn't cover the following aspects in the case studies:

- the SIGA system present in two of the case studies;
- the EasyVista CRM, the Active Directory and iPCcontactos systems present in the Citizen Spaces;
- a set of information entities in each case study;
- the Cashier CRM module in the Citizen Spaces;

Another limitation is the few number of case studies where the Reference Architecture is evaluated. With more case studies, more validated would be the solution. In the next section we explain what can be done in future work.

5.5 Future Work

For future work the first idea is to continue to evaluate the Reference Architecture in more case studies from the Portuguese Public Administration, in particular, the case from Social Security, which is a complex case study that we couldn't get access in time.

The goal with the continuous evaluation in case studies, is to improve the Reference Architecture so that it can be more adapted to the Portuguese public sector. In the two case studies evaluated, we made an observation at the end of each of them to analyse what requirements weren't covered by our Reference Architecture. If in these observations a pattern was found, that pattern would be added to the Reference Architecture. With a greater number of case studies and their observations, more patterns would be found and would be added to the Reference Architecture. This would result in a Reference Architecture more adequate to the Portuguese public sector.

Another idea for future work is the definition of a Reference Architecture for the CRM domain, specialized in evaluating architectures in CRM domain. The goal with definition of a Reference Architecture as described, is to provide grades to Enterprise Architectures in the CRM domain, to verify if these architectures follow the best practices in the CRM domain. Note that the Reference Architecture in this work intended to be a guide for the definition of Enterprise Architectures for specific cases in the CRM domain and it isn't focused on evaluating other architectures. The evaluation done in this work is to prove that our Reference Architecture based on CRM commercial solutions offer benefits and pitfalls in relation to other architectures implemented on the field and that satisfies the requirements of those cases. These are the ideas for the future work regarding the theme approached in this thesis.

References

- AMA (2014a). *Cláusulas Técnicas Balcão Multiserviços*. Agência para a Modernização Administrativa, Rua Abrantes Ferrão 10 3ºG, 1600-001 Lisboa, Portugal.
- AMA (2014b). *Cláusulas Técnicas NOVO CENTRO DE CONTACTO DO ESPAÇO DO CIDADÃO*. Agência para a Modernização Administrativa, Rua Abrantes Ferrão 10 3ºG, 1600-001 Lisboa, Portugal.
- AMA (2014c). *Espaço do Cidadão Mediadores de Cidadania - Documento de Desenho Funcional e Técnico*. Agência para a Modernização Administrativa, Rua Abrantes Ferrão 10 3ºG, 1600-001 Lisboa, Portugal.
- AMA (2014d). *Espaço do Cidadão Mediadores de Cidadania - Documento de Especificação de Requisitos*. Agência para a Modernização Administrativa, Rua Abrantes Ferrão 10 3ºG, 1600-001 Lisboa, Portugal.
- Barrish, J. (2014). Top crm software. <http://www.capterra.com/customer-relationship-management-software/#infographic>.
- Baskerville, R. L. (1999). Investigating information systems with action research. *Communication of the AIS*, 2(3es).
- Bauer, M. (2012). Internet-of-things architecture. Technical report, European Commission with the Seventh Framework Programme.
- Buttle, F. (2009). *Customer relationship management : concepts and technologies*. Butterworth-Heinemann Ltd. ISBN 9781856175227.
- Cloutier, R., Muller, G., Verma, D., Nilchiani, R., Hole, E., and Bone, M. (2010). The concept of reference architectures. *Syst. Eng.*, 13(1):14–27. ISSN 1098-1241.
- Cruz, A. (2015). Crm analysis on features and data models. Technical report, Instituto Superior Técnico. <http://pt.slideshare.net/AndrCruz14/technical-report-45168794>.
- Cruz, A. and Vasconcelos, A. (2015). Towards a reference enterprise application architecture for the crm domain. *17th International Conference on Enterprise Information Systems*.
- Dietz, J. L. G. (2007). Building strategy into design, the Hague: Academic service.

- Fardoie, S. R. and Monfared, M. (2008). A new design architecture for e-crm systems (case study: tour package choice in tourism industry). *Management of Innovation and Technology, 2008. ICMIT 2008. 4th IEEE International Conference*, pages 463–468. ISBN 978-1-4244-2329-3.
- Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns: Elements of Reusable Object-oriented Software*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Gilchrist, R. and Hariharan, M. (2009). The microsoft dynamics crm security model. Technical report, Microsoft.
- Greefhorst, D. and Proper, E. (2011). *Architecture Principles: The Cornerstones of Enterprise Architecture (The Enterprise Engineering Series)*. Springer. ISBN 3642202780.
- Haren, V. (2009). *ArchiMate 1.0 Specification*. The Open Group. Van Haren Publishing.
- Haren, V. (2012). *ArchiMate 2.0 Specification*. The Open Group. Van Haren Publishing.
- Hoogervorst, J. (2009). Enterprise governance and enterprise engineering. *MIS Quarterly: Management Information Systems*.
- Hoogervorst, J. A. P. (2004). Enterprise architecture: Enabling integration, agility and change. *International Journal of Cooperative Information Systems*, 13:213–233.
- IEEE (2000). IEEE Recommended Practice for Architectural Description of Software-Intensive Systems. *IEEE Std 1471-2000*.
- ISO (2001). Software engineering - product quality, ISO/IEC 9126-1. Technical report, International Organization for Standardization.
- Janjicek, R. (2005). Crm architecture for enterprise relationship marketing in the new millenium. Technical report.
- Jin, M., Kung, D., and Peng, W. (2010). Research of information system technology architecture. *Industrial and Information Systems (IIS), 2010 2nd International Conference*, 2:293–296. ISBN 978-1-4244-7860-6.
- Lankhorst, M. (2005). *Enterprise Architecture at Work: Modelling, Communication and Analysis*. Springer. ISBN 3540243712.
- Lapkin, A. (2008). Gartner clarifies the definition of the term 'enterprise architecture'.
- Laudon, K. C. and Laudon, J. P. (2012). *Management Information Systems: Managing the Digital Firm*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 12th edition. ISBN 9780132142854.
- Lindström, . (2006). On the syntax and semantics of architectural principles. *39th Annual Hawaii International Conference on Systems Sciences*.

- Microsoft (2006). Microsoft dynamics customer model departments work poster. https://fenix.tecnico.ulisboa.pt/downloadFile/3779580626053/Microsoft%20Dynamics_CustomerModel_Departments_Work_Poster_LoRes.pdf.
- Microsoft (2008a). <http://www.crmssoftwareblog.com/software/>.
- Microsoft (2008b). Customer service. http://cdn.crmssoftwareblog.com/wp-content/uploads/Customer_Service_Brochure.pdf.
- Microsoft (2008c). Marketing automation. http://cdn.crmssoftwareblog.com/wp-content/uploads/Marketing_Automation_Brochure.pdf.
- Microsoft (2008d). Sales force automation. http://cdn.crmssoftwareblog.com/wp-content/uploads/Sales_Automation_Brochure.pdf.
- MIT (2001). Mit process handbook. <http://ccs.mit.edu/ph/>.
- Muller, G. and Hole, E. (2007). Reference architectures; why, what and how. White paper, Embedded Systems Institute and Stevens Institute of Technology.
- op't Land, M., Waage, M., Cloo, J., and Steghuis, C. (2009). Enterprise Architecture – Creating Value by Informed Governance.
- Oracle (2007a). Oracle crm software. http://promero.com/siebel_crm_overview.asp.
- Oracle (2007b). Siebel crm on demand marketing. http://promero.com/assets/pdf/crm_ondemand_marketing_feature_sheet.pdf.
- Oracle (2007c). Siebel crm on demand sales. http://promero.com/assets/pdf/crm_ondemand_sales_feature_sheet.pdf.
- Oracle (2007d). Siebel crm on demand service. http://promero.com/assets/pdf/crm_ondemand_service_feature_sheet.pdf.
- Oracle (2011). Siebel security guide. http://docs.oracle.com/cd/B40099_02/books/PDF/Secur.pdf.
- Pereira, C. M. and Sousa, P. (2003). Getting into the misalignment between business and information systems. *10th European Conference on Information Technology Evaluation*, ECITE Press:499–511.
- Pereira, C. M. and Sousa, P. (2005). Enterprise architecture: Business and it alignment. *Symposium on Applied Computing*.
- Richardson, G., Jackson, B., and Dickson, G. (1990). A principle-based enterprise architecture: Lessons from texaco and star enterprise. *MIS Quarterly: Management Information Systems*, 14:285–403.

- Ruivo, J. M. P. (2012). *Usabilidade e Desenho de interfaces para a Aplicação de Gestão de Atendimento e CRM*. log, Calçada do Marquês de Abrantes 45 3º Dto, 1200-718 Lisboa, Portugal.
- Sage (2012). Sage crm professional. http://www.source1consultants.com/images/SageCRM_Professional.pdf.
- Salesforce (2000). <http://www.salesforce.com/eu/?ir=1>.
- Salesforce (2012). Selecting the right salesforce edition. <http://www.slideshare.net/rolandgraham/compare-edition-datasheet>.
- Spewak, S. H. and Hill, S. C. (1993). *Enterprise Architecture Planning: Developing a Blueprint for Data, Applications and Technology*. QED Information Sciences, Inc., Wellesley, MA, USA. ISBN 0-89435-436-1, 9780471599852.
- Stelzer, D. (2009). Enterprise architecture principles: Literature review and research directions. *ICSOC/ServiceWave 2009 International Workshops*, pages 23–27.
- SugarCRM (2004). <http://www.sugarcrm.com/>.
- SugarCRM (2014). Editions comparison chart. <http://e2benterprise.com/docs/sugareditionscomparison.pdf>.
- The Open Group (2009). TOGAF version 9.1 "Enterprise Edition".
- Vasconcelos, A. (2007). *Arquitecturas dos sistemas de informação: Representação e avaliação*. Phd thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa.
- Vasconcelos, A., Pereira, C. M., Sousa, P., and Tribolet, J. (2005). Open issues on information system architecture research domain: the vision.
- Vasconcelos, A., Sousa, P., and Tribolet, J. (2008). Enterprise architecture analysis: An information system evaluation approach. *International Journal of Enterprise Modelling and Information Systems Architectures*, 3(2):31–53.
- Winter, R. and Aier, S. (2011). How are enterprise architecture design principles used? *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*, pages 314–321. ISBN 978-1-4577-0869-5.
- Winter, R. and Fischer, R. (2010). Essential layers, artifacts, and dependencies of enterprise architecture. *Enterprise Distributed Object Computing Conference Workshops, 2006. EDOCW '06 10th IEEE International*, 2:30. ISBN 0-7695-2743-4.

Appendix A

ACM Business Processes



Figure A.1: ACM CRM Customer Management Processes

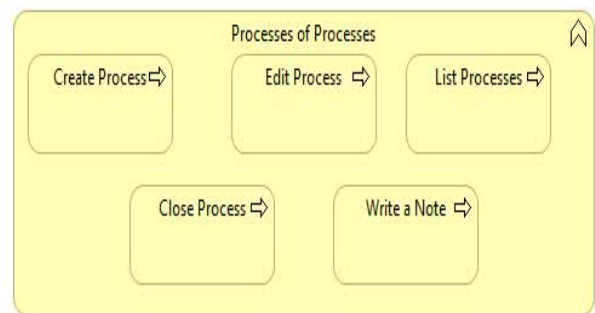


Figure A.2: ACM CRM Processes Management Processes

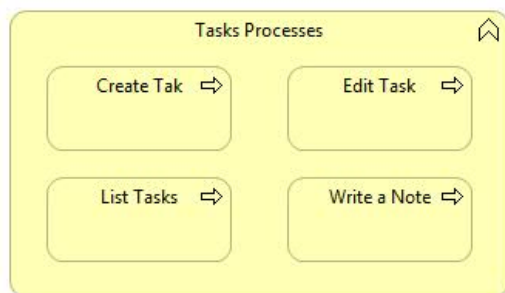


Figure A.3: ACM CRM Task Management Processes



Figure A.4: ACM CRM Follow Up Management Processes

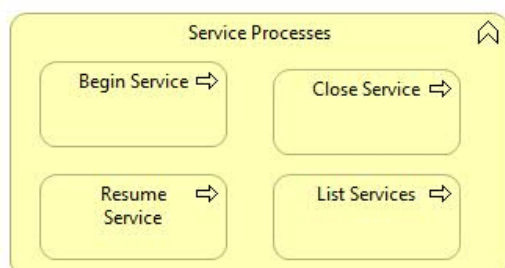


Figure A.5: ACM CRM Service Management Processes

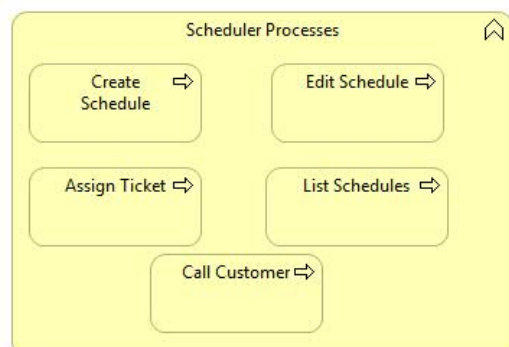


Figure A.6: ACM CRM Service Management Processes

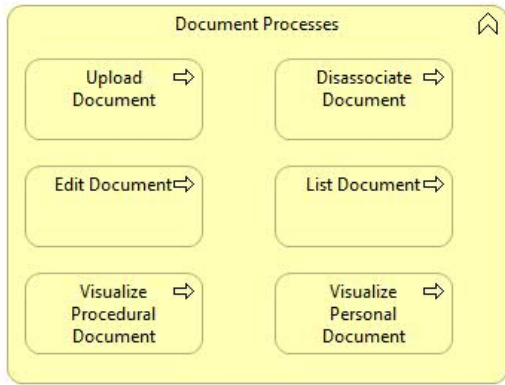


Figure A.7: ACM CRM Document Management Processes



Figure A.8: ACM CRM Administration Management Processes

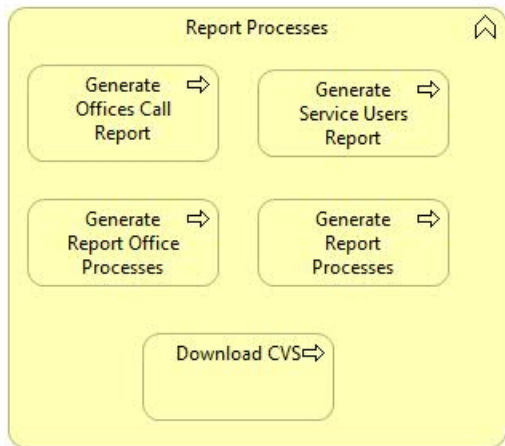


Figure A.9: ACM CRM Report Management Processes

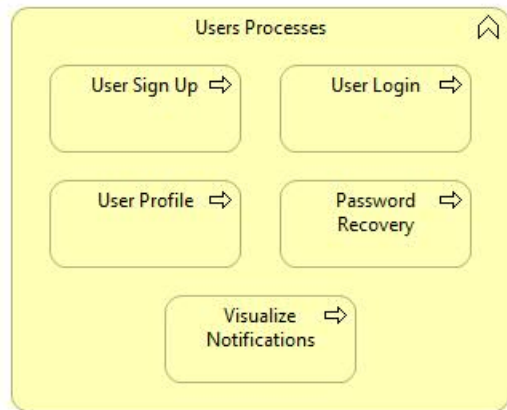


Figure A.10: ACM CRM User Management Processes

Appendix B

Information Entities Clustering

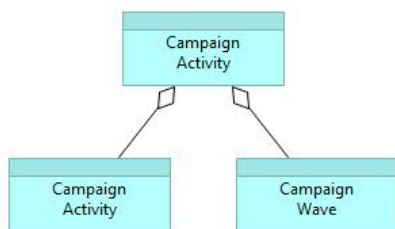


Figure B.1: Campaign Activity information entity clustering

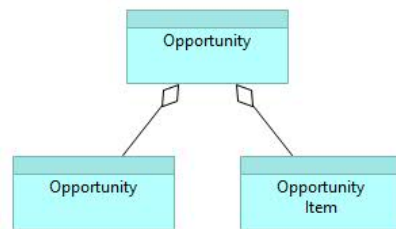


Figure B.2: Opportunity information entity clustering

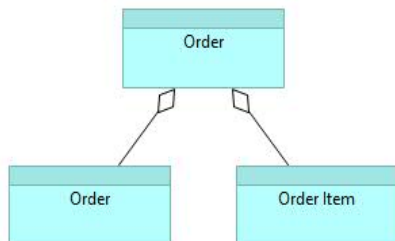


Figure B.3: Order information entity clustering

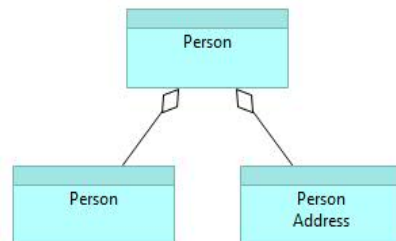


Figure B.4: Person information entity clustering

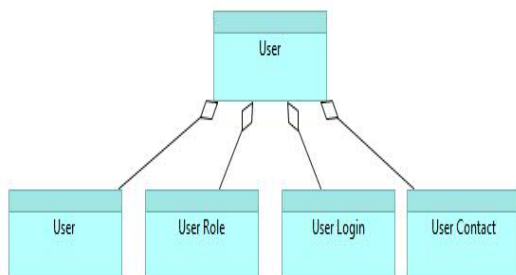


Figure B.5: User information entity clustering

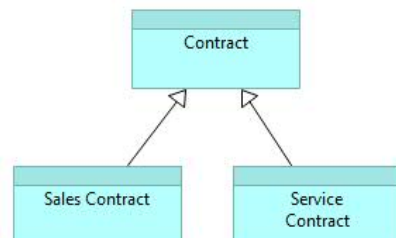


Figure B.6: Contract information entity division

Appendix C

Case Studies Services Realization

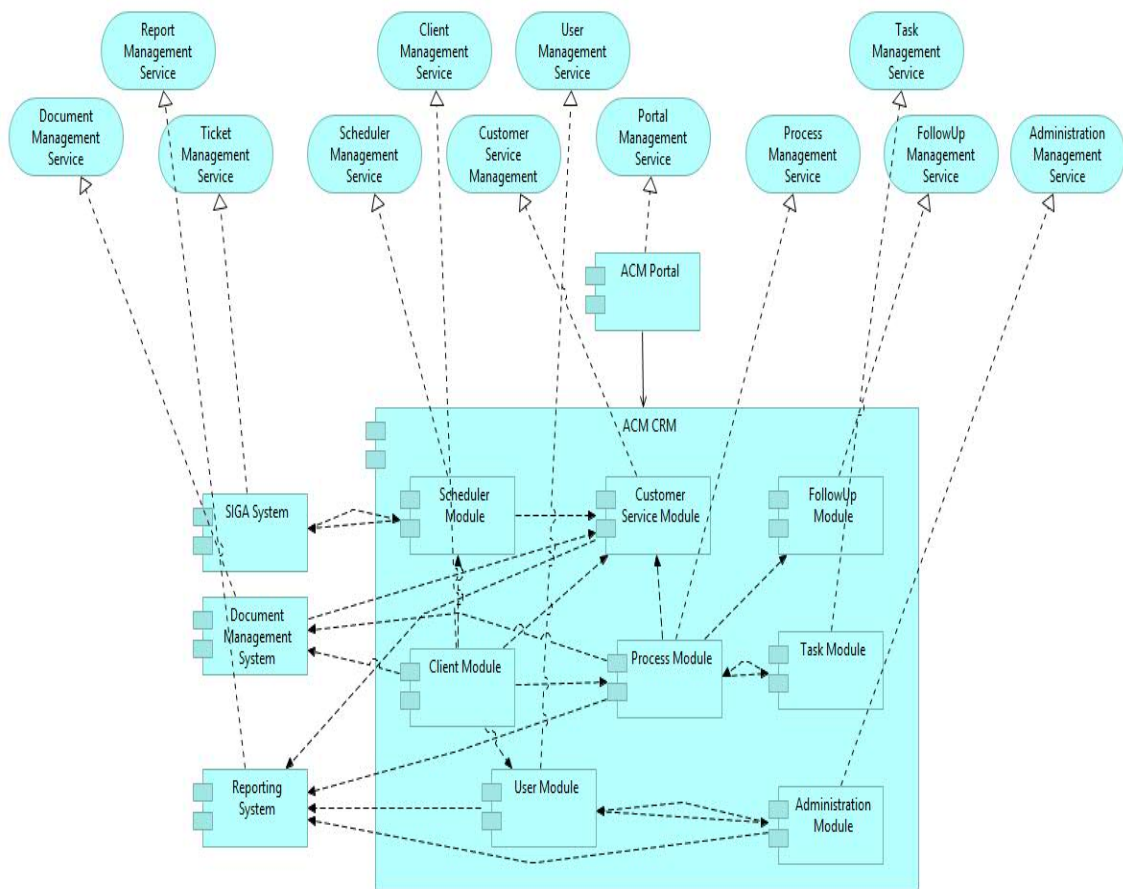


Figure C.1: ACM Services Realization in Current State

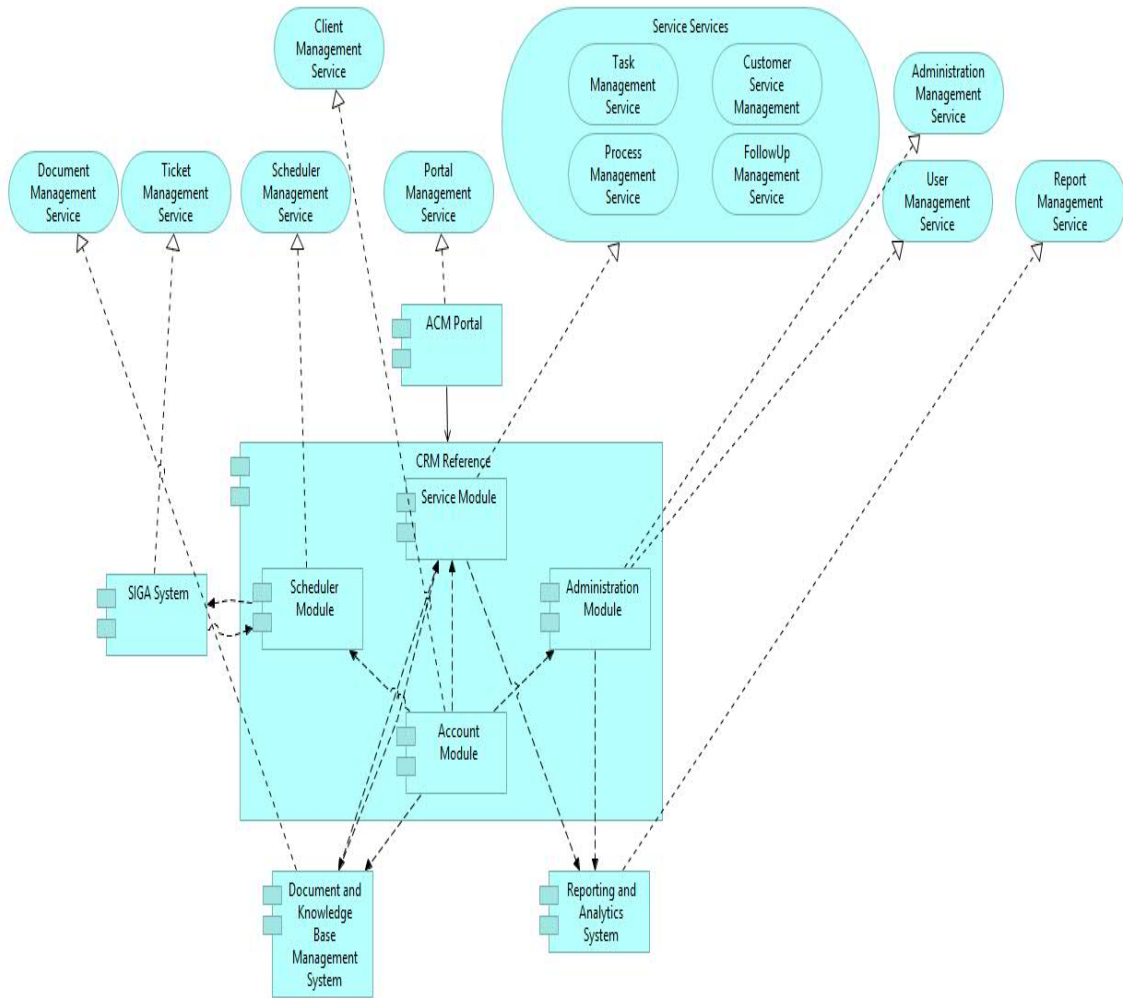


Figure C.2: ACM Services Realization by architecture from Ref. Architecture

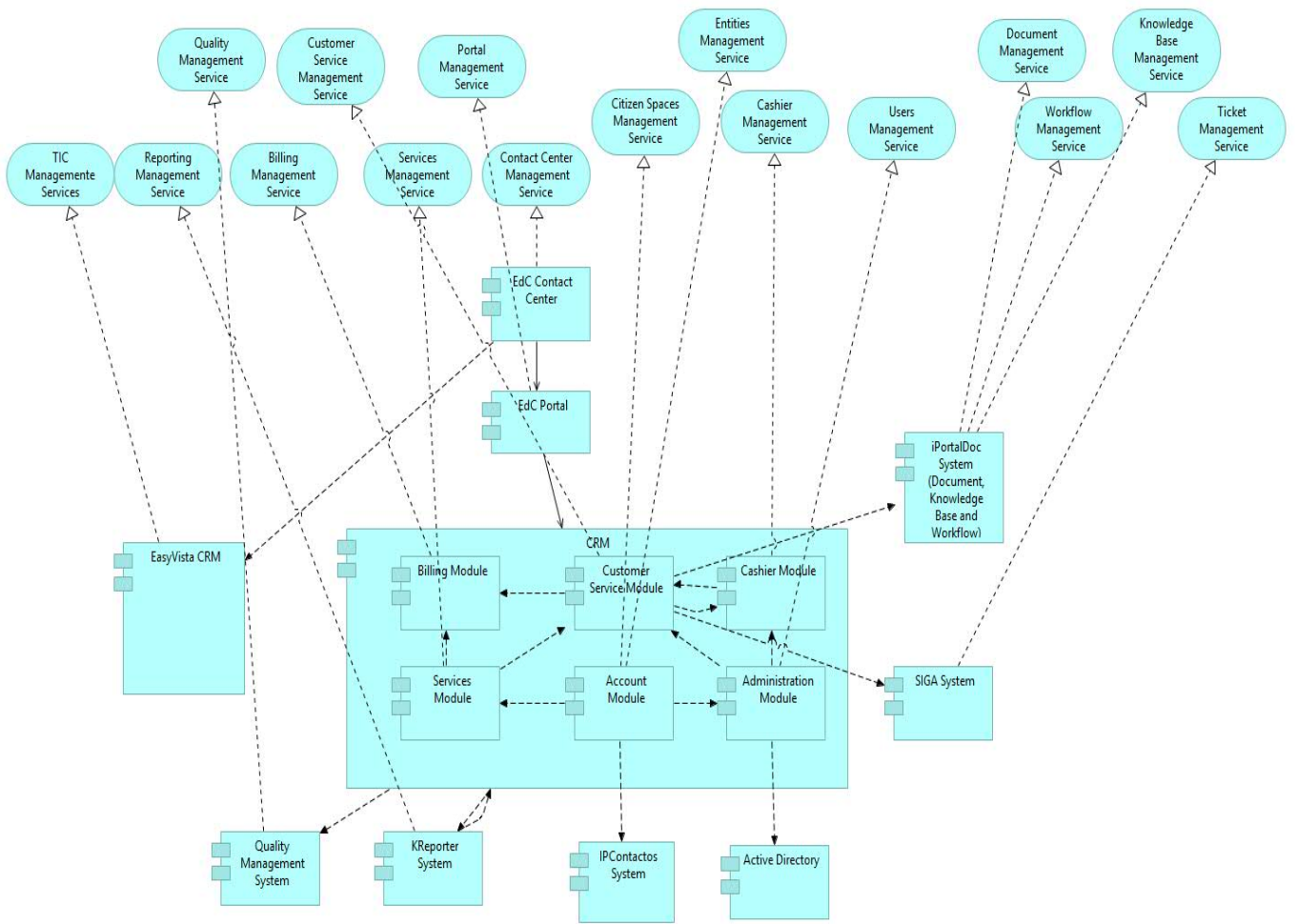


Figure C.3: EdC Services Realization in Current State

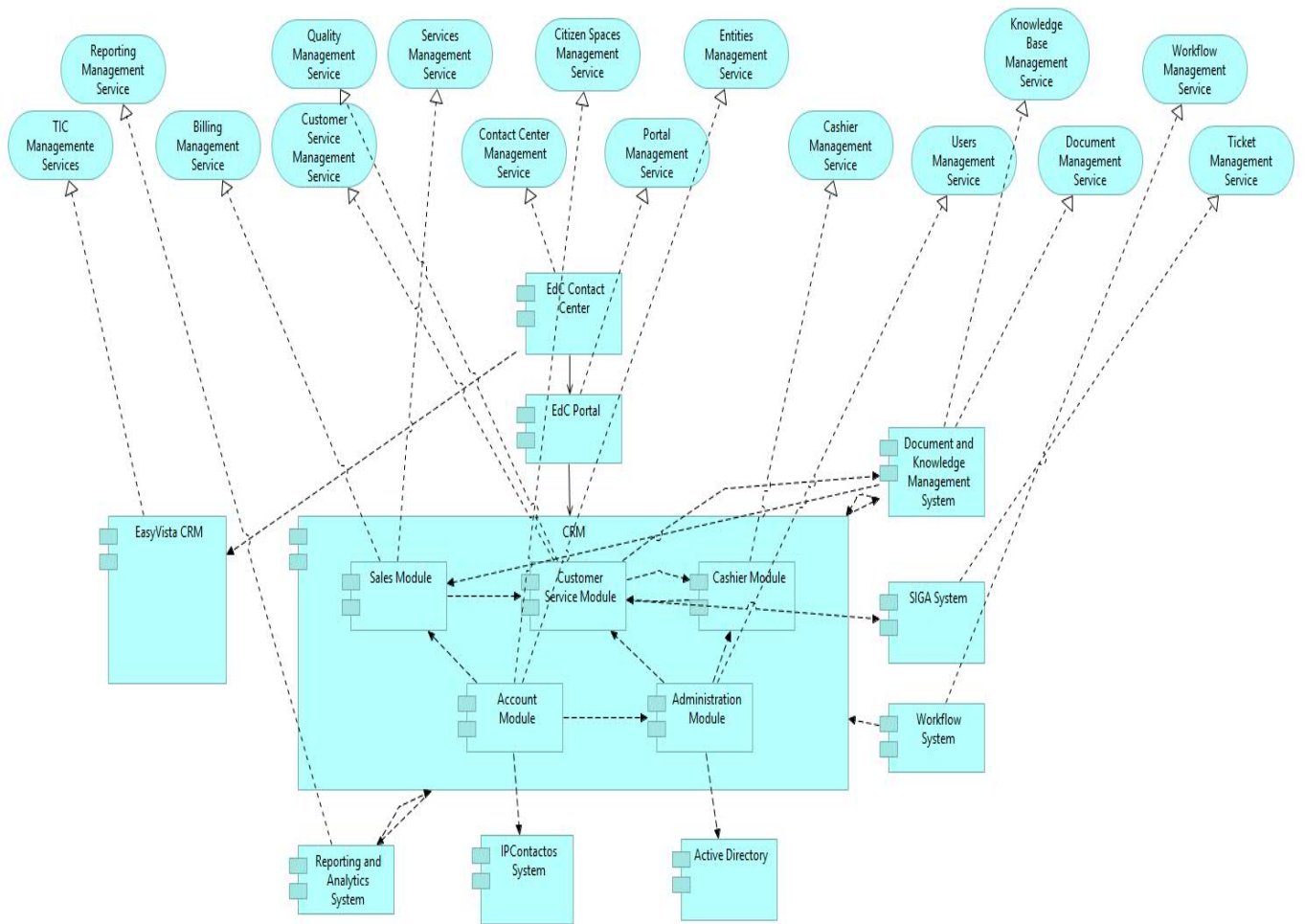


Figure C.4: EdC Services Realization by architecture from Ref. Architecture

Appendix D

Customer Relationship Management Data Models

D.1 Buttle Components

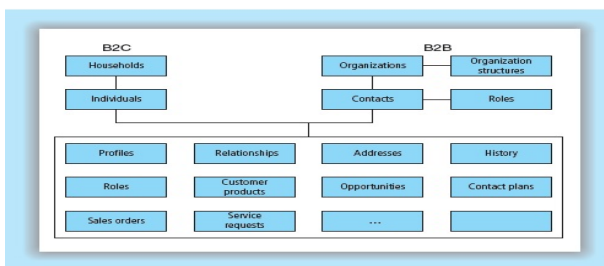


Figure D.1: CRM customer components from Buttle [2009]

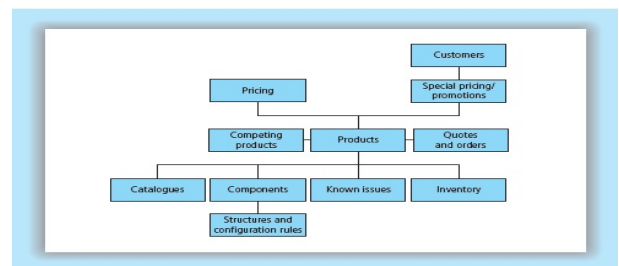


Figure D.2: CRM products components from Buttle [2009]

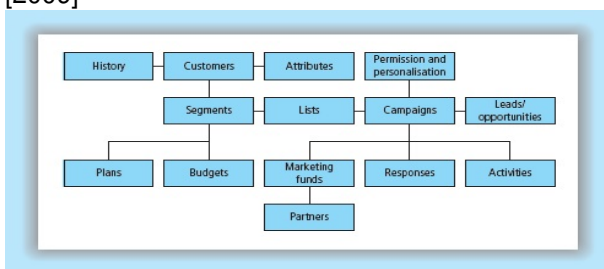


Figure D.3: CRM marketing automation components from Buttle [2009]

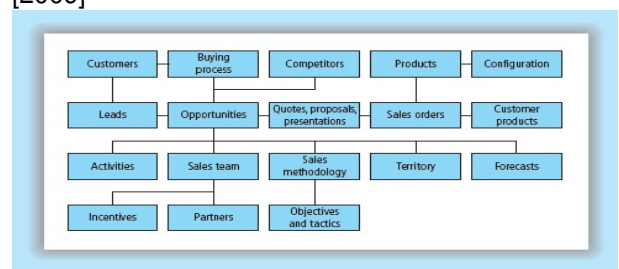


Figure D.4: CRM sales-force automation components from Buttle [2009]

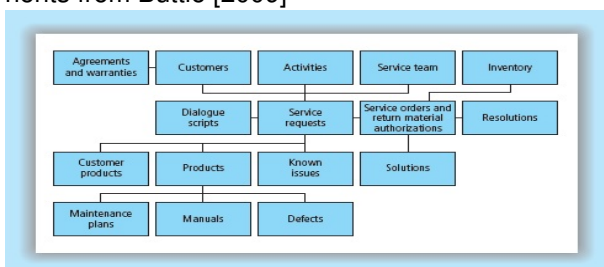


Figure D.5: CRM service automation components from Buttle [2009]

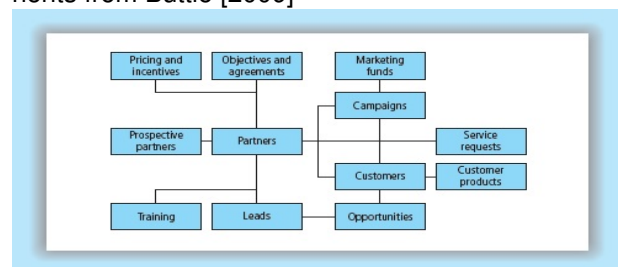


Figure D.6: CRM partner relationship management components from Buttle [2009]

D.2 CRM Solutions Information Entities Mapped

Table D.1: Salesforce Information Entities Mapped - Part 1

Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
AcceptedEventRelation				
Account	X	X	X	X
AccountContactRole				
AccountFeed				
AccountHistory				
AccountOwnerSharingRule				
AccountPartner				
AccountShare				
AccountTag				
AccountTeamMember				X
AccountTerritoryAssignmentRule				
AccountTerritoryAssignmentRuleItem				
AccountTerritorySharingRule				
ActivityHistory				
AdditionalNumber				
AllowedEmailDomain				
ApexClass				
ApexComponent				
ApexLog				
ApexPage				
ApexTestQueueItem				
ApexTestResult				
ApexTrigger				
Approval				
ArticleTypeDataCategorySelection				
Asset				X
AssetFeed				
AssetTag				
AssignmentRule				
AsyncApexJob				
AttachedContentDocument				
Attachment				
AuraDefinition				
AuraDefinitionBundle				
AuthConfig				
AuthConfigProviders				
AuthProvider				
AuthSession				
Bookmark				
BrandTemplate				
BusinessHours				
BusinessProcess				
CallCenter				
Campaign	X	X		X
CampaignFeed	X			
CampaignMember				
CampaignMemberStatus				
CampaignOwnerSharingRule				
CampaignShare				
CampaignTag				
Case	X	X	X	
CaseArticle				
CaseComment				
CaseContactRole				
CaseFeed				
CaseHistory				
CaseMilestone				
CaseOwnerSharingRule				
CaseShare				
CaseSolution		X		
CaseStatus				
CaseTag				
CaseTeamMember				
CaseTeamRole				
CaseTeamTemplate				
CaseTeamTemplateMember				
CaseTeamTemplateRecord				

Table D.2: Salesforce Information Entities Mapped - Part 2

Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
CategoryData				
CategoryNode				
CategoryNodeLocalization				
ChatterActivity				
ChatterAnswersActivity				
ChatterAnswersReputationLevel				
ChatterConversation				
ChatterConversationMember				
ChatterMessage				
CollaborationGroup				
CollaborationGroupFeed				
CollaborationGroupMember				
CollaborationGroupMemberRequest				
CollaborationInvitation				
CombinedAttachment				
Community				
Contact	X	X		X
ContactFeed				
ContactHistory				
ContactOwnerSharingRule				
ContactShare				
ContactTag				
ContentDocument				
ContentDocumentFeed				
ContentDocumentHistory				
ContentDocumentLink				
ContentVersion				
ContentVersionHistory				
ContentWorkspace				
ContentWorkspaceDoc				
Contract	X	X		X
ContractContactRole				
ContractFeed				
ContractHistory				
ContractLineItem				
ContractLineItemHistory				
ContractStatus				
ContractTag				
CronTrigger				
CronJobDetail				
CurrencyType				
CustomBrand				
CustomBrandAsset				
CustomObjectFeed				
CustomPermission				
CustomPermissionDependency				
DandBCompany				
Dashboard		X		
DashboardComponent				
DashboardComponentFeed				
DashboardFeed				
DashboardTag				
DatacloudCompany				
DatacloudContact				
DatacloudDandBCompany				
DatacloudOwnedEntity				
DatacloudPurchaseUsage				
DatacloudSocialHandle				
DatedConversionRate				
DcSocialProfile				
DcSocialProfileHandle				
DeclinedEventRelation				
Division				
DivisionLocalization				
Document	X	X		
DocumentAttachmentMap				
DocumentTag				
DuplicateRecordItem				
DuplicateRecordSet				

Table D.3: Salesforce Information Entities Mapped - Part 3

Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
EmailMessage	X	X	X	
EmailServicesAddress		X		
EmailServicesFunction				
EmailStatus				
EmailTemplate		X		
Entitlement				X
EntitlementContact				
EntitlementFeed				
EntitlementHistory				
EntitlementTemplate				
EntityHistory				
EntitySubscription				
EnvironmentHubMember				
Event				X
EventFeed				
EventRelation				
EventTag				
EventWhoRelation				
ExternalDataSource				
ExternalDataUserAuth				
FeedComment				
FeedItem				
FeedLike				
FeedPollChoice				
FeedPollVote				
FeedPost				
FeedTrackedChange				
FieldPermissions				
FiscalYearSettings				
Folder				
ForecastingAdjustment				
ForecastingFact				
ForecastingItem		X	X	X
ForecastingQuota				
ForecastShare				
Group				X
GroupMember				
HashtagDefinition				
Holiday				
Idea				
IdeaComment				
IdeaTheme				
KnowledgeableUser				
KnowledgeArticle	X	X		
KnowledgeArticleVersion				
KnowledgeArticleVersionHistory				
KnowledgeArticleViewStat				
KnowledgeArticleVoteStat				
Lead	X	X	X	X
LeadFeed				
LeadHistory				
LeadOwnerSharingRule				
LeadShare				
LeadStatus				
LeadTag				
LimitAllocationPerApp				
LineitemOverride				
LoginHistory				
LookedUpFromActivity				
MailmergeTemplate				
MilestoneType				
Name				
Network				
NetworkActivityAudit				
NetworkMember				
NetworkModeration				
NewsFeed				

Table D.4: Salesforce Information Entities Mapped - Part 4

Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
Note				
NoteAndAttachment				
NoteTag				
OauthToken				
ObjectPermissions				
ObjectTerritory2AssignmentRule				
ObjectTerritory2AssignmentRuleItem				
ObjectTerritory2Association				
OpenActivity				
Opportunity	X	X	X	X
OpportunityCompetitor	X			
OpportunityContactRole		X		X
OpportunityFeed				
OpportunityFieldHistory				
OpportunityHistory			X	
OpportunityLineItem	X		X	X
OpportunityLineItemSchedule				
OpportunityOverride				
OpportunityOwnerSharingRule				
OpportunityPartner				
OpportunityShare				
OpportunitySplit				
OpportunitySplitType				
OpportunityStage				
OpportunityTag				
OpportunityTeamMember				
Order	X		X	X
OrderFeed				
OrderHistory				
OrderItem			X	X
OrderItemFeed				
OrderItemHistory				
Organization	X		X	
OrgWideEmailAddress				
OwnedContentDocument				
PackageLicense				
Partner				X
PartnerNetworkConnection				
PartnerNetworkRecordConnection				
PartnerRole				
Period				X
PermissionSet				
PermissionSetAssignment				
Pricebook2	X			X
Pricebook2History				
PricebookEntry				
ProcessDefinition				
ProcessInstance				
ProcessInstanceHistory				
ProcessInstanceNode				
ProcessInstanceStep				
ProcessInstanceWorkitem				
ProcessNode				
Product2	X	X	X	X
Product2Feed				
ProductEntitlementTemplate				
Profile				
ProfileSkill				
ProfileSkillEndorsement				
ProfileSkillEndorsementHistory				
ProfileSkillFeed				
ProfileSkillHistory				
ProfileSkillShare				
ProfileSkillUser				
ProfileSkillUserHistory				
PushTopic				

Table D.5: Salesforce Information Entities Mapped - Part 5

Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
QuantityForecast				
QuantityForecastHistory				
Question				
QuestionDataCategorySelection				
QuestionReportAbuse				
QuestionSubscription				
QueueSubject				
Quote	X	X	X	X
QuoteDocument				
QuoteLineItem				
RecentlyViewed				
RecordType				
RecordTypeLocalization				
Reply				
ReplyReportAbuse				
Report		X		
ReportFeed				
ReportTag				
ReputationLevel				
ReputationPointsRule				
RevenueForecast				
RevenueForecastHistory				
RuleTerritory2Association				
SamlSsoConfig				
SearchPromotionRule				
Scontrol				
ScontrolLocalization				
SelfServiceUser				
ServiceContract				
ServiceContractFeed				
ServiceContractHistory				
ServiceContractOwnerSharingRule				
ServiceContractShare				
SetupEntityAccess				
SignupRequest				
Site	X			
SiteHistory				
SlaProcess			X	
Solution				
SolutionFeed				
SolutionHistory				
SolutionStatus				
SolutionTag				
StaticResource				
StreamingChannel				
TagDefinition				
Task	X	X		X
TaskFeed				
TaskPriority				
TaskRelation				
TaskStatus				
TaskTag				
TaskWhoRelation				
Territory	X			X
Territory2				
Territory2Model				
Territory2ModelHistory				
Territory2Type				
ThirdPartyAccountLink				
Topic				
TopicAssignment				
TopicFeed				
TwoFactorInfo				
UndecidedEventRelation				
User	X	X	X	X
UserAccountTeamMember				
UserConfigTransferButton				
UserConfigTransferSkill				
UserFeed				
UserLicense				
UserLogin		X		
UserMembershipSharingRule				
UserPackageLicense				
UserPreference				

Table D.6: Salesforce Information Entities Mapped - Part 6

Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
UserProfile				
UserProfileFeed				
UserRecordAccess				
UserRole				
UserShare				
UserTeamMember				
UserTerritory				
UserTerritory2Association				
Vote				
WebLink				
WebLinkLocalization				
WorkAccess				
WorkAccessShare				
WorkBadge				
WorkBadgeDefinition				
WorkBadgeDefinitionHistory				
WorkBadgeDefinitionShare				
WorkCoaching				
WorkCoachingFeed				
WorkCoachingHistory				
WorkCoachingShare				
WorkFeedback				
WorkFeedbackHistory				
WorkFeedbackQuestion				
WorkFeedbackQuestionHistory				
WorkFeedbackQuestionSet				
WorkFeedbackQuestionSetHistory				
WorkFeedbackQuestionSetShare				
WorkFeedbackQuestionShare				
WorkFeedbackRequest				
WorkFeedbackRequestFeed				
WorkFeedbackRequestHistory				
WorkFeedbackRequestShare				
WorkFeedbackShare				
WorkGoal				
WorkGoalCollaborator				
WorkGoalCollaboratorHistory				
WorkGoalFeed				
WorkGoalHistory				
WorkGoalLink				
WorkGoalShare				
WorkPerformanceCycle				
WorkPerformanceCycleFeed				
WorkPerformanceCycleHistory				
WorkPerformanceCycleShare				
WorkReward				
WorkRewardFund				
WorkRewardFundHistory				
WorkRewardFundShare				
WorkRewardFundType				
WorkRewardFundTypeHistory				
WorkRewardFundTypeShare				
WorkRewardHistory				
WorkRewardShare				
WorkThanks				
WorkThanksShare				

Table D.7: Microsoft Information Entities Mapped - Part 1

Microsoft	SugarCRM	SageCRM	Oracle Siebel
Account	X	X	X
AccountLeads			
ActivityMimeAttachment			
ActivityParty	X		
ActivityPartyRollupByAccount			
ActivityPartyRollupByContact			
ActivityPointer			
Annotation			
AnnualFiscalCalendar			
Appointment			
AsyncOperation			
AttributeMap			
BulkOperation			
BulkOperationLog			
BusinessUnit			X
BusinessUnitMap			
BusinessUnitNewsArticle			
Calendar		X	
CalendarRule			
Campaign	X	X	X
CampaignActivity		X	X
CampaignActivityItem			
CampaignItem			
CampaignResponse			
ColumnMapping			
Commitment			
Competitor			X
CompetitorAddress			
CompetitorProduct			
CompetitorSalesLiterature			
ConstraintBasedGroup			
Contact	X		X
ContactInvoices			
ContactLeads			
ContactOrders			
ContactQueues			
Contract	X		X
ContractDetail			
ContractTemplate			
CustomerAddress			
CustomerOpportunityRole			
CustomerRelationship			
Discount			
DiscountType			
DocumentIndex			
DuplicateRecord			

Table D.8: Microsoft Information Entities Mapped - Part 2

Microsoft	SugarCRM	SageCRM	Oracle Siebel
Email	X	X	
EntityMap			
Equipment			
Fax			
FilterTemplate			
FixedMonthlyFiscalCalendar			
Import			
ImportFile			
ImportMap			
Incident	X	X	
IncidentResolution	X		
IntegrationStatus			
InternalAddress			
Invoice			X
InvoiceDetail			
KbArticle	X		
KbArticleComment			
KbArticleTemplate			
Lead	X	X	X
LeadAddress			
LeadCompetitors			
LeadProduct			
Letter			
License			
List			
ListMember			
LookUpMapping			
MailMergeTemplate			
MonthlyFiscalCalendar			
Opportunity	X	X	X
OpportunityClose			
OpportunityCompetitors			
OpportunityProduct		X	
OrderClose		X	X
Organization		X	X
OrganizationUI			
OwnerMapping			
PhoneCall	X	X	
PickListMapping			
PluginType			
PluginAssembly			
PriceLevel			X
PrincipalObjectAccess			
Privilege			
PrivilegeObjectTypeCodes			
Product	X	X	X
ProductAssociation			
ProductPriceLevel			
ProductSalesLiterature			
ProductSubstitute			
QuarterlyFiscalCalendar			
Queue			
QueueItem			
Quote	X	X	X
QuoteClose			
QuoteDetail			
RelationshipRole			
RelationshipRoleMap			
Resource			
ResourceGroup			
ResourceSpec			
Role	X		

Table D.9: Microsoft Information Entities Mapped - Part 3

Microsoft	SugarCRM	SageCRM	Oracle Siebel
RolePrivileges			
RoleTemplate			
RoleTemplatePrivileges			
DuplicateRule			
DuplicateRuleCondition			
SalesLiterature			
SalesLiteratureItem			
SalesOrder			
SalesOrderDetail			
SavedQuery			
SdkMessage			
SdkMessagePair			
SdkMessageRequest			
SdkMessageRequestField			
SdkMessageRequestInput			
SdkMessageResponse			
SdkMessageResponseField			
SdkMessageFilter			
SdkMessageProcessingStep			
SdkMessageProcessingStepImage			
SemiAnnualFiscalCalendar			
Service			X
ServiceAppointment			
ServiceContractContacts			
Site			
StatusMap			
StringMap			
Subject			
Subscription			
SubscriptionClients			
SubscriptionSynclInfo			
SystemUser			
SystemUserLicenses			
SystemUserPrincipals			
SystemUserRoles			
Task	X		X
Team	X		
TeamMembership			
Template			
Territory			X
TransformationMapping			
TransformationParameterMapping			
UnresolvedAddress			
UoM			
UoMSchedule	X		X
UserFiscalCalendar			
UserQuery			
UserSettings	X	X	X
WorkflowCompletedScope			
WorkflowWaitSubscription			
Workflow	X		
WorkflowDependency			

Table D.10: SugarCRM Information Entities Mapped - Part 1

SugarCRM	SageCRM	Oracle Siebel
accounts	X	X
accountsaudit		
accountsbugs		
accountscases		
accountscontacts		
accountsopportunities		
aclactions		
aclfields		
aclroles		
aclrolesactions		
aclrolesusers		
activities		X
activitiesusers		
addressbook		
addressbooklistitems		
addressbooklists		
bugs		
bugsaudit		
calls	X	
callscontacts		
callsleads		
callsusers		
campaignlog		
campaigntrkrs		
campaigns	X	X
campaignsaudit		
cases	X	
casesaudit		
casesbugs		
categorytree		
comments		
config		
contacts		X
contactsaudit		
contactsbugs		
contactscases		
contactsusers		
contracttypes		
contracts		X
contractsaudit		
contractscontacts		
contractsopportunities		
contractsproducts		
contractsquotes		
currencies		
customfields		
customqueries		
dashboards		
datasets		
datasetattributes		
datasetlayouts		
documentrevisions		
documents		X
documentsaccounts		
documentsbugs		
documentscases		
documentscontacts		
documentsopportunities		
documentsproducts		
documentsquotes		
documentsrevenueitems		

Table D.11: SugarCRM Information Entities Mapped - Part 2

SugarCRM	SageCRM	Oracle Siebel
eapm		
emailaddrbeanrel		
emailaddresses		
emailcache		
emailmarketing		
emailmarketingprospectlists		
emailtemplates		
emailman		
emails	X	
emailsbeans		
emailsemailaddrrel		
emailstext		
expressions		
fieldsmetadata		
filters		
folders		
foldersrel		
folderssubscriptions		
forecastmanagerworksheets		
forecastmanagerworksheetsaudit		
forecastschedule		
forecasttree		
forecastworksheets		
forecasts	X	X
ftsqueue		
holidays		
importmaps		
inboundemail		
inboundemailautoreply		
inboundemailcachets		
jobqueue		
kbcontents		
kbcontentsaudit		
kbdocumentrevisions		
kbdocuments		
kbdocumentskbtags		
kbdocumentsviewsratings		
kbtags		
leads	X	X
leadsaudit		
linkedddocuments		
manufacturers		
meetings		
meetingscontacts		
meetingsleads		
meetingsusers		
notes		
notifications		
notificationsaudit		
oauthconsumer		
oauthnonce		
oauthtokens		
opportunities	X	X
opportunitiesaudit		
opportunitiescontacts		
outboundemail		
pdfmanager		
productbundlenote		
productbundlenotes		
productbundleproduct		
productbundlequote		
productbundles		
productcategories		
productproduct		
producttemplates		
producttemplatesaudit		
producttypes		
products	X	X
productsaudit		
project		
projectresources		
projecttask		
projecttaskaudit		
projectsaccounts		
projectsbugs		
projectscases		
projectscontacts		
projectsopportunities		
projectsproducts		
projectsquotes		
projectsrevenueineitems		
prospectlistcampaigns		
prospectlists		
prospectlistsprospects		
prospects		X

Table D.12: SugarCRM Information Entities Mapped - Part 3

SugarCRM	SageCRM	Oracle Siebel
quotas		X
quotes	X	X
quotesaccounts		
quotesaudit		
quotescontacts		
quotesopportunities		
recordlist		
relationships		
releases		
reportcache		
reportmaker		
reportschedules		
revenueineitems		
revenueineitemsaudit		
roles		
rolesmodules		
rolesusers		
savedreports		
savedsearch		
schedulerulers		X
schedulerstimes		
sessionactive		
sessionhistory		
shippers		
styleguide		
subscriptions		
sugarfavorites		
systems		
tasks		
taxrates		
teammemberships		
teamnotices		
teamsets		
teamsetsmodules		
teamsetsteams		
teams		
timeperiods		
tracker		
trackerperf		
trackerqueries		
trackersessions		
trackertrackerqueries		
upgradehistory		
userpreferences		
users	X	X
usersfeeds		
usersholidays		
userslastimport		
userspasswordlink		
userssignatures		
vcals		
versions		
weblogichooks		
workflow		
workflowactions		
workflowactionshells		
workflwalerts		
workflwalertshells		
workflowschedules		
workflowtriggershells		

Table D.13: Sage CRM Information Entities Mapped

SageCRM	Oracle Siebel
Account	X
Account Progress	
Address	
Business Calendar	
Business Calendar Items	
Call List Tracker	
Campaign	X
Case	
Communication	X
Communication Link	
Company	X
Email	
Forecast	X
History	
Leads	X
Marketing	X
Opportunity	X
Opportunity History	
Opportunity Item	
Opportunity Progress	
Order items	X
Orders	X
Person	X
Person Link	
Phone	
Products	X
Quotes	X
Recurrance	
SLA	
SLA Severity	
Target List	
User Contacts	X
Users	X
Wave	X
Wave Item	

Table D.14: Buttle Components Mapped

Buttle Components	Salesforce	Microsoft	SugarCRM	SageCRM	Oracle Siebel
Agreements and Warranties	X	X	X		X
Budgets					X
Buying Process					
Campaign	X	X	X	X	X
Campaign Activities		X	X	X	X
Campaign Responses	X	X			X
Case	X	X	X	X	
Case Resolution	X	X			
Case Solution	X	X			
Competitor		X			X
Competitor Product		X			
Contact	X	X	X		X
Customer		X			
Customer Product	X			X	X
Dialogue Scripts					
Forecast	X		X	X	X
Individual				X	X
Inventory					
Known Issue	X	X	X		
Lead	X	X	X	X	X
Lists		X			X
Marketing Funds					X
Opportunity	X	X	X	X	X
Orders	X	X		X	X
Organization	X	X		X	X
Partner	X				X
Plans					X
Pricing	X	X			X
Product	X	X	X	X	X
Prospect			X	X	X
Quotas			X		X
Quotes	X	X	X	X	X
Sales Activities	X	X	X		X
Sales Methodology					
Sales Team			X		X
Segments					X
Service Activities	X	X	X		X
Service Request		X			X
Service Team			X		X
Territory	X	X			X
Catalog					X

