Revista de Gestão da Tecnologia e Sistemas de Informação
Journal of Information Systems and Technology Management
Vol. 4, No. 3, 2007, p. 287-300
ISSN online: 1807-1775

AN INTEGRATED METHODOLOGY FOR CUSTOMER RELATIONSHIP MANAGEMENT CUSTOMIZATION

Ricardo Colomo Palacios
Juan Miguel Gómez Berbís
Ángel García Crespo
Universidad de Carlos III de Madrid, Spain

ABSTRACT

The importance and presence of technological solutions in organizations supporting CRM are a vital business fact from the late nineties. Presently, the manufacturers figure in the market has dramatically decreased because of continuous takeovers and merges, but it has on the other hand gained momentum because of the sudden open-source and on-demand solutions appearance. In this scope, a unified methodology centered on CRM solutions is of paramount importance since it has traditionally been linked to either system integration or overall solution design. Based on the two de-facto complementary standards for the implementation and development of Information Systems, namely the ESA and Dyché CRM systems implementation methodology, in this paper, we provide a CRM business solutions customization methodology which pertains independently to the integration and tool maker perspective.

Keywords: Customer Relationship Management, CRM, Methodology, Customization, Software Engineering.

Recebido em/*Manuscript first received*: 17/04/2007 Aprovado em/*Manuscript accepted*: 05/11/2007

Endereço para correspondência/ *Address for correspondence*

Ricardo Colomo Palacios, Dpto. de Informática, Universidad de Carlos III de Madrid, Spain, Av. Universidad 30, Leganés, 28911, Madrid (Spain) Phone: +34 91 624 59 58 Fax: +34 624 94 30 E-mail: ricardo.colomo@uc3m.es

Juan Miguel Gómez Berbís, Dpto. de Informática, Universidad de Carlos III de Madrid, Spain, Av. Universidad 30, Leganés, 28911, Madrid (Spain) Phone: +34 91 624 59 58 Fax: +34 624 94 30 E-mail: juanmiguel.gomez@uc3m.es

Ángel García Crespo, Dpto. de Informática, Universidad de Carlos III de Madrid, Spain, Av. Universidad 30, Leganés, 28911, Madrid (Spain) Phone: +34 91 624 59 58 Fax: +34 624 94 30 E-mail: angel.garcia@uc3m.es

ISSN online: 1807-1775
Publicado por/*Published by*: TECSI FEA USP – 2007

1. INTRODUCTION

Customer relationship management (CRM) has become a research focus in the academic field since Ives and Learmonth (1984) put forward the customer relationship life cycle (CRLC) concept. CRM refers to a customer-focused business strategy. There are several definitions of CRM in the literature, for example the one provided by Dyché (2002) who defines the concept as "The infrastructure that enables the delineation of and increase in customer value, and the correct means by which to motivate customers to remain loyal, indeed to buy again". Other authors (Greenberg, 2001) define CRM in terms of the kind of process that CRM makes possible:

- Having an integrated, single view of customers, by using analytical tools.
- Managing customer relationships in a single way, regardless of the communication channel: telephone, website, personal visit...
- Improving the effectiveness and efficiency of the processes involved in customer relationships.

A proof-of-concept example is the CRM portal CRMGuru (http://www.crmguru.com) where the CRM concept is defined as the following: "Customer Relationship Management (CRM) is a business strategy to select and manage customers to optimize long-term value. CRM requires a customer-centric business philosophy and culture to support effective marketing, sales, and service processes. CRM applications can enable effective Customer Relationship Management, provided that an enterprise has the right leadership, strategy, and culture". Although all CRM definitions differ somewhat, they all focus on such individual and longitudinal buyer-seller relationship when both parties benefit from the relationship established (Sin, Tse & Yim, 2005).

Speaking in layman terms, business firms around the globe are spending billions of dollars a year on what software suppliers call ''CRM technology'' (Zablah, Bellenger & Johnston, 2004). Forrester (Bald, Ragsdale, Ferrusi Ross& Schuler, 2006) pointed that enterprises would spend about $3 billion worldwide on new CRM software licenses in 2006. In principle, companies can spend two to three times the initial license costs on systems integration services and new installation/support hardware. Therefore, worldwide expenditures on CRM consulting and systems integration are likely in the range of $6 billion per year.

In the hardware significant investments scenario, software is acquired by means of software licences and professional services for the purchased "suite solutions" customization. Recently, open-source and free-software software companies have emerged as hard competitors coming into the arena. These new players effectively offer more competitive prices for their solutions, but, particularly, they object to forcing customers to customize their business software solutions using precise development platforms for that very purpose. In this new environment, where open-source solutions share space with standard commercial solutions, it is necessary a unified methodology for the customization of the set of tools which will be available in the short term future. This methodology should not be based on commercial products, as it happens with proprietary methodologies stemming from software suppliers but on a different angle.

This angle encompasses the development processes that will take place with the goal of adapting "out of the box" functionalities of software suite solutions to the customer requirements.

In addition, the documentation and all development efforts must comply with general purpose and world widely extended methodologies that enable knowledge transfer and sharing, remarkable maintenance and availability of the software and widely-used collaboration schemas and commitments.

It is the main goal of this paper to present such methodology since we believe it is of the utmost importance to count on a software methodology based on the European Space Agency (ESA) software methodology to customize and apply to CRM solutions. The remainder of this paper is organized as follows. In next section, we analyze the current concept and market of CRM. Third section describes briefly our proposed methodology and in the subsequent section we discuss the developed methodology in a more detailed manner. Finally, the last section outlines our main conclusions and presents our future work.

1 CUSTOMER RELATIONSHIP MANAGEMENT TODAY

In this section, we will present the CRM world view as it is today. We summarize and explain the CRM concept and why it is so important for current the software business landscape.

1.1 CRM Solutions

The current CRM solutions market is undergoing deep changes. Since 1999 a number of merges and takeovers have favoured significant changes in terms of CRM and ERP scene agents, to the extent that currently the main players of these traditional market segments are essentially the same ones. The initial business model based on the sale of software licences has been combined with the irruption of the ASP model (internet based), which allows organizations with lower incomes to adopt this kind of technology with a lower risk and technological infrastructure, apart from reducing expenses. According to Gartner, SAP is the market leader closely followed by Siebel, Peoplesoft, Oracle and Salesforce.com. This latter company offers their services only following the ASP business model.

On the other hand, Microsoft does not enjoy a privileged pole position in the sector, however they are strongly trying to change that with the release of Microsoft Dynamics CRM 3.0. This situation has changed with the merges of Oracle, PeopleSoft and Siebel, and will be modified with the generalization of ASP services and the very likely emergence and prominence of open-source solutions in the CRM market. Nowadays, leading the whole lattice of open-source solutions, SugarCRM (fully-based on open-source technologies such as PHP or MySQL) and Hipergate (fully-based on PostgreSQL and Tomcat) claim to be fully-fledged alternatives to commercial solutions.

According to Herbert (2006), Open source CRM promises freedom from vendor lock-in, flexibility to tightly map the software to business processes, and extensibility to grow with changing business needs. On the downside, the solutions lack enterprise-class

functionality, have unproven scalability, and aren't backed by the deep pockets of a Microsoft or a SAP, meaning buyers must assume some risk around product support, maintenance, and upgrades.

The technological environment has also changed since the nineties. The unbelievable breakthrough of the Internet has meant the adoption of browser as a fundamental client. The extension of solutions scope from LAN environments to WAN environments where traditional customers based on executable programs installed in client machines or intelligent phones, PDAs and mobile devices can be equally found.

Regarding the technology that is been used, three types of CRM can be identified:

- Operational CRM. Is the automation of horizontally integrated business processes involving front-office customer touch points, such as customer service, sales and marketing, via multiple, interconnected delivery channels (phone, email, web…) and integration between front office and back office.
- Collaborative CRM. Various departments of a company, such as sales, technical support, and marketing, share any information they collect from interactions with customers, embracing all customer 'touch points' (communication channels such as e-mail, phone, fax, and Web pages).
- Analytical CRM. Comprises all programming that analyzes data about an enterprise's customer and presents it so that better and quicker business decisions can be made. Customer data can be captured from different sources on the operational side of the CRM equation and stored in data marts or data warehouses such as customer repositories. CRM analytics can provide customer segmentation groupings, profitability analysis, personalization, event monitoring, what-if scenarios and predictive modelling.

1.2 CRM & Methology

The CRM customization methodology has always come from traditional suite suppliers. The relevance of business and return on investment (ROI) figures and the installed applications have fully justified this circumstance. In addition, the integrator's importance in industry has flooded the current landscape with adaptations of corporate methodologies for CRM solutions. An example of these methodologies is Summit Ascendant, a family of business process methodologies that were originally developed and maintained by Price WaterhouseCoopers Consulting and are now part of IBM's Rational brand, Fujitsu Macroscope or Accenture Delivery Methods.

On the other hand, the open-source phenomenon has not impacted the specific and tailor-made methodologies for designing and transferring CRM solutions. However, the importance of the phenomenon itself has clearly meant a new and promising research line: the use of software engineering techniques for open-source solutions.

Particularly, in this section, we will discuss proprietary methodologies of most prominent software suppliers in the CRM market and then the main contributions of

methodologies research for the transfer of CRM systems.

Siebel, a company recently acquired by Oracle, suggest the use of the eRoadmap methodology for the transfer of Siebel solutions or projects in which Siebel solutions are involved. The methodology is based on the following stages (Kale, 2003): Plan Stage. Definition Stage. Discover Stage. Design Stage. Configuration Stage. Validation Stage. Deployment Stage. Sustain Stage.

Another remarkable player in the operational CRM solutions market is SAP. The German company proposes the ASAP methodology (for Accelerated SAP) to support their customers in transferring their solutions with competitive costs and time constraints. This methodology is designed for ERP solutions mainly, but given its common integration with CRM solutions, it is deemed relevant for our work. The components of ASAP can be used together or individually and are called accelerators. Accelerators are based on the best practices of SAP customers from around the world and consist of a number of templates, questions, and scenarios that require user input to help the user determine the best way to implement their SAP system.

The ASAP Roadmap is a high-level implementation plan. It divides the implementation into five phases. These phases are often referred to as "steps" so as not to be confused with a phased implementation of functionality; they are steps in the overall implementation process.

- Project Preparation. During the Project Preparation phase, the project team makes initial plans and preparation for the implementation. They set up the project management office (PMO), define the statement of work, and publish the detailed project plan.
- Business Blueprint. During the Business Blueprint phase, all stakeholders work to understand the project's business goals and to determine the business processes required to support those goals. This is the scope definition.
- Implementation. During the Implementation phase, the project team configures the SAP modules and implements defined processes based on the business blueprint. Design and development activities for modifications identified during the Business Blueprint phase begin. The system integration test is conducted.
- Final Preparation. During the Final Preparation phase, the team completes final preparation activities before going live, including final system testing, end-user training, data cutover, and system cutover to a production environment.
- Go Live and Support. During the Go-Live and Support phase, the project-oriented pre-production environment is transitioned into a successful, live production operation.

The methodologies for CRM transfer are independent of software suppliers which are not concerned with technical literature. A remarkable effort has been carried out in the "CRM Iris methodology" (Chalmeta, 2006) project. The aforementioned methodology was developed by the Universidad Jaime I de Castellón since 2000 and it

encompasses multiple aspects from the transfer of CRM tools and systems to the building-up of information systems. The main elements of this effort are as follows:

- Project management and prerequisites.
- Definition of the company's organisational framework.
- Definition of a customer strategy.
- Designing a customer relationship assessment system.
- Process map.
- Human resources organisation and management.
- Construction of the information system.
- Implementation.
- Monitoring.

A more development-driven initiative is presented in the (Kim, 2004) recommendation. Through two business cases analysis, a twelve factor-driven model is presented to understand and handle the social elements to be considered in this kind of models. The proposed process model elucidates the technology and social factors associated with CRM development, and how they influence each other and then lead to the IS development consequences. The model aids in the empirical detection of repeating patterns of social activity in IS development.

Independently to the existence of a method for the adoption of CRM capabilities in an organization, the need to rely on a common and unified methodology for software design and transfer in the scope of an organization is out of question. Even if a company based the complete set of software solutions in ERP-like commercial solutions, it would be required to establish a number of procedures, provided the traditional mobility of IT work market is maintained. These procedures would enable knowledge transfer and management as well as risk and investment management, among others.

The most relevant methodologies in the CRM context spawn many concepts which lack of the appropriate depth in Software Engineering aspects that are woven together and are covered by this discipline in a broader manner. However, CRM methodologies neither care nor study them properly, what unleashes the full potential of a new research field.

2 THE ESA METHODOLOGY: COMMITMENT DOES THE TRICK

The history of the European Space Agency (ESA) software methodology is a direct consequence of the need of standardization in software engineering methodologies. In the mid seventies, the ESA project members involved in software

development were often brilliant engineers, able to find innovative solutions to problems. However, they were not used to working in a project environment with rigorous cost and schedule constraints. Each followed their own methods and there was little project discipline (Jones & Mazza, 1997). At that point in time, one of the ESA founders was nominated as project manager to lead a very critical software development at the European Space Operations Centre (ESOC). The project was a tremendous failure. Delivery was always late and major hardware and operating system problems were constantly arising. The idea of reducing requirements and working on those essentials for the launch came up. But, where were the requirements? There were no written requirements. How could costs and schedule be guaranteed under these circumstances?

In May 1977, resulting from such critical situation, the ESA established the Board for Software Standardisation and Control (BSSC). Since then, the BSSC has produced a highly successful software engineering standard, first issued in 1984, which has been applied extensively, mainly in the context of the ESA cutting-edge software projects. The specifications released by the BSSC were named the ESA Software Engineering Standards and it is renewed every three or four years. The last issued version of these tandards is ESA PSS-05-0. The PSS-05-0 standard describes the processes involved in the complete life cycle of a single software project from its inception to the retirement of the software. The standard is divided into two areas, namely the Production Process, which has six phases and the Managing Process, which counts on four principal phases. The Production Process is composed by:

- User Requirements (UR) Definition Phase
- Software Requirements (SR) Definition Phase
- Architectural Design (AD) Phase
- Detailed Design (DD) Phase
- Transfer (TR) Phase
- Operations and Maintenance (OM) Phase

Additionally, the Managing Process is composed by four principal phases:

- Software Project Management
- Software Configuration Management
- Software Verification and Validation
- Software Quality Assurance

Fundamentally, the ESA specifications encourage the development of a formalized, knowledge-aware, cross-cultural set of guidelines to face software projects. First of all, the ESA is concerned on work organization as discussed in (Strübing, 1993). In practice, developers tend to mix the different phases of the development process. This mixing is enforced by the constraints of the work situation, because complicated tasks and external requests –such as deadlines– perturb the order of the phases. Professional experience, as well as a need for motivation or change, create individual

characteristics which encourage (or discourage) such phase mixture. Already explicitly specified in the organisation as well as in the degree of division of labour, the process model presents organisational necessities that affect developers' individual activities.

Secondly, cross-cultural teams tackle with software projects differently. Cross-cultural studies suggest that the division of tasks might also be influenced by national features. For instance, an empirical study detailed in (Schroll-Machl, 1996) concludes that there are differences in the division of labour between Germany and the USA. She describes German workgroups as preferring to work on a problem as a whole, while American groups work in an output-oriented way on the separated components of a problem. Therefore, the organization of the work must be explicitly declared at the beginning of the project because the organisation of individual work in software development may also depend on societal or political factors within the boundaries of the individual culture.

Knowledge management is a vital cornerstone of the ESA standard. Knowledge management is required and applied for various reasons such as long duration projects, transfer of expertise to other projects and change in staff complement. In this case, the differences between team members may be an advantage from which the developing process could get benefit. The European Space Operations Center (ESOC), in Germany, is one of the four establishments in Europe of the ESA. Here, a working group was set up with the goal to look into the area of knowledge management (Mugellesi, 2004). The group has investigated the standard directions such as:

- Knowledge preservation within individual domain or project.
- Cross functional teams.
- Knowledge evolution.

Furthermore, the ESA standard deals with *organizational patterns* (Coplien, 2004). Organizational patterns take into account the structures of human relationships, rather than the technological links, to drive the real architecture of software. The fundamental principles of this theory come into life when thinking about the principles in software developing. Each software development organization has its own architectural principles. Such principles as coupling and cohesion and modularity are almost universal whereas some other principles such as some design styles as object orientation are subject of particular software design patches. The ESA standard defines such boundaries using their last but not least feature: formalization.

Formalization is an extreme form of a formation process, involving language (Siefkes, 1997). Concepts and their values are explicitly declared and rules are set up for certain mental or social tasks, to control and direct the characteristics of objects and behaviour. The goal of formalization is to isolate unrelated objects and activities from their surroundings, in particular to prevent them from changing. Formalization means to establish a rational and by-the-book process for all procedures. This also means to overcome social and cultural limitations by imposing a set of rules.

In the ESA standard, all these features contribute to make from the Production Process and the Managing Process, a rationally-supported methodology which establishes a common basis for software projects.

3 A PROPOSAL FOR INTEGRATED CRM CUSTOMIZATION METHODOLOGY

In previous sections, the importance and global-view of CRM solutions has been stated. As it can be inferred from the aforementioned, transfer projects for a CRM solution have strategic features. Understanding the complexity of CRM Projects is critical to planning this particular kind of projects. In this sense, a reasonable transfer roadmap of this kind of project including the connection of a corporate strategy can be found in Dyche (2002). The roadmap counts on a number of phases in which six steps are performed and accomplished integrating a set of tasks, as it is depicted in Figure 1.

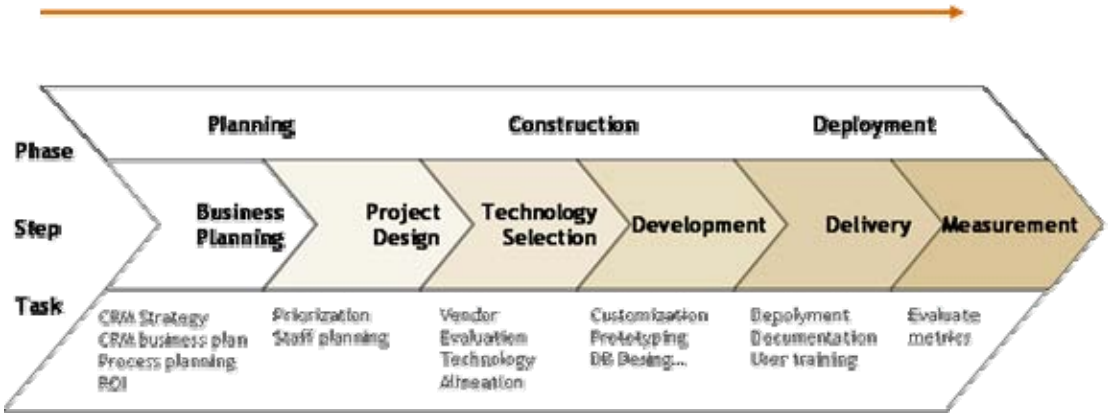

Figure 1. CRM implementation roadmap

As can be noticed from the figure, customization is one of the "Development Step" tasks. Nevertheless, this task is deemed vital for the adaptation of the company needs in terms of the functionalities of the tool, avoiding influence in the opposite sense, i.e. the adaptation of the company to the tool features. This could be appropriate for some types of companies but in most of them, a consistent culture and solid results do not advice it.

As it has been underscored previously, CRM solutions implementation methodologies do not support at its full potential Information Systems development, particularly those concerned which the application of methods and tools related to Software Engineering. From this standpoint, some best practices located in the CRM Implementation Roadmap gearing toward success in software development projects are put aside. The detection of such circumstance has been the main inspirer of the work presented in this paper.

On the one hand, the ESA methodology is a de-facto Standard for Information Systems development at European level and, on the other hand, Dyché CRM Implementation Roadmap is envisaged as a reference model used in the software transfer of this kind of Information Systems. Both models are complemented due to its cascade sequential nature and their methodological focus: CRM Implementation

Roadmap is based on CRM projects and the ESA methodology on business processes. While the CRM Implementation Roadmap can work in the CRM systems transfer with lightweight customizations or "Out of the Box" systems, significant changes in the software suites or bundles, planning needs, documentation and development methodology makes it very important to use a methodology based on Software Engineering principles. In the European context, the most relevant methodology is promoted by the ESA.

Our proposal consists of a best-of-breed mapping of the ESA methodology detailed in section 3 with the aforementioned phases of the CRM methodology. By matching the various stages of this methodology with the ESA, the mapping benefits from the positive features previously outlined regarding formalization, cross-cultural and organizational patterns in a widely-used software methodology. The mapping is depicted in Figure 2.

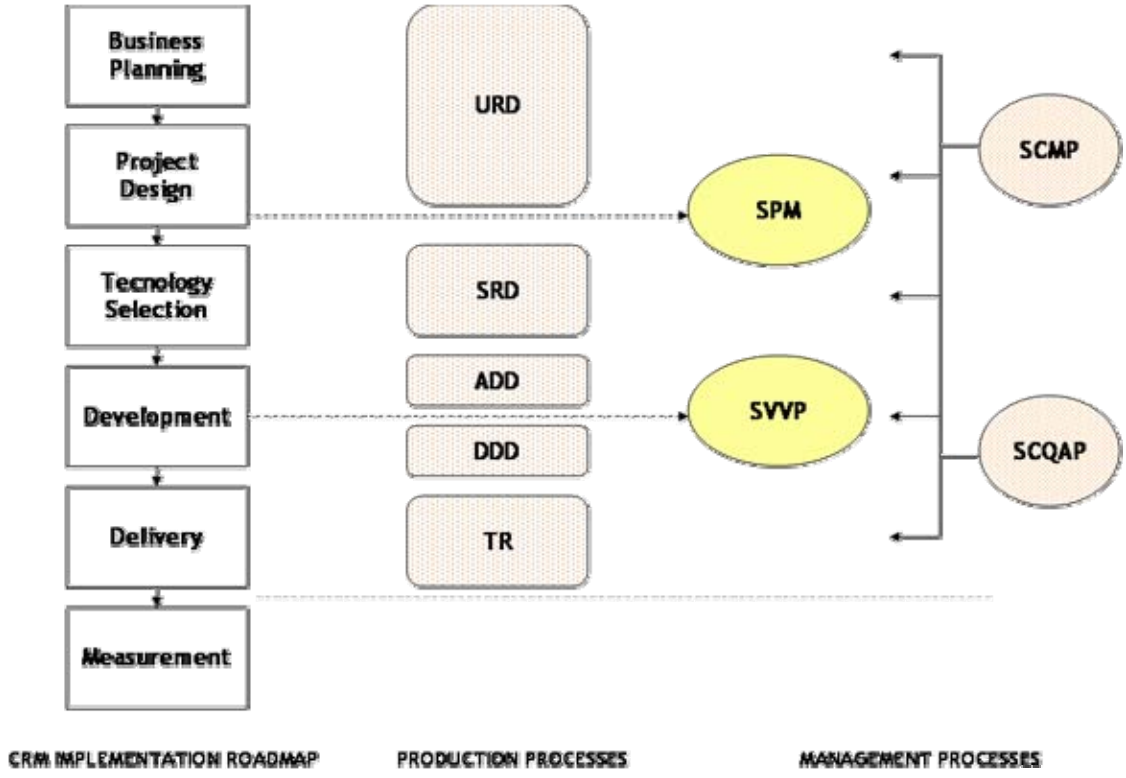

Figure 2. Unified CRM & ESA Methodology Matching

In a nutshell, our mapping unveils the similarity of both approaches. Particularly, between the Production Process and the CRM stages, since the Managing Process phase (Software Project Management, Software Configuration, Software Verification and Validation and Software Quality) is horizontal to the other phases.

Firstly, the Business Planning CRM phase involves the critical activity of planning business requirements of a corporate CRM strategy and the definition of the corresponding boundaries. Hence, this phase corresponds neatly with the ESA User

Requirements (URD) phase, which aims at enacting the 'problem definition phase' of the life cycle. The phase refines an idea about a task to be performed using computing equipment, into a definition of what is expected from the computer system. Secondly, the CRM Project Design phase identifies the business process the CRM product will support and encompasses the design and implementation strategy. Such phase is equivalent to the Software Project Management (SPM) phase (from the Managing Phase, ultimately) where these constraints are set up and provided for the rest of the software project to be concluded successfully.

In addition, the CRM Technology Selection phase ranges from being as simple as choosing "off-the-shelf" components to a comprehensive and exhaustive evaluation of CRM products. This phase is highly tuned and related with the ESA Software Requirements (SRD) phase where requirements to be fulfilled are determined and consequently, decisions related to technology are taken.

The CRM Development phase is clearly the most ambitious and broad with regards to construction and customization of the CRM product, what implies designing the architecture and building the detailed design. Hence this CRM phase encompasses the ESA Architectural Design phase (ADD) and the Detailed Design phase (DDD). Furthermore, the Software Verification and Validation (SVV) phase of the ESA is also comprised in this phase, since verification and validation tests are prone to happen during development.

Finally, the CRM Delivery phase, which is often overlooked or lumped into development, attempts to leverage the corporation infrastructure and to dispatch the resulting software. This is completely covered by the ESA Transfer Phase (TR), whose goal is to handle the transfer of the involved technical processes in the software project by applying them to the new scenario.

Last but not least, the ESA Software Configuration Management (SCM) and the Software Quality Management (SQAP) phase are horizontal to the whole software project, since they receive feedback and are modified throughout the process.

It has been our proposal to coordinate both approaches in this section. By achieving a unified methodology from a high-level perspective, breakthroughs from both approaches are reinforced and more accurately attained.

However, there is a remaining gap between the CRM requirements repositories (i.e. software requirements that are already configured and come "out of the box" with CRM commercial suites, for example) and the ESA own mechanism to trace the correctly addressing of software requirements. This issue is addressed and solved in the next section.

Customization Matrix: Bridging the Gap

In principle, most CRM systems come with a number of features already configured, ready-to-use and "out of the box". They offer these features as a response to a set of Requirements Repositories (RR). These repositories encapsulate a number of functionalities that might vary from single CRM Multimedia Message Service (MMS)

notification and request to Global Positioning Systems (GPS) or Geographical Information Systems (GIS) providing information about business resources or complex B2B integration protocol systems. Since these requirements might be interesting or critical depending on the organization, our proposal is to relate such RR with a software methodology mechanism that ensures the correct addressing of those requirements.

More particularly, the ESA Traceability Matrix is a well-founded mechanism to address software requirements from the UR and SR phases in the following architecture and requirement-oriented phases such as the Architectural Design and Detailed Design phases.

Hence we present a Customization Matrix that relates the elements of the CRM Requirements Repositories with the software requirements belonging to the previously mentioned phases of the ESA and hereby, of our unified methodology. The main contribution of the matrix is to establish a relationship between the Requirements Repositories elements shown in the rows and the software requirements described by means of columns. Each correspondence may have three values, namely: *total*, *partial* and *not supported. Total* means that the CRM requirement is fully supported by software requirements and *Partial* limits such support. Finally, *Not Supported* neglects completely the support of such requirement. An example of the Customization Matrix is shown in Table 1.

Table 1. Customization Matrix

	FNC_1	**FNC_2**	**…**	**FNC_3**	**Coverture**
RQ1. CTI Integration for MMS					*Not Supported*
RQ2. Global Positioning System (GPS) GIS Integration		X			*Partial*
RQ3. Multilanguage support					*Not Supported*
RQ4. B2B Protocol Handling	X				*Total*

As a result, CRM “out of the box” functionalities required or encouraged by an organization are shown in the Customization Matrix and can be subject of a further analysis. For instance, an organization Customization Matrix could be used to measure customer satisfaction, critical functionalities or forthcomings of a particular commercial suite.

4 CONCLUSIONS

The importance and prominence of CRM solutions in the IT business market is unveiled as a key element for the current Information Systems scenario. As a consequence of its relevance, the market presents multiple IS solutions that support CRM processes. This work proposes an integrated methodology, supplier and platform independent (what is independent? The platform?) to customize CRM solutions, either based on traditional platforms or internet based. The use of such methodology based on methodological standards implies a guarantee of knowledge transfer and maintenance of commercial applications that warrantees the consulting firm change or technology shift without significant knowledge loss.

For organizations, counting on a methodology fully based on development standards made up for CRM software customization can be a solution for one of the problems of CRM implementation: fully functional solution total cost ownership. Our proposal aims to be a customization cost cutter, because it can be a way to homogenize the whole IT solution to ensure global configuration management and agile and integrated parameterization process.

REFERENCES

Bald, W., Ragsdale, J., Ferrusi Ross, C. & Schuler, I. (2006). “How To Select A CRM Professional Services Provider. Forrester Best Practices”. From http://www.forrester.com/Research/Document/Excerpt/0,7211,38823,00.html

Chalmeta, R. (2006). “Methodology for customer relationship management”. The Journal of Systems and Software. 79(7), pp. 1015 - 1024

Coplien, J. (2004). “Organizational Patterns: Beyond Technology to People”. International Conference on Enterprise Information Systems. IS 15.

Dyche, J. (2002). The CRM handbook: A Business Guide to Customer Relationship Management, Addison-Wesley.

Greenberg, P. (2001). CRM at the Speed of Light: Capturing and Keeping Customers in Internet Real Time. McGraw-Hill Osborne Media.

Herbert, L. (2006). CRM Continues To Gain Momentum But Isn't Enterprise-Ready. Forrester Research. Available OnLine: http://www.forrester.com/Research/Document/Excerpt/0,7211,39163,00.html

Ives B. & Learmonth G.P. (1984). The information system as a compete it vie weapon. Communications of the ACM. 27(12), 1193-1201.

Jones, M.; Mazza, C. (1997). 1977-1997: Twenty Years of Software Engineering Standardisation in ESA. ESA Bulletin. 90. Available online:

http://www.esa.int/esapub/bulletin/bullet90/b90jones.htm

Kale, V. (2003). Implementing Siebel. Prentice Hall.

Kim, H. (2004). A Process Model for Successful CRM System Development". IEEE Software. 21(4), 22-28.

Mugellesi Dow, R. (2004). A knowledge management initiative in ESA/ESOC, 55th International Astronautical Congress.

Schroll-Machl, S. (1996). Kulturbedingte Unterschiede im Problemlösungsprozess bei deutschamerikanischen Arbeitsgruppen. In: Thomas, Alexander (Hrsg.): Psychologie des interkulturellen Handelns. Göttingen et al. : Hofgrefe. 383-409.

Siefkes, D. (1997). Computer Science as Cultural Development: Toward a Broader Theory. Technische Universität Berlin, FB Informatik. Available online:

http://tal.cs.tu-berlin.de/siefkes/ texte/1997/BroaderTheory.rtf

Sin, L.Y.M. and Tse, A.C.B. & Yim, F.H.K. (2005). CRM: conceptualization and scale development. European Journal of Marketing, 39(11/12), 1264-1290.

Strübing, J. (1993). Subjektive Leistungen im Arbeitsprozess: Eine empirische Untersuchung von Arbeitsstilenin der Programmierarbeit. Wiesbaden: Deutscher Universitäts Verlag.

Zablah, A. R., Bellenger D. N. & Johnston, W. J. (2004). An evaluation of divergent perspectives on customer relationship management: Towards a common understanding of an emerging phenomenon. Industrial Marketing Management, 33, 475-489.