

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación :

INGENIERO DE TELECOMUNICACIÓN

Título del proyecto:

DESIGN AND DEVELOPMENT OF A BUSINESS-TO-
BUSINESS INTEGRATION MODULE FOR OPENBRAVO
ERP

Juan Marcos Bernal Alejo

Carlos del Río Bocio

Pamplona, 1 de Julio de 2011

Acknowledgements

A mi familia.

A mis amigos.

A mis jefes y compañeros de trabajo.

A Carlos del Río por su comprensión.

A Rocío.

Contents

ACKNOWLEDGEMENTS	III
CONTENTS	V
LIST OF FIGURES AND TABLES	VII
CHAPTER 1. INTRODUCTION	1
1.1. MOTIVATION FOR THE DESIGN AND DEVELOPMENT OF A B2B INTEGRATION MODULE FOR OPENBRAVO ERP	1
1.2. THESIS ORGANIZATION	2
CHAPTER 2. THEORETICAL BODY OF KNOWLEDGE	3
2.1. ENTERPRISE RESOURCE PLANNING (ERP SYSTEMS)	3
2.1.1. <i>Theoretical Background</i>	4
2.1.3. <i>History of ERPs</i>	8
2.1.4. <i>Openbravo ERP</i>	11
2.2. SUPPLY CHAIN MANAGEMENT	33
2.2.1. <i>Theoretical Background</i>	33
2.2.2. <i>History of SCM</i>	35
2.2.3. <i>Just-In-Time, Kanban and Lean Management</i>	37
2.2.4. <i>B2B Integration</i>	38
2.3. SOFTWARE DEVELOPMENT METHODOLOGY AND OPEN SOURCE	44
2.3.1. <i>Software Development Methodology</i>	44
2.3.2. <i>Software Development Process</i>	46
2.3.3. <i>Open Source Software</i>	48
CHAPTER 3. OPENBRAVO ERP CORE BUSINESS FLOWS. B2B INTEGRATION GAP	55
3.1. CORE BUSINESS FLOWS FOR OPENBRAVO ERP	55
3.1.1. <i>Procure To Pay: Requisition to Receipt</i>	55
3.1.2. <i>Procure To Pay: Supplier Invoice to Payment</i>	58
3.1.3. <i>Financial Account to Reconciliation</i>	58
3.1.4. <i>Period End Close to Financial Report</i>	59
3.1.5. <i>Customer Return to Credit</i>	60
3.1.6. <i>Customer Return to Replacement</i>	61
3.1.7. <i>Supplier Return to Debit</i>	62
3.1.8. <i>Supplier Return to Replacement</i>	63
3.1.9. <i>Wall to Wall Accuracy Physical Count</i>	64
3.1.10. <i>Asset Acquisition to Dispose</i>	65
3.1.11. <i>Tax Payment</i>	66
3.1.12. <i>Sales Order to Cash</i>	66
3.2. B2B INTEGRATION FUNCTIONAL GAP	68
3.2.1. <i>Platform Functional Gap</i>	69
3.2.2. <i>PO Management Gap</i>	70
3.2.3. <i>Shipping and Goods Movement Gap</i>	70
3.2.4. <i>Warehouse Management Gap</i>	71
3.2.5. <i>Invoice Settlement and Payment Acknowledgement Gap</i>	72
3.2.6. <i>Vendor / Supplier Relationship Management</i>	72
3.2.7. <i>Reporting Functional Gap</i>	72
3.2.8. <i>Master Data Management Gap</i>	73

CHAPTER 4. FUNCTIONAL AND TECHNICAL DESIGN	75
4.1. FUNCTIONAL DESIGN	75
4.1.1. <i>Design considerations and constraints</i>	75
4.1.2. <i>B2B Integration Module purpose and scope</i>	75
4.1.3. <i>User roles & profiles</i>	77
4.1.4. <i>B2B Integration Platform Extensions</i>	77
4.1.5. <i>B2B Integration Master Data Management Extensions</i>	80
4.1.6. <i>Purchase Order Management Extensions</i>	82
4.1.7. <i>Vendor Managed Inventory Extensions</i>	88
4.1.8. <i>Evaluated Receipt Settlement Extensions</i>	90
4.1.9. <i>B2B Integration Reports and Widgets</i>	91
4.2. TECHNICAL DESIGN.....	96
4.2.1. <i>Technical Design Considerations</i>	96
4.2.1. <i>Module Architecture</i>	96
4.2.2. <i>Technical Implementation of Platform Extensions</i>	100
4.2.3. <i>Technical Implementation of Master Data Management Extensions</i>	102
4.2.4. <i>Technical Implementation of Purchase Order Management Extensions</i>	104
4.2.5. <i>Technical Implementation of Vendor Managed Inventory Extensions</i>	106
4.2.6. <i>Technical Implementation of Evaluated Receipt Settlement Extensions</i>	107
4.2.7. <i>Technical Implementation of Widgets and Reports</i>	108
CHAPTER 5. PROJECT DEFINITION. DEVELOPMENT PLAN.	109
5.1. TASKS BREAKDOWN AND COST ESTIMATION.....	109
5.1.1. <i>Effort Diagram</i>	109
5.1.2. <i>Overhead Activities</i>	112
5.1.3. <i>Tasks Dependencies</i>	112
5.2. PROJECT PLAN	113
5.2.1. <i>Available Resources</i>	113
5.2.2. <i>Project Scope</i>	113
5.2.3. <i>Project Scheduling</i>	113
CHAPTER 6. PUBLISHING AND SUPPORT PLAN. FUTURE DEVELOPMENTS.....	115
6.1. OPENBRAVO FORGE AND CENTRAL REPOSITORY	115
6.1.1. <i>Community communication and collaboration</i>	115
6.1.2. <i>Functionality for end users and module developers</i>	117
6.2. B2B INTEGRATION PROJECT	121
6.2.1. <i>Project Name, Project Logo and Description.</i>	121
6.2.2. <i>News Section</i>	122
6.2.3 <i>Code Repository</i>	123
6.2.4. <i>Wiki</i>	123
6.2.5. <i>Forums and Support</i>	123
6.2.6. <i>Issue Tracker</i>	123
6.2.7. <i>Available Downloads</i>	123
6.3. FUTURE PLANS AND EXTENSIONS.....	123
6.3.1. <i>Supporting the core B2B Module</i>	123
6.3.2. <i>Extending the B2B Integration module</i>	124
CHAPTER 7. CONCLUSIONS	127
GLOSSARY.....	129

List of figures and tables

FIGURE I - EXAMPLE OF STANDARD ERP FUNCTIONALITY	4
FIGURE II - EXAMPLE OF INTEGRATED BUSINESS PROCESSES.....	6
FIGURE III - EVOLUTION OF ERPS TIMELINE	10
FIGURE IV - OPENBRAVO ERP, THE AGILE ERP	11
FIGURE V - OPENBRAVO ERP FUNCTIONALITY OVERVIEW	12
FIGURE VI - OPENBRAVO ERP FUNCTIONALITY: MASTER DATA MANAGEMENT	13
FIGURE VII - OPENBRAVO ERP FUNCTIONALITY: FINANCIAL MANAGEMENT AND ACCOUNTING.....	14
FIGURE VIII - OPENBRAVO ERP FUNCTIONALITY: PROCUREMENT MANAGEMENT.....	16
FIGURE IX - OPENBRAVO ERP FUNCTIONALITY: WAREHOUSE MANAGEMENT	17
FIGURE X - OPENBRAVO ERP FUNCTIONALITY: SALES MANAGEMENT.....	18
FIGURE XI - OPENBRAVO ERP FUNCTIONALITY: PROJECT AND SERVICE MANAGEMENT.....	19
FIGURE XII - OPENBRAVO ERP FUNCTIONALITY: PRODUCTION MANAGEMENT.....	20
FIGURE XIII - OPENBRAVO ERP FUNCTIONALITY: BUSINESS INTELLIGENCE.....	21
FIGURE XIV - CLIENT-SERVER PARADIGM.....	22
FIGURE XV - LOGIN SCREEN	23
FIGURE XVI - DEFAULT WORKSPACE EXAMPLE	23
FIGURE XVII - OPENBRAVO ERP ROLE SELECTION TOOL.....	24
FIGURE XVIII - OPENBRAVO ERP TABBED NAVIGATION EXAMPLE.....	25
FIGURE XIX - OPENBRAVO ERP GRID VIEW AND FORM VIEW. FILTERING	26
FIGURE XX - OPERATING ENVIRONMENT OF OPENBRAVO ERP.....	26
FIGURE XXI - DATA ACCESS LAYER DIAGRAM	28
FIGURE XXII - DIFFERENT TYPES OF OPENBRAVO ERP MODULES	29
FIGURE XXIII - EXAMPLE OF COMBINATIONS OF DIFFERENT MODULE TYPES.....	30
FIGURE XXIV - OPENBRAVO'S CENTRAL MODULE REPOSITORY	30
FIGURE XXV - MODULE MANAGEMENT CONSOLE	31
FIGURE XXVI - EXPECTED EVOLUTION OF SUPPLY CHAIN AFTER B2B INTEGRATION	39
FIGURE XXVII – THE THREE MOST COMMON SOFTWARE DEVELOPMENT MODELS	44
FIGURE XXVIII - BUSINESS FLOW: PROCURE TO PAY.....	56
FIGURE XXIX - SENDING A PURCHASE ORDER TO A SUPPLIER.....	57
FIGURE XXX - BUSINESS FLOW: FINANCIAL ACCOUNT TO RECONCILIATION	59
FIGURE XXXI - BUSINESS FLOW: PERIOD END CLOSE TO FINANCIAL REPORT	60
FIGURE XXXII - BUSINESS FLOW: CUSTOMER RETURN TO CREDIT	61
FIGURE XXXIII - BUSINESS FLOW: CUSTOMER RETURN TO REPLACEMENT	62
FIGURE XXXIV - BUSINESS FLOW: SUPPLIER RETURN TO DEBIT	63
FIGURE XXXV - BUSINESS FLOW: SUPPLIER RETURN TO REPLACEMENT	63
FIGURE XXXVI - BUSINESS FLOW: WALL TO WALL ACCURACY PHYSICAL COUNT	65
FIGURE XXXVII - BUSINESS FLOW: ASSET ACQUISITION TO DISPOSE	66
FIGURE XXXVIII - BUSINESS FLOW: ORDER TO SHIPMENT	67
FIGURE XXXIX - BUSINESS FLOW: CUSTOMER INVOICE TO CASH	67
FIGURE XL - BUSINESS FLOW: COLLECTION.....	68
FIGURE XLI - B2B INTEGRATION MODULE OVERVIEW	76
FIGURE XLII - ADD NEW PRODUCT MOCKUP	81
FIGURE XLIII - B2B INTEGRATED PO MANAGEMENT	83
FIGURE XLIV - B2B PO ORDER AND ASN STATUS.....	87
FIGURE XLV - NEW ADVANCED SHIPMENT NOTICE MOCKUP	89
FIGURE XLVI - MY PRODUCTS SUMMARY WIDGET MOCKUP	92

FIGURE XLVII - MY PURCHASE ORDERS WIDGET MOCKUP	92
FIGURE XLVIII - MY B2B ORDERS WIDGET MOCKUP.....	93
FIGURE XLIX - MY INVOICES WIDGET MOCKUP.....	93
FIGURE L - MY OPEN INVOICES WIDGET MOCKUP	94
FIGURE LI - MY SHIPMENTS WIDGET MOCKUP.....	94
FIGURE LII - MY PAYMENTS WIDGET MOCKUP	95
FIGURE LIII - B2B INTEGRATION MODULE ARCHITECTURE.....	97
FIGURE LIV - TASK DEPENDENCIES DIAGRAM.....	113
FIGURE LV - PROJECT DEVELOPMENT PLAN	114
FIGURE LVI - OPENBRAVO FORGE LOGO	115
FIGURE LVII - OPENBRAVO WIKI LOGO.....	115
FIGURE LVIII - OPENBRAVO CODE REPOSITORIES	117
FIGURE LIX - DIAGRAM OF AVAILABLE SERVICES IN CENTRAL REPOSITORY	118
FIGURE LX - B2B INTEGRATION PROJECT PAGE	121
FIGURE LXI - B2B INTEGRATION FOR OPENBRAVO ERP LOGO.....	122

Chapter 1. Introduction

1.1. Motivation for the design and development of a B2B integration module for Openbravo ERP

The goal of this thesis is to design and develop a B2B Integration module for Openbravo ERP.

Openbravo ERP is an Enterprise Resource Management, open source software designed for SMEs, featuring a native web interface and a modular architecture that allows to easily extend and customize its functionalities. The core of the product contains tools to manage the purchasing, sales, warehouse, financial and accounting processes of an organization, typically for those occurring inside the internal boundaries of the organization.

Business-To-Business or B2B relationships consist on processes such as buying and selling of products and services, shipping, invoicing, paying and collecting money, just the same as standard transactions with persons, only that the way they are carried out is utterly different. These relationships usually involve the negotiation and evolution of workflows, contracts and management of B2B relation by appointed people in the companies' side: sales people, collectors, supply chain managers, etc. In those workflows, each organization takes the necessary data and performs transactions against their own information systems, information systems such as Openbravo ERP. This information could be transferred or fed between those systems.

In order to apply advanced Supply Chain Management techniques, and advanced Warehouse Management ideas such as Just-In-Time, Kanban and Lean Management, a very high level of trust and data sharing is required between the two or more organizations involved in these processes.

Different approaches will be taken, with the aim of creating a module for Openbravo ERP which will serve as a base for B2B Integration, this meaning, an infrastructure to integrate human and automated processes of B2B transactions. This thesis will discuss the existing problematic and current points of pain in B2B transactions, and will analyze the state of the art in software solutions. Taking into account the current theoretical background on Supply Chain Management and B2B Integration, a functional and technical design for a B2B module for Openbravo ERP will be proposed.

Afterwards, this thesis will study the cost and resources necessary for developing this extension and a cost-compromise will be taken, so that the most useful but basic functionalities will be developed and published and licensed as open source software.

After publishing the module and making the code publicly available, this thesis will also explain which infrastructure is needed for the support and follow up of the development of this module.

The module and subsequent project will be maintained and possibly extended beyond the scope of this text.

1.2. Thesis Organization

This thesis is divided into 7 chapters following the introduction.

Chapter 2 provides the theoretical background needed in order to understand and design a B2B Integration module for Openbravo ERP. It covers the three main areas of knowledge: explanation and history of **Enterprise Resource Planning** systems, **Supply Chain Management** theory, and Information Technology, **Open Source and Software Development Methodology**.

Chapter 3 studies the **core** business processes covered by **Openbravo ERP** and confronts them to the theories on Supply Chain Management explained beforehand, so that the set of missing functionalities, known as **Functional Gap** will arise, and serves as the reference for the Functional Design.

Chapter 4 presents the **Functional Design** of the B2B Integration module, a document which explains which will be the new functionalities, how will they be used, and which is the problem that they solve. Based on this Functional Design, a **Technical Design** is presented alongside, which defines in a technical manner how the new functionalities will be deployed within the existing technology and capabilities of the Openbravo ERP platform.

Chapter 5 will evaluate and measure the **cost** of developing each of the software artifacts, based on the Technical Design. With this information, a reasoned explanation and for each of the development units, a priority is given, so that taking into account resources and pay-offs the development **project plan** is presented.

Chapter 6 gives an explanation on which are the tools needed to publish, update, support and track a software project of this kind. Within the Openbravo ERP environment, this is provided by the specialized web portal known as **Openbravo Forge**.

Chapter 7 presents the **conclusions** of this thesis, presenting the big picture of the work that has been done, its impact and future development capabilities.

Chapter 2. Theoretical body of knowledge

This chapter provides the theoretical background needed in order to understand and design a B2B Integration module for Openbravo ERP. It covers the three main areas of knowledge: explanation and history of **Enterprise Resource Planning** systems, **Supply Chain Management** theory, and some software engineering concepts: **Open Source and Software Development Methodology**.

Enterprise Resource Planning systems are ubiquitous in the industrialized world; it is unheard of any organization that can manage itself without a centralized software system, once a sufficient organization size is reached. Therefore an enhanced B2B collaboration and integration will have to happen within the context of the organization's ERP system. This will usually involve the extension of the capabilities of a given ERP system, whether it comes in the form of an integration of two separate software systems, or a native extension module for an ERP system.

Supply Chain Management is a complex and old subject, of which the study and academic discipline can track be traced back to ancient roman times. Running an organization, independently of its purpose, involves having the right resources, in the right amount, in the right place at a given time to advance in its purpose. It doesn't matter if this organization is an army, where soldiers, weapons, war machines and supplies have to be produced, trained and moved properly, or a farm plantation, in which seeds, crops, farm machines and people is needed at the proper field or warehouse. The latest organizational theories on Supply Chain Management advocate for a close and automated integration of day to day, Business-To-Business activities between customers and vendors.

Software Development methods have varied much since the days of the first mechanical computers. While at some point they could be considered as part of Industrial Management Theories, Software Engineering techniques have long evolved and its implementation is key to developing, updating, maintaining and publishing high quality software nowadays. Furthermore, the evolution of Internet and data networks has provided a new paradigm for software and intellectual property, Free Open Source Software, in which the source code of the software programs is publicly available and can be used or modified freely, and is often developed and maintained by a non-profit community of software developers. Open Source software is by definition attached to some specific Software Development techniques which maximize software availability and quality.

2.1. Enterprise Resource Planning (ERP Systems)

This section tries to define and explain what is an ERP system. As it is some complex business software, there are some characteristics that stand out and make it different from traditional desktop user software, and its implementation and use is not as immediate.

2.1.1. Theoretical Background

Enterprise Resource Planning (ERP) integrates core business areas such as manufacturing, distribution, financials and human resources. ERP is often implemented in companies together with process-oriented organization or Supply Chain Management (SCM). In order to manage the information flow of such structures new IT systems are generated – known as ERP-Systems. IT systems of this kind allow managers from all departments to look vertically and horizontally across the organization to see what others are accomplishing or not.

It attempts to integrate all departments and functions across a company onto a single computer system that can serve all those different department's particular needs. ERP-systems also implement and automate business processes, putting them into a useful format that is standardized across the corporation and between their suppliers and customers.

ERP systems capture data about historical activity, current operations and future plans and organize it into information people can use to help develop business strategies.

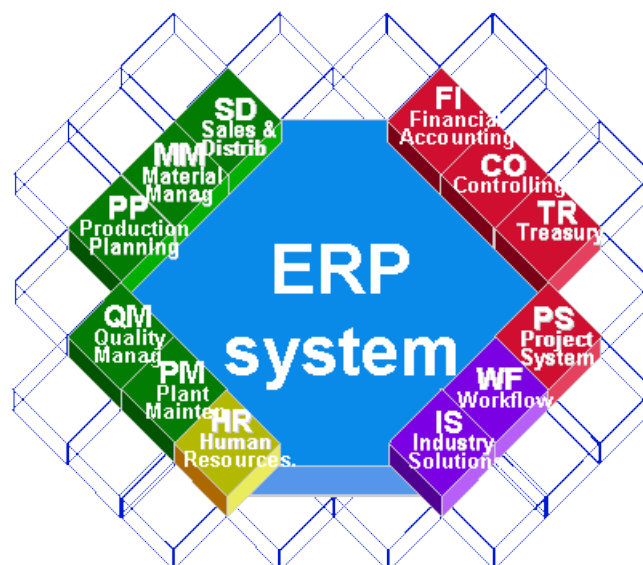


Figure 1 - Example of standard ERP functionality

In the ERP industry, the systems are often referred to as the 4M's. Man, Money, Materials and Machines. This type of system brings all four aspects of business together, giving them a synergistic value. ERP is an enabling technology that can give corporations a strong competitive edge. In addition, this technology is as close to virtual enterprises as business today has ever seen.

Typical Characteristics and Functionality

ERP systems typically include the following characteristics:

- An integrated system that operates in real time (or next to real time), without relying on periodic updates.
- A common database, which supports all applications.
- A consistent look and feel throughout each module.

Functionality may include modules for management some or all of these areas:

- Finance/Accounting
 - General ledger, payables, cash management, fixed assets, receivables, budgeting, consolidation
- Human resources
 - Payroll, training, benefits, 401K, recruiting, diversity management
- Manufacturing
 - Engineering, bill of materials, work orders, scheduling, capacity, workflow management, quality control, cost management, manufacturing process, manufacturing projects, manufacturing flow, activity based costing, product lifecycle management
- Supply chain management
 - Order to cash, inventory, order entry, purchasing, product configurator, supply chain planning, supplier scheduling, inspection of goods, claim processing, commissions
- Project management
 - Costing, billing, time and expense, performance units, activity management
- Customer relationship management
 - Sales and marketing, commissions, service, customer contact, call center support
- Data services
 - Various "self-service" interfaces for customers, suppliers and/or employees
- Access control
 - Management of user privileges for various processes

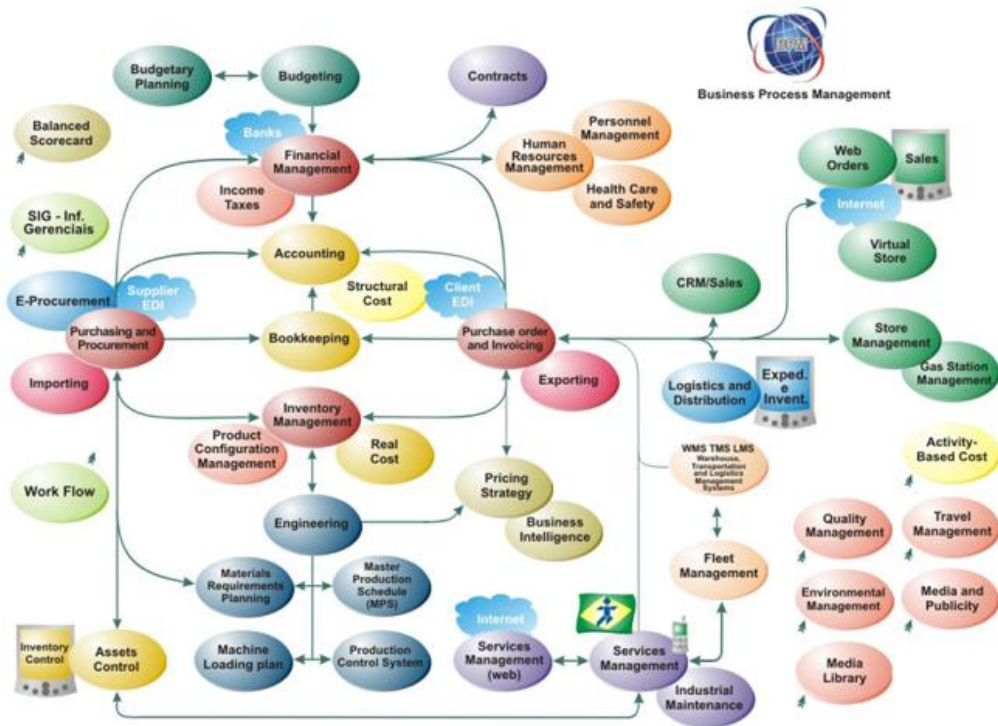


Figure II - Example of integrated Business Processes

Modularity and Extension Capabilities

ERPs are complex systems that try to deliver what the product team developing the software considers to be “Best Practices” of the Industry. This means that the software reflects the vendor's interpretation of the most effective way to perform each business process.

Not every organization interested in deploying an ERP solution will find that every ERP software fits their needs. Even the most complex of ERP systems won't be usable by many organizations since their business needs and workflows may be very uncommon.

ERP systems usually allow modifying or extending their functionality, and may boast a modular architecture to easily extend and re-use solutions. Systems vary in the convenience with which the customer can modify these practices.

Implementation of ERP systems in Organizations

The lifecycle of an ERP implementation begins with the understanding of the critical goals and objectives of the business enterprises in running of the businesses and focusing on the process of streamlining the business processes which are being planned to get integrated technologically.

To start with, while building the Business Case or the Business Blueprint, the first focus that is needed even before the initiation of the ERP project is to analyze what the project will be trying to accomplish and at what cost, using what resource in time, money and man efforts, besides also working on the ROI (Return on Investment) details and a communication plan for all strata of the users involved in the implementation that include the top management, operational management, all the staff members at all levels in an organization who would be the users, stakeholders, sponsors, etc. Another very important initial activity involved in the

ERP implementations is the designing of the information system for a business, which is an important role of an System Enterprise Architect because unless the requirements are not defined in advance and well understood, the result could be fiasco because the goal of the ERP is the application of the technology to what is needed for the business, and which technology tools involved in designing an ERP are used.

Another salient aspect to be considered in ERP project lifecycle is the prototype based training session with the users which ought to involve the actual business enterprise data in a test environment to simulate the ERP package software running the business. This implies the loading of actual operational data such as bills-of-materials, routings, business partners, customers, suppliers, etc. into a test environment for simulation with the ERP software. This approach provides the business user with an exact look and feel of the information as well as the user interfaces that is used in the business operations and system design respectively, and also facilitates to trace the gaps in the business processes. The testing of business processes along with the ERP implementation can plug lot of gaps simultaneously in the development phase.

Normally, a popular approach before ERP implementation is to have a trial run of the existing business processes or pre-defined business processes through the new system in a simulated environment using the actual data of the business enterprise. This process is often referred to as the Conference Room Pilot (CRP) and is normally is designed to be the final verification that the new system is set-up correctly to function in the live business environment.

The normal lifecycle of an ERP Implementation Project shall consist of the following milestones:

1. **Business Process Study.** Regular interactions with the client grow in order to understand the various business processes and the way they are presently carried out.

Setting the objectives of ERP implementation; setting the expectation of the client;

2. **Pre-Implementation Training.** A pre-implementation training, detailing the concepts and features, shall be given to all the end-users at the client site. This shall improve the lead time in collecting the inputs for the preparation of specifications. This activity will take place in most normal ERP implementation life cycle stages.
3. **Requirement Analysis.** The requirements of the client are collected, using the SRS (Software Requirements Specification) form, to further the study and knowledge transfer of the user processes involved.
4. **GAP Analysis.** In relation to Information Technology, this analysis points out the differences or gaps that exist between the standard capabilities of the desired system and the expectations of the client. The deliverable is a GAP Analysis Report.

In order to get maximum benefit out of the ERP processes must be re-engineered and then mapped with the redesigned processes;

5. **BMR (Business Mapping Report).** This follow-up of the GAP Analysis should address the identified solution(s) for the gap areas.

- a. **Master Data preparation and management:** This is perhaps the most ill-treated activity which ultimately leads to delays if not addressed at early stages of the project planning. A team must look after these before and after the project to ensure accuracy in master data as this may lead to erroneous results causing panics after Go Live.
6. **Project Plan.** This milestone shall set the actual plan for execution and roll out for the roles and task allocations.
7. **Installation of Software.** The unmodified installation of the complete software should now be carried out to check the infrastructure preparedness at the client site.
8. **Customization of Forms & Reports.** The client's requirement as to the customization of forms (user interfaces) and reports (existing as well as new requirements) are addressed here.
9. The system must be tested for complete processes and each type of transactions so as to check its robustness.
10. **Migration of Historical Data.** Beyond the opening balance incorporation, if the client desires to bring the historical data, this step will be required to plan the conversion and clean up and preparation of the data for the new environment.
11. **Design of Routines & Workarounds.** Though this is not a milestone, the additional routine and workaround requirements (as identified in the GAP & BMR) shall be addressed through this step.
12. **Setup and Configuration.** The new environment shall be set up based on the SRS and the software shall be configured (on different deployment methods) fully.
13. **Testing Environment.** A testing environment (a simulation of the live environment) shall be created in the new software to enable the end-users to acquaint and equip themselves for beginning in the live environment.
14. **End-User Training.** The next milestone shall be giving training to the end-users on the setup, configuration, and transaction processing and report generation. This shall be the post-implementation training phase.
15. **System Walk-Through.** This milestone requires entry of sample transactions by the end-users in the testing environment.
16. **Go Live.** This is the Sign-Off phase where the project gets implemented.
17. **Post Implementation Support.** An immediate support commitment, subsequent to the implementation, and the detailed modes of support, etc., are given to the client.

2.1.3. History of ERPs

ERP Software has been around from the early development of computers and computer software. Since computers have been invented to increase productivity, they were focused on business needs. First software developed in 1960's was able to manage the demand and ordering. This MRP software was focused only at the need and not at the timing. In the 1970's MRP II could manage the demand, and ordering on a time schedule. This could be used for planning process. SolutionS for accounting management became more and more popular. MRP II systems evolved into ERP that could offer a complete solution for integrated financial applications, cash, people, resources and inventory management.

In 1960s the first ERP was born as a method for planning and scheduling materials for J. I. Case complex manufactured products.

First MRP solutions were expensive, cumbersome and they required a large technical staff to support the mainframe computers. In 1972 in Germany emerged SAP from the work of five engineers from Mannheim. This served the purpose of providing integrated business solutions for market standard.

Lowson Software enterprise started in 1975 by John Cerullo, Bill Lawson and Richard Lawson was the first company who put the bases of pre-packaged enterprise technology solutions that could be a good alternative to what market already offered as customized business software applications.

Material Requirements Planning (MRP) became a fundamental concept in control and management of production in 1976.

Oracle Corporation begins his activity in 1977. The founder of this corporation was Larry Ellison

The Baan Corporation who can provide administrative and financial consulting services started in 1978 by Jan Baan.

The first commercial SQL relational database management system was released on the market in 1979 by Oracle.

In the early 1980s MRP II starts to be a more effective solution than MRP for distribution management activities.

In 1981 Baan starts using UNIX as their main operating system, then a year later Baan delivers its first software product.

The release of MRP II in 1980s on the market led to a new way of coordinating manufacturing processes throughout all the stages from the product planning, purchasing parts, and inventory control and product distribution. Companies could follow better the process of production and distribution thus giving them the ability to better control them.

Then in the 1990s Enterprise Resource Planning (ERP) started to be more complex and use multi module application software. This generation of ERP could be used to improve the performance and internal business processes. This system could integrate the business activities across functional departments for every stage of production. The new ERP software systems could include application modules for supporting human resources, marketing, finance and accounting. Modern ERPs are very complex systems that often are considered by some clients as very hard to use with all their functionality and they need complex setting up of hardware and application software. Every business has its particular needs from such system and that's why a lot of ERPs vendors offer customized solutions.

ERP got its current name, only after evolution. From Inventory Management and control (IMC) of 1960s to Materials Requirement Planning (MRP) of 1970s to Manufacturing Requirements Planning or MRP of 1980s then to Enterprise Resource Planning system (ERP) from 1980 to till date.

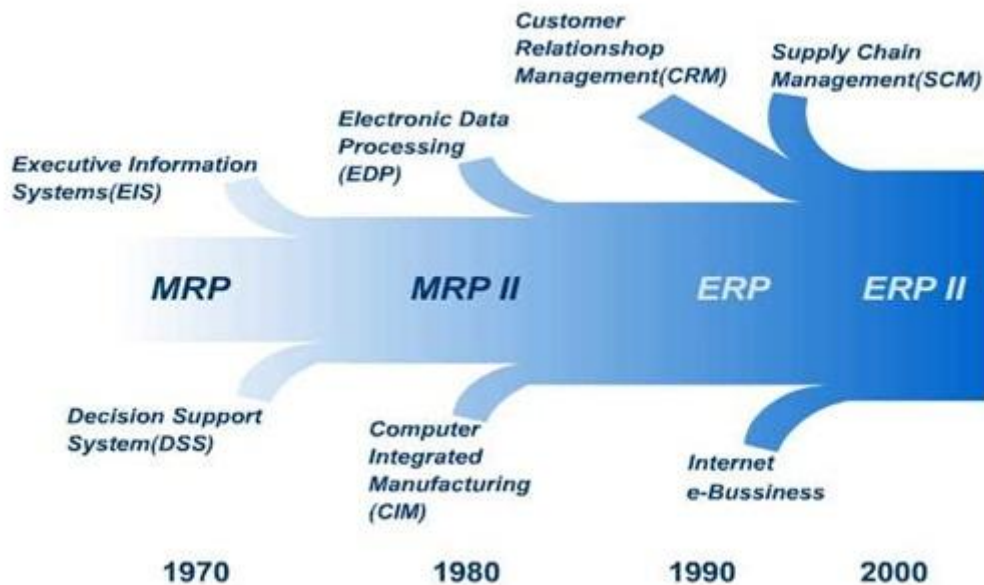


Figure III - Evolution of ERPs timeline

ERP II

Today, ERP is the foundation of businesses domestically and globally. It is used as a management tool and gives organizations a great competitive advantage.

But the consulting firm that created the term a decade ago, Gartner, claimed ERP was dead.

Even though Gartner recognized that the need for an information backbone for an enterprise didn't go away. As e-business became business as usual, sharing accurate real-time information about orders and inventory turned critical to success, not just across an enterprise, business needed to move that information across a supply chain.

Gartner introduced a new term to describe the enterprise systems for the 21st century: ERP II. In 1990 when the ERP term was coined, ERP was enterprise centric with very little awareness of anything going on around it, but ERP II claims that the key is sharing information outside the enterprise.

Role	Traditional ERP was concerned with optimizing an enterprise. Internal optimization, however, will only take you so far. ERP II systems are about optimizing the supply chain through collaboration with trading partners.
Domain	ERP systems focused on manufacturing and distribution. ERP II systems will cross all sectors and segments of business, including service industries, government, and asset-based industries like mining.
Function	As ERP systems cross sectors and segments, they will no longer be able to present all things to all people. ERP II vendors to pick the industries in which they're going to play, and focus on

providing deep functionality for those users.

Process In ERP systems, the processes were focused on the four walls of the enterprise. ERP II systems will connect with trading partners, wherever they might be, to take those processes beyond the boundaries of the enterprise.

Architecture Old ERP systems were monolithic and closed. ERP II systems will be Web-based, open to integrate and interoperate with other systems, and built around modules or components that allow users to choose just the functionality they need.

Data Information in ERP systems is generated and consumed within the enterprise. In an ERP II system, that same information will be available across the supply chain to authorized participants.

Table i - Differences between ERP and ERP II

ERP II systems are not just the backbone of the enterprise. They are also the information link for an enterprise in the supply chain. That's because the business of tomorrow is going to play multiple roles in multiple supply chains, from traditional sources to electronic marketplaces.

The challenge for ERP II is two-fold. First, it's to aggregate and manage the data surrounding all the transactions of an enterprise as accurately as possible in real time. Then, it's to open up the system to make that information available to trading partners.

2.1.4. Openbravo ERP

Openbravo ERP is the leading open-source business solution software for enterprise resource planning functionality for SMEs (Small and Medium Enterprises). It deploys state-of-the art technology stack, great performance and stability, a fully functional and highly productive package, in a competitive-cost solution.



Figure IV - Openbravo ERP, the agile ERP

Openbravo ERP has been downloaded more than 2 million times and official company figures put the implementation rate on about one thousand organizations. It has won several product awards (*Infoworld Bossie Award* in 2009 and 2010, *IBM EMEA Innovative Solutions 2009 Award*, *Open Source Business Awards 2008*) and is considered as a real world alternative to proprietary ERP software packages.

The core product is licensed under the Openbravo Public License (OBPL), based on the Mozilla Public License (MPL). The OBPL has been derived from the MPL by substituting references to the Openbravo Software for that of the Mozilla Foundation.

The OBPL also includes a branding clause, whereby anyone redistributing the code, in original or modified form, must maintain the Openbravo logo and link on start-up or login screens and on the user interface. While this may seem an additional obligation on developers, Openbravo believes it places an undue burden on them.

MPL is recognized and accepted by the Open Source community and it is suitable for software that includes other open source technologies licensed under various non-copyleft licenses such as Apache Software License or LGPL.

Functionality

Openbravo ERP's functionality is defined around 3 core areas (*Procurement Management, Warehouse Management, Sales Management*), 2 extended areas (*Production Management and Project and Service Management*), and all tied together by the main *Master Data Management* area and influencing into *Financial Management & Accounting* area. Finally, *Business Intelligence (BI)* functionality is offered to leverage and use the data stored in the ERP to make business decisions.

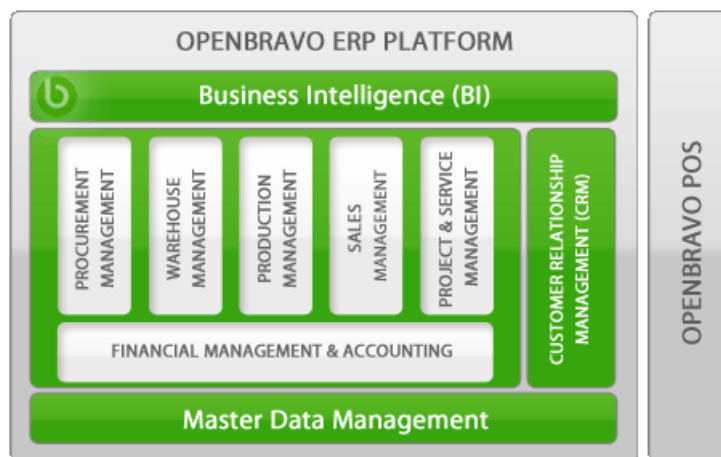


Figure V - Openbravo ERP Functionality Overview

Master Data Management. *Products, components, bills of materials, customers, vendors, employees etc.*

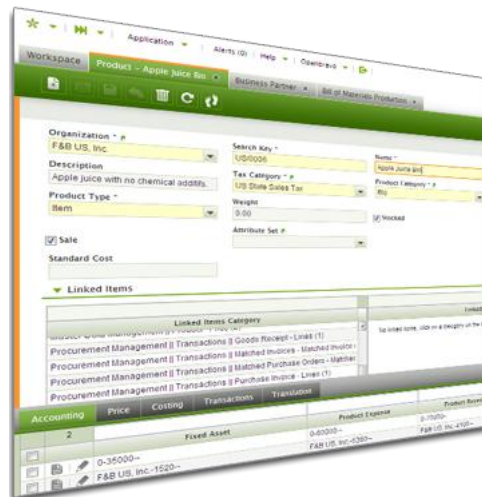


Figure VI - Openbravo ERP Functionality: Master Data Management

The correct management of the master data of an organization company (products, customers, vendors, etc.) is a fundamental aspect to be sure of the coherence and tracking of processes. Maintaining exclusive coding, avoiding duplications, and sharing the relevant information among all areas of your company are some of the challenges faced today by all types and sizes of organizations. Openbravo ERP helps the Organization to organize and centralize the key data of the organization, easing the rapid and easy flow of information among all areas implicated in different company processes.

- Products and Components
 - Product categories.
 - Product indexes: product types (item, service, cost), with specialized management for each. Particular warehouse management definition for each product (management of stock, tracking ability). Characteristics. Product images.
 - Measurement units, conversion between units, variable weight units.
 - Materials list (products made up of others).
 - Vendors by product.
 - Price lists. Definitions of rates with differentiation (for example, from sales rates to purchasing rates). Automatic process of generation of rates.
 - Rates. Price rates, applicable prices, price limits, specialized rules for applying prices to purchasing and sales.
 - Freight categories.
 - Transportation services (integrated with business partners).
 - Substitute products.
- Business Partners
 - Customers, vendors, employees. Functions characterized by internal use (delivery/receipt of material, invoicing, collection, social direction, others). Contacts associated with direction. Third party grouping. Areas of interest (for commercial analysis).
 - Customers. Sales rates. Invoicing method (immediate, delivery notes served, order completely delivered, periodical). Payment forms and terms (conditions

of payment). Printing format and number of documents discernable by customer. Permitted risk (credit).

- Vendors. Prices and discounts. Form and period of payment (conditions of payment).
- Employees. Related with customer agent.
- Groups of business partners (segments or categories).
- Payment Terms (expiration dates, fixed payment days, working days, multiple expiration dates).
- Periodic invoicing calendars (monthly, fortnightly, weekly), with cut-off dates for each case. Possibility of mixed use calendars.
- Purchasing and volume sales discounts. Relation of articles, scalable discounts.
- Sales channels (sales representatives, telesales).
- Areas of interest.
- Third party activity reports.

Financial Management and Accounting. *Chart of accounts, accounts, budgets, taxes, general accounting, accounts payable, accounts receivable, bank accounting, balance sheet, P&L, fixed assets, etc.*

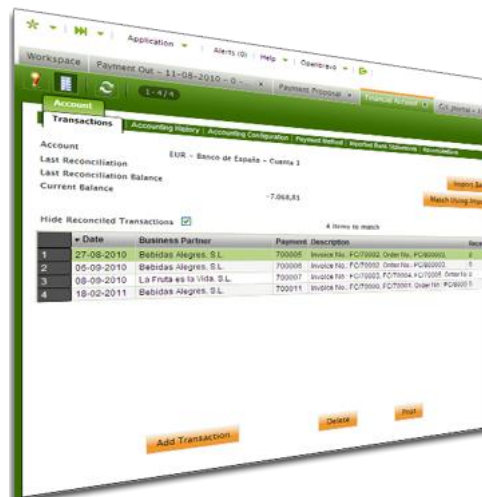


Figure VII - Openbravo ERP Functionality: Financial Management and Accounting

The financial management and accounting functionalities provided by Openbravo ERP are designed to minimize manual data input on behalf of the user, thereby freeing them from tedious, routine tasks and allowing greater focus on other, more value added tasks. This increase in productivity is due to the financial department acting as collector of all the relevant actions generated from the other management departments. This occurs in such a way that these have an automatic reflection in the general accounting, in the accounts receivable and accounts payable as soon as they are produced.

- Accounting Solution
 - Default charts of accounts
 - Definition of charts of accounts
 - Fiscal year and year to year management

- Budgeting
- Tax categories
- Tax brackets. Determining flexible taxes based on regions, products, or business partners
- Linked accounting. Direct navigation from accounting entries to documents and vice versa
- Manual entries.
- Journal entries
- Trial balance
- General ledger
- P&L statements (income statements)
- Balance sheets
- General accounting plan statements
- Standard OFX module
- Accounting tabs for transactions module
- Custom accounting
- Receivables and Payables
 - Generation of debt payments (from invoicing)
 - Issue of debt payments
 - Management (cancellation, merging and division) of accounting documents. Remittances (according to bank records)
 - Issue of cash accounts. Multi-cash account
 - Cash account journal (cash audit). Cash account entries by type Expenses, Receipts, Balance, Deposits, Bills, Order (for cash payment on delivery note: possibility of recovering expenses before invoicing. Automatic generation of entries for cash payments and payments on delivery note
 - Bank statements. Portfolio issues selection assistant
 - Manual liquidations. Other expenses (payroll, taxes, etc.)
 - Cash account reports, bank, debt payments by position
 - Advanced Payables and Receivables Management module
 - Credit management - Dunning module
- Assets
 - Definition of the assets, fixed assets, acquisition price, and accounting valuation
 - Temporary or percentage based amortization
 - Amortization plans
 - Advanced asset management module
- Internationalization
 - Support for multi-currency
 - Support for multi-schema accounting, which enables the same transaction to be accounted in different rules, different charts of accounts, in different base currencies and with a different accounting calendar
 - Support for international bank account numbers
 - Support for multiple languages, defined at user level

Procurement Management. Rates, purchase orders, goods receipts, invoice registration and accounting, purchase planning, etc.

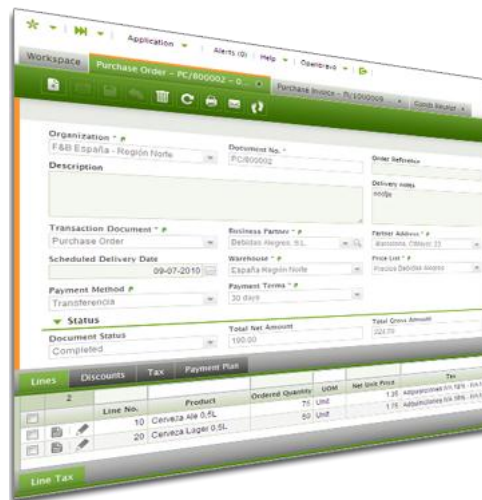


Figure VIII - Openbravo ERP Functionality: Procurement Management

Openbravo's procurement module is designed to control the supply chain, with streamlined functionality to minimize acquisition costs and optimize collaboration with trading partners. Fully integrated with Financial Management & Accounting and Warehousing, this module allows to:

- Maximize cash flow by standardizing and automating AP processes.
- Reduce inventory and distribution costs, while maintaining high service levels.
- Share inventory and usage information with suppliers through secure role-based browser access, and easy web services integration.
- Control employee purchases through centralized web-based requisition management.
- Integrate warehouse processes to reduce waste, errors, and cycle times.
- Respond quickly to inventory shortages, supply changes, and shipment delays.

Openbravo ERP's end-to-end handling of the flow of supply guarantees the integrity, tracking, and consistency of the process. Each document in the supply process is based on the information contained in the previous document, so that repetitive introduction of data and human errors are avoided. In this way, it is possible to navigate through different documents that conform to a determined flow (order, goods receipt, invoice, payment) and know in real time the state of any given order (pending, delivered, partially delivered, invoiced, etc). The natural integration of this process with accounting guarantees that the finance department always has up to date and reliable data at its disposal. Key functionality includes:

- Purchase planning, based on production necessities, keeping in mind minimum stock levels, goods receipts dates, and pending requests.
- Support for purchase requisitions for centralized purchase management.

- Purchase orders. Rates applications: prices, discounts and price limit controls. Warehouse control of goods pending receipt. Correction of orders. Creation of purchase orders from sales orders. Refunds to the vendor.
- Receipts. Automatic creation from pending order lines. Automation of incoming goods (location according to priority). Refunds to the vendor (according to stocks). Cancellation of delivery notes.
- Purchase invoices. Application of rates: prices, discounts and control of price limit. Automatic creation from order lines or delivery note lines pending invoicing. Invoicing of goods processed in consignment. order. Cancellation of invoice (leaving pending for invoicing the associated documents).
- Relationship between orders, delivery notes and invoices.
- Expense Invoices.
- Massive printing of documents.
- Purchasing order reports, vendor invoices.

Warehouse Management. *Warehouses and bins, warehouse units, lots, serial numbers, packages, labels, receipts and deliveries, movements between warehouses, inventories, stock valuation, transport, etc.*

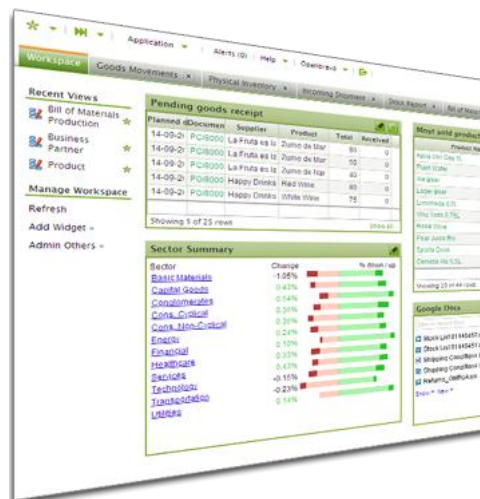


Figure IX - Openbravo ERP Functionality: Warehouse Management

The warehouse management processes built into Openbravo ERP allow the inventory in the organization to always be up to date and correctly valued. The possibility of defining the warehouse structure of an organization to unit level (storage bins) facilitates the exact localization of the stock at any time. Additionally, the capacity for managing product lots and the possibility of using serial numbers assure compliance with the tracking requirements imposed by the majority of industries.

- Warehouses and storage bins (multiple warehouse use available).
- Stock products in multiple units (for example in kilograms and boxes).
- Personalized product attributes in the warehouse (color, size, quality description, etc.).
- Lot and serial numbers.
- Management of bundles in warehouses.

- Restocking control.
- Traceability configurable by product.
- Movement among warehouses.
- Picking strategies (according to stock, with rules of priority by expiry, location, etc.).
- Physical inventory. Inventory planning. Continuous inventory.
- Reports of movements, tracking, stock, arrivals/departures, expiry, inventories, locations, etc. Personalized reports.

Sales Management. *Prices, rates, varying quantity sales orders, shipments, invoicing, volume discounts, commissions, CRM, etc.*

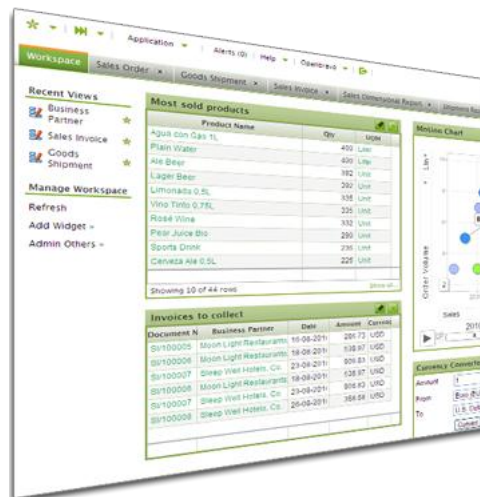


Figure X - Openbravo ERP Functionality: Sales Management

The functionality of Openbravo ERP in the Sales Management module is designed with the objective of allowing maximum flexibility and adaptability in its execution, needed in any commercial process. It is possible to link documents (orders, shipments, invoices) in any order that the company requires or even disregard any one of these if is not necessary. All this is achieved without sacrificing the coherence and integrity of information and safekeeping the tracking of processes.

- Sales areas.
- Sales orders. Direct sales. Pre-sales. Telesales. Rate applications: prices, discounts and price limit controls. Reserves of goods in warehouse for unprocessed orders. Notification of customer risks. Correction of orders.
- Types of order documents: estimations (with and without reserves of goods), standards, warehouses (automatic delivery note generation), Points of sale (automatic generation of delivery note and invoice).
- Delivery note. Automatic creation from pending order lines. Automation of the outgoing orders (according to stocks, with rules of priority by expiry, location, etc). Cancellation of delivery notes.
- Automatic generation of delivery notes.
- Invoicing process. For all types of invoicing: immediate, delivered goods, order completely delivered. Periodically, (weekly, fortnightly, monthly).

- Invoice registration. Rate applications: prices, discounts and price limit controls. Automatic creation from order lines or delivery note lines of delivery notes pending of being invoiced. Notification of customer risks. Invoice cancellation (leaving associated documents pending invoice).
- Massive printing of documents (orders, delivery notes, invoices), with individual selection criteria for the user.
- Possibility of the creation of documents in any order and to disregard documents not required (Order-Delivery note-Invoice; Order-Invoice-Delivery note; Delivery note-Invoice; Invoice).
- Commissions.
- Reports of orders, sales order supplied, delivery notes, invoices, not invoiced orders, details of invoicing.
- Integrated with order capture systems by (palm and pocket PC).
- Standardized customer information (360° vision).
- Management of requests. Integration with E-mail.

Project and Service Management. *Projects, phases, tasks, resources, budget, expenses and expense invoicing, related purchases, etc.*

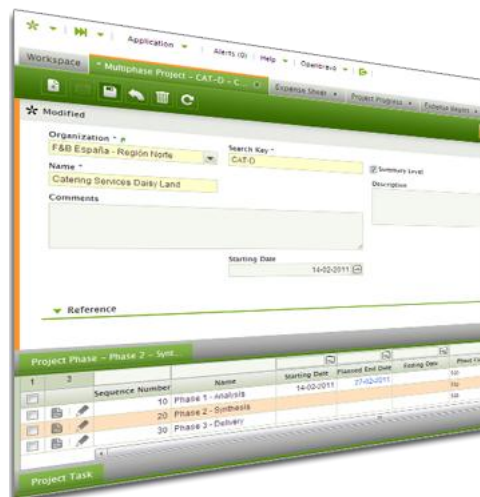


Figure XI - Openbravo ERP Functionality: Project and Service Management

This functionality is oriented towards companies whose activities are based on the delivery of projects and services. With relationship to projects, Openbravo ERP allows for the management of budgets, phases, tasks, expenses and purchases related with each individual project. These projects may be related to monitoring construction projects or even sending out and sales and purchase related requests. The service component permits companies to define services and resources and control all activities. These activities may or may not be billable, for internal or external customers, and be monitored for incurred expenses at a detailed level.

- Project types, phases and tasks.
- Expenses associated with a project.
- Historical salary categories associated to project costs.
- Made-to-order projects and generation of sales orders from templates.

- Construction projects. Invoice to origin (by project).
- Rates by project.
- Budget report. Tracking of actions regarding budget estimates.
- Generation of purchase orders.
- Project reports.
- Resources.
- Register of services.
- Internal expenses.
- Invoicing of expenses.
- Invoicing of services.
- Levels of service.
- Activities report.

Production Management. *Plant structure, production plans, BOM's, MRP, manufacturing orders, job reports, costs of production, work incidences, preventive maintenance types, etc.*

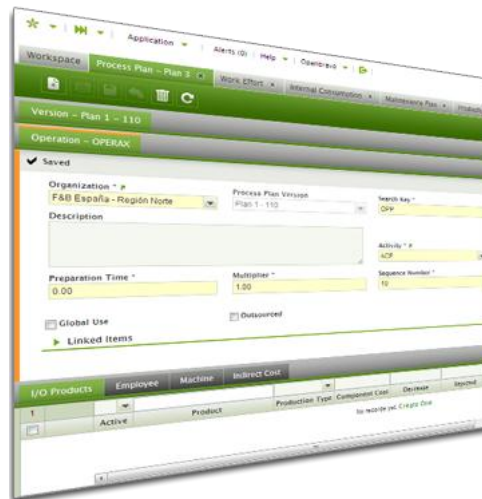


Figure XII - Openbravo ERP Functionality: Production Management

The production functions and plant management in Openbravo ERP allow a complete shaping of the productive structure of each organization (sections, cost centers and work centers) as well as the relevant data for production: production plans (operation sequences), and products used to make one another. Currently, the functionality provided by Openbravo ERP is oriented towards covering the usual necessities of a discrete production environment: production planning and requests related to procurement using MRP, creation of manufacturing orders, job reports (notification of times and consumption), calculating costs of production, notification of job incidents and maintenance reports.

- Plant structures.
- HFG's (Homogeneous Functional Groups) or Cost Centers.
- Work centers and machines.
- Materials Requirement Planning (MRP) keeping in mind client requests, existing clients, stock levels, and minimum order quantities.
- Production plans, with multiple arriving products and multiple departing products.

- Production orders.
- Sequence creation and products for each order phase.
- Confirmations with data relating to the production plan and pre-filled sequence.
- Calculation of production costs with the possibility of adding indirect costs.
- Work incidences.
- Types of equipment and management of each piece of equipment.
- Preventive maintenance and maintenance types.

Business Intelligence (BI). *Reporting, multidimensional analysis (OLAP), balanced scorecards.*

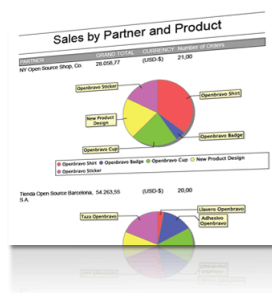


Figure XIII - Openbravo ERP Functionality: Business Intelligence

Nowadays, business organizations handle a great deal of data in the practice of their business activities. This does not necessarily mean that they have available to them the necessary information for the management of their enterprise. The Business Intelligence component of Openbravo ERP, integrated into the management system, shows relevant information in a visible manner. The predefined balanced scorecard will allow verifying, through the monitoring of a series of key indicators, if the defined strategy at a high level is being correctly implemented in an organization.

- Integrated with application management.
- Reports definable by user.
- Pre-established dimensions (third party, product, product category, product category, project, campaign, etc.) and dimensions established by use.
- Predefined balanced scorecard.

Platform Functionality. There is a whole set of functionality designed to improve day to day productivity, on the very design of the user experience.

- *Role-based workspaces.* A user-configurable "home page" with extensible widgets that integrate data from Openbravo or other enterprise applications, such as CRM, BI, or content management, or from productivity applications like email, calendar, Twitter or Facebook.
- *Multi-tab User Interface.* Empower multi-tasking users with the ability to work and process independently in multiple tabs. As a row is selected, the application tab is updated with information about that row—making this feature much more useful than using multiple browser tabs.

- *In-Grid Editing.* Ability to edit data directly while in grid view, with automatic saving of the data as the user moves to another row.
- *Alerts.* Notifications can be programmed to alert the user whenever a certain condition - for example when inventory is low, a customer has overdue payments, etc. - is met. These Alerts are then made clearly visible to user as soon as they log in to the application.
- *Linked Items.* From any record in the application, users can access any other record in the application that is linked to it, as long as their permissions allow for it. Finding related invoices, contacts or any specific shipping receipt can be done from any record in the ERP.
- *Export Data, Attachments.* Export a single file or a collection of files to Excel, CSV, or PDF directly from the application. Exported files can also be attached to any file in the application for easy retrieval and management.
- *Keyboard shortcuts and Keyboard navigation.*
- *Audit Trail.* Any record in the system can be audited and traced to the user who created it, or the latest user to edit it.

Technology and User Interface

Openbravo ERP has a native web-based architecture, based on the server-client architecture. The software runs in a remote machine (server) and its users access the system by connecting to the server and opening sessions through a compatible web browser (client).

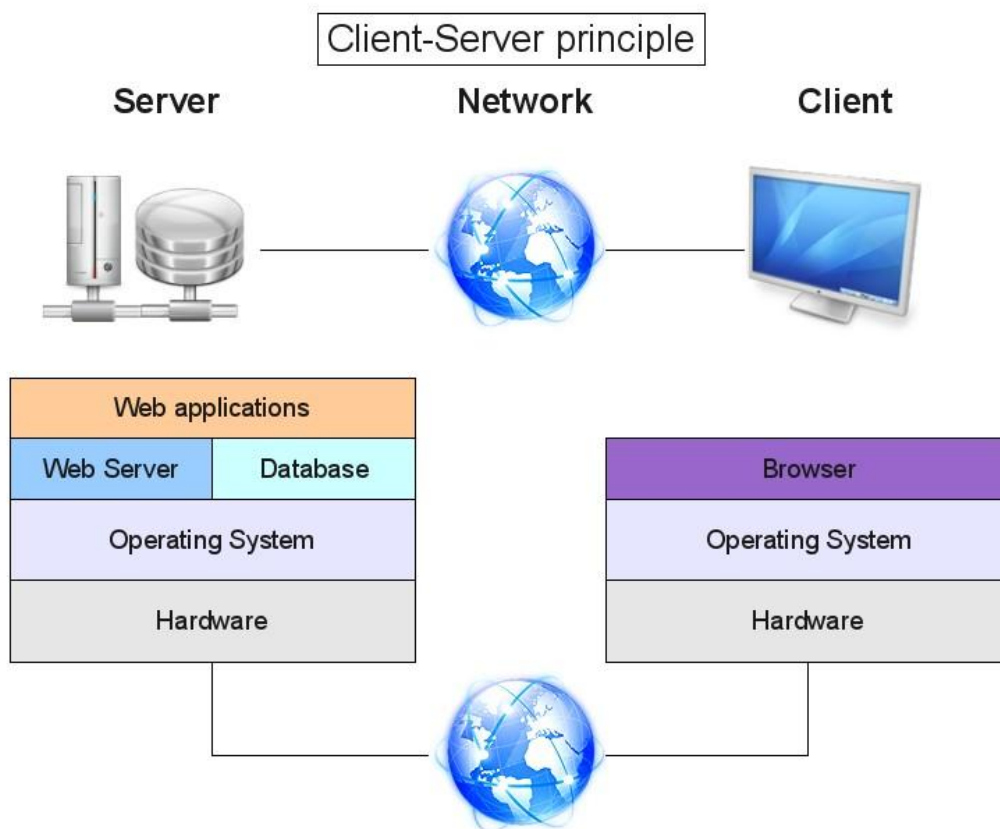


Figure XIV - Client-Server Paradigm

Openbravo ERP operating environment consists in one or more Database Servers, which store the data, along with an Application Server that responds to user petitions. These petitions are relied through a Web Server and a Web Browser.

Users access Openbravo ERP by typing an appropriate URL in their browser to load the Login Screen, and after introducing their Username and Password, they will be presented with their Default Workspace.

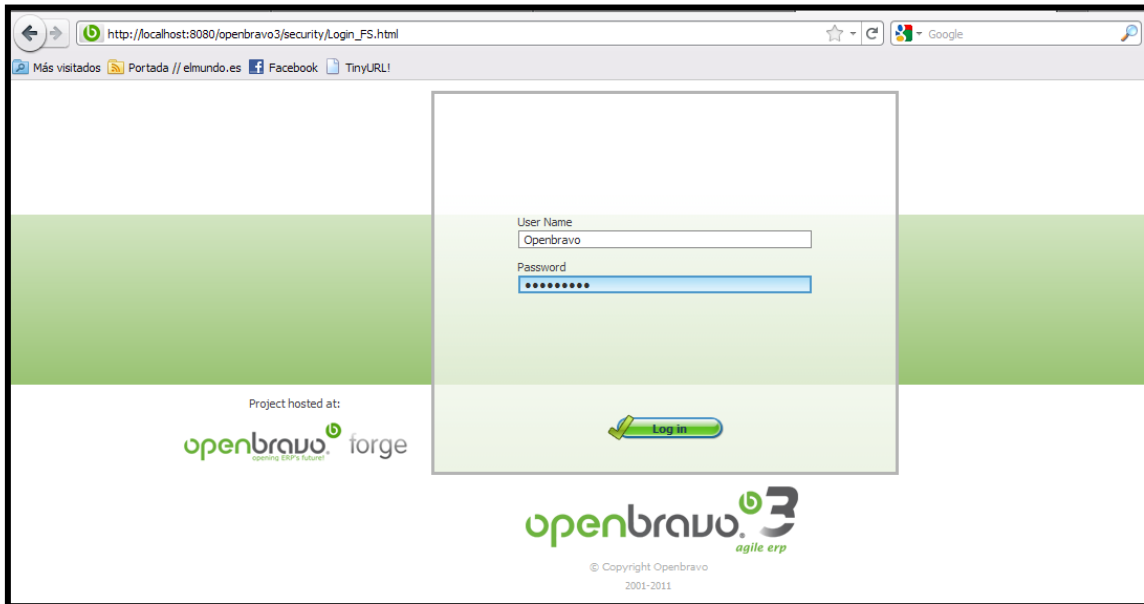


Figure XV - Login Screen

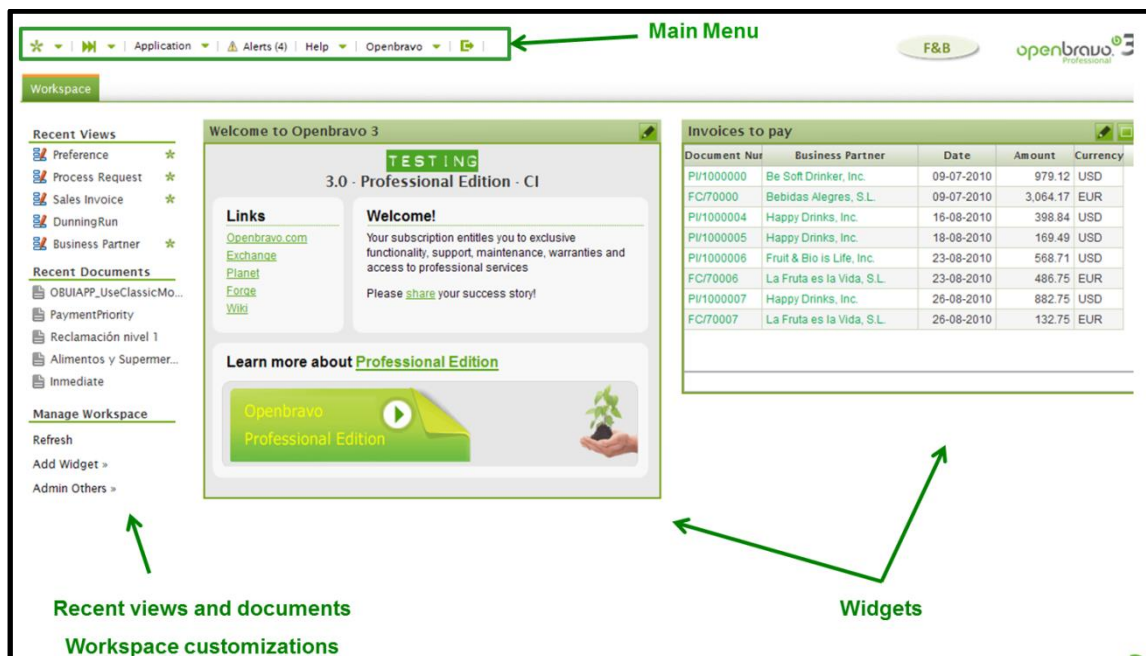


Figure XVI - Default Workspace example

Menu and access privileges depend on the role used to log in to the application.

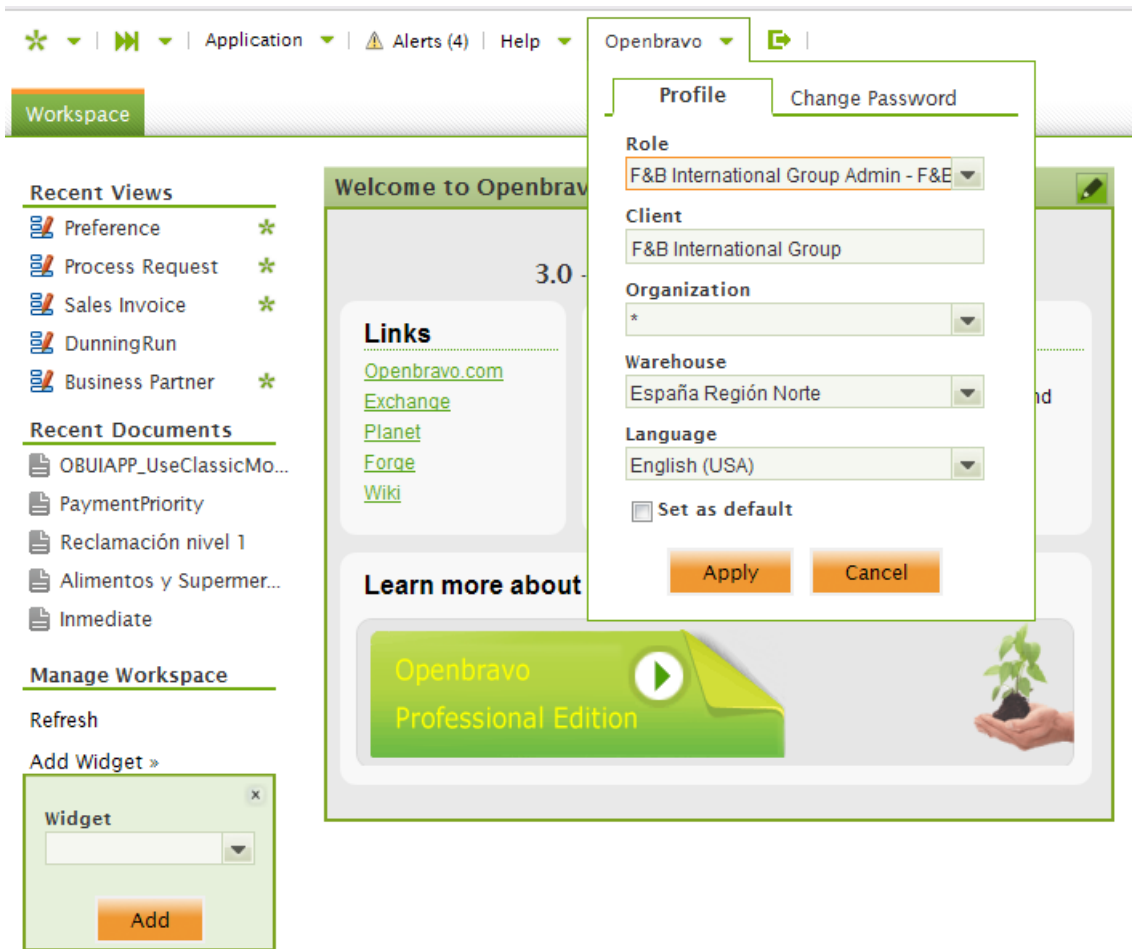


Figure XVII - Openbravo ERP Role selection tool

Access to processes, data and reports of the ERP can be done using the Create New, Quick Launch or the Application Menu buttons. When opened, a new tab is created in the workspace.

Workspace Sales Dimensional Report x Pending Goods Receipts x

Business Partner Organization F&B US, Inc.

From Order Date To Order Date

Document No.

Happy Drinks, Inc.		Lines 2	Reception date	<input type="text" value="31"/>
Document No. PO/800012			Date 07-09-2010	
<input type="checkbox"/> Red Wine		80.00	80.00	<input type="text"/>
<input type="checkbox"/> White Wine		75.00	75.00	<input type="text"/>

Fruit & Bio is Life, Inc.		Lines 3	Reception date	<input type="text" value="31"/>
Document No. PO/800014			Date 12-09-2010	
<input type="checkbox"/> Apple Juice Bio		90.00	90.00	<input type="text"/>
<input type="checkbox"/> Orange Juice Bio		80.00	80.00	<input type="text"/>
<input type="checkbox"/> Pineapple Juice Bio		50.00	50.00	<input type="text"/>

Be Soft Drinker, Inc.		Lines 5	Reception date	<input type="text" value="31"/>
Document No. PO/800013			Date 09-09-2010	
<input type="checkbox"/> Diet Soda		40.00	40.00	<input type="text"/>
<input type="checkbox"/> Energy Drink		50.00	50.00	<input type="text"/>

Figure XVIII - Openbravo ERP tabbed navigation example

Navigation between tabs is similar to tabbed navigation in browsers. Most of the data structures representing documents and actions (Invoices, Products, etc.) can be seen in what are so-called “Windows”, and are opened as new tabs. The records are viewed and filtered in the Grid View, and individual records are opened in Form View. Both views can be mixed within a single tab.

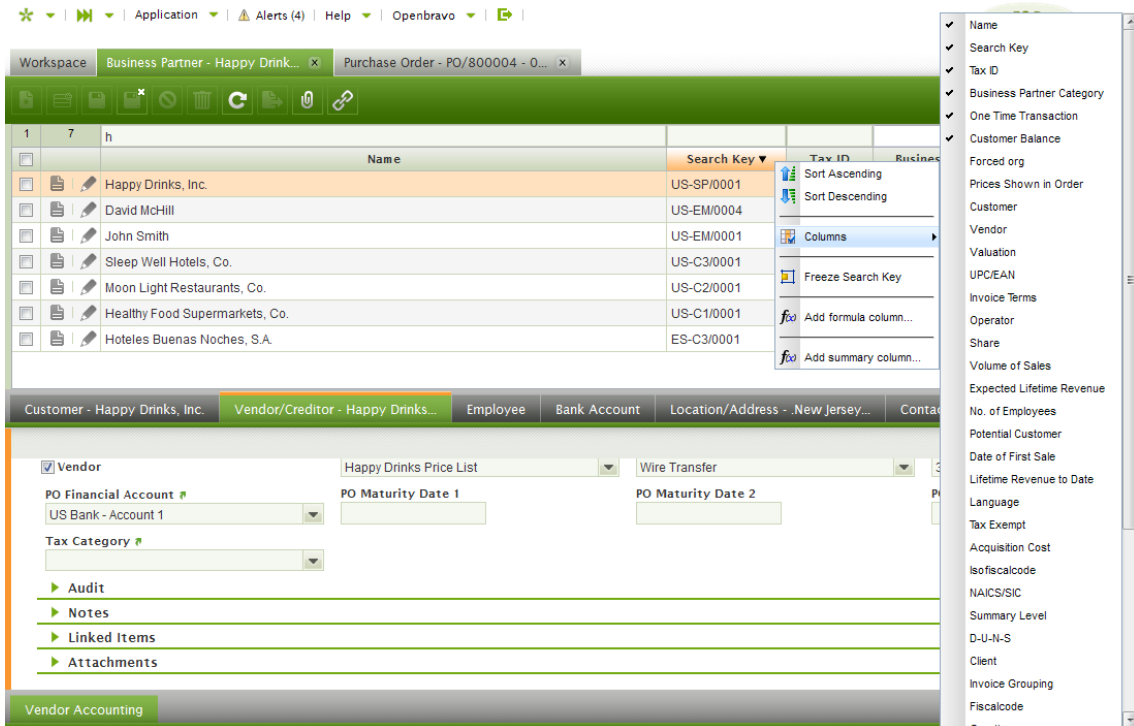


Figure XIX - Openbravo ERP grid view and form view. Filtering

Openbravo ERP is built using open standards, around a unique combination of well-proven MVC and MDD development frameworks, executed by Openbravo's WAD engine.

MVC is a proven web applications development framework, which helps to decouple the database, user interface elements, and business logic. The separation of these elements into different files results in a more structured code, facilitating development and maintenance.

MDD is a software design approach that relies on metadata stored in a dictionary to model the behavior of the application. This results in a drastic reduction in manual coding and fewer bugs, allowing business experts with little coding experience to configure the application to suit the needs of each enterprise.

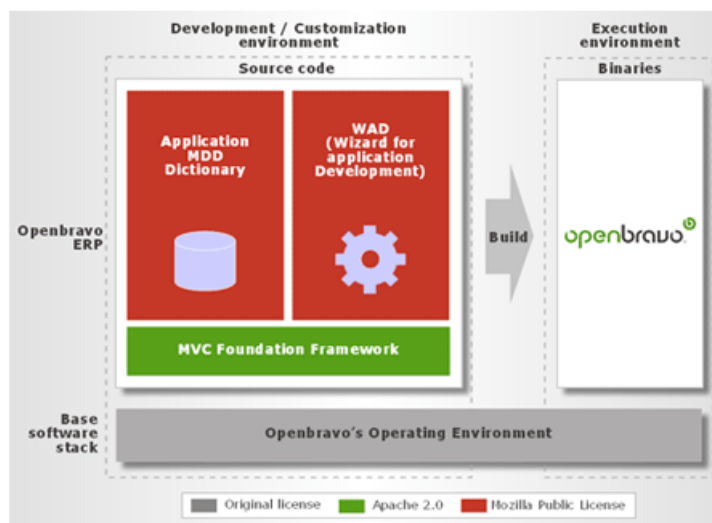


Figure XX - Operating Environment of Openbravo ERP

- WAD (Wizard for Application Development). The engine, built by Openbravo, automatically generates the application binaries from the MDD dictionary. The files generated by WAD are compliant with the MVC standard.
- Application MDD Dictionary. Stores the metadata which describes each element of the application and its behavior.
- MVC Foundation Framework. A set of sturdy programming utilities, either selected from the best open source candidates available or built by Openbravo when no candidates are available. These utilities facilitate web-based MVC application development.

Openbravo ERP uses modern but proven technologies to meet the strict performance and scalability requirements of enterprise grade environments:

- Java and Javascript
- SQL and PL/SQL
- XML
- XHTML

Openbravo also leverages on a number of recognized open source frameworks for a more efficient development process. Composed of well-known third party applications such as Apache Http Server and Tomcat, and a PostgreSQL or Oracle database, that can be installed in a multitude of Operating Systems, including GNU/Linux or Microsoft Windows.

The Data Access Layer is another key component of Openbravo technology. The goal of the DAL development is implement business logic in Java. It provides the following functionality:

- Type safe querying and retrieval of business objects from the database.
- A convenient API to update or create new data in the database.
- A type safe interface to update information of a business object, increased productivity by making the properties of a business object directly visible through getters and setters (in the IDE).
- Transaction and context handling.
- Security and validation checking.
- Automatically maps new entries in the Application Dictionary to database tables and columns.
- Generates Java class business objects (and their associations) on the basis of the Application Dictionary model.

The Data Access Layer is a distinctive feature of Openbravo ERP. The image below shows the envisioned architecture for the data access layer in this and following releases of Openbravo ERP.

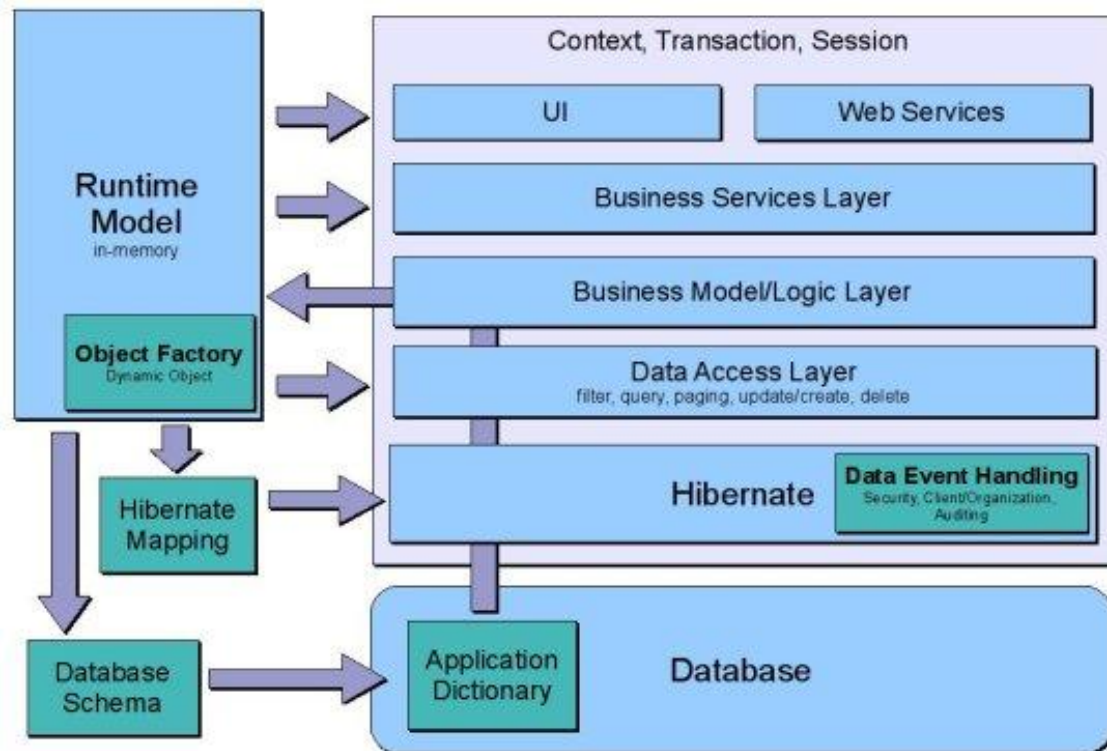


Figure XXI - Data Access Layer diagram

This architecture is partially implemented in 2.50 and will be extended and completed in following releases:

- Runtime model (2.50): the runtime model is the main driver for generating the business objects and the Hibernate mapping. It's also used extensively in security, export/import and in webservices implementations.
- Hibernate Mapping (2.50): from the runtime model the DAL (during initialization) generates a Hibernate mapping. This Hibernate mapping is used to initialize Hibernate.
- Database Schema: the runtime model (actually the application dictionary) can be used to update the database schema. In 2.50 this is not available as part of the DAL but as part of the DBSourceManager product.
- Data Access Layer (2.50): the Data Access Layer (DAL) provides an API to store, query and remove business objects from the database.
- Business Model/Logic Layer (some examples in 2.50): the business model/logic layer contains the implementation of the business processes.
- Business Services (not in 2.50, except for REST webservice): the service layer exposes the business logic to the outside world. In 2.50 this layer contains the REST Webservice provided by the DAL.

The complete architecture runs inside of a context which provides security and transaction handling.

Openbravo ERP automatically generates an XML mapping of all the database structure. Making use of this, a whole set of REST Webservices are created and always updated, so that other

systems and machine can connect to Openbravo ERP and perform CRUD operations over the database remotely (Create, Retrieve, Update and Delete).

Modularity and Customization

Openbravo ERP has a modular architecture that allows for a very easy extension and implementation of new functionality.

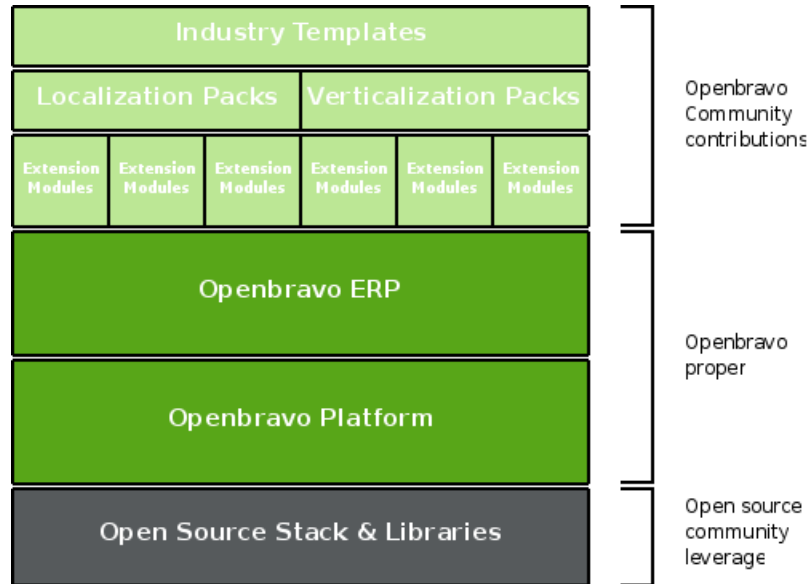


Figure XXII - Different types of Openbravo ERP modules

Openbravo ERP itself can be considered the main “module” inside the Openbravo Platform. This platform could be used to create and develop almost any kind of open-source web-based solution, not necessarily an ERP.

Over Openbravo ERP, extension modules can be installed. These modules contain the same technology as the main, core Openbravo ERP module. Extension modules can be distributed together in Module Packs. And finally, Industry Templates are packs which also contain customization of the core Openbravo ERP experience, usually aimed at specific industries or business areas, and can completely transform how Openbravo ERP is used.

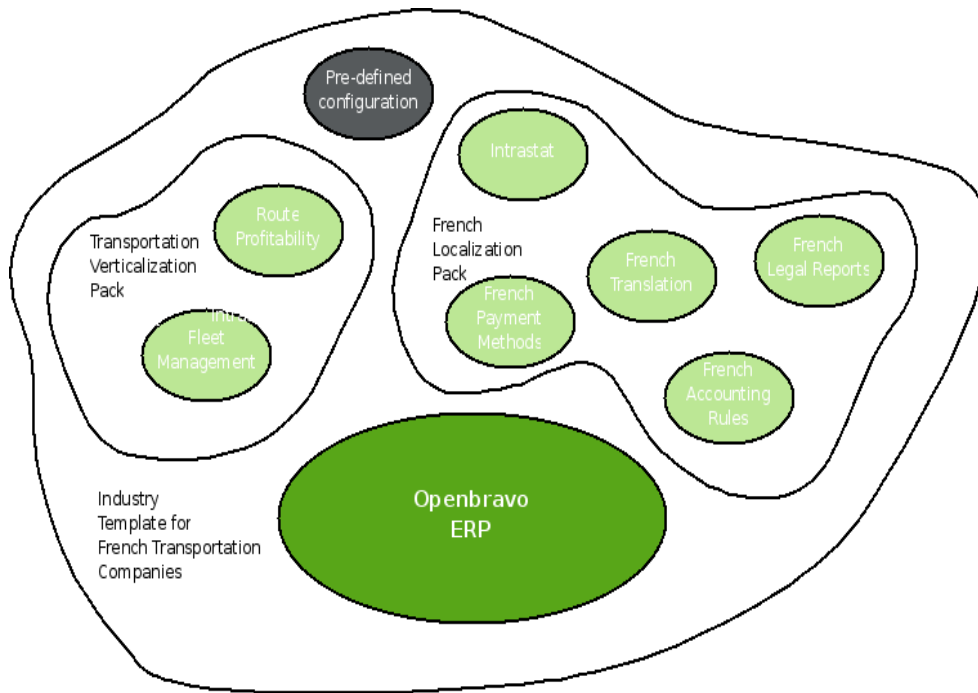


Figure XXIII - Example of combinations of different module types

The Central Repository is a platform used to manage, exchange and share modules developed over Openbravo ERP. It can be accessed at <http://forge.openbravo.com/>. Further information on this platform, related to the development and maintenance of the B2B Integration Module can be found in Chapter 7 of this text.

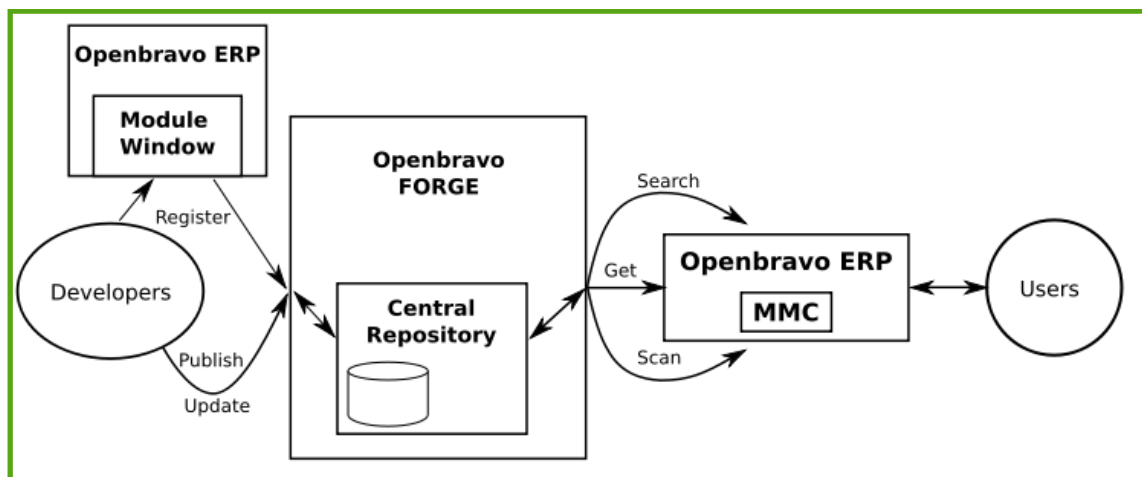


Figure XXIV - Openbravo's Central Module Repository

From any Openbravo ERP instance connected to the internet, new modules can be downloaded and installed from the application at the Module Management Console, as well as updating them.

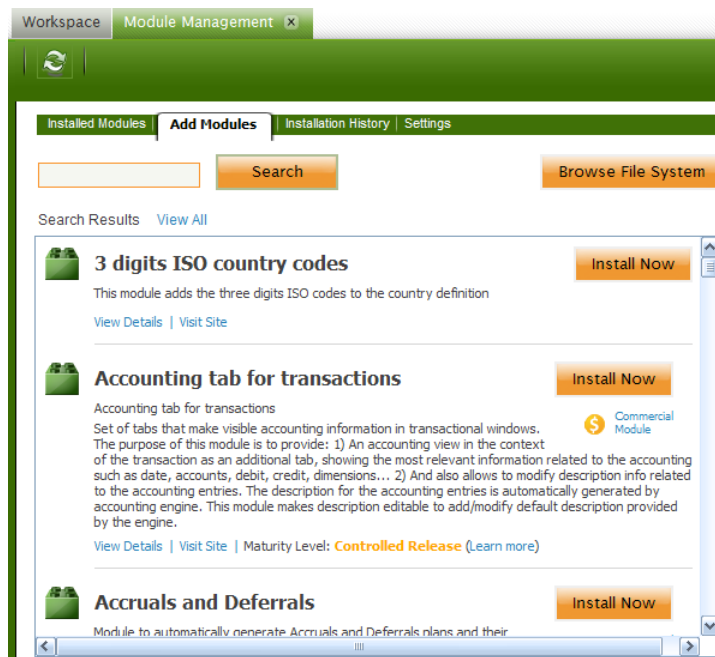


Figure XXV - Module Management Console

Module commercialization occurs at Openbravo Exchange. The Openbravo Exchange is the marketplace to find solutions for Openbravo ERP. There is a growing list of production-grade apps and services that extend the use of Openbravo Professional Edition.

- Extension modules and packs seamlessly provide additional functionality for Openbravo Professional Edition.
- Connectors provide integration between Openbravo Professional Edition and other solutions.
- Localizations include functionalities to adapt to local needs and may include language packs, Chart of Accounts and additional, country-specific functionality.
- Solutions contain packaged modules that work together to meet the needs of a particular horizontal functional area, industry vertical, or customer segment.
- Tools enhance the implementation and maintenance of Openbravo Professional Edition.

Development Tools

Openbravo ERP platform comes with different development tools for extending or customizing the core functionality. Some of them are included in the package while other ones are separate software solutions, always open source. All of the modules developed and published both in Openbravo Forge and Openbravo Exchange have been developed with these development tools.

Development Tools provided by the platform are:

- **Application Dictionary.** It's a tool to define the metadata referring Openbravo's platform components. It includes everything such as tables, windows, views, selectors, dropdown lists, links and images. Allows for rapid iterative development and prototyping. Functional developers can define and generate large percentage of a

system using only the application dictionary, while admitting easy maintenance and standard look and feel of the core Openbravo ERP features.

- **Datasets.** Datasets allow to export the sets of data from different tables in one step. This is especially useful to manage and distribute the module along with reference data, for instance tax rates, regions or default data in a new table(s) added by a module.
- **Alerts.** Alerts are non-intrusive informational messages to individual users or groups (roles) about anything within the system. One can define unlimited number of alerts for various statuses, errors, informative purposes, reminder etc. Some examples are:
 - Errors in the application dictionary that the System Administrator needs to be alerted about (e.g. a table without a an identifier)
 - Errors in master data (e.g. a Business partner without an address)
 - Notifications about critical situations (e.g. overdue payments)
- **Widgets** are the elements which create the My Openbravo main screen. There are three types depending on the superclass inherited
 - HTML Widget. Pieces of html code can be included in a widget.
 - URL Widget. Link to other outbound webs.
 - Query/List Widget. Shows information from the database. HQL clause determines the information shown.
- **Stored Procedures, Java Processes, and Background processes** are pieces of Java or SQL code that can be arbitrarily executed against the operating environment. They can be embedded to buttons, or can be triggered automatically when completing an activity. They are allowed to do permanent changes to the database.
- **Reports** can be included into Openbravo to export data in a conclusive fashion in PDF, XLS and CSV formats.

External Development Tools include:

- *Eclipse IDE* as an integrated development environment. Used mainly as a Java coder, it extends itself as a source code repository, file manager, database development tool, and HTML editor.
- *Mercurial* as a source code repository.
- *Ant* tasks to build and manage the applications.
- *Firebug* or *Chrome Development tools* to debug and test Javascript code.
- *Apache Jmeter* as a performance monitor.

2.2. Supply Chain Management

This section explains the current business Management theories behind Supply Chain Management (SCM), reviewing the main definitions and problems it tries to address. Some key concepts that require a high level of integration in Business-To-Business processes, such as Just-In-Time (JIT), Kanban, and Lean Management will be introduced and its relationship to B2B Integration demonstrated.

2.2.1. Theoretical Background

According to the CSCMP (Council of Supply Chain Management Professionals), *supply chain management* encompasses the planning and management of all activities involved in sourcing, procurement, conversion, and logistics management. It also includes the crucial components of coordination and collaboration with channel partners, which can be suppliers, intermediaries, third-party service providers, and customers. In essence, supply chain management integrates supply and demand management within and across companies.

Supply chain management is a cross-function approach including managing the movement of raw materials into an organization, certain aspects of the internal processing of materials into finished goods, and the movement of finished goods out of the organization and toward the end-consumer. As organizations strive to focus on core competencies and becoming more flexible, they reduce their ownership of raw materials sources and distribution channels. These functions are increasingly being outsourced to other entities that can perform the activities better or more cost effectively. The effect is to increase the number of organizations involved in satisfying customer demand, while reducing management control of daily logistics operations.

Less control and more supply chain partners led to the creation of supply chain management concepts. The purpose of supply chain management is to improve trust and collaboration among supply chain partners, thus improving inventory visibility and the velocity of inventory movement.

Activities of Supply Chain Management

Several models have been proposed for understanding the activities required to manage material movements across organizational and functional boundaries. Supply chain activities can be grouped into strategic, tactical, and operational levels.

Strategic level

- Strategic network optimization, including the number, location, and size of warehousing, distribution centers, and facilities.
- Strategic partnerships with suppliers, distributors, and customers, creating communication channels for critical information and operational improvements such as cross docking, direct shipping, and third-party logistics.
- Product life cycle management, so that new and existing products can be optimally integrated into the supply chain and capacity management activities.
- Information technology chain operations.
- Where-to-make and make-buy decisions.
- Aligning overall organizational strategy with supply strategy.

- It is for long term and needs resource commitment.

Tactical level

- Sourcing contracts and other purchasing decisions.
- Production decisions, including contracting, scheduling, and planning process definition.
- Inventory decisions, including quantity, location, and quality of inventory.
- Transportation strategy, including frequency, routes, and contracting.
- Benchmarking of all operations against competitors and implementation of best practices throughout the enterprise.
- Milestone payments.
- Focus on customer demand and habits.

Operational level

- Daily production and distribution planning, including all nodes in the supply chain.
- Production scheduling for each manufacturing facility in the supply chain (minute by minute).
- Demand planning and forecasting, coordinating the demand forecast of all customers and sharing the forecast with all suppliers.
- Sourcing planning, including current inventory and forecast demand, in collaboration with all suppliers.
- Inbound operations, including transportation from suppliers and receiving inventory.
- Production operations, including the consumption of materials and flow of finished goods.
- Outbound operations, including all fulfillment activities, warehousing and transportation to customers.
- Order promising, accounting for all constraints in the supply chain, including all suppliers, manufacturing facilities, distribution centers, and other customers.
- From production level to supply level accounting all transit damage cases & arrange to settlement at customer level by maintaining company loss through insurance company.

Problems addressed by SCM operations

Supply chain management must address the following problems:

- **Distribution Network Configuration.** Number, location and network missions of suppliers, production facilities, distribution centers, warehouses, cross-docks and customers.
- **Distribution Strategy.** Questions of operating control (centralized, decentralized or shared); delivery scheme, e.g., direct shipment, pool point shipping, cross docking, DSD (direct store delivery), closed loop shipping; mode of transportation, e.g., motor carrier, including truckload, LTL, parcel; railroad; intermodal transport, including TOFC (trailer on flatcar) and COFC (container on flatcar); ocean freight; airfreight;

replenishment strategy (e.g., pull, push or hybrid); and transportation control (e.g., owner-operated, private carrier, common carrier, contract carrier, or 3PL).

- **Trade-Offs in Logistical Activities.** The above activities must be well coordinated in order to achieve the lowest total logistics cost. Trade-offs may increase the total cost if only one of the activities is optimized. For example, full truckload (FTL) rates are more economical on a cost per pallet basis than less than truckload (LTL) shipments. If, however, a full truckload of a product is ordered to reduce transportation costs, there will be an increase in inventory holding costs which may increase total logistics costs. It is therefore imperative to take a systems approach when planning logistical activities. These trades-offs are key to developing the most efficient and effective Logistics and SCM strategy.
- **Information.** Integration of processes through the supply chain to share valuable information, including demand signals, forecasts, inventory, transportation, potential collaboration, etc.
- **Inventory Management.** Quantity and location of inventory, including raw materials, work-in-progress (WIP) and finished goods.
- **Cash-Flow.** Arranging the payment terms and methodologies for exchanging funds across entities within the supply chain.
- **Supply chain execution** means managing and coordinating the movement of materials, information and funds across the supply chain. The flow is bi-directional.

2.2.2. History of SCM

Supply Chain Management holds its roots in Logistics, what is now considered a sub-set of SCM. Logistics received recognition in military operations during World War II. It gained its momentum as it contributed to the effective distribution of machinery and supplies to troops. A service delivery failure here may mean an increase in unnecessary fatalities. Peter Drucker (a business guru in the 1960's) identified logistics as a growing concern within business. This generated more prominence towards the practice of logistics.

As the economies in North America evolved in the 1970's and 1980's, transportation deregulation changed the competitive landscape of business. Carriers were free to charge their customers (Shippers) a competitive rate for their shipments.

Warehousing companies that typically acted as surplus inventory storage locations, married up with transportation companies to offer customers full-service solution capabilities. This formed the beginning of the 3rd party logistics business and paved the way for outsourcing logistical activities.

A growing number of corporate senior managers in the late 1980s and 1990s came to realize that the flow of products and services provided opportunities for improved competitive performance, if managed more efficiently. Not only did changing market forces affect these activities, but they could also influence the impact of the forces on the competitive position of the company.

Reducing the time it takes to perform delivery or any of the other order fulfillment activities means that customers receive ordered goods more quickly which also reduces the inventory costs.

This concept of SCM evolved over time. The three distinct phases of SCM evolution are:

Phase	Physical Distribution Management	Logistics Management	Supply Chain Management
Date	1960s	1970s-1980s	1980s-90s
Focus	Distribution of final product	Optimizing Internal Operations (function-oriented)	Integration across organizational boundaries (process-oriented)
Concepts Added	<ul style="list-style-type: none"> • Inventory • Out-bound transportation • Warehousing • Order processing 	<ul style="list-style-type: none"> • Manufacturing • Procurement • In-bound transportation 	<ul style="list-style-type: none"> • Management of Customer Relationship • Management of Information • External logistics processes and suppliers

Table ii - Evolution of SCM theory

The major difference between the concept of logistic management and supply chain management is the level of information gathered, processes, analyzed and used for decision making. An SCM-based organization not only having concerns with its immediate clients but also handles and forecasts the factors affect directly or indirectly their supplier or suppliers or on their client or clients. If the information part is excluded out of supply chain model then the logistics management part can be seen.

With the advent of globalization, firms began to seek ways of cutting their production costs. Thus, multi-national corporations re-located their factors of production to low-wage countries to gain a competitive advantage.

Increasingly, more and more countries are joining the World Trade Organization (WTO) and opening their country to foreign capital investment (most recently in India and China).

Retail giants like WalMart exploit these new efficiencies and increase their imports from new emerging economies to reduce product prices in their stores. Thus, the new challenge is how to manage the product and information flows around the world. The increased pressure on managing these operations further underscored the importance of logistics as an area for optimization.

Another contributor that led to an increased presence for logistics was the explosion in information technology and use of computers throughout the 1980's and onwards.

The cost of computing has decreased year after year since then and computing power rose exponentially. The use of the Internet and increased bandwidth capacity further enhanced and

enabled quick connectivity and collaborative relationships that reduced inventories and created a Just-In-Time operating opportunity for organizations.

These efficiencies reduced errors, increased fill-rates and cut overall operating costs for organizations.

The new paradigm became known as the “systems approach” to supply chain management and introduced the concept of trade-offs. In order to achieve least total supply chain cost, operational integration of the 5 main areas of logistics must be simultaneously optimized:

- Warehousing
- Transportation
- Inventory
- Order Processing
- Lot Quantities

2.2.3. Just-In-Time, Kanban and Lean Management

Business Management theory has been traditionally a western social sciences discipline, but since the end of the twentieth century it started borrowing a lot of new concepts from traditional Japanese Management.

The business environment structure in Japan accounts for very large corporations, with integrated processes and systems. It is in these companies where Just-In-Time, Kanban and Lean Management techniques were first implemented.

Business-To-Business integration is key to applying these concepts into modern, global, information systems, and the improvement of supply chains and integrated resource planning systems passes by implementing these business practices.

Just-in-Time (JIT) is a production strategy that strives to improve a business' return on investment by reducing in-process inventory and associated carrying costs. Just In Time production method is also called the Toyota Production System. To meet JIT objectives, the process relies on signals or *Kanban* between different points in the process, which tell production when to make the next part. *Kanban* are usually 'tickets' but can be simple visual signals, such as the presence or absence of a part on a shelf. Implemented correctly, JIT focuses on continuous improvement and can improve a manufacturing organization's return on investment, quality, and efficiency. To achieve continuous improvement key areas of focus could be flow, employee involvement and quality.

Quick notice that stock depletion requires personnel to order new stock is critical to the inventory reduction at the center of JIT. This saves warehouse space and costs. However, the complete mechanism for making this work is often misunderstood.

Kanban is not an inventory control system. Rather, it is a scheduling system that to better understand what to produce, when to product it and in which quantities. It can be understood as a supply depletion management, and supply needs management system. *Kanban* uses the

rate of demand to control the rate of production, passing demand from the end customer up through the chain of customer-store processes.

In the late 1940s, Toyota began studying supermarkets with a view to applying store and shelf-stocking techniques to the factory floor, figuring, in a supermarket, customers get what they need, at the needed time, and in the needed amount. Furthermore, the supermarket only stocks what it believes it will sell, and customers only take what they need because future supply is assured. This led Toyota to view a process as a customer of preceding processes, and the preceding processes as a kind of store. The customer process goes to this store to get needed components, and the store restocks. As in supermarkets, originally, signboards were used to guide "shoppers" to specific restocking locations.

Many manufacturers have implemented electronic *kanban* systems. Electronic *kanban* systems, or **E-Kanban** systems, help to eliminate common problems such as manual entry errors and lost cards. *E-Kanban* systems can be integrated into enterprise resource planning (ERP) systems. Integrating *E-Kanban* systems into ERP systems allows for real-time demand signaling across the supply chain and improved visibility. Data pulled from *E-Kanban* systems can be used to optimize inventory levels by better tracking supplier lead and replenishment times.

Lean Management is a production practice that considers the expenditure of resources for any goal other than the creation of value for the end customer to be wasteful, and thus a target for elimination. Working from the perspective of the customer who consumes a product or service, "value" is defined as any action or process that a customer would be willing to pay for.

Lean manufacturing is a management philosophy derived mostly from the Toyota Production System.

2.2.4. B2B Integration

Companies are redesigning Supply Chain Management processes to reduce non-value-added complexity and requirements in order to improve supply chain application performance. The intent to reduce the complexity is in conflict with the reality of their business processes which are becoming more externalized and complex. They need to provide both the ability to solve end-to-end supply challenges as well as provide a simple and non-complex integration and process architecture.

The current thinking is that instead of linking processes across multiple enterprise applications and aggregated data warehouses, what is desirable is to create a system that can support an integrated business network model.

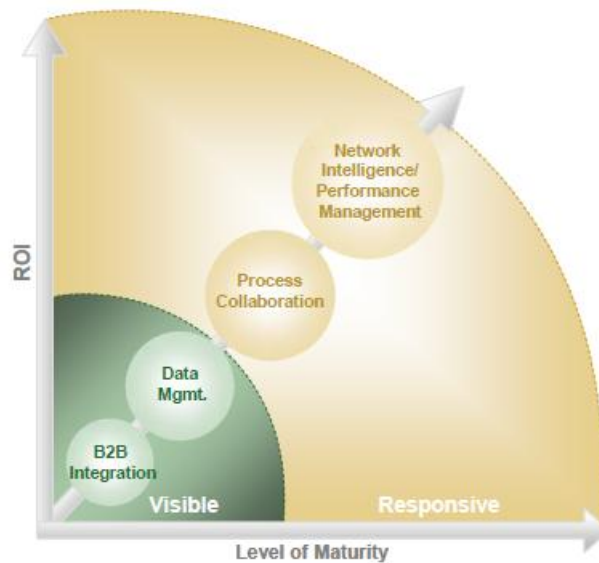


Figure XXVI - Expected evolution of supply chain after B2B Integration

Enabling B2B Integration first within the enterprise and then into the extended enterprise is the essential for the effective supply chains that newer management models and business model require. The extension of supply chains on both the sell side and the buy side has resulted in a corresponding increase in the number of trading partners. It is not possible to rely only on EDI or data – integration to enable connectivity between trading partners to the costs and lack of technology maturity at the trading partners.

Companies need to use multiple methods to establish electronic communication across their full business partner community. Once the enablers for electronic communication are put in place, more advanced processes dealing with forecast collaboration, order management, supply chain event management, etc. can be deployed.

However, perhaps the most important benefit of trading partner connectivity is that it establishes a pipeline for process collaboration. Once a pipeline is created for basic purchasing transactions, it can be used to share critical planning and status information. This information can include sales activity, forecasts, inventory positions, and work-in-process and shipment statuses. Moreover, this information can be shared daily or weekly (or even in real time for certain processes), versus monthly or quarterly, creating leaner and more synchronized processes.

Evolution of Supply Chains raises the importance of B2B Integration

In the early days of manufacturing, vertically integrated companies would produce all of the raw materials and component parts required to build their products. The past 100 years have seen radical transformations in value chains. Corporations have become more specialized, depending upon a network of partnerships to help them design, manufacture, transport and service their products. To illustrate the changing nature of value chains, the dynamics of three of the larger manufacturing sectors—automotive, electronics and consumer products will be explored.

Automotive

In the early days of car manufacturing, companies such as Ford produced all of the materials and performed all of the manufacturing necessary to build a vehicle. The OEM would produce everything from the raw materials such as steel, glass and rubber to the various parts in the engine, exhaust and suspension systems. The automotive supply chain has transformed considerably from the original vertical integration model.

Today's automotive industry is horizontally structured. OEMs are removed from the raw materials process almost entirely. Specialized Tier 3 suppliers produce the steel, aluminum, rubber, glass and leather materials needed for today's vehicles. OEMs have transformed primarily into brand owners with much lighter supply chain management and production responsibilities. Instead, design, development and assembly of vehicle components and subsystems are performed by Tier 1 suppliers. Today's automotive OEMs are focused on driving market demand, innovating product design and enhancing the customer experience. Automotive manufacturers enjoy much higher profit margins from financing services, extended warranties and aftermarket parts than they do from new vehicle sales. Consequently, the industry focus is shifting from the traditional model of just selling cars to a new paradigm centered upon higher margin add-on service transactions.

Electronics

The high tech industry offers another example. Just a few decades ago when mainframe computers were introduced, companies such as IBM owned the entire value chain. OEMs manufactured all the components—storage, CPU, memory, displays and peripherals. Software applications were developed by the hardware manufacturer as well. Early mainframe OEMs such as GE, IBM and Honeywell developed the operating systems, databases and business applications for their platforms. In fact, the OEM often provided all the support services including data center hosting, call center support, systems management and application upgrades. The high tech supply chain has transformed from its originally vertical integrated model. Today's computer value chain is horizontally structured. High-end server equipment is assembled and marketed by an OEM brand owner. However, the hardware components are made by various independent suppliers. For example, the memory may be manufactured by Kingston, the CPU by Intel and the storage by Seagate. Software applications are developed by an independent community of developers. For example, a server-grade operating system might be developed by Microsoft, business applications by SAP and a database management system by Oracle. OEMs offer services such as hardware maintenance, call center support and systems integration. However, many corporate customers prefer to buy custom development, application hosting and IT outsourcing from specialized providers such as Accenture, Wipro and EDS.

Consumer Products

Consumer products companies are transforming the value chain as well. Historically, brand owners such as P&G, Coca-Cola and Nike performed design, development, manufacturing, sales and distribution of their products. Raw materials such as sugar, corn or wheat were often sourced from third parties. However, manufacturing and supply chain management were

considered core competencies of consumer products leaders. Today, consumer products companies are developing more specialized operations focusing primarily on activities in which they can gain a competitive advantage. Some are becoming brand companies with strengths in market research, product design and demand creation. Others are specializing in manufacturing, acting as contractors for their retail customers or even other consumer products brands. Non-core functions are outsourced to specialized third parties around the world. In the apparel sector, third party contract manufacturers are used to produce clothing and footwear in low cost geographies.

In the food industry, specialized brokers often are used to manage the product sale and customer relationships with selected retailers. In the beverage industry, distributors perform store delivery, product replenishment and regional marketing. In the media and entertainment segment, Fourth Party Logistics Providers (4PLs) provide expertise in category management, new product introductions and in-store advertising.

Specialization Partners and external data sources

The examples above demonstrate the transformation that has occurred in supply chains over the past few decades. Manufacturers have migrated from vertically integrated models to more specialized approaches leveraging outsourcing partners. To support the new model, manufacturers have built an international community of specialized partners to help manage their supply chain. Specialists take many forms, employing a variety of business models.

Business Function	Specialized Trading Partner
Design and Assembly	Original Design Manufacturers (ODMs) Contract manufacturers Postponement specialists
Transportation	Marine transportation Air transportation Rail transportation Full truck load (ground transportation) Less than truckload (ground transportation)
Logistics	Third party logistics providers Customs brokers Importers Exporters Freight forwarders Consolidators
Sales Channels	Distributors Bottlers Brokers Agents Resellers
Marketing Services	Fourth party logistics providers Marketing specialists
Financial Services	Banks for payments and foreign exchange Lenders and Factoring Providers Commercial Trade Insurers

Table iii - Specialization of trading partners by area

Specialization and outsourcing are not limited to the supply chain. Increasingly, back office business processes are being sourced to specialized third parties. Examples of outsourcing exist in every major business function. Traditionally, business process outsourcing has focused on transferring selected functions within an organization to an external firm. Common examples of outsourcing selected business functions include:

- **Payroll.** Payroll processing is one of the most popular business functions to outsource. Specialized providers will assume responsibility for employee pay distribution on behalf of human resources. The processors will host the software applications which calculate personnel salaries, tax withholdings and benefit contributions. Payroll providers can also manage the actual payment processes including paper check printing, electronic funds transfer or stored value cards.
- **Information Technology.** A wide variety of information technology outsourcing providers are on the market. Some firms specialize in running a corporate IT infrastructure including network management, desktop support and data center operations. Others will take on a broader scope including business process consulting, custom software development and ongoing application maintenance.
- **Marketing.** Outsourcing of selected marketing functions to specialized third parties is a common practice. Traditionally, corporations have sought outside assistance with brand development, advertising and public relations. In the past 10 years, more specialized firms have emerged to manage primary research projects, product concept testing, and website design and search engine optimization.

A new breed of outsourcing is emerging that extends beyond the selective sourcing approaches used in the past. In the new model corporations are beginning to outsource entire functional disciplines to third party providers. Examples of business process outsourcing include:

- **Finance and Accounting.** Business process outsourcing for finance and accounting is a nascent, but quickly growing, area. Specialized providers can offload the entire finance function from an organization including accounts payable, accounts receivable, tax management, treasury management, risk management and regulatory activities.
- **Human Resources.** Another growing area of business process outsourcing is human resources. A number of multi-national corporations have contracted with third parties to provide personnel management, organizational development, recruiting and hiring, benefits administration, and compensation planning and strategy and performance management services.

External Data Sources in ERP systems

In both the manufacturing and services sector the trend is clear. Corporations are specializing. Industry leaders are developing core competencies and deep expertise in particular niche functions. The specialized focus enables higher levels of innovation with better economies of scale. The benefits of the new business models are significant. However, the risks should not be underestimated. The specialization of roles that occurs with horizontally structured value chains creates a strong dependency upon business partners for day-to-day operations. The implications of the change can be substantial, particularly if the choice of business partners

proves to be problematic. Consider what the impacts would be if a key value chain partner became financially insolvent or suspended operations temporarily. The disruption to a supply chain and business operations could last for days, if not weeks.

There are implications for information technology strategy as well. In today's specialized value chain, information systems are dependent upon your business partners as well. Applications such as ERP quickly become inoperable without data feeds from external sources. In fact, a high percentage of the data housed in enterprise applications actually originates from external business partners.

Business Application	Data Sourced from Outside the Enterprise	External Business Partner Sources
Enterprise Resource Planning	Customer Forecasts, Customer Orders	Customers, Distributors, Brokers, Agents, Resellers
Procurement and Sourcing	Product Catalog, Pricing, Promotions, Vendor	Vendor/Suppliers, Marine, Air, Rail Transportation
Transportation Management System	Shipment Status, Import/Export Documentation	Providers, Freight Forwarders, Customs Brokers, LTL, TL, Parcel Carriers, 3PLs
Finance and Accounting	Supplier Invoices, Customer Invoices, Remittance Advices, Payroll, General Ledger	Vendors, Customers, F&A BPO Providers, Payroll BPO Providers
Treasury Workstation	Bank Account Statements, Foreign Exchange Transactions, Securities Ownership	Cash Management Banks, Securities Broker/Dealers, Foreign Exchange Banks
Human Resources	Recruiting, Performance, Compensation, Employee records	HR BPO Providers
Information Technology	Data center, network, application status, Trouble ticket status	IT Outsourcing Providers

Table iv - Sources of external data of ERPs

2.3. Software Development Methodology and Open Source

This section will try to introduce the concept of Software Development Methodology, part of the Software Engineering discipline, to explain the methods and work to be done as part of the development of the B2B Integration Module.

It will also describe the characteristics of Open Source Software (OSS), which is the kind of software licensing of Openbravo ERP. This will be key to understand the specific environment on which the B2B Integration Module will be developed and published.

2.3.1. Software Development Methodology

Software Engineering, as described by the IEEE Computer Society, is defined as “the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software”. It is the application of Engineering to software because it integrates significant mathematics, computer science and practices whose origins are in Engineering.

This engineering science as then created a number of models or methodology frameworks, used to structure, plan and control the process of developing software and information systems. They are known as “Software Development Methodologies”, and each of these models is composed by a number of phases, steps, and milestones. Each of the phases requires some activities to be done and validated before passing to the next one.

The most widely known and simple methodologies, which have many things in common, are the Waterfall Model, Prototyping, and Spiral Development:

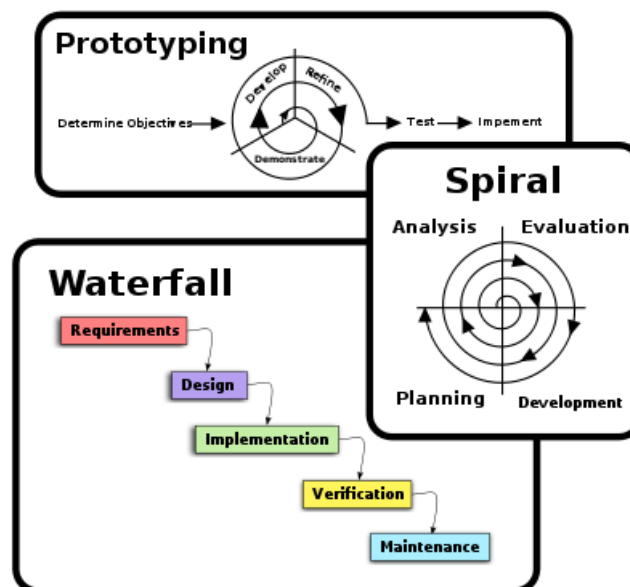


Figure XXVII – The three most common Software Development Models

The Waterfall Model

The waterfall model is a sequential design process, often used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through

the phases of Conception, Initiation, Analysis, Design, Construction, Testing, Production/Implementation and Maintenance.

The basic principles are:

- The Project is divided into sequential phases, with some overlap and splashback acceptable between phases.
- Emphasis is on planning, time schedules, target dates, budgets and implementation of an entire system at one time.
- Tight control is maintained over the life of the project via extensive written documentation, formal reviews, and approval/signoff by the user and information technology management occurring at the end of most phases before beginning the next phase.

The waterfall development model originates in the manufacturing and construction industries: highly structured physical environments in which after-the-fact changes are prohibitively costly, if not impossible. Since no formal software development methodologies existed at the time, this hardware-oriented model was simply adapted for software development.

Prototyping

Software Prototyping is the development approach of activities during software development, the creation of prototypes, incomplete versions of the software program being developed.

The basic principles are:

- Not a standalone, complete development methodology, but rather an approach to handling selected parts of a larger, more traditional development methodology (i.e. incremental, spiral, or rapid application development (RAD)).
- Attempts to reduce inherent project risk by breaking a project into smaller segments and providing more ease-of-change during the development process.
- User is involved throughout the development process, which increases the likelihood of user acceptance of the final implementation.
- Small-scale mock-ups of the system are developed following an iterative modification process until the prototype evolves to meet the users' requirements.
- While most prototypes are developed with the expectation that they will be discarded, it is possible in some cases to evolve from prototype to working system.

A basic understanding of the fundamental business problem is necessary to avoid solving the wrong problem. There are mainly two types of prototyping:

- Throwaway prototyping, in which the model will eventually be discarded and each new prototype will be developed from scratch, using the ideas and concepts learnt.
- Evolutionary Prototyping, in which each new prototype is derived directly from the discarded one, by re-examining the parts from it that doesn't fit the specifications and keeping the valid ones.

Spiral Development

The spiral model is a software development process combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts. Also known as the spiral lifecycle model (or spiral development), it is a systems development method (SDM) used in information technology. This model of development combines the features of the prototyping model and the waterfall model. The spiral model is intended for large, expensive and complicated projects.

The basic principles are:

- Focus is on risk assessment and on minimizing project risk by breaking a project into smaller segments and providing more ease-of-change during the development process, as well as providing the opportunity to evaluate risks and weigh consideration of project continuation throughout the life cycle.
- Each cycle involves a progression through the same sequence of steps, for each part of the product and for each of its levels of elaboration, from an overall concept-of-operation document down to the coding of each individual program.
- Each trip around the spiral traverses four basic quadrants: (1) determine objectives, alternatives, and constraints of the iteration; (2) evaluate alternatives; Identify and resolve risks; (3) develop and verify deliverables from the iteration; and (4) plan the next iteration.
- Begin each cycle with an identification of stakeholders and their win conditions, and end each cycle with review and commitment.

2.3.2. Software Development Process

No matter what methodology is used, software development process will feature some common activities and steps. In addition, these activities need some supporting documentation and tools to be accomplished.

Requirements and Specifications

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users.

Requirements analysis is critical to the success of a development project. Requirements must be documented, actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design. Requirements can be architectural, structural, behavioral, functional, and non-functional.

Documents related to this phase are the *Business or User flow Analysis* and the *Functional Gap Analysis*.

A functional specification in systems engineering and software development is the documentation that describes the requested behavior of an engineering system. The

documentation typically describes what is needed by the system user as well as requested properties of inputs and outputs (e.g. of the software system).

Architecture and Design

The software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both. The term also refers to documentation of a system's software architecture. Documenting software architecture facilitates communication between stakeholders, documents early decisions about high-level design, and allows reuse of design components and patterns between projects.

Software design is a process of problem solving and planning for a software solution. After the purpose and specifications of software are determined, software developers will design or employ designers to develop a plan for a solution. It includes low-level component and algorithm implementation issues as well as the architectural view.

Documents related to this phase are the *Functional Design* and *Technical Design*.

Implementation and Testing

Implementation is commonly known as plan *Computer Programming* (often shortened to programming or coding) and it is the process of designing, writing, testing, debugging, and maintaining the source code of computer programs. This source code is written in a programming language. The purpose of programming is to create a program that exhibits a certain desired behavior. The process of writing source code often requires expertise in many different subjects, including knowledge of the application domain, specialized algorithms and formal logic.

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs (errors or other defects).

Software testing can also be stated as the process of validating and verifying that a software program/application/product:

- meets the business and technical requirements that guided its design and development
- works as expected
- It can be implemented with the same characteristics.

Documents related to this phase are the *Code Documentation*, *Code Comments*, and *Testing Battery*.

Maintenance

Software maintenance in software engineering is the modification of a software product after delivery to correct faults, to improve performance or other attributes.

The integral part of software is the maintenance part which requires accurate maintenance plan to be prepared during software development and should specify how users will request modifications or report problems and the estimation of resources such as cost should be included in the budget and a new decision should address to develop a new system and its quality objectives. The software maintenance which can last for 5–6 years after the development calls for an effective planning which addresses the scope of software maintenance, the tailoring of the post delivery process, the designation of who will provide maintenance, an estimate of the life-cycle costs.

Documents related to this phase are the *User Manuals, Configuration Manuals, Support Plan and Deployment Plan*.

Software Development Tools

Software Development in some ways is not too different from manufacturing. Each of the activities described above need some supporting tools, the same as the assembly of a product will need a set of tools and machines, different for each phase.

- **Programming Tools.** These include from enhanced *source code editors*, designed for each different programming language (Java, C++, SQL, etc.), to *database development tools* for each database family (PostgreSQL, Oracle, MySQL, etc.), *source code compilers*, *source code interpreters* and *source code formatters* (indent),
- **Testing and Integration Tools.** Including *Debuggers* and *Bug Tracking tools*, *Build Automation* software, *Performance Analysis* tools or *Sizing* Tools.
- **Source Code Revision Tools.** Both *versioning*, *code sharing*, *code coverage* and *code revision* sites. Includes backup and recovery tools.
- **Document Control Tools.** This type of tools may vary from traditional *Word Processors* to *Software Diagramming*, or even *Document Management* tools.
- **Integrated Development Environments (IDEs).** An IDE is a software application that provides comprehensive facilities to computer programmers for software development, all consolidated into one place. They may include file systems explorers, specialized editors for many different programming, scripting and markup languages, source code revision and sharing tools, and history and back-up tools.
- **Computer Aided Software Engineering (CASE).** CASE tools are information systems designed to *manage the whole process of software development*, mainly from an engineering and management point of view. They mostly include integrated tools to support strategic planning, project management, requirements gathering, etc.

2.3.3. Open Source Software

Open-source software (OSS) is computer software that is available in source code form: the source code and certain other rights normally reserved for copyright holders are provided under a software license that permits users to study, change, improve and at times also to distribute the software.

Some open source licenses meet the requirements of the Open Source Definition. Some open source software is available within the public domain.

Open source software is very often developed in a public, collaborative manner. Open-source software is the most prominent example of open-source development and often compared to (technically defined) user-generated content or (legally defined) open content movements.

Definition and Examples

The Open Source Definition is used by the *Open Source Initiative* to determine whether or not a software license can be considered open source. The definition is a short text as follows:

Introduction. Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. **Free Redistribution.** The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.
2. **Source Code.** The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.
3. **Derived Works.** The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
4. **Integrity of The Author's Source Code.** The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.
5. **No Discrimination Against Persons or Groups.** The license must not discriminate against any person or group of persons.
6. **No Discrimination Against Fields of Endeavor.** The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.
7. **Distribution of License.** The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8. **License Must Not Be Specific to a Product.** The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.
9. **License Must Not Restrict Other Software.** The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.
10. **License Must Be Technology-Neutral.** No provision of the license may be predicated

on any individual technology or style of interface

Table v - Definition of Open Source Software

Some examples of widely used and popular Open Source Software are the Apache HTTP Server, the internet browser Mozilla Firefox, the GNU/Linux operating system, MediaWiki software powering Wikipedia, and Google's Android operating system for mobile phones and tablets.

Differences with closed source paid software and closed source free software

The debate over open source vs. closed source (alternatively called proprietary software) is sometimes heated.

One source of conflict is related to economics: Making money through traditional methods, such as sale of the use of individual copies and patent royalty payment (generally called licensing), is more difficult and in many ways against the very concept of open source software.

Some closed-source advocates see open source software as damaging to the market of commercial software. This is one of the many reasons, as mentioned above, that the term free software was replaced with open source — because many company executives could not believe in a product that did not participate economically in a free-market or mixed-market economy.

The counter to this argument is the use of open source software to fuel the market for a separate product or service. For example:

- Providing support and installation services; similar to IT Security groups, Linux Distributions, and Systems companies.
- Using the software as a stepping stone to sell a higher-end product or service;
- Cost avoidance / cost sharing: many developers need a product, so it makes sense to share development costs

Since open source software is open, defects and security flaws are more easily found. Closed-source advocates argue that this makes it easier for a malicious person to discover security flaws. Further, that there is no incentive for an open-source product to be patched. Open-source advocates argue that this makes it easier also for a patch to be found and that the closed-source argument is security through obscurity, which this form of security will eventually fail, often without anyone knowing of the failure. Further, that just because there is not an immediate financial incentive to patch a product, does not mean there is not any incentive to patch a product. In addition, if the patch is that significant to the user, having the source code, the user can technically patch the problem themselves. These arguments are hard to prove. Open Source software, generally has a lower percentage of bugs than some commercial software.

Funding. Commercial Open Source Companies. Licensing

Since GNU and some other open source licenses stipulate that derived works must distribute their intellectual property under an open source license, commercial companies engaging in open source software development activities have developed legal and technical mechanisms to foster their commercial goals:

- **A dual-license model**, where a code base is published under a traditional open source license and a commercial license simultaneously. Vendors typically charge a perpetual license fee for additional closed-source features, supplementary documentation, testing, and quality, as well as intellectual property indemnification to protect the purchaser from legal liability. This is the case for Openbravo ERP with its Professional Edition and Openbravo Commercial License.
- **Functional encapsulation**, where an open source framework or library is installed on a user's computer separately from the commercial product, and the commercial product uses the open source functionality in an "arm's length" way (under the argument that the commercial product was shipped without the open source library, even though it uses it). Vendors typically charge a perpetual license fee for the functionality that they provide under closed source, as they usually don't provide services or other direct value for the open source elements.
- **A software as a service model**, under the argument that the vendor is charging for the services, not the software itself (because the software is never shipped to customers or installed on their computers). Vendors typically charge a monthly subscription fee for use of their hosted applications.
- **Not charging for the software**, but only for the support, training, and consulting services that assist users of the open source software. Vendors typically charge an annual fee for support, per-student fees for training, and per-project fees for consulting engagements.

The underlying objective of these business models is to harness the size and international scope of the open source community (typically more than an order of magnitude larger than what would be achieved with closed source models) for a sustainable commercial venture.

There is considerable debate about whether vendors make a sustainable business from an open source strategy. In terms of a traditional software company, this is probably the wrong question to ask. Looking at the landscape of open source applications, many of the larger ones are sponsored (and largely written) by system companies such as IBM and Sun who may not have an objective of software license revenues. Their motivation tends to be more strategic, in the sense that they are trying to change the rules of a marketplace and reduce the influence of vendors such as Microsoft. In the case of smaller vendors doing open source work, their objectives may be less "immediate revenue growth" and more "developing a large and loyal community," which may be the basis of a corporate valuation at merger time.

Development philosophy, differences with traditional model

The Cathedral and the Bazaar is a concept used to explain development philosophy of Open Source Software communities, as opposed to traditional proprietary software

In the traditional model of development, the cathedral model, development takes place in a centralized way. Roles are clearly defined. Roles include people dedicated to designing (the architects), people responsible for managing the project, and people responsible for implementation. Traditional software engineering follows the cathedral model.

The bazaar model, however, is different. In this model, roles are not clearly defined. Software developed using the bazaar model should exhibit the following patterns:

- **Users treated as co-developers.** The users are treated like co-developers and so they should have access to the source code of the software. Furthermore users are encouraged to submit additions to the software, code fixes for the software, bug reports, documentation etc. Having more co-developers increases the rate at which the software evolves. Linus's law states that, "Given enough eyeballs all bugs are shallow." This means that if many users view the source code they will eventually find all bugs and suggest how to fix them. Note that some users have advanced programming skills, and furthermore, each user's machine provides an additional testing environment. This new testing environment offers that ability to find and fix a new bug.
- **Early releases.** The first version of the software should be released as early as possible so as to increase one's chances of finding co-developers early.
- **Frequent integration.**
- Code changes should be integrated (merged into a shared code base) as often as possible so as to avoid the overhead of fixing a large number of bugs at the end of the project life cycle. Some open source projects have nightly builds where integration is done automatically on a daily basis.
- **Several versions.** There should be at least two versions of the software. There should be a buggier version with more features and a more stable version with fewer features. The buggy version (also called the development version) is for users who want the immediate use of the latest features, and are willing to accept the risk of using code that is not yet thoroughly tested. The users can then act as co-developers, reporting bugs and providing bug fixes.
- **High modularization.** The general structure of the software should be modular allowing for parallel development on independent components.
- **Dynamic decision making structure.** There is a need for a decision making structure, whether formal or informal, that makes strategic decisions depending on changing user requirements and other factors

Specialized Tools for OSS development

In OSS development the participants, who are mostly volunteers, are distributed amongst different geographic regions so there is need for tools to aid participants to collaborate in source code development. Often these tools are also available as OSS.

Revision control systems such as Concurrent Versions System (CVS) and later Subversion (svn) and Git are examples of tools that help centrally manage the source code files and the changes to those files for a software project.

Utilities that automate testing, compiling and bug reporting help preserve stability and support of software projects that have numerous developers but no managers, quality controller or technical support. Building systems that report compilation errors among different platforms include Tinderbox. Commonly used bugtrackers include Bugzilla and GNATS.

Tools such as mailing lists, IRC, and instant messaging provide means of Internet communications between developers. The Web is also a core feature of all of the above systems. Some sites centralize all the features of these tools as a software development management system.

Chapter 3. Openbravo ERP Core Business Flows. B2B Integration Gap.

This chapter studies the **core** business processes covered by **Openbravo ERP** and confronts them to the theories on Supply Chain Management explained beforehand, so that the wet of missing functionalities, known as **Functional Gap** will arise, and serves as the reference for the Functional Design.

3.1. Core Business Flows for Openbravo ERP

Openbravo ERP, in its 3.0 version, is based on a narrower functional footprint than previous releases. This release focus exclusively on the key flows required by the large majority of ERP customers, with the aim to:

- Eliminate complexity from the product;
- Perfect the core functionality;
- Improve adoption rate and successful implementations;

Moreover, this smaller core functionality, hard-proof to errors and bugs, fully open-sourced and documented, in combination with the powerful modular architecture that had been developed in previous versions, allows better looking and more professional extensions, and expands reasonably the adaptability of the software.

In this sense, it is important to note that any functional extension, especially the ones that will allow new ways of interacting with vendors and customers such as a B2B Integration and Supply Chain Management extension, should know very well the core business flows and should know exactly in which point of these flows it is going to affect standard operation.

The current twelve core workflows, as they are defined in the Openbravo ERP documentation are: *Procure-To-Pay* (divided into *Requisition to Receipt* and *Supplier Invoice to Payment*), *Financial Account to Reconciliation*, *Period End Close to Financial Report*, *Customer Return to Credit*, *Customer Return to Replacement*, *Supplier Return to Debit*, *Supplier Return to Replacement*, *Wall to Wall Accuracy Physical Count*, *Asset Acquisition to Dispose*, *Tax Payment*, and finally, *Sales Order to Cash* (which is divided as well in three sub-workflows: *Order to Shipment*, *Customer Invoice to Cash* and *Collection*).

Each workflow demonstrates a key operation to the management of the organization, involving the transfer of materials, money or information, either from inside or outside the boundaries the organization.

3.1.1. Procure To Pay: Requisition to Receipt

This workflow manages the life-cycle of a purchase process, from the moment the Purchase Manager places the Purchase order for products to the supplier, to the moment the warehouse staff receives the products in stock.

This process refers to an external supplier. It usually starts with a purchase requisite from one of the departments in your organizations (for smaller companies that step is often omitted), which is converted, after some price and terms negotiation with your vendor, to a purchase order which is sent to your vendor. The vendor, usually upon receiving your order will ship you the ordered products (not necessarily all the products you ordered are shipped to you as it depends on the products' availability) together with an invoice. The invoice will be paid by your accounting team, and the process will be completed.

The abstract *Procure to Pay* business process is defined by the following diagram:

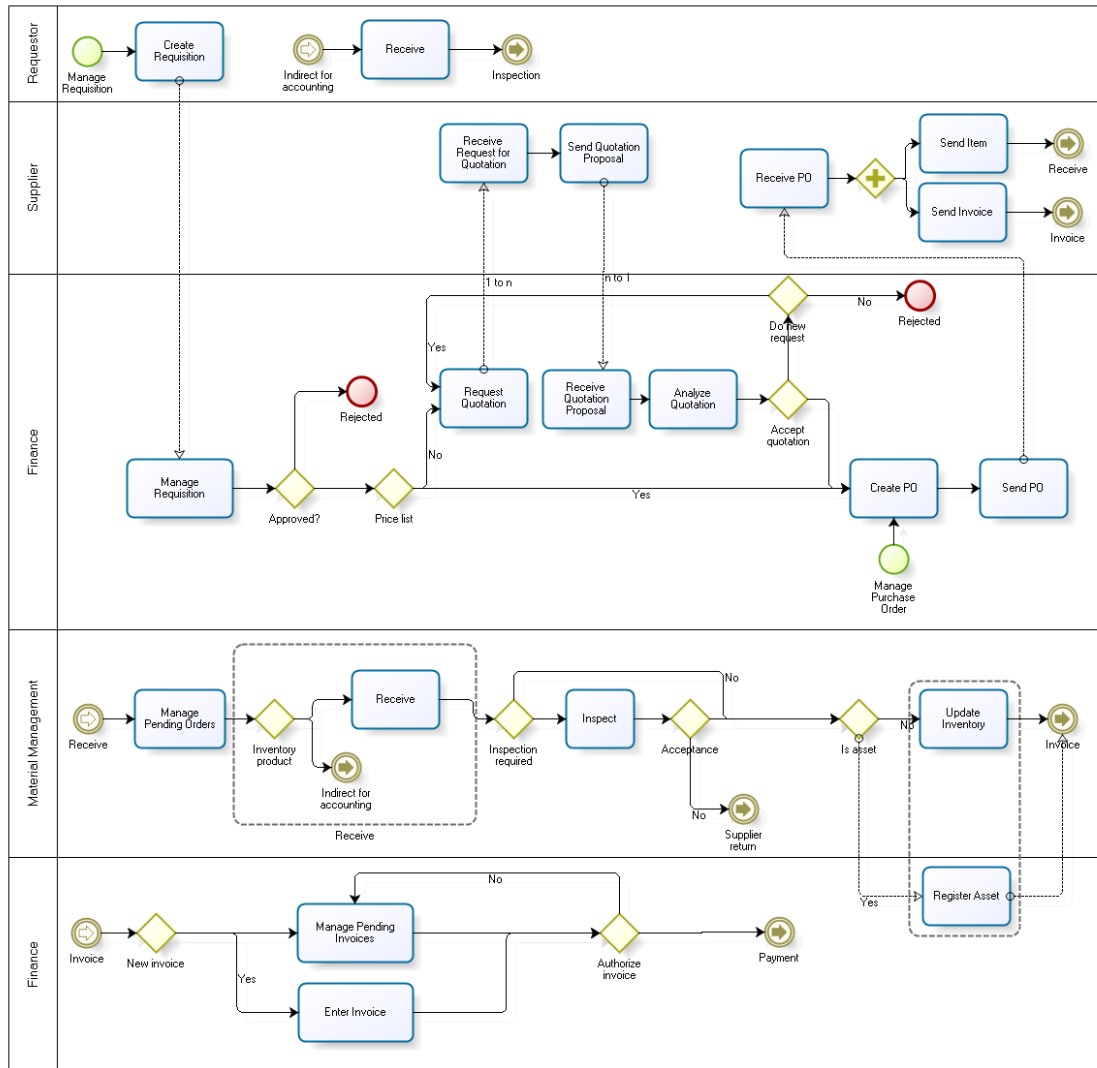


Figure XXVIII - Business Flow: Procure To Pay

The process can be summarized in these steps:

- Entering and Managing requisitions.** A Requisition is a document that specifies a request to order products. The requisition can be entered by anyone in the company when a user detects a low stock level or is expecting a high demand. The Requisition is reviewed by the Purchasing department and as a result, a Purchase Order can be created automatically by the Purchase Management. The *Requisitions Flow* is fully

supported by Openbravo ERP, and it is a fully internal process, no agents external to the organization are involved.

- **Requesting and managing quotations.** A Quotation is a document issued by a supplier which states the cost for some products or services to be delivered, ordered from a specific vendor with the related prices, terms and conditions. If accepted by the Organization, it can be converted into a Purchase Order directly, as it serves as a commitment by a Supplier to issue these products or services at that cost. Currently, the *Quotations Flow* is not entirely supported by Openbravo ERP, other than directly creating different versions of Purchase Orders. The product is lacking a way of managing current and past quotations altogether. Furthermore, the sending and receiving of quotations (by email or printing to send them by fax) is not supported inside Openbravo ERP.
- **Ordering the goods.** A Purchase Order is a document that specifies products or services ordered from a specific vendor with the related prices, terms and conditions. It also serves as commitment by the Organization to the Supplier to acquire and pay these products or services at that cost. The document has to be sent to the Supplier. Currently, Openbravo ERP allows creating a PDF version of this document so that it can be printed and faxed to the vendor, or emailed. Once it has been sent, the communication with the vendor is managed totally outside of Openbravo ERP. There is no way of capturing relevant metrics or history of this communication.

powered by openbravo

Email Document

To: supplier@yahoo.com Esautomátix

From: Spruch BBAdmin

Subject: Ordering of goods

Message Body: Find attached our order

References will be replaced by values:
@cus_ref@: The document reference of the customer
@our_ref@: The reference of the document
@cus_nam@: The name of the customer
@sal_nam@: The name of the sales rep.

File name	Archive
Purchase_Order-800023.20090918-095548.pdf	

Browse...

Add Attachment

Send Cancel

Figure XXIX - Sending a Purchase Order to a Supplier

- **Receiving the goods.** Upon reception of goods, a Goods Receipt document is filled out. This document consists on a header and lines, and it is a subset of the original Purchase Order, while it could also contain lines from other, past purchase orders. Currently in Openbravo ERP, goods are expected to be received by the date specified in the purchase order, and there is no way of telling whether the supplier has

dispatched the goods before. It is in the moment of the reception of the goods when a purchase order and its goods receipt are confronted.

- **Updating inventory.** Inventory is updated automatically once the Goods Receipt document is completed.

3.1.2. Procure To Pay: Supplier Invoice to Payment

In Openbravo ERP, when generating an invoice, the payments details, such as the form of payment and payments terms (date and amount) are also created. This workflow manages the second part of the life-cycle of a purchase process, from the moment the goods have been received and inventory has been updated, to the moment a payment is made to the supplier against an invoice it has issued. These are the main sub-processes for it:

- **Review invoices pending to be paid and its payment details.** The supplier will invoice the organization and send the documents. Pending invoices can be reviewed by the Organization's management, and the data has to be typed into the system. Payment details usually will follow those contained in the purchase order. Furthermore, the vendor could contact and inquire the Organization regarding some invoice it has sent, to check possible inaccuracies.
- **Decide whether to pay or to renegotiate.** Depending on the data provided by the ERP system, the Organization can try to renegotiate some invoices. Currently, this has to be done confronting data in Openbravo ERP and the supplier's own information system.
- **Perform the payment depending on the payment method.** The payment is made by the Organization to the Vendor. Usually, a bank wire transfer is made to the bank account of the vendor; the Organization staff will take this information from its records in Openbravo ERP master data management. Once this payment has been made, a payment document has to be created in Openbravo ERP. These will then serve as a kick off for the Bank Reconciliation and Accounting business flows. Note that currently, there is no way of telling if the Vendor received the payment or not, it is done blindly and the ERP is updated always trusting that this process was completed successfully.

3.1.3. Financial Account to Reconciliation

The reconciliation of financial accounts for bank statements or for cash consists on updating the information held in the ERP referring to the status of financial accounts against the actual information, which comes in the form of bank statements (paper or electronic) or direct information of cash boxes (cash count).

The information in the ERP comes from payment documents, generated against purchase or sales invoices or orders, reviewed in other workflows.

The abstract *Financial Account to Reconciliation* is defined by the following business process diagram:

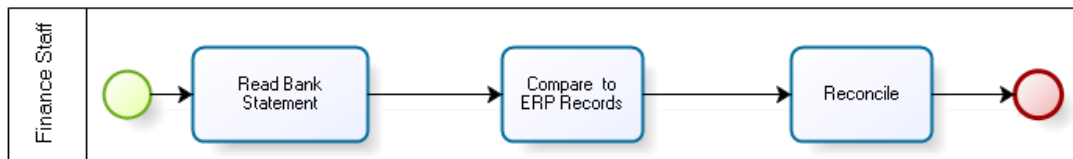


Figure XXX - Business Flow: Financial Account to Reconciliation

It consists of the following sub processes:

- **Read/Import Bank Statement or Cash.** A bank statement is normally received in a paper or electronic format. Regardless, it is a list of outgoing and incoming payment transactions that your bank has recorded. Bank Statements can be imported in an electronic format. Cash can be directly counted from cash boxes.
- **Compare to ERP Records.** Openbravo ERP should have a thorough description of the existing financial accounts, and its history of operations. The ERP records should list the same incoming and outgoing transactions than a bank statement. If this is not the case, the records have to be reconciled.
- **Reconcile (using automatic feature with the algorithm or manually through the form).** Reconciliation is the process by which incoming or outgoing payments are matched to expected incoming or outgoing payments in the ERP. For instance, a bank statement may indicate that a specific payment has been already made while the ERP had not been updated yet. Reconciliation is almost entirely an internal process, with no B2B interaction, other than enquiries to Vendors and Customers to manage exceptions. It can however benefit from the extra input if Vendors or Customers could directly inform of the status of their payments.

3.1.4. Period End Close to Financial Report

This workflow manages the life-cycle of an accounting period, from the moment the finance staff opens it to the moment they close it and finally permanently close it.

Note: this business flow is entirely internal to the Organization, and as such, no B2B Integration can be applied to it. It can surely benefit from it by being able to create more accurate reports if B2B Integration with vendors or customers is in place, however.

The abstract *Period end Close to Financial Report* business process is defined by the following diagram:

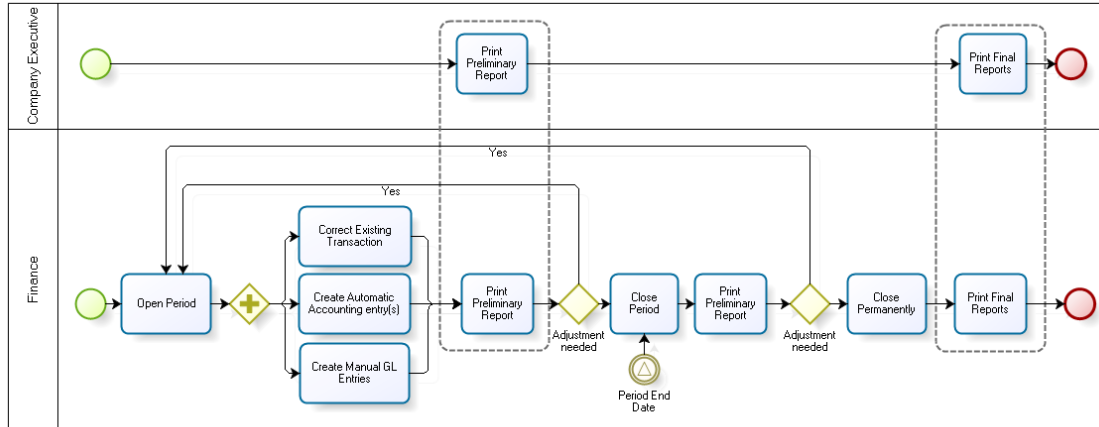


Figure XXXI - Business Flow: Period End Close to Financial Report

It consists of the following sub processes:

- **Open Period.** Each financial period (usually a fiscal year) has to be created in Openbravo ERP.
- **Create, Correct and Delete accounting entries.** Accounting entries are created automatically by the proper transactional documents (Invoices, Shipments, Payments, etc.). Manual Entries can be created as well, however this should be a less frequent case.
- **Close Period.** Each financial period (usually a fiscal year) comes to an end, and it is closed. Financial Reports are generated automatically, and when validated, the period can be permanently closed.

3.1.5. Customer Return to Credit

This workflow manages the return of goods from a customer that results in additional credit to that customer.

The abstract *Customer Return to Credit* process is defined by the following diagram:

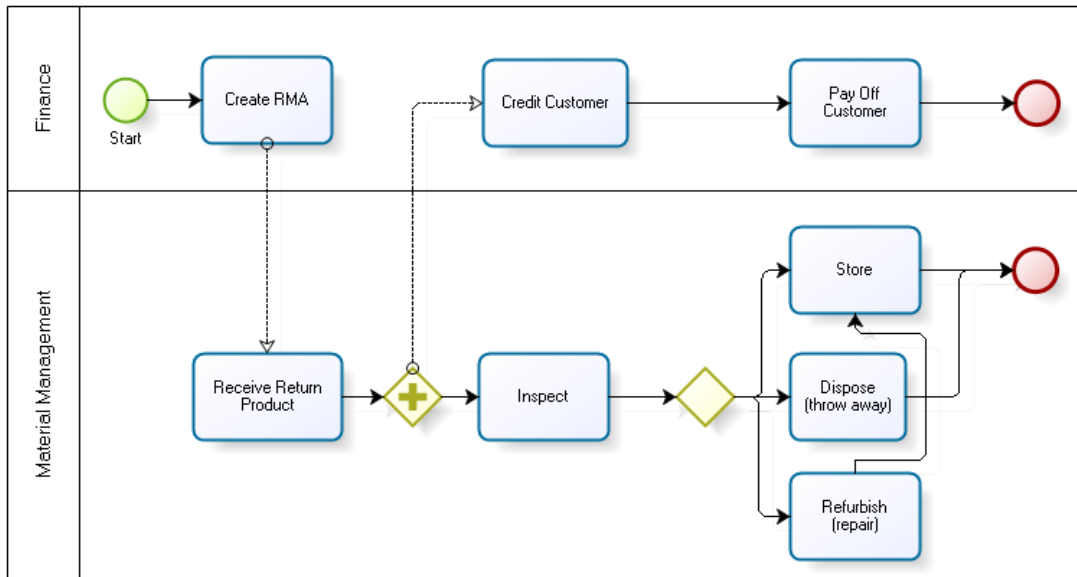


Figure XXXII - Business Flow: Customer Return to Credit

The following sub-processes form it:

- **Return Merchandise Authorization (RMA)** - authorize the return of a product from a customer.
- **Receive Returned Product** - receive the product into the warehouse
- **Inspection and depending on the outcome:**
 - Store the product
 - Dispose the product
 - Refurbish (Repair) the product
- **Credit Customer** - credit the customer with the value of the returned product
- **Pay Off Customer** - manage the reimbursement or settlement

3.1.6. Customer Return to Replacement

This workflow manages the return of goods from a customer that requires replacement for the goods. It is similar to the previous workflow with the exception that a new shipment of product has to be done.

The abstract process *Customer Return to Replacement* is outlined by the following diagram:

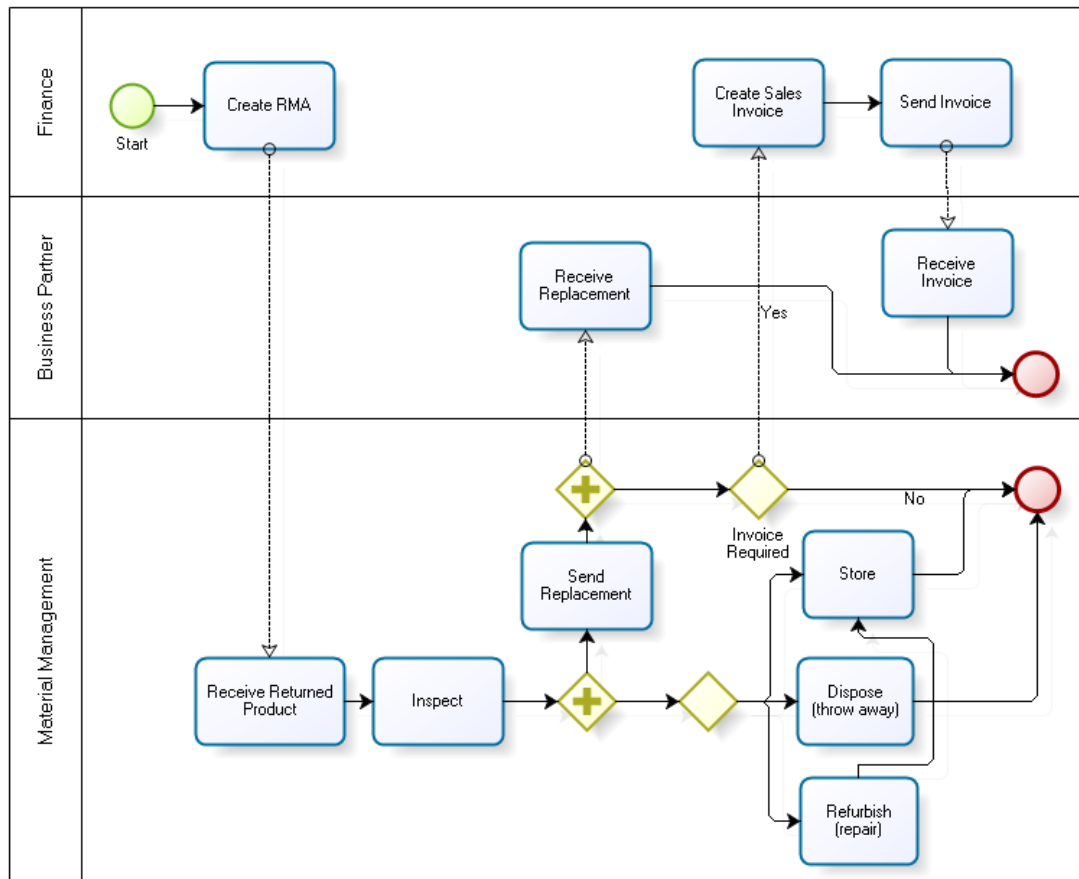


Figure XXXIII - Business Flow: Customer Return to Replacement

The following sub-processes form it:

- **Return Merchandise Authorization (RMA)** - authorize the return of a product
- **Receive Returned Product** - receive the product into the warehouse
- **Inspection and depending on the outcome:**
 - Store the product
 - Dispose the product
 - Refurbish (Repair) the product
- **Send Replacement**
- **Create and Send a Sales Invoice**

3.1.7. Supplier Return to Debit

This workflow manages the return of goods the Organization has purchased back to the Vendor and request of debit.

The abstract *Supplier Return to Debit* process is defined by the following diagram:

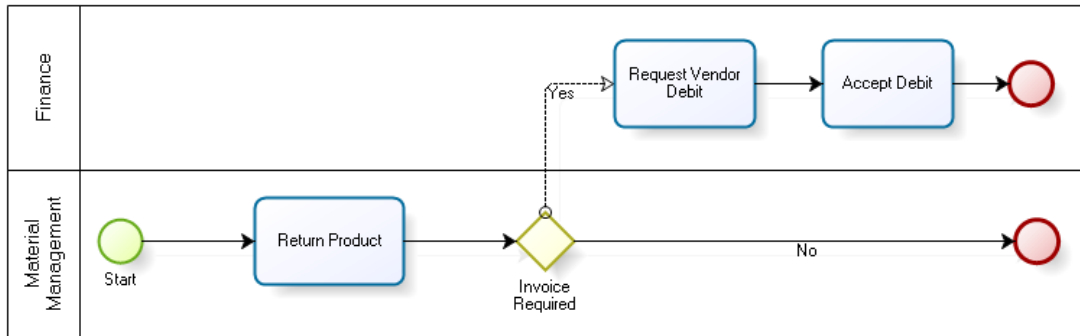


Figure XXXIV - Business Flow: Supplier Return to Debit

The following sub-processes form it:

- **Return a product.**
- **Request Vendor Debit.** Request that the vendor will apply a discount or accept that it has a debit with the organization.
- **Accept Debit.**

3.1.8. Supplier Return to Replacement

This workflow manages the return of goods the Organization has purchased back to the Vendor, requesting replacement instead of debit (previous outlined flow).

The abstract *Supplier Return to Replacement* process is defined by the following diagram:

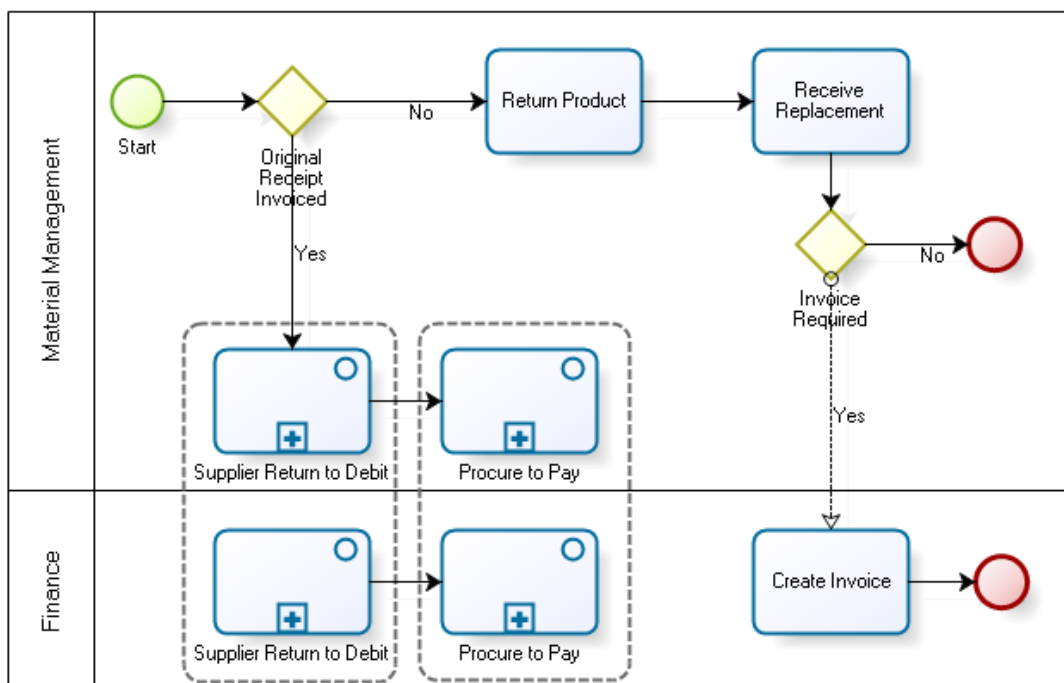


Figure XXXV - Business Flow: Supplier Return to Replacement

The following sub-processes form it:

- **Return Product**
- **Receive Replacement**
- **Create Invoice.** This invoice is created in a similar manner to previous flows, with the exception that it will only apply to the returned product. In case this product had already been invoiced, the return product process should consist on doing two complete flows instead:
 - *Supplier Return to Debit* sub-process
 - *Procure to Pay* sub-process

3.1.9. Wall to Wall Accuracy Physical Count

In Openbravo most of the warehouse movements are created automatically, based on the transactions of sales and procurement processes. However, operating a warehouse also involves several manual activities, which have supporting processes in Openbravo. These activities are gathered in a *Wall to Wall Accuracy Physical Count* business process, and are: **Goods Movements** and **Physical Inventory**.

Goods Movements transfers inventory between storage bins or warehouses. Possible reasons for goods movements are:

- Goods received at warehouse from another part or warehouse
- Inventory movement due to conversion of goods.
- Inventory movement for goods taken out of consigned inventory for consumption.

Physical Inventory is the activity to count individual items in stock at a particular point in time, and to update their inventory count within the system. Possible reasons for physical inventory are:

- To verify the physical amount, condition and location of inventory items.
- To identify, document and add items to inventory list that are on-hand and meet qualifying criteria, but are not currently shown as part of the inventory.
- To ensure that legitimately transferred or disposed of items are no longer carried on the inventory listing.
- To identify any missing or damaged items that need to be located, repaired or replaced.

These manual activities don't require interaction with a Business Partner, whether it is Customer or Vendor, they are thus, not eligible for B2B Integration activities, even if Vendor Managed Inventory and Kanban systems are in place.

The abstract *Wall to Wall Accuracy Physical Count* is defined by the following business process diagram:

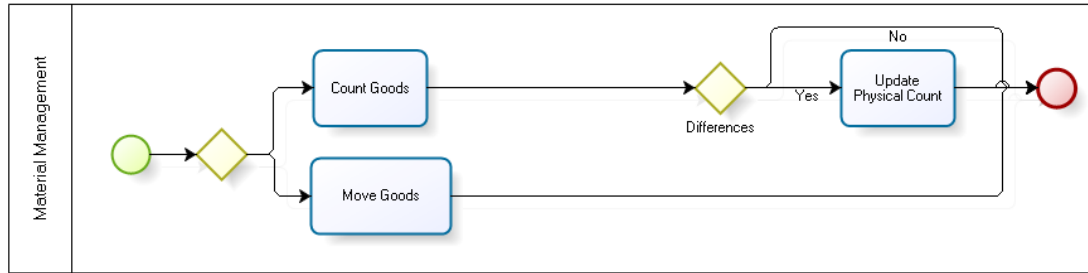


Figure XXXVI - Business Flow: Wall to Wall Accuracy Physical Count

These are the main sub-processes in it:

- **Counting goods and updating physical count.** Counting goods or also called physical inventory consists on personally inspecting a warehouse bin and writing down how many quantities of each item there are. Later they are confronted to the Book Quantity, which indicates the theoretical amount of a particular product in that bin. The Quantity Count will be edited by the user once he has the actual count of that same product within the warehouse.
- **Moving goods.** If the Warehouse Staff needs to move goods from one storage bin to another.

3.1.10. Asset Acquisition to Dispose

This workflow manages the whole life-cycle of an asset:

- requesting an asset
- purchasing a product
- defining the product as an asset
- defining depreciation/appreciation of an asset
- accounting for depreciation/appreciation on a yearly or monthly basis
- disposing an asset

Assets are everything of value that is owned by the company, for example equipment, vehicles, buildings and stock. In Openbravo ERP, you can record the value of assets and their amortization.

Amortization by percentage means that a specified percentage of the total value of the asset is recorded each amortization period (monthly or yearly). For example 1% of the total value per year until the amortization is complete. Amortization by time means that the full amortization figure is split according to the amortization period, for example 10,000 euro recorded as 1000 euro a year for ten years.

The abstract *Asset Acquisition to Dispose* is defined by the following business process diagram:

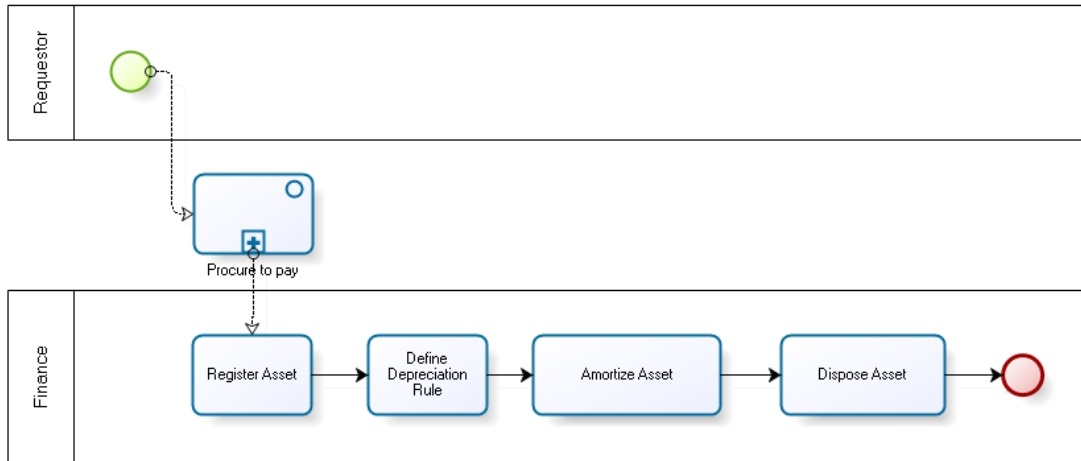


Figure XXXVII - Business Flow: Asset Acquisition to Dispose

Asset Management is an entirely internal operation, very closely related to accounting. No B2B Integration is expected in this business flow.

3.1.11. Tax Payment

Tax Payment is the process by which an organization settles payments done or received to the authorities. Two kind of general tax payment types exist.

- Regular Taxes. To be paid annually, or exceptionally, involve the payment of a quantity that depends on the accounting information.
- VAT Taxes. The calculation will be the difference between the sales tax, which we received from customers but is not ours to keep, and the purchase tax, which we paid to vendors but that we should receive back.

Taxation on buying or selling of products and services is already taken into account by the previous processes. In order to receive or make the payment of taxes to the authorities, no B2B Integration is needed.

Integration between the authorities systems' and Openbravo ERP is not covered by B2B Integration, because of two main reasons:

- It has to be covered as a specific case for each country, thus no generalization can be made.
- The authorities will not be flexible or will not be willing to use systems which are not their own, on the contrary, possibly will force the Organization to use their own system.
- It is already covered in professional localization packs in the Openbravo ERP community.

3.1.12. Sales Order to Cash

The blood flowing through every commercial entity is customers' orders. All businesses exist to make money from selling products (or a service, which can be considered as a type of product

in our context). Like other core processes, Order-To-Cash can be broken down into the following sub-processes:

- Order To Shipment.** The information included typically an order number and date, shipping and receipt dates, a customer purchase order number, the buyer's name and address, the shipping address (if different), and a list of the items ordered, including quantity and warehouse storage location.

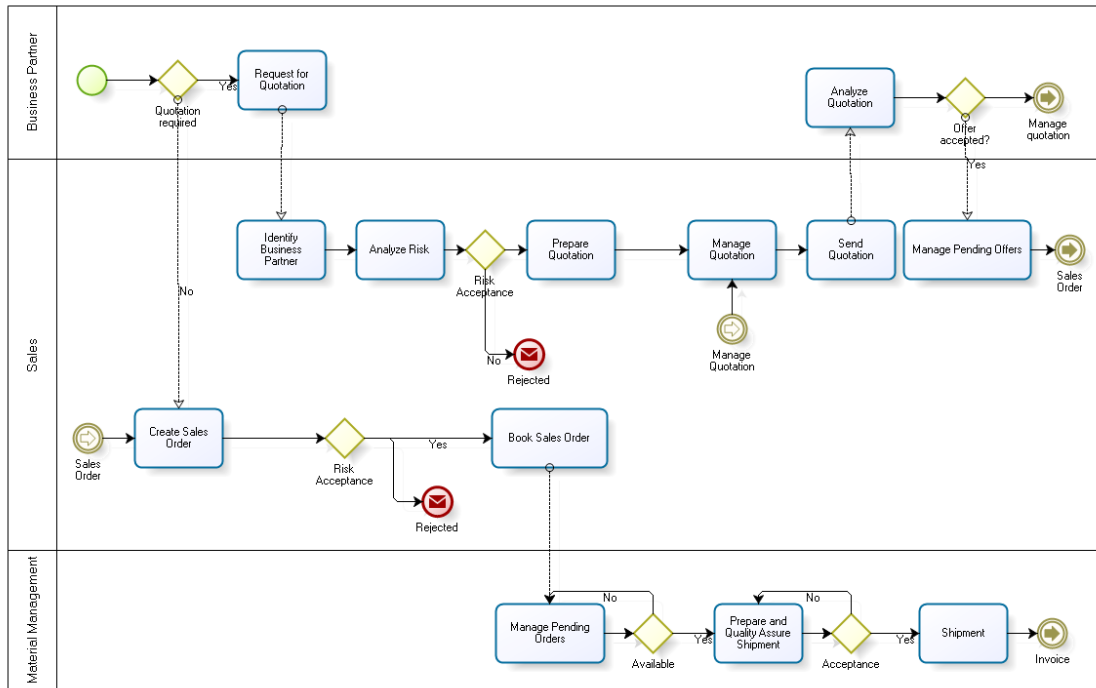


Figure XXXVIII - Business Flow: Order To Shipment

- Customer Invoice to Cash.** Billing a sales order creates a record of debt owed to you by your customer for a sale. When you bill a sales order, your accounts receivable ledger increases by the amount of the bill. When the bill has been paid, the Cash is increased.

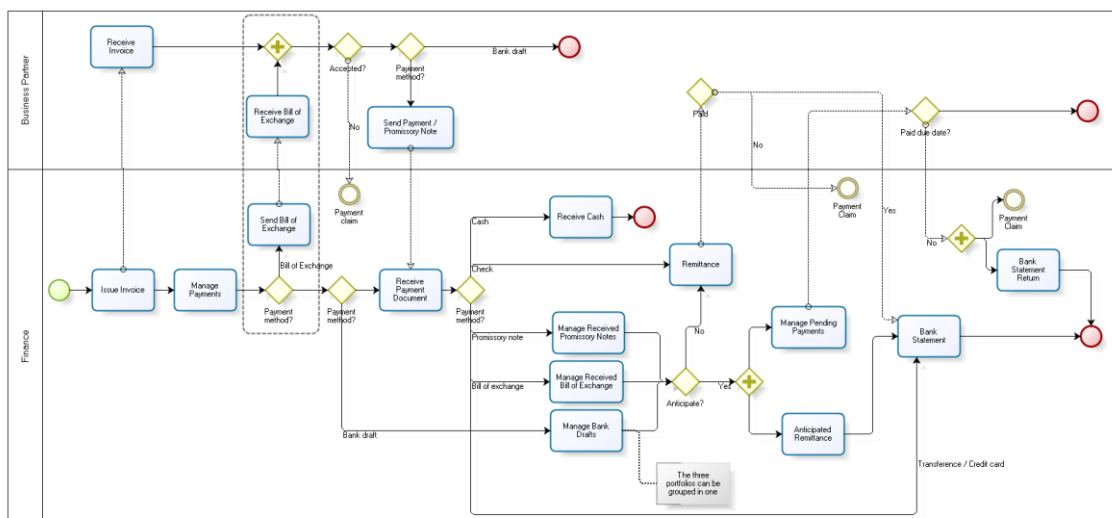


Figure XXXIX - Business Flow: Customer Invoice to Cash

- **Collection.** The process of collecting amounts receivable (the majority of which is normally made up of "open" invoices that have passed their due date). Effective collection is vital to both physical cash flow and cash flow projections.

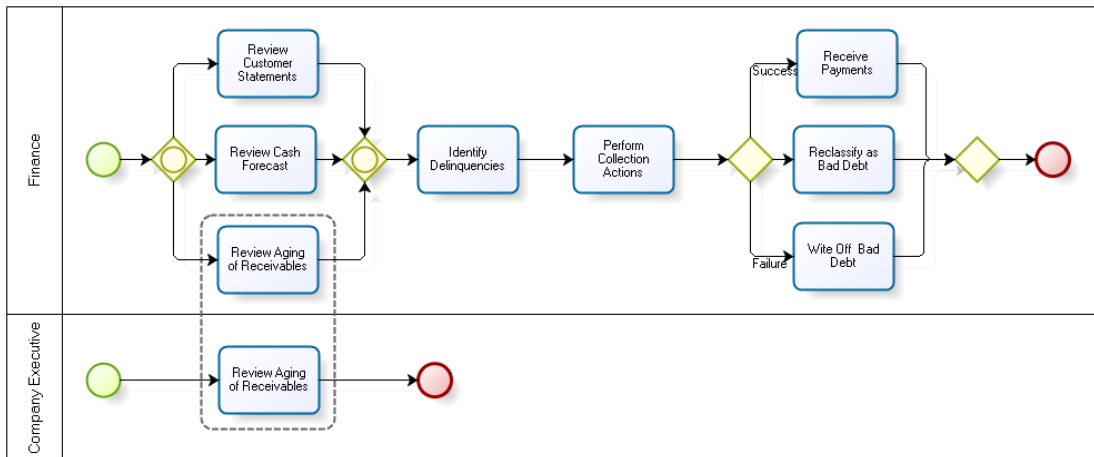


Figure XL - Business Flow: Collection

3.2. B2B Integration Functional Gap

Openbravo ERP provides a reduced yet well rounded and powerful set of basic functionalities. Per design, these main flows that have been described before represent the *Core* of the product, a minimum common definition of the needs of an SME, the target type of organization to which Openbravo ERP is designed for.

These functionalities will almost never fit completely the whole set of business processes that a given organization performs in its normal, day to day operation. The objective of a successful ERP product is not to cover every possible business case, but to deliver the best and most easy to use experience around what it can be considered the best practices or main business flows. If it had too many options or functionalities, it will be too big and bloated for most organizations, and even then, it will be missing some functionality that could be considered core for some other organizations. This is why most ERP packages, and specially Openbravo ERP, give the implementer the means and tools to adapt and extend it.

Within the implementation of an ERP system, it is always assumed that some degree of customization will have to be performed. Openbravo ERP has a modular architecture, which allows customizing existing capabilities and extending them beyond standard scope using downloadable modules, while maintaining stability through regular, official patches that correct bugs and errors. Moreover, by design it is Open Source Software, which permits to see how each piece of code is executed, giving the users and implementers the possibility to modify everything to fit its own uses.

On previous chapters the challenges and desirable characteristics of an integrated supply chain have been discussed. The previous section of this chapter presented the existing core business flows as they are designed to be performed in Openbravo ERP. There have been identified a

set of areas in which Openbravo ERP is not covering or is well lacking the ability to perform B2B Integrated Supply Chain Management, both from the perspective of the organization's management and from the vendor's perspective. These gaps have been divided into the following areas, which are further developed after: *Platform Functional Gap*, *Purchase Order or PO Management Gap*, *Shipping and Goods Movement Gap*, *Warehouse Management Gap*, *Invoice Settlement and Payment Acknowledgement Gap*, *Vendor / Supplier Relationship Management Gap*, and *Master Data Management Gap*.

3.2.1. Platform Functional Gap

Openbravo ERP has been initially designed as a traditional ERP, aimed for SMEs, following the business processes that have been described in the first part of this chapter. Some design decisions have been taken and implemented with this target in mind, hence the software and product architecture that was described in Chapter 2 of this text. The *Platform Functional Gap* refers to the missing basic functionality missing or not too well aimed with Business-To-Business connectivity in mind.

- Openbravo ERP is designed to be used by employees or agents within the boundaries of the organization. Customers, Vendors and other stakeholders were not taken into account when designing the *user access and security model*. Currently, there is no simple way of enabling customers or vendors to access the ERP with a default role, function, and security context.
- Evidently, Openbravo ERP is lacking the whole data structures to define the *parameters of the B2B Integration*, both at the Global, Organization and Business Partner level.
- Openbravo ERP is lacking the whole data structures to define Business Partners (Vendors or Customers) as *B2B Contacts*. Not only it will be advisable to keep a record of which Business Partners are enabled for B2B Integration, but also which users or contacts from each business partner have credentials to access the system, and which default role will they take when they log in to the system (this role will define their security access to different parts of the system).
- The current email engine of Openbravo ERP is very basic. It only allows for basic, plain text email templates, which doesn't allow for inserting complex structures such as hyperlinks, images, tables, etc. It will be advisable to have a *complex HTML Emailing Engine*. Furthermore, there is no tracking of the sent or unsent emails, and in case an error will occur, there is no way to monitor the emailing log, or to re-send the email.
- The only way of transferring and sharing documents in Openbravo ERP, is currently through the *attachments* functionality. While it is terribly useful, it has no version tracking, audit trail, or notifications. There is no way either a user can upload documents to its own workspace, and edit them online.
- Openbravo boasts a complete audit trail for documents. When configured, every record or document created, modified, or deleted by any user will be recorded, and it can be used to track and correct both human mistakes and malicious data modification and destruction. This is a basic feature needed for any B2B Integration initiative. However, it had been designed with security purposes. A more complex B2B *Activity*

Log, to track actions performed by internal and external users, and used as a base for a notifications feed.

Most if not all of this lacking functionality should be addressed even before starting to design B2B Business Processes which imply both internal and external users.

3.2.2. PO Management Gap

Purchase Order exchange and Quotations are an integral part of B2B activities and Supply Chain Management. Traditionally, and as it has been demonstrated by the core business flows in Openbravo ERP, this section happens outside of the ERP, or in the case it should happen in it, requires a re-keying of the information back into the ERP when a new version of a purchase order, or a new quotation document is received by the purchase staff of the Organization.

PO Management Gap refers to the missing functionality to engage in efficient, smooth B2B integrated transactions on the purchase side of the supply chain.

- The biggest and most important gap in the current functionality refers to the obligation of creating a PDF file of a purchase order to send it to a Vendor. Even though this document can be attached to an automatically sent email, the email body and subject doesn't contain information or important parameters about this PO, such as Scheduled Delivery Date, Delivery Notes, Address, etc. There is no electronic document template.
- Second in importance, Openbravo lacks a method to manage *Quotations or Purchase Order versioning*. Any approximation using the current system is incomplete and requires the re-keying of information for every PO version or Quotation.
- Since the PO Management flow happens outside the system, it is impossible to capture information about the Vendor's acknowledgement, acceptance or rejection of a Purchase Order.
- The above problem may lead to situations in which the delay in the response or even lack of response of the Vendor may render one or more Purchase Orders invalid. In addition, PO validity time or deadline information is not currently stored nor printed in Openbravo ERP.
- It is impossible to capture information about the timing and responses of a Vendor in the whole quotations and PO management workflow.
- Specification changes to PO are impossible to perform, or to track at all.
- The current system of PO management requires a follow up of every quotation and PO, even if it is routine work. B2B Integration best practices advise for managing only exceptional PO or quotations, and let automated and integrated systems handle the routine transactions. The time required for management of Purchase Orders is very big.

3.2.3. Shipping and Goods Movement Gap

ERP systems were conceived as an evolution and integration of separate Manufacturing Planning and Inventory systems. This is because goods movement is an integral part to most businesses, and the transportation of goods makes up the biggest percentage of Business-To-Business transactions.

Openbravo ERP takes care of incoming and outgoing shipments by creating Goods Receipt and Goods Shipment documents, which usually follow closely purchase or sales orders, and purchase or sales invoices. It does, however doesn't get past the point in which the goods arrive or abandon each of the Organization's warehouses, meaning that it has no control whatsoever on when and how will the goods arrive to a customer or leave the vendor's warehouse, or how the transit of these goods is taken.

B2B Integrated Supply Chain management advocates for total awareness of the status of shipments and goods, from the moment they are created, prepared and shipped by the vendor, to the moment the goods are finally in the customer's hands.

In this sense, the biggest gap is the absence of an *Advanced Shipment Notice* document in Openbravo ERP. This document is sent by the supplier to the Organization and is a detailed account of the exact products with their quantities that are being dispatched, along with a scheduled delivery date.

When the goods finally arrive this Advanced Shipment Notice document should be converted, or serve as the base for a traditional Goods Receipt document.

Vendor visibility about their orders to be delivered, shipments in transit, and shipments which have partially or completely arrived is not immediate either for the Organization, not counting that the vendor doesn't have an easy access to this data.

3.2.4. Warehouse Management Gap

Even though warehouse operations would apparently be considered an internal-only activity, the truth is that the field of warehouse management shows one of the highest potential for big payoffs in B2B Integration with the lower investment of resources. Openbravo ERP's warehouse and inventory management has been evidently designed without Vendor / Customer integration in mind, so there are many gaps within the current functionality.

- The system doesn't track the minimum stock, maximum stock or preferred order quantity for products. Supply Chain Management advocates for lowering inventory as an abstract mean of lowering costs and making the chain more efficient. If there are no ways of knowing if a product is under or over stock, and how much stock would be advisable to order, the system cannot engage in efficient Supply Chain Management practices, let alone Business-To-Business integration.
- Vendor Managed Inventory is not possible if there is no easy way for the Vendor or even the supply chain manager to check inventory levels for their managed stock.
- There is no way of signaling the Vendor if a product that it supplies is being depleted and will need replenishment soon. This is managed directly through purchase orders instead of letting the Vendor replenish it by itself. Kanban methods are not possible with the current configuration.
- There is no supply or stock forecasting.

3.2.5. Invoice Settlement and Payment Acknowledgement Gap

Invoicing and Payment execution are processes that are exclusively managed by the internal staff of the Organization in the current business flows. Moreover, information is only gathered within internal sources, and both vendor and customer invoices have to be manually entered into the system, whether they are received in paper or electronic form. There is a big gap for B2B Integration in Invoice Settlement and Payment Acknowledgement.

- Matching invoices against purchase orders or incoming shipments is a tedious process, even more if the information being managed by both the Organization and Vendor is misaligned.
- Purchase Invoice information, even if provided directly by vendors, has to be entered by an internal user into Openbravo ERP. There is no way a Vendor can enter Purchase Invoice information, or validate an invoice that has been keyed into the system by an internal user.
- When payments are executed, Openbravo ERP will consider them correct even if there is an error outside the process in the ERP. Openbravo lacks the functionality to allow a Vendor to acknowledge a payment.

3.2.6. Vendor / Supplier Relationship Management

Vendor (Supplier) relationship management is carried out outside of Openbravo ERP. For B2B Integration strategies to take place, effective means of VRM / SRM have to be integrated within the B2B transactions of the ERP.

- Openbravo ERP lacks a way of notifying vendors about important issues in a massive way, either regularly (Newsletter) or exceptionally (Massive Email).
- There is no way of tracking and recording the history of interactions between a Vendor and its Supply Chain Manager responsible. An Issue tracker could be advisable.
- Other means of communication (Email, Phone Calls, Fax ...) happen outside the ERP. Instant messaging through a chat inside the Openbravo Workspace could be implemented to further.
- Notes functionality has just been incorporated into Openbravo ERP. While useful, it is not entirely usable for B2B communication within shared documents, since it has no visibility and security rules whatsoever. Of course, this comes along with the traditional design of an ERP to be accessed only by internal organization users.

3.2.7. Reporting Functional Gap

Reporting is the extraction and display of useful data from the ERP to make business decisions. B2B Integration activities, if implemented, will need an increased level of awareness and data sharing, thus the reporting functional gap is big for Openbravo ERP.

- As it has been stated before, supplier performance data is not captured in the ERP, this means that it is impossible to report so.

- Workspaces and widgets have great potential to become the “control center” for supplier and customer operations, regardless if an internal or external users access the system. However it lacks of useful widgets to dig down important information for B2B activities: order status, stock levels, payment and invoice status, etc.

3.2.8. Master Data Management Gap

Master Data Management Gap refers to the missing capabilities and data structures in Openbravo ERP regarding efficient B2B integrated activities. In the current state of development of Openbravo ERP, and given the narrow functional footprint it has acquired in the latest versions, a lot of necessary inputs and data have no place to be stored right now.

- Vendor and Customer master data details are managed by the Organization’s internal users. Not only basic data such as Name, Tax ID details, etc. but also addresses, bank account numbers, contacts, etc. The external users cannot verify, change or cancel this data themselves. This is both error-prone and a small burden for internal users in charge of master data quality.
- Contract Management functionality is missing from Openbravo ERP. This would mean both a direct archiving of official documents in the ERP and the history of the relationship commercial terms: Delivery terms, payment rules, contract duration, etc.
- Products need fields for storing its preferred vendor, minimum and maximum stock, order quantity, vendor naming and reference, and the possibility for the vendor to upload pictures, schematics, and logos. Moreover, new products and delisted products are not updated correctly if this is an internal user’s task.
- Pricing information has to be up to date if purchase orders and quotations created in the ERP are to be accurate. Pricelists change over time and currently the internal users have to re-key or modify them based on some rules or document received by the Vendors. Apart from the obvious errors introduced by manual recording of this data, internal users spend quite a time doing this. Vendor’s could take care of this operation to be sure that the purchase orders and invoices they receive always have the proper price applied.
- Personal Data of external users and contacts is very error-prone. In a B2B integrated environment, this data would be managed by the vendor or customer instead of internal organization’s staff.
- B2B Integration, if it is to be implemented into Openbravo ERP, will need a whole new set of data structures to store the configuration parameters.

Chapter 4. Functional and Technical Design

4.1. Functional Design

The functional design of the B2B Integration Module tries to describe the expected behavior of the extension module and provides a detailed description of the new functionality and customizations, explained with use cases and business diagrams.

This document also serves as the basis for the *Technical Design*, where it is explained how to accomplish technically the new features described in here.

4.1.1. Design considerations and constraints

Some previous considerations:

- The B2B Integration Functional Gap which was described on the previous chapter is the base for the Functional Design. Features and Functionality designed will try to solve the problems that have been explained and fill in the missing functionality.
- The new features and transactions shall integrate with the existing business flows.
- A holistic and integrated approach is to be taken. The extended features shall work together well and integrate themselves with the features of Openbravo ERP.
- When possible, existing capabilities of Openbravo ERP will be re-used and/or adapted. Re-designing some new artifact from scratch is not advisable.
- When designing a new feature or a functional solution to one of the described problems, it has to constraint by the existing Openbravo ERP technology. Even though the technical solution to the functional design is covered in the *Technical Design* document, designs have to comply with what is possible to do with the existing capabilities of the platform technology and its development tools.

4.1.2. B2B Integration Module purpose and scope

As it has been explained in previous chapters, Business-To-Business integration strategies require a big level of trust between business partners (Vendor-To-Organization and Organization-To-Customer), establishing shared goals and coordination to streamline the transactions in which many parties are involved.

This module aims to construct a platform around which common B2B transactions can be managed and tracked by the Organization, Vendors and Customers. It is built around Openbravo ERP, and will integrate with the existing Openbravo ERP core main flows.

Most of the important B2B operations, and those which are better eligible for a dedicated integration, are those done on the supply side of the materials, information and money flow described in chapter 2. While giving customers a good service and capabilities is important, big improvements and benefits can be obtained by streamlining the Supply Chain, and in this area is in which automated systems and integrated ERPs can be more useful.

While always maintaining extensibility, in its first version or iteration, the B2B Integration module will focus on improving and integrating Supplier relationships and interactions with

Vendors. In this sense, from the point of view of the Organization, only purchase flows are going to be taken into consideration.

Based around the Openbravo ERP platform, and fully integrated with the ERP, the B2B Integration module will provide a Vendor Portal, in which both internal and external users will be able to log-in and perform transactions against the same set of data.

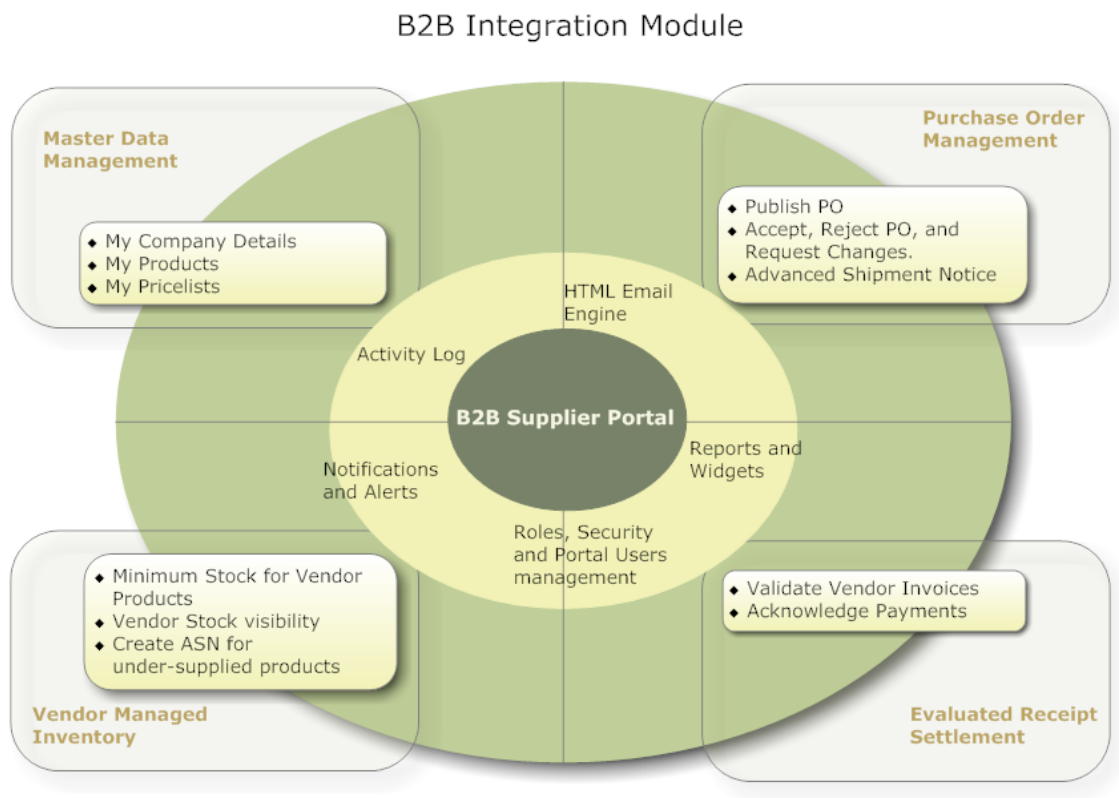


Figure XLI - B2B Integration Module Overview

The platform should be leveraged so that what had been initially designed as a web-based ERP for internal users will make it easier to create and manage external users (Vendors), communicate with them and track the transactions that they perform, hence the *B2B Integration Platform Extensions* which will be described afterwards.

It should also bring value for Vendors to engage them in the use of the Vendor Portal; hence it will provide extended reporting functionality and specific widgets so that Vendors can manage their company details, products, pricing, stock levels of which they are responsible, and the documents and transactions they have made. The B2B Integration module with the Vendor Portal will provide thus *Master Data Management Extensions* and new *Reports and Widgets*.

The Vendor Portal streamlines existing business flows (*Purchase Order Management Extensions*), provides greater transparency and data availability (*Evaluated Receipt Settlement*

Extensions) and enables new Business-To-Business collaboration models (*Vendor Managed Inventory Extensions*).

4.1.3. User roles & profiles

The B2B Vendor Portal will feature two general roles for both the Vendor and an internal user within the Organization. These two roles will have distinct views and will be the actors of the processes described below.

These are general roles that don't have to follow strictly the role model in Openbravo ERP; they serve as examples and as the performers of the transactions that the B2B Integration Module will allow. Besides, an existing user of the ERP can be mapped to several roles, and all these actions could be performed by a different role than the standard one.

B2B Vendor Contact

The B2B Vendor Contact role will be the one used by the Vendor's responsible person to engage in B2B activities. The general role will have access to all of the processes described below, but a consultant could fine tune this role and create some others for specific partners, for example, B2B Vendor PO Manager, B2B Vendor Stock Manager, etc.

The B2B Vendor Contact should only be able to see information relative to the Business Partner it belongs to, and only from the organization where its Business Partner belongs.

This role will be mapped to users which are also User / Contacts of a B2B enabled Business Partner, under the Contact tab of the general Business Partner window.

Supply Chain Manager

The Supply Chain Manager role will have all the permissions needed to engage in any of the processes described below from the Organization's point of view. A consultant could fine tune other roles such as Procurement Manager, Inventory Manager or Finance and Accounting roles to remove permissions or add B2B permissions as well.

The B2B Vendor Contact should be able to see information relative to every B2B enabled Business Partner belonging to its Organization.

This role will be mapped to users which are also considered employees, under the Employee tab of the general Business Partner window.

4.1.4. B2B Integration Platform Extensions

The main challenge for creating a B2B Vendor Portal as an extension module of Openbravo ERP, integrated with the existing business flow, data structures, and technology platform, is correctly designing the participation and collaboration model that will allow Vendors to log-in to the system to check information and perform transactions.

It is important to keep in mind that Openbravo ERP was designed to be used by employees or agents within the boundaries of the organization. Customers, Vendors and other stakeholders were not taken into account when designing the *user access and security model*. Currently, there is no simple way of enabling customers or vendors to access the ERP with a default role, function, and security context.

In addition to leverage the platform to allow creating and maintaining external users that will log into the system, some other platform improvements derived from this necessity and other functional gaps have to be made, such as a new emailing engine with HTML capabilities, an Activity Log, or even specific, B2B Configurations.

B2B Integration Configuration

- i. B2B Integration capabilities will occur at many levels. Initially, a Vendor, which is a type of Business Partner, will belong to an organization which will in turn belong to a client (Note: organization and client in this context are referring to the organization model of data division of Openbravo ERP, described in the [section 2.1.4 Openbravo ERP – Functionality](#)).
- ii. A new window and data structure will be created to store **B2B Global Parameters**, values that will be used for the whole Vendor Portal, for example, *Portal Name, Portal URL, Email Configuration*, etc. Further fields can be added to this data structure if needed by the different functional B2B extensions.
- iii. A new window and data structure will be created to store **B2B Organization Parameters**. For each B2B enabled organization, a record can be created to store values specific to this organization, for example, *Default SCM Role, Default SCM User, Default Vendor Contact Role*, etc. Further fields can be added to this data structure if needed by the different functional B2B extensions.
- iv. A new window and data structure will be created to store **B2B Vendor Parameters**. For a Vendor to be able to engage in B2B Integration activities, a record will have to be created, and it will store all kinds of specific values for this Vendor. Most B2B functional extensions will add fields to this data structure, such as preferred lead time, whether the Vendor will auto-accept purchase orders, or auto-acknowledge payments, etc.
- v. It will also be possible to access the B2B Vendor Parameters data structure from a new tab under the Business Partner tab.
- vi. Most of this configuration values should only happen once, when installing the B2B Integration module for the first time.

Users, Roles, Wizards

- i. Each new external user will be mapped to a User Contact of a Business Partner. Some prerequisites will have to be met:
 - a. B2B Global Parameters are properly configured
 - b. The Business Partner's organization is properly configured in B2B Organization Parameters.
 - c. The Business Partner is a Vendor
 - d. The Vendor is configured under the B2B Vendor Parameters tab.

- ii. If the above are correct, any contact belonging to this Vendor could be granted access to the Vendor Portal. Since configuring users and permissions is very tricky under Openbravo ERP, and it has been designed mostly for internal users, a wizard will be created.
- iii. The new **“Grant Access to Vendor Portal”** button will be visible in the Contact tab under the Business Partner window. When this process is executed, the following actions will happen:
 - a. The process will check that the prerequisites are met.
 - b. The process will make this contact an ERP user, granting him a new username (same as its email) and a random password.
 - c. A default role and organization will be assigned to this new user.
 - d. An email will be sent to this organization’s default Supply Chain Manager, informing about the new user.
 - e. An interactive, HTML email will be sent to this contact. It will feature instructions, the username and password, links to log-in into the Vendor Portal, and an advice to change the auto-generated password.
- iv. The new user will be able to log-in into the system and make B2B activities in with his own name, on behalf of the Business Partner he belongs to.
- v. If a user has already been granted access to the Vendor Portal, a new button will be visible, named **“Revoke Access to Vendor Portal”**. When run, the process will delete the username and password from the contact record, thus preventing the user from accessing the ERP again.

Activity Log

- i. When users external to the organization are able to access the system and making changes to the same set of data that the Organization is using to manage its operations, the standard Audit Trail feature is not enough to track users and data for both informational and security purposes.
- ii. An Activity Log is to be created. This will be an utility to be used by any process to easily log what activities are happening.
- iii. A new window and data structure will be created to see the following information about an activity:
 - a. Activity Name
 - b. Date and Time
 - c. Sent Email
 - d. Tab and Record involved
 - e. URL

HTML Email Engine

- i. An enhanced emailing engine to be used by any process is necessary to increase communication and awareness between internal and external users.
- ii. HTML Templates are also important so that it is possible to insert complex structures such as hyperlinks, images, tables, etc.

- iii. A new window will be created to track sent and un-sent emails, and to be able to manually re-send an email.

4.1.5. B2B Integration Master Data Management Extensions

In Chapter 3 it was discussed that proper and efficient B2B Integration needed increased data sharing and transparency capabilities that the current Openbravo ERP platform lacks. Allowing Vendors to access the same information system used in transactions may benefit very easily if they are also allowed, encouraged and made responsible for the quality of their Master Data.

Company Details

- i. A new window mirroring the general Business Partner data manager, named “**My Company Details**” will be created. It will allow a B2B Vendor Contact to check and correct information concerning its own organization.
- ii. Basic data such as Name, Tax ID, Company Website, or other references will be able to be edited.
- iii. The B2B Vendor Contact will be able to create, edit and delete information concerning both its Locations and Bank Accounts
- iv. It will be entitled to check which Discounts are in place.
- v. It will be able to create and edit Contact persons of its company. Granting access to the Vendor Portal will still be a Supply Chain Manager’s prerogative.
- vi. The B2B Vendor will be able to edit which of its purchase pricelist versions is the one to be used by the Supply Chain Manager in Ordering, Shipping, and Invoicing.
- vii. Also details such as Payment Terms, Payment Method, Maturity Dates and Vendor Parameters should be visible but not editable.

Products

- i. A new window mirroring the general Product data manager, named “**My Products**” will be created. It will allow a B2B Vendor Contact to check and correct information concerning its own product catalog.
- ii. The B2B Vendor Contact will be entitled to edit the Name, Tax Category, Description and Weight of their products.
- iii. Pictures, schematics, logos and other reference files could be attached to each product.
- iv. Name and Reference used internally by the Vendor will also be editable, to facilitate communication between Vendor and Organizations databases.
- v. Two lower level tabs, *Price* and *Transactions* will be available in read only mode, so that the B2B Vendor can check what prices does the product takes in different pricelist versions, and in which transactions (Orders, Shipments, Invoices) it has been involved.
- vi. A new button “**Add New Product**” will be created, so that the B2B Vendor can update its own catalog. It will ask for the Name, Description, Vendor Product Code, Vendor Product Name, Unit of Measure and Tax Category of the new product.

The image shows a software mockup for adding a new product. It features a green header bar with the text "Add new product". Below the header, there are several input fields: "Name" (a text box), "Description" (a text area), "Vendor Product Code" (a text box), "Vendor Product Name" (a text box), "UOM" (a dropdown menu with "Unit" selected), and "Tax Category" (a dropdown menu with "Exempt" selected). At the bottom of the form are two buttons: "OK" and "Cancel".

Figure XLII - Add New Product mockup

- vii. The **Search Key** and **Product Category** for the new product will be configured by the Supply Chain Manager in the B2B Vendor Configuration tab.

Pricelists

- i. The new Vendor Portal functionality allows Vendors to be in charge of their own products pricelist information, a new window for B2B Vendor Contact view has been created, named **"My PriceList Versions"**.
- ii. This functionality requires that each B2B enabled Vendor has its own Pricelist, which is configured under the Vendor tab of Business Partner general window by the Supply Chain Manager.
- iii. The B2B Vendor will be able to edit the description of the pricelist assigned to it.
- iv. Under the tab "Price List Version", the B2B Vendor Contact will be able to create new pricelist versions, and to add products and their Net Standard Price, Net List Price and Price Limit.
- v. A new button, **"Copy Pricelist Version"** will be created so that the process of copying an old pricelist version to use it as a template for a new one is possible.

Pricelist Alerts

- i. A new alert will be created so that the B2B Vendor Contact is aware that the **Pricelist Expiration Date**, a new field created in Pricelist Versions tab, has been reached.
- ii. The Supply Chain Manager should receive the alerts for all pricelist versions belonging to a B2B enabled Vendor which have expired as well.

4.1.6. Purchase Order Management Extensions

Purchase Order and Quotations Management functionality was poor and unprepared for integrated B2B transactions, as it had been explained before in Chapter 3, the PO Management gap was big.

Essential missing functionality and shortcomings were the transmission of Purchase Order document in PDF form from the Organization to the Vendor, the lack of quotations or purchase order versioning functionality, and the absence of Advanced Shipment Notice documents in Openbravo ERP.

The new functionality to develop will leverage the new capabilities of the platform which allow Vendors to log into the Vendor Portal and perform transactions.

The new workflow is summarized in the following diagram:

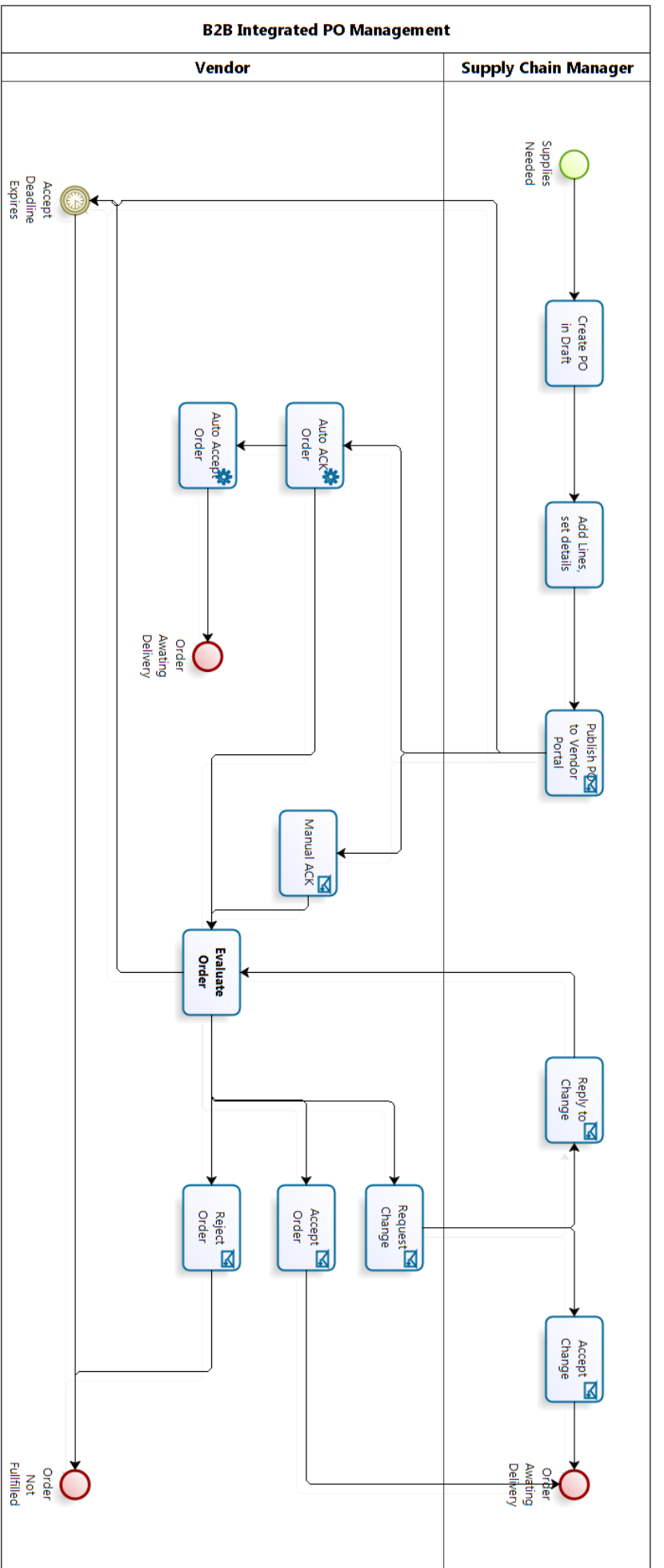


Figure XLIII - B2B Integrated PO Management

Publishing Purchase Orders

- i. Following a necessity of supplies, the Supply Chain Manager will start by creating a standard Purchase Order into Openbravo ERP. First it will enter the data in the header of the purchase order, corresponding to general data such as Vendor, Scheduled Delivery Date, Delivery Location, Notes, etc. The note will be created in Draft status.
- ii. If the selected Business Partner is enabled for the B2B Vendor Portal, and the selected document type is marked as a B2B Purchase Order document, the B2B Integrated PO Management flow will start.
- iii. In this case a new field called “B2B Purchase Order status”, always visible, will change from “N/A – Not Applicable” (which is not modified for traditional, non b2b integrated purchase orders) to “DR – Draft”.
- iv. A new button and new fields will be visible:
 - **Publish PO to Vendor Portal** (Button)
 - **Publish Date** (Date field)
- v. The Supply Chain Manager will start filling the lines for this purchase order. Only products offered by this Vendor should be initially selectable, since the selected pricelist version should already belong to this Vendor, and it should already be updated by the Vendor (see section named *Pricelists* on this functional design). Products and Order quantities should be filled; the price will automatically be filled by the one present in the pricelist.
- vi. When the Supply Chain manager considers the order to be ready to be sent to the Vendor, he or she will run the “Publish PO to Vendor Portal” process, selecting a **Deadline Date** for this purchase order.
- vii. This process shall send an email with the proper details of the PO to the main B2B Contact of this Vendor. It will change the status of the purchase from “DR – Draft” to “PUB – Published”. It will also make visible the following fields:
 - **ACK Date** (Date field)
 - **Acceptance Date** (Date field)
 - **Deadline Date** (Date field)
 - **Vendor proposed changes** (Text box field)
 - **SCM proposed changes** (Text box field)
- viii. Now the Vendor is notified and the PO has to be acknowledged, accepted or rejected by them. It could also request a change, keeping in mind that once the Deadline Date is met, the order will be considered cancelled (thus changing its status to “OUT – Deadline expired”).

Acknowledgement, Acceptance and Change Requests of POs

- i. The Vendor Contact will receive a notification via email that a new PO has been published in the Vendor Portal. This email will have a summary of the header data of the purchase order, and it will explain the next steps in the Purchase Order Management flow. The original PDF hard copy of the purchase order could be attached.
- ii. Depending on how this B2B Business Partner is configured, three different outcomes are possible:

- a. If it is configured so, the order can be **auto-acknowledged**. This will change the B2B Purchase Order Status to *“ACK – Acknowledged”* automatically.
 - b. If the above is true, and the vendor has been configured that way, the order can also be **auto-accepted**. This will change the B2B Purchase Order Status to *“OK – Accepted”* automatically.
- iii. If none of the above is true, the B2B Purchase Order Status will not change, waiting for a manual interaction of the Vendor.
- iv. The Vendor Contact will log in to the application. A direct link to the purchase order document will be available, which will take the user to a tab opening the new B2B Vendor Orders window. This window is a copy of the Purchase Order window, but with restricted fields modification capabilities and only the Vendor’s purchase order will be visible for the Vendor Contact. The new B2B Dates fields and the new text boxes fields will also be visible.
- v. If the order hadn’t been yet acknowledged, the **“ACK Order”** button will be visible.
- vi. Once the purchase order has been acknowledged, three buttons will be visible for the vendor contact:
 - **Accept Order.** This will change the B2B Purchase Order Status to *“OK – Accepted”*, and the Purchase Order will be considered valid. This implies the acceptance of the prices, products, and quantities, and the terms and conditions, including dates, etc. The Vendor now has an obligation to ship the items to the Organization and has the right to invoice for the goods or services.
 - **Reject Order.** The order will be rejected and will be considered cancelled and not valid. This will change the B2B Purchase Order Status to *“KO – Rejected”*
 - **Request Change.** Pressing this button will make a pop-up appear with a text box asking for a description of the proposed change. Once the process is run, this description will be appended to the *“Vendor Proposed Changes”* field, and will in turn change the B2B Purchase Order Status to *“NSCM – Changes Proposed by Vendor”*. The SCM will be notified through the Activity Log system.
- vii. If a change is requested, the Supply Chain Manager can access this purchase order again to either:
 - a. **Accept the change.** The SCM will have to manually make this change to the necessary fields of the purchase order. This will change the B2B Purchase Order Status to *“OK – Accepted”* and the vendor will receive an email notification.
 - b. **Reply to the proposed change**, which will in turn change the B2B Purchase Order Status to *“NB2B – Replied to change”*, the vendor will receive again an email notification. A new button appropriately called **“Reply to Vendor Changes”** will be used for this, which will ask the SCM for a response. The text box *“SCM Proposed Changes”* will be updated.
- viii. In any case, the Supply Chain Manager will be notified, since all these transactions will appear in the Activity Log.

- ix. If a Purchase Order has been accepted in the B2B Purchase Order Management flow, it should change its standard Openbravo ERP status from *Draft* to *Complete*. Up until this flow and the whole B2B Integration module has been published and tested in real-world scenarios, this process won't be automatized, in order to assure the proper behavior of the standard Openbravo ERP business flows.

Advanced Shipment Notice for a PO

- i. Once a PO has been accepted and has been completed, regardless its origin, had it passed through the new *B2B PO Management* flow or the standard *Procure To Pay* business flow suggested by Openbravo ERP, the goods will have to be moved from the Vendors location to the Warehouse and Storage Bin specified in the Purchase Order.
- ii. If the Vendor is enabled for B2B Integration, it can issue an Advanced Shipment Notice (ASN) the moment the goods leave their location. This ASN can be created using a PO as a basis, as it is described here, or manually out of a stock necessity in a Vendor Managed Inventory, as shown in section [4.1.7 Vendor Managed Inventory Extensions – Create ASN for undersupplied products](#) of this chapter.
- iii. When the Vendor Contact logs into the Vendor Portal, under the Purchase Order window there will be a new button, “**Create ASN from PO**”, if the given order's B2B Purchase Order Status is “*OK – Accepted*” and the order has been completed.
- iv. This process will create a new document of the type “**Advanced Shipment Notice**”, a sub-document of the type “Goods Receipt”. Essentially it is the same as a standard Openbravo ERP incoming shipment in Draft status, with new visible fields in both the Vendor Contact and Supply Chain Manager views.
- v. A single Purchase Order can be delivered in more than one shipment, thus a new ASN, each one expressing the proper product and quantities should be created in this case. In addition, the Vendor Contact should make sure that the “**Estimated Delivery Date**” is correct for each Advanced Shipment Notice.
- vi. A new field, always visible, for Incoming Shipments / Goods Receipt will be created: “**B2B Shipment Status**”. Standard Goods Receipt documents will always have this field to “*N/A – Not Applicable*”; while newly created ASNs will have changed the status to “*DR – Draft*”.
- vii. The Vendor will now do its normal picking and packing operations, and will gather the necessary products in the quantities specified in the ASN document. Once the goods are ready and the shipment leaves the Vendor's location, the new “**Shipment Dispatched**” check should be ticked. This will automatically change the B2B Shipment Status from “*DR – Draft*” to “*ER – En Route*”. The Supply Chain Manager will be notified via the Activity Log.
- viii. Once the shipment physically arrives to the Warehouse, and the Supply Chain Manager checks that it contains the proper products and quantities, it should tick the checkbox “**Shipment Arrived**”, which will in turn trigger the change of the B2B Shipment Status from “*ER – En Route*” to “*SA – Shipment Arrived*”.
- ix. Two new date fields, “**Dispatch Date**” and “**Shipment Arrival Date**” will be created. They will be automatically updated (or nulled) depending on the *B2B Shipment Status*.

The following diagram displays the possible status changes and the triggering actions of the key “B2B Purchase Order Status” and “B2B Shipment Status” flags:

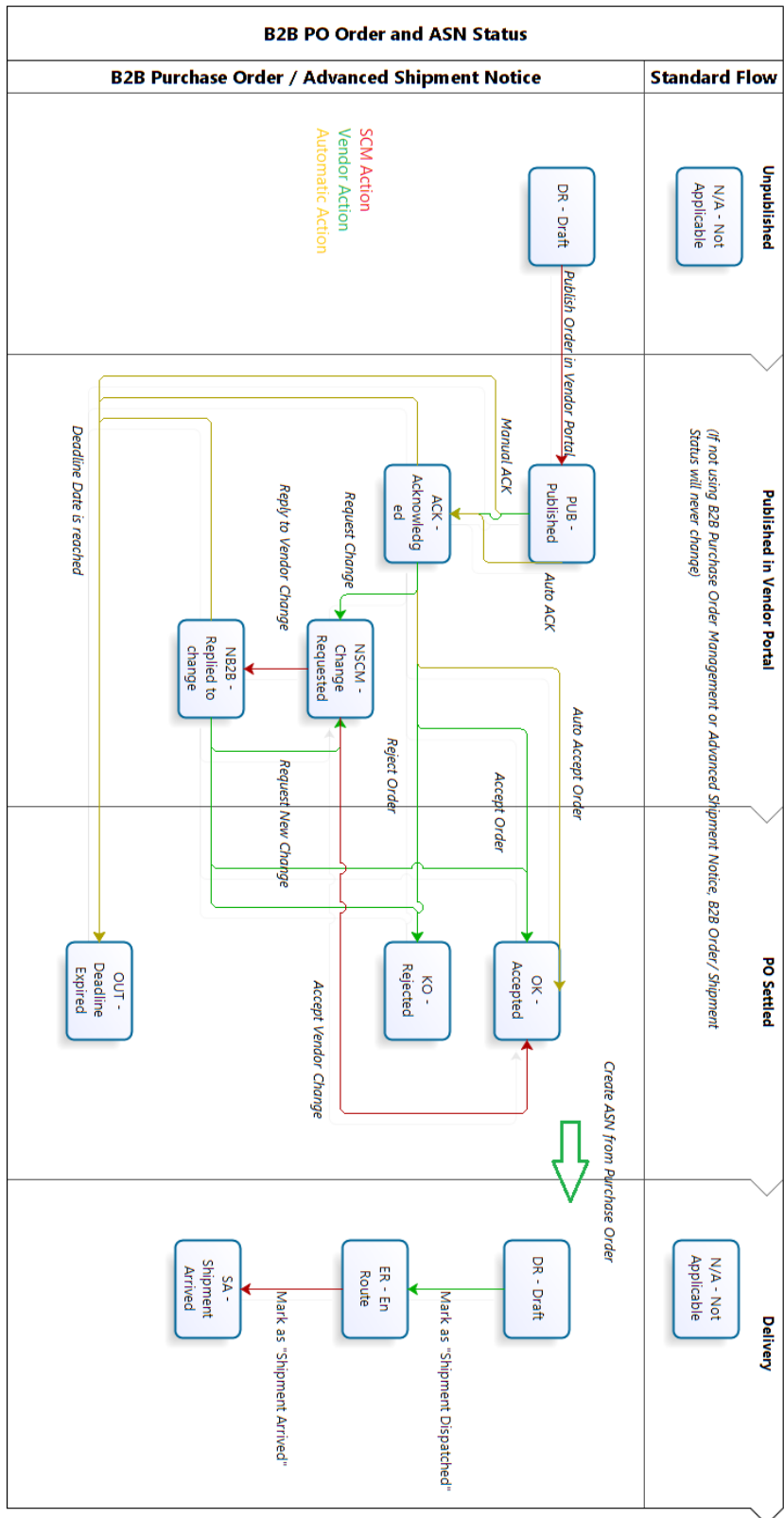


Figure XLIV - B2B PO Order and ASN Status

4.1.7. Vendor Managed Inventory Extensions

Minimum Stock and Order Quantity for Products

- i. For B2B enabled vendors, it should be possible to check on the stock level of their products in order to engage in VMI operations.
- ii. Indicators regarding whether this stock level is too much or too little are necessary, so two new fields are going to be created in the Product window, under the *B2B Product Configuration* field group: **Minimum Stock** and **Maximum Stock**. It should be possible to leave these integer fields empty.
- iii. In VMI and Kanban methodologies, Vendors will resupply products under stock without the need for purchase orders. A new checkbox field (**Auto-Resupply**), and a new integer field (**Resupply Qty**) will be created under the *B2B Product Configuration* field group.

Vendor Managed Storage Bin

- i. In Vendor Managed Inventory operation, the Vendor not only is allowed but actively encouraged to check the stock of the products of which it is responsible for the supply.
- ii. In order to enable this functionality in the Vendor Portal, it should be possible to configure a Storage Bin for a B2B enabled Vendor. A new field in the B2B Vendor Configuration tab called “**Vendor Managed Storage Bin**” is going to be created. It will allow selecting one of the storage bins of the B2B enabled organization.
- iii. A new window will be created for the Vendor Contact view, also with the name **Vendor Managed Storage Bin**. It will be a copy of the standard warehouse window, without the main, 0 level tab, and will only allow to see details of the storage bin configured in the step above.
- iv. Only this vendor’s products and this vendor’s b2b transactions (shipments) should be visible under the *Bin Contents* and *Product Transactions* tab.

Stock Alerts

- i. In Vendor Managed Inventory operation, it’s the Vendor’s responsibility to assure that the stock quantity of some products is always above the minimum stock level, and it is the Vendor’s task to automatically supply and ship these products.
- ii. A new alert will be created to notify the Vendor Contact when a product is under stocked and needs replenishment. The triggering conditions will be:
 - a. The product has a Vendor, and this vendor is a B2B enabled Business Partner. This B2B enabled Business Partner should also have configured a Storage Bin.
 - b. The product’s *Minimum Stock* field is not null.
 - c. The product’s *On Hand Quantity* in this storage bin is less than the minimum stock indicated in the Product’s record.
- iii. This alert will have the following aspect: “*Product X is under minimum stock in Storage Bin Y*”, and should link to the appropriate record on the newly created “*Bin Contents*” tab in the Business Partner view.
- iv. In case a product under stock has the “Auto-Resupply” checkbox marked, a new alert should be created with the following aspect: “*Product X should be shipped in Y numbers to Storage Bin Z*”, where Y is the configured **Resupply Qty** number indicated in the Product’s record.

Create Advanced Shipment Notice of undersupplied products

- i. Once a PO has been accepted and has been completed, regardless it origin, had it passed through the new *B2B PO Management* flow or the standard *Procure To Pay* business flow suggested by Openbravo ERP, the goods will have to be moved from the Vendors location to the Warehouse and Storage Bin specified in the Purchase Order.
- ii. If the Vendor is enabled for B2B Integration, it can issue an Advanced Shipment Notice (ASN) the moment the goods leave their location. This ASN can created using a PO as a basis, as described in section [4.1.6. Purchase Order Management Extensions – Advanced Shipment Notice for a PO](#) of this chapter, or manually out of a stock necessity in a Vendor Managed Inventory, as below.
- iii. When the Vendor Contact logs into the Vendor Portal, under the Vendor Managed Inventory menu item, a new process “New Advanced Shipment Notice” will be available.
- iv. This process will create a new document of the type “**Advanced Shipment Notice**”, a sub-document of the type “Goods Receipt”. Essentially it is the same as a standard Openbravo ERP incoming shipment in Draft status, with new visible fields in both the Vendor Contact and Supply Chain Manager views.
- v. The process will bring a pop-up which will ask for the **Dispatch Date**, **Estimated Delivery Date** and **Description** for the new ASN.
- vi. It will also ask if it is necessary to **add new lines** for each vendor-managed product under stock should be added. In this case, the defined *Resupply Quantity* for each product will be used as the *movement quantity* in each line.

The screenshot shows a dialog box titled "New Advanced Shipment Notice". It features a green title bar with a close button (X) on the right. Below the title bar is a grey bar with an information icon (i). The main content area is white and contains the following elements:

- A checkbox labeled "Add lines for autosupply products under stock" which is checked.
- A "Dispatch Date" field with the value "15-06-2011" and a calendar icon.
- An "Estimated Delivery Date" field with the value "15-06-2011" and a calendar icon.
- A "Description" text area.
- Two buttons at the bottom: "OK" and "Cancel".

Figure XLV - New Advanced Shipment Notice mockup

- vii. A new field, always visible, for Incoming Shipments / Goods Receipt will be created: **“B2B Shipment Status”**. Standard Goods Receipt documents will always have this field to *“N/A – Not Applicable”*; while newly created ASNs will have changed the status to *“DR – Draft”*.
- viii. The Vendor will now do its normal picking and packing operations, and will gather the necessary products in the quantities specified in the ASN document. Once the goods are ready and the shipment leaves the Vendor’s location, the new **“Shipment Dispatched”** check should be ticked. This will automatically change the B2B Shipment Status from *“DR – Draft”* to *“ER – En Route”*. The Supply Chain Manager will be notified via the Activity Log.
- ix. Once the shipment physically arrives to the Warehouse, and the Supply Chain Manager checks that it contains the proper products and quantities, it should tick the checkbox **“Shipment Arrived”**, which will in turn trigger the change of the B2B Shipment Status from *“ER – En Route”* to *“SA – Shipment Arrived”*.
- x. Two new date fields, **“Dispatch Date”** and **“Shipment Arrival Date”** will be created. They will be automatically updated (or nulled) depending on the *B2B Shipment Status*.

4.1.8. Evaluated Receipt Settlement Extensions

Evaluated Receipt Settlement is the process by which Invoices are jointly evaluated and agreed before being issued, by using some information sharing system. In Openbravo ERP, purchase invoices match purchase order and / or goods receipts if the core, standard *Procure To Pay* business flow is followed.

Additionally, payment tracking and payment acknowledgement present functional gaps for B2B Integration as it has been described in Chapter 3.

Leveraging the Vendor Portal platform, once the previous Purchase Order Management and Vendor Managed Inventory extensions are in place is quite immediate, both for Invoices or Payments

Create Invoice from Shipments and Purchase Orders

- i. Purchase Invoices will be created normally from purchase orders or goods receipt, regardless if these documents belonged to B2B integrated operation or not.
- ii. These Invoices will normally match what had been ordered or shipped, thus discrepancies between the ones issued by the vendor should be minimal.
- iii. Invoices belonging to the B2B Vendor will be available to check for the B2B Vendor Contact Role, including its payment details and payment plan, in a new window named **“My Invoices”**

Validate Invoice

- i. To engage in ERS operation, invoices must be validated by the B2B Vendor.
- ii. A new field for invoices corresponding to B2B enabled vendors will be available, **“B2B Invoice Validation Status”**. Available statuses are *“Not Applicable”*, *“Not Validated”*, *“Auto-Validated”* and *“Manually Validated”*.

- iii. In the *B2B Vendor Configuration* tab, it should be possible to specify if a B2B Vendor will auto-validate invoices, or after how many days without validating an invoice, it will change to AUTO status without manual interaction.
- iv. If the above is not true, the B2B Vendor Contact should be able to run the process “**Validate Invoice**”, by which it will accept that the data contained in that invoice is correct, and change the invoice status from “*Not Validated*” to “*Manually Validated*”.

Acknowledge Payment

- i. Incoming Payments for B2B Vendors can be acknowledged. This means that the B2B Vendor indicates that some of the payments present in the Organization’s ERP records arrived to them. A button will be present in the special view “**My Payments**” to do this operation, setting the *B2B ACK Status* of the payment to OK.
- ii. If an enabled B2B Vendor is configured to Auto-ACK payments, in the B2B Vendor Configuration tab, payment’s *B2B ACK Status* will automatically be set to AUTO.

4.1.9. B2B Integration Reports and Widgets

Reporting and information visibility is one of the key features and benefits of implementing a Supplier Portal. Furthermore, enhanced information availability counts will be a great pitch to convince Vendors to make use of the Vendor Portal functionality.

It should then be possible to navigate from the reports and widgets to the corresponding shown record.

Matched Purchase Order Lines with Shipment Lines

- i. Tracking of Shipment and Purchase Orders can be difficult in separated views for Purchase Order Management and Vendor Managed Inventory shipment windows.
- ii. A copy of the “Matched Purchase Orders” window is going to be made available for B2B Vendors. It should be renamed to “**Matched PO Lines**” and only lines in which the purchase order’s business partner is the same as the logged Vendor should be visible.
- iii. This will allow the B2B Vendor to track lines of their shipments directly to their purchase orders in which they were ordered.

Matched Purchase Invoice Lines with Shipment Lines

- i. Tracking of Shipments and their corresponding Invoices can be difficult in separated views for Vendor Managed Inventory shipment and Evaluated Receipt Settlement invoice windows.
- ii. A copy of the “Matched Invoices” window is going to be made available for B2B Vendors. It should be renamed to “**Matched Invoice Lines**” and only lines in which the shipment’s business partner is the same as the logged Vendor should be visible.
- iii. This will allow the B2B Vendor to track lines of their invoices directly to the shipment line in which the goods were delivered.

Product Summary Widget

- i. Product related information is scattered around various windows and configuration views around Openbravo ERP and the Vendor Portal: *My Products* window, *Pricelist Version* window, *Storage Bin Contents* view, etc.

- ii. A new widget called **My Products Summary**, summarizing information from various sources and data structures will be created to give a condensed view and to serve as the control center for Vendor products.
- iii. For each product, it will present the following data:
 - a. Storage Bin and Product Name
 - b. Quantity on Hand and Minimum Stock
 - c. Auto-Resupply check and Resupply Quantity.
 - d. Net Standard Price and List Price for the currently active pricelist version

My Products Summary							
Storage Bin	Product	Quantity on Hand	Minimum Stock	Resupply Qty.	Auto Resup	Net Standard P	List Price
USA111	Raw material A	10,000		30,000	<input type="checkbox"/>	2.00	5.00
spain111	Raw material A	0		30,000	<input type="checkbox"/>	2.00	5.00
spain111	Raw material A	10,000		30,000	<input type="checkbox"/>	2.00	5.00

Figure XLVI - My Products Summary widget mockup

My Purchase Orders and My B2B Orders Widgets

- i. It will be necessary to present a summary of completed Purchase Orders for the B2B Vendor so that it is aware of which orders are in process or have been delivered. A new widget will be created, called **My Purchase Orders**, presenting the following data for each PO:
 - a. Document Number and Order Reference
 - b. Order Date and Scheduled Delivery Date
 - c. Description (expanded view) and Delivery Notes
 - d. Also in expanded view: Warehouse, Amount and Currency

My Purchase Orders				
Document Number	Order Reference	Order Date	Scheduled Delivery Date	Delivery Notes
800005		26-05-2011	31-05-2011	
800008		27-05-2011	30-05-2011	
800004		23-05-2011	23-05-2011	

Figure XLVII - My Purchase Orders widget mockup

- ii. In order to facilitate the management of various ongoing Purchase Order engaging in the PO Management business flow, a new widget will be created, named **My B2B Orders**, presenting the following data:
 - a. Document Number
 - b. B2B Order Status
 - c. Publish Date and Deadline Date
 - d. Amount and Currency
 - e. Scheduled Delivery Date
 - f. ACK Date and Acceptance Date (both in expanded view)

My B2B Orders						
Document Number	B2B Order	Publish Date	Deadline Date	Amount	Currency	Sched Delivery
800005	OUT	26-05-2011	25-05-2011	37.46	EUR	31-05-2011
800008	OK	27-05-2011	28-05-2011	0.00	EUR	30-05-2011

Figure XLVIII - My B2B Orders widget mockup

My Invoices and Open Invoices Widgets

- i. It will be necessary to present a summary of history of purchase invoices for the B2B Vendor. A new widget will be created, called **My Invoices**, presenting the following data for each invoice:
 - a. Document Number and Order Reference
 - b. Invoice Date
 - c. Payment Method and Payment Terms
 - d. Total Amount and Currency
 - e. Validation Status
 - f. In expanded view only: Payment Complete checkbox, Paid Amount, Outstanding amount, Days till due, and Due Amount

My Invoices							
Document Number	Order Reference	Invoice Date	Payment Method	Payment Terms	Total An	Currency	Validatio
10000004		27-05-2011	6 (Spain)	90 days	0.00	EUR	N/A
10000002		27-05-2011	6 (Spain)	90 days	-37.46	EUR	N/A
10000001		27-05-2011	6 (Spain)	90 days	37.46	EUR	N/A
10000003		27-05-2011	6 (Spain)	90 days	37.46	EUR	N/A
10000000		18-05-2011	6 (Spain)	90 days	246.40	EUR	AUTO

Figure XLIX - My Invoices widget mockup

- ii. In order to help the B2B Vendor on generating its Cash Flow Reports, a new widget will be created, named **My Open Invoices**, presenting the following payment data for each invoice:
 - a. Days till due and Due amount and Currency
 - b. Document Number
 - c. Paid Amount and Outstanding amount
 - d. Total Amount and Payment Method

My Open Invoices									
Days Till Due	Due Amount	Currency	Document Number	Paid Amount	Outstanding	Total Amount	Payment Method		
0.00	0.00	EUR	10000004	0.00	0.00	0.00	6 (Spain)		
90.00	0.00	EUR	10000000	-246.40	0.00	246.40	6 (Spain)		
90.00	0.00	EUR	10000003	-37.46	0.00	37.46	6 (Spain)		
Σ		0.00		Σ	-283.86	Σ	0.00	Σ	283.86

Figure L - My Open Invoices widget mockup

My Shipments Widget

- i. It should be very useful to see a summary of present and past shipments and advanced shipment notices, and their status.
- ii. A new widget summarizing information directly from the shipments window will be created.
- iii. For each shipment header of the B2B Vendor, it will present the following data:
 - a. Document Number and Order Reference
 - b. Dispatch Date and Movement Date
 - c. Description
 - d. Warehouse (only in expanded view)
 - e. B2B Shipment Status

My Shipments					
Document Number	Order Reference	Dispatch Date	Movement Date	Description	Ship. St
PRE-500009-SUF		30-05-2011	30-06-2011		DR
PRE-500008-SUF		30-05-2011	31-05-2011		DR
PRE-500006-SUF		27-05-2011	30-05-2011		SA
PRE-500007-SUF		30-05-2011	30-05-2011	hellou byayby	DR
PRE-500004-SUF			27-05-2011		N/A
PRE-500003-SUF			27-05-2011	(*R*=PRE-500004-SUF)	N/A
PRE-500005-SUF			27-05-2011		N/A
PRE-500002-SUF			27-05-2011		N/A
PRE-500001-SUF			27-05-2011	(*R*=PRE-500002-SUF)	N/A
PRE-500000-SUF		25-05-2011	20-05-2011		ER

Showing 10 of 11 rows Show all...

Figure LI - My Shipments widget mockup

My Payments Widget

- i. It should be very useful to see a summary of the account present in Openbravo ERP related to payments made to the B2B
- ii. A new widget summarizing information from the different payments views will be created.
- iii. For each payment executed against the B2B Vendor, it will present the following data:
 - a. Document Number and Reference No.

- b. Date
- c. Amount and Currency
- d. Payment Method
- e. ACK status

My Payments						
Document Number	Reference No.	Date	Amount	Currency	Payment Method	ACK Status
700002		25-08-2011	37.46	EUR	6 (Spain)	N/A
700001		25-08-2011	37.46	EUR	6 (Spain)	N/A
700003		25-08-2011	37.46	EUR	6 (Spain)	N/A
700000		16-08-2011	246.40	EUR	6 (Spain)	OK

Figure LII - My Payments widget mockup

4.2. Technical Design

This section describes the software artifacts, data structures and Application Dictionary components that will conform the B2B Integration Module. The reader of this section should be familiar with Openbravo ERP technology, modularity and development tools.

4.2.1. Technical Design Considerations

The objective of this document is to describe the technical implementation of the functional description of the new features. It should also serve as the basis for the Development Plan, by delimiting the development cost of each functionality, and to the developer as a basis for coding.

Some considerations:

- Every Data Structure, Software Artifact and AD item, should be developed in a way that allows data sharing between the Vendor and the Organization, while preventing the Vendors to see data which correspond to another Business Partner.
- B2B Integration should be organization-independent. This means that some organizations may engage in this activity while others won't.
- Existing core business flows should not be affected by B2B integrated operation.
- Extending the core and developing new functionality is always preferable to customizing exiting features.
- B2B Integration info and messages will be fully developed in English, but should be fully translatable by other modules

4.2.1. Module Architecture

Following design functional division, the B2B Integration module will be divided in 5 sub-modules, with tight dependency architecture.

Dependency is a constraint put in place by the developer so that some pieces of code and data structures need some parent ones to be installed properly. Some modules are dependent on other, as a means of abstracting this phenomenon. Updating a single module may not affect the others, and publishing and revising code is made much simpler using this architecture.

In addition, future expansions and translations of the B2B Integration module can take advantage of this programmatic module architecture. Independent Translations or extensions can also be applied to every single module.

The following diagram represents the dependency structure:

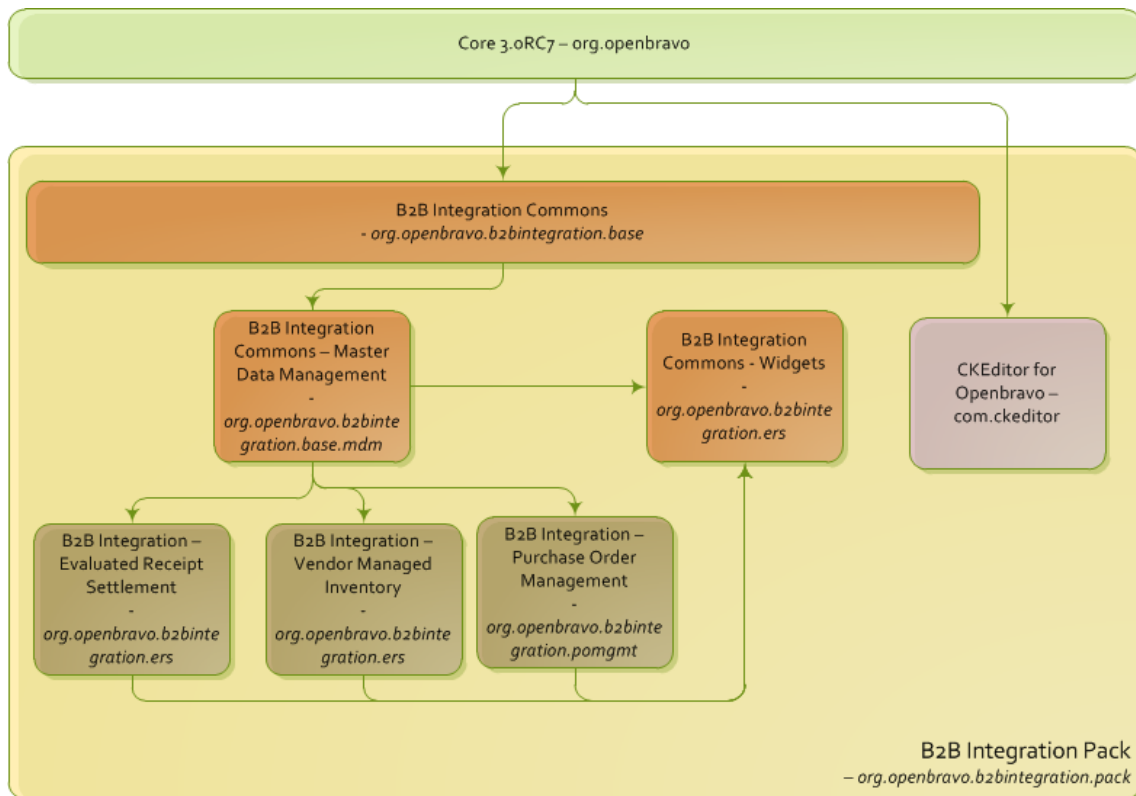


Figure LIII - B2B Integration module architecture

Each module will be now described in more detail:

B2B Integration Commons

This module provides the common framework and software artifacts for the modules in the B2B Integration Pack.

Name	B2B Integration - Commons
Java Package	org.openbravo.b2bintegration.base
DB Prefix	B2B
Dependency	Core 3.0RC7

Software artifacts and data structures belonging to this module will be used by several other dependent upstream modules. Therefore, common data structures used by more than one module in the B2B Integration pack should belong to this module. Things included in this module are the HTML emailing framework, Activity Log, basic authentication for vendors and wizards for new B2B Partners and users, datasets containing base roles.

B2B Integration Commons

This module provides the software artifacts and required sources and processes to provide easy Master Data Management within the B2B Integration Pack.

Name	B2B Integration - Commons Master Data Management
Java Package	org.openbravo.b2bintegration.base.mdm
DB Prefix	B2BMDM
Dependency	B2B Integration - Commons

This module implements modified windows for the most common Masterdata windows, such as Business Partner, Product, Price List, etc. Vendors can create, maintain and update their Company Details, Products, and create different versions of their pricelists.

B2B Integration – Purchase Order Management

This module provides the functionality to engage in B2B Purchase Order management and B2B negotiation.

Name	B2B Integration – Purchase Order Management
Java Package	org.openbravo.b2bintegration.pomgmt
DB Prefix	B2BPO
Dependency	B2B Integration - MDM

It includes the basic framework for B2B Purchase Order Management and Negotiation. Purchase Orders can be published to the B2B Vendor Portal and your Vendors will receive notifications via email and alerts. You may track how your vendors acknowledge, accept, reject or negotiate POs, and create Advanced Shipment Notices based on an agreed PO.

B2B Integration – Vendor Managed Inventory

This module provides the functionality to engage in VMI (Vendor Managed Inventory).

Name	B2B Integration – Vendor Managed Inventory
Java Package	org.openbravo.b2bintegration.vmi
DB Prefix	B2BVMI
Dependency	B2B Integration - MDM

This module allows for Vendors to control their Products in shared or vendor managed warehouses. Vendors will receive notifications when a product is close or beyond their minimum stock, and they can automatically create shipments from a list of under-supplied products. A shipment can now be created by Vendors so that it is tracked from the moments the goods are dispatched (Advanced Shipment Notice), and the Supply Chain Manager can notify the Vendor when the shipment arrives.

B2B Integration – Evaluated Receipt Settlement

This module provides the functionality to engage in ERS (Evaluated Receipt Settlement)

Name	B2B Integration - Evaluated Receipt Settlement
Java Package	org.openbravo.b2bintegration.ers
DB Prefix	B2BERS
Dependency	B2B Integration - MDM

This module includes the views and uses so that vendors can validate their invoices and acknowledge their payments, with no contradicting information stored on emails or physical documents.

B2B Integration - Commons Widgets

This module provides the widgets for all the common B2B functionality, both in the Business Partner and Supply Chain Manager views.

Name	B2B Integration - Commons Widgets
Java Package	org.openbravo.b2bintegration.ers
DB Prefix	B2BWIDG
Dependency	B2B Integration - MDM

This module implements widgets to check and administer Business Partner's orders, shipments, invoices, open invoices and payments.

B2B Integration - Commons Widgets

This module provides the widgets for all the common B2B functionality, both in the Business Partner and Supply Chain Manager views.

Name	B2B Integration - Commons Widgets
Java Package	org.openbravo.b2bintegration.ers
DB Prefix	B2BERS
Dependency	B2B Integration - MDM

This module implements widgets to check and administer Business Partner's orders, shipments, invoices, open invoices and payments.

CKeditor for Openbravo - WYSIWYG Html Editor

This module provides the CKEditor source and a sample widget so that it can be used in Openbravo. CKEditor is a text editor to be used inside web pages. It's a WYSIWYG editor, which means that the text being edited on it looks as similar as possible to the reality.

Name	CKeditor for Openbravo - WYSIWYG Html Editor
Java Package	com.ckeditor
DB Prefix	CKED
Dependency	Core 3.0RC7

Because CKEditor is licensed under flexible Open Source and commercial licenses, you'll be able to integrate and use it inside any kind of application. This is the ideal editor for developers, created to provide easy and powerful solutions to their users.

B2B Integration Pack

The B2B Integration framework is a collection of interdependent modules that provide several B2B Integration capabilities to Openbravo, such as Shared Masterdata, B2B PO Management, Vendor Managed Inventory and Evaluated Receipt Settlement.

Name	B2B Integration Pack
Java Package	org.openbravo.b2bintegration.pack
DB Prefix	(none)
Dependency	Core 3.0RC7
Includes	B2B, B2BMDM, B2BPO, B2BVMI, B2BERS, B2B

This pack will always provide the latest version of the B2B Integration Framework, including all core modules. This should be the base standalone module, and extensions should always be dependent on this module. Furthermore, the B2B Integration Framework versioning should always follow that of this module.

4.2.2. Technical Implementation of Platform Extensions

B2B Integration Configuration

- i. Three new tables will be created to hold the B2B Integration configuration:
 - a. B2B_GLOBAL_PARAMETERS (Only one record should be allowed to be created in this table, per client)
 - b. B2B_ORG_PARAMETERS
 - c. B2B_VENDOR_PARAMETERS (this table should hold foreign keys to C_BPARTNER table)
- ii. Along with these tables, corresponding windows and their menu items, all featuring only one tab will be created:
 - a. B2B Global Parameters
 - b. B2B Organization Parameters
 - c. B2B Vendor Parameters
- iii. A new tab under Business Partner will be created, for the new B2B_Vendor_Parameters table. The foreign key to C_BPARTNER will also serve as the link to the parent column in this tab. It should be of level 2, under the "Vendor" tab.
- iv. The above windows will fall under a new menu item of folder type: **B2B Integration Setup**.

Users

- i. User / Contacts of Business Partners records are held in the same table as ERP Users (AD_USER table). For a user to have access to Openbravo ERP, its fields "username" and "password" should be not null. The same record which represents a contact of a B2B Vendor is going to be used as his/her ERP user record.
- ii. A new field of type Yes/NO will be created: "Is Vendor Portal User".

- iii. Two new buttons will be created, added to AD_USER table: “Grant Access to Vendor Portal” and “Revoke Access”. They will feature alternate display logic depending on the value of the above new field.
- iv. Two new standard processes, with no parameters will be created in the Application Dictionary, to hold the wizards These process will correspond to the Java Classes:

<code>org.openbravo.b2bintegration.base.wizards.AddVendorPortalPrivilegesToContact</code>
<code>org.openbravo.b2bintegration.base.wizards.RevokeVendorPortalPrivilegesToContact</code>

- v. The email will be used as the username, and the password should be generated randomly. The user may always change it later.

Data Access

- i. When a new B2B Vendor Contact type of user logs into the system and into the new windows, only data belonging to the proper Business Partner shall be shown. This will be accomplished with HQL queries.
- ii. A new field, clone of *c_bpartner_id*, for AD_USER will be created. Since it will be used in HQL Queries, it will be of type text instead of reference type.
- iii. In order to filter in tabs, the following query *whereclause* template can be used:

<code>e.businessPartner in (select us.businessPartner.id from ADUser as us where us.id=@user@)</code>

Roles

- i. Two new flags to be used by wizards and other processes will exist: “**Is B2B Vendor Contact Role**” and “**Is Supply Chain Manager Role**”, in the table AD_ROLE.
- ii. B2B Vendor Contact and Supply Chain Manager default roles will be stored in a new dataset, to be exported with the rest of the B2B Integration module.

Activity Log

- i. The activity log will be stored in a new table called B2B_ACTIVITYLOG.
- ii. The table shall have references to AD_USER, AD_TAB, AD_RECORD_ID, in addition to the standard fields (activity name, datetime, etc.)
- iii. A new class to give utility features to others, named (*org.openbravo.b2bintegration.base.activitylog.ActivityLogUtils*) will be created.
- iv. The methods inside the class will be:
 - a. Log (User who, String what)
 - b. Log (User who, String what, BaseOObject where)
 - c. Log (User who, String what, B2BEmailInstance sentemail)
 - d. Log (User who, String what, BaseOObject where, B2BEmailInstance sentemail)

HTML Email Engine

- i. A new table to store the names, languages and locations to physical files inside the file system containing the HTML email templates will be created: **B2B_EMAIL_TEMPLATE**,

- and a new window with a single tab: **B2B Email Templates**, under the *B2B Integration Setup* menu.
- ii. Default Email Templates will be stored in a folder under the main module root, with the name: *web/org.openbravo.b2bintegration.base/emailtemplates*
 - iii. A new dataset to hold the B2B Email Templates location, pointing to the B2B_EMAIL_TEMPLATE table should be created.
 - iv. The sent / unsent emails data structure will be composed by three tables, two of them child of the first one:
 - a. B2B_EMAILINSTANCE, containing basic data about every email that has been sent or had to be sent in the system, including email template, user, date and time, status, etc.
 - b. B2B_EMAILPARAMS, parameters both for the body and subject of the email templates in this instance. Contains pair-keys (ParameterName, ParameterValue)
 - c. B2B_EMAILRECIPIENT, original recipients for the email, and the type of recipient (TO, CC or BCC).
 - v. The interaction with this data structure will be done through a new window with three tabs: **B2B Sent Emails**.
 - vi. The main tab will contain a button to send or resend any email instance, using the stored parameters and recipients, and mapped to the following java class: *org.openbravo.b2bintegration.base.emailengine.ReSendEmailButton*
 - vii. The class that can be used by every other process in the system, which will include all the emailing logic, will be named *org.openbravo.b2bintegration.base.emailengine.B2BEmailMonitor*

CKEditor for Openbravo ERP

- i. Publicly available, CKEditor for Openbravo ERP module is included in the distribution. It features a WYSIWYG html editor, which will prove useful when editing and creating new HTML Email Templates files.
- ii. Currently, it presents a widget with HTML editor, but no further functionality. In the future, it could be integrated with the Emailing engine to automatically update and create HTML emails.
- iii. The author of the CKEditor module is the same author of the B2B Integration module, described in this document. CKEditor itself is an independent piece of Open source software.

4.2.3. Technical Implementation of Master Data Management Extensions

Company Details

- i. My Company Details will be a new window, cloned from the Business Partner general window, removing non useful tabs.
- ii. The main tab will be renamed to “My Company Details”. The UI Pattern will be changed to “Edit Only”, non massive, and default edit view.
- iii. Location, User/Contact and Bank Account tabs will be standard (Create, Edit and Delete).

- iv. Vendor and Discounts will be edit only.
- v. Some fields will be made read-only.
- vi. This new window will be intended only for B2B Vendor Contact role, so it will have an HQL Where Clause to restrict data access to the Vendor it belongs to.

Products

- i. Openbravo ERP has a functional gap that prevents from assigning Business Partners as vendors to Products, and the necessary fields to store vendor product information. A new reference to C_BPARTNER will be created in the M_PRODUCT table, along with the new fields *EM_B2BMDM_VENDORPRODUCTCODE* and *EM_B2BMDM_VENDORPRODUCTNAME*.
- ii. These new fields will be added to the general Product window under a new field group, **"B2B Vendor Product Configuration"**.
- iii. My Products will be a new window, cloned from the General product window, removing non useful tabs.
- iv. The main tab will be renamed to "My Products". The UI Pattern will be changed to "Edit Only". Only Product Price and Transactions sub-tabs will be kept, changing the UI Pattern to "Edit Only" and "Read Only", respectively.
- v. This new window will be intended only for B2B Vendor Contact role, so it will have an HQL Where Clause to restrict data access to the Vendor it belongs to.
- vi. In order to allow the B2B Vendor Contact to create new products, a new button along with the corresponding Java process will be created. This process' java class will be *org.openbravo.b2bintegration.mdm.product.AddVendorProduct*.
- vii. The new process will ask for the parameters described in the functional design.
- viii. For new products, the Search Key and Product Category will be taken from configurations made at the B2B Vendor, B2B Organization, or B2B Global level. Those will be foreign keys to AD_SEQUENCE and M_PRODUCT_CATEGORY in all these three tables.

Pricelists and Pricelists Alerts

- i. B2B Integration and the Vendor Portal won't work unless each B2B enabled Vendor has its own Pricelist. Then the B2B Vendor Contact can create and edit new pricelist versions, only for the Pricelist it has been assigned.
- ii. My Pricelists Versions will be a new window, cloned from the main Price List general window.
- iii. This new window will be intended only for B2B Vendor Contact role, so it will have an HQL Where Clause to restrict data access to the Vendor it belongs to.
- iv. The three existing tabs will be kept as they are.
- v. A new field, *EM_B2BMDM_EXPIRYDATE*, will be created. This date type column will be used to show pricelists versions expirations date.
- vi. A new alert will be created to warn both Supplier Chain Managers and B2B Vendor Contacts.
- vii. A new button to Copy Pricelist Versions will be created; the process will use the java class *org.openbravo.b2bintegration.base.mdm.pricing.CopyPriceListVersion*.
- viii. This process will use the *DalCopyUtil* to easily clone price list versions.

4.2.4. Technical Implementation of Purchase Order Management Extensions

Every button in this area should have its own java process.

PO status and new windows and fields

- i. B2B Purchase Order Management functionality will require creating a new column in the C_ORDER table to represent the Shipment Status.
- ii. A new List Reference will be added to represent the possible values of this column:
 - a. N/A – Not Applicable
 - b. DR – Draft
 - c. PUB –Published
 - d. ACK – Acknowledged
 - e. OK – Accepted
 - f. KO – Rejected
 - g. OUT – Expired
 - h. NSCM – Changes Proposed
 - i. NB2B – Replied to Changes
- iii. More new columns will be created in the C_ORDER table: Publish Date, ACK Date, Expiry Date, Acceptance Date, Vendor Proposed Changes and SCM Changes.
- iv. My Purchase Order will be a new window, cloned from the General Goods Receipt window, removing non useful tabs.
- v. The main tab will be renamed to “My Purchase Orders”. The UI Pattern will be changed to “Edit Only”. The window will represent both traditional Purchase Orders and B2B Purchase Orders.
- vi. This new window will be intended only for B2B Vendor Contact role, so it will have an HQL Where Clause to restrict data access to the Vendor it belongs to.
- vii. All the new fields will be added to both My Purchase Orders and Purchase Order windows.
- viii. These fields will only be visible when the B2B Status of the Purchase Order is something different than “N/A – Not Applicable”.

Publishing, Acknowledging and Accepting Purchase Orders

- i. B2B Orders will be a new type of document based on Purchase Orders. A new check will be created in the C_DOCTYPE table to represent this.
- ii. A new trigger will control that only B2B orders change their status from “N/A – Not Applicable” to “DR – Draft”.
- iii. Auto ACK and Auto Accept checks will be created in B2B_VENDOR_PARAMETERS table.
- iv. A new button, “Publish Order” will be only visible for Supply Chain Managers. It will ask for an expiry date. Depending on the configuration of the Auto ACK and Auto Accept buttons, the B2B Order status will change to PUB, ACK or OK.
- v. B2B Vendor Contacts will have a button to ACK the order if it is in PUB status.
- vi. B2B Vendor Contacts will have two buttons to Accept or Reject the order if is in ACK status.

- vii. A new background process will be created that will update every order's B2B Status to "OUT – Expired" when the following conditions have been met:
 - a. The expiry date of the PO is lower than the current date
 - b. The PO is of "B2B Purchase Order Type"
 - c. The B2B Status of the order is not "DR –Draft", "OK – Accepted" or "KO – Rejected".

PO Negotiation

- i. B2B Vendors can request a change for acknowledged purchase orders.
- ii. A new button will be created in the My Purchase Orders window, and this process will change the status to "NSCM".
- iii. If this is the case, a new button will be visible in the general Purchase Order window, the Supply Chain Manager can either Accept the Order or Reply to the changes, which will change the status to "NB2B"

Advanced Shipment Notices

- i. Advanced Shipment Notice functionality will require creating a new column in the M_INOUT table to represent the Shipment Status.
- ii. A new List Reference will be added to represent the possible values of this column:
 - a. N/A – Not Applicable
 - b. DR – Draft
 - c. ER – En Route
 - d. SA – Shipment Arrived
- iii. Two new checks "Shipment Dispatched" and "Shipment Arrived" will be created in the M_INOUT table.
- iv. Two new columns, Dispatch Date and Estimated Delivery Date will be added to the M_INOUT table.
- v. My Shipments will be a new window, cloned from the General Goods Receipt window, removing non useful tabs.
- vi. The main tab will be renamed to "My Shipments". The UI Pattern will be changed to "Edit Only". The window will represent both traditional Goods Shipments and Advanced Shipment Notices.
- vii. This new window will be intended only for B2B Vendor Contact role, so it will have an HQL Where Clause to restrict data access to the Vendor it belongs to.
- viii. Dispatch Date and Estimated Delivery Dates will be added to both windows. The appropriate display logic and read only logic will be applied depending on the B2B Shipment Status.
- ix. Shipment Dispatched check will be added to My Shipments window and Shipment Arrived check will be added to Goods Receipt window.
- x. A new trigger, B2BVMI_SHIPMENTSTATUS_TRG will be added to the database. It will control that the B2B Shipment Status in the M_INOUT table changes accordingly to what both B2B Vendor Contact and Supply Chain Manager perform over their windows.

Create an Advanced Shipment Notice from a PO

- xi. A new button will be added to the My Purchase Orders window. It will only be visible when the B2B PO Status of the order is “OK – Accepted”.
- xii. It will allow the B2B Vendor Contact to create an ASN out of this Purchase Order. Since an ASN is a document type of Goods Receipt, the same functionality of the “Create Lines From Order” in the Goods Receipt window can be copied.

4.2.5. Technical Implementation of Vendor Managed Inventory Extensions

Storage Bin and Products visibility

- i. A new foreign key to M_LOCATOR will be created in the B2B_VENDOR_PARAMETERS table to store which Storage Bin has been assigned to each B2B Vendor.
- ii. The new window, “Vendor Managed Storage Bin” will be a clone of the general Warehouses and Storage Bins window.
- iii. The main tab will be changed to edit only, only the warehouse to which the Vendor’s Storage Bin belongs to will be visible.
- iv. The Storage Bin tab will also be changed to edit only. Only the Storage Bin from this warehouse and belonging to this B2B Vendor will be visible.
- v. Both in Storage Bin Contents and Product Transactions tab, only products belonging to this B2B Vendor will be visible.
- vi. This new window will be intended only for B2B Vendor Contact role, so it will have an HQL Where Clause to restrict data access to the Vendor it belongs to.

Minimum Stock and Stock Alerts

- i. Minimum Stock and Maximum Stock columns will be added to the M_PRODUCT table.
- ii. These fields will also be added to both the general Product window and the My Products window. They will only be editable in the general Product window.
- iii. Two new alerts will have to be created for when a B2B Vendor Product in a Vendor Managed Storage Bin stock is under the minimum and over the maximum values, respectively.

New Advanced Shipment Notice

- i. Two new columns will be added to the M_PRODUCT table, EM_B2BVMI_AUTORESUPPLY and EM_B2BVMI_ORDERQTY. They represent if a B2B Vendor Product has to be automatically resupplied, and how much will be the resupply quantity.
- ii. These fields will also be added to both the general Product window and the My Products window. They will only be editable in the general Product window.
- iii. A new process with its own menu item, will be created, “New Advanced Shipment Notice”, with the java class being *org.openbravo.b2bintegration.vmi.CreateASN*
- iv. Its purpose is to create an ASN, a type of Goods Receipt, this being, a new record in the M_INOUT table and perhaps some new records in the M_INOUTLINE table. This process will have some parameters, and will ask the products needed to be resupplied are to be added as new lines. The complete description can be found in the Functional Design.

- v. The ASN will be of the same type as the one described in this same Technical Design, in the previous section, "[Technical Implementation of Purchase Order Management Extensions](#)".
- vi. The Document Type that the new ASN will have will be taken from a newly created field in the B2B_ORG_PARAMETERS table, a foreign key to C_DOCTYPE.

4.2.6. Technical Implementation of Evaluated Receipt Settlement Extensions

Invoice Validation

- i. Invoice Validation functionality will require to create a new column in the C_INVOICE table to represent the Validation Status of the invoice.
- ii. A new List Reference will be added to represent the possible values of this column:
 - a. N/A – Not Applicable
 - b. P – Pending Validation
 - c. OK – Invoice validated
 - d. AUTO - Automatically Validated Invoice
- iii. Also a new column will be added to B2B_VENDOR_PARAMETERS table to check whether for this B2B Vendor invoices will be auto-validated or not.
- iv. The new window, My Invoices, will be created for B2B Vendor to check the status and details of their invoices. It will be a copy of the "Purchase Invoice" general window.
- v. This window will feature a button to change the status of the invoice, the process class will be *org.openbravo.b2bintegration.ers.buttons.ValidateInvoiceButton*
- vi. The new ACK Status field will be added to both the Purchase Invoice and My Invoices windows.

Payment Acknowledgement

- vii. Payment Acknowledgement functionality will require to create a new column in the FIN_PAYMENT table to represent the ACK Status of the Payment
- viii. A new List Reference will be added to represent the possible values of this column:
 - a. N/A – Not Applicable
 - b. P – Pending Acknowledgement
 - c. OK – Payment Acknowledged
 - d. AUTO - Automatically Acknowledged Payment
- ix. Also a new column will be added to B2B_VENDOR_PARAMETERS table to check whether for this B2B Vendor payments will be considered auto-acknowledged or not.
- x. The new window, My Payments, will be created for B2B Vendor to check the status and details of their payments. It will be a copy of the "Payment Out" general window.
- xi. This window will feature a button to change the status of the payment, the process class will be *org.openbravo.b2bintegration.ers.buttons.AckPaymentButton*
- xii. The new ACK Status field will be added to both the Payment Out and My Payments windows.

Background Process Update Invoice and Payment B2B Status

- i. A background process will update the status of the invoices and payments of B2B Vendors. This background process will be created in the Application Dictionary, and its

Java class will be *org.openbravo.b2bintegration.ers.UpdateInvoiceAndPaymentB2BStatus*

- ii. A new field will be created in the B2B_VENDOR_PARAMETERS table, to indicate the numbers of days that have to pass before an invoice can be auto-validated.
- iii. The process will first update invoices and payments to P – Pending status if they belong to a B2B Vendor
- iv. Later, it will set to AUTO – Auto Validated / Auto Acknowledged those invoices and payments that have to be automatically modified.

4.2.7. Technical Implementation of Widgets and Reports

Widgets have no special technical implementation, other than creating the appropriate HQL query so that the functionality described in the Functional Design is achieved.

To provide filtering, a new java class implementing FilterExpressions was created:
org.openbravo.b2bintegration.base.filterexpressions.UserBusinessPartnerId

Each widget should be assigned to the proper role.

Chapter 5. Project Definition. Development plan

The objective of this chapter is to account for all the activities needed to be done in order to plan, develop, and publish the B2B Integration Module. In order to obtain a Project Plan, these activities and tasks will have to be valued and given a cost in hours, as well as establishing the hierarchies and logical dependencies between them.

5.1. Tasks Breakdown and Cost Estimation

The cost of developing the B2B Integration Module for Openbravo ERP will be calculated, based upon the functional footprint explained on the Functional Design document, for which the technical implementation approach was described on the Technical Design document.

The Technical Design document allows furthering dividing the work to be done in the smallest but complete units of work, intended to be done by a single consultant or programmer. Each of these small units of work its duration is estimated and given a cost in hours, and the total cost of developing will arise.

The detail of the Technical Design also allows establishing priorities and dependencies in the different tasks and areas.

5.1.1. Effort Diagram

What follows is the complete task breakdown for the development activities, grouped by functional and technical area, and the aggregate sum of the total cost of development.

B2B Integration module extensions		Total Man-Hours:	373
Modularity			8
	Creation of Module Structure	4	
	Registering Modules in the Central Repository	4	
Platform Extensions			128
B2B Integration Configuration		20	
	Create new tables	10	
	Create new windows, tabs and menu items	10	
Users			36
	Create new fields in User / Contact tab	4	
	Grant Access Wizard	24	
	Revoke Access Process	8	
Roles			6
	New fields to identify roles	2	
	Dataset and Dataset query	4	
Data Access			8
	Fields and trigger in AD_USER	4	
	WhereClause	4	
Activity Log			16

Create new table	4	
Create new window, tab and menu item	6	
Create utility java file	6	
HTML Email Engine		40
Create new Template data structure	4	
Create new window for Template	4	
Create new tables for Email Instance	8	
Create new window for Email Instance	4	
Email Instance helper class	4	
B2B Email Sender process	4	
B2B Email Monitor process	12	
CKEditor for Openbravo		2
Include module in pack	2	
Master Data Management Extensions		50
Company Details		6
My Company Details new window	6	
Products		26
New columns in M_PRODUCT	4	
Modifications to Product window	2	
My Products new window	4	
New columns in B2B Configuration	8	
Add Product process	8	
Pricelists and Pricelists Alerts		18
New field in Pricelist Version	2	
My Pricelist Versions new window	6	
Copy Pricelist process	6	
Expired Pricelist alert	4	
Purchase Order Management		75
Purchase Order Status and new windows and fields		14
PO Status field	2	
New Date and text fields in C_ORDER	2	
My Purchase Orders new window	6	
Modifications to Purchase Order window	4	
Publishing, Acknowledging and Accepting POs		26
B2B Order Doctype fields and trigger	4	
New fields in B2B Vendor Configuration	2	
Publish Order button and logic	6	
ACK Order button and logic	3	
Accept Order button and logic	3	
Reject Order button and logic	3	
Background expiry date process	5	
PO Negotiation		9

Request changes button and logic	3	
SCM Accept changes button and logic	3	
SCM Reply to changes button and logic	3	
Advanced Shipment Notices		20
B2B Shipment Status field	2	
New Date and check fields in M_INOUT	2	
My Shipments new window	6	
Modifications to Goods Receipt window	4	
Trigger logic	6	
Create ASN from PO		6
Create ASN button and logic	6	
Vendor Managed Inventory		26
Storage Bin and Products visibility		8
Foreign Key to M_LOCATOR	2	
Vendor Managed Storage Bin new window	6	
Minimum Stock and Stock Alerts		6
New fields in M_PRODUCT	2	
New Alerts	4	
New Advanced Shipment Notice		12
Create ASN button and Logic	6	
Add products under stock option	6	
Evaluated Receipt Settlement		38
Invoice Validation		16
Invoice Validation Status field	2	
New fields in C_INVOICE	2	
New fields in B2B_VENDOR_PARAMS	2	
My Invoices new window	6	
Modifications to Purchase Invoice window	4	
Payment Acknowledgement		16
Payment Acknowledgement Status field	2	
New fields in FIN_PAYMENT	2	
New fields in B2B_VENDOR_PARAMS	2	
My Payments new window	6	
Modifications to Payments Out window	4	
Background Process		6
Create Background Process	6	
Widgets and Reports		48
FilterExpression		4
Create FilterExpression	4	
Widgets		44
Products Summary Widget	8	
My Purchase Orders	6	

My B2B Orders	6		
My Shipments new window	6		
My Invoices	6		
My Open Invoices	6		
My Payments	6		

Table vi - Development Effort Diagram

So the total cost of development rises up to 373 Man-Hours.

5.1.2. Overhead Activities

In every software development project there are side activities that will also have some cost and that will take time to complete. While these are not considered in the total cost of developing, they will be considered later in the project planning.

These activities have to be done alongside the development of the modules, they are support or side activities needed. Furthermore, the duration of these activities depends on the total duration of the software development.

The cost has been estimated using the above basis of 373 Man-Hours of software development.

Activity	Cost
Writing Project Documentation (Functional Documentation, Technical Design, Test Battery)	20
Overhead costs of Software Revision Control and security back-ups.	6
Writing User Manuals, Tutorials, Installation and Configuration Manuals, and Troubleshooting Guide	30
Integration Testing	16
Publishing and Support Plan	20
Total Man-Hours:	92

Table vii - Overhead Activities diagram

This gives a total of 92 Man-Hours in overhead activities. The total project cost will add up to $373+92 = 465$ Man – Hours.

5.1.3. Tasks Dependencies

For the B2B Integration module project, task dependencies follow closely the module architecture diagram of the Technical Design. When adding the Overhead Activities, the tasks dependencies results as follows:

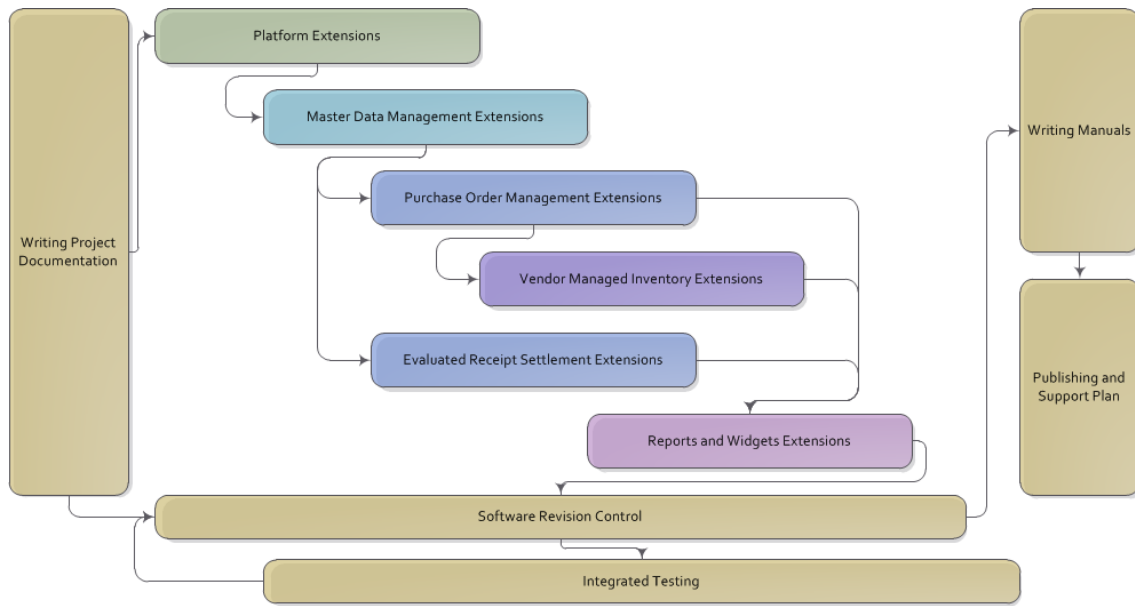


Figure LIV - Task dependencies diagram

5.2. Project Plan

Once the cost in Man-Hours has been calculated, resources and scope restrictions are to be taken into consideration to develop the project plan.

5.2.1. Available Resources

Initially, it will be assumed that only one actor, with both the skills of a consultant and a programmer is available. It will also be considered that it has infinite availability, this is, this programmer may work indefinitely in this project.

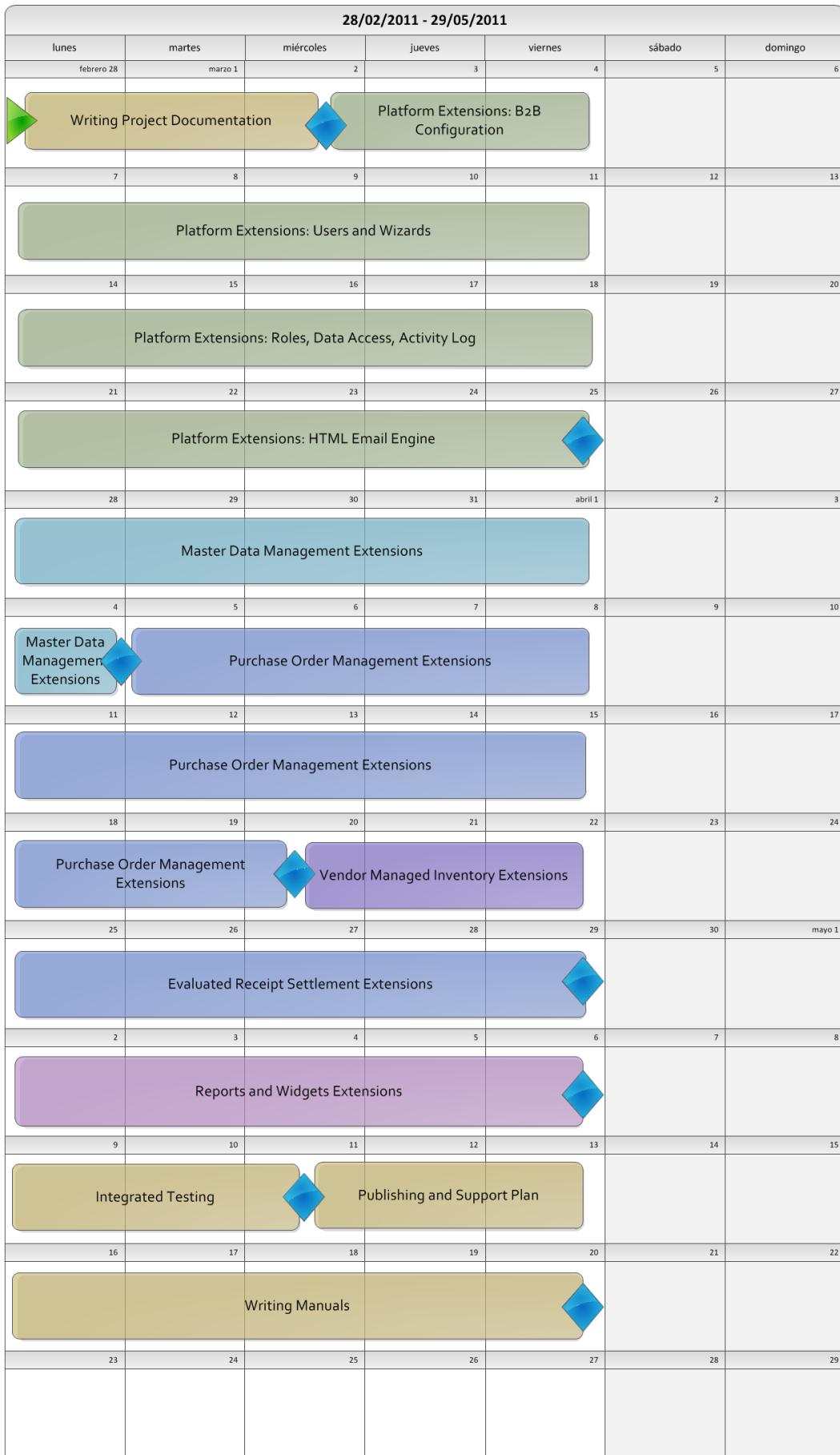
Following a traditional work schedule, 8 hours a day, 5 days a week, and taking into account some variability to process the equivalent of 25 vacations days per year, it is assumed that this programmer – consultant has between 176 and 182 hours available per month.

5.2.2. Project Scope

Since the total cost of developing adds up to 465 Man – Hours, it is considered feasible to be done with the available resources. The mean calculated time to complete it will be 2.6 months of time, or about 12 weeks.

5.2.3. Project Scheduling

Fitting the project schedule in 12 weeks, taking into account dependencies and the cost of delivering every different tasks, creates the following Project Plan diagram:



Figure

LV - Project Development Plan

Chapter 6. Publishing and Support Plan. Future Developments

This chapter describes how the official mechanisms for publishing Openbravo ERP modules work, the Openbravo Forge and Central Repository. In addition, the B2B Integration project page will be presented and its available features will be described. Last, future support actions and possible extensions will be discussed.

6.1. Openbravo Forge and Central Repository

Openbravo Forge is the central point of communication between members of the Openbravo Community. The Openbravo Community is composed by official Openbravo ERP developers, Openbravo Partners, independent consultants, coders and developers, users and ERP enthusiasts.



Figure LVI - Openbravo Forge logo

It serves mainly two functions. First, it is used to coordinate the community, provide news and announcements, community support through forums, and documentation through the official Openbravo Wiki for both Openbravo ERP and Openbravo POS products.

Secondly it can be used to start and publish projects around Openbravo ERP, by providing some common tools and functionality, and it is the only access point to publishing new modules in the Central Repository.

Openbravo Forge can be accessed at <http://forge.openbravo.com/>

The Central Repository is the information system that is accessed by all Openbravo ERP installed instances to check for existing modules, download them, and update already installed modules. It is integrated with the forge so that projects can be associated to modules.

6.1.1. Community communication and collaboration

Openbravo Wiki is linked to Openbravo Forge, this site provides supporting documentation for the Openbravo Community, including people installing, configuring, using, developing or localizing Openbravo solutions.



Figure LVII - Openbravo Wiki logo

It also provides General Documentation on many subjects:

- General Information
 - Information on Getting Started with Openbravo, which provides a summary of resources
 - Features being considered for development in the near future can be seen in the Roadmap.
 - The current Openbravo 3 Development Status and Release Notes are available.

- Support Policy
- System Administrator's Guide
 - Information on System Requirements
 - Upgrade Guide
 - Openbravo Appliance Administration Guide
- Configuration Guide
 - The Quick Guide describes the main configuration steps needed for Openbravo to be able to run fundamental functional flows for your company.
 - The Configuration Manual contains more detail on configuring Openbravo, in particular, the configuration of specific business flows.
- User Documentation
 - The User Manual explains how to access Openbravo and navigate through the user interface.
 - The Business Flow Descriptions contain step by step instructions on how to execute key Openbravo functional flows.
 - The Screen Reference chapter contains the meta-data description of all the screens, fields and other Openbravo entities.
- Developer's Guide
 - The How-to sections describe Openbravo development topics using a focused development goal. There are how-tos on modularity, extending the datamodel, adding windows and webservice, the data access layer and more.
 - The Concepts section gives a detailed description of all relevant Openbravo ERP development concepts. The content ranges from common development topics (such as the project structure, build tasks) to modularity and the application dictionary. Each of the main layers of the application is discussed: database, middle-tier and web-tier with webservice.
 - The Reference section consists of a detailed description of the data model from different points of view: the database model, the entity model, the hibernate mapping and the REST XML Schema. In addition javadoc and other reference-like topics are discussed in this section.
 - The Examples section contains examples of existing code in the Openbravo ERP application. The purpose of this chapter is to give directions for the reader to study current Openbravo code and use that as the basis for own custom code
 - Finally the Tips and Tricks section contains troubleshooting tips. The tips and tricks are based on experience and user questions and solutions in the forums.
- Localization Guide
 - Localization Process
 - Translating Openbravo
 - Accounting and Tax Reports
 - Configuring Payment Methods

Openbravo Wiki can be accessed at <http://wiki.openbravo.com>

Every Open Source Software should make code publicly available and its development status always accessible. Openbravo Code is the website linked to Openbravo Forge to provide this feature.

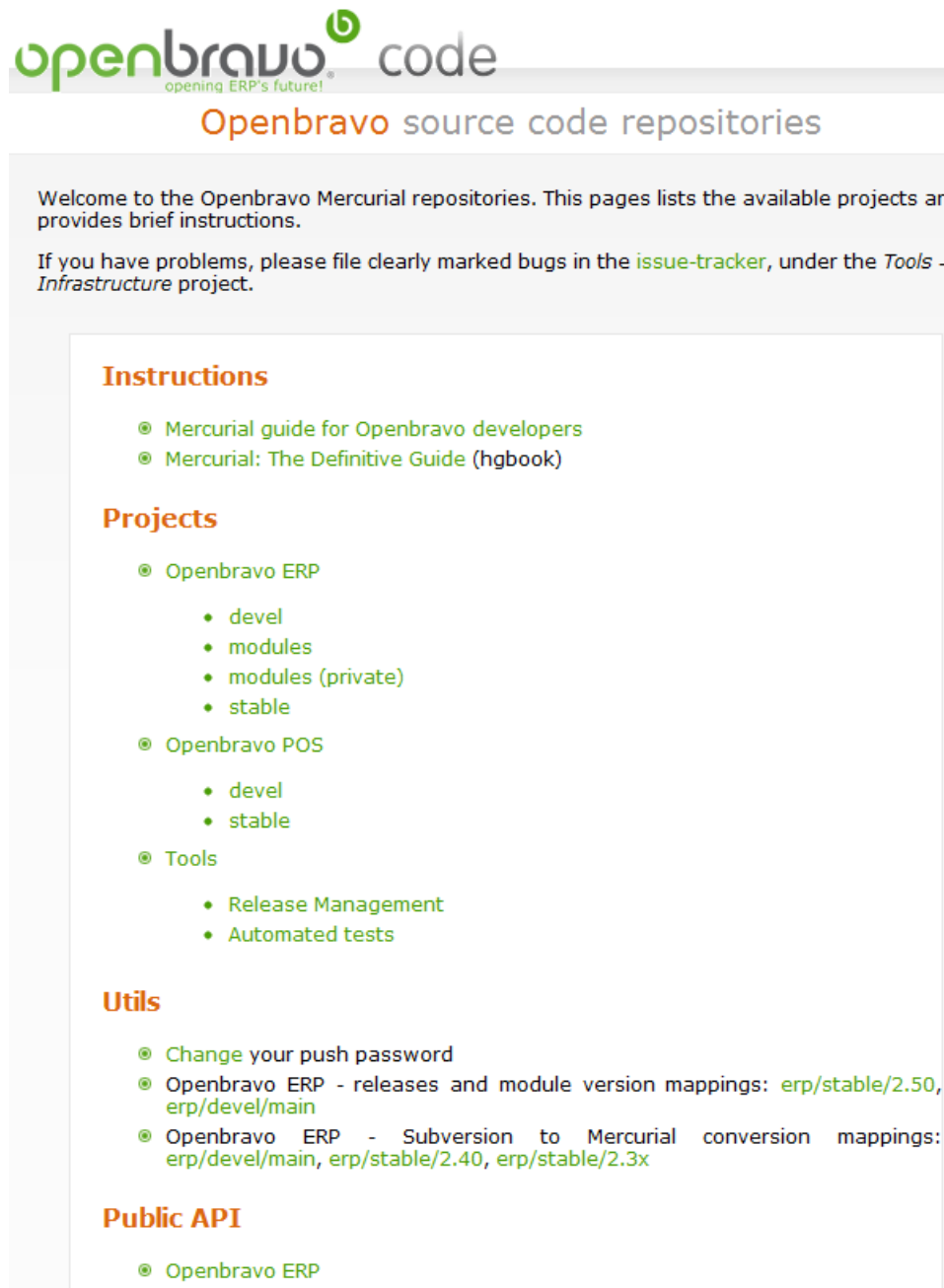


Figure LVIII - Openbravo Code repositories

Openbravo Code Repositories can be accessed at <http://code.openbravo.com>

6.1.2. Functionality for end users and module developers

Openbravo Forge is a collaboration platform where third parties can independently develop projects which are synergistic with Openbravo software, therefore increasing the number of solutions available to the Openbravo ecosystem.

The Openbravo Forge objectives are to:

- Provide an infrastructure for community members to develop verticals, plug-ins and localizations
- Provide an infrastructure for partners and consultants to privately develop verticals, plug-ins and localizations
- Provide a way to recognize peoples' contributions and make them visible
- Host the central repository where all the modules are stored
- Allow people to place their own ads linking to web-stores to sell licenses and/or support (people can place a little ad and link in their projects)
- Provide a directory of available projects for Openbravo ERP & POS
- Provide a member directory when people can look for other community members and see their activity

It is important to notice that all community members can use this infrastructure for free for any open source product.

Openbravo Forge benefits

For all Openbravo community:

- Registering a project in the Forge is a pre-requisite to publish it in the Central Repository. It is a great tool to increase dissemination of both free and commercial products.
- People can place a link to their own web-store to sell licenses or support for registered projects. It allows community members to increase their visibility and recognition.

To decouple the Openbravo Forge development and the Modularity project services are designed to be operated from Openbravo ERP or from the Forge. The first group of services are architected as web services provided by the Central Repository –so the Forge- and consumed by Openbravo ERP components that are responsible to provide a user interface for them. The second groups are services provided directly by the Forge without any dependency on Openbravo ERP.

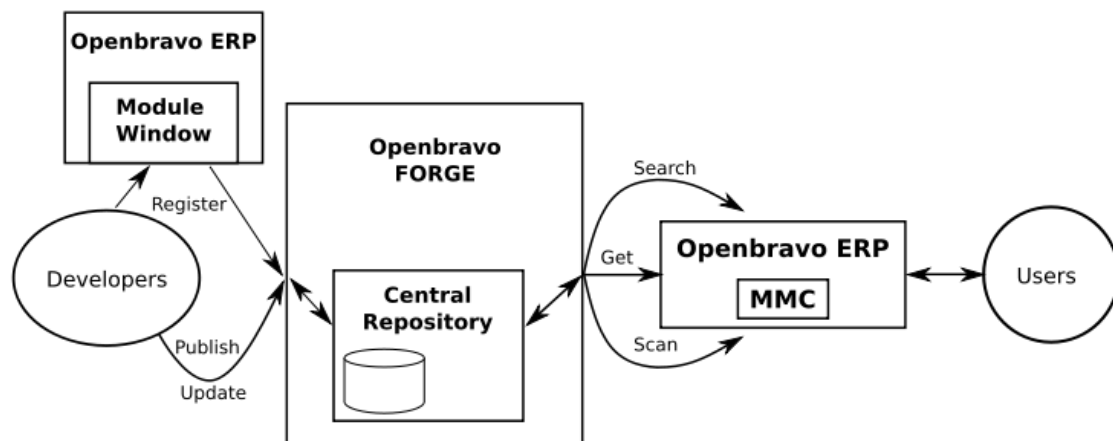


Figure LIX - Diagram of available services in Central Repository

Services for developers:

- **Register a module:** a developer registers a new module in the Central Repository to get global unique identifiers to be used for development (it's required to register a module before starting the module's development).
- **Publish a new version of a module:** a module can have different versions in its life cycle. For each of them module owners need to update the information in the Central Repository to make this new version available for users.
- **Update the information of a version of a module:** module owners are allowed to modify some information in a version of a module (eg. Extend the compatible versions of a module it depends on).

Services for users:

- **Search for modules:** from the MMC users can query the Central Repository to get a list of modules that match with user request.
- **Detailed information of a module:** from the MMC users can query the Central Repository to get all the information available from a module.
- **Scan for updates of a list of module versions:** from the MMC users can request to the Central Repository if there are available updates (newer versions) for all the modules installed in their instances (or just for one of them).
- **Get the code of a module version (to be installed) :** for both installing new modules and updating current ones it is needed to get the compressed file (.xob file) that stores module code.
- **Check consistency:** from a list of modules version, the Central Repository validates that all of them can be installed together, validating all dependencies.

Openbravo Forge is a free tool for people registering projects under open source licenses. Additionally, Openbravo Forge provides additional features for Openbravo Partners:

- Openbravo Forge is a professional tool. Partners have the exclusive right to register private projects. This is designed to give them all the tools (code repository, bug tracker, documentation repository, ...) to manage client customizations
- Projects sold in collaboration with Openbravo are encouraged to be performed in the Forge to maximize the contribution

Overview of main services

When a project is registered, Openbravo Forge offers the following services:

- **Forums.** Allow coordination between different project members and are an excellent tool to give support to your users.
- **News.** News is the place to inform project members and users about the recent milestones of a project, like for example releasing a new version of the software.
- **Download.** Store files and releases for easy downloading
- **Bug Tracking.** A bug tracker system that helps to keep track of your bugs and development tasks.
- **Code.** Subversion or Mercurial source control system to store the source code of the project's application.

- **Wiki.** Wikis are great systems to store documentation and coordinate development efforts.
- **Module.** Module allows user to publish modules Openbravo ERP Central Repository

Projects directory

Openbravo project directory is the directory of Openbravo ERP & POS projects of Openbravo ecosystem.

Openbravo directory can host:

- Projects registered within the Forge
- Projects hosted fully or partially in external systems
- Projects can be classified in a browsable category taxonomy. Every registered project can belong to up 3 categories

Users can:

- Search the project directory
- Browse using the category taxonomy
- See latest additions following the Forge Latest Activity
- Sort it by different criteria (creation date, alphabetically, etc)

6.2. B2B Integration Project

B2B Integration module is published in Openbravo Forge as a project. It can be accessed at <http://forge.openbravo.com/projects/b2bintegration>, and this will be the site which will centralize activities around the B2B Integration module and project described in this thesis

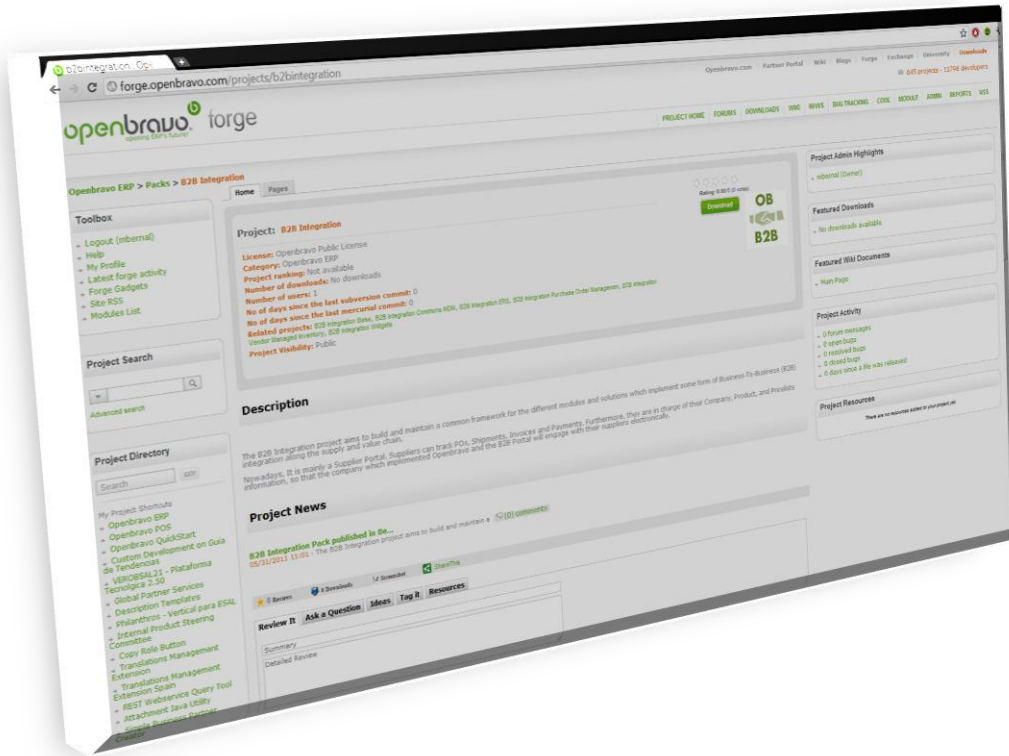


Figure LX - B2B Integration Project page

The source code and module have been published under the Openbravo Public License 1.1, which is a form of Open Source License. The main motivation for this is that the project was never intended to create profit, and that it can be used as a great example of what can be achieved in Openbravo ERP freely.

6.2.1. Project Name, Project Logo and Description.

Project Name will be “B2B Integration”. The module description will be:

“The B2B Integration project aims to build and maintain a common framework for the different modules and solutions which implement some form of Business-To-Business (B2B) integration along the supply and value chain.

Nowadays, It is mainly a Supplier Portal. Suppliers can track POs, Shipments, Invoices and Payments. Furthermore, they are in charge of their Company, Product, and Pricelists information, so that the company which implemented Openbravo and the B2B Portal will engage with their suppliers electronically”.

Screen captures of the main platform, master data management, purchase order management, vendor managed inventory and evaluated receipt settlement functionality will be displayed.

A Logo has been created for the B2B Integration Module:



Figure LXI - B2B Integration for Openbravo ERP logo

6.2.2. News Section

The news section will be active. It will feature an announcement whenever a new version of the module is published. The first announcement was:

“The B2B Integration project aims to build and maintain a common framework for the different modules and solutions which implement some form of Business-To-Business (B2B) integration along the supply and value chain. It is open to users, testers and collaborators, published under the OBPL license. <http://bit.ly/l2tVV7>”

Nowadays, It is mainly a Supplier Portal. Suppliers can track POs, Shipments, Invoices and Payments. Furthermore, they are in charge of their Company, Product, and Pricelists information, so that the company which implemented Openbravo and the B2B Portal will engage with their suppliers electronically.

It is composed by the following interdependent sub-modules, which shouldn't be downloaded standalone:

- **B2B Integration Commons** - This module provides the common framework and software artifacts for the modules in the B2B Integration Pack, such as HTML emailing engine, B2B Vendor and users configuration, roles, Activity Log, etc.*
- **B2B Integration Commons Masterdata** - This module provides the software artifacts and required sources and processes to provide easy Master Data Management within the B2B Integration Pack. Mainly, Business Partner, Product and Pricelist management.*
- **B2B Integration PO Management** - Includes the basic framework for B2B Purchase Order Management and Negotiation. Purchase Orders can be published to the B2B Vendor Portal, and your Vendors will receive notifications via email and alerts. You may track how your vendors acknowledge, accept, reject or negotiate POs, and create Advanced Shipment Notices based on an agreed PO.*
- **B2B Integration - Vendor Managed Inventory** - This module allows for Vendors to control their Products in shared or vendor managed warehouses. Vendors will receive notifications when a product is close or beyond their minimum stock, and they can automatically create shipments from a list of under-supplied products. A shipment can now be created by Vendors so that it is tracked from the moments the goods are dispatched (Advanced Shipment Notice), and the Supply Chain Manager can notify the Vendor when the shipment arrives.*

- **B2B Integration - Evaluated Receipt Settlement** - This module includes the views and uses so that vendors can validate their invoices and acknowledge their payments, with no contradicting information stored on emails or physical documents.

- **B2B Integration - Common Widgets** - This module implements widgets to check and administer Business Partner's orders, shipments, invoices, open invoices and payments.

Tutorials, Screenshots, Diagrams and Configuration Manuals will be published in a few days.

Disclaimer: This project is not endorsed or related to Openbravo S.L.U. You can contact me <jmarcos.bernal@gmail.com> if you are further interested in this project. Support will only be available on a free will basis and within the forums of the Forge."

6.2.3 Code Repository

Code for all modules will be available as a mercurial repository under the project umbrella.

6.2.4. Wiki

The Wiki section of the project will feature the following documents:

- User Manual and Tutorials
- Configuration Manual
- Functional Documentation
- Technical Documentation

6.2.5. Forums and Support

Forums will be activated for support.

6.2.6. Issue Tracker

The Issue Tracker will be activated for support. Feature Requests and Bugs will be tracked using this tool.

6.2.7. Available Downloads

All the **.obx** files generated by the distinct sub-modules will be available for download. The "Module" Section will be associated and display the download of the main module pack: "*org.openbravo.b2bintegration.pack*".

6.3. Future Plans and Extensions

This section explains the future immediate plans after the publishing of the code and creation of the B2B Integration forge along with the documentation in Openbravo Forge, and analyzes the future possible functional extensions for the B2B Integration Module.

6.3.1. Supporting the core B2B Module

Support is going to be given to this module, through the usual channels that the project in Openbravo Forge provides. Forums will be used to answer user and consultants' issues, or

inquires. Through this channel, too, official communication with the module's author and collaborators will be possible.

Bugs will be corrected and new versions of the module will be released, perhaps adding incremental functionality.

It will be desirable to give the project more visibility and projection, so that possible users and implementers, developers willing to collaborate, or even partners interested in commercial opportunities around the B2B Integration framework for Openbravo ERP.

- Some promotional materials are going to be created, including a Product Brief, a small 3-page long summary of the functionality and advantages provided by the B2B Integration module.
- Challenges and Problems solved by the Vendor Portal scorecard.
- Screenshots of each new menu, new Window and Tab, and modified
- Diagrams and Functionality Summaries
- For developers, a small Introduction document, "B2B Integration Module for Developers", including an appeal for collaboration.
- This Thesis document will be available for download.

To make it even easier for users, consultants, testers and developers, it will be possible to develop a set of Demo Data, to be installed to an existing Openbravo ERP instance, with fictional company data, and fictional B2B Partners to demonstrate the flows and as training material.

This Demo Data could be added to a Demo Server to allow users visiting the B2B Integration Project's website to test directly the functionality.

6.3.2. Extending the B2B Integration module

Extending the B2B Integration framework through modules is possible. Not only that the platform requirements are now in place to create useful B2B extensions, but the existing functionality around Purchase Order Management, Vendor Managed Inventory and Evaluated Receipt Settlement can be further perfected. In addition, many of the proposed functionality didn't make it to fit in the scope of this first version.

Below it is a description of possible, sound extensions that have been imagined by the author:

B2B Integration Spanish Translation module

Openbravo ERP is a company based in Spain and the Spanish Translation Pack, along with the Spanish Professional Localization Pack, which include functionality to fill in official Spanish Tax Reports and Spanish Chart of Accounts are among the most downloaded modules of the Central Repository.

An extension module translating the whole B2B Integration Pack module could be developed to tackle into this user base and to demonstrate the feasibility of a translation for the work done, exposing possible bugs and vulnerabilities.

Enhanced Purchase Order Management Capabilities

The Purchase Order Management module is one of the main components of the B2B Integration Pack. Its functionality is comprehensive but direct confrontation to real-world scenarios may expose important functional gaps.

The author considers that this module may be a perfect target for the first extensions, both in functionality and usability.

Examples include an enhanced Purchase Order and Quotations versioning feature, enhanced deadline and vendor awareness capabilities, PDF version of the orders attached to the emails, and the ability to accept or reject orders from an embedded link in the email. Also, the ability for B2B Vendors to modify directly the published purchase order and to keep track of this until an agreement is made.

Request for Quotations and Inverse Bids

Request for Quotations is a method by which an organization openly publishes a list of required products and services, or specifications for a project, in a paper or document known as RFQ. Different interested contractors may compete or send different quotations to address the published RFQ with different degrees of completion.

The B2B Integration Platform, with its PO Management functionality, could be leveraged to support this.

The second part of this RFQ, which can also happen independently, is an inverse bid, by which the organization publishes a definitive RFQ, and vendors compete between them for the contract by trying to provide the lowest price possible. When different bids are made, B2B Vendors could be notified.

Delivery Schedule

A simple to do but absent functionality would be a report to show both B2B Vendors and Supply Chain Managers the delivery schedule for a given time frame.

Supply Forecast, Depletion Forecast

Advanced techniques for stock depletion, supply forecast, and demand forecast exist as complicated mathematical and logical functions. Once the B2B Integration is in place and sufficient data is available, these prediction algorithms could be applied and its results shown for both B2B Vendors and Supply Chain Managers.

Extended Reporting Capabilities

Reporting as it is right now is only informative and quite limited. Extended reporting included also qualitative and aggregated data is a very good target for extension modules.

Supply Chain Manager Control Center. Scorecard Matrix

A centralized widget to show real-time data summaries of stock, supply and warehouse statuses and disruptions for the whole supply chain.

Online Tutorials

When B2B Vendors first log-in the Vendor Portal they may be a bit lost. Tutorials and How-To's could be directly embedded into their workspace.

Access levels

Only two default roles have been created, Supply Chain Manager and B2B Vendor Contact. A new module including datasets for new roles and permissions so that some users, either internal or external to the organization, may be able to access some of the B2B Functional Areas while not others.

Attachments and Print Form emails

Emails are currently not attaching the Print Form of Shipments, Invoices and Purchase Orders, as the standard ERP flows do. There is room for enhancement here.

Configurable Email Notifications

Currently, all email notifications will arrive to the same user, and there is no way to configure which ones should be sent and which shouldn't. Email Notifications could be enhanced to allow configuration at the user level.

Vendor view of activity log

Activity Log is designed for Supply Chain Managers to track the activities being done by all B2B Vendors. It wouldn't be very difficult to create a view for B2B Vendors to see only their activities based on the existing platform.

Per line

Request for changes and modifications are done at the header level. It could also be possible to request line modifications or create ASNs only for some lines.

B2B for Customers

At the start of the Functional Design, it was stated that the Vendor side of B2B Integration was far more important than the Customer side of it. However, now that the B2B Integration Platform is in place, the B2B Customer Portal counterpart could be designed and developed.

Specification Changes

The B2B Vendor Portal could be leveraged to allow specification changes of closed Purchase Orders, and to keep track of them.

Contract Management

Contract management and information about varying payment and delivery conditions is done outside the ERP. Now, the B2B Vendor Portal could be leveraged to include this functionality negotiate, and keep track of past contract modifications.

Supply Chain Event Management

Events happening all around the supply chain could be captured and shown to the Supply Chain Manager. A complex study and design of the solution should be done

Chapter 7. Conclusions

After all the work left behind, some conclusions can be drawn.

In this work, the design and development of a B2B Integration module for Openbravo ERP has been explained.

It has been studied the environment on which B2B Integration for Openbravo ERP would occur involved the study of three different disciplines, in the fields of Information Systems Management (ERP), Supply Chain Management, and Software Engineering. From these three parent disciplines, the methodology and theory needed to develop this module was presented.

A full study of the Core Business flows of Openbravo ERP was necessary to extract the requirements and specifications, which happened in the form of a Functional Gap. From this previous study, priorities were established and a more narrow scope to the project was presented in the Functional and Technical Design documents.

The first of them was used to construct and explain the behavior of the new system, and out of this ones, the second one presented a clear structure and picture of which software artifacts were necessary.

Given the magnitude of this work, it was needed to value and give some time duration for each of the activities, and a project plan was made.

The development took some 12 weeks to be completed, and once it was finished, the B2B Integration module was published in the Internet, and a platform to support and maintain it was also put in place, as a project in Openbravo Forge.

Now this module has been released and it is ready for anyone to use and download, it is expected that it will attract some limited attention and this B2B Integration project will be alive and healthy.

This B2B Integration module may be the first step for extending the Openbravo ERP platform to an ERP II system. The business flows are always changing but the functional gap is lower now.

The project was made by applying tested and proven software development methodology: requirements, design, implementation and support phases. It is however of great satisfaction to prove that it has worked once again to be able to cope with the size of this undertaking.

As a project, it is not ending with this thesis, as future enhancements and hopefully, a good number of implementations can be made. It may attract some commercial opportunities and the abilities and knowledge learnt will surely be useful in the future.

And finally, on a personal note from the author, a great feeling of accomplishment after finalizing this work is reached, which could never had been possible with the help and support of so many people.

Glossary



INGENIERO DE TELECOMUNICACIÓN

PROYECTO DE FINAL DE CARRERA

Autor: J. Marcos Bernal Alejo

Tutor: Carlos del Río Bocio

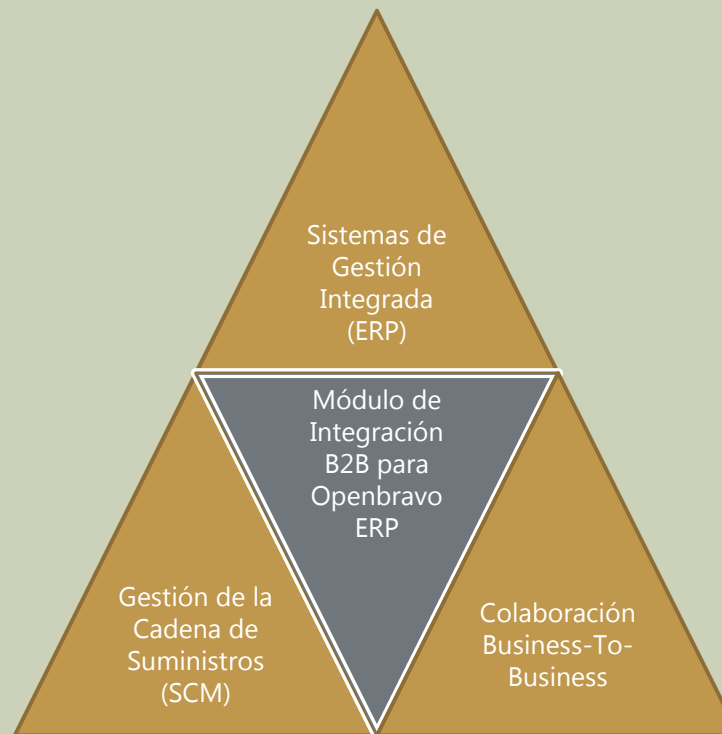
30 de Junio de 2011

DESIGN AND
DEVELOPMENT OF
A BUSINESS-TO-
BUSINESS
INTEGRATION
MODULE FOR
OPENBRAVO ERP

- J.Marcos Bernal Alejo
 - Ing. Telecomunicación (2002-)
 - 3 años de experiencia como consultor funcional y técnico en Openbravo ERP
 - Erasmus en Tampere, Finlandia (2006-2007)
 - Business and Industrial Management
 - Software Engineering
 - Enterprise Information Systems

- PFC

- Estudio del estado del arte en ERPs, SCM, y B2B
- Aplicación de metodología de desarrollo de software
- Diseño, desarrollo y publicación de un módulo B2B para Openbravo ERP



¿Quién soy yo?

¿De qué trata este PFC?

Tendencias ERP

Tendencias SCM

Tendencias B2B

Open Source y Openbravo ERP

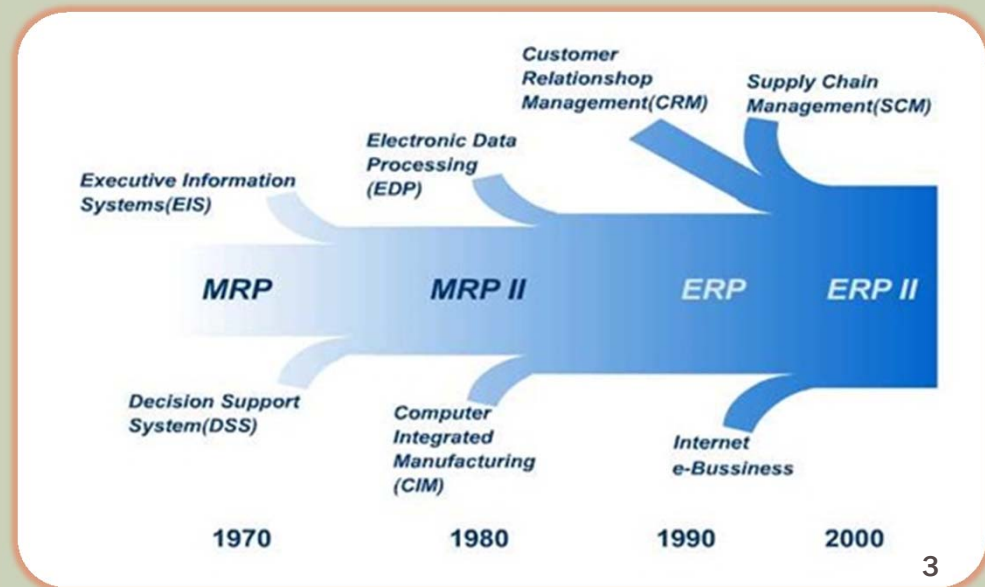
Tendencias en Sistemas de Gestión Integrada (ERPs)

- Funcionalidad

- Especialización por industria
- ERPs para PYMES
- Optimización de Cadena de Suministros vs Procesos Internos
- Tiempo Real.
- Transparencia

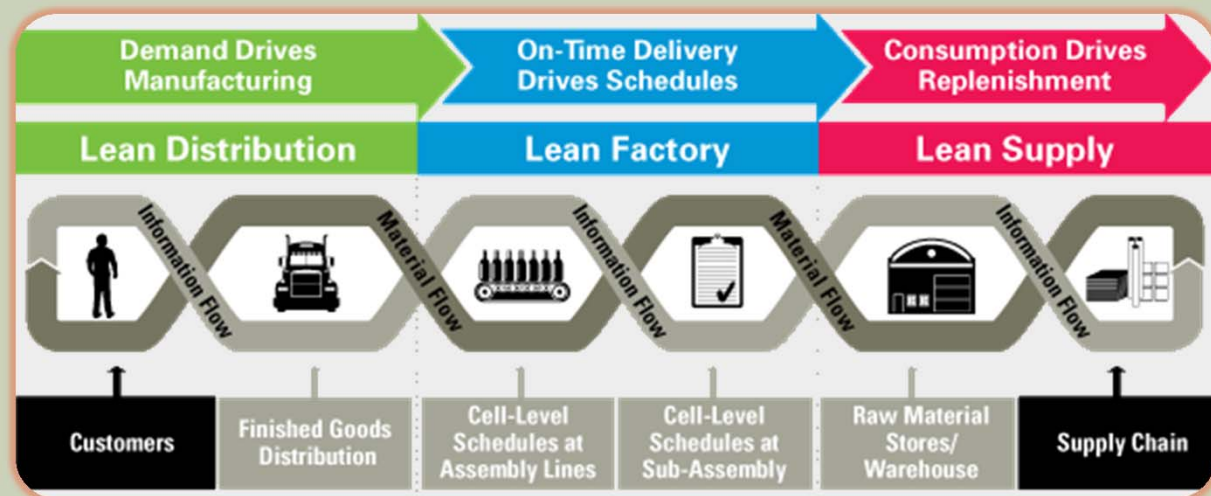
- Tecnología

- Basados en Web
- Diseño Modular
- Interoperables e Integrables



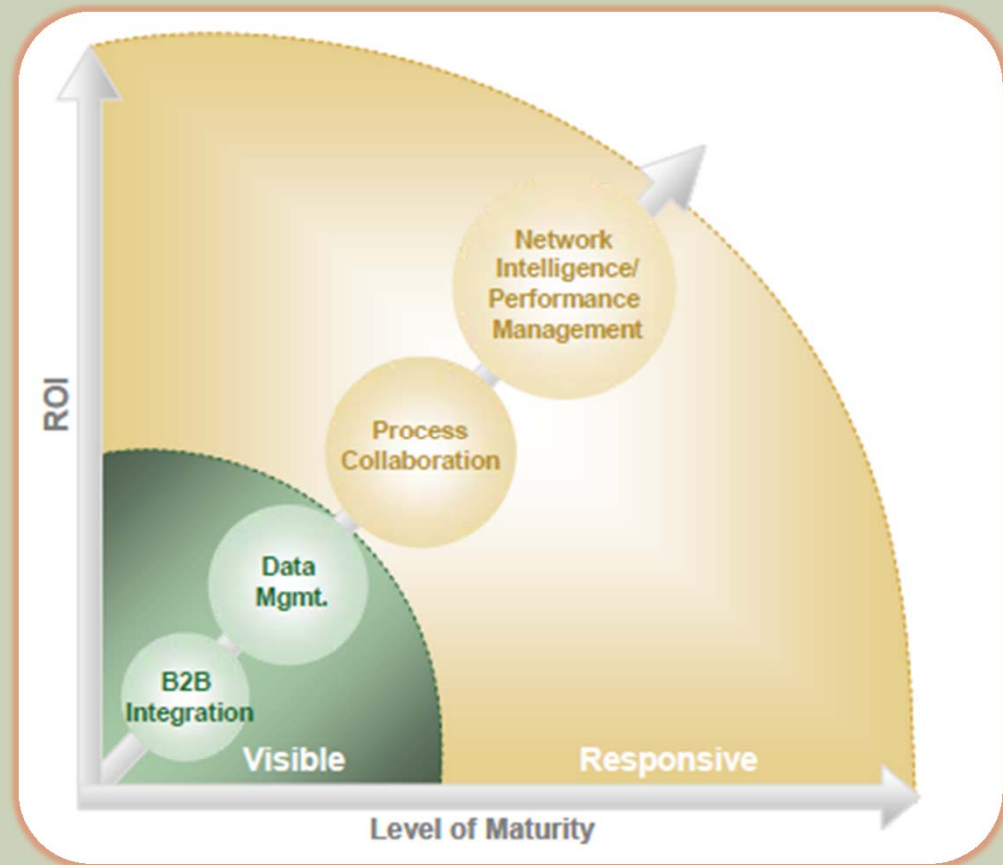
Tendencias en Gestión de la Cadena de Suministros

- Gestión de la red dispersa de distribución
- Gestión integrada del flujo de materiales, dinero e información
- Economía Ecológica
- Filosofía Kaizen Japonesa
 - Just-In-Time
 - Kanban
 - Lean Management



Tendencias en Colaboración Business-To-Business

- Evolución del modelo de empresa vertical
- Especialización de proveedores y externalización
- Fuentes externas de datos en ERPs
- Fracaso de las iniciativas EDI (*Electronic Data Interchange*)

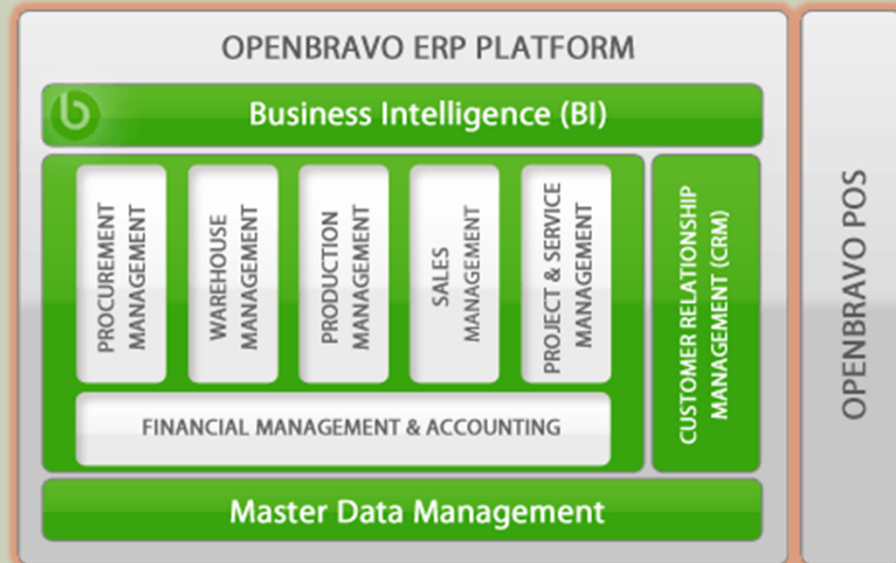


Software de Código abierto. Openbravo ERP

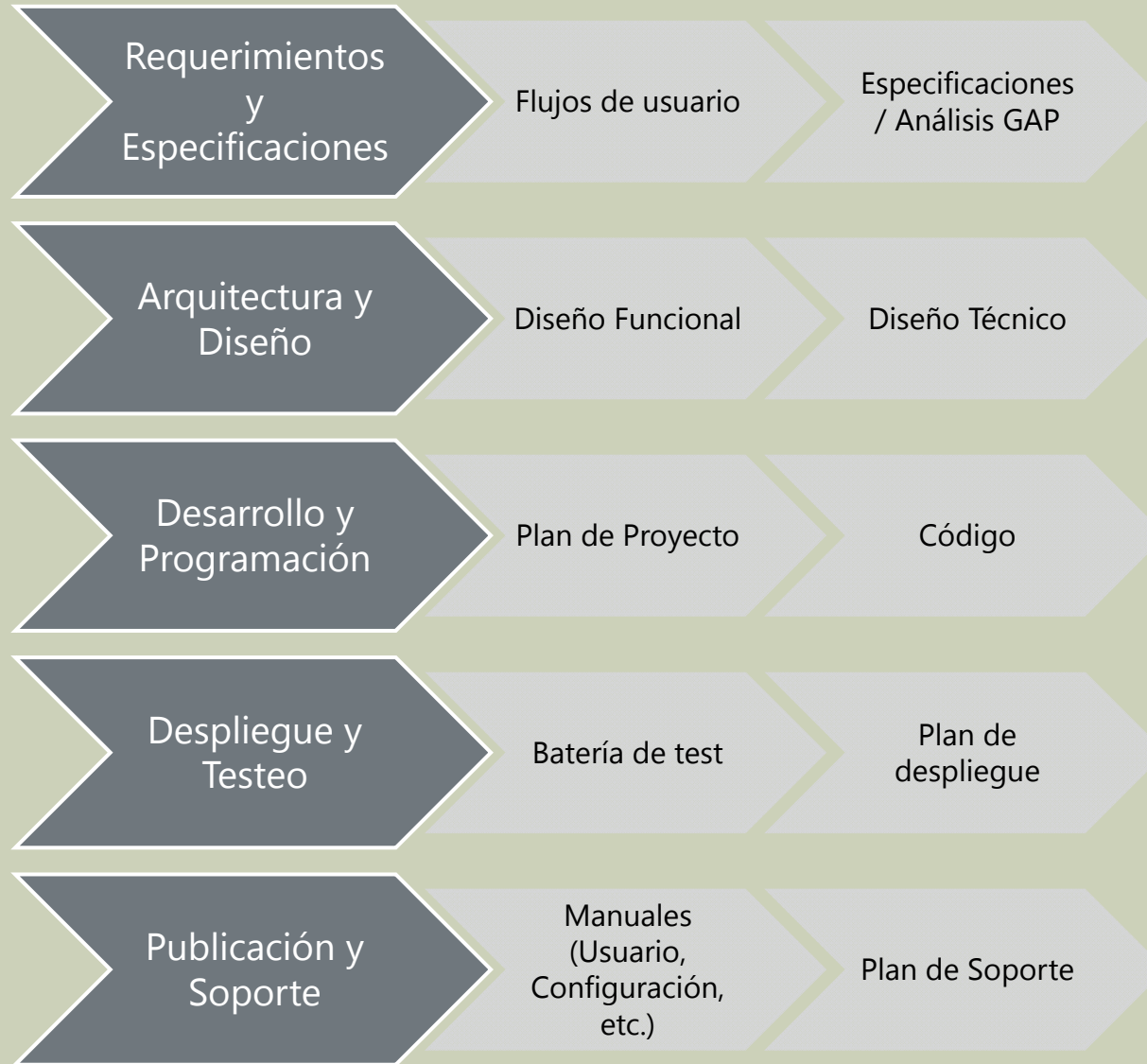
Open Source



Openbravo ERP



Metodología de desarrollo de software



Proyecto de desarrollo de Software

Análisis de flujos de negocio y GAP Funcional

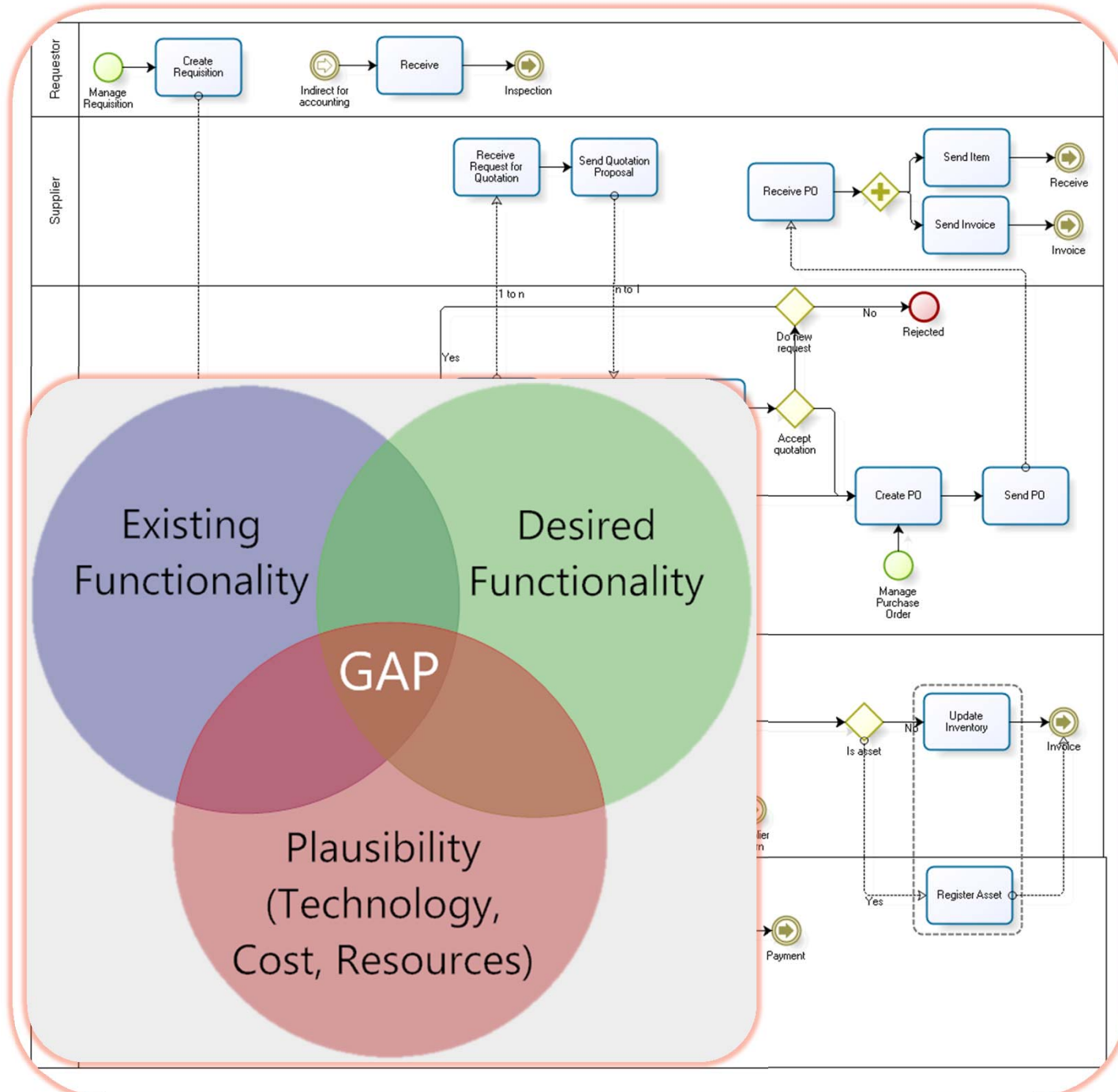
Diseño Funcional y Técnico

Plan de Proyecto

Desarrollo

Publicación

Planes Futuros



Análisis de flujos de negocio y GAP Funcional

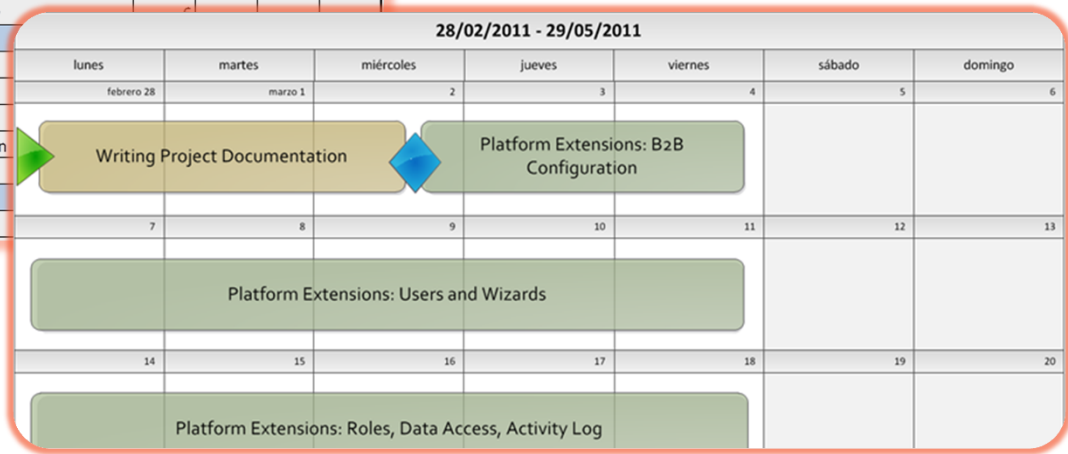
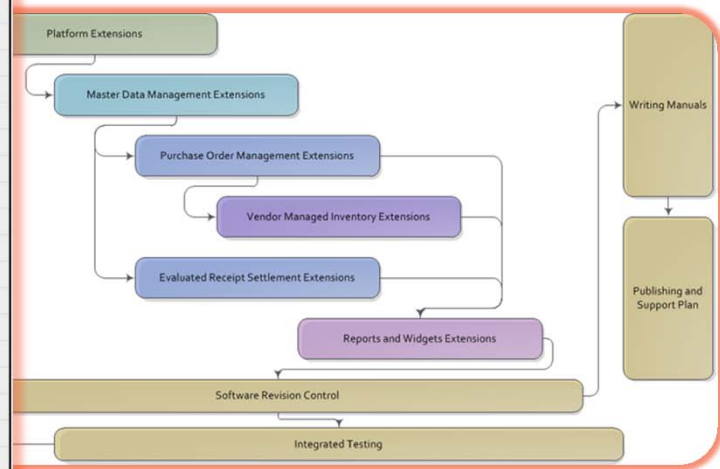
Diagramas de procesos BPMN

Gap Tecnológico

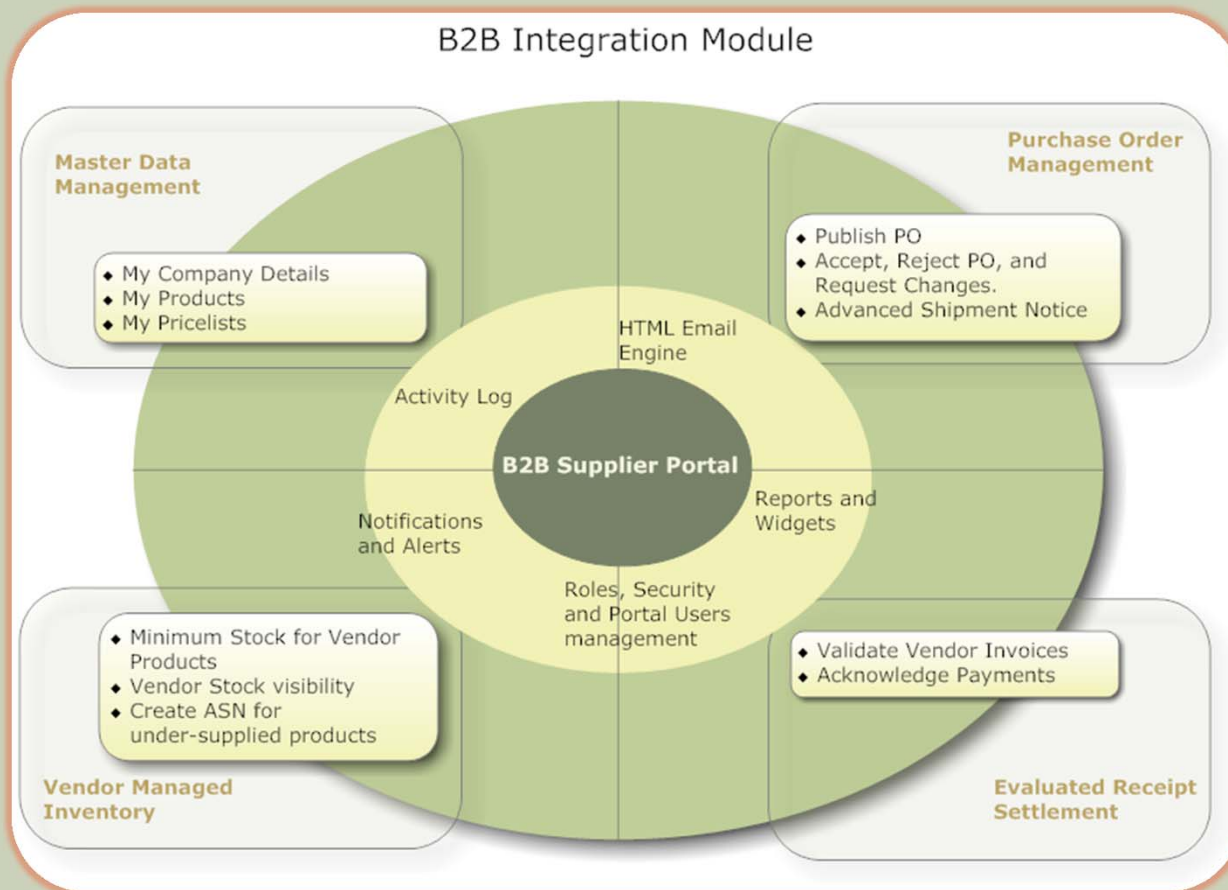
Gap Funcional

B2B Integration module extensions		Total Man-Hours:	373
Modularity		8	
Creation of Module Structure			
Registering Modules in the Central Repository			
Platform Extensions			
B2B Integration Configuration			
Create new tables			
Create new windows, tabs and menu items			
Users			
Create new fields in User / Contact tab			
Grant Access Wizard			
Revoke Access Process			
Roles			
New fields to identify roles		2	
Dataset and Dataset query		4	
Data Access			
Fields and trigger in AD_USER		4	
WhereClause		4	
Activity Log			
Create new table		4	
Create new window, tab and menu item		6	
Create utility java file		6	
HTML Email Engine			
Create new Template data structure		4	
Create new window for Template		4	
Create new tables for Email Instance		8	
Create new window for Email Instance		4	
Email Instance helper class		4	
B2B Email Sender process		4	
B2B Email Monitor process		12	
CKEditor for Openbravo			
Include module in pack		2	
Master Data Management Extensions			
Company Details		6	
My Company Details new window			
Products			
New columns in M_PRODUCT			
Modifications to Product window			
My Products new window			
New columns in B2B Configuration			
Add Product process			
Pricelists and Pricelists Alerts			
New field in Pricelist Version			

Activity	Cost
Writing Project Documentation (Functional Documentation, Technical Design, Test Battery)	20
Overhead costs of Software Revision Control and security back-ups.	6
Writing User Manuals, Tutorials, Installation and Configuration Manuals, and Troubleshooting Guide	30
Integration Testing	16
Publishing and Support Plan	20
Total Man-Hours:	92



- Portal B2B para proveedores
 - Usuarios Internos y Externos realizan transacciones sobre la misma base de datos
 - Integrado con los flujos estándar de Openbravo ERP
 - Posibilita nuevos modelos de trabajo



Módulo B2B Integration - Extensiones Desarrolladas

Extensiones de
Plataforma

Datos Maestros

Gestión de Pedidos
de Compra

Vendor Managed
Inventory (VMI)

Evaluated Receipt
Settlement

Widgets

The screenshot displays the Openbravo Business Partner interface. At the top, the browser address bar shows the URL: localhost:9080/openbravo/#%7Bst:1,bm:%5B%7BviewId:_OBMyOpenbravoImplementation__params:%7BmyOB:true,canClose:false,tabT... The interface includes a navigation menu on the left with options like 'Inbox (1)', 'Drafts (6)', and 'All Mail'. The main content area shows an email notification titled 'Your access details to QA Testing's B2B Portal' from makobernal@gmail.com to jmarcos.bernal. The email body contains the text: 'Dear Barcelona Contact, You have been granted access to QA Testing's B2B Portal. Your account is now activated and your log-in details are as follows: Username: jmarcos.bernal@gmail.com Password: S9BSnYSV... Log-in to http://localhost:9080/openbravo/#%7Bst:2,bm:%5B%7BviewId:_OBMyOpenbravoImplementation__params:%7BmyOB:true,canClose:false,tabTitle:_... Please refer to the user manual for more details. Yours, The QA Testing's B2B Portal. This is an automated message.' Below the email, there is a 'B2B Activity Log' table with columns for 'When', 'Who', 'What', and 'URL'. The table contains several rows of activity logs, including 'Created an ASN from a Purchase Order', 'Purchase Order has been automatically accepted', and 'Purchase Order has been automatically acknowledged'. At the bottom of the screenshot, a 'WYSIWYG Html Editor - CKEditor for Openbravo' is visible, showing a rich text editor with a toolbar and a text area containing the text: 'This is some sample text. You are using CKEditor.'

Extensiones de Plataforma

Administración de usuarios externos

Seguridad y Roles

Configuración Portal B2B

Motor Email HTML + Editor WYSIWYG

Log de actividades

- B2B Global Parameters
- B2B Organization Parameters
- B2B Vendor Parameters
- B2B Email Templates

Datos Maestros

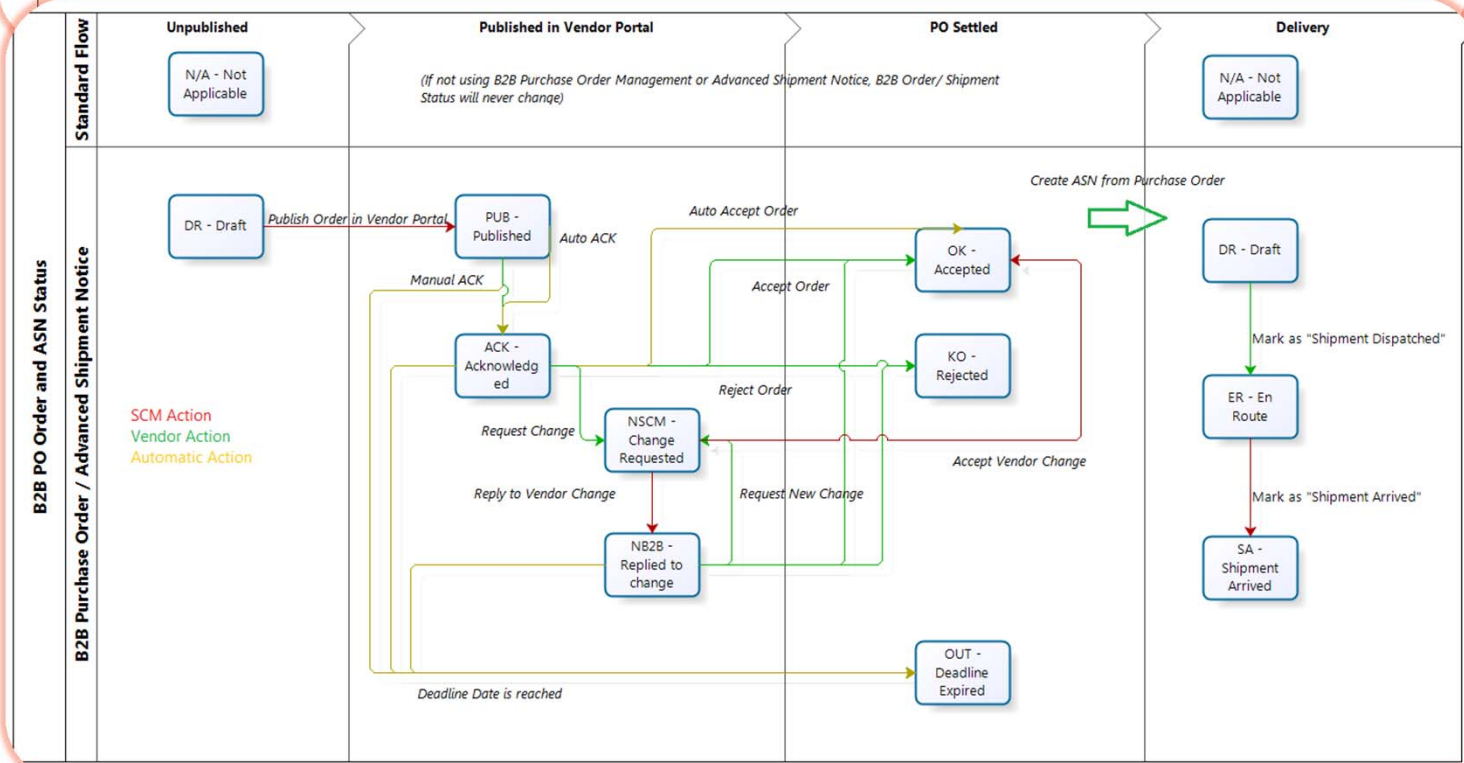
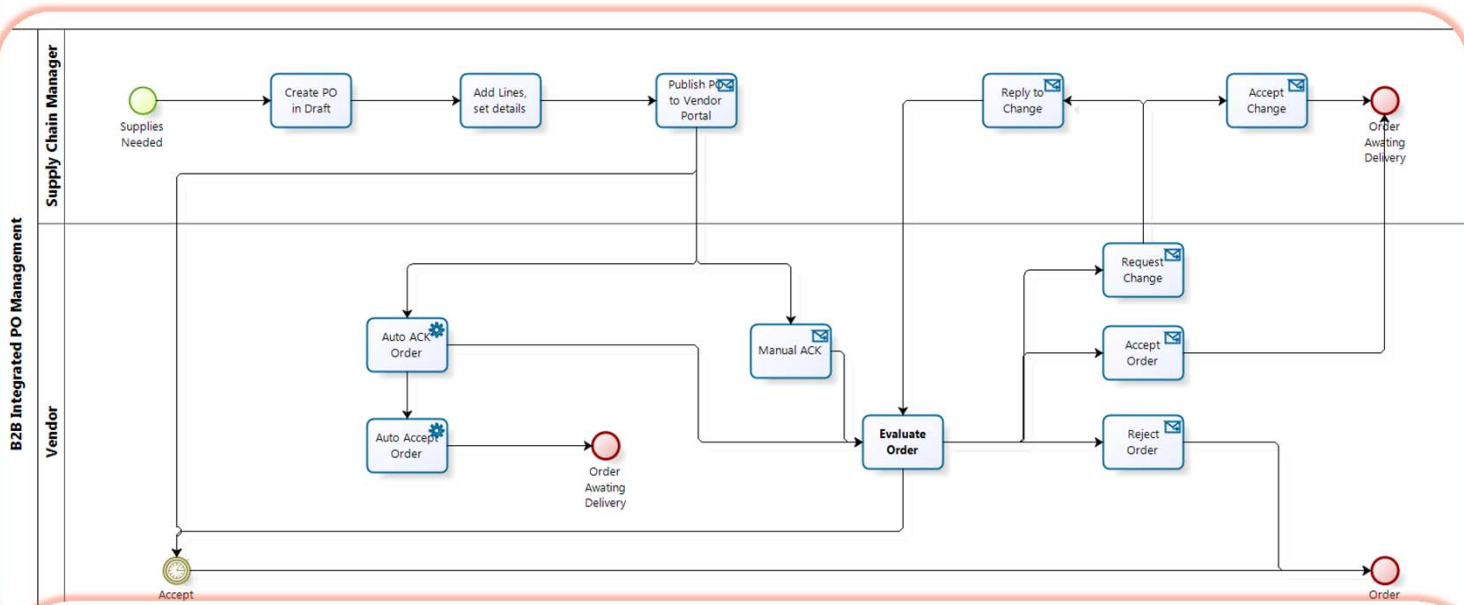
Info Organización

Productos

Listas de Precios

The screenshot displays the Openbravo ERP interface. The top window shows 'My Company Details' for 'MP Enrique' with fields for Organization (Spain), Search Key (vba), Name (MP Enrique), Name 2 (Materias Primas Enrique S.A.), Description (Proveedor de materias primas puras), Tax ID (44688500V), and URL (http://www.mpe.es). A 'Price List Version - Purchase' dialog is open, showing fields for Name (Purchase), Description, Vendor Product Code, Vendor Product Name, UOM (Unit), and Tax Category (Exempt). At the bottom, a 'Product Price' table lists items like Raw material A, B, C, Service A, and Service B with their respective prices.

	Product ▲	Net Standard Price	Net List Price	Price Limit
	Raw material A	2.00	5.00	2.00
	Raw material B	2.00	2.00	2.00
	Raw material C	2.00	2.00	2.00
	Service A	2.00	2.00	2.00
	Service B	2.00	2.00	2.00



Gestión de Pedidos de Compra

Publicación de Pedidos de Compra en Portal B2B

Acuse de Recibo, Aceptación y Rechazo de Pedidos de Compra

Solicitudes de modificación y negociación

Aviso Adelantado de Envío para Pedidos de Compra

Storage Bin	Product	Quantity on H	Minimum Stoc	Resupply Qty.	Auto Resup	Net Standard P	List Price
USA111	Raw material A	10,000		30,000	<input type="checkbox"/>	2.00	5.00
spain111	Raw material A	0		30,000	<input type="checkbox"/>	2.00	5.00
spain111	Raw material A	10,000		30,000	<input type="checkbox"/>	2.00	5.00

B2B Integration Configuration

B2B Vendor: MP Enrique

Vendor Product Code: AZ-R34

Vendor Product Name: Vendor Name

Minimum Stock: 7,500

Resupply Qty: 30,000

AutoResupply

New Advanced Shipment Notice

Add lines for autoresupply products under stock

Dispatch Date: 15-06-2011

Estimated Delivery Date: 15-06-2011

Description:

OK Cancel

Inventario gestionado por el proveedor (VMI)

Almacén administrado por proveedor

Stock Mínimo y Alertas

Aviso Adelantado de Envío

Workspace Business Partner - MP Enrique x B2B Vendor Parameters - MP Enr... x

Organization * Active Business Partner MP Enrique

Vendor Supply Chain Manager * EA - Employee A

B2B Vendor Product Defaults

Default B2B Vendor Product Category: Services
 Default B2B Vendor Product Sequence for Search Key: DocumentNo_AD_Client

B2B Purchase Order Management Options

Automatically Acknowledge Orders (Auto ACK) Accept PO's Automatically

B2B Evaluated Receipt Settlement Options

Validate Invoices automatically Auto Validate invoices after N days * 0 Automatically Acknowledge Payments (Auto ACK)

B2B Vendor Managed Inventory Options

Openbravo - My Invoices - x Gmail - Inbox - makoberna x

localhost:9080/openbravo/#%7Bst:1,bm:%5B%7BviewId:_OBMyOpenbravoImplementation__params:%7BmyOB:true,canClose:false,tabTitle:_Workspace_%7D%7D,%7BviewId:_4028

Application Alerts (0) Help makobernal@gmail.com yourCOMPANY

Workspace My Invoices - 10000000 - 18-05... x

Document Status: Completed | Total Gross Amount: 246,40 | Total Net Amount: 224,00 | Currency: EUR | Payment Complete: No | B2B Invoice Validation Status: AUTO - Automatically Validated Invoice

Payment Terms * 90 days Payment Method 6 (Spain) Purchase Order Order Reference

Payment monitor

Total Paid	Outstanding Amount	Days Till Due *	Due Amount
-246.40	0.00	90	0.00

Document Status Completed

Status

B2B Invoice Validation Status * AUTO - Automatically Validated Invoice

Audit

Lines	Tax	Discounts	Payment Out Plan - 10000000 - ...	Reversed Invoices
1	1			
			Due Date	Payment Method
			Expected Amount	Paid
			Outstanding	Last Payment Out Date
			Number of	
			16-08-2011	6 (Spain)
			-246.40	-246.40
			0.00	16-08-2011

Evaluated Receipt Settlement (ERS)

Creación de Facturas a partir de Pedidos y Envíos

Validación Conjunta de Facturas

Acuse de Recibo de Cobros / Pagos

Document Number	Order Reference	Dispatch Date	Movement Date	Description	Ship. St
PRE-500009-SUF		30-05-2011	30-06-2011		DR
PRE-500008-SUF		30-05-2011	31-05-2011		DR
PRE-500006-SUF		27-05-2011	30-05-2011		SA
PRE-500007-SUF		30-05-2011	30-05-2011	hellou byayby	DR
PRE-500004-SUF			27-05-2011		N/A

Document Number	Order Reference	Order Date	Scheduled Delivery Date	Delivery Notes
800005		26-05-2011	31-05-2011	

Document Number	B2B Order	Publish Date	Deadline Date	Amount	Currency	Sched Delivery
800005	OUT	26-05-2011	25-05-2011	37.46	EUR	31-05-2011
800008	OK	27-05-2011	28-05-2011	0.00	EUR	30-05-2011

Storage Bin	Product	Quantity on Hand	Minimum Stock	Resupply Qty.	Auto Resup	Net Standard P	List Price
USA111	Raw material A	10,000		30,000	<input type="checkbox"/>	2.00	5.00
spain111	Raw material A	0		30,000	<input type="checkbox"/>	2.00	5.00

Document Number	Order Reference	Invoice Date	Payment Method	Payment Terms	Total Amount	Currency	Validation
1000000		07-05-2011	0 (Spain)	00 (Spain)	0.00	EUR	N/A

Days Till Due	Due Amount	Currency	Document Number	Paid Amount	Outstanding	Total Amount	Payment Method
0.00	0.00	EUR	10000004	0.00	0.00	0.00	6 (Spain)

Document Number	Reference No.	Date	Amount	Currency	Payment Method	ACK Status
700002		25-08-2011	37.46	EUR	6 (Spain)	N/A
700001		25-08-2011	37.46	EUR	6 (Spain)	N/A
700003		25-08-2011	37.46	EUR	6 (Spain)	N/A
700000		16-08-2011	246.40	EUR	6 (Spain)	OK

Σ 0.00 Σ -283.86 Σ 0.00 Σ 283.86

Widgets

Cuadro Resumen Productos

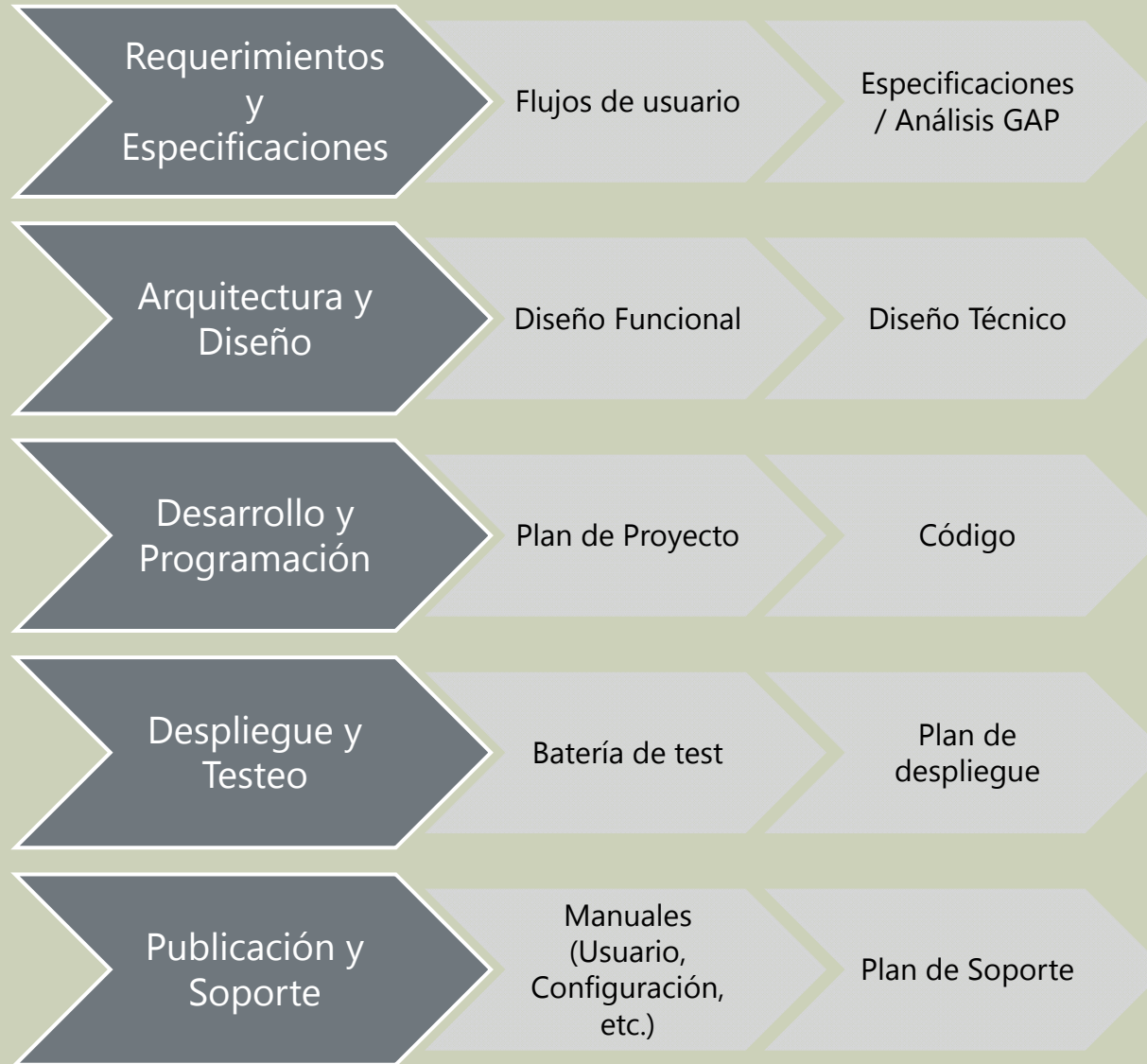
Pedidos de Compra y Pedidos B2B

Envíos de Material

Facturas y Facturas Abiertas

Cobros / Pagos

Metodología de desarrollo de software



Proyecto de desarrollo de Software

Análisis de flujos de negocio y GAP Funcional

Diseño Funcional y Técnico

Plan de Proyecto

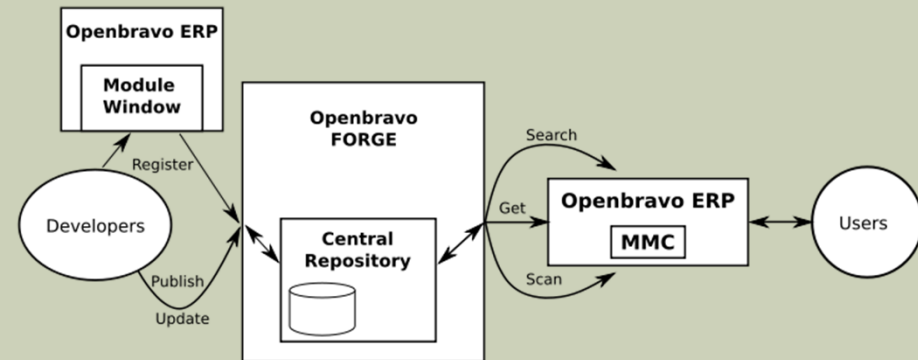
Desarrollo

Publicación

Planes Futuros

Publicación. Openbravo Forge.

- Empaquetado y Publicación del módulo en el CR de Openbravo ERP



- Proyecto en *Openbravo Forge*

- Foros
- Issue Tracker
- Repositorio de código
- Wiki
- Sección de Noticias
- Sección de Descargas
- Reseñas, Ideas, Preguntas



<http://forge.openbravo.com/projects/b2bintegration>



Planes de futuro y extensiones

- **Mantenimiento y Soporte**

- Soporte directo a usuarios
- Publicación periódica de nuevas versiones
- Materiales de soporte y marketing
- Servidor de Demo

- **Extensiones futuras**

- Traducción y localización española
- Extensión de la gestión de pedidos de compra
- Informes extendidos, widgets y centro de control
- Funcionalidades demandadas por los usuarios

- **Proceso de diseño**
 - Búsqueda de problema
 - Análisis Flujos y Gap Funcional
 - Proyecto de desarrollo de SW
 - Publicación, Mantenimiento y Soporte
- **Proyecto vivo**
- **Beneficios directos**
 - Futuro de los ERP
 - Base para futuros PFCs de temática similar
 - Demostración de Tecnología
- **Contribución libre a una comunidad open-source**

Conclusiones

**GRACIAS POR SU
TIEMPO**

¿Dudas?