



OpenERP

Overview, Introduction to Administration & Development



– EFREI –

– ESTIA –

Guillaume Rivière

Last update: March 2018



OUTLINE

1. Overview

- Birth and growth
- Customers, Partners, Vendors

2. Administration

- Versions, Architecture
- Main modules

3. Development

- Module programming
- Webservice programming

4. Documentation



WARNING !!!



- You are going to learn **Open ERP**
OPEN SOURCE MANAGEMENT SOLUTION
 - But keep in mind to select an ERP **only** because it is the best adapted to the case of your company

Our competitors use this ERP !!!

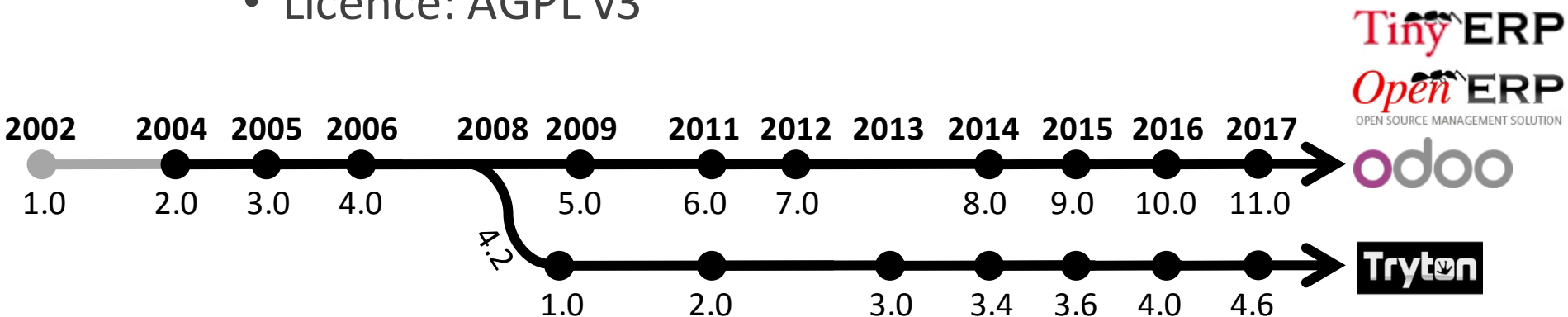
This is the only ERP I know...

This ERP fits all the needs of my company.



History of OpenERP

- Initiated by Fabien Pinckaers
 - in 2002, in Belgium
 - Written in Python
 - Database is PostgreSQL
 - Open Source (*but not free software, according to FSF*)
 - Licence: AGPL v3



OpenERP today

- Used for teaching since 2008
 - The CERTA network teach accounting to BTS
 - CERPEG
- Around the world, we find dozens of **Customers, Vendors and Partners**
- Still supported Long Term Stable versions (LTS)
 - Version 7.0 released in December 2012
 - Version 8.0 released in September 2014
 - Version 9.0 released in October 2015
 - Enterprise (20€/user/month) & Community (90%) Editions

Hardware needed

- 5 employees: 2 CPU 2GB RAM
- 20 employees: 4 CPU 8GB RAM
- 90 employees: 2x 8 CPU 32GB RAM
 - Split application and database server
- 250+ employees: 2x 12 CPU 128 GB RAM
 - Load balancing

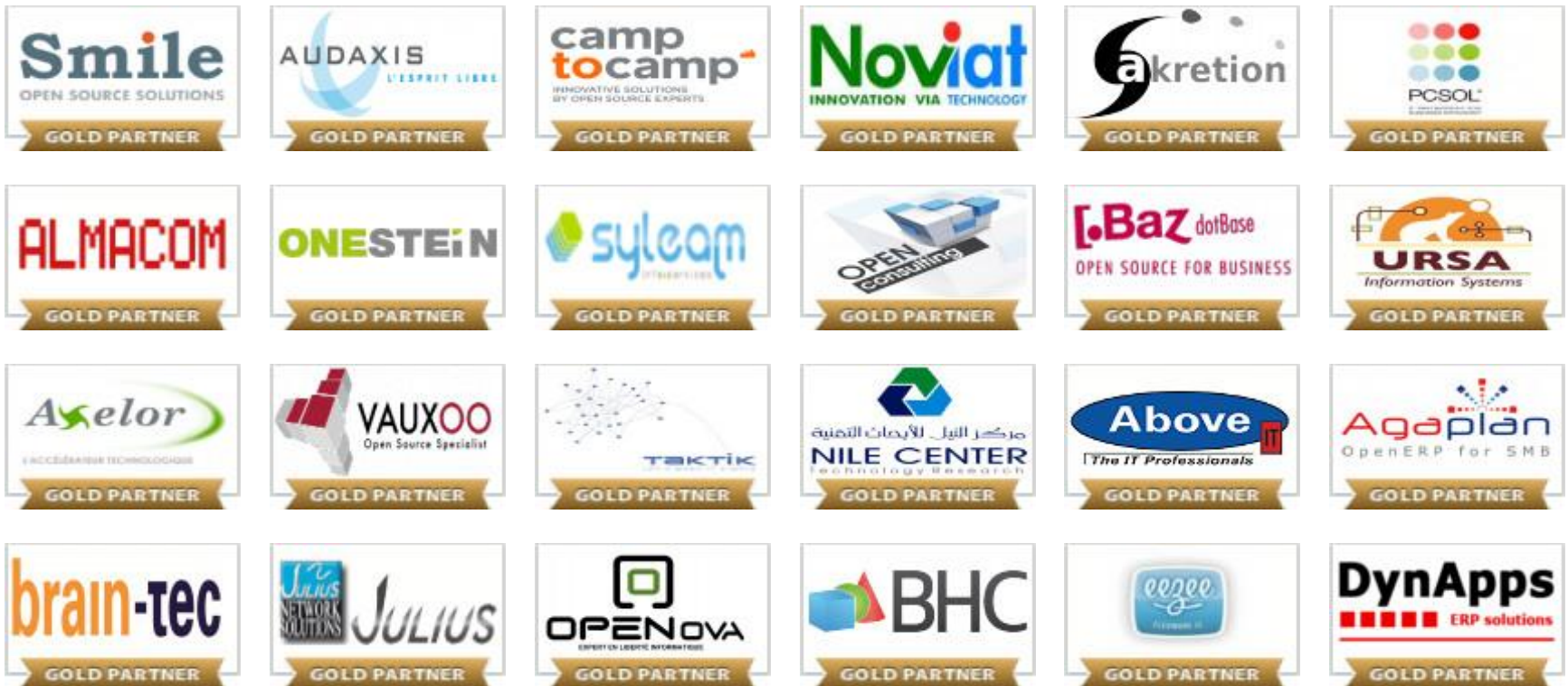
Vendors / Consultants

- Smile
- Julius
- Business & Decision
- Aunéor Conseil
- Anybox
- Camptocamp
- ZAD solutions
- Audaxis
- *and many many others...*



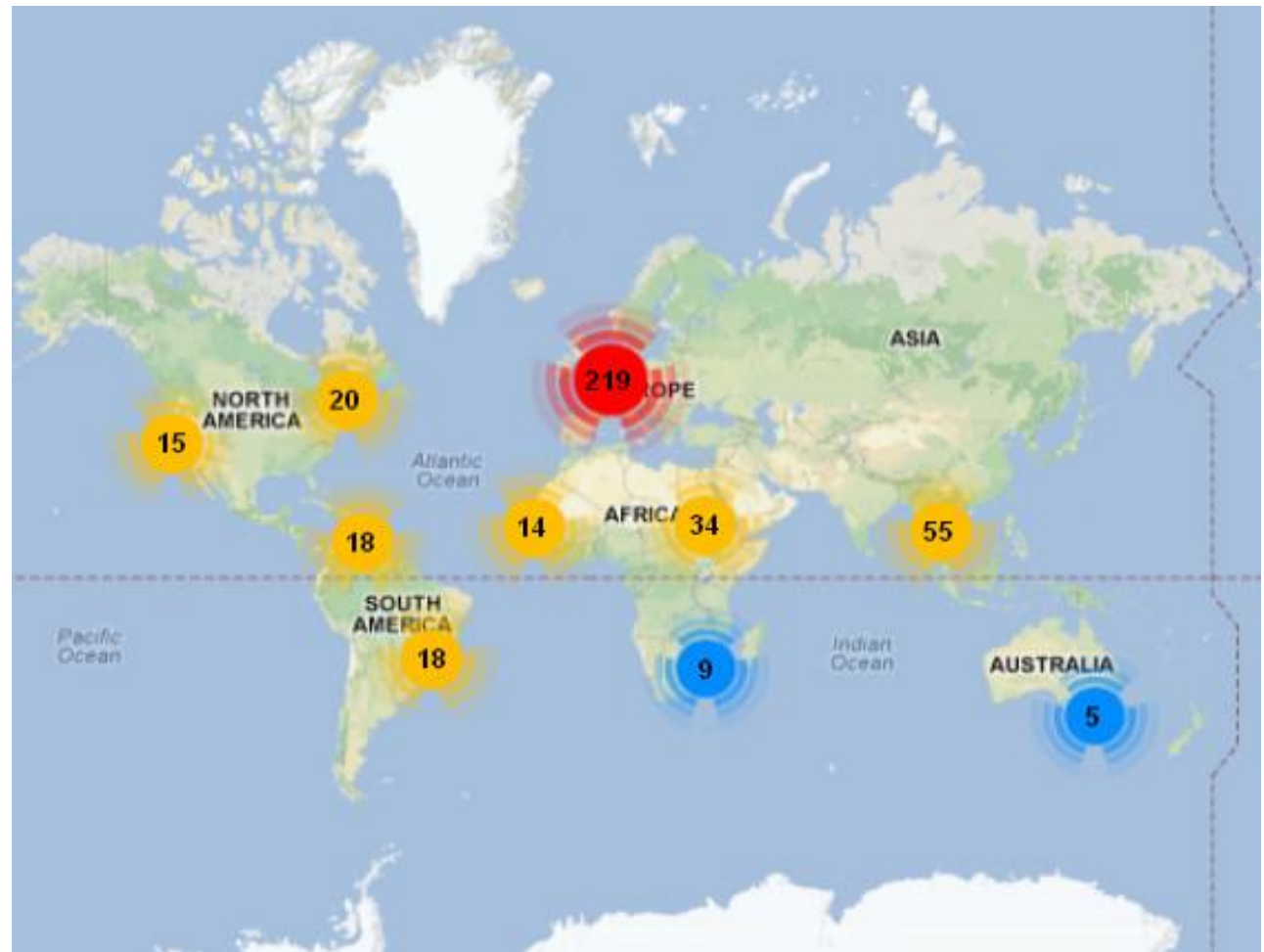
Partners

- Gold / Silver / Ready / Training Center
 - Pay annual fees



Partners' world map

Western Europe	219
Africa	57
Asia	55
North America	35
South America	36
Australia	5



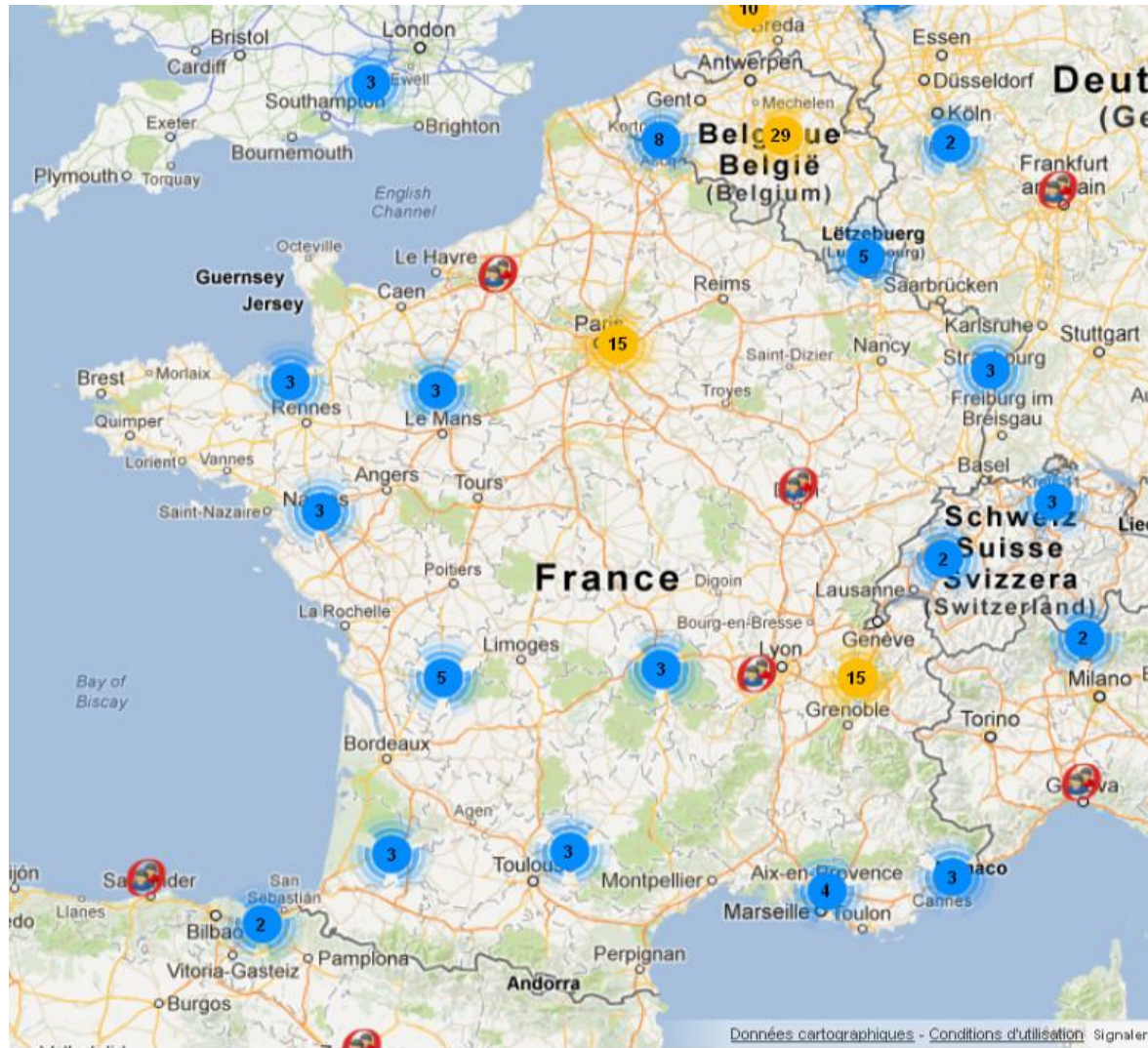
Partners worldwide



Partners in France



(March 2013)



Customer references



OUTLINE

1. Overview

- Birth and growth
- Customers, Partners, Vendors

2. Administration

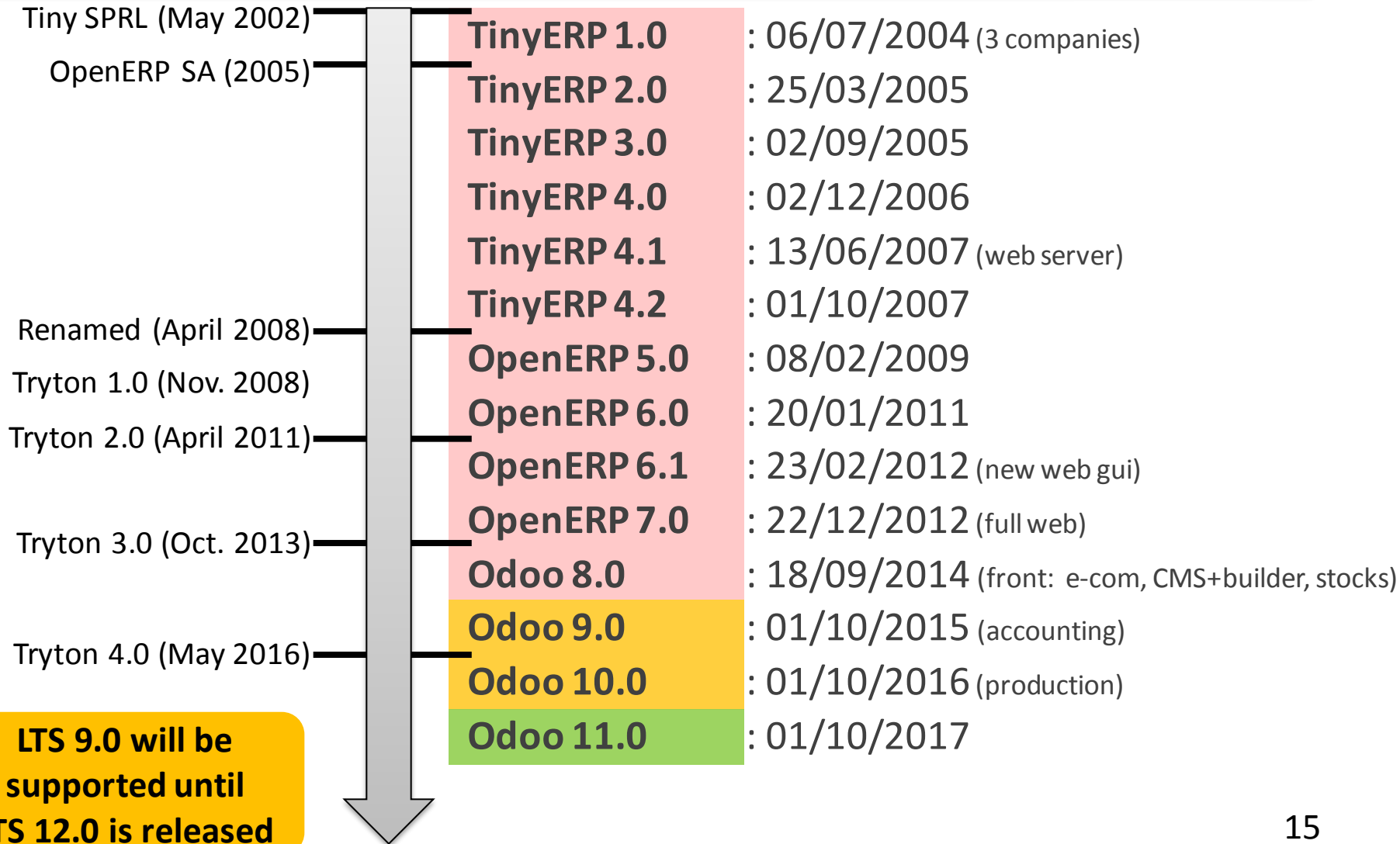
- Versions, Architecture
- Main modules

3. Development

- Module programming
- Webservice programming

4. Documentation

Timeline of versions



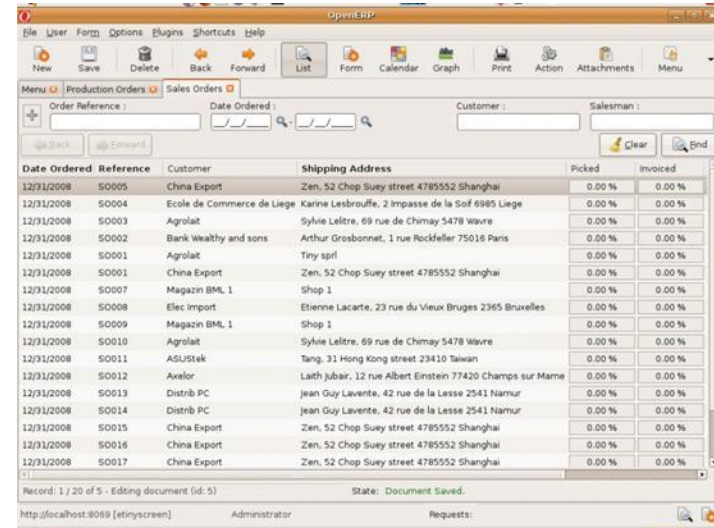
Elements to run OpenERP

- Client

- GTK client: until version 6.x
- Web client: since version 4.1
- « full web »: since version 7.0

- Server




- Separated web server: from 4.1 to 6.x
- Version 7.0: one-piece server
 - v7.0 : Python \geq 2.6 (Python 2.7)
 - v6.1 : Python \geq 2.6
 - v6.0 : Python \geq 2.5



The screenshot displays the OpenERP web client interface. At the top, there is a menu bar with options like 'File', 'Liver', 'Form', 'Options', 'Plugins', 'Shortcuts', and 'Help'. Below the menu is a toolbar with icons for 'New', 'Save', 'Delete', 'Back', 'Forward', 'List', 'Form', 'Calendar', 'Graph', 'Print', 'Action', 'Attachments', and 'Menu'. The main content area shows a 'Sales Orders' view with a search bar for 'Order Reference' and 'Date Ordered', and a 'Customer' field. Below this is a table with columns: 'Date Ordered', 'Reference', 'Customer', 'Shipping Address', 'Picked', and 'Invoiced'. The table contains 17 rows of data, including entries for 'China Export', 'Ecole de Commerce de Liege', 'Agrolat', 'Bank Wealthy and sons', 'Magazin BML 1', and 'ASUStek'. At the bottom, there is a status bar showing 'Record: 1 / 20 of 5 - Editing document (id: 5)' and 'State: Document Saved.'.

Date Ordered	Reference	Customer	Shipping Address	Picked	Invoiced
12/31/2008	S0005	China Export	Zen, 52 Chop Suey street 4785552 Shanghai	0.00 %	0.00 %
12/31/2008	S0004	Ecole de Commerce de Liege	Karine Lesbrouffe, 2 Impasse de la Soif 6985 Liege	0.00 %	0.00 %
12/31/2008	S0003	Agrolat	Sylvie Lelitre, 69 rue de Chimay 5478 Wavre	0.00 %	0.00 %
12/31/2008	S0002	Bank Wealthy and sons	Arthur Grosbonnet, 1 rue Rockefeller 75016 Paris	0.00 %	0.00 %
12/31/2008	S0001	Agrolat	Tiny sprl	0.00 %	0.00 %
12/31/2008	S0001	China Export	Zen, 52 Chop Suey street 4785552 Shanghai	0.00 %	0.00 %
12/31/2008	S0007	Magazin BML 1	Shop 1	0.00 %	0.00 %
12/31/2008	S0008	Elec import	Etienne Lacarte, 23 rue du Vieux Bruges 2365 Bruxelles	0.00 %	0.00 %
12/31/2008	S0009	Magazin BML 1	Shop 1	0.00 %	0.00 %
12/31/2008	S0010	Agrolat	Sylvie Lelitre, 69 rue de Chimay 5478 Wavre	0.00 %	0.00 %
12/31/2008	S0011	ASUStek	Tang, 31 Hong Kong street 23410 Taiwan	0.00 %	0.00 %
12/31/2008	S0012	Axelor	Laith jubair, 12 rue Albert Einstein 77420 Champs sur Marne	0.00 %	0.00 %
12/31/2008	S0013	Distrb PC	Jean Guy Lavente, 42 rue de la Lesse 2541 Namur	0.00 %	0.00 %
12/31/2008	S0014	Distrb PC	Jean Guy Lavente, 42 rue de la Lesse 2541 Namur	0.00 %	0.00 %
12/31/2008	S0015	China Export	Zen, 52 Chop Suey street 4785552 Shanghai	0.00 %	0.00 %
12/31/2008	S0016	China Export	Zen, 52 Chop Suey street 4785552 Shanghai	0.00 %	0.00 %
12/31/2008	S0017	China Export	Zen, 52 Chop Suey street 4785552 Shanghai	0.00 %	0.00 %

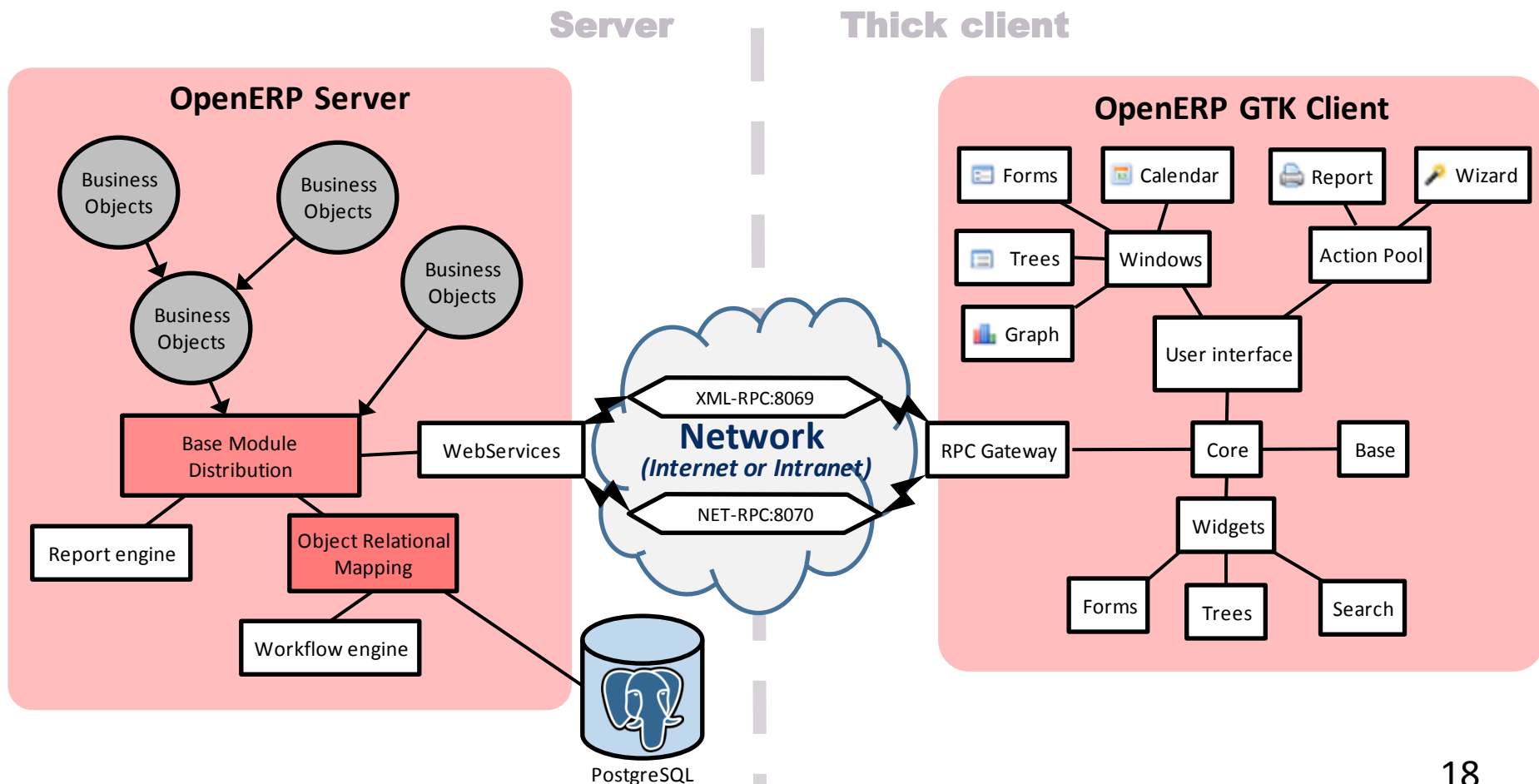
Compatibility

- Server operating system
 - Linux, Windows, Mac OS   
- Web browser

	(OpenERP 7)	(Odoo 10)
– Mozilla Firefox	(≥ 13)	✓
– Microsoft IE	(≥ 9.0)	-
– Microsoft Edge	-	✓
– Safari	(N.C.)	✓
– Google Chrome	(≥ 22)	✓
- OpenERP Mobile Client (Smartphones, tablets)
 - Application Android, iOS (alternative: myodoo.com)

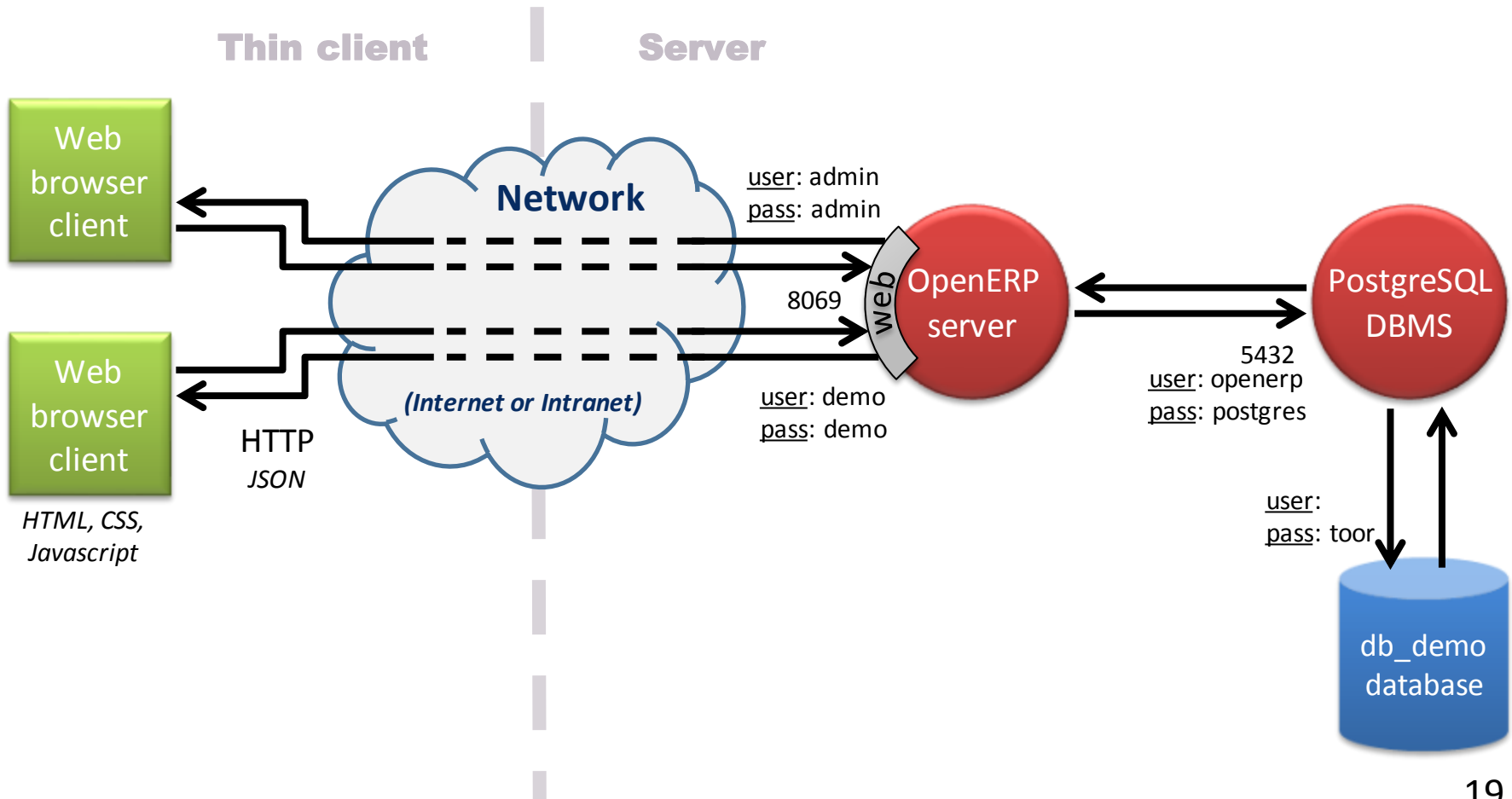
Technical Architecture

- OpenERP v6 (for GTK client)

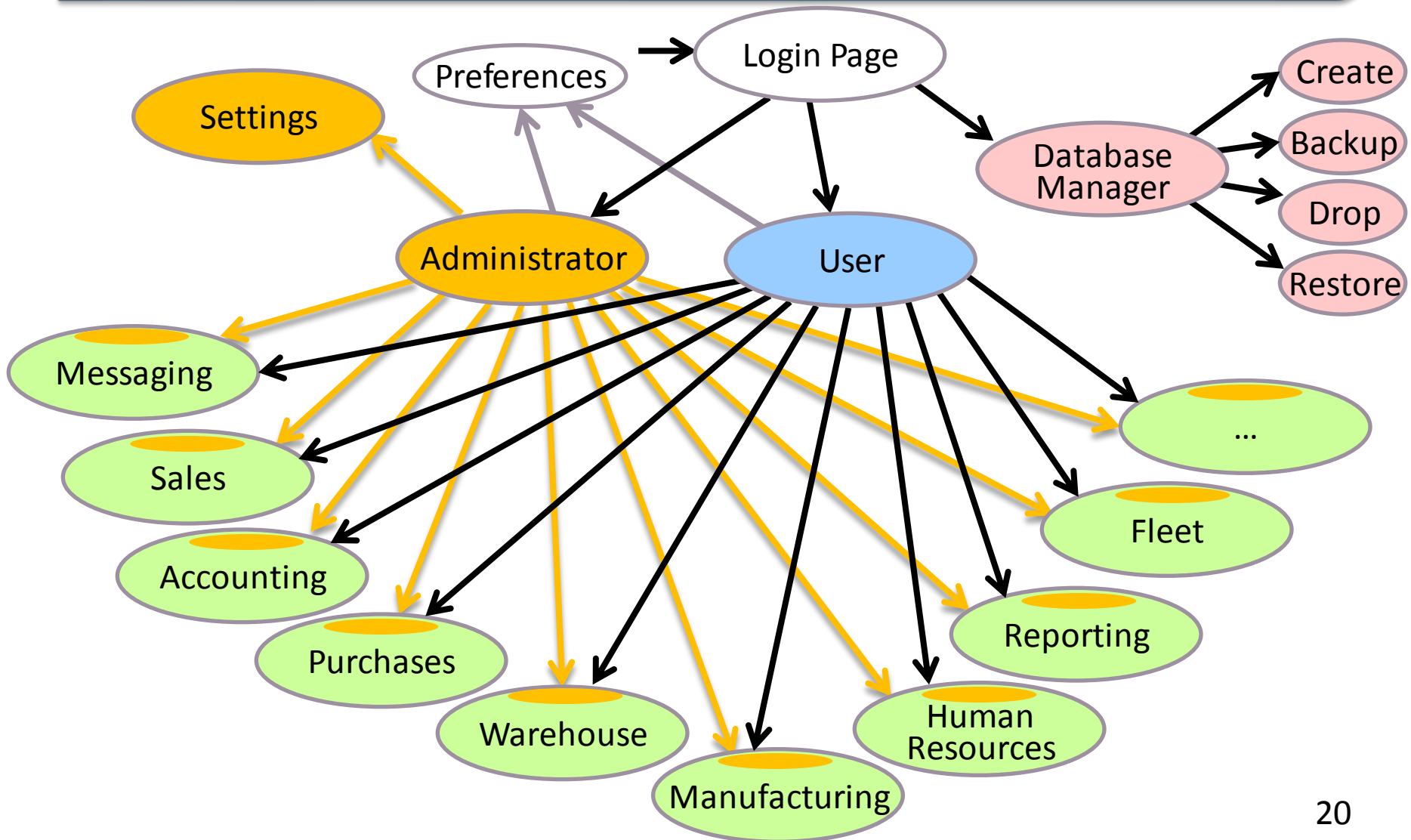


Technical Architecture

- OpenERP v7 v8 v9 (web client)



Global navigation

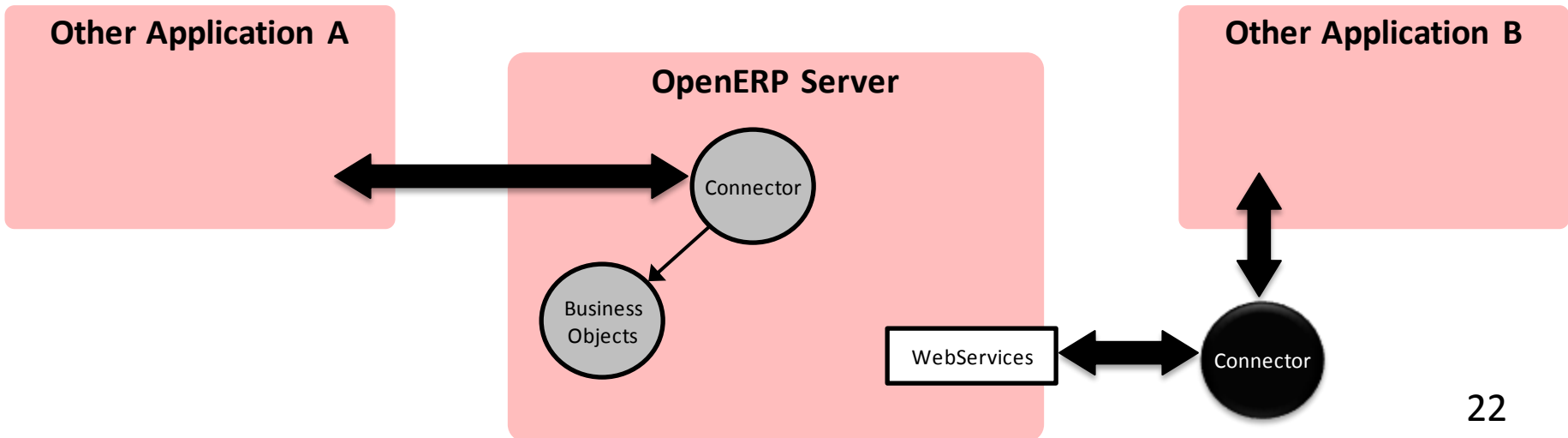


Administration

- Administrator(s) must define
 - The users
 - The access rights
 - The languages
 - The menus
 - The basis data of the company
 - Products, work centers, flows, etc.
 - The modules to install
 - The connections with other software
 - and many other things...

Connectors

- Enterprise Application Integration (EAI)
- Interoperate OpenERP and other programs
 - Webservice (program in Python, Java, PHP, C/C++, ...)
 - Business objects (module in Python)
 - <http://openerp-connector.com/>



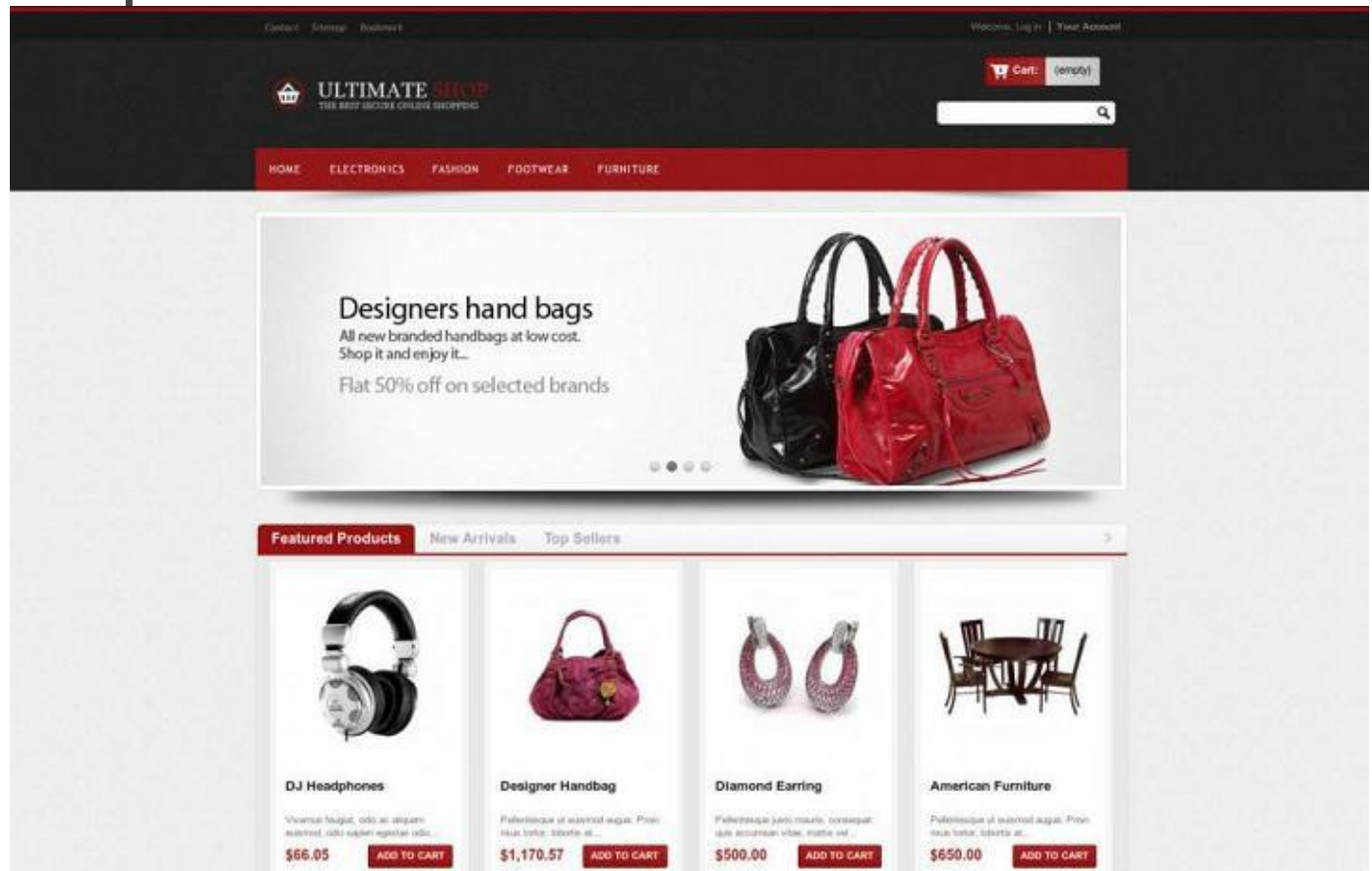
Connectors

- CMS
 - Joomla! (PHP/MySQL)
- E-commerce
 - OSCommerce (PHP/MySQL)
 - Magento (PHP)
 - PrestaShop (PHP/MySQL)
- CMS + eCommerce
 - eZ Publish (PHP/MySQL, PostgreSQL, SQLserver, Oracle)
- Reporting
 - JasperReports (Java)
 - ReportLab (Python)
- Business Intelligence
 - Pentaho (Java)
 - SpagoBI (Java)



E-commerce

- Prestashop



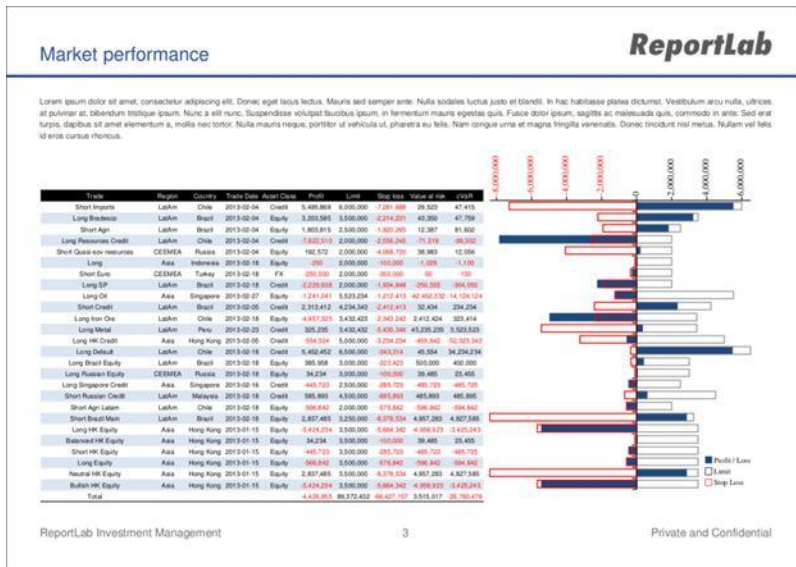
Reporting

- JasperReports



Reporting

- ReportLab
 - Financial reporting and dynamic graphics



- Natively used in OpenERP for reporting diagrams

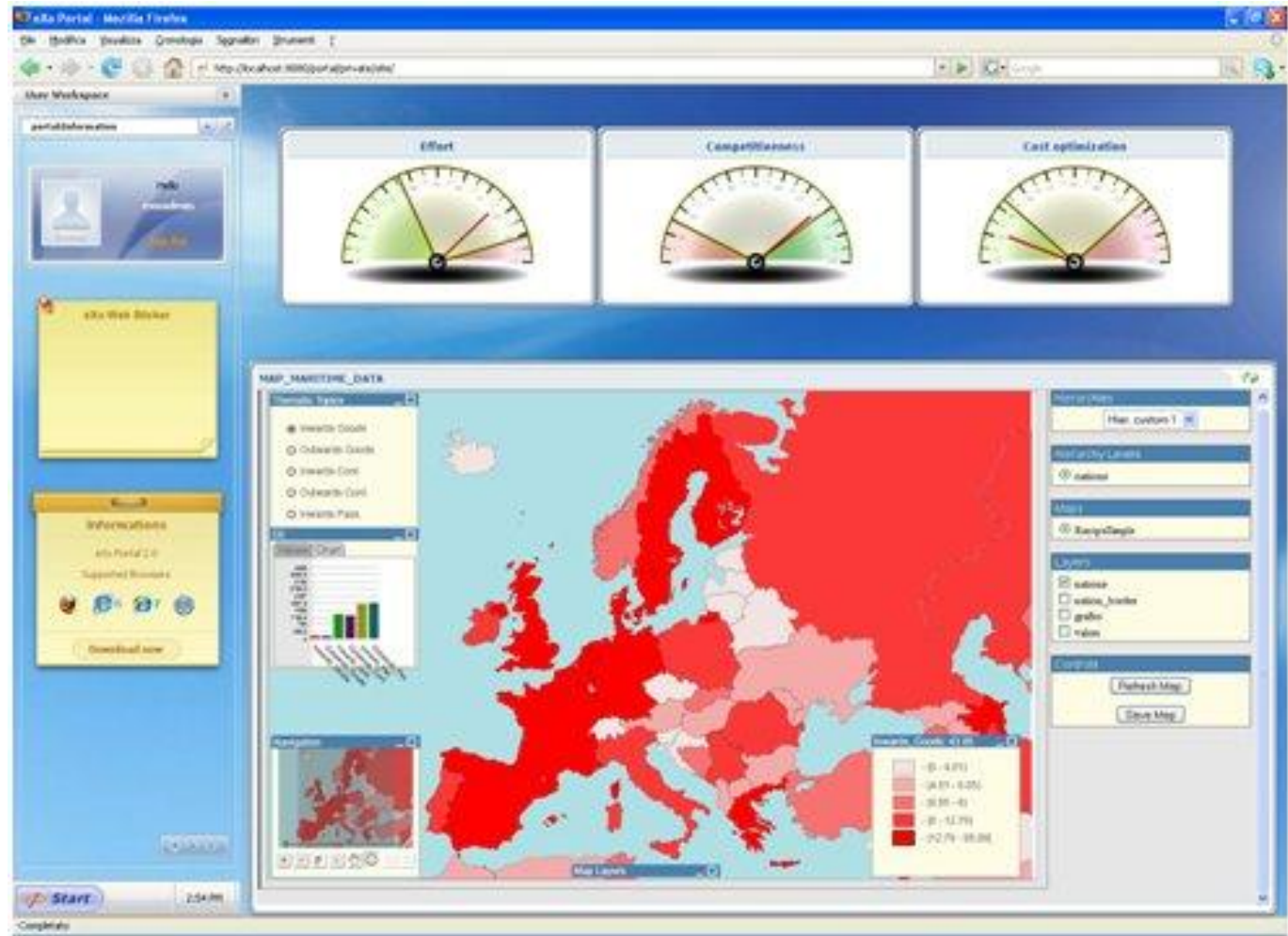
Business Intelligence

- Pentaho



Business Intelligence

- SpagoBI



VOIP

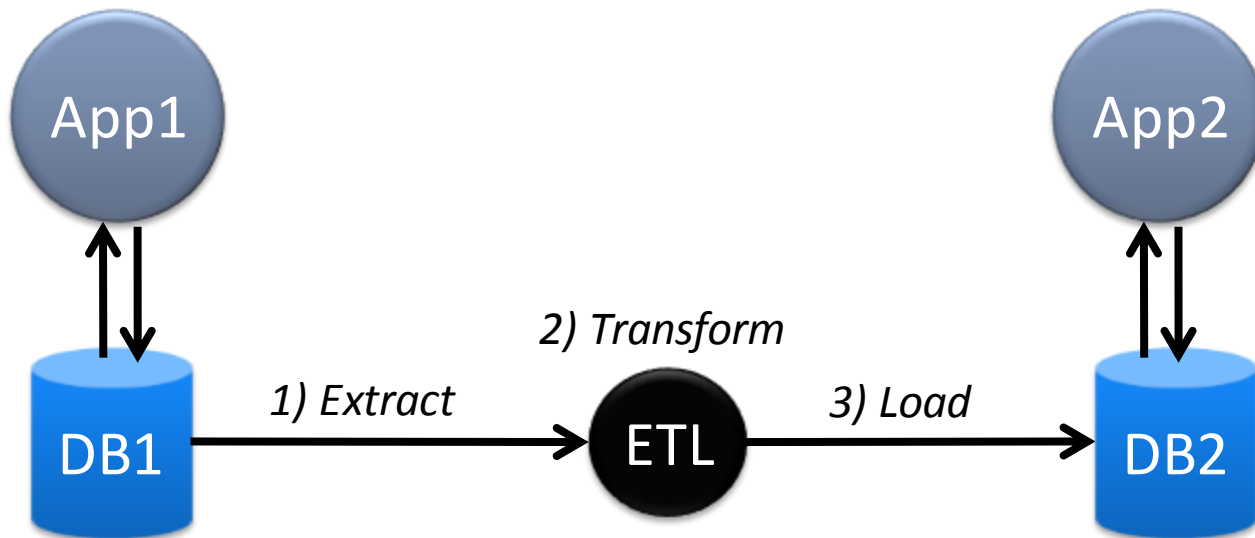
- CRM
 - Routing phone calls
 - Open the customer master record when they are calling
 - Call-center VoIP with Asterisk server
 - Automatically call a customer

The screenshot displays the OpenERP CRM interface for a customer record. The browser title is "OpenERP - socket://admin@localhost/erpv61_ast2". The interface includes a top toolbar with icons for New, Save, Delete, Previous, Next, List, Form, Calendar, Diagram, Graph, Print, Action, and Attachments. The main content area shows the customer "Akretion" with fields for Name, Reference, Partner Firm, Language (English), and checkboxes for Customer and Supplier. Below this are tabs for General, Sales & Purchases, Accounting, History, and Notes. The "Partner Contacts" section is active, showing details for "Raphaël Valyi" (Sir) with fields for Contact Name, Title, Function, Phone (+55 2125162954), Mobile, Fax, E-Mail (contact@akretion.co), and Postal Address (Rua Acre nº47, CEP 20.081-000, Rio de Janeiro, Brazil). A right-hand sidebar contains a list of actions: Labels, Overdue Payments, Send an SMS, Mass Mailing, Create Opportunity, Events, Emails, Schedule a Meeting, Schedule a Call, Monthly Turnover, and Receivables & Payables. The status bar at the bottom indicates "No record selected" and "State: Document Saved."

Click2dial

ETL

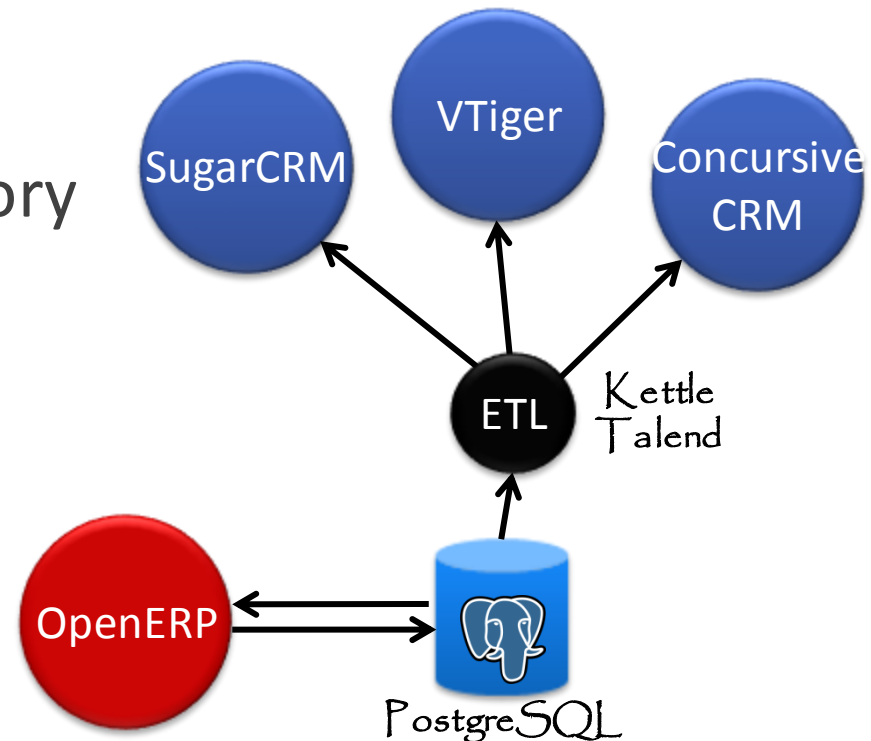
- **Extract-Transform-Load**



- Integrate data from multiple databases of different applications

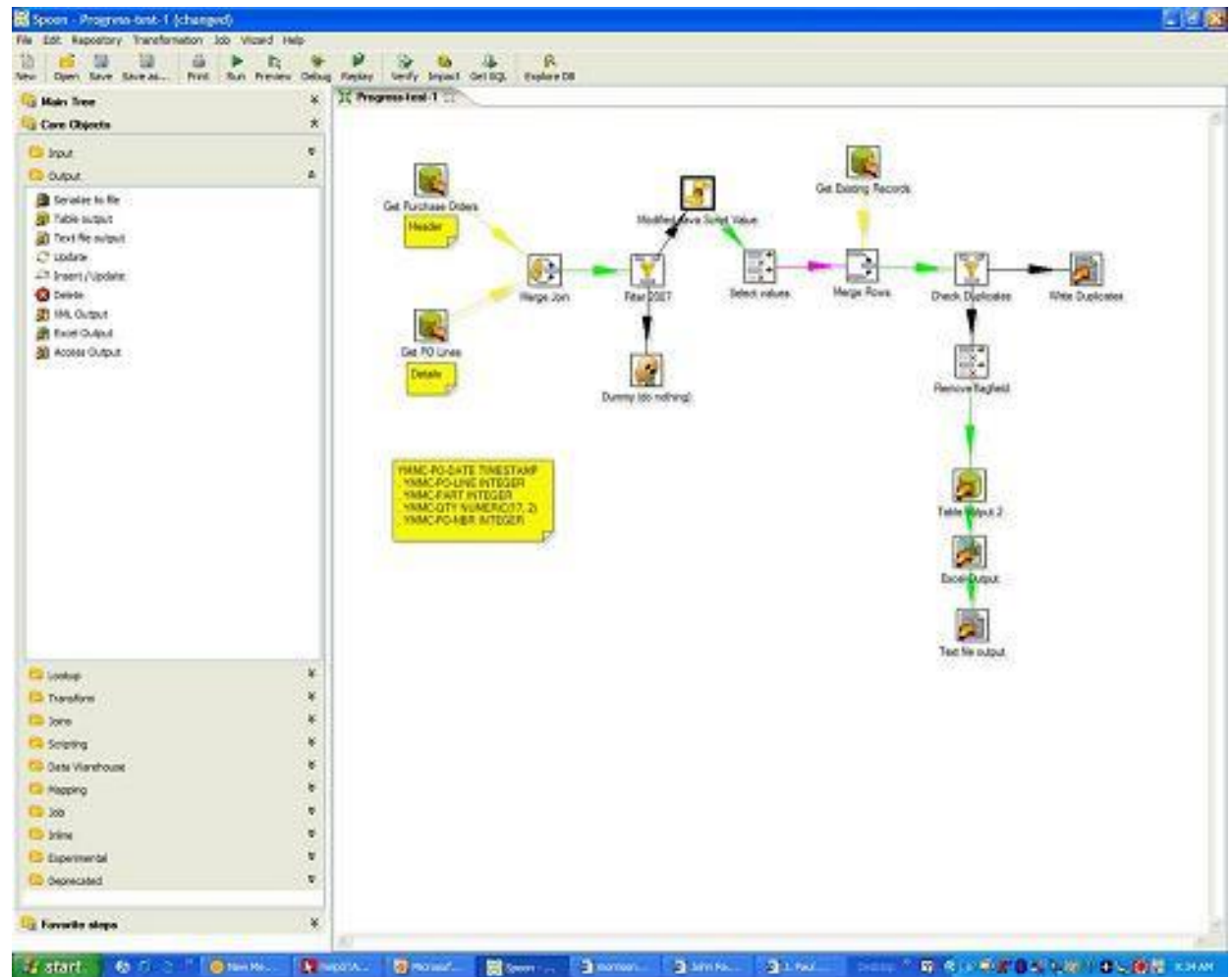
ETL

- CRM
 - Targeting the customers
 - Data mining of sales history
 - Weka (Java)
 - SugarCRM (PHP)
 - vTiger (PHP)
 - Concurive CRM (Java)



ETL

- Kettle



ETL

- Talend

Talend Open Studio (2.0.0.r3065) | MyUser@Name.com | Proj project1

File Edit View Window Help

100%

Repository Navigator

- Business Models
- Job Designs
 - Misc
 - v07
 - v08
 - v081
 - AppendLog 0.1
 - ComprehensiveMaster 0.1
 - LogStatCatcher_2 0.1
 - Master 0.1
 - MrandMycolor 0.1
 - StatLogCatch_DB1 0.1
 - StatLogCatchDie 0.1
 - NMLZDB 0.1
- Contexts
 - ContextDB 0.1
- Code
- Metadata
- Documentation
- Recycle bin

Outline Code Viewer

- ELTMySQLInput_1
- ELTMySQLInput_2
- ELTMySQLInput_3
- ELTMySQLMap_1
- ELTMySQLOutput_1
- tWarn_1
- tWarn_2
- tStatCatcher_1
- tLogCatcher_1
- tMsgOutput_1
- tMsgOutput_2

Job: ETLMySQLInsert

owners (Table) cars (Table) resellers (Table) ELT MySQL Insert results (table, order) tWarn_1 tWarn_2 tStat_Catcher_1 tMsgOutput_1 tStat_Catcher_1 tMsgOutput_2

Designer Code

Properties PerDoc RealExp Tasks Problems (Job ETLMySQLInsert) Run (Job ETLMySQLInsert) Modules Scheduler Error Log

Contexts

ELTMySQLInsert

Context Type: Repository CONTEXT: ContextDB Default Context: Test

Test Prod

Ask for Confirmation Parameters: Add Remove Context: Copy Rename Remove

Prompt	Name	Prompt Message	Type	Default	Comment	Script code
<input checked="" type="checkbox"/>	Test	talend-studio	String	talend-studio-test		\$_context[Test]

CRM

- SugarCRM
 - B2B
 - B2C

The screenshot displays the SugarCRM web interface within a Mozilla Firefox browser window. The interface includes a navigation menu with options like Home, Sales, Marketing, Support, Accounts, Collaborative, Tools, and Reports. The main content area is divided into several sections:

- My Top Open Opportunities:** A table listing top opportunities with columns for Opportunity Name, Amount, Expected Close Date, and Status. Data includes entries for White Cloud Co., B2B, Double Cloud Co., and J&B Foods Ltd.
- My Calls:** A table listing calls with columns for Close, Subject, Duration, Start Date, and Accept. Data includes entries for 'Bad time, will call back' and 'Get more information on the proposed deal'.
- My Meetings:** A table listing meetings with columns for Close, Subject, Duration, Start Date, and Accept. Data includes entries for 'Sales call on the phone' and 'Sales call on the phone'.
- My Accounts:** A table listing accounts with columns for Account Name, Phone, and Date Entered.
- My Pipeline:** A funnel chart showing the sales pipeline with stages: 100% (Prospect), 10% (Qualified Lead), 5% (Sales Analysis), and 1% (Final Decision). The funnel is color-coded with red, green, blue, orange, and green segments.
- My Leads:** A table listing leads with columns for Name, Office Phone, and Date Created. Data includes entries for 'Accountants Practice' and 'Acme Truck'.

CRM

- vTiger
 - Fork of SugarCRM



PLM / PDM

- OpenERP PLM
– OmniaSolutions



Advanced Search

Name : Part Number : Status : Description :

Buttons: Pulsed, Cerca

Name	Part Number	Revision	Status	Description
CM000011	CM000011	0	Draft	VITE TCEI UNI 5931
FC052402	FC052402	0	Draft	FLANGIA ECCENTI
FC052403	FC052403	0	Draft	FLANGIA ECCENTI
FC052405	FC052405	0	Draft	FLANGIA MOTORE
FC072407_A	FC072407_A	0	Draft	SUPPORTO FISSO
FC072408	FC072408	0	Draft	RULLO PER TELESC
FC072409	FC072409	0	Draft	PERNO PER RULLO
FC072424	FC072424	0	Draft	FLANGIA
FC072491_A	FC072491_A	0	Draft	ASSIEME SUPPORT
KIM006	KIM006	0	Draft	MOTORE OMT315
KMC100	KMC100	0	Draft	CUSCINETTO INA

Explose Bom | Where Used Bom

name	description
FC072491_A	ASSIEME SUPPORTO FISSO
FC072407_A	SUPPORTO FISSO TELESCOPIO
FC052402	FLANGIA ECCENTRICA CON CHIAVETTA
FC052403	FLANGIA ECCENTRICA
FC052405	FLANGIA MOTORE ECCENTRICA
FC072424	FLANGIA
KMV110	SPINA CILINDRICA TEMPERATA DIA 12X90
KMV143	VITE TE M24X2X100 CL.8.8 UNI5740 INTERAMENTE FILETTATA

Creator: Administrator, Date Created: 06/06/2011 21:3..., Attachment Name: FC072491_A-00..., Revision Index: 0

OK Cancel

Manage management systems

- Certifications
 - Quality (ISO 9001)
 - Environment (ISO 14001)
 - Security (ISO 27001, PCI-DSS)
 - IT services (ISO 20001, ITIL)
 - Health & Safety (OHSAS 18001)
- Extra menus and tabs
 - In several modules

www.pcsol.be

www.savoirfairelinux.com

User Interface

- User interfaces of ERPs are often quite simple

The screenshot displays the OpenERP v7.0 user interface. It features a top navigation bar with modules like Messaging, Sales, Invoicing, Purchases, Warehouse, Manufacturing, Tools, and Reporting. A left sidebar menu is highlighted in yellow, showing the current module's sub-menu. The main view is a table of Manufacturing Orders, highlighted in blue. A red box highlights the top navigation bar.

(1) Modules

(2) Menu of the current module

(3) Main view

Reference	Scheduled Date	Product	Product Quantity	Total Hours	Total Cycles	Source Document	Status
MO/00003	12/02/2013 12:16:44	[C-Case] Computer Case	3.000	0.00	0.00	MO/00001	Awaiting Raw Materials
MO/00002	12/04/2013 12:16:44	[LAP-CUS] Laptop Customized	1.000	0.00	0.00		Awaiting Raw Materials
MO/00001	12/04/2013 12:16:44	[PCSC349] PC Assemble SC349	3.000	7.13	1.50		Awaiting Raw Materials
			7.000	7.13	1.50		

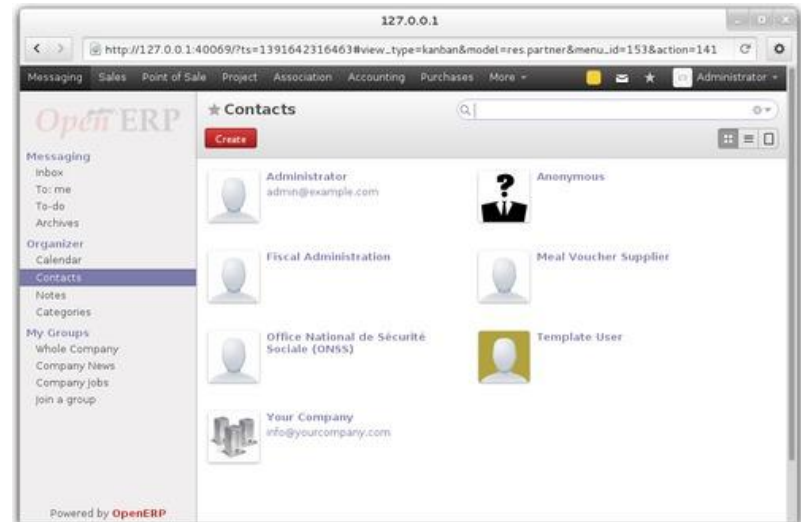
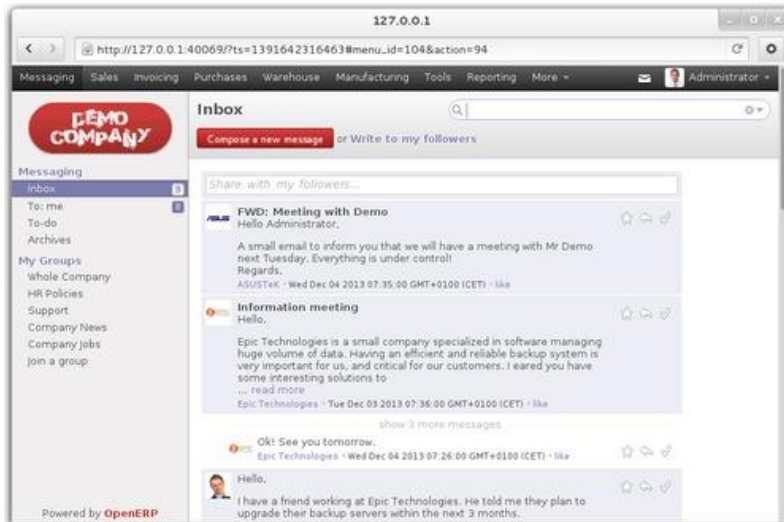
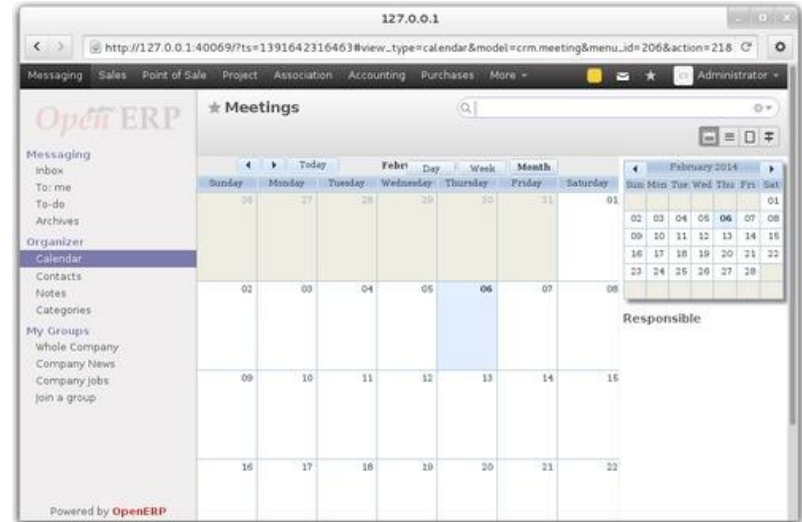
Main modules

Messaging Sales Invoicing Purchases Warehouse Manufacturing Tools Reporting More ▾   Administrator ▾

- Messaging
- Sales
- Point of Sales
- Project
- Association
- Accounting
- Purchases
- Warehouse
- Manufacturing
- Marketing
- Human Resources
- Fleet
- Events
- Knowledge
- Tools
- Reporting

Messaging

- Inbox
- Meetings calendar
- Contacts
- ...



Sales

- Customers
- Products
- ...

The image displays two screenshots of the OpenERP v7.0 web interface. The left screenshot shows the 'Products' view, featuring a grid of product cards with details such as name, price, on-hand quantity, and forecasted quantity. The right screenshot shows the 'Customers' view, displaying a list of customer profiles with their names, roles, and contact information.

Products View:

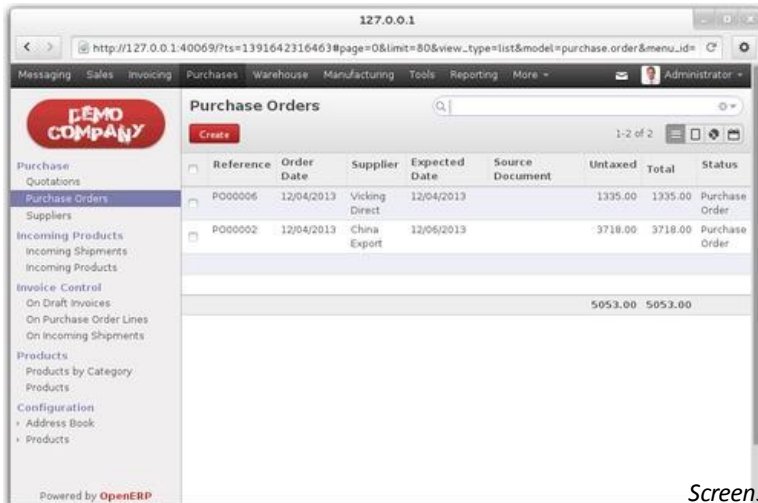
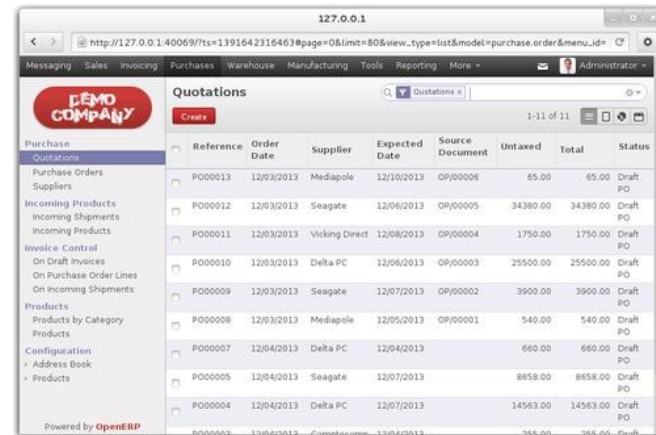
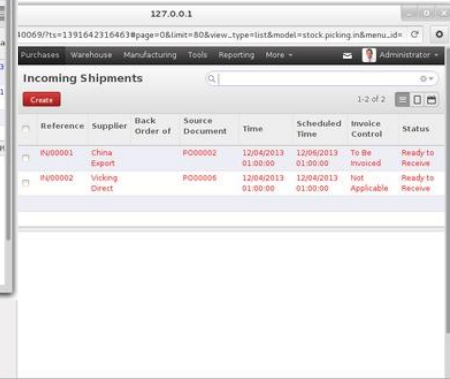
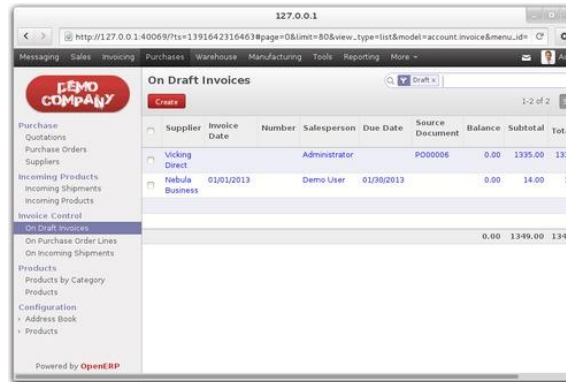
Product Name	Price	On hand	Forecasted
[ADPT] USB Adapter	18.00	0.000 Unit(s)	5.000 Unit(s)
[CARD] Graphics Card	885.00	16.000 Unit(s)	16.000 Unit(s)
[C-Case] Computer Case	25.00	0.000 Unit(s)	0.000 Unit(s)
[CD] Blank CD	20.00	0.000 Dozen(s)	0.000 Dozen(s)
[CPUa8] Processor AMD 8-Core	1980.00	0.000 Unit(s)	15.000 Unit(s)
[CPUi5] Processor Core i5 2.70 Ghz	2100.00	0.000 Unit(s)	0.000 Unit(s)
[DC] Datacard	40.00	0.000 Unit(s)	50.000 Unit(s)
[DVD] Blank DVD-RW	24.00	0.000 Dozen(s)	25.000 Dozen(s)

Customers View:

Customer Name	Role	Location	Contact Info
Agrolait	Components Buyer	Wavre, Belgium	info@agrolait.com
Arthur Gomez	Software Developer at Spark Systems	São Paulo, Brazil	
Ayaan Agarwal	Director at Best Designers	Mumbai, India	info@bestdesigners.in
Benjamin Flores	Business Executive at Nebula Business	Rosario, Argentina	ben@nebula.ar
Angel Cook	General Manager at Chamber Works	Detroit, United States	angel.cook@chamberworks.com
Axelar	Partner / Gold Services	Champs sur Marne, France	info@axelar.com
Bank Wealthy and sons	Consultancy Services / Partner / Gold	Birmingham, United Kingdom	email@wealthyandsons.com
Best Designers	Partner / Bronze / Partner / IT Services	Mumbai, India	info@bestdesigners.in

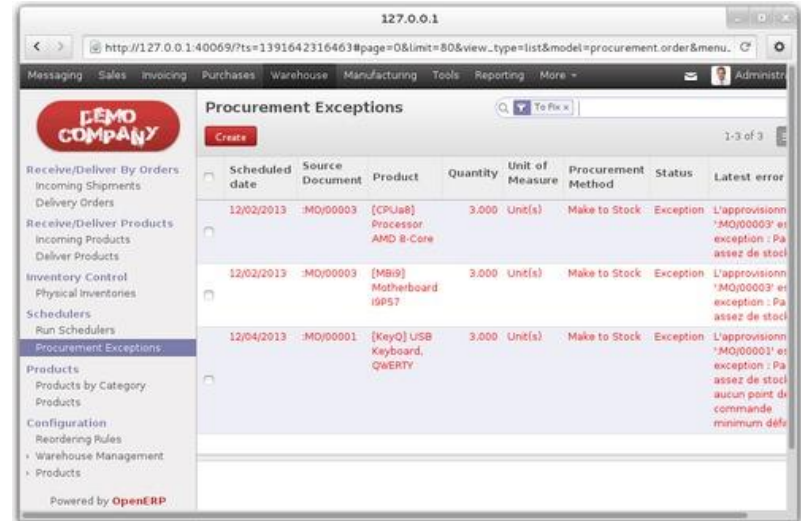
Purchases

- Quotations
- Purchases orders
- On Draft Invoices
- Incoming shipments



Warehouse

- Procurement Exceptions
- Incoming Shipments
- Incoming Products
- ...



127.0.0.1

http://127.0.0.1:40069/?ts=1391642316463#page=0&limit=80&view_type=list&model=procurement.order&menu_id=...

DEMOCOMPANY

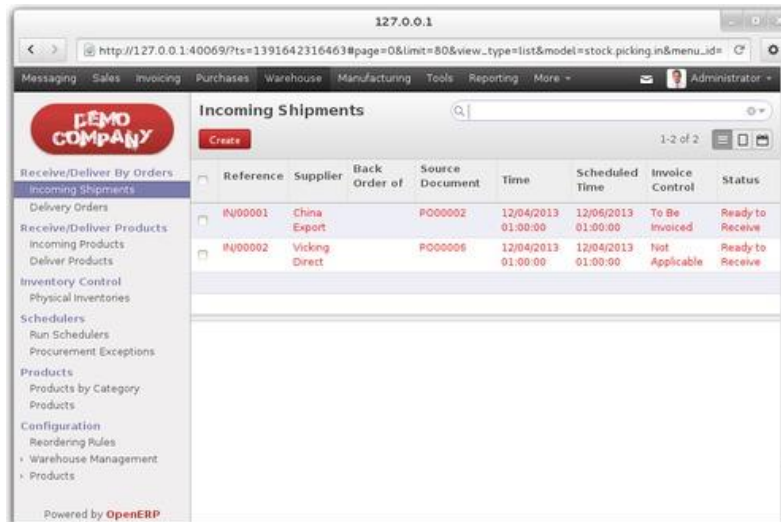
Procurement Exceptions

Create

1-3 of 3

Scheduled date	Source Document	Product	Quantity	Unit of Measure	Procurement Method	Status	Latest error
12/02/2013	MO/00003	[CPUa8] Processor AMD B-Core	3,000	Unit(s)	Make to Stock	Exception	L'approvisionnement 'MO/00003' est exception ; Pas assez de stock
12/02/2013	MO/00003	[MBi9] Motherboard i9P57	3,000	Unit(s)	Make to Stock	Exception	L'approvisionnement 'MO/00003' est exception ; Pas assez de stock
12/04/2013	MO/00001	[KeyQ] USB Keyboard, QWERTY	3,000	Unit(s)	Make to Stock	Exception	L'approvisionnement 'MO/00001' est exception ; Pas assez de stock aucun point de commande minimum défini

Powered by OpenERP



127.0.0.1

http://127.0.0.1:40069/?ts=1391642316463#page=0&limit=80&view_type=list&model=stock.picking.in&menu_id=...

DEMOCOMPANY

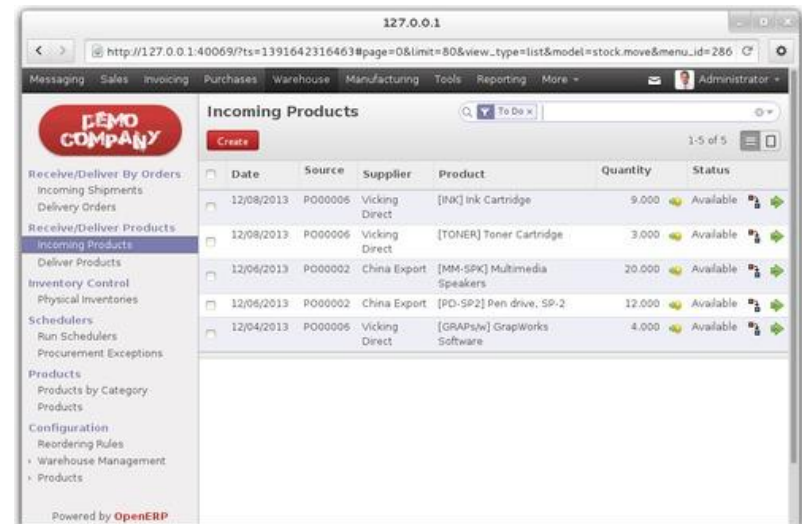
Incoming Shipments

Create

1-2 of 2

Reference	Supplier	Back Order of	Source Document	Time	Scheduled Time	Invoice Control	Status
IN/00001	China Export		PO00002	12/04/2013 01:00:00	12/06/2013 01:00:00	To Be Invoiced	Ready to Receive
IN/00002	Vicking Direct		PO00006	12/04/2013 01:00:00	12/04/2013 01:00:00	Not Applicable	Ready to Receive

Powered by OpenERP



127.0.0.1

http://127.0.0.1:40069/?ts=1391642316463#page=0&limit=80&view_type=list&model=stock.move&menu_id=286

DEMOCOMPANY

Incoming Products

Create

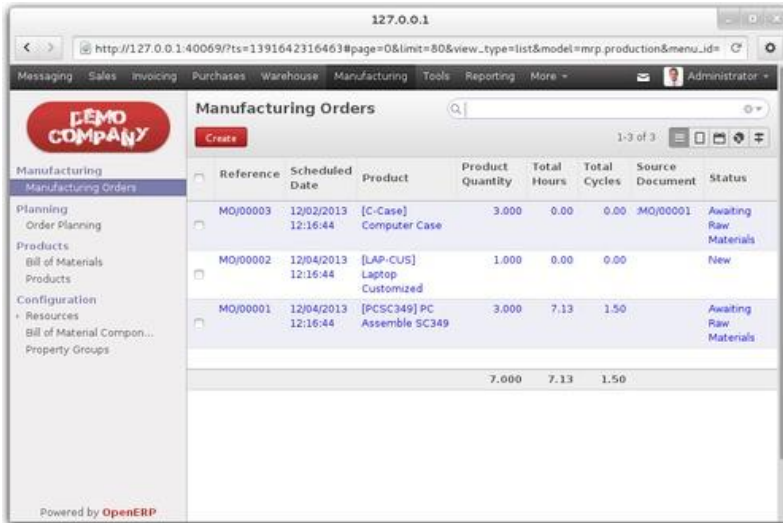
1-5 of 5

Date	Source	Supplier	Product	Quantity	Status
12/08/2013	PO00006	Vicking Direct	[INK] Ink Cartridge	9,000	Available
12/08/2013	PO00006	Vicking Direct	[TONER] Toner Cartridge	3,000	Available
12/06/2013	PO00002	China Export	[MM-SPK] Multimedia Speakers	20,000	Available
12/06/2013	PO00002	China Export	[PD-SP2] Pen drive, SP-2	12,000	Available
12/04/2013	PO00006	Vicking Direct	[GRAPs/w] GrapWorks Software	4,000	Available

Powered by OpenERP

Manufacturing

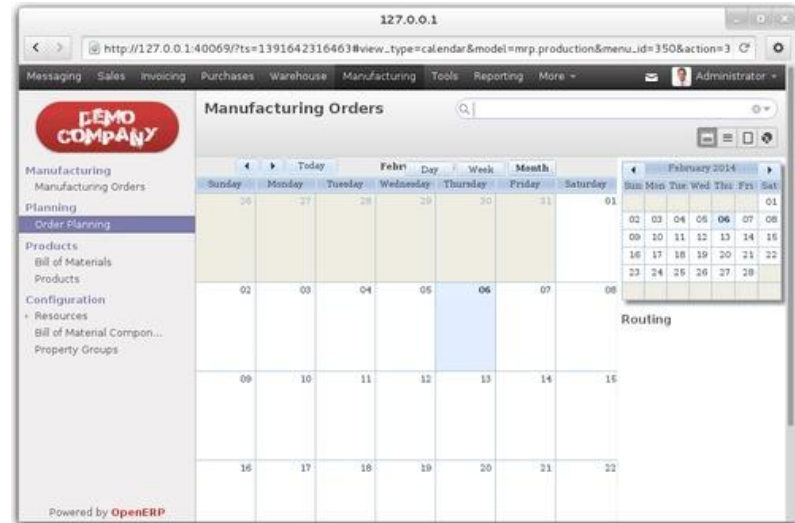
- Manufacturing orders



The screenshot shows the 'Manufacturing Orders' list view in OpenERP. The interface includes a navigation menu on the left with options like 'Manufacturing Orders', 'Products', and 'Configuration'. The main area displays a table with columns for Reference, Scheduled Date, Product, Product Quantity, Total Hours, Total Cycles, Source Document, and Status. Three orders are listed, including one for 'Computer Case' and another for 'Laptop Customized'. A summary row at the bottom shows totals for Product Quantity (7.000), Total Hours (7.13), and Total Cycles (1.50).

Reference	Scheduled Date	Product	Product Quantity	Total Hours	Total Cycles	Source Document	Status
MOJ00003	12/02/2013 12:16:44	[C-Case] Computer Case	3.000	0.00	0.00	.MOJ00001	Awaiting Raw Materials
MOJ00002	12/04/2013 12:16:44	[LAP-CUS] Laptop Customized	1.000	0.00	0.00		New
MOJ00001	12/04/2013 12:16:44	[PCSC349] PC Assemble SC349	3.000	7.13	1.50		Awaiting Raw Materials
			7.000	7.13	1.50		

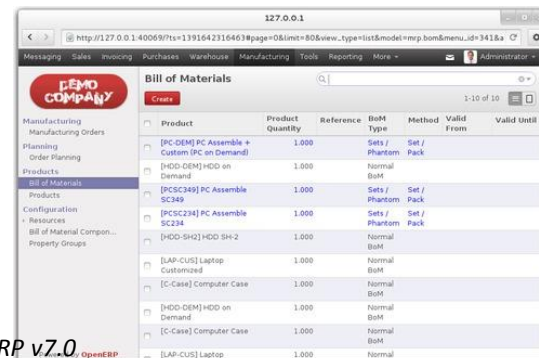
(List view)



The screenshot shows the 'Manufacturing Orders' calendar view in OpenERP. The interface features a navigation menu on the left and a main area with a calendar grid for February 2014. The grid shows dates from Sunday to Saturday, with some cells containing numbers representing orders. A 'Routing' section is visible on the right side of the calendar.

(Calendar view)

- Bill of materials

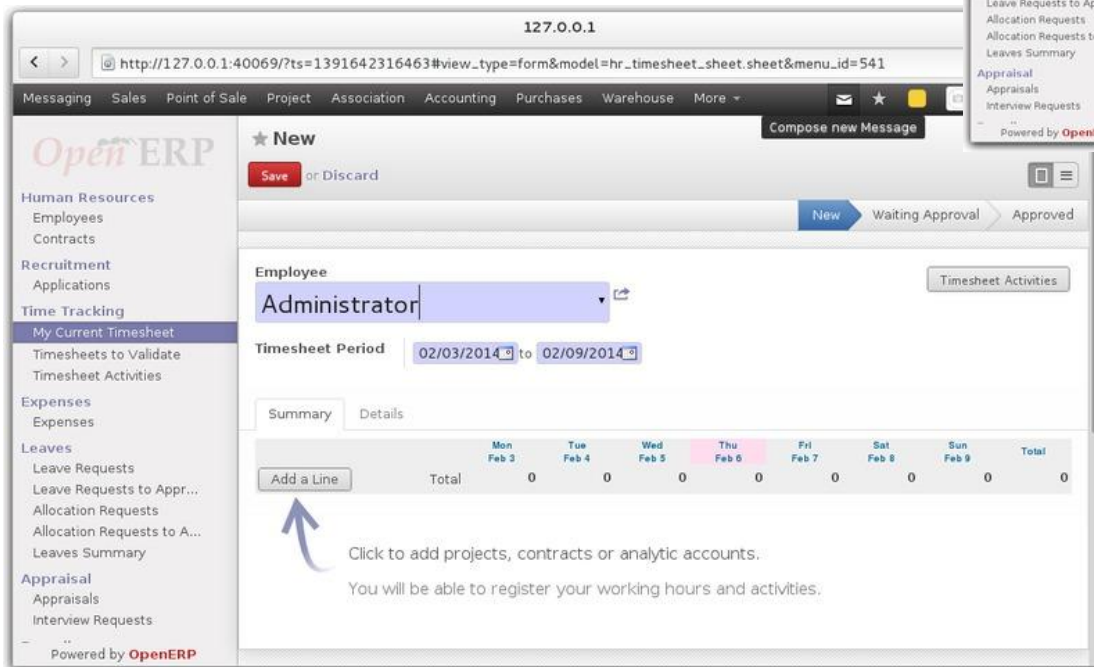


The screenshot shows the 'Bill of Materials' view in OpenERP. The interface includes a navigation menu on the left and a main area with a table listing components. The table has columns for Product, Product Quantity, Reference, BoM Type, Method, Valid From, and Valid Until. Components include 'PC Assemble + Custom (PC on Demand)', 'HDD (DEM) HDD on Demand', 'PC Assemble SC349', 'PC Assemble SC344', 'HDD (SH2) HDD SH-2', 'LAP-CUS Laptop Customized', 'C-Case Computer Case', 'HDD (DEM) HDD on Demand', 'C-Case Computer Case', and 'LAP-CUS Laptop'.

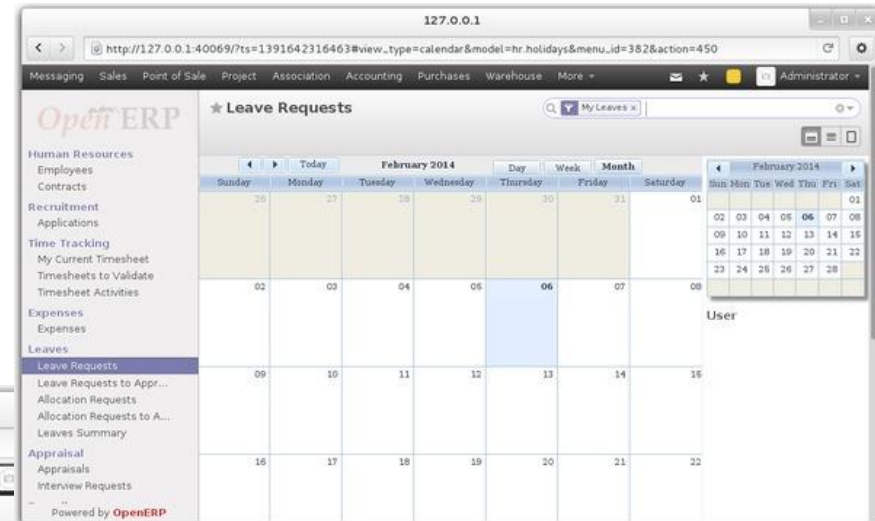
Product	Product Quantity	Reference	BoM Type	Method	Valid From	Valid Until
[PC-DEM] PC Assemble + Custom (PC on Demand)	1.000		Sets / Phantom	Set / Pack		
[HDD-DEM] HDD on Demand	1.000		Normal BoM			
[PCSC349] PC Assemble SC349	1.000		Sets / Phantom	Set / Pack		
[PCSC234] PC Assemble SC344	1.000		Sets / Phantom	Set / Pack		
[HDD-SH2] HDD SH-2	1.000		Normal BoM			
[LAP-CUS] Laptop Customized	1.000		Normal BoM			
[C-Case] Computer Case	1.000		Normal BoM			
[HDD-DEM] HDD on Demand	1.000		Normal BoM			
[C-Case] Computer Case	1.000		Normal BoM			
[LAP-CUS] Laptop	1.000		Normal BoM			

Human Resources

- Timesheets
- Leave Requests
- ...



This screenshot shows the 'New' form for creating a timesheet in OpenERP. The form is titled 'New' and includes a 'Save or Discard' button. The 'Employee' field is set to 'Administrator'. The 'Timesheet Period' is set from '02/03/2014' to '02/09/2014'. Below the form, there is a table with columns for days of the week (Mon Feb 3, Tue Feb 4, Wed Feb 5, Thu Feb 6, Fri Feb 7, Sat Feb 8, Sun Feb 9) and a 'Total' column. All values in the table are currently zero. An 'Add a Line' button is located to the left of the table. A blue arrow points to this button with the text: 'Click to add projects, contracts or analytic accounts. You will be able to register your working hours and activities.'



This screenshot shows the 'Leave Requests' calendar view in OpenERP. The calendar displays the month of February 2014. The days are color-coded: yellow for days with no requests, light blue for days with one request, and dark blue for days with multiple requests. The 'Leave Requests' menu item is highlighted in the left sidebar. The calendar shows requests for the 6th, 13th, and 16th of February.

Point of sales

- Tactile interface

The screenshot displays the Open ERP Point of Sale interface. At the top, the header includes the 'Open ERP' logo, the user role 'Administrator', a home icon, a '+' button, the time '16:08', and a 'Fermer' button with a green indicator light. A search bar on the right contains the text 'Recherche d'articles'.

The main content area is divided into two sections. On the left, a box indicates 'Votre panier est vide' (Your basket is empty) with a total of '0.00 €' and taxes of '0.00 €'. Below this is a 'Payer' button and a numeric keypad with buttons for digits 1-9, 0, '+/-', and a clear button. The keypad also includes buttons for 'Qté', 'Rem.', and 'Prix'. At the bottom left, a dropdown menu for 'Sélectionnez un caissier' (Select a cashier) shows 'Cathy' as the selected option.

The right section features a grid of product categories and items. The categories are: 'Boissons' (Beverages), 'Alimentation' (Food), 'Fruits frais' (Fresh fruits), 'Légumes frais' (Fresh vegetables), and 'Autre' (Other). The items displayed include: 'Lays Pickles 250g' (1.48 €), 'Evian 2L' (1.26 €), 'Perrier 50cl' (0.71 €), 'Avancé' (150.00 €), 'Belle-Vue Kriek 25cl' (0.83 €), 'Raisins noirs' (4.80 €/Kg), 'Boni Oranges' (1.98 €/Kg), 'Boon Framboise 37.5cl' (2.53 €), and several other items with prices ranging from 0.34 € to 0.90 €/Kg.

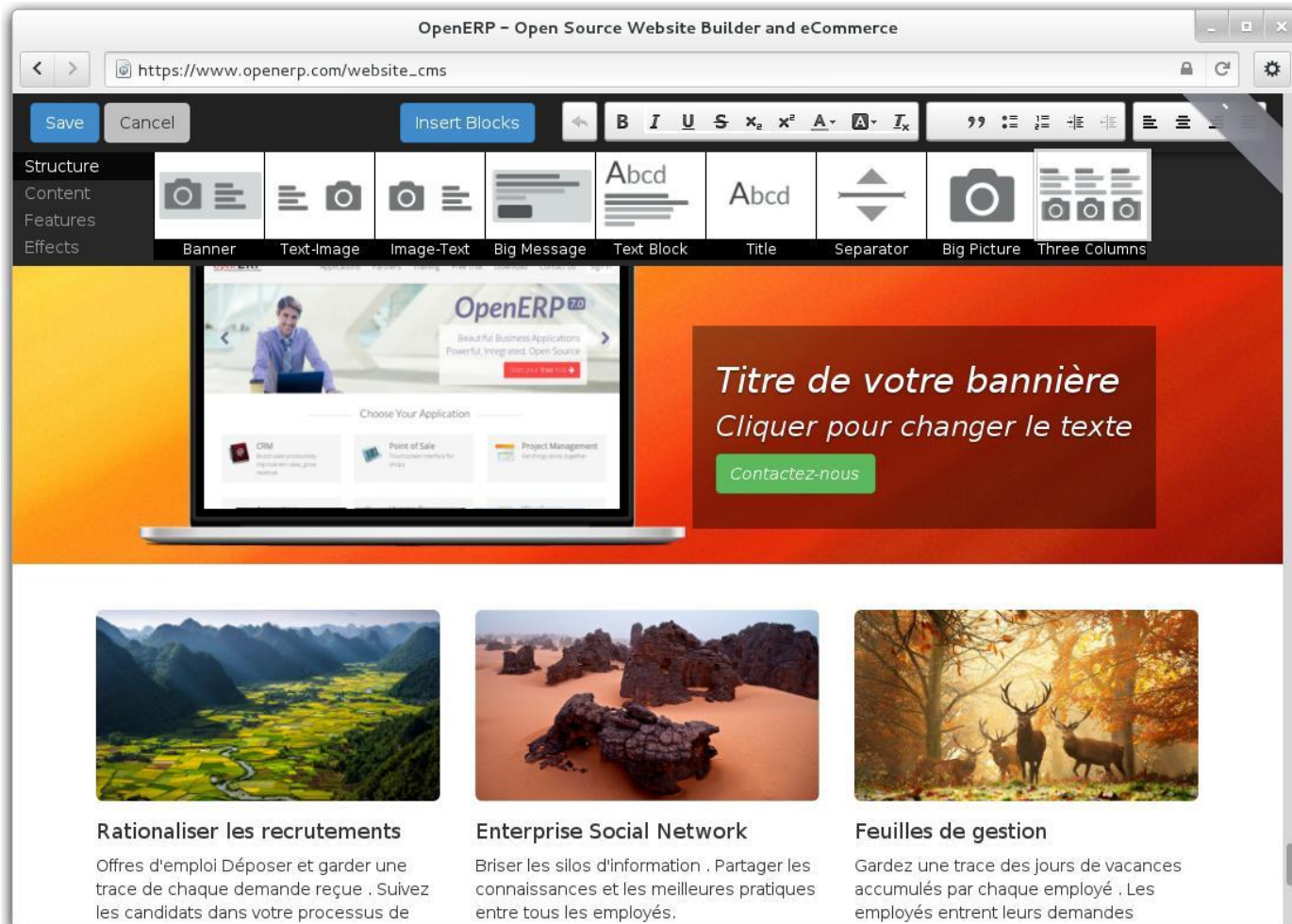
Point of sales

- **PosBox** www.indiegogo.com/projects/opensource-your-shop
 - Raspberry-Pi + Tablet + Printer + Scanner



CMS & E-commerce

- Available since **Odoo 8.0** + website builder



OpenERP – Open Source Website Builder and eCommerce

https://www.openerp.com/website_cms

Save Cancel Insert Blocks

Structure
Content
Features
Effects

Banner Text-Image Image-Text Big Message Text Block Title Separator Big Picture Three Columns

Titre de votre bannière
Cliquer pour changer le texte
Contactez-nous

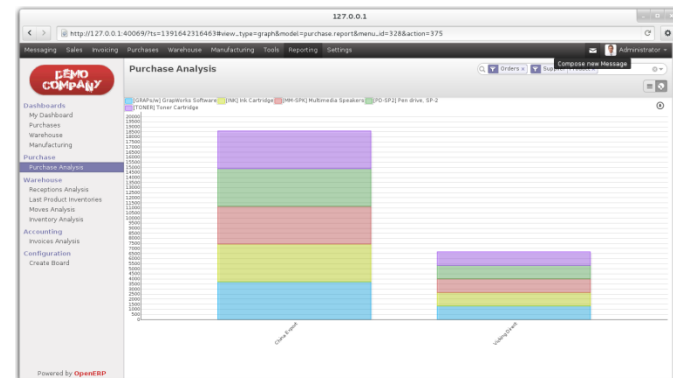
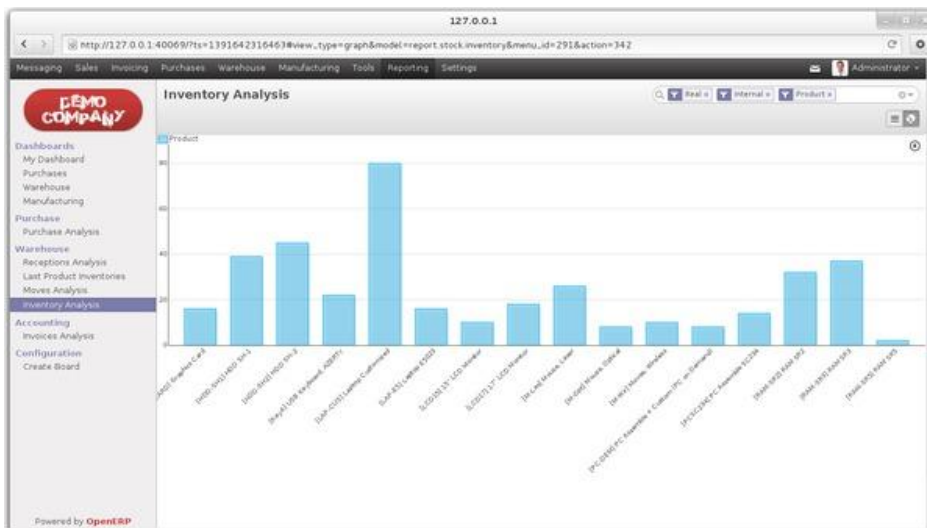
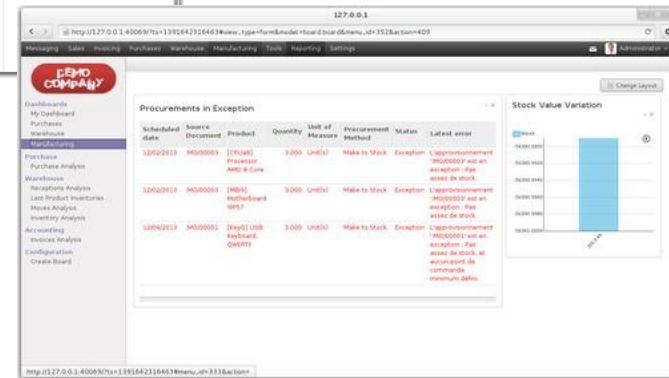
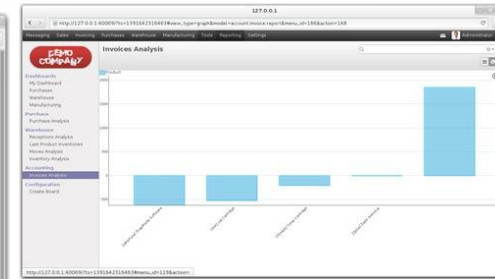
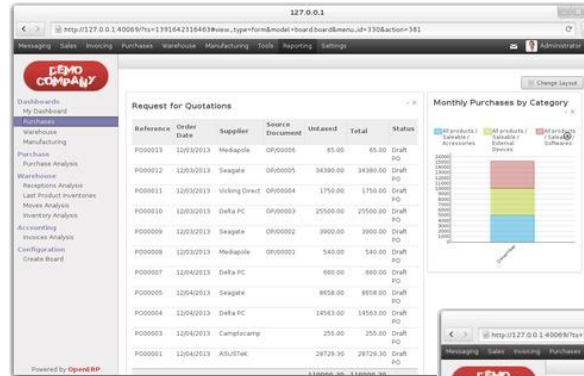
Rationaliser les recrutements
Offres d'emploi Déposer et garder une trace de chaque demande reçue . Suivez les candidats dans votre processus de

Enterprise Social Network
Briser les silos d'information . Partager les connaissances et les meilleures pratiques entre tous les employés.

Feuilles de gestion
Gardez une trace des jours de vacances accumulés par chaque employé . Les employés entrent leurs demandes

Reporting

- Invoice Analysis
- Purchase Analysis
- Inventory Analysis
- ...



Screenshots of OpenERP v7.0

Settings (Administrator)

- Install new module

The screenshot displays the OpenERP Administrator interface. The browser address bar shows the URL: `http://127.0.0.1:40069/?ts=1391642316463#view_type=kanban&model=ir.module.module&action=38`. The page title is "127.0.0.1". The navigation menu includes: Messaging, Sales, Point of Sale, Project, Association, Accounting, Purchases, Warehouse, Manufacturing, Marketing, and More. The main content area is titled "★ Apps / Not Installed Modules" and features a search bar and a "Compose new Message" button. A sidebar on the left lists various system components: Modules (Apps, Updates, Installed Modules, Not Installed Modules, Update Modules List, Apply Scheduled Upgra...), Configuration (Sales, Purchases, Warehouse, Manufacturing, Project, Accounting, Human Resources, Marketing, Knowledge, General Settings), Companies (Companies), Users (Groups, Users, OAuth Providers), Translations (Languages, Lead's Translation), and a footer note "Powered by OpenERP".

Module Name	Description	Status
CRM	Leads, Opportunities, Phone Calls <i>crm</i>	Installed
Social Network	Discussions, Mailing Lists, News <i>mail</i>	Installed
invoicing & Payments	Send Invoices and Track Payments <i>account_voucher</i>	Installed
Point of Sale	Touchscreen Interface for Shops <i>point_of_sale</i>	Installed
Project Management	Projects, Tasks <i>project</i>	Installed
Notes	Sticky notes, Collaborative, Memos <i>note</i>	Installed
Issue Tracker	Support, Bug Tracker, Helpdesk <i>project_issue</i>	Installed
Accounting and Finance	Financial and Analytic Accounting <i>account_accountant</i>	Installed
Sales Management	Quotations, Sales Orders, Invoicing <i>sale</i>	Installed
Warehouse Management	Inventory, Logistic, Storage <i>stock</i>	Installed
MRP	Manufacturing Orders, Bill of Materials, Routing <i>mrp</i>	Installed
Purchase Management	Purchase Orders, Receptions, Supplier Invoices <i>purchase</i>	Installed
Employee Directory	Jobs, Departments, Employees Details <i>hr</i>	Installed
Timesheets	Timesheets, Attendances, Activities <i>hr_timesheet_sheet</i>	Installed
Recruitment Process	Jobs, Recruitment, Applications	
Leave Management	Holidays, Allocation and Leave	
Expense Management	Expenses Validation, Invoicing	

Settings (Administrator)

- Parameter workflows

The screenshot displays the OpenERP administrator interface. The top navigation bar includes modules like Messaging, Sales, Point of Sale, Project, Association, Accounting, Purchases, Warehouse, Manufacturing, Marketing, and More. The left sidebar lists various configuration options under 'Configuration' and 'Companies'. The main content area is titled 'Workflows' and shows a list of workflow configurations.

<input type="checkbox"/>	Name	Resource Object	On Create
<input type="checkbox"/>	account.invoice.basic	account.invoice	<input checked="" type="checkbox"/>
<input type="checkbox"/>	account.voucher.basic	account.voucher	<input checked="" type="checkbox"/>
<input type="checkbox"/>	hr.payslip.basic	hr.payslip	<input checked="" type="checkbox"/>
<input type="checkbox"/>	hr_timesheet_sheet.sheet	hr_timesheet_sheet.sheet	<input checked="" type="checkbox"/>
<input type="checkbox"/>	hr.wkf.expenses	hr.expense.expense	<input checked="" type="checkbox"/>
<input type="checkbox"/>	hr.wkf.holidays		
<input type="checkbox"/>	idea.wkf		
<input type="checkbox"/>	marketing.campaign.basic		
<input type="checkbox"/>	marketing.campaign.segment.basic		
<input type="checkbox"/>	mrp.production.basic		
<input type="checkbox"/>	mrp.production.workcenter.line.basic		
<input type="checkbox"/>	mrp.repair.basic		
<input type="checkbox"/>	occasions.wkf		
<input type="checkbox"/>	Payment Order Workflow		
<input type="checkbox"/>	POS Session Workflow		
<input type="checkbox"/>	Pos workflow		
<input type="checkbox"/>	procurement.order.basic		
<input type="checkbox"/>	project.phase.wkf		
<input type="checkbox"/>	Purchase Order Basic Workflow		
<input type="checkbox"/>	sale.order.basic		

The bottom window shows a detailed view of the 'account_invoice.basic' workflow diagram. It features a flowchart with nodes: 'draft', 'proforma2', 'open', 'paid', 're-open', and 'cancel'. Transitions between nodes are labeled with conditions such as 'invoice_open=True', 'invoice_cancel=True', and 'open_test=True'.

Settings

(Administrator)

- Example: workflow of an idea

The screenshot shows the OpenERP administrator interface. The browser address bar displays the URL: `http://127.0.0.1:40069/?ts=1391642316463#id=1&view_type=form&model=workflow&menu_id=54&action=32`. The page title is "Workflows / idea.wkf". The interface includes a navigation menu on the left with categories: Modules, Configuration, Companies, Users, and Translations. The main content area shows the workflow configuration for "idea.wkf" with a "Resource Object" of "idea.idea". A table lists the workflow steps:

Name	Workflow	Kind	Flow Start	Flow Stop
cancel	idea.wkf	Function	<input type="checkbox"/>	<input checked="" type="checkbox"/>
close	idea.wkf	Function	<input type="checkbox"/>	<input checked="" type="checkbox"/>
draft	idea.wkf	Function	<input checked="" type="checkbox"/>	<input type="checkbox"/>
open	idea.wkf	Function	<input type="checkbox"/>	<input type="checkbox"/>

At the bottom of the page, it says "Powered by OpenERP".

Settings

(Administrator)

- Example: workflow of an idea

The screenshot displays the OpenERP administrator interface for configuring a workflow. The browser address bar shows the URL: `http://127.0.0.1:40069/?ts=1391642316463#id=1&view_type=diagram&model=workflow&menu_id=54&action=32`. The page title is "Workflows" and the specific workflow being edited is "idea.wkf". A "New Node" button is visible. A warning message states: "When customizing a workflow, be sure you do not modify an existing node or arrow, but rather add new nodes or arrows. If you absolutely need to modify a node or arrow, you can only change fields that are empty or set to the default value. If you don't do that, your customization will be overwritten at the next update or upgrade to a future version of OpenERP."

The workflow diagram consists of four nodes: "draft", "open", "close", and "cancel". The transitions are as follows:

- "draft" to "open" with the label "idea_open True".
- "open" to "close" with the label "idea_close True".
- "open" to "cancel" with the label "idea_cancel True".

The interface includes a sidebar with navigation menus for Modules, Configuration, Companies, Users, and Translations. The footer indicates "Powered by OpenERP".

Getting sources

- Launchpad (Sourceforge, GNU Savannah, ...)



- `https://launchpad.net/openobject`
- Version under developement (trunk)
 - `https://launchpad.net/openobject-server/trunk`
 - `https://launchpad.net/openobject-addons/trunk`
 - `https://launchpad.net/openerp-web/trunk`
- Older versions
 - `https://launchpad.net/openobject-server/7.0`
 - `https://launchpad.net/openobject-addons/7.0`
 - `https://launchpad.net/openerp-web/7.0`

Download sources

- **Bazaar (cvs, svn, git, ...)**



- `bzr branch lp:openobject-server`
- `bzr branch lp:openobject-addons`
- `bzr branch lp:openerp-web`

Remove `.bzr` hidden directories!

- **Older versions**

- `bzr branch lp:openobject-server/7.0`
- `bzr branch lp:openobject-addons/7.0`
- `bzr branch lp:openerp-web/7.0`

- `bzr branch lp:openobject-server/6.0`
- `bzr branch lp:openobject-addons/6.0`
- `bzr branch lp:openerp-web/6.0`

Verticalizations

- **Hostels**
 - <https://launchpad.net/~hotel-core-editors>
- **Electronic Medical Record and Hospital Information System**
 - <https://launchpad.net/medical>
- **Construction**
 - <https://launchpad.net/openerp-btp>
 - <https://launchpad.net/openerp-construction>
- **Pedagogy / Education Management System**
 - <https://launchpad.net/pedagogy>
 - <https://launchpad.net/openerp-ems>

Verticalizations

- **Textile & Spinning Manufacturing Industry**
 - <https://launchpad.net/openerp-textiles>
- **Humanitarian non-governmental organization**
 - <https://launchpad.net/openerp-humanitarian-ngo>
- **Library Management System**
 - <https://launchpad.net/openerp-library>
- **Veterinary and production farms**
 - <https://launchpad.net/openerp-openvet>
- **Agro farm management**
 - <https://launchpad.net/agrooffice>

OUTLINE

1. Overview

- Birth and growth
- Customers, Partners, Vendors

2. Administration

- Versions, Architecture
- Main modules

3. Development

- Module programming
- Webservice programming

4. Documentation

Python

- Language used to program OpenERP modules
- Don't need to become an expert
- Short 12-slides-introduction
 - Indentation
 - Functions
 - Data structures
 - Lists, Tupples, Dictionnaires
 - Classes



3. DEVELOPMENT

Open ERP models

Module description

Fields of the model (columns)

Views (in XML)

Webservice communication

OpenERP database structure

127.0.0.1

http://127.0.0.1:40069/#page=0&limit=80&view_type=list&model=ir.model&menu_id=37&action=16

Messaging Sales Invoicing Purchases Warehouse Manufacturing Tools Reporting Settings Administrator

DEMO COMPANY

Models 1-22 of 22

Create

<input type="checkbox"/>	Model	Model Description	Type	Transient Model
<input type="checkbox"/>	res.bank	Bank	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.company	Companies	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.config	res.config	Base Object	<input checked="" type="checkbox"/>
<input type="checkbox"/>	res.config.installer	res.config.installer	Base Object	<input checked="" type="checkbox"/>
<input type="checkbox"/>	res.config.settings	res.config.settings	Base Object	<input checked="" type="checkbox"/>
<input type="checkbox"/>	res.country	Country	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.country.state	Country state	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.currency	Currency	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.currency.rate	Currency Rate	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.currency.rate.type	Currency Rate Type	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.groups	Access Groups	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.lang	Languages	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.partner	Partner	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.partner.bank	Bank Accounts	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.partner.bank.type	Bank Account Type	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.partner.bank.type.field	Bank type fields	Base Object	<input type="checkbox"/>
<input type="checkbox"/>	res.partner.category	Partner Categories	Base Object	<input type="checkbox"/>

Modules
Apps
Updates
Installed Modules
Update Modules List
Apply Scheduled Upgra...

Configuration
Purchases
Warehouse
Manufacturing
Invoicing
General Settings

Companies
Companies

Users
Groups
Users

Translations
Languages
Load a Translation
Import / Export
Application Terms

Powered by **OpenERP**

OpenERP database structure

- Some of the numerous predefined models:
 - `base.module.configuration`
 - `base.module.import`
 - `base.module.update`
 - `res.users`
 - `res.partners`
 - `res.company`
 - `res.country`
 - `res.lang`
 - `res.currency`
 - `workflow`
 - `workflow.instance`
 - `workflow.triggers`
 - `workflow.activity`
 - `ir.ui.view`
 - `ir.ui.menu`
 - `ir.filters`
 - `ir.translation`
 - `ir.cron`

res.users

- Fields of the predefined model **res.users** :

Field name

- id
- login
- login_date
- password
- new_password
- active
- user_email
- signature
- company_id
- company_ids
- groups_id
- partner_id
- action_id
- menu_id

Field type

- integer
- char
- date
- char
- char
- boolean
- char
- text
- many2one
- many2many
- many2many
- many2one
- many2one
- many2one

Used in the modules:

auth_crypt,
auth_ldap,
auth_oauth,
auth_oauth_signup,
auth_openid,
auth_signup,
base,
hr,
mail,
note,
point_of_sale,
sale_crm,
share,
survey,
...

product.product

- Some fields of the model **product.product** :

Field name

- code
- color
- price
- active
- image
- track_incoming
- track_outgoing
- track_production
- seller_id
- seller_delay
- seller_qty
- warehouse_id
- qty_available
- ...

Field type

- char
- integer
- float
- boolean
- binary
- boolean
- boolean
- boolean
- many2one
- integer
- float
- many2one
- float

Used in the modules:

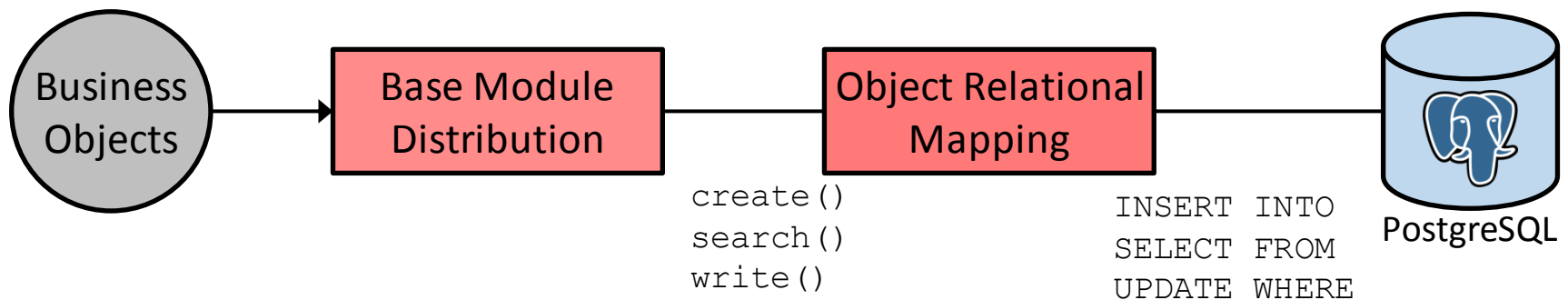
event_sale,
mrp,
point_of_sale,
procurement,
purchase_requisition,
stock,
stock_location,
warning,
...

A module

- A module defines a namespace
 - Several Python classes can be defined in the same namespace
- Each class defines a « model » / « business object », and fields are stored in PostgreSQL
 - Thanks to the ORM
 - All classes must inherit the class `osv.osv`
 - osv = Object Service
- Models can be shared by different modules

ORM

- **Object Relational Mapping**



- Technique to program with a relational DB as if it was an object-oriented one
- Makes the bridge between relational and OO worlds
- Centralizes data integrity verifications and access rights management

Methods of the ORM

- **create**
- **search**
- **read**
- **browse**
- **write**
- **unlink**
- **copy**
- copy_data
- default_get
- fields_get
- name_get
- export_data
- import_data
- view_init
- check_access_rule
- get_xml_id
- name_search
- perm_read
- read_group
- fields_view_get

– Can be called from business objects

– Can also be called through WebServices

The create() method

create(cr, user, vals, context=None)

Create new record with specified value

- Parameters:**
- **cr** – database cursor
 - **user** (integer) – current user id
 - **vals** (dictionary) – field values for new record, e.g. {'field_name': field_value, ...}
 - **context** (dictionary) – (optional) context arguments, e.g. {'lang': 'en_us', 'tz': 'UTC', ...}

Returns: id of new record created

- Raises:**
- **AccessError**
 - if user has no create rights on the requested object
 - if user tries to bypass access rules for read on the requested object.
 - **ValidateError** – if user tries to enter invalid value for a field that is not in selection
 - **UserError** – if a loop would be created in a hierarchy of objects a result of the operation (such as setting an object as its own parent)

The search() method

```
search(cr, user, args, offset=0, limit=None, order=None, context=None, count=False)
```

Search for records based on a search domain

- Parameters:**
- **cr** – database cursor
 - **user** (integer) – current user id
 - **args** – list of tuples specifying the search domain [('field_name', 'operator', value), ...]. Pass an empty list to match all records.
 - **offset** – (optional) number of results to skip in the returned values (default: 0)
 - **limit** – (optional) max number of records to return (default: None)
 - **order** – (optional) columns to sort by (default: self._order=id)
 - **context** (dictionary) – (optional) context arguments, e.g. {'lang': 'en_us', 'tz': 'UTC', ...}
 - **count** – (optional) if True, returns only the number of records matching the criteria, not their ids

Returns: id or list of ids of records matching the criteria

Return type: integer or list of integers

- Raises:**
- **AccessError**
 - if user tries to bypass access rules for read on the requested object.

The `browse()` method

```
browse(cr, user, select, context=None, list_class=None,  
fields_process=None)
```

Fetch records as objects allowing to use dot notation to browse fields and relations

Parameters:

- **cr** – database cursor
- **user** (integer) – current user id
- **select** – id or list of ids
- **context** (dictionary) – (optional) context arguments, e.g. {'lang': 'en_us', 'tz': 'UTC', ...}

Returns: object or list of objects requested

The write() method

write(cr, user, ids, vals, context=None)

Update records with given ids with the given field values

Parameters:

- **cr** – database cursor
- **user** (integer) – current user id
- **ids** – object id or list of object ids to update according to vals
- **vals** (dictionary) – field values to update, e.g. {'field_name': new_field_value, ...}
- **context** (dictionary) – (optional) context arguments, e.g. {'lang': 'en_us', 'tz': 'UTC', ...}

Returns: True

Raises:

- **AccessError**
 - if user has no write rights on the requested object
 - if user tries to bypass access rules for write on the requested object
- **ValidateError** – if user tries to enter invalid value for a field that is not in selection
- **UserError** – if a loop would be created in a hierarchy of objects a result of the operation (such as setting an object as its own parent)

The unlink() method

`unlink(cr, user, ids, context=None)`

Delete records with given ids

Parameters:

- **cr** – database cursor
- **user** (integer) – current user id
- **ids** – object id or list of object ids to update according to vals
- **context** (dictionary) – (optional) context arguments, e.g. {'lang': 'en_us', 'tz': 'UTC', ...}

Returns: True

Raises:

- **AccessError**
 - if user has no write rights on the requested object
 - if user tries to bypass access rules for write on the requested object
- **UserError** – if a loop would be created in a hierarchy of objects a result of the operation (such as setting an object as its own parent)

3. DEVELOPMENT

Open ERP models

Module description

Fields of the model (columns)

Views (in XML)

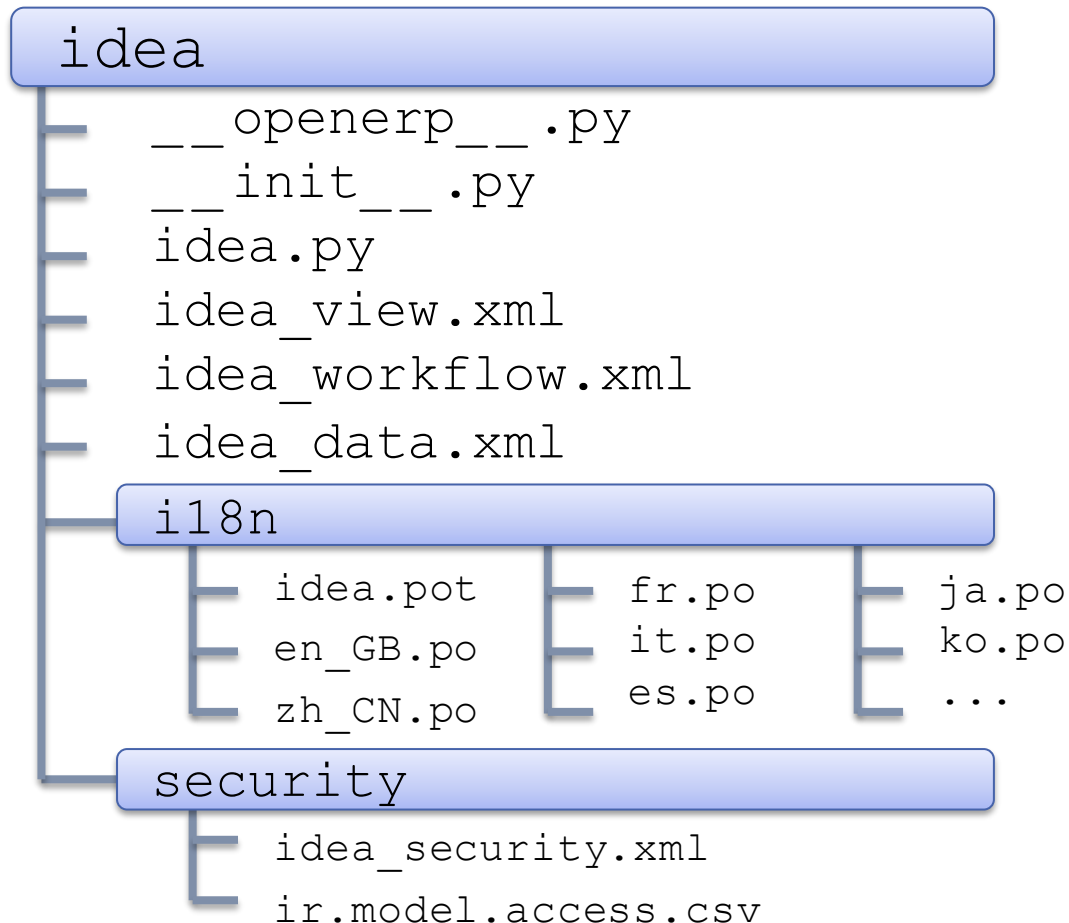
Webservice communication

Writing a module

- A module is composed of several files
 - Python files
 - Meta-informations about the module
 - Model for the ORM (Object-Relational Mapping)
 - XML files
 - Description of the views
 - Description of the workflows
 - Definition of demonstration data

Example of the module Idea

- Each module is contained in one directory



Python class of a model

• RESERVED MEMBERS



- `_name`: business object name `'module_namespace.object_name'`
- `_columns`: the fields of the model
- `_defaults`: default values for fields
- `_inherit`: `_name` of the parent
- `_sql_constraints`: tuples defining SQL constraints
 - (name, sql_def, message)
- `_constraints`: tuples defining Python constraints
 - (func_name, message, fields)
- `_order`: the field to sort records (default field: `'id'`)
- `_auto`, `_inherits`, `_log_access`, `_rec_name`,
`_sql`, `_table`

Python class of a model

- Example of members declaration

```
class idea_idea(osv.osv):  
    """ Idea """  
    _name = 'idea.idea'  
    _inherit = ['mail.thread']  
    _columns = {  
        'name': fields.char('Idea Summary', size=64, required=True,  
        'description': fields.text('Description', help='Content of  
        'state': fields.selection([('draft', 'New'),('open', 'Accep  
    }  
    _sql_constraints = [  
        ('name', 'unique(name)', 'Idea name must be unique')  
    ]  
    _defaults = {  
        'state': lambda *a: 'draft',  
    }  
    _order = 'name asc'
```


3. DEVELOPMENT

Open ERP models

Module description

Fields of the model (columns)

Views (in XML)

Webservice communication



Fields of the model

- Declared as a dictionary in the `_columns` member of the class
- Example

```
class idea_idea(osv.osv):
    _columns = {
        'create_uid': fields.many2one('res.users', 'Creator', requ
        'name': fields.char('Idea Summary', size=64, required=True
        'description': fields.text('Description', help='Content of
        'category_ids': fields.many2many('idea.category', string='
        'state': fields.selection([('draft', 'New'),
            ('open', 'Accepted'),
            ('cancel', 'Refused'),
            ('close', 'Done')]),
            'Status', readonly=True, track_visibility='onchange',
    )
}
```

Columns (fields)

- **RESERVED COLUMNS NAMES (i.e. FIELDS)**

- `id`: unique identifier for the object (created by ORM, don't add it!) 
- `name`: defines the label to display in lists, etc.
- `state`: defines object life-cycle stages (used for workflow)
- `active`: defines visibility
- `sequence`: defines order
- `create_date`, `create_uid`, `write_date`, `write_uid`:
used to log meta information about the record
(automatically defined and set by the ORM, so don't define this column!) 
- `parent_id`, `parent_left`, `parent_right`: defines tree structure on records

Type of fields

- Basic types
 - integer
 - float
 - boolean
 - char
 - text
 - selection
 - date
 - datetime
 - time
 - binary
- Relational types
 - many2one
 - one2many
 - many2many
 - ~~one2one~~ → many2one

Fields declaration

- Optional parameters
 - `string` : the field label (**required**)
 - `required` : set it to *True* if mandatory field
 - `readonly` : set it to *True* if not editable field
 - `help` : text for the help tooltip
 - `context` : dictionary with contextual parameters (for relational fields)
 - `states` : dynamically changes the field's attributes according to the changing state of the object

integer type

```
fields.integer(string='Field Name' [, Opt. Param.] ),
```

string parameter, which defines the label

```
fields.integer('Field Name' [, Optional Parameters] ),
```

- Example of declaration

```
_columns = {  
  'level': fields.integer('Level of development'),  
  'code': fields.integer('Reference code', readonly=True),  
}
```

- Example of assignment

```
self.write(cr, uid, ids, {'level': 5})
```

float type

```
fields.float('Field Name' [, Optional Parameters]),
```

- Example of declaration

```
_columns = {  
    'weight': fields.float('Product weight in kg'),  
    'price': fields.float('Price in EUR', digits=(12,6)),  
    'rate': fields.float('FRS-EUR exchange rate', digits=(12,6),  
                          readonly=True),  
}
```

- Example of assignment

```
self.write(cr, uid, ids, {'weight': 0.256})
```


boolean type

```
fields.boolean('Field Name' [, Optional Parameters]),
```

- Example of declaration

```
_columns = {  
    'is_active': fields.boolean('Active product'),  
    'msg_unread': fields.boolean('Unread message', readonly=True),  
}
```

- Example of assignment

```
self.write(cr, uid, ids, {'msg_unread': False})
```

char type

- A string of limited length

```
fields.char('Field Name', size=n [, Optional Param.] ),
```

- Example of declaration

```
_columns = {  
    'name': fields.char('Category Name', size=64, required=True),  
    'code': fields.char('Reference code', size=9, readonly=True),  
}
```

- Example of assignment

```
self.write(cr, uid, ids, {'code': '000AR400M'})
```

text type

- A string with no limited length

```
fields.text('Field Name', [, Optional Parameters]),
```

- Example of declaration

```
_columns = {  
  'description': fields.text('Description'),  
  'purchase_msg': fields.text('Message for purchase order'),  
}
```

- Example of assignment

```
self.write(cr, uid, ids, {'description': 'This article...'})
```

selection type

- Selection between predefined values

```
fields.selection(values, 'Field Name', [, Opt.Param.]),
```

– where the parameter `values` is a list, or a tuple, of 2-tuples formed as `(key, value)`

- Example with a list of 2-tuples:

```
values = [('n', 'Unconfirmed'), ('c', 'Confirmed')]
```

- Example with a tuple of 2-tuples:

```
values = (('n', 'Unconfirmed'), ('c', 'Confirmed'))
```

selection type

- Examples of declaration

```
_columns = {  
  'size': fields.selection(  
    [('-1', 'Undefined'), ('0', 'Small'),  
     ('1', 'Medium'), ('2', 'Large')],  
    'Package size'),  
}
```

```
countries = [('fr', 'France'), ('es', 'Spain'), ('it', 'Italy')]  
  
_columns = {  
  'dep': fields.selection(countries, 'Departure'),  
  'arr': fields.selection(countries, 'Arrival'),  
}
```

- Example of assignment

```
self.write(cr, uid, ids, {'size': '-1'})
```

date type

```
fields.date('Field Name' [, Optional Parameters]),
```

- Example of declaration

```
_columns = {  
    'birthday': fields.date('Date of birth'),  
    'last_mod': fields.date('Last modification', readonly=True),  
}
```

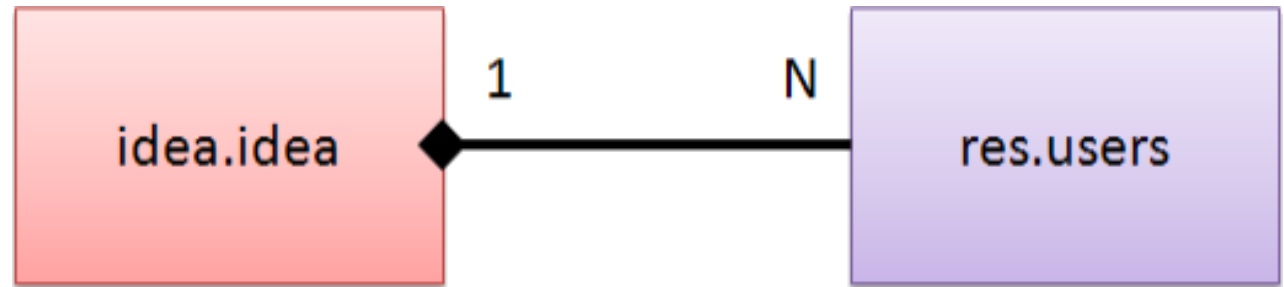
- Example of assignment

```
self.write(cr, uid, ids, {'last_mod': datetime.now()})
```

many2one relation

```
fields.many2one('model', 'Field Name' [, Opt.Param.] ),
```

- Example



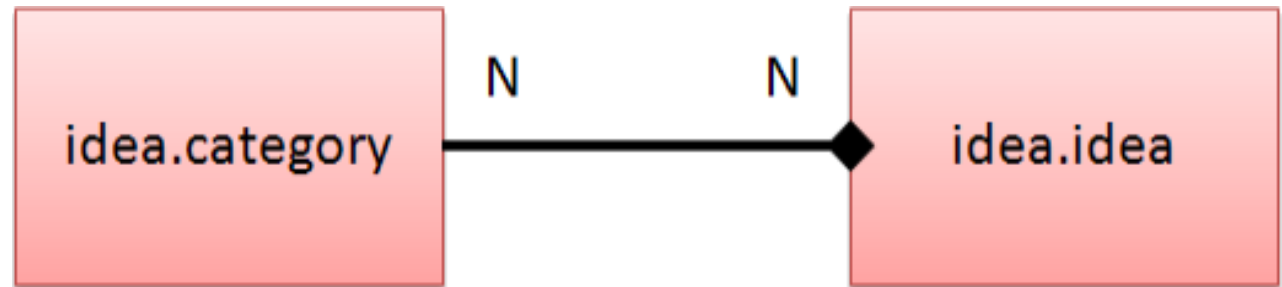
```
class idea_idea(osv.osv):  
    _columns = {  
        'create_uid': fields.many2one('res.users', 'Creator'),  
    }
```

- An idea is created by **1** (and only 1) user,
but a user can create **N** ideas

many2many relation

```
fields.many2many('model', string='Field Name' [, Par]),
```

- Example



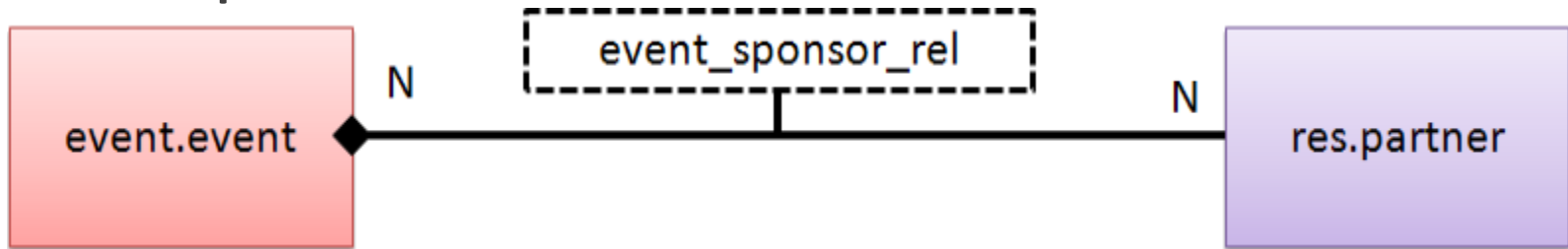
```
class idea_idea(osv.osv):  
    _columns = {  
        'category_ids': fields.many2many('idea.category',  
                                          string='Tags'),  
    }
```

- An idea can belong to **several** categories,
and a category can describe **several** ideas

many2many relation

```
fields.many2many('modelB', 'rel', 'idA', 'idB' [, Par]),
```

- Example

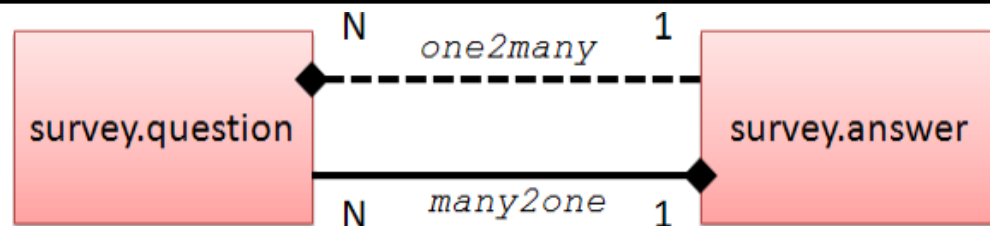


```
class event_event(osv.osv):  
    _columns = {  
        'sponsor_ids': fields.many2many('res.partner',  
                                         'event_sponsor_rel',  
                                         'event_id',  
                                         'sponsor_id',  
                                         'Sponsors'),  
    }  
}
```

one2many "virtual" relation

```
fields.one2many('model', 'id', 'Field Name', [, Par.] ),
```

- Example



```
class survey_question(osv.osv):  
    _columns = {  
        'answer_choice_ids': fields.one2many('survey.answer',  
                                             'question_id',  
                                             'Answer'),  
    }
```

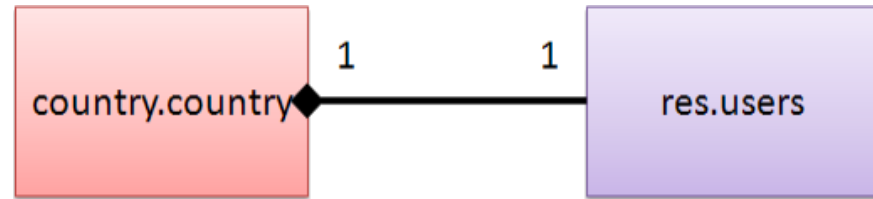
Match corresponding foreign key

```
class survey_answer(osv.osv):  
    _columns = {  
        'question_id': fields.many2one('survey.question',  
                                       'Question'),  
    }
```

one2one relation (deprecated)

```
fields.one2one('model', 'Field Name' [, Opt.Param.]),  
fields.many2one('model', 'Field Name' [, Opt.Param.]),
```

- Example



```
class country_country(osv.osv):  
    _columns = {  
        'president_id': fields.one2one('res.users', 'President'),  
    }
```

```
class country_country(osv.osv):  
    _columns = {  
        'president_id': fields.many2one('res.users', 'President'),  
    }
```

- Each country has **1** (and only 1) president,
and a person can be president of only **1** country 96

Relation fields' names

- By convention,
 - many2one fields end with '_id'
 - one2many fields end with '_ids'
 - many2many fields end with '_ids'

```
class survey_answer(osv.osv):  
    _columns = {  
        'question_id': fields.many2one(  
  
class survey_question(osv.osv):  
    _columns = {  
        'answer_choice_ids': fields.one2many(
```

```
class idea_idea(osv.osv):  
    _columns = {  
        'category_ids': fields.many2many(
```

3. DEVELOPMENT

Open ERP models

Module description

Fields of the model (columns)

Views (in XML)

Webservice communication

XML files for views

- Views (`ir.ui.view`)
 - Tree view
 - Form view
 - Search view
- Actions (`ir.actions.act_window`)
- Menus (`menuitem`)

Tree view

127.0.0.1

http://127.0.0.1:40069/#page=0&limit=80&view_type=list&model=idea.idea&menu_id=356&action=413

Messagerie Ventes Comptabilité Purchases Entrepôt Manufacturing Tools Autres options Demo User

DEMO COMPANY

Configuration

- Idées
- Idées
- Catégories

Idées

Créer

1-2 sur 2

<input type="checkbox"/>	Résumé de l'Idée	Créateur	État
<input type="checkbox"/>	Use 3D printers for rapid prototyping	Administrator	Terminé
<input type="checkbox"/>	Use laser cutting for rapid prototyping	Demo User	Acceptée

Columns/Fields

Powered by **OpenERP**

Tree view

- Extracted from `idea_view.xml`

```
<record model="ir.ui.view" id="view_idea_idea_tree"> ← XML id
  <field name="name">idea.idea.tree</field> ← Name of the view
  <field name="model">idea.idea</field> ← Name of the business object

  <field name="arch" type="xml"> ← Use XML tags to define the view
    <tree colors="blue:state == 'draft';black:state in ('open',
      <field name="name"/> ← Names of the columns to display
      <field name="create_uid"/> ←
      <field name="state"/> ←
    </tree>
  </field>

</record>
```

Search view

The screenshot shows a web browser window with the URL `http://127.0.0.1:40069/#page=0&limit=80&view_type=list&model=idea.idea&action=413`. The browser title is `127.0.0.1`. The top navigation bar includes `Messagerie`, `Ventes`, `Comptabilité`, `Purchases`, `Entrepôt`, `Manufacturing`, `Tools`, and `Autres options`. The user is identified as `Demo User`.

The main content area is titled `Idées` and features a `Créer` button. Below the title, there are three items with checkboxes:

- Résumé de l'Idée
- Use 3D printers for rapid prototyping
- Use laser cutting for rapid prototyping

A search bar is located at the top right of the main content area. A dropdown menu is open, showing filter options:

- Filtres**
 - Nouvelle
 - En cours
 - Acceptée
- Filtres personnalisés**
 - ▶ Enregistrer le filtre actuel
 - ▶ Recherche avancée
 - ▶ Ajouter au tableau de bord
- Grouper par ...**
 - Créateur
 - Catégorie
 - État

Two blue arrows point from text annotations to the dropdown menu:

- An arrow points from the text `« domain » filters` to the `Filtres` section.
- An arrow points from the text `« context » filters` to the `Grouper par ...` section.

The footer of the page reads `Fourni par OpenERP`.

Search view

- Extracted from `idea_view.xml`

```
<record model="ir.ui.view" id="view_idea_idea_search"> ← XML id
  <field name="name">idea.idea.search</field> ← Name of the view
  <field name="model">idea.idea</field> ← Name of the business object
  <field name="arch" type="xml"> ← Use XML tags to define the view
    <search string="Ideas">
      <field name="name" string="Idea"/>
      <filter string="New" domain="['state','=','draft']" />
      <filter string="In Progress" domain="['state','=','open']" />
      <filter string="Accepted" domain="['state','=','close']" />
      <field name="category_ids"/>
      <group expand="0" string="Group By...">
        <filter string="Creator" context="{ 'group_by': 'create_uid' }" />
        <filter string="Category" context="{ 'group_by': 'category_ids' }" />
        <filter string="Status" context="{ 'group_by': 'state' }" />
      </group>
    </search>
  </field>
</record>
```

Filters

- Condition: (Field,Operator,Value)
 - Several conditions
 - Prefix notation (Polish notation)

```
[ '|', ('country_code', '=', 'be'), ('country_ code', '=', 'fr') ]
```

– XML characters

Operator	XML
=	=
≠	!=
<	<
≤	<=
>	>
≥	>=

Form view

127.0.0.1

http://127.0.0.1:40069/#id=1&view_type=form&model=idea.idea&menu_id=356&action=413

Messagerie Ventes Comptabilité Purchases Entrepôt Manufacturing Tools Autres options

DEMO COMPANY

Configuration

- Idées
- Idées
- Catégories

Idées /

Use 3D printers for rapid prototyping

Modifier Créer Autres options

1 / 2

Nouvelle Acceptée Terminé

Use 3D printers for rapid prototyping

Organization

Description

3D printers allow to print parts in ABS plastic after designing it with CAD.

Tags of the categories

Name of the idea

Description label

Description

State buttons

Powered by OpenERP

Form view

- Extracted from `idea_view.xml`


```
<record model="ir.ui.view" id="view_idea_idea_form"> ← XML id of the view
  <field name="name">idea.idea.form</field> ← Name of the view
  <field name="model">idea.idea</field> ← Name of the business object
  <field name="arch" type="xml"> ← Use XML tags to define the view
    <form string="Idea" version="7.0">
      <header>
        <button name="idea_open" string="Open" states="draft" class="oe_highlight"/>
        <button name="idea_close" string="Accept" states="open" class="oe_highlight"/>
        <button name="idea_cancel" string="Refuse" states="open" class="oe_highlight"/>
        <field name="state" widget="statusbar" statusbar_visible="draft,open,close"/>
      </header>
      <sheet>
        <label for="name" class="oe_edit_only"/>
        <h1><field name="name"/></h1>
        <label for="category_ids" class="oe_edit_only"/>
        <field name="category_ids" widget="many2many_tags"/>
        <label for="description"/><newline/>
        <field name="description"/>
      </sheet>
    </form>
  </record>
  <div class="oe_chatter">
    <field name="message_follower_ids" widget="mail_followers"/>
```

Action and menuitem

- Extracted from `idea_view.xml`

```
<record model="ir.actions.act_window" id="action_idea_idea">
  <field name="name">Ideas</field>
  <field name="res_model">idea.idea</field>
  <field name="view_type">form</field>
  <field name="view_mode">tree,form</field>
  <field name="search_view_id" ref="view_idea_idea_search"/>
</record>

<menuitem name="Ideas" parent="menu_ideas" id="menu_idea_idea"
          action="action_idea_idea" sequence="1"/>
```



- Must be defined at **after** the declarations of the views

3. DEVELOPMENT

Open ERP models

Module description

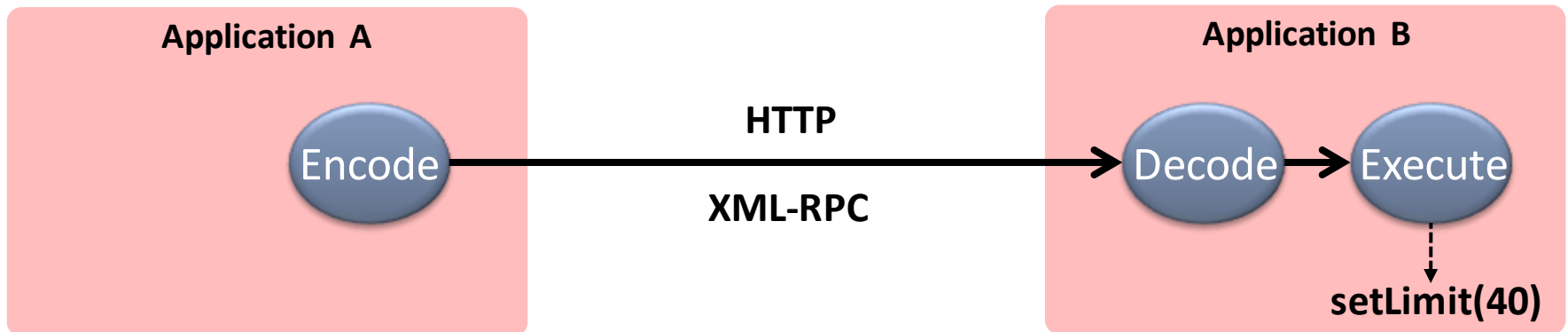
Fields of the model (columns)

Views (in XML)

Webservice communication

Webservices

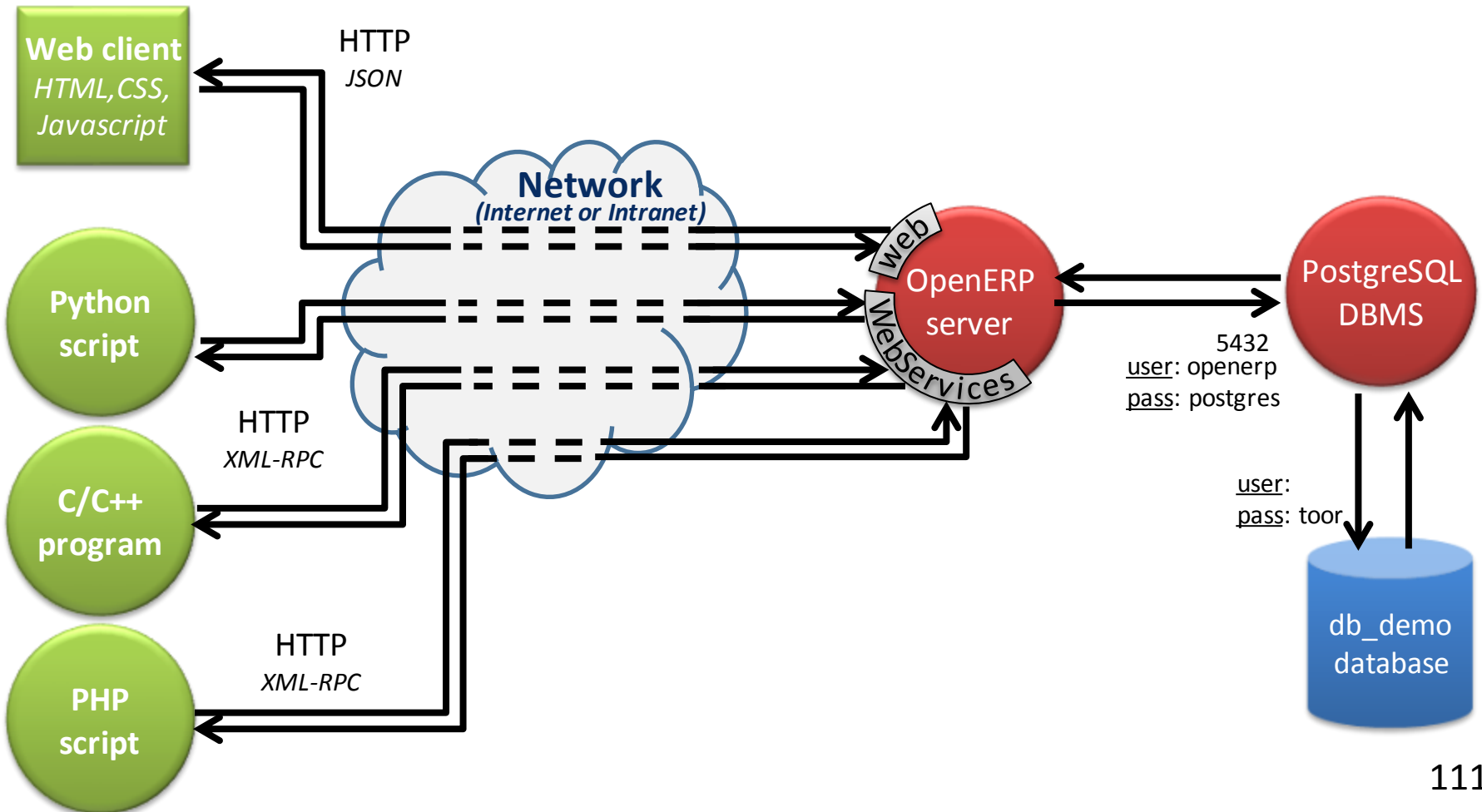
- SOA: **S**ervice **O**riented **A**rchitecture
- WOA: **W**eb **O**riented **A**rchitecture
- XML-RPC: XML **R**emote **P**rocedure **C**all \Rightarrow OpenERP



```
<?xml version="1.0" ?>
<methodCall>
  <methodName>setLimit</methodName>
  <params>
    <param><value><int>40</int></value></param>
  </params>
</methodCall>
```

Webservice with OpenERP

- OpenERP v7 v8 v9



OpenERP webservice

- Allows calling the methods of the ORM
 - **Create**: create a new record
 - **Write**: update a record
 - **Search**: look records matching a criteria
 - **Read**: get the values of a record
 - **Unlink**: delete a record
- Steps
 - 1 : connexion and identification
 - 2 : execution
 - 3 : disconnexion

Calling ORM methods

- Can be called from any program written with a language having a XML-RPC library
 - Python
 - C/C++
 - Objective-C
 - Java (*and also with android-xmlrpc*)
 - JavaScript
 - PHP (*with Pear library for PHP*)
 - Perl
 - ...

The create() ORM method

- `create({'field': 'value'})`
 - Creates a new record with the specified value
 - Returns: id of the new record
- Example with Python

```
import xmlrpclib

# Get the uid
sock_common = xmlrpclib.ServerProxy('http://127.0.0.1:8069/xmlrpc/common')
uid = sock_common.login('db_name', 'username', 'passwd')

# Ask for execution on OpenERP
sock = xmlrpclib.ServerProxy('http://127.0.0.1:8069/xmlrpc/object')
partner_id = sock.execute('db_name', uid, 'passwd',
                          'res.partner',
                          'create',
                          {'name': 'Fabien Pinckaers', 'lang': 'en_US', })
```

Example with PHP

- Example with PHP (*without error checking*)

```
require_once('XML/RPC.php'); // Include PEAR library for XML-RPC

// Login

$client = new XML_RPC_Client('/xmlrpc/common', "http://127.0.0.1:8069");
$client->setDebug(0);

$msg = new XML_RPC_Message('login');
$msg->addParam(new XML_RPC_Value('db_name', 'string'));
$msg->addParam(new XML_RPC_Value('username', 'string'));
$msg->addParam(new XML_RPC_Value('passwd', 'string'));

$res = $client->send($msg); // Send execution request on OpenERP
$val = $res->value();
$uid = $val->scalarval();
```

Example with PHP

```
// Request
$client = new XML_RPC_Client('/xmlrpc/object', "http://127.0.0.1:8069");
$client->setDebug(0);

$structVal = array(
    'name' => new XML_RPC_Value('Fabien Pinckaers', 'string'),
    'lang' => new XML_RPC_Value('en_US', 'string'),
);

$msg = new XML_RPC_Message('execute');
$msg->addParam(new XML_RPC_Value('db_name', 'string'));
$msg->addParam(new XML_RPC_Value($uid, 'int'));
$msg->addParam(new XML_RPC_Value('passwd', 'string'));
$msg->addParam(new XML_RPC_Value('res.partner', 'string'));
$msg->addParam(new XML_RPC_Value('create', 'string'));
$msg->addParam(new XML_RPC_Value($structVal, 'struct'));

$res = $client->send($msg); // Send execution request on OpenERP
$val = $res->value();
$id = $val->scalarval();
```


The search() ORM method

- `search([('arg1', '=', 'value1')...],
offset=0, limit=1000)`
 - `arg1, arg2, .. ,argN`: list of tuples specifying search criteria
 - `offset`: optional number of records to skip
 - `limit`: optional max number of records to return
 - Returns: list of IDS of records matching the given criteria

The read() ORM method

- `read([IDS], ['field1', 'field2', ...])`
 - fields: optional list of field names to return (default: all fields)
 - Returns: the id of each record and the values of the requested field

The write() ORM method

- `write([IDS],
{'field1':'value1','field2':3})`
 - Updates records with given ids with the given values
 - values: dictionary of field values to update
 - Returns: True

The `unlink()` ORM method

- `unlink([IDS])`
 - Deletes records with the given ids
 - Returns: True

OUTLINE

1. Overview

- Birth and growth
- Customers, Partners, Vendors

2. Administration

- Versions, Architecture
- Main modules

3. Development

- Module programming
- Webservice programming

4. Documentation











Documentation

- Documentation
 - `https://doc.odoo.com/`
 - `https://doc.odoo.com/7.0/fr/`
- Documentation technique
 - `https://doc.odoo.com/trunk/server/`
- Free E-Books
 - `https://www.odoo.com/ebooks`







Books

Authors	Title	Editor	Year	Pages	
Fabien Pinckaers, Geoff Gardiner	Tiny ERP-Open ERP : Pour une gestion d'entreprise efficace et intégrée	Eyrolles	2008	276	
F.Pinckaers, E.Van Vossel, B. Proust	Gestion commerciale et marketing avec Open ERP	Eyrolles	2012	183	
Pinckaers,Gardiner, Van Vossel	Open ERP: a modern approach to integrated business management	Tiny SPRL	2011	456	
Els Van Vossel, Fabien Pinckaers	Streamline your Manufacturing Processes with OpenERP	Tiny SPRL	2011	230	
Fabien Pinckaers, Geoff Gardiner	Open ERP for Retail and Industrial Management	Tiny SPRL	2009	320	
Els Van Vossel, Fabien Pinckaers	Integrate your Logistic Processes with OpenERP	Tiny SPRL	2011	230	
Els Van Vossel, Fabien Pinckaers	Drive your Sales & Marketing Activities with OpenERP	Tiny SPRL	2011	202	


Books

Authors	Title	Editor	Year	Pages	
Els Van Vossel, Fabien Pinckaers	OpenERP for Accounting and Financial Management	Tiny SPRL	2012	188	
Fabien Pinckaers, Geoff Gardiner	Analytic & Financial Accounting Book	Tiny SPRL	2009	157	
Fabien Pinckaers, Els Van Vossel	Open Source Accounting with OpenERP	Tiny SPRL	2011	176	
Y. Delsart, C. Van Nieuwenhuysen	OpenERP evaluation with SAP as reference	Feridis	2011	132	
Simon André	Piloter Prestashop avec OpenERP	Anybox	2012	27	
Florent Pigout	Piloter Magento avec OpenERP	Anybox	2011	91	
	Open Object Business Intelligence	Tiny SPRL	2009	113	
	Open Object Developer Book	Tiny SPRL	2009	213	
	OpenERP Agile Implementation Memento	Open Object	2010	5	
	OpenERP Technical Memento	Open Object	2010	18	

Books

Authors	Title	Editor	Year	Pages	
Greg Moss	Working with OpenERP	Packt	2013	334	
M. Belaïssaoui, A. Elkafil	Gestion Commerciale avec OpenERP 7 par la pratique	Éd. Univ. Européennes	2013	76	
Karl Hellec	Open ERP : Découverte et mise en œuvre	GEP	2013	110	
Daniel Reis	Odoo Development Essentials	Packt	2015	214	
Daniel Reis	Odoo 10 Development Essentials	Packt	2016	289	
Greg Moss	Working with Odoo 10	Packt	2017	415	

Books for high school :

S. Fasciotti, F. Roche, <i>et al.</i>	2 nd Bac Pro - Informatique - Office 2010, Access, Ciel, EBP PGI, OpenERP	Nathan	2012	245	
D. Le Rouzic	Activités sur... OpenERP Réseau	B. Lacoste		160	