SocrateCloud Architecture

The difference from Compiere

Whitepaper

November 2012

version 1.0

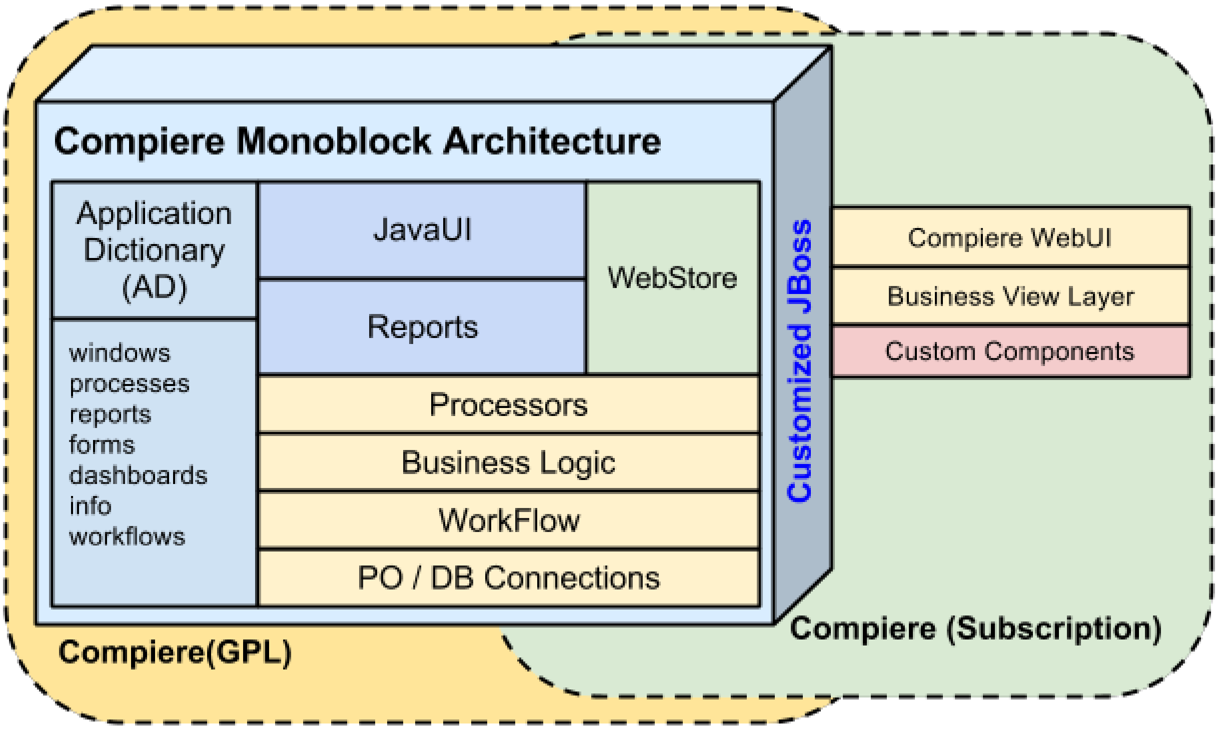
***SocrateCloud** is the natural consequence of 7 years of intensive development around the Compiere platform and is used today in small and medium companies, but also in enterprise environments with hundreds of concurrent users and millions of transactions. This paper describes briefly the architectural components which makes the difference from the original Compiere system and requires an understanding of the original Compiere architecture.*

Cloud Architecture

Cloud computing means automatic scalability and provisioning, even if we speak about applications hosted in public clouds with shared instances, when the user does not know what's behind the application or applications hosted in private clouds, where the infrastructure behind is dedicated to one customer.
Scalability means to have enough resources when black friday arrives or when there are peaks in orders/transactions during the special events, or when the number of users is increasing exponentially.
In order to assure the automatic scalability, the prerequisite is to have a partitionable application and modularized architecture with a clear split between the webUI, reporting, processors, business logic and DB connectivity.

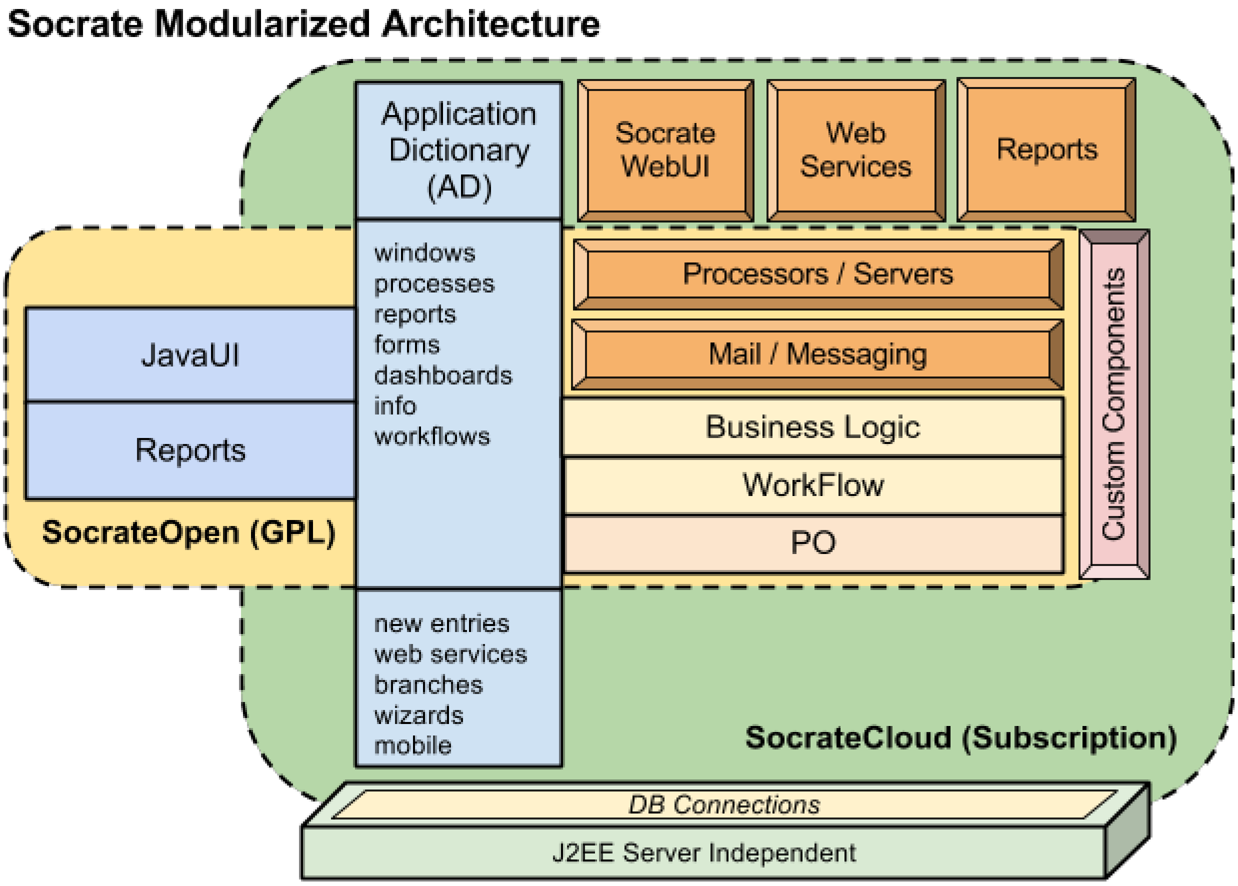
Compiere monoblock architecture

The heritage of Compiere is an architecture where everything (DB connections, business logic, presentation layer and reporting) is build into one component, which makes impossible to scale for a large volume of users and transactions.



SocrateCloud modularized architecture

SocrateCloud allows to scale automatically based on the resource utilization, as for example if suddenly the number of users explodes and the workload of the application server exceeds a certain limit, more instances (application components) are provisioned instantly. This applies for webUI, reporting, DB connections and processors and it is designed in such a way that a constant performance level is assured for the end user.



Load balancing mechanisms assure that user requests are redirected automatically to available resources. For example if more users access heavy reports, more instances with the reporting component are provisioned for taking the workload of the main application servers/components.

Cloud deployments

First of all, don't be confused by the "Cloud" word from SocrateCloud name. Cloud means "Cloud Architecture" = *unlimited scalability and instant provisioning* and does not mean that the application cannot be deployed on-premise.
However in order to achieve maximum scalability we recommend to deploy on a cloud-based infrastructure, even if we speak about on-premise deployments or private clouds (most of public clouds assure the instant provisioning).

One of the non negotiable principle which surrounds Socrate is the "cloud independence" philosophy, which means that SocrateCloud can migrate easily from one cloud to another one, dependent of course of the cloud infrastructure capabilities.
Today it can be deployed in three editions:

- on-premise
- private cloud based on AWS private instances
- public cloud based on AWS shared instances

SocrateCloud used a lot of components from AWS (Amazon Web Service) and we are planning to support also other public clouds, such as Oracle Public Cloud, dependent on the service availability around the world.

Application Dictionary enhancements

SocrateCloud preserves the Model Driven Architecture and the standard AD capability exposed by Compiere. The Application DIctionary (AD) was extended with advanced functionalities for a more granular and detailed configurability.

Customization at Tenant Level

To obtain maximum configurability in cloud environments, we have extended the AD with a lot of new functionalities, the most important is that with SocrateCloud you can customize the windows, processes, reports etc. at Tenant level and not on System level, as it is in Compiere and other similar packages.
Of course, System level configuration remains valid and AD definitions can be grouped based on Business Branches in order to expose only the specific functionalities per each industry/vertical.
All customizations are easy to be marked as “Customization” to be kept safe for the new releases.

Tables

For advanced configurability, the column definitions were extended for "smart copy" functionality, by which we can mark what columns should be copied. Further extensions are related to mobileUI which will be deployed in future releases.

Windows, Tabs & Fields

At Tab level we can specify if lines can be deleted (in Compiere it is possible only at table level), business logic for warnings at save records, options to search in attachments for performance improvement reasons. You can mark now how a fields group is shown by default on the webUI - open or collapsed.
Windows configurability was extended by the introduction of callouts (both JavaUI and WebUI) for a field within a window.

Processes

A lot of improvements were done in the definition of the reports and processes: you can expose parameters in multiple columns, you can add *like* searches and read-only parameters, you specify the default focus and if a parameter should have values instead of "all".

Wizards

New wizards functionality was introduced in order to shorten the setup process during the implementation projects. Wizards are inherited from System for different business lines (branches) and can be modified at tenant level.

Socrate WebUI

The webUI component was rebuild from scratch in order to achieve the highest performance and productivity, preserving the usability aspects appreciated by the users, both from the JavaUI and CompiereWebUI.
Comparing with CompiereWebUI it is at least ten times faster and much richer in features and the same comparison applies also to the JavaUI.
The new web interface was designed to be successfully used on netbooks with small screens, but also on laptops/compute with large screens. From that reason, the menu can be hidden and Favorites can be accessed from top menu bar or based on the Search function.

Advanced Grids

The grids are developed using SmartGWT/ListGrid component and have a lot of new functionalities such as: Column selection, Column ordering, Group By, HighLighting and conditional formatting, Custom columns/formulas, Searching directly into grid etc. The grid layouts can be saved for every entity at user level preference.
On the same screen, you can see master-detail data.
There is no grid pagination in SocrateWebUI! The grid is populated 'live', as the user is scrolling down, in chunks of 75 rows; this method which allows the seamless navigation in grids with large amounts of records. Column order is preserved through the caching mechanisms and user preferences.

Caching

This high performance is achieved due to the advanced caching mechanism implemented in the webUI module. For security reasons, all private data for each user is kept separately within his session.
The Windows Metadata is saved both on the Client Cache and on the Server Cache. The metadata is saved after the first query and if another session (user) is opening the same Window, the metadata is used from the Server Cache.
Besides the metadata, the Server Cache also stores the queries constructions which reduce significantly the execution time.

Windows

InfoWindows

Info windows are fully customizable from the Active Dictionary.

NewRecord Windows

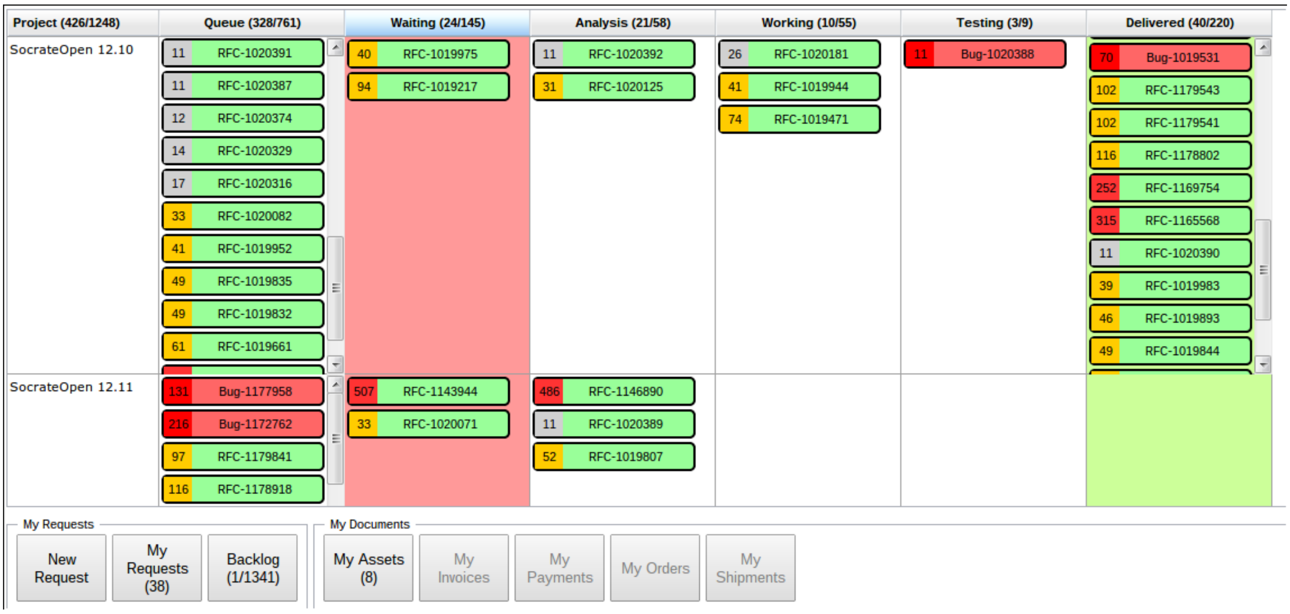
For entering new entities (new lead, new task, new request, new Business Partner etc.) special simplified forms are used for quick entries, The forms are fully customized in the AD, based on two role types: internal and external (restricted BP).

Search

You can search quickly in menu or in grids. Favorites queries can be saved. Each user can save the most frequently queries and use them next time directly.

K-Boards

Kanban boards is a visualization component based on Kanbans. It allows the visual management of the information flows and you can apply lean management concepts for service management, sales pipelines and different other areas, bringing more flexibility and visibility to the business processes.



Project (426/1248)	Queue (328/761)	Waiting (24/145)	Analysis (21/58)	Working (10/55)	Testing (3/9)	Delivered (40/220)
SocrateOpen 12.10	11 RFC-1020391 11 RFC-1020387 12 RFC-1020374 14 RFC-1020329 17 RFC-1020316 33 RFC-1020082 41 RFC-1019952 49 RFC-1019835 49 RFC-1019832 61 RFC-1019661	40 RFC-1019975 94 RFC-1019217	11 RFC-1020392 31 RFC-1020125	26 RFC-1020181 41 RFC-1019944 74 RFC-1019471	11 Bug-1020388	70 Bug-1019531 102 RFC-1179543 102 RFC-1179541 116 RFC-1178802 252 RFC-1169754 315 RFC-1165568 11 RFC-1020390 39 RFC-1019983 46 RFC-1019893 49 RFC-1019844
SocrateOpen 12.11	131 Bug-1177958 216 Bug-1172762 97 RFC-1179841 116 RFC-1178918	507 RFC-1143944 33 RFC-1020071	486 RFC-1146890 11 RFC-1020389 52 RFC-1019807			

Same visualization component is used by the Calendars and Tasks.

For more details, please consult the functional whitepaper.

Reporting and Printing

Report Engine

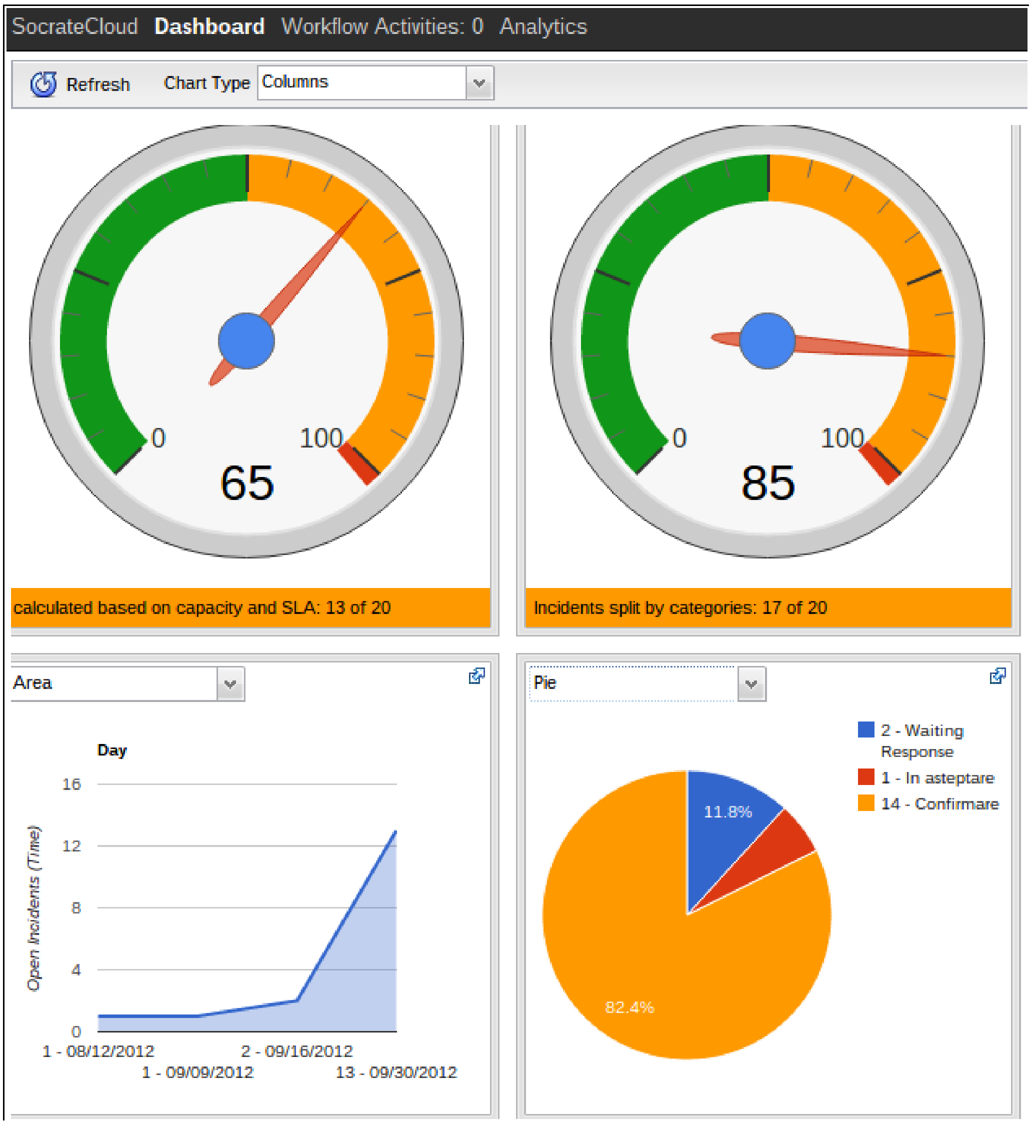
Compiere report engine is the same as in the Community Edition, but the PDF component for generating PDFs was replaced with a GPL open source components.The print formats are fully customizable.

Printing

When we talk about the automated (background) processes for printing, such as Batch invoices printing, SocrateCloud uses GoogleCloud Print technology for printing purposes, You have total control of what and where you print, customizable at the print format level.
If you want to print manually, you can choose the printer as you want.

Dashboards

New dashboards were designed based on GoogleChart API. The dashboards are role based and can be defined in SocrateCloud. They can be accessed in a separate tab.



calculated based on capacity and SLA: 13 of 20

Incidents split by categories: 17 of 20

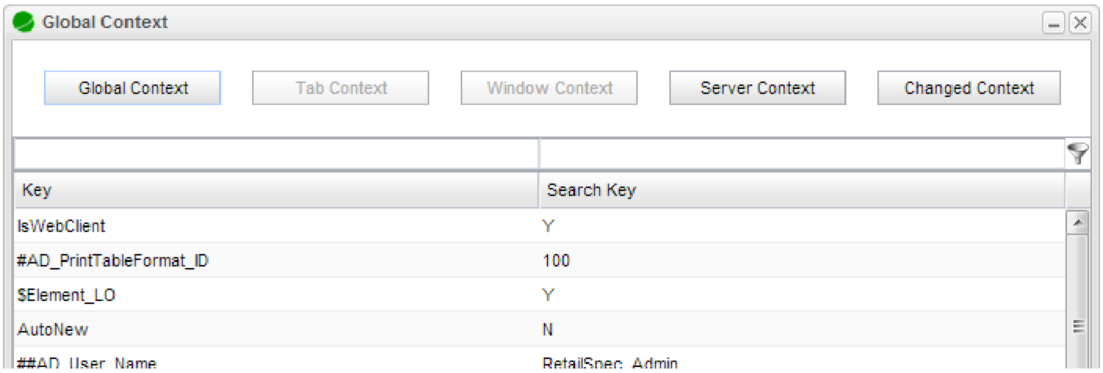
SocrateBI

For enterprise customers, smart visualization and advanced analytics are available through the MicroStrategy platform which is embedded and integrated with SocrateCloud.
SocrateBI comprises a set of BI applications for different business areas (Sales, Financial, Inventory etc.), a data warehouse (SocrateDW) and ETL components (SocrateETL).

Monitoring

Logs and Context

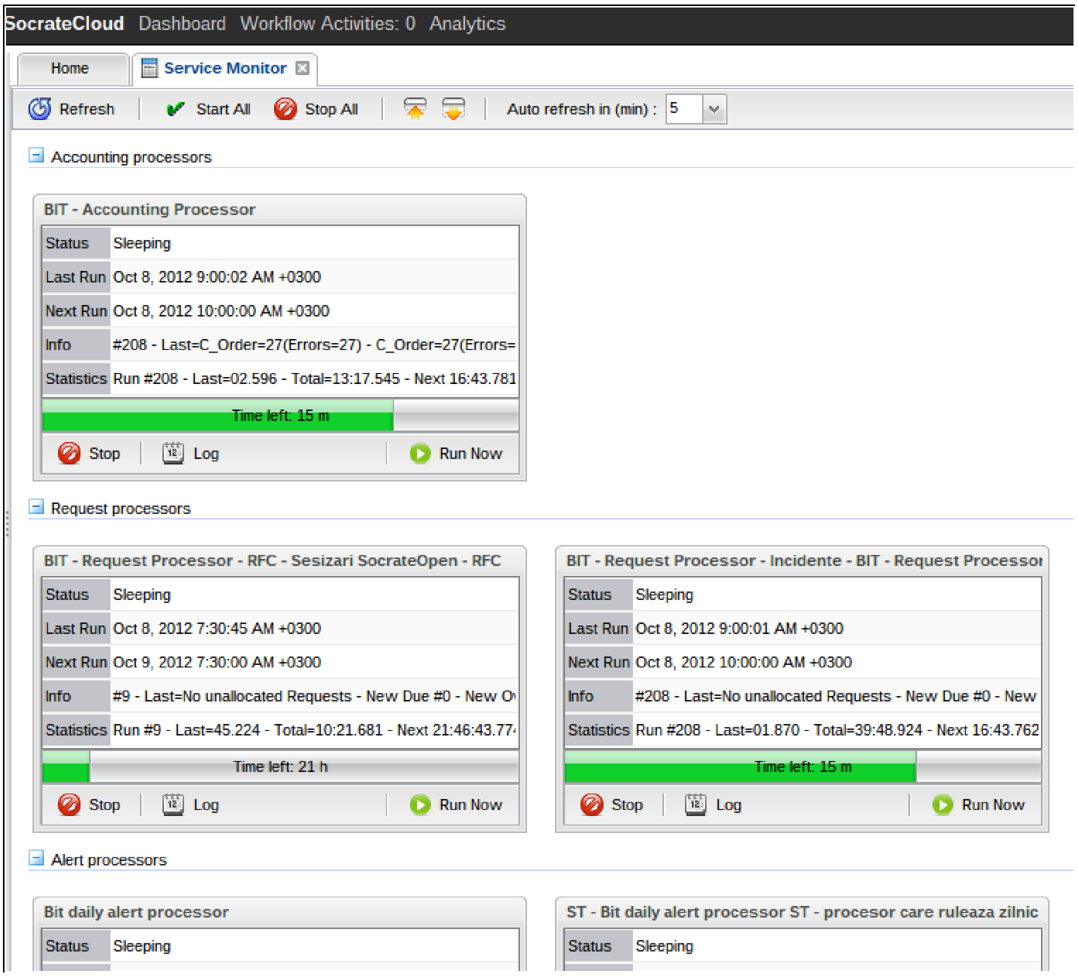
Context is now displayed in a more friendly way and filters can be applied. You can see the global context, tab & window context, server context and changed context. With that you can easily adjust display logic or read only logic for fields as your business request.



Key	Search Key
IsWebClient	Y
#AD_PrintTableFormat_ID	100
$Element_LO	Y
AutoNew	N
##AD_User_Name	RetailSpec_Admin

Socrate Processors Monitor

Monitoring the processors ("servers") is now very easy with more advanced functionalities and you can have access to Accounting processors, Request processors, Alerts, Scheduled jobs etc. Each processor can be started or stopped manually.



Accounting processors

BIT - Accounting Processor

Status	Sleeping
Last Run	Oct 8, 2012 9:00:02 AM +0300
Next Run	Oct 8, 2012 10:00:00 AM +0300
Info	#208 - Last=C_Order=27(Errors=27) - C_Order=27(Errors=
Statistics	Run #208 - Last=02.596 - Total=13:17.545 - Next 16:43.781

Request processors

BIT - Request Processor - RFC - Sesizari SocrateOpen - RFC

Status	Sleeping
Last Run	Oct 8, 2012 7:30:45 AM +0300
Next Run	Oct 9, 2012 7:30:00 AM +0300
Info	#9 - Last=No unallocated Requests - New Due #0 - New O
Statistics	Run #9 - Last=45.224 - Total=10:21.681 - Next 21:46:43.77

BIT - Request Processor - Incidente - BIT - Request Processor

Status	Sleeping
Last Run	Oct 8, 2012 9:00:01 AM +0300
Next Run	Oct 8, 2012 10:00:00 AM +0300
Info	#208 - Last=No unallocated Requests - New Due #0 - New
Statistics	Run #208 - Last=01.870 - Total=39:48.924 - Next 16:43.762

Alert processors

Bit daily alert processor

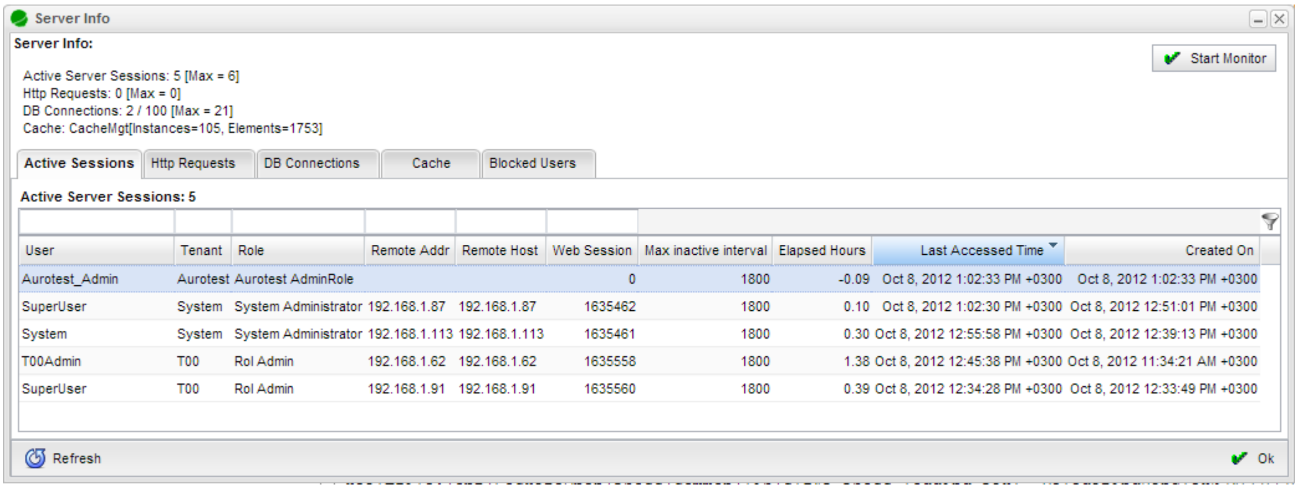
Status	Sleeping

ST - Bit daily alert processor ST - procesor care ruleaza zilnic

Status	Sleeping

Server Monitor

As System or SuperUser role you have access to the Administrator page to control memory cache, to consult trace log and to see other supplemental information about servers activity.



Server Info:

Active Server Sessions: 5 [Max = 6]
Http Requests: 0 [Max = 0]
DB Connections: 2 / 100 [Max = 21]
Cache: CacheMgt[Instances=105, Elements=1753]

Active Server Sessions: 5

User	Tenant	Role	Remote Addr	Remote Host	Web Session	Max inactive interval	Elapsed Hours	Last Accessed Time	Created On
Aurotest_Admin	Aurotest	Aurotest AdminRole			0	1800	-0.09	Oct 8, 2012 1:02:33 PM +0300	Oct 8, 2012 1:02:33 PM +0300
SuperUser	System	System Administrator	192.168.1.87	192.168.1.87	1635462	1800	0.10	Oct 8, 2012 1:02:30 PM +0300	Oct 8, 2012 12:51:01 PM +0300
System	System	System Administrator	192.168.1.113	192.168.1.113	1635461	1800	0.30	Oct 8, 2012 12:55:58 PM +0300	Oct 8, 2012 12:39:13 PM +0300
T00Admin	T00	Rol Admin	192.168.1.62	192.168.1.62	1635558	1800	1.38	Oct 8, 2012 12:45:38 PM +0300	Oct 8, 2012 11:34:21 AM +0300
SuperUser	T00	Rol Admin	192.168.1.91	192.168.1.91	1635560	1800	0.39	Oct 8, 2012 12:34:28 PM +0300	Oct 8, 2012 12:33:49 PM +0300

J2EE Server independence

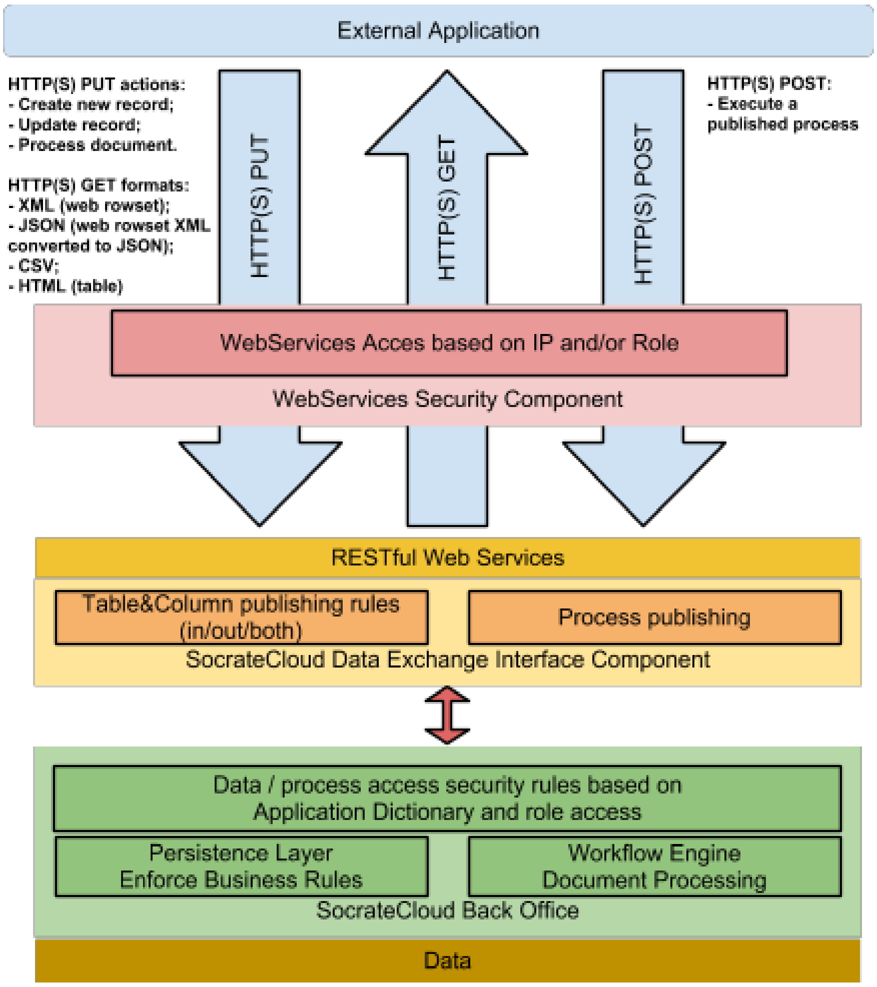
GlassFish is the default Application Server used. It is available both in SocrateOpen - the community edition - and in SocrateCloud. However, JBOSS v.7 is also a valid option. SocrateCloud is also supported with Oracle WebLogic Server 11g, Oracle Exalogic Elastic Cloud, and Oracle Exadata Database Machine, providing a lower cost and high performance infrastructure for database and application workloads across on-premise and cloud based environments.

DB Connections

For higher scalability the DB connections are handled through the application servers and they are not anymore embedded in the Persistence Object (PO).

WebServices

SocrateCloud Data Exchange Interface is based on RESTful web services and is designed to provide a simple setup, yet secure and controlled data transfer to and from the application. Used methods are HTTP GET and HTTP PUT. Execution process is done with the HTTP POST method. All this services definition can be made at a Tenant level. The security mechanism is based on Role, User and/or IP (IPs). Thanks to RESTful technology, this services can be used from most of the programing languages, not only from Java environment.



GoogleApps Integration

If you have a GoogleApps for Business account you can use the credentials from GoogleApps in order to sign-in in SocrateCloud and the security policy from Google will apply.

The SocrateCloudCRM functionality is tightly integrated with GoogleApps:

- Shared Contacts: Contacts defined in SocrateCloud can be shared to the GAL (Global Address List) in order to be used in GoogleApps
- Tasks and Events: Tasks defined in SocrateCloud are automatically seen in the Google Calendar
- Email integration: from emails received in GoogleApps you can easily add a new contact, new opportunity, new task, which are automatically seen in SocrateCloud.

Custom components development

To preserve the competencies from the open source community and from the Compiere partners community, components development remains similar as for Compiere. At the base are used extended PO objects and ModelValidator framework. The existing code written for compiere GPL is 100% compatible with SocrateCloud.

In addition, for SocrateCloud WebUI, BITSoftware provides to the authorized partner a complete SDK to be used for developing purposes of new forms or other custom visualization components.

We have preserved the standard mechanism thru which you can add new entities using Application Dictionary (Tables, Columns, Windows etc), maked distinct then Dictionary and with “User Maintained”, or modifying with “Customization”.

Security

In order to achieve the highest security standards, SocrateCloud use specials methods for authentication, data encryption and transaction audit.
Authentication mechanism are based on Single Sign On, 2-way authentication using the best existing standards and methods (OAuth), with encryptions based on SHA-256 algorithms.
Besides the implemented security mechanisms, the whole developing, testing and deployment processes are based on Secure Programming concepts and are fully audited.

From functional perspective we preserved all functionalities from Compiere standard on top of which we have added access rights for Document Actions, K-boards, Mobile devices, BI.

Data & Application Liberation

In order to offer the highest level of warranty about availability and safety of customers data, SocrateCloud provides a Data Liberation mechanism. Based on it, our customers can, always and anytime, move their data from a cloud infrastructure to their own private infrastructure. But ERP/CRM data without the application makes no sense. In this respect, SocrateCloud customers can continue to run their business switching either to SocrateOpen - the GPL open source version - or to SocrateCloud hosted on-premise.

For more detailed information, please consult the SocrateCloud Security Whitepaper.