

## Spagic Console

Author

Daniela Butano

1	Document Goal.....	3
2	Versions History .....	3
3	Introduction .....	4
3.1	Requirements .....	4
3.2	Installation .....	4
4	Authentication.....	6
5	Processes displaying.....	7
5.1	Search.....	8
5.1.1	Search by taxonomy .....	8
5.1.2	Search by iter .....	8
5.2	Process Detail.....	10
5.3	Graphical representation of the process .....	11
6	Processes instances displaying.....	12
6.1	Search.....	13
6.1.1	Search by taxonomy .....	13
6.1.2	Search by iter .....	13
6.2	Graphical representation of the process execution .....	14
6.3	Single restart.....	16
6.3.1	Known limitations.....	17
7	Messages Displaying .....	18
8	Advanced Search.....	19
9	Massive restart .....	20
10	Backup and Delete Monitoring Data .....	21

## 1 Document Goal

The goal of this document is to provide you with an introduction on using the Spagic Console.

## 2 Versions History

<b>Version/Release n°:</b>	1.0	<b>Date</b>	29/06/2007
<b>Description</b>	First release (English version)		
<b>Version/Release n°:</b>	1.1	<b>Date</b>	18/07/2007
<b>Description</b>	Added some details for Restart Single and Restart Massive features		
<b>Version/Release n°:</b>	1.2	<b>Date</b>	03/08/2007
<b>Description</b>	Added some details for Restart Massive features. Added Backup and Delete Monitoring Data section		

## 3 Introduction

**Spagic Console** is the Enterprise Monitoring released by open source Spagic platform.

Spagic Console contains:

- **System monitoring** to visualize resources such as the threads, the memory, the queues used by ESB platform;
- **Service monitoring** to visualize processes/instances and theirs properties, processes flows and execution s processes flows, relevant data. This feature allows to restart the processes in error state, one at a time or more at a time;
- **Business indicators (BAM)** to visualize reports and dashboards, realized on SpagoBI, easy extendible through the realization of report and dashboard towards its own business intelligence indicators.

The application was developed using the Ajax technique to increase the page's interactivity, speed, functionality.

The url to launch the application is: <http://host:port/Spagic/>.

### 3.1 Requirements

Software requirements to use the Spagic Console application:

- ✓ Database MySQL or Oracle
- ✓ Apache Tomcat 5.5.17 <http://tomcat.apache.org>
- ✓ JDK 1.5.0\_11 or later <http://java.sun.com/>
- ✓ Mozilla Firefox 2.0.0.x <http://www.mozilla.com>

### 3.2 Installation

To install Spagic Console follow the next steps:

1. Create into a database a new schema "*spagobi*" for SpagoBI environment, create a new user "*spagobi*" with password "*bispago*" and associate him the schema privileges for writing and reading. Execute the "*spagobi-  
<database type>.ddl*", released with Spagic in the package *spagic-console/DDL*, to create the tables, and execute the script *loadSpagobi-  
<database type>.sql* to load the configuration data.
2. Install the Apache Tomcat 5.5.17; to install it refer to its installation documentation (<http://tomcat.apache.org>)
3. Install the web application SpagicConsole, you should copy into *apache-tomcat-5.5.17\webapps* folder the *Spagic.war* released with Spagic.



4. **If you used a configuration different from the default suggested in this document**, before starting Tomcat it 's necessary to verify the following Spagic Console **configuration files** and update them:


1. *|SpagicConsole|WEB-INF|conf|cms.xml*: set the location of the SpagoBI CMS repository in the *value* attribute of the "*repository\_path*" parameter, if you installed the folder externally to the Web application. For example, if you installed *cmsspagobi* folder into "C:\temp\cmsspagobi" set the value attribute as *value="/temp/cmsspagobi"*;

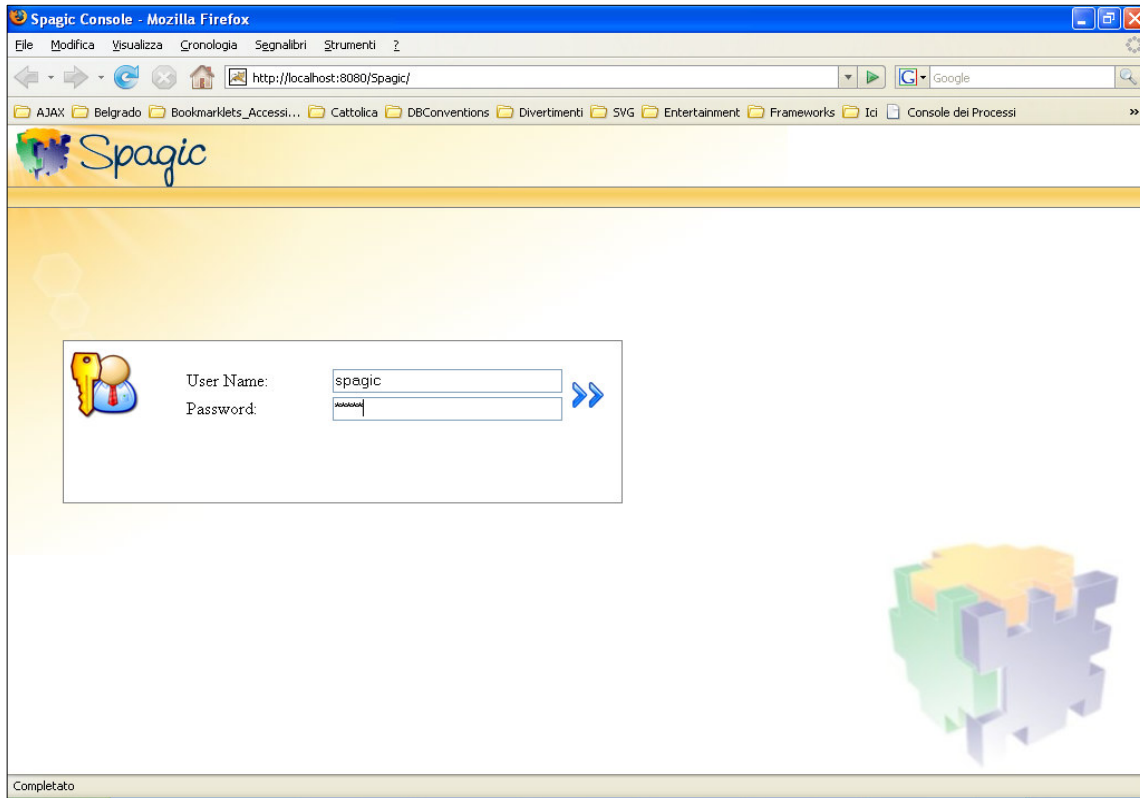
2. `\SpagicConsole\WEB-INF\conf\data_access.xml`: set the values of parameters *connectionString*, *user*, *userPassword* with the proper values to connect to the persistent layer metadb (default schema is "smx" and default user is "smx");
3. `\SpagicConsole\WEB-INF\conf\spagobi\spagobi.xml`: in the parameter `<HIBERNATE-CFGFILE>` set the location of Hibernate configuration file used by *SpagoBI*. If you installed MySQL database, the file used is `\SpagicConsole\WEB-INF\classes\hibernate.cfg.mysql.xml`.  
Update the Hibernate configuration file with the proper values to connect to SpagoBI schema (default schema is "spagobi").
4. `\SpagicConsole\WEB-INF\conf\jmx\server.xml`: set the *jmxUrl* to the URL for connecting to ServiceMix by JMX. The URL is written by ServiceMix on its console, on the startup.
5. Update the Hibernate configuration file `\SpagicConsole\WEB-INF\classes\hibernate.cfg.xml` so it could link to Spagic database (default schema and user is "smx");
6. If you used *UDDI*, update the file `\SpagicConsole\WEB-INF\classes\ServiceRegistry.properties` to set the URL of jUUDI application and all the parameters for the Spagic database connection.

## 4 Authentication

You can start the application from : <http://localhost:8080/Spagic/> .

The Spagic Console first page is the authentication page.

Insert the user name and the password given to you by the administrator and press the button 



Once the user name and password are confirmed the application shows those menu voices to which the user is enabled.

These are the foreseen characteristics:



**System** – allows the access to the services offered by the Service Manager used, at present ServiceMix3.1. It is possible to observe the system parameters – such as the number of threads used, the memory, and the classes uploaded run-time.



**Service** – consents the visualization of the processes (definitions, characteristics, flow, endpoints) and the processes' instances (dynamic flow, messages, important data, advanced search).



**Reports** - allows the displaying of the reports concerning the processes and their instances.



**Dashboards** – to display the dashboards.



## 5 Processes displaying

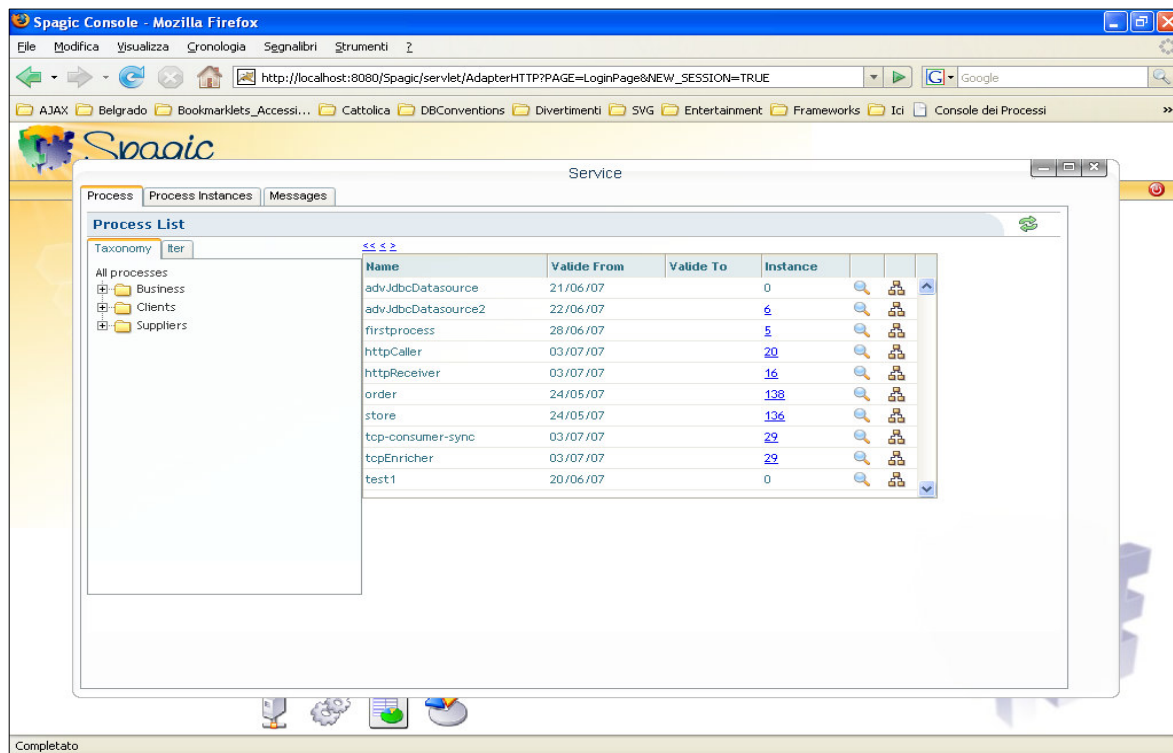
By pressing the **Service** icon a window containing the following two tabs is being displayed:

- **Processes Definition**
- **Processes Instances**
- **Messages**

The *Processes Definition* tab contains: on the left side – a section allowing the user to make a search by taxonomy or by iter; on the right side – the listed processes, ordered by process name. If no search is made, the list will contain all the processes. When more versions are available, only the latest versions will be displayed.

The list contains:

- *Name* of the process defined in the Spagic Studio;
- *Start Date* that corresponds to the process publication date on the repository;
- *End Date* that corresponds to the publication date of a new version of the process. A process cannot be executed after the its End Date;
- *Instances* contains the number of instances associated with the process; the instances can be seen by clicking on the link;
-  Icon allowing to see detailed information;
-  Icon for displaying the process graph;



On top of the right side of the page, you can find the button that allows you to update the list. In this version, when the list is updated, any filter parameter will be lost.

## 5.1 Search

It is possible to search the processes by taxonomy or by iter, using the section on the left side of the page containing the list of the processes.

### 5.1.1 Search by taxonomy

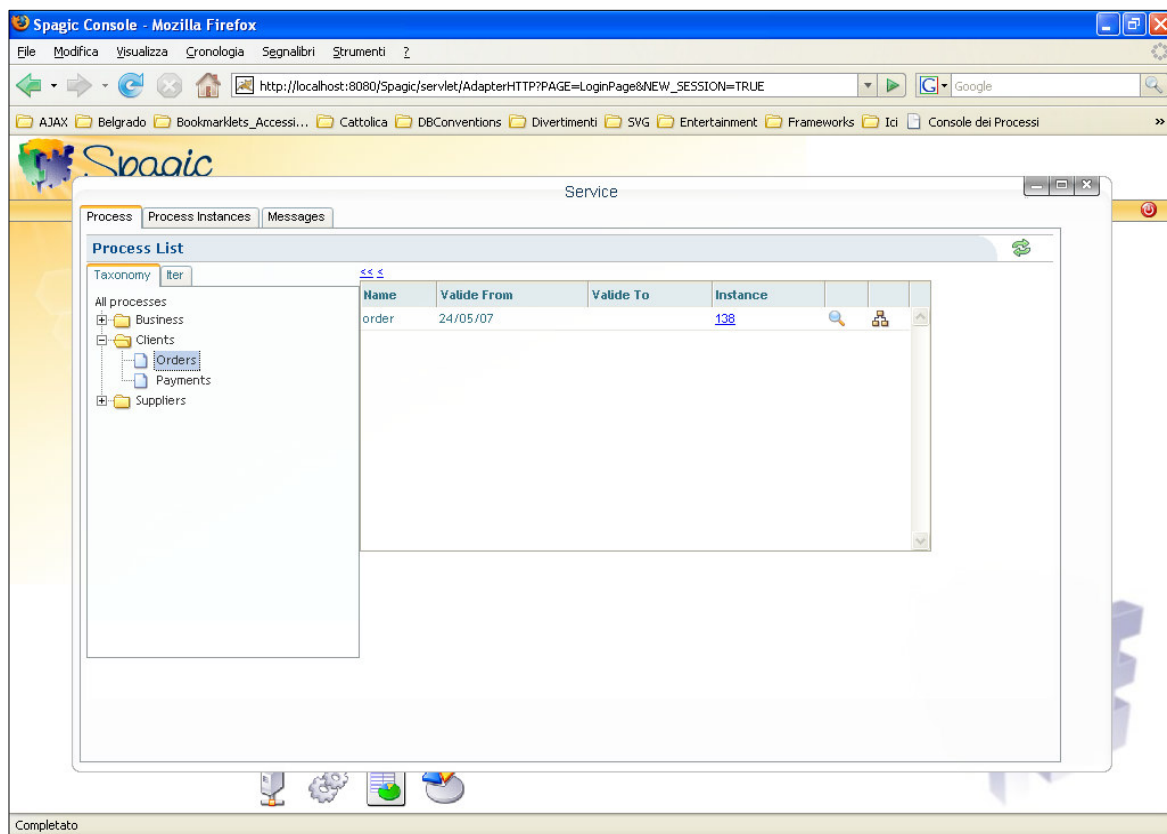
Spagic Studio offers the possibility of cataloguing by taxonomy a process published on the database.

The taxonomies (*Client* and *Supplier*, on the example in the image below) contain the classifications (on the example below, *Order* and *Payments* are the classifications of the taxonomy *Client*).

A process can be associated with more classifications, even of the same taxonomy.

When you choose a taxonomy from the taxonomies tree (the taxonomies are the folders), the classifications that the taxonomy contains are being displayed.

When you choose a classification, the processes list is being updated and will contain only the processes belonging to that classification.



### 5.1.2 Search by iter

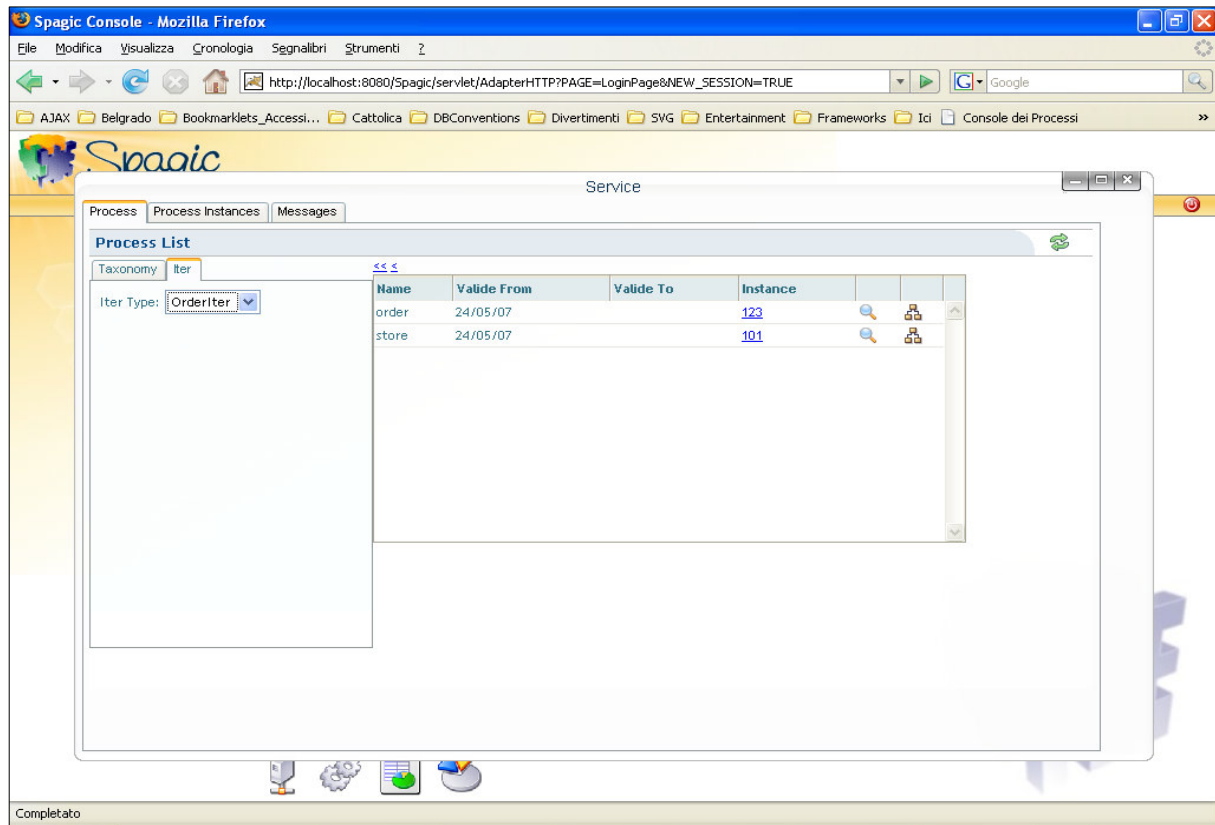
It is possible to make a search by iter, choosing the *Iter Type* combo. The list is automatically updated when you choose an element of the combo.

If you choose "Any Iter", all the processes will be displayed.

If you choose "No Iter", only the processes with no iter associated will be displayed.




If you choose a specific iter, the processes associated with that iter will be displayed; the *Instances* field contains the number of instances associated with that iter, that may not be equal to the total number of instances (because of the fact that, for instance, the association iter-process has been made subsequently).



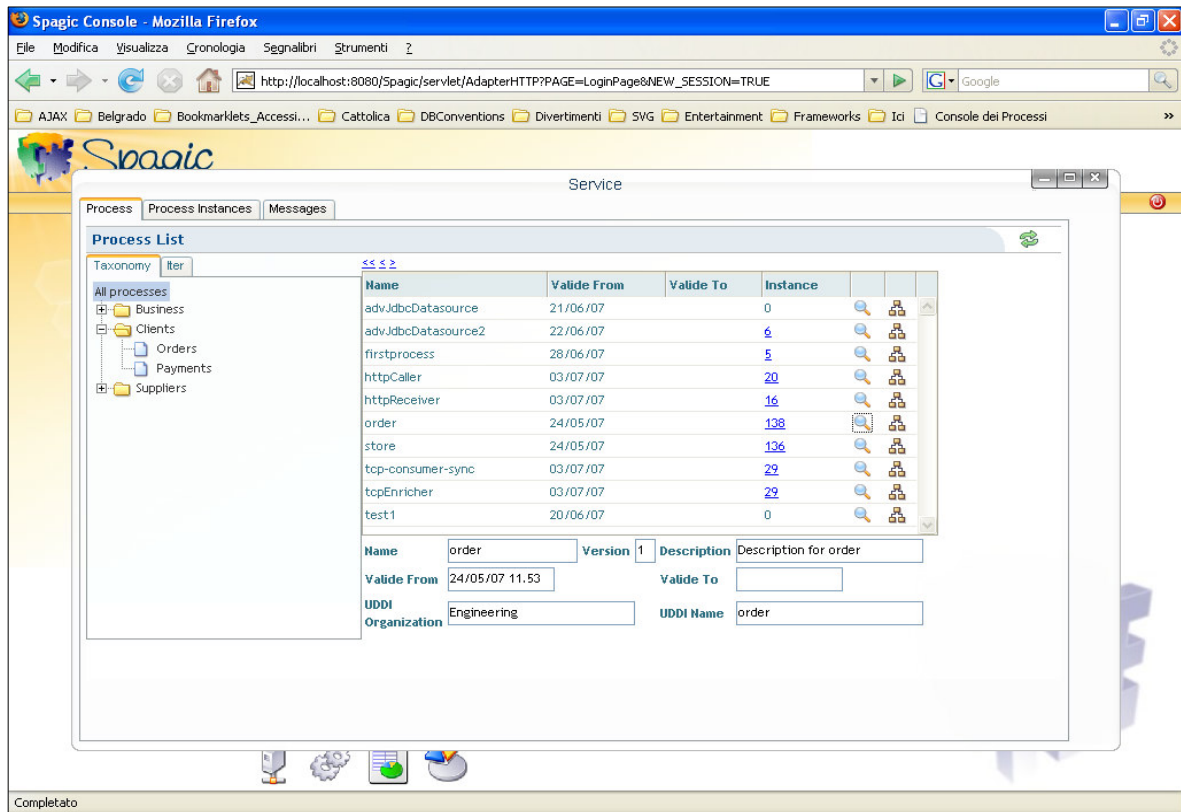
Name	Valide From	Valide To	Instance
order	24/05/07		123
store	24/05/07		101

## 5.2 Process Detail

Choosing the icon  from the processes list, you can see further details:

- *Version*, is always the latest version of the selected process;
- *Description* of the process (it is not possible yet to classify the description in Spagic Studio);
- *UDDI Organization*, contains the organization to which the process belongs;
- *UDDI Name*, name of the service that coincides with the name of the process.

When you choose a new element from the list, the details are automatically updated.




The screenshot shows the Spagic Console interface. On the left, there is a 'Process List' table with columns: Name, Valide From, Valide To, Instance, and icons. The table lists several processes, including 'advJdbcDatasource', 'advJdbcDatasource2', 'firstprocess', 'httpCaller', 'httpReceiver', 'order', 'store', 'tcp-consumer-sync', 'tcpEnricher', and 'test1'. The 'order' process is selected, and its details are shown in the 'Service' panel on the right. The details include: Name (order), Version (1), Description (Description for order), Valide From (24/05/07 11:53), Valide To, UDDI Organization (Engineering), and UDDI Name (order).

Name	Valide From	Valide To	Instance
advJdbcDatasource	21/06/07		0
advJdbcDatasource2	22/06/07		6
firstprocess	28/06/07		5
httpCaller	03/07/07		20
httpReceiver	03/07/07		16
order	24/05/07		138
store	24/05/07		136
tcp-consumer-sync	03/07/07		29
tcpEnricher	03/07/07		29
test1	20/06/07		0

**Service Details:**

Name	order	Version	1	Description	Description for order
Valide From	24/05/07 11:53	Valide To			
UDDI Organization	Engineering	UDDI Name	order		

## 5.3 Graphical representation of the process

Choosing the icon  from the processes list, it is possible to display the process graph.

The process graph is, by definition, a static view; it shows the process components: the binding components (represented by a rhombus), the service engine (represented by a rectangle) and the direction of the process flow (represented by the arrows).

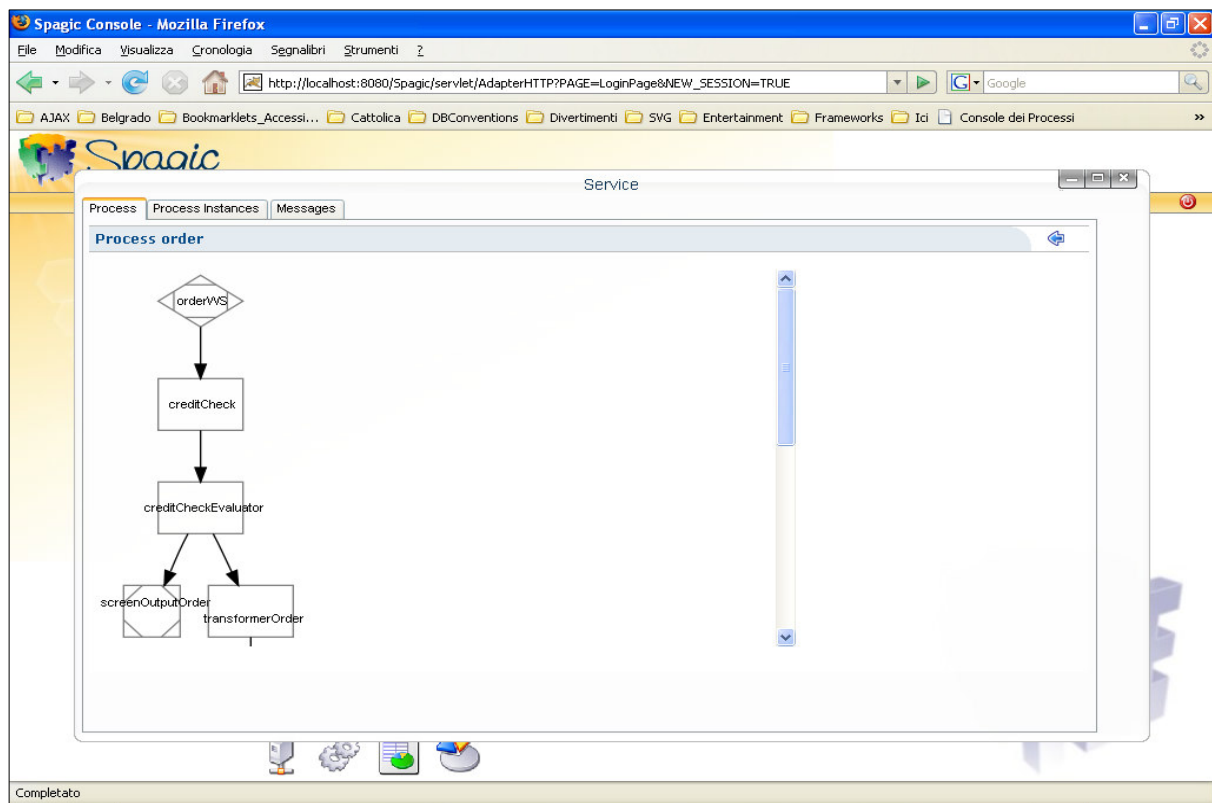
Clicking on the single element of the graph it's possible to display a window containing *Name* and *Description* of the end-point and *Service*, that is the type of the used component (servicemix-lwcontainer XsltComponent, for instance) and *Properties*.

Using the mouse you can change the position of the window or its dimensions, you can minimize it or close it.

When you choose a new element of the graph, the data in the window are updated.



Clicking this button you can turn back to the processes list.




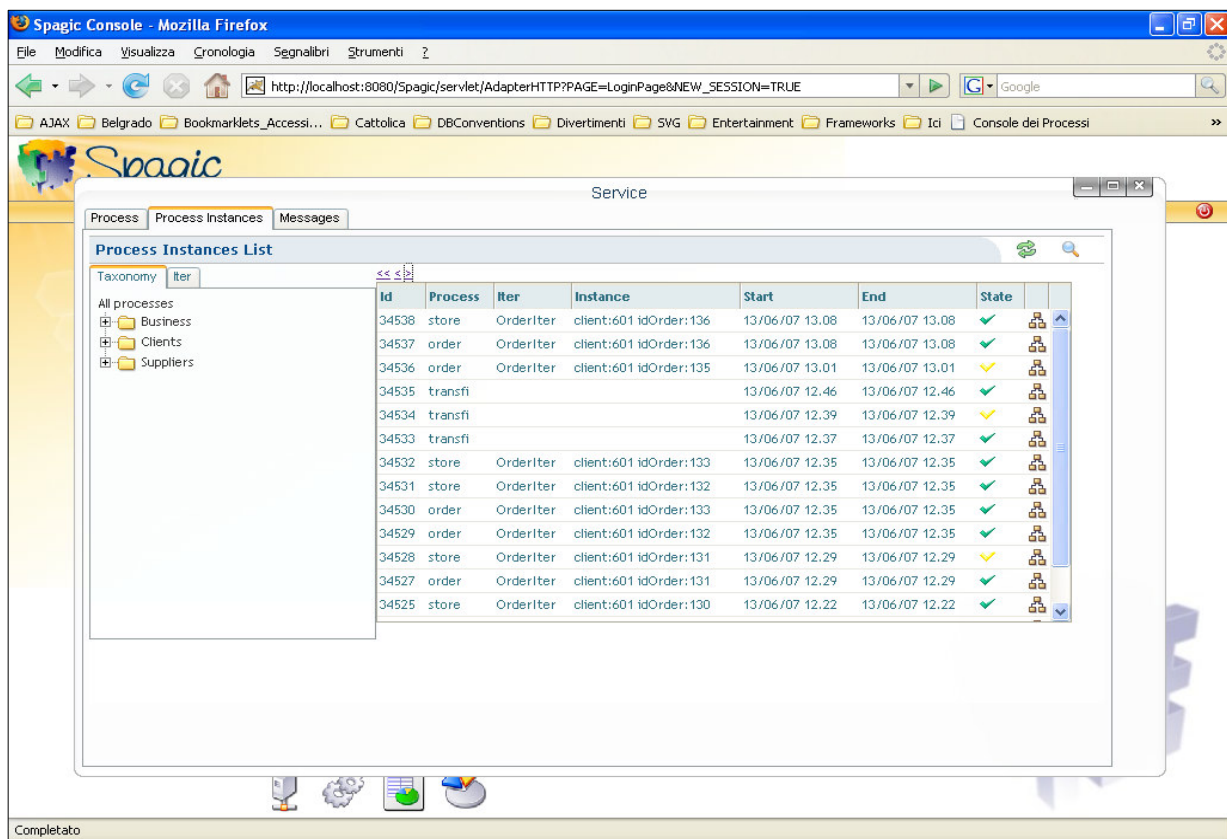
## 6 Processes instances displaying














By pressing the **Service** icon and the *Process Instances* tab, the page containing the instances of process is being displayed.

The page contains: on the left side – a section allowing the user to make a search by taxonomy or by iter; on the right side – the listed instances, ordered by start execution. If no search is made, the list will contain all the process instances.

The list contains:

- *Id* of process instance;
- *Nome* of the process defined in Spagic Studio;
- *Iter* associate to the instance;
- *Instance* containing the attributes with theirs values that identify the iter;
- *Start* of execution of the process;
- *End* of execution of the process ;
- *State* to visualize the state of the instance that could assume three possible values: the *executed* state (green flag) , the *active* state (yellow flag) or the *fault* state (red flag)
- Icon to display the process execution as a graph 



Id	Process	Iter	Instance	Start	End	State	Icon
34538	store	OrderIter	client:601 idOrder:136	13/06/07 13.08	13/06/07 13.08	Green	
34537	order	OrderIter	client:601 idOrder:136	13/06/07 13.08	13/06/07 13.08	Green	
34536	order	OrderIter	client:601 idOrder:135	13/06/07 13.01	13/06/07 13.01	Yellow	
34535	transfi			13/06/07 12.46	13/06/07 12.46	Green	
34534	transfi			13/06/07 12.39	13/06/07 12.39	Yellow	
34533	transfi			13/06/07 12.37	13/06/07 12.37	Green	
34532	store	OrderIter	client:601 idOrder:133	13/06/07 12.35	13/06/07 12.35	Green	
34531	store	OrderIter	client:601 idOrder:132	13/06/07 12.35	13/06/07 12.35	Green	
34530	order	OrderIter	client:601 idOrder:133	13/06/07 12.35	13/06/07 12.35	Green	
34529	order	OrderIter	client:601 idOrder:132	13/06/07 12.35	13/06/07 12.35	Green	
34528	store	OrderIter	client:601 idOrder:131	13/06/07 12.29	13/06/07 12.29	Yellow	
34527	order	OrderIter	client:601 idOrder:131	13/06/07 12.29	13/06/07 12.29	Green	
34525	store	OrderIter	client:601 idOrder:130	13/06/07 12.22	13/06/07 12.22	Green	

On top of the right side of the page, you can find, in order, the button that allows you to update the list, the button to visualize the *Advanced Search* page and the button to do massive restart of processes.

## 6.1 Search

It is possible to search the process instances by taxonomy or by iter, using the section on the left side of the page containing the list of the process instances.

### 6.1.1 Search by taxonomy

Spagic Studio offers the possibility of cataloguing by taxonomy a process published on the database.

When you choose a taxonomy from the taxonomies tree (the taxonomies are the folders), the classifications that the taxonomy contains are being displayed.

When you choose a classification, the instances list is being updated and will contain only the instances belonging to that classification.

### 6.1.2 Search by iter

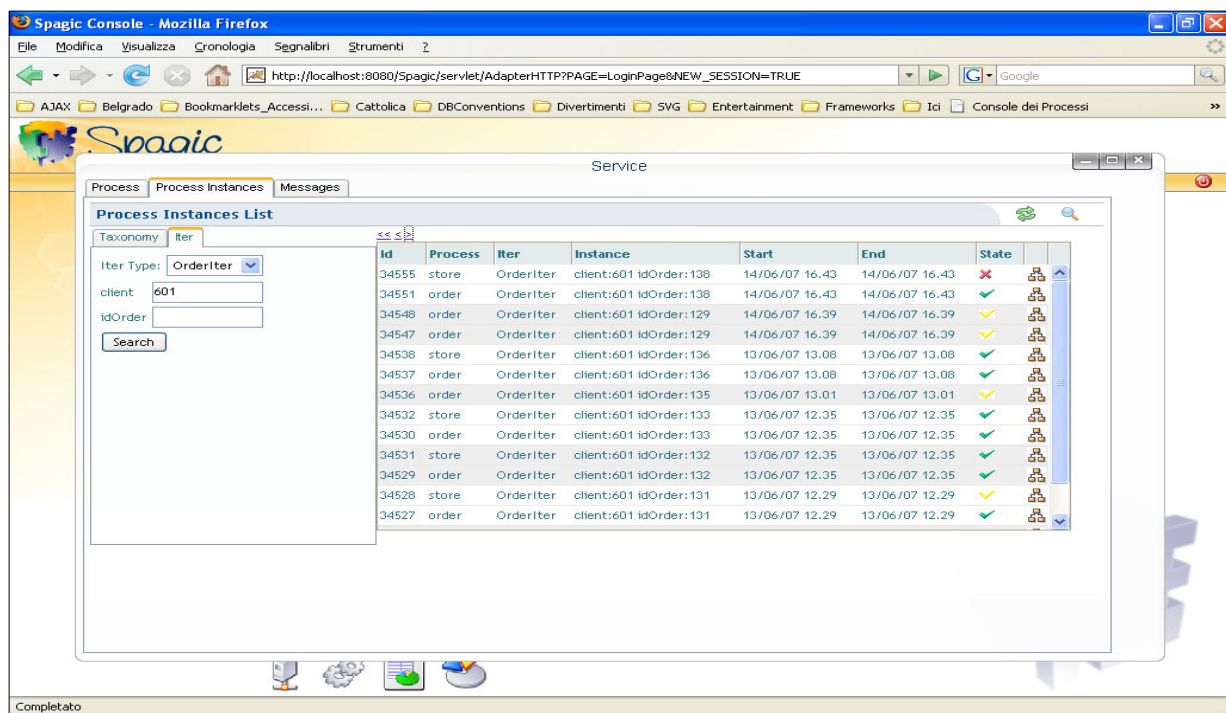
It is possible to make a search by iter, choosing the *Iter Type* combo. The list is automatically updated when you choose an element of the combo.

If you choose "Any Iter", all the instances will be displayed.

If you choose "No Iter", only the instances with no iter associated will be displayed.


If you choose a specific iter, the instances associated with that iter will be displayed; in this case the instances are order by iter instance and not by *Start* of execution. The instances are grouped by iter instance, using the alternate background color.

When you choose an iter, you can specify the search using the attributes dynamically displayed under the *Iter Type* combo. The search is a LIKE search.



Id	Process	Iter	Instance	Start	End	State
34555	store	OrderIter	client:601 idOrder:138	14/06/07 16.43	14/06/07 16.43	✗
34551	order	OrderIter	client:601 idOrder:138	14/06/07 16.43	14/06/07 16.43	✓
34548	order	OrderIter	client:601 idOrder:129	14/06/07 16.39	14/06/07 16.39	✓
34547	order	OrderIter	client:601 idOrder:129	14/06/07 16.39	14/06/07 16.39	✓
34538	store	OrderIter	client:601 idOrder:136	13/06/07 13.08	13/06/07 13.08	✓
34537	order	OrderIter	client:601 idOrder:136	13/06/07 13.08	13/06/07 13.08	✓
34536	order	OrderIter	client:601 idOrder:135	13/06/07 13.01	13/06/07 13.01	✓
34532	store	OrderIter	client:601 idOrder:133	13/06/07 12.35	13/06/07 12.35	✓
34530	order	OrderIter	client:601 idOrder:133	13/06/07 12.35	13/06/07 12.35	✓
34531	store	OrderIter	client:601 idOrder:132	13/06/07 12.35	13/06/07 12.35	✓
34529	order	OrderIter	client:601 idOrder:132	13/06/07 12.35	13/06/07 12.35	✓
34528	store	OrderIter	client:601 idOrder:131	13/06/07 12.29	13/06/07 12.29	✓
34527	order	OrderIter	client:601 idOrder:131	13/06/07 12.29	13/06/07 12.29	✓

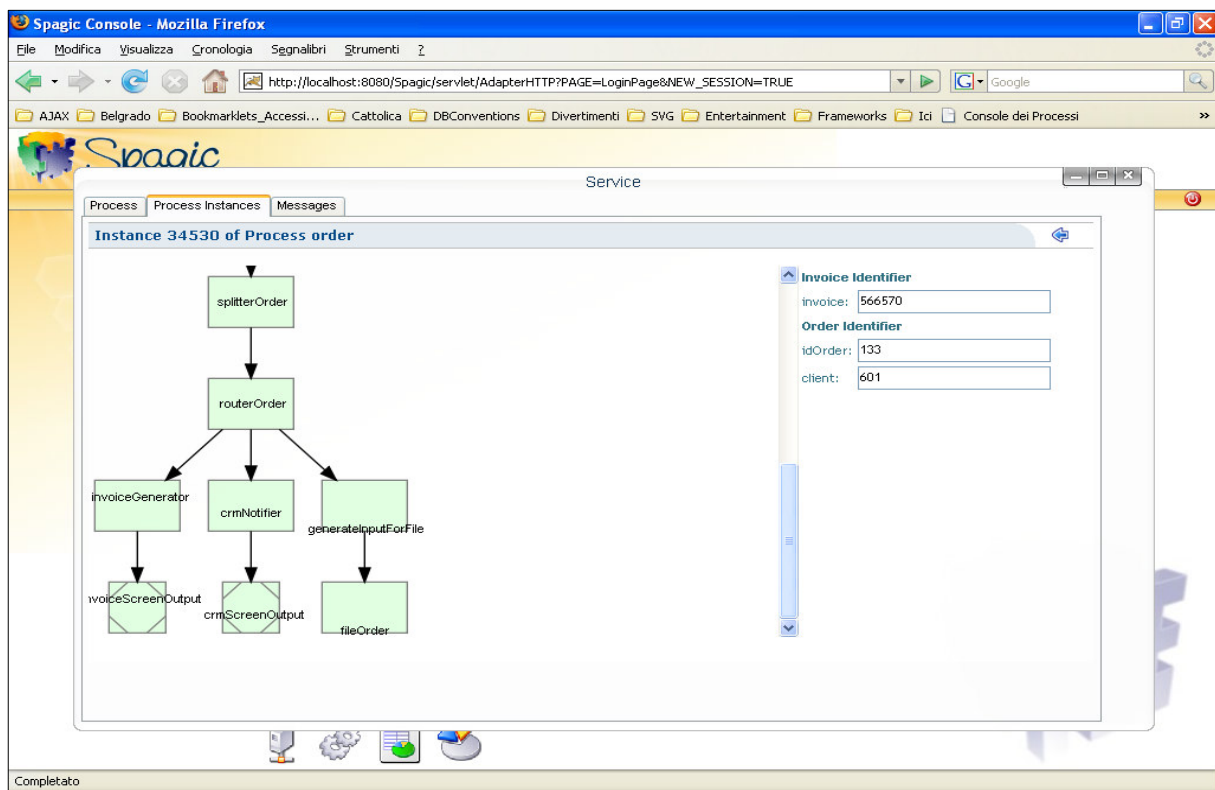
## 6.2 Graphical representation of the process execution

Choosing the icon  from the instances list, it is possible to display the process execution graph.

The graph is composed by the binding components (represented by a rhombus) and the service engines (represented by a rectangle); the direction of the execution flow is represented by the arrows.

The color of the component represents the component state: the green color means *executed*, the yellow color means *active* and the red means *fault*.

On the right side of the page all the relevant data associate to the endpoints composing the process are visualized, group by the catalog. The relevant data are defined in Spagic Studio. On the example in the image below, we have two catalog Invoice *Identifier* and Order *Identifier* containing in order the relevant data: *invoice* and *idOrder*, *client*.



Clicking on the single element of the graph it's possible to display a window containing two tabs: *Details* and *Relevant Data*. Using the mouse you can change the position of the window or its dimensions, you can minimize it or close it.

When you choose a new element of the graph, the data in the window are updated.

The *Detail* tab, as in the processes, contains *Name* and *Description* of the end-point and *Service*, that is the type of the used component (servicemix-lwcontainer XsltComponent, for instance) and *Properties*.

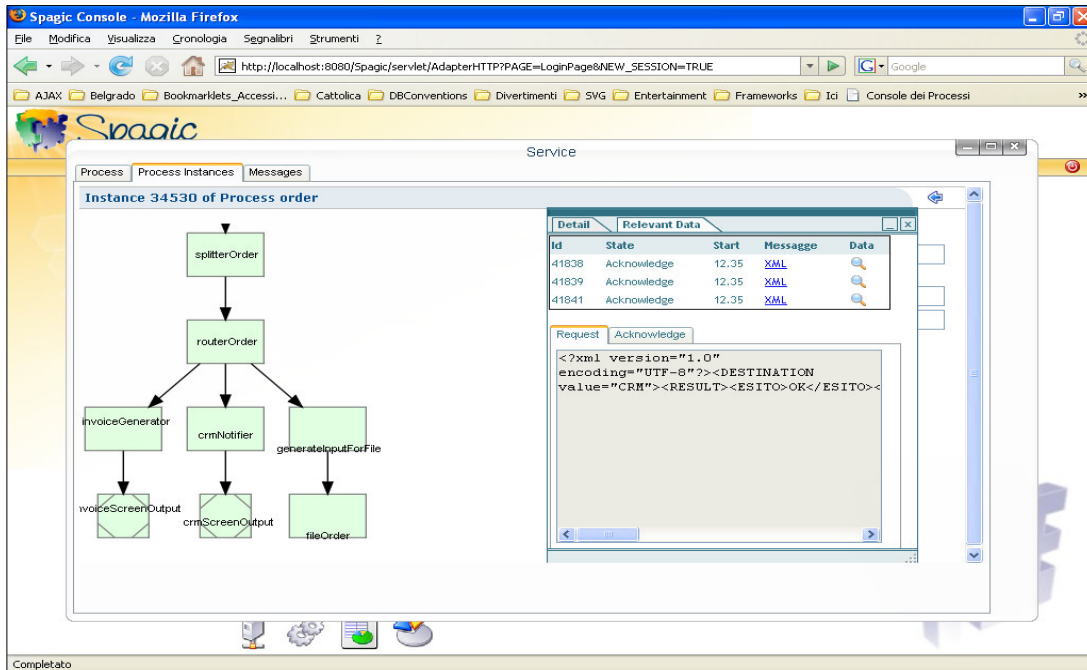
The *Relevant Data* tab contains the exchange messages having as target the component selected. The list display the *State*, the *Start* of the message reception, the link to visualize the XML message and the link to visualize the relevant data contained in the message.

On the example in the image below, by selecting the *routerOrder* component, three messages having as a target the *routerOrder* component are displayed.

By selecting the *XML* link, the *XML* message is visualized. Because is a mep INONLY, there are two parts:



INONLY\_ACTIVE message displayed as *Request* and INONLY\_DONE displayed as *Acknowledge*.



The screenshot shows the Spagic Console interface in Mozilla Firefox. The main window displays a process flow diagram for 'Instance 34530 of Process order'. The flow starts with a 'splitterOrder' component, followed by a 'routerOrder' component. The 'routerOrder' component branches into three parallel paths: 'invoiceGenerator', 'crmNotifier', and 'generateOutputForFile'. Each path leads to a corresponding output component: 'invoiceScreenOutput', 'crmScreenOutput', and 'fileOrder' respectively. On the right side, a 'Detail' window is open, showing a table of messages. The table has columns for 'Id', 'State', 'Start', 'Message', and 'Data'. The messages listed are:

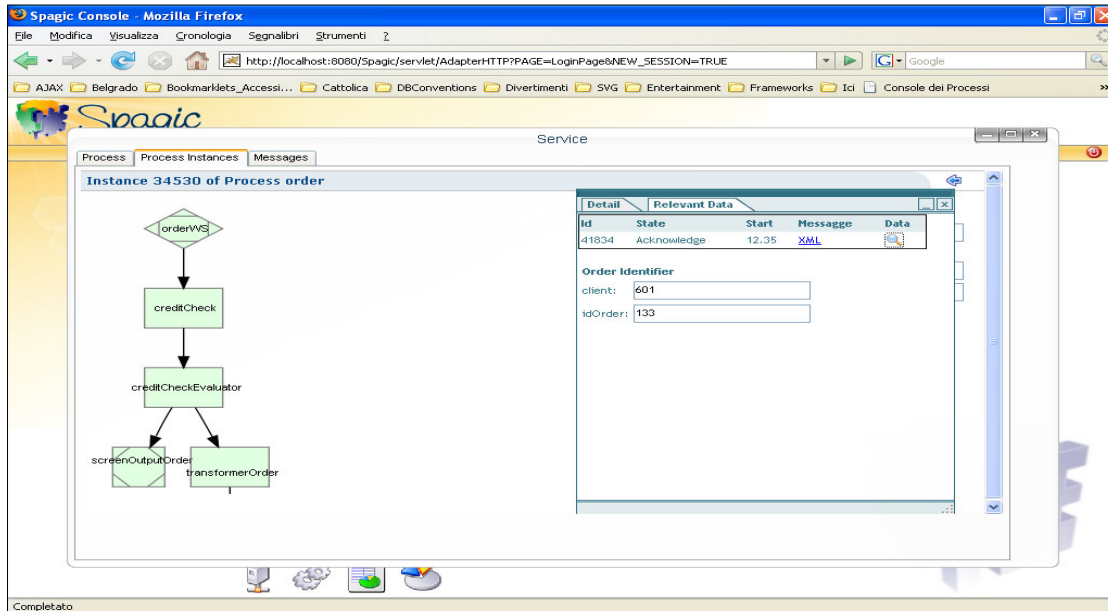
Id	State	Start	Message	Data
41838	Acknowledge	12.35	<a href="#">XML</a>	
41839	Acknowledge	12.35	<a href="#">XML</a>	
41841	Acknowledge	12.35	<a href="#">XML</a>	

Below the table, there is a 'Request' tab showing an XML message:

```
<?xml version="1.0"
encoding="UTF-8"?><DESTINATION
value="CRM"><RESULT><ESITO>OK</ESITO></
```

On the example in the image below, by selecting the *creditCheck* component, the messages list is refreshed.

By clicking on *Data*, the relevant data contained in the message are displayed.



The screenshot shows the Spagic Console interface in Mozilla Firefox. The main window displays a process flow diagram for 'Instance 34530 of Process order'. The flow starts with a decision component 'orderVVS', followed by a 'creditCheck' component, then a 'creditCheckEvaluator' component. The 'creditCheckEvaluator' component branches into two parallel paths: 'screenOutputOrder' and 'transformerOrder'. On the right side, a 'Detail' window is open, showing a table of messages. The table has columns for 'Id', 'State', 'Start', 'Message', and 'Data'. The messages listed are:

Id	State	Start	Message	Data
41834	Acknowledge	12.35	<a href="#">XML</a>	

Below the table, there is a 'Data' tab showing the relevant data contained in the message:

```
Order Identifier
client: 601
idOrder: 133
```

## 6.3 Single restart

This function allows restarting the execution of a process in the *Fault* state. Only process instances in the fault state (visualized in the list with a red flag) can be restarted.


The restart functionality creates a new instance that is equal to the previous until to the component that generated the error, and then it is a new execution of the process.

Before restarting a process, it is possible to modify manually the exchange messages in the *Fault* state performing the following steps:

1. Selection of the component that generated the error (visualized with the red color), in the graph representing the process execution;
2. Selection of the *Relevant Data* tab;
3. Selection of the message in the *Error* state;

The message detail will be displayed. On the example in the image below, the message is composed by two parts: the *Request* and the *Error*.

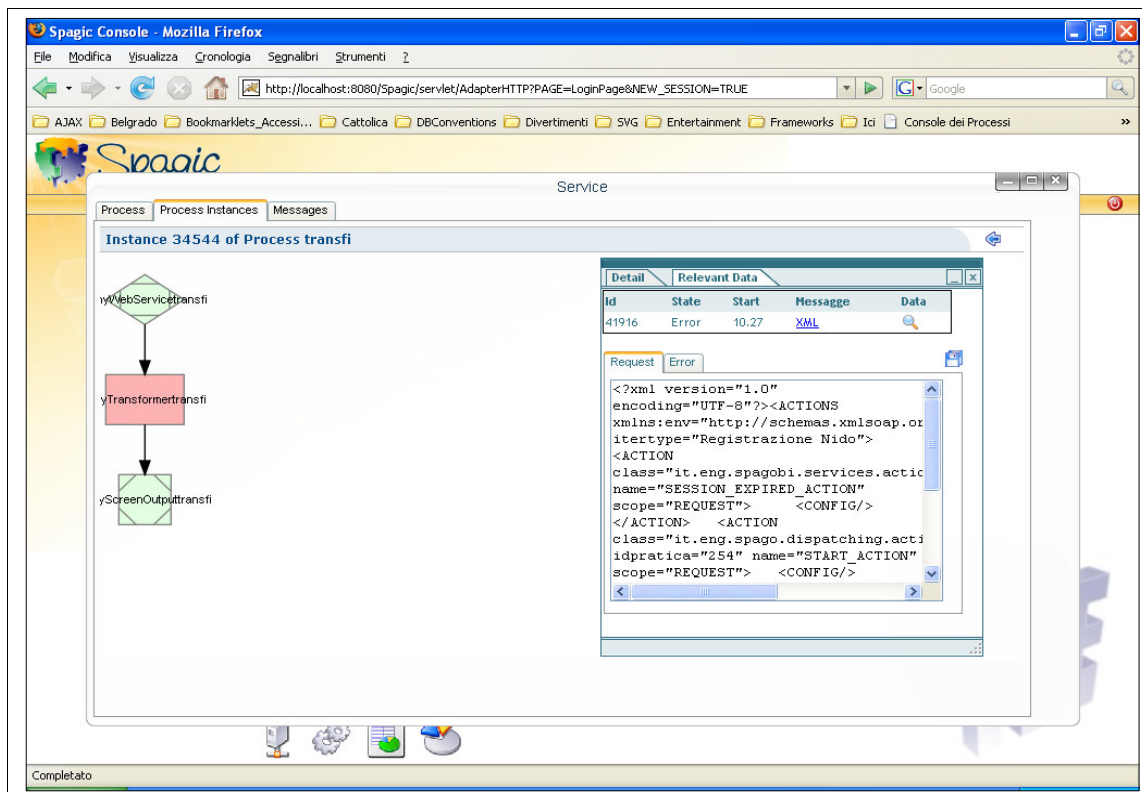
Only the *Request* message can be modified: to modify the message, select the *Request* tab, edit the content and save it clicking *Save* button. It is possible to modify more messages for a unique execution (for example all messages having as source the splitter component).

To restart the execution, select, the icon  *restart* on the top right side. If no problems occur, the page is reloaded with the new instance.



Usually when the restarted instance is displayed, it's still executing, so you will see some services in *Executing* state (with yellow color).

It is possible re-starting a failed process more than one time.



Id	State	Start	Message	Data
41916	Error	10.27	XML	

```

<?xml version="1.0"
encoding="UTF-8"><ACTIONS
xmlns:env="http://schemas.xmlsoap.org
itertype="Registrazione Nido">
<ACTION
class="it.eng.spagobi.services.actio
name="SESSION_EXPIRED_ACTION"
scope="REQUEST">
</ACTION>
<ACTION
class="it.eng.spago.dispatching.acti
idpratica="254" name="START_ACTION"
scope="REQUEST">
</ACTION>

```



On the *Process Instances* tab, the restarted instances are highlighted with the lightning icon and usually they are presented after the original process (the faulted one).

Process Definitions Process Instances Messages		
Process Instances List		
Taxonomy	Iter	
All processes		
Business		
Clients		
Suppliers		
	Id	Process
	Iter	Instance
	Start	End
	State	
	34846	store
	34845	store
	34844	order
	34843	store
	34842	order
	34841	order
	34811	SampleJdbcForRestart
	34803	transfi
	34807	transfi
	34813	transfi
	34802	transfi
	34789	SampleJdbcForRestart
	34790	SampleJdbcForRestart

### 6.3.1 Known limitations

There are some limitations to the processes that you can restart: for example the following process cannot be restarted, if it fails, because it involves only a single exchange whose type is In-Out.



If we try to restart this process, the exchange will be resent, but when the HTTP Binding component receives the response from the Transformer component, it doesn't know what to do with this response because there is no a client that invoked it, to which sending that response.

For this reason, if the process contains some exchanges of type In-Out, the restarted process may be not fully equivalent to a "normal executed" process: all responses to In-Out exchanges will be not managed by the components that receives them.

## 7 Messages Displaying

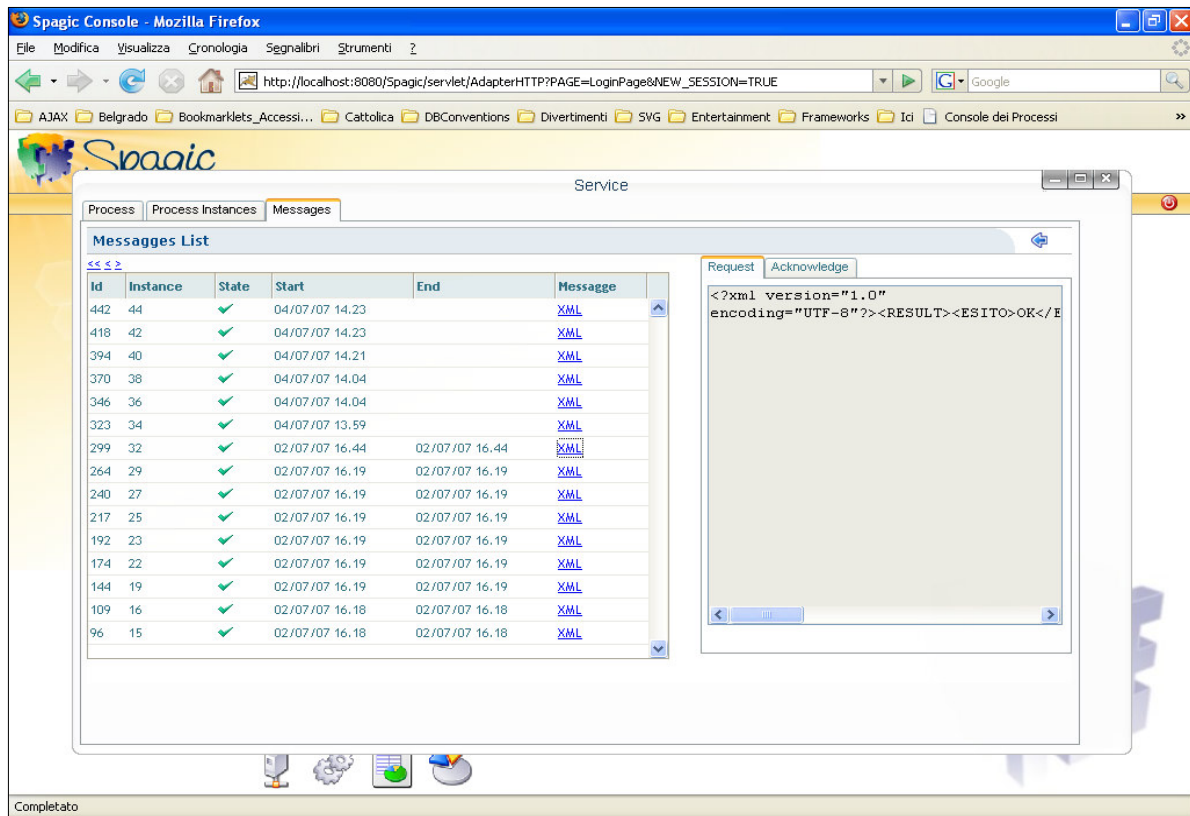
Spagic Console allows visualizing all the exchange messages by selecting a component belonging to a process. By pressing the **Service** icon and the *Messages* tab, a filter page is being displayed.

The page contains:

- *Process*: to select a process between those published in the MetaDB. By selecting the process, will be displayed:
  - The *Component* field containing all components belonging to the process,
  - All attributes associates to endpoints composing it;
- *Component*;
- *From To*: with the calendar it is possible to select year, month, day, hours and minutes;
- *State* containing the possible state relative to process instance: *Executing*, *Executed*, *Fault*;


By clicking the button *Search* a list of exchange messages will be display.

By selecting the *XML* link in the elements of the list, the messages composing the exchange messages will be visualized on the right side.



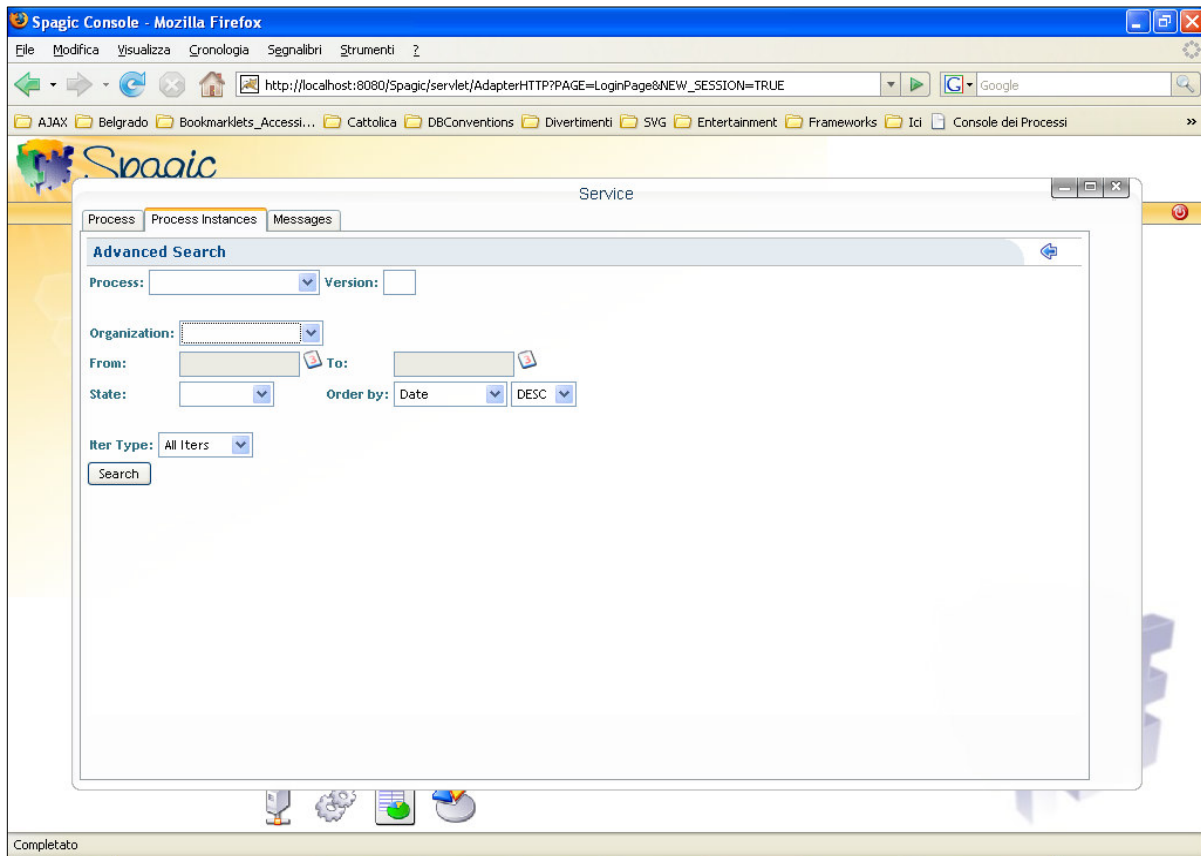
The screenshot shows the Spagic Console interface in a Mozilla Firefox browser. The main window displays the 'Service' tab with a 'Messages' sub-tab selected. On the left, there is a 'Messages List' table with columns: Id, Instance, State, Start, End, and Message. The table contains 18 rows of data, each with a green checkmark in the 'State' column and a blue 'XML' link in the 'Message' column. On the right, there is a panel with two tabs: 'Request' and 'Acknowledge'. The 'Request' tab is active, showing an XML message body: `<?xml version="1.0" encoding="UTF-8"?><RESULT><ESITO>OK</ESITO></RESULT></xml>`.

## 8 Advanced Search

Selecting the icon  *Advanced Search* in the *Process Instances* list, on the top of the right page, the *Advanced Search* page is displayed.

The page contains:

- *Process* to select a process between those published in the MetaDB. By selecting the process, will be displayed:
  - The *Version* field containing the last version published in the database. Modifying it, it is possible to visualize to process instances relative to previous versions.
  - All attributes associates to endpoints composing it;
- *Organization* contains all organizations defined in the service registry;
- *From To*: with the calendar it is possible to select year, month, day, hours and minutes;
- *State* containing the possible state: *Executing*, *Executed*, *Fault*;
- *Order by* follows to order the list by fields: *Date*, *Organization*, *Process* (ASC or DESC)
- *Iter Type* follows to select an iter and the attributes that identify it




The screenshot shows the Spagic Console interface in Mozilla Firefox. The browser's address bar displays the URL: `http://localhost:8080/Spagic/servlet/AdapterHTTP?PAGE=LoginPage&NEW_SESSION=TRUE`. The page title is "Spagic Console - Mozilla Firefox". The main content area is titled "Service" and contains a tabbed interface with "Process", "Process Instances", and "Messages". The "Process Instances" tab is active, showing the "Advanced Search" form. The form includes the following fields and controls:


- Process:** A dropdown menu.
- Version:** A text input field.
- Organization:** A dropdown menu.
- From:** A date/time picker.
- To:** A date/time picker.
- State:** A dropdown menu.
- Order by:** A dropdown menu with "Date" selected, and a "DESC" button.
- Iter Type:** A dropdown menu with "All Iters" selected.
- Search:** A button to execute the search.

The bottom of the browser window shows a status bar with the text "Completato" and several icons.

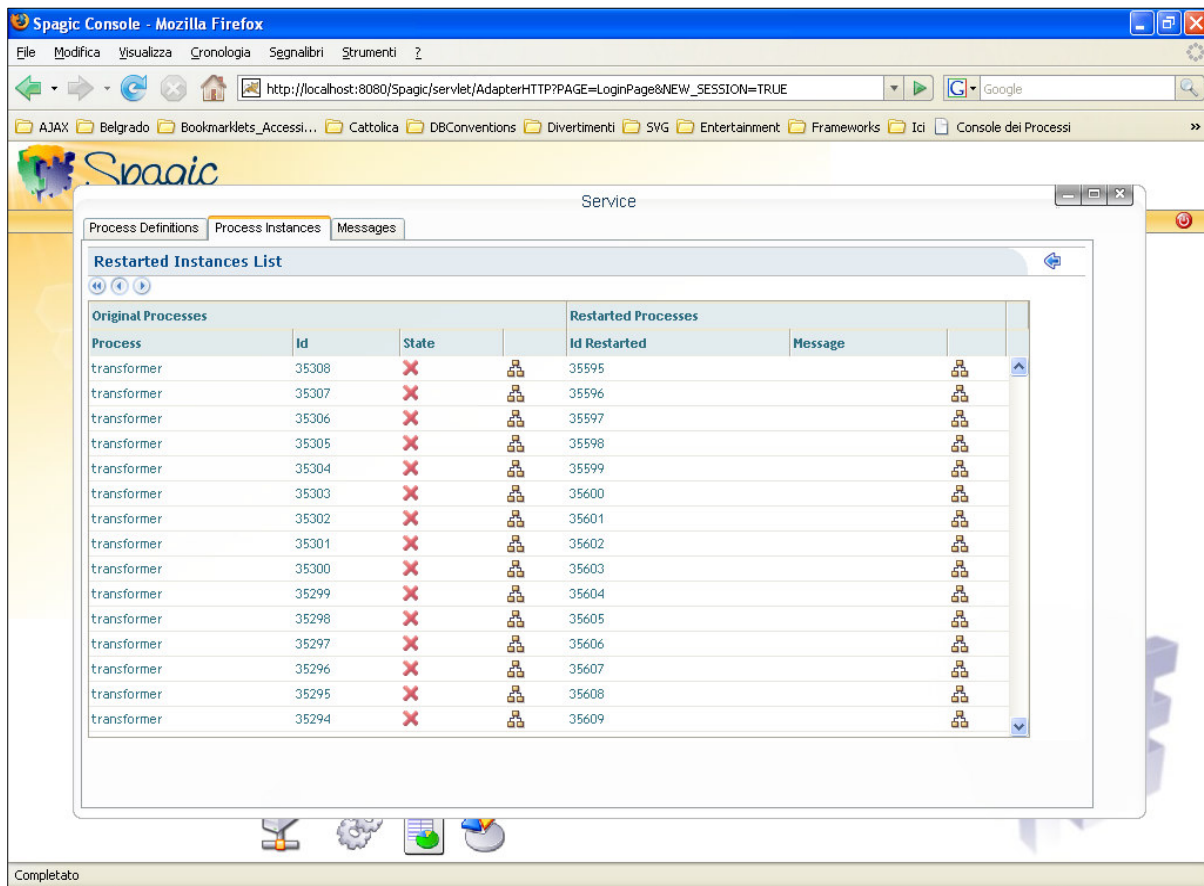
## 9 Massive restart

The massive restart is an extension of the single restart functionality; it allows restarting more processes in the *Fault* state, in a single step. It doesn't allow modifying the message exchanges.

To use the *Massive Restart* feature, select  icon on top-right corner of the page *Process Instances*: a filter page will be displayed.

By clicking the *Search* button, a process instances list will be visualized: choosing the icon , the massive restart process will start.

When the process terminate, a page containing the result of process execution will be displayed (see an example in the image below) .




The screenshot shows the Spagic Console interface in a Mozilla Firefox browser. The 'Process Instances' tab is active, and a 'Restarted Instances List' dialog is open. The dialog contains two tables:

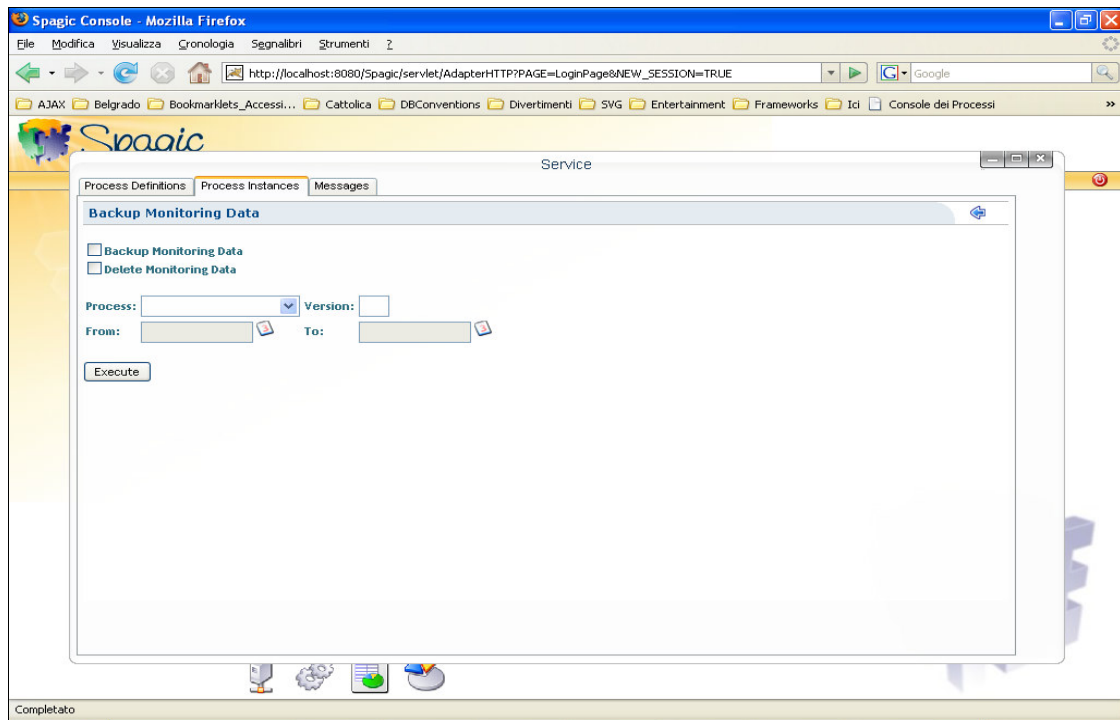
Original Processes			Restarted Processes	
Process	Id	State	Id Restarted	Message
transformer	35308	×	35595	
transformer	35307	×	35596	
transformer	35306	×	35597	
transformer	35305	×	35598	
transformer	35304	×	35599	
transformer	35303	×	35600	
transformer	35302	×	35601	
transformer	35301	×	35602	
transformer	35300	×	35603	
transformer	35299	×	35604	
transformer	35298	×	35605	
transformer	35297	×	35606	
transformer	35296	×	35607	
transformer	35295	×	35608	
transformer	35294	×	35609	

The page displays a list containing some information about the original process instances(process name, instance id, state, link to visualize the execution graph) and the restarted process instances (instance id, link to visualize the execution graph).

## 10 Backup and Delete Monitoring Data

Through the console, it's possible to execute the monitoring data backup into a different repository. It is also possible to delete, at the same time or subsequently, the monitoring data from the current repository.

To use these features, select the *icon*  on top-right corner of the page *Process Instances*: a filter page will be displayed.



Check the box *Backup Monitoring Data* and/or *Delete Monitoring Data*, set the filter parameters and then click the *Execute* button: the backup and/or the delete process will start; at the finish a message about the execution will be displayed.

If the backup process terminates with errors, the delete process will not be executed.