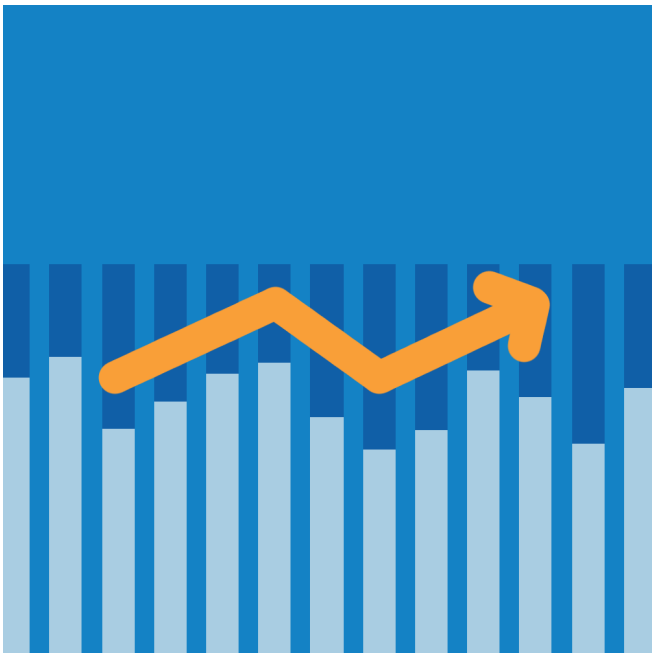


Alflytics

Open Source Business Intelligence solution over Alfresco

Version 5.0 Early Access

https://github.com/fcorti/Alflytics

Latest revision: 26 August 2017

Index

1. Introducing Alflytics

Alflytics is the Open Source Business Intelligence (OSBI) solution over Alfresco content and processes, built using the Pentaho platform. With Alflytics is provided a pure Business Intelligence solution to extract, store and enquiry content data (documents, folders, metadata, tags, categories, etc.), audit trail and workflows at a very detailed level, with the goal to be easily customized and extended with other entities coming from external Services like: ERP, CRM, custom Applications, etc.

Alflytics uses the data coming from Alfresco platform, the ability of the Pentaho platform to build reports, dashboards and free analysis, and the latest techniques in Data Warehousing, to define a complete set of analytics and insights on top of your preferred ECM+BPM solution.

1.1 About the project

Alflytics is invented, designed, developed and maintained by Francesco Corti, Developer and member of the Alfresco and Pentaho communities. Below some high level characteristics, defining the solution, to better understand the project and its nature.

1.1.1 Open Source

Alflytics is purely an **Open Source** project. This means that every source code, script or configuration is available and given as included in the release/distribution. Also the frameworks and platforms used by Alflytics are Open Source, even if the precise details depend on the Vendor and the Owner of the solution itself.

1.1.2 Free of charge

Alflytics is **free of charge** and not related to number of user, dimension of the repositories or any other detail of the final solution.

1.1.3 License

Alflytics is released under the **GNU Lesser General Public License (LGPL) Version 2.1**. This license has been selected to make easier the adoption of the solution from final users and Companies interested to use Alflytics in a vertical solution, for specific needs or specific use cases.

1.1.4 Goals

The goal of Alflytics is to become more adopted and make it **the most reliable Business Intelligence solution** over the entire Alfresco platform. To reach that goal we kindly ask you to contribute to the project (as described into one of the following chapters).

1.2 The history

The origins of Alflytics as a project are in a similar project called Alfresco Audit Analysis and Reporting (A.A.A.R.). A.A.A.R. was born in 2012, when Francesco Corti was talking with a bunch of Customers and all of them were asking for more analytics capabilities on Alfresco. In that period Alfresco didn't cover that topic at all, even if the interest around reporting where high. At that time, Francesco was working with Pentaho Suite and had the idea to experiment an integration between the two frameworks (in 2012 both Alfresco and Pentaho were emerging Open Source frameworks in their respective fields).

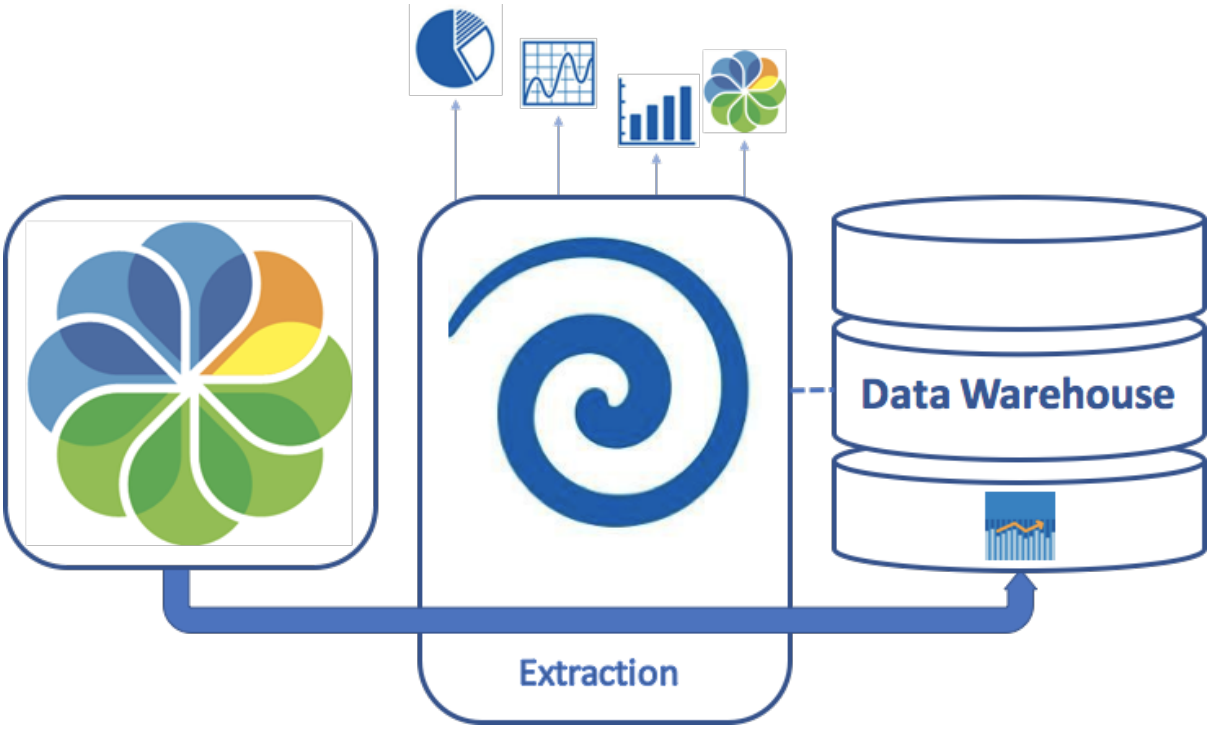
In 2013 Francesco released the very first version of A.A.A.R. and decided to show it at the Alfresco Summit in Barcelona. The same year, the solution has been presented to the Pentaho Community Meeting and few months later it has been published as one of the first addons in the Pentaho Marketplace. Since then, each four/six months a new release has been published, with new features and enhancements and the solution has been regularly presented in public initiatives in both the Alfresco and Pentaho communities.

In 2017, during the complete refactoring of the fifth version of the solution, A.A.A.R. has been renamed in Alflytics: with the same idea (still considered a success as solution), a new logo and of course new enhancements.

Today Alflytics is a mature and complete Business Intelligence solution over Alfresco platform and continue to be (proudly) a community project, totally (and really) Open Source.

1.3 Overview of the architecture

One key concept of the solution is that is built as an integration of existing platforms (Alfresco and Pentaho), instead of “reinvent the wheel” and everything from scratch, one more time. The main idea is that Alflytics: 1) extracts audit data, content repository and processes from Alfresco, 2) stores the data in a Warehouse (a database), 3) creates a set of analytics on top of the Data Warehouse using the Pentaho platform. Optionally, static versions of reports can be published into Alfresco in well-known formats (pdf, xlsx, csv, etc.).



1.4 The components

As a Pentaho addon, Alflytics uses different components with the goal to request the lowest effort for the developers and the best result for the users. Below a list of the possible components that can be used, with a brief description and some further details, for example if they are mandatory to used or optional.

- **Alfresco platform**. It is, at the same time, source and target for the flow. As source, Alfresco is enquired using the standard REST APIs to extract the full list of audit trails, content repository (documents and folders) and processes (with definitions, processes, tasks, items, etc.). As target for the flow, static versions of the reports can be published in the Alfresco folders, to be used by the final users according to the Alfresco permits and roles.

- **Pentaho Analytics platform**. This is the hosting web application where Alflytics really acts as a so called Sparkl Application. This component covers the core back-

end and front-end capabilities. It also stores: the ETL jobs, reports, dashboards and free analysis.

- **Alflytics data warehouse**. It is a pure database, storing the extracted data, organized in a specific manner, following the best practices in literature about Data Warehousing and Business Intelligence. From a physical point of view the Alflytics warehouse is a Relational DBMS composed by dimension tables and fact tables. Alflytics solution supports Hyper SQL, PostgreSQL, MySQL.

- **Saiku Analytics** (optional). It is a third party tool used to "pivot" data stored into the Alflytics data warehouse. Saiku Analytics let the final user able to execute the so called "free analysis" into real OLAP sessions and dive deep into the data in the way they will want.

- **Pentaho Reporting** (optional). It is a great tool of the Pentaho suite, used to develop Pentaho reports with an easy User Interface (called Pentaho Reporting Designer). Pentaho Reporting is used in the development environment only and let the developers able to customize and create nice reports on top of the Alflytics data warehouse.

- **Pentaho CTools** (optional). CTools are a set of tools and components created to help you build custom dashboards on top of Pentaho. There are specific CTools for different purposes. Pentaho CTools are used in development only and let the developers able to customize and create nice dashboards on top of the Alflytics data warehouse.

- **Pentaho Data Integration** (optional). It is a great tool of the Pentaho suite, used to manage the ETL jobs stored in the Pentaho repository (into the Pentaho Analytics platform). The Pentaho Data Integration tool is scalable from several points of view and gives to Aflytics the ability to scale on huge repositories and architecture. Pentaho Data Integration is not mandatory to be used. It is recommended for scalability only.

1.5 Compatibility matrix

Below the matrix to understand which Alflytics version to use depending on your Alfresco and Pentaho versions.

 Alfresco	 Pentaho 7.1
Community Edition 5.2	Alflytics 5.0

Content Services (ex One) 5.2	Alflytics 5.0

Previous versions and the compatibility matrix can be found in Appendix II (previous versions). We always recommend to move to the latest versions.

1.6 References and resources

Below some references to the Alflytics project.

- Source code: https://github.com/fcorti/Alflytics
- Sourceforge: https://sourceforge.net/projects/aaar/
- Documentation and official website: http://fcorti.com

2. How to use Alflytics

Alflytics is designed to be easy to use, easy to be customized/extended and ready to scale on huge repositories. In this section we are going to see how to use Alflytics and what does it means that its use it is easy.

Using Alflytics means doing the following tasks.

- **Extracting** data from the sources (by default Alfresco) into the Alflytics Data Warehouse. This task is mandatory to see the data into the Alflytics reports, dashboards and free analysis.

- **Publishing** static reports directly into your Alfresco platform with your preferred format (pdf, xlsx, txt, cvs, ecc.). You can avoid this task if you want to have interactive reporting.

- **Using analytics** on top of the Data Warehouse, using Pentaho and in particular: interactive reports, dashboards and free analysis.

2.1 Extracting from Alfresco

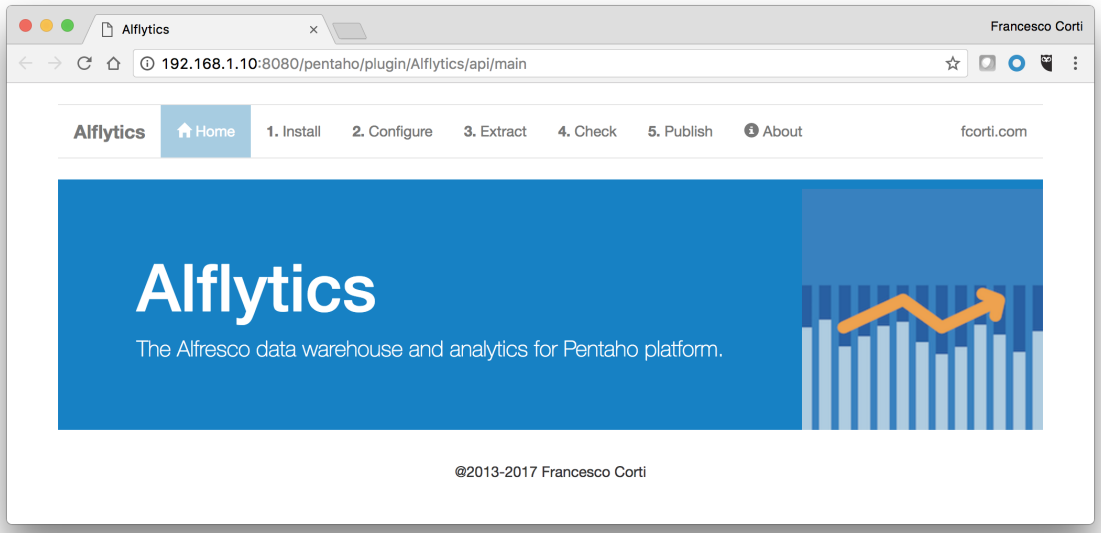
With the extraction task, a batch gets the data out from Alfresco into the Alflytics Data Warehouse. This task is mandatory to see the data into the interactive reports, dashboards and free analysis.

The extraction is requested through a web page, in a default installation, or executing a Pentaho Job if Pentaho Data Integration is used. In the following description, we will see how to extract data using a default installation and the Alflytics Administration Web pages.

The Alflytics Administration Web pages can be easily accessed using the following url, after the Alflytics solution is correctly installed (for further details on how to install Alflytics, check the related chapter).

```
http://<server>/pentaho/plugin/Alflytics/api/main
```

In the following screenshot, the Alflytics Administration Home page, used as starting point to manage the whole instance of Alflytics.



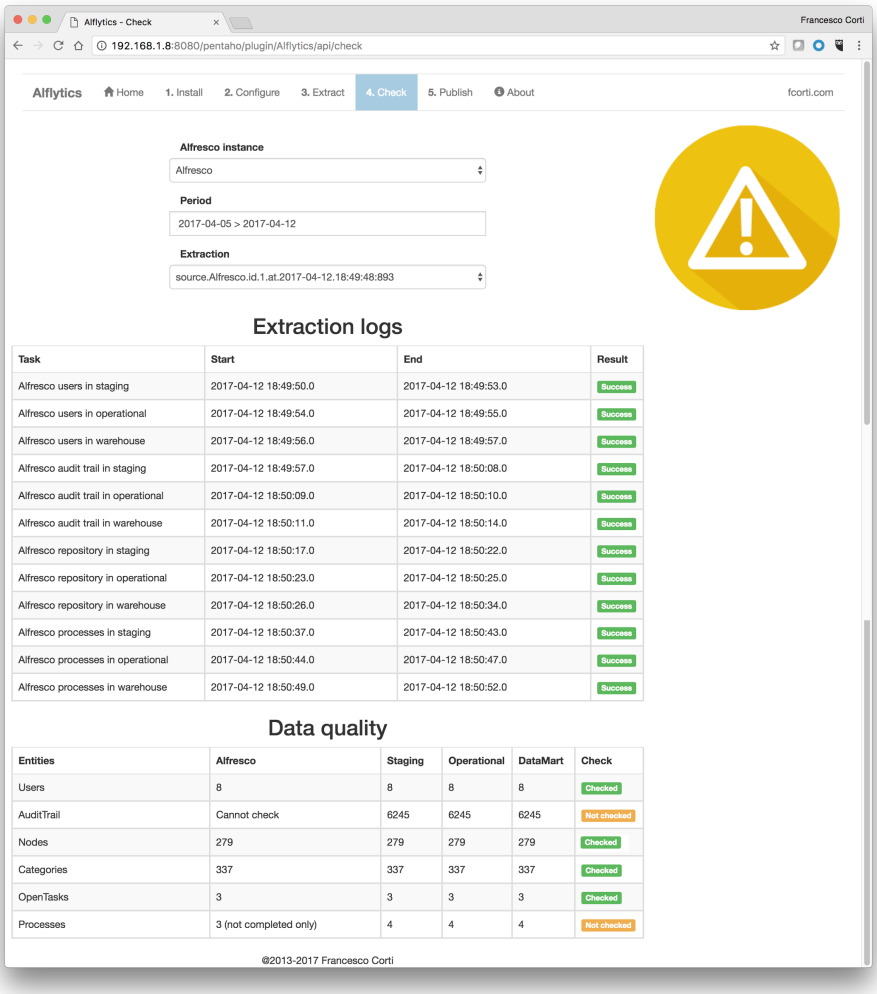
If you take a look at the upper menu, the item called `3.Extract` is the one used to get data from Alfresco. Click on it, to have access to the extraction page. As you can see directly in the target page, the extraction is very straightforward and does not require any difficult setup or question. Simply fill the Pentaho administrator login and password (by default `admin` and `password`) and click the `Extract` button to start the process.

The `Extract` button only starts the extraction process, that could require few seconds or some minutes to be completed. If the web page replies with a result, this does NOT mean the extraction process is finished. The duration of the extraction process depends (mainly) on the Alfresco repository and the network. Always check the `catalina.out` file stored into the `<pentaho-server>/tomcat/logs`, for further details on the extraction task.

Once the extraction task is finished, you can check how the extraction went, using the Alflytics Data Quality dashboard. You can access to the Alflytics Data Quality dashboard using the upper menu of the Administration Web pages. More in particular clicking the `4.Check` item.

The Alflytics Data Quality web page is available to check how the extractions went, with a easy to understand dashboards. In the dashboard you can select the Alfresco instance, the

period to filter extractions and each extraction from an identifier, quite easy to read because based on a timestamp. Once the extraction id is selected, two tables show you the duration of the sub-extractions (Alfresco users, audits repository and processes) with their success/failure and a detail of the number of entities extracted. Below a screenshot showing how the Alflytics Data Quality web page look like.



You can check into this Alflytics Data Quality web page every time you want to understand how the extraction task went, also if the extraction is scheduled as an automatic job.

2.2 Publishing reports into Alfresco (optional)

Once the data are updated into the Alflytics Data Warehouse, you can decide to use the analytics (see the following paragraph) or publish a static version of your reports (usually in PDF or Excel format), directly into the Alfresco repository.

The publication of the reports can be done using a web interface or a command, launched from the terminal. The launch of the publication process from the web interface, is again very

straightforward using the `5.Publish` item of the upper menu in the Alflytics Administration Home page.

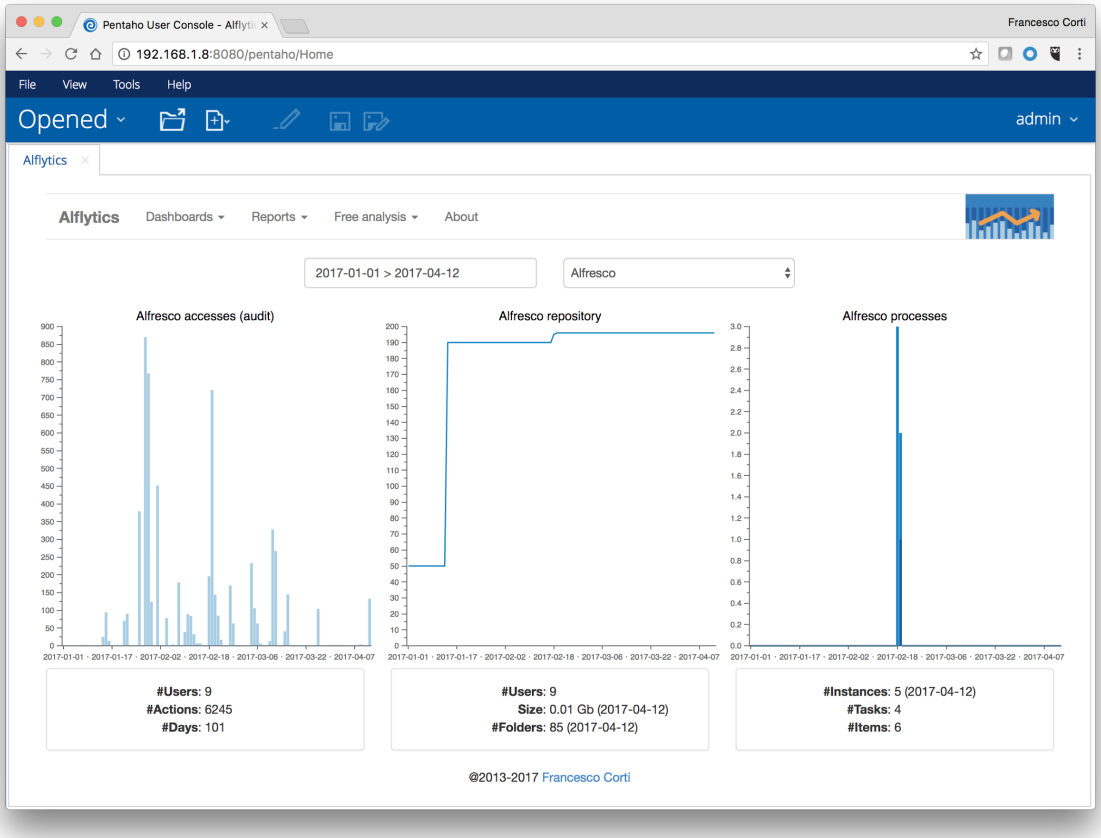
Also in this case, the publication button only starts the process and you should always check the `catalina.out` file stored into the `<pentaho-server>/tomcat/logs`, for further details on the task.

Publishing reports require two things: the first is to enable the FTP services on Alfresco and the second is to correctly setup the connection parameters into the `DWH_REPORTS` table. To enable FTP on Alfresco add `ftp.enable=true` in `alfresco-global.properties` and restart it. To setup the setup the connection parameters into the `DWH_REPORTS` table, use your preferred DB client and update the fields of the table. Everything is extremely straightforward to do and understand.

2.3 Using analytics

Using a real Business Application you can benefit of several advantages like, for example, having a full featured platform available for your needs. Alflytics is released with a set of dashboards, interactive reports and free analysis, but these are only examples a solid starting point. You can easily build your own analytics tools, to better cover the use cases.

The starting point of you discovery is the main dashboard. The main dashboard is available using the Pentaho User Console, more in particular clicking the upper menu at the item `Tools`, and then `Alflytics`. Below a screenshot showing how the dashboard looks like.



If you want to integrate this dashboard into an external application or use it without the Pentaho User Console, you can have direct access using the following URL.

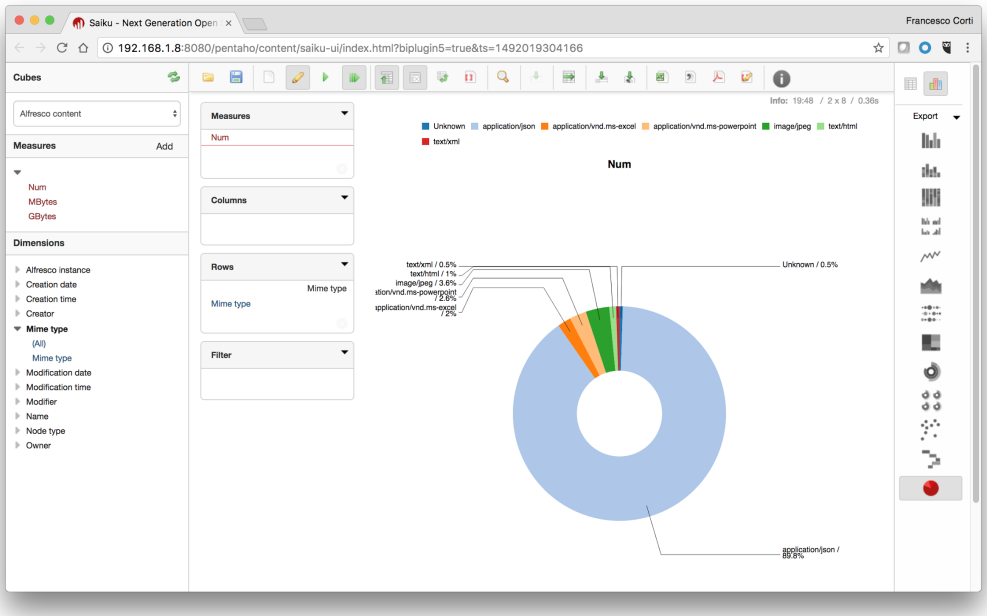
```
http://<server>/pentaho/api/repos/%3Apublic%3AAlflytics%3ADashboards
%3Amain.wcdf/generatedContent
```

Directly into the dashboard you can see an upper menu, listing also reports and free analysis. Below a list of each available report and free analysis, as a first introduction.

- Reports
  - Alfresco users
    - Most active users
    - Audit per user (Excel)
    - Repository per creator
    - Repository per modifier
    - Repository per owner
  - Alfresco audit trail
    - Audit per actions
    - Most accessed content
    - Audit per day
    - Audit per hour
    - Audit trail (Excel)

  - Alfresco repository
    - Repository summary
    - Most used categories
    - Most used tags
    - Repository per category
    - Repository per tag
    - Repository per node type
    - Content per mime type
    - Creations per day
    - Modifications per day
  - Alfresco processes
    - Process instances
    - Process tasks

- Free analysis (available only with Saiku Analytics installed)
  - Alfresco audit trail
    - Alfresco audit trail (alfresco-access)
  - Alfresco repository
    - Alfresco content
    - Alfresco folders
    - Alfresco categories
    - Alfresco tags
  - Alfresco processes
    - Alfresco process instances
    - Alfresco process tasks



Please remember all the analytics are available for customizations and copies. This means the variety of analytics don't begin and end with this set, but can be improved (and customizes) as much as your fantasy will require.

3. How to install Alflytics

Installing Alflytics is very straightforward because it is released as a so called "Pentaho Application", containing everything is requested to correctly configure and use the solution. No installations or configurations are requested on the Alfresco side and it can work on top of Pentaho Analytics platform, with an HyperSQL, PostgreSql or MySql database. In this section we are going to described how to install it, using the wizard (simplest) or manually (advanced), depending on your needs.

3.1 Preparing the installation

As discussed above, Alflytics is a Pentaho Application developed using Sparkl (the application builder). In this section we are going to describe the setup of the working environment, to prepare the real installation of the Alflytics application.

3.1.1 Minimum requirements

Being a Pentaho Application deployed into the Pentaho Analytics platform, Alflytics minimum requirements are strictly related to the Pentaho version. Alflytics 5.0 in particular, runs on the Pentaho 7.0 platform and for this reason the minimum requirements are explicitly described into the Pentaho's official documentation and copied below as reference.

Hardware

Processor:	Intel EM64T or AMD64 Dual-Core
RAM:	8 GB with 4 GB dedicated to Pentaho servers
Disk Space:	20 GB free after installation (Data Warehouse is not included and can vary)

Operating System (64 bit)

Ubuntu Server 14.04 LTS & 16.04 LTS
CentOS 6 & 7
Red Hat Enterprise 6 & 7
SUSE Linux SLES 11 (SP3+)
Microsoft Windows 2008 Server R2 & 2012 Server R2

Alflytics is developed (and mainly tested) using a Linux platform, in particular Ubuntu Server 16.04 LTS. The adoption of a Linux based Operating System is highly recommended

(because experienced as more stable) even there aren't reasons why the solution should not work on a Windows based OS.

3.1.2 Installing Pentaho Analytics platform

Installing Pentaho Analytics platform is very straightforward, very close to an "unzip" command of the distribution package. If you'd prefer a step by step tutorial, a simple blog post is what you need.

The only prerequisite for installing the Pentaho Analytics platform is having Java 8 on board. If you don't have it already or you don't how to install it, you can easily ask Google. It is a very basic task and there are a lot of tutorials in the we for every Operating System and distribution. If you are lazy to search, this tutorial is what can help you for a Linux platform.

After Java 8 is available into your environment you can go ahead downloading the Pentaho Analytics platform 7 from the official website or the sourceforge web page. In our case, we are going to install the Pentaho Analytics platform 7 Community Edition.

Once the `pentaho-server-ce-7.1.0.0-12.zip` file is downloaded, you can unzip it into the desktop or everywhere else you will like (usually the `/opt` path is suggested for Linux Operating System). All the Pentaho Analytics platform is now available in the `pentaho-server` folder.

Probably you cannot believe it, but this is enough to install Pentaho Analytics platform 7 into your environment.

Before considering completed this task, you should consider to reduce the default RAM, Pentaho is going to use. By default the Pentaho platform starts with 6Gb of RAM and in most cases is more than enough. If you don't have so much RAM into your environment, or you want more, you can tune it editing the `start-pentaho.sh`/`start-pentaho.bat` script, depending on your Operating System.

Pentaho Analytics platform 7 is mainly a web application deployed in a tomcat servlet container. You can easily run the Pentaho Analytics platform 7, opening a terminal and executing the `start-pentaho` script stored in the `pentaho-server` folder (the one with the `sh` extension for the Linux based Operating Systems and the one with the `bat` extension for the Windows based Operating Systems).

To check what is happening always look at the log file described below.

`<pentaho-server>/tomcat/logs/catalina.out`

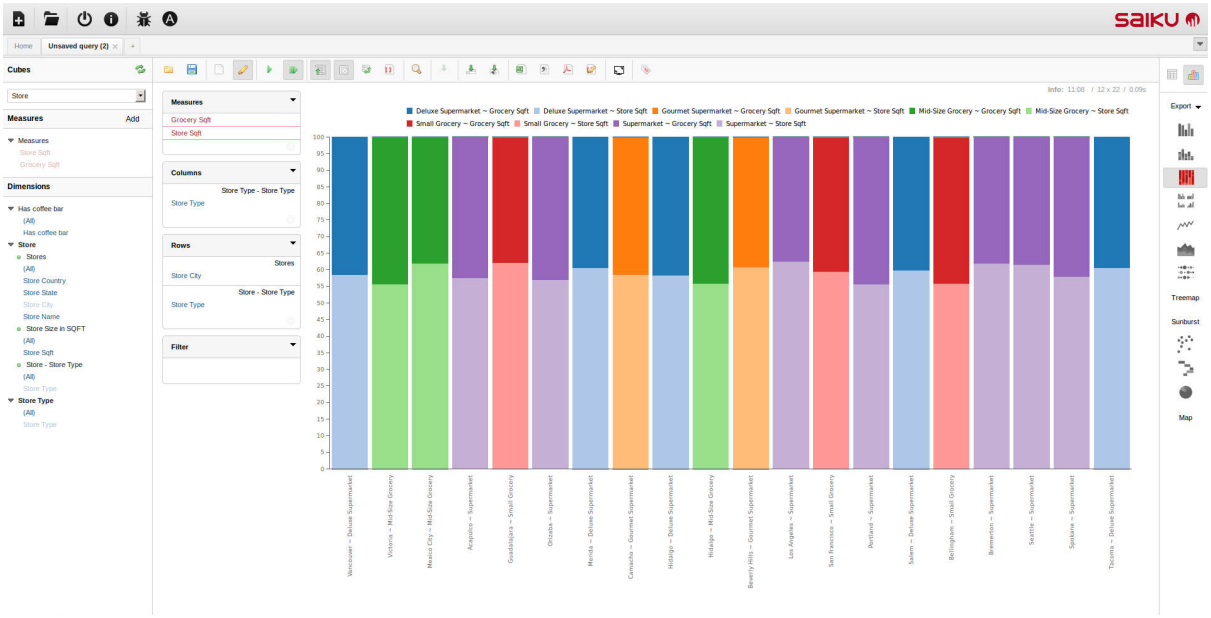
Once, the application is started, you can access to the user interface using a web browser at the link below.

`http://<server>:8080/pentaho`

With Pentaho Analytics platform Community Edition is suggested to use Mozilla Firefox or Google Chrome instead of MS IE Explorer (not well supported today). To access as administrator you can use the `admin` user with password `password`.

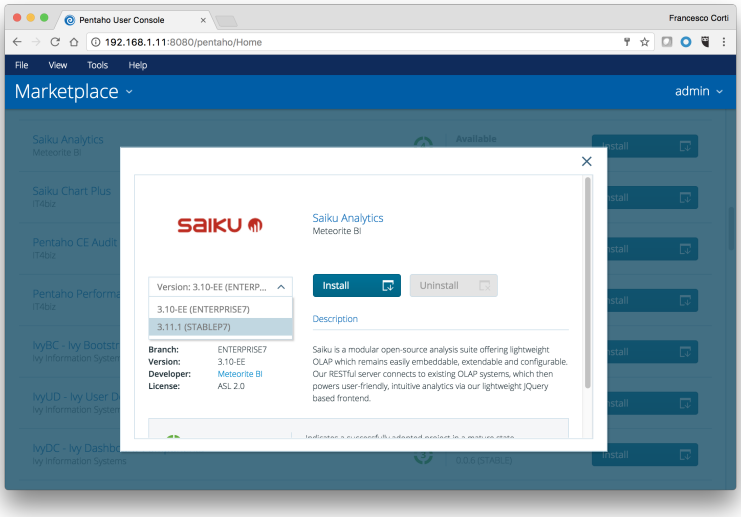
3.1.3 Installing Saiku analytics (optional)

Saiku Analytics allows business users to explore complex data sources, using a familiar drag and drop interface and easy to understand business terminology, all within a browser. Saiku analytics is used by the Alflytics solution to enable the user to the so called “free analysis”. It is not mandatory to install (and use) the Saiku Analytics but it’s recommended if you will want to use OLAP and pivoting over the data.



Saiku Analytics is released with a Community Edition and an Enterprise Edition. You can start evaluating the features of the Community Edition (free of charge) and upgrade to the Enterprise Edition later on.

Saiku Analytics installation is very straightforward, starting from an existing Pentaho Analytics platform. First of all access to the Pentaho User Console as administrator (otherwise you will not have permits to access to the marketplace). Click on the `Home` button and then on `Marketplace`. In the marketplace list, search for the Saiku Analytics and be sure you are installing the latest Community Edition, not the Enterprise Edition. Below a screenshot showing how the user interface look like.



After selecting the latest Community Edition, click on the `Install` button. Once the Saiku Analytics is installed, a message box will tell you the result. As clearly described, a JAR library linked directly in the message box must be downloaded and copied into the `<pentaho-server>/tomcat/webapps/pentaho/WEB-INF/lib` folder.

Now it's time to stop the Pentaho Analytics platform, executing the `stop-pentaho` script stored in the `pentaho-server` folder.

Once the Pentaho Analytics platform is stopped, edit the file `<pentaho-server>/pentaho-solutions/system/importExport.xml`, adding the lines in bold described below.

```
...
<bean id="IRepositoryContentConverterHandler" ... >
 <constructor-arg>
  <util:map id="convertersMap">
   <entry key="mondrian.xml" value-ref="streamConverter"/>
   <entry key="jpeg" value-ref="streamConverter"/>
   <!-- Add the line below here!-->
   <entry key="saiku" value-ref="streamConverter"/>
  </util:map>
 </constructor-arg>
</bean>
...
<bean id="DefaultExportHandler" ... >
 <property name="repository" ref="unifiedRepository" />
 <property name="localeExportList">
  <list>
   <value>.xanalyzer</value>
   ...
   <!-- Add the lines below here! -->
   <value>.saiku</value>
```

```
  </list>
 </property>
</bean>
...
<bean

class="org.pentaho.platform.plugin.services.importer.LocaleImportH
andler">
 <constructor-arg>
  <list>
   <value>xaction</value>
   ...
   <!-- Add the lines below here! -->
   <value>saiku</value>
  </list>
...
```

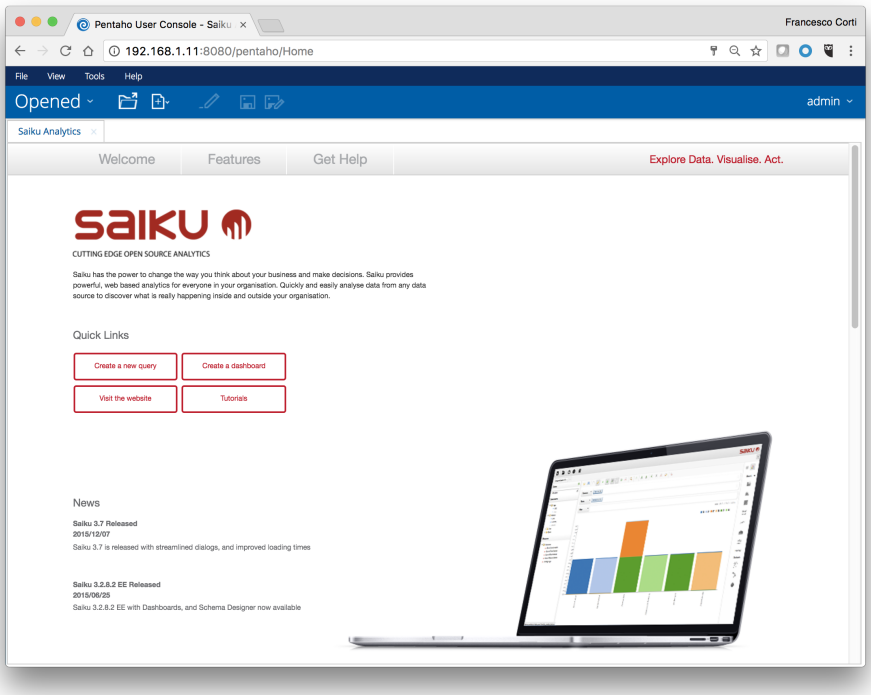
Then edit the file `<pentaho-server>/pentaho-solutions/system/ImportHandlerMimeTypeDefinitions.xml`, adding the lines in bold described below.

```
...
<ImportHandler
class="org.pentaho.platform.plugin.services.importer.RepositoryFil
eImportFileHandler">
 <MimeTypeDefinitions>
  ...
  <MimeTypeDefinition                          mimeType="text/xml">
   ...
  </MimeTypeDefinition>

  <!--      Add      the      lines      below      here!      -->
  <MimeTypeDefinition                    mimeType="application/json">
   <extension>saiku</extension>
  </MimeTypeDefinition>

 </MimeTypeDefinitions>
</ImportHandler>
```

Last, but not least, there is the (free of charge) license to download and install into the `<pentaho-server>/pentaho-solutions/system/saiku` folder. Save the license file with name `license.lic`. To get the (free of charge) license, you can access directly to the url http://licensing.meteorite.bi. All the registration process and request of licenses for Community Edition is free of charge.

After this tasks are done, start again the Pentaho Analytics platform, executing the `start-pentaho` script stored in the `pentaho-server` folder.

To check if everything is working fine, access to the Pentaho User Console (`http://<server>:8080/pentaho`) and click on `File`, then `New`, then `Saiku Analytics`. Below a screenshot describes what the browser look like.



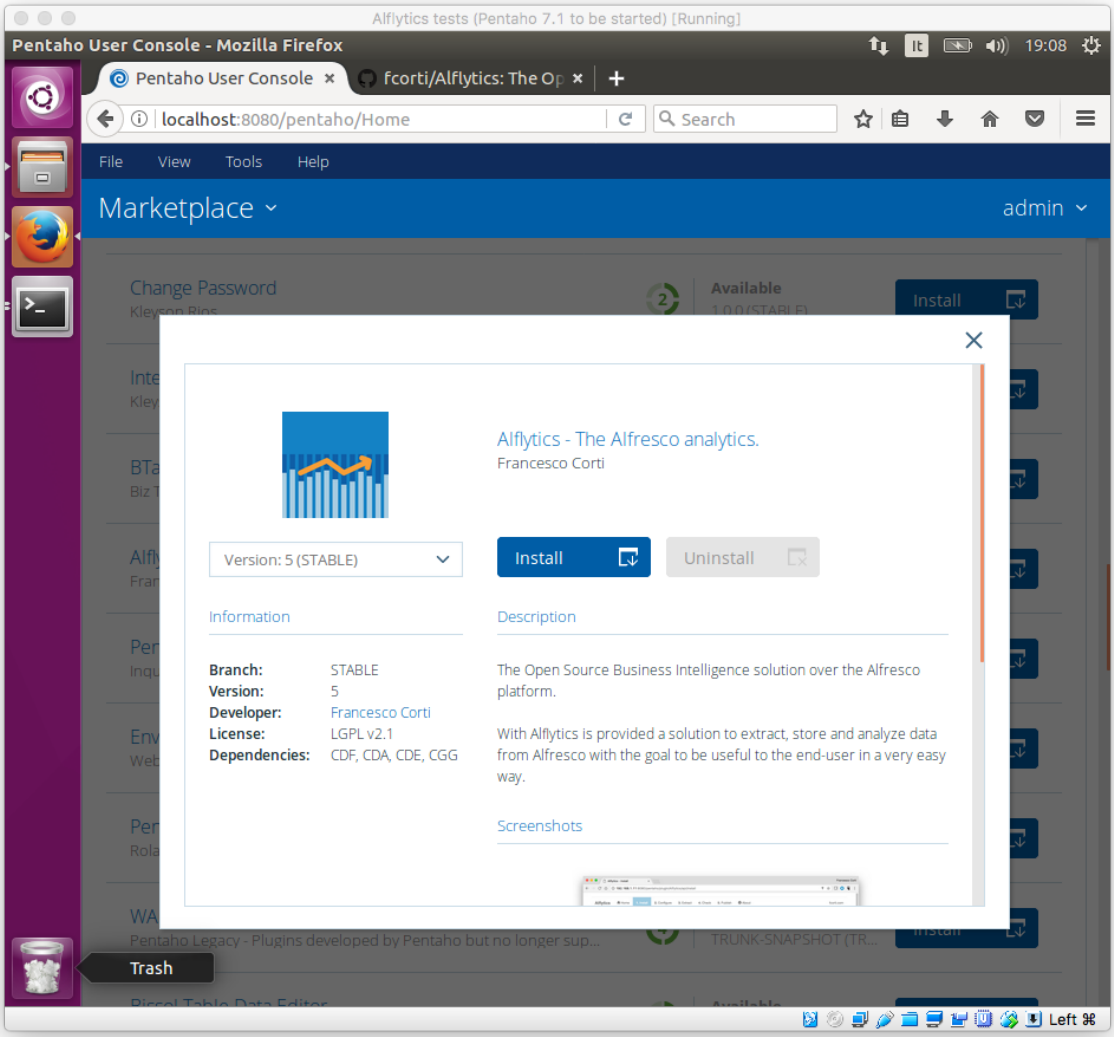
3.2 Getting Alflytics

There are two possible ways to get Alflytics: using the Pentaho Marketplace and downloading it manually. Below a brief description of each possible way. Both the possible paths get Alflytics into your Pentaho Analytics platform, with exactly the same result. Feel free to choose your preferred one, according to your network limitations and preferences.

3.2.1 Getting Alflytics using the Pentaho marketplace

Getting Alflytics using the Pentaho marketplace is similar to what we described for installing Saiku analytics.

First of all access to the Pentaho User Console as administrator (otherwise you will not have permits to access to the marketplace). Click on the `Home` button and then on `Marketplace`. In the marketplace list, search for the Alflytics. Click on the `Install` button and wait for the message box sharing the result of downloading.



Once the message box will come with a positive answer, it's time to restart the Pentaho Analytics platform executing the `stop-pentaho` script and the `start-pentaho` script later on. Both the scripts are stored in the `pentaho-server` folder.

3.2.2 Downloading Alflytics

Downloading and plugging Alflytics into the Pentaho Analytics platform is very straightforward.

To download the latest Alflytics distribution you can check into the GitHub project at the link below.

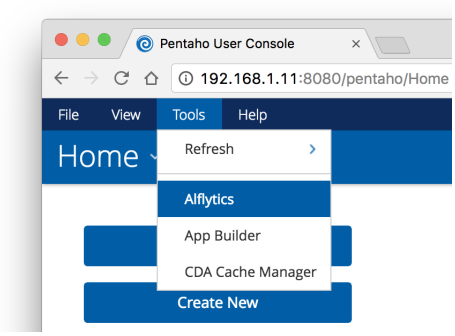
https://github.com/fcorti/Alflytics/releases

Download the `zip` file called `Alflytics_vX.X.zip` from the `Downloads` section and unzip it into the folder `<pentaho-server>/pentaho-solutions/system`. You should find then, a new folder called `Alflytics`.

Once the Alflytics distribution is plugged, it's time to restart the Pentaho Analytics platform executing the `stop-pentaho` script and the `start-pentaho` script later on. Both the scripts are stored in the `pentaho-server` folder.

3.2.3 Checking Alflytics

If you downloaded Alflytics using the Pentaho Marketplace ot the manual setup, you can easily check the existence of Alflytics into your Pentaho Analytics platform, accessing to the Pentaho User Console and checking into the `Tools` item in the upper menu. If everything has been properly setup, the menu should contain an `Alflytics` item, as shown in the image below.



Clicking on the `Alflytics` item, a blank panel will be shown and an error will be logged into the `catalina.out`. In case you did it, don't worry, this is regular behaviour, considering you have not installed the dashboards yet.

3.3 Installing Alflytics

Now that Alflytics is plugged into the Pentaho Analytics platform, it's time to install everything is requested to have your fully featured Open Source Business Intelligence solution.

By default Alflytics comes with the Data Warehouse stored on a HyperSQL (or HSQL) database. HSQLDB is a lightweight RDBMS written in Java, extremely useful for small databases and/or evaluation purpose. HSQLDB is particularly interesting for our use case because it is distributed in the Pentaho Analytics platform and does not require external or different installations.

HyperSQL is recommended to be used on evaluation use cases only. If you plan to experiment Alflytics or to develop a POC on a small Alfresco repository, there are no concerns on using the default release. If you plan to use Alflytics in a production environment or a large repository, you should consider to use PostgreSql or MySql instead.

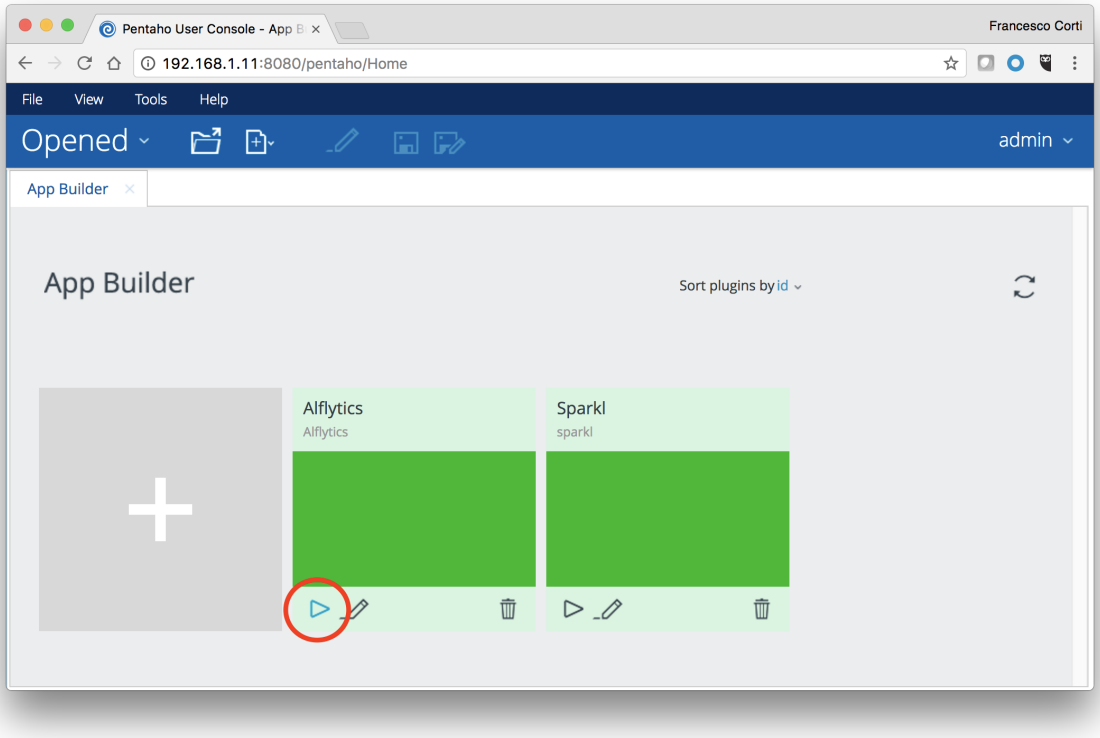
Below a detailed description of the tasks to follow to install the easiest version of Alflytics.

3.3.1 Accessing the Alflytics wizard

Alflytics has a wizard to make the installation task extremely easy. To access the installation wizard you simply have to access to the Administration Web pages, at the link below.

```
http://<server>/pentaho/plugin/Alflytics/api/main
```

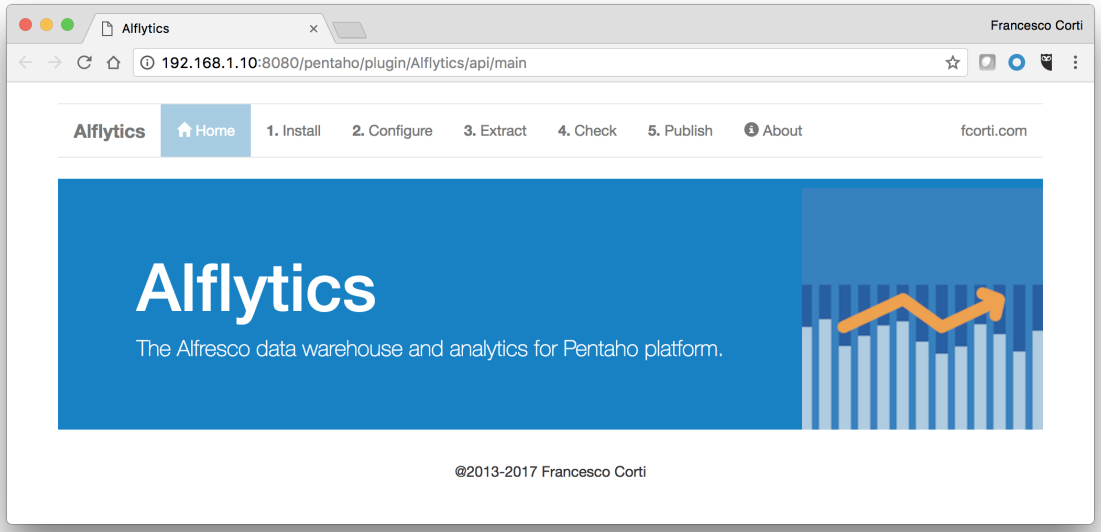
As alternative, click on the `App Builder` item of the menu shown in the previous image (in the Checking Alflytics paragraph). Once the `App Builder` item is clicked, a panel will appear as described in the screenshot below.



To access to the Alflytics wizard, click on the arrow, as highlighted in the screenshot.

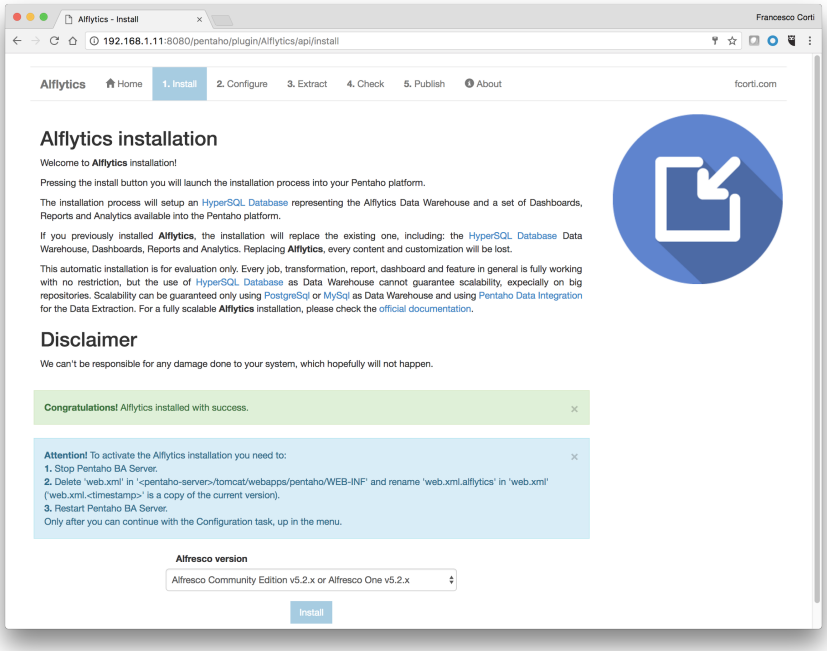
3.3.1 Installing Alflytics using the wizard

The Alflytics Administration Web pages can easily be used, starting from the upper menu shown in the screenshot below.



Each task is numbered to define the right order in terms of execution. The first task to follow, on the right of the `Home`, is about installing Alflytics. Click on the `1.Install` item to access to the installation page. As you can see directly in the page, the installation is very straightforward and does not require any difficult setup or question. Simply select the Alfresco version you are going to use as source of your data, and click the `Install` button.

Below a screenshot showing the result, if everything is successfully installed. As usual, if something goes wrong or you want to be sure about the details of the task, check the `catalina.out` of the Pentaho Analytics platform.

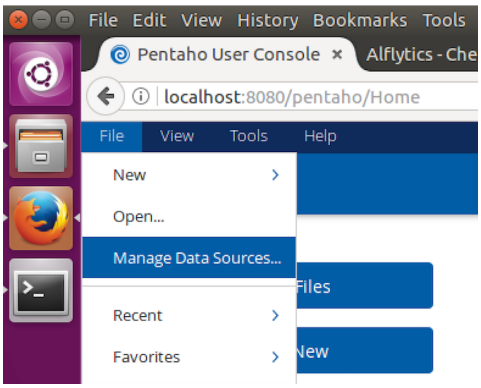


To activate the Alflytics installation, there are few tasks more to do. Below the list to follow.

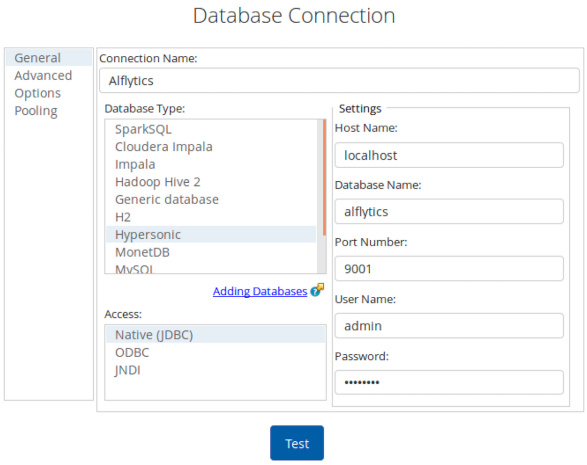
First of all stop the Pentaho Analytics platform, executing the `stop-pentaho` script stored in the `pentaho-server` folder. Then delete the `web.xml` file in `<pentaho-server>/tomcat/webapps/pentaho/WEB-INF` and rename `web.xml.alflytics` as `web.xml` (`web.xml.<timestamp>` is a copy of the current version of the file). Restart Pentaho Analytics platform again, executing the `start-pentaho` script stored in the `pentaho-server` folder.

There is currently a not solved issue on the datasource creation. The good news is that it affects only the installation and can be easily solved with an extra-task. Below the step-by-step description of the extra-task mentioned above.

Access to the Pentaho User Console and click on the `Manages Data Sources` item in the upper menu.



Select the `Alflytics` data source of the `JDBC` type and edit it (using the gear icon). Once in the data source edit window, change all the fields according to the image below. The click on the `test` button to check everything is working fine, and then click on `Ok`.



Congratulations, your Alflytics environment is now installed into your Pentaho Analytics platform. Now it's time to configure the Alfresco source, clicking the `2.Configuration` item, up in the menu.

3.3.2 Configuring the access to Alfresco

The access to Alfresco is relevant to extract the data into the Alflytics Data Warehouse (a database schema into the RDBMS). To complete the task, click on the `2.Configuration` item, up in the menu, and complete the requested fields. Once correctly filled, press the `Save` button and a message will tell you the result.

The configuration wizard updates the `DWH_ALF_DIM_INSTANCES` table into the Alflytics Data Warehouse (a database schema into the RDBMS). You can manually setup the `DWH_ALF_DIM_INSTANCES`, if you want to change the settings and have a full control.

In some cases is experienced that the fields are not correctly updated in the form, after the `Save` button has been pressed. This is caused by the database caching done by the Pentaho platform. If you are sure you filled the fields with the correct values and pressed the `Save` button, feel free to ignore this behaviour. As alternative, to be sure, use your preferred database client and access directly to the `DWH_ALF_DIM_INSTANCES` table, to check (and eventually modify) the fields.

After the configuration has been completed, it's time to extract the data from Alfresco and only them using dashboards, reports and free analysis. To understand how to do it, follow the chapter dedicated to "How to use Alflytics".

3.3.3 Troubleshooting

If you have any kind of issue during the installation and/or configuration tasks, the suggestion is to check the `catalina.out` file of the Pentaho Analytics platform. The log file is very detailed and always reports a lot of useful information. Only after a deep study of the `catalina.out` file, if you don't know how to solve the issue, try to ask for support as described into the "Contacts and support" chapter.

3.4 Manual installation of Alflytics (advanced)

In this section we will see how to manually install Alflytics. The reasons why you could want to install it manually, could be different. For sure you could want to scale and adopt an Enterprise database, or simply you should want to control each component.

3.4.1 Installing Alflytics on PostgreSql

Installing Alflytics using PostgreSql is highly recommended as alternative to HyperSQL to store the Data Warehouse. Please note that adopting PostgreSql some features won't work:

we are talking about the web pages related to Alflytics Admin (except for the Data Quality that will continue to be supported).

Below the steps to follow to install Alflytics on PostgreSql.

3.4.1.1 Prerequisites

Before moving to the Alflytics specific tasks, let's check the prerequisites of the environment to prepare the installation. The prerequisites to check are mainly about:

- Installing PostgreSql and PgAdmin3.
- Installing Pentaho Analytics platform.
- Getting Alflytics (without installing it).

Below one detailed description for each topic.

Installing PostgreSql and PgAdmin3

If PostgreSql is not installed you can proceed with the tasks below, according with the official documentation described here. If you already have PostgreSql installed, for example because an Alfresco instance is up and running, you can use it for your development purpose.

To install PostgreSql, open a terminal and execute the commands below. Together with PostgreSql we are going to install also PgAdmin3 to administer it.

```
sudo apt-get install postgresql postgresql-contrib pgadmin3
sudo -u postgres psql postgres
\password postgres
// Digit 'postgres' (two times) to setup the user's password.
\q
sudo nano /etc/postgresql/9.5/main/pg_hba.conf
```

Change the line described below, setting `peer` to `md5`.

```
# Database administrative login by Unix domain socket
local   all             postgres                                md5
```

To complete the installation, reload the database service executing the command below.

```
sudo /etc/init.d/postgresql reload
```

Last, but not least, execute `pgadmin3` and add the localhost connection to administer the database instances.

Installing Pentaho Analytics platform

You can follow the instructions on paragraph 3.1.2 Installing Pentaho Analytics platform.

Getting Alflytics (without installing it)

You can follow the instructions on paragraph 3.2 Getting Alflytics.

Please DO NOT install Alflytics but only download it into the Pentaho Analytics platform.

3.4.1.2 Creating the Alflytics schema

After checking the prerequisites, it's time to create a new database schema called `alflytics`. You can easily do it using the PgAdmin3 and executing the commands below.

```
DROP DATABASE IF EXISTS alflytics;
CREATE DATABASE alflytics;
ALTER DATABASE alflytics OWNER TO postgres;
```

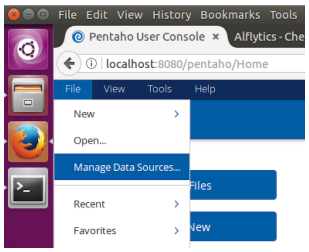
After this you can execute the script called `alflytics.postgresql.sql` and stored into the path below.

```
<pentaho-server>/pentaho-
solutions/system/Alflytics/endpoints/kettle/src/5.2/data/postgresql
```

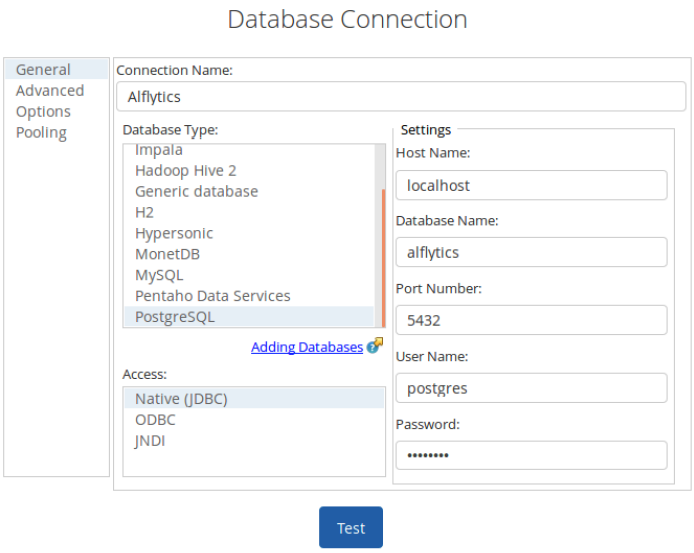
Last but not least, update the table `DWH_ALF_DIM_INSTANCES` with the right data about connecting the Alfresco instance. Please be sure you update at least the following fields: `LOGIN`, `PASSWORD`, `PROTOCOL`, `HOST`, `PORT`.

3.4.1.3 Creating the Alflytics data source on Pentaho Analytics platform

To create the Alflytics data source on Pentaho Analytics platform, access to the Pentaho User Console and click on the `Manages Data Sources` item in the upper menu. Below a screenshot showing a portion of the page to show how it look like.



Once selected the item, a modal window appears with the list of the existing data sources. Click on the gear (on the top right of the modal window) and select the `New connection` item. Once in the data source edit window, fill all the fields according to the image below.



Please pay attention to the capital letters in `Alflytics`. The data source are case sensitive depending on the tools you are going use and an error on the name could cause a lot of unpredictable issues.

Then click on the `test` button to check everything is working fine, and then click on `Ok` to save the data source from the available ones.

3.4.1.4 Importing the Pentaho repository

In this task we are going to import the ETL jobs/transformations, reports, dashboards and OLAP queries into the Pentaho repository. The very first task is to stop the Pentaho Analytics platform, executing the `stop-pentaho` script stored in the `pentaho-server` folder. Once done, it's time to set up the Alflytics installation with a collection of resources. Below it is described a task for each type of resources to make the installation easier.

Let's start replacing the data quality dashboard. To complete the task, copy the files from the folder:

```
<pentaho-server>/pentaho-solutions/system/Alflytics/endpoints/kettle/src/5.2/dashboards/postgresql
```

To the target folder described below.

```
<pentaho-server>/pentaho-solutions/system/Alflytics/dashboards
```

Please note that the copy is in reality a replacement, because a copy of the files with the same names are already stored in the target folder. The default version of the files are developed to point to the HyperSQL database, instead we want the same dashboard pointing to the PostgreSql database.

Now that the data quality dashboard is updated, let's copy the other resources into the import folder, used by Pentaho during restarting. To complete the task, copy the zip files listed below, from the following folder.

```
<pentaho-server>/pentaho-
solutions/system/Alflytics/endpoints/kettle/src/5.2/repository
```

To the target folder described below.

```
<pentaho-server>/pentaho-solutions/system/default-content
```

The files to copy are:

- `Alflytics.Datasources.postgresql.zip`
- `Alflytics.Dashboards.postgresql.zip`
- `Alflytics.ETL.zip`
- `Alflytics.OLAP.zip`
- `Alflytics.Reports.postgresql.zip`

Once completed, it's time to restart the Pentaho Analytics platform again, executing the `start-pentaho` script stored in the `pentaho-server` folder. After restarting you will see the zip file imported into the `default-content` folder are renamed with a postfix containing a timestamp. This means the content has been imported into the Pentaho repository. To check everything is correctly imported, open the Pentaho User Console and dive into the repository (using the `Browse` item in the main dashboard).

If you defined a password for the `postgres` user, different from `postgres`, remember to update it again in the datasource, using the Pentaho User Console.

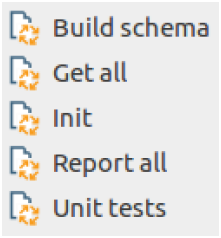
3.4.1.5 Updating the jobs to work with PostgreSql

Now that everything is correctly on board of the Pentaho Analytics platform, it's time to setup the main jobs to work with PostgreSql as default database. To update the jobs, you need a Pentaho Data Integration installation into a development environment.

If you decide to install and use Pentaho Data Integration for scalability reasons, you can use the production installation to update the jobs. As alternative you can choose to install and use an instance into your development environment.

To install Pentaho Data Integration (into your production environment or your development environment) you can follow the section dedicated to 'Using the Pentaho Data Integration for ETL' in the 'Scalability' chapter.

Once Pentaho Data Integration is correctly installed and setup to point to the Alflytics repository on Pentaho Analytics platform, open all the jobs listed below to update the default value of the `alflytics.dbType` parameter.



We would like to remember that all the jobs/transformations are available into the Pentaho repository at the path `/public/Alflytics/ETL`.

To update the default value of the `alflytics.dbType` parameter for each job, simply open the job into the Spoon user interface (the Pentaho Data Integration IDE). Edit the properties of the job (double clicking on the job name on the left panel of the user interface) and select the `parameters` tab. In the panel you will see all the parameters with their default values. Change the `Default value` for `alflytics.dbType` from `hsql` to `postgresql`. Press the `Ok` button and save the job back into the repository.

3.4.1.6 Updating the database schema

By default the database instance is distributed with the minimum tables and data structures. To create all the data structures (tables, indexes, sequences, etc.) useful to make Alflytics work, you have to execute the `Build schema` job. With the default parameters, one time only (if you don't have any error). You can easily launch the job using the Spoon user interface (the Pentaho Data Integration IDE).

Once the `Build schema` job has been executed, you will see the number of tables into the `alflytics` schema increasing in number (around a dozen of tables are automatically created).

This is all you need to install Alflytics on PostgreSql. Now you can continue extracting data as described into the chapter dedicated to how to use Alflytics.

3.4.2 Installing Alflytics on MySql

Installing Alflytics using MySql is highly recommended as alternative to HyperSQL to store the Data Warehouse. Please note that adopting MySql some features won't work: we are talking about the web pages related to Alflytics Admin (except for the Data Quality that will continue to be supported).

Below the steps to follow to install Alflytics on MySql.

3.4.2.1 Prerequisites

Before moving to the Alflytics specific tasks, let's check the prerequisites of the environment to prepare the installation. The prerequisites to check are mainly about:

- Installing MySql and MySql Workbench.
- Installing Pentaho Analytics platform.
- Getting Alflytics (without installing it).

Below one detailed description for each topic.

Installing MySql and MySql Workbench

To install MySql proceed to get it using the commands below.

```
sudo apt-get install mysql-server mysql-workbench
```

This command will install also MySql Workbench for administration. During the installation process, the definition of the `root` password will be requested. Last, but not least, execute `mysql-workbench` and add the `localhost` connection to administer the database instances.

Installing Pentaho Analytics platform

You can follow the instructions on paragraph 3.1.2 Installing Pentaho Analytics platform.

Getting Alflytics (without installing it)

You can follow the instructions on paragraph 3.2 Getting Alflytics.

Please DO NOT install Alflytics but only download it into the Pentaho Analytics platform.

3.4.2.2 Creating the Alflytics schema

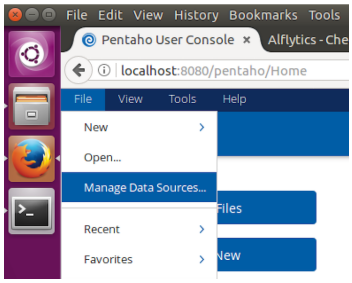
After checking the prerequisites, it's time to create a new database schema called `alflytics`. You can easily do it using the MySql Workbench and executing the script called `alflytics.mysql.sql`, stored into the path below.

```
<pentaho-server>/pentaho-
solutions/system/Alflytics/endpoints/kettle/src/5.2/data/mysql
```

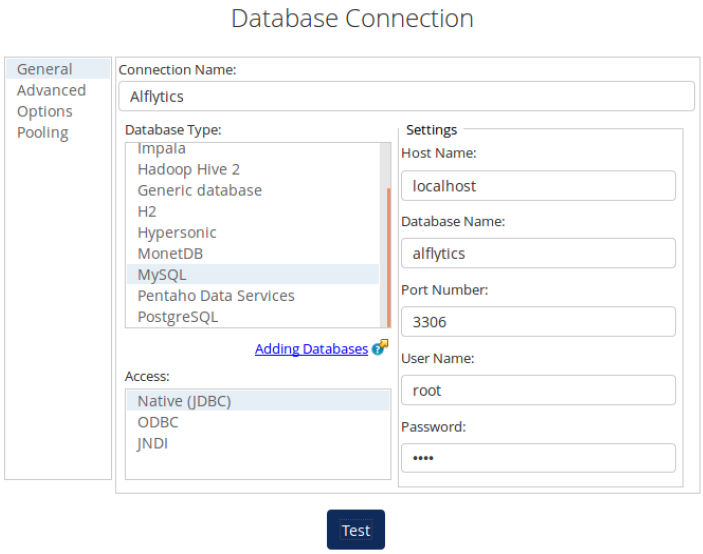
Last but not least, update the table `DWH_ALF_DIM_INSTANCES` with the right data about connecting the Alfresco instance. Please be sure you update at least the following fields: `LOGIN`, `PASSWORD`, `PROTOCOL`, `HOST`, `PORT`.

3.4.2.3 Creating the Alflytics data source on Pentaho Analytics platform

To create the Alflytics data source on Pentaho Analytics platform, access to the Pentaho User Console and click on the `Manages Data Sources` item in the upper menu. Below a screenshot showing a portion of the page to show how it look like.



Once selected the item, a modal window appears with the list of the existing data sources. Click on the gear (on the top right of the modal window) and select the `New connection` item. Once in the data source edit window, fill all the fields according to the image below.



Please pay attention to the capital letters in `Alflytics`. The data source are case sensitive depending on the tools you are going use and an error on the name could cause a lot of unpredictable issues.

Then click on the `test` button to check everything is working fine, and then click on `Ok` to save the data source from the available ones.

3.4.2.4 Importing the Pentaho repository

In this task we are going to import the ETL jobs/transformations, reports, dashboards and OLAP queries into the Pentaho repository. The very first task is to stop the Pentaho Analytics platform, executing the `stop-pentaho` script stored in the `pentaho-server`

folder. Once done, it's time to set up the Alflytics installation with a collection of resources. Below it is described a task for each type of resources to make the installation easier.

Let's start replacing the data quality dashboard. To complete the task, copy the files from the folder:

```
<pentaho-server>/pentaho-
solutions/system/Alflytics/endpoints/kettle/src/5.2/dashboards/mysql
```

To the target folder described below.

```
<pentaho-server>/pentaho-solutions/system/Alflytics/dashboards
```

Please note that the copy is in reality a replacement, because a copy of the files with the same names are already stored in the target folder. The default version of the files are developed to point to the HyperSQL database, instead we want the same dashboard pointing to the MySql database.

Now that the data quality dashboard is updated, let's copy the other resources into the import folder, used by Pentaho during restarting. To complete the task, copy the zip files listed below, from the following folder.

```
<pentaho-server>/pentaho-
solutions/system/Alflytics/endpoints/kettle/src/5.2/repository
```

To the target folder described below.

```
<pentaho-server>/pentaho-solutions/system/default-content
```

The files to copy are:

- `Alflytics.Datasources.mysql.zip`
- `Alflytics.Dashboards.mysql.zip`
- `Alflytics.ETL.zip`
- `Alflytics.OLAP.zip`
- `Alflytics.Reports.mysql.zip`

Once completed, it's time to restart the Pentaho Analytics platform again, executing the `start-pentaho` script stored in the `pentaho-server` folder. After restarting you will see the zip file imported into the `default-content` folder are renamed with a postfix containing a timestamp. This means the content has been imported into the Pentaho repository. To check everything is correctly imported, open the Pentaho User Console and dive into the repository (using the `Browse` item in the main dashboard).

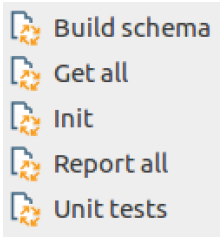
3.4.2.5 Updating the jobs to work with MySql

Now that everything is correctly on board of the Pentaho Analytics platform, it's time to setup the main jobs to work with MySql as default database. To update the jobs, you need a Pentaho Data Integration installation into a development environment.

If you decide to install and use Pentaho Data Integration for scalability reasons, you can use the production installation to update the jobs. As alternative, if you don't plan to use Pentaho Data Integration, it is suggested to install and use an instance into your development environment.

To install Pentaho Data Integration (into your production environment or your development environment) you can follow the section dedicated to 'Using the Pentaho Data Integration for ETL' in the 'Scalability' chapter.

Once Pentaho Data Integration is correctly installed and setup to point to the Alflytics repository on Pentaho Analytics platform, open all the jobs listed below to update the default value of the `alflytics.dbType` parameter.



We would like to remember that all the jobs/transformations are available into the Pentaho repository at the path `/public/Alflytics/ETL`.

To update the default value of the `alflytics.dbType` parameter for each job, simply open the job into the Spoon user interface (the Pentaho Data Integration IDE). Edit the properties of the job (double clicking on the job name on the left panel of the user interface) and select the `parameters` tab. In the panel you will see all the parameters with their default values. Change the `Default value` for `alflytics.dbType` from `hsql` to `mysql`. Press the `Ok` button and save the job back into the repository.

3.4.2.6 Updating the database schema

By default the database instance is distributed with the minimum tables and data structures. To create all the data structures (tables, indexes, sequences, etc.) useful to make Alflytics work, you have to execute the `Build schema` job. With the default parameters, one time only (if you don't have any error). You can easily launch the job using the Spoon user interface (the Pentaho Data Integration IDE).

Once the `Build schema` job has been executed, you will see the number of tables into the `alflytics` schema increasing in number (around a dozen of tables are automatically created).

This is all you need to install Alflytics on MySql. Now you can continue extracting data as described into the chapter dedicated to how to use Alflytics.

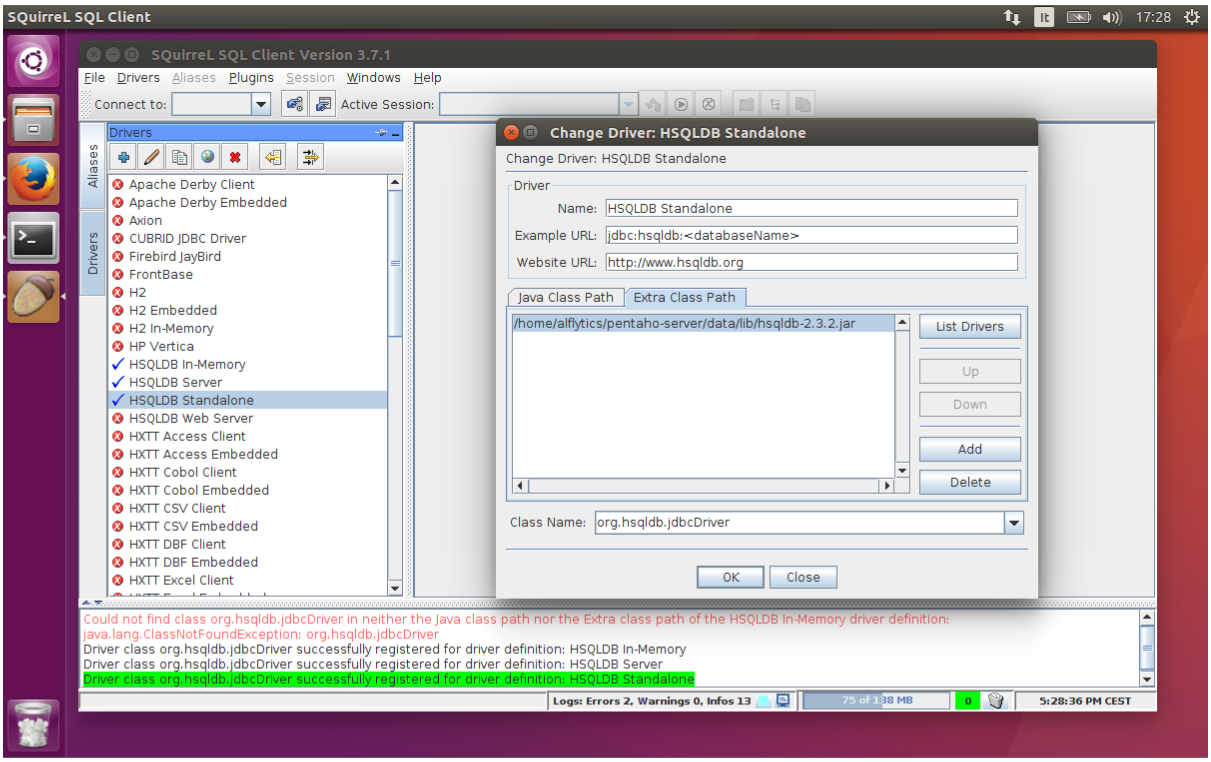
3.4.3 Installing SQuirreL SQL client for HyperSQL

To have a full access to the Alflytics Data Warehouse on HyperSQL, you need a database client. There are several useful database clients, and honestly you can decide to use your preferred one, but here we are going to explain how to install (and use) SQuirreL Universal SQL Client.

The official website and documentation is extremely clear and straightforward. Below are shared some further details on specific configurations, useful to our use case.

- The installation is a standard installation, as documented in the official SQuirreL SQL Client website.

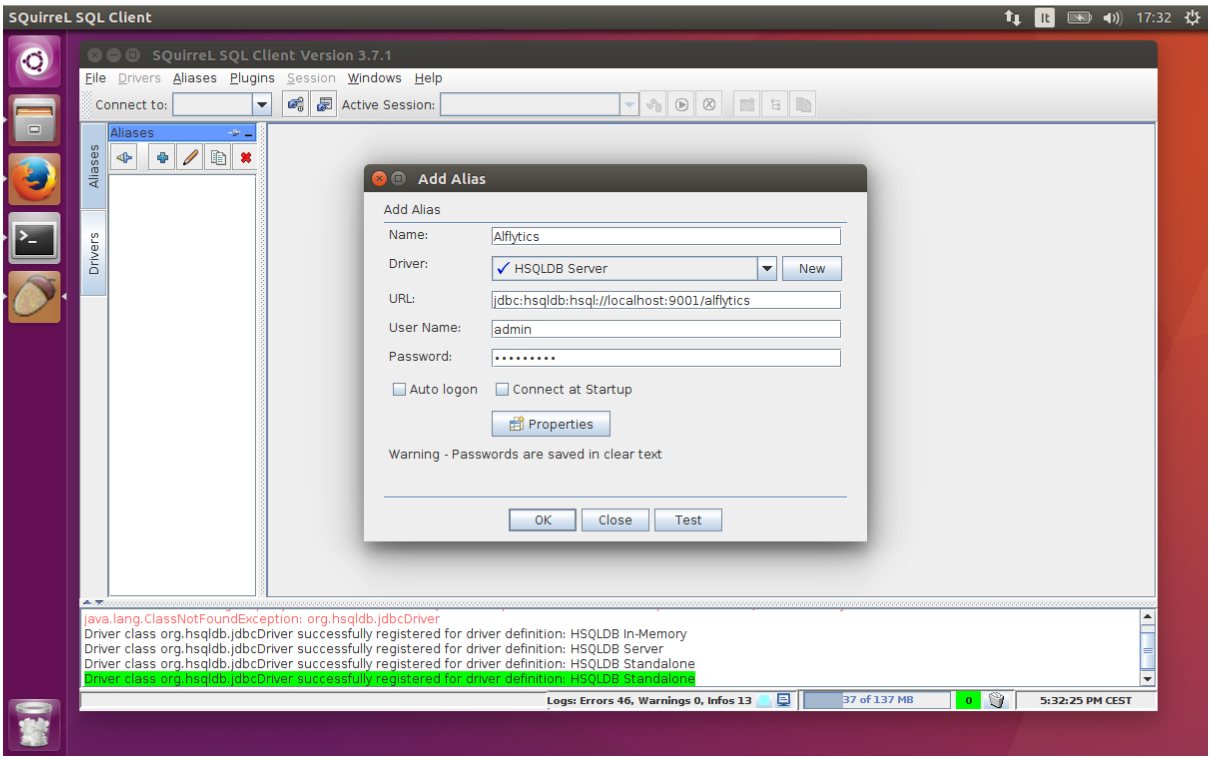
- To have access to the Alflytics Data Warehouse on HyperSQL (or HSQLDB), you need to add few new drivers, using the SQuirreL User Interface. From the driver list, identify the HSQLDB In-Memory, HSQLDB Server and HSQLDB Standalone. For each driver, modify it adding an extra class-path. The class-path to add is the JAR file in the `<pentaho-server>/data/lib/hsqldb-<version>.jar` path. Below a screenshot showing the user interface.



- To have access to the Alflytics Data Warehouse on HyperSQL (or HSQLDB), you need to add an alias, using the SQuirreL User Interface. From the alias list, add a new one, with the settings below.

  Name: Alflytics
  Driver: HSQLDB Server
  URL: `jdbc:hsqldb:hsql://<server>:9001/alflytics`
  User Name: `admin`
  Password: `password`

  Clicking on the `OK` button and you will have access to the whole Alflytics Data Warehouse with a full control. Below a screenshot showing the user interface.



3.5 About scheduling extractions and publications

The extraction of data from Alfresco and publication of report into Alfresco are possible using two different endpoints available in the Pentaho platform. Using `cURL`, or any other client supporting HTTP calls, is straightforward to automatically invoke the two tasks. On practical example is: scheduling extractions and publications.

The endpoint used to extract from Alfresco is available at:

```
http://<server>:<port>/pentaho/plugin/Alflytics/api/alflyticsextract
```

The endpoint used to publish reports into Alfresco is available at:

```
http://<server>:<port>/pentaho/plugin/Alflytics/api/alflyticspublish
```

As a consequence, you could easily schedule the following commands using CRON (or your preferred scheduler).

```
curl -u <login>:<password>
http://<server>:<port>/pentaho/plugin/Alflytics/api/alflyticsextract

curl -u <login>:<password>
http://<server>:<port>/pentaho/plugin/Alflytics/api/alflyticspublish
```

(login and password are Pentaho login and password, by default `admin` and `password`).

4. Customizing Alflytics

Customizing Alflytics means not only one thing but could be on several different parts. First of all, you could want to customize reports, modifying the existing ones or adding new ones. You could also want to customize dashboards and free analysis. Last but not least you could want to extract data related to Alfresco custom models or include other sources in the Data Warehouse like: ERPs, CRMs and/or custom applications. In the following paragraphs we are going to cover the most common use cases.

4.1 Customizing reports

What customizing reports means
Installing Pentaho reporting
The available reports
Basis of Pentaho reporting

4.2 Customizing dashboards

About CTools and in particular CDE.

4.3 Customizing free analysis

About Mondrian Schema.

4.4 Customizing the warehouse and the ETL

As you can easily imagine, Data Warehouse and ETL are strictly related. Alflytics is designed to be modular on ETL jobs and transformations and guarantee the lowest impact on your management because everything is essentially automatic. In this paragraph you will learn how to configure (and customize) the Data Warehouse and ETLs for specific needs as: the Alfresco custom model, the Alfresco Audit Trail and the integration of other sources.

4.4.1 How to extract Alfresco custom models

One of the most interesting enhancements of Alflytics is about the extraction of the custom models metadata. Custom models are completely under your control to be extracted and analyzed in the way you want.

The way how Alflytics extracts the custom metadata can be managed using some tables: `DWH_ALF_CLASSES` and `DWH_ALF_CLASS_PROPERTIES`. The `DWH_ALF_CLASSES` table stores all the Alfresco classes (i.e. types and aspects) you are interested to extract. The `DWH_ALF_CLASS_PROPERTIES` table stores all the Alfresco metadata you are interested to extract.

To customize the extraction and analytics:

- Add one row in the `DWH_ALF_CLASSES` for each Alfresco classes (i.e. type/aspect) you want to extract and analyze.
- Add one row in the `DWH_ALF_CLASS_PROPERTIES` for each Alfresco property you want to extract and analyze.

Below a description of the meaning of each field of both the two tables.

DWH_ALF_CLASSES

`ID`	Unique identifier of the class in the whole table (integer value).
`NAME`	Description of the Alfresco type/aspect.
`IS_FILE`	Flag indicating if the class is a `cm:content` sub-type. Values:`Y`/`N`.
`IS_FOLDER`	Flag indicating if the class is a `cm:folder` sub-type. Values:`Y`/`N`.
`IS_ASPECT`	Flag indicating if the class is an aspect. Values:`Y`/`N`.
`IS_ROOT`	Flag indicating if the class has a parent classes extracted. Values:`Y`/`N`.
`OPE_TABLE_NAME`	Name of the table hosting the data in the Operational layer for the class. Give it a name similar to `OPE_ALF_<className>`.
`DWH_TABLE_NAME`	Name of the table hosting the data in the Data Warehouse layer for the class. Give it a name similar to `DWH_ALF_<className>`.
`ORDINAL`	Integer defining the order of the build for the data structures. Low numbers are build before higher numbers.
`INSTANCE_ID`	Foreign key to `DWH_ALF_DIM_INSTANCES` indicating the Alfresco instance.
`IS_ACTIVE`	Flag indicating if the class should be extracted or not. Values:`Y`/`N`.

DWH_ALF_CLASS_PROPERTIES

`ID`	Unique identifier of the class in the whole table (integer value).
`NAME`	Description of the Alfresco property.
`TYPE`	String indicating the Alfresco type. Examples: `d:text`, `d:datetime`, `d:int`, `d:category`, etc.

IS_MULTIVALUE	Flag indicating if the property is multivalue. Values:Y/N.
OPE_TABLE_NAME	Name of the table hosting the property values in the Operational layer. Used only if IS_MULTIVALUE is Y. Give it a name similar to DWH_ALF_<propertyName>.
IS_OPE_INDEXED	Flag indicating if the property values should be indexed in table. Values:Y/N.
DWH_TABLE_NAME	Name of the table hosting the property values in the Data Warehouse layer. Used only if IS_MULTIVALUE is Y. Give it a name similar to OPE_ALF_<propertyName>.
IS_DWH_INDEXED	Flag indicating if the property values should be indexed in table. Values:Y/N.
JSON_PATH	Path of the property value in the source JSON extracted from Alfresco.
FIELD_NAME	Name of the field in table.
CLASS_ID	Foreign key to DWH_ALF_CLASSES indicating the Alfresco class.
IS_ACTIVE	Flag indicating if the property should be extracted or not. Values:Y/N.

As an example, you can check the standard properties indexed for cm:content and cm:folder.

Before starting to change the tables, be sure you are following exactly the tasks listed below.

4.4.1.1 Initializing the Data Warehouse structures

Using the Spoon User Interface, open and run the job called Init. The job is stored into the /public/Alflytics/ETL folder of the Pentaho repository. Launching the job, be sure you correctly updated the parameters related to the Database Type (hsql, postgresql or mysql), the Pentaho URL components and the Alfresco instance id.

The execution of this job will re-init the Data Warehouse, removing the custom data structures. Check the log to verify the task is executed without any error or issue.

4.4.1.2 Modify DWH_ALF_CLASSES and DWH_ALF_CLASS_PROPERTIES

Now it's time to change the two tables, accordingly with the description above in the paragraph.

4.4.1.3 Build again the Data Warehouse structures

Using the `Spoon` User Interface, open and run the job called `Build schema`. The job is stored into the `/public/Alflytics/ETL` folder of the Pentaho repository. Launching the job, be sure you correctly updated the parameters related to the Database Type (`hsql`, `postgresql` or `mysql`), the Pentaho URL components and the Alfresco instance id. Leave `doCreate` and `doDelete` with `true` values.

The execution of this job will create all the custom data structures into the Data Warehouse. Check the log to verify the task is executed without any error or issue.

4.4.1.4 Extract and use it usual

Now the Data Warehouse and settings should be fine and it's extract again the data from Alfresco, exactly in the way it's explained in this document. Everything is automatically managed and nothing change to your regular behaviour.

4.4.2 How to extract custom Alfresco audit trails

Coming Soon!

4.4.3 How to integrate other sources

Integrating different sources into the Data Warehouse is possible thanks to the modularity of the ETL developement. We won't treat the technical details here because it would require a lot of time (and effort) but the suggestion is to start studying the source code of the ETL from the `Get all` job in the `/public/Alflytics/ETL` folder of the Pentaho repository.

If you need to integrate your CRM, ERP, custom data into the Data Warehouse and want to analyze them, together with the data coming from Alfresco, this is the right place to start.

5. Scalability

Alflytics is fully scalable, thanks to the use of a pure (and Enterprise) Business Intelligence solution. The Alflytics scalability is possible following different paths. Each path gives to the architecture a "boost" on a specific topic or components. In the following paragraphs we are going to describe all the available paths, with dedicated descriptions and setup.

5.1 The Alflytics Data Warehouse on PostgreSql or MySql

This scalability is given by the use of a scalable database, storing the data coming from the sources. In the previous chapters we saw how to install Alflytics using a PostgreSql database or a MySql database. By their nature, those two databases are highly scalable and guarantee great performances on huge repositories.

We won't treat more than this here, the scalability topic using PostgreSql or MySql, but it is highly recommended to adopt them, especially in production environments or huge repositories (also in POC or evaluation tasks).

5.2 Tuning the Pentaho Analytics platform

We won't treat here all the possible tuning Pentaho platform has, but we would like to suggest how to make it work better for our purpose. The main tuning is through the `JAVA_OPTS` you can find into the `start_pentaho` script, inside the `<pentaho-server>` folder.

For further details, please check the Pentaho Official Documentation.

5.3 Using the Pentaho Data Integration for ETL

Pentaho Data Integration is another great tool of the Pentaho family. Pentaho Data Integration is a powerful ETL engine, useful to manage huge flows of data from generic sources (in our case Alfresco) to generic targets (in our case the Alflytics database). Because of its nature, the Pentaho Data Integration it is recommended to be used when you need to scale managing huge repositories (but should be better to say: huge flows of data). We won't treat more than this the scalability of Pentaho Data Integration but please remember that it could be installed using clustering and on different architecture to improve the efficiency of the ETL jobs and transformations.

Before describing how to install Pentaho Data Integration for the Alflytics benefit, let's introduce `Spoon`. Spoon is the IDE for Pentaho Data Integration. Its Graphical User Interface is extremely friendly and let the developer able to manage the jobs/transformations, quite easily.

5.3.1 Installing Pentaho Data Integration

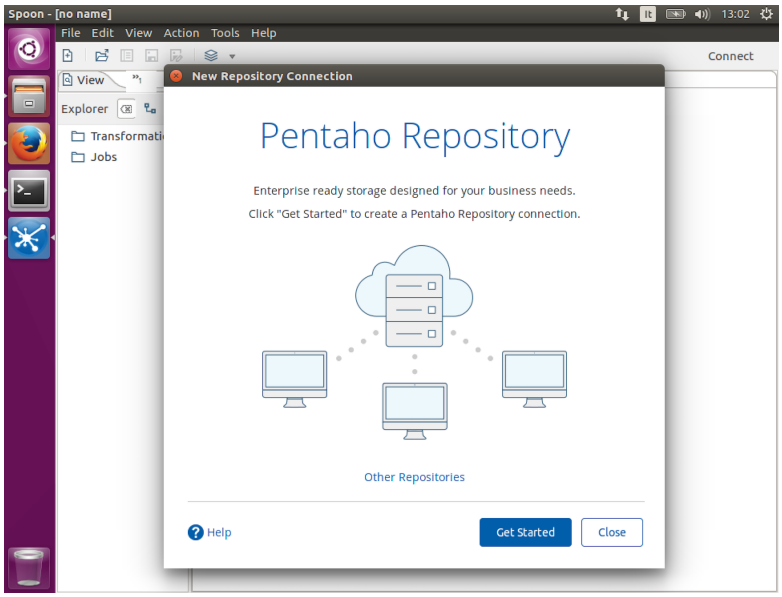
To install Pentaho Data Integration into your environment (development environment or production environment), there are few easy tasks to follow. The tasks are described with different approaches and in different scenarios in several articles in the Web. As reference, we would suggest one of the easiest at this link.

If you are using MySql as database, please remember to download the correct `mysql-connector-...jar` library from the official MySql Website and copy it into the `<data-integration>/lib` folder. Please be sure to download the correct version of the library for your version of MySql, otherwise you could experience unexpected errors and results.

5.3.2 Creating the connection to the Alflytics repository

Before using the Pentaho Data Integration for the first time, you have to setup the connection to the Alflytics repository. The task is extremely straightforward and can be done using the Spoon User interface. To access to the Spoon IDE, launch the `spoon` script into the `<data-integration>` folder.

From the main window, on the top right you should see a `Connect` button. Clicking on it, you will see a window appear, similar to the one in the following screenshot.



Click on Get started and fill the `Display name` (digit Alflytics or choose your preferred name) and URL. Be sure you digit the correct Pentaho Analytics Server URL. Click on the `Finish` button to complete the task. Don't connect immediately, but click on the Finish button to go to the main window of the Pentaho Spoon User interface.

Now click again on the `Connect` button on the top right of the main window, and select the new connection (in our case we named `Alflytics`). The login and password requested are the Pentaho Analytics ones (by default `admin` as user Name and `password` as Password).

Once connected, click on the `File` item in the upper menu and then `Open`. Under the following path you can find the root of all the jobs and transformations of the solution.

`/public/Alflytics/ETL`

`Get all` is the main job used to extract the whole Alfresco data. You can studying this, for a better comprehension of how Alflytics works.

5.3.3 Creating the JNDI connection to the Alflytics repository (for publishing reports only)

Reports use a JNDI connection to the data source so it's mandatory to define it before running the first execution. Creating a JNDI connection on Pentaho data Integration is straightforward: edit the file at `<data-integration>/simple-jndi/jdbc.properties` and add the following lines (of course, be sure to replace the right settings for your installation).

```
# If you are using HyperSQL.
Alflytics/type=javax.sql.DataSource
Alflytics/driver=org.hsqldb.jdbcDriver
Alflytics/url=jdbc:hsqldb:hsql://<server>:9001/alflytics
Alflytics/user=<login> (by default admin)
Alflytics/password=<password> (by default password)

# If you are using PostgreSQL.
Alflytics/type=javax.sql.DataSource
Alflytics/driver=org.postgresql.Driver
Alflytics/url=jdbc:postgresql://<server>:5432/alflytics
Alflytics/user=<login>
Alflytics/password=<password>

# If you are using MySQL.
Alflytics/type=javax.sql.DataSource
Alflytics/driver=org.mysql.Driver
Alflytics/url=jdbc:mysql://<server>:3306/alflytics
Alflytics/user=<login>
Alflytics/password=<password>
```

Then, restart Pentaho Data Integration and the JNDI connection is defined.

Publishing reports require two things: the first is to enable the FTP services on Alfresco and the second is to correctly setup the connection parameters into the `DWH_REPORTS` table. To enable FTP on Alfresco add `ftp.enable=true` in `alfresco-global.properties` and restart it. To setup the setup the connection parameters into the `DWH_REPORTS` table, use your preferred DB client and update the fields of the table. Everything is extremely straightforward to do and understand.

5.3.4 Creating and scheduling the scripts to run the ETLs

Using Pentaho Data Integration you will be able to schedule the execution of extractions and publications quite easily. For this purpose, Pentaho Data Integration comes with a nice tool, called `kitchen`, used to run a job/transformation from the command line. The idea is to

share how to invoke `kitchen` for extracting/publishing into Alflytics and then see how to easily schedule the commands, using a CRON (or similar) scheduler.

5.3.4.1 Developing the kitchen command for Alflytics extraction

To launch a Pentaho Data Integration job using kitchen you need a simple syntax. Below the kitchen command to launch the extraction from Alfresco.

```
cd <pentaho_data_integration_path>;
./kitchen.sh
  /rep:"Alflytics"
  /job:"Get all"
  /dir:/public/Alflytics/ETL
  /user:<login> (by default admin)
  /pass:<password> (by default password)
  /log="/<path>/alflytics.log" (choose your preferred path and file name)
  /level:Basic
  /param:alflytics.dbType=hsql (but could be also postgresql or mysql)
  /param:alflytics.getData.alfresco.entity.audits=true
  /param:alflytics.getData.alfresco.entity.processes=true
  /param:alflytics.getData.alfresco.entity.processes.getItems=true
  /param:alflytics.getData.alfresco.entity.repository=true

/param:alflytics.getData.alfresco.entity.repository.removeDeleted=tr
ue
  /param:alflytics.pentaho.protocol=http
  /param:alflytics.pentaho.host=<server>
  /param:alflytics.pentaho.port=<port>
  /param:alflytics.pentaho.login=<login> (by default admin)
  /param:alflytics.pentaho.password=<password>; (by default password)
```

5.3.4.2 Developing the kitchen command for Alflytics publication

To launch a Pentaho Data Integration job using kitchen you need a simple syntax. Below the kitchen command to launch the extraction from Alfresco.

```
cd <pentaho_data_integration_path>;
./kitchen.sh
  /rep:"Alflytics"
  /job:"Report all"
  /dir:/public/Alflytics/ETL
  /user:<login> (by default admin)
  /pass:<password> (by default password)
  /log="/<path>/alflytics.log" (choose your preferred path and file name)
  /level:Basic
  /param:alflytics.dbType=hsql (but could be also postgresql or mysql)
```

`/param:alflytics.pentaho.protocol=http`
`/param:alflytics.pentaho.host=<server>`
`/param:alflytics.pentaho.port=<port>`
`/param:alflytics.pentaho.login=<login>` (by default `admin`)
`/param:alflytics.pentaho.password=<password>;` (by default `password`)

5.3.4.3 Scheduling the kitchen commands

As you can easily imagine, the commands above could be used in a script file and scheduled using CRON (or your preferred scheduler). As a best practice we suggest to schedule this command during the night time, once a day, or on weekly basis according to your needs.

6. How to contribute to the project

How many effort would you require to develop from scratch a solution like Alflytics?
From another point of view: how many time did you save using Alflytics?
How much did you pay for Alflytics?

Because of those simple questions, you should evaluate to contribute to the project.
There are two simple way to give back something to the project:

1) DONATION. Write to Francesco (http://fcorti.com) to know how and be included in the Supporter list.

2) TIME. Replying to answers and giving support to other developers, probably less expert than you in Alflytics, Alfresco or Pentaho.

3) IMPROVEMENTS. If you developed something useful for the other developers like dashboards, reports, new features or bugfix, you should evaluate to give back the artifacts to the community. Don't know how? Write to Francesco (http://fcorti.com).

7. Support and contacts

Having support in an open source project is a value and make the developers happy. Unfortunately it requires a lot of effort and it could be painful especially because the solution is based on a complex stack. Please go through the following paragraphs to understand how to get support for the solution and compress the effort for the best result.

7.1 Disclaimer

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. We can't be responsible for any damage done to your system, which hopefully will not happen.

7.2 Preparing the request of support

In this paragraph is described the way to "package" everything is requested to ask for support in forums or directly to the developers of the solution. Alflytics is a complex environment and to reproduce the use cases (and the errors), the support needs a complex stack of things (Alfresco, Pentaho, etc.). To make the stack easier, Alflytics is able to work also without Alfresco. Let's see below how to define your `alflytics_support_package`.

The `alflytics_support_package` is a ZIP file, composed by three other ZIP files. Below the list of the three ZIP files.

1. The `dwh` ZIP file containing the dump of the `alflytics` warehouse.
    a. If you are using HyperSQL, the dump contains simply the `<pentaho-server>/pentaho-solutions/Alflytics/data` folder. Please remember to zip the folder when Pentaho Analytics Platform is stopped.
    b. If you are using PostgreSql, the dump contains a SQL script useful to rebuild the `alflytics` warehouse in an empty schema. The developer giving support should be able to run the SQL script using PgAdmin3 and have an exact copy of the `alflytics` warehouse in his/her development environment.
    c. If you are using MySql, the dump contains a SQL script useful to rebuild the `alflytics` warehouse in an empty schema. The developer giving support should be able to run the SQL script using MySQL Workbench and have an exact copy of the `alflytics` warehouse in his/her development environment.

2. The `tmp` ZIP containing the `Alflytics` folder in the temporary directory on the file system. The `Alflytics` folder is created during the extraction from Alfresco and contains all the incremental data from the sources. Please be sure the zip file is defined after an extraction with issues or failures. You can find the `Alflytics` folder in `<pentaho-server>/tomcat/temp`, if you are not using Pentaho Data Integration or the temporary directory (`/tmp` in a linux based operating system), if you are using Pentaho Data Integration.

3. The `log` ZIP containing the Pentaho `catalina.out` file. Please include only the parts of the catalina.out related to the extraction with issues or failures.

7.3 How to get support from the community

After you prepared an `alflytics_support_package` ZIP file, feel free to use one of the channels listed below to request for support.

- Alfresco community (https://community.alfresco.com). Raise a question in the official Alfresco community portal describing exactly the issue you are experiencing into your

environment. If you think it could help, attach or link the `alflytics_support_package` ZIP file.

- Github repository of the project (https://github.com/fcorti/Alflytics). Open an issue describing exactly the issue you are experiencing into your environment. If you think it could help, attach or link the `alflytics_support_package` ZIP file.

- Write to Francesco (http://fcorti.com). You can find all the way to contact the principal developer of the solution into his personal website. If you think it could help, attach or link the `alflytics_support_package` ZIP file.

7.4 How to get professional support

If you require professional services for customizations, complex projects and advanced bugfix, write to Francesco (http://fcorti.com). You can find all the way to contact the principal developer of the solution into his personal website.

Appendix - FAQs

In this section you will find some relevant FAQs collected during support or submitted from the users.

FAQ 1 - `SyntaxError: Empty JSON string (script#5)` during extraction task

During the extraction task an error similar to the following can appear.

```
2017/08/20 20:00:24 - getQueries.0 - Finished processing (I=0, O=0, R=1,
W=1, U=0, E=0)
org.pentaho.di.core.exception.KettleValueException:
Javascript error:
SyntaxError: Empty JSON string (script#5)

      at
org.pentaho.di.trans.steps.scriptvalues_mod.ScriptValuesMod.addValues(Script
tValuesMod.java:475)
      at
org.pentaho.di.trans.steps.scriptvalues_mod.ScriptValuesMod.processRow(Scri
ptValuesMod.java:541)
      at org.pentaho.di.trans.step.RunThread.run(RunThread.java:62)
      at java.lang.Thread.run(Thread.java:748)
Caused by: org.mozilla.javascript.EcmaError: SyntaxError: Empty JSON string
(script#5)
      at
org.mozilla.javascript.ScriptRuntime.constructError(ScriptRuntime.java:3785
)
      at
org.mozilla.javascript.ScriptRuntime.constructError(ScriptRuntime.java:3763
)
      at org.mozilla.javascript.NativeJSON.parse(NativeJSON.java:146)
      at org.mozilla.javascript.NativeJSON.execIdCall(NativeJSON.java:122)
      at
org.mozilla.javascript.IdFunctionObject.call(IdFunctionObject.java:129)
      at
org.mozilla.javascript.optimizer.OptRuntime.call1(OptRuntime.java:66)
      at org.mozilla.javascript.gen.script_3._c_script_0(script:5)
      at org.mozilla.javascript.gen.script_3.call(script)
      at
org.mozilla.javascript.ContextFactory.doTopCall(ContextFactory.java:426)
      at
org.mozilla.javascript.ScriptRuntime.doTopCall(ScriptRuntime.java:3178)
      at org.mozilla.javascript.gen.script_3.call(script)
      at org.mozilla.javascript.gen.script_3.exec(script)
      at
org.pentaho.di.trans.steps.scriptvalues_mod.ScriptValuesMod.addValues(Script
tValuesMod.java:388)
      ... 3 more
```

```
...

2017/08/20 20:00:24 - Get queries - ERROR (version 7.1.0.0-12, build 1 from
2017-05-16 17.18.02 by buildguy) : Errors detected!
2017/08/20 20:00:24 - Get queries - Transformation detected one or more
steps with errors.
2017/08/20 20:00:24 - Get queries - Transformation is killing the other
steps!
2017/08/20 20:00:24 - Get queries - ERROR (version 7.1.0.0-12, build 1 from
2017-05-16 17.18.02 by buildguy) : Errors detected!
2017/08/20 20:00:24 - Get all - Starting entry [Write To Log 2]
2017/08/20 20:00:24 - Query retrieve failure. - Cannot retrieve queries
using the Pentaho REST services on 'http://localhost:8080'.
2017/08/20 20:00:24 - Query retrieve failure. - Please check the correct
URL before requesting support.
2017/08/20 20:00:24 - Query retrieve failure. - If the URL (port included)
is wrong, update the ETLs using the Spoon User Interface.

...
```

In case you get this error, the `Get queries` transformation cannot retrieve the query to be executed. `Get queries` transformation retrieves the query using a Pentaho REST service. By default the installation assumes that Pentaho is available at `http://localhost:8080`. if you will use a different port (or URL), you have to update the default parameters of the ETLs using Spoon. Then you can launch the extraction again.

Appendix II - Previous versions

TODO