Projet HUB

Pizza Team BDDF

Exported on 06/28/2018

1 Table of Contents

Target release	
Epic	
Document status	DRAFT
Document owner	
Architect	Jerome DUCOURTIOUX[1], David Sigogne
Developers	Phuong PHAM THI MAI[2]
Integrators	
Datamanagment	Masisa DOMBOLO[3]
Security	Mickael Akkani

1 http://dgitlx33.dns21.socgen:8090/display/~A372915
2 http://dgitlx33.dns21.socgen:8090/display/~A469255
3 http://dgitlx33.dns21.socgen:8090/display/~A383489

2
Goals

2.1 The goal is to implement a web application allowing an administrator to configure a scheduled extraction of datas based on the requirements (columns of interest, frequency, destination) made by an end user that will consume it either directly or through a third-part application

3 Use case N°1

An administration user should be able to enter parameters describing the extraction required by a final user that will consume the extraction.

Those parameters will be stored to be able to modify the extraction process or for audit purpose

A scheduled treatment based on those parameters will allow the creation of tables containing datas (stored in the Hub project) upon which a file (in the format required by the user) is created and stored in a specific project on HDFS

4 HubProject[4]

4 http://dgitlx33.dns21.socgen:8090/download/attachments/10322819/PojetHub_v1.pptm?api=v2&modificationDate=1529406852004&version=3

5 HUB-Specifications.docx[5]

5 http://dgitlx33.dns21.socgen:8090/download/attachments/10322819/HUB-Specifications.docx?api=v2&modificationDate=1528734030145&version=1

6 Assumptions

An end user acting as an administrator should be able through an interface to select

- a source
- a set of columns
- the frequency of updating
- the format of the extraction
- the destination of the extraction (according to LDAP and rangers habilitation).

The source of the data displayed through the IHM will come from the datamanagement catalog.

All the subscription should be stored in a database so that :

- An audit could be done
- A search for already existing subscriptions souhld be done
- An updating of a subscription already existing should be done

The direct user of the interface will be an administrator, not the final consumer of the extration

The result of the extraction should be a file of any format containing a set of columns updated according to the frequency entered by the user put in a project space accessible by the consumer

The direct consumer of the extraction could be a collaborator habilitated to the project space or a third application

Security :

The application should ask for authentification

The application should act as a specific applicative account (one for each entity). The applicative account should have rights to access all the projects, and a set of application directories (updated on demand)

A job father will be deployed in production. This job will be abe to create job child according to the subscription parameters. (Today the application will be able to deploy a single job directly for one extraction)

The destination could be either his mailbox or a project he is habilitated to or in a long term a third application or even in a datalab

This service will be developped on the BDDF perimeter at first but could be extended to other entities.

The interface will be used by an administrator within the CCBD not directly by the end consumer

On the go of the security to store a data flow on the datalake, the configuration will be done by the administrator

If a new subscription appear, the job father will be able to launch of job child accordingly to the subscription

If a subscription is modified (modification of the set of columns for instance), the job father will reflected on the job child accordingly

The job father will be deployed in production at the start of the project.

There will be one subscription for each file collected. => if a new file is collected in a same directory the user will have to ask fo a new subscription

The job father will create a job child so that the job will be scheduled to extract data from a source and copy it in a project space

The application will have to test the rights of the 3D-like application account and not the right of the administrator himself

Several profiles will be implemented to isolate the usage. For instance, a 3DFPHADM account will be created to be able to all the BDDF lake-directories and BDDF-projects directories (in the limit that the dataflow has no >C2 datas)

The extraction results of the subscription will be stored in a specific project of in a datalab

Pré-requis :

Création GIT, coordinator oozie,ability to query Rangers

A treatment

Technologies : oozie, Hive, Postgresql, python, html, css, js, spark

7 Requirements

#	Title	User Story	Importance	Notes
1	Using parameters that describe where is the source in the lake and the datas expected to be exposed from that source è extract the datas and store it			
2	Build the interface to expose • data source to allow end user choosing the data source in witch he wants to command data extraction • datas from the data source choosen to allow end user choosing the datas he want to extract			
3	Add right levels in order to expose information regarding who is the end-user connected			
4	Add encryption capabilities using and perhaps adapting some existing solution			
5	Add other function (like proposing datasource…)			

8 User interaction and design

Actions :

Define the data model of the database

9 Questions

Below is a list of questions to be addressed as a result of this requirements document:

Question	Outcome

10 Not Doing