Creació àgil d'un clúster Big Data

Eric Ciutad Castejon

Resum—Els sistemes Big Data han crescut exponencialment en popularitat aquests últims anys ja que aporten informació molt valuosa a les organitzacions que poden explotar-la. Malauradament implementar aquest sistema és una tasca molt complexa que té un cost molt elevat. Amb aquest projecte es pretén reduir el cost de l'implementació d'un sistema Big Data construint un entorn de proves, on poder executar aplicacions distribuïdes sense la necessitat d'obtenir una gran infraestructura per fer-ho. En aquest entorn tindrem un clúster en una sola màquina amb múltiples contenidors Docker que actuaran com a nodes. A aquests contenidors se'ls hi assignarà un rol dins del clúster depenent de les necessitats del client. La instal·lació es farà de manera ràpida gràcies a l'automatització del procés, reduint així els costos en experts, infraestructura i temps d'instal·lació. També es farà una segona aplicació que demostrarà el correcte funcionament del clúster creat.

Paraules clau—Big Data, Hadoop, HDFS, Flume, Clúster, Node, Cloudera, Docker, Contenidor, Java

Abstract—Big Data systems have grown exponentially in popularity during these past few years. They allow organizations to get valuable information for them to exploit. Unfortunately implementing this system is a very complex task and has a very high cost. The goal of this project is to reduce the cost of the implementation of a Big Data system by building a testing environment, where you can run distributed applications without the need to obtain a large infrastructure to do so. In this environment we will have a single-machine cluster with multiple Docker containers acting as nodes. Depending on the client's needs theses nodes will have different roles. The installation will be fast thanks to the automation of the process, reducing costs in experts, infrastructure and deployment time. There will also be a second application that will demonstrate the functionality of the created cluster.

Index Terms— Big Data, Hadoop, HDFS, Flume, Cluster, Node, Cloudera, Docker, Container, Java

1 Introducció

1.1 Introducció preliminar

AQUEST projecte ha estat proposat per l'empresa on estic treballant actualment, Everis. És una consultora multinacional que pertany al grup NTT DATA que ofereix solucions tecnològiques i de negoci als seus clients. La companyia treballa en els sectors de telecomunicacions, entitats financeres, indústria, energia, sanitat i administració pública. NTT DATA és la sisena companyia de serveis de IT del món amb més de 70000 professionals i augmenta les capacitats i recursos de Everis per tal de donar un servei millor i més innovador als seus clients.

1.2 Introducció del projecte

Avui en dia es generen una gran quantitat de dades. Aquest gran volum dades, que poden provenir de diferents fonts i tenir una gran varietat, es poden tractar mitjançant Big Data per obtenir i generar valor d'una manera veloç podent així obtenir valuosa informació sobre aquestes en el moment precís.

Però què és Big Data? Andrea de Mauro[8] argumenta que Big Data és tota la informació que requereix d'una tecnologia i mètodes analítics específics per poder extreure'n un valor, caracteritzada per les 4 V: Volum, Velocitat, Varietat i Veracitat. Gràcies a aquestes característiques normalment es requereix d'una agrupació de màquines físiques o virtuals que actuen com un sol sistema per obtenir un gran paral·lelisme anomenat clúster. Les màquines que formen part del clúster s'anomenen nodes.

¿Quina és la diferència entre tractar dades amb Big Data i tractar-les amb un sistema tradicional? Un sistema de bases de dades tradicional troba cada cop mes difícil tractar dades a mesura que aquestes creixen ja que no té integrat el concepte de treballar de manera distribuïda. Així doncs per analitzar-les, processar-les i carregar-les en una base de dades de manera tradicional es trigaria massa temps.

Si les empreses poguessin analitzar totes aquestes dades que poden disposar, obriria a aquestes noves oportunitats de negoci. Andrew McAfee i Erik Brynjolfsson argumenten que les empreses que utilitzen Big Data son, de mitja, un 5% més productives i un 6% més rentables que els seus competidors [7]. Malauradament crear i implementar un sistema Big Data es una tasca molt complexa que requereix de molts recursos i infraestructura.

- *E-mail de contacte: eric.ciutad@e-campus.uab.cat*
- *Menció realitzada: Enginyeria del Software.*
- *Treball tutoritzat per: Daniel Ponsa Mussarra (Ciències de la computació)*
- *Curs 2017/18*

Amb aquest projecte es pretén agilitzar el temps d'instal·lació i reduir el cost de la implementació d'un sistema Big Data per a un possible client al automatitzar la creació d'un clúster d'una única màquina física real i disposarà de contenidors Docker com a nodes on es puguin realitzar proves. D'aquesta manera el client disposarà d'un entorn on poder executar aplicacions distribuïdes creat de manera simple, sense la necessitat d'adquirir una gran infraestructura ni de mantenir a experts en el sector. Aquest client podrà ser extern, que no forma part de Everis, o intern, que si que en forma part.
Aquest client explicarà totes les seves necessitats a l'administrador de l'aplicació. Proporcionarà una sola màquina física o virtual amb uns requisits específics per realitzar la instal·lació.

Així doncs, en aquest projecte es desenvoluparan dues aplicacions Java que realitzaran la creació i instal·lació automatitzada d'un clúster personalitzat amb les preferències de l'usuari. Un cop finalitzat el desenvolupament d'aquestes dues aplicacions es farà un demostrador per poder comprovar el seu correcte funcionament.
Finalment s'ha plantejat ampliar el projecte per a poder realitzar la instal·lació i desplegament dels nodes en una plataforma Cloud per a poder obtenir un entorn funcional on el rendiment sigui real.

2 Estat de l'art

Actualment moltes empreses estan interessades en el Big Data, però no totes tenen els recursos necessaris per fernen ús. Implementar un sistema Big Data comporta els següents costos:

- Software i Infraestructura: La infraestructura bàsica consisteix en magatzems de dades, servidors, xarxes i eines de monitorització. Utilitzar i mantenir aquestes eines pot comportar un cost molt elevat.

- Desenvolupadors: La implementació i utilització d'un sistema Big Data requereix d'especialistes i persones amb el coneixement necessari i experiència en el sector que puguin aportar les solucions desitjades.

- Temps d'implementació: Durant la instal·lació d'un sistema Big Data, es requereixen moltes validacions i instal·lacions prèvies abans de tenir un sistema completament funcional. Això fa que es requereixi de molt de temps per realitzar la instal·lació manualment.

Com s'ha mencionat aquest projecte automatitza tot el procés d'instal·lació d'un clúster, per lo que aquests 3 factors es veurien reduïts. Això és així gràcies a que disposarem només d'una màquina reduint així costos en infraestructura. A més només caldrà comentar-li els requisits a l'administrador reduint així els costos en experts i l'automatització fa totes les comprovacions i instal·lacions requerides, reduint així el cost en temps d'implementació.

3 Objectius

En aquest apartat es defineixen els objectius principals del projecte i tots els seus subobjectius.

L'objectiu principal del projecte és construir i demostrar el funcionament d'una aplicació capaç d'automatitzar la instal·lació d'un clúster de manera senzilla amb la distribució de Cloudera Hadoop. Podem desglossar aquest objectiu en diferents subobjectius:

3.1 Objectiu 1: Instal·lació de l'entorn de desenvolupament i aprenentatge de la configuració manual d'un clúster amb Cloudera

En aquesta fase s'ha d'emular la funcionalitat final del projecte de manera manual i s'han de solucionar tots els problemes de connectivitat entre nodes i tots aquells que puguin sorgir durant la instal·lació per posteriorment poder automatitzar el procés. També s'ha de realitzar la instal·lació de l'entorn de desenvolupament que s'ha d'utilitzar i de totes les eines necessàries per poder dur a terme el projecte.

- **Objectiu 1.1: Formació bàsica en l'eina Docker**

L'eina principal a utilitzar en aquest projecte és Docker.
S'utilitza Docker per poder fer servir diferents contenidors per a que actuïn com a nodes del nostre clúster. Però per què utilitzem contenidors Docker i no màquines virtuals?
Un contenidor Docker és un paquet executable d'una peça de software, que conté tot lo necessari per executar-se. Gràcies a això els contenidors es poden executar en diferents entorns ja que comparteixen diferents recursos del sistema operatiu de la màquina on s'estan executant, a diferència de les màquines virtuals que l'inclouen. Això fa que siguin molt menys pesats. En la següent figura podem veure representat gràficament les diferències entre les dues.

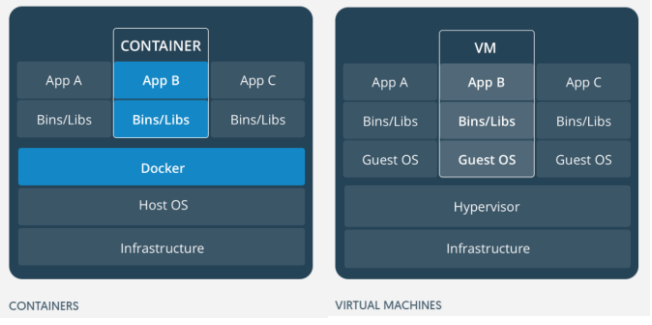

Figura 1: Diferències entre contenidors Docker i màquines virtuals. [2]

Encara que s'utilitzarà una API de Java per gestionar els contenidors, els primers tests s'han de fer des d'un terminal. Així doncs s'ha d'aprendre a utilitzar l' eina de Docker de manera efectiva per al bon desenvolupament del projecte.

- **Objectiu 1.2: Emular la funcionalitat de l'instal·lador manualment**

Amb uns coneixements bàsics de Docker podem començar a crear el primer contenidor utilitzant una imatge de Cloudera Docker. La imatge de Cloudera Docker és una distribució de la companyia Cloudera que inclou CDH [4] i Cloudera Manager, que ens facilita la prova, desplegament i creació de nous nodes i clústers. També podem observar l'estat d'aquests nodes i ajudar-nos gràficament i interactivament a crear-los i gestionar-los.
Un cop tenim un contenidor funcional de Cloudera Docker hem de ser capaços d'afegir nodes i comunicar-los amb el node de Cloudera per a que aquest pugui gestionar-los. Aquests nodes han de ser creats a partir d'una imatge base on hi ha instal·lades totes les parcel·les dels serveis de Cloudera. Les parcel·les són distribucions formades per els fitxers i totes les dades addicionals que utilitza Cloudera Manager per instal·lar els serveis de CDH als nodes.
El contenidor de la imatge del node base ha de ser capaç de ser afegit al clúster des de Cloudera Manager. Un cop afegit correctament se l'hi ha de poder assignar un rol dins d'aquest clúster. Un rol és la competència que tindrà un node dins d'un clúster.

3.2 Objectiu 2: Crear l'instal·lador d'arquitectures Big Data

L'objectiu principal del projecte és el desenvolupament d'una aplicació Java que anomenarem Instal·lador d'arquitectures Big Data.
Després de crear un entorn funcional on puguem comunicar múltiples nodes amb el principal, ja podem començar a crear l'aplicació Java de l'instal·lador.
L'instal·lador ha de crear diferents nodes, que en comptes de ser màquines físiques o virtuals seran contenidors Docker.
Un cop creats els contenidors els haurà d'afegir al clúster. Finalment assignarà rols a aquests contenidors per a que puguin realitzar algun servei depenent dels requisits del client. En el desenvolupament del projecte s'han aplicat dues restriccions:

- El projecte s'ha de desenvolupar en Java per a generar de manera ràpida i automàtica el desplegament del clúster.
- S'ha de crear el nostre clúster amb una distribució de Cloudera de Apache Hadoop, CDH. Aquesta distribució ofereix diferents serveis 100% open source complementaris a Hadoop.

Hadoop és una plataforma altament escalable, dissenyada per processar grans volums de dades en un gran nombre de nodes operant en paral·lel.
En la Figura 2 podem veure una perspectiva general del funcionament de l'instal·lador.

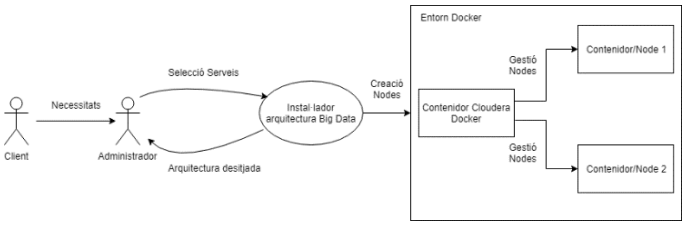

Figura 2: Perspectiva de l'instal·lador.

Aquesta aplicació ha de disposar d'una interfície gràfica que s'encarregarà de demanar tots els inputs necessaris. S'ha de poder seleccionar els diferents serveis [4] que necessitem per al nostre clúster i el número de nodes que volem per a cada un d'ells. Un cop seleccionats s'ha de realitzar la instal·lació. Aquests serveis els podem veure a la secció 6 sota la secció 'Serveis CDH'.
Aquesta s'ha de fer en una sola màquina on es crearà el clúster i s'afegiran tots els nodes (contenidors Docker) seleccionats i on es podran fer proves respecte la seva funcionalitat. Així doncs, no es tindrà un rendiment d'un entorn distribuït real, ja que no es disposa realment d'aquestes màquines i recursos. L'instal·lador esta pensat per a que sigui utilitzat per un administrador, que prèviament haurà recollit tots els requisits dels clients per saber el sistema a crear.

- **Objectiu 2.1: Dissenyar la interfície gràfica**

L'aplicació ha d'estar formada per diferents finestres. Aquestes demanaran diferents inputs a l'usuari necessaris per la creació de la seva infraestructura. Per a realitzar la interfície primer s'ha de dissenyar un prototip en paper.

- **Objectiu 2.2: Implementació de l'instal·lador**

S'ha de desenvolupar el codi per a poder fer la instal·lació de l'arquitectura desitjada a una màquina amb Docker instal·lat.
Per desenvolupar la interfície s'utilitzarà la llibreria Java Swing. Amb la API de Docker per Java es podran gestionar les imatges i contenidors durant tota la gestió de Docker. La API de Cloudera darà possible la connexió i assignació de rols als nodes durant la instal·lació i gestió del clúster.

3.3 Objectiu 3: Crear una aplicació demostrador per comprovar el correcte funcionament de l'instal·lador

Per a comprovar el correcte funcionament de l'instal·lador s'ha de crear una altre aplicació Java que llegirà dades de Twitter i les emmagatzemarà de manera distribuïda utilitzant una arquitectura específica creada prèviament per l'instal·lador. Aquesta aplicació s'utilitzarà per demostrar que l'instal·lador ha creat una infraestructura funcional de forma automàtica. També es podrà veure una primera part del cicle del Big Data, la captura de dades.

3.4 Objectiu 4: Migració de l'entorn d'execució local a Cloud

Amb una arquitectura distribuïda desplegada en un sol node es podrà veure i provar si l'arquitectura és funcional, però el rendiment no serà el mateix que el d'un clúster real, es a dir, un clúster amb més d'una màquina física. Per aquesta raó es vol ser capaç de crear aquesta arquitectura en un entorn Cloud on poder disposar d'un entorn Big Data real.

4 PLANIFICACIÓ

El projecte s'ha dividit en diferents tasques, que han estat planificades al llarg del temps de l'assignatura. La planificació inicial es va modificar depenent de la durada real de les tasques i el temps que es va tardar en completar-les. A la Figura 14 de l'annex podem veure un Diagrama de Gantt que representa la planificació final.

Tasques

En aquest apartat es llisten les tasques realitzades i l'objectiu a la que pertanyen.

Les tasques relacionades amb l'Objectiu 1 son:

- Recol·lecció dels requisits del sistema
  - o Reunions amb el tutor de l'empresa per poder extreure els requisits principals del projecte.
- Instal·lació entorn de treball
  - o Instal·lar VMware i crear màquina virtual
  - o Creació de la màquina virtual amb CentOS 7
  - o Instal·lar Docker a la màquina virtual
- Formació Docker
  - o Aprenentatge funcionament Docker
- Aprenentatge de la configuració manual d'un clúster Big Data amb Cloudera
  - o Executar imatge Cloudera Docker
  - o Crear imatge base per a que el contenidor de Cloudera pugui afegir el contenidor resultant com a node
  - o Afegir nou contenidor com a node del clúster i establir connexió

Les tasques relacionades amb l'Objectiu 2 son:

- Desenvolupament de l'instal·lador
  - o Disseny interfície
  - o Desenvolupar aplicació per gestionar contenidors Docker utilitzant la API de Docker per Java.
  - o Desenvolupar interfície gràfica per a la selecció de les diferents tecnologies a escollir.
  - o Desenvolupar aplicació utilitzant la API de Cloudera per gestionar el clúster.

Les tasques relacionades amb l'Objectiu 3 son:

- Desenvolupament Demostrador
  - o Disseny de l'aplicació Java de tractament de dades de Twitter utilitzant l'arquitectura resultant.
  - o Desenvolupar fitxer de configuració de Flume.
  - o Desenvolupar codi per l'aplicació Java per realitzar l'extracció de dades de Twitter.

Les tasques relacionades amb l'Objectiu 4 son:

- Ampliació de la funcionalitat de l'instal·lador
  - o Documentació i selecció plataforma Cloud
  - o Ampliació de l'aplicació per a executar-se en entorns Cloud

5 METODOLOGIA

Totes les tasques completades mencionades a l'apartat anterior han estat realitzades de manera lineal seqüencial ja que majoritàriament no es pot començar una sense acabar l'anterior perquè hi ha dependències entre elles.
Tot i així hi ha excepcions:

- La formació en Docker es va realitzar intercaladament amb altres tasques, ja que es va anar fent a mesura que es necessitaven certs coneixements per poder gestionar l'entorn Docker.
- Les tasques 'Creació de la imatge base' i 'Afegir-la com a node de Cloudera' també es van realitzar intercalades, ja que molts dels problemes trobats al intentar afegir-la s'han solucionat modificant la imatge.

Cada 2 o 3 setmanes aproximadament s'ha realitzat una reunió amb el tutor acadèmic per revisar els informes realitzats per poder aplicar els seus comentaris i corregir-ho a temps per les entregues.
Les reunions amb el tutor de l'empresa s'han realitzat sempre que han sorgit dubtes o s'ha acabat una tasca.

Durant el desenvolupament d'aquest projecte s'han utilitzat 3 repositoris diferents per a diferents funcionalitats:

- DockerHub: Repositori de Docker ha permès emmagatzemar i mantenir un control de versions sobre les imatges creades.
- Bitbucket: Repositori basat en Git on s'ha gestionat tota la documentació creada des de el principi del projecte.
- Repositori Everis: Repositori privat de l'empresa on s'ha mantingut un control de versions de les 3 aplicacions Java del projecte.

6 EINES UTILITZADES

Les eines principals que s'han utilitzat per a la realització d'aquest projecte son:

STS: Spring Tool Suite, entorn de desenvolupament basat en Eclipse modificat per al desenvolupament d'aplicacions Spring. S'utilitza en les tasques relacionades amb l'Objectiu 2, 3 i 4.

Maven: És una eina per la gestió i construcció de projectes Java. Utilitza un fitxer xml (pom.xml) on descriu totes les dependències del projecte a altres components externs. S'utilitza en les tasques relacionades amb l'Objectiu 2, 3 i 4.

Docker: Docker és una tecnologia que ens permet construir, executar i testejar aplicacions basades en Linux mitjançant l'empaquetament d'aquestes en contenidors. Aquests contenidors disposen de tot lo necessari per executar l'aplicació permetent així separar aquesta de la infraestructura. S'utilitza en les tasques relacionades amb l'Objectiu 1, 2, 3 i 4.

Apache Hadoop: Framework que permet el processament de grans volums de dades a través de sistemes distribuïts. Inclou el seu propi sistema de fitxers HDFS. S'utilitza en les tasques relacionades amb l'Objectiu 2, 3.

Cloudera Docker: Imatge Docker que inclou CDH i Cloudera Manager. S'utilitza en les tasques relacionades amb l'Objectiu 1 i 2.

CDH: És una distribució de Cloudera que inclou Apache Hadoop integrat amb altres tecnologies com Flume, HBase, Hive, Hue, Cloudera Impala, Spark, etc. És una de les distribucions més populars de Hadoop. S'utilitza en les tasques relacionades amb l'Objectiu 2 i 4.

Cloudera Manager: Aplicació que permet gestionar clústers de CDH de manera senzilla. Ajuda a automatitzar la instal·lació de diferents serveis i nodes per reduir el temps total i també inclou eines de diagnòstic. [1] S'utilitza en les tasques relacionades amb l'Objectiu 1, 2 i 4.

Serveis CDH

HDFS (Hadoop Distributed File System): Sistema d'arxius distribuït de Hadoop. És escalable i té una alta disponibilitat ja que replica la informació en diferents nodes. S'utilitza en les tasques relacionades amb l'Objectiu 2 i 3.

Flume: Sistema per obtenir, agregar i moure grans quantitats de dades des de diferents fonts [5]. S'utilitza en les tasques relacionades amb l'Objectiu 2 i 3.

Impala: Motor de consultes SQL per el processament en paral·lel de dades emmagatzemades en un clúster. S'utilitza en les tasques relacionades amb l'Objectiu 2.

YARN (Yet Another Resource Negotiator): Tecnologia que permet la gestió de recursos i la planificació de tasques. S'utilitza en les tasques relacionades amb l'Objectiu 2.

HBase: Base de dades distribuïda no relacional i escalable de Hadoop. Dissenyada per donar accés a taules amb milions de registres. Hi ha tecnologies que s'integren amb HBase per oferir un accés SQL a aquestes dades. S'utilitza en les tasques relacionades amb l'Objectiu 2.

7 TREBALL REALITZAT

Abans de començar a desenvolupar l'aplicació principal es va realitzar tota la funcionalitat final de l'instal·lador manualment. Això va servir per tenir un primer contacte amb totes les eines i trobar possibles problemes i solucions per al futur.
Primerament es van instal·lar totes les eines de desenvolupament. Es va fixar utilitzar CentOS, un sistema operatiu Linux per poder instal·lar Docker i crear el clúster en un portàtil proporcionat per l'empresa. Com aquest disposa d'un sistema operatiu de Windows es va utilitzar una màquina virtual creada mitjançant VMware.
Un cop instal·lades totes les eines es van crear a partir d'un terminal tots els contenidors necessaris de Docker. Hi ha dos contenidors principals: el contenidor de Cloudera Docker i el contenidor base que es fa servir per crear els diferents nodes.
El contenidor de Cloudera Docker serveix per iniciar el clúster amb uns serveis i rols mínims assignats automàticament a aquest mateix contenidor.
El contenidor base es un contenidor amb la imatge del sistema operatiu CentOS on es van realitzar diferents personalitzacions:

- Es van instal·lar els serveis de SSH per poder accedir als contenidors remotament.
- Es va instal·lar el servei VIM per poder modificar diferents fitxers.
- Es va crear un usuari amb permisos root i que no requerís contrasenya a l'hora d'executar comandes.
- Es van instal·lar diferents serveis necessaris per que aquest contenidor pogués ser afegit i gestionat pel clúster.
- Es van instal·lar totes les parcel·les de Cloudera.

Un cop creats i executats tots els contenidors ja es va poder crear el clúster complet mitjançant Cloudera Manager.
Al mateix temps que realitzava aquesta tasca es va poder aprofundir en els coneixements sobre Docker.
Un cop clara la funcionalitat principal es va començar el desenvolupament de l'instal·lador.
Al començament d'aquesta fase es van fer vàries reunions amb els tutor de l'empresa i el tutor acadèmic per definir les funcionalitats del projecte. Com a resultat es va realitzar

un document de visió on s'identifiquen els stakeholders i els requisits.
En la Figura 3 es pot veure la funcionalitat bàsica de l'instal·lador a partir del diagrama de casos d'ús.

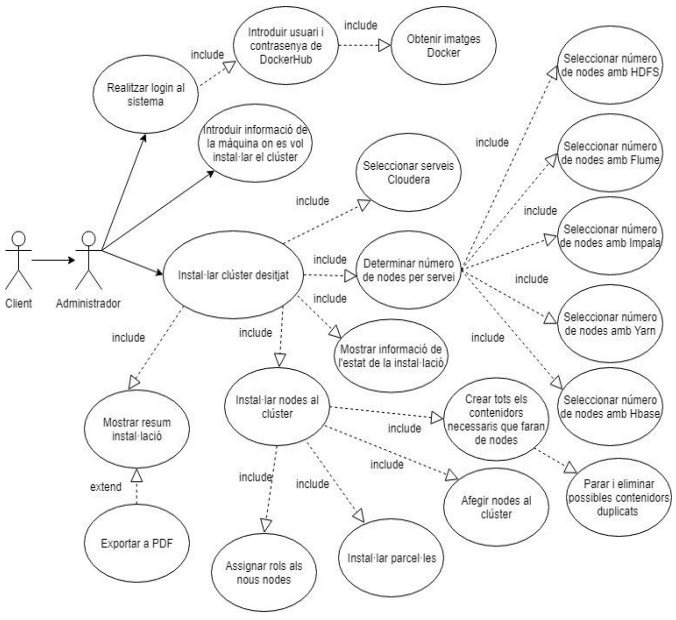

Figura 3: Diagrama de casos d'ús

Com l'aplicació requereix d'una interfície es va crear inicialment un prototip en paper amb el tutor de l'empresa, que posteriorment va ser revisat pel tutor acadèmic. A la Figura 13 de l'Annex podem veure el prototip.
El desenvolupament Java s'ha fet mitjançant l'entorn eclipse. La creació de la interfície d'usuari s'ha realitzat amb la llibreria de Java Swing. La gestió dels contenidors ens la permet gestionar la API de Docker per Java. La gestió del clúster la permet gestionar la API de Cloudera Manager per a Java. Les dependències d'aquesta API s'han afegit a través de Maven.
En un principi l'instal·lador era només una sola aplicació Java, però per problemes de dependències internes incompatibles entre del dues API principals es va dividir en dues diferents. Una aplicació per realitzar tota la gestió de contenidors Docker i una altre per gestionar el clúster. Aquestes s'executen seqüencialment a partir d'un únic script bash.
En la Figura 4 podem veure el flux d'execució de l'instal·lador.

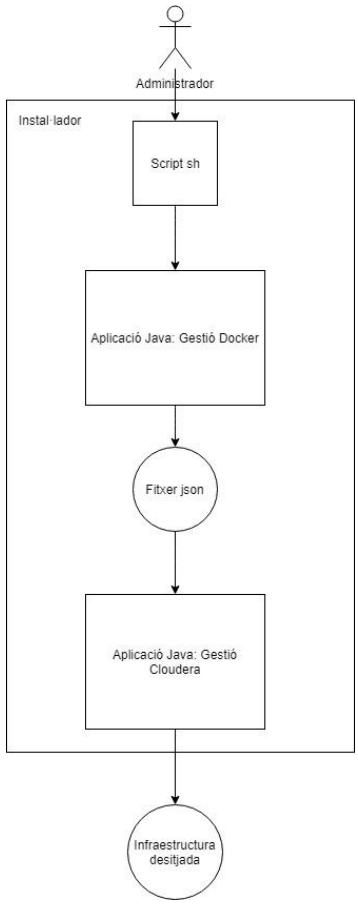

Figura 4: Flux d'execució de l'instal·lador.

La funció principal de l'aplicació de gestió Docker és l'automatització de la creació, execució i configuració de tots els contenidors Docker necessaris per a realitzar la instal·lació.
També s'encarrega d'executar la interfície per recollir tots els inputs de l'usuari.

Els requisits funcionals principals de la interfície son els següents:

- Tan aviat com l'usuari inicïi l'instal·lador, el sistema ha de mostrar una finestra de 'log in' per accedir al repositori de Docker on hi ha les imatges guardades.
- Tan aviat com s'hagin introduït les credencials de DockerHub, el sistema ha de mostrar una finestra que demani les credencials de la màquina on es vol instal·lar el clúster.
- Tan aviat com el sistema inicïi el contenidor de Cloudera, aquest ha de mostrar una llista de serveis disponibles i on es podrà seleccionar el número de nodes que es vol per a cada un.
- Si el sistema no està en una finestra on demana inputs d'usuari, aquest ha de mostrar una pantalla de progrés per mostrar l'estat de la instal·lació.
- Tan aviat com es finalitzi la instal·lació, el sistema mostrarà un resum de tots els serveis i nodes instal·lats que es podrà exportar a PDF.

A l'usuari se l'hi demana la següent informació:

- Usuari i contrasenya de DockerHub per poder accedir i descarregar les imatges Docker.
- Un usuari i contrasenya de la màquina on es vol realitzar al instal·lació.
- Adreça IP d'aquesta màquina.
- Número de nodes per servei que desitgem que tingui el nostre clúster.

Aquests es guarden en un fitxer json temporal per a un posterior ús.
En la següent part podem veure diferents finestres on es mostra el flux d'execució.
Primerament l'instal·lador demana que s'introdueixi l'usuari i contrasenya de DockerHub per poder descarregar les imatges. Això es pot veure a la Figura 5:

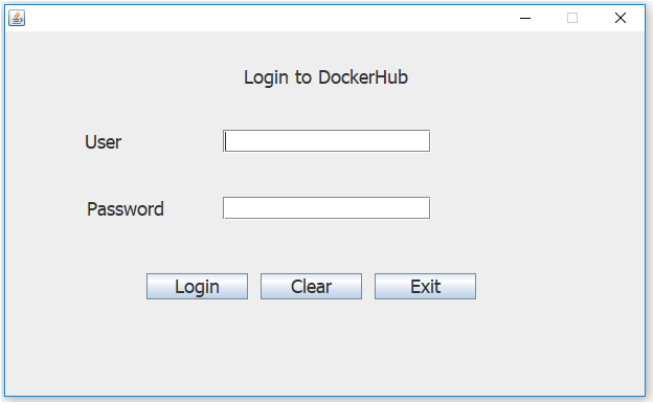

Figura 5: Finestra login Docker

Un cop omplerta la primera finestra, l'instal·lador demana que s'introdueixin totes les dades relacionades amb la màquina on es vol instal·lar el clúster. Es pot observar a la següent figura:

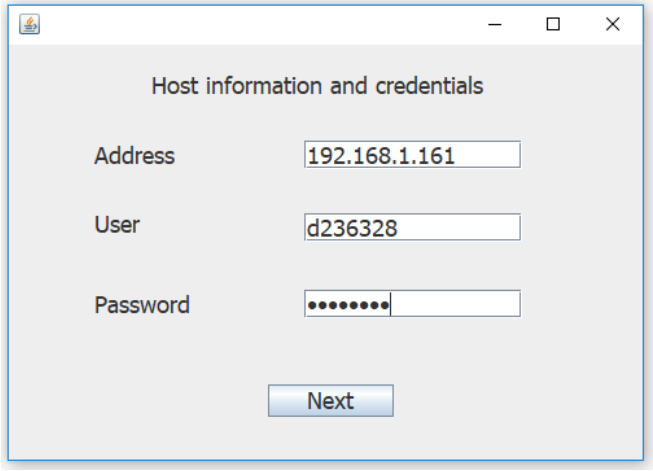

Figura 6: Finestra d'informació de la màquina on es vol realitzar la instal·lació.

Un cop omplerta tota la informació l'aplicació procedirà a iniciar el contenidor principal de Cloudera. Aquest procés no és immediat, per lo que s'ha creat una finestra de càrrega que mostra informació referent al pas en que es troba la instal·lació. Es pot veure a la figura 7:

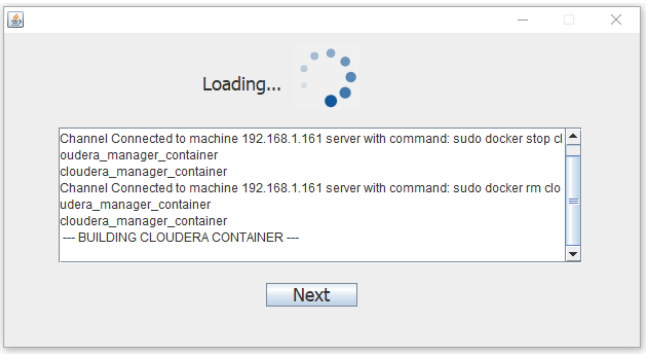

Figura 7: Finestra de càrrega de la creació del contenidor de Cloudera.

Un cop creat el contenidor de Cloudera l'instal·lador demanarà el número de nodes que es vol per a uns serveis específics. Aquests serveis estan llistats a l'apartat 6 sota la secció 'Serveis CDH'.

En la següent figura es pot veure la pantalla de selecció:

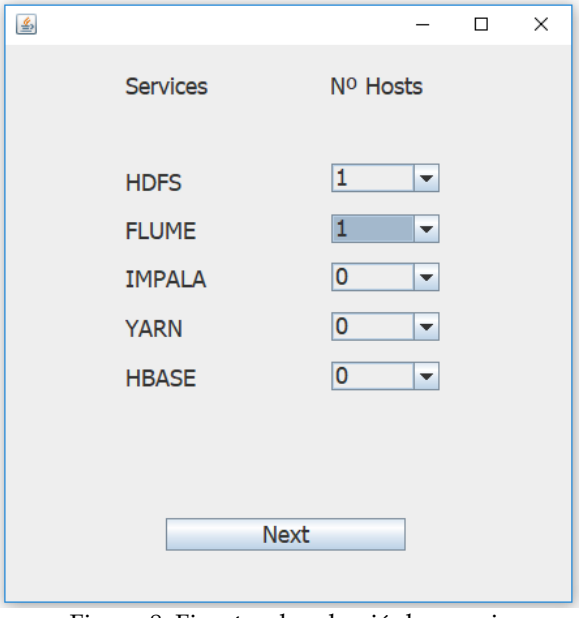

Figura 8: Finestra de selecció de serveis

Un cop seleccionats l'instal·lador començarà a crear tots els contenidors especificats i inicialitzarà Cloudera Manager en el contenidor de Cloudera. Se'ns mostra una altre finestra de càrrega com es pot veure a la Figura 9:

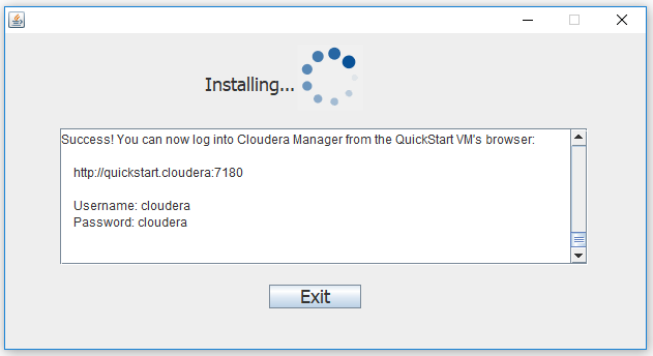

Figura 9: Finestra de càrrega de la creació dels contenidors especificats.

Un cop finalitzat, es guarda en un fitxer json tota la informació introduïda per l'usuari. Aquí conclou l'execució de l'aplicació de gestió Docker.

L'aplicació de gestió de Cloudera llegeix el fitxer json per saber com ha de realitzar la configuració. Posteriorment s'encarrega d'afegir tots els contenidors a l'entorn de Cloudera, es a dir, fer-los visibles des de el node principal (contenidor Cloudera Docker), afegir-los al clúster com a nous nodes, configurar-los i assignar-los a un rol. Mostrarà una altre pantalla de càrrega que es pot veure a la Figura 10:

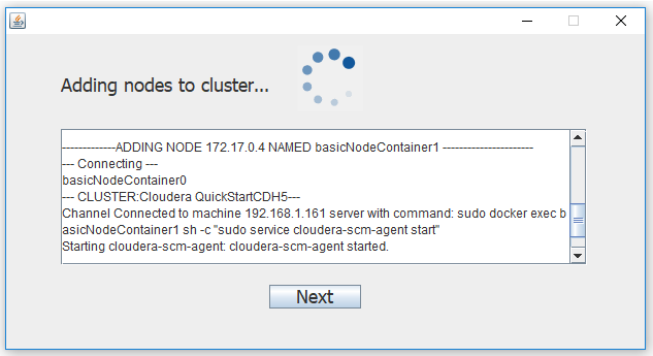

Figura 10: Pantalla de càrrega final

Un cop finalitzada la instal·lació es mostrarà un resum amb tota la informació del clúster que podrà ser exportada a PDF si es desitja. Es pot veure a la figura 11:

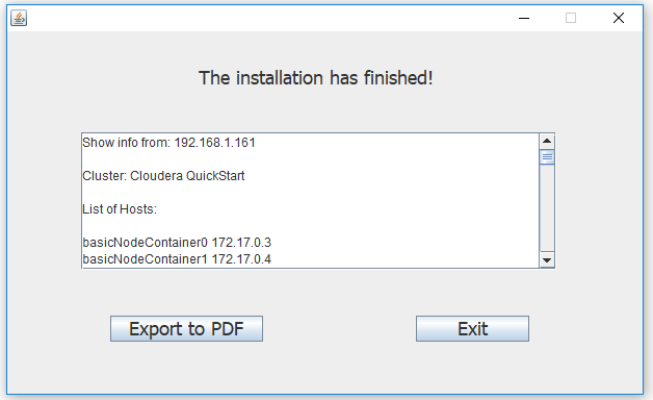

Figura 11: Finestra que mostra resum de la instal·lació.

Un cop acabada l'aplicació de l'instal·lador es va començar el desenvolupament del demostrador.
Aquesta aplicació vol demostrar el correcte funcionament del clúster instal·lat al mateix temps que realitzem la primera fase del procés del Big Data, la obtenció d'informació. Aquesta és una aplicació Java que permet la lectura de dades de Twitter sense filtrar a partir del servei de Flume i les emmagatzema en HDFS del nostre sistema. Apache Flume és un servei que recol·lecta, agrega i mou grans quantitats de dades de diferents fonts a un sistema d'arxius centralitzat, en aquest cas HDFS, el sistema d'arxius de Hadoop. Aquest demostrador utilitzarà una arquitectura prèviament creada per l'instal·lador per veure que s'ha instal·lat tot correctament i que el sistema és funcional. Aquesta estarà formada per un node amb el rol 'Agent' de Flume i un altre amb el rol 'DataNode' de HDFS.
Un Agent de Flume recol·lecta dades de diferents fonts externes, en el nostre cas Twitter. Les emmagatzema en un Canal Flume, que manté la informació fins que la 'Sink' l'emmagatzema a HDFS. Podem veure un diagrama d'aquest funcionament en la Figura 12:

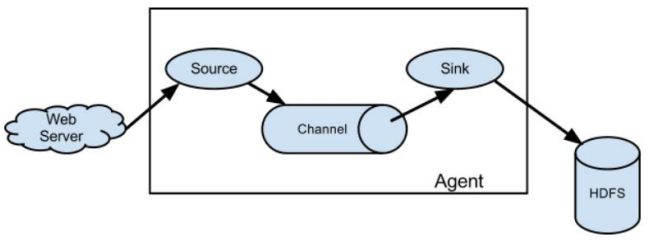

Figura 12: Funcionament d'un Agent de Flume [5]

HDFS està compost d'un únic NameNode i múltiples DataNodes. Un NameNode s'encarrega de gestionar els DataNodes i informar al client que fa la petició d'emmagatzematge els DataNodes disponibles depenent d'un factor de replicació. Aquest client enviarà les dades a guardar al primer DataNode que a la vegada que les emmagatzema les enviarà al segon. Aquest les enviarà al tercer, el tercer al quart, etc. Finalment tindran totes les dades replicades en diferents DataNodes per tenir una alta disponibilitat.
Per poder veure de manera senzilla com funciona HDFS consultar la font 9 de la bibliografia.
Així doncs l'aplicació demostrador modificarà la configuració del nostre node Flume per poder recol·lectar dades de Twitter. Un cop executada podem entrar al contenidor corresponent a HDFS i veure que s'estan inserint les dades en temps real. A la Figura 16 de l'annex podem veure un fragment de les dades de Twitter recol·lectades.

8 Resultats i treball futur

En aquest apartat s'expliquen els resultats obtinguts en la fase final del projecte.

En l'estat final del projecte es pot realitzar tota la instal·lació d'un clúster real d'una arquitectura Big Data mitjançant l'instal·lador i és completament funcional.

Durant el desenvolupament de l'instal·lador han aparegut unes certes restriccions per a un possible client, a l'hora d'utilitzar aquesta aplicació. Aquestes restriccions son:

- Tenir disponible una màquina Linux amb Docker instal·lat (pot ser una màquina virtual).
- Aquesta màquina ha de tenir accés a la xarxa per poder comunicar-se amb el node principal on s'executarà l'instal·lador. En el cas de ser una màquina virtual s'ha de configurar amb 'Bridged Network'.
- Tenir el protocol SSH instal·lat en aquesta màquina.
- Disposar d'un usuari amb permisos Root.
- Aquest usuari ha de poder executar comandes Root sense necessitat d'autentificació amb contrasenya.

Si es satisfan aquests requisits no hi haurà problema per utilitzar l'instal·lador.
Durant el desenvolupament de l'instal·lador es van trobar diferents problemes:

- Falta de documentació API de Cloudera Manager per Java: Hi ha una mancança important d'informació respecte aquesta API. Degut a això es va allargar mes del compte el temps de desenvolupament de la tasca 'Desenvolupar codi utilitzant la API de Cloudera per gestionar els nodes del clúster' respecte la planificació.
- Problema de dependències internes de les APIs utilitzades: Les dues dependències principals del projecte, API de Docker i API de Cloudera Manager, tenen incompatibilitats amb una de les seves dependències internes. Després d'intentar-ho solucionar de manera externa, es va arribar a la conclusió de que s'havia de dividir l'aplicació Java de l'instal·lador en dos. Una aplicació per realitzar tota la gestió de contenidors Docker i una altre per gestionar el clúster. Aquestes dues aplicacions es criden des d'un mateix script, per lo que l'única diferencia visible per a qui utilitzi l'instal·lador serà la utilització d'aquest script.
- Problema amb la interfície degut a la solució del problema de les dependències: Un cop dividida l'aplicació de l'instal·lador en dos, es va tenir que replantejar la manera de desenvolupar la interfície. El problema principal es que els inputs obtinguts d'aquesta son necessaris en totes les parts de l'instal·lador. Per solucionar-ho s'ha implementat de manera que la interfície s'executi a l'aplicació de gestió de contenidors Docker. Aquí es guarden temporalment els inputs obtinguts en un fitxer json que es llegit per l'aplicació de gestió del clúster.

Els resultats en referència al rendiment d'aquest projecte poden variar depenent de les especificacions de les màquines del client. Per exemple, l'instal·lador es va començar desenvolupant en un portàtil amb 8 Gb de RAM amb una màquina virtual amb 6 Gb assignats. Això feia que aquest anés extremadament lent al iniciar la màquina virtual i tots els contenidors necessaris. Un cop vist aquest fet, es va traslladar el desenvolupament a un portàtil amb 16 Gb de RAM amb una màquina virtual de 8 Gb on es possible realitzar el desenvolupament en condicions.

També podem dir que realitza la instal·lació bastant més ràpida si ho comparem amb una persona que ha d'investigar el com fer-ho i ha de realitzar tots els passos previs manualment. Aquest passos poden incloure validacions tècniques de les especificacions de les diferents màquines, instal·lació dels paquets necessaris per incloure la distribució de Cloudera, comprovació de connectivitat entre nodes, etc. Aquesta persona la completaria com a mínim en un rang de 8 hores, en canvi l'instal·lador la completa aproximadament en 15 minuts.
Podem confirmar que el resultat de l'instal·lador és l'esperat gràcies a que s'ha pogut realitzar una aplicació per demostrar el correcte funcionament del clúster.
Les tasques referents a l'Objectiu 4 finalment no es podran dur a terme en aquest període de temps. Això és degut a que les parts més crucials del projecte han ocupat la major part del temps i s'ha preferit donar-los-hi prioritat abans que al Objectiu 4, ja que requereix que totes les parts anteriors funcionin correctament i estava planificat com un objectiu opcional.
Serà una tasca que s'acabarà durant la meva estada a Everis en el futur.
Per poder millorar l'aplicació s'han plantejat diferents tasques futures:

- Major grau de personalització de serveis i rols dins de l'aplicació, ja que ara és bastant limitat i només es poden triar 5 dels serveis que ofereix Cloudera.
- Millora de la interfície estèticament i ampliació de les seves funcionalitats, ja que actualment la instal·lació no permet tirar enrere ja que s'han de desfer tots els passos realitzats fins a aquell punt, incloent creacions de contenidors i assignació de rols.
- Realitzar un control d'inputs d'usuari.

9 CONCLUSIONS

En aquesta part s'exposen unes conclusions respecte al treball realitzat.

En l'estat actual del projecte es pot dir que compleix amb un 100% de l'Objectiu 1, 2 i 3: S'ha aconseguit crear i implementar un sistema Big Data mitjançant la instal·lació de diferents recursos de manera àgil en una màquina virtual. Aquest fet ajuda a reduir el cost d'una implementació real en relació a la infraestructura i el temps. Això és degut a que no s'han d'adquirir les màquines físiques per crear el clúster, ja que es pot fer en una màquina virtual. Tot i així, depenent de la infraestructura del client, es podria realitzar la instal·lació en una màquina física. També ajuda a reduir el temps d'instal·lació i el cost en la manutenció d'experts en el sector. Això es així ja que ho permet fer de manera automàtica sense tenir a una persona especialitzada dedicada exclusivament a la instal·lació, i aquesta no s'ha de preguntar el 'com' fer-ho.

Com s'ha comentat anteriorment el rendiment esta relacionat directament amb les especificacions de la màquina del client. Es pot concloure que aquest fet és un problema ja que no es pot controlar la infraestructura del client i s'haurà d'informar anticipadament d'aquesta limitació.
També s'ha explicat anteriorment que el rendiment al executar una aplicació tampoc serà òptim, perquè no es disposa d'un entorn distribuït real. Això fa que no es tinguin tots els recursos dels que disposaria un clúster real ja que es realitza la instal·lació en una sola màquina i estem limitats per les seves especificacions. Per solucionar-ho es pretén ampliar el projecte per la seva execució en entorns Cloud, però tal i com s'ha explicat anteriorment es realitzarà en un futur dins de l'entorn empresarial.
Sabent això es pot assegurar que actualment l'ús d'aquest instal·lador és crear un entorn on realitzar proves, on no es busca tenir un màxim rendiment sinó on provar les funcionalitats desitjades de diferents aplicacions distribuïdes.
També hem pogut demostrar el funcionament correcte de un clúster creat per l'instal·lador que ens ha servit per aprendre sobre les tecnologies Flume i HDFS i sobre una de les primeres fases del Big Data, la obtenció de dades.

Agraïments

Agrair en primer lloc al meu tutor acadèmic Daniel Ponsa pel suport donat durant el desenvolupament del projecte. També voldria agrair a Ferran Fernandez, tutor de Everis, per tota la seva ajuda i expertesa en el tema i per estar sempre estar disponible i disposat a ajudar-me. Finalment voldria agrair a la meva família i als meus amics per donar-me el suport necessari per ajudar-me a tirar endavant el projecte en moments de molta càrrega de treball.

Definicions

Nodes: Maquines físiques, virtuals o contenidors Docker que formen part d'un clúster.

Clúster: Agrupació de nodes que actuen com un sol sistema per obtenir un gran paral·lelisme.

Parcel·les: Distribucions formades per els fitxers i totes les dades addicionals que utilitza Cloudera Manager per instal·lar els serveis de CDH als nodes. [3]

Imatge Docker: És un executable que inclou tot lo necessari per executar una aplicació. [2] Una imatge Docker es pot instal·lar a qualsevol lloc amb Docker instal·lat, ja sigui un servidor virtual, un portàtil, etc. [6]

Contenidor Docker: unitat bàsica de Docker. És una instància o execució d'una imatge.

Bibliografia

[1] Introducció Cloudera Manager, https://www.cloudera.com/documentation/enterprise/5-4-x/topics/cm_intro_primer.html, Data darrer accés: 22/05/18
[2] Introducció Docker, https://docs.docker.com/get-started/#containers-and-virtual-machines, Data darrer accés: 22/05/18
[3] Definició parcel·les, https://www.cloudera.com/documentation/enterprise/5-6-x/topics/cm_ig_parcels.html, Data darrer accés: 22/05/18
[4] Serveis de CDH, https://www.cloudera.com/products/open-source/apache-hadoop/key-cdh-components.html, Data darrer accés: 22/05/18
[5] Introducció Flume, http://flume.apache.org/FlumeUserGuide.html, Data darrer accés: 22/05/18
[6] J. Turnbull, The Docker Book. V1.0.7, https://www.dockerbook.com/, Data darrer accés: 22/05/18
[7] A McAfee, E Brynjolfsson. Big Data: The Management Revolution. Harvard Business Review, 2012.
[8] Andrea De Mauro, Marco Greco, Michele Grimaldi, (2016) "A formal definition of Big Data based on its essential features", Library Review, Vol. 65 Issue: 3, pp.122-135, https://doi.org/10.1108/LR-06-2015-0061, Data darrer accés: 22/05/18
[9] Còmic sobre el funcionament de HDFS: https://docs.google.com/file/d/0B-zw6KHOtbT4MmRkZWJjYzEtYjI3Ni00NTFjLWE0OGItYTU5OGMxYjc0N2M1/edit?pli=1, Data darrer accés: 02/06/18

APÈNDIX

A1. DISSENY INICIAL INTERFÍCIE

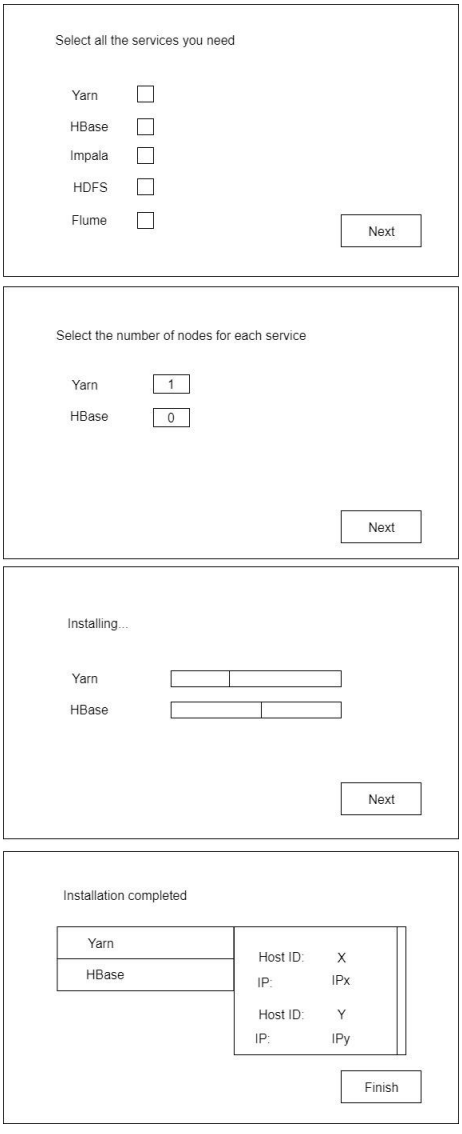

Figura 13: Primer disseny de la interfície

A2. DIAGRAMA DE GANTT

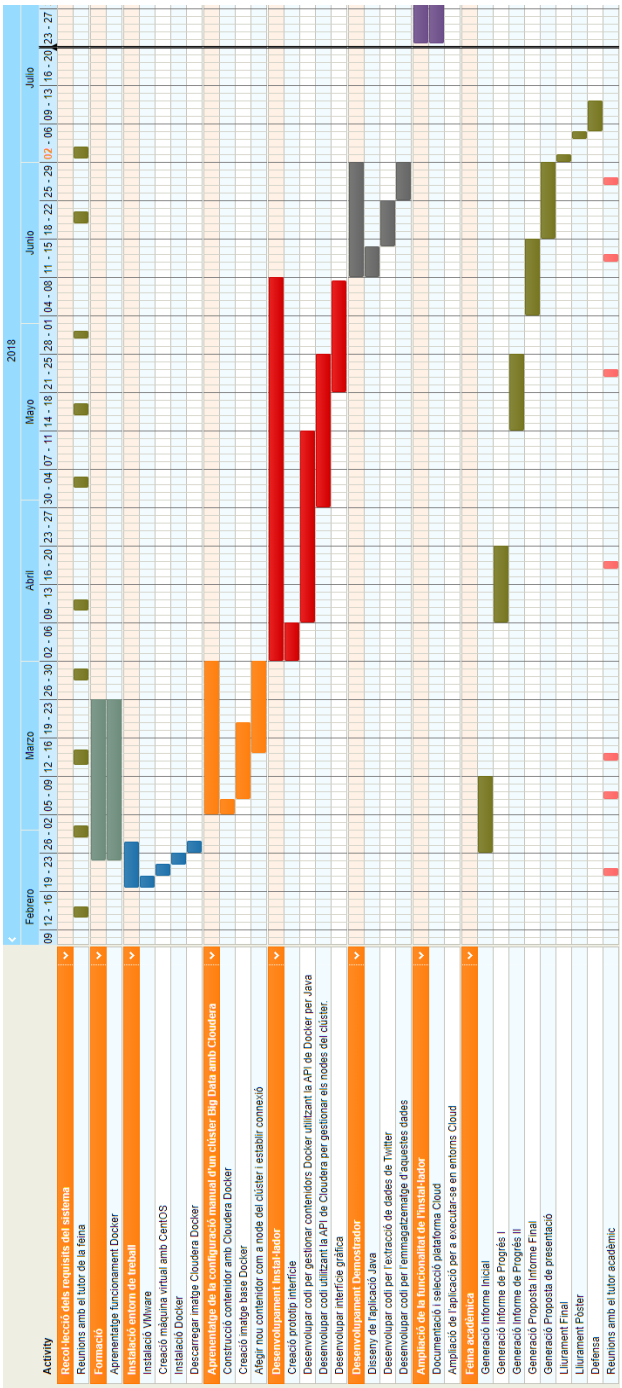

Figura 14: Diagrama de Gantt

A3. ARBRE D'OBJECTIUS

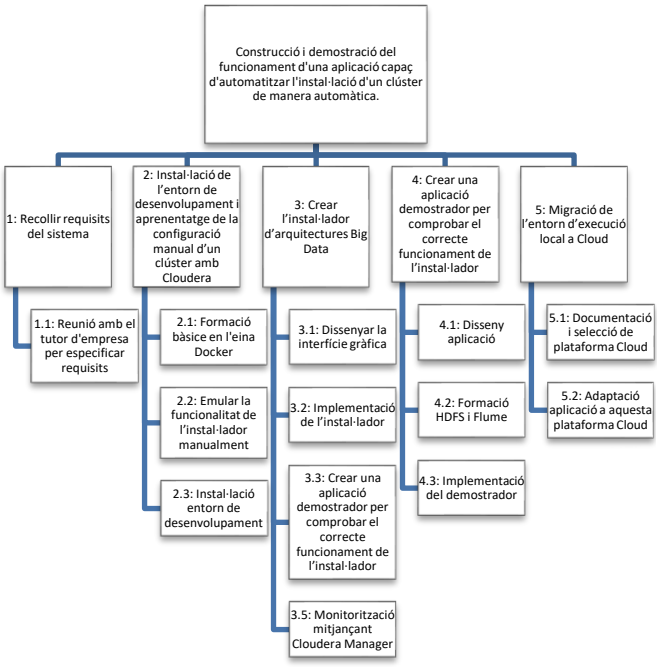

Figura 15: Arbre d'objectius del projecte

A4. DADES DE TWITTER

Figura 16: Dades de Twitter carregades a HDFS