



UNIVERSIDAD DE CASTILLA-LA MANCHA

**Escuela Superior de Informática**

**Ingeniería en Informática**

**Anteproyecto**

**Geo-Cloud**

Modelling and implementation of the Geo-Cloud experiment for  
the Fed4FIRE European Project

**Autor:** Rubén Pérez Pascual

**Director:** Jonathan Becedas Rodríguez

**Tutor:** Carlos González Morcillo

24 de febrero de 2014

Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Problemática . . . . .	3
<b>2. Objetivos</b>	<b>4</b>
<b>3. Métodos y fases de trabajo</b>	<b>6</b>
<b>4. Medios que se pretenden utilizar</b>	<b>7</b>
4.1. Recursos Hardware . . . . .	7
4.2. Recursos Software . . . . .	8
4.3. Documentación . . . . .	9

## 1. Introducción

Las compañías dedicadas a observación de la Tierra, conocidas como *EO industries* en la jerga internacional, han incrementado la venta de imágenes en un 550% en la última década. Se estima que este aumento sea de cuatro veces superior en lo que resta de década. Por esta motivo, observación de la Tierra está reconocido como uno de los campos clave en el ámbito europeo. Para apoyar dicho ámbito, la Comisión Europea junto con la Agencia Espacial Europea y la Agencia Europea del Medioambiente (EC, ESA y EEA por sus siglas en inglés respectivamente) han creado el sistema GMES/Copernicus, de modo que aumente la capacidad europea en temas relacionados con la observación de la Tierra.

La observación de la Tierra consiste en el registro de datos espacio temporales del mundo, y se aplican a numerosos sectores: monitorización del medioambiente, desastres naturales y seguridad ciudadana entre otros. Los sensores que aportan mayor información al respecto son los satélites de observación de la Tierra.

Sin embargo, dichos sistemas proporcionan la información a unos precios elevados, lo que limita el acceso a la misma de un mayor número de usuarios de otros ámbitos y del público en general. Esto ocurre, principalmente, por dos motivos:

1. La tecnología empleada es cara.
2. Los caminos que sigue la información desde que se obtiene con los satélites hasta que llega a los usuarios requieren de unas infraestructuras específicas, que conllevan un mayor coste que el propio de la instrumentación empleada, es decir, el satélite.

Sin embargo, en los últimos años la tecnología de internet del futuro ha ido desarrollando nuevas características, sobre todo con la aparición de sistemas distribuidos o también denominados *cloud* o nube. La nube puede contribuir a agilizar tanto la infraestructura, anteriormente descrita, como el servicio que se proporciona al cliente final, debido a sus características:

- Es elástica
- Es escalable

- Es flexible
- Puede trabajar bajo demanda

En el proyecto GEO-Cloud se estudia la posibilidad de transferir los centros de datos de Elecnor Deimos a la nube, con el objetivo de procesar y distribuir las imágenes capturadas con satélites de observación de la Tierra en nube, y comprobar si realmente esta tecnología puede emplearse en los tradicionales sistemas de observación de la Tierra.

GEO-Cloud nace, por tanto, en el seno del proyecto europeo integrador Fed4FIRE (IP FP7 project nº 318389).

Fed4FIRE[5] es un proyecto de integración FP7 cuya causa es la búsqueda y la experimentación con la internet del futuro. El proyecto está conformado por 17 organizaciones de 8 países distintos. Principalmente, sus objetivos se basan en construir diversas herramientas<sup>1</sup> para que los investigadores puedan realizar diversos experimentos, entre las que nosotros utilizaremos las siguientes:

- Virtual Wall
- PlanetLab Europe
- BonFIRE

El proyecto GEO-Cloud consiste en la construcción de un experimento con las distintas secciones:

- Diseñar y construir un orquestador en la nube BonFIRE que permita el procesamiento de las imágenes obtenidas por los satélites y controle todo este proceso. Estos procesadores son proporcionados por la empresa Elecnor Deimos y se tratarán como caja negra.
- Realizar un sistema de archivo y catálogo para conseguir que el resultado obtenido del procesamiento anterior quede almacenado en la nube y accesible por los clientes finales.

---

<sup>1</sup>Para más información ir a <http://www.fed4fire.eu/>

- Diseñar y construir un pequeño experimento sobre PlanetLab Europe, para obtener las medidas reales de los parámetros y métricas involucradas en las comunicaciones de todos los componentes como clientes, estaciones de tierra, nube y constelación de satélites.
- Diseñar y construir el software que simulará sobre VirtualWall, los distintos satélites, estaciones de tierra y clientes accediendo a los recursos mediante un servicio web. Para este proceso será necesario conocer los distintos parámetros y métricas de comunicación reales (obtenidos en el paso anterior) para construir un experimento realista.
- Construir un experimento aunando lo desarrollado en los tres testbeds antes mencionados.
- Finalmente, para validar el sistema se simulará un caso de uso.

### 1.1. Problemática

En las empresas actuales de observación de la Tierra, tanto por seguridad como por coste de inversión, las instalaciones donde se procesan y distribuyen los datos están en localizaciones propiedad de las empresas explotadoras de los datos, tanto por seguridad como por coste de inversión, lo que conlleva las siguientes limitaciones:

- Dichas infraestructuras deben sobredimensionarse para almacenar los datos procedentes de los satélites.
- La información debe ser procesada en la infraestructura y después distribuida con métodos tradicionales a los usuarios finales.
- La arquitectura del sistema impide que el servicio ofrecido sea flexible.
- Estos sistemas no son escalables.

Por contrapartida, en los últimos años la tecnología de internet del futuro ha ido desarrollando nuevas características, sobre todo con la aparición de sistemas distribuidos o también denominados *cloud* o nube. La nube puede contribuir a agilizar tanto la infraestructura,

anteriormente descrita, como el servicio que se proporciona al cliente final, debido a sus características:

- Es elástica
- Es escalable
- Es flexible
- Puede trabajar bajo demanda

Geo-cloud pretende enfrentar dos implementaciones. La primera y ya asentada configuración propia de las compañías procesando sus propias imágenes y almacenándolas, con todo lo que conlleva. La segunda, el proyecto GEO-Cloud que estudia la posibilidad de transferir los centros de datos de Elecnor Deimos a la nube, con el objetivo de procesar y distribuir las imágenes capturadas con satélites de observación de la Tierra en nube, y comprobar si realmente esta tecnología puede emplearse en los tradicionales sistemas de observación de la Tierra.

## 2. Objetivos

Según lo expuesto en la sección 1, sería deseable disponer de un sistema que simule una constelación de satélites y varias estaciones de tierra con sus respectivas métricas en las comunicaciones. Además, que el procesamiento de los datos en bruto obtenidos en las estaciones de tierra se realice en la nube y la adquisición de esas imágenes procesadas por parte de los clientes sea real.

Por ello en este proyecto se plantean los siguientes objetivos:

- **Implementación de los servicios basados en Cloud en la plataforma Bon-FIRE:** Se pretende desarrollar un servicio de procesamiento de datos Raw y creación de orto-imágenes en Cloud. Para esta implementación será necesario conocer la arquitectura de la cadena de procesamiento que se implementa en Elecnor Deimos. Además se

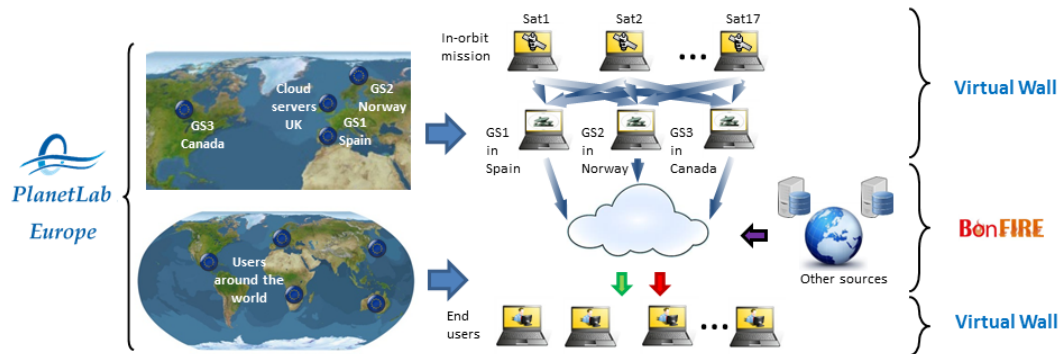


Figura 1: Geo-Cloud en Fed4FIRE

debe de dar soporte a las peticiones de los clientes con lo cuál se deben de implementar servicios de distribución de estos resultados entre los usuarios finales.

- **Implementación de la constelación de satélites, estaciones de tierra y clientes sobre PlanetLab Europe:** Otro aspecto a tener en cuenta serían las métricas. Para esto se usará PlanetLab Europe, con la que conseguiremos simular conexiones de red entre distintos nodos distribuidos por el mundo. A través de la implementación del modelo de red en PlanetLab Europe se podrán medir las características de una red real. Se realizarán dos capas en esta implementación. La primera modelará las estaciones de tierra y la constelación; la segunda, los clientes finales accediendo a los servicios Cloud.
- **Implementación de la constelación de satélites, estaciones de tierra y clientes sobre VirtualWall:** Tras realizar la implementación sobre PlanetLab Europe, se conseguirá toda la información de una red real, con lo que se realizará un modelo sobre VirtualWall de acuerdo a esos parámetros. Además esta plataforma permitirá tener completo control sobre los recursos y todos los factores envueltos en los distintos casos de prueba, con lo que se podrá evaluar su comportamiento.
- **Simulación de distintos escenarios y obtención de resultados:** Tras realizar la implementación de lo anteriormente expuesto, se procederá a realizar uno o varios casos de prueba con los que obtengamos datos para contrastar como actúa este servicio de procesamiento de obtención de geo-imágenes en la nube frente a la misma implementación sobre un centro de procesamiento de datos.

### 3. Métodos y fases de trabajo

Para llevar a cabo los objetivos propuestos en este documento se dividirá el proyecto en las siguientes fases:

1. **Estudio del estado del arte:** En esta fase será necesario conocer los distintos testbeds proporcionados por el consorcio *Fed4FIRE* y comprender bien los aspectos relativos a su funcionamiento. Se realizará un estudio para obtener las métricas a tener en cuenta a la hora de la implementación de la simulación. Se realizará un estudio en profundidad sobre las posibilidades que ofrece BonFIRE para la creación y control de los experimentos. Se estudiará la plataforma Virtual Wall para la simulación de nodos. Se estudiará la plataforma PlanetLab Europe para la obtención de las métricas reales y poder ser cercano a la realidad. Se estudiará el centro de proceso de datos de la empresa Elecnor Deimos y más concretamente el núcleo de procesamiento de los datos raw para poder implementarlo en la nube que nos proporciona BonFIRE.
2. **Estudio de la viabilidad:** El proyecto se adaptará realizando pruebas para verificar el correcto funcionamiento del mismo y su puesta en marcha. Se dispondrán de los distintos testbeds y el computador de desarrollo donde poder probar la plataforma. Para el uso óptimo de la aplicación es muy recomendable determinar qué requisitos, tanto hardware como software, serán necesarios.
3. **Análisis de la arquitectura del sistema:** Partiendo de los requisitos más importantes que debe cumplir este proyecto, se realizará un análisis más concreto de los componentes que formarán la arquitectura teniendo en cuenta los principios de modularización y extensibilidad.
4. **Diseño de la arquitectura del sistema:** En esta fase se intentará concretar, a un nivel más cercano a la implementación, cada una de las clases que conformarán la arquitectura del sistema. Se podrá utilizar un lenguaje específico para este propósito como *UML*. Esta fase se realizará de forma concurrente junto a las fases de desarrollo y pruebas, realimentándose para mejorar el diseño de la arquitectura.
5. **Desarrollo del sistema:** Se dará comienzo a la implementación de todos los módulos



que formarán el sistema, en el lenguaje y formato escogido, siguiendo la arquitectura del sistema. Se usará una metodología iterativa incremental para el desarrollo del mismo.

6. **Pruebas del sistema:** Se tratará de ir realizando pruebas tanto de caja negra como de caja blanca de cada una de las clases implementadas. Cuando se termine el sistema, se harán pruebas de integración de todos los componentes y su funcionamiento.

7. **Evaluación de los resultados:**

Se evaluará la ganancia en eficiencia respecto a un sistema de procesamiento de imágenes raw en un CPD el cual no se beneficie de los beneficios de la ejecución y distribución en la nube.

8. **Documentación del proyecto:** De forma paralela al desarrollo del proyecto se irá generando la documentación del mismo (diagramas de clases, de análisis, requisitos, manual de usuario, etc.) para completarla al final con las conclusiones y las pruebas finales que se realicen.

## 4. Medios que se pretenden utilizar

El propósito del proyecto es realizar una simulación de un sistema distribuido robusto, modular y altamente escalable. La distribución del proyecto se realizará mediante alguna licencia libre como GPLv3[7].

Se seguirá el *proceso unificado de desarrollo de software* durante la realización del proyecto. Para la notación se usará *UML*[10].

### 4.1. Recursos Hardware

Se dispondrá de varios *Testbed* proporcionados por el proyecto europeo *Fed4FIRE* los cuales son:

1. *PlanetLab Europe*[6]: es la rama europea del sistema PlanetLab, la red más grande del mundo para realizar experimentos. Tiene servidores distribuidos a lo largo de todo el

mundo y se pueden evaluar distintas métricas de red como latencia, ruido, pérdida de paquetes, etc. Esta facility se usará para obtener las medidas reales de los parámetros que se utilizarán en los nodos desarrollados en Virtual Wall y BonFIRE.

2. *Virtual Wall*[11]: Consiste en numerosos nodos físicos (concretamente 100) mediante los cuales se pueden realizar simulaciones. Es totalmente configurable, permitiendo crear la red deseada para el experimento y definir los parámetros de la red.
3. *BonFIRE*[4]: Es una plataforma Cloud orientada a la experimentación. Contiene 7 test-beds en Europa geográficamente distribuidos que ofrecen servicios de almacenamiento, procesamiento y networking de manera heterogénea.

Además se dispondrá de un computador localizado en *Elecnor Deimos* en Puertollano para el desarrollo y la ejecución de los casos de prueba. El hardware de este constará de un procesador Intel Core i5 a 3,2 MHz, 8 Gb de memoria RAM y 1 Tb de disco duro.

## 4.2. Recursos Software

Los sistemas operativos, entornos de programación y herramientas que se utilizarán en el desarrollo del proyecto serán Software Libre.

- *Sistema Operativo CentOS*<sup>2</sup>.
- *GNU Make* [14]: herramienta para la generación automática de ejecutables, que se empleará para automatizar cualquier proceso susceptible de ello.
- *GNU Emacs* [13]: editor de textos extensible y configurable, para el desarrollo e implementación del proyecto.
- *Git* [2]: ofrece un sistema de control de versiones distribuido y la posibilidad de llevar ese control fuera de línea, frente a los sistemas de control de versiones centralizados como CVS o Subversion.

---

<sup>2</sup><http://www.centos.org/>

- *Python* [8]: lenguaje interpretado multiparadigma y multiplataforma que se usará para programar gran parte de este proyecto. Se usará la versión 2.7.4 para el desarrollo de este proyecto.<sup>3</sup>
- *Ruby* [3]: lenguaje interpretado, de propósito general y orientado a objetos que se usará para la construcción de la plataforma de procesamiento de datos raw en la nube. Se usará la versión 1.9 para el desarrollo de este proyecto.
- *Nepi*[1]: Es una plataforma basada en Python para diseñar y realizar experimentos en plataformas de evaluación de redes como por ejemplo PlanetLab. Se usará para realizar la simulación sobre PlanetLab y obtener los parámetros reales para realizar la simulación.
- *GeoServer* [9]: Es una plataforma open source escrito en Java que permite a los usuarios compartir y editar datos geospaciales. Este servicio se utilizará como archivo y catálogo para almacenar los resultados del procesamiento anterior.

### 4.3. Documentación

- $\text{\LaTeX}$  [12]: lenguaje de marcado de documentos, utilizado para realizar este documento y, en un futuro, la documentación del proyecto.
- $\text{\BibTeX}$ : herramienta distribuida con  $\text{\LaTeX}$  para generar las listas de referencias tanto de este documento como de la memoria del proyecto.

---

<sup>3</sup>Página web oficial de Python

## Referencias

- [1] Quereilhac A., Tribino J., Guevgeozian L., and Kouvakas A. Nepi v3.0. 2013.
- [2] S. Chacon. *Pro Git*. Apress, 2009.
- [3] Ruby Software Community. Ruby Documentation. <https://www.ruby-lang.org/en/documentation/>, última consulta el 16 de Febrero de 2014.
- [4] BonFIRE Consortium and University of Southampton. BonFIRE Documentation. <http://doc.bonfire-project.eu/R4.0.5/>, última consulta el 24 de Febrero de 2014.
- [5] Fed4FIRE Consortium. Fed4FIRE Documentation. <http://doc.fed4fire.eu/index.html>, última consulta el 24 de Febrero de 2014.
- [6] PlanetLab Europe. PlanetLab Documentation. <http://doc.planet-lab.eu/html/>, última consulta el 24 de Febrero de 2014.
- [7] Free Software Foundation. Licencia GPL versión 3 de la Free Software Foundation. <http://www.fsf.org/licensing/licenses/gpl.html>, última consulta el 13 de Enero de 2014.
- [8] Python Software Foundation. Python v3.4 documentation. última consulta el 18 de Febrero de 2014.
- [9] GeoServer. Geoserveruser manual. <http://docs.geoserver.org/stable/en/user/>, última consulta el 16 de Febrero de 2014.
- [10] Object Management Group. Omg unified modeling language, infrastructure. Abril, 5 , 2010.
- [11] iMinds. VirtualWall Documentation. <http://www.iminds.be/en/develop-test>, última consulta el 24 de Febrero de 2014.
- [12] L. Lamport. *LaTEX: User's Guide & Reference Manual*. Addison-Wesley, 1986.
- [13] R. Stallman, A. Universitet, and D. Afdeling. *GNU Emacs manual*. Free Software Foundation, 1993.

- [14] R.M. Stallman and R. McGrath. *GNU make*, volume 2002. Free Software Foundation, 1988.