Simanfor Basics Extended

Moisés Martínez

5 de noviembre de 2019

1. Introducción

Este documento describe el proceso de desarrollo de la nueva versión del simulador "Simanfor". Esta nueva versión será desarrollada con el fin de desplegar el simulador en diferentes entornos de despliegue formados por nodos. Un nodo es un unidad básica de ejecución que estará formada por un procesador (n número de cores) y un conjunto de memria ram no distribuida. En base a esta definición se pueden definir tres posibles entornos:

- Entorno base: El simulador podrán desplegarse en un entorno local constituido por un único nodo. Este entorno permitirá la ejecución de las simulaciones consumiendo un mayor tiempo de ejecución.
- Entorno cloud: El simulador podrá desplegarse en diferentes nodos de manera que podrán acelerarse el tiempo de ejecución. Los diferentes nodos podrían tener diferentes características.
- Entorno Supercomputación: El simulador podrá desplegarse en entorno de supercomputación. Al igual que en el caso anterior, los diferentes nodos que forman el entorno podrán tener diferentes características, pero deberán compartir los diferentes sistemas de comunicación: (1) MPI; o (2) OpenMP.

Además se utilizará una aplicación web que permitirá simplificar el proceso de interacción con el simulador.

2. Requisitos del sistema

Para el desarrollo del simulador se han definido una serie de requisitos básicos que deben cumplirse para el correcto desarrollo del simulador:

- Debe mantenerse el sistema de almacenamiento actual en Microsoft SQL.
- Se deben utilizar los datos existentes actualmente en el sistema de almacenamiento actual.

- El sistema debe funcionar de forma similar en diferentes entornos de trabajo de manera transparente (un único pc, múltiples pcs y un supercomputador) para el usuario del simulador.
- La configuración del experimento a simular se debe realizar mediante la utilización de un fichero de configuración, donde se deberá especificar el entorno y los diferentes elementos básicos que define la simulación.
- El sistema deberá utilizar la actual base de datos disponible.
- El sistema debe ofrecer un interfaz de interacción similar al actual. Es decir, se deberá conservar la actual pagina web o deberá desarrollarse una nueva versión que tenga la misma funcionalidad.
- El sistema deberá generar un fichero de tipo XSL (formato excel) que debe cumplir la estructura del fichero de salida proporcionada por la universidad.
- El sistema deberá tomar como entrar un fichero de tipo XSL (formato excel) que debe cumplir la estructura del fichero de entrada proporcionada por la universidad.
- El proceso de simulación deberá poder ejecutarse sobre diferentes sistemas operativos, es decir deberá poder ejecutarte al menos sobre sistema operativos de tipo Linux y Windows. Aunque se recomienda su utilización sobre entornos Linux, ya que algunas de las librerias que serán utilizadas puede que no funcionen de manera correcta en los sistemas operativos de Microsoft Windows.

3. Tecnologías

Para el desarrollo del simulador se han seleccionado las siguientes tecnologías:

- Python: Es el lenguaje de programación que será utilizado para el desarrollo del simulador. Este lenguaje ha sido seleccionado debido a su versatilidad a la hora de manipular datos y a las diferentes herramientas que serán utilizadas para el desarrollo del simulador. Se utilizará la versión 3.6 debido a que la versión 2.7, que es la más común, dejará de tener mantenimiento a partir de Enero de 2017.
- Microsoft SQL: Es el tipo de base de datos que será utilizada para el almacenamiento de la información. Se ha decidido utilizar este tipo de base de datos debido a que la información que utiliza el actual simulador se almacena en este tipo de base de datos, por lo que no se modificará ni la estructura de los datos ni su modelo de almacenamiento. Se recomienda la actualización de la actual versión del servidor Microsoft SQL.

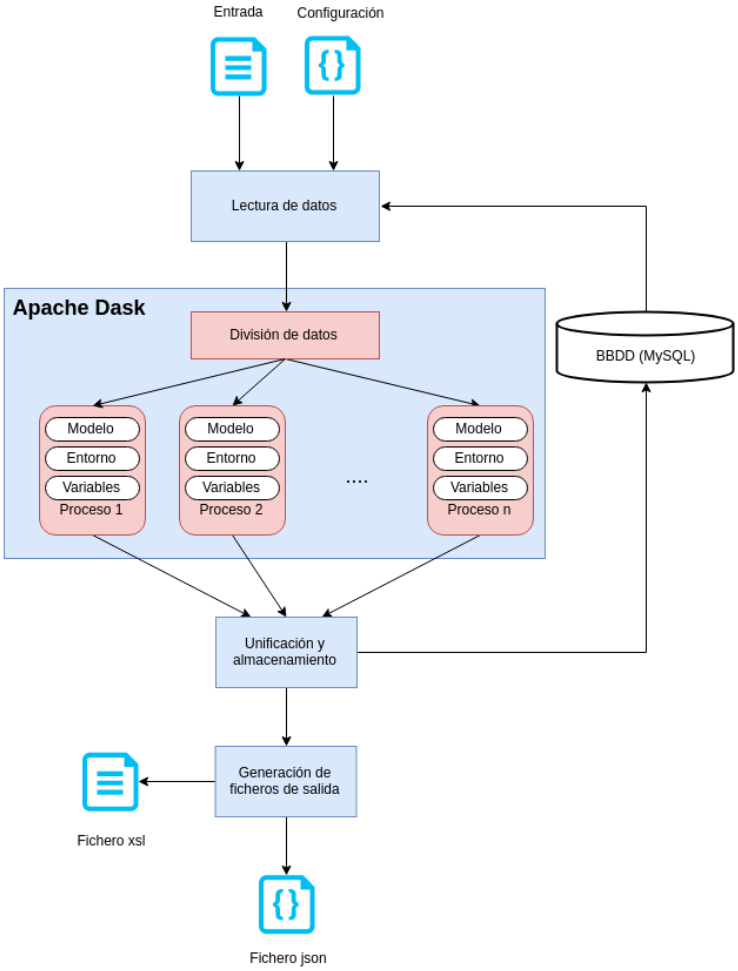
- Apache Dask: Es una herramienta para el procesamiento de grandes volúmenes de datos. Apache Dask permite la utilización de estructuras de datos basadas en dos de los formatos de manipulación de datos más extendidos: (1) Pandas; y (2) matrices de Numpy. Estos conjuntos de datos puedes ser ejecutados en una única máquina o un en un cluster de máquinas de forma transparente para el desarrollador disminuyendo el tiempo de ejecución y resolviendo los posibles problemas de carga en memoria RAM que supone el procesamiento de grandes volúmenes de datos.
- Dask MPI: Es una extensión desarrollada sobre Apache Dask que permite la ejecución de procesos distribuidos basados en Dask en entorno MPI. Esta extensión permite la ejecución de entornos que normalmente se encuentran desplegados en supercomputadoras de alto rendimiento, instituciones de investigación académica y otros grupos donde MPI ya ha sido instalado. Dask-MPI proporciona una interfaz conveniente para iniciar su clúster desde un script por lotes o directamente desde la línea de comandos.

4. Diseño técnico

En base a la información que hemos obtenido por parte de la Universidad de León, la estructura del nuevo simulador estaría formado por cuatro componentes que se pueden observar en la figura 4:

- Sistema de lectura de ficheros: Este sistema se ocupará de la configuración del entorno utilizando el fichero de configuración de entrada y el fichero de datos de entrada.
  - Fichero de entrada: Este fichero almacenará la información que utilizará el simulador para generar la salida. Su estructura se basa en la proporcionada por la universidad y se compone de un conjunto de filas y columnas (fichero xls) con información sobre las diferentes parcelas de terreno a simular.
  - Fichero de configuración: Este fichero almacena la configuración del simulador. Esta información se divide en cuatro grupos: (1) información de conexión y almacenamiento para la definición de los conectores de bases de datos, fichero de logs y ficheros de resultado; (2) información referente al modelo a utilizar para generar la simulación; (3) información referente al entorno para la generación de la simulación (Esta información dependerá del modelo elegido); y (4) valores de las diferentes variables y constantes que utilizará el modelo. Parte de esta información puede estar almacenada en la base de datos relación.
- Sistema de ejecución: El sistema de ejecución es el conjunto de procesos de ejecución que producirán la simulación que variará entre 1 y N. El número de proceso debe ser siempre inferior al número de nodos disponibles.

Figura 1: Arquitectura básica de funcionamiento del simulador



Para la distribución de la información entre los nodos se utilizará Apache Dask, un sistema ligero de computación paralela para la manipulación de datos almacenados en tablas tabulares. Apache Dask ha sido seleccionado debido a su facilidad de uso y que la información utilizada en el proceso de simulación que puede modelarse como una tabla tabular.

- Sistema de unificación y almacenamiento: El sistema de unificación recogerá la información de cada uno de los nodos de ejecución y se almacenará en la base de datos.
- Sistema de generación de ficheros: El sistema de generación de ficheros almacenará la información en diferentes tipos de ficheros, que podrán ser almacenados en un repositorio con el fin de poder ser descargados a tra-

vés de una aplicación web. El sistema generará dos tipos de ficheros de resultado:

- JSON: Un fichero de tipo JSON para la utilización de la información en otro tipo de aplicaciones de la nube.
- XSL: Un fichero de tipo XLS de formato muy similar al generado actualmente por el simulador.

5. Descripción detallada de componentes

5.1. Sistema de lectura de ficheros

Como se decribío anteriormente el sistema del lectura de ficheros procesa los ficheros de entrada que utilizará el simulador. Este sistema utiliza dos ficheros para iniciar el proceso de simulación. El primero de los ficheros se corresponde la información de entrada que se corresponde con las diferentes parcelas que deben ser simuladas. El segundo fichero define la configuración del entorno que será utilizado para la realización de la simulación en formato json.

- Información básica: Esta información se corresponde con la información de conexión a la base de datos indicándose la dirección ip, puerto, usuario y password. Además de la información de los conectores de almacenamiento, se incluye la localización de los ficheros de logs que generá el simulador y la configuración de los repositorios en los cuales deberán almacenarse los diferentes ficheros de salida.
- Información del modelo: Esta información se corresponde con la localización de los diferentes modelos que podrán ser utilizados. Estos modelos deben ser desarrollados en python (versión 3.6) y almacenados en el código fuente del simulador o en una localización en la nube de manera que el simulador pueda descargarlos al iniciar una simulación. Inicialmente estos modelos se almacenarán en la el código fuente del simulador pero existe la posibilidad de que pudieran ser descargados de la nube en futuros iteraciones del proceso de desarrollo.
- Información del entorno: Esta información se corresponde con la información del entorno que será utilizado para la ejecución de la simulación.
- Definición de variables: Esta información se corresponde con las diferente variables y constantes que utilizará el modelo.

La información del fichero de configuración depende del conocimiento del usuario es decir que el usuario debe conocer como configurar correctamente el fichero de configuración de la simulación. Aunque, debido a que es obligatorio permitir la utilización del interfaz web, se debería poder crear el fichero de configuración mediante un formulario de manera que la ejecución se base en la elección del usuario a través del formulario.

5.2. Sistema de interconexión con la aplicación web

Debido a que existen diferentes modalidades de ejecución es necesario definir como se realizaría la interacción con el interfaz web y las instancias de simulación. Para ello se define dos modalidades de ejecución:

- Modalidad mono: Esta modalidad se corresponde con la ejecución en un entorno donde sólo existe un nodo de computación en donde se encuentra desplegado el simulador y la aplicación web. La modalidad mono sólo permitirá la ejecución del simulador en un entorno compuesto con un sólo nodo de procesamiento, es decir la simulación se ejecutará mediante un proceso secuencial ya que no existe la posibilidad de distribuir el computo entre diferentes nodos.
- Modalidad multi: Esta modalidad se corresponde con la ejecución en un entorno donde existen diferentes nodos de computación, es decir el simulador tiene la posibilidad de distribuir su computo entre diferentes nodos con el fin de disminuir el tiempo de simulación o procesar conjuntos de datos de gran tamaño. La modalidad multi utilizará las capacidades de Apache Dask con el fin de minimizar el tiempo de simulación en entorno distribuidos (cloud) o de supercomputación.

Este sistema supone la modificación de ciertos elementos de la aplicación web, con el fin de poder generar el fichero de configuración del nuevo simulador y definir el modelo de ejecución del simulador el cual dependerá del entorno en el cual se ejecute el simulador. Es necesario analizar como se podrá desplegar este tipo de trabajo tanto en entornos cloud, como en entornos de supercomputación.