Big Data en Live coding

Guillem Borrell

Kernel Analytics

1. Tipo de Contribución

☐ Charla corta
☐ Charla extendida
☐ Póster
☑ Taller

2. Idioma

☑ Español
☐ Inglés

3. Nivel

☑ Avanzado
☐ Medio
☐ Iniciación

Keywords: Data Engineering, Data Sciencie, Big Data ...

4. Resumen

Supongamos que nos encargan un prototipo de sistema capaz de recoger una gran cantidad de datos desde terminales (teléfonos móviles, navegadores, dispositivos IoT...) con el objetivo de volcar toda esta información en un cluster hadoop. ¿Por dónde empezamos? ¿Por descartar Python e irnos a Java y derivados? ¿Pedimos presupuesto para ampliar el clúster Kafka y buscamos programadores en Scala en LinkedIn? El objetivo de esta sesión de live coding es desarrollar en **90 minutos** una aplicación con todos los ingredientes que uno espera de un pipeline de adquisición y proceso de datos; asegurando que el resultado podrá escalar a miles de conexiones concurrentes. Se tratarán temas como el diseño de API eficientes, la programación asíncrona, las colas de mensajes y el análisis de grandes volúmenes de datos. El objetivo secundario es el de introducir algunas herramientas novedosas (y en algunos casos desconocidas) como ASGI, Starlette, uvicorn, NNG, Trio, Skein, Dask... Al finalizar todos los asistentes se conectarán a la aplicación desarrollada para generar tantos datos como sea posible y analizarlos después.

5. Presentación

Python está viviendo dos pequeñas revoluciones contemporáneas. La primera es la llegada de la programación asíncrona como parte del propio lenguaje. La consecuencia directa ha sido un aumento de la capacidad del intérprete ante grandes cargas heterogéneas de trabajo. En la actualidad los servidores web sobre Python tienen poco que envidiar a los implementados en Go, Java o Node, lo que aumenta las posibilidades del uso de Python para el desarrollo de grandes aplicaciones distribuidas. La segunda es la entrada de Python como lenguaje de ingeniería de datos gracias a la aparición de Dask, pyarrow o Skein. Uno puede utilizar Python como lenguaje de análisis de datos dentro de un cluster Hadoop sin necesitar infraestructura específica para almacenar los datos o ejecutar los procesos batch.

La consecuencia es que uno puede construir con Python, poco código y una infraestructura razonable sistemas capaces de ingestar cantidades de información que antes se creían sólo accesibles a plataformas basadas en Java.

Para demostrar todo lo anterior se desarrollará desde cero una herramienta basada en las tecnologías mencionadas, se ejecutará en un servidor en la nube y se comprobará si es capaz de soportar una carga razonable (la de todos los presentes en la sesión que queran conectarse con sus ordenadores y teléfonos). La aplicación en cuestión utilizará la capacidad innata de los humanos para leer

para hacer crowdsourcing y comprobar si un algoritmo de OCR [1] está funcionando adecuadamente. Durante el desarrollo se mostrarán las capacidades que nos aportan estas nuevas herramientas desarrollando una aplicación completa durante la duración del taller, con especial atención a la programación asíncrona aplicada a servidores web, la optimización del flujo de ejecución y la persistencia de datos.

Este taller se puede considerar como una versión actualizada y mejorada de [2]. Todo el código correspondiente a esta sesión se encuentra alojado en [3].

6. Pre-requisitos para atender a la presentación

Se trata de una sesión relativamente avanzada, donde se asume cierto conocimiento de las tecnologías más habituales en ingeniería y análisis de datos y fundamentos de desarrollo web.

El objetivo es desarrollar en live coding una aplicación distribuida completa. El tiempo establecido de 60 minutos para un taller dejaría muy poco margen en el caso de surgir algún inconveniente técnico durante el desarrollo o durante la demo.

Los asistentes deberán contar con un teléfono móvil, tablet u ordenador conectado a Internet durante el taller para participar en la demo final. En el caso que se quiera seguir la sesión de live coding deberán contar con un ordenador o una máquina virtual con Linux, Anaconda y Git instalados.

7. Otros requerimientos técnicos

Para la sesión de live coding necesitaría poder utilizar mi portátil para proyectar y conectarlo a Internet, además de una mesa o un atril donde apoyarlo.

Referencias

1. Wikipedia: Optical character recognition. https://en.wikipedia.org/wiki/Optical_character_r (2019)
2. Borrell, G.: Python for distributed systems. https://youtu.be/cYMfc3vgns8 (2016)
3. Borrell, G.: Github: Taller pycon es 2019. https://github.com/guillemborrell (2019)