

# FABIOLA: Herramienta para la Optimización de Problemas en entornos de Big Data

Luisa Parody, Ángel Jesús Varela-Vaca, Rafael M. Gasca

Departamento de Lenguajes y Sistemas Informáticos,  
Universidad de Sevilla, Spain  
{ajvarela,lparody,gasca}@us.es,  
<http://www.idea.us.es>

**Abstract.** Dentro de las organizaciones, los problemas de optimización pueden encontrarse en numerosos ejemplos, tales como minimizar los costes de producción, los errores producidos, o maximizar la fidelidad de los clientes. La resolución de estos problemas es un reto que conlleva un esfuerzo extra. Hoy en día, los problemas de Big Data se suman a estos problemas de optimización en dichas empresas. Desafortunadamente, afrontar estos problemas en la pequeña y mediana empresa es extremadamente difícil o incluso imposible. En este artículo, proponemos la arquitectura llamada FABIOLA, que permite describir los datos distribuidos y estructurados en problemas de optimización que pueden ser paralelizados. Además, Fabiola aplica las técnicas de Programación con Restricciones para poder devolver la solución a dichos problemas de optimización.

**Key words:** Big Data, Problemas de Optimización, Programación con Restricciones, Estructura de Datos

## 1 Introducción

Hoy en día, se generan grandes volúmenes de datos por los sistemas de información de las organizaciones. El concepto de Big Data viene definido por los datos que exceden las capacidades de los entornos hardware y las herramientas de software comunes necesarias para captar, gestionar y procesar dichos datos dentro de un periodo de tiempo competente [1]. Las nuevas tecnologías software están evolucionando constantemente para facilitar la gestión de Big Data. Uno de los principales desafíos en las soluciones de Big Data es aislar a los usuarios de utilizar datos y localizaciones heterogéneos. Esta es la razón por la que se desarrollan tantísimos componentes en el ecosistema de Hadoop [2]. Sin embargo, los problemas de optimización usando las técnicas de programación con restricciones es un problema aún por resolver.

En este artículo, se presenta la arquitectura llamada Fabiola (del inglés, FAst BIg cOstraint LAB), una solución centrada en problemas de optimización de datos construida sobre Hadoop. FABIOLA soporta la distribución de problemas de optimización con el fin de obtener soluciones óptimas para subconjuntos

de datos independientes y distribuidos. Los problemas de optimización de restricciones se compilan en los motores map-reduce ejecutada sobre Hadoop que además puede combinarse con Hive [3] con el fin de inferir más información después de encontrar los valores óptimos.

## 2 Arquitectura de Fabiola

La Figura 1 muestra los principales componentes de la arquitectura de Fabiola y su integración en la infraestructura de Big Data basada en Hadoop. Fabiola está de los siguientes cuatro componentes:

1. **Interfaz de Usuario (Fabiola UI)** es el lado del cliente (aplicación web) que permite subir y cargar datos de fuentes externas heterogéneas.
2. **Fabiola Metastore** se proporciona como catálogo del sistema donde los datos se organizan en forma de tablas y esquemas, que son la representación/vista virtual de los datos.
3. **Nodos Fabiola** son los nodos resolutores sobre la tecnología Hadoop que permite ejecutar los COPs. Las posibles soluciones de dichos COP completan las salidas de cada tupla en el Metastore.
4. **Fabiola Dashboard** es el componente de reporte y consulta que permite a los usuario realizar consultas de manera fácil y la visualización de los datos y resultados.

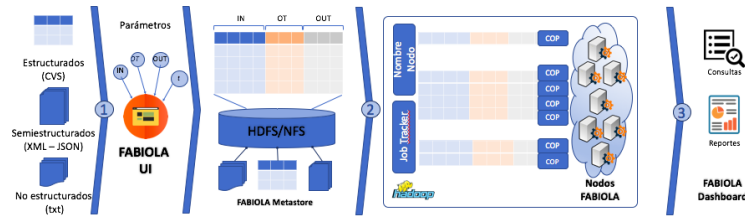


Fig. 1. Arquitectura de FABIOLA

## 3 Ejemplo de Aplicación: Problema de Selección de Distribuidores en la Cadena de suministro de Automóviles

En el área de la industria automovilística, y en cualquier otro tipo de industria, donde tanto la producción como la distribución son fundamentales, es necesario transportar los productos desde los centros de producción ( $n$  Plantas) hacia los centros de distribución ( $m$  Dist), donde  $n \ll m$ . Las diferentes demandas en estos centros de distribución representan distintas soluciones, que pueden representarse por un array bidimensional, llamado  $Assign[n, m]$ . El objetivo es obtener el coste de transporte mínimo con el fin de mantener ventajas competitivas. Las

restricciones de este problema se definen acorde con las características de las diferentes empresas de logísticas del mercado. Siguiendo las partes necesarias en Fabiola, los componentes son:

- **Modelo de Datos:** los datos necesarios para resolver el problema, definido por:
  - **Entrada:** {Capacidad: Array[NFact] of Integer, Demanda: Array[MDist] of Integer, Cost: Array[NFact,MDist] of Integer, NombreEmpresaLogística: String}
  - **Salida:** {TotCost: Integer, Assign: Array[NFact,MDist] of Integer}
- **Problema de Optimización de Restricciones:** El objetivo es *Minimizar* (*TotCost*), además, las siguientes restricciones se tienen que cumplir:

$$\forall p \in \{1..n\} \quad \sum_{c=1}^m Assign[p, c] \leq capacidad[p]$$

$$\forall c \in \{1..m\} \quad \sum_{p=1}^n Assign[p, c] = demanda[c]$$

$$TotalCost = \sum_{p=1}^n \sum_{c=1}^m Assign[p, c] * cost[p, c]$$

- **Ejemplos de Consultas:** ¿Cuáles son las asignaciones y el *TotCost* para la empresa de logística cuyo nombre es “DHL”? ¿Cuáles son las empresas de logística cuyo *TotCost* es menor a 50.000?

## 4 Trabajos Relacionados

Uno de los grandes retos que se presentan hoy en día en Big Data es cómo resolver problemas de optimización cuando los datos son heterogéneos, incompletos e incluso introducen cierta incertidumbre [4, 5]. Además, la rapidez para responder ciertos tipos de preguntas en dicho entorno incrementan la complejidad de resolución. Aunque Fabiola no es una nueva solución de Big Data, si tiene como objetivo ser parte del ecosistema de Hadoop. Fabiola proporciona los componentes necesarios para solucionar problemas de optimización con restricciones con datos distribuidos.

La Programación con Restricciones (CP) presenta grandes retos de escalabilidad resolviendo algunos tipos de problemas muy complicados. Sin embargo, CP ha sido aplicado exitosamente a diferentes dominios resolviendo problemas de optimización, como por ejemplo, en planificación y organización. Big Data proporciona en el paradigma de CP una nueva perspectiva respecto al tamaño y volumen de datos, y es una gran oportunidad para ganar eficiencia y optimización en procesos operacionales [6]. Además, Big Data afronta nuevos retos [7] cuando se trata de toma de decisiones automáticas que incluyen muchos (miles) de variables de decisión y hay que optimizar el consumo de recursos, sostenibilidad de

recursos, y finanzas. Aun así, los problemas de optimización en CP necesitan más flexibilidad y adaptabilidad ya que la exploración es heterogénea, enorme y la generación dinámica de datos requiere de una adaptación rápida del problema de optimización con el fin de obtener soluciones más holísticas y hoy en día no hay herramientas que lo resuelvan.

## 5 Conclusiones

Los problemas de optimización se encuentran en numerosos ejemplos reales. Se convierten en un problema aún más complejo cuando los datos están en un entorno de Big Data, lo que implica una gran cantidad de información, distribuida y heterogénea. Se presenta la arquitectura del marco de trabajo Fabiola. Fabiola permite soportar la definición y resolución de Problemas de Optimización de Restricciones, aislando a los usuarios de dónde se encuentran los datos, y cómo se encuentran las salidas óptimas. Se proporciona además un ejemplo de aplicación con el fin de mostrar la flexibilidad de la propuesta.

## Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Ciencias y Tecnología de España (TIN2015-63502-C3-2-R) y los Fondos Europeos de Desarrollo Regional (ERDF/FEDER).

## References

1. Biswapesh Chattopadhyay, Liang Lin, Weiran Liu, Sagar Mittal, Prathyusha Aragonada, Vera Lychagina, Younghee Kwon, and Michael Wong. Tenzing a sql implementation on the mapreduce framework. In *Proceedings of VLDB*, pages 1318–1327, 2011.
2. Apache hadoop. Available at <http://wiki.apache.org/hadoop>, 2016.
3. Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Suresh Anthony, Hao Liu, Pete Wyckoff, and Raghotham Murthy. Hive: A warehousing solution over a map-reduce framework. *Proc. VLDB Endow.*, 2(2):1626–1629, August 2009.
4. Alexandros Labrinidis and H. V. Jagadish. Challenges and opportunities with big data. *Proc. VLDB Endow.*, 5(12):2032–2033, August 2012.
5. Nasser T and Tariq RS. Big data challenges. *Computer Engineering & Information Technology*, 4(3):1–10, 2015.
6. Barry O’Sullivan. Opportunities and challenges for constraint programming. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI’12, pages 2148–2152. AAAI Press, 2012.
7. Eugene C. Freuder and Barry O’Sullivan. Grand challenges for constraint programming. *Constraints*, 19(2):150–162, 2014.