

Noticias del Departamento de Ingeniería de Sistemas

Novedades



- *Felicitaciones a Sebastián! Flamante Licenciado en Sistemas de Información!*
- *Artículos aceptados en: XI Conference on Cloud Computing, Big Data & Emerging Topics (JCC-BD&ET)*

Sebastián Yaupe: Nuevo Licenciado en Sistemas de Información

Especificación de Procesos para Preparación de Datos en Sistemas Big Data

Resumen. Para extraer información de los datos en sistemas big data, los mismos deben atravesar una serie de escenarios con diferentes funciones, como ingestión, filtrado, transformación, etc.

Considerando la necesidad de entender estos escenarios, existen antecedentes en el modelado de la preparación de los datos para depósitos de datos, que describen actividades relevantes. En particular, se asocia el concepto de procesos considerando que un proceso es una concatenación lógica de actividades, a través del tiempo y lugar, impulsadas por eventos y que a través de su proceso de transformación, cumplen un determinado fin.

Esta definición contiene un elemento esencial en la especificación de procesos – las actividades — que

se pueden definir como acciones sobre un objeto, debido a que el proceso de transformación ocurre a través de ellas. Las actividades están encadenadas a través de una secuencia lógica, que determina en su conjunto las condiciones y capacidades.

En la literatura se utilizan varios términos para sistemas que implementan procesos: sistema de workflow (WfM), Business Process Management Suite (BPMS), motor de workflow y Process Engine. La importancia de la gestión de flujos de trabajo (workflows) viene de la posibilidad de representar explícitamente las estructuras de los procesos en los modelos y en la actuación controlada sobre los procesos de acuerdo a esos modelos.

Con estas consideraciones, este trabajo de tesis se enfoca en incorporar modelado de workflows de procesos en actividades para preparación de datos utilizando BPMN; pero a diferencia de trabajos relacionados, se propone extender a particularidades propias de sistemas big data, como actividades en lagos de datos. El modelo resultante de esta de tesis

es instanciado en un caso de estudio en el dominio geoespacial (análisis de turbidez en el Canal Principal de Riego del Alto Valle), para mostrar su aplicabilidad y la especialización requerida en la instanciación específica.

La tesis fue dirigida por la Dra. Alejandra Cechich y defendida el 27/04/23.

Mis Recuerdos de Estudiante por SEBASTIÁN YAUPE

Cada vez que recuerdo mi camino como estudiante, pienso en la dedicación y esfuerzo que sostuve durante el transcurso de la carrera. Sin embargo, es inevitable no acordarme de las anécdotas compartidas entre clases con los compañeros, y esos nervios que atacaban al ver y escuchar la notificación de un email, que anunciaba las notas de los parciales.

Este camino me mostró mis capacidades, pero también, me dio lecciones de vida que son muy importantes. Algo que aprendí durante la carrera y que voy a llevar conmigo toda



la vida es que “el entorno se encuentra en cambio constante y los sistemas deben adaptarse a los mismos”. Con el tiempo, pude ver presente el concepto antes mencionado, en diferentes situaciones de la vida. En resumen, lo traduzco a mantener la mente abierta, en el aprendizaje continuo y a enfrentar diferentes acontecimientos como una oportunidad para crecer.

Por último, mencionar que estoy muy contento y agradecido de la comunidad que conforma la FAI. Los profesores siempre dispuestos a compartir sus conocimientos y a despejar nuestras dudas, desde luego que hicieron el camino mucho más ameno.

Artículos aceptados en
JCC-BD&ET 2023



1. A Model of Reusable Variation in Big Data System Development

por ALEJANDRA CECHICH, AGUSTINA BUCCELLA, CAROLINA VILLEGAS, AYLÉN MONTENEGRO, ANGEL MUÑOZ, ANDREA RODRÍGUEZ

Abstract. Nowadays, due to the increasing presence of artificial intelligence in software systems, development teams face the challenge of working together to integrate tasks, resources, and roles in a new field, named AI Engineering. Proposals, in the way of models, highlight the needs of integrating two different perspectives – the software and the decision-making (analytic, ML, DL, and so on) systems. But there is something more – both systems must achieve high quality levels for different properties; and this

is not a straightforward task. Quality properties, such as reusability, traditionally evaluated and reinforced through modeling in software systems, do not exactly apply similarly in DM systems.

In this paper, we propose a model for managing reusable assets in AI engineered systems by linking software product line modeling and variety identification. The proposal is exemplified through a case study in the agriculture precision domain.

2. CoVaMaT: Functionality for Variety Reuse through a Supporting Tool

por LÍAM OSYCKA, ALEJANDRA CECHICH, AGUSTINA BUCCELLA, AYLÉN MONTENEGRO, ANGEL MUÑOZ

Abstract. Developing reusable Big Data Systems (BDSs) implies dealing with modeling variety as reusable assets. Conceptually speaking, these assets might be similar to reusable software artifacts built under software product line (SPL) engineering; however, similar does not imply they are the same. Variety identification in BDSs is more related to collecting and preparing data, and of course, analytics; meanwhile SPLs model reusable pieces of software. Although in the end all it is about software, its nature differs as treatment for its reuse does.

In this paper, we introduce our proposal for modeling reusable variety by describing the way it is processed by our supporting tool CoVaMaT (Context-Based Variety Management Tool). We exemplified its functionality through two case studies in the precision agriculture domain.

Competitividad Digital

Internet de las Cosas: Los dispositivos IoT pueden ser testigos de cómo comenzó un incendio!!

Los incendios son difíciles de investigar ya que a menudo destruyen

la evidencia de su causa. Ahora, los científicos son capaces de mostrar cómo los dispositivos de Internet de las cosas (IoT) pueden revelar pistas clave, pero también pueden ayudar potencialmente a los pirómanos a iniciar incendios en primer lugar.



Las investigaciones criminales ya están explorando cada vez más cómo los dispositivos IoT podrían registrar datos forenses útiles. Por ejemplo, en 2021, un jurado encontró a un hombre en Alabama culpable del asesinato de su esposa según los datos de su iPhone. La aplicación de salud iOS del dispositivo registró 18 pasos durante un tiempo en el que afirmó estar dormido la noche del homicidio.

Los científicos de la Universidad de Lausana¹ en Suiza querían ver si los dispositivos IoT podrían servir como testigos digitales de un incendio provocado, ya que los dispositivos podrían almacenar datos en una placa capaz de sobrevivir al incendio, o transmitir datos a la nube o teléfonos vinculados antes de su incineración.

En un experimento en un apartamento de varias habitaciones en un edificio programado para la demolición, los investigadores examinaron cómo los dispositivos IoT podrían resultar útiles en un ataque incendiario que involucre derrames de gasolina y diesel. En este escenario, encontraron que los dispositivos, incluidos sensores de temperatura y movimiento, cámaras inteligentes, detectores de humo y un asistente de voz, ayudaron a revelar el momento preciso en que comenzó el incendio, la habitación en la que comenzó y la línea de tiempo de cómo se propagó.

En otro experimento en un departamento de una sola habitación en un

¹<https://www.unil.ch/index.html>



edificio diseñado para ayudar a capacitar a los bomberos, los científicos exploraron cómo los dispositivos IoT podrían no solo ayudar a los investigadores de incendios, sino también a los pirómanos. Para este escenario, los investigadores instalaron un calentador de inmersión en una caja llena de espuma de poliestireno como un dispositivo de encendido rudimentario. Conectaron el calentador a un tomacorriente inteligente en la cocina y luego usaron el tomacorriente para encender el calentador de forma remota. (Un informe de 2017 encontró de manera similar que una estufa IoT también podría usarse como un dispositivo de incendio provocado por control remoto).

Los científicos descubrieron que a menudo enfrentaban dificultades para recopilar datos de los propios dispositivos IoT. A veces, los incendios destruyeron los chips de memoria, mientras que otras veces los dispositivos no tenían chips de memoria o tenían protecciones de lectura en su memoria interna que los investigadores no pudieron superar.

Sin embargo, los datos almacenados en los teléfonos inteligentes vinculados a menudo resultaron informativos. En el primer experimento, una aplicación recopiló una gran cantidad de datos de temperatura, humedad y movimiento, mientras que otra aplicación proporcionó videos grabados durante el incendio. En el segundo experimento, la aplicación del tomacorriente inteligente registró la hora y la habitación cuando se realizó la solicitud para activar el tomacorriente. Los datos del sensor de calor y dióxido de carbono sugirieron que el fuego comenzó unos seis minutos después de que se activara. Además, los dispositivos IoT en el apartamento revelaron que el fuego creció rápidamente después de la ignición, a diferencia de una acumulación lenta, como podría suceder si, por ejemplo, se dejara un cigarrillo desechado en un sofá.

Estos nuevos hallazgos ayudan a “abrir una nueva frontera”. En el pa-

sado, los investigadores forenses tenían que confiar en la evidencia física para reconstruir lo que sucedió en la escena del crimen o en la escena del incendio, pero ahora tienen un aliado que los ayuda a recopilar información más detallada.

Si te interesa el tema, puedes ver más de esta investigación en el artículo “The Internet of Things: Fire Sleuth, Fire Starter”², publicado en Junio de 2023 en *IEEE Spectrum*. O puedes acceder a artículos especializados como “Ok Google, Start a Fire. IoT devices as witnesses and actors in fire investigations”, por Francesco Servida, Manon Fischer, Olivier Delémont, Thomas R. Souvignet, en *Forensic Science International*, 2023³

Mesa del Arquitecto

La computación sin servidor (serverless computing) es un nuevo paradigma que proporciona una plataforma para desarrollar e implementar aplicaciones sin tener que gestionar una infraestructuras subyacente!!!

En los últimos años, los desarrolladores empezaron a pensar en operar sus sistemas en lugar de operar sus servidores, considerando las aplicaciones como flujos de trabajo, lógica distribuida y almacenes de datos gestionados externamente. Esta forma de el trabajo puede considerarse *sin servidor (serverless)*, no porque los servidores no estén ejecutando, sino porque los desarrolladores no necesitan pensar más en ellos.

Los principales proveedores de la nube (por ejemplo, Amazon Web Services (AWS), Lambda Microsoft Azure Functions y Google Functions) han introducido plataformas informáticas *serverless* que permiten que los desarrolladores se centren únicamente en la lógica del negocio, es decir, excluyendo factores como escalabilidad, aprovisionamiento, e infraestructura. En este caso, el programa técnicamente se ejecuta en servidores externos con el apoyo de proveedores de

servicios en la nube. En las tecnologías *sin servidor*, el proveedor de la nube asigna dinámicamente los servidores; es decir, el código se ejecuta en contenedores sin estado activados por eventos, que pueden durar más de una invocación.

Además, las soluciones *sin servidor* incluyen diferentes tecnologías, que se pueden clasificar como *back-end* como servicio (BaaS) y funciones como servicio (FaaS).

BaaS permite el reemplazo de componentes del lado del servidor con servicios de componentes *off the shelf*. Permite subcontratar los aspectos detrás de escena de una aplicación para que los desarrolladores puedan elegir escribir y mantener la lógica de la aplicación en el *front-end*. Tales ejemplos incluyen sistemas de autenticación remota, gestión de bases de datos, almacenamiento en la nube y alojamiento.

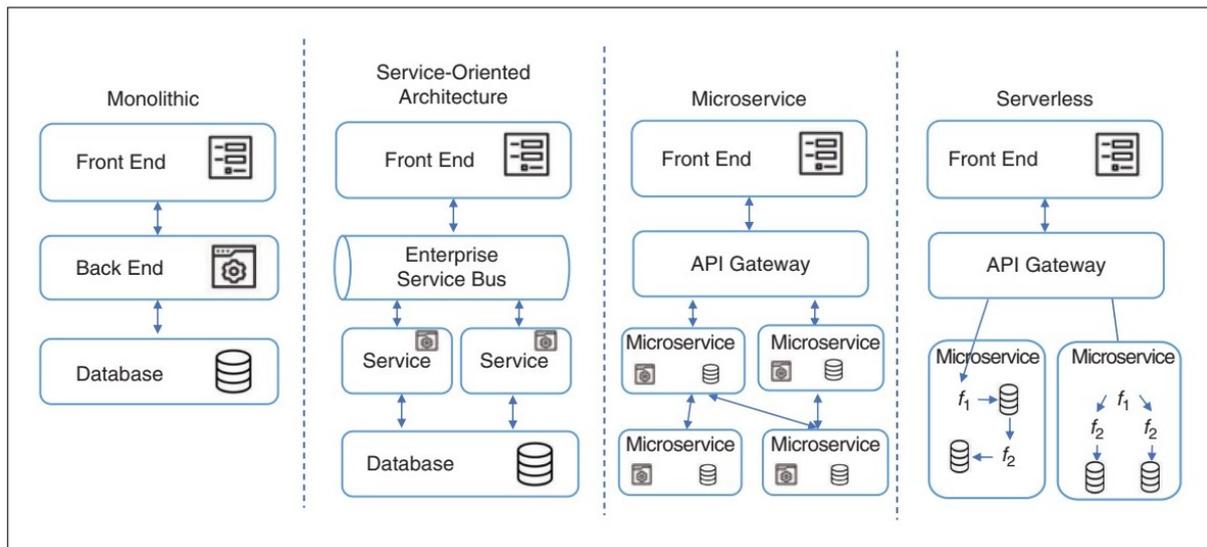
FaaS es un entorno utilizado para software en ejecución, donde las aplicaciones *serverless* son sistemas basados en eventos y basados en la nube y en los que el desarrollo de aplicaciones se construye únicamente mediante una combinación de servicios de terceros, lógica del lado del cliente y llamadas a procedimientos remoto alojados en la nube. FaaS permite a los desarrolladores implementar código que, al activarse, se ejecuta en un entorno aislado. Cada función típicamente describe una pequeña parte de una aplicación completa. El tiempo de ejecución de las funciones está limitado (por ejemplo, 15 min para AWS lambda), por lo que las funciones no están constantemente activas. En cambio, las plataformas FaaS escuchan los eventos que instancian las funciones; por lo tanto, las funciones deben desencadenarse por eventos, como solicitudes del cliente, eventos producidos por cualquier sistema externo, flujos de datos, etc.

La computación *sin servidor* tiene varias características en común con los *microservicios*, pero también tiene varias diferencias. La siguiente figura muestra su evolución:

²<https://spectrum.ieee.org/iot-for-arson-forensics>

³<https://doi.org/10.1016/j.forsciint.2023.111674>.





Para comprender cuál usar, debe entenderse cómo comparar estas dos tecnologías. Los microservicios son servicios pequeños y autónomos con un único y propósito claramente definido destinado a reducir la complejidad de aplicaciones. Permiten la descomposición vertical de las aplicaciones en un subconjunto de servicios independientes impulsados por el negocio y conectados con protocolos de comunicación ligeros, por lo general interfaces de programación (API) en HTTP.

La diferencia clave entre los microservicios y los sistemas sin servidor es la granularidad más pequeña de las funciones sin servidor y la adopción del paradigma impulsado por eventos. Mientras que es posible desarrollar microservicios basados en eventos, es muy recomendable desarrollar aplicaciones sin servidor impulsadas por eventos. Es posible componer diferentes funciones sin servidor para crear un microservicio utilizando diferentes patrones, como el Patrón Saga⁴ y otros. Más detalles so-

bre patrones serverless para microservicios se pueden encontrar en un artículo de Taibi et al⁵

Si te interesa el tema, puedes ver más diferencias entre las arquitecturas, así como ventajas y desventajas del mundo serverless en el artículo completo, en la revista IEEE Software de Enero/Febrero 2021⁶.

¿Sabías qué ...

imágenes cerebrales no invasivas pueden distinguir gestos hechos con las manos?



Un estudio realizado por investigadores de la Universidad de California, San Diego (UCSD) y del Sistema

de Salud de Asuntos de Veteranos, utilizó imágenes cerebrales no invasivas para diferenciar los gestos de las manos como un paso hacia una interfaz cerebro-computadora.

Los investigadores aplicaron magnetoencefalografía (MEG), que utiliza un casco equipado con una serie de sensores para detectar campos magnéticos en el cerebro generados por corrientes eléctricas que pasan entre neuronas. Evaluaron la capacidad de usar MEG para distinguir entre gestos con las manos realizados por voluntarios, a quienes se les dijo aleatoriamente que hicieran un gesto del juego Piedra, Papel, Tijeras. Los investigadores encontraron que las mediciones del algoritmo de aprendizaje profundo MEG-RPSnet podían diferenciar los gestos de las manos con una precisión superior al 85 %!

Si te interesa el tema, puedes ver el artículo ⁷: “Magnetoencephalogram-based brain-computer interface for hand-gesture decoding using deep learning”, Mayo, 2023.

⁴M. Štefanko, O. Chaloupka, and B. Rossi, “The Saga pattern in a reactive microservices environment”, in Proc 14th Int. Conf. Softw. Technologies (ICSOFT 2019), p. 483–490.

⁵D. Taibi, N. El Ioini, C. Pahl, and J. R. Schmid Niederkofler, “Patterns for serverless functions (Function-as-a-Service): A multivocal literature review”, in Proc. 10th Int. Conf. Cloud Comput. Serv. Sci. (CLOSER 2020), pp. 181–192.

⁶<https://www.computer.org/csdl/magazine/so/2021/01/09305905/>

⁷<https://pubmed.ncbi.nlm.nih.gov/37183188/>

