

REPORT REPRINT

Pivotal pitches Spring Cloud Data Flow for continuous data processing

MATT ASLETT

10 FEB 2017

Developed specifically to serve cloud-native applications, Spring Cloud Data Flow is positioned as an enabler for continuous stream- and batch-based microservices-based applications.

THIS REPORT, LICENSED EXCLUSIVELY TO PIVOTAL, DEVELOPED AND AS PROVIDED BY 451 RESEARCH, LLC, SHALL BE OWNED IN ITS ENTIRETY BY 451 RESEARCH, LLC. THIS REPORT IS SOLELY INTENDED FOR USE BY THE RECIPIENT AND MAY NOT BE REPRODUCED OR REPOSTED, IN WHOLE OR IN PART, BY THE RECIPIENT, WITHOUT EXPRESS PERMISSION FROM 451 RESEARCH.



©2017 451 Research, LLC | WWW.451RESEARCH.COM

Spring Cloud Data Flow was created by redesigning the Spring XD integration project to better serve cloud-native applications. It is positioned by Pivotal as an enabler for continuous stream- and batch-based microservices-based applications.

THE 451 TAKE

We were remiss in not mentioning Spring Cloud Data Flow in our overview of continuous data integration in late 2016. To some extent, this can be explained by the fact that the project is targeted more at application developers, rather than the data management professionals that have more typically been responsible for data integration initiatives. This is changing, however, and the continuous data integration approach itself is a recognition that the development and management of data processing and integration pipelines need to reflect the increased agility that modern application development concepts can provide. With the Spring Cloud Data Flow project, Pivotal is well positioned to serve the growing need for data-driven, cloud-native microservices-based applications.

CONTEXT

In late 2016, 451 Research made the case for a continuous approach to data integration, to enable organizations to satisfy the desire to increase the frequency with which they analyze data, in order to deliver greater business agility and business benefits, such as accelerated development times and improved customer service. Continuous data integration is not to be confused with continuous integration, which 451 Research defines as the practice of developers regularly pushing code into a shared repository to drive an automated build and test process, as part of a larger continuous delivery and deployment application-development strategy.

However, we believe that the key stages in a continuous integration and delivery process (plan, code, build, test, release, deploy, operate, measure) could also equally be applied to the process of developing, deploying and managing data integration pipelines that are responsive to changing business and data processing requirements. The argument struck a chord with Pivotal, which has been working toward more agile approaches to data integration and data processing through the open source Spring Cloud Data Flow project, which supports composable and continuously-delivered microservices for ingesting, transforming, storing and analyzing data to serve cloud-native applications.

The roots of Spring Cloud Data Flow can be traced back to Spring XD, a project that was launched in early 2013, almost simultaneously with the birth of Pivotal via the combination of EMC and VMware's software assets. Spring XD (eXtreme Data) was designed to enable users to build distributed data processing pipelines for both real-time and batch applications, typically deployed on Apache Hadoop or Apache Spark.

The switch to support composable microservices and cloud-native applications began with the launch of Spring Cloud Data Flow in mid-2015, essentially redesigning Spring XD to take advantage of the Spring Boot microservices framework, with the former Spring XD runtime replaced by native support for multiple runtimes, such as Cloud Foundry, Apache YARN, Apache Mesos and Kubernetes. As a result of Spring Cloud Data Flow, the former Spring XD sub-project, Spring Integration, has evolved into Spring Cloud Stream to enable the creation of event-driven streaming applications. The former Spring Batch is now Spring Cloud Task, which enables the creation of batch jobs that can be run as short-lived executable data applications.

This is no mere rebranding exercise, however. As noted, Spring Cloud Data Flow has been redesigned to act as an orchestration process for RESTful message-driven microservices-based applications using Spring Cloud Stream and Spring Cloud Task.

According to Pivotal, Spring Cloud Data Flow overcomes a number of limitations of traditional approaches to data integration and data processing, supporting multi-direction data flows, improved agility, and cloud-native applications, and delivering continuous data delivery, improved data reuse, and improved scalability and resilience.

The resulting microservices applications can run natively in multiple runtimes via implementations for Cloud Foundry, Apache Yarn, Apache Mesos and Kubernetes. Spring Cloud Data Flow also offers binders for messaging software, including Apache Kafka, RabbitMQ, Google Pub/Sub and JMS, for integration with external data sources and targets. While the Spring Cloud Data Flow project was born in mid-2015, version 1.0 became generally available in July 2016, and is now up and running with five Pivotal customers in production. Cited use cases include modernizing ETL (extract, transform and load) to support heterogeneous data sources and continuous integration and delivery processes, as well as modernization of data integration processes to support event-driven architecture.

COMPETITION

We previously cited a number of different technologies and vendors that could be used by enterprises to serve a continuous data integration approach. These include the similarly named Hortonworks DataFlow (HDF) offering, which combines Apache NiFi for creating data pipelines to ingest, transform and deliver data for storage; Apache Kafka for publish and subscribe data ingestion; and Apache Storm for streaming analytics. Another vendor directly targeting continuous data integration is Striim, which offers a visual interface and declarative programming interface to create data-ingestion and -integration pipelines that are then also deployed, monitored, visualized and analyzed.

Striim's capabilities for the integration and analysis of streaming data are based on its change data capture (CDC) approach, which enables users to capture and ingest a continuous feed of data as it changes on the source platform (in addition to an initial bulk ingestion). There are various CDC-based products and services that could also be part of a continuous data integration strategy, including more traditional offerings from the likes of Attunity, Talend and IBM, and open source projects such as Databus (originally created by LinkedIn). HVR's software, for example, is designed to provide asynchronous log-based data capture and change-based data replication, as well as data definition language (DDL) generation.

StreamSets is another vendor in this space with the open source Data Collector tool for building and operating data-movement pipelines; more recently, it expanded its portfolio with the launch of Dataflow Performance Manager, which is designed to 'map, monitor and master' data flows, rather than simply connect and visualize them. The major cloud providers also have relevant services that are designed to enable users to continuously integrate data into their respective cloud services, such as the AWS Kinesis family of stream-processing services, Google's Cloud Dataflow or Microsoft's Azure Data Factory.

With regard to the microservices-based approach, MapR is another vendor that could come into consideration with its microservices framework. It leverages MapR Streams (which supports the Apache Kafka API) and is integrated with its file system (MapR-FS) and NoSQL database (MapR-DB) as part of its overall Converged Data Platform offering.

SWOT ANALYSIS

STRENGTHS

Pivotal enjoys the benefits of being owned by a much larger parent company (Dell Technologies), as well as the autonomy to function like a startup targeting emerging application-development and cloud transformation projects.

WEAKNESSES

The company's portfolio of products is broad, and it is arguably better known by customers for individual products (e.g., Cloud Foundry, or the Greenplum Database or Spring) than for its full suite of products and services.

OPPORTUNITIES

Pivotal has the potential to exploit the breadth of its product portfolio for cross- and upselling opportunities, particularly in long-standing customers embarking on digital and cloud transformation initiatives.

THREATS

The breadth of the portfolio means that the company is competing on multiple fronts, with both focused, emerging startups and established incumbents alike.