GAIN BIG INSIGHT FROM BIG DATA

CSC Big Data Platform as a Service (PaaS), powered by Infochimps, a CSC Big Data Business, makes it faster and far less complex to develop and deploy enterprise Big Data applications. Whether you need real-time analytics on multisource streaming data, a scalable NoSQL database or an elastic Hadoop cluster, CSC provides the easiest step to Big Data. CSC Big Data PaaS enables large-scale data processing, data collection and integration, data storage, data analysis and visualization, and infrastructure management. Coupled with our expert team and a revolutionary approach to tying it all together, we help you accelerate your Big Data projects.

EXPLORE THE TECHNOLOGY

CSC Big Data PaaS is a suite of managed Big Data infrastructure, which means we provide the expertise, maintenance and support so that your team and resources can stay focused on creating insights, not herding technology.

CSC Big Data PaaS represents the machine systems, managed by CSC, that you leverage to process or store your data. Figure 1 shows an overview of our Big Data PaaS model. We categorize analytics into three buckets: Streams for real-time stream processing, Queries for databases and ad hoc query engines, and Hadoop for batch processing.

While you can interact with platform technologies directly if you’d like to, such as connecting directly with a database and using native Hadoop tools, our solution architecture includes the Application Layer, which provides application development accelerators to make performing analysis and building apps easier, and components that can be bundled with CSC Big Data PaaS, such as industry analytics and applications, business intelligence tools and statistical analysis.

Underneath all of this is a DevOps configuration management and provisioning toolset, which serves as the foundation for a repeatable, robust CSC Big Data PaaS. It’s built from a combination of leading open source technologies allowing CSC to orchestrate infrastructure at the system diagram level (manage clusters, not machines) and deploy to a variety of cloud providers to meet any client needs.
USING CSC BIG DATA PaaS
CSC Big Data PaaS has several key interface points. This document will explore in more detail these key areas:

STREAMS
• Collect data
• Perform real-time streaming data processing

QUERIES
• NoSQL database and ad hoc queries
• Build interactive data apps

HADOOP
• Perform large-scale batch data processing

STREAMS
Streams leverages Apache Storm and Apache Kafka to provide scalable, distributed, fault-tolerant data flows. Storm is multinode and modular, so you can have fine-grained control over how you modify and scale your data flows and stream processing.

FLOW COMPONENTS
All data processed by Streams exists as a stream of data-called events, which consists of a body and some associated metadata. Events flow through a series of logical nodes chained together.

Streams connects listeners, which produce events, to writers, which consume events. Events can be modified in-stream by processors as part of Storm topologies.
Streams extends Storm and Kafka by adding hardened deployment and monitoring functionality, a development toolkit for easier development of data flows and stream processes, and an array of standardized listeners and writers for easier data integration and ingestion. The standard listeners and writers, which are very customizable, provide the primary interfaces for data input and data delivery (shown in Figure 2).

Standard data listeners include:

- HTTP (post, stream, Web socket)
- TCP/UDP (log or machine data)
- Data partners (Gnip, Moreover, etc.)
- Batch upload (S3 or OpenStack Swift bucket, FTP, Sqoop, etc.)
- Custom connectors for unique or legacy systems

Standard data writers target:

- Applications (REST API)
- Databases (Elasticsearch, HBase, external databases)
- File systems (Hadoop HDFS, S3 or OpenStack Swift, local file system)

ARCHITECTURE DESIGN

Designing data flows to take advantage of the strengths of CSC Big Data PaaS and to match the output that your applications or databases expect requires critical upfront focus on architecture and design. To speed you along in your pursuit of this goal, CSC will help you:

- Choose which prebuilt listeners, writers and processors are appropriate for your data flow
- Choose what sorts of processing should be done in-stream vs. batch or offline
- Design your data flows so they can match the schema of your business applications
MULTIPLEXING AND DEMULTIPLEXING
Not only can Streams process events from a listener, through a chain of processors, and into a writer; it also can split or recombine flows, as shown in Figure 3. Splitting flows is useful for several reasons:

• Different events may need to be routed to different downstream applications or writers (high security vs. low security, regulated vs. nonregulated, etc.).
• The same event may need to be duplicated and routed to two different downstream applications or writers (taking advantage of two different databases for the same data, sending a sample of live data to a testing environment, etc.).
• Aggregating events from several different flows is also a common pattern. Different systems (POS systems, Web logs, mobile devices) may be normalized in their own separate flows and then combined before being written to a database.

Figure 3. CSC Supports Process Event Flow Aggregation and Disaggregation
PERFORM STREAM PROCESSING WITH PROCESSORS

DEVELOPING PROCESSORS

The true power of Streams is the ability for developers to write and chain together processors within a flow — enabling real-time analytical processing. While processors can be developed by CSC, in many cases your organization will choose to do much of this development in-house. Processors will often contain custom business logic (“Filter all events older than 2 weeks” or “Transform data from source X so it looks like source Y”). CSC makes it easy for you to develop, test and deploy your code and business logic into the flow.

A processor in Streams receives an event and must decide what to do. A processor has access to:

• The string body of the event
• Key-value metadata associated with the event
• Global configuration information available from the Cloud API
• Any databases or other shared platform resources

A processor has complete freedom as to what it does when it receives an event. It can:

• Transmit the event unchanged, perhaps incrementing counters somewhere else in the platform (counting, monitoring)
• Modify the event and then transmit it (normalization, transformation)
• Use a database or external Web service to insert new data into an event and transmit it (augmentation)
• Not emit the event at all (filtering).

Keep in mind that a processor can only examine the individual event it is currently processing or reach out to shared resources like configuration or databases. It cannot look at a different event at the same time.

DEPLOY PACK

CSC Big Data PaaS uses the concept of a “deploy pack” to define custom processors and to control deployment within Streams.

A deploy pack is a Git repository with the following purpose:

• Hold the implementation of various processors
• Provide an environment for local testing of processors
• Define how processors, listeners and writers are wired together in a real data flow

Your instance of CSC Big Data PaaS will come with a deploy pack that you can clone, modify and push changes back to. When you push your changes back, CSC Big Data PaaS will take necessary actions to deploy the changes you’ve made live into Streams.

USING THE TRIDENT DSL TO STITCH TOGETHER PROCESSORS

Trident is a Java framework built on top of Storm that is included with the open source, community Storm project. Trident’s goal is to let you describe the data transformation or flow you’re trying to build as simply, quickly and elegantly as possible.

The benefits to this approach:

• Instead of the traditional Java write/compile/package/upload cycle familiar to most Big Data developers, Trident helps you increase speed of development and improve your ability to iterate.
• Trident provides various transactionality guarantees (which Storm alone doesn’t provide out of the box).
• Trident uses a “fast batch” processing model.
• Trident provides useful functions such as streaming joins/aggregations and streaming filters (which Storm alone doesn’t provide out of the box).

QUERIES
NOSQL DATABASE AND AD HOC, QUERY-DRIVEN ANALYTICS
CSC Big Data PaaS offers a choice of distributed, linearly scalable NoSQL databases and Hadoop query engines. Not every database can perform every task well, and one of the advantages of CSC Big Data PaaS is that data can be routed or mirrored across databases to optimize for different use cases or modes of access. The databases and query engines currently offered by CSC Big Data PaaS include:

• Elasticsearch: A distributed, linearly scalable database and document store with a rich data model oriented around full-text search and faceted analytical queries
• HBase: A distributed, linearly scalable database appropriate for time series analysis and other numerical or otherwise flat data that can handle billions of rows of data and is deeply integrated with Hadoop
• Hive: A SQL-based query interface that provides data warehousing functionality on top of Hadoop
• Impala/Stinger: Interactive query engines that leverage Hive’s SQL-based query interface but allow much faster, ad hoc access to data in Hadoop
• Vayacondios: A fast, analytics-oriented document store and server, based on MongoDB, for small amounts of configuration information and time-based event data, logging and monitoring metrics
• External databases: We can also send data to an external database, managed/ supported by a third party

CSC offers multiple databases for Big Data applications because leveraging the strengths of many databases to support an enterprise application is often the only way possible to build robust, scalable Big Data applications. Your financial time series analysis application might have several thousand or a million users (a small number) but tens of billions of data points for those users to analyze (a large number). No database in existence can adequately handle both the user management and querying over billions of your data points. Commonly, developers choose to store information such as user management in a different data store, either inside or outside the scope of the Big Data PaaS hosting option.

For these and other reasons, your Big Data application will need to query data from multiple databases.

This can present a problem: because data now lives in multiple places, how can you be sure you’re reading it from the right place or writing it to the right place? The simplest answer is that you must understand the structure of the Big Data problem you’re solving and use the right databases to store the correct data in an appropriate schema so that your application can be efficient and powerful.

CSC Big Data PaaS will let you:
• Choose the most appropriate database for your application
• Understand what data you can continue to use from your own environment
• Design a schema for your new data that matches the strengths of the database(s) you’re using
INTERACTING WITH YOUR DATA
The way you interact with your data will depend on the database in which it is stored.

ELASTICSEARCH
Elasticsearch is an open source, Lucene-based, distributed, linearly scalable database with a rich data model, optimized for full-text search. Elasticsearch is a great choice for housing data that users will query via a full-text search interface. Powerful query predicates, faceted search, geo-queries and time series are all natively supported by Elasticsearch and easily accessible through its APIs.

Elasticsearch is also schema-less. Different rows of the same “table” in Elasticsearch can have a completely different document structure. New fields can be added to existing documents on the fly.

Elasticsearch’s rich document model means it can support the associations necessary to power a traditional Web application.

There are several options for interacting with Elasticsearch:
• An HTTP-based, REST API, accessible from any Web framework that allows for external API calls (including JSONP) or any programming language or tool that can make HTTP requests and parse JSON
• Using Apache Thrift from any programming language that supports it
• Using the memcached protocol
• Using the native Java API
• Via one of many client libraries

HBASE
HBase is a distributed, scalable key-value database with a strong focus on consistent data and being able to partition cleanly across data centers.

HBase is the right choice when data can naturally be sorted and indexed by a “row key.” Time series data keyed by time or user data keyed by user ID are both good use cases for HBase. HBase also couples very cleanly to the Hadoop framework, meaning that MapReduce jobs run very efficiently on data stored in HBase.

There are several options for interacting with HBase:
• An HTTP-based, REST API, accessible from any Web framework that allows for external API calls or any programming language or tool that can make HTTP requests and parse JSON
• Using Apache Thrift or Avro from any programming language that supports them
• Using the native Java API
• A command-line shell

HIVE/IMPALA/STINGER
Hive is a SQL-based query interface for Hadoop. It traditionally sits on top of MapReduce, translating the HiveQL (SQL-like) queries into batch jobs that can be executed on data in the Hadoop Distributed File System (HDFS).

Hive is the right choice when you are looking to empower business analysts and data experts in your enterprise who have a lot of experience leveraging the SQL query language and reporting approach. By coupling the use of Hive with best practices for metadata management and organizing your data in Hadoop, SQL can provide a very powerful, accessible way to perform ad hoc queries on large amounts of HDFS data.
Impala and Stinger are open source, community projects for Cloudera and Hortonworks, respectively, that provide alternative engines for executing Hive queries. Impala leverages Cloudera’s Impala interactive query engine, and Stinger leverages Hortonworks’ Tez interactive query engine/alternative Hadoop processing execution environment. In both the case of Impala and of Stinger, the SQL-like HiveQL language is still the interaction point, and the various interfaces below apply in all cases.

There are several options for interacting with Hive, Impala and Stinger:

- An HTTP-based, REST API, accessible from any Web framework that allows for external API calls or any programming language or tool that can make HTTP requests and parse JSON
- Using Apache Thrift or Avro from any programming language that supports them
- A command-line shell
- Hue graphical user interface (see Figure 4).

VAYACONDIOS

Vayacondios is based on MongoDB, a fast, analytics-oriented, document store, optimized for fast, in-place updates and rich queries.

Vayacondios plays the role of the Command & Control database for a given CSC Big Data PaaS solution, and it can hold various pieces of information needed for cross-platform configuration and orchestration, as well as for out-of-band monitoring and logging.

Cross-platform monitoring metrics are automatically stored into Vayacondios. Customers can write their own configuration, out-of-band logging and alerts, and small-scale event data into Vayacondios as required for their given application.

Vayacondios is exposed as part of the Cloud API, which provides an HTTP-based API endpoint. With simple JSON commands, you can orchestrate dynamic cloud operations and access unified views into your data and applications.

For example, you might be creating a data flow that is gathering social media data on behalf of a client tracking the words “water” and “tea.” If this user adds the tag “coffee” within your application, you might want the end result to be that Streams automatically starts collecting data about “coffee.”
HADOOP

Elastic Hadoop and Large-Scale Batch Analytics

PERFORM HADOOP PROCESSING
Every interaction with Hadoop occurs either via a MapReduce job that executes as a batch process or as an interactive SQL query over accumulated, historical data. CSC Big Data PaaS exposes Hadoop through a variety of interfaces.

DESIGNING CLUSTERS
CSC Big Data PaaS provides clusters that are optimized for your Hadoop use cases. Clusters can be in a variety of configurations, each configuration available in the appropriate size for your problem:
• Permanent clusters: up 24/7, traditional model, shared execution environment, and so forth
• Ephemeral clusters: only up when you need them
• Ad hoc clusters: for small teams, individuals or testing

NATIVE HADOOP ECOSYSTEM
CSC Big Data PaaS currently supports the Cloudera Distribution of Hadoop and Hortonworks Data Platform. In addition to the core HDFS and MapReduce components required to support Hadoop MapReduce job execution, CSC Big Data PaaS can include installations of common ecosystem tools, such as:
• CSC Big Data PaaS Queries components that integrate closely with Hadoop, including HBase, Hive, Impala and Stinger
• Pig: a high-level data-flow language and execution framework
• Zookeeper: a high-performance coordination service for Big Data processes
• Mahout: a machine-learning and data-mining library
• Sqoop: a Hadoop tool for interacting with and importing data from RDBMS relational data stores
• Oozie: a workflow orchestration and scheduling tool for Hadoop processing queries and jobs

Combined with the traditional Java interface, both experienced and new Hadoop developers will find the right combination of interfaces for Hadoop to provide both power and accessibility.

OPTIMIZING AND MONITORING A JOB
Hadoop distributions provide a built-in JobTracker (in Hadoop 1.x) or Resource Manager (in Hadoop 2.x) for coordinating the execution of jobs across a cluster. These resource managers provide basic counters, progress meters and job logs and are exposed by CSC Big Data PaaS.

CSC Big Data PaaS additionally includes an improved Hadoop job-tracking dashboard built into Hue. This dashboard recapitulates much of the information available through the existing JobTracker but adds user experience improvements that illustrate the job's completion and additional helpful information.

CSC can provide guidance and support in developing the best possible Hadoop processing jobs and queries, in collaboration with our customers and partners. Monitoring and optimization of Hadoop apps and analytics can be complex, given the many operational and configuration nuances of Hadoop and the many tools that are all interacting with one another.
RUNNING WORKFLOWS
A workflow is a set of Hadoop jobs that run in sequence, with a given job’s execution dependent on prior jobs in the sequence. Often as part of complex Hadoop processing operations, scheduling and dependency management becomes a key to effective use of Hadoop in production. Oozie is the standard ecosystem tool provided for this function and the option that CSC Big Data PaaS provides by default; however, many other tools in the community are developed to assist with these types of operations, such as Azkaban and Chronos.

DEPLOYING THE CSC BIG DATA PAAS
CSC Big Data PaaS offers a choice of deployment options to suit different use cases and data protection requirements:

PUBLIC CLOUD
CSC Big Data PaaS can be quickly deployed as a fully managed service in public clouds such as Amazon Web Services. CSC seamlessly supports the instant scalability and ease of use of multitenant public clouds. Public clouds are also ideal for rapidly developing pilot projects before selecting another deployment option.

VIRTUAL PRIVATE CLOUD
Many enterprises choose virtual private clouds to deploy their Big Data projects so they can gain the advantages of a trusted 24/7-hosted, single-tenant environment with the confidence of enterprise-class service-level agreements. CSC Big Data PaaS can be bundled with CSC CloudCompute, which provides a global network of secure, hosted Infrastructure as a Service.

ENTERPRISE PRIVATE CLOUD
If your requirements are to manage your own private Big Data cloud, you can still gain all the advantages of CSC Big Data PaaS. In addition to providing a powerful, flexible foundation for Big Data projects, CSC eliminates the burden of administering complex Big Data technologies, letting enterprises focus on their applications instead. CSC Big Data PaaS can be bundled with CSC BizCloud, which provides a complete, on-premises Infrastructure as a Service solution.

DEDICATED CLUSTER
This is a dedicated solution, purpose built to host Hadoop and other elements in a scalable and resilient architecture. Options include CSC-hosted Dedicated Cluster, Dedicated Cluster in a partner data center, and Dedicated Cluster in a customer data center.

CASE STUDY: INFOMART™
Infomart is Canada’s largest provider of news and broadcast media monitoring and financial and corporate data. Infomart’s goal is to continue to strengthen and extend these data offerings with social media monitoring and analytics capabilities. By coupling social media with its other data sources, Infomart moves toward a much more comprehensive service and achieves significant value-adds for its customer base.
As shown in Figure 5, Infomart uses the Cloud API in conjunction with Streams to link incoming Gnip social media content with the specific rules defined by Infomart customers. Infomart owns the relationship and rules definitions within Gnip, so by using the Cloud API, Infomart can communicate configuration changes. Streams will use these mappings to dynamically make appropriate connections to Gnip and to route Gnip data to the correct customer indices within Elasticsearch.

Infomart is writing a series of processors that perform in-stream analytics in Streams such as real-time sentiment analysis. Wukong and the Deploy Pack are simplifying that process. Infomart also is writing Hadoop jobs that run against data in the Elasticsearch database or direct from HDFS. Included with Elasticsearch, the Elasticsearch API allows for powerful, complex queries of data in the Elasticsearch database. Infomart is using this interface point for its applications and reporting views.

By leveraging CSC Big Data Platform as a Service, Infomart has unlocked tens of millions of dollars of new revenue — a 100x + ROI over its Big Data platform investment.

ABOUT CSC

CSC makes it faster, easier and far less complex to build and manage Big Data applications and quickly deliver actionable insights. With CSC Big Data PaaS, enterprises benefit from the easiest way to develop and deploy Big Data applications in public clouds, virtual private clouds, enterprise private clouds, or dedicated clusters.

Learn more at www.csc.com/bigdata. Contact us at www.csc.com/contact_us to request a demo or for more information.
Worldwide CSC Headquarters

The Americas
3170 Fairview Park Drive
Falls Church, Virginia 22042
United States
+1.703.876.1000

Asia, Middle East, Africa
Level 9, UE BizHub East
6 Changi Business Park Avenue 1
Singapore 468017
Republic of Singapore
+65.6809.9000

Australia
26 Talavera Road
Macquarie Park, NSW 2113
Australia
+61(2)9034.3000

Central and Eastern Europe
Abraham-Lincoln-Park 1
65189 Wiesbaden
Germany
+49.611.1420

Nordic and Baltic Region
Retortvej 8
DK-2500 Valby
Denmark
+45.36.14.4000

South and West Europe
Immeuble Balzac
10 place des Vosges
92072 Paris la Défense Cedex
France
+33.1.55.707070

UK, Ireland & Netherlands
Floor 4
One Pancras Square
London
N1C 4AG
United Kingdom
+44.020.3696.3000

About CSC
CSC is a global leader in next-generation IT services and solutions. The company’s mission is to enable superior returns on our clients’ technology investments through best-in-class industry solutions, domain expertise and global scale. For more information, visit us at www.csc.com.