Cask Data Application Platform

Cask Data Application Platform (CDAP)

CDAP is an open source, Apache 2.0 licensed, distributed, application framework for delivering Hadoop solutions. It integrates and abstracts the underlying Hadoop technologies to provide simple and easy-to-use APIs and a graphical UI to build, deploy, and manage complex data analytics applications in the cloud or on-premises.

Accelerated ROI
Faster time to market, faster time to value

Minimize Cost
Dramatically increase developer productivity and time to production

Stay Flexible and Future Proof
Distribution and deployment agnostic

Onboard Rapidly
Simple, easy, and standard APIs for developers & operations

Promote Reuse and Self-Service
Extensible libraries and point-and-click user interfaces

Support Different Workloads
Streaming and batch, transactional and non-transactional

Container Architecture

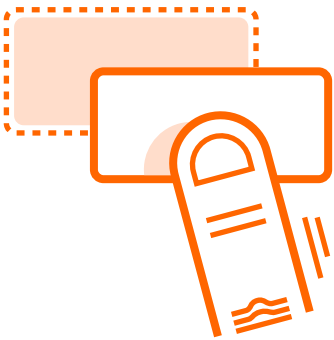
CDAP provides a container architecture for your data and applications on Hadoop. Simplified abstractions and deep integrations with diverse Hadoop technologies dramatically increase productivity and quality. This accelerates development and reduces time-to-production to get your Hadoop projects to market faster.

- Predictable, repeatable, and reliable runtime environment through a container architecture to wrap applications, data services, and all their dependencies, libraries, and configurations into a single package
- Abstracted APIs, reusable libraries, testing frameworks, Maven archetypes, and local standalone environment for rapid development of end-to-end solutions on Hadoop
- Scalable and reliable production runtime environment and operational tools for easy deployment and management of solutions on Hadoop
- Open, standards-based architecture, and REST APIs to integrate and extend existing infrastructure
- Consistent support for batch and real-time data pipelines
- Broad ecosystem integration for runtime, transport and storage, MapReduce, Spark, Spark Streaming, HBase, Tigon, Kafka, and more
- Build-once-run-anywhere flexibility for running solutions on a variety of public or private cloud infrastructures or on-premises; migrate to cloud or vice-versa effortlessly
- Enhanced reusability for datasets and applications increases productivity, quality, and reliability, and decreases time-to-market effortlessly

Self Service Hadoop

CDAP provides rich visual interfaces, seamless integrations, and simple APIs to broadly expand user access to Hadoop. From data scientists to data analysts to data architects and app developers, CDAP enables platform teams to easily extend access to data and analytics while enforcing standard best practices and company policies. Easily meet your cost savings and revenue objectives by enabling more self-service and empowering the domain experts in your organization.

- Empower your developers and internal customers to quickly go from ideation to deployment of Hadoop solutions using our sleek visual interfaces and interactive shell
- Simplify, streamline, and automate deployment and monitoring of solutions in CI/CD or other environments using the comprehensive REST APIs and DevOps tools
- Gain a competitive edge and organizational efficiencies by using CDAP Extensions (Cask Hydrator and Cask Tracker) for ingesting, processing, and tracking data
- Gain greater insights with an interactive dashboard for monitoring applications and data, and drill-downs to identify issues faster with log integrations
- Seamless integration with visualization and data services making datasets immediately available to reports and applications
- Connectivity to disparate datasets using ODBC/JDBC drivers to integrate with Excel, Tableau, and others
- A comprehensive collection of pre-built building blocks to support data manipulation, data storage, and data analytics for rapidly building smarter end-to-end solutions without writing manual code



Enterprise and Production

CDAP is an open architecture based on standards that make it an ideal solution for embedding into Hadoop solutions and aligning with existing enterprise architectures to enable a modern enterprise big data platform. CDAP includes extensive security, compliance and multi-tenancy features along with integrated tools and a simplified architecture. You can enable new processes and accelerate your transformation to a data-driven business by getting applications to market faster - without sacrificing your enterprise requirements.

- Deep Enterprise integrations for security, authorization and authentication, such as LDAP, JASPI, Active directory, KerberOS and Apache Sentry
- Flexible, multi-tenant deployment capabilities to accommodate shared data and application infrastructure
- Maximum flexibility and reduced risk with insulation from changes in the fast evolving big data ecosystem
- Improved visibility through in-built capturing of technical, business, and operational metadata, with the ability to track data flow and identify provenance
- Deep integrations with underlying Hadoop projects accelerates access to the latest technologies and capabilities, product prototyping, and delivery
- Support for the latest Hadoop distributions from Cloudera, Hortonworks, MapR, Microsoft Azure, and Amazon EMR
- Full development and production support with access to technical experts
- Ensure success with your Hadoop projects with an average training time of 1 week and go-to-market cycle of 12 weeks

It's built for

Developers

Operation managers

Data Engineers

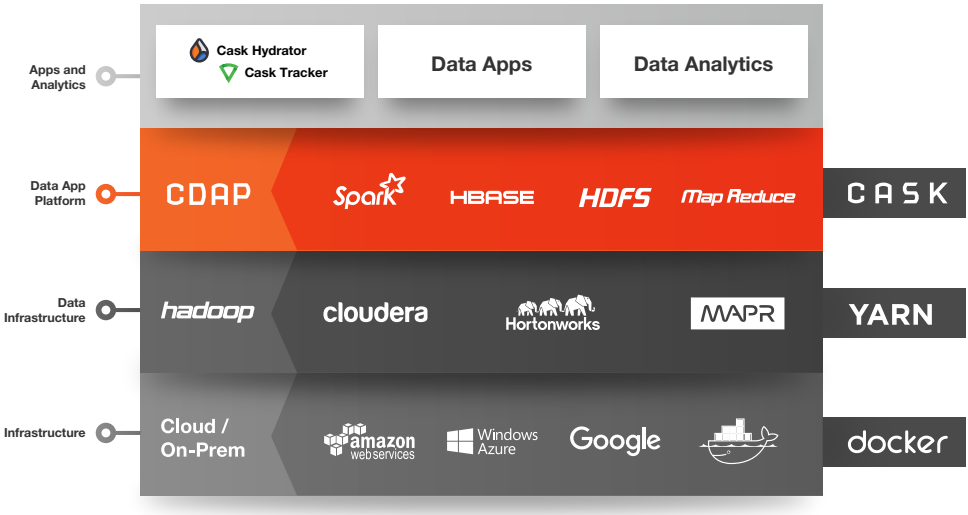
Data Scientists

You can build

- Data Ingestion Applications - Batch or Real-time Data
- Data Processing Workflows
- Real-time Applications
- Data Services
- Predictive Analytics Applications
- Business Analytics Applications
- Social Applications, and many more

Containers on Hadoop

Cask Data Application Platform (CDAP) integrates and abstracts the underlying infrastructure and provides containers for your data and applications. CDAP lets you spend your time delivering applications and insights, not Infrastructure and Integration.



CDAP is based on a container architecture for data and applications on Hadoop. It uses distributed containers to standardize and encapsulate the data and programs stored and running in systems like HDFS, HBase, Spark, and MapReduce.

Encapsulation of data access patterns and business logic enables portability and reusability.

Standardization of data in varied storage engines and compute on varied processing engines simplifies security, operations, and governance.

Packaging of data and applications simplifies the full production lifecycle.

Data Containers

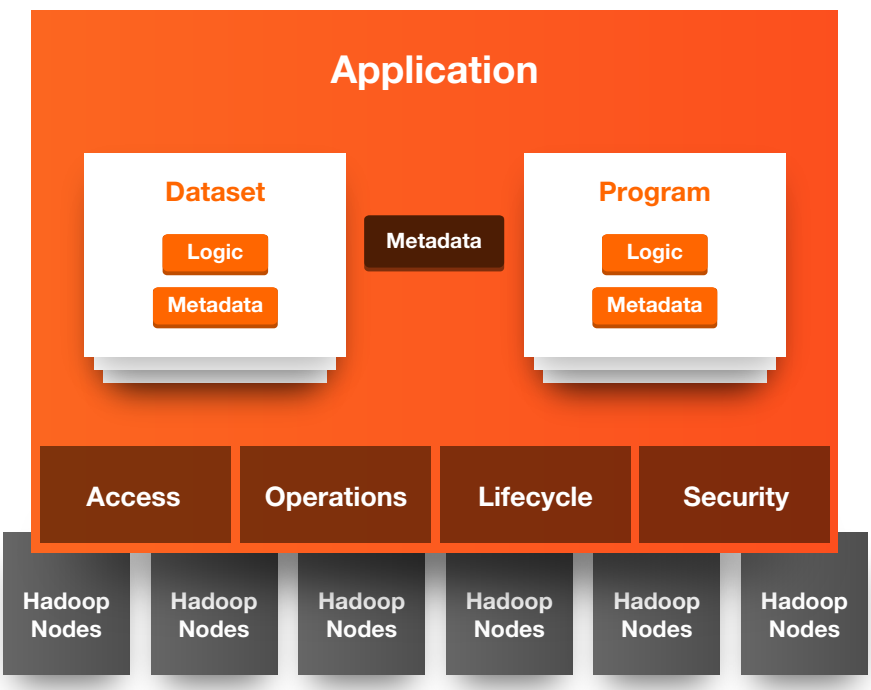
CDAP Datasets provide a standardized, logical container, and runtime framework for data in varied storage engines. They integrate with other systems for instant data access and allow the creation of complex, reusable data patterns.

Program Containers

CDAP Programs provide a standardized, logical container, and runtime framework for compute in varied processing engines. They simplify testing and operations with standard lifecycle, and operational APIs and can consistently interact with any data container.

Application Containers

CDAP Applications provide a standardized packaging system and runtime framework for Datasets and Programs. They manage the lifecycle of data and apps, and simplify the painful integration and operations processes in heterogeneous infrastructure technologies.



Case Study

Data Lake

Building an enterprise data lake requires building a reliable, repeatable, and fully operational data management system, which includes ingestion, transformations, and distribution of data. It must support varied data types and formats, and must be able to capture data flow in various ways. The system must support the following:

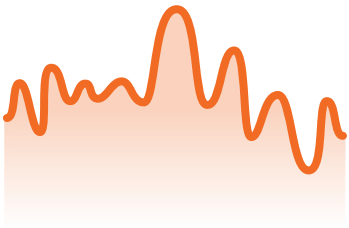
- Transform, normalize, harmonize, partition, filter, and join data
- Interface with anonymization and encryption services external to the cluster
- Generate metadata for all data feeds, snapshots, and datasets ingested, and make it accessible through APIs and web services
- Perform policy enforcement for all ingested and processed data feeds
- Tracking and isolation of errors during processing
- Performing incremental processing of data being ingested
- Reprocessing data in case of failures and errors
- Apply retention policies on ingested and processed datasets
- Setup common location format (CLF) for storing staging, compressed, encrypted, and processed data
- Filtered views over processed datasets
- Monitoring, reporting, and alerting based on thresholds for transport and data quality issues experienced during ingestion. This helps provide the highest quality of data for analytics needs
- Annotate Datasets with business/user metadata
- Search Datasets using metadata
- Search Datasets based on schema field names and types
- Manage data provenance (lineage) as data is processed/transformed in the data lake

Outcome

- A team of 10 Java (non-Hadoop) developers were able to build an end-to-end ingestion system with the capabilities described above using CDAP. **Lower barrier to entry**
- These developers provided a self-service platform to the rest of the organization(s) to ingest, process and catalog data. Abstractions helped them build at a much faster pace and get it to their customers faster. **Time to market**
- The ingestion platform standardized and created conventions for how data is ingested, transformed and stored on the cluster, allowing the platform users to on-board at much faster rate. **Time to value**
- CDAP's native support for incremental processing, reprocessing, tracking metadata, workflow, retention, snapshotting, monitoring, and reporting expedited the efforts to get a system to their customers. **Time to market**
- CDAP was installed in 8 clusters with 100s of nodes. **Enterprise scale**
- Using Cask Tracker, Data Lake users were able to locate Datasets faster and had faster access to metadata, data lineage, and data provenance. This allowed them to efficiently utilize their clusters and also aided them in data governance, auditability, and improving the data quality of Datasets

Case Study

High Volume Streaming Analytics



Building a high speed, high volume streaming analytics solutions with exactly-once semantics is complex, resource intensive, and hard to maintain and enhance. This use-case required data collection from web logs, mobile activity logs and CRM data, in real-time and batch. The data collected was then organized into customer hierarchies and modeled to deliver targeted ad campaigns and marketing promotion campaigns. It also had to provide advanced analytics for tracking the campaigns in real-time. This application had to support the following:

- Support processing of ~38 billion transactions per day in real-time
- Categorizing customer activity into buckets and hierarchies
- Generating unique counts in real-time, to understand audience reach, tracking behavior trend, and the like
- Generate hourly, daily, monthly, and yearly reports on multiple dimensions
- Provide unique stat count on an hourly basis rather than weekly
- Reprocessing data without side effects due to bug fixes and new features
- Exactly-once processing semantics for reliable processing
- Processing data both in real-time and batch

Outcome

- CDAP's abstraction and its real-time program simplified building this application and in getting it to market faster
- The team replaced a MapReduce based batch system to a real-time system, delivering insights every minute instead of days
- CDAP's exactly-once and transactional semantics provided high-degree of data consistency during failure scenarios making it easy to debug and reason the state of data
- CDAP's Standalone and Testing frameworks allowed the developers to build this application efficiently. No distributed components were required to run functional tests

Case Study

Information Security Reporting

In a large enterprise environment there are traditional information sources that house a great deal of data. There is a constant need to load data into Hadoop clusters to perform complex joins, filtering, transformations, and report generation. Moving data to Hadoop is cost-effective as there is the need to run many complex, ad-hoc queries that would otherwise require expensive execution on traditional data storage and querying technologies.

This customer had attempted to build a reliable, repeatable data pipeline for generating reports across all network devices which access resources. Data was aggregated into five different Microsoft SQL Servers. Aggregated data was then periodically (once-a-day) staged into a secured (KerberOS) Hadoop cluster. Upon loading the data into the staged area, transformations (rename fields, change type of field, project fields) were performed to create new datasets. The data was registered within Hive to run Hive SQL queries for any ad-hoc investigation. Once all the data was in final independent datasets, the next job was kicked off that joins the data from across all five tables to create a new uber table to provide a 360 degree view for all network devices. This table is then used to generate a report that is part of another job. Following are the challenges the customer faced:

- Ensuring that the reports aligned to day-to-day boundaries
- Restarting the failed jobs from the last point where they had failed (had to reconfigure pipelines to restart failed jobs)
- Adding new sources required a lot of setup and development time
- Inability to test the pipeline before it was deployed — this led to inefficient utilization of the cluster as all the testing was performed on the cluster
- They had to cobble together a set of loosely federated technologies -- Sqoop, Oozie, MapReduce, Spark, Hive, and Bash Scripts

Outcome

- The in-house Java developers with limited knowledge of Hadoop built and ran the complex pipelines at scale within two weeks after four hours of training
- The visual interface enabled the team to build, test, debug, deploy, run, and view pipelines during operations
- The new process reduced system complexity dramatically, which simplified pipeline management
- The development experience was improved by reducing inappropriate cluster utilization
- Transforms were performed in-flight with error record handling
- Tracking tools made it easy to rerun the process from any point of failure

Case Study

Real-time brand and marketing campaign monitoring

Enterprises use Twitter to know when people are talking about their brand and understanding sentiment toward their new marketing campaigns. Real-time monitoring capabilities on Twitter allows them to keep a close eye on the results of marketing efforts.

Developing a real-time pipeline that ingests the full Twitter stream, then cleanses, transforms, and performs sentiment and multi-dimensional analysis of the Tweets that were related to the campaign delivers a valuable real-time decision making platform. The aggregated data is exposed through REST APIs to an internal tool for visualization, making consumption of the output easier.

The pipeline is built using Storm, HBase, MySQL, and JBoss. Storm is used to ingest and process the stream of Tweets. The Tweets are analyzed using NLP algorithms to determine sentiment. They are aggregated on multiple dimensions like number of re-tweets and attitude (positive, negative, or neutral). The aggregations are stored in HBase. Periodically (twice-a-day) the data from HBase is moved into MySQL. JBoss exposed REST APIs for accessing the data in MySQL.

The goal of this use-case was to reduce the overall complexity of the pipeline, moving away from maintaining a separate cluster for processing the real-time Twitter stream, integrate NLP scoring algorithms for sentiment analysis, and exposing the aggregated data from HBase with lower latency, thereby reducing the latency between the data being available in HBase to delivery via REST API. The result: an easy to build, deploy, and manage real-time pipeline with better operational insights.

Outcome

- Cask Hydrator pipeline for processing full twitter stream was built in 2 weeks
- Cleansing, Transforming, Analyzing, and Aggregating tweets at about 6K/sec in-flight
- Consolidated infrastructure into a single Hadoop cluster
- Java Developers were able to build the pipeline and plugin with a smaller learning curve
- CDAP Service on OLAP Cube reduced the expensive data movement and reduced the latency between the aggregation being generated to the results being exposed through REST APIs, allowing them to make better decisions faster
- CDAP and Cask Hydrator seamlessly and transparently provided easy operational insights through custom dashboards and aggregated logs for debugging

Try it now!

Download CDAP Standalone to build your application:
cask.co/downloads

CDAP Standalone Docker through Kitematic:
cask.co/cdap-docker

Use Cloudera Manager CSD to install on cluster:
cask.co/cdap-cm

Use Ambari to install on HDP cluster:
cask.co/cdap-ambari

For more information please visit:
cask.co/products/hydrator

650-469-DATA

Sales: sales@cask.co

Support: support@cask.co

General: info@cask.co

TAP IN @ CASK.CO

150 Grant Ave, Palo Alto, CA 94306