

[] CASK DAP

The Cask Data Application Platform is an application server for modern data architecture. It provides a robust platform for the development, deployment and management of data applications and the management of data.

Running either standalone, deployed within the enterprise or pushed to the cloud, DAP comes with numerous standard services. Out-of-the-box, these include transaction management, datasets, naming, messaging (data availability notification, data modification, service communication), QoS, performance, scalability, security and integration [DOCNOTE: FIXME! What's this? Check entire list].

DAP's container model allows for the integration of a variety of processing paradigms with these standard services. It provides a common environment, the abstraction of a unified API, lifecycle management of applications and a programming model for data applications and their data.

You can run simple Map-reduce applications; complete an ETL (extract, transform, load) pipeline; build complex, enterprise-scale data-intensive applications. Developers can build and test their applications end-to-end using the exact same code.

How It Works

- Datasets (and Transactions using Cask Tephra)
- Metrics, Logging and Monitoring
- Realtime (Flows)
- Batch Processing using Map-reduce Jobs and Workflow Schedules
- Ad-Hoc SQL Queries
- Stored Procedures
- Different Runtimes: Single-node, Sandbox and Enterprise
- Management Dashboard

[insert architecture diagram]

Getting Started

You can get started with Cask DAPw by building directly from the latest source code:

```
git clone https://github.com/cask/dap.git
cd dap
mvn clean package
```

After the build completes, you will have a distribution of the Cask DAP Single-node SDK under the `dap-distribution/target/` directory.

To build for installation on a Hadoop Cluster, see the [Cask Building and Installation Guide](#) or the copy included in the source distribution in `/docs/developer-guide/source/install.rst`.

Take the `dap-<version>.tar.gz` file and unzip it into a suitable location.

Step 1: Installation and Startup

Start the Cask DAP from a command line in the SDK directory:

Step 2: The Dashboard

When you first open the Dashboard, you'll be greeted by:

Step 3: Inject Data

Click on the Flow name (LogAnalyticsFlow),

Step 4: Query Procedure

Now let's see the results of our event.

Step 5: Modify the Code

Now let's try something different.

Step 6: Redeploy and Restart

We now need to stop the existing Application.

Step 7: Checkout the Results

Click on the name of the Procedure ...

Step 8: Stop the Server

To stop the Server...

Where to Go Next

Now that you've had a look at Cask DAP, take a look at:

- Examples, located in the `/examples` directory of the Cask DAP SDK;
- Selected Examples (demonstrating basic features of the Cask DAP) are located on-line, at [<http://cask.com/developers/docs/dap/current/en/examples.html](http://cask.com/developers/docs/dap/current/en/examples.html)
- Developer Guides, located in the source distribution in `/docs/developer-guide/source` or online at <http://cask.com/developers/docs/dap/current/en/index.html>;

Developer Guides:

- Introduction
- Quick Start
- Examples
- Programming Guide
- Advanced Features
- Querying Datasets with SQL
- Testing and Debugging
- Security
- Operations
- HTTP REST API
- Javadocs
- Release Notes
- FAQ

How to Contribute

Interested in helping to improve Cask DAP? We welcome all contributions, whether in filing detailed bug reports, submitting pull requests for code changes and improvements, or by asking questions and assisting others on the mailing list.

Bug Reports & Feature Requests

Bugs and tasks are tracked in a public JIRA issue tracker. Details on access will be forthcoming.

Pull Requests

We have a simple pull-based development model with a consensus-building phase, similar to Apache's voting process. If you'd like to help make Cask DAP better by adding new features, enhancing existing features, or fixing bugs, here's how to do it:

1. If you are planning a large change or contribution, discuss your plans on the `cask-dap-dev` mailing list first. This will help us understand your needs and best guide your solution in a way that fits the project.
2. Fork Cask DAP into your own GitHub repository.
3. Create a topic branch with an appropriate name.
4. Work on the code to your heart's content.
5. Once you're satisfied, create a pull request from your GitHub repo (it's helpful if you fill in all of the description fields).
6. After we review and accept your request, we'll commit your code to the cask/dap repository.

Thanks for helping to improve Cask DAP!

Mailing List

Cask DAP User Group and Development Discussions: cask-dap-dev@googlegroups.com

License and Trademarks

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Cask, Cask DAP and Cask Data Application Platform are trademarks of Cask, Inc. All rights reserved.

Apache, Apache HBase, and HBase are trademarks of The Apache Software Foundation. Used with permission. No endorsement by The Apache Software Foundation is implied by the use of these marks.