
Tessera

Tessera is a front-end interface for the Graphite metrics system, which provides a large selection of presentations, layout, and interactivity options for building dashboards.

The biggest key differences between Tessera and other frontends are the separation of queries from presentations, and the ability to apply arbitrary transformations to the presentations & queries, allowing for a large degree of interactivity. Tessera is initially focused on information presentation - it does not NOT address the areas of metric discovery or query composition (although it may in the future).

Overview

Tessera consists of a small python webserver written with Flask with a SQL backing store. The server manages storing and searching for dashboards, managing tags for organization, serving the basic UI assets, and providing a ReST API for the front-end to use.

Dashboards are big lumps of JSON data describing the queries, presentations, and layout, wrapped in a small shell of SQL metadata. Most of the heavy lifting is done by the javascript front-end, which is responsible for all rendering, data fetching from graphite, and editing of dashboards.

Quick Start

From Docker

If you're familiar with docker and already have a Graphite instance running, you can boot up an instance of tessera with the demo dashboards preloaded and easily point it at your existing Graphite installation using the [aalpern/tessera-simple](#) image. If you don't have a Graphite installation handy, a Docker image like [nickstenning/graphite](#) can get that up and running quickly.

```
1 docker run -P -e GRAPHITE_URL=http://graphite.host -it aalpern/tessera-simple
```

From PyPi

Tessera can be installed easily from PyPi with [pip](#) or [easy_install](#). This method of installation is only recommended for casual use.

```
1 pip install tessera
```

After installation, create a `config.py` script for your local settings.

Example `config.py`:

```
1 GRAPHITE_URL="http://graphite.example.com"
2 SECRET_KEY="adf71812-9d57-88d3-dfe8-1e9860d2b7ab"
```

Initialize the database:

```
1 TESSERA_CONFIG=/path/to/config.py tessera-init
```

Then launch the service:

```
1 TESSERA_CONFIG=/path/to/config.py tessera
```

The service will be available on localhost:5000 by default, with the demo dashboards loaded.

From Source

The TL;DR version:

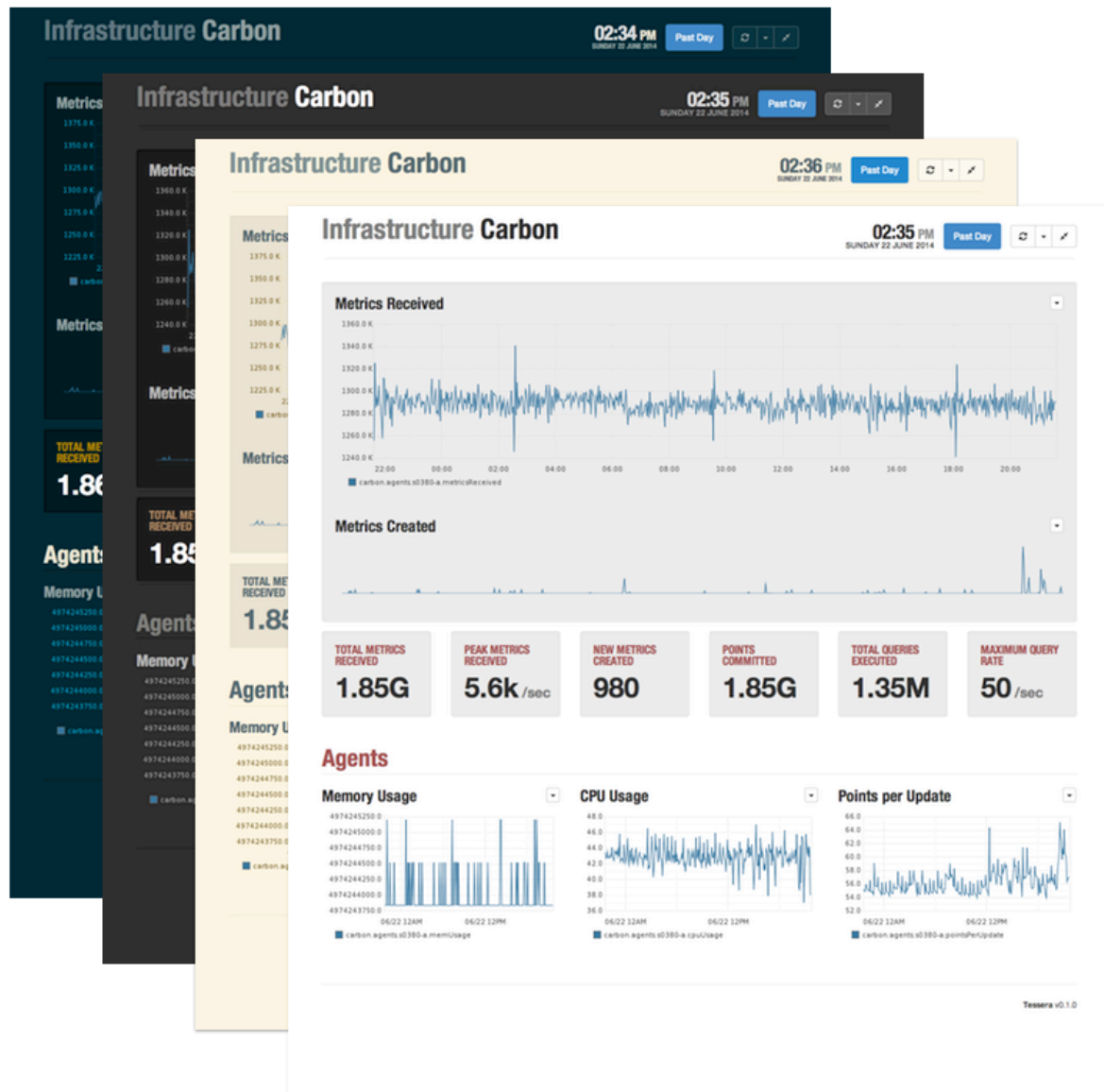
```
1 ./script/setup
2 ./script/server &
3 cd tessera-server
4 inv json.import '../demo/*'
```

Documentation

Documentation is hosted on GitHub Pages at tessera-metrics.github.io/tessera/docs.

- Getting Started for getting started and running from source and importing the demo dashboards.
- Tutorial for creating the most basic possible dashboard.
- API endpoints and data format for accessing Tessera via the HTTP REST API
- Extending Tessera, for adding new dashboard items, transformations, and actions.

Some additional writeup of why Tessera exists can found on the introductory blog post that was originally posted on Urban Airship's company blog - [Introducing Tessera, a Graphite Frontend](#).



Copyright & License

Copyright © 2014, Urban Airship and Contributors

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.

Third-party software libraries included with this project are distributed under their respective licenses.