intake_splunk Documentation

Release 0.1.0+0.g6f2d28f.dirty

Joseph Crail

Nov 22, 2018

Contents:

This package accesses tabular data in Splunk, and can be used by Intake to load that into pandas dataframes.

CHAPTER 1

Quickstart

intake-splunk provides quick and easy access to tabular data stored in Apache Splunk

This plugin reads splunk query results without random access: there is only ever a single partition.

1.1 Installation

To use this plugin for intake, install with the following command:

```
conda install -c intake intake-splunk
```

1.2 Usage

1.2.1 Ad-hoc

After installation, the function intake.open_splunk will become available. It can be used to execute queries on the splunk server, and download the results as a list of dictionaries.

Three parameters are of interest when defining a data source:

- query: the query to execute, using Splunk's **‘Query Syntax‘_**

1.2.2 Creating Catalog Entries

To use, catalog entries must specify driver: splunk.

1.2.3 Using a Catalog

CHAPTER 2

API Reference

intake_splunk.core.SplunkSource(query, url, auth)	Execute a query on Splunk

class intake_splunk.core.**SplunkSource**(*query*, *url*, *auth*, *chunksize=5000*, *metadata=None*)

Execute a query on Splunk

Parameters

query [str] String to pass to Splunk for execution. If it does not start with “|” or “search”, “search” will be prepended.

url [str] Endpoint on which to reach splunk, including protocol and port.

auth [(str, str) or str] Username/password to authenticate by.

chunksize [int]

Attributes

cache_dirs

datashape

description

hvplot Returns a hvPlot object to provide a high-level plotting API.

plot Returns a hvPlot object to provide a high-level plotting API.

plots List custom associated quick-plots

Methods

`close`()	Close open resources corresponding to this data source.
`discover`()	Open resource and populate the source attributes.
`read`()	Load entire dataset into a container and return it
`read_chunked`()	Return iterator over container fragments of data source
`read_partition`(i)	Return a (offset_tuple, container) corresponding to i-th partition.
`to_dask`()	Return a dask container for this data source
`to_spark`()	Provide an equivalent data object in Apache Spark
`yaml`([with_plugin])	Return YAML representation of this data-source

set_cache_dir	

`to_dask``()`

Return a dask container for this data source

CHAPTER 3

Indices and tables

- genindex
- modindex
- search

Index