Venue DataStore Client API

Robert Olson

25th July 2003

olson@mcs.anl.gov

Abstract

The venue datastore provides a mechanism for the venue to store data on behalf of the venue's users. The client API described here provides easy access for the clients of a venue to access the data therein.

Contents

1 Introduction

2 `AccessGrid.DataStoreClient` — Client API to the Venue DataStore

`GetVenueDataStore`(*venueURL*)

Return a `DataStoreClient` instance representing the default datastore on the venue with url *venueURL*.

class `DataStoreClient`(*uploadURL*, *datastoreURL*)

A `DataStoreClient` instance provides access to the datastore represented by *uploadURL* and *datastoreURL*.

Clients typically will use the module function `GetVenueDataStore` to return a `DataStoreClient` instance for the given venue URL.

class `DataStoreShell`(*datastoreClient*)

A `DataStoreShell` instance provides a command-line interface to the files in a datastore.

2.1 DataStoreClient Objects

The `DataStoreClient` class defines the following methods:

`LoadData``()`
Reload the client's cache of venue data information.

`QueryMatchingFiles``(`*pattern*`)`
Return a list of filenames in the venue that match the unix-style file glob *pattern*.

`QueryMatchingFilesMultiple``(`*patternList*`)`
Return a list of filenames in the venue that match any of the unix-style file globs in *patternList*.

`GetFileData``(`*filename*`)`
Return the metadata in the cache for file *filename*. The metadata is returned as a Python dictionary. Keys of interest in this dictionary include

name Name of the file

size Size of the file, in bytes

checksum MD5 checksum of the file

uri DataStore URL for the file.

`Download``(`*filename, localFile*`)`
Download *filename* to the local file *localFile*.

`Upload``(`*localFile*`)`
Upload local file *localFile* to the datastore. The uploaded file will be named with the basename of *localFile*. If that file already exists, an exception will be raised.

`RemoveFile``(`*file*`)`
Remove the file named *file* from the venue datastore.

`OpenFile``(`*file, mode*`)`
Open a venue file named *file*.

If *mode* is "r", the file will be opened for reading. In this implementation, the file will be downloaded and a filehandle to that local file returned.

If *mode* is "w", the file will be opened for writing. In this implemetnation, a file handle to a new local file will be returned. When the file handle is closed, the file will be uploaded to the venue.

2.2 DataStoreShell Objects

The `DataStoreShell` class defines the following methods:

`run``()`
Start the command processor.

Module Index

A

Index