Simplifying Client Interactions with SMI-S using PyWBEM

Using SMI-S 1.8.0 Mock Servers

Michael Walker
Background

- Pywbem and pywbemtools are python packages that provide access to WBEM Servers.
  - Pywbem provides a python API for accessing WBEM Servers
  - Pywbemtools include a Command Line interface for accessing WBEM Servers via command prompts
- They also provide a mechanism for mocking WBEM Servers, including SMI-S Servers
- Mock servers can be extremely useful for developing applications and tools on your local system
Uses of Mock Servers

- Explore SMI-S 1.8.0 features
  - The mock servers support the WBEM Server Profile and Advanced Metrics for Arrays (both are new in SMI-S 1.8.0)
- Develop Tools to be used with SMI-S Servers
- Develop client software for SMI-S Servers
- As a local test environment for testing tools and applications
Publicly available SMI-S 1.8.0 Mock Servers

- A publicly available GitHub repository can be found here:
  - [https://github.com/FarmerMike252/SMI-S_Mocks](https://github.com/FarmerMike252/SMI-S_Mocks)
- The mock servers were developed to illustrate some of the new features in SMI-S 1.8.0 (which is now an ISO Standard)
  - The link to the ISO Standard is: [iso.org/search.html?q=24775](https://iso.org/search.html?q=24775)
- A set of “Quick Start Guides” are also in the repository
  - These guides are designed to help IT users find useful information (e.g., Hardware, product, software, capacity and performance information) in an SMI-S server.
  - The guides are pywbemcli (part of pywbemtools) command line scripts to access WBEM servers (the scripts work on the Array mock server).
  - These guides are illustrated in video’s on the SNIA pywbem page: [https://www.snia.org/pywbem](https://www.snia.org/pywbem)
  - The Quick Start Guides folder also contains the presentation used in the installation and setup video (for installing python, virtual environments, pywbem and pywbem tools).
Github Mocks

FarmerMike252 / SMI-S_Mocks

- master
- 2 branches
- 0 tags

About
A prototype of a repo of SMI-S Mocks using py2bem

Readme

Releases
No releases published
Create a new release

Packages
No packages published
Setup Presentation

FarmerMike252 / SMI-S_Mocks

Fixed the Quick Start Guide for Capacity (from a duplicate of Product).

d19aa61 on Feb 18

PywbemPIInstallationAndSetup.pdf
Updated based on work done on the videos done for SNIA.

QSG-ArrayCapacity.pdf
Updated based on work done on the videos done for SNIA.

QSG-ArrayHardware.pdf
Fixed the Quick Start Guide for Capacity (from a duplicate).

QSG-ArrayPerformance.pdf
Updated based on work done on the videos done for SNIA.

QSG-ArrayProduct.pdf
Updated based on work done on the videos done for SNIA.
The ISO Standard

27 RESULTS FOUND (1 MS)

ISO/IEC 24775-8:2021 INFORMATION TECHNOLOGY — STORAGE MANAGEMENT — PART 8: MEDIA LIBRARIES

This version of the specification models various details of the following objects of the media library for monitoring. Library Drives Changer Devices Slots IO Slots SCSI Interfaces and SCSI and FC Target Ports Physical Tapes Physical Package Magazines In general, a CIM client can monitor the health and status of ...

ISO/IEC 24775-7:2021 INFORMATION TECHNOLOGY — STORAGE MANAGEMENT — PART 7: HOST ELEMENTS

The host-base storage portion of the Storage Management Technical Specification defines management profiles for autonomous, component and abstract profiles for management of host-based storage devices. The
SMI-S QUICK START GUIDES

Twenty-year SNIA veteran Mike Walker has created a series of videos titled "SMI-S Quick Start Guides" that provides developers using the SMI-S storage management specification instructions on how to mine useful information in an SMI-S server using the Python-based PyWBEM open source tool. Be sure to check this site often for more installments.
Using the mock servers

- Feel free to use the mock servers in the Github repository
- Download or clone the repository
- Each mock is made up of three files:
  - A leaflist xml file – which defines the classes (mofs) to be loaded into the mock server
  - An instance mof file – that contains instances to be loaded in the mock server
  - A mockload python program – that performs the loading of the classes and instances

These files need to be copied into your virtual environment working directory.
Please take a moment to rate this session.

Your feedback is important to us.