

Python ADDE Project

David Santek, Becky Schaffer, Jon Beavers,
Tommy Jasmin, Jerry Robaidek, Dave Hoese

2019 McIDAS Users' Group Meeting
17 September 2019

Overview

- A cornerstone of McIDAS is the modular and extensible Abstract Data Distribution Environment (ADDE):
 - Servers for both local and remote data access
 - All satellite data in SSEC's Satellite Data Services are accessible via ADDE
 - Used throughout the McIDAS community by both McIDAS-X and McIDAS-V

We expect the ADDE data delivery protocol to be important for many years to come.

Overview

- ADDE servers primarily written at SSEC, with Unidata, EUMETSAT, and AEMET also contributing.
- More than 20 servers, including:
 - Satellite: GOES ABI, Himawari AHI, MODIS, AVHRR, VIIRS, Meteosat, COMS
 - Grid: GRIB, McIDAS Grid
 - Other: netCDF, HDF, GeoTIFF, Text

Objectives

- ADDE infrastructure refactoring:
 - Move ADDE server development into a **modern programming environment**.
- Ensure a future for ADDE for new satellites and other image data:
 - Motivated by the fact that **current SSEC ADDE developers are semi-retired**
 - **ADDE continues to be an important data delivery mechanism** for SSEC, Unidata, and many of the current McIDAS-X and McIDAS-V users.
- Outcome will be Python library and functional ADDE server that will:
 - Interface and be packaged with McIDAS-X and -V
 - Work on Linux, Windows, and Mac
 - Enable a programmer or scientist with 1 year of Python experience to create their own ADDE server without knowledge of underlying McIDAS Fortran and C code.

pyADDE Status

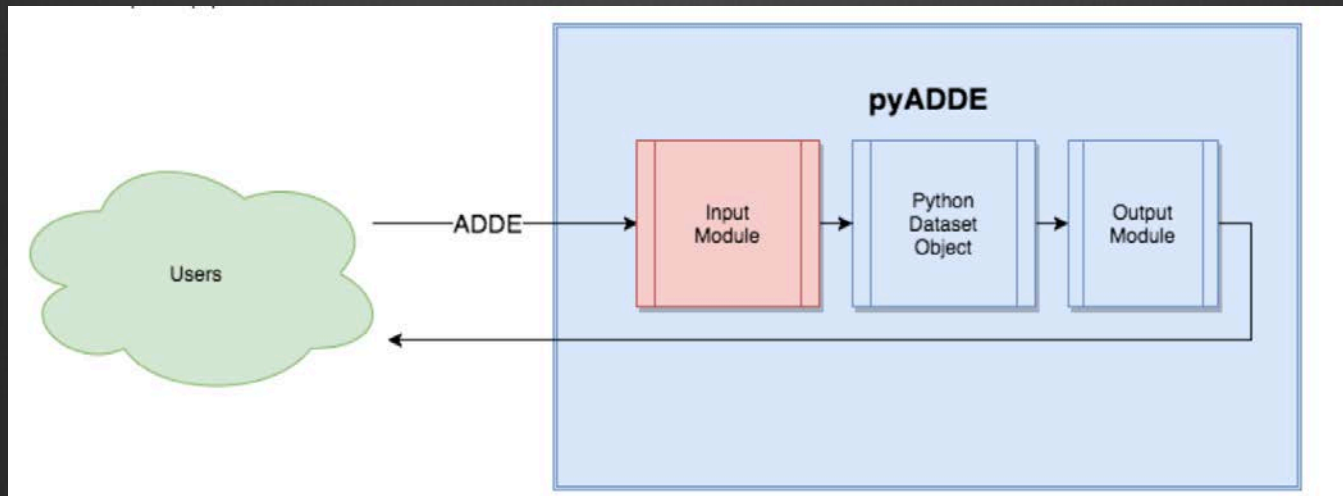
Phase 1: Study - Complete

- Investigate the current ADDE protocol and implementation:
 - Evaluate the feasibility, and
 - Estimate the effort to enable the use of Python to write ADDE servers.

Yes, it is programmatically possible to develop servers independent of McIDAS-X

Python ADDE Servers

- Write a new server without McIDAS-X knowledge
- Only *Input Module* will need to be written:
 - Read native file
 - Compute lat/lon for each n^{th} point
 - Reshape into Python Dataset Object
 - pyADDE does the rest



pyADDE Status

Phase 2: Prototype - Complete

- A Python directory server was written
- A Python data server was written
 - A basic server to read a PNG file and return the data along with McIDAS-necessary metadata (directory and navigation block)
- The **ADDE protocol tests were successful** for both local and remote instances of the Python directory and data servers on Linux
 - **Note: This uncovered an unexpected issue in the transmission protocol not seen with typical McIDAS-X ADDE servers.** Python seemingly does not block by default on stdout going to a socket (remote serving), whereas going to a pipe works fine (local serving). Issue resolved by calling `os.set_blocking(1, True)`.

pyADDE Status

Phase 3: Refactoring – Beginning Oct. 1

- Rewrite low-level C and Fortran routines into Python:
 - Python ADDE servers and library (pyADDE) is a stand-alone package
 - ADDE servers can be written without in-depth knowledge of McIDAS-X
- Investigate interfacing Python ADDE servers to the SatPy library:
 - Access to a variety of geostationary and polar satellite data, through a common API
 - New Python ADDE servers could be written without a deep understanding of data formats used by the different satellites
- Write a Python version of the McIDAS-X ABI server
 - Document and test with Windows and Linux

Summary

- Advantages:
 - **Evolution** rather than revolution
 - May make use of McIDAS-X Fortran and C library functions in the short-term
 - **Extensions** more easily implemented
 - Return other formats (e.g., netCDF), not only McIDAS-X Area
 - Perhaps additional geographic projections (using **proj.4**)
- Status:
 - **Prototype server** being developed
 - Plan is to have first version of **pyADDE** available by **mid-2020**