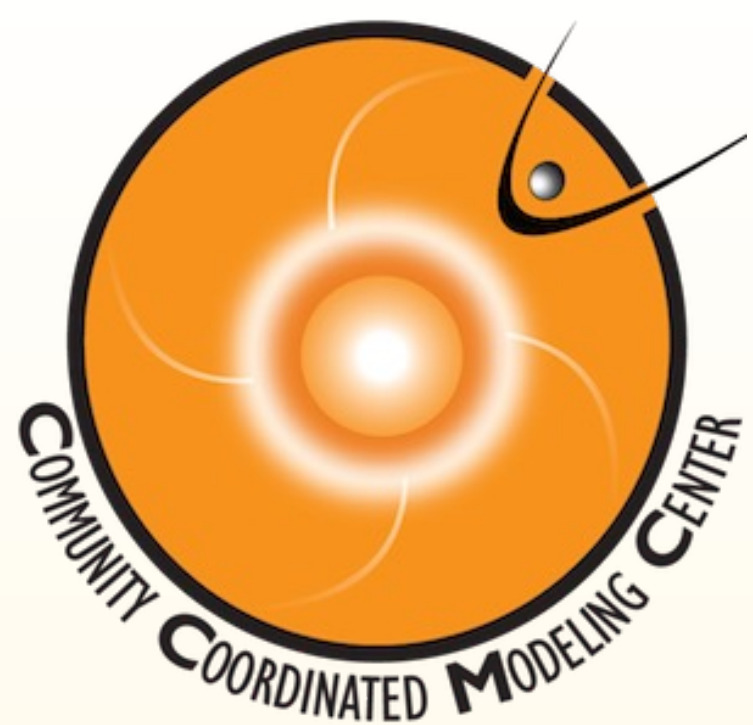




Interoperability of tools at the CCMC

Darren DeZeeuw¹, Lutz Rastaetter¹, Masha Kuznetsova¹, Chiu Wiegand¹, Rick Mullinix¹,
Josh Pettit¹, Jack Wang¹, Katherine Garcia-Sage¹, Oliver Gerland²

¹Community Coordinated Modeling Center, NASA Goddard Space Flight Center, Greenbelt, MD, USA

²Ensemble Government Services LLC, Hyattsville, MD, USA

Abstract:

The CCMC has a diverse set of tools in many languages that support utilization of simulation outputs accessible through CCMC interactive archives. Some model output post-processing and analysis tools are delivered to CCMC by the community. One such model output post-processing/utilization tool developed by the CCMC is the open source Kamodo package. Kamodo is developed primarily in Python (with some C) and has already established strong interoperability with other Python libraries inside PyHC and out. While that interoperability is important, interoperability with other languages and tools is as important. Many models are written in Fortran or C, and their ability to pull in data from a Python tool or export directly to other analysis software in a different language will greatly increase scientific productivity. Interoperability within Python is important, but broader interoperability is just as important.

CCMC Tools:

The CCMC has a 20+ year history of supporting the heliophysics community with a portal to modern space science research models as well as a variety of online visualization and analysis tools. Nearly 100 models are available to run through one of these three systems:

- Runs-on-Request
- Continuous Run
- Instant Run



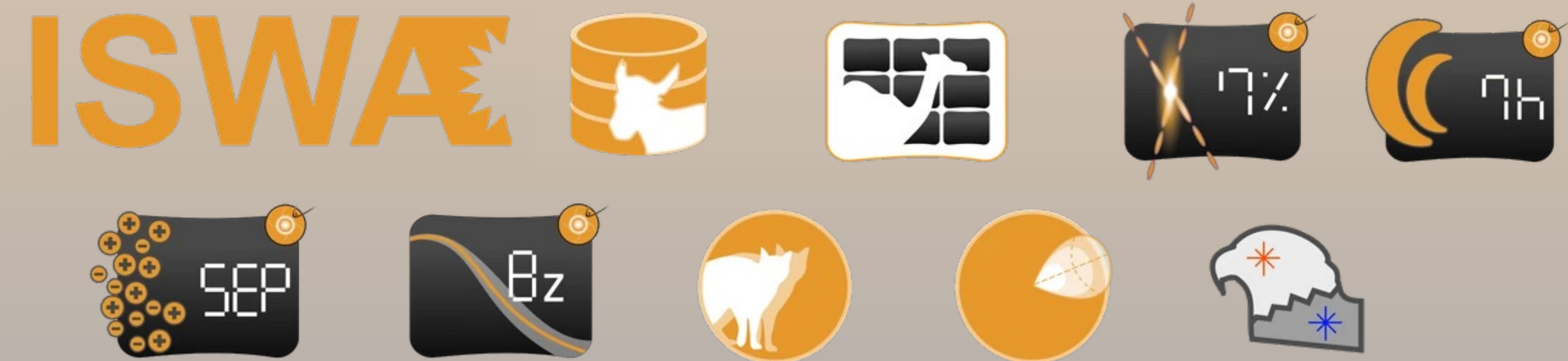
Model run output can be analyzed and visualized through:

- Online web visualization
- Movies-on-Request
- VMR Tools for real event analysis and data/model comparisons
- Kamodo (open-source Python toolkit)
- Kameleon (nearing end of life)
- OpenSpace (a partnership with the OpenSpace team to bring model output to planetarium shows)



In addition, several other space weather tools are available that work with model output, available observational data, and user contributed information, such as:

- ISWA (serving CCMC real-time/continuous model outputs and observational data)
- DONKI (comprehensive online database of space weather events for the community)
- CAMEL (framework for comparing space weather and space science model outputs with observational datasets.
- Flare Scoreboard (validation for predicting solar flare events)
- CME Scoreboard (validation for coronal mass ejections arrival time at Earth)
- SEP Scoreboard (validation for predicting solar energetic particle events)
- IMF Bz Scoreboard (validation for interplanetary magnetic field forecasts at L1)
- StereoCAT (enable space weather forecasters and researchers to quickly calculate CME kinematic properties)
- SWPC CAT Web (calculate CME kinematic properties the SWPC CME analysis tool)
- EEGGL (uses observational data to specify input parameters for the Gibson-Low flux rope model)



CCMC Interoperability:

While each of these tools is valuable, they become more powerful when they work together. The tools at the CCMC are a blend of:

- web interface
- web API
- HAPI served data
- bash scripts
- IDL code
- Fortran and C/C++ code
- Python

These all work with model output in numerous data formats.

Many CCMC workflows depend on interoperability. One example is the continuous run of the SWMF which feeds into automated scripts that generate graphics fed into several ISWA cygnets and data updates on the HAPI server.

Interoperability is crucial for these systems to function properly. One challenging issue in this is language interoperability. Tools in Fortran or C/C++ were not too challenging to get to talk to each other, but with the rise of more tools in Python this has led to roadblocks. The CCMC is working to ensure that new tools developed in Python can be called from old or new tools in Fortran for example.

Kamodo Interoperability:

Kamodo is a new open-source Python toolkit developed by the CCMC. Kamodo is primarily a Python package with some C code, with work ongoing to enable more interlanguage support. However, when focusing primarily on Python, Kamodo has strong interoperability with other Python packages. One of the strengths of Python and open-source is that if somebody else already provides a needed tool, then it is easy to import that package and use it.

Kamodo uses many other Python packages, such as numpy, scipy, and pandas, but it also tightly interoperates with several others, such as spacepy, pysat, astropy, and HAPI. Here is a sample interconnected workflow:

- We have loaded data that is in GSM (geocentric solar magnetic) coordinates from SWMF-GM, and we want to extract values along the Goes12 trajectory and compare the results to observed Bz values in GSE (geocentric solar ecliptic) coordinates.
- The data is transformed into GEO (cartesian geographic) through a call to spacepy,
- Kamodo uses the HAPI server at SSCWeb to get the satellite position in GEO coordinates,
- Kamodo does the interpolation in time and space to extract the values,
- Another call is made to spacepy to transform the GEO data into GSE,
- Kamodo again uses HAPI to get satellite data from CDAWeb,
- The resulting plot is made with data/model comparisons in chosen coordinates.

