Data Rods Explorer 1.0.0

Browse NASA data services for land surface model maps and time series

About DRE

The Data Rods Explorer (DRE) is a web client app that enables users to browse several NASA-hosted observation retrievals and land surface model (LSM) outputs by variable, space and time. The user can obtain a map of a variable at any valid date-time stamp; plot a time series of values available; compare two LSM variables' time series over a mutually-available time period; and plot year-on-year changes in model outputs for a given LSM or retrieval variable. Tabular data and plots of time series can be downloaded, as well as graphic images of the plots.

DRE was developed in 2015-2016 by Dr. Gonzalo Espinoza as part of his PhD work at the University of Texas at Austin, and by Shawn Crawley at Brigham Young University, Provo UT. The development was directed by Dr. David Arctur, and funded by NASA award ACCESS-13-0056, William Teng, PI. Questions on software development should be sent to Gonzalo Espinoza. Initial point of contact for NASA data-related questions is William Teng. All general questions related to data rods should go to gsfc-help-disc@lists.nasa.gov.

Headings below with underscores are links to other pages.

Features and Utilities

This guide provides instructions and examples for using the Data Rods Explorer. There are three main types of usage described here:

- Browsing the NASA data sets
- Working with the DRE user interface tools
- Understanding notifications, causes of data service errors, and how to test the data services directly

NASA References

Links to NASA reference pages and the Giovanni data server.

Keeping Data Rods Explorer Up to Date

DRE is designed to be easily updated in the event of changes in NASA models and variables represented by data rods, as well as in data service request syntax. This section explains how DRE would be maintained.

- Using the NASA Data Rods Variables Info spreadsheet
- Reading and editing the model_config.txt file
- Running the enddates_bounds.py script (cron job)
- Synchronization issues between the NASA CMR metadata and actual Data Rods availability

Developer's Corner: DRE App Process Flow

This part of the guide is intended to help software developers to understand the main events and sequences of program flow. It is very important for future developers to keep this up to date when changing or adding functionality, in addition to versioning in the DRE github repo.

Future enhancements

- GLDAS-1 to be removed 12/31/2016; GLDAS-2 to be expanded
- MERRA-Land model and data layers to be added (by Spring 2017)
- TRMM is now a static product. For RT precipitation, NASA will add GPM (Global Precipitation Measurement) (TBD)
- Show grid map legend (see NASA Giovanni section for work-around)
- DRE allows data rod location selection only by point-and-click with a mouse. A future version should enable data rod location from direct user text entry of lat-long coordinates.

Web App Metadata

Title: Data Rods Explorer
Subtitle: Browse NASA data services for land surface model maps and time series.
Authors: Gonzalo Espinoza, David Arctur, Shawn Crawley
Owners: Gonzalo Espinoza
Source repo: https://github.com/gespinoza/datarodsexplorer

External files used by app:
https://github.com/gespinoza/datarodsexplorer/blob/master/tethysapp/data_rods_explorer/public/data/model_config.txt,
https://apps.hydroshare.org/static/data_rods_explorer/data/dates_and_spatial_range.txt

Cron job expected by the app: Python script enddate_bounds.py, run nightly at 3am ET, which reads model_config.txt, queries NASA CMR, and updates dates_and_spatial_range.txt.

Keywords: NASA, time series, data rods, hydrometeorological, land surface model, data assimilation, LDAS, NLDAS, GLDAS, TRMM, LPRM, AMSRE, GRACE, downward radiation flux, precipitation, rain rate, snow rate, soil moisture, surface runoff, subsurface runoff, baseflow, evapotranspiration, soil temperature

Data Rods Explorer - Features & Utilities

Contents

This guide provides instructions and tutorials for using the **Data Rods Explorer (DRE)**. There are three main types of usage described here: (a) browsing the NASA data sets; (b) working with the DRE user interface tools; (3) understanding notifications, causes of data service errors, and how to test the data services directly.

Browsing the NASA data sets

Starting the Data Rods Explorer

When you first open **Data Rods Explorer,** it reads two text files: (1) the model configuration file (model_config.txt) and (2) a file containing spatial and temporal bounds for each model (dates_and_spatial_range.txt). These are further discussed in Keeping DRE Up To Date.

By default, DRE initially opens showing the first model listed in the model_config.txt file, and with the first variable listed for this model. The default date is the latest date for which data is available with that model, see Figure 1 below. The blue rectangle is the map extent for the selected model in the Product-Model list.

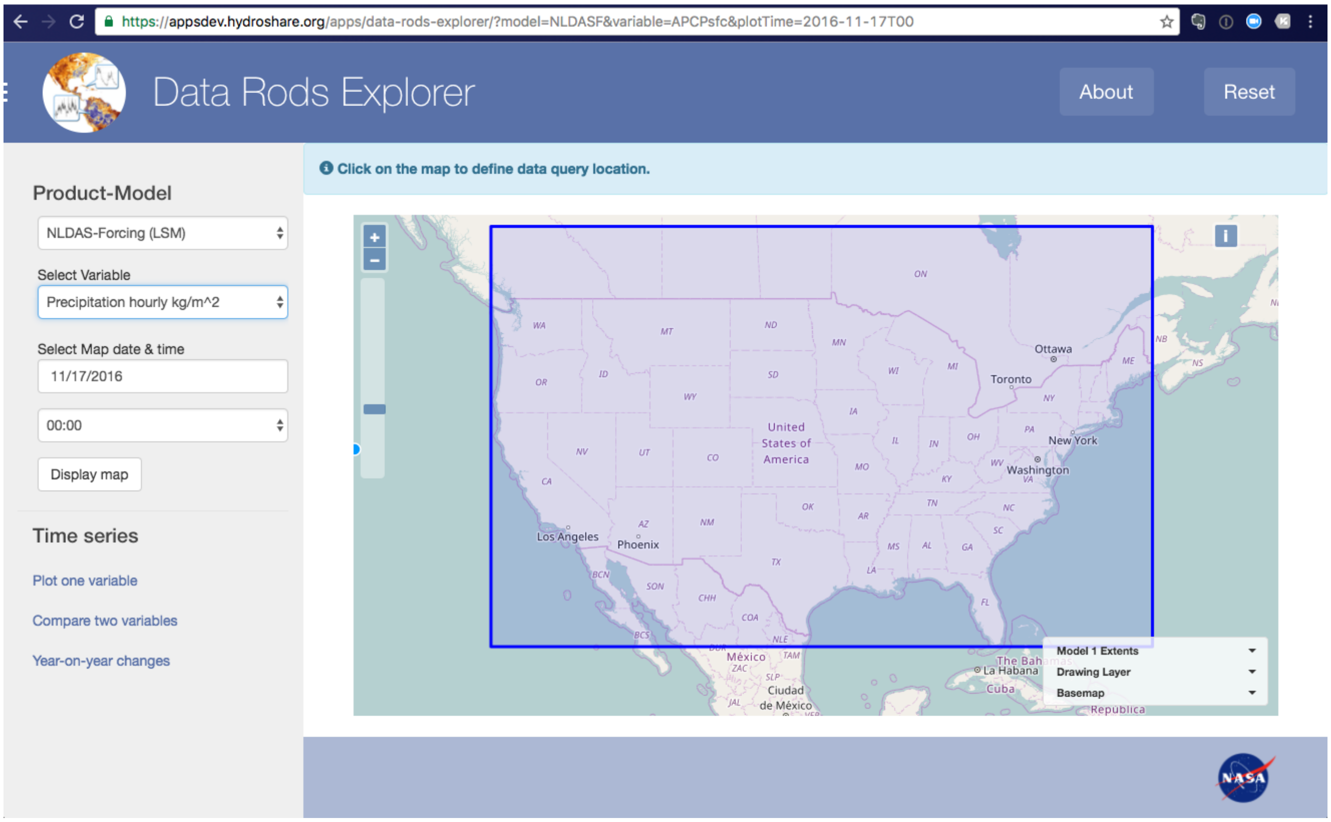


Figure 1. DRE initial startup with defaults

However, DRE can be started to display any given model and variable, date and time. These are all valid ways of starting DRE:

- https://apps.hydroshare.org/apps/data-rods-explorer/
- https://apps.hydroshare.org/apps/data-rods-explorer/?model=GLDAS
- https://apps.hydroshare.org/apps/data-rods-explorer/?model=AMSRED&variable=soil_moisture_c
- https://apps.hydroshare.org/apps/data-rods-explorer/?model=GRACE&variable=sfsm_inst&plotTime=2011-10-02T00

Because DRE populates all pull-down lists and date-time selection fields with valid contents, it will always fill in missing parameters with default values.

Tip: As you start exploring NASA data availability, you may notice that some models are no longer updated (static end date), while others are continually updated but lag the current date by a few days to several weeks. This lag or latency is due to differing workflows to process the incoming spatial data for time series indexing. NLDAS variables are usually available within a few days of the current date, while GLDAS variables may lag by a month, and TRMM by more than that. Due to these variations, a script is run nightly at 3am ET to update an external file listing all the products/models' data availability. You can check this file at any time here: dates_and_spatial_range.txt (apps server)

Selecting and displaying gridded maps of model variables

To pick a model, click on the pulldown field directly below the **Product-Model** heading in the left margin panel, see Figure 2 below. Once the model is chosen, the list of variables is updated to match, see Figure 3. Figure 4 shows picking the date and time.

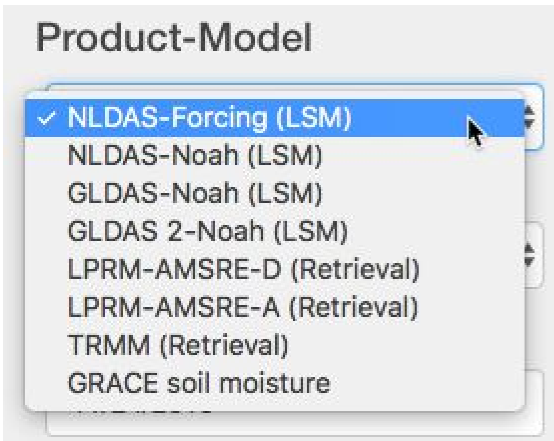


Figure 2. DRE Product-Model List

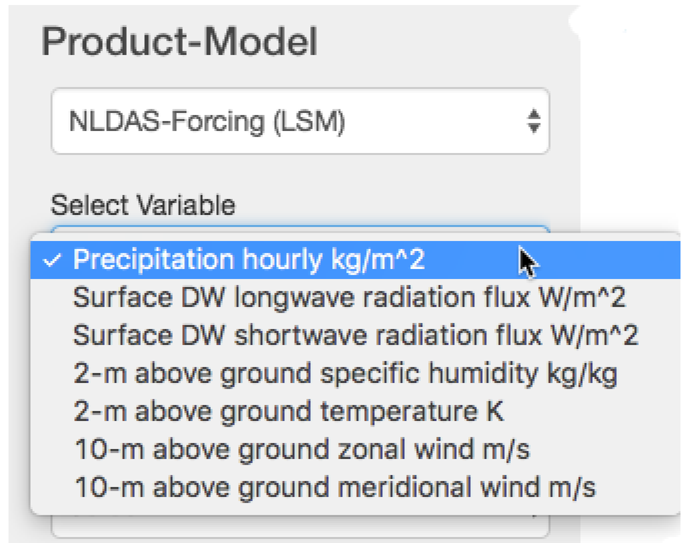
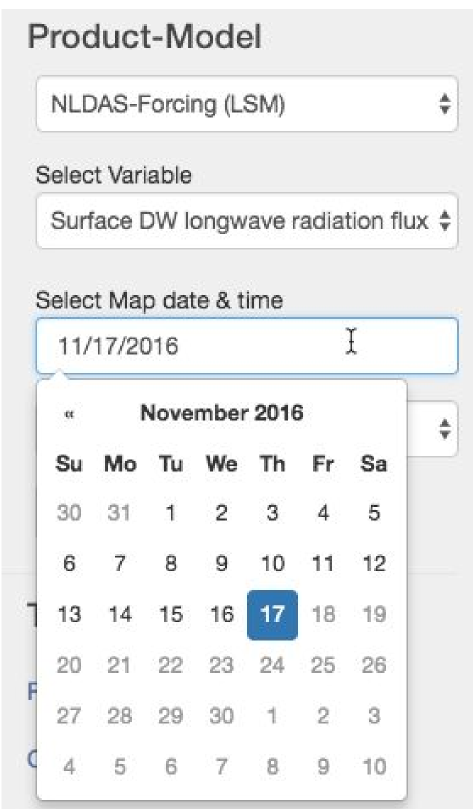


Figure 3. DRE Variable List for NLDAS-Forcings



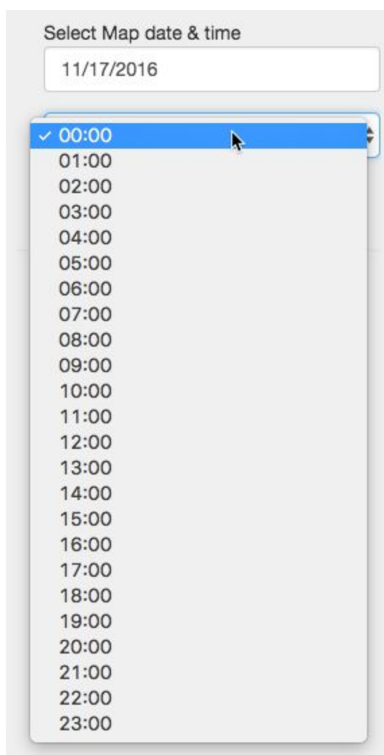


Figure 4. DRE Date and Time Pickers

- Note the dates after Nov 17 in Figure 4 are grayed out; these are fenced off so the user cannot try to pick them for data (this was screen-captured on Nov 22).
- Similarly, the user can't pick a date earlier than the model's begin date as reported in dates_and_spatial_range.txt.
- See [Quickly navigating the date picker]() below for some helpful tips.

Once the desired model, variable, date and time are all chosen, the user would click on the Display Map button. After 10-15 seconds while the NASA Giovanni server constructs the map, it is displayed.

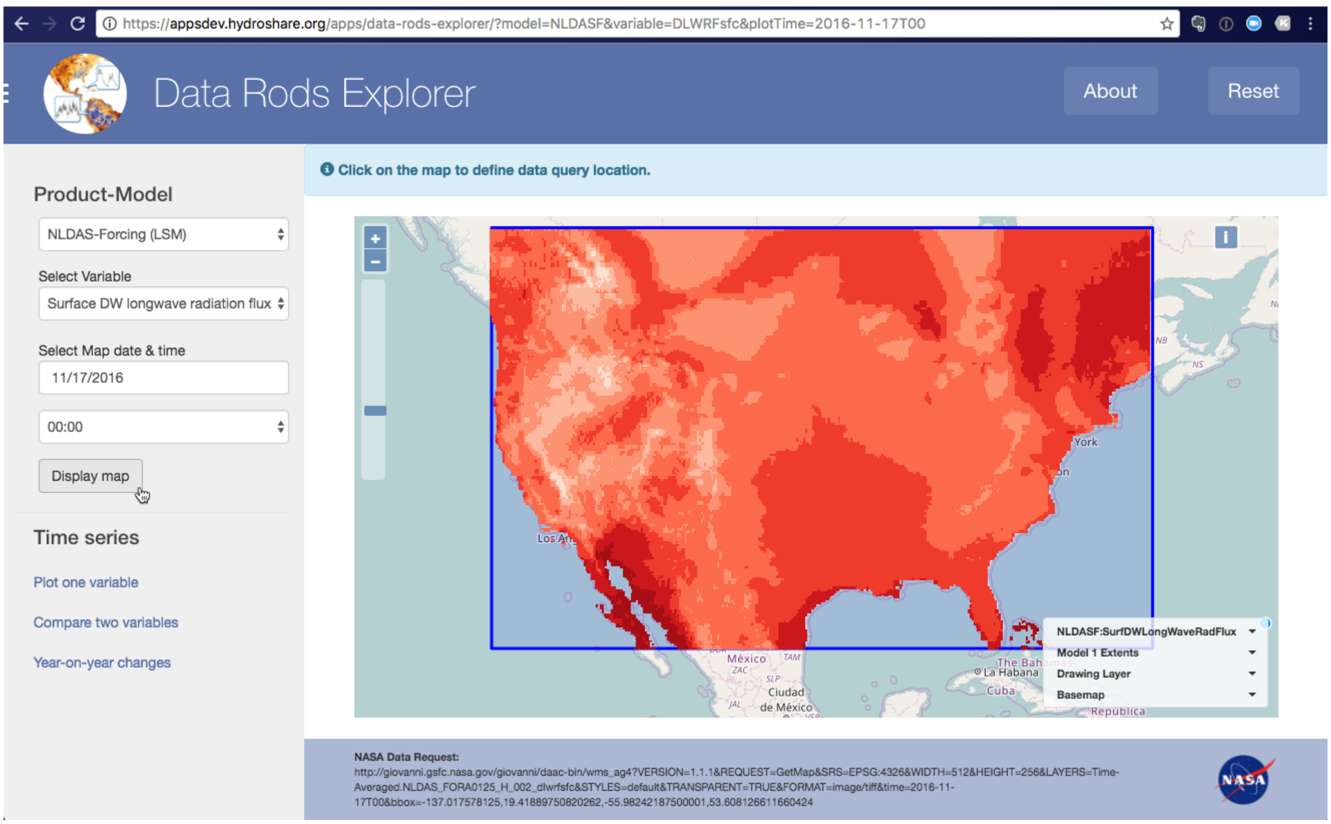


Figure 5. Display Map of NLDAS Surface downward longwave radiation flux

- Notice the URL at bottom of screen; this is the raw data service request sent to NASA server; you can copy/paste it into a browser tab in case you have any questions about results.
- Note the lack of color legend; see discussion and instructions for getting legend from Giovanni.
- You can zoom into a desired area by holding down the shift key while dragging the mouse across the area of interest.
- If you zoom in or out, and want to return to the model bounds, click the mouse once on "**Model 1 Extents**" or "**NLDASF:SurfDWLongWaveRadFlux**" (or other chosen model-variable) in the map control panel in the lower-right corner of the map window.
- You can stack multiple model-variable maps by selecting additional ones; use Reset button to clear map layers.
- You can also hide/show and control transparency of model-variable layers; see Using the map controls panel below.

Plotting time series of one variable

1. Display the map of a model-variable and zoom to area of interest.
2. Click on a point-location for the time series (data rod), and check the dates range.
3. Click Plot one variable, then the Plot button. You can see exact values on the plot by hovering your mouse over the graph lines.
4. Notice the URL at bottom of screen; this is the raw data service request sent to NASA server; you can copy/paste it into a browser tab in case you have any questions about results.

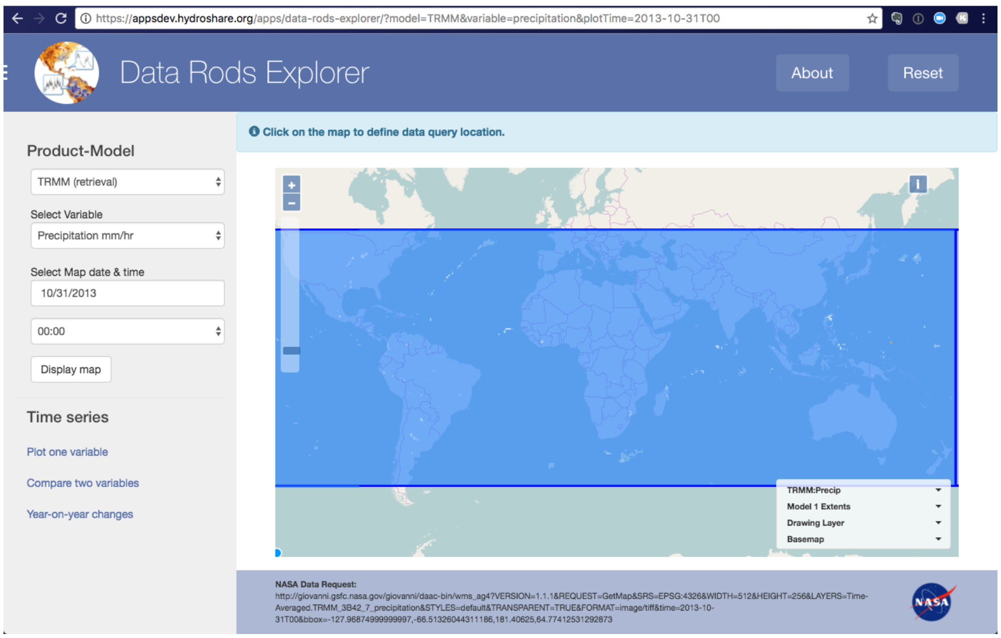


Figure 6. Display map of TRMM Precipitation on Oct 31, 2013

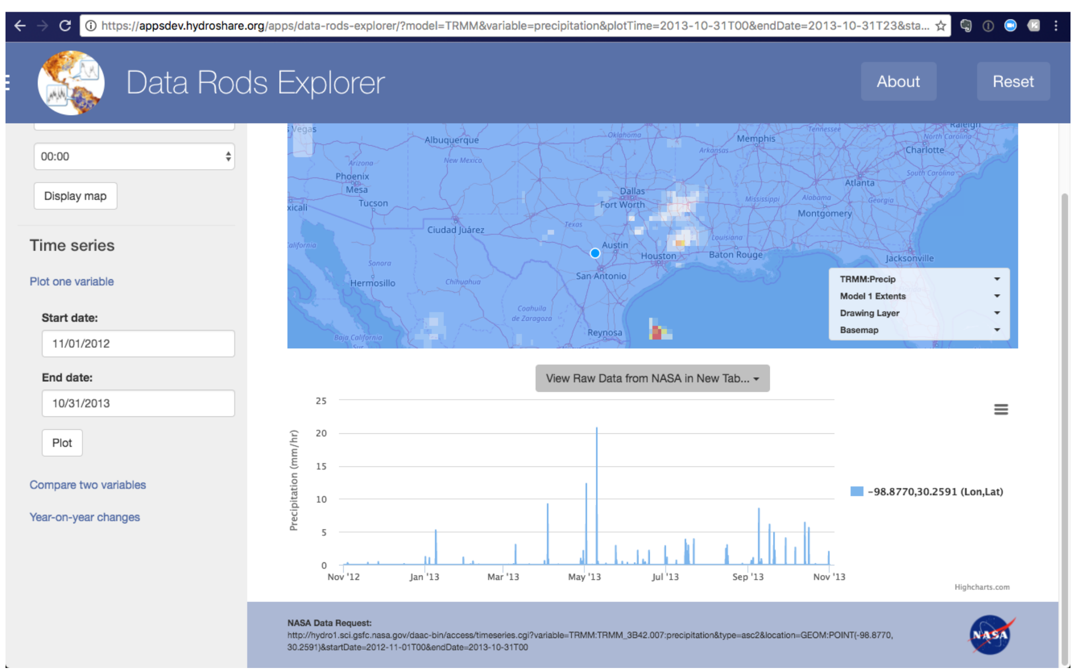


Figure 7. Plot of TRMM Precipitation, November 2012-October 2013

Comparing time series of two variables

1. Pick a model & variable in the lists just described, then click Compare two variables, and pick a second model & variable. Note that **Model 2 extent** is represented by a red box on the map. Also note that the date-time pickers adjust to only allow valid times for both variables selected.
2. Click Plot button to see both plots overlaid.

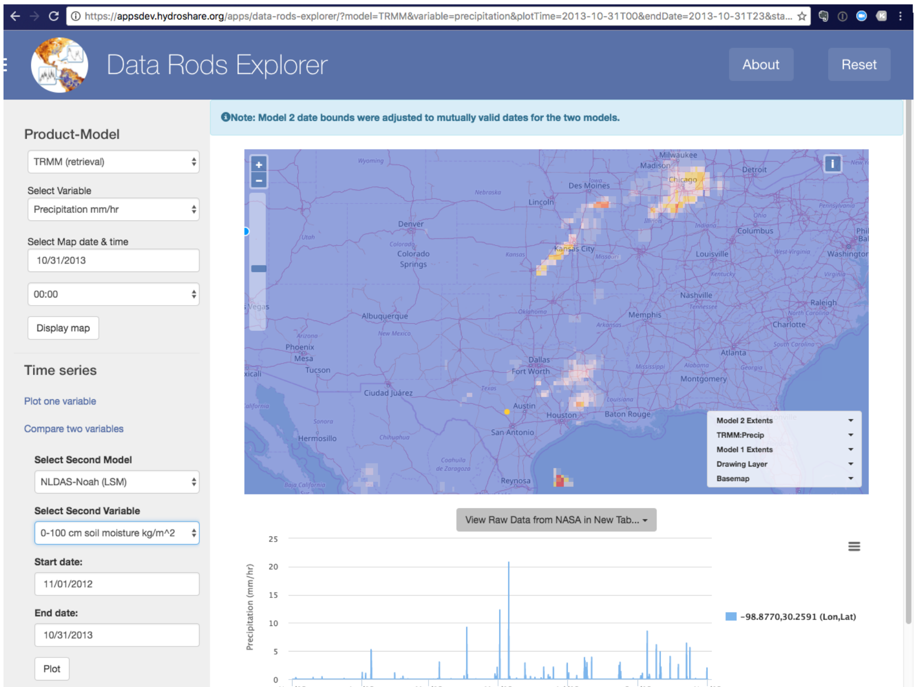


Figure 8. Picking second model for comparison (NLDAS 0-100cm soil moisture)

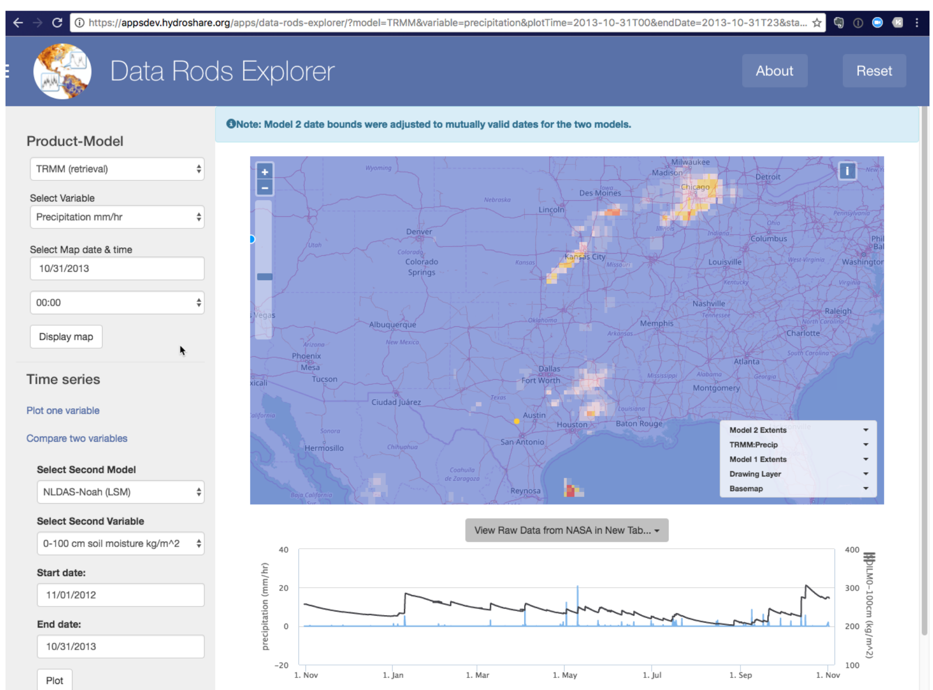


Figure 9. Model 1 plot (blue) uses left y-axis; Model 2 plot (black) uses right y-axis

[Plotting year-on-year changes in a model variable]

For many studies, it is useful to look at year-to-year comparisons for a particular model-variable. This time series plotting option allows you to do that.

1. Pick the model & variable as shown above, then click Year-on-year changes in the Time Series options.
2. In the blank field under “Select years:”, click the mouse to see a popup of years available for the chosen model & variable. Click on a year, then click on the blank field again to add a second year, and so on.

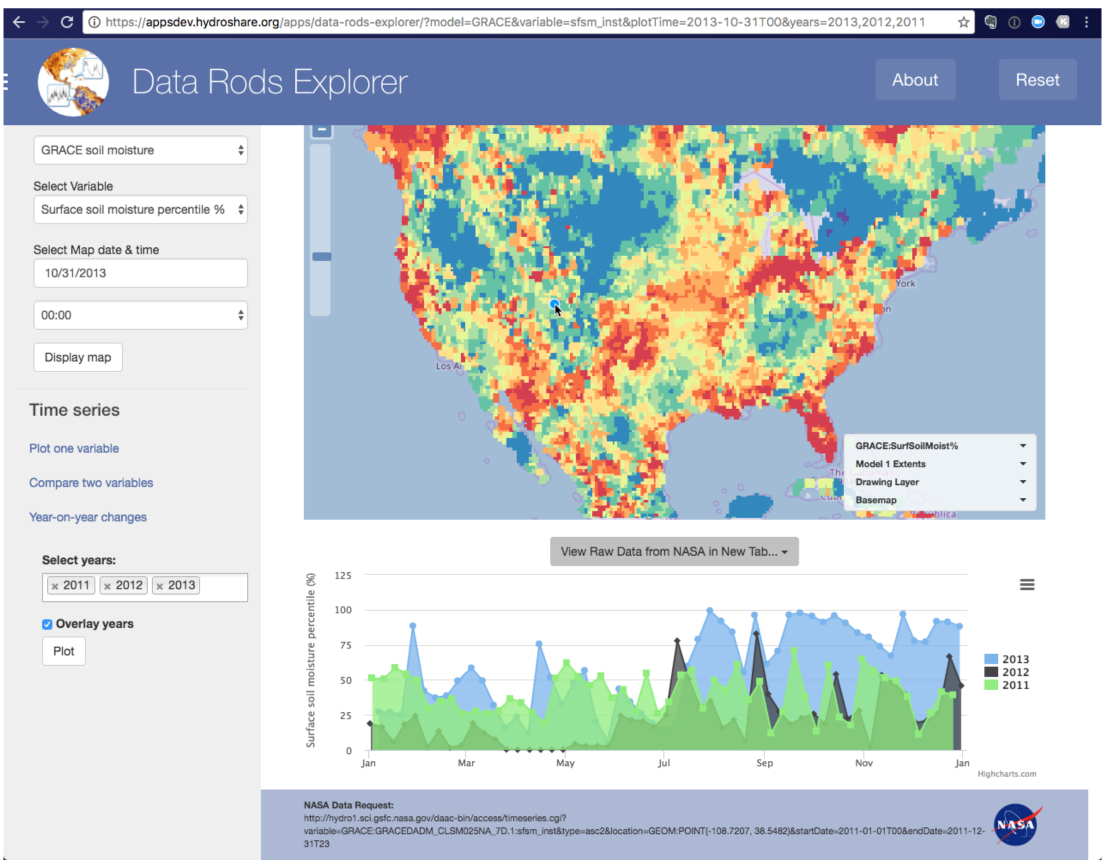


Figure 10. Plotting Year-to-Year comparisons with GRACE soil moisture

- The checkbox “**Overlay years**” below the year-picker overlays the data values for each calendar year chosen. Unchecking this box allows you to see each year in series on a multi-year time scale, not as an overlay.
- Note that you can show/hide individual years by clicking on them in the legend on the right.

Using the DRE user interface tools

Quickly navigating the date picker

The “date picker” is the tool that lets you pick a date for a model-variable map, or a date range for time series plots. This is a combination field: you can directly type in a date (mm/dd/yyyy format), or you can navigate to a desired date in the popup calendar. By clicking on the top-center label of the calendar popup, you can quickly change the scale of months & years to browse, see examples below for NLDAS (data range = Jan 2, 1979 to the present; sampled on Nov 28, 2016).

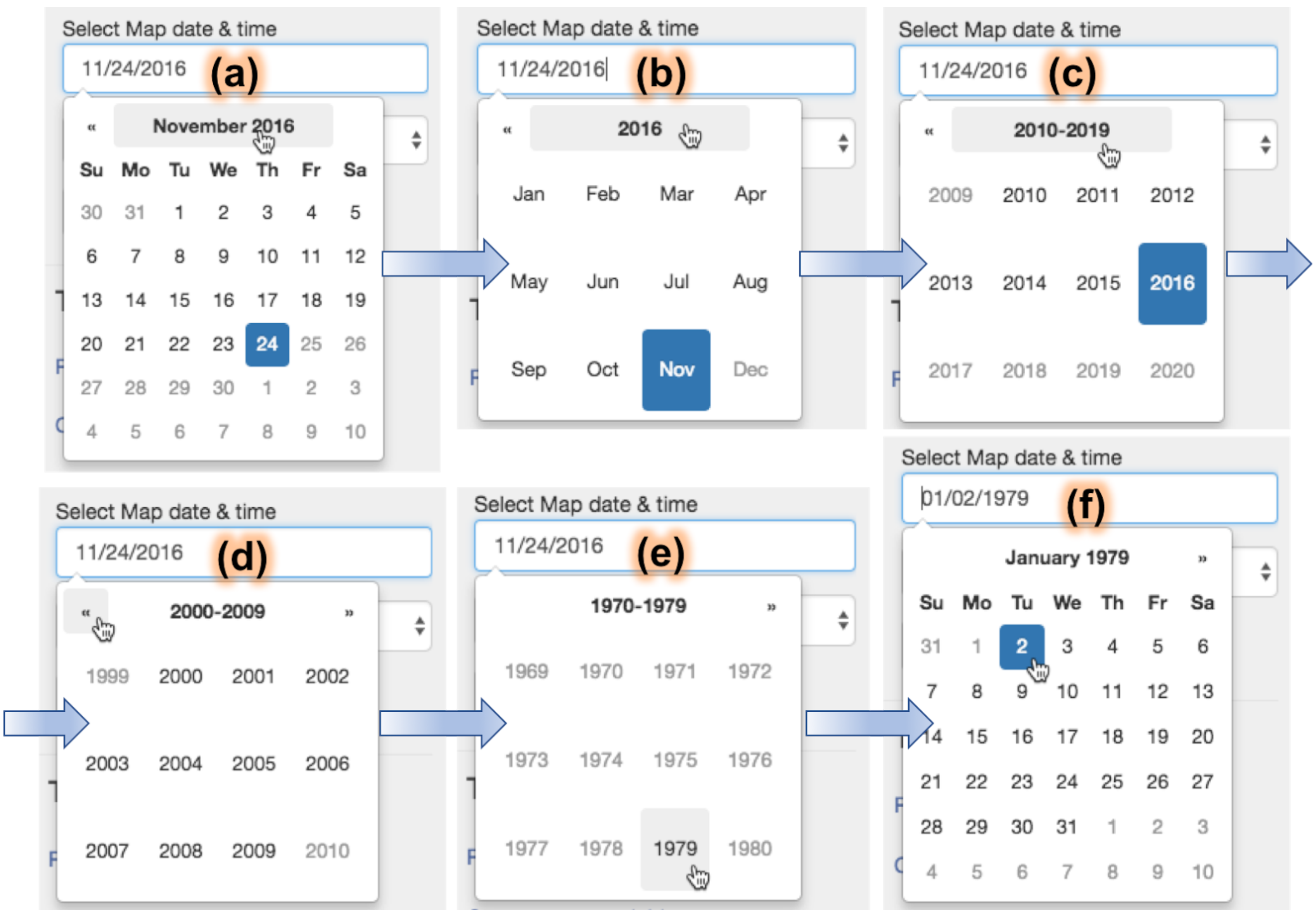


Figure 11. Navigating Days-Months-Years in the Date Picker

Notice that dates after Nov 24 in (a) are grayed out; this is because they are “fenced off” as not ready for viewing. Similarly, years prior to 1979 (e) are not available, nor are dates before Jan 2, 1979 (f).

Tip: Sometimes you may find that you can get a gridded variable map for a date and time, but you cannot get a time series with that end date/time. If this happens, **try changing the end-date to a day earlier, or a month earlier**. If it works then, it was possible that the time series was in the process of being updated, and was not yet indexed through the end date indicated in the NASA CMR registry.

Using the map controls panel

As mentioned elsewhere, you can request and stack multiple model-variable grid maps. You can hide or make any of these layer transparent, to compare with an underlying layer, or to see the underlying basemap. This example reveals the underlying variable grid (GLDAS2-Evapotranspiration) through the LPRM AMSRE-D Optical Depth C-band grid layer.

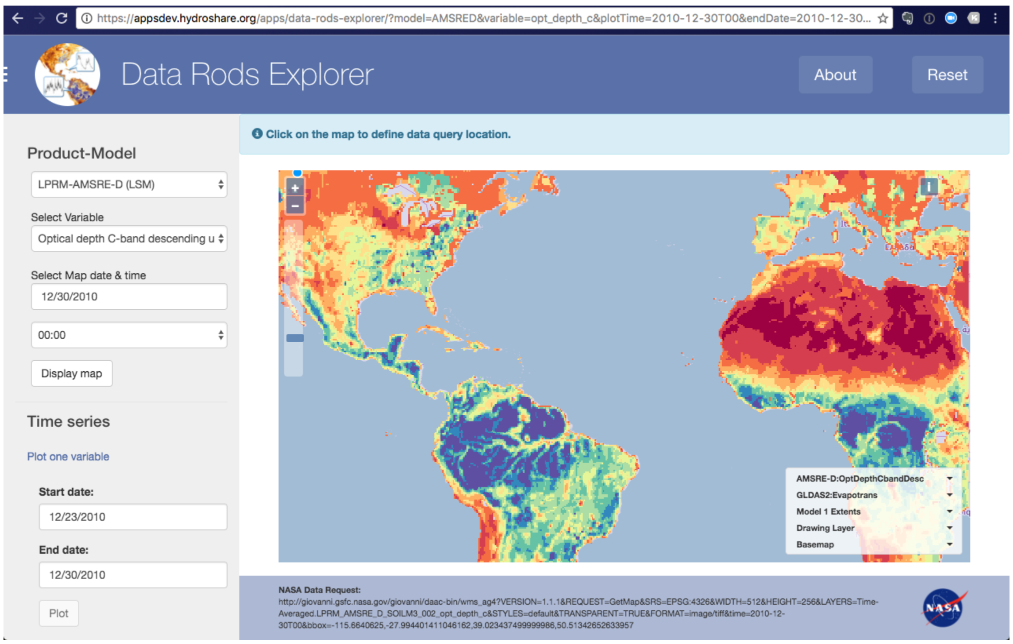


Figure 12a. LPRM-AMSRE-D Optical Depth C-band plotted over GLDAS2-Evapotranspiration

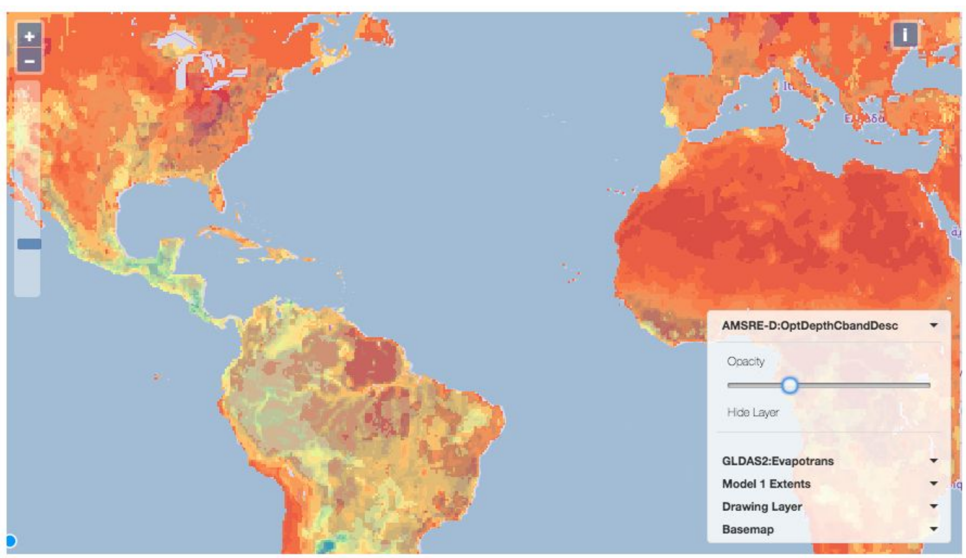


Figure 12b. Making LPRM layer transparent enough to see GLDAS2-Evapotranspiration layer

Note: the **Drawing Layer** in the Map Controls Panel refers to the plot point you can place on the map, for location of the time series plots.

Using the Reset button

The Reset button (upper-right of browser window) is a shortcut for clearing the the model variables, dates, maps and plot graphics chosen during a session. It does this very simply by refreshing the DRE URL to the default: https://apps.hydroshare.org/apps/data-rods-explorer.

Downloading time series data and plot graphics

There are a number of ways you can download the plotted time series data:

- Downloading in Ascii-text tabular format, with or without metadata
- Downloading in netCDF format
- Generating a detailed plot using the NASA plot service
- Downloading the DRE's plot image itself as JPG, PNG, SVG, or PDF

The way to get the most metadata (ie, descriptions of variable and time series) is to download in Ascii text format from the menu button View Raw Data from NASA in New Tab… which appears between the map and the graph after plotting a time series, see Figure 13a below. The ASCII and Plot choices will display the data in a new browser tab, while the NetCDF option will open a file-download dialog to allow you to save the output.

The CSV and XLS download options shown in Figure 13b only give you the minimum data needed to draw the plot itself, that is, the table of values without descriptive metadata. The other download options in Figure 13b are to download a copy of the plot image itself in various formats.

These two approaches for downloading the plot data are both provided because the first approach leverages the NASA data services directly, while the second approach uses the HighCharts plug-in tools. Most of the metadata has been discarded by the time HighCharts plots it in DRE.

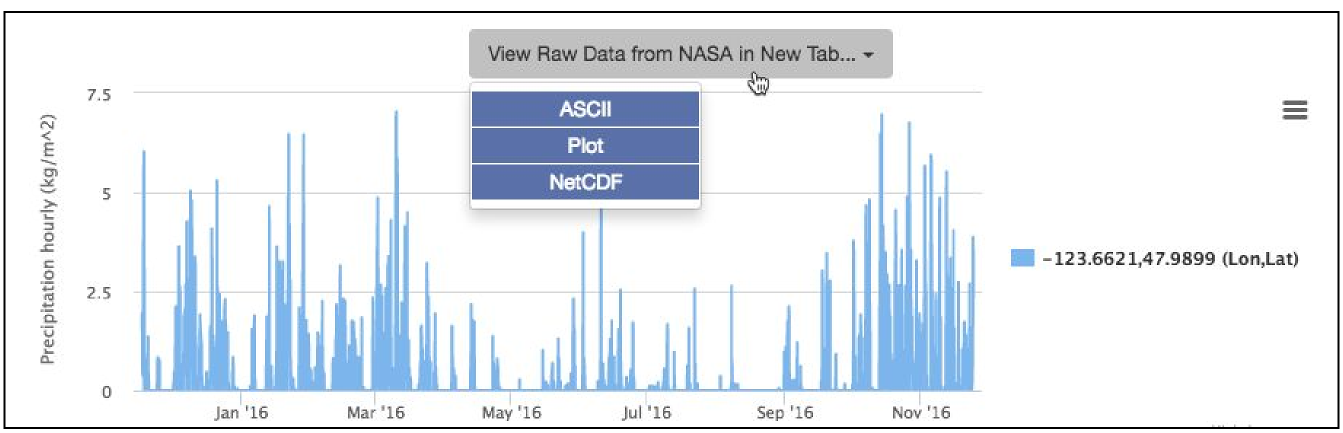


Figure 13a. Download Choices (Part 1) for Time Series Data

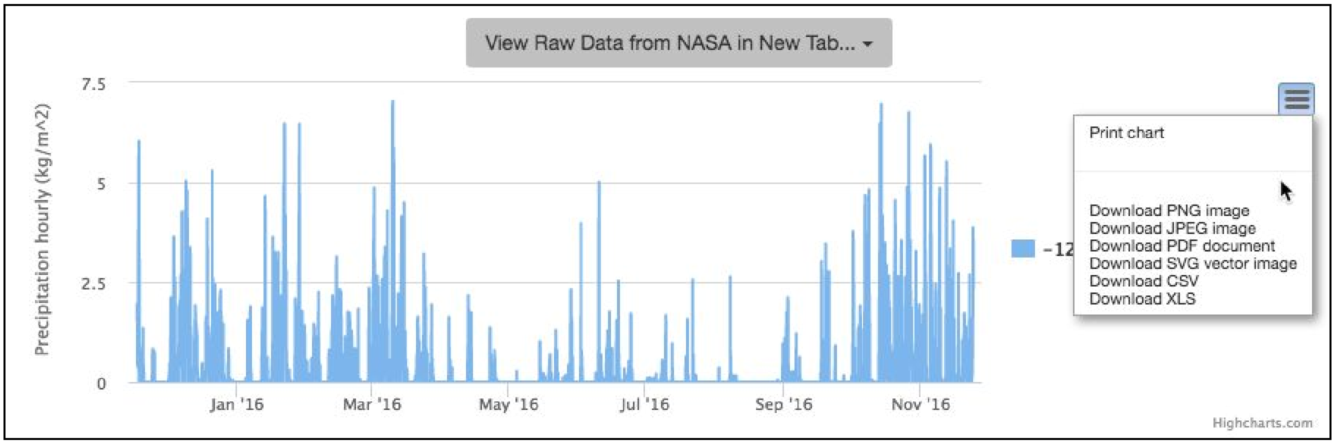


Figure 13b. Download Choices (Part 2) for Time Series Data and Plot Image

One other point to make regarding the View Raw Data choices: if your plots have two variables or year-to-year changes, then choosing to View Raw Data as ASCII, Plot or NetCDF will create multiple new browser tabs, one for each variable-time series. **Tip**: IF YOU DON'T SEE multiple new tabs, this could be due to your browser's pop-up blocker (depends on the browser). Check online for your browser's method of adding an exception to the pop-up blocker, and add an exception for the following sites: **appsdev.hydroshare.org**, and **apps.hydroshare.org**.

Specifying exact long-lat coordinates for time series plots

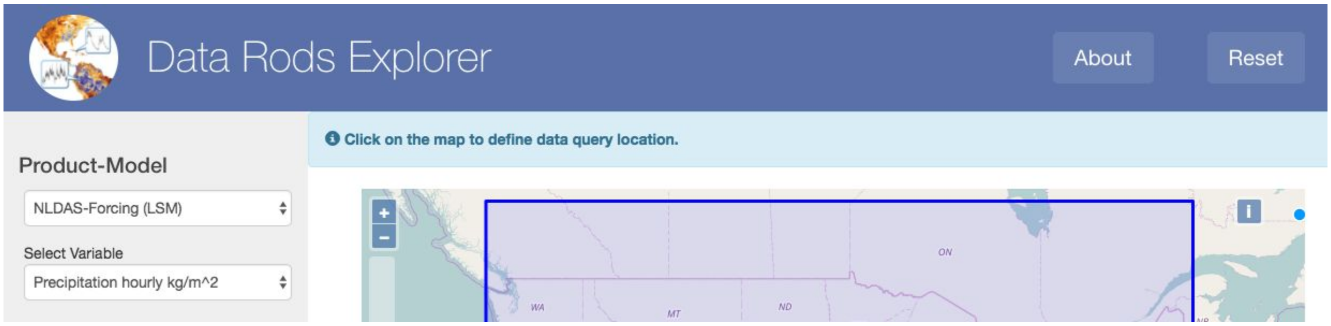
in all this guidance, point locations for time series plots are indicated as chosen by point-and-click with the mouse. There is presently no way to tell DRE to plot a time series for user-entered longitude-latitude coordinates. However, DRE can provide you the necessary content to construct custom queries of your own. Note that exact (longitude, latitude) coordinates are shown in the NASA Data Request URL displayed at the bottom of the DRE browser frame. You can copy a representative NASA Data Request URL into a new browser tab, and edit the location and other parameters.

Tip: In the case that you just want to use a chosen long-lat point location for time series of multiple variables, DRE lets you do this now. As long as the time range and point location you pick are valid for all the desired variables, you can change and plot multiple different model-variables without having to move or reselect the plot location point.

Notifications and Testing

Notices and error messages

1. Messages are displayed between the Data Rods Explorer title bar and the map frame. These are either informative instructions, or explanatory about error conditions. Informative messages are displayed with blue text against a blue background box. The following example shows the most common informative message: "Click on the map to define data query location."



2. At the time of DRE development, the GLDAS model-variables were only available as time series data and plots, not as grid maps. This is why the following message is displayed upon picking the GLDAS model:

   > **GLDAS does not support the "Display Map" function, but data rods data can still be obtained under the "Plot one variable", "Compare two variables", or "Year-on-year changes" options.**

   When GLDAS is selected as the model, the Display Map button is disabled (grayed out). Note that GLDAS will be phased out after Dec 31, 2016.

3. Error messages are displayed in brown text and background box, as shown below. The following message appears if the requested map service is down, or if a map is requested out of temporal or spatial bounds.

   > **⚠A map could not be retrieved for the chosen parameters.**

   If this message persists for a map you expect to be available, try running the NASA Data Request URL displayed at the bottom of the DRE window (see Testing the NASA Data Services section below).

4. The following message appears when the user has clicked on the map outside the model extent.

   > **⚠Query location outside of model extents. Please choose a new location.**

   If the model extent bounds are not obvious at the time you get this message, you can quickly zoom to the model extent by clicking once on **Model 1 Extents** (or model 2) in the map control panel.

5. On rare occasions, Error 999 might appear:

   > **⚠ERROR 999: Data returned by NASA was either missing the expected "Date&Time" line to indicate the start of the data, or returned with an empty dataset.**

   This may happen during brief periods (usually no more than a few hours) between the time the gridded map data became available, and when the data rods indexing was completed. The CMR metadata (including temporal bounds) will match the gridded data, while the data rods may not be ready. So you might be able to draw the gridded variable map, but not see the time series plot. You can try again in a few hours, or change your data request to an earlier time period.

6. On other rare occasions, you might get the Unknown Error:

   ⚠An unknown error occured.

   Try hitting the Reset button, and retrying your requests. If this error persists, take note of the conditions in which it occurred and contact Gonzalo Espinoza.

Testing the NASA data services

DRE accesses NASA data in three main ways: (1) CMR metadata requests to establish temporal and spatial bounds; (2) web map service to obtain the model-variable grid maps; and (3) time series (data rods) service to obtain the data in tabular ascii, netCDF, and plot formats. DRE does not display the CMR bounds queries, but it does display the most recent map or time series query in the blue box below the map frame.

Map request for TRMM Precipitation:

NASA Data Request:
http://giovanni.gsfc.nasa.gov/giovanni/daac-bin/wms_ag4?VERSION=1.1.1&REQUEST=GetMap&SRS=EPSG:4326&WIDTH=512&HEIGHT=256&LAYERS=Time-Averaged.TRMM_3B42_7_precipitation&STYLES=default&TRANSPARENT=TRUE&FORMAT=image/tiff&time=2010-12-30T00&bbox=-134.5126953125,19.41889750820262,-58.48730468750001,53.608126611660424

Data rod request for GLDAS Evapotranspiration:

NASA Data Request:
http://hydro1.sci.gsfc.nasa.gov/daac-bin/access/timeseries.cgi?variable=GLSDAS2:GLDAS_NOAH025_3H_v2.0:Evap_tavg&startDate=2010-12-23T00&endDate=2016-11-20T00&location=GEOM:POINT(-101.9531, 32.3985)&type=asc2

You can copy/paste/execute these URLs into a new browser tab, in case of questions about whether the NASA service is working as expected.

Tip: Data rod requests for Ascii tabular, netCDF, and plots are almost identical. The only parameter you need to change for a time series to get a different format result is &type, as follows:

- &type=asc2 (ascii tabular)
- &type=netcdf
- &type=plot

In case you wish to test all the NASA model-variable data services as a batch process, we have provided a test script called unit_tests.py. A user can invoke this directly from a web browser, as follows:

1. Test all the NASA data services by opening the following link in a new browser tab: https://apps.hydroshare.org/apps/data-rods-explorer/run-tests
   This tests all map and data rods services, with the location of each time series chosen to be midpoint of the model's spatial extent. The default mode for this test is to only query each data service endpoint for the last week of data, which could take a total of a few minutes of clock time for all 55 tests. To test the full temporal range, include the "full" parameter as follows

(takes up to 30 min for 55 tests at the time of this writing): https://apps.hydroshare.org/apps/data-rods-explorer/run-tests/?full=true

2. A developer could execute “python filepath/unit_tests.py” in a terminal window.
3. A developer could run the unit_tests.py script in the python SDE (eg, pycharms).

If run in a browser window, the output will appear in the same browser window; if run from a console terminal, the output will appear in that console window. The output indicates the runtime required, and provides specific URLs that fail to respond. These can be tested again later and/or sent to NASA tech support.

Data Rods Explorer - References for NASA models

Contents

NASA Servers and Search Portals

Initial point of contact for NASA data-related questions is William Teng. All general questions related to data rods should go to gsfc-help-disc@lists.nasa.gov.

NASA Giovanni

Giovanni is the reference data server for all data rods: http://giovanni.gsfc.nasa.gov/giovanni

Note about map legends: It is not yet practical to request and receive a usable map legend for each model variable as it displayed in the DRE map pane. However, users can use Giovanni if it becomes important to know the meanings of the color choices in the DRE grid map displayed for each variable. For example, when you click on the link below, you are taken to a new window showing Giovanni's faceted search screen. If you click "Plot" at the bottom of the UI, you should get the same map as the displayed WMS map in DRE but with legend and all the rest: http://giovanni.gsfc.nasa.gov/giovanni/#service=TmAvMp&starttime=2016-11-14T00:00:00Z&endtime=2016-11-14T00:00:00Z&data=NLDAS_NOAH0125_H_002_soilm0_100cm&dataKeyword=nldas_noah_0125_h (tested Nov 22, 2016)

NASA Global Change Master Directory (GCMD)

Discover Earth science data and services.
Note: the GCMD catalog content is being migrated to the newer CMR, see next item

- Portal search for datasets

NASA Common Metadata Repository (CMR)

The CMR is an earth science metadata repository for NASA EOSDIS data.

- Portal search
- CMR Search API documentation
- Example of finding end-date and bounding-box for models in CMR

References for models and variables

- **NASA Land Surface Models supported**: NLDAS-Forcing, NLDAS-Noah, GLDAS-Noah, TRMM, LPRM AMSRE-D, LPRM AMSRE-A, GRACE
- GES DISC Documentation and FAQ
- Data Rods Time Series Reference (web)
- Data Rods Variables Info (spreadsheet)
- dates_and_spatial_range.txt file (updated in nightly cron job at 3am ET) Here are the contents on Nov 28, 2016:

```
Model name | Begin time | End time | N, E, S, W bounds
NLDASF|01/01/1979 13:00:00|11/25/2016 12:00:00|53.0, -67.0, 25.0, -125.0
NLDAS|01/02/1979 01:00:00|11/25/2016 00:00:00|53.0, -67.0, 25.0, -125.0
GLDAS|02/24/2000 00:00:00|10/31/2016 21:00:00|90.0, 180.0, -60.0, -180.0
GLDAS2|01/01/1948 03:00:00|12/31/2010 21:00:00|90.0, 180.0, -60.0, -180.0
AMSRED|06/19/2002 00:29:47|10/03/2011 21:49:11|90.0, 180.0, -90.0, -180.0
AMSREA|06/19/2002 01:19:10|10/03/2011 20:59:52|90.0, 180.0, -90.0, -180.0
TRMM|12/31/1997 22:30:00|08/31/2016 22:29:59|50.0, 180.0, -50.0, -180.0
GRACE|01/06/2003 00:00:00|05/03/2015 23:59:59|74.0, -45.75, 10.75, -171.0
```

LDAS - Land Data Assimilation System

Reference URL: http://ldas.gsfc.nasa.gov/index.php
The Land Data Assimilation System (LDAS) is a methodology for compiling hydrologic Land-Surface Models (LSMs) and forcing them with observations to remove National Weather Prediction (NWP) forcing biases. Forcings consist of precipitation gauge observations, satellite data, radar precipitation measurements, and output from numerical prediction models derived from existing Surface Vegetation Atmosphere Transfer Schemes (SVATS).

- LDAS FAQ, eg, “is there a shapefile for the NLDAS grid?” (yes)

NLDAS - National Land Data Assimilation System

Reference URL:

- General info: http://ldas.gsfc.nasa.gov/nldas/
- Readme: http://hydro1.gesdisc.eosdis.nasa.gov/data/NLDAS/README.NLDAS2.pdf

Start date: 1979-01-01T13:00:00Z
End date: (ongoing)
Temporal resolution: 1 hour
Spatial resolution: 0.125 x 0.125 degree

NLDAS-Forcing variables: hourly precipitation, surface downward (DW) shortwave and longwave radiation flux, 2-m above ground specific humidity, 2-m above ground temperature, 10-m above ground zonal wind, 10-m above ground meridional wind.

NLDAS-Noah v2 variables: evapotranspiration, ground heat flux, latent heat flux, sensible heat flux, surface runoff, 0-10cm soil moisture, 10-40cm soil moisture, 40-100cm soil moisture, 0-100cm soil moisture, 100-200cm soil moisture, 0-10cm soil temperature.

Tip: Note that many of these variables have volume units of (kg/m^2) or rates like (kg/m^2/s). This is a convenience unit for water equivalence variables, which is the same as “mm water equivalent” for volumes, and mm/s for rates. See FAQ on rain-unit for details about this conversion.

GLDAS - Global Land Data Assimilation System

Reference URLs:

- General info: http://ldas.gsfc.nasa.gov/gldas/
- Readme: http://hydro1.gesdisc.eosdis.nasa.gov/data/GLDAS_V1/README.GLDAS.pdf

Start date: 2000-02-24T00:00:00Z
End date: *(discontinued after Dec 31, 2016)*
Temporal resolution: 3 hours
Spatial resolution: 0.25 x 0.25 degree

GLDAS-Noah v1 variables: evapotranspiration, precipitation rate, rain rate, snow rate, surface runoff, subsurface runoff, 0-10cm soil moisture, 10-40cm soil moisture, 40-100cm soil moisture, 0-100cm soil moisture, near-surface air temperature, 0-10cm soil temperature, near-surface wind magnitude.
Tip: Note that many of these variables have volume units of (kg/m^2) or rates like (kg/m^2/s). This is a convenience unit for water equivalence variables, which is the same as “mm water equivalent” for volumes, and mm/s for rates. See FAQ on rain-unit for details about this conversion.

GLDAS 2 - Global Land Data Assimilation System

Reference URLs:

- General info: http://ldas.gsfc.nasa.gov/gldas/
- Readme: http://hydro1.gesdisc.eosdis.nasa.gov/data/GLDAS/GLDAS_NOAH025_3H.2.0/doc/README_GLDAS2.pdf

Start date: 1948-01-01T03:00:00Z
End date: 2010-12-31T21:00:00Z
Temporal resolution: 3 hours
Spatial resolution: 0.25 x 0.25 degree

GLDAS-Noah v2 variables: evapotranspiration, rain rate, snow rate, surface runoff, subsurface runoff.
Tip: Note that many of these variables have volume units of (kg/m^2) or rates like (kg/m^2/s). This is a convenience unit for water equivalence variables, which is the same as “mm water equivalent” for volumes, and mm/s for rates. See FAQ on rain-unit for details about this conversion.

Note: GLDAS-2 will be enhanced during 2017, to take the place of the GLDAS-1 model being retired.

GRACE - Gravity Recovery and Climate Experiment

Reference URLs:

- General info: http://drought.unl.edu/MonitoringTools/NASAGRACEDataAssimilation.aspx
- Readme: http://hydro1.gesdisc.eosdis.nasa.gov/data/GRACEDA/GRACEDADM_CLSM025NA_7D.1.0/doc/README_GRACEDADM.pdf

Start date: 2003-01-06T00:00:00Z
End date: 2015-05-03T23:59:59Z
Temporal resolution: 7 days
Spatial resolution: 0.25 degree

GRACE variables: 0-2 cm surface soil moisture percentile, 0-100 cm root zone soil moisture percentile, shallow groundwater percentile. Percentile values are based on the period 1948-2012.

LPRM - Land Parameter Retrieval Model

Reference URL: [LPRM overview and metadata]()
AMSRE-Descending Start date: 2002-06-19T00:29:47Z
AMSRE-Descending End date: 2011-10-03T21:49:11Z
AMSRE-Ascending Start date: 2002-06-19T01:19:10Z
AMSRE-Ascending End date: 2011-10-03T20:59:52Z

AMSR-E Descending/Ascending variables: soil moisture C-band percentile, soil moisture X-band percentile, soil moisture uncertainty C-band, soil moisture uncertainty X-band, optical depth C-band, optical depth X-band, 2mm skin temperature AMSRE.

This Level 3 (gridded) data set's land surface parameters, surface soil moisture, land surface (skin) temperature, and vegetation water content, are derived from passive microwave remote sensing data from the Advanced Microwave Scanning Radiometer-Earth Observing System (AMSR-E), using the Land Parameter Retrieval Model (LPRM). There are two files per day, one ascending (daytime) and one descending (nighttime), archived as two different products. The data set covers the period from June 2002 to October 2011 (when the AMSR-E on the NASA EOS Aqua satellite stopped producing data due to a problem with the rotation of its antenna), at a spatial resolution of 0.25 degree. The data are stored in netCDF format.

TRMM - Tropical Rainfall Measuring Mission

Reference URL: http://trmm.gsfc.nasa.gov/
Start date: 1997-12-31T22:30:00Z
End date: (ongoing)
Temporal resolution: 3 hours

The TRMM satellite, a joint mission between the National Aeronautics and Space Administration (NASA) and the Japan Aerospace Exploration Agency (JAXA), collected 17 years of global tropical rainfall and lightning data since December 1997 and April 2015. While the satellite came down in April 2015, the TRMM product 3B42, from which data rods are generated, continues. 3B42 v7 is a merged product, so, even without TRMM inputs, the product continues with other inputs.

TRMM is now a static product. For RT precipitation, we will add GPM (Global Precipitation Measurement) (TBD)

MERRA-Land (forthcoming, not ready yet)

Land Surface Model suite of surface hydrology variables, similar to LDAS
http://disc.sci.gsfc.nasa.gov/uui/search/%22MERRA-2%22 - sample search for MERRA products
http://disc.sci.gsfc.nasa.gov/mdisc/documentation/MERRA-Land_Documentation_20120224.pdf

Keeping the Data Rods Explorer Up to Date

Contents

Using the NASA Data Rods Variables Info spreadsheet

The NASA Data Rods Variables spreadsheet is maintained by NASA GES-DISC staff, and provides the basic information to be used within DRE for all data server requests. Initial point of contact for NASA data-related questions is William Teng. All general questions related to data rods should go to gsfc-help-disc@lists.nasa.gov.

Reading and editing the model_config.txt file

The `model_config.txt` file is constructed from content in the NASA Data Rods Variables spreadsheet mentioned above. This provides a configuration specification for each model and variable, which enables constructing queries to the NASA Common Metadata Repository (CMR) and to the Giovanni data server at the user's direction, for all purposes of the DRE. This only needs to be updated if something changes about the NASA CMR or data servers, or if there are more models and/or variables to be described. Any such changes from NASA should be communicated and confirmed by means of the NASA Data Rods Variables spreadsheet.

We will look at the 2-line header block and the first model in the `model_config.txt` file, as of Nov 20, 2016. The top line in the header block identifies the model-specific content needed for data requests (with "~" field separator character); the second line in the header block identifies the variable-specific details needed for each model (with "|" field separator character).

```
Model_Display_Name~Model_Key_Name~Official_Model_Short_Name~Model_Version~Model_Data_Url_Pattern
Model_Key|Data_Rods_Variable_ID|WMS_Layer_Name|Variable_Dropdown_Name|Units|Variable_Layer_Name
```

Below these header lines, each model block is preceded by a blank line. The first line in each model block defines the model identifiers and the data service endpoint URL template. Note the use of placeholders {0} for the model variable identifier, {1} for the location coordinates, {2} for the start date, and {3} for the end date. These are all filled in through user selections while running DRE.

```
NLDAS-Forcing
(LSM)~NLDASF~NLDAS_FORA0125_H~002~http://hydro1.sci.gsfc.nasa.gov/daac-bin/access/timeseries.cgi?variab
le=NLDAS:NLDAS_FORA0125_H.002:{0}&type=asc2&location=GEOM:POINT({1})&startDate={2}&endDate={3}
```

The remaining lines for each model block define each variable, using the vertical bar "|" character as a field separator. These fields provide the content that shows in the data rods URL placeholder {0}, in model and variable pulldowns, in the map controls, and in time series plots (labels and units).

```
NLDASF|APCPsfc|NLDAS_FORA0125_H_002_apcpsfc|Precipitation hourly|kg/m^2|NLDASF:PrecipHourly
NLDASF|DLWRFsfc|NLDAS_FORA0125_H_002_dlwrfsfc|Surface DW longwave radiation
flux|W/m^2|NLDASF:SurfDWLongWaveRadFlux
NLDASF|DSWRFsfc|NLDAS_FORA0125_H_002_dswrfsfc|Surface DW shortwave radiation
flux|W/m^2|NLDASF:SurfDWShrtWaveRadFlux
NLDASF|SPFH2m|NLDAS_FORA0125_H_002_spfh2m|2-m above ground specific
humidity|kg/kg|NLDASF:2mAbvGrndSpecifHumid
NLDASF|TMP2m|NLDAS_FORA0125_H_002_tmp2m|2-m above ground temperature|K|NLDASF:2mAbvGrndTemp
NLDASF|UGRD10m|NLDAS_FORA0125_H_002_ugrd10m|10-m above ground zonal wind|m/s|NLDASF:10mAbvGrndZonalWind
NLDASF|VGRD10m|NLDAS_FORA0125_H_002_vgrd10m|10-m above ground meridional
wind|m/s|NLDASF:10mAbvGrndMeridWind
```

Running the enddate_bounds.py script (cron job)

Each night at 3am US Eastern Time, a cron job executes enddate_bounds.py, which parses the model_config.txt file, and builds URLs to query the NASA Common Metadata Repository (CMR) for the spatial and temporal bounds for each model's datasets. This is a multi-step procedure:

1. Read the model_config.txt file and construct two CMR requests (begin date & end date) for each model. Here are the current initial queries to CMR for the end dates of each model. Note that `&sort_key=start_date` gets the begin date, and `&sort_key=-start_date` gets the end date.

- https://cmr.earthdata.nasa.gov/search/granules?short_name=NLDAS_NOAH0125_H&version=002&page_size=1&sort_key=-start_date
- https://cmr.earthdata.nasa.gov/search/granules?short_name=GLDAS_NOAH025SUBP_3H&version=001&page_size=1&sort_key=-start_date
- https://cmr.earthdata.nasa.gov/search/granules?short_name=GLDAS_NOAH025_3H&version=2.0&page_size=1&sort_key=-start_date
- https://cmr.earthdata.nasa.gov/search/granules?short_name=TRMM_3B42&version=7&page_size=1&sort_key=-start_date
- https://cmr.earthdata.nasa.gov/search/granules?short_name=LPRM_AMSRE_D_SOILM3&version=002&page_size=1&sort_key=-start_date
- https://cmr.earthdata.nasa.gov/search/granules?short_name=LPRM_AMSRE_A_SOILM3&version=002&page_size=1&sort_key=-start_date
- https://cmr.earthdata.nasa.gov/search/granules?short_name=GRACEDADM_CLSM025NA_7D&version=1.0&page_size=1&sort_key=-start_date

2. The CMR queries each get a short XML response, which contains the URL for the model's first granule (for begin date) or the last granule (for end date).
3. Enddates_bounds.py then sends these two URLs for each model, and parses the relevant dates from the respective granules. The spatial extent is included with the temporal bounds.
4. The dates and spatial extents are written out to dates_and_spatial_range.txt.

Here is the output for the NLDAS-Noah begin-date query (the others are essentially the same):


```
<results>
  <hits>332016</hits>
  <took>117</took>
  <references>
    <reference>
      <name>
        NLDAS_NOAH0125_H.002:NLDAS_NOAH0125_H.A19790102.0100.002.grb
      </name>
      <id>G1238384728-GES_DISC</id>
      <location>
        https://cmr.earthdata.nasa.gov:443/search/concepts/G1238384728-GES_DISC/1
      </location>
      <revision-id>1</revision-id>
    </reference>
  </references>
</results>
```


Visiting the URL in the highlighted text above results in a text file download with the following xml content:


```
<Granule>
  <GranuleUR>NLDAS_NOAH0125_H.002:NLDAS_NOAH0125_H.A19790102.0100.002.grb</GranuleUR>
  <InsertTime>2012-03-05T00:08:43Z</InsertTime>
  <LastUpdate>2012-03-05T00:08:43Z</LastUpdate>
  <Collection>
    <ShortName>NLDAS_NOAH0125_H</ShortName>
    <VersionId>002</VersionId>
  </Collection>
  <DataGranule>
    <SizeMBDataGranule>6.569757461547852</SizeMBDataGranule>
    <ProducerGranuleId>NLDAS_NOAH0125_H.A19790102.0100.002.grb</ProducerGranuleId>
    <DayNightFlag>UNSPECIFIED</DayNightFlag>
    <ProductionDateTime>2012-03-05T00:08:43Z</ProductionDateTime>
  </DataGranule>
  <Temporal>
    <RangeDateTime>
      <BeginningDateTime>1979-01-02T01:00:00Z</BeginningDateTime>
      <EndingDateTime>1979-01-02T01:00:00Z</EndingDateTime>
    </RangeDateTime>
  </Temporal>
  <Spatial>
    <HorizontalSpatialDomain>
      <Geometry>
        <BoundingRectangle>
          <WestBoundingCoordinate>-125.0</WestBoundingCoordinate>
          <NorthBoundingCoordinate>53.0</NorthBoundingCoordinate>
          <EastBoundingCoordinate>-67.0</EastBoundingCoordinate>
          <SouthBoundingCoordinate>25.0</SouthBoundingCoordinate>
        </BoundingRectangle>
      </Geometry>
    </HorizontalSpatialDomain>
```

```
  </Spatial>
  <OnlineAccessURLs>
    <OnlineAccessURL>
<URL>http://hydro1.gesdisc.eosdis.nasa.gov/data/NLDAS/NLDAS_NOAH0125_H.002/1979/002/NLDAS_NOA
H0125_H.A19790102.0100.002.grb</URL>
    </OnlineAccessURL>
  </OnlineAccessURLs>
  <Orderable>false</Orderable>
  <DataFormat>GRIB</DataFormat>
</Granule>
```

The yellow highlight in this last XML file contains the beginning date and time of the NLDAS-Noah model outputs (all variables). The orange highlight contains the spatial bounds.
Here is a sample output of enddates_bounds.py, stored in dates_and_spatial_range.txt

```
Model name | Begin time | End time | N , E , S , W bounds
NLDASF|01/01/1979 13:00:00|11/06/2016 12:00:00|53.0, -67.0, 25.0, -125.0
NLDAS|01/02/1979 01:00:00|11/06/2016 00:00:00|53.0, -67.0, 25.0, -125.0
GLDAS|02/24/2000 00:00:00|09/30/2016 21:00:00|90.0, 180.0, -60.0, -180.0
GLDAS2|01/01/1948 03:00:00|12/31/2010 21:00:00|90.0, 180.0, -60.0, -180.0
AMSRED|06/19/2002 00:29:47|10/03/2011 21:49:11|90.0, 180.0, -90.0, -180.0
AMSREA|06/19/2002 01:19:10|10/03/2011 20:59:52|90.0, 180.0, -90.0, -180.0
TRMM|12/31/1997 22:30:00|08/31/2016 22:29:59|50.0, 180.0, -50.0, -180.0
GRACE|01/06/2003 00:00:00|05/03/2015 23:59:59|74.0, -45.75, 10.75, -171.0
```

The NLDAS bounds found in the previous XML outputs are highlighted in yellow. Note that the end-date for NLDAS came from a separate CMR query, with `&sort_key=-start_date` .

Synchronization issues between the NASA CMR metadata and actual Data Rods availability

Some of the NASA models accessible through DRE are in ongoing development (NLDAS, GLDAS, TRMM), while others are completed and won't continue to change. However, several more models and variables will be added by 2017, so we will be keeping up with those as well. The model datasets provided for DRE have all been specially organized for gridded map & time series access via the NASA Giovanni data server. Whenever the data rods services for NLDAS, GLDAS and TRMM are being updated, there is a window of time from a few hours to a few days, in which the CMR metadata for the original gridded data is not in sync with the data rods services, ie, the data rods may lag the source data. During these times, when we run the enddates_bounds.py cron job to update the dates_and_spatial_range.txt file, the DRE can think it has more recent data for some models than it actually has.

For this reason, before periods where you might conduct a significant level of browsing, plotting, downloads, or especially demonstrations for an audience, it is always a good idea to test and verify the data rods available date ranges and data service health using the DRE unit_tests.py utility developed by Shawn Crawley. See Testing the NASA data services section for details.

Data Rods Explorer: App Process Flow

Contents

Introduction

This portion of the guide is intended to help software developers understand the main events and sequences of program flow. It is very important for future developers to keep this up to date when changing or adding functionality, by versioning in the DRE github repo.

Two important data files mentioned below are the model_config.txt and dates_and_spatial_range.txt. The model_config.txt is maintained and accessed in the github repo, but the github version of dates_and_spatial_range.txt is typically out of date, since this file is updated by nightly cron job. It is maintained in https://apps.hydroshare.org/static/data_rods_explorer/data/dates_and_spatial_range.txt.

User opens the Data Rods Explorer app

1. The “home” function in controllers.py is called
    a. The model_config.txt file is parsed and stored in global python variables
    b. The dates_and_spatial_range.txt file is parsed for the spatial and temporal fences and stored in global variables
    c. The Tethys Gizmos for the user inputs are all defined
        i. Map view
        ii. Date selectors
        iii. Plot buttons
        iv. Model selectors
        v. Year selectors
    d. All previously mentioned variables and Gizmos are passed in the context variable to the Django render function
        i. This function parses the app_base_dre.html file, and replaces all of the django bracket context variable indicators {{ }} with their corresponding context variable.
2. The Django-rendered app_base_dre.html is loaded into the browser window, parsed and processed to be loaded as the app’s home page
3. All JavaScript <script> tags in app_base_dre.html are loaded:
4. The main.js script has an “on page ready” listener (denoted by $(function(){}) which fires and does the following:
    a. Modifies page styling (CSS)
    b. Adds a “singleclick” event listener to the map
    c. Adds listeners to automatically resize/redraw the map when page size changes
    d. Programs the “Reset” button through a “click” listener
5. At the bottom of app_base_dre.html, some <script> tags contain embedded JavaScript code that does the following:
    a. Receives the global config variables from the context and sets them as global JavaScript variables in the model_objects.js script
    b. Calls the loadDefaultHome function from load_pages.js, which does the following:
        i. Parses URL for parameters specifying selected models, variables, and dates
        ii. Either sets the corresponding inputs based on what is in the URL, or if nothing is in the URL, default values are set in the inputs and the URL
        iii. Loads the model 1 extents layer onto the map, which shows the boundaries for which the current model are valid

User clicks on any of the Time Series plot options (Plot one variable, Compare two variables, or Year-on-year changes)

1. The “onClickLink” function from load_pages.js is called, which does the following:
    a. Either shows or hides the user inputs for the option clicked (shows if hidden, hides if shown)
    b. Hides all other user inputs for options not clicked so only one set of inputs is showing. This was mainly done so only one “Plot” button is showing at a time.
    c. Adds or removes the Model 2 extents layer on the map depending on whether the “Compare two variables” was clicked to show/activate the user inputs, or hide them
    d. Based on the option selected, the default values from the user inputs are set and then reflected in the URL

User chooses a different model from the model dropdown

1. The corresponding on-change function (oc_model, oc_model2) from on_change.js is called, which does the following:
    a. The change is reflected in the corresponding URL parameter
    b. The global model fences variable is checked for both the spatial and temporal bounds
    c. The temporal bounds are updated for the date pickers (disabling dates outside of the bounds)
    d. The model extents are redrawn to reflect the spatial bounds

User chooses different variable/dates/time from a dropdown

1. The corresponding on-change function (oc_variable, oc_years) from on_change.js is called, which does the following:
    a. The change is reflected in the corresponding URL parameter

User clicks on the map

1. The map’s “singleclick” event listener registered in main.js is fired and does the following:
    a. Removes any previously drawn point, and then draws a new one at the clicked location
    b. Sets a hidden location input to the lon/lat coordinates of the clicked point
    c. Checks if the click was made within the extent fences
        i. An error is thrown if the click is outside of the model extents and the plot buttons are disabled
        ii. Plot buttons are enabled if the click point is valid

User clicks the “Display Map” button

1. The “load_map” function from helpers.js is called, which does the following:
    a. Parses parameters (corresponding to currently selected model/variable/plotTime) from the URL
    b. Sets a number of hidden form inputs to reflect the map’s current configuration (zoom level, center x and y, and extents)
    c. Serializes the form inputs
    d. Makes a “POST” AJAX call with the serialized form inputs to the map python controller url-mapping corresponding to the “get_map_layer” function in controllers.py
2. The “get_map_layer” function in controllers.py is called and does the following:
    a. Extracts parameters (those that were passed with the AJAX call) from the request object
    b. Assembles a NASA request URL that points to a TIFF image of the layer as defined by the request parameters
    c. The tiff image is zipped in a temporary directory
    d. The temporary zipped tiff image is uploaded to GeoServer as a new WMS layer
    e. The corresponding layer name and geoserver url are returned to the awaiting AJAX success function within the “load_map” function in helpers.js
3. The AJAX success function does the following:
    a. Creates a new WMSTile layer using the OpenLayers API
    b. Adds the layer to the map and legend

User clicks the “Plot” button for “Plot one variable”

1. The “createPlot” function in helpers.js is called, which does the following:
    a. Sets a number of hidden inputs to reflect the map’s current configuration (zoom level, center x and y, and extents
    b. Serializes the form inputs
    c. Makes a “POST” AJAX call with the serialized form inputs to the map python controller url-mapping corresponding to the “plot” function in controllers.py
2. The “plot” function in controllers.py is called and does the following:
    a. Extracts parameters (those that were passed with the AJAX call) from the request object
    b. Assembles a NASA request URL that points to an ascii file of the time series data corresponding to the requested parameters
    c. Retrieves the ascii data and parses the time/value pairs out of it
    d. Configures a Time Series plot view Tethys Gizmo to show the time/value pairs
    e. Configures a url dictionary that provides URL endpoints to the ascii data, a NASA plot of the data, and a potential (future) WaterML download of the data
    f. Sets the time series Gizmo and url dictionary to the context variable
    g. Django-renders the plot.html page, which parses it and replaces all Django context variable indicators with the values of the corresponding context variables
    h. Returns the django-rendered plot.html HTML string to the awaiting AJAX success call within the “load_map” function in helpers.js
3. The AJAX success function does the following:
    a. Initializes/renders the time series plot view Tethys Gizmo with the HighCharts API

User clicks the “Plot” button for “Compare two variables”

1. Same as step 1 of “Plot one variable ‘Plot’ button is clicked”
2. Same as step 2 of “Plot one variable ‘Plot’ button is clicked”, except steps 2b and 2c happen for now for two separate model/variable NASA URL endpoints resulting in two distinct time series.
3. The AJAX success function does the following:
    a. Initializes/renders the time series plot view Tethys Gizmo with the HighCharts API
    b. Calls a “two_axis_plot” function which modifies the time series plot to display two distinct axes for the separate time series

User clicks the "Plot" button for "Year-on-year changes"

1. Same as step 1 of "Plot one variable 'Plot' button is clicked"
2. Same as step 2 of "Plot one variable 'Plot' button is clicked", except steps 2b and 2c happen for now for multiple NASA URL endpoints, one for each year that the user selected. Thus, multiple time series will be parsed.
    a. If the user selected to overlay all of the years, while parsing the time series data, the actual year is replaced with the year "2000" for every time in the time/value pairs so that all the data will overlap
3. The AJAX success function does the following:
    a. Initializes/renders the time series plot view Tethys Gizmo with the HighCharts API
    b. If the user selected to overlap the years, then the x-axis labels are programmatically modified to only show the Month since years will all be set to "2000" for overlapping

Testing NASA Data Services

See Testing the NASA Data Services in the DRE Features and Utilities Guide.

DRE Regression Tests

A list of DRE Regression Tests have been compiled. If a developer has made any significant changes to DRE source code, the developer should verify all prior functionality still exists by running these. If you add functionality, be sure to update the list of tests.