

# Dataset Manual



**Version 5.4.0**  
**Nov. 19, 2018**

Last modified on March 26, 2020

## Table of Contents

Definitions	4
System Interactions with Datasets	4
Organization of Datasets	4
Parts of a Dataset	5
Dataset Format	6
Types of Datasets	6
Map Projection	7
Image Format	7
Video Format	7
Alternative Format – KML	7
Alternative Format – WMS	7
Audio Format	7
Resolution	8
Dataset Considerations	8
Seams	8
Warping	8
Color Suggestions	9
Orientation of Data	9
Frame Rate	9
File Names	9
Labels and Colorbars	10
Captions	11
Media Folder	11
Differences in Playlist Files	12
Dataset playlist.sos Files	12
A playlist.sos Example	12
Basic Options in the Playlist	13
Attributes for Texture Datasets	13
Attributes for Time Series Datasets	13
Autorun Datasets	14
Picture In a Picture	14
PIP Style	14
PIP Timing	15
PIP Size	15
Text PIP	16
Moving PIP	17
Shared PIP	17
Live Video PIP	18

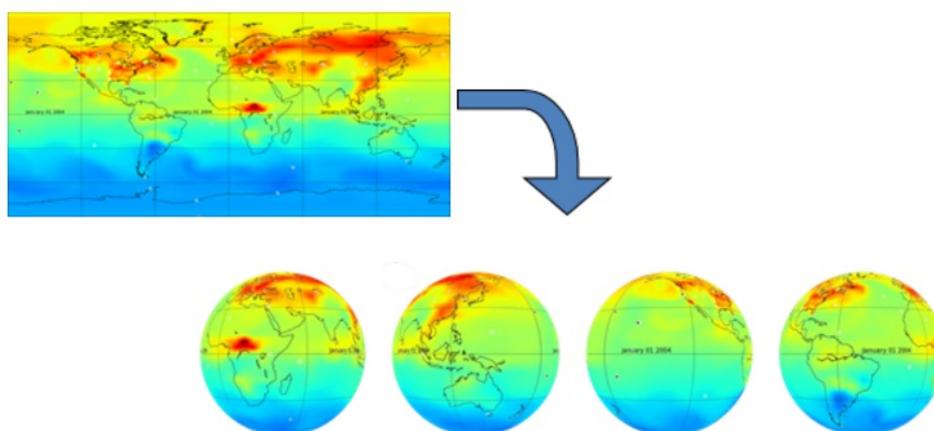
Incorporating in a Playlist	18
Requirements	18
Limitations	18
<b>Annotation Icons</b>	<b>19</b>
Dataset Specific	19
General Icons	19
<b>Layers</b>	<b>19</b>
Predefined Layers	19
Orienting Layers	20
<b>Overlays</b>	<b>20</b>
<b>Using KML Data</b>	<b>20</b>
Implementation Notes	20
Special Notes for KML	20
Limitations for KML	21
<b>Using WMS Data</b>	<b>21</b>
Special Notes for WMS	21
Limitations for WMS	21
<b>Real-time Datasets</b>	<b>22</b>
<b>Tools to Create Datasets</b>	<b>22</b>
<b>Notes about Site-Custom Datasets</b>	<b>22</b>
<b>Submitting a Dataset to the SOS Catalog</b>	<b>22</b>
<b>Visual Playlist Editor</b>	<b>23</b>

# Definitions

1. **Content** – General term that we use for anything that can be displayed on the sphere and should be stored somewhere in /shared/sos/media - mp4, jpg, png, pip, overlay, label, colorbar, caption...
2. **Dataset** – A packaged collection of coherent content, which may include multiple layers, labels, legends, colorbars, captions, etc
  - o **Texture** – A single, static image on the sphere that rotates
  - o **Time series** – animates through time and by default doesn't rotate. Can be an image sequence or a mpeg4
    1. **Image sequence** – a directory of images that are played in sequence
    2. **MPEG4** – the only video format accepted by SOS
3. **Presentation playlist** – A collection of datasets grouped together in a list for a presentation
4. **playlist.sos** – A text file that specifies how a dataset should be displayed on the sphere. Each dataset must have its own playlist.sos file

# System Interactions with Datasets

When a dataset is projected on the sphere, you are really looking at four images that have been merged together seamlessly around the sphere. The Science On a Sphere® software splits the images that you display into four disk images every time you load a new dataset on to the sphere. Because all of the work is done by the software automatically, you don't need to do anything except point the system to where the data is located by creating a playlist.



# Organization of Datasets

Science On a Sphere® comes with over 550 datasets preloaded on the system. All of the NOAA-provided SOS datasets are put into one of the seven main categories. These categories are:

- Air
- Extras
- Land
- People
- Snow and Ice
- Space
- Water

The "Extras" category contains assorted datasets that don't fit into the other categories. Within each category there are many subcategories. Datasets can be put into multiple categories and subcategories. A full list of all of the datasets sorted into their categories is available in the [Data Catalog](#).

This organization is used on the SOS website catalog, in the SOS Remote app, and in the SOS Stream GUI library. The datasets are stored on the SOS computer in directories that use an old naming scheme, so it's similar, but not exactly the same. For example, the Air datasets are stored in a directory called atmosphere and the Space datasets are in a directory called astronomy. These old names were maintained for consistency for older sites.

All the datasets are stored on the SOS computer in [/shared/sos/media](#) . The directories that you will find in here are:

- **astronomy** - contains the Space datasets
- **atmosphere** - contains the Air datasets
- **database** - contains a file used by the SOS Remote app
- **extras** - contains narrated movies that are now dispersed, plus Extras datasets
- **land** - contains the Land datasets

- **models** - contains models that are now dispersed in Air, Ocean, and Snow and Ice
- **oceans** - contains the Water datasets
- **overlays** - contains the overlay datasets
- **playlists** - contains files used for downloading datasets

Within these category directories you will find a separate folder for each of the datasets. In some cases, related datasets are grouped together into subfolder. For example, in the land directory, there is a `blue_marble` directory that contains four subdirectories for the four Blue Marble datasets.

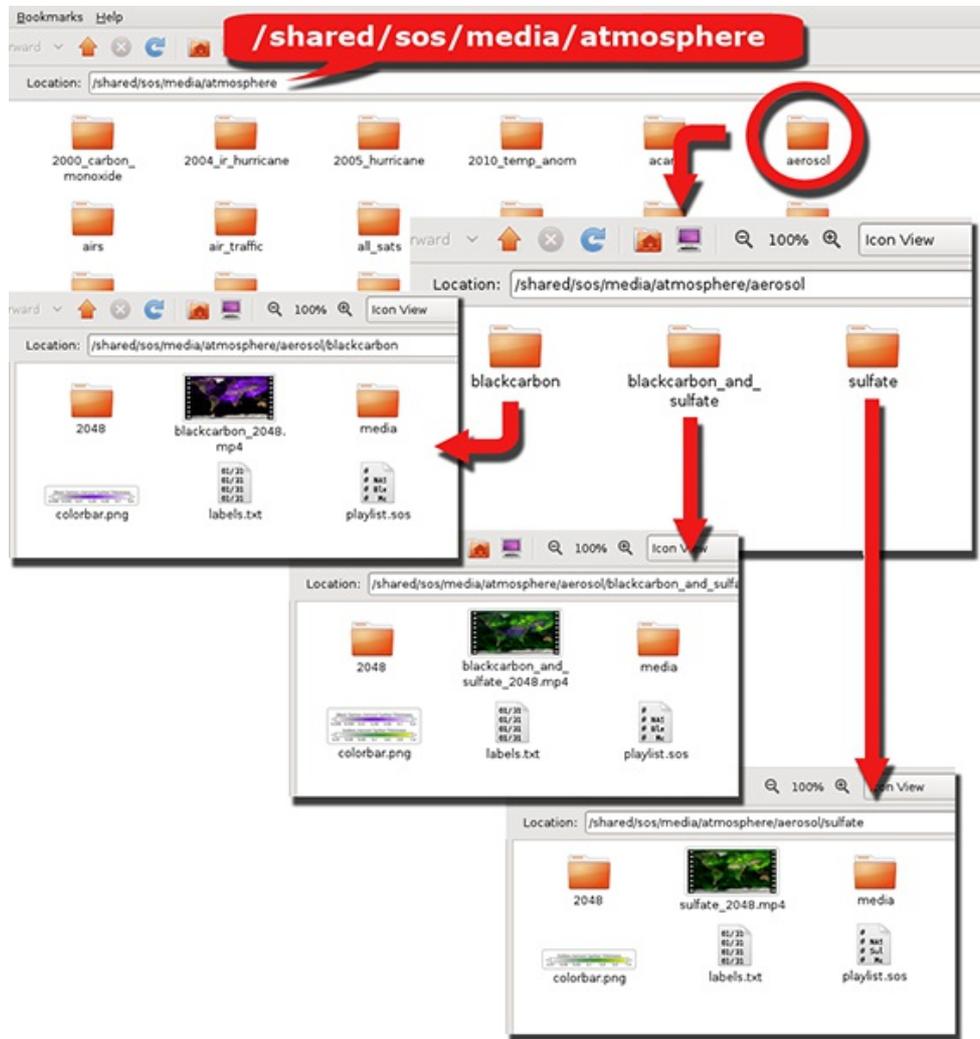
All datasets are stored in just one location, regardless of how many categories they are in on the website and SOS Remote App. For example, the Japan Earthquake, Tsunami Wave Propagation, and Wave Heights Combo dataset can be found in the Land and Water categories, but is stored only in the oceans directory. You can find a dataset's location by click the FTP link on the description page on the website, by pressing the Details button in the SOS Stream GUI when the dataset is loaded, or by pressing the Info button on the SOS Remote App when the dataset is loaded.

All datasets created by the site have to go in the site-custom directory in `/shared/sos/media`. In the site-custom directory, we recommend that you create a folder for each individual dataset that you create, but we leave that up to each site. If you use the Visual Playlist Editor, it requires you to save each dataset into a folder of its own.

## Parts of a Dataset

Every dataset that is added by the user to the SOS Data Catalog should be in its own individual folder in the directory `/shared/sos/media/site-custom`. Do not put your site-created content into the folders created by NOAA. Complementary or similar datasets can be grouped together in a folder that contains individual folders for each dataset. There are many content elements that can be used to create a dataset. For the NOAA-provided datasets you will find the following elements in each folder (not all of these elements are available for each dataset):

- An equatorial cylindrical equidistant JPEG or PNG file named for resolution
- A folder with an equatorial cylindrical equidistant image sequence named for the resolution of the images
- An equatorial cylindrical equidistant video (.mp4) of the data
- Text file labeled `labels.txt`
- Text file labeled `playlist.sos`
- Text file in the SubRip Subtilte (.srt) file format
- Color bars and other supporting images
- Media folder with thumbnails, videos, and supporting documents



In the above example of a NOAA provided dataset, there are three datasets that are related and all from the same source. To keep them together, a folder was created in the atmosphere category called aerosol. In the aerosol folder, each dataset was given its own folder: sulfate, blackcarbon, and blackcarbon\_and\_sulfate. Notice that there are no spaces in the names! In each of the individual dataset folders there is an image sequence named for the resolution of the frames, a MPEG4 video, a media folder with thumbnails, a colorbar, labels, and a playlist.sos.

A uniform naming convention has been used among the folders in the NOAA provided directories. Images that are projected onto the sphere are named for their resolution, movies that are projected onto the sphere are named for their dataset name and resolution, all labels are named labels.txt, etc. This has been done to make it easy for the user to know what is available in each folder. This naming convention doesn't need to be used for site-custom datasets.

The only two elements that are required to have a dataset are the playlist.sos file and something to be displayed on the sphere, either a single image, an image sequence, or a MP4 video file. All of the other elements are optional.

## Dataset Format

### Types of Datasets

#### Texture - Single Image

Textures can be displayed on the sphere and rotate around the poles of the sphere, as in planetary rotation. A good example of a texture is Mars. In the Mars folder you will find just one image, named for its resolution, which is projected on the sphere. Often, the textures are available in several different resolutions. As the resolution increases, so does the loading time on the sphere. Textures can be rotated in any way using the functions either in the Control menu of the SOS Stream GUI or in the remote, and by default rotate around the poles of the sphere. Textures will rotate indefinitely until pause is pressed or another dataset is loaded.

#### Time Series - Image Sequence or MPEG4

There are several different ways that time series work. In its raw form a time series is an image sequence. SOS will display in sequential order all the images in a single directory. The images can also be converted to a MPEG4. This is the preferred format for showing time series because the system can play the MPEG4 files at a higher resolution and faster frame rate than the image sequences and they take up less disk space. There is no limit to the length of a time series. Time series by default don't

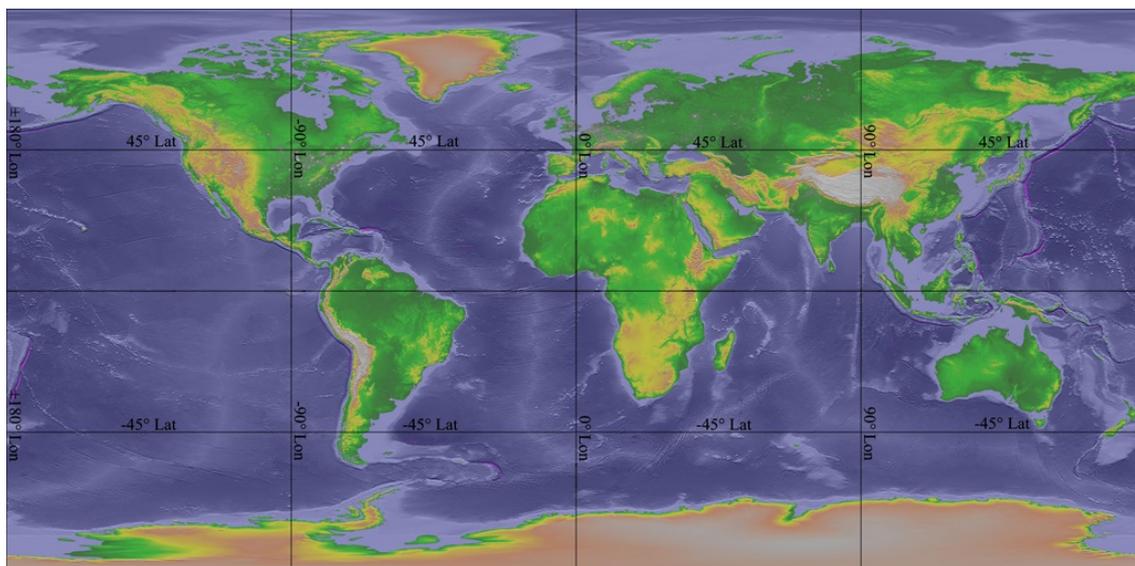
rotate, but can also be set to rotate while they are animating through a feature in the playlist and a button on the remote.

Transitions, special effects, and other computer graphics techniques can be added to a sequence through the use of off the shelf software like Final Cut Pro. Time series will loop indefinitely until pause is pressed or another dataset is loaded.

NOTE: Even if you make an MPEG4, it's good to keep the image sequence, if available.

### Map Projection

The map projection used is the Equatorial Cylindrical Equidistant projection. This is a simple latitude/longitude grid with the lines evenly spaced where the image is twice as wide as it is tall. To be consistent with the SOS Data Catalog, it is recommended that the 0° line go through the middle of the map, with the +/-180° lines at the edges. In order for the data to wrap properly around the sphere, it is imperative that you follow the specifications for the data format closely. Images in the wrong format will project on the sphere, but they will not correctly represent the size of the continents.



### Image Format

SOS will accept most common formats (JPEG, PNG, GIF, TIFF, etc.), but JPEG and PNG are preferred for images and image sequences. In addition to pointing to files that are local on your computer, you can also use URLs, such as <http://example.com/image.jpg>.

### Video Format

Render the video with the MPEG4 video codec at a minimum of 25 mbps. Just because a file has a .mp4 extension does not mean it will play perfectly on SOS. Be sure to check the codec that was used to render the file. The H.264 codec should not be used because it can cause errors in the SOS software.

### Alternative Format – KML

The software also has the ability to display KML (Keyhole Markup Language) files on Science On a Sphere. KML is a popular specification and actively used with Google Earth for displaying data on a sphere. The initial SOS KML capability supports a limited set of the entire specification, which includes many of the commonly used KML features you would typically display in Google Earth. An SOS playlist can reference both KML and the compressed KMZ formats using the layerdata attribute. Jump to the section on [KML](#) for more information.

### Alternative Format – WMS

SOS has an experimental capability to specify and load images directly from an Open GeoSpatial Consortium (OGC) Web Mapping Service (WMS). This feature requires an internet connection and will not work unless the SOS system has access to the internet and the referenced WMS Server. A WMS provides a service allowing users to request data through URLs using specific

key value pairs defining terms such as the width, height, image type, etc. A unique feature of the WMS standard allows users to request subsets of imagery by defining a bounding box using a lower left and upper right latitude and longitude coordinates. The combination of these features allows users to host very large high resolution imagery and users can request smaller versions or subsets of the original imagery. SOS takes advantage of this functionality through the magnifying glass, allowing users to see more detail as you increase the zoom level on the sphere. Jump to the section on [WMS](#) for more information.

### Audio Format

The audio player for SOS is fairly versatile and most common formats will work with SOS, such as mp3, mpeg4, wav, ogg, and aif. The audio file can either be a standalone file or embedded in the same MPEG4 file as the video.

### Resolution

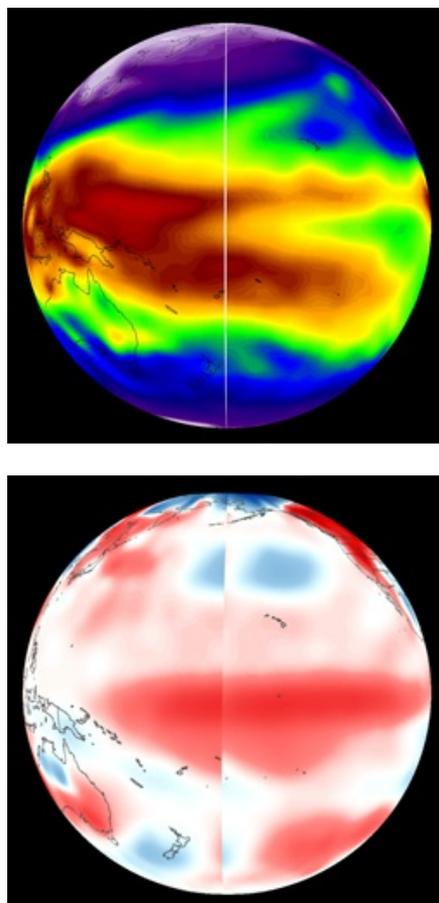
The recommended resolutions are 2048x1024 for time series and 4096x2048 for textures, though other 2x1 resolutions, such as 3000x1500 will work. Higher resolutions are possible for the animations, but not all SOS systems in the SOS Users Collaborative Network are able to animate at 30 frames per second for higher resolutions. Make sure to render the videos with square (1:1) pixels for proper playback. If possible, the highest resolution of the animation that is generated should also be provided, in addition to the 2048x1024 file, so that it is available for new and upgraded sites that have the ability to play them. If an image sequence is provided, it will be rendered into a MPEG4 file that is 2048x1024 unless there is a specific reason to keep the animation as an image sequence. For single images, resolutions above 4096x2048 are possible, though load time increases with resolution size.

As of release 5.0, the SOS software is capable of working with higher resolution 4K projectors. If you wish to create higher resolution images and movies to take advantage of the higher resolution quality of 4K projection, the same 2:1 ratio rules apply. Movies with 4096x2048 resolutions will work on such systems, although decreased FPS rates may be necessary and the load time will be longer. Single images of 8192x4096 will display well on 4K projectors and images as large as 16384x8192 should still work, although resolutions above 8192x4096 will result in only slightly increased quality. We recommend that you test any high resolution images and movies you create on a full 4K projector system to ensure they will display as you expect.

## Dataset Considerations

### Seams

It is important that the data fill the entire image space. If there are borders or extra space around the edges then a seam will appear on the sphere with spots on the poles. It is also important that the data match on either side of the edges of the image. If you don't take this into account, you'll end up with seams like the ones shown here:



In addition to making sure that the data fills the entire frame, from 180° West to 180° East, also make sure that the data fills the entire frame from 90° South to 90° North. If there is missing data at the poles, fill in the area with a solid color or a basic land/ocean background to ensure that the dataset wraps properly around the sphere without stretching vertically.

### Warping

When working with a spherical surface, warping is always something to consider. The least amount of warping occurs near the equator, while the most warping occurs at the poles. Because of this, it is recommended that any text and labels are placed near the equator. Supplementary text, labels and images that are displayed as PIPs don't warp if their position is set with the pipcoords attribute in the playlist. Datasets can be tested for warping issues using CC Sphere in Adobe After Effects or 3D Sphere in Photoshop. There are some plugins for After Effects such as Cycore Effect's Sphere Utilities that can also help with spherical warping. In addition, these

programs can be used to check for seams.

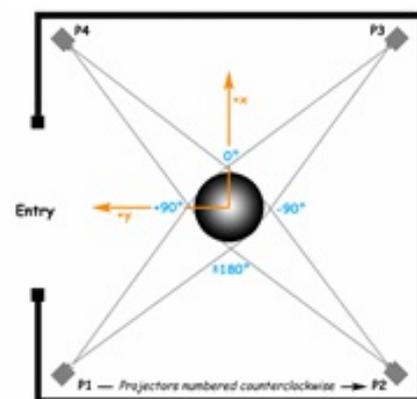
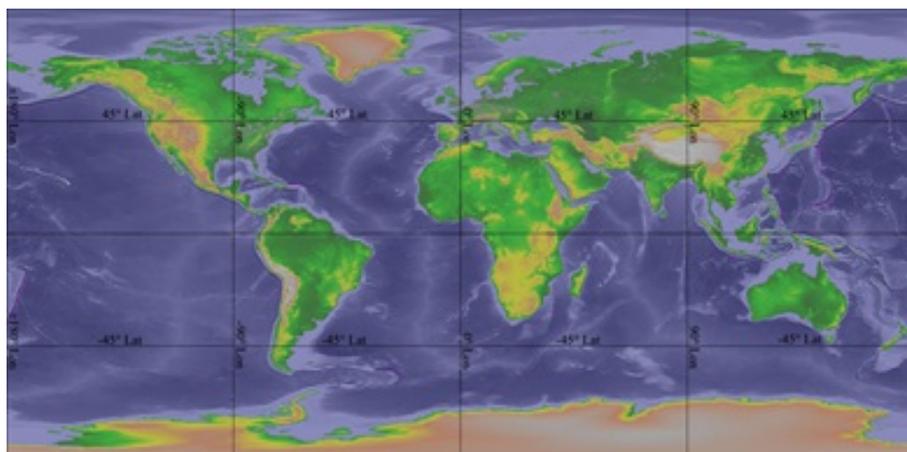
### Color Suggestions

A color scale can dramatically change the emphasis and message of a dataset. Because of this, the Science On a Sphere Users Collaborative Network has had many discussions on the color scales that are used for SOS datasets. The goal is to create datasets with well-chosen color scales that are meaningful, intuitive, and scientifically accurate. Several conclusions are the result of these discussions:

- It can be confusing to users when the same color scheme and their associated color bars are used for two completely unrelated datasets.
- The same color should not be used to represent more than one thing. i.e. if ice is shaded white, then white should not also be used for areas of missing data.
- Using rainbow-colored legends and color schemes is often confusing to the audience and hard to parse. Instead, consider using shades of green to represent phytoplankton, and blue and red gradations to represent temperature anomalies. i.e. use “meaningful” colors
- Avoid using full sphere backgrounds that are completely or pre-dominantly white. The seams between projectors become more apparent when using solid white/bright backgrounds. If using bright background colors, consider adding some noise/texture to them. Same goes for PIPs, especially those displayed at the seams between projectors

### Orientation of Data

The maps created for SOS should be centered on the Prime Meridian, so that 0°N,0°E is the center of the image, as seen below left. The map on the next page is from the dataset in the extras category called SOS Coordinate System that is useful when learning how datasets are loaded on the sphere. The center of the map loads between projectors three and four with the edges of the map lining up between projectors one and two, as seen in the diagram to the right of the map.



### Frame Rate

You can animate a time series at any rate, but 30 frames per second is the recommended speed. We try to create our time series so that they look smooth and animate well at 30 fps. The frame rate is sometimes limited based on the time resolution of the data and the type of data. It is important to keep this in mind when creating a time series so that you make enough images to ensure that the dataset plays for a reasonable length. If you only make 30 images, then it will only take one second to loop through the dataset at 30 frames per second. The optimal playback speed is chosen based on the number of frames and the degree of change between each frame in the sequence. To get smooth time series, the changes between each frame should be small and the playback speed high. If a time series is coarse, then it might animate better at a slower frame rate such as 10 – 15 fps.

### File Names

Single images are typically named for their resolution, such as 4096.jpg. Image sequences are kept in folders that are named for their resolution, and the images themselves should be named to sort in ascending order from earliest to latest. This can either be done with a time stamp in the file name, or a frame number in the file name with a sufficient number of leading zeros to ensure proper sorting, shown in the example below. Videos should be named based on content and resolution, such as hurricanes\_2048.mp4. By including the resolution in the file names, SOS users are able to easily determine what is available and appropriate for their system. Linux and the SOS software do not handle spaces and special characters in file names well. Do not use spaces and special characters in file names!



### Named by Date

```
snow_ice_2048_20110730.png  
snow_ice_2048_20110731.png  
snow_ice_2048_20110801.png  
snow_ice_2048_20110802.png  
snow_ice_2048_20110803.png  
snow_ice_2048_20110804.png  
snow_ice_2048_20110805.png  
snow_ice_2048_20110806.png
```

### Named by Order

```
sos_jpl_4096.0001.jpg  
sos_jpl_4096.0002.jpg  
sos_jpl_4096.0003.jpg  
sos_jpl_4096.0004.jpg  
sos_jpl_4096.0005.jpg  
sos_jpl_4096.0006.jpg  
sos_jpl_4096.0007.jpg  
sos_jpl_4096.0008.jpg
```

## Labels and Colorbars

Labels and colorbars are important for providing context. Labels and color bars can be in the frames or projected on top of them externally. It is recommended that you do not add the labels and color bars directly to the frames that you create. By keeping them as external images, you have much more flexibility with their size and position within the playlist.sos file. If you do choose to put your labels and colorbars directly on the images that you create (we call this burning them in), make sure that you make them big enough so that they are legible on the sphere and that they are far enough from the poles that they don't get too warped.

One of the nice things about keeping the labels and color bars external is that they don't move as you rotate a dataset. They stay in the same position relative to the projectors. Labels and color bars that are part of the frame rotate with the frame, which can cause viewing trouble for the audience as you move the sphere about.

SOS version 5.3 and after: Use the [SOS Visual Playlist Editor](#) to set the label position to a non default value, and use the [Label Editor](#) to fully customize the format (color, font, etc) of label text.

SOS version 5.2 and before: Within the playlist.sos file you can set the position using the `labelposition` attribute, which is set by the x and y position as a pair of coordinates (x,y). Both x and y can vary from -1 to 1. The default position is (-0.3, -0.5). The label color can be changed with the `labelColor` attribute which can be set to R, G, B, Alpha, (or the symbolic names: white, black, red, green, blue, ...). The default color for the labels is white. While you can change the position and color, you cannot change the size or font.

For every dataset that needs a timestamp, a simple text file called labels.txt should be generated that contains one line for each frame in the animation. A labels.txt file cannot be used with a single image. If you have labels for a times series that contains 2000 frames, then you need a text file that has 2000 lines. In the playlist.sos file, if `label = default` is included, then the image file names appear as the labels on the sphere. The labels file should be stored in the dataset folder and is typically named labels.txt. The labels usually contain the date and maybe a title. Here are some examples of the content of labels.txt files:

```
07/24/2004 06:45  
07/24/2004 07:00  
07/24/2004 07:15  
07/24/2004 07:30  
07/24/2004 07:45  
07/24/2004 08:00  
07/24/2004 08:15  
07/24/2004 08:30  
07/24/2004 08:45
```

```
SSEC 08/31/2005 Katrina  
SSEC 08/31/2005 Katrina  
SSEC 08/31/2005 Katrina  
SSEC 08/31/2005 Katrina  
SSEC 08/31/2005 Katrina Maria  
SSEC 09/01/2005 Maria  
SSEC 09/01/2005 Maria  
SSEC 09/01/2005 Maria  
SSEC 09/01/2005 Maria
```

If you don't want to include the title in your labels file, then you can make an image of title that you can include as a Picture in a Picture. This allows you to choose the font and color scheme of your choosing when you make the title image. This is also a nice option because then you don't need to insert the title into every line of your labels file. Here is an example of a title that was made into an image for pip:



Alternatively, you can directly create a title - or any text in general - on SOS by using a [Text PIP](#). In addition to using the labels.txt file, there are other ways of labeling the content that is on the sphere. Colorbars and legends can be added using the Picture in a Picture feature. As with the timestamps, do not "burn in" colorbars and legends. Make sure to test the size and fonts of colorbars and legends on the sphere to ensure legibility. Most colorbars and legends are generally too small when originally displayed on SOS and have to be increased in size. Also, make sure to test the size to ensure that the colorbars aren't so large that they hinder the audience from seeing the underlying dataset.

## Support: Dataset Manual

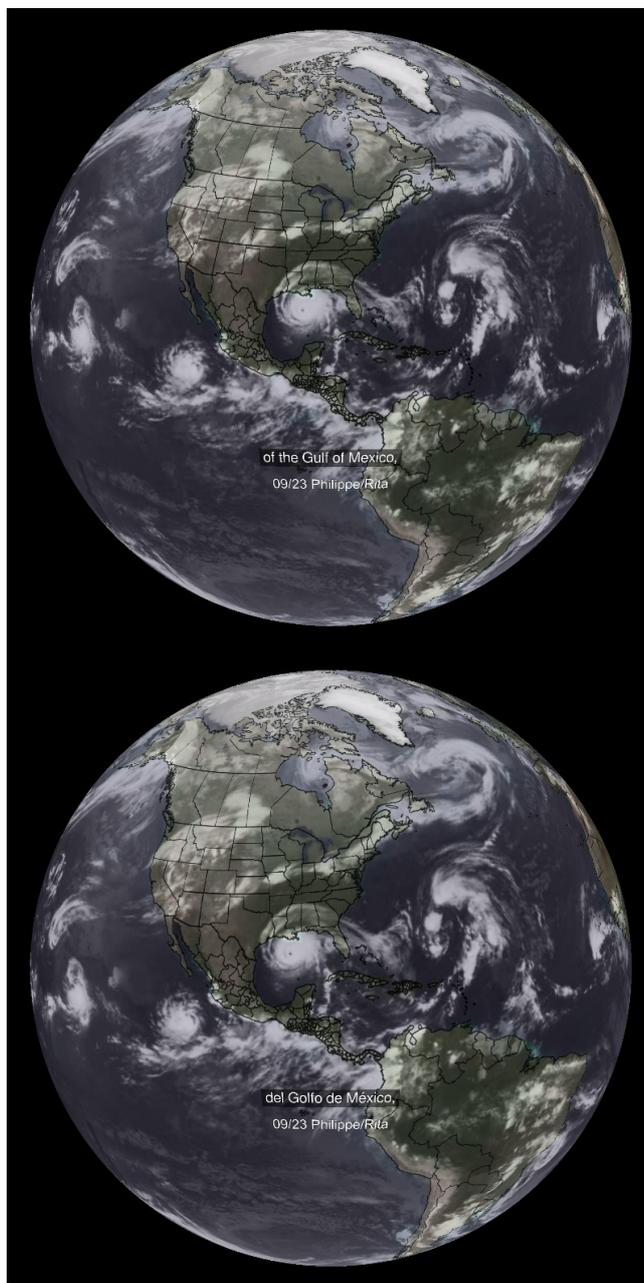
---

The purpose of adding labels to the sphere is to aid the visitor in understanding the dataset. Consider using pictographs for scales because they have been found to be intuitive and beneficial for visitor understanding. Also, using country or city labels can help visitors orient themselves in a global context. Another suggestion is to use vertical temperature colorbars because that is how most visitors are accustomed to reading thermometers. Size, orientation and placement of colorbars and legends are important for improving the visitor's understanding of the dataset.

There is a lot of flexibility with the color bars. They are inserted into the playlist.sos file as a pip. Using this function, you can not only set the position, size and transparency, but also when the color bar appears, how long it stays visible, and how quickly it fades in and out. The color bars can be any common image format such as GIF, JPEG, PNG, TIF, etc. Color bars are generally named color\_bar in order to keep all of the various images in the file separate.

### Captions

Captions can be displayed on SOS from standard SupRip Subtitle (SRT) files that are read in to SOS directly. An SRT file is an industry standard file that contains a sequential set of subtitles with their start and end timecodes. Captions provide text for hard of hearing (the left image below) or non-English speaking audiences (the right image below) and can also be used to show movies in AutoRun with the sound muted. [Text formatting](#) may also be fully customized in the [SOS Visual Playlist Editor](#). NOAA datasets will be getting closed captions added to them for your use, but will not be shown by default. The captions may then be turned on in the Remote App or displayed all the time by setting the visibility in your presentations.



### Media Folder

## Support: Dataset Manual

---

Each NOAA provided dataset has a “media” folder that contains a rendered global view of the dataset as well as two thumbnails. These thumbnails are used in the SOS Stream GUI as well as the iPad. The media folder is not required for site-custom datasets, but if you don’t provide a thumbnail, then a question mark icon will show up in the software. In order for custom thumbnails to show up in the software for your datasets, create a folder in your dataset folder called media. The media folder should contain two images in the .jpg format with the following names and resolutions:

```
thumbnail_small.jpg (128x128)
thumbnail_big.jpg (800x800)
```

## Differences in Playlist Files

A point of confusion for many SOS users is the difference between a presentation playlist and a playlist.sos file. While all of the same attributes can be used in both, they serve two distinct purposes. A playlist.sos file can be thought of as a configuration file for a dataset. It contains the name, the path to the data to be displayed, and any other settings you wish. Each playlist.sos file should be stored with the content pieces it refers to (though this isn’t required) and should reference just one dataset. A presentation playlist groups multiple datasets into a list that can be used for a presentation. Presentation playlists have to end with the extension .sos and can be named anything as long as there are no spaces or special characters in the name. All presentation playlists should be stored in the sosrc directory in the home folder for each user. You can read more about presentation playlists in the [Presentation Manual](#).

There are two places where you can modify a dataset: in your presentation playlist (such as weather\_overview.sos) or in the playlist.sos file. When you modify a dataset in a presentation playlist, the changes will only apply in that specific playlist. If you modify a playlist.sos file, then every presentation playlist that points to that playlist.sos file will reflect those changes. The playlist.sos files that you create should be considered the master copy. Note: Changes made to playlist.sos files that are provided by NOAA will be overwritten every week when the sync with NOAA FTP server occurs. If you want to make changes to those playlist.sos files, first copy them into your site-custom folder.

## Dataset playlist.sos Files

There is a fairly strict format that must be followed within the playlist.sos file. Any specifications that are made in the playlist.sos will be default settings for how that dataset is displayed. Here is an example of what is contained in the playlist.sos file for the Blue Marble dataset:

```
name = Blue Marble
data = 4096.jpg
fps = 40
tiltx = 23.5
category = land
catalog_url = http://sos.noaa.gov/Datasets/dataset.php?id=82
majorcategory = Land
```

At a bare minimum, you have to include the name and data (or layerdata) attributes. Everything else is optional. For site-custom datasets created by the site, there are some attributes that don’t apply and also a few that are just for site-custom datasets. The playlist.sos files can be created with the Visual Playlist Editor or written by hand using a program like gedit or Notepad. For a complete listing of attributes available for the playlist.sos file, see the [Playlist Reference Guide](#). The SOS software will ignore any lines that begin with a pound sign (#). This is a great way to temporarily ignore some attributes or to add comments.

Because all of the content pieces should be stored in the same folder as the playlist.sos file, it is not necessary to include the entire path to the files. You only need to include the data name. For example, to include labels all you need to type is `label = labels.txt`. If the data is stored in another location, then the path needs to be included. For example, `label = /shared/sos/media/atmosphere/dataset/labels.txt`

There can be multiple playlist.sos files in one folder for different versions of the dataset. The file names simply need to start with playlist and end with .sos and **there must be one file that is named playlist.sos**. For example, you could have playlist.sos, playlist\_with\_audio.sos, and playlist\_extra\_labels.sos all in the same folder. If you don’t have a playlist.sos file then none of the variations will show up in the data catalog on the iPad.

The “include” lines used in presentation playlists should not be used in a playlist.sos file, since the purpose of the playlist.sos file is to describe a single self-contained dataset with optional layers, PIPs, etc. Only presentation playlists should use the include attribute.

## A playlist.sos Example



In this blue\_marble example, there are two playlist.sos files in the folder for the blue\_marble dataset, playlist.sos and playlist\_audio.sos. Both playlists point to the same data, and the only difference is that one includes audio and a timer and the other doesn't. Notice that the audio files have been put into their own folder. If there are multiple audio files or PIPs, a folder can be created in the dataset folder that contains those files. While this isn't required, it helps to keep the folder uncluttered.

When files that are referenced in the playlist.sos file aren't in the same directory as the playlist.sos, the path to the file needs to be included. Take note in the playlist\_audio.sos file how the audio points to audio/BlueMarble.mp3 since the mp3 file isn't in the same directory as the playlist.sos. Either relative paths (audio/BlueMarble.mp3) or full paths (/shared/sos/media/land/blue\_marble/blue\_marble/audio/BlueMarble.mp3) can be used in the playlist.sos files. Be careful to avoid typos, as the dataset won't work if anything is wrong!

## Basic Options in the Playlist

You can optimize how a dataset is displayed by understanding all of the attributes that are available to you in the playlist.sos files. You can do much more than simply display the dataset. The [Visual Playlist Editor](#) can be used to create both presentation playlists and playlist.sos files and gives you the ability to set all the attributes that are available through an intuitive user interface. All of the attributes available for playlists can be found in the [Playlist Reference Guide](#).

### Attributes for Texture Datasets

For a texture dataset, there are only a few attributes that you need to consider. When a texture dataset is initially loaded on the sphere, you can set whether you want it to rotate immediately or only after play is pressed. The attribute `animate` in the playlist controls this. If `animate` is not included in the playlist, then the default is for the dataset to automatically start rotating. `animate` can be set to either 0 or 1. 0 will prevent the dataset from animating until play is pressed, and 1 will cause the dataset to start rotating immediately when loaded. For a texture, `fps` is used to define how fast the dataset will rotate, while for a time series, it defines the animation rate. Another common attribute used with textures is the tilt option. For instance, we have our Earth textures set to load at a 23.5° tilt to resemble the Earth's actual tilt. This is also useful if you are loading a dataset that highlights the poles, which are hard to see if there is no tilt. To set the tilt, set `tiltx`, `tilty`, and `tiltz` to the number of degrees that you want each axis tilted. The tilt can be positive or negative.

### Attributes for Time Series Datasets

For a time series, you have all of the attributes mentioned for the texture, plus many more. Rather than causing a dataset to rotate, `animate` causes a time series to start animating, but the functionality is the same. The default is for the dataset to start animating immediately. When a presentation is docent-led, it is often helpful to have the time series animate only after play has been pressed. This gives the docent time to provide background information about the dataset and explain what is going to happen. (In Autorun mode "animate" is automatically set to 1 regardless of what is in the playlist.) Another option is to set `firstdwell`, which is an amount of time that the system lingers on the first frame before animating. The default is zero seconds. The time is listed in milliseconds, so `firstdwell = 4000` will dwell on the first frame for 4 seconds. You can also dwell on the last frame by setting `lastdwell`. When `lastdwell` is not set, the dataset loops continuously without pausing. Especially with model data, it is nice to set `lastdwell` so that the audience can get a good look at the last frame before the dataset loops again.

With particularly long datasets it's sometimes nice to show only a piece of the dataset. You can do that by setting the `startframe` and `endframe` to the frame numbers that you want to start and end on. An example of when to use this would be if you just want to show a loop of Hurricane Katrina, not the entire 2005 season. You would use the 2005 Hurricane dataset, but set the `startframe` and `endframe` so that only the piece of the dataset when Hurricane Katrina was visible is shown. The `endframe` can be a negative number, which counts back from the end. Another way to shorten a dataset is to set the `skip` option, which allows you to set a skip

factor. When `skip` is set to one, it skips every other image, and when it's set two, it plays every third image.

To stop an animation, you can simply pause a dataset with the remote. But if you want to stop on an exact frame, then you should use `stopframe` in the playlist. This lets you set an exact frame that you want the animation to stop on and start animating again after you press play. This is a good feature to use with model data when you want to look at a particular year. To proceed past the frame that you stopped on, you must advance one frame and then press play.

Another option that you have for times series is to not only have them animating, but also rotating. For example, the default for the Indian Ocean Tsunami dataset is for the base image to stay stationary while the waves propagate across the ocean. This means that only the audience standing in front of the Indian Ocean can see the waves. When `zrotationenable` is set to 1, then the dataset will rotate about its z axis while it animates. You can also use `zfps` and `zrotationangle` to set the frames per second rate for the dataset and the angle at which the dataset rotates. Make sure that you set your `zfps` at a rate that allows your audience to still grasp what they are looking at before it rotates out of site. For especially busy animations, it could be distracting to the audience to see both the animation and the rotation.

### Autorun Datasets

There are also some functions in the playlist that should be specified when using Autorun. Autorun cycles through the datasets in a playlist automatically, showing each dataset for a specific amount of time. You can specify the amount of time each dataset is shown by setting `timer` to the number of seconds desired. If this is not specified, then each dataset is shown for 180 seconds. If `timer` is specified and you are not showing the playlist in Autorun mode, then `timer` will be ignored. It's important to use `timer` when you also have accompanying audio tracks so that the dataset is shown for the length of the audio track. You will want to make sure that the audio is synced with the playlist. You can set audio for each dataset by specifying the desired track with the "audio" attribute. The audio tracks must be compatible with the Linux mplayer such as .mp3, .mp4, .wav, or .ogg. Audio tracks are available from NOAA for a limited number of datasets. They provide a good way to give your audience information when a docent is not available.

In order to restart a dataset, including the audio and any PIPs that have been added, the `duration` attribute needs to be set for the length of the dataset in seconds. If `duration` is not set, the dataset will loop indefinitely, but the audio and the pips will not loop.

## Picture In a Picture

Picture in a Picture (PIP) allows you to display single pictures (any of the previously mentioned image formats works), an image sequence, or videos (MPEG4 only) on top of any dataset. This feature can be used to display any image, but is commonly used to display colorbars, charts and graphs, logos, and other images that supply supplemental information. Images that you are going to use as PIPs can be stored in the dataset folder that they go with.

When used for a colorbar, a PIP can help label a dataset, as seen at right. It is not recommended to embed colorbars or other supplement imagery into the maps that you create. Leave them as additional image files that can be added in the playlist.sos file. This gives the user complete control over the position and size of the PIP and gives presenters the ability to turn them off on the fly using the SOS Remote app.

A PIP can also be used to provide a close-up view of a region or give the viewer additional context for what they are seeing. In the example at right, the underlying dataset shows the tracks of elephant seals in red, and the PIP is a picture of actual elephant seals. Multiple PIPs can be shown at the same time, or staggered to create a slideshow effect. Make sure to consider the placement of the PIP in order to not block information in the underlying dataset, especially if the PIP is displayed for an extended period of time.

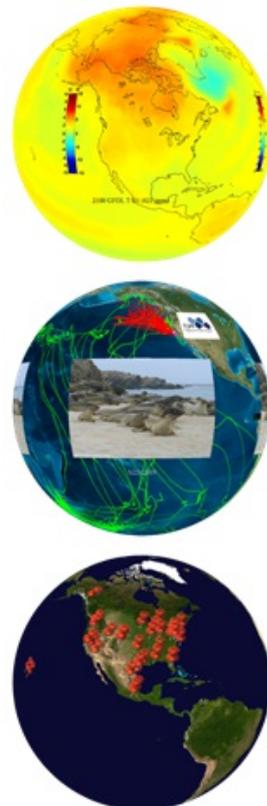
By using PIPs that are PNG's with a transparent background, many different shapes can be projected on the sphere with the underlying dataset as a background. PIPs can be set to display in specific locations on the sphere as markers, as seen at right. Here each pushpin is a PIP that identifies the location of a SOS installation.

Standard PIPs shouldn't be any larger than 1024x1024 in resolution size. Be aware that overlapping and warping can occur if the display size of a PIP is set too large. Make sure to test each dataset before distributing it to other sites, checking the PIP size, placement and timing. PIPs can also be MPEG4 files or image sequences.

Each PIP must be specified with the `pip` attribute. You can point to an image, time series, an image url (for example: `http://example.com/image.jpg`), or a live stream (for example: `rtsp://server_name/stream_name.sdp`). All of the following modifying PIP attributes must then be listed below that PIP. To add another PIP, simply add another line that starts with `pip` and then list the modifying attributes in the lines below it. You can add as many PIPs as you want.

### PIP Style

There are three different styles for PIPs: `projector`, `room`, and `globe`. `projector` is the default, where the PIP is replicated four times and placed with the default position centered in front of each projector. As the imagery rotates, the PIP remains stationary in `pipstyle =`



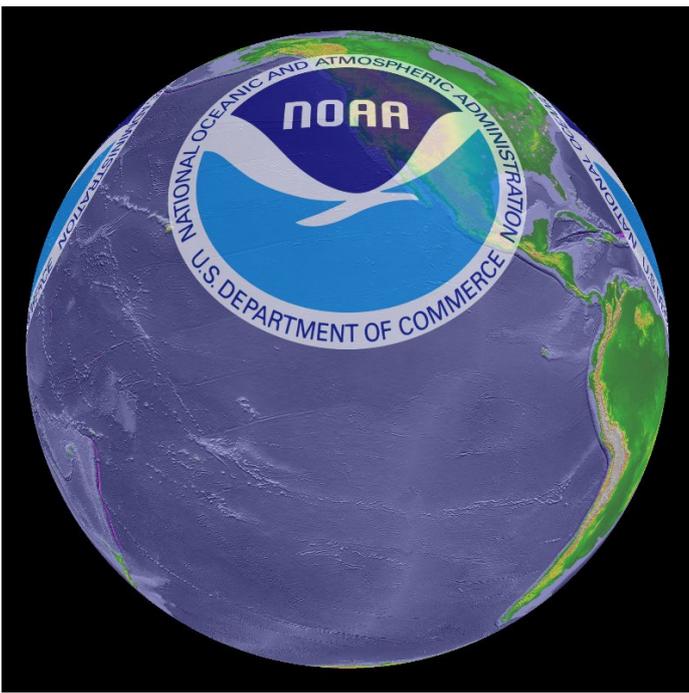
**projector** . A **pipstyle** of **globe** places one PIP on the globe, by default with a latitude and longitude of 0,0. As the sphere is tilted and rotated, this PIP moves with the globe. This allows you to use PIPs as geo-referenced markers. The center of the image is placed at the specified latitude and longitude. A **pipstyle** of **room** places one PIP on the globe, by default with a latitude and longitude of 0,0. As the sphere is tilted and rotated, this pip remains stationary relative to the room, with the sphere data sliding underneath it. See **Orientation of Data** to figure out where 0,0 is set in your room.

### PIP Timing

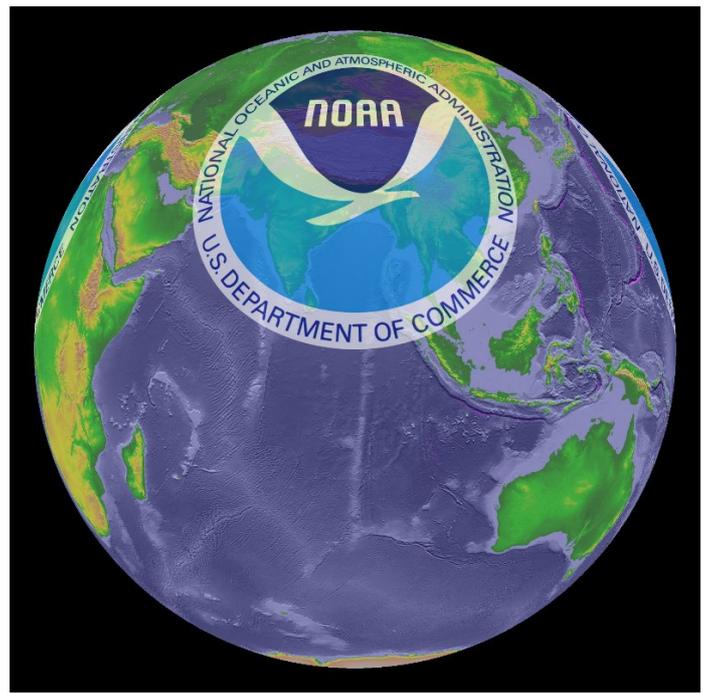
The **piptimer** attribute has to be set (in seconds) so that the system knows how long to display the PIP. If the **piptimer** attribute is set to 0, then the PIP will be displayed for the duration of the dataset, which is the default. You can delay the appearance of a PIP by using **pipdelay** , which is in seconds. Rather than having the PIPs appear abruptly, you can use the **pipfadein** and **pipfadeout** to fade the PIP in and out in a specified number of seconds. The time to fade in and out a PIP is included from the total amount of time allotted for the **piptimer** . By default, a series of PIPs will play through only once. You can set **duration** to a given number of seconds to restart the underlying dataset and the PIPs.

### PIP Size

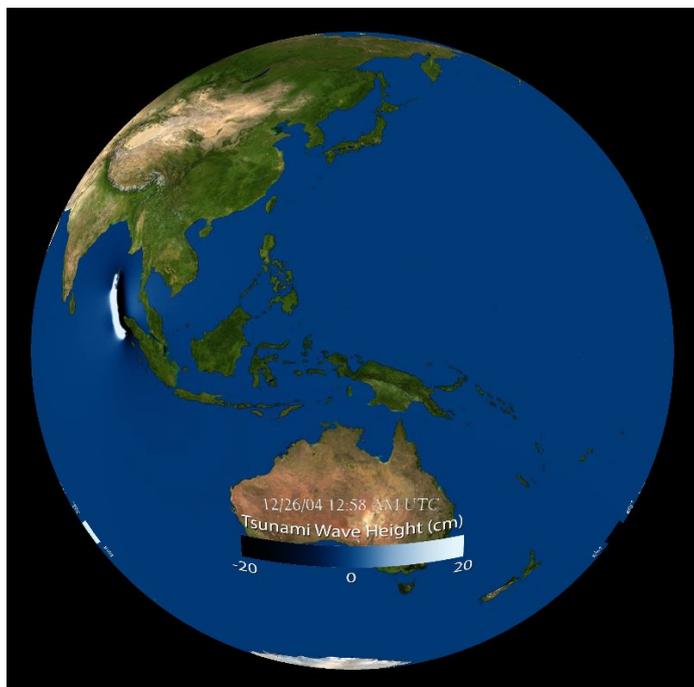
In order for the PIP to be an appropriate size for the sphere and in the proper proportions, you have to set the **pipwidth** and **pipheight** . The width and height are measured in degrees latitude and longitude. If you set just the height or the width, the software will automatically scale the image. If you are using **pipstyle = projector** you won't want to make your PIP more than 90 degrees wide because the PIP appears four times (once for each projector) and it will start to overlap. In addition to the PIP size, you will also need to determine where you want it displayed on the sphere. If nothing is specified, then the PIP will appear in the middle of each of the projector views. To adjust the position of the PIP, use **pipvertical** and **piphorizontal** . Both of these are in degrees. **pipvertical** is the vertical position of the image relative to the equator, with positive degrees above the equator. Be careful as you move the PIP up and down with **pipvertical** because the image follows the lines of longitude and becomes warped at the poles. The horizontal position is relative to the center of the projector, with positive degrees east of the project. An alternative to using **pipvertical** and **piphorizontal** is to use **pipcoords** , which is set in degrees latitude and longitude. The benefit of using **pipcoords** is that there is no warping of the images, even near the poles. **pipcoords** is also used with **pipstyle = room** and **globe** to position the PIP.



Above, the logo was added as a PIP with the positioning set by using **pipcoords = 30,0** . Notice how the logo maintained it's shape.



In this case, the logo was added as a PIP with the positioning set by using **pipvertical = 30** . Notice how the logo warped along the lines of longitude.



Above, the colorbar was added as a PIP with the positioning set by using `pipcoords = -35,0`. Notice how the logo maintained its shape, which, for something like a colorbar actually ends up looking curved.

In this case, the colorbar was added as a PIP with the positioning set by using `pipvertical = -35`. Notice how the colorbar maintained its shape along the lines of latitude so that it still looks straight.

When a PIP is a mp4 file, the default playback speed is the frame rate of the dataset on which it is overlaid. If you want to control the frame rate of the PIP, then use `pipfps` to set a new frame rate. The final option to set with a PIP is `pipalpha`, which lets you adjust the transparency. If not specified, the pip shows up opaque. If you don't want your pip to completely block the underlying image you can adjust the opacity of the image from 0, which is completely transparent to 1, which is completely opaque.

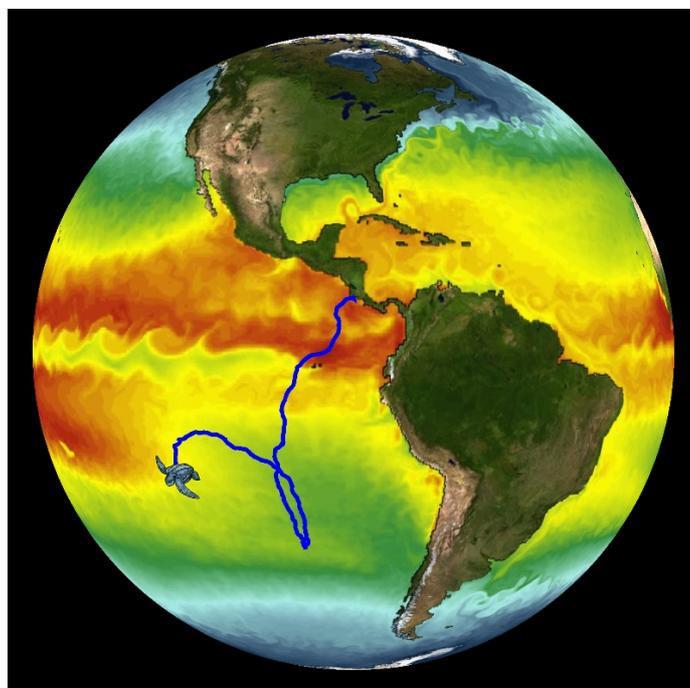
## Text PIP

A Text PIP is a special kind of PIP that displays text only. A Text PIP has all the same attribute specifications as a normal PIP, such as width, opacity, and fadein time. Rather than using an image editing program to create text and save it as an image file for a PIP, you can use the [SOS Visual Playlist Editor](#) and enter text directly into the [PIP Text Editor](#) and save the Text PIP to your SOS dataset. A Text PIP gets written to an html file. You should only use the PIP Text Editor to create and edit Text PIP files, and you should not create or edit it by hand. The following image shows an example of text in multiple languages that was easily created using Text PIPs:



## Moving PIP

Each type of PIP (image, image directory, movie, text) can be given a simple path file that contains coordinate locations that indicate how to automatically move the PIP on the sphere as the dataset is animating. The path file is a simple comma separated value file (.csv file format) that contains a list of increasing frame numbers, each with a latitude and longitude value. As the dataset is animating, the PIP will be moved to the location specified in the file that corresponds to the current frame being displayed on SOS. Option to render a line path that follows the moving PIP is provided and is turned ON by default. The moving PIP feature makes it easy to show animal migrations (example below shows leatherback sea turtle track), hurricane tracks, etc. on SOS without having to render a moving object into the underlying global movie or image data. Use the [SOS Visual Playlist Editor's PIP Path Editor](#) to associate a csv file with your PIP. For a full example on how to create a Moving PIP, including detailed documentation for creating the csv path file, please see the [Moving PIP Manual](#) document.



## Shared PIP

## Support: Dataset Manual

A Shared PIP is a special PIP that can display continuously over multiple clips in a playlist or between playlists. Such a PIP is useful during a SphereCast, for example, where you might have a video feed of a remote presenter on the sphere talking about a series of clips in the playlist. The Shared PIP stays active until you explicitly stop it. In its current implementation, a Shared PIP supports Live Video PIPs as described in the next section and static PIP images.

A Shared PIP is set up through the Shared PIP dialog box, located in the SOS Stream GUI's Utilities menu. In the Shared PIP dialog box that pops up, you specify a Live Video PIP in the same way as detailed in the section below. For a static PIP image, you can click the "Browse" button to locate the image on your SOS computer. The following PIP attributes work with a Shared PIP: `pip`, `pipstyle`, `pipwidth`, `pipheight`, `pipcoords`, `piphorizontal`, `pipvertical`, `pipalpha`.

Once you press "Start", the PIP will show up on the sphere, and will remain active even if you switch to a different clip. Press "Stop" to delete the Shared PIP.

Shared PIP Keywords

pip:    
Name of pip file (\*required)  
Ex. rtsp://stream\_name.webcam\_image.png

pipstyle:   
Ex. room, globe, projector

pipwidth:   
Width of pip in degrees

pipcoords:   
Location of pip on the sphere  
Ex. latitude,longitude (no spaces)

other:   
Specify pipalpha and/or pipheight keywords  
(separate with semicolon and no spaces)  
ex: pipalpha=0.6 or pipalpha=0.6;pipheight=35

## Live Video PIP

A Live Video PIP is a PIP that contains a video that is streaming either from a webcam connected to a local SOS computer, or from an RTSP stream. RTSP (real time streaming protocol) is an application-level protocol that controls the delivery of a real-time data stream, such as live audio and video. This feature may be useful for sites receiving SphereCasts, where, instead of needing a separate display in the SOS room to show video of the host site's presenter, the video can be shown directly onto the sphere. This feature may also be useful if a site wants to show a real-time video feed of a remote presenter onto their sphere for a particular in-house presentation.

### Incorporating in a Playlist

A Live Video PIP is specified in a presentation playlist file (or in a clip's playlist.sos file) similar to how a normal PIP is specified. For example, if you are showing a tsunami dataset and you want to show a live video on the sphere of a remote presenter discussing the dataset (assuming the presenter is setup to broadcast an RTSP stream), one entry in your presentation playlist file might look like:

```
include = /shared/sos/media/oceans/japan_tsunami_waves/playlist.sos
rename = Japan Tsunami with Live Presenter
pip = rtsp://server_name/file.sdp
pipstyle = room
pipcoords = 0,135
pipwidth = 65
```

In the above example, "server\_name/file.sdp" would need to be replaced by the actual name of the remote presenter's RTSP stream.

If you are using a webcam attached to your SOS computer, simply type `webcam` for the `pip` attribute, as in: `pip = webcam`

The following PIP attributes work with a Live Video PIP: `pip`, `pipstyle`, `pipwidth`, `pipheight`, `pipcoords`, `piphorizontal`, `pipvertical`, `pipalpha`.

Once you select this clip via the iPad or SOS Stream GUI, the live stream should pop up on your sphere as a normal pip does (note, however, it may take a few extra seconds to a minute for the stream to show up on the sphere, depending on network speed etc.).

## Requirements

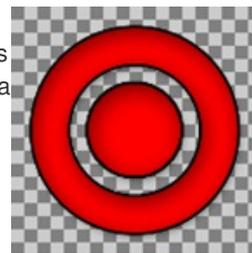
- An RTSP source to broadcast the remote presenter: There are various live video streaming solutions available. Currently, we use Apple's streaming QuickTime technology with the freely available QuickTime Broadcaster. Other streaming technologies that support RTSP may be used. If using QuickTime Broadcaster with the NOAA SOS video server for hosting a SphereCast, you would need to contact us in advance for setup instructions (see [http://sos.noaa.gov/Support/host\\_spherecasting.html](http://sos.noaa.gov/Support/host_spherecasting.html) for more information).
- A reasonably high-speed internet connection is required to send/receive a live video feed. We recommend a dedicated bandwidth of at least 1.5 MBits/sec, though a higher 3-4 MBits/sec is preferred

## Limitations

- The webcam currently does not support audio, and may exhibit a delay in frame rate overtime.
- Although RTSP supports both live data feeds and stored video/audio clips, in our current implementation, only live data feeds are supported for display in a PIP.
- If the live stream is stopped by the host while a Live Video PIP is being shown on the sphere, SOS Stream GUI will hang for about two minutes, and then it will resume normal activity. So, the best thing to do if you notice a live video stream is no longer working on the sphere is to wait for at least two minutes before using any controls on the iPad or SOS Stream GUI, otherwise, you may have to manually stop SOS and restart it.

## Annotation Icons

The SOS Remote app, through the annotation feature, gives presenters the ability to draw on the sphere and place icons on the sphere. There is a set of default icons that come with the SOS Remote app. In addition, sites have the ability to create custom icons. If you would like to create your own icons, use a transparent PNG with a minimum resolution of 256x256. To the right is an example of one of the default icons. Custom icons can either be specified for specific datasets, or made available in the default icon library.



### Dataset Specific

To add an icon to your dataset's playlist.sos file so that it shows up in the Icons dialog when you load the dataset, simply add an `icons = value` attribute/value pair to the dataset's playlist.sos file and place the icon in the dataset directory. Note that you can specify more than one icon by making a comma separated list with no spaces.

For example, if you create a satellite icon and a rocket icon and want to add those icons to your Blue Marble dataset, your Blue Marble playlist.sos file might look like this:

```
name = Blue Marble (23 degree tilt)
data = 4096.jpg
category = land
icons = satellite.png,rocket.png
```

In this case, the icon files should be placed in the same directory as the playlist.sos file. In other words, use relative paths when specifying the icons in the playlist.sos file. Once you load the dataset on SOS and then open the Icons dialog, the two icons you added will appear at the top of the list of available icons.

Another way to specify an icon is via the presentation playlist located in the sosrc directory. You specify the `icons = value` attribute/value pair for a clip here as well, however, the pathname of the icon file must be specified relative to the location of the clip's playlist.sos file. For example, if you have an icon called turtle.png located in your site-custom folder, and you would like to make this icon available with the Loggerhead Sea Turtle dataset, you can add that to your playlist as follows:

```
# Loggerhead Sea Turtle Tracks
include = /shared/sos/media/oceans/LoggerheadSeaTurtleTracks/playlist.sos
icons = ../../site-custom/turtle.png
```

### General Icons

Finally, if you have a general set of icons that you create and that your site may use often, you can add these icons to the default icon library so that they are automatically available with every dataset. To do this, simply add your icons to the directory `/shared/sos/etc/AnnotationIcons/`.

## Layers

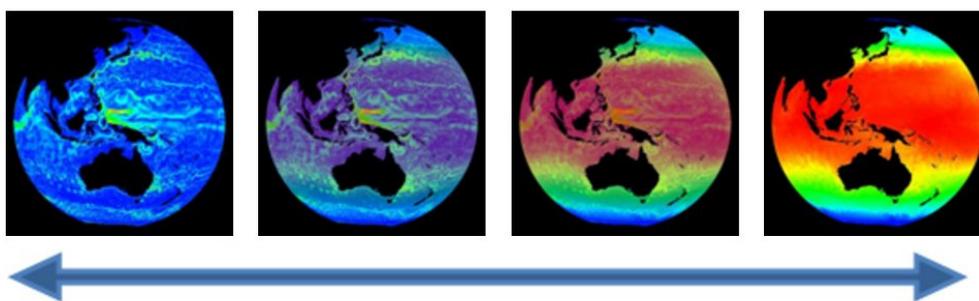
The layering capability in SOS allows presenters to dynamically turn layers on and off. A multi-layer display can be created either statically in the dataset playlist beforehand, or interactively using the SOS Remote. By using the Display Elements list in [SOS Remote](#), the user can toggle individual layers on and off, adjust the level of transparency of each layer, or delete a layer. Any labels/captions or PIPs associated with a clip are now also listed individually in the Display Elements list. These can be interactively manipulated like any other layer.

### Predefined Layers

A multi-layer dataset may be defined by using the `layer` attribute. Each use of a `layer = name` attribute/value pair within a playlist.sos file defines a new layer and specifies the name of the layer. The specified name of the layer is used to identify it in the layer table in SOS Remote's Layers tab. Each new layer specified appears visually on top of any previous layers.

The `layerdata` attribute is repeated for each layer to specify the corresponding data file for the layer. A layer defined this way may have a `layervisible = no` attribute/value pair defined to specify that the layer is not initially visible. A layer may also have a `layeralpha` attribute pair to further specify the initial opacity of the layer. An alpha value of 0.0 means that the layer is totally transparent, and 1.0 means the layer is totally opaque. A slider in the SOS Remote interface is available to interactively manipulate the opacity of each layer.

**Note:** For compatibility with versions of SOS prior to SOS Version 4.0, a default layer is created when the `data =` playlist attribute is seen in a playlist before the `layer =` attribute. The name of this default layer will be the same as the name of the dataset, given by the `name =` attribute.



### Orienting Layers

In order for layers to overlap properly, it is important to make sure that the maps are oriented identically. In the case where two layers have different center points, you can set `layereast`, `layerwest`, `layernorth`, and `layersouth`. These commands specify the geographic extent of the data within the layer. They specify the east and west edges of the data in degrees east longitude, and the north and south edges in degrees north latitude.

### Overlays

In addition, we have created new overlays which are located in the `/shared/sos/media/overlays` directory, and which will show up as a library category on SOS Stream GUI and on the iPad. You will find them accessible through a button on the presentation tab of the iPad. The overlays contain useful earth-related transparent layers (specified as clips in a standard playlist.sos file format) that can be used for both pre-programmed layering, as well as, interactive layering. An example of a layer that will be in this category is Country Borders. If a site wants to add more overlays for general use, they should be placed in the site-custom folder with a playlist.sos file that has the category defined as overlay. Examples of playlist.sos files for overlays can be found in the `/shared/sos/media/overlays` directory.

You can have your custom overlays appear in the Overlays dialog of the iPad app just as the NOAA-managed overlays appear for quick and dynamic layering. To do this:

1. In your playlist.sos file, add the following attribute/value pair (this is optional and allows your overlay to show up in the overlays category in SOS Stream GUI's Library menu): `category = overlays`
2. In your playlist.sos file, add the following attribute/value pair (this is what makes your overlay appear in the iPad app's Overlays dialog): `subcategory = Overlays`
3. On the SOS Computer's SOS Stream GUI application, select the Library menu > Update Library menu option to update the Data Catalog with your new overlay dataset. Once this is complete, on the iPad app's Settings tab, select the Update Now button, and your overlay dataset will appear in the Overlays dialog of the iPad.

### Using KML Data

SOS supports Keyhole Markup Language (KML) data in addition to the previously existing movie and image formats. KML is a popular specification and actively used with Google Earth for displaying data on a sphere. The initial SOS KML capability supports a limited set of the entire specification, which includes many of the commonly used KML features you would typically display in Google Earth. More information on KML itself can be found here: <https://developers.google.com/kml/documentation/kmlreference>

An SOS playlist can reference both KML and the compressed KMZ formats using the `layerdata` attribute.

#### Implementation Notes

Typically, KML files are used with Google Earth which allows users to display information on a virtual sphere similar to SOS. There are a couple of differences to be aware of. Google Earth has additional space around the sphere where legends, icons, or other meta information can be displayed. SOS has only the sphere for displaying data. By default, all ancillary information is displayed at point 0° North, 180° East. Each subsequent piece is staggered from this starting point. This is user configurable. Within the playlist, use `kmllegendxoffset` and `kmllegendyoffset` to specify a new location.

KML Placemarks or Icons referenced in KML may appear small on the sphere. Additional playlist parameters have been included to scale icon's to make them more visible on the sphere. Use `kmlplacemarkscale` to scale these features if necessary.

More information on these commands can be found in the [Playlist Reference Guide](#).

#### Special Notes for KML

Often, KML files reference remote data via a web address. KML files of this nature require SOS to have access to the internet to retrieve these files. Depending upon your network connectivity and the number of external links referenced in the KML file, the initial load may take some time. SOS will perform local caching of downloaded files and subsequent loads will perform faster.

## Support: Dataset Manual

---

*It is strongly recommended to test KML files prior to any presentation* to insure data is cached locally and the presentation is not delayed by waiting for remote files to be retrieved. When an SOS playlist references a KML dataset, SOS will parse the file and store any temporary or cache information in the system temporary directory. The default is `/tmp` on SOS systems.

### Limitations for KML

The SOS software does not support the entire KML specification. Here is a list of major items not currently supported in this release: Tours, Fly To, Features with 3 Dimensions, Resource Map, Model's, Regions. If KML data or KML data isn't displaying correctly, please contact the SOS support team and include the problem KML file in your message.

You cannot have multiple KML layers defined within a single playlist item because we do not support time matching capabilities between various KML files. Future versions should allow multiple static KML files.

## Using WMS Data

SOS supports loading imagery directly from the Open GeoSpatial Consortium (OGC) Web Mapping Service (WMS). This feature requires an internet connection and will not work unless the SOS system has access to the internet and the referenced WMS Server. More information on the WMS standard is available here: <http://www.opengeospatial.org/standards/wms>

A WMS provides a service allowing users to request data through URLs using specific key value pairs defining terms such as the width, height, image type, etc... A unique feature of the WMS standard allows users to request subsets of imagery by defining a bounding box using a lower left and upper right latitude and longitude coordinates. The combination of these features allows users to host very large high resolution imagery and users can request smaller versions or subsets of the original imagery. SOS takes advantage of this functionality through the magnifying glass, allowing users to see more detail as you increase the zoom level on the sphere.

A typical WMS URL, will look like the following:

```
http://neowms.sci.gsfc.nasa.gov/wms/wms?version=1.3.0&service=WMS&REQUEST=GetMap&LAYERS=MODAL2_D_CLD_CI&CRS=CRS:84&FORMAT=image/png&HEIGHT=1800&WIDTH=3600&TRANSPARENT=TRUE&BBOX=-180.0,-90.0,180.0,90.0&STYLES=rgb&STYLE=
```

To use this URL with SOS, you would specify data as the following (note: only the `layerdata` attribute may be used to specify WMS data):

```
#WMS Data Example
layerdata=//WMS//http://neowms.sci.gsfc.nasa.gov/wms/wms?version=1.3.0&service=WMS&REQUEST=GetMap&LAYERS=MODAL2_D_CLD_CI&CRS=CRS:84&FORMAT=image/png&HEIGHT=<IMAGE_HEIGHT>&WIDTH=<IMAGE_WIDTH>&TRANSPARENT=TRUE&BBOX=<BOUNDING_BOX>&STYLES=rgb&STYLE=
```

Four (4) things have changed:

```
//WMS//
HEIGHT=<image+height>
WIDTH=<image_width>
BBOX=<bounding_box>
```

All 4 items are required in order for WMS data to work correctly with SOS. The first item indicates the following path to data is a dynamic WMS URL. The last 3 are placeholders for dynamic fields that change while in use for SOS. For each WMS URL used within SOS, these values must be replaced exactly as above. SOS will automatically replace these values when loading the data.

A tutorial specifically for using WMS with SOS can be found here: <http://sos.noaa.gov/Docs/WMS-tutorial.pdf>.

### Special Notes for WMS

All WMS URL's are remote data and require SOS to have access to the internet to retrieve the corresponding imagery. Depending upon your network connectivity and the performance of the remote WMS server, the initial load may take some time. SOS will perform local caching of downloaded files and subsequent loading will perform faster.

*It is strongly recommended to test WMS Playlists prior to any presentation* to insure data is cached locally and the presentation is not delayed by waiting for remote files to be retrieved.

When SOS playlist references a WMS dataset, SOS will retrieve and store any temporary zoom files or cache information in the system temporary directory. The default is `/tmp` on SOS Systems.

### Limitations for WMS

With the SOS magnifying glass enabled, SOS will determine a bounding box for the area currently under view and dynamically retrieve and load that image. A bounding box cannot be determined around either the North or South Pole. The nearest image that does not cross the pole will be used.

## Real-time Datasets

There is a collection of over 40 real-time datasets that are provided by NOAA. Because these datasets tend to be quite large and internet speeds vary from site to site, the SOS software can be modified to adjust the number of real-time datasets downloaded. Typically, sites are set to download real-time data either every hour or every three hours. The frequency of the downloads and the amount of real-time datasets downloaded can be adjusted for each site. In `/shared/sos/media/playlists` are various real-time dataset playlists that vary from just a few datasets to all of the real-time datasets. You can also create your own playlist of the real-time datasets that your site is interested in using. A crontab is then used to keep all of the real-time datasets in your playlist up to date. For more information on the crontab, visit: [How to Set up Automatic Dataset Downloads](#).

## Tools to Create Datasets

Because Science On a Sphere® uses common image and video formats, you can use many tools to create and edit datasets. Some of the common tools used are Photoshop, FinalCut Pro, ImageMagick, GIMP, etc. You can use whatever you have available and are comfortable using. A program like FinalCut Pro can be used to add transitions, special effects and other computer graphic techniques. At a higher level, tools like IDL, AWIPS, McIDAS, and other image analysis applications are typically used to create imagery from scientific datasets. As an example, we have used AWIPS (Advanced Weather Information Processing System) to create images from numerical forecast models. A graphics designer can use a 3D modeling application, such as 3D Studio, to create advanced visualizations for SOS.

GIS programs such as ArcGIS or Quantum GIS can be used to create maps. A tutorial for creating SOS datasets with Quantum GIS is available here: [QGIS Tutorial](#).

## Notes about Site-Custom Datasets

There are many settings that can be included in the `playlist.sos` file. For a dataset added by a site into the site-custom folder, the `playlist.sos` file should contain the following elements:

```
name = My Custom Dataset
data = 4096.jpg
creator = My Museum
subcategory = land
keywords = land, Earth, My Museum
description = {{ A description for my dataset that will appear on the iPad. The description is allowed to span multiple lines and must use be enclosed with the braces characters. }}
```

The first two lines are required, and the remaining lines are all optional, but provide helpful information and allow you to more easily find your dataset. There is no need to add a category because all datasets in the site-custom folder are automatically tagged for the site-custom category. In order for a new dataset to show up in the site-custom library on the local SOS computer, make sure to update the library by clicking “Library” > “Update Library...” in the SOS Stream GUI. After you have updated the library, you have to click the “Update Now” button in the Settings tab of the iPad in order for it to show up on the iPad. It will not appear on the iPad if it is not stored in the site-custom folder on the computer. In the Data Catalog tab of the iPad you will be able to find the new dataset in the subcategory that you assigned under the site-custom main category. You can create as many different subcategories as you like within site-custom.

We have viewed many datasets and are happy to review your site-custom datasets and provide feedback as desired. Particularly for datasets that will be submitted for consideration for the SOS Catalog, we suggest providing a draft version to the SOS team for feedback.

## Submitting a Dataset to the SOS Catalog

## Support: Dataset Manual

If you decide to submit a dataset for consideration to be part of the SOS catalog, we will need all the content pieces and the playlist.sos file, along with a written description of the dataset, a list of notable features, and the credits for the dataset. All of this documentation is used to create an entry in the SOS Dataset Catalog, as seen at right. The written description should be a simple overview of the dataset that highlights the source of the data, whether it is modeled or measured, what it shows, and why it's important. It should be a non-technical description that is easily understood. The "Notable Features" is a bulleted summary of the highlights from the description that presenters can use when showing the dataset to viewers. The credits are used to fill in the column on the right-hand side. The credits listed are:

- Dataset Source
- Dataset Developer
- Dataset Visualization Developer
- Contact (optional)

These can be the same for each listing, or all different and can include links to the original sources. For examples, visit the [SOS Dataset Catalog](#).

## Visual Playlist Editor

The Visual Playlist Editor (VPLE) allows you to easily construct new datasets for your system. You simply add the layers, pips, title, and other settings that you want and when you save the dataset, everything you referenced is saved into a single folder along with an automatically generated playlist.sos file.

When creating a new dataset, make sure to specify only a subcategory. The major category will automatically be site-custom. If you forget to specify a subcategory for a dataset, then it will be put into an uncategorized subcategory in the site-custom category. You may also wish to include keywords and creator for your site-custom dataset, which will also be added to the SOS Data Catalog entry for your dataset as well. You can also add a description to your playlist.sos file that will show up on the iPad and the SOS kiosk.

When you use the VPLE to modify datasets in your presentation playlist, changes you make to the datasets will affect only the presentation playlist that you are editing and not the underlying playlist.sos files for each dataset. The playlist.sos file is the master copy of how the dataset is displayed and should not be edited in any of the NOAA-supplied datasets (the VPLE prevents that, by default). If you make changes in the playlist.sos files, the changes will appear in everyone's playlists and weekly NOAA dataset updates may overwrite your changes.

For more information on the VPLE, please refer to the [Visual Playlist Editor Manual](#).



The screenshot shows the Science On a Sphere website interface. At the top, it says "Science On a Sphere" and "National Oceanic and Atmospheric Administration". Below that is a search bar and navigation links for "Categories" and "Live Programs". The main content area is titled "Blue Marble" and includes a "Description" section with a large image of Earth. To the right of the description is a "Details" section with links for "Media Preview", "Interactive Sphere", "View Movie", and "Download Movie". Below that are "Categories" and "Keywords". At the bottom of the page, there are three small thumbnail images of Earth from different perspectives.