Portal Creation Guide

Data providers and other community stakeholders must currently invest operational and development resources producing search clients to serve a segment of the community. Earthdata Search portals reduce the overhead of providing a search interface similar to Earthdata Search to targeted users. It provides several useful features, including:

- Running a maintained, up-to-date instance of Earthdata Search without incurring the cost of hardware or administration
- The ability to present a limited set of collections to users, scoped to only those relevant to the target user community
- Presentation of additional logos, navigation, and page titles
- A simple URL parameter https://search.sit.earthdata.nasa.gov/search?portal=your-portal-name
  this is backwards compatible with the previously established bookmark-able links e.g. (https://search.earthdata.nasa.gov/portal/your-portal-name)
- Interoperability with the full Earthdata Search site, allowing users to optionally and easily search the full list of collections, and see their saved projects and past retrievals from the portal in Earthdata Search
- Inclusion of custom provider-developed configuration allowing for the manipulation of basic Earthdata Search user interface elements such as showing the only-granules and non-eosdis additional filter checkboxes

The remainder of this document will walk through the process of creating and configuring a portal.

Step 1: Reach out!

We're here to help, and we want to understand community needs so we can refine our features. Email Valerie Dixon (valerie.dixon@nasa.gov) to get started with your portal and get in touch with developers.

Step 2: Fork our GitHub repository

Important: The workflow for configuring portals is geared toward teams who have familiarity with basic development tools such as git. If this is not the case for you, please reach out to us and let us know; we may be able to assist you with basic configuration.

Because portal configurations are powerful enough to modify Earthdata Search's functionality in arbitrary ways, we must handle portal configuration through our Github project (https://github.com/nasa/earthdata-search/). This allows us to understand incoming changes, test existing portals against new features, and prevent insecure or malicious code from entering our operational environment.

To provide us configuration most efficiently, you will need to fork our GitHub repository.

Step 3: Startup a Local instance of Earthdata Search

Clone the code from your fork and follow the README to get a local instance of Earthdata Search set up.

Step 4: Configure your portal

To create your portal you will need to create a new directory in "/portals" with a unique name you want as your portal identifier, for this example we will use "example". You then will create the config.json file ("/portals/example/config.json").

A significant amount of configuration can be done through this config.json file. You can view an example configuration at https://github.com/nasa/earthdata-search/blob/main/portals/default/config.json or browse https://github.com/nasa/earthdata-search/tree/main/portals to see all portal configurations. Here is an explanation of the configuration:

Key	Data Type	Explanation
`features`	Object	Feature configuration
`advancedSearch`	Boolean	Enable/disable the advanced search feature
`authentication`	Boolean	Enable/disable all features that rely on authentication (logging in/projects). Setting to `false` will disable all features that rely on authentication.

`featureFacets`	Object	Feature Facet configuration
`showAvailableFromAwsCloud`	Boolean	Show/hide the Available from AWS Cloud feature facet
`showCustomizable`	Boolean	Show/hide the Customizable feature facet
`showMapImagery`	Boolean	Show/hide the Map Imagery feature facet
`footer`	Object	Footer configuration
`attributionText`	String	Text to display in the footer, example "NASA Official: ..."
`displayVersion`	Boolean	Display the current application version in the footer

`primaryLinks`	Array	Array of links to display on the left side of the footer
		
		This array is made up of objects with two values:
		
		``
		{
		    href: "The URL the link will follow",
		    title: "The viewable text of the link"
		}
		``

`secondaryLinks`	Array	Array of links to display on the right side of the footer. This uses the same objects as `primaryLinks`
`moreInfoUrl`	String (Optional)	This is the URL that users will be sent to when clicking on the logo image. If no value is provided clicking on the logo image will return the user to the EDSC portal home (/portal/example/search).
`pageTitle`	String	This will appear after the default portal's `pageTitle` in the HTML page title (see in the tab of the browser). For example if EDSC is the default portal then the page title will be "Earthdata Search :: <pageTitle>	Earthdata Search"
`parentConfig`	String	The parent config file for this portal. If your portal will be deployed to EOSDIS Earthdata Search this value should be "edsc". If not, set this to "default".
`portalBrowser`	Boolean	Toggle to set whether the portal will appear as a listed portal in the portal browser modal

query	Object (Optional)	These parameters are used to filter the list of collections exposed in the portal.
		
		Current supported values:
		
		• consortium (Array of strings)
		• dataCenter (String)
		• echoCollectionId (this narrows the results to a single collection, only used as an example) (String)
		• hasGranulesOrCwic (set to null to include all collections)
		• project (String)
		• tagKey (String or String array)
		
		One common pattern is to show all the datasets in a dataCenter. This example shows the ORNL DAAC portal configuration:
		
		`"query": {`
		`  "dataCenter": "ORNL_DAAC",`
		`  "hasGranulesOrCwic": null`
		`},`
		
		Likewise, for Science Project-focused portals, here is the AIRMOSS project portal:
		
		`"query": {`
		`  "project": "AirMOSS",`
		`  "hasGranulesOrCwic": null`
		`},`
		
		A more nuanced approach can be applied using tags in CMR. Here is an example from the Suborbital portal using the tagKey parameter (note that the tagKey can contain simple wildcards):
		
		`"query": {`
		`  "tagKey": ["gov.nasa.impact.*"],`
		`  "hasGranulesOrCwic": null`
		`},`
		
		If you need support of another CMR query parameter, please contact us.
		
		Refer to the CMR documentation for information on parameters:
		https://cmr.earthdata.nasa.gov/search/site/search_api_docs.html

title	Object	Object containing the primary and secondary title for the portal.
primary	String	The primary field in the title object will be displayed on the portal browser modal as the top identifier. It will also be displayed on the top left portal browser context sidebar header and on the leave portal button on the bottom of the collection search results while inside of a portal.
secondary	String	The secondary field in the title object will be displayed below the primary on the portal browser modal and in parentheses on the top left portal browser context sidebar header. It is meant to be the fullname of the project, endeavor, organization behind the creation of the portal.
ui	Object	UI configurations
showNonEosdisCheckbox	Boolean	Show/hide the "Include non-EOSDIS collections" checkbox
showOnlyGranulesCheckbox	Boolean	Show/hide the "Only include collections with granules" checkbox
showTophat	Boolean	Show/hide Tophat (**Note:** This field is used only for the `defaultPortal` in the deployed environment for EDSC, as such new portals cannot hide this UI field in the deployed env. It can however, be changed locally if you set the `defaultPortal` to the created portal in `overrideStatic.config.json`)

`

Inheriting Configuration Values

Portals inherit configuration values from a parent configuration. See the "parentConfig" value above. This allows you to use all of a parent's config values and keep your portal configuration minimal if your portal needs most of the same features as the parent configuration.

Adding Portal Logo

To add a new logo image, which are displayed in the left toolbar to establish the search context or on the the portal browser modal (optionally. see portalBrowser boolean config field).

Save your image into the *./portals/myportal/images/* directory and **name this file *logo.png***

Custom Logo Best Practices:

- Images will be displayed at a maximum of 75px tall by 75px wide. Logos should be trimmed to fit within those dimensions.
- It is highly encouraged that images be provided at twice their displayed size (50px x 114px maximum) for better display on high resolution screens.
- **Images must be named *logo.png***

Adding Portal to the index.js list

In the */portals* directories you will need to add the name of your portal to the portals list in the index.js file. The string you enter should be the same as the name of the directory for your portal. Portals should be added in alphabetical order.

```
portals [
   'above',
     ...
   'your-portal-example'
]
```

Step 5: Incorporating Changes into Earthdata Search

When you have made and tested your changes, push them to your forked copy of the Earthdata Search repository and issue a pull request into the main repository using the pull request template. Our developers will review and incorporate the changes as soon as possible.

Our normal progression to operations is as follows:

1. When a pull request is merged, it starts a build. Within a few minutes of the build becoming green, those changes get pushed to our SIT environment for testing: https://search.sit.earthdata.nasa.gov
2. Every other Wednesday, our current UAT environment (https://search.uat.earthdata.nasa.gov) is deployed to our operational environment (https://search.earthdata.nasa.gov) and our SIT environment from the previous Monday is deployed to our UAT environment.

So the progression from a pull request being accepted to it becoming an operational change takes about a month, during which it is available in testing environments.

While we would prefer to stick to this schedule wherever possible, we understand you may have differing schedule needs. If you need it to reach operations sooner, please indicate the desired date on the pull request and we will do our best to accommodate.