Earth System Grid Federation
On-Demand IDX Data Server

Mid-Project Report - May, 2015

Cameron Christensen - Giorgio Scorzelli - Valerio Pascucci - Peer-Timo Bremer

Overview

The on-demand IDX data server is designed to provide streaming hierarchical versions of equivalent NetCDF climate data volumes in a user-directed manner such that specific timestep fields are converted just-in-time. This permits the bulk of the data to remain on the server and facilitates interactive analysis and visualization by immediately sending results for specific data requests. Initial conversions are cached for future use, amortizing the cost of future requests. This is especially important for large domain data.

Figure 1 shows the current design of the overall system. Beginning at the ESGF search page (top-left of the diagram), the user can download a ViSUS configuration file describing how to load the selected climate dataset. When the user selects a dataset its corresponding IDX metadata is created and registered with an associated ViSUS server. This configuration will contain references to the multiple volumes that are part of the same climate model. The data can be loaded in a ViSUS client or compatible component. Available datasets can also be listed directly from the associated server.

Once the user has the URL the dataset can be opened from any IDX-compatible client, such as UV-CDAT (top-right). Its hierarchical natures allows coarse resolution data to be streamed very quickly, providing a preview of the final data and facilitating interactivity. When data is requested, a remote query is made to the ViSUS server, which checks if the data already exists in the cache. If so, it sends the cached data immediately. Otherwise, it calls the climate data converter service which converts the data and returns. After conversion the data is available in the cache and a repeated attempt to access the data will succeed. In the following we describe the details of the system, the current state of its implementation, and our next steps.

Data Conversion

The IDX data format utilizes an hierarchical z-order to facilitate fast loading of coarse resolution data as well as better spatial locality for more efficient sub-region reads. A copy of the original climate data is reordered but not modified in any way. Loading the full resolution data provides identical results to loading the original NetCDF volume.

An IDX dataset will be requested the same way as any other data, using the ESGF search page or advanced query interface. The following paragraphs describe the steps taken when an IDX version of a dataset is requested.

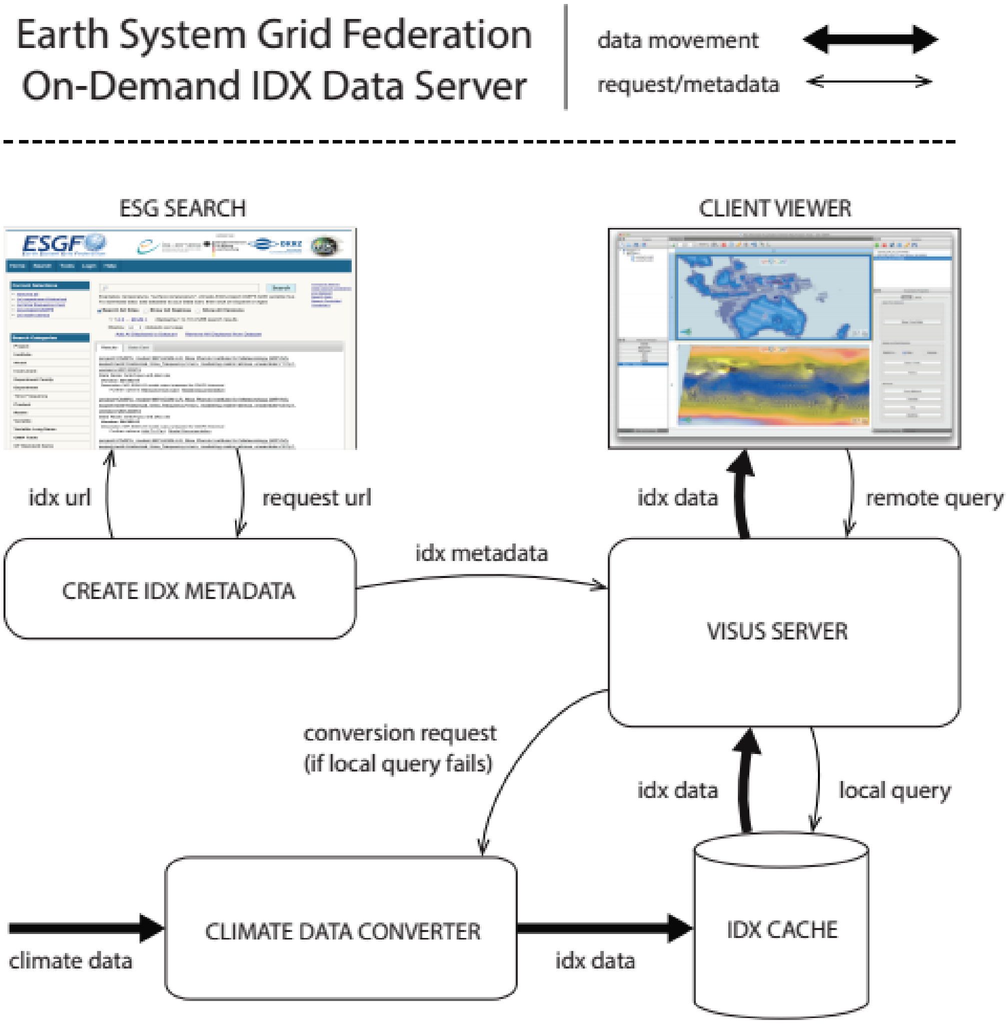

Figure 1. Current design of the interface between ESGF data provider node and ViSUS streaming data distribution server. A web request (top-left) allows to generate a configuration file that is added to the ViSUS server and downloaded to the client for reference in the bookmarks. A ViSUS client (top-right) can refer to the downloaded configuration to initiate specific requests to fields and time steps to visualize. The ViSUS server back-end (bottom) checks for each request if the data is available and, if not, it starts a process converting data at the smallest granularity possible.

Creating IDX Metadata

The IDX metadata corresponding to a given climate dataset is created the first time a climate dataset is requested, and retrieved from the cache for each successive load. The metadata is never removed so this is a one time operation.

IDX metadata creation determines the IDX volumes based on their description in a CDAT xml metadata file (or directly from a NetCDF file). The process only takes a fraction of a second. If not available, the CDAT xml is created by the ***cdscan*** program included with the UV-CDAT tools. Though it can take some time for *cdscan* to process all input files for a given dataset, it is a one time process and the results will be saved for any future use. There has even been discussion about including this broadly useful metadata as a standard component of all climate volumes.

Registering Metadata

The IDX metadata for the newly created volumes is registered with an associated ViSUS Server that will perform the conversion of the data as it is requested. Creating the IDX metadata is also a one-time operation.

Data access URLs

The URLs pointing to the newly-created IDX volumes are returned by the ESGF web page as a downloadable configuration file. The user will load the configuration using the client viewer application to retrieve data from the remote server. A user can also load a URLs directly but this is not a recommended approach.

Client Viewer

Any application can be enabled to read data from the ViSUS Server. This functionality has been previously integrated with UV-CDAT, and IDX plugins exist for both Paraview and VisIt. In the current tests we are focusing on the extensions around the server and testing the results with the native ViSUS client.

On-Demand Conversion

When a request for data is received by the ViSUS Server, the server first checks to see if the data is available in the local cache. If not found, the server issues a request to convert the smallest portion of data that contains the region of the desired field at the given timestep. This data is read from the ESGF archive and written to the local cache. The process takes less than a second for most requests in the datasets currently tested. More details can be found in the Conversion section below.

Cached Data

Once data has been converted it remains in the local cache indefinitely. Therefore future requests for the same data will be handled even more efficiently, amortizing the cost of the original conversion.

We plan to add functionality that allows removing data automatically from the cache when it becomes full. We will first adopt a least recently used policy. The IDX metadata is never removed because it is very small, so if the same dataset is requested in the future the urls will be provided immediately and the on-demand conversion is instantiated again. This is a classical tradeoff of speed vs space. Practically we expect it to work well since people tend to work on a particular dataset with focus on its analysis during a confined period of time.

IDX Data Access

IDX data is accessed via a server installed in association with an ESGF data node. The server must have access to the cache of converted climate data. We have successfully installed the IDX server as an Apache module at LLNL. It can be accessed from http://pcmdi11.llnl.gov:8080. The direct access to this link will provide a list of datasets available and each link in the page performs the associated metadata conversion and returns the relative ViSUS configuration file visus.config. There is also a link to monitor the cache size for this server.

Usage

The on-demand conversion operates as a service with a RESTful (REpresentational State Transfer) interface in conjunction with the existing ESGF search system and the ViSUS server Apache module. The interface consists of two phases: creation and conversion.

Creation

The first time a client requests that a given dataset be delivered as an IDX volume, the empty IDX volumes are initialized from the original NetCDF or CDAT XML file. This conversion is called using the syntax: `http://converter.service.name:port/create?dataset=<xml_or_nc_path>&destination=<path_to_idx_cache>&server=http://idx.server.name:port/mod_visus&username=<user>&password=<pass>`. This operation creates empty IDX volumes corresponding to each domain contained in the original dataset (e.g. lon/lat/time and lon/lat/lev/time). The datasets are then registered with the specified ViSUS server and their urls returned in an xml document that can be loaded by client applications. The process is completed in a fraction of second second from a CDAT XML file and only needs to be performed one time ever.

This creation should be called by the ESGF search page. For now, we have a sample page which calls it directly.

Conversion

When the IDX volumes are created, no data is actually converted. This conversion occurs only upon client request for a specific field at a given timestep. The syntax to call the on-demand conversion service is `http://converter.service.name:port/convert?idx=<idx_dataset.idx>&field=<fieldname>&time=<timestep>&box=<region_to_convert>&resolution=<resolution_level>` (box and resolution are optional). This call is performed automatically from the local ViSUS Server. The time to convert a timestep field to IDX depends on the original data size. A domain of 1024x512x32 (float32) can be converted in less than a second, 1024x512 (float32) 2D data took between 0.05 and 0.15 seconds, and 1024x512x26 (float32) 3D data took between 0.8 and 1.1 seconds.

It is worth noting that even though larger domains may require more time to convert, this process is amortized across future requests. Further, subdomain conversion is also supported in the case of extremely large domains, but the practical effectiveness of such partial conversion will depend upon the possibility to access partial data in the original file format.

Data Access

Note that the path to the .xml or .nc file must be accessible by the on-demand conversion service and the cache must be accessible to the IDX server. For the demonstration setup, both the converter service and IDX server are installed on the same machine, as http://pcmdi11.llnl.gov:42299 and http://pcmdi11.llnl.gov:8080 respectively, but only the latter is externally accessible.

Preliminary Demonstration

We have configured the on-demand conversion service and ViSUS server on **http://pcmdi11.llnl.gov:8080/**. The demonstration web page allows to download the configuration files for the datasets currently available on the server in addition to checking the current state of the cache to verify how much data has been converted.

We are attaching to this report a video with a basic demonstration of the usage of the system. The focus is on showing how to activate the initial connection and conversion. Continued use of the system would increase the usage of server and client side of caches and therefore amortize even more the data conversion cost. Our on-demand conversion system facilitates the delivery of specific climate data in seconds that would previously required long transfers. Combined with our user-specified dynamic analysis system and remote streaming, large multi-ensemble data comparisons can be achieved which were previously extremely cumbersome or impossible.

Open problems

Though we have found the majority of datasets to work just fine, there are issues with metadata creation that affect some datasets.

We also plan to improve the metadata conversion to incorporate additional information such as physical bounds so that datasets can be rendered in an expected manner (e.g. a square domain would be stretched along the longitudinal axis). This may become a deeper issue once we explore the role of regridding in data conversion. A related concern is the calculation of data ranges exclusive of the “missing_value”. We need to understand existing practice for dealing with this convention.

Because Apache threads and worker processes do not share memory in the traditional manner, we are currently using the apache *worker MPM* (Multi Processing Module) to host data, limited to a single multithreaded process rather than multiple processes, which could affect the ultimate scalability of the system. However, our concern can be alleviated by utilizing Apache shared memory so that registering new datasets can be communicated to all existing processes.

For efficiency, we utilize lock files to protect against the wasted effort of duplicate simultaneous conversions. Unfortunately, some interplay between the server requests and the threaded system seems to be rendering these less effective and some duplicate conversions seem to be getting performed. Perhaps the conversions are carried out faster than the requests can be made, but this needs to be investigated further.

Beyond the technical aspects of the system, data deployment currently relies on passing around configuration files. This can become overwhelming for the user, therefore developing some means to automatically group and identify data will become imperative. The user can always return to the original search page. Since datasets will only be created one time, successive queries will simply return the existing configuration. Another possibility is to utilize the method incorporated into the existing UV-CDAT integration that retrieves a list of available datasets directly from the server.

We still have to address the issues related to cache management and its policies.

Final Demonstrations

Once things are more stable, we would like to test this system using data that is relevant to current and future research, and demonstrate the ability to blend remote datasets from different locations. In addition, we would like to demonstrate the system’s ability to convert extremely large domain data which we believe in some cases might enable sharing some datasets to a broad community for the first time. We will work in collaboration the Dean Williams and the ESGF team to determine which demonstrations will be most relevant for the goals of the project.