EUROPEAN COMMISSION
Community research

Climate Information Portal

CLIPC DELIVERABLE (D -N°: D5.4)

Tape Archive Interface

Dynamic tape archive extraction and post-processing system

File name: {CLIPC_Deliverable5_4_final.doc }

Author(s): *Mark Elkington(UKMO), Hamish Struthers(LIU/NSC), Emma Hibling (UKMO), Ag Stephens (STFC), Prashanth Dwarakanath (LIU/NSC), Victoria Bennett (STFC)*

Reviewer(s): *Sébastien Denvil (IPSL), Victoria Bennett (STFC), Alan Iwi (STFC).*

Release date for review: *16/05/2016*

Final date of issue: *27/05/2016*

Reporting period: *01/12/2013 - 31/05/2016*

Revision Table			
Version	**Date**	**Name**	**Comments**
0.1	16 May 2016	Draft	For internal review
1	27 May 2016	Final	

Abstract

The rapid expansion in size and scope of climate data simulations forecast for the near future means that the previous model of storing all data on online disk in central archives is unlikely to provide an effective long-term data access environment for a future CLIPC system. This report describes two demonstrator projects to investigate the expansion of the Earth System Grid Federation to incorporate data holdings held in tape archives at modeling centres. The Linköping University/National Supercomputer Centre project investigates an approach for direct user query/retrieval of data items held in the Swedish Meteorological and Hydrological Institute's MARS tape archive. The UK Met Office project investigates an alternative approach to allow an ESGF node and a modeling centre to make strategic decisions on which data will be held in the ESGF node and in linked tape archives and adjust this split based on user demand for specific data sets.

Project co-funded by the European Commission's Seventh Framework Programme (FP7: 2007-2013) under the grant agreement n°607418

Table of Contents

Abbreviations:

CDO	Climate Data Operators
CoG	The ESGF user interface system
DRS	ESGF Data Referencing System
ESGF	Earth System Grid Federation
GRIB	World Meteorological Organisation Gridded Binary file format.
NCO	netCDF operators
netCDF	Network Common Data Form
SAX	Simple API for XML
SODA	System of Online Data Access
Solr	Searching On Lucene w/Replication (HTTP based search application)
WPS	OGC (Open Geospatial Consortium) Web Processing Service

Executive Summary

The climate modelling community is preparing for the next major model inter-comparison programme – CMIP6. This activity and associated modelling projects are expected to generate an archive in the Earth System Grid Federation (ESGF) over the next 5 years of at least 30-50 petabytes. Making this available online to users is costly and trade-off decisions have to be made.

Experience gained with the CMIP5 archive would suggest that centralised online model may need to be adapted to make better use of the climate data produced and held by the modelling centres; specifically:

- A significant proportion of the data held in the archive is rarely or never accessed; though the cost of submission, archiving and management of the data in ESGF is high and the modelling groups responsible for generating these data are also maintaining their own tape/disk archives.

- Users outside of the groups involved in model inter-comparison often require data that has been produced as part of the MIP climate simulations but has not been requested to be archived by the CMIP5 project (especially high volume sub-daily data for use in regional simulations).

It is clear that the ESGF will need to adopt a different long term solution for providing data access, with heavily used core datasets held in the ESGF online disk archives, and the less popular datasets held in existing modelling centre disk/tape archives and only moved to ESGF when requested by users or a group within the climate research or climate impacts community. Given the wide scope of the datasets that may be required to support climate impact projects it is important that this approach forms part of a future CLIPC environment.

In task 5.4 of the CLIPC project, work was undertaken to develop two interfaces between ESGF and remote climate archives held on robotic tape archive systems. These two activities looked at different aspects of a future distributed archive extension to ESGF:

> **Linköping University/National Supercomputer Centre** implemented a demonstration system that allows a user to request EURO4M data produced by SMHI and held in the MARS archive in GRIB format. Their SODA (System of Online Data Access) system enables a user to access EURO4M data in the MARS archive through the standard ESGF CoG[1] interface. To achieve this, metadata from the MARS archive is transferred to the ESGF Solr database to be used in the data search systems. When a user requests download of data from the EURO4M dataset, it uses the standard wget mechanism of ESGF, but is routed through the archive-specific plugin to the SODA scheduler and download management services.

[1] CoG is the current query/retrieval interface to the ESGF network - https://www.earthsystemcog.org/projects/cog/

Preliminary testing of this demonstrator has been completed, and it is the intention that the SODA service will ultimately become part of the ESGF software package.

The **UK Met Office** demonstration system looked at the issue of managing the distribution of climate datasets between the ESGF archives and the local modelling centre archives. This system has been deployed between the Met Office and the CEDA ESGF node, with the Met Office MASS archive system used as the demonstrator tape archive. The system allows the Met Office and CEDA to agree which datasets are routinely ingested into the ESGF archive and which data is held in the MASS archive. If the Met Office or CEDA receive requests for data held in the MASS archive, it can be made available and uploaded immediately. Care has been taken to consider the typical lifecycle of climate data, in order to deal with problems that are identified with data after it has been published. Functional and performance testing during 2016 has been successful, and it is the intention to use this system to support the management of all Met Office datasets for CMIP6.

Although the two demonstrators take different approaches to the problem of distributed access to climate data held in tape archives, there are common features that could be used to support a unified approach. For example, the interfaces developed for the Met Office demonstrator implement the services that are required for the SODA plugins in the LIU/NSC demonstrator. It would also be possible to configure SODA to respond to data upload requests from the modelling centre as implemented in the Met Office solution.

While the results of this work will be finding immediate application for the CMIP6 project, any future evolution of the CLIPC portal should consider the option of introducing features of both developments, in order to open up access to the wider range of data held in modelling centre tape archives.

1. Introduction

The Earth System Grid Federation [ESGF] has evolved over fifteen years of development to serve many data projects in the climate science domain (Chunpir et al., 2015[2]). The current peer-to-peer ESGF implementation started development in 2011, driven largely by the data archiving and access demands of the CMIP5 programme. For CMIP5 the ESGF supported a production model where modelling groups produced large volumes (~1.8 Pbyte) of climate data, which were then made available to end-users via download from online disk at the various ESGF nodes. Since these data sets would be prohibitively expensive to recreate, the large ESGF nodes replicated most of the data between nodes or into tape storage systems to support data recovery.

The CMIP5 data production model was a practical solution for the data volumes involved however experience of the use of CMIP5 data by end-users over the past 4-5 years has shown that a significant proportion of the data has not been downloaded[3]. Moreover, for many users of the data, immediate online access is not practically useful when terabytes of data are typically required to support analysis activities across multiple models/simulations. In addition, users outside of the science community focussed on model inter-comparison have requested additional variables that have been produced by many modelling groups for the CMIP5 simulations but were not requested for the CMIP5 archive. Some mechanism needs to be in place to allow other user groups to request existing data that is not currently in the CMIP archives, but does exist in the archives of the major modelling centres.

As we move into the era of >10^2 Petabyte climate data archives for CMIP6, the practicality and need to provide online access to the entire archive will come under scrutiny. A technological approach would suggest introducing Hierarchical Storage Management (HSM) solutions at each ESGF node, but this would in some ways replicate the mass storage technologies already in place at most climate modelling centres.

[2] Chunpir, H.I., T. Ludwig, D.N. Williams 2015: *Evolution of e-Research: From Infrastructure Development to Service Orientation*. in "Design, User Experience and Usability: Interactive Experience Design", Aaron Marcus (ed), 4th International Conference, DUXU 2015, Proceedings, Part III, DOI 10.1007/978-3-319-20889-3

[3] A full usage analysis of the CMIP5 archive is not available, but *ad-hoc* studies on subsets of the data have revealed significant variation in the number of downloads across the variables in a model simulation. For example, a recent analysis of usage of data provided by one of the contributing institutes revealed that some variables have been downloaded thousands of times, but typically 15-30% of the variables are never downloaded and another 20-30% are downloaded less than 10 times (though these may have been downloads to "community repositories" which are then used by multiple users). Obviously these statistics will vary across models, experiments and simulations, but it is clear that a significant proportion of the online disk archive is being used to store data requested by the scientific community in the project specification, but have never been downloaded. With the more than trebling of requested variables from the CMIP6 experiments it is likely that this issue will become more important and a more flexible archiving arrangement may be beneficial.

With this context, the CLIP-C project undertook to develop demonstrator systems for the dynamic extraction, file reformatting (including metadata) and serving of climate model output currently held in tape archives. Two important tape data archives were targeted by these demonstrator systems - the EURO4M regional reanalysis data held on the MARS tape archive system at SMHI and the Met Office MASS storage system - and their integration with the ESGF infrastructure run by the STFC Centre for Environmental Data Analysis (CEDA). In both cases, emphasis was placed on ensuring that software components and technologies were compatible or could be integrated with the current ESGF software and that the overall system designs were, where possible, flexible/adaptable for reuse on other tape archive systems. Thus we do not expect this approach to completely replace the current ESGF model, but could see it offering more options in the medium term, with online access provided to the most popular data sets and the technologies described here to allow other data to be:

- produced by the modelling centres on user demand and passed to the user via archive synchronisation with an ESGF node (‘Data Push Model’)
- held in the modelling centre’s mass storage facility and retrieved via an interoperable request between the ESGF node and the modelling centre (‘Data Pull Model’).

Because of the different access patterns, technologies and user requirements for the “PUSH” and “PULL”, in the short and medium term it was considered prudent not to try to develop a generic solution that covers all use cases, but instead two solutions were worked on simultaneously with some interaction/communication and knowledge sharing between groups. This was considered to be a fast and cost-effective approach to the demonstrator development.

The identification of key climate data sets is a key part of the Copernicus programme with the aim of providing easy-to-access archives of these data sets. However, many data analyses will lead to requests for additional variables, and/or more extensive temporal/spatial coverage. Extending the Copernicus concept to include support for accessing datasets held in linked archives or produced on user demand at specialist modelling centres will be important in order to provide for a long-term, flexible and affordable system. Two clear examples of future model data archives that may need to be accessed in this way are the Met Office CMIP6 archive and the UERRA European Regional Reanalysis data services described here.

2. ‘Data Pull model’ (SMHI/NSC)

Objectives/Background

The EURO4M modelling activity at SMHI has produced a large amount of data, stored in the GRIB format and using the MARS library system to manage and serve data. Since the task T5.4 called for a tape extraction mechanism, it was decided that the data on the MARS system was an excellent candidate to function as a demonstrator. The need was therefore for a system that achieved the following:

- Publish data from the MARS system onto ESGF, without actually having to bring it all to disk first. The Solr database is populated with information from the MARS EURO4M metadata/DRS[4] information without the need to transfer the data files to the ESGF data node.
- Provide a means for users to discover these offline datasets using the ESGF search interface, and place a request for the data.
- Use CoG to perform an on-demand retrieval of datasets from MARS, a format conversion to netCDF including all the necessary metadata as defined in the DRS, and finally serving the files through a secured fileserver.
- It must provide request scheduling and online cache management features, thereby allowing sites to use it in a scenario where it is not feasible to make all of the data available online, at the same time.

The system designed to meet these objectives is called SODA – System of Online Data Access. A ‘plugin’ component to integrate SODA with the backend storage system is used. This ensures that SODA can be easily deployed with tape storage systems other than the SMHI MARS system.

Data Pull - Actions

Four distinct actions that the SODA system must perform, either internally or interacting with the user or the ESGF node, have been identified. These are summarised below.

Action 1: Data publication

The administrator decides what MARS data needs to be published, and creates constraints which ensure that only metadata pertaining to the selected datasets is queried from MARS.

[4] DRS – Data Reference System, the means by which the ESGF uses metadata ‘facets’ (e.g. experiment, model, institute etc.) to organise climate data products in its archive and by which users can search those holdings.

This metadata is used to create a data questionnaire, which is in turn used to ‘publish’ data onto ESGF using a custom version of the ESGF publisher. For SODA publications, only the metadata is pushed to the Solr database, no data files are physically transferred to the ESGF node. Populating the Solr database in this way enables discovery and ordering via the CoG.

Action 2: Data discovery and ordering

For the user, the data discovery process is straight-forward: they discover the data exactly as they would have for a regular ESGF online data set.

The available options for an offline data set would be different: instead of a regular wget script download, users will be presented with an ’Order Data’ option, which passes on the list of requested datasets to the SODA system. SODA returns the user a status URL which provides the status of their request. When the data has been staged by SODA, the status URL would be updated with a ’Download wget script’ link, after which the workflow is identical to the regular ESGF data download workflow.

Action 3: Data retrieval (Data Pull)

When SODA receives a request from ESGF, it reformats the request using the backend plugin (in this case the MARS plugin) and checks the cache to establish if the requested data is already available.

If the requested data is not available in the online cache, SODA schedules a data retrieval request from MARS.

The retrieved GRIB data is converted to netCDF and staged onto the SODA file server which uses ESGF authentication protocols.

Action 4: Data purge (cache cleanup)

Since the amount of disk cache on the SODA file server is limited, when there is no more space to stage requested files, files are selected for removal, prioritising removal of files that have already been fetched, or not requested recently. These actions are managed by the SODA file server.

Since the whole process of file retrieval and conversion is time consuming, purges will only be done when necessary.

Technical Solution

The key technical goals for the SODA demonstrator were discussed at the CLIPC WP5 'Workshop on tape archive integration with ESGF' held at ECMWF, 22-23 September 2014. Based on these discussions and subsequent development work, two main SODA workflows and the corresponding SODA components have been identified, along with how these SODA components should interact internally and with the ESGF node.

Workflow 1: Data Publication

SODA manages the publication of MARS data using the MARS plugin (which is aware of the project DRS) to generate a metadata retrieval (labelled a 'data questionnaire'). This metadata retrieval contains all of the necessary information to complete a SODA ESGF publication, whereby the Solr database is populated with project metadata for the offline data, but no actual files are transferred to the data node. This action is completed using the ‘push’ mode of the ESGF publisher. The Data publication workflow is depicted in Figure 1.

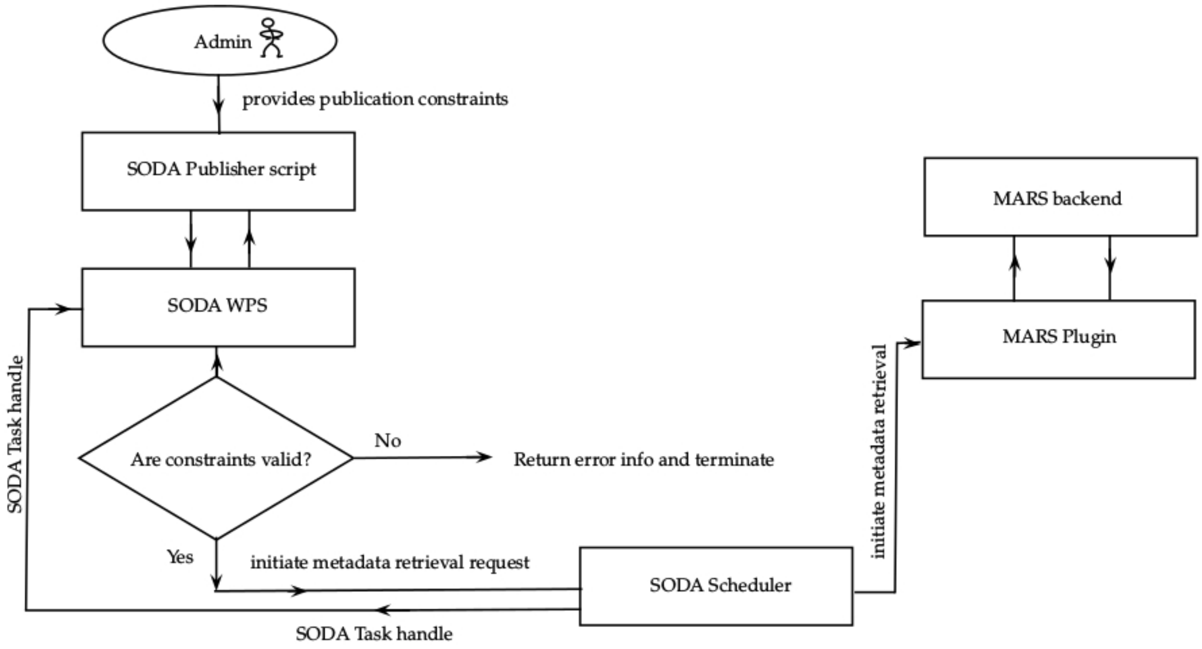

Figure 1: SODA publication workflow

SODA components and steps involved in a SODA publication of offline (EURO4M MARS) data

Workflow 2: User Download

Discovery of offline data through the ESGF-CoG is designed to closely follow the typical discovery for online data, the only difference being that the offline data is marked as 'offline' on CoG for users. Once a request for offline data has been initiated by a user, a service URL is created which shows the current status of the download request. The SODA scheduler initiates a retrieval request to the MARS plugin, which in turn initiates a MARS query to fetch the requested data. Once the data has been retrieved and reformatted to netCDF (by the MARS plugin), the file(s) are placed on the SODA file server, the status URL is updated, and the user is sent a notification including the appropriate wget script for data download. This download workflow is schematically shown in Figure 2.

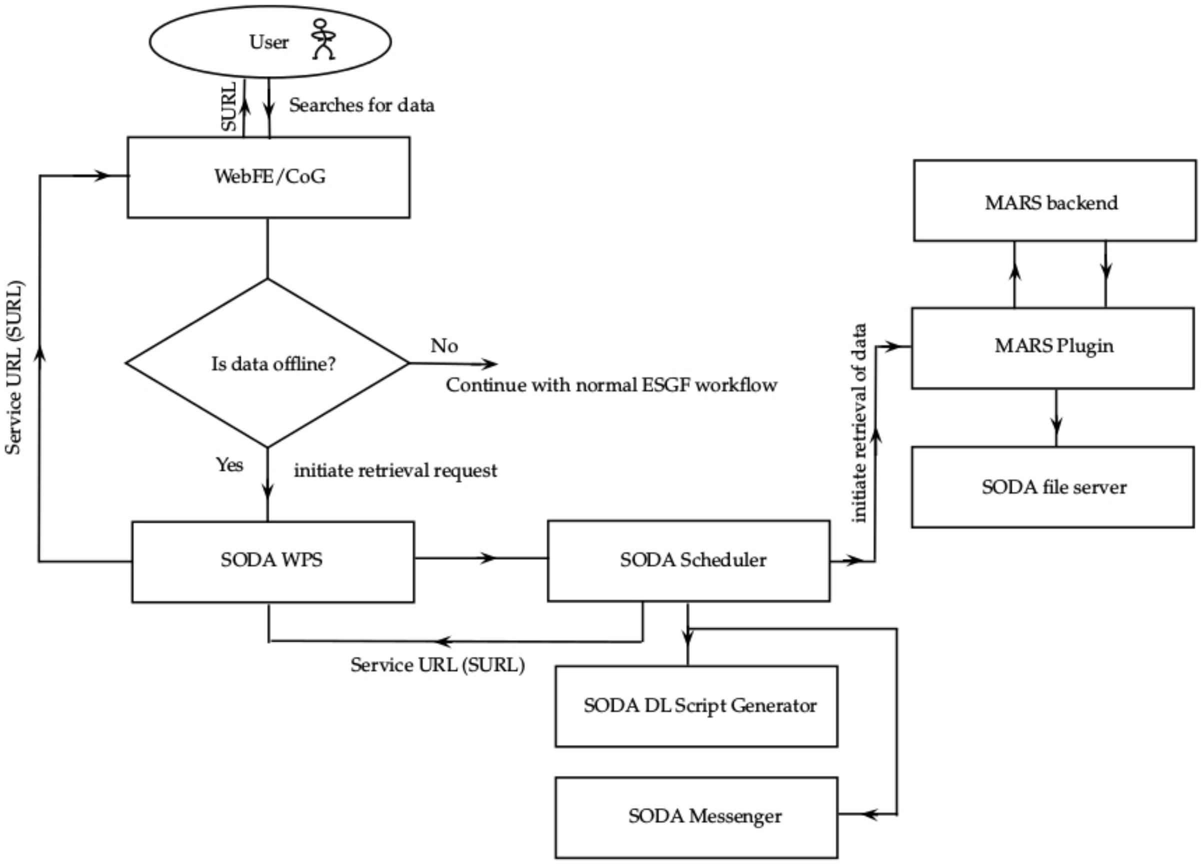

Figure 2: SODA download workflow

SODA components and steps involved in a user initiated SODA download request through the ESGF CoG

SODA Components:

SODA has been implemented with a combination of components developed by LIU/NSC for this project and a number of open-source components. SODA code and developer documentation is available via github (https://github.com/snic-nsc/clipc-dev).

Internal

- SODA WPS: interacts with ESGF Web front end/CoG and SODA-generated wget script.
- MARS plugin
- SODA Publisher: script to obtain data questionnaire from SODA WPS and perform insertion into ESGF database and other tasks to complete 'ESGF publication'.
- SODA Scheduler: to schedule publication requests and downloads requests.
- SODA Messenger: to email notifications
- SODA file server: for online hosting of requested files.

External dependencies

- WPS (Birdhouse) manages SODA internal actions.
- The SODA scheduler is based on Celery.
- RabbitMQ is used for managing message passing (SODA messenger).
- SQLite manages the SODA download database.
- The SODA fileserver is built using the Apache file server.
- NCO for metadata editing as part of the GRIB to netCDF conversion
- CDO for reformatting and metadata editing (GRIB to netCDF).

Technical Implementation Features

Backend Plugin

The backend plugin is designed as a callable external component with predefined function prototypes. For the demonstrators system discussed here the backend plugin refers to the MARS plugin, which is responsible for:

- the actual interaction with backend tape system,
- performing ESGF search facet to MARS query mapping.
- executing MARS retrieval request and GRIB to NETCDF conversion, including metadata editing.

Based on these responsibilities, the plugin must be aware of:

- project-specific DRS,
- project-specific custom key/value pairs, for instance attributes which are common to all datasets belonging to a project but are not available from the DRS.

wget Script and Download Tracking

The SODA file server includes a system for logging user downloads. This is to ensure that files which have been fetched completely by the user can be purged from the cache, in the event of high load on the cache.

The wget script should use checksums provided by the server so that in the event of a subsequent execution, only files that have not been downloaded will be fetched. No attempt to fetch files already successfully downloaded will be executed. The wget script authentication uses the same OpenID system as normal ESGF wget authentication.

Testing / Deployment

To date, testing has been completed on a number of components of the SODA system (SODA scheduler, SODA messenger, SODA file server and MARS plugin).

A small set of EURO4M (MESAN) data has been published using the normal ESGF procedure whereby the datasets are stored on disk on the ESGF data node in netCDF format. This test publication has been used to confirm the EURO4M metadata, file formatting and DRS representation are correct. Data can be accessed by searching for the ‘clipc’ project on the ESGF. This exercise also confirms the scripts and metadata mapping for GRIB to netCDF EURO4M file conversion.

A skeleton SODA system for the download workflow (see Figure 2) has been built and preliminary testing completed. The test system executes all the steps from an ESGF query to the fetching of the corresponding GRIB files from MARS and creating the service URL for the user.

The SODA system is being designed and developed to be fully integrated into ESGF and comply with the necessary security constraints and configuration.

3. 'Data Push model' (Met Office/STFC)

Objectives/Background

The climate modelling activity at the Met Office generates many more simulations, and climate variable datasets from those simulations, than can be effectively supported at the ESGF archives. Moreover, it is also clear that some of the Met Office data held in relatively expensive online archives at the ESGF nodes have rarely been accessed.

The Met Office is investing heavily in rationalising all aspects of its climate data production chain to support "climate service" operations with the aim of being able to generate requested data sets on-demand and thus provide for more flexibility in decisions about what data will be stored online. This collection of services is known as the Climate Data Dissemination System (CDDS).

Given this background the Met Office is keen to investigate techniques that would allow data producers and data access centres to make more efficient use of archiving facilities at both organisations.

The general concept is illustrated in Figure 3.

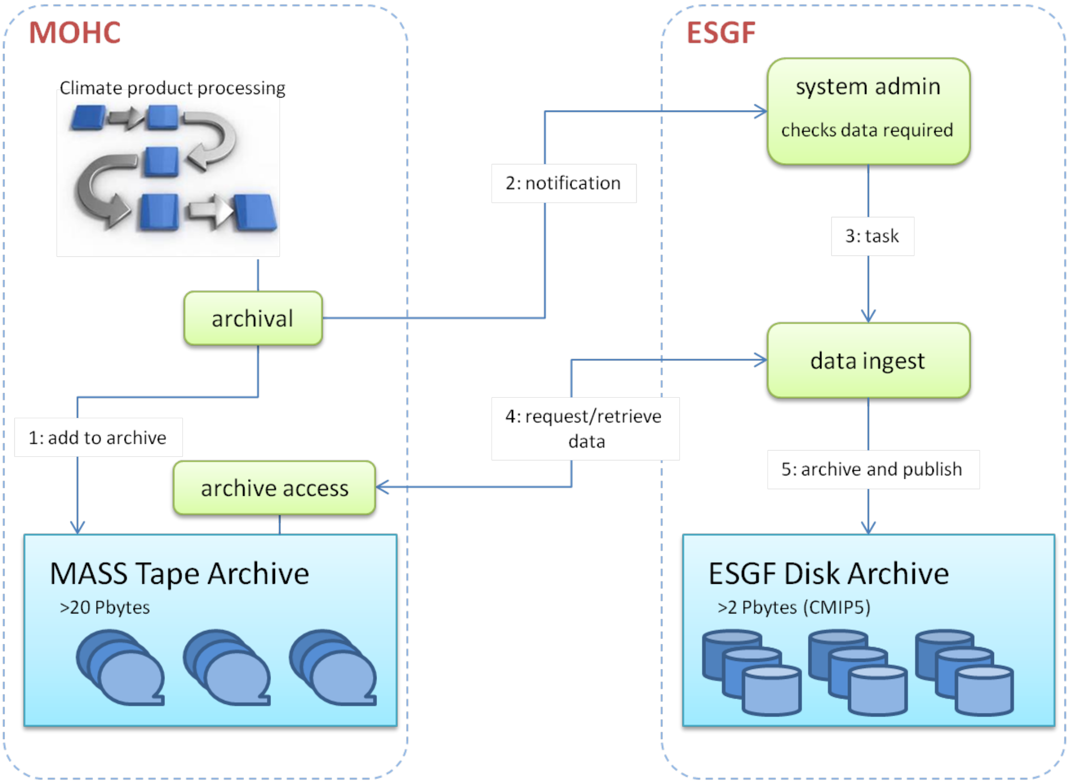

Figure 3: Data Push Demonstrator Concept

Once deployed, it would be possible to limit the data sets held in the ESGF nodes to the variables (atomic datasets) that are most in demand. Users wishing to access less commonly used variables could place a request with the access centre to see if these data have been archived by the modelling centre for the simulations they are interested in. If they do exist in the data producer archives they could be transferred to the data access centre, allowing access by the user community through the standard ESGF access methods. If they have not yet been produced, but can be produced from the existing raw simulation data, then they could be produced and made available to end users in the same way as the standardised variables (e.g. complying with the versioning, metadata annotation and quality control requirements used for the core variables initially ingested into ESGF.

In addition to providing flexibility as to where archived products will reside, one serendipitous objective was to provide functionality that would allow the data access centre and data producers each to check their data holdings across the system. Practical experience of handling millions of data files in long term archives have demonstrated that it is important to be able to do periodic audit checks to establish that all expected files are present at the ESGF node and at the correct version. The implemented demonstrator allows the ESGF node manager to compare the status and version of each variable held in the ESGF archive with the corresponding variable held in the remote tape archive.

Data Push - Use Cases

A number of key use cases were developed to help define the required functionality for the Data Push model. These are summarised below.

Use Case 1: **Make producer archived products available to ESGF node**

In this use case, individual atomic datasets[5] held within the Met Office MASS archive system are marked as available for access by the ESGF node. A notification of the change in status is passed to the ESGF via a messaging server at the node that is responsible for supporting the data from this modelling centre.

The ESGF node has a routine process that checks the messaging queue and takes action based on the message content. If a dataset required by the ESGF is made available in the Met Office archive (i.e. it matches a requested variable for a project that the ESGF node is supporting), a request to download the data from the MASS archive using its standard client interface is submitted, and the specified dataset(s) are retrieved. The datasets are then ingested and published in the ESGF node in the normal way.

Pre-existing State:

[5] An atomic dataset is defined as a set of one or more files for a single climate variable covering a complete climate simulation.

- Data set is produced by Met Office climate product system.
- Data set is subject to project specific quality assurance procedures.

Steps:

1. Dataset is moved to the MASS archive system and located using a local, project specific data reference system (DRS).
2. Data set is marked as "available" by dataset administrator.
3. Notification message is sent to linked ESGF node - message provides information on dataset access status and location (using the DRS). Message is stored in queue at ESGF node.
4. An automated scheduled process at ESGF checks message queue at frequent intervals and triggers process to retrieve data from Met Office archive using MASS client.
5. Data is retrieved, quality checked and published in the ESGF archives, and becomes available to end-users.

Outcome:

Data has been moved from local Met Office MASS archive to ESGF node. ESGF node retains autonomy on which data it wants (or can afford) to make available through its own archive, and which data will reside in the MASS archive until there is user demand for those datasets.

Use Case 2: Verify consistency of data holdings at local producer archive against ESGF archive

Experience with long-term management of online climate data sets in ESGF involving $>10^6$ files has shown that it is essential to be able to verify the status of the ESGF archive. In this use case the ESGF accesses the MASS system in order to obtain a status listing covering each dataset associated with a specified subset of the ESGF archive (e.g. project, experiment). The listing will include the identification of each atomic dataset and its current status (available, embargoed, or withdrawn) and version (date).

Pre-existing State:

- Datasets have been transferred from the modelling centre to an ESGF node.

Steps:

1. The ESGF node establishes a routine process to produce status listings of the modelling centre's data holdings at an appropriate granularity (typically for an entire project or individual experiments).
2. The status listing(s) will be routinely compared with the ESGF holdings of the equivalent atomic data sets and a "difference" list generated. The reported differences will include version, status, file number, and optionally checksum digest information.
3. According to predefined rules established at the ESGF node, the difference list items will result in either:
    - Automated transfer of datasets from the modelling centre or withdrawal of datasets at the ESGF node.
    - Manual intervention by ESGF data managers to understand and resolve a reported difference.
    - No action.

Outcome:

The modelling centre and ESGF are able to perform regular, automated checks on the consistency of their data holdings. The modelling centre archive is able to provide a backup source for the data held in the ESGF node.

Use Case 3: Change status of data held in ESGF archive

There are several reasons why the status of data accessible from the ESGF archive may change. Typically, end-user analysis of the released data may lead to the discovery of errors in the dataset that were not detected during the automated quality control processes. Alternatively, the modelling centre responsible for producing the dataset may discover an error in the simulation configuration after making the data available to the ESGF node. In these cases the affected dataset(s) need to be either withdrawn from the archive, or replaced with an updated version.

Pre-Existing State:

- Data sets have been made available and transferred to ESGF node.
- A problem has been discovered with a number of atomic data sets and has been reported to the data producer.

Steps:

1. Data set is marked as “withdrawn” in the local archive (e.g. MASS) by the dataset administrator at the modelling centre
2. A notification message of status change is sent to the linked ESGF node - the message provides information on dataset access status and location (using the DRS). The message is stored in a queue at the ESGF node.
3. An automated scheduled process at ESGF checks the message queue at frequent intervals and identifies that some datasets in the ESGF archive have had a status change (in this case ‘available’ in the ESGF archive and ‘withdrawn’ in the MASS archive). This is reported to the ESGF data managers.
4. The ESGF staff use existing administration tools to: a) disable user access to the affected data sets, and b) notify existing users of these data sets that they have been withdrawn.

Outcome:

Change of status of the affected data sets have been synchronised across the archives.

Notes:

Experience with previous large climate experiments has shown that although the withdrawal of a dataset is important to ensure that the impact of data errors on the science is minimised, the occurrence of these errors is relatively rare (largely due to routine QA procedures performed by the modelling centre and ESGF centre). For this reason the withdrawal action is managed through manual intervention. The ESGF centre may choose to physically delete any withdrawn datasets or just unpublish them. If the datasets are regenerated by the modelling centre a new version will be archived in the local archive and made available to ESGF. The ESGF message check will identify that a new version of the data is available and transfer it in the normal way.

Use Case 4: Retrieve products on demand

For some datasets the ESGF node could decide to routinely download only the most commonly used datasets from a climate simulation. For example, the most popular datasets (e.g. monthly/daily data) would be retrieved routinely as the modelling group makes the atomic datasets available (as described in Use Case 1). The high volume sub-daily atomic datasets would be only retrieved following user request. It is not the intention that this "on-demand" request would make requested data available immediately; rather it would collect the data and make it available within a few days which we believe is appropriate for most climate research purposes.

Pre-Existing State:

- Data sets have been created and stored in the Met Office archive and marked as available.
- The ESGF transfers a constrained list of data sets into its own archive.
- The ESGF node configures the status check on the local archive to be performed on a regular (daily) basis.

Steps:

1. End-user requests information on availability of a set of variables that do not appear in the EGSF archive search.
2. ESGF administrators perform search for requested variables in ESGF and remote producer archive(s) using the latest status check report.
3. If the variables are available in the Met Office archive, ESGF extends it's ingest filters to accept those variables during the next routine pass through the message queue.
4. An automated scheduled process at ESGF checks message queue at frequent intervals (as described in Use Case 1) and triggers process to retrieve the variables from Met Office archive using MASS client.
5. The variables are retrieved, quality checked and published in the ESGF archives, and become available to end-users.
6. User requesting the data is notified that data is available.

Notes:

If the ESGF chose to only store a small fraction of the available data in the ESGF node, it would be necessary to automate Steps 2 and 3 - introducing a new user service; "search modelling centre data holdings".

Technical Solution

The key technical goals for the Met Office tape archive access demonstrator were agreed between the Met Office and STFC-CEDA (who operate an ESGF node) as:

- autonomous operations for producer(Met Office) and data centre (STFC-CEDA),
- persistent, recoverable messaging,

- support for project specific data organisation,
- support for product lifecycle states typical in climate modelling projects and product versioning,
- support for long-term archive verification (i.e. data in accessible archive is what the producer expects to be there).

The main features of the CDDS implementation of the tape archive integration are shown in Figure 4.

Met Office operations:

As climate data sets are produced by the Met Office the Climate Research Experiment Management System (CREM) is used to ingest them into the MASS tape archive. It does this using the CDDS Archiver capability developed for this project. The CDDS archiver uses the shared configuration file to understand where to put the datasets in the MASS archive and organises the data set files into locations that are tagged with a version date and an appropriate status – e.g. "*available*" if they are to be moved to the ESGF.

Once the data is ingested the Archiver sends a message to the event queues at ESGF and the message includes project specific configuration information relevant to the DRS used to enable the ESGF to correctly locate the new data sets.

CEDA Operations:

Scheduled checks on the event queue by the CDDS Archiver installation at CEDA will identify data sets made available in the Met Office MASS archive. It will check them against the status of data already held in the ESGF archive and if they are not present, or are not at the latest version will then use the integrated MASS interface (Moose API) to retrieve the relevant datasets from MASS.

To support the routine verification of the archive status, CEDA can use the Status Check functionality available from the CDDS Archiver to produce archive inventory listings for selected subsets of the MASS archive (e.g. a single experiment or an entire MIP, etc). These listings can then be used to verify the status and versioning of the Met Office climate data in the CEDA ESGF node.

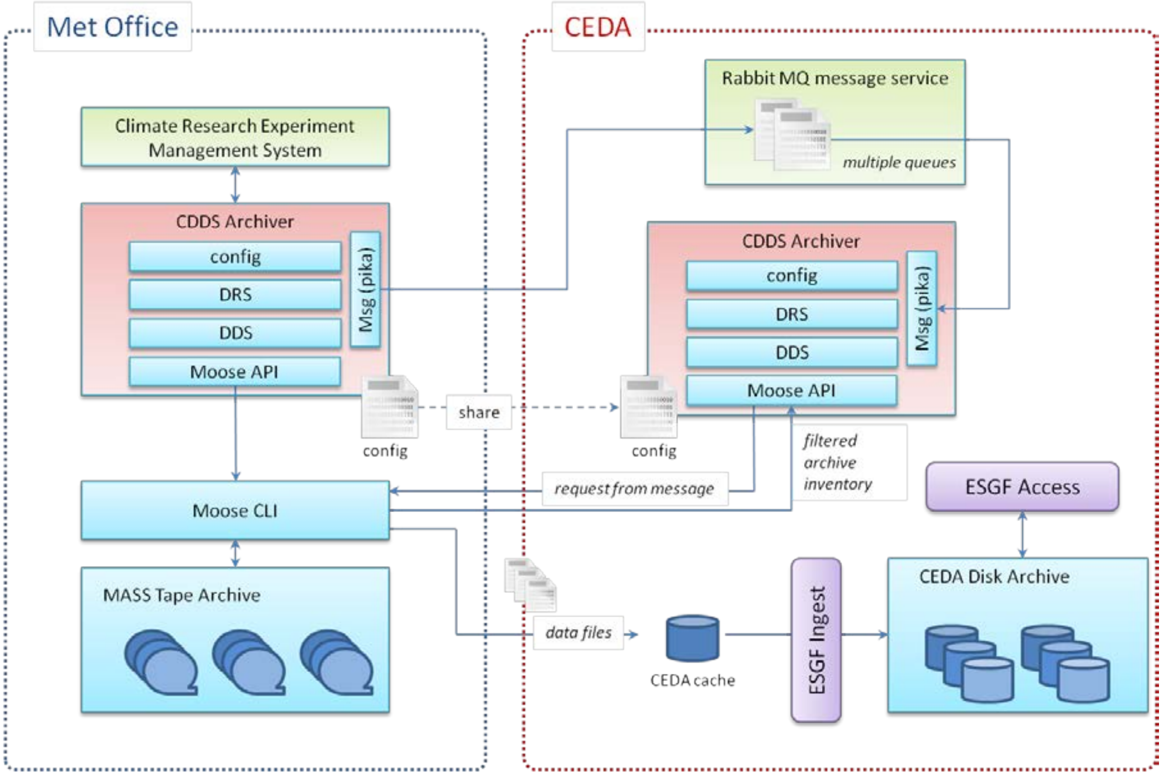

Figure 4: Data Push Technical Implementation Context

CDDS Archiver deployment to synchronise the Met Office and CEDA data archives

The deployed system complies with the well-established security configuration at the Met Office and CEDA. Specific ports are configured for Met Office interaction with the Rabbit MQ server at CEDA. Similarly access to the Met Office is restricted to specific accounts and specific servers at CEDA. The Rabbit MQ server is configured to support disaster recovery.

Technical Implementation Features

The interface consists of a small set of python modules and a supporting set of configuration files in INI file format. A RabbitMQ message queue server is run by CEDA. The python pika library is used to interface with RabbitMQ.

Shared data referencing solution

Design: A simple class is used to represent files and file sets using the project-specific data reference syntax to define valid facets and their values. The class provides services that return the paths to file sets on local disk or in the MOHC tape archive.

The data reference syntax for a particular project is specified via a plain-text config file in INI file format. The INI file is used to define:

- The facets (elements) used to build a file name,

- The facets used to build a path on local disk to the directory containing the file set,
- The facets used to build a path in the tape archive to the directory containing the file set,
- The dataset id for the file set.

Some projects may define facets that are derived from other facets. For example, GeoMIP data reference syntax includes “frequency” and “realm” facets that must be derived by parsing the matching MIP table. The data reference syntax class allows custom facet handlers to be supplied to derive new facets if necessary.

Justification: Data reference syntaxes for different projects vary, and they can often go through multiple versions before they stabilise. We therefore felt that a template-driven method of defining a project’s data reference syntax would be sufficiently flexible while still enabling configuration management[6]. The same underlying data reference syntax library can be shared by MOHC and CEDA. The only local customisation required is a small modification to the configuration file to define local directory paths.

Implementation: Project-specific data reference syntaxes are defined using an INI configuration file. For example, a CORDEX file might contain a line like the one below to define the facets used to build a file name:

```
name = variable_domain_drivingModel_experiment_ensemble_rcmModelName_version_frequency_[date]
```

Product versioning and state management

Design: The CMIP6 programme is planning to change the versioning scheme to version control the file-set level (i.e. all files for one variable in one climate simulation). We defined four different possible states for file sets in the MOHC tape archive, summarised below:

[6] In other parts of the CLIPC project, work is ongoing to define and harmonise the DRS for different climate datasets.

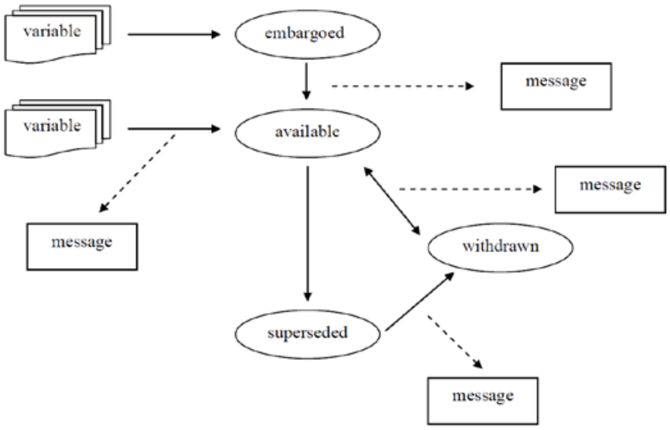

Figure 5: Product State and Versioning

CDDS Archiver supports typical data product lifecycle

Valid state transitions are represented by solid arrows. Messages are sent when a variable moves to “*available*” or to “*withdrawn*”. States and versions are indicated in the MASS archive using paths.

Justification: For MOHC, the MASS archive is intended to be the canonical source of data to be sent. We therefore required a design that supported the normal workflow (gradually accumulating file sets in an “embargoed” state before making them available) and the error-handling workflow (withdrawing file set(s) if problems are found) while leaving the archive in an unambiguous state, and also while allowing CEDA to compare the contents of their archive with MASS.

The path-based design enables file sets to be quickly moved from one state to another as an atomic action, so it is not possible for files to be “left behind” in an old state or version.

Implementation: States are represented by a simple state-machine class which provides services indicating whether a transition is valid, whether a file set can be sent to MASS in a particular state, and whether a message is sent on entering a state.

MASS paths are built using data reference syntax facets, with sub-directories representing the state and version. For example, an embargoed variable might be archived in `/CMIP6/MOHC/UKESM1.0-N96ORCA1/ssp2-45/r1i1p1f1/Amon/gn/embargoed/clt_20160420`. The state is indicated by a sub-directory called “embargoed” below the data reference syntax directories. Files are stored in a sub-directory named “`<drs-variable name>_timestamp`”, where timestamp is the date that the directory was created in the archive. State transitions are done by a simple directory move, which is an atomic action in the tape archive. For example, to make the `clt_20160420` file set available for access by the archive centre, it is moved to a sub-directory of available/clt_20160503. The timestamp is updated to reflect the date of the state change.

Messaging system and message handling

Design: The key design principle behind the messaging system was to ensure that the MOHC archive and the CEDA archive remains consistent in the event of communication errors. Messaging problems should not prevent the movement of data into the MOHC tape archive, and it should be possible to resend messages at a later date. At CEDA, messages should be stored securely such that they can be recovered and processed in the correct order following an outage or another problem.

Messaging is handled using a connection manager class, whose responsibility is to manage a single, open connection to the RabbitMQ server. If a connection cannot be opened, or if an open connection is unexpectedly closed, the manager tries to reconnect several times, before quietly failing safe. Messages are sent or read via the managed connection. Messages that cannot be published are saved to files on local disk so they can be re-sent later on.

At CEDA, RabbitMQ queue and message options are used to ensure that messages and queues are durable to prevent messages being lost.

Justification: RabbitMQ was chosen as a messaging server because it is a widely-used open source messaging system with python APIs.

The connection manager class supports the “do your best and fail-safe” design requirement. It prevents messaging problems from rippling up to stop data transfers to and from the MOHC tape archive. The message store on local disk enables messages to be re-sent after errors. It is also possible to reconstruct messages following major failure (communication problems at the same time as problems with the local message store) by searching the MOHC archive for file sets and rebuilding matching messages.

Configuring RabbitMQ to use durable queues and messages ensures that CEDA will be able to process messages in the correct order following system or RabbitMQ server outages. This ability comes at the cost of performance, since making messages persistent means saving copies of them to disk. However, the design assumes that CEDA will be processing messages intermittently (e.g. checking for new messages once a day) rather than in real time, so messaging performance is not critical. Performance tests have confirmed that messaging overhead is minimal compared to file transfer overhead.

Implementation: A Python class is used to create and maintain a single connection to the RabbitMQ server. The connection is opened when the object is created. The class also provides an interface to perform messaging operations using the connection, trapping and logging errors if they occur. In the event of a message publishing failure, messages are converted to a simple plain-text format and saved to local disk. At the CEDA part of the system, queues are configured to be persistent and durable. Messages are also marked as durable, so the message queues maintain their state even after unexpected outages.

MOHC archive access for public access system

Design: The MOHC tape archive has a command-line interface that is available on MOHC systems and at CEDA. A small python wrapper already existed for the command-line interface which provided some basic services (run command, distinguish between retryable errors and fatal errors). A simple interface was built on top of this to provide general “put”, “get” and “list” services for the data dissemination system.

Justification: Encapsulating the tape archive interface in a separate class is intended to make it easier to provide a further layer of abstraction to support other tape archive interface layers for ESGF nodes that can retrieve from multiple sources.

Implementation: Currently a simple library has been built on top of the low-level MASS archive wrapper. The library provides the calls required by the data transfer system (“get”, “put”, “list”, “make directory”). The interface builds appropriate command-line options and arguments to interact efficiently with the MOHC tape archive.

Archive verification

Design: It is expected that the MOHC tape archive directory containing data to be shared will be a deep structure containing a very large number of files (>10^8). The design therefore needed to support both dynamic listing of the archive (expected to be slow) and an offline mode, where listings could be done in batch, saved and filtered later to do cross-checks of the MOHC and CEDA archive.

A “*filter facets in tape archive*” method was designed that allowed setting fixed data reference syntax facets (that must be matched for all returned tape archive paths) along with optionally including and excluding specified facets. The returned result set can either be immediately processed (for example, by building local paths for each matching facet and checking they exist), or it can be saved to a file for later filtering.

Justification: The MOHC tape archive listing interface limits the options for listing with searches. As a result, the most likely use cases will produce large trees of results which will take a long time (minutes) to produce and will need to be filtered in memory to apply the required constraints. The design supports:

- Parsing the listing output as a stream to avoid needing large temporary memory space

- Filtering as it parses to build up as small a result set as possible
- Storing the result set for later filtering, to avoid having to wait for the initial listing to complete every time

Implementation: The filter method uses the supplied fixed facets to construct the wildcard-based tape archive path. The resulting recursive listing output is parsed using a SAX parser and a result set is constructed by applying any additional fixed facets, before applying the include/exclude constraints if specified. The matching results are returned as an object which can be serialised and saved to a file in JSON format.

Implementation/Testing

The system illustrated in Figure 4 was deployed at CEDA and the UK Met Office in early 2016. Since then detailed system testing and network performance tuning has been completed. A recent end-to-end test of the CDDS demonstrated that this approach to archive management and synchronisation between an ESGF node and the modelling centre's internal tape archive had significant functional and efficiency benefits over the existing technology (used for CMIP5 and more recent model inter-comparison projects).

Following this successful testing it has subsequently been agreed between both organisations that this new approach will be used for the CMIP6 programme in addition to the clipc operations.

4. Future Development

The two technology projects described in this report have already demonstrated the utility of integrating tape archive access into the ESGF to allow wider access to climate data produced by modelling centres.

The NSC/LIU project set out to investigate the possibility of developing an extension to the ESGF software environment which would allow end-users to directly query and retrieve data held in external tape archives. All of the components necessary to implement the download workflow of EURO4M data from the SMHI MARS archive using SODA have been functionally tested. The next stage will be to integrate these components into the ESGF system code, and complete the implementation and deployment for the SODA publication workflow. Once SODA is completed, tested and integrated into the ESGF system, it is envisaged that the SODA code will be distributed with ESGF code in future ESGF releases and SODA will be integrated as an option into the ESGF installer

The Met Office/SFTC project has completed the development of a system to integrate the Met Office’s MASS tape archive system with STFC’s CEDA archive to support dynamic decisions on where individual data sets are held. Testing has shown that this is a significantly more effective solution than the data transfer systems used for CMIP5. As the new software can work independently of the ESGF software it has been straightforward to deploy and test operationally. As a result it has been decided that the new capability, developed under the CLIPC project, will be used to support the management of the CMIP6 datasets between the two organisations. It is currently undergoing operational testing prior to the start of the CMIP6 data deliveries.

By design there are clear differences between the two demonstration projects. The NSC/LIU project looked at the issues of extending the ESGF system to support an end-user querying holdings in a remote tape archive and ultimately retrieving data items of interest. The Met Office/STFC project looked at the issues of distributing climate data sets between an online end-user accessible archive (ESGF) for popular data sets and a remote tape archive (MASS) for more specialised data; and how to efficiently migrate data from MASS to ESGF on demand.

Although different, there are some key common elements in the two projects. For example both projects utilise wrappers on interfaces to the tape archive to query the archives content and retrieve selected data sets. It would be relatively straightforward, for example to provide wrappers for the MASS archive to take advantage of the SODA interface in a future ESGF implementation.

It is the intention that the results of both projects will be used operationally in the next few years. Operational experience will provide valuable information to guide the future development of a more extensive interface to remote modelling centre archives in time to support the expected explosion in data volumes resulting from increasingly higher resolution climate simulations and massive ensemble experiments (1000s of elements). The key features of such a future development should include:

1. Synchronised archive holdings between ESGF node and modelling centre tape archive based on agreed policies.

2. Data set lifecycle support to manage removal from ESGF node if quality issues are identified in the data or the datasets are no longer considered high priority for online access.

3. Direct end-user query, selection and retrieval of data held in remote archives.

4. ESGF archive verification functionality (does the ESGF contain a complete collection of the data sets it should hold with the latest released version for each data set).

5. Published tape archive query and select/retrieve interfaces to support plugin development.

6. Support for project-specific DRS specifications.