LIVING DOCUMENT

ESGF IMPLEMENTATION PLAN

INTEGRATION OF COMPONENTS, HARDWARE, NETWORKS, AND SOFTWARE TOOLS NEEDED FOR LARGE-SCALE HETEROGENEOUS DATA MANAGEMENT

ESGF EXECUTIVE COMMITTEE

DEAN N. WILLIAMS (CHAIR), MICHAEL LAUTENSCHLAGER (CO-CHAIR), LUCA CINQUINI, SEBASTIEN DENVIL, MARTIN JUCKES, ROBERT FERRARO, DANIEL DUFFY, CECILIA DELUCA, V. BALAJI, BEN EVANS, AND CLAIRE TRENHAM

ESGF WORKING TEAM LEADS

SASHA AMES, RACHANA ANANTHAKRISHNAN, KATHARINA BERGER, NICOLAS CARENTON, LUCA CINQUINI, ELI DART, CECELIA DELUCA, CHARLES DOUTRIAUX, DANIEL DUFFY, PRASHANTH DWARAKANATH, SANDRO FIORE, SAM FRIES, STEPHAN KINDERMANN, MATTHEW HARRIS, MARY HESTER, LUKASZ LACINSKI, PHILIP KERSHAW, PAOLA NASSISI, BIBI RAJU, TORSTEN RATHMANN, GEORGE RUMNEY, MARTINA STOCKHAUSE, AND TOBIAS WEIGEL

MAY 4, 2016

THE EARTH SYSTEM GRID FEDERATION REPRESENTS A MULTINATIONAL EFFORT TO SECURELY ACCESS, MONITOR, CATALOG, TRANSPORT, AND DISTRIBUTE PETABYTES OF DATA FOR CLIMATE CHANGE RESEARCH EXPERIMENTS AND OBSERVATIONS.

ABSTRACT

THE IMPLEMENTATION PLAN IS A LIVING DOCUMENT THAT DESCRIBES HOW THE EARTH SYSTEM GRID FEDERATION DATA MANAGEMENT SYSTEM WILL BE DEPLOYED, INSTALLED, AND TRANSITIONED INTO AN OPERATIONAL SYSTEM. IT CONTAINS AN OVERVIEW OF THE SYSTEM, A BRIEF DESCRIPTION OF THE MAJOR TASKS INVOLVED IN THE IMPLEMENTATION, THE OVERALL RESOURCES NEEDED TO SUPPORT THE IMPLEMENTATION EFFORT (SUCH AS HARDWARE, NETWORK, SOFTWARE, FACILITIES, MATERIALS, AND PERSONNEL), AND ANY SITE-SPECIFIC IMPLEMENTATION REQUIREMENTS. THE PLAN IS DEVELOPED DURING THE DESIGN PHASE AND UPDATED DURING THE DEVELOPMENT PHASE; THE FINAL VERSION IS PROVIDED IN THE INTEGRATION AND TEST PHASE AND IS USED FOR GUIDANCE DURING THE IMPLEMENTATION PHASE.

REVISION HISTORY

Date (MM/DD/YYYY)	Identifier (YYYY.revision#)	Author (Name/ID)	Change Description	Document Owner (Name/email)
04/5/2016	2016.04.01	Dean N. Williams	Initial document draft.	Dean N. Williams williams13@llnl.gov
05/04/2016	2016.05.04 Version 1.0	Dean N. Williams	Initial Release	Dean N. Williams williams13@llnl.gov

TABLE OF CONTENTS

Executive Summary

Disseminating the Earth System Grid Federation's (ESGF's) plans, progress, and results internally and externally is the key to sustainability and growth and helps create a culture of transparency, trust, and openness. This document, the ESGF Implementation Plan, presents step-by-step guidance for planning and implementing the improved ESGF data strategies high profile community science projects requirements. The plan is intended to guide and inform the steering committee, senior executive committee, working team leaders, support staff, project representatives, and research/data analysts. It is a living document and will be updated in conjunction with the annual ESGF Face-to-Face (F2F) technical conference.

Mapping ESGF's performance and development advances identifies and eliminates duplicate efforts and inconsistencies in the infrastructure and data workflow processes. This plan helps document how ESGF responds to issues of widely varying scope. ESGF is already required to report advances and quality measures in annual reports—for example, the Annual ESGF Face-to-Face Conference Reports (http://esgf.llnl.gov/reports.html)—and new project requirements are scheduled to start affecting ESGF components and operations by the end of 2016 and beyond (e.g., in time for the arrival of the Coupled Model Intercomparison Project Phase 6 data).

This implementation plan is a companion document to the 2016 5th Annual ESGF Face-to-Face Conference Report (http://esgf.llnl.gov/media/pdf/2015-ESGF_F2FConference_report_web.pdf). As in the report, we recommend here that all data and modeling centers install the latest ESGF software stack, in preparation for pending community projects that will rely on ESGF.

Selecting the right implementation approach and communicating the plan in a coherent guide is paramount for a successful multi-level infrastructure platform such as ESGF. By devoting substantial time up front to carefully crafting an implementation strategy, ESGF saves time by avoiding major mid-stream adjustments. To accomplish this, working teams take the following steps before selecting an approved implementation strategy:

1. Identifying specific problems faced by the working team, and documenting the most likely causes;
2. Reviewing available materials that describe what other subcomponents have done to improve ESGF node implementations;
3. Considering all resource and service implications; and
4. Considering interference with other working teams.

Once the strategy has been selected, the working teams will create a road map for the implementation process. An implementation plan completed by each working team helps:

1. Identify working team goals and strategies;
2. Plan the approach;
3. Estimate the time and expenses associated with implementation; and
4. Identify optimal hardware, network, and software component performance measures.

When completed, the working teams will share their implementation plan with other working teams, which ensures that collaborating teams: (1) are aware of the efforts underway; and (2) understand the timeline, dependencies, budget, and resources needed. This information dissemination step is discussed later in this living document (in **Section 2**). For determining specific resources needed for each task, see **Section 2**.

Overview

The Implementation Plan for the Earth System Grid Federation (ESGF) describes how the federated system is deployed, installed, and operated to support the Sixth Phase of the Coupled Model Intercomparison Project (CMIP6) and other projects, such as Observations for Model Intercomparisons Project (Obs4MIPs), the Coordinated Regional Climate Downscaling Experiment (CORDEX), Accelerated Climate Modeling for Energy (ACME), and the Collaborative Reanalysis Technical Environment Intercomparison Project (CREATE-IP). It contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort (such as hardware, networks, software, facilities, materials, and personnel), and any site-specific implementation requirements. Also, it distinguishes between short-term (2016) and mid-term (2017–2019) implementations. The plan is ongoing and currently in the development phase (as ESGF, in its many forms, has been in operation for almost two decades). The plan also includes an integration and testing phase, which will be used for guidance during the implementation phase.

The plan is organized by implementation task, as shown in **Table 2**. The major ESGF subcomponent tasks described in the subsections are not site-specific. Instead, they are generic or overall project tasks that are required to install and/or connect hardware, network, software, prepare data, and validate the ESGF federated infrastructure as a whole. Implementation approaches are reviewed, and then advantages, disadvantages, risks, issues, estimated time frames, and estimated resource requirements for each of the options are identified and considered for development and execution. The subcomponent considerations for development include:

- Incremental implementation or phased approach;
- Parallel execution;
- One-time conversion for switchover; and
- Any combination of the above.

Included in the subcomponent implementation information is the description of each major task, if appropriate, including:

- What the task will accomplish;
- Resources required to accomplish the task;
- Key personnel responsible for the task; and
- Criteria for successful completion of the task (e.g., “ESGF project and user acceptance”).

The following are examples of major subcomponent tasks:

- Provide overall planning and coordination for the implementation.
- Provide appropriate training for personnel.
- Ensure that all manuals applicable to the implementation effort are available when needed.
- Provide all needed technical assistance.
- Schedule any special computer processing required for the implementation.
- Perform site surveys before implementation.
- Ensure that all prerequisites have been fulfilled before the implementation date.

- Provide personnel for the implementation team.
- Acquire special hardware, software, and/or network connections.
- Perform necessary data conversion before loading data into the ESGF system.
- Review any security policy changes and vulnerabilities within the task.
- Prepare ESGF site facilities for implementation.

Evaluation and execution of necessary changes after subcomponent implementation is an ongoing process. These changes may include, but are not limited to, personnel and technology equipment adjustments and budgetary considerations from designated funding sponsors.

ESGF is committed to using well-established quality standards as a foundation for new subcomponent design and providing the community of users with the most up-to-date documentation and training possible. Such standards are already being employed to ensure that current subcomponent offerings meet best practices in software design, engineering, and implementation. ESGF is systematically and rigorously developing, testing, evaluating, and documenting its subcomponents and associated tasks via teleconferences, face-to-face workshops and conferences, written reports, and journal publications. Other external living plans and policy documents help guide the evolution and ensure the success of ESGF, including:

- ESGF Governance Policy (http://esgf.llnl.gov/governance.html);
- Logo Requirement and Usage Guidelines (http://esgf.llnl.gov/logo_requirements.html);
- ESGF Strategic Roadmap (http://esgf.llnl.gov/media/pdf/2015-ESGF-Strategic-Plan.pdf);
- Software Security Plan (http://esgf.llnl.gov/media/pdf/ESGF-Software-Security-Plan-V1.0.pdf);
- ESGF Federation Policies and Guidelines (under development);
- User Training Plan (under development);
- Root Certificate Authorities Policy (under development);
- Data Storage and Replication Plan (under development); and
- ESGF Readiness Document (under development).

To expedite the execution of key ESGF features in preparation for the CMIP6 archive—the primary driver for most development—we have prioritized the development efforts of each subcomponent. The first CMIP6 data sets are scheduled to arrive at the beginning of 2017, and thus most short-term ESGF subcomponent implementation activities must be completed by December 31, 2016.

1. Introduction

The consortium of institutions behind the Earth System Grid Federation (ESGF) has decades of experience helping the climate community build the right software solution stack for a multitude of large-scale, heterogeneous, data management projects. The ESGF team has helped to determine the architectural requirements for many national and international agencies' software investments and to foster interagency partnership. Today's ESGF information technology (IT) strategy is focused on outcomes, and there is no greater opportunity to improve outcomes than by optimizing the software that manages high-profile climate data, thereby supporting scientific productivity and knowledge discovery. By staying focused and delivering on community requirements, ESGF is able to keep to its mission of providing and supporting secure access, monitoring, cataloging, transport, and distribution of large-scale heterogeneous data for climate change research experiments and observations.

This section provides an overview of the ESGF system and includes any additional information that may be appropriate for the basic understanding of it implementation plan.

1.1 Purpose

The ESGF executive committee oversees performance and resources across a large number of working teams. The ESGF working team leads contribute to the success of ESGF by implementing their working subcomponent tasks. How these subcomponents should be implemented, in general terms, is addressed in this plan.

Formalizing implementation strategies and guidelines should be at the forefront of our organization's improvement efforts because it:

- Prevents compromises in the quality of software development and execution;
- Minimizes costly events by managing personnel and resources;
- Aids in reporting and tracking of major developments that impact feature development timelines and services;
- Helps develop sponsor and community trust;
- Mitigates and manages data flow and workflow improvements throughout the federation; and
- Helps manage overall expectation by all involved.

The importance of creating an ESGF implementation plan cannot be overestimated. Such a plan helps to guide and inform the steering committee (i.e., funding sponsors), senior executive committee (i.e., chief officers), working team leaders (i.e., day-to-day leaders), support staff (e.g., software developers, engineers, and scientists), project representatives (e.g., CMIP, CORDEX, ACME, and Obs4MIPs), and research/data analysts (i.e., user community). It is important to include representatives from all of those domain perspectives affected by our strategic implementation plan in the plan's development, including both individuals who serve as champions for our strategies and those who may oppose our strategy, so that their concerns may be reflected in this living document.

1.2 Requirements Gathering

Requirements gathering, the act of trying to understand a problem by talking to a selection of actual and potential users, is a critical step for all successful IT projects, and ESGF is no exception. Requirements

gathering necessitate full understanding and documentation of multiple community projects before a solution is built. In our agile project development approach, only a "broad-brush" understanding of the problem is required to start work; the gaps in knowledge are filled in as the project progresses. This approach helps with rapid prototyping and request turnaround.

The most up-to-date and recent ESGF requirements can be obtained from the 2016 5th Annual ESGF Face-to-Face Conference Report, **Chapters 3 (Conference Findings), 5 (ESGF Data Center Requirements and Findings), and 9 (Community Developments and Integration).** Additional requirements can also be found in **Appendices D (CMIP Requirements Document) and H (CMIP6 Requirements from WIP Position Papers).** These high-level requirements demonstrate the connection between community projects and the major tasks of the working teams. They also prioritize the tasks.

As described in the ESGF annual conference reports, over the course of several days of direct interaction, the ESGF team discusses, summarizes progress, and plans future activities in all areas of data management, computing, and workflows for community projects (CMIP5, Obs4MIPs, CORDEX, ACME, etc.). During the conference, it is not uncommon for ESGF team members to:

- Schedule project requirement meetings for particular topic areas (review of old system, interview with users, potential new features, etc.) and work through them with the appropriate high-level community project leads;
- Document community project meeting discussions and report back to the ESGF executive and steering committee on needs and future requests;
- Take the initiative to work through foreseen and unforeseen community needs and formulate additional requirements; and
- Write reports and white papers at the end of the requirements phase, with recommendations to projects.

To close the requirements loop, the ESGF team (i.e., executive committee and working team leads) reports back to the projects on problems, developers devise solutions, and a community of users tests solutions before they are put into production. Additional improvement requirements have been obtained from the community through user surveys.

1.3 System Overview

Figure 1 summarizes four different architectural views of the ESGF system available as of the beginning of 2016. Diagrams such as this help communicate the requirements solution to key stakeholders in the climate community, though not all requirements will be relevant to all the stakeholders. For example, some projects might be interested in the ESGF index node architecture view, while others may be interested in the data node–only view or the security identity provider view or the compute node view. Each view focuses on a specific aspect of ESGF functionality, and there are guidelines on how the implementation of the components is represented and documented for development. The 2016 5th Annual ESGF Face-to-Face Conference Report discusses the various views of the architecture and details the open-source components of the enterprise system constructs. See **Chapters 7 (Technology**

Developments) and 8 (Technology Integration of Interoperable Services). Software architecture documentation is a key responsibility of the working team leads.

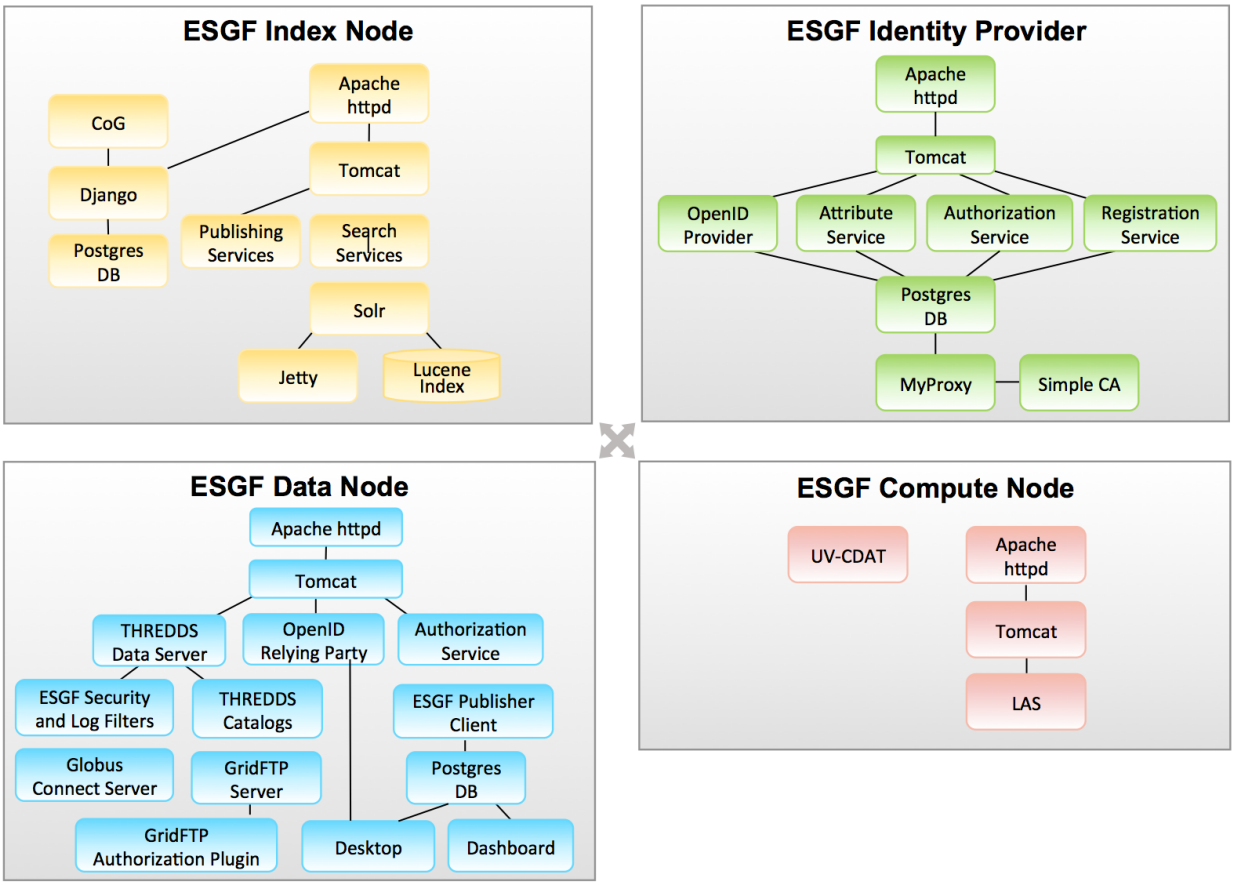

Figure 1. *Current ESGF software stack architecture at the beginning of 2016, representing Release Version 2.2.3.*

1.3.1 System Description

The ESGF system software stack runs on two primary Linux operating systems—Community Enterprise Operating System 6 (CentOS6) and Red Hat Enterprise Linux 6 (RHEL6). System files include libraries of functions, system services, drivers for hardware and networks, system performance and metrics, and specific project configuration files. The programs that are part of the system software include language compilers, file management tools, system utilities, test suites, and performance measures.

The system software stack can be installed on the bare-metal machine or on virtual machines running either of the two supported Linux operating systems. A system administrator can install or update the software to run one or more of the four architectural views (i.e., any combination of the data node, index node, identity provider, and/or compute node) by running the ESGF installation programs. Unlike application programs, however, the ESGF system software stack is not meant to be installed by the end user—it can only be installed by experienced Linux administrators (i.e., ESGF administrators). For example, while the end user might use the ESGF front-end user interface via a Web browser every day, they probably will not have much use for Apache, Tomcat, Postgres, Solr, etc. (unless, of course, they are a computer programmer).

Since the system software stack runs at the most basic level of a computer, it is called "low-level" software. It generates the user interface and allows the operating system to interact with the hardware. Fortunately, the end user does not have to worry about what the system software is doing, as it runs in the background. That is, the end user is performing "high-level" work and, per the requirement, should not have to worry about lower-level details.

For a more in-depth description of the system software and its implementation, see the 2016 5th Annual ESGF Face-to-Face Conference Report, **Chapter 7 (Technology Development) and Appendix G**.

1.3.2 System Organization

The ESGF collaboration formed teams to carry out the various responsibilities inherent in a software development effort. All working team organizations have a structure, even if it is only an implicit one, that defines the roles and responsibilities appropriate to the working team's primary activities. Team structure, including the division of roles, responsibilities, and authority, should reflect the ESGF enterprise goals and directives, the team's culture, the nature of business, and available talent. The working teams' structure is discussed in the 2016 5th Annual ESGF Face-to-Face Conference **Appendix G**. ESGF working teams are listed, along with a brief description, in **Table 1**.

Table 1. *The current list of ESGF working teams and designated working team leads.*

Working Team	Working Team Leads	Team Goals
1. CoG User Interface Working Team	Cecelia DeLuca (NOAA) and Luca Cinquini (NOAA)	Improve ESGF search and data cart management and interface
2. Metadata and Search Working Team	Luca Cinquini (NASA)	Implement ESGF search engine based on Solr5 and discoverable search metadata
3. Publication Working Team	Sasha Ames (DOE) and Rachana Ananthakrishnan	Enable capability for publishing CMIP and other project data sets to ESGF
4. Node Manager Working Team	Sasha Ames (DOE) and Prashanth Dwarakanath (IS-ENES)	Manage ESGF nodes and node communications
4a. Tracking/Feedback Notification Working Team	Sasha Ames (DOE) and Prashanth Dwarakanath (IS-ENES)	Implement user and node notification of changed data in the ESGF ecosystem
5. Identity Entitlement Access Management Working Team	Philip Kershaw (IS-ENES) and Rachana Ananthakrishnan (DOE)	Implement ESGF X.509 certificate-based authentication and improved interface
6. Compute Working Team	Charles Doutriaux (DOE) and Daniel Duffy (NASA)	Develop data analytics capability within ESGF
7. Quality Control Working Team	Martina Stockhause (IS-ENES) and Katharina Berger (IS-ENES)	Integrate external information into the ESGF portal
8. Installation Working Team	Nicolas Carenton and Prashanth Dwarakanath (IS-ENES)	Install the components of the ESGF software stack
9. Dashboard Working Team	Paola Nassisi (CMCC) and Sandro Fiore (IS-ENES)	Monitor the Earth System Grid Federation in terms of system metrics and data usage statistics
10. International Climate Network Working Group	Eli Dart (DOE/ESnet) and Mary Hester (DOE/ESnet)	Increase data transfer rates between the ESGF climate data centers
11. Data Transfer Working Team	Lukasz Lacinski (DOE) and Rachana Ananthakrishnan (DOE)	Enhance ESGF data transfer and web-based download
12. Software Security Working Team	George Rumney (NASA) and Dan Duffy (NASA)	Implement security measures to identify vulnerabilities in the ESGF software and provide continuous improvement to the ESGF software development life cycle.
13. Support Working Team	Torsten Rathmann (IS-ENES) and Matthew Harris (DOE)	Develop frequently asked questions regarding ESGF and housed data
14. Documentation Working Team	Matthew Harris (DOE) and Sam Fries (DOE)	Document the use of the ESGF software stack
15. Replication and Versioning Working Team	Stephan Kindermann (IS-ENES) and Tobias Weigel (IS-ENES)	Create replication tool for moving data from one ESGF center to another; in addition, preserve versioning history of the ESGF published data sets
16. Provenance Capture Working	Bibi Raju (DOE)	Enable ESGF provenance capture for

Working Team	Working Team Leads	Team Goals
Team		reproducibility and repeatability

Individual working teams tend to be fully responsible for technical decisions that affect their subcomponent, but they also must ensure that their efforts will fit into the larger ESGF enterprise. Many of the end-component products run independently as well as in the larger system software stack. Specialization might require some software components to have their own development group, while other components are tightly integrated and pull from many other software development efforts for final implementation.

The role of the ESGF Executive Committee (XC) is to partner with the science domain to help develop use cases, manage all implementations, gather and allocate resources, and provide high-level oversight—all with the end goal of supporting community projects. **Figure 2** shows the high-level organizational chart with the lines of communication between the Steering Committee (i.e., international funding agencies and sponsors) and their principle investigators (i.e., representing the Executive Committee). In keeping this fluid, key advisory panels are established for project-specific requirements gathering. For example, there is a single panel for CMIP6, called the Working Group on Coupled Modeling (WGCM) Infrastructure Panel (WIP), which feeds the ESGF XC its project requirements for working team implementation.

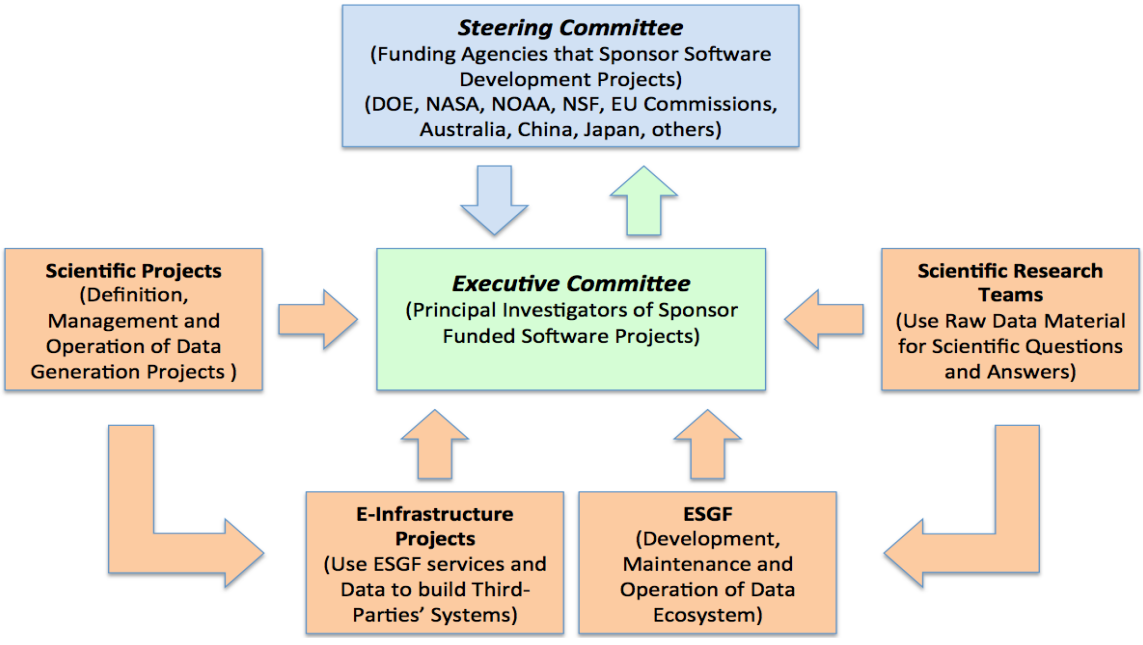

Figure 2. *Managing ESGF's system organization for success.*

Figure 3 shows how both the Steering and Executive committees work closely together with supported projects and non-funded organizations to define and implement new technologies for the scientific community at large—especially those in the areas of data management, distributed computing, networking, analysis, and visualization. The full ESGF governance document can be viewed online at: http://esgf.llnl.gov/governance.html.

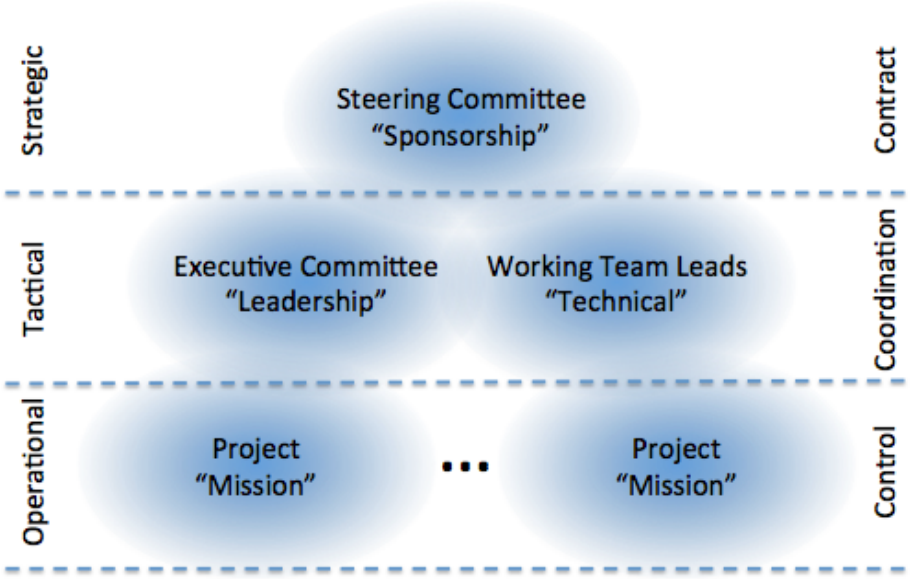

Figure 3. *ESGF's governance structure for process and communication.*

1.4 Project References

As working team projects are completed, many useful project reference documents and project management templates will be posted on the ESGF website (http://esgf.llnl.gov) or the ESGF GitHub site (accessible from the ESGF website - must register before gaining access) for future reference. These may range from project methodology templates to examples of how a particular part of a project was addressed (e.g., implementation plans).

There will also be ESGF and community standard documents, made accessible by supporting and external collaborating software projects (such as ES-DOC, DRS, CMOR, and others), which enables pertinent information to be logged into the system for future reference by external and internal users. For more information regarding external collaborating projects, see the 2016 5th Annual ESGF Face-to-Face Conference Report, **Chapter 9 (Community Developments and Integration) and Appendix H.** Many of these types of documents can be already found on the external ESGF website (http://esgf.llnl.gov) or the internal ESGF Confluence site hosted at LLNL (must register before gaining access). These documents represent a library of information for the old and new working teams to draw upon. It also enables version control of important ESGF reference documents.

2 Short-Term (2016) and Mid-Term (2017–2019) Implementation

At the 5th Annual ESGF Face-to-Face Conference, short-term tasks were described within use cases given by five community project representatives (i.e., CMIP, ACME, Obs4MIPs, CORDEX, and CREATE-IP). These short-term tasks were noted by the projects as high-priority items that must be completed before the end of 2016 in order for their projects to succeed. Longer-term tasks—equally important to the projects when their time comes for implementation—were put into the mid-term time schedule solely due to the time it would take to complete them and resource and service limitations.

Designated working teams implement short-term and mid-term tasks. Resources needed for the implementation come from the ESGF Steering Committee and the climate project communities (e.g., CMIP, Obs4MIPs, CORDEX, ACME).

The working teams' structure is essential to the successful implementation of ESGF tasks. By combining a diverse background of people (i.e., researchers, computer scientists, mathematicians, etc.), these teams can collaboratively achieve goals more quickly and effectively than can individuals working alone. Teams also pool their provided resources to lower costs and to increase task capability throughout the federation. All teams are virtual, with members located in various countries across the globe.

In support of the working teams described in the following sections, the ESGF software security plan (see ESGF Software Security Plan - http://esgf.llnl.gov/media/pdf/ESGF-Software-Security-Plan-V1.0.pdf) is designed to support both major and minor ESGF software releases within the context of the ESGF Software Development Life Cycle (SDLC). The security plan emphasis is on the "release" phase of ESGF (including its prerequisites) and depends upon development and maintenance (design and build) aspects as well. The pre-release ESGF software security plan focuses strongly on:

- Requirements definition;
- Design (including secure coding practices and threat modeling);
- Implementation; and
- Verification (including security testing).

The release phase ESGF software security plan focuses strongly on:

- Inventory update;
- Change documentation;
- Security review (minor | major);
- Issue resolution; and
- Certification of release.

In addition, the ESGF Executive Committee shall charter and oversee a Software Security Working Team (SSWT) whose duties emphasize developing release procedures and helping to guide ESGF along a continuous improvement path to a more secure methodology and architecture.

To communicate, the virtual working teams use low- or no-cost collaborative tools such as Google Docs, WebEx or Go-To-Meeting video and teleconferencing, and Confluence. Each working team is directly or indirectly focused on a specific short- or mid-term task needed by one or more ESGF-supported

community project. For example, the Metadata and Search Working Team develops and improves federation-wide search capability for large and growing data sets (on the order of tens of petabytes of data), while the International Climate Network Working Group (ICNWG) is assisting the Replication Working Team in moving petabytes of data between ESGF sites using the latest network technology.

This section provides a full description of the implementation and major and minor tasks involved in ESGF's short-term (2016) and mid-term (2017–2019) efforts. It includes our working team leads and pulls from past ESGF F2F Conference Reports to fill in the gaps. Note that a summary of the 2016 short-term implementation plan is referenced in the 2016 5th Annual ESGF Face-to-Face Conference Report, **Chapter 10**.

Table 2 shows the working teams, subtasks, status, work priority, and milestones. The summarized implementation table helps to assess the planning process for a task, estimates when a task is completed, and manages dependencies between tasks.

Table 2. *Project implementation plan at a glance. (Priority H – high, M – medium, L – Low)*

Task No.	Task	% Complete	Status	Day Started	Day to be Completed	Actual Date Completed	Priority	Milestone
1	**CoG User Interface Working Team**							
1.1	*Support CoG deployments*	50	On Schedule	01/2016	12/2016		H	
1.2	*CoG with Globus API*	90	Almost Done	02/2016	06/2016		H	
1.3	*Improve speed and responsiveness*	0	Not Started	NA	12/2016		M	
1.4	*Multiple virtual organizations*	0	Not Started	NA	12/2017		H	
1.5	*Integrate with computing analysis and visualization UI*	0	Not Started	NA	12/2019		H	
2	**Metadata and Search Working Team**							
2.1	*Policy for index nodes*	10	On Schedule	01/2016	05/2016		H	
2.2	*Automatic metadata updates*	90	Almost Done	01/2016	05/2016		H	
2.3	*Tag data with multiple activities*	50	On Schedule	03/2016	06/2016		H	
2.4	*Enforce controlled vocabularies*	10	Started	02/2016	12/2016		H	
2.5	*Support other data types*	0	Not Started	NA	12/2017		M	
2.6	*Solr cloud architecture*	50	Ahead of Schedule	01/2016	12/2017		H	
2.7	*Search by location and time*	0	Not Started	NA	12/2017		M	
3	**Publication Working Team**							
3.1	*Recommend changes to publisher tool*	50	On Schedule	01/2016	12/2017		M	
3.2	*Test suite*	75	On Schedule	11/2015	06/2016		L	
3.3	*New esgscan_directory*	75	On Schedule	01.2016	06/2016		L	
3.4	*Drs_lite*	0	Not Started	04/2016	06/2016		L	
3.5	*Support TDS new feature*	0	Not Started	07/2016	09/2016		L	
3.6	*Multiple facet values*	75	Ahead of Schedule	02/2016	06/2016		M	
3.7	*CV integration/CMOR checker*	0	Not	04/2016	06/2016		H	

Task No.	Task	% Complete	Status	Day Started	Day to be Completed	Actual Date Completed	Priority	Milestone
			Started					
3.8	*Support for new data services*	75	Almost Done	01/2016	05/2016		M	
3.9	*Best practices document*	10	Started	04/2016	04/2016		L	
3.10	*Ingest Service updates*						M	
3.11	*Update Ingest Service to replace publisher tool*	0	Not Started	N/A	12/17		L	
4	**Node Manager Working Team**							
4.1	*Shard file generation*	0	Not Started	04/2016	06/2016		M	
4.2	*Vet member nodes*	0	Not Started	04/2016	05/2016		H	
4.3	*Integrate with desktop/dashboard*	10	Design Proposed	01/2016	08/2016		H	
4.4	*Port to Flask*	0	Not Started	N/A	06/2017		L	
4.5	*Standby node mode*	0	Not Started	N/A	03/2017		M	
4.6	*Administrators' console*	0	Not Started	N/A	08/2017		M	
4a	**Tracking/Feedback (Notification) Working Team**							
16.1	*Proof of concept "Cron" job*	0	Not Started	04/2016	09/2016		M	
16.2	*Real-time tracking queries in GUI*	0	Not Started	N/A	12/2017		L	
16.3	*Installer integration*	0	Not Started	N/A	04/2017		L	
5	**Identity Entitlement Access Management Working Team**							
5.1	*Pilot integration of Live Access Service with CEDA ESGF OAuth 2.0 service*	100	Done	02/2016	03/2016	03/2016	H	
5.2	*Implement service discovery mechanism for OAuth 2.0*	5	Design Proposed	03/2016	05/2016		H	
5.3	*Deploy OAuth 2.0 and Online CA services operationally with ESGF Identity Providers*	0	Not Started	06/2016	07/2016		H	
5.4	*Pilot integration of Globus with ESGF OpenID Connect service*	5	Design Proposed	11/2016	12/2016		M	
5.5	*Retire MyProxyCA service (replaced by Online CA service)*	0	Not Started	10/2016	10/2016		M	
5.6	*Implement and Integrate OpenID Connect into ESGF*	0	Not Started	09/2016	12/2016		M	
5.7	*Retire OpenID 2.0 service*	0	Not Started	N/A	03/2017		H	
5.8	*Implement Attribute (role) registration web service interface*	0	Not Started – may require reprioritization	01/2017	03/2017		H	
5.9	*Implement support for sign-in using other federation identities, e.g., InCommon and ESA identity federation*	0	Not Started	02/2017	06/2017		L	
5.10	*Evaluate Assent for use with ESGF—enables integration of Eduroam with Single sign-on*	0	Not Started	06/2017	09/2017		L	
6	**Compute Working Team**							
6.1	*Finalize Version 1 of WPS API*	50	On Schedule	01/2016	04/2016		H	
6.2	*Create and finalize API for computational kernels*	10	On Schedule	03/2016	06/2016		M	

Task No.	Task	% Complete	Status	Day Started	Day to be Completed	Actual Date Completed	Priority	Milestone
6.3	*Deploy proof of concept at LLNL*	50	On Schedule	01/2016	07/2016		M	
6.4	*Define standard data set and unit tests*	0	Not Started	04/2016	08/2016		M	
6.5	*Update to use cases*	0	Not Started	04/2016	10/2016		M	
6.6	*Deploy second proof of concept at GSFC*	40	On Schedule	01/2016	12/2016		M	
7	**Quality Control Working Team**							
7.1	*Errata Service: Fix errata concept (WIP paper)*	0	Not Started	04/2016	05/2016		H	
7.2	*Citation Service: Create development-operational instances; APEX GUI availability for modelling centers*	0	Not Started	04/2016	05/2016		M	
7.3	*Errata Service: Fix issue information design + registration test on remote issue tracker and Handle Service + Django errata module development*	0	Not Started	05/2016	06/2016		M	
7.4	*Citation Service: Revise APEX GUI; integrate citation information in ESGF CoG; finalize citation landing page design and content*	0	Not Started	05/2016	06/2016		M	
7.5	*Errata Service: Register operable issues + integrate CoG errata module*	0	Not Started	06/2016	09/2016		M	
7.6	*Citation Service: Add functionality to citation service (early citation is operational)*	0	Not Started	06/2016	09/2016		L	
7.7	*Draft on recommendations for the integration of external information into ESGF*	0	Not Started	06/2016	09/2016		L	
7.8	*Errata Service: full operability*	0	Not Started	09/2016	12/2016		M	
7.9	*Citation Service: Integrate citation into LTA/IPCC-DDC and DataCite DOI process*	0	Not Started	09/2016	12/2016		M	
7.10	*Final version of recommendations for the integration of external information into ESGF*	0	Not Started	09/2016	12/2016		L	
8	**Installation Working Team**							
8.1	*ESGF admin guide review—configuration part*	0	Not Started	N/A	04/2016		H	
8.2	*ESGF test suite review—CoG support*	15	Test suite reviewed – CoG support ongoing	N/A	05/2016		H	
8.3	*Integrate publication test suite written by PWT*	0	Not Started	N/A	06/2016		H	
8.4	*Make Globus Online credentials storage optional during installation*	0	Not Started	N/A	08/2016		H	
8.5	*RHEL/CentOS/SL 7 Support*	0	Not Started	N/A	10/2016		H	
8.6	*Migrate tomcat component installation to RPM*	0	Not Started	N/A	2017		M	
8.7	*Develop new Python based installation script*	0	Not Started	N/A	2017		M	
8.8	*Implement continuous integration*	0	Not	N/A	2017		L	

Task No.	Task	% Complete	Status	Day Started	Day to be Completed	Actual Date Completed	Priority	Milestone
	based on existing continuous build infrastructure and apps		Started					
8.9	*Dockerize the data node stack*	0	Not Started	N/A	2017		L	
9	**Dashboard Working Team**							
9.1	*Coarse-grained statistics system*	95	Almost done	09/2015	04/2016		H	
9.2	*Project-specific statistics for Obs4MIPs and CMIP5*	50	On Schedule	02/2016	05/2016		H	
9.3	*Design and first implementation of federated statistics views*	20	On Schedule	02/2016	10/2016		H	
9.4	*Integration of perfSONAR statistics into the dashboard*	0	Not Started	N/A	12/2016		H	
9.5	*First implementation of new front-end presentation layer*	0	Not Started	N/A	12/2017		M	
9.6	*Extended set of statistics based on user requirements*	0	Not Started	02/2016	12/2019		M	
10	**International Climate Network Working Group**							
10.1	*Deploy perfSONAR at sites*	10	Started	06/2015	12/2016		H	
10.2	*Publish data with GridFTP URLs*	50	Started	01/2016	06/2016		H	
10.3	*Configure perfSONAR tests at selected sites*	0	Not Started	N/A			M	
10.4	*Deploy DTNs*	50	Started	06/2015	12/2016		H	
10.5	*Test Synda replication in collaboration with replication team*	50	Started	02/2016	06/2016		H	
10.6	*Scale up DTNs*	0	Not Started	N/A	12/2018		H	
10.7	*Create agreed reference design for next-generation ESGF deployment*	0	Not Started	N/A	12/2018		M	
11	**Data Transfer Working Team**							
11.1	*Deploy Globus Connect Server with automated certificate provisioning as the default installation option on all data nodes in ESGF*	80	In Progress	N/A	5/2016		H	
11.2	*Integrate Globus Transfer support for public data sets via CoG into the installer and deploy at various nodes*	70	In Progress	N/A	5/2016		H	
11.3	*Implement Globus Transfer support for restricted data sets. Integrate with the ESGF installer and make it available for deployment*	15	In Progress	N/A	8/2016		H	
11.4	*Integrate Globus Transfer as an option for data replication using Synda. Test and integrate with Synda. Participate in performance testing, in collaboration with ICNWG for optimization of performance replication*	60	In Progress	N/A	10/2016		H	
11.5	*Integrate use of OAuth for delegation of credentials to Globus, for improved user experience for browser-based flows using Globus for data transfer. Implement, test, and deploy this feature*	0	Not Started	N/A	11/2016		M	
12	**Software Security Working Team**							
12.1	*Coordinate audits of the ESGF software suite release*	0	As-Needed	N/A	N/A		H	

Task No.	Task	% Complete	Status	Day Started	Day to be Completed	Actual Date Completed	Priority	Milestone
	candidates using static and dynamic tools and code inspection as necessary							
12.2	*Document findings and aid in issue resolution for all findings of moderate or higher impact*	0	As-Needed	N/A	N/A		H	
12.3	*Inform and assist the ESGF XC in assessing risks related to findings*	0	As-Needed	N/A	N/A		M	
12.4	*Support the ESGF Risk Executive in performing their function*	0	As-Needed	N/A	N/A		M	
12.5	*Document ESGF Site best practices regarding protective measures*	0	Contained in Software Security Plan	4/2016	Goal: DRAFT approval + 3 mo.		M	
12.6	*Participate in documenting the "as-is" state of the ESGF software suite build process*	0	Not Started	TBD (upon DRAFT approval)	Goal: DRAFT approval + 6 mo.		M	
12.7	*Maintain both the major and minor release procedures*	0	On-Going	N/A	N/A		L	
13	**Support Working Team**							
13.1	*Create support documentation plan*	0	Not Started	N/A	4/2016		M	
13.2	*Revise documentation structure*	0	Not Started	N/A	5/2016		M	
13.3	*Revise content of ESGF User Guide*	0	Not Started	N/A	10/2016		H	
14	**Documentation Working Team**							
14.1	*Upgrade appearance and design*	0	Not Started	N/A	09/2016		M	
14.2	*Refactor and reorganize documentation, splitting into user-, administrator-, and developer-centric hubs*	0	Not Started	N/A	11/2016		M	
14.3	*Coordinate with SWT to refactor and reorganize wiki*	0	Not Started	N/A	12/2016		M	
15	**Replication and Versioning Working Team**							
15.1	*PID service performance and reliability tests*	30	On Schedule	03/2016	04/2016		M	
15.2	*Install and test Synda installations at DTNs of core sites*	5	Started	03/2016	06/2016		M	
15.3	*Initial deployment of PID infrastructure components and publication workflow tests at multiple sites*	0	Not Started	05/2016	10/2016		H	
15.4	*Integrate and test Synda with Globus at core sites*	0	Not Started	N/A	07/2016		M	
15.5	*Define consistency requirements between replica sets at core sites and define policies ensuring these consistency requirements*	0	Not Started	N/A	08/2016		L	
15.6	*Complete full integration of PID services with web GUI*	0	Not Started	07/2016	10/2016		M	
15.7	*Test large transfers between core sites based on Synda and work on optimizing end-to-end transfer bandwidth*	0	Not Started	N/A	11/2016		M	
15.8	*Complete initial integration of replica publication at core*	0	Not Started	N/A	11/2016		M	

Task No.	Task	% Complete	Status	Day Started	Day to be Completed	Actual Date Completed	Priority	Milestone
	sites to the Synda-based replication workflow							
15.9	*Integrate replication and versioning with PID infrastructure as part of ESGF CMIP6 publication workflow*	0	Not Started		11/2016		M	
16	**Provenance Capture Working Team**							
16.0	*Under development*	0	Not Started	N/A	N/A		H	

2.1 ESGF/CoG User Interface Working Team

2.1.1 Description of Implementation

CoG is a Python/Django web application that is now deployed at each ESGF node as the main front end to the back-end data and user services. Some of its major features include support for hosting multiple scientific projects within the same installation, federating with other ESGF nodes, enabling several collaboration tools for scientists, searching and downloading ESGF distributed data, and exposing a formal process for describing and administering projects.

2.1.2 Points of Contact

Team Lead: Cecelia DeLuca, NOAA Global System Dynamics (cecelia.deluca@noaa.gov)
Team Lead: Luca Cinquini, NASA Jet Propulsion Laboratory (luca.cinquini@jpl.nasa.gov)

2.1.3 Major Tasks

Short-term (2016)

- Support CoG deployment and use throughout the federation (e.g., provide help to administrators installing CoG, February–April 2016).
- Establish and revisit guidelines for configuring CoG sites across ESGF (March–April 2016).
- Integrate CoG with Globus Application Programming Interface (API) for downloading restricted data sets (April–June 2016)
- Integrate ESGF ingestion services for publishing smaller data sets from a large community of principal investigators (July–August 2016).
- Implement requirements provided by the WIP in support of CMIP6, including display of quality flags, persistent identifiers (PIDs), digital object identifiers (DOIs), and errata; and integrate these with Earth system documentation (ES-DOC) model metadata (ongoing through December 2016).
- Improve speed and responsiveness of application, especially the search interface and user federation (November–December 2016)
- Continue supporting CoG users throughout the federation, improve and extend the online documentation (ongoing).

Mid-term (2017–2019)

- Enable versioning of wiki documents (2017).
- Implement model for supporting multiple CoG federations in different scientific domains (climate, medicine, hydrology, etc.) (2017).
- Develop new Python-based script for CoG installation (2017).

- Integrate with ESGF workflow services for distributed computation (2018–2019).
- Integrate with ESGF analysis and visualization services (2018–2019).
- Upgrade the underlying JavaScript library to a different or more recent framework (2017–2019).
- Implement other requirements as they emerge and are prioritized from ESGF administrators and users (ongoing).

2.1.4 Security Overview

As the main front end to the software stack, CoG is the likely first target of any attack directed to the distributed ESGF infrastructure. Therefore, CoG development and operations must follow strict guidelines for web application security. The Django framework itself offers several features for protection against the most common attacks, and it will be kept up-to-date with the latest releases to continuously leverage the improved security support. Additionally, each CoG release is subjected to a dynamic scan through the JPL web application software after it is deployed in the test environment. Static and code scans will also be executed at regular intervals, as established by the ESGF SSWT, as will cross-checking against the public database of known common vulnerability and exposures (CVEs).

2.2 ESGF Metadata and Search Working Team

2.2.1 Description of Implementation

The ESGF Metadata and Search Working Team is responsible for the development, operation, and support of the server-side services used to publish and search metadata throughout the federation. The software is composed of a Java servlet web application that serves as the front end for an Apache/Solr search engine. ESGF uses advanced Solr features such as multiple cores, distributed searches, and automatic replication. The Publishing service is exposed to clients through a legacy Hessian protocol and through a newer representational state transfer (RESTful) API that supports PUSH and PULL operations. Security is based on X.509 certificates. The Search service can be used either through the CoG web interface, or directly by clients through its RESTful API.

2.2.2 Points of Contact

Team Lead: Luca Cinquini, NASA Jet Propulsion Laboratory (luca.cinquini@jpl.nasa.gov)

2.2.3 Major Tasks

Short-term (2016)

- Support deployment of ESGF publishing and search services across the federation (March–April 2016).
- Establish policy for establishing an official ESGF Index Node. Change ESGF installer default to not install an ESGF index node (May 2016).
- Develop tools and services to support atomic metadata updates (May 2016).
- Support tagging of data sets for multiple projects (June 2016).
- Support indexing of CMIP6 data sets (ongoing 2016–2017)
- Revise and improve documentation, especially the RESTful ESGF query syntax (June 2016).
- Implement data validation against controlled vocabularies (July–December 2016)
- Make continuous upgrades to the Solr distribution, including defining a process for migrating the metadata indexes (ongoing).

- Implement and prioritize other requirements to search back end and front end (CoG) as they emerge from CMIP6 and other projects (ongoing)

Mid-term (2017–2019)

- Discontinue support for legacy Hessian publishing services (early 2017).
- Support partitioning of search space across multiple virtual organizations (e.g., ESGF and ACME) (2017).
- Support indexing, searching, and retrieval of other types of data besides Network Common Data Form (netCDF; e.g., Hierarchical Data Format, GRIdded Binary, images, movies, etc.) (2017).
- Enable searches by geophysical location (2017).
- Establish Solr-Cloud architecture to scale to 10–100 times the current metadata volumes (2017). Develop installer for ESGF Solr-Cloud deployment. Author documentation for node administrators (2017).
- Develop new Python-based installer for ESGF search components (Solr and ESGF search web application) (2018).
- Integrate metadata and search service within broader ESGF workflows running on distributed nodes, including tasks for data transfer, analysis, and visualization (2018–2019).
- Generalize ESGF publishing and search services to other scientific domains (2017–2019).
- Experiment with other big data search technologies and engines, such as NoSQL and MongoDB (2017–2019).

2.2.4 Security Overview

The ESGF publishing and search services are deployed as a Java servlet web application within the Apache/Tomcat container. This working team will follow the standard guidelines established by the ESGF Software Security Working Team for development of web applications. Additionally, this component will be subjected to the standard ESGF verification process before each major release, which includes static and dynamic scans, as well as cross-checking all third party libraries with the list of known CVEs.

2.3 ESGF Publication Working Team

2.3.1 Description of Implementation

ESGF publication meets several needs through corresponding software products. The publish client tool is a collection of Python scripts that enable data publishing, unpublishing, metadata queries, and updates, all of which run on the data node, while the Ingest services allow for remote publishing via graphical user interface (GUI) or API interfaces. Additionally, there are external publication-oriented scripts that comprise a comprehensive publication workflow effort.

2.3.2 Points of Contact

Team Lead: Sasha Ames, LLNL/Department of Energy (DOE) (sasha@llnl.gov), Publication tool
Team Lead: Rachana Ananthakrishnan, ANL/DOE UChicago (ranantha@uchicago.edu), Ingest services

2.3.3 Major Tasks

Short-term (2016)

- Write best practices document (April 2016).

- Release publisher to support new data services: HPSS and Globus URLs (May 2016).
- Make schema changes to support publisher integration with Errata and PID services (June 2016).
- Implement Test Suite (June 2016).
- Implement new esgscan_directory tool for mapfile generation (June 2016).
- Implement new drs_lite tool for versioning and DRS management (June 2016).
- Introduce multiple facet values functionality (June 2016).
- Integrate CV (ini file generation) and CMOR checker (June 2016).
- Make changes to support TDS new features (September 2016).
- Introduce Ingest Service features:
    - Integrate with CoG for UI publication to be a feature of CoG (April 2016).
    - Add replicated data publication as a first class flow (June 2016).
    - Support facet management and esg.ini files (April 2016).
    - Support dataset versioning (July 2016).

Mid-term (2017–2019)

- Implement 2015 recommended changes to esg-publisher scripts (December 2017).
- Migrate of TDS and Solr publishing into ingest service (service no longer calls publisher client; late 2017?).
- Migrate additional publisher features (TBD) to ingest service (early 2018).
- Capture provenance regarding the publication process; include in metadata (2017).
- Coordination with other activities that make use of publishing (e.g., automated replication, publication, and version control) (2018).
- Retire esg-publisher tool in favor of ingest service (for remote and local publication) (2018?).
- Improve support for new science domains and data formats (e.g., images) (ongoing).
- Make changes to complement updates of new searchable metadata types (ongoing).
- Performance test ingest services to inform the automated replication and publisher process (including data movement) (2018).
- Perform flexible quality checking beyond CMOR checker for CMIP6 (2018).
- Implement performance metrics of the publication process (based on live metrics) for tuning (2018).
- Integrate pattern discovery/machine learning processes that are triggered by the publication process (2019).

2.4 ESGF Node Manager, Tracking, and Feedback Working Team

2.4.1 Description of Implementation

The Node Manager is composed of (1) a Django module that implements a REST API called by other Node Managers or administrator clients and (2) a Python "daemon" that initiates status checks and action items. A directory-based task queue (single consumer) bridges the two components.

2.4.2 Points of Contact

Team Lead: Sasha Ames, LLNL/DOE (sasha@llnl.gov)
Team Lead: Prashanth Dwarakanath, National Supercomputer Center at Linköping University (NSC/LiU) (pchengi@nsc.liu.se)

2.4.3 Major Tasks

Short-term (2016)

- Determine shard file generation requirements (still needed?) (April 2016).
- Implement shard file generation (if needed) (June 2016).
- Test communication patterns in virtual environment (May 2016).
- Test Version 0 in test federation (April 2016).
- Vet member nodes (May 2016).
- Initiate production release of Version 0 (TBD in 2016).
- Perform Dashboard/desktop integration for metrics sharing (August 2016).
- Track and provide feedback on proof of concept "cron" service (September 2016).
- Test release Version 1 (features not in Version 0, 2016).

Mid-term (2017–2019)

- Initiate production release of Version 1 (TBD in 2017).
- Port API to Flask (mid 2017).
- Stand up supernode feature (early 2017).
- Implement administrators' console (mid-year).
- Integrate with other ESGF services (evaluate needs on case-by-case basis, all TBD in 2017):
    - CoG
    - Search
    - Publisher
    - Identity, Entitlement, and Access Management (IdEA) services
    - Compute
    - Future installation and component version management stack
- Integrate (with CoG) tracking control interface to enable cron script settings or immediate tracking queries (TBD in 2017).
- Install Tracking Feedback module (Early 2017).
- Replace esg-node control command scripts? (TBD, in 2018).

2.4.4 Security Overview

The Node Manager is based on Django. We will update the package to recent stable versions that address security concerns. Moreover, the module will undergo security scans as was done for other web components within the software stack.

2.5 ESGF Identity, Entitlement, and Access Management Working Team

2.5.1 Description of Implementation

The IdEA system (see **Figure 4**) secures access to resources within the ESGF federation (typically data, but looking to the future, also computational resources). Access is mediated via components for authentication (OpenID 2.0 for browser-based Single Sign-on and MyProxyCA for Command Line Interfaces) and authorization (Security Assertion Markup Language [SAML] Attribute and Authorization Services, respectively). Role-based access control is employed, whereby resources are secured by rules based on access roles. A given authority governs access to resources spanning multiple nodes, a concept known as virtual organizations (VOs). For example, the Program for Climate Modeling and

Intercomparison is responsible for a VO, which governs access to CMIP5 data served from multiple ESGF nodes around the world.

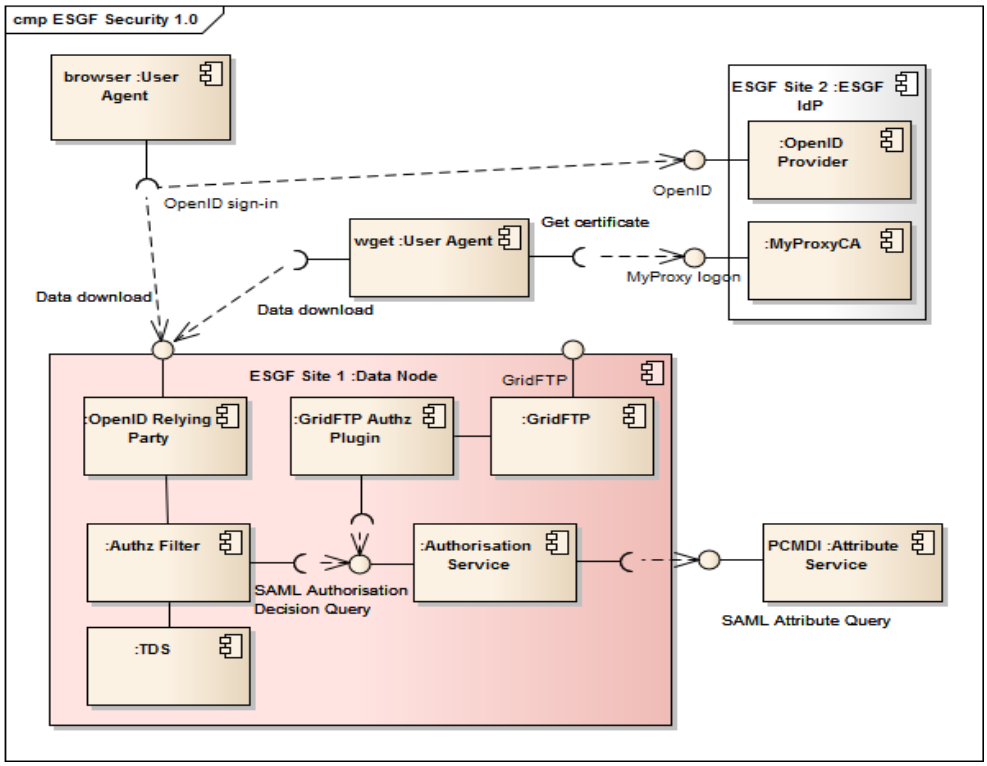

Figure 4: *ESGF IdEA major components.*

2.5.2 Points of Contact

Team Lead: Philip Kershaw, Centre for Environmental Data Analysis (CEDA)/ Science and Technology Facilities Council (STFC) (Philip.kershaw@stfc.ac.uk)
Team Lead: Rachana Ananthakrishnan, ANL/DOE UChicago (ranantha@uchicago.edu)

2.5.3 Major Tasks

The major activity for ESGF IdEA is to integrate support for OAuth 2.0 as part of wider migration from OpenID 2.0 to OpenID Connect. This is critical to ESGF's development and security because OpenID 2.0 is now in the process of being deprecated across the IT sector.

Short-term (2016)

- Pilot integration of Live Access Server (LAS) with ESGF OAuth 2.0 service deployed at CEDA. This will enable LAS to obtain delegated user credentials to access secured data from ESGF services (March 2016).
- Implement service discovery mechanism for OAuth 2.0[1]. This is an important milestone in deprecating OpenID 2.0[2] and replacing it with OAuth 2.0 (May 2016).
- Deploy OAuth 2.0 operationally with ESGF Identity Providers (IdPs) (July 2016).

[1] https://www.rfc-editor.org/rfc/rfc6749.txt
[2] http://openid.net/specs/openid-connect-core-1_0.html

- Pilot integration of Globus with ESGF OAuth 2.0 service deployed at CEDA, which would enable full integration of Globus Transfer for data download (via GridFTP) with ESGF access control. This is dependent on the status of Globus support for OAuth 2.0. If not possible, integration will be deferred until ESGF can support OpenID Connect (August 2016).
- Retire MyProxyCA. An existing HTTP-based SLCS replaces it (October 2016).
- Implement and integrate OpenID Connect into ESGF. OpenID Connect is a direct replacement to OpenID 2.0. It builds on the OAuth 2.0 solution (ongoing to December 2016).

Mid-term (2017–2019)

- Retire OpenID 2.0. OpenID Connect service replaces it (March 2017).
- Roll out OpenID Connect in operational federation (2017).
- Implement the Attribute Registration web service interface. This will provide greater flexibility for the management of registration of users for access to resources (2017).
- Implement a SAML *bridge* to the ESGF Identity Provider to reduce the number of identity providers that need to be run within the federation. This will enable user sign-in using institutional credentials for users whose home organizations are part of Shibboleth[3] federations (e.g., US *InCommon* federation and European Space Agency federation) (2017).
- Evaluate the *Assent*[4] system for use with ESGF IdEA. Assent enables single sign-on via Eduroam (*Edu*cation *Roam*ing), the popular system used in the academic sector for obtaining Internet connectivity. Based on a positive assessment and resourcing, support for Assent will be integrated into ESGF IdEA. (2017–2018).
- Investigate replacement of X.509-based user authentication for command line interpreter access. Public key infrastructure–based user authentication, though robust, can be complex for end users to configure. Solutions such as two-legged OAuth 2.0, application passwords, and the SAML ECP[5] may provide simpler alternatives and should be evaluated with a view to full integration into ESGF (2018–2019).

2.5.4 Security Overview

Currently, the system uses the Java Spring framework for OpenID interfaces, together with Java implementations of SAML services. The new CoG interface implemented with the Python Django framework also includes an OpenID interface. MyProxyCA is implemented in C and is distributed as a Red Hat Package Manager (RPM). The system will be kept secure by working closely with the ESGF Software Security Working Team and follow guidelines set out by this team. This will include regular review and updating of the software, including monitoring of CVEs, notices, and advisories related to package dependencies and through regular operating system patching for host machines.

A key driver for the roadmap is to ensure the system reflects current industry practice and to promptly retire technologies as they are superseded by newer alternatives. The migration from OpenID 2.0 to OpenID Connect is an important example. In this way, system security can be better maintained.

[3] https://shibboleth.net/

[4] https://jisc.ac.uk/assent

[5] https://wiki.oasis-open.org/security/SAML2EnhancedClientProfile

2.6 ESGF Compute Working Team

2.6.1 Description of Implementation

The Compute Working Team (CWT) is engaged in developing the capability to enable data-proximal analytics throughout ESGF. The team started working through several potential analytical use cases for the data stored within ESGF and is using relatively simplified operators to define and create an Application Programming Interface (API). An initial version of the API has been created and is being made available as a Web Processing Service (WPS). Proof of concept implementations of the WPS and API are being put into place using basic functions such as model ensemble averaging and anomalies.

The implementation team's aim is to continue to develop the API and expand the proof of concepts into production level prototypes to be deployed across multiple ESGF sites to facilitate a basic compute service as a part of the ESGF stack. The compute service will enable access to varying levels of compute capabilities, from single servers to large-scale computational systems, at the sites where data is generated or stored. Access to these services will be through scripts, thick client graphical interfaces, and smart-client web browsers, such as CDAT-Web.

2.6.2 Points of Contact

Team Lead: Charles Doutriaux, Lawrence Livermore National Laboratory (doutriaux1@lln.gov)
Team Lead: Dan Duffy, NASA (Daniel.q.duffy@nasa.gov)

2.6.3 Major Tasks

Short-term (2016)

- Finalize Version 1 of the WPS API definition for the prototype use case (anomaly) and simple canonical operations (such as subsetting, average, maximum, and minimum) (April 2016).
- Create and finalize an API for the computational kernels to be used by the WPS API (June 2016).
- Deploy proof-of-concept server at LLNL with serial implementations of analysis operations (June 2016).
- Connect the initial proof of concept server at LLNL with the CDATWeb Interface (July 2016).
- Define a standard set of climate data to be initially exposed with a set of unit tests, data, input, and output to be used to verify the implementation of the WPS API (August 2016).
- Update use cases registered on Confluence (November 2016).
- Deploy second proof-of-concept server at GSFC with the initial use cases (December 2016).

Mid-term (2017–2019)

- Expand the number of proof-of-concept nodes to other sites (2017).
- Implement parallel analytics within existing proof-of-concept nodes (2017).
- Begin to explore federated analytic capabilities that will enable operations to act across data sets that are only accessible on a single ESGF node (2017).
- Create a Version 2 of the WPS API and computational kernels (2017).
- Complete full security vetting of the Version 1 specification and computational kernels, create production prototypes, and perform end-user testing (2017).
- Integrate the compute services as part of the ESGF compute node installation (2017).
- Ensure tighter integration with CDATWeb (2017).
- Create production-level services of the Version 1 specification (2018).

- Complete full security vetting of Version 2 specification and computational kernels, elevate the Version 2 implementation to production prototypes, and perform end-user testing (2018).
- Create a Version 3 of the WPS API and computational kernels (2019).
- Perform full security vetting of the Version 2 specification and computational kernels, create production prototypes, and perform end-user testing (2019).

2.6.4 Security Overview

The service's security is critically important and must be closely monitored. Rather than enable end users to run arbitrary code on compute nodes within an ESGF site, the CWT is developing standard operations (such as averaging and anomalies) that users will be able to execute through the API. Thus, the only applications that are being run on the compute nodes are ones that have been written and fully security vetted by the ESGF community.

The exposure of these services is through a standard WPS, which is a well-known and utilized web service throughout the geospatial community. This will be built using familiar frameworks, such as Python, Django, or Play, with the back-end computational operations potentially being written in a variety of different languages. Fortunately, the framework offers several features for protection against the most common attacks, and it will be kept up to date with the latest releases to continuously leverage the improved security support.

Additional methods will need to be implemented across multiple ESGF sites to ensure the compute node cannot be abused and turned into a tool for denial-of-service attacks. These include test, development, and pre-production capabilities. The test and development compute capabilities will be enabled for use by the CWT developers, with tightly controlled access. As new capabilities are added, several steps will need to happen: code reviews must be held to ensure the security of the resulting service is adequate, any security concerns identified must be addressed, and new capabilities must be pushed into pre-production for more intensive testing, prior to being pushed into full production.

2.7 ESGF Quality Control Working Team

2.7.1 Description of Implementation

The ESGF Quality Control Working Team aims to improve the quality of ESGF user services with regard to additional (external) documentations. This implies the storage of selected metadata for unpublished events for provenance as well as the integrity of data citations. The storage of external metadata/ancillary metadata in the ESGF index and its integration in the ESGF search enables other external services (e.g., furtherInfoUrl/ES-DOC or the long-term data archival/Intergovernmental Panel on Climate Change Data Distribution Center [IPCC-DDC]) to automatically access this ancillary metadata.

2.7.1.1 Metadata Persistence

Data consistency within the ESGF necessitates a revision of the publication process to preserve core information about previous data set versions and provenance information on the data set revisions. This in turn requires integrating ancillary information from non-ESGF CMIP6 services (e.g., errata, citation) into the ESGF data publication process and the ESGF Services Search API and CoG portal, reviewing their accessibility at user interfaces, and ensuring the transparency of data versions.

2.7.1.2 Errata Information

The ESGF Quality Control Working Team aims to define and to establish a stable and coordinated procedure to collect and give access to errata information related to data sets hosted by ESGF. Building such a system requires new tools and will imply including dependencies to other ESGF components (a remote Git repository, the PID Handle Service, and the CoG front end).

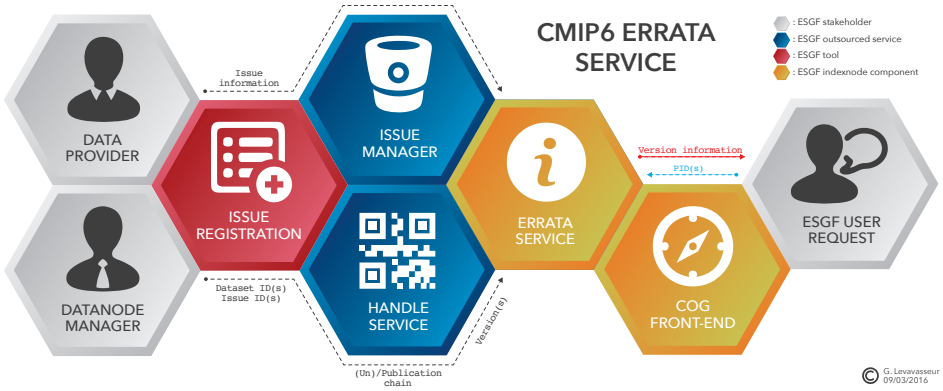

Figure 5: *CMIP6 Errata Service architecture.*

2.7.1.3 Data Citation Service

The data citation service, cmip6cite.wdc-climate.de, aims to provide data citation information for humans (citation landing page and integration in ESGF CoG user portal) as well as machines—e.g., CMIP6 services such as the long-term IPCC-DDC archive or the landing page of the furtherInfoUrl page or data citation services like OpenAire. Components of the data citation service include an Oracle database with an APEX GUI, an API to access citation information in XML and JSON, a data citation landing page and a metadata set on the Open Archives Initiative server for harvesting.

The integration of data citation information into ESGF requires the incorporation of ancillary metadata registration in ESGF publisher as an ESGF publication process, the display of core citation information in CoG portal together with ancillary metadata link to data citation landing page, and the integration of data set versions and ancillary information in the ESGF Search and CoG portal. For data citation consistency, metadata for unpublished revised data sets need to be displayed upon user request. Other interfaces to be supported are to furtherInfoUrl/ES-DOC/CIM, LTA/IPCC-DDC, and DataCite.

2.7.1.4 Integration of Additional External Information

The ESGF-QCWT's approach to integrating external information can be used to integrate information provided by other services, e.g., quality control/assurance results or information on ES-DOC information simulation and modeling.

2.7.2 Points of Contact

ESGF Implementation: Katharina Berger, DKRZ (berger@dkrz.de)
Errata Information: Guillaume Levavasseur, IPSL (glipsl@ipsl.jussieu.fr)
Data Citation Service: Martina Stockhause, DKRZ (stockhause@dkrz.de)

2.7.3 Major Tasks

Short-term (2016)

2.7.3.1 Metadata Persistence

- [Katharina Berger] Support versioning (August).
- [Luca Cinquini/Katharina Berger] Ensure persistence of Solr Index (August–September).

2.7.3.2 Errata Information

- [Guillaume Levavasseur] Deployment of an outsourced issue tracker (May).
- [Guillaume Levavasseur] Provide an ESGF client to record issue (June).
- [Atef Bennasser] Develop an errata module (June).
- [Atef Bennasser] Integrate errata module into CoG (September).

2.7.3.3 Data Citation Service

- Set up basic components in development state: Citation database, API Service (JSON, XML), landing page, APEX GUI (in testing phase), and test integration of citation information in CoG portal discussed and agreed with CoG team (March 2016).
- [Martina Stockhause] Insert initial model citation metadata (May 2016).
- [Martina Stockhause/H-H. Winter] Complete APEX GUI release (June 2016).
- [Martina Stockhause/M. Kurtz] Design Citation Landing Page and finalize content (June 2016).
- [Katharina Berger] Integrate citation information in ESGF (June 2016).
- [Martina Stockhause] Add functionality (e.g. references to citation service and integrate CV of DRS components (September 2016).
- [Katharina Berger/M. Stockhause] Add quality control for citation information completeness and integration of LTA metadata (September 2016).
- [Martina Stockhause] Integrate citation into LTA/IPCC-DDC and DataCite DOI process (December 2016).

2.7.3.4 Integration of Additional External Information

- [Martina Stockhause] Discuss with ES-DOC and Quality Control joining the ESGF-QCWT (April – December 2016).
- [Martina Stockhause] Discuss with external services using ancillary data, e.g. furtherInfoUrl (June–December 2016).

2.7.4 Implementation Schedule

Table 3. *ESGF-QCWT implementation schedule.*

Dates	Description
From April–May	Errata Service: Fix errata concept by final WIP Paper review Citation Service: Create development operational instances Insert initial model citation metadata Ensure APEX GUI availability for modelling centers
From May–June	Errata Service: Fix issue information design + registration test on remote issue tracker and Handle Service + Django errata module development Citation Service: Revise APEX GUI; integrate citation information in ESGF CoG; Finalize citation landing page design and content
From June–September	Errata Service: Enable operable issue registration + errata module CoG integration Citation Service: Add functionality to citation service (early citation is operational)

Dates	Description
	Complete draft on recommendations for the integration of external information into ESGF
From September–December	Errata Service: Full operability Citation Service: Integrate of citation into LTA/IPCC-DDC and DataCite DOI process Complete final version of recommendations for the integration of external information into ESGF

2.7.5 Security Overview

The libraries used by the framework will be always kept up to date with the latest release in order to avoid bugs and vulnerabilities that may arise over time. In addition, each release of the QCWT framework is designed and implemented to follow strict guidelines for web application security.

2.8 ESGF Installation Working Team

2.8.1 Description of Implementation

The IWT is responsible for the development and maintenance of the ESGF installer, which is a set of bash scripts executed by ESGF node administrators to install or upgrade, configure, and run their ESGF nodes. The team is also responsible for packaging the modules that make up the stack and for managing releases. These activities rely on building tools such as a Jenkins server, and also rely on testing tools such as a test federation and a test suite.

2.8.2 Points of Contact

Team Lead: Nicolas Carenton, IPSL (nicolas.carenton@ipsl.jussieu.fr)
Team Lead: Prashanth Dwarakanath, NSC/LiU (pchengi@nsc.liu.se)

2.8.3 Major Tasks

Short-term (2016)

- Establish and revisit guidelines for configuring ESGF sites after installation (March–April 2016).
- Review the test suite and rewrite the user creation and group registration module to fit the CoG web interface (April–May 2016).
- Integrate the publication test suite written by PWT (June 2016).
- Make Globus Online credentials storage optional. Some centers are willing to run a GridFTP server as part of their ESGF node and disagree with the Globus registration part, which does require to store credentials on Globus servers (August–September 2016).
- Support CentOS/RHEL/Scientific Linux 7. Version 6's deprecation is scheduled for November 2020. If nodes OS are not migrated from 6 to 7 prior to any CMIP6 data set publication, then sites will have to go through OS migration and data republication during a critical CMIP6 phase (October–November 2016).

Mid-term (2017–2019)

- Switch to RPM-based installation for tomcat component (2017).
- Develop new Python-based script for ESGF installation (2017).
- Dockerize the data node stack (2017).
- Implement continuous integration using existing continuous build workflow and infrastructure (2018–2019).

2.9 ESGF Dashboard Working Team

2.9.1 Description of Implementation

The Dashboard Working Team is responsible for the design and the implementation of a distributed and scalable monitoring framework used to collect and visualize local and federated data usage statistics about the Earth System Grid Federation. The framework is composed of a back end, which acts as an information provider collecting and storing all the metrics in the system catalog, and a web front end, which provides the user with a comprehensive view of the statistics at different level of granularity.

2.9.2 Points of Contact

Team Lead: Paola Nassisi, Euro-Mediterranean Center on Climate Change Foundation, (paola.nassisi@cmcc.it)
Team Lead: Sandro Fiore, Euro-Mediterranean Center on Climate Change Foundation, (sandro.fiore@cmcc.it)

2.9.3 Major Tasks

Short-term (2016)

- Finalize the design phase of the data warehouse system to support cross-project and project-specific fine-grained statistics (February 2016).
- Complete official release of the coarse-grained statistics system (including IdP-based AuthN) (April 2016).
- Implement the new cross-project statistics view and related back-end libraries and integrate them into the dashboard module (April 2016).
- Create new project-specific views and back-end libraries for Obs4MIPs and CMIP6 (May 2016).
- Implement REST APIs single-node base to support federated-level statistics (May 2016).
- Complete first implementation of the federated statistics (October 2016).
- Complete final implementation of the federated statistics (November 2016).
- Complete extended set of views with geo-location (federation-level) statistics (November 2016).
- Complete new front-end presentation layer design (December 2016).
- Complete design of the perfSONAR statistics integration into the dashboard module (December 2016).

Mid-term (2017–2019)

- Complete new front-end presentation layer implementation (first release in 2017, new releases 2018–2019).
- Complete new view for publication/SOLR statistics (2017).
- Complete extended set of statistics according to new user requirements (2017–2019).
- Integrate perfSONAR data into the ESGF monitoring framework extending the set of views (2017 first release, 2018–2019 updated releases).
- Improve support for statistics related to new projects (2017) and science domains (2018–2019).

2.9.4 Security Overview

As a web application, the front end of the monitoring framework could be the target of a possible attack to the federation. For this reason, the libraries used by the framework will be always kept up to date with the latest release in order to avoid bugs and vulnerabilities that may arise over time. In addition, each release of the framework is designed and implemented to follow strict guidelines for web

application security. With regard to the past, the dashboard interface will be also extended and secured through an authentication module that will allow only registered users to get access to the statistics views. Moreover, it is subjected to a dynamic scan through the JPL web application software in order to highlight potential security gaps and correct them before the final release in the production environment. Static and code scans will also be executed at regular intervals as established by the ESGF SSWT.

2.10 ESGF International Climate Network Working Group

2.10.1 Description of Implementation

The International Climate Network Working Group (ICNWG) works to use network infrastructures and resources to easily and efficiently enable large-scale data transfers between key climate data repository centers. The ICNWG is primarily responsive to CMIP and ACME project requirements. The participating working group centers include ANL, CEDA, DKRZ, IPSL, KNMI, LLNL, NCI, and ORNL. ICNWG collaboratively works with each center to understand the existing network infrastructures and policies, and works on an appropriate infrastructure design for the sites focusing on two use cases for network resources: 1) the large-scale replication of data between the centers; and 2) easy and efficient end-user (climate researchers') access data. ICNWG is a cross-collaborative working group with other ESGF teams such as the Replication and Versioning Working Team. A separate document describes the ESGF site requirements for node participating in the CMIP project.

2.10.2 Points of Contact

Team Lead: Eli Dart, ESnet, Lawrence Berkeley National Laboratory (dart@es.net)
Team Co-Lead: Mary Hester, ESnet, Lawrence Berkeley National Laboratory (mchester@es.net)

2.10.3 Major Tasks

Short-term (2016)

- Work with participating sites (DKRZ, CEDA, NCI, LLNL, IPSL and KNMI) to deploy perfSONAR servers (or virtual instances) (June 2016).
- Each participating site will need to configure perfSONAR tests to measure network performance that will be viewable from a perfSONAR dashboard at icnwg.es.net. This will help ensure that the physical layer and connections between sites are performing as expected (June 2016).
- Work with each site to deploy four high-performance DTNs designated for data transfers connected at 10 gigabits per second at each of the participating sites (June 2016).
- Work with participating sites to publish GridFTP URLs and use Synda for initial replication tests (June 2016).
- Run and publish tests to the broader ESGF community for operational replication workflow (July 2016).

Mid-term (2017–2019)

- Each site should plan to deploy additional DTNs to scale with data requirements. This will also mean an upgraded connection to a wide-area network to 40 gigabits per second to achieve 1 petabyte/week transfers (with a throughput of 14 gigabits per second) (2018).
- Create an agreed-upon reference design for ESGF center network infrastructure deployment that

can handle the increasing scale of climate data ingest and egress (2018).

2.10.4 Security Overview

Participating ICNWG sites are requested to work with their local ICT representatives to develop a high-performing network infrastructure plan that fits within their institution's security policies. Based on past experiences, it is best to have designated data transfer servers located near the institution's network perimeter for data ingest and egress while using access control lists as the main security policy instead of an enterprise firewall.

2.11 ESGF Data Transfer Working Team

2.11.1 Description of Implementation

The Data Transfer Working Team is responsible for the components for data access and transfer. This team handles a few primary use cases: end user data download from the ESGF archive, data transfer for publication, and data replication across selected data nodes within the federation. Both HTTP/S and GridFTP servers are supported for moving data, and all ESGF data nodes install both these servers by default. This team also maintains the authorization callouts needed to secure the data download pieces. The team has strong ties with the replication and version working team and the aforementioned ICNWG.

Access methods for end users to the data include script-based download and use of Globus Transfer for managed and high-performance download via the browser interface. The Globus Transfer interface is an option supported in CoG. Synda is the replication tool of choice, and integration of high-performance data movement options is a high-priority issue for this team.

2.11.2 Points of Contact

Team Lead: Rachana Ananthakrishnan, Argonne/DOE UChicago (ranantha@uchicago.edu)
Team Lead: Lukasz Lacinski, Argonne/DOE UChicago (lukasz@uchicago.edu)

2.11.3 Major Tasks

The major activities for ESGF Data Working Team is to make Globus Transfer available as managed data transfer for all data sets in ESGF and to make Globus Transfer available as the high-performance data transfer option for replication use cases.

Short-term (2016)

- Deploy Globus Connect Server with automated certificate provisioning as the default installation option on all data nodes in ESGF (May 2016).
- Integrate Globus Transfer support for public data sets via CoG into the installer and deploy at various nodes (May 2016).
- Implement Globus Transfer support for restricted data sets. Integrate with the ESGF installer and make it available for deployment (August 2016).
- Integrate Globus Transfer as an option for data replication using Synda. Test and integrate with Synda. Participate in performance testing, in collaboration with ICNWG for optimization of performance replication (October 2016).

- Integrate use of OAuth for delegation of credentials to Globus, for improved user experience for browser-based flows using Globus for data transfer. Implement, test, and deploy this feature (November 2016).

Mid-term (2017–2019)

- Support installation of new data nodes with data transfer components, use of Globus at replication sites, publication services, and client support via other interfaces (scripts, CoG, etc.)
- Complete integration and support of Globus Transfer as an option in Synda in production.
- Investigate use of Globus Sharing for data access in ESGF, eliminating the need for user certificates for data download. Prototype and test for evaluation.
- Investigate use of Globus HTTP/S support for data access in ESGF to streamline the data access mechanism, access control checks, and clients, irrespective of protocol. Prototype and test for evaluation.
- Investigate use of Globus Transfer and sharing by third-party portals/applications developed on the ESGF ecosystem. Support integration of Globus with such applications.

2.12 ESGF Software Security Working Team

2.12.1 Description of Implementation

The ESGF SSWT provides support for both major and minor ESGF software releases within the context of the ESGF SDLC. This includes both pre-release and release phases, with strong emphasis on ensuring secure and stable code releases. The security plan's emphasis is on the "release" phase of ESGF (including the software prerequisites) and depends upon ESGF development and maintenance (design and build) aspects as well. The pre-release ESGF software security plan focuses strongly on requirements definition, design (including secure coding practices and threat modeling), implementation, and verification (including security testing). The ESGF release phase software security plan focuses strongly on inventory updates, change documentation, security review (minor | major), issue resolution, and certification of release. In addition, duties shall include helping to guide ESGF along a continuous improvement path to a more secure methodology and architecture. See the ESGF Software Security Plan (http://esgf.llnl.gov/media/pdf/ESGF-Software-Security-Plan-V1.0.pdf) for details.

2.12.2 Points of Contact

Team Lead: George Rumney, NASA (george.rumney@nasa.gov)

Team Lead: Dan Duffy, NASA (daniel.q.duffy@nasa.gov)

2.12.3 Major Tasks

See the ESGF Software Security Plan for the current, complete list of activities. The following is a condensed summary.

Short-term (2016)

- Coordinate audits of the ESGF software suite release candidates using static and dynamic tools and code inspection as necessary (as needed in 2016).
- Document findings and aid in issue resolution for all those of moderate or higher impact (as needed in 2016).
- Inform and assist the ESGF XC in assessing risks related to findings (as needed in 2016).
- Support the ESGF Risk Executive in performing his/her function (as needed in 2016).

- Document ESGF Site best practices regarding protective measures (July 2016).
- Participate in documenting the "as-is" state of the ESGF software suite build process (September 2016).
- Maintain both the major and minor release procedures (ongoing in 2016).

Mid-term (2017–2019)

- Collaborate with ESGF sites in enhancing security best practices as an integral part of the ESGF SDLC (ongoing).
- Participate in defining a "to-be" state of the ESGF software suite (2016–2017).
- Collaborate with ESGF in implementing a transition from the "as-is" to "to-be" ESGF build process (2017).
- Enhance the major and minor release procedures in response to the evolution of the ESGF software suite (ongoing).

2.13 ESGF Support Working Team

2.13.1 Description of Implementation

The ESGF Support Working Team is responsible for responding to all user questions sent to the ESGF-user mailing list as well as developing, maintaining, and supporting ESGF user documentation. This currently consists of the ESGF website, CoG tutorial, user FAQs, and Wiki. Of course, all written help should easily be accessible. The best solution would be to have all documentation for end users in one place. Additionally, new techniques interesting to end users—for example, downloads with Globus and CMIP6-related improvements—should be documented as soon as they are available.

2.13.2 Points of Contact

Team Lead: Matthew Harris, Lawrence Livermore National Laboratory (harris112@llnl.gov)
Team Lead: Torsten Rathmann, DKRZ (rathmann@dkrz.de)

2.13.3 Major Tasks

Short-term (2016)

- Create a support documentation plan, including, if possible, the move of user information and help to CoG (April 2016).
- Revise documentation structure according to support documentation plan (May 2016).
- Revise the content of the ESGF User Guide (September–October 2016).
- Furnish additional documents for end users concerning Globus, PIDs, CMIP6 errata, and DOIs (as soon as available).

Mid Term (2017–2019)

- Continue operational user support and documentation of ESGF improvements for end users (ongoing 2017–2019).

2.13.4 Security Overview

The ESGF support websites are deployed as Jekyll static websites and are hosted at GitHub.

2.14 ESGF Documentation Working Team

2.14.1 Description of Implementation

The ESGF Documentation Working Team is responsible for managing the documentation generated by other working teams. The team manages esgf.llnl.gov, which offers additional documents, such as sponsors, acknowledgements, governance, committees, publications, tutorials, supported projects, wikis, and much more.

2.14.2 Points of Contact

Team Lead: Sam Fries, Lawrence Livermore National Laboratory (fries2@llnl.gov)
Team Lead: Matthew Harris, Lawrence Livermore National Laboratory (harris112@llnl.gov)

2.14.3 Major Tasks

Short-term (2016)

- Upgrade appearance and design at specific nodes (August 2016).
- Refactor and reorganize documentation, splitting into user-, administrator-, and developer-centric hubs (November 2016).
- Coordinate with SWT to refactor and reorganize wiki (December 2016).

Mid-term (2017–2019)

- Continue identifying team-generated documentation and linking out to it from esgf.llnl.gov (ongoing 2017–2019).

2.14.4 Security Overview

ESGF documentation is entirely written in markdown and static HTML. There are no immediate vectors for attack, and much of the documentation is hosted on GitHub's wiki system, so there is no threat to federation servers.

2.15 ESGF Replication and Versioning Working Team

2.15.1 Description of Implementation

Data replication covers the transport of large data aggregates between ESGF nodes. Specific "core nodes" are responsible for maintaining a consistent replica set (consistency requirements include versioning information). The replication manager tool used is Synda (developed at IPSL), supporting http- and GridFTP-based transfers. Synda integration with Globus is done in the ESGF data transfer working team. Replication activities to core sites also include close cooperation with the ICNWG team to optimize the transportation paths (e.g., ESGF data nodes → DTNs) and end-to-end transfer capacities.

Replication and versioning (for CMIP6) is supported by integration with the new (ESGF/CMIP6) PID infrastructure supporting CMIP6 data management. Implementation of the PID infrastructure aims to make all involved components highly scalable and reliable and integrate them well with the existing ESGF components such as the publisher and catalog without introducing additional barriers to publication. The PID infrastructure is also used by the errata services, and development for this strong interaction has already begun.

2.15.2 Points of Contact

Team Lead: Kindermann Stephan, DKRZ (kindermann@dkrz.de)

Team Lead: Tobias Weigel, DKRZ (weigel@dkrz.de)

2.15.3 Major Tasks

Short-term (2016)

- Test performance and reliability of PID services accomplished and optimization options determined (April 2016).
- Install and test Synda installations at DTNs of core sites (June 2016).
- Initial deployment of PID infrastructure components and publication workflow tests at multiple sites (July 2016).
- Integrate and test Synda with Globus (July 2016).
- Define consistency requirements between replica sets at core sites and define policies ensuring these consistency requirements (August 2016).
- Full integration of PID services with web GUI (search/CoG, PID-specific information services) (October 2016).
- Test large transfers between core sites based on Synda and work on optimizing end-to-end transfer bandwidth (November 2016).
- Complete initial integration of replica publication at core sites to the Synda-based replication workflow (this includes PID replica registration) (November 2016).
- Integrate replication and versioning with PID infrastructure as part of ESGF CMIP6 publication workflow (November 2016).

Mid-term (2017–2019)

- Coordinate "pull mode" replication activities at core sites (e.g. by maintaining a versioned central repository of Synda configurations used at sites) and agreement on notification mechanism triggering automatic data pull activities (early 2017).
- CoordinateCMIP6 replication team, responsibilities at sites, problem escalation handling, etc.
- Start with operational CMIP6 replication (early 2017).
- Integrate data/metadata consistency checks as part of replication (early 2017).
- Integrate PID infrastructure with replication infrastructure (PID changes trigger replication policies, and replication activities update PID information) (late 2017).
- Establish automatic replication procedure between core sites (early 2018).
- Complete first version of a command-line information tool for displaying PID record information on data objects (early 2017).

2.15.4 Security Overview

No security plan is needed for replication and versioning development.

3 Implementation Support

Certain basic facilities, including computer equipment and supplies, are needed to stand up an ESGF site. Historically, ESGF sites have not allocated adequate resources and administrative support to the federated petascale multi-project archive. Moreover, agencies or projects sometimes receive or acquire superfluous resources or resources poorly matched to their needs with the ESGF enterprise. ESGF sites may also undertake comprehensive reappraisal of projects that can significantly increase or decrease the size of staff (particularly ESGF administrator support) and support facilities. Defining ESGF implementation needs is therefore essential for good project, agency, and site planning.

3.1 Site Descriptions and Requirements

Facilities may have different levels and types of resources, but they often face common challenges. This document recommends several strategies for addressing hardware, network, and software requirements and managing facility resources for the overall good of the federation. It also includes implementation requirements by site for primary data centers. Together, there are over 40 federated ESGF sites throughout the globe.

Each ESGF site may participate in one or more projects, such as CMIP6, ACME, Obs4MIPs, CORDEX, and others. Projects like CMIP6 and ACME have substantially different infrastructure requirements than do projects like Obs4MIPs and CORDEX. Sites, therefore, need to respond to their supported project requirements. CMIP6 requirements are captured in a separate document. The minimal resource requirements for all ESGF sites are given below:

- ***Storage:*** Has adequate data storage to serve their data to their external community;
- ***Front-end Linux server(s):*** Has a Linux platform running CentOS6 or RHEL6 operating system;
- ***Software:*** Installs and operates the ESGF basic software stack which includes Postgres, Tomcat, Apache, and TDS;
- ***Staff:*** Has a designated, part-time ESGF administrator.

Certain ESGF nodes will also function as “super-nodes”, and provide more extensive services that are essential to the reliable operation of the ESGF. These sites have the following additional requirements:

- ***Metadata Index:*** Has a set of Solr engines and corresponding back-end indexes to publish and search the data, including a master/slave configuration for local holdings, plus replica shards for all other nodes in the federation;
- ***Compute:*** Has shared compute clusters/servers for remote data reduction, derived data, large-scale data manipulation and visualization, and data streaming;
- ***Identity provider:*** Serves as an identity provider for ESGF Single Sign-On access control attributes and user authentication X.509 certificates;
- ***Node Manager:*** Runs a “super-node” instance of the node manager (once deployed);
- ***Network:*** Connects to the wide-area network at 10 gigabits per second with designated GridFTP/Globus-enabled servers, a configured Synda installation, and a perfSONAR instance for network performance measurement in a Science DMZ infrastructure;

- ***Software:*** Installs and operates the entire ESGF software stack includes Postgres, Apache httpd, Tomcat, Solr, THREDDS Data Server (TDS), LAS, and Ultra-scale Visualization Climate Data Analysis Tools (UV-CDAT); and

Sites that are primarily data repositories are encouraged to leverage some of the services supported by super-node sites, such as a metadata index and identity provider, and focus instead on supporting local services for data download and possibly analysis.

3.2 Hardware, Software, Facilities, and Materials

Adequate hardware, software, facilities, and materials must be provided to ESGF sites for simple installation and project use. Moreover, the basic aspects/foundation of the enterprise system—composed of hardware, networks, and software—work together harmoniously to form the fundamental overall success of ESGF and the projects that it supports. Choosing the appropriate hardware and operating system is key for consistent and sustainable operations. Currently, the ESGF software stack (i.e., application) has been ported to hardware that supports RHEL6 or CentOS6. (Porting ESGF to RHEL7 or CentOS7 is under development.)

For ESGF operations, the operating system and the ESGF software stack must be reviewed frequently—at least every six months—for compatibility, software security, version control, and expectation of future support. Both the operating system and ESGF software upgrades are thoroughly tested on test federation sites before going into production. Appropriate ESGF security measures are taken before each major and minor software release. Details of the ESGF software security scanning measures can be viewed in the ESGF Software Security Plan (http://esgf.llnl.gov/media/pdf/ESGF-Software-Security-Plan-V1.0.pdf).

This section list supported software, materials, equipment, and facilities required for implementation.

3.2.1 Hardware

Computer hardware used for ESGF operations consists of the central processing unit and the various peripheral units, including compute clusters and mass storage devices. In general, any hardware ranging from the largest high-performance computer to the smallest laptop can run the ESGF software stack—just as long as the hardware is running RHEL6 or CentOS6. Other flavors of Linux may run ESGF; however, the ESGF team will **not** support this effort, and there are **no** security guarantees when running the software for unsupported platforms (**i.e., run at your own risk**).

ESGF works well with two forms of data storage hardware: rotating hard disk drives and magnetic tapes. For faster data retrieval and manipulation, it is recommended that data are stored on rotating hard disk. Such is the case for CMIP data, which are distributed throughout the federation on petabyte-scale disk farms (i.e., a large room or rooms filled with disk drives). Secondary storage may be internal or external to the primary compute server and takes the form of magnetic tapes (i.e., HPSS). With HPSS, data requests are staged to a temporary scratch space. Once the data are all there, an e-mail message is sent to the user indicating that the data are ready for download or data manipulation. The storage in scratch space is temporary; therefore, the user has a limited time to retrieve requested data.

For most supported ESGF projects, storage (whether rotating hard disk drives or magnetic tapes) is very limited, and thus short-, mid-, and long-term storage planning is required for data management. For example, careful planning and completion of a formal **CMIP6 Data Storage and Replication Plan** will be required to ensure adequate space will be available to hold the tens of petabytes of CMIP6 data, which will include PIDs and DOIs for quick and transparent reference of storage locations. This plan will evenly distribute the petabytes of CMIP6 data throughout the federation's storage hardware for fast data access and manipulation. Automated replication and derivation of additional variables can demand significantly more disk space than CMIP5, and so before these features are implemented, sufficient primary and secondary storage must be furnished to support the projected CMIP6 data files and derived data and allow for future expansion.

3.2.2 Software

ESGF's computer software (the programs that interact with the user) helps manage the federated databases, aid in the evaluation process, prepare and send assessment notices, generate metrics reports, produce visual plots, and provide other output. ESGF software also allows the use of the Internet and worldwide web for use in searching and accessing data, furnishing compute resources, and communicating with researchers and peers. There are three basic categories of ESGF's computer software that are needed for implementation: operating system software (i.e., RHEL6 or CentOS6), utility software (e.g., ESGF Autoinstaller), and application software (i.e., ESGF software stack).

The operating system is generally supplied with the computer and manages overall operation of the system, including file maintenance and communications between the processor and peripherals. ESGF focuses its application software to run under RHEL6 and CentOS6, two types of operating systems that have become standard in the industry.

ESGF utility software performs such routine functions as installing, replicating, and moving files; managing the simultaneous operation of multiple peer-to-peer nodes; diagnosing hardware, network, and software issues, and reporting back issue status. AppSEC security scanning software searches for destructive viruses and security breaches. Some utility software is developed by the federation, while other components are provided by the computer system vendor and purchased separately by one or more of the supporting ESGF sponsors (e.g., Atlassian's Confluence and JIRA project management and planning software).

The ESGF custom application software performs the actual functions required by the user. Multiple teams working in concert customize and implement the tools that enable federated database management, data transfers, security, metadata and search capability, state-of-the-system evaluation, provenance capture, and much more. (See **Table 2** for the exhaustive list of custom application software tasks needed for the success of ESGF.)

Table 1 details, by jurisdiction, the custom ESGF software designed to perform specific tasks, and which can be specifically tailored to meet project-specific requirements. Most ESGF software packages fall in this category. For example, the CoG front-end and esg.ini configuration files can often be customized to reflect actual or desired project requirements, and the prompts and help information can be tailored to

reflect project terminology and convention. All ESGF custom software "source code" is under the Berkeley Software Distribution (https://opensource.org/licenses/BSD-2-Clause) open-source license agreement. This imposes minimal restrictions on the redistribution of the ESGF software stack.

The major disadvantages of ESGF custom software are the time and expense of writing and testing the subcomponents. Particular attention must be paid to ensuring that user requirements are clearly conveyed to programmers and reflected in the end product, which should not be accepted until proper testing has been completed. Future modifications to the subcomponents, even those of a minor nature, can involve institutional and system administrator approval—as seen in the scanning for security vulnerabilities—and can be a time-consuming, costly, and rigorous job.

3.2.3 Facilities

Identifying and providing basic physical facilities and accommodations required for ESGF operations is left up to each ESGF site. However, some site facilities should include assembling and testing of hardware, software, networks, and other components as noted above, and will constitute a test network for the ESGF. ESGF has a procedure for verifying and testing before components are added to operations.

3.2.4 Reference Material

ESGF reference materials are needed to promote compliance with policy and regulations, uniformity in operations and procedures, and adherence to generally accepted assessment principles and practices by all. Along with this living document, the following kinds of materials are living document and are kept current and readily available for access:

- ESGF Governance Policy (http://esgf.llnl.gov/governance.html);
- Logo Requirement and Usage Guidelines (http://esgf.llnl.gov/logo_requirements.html);
- ESGF Strategic Roadmap (http://esgf.llnl.gov/media/pdf/2015-ESGF-Strategic-Plan.pdf);
- Software Security Plan (http://esgf.llnl.gov/media/pdf/ESGF-Software-Security-Plan-V1.0.pdf);
- ESGF Federation Policies and Guidelines (under development);
- User Training Plan (under development);
- Root Certificate Authorities Policy (under development);
- Data Storage and Replication Plan (under development); and
- ESGF Readiness Document (under development).

3.3 Personnel, Personnel Requirements, and Staffing

Organization of the ESGF deployment working teams and assignment of specific roles to team members is managed by the ESGF XC and assisted by the ESGF working team leads. The personnel and resources needed for implementation depend on the size of the working team and the complexity of the deployment. Therefore, assessment of the core competencies of the ESGF research and IT staff and their skill sets with regard to current and future ESGF technologies is essential. Suggestions for managing personnel training issues include:

- Delay the deployment until the staff is fully trained in the new technologies.

- Outsource portions of the work to cover weak points, and then have ESGF staff members learn the required skills from the contracted personnel.
- Outsource the deployment, support, and maintenance of the ESGF enterprise, if necessary.

The ESGF XC represents the overall management of the ESGF enterprise, which is critical for success. It is responsible for ensuring that the deployment working team understands and achieves its goals and has the proper personnel and staffing to carry out its mission. Although software development staffing needs can change with the implementation plan and vice versa, ESGF operations require the ongoing support of site administrators and help desk and support personnel. The ESGF project has always needed people who are trained in its operations, thoroughly understand the peer-to-peer environment, and are familiar with hosted data sets.

Core working teams are made up of experts in security, networking, interoperability, applications testing, and domain scientists. Team members must have detail-oriented project management skills, hands-on technical experience, and the ability to be innovative and master new technologies quickly and independently. ESGF XC members have strong analytical abilities and have the ability to link the ESGF's project vision with the details needed to achieve its mission and goals.

ESGF project scope and objectives are documented in the 2016 5th Annual ESGF Face-to-Face Conference Report. In great detail, it identifies the working teams and their responsibilities for planning and testing the deployment of the features needed for supported projects.

3.3.1 Training of Implementation and Operational Staff

This section addresses the training, if any, necessary to prepare staff for implementing and maintaining the system; it does not address user training, which is the subject of the **User Training Plan**. Described is the type and amount of training required for each of the following areas, if appropriate, for the system:

- System hardware, network, and software installation
- System support
- System maintenance and modification

The training curriculum will list the courses that will be provided, a course sequence, and a proposed schedule. For modeling centers wanting to stand up an ESGF node, it will identify which courses should be attended by which staff members, as determined by job position description. These courses will provide course names and outlines of the content and material; describe who the course is intended to train; identify required resources, support materials; and highlight the instructor's expertise train to teach the course(s).

3.4 Performance Monitoring

As mentioned in Section 2.9, the Dashboard is the ESGF module responsible for managing and visualizing the ESGF usage statistics. It provides an easy access to data and statistics related to the nodes of the federation, such as clients' distribution, monitoring information (nodes status, registered users and deployment), real time or aggregated system status information, and data usage statistics according different views (for example, by time, model, variable, or experiment). Gathering and visualizing such statistics is essential for evaluating how the data provided through the federation are shared among

users, how many data have been downloaded over time, and what the most interesting data are for the users.

In order to display metrics such as throughput, packet delay, packet loss, packet traces, and additional data types are being added all the time. During 2016, the design for the integration between the Dashboard and perfSONAR components will be completed, and preliminary unit tests will be performed to validate specific links and interactions between the two components. Data performance monitoring such as errata information, PIDs, and DOIs will be provided through the CoG interfaces to help ensure data quality and enable provenance capturing.

3.5 Configuration Management Interface

For publications, projects are configured by means of the esg.ini file. We are transitioning toward using single esg.ini for each project (or sub-project) with a distinct configuration, rather than a single, global esg.ini file. That global file will persist for legacy modules that require it in its established location but will be generated from a template during node installation and updated by the node administrator when node-specific changes occur. The project-specific configurations are to be managed by a project lead. The configuration will either be defined using the controlled variable (CV) service or by configuring a master esg.ini for that project or relevant project subset and managed in GitHub (or other web-based revision control system). The file will be managed manually by node administers interacting with GitHub (e.g., clone, pull of the repo), or the ESGF Node Manager will be used (once the feature is complete) to propagate the file changes as they occur.

For the user front end, CoG allows each hosted project to configure its home page, documentation, logos, and searching differently. ESGF administrators have agreed on a set of common configuration options for the main CoG project that is hosted at their sites. This is the project that users first see when they access that ESGF node at a particular host site. These conventions are documented on the web (see https://www.earthsystemcog.org/projects/cog/software/) and leverage configuration files that are maintained on the ESGF GitHub repository.

For user registration and authentication, ESGF implements identity provider white list configurations in esgf_idp.xml and esgf_idp_static.xml files as well as OpenID known provider configurations in the esgf_known_providers.xml file. For publication permissions and access permissions, ESGF implements registration and attribute service configurations in esgf_ats.xml and esgf_ats_static.xml as well as policies configurations in esgf_policies_common.xml and esgf_policies_local.xml. For distributed search, ESGF implements solr shard configurations in esgf_shards.xml and esgf_shards_static.xml files.

The node manager will be used (once the feature is complete) to propagate the files changes as they occur. But *_common.xml and *_static.xml files will be managed manually by the ESGF node administrator interacting with GitHub and will be used to override propagated files configuration if needed.

4 Site Implementation Requirements

One of the ESGF XC's responsibilities is to define and track the success of implementations to determine ESGF's impact's on the scientific and technical communities which it supports, and at the same time, help in the planning climate project's priorities and directions. By definition, our metrics for success for the implementations will gauge and track the quantifiable acceptance and use of our software components by the science communities. These metrics can be directly collected through observation, such as the number of days to complete research, or the number of collaborative software components found useful by groups, projects, or individuals. Metrics will be derived directly from observable quantities, such as the thousands of users logging onto the ESGF infrastructure, downloading data, using collaborative analysis tools, and possibly interacting with each other through discussion forums, blogs, and other social means. When monitoring collaborative research tools in a complex system (such as ESGF), the dashboard effort will be used to assess collaborative projects and to monitor the health of the enterprise infrastructure system.

More broadly, because of the intense interest in metrics within the climate community and ESGF, the ESGF XC will span the entire subfield of metrics to capture the following main categories:

- Adoption of our new implementations within the climate community.
- Adoption of our new implementations within other scientific communities.
- The number of users and data sets recorded in our registries/archives, the frequency of interaction with those registries/archives, and other similar metrics.
- Success stories about the ease and user friendliness of our ESGF infrastructure system, and the additional functionality and power provided as compared with the previous infrastructure; or conversely, the number of bugs filed in a Help Desk or similar application, identifying the system's limitations and drawbacks.
- Response to external material documents for additional community project inclusion.

APPENDIX 1: GLOSSARY

Table 4. *A glossary list of all the terms and abbreviations used in the manual.*

Acronym	Definition
ACME	Accelerated Climate Modeling for Energ —DOE's effort to build an Earth system modeling capability tailored to meet the climate change research strategic objectives (climatemodeling.science.energy.gov/projects/accelerated-climate-modeling-energy/).
ANL	Argonne National Laboratory—Science and engineering research national laboratory near Lemont, Illinois, operated by the University of Chicago for DOE (www.anl.gov).
API	Application programming interface (en.wikipedia.org/wiki/Application_programming_interface/).
CA	Certificate authority (en.wikipedia.org/wiki/Certificate_authority).
CEDA	Centre for Environmental Data Analysis—Serves the environmental science community through four data centers, data analysis environments, and participation in numerous research projects that support environmental science, advance environmental data archival practices, and develop and deploy new technologies to enhance data access (www.ceda.ac.uk).
CentOS	Community Enterprise Operating System—Open-source Linux flavor operating system (www.centos.org/).
CMCC	Fondazione Centro Euro-Mediterraneo sui Cambiamenti Climatici (Euro-Mediterranean Center on Climate Change Foundation)—CMCC Foundation is a non-profit research centre (www.cmcc.it) that aims at furthering knowledge in the field of climatic variability, including causes and consequences, through the development of high-resolution simulations and impact models (www.cmcc.it/cmccesgf-data-node/).
CMIP	Coupled Model Intercomparison Project—Sponsored by the World Climate Research Programme's Working Group on Coupled Modeling, CMIP is a community-based infrastructure for climate model diagnosis, validation, intercomparison, documentation, and data access (cmip-pcmdi.llnl.gov).
CMOR	Climate Model Output Rewriter—Comprises a set of C-based functions that can be used to produce Network Common Data Form files that comply with climate forecast conventions and fulfill many requirements of the climate community's standard model experiments (pcmdi.github.io/cmor-site/).
CoG	The name of the content management system primarily developed by NOAA (www.earthsystemcog.org).
CORDEX	Coordinated Regional Climate Downscaling Experiment—Provides global coordination of regional climate downscaling for improved regional climate change adaptation and impact assessment (www.cordex.org).
CREATE-IP	Collaborative REAnalysis Technical Environment Intercomparison Project—Data collection, standardization, and ESGF distribution component (rthsystemcog.org/projects/create-ip/).
CV	Controlled variable (en.wikipedia.org/wiki/Control_variable).
CVE	Common vulnerability and exposures
DOE	U.S. Department of Energy—Government agency chiefly responsible for implementing energy policy (www.doe.gov).
DKRZ	Deutsches Klimarechenzentrum (German Climate Computing Centre)—Provides high-performance computing platforms and sophisticated, high capacity data management and services for climate science (www.dkrz.de).
DTN	Data Transfer Node

Acronym	Definition
ESGF	Earth System Grid Federation—Led by Lawrence Livermore National Laboratory, a worldwide federation of climate and computer scientists deploying a distributed multipetabyte archive for climate science (esgf.llnl.gov).
ESnet	Energy Sciences Network
EU	European Union—A unique economic and political partnership between 28 European countries that together cover much of the47ikipenent (europa.eu/index_en.htm).
GFDL	Geophysical Fluid Dynamics Laboratory—NOAA's GFDL develops and uses mathematical models and computer simulations to improve the understanding and prediction of atmospheric, oceanic, and climatic behaviors (www.gfdl.noaa.gov).
GUI	Graphical user interface
IdEA	Identity, Entitlement, and Access Management
IPCC-DDC	Intergovernmental Panel on Climate Change Data Distribution Center
IS-ENES	Infrastructure for the European Network for Earth System Modeling—Second-phase project of the distributed e-infrastructure of models, model data, and metadata of the European Network for Earth System Modelling (is.enes.org).
IPSL	Institut Pierre-Simon Laplace—Nine-laboratory French research institution whose topics focus on the global environment. Main objectives include understanding (1) the dynamic chemical and biological processes at work in the Earth System, (2) natural climate variability at regional and global scales, and (3) the impacts of human activities on climate (www.ipsl.fr/en/).
IT	Information technology (en.wikipedia.org/wiki/Information_technology).
JPL	Jet Propulsion Laboratory—A federally funded research and development laboratory and NASA field center in Pasadena, California (www.jpl.nasa.gov).
KNMI	Royal Netherlands Meteorological Institute—Dutch national weather service and the national research and information center for meteorology, climate, air quality, and seismology (www.knmi.nl/over-het-knmi/about).
LAS	Live Access Server (www.ferret.noaa.gov/LAS).
LIU	Linköpings Universitet's National Supercomputer Centre in Sweden—Houses an ESGF data node, test node, ESGF code sprint, user support, and Bi and Frost clusters (www.nsc.liu.se/).
LLNL	Lawrence Livermore National Laboratory—DOE laboratory that develops and applies world-class science and technology to enhance the nation's defense and address scientific issues of national importance (www.llnl.gov).
MongoDB	A cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents (en.wikipedia.org/wiki/MongoDB).
NASA	National Aeronautics and Space Administration—U.S. government agency responsible for the civilian space program as well as aeronautics and aerospace research (www.nasa.gov).
NCI	National Computational Infrastructure—Australia's high-performance supercomputer, cloud, and data repository (nci.org.au).
NetCDF	Network Common Data Form—Machine-independent, self-describing binary (www.unidata.ucar.edu/software/netcdf/).
NOAA	National Oceanic and Atmospheric Administration—Federal agency whose missions include understanding and predicting changes in climate, weather, oceans, and coasts and conserving and managing coastal and marine ecosystems (www.noaa.gov).
NoSQL	Non SQL (en.wikipedia.org/wiki/NoSQL).

Acronym	Definition
NSC/LiU	National Supercomputer Center at Linköping University
NSF	National Science Foundation—Federal agency that supports fundamental research and education in all the nonmedical fields of science and engineering (www.nsf.gov).
OAuth 2.0	An open standard for authorization, commonly used as a way for Internet users to log into third party websites (oauth.net/2/).
Obs4MIPs	Observations for Model Intercomparisons—Database used by the CMIP modeling community for comparing satellite observations with climate models (www.earthsystemcog.org/projects/obs4mips/aboutus/).
OpenID	An open standard and decentralized authentication protocol. (CoG uses an ESGF OpenID as its authentication mechanism.)
PID	Persistent identifier—A long-lasting reference to a digital object, a single file, or set of files (en.wikipedia.org/wiki/Persistent_identifier/).
REST	Representational state transfer (en.wikipedia.org/wiki/Representational_state_transfer)
RHEL	Red Hat Enterprise Linux—Commercial version of the Linux operating system (www.redhat.com/en/technologies/linux-platforms/enterprise-linux).
RPM	Red Hat Package Manager (en.wikipedia.org/wiki/RPM_Package_Manager)
SAML	Security Assertion Markup Language
SDLC	Software Development Life Cycle
Solr	Open-source enterprise search platform built on Lucene that powers the search and navigation features of many commercial-grade websites and applications (lucene.apache.org/solr/).
TDS	THREDDS Data Server (www.unidata.ucar.edu/software/thredds/current/tds/)
UI	User interface (en.wikipedia.org/wiki/User_interface)
UV-CDAT	Ultrascale Visualization–Climate Data Analysis Tools — Provides access to large-scale data analysis and visualization tools for the climate modelling and observational communities (uvcdat.llnl.gov).
VO	Virtual organization (en.wikipedia.org/wiki/Virtual_organization)
WIP	WGCM Infrastructure Panel—Serves as a counterpart to the CMIP panel and will enable modeling groups, through WGCM, to maintain some control over the technical requirements imposed by the increasingly burdensome MIPs (www.earthsystemcog.org/projects/wip/).
XC	Executive Committee (en.wikipedia.org/wiki/Executive_Committee)