



# CONFIGURATION MANAGEMENT PLAN

Version 2-91-P  
Document Control Number 1000-00000  
2010-10-05

Consortium for Ocean Leadership  
1201 New York Ave NW, 4<sup>th</sup> Floor, Washington DC 20005  
[www.OceanLeadership.org](http://www.OceanLeadership.org)

in Cooperation with

University of California, San Diego  
University of Washington  
Woods Hole Oceanographic Institution  
Oregon State University  
Scripps Institution of Oceanography

## Document Control Sheet

Version	Description
2-91-P	Public version

Note: This document has been edited to remove information that is considered confidential and/or sensitive to ongoing or future financial negotiations for OOI procurements. Information removed has been replaced by the insertion of "[redacted]".

## Table of Contents:

<b>Document Control Sheet.....</b>	<b>Error! Bookmark not defined.</b>
1 Configuration Management System Structure.....	1
1.1 Purpose .....	1
1.2 Scope.....	1
1.3 Configuration Process .....	1
1.3.1 Drawing Standard.....	1
1.3.2 Software Versioning and Configuration Process.....	1
1.3.3 Implementing Organization Documentation Assurance Process.....	1
1.3.4 Implementing Organization Configuration Management Process.....	2
1.4 Responsibility & Authority .....	2
1.5 Configuration Controlled Documents .....	2
1.6 File Formats and Applications .....	3
1.7 File Naming Conventions (Non-Controlled).....	4
1.7.1 Terms and Definitions .....	4
1.7.2 Conventions .....	4
1.7.3 Acronyms .....	4
1.8 File Numbering Conventions (Controlled).....	5
1.8.1 Series Number .....	5
1.8.2 Sequence Number.....	6
1.8.3 Control Assignment Validation Criteria .....	6
1.8.4 Controlled Files Outside of the DMS.....	6
1.9 Other Naming Conventions.....	6
1.9.1 Observatory - Instrument Name .....	6
1.10 Technical Data Baselines .....	8
2 Document and Information Types .....	9
2.1 Hardware and Software Documentation .....	9
2.2 Document (Drawing) List.....	9
2.3 Procurement Documentation .....	9
2.4 Source Control Document / Vendor Item Drawing .....	10
2.5 Specifications .....	10
2.6 Hardware Parts Lists and Bills of Material .....	10
2.7 Hardware Assembly Drawings.....	10
2.8 Hardware Assembly Procedures .....	10
2.9 Hardware Configuration Database .....	11
2.10 Software Configuration Database .....	11
2.11 Program Management Support Information.....	11
2.11.1 Program/Project Organization .....	12
2.11.2 Work Breakdown Structure .....	12
2.11.3 Schedules and Activities .....	12
3 Document Management.....	13
3.1 Document Style and Format .....	13
3.2 Document Change Control.....	13
3.3 Change Request Process and Numbering Convention .....	13
3.4 Document Terms / Information Fields .....	13
3.4.1 Controlled Document.....	13
3.4.2 Controlled Document Copy .....	13
3.4.3 Draft (preliminary) Documents.....	13
3.4.4 Candidate Draft (review ready preliminary) Documents .....	14
3.4.5 Released Documents .....	14
3.4.6 Public Documents.....	14
3.4.7 Approval .....	14
3.4.8 Author .....	14
3.4.9 Release Date .....	14
3.4.10 Version Number .....	14
3.4.11 Version Date .....	14

## Configuration Management Plan

---

3.5	Document Management Application, Alfresco.....	15
3.5.1	General Overview .....	15
3.5.2	OOI Configuration.....	15
3.5.3	Attributes in Microsoft Excel and Word Documents .....	18
4	Collaboration Applications.....	19
4.1	Overview.....	19
4.2	Confluence .....	19
4.2.1	OOI Configuration.....	19
4.3	JIRA .....	20
4.3.1	OOI Configuration.....	21
4.4	Subversion .....	22
4.4.1	OOI Configuration.....	22
4.5	DOORS .....	25
4.5.1	OOI Configuration.....	25
4.6	Program Management Portal (SAF) .....	27
5	Design Reviews .....	28
5.1	Preliminary Design Review (External).....	28
5.2	Final Design Review (External) .....	29
5.3	Internal Final Design Review ( <i>ifdr</i> ) (Technical / Engineering) .....	31
5.3.1	<i>ifdr</i> Overview .....	31
5.3.2	Conducting an Internal Final Design Review.....	31
5.3.3	Internal Final Design Review Criteria .....	32
5.3.4	Design Verification .....	33
6	Configuration Control Structure .....	34
6.1	Baselines .....	34
6.2	Engineering Change Classes .....	34
6.2.1	Class I Changes .....	34
6.2.2	Class II Changes .....	35
6.2.3	Deviations.....	35
6.2.4	Engineering Change Content .....	35
6.3	PMO / IO Change Control Boards.....	36
6.4	System Level Change Control Board .....	36
6.5	NSF Program Manager Review.....	38
7	Change Control Process .....	39
7.1	Requests for Engineering Changes .....	39
7.1.1	Assessing the Impact of Requested Changes .....	39
7.1.2	Change Control Documentation .....	39
7.1.3	Funding for Changes:.....	39
7.2	Change Control Board Operation .....	40
7.2.1	Records of Meetings.....	40
7.2.2	Timing of Meetings .....	40
7.3	Change Control Board Authority.....	40
7.3.1	PMO / IO CCBs.....	40
7.3.2	System Level Change Control Board.....	41
7.4	Review Authority .....	41
7.5	Consensus and Resolution .....	41
7.6	ECR Closeout .....	41
	Appendix A-1 Engineering Change Request Form.....	1
	Attachment B-1 OOI Program Terms and Definitions .....	1
	Attachment B-2 Site Names and Codes .....	1
	Appendix A-2 OOI Change Control Process Table .....	2

## 1 Configuration Management System Structure

### 1.1 Purpose

The Configuration Management Plan (CMP) establishes the activities, responsibilities, processes, and methods used to document and maintain the design of the Ocean Observatories Initiative (OOI) program and to manage changes to the design and scope of the system of systems.

### 1.2 Scope

The CMP is applicable to all OOI systems and subsystems hardware and software technical data, designs and software code and hardware developed or delivered as part of the OOI program. The OOI "program" includes all of the National Science Foundation (NSF) funded resources and materials under the American Recovery and Reinvestment Act (ARRA), Major Research Equipment and Facilities Construction (MREFC) and Research & Related Activities (R&RA) (R&RA applicability to be addressed as part of Operations & Maintenance (O&M) development) agreements with the Consortium for Ocean Leadership (Ocean Leadership) and sub-awardees. The OOI Program includes "projects" which are groups of tasks undertaken by a specific sub-awardee or resources within the larger program.

The CMP defines the roles, responsibilities, and authorities of the OOI team members in the configuration management of the OOI planning, requirements flow, design, development, and implementation phases.

### 1.3 Configuration Process

The Implementing Organization (IO) shall have the following standards and processes in place and follow the standards and processes in the performance of the contract:

- a) Drawing Standard / Software Coding Standards
- b) Software Versioning and Configuration Process
- c) IO Documentation Assurance Process
- d) IO Configuration Management Process

#### 1.3.1 Drawing Standard

The IO has the option of utilizing its own drawing standard if it is similar to or exceeds the American Society of Mechanical Engineers (ASME) Y14.100 (Engineering Drawing Practices) requirements, otherwise ASME Y14.100 shall be used as the drawing standard. The standard must provide engineering definition to assure the functional interchangeability of parts when procured from original or alternate sources.

#### 1.3.2 Software Versioning and Configuration Process

[Reserved]

#### 1.3.3 Implementing Organization Documentation Assurance Process

The IO shall have a process in place that assures that the Technical Data Package delivered has undergone a thorough and structured creation, check, and approval cycle.

### 1.3.4 Implementing Organization Configuration Management Process

The IO shall have a Configuration Management Process in place that describes the IO's configuration control processes. Each IO shall assign the duties of a Configuration Manager (CM) to one of the members of the IO project office. This process shall also include a structured Change Control Board (CCB) process compliant with OOI policies and procedures.

## 1.4 Responsibility & Authority

The responsibilities and authority of the members of the OOI project teams are defined by the cooperative agreement between the National Science Foundation (NSF) and the Consortium for Ocean Leadership (Ocean Leadership), and the subaward contracts between Ocean Leadership and the IOs. The roles of the members are outlined below.

NSF Program Director: The NSF Program Director is the program sponsor and has the sole authority to revise any program baselines. Program Baselines are defined as: the milestone point-in-time scope and "state" of the related programmatic metrics. Specifically, at the NSF level, these are the program technical scope, cost and schedule.

Program Director for Ocean Observing Activities: The Program Director for Ocean Observing Activities at Ocean Leadership is the principal investigator and program manager. He/she has the authority and responsibility to carry out the policies and activities described herein, and may delegate such authority as required. The OOI Program Director has primary responsibility for all issues that bridge the construction project and operations and maintenance of the system of systems

Project Manager: The OOI Project Manager has primary responsibility for all system of system and system level issues. The Project Manager will work with the OOI System Engineer and the systems Contracting Officer's Technical Representatives (COTRs) to implement the configuration management process as described herein across the collaboration to ensure that requirements and interfaces are fully defined, documented and maintained.

Configuration Manager: The Configuration Manager will maintain the document list, assure changes are completed correctly, and supervise distribution (posting) of new revisions and the recall of obsolete documents. The Configuration Manager reports to the Project Manager or delegate. The OOI Program Office will assign a Configuration Manager for the OOI, and each IO shall assign the duties of configuration management to a member of the IO project office.

## 1.5 Configuration Controlled Documents

Documents and records will be maintained as described in this plan, including the following key products:

- Science Prospectus and Traceability Matrix
- Science Requirements
- Engineering Requirements Documents
- Interface Agreement/Control Documents
- Dynamic Object Oriented Requirements System (DOORS) Database
- Engineering Design Documents (e.g., Final Network Design, drawings, specifications)
- Technical Data Packages (TDP)
- Network Design (documents and drawings)
- Program Plans and Policies (e.g., Earned Value Management System Guide, System Engineering Management Plan, Risk Management Plan, Cost Estimating Plan)
- Project Execution Plan (PEP)
- Work Breakdown Structure (WBS) and related support documents
- Integrated Master Schedule (IMS)
- Cost Book Database
- Program Management Baseline Documents (including test plans)
- Logistic Support Documents

## Configuration Management Plan

A Technical Data Package (TDP) is a technical description of an item adequate for supporting an acquisition strategy, development, manufacturing development, production, engineering, and logistics throughout the item's lifecycle. The technical description defines the required design configuration and procedures required to ensure adequacy of item performance.

A TDP is comprised of a variety of data that will define the item/system. The categories of data in a TDP include, but are not limited to:

- Product Definition Data
- Engineering Drawings
- Associated Lists
- Specifications
- Standards
- Performance Requirements
- Quality Assurance Provisions
- Reliability Data
- Packaging Details
- Modeling Data

"Product definition data" denotes the totality of data elements required to completely define a product. Product definition data includes geometry, topology, relationships, tolerances, attributes, and features necessary to completely define a component part or an assembly of parts for the purpose of design, analysis, manufacture, test, inspection and maintenance.

### 1.6 File Formats and Applications

The following shall be used to create and exchange information between IOs and the OOI Program Office and in preparation of any technical data package.

Type	File Format /Application
Requirements Database	DOORS
Document Authoring	Word 2003 or compatible
Document Distribution	Portable Document Format (PDF)
Engineering Drawings	CAD interchange with 2d AutoCAD version 12 (aka 2008) or AutoCAD LT for 2d. Note: 3D models not necessary for exchange.
General Graphics	Note: vector form
Network or Diagram Graphics	Enterprise Architect (Visio Omni / Graffle optional)
Scheduling and Project Management	Project Server 2007
Technical Data Distribution	Excel 2003 or compatible

Collaborative Application	Description
Confluence	Web collaborative environment / Wiki
Alfresco	Document Management System
JIRA	Action Item / Issue Tracking
GIT/Subversion	Software Code CM
DOORS and Software Architect (CI)	Requirements Tracking / Tracing
Program Management Portal (SAF)	Change Control, Risk Management, and other tools

Note: See [Collaborative Applications Section 4](#) for details.

## 1.7 File Naming Conventions (Non-Controlled)

The following naming convention shall be used to exchange information between IOs and the OOI Program Office and in preparation of any technical data package of non-controlled documents/drawings.

This naming convention is also used for documents prior to their submittal to the DMS or prior to the issuance of a control number.

File Name: Short Descriptive Name, Scope Level, YYYY-MM-DD ver #-##.ext

**Short Descriptive Name:** relative to the subject or name of the document

**Scope Level:** identifying the scope of the document, from a sensor, platform, location, observatory, or top level, (e.g.: OOI, Irminger, RSN)

**Date:** in four digit year, dash, two digit month, dash, two digit day, representing the completion date or release date of the document, incremented as changes are made

**Version:** "ver" indicating the major release number and the minor working level, number dash number. A required element if a released version of a configuration controlled document. Used for version tracking in non-configuration controlled documents. (e.g.: ver\_2-05) Use a leading zero for "dash" 1 through "dash" 9 (e.g. -01, -02,...-09)

Note: Underscores are used as separators, no spaces permitted in file names. Dashes are to be used only for the date string and version number string. For example:

- *CMP\_OOI\_2008-01-21\_ver\_2-05.doc [Configuration Management Plan]*
- *Preliminary\_Design\_Document\_CI\_2007-07-24\_ver\_3-01.doc*

### 1.7.1 Terms and Definitions

The OOI Program terms and definitions list is maintained by the OOI SE and is part of the CMP by attachment. The Program terms and definitions are maintained in the DOORS database as a Reference Module and will be updated throughout the lifecycle of the program.

### 1.7.2 Conventions

The OOI Program and IO Systems will develop multiple naming and alpha-numeric identification and tracking conventions. The conventions will be maintained by the IO SE and OOI SE and are part of the CMP by attachment. "Attachment B-1, OOI Program Naming and Identification Conventions" will be updated throughout the lifecycle of the program.

### 1.7.3 Acronyms

The following acronyms are acceptable for file naming use within OOI.

Title	Acronym
Basis of Estimate (by WBS Element)	BOE
Cost Estimating Plan	CEP
Configuration Management Plan	CMP
Earned Value Management Plan	EVM_Plan
Environmental, Health and Safety Plan	EHSP
Education and Public Engagement	EPE
Final Network Design	FND
Integrated Master Schedule	IMS
Operations and Maintenance Plan	OM_Plan
Project Execution Plan	PEP
Quality Assurance and Quality Control Plan	QA_QC_Plan
Systems Engineering Management Plan	SEMP
Work Breakdown Structure	WBS



## 1.8 File Numbering Conventions (Controlled)

All controlled and released level documents shall have a unique OOI / IO document number assigned and added as a prefix to the document file name.

The document number shall be comprised of a four digit Series Number and a five digit Sequence Number. The document number shall follow the following format (#### - #####).

The OOI and IO level configuration managers shall assign and track the requests and issuance of the document numbers. The OOI Configuration Manager has primary responsibility for all system level configuration issues. The OOI Configuration Manager shall disperse and assign file series and sequence numbers from the system level. The IOs shall implement a system-level umbrella file number/naming convention to append to the OOI / IO series, specific to the requirements of the IO configuration management process for controlled documents and drawings. The IO Configuration Manager/SE shall assign from within the series identified for each IO.

### 1.8.1 Series Number

The file Series Numbering convention shall follow the integrated Work Breakdown Structure (WBS), with Series Numbers corresponding to the second level of the WBS.

Series Number	Assignment
1000	OOI/Program Level
2000	Cyberinfrastructure
3000	Coastal Global
4000	Regional
5000	Operations and Maintenance
6000	(reserved)
7000	Education and Public Engagement
8000	(reserved)
<b>OOI Program Level</b>	<b>Documents</b>
1000 – 1099	OOI Programmatic Documents
1100 – 1199	OOI Systems Engineering Documents
1200 – 1299	OOI Installation Documents
1300 – 1309	Engineering Change Requests
1330 – 1339	OOI Specifications
	<b>Drawings</b>
1500 – 1599	OOI Programmatic Drawings
1600 – 1699	OOI Systems Engineering Drawings
1700 – 1799	OOI Installation Drawings
<i>2<sup>nd</sup> Level WBS = n</i> <i>n = 2, 3, 4, or 7</i>	
<b>IO System / Subsystem</b>	<b>Documents</b>
n000 - n099	IO Programmatic Documents
n100 - n199	IO Systems Engineering Documents
n200 - n299	IO Installation Documents
n300 - n499	IO Subsystem Documents
	<b>Drawings</b>
n500 - n599	IO Programmatic Drawings

## Configuration Management Plan

n600 - n699	IO Systems Engineering Drawings
n700 - n799	IO Installation Drawings
n800 - n999	IO Subsystem Drawings
<i>Note: CI = 2, CG = 3, RSN = 4, EPE = 7</i>	
<i>Example:</i>	
4800 - 4999	RSN Subsystem Drawings

### 1.8.2 Sequence Number

The file Sequence Numbering convention shall be determined by the OOI Configuration Manager and the IO Configuration Managers for the Series Numbers under their respective control. Different Sequence Number conventions may be used for different types of documents.

### 1.8.3 Control Assignment Validation Criteria

The Document Management System (DMS) "Alfresco" assigns a sequential "revision" number to the document when it is loaded and each time the document is changed. This "revision" number is not recorded in or on the file or document.

The CM must ensure that the "Version" number assigned and tracked for OOI controlled files is included on the cover and in the document control sheet of each file, prior to assigning a control number. The "Version" number must be recorded in the DMS in the notes section along with the date the document was ratified by CCB.

### 1.8.4 Controlled Files Outside of the DMS

Checking-out, downloading, exporting or off-line editing of a controlled document outside of the DMS requires that the file name be appended with the versioning convention described in Section 1.7. The OOI Version Number should be appended to the file name. In all cases the DMS document is the document of record, and the Configuration Manager will manage all updates from external sources.

## 1.9 Other Naming Conventions

The OOI Program and IOs will develop multiple naming and alpha-numeric identification and tracking conventions. The conventions will be maintained by the IO SE and OOI SE and are part of the CMP.

### 1.9.1 Observatory - Instrument Name

The Configuration Item naming convention below is used to uniquely identify components in the system at the instrument level. This convention will be used to identify items such as Control Centers, Moorings, Buoys, Nodes, Cables, Instruments and Sensors in a consistent manner across all documentation and databases. Accurate and consistent identification of components is required for effective fault management, auto-discovery, performance reporting, sensor data storage, etc. The identification of the instrument and accurate correlation between an item identifier and all databases that contain supporting information is supported by the Instrument ID naming convention.

An OOI Node is defined as an entity that aggregates ports and/or distributes power, time, and communications. An OOI Instrument is defined as a collection of transducers, data samplers, data loggers and transducer controllers, with a known internal structure/geometry.

## Configuration Management Plan

### 1.9.1.1 Instrument ID Field (total characters: 27)

	<i>Observatory / Array</i>	<i>Site Prefix</i>	<i>Site Suffix</i>	<i>Format "separator" B</i>	<i>Node Type</i>	<i>Node Site Sequence</i>	<i>Format "separator" B</i>	<i>Port Num</i>	<i>Format "separator" C</i>	<i>Instrument Class</i>	<i>Instrument Series (discriminator)</i>	<i>Instrument Sequence Number</i>
<b>Char Type</b>	Alpha	Numeric	Alpha-Numeric	Symbol	Alpha	Alpha-Numeric	Symbol	Numeric	Symbol	Alpha-Numeric	Alpha	Numeric
	AA	##	CCCC	-	AA	CCC	-	##	-	CCCCC	A	###
<b>Number of Characters</b>	2	2	4	1	2	3	1	2	1	5	1	3
<b>Mandatory Character</b>		"0#" leading zero for single digit		"," dash			"," dash		"," dash			"00#" leading zeros for single digit

### 1.9.1.2 Example Instrument ID

Example: *RS02SUM2-MJ01B-03-PRESTA102*

*RS      02      SUM2      -      MJ      01B      03      -      PREST      A      102*

RSN, site number 2, Hydrate Ridge/Summit Southern 2, Medium Power Junction Box number 01B, port number 3, Pressure instrument, series A, instrument number 102

### 1.9.1.3 Field Names and Descriptions

**Observatory / Array** 2 character alpha code that denotes the OOI Sub-System of the item.

**Site Prefix** 2 character numeric code that denotes the Site Prefix of the item (leading zero).

**Site Suffix** 4 character alphanumeric code that provides a representation of the sub-system specific site name in abbreviated alphanumeric characters

**Node Type** unique 2 character alpha code providing the type of node name in abbreviated alpha characters. spaces filled with underscore.  
Note: reserved characters "CA" for cable assembly and id the nodes on each end of the cable.

**Node Site Sequence** 3 digit mixed alpha-numeric code sequentially numbering the nodes at a site.

**Port Number** 2 digit numeric code that sequentially identifies the ports on a node.

**Instrument Class** 5 character mixed alpha-numeric code that represents an instrument class.

**Instrument Series** single alpha code that denotes a specific model or series of instruments, a unique identifier in combination with the instrument class.

**Instrument Sequence Number** 3 digit numeric code that sequentially identifies the instruments on a port or node.

### 1.9.1.4 Example System – Site Names and Codes

See 1000-00001 B2 Table.

### 1.10 Technical Data Baselines

The Technical Data Package (TDP) is the complete documented set of technical information, data, software, and drawings that completely describe the system to the level necessary to manufacture, implement, test, and maintain the system. The IOs shall prepare and provide the following baselines of the TDP:

- a) **NSB Approved Baseline:** This baseline shall consist of the information included in the Final Design Review (November 2008) plus any modifications or updates resulting from the FDR. It defines the complete system (cost, schedule, technical) in sufficient detail to begin the construction phase of the program.
- b) ***ifdr* Baseline:** The TDP must define the complete system from contract award leading up to the internal final design review (*ifdr*).
- c) **Production Baseline:** The establishment of the production baseline shall be determined after the design has been completed, all drawings and their associated parts lists and wire lists have been developed and or updated as per the *ifdr*, but prior to the manufacture and deployment of the system.
- d) **Final Production Baseline:** The Final TDP shall consist of the Production Baseline TDP including all changes via the ECR process and corrections of discrepancies discovered during the production and deployment phase of the program. This TDP shall represent the delivered production system.
- e) **As-Commissioned (as-built) Baseline:** The As-Commissioned TDP shall consist of the Final Production Baseline TDP including all changes via the ECR process and corrections of discrepancies discovered during the deployment, waivers and deviations, testing and commissioning phase of the program. This TDP shall represent the installed-tested system, including any deployment location specific information.

## 2 Document and Information Types

### 2.1 Hardware and Software Documentation

Documentation generated for the purposes of defining the design, manufacture, installation and service of OOI hardware shall be created, approved and changed per the processes noted in this document and the *OOI System Engineering Management Plan 1100-00000*. The word "drawing" can be used as a synonym for document.

Hardware documentation can include:

- Requirements
- Specifications
- Engineering Analysis and Decision Reports
- Assembly procedures
- Assembly drawings and schematics
- Bill of Materials and parts lists
- Test procedures
- Routing sheets, travelers, test result forms
- Installation and commissioning procedures and forms

Documentation generated for the purposes of describing the functionality, interfaces and use of software shall be created, approved and changed per the processes noted in this document and the *OOI System Engineering Management Plan 1100-00000*.

Software documentation can include (in addition to items listed above):

- Use cases
- Interface and Usability Design Documentation (including wireframes)
- Source Code, for all iterations
- Build environment lists and procedures
- Software Requirements Documentation
- Software Design and Architecture Diagrams and Documentation (including any UML-related material available)
- System Logical and Physical Architecture Diagrams
- Data diagrams and documentation (data schemas, data flow diagrams, entity relationship diagrams, etc.)
- Administrator and User Manuals
- Maintenance Processes and Procedures

### 2.2 Document (Drawing) List

Each Configuration Manager will maintain a document list, and will be the source of assigning new numbers and titles.

### 2.3 Procurement Documentation

Procurement documents are those documents utilized in bid and proposal activities, which include buyer's invitation for bid, invitation for negotiations, request for information, request for quotation, request for proposal and seller's responses, specifications, statements of work, and requirements. These documents are created by the engineering development effort and become part of a documented baseline with sufficient information for the purchase or manufacture of needed items. The complexity of the documentation may range from a vendor's part number and corresponding specification sheet to a detailed specification and performance criteria to achieve reasonable confidence that a competitive selection will yield the expected technical/programmatic result.

### 2.4 Source Control Document / Vendor Item Drawing

Source Control Documents (SCD) are used to describe qualification and acceptance requirements for items procurable from a specialized or pre-qualified segment of industry. Vendor Item Drawings are used when the manufacturer's literature does not completely define the required configuration or behavior of the item specified. These documents supplement the existing standard item documentation by defining any missing or unique design attributes. Source control requires selection from "approved sources" only.

### 2.5 Specifications

A specification is referenced by a contract or procurement document. It provides the necessary details about the specific requirements. Specifications may be written by government agencies, standards organizations (ASTM (American Society for Testing), International Organization for Standardization (ISO), European Committee for Standardization (CEN), etc), trade associations, corporations, and others.

Specifications are used when the needed items must be defined "from scratch" to support vendor selection. The specification establishes the complete set of detailed characteristics of the item to be delivered, and is typically much more comprehensive than an SCD.

### 2.6 Hardware Parts Lists and Bills of Material

Each hardware assembly drawing shall be accompanied with a Parts List that describes all of the materials and components that are required to build an assembly or subassembly, including quantity, part/model # (as appropriate) and part version (as appropriate). The Parts List may be a stand-alone document or included as part of the assembly drawing.

Each system shall be accompanied by a Bill of Material (BOM) generated from the manufacturing process and intended for use as a system parts lists and Modular BOM. The BOM shall be hierarchical in nature with the top level representing the sub-systems and provide incorporation or a link to the Parts Lists. The BOM must include, as available: part/model number, quantity, version, supplier and estimate price, and may include actual pricing and receipt history. The BOM is not intended to be an asset tracking system, but will support programmatic data requirements.

### 2.7 Hardware Assembly Drawings

Assembly drawings shall be developed for each module, top-level hardware assembly or subassembly that is to be manufactured or delivered for OOI. Assembly drawings shall clearly state the version of the assembly or subassembly they represent. The document number and revision of the assembly drawing may be used as the part number and revision of the assembly or subassembly it represents. An extension may be added to the base document number to indicate that multiple configurations of the item exist. Each IO shall submit a proposed list of drawings based on WBS top-level breakdowns to the OOI SE for review and approval prior to the development of each baseline TDP.

### 2.8 Hardware Assembly Procedures

Each hardware assembly drawing shall be accompanied with an Assembly Procedure that describes all of steps and tools required to build an assembly or subassembly. The assembly procedure may be a stand-alone document or included as part of the assembly drawing.

## 2.9 Hardware Configuration Database

The OOI Program Office shall establish a database that maintains the "as built" configuration of OOI devices / hardware including, software, and firmware installed. Records shall be maintained by each IO relative to hardware location in the development and implementation phases of the project, culminating with the final installation location within the OOI. This location and installation information will be added to the hardware configuration database. Each IO shall support the operation of the database by providing data directly or indirectly to the database application or the OOI Program Office. Although this information can co-existent with asset tracking information, it is intended to be incorporated into the TDP as high density as-built/as-installed technical location information, and not simply address or geographic coordinate system.

The database shall contain the following information:

- Part number, serial number, and revision of the final hardware assembly
- Part number, serial number, and revision of all hardware subassemblies integrated into a hardware assembly
- Part number and serial number of major components integrated into a hardware assembly
- Firmware
- Software version and configuration
- Installed firmware version installed
- Date of manufacture
- Installation Details

## 2.10 Software Configuration Database

The OOI Program Office shall establish a database that maintains the "release" configuration of OOI software and firmware. Each IO must support the operation of the database by providing data directly or indirectly to the database application or the OOI Program Office.

The database shall contain the following information:

- Version number of the software build/release, including any integrated applications, modules or libraries
- Software installation instructions
- Software version description
- Build instructions including build environment

## 2.11 Program Management Support Information

The OOI Program Office shall establish a set of information including but not limited to the following:

- Processes
- Procedures
- Training materials
- Manuals
- Assembly and test records
- Acceptance
- Plans
- Budgets
- Costbooks and Basis of Estimates
- Risk Registers
- IO Schedules and Integrated Master Schedule

The documentation is generated for the purposes of supporting the management of the OOI including management of the work of defining the design, manufacture, installation and service of OOI. The information shall be created, approved and changed per the processes noted in this document and the *OOI Systems Engineering Management Plan 1100-00000*.

The program management support information will be developed and archived in software applications and databases. Each IO shall provide data directly or through the OOI Program Office in support of the operation of the databases. Program management activities must be monitored and reported on at periodic intervals as defined by the OOI Program Office.

### 2.11.1 Program/Project Organization

The *OOI Project Execution Plan 1001-00000* establishes the cooperative partners, roles, and responsibilities. Cooperative agreements place further definition on expected System Engineering tasking, including definition of deliverables each institution will be responsible to provide.

### 2.11.2 Work Breakdown Structure

The OOI WBS and WBS Dictionary establish the individual elements of work that will be managed. The WBS is a controlled and maintained document.

### 2.11.3 Schedules and Activities

The OOI Project Management Control System (PMCS) is a controlled and maintained set of cost and schedule baselines for each WBS element. The PMCS high-level milestones form the basis for scheduling all System Engineering and Integration/Test activities. Task status is reported monthly, and any significant variance from cost or schedule baseline is analyzed and reported. The *OOI Earned Value Management Plan 1005-00000* details the process outlined in this paragraph.



### 3 Document Management

The Configuration Manager is responsible for tracking and maintenance of the document list (1199-00000 Master Document Tracking) with version numbers and dates. Authors of preliminary documents are responsible for updating the date on the document list and document within a day of the change and must provide an electronic file and .PDF file for the electronic repository upon issuance.

#### 3.1 Document Style and Format

The Configuration Management Plan is the template for document style and format. All documents must contain a title page, change control tracking page and a table of contents. Each page is to be numbered and include the document name, revision level and page numbers.

#### 3.2 Document Change Control

Preliminary (draft and candidate draft) documents may be changed by the originator and distributed as needed. The originator must advise the Configuration Manager of the date when the revision occurs. Controlled documents may be changed only via an Engineering Change Request (ECR) or similar change system. Supporting documents may be revised through the IO change control system.

#### 3.3 Change Request Process and Numbering Convention

An Engineering Change Request shall be completed by the requestor of the change or a member of the Change Control Board. The person requesting the change shall do so via the Change Control tool in the Program Management Portal (SAF). The Change Control tool automatically assigns an appropriate document number to the ECR. ECR numbering shall adhere to the following format:

130x-nnnnn

where:

x is the PMO/IO 2<sup>nd</sup> level WBS series number (e.g., System Level 0, PMO 1, CI 2, CG 3, RSN 4)

nnnnn is the unique ECR number serially assigned by the automated system.

Note: The Engineering Change Request process is managed by the OOI CCB Chairperson or a designated member or CMA with the delegated authority.

#### 3.4 Document Terms / Information Fields

##### 3.4.1 Controlled Document

Any process or program document that is formally approved, maintained, distributed, and revised via a document control system.

##### 3.4.2 Controlled Document Copy

A copy of a controlled document that has been formally issued by the Configuration Manager and whose distribution is maintained by the Configuration Manager.

##### 3.4.3 Draft (preliminary) Documents

Preliminary documents have a title and version number, include the word "Draft," an identified originator, and the date of the current preliminary version clearly shown both on the document and in the document list.

### 3.4.4 Candidate Draft (review ready preliminary) Documents

Preliminary documents that are mature and ready for review have a title and number, include the words "Candidate Draft," an identified originator, and the date of the current preliminary version clearly shown both on the document and in the document list. Candidate Draft versions of document deliverables are structurally complete, contain mature information, have been internally reviewed and edited, and are ready for a release quality review. Comments and corrections from a candidate reviews are incorporated into the final/release versions.

### 3.4.5 Released Documents

When a document has been reviewed and approved by the appropriate level program/project manager or Board, it is considered released and is subject to formal distribution and revision control. Released documents will NOT have qualified titles such as "draft," "candidate draft," or "preliminary."

### 3.4.6 Public Documents

When a document has been reviewed and redacted for any information that may interfere with fair procurement and bidding, it must be approved by the appropriate level program/project manager prior to release for unrestricted public access. The public document is subject to formal distribution and revision control. Public documents will have a designation of "Public" in the version text.

### 3.4.7 Approval

The process by which a document is reviewed and determined to be suitable for use by knowledgeable staff and management. Levels of approval exist at the IO and OOI Program Offices, and are by default the Project Manager at the IO level and the Project Director at the OOI level.

### 3.4.8 Author

The person or entity that creates the document; typically used for text documents. Interchangeable with Originator.

### 3.4.9 Release Date

This is the date of the original release of the document defined as the date of the document was formally approved. Preliminary documents have no release date. This date is recorded internal to the document, and is not represented by the file naming convention date.

### 3.4.10 Version Number

The latest version of a document is defined by the version date in the document file name and a numeric version number. It indicates the major release number and the minor working level, number dash number. A required element if a released version of a configuration controlled document. Used for version tracking in non-configuration controlled documents. (e.g., ver\_2-05)

### 3.4.11 Version Date

The date of the current preliminary (before release) version. It must be updated each time the document is changed. At the time of release, this date remains the date the document was last updated prior to release. The date is removed from the file name upon entry into the DMS or control. (YYYY-MM-DD)

### 3.5 Document Management Application, Alfresco

#### 3.5.1 General Overview

The Alfresco Document Management software is the basis for the OOI Document Management System (DMS) portion of the Collaboration Tools. Document Management software enables a unified, extendible digital solution of how documents are created, stored, filed, retrieved, secured, recovered, retained, archived, distributed and authenticated; all of which span near-unlimited locations (only limited by connectivity).

The central repository aspect of the OOI DMS will efficiently store libraries of documentation, as well as past revisions and versions. This central repository not only allows for disparate groups and individuals to gain access to the proper documentation, but also provides an easy to use, single source of access to all of the documentation they require. It also enables various policies that documents within the repository are subject to, including but not limited to organizational security, disaster recovery, retention, and archive policies.

Version Controls with Document Management software give strong support to the change process within tasks and project, which the OOI DMS will automatically inherit from the Alfresco base. This allows for previous versions of documents to be archived, thereby enabling better program oversight as documentation can be monitored within as iterative states.

Document Management software also enables is a true sense of workflow associated to each critical document within a project and/or organization. This allows documents to be controlled in a fashion where creation, editing, and deletion is tracked, monitored and managed. Workflow is defined more narrowly as the automated movement of documents or items through a sequence of actions or tasks that are related to a business process. Workflows are used to consistently manage common business processes within an organization by enabling the organization to attach business logic to documents or items in a DMS or library. Business logic is essentially a set of instructions that specifies and controls the actions that happen to a document or item.

Workflows within Document Management streamlines the cost and time required to coordinate common business processes, such as project approval or document review, by managing and tracking the human tasks involved with these processes. Workflows shall be developed for the OOI program, and incorporated into the OOI DMS. Additional benefits include increased oversight on documents, allowing serial or parallel approval and editing processes for individual or groups of documents. Each will require different levels of input, editing and screening, and each may need different people or organizations to approve.

Alfresco uses roles to determine what a user can and cannot do in a space. These roles are associated with permissions, are set for each specific space, while Administrators have all rights in all spaces. This way, only those with the proper authority to create, edit, or delete content and information will be able to do so. Individual work spaces will not exist, thereby eliminating additional storage and management requirements.

#### 3.5.2 OOI Configuration

The OOI Alfresco Document Management System (DMS) is hosted and maintained by the network administrators at the OOI's IT group located at the Washington, D.C. OOI facility.

### 3.5.2.1 Roles and Permissions

The following table shows the modified roles and permissions for the OOI Alfresco DMS:

All permissions apply to the invited space	Collaborator	Contributor	Editor	Consumer (4)
See invited space	X	X		X
View content	X	X		X
Copy content	X	X		X
Preview content in template	X	X		X
View content properties	X	X		X
Check in content to invited space	X			
Checkout content to different space.	X			
Update/edit content created by other users	X			
Update properties for content created by other users	X			
Edit existing discussions	X			
Create/add new content (1)	X	X	X	
Cut/delete content created by other users				
Create child spaces in the invited space	X	X		
View content rules	X	X		
Checkout content to same space.	X			
Contribute to existing discussions	X	X		
Invite others				
Start new discussion topic	X	X		
Delete content created by other users				
Same access rights as content owner				
Take ownership of content				
Create space rules				

**Note:** (1) A creator automatically owns his/her own created content.

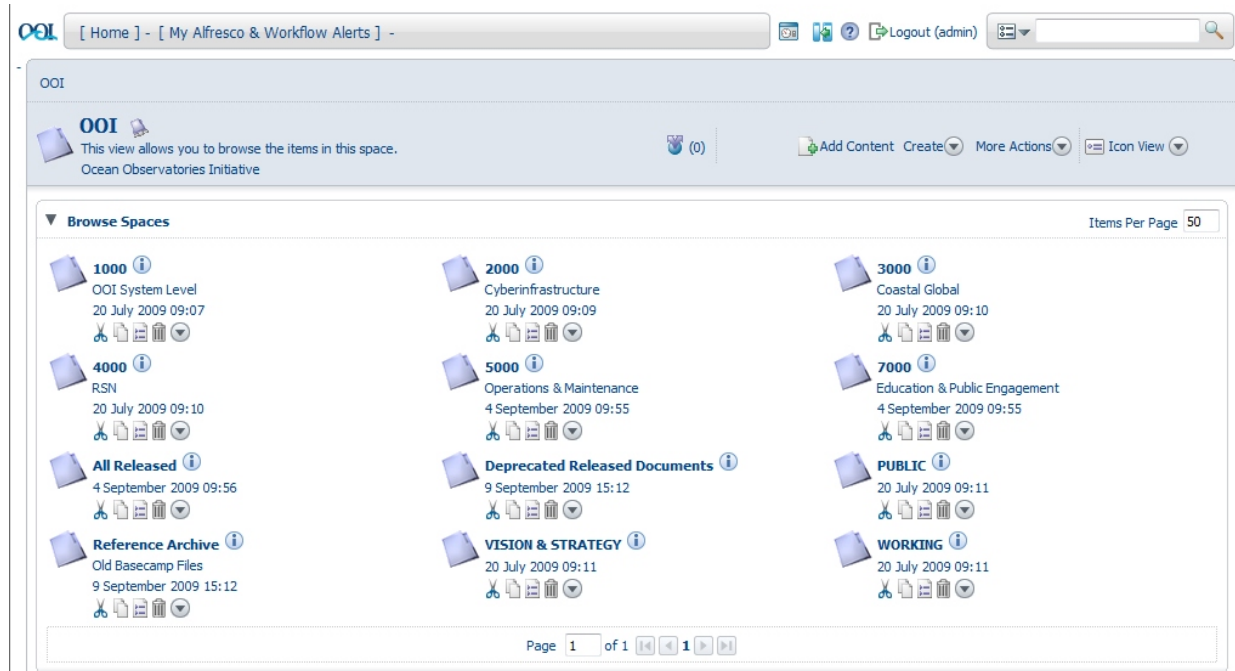
(2) The Editor role has been modified for use when someone needs to have permissions to submit to a folder in which they have no consumer rights.

(3) The group "Configuration Manager" has been associated with role of "Coordinator."

(4) The group "Public" has been associated with the role of "Consumer."

### 3.5.2.2 Directory Structure

When entering the OOI DMS, users are presented with the top level OOI portal view of the system. Here is where the File System layout of the OOI DMS will be found, which includes these base directories:



Within each directory, users will find all managed documentation associated with that directory. Each document will display attributes such as its name, the date/time stamp of its latest revision, size of file, as well as display icons which represent what functions are available to the user based on their roles and permissions (e.g., Edit Offline, Download, View Details, Copy, etc.).

### 3.5.2.3 Version Control

All documents entered into the OOI DMS are subject to version controlling. Thus, documents within the OOI DMS will support change process by being associated with their prior versions, and having fully preserved previous versions of the documents also available to users and groups with the proper privileges.

### 3.5.2.4 Workflows

It should also be noted that all documents entered into the OOI DMS are subject to workflow handling. Thus, documents within the DMS will be attributed to various stages and will follow change control and process linked to the defined Directory Structure. Documents may be subject to individual or group review, prior to its availability or accessibility.

### 3.5.3 Attributes in Microsoft Excel and Word Documents

The document properties are the same in both Excel and Word. The DMS uses the information in the document properties to populate the Alfresco metadata when the document is uploaded. All Excel and Word document files must have the following fields populated with correct information:

1. **Title:** Enter the title of the document. This is the title that appears on the cover page of the document. It uses spaces, does not use underscores, does not include a date and does not include a version number.
2. **Subject:** Enter an optional expanded description for the document.
3. **Author:** Enter the full last name, comma, space, and first initial of the person that is the author of the document.
4. **Manager:** Enter the full last name, comma, space, and first initial of the person that is responsible for configuration management of the document.
5. **Company:** Enter the acronym for the implementing organization (i.e., OOI, RSN, CG, or CI)
6. **Date Completed:** Enter the release date for the particular document version.
7. **Document Number:** Enter the nine digit document number in the "nnnn-nnnnn" format following the guidelines of [Section 1.8 of this document](#).
8. **Status:** Enter the current status of the document from the standard three defined by the CMP. Enter either Draft (preliminary, working version), Candidate Draft (Edited and mature version, ready for distributed review), or Released (reviewed, approved, and distributed for use).
9. **Version:** A "custom" document property for version (OOIVersion) is available on the last tab. A "revision" field is automatically assigned by Alfresco, starting the first time the file is added to the database, but is not user definable. All Versioning information must be maintained in the document properties, titles, headers, footers, as well as the document control sheet. The template uses document properties and update tools to manage these items.

## **4 Collaboration Applications**

### **4.1 Overview**

Five (5) Collaboration Tools are used to provide the OOI Program Office and all IOs the ability to digitally centralize the creation, control and distribution of information and assets. This includes, but is not limited to, information management, document management, issues management, software versioning and control, as well as requirements management, all of which fall under the purview of the Collaboration Tools. The DMS is detailed in Section 3.

### **4.2 Confluence**

Atlassian's Confluence Enterprise is the Wiki to be used by the OOI. The OOI Wiki shall be used not only as social software, allowing for collaborative communication throughout various associated communities, but also as an information portal where a vast amount of disparate information can be collected, jointly authored, collated, managed and distributed from a single source. Acting as a singular source of dynamic information creation, editing, posting and storage, Wikis create a community of users and information seekers, allowing for a simplified view of important information, while also enabling a feedback mechanism which not only allows for additional input, but also foster continual community interest, communication, and growth.

The OOI Wiki enables enterprise information management, allowing contributors from across organizations and geographic locations to pool and share all of their information together using a simple but effective interface, boosting general information sharing within an organization and out. As a main collaboration tool, the enterprise-level OOI Wiki incorporates functionality necessary for multiple users and groups, with access and security considerations. Different levels of permissions, either based on user or group, can be fine-grained to decide who is able to view, create, edit and comment on particularly information.

Built upon Atlassian's Confluence Enterprise Wiki, the OOI Wiki will also enable additional corporate capabilities. The OOI Wiki is a web-based platform which runs on just about every application server and database commonly available. The enterprise functionality includes the flexibility and scalability of the software to grow with an organization, while also allowing for large amounts of functionality and feature growth, as well as customization.

Confluence is organized by "spaces" which section off areas based on user requirements (e.g., by topic, by interest, temporally). Spaces can contain articles, documents, comments, lists, calendars, and any list of digital assets (including Excel files, pictures, etc.) The Confluence platform also provides full web service interfaces (SOAP and XML-RPC) for current and future applications or scripts to remotely update content, manage users, or administer individual spaces.

#### **4.2.1 OOI Configuration**

The OOI Program Office shall use the Confluence Enterprise Wiki instance in a standard corporate, best practice use of a wiki environment. The OOI Wiki is hosted and maintained by the network administrators at OOI's IT group at the University of California, San Diego.

"Spaces" have been defined within the OOI Wiki for the sake of hierarchically arranging information based on Program Office or specific topic areas. The naming convention for "Spaces" is designated by Group or IO, and then topic area. For example, one "space" has been designated as the "OOI Engineering Space", which denotes that this OOI space belongs to any topic area related to systems engineering efforts. Once you have entered a space, you will not only see the creator of the space, but also the last person who has edited the space, or articles within.

## Configuration Management Plan

The main “Spaces” defined in the OOI Wiki are:

Space	Description
OOI Vision and Strategy Space	Vision and strategy space for furthering the community scientific and education goals of the OOI.
OOI Program Management Space	Program Level Collaboration Area
OOI Engineering Space	Systems Engineering and Integration Collaboration Space
OOI Public Space	Public information exchange area and repository for public document versions.
OOI Collaborative Web Presence	Collaborative workspace for the development and deployment of an OOI Web Presence

The Configuration Manager is responsible for facilitation of the “Spaces” on the OOI Wiki, as well as the content within each of the “Spaces”. Authors of preliminary documents, postings, or digital asset within the OOI Wiki are responsible for updating the information on the OOI Wiki. Notifications are provided to the appropriate groups and users through both RSS feeds, as well as automated e-mail notification, providing a brief synopsis upon issuance. Authors are also responsible for monitoring any comments received on their posting, and bringing any major contention to the Configuration Manager’s attention.

Before release, the controls on preliminary posting are minimal and intended to facilitate the communication of early information, numerous changes in a short period of time, and community feedback.

The OOI program office shall adhere to security policies and privileges which will grant certain groups and users additional access and management capabilities. These are intended to give certain groups and users privileges of creating, approving, editing, and/or posting information, while at the same time limited functionality and/or capabilities of others.

### 4.3 JIRA

As part of the OOI Collaboration Tool set, JIRA shall be the platform of choice for the OOI Issues Tracking and Management System. Issues Tracking and Management Systems are used to prioritize, assign, track, report and audit 'issues,' related to tasks, projects or any number of engagements, including software and hardware bugs, as well as help desk tickets and change requests.

Issues tracking and management systems also implement a workflow foundation, which in turn regulate entered issues into “states”. Issues move from state to state, such as “submitted”, “open” “evaluation”, “working”, “in testing”, “closed”, and moves can be bidirectional. Moves in state can be initiated by those with proper authority, or through well-defined inputs and outputs, can be automated via business logic.



Issues to be processed and worked through the OOI Issues Tracking and Management System basically follow these steps through the process:

1. An issue is submitted by a scientist, engineer, or test engineer, via a web browser or e-mail submittal.
2. The issue management system logs the issue and relocates the issue to a predefined representative's inbox.
3. The representative evaluates the issue and assigns it to an appropriate team member.
4. Work is done on the issue, documented in the system, and closed.
5. The originator is notified that the issue has been resolved.

The OOI Issues Tracking and Management (ITM) System will also allow for series of reporting and auditing, which will be able to provide basic statistical analysis, but will also be able to give true trend analysis, enabling business intelligence. Some reporting includes:

- unresolved high-priority issues
- number of issues per customer
- average resolution time
- estimated time vs. actual time taken
- number of issues created per day/week/month/year
- a breadth of other customizable reports

### 4.3.1 OOI Configuration

The OOI ITM system, based on the JIRA platform, will initially be largely used by the Cyberinfrastructure IO in order to help facilitate the collation, organization, distribution and resolution of the various technical issues likely to arise during development and implementation of the OOI. The OOI ITM will also be hosted and maintained by the network administrators at the OOI's IT group at the University of California, San Diego.

In addition to its primary role, OOI ITM will also be used by and benefit the OOI through its contribution and capabilities afforded to programmatic and program management aspect of OOI. The JIRA-based ITM system shall be used as system of record for issues that arise concerning both technical and non-technical tasks, including performance, scope and timelines of operational, managerial, and programmatic tasks within the programs.

#### 4.3.1.1 Infrastructure Layout

There is a standard hierarchy within JIRA that will also be applicable to the OOI ITM. The highest level in the hierarchy is "Projects", which is where the main categories are defined. Within Projects, there are Sub-Projects (also known as Components), Tasks and Sub-Tasks.

Tasks and Sub-Tasks can be a variety of entities, including issues and/or bug. Once created, tasks and sub-task can be assignable.

#### 4.3.1.2 Roles and Permissions

Roles and permissions within the OOI ITM system will follow OOI standards set with the OOI DMS based on Alfresco. There are also options available to change or deviate from these roles and permissions for specific instances only applicable to the JIRA-based ITM, and modifications will be made on a case-by-case basis.

There are three classes of user groups currently defined in the OOI ITM system; however, these are subject to change as OOI project dynamics may dictate:

- Administrators
- Developers
- Users

## 4.4 Subversion

Software Version Controls systems store all source code, as well as a record of all changes and current check-outs of any source code, which all relies on a file-based storage mechanism. The OOI Program Office shall use Subversion, developed by CollabNet, as its Software Version Control system. It supports features such as multiple simultaneous check-outs, conflict resolution, shelving and un-shelving (shelving is a way to save a set of pending changes without committing them to source control, while still making them available to other users), branching and merging, as well as the ability to set security levels on any level of a source tree, alongside the most visible features of document versioning, locking, rollback, and atomic commits.

The source control mechanism integrates with work items as well. For example, when a check-in occurs, a developer can choose to have their code associated with one or more specific work items, thus indicating that the check-in works towards solving specific issues. Administrators can enforce check-in policies that require Code Analysis requirements to have passed, as well as to enforce the association of check-ins with work items, or update the state of associated work items through an issues management system (like flagging a bug as "fixed" in JIRA when checking in code that has the bug fixed). Individual versions of files can be assigned labels, and all files with the same label forms a release group.

The manner and mechanism in which Software Version Control systems store the source code is important. Subversion uses a "filesystem" for source code storage, and is best described as a three dimensional filesystem. Most representations of a directory tree (tree view) are two dimensional, but Subversion's adds a dimension of revisions. Each revision in the Subversion filesystem has its own root, which is used to access contents at that revision. Files are stored as links to the most recent change; thus a Subversion repository is quite compact. The storage space used is proportional to the number of changes made, not to the number of revisions.

### 4.4.1 OOI Configuration

While mostly facilitating the Cyberinfrastructure (CI) IO needs for software version controls, the OOI maintains an instance of Subversion as the OOI Software Version Control (SVC) system hosted and maintained by the network administrators at the OOI's IT group at the University of California, San Diego. The OOI SVC is an integrated repository covering software artifacts for all elements of the OOI Integrated Observatory. In its core structure, it follows the software system breakdown structure.

The software system breakdown structure defines the following hierarchical elements:

- Program (OOI)
  - System (L3): CI, CGSN, RSN, EPE
    - Subsystem (L4)
      - Component
        - Sub-component

The repository is structured hierarchically as structured above. For each of the elements of the structure, it adds specific sub-directories depending on the type of system structure element.

In its instantiation, the OOI SVC repository is a directory hierarchy with directories for system elements according to their position in the structure. Each of the system elements has a substructure according to its type. Components can also have dependencies on other components on the same level. Components and subcomponents are designed that they are least dependent on other components. Individual technologies wrapped as services are candidates for components.

## Configuration Management Plan

While the OOI SVC hierarchy and structure are subject to minor changes, the initial structure follows:

Level	System Element	Type
2	OOI-Code-Repository	Root Node
3	OOI-Documentation	Document Node
3	OOI-Software-Design	Design Node
3	OOI-Verification&Validation	V&V Node
3	CI-SWD	System Node
4	COI-Subsystem	Build Project
5	IdentityManagement	Build Project
4	COI-Integration	Build Project
4	COI-Verification&Validation	V&V Node

### 4.4.1.1 Structure Process

The code repository structure is hierarchical and supports a hierarchical roll-up build process. Each element in the software system structure break-down (program, system, subsystem, component, subcomponent) defines its own build project with a separate “makefile”, build process and “verification & validation” process (plan, procedures, test cases).

Subsystem software components are individually built and tested and rolled up as binaries into the next higher integration project. The integration project kicks in and builds the integrated binary for roll up to the next level. Dependencies between components need to be respected when determining the build order. OOI has tools to do this automatically, such as Java Maven.

The trees are as flat as possible on the component/subcomponent level, while minimizing dependencies. For each of these build projects on the various levels of the system there exists one responsible individual (Component Software Development Manager) overseeing the build process and assisting the integration at the next higher level. This individual also assigns permissions to who can access and modify code in this project. The Software Development Manager of a higher level can fulfill this role also for some or all of the components on lower levels.

### 4.4.1.2 File Versioning Process

Check-in and check-out of files are managed at the level of the respective build project. It is the decision of the SW Development Manager whether to use an optimistic or pessimistic locking scheme.

- **Optimistic locking** relies on no explicit locks on source code and files. It assumes few conflicts and a disciplined testing and check-in process. It is most suitable for agile, small build projects with good communication.
- **Pessimistic locking** issues locks on files to individuals before they are editing. It is much more rigid and can prevent quick turn-around times. Benefits can be seen in larger distributed development teams with high level of contention for specific files. For most source code, this is typically not necessary

### 4.4.1.3 Verification & Validation Process

The test process follows the QA\_QC\_Plan with its V&V strategy for the respective component, depending on its criticality. For low/medium critical components, an agile development approach with continuous integration and automated unit/integration testing is a good choice.

## Configuration Management Plan

### 4.4.1.4 Releases Management Process

Branching and merging follows the SW integration plan defined on the system level. Branches are required when released systems need updates while the development already progresses towards the next release.

### 4.4.1.5 Roles and Permissions

The following table shows each role and permissions for that role:

Role	Key Responsibilities (non-encompassing)
CM Manager	<ul style="list-style-type: none"><li>• Preparing the Configuration Management Plan,</li><li>• Appointing the CM Administrator(s),</li><li>• Controls the establishment of configuration management,</li><li>• Establishes access authorizations</li></ul>
CM Administrator	<ul style="list-style-type: none"><li>• Establishes configuration management and product library,</li><li>• Administering products and product configurations,</li><li>• Establishes the CM procedures regarding the data exchange e.g. with acquirers/partners/sub-suppliers.</li></ul>
SW Development Manager	<ul style="list-style-type: none"><li>• Supports development of the system and enabling system architecture,</li><li>• Designs the system elements in accordance with the system element specifications,</li><li>• Identifies technical risks and chances, using their experience</li></ul>
SW Developer	<ul style="list-style-type: none"><li>• Develops Software Modules</li><li>• Integrates Software Modules into Software Components</li><li>• Supports the Inspector in the test of software elements</li></ul>
SW Integrator	<ul style="list-style-type: none"><li>• Installs and supports a system or enabling system</li><li>• Prepares tests during the development phase and system tests to be demonstrated to the acquirer</li></ul>
Component QA Manager	<ul style="list-style-type: none"><li>• Ensures function and availability of the required measurement and evaluation environment in cooperation with the inspector,</li><li>• Exercises unlimited access to all quality-related processes and all rights in order to execute the above tasks,</li></ul>
Component Tester	<ul style="list-style-type: none"><li>• Uses QA/evaluation environment in accordance with the specifications of the test documentation,</li><li>• Creates evaluation objects using the specified evaluation specification/evaluation procedure and initiating corrective action if required</li></ul>
Quality Manager	<ul style="list-style-type: none"><li>• Creates standards for the quality management reporting system of the projects (as basis for improvement of the quality management system)</li><li>• Executes project and sub-supplier audits</li><li>• Executes audits as required</li><li>• Exercise unlimited access to all quality-related processes and all rights required to fulfill the above tasks</li></ul>
Technical Author	<ul style="list-style-type: none"><li>• Develops acquirer documentation and preparing the documentation concept</li><li>• Prepares training documents and CAT (computer-aided training)</li></ul>

## 4.5 DOORS

Requirements, their definitions, and their facilitation and management, are key to the success of building the OOI. Driven by project management best practices and programmatics, the OOI use of the OOI Dynamic Object Oriented Requirement System (DOORS) as a Requirements Management system, with emphasis on the relationships between, and orchestration of, activities across all OOI lifecycle aspects, including program management, system acquisition, development, transition/deployment, sustainment, and operational use in the context of system of systems is a necessary precondition for success, and is a major focus.

Based on Telelogic's (an IBM company) DOORS platform, OOI DOORS is a Requirements Management system that regulates requirements as modules containing trees of text objects, qualified by an arbitrary number of user-defined attributes, and cross-linked by directional links. It enables requirements traceability through its ability to store multiple documents and tables containing project requirements, attributes, cross-links as well as other information, with no loss of overview, visibility, control or responsibility in the allocating element(s).

OOI DOORS enables these types of capabilities related to requirements management:

- All individual requirements are tagged with multiple attributes and can be filtered by attribute value without losing the remainder of information;
- Requirements are be organized into modules, by level and by subject matter;
- Requirements from separate modules can be linked showing one-to-one, one-to-many, etc. relationships. This is an important feature allowing traceability through the project lifecycle.
- Requirements can be linked to external documents, such as OOI test plans and industry standards

OOI DOORS is more than a requirement management platform. It is a multi-user, multi-access database environment. It ensures that all information, including historical versions, is stored. Long-term, this ensures a full audit trail of the justification and reasoning behind any particular mandated requirement. DOORS also allow coverage and gap analysis, through its ability to give OOI an aggregate level view of requirements and their relationships. It also provides strict configuration management and change control functionality. As an additional capability, OOI can readily import and export DOORS-related information into and out of Microsoft Word and Excel documents and Access Tables.

### 4.5.1 OOI Configuration

The OOI DOORS platform is hosted and maintained by the network administrators at the OOI's IT group at the University of California, San Diego. It incorporates all of the inherent capabilities and general functionality of object-oriented requirement management systems, as well as customized OOI-specific configuration policies and conventions.

OOI DOORS is comprised of various major interlinked objects, including levels, modules, objects and views.

<b>Levels (of a Project)</b>	<p>Levels are the top aggregate of objects within a Project. They define the basic structure and hierarchy of the requirements. OOI has four levels of requirements:</p> <ol style="list-style-type: none"> <li>1. Level 1) OOI Controlling Policies and Overarching Science Themes</li> <li>2. Level 2) User Requirements</li> <li>3. Level 3) System Requirements, and Cross-System Interface Requirements</li> <li>4. Level 4) Subsystem Requirements</li> </ol>
<b>Modules</b>	<p>A database collection of requirements, headings, info, tables, figures, TOC, etc for a specific project entity. Terms related to DOORS modules include:</p> <ul style="list-style-type: none"> <li>• <b>Module Tree:</b> shows levels and modules based on the project's organization and work breakdown structure</li> <li>• <b>Custodian:</b> Person responsible for each module</li> </ul>

## Configuration Management Plan

<b>Objects</b>	Objects are line items within Modules. Objects in DOORS contain - <ul style="list-style-type: none"><li>• ID: A unique number identifying each object.<ul style="list-style-type: none"><li>– Created automatically by DOORS sequentially upon entry of any new OBJECT TEXT</li></ul></li><li>• One of the following:<ul style="list-style-type: none"><li>– Header: Section number and a title describing the contents of that section</li><li>– Information: Any entry that is not a header or requirement.<ul style="list-style-type: none"><li>▪ Use for context, description, general information, explanation, etc. Will normally not be linked.</li><li>▪ May amplify designation by prefacing object text with “Information:” in bold</li></ul></li><li>– Requirement: A verifiable shall statement (usually linked for traceability)</li></ul></li><li>• OBJECT TEXT: The text of a header, information or requirement. (the “main column set of words”).</li><li>• OBJECT SHORT TEXT: A brief summary of the requirement (optional)</li><li>• One or more Attributes: A DOORS term for metadata associated with objects and links, used for additional information, control and filtering</li></ul>
<b>Views</b>	A display of selected objects in a specified format. A view resides in a module, but can draw and display information from other linked modules (higher level, lower level, same) and is named to reflect what is visible
<b>Documents</b>	Any view may be defined as a “document” for a specific need, and exported to Word or a PDF file for distribution or placed in the OOI DMS. A document is typically a collection of requirements, headings, information, tables, figures, TOC, etc., created by a view of a module which may include information from one or more other modules

### 4.5.1.1 DOORS Module Organization

Within DOORS, OOI Requirements modules are organized in sections, using a standard outline, shown below. Each section may be organized into subsections according to content. DOORS maintains an internal hierarchical numbering system for section and subsection headers.

Section #	Section Name and Description
1	Introduction (This section contains descriptive and background information about the module)
2	References (This section is optional; if used, it contains references to related DOORS modules and OOI documents)
3	Requirements (This section contains the requirements, formulated as “shall” statements)
4	Common Requirements (Not currently used)
5	Interface Requirements (This section contains inter-IO and intra-IO interface requirements, formulated as “shall” statements)
6	Adjudication Repository (This section contains requirements that are under review and have not yet been baselined)

### 4.5.1.2 Linking and Traceability

DOORS supports the concepts of links, suspect links, link modules and traceability. A link is a connection between two requirements in the same or different modules, indicating some type of relationship between the requirements. In the OOI DOORS module hierarchy, links are maintained from Level 2 requirement parents to Level 3 requirement children, and from Level 3 requirement parents to Level 4 requirement children. Links are also maintained from Level 3 interface requirements to Level 4 subsystem requirements on both sides of the interface. Links may be analyzed for completeness and consistency using DOORS tools such as the Traceability Explorer and suspect link filters. Suspect links are identified by DOORS when the requirement on one end of a link has changed. A view displays those requirements at the other end of the link with changed object text (and any other selected attributes), and the module custodian may clear the suspect link flag after review and disposition.

### 4.5.1.3 Updating DOORS modules

Changes to DOORS modules are made via Engineering Change Request (ECR) (see section 6). ECRs for changes to Level 1 requirements, Level 2 requirements, and Level 3 interface requirements originate and are approved at the System Level CCB. ECRs for changes to Level 3 system requirements and Level 4 requirements originate and are approved at the IO Level CCB, and may also be promoted to and approved at the System Level CCB, if they represent Class I changes, or at the discretion of the System Level CCB chair.

After an ECR for updating a DOORS module is approved at all required CCB levels, the module updates are finalized in DOORS, and the module is baselined, using the DOORS internal baselining functionality. Baselines for OOI requirements modules are numbered using a pattern of n.m, where m is incremented for each ECR that changes the module, and n is incremented for significant baseline changes, at the discretion of the DOORS module custodian. A baseline suffix of the form <IO>|SL-CCB-yyyy-mm-dd is appended to the baseline number in DOORS, where yyyy-mm-dd is the date of the highest level CCB where the ECR was approved.

### 4.5.1.4 Maintaining Exports of DOORS modules in the OOI DMS

Excel exports of each OOI DOORS module are maintained for Reference Only purposes in the OOI Document Management System (DMS), Alfresco. These exports contain a select set of DOORS attributes. The exports are updated each time the DOORS module is baselined.

## 4.6 Program Management Portal (SAF)

The Program Management Portal (also called SAF) contains a collection of custom designed, web-enabled, collaborative tools. Current tools include Change Control, Risk Management, Instrument Application, and Program Tools (which includes the 1199 document, the WBS dictionary, the OOI Team Directory, and Acronyms and Definitions).

## 5 Design Reviews

Design reviews provide the baseline and milestone points for the system configuration.

### 5.1 Preliminary Design Review (External)

The OOI Program Office and the IOs shall facilitate and conduct a Preliminary Design Review (PDR).

PDR is the external sponsor and science community review conducted to evaluate the progress, technical adequacy and risk associated with the emerging design approach. Emphasis is on complete understanding of the requirements, including environments and operating modes/states, to ensure that subsequent detailed engineering design activities can be initiated with minimum risk of rework or wasted effort. Unlike informal peer interaction or periodic meetings, the PDR is specifically structured for the purpose of validating the requirement set and assessing initial design progress.

The PDR scope ranges from the system level concentrating on aggregate system performance and external interfaces, but also serves as the vehicle to identify and resolve any remaining configuration level item (configuration item) interface or requirements disconnects. Configuration item reviews focus on required behavior within the configuration item and establishing defined interfaces to other CLIs.

PDR shall be conducted after preliminary design efforts, but before the start of detail design. The review shall be conducted for each module and configuration item or aggregate of configuration items to:

- (1) Evaluate the progress, technical adequacy, and risk resolution (on a technical and schedule basis) of the selected design approach.
- (2) Determine its compatibility with performance and engineering specialty requirements.
- (3) Evaluate the degree of definition and assess the technical risk associated with the selected manufacturing methods/processes.
- (4) Establish the existence and compatibility of the physical and functional interfaces among the configuration item and other items of equipment, facilities, computer software, and personnel.

For software, the review shall focus on:

- (1) The evaluation of the progress, consistency, and technical adequacy of the selected top-level design and test approach.
- (2) Compatibility between software requirements and preliminary design.
- (3) The preliminary version of the operation and support documents



## 5.2 Final Design Review (External)

The OOI Program Office and the IOs shall facilitate and conduct a Final Design Review (FDR).

FDR is the external sponsor's review, conducted to validate the design, cost, schedule and program management control systems. FDR scope includes assessment of the technical and project-management components of the program, in order to assess the full readiness for construction, and the level of confidence that the project can be delivered within the parameters defined in the project baseline.

FDR Criteria:

- 1) **Final construction-ready design:** delivery of designs, specifications and work scopes that can be placed for bid to industry-requires:
  - a) Key functional (science, system and sub-system) requirements and performance characteristics, including internal interfaces and interconnections
  - b) System architecture and equipment configuration-including how the OOI will interface with other systems
  - c) Operational concept
  - d) Reliability criteria, analysis, and mitigation
- 2) **Tools and technologies needed to construct the project**
  - a) technical maturity of critical components (including core sensors)
    - i) Industrialization of key technologies needed for construction (made consistently-not necessarily COTS)
  - b) Overall development and production schedule (within resource loaded schedule) of outstanding components in pre-construction phase, including
    - i) Milestone reviews
    - ii) Design reviews
    - iii) Major tests
- 3) **Project execution plan** including [refine and elaborate existing]
  - a) Project organization/governance including
    - i) Organizational structure (tied to WBS-roles, responsibilities, reporting)
    - ii) Governance, including advisory structure
    - iii) Completion of recruitment of key staff and cost account managers needed to accomplish the project
  - iv) Managing sub-awardees**
  - b) Acquisition-Acquisition plans, sub-awards and subcontracting strategy-includes
    - i) Competition strategy
    - ii) Types of contracts to be awarded
    - iii) Contractor(s) responsible for developing and implementing the system, where feasible
  - c) Internal and institutional oversight plans, advisory committees, and plans for building and maintaining effective relationships with the broader research community that will eventually utilize the facility to conduct research [Governance]
  - d) Education and outreach plans
  - e) Environmental compliance (NEPA) [see #7]
  - f) Plans for transitioning to operational status
  - g) Configuration control plans**
  - h) Working with interagency and international partners**
    - i) Finalization of commitments with interagency and international partners

- 4) **Fully implemented Project Management Control System, includes:**
  - a) **Baseline version of resource-loaded schedule**
  - b) **Mechanisms to generate reports-using EVMS-on monthly basis and use as a management tool**
  - c) **Path dependencies, schedule float, and critical path are defined**
- 5) **Updated budget and contingency, including risk analysis, presented in a detailed WBS format with WBS dictionary defining scope of all entries**
  - a) Refined bottom-up cost and risk estimates and contingency estimates
  - b) Refined description of the basis of estimate for budget components
    - i) Majority of cost estimates derived from external information
    - ii) Basis of estimates integrated in WBS dictionary/cost book
  - c) Refined project risk analysis and description analysis methodology
    - i) Risks include cost escalation and volatility in OMB escalators, etc.
  - d) Refined contingency and contingency management (budget, scope, schedule)
    - i) Prioritized scope [to achieve science, time vs. dollars, critical points where in schedule decisions will be made to ensure zero cost overruns. Plan with Front-End options]
    - ii) Integration of prioritized scope in schedule and cost (including O&M for upscope)
- 6) **Fit-up and installation details of major components and commissioning strategy [Fit-up synonymous with interface integration, hardware and software]**
  - a) Systems integration
  - b) Testing and acceptance
    - i) Number of tests
    - ii) Criteria for entering into testing
    - iii) Exit criteria for passing test
    - iv) Where test will be conducted
  - c) Commissioning
  - d) Operational readiness criteria-by component and by project
- 7) **Plans for QA and ESH-reporting and mitigation** [permitting]  
[Plans relative to the construction period including OL management of IO policy and procedures implementation]
- 8) **Updated operating estimates**

Note: Bold items indicate direct reference to the NSF Large Facilities Manual (NSF 07-38)

### 5.3 Internal Final Design Review (*ifdr*) (Technical / Engineering)

#### 5.3.1 *ifdr* Overview

The *ifdr* is an internal **Engineering Technical Design Review** (not to be confused with the NSF Final Design Review process) conducted by the OOI Program Office to evaluate the progress, technical adequacy, and risk associated with the detail design solution **prior to the release of drawings/specifications for manufacture or purchase of materials**. Emphasis is on complete representation of the design; to the degree to which the proposed design meets the associated requirements, the nature, and extent of any derived requirements that are introduced as a result of specific design choices, and the overall risk to proceed into the verification period.

*ifdr*'s are conducted as early as stable design solutions become available. Reviews at this level focus on required behavior within the sub-system or configuration item and establish defined interfaces to other configuration items. Review format is matched to the complexity and risk associated with a given configuration item. A system level *ifdr* is conducted once all configuration item (subsystem and IO system) *ifdr*'s have taken place. The system level *ifdr* concentrates on aggregate system performance and external interfaces, but also serves as the vehicle to identify and resolve any remaining configuration item design deficiencies or open items.

All OOI systems are required to undergo an *ifdr* after the external FDR and start of the Construction Phase of the Program (ARRA and MREFC). This type of *ifdr* is commonly referred to as a critical design review.

#### 5.3.2 Conducting an Internal Final Design Review

At least one week prior to the *ifdr*, copies of supporting materials will be loaded into a common work area for review. At a minimum, the necessary supporting materials shall include the System Requirements Document for the configuration item, evidence showing that the design reflects the requirements and the degree to which it is ready for verification, and a copy of the meeting agenda. Additional lead time and technical content is encouraged whenever circumstances permit.

Representatives from the following organizations are required to conduct the *ifdr*:

- Project Management
- Quality
- Systems Engineering
- System Architect
- Software development manager
- Marine IO senior software engineer
- Documentation authors
- Representatives from key interface areas
- Other parties as may be identified for technical or programmatic expertise

Engineering Requirements documents are reviewed to confirm that they are complete and current:

- Functional requirements identified
- Requirements are traceable
- Modes and states identified
- All interfaces identified
- Environments identified
- Verification methods are identified
- Controlled document format

Risk Assessment is made based on the degree of uncertainty still remaining in the design effort:

- Unknown or uncertain requirements and interfaces
- Key assumptions, confidence level, and dissenting opinions as to validity
- External dependencies
- Remaining actions, responsibility, schedule, and impact

The proposed design is carefully examined to determine whether all requirements have been addressed, are reflected in the design and that the methods selected for performing the required functions are logically sound.

Within one week after the *ifdr*, a set of meeting minutes will be prepared that include the following information:

- Meeting date, location, and purpose
- A summary of key findings and discussions
- Determination of configuration item development status
- A copy of the pre-meeting agenda for reference
- A list of all attendees, including position and institution represented
- A detailed list of all action items arising during the meeting

Action items recorded at these events will be reflected in the Project Level Action Item Tracking System and monitored through closure.

### 5.3.3 Internal Final Design Review Criteria

***An ifdr must be conducted prior to any IO releasing the first production drawings to manufacture components of any production equipment.*** The *ifdr* may be incremental, provided that the completing *ifdr* increment will take into account the inter-relation of the entire system and address issues that arise with respect to conflicts in module fit and operation with relation to each other and the system.

The IOs shall describe the complete subsystem design, highlighting all design changes made since the last review and, and provide rationale for the changes.

For large complex configuration items, the *ifdr* may be a progressive or incremental review, culminating in a system level *ifdr*, which essentially reviews the completeness of preceding *ifdrs* and ensures adequate interfaces between the configuration items.

Entry Criteria:

- Successful completion of all action items related to the previous review (PDR, external FDR)
- Advanced scheduling and published agenda
- Acceptance of all applicable requirements
- Successful demonstration of the prototype system or critical components (if developed)

Exit Criteria:

- Acceptance of published minutes to include list of attendees and sub-IOs
- Completion of all action items assigned to the IOs
- Acceptance of any requirements due at the *ifdr*
- Concurrence from the OOI Program Office/IO members that all issues in the review agenda have been addressed
- The PMO Project Manager is the final arbiter of unresolved issues.

The review shall be conducted for each module and new configuration item or aggregate of configuration items to:

1. Evaluate the progress, technical adequacy, and risk resolution (on a technical and schedule basis) of the selected design approach.
2. Determine its compatibility with performance and engineering specialty requirements.
3. Evaluate the degree of definition and assess the technical risk associated with the selected manufacturing methods/processes.
4. Establish the existence and compatibility of the physical and functional interfaces between the configuration item and other items of equipment, facilities, computer software, and personnel.

5. For new development hardware configuration items, assess the results of produce-ability analyses conducted on system hardware.
6. For new development hardware configuration items, review preliminary hardware product specifications.
7. The IOs shall supply a copy of all existing engineering calculations related to calculating design life. COTS supplied data sheets and COTS supplied calculations for their equipment design life shall be included with the design life calculations as they become available.

For software, the review shall focus on:

8. The evaluation of the progress, consistency, and technical adequacy of the selected top-level design and test approach.
9. Compatibility between software requirements and preliminary design.
10. The updated versions of the operation and support documents.

### 5.3.4 Design Verification

Verification is not a substitute for the word "test", but rather a family of methods by which confidence is gained that the requirement can/has been met. Once the requirement is properly stated, the goal is to select the method most suited to obtaining the needed requirement confidence at the lowest cost and schedule impact to the project.

## 6 Configuration Control Structure

### 6.1 Baselines

The management element of the configuration control process includes the preparation, justification, evaluation, coordination, disposition, and implementation of proposed engineering configuration items or program technical data changes and/or deviation from the requirements. The systematic change management process is progressive and evolves with the maturity and complexity of the program.

Starting with the OOI definition of the preliminary infrastructure, architecture and technical approach, the initial design and capability, cost, and schedule baselines will be set. A baseline update will be made at the completion of the final (production/critical) engineering technical design phase. External advisors will participate in the baseline reviews/updates and provide assessment and advice on the OOI design, capability, cost, and schedule (including any associated risks).

In addition to the baselines noted above, the OOI Program Office will set a performance measurement baseline for use with the Earned Value Management System (EVMS) at the start of the MREFC program. This performance measurement baseline sets the detailed scope of work, associated cost, and detailed schedule for each EVMS control account. The IOs and OOI Program Office shall facilitate the review, for approval or rejection, of all Engineering Change Requests (ECRs) prior to IO incorporation into the TDP.

### 6.2 Engineering Change Classes

All proposed changes shall be categorized as Class I or Class II as defined below. Class I and Class II changes may be included within the same ECR. However, should this be the case, each document identified by the ECR must identify the appropriate class.

#### 6.2.1 Class I Changes

A change shall be classified Class I when the change is to a controlled design document, controlled policy/plan document, statement of work or contract **and** one or more of the following statements apply:

- a) Affects any physical or functional requirement in approved configuration documentation, (form, fit and function as related to a requirement),
  - 1. Technical requirements and specifications that affect reliability, maintainability, availability, form, fit, function or interface characteristics
  - 2. Interchangeability, substitutability, or replace-ability as applied to configuration items and to all subassemblies and parts of repairable configuration items
- b) Affects any approved functional, allocated or product configuration documentation, cost, warranties, or contract milestones,
  - 1. Cost to the OOI program in excess of \$25,000 per control account, singular and cumulative per control account
  - 2. Schedule to the OOI program in excess of 4 weeks increase in work package schedule, singular and cumulative per control account

- c) Affects approved product configuration documentation **and** one or more of the following:
  - 1. safety, correction of a hazard or conformance to applicable design standards,
  - 2. compatibility, interoperability, interfaces, or logistic support,
  - 3. retrofit of tested or delivered units,
  - 4. interchangeability, substitutability, or replace ability of any item down to non-repairable subassemblies,
  - 5. sources on a source control drawing
- d) Affects system configuration to the extent that retrofit (replacement of components) action would be taken on a formally tested or commissioned component.
- e) A reallocation of funding at the Control Account level of the WBS.

### 6.2.2 Class II Changes

A change shall be classified Class II when the change is to a controlled design document, controlled policy/plan document, statement of work, or contract **and** not categorized as Class I, such as:

- a) Minor in nature, such that the cost of processing the change request may equal or exceed the cost of performing the work,
- b) Do not exceed any single difference of 10% of the control account baseline budget or \$25,000 between a control account estimate to complete and the baseline budget to complete, whichever is lower,
- c) EVM items, Task level or lower, under the management of a CAM,
- d) Correction of typographical errors, dimensions, graphical or pictorial representations that do not infer changes to process or technical approach.
- e) A reallocation of funding at or below the Work Package level of the WBS.

### 6.2.3 Deviations

A deviation is a specific written authorization to move away from a particular requirement for a specific number of units or a specified period of time. A deviation is not intended for changes to configuration documents. Deviation requests within OOI have limited utility.

### 6.2.4 Engineering Change Content

Each ECR must conform to the following minimum requirements:

- a) Each ECR must be clear, unambiguous, and describe in detail the change in the description of change, purpose, reason, or category. In addition, the actual change must be identified in the field of the ECR by either a pictorial representation of the change, e.g., "Original/Previous" and "Requested/Proposed" scheme or a detailed description of the change.
- b) Identify whether a retro-fit is required. Should a retro-fit be required, the ECR must identify each system and/or subsystem that is affected by the change(s).

### 6.3 PMO / IO Change Control Boards

The PMO / IO CCBs will consist of the following persons, at a minimum, with the chairperson approving additional guests and members. The Quality Manager and the Safety representative roles may be filled by persons responsible for ensuring that the applicable policies and procedures are followed within the project and may be a member of the project, institution, or company.

Chairperson: PMO / IO Systems Engineer  
Members: PMO / IO Project Manager  
PMO / IO Program Director/Principal Investigator  
PMO / IO Quality Manager  
Design Cognizant Engineers  
PMO / IO Control Account Managers  
PMO / IO Safety representative  
PMO / IO Budget representative

The PMO / IO CCB's roles and responsibilities include:

1. Track, review, evaluate and document all requested changes and board actions.
2. Ensure requested change is beneficial to the system.
3. Evaluate alternatives that would achieve the same results.
4. Evaluate all impacts and their effect on scope, cost, and schedule.
5. Approve or reject requests for change.
6. Document the approved change, and communicate decisions to all stakeholders.
7. Ensure the EVMS system is appropriately updated when a change has been approved.
8. Track and report the cost and schedule changes to the COTR and OOI Program Office.
9. Forward all PMO / IO approved Class I ECRs to the System Level CCB for review and approval.
10. Notification of System Level CCB and provision of documentation for all PMO / IO approved Class II ECRs (notification only, not for approval).

### 6.4 System Level Change Control Board

The System Level CCB will convene in person or via technology once per month, or as required by the needs of the program. The OOI SE's shall provide advance notification to the SL CCB Chairperson of any in-process critical ECRs as soon as practicable to ensure prompt scheduling of the SLB and processing of requests.

The System Level CCB will consist of the following persons, at a minimum, with the chairperson authorizing additional participants and members. The Quality Manager and the Safety representative roles may be filled by persons responsible for ensuring that the applicable policies and procedures are followed within the project and may be a member of the project, institution, or company.

Chairperson: OOI Systems Engineer  
Members: OOI Program Director/Principal Investigator  
OOI Project Manager  
OOI COTRs  
OOI Associate Director (Science)  
OOI Quality Representative  
OOI Control Account Manager (As appropriate)  
OOI Safety representative  
OOI Budget representative  
IO Program Directors/Principal Investigators



## Configuration Management Plan

---

IO Project Managers (PM's)  
IO Systems Engineers  
IO Designated Representative of Project Scientists (PSs)  
Design Cognizant Engineers

The System Level CCB's roles and responsibilities include:

1. Track, review, evaluate and document all requested changes and board actions.
2. Ensure requested change is beneficial to the system.
3. Evaluate alternatives that would achieve the same results.
4. Evaluate all impacts and their effect on scope, cost, and schedule.
5. Approve or reject requests for change.
6. Document the approved change, and communicate decisions to all stakeholders.
7. Ensure the EVMS system is appropriately updated, when a change has been approved.
8. Track and report the cost and schedule changes to the NSF through standard program systems.
9. Forward to the NSF Program Director any Class I change that:
  - Exceed the overall program baselines for cost or schedule.
  - Deviate from the NSF Level 2 requirements sources.
  - Requires reallocation of funding in level 2 of the WBS (specifically WBS items 1.1 through 1.4) in excess of \$250,000.
  - Exceed schedule contingency in excess of 45 days
  - Exceed contingency use in excess of \$150,000

### 6.5 NSF Program Manager Review

The NSF Program Manager will review for approval Class I ECRs, which exceeds the overall program level baselines for budget, schedule or scope, in person or via technology as required by the needs of the program. The OOI Project Director shall provide advance notification to the NSF Program Manager of any in-process critical ECRs as soon as practicable to ensure prompt scheduling of the review and processing of the request.

The NSF PM review will be supported by, at a minimum:

- OOI Program Director/Principal Investigator
- OOI Project Manager
- OOI Associate Director (Science)
- OOI Systems Engineer

(with the NSF PM authorizing additional participants and members as required)

## **7 Change Control Process**

All requested changes must be evaluated for risk and impact on design/capability, schedule, and cost, not only within an IO, but among IOs and within the OOI Program/Project. Changes in the OOI Project are controlled through a formal approval process. The OOI change control process is multi-level with the applicable review/approval level assigned based on the potential impact to the program.

### **7.1 Requests for Engineering Changes**

Changes are generated by a need to increase or decrease scope, changed deliverables, technology complexity, engineering design, a change in funding, an unexpected or unforeseen event, insufficiently defined contracts, vendor delivery problems or from a variety of other avenues.

Whenever any party determines that some aspect of an accepted control account should be changed, then that party must submit a Request for Change proposal to the PMO / IO CCBs. The change control request form is attached to this plan. The change request:

1. Identifies the WBS element in question.
2. Describes the aspect of the WBS element to be changed as part of the request.
3. Includes a description of the cost, schedule and scope impacts, from the requestor's point of view, of leaving the control account as-is compared to incorporating the suggested change. (This provides the Change Control Board a better understanding of why the change is being submitted and what importance it has from the perspective of the submitting party.)

#### **7.1.1 Assessing the Impact of Requested Changes**

Once a Request for Change has been submitted to a CCB, the change is circulated to those parties that the CCB identifies may be impacted by the change. These parties are responsible for producing an estimate of the effects of implementing the proposed change. Proposed changes should account for:

1. Additional management effort to revise the schedule and notify affected parties.
2. Impact on control account attributes.
3. Impact on control account design documents.
4. Impact on quality of the system.
5. The risk increased cost of changes at later stages of the project (exponential factor).

In the interest of efficiency, a CCB may process a series of change proposals as a group, depending upon the frequency and importance of the change proposals.

#### **7.1.2 Change Control Documentation**

A formal document management system will be used for both the formal processing of proposed changes and their documentation. Each request for change whether ultimately approved or rejected will be documented in the system and thus available to all OOI team members.

In addition, all approved changes must be reflected in the EVMS, particularly in the Cost of Work Scheduled.

#### **7.1.3 Funding for Changes:**

Funds required for change implementation must be identified from one of two sources:

(1) re-allocated funds within the respective level 3 WBS account which is the subject of the change, or (2) the contingency budget.

## 7.2 Change Control Board Operation

The CCBs shall provide rapid response to proposed changes as required to maintain the program schedule and minimize cost impacts.

CCB Quorum for an official CCB meeting shall have a representative of each sub-group or specialization represented by at least one member. Board members that are unable to attend a given CCB session may delegate their authority to a selected delegate/proxy. Delegation/Proxies should be provided to the CCB session chairperson as far in advance as possible. Delegation of representation for quorum purposes must be reviewed for approval by the chairperson prior to the session. Attendance may be through physical attendance, phone, computer, or other electronic means. Other project staff or non-collaboration members may attend the CCB (in addition to the CCB members), with the approval of the chairperson.

### 7.2.1 Records of Meetings

Minutes of CCB meetings shall be maintained and include (at a minimum):

- Attendees and Delegates
- Change Requests reviewed
- Actions
- Change Request outcomes
  - (additional information requested, approved, denied, closed, etc.)

### 7.2.2 Timing of Meetings

The CCB shall be held at least monthly, but may be convened more frequently at the chairperson's discretion and the needs of the program. The CCB chairperson has discretion to cancel or postpone a meeting based on mitigating circumstances in the best interest of the program.

## 7.3 Change Control Board Authority

In conjunction with the Change Class descriptions in Section 6, the CCBs shall operate within the following authority.

### 7.3.1 PMO / IO CCBs

The PMO / IO CCBs are authorized to manage the configuration of the system within (below) the following thresholds:

Authorized:

- A technical change that does NOT change the scope of work, interface, or a deliverable for the project.
- A change to the Integrated Master Schedule (IMS) less than one month, as long as the critical path is not affected.
- A reallocation of funding at or below the Work Package Level of the WBS.
- A change that does not require use of contingency funds.

The PMO / IO CCB forwards to the System Level CCB any PMO / IO approved/recommended changes meeting or exceeding the following thresholds:

- A technical change that changes the scope of work, interface, or a deliverable for the project.
- A change to the IMS greater than one month, or that is on or affects the critical path.
- A reallocation of funding at or above the Control Account Level of the WBS.
- A change that requires use of contingency funds.

### 7.3.2 System Level Change Control Board

The System Level CCB will review changes forwarded by the PMO / IO CCBs, and by consensus either approve or reject the change request, noting any dissenting opinions. The system level review ensures any potential technical and project management **integration** impacts are evaluated.

The System Level CCB will also review changes forwarded by the Risk Management Board (RMB) in response to handling risk and opportunity items that involve changes in scope, requirements, changes to the IMS, or use of contingency funds. The functioning of the RMB is described in detail in the *Risk Management Plan 1007-00000* and is part of the integrated change management controlling the mitigation of risk. The CCB will make the final decision for changes in risk handling based on the recommendations of the RMB and their own review of the items.

However, the System Level CCB will forward to NSF for concurrence any recommended changes that:

- Exceed the overall program baselines for cost or schedule.
- Deviate from the NSF Level 1 requirements sources and Level 2 requirements.
- Requires reallocation of funding in level 2 of the WBS (specifically WBS items 1.1 through 1.4) in excess of 10% of the level 2 value, or greater than \$150,000.00.

### 7.4 Review Authority

Authority	Class I Change	Class II Change
NSF	Review and Approve Class 1 ECRs exceeding OOI Baselines of Cost, Schedule or Scope, Level 1 and 2 Requirements	Not Reviewed
System Level CCB	Review and Approve All Class 1 ECRs	Notification Only
PMO / IO CCB	Review and Approve (Individual PMO / IO)	Review and Approve (Individual PMO / IO)

Note: See Appendix 2 for additional details.

### 7.5 Consensus and Resolution

The CCBs shall adjudicate each ECR by consensus noting any dissenting opinions. For a given ECR, the CCBs shall choose to Defer, Approve, Approve with Liens, Disapprove, or pass the ECR to the next higher CCB.

If a majority consensus cannot be reached by the CCB, the chairperson shall notify the next higher level CCB chairperson and provide a copy of the ECR with notes from the review. The next higher level CCB will review the ECR and provide guidance for the originating CCB to assist in the decision, or provide contractual direction through the Program Office.

In the case where the System Level CCB cannot reach a majority consensus, the chairperson shall request input from the advisory resources maintained by the program. The System Level CCB chairperson, after consultation with the advisory resources, may provide binding direction to the CCB and approve or disapprove the ECR.

### 7.6 ECR Closeout

For ECRs that have been Approved or Approved with Liens, the applicable OOI, PMO, or IO Configuration Manager shall take actions to closeout the ECR in a timely fashion. The owner of the document in question is accountable for completion of any Liens and/or Actions. Administrative items are generally expected to be completed within two week of the CCB. Other items are expected to be

## **Configuration Management Plan**

---

completed within four weeks of the CCB. The applicable OOI or IO Configuration Manager shall be accountable for closing the ECR in the Configuration Management tool and for posting the completed document to the applicable Document Management System.

## **Appendix A-1 Engineering Change Request Form**

Appendix A-1 has been deprecated. The Engineering Change Request form is electronically maintained as part of the Software Architecture Framework Change Management Application.



## **Attachment B-1 OOI Program Terms and Definitions**

Attachment B-1 has been deprecated. The OOI Program Terms and Definitions are electronically maintained as part of the DOORS requirements database application.

## **Attachment B-2 Site Names and Codes**

Attachment B-2 is maintained as a separate file, DCN 1000-00001.

## Configuration Management Plan

	Class I ECR		Class II ECR
<b>OOI Change Control Process</b>  <b>(quick reference, see sections 5 and 6 for full requirements)</b>	Physical or functional requirement in approved configuration documentation (required Form, Fit, and Function, Safety). Cost or Schedule to the OOI program in excess of \$25,000 cost per control account or in excess of 4 weeks increase in work package schedule, any critical path schedule impact, singular and cumulative for both cost and schedule (non-critical). Technical requirements and specifications that affect reliability, maintainability, availability, or interface characteristics. Configuration to the extent that retrofit (replacement of components) action would be taken.		Minor in nature, such that the cost of processing the change request may equal or exceed the cost of performing the work. Do not exceed any single difference of 10% of the control account baseline budget or \$25,000 between a control account estimate to complete versus the baseline budget to complete, whichever is lower. Task level, under the management of a CAM. Correction of typographical errors, dimensions, graphical or pictorial representation. Schedule less than 4 weeks non-critical path.
<b>Board Level</b>	<b>Class I Authority</b>		<b>Class II Authority</b>
PMO / IO	Review and approval all PMO / IO internally generated ECRs. Non-Critical Path up to one month. Funding within WP level. No contingency.		Review and approval of all PMO / IO internally generated ECRs.
	Forward all Class I ECRs to System Level CCBs that change scope of work, interface or a deliverable. Changes that exceed one month in schedule impact and any on the Critical Path. Funding changes at a Control Account Level.		Notify System Level CCB of all Class II approvals.
System	Review and approval of all PMO / IO approved Class I forwarded over the PMO / IO threshold.		
	Forward to <b>NSF PM</b> for concurrence any change that exceeds the overall program baselines for cost or schedule. Any deviation from the NSF Level 1 Requirements. Any change that reallocates funding at the level 2 WBS in excess of 10% of the level 2 value, or use of contingency above \$150,000.		
NSF PM	Review for concurrence any ECR any change that exceeds the overall program baselines for cost, schedule, or deviates from the Level 1 Requirements. Any change that reallocates funding at the level 2 WBS in excess of 10% of the level 2 value.		

**Appendix A-2 OOI Change Control Process Table**