VERA C. RUBIN
OBSERVATORY

Vera C. Rubin Observatory
Systems Engineering

Project-wide Documentation Proposal for Rubin Observatory Operations

(Report from the Rubin Observatory Documentation Working Group)

Chuck Claver (chair), David Cabrera (co-Chair), Rob McKercher (co-chair), John Andrew, Diane Hascall, Patrick Ingraham, Tony Johnson, Kristen Metzger, Austin Roberts, Jonathan Sick, Matthew Rumore.

SITCOMTN-014

Latest Revision: 2022-11-18

DRAFT

Abstract

This technical note is a report to recommend a future state for Rubin Observatory Operations documentation by the Project-wide Documentation Working Group, as a final deliverable for the charge described in `LSE-489.lsst.io`. This proposal presents a high-level documentation strategy for Rubin Observatory Operations with suggested methodologies to transition Construction Project documents and organizations for the technical documentation package needed to establish construction completeness and operational readiness. The presented implementation plan for transitioning includes an estimated level of effort, resources, scheduling and risks if not implemented, as well as a method of sustained maintenance throughout Operations.

Change Record

Version	Date	Description	Owner name
0.0	2021-06-26	First draft of preliminary content.	Chuck Claver
0.1	2021-08-30	Input to Documentation Portal.	Jonathan Sick
0.2	2021-09-14	Preliminary content, Portal architecture and effort.	Chuck Claver, Jonathan Sick
0.3	2022-11-18	Incorporate draft from Confluence pages and shared files as pre-draft.	David Cabrera, Matthew Rumore

Document curator: Matthew Rumore
Document source location: `https://github.com/lsst-sitcom/sitcomtn-014`

Contents

Project-wide Documentation Proposal for Rubin Observatory Operations

1 Introduction

This document serves as the proposal developed by the Vera C. Rubin Observatory Project-wide Documentation Working Group to fulfill its charge defined in LSE-489. Included are recommendations and suggestions for a systematic structure and methodology within the Rubin Observatory operations to create, update and release different types of technical documentation; collect them from the technical groups across the different system within Rubin Observatory; classify the information by system, purpose and users' needs; and, create or modify tools to manage and retrieve information with reliability within a flexible and extendable system. The Documentation Working Group considered the wide berth of topics and technical areas, their differing requirements and processes, the variety of produced information, and large, diverse population involved with Rubin Observatory. The strategies in this proposal aim to support Rubin Observatory staff to reliably and confidently demonstrate operations readiness [SITCOMTN-005] and execute operational expectations throughout operations — lasting impact on science, strong work safety and environmentally sustainable culture with continuous improvement, and integrated and diverse community interaction.

To guide the proposal, the Documentation Working Group reported in chair on its effort to inventory existing documentation and repositories predominant focused on the construction projects: the National Science Foundation (NSF) major research equipment and facilities construction (MREFC) effort managed by the LSST Corporation (LSSTC), and the Department of Energy (DOE) major items of equipment (MIE) effort for the Legacy Survey of Space and Time (LSST) Camera (LSSTCam) managed by SLAC National Accelerator Laboratory. Information is located on different official and non-official repositories, some considered temporary since they will not be continually supported throughout operations. This proposal includes steps to transition documentation from the construction, commissioning and pre-operations efforts into operations within an integrated, project-wide approach. An evaluation of the level of effort and resources is addressed, as well as potential impacts and risks.

As a measure to systematically and methodologically handle the information, four *Documentation Views* have been developed to create an efficient internal structure that bases its approach on the users, staff and communities interested on accessing the information. Their primary intent is to record, classify and categorize documentation, catalog their respective repositories, define ownership and target audiences, and organize relationships and dependencies between documents. The Documentation Views are detailed in Section 2. Crucially, this structure should sustainably allow groups to identify primary sources, i.e., the normative source of information, thereby supporting an orderly way to reference information project-wide. The Documentation Working Group intends the implementation of the Documentation Views will constitute a deliverable from the construction projects to the operations teams as a method to show construction completeness and operations readiness. It is critical that this effort and responsibility is shared between construction project subsystems, pre-operations and operations technical staff, particularly during knowledge transfer and documentation strategy transition stages. The managing groups will complete the Views such that their documentation are organized in a common way and operations staff are capable of managing the content before taking ownership. It is expected that no content currently used in construction or pre-operations will be lost. This content should be maintained as-is and available for any future needs; however, it is not necessary to incorporate everything into the future documentation scheme described in this proposal. A process should be devised to determine if information should be transitioned or archived.

The Documentation Views will fuel the proposed *Documentation Portal*, a web application to provide access and discovery to different documents and documentation types and assisting users in retrieving information and its primary source. The proposed architecture includes products already implemented for Rubin Observatory, namely the LSST the Docs (LTD) documentation delivery platform. The Documentation Working Group expects the majority of effort for software developers will be understand the use-cases and how group's utilize the Documentation Views in practice and the development of software tools to serve the Documentation Portal via repository metadata. The new Documentation Portal is detailed in Section 3.2.

Project and team managers with product owners should apply a graded approach to this proposal's recommendations and suggestions, preferably tailoring how the create and manage the Views and workflows for their ability to maintain reliable information throughout operations. This proposal is meant to provide guidance and framework for staff responsible for technical information and effectively communicating it, and the Views must be created for their specific needs. If people are having difficulty finding information, determining where the source of truth is located, or managers/auditors cannot determine if construction completeness is being achieved, it's appropriate for the technical group to reflect on how they can expand on using the Documentation Views for improvement. While documentation specialist staff will assist in the development and implementation, technical staff that own the information are primarily responsible, and internal or external stakeholders or customers should provide feedback. Examples such as lists and diagrams are provided throughout this document (e.g., Section 2.2, Appendix A) to demonstrate the core concepts. The Documentation Working Group made an extensive effort to use currently available information for direct application across a small set of systems while validating the proposal. They may be used as an initial starting point or adapted to the specific needs and processes of those served by Rubin Observatory. The Documentation Working Group highly encourages exploiting software to retrieve information from repositories to help or automatically build the Documentation Views. Section 3 includes recommended workflows and a path forward of specific repositories for operations.

2 Documentation Views

There are four *Documentation Views* developed with customized categories and subcategories to functionally describe and organize the technical documentation for Rubin Observatory. The set of categories were developed by the Documentation Working Group to assist the group responsible for the documents in considering users' or communities' interest and their informational needs during observatory operations. Each view is represented by a *tree structure*, a widely used way of representing hierarchical structures typically in a graphical form. Examples in this proposal include classical node-link diagrams and tree view outlines that are built with *branches* of connected nodes, or *leaf-nodes*, starting from *root-nodes* representing the highest level in the hierarchy. Different tree structure diagrams can be used depending on the particular use case and how to effectively represent the information. (Wikimedia Foundation, 2021)

Each View has its own purpose. They are designed to provide a framework structure for staff and users to search, reference and retrieve currently available information while considering their expected interests consistently project-wide. In addition, the Views will provide an orderly way to introduce new documentation and meet an extended goal of facilitate referencing and minimize replication.

The four views are:

- **Product View** — For organizing the ownership of documentation, describing internal systems and providing structure for linking and cross referencing documentation or informational dependencies. Notably includes the separation of Rubin Observatory Departments.
- **Storage View** — For describing all project documentation storage locations and repositories. In conjunction with the Product View, identifying or determining the normative source of information.
- **Access View** — For describing the user base and documentation applicable to their use cases.
- **Topic View** — For searching and discovering documentation.

The owner or a designated responsible group will use the Documentation Views to develop an effective way to communicate the various pieces of documentation, how they are interconnected, dependencies or connections to another group's documentation, and utilization for various documents. Stakeholders (e.g., internal and external technical groups, Rubin observatory communities) should be able to retrieve current information with consistency and reliability. It is the responsibility of the owner or designated responsible group to keep this information up-to-date and useful to all stakeholders. Stakeholders should provide regularly feedback over the course of operations.

This framework will improve the ability of pre-operations and operations staff to clearly and robustly establish the criteria and requirements for construction completeness and operations readiness; for example, all identified normative sources of information should be completed, or a use case where multiple documents referencing one normative source better serves multiple stakeholders than one document. The framework should also contribute to manager's and auditor's ability to understand the system at-hand for their consideration; for example, a document that is referenced isn't identified as a normative source of information or requires a different repository (e.g., DocuShare), or interfaces between systems should be further defined to ensure operational requirements are met.

The following subsections propose how each Documentation View tree is constructed, various examples of use, recommendations for technical teams to consider, and capture any assessments of needed resources.

2.1 Product View

The *Product View* is based on a common approach used in product management, a *product tree diagram*. The Documentation Working Group developed a framework customized for Rubin Observatory operations to facilitate the development of documentation and effective communication consistently project-wide. The two primary purposes are to convey ownership and responsibility of technical documentation and more readily describe the complicated systems with context and categorization. The Product View allows one to identify information as a normative source (e.g., one source of truth), then provide a manner which one can associate, link and cross reference documentation and informational dependencies from respective normative sources or references thereafter. The expected primary users are managers, product owners, engineers, specialists, technicians, and users which want to search downwards in a system's hierarchy.

The Product View will consist of product trees developed by the owning *Rubin Observatory Departments*, their technical teams and product owners. Each department's product tree(s) should be constructed to best organize and compartmentalize the set of *products* that make up the complicated systems under their purview. These "products" can span all manner of objects; such as department-specific categories, documents, hardware (systems, servers, networking, etc.), data (data sets, validation tests, verification artifacts, etc.), software interface information (alarms and notifications between hardware or software systems, support data storage and metadata schema, etc.), or subject-matter expert support. Each department or group is responsible to manage the product trees in addition to updating their technical documentation in a fashion that supports stakeholders' access to the associated information, interfaces and requirements. Stakeholders (i.e., department technical groups, other departments, or external communities) should be able to retrieve current information with consistency and reliability. In light of this need, the Documentation Working Group recommends all Product View trees be available via one of the project repositories — it is suggested to create an appropriate lsst.io website so departments and groups changes can easily make changes via project-wide and the department's processes.

Products in the Product View are defined by the owning department in terms of a system or set of systems which can be grouped or decomposed into subsystems or individual components that is appropriate for them and their stakeholders. The root-nodes and first set of leaf-nodes of each product tree are a predefined set of topics and categories created by the Documentation Working Group — the *generic categories*. The generic categories were developed to capture critical operational aspects of a product while being extendable to subsystems of a product, where the subsystem may be a product with subsystems in its own right; i.e., a *Level-1 product* can be comprised of multiple Level-2 or lower-level subsystems, and those subsystems which are products are *Level-2+ products*. Importantly, these terms do not correspond to the construction phase definition of Work Breakdown Structure (WBS) or colloquial "subsystem" used across the construction project. It is at the departments discretion on how to best organize and characterize these products and their product trees to manage their systems and flow of information. It is expected that all generic categories are applicable to the product trees associated with Level-1 and Level-2+ products alike. While there will be exceptions for low-level systems, the Product View and its generic categories are designed to robustly capture construction completeness and operations readiness. They should generate sufficient discussion between technical groups and the stakeholders to ensure key information is identified and provided.

2.1.1 Departments for Rubin Observatory operations

The Rubin Observatory departments are described as to their associated scope, systems and/or products descriptions, and contacts to identify managers, product owners, or other key personnel (e.g., organizational chart, contact list). This section lists each department and respective scope. If the scope description below does not already include the systems and products owned by the department, a high-level product tree diagram should be created. Contact information should be available to individuals that require it, and it should be clear when it's appropriate to contact the individual or group.

Director's Office (DO) — The Vera C. Rubin Observatory Director's Office is responsible for the overall management of the observatory and the LSST survey, as well as fulfilling the mission of the observatory and realizing its vision. The Director's Office includes a Directorate, Administrative Operations, Safety, Communications, and In-Kind Contributions teams.

Observatory Operations (OO) — The Chilean-based Rubin Observatory Operations Department is responsible for operating and maintaining the telescope, camera systems, and summit facilities in order to collect the raw imaging and housekeeping data needed by the LSST. The primary tasks include maintaining the operating facilities, conducting the night-time survey operations, real-time assessment of image quality and observing efficiency, performing the daily calibration, and collecting and analyzing engineering data.

Data Production (DP) — The role of the Rubin Observatory Data Production department is to accept data from the Observatory's telescopes and ancillary systems; to process that data to generate science ready data products; to archive both raw data and derived data products; and, subject to approval from the Science Performance department and the Data Release Board, to make that data available to the scientific community. The Data Production department will develop, maintain and operate the networks, compute and storage hardware, and software that constitutes the Rubin Observatory Data Management System for the duration of the operational period.

System Performance (SP) — Rubin Observatory System Performance department is responsible for ensuring that the LSST as a whole is proceeding with the efficiency and fidelity needed to achieve its science requirements at the end of the 10-year survey. This includes the Wide-Fast-Deep (WFD) survey and all Special Programs (deep drilling fields and mini-surveys). To meet this goal, the System Performance department will track and optimize the integrated performance of the entire system. This includes the performance of the observatory and the progress of the survey with respect to its science objectives, the ability of the community to access and analyze the data and publish results on the four LSST science pillars at an appropriate rate, the evaluation of strategies for improving the survey strategy, and the development of mitigation strategies together with other relevant departments to minimize the impact of changes in the system performance on the overall LSST science.

Education and Public Outreach (EPO or EP) — The mission of the Rubin Observatory Education and Public Outreach program is to offer accessible and engaging online experiences that provide non-specialists access to, and context for, Rubin Observatory data so anyone can explore the Universe and be part of the discovery process. EPO serves as a website that highlights and contextualizes the scientific power of Rubin Observatory for non-specialists and hosts all online resources.

2.1.2 Generic Categories for the Product View

The set of generic categories are provided as a basis to define and associate critical systems and objective elements for each department and their systems; they are not products in themselves. They are designed by the Documentation Working Group to relay information required to support and evaluate operations readiness and operations specifically for Rubin Observatory in a concise manner by standardizing the distribution of information and products. Further opportunities arise with a well designed set of product trees: clearly establish relationships and dependencies between systems, serve to introduce the department/system in a digestible manner, and create a more adaptable structure to organize and target information between technical groups internal or external to the department.

The generic categories are separated by five high-level categories — Technical Design, Procedures, Safety and Emergency Response, Evaluation and Archival Documents. Each high-level category includes a few subcategories, some of which have further subcategories. The generic categories are meant to apply to a variety of systems (e.g., hardware-centric, software-centric, hardware and software distributed, process and protocol dedicated) such that information is associated with all categories and subcategories for practically all systems or subsystems described via a Product View tree. It may be difficult at first for technical groups to perform a logical and relevant decomposition of the systems such that the generic categories are applicable to all product trees, especially if a system can change in different scenarios, contexts or states (e.g., one or more product trees could account for maintenance or on-sky operations of the Simonyi Survey Telescope, each state's different components and changing interdepartmental interfaces). However, even in the case where the owner and stakeholders agree a category doesn't apply, the generic categories are a way to discuss, bound and described inter- and intra-departmental interfaces, requirements and key expectations.

It is important to note that the Product View is specifically intended to not have teams recreate or reproduce documentation or technical information. The Product View should be used to collect and categorize information and documentation types, identify normative sources, and help identify and resolve any gaps between the group(s) and stakeholder(s). Teams can and should refer to appropriate documentation or information therein to create product trees; this can sometimes take the form of a diagram or a narrative, and the representation may differ between the categories within a product tree. Within the Product View as a whole, consisting of many product trees, it is intended systems and subsystems refer to higher- or lower-level product trees to reduce replication and the risk of confusing users accessing information. Furthermore, the referential nature can be applied for interdepartmental information and dependencies such that it's clear which department owns the information and which departments utilize it, e.g., interface control documents (ICDs) and the N-squared diagram could be sufficient references.

Here are the generic categories:

- Technical Design

  - System Description
    * Description of System(s)
    * Definition of Sub-systems
  - Technical Design Specifics
    * System-level and Intra-department System Interfaces
    * Sub-system Level Information
    * Inter-department Interfaces
  - Access Interfaces (Physical and/or Software)
- Procedures
  - Operational Procedures
  - Maintenance Procures
    * Preventive Maintenance
    * Reactive Maintenance
    * Turn-over Protocols (e.g., shift change, operational-to-maintenance status change)
    * Sub-system Isolation
  - Software Access and Use Documentation for Users
  - Software Development Documentation for Developers
  - Manuals and Data Sheets
- Safety and Emergency Response (System-level and relevant Sub-systems)
  - Emergency Procedures and Contacts
  - Hazards and Hazard Analysis
  - Mitigations and Verification Artifacts
    * Protocols
    * Energy Isolation
- Evaluation
  - Performance
  - Failure Effects and Failure Mode and Effects Analysis (FMEA)
  - Validation/Verification Test Plans
  - Test Reports
    * Technical Reports (Test and/or Analysis)
    * Verification Reports
- Archival Documents
  - Communications (e.g., Request for Information [RFI])
  - Construction Information, Unrelated to Commissioning/Operations

2.1.3 Examples of Product View trees

Here are two examples of product tree referencing, using Auxiliary Telescope (AuxTel), LSST-Cam and Commissioning Camera (ComCam). Note that it would be beneficial to design product trees which take advantage of the similarities between LSSTCam and ComCam, even though there will be major differences, too. For these three systems, the Access Interfaces category would include a common set of software, e.g., LSST Observing Visualization Environment (LOVE), Nublado (Jupyter) interface, Script Queue, Watcher. This common set of software could be included in a higher-level product tree (potentially a Level-2 product under the Observatory Operations department) or as a referential Level-2+ product trees; then, not only is the information referenced to all three efficiently, but the software product tree(s) can be designed to easily indicate differences between the three systems, say, for the Software Development categories. As a second example, the three systems interface with a set of external systems simultaneously connected to them, e.g., Engineering Facility Database (EFD), Environmental Awareness System (EAS), Global Interlock System (GIS). These external systems could also be set up as referential product trees which are referenced by AuxTel, LSSTCam and ComCam, as well as others.

Figure 1 depicts the Rubin Observatory departments and a subset of their systems and products into Level-1 products, with some Level-2+ products included.

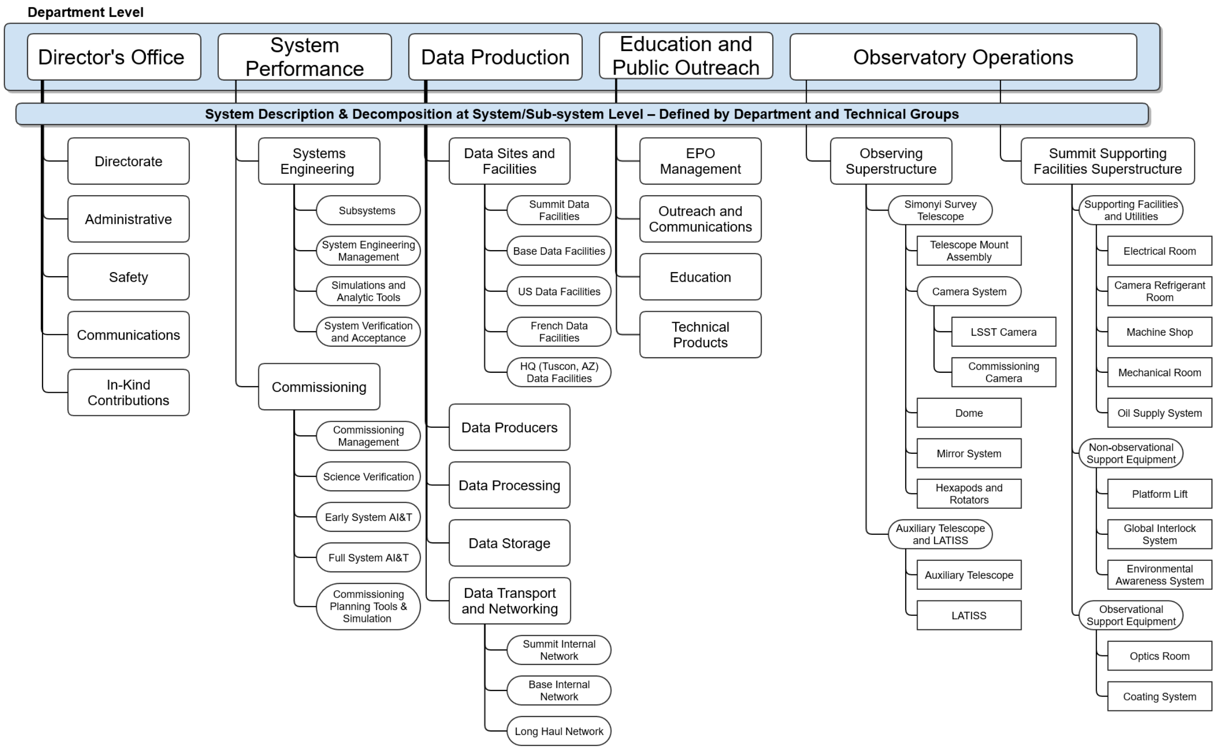
Figure 2 is a more comprehensive example depicting the Technical Design high-level category from the generic categories. Each subcategory (yellow) and each sub-subcategory (blue) is addressed such that the reader has an idea of what the system is, how it is used, and documentation with references therein for retrieving critical information.

Note that while it may seem Figure 1 and Figure 2 are part of one example, that is not the intended purpose. As described in the first example regarding AuxTel, LSSTCam and ComCam, it would be beneficial to organize the Product View to leverage the replicated information such as common software. This was not considered for the two figures.

2.2 Storage View

The purpose of the *Storage View* is to capture the official repositories retaining, archiving, organizing, and accessing Rubin Observatory technical documentation in a reliable, consis-

FIGURE 1: Example of Department Decomposition at System and Subsystem Level.



	Rubin Observatory Operations	**System Performance**	**Data Production**
Technical Design	ICDs/IDDs Requirements	Validation of Science Platform	Science Platform Design and Operations
Procedures	Day/Night Shift Turn-over, Energy Isolation Protocols, Calibration Procedures	Predictive Analysis, Hazard Validation	Data Release and Prompt Alerts Processing
Evaluation	Verification Test Plans, Failure Effects, Test Reports	Failure Mode and Effect Analysis	Assertion Testing

tent manner for users, developers and management. As a fundamental corollary, the Storage View provides a method for departments and technical groups to define the normative source for information, i.e., a single, definitive source of truth. With canonical information storage, others can develop associations, links and references to impart information to inter-departmental or external documents or groups, allowing a reduction in replication, reliable information flow, and preservation of informational dependencies. Unless mandated by the project, the department or responsible group can choose the manner and method of storing

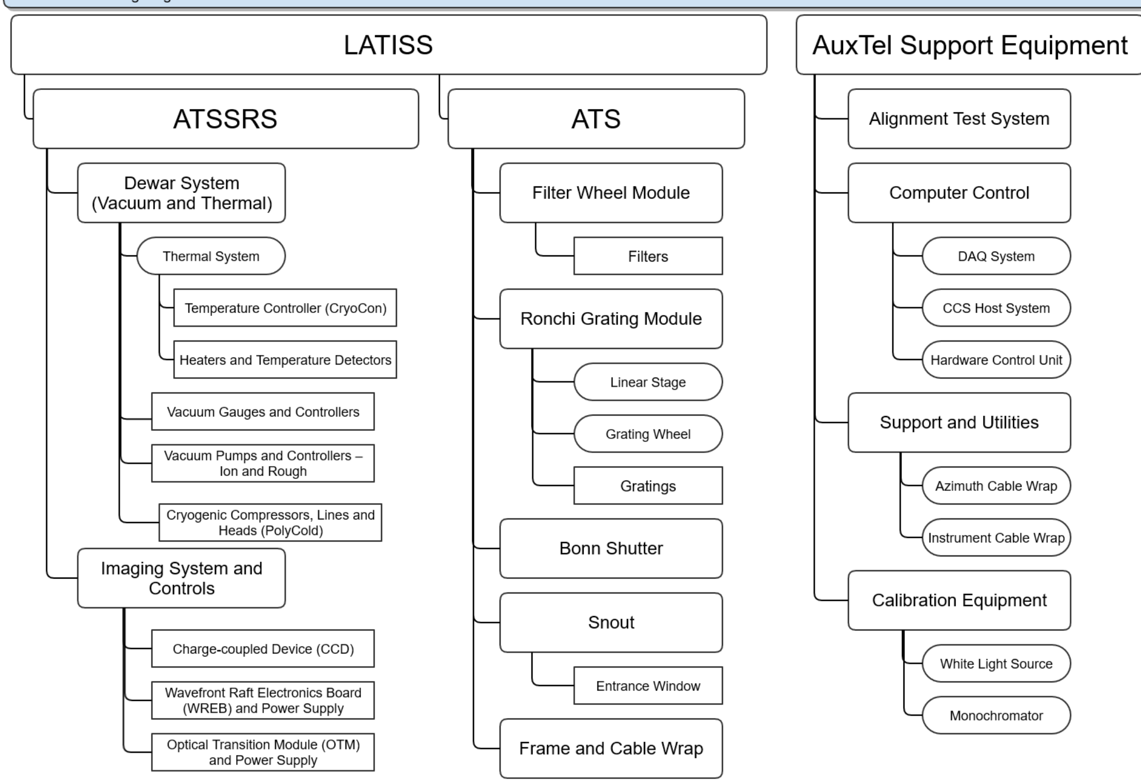
FIGURE 2: Example of Technical Design Generic Category from Product View, for an Auxiliary Telescope Subsystem.

System Level Product Tree – Auxiliary Telescope and LATISS – Technical Design

System Description

Description of System(s)
The Auxiliary Telescope (AuxTel) is a 1.2-meter telescope responsible for providing atmospheric calibration data for the main Simonyi Survey Telescope – see LTS-521 for AuxTel requirements. Two instruments are generally mounted on AuxTel using two nasmyth ports: a high-speed CMOS camera used for alignment testing, and the LSST Atmospheric Transmission and Slitless Spectrograph (LATISS) Instrument. the LATISS instrument is composed of two components: the AuxTel Spectrograph (ATS) Instrument and AuxTel Spectrograph Sensor and Readout System (ATSSRS) – see DOC-30997 for details on ATS and TSTN-006 for details on LATISS.

Definition of Sub-system(s)
See the following diagram.



Technical Design Specifics

System-level and Intra-department System Interfaces
Equipment is controlled by software used across the project: Observatory Control System (OCS), Telescope Control System (TCS), Camera Control System (CCS), LSST Stack, and Data Acquisition System (DAQ) – see respective product trees for information.
See TCS product tree regarding LTS-306 and LTS-307 Interface Control Documents (ICDs).
See the Global Interlock System (GIS) product tree regarding the AuxTel Safety Controller.

Sub-system Level Information
Two instruments are generally mounted on AuxTel (port is interchangeable): nasmyth-port-1 contains a high-speed CMOS camera used for alignment testing, and nasmyth-port-2 contains the LATISS Instrument. The LATISS instrument is composed of the ATS Instrument and ATSSRS – see DOC-30997 for includes details on ATS and TSTN-006 for details on LATISS. The ATS has three modes: direct imaging mode, high-resolution mode and low-resolution mode.

Inter-department Interfaces
LTS-496 is the mechanical ICD for the AuxTel instrument.
LTS-429 is the ICD between the AuxTel Lower Enclosure and Dome.
LTS-428 is the ICD between the AuxTel and Pier.
DOC-30997 includes the ICD between the ATS and camera.
Other AuxTel internal ICDs are found in DocuShare Collection-6445.

Access Interfaces

Physical Interfaces
Physical interfaces are not included for this example.

Software Interfaces
DOC-30997 describes ATS software interface.
Other software interfaces are not included for this example.

information and transposing information to other locations, in conjunction with the stakeholder(s). The primary users are managers, engineers, specialists, technicians, web development staff, system administrators, and documentation specialist.

To ensure a high degree of reliability, it is crucial staff store all operational and historical information within officially designated locations. Specifying official repositories will limit the level of effort in the utilization and maintenance of multiple documentation systems, prevent or highly discourage using non-official storage repositories, and reduce risk to storing and accessing operational and historical information. By limiting the available repositories, effort and resources can focus on best using the official storage locations or modify them in an orderly, consistent manner.

As a shared responsibility between system administrators and departments, all official repositories should be well maintained. System administrators must ensure access throughout operations (e.g., readily available and backed-up) and a level of expertise must be on-hand for maintenance, issues and general assistance to staff. Departments and staff must maintain their information and its organization within the repositories. They should recognize and act when information or utilization becomes outdated or unused as to prevent large amounts of depreciated content, stress on storage and bit rot. This is especially important to the development, deployment and maintenance of the Documentation Portal architecture (Section 3.2), as leveraging specific use cases and repository's native or project-developed metadata fields are key components. Additionally, this effort will better suit the project and departments to formalizing a long-term organizational structure of each repository, create customized workflows, transition to the Documentation Views, and implement new or updated documentation.

Non-official storage locations pose unique operational and managerial risks which can lead to information and data loss, increased information security risks, and impact to the project's schedule and budget. This includes information control (e.g., access, enforced version control, archiving), reliable data recovery, unknown or unplanned risks, and lack of available support (e.g., available labor, expertise). Platforms may become unsupported or unavailable and the information therein becomes lost or otherwise inaccessible. This risk has already been realized within the construction project, although of minor impact, as evident from a small number databases (e.g., personal drives, old network drives) that are difficult to access or have lost or inaccessible information. [chair] Additionally compounding the risk to information access, it becomes increasingly difficult to share information, especially when the repository is ill-defined, uncertainty of what replicated information is the most current, and uncertainty of which repository is the normative source of information. Without a discrete set of storage locations, it is impossible to guarantee reliable and navigable information or data across multiple platforms throughout operations.

The Storage View tree is similar to a product tree, with the basis of design a result of the Construction Documentation Inventory. This section summarizes some important details from the document, but chair includes more information such as the inventory for each repository. Each repository is a root-node, similar to a Level-1 product, and the first set of leaf-nodes captures the location's organizational structure and its metadata structure. Critically, the Storage View trees should readily indicate what are the normative sources of information so other Documentation Views can reliably refer to the accurate and up-to-date information. It will be the responsibility of the department or technical group to define the organizational structure, with documentation specialists assisting in developing a consistent approach across the project and sharing lessons learned between groups. Software developers require the metadata and its structure to develop the Documentation Portal, including the development of additional metadata fields with assistance from system administrators. Metadata information was not surveyed by the Documentation Working Group; however, metadata fields and structure must be documented for development, changes and implementation of the Documentation Portal. Recommendations and suggestions by the Documentation Working Group to transition and use the respiratory in operations are provided, including some suggested repository's organizational structure. The level of effort and schedule impacts are included where appropriate.

2.2.1 DocuShare

Xerox® DocuShare® content management system (Xerox, 2021) is the Construction Project's official document repository. It was selected during the design and development phase to meet the NSF requirement for a document management system. The Construction Project Office expects DocuShare to be the repository for official versions of management policies, plans and procedures, design documents, safety documentation, hazard analyses, released requirements and interface control documents generated from the SysML model, and project standards, guidelines and templates (this list is not intended to be exhaustive).

Three of DocuShare's advantages are handles, version control and co-location. Each object has a unique identifier called a handle, which follows the object regardless of versioning or location(s) in the directory structure (called collections). Each handle has a version history that lists all previously uploaded files for the object in question. One of those versions is designated as the preferred version, which represents the document's official, approved version and is served by the database when clicking the object's title or from a properly formatted URL shared or hosted outside of DocuShare. The object's handle does not change when/if a new version of the document is created. Lastly, objects can appear simultaneously in as many collections as are necessary and/or relevant to the document. This is facilitated by an object's "locations" property; locations are added as appropriate, and the handle automatically appears in the newly added collection. The combination of handles, versioning and co-location creates a system where each document is represented by a single record, avoiding duplication and/or version confusion.

The web interface for DocuShare is . Currently, DocuShare contains more than $30,000$ documents in more than $10,000$ collections. Creation, retention and version control of those documentation classes generally have been well managed; however, the bulk of DocuShare documents likely represents objects created ad hoc by general project staff for specific purposes. In addition, there are thousands of pieces of work product that may have archival significance but likely will not be useful for Operations.

As a tool heavily used by the project, it is recommended to continue the use of DocuShare into operations. Operations should use the DocuShare instance currently in use by the construction project with the Archive Server add-on enabled, as the continuity afforded by doing so is important and useful. Under this model, the construction project content would be moved to the archive server, and the active server would contain only operations and operations-relevant construction content following an agreed-upon directory structure. Until this transition is completed project-wide, the current location for operational documentation is Collection-602. Documentation specialists have been encouraging sub-system staff to organize objects in this collection and store operations information therein. In the few cases of feedback, this new setting has been positively received, citing the DocuShare structure for construction is difficult to understand and unintuitive for new staff, especially for operations staff.

The suggested DocuShare branch of the Storage View for operations starts with the a collection for each department. Additional collections can be created based on the project's needs as a whole (e.g., interdepartmental items such as widely used software tools). The next level of nodes is defined by the system's and/or product's department or owner. Further lower-level nodes would depend on how the department and owners want to parse the information, e.g., separation by subsystem, separation by topics such as reports, or a combination of the two. As for construction documentation, it is recommended to have a consistent organizational structure when transferring over to the archive server. The archiving process should include a determination by the department if the information is useful in operations (determining if it should be archived) and if the information should be queryable in the archive (e.g., raw data). This is especially topical for vendor documentation and deliverables.

See Section 3.3 for a detailed discussion on the reasoning, recommended actions and expected resources to implement and maintain DocuShare in operations. A more detailed analysis of arguments and considerations for this proposal's recommendation is available in DocuShare Options Trade Study for the Documentation Working Group. [Document-36788]

2.2.2 LSST the Docs (www.lsst.io)

LSST the Docs (LTD), also known by its URL "lsst.io", is a documentation hosting platform built and operated by the SQuaRE team within the Data Management group. LTD hosts *versioned static websites*, meaning any website built from HTML, CSS and JavaScript that doesn't need an active server to render content (as opposed to say Confluence, DocuShare, or Drupal websites). Static websites are a natural fit for documentation projects that originate from repositories hosted by GitHub (GitHub, Inc.), so LTD is uniquely developed to be built around versioned documentation in GitHub. *lsst.io* is one such deployment used as a hosting domain for Rubin Observatory, where all subdomains of lsst.io are independent documentation projects. The technical motivation and design of LTD are documented in SQR-006: The LSST the Docs Platform for Continuous Documentation Delivery. SQR-006 The key technical features of LTD are:

- high reliability, scaling, and security: documentation is hosted in Amazon S3 and served through the Fastly content distribution network. We don't operate any servers that receive traffic from users;
- versioned documentation; and
- flexibility to host any type of static website.

Using LTD documents provides a simple use case with additional features developed by the

SQuaRE team. The root URL for a documentation project hosts the ”default” version, which has a configurable meaning for each project (such as software release versions, temporary collaborative drafts, or an active version in DocuShare). Users can also browse other versions of the documentation through the ”/v/” dashboard pages (for example `https://www.lsst.io/v/`).

LTD hosts two types of documentation projects: *guides* and *documents*. Guides are multi-page websites convenient for user interaction and navigation (e.g., Data Management Developer Guide, T&S software guides at `https://obs-controls.lsst.io`). Documents are ”single-page” artifacts, analogous to documents that might be found in DocuShare, and they are sometimes referred to as *technical notes* (shortened to ”technote” or an appended ”TN”) — see SQR-000: The LSST DM Technical Note Publishing Platform for the motivation to create technical notes. [SQR-000] Though not required, guides are generally authored using an open-source tools using themes that is maintained by the SQuaRE team via documenteer SQuaRE Team (2022) (see `https://documenteer.lsst.io/` for corresponding guide). Besides documentation tied to specific software projects or services, guides can also collect procedures for teams, see the DM Developer Guide (`https://developer.lsst.io`) or the Observatory Operations Documentation (`https://obs-ops.lsst.io`).

DM has embraced the use of LTD by hosting a guide on lsst.io for every software project or service. Further, DM has developed most of its change-controlled documents (LDM) on its lsst.io site to take advantage of the sophisticated collaboration features that GitHub offers (for an example, see `https://ldm-151.lsst.io`). Change-controlled documents are submitted to DocuShare for archival once approved using a release process mediated through GitHub, Jira, and the relevant control board. LTD is currently hosting documents from the DMTR, LDM, LPM, LSE, and SCTR document series (note that this includes test and verification reports).

A prime example of the LTD and GitHub tools used by the project for operations activities is the T&S Commandable Service Abstraction Layer (SAL) Component (CSC) XML package and user guides, as summarized by `https://ts-xml.lsst.io`. The CSC data objects are defined in XML for SAL to consume and produce language-specific libraries that enable communication over the Data Distribution System (DDS) network. These XML files are critical for defining the configuration and interaction of the systems within the Summit Facility observing environment.

A unique example is the homepage for the LTD documentation platform, `https://www.lsst.`

io, which serves as a portal for LTD indexed documentation for searches and faceted browsing capabilities. (SQuaRE Team, 2021a) Users can search across metadata and full text (this feature is powered by the commercial service Algolia (Algolia, 2021a) in conjunction with a scraper bot built by SQuaRE) or browse through curated collections. The site itself is built with React/Gatsby.js (https://github.com/lsst-sqre/www_lsst_io), the search database is SaaS (Algolia, 2021b)), and the bot that indexes content into the search database is called Ook (https://github.com/lsst-sqre/ook). It is still in development and the current status is documented at https://www.lsst.io/about/.

As a customized set of tools that is heavily used by the project, it is recommended to continue the use of LTD and its software system into operations. The project should use documents and technotes to capture information that is static, rarely changed or serving a temporary need (e.g., proposals, documenting proof or principles, status updates); whereas, guides should be used for information that is actively updated with current information (e.g., troubleshooting guides, procedures not under change control). Neither LTD documentation type should be used for documents under change control albeit one can link to the change-controlled document within a document or guide. Further, as the LTD software system, it's additional utilities and https://www.lsst.io are designed and built by the SQuaRE team, there is considerable opportunity to leverage the expertise and experience already developed to implement a documentation portal for other platforms (see Section 3.2).

The LTD branch high-level nodes of the Storage View could be by document type (i.e., guides or documents). Following the department/owner or specific technote series (e.g., RTN, SITCOMTN) could be the next level of nodes.

2.2.3 Confluence and Jira

Confluence (Atlassian Corporation, 2021) and Jira (Atlassian, 2021) are part of the Atlassian Corperation suite of tools used by the construction and operations projects for many purposes. They are collaborative tools where teams/groups can document, share and develop information. Confluence is organized into *spaces*, each with a varying number of pages and sub-pages. Jira is organized into *projects*, each of which tracks a list of enumerated tickets or issues. Both include a large variety of features, tools and extendable add-ons to manipulate or organize the information. Many of these spaces and projects are very specialized, with some set up for personal use.

On the construction project, there are two instances of each software tool, requiring different credentials — `https://confluence.lsstcorp.org` and `https://jira.lsstcorp.org` is for the NSF MREFC effort hosted by LSSTC out of Tucson, Arizona; and, `https://confluence.slac.stanford.edu/` and `https://jira.slac.stanford.edu/` is for the DOE MIE effort for LSSTCam hosted by SLAC National Accelerator Laboratory out of Stanford, California. Rubin Observatory operations and pre-operations staff currently use the LSSTC tools.

Currently, the LSSTC Jira instance includes the following normative source of information: (1) verification elements, plans, cycles, cases, results, etc., (2) construction-related risks, opportunities and mitigations, (3) the Failure Reporting Analysis and Corrective Action System (FRACAS) for failures and corrective actions, and (4) hazard mitigation verification. However, it is required operations include risks, opportunities and mitigations in the NOIRLab managed risk management software tool, Alcea Tracking Solutions software `https://noirlab.alceatech.com/saml2/sso`. The project must determine if the normative source for information on risk, opportunities and mitigations will continue to be Jira or moved to the NOIRLab risk management tool. If Jira is the normative source of information for these items, the project must develop a method to sync information to the NOIRLab tool.

As tools that are heavily used by the project, it is recommended to continue the use of Confluence and Jira into operations. With a wiki like Confluence which that integrates well with Jira, the two are especially convenient for rapidly developing new ideas, taking and storing meeting minutes, collecting information in interactive tables, and drafting outlines for future documentation, all while tracking the tasks and being able to actively report on status. Other powerful features include simultaneous editing, native sharing options internal to Confluence/Jira or external such as email, etc. As operational-based information takes form, departments and technical groups should consider the spaces and projects to prevent having a large number of unused or old areas, as seen with the current Confluence and Jira instances. In the case of Confluence, project staff must be fastidious in moving content needed on a long-term basis into other official storage locations at the earliest appropriate stage (e.g., DocuShare, lsst.io). Users must understand the limitations of information and provided guidance as to how and when information should be moved from Confluence or Jira to another storage location.

The Confluence branch of the Storage View would begin with the spaces. Most spaces have a relatively small number of top-level pages which are tailored to a specific need (e.g., separation of subsystems), and these top-level pages can naturally be the next level of nodes. The Jira branch would begin with the projects, but the next level nodes are not as apparent and should be determined by the owner or responsible group. There are many ways to create the next level nodes for Jira; for example, most projects have a natural breakdown of structure used to organize relationships between tickets (e.g., epics). Lower-level nodes will depend on use; they could include items such as meeting notes or customized dashboards. The NOIRLab risk tool should also have a branch in the Storage View.

2.2.4 Engineering models in Solidworks Product Data Management (PDM)

Solidworks Product Data Management (PDM) Professional (Solidworks, 2021) is the official computer-aided design (CAD) model repository for the Telescope and Site construction group, including vendor documentation and deliverables. It uses a check-out / check-in system to allow configuration management of the design. Each check-in produces a new version of the part or assembly. Earlier versions can be accessed if needed to compare designs or revert to an earlier design. A workflow feature allows the designs to go through a review process until the design is approved and locked from further changes. A revision process is also included in the workflow to allow for changes to the designs if needed after final approval. The software allows for vault replication at multiple sites and the project currently has a server operating in Tucson, Arizona and Chile to support CAD users at multiple sites. Solidworks PDM is the normative source of information for system decomposition.

The current configuration of the PDM vault contains top-level access to baseline design data and interface control documents (ICDs) (drawings) along with as-designed vendor subsystems. This includes the original baseline design, early designs and further development throughout construction. In addition, a series of design and drafting standards is also stored in the PDM system. Two servers are hosted: the main server located in Tucson, Arizona, and a replicated server in La Serena, Chile to allow faster file access.

In operations, as-built design information will be needed to help with logistics planning for maintenance and future design upgrades. No information has been deleted or archived at this time. The Documentation Working Group recommends the vault is reorganize so that legacy data is archived for access but is not easily mistaken for as-built design data. This will require a full-time resource for 3 to 6 months. The specific structure for the branch of the Storage View is highly dependent on this reorganization. For example, the high-level nodes could be by subsystem or physical location, lifecycle of particular drawings (e.g., as-designed, as-built), or use within specific states of the telescope (e.g., on-sky imaging, preventive maintenance shutdown).

See Section 3.4 for a detailed discussion on the reasoning, recommended actions and expected resources to implement and maintain Solidworks PDM in operations.

2.2.5 MagicDraw

MagicDraw (No Magic, Inc.) is a tool used to maintain a model of the Rubin Observatory system by creating a relational database between system elements. There are a number of elements capture in the MagicDraw database defined as the normative source of information, with the information synced and/or exported to other repositories. As the normative source of information, it will be up-to-date when given over from construction and continue active use within operations. Details on how Rubin Observatory uses MagicDraw are included in the user guides collected on the following Confluence page: `https://confluence.lsstcorp.org/display/SYSENG/MagicDraw+LSST+Users+Guide`.

- **Hazard Analysis** — normative source — synced to Jira for verification tasks.
- **Failure Mode and Effects Analysis (FMEA)** — normative source — known to be incomplete.
- **Requirements** — normative source — exports to DocuShare.
- **Verification Elements** — normative source — synced to/from Jira.
- **Verification Plans/Cycles/Cases** — synced to/from Jira via Syndeia™ (Intercax, LLC). See Section 2.2.3 for normative source).
- **SAL Commands, Events, Telemetry** — imported from CSC XML.
- **Operations Concepts** — source of truth.
- **System-level State Machine** — source of truth.
- **Interlocks** — modeled from source material.
- **Structural Decomposition** — will be synced from Solidworks in the future.

It is recommended to continue the use of MagicDraw into operations. The user guides should be relocated into a more appropriate storage location, such as lsst.io. The content in MagicDraw should be reviewed to create the tree for the Storage View that can readily reflect normative sources of information and how/what references or depends on this information. The effort required to complete the FMEA information will depend on the level of completeness when handed over to operations. Effort to sync the structural decomposition with Solidworks will be evaluated when the system information is more complete.

2.2.6 Euporie

Euporie is a network drive on a server managed by the construction project; and, it is accessible with Rubin Observatory credentials through VPN at `smb://euporie`. The drive contains a number of personal directories and a directory named ”TS-Deliverables” managed by the Telescope and Site group. Stored in subdirectories of TS-Deliverables are vendor-supplied documentation as contract deliverables (design documentation, construction drawings, manuals and other miscellaneous information), with each subdirectory.

It is recommended to discontinue use of Euporie for operations. A review of the content of personal and shared directories is recommended to determine what information should be relocated on a case-by-case basis. Additionally, a determination should be made if information should be archived as construction-related or moved to a more active storage location for operations staff access. After this is complete and access will continue for purpose unrelated to operations, users must understand the limitations of information (namely for long-term operational activities) and provided guidance as to how and when information should be moved from Euporie to another storage location.

2.2.7 GitHub

GitHub™ (GitHub, Inc.) is used by the project for software and documentation collaboration, storage, version control via git™ (Software Freedom Conservancy, Inc.) and deployment. GitHub is primarily utilized via *repositories*, or repos. Repositories are owned by individual users or an *organization*, and organizations can include *teams* for additional granularity.

The Rubin Observatory construction and operations projects use a large number of GitHub repositories, organizations and teams, primarily to ease access control. The main project GitHub organization is `https://github.com/lsst`; however, not all operations software is found in this organization. On the construction project, each subsystem on the construction project generally has its own GitHub organization, though in some cases subsystems have additional GitHub organizations for specific teams or projects. For example, the Science Platform software is contained in https://github.com/lsst-sqre, Telescope & Site software in https://github.com/lsst-ts, EPO software in https://github.com/lsst-epo and Camera software in several organizations.

As a heavily used tool by the project, it is recommended to continue the use of GitHub via git into operations. Effort by construction, pre-operations and operations teams already implemented infrastructure for utilization and the GitHub collaboration structure, including basic structure for operations. There are a large number of repositories containing the normative source of information, and the project should ensure these repositories are clearly indicated as such and what respective information is the normative source.

2.2.8 Drupal

Drupal (Drupal Association) is an open source web content management framework used by the project for its websites' back-end, including https://www.lsst.org, https://project.lsst.org and sites created to facilitate project meetings and reviews. To avoid the complication of having to make multiple site updates when content changes, these documents of each of these sites are served by hyperlinks that pull files from whatever repositories contain their normative sources of information. While it is possible to upload discrete files to a specific site's server location, by policy and standard the project eschews doing so.

The major exception is regarding project-level meetings and reviews sites, such as the Project and Community Workshop (PCW) or agency reviews. Their site directories contain document and presentation files in order to preserve the content presented at the time. At the conclusion of the event, these files are uploaded to DocuShare in a collection specific to the event. Documents such as policies, requirements, and design documents are uploaded as a ZIP file to preserve a snap-shot while preventing replication and multiple handles that may cause confusion with the normative source of information. The method described for project-level meetings and reviews may not be implemented by all subsystems on the construction project, and the project must consider if and how to archive this information.

The project must consider if and how its appropriate to continue using Drupal for web content management into operations. If used in operations, it must be clear how and where normative sources of information are used (e.g., DocuShare). It must be determined the level of change control appropriate for websites, as well as files in site directories or hyperlinked files without change control.

2.2.9 LSSTCam information

There are multiple storage locations for information, documents and data produced during the MIE effort for LSSTCam:

2.2.10 Engineering Facility Database (EFD)

2.2.11 Primavera P6

2.2.12 Verification reports

2.2.13 CMMS

2.2.14 NOIRLab Risk Tool

2.3 Access View

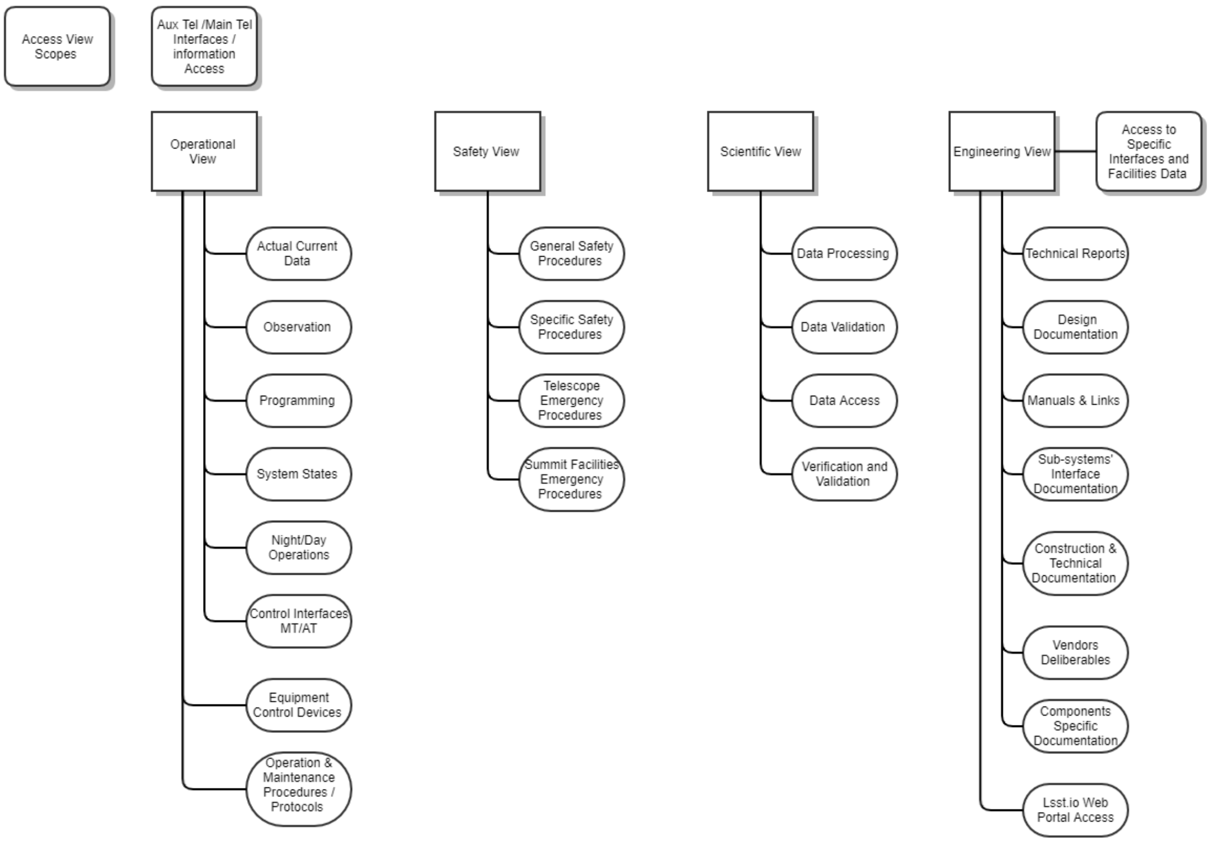
Access view is to categorize information into trees serving role-based needs to stakeholders. It's purpose is to assist users in retrieving and discovering documentation and information applicable to their use cases. As such, the access trees will be highly tailored to use case and role-based needs. In general, the information is distributed across multiple repositories, and the Access View can be agnostic to that.

The proposal includes an example with the following root-nodes: Operational View, Safety View, Scientific View, and Engineering View. These are explained in the following subsections and Figure 3. Project should consider use cases and desirable, e.g., Documentation View, Day-time and Night-time Views, Safety and Security View, Outreach View.

2.3.1 Operational View

Operational View helps with interfaces of real-time data and visualization, observations and its programming, and system state. It may be natural to separate via leaf-nodes the Main Telescope from the AuxTel to capture specific differences such as control interfaces. Another natural break down could be day- and night-time operations. Groups should take advantage of commonalities and referential nature of the Documentation Views. Users include use of

Figure 3: Temporary Caption.



equipment control devices, procedures and protocols for operation and maintenance, such as control room staff, viewing assistants.

2.3.2 Safety View

The Safety View focuses on accessing the different information related directly to Safety Procedures and Protocols that apply to the different systems. This can help add consistency and appropriate requirements as necessary. This Access Would lead users to General Safety procedures, Specific Safety Procedures, Emergency Procedures of telescope and summit facilities. These Collections shall be available to cover the information needed during activities authorized by the safety and security team

Users include safety, engineering, security, maintenance staffs and management.

2.3.3 Scientific View

The Scientific View aims at access to scientific platforms and interfaces that correspond to Rubin Observatory science objectives.

2.3.4 Engineering View

The purpose of the Engineering View is access to Rubin Observatory technical information, including:

- Accessing specific interfaces and facilities data
- Sub-systems' Documentation and components to implement an activity The Linking access structure would lead users to:
- Technical Reports
- Design Documentation
- Manuals and links to other components documentation
- Sub-systems' Interface Documentation
- Construction Documentation. Technical Documentation for the software is separated from hardware Documentation. Technical Documentation within Engineering view:
- Subsystem Documentation Interfaces
- Manuals
- Vendors deliverables
- Specific documentation by component

There are many way and justifications on how to break this view into leaf-nodes. As the Documentation Working Group developed this, many people readily used a system decomposition as one form. Other found it potentially useful to break the tree down to specific activity types, e.g., Validation/Verification, Failure Evaluation, Maintenance, Engineering Control, Engineering Design. Combining these example breakdowns can also be considered. With such sprawling associated subjects, it's more crucial to develop a strategy to break these leaf-nodes down further by considering categories, impacts to users and how they could want to find associated information. Note that the Topic View should also be considered when performing this development exercise.

2.4 Topic View

Describe here the properties of the Topic View and provide diagrams showing the proposed structure for this view.

3 Transition and implementation planning in operations

the need of defining the workflow of the documentation and data produced leads to create an official group of determined actions to officially save the operational documentation, classify it, separate it by the specific Department, System, Subsystem, facility, type, and application.

Add here an overview of the proposed implementation plan including the following specific areas of implementation:

- Primary Sources
- Documentation Portal
- DocuShare
- PDM Works
- Resources and Responsibilities
- Schedule
- Transition Workflow
- Special Topics (e.g., Bilingual)

3.1 Primary Documentation Sources

Describe here the proposed location of the primary documentation sources. These are a subset of the identified source listed in SITCOMTN-012.

Need to take information from Storage View subsections, then find gaps to confirm add with DWG or identify potential gaps for project/departments/groups to consider.

3.2 The Documentation Portal

The Documentation Working Group recommends the creation of a *documentation portal* web application as a means of making documentation content discoverable and accessible to Rubin Observatory staff. This documentation portal will provide interfaces for both searching (based on content and metadata) and browsing (based on hierarchical categorization) of documentation resources. Documentation does not reside within the portal itself. Rather, the portal's objective is to efficiently link the user to the document, where it is stored in any of the observatory's adopted storage platforms (Section 3.1). The new Documentation Portal discussed herein is based upon the website ()lsst.io-cite which provides a search and browsing interface for the Rubin Observatory's public-facing technical documentation operated by the SQuaRE (Data Management) team. (A detailed description of , LSST the Docs (LTD) and applicable software is available in [chair].) The new portal will be accessible only those with Rubin Observatory staff credentials, and will be purpose-built for observatory and survey operations. This section describes the design principles, technical architecture, security model, and cost estimate of the Rubin Observatory Documentation Portal.

3.2.1 Design requirements

The design requirements of the Documentation Portal reflect the recommendations made by the Documentation Working Group elsewhere in this report:

1. The role of the Documentation Portal is to link to documentation resources. The portal itself does not host the content itself, nor provide user interfaces for creating and maintaining new versions of documentation content. This requirement reflects the Documentation Working Group's recommendation that documentation content should be hosted on a select set of platforms that are idiomatic for the content and the teams that

work with that content (Section 3.1).

2. The Documentation Portal must provide equal support for content stored in any of the storage platforms (Section 2.2).

3. The Documentation Portal must be capable of supporting several hierarchical browsing schemes for accessing content based on different organizational views of the documentation (Section 2).

4. The Documentation Portal should automatically update and sort documentation content, to the greatest extent possible. In other words, curators should only need to maintain documentation in the document's storage platform, without any administrative action through the Documentation Portal's user interface or infrastructure. A consequence of this requirement is that the Documentation Portal should not persist information about a document that is not available from the document's own storage platform.

5. The Documentation Portal should be secured so that it is only accessible to users with Rubin Observatory staff credentials.

6. The Documentation Portal should not maintain fine-grained access control for specific documents or categories of documents. For secured documents, the portal relies upon the security mechanisms of the document's own storage platform. The portal should also reduce its metadata storage of confidential documents to ensure that content cannot be inferred from a search, for example.

3.2.2 Technical architecture

The architecture described here is based upon that which is already put into production with the portal for public-facing documentation. Starting from this working archetype relieves a great deal of technical risk and development from the new portal's implementation. Both portals share the use of Algolia (Algolia, 2021a) as a search back-end and Ook (SQuaRE Team, 2021b) as a content indexing service. The Documentation Portal will have an independent front-end to support the specific Access Views recommended by the Documentation Working Group (Section 2.3). The Documentation Portal will also use a separate instance of the Algolia database to eliminate any risks associated with leaking internal documentation to the public-facing portal.

Figure 8 depicts the components of the Documentation Portal.

3.2.2.1 Algolia The core function of the portal is to enable access to documentation through browsing and search. To implement this, the portal needs a back-end service that contains metadata about Rubin Observatory's documentation holdings and provides interfaces to access and query that metadata from the front-end (i.e., the website). This search database and interface could be made for "free" with entirely open-source components such as Elasticsearch (Elasticsearch B.V., 2022) and in-house web service. However, a search database and service are sufficiently generic that we cannot add value by making it in-house, and in fact developing, tuning, and operating this service, would costly in terms of labor. For the public documentation portal, we opted to use Algolia and recommend that we make the same choice for the new Documentation Portal.

Although is currently operating on a free open-source license of Algolia, the new internal Documentation Portal would be an entirely paid license. Algolia prices based on record counts and request rates. In operating , we found record count to be the limiting factor. At the moment, $1,000$ records costs $1 per month. $1,000,000$ records would cost $850 per month with volume discounts. For reference, the service currently uses $110,000$ records to host all technical notes and change-controlled documents with drafts hosted on LSST the Docs (LTD), also known as lsst.io.

Note that a single document is composed of potentially as many records in Algolia. To optimize full-text search, we break a document into smaller records, generally across section boundaries. Our current algorithms generally produce small record, so it is possible to reduce costs by tuning how we segment content into Algolia records. Furthermore, each sorting option requires a separate pre-sort index. Sorting documents by date, and document number, in addition to relevance, consumes three times as many records as only sorting by relevance.

3.2.2.2 Ook The Ook service is responsible for continuously indexing content into Algolia. (SQuaRE Team, 2021b) Whereas we chose Algolia to provide a turn-key search database service, for we chose to build the indexing service in-house to have complete control over how documents are indexed, and what metadata is associated with each document. For example, LaTeX-based documents are indexed based on metadata exported from the Lander PDF landing page generator (also developed in-house), which in turn parses LaTeX syntax in the document source to access metadata such as titles, authors, and so on. The configurability of Ook is beneficial to indexing other types of highly specialized documentation.

The key design principle of Ook is that metadata is extracted from the document as it appears in its repository, rather than requiring direct human interaction with Ook or Algolia to curate the data. This allows Ook to scale well across an organization as large and varied as Rubin because individual teams manage documents as they already do in the repositories they are already familiar with.

Ook is built such that new content types can be added by writing additional Python-based workflows for each content type. Ook itself provides utilities for queuing ingests, converting content and formatting data for Algolia, and working with the Algolia service itself.

Ook indexing operations can be triggered several different ways. For example, the lsst.io service publishes messages to a Kafka cluster whenever documentation is published on that platform; Ook subscribes to those messages and queues indexing workflows. Ook indexing operations can also be scheduled through an HTTP API. Generally, the goal is to trigger indexing operations automatically whenever the source material changes.

The existing Ook indexing workflows work by downloading content from websites and web services (HTTP APIs). If content is not easily accessible, it would be possible to develop an alternative workflow, such as submitting copies of the document directly to Ook for indexing. Some document repositories may offer online access but have hard-to-use APIs, DocuShare being a prime example. These difficulties can be worked around, for example by emulating a web browser to download content and metadata, but at the cost of more fragile indexing workflows.

Ook is currently operated as a Kubernetes application in the Google Cloud. This arrangement is ideal for minimizing operations cost, and providing convenient scaling.

3.2.2.3 Front-end application The front-end web application is how users (Rubin Observatory staff) find documentation. The web application does not provide the document itself—instead, the web application provides a search result card that the user can click on to access the document in its original repository. The Algolia services provides all browsing and search functionality; the front-end application provides the user interface over top of Algolia.

In addition to providing a link to the original document, the portal can also provide an immediate view of a document's metadata. Although the front-end application can show a basic view of a document's metadata based on metadata common to all records, the website can be developed to show additional metadata for different types of documents.

For , we built the site as a React JavaScript application. This allowed us to use and customize the pre-made widgets provided by Algolia for building the user interface.

The front-end application will be accessible only to users who log in. The simplest way to approach this is by putting the application behind a VPN so that the application is completely separate from security concerns. Another approach would be to place the application behind an OAuth proxy to provide a slightly better user experience.

3.2.3 Support for multiple views

In Section 2, the Documentation Working Group outlines several views for hierarchically arranging documents trees. These views correspond to navigational structures in the front-end application. Although the front-end application code is generally "aware" of the different trees, individual documents are placed in the tree on the basis of metadata in their Algolia records, so that the Algolia service can pre-sort and filter documents into the trees. Since Ook supplies metadata to Algolia, and Ook in turn leverages metadata native to the document and the document's repository, the responsibility for curating documents into different views is the responsibility of individuals managing documents in each repository.

3.2.4 Information security

Documentation has multiple types of security concerns, such as control over who and how documents are updated, control over who can access documents, and ensuring the long term integrity and preservation of information. Since the Documentation Portal is not the canonical repository for any documentation, the portal is not involved in controlling document updates and preservation. The portal's key security concern is access control.

The portal is designed to only provide authentication-based access control. Any Rubin Observatory staff member with credentials can access any metadata records contained within the Documentation Portal. Once a user selects a document to view, they are forwarded to that documentation repository and must authenticate with that repository and be subject to its access control rules.

However, the metadata contained in the Documentation Portal can be potentially rich, even including the full-text content of a document to enable search functionality. It is conceivable that some of this metadata may not be appropriate for observatory-wide access (such as information with export controls). In these cases, the most realistic approach to preserving strict access controls in these situations is to limit what metadata is available. In order of strictness, the following approaches can be used:

1. Omit full-text content of a document from Algolia records.
2. Limit or obfuscate other metadata (such as titles) in the Algolia records.
3. Omit the individual documents altogether from Algolia and instead link to a documentation landing page hosted by the secure document repository itself.

In all cases, controlling how documents are indexed is done by configuring the indexing service, Ook.

3.2.5 Effort estimate

The key tasks for implementing this plan are:

1. Building additional content indexing workflows into Ook and updating the metadata schema to accommodate all use cases.
2. Build the Documentation Portal web application.
3. Curate documents so that they are uniformly present in their document repositories and have any metadata that is expected by the Ook indexing workflows. This can be distributed to teams working in each document repository.
4. Support for continual maintenance and additional features (e.g., new content types).

To provide a sense of the scale of the first two tasks, the implementation of the portal is a useful reference. Building the first version of Ook, which indexes LaTeX documents and Sphinx-based technical notes in lsst.io required four weeks of work. Building the front-end website required six weeks.

Since that original implementation, the SQuaRE team has gained more experience building React web applications and working with Algolia, so it is reasonable to expect less than four weeks of work to build out the internal web application. Although Ook can be expanded as-is, the key uncertainly is in the number and complexity of the additional indexing pipelines that need to be built for additional document types and repositories. Generally, though, an internal Documentation Portal can likely be stood up within one to three months, with a potential long tail of effort to continue to add support for additional content types.

3.3 Path Forward with DocuShare

As part of the proposal, the Documentation Working Group considered and evaluated options for Xerox® DocuShare® (Xerox, 2021) use within operations. It is expected it will continued to be used in operations. It is recommended that Rubin Observatory Operations should use the DocuShare instance currently in use by the construction project with the Archive Server add-on enabled, as the continuity afforded by doing so is important and useful. DocuShare's version control, version history, permissions and co-location capabilities are valuable tools for effective document management and should be maintained. Additionally, continuing DocuShare use will ease the transition from construction to operations by providing a single shared platform, thereby avoiding compatibility issues, eliminating the need to migrate documents, preventing loss of documentation and avoiding confusion by retaining legacy handles and references. Enabling the Archive Server will allow operations staff to structure and manage DocuShare according to its needs while retaining access to all documentation accumulated during construction. The Documentation Working Group believes this recommendation's benefits outweighs any required new and/or continuing costs and labor resources for setup and ongoing maintenance.

The recommendation is consistent with NOIRLab document management, which has selected DocuShare as the lab's document repository. By also using DocuShare, the repository for Rubin Observatory Operations will be compatible with the knowledge and skills held by the business unit responsible for NOIRLab document management. While it's possible to use NOIRLab's DocuShare, the Documentation Working Group believes that to be less advantageous than retaining the construction project DocuShare instance: switching to NOIRLab's DocuShare would result in loss of legacy document handles and would require labor-intensive and time-consuming document migration. Continuing to use the current DocuShare will retain document version histories and allow persistence of document handles for foundation documents, many of which are known within the project and community as much by their handles as they are by their titles. Lastly, each instance of DocuShare has a $2,000,000$ document limit. If the construction project DocuShare instance is retained, the entire document limit is available to Rubin Observatory. If NOIRLab's DocuShare instance is used, Rubin would have to share the document limit with other lab units and programs.

However, the Documentation Working Group recognizes that simply continuing with the current DocuShare instance as-is fails to address long-standing issues of clutter, inadequate metadata, and poor classification; therefore, we recommend enabling the Archive Server add-on. The add-on provides a second server into which documents whose lifecycles have ended can be sequestered, freeing the active server to be structured and managed according to best practices and needs for operational activities. While archived documents no longer reside in the active server and are not represented in its directory structure, they are viewable and searchable and can be restored to the active server if necessary. The two servers are connected and can be interacted with using a single user interface. Under this model, the construction project content would be moved to the archive server, and the active server would contain only operations and operations-relevant construction content following an agreed-upon directory structure.

Further, to mitigate the risk of replicating the clutter and less-than-optimal classification of construction-era DocuShare, the working group recommends Rubin Operations adopt a more formal workflow for how and by whom DocuShare content is created, formatted, and updated. This workflow would involve something similar to NOIRLab's steward concept. NOIRLab's document management plan expects each business unit or program to designate a person or persons to lead the group's documentation activities in DocuShare. These stewards are to be highly trained so they can perform DocuShare functions, provide guidance to others, enforce standards, and correct errors. The idea is to have fewer but better-trained users with full DocuShare permissions and functionality. While this will result in a less distributed workload, consistency and quality will improve, leading to greater tool utility. The Operations departments should follow a similar approach, and as much as possible, the workflow should occur within DocuShare.

Retaining the current DocuShare instance and enabling the Archive Server add-on requires both new and continuing costs and labor. The calendar year 2020 renewal cost for the current DocuShare instance was $22,000, and Rubin's IT resources are expended to maintain the service, which is hosted on a physical server in Tucson, Arizona. The archive server add-on carries a one-time cost of $7,500 and an additional $1,400 per year for support, according to a December 2020 quote from Xerox® . Additionally, IT resources will be needed to configure the add-on, and some training likely will be required. Despite these considerations, the Documentation Working Group believes the benefits outweigh the costs, and the required resources are less burdensome than those that would be needed to adopt another repository.

Further information on current DocuShare use is detailed in [chair]. A more detailed analysis of arguments and considerations for this proposal's recommendation is available in DocuShare Options Trade Study for the Documentation Working Group. [Document-36788]

3.4 Path Forward with Solidworks Product Data Management (PDM)

As part of the proposal, the Documentation Working Group considered and evaluated options for Solidworks Product Data Management (PDM) Professional (Solidworks, 2021) use in operations. It is expected it will continued to be used in operations as the official CAD model repository, as it is currently is in construction for the Telescope & Site subsystem. The utility provided by the configuration management and modeling software not only provides the needed historical reference, including the baseline and as-built designs, but the system and construction project files will be needed for design changes and upgrades throughout the lifecycle of the Rubin Observatory.

Since the observatory design and construction has yet to be completed, an overall CAD model of the observatory is not available at this time. The Documentation Working Group recommends that once commissioning is complete, an overall CAD model of the observatory is produced. As the internal systems of the observatory are complex, this initial iteration should only represent high-level systems and interactions at the beginning of operations to permit operations staff time to understand how the systems interact in a mechanical phase space. Although, a fully detailed CAD model should be created as soon as practicable have a reliable, version-controlled reference throughout the design lifecycles taking place throughout operations. This effort will require a full-time resource with a high-level of expertise in CAD modeling with the Solidworks software for 6 months .

The PDM system uses a data card feature that allows searchable metadata to be associated with each file in the vault. Vendor files not created using Solidworks do not contain the metadata used to populate these data cards. Since converted CAD models from vendors have blank data cards, it is recommended that the final as-built CAD models imported to Solidworks file format have the metadata entered to allow easy search access. This will require a full-time resource for an estimated time of 3-6 months.

A workflow process is currently in place for all files in the PDM system. This workflow allowed the baseline design models and ICDs to be checked by the design engineering group and approved by the systems engineering group. Once approved for release, files are locked to avoid changes. A workflow revision process is in place to allow for changes after the release. It's recommended that the both workflow processes be reviewed by the operations teams so they may be modified as needed to work with systems that the operations team will use throughout the life of the observatory.

The PDM system and associated design software must be administered. The project should continue hosting servers in both locations, Tuscon and La Serena. The project IT group currently administers the servers and performs software installation and upgrades when needed on users' computers. It's recommended to have a CAD and PDM vault administrator for managing the system, implementing periodic changes and assisting user's with upgrades. This position will be needed on a part-time basis over the life of the observatory. Management must consider if the role should be filled by internal staff or the software value-added reseller (VAR) offers contract services.

The PDM system uses a software system that uses a set of licenses to allow users to access the PDM vault. The construction team currently uses 12 CAD editor licenses (allows changes access to files) and 5 viewer licenses. Each license has a yearly maintenance fee that allows for annual upgrades as well as access to service packs for the software during the current year and technical support for the software. It is recommended that the operations team determine how many licenses will be needed during operations to accurately estimate the annual budget and allow access with editor or viewer use to staff that require it.

3.5 Required Resources & Subsystem Responsibilities

Description of the estimated resources need to carry out this implementation plan and what the roles and responsibilities are for each subsystem in this context.

Need to take information from Storage View subsections, then find gaps to confirm add with DWG or identify potential gaps for project/departments/groups to consider.

The departments and technical groups are responsible; documentation specialists primary assistants; and both depend on system administrators and software developers on case-by-case need. Transition Joint Work.

1. Documentation Quantification
2. Documentation Selection
3. Proposed repositories Reviewing
4. Starting and Finishing Scheduling

Recommend new Operations Documentation Committee/Group for project-wide support, common tools and distributed development of them, lessons learned, etc. Project-provided charge and scope. Composition: documentation lead(s), documentation specialists, SQuaRE seat, ex-officio manager (department or subgroup?) Need Chuck, Austin, Leanne input to min (and possibly max) bound level of effort for this role.

3.6 schedule

Outline the schedule needed for implementation to meet the objective delivering a coherent tachnical document packed at the end of the Project Construction effort.

Need to take information from Storage View subsections, then find gaps to confirm add with DWG or identify potential gaps for project/departments/groups to consider.

4 Transition Plan and Workflow

Describe the process for transition from the current state to the future state.

5 Risk Assessment

Describe the risk exposure if various part of the whole of this implementation is not conducted.

Once rest of report is in good draft form, meet with Chuck after preliminary outline is available. Some information in other portions of report. IF, THEN statements needed?

Create at least one project managed risk; confirm if should remain the worst offender or some sum-of-parts roll-up. Are non-project managed risks managed by someone else, or just historical justification background?

A Specific department and group information

The Documentation Working Group collected a large amount of information specific to departments or technical groups. This section collects that information for consideration during future implementation.

A.1 Summit facility decomposition examples

This section includes potential ways of capturing the decomposition of facilities, subsystems and components located on the summit.

A.1.1 Decomposition of Telescope Pier and Enclosure facility

This list is one way to decompose the Telescope Pier and Enclosure facility by subsystems for the Product View tree.

- Dome
  - Dome Enclosure & Mechanical Structure
  - Dome Rotator and Rotation Enclosure
  - Azimuth Rotation
  - Aperture Shutters
  - Thermal Ventilation
  - Heat Control
  - Light Wind Screens
  - Read Access Door
  - Louvers & Lift Baffling
  - Bridge Camera
  - Access to External Surface
  - Done Control System & Electrical System
- Calibration System
  - Bean Projector
  - Laser System
  - White Light Sources

  - Fiber Spectrographs
  - Electrometer
  - Calibration Screen
- Dome Bridge Crane
  - Temporary Crane
  - Permanent Crane
  - Crane Control System & Electrical System
  - Crane Hooks & Specifications
- Telescope Mount Assembly (TMA)
  - Camera Cable Wrap
  - Oil Bearing System
  - TMA Magnets
  - TMA Control System & Electrical System
- Camera System
  - ComCam
  - LSSTCam
    - Camera Cart
    - Camera Body Assembly
      - Mechanical Structure
      - Filter Changer
      - L1 Lens
      - L2 Lens
      - L3 Lens
      - Cryostat
      - Utility Trunk
    - Camera Hexapod & Rotator
    - Integration Structure
- Hexapod & Rotator
  - Camera Hexapod & Rotator
    - Hexapod Rotator
    - Hexapod Actuators
  - M2 Hexapod

        * Hexapod Rotator
        * Hexapod Actuators
    - Hexapod & Rotator Control System & Electrical System
    - Mechanical Structure
- Mirror System
    - M1/M3 System & Cell Assembly
        * M1/M3 Cell
        * M1/M3 Glass & Mirror
        * M1/M3 Support (Axial, Passive and Active Systems)
            · Static Supports
            · Hard Points
            · Pneumatic Actuators
            · Independent Measuring System (IMS)
            · Vibration Measuring System
            · Laser Tracker
        * M1/M3 Thermal Control System
        * M1/M3 Control Systems & Electrical System
        * M1/M3 Maintenance Cart
    - M2 System & Cell Assembly
        * M2 Cell
        * M2 Glass & Mirror
        * M2 Support
            · Tangent Links
            · Force Actuators & Elecromechanical Driving Motors
            · Independent Measuring System (IMS)
            · Vibration Measuring System
            · Cell Temperature Sensors
        * M1/M3 Thermal Control System
        * M1/M3 Control Systems & Electrical System
        * M2 Maintenance Cart
    - Thermal System & Facilities
        * Coolant Recirculation Systems
        * Fan Coil Units
        * Thermocouple Monitoring System
        * Air Nozzles

    * Thermal System Control Systems & Electrical System

- Coating Plant Facility
  - Upper Chamber
    * Sputtering Coating System
    * Chamber System
    * Cooling System
    * Equipment Control System & Electrical System
  - Lower Chamber
    * Upper & Lower Vessel Vacuum Pump
    * Equipment Control System & Electrical System
    * Mirror Cart
    * Overhead Bridge Crane
  - Coating Chamber
    * Coater SKYVA 8.4
      · Vacuum Systems
      · Water System
      · Lifting Unit
      · Pneumatic System
      · Process Gas Supply & Control System
      · Upper Vessel Support
      · Chamber CPL
      · Main Drive
      · Lower Vessel Cart
      · Installation Chamber Illumination System
    * Ancillary Equipment
      · Assembly and Handling Tooling
      · Component Fixtures
      · Measurement Equipment
      · Test Assembly
      · Platforms & Framework Bridge
      · Ladders & Pedestals
    * Coater Control Systems & Electrical Systems
    * Magnetron
    * Washing Station
    * Electrical Equipment Cleaning Station

A.1.2 Decomposition of Service Building facility

This list is one way to decompose the Service Building facility by location for the Product View tree.

- Utility Ground Floor (Level 1)
  - Mechanical Room
  - Oil Supply System Room
  - Machine Shop
  - Camera Refrigerant Room
  - Electrical Room
- Control Floor (Level 2)
  - Office Areas & Conference Room
  - Control Room
  - Computer Room
    - Racks
    - Uninterruptible Power Sources
    - Server Computers
  - Break Room
- Maintenance Floor (Level 3)
  - Receiving Area & Truck Ramp
  - Mirror Cart Rails & Mirror Maintenance Area
  - Camera Maintenance Area & Cleanroom
  - Platform Lift
- Mezzanine Floor (Level 4)
  - Staging Platform
  - Elevated Catwalk
  - Camera HVAC
- Enclosure Ground Floor (Level 5)
  - Pier Base
  - Optics Room

- Machine Shop
- Lower Enclosure Base
- General Utility Room
- Platform Lift
- Electronics Laboratory

A.1.3 Decomposition of Auxiliary Telescope facilities

Figures 11 and 12 are one way to decompose the Auxiliary Telescope facilities by location for the Product View tree.

B References

Algolia, 2021a, The flexible ai-powered search & discovery platform, URL `https://www.algolia.com/`,
Accessed: 2021-06-06

Algolia, 2021b, SaaS: Secure & reliable search that scales with your business, URL `https://www.algolia.com/solutions/saas/`,
Accessed: 2021-06-06

Atlassian, 2021, Jira: Issue & project tracking software, URL `https://www.atlassian.com/software/jira`,
Accessed: 2021-06-06

Atlassian Corporation, 2021, Confluence: Your remote-friendly team workspace, URL `https://www.atlassian.com/software/confluence`,
Accessed: 2021-06-06

[LSE-489], B. Blum, Ž. Ivezić, S. Kahn, V. Krabbendam, 2020, *Charge to the Project-Wide Documentation Working Group*, LSE-489, URL `https://LSE-489.lsst.io`

[SITCOMTN-012], (chair), C.C., (co Chair), D.C., (co chair), R.M., et al., 2021, *Rubin Observatory Construction Documentation Inventory*, SITCOMTN-012, URL `https://sitcomtn-012.lsst.io/`,
Vera C. Rubin Observatory Commissioning Technical Note

[SITCOMTN-005], Claver, C., Bauer, A., Bechtol, K., et al., 2021, *Construction Completeness and Operations Readiness Criteria*, SITCOMTN-005, URL `https://sitcomtn-005.lsst.io/`,
Vera C. Rubin Observatory Commissioning Technical Note

Drupal Association, Open source content management tools, URL `https://www.drupal.org/`,
Accessed: 2022-11-10

Elasticsearch B.V., 2022, Free and open search: The creators of elasticsearch, elk & kibana □ elastic, URL `https://www.elastic.co/`,
Accessed: 2022-10-10

GitHub, Inc., Where the world builds software, URL `https://www.github.org/`,
Accessed: 2021-06-06

Intercax, LLC, The digital thread platform for model-based engineering, URL https://intercax.com/products/syndeia/,
Accessed: 2022-11-10

[Document-36788], McKercher, R., 2020, *DocuShare Options Trade Study*, Document-36788, URL https://docushare.lsst.org/docushare/dsweb/Get/Document-36788

No Magic, Inc., Magicdraw, URL https://www.3ds.com/products-services/catia/products/no-magic/magicdraw/,
Accessed: 2021-06-06

[SQR-000], Sick, J., 2015, *The LSST DM Technical Note Publishing Platform*, SQR-000, URL https://sqr-000.lsst.io/,
Vera C. Rubin Observatory SQuaRE Technical Note

[SQR-006], Sick, J., 2016, *The LSST the Docs Platform for Continuous Documentation Delivery*, SQR-006, URL https://sqr-006.lsst.io/,
Vera C. Rubin Observatory SQuaRE Technical Note

Software Freedom Conservancy, Inc., Open source distributed version control system, URL https://git-scm.com/,
Accessed: 2022-11-10

Solidworks, 2021, Solidworks PDM, URL https://www.solidworks.com/product/solidworks-pdm,
Accessed: 2021-06-06

SQuaRE Team, 2021a, *Vera Rubin Observatory Technical Documentation Portal*, URL https://github.com/lsst-sqre/www_lsst_io,
Accessed: 2022-10-10

SQuaRE Team, 2021b, Ook, librarian indexing bot, URL https://github.com/lsst-sqre/ook,
Accessed: 2022-10-10

SQuaRE Team, 2022, *Documenteer*, URL https://github.com/lsst-sqre/documenteer,
Accessed: 2022-11-13

Wikimedia Foundation, 2021, Tree structure, URL https://en.wikipedia.org/wiki/Tree_structure,
Accessed: 2022-05-23

Xerox, 2021, Docushare enterprise content management platform, URL https://www.xerox.com/en-us/services/enterprise-content-management/docushare, Accessed: 2022-10-10

C Acronyms

Acronym	Description
API	Application Programming Interface
CMMS	Computerized Maintenance Management System
CSC	Commandable SAL Component
ComCam	The commissioning camera is a single-raft, 9-CCD camera that will be installed in LSST during commissioning, before the final camera is ready.
DDS	Data Distribution System
DM	Data Management
DMTR	DM Test Report
DOE	Department of Energy
DP	Data Production
EFD	Engineering and Facility Database
EPO	Education and Public Outreach
FMEA	failure modes and effect analysis
FRACAS	Failure Reporting Analysis and Corrective Action System
GIS	Global Interlock System
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HVAC	Heating, Ventilation, and Air Conditioning
IMS	Integrated Master Schedule
IT	Information Technology
L1	Lens 1
L2	Lens 2
L3	Lens 3
LDM	LSST Data Management (Document Handle)
LOVE	LSST Operations Visualization Environment
LPM	LSST Project Management (Document Handle)
LSE	LSST Systems Engineering (Document Handle)

LSST	Legacy Survey of Space and Time (formerly Large Synoptic Survey Telescope)
LSSTC	LSST Corporation
LaTeX	(Leslie) Lamport TeX (document markup language and document preparation system)
M1	primary mirror
M2	Secondary Mirror
M3	tertiary mirror
MIE	Major Item of Equipment
MREFC	Major Research Equipment and Facility Construction
NSF	National Science Foundation
PCW	Project Community Workshop
PDF	Portable Document Format
RTN	Rubin Technical Note
S3	(Amazon) Simple Storage Service
SAL	Service Abstraction Layer
SE	System Engineering
SLAC	SLAC National Accelerator Laboratory
SP	Story Point
SQR	SQuARE document handle
SQuaRE	Science Quality and Reliability Engineering
SaaS	Software as a Service
T&S	Telescope and Site
TMA	Telescope Mount Assembly
TS	Test Specification
URL	Universal Resource Locator
VPN	virtual private network
WBS	Work Breakdown Structure
WFD	Wide Fast Deep
XML	eXtensible Markup Language

FIGURE 4: Temporary Caption.

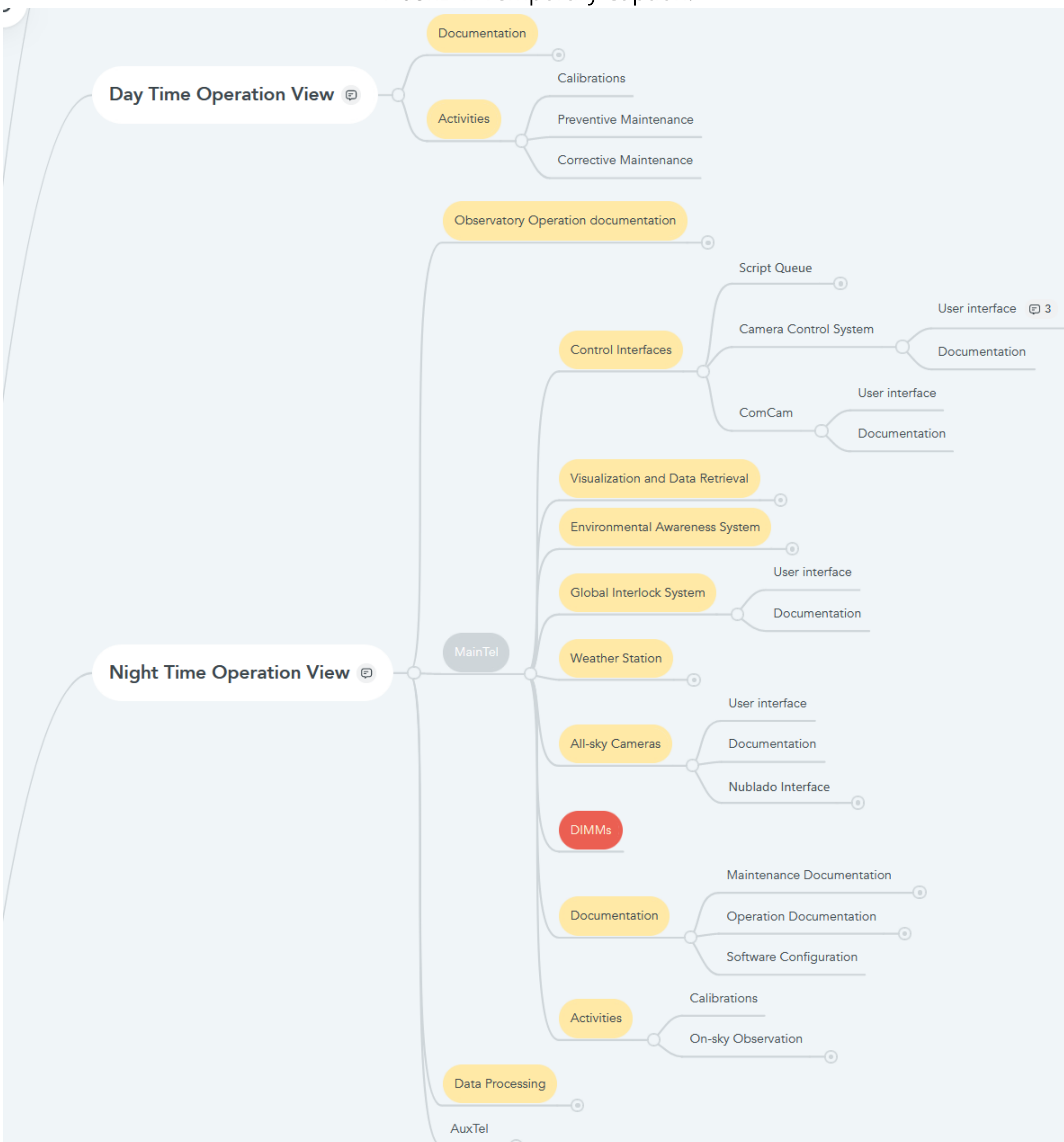


FIGURE 5: Temporary Caption.

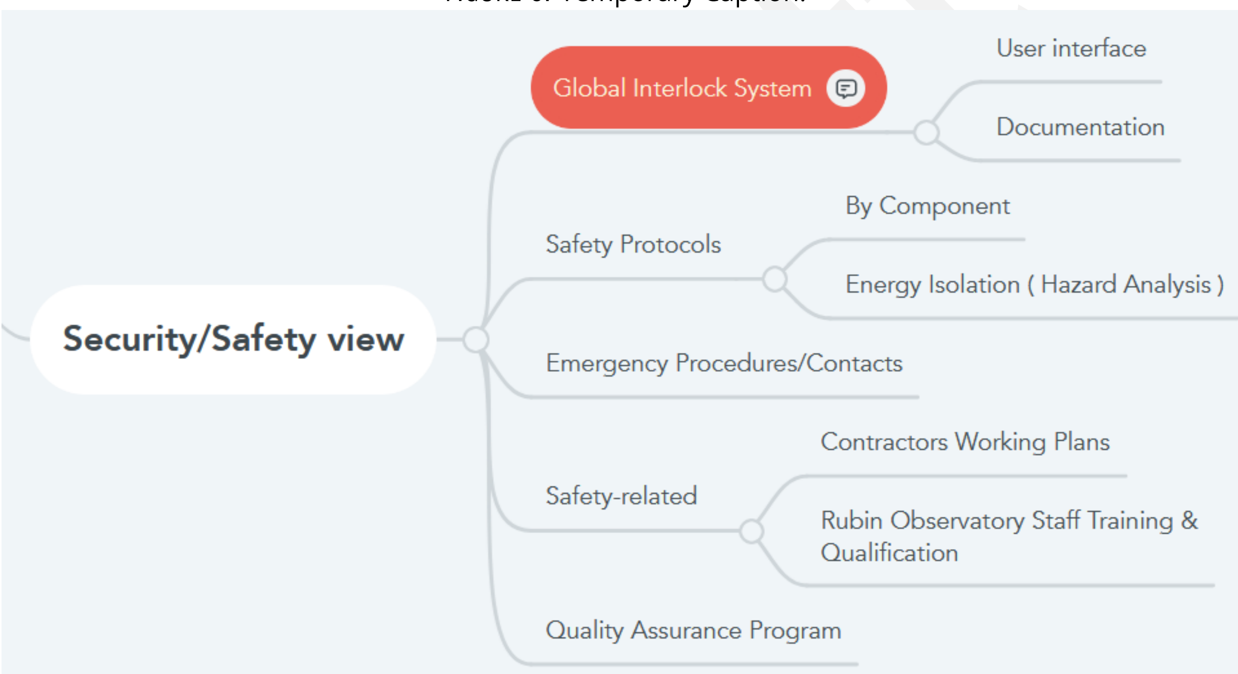


Figure 6: Temporary Caption.

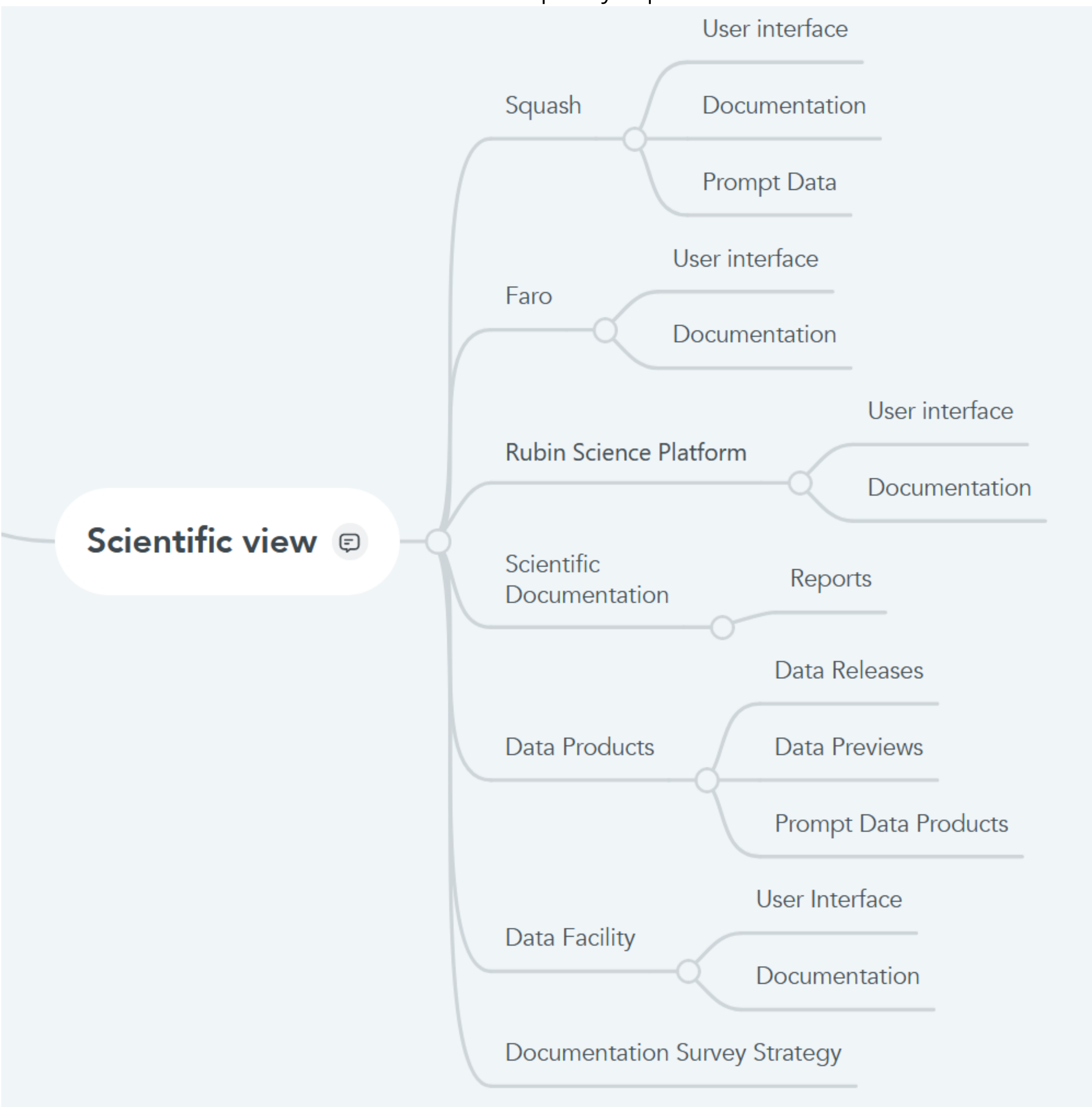


FIGURE 7: Temporary Caption.

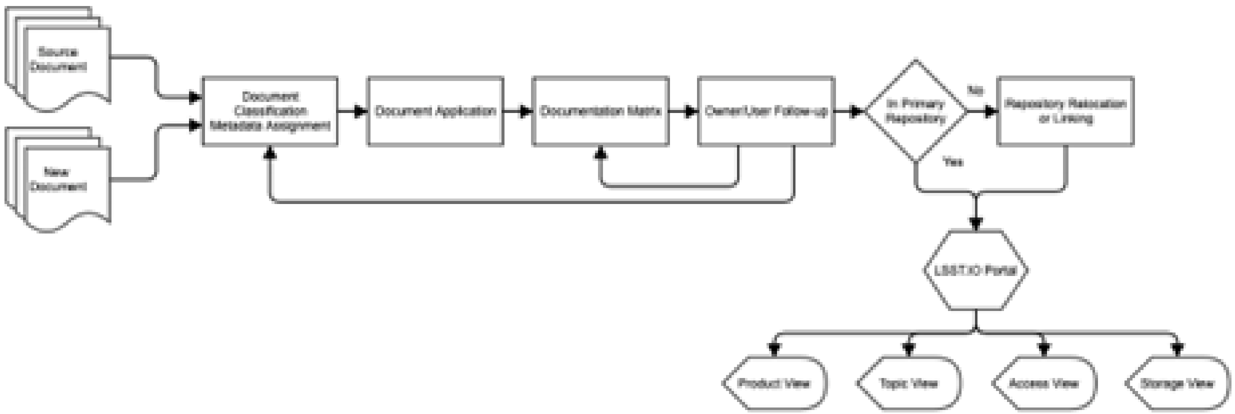


FIGURE 8: Architecture of the Proposed Documentation Portal. Users find documents on the web portal application, which in turn provides links into the original documentation repositories. Data in the portal application is supplied by the Algolia search service (Algolia, 2021a), which in turn gets its metadata from the original documents in their repositories via the Ook indexing service (SQuaRE Team, 2021b).

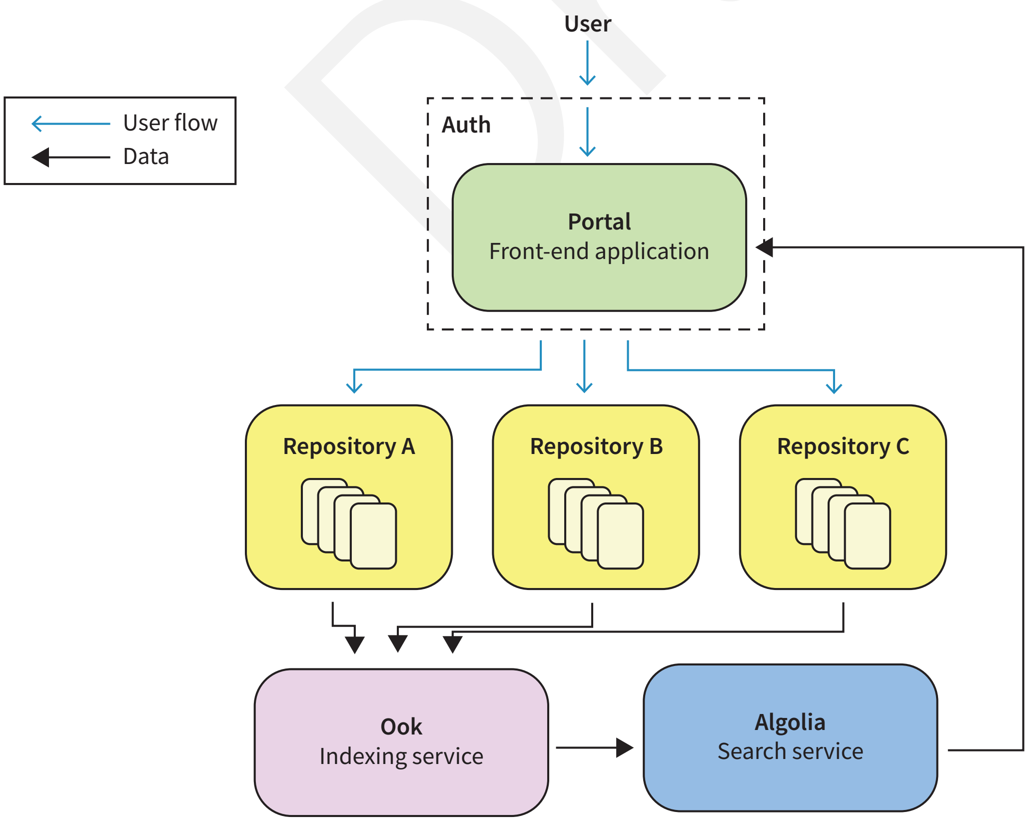


FIGURE 9: Temporary Caption.

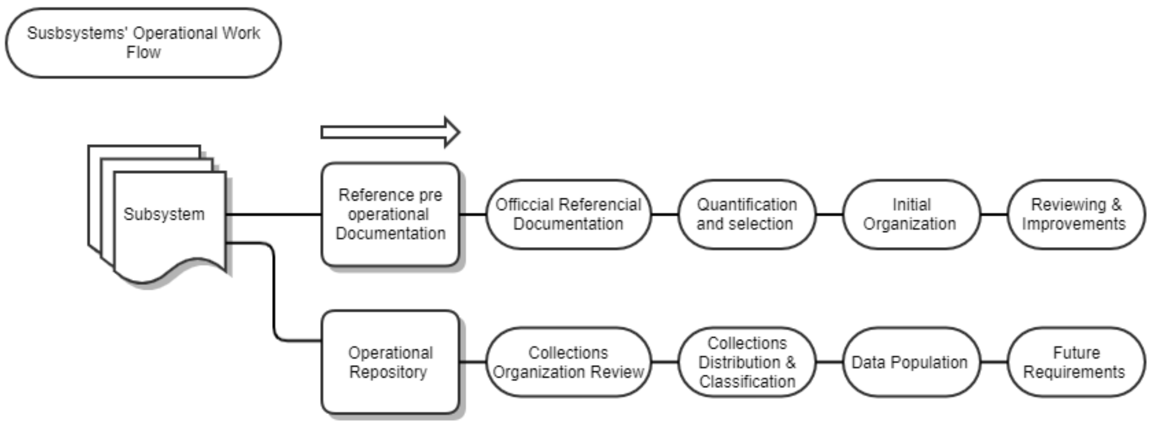


FIGURE 10: Temporary Caption.

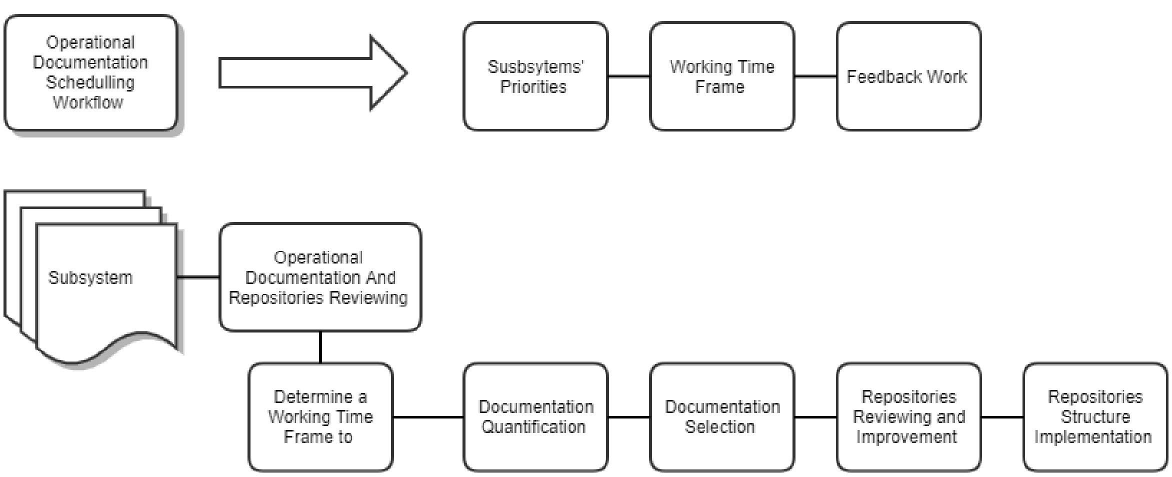


FIGURE 11: Example of Decomposition of Auxiliary Telescope Facilities (1 of 2).

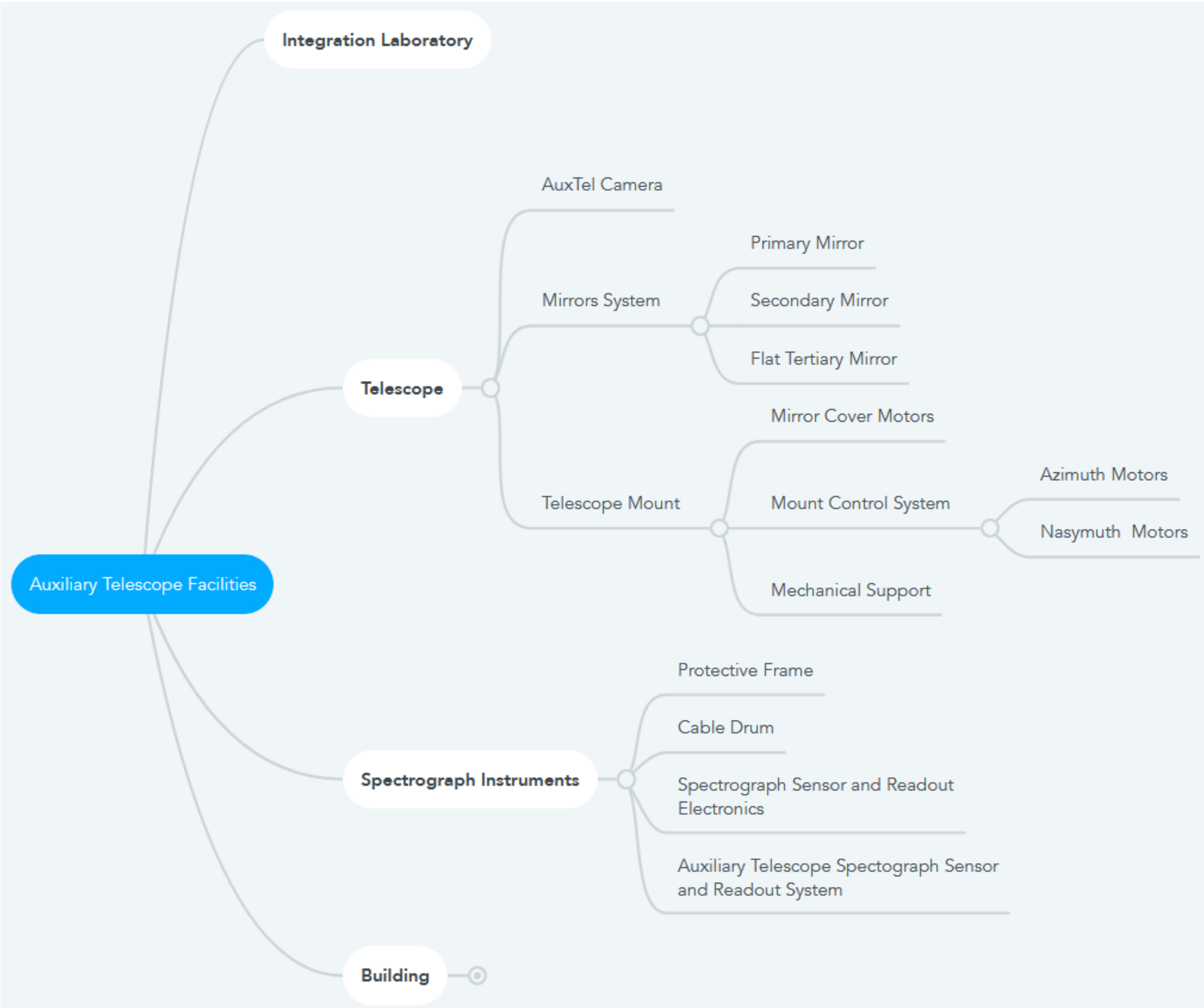


Figure 12: Example of Decomposition of Auxiliary Telescope Facilities (2 of 2).