LARGE SYNOPTIC SURVEY TELESCOPE

Large Synoptic Survey Telescope (LSST)
Systems Engineering

LSST Science Platform Vision Document

M. Jurić, D. Ciardi, G.P. Dubois-Felsmann, L.P. Guy

LSE-319

Latest Revision: 2019-07-24

Abstract

This document defines and describes the "LSST Science Platform," a set of integrated web applications and services deployed at the LSST Data Access Centers (DACs) through which the scientific community will access, visualize, subset, and perform next-to-the-data analysis of the data collected by the Large Synoptic Survey Telescope (LSST).

These services can be broken down to three different "Aspects": a web **Portal**, designed to provide essential data access and visualization services through a simple-to-use website, a **Notebook** environment, that will provide a Jupyter Notebook-like interface, based on JupyterLab, enabling next-to-the-data analysis, and an extensive set of **Web APIs** that the users will be able to use to remotely examine the LSST data set using tools they're already familiar with.

This document lays out the high-level vision for the aforementioned Aspects and some associated backend services. It is intentionally brief, and meant to generally guide the flow-down of requirements and development of product specifications, prioritization, and plans for the Agile development of the relevant elements of the DM system.

Change Record

Version	Date	Description	Owner name
	2017-05-22	Initial version	Mario Juric
1.0	2017-09-11	Approved in LCR-1013	Tim Jenness
	2019-03-08	Began changes for LSP review	Leanne Guy
2.0	2019-07-24	Approved in LCR-1728	Approver: V. Krabbendam; Document: T. Jenness

Document curator: Leanne Guy

Document source location: `https://github.com/lsst-dmsst/LSE-319`

Contents

LSST Science Platform Vision Document

1 Preface

The purpose of this document is to lay out the high-level vision for the **LSST Science Platform (LSP)**, a set of web applications and services through which the the scientific community will access, visualize, interact with, and analyze LSST data holdings. With its companion document — the Data Products Definition Document (DPDD) — it defines the high-level vision for LSST's end-user deliverables.

To a future LSST user, this document should illustrate what will be made available to the science community through the LSST Data Access Centers. To LSST builders, it provides direction on how to flow down the LSST System Requirements Document (LSR) and Observatory System Specficiations (OSS) to Data Management requirements (DMSR) as they pertain to the end-user services provided at the LSST Data Access Centers.

Though under strict change control, this is a living document. LSST will undergo a period of construction and commissioning lasting no less than seven years, followed by a decade of survey operations. To ensure its continued scientific adequacy, the high-level vision for the LSST Science Platform will be periodically reviewed and updated.

2 Introduction

2.1 Goals

The LSST is a facility whose primary mission is to acquire, process, and make available the data collected by its telescope and camera, [1] as well as enable "next-to-the-data" creation of added-value *User Generated* data products (see the SRD and the LSR).

This document describes the vision for the services to be put into place to fulfill the "*making available*" and "*User Generated* data product creation" aspects of LSST's mission. Its aim is to present a high-level description of the data access and analysis services provided at the LSST Data Access Centers. It should be read in conjunction with the LSST Data Products Definition Document (DPDD), which provides the high-level description of LSST data products.

Note that the present document uses the revised, user-facing terminology of "Prompt", "Data Release", and "User Generated" data products, in preference to the corresponding original language of "Level 1", "Level 2", and "Level 3" data products. This change, and the correspondence, is discussed in detail in document LPM-231. The LSST requirements documents continue to use the original language.

2.2 LSST Science Platform Overview

We define the **LSST Science Platform as a set of web applications and services made available to the scientific community to access, visualize, subset, and perform next-to-the-data analysis of the LSST data set**. It represents the integrated set of services that will be offered to LSST users.

The platform exposes the LSST data and services to the user through three primary user-facing "*Aspects*" — the web **Portal**, the **Notebook** analysis environment, and a machine-accessible **Web API** interface. These Aspects provide three different ways to access the data sets and analysis services provided in the LSST Data Access Centers (Figure 1).

The **Portal Aspect** is a web portal designed to provide the essential data access and visualization services through a simple-to-use website. It will enable browsing and visualization of the available datasets in ways the users are accustomed to at archives such as IRSA, MAST, or

[1] This includes the raw and processed calibration and engineering data, in addition to the data collected by the science sensors. Because much of LSST science will be systematics limited, access to engineering data will enable a better understanding and correction of subtle instrumental and/or environmental effects.

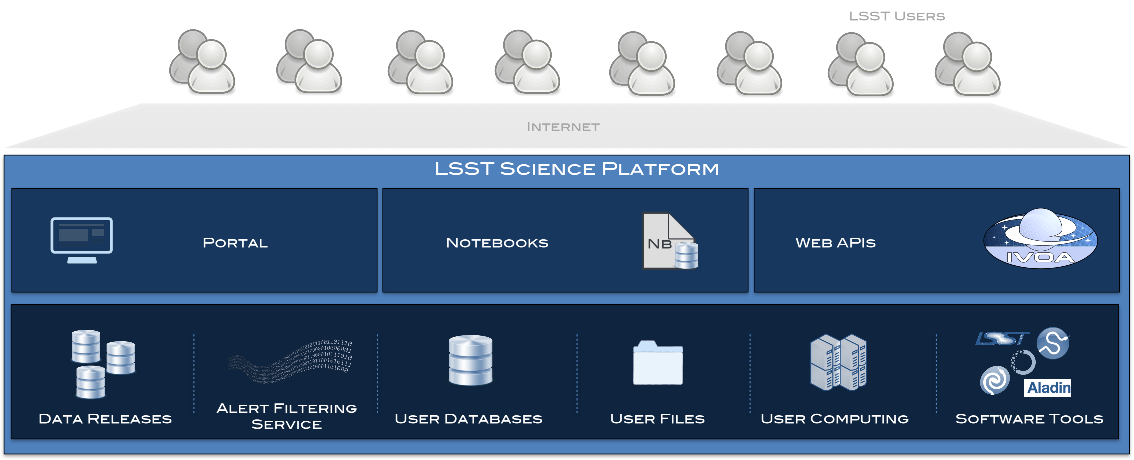

FIGURE 1: A high-level, layered, view of the LSST Science Platform. The LSST data will be exposed to the users through the web Portal, the Jupyter Notebook interface, and machine-accessible Web APIs. The web Portal component will provide the essential data access and visualization services common to present day archives. The Notebook component, based on the Jupyter family of technologies (JupyterHub and JupyterLab) will allow for more sophisticated next-to-the-data analysis. These user-visible services will provide access to the underlying core LSST data sets — the data releases and alert streams — and be supported by the User Database, File Storage, Computing, and Software Tools components. Together, they will enable the users to access, sub-select, analyze, and perform added-value processing of all flavors of LSST Data Products (see text for detail).

the SDSS archive. We describe it in more detail in Section 3.1.

The **Notebook Aspect** will provide a Jupyter Notebook-like interface, based on JupyterLab, and is geared towards enabling next-to-the-data analysis. The user experience will be nearly identical to working with Jupyter notebooks locally, except that computation and analysis will occur with resources provided at the LSST Data Access Center. This is an implementation of the "bringing computation to the data" paradigm: rather than imposing the burden of downloading, storing, and processing (potentially large) subsets of LSST data at their home institutions, we will enable our users to bring their codes and perform their analysis at the LSST DAC. This reduces the barrier to entry and shortens the path to science for the LSST science community. We describe it in more detail in Section 3.2.

The **Web API Aspect** of the LSST Science Platform will expose the services offered by the LSST Data Access Centers to other software tools and services using commonly accepted protocols. For example, industry-standard protocols such as WebDAV may be used to expose file data, or Virtual Observatory protocols for access to catalogs or images (TAP and SIAv2, respectively). This interface will open the possibility for remote access and analysis of the LSST data set using applications that the users are already comfortable with (e.g., TOPCAT or libraries like Astropy). Furthermore, the offered APIs will allow for federation with other astronomical archives, bringing added value to the LSST dataset. We describe it in more detail in Section 3.3.

The user-facing Aspects all depend on a common set of back-end services, which facilitate the LSP's operation as an integrated whole. The Data Releases will be organized as catalogs kept in relational database management systems, as well as repositories of files. The alert distribution system will facilitate the distribution of Alert Streams to community brokers and end-users (see the DPDD for details). These services will be complemented by additional User Database, File Storage, and Batch Computing services, as well as a pre-installed Software Tools suite, containing both LSST and community software. They will provide the computational power, data storage, and analytics capabilities needed to enable LSST data analysis as well as the creation and federation of *User Generated* data products. We further describe these in Section 4.

Finally, the LSST Science Platform is being envisioned to enable and encourage collaborative work. The capabilities ranging from sharing of derived datasets within smaller groups, collaborations, or with the broader LSST community, to collaborative visualization and editing capabilities expected to become available within the JupyterLab ecosystem (Section 3.6).

3 User-facing Services

3.1 Portal Aspect

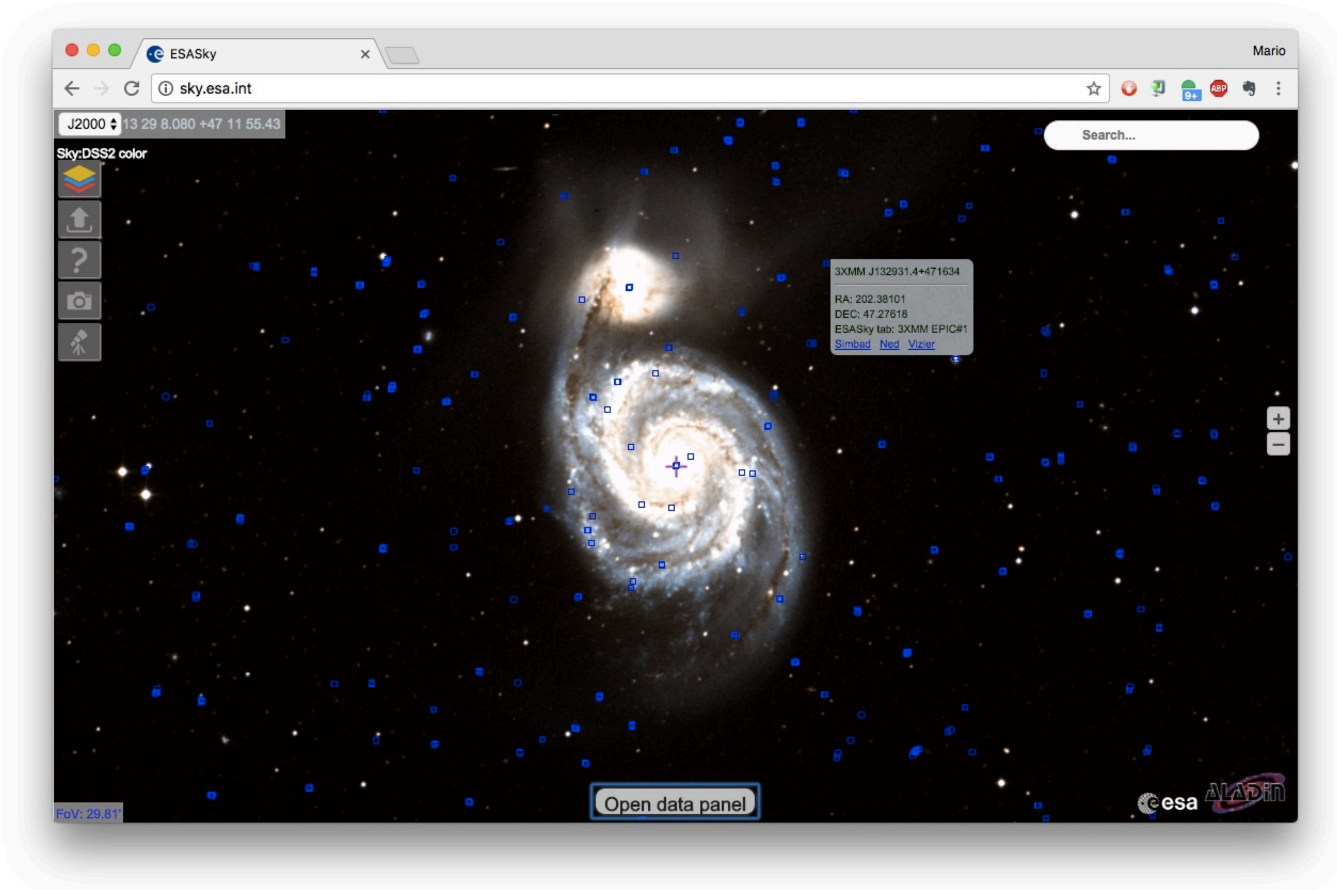

FIGURE 2: The "ESA Sky" web portal interface to ESA Archive holdings. The LSST portal user experience will support similar modern pan/zoom/select metaphor for exploration and visualization of the LSST data set.

The **Portal Aspect** is a web portal designed to provide the essential data access and visualization services through a simple-to-use website. It is to enable browsing and visualization of the available datasets in ways the users are accustomed to at archives such as IRSA, MAST, or the SDSS archive. To those we will add an enhanced level of interactivity in line with expectations for then-contemporary archive portals (similar to that found today in ESASky and the DECaLS Viewer). Examples of the types of user experiences to be offered through the LSST portal are shown in Figures 2 and 3.

Through the Portal, the users will be able to view the LSST images, request subsets of data (via

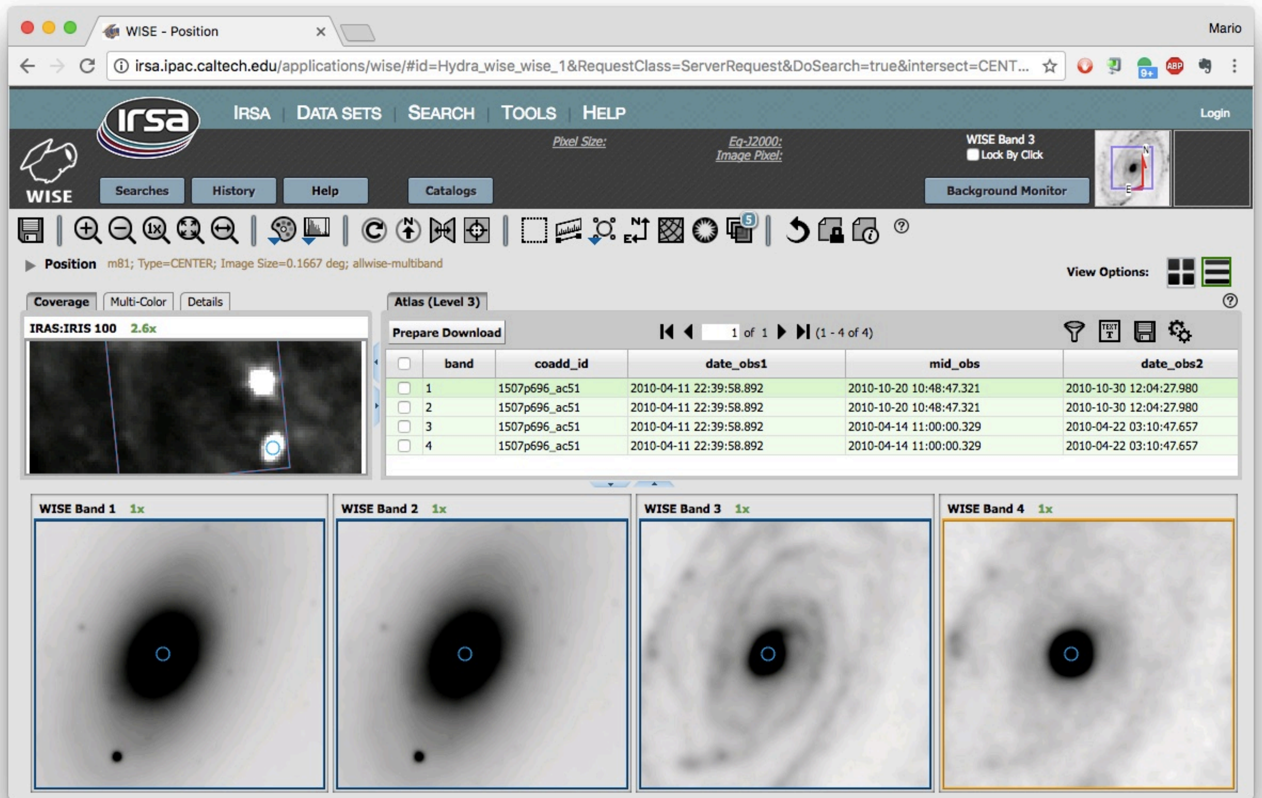

FIGURE 3: The web portal interface to the WISE data set at the Infra-Red Science Archive at IPAC. The LSST portal is being built by extending the Firefly toolkit that powers the IRSA/WISE archive.

simple forms or ADQL queries), store the results of such queries to their personal workspaces, as well as download them. The Portal will also make it possible to construct commonly requested plots, and generally explore the LSST dataset in a way that allows the users to identify and access (subsets of) data required by their science case.

Virtually all LSST users will use the Portal as their first point of entry to access and explore the LSST data set. When developing the Portal, we will therefore **emphasize the user experience and exploratory capabilities**, over analysis features. The latter are expected to be more directly satisfied by the Notebook Aspect.

3.2 Notebook (JupyterLab) Aspect

The **Notebook Aspect** will be provided to allow for more sophisticated data selection, analysis, and creation of added value *User Generated* data products. A screen capture of a mature prototype of the Notebook Aspect is shown in Figure 4.

The Notebook Aspect user experience will be nearly identical to working with Jupyter notebooks locally, except that computation and analysis will occur at resources provided at the LSST Data Access Center. This is an implementation of the "bringing analysis to the data" paradigm: rather than imposing the burden of downloading, storing, and processing (large) subsets of LSST data at their home institutions, we will enable our users to bring their codes to and perform their analysis at the LSST DAC. We expect this will reduce the barrier to entry and shorten the path to science for the LSST science community.

We will provide JupyterLab instances to LSST users in an environment carrying a library of preinstalled commonly used and useful software tools: Astropy, the LSST science pipelines, Anaconda Scientific Python Distribution, the PyViz visualization toolkit, and others. The users will be able to upload and install their own tools as well. Non-trivial shared computing cluster resources will be accessible through this environment as well, enabling the generation of *User Generated* data products.

The Notebook Aspect of the science platform will play a key role in commissioning, quality assessment, and science validation of the as-built system. It will be the primary method of performing interactive analysis of acquired data (e.g., adjusting and executing prepared notebooks driving commissioning tasks). Through the terminal functionality of the JupyterLab environment, it will also permit commanding the batch resources to execute larger processing tasks. Due to this, we expect the Notebook Aspect to reach maturity earlier than the others,

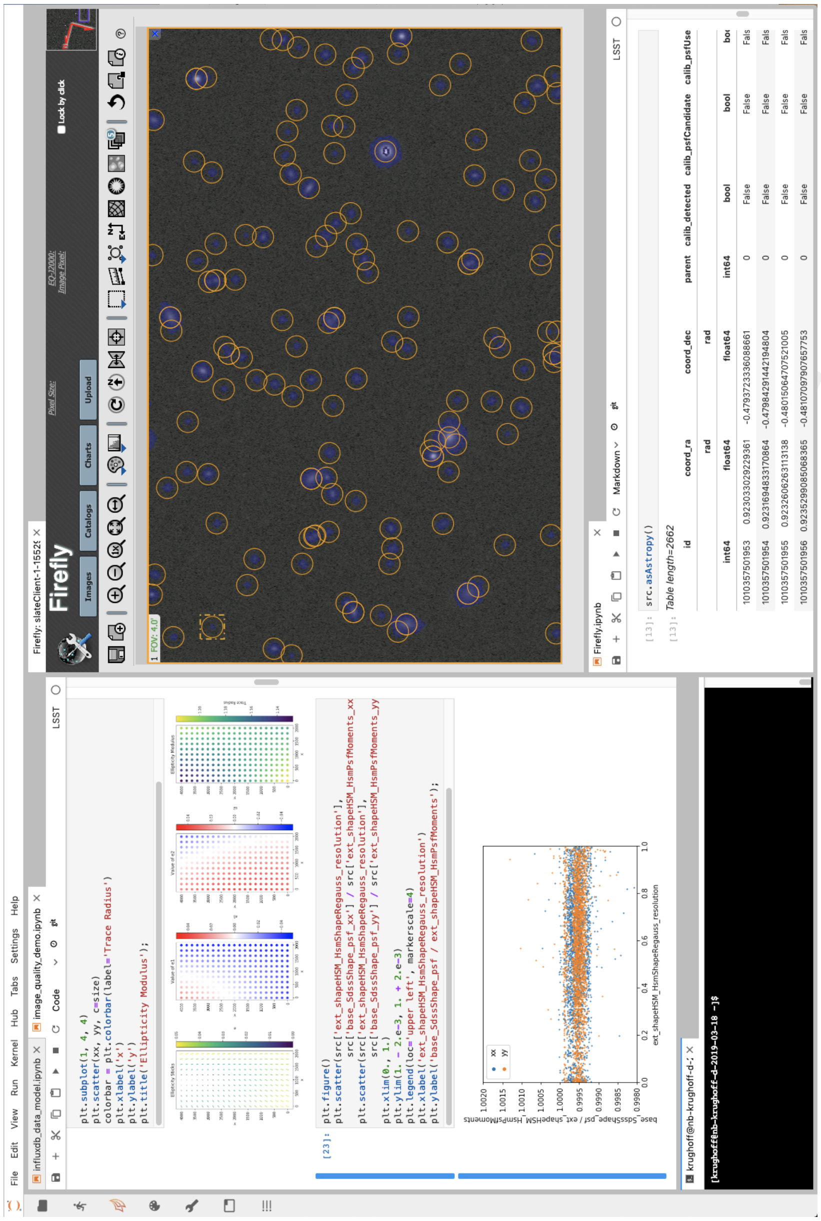

FIGURE 4: A screen capture of the Notebook interface running LSST image processing code within a notebook.

and certainly in time for commissioning.

3.3 Web API Aspect

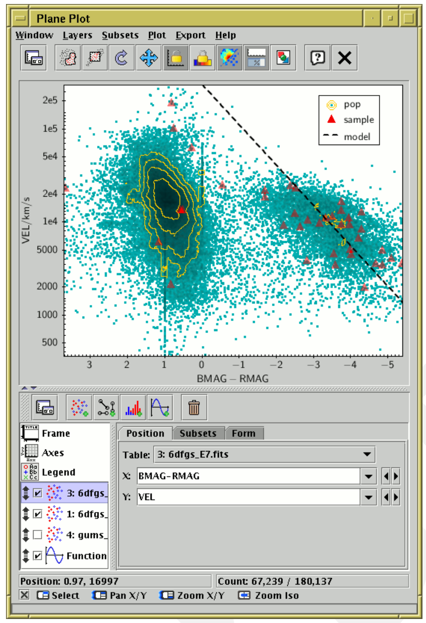

FIGURE 5: A screen capture of Tool for OPerations on Catalogues And Tables (TOPCAT), that is capable of remotely accessing catalogs using VO protocols. Tools such as these will be able to directly access the data sets served by the LSST DACs (figure credit: Mark Taylor, `http://www.star.bris.ac.uk/~mbt/topcat/sun253/sun253.html`).

Backend Platform services — such as access to databases, images, and other files — will be exposed to the public Internet through machine-accessible web APIs. These will serve the data using community-accepted formats and protocols, making it easy to remotely access the LSST data and DAC services. Furthermore, to ensure maximal exposure of the DAC services through the Web APIs, the other two Aspects of the Platform — Portal and Notebook — will internally access the LSST datasets using the same Web APIs to the greatest extent possible.

Exposing the LSST data through Virtual Observatory interfaces plays a particularly important role. This will allow the discoverability of LSST data products from within the Virtual Observatory, and federation of the LSST data set to other archives. It will also enable the use of widely utilized tools such as TOPCAT or DS9 by the end-users, further lowering the barrier to access to LSST data, and shortening the path to science. It will also allow these tools to be used in commissioning, easing the way for scientists new to the LSST environment to access the data and make meaningful contributions to this time-compressed activity.

LSST will follow a “VO-first” strategy, using IVOA-standard interfaces wherever practical, though we may also provide some services using custom protocols in areas where relevant standards have not yet been developed. LSST will actively participate in the IVOA community, and will propose evolutions of standards when we find problems and where we feel that our custom work could be of broader benefit to the community. We will also use our engagement with the IVOA to ensure that we are following mainstream interpretations of the standards and to maintain contact with the authors of client software such as TOPCAT.

Additional details on the standards to be supported are provided below.

3.4 Integrated environment

All the Aspects of the LSST Science Platform are intended to be *well integrated*, enabling a seamless workflow so the users will be able to move back and forth between them as needs dictate. The aim is to enable a user to find or create data in one Platform Aspect, and view or analyze that data in another.

As an example of how these connections can aid a user in exploring the LSST data, data queries will be shareable across the Portal, Notebook, and Web API Aspects. This will allow a user to build a query using the Portal's TAP query UI component, view the (possibly preliminary) results by browsing them in place in the Portal, and then access the final results from a JupyterLab notebook or a remotely connected client (e.g., TOPCAT) for further analysis. The reverse flow will also be enabled; a user can code and submit a complex ADQL query in the Notebook, and then browse and visualize the results in the Portal.

By making the environments integrated, we allow for a shallower learning curve and a gradual transition to more complex environments at the point they are needed. For example, a user may begin interacting with the LSST dataset using the Portal Aspect but may ultimately reach the limitations of the selection and analysis tools it provides. The integrated nature of the platform will allow such a user to switch to the Notebook Aspect, and continue working on the data analysis started in the Portal. There, they will be able to import the analysis artifacts (e.g., catalog subsets) as standard Python objects (e.g., as astropy.table).

3.5 Next to Data Processing

Many LSST science cases will require analysis of the full LSST Object and/or Source tables; subsetting and downloading reduced catalogs will not suffice for science that requires, for example, fitting and computing features of light curves for large numbers of Objects, creating maps of stellar density, or training machine learning models. The LSST Object table alone will be ≈ 50TB for Data Release 1 and ≈ 300TB for Data Release 11 at the close of the 10-year survey. The LSST Science Platform will enable user-driven large-scale search and analysis capabilities of the LSST data by providing a *Next-to-Data* processing environment, allowing users to bring their analysis to the data. This environment will be supported by backend services, as described in 4

3.6 Supporting Collaborative Work

The LSST Science Platform will provide support for collaborative work at two levels:

- **Shared workspaces**: Creation and sharing of data sets — catalogs, images, queries, and other data products — within either pre-defined or dynamically-created groups (e.g., a research group at a university, or a large science collaboration). Such groups would have access to a shared virtual "workspace" within the LSST DAC. This workspace will include shared files, shared catalogs (stored in user databases) as well as computing cycles allocated to the group as a whole. This shared workspace will be equally "visible" from all three Aspects of the platform — e.g., uploads to the workspace will be possible either through a form in the Portal, POSIX-style file access from JupyterLab, or using an external file transfer client via WebDAV or VOSpace.
- **Shared editing**: Although we have no formal requirement to provide shared editing capabilities, we envisage supporting "Google Docs"-like collaborative editing, visualization, and data analysis capabilities, via the integration of 3rd-party tools, *if and when these technologies become available in upstream products*.

The levels of support for collaboration described above are responsive to the large majority of user needs identified by end-user focus groups in R&D. At the same time, they minimize the technical risks by leveraging widely used and well understood technologies (SDSS-like MyDB user databases, backend authentication & authorization mechanisms, VO protocols, Jupyter).

4 Backend Services

The user-facing Aspects of the LSST Science Platform will be built on top of a number of back-end services that can roughly be divided into three categories: **database services**, **file services**, and **batch computing** services (bottom row of Figure 1). The details of these services are described in the Data Management Design Document (LDM-148) and other associated documents; here, we only provide the high-level guidance as to the capabilities which these services will need to expose to the user (through the three Aspects).

4.1 Database Services

Key LSST catalogs, both for **Prompt** and **Data Release** data products, will be stored in relational databases and made available for querying by users using the Astronomical Data Query Language (ADQL), with some restrictions. These products, and expectations surrounding their schemas, are further described in the DPDD.

Besides serving the LSST catalogs, LSST databases will also provide a per-user database space allocation. Within this allocation, end-users (including groups) will be able to store selected or transformed subsets of the LSST dataset, or upload related datasets for joining to the LSST dataset. The size of this allocation is determined by the SRD requirement to provide 10% of total LSST computing and storage resources to LSST users. All users will have a small initial allocation of space, with the possibility of applying to a Resource Allocation Committee for additional quota.

4.2 File Services

The LSST Science Platform will also provide a per-user file space allocation. End-users (including groups) may use this allocation to upload code, store selected or transformed subsets of the LSST dataset (e.g., images), and in general keep files needed to support their data analysis work. Note that some of this space may be provided in form of an object store, rather than a file system with POSIX-like semantics.

The size of this allocation is determined by the SRD requirement to provide 10% of total LSST computing and storage resources to LSST users. As in the case of the user databases, all users will have a small initial allocation of space, with the possibility of applying to a Resource Allocation Committee for additional quota.

4.3 Batch Computing Services

Analysis performed through the Portal, Notebook, and Web API Aspects will be served by a shared computing cluster. This cluster will be managed by a workload management system that ensures resources are allocated to individual users or groups based on pre-determined operational policies. The size of the batch computing resource is determined by the SRD requirement to provide 10% of total LSST computing and storage resources to LSST users. Again, all users will have a small initial allocation of batch CPU time, with the possibility of applying to a Resource Allocation Committee for additional quota.

The users will be able to launch jobs on the batch computing cluster primarily utilizing the APIs exposed through the Notebook and Web API Aspects of the LSST Science Platform. Some functionality exposed through the Portal may potentially utilize the batch computing cluster as well.

5 Development Methodology and Prioritization Guidance

5.1 Iterative Development Leveraging Existing Technologies

The services constructed for the LSST Science Platform will be developed following the iterative Agile methodology. While most of LSST software development follows this approach, adopting it is especially advantageous for user-facing services. There, iterative development and nearly continuous stakeholder feedback can provide guidance as to the details of features to be implemented, the continued validity of the approach taken, and the expected focus of intermediate milestones.

The development of the Portal, Notebook, and Web API Aspects will start from significant existing code bases and prior art. This is a deliberate approach designed to minimize technical risk and leverage end-user familiarity with these interfaces. The latter also reduces the barrier to user adoption of the products eventually delivered for LSST.

The **Portal Aspect** is based on the existing, production quality, archive portal interface developed at IRSA/IPAC — the *Firefly* toolkit. The primary challenge is integrating the existing Firefly code, and updating the user experience to conform to anticipated user expectations (e.g., supporting all-sky maps and pan/zoom/click-type exploration). Consistent with the general philosophy, DM should look at achieving the necessary upgrades by re-using existing well-known libraries and tools (e.g., the Plotly library is being adopted to provide 2D and 3D visualizations of tabular data).

The **Notebook Aspect** environment will be based on the open-source JupyterLab product delivered and maintained by the Jupyter team. The development of this Aspect of the LSST Science Platform will focus on deployment and integration with the LSST-specific backend services and other Aspects of the platform, rather than developing new or radically different features within the JupyterLab product.

Finally, the **Web API Aspect** is envisioned as implementing existing, widely-adopted, community protocols (in particular, ones from the Virtual Observatory suite of protocols and standards). Similarly to the other Aspects, it will benefit from leveraging existing codes and libraries wherever appropriate. We view providing a robust TAP (including ObsTAP) service as particularly critical, both for external and internal use. Other IVOA standards that we expect to support include SCS for simple catalog searches, SIAv2 for image searches, SODA for image cutouts and mosaics, and VOSpace (in addition to WebDAV) for access to user files. We are exploring the applications of the recently settled DataLink protocol for providing a richer view of LSST's data holdings and their relationships. We will also ensure that LSST data are well-represented in the IVOA Registries, particularly to enable the convenient discovery of LSST data through non-LSST client tools.

5.2 Prioritization Guidance

Here we give some overall feature prioritization guidance, to enable the construction of initial (mostly functional) requirements and intermediate development milestones.

Portal Aspect:

1. Deployment of the initial Firefly back-end within the (prototype) LSST Data Access Center at NCSA.
2. Integration of the initial Firefly front- and back-ends with LSST Science Platform backend services. For example, this includes the authentication and authorization mechanisms, relational databases, file stores, etc.
3. User experience improvements, such as addition of all-sky maps with pan/zoom/select navigation metaphors, modernization of the look-and-feel, streamlining of the UI and deprecation of rarely used widgets. **Once this level of functionality is met (at scale), the Portal Aspect will have achieved the minimum level of viability for deployment to operations**.
4. Improved user workflow integration with other Aspects of the LSST Science Platform. For example, it should be possible to begin data exploration in the Portal Aspect (e.g., by interactively selecting data sets) and seamlessly transfer the sub-selected catalogs and images to the Notebook Aspect's JupyterLab environment for further, more complex, analysis using provided Python libraries.
5. Addition of new widgets and abilities to the Portal, that address most requested and broadly useful end-user needs.
6. Widget-level integration with JupyterLab.

Notebook Aspect:

1. Deployment of the initial JupyterLab product within the (prototype) LSST Data Access Center at NCSA.
2. Integration of the JupyterLab product with LSP backend services, most notably authentication and authorization, user management, databases, and file stores. **Once this level of functionality is met (at scale), the Notebook Aspect will have achieved the minimum level of viability for deployment to commissioning and operations**.
3. Development of libraries and utilities to ease the submission of user-written code from Jupyter notebooks to the batch compute system.
4. Creation and curation of a library of 3rd party code that will be made available to LSP end-users.

Web API Aspect:

1. Development and deployment of initial data access APIs needed to satisfy the backend needs of the Portal and Notebook Aspects. These may not yet "speak" the final, standards-compliant, protocols.
2. Integration of the Web API Aspect with LSP backend services, most notably authentication and authorization, user management, databases, and file stores.
3. Deployment of key tabular data in IVOA-compliant data models — particularly image metadata in the ObsCore format.
4. Deployment of critical protocols (including SCS, TAP, ObsTAP, SIAv2, SODA, VOEvent streaming support, and VO Registry support) at commonly-encountered levels of standards compliance (e.g., the most commonly used ADQL features). **Once this level of functionality is met (at scale), the Web API Aspect will have achieved the minimum level of viability for deployment to operations**
5. Further improvement of standards-compliance as well as deployment of additional standards-compliant protocols, e.g., DataLink, throughout the Web API Aspect, and integration with all other elements of the Platform.
6. Further conformance of tabular data to IVOA and community data models, notably CAOM2.

It is assumed that the development of backend services will be driven by the needs of the front-end Aspects.

6 Design for Evolution

This document captures DM's response to our best estimate of what the expectations of LSST users are likely to be, starting with LSST Commissioning and through the first few years of LSST Operations.

Through a decade of operations, it is likely that both the user expectations will change, as well as the technologies available to respond to them. For example, the emergence of Jupyter as the dominant mode of remote data analysis in the astronomical community has caused the present vision and design of the LSST Science Platform to be markedly different than the original conceptual design of the Science User Interface and Tools (LDM-131). There is no reason to believe that similar shifts will not happen in Operations as well.

The LSST will therefore proactively design the LSST Science Platform services with such an evolution in mind, to a degree permitted by the available budget and schedule constraints. An example of such design for evolution is the concept of loosely coupled aspects itself (the Portal, Notebook, and Web APIs), that all expose different views of the same underlying data model and workspace. Design principles like these will allow for additions of new LSST Science Platform Aspects (e.g., a different Jupyter-like technology, should one emerge), replacements of Aspects (e.g., migrating to a different Portal technology), as well as retirement of Aspects that are not widely used.

References

[1] **[LSE-29]**, Claver, C.F., The LSST Systems Engineering Integrated Project Team, 2017, *LSST System Requirements (LSR)*, LSE-29, URL https://ls.st/LSE-29

[2] **[LSE-61]**, Dubois-Felsmann, G., Jenness, T., 2018, *LSST Data Management Subsystem Requirements*, LSE-61, URL https://ls.st/LSE-61

[3] **[LPM-17]**, Ivezić, Ž., The LSST Science Collaboration, 2018, *LSST Science Requirements Document*, LPM-17, URL https://ls.st/LPM-17

[4] **[LSE-163]**, Jurić, M., et al., 2017, *LSST Data Products Definition Document*, LSE-163, URL https://ls.st/LSE-163

[5] **[LDM-148]**, Lim, K.T., Bosch, J., Dubois-Felsmann, G., et al., 2018, *Data Management System Design*, LDM-148, URL https://ls.st/LDM-148