SLIDE 1

Hi, my name is Melissa Graham and I lead the Rubin Observatory Community Engagement team, which is the team responsible for supporting astronomers and students to do science with the Rubin data. Today I want to walk you through the experiences of a user doing science with the future Legacy Survey of Space and Time, the LSST.

The LSST will cover about a third of the southern sky each night. Over 10 years, the LSST will acquire millions of images, containing billions of objects, and make trillions of measurements of galaxies, stars, supernovae, and asteroids.

This unprecedented data volume and complexity requires a proportionately innovative and powerful platform for data processing and scientific analysis.

SLIDE 2

To give you a better sense of the scale, the image at left represents a tiny fraction of the LSST camera's full field of view, but the number of stars and galaxies you can see is representative of ten years of accumulated data. The final LSST ten-year sky map will be like having about three million of these tiled over the entire southern sky.

At right I'm showing a graph of how the Rubin Observatory's cumulative data holding will start at about forty petabytes with the first data release, and increase to about three hundred petabytes by the end of the ten year survey. To give you a relative sense of scale, in the first year of the LSST the Rubin Observatory's data holdings will surpass the volume of all prior NOIRLab data.

SLIDE 3

Before I jump in and show you some of the capabilities available to users of the Rubin Science Platform, and how Rubin Observatory plans to support scientists and students, I want to share the overall goal: to democratize science by removing barriers to participation in the Legacy Survey of Space and Time.

The Rubin Observatory takes several approaches to meeting this goal. One is to provide abundant, discoverable documentation of the end to end system, along with clear entry points and tutorials at the beginner through advanced levels. Another approach is to provide support systems that are asynchronous, distributed, and friendly. This addresses the particular barriers of knowledge bottlenecks, and of people feeling like they need to know someone "in the Rubin community" in order to get help. I will talk more about documentation and support at the end of this presentation.

The Rubin Science Platform is, essentially, a stable software environment with computational resources located next to the data. It is entirely browser-based and requires no data downloads or software installations, thereby removing barriers to participation that would otherwise arise in situations of limited access to hardware, software, and bandwidth.

At all times the Rubin Observatory prioritizes research inclusion and, especially now in this early commissioning stage as we begin to onboard new users during our Data Previews, prioritizes seeding expertise across a variety of institutional types, astronomical fields, career stages, and global locations.

The final bullet point here is more of a summary of our overall approach to democratizing science, which is to enable anyone, working in any astronomical field from the Solar System to the Milky Way to transients to cosmology, to become a power user of the Rubin Science Platform, and support our power users to push the cutting edge of science with the massive datasets of the LSST.

SLIDE 4

At the moment, the Rubin Science Platform is in regular use by hundreds of Rubin staff and community members, and this will more than double, just in the next year. The Rubin Science Platform is currently deployed at 12 sites globally, and the primary site for the science community is the Interim Data Facility in the Google Cloud, where we are currently serving simulated LSST-like data products. This increase in the number of active users will continue throughout commissioning until we have thousands of users by the start of Operations.

SLIDE 5

Now let's start on our journey of the Rubin Observatory user experience, which begins on the main landing page by pointing any web browser to the URL data.lsst.cloud. You can see the three aspects of the Rubin Science Platform here: the Portal, Notebook, and API aspect, which stands for Application Programming Interface and supports remote access. Today's tour will focus on the user experience with the Portal and Notebook aspects.

Let's say a user is interested in exploring measurements for the simulated stars that are currently available as part of Data Preview 0. They select the Portal option to open the Portal interface, which looks like this.

SLIDE 6

You'll see the acronym TAP in many places, and it stands for Table Access Protocol, which is a standard way of accessing tabular data. The user can browse the catalogs that are available – at the moment, only the simulated Data Preview 0 catalogs – and also the tables associated with that catalog – in this case, measurements of the simulated stars and galaxies locations, sizes, brightnesses, and so on, which the user can browse in the table at right.

SLIDE 7

To query and retrieve data from the Portal, the user can specify an area on the sky – for this demo I am using a 1 degree radius near the center of the simulated data set. The user can also decide which columns of data to retrieve from the table using the blue check boxes, such as sky location or brightness. Under constraints, users can place restrictions on the object's measurements. For example, I am restricting this query to only return objects with an extendedness value of 0, which means they appear as point sources and are likely to be stars, not galaxies. As you can see down at the bottom, the Portal has a default limit of fifty thousand objects, which is a good number to use as a test.

I want to point out that – not that we recommend this – but if a beginner-level user does not read any of the Rubin documentation or know anything about the data, the Portal interface is giving them sufficient information to set up this kind of basic search, because all the tables and columns have descriptions.

I also want to point out that users also have the option to "Populate and edit ADQL", which stands for astronomical data query language. If I click this button, it takes me to

SLIDE 8

this page, where all of the same query restrictions that I input to the Portal interface have been translated into the astronomical data query language format for the user. As you can see, this page also provides helpful guidance and examples.

In order to use the ADQL functionality, there is actually no need to set up a query using the main interface like we just did –

SLIDE 9

advanced users who are already familiar with the tables and their columns can go straight to this page by selecting query type "Edit ADQL" from the main portal interface.

Once the user is happy with the query they have set up, clicking on the search button at lower left will submit the query to the TAP service. This particular query only took about five seconds to execute.

SLIDE 10

The results page provides users with a projection of their retrieved objects superimposed on a sky image at upper left, a plot of two default columns retrieved from the table – in this case it is a heat map of the objects in sky coordinates – at upper right, and a table of the retrieved data along the bottom.

SLIDE 11

With a few minutes use of the Portal's plotting functionality, the user can create what is a very common figure for astronomers, a color-magnitude diagram, which plots star brightness versus star color, or temperature, and is commonly used to identify stellar populations. In this case, the simulated data that is available was actually designed for cosmology and not Galactic science, so this does not much resemble a real color-magnitude diagram. However, this is still a great demo of the Portal's basic functionality, especially for interactive scientific discovery – you can also see how the object selected in the table appears as orange in the plot, and how a pop-up of x and y values appears at my mouse location. Further restrictions can be placed on the data being plotted, such as here where I've restricted to objects with values in these columns to be less than 24. The plot automatically updates with every restriction added.

SLIDE 12

Now I want to continue this journey of the user experience through the notebook aspect of the Rubin Science Platform.

SLIDE 13

You heard all about a JupyterLab interface, and Jupyter Notebooks, earlier today from the NOIRLab CSDC folks. The main landing page of the notebook aspect in the Rubin Science Platform is standard, with the file system browser at left and a workspace at right. At the moment Rubin Observatory offers a limited set of tutorial notebooks for use with simulated LSST-like data, which covers our beginner- and intermediate-level users. Before showing you some science that can be done within notebooks, I want to emphasize that this is a platform for more than that: it is an interactive analysis environment that includes, for example, an interface to the batch processing system and parallelization infrastructure to support big data processing by users.

SLIDE 14

Our first notebook contains a lot of introductory narrative about how notebooks are mix of text and executable code cells, how to use them safely, and how to perform basic functions. Again, not that we recommend users skip the documentation, but if they do, they would find themselves informed about how to use notebooks and how to query and retrieve LSST data.

SLIDE 15

For example, this first notebook uses astronomical data query language to do the exact same search and retrieval that we just did in the portal, and then uses a basic plotting package to generate the same color-magnitude diagram.

SLIDE 16

This first notebook also demonstrates how to use one of the most powerful pieces of Rubin Observatory software, the data butler, to query and retrieve image data. In this example we're also using a package from the LSST Science Pipelines, called afwDisplay, to show the image.

SLIDE 17

And here I want to take a moment and really emphasize the importance and power of the LSST Science Pipelines, and of the data butler in particular. With the sheer volume of data we will have from the LSST, traditional archive and access architectures will not work. The term "LSST Science Pipelines" refers to all of the software developed, or adopted and optimized, by Rubin Observatory staff to process the LSST data. The Rubin Science Platform gives users access not just to the data, but to these requisite software packages and processing resources for query, retrieval, and analysis.

The data butler is a module that provides an abstracted data access interface. It can be used to read and write data without having to know the details of file formats or locations. There are no specific graphics to illustrate the data butler, but the two pop culture references I think of are the "pay no attention to that man behind the curtain" moment in the Wizard of Oz, and the scene in the 1995 hit Clueless when Cher uses a computer program to help her find and retrieve the perfect outfit from her massive closet. That's because the data butler allows users to ignore what is happening behind the scenes in terms of where and how the millions of images are stored, and enables users to flexibly query and retrieve only the data that they want.

SLIDE 18

For example, with a couple of lines of code and a short wait, the butler can find all of the individual images that overlap any region of sky, as I'm showing at left, along with metadata such as acquisition date in order to understand how the area of sky was surveyed by the LSST, as I'm showing at right.

Since the Rubin Science Platform provides users with both a stable software environment which always has an up-to-date version of the LSST Science Pipelines, and computational resources, users can perform custom reprocessing of the LSST images without having to write their own code. They can use the same pipelines that were used to make the LSST data releases, and output data products with the same format, but use customized input parameters.

SLIDE 19

One simple example that I'm likely to make use of is to combine all images that were acquired in the few weeks before a supernova explosion, to look for faint precursor events. But, the ability to apply custom processing with the LSST Science Pipelines has applications across the four science pillars of the Rubin Observatory, and beyond.

SLIDE 20

The final example that I want to show to demonstrate the power of the Rubin Science Platform is how it supports interactive catalog visualization using third-party packages such as bokeh, holoviews, and datashader, if you're familiar with those. In the previous examples I limited my search queries to fifty thousand objects, but of course in the Rubin era it would be a major risk to science to remain bound by such restrictions. This particular notebook is the work of Rubin staff members Keith Bechtol and Leanne Guy, and it retrieves and plots twenty-three million objects from the simulated data set.

SLIDE 21

This series of screenshots shows how users can interact with millions to billions of objects using these plotting packages which, with the underlying processing power of the Rubin Science Platform, automatically aggregates data into heatmaps and only plots individual points where and when they can be distinguished.

SLIDES 22-26 (replace with animated gif if possible)

Packages that enable plots to be linked and interactive, like in this series where selecting different regions in the left plot changes the histogram in the right, are also possible to use with millions of objects. When these plotting packages are used in the Rubin Science Platform, the interaction is quick, with each new render only a second or two. These capabilities will enable all users to do big science with the big data sets available from Rubin Observatory.

SLIDE 27

I want to touch on two final aspects of the Rubin Science Platform user experience before concluding: documentation and support. When users click on the Documentation menu item they are taken to

SLIDE 28

this page, from which they can access three main documentation resources. The first is the data release documentation, which contains high-level processing summaries and descriptions of the data products such as the types of images available, and schema for the catalogs. Since participation in the current release of simulated data, which we call Data Preview 0, is limited to a few hundred users, this documentation also includes basic access instructions and the schedules and connection info for all our virtual programming that we are providing, for user training and networking.

The second is user guides for the Rubin Science Platform, which contain instructions on how to use the various features. The third is the full set of documentation for the LSST Science Pipelines, which contains details for the python modules, use-case examples, answers to frequently asked questions, and descriptions of the software releases. I will spare you many screenshots full of documentation examples, and just say that multiple Rubin subsystems are collaborating to provide users with a full suite of documentation.

SLIDE 29

If users instead click on the Support menu item, they are taken to a page that will guide them to either a GitHub Issues repository to report technical problems like login errors, gateway timeouts, bug reports, lost passwords, and so on, or to the Rubin Community Forum for everything else, and especially for science support.

SLIDE 30

These are screenshots of the landing page of the Rubin Community Forum. At left is an expandable banner with information about the forum itself, and at right is the main interface to the forum's categories with a list of the most recent discussions. Anyone may make an account in the forum and post new questions and, crucially, post answers to other people's questions. When Rubin has thousands of users, knowledge bottlenecks become a real risk to science. There will be many instances where other users could just as easily – if not better in some niche science cases – answer questions.

For example, at bottom right there is a topic with the title "How to predict magnitudes from spectra". Synthesized photometry is a very common need, and there already exists a python module to do this with the LSST filters in the survey strategy software package. I happened to respond to this query first, but there are many people who could have answered that question. And now, any user that searches the Rubin Community Forum for the term "synthetic photometry" will find this question, and its answer. The fact that this user's question was publicly posted where I could see it, and not, for example, sent in an email to their Rubin-related colleagues, is also a good feature of the forum because it reminded me that we could better connect to the survey strategy software documentation, so that this information is more discoverable to users.

To summarize, the Rubin Community Forum enables users to crowd-source their support when possible, and this model leaves Rubin staff free to focus on the user questions that require their expertise.

SLIDE 31

In conclusion, I want to leave you with the core message that the Rubin Science Platform user experience will be access to a powerful platform built specifically for the LSST era, with abundant and sustainable support for everyone to do big science.