Large Synoptic Survey Telescope

www.lsst.org

Advancing the LSST Operations Simulator

Abhijit Saha[1], Francisco Delgado[1], Kem H. Cook[2], Stephen Ridgway[1], Srinivasan Chandrasekharan[1], Catherine Petry[3] and the Operations Simulator Group

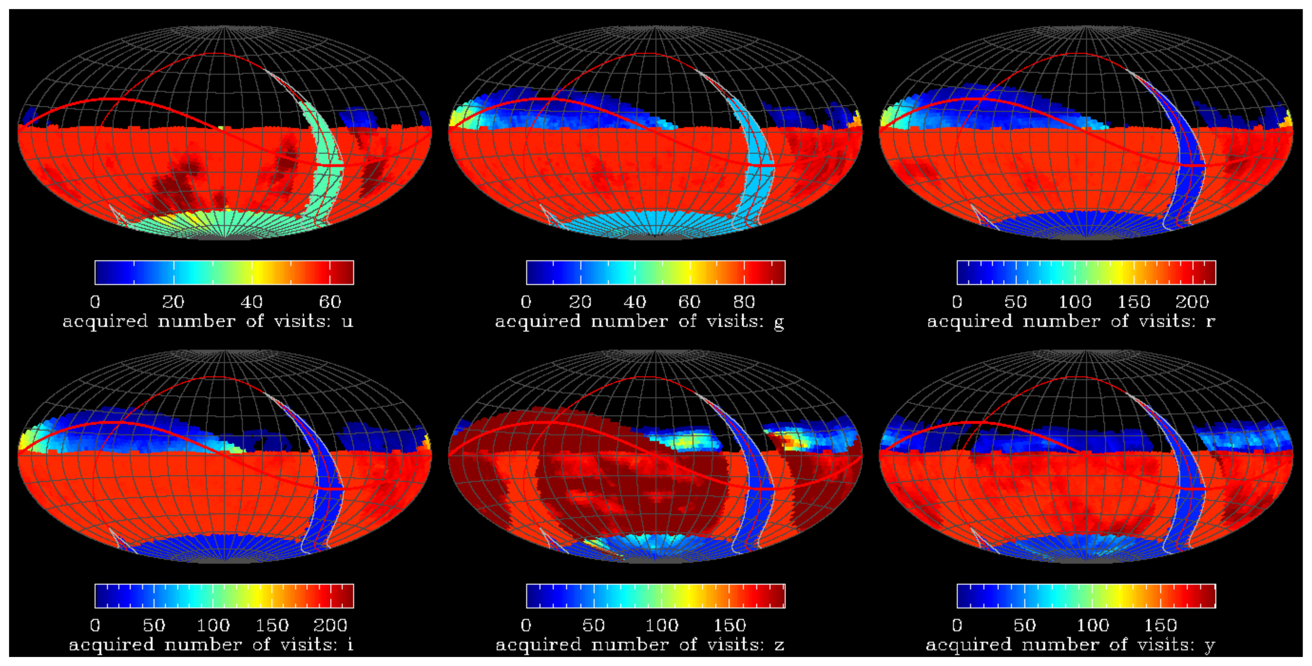
[1]NOAO, [2]Eureka Science, [3]Unaffilliated

The Operations Simulator for the Large Synoptic Survey Telescope (LSST; http://lsst.org) allows the planning of LSST observations that obey explicit science driven observing specifications, patterns, schema, and priorities, while optimizing against the constraints placed by design-specific opto-mechanical system performance of the telescope facility, site specific conditions (including weather and seeing), as well as additional scheduled and unscheduled downtime. A simulation run records the characteristics of all observations (e.g., epoch, sky position, seeing, sky brightness) in a MySQL database, which can be queried for any desired purpose. Derivative information digests of the observing history are made with an analysis package called Simulation Survey Tools for Analysis and Reporting (SSTAR). Merit functions and metrics have been designed to examine how suitable a specific simulation run is for several different science applications. This poster reports recent work which has focused on an architectural restructuring of the code that will allow us to a) use "look-ahead" strategies that avoid cadence sequences that cannot be completed due to observing constraints; and b) examine alternate optimization strategies, so that the most efficient scheduling algorithm(s) can be identified and used: even few-percent efficiency gains will create substantive scientific opportunity. The enhanced simulator will be used to assess the feasibility of desired observing cadences, study the impact of changing science program priorities, and assist with performance margin investigations of the LSST system.

What is OpSim?

- The Operations Simulator (OpSim) simulates observations made over a desired period of time (nominally 10 years)
    - Science-driven survey requirements and goals specify how the observations are to be made.
    - OpSim simulates the pointings that are consistent with the telescope constraints: the availability of targets on the sky at any given time, sky illumination by the lunation cycles, weather and observing condition patterns for the observing site, and both stipulated and randomly imposed down-time for maintenance and repair.
    - OpSim output is a list of positions and epochs (along with meta-data such as airmass, sky-brightness, transparency) of all observations made by the simulated telescope over the survey period.
    - Analysis of the `observing history' tests whether the observations suffice for a given desired science case.
- The role of OpSim:
    - Makes it possible to see whether a desired set of observations can in fact be obtained within the duration of the LSST survey.
    - Allows us to examine the consequences of adding additional observations from a new science driver, or the outcome from changing the observing strategy for existing science cases. What is the impact of the change on all other science cases?
    - Used for system margin analysis and to advise engineering trade-off decisions.
    - Expected to evolve into the system that manages the real-time observation planning for the actual telescope.

A Primary Achievement



- A relatively simple approach provides an existence proof for the feasibility of the survey.
- Standard reports from SSTAR created using observing history files of an OpSim run provide many diagnostics of a survey.
- Figure 1 shows the number of visits in the various pass-bands as a function of position on the sky from all the contributing science drivers in the baseline model for the survey (OpSim Version 2.6)

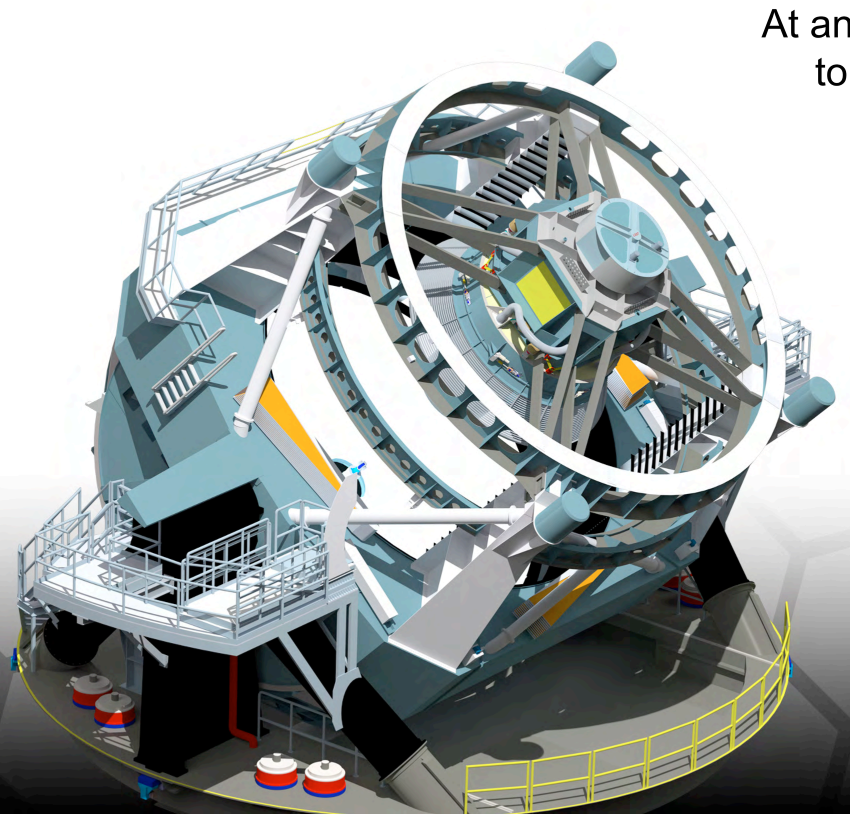
***Figure 1**. The number of visits in each pass-band as a function of position in a 10 year "baseline" survey. The number of visits acquired for each field is plotted in Aitoff projection for each filter. All visits acquired by all observing modes are included in this plot.*

How Does OpSim Work?

A simulated survey is driven by at least one but usually more "observing modes." An observing mode is a cadence strategy designed to visit and revisit specific fields on the sky in a way that meets a particular science objective. The observing mode is described by a set of input configuration parameters. Each science case will typically have its own observing mode, although multiple science cases may be consolidated into one mode. Examples of modes are:

Wide-Fast-Deep Universal Cadence: The "universal" cadence for the "Wide-Fast-Deep" (WFD) survey, covering ~18,000 deg^2 with two visits per night separated by about 30 min. on average every few nights in different pass-bands.
Deep Drilling: A few select fields, with multiple exposures in rapid succession are taken in all bands for *m* minutes, and then repeating every *n* nights.

At any time along the simulated survey, each survey field computes a "demand" to be observed for each science mode based on an algorithm specific to the observing mode and the timing requirements and history and timing of past observations of that field. These are filtered or masked by whether the object is available in the sky at that instant, and by the current observing conditions. The demand functions thus modulated are then weighed against the telescope overhead `cost function' (e.g. slew time, filter change) for a prospective target, and a "greedy" algorithm is used for target selection.



January 2013

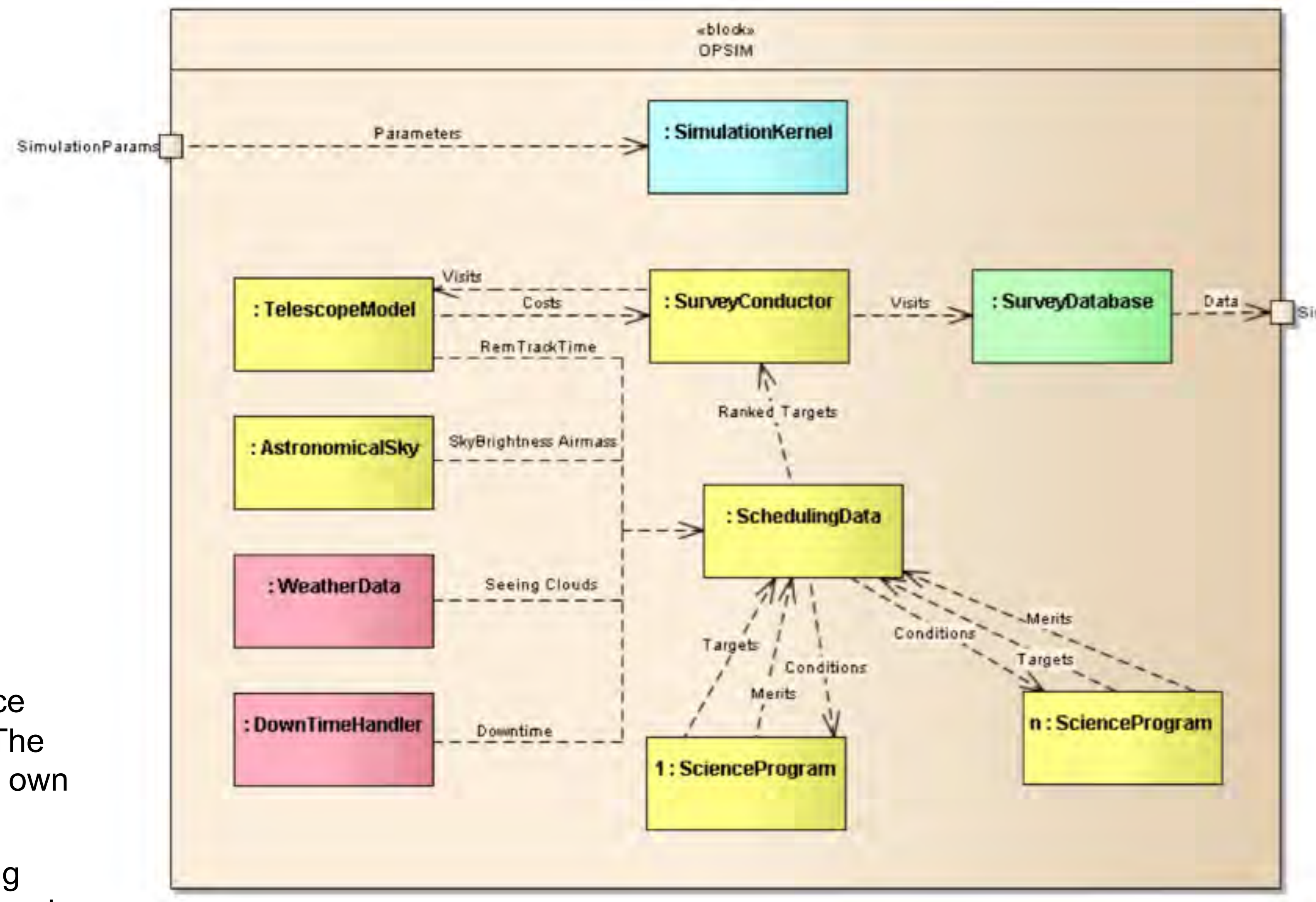
The Larger Role for OpSim and the Need for a New OpSim Architecture

- OpSim role is central to the operations era, both for planning and for actual telescope scheduling
- OpSim must evolve into the schedule driver for the real telescope – modularity is essential
- Efficient/optimal scheduling is the primary goal for the operations phase – an increase by even a few percent creates the opportunity to include more science cases than are now in consideration
- Current implementation of OpSim, only optimizes the "next" telescope visit, and ignores the effect this choice has for subsequent visits, and the consequent overall impact on efficiency: the most egregious example of this is when a sequence of observations is started that cannot complete because the target cannot be accessed at the required future epoch in the sequence.

Architecture Features Needed

- Add the ability to implement strategies which can "look-ahead". This cannot be supported by the Version-2 structure of OpSim.
- Implement alternative, often more complex observing modes, which cannot be done easily or at all with the current architecture.
- Add "warm-start" capability. Re-definition or re-prioritization of the science goals mid-way into the survey means planning the survey given the history of what has been observed already and a new set of "proposals" and observing modes
- Increase modularity, so that when adapting it for the real telescope, we need only exchange some of the modules, keeping in place the processing mechanism.

Accommodating these needs, we arrive at a design for Version 3 (see Figure 2).



1. *The simulator Kernel receives the simulation parameters, distributes them to the other components, and coordinates information for the simulation (flows not shown to avoid clutter).*
2. *Each ScienceProgram (or observing mode) provides its own list of targets to the SchedulingData component, which also receives the precomputed conditions from AstronomicalSky and DownTimeHandler.*
3. *Given the time profile of these conditions for each target, the SciencePrograms perform the visibility and merits (demand function) computation according to their own science goals, parameters and visit history.*
4. *These merits are stored in SchedulingData and the ranked targets sent to SurveyConductor, which analyzes the costs of the targets from the TelescopeModel to produce the schedule for the next N visits.*

***Figure 2**. The Internal Block Diagram shows the main components of the operations simulator and a simplified set of data flows in Version 3 of OpSim. The modularity of this design allows easy adaptation to the actual scheduler for the real telescope: the yellow components need not change, but the pink ones, which currently are themselves simulators, will be replaced by real time information feeds.*

OpSim Version 3 Planned Next Steps

- Implement look-ahead capabilities
- Increase flexibility for algorithm experimentation
- Expand capability for scripted cadences
- Improve database structure and performance
- Improve code architecture and performance

LSST is a public-private partnership. Design and development activity is supported in part by the National Science Foundation. Additional funding comes from private gifts, grants to universities, and in-kind support at Department of Energy laboratories and other LSSTC Institutional Members.