VERA C. RUBIN OBSERVATORY

U.S. National Science Foundation | U.S. DEPARTMENT of ENERGY | Office of Science

Rubin Observatory Data through Two AI Eras

Yusra AlSayyad

for Rubin Obs Data Management

July 16 2026

FOOD?

The End of Science

**The quest for knowledge used to begin with grand theories.
Now it begins with massive amounts of data. Welcome to the Petabyte Age.**

July 2008

CHRIS ANDERSON SCIENCE JUN 23, 2008 12:00 PM

The End of Theory: The Data Deluge Makes the Scientific Method Obsolete

Illustration: Marian Bantjes "All models are wrong, but some are useful." So proclaimed statistician George Box 30 years ago, and he was right. But what choice did we have? Only models, from cosmological equations to theories of human behavior, seemed to be able to consistently, if imperfectly, explain the world around us. Until now. Today

wired.com/2008/06/pb-theory

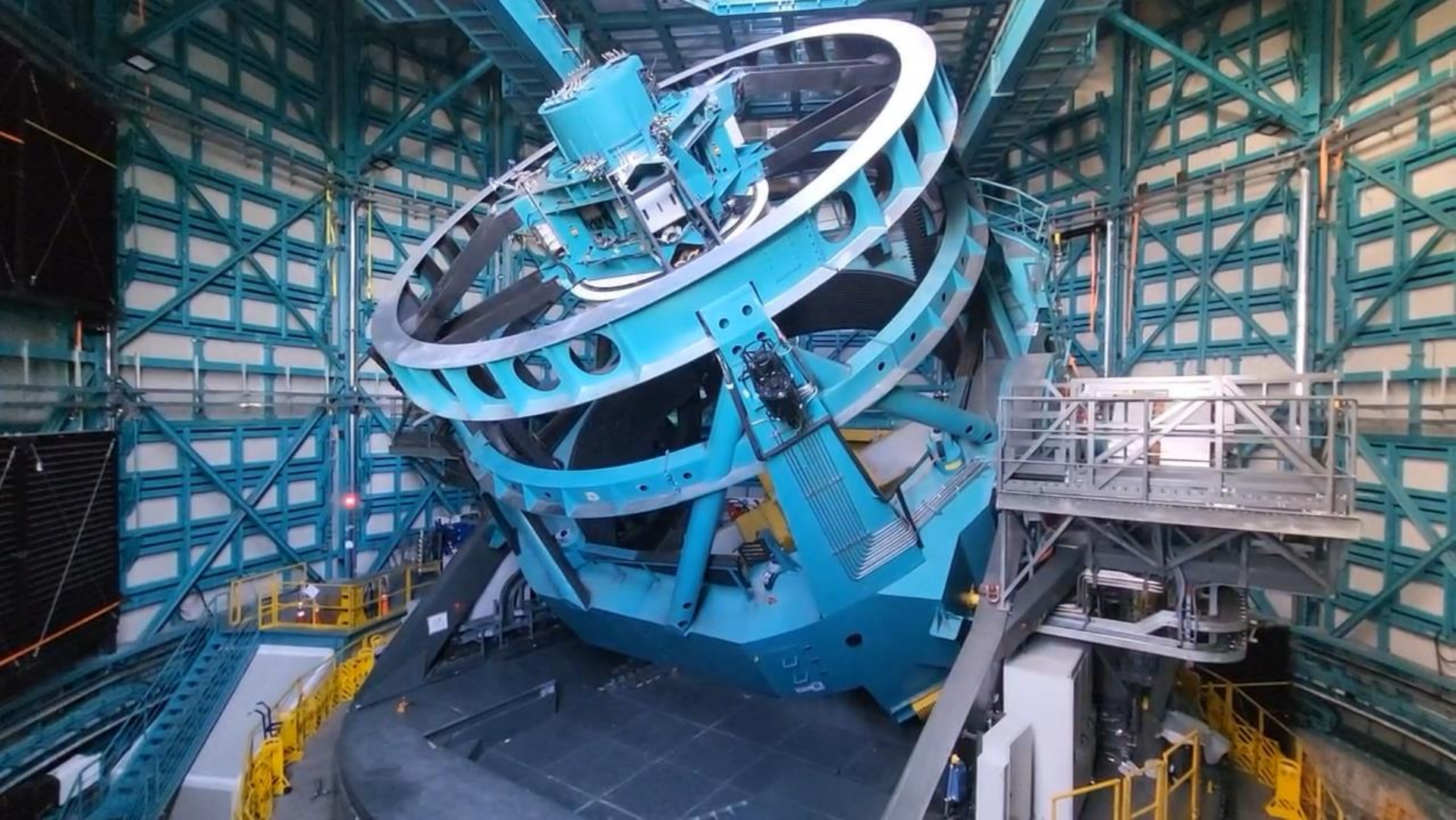
MICHAEL D. LEMONICK SCIENCE JUN 23, 2008 12:00 PM

Watching the Skies: Space Is Really Big — But Not Too Big to Map

True, imaging the entire sky, and understanding those images, won't be easy. The first telescope that will be able to collect all that data, the Large Synoptic Survey Telescope, won't be finished until 2014. Perched atop Cerro Pachón, a mountain in northern Chile, the LSST will have a 27.5-foot mirror and a field of view 50 times the size of the full moon seen from Earth. Its digital camera will suck down 3.5 gigapixels of imagery every 17 seconds. "At that rate," says Michael Strauss, a Princeton astrophysicist, "the numbers get very big very fast."

The LSST will cover the sky every three days. And within the petabytes of information it collects may lurk things nobody has even imagined — assuming astronomers can figure out how to teach their computers to look for objects no one has ever seen. It's the first attempt to sort astronomical data on this scale, says Princeton astrophysicist Robert Lupton, who oversaw data processing for the SDSS and is helping design the LSST. But the new images may allow him and his colleagues to watch supernovae explode, find undiscovered comets, and maybe even spot that killer asteroid.

wired.com/2008/06/pb-asteroids



The Data Management Challenge

Fast Data for science cases that require rapid follow up

Prompt Products

Near-real-time processed data released within minutes to days

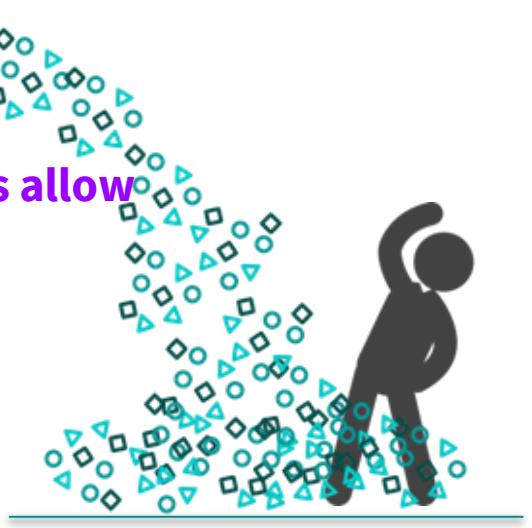
- **Alerts (within minutes):** Notify scientists of detected sky changes
- **Catalogs (~24 hours):** Databases of information about detected objects
- **Raw & Processed Images (~80 hours):** Available to data-rights holders

Up to 10 million alerts per night

Available
Now.
As templates allow

Fall 2026

Fall 2026



Navigating the Alert Stream

- To manage, Rubin parnters w/ **community brokers** — software systems that process & serve alerts to scientists
- Scientists access alerts via broker websites & can filter on their research area
- Alerts are **world public**

Alerce

Antares

AMPEL

Fink

Lasair

Babamul

Pitt-Google Broker

Data Releases for static sky science and statistical studies of variability

Annual Data Releases

Calibrated images and catalogs of billions of objects and their measurements (such as shapes, colors, positions) and brightness as a function of time.

Data Preview 2 (DP2): data release of 30,000 commissioning visits available **Fall 2026**.
Coadds and catalogs-only available July 27 as "Early DP2"

1. Take all data collected to date
2. Combine to emulate longer exposures…
3. …to reveal fainter objects and extract most info possible

Serve the catalogs in a way that supports data science

2009: Astronomers are going to query 100s of TB of catalogs and want millions of rows at a time.

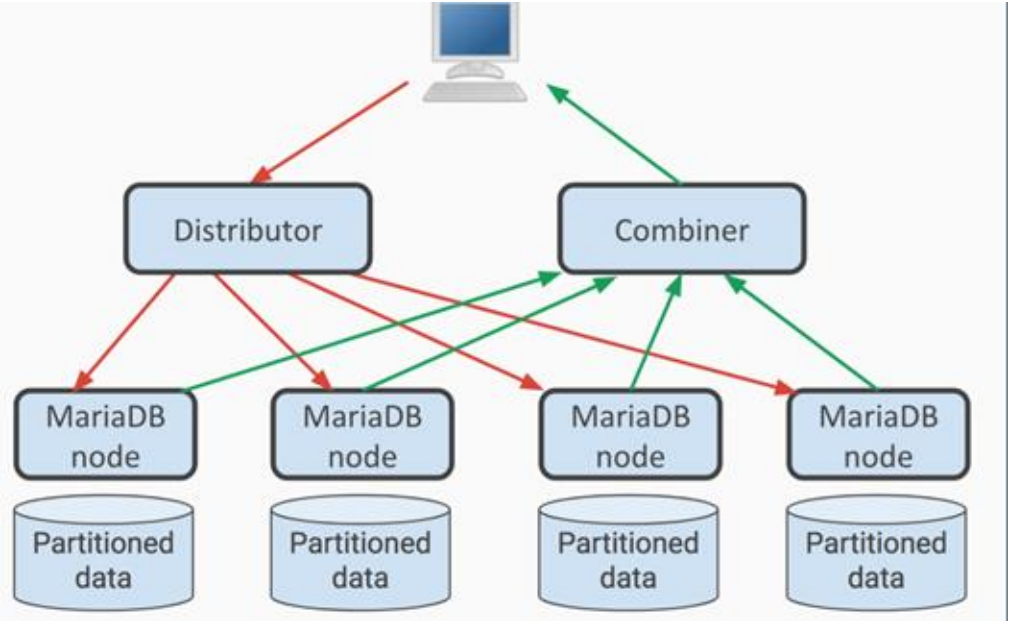
Need a parallel, distributed sql database with shared table scans!

Build QSERV

HTM-sharded

Maximizes bandwidth for the collective.

Get scientists those **rows!**



qserv.lsst.io

Actively drive the Data Management bus through the evolving data science landscape

Separation of
Compute and Storage

data lakes, S3

Spark announced 2010, Dask 2014

Serve the catalogs in a way that supports data science

2015: Astronomers are going to want to use Dask-like tools.

- Get scientists those **columns**!
- Maximizes bandwidth for a single user.
- => Serve catalogs in **parquet** format for next-to-database batch OLAP

Now: Astronomers still want columns, but only for a particular region of the sky. Need healpix-sharded columns!

=> LINCC Framework's LSDB/HATS: docs.lsdb.io

Data Preview 2 will have all 3. For data release 1 we'll collapse 2 and 3 probably.

Actively drive the Data Management bus through the evolving data science landscape

JupyterLab announced 2016

Rubin Science Platform

data.lsst.cloud

Portal

Discover data in the browser

Learn more about the portal.

Notebooks

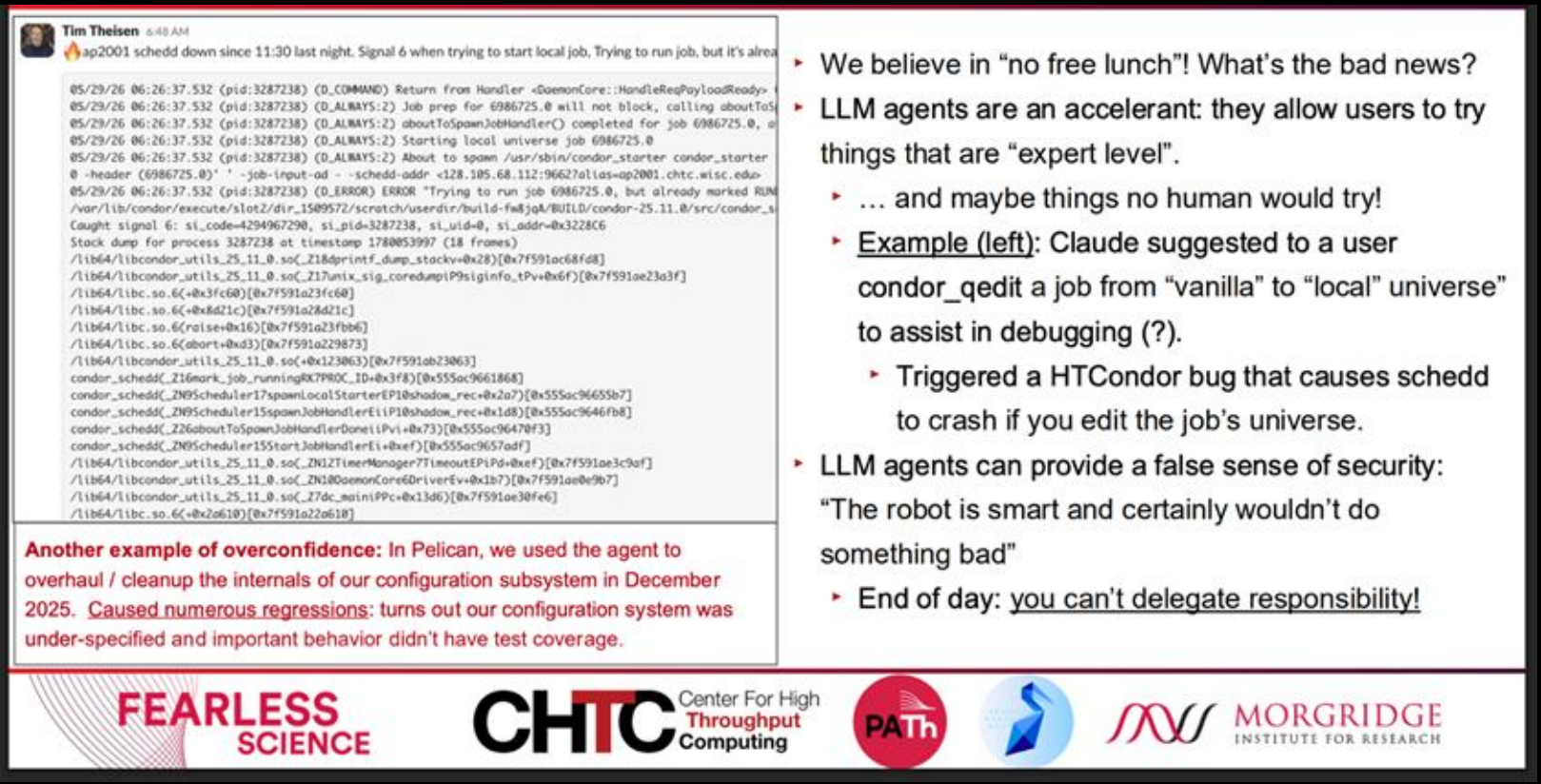
Process and analyze LSST data with Jupyter notebooks in the cloud

Learn more about notebooks.

APIs

Learn how to programatically access data with Virtual Observatory interfaces


2024 - …



Slide: Brian Bockelman's talk at HTCondorWeek

PyVO

99% of all queries to the TAP server from outside the RSP

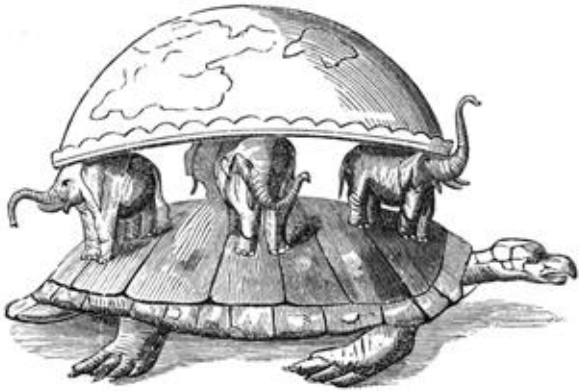
1%

Claude reads the documentation!

- Frossie Economou

Ask me now or during the break about ways in which we use AI in operations:

- Era 1. Planned and completed in construction:
  - Automated Feature-Based Telescope **Scheduler** (*Lynne Jones; ljones.uw@gmail.com*)
  - Real-Bogus **classification** of alerts (*Tatiana Acero Cuellar; taceroc@udel.edu)*
  - **Deblending** overlapping galaxies (*Fred Moolekamp; fred.moolekamp@gmail.com*)



wikipedia

- Era 2. Foundation Model / Gen AI (Operations WIP):
  - Philosophy on using **Claude Code** for production code
  - **MCP Server** for debugging HTCondor workflows (*Jim Chiang; jchiang@slac.stanford.edu*)
  - **PSF modeling** (*Pierre Francois Leget; leget@astro.princeton.edu*)
  - **Weather prediction** at the summit (*Johnny Esteves; jesteves@fas.harvard.edu*)
  - **Wave front estimation** for active optics (*John Franklin Crenshaw; jfc20@uw.edu*)
  - RAG Chatbot for **User Support** (*Leanne Guy; lguy@lsst.org)*

MICHAEL D. LEMONICK SCIENCE JUN 23, 2008 12:00 PM

Watching the Skies: Space Is Really Big – But Not Too Big to Map

The LSST will cover the sky every three days. And within the petabytes of information it collects may lurk things nobody has even imagined — assuming astronomers can figure out how to teach their computers to look for objects no one has ever seen.


Born To Die

Resources

yusra@astro.princeton.edu

- Apr 2025: First Photon with LSSTCam
- Jun 2025: Data Preview 1 dp1.lsst.io (ComCam data)
  First Look skyviewer.app
- Feb 2026: Alert Stream Public prompt-products.lsst.io
- Jun 2026: LSST Starts
- Jul 27 2026: **Early** Data Preview 2 dp2.lsst.io coadds and catalogs only

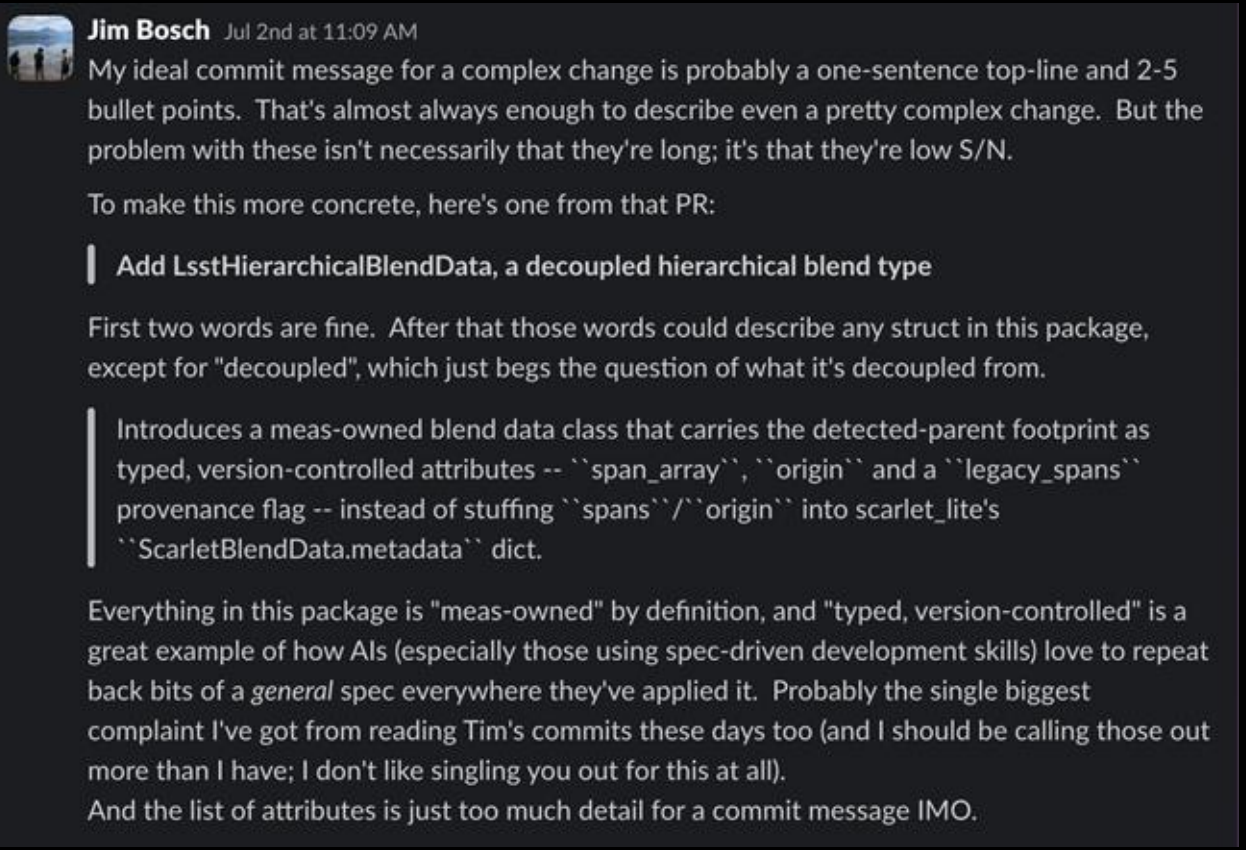
- August 17 2026: For folks in the Mid-Atlantic, come to our open **EDP2 Hack Week** at Princeton and hack along side half of Pipelines team: lsst.princeton.edu/events/2026/rubin-early-data-preview-2-hack-week

Fall 2026 releases:

- Prompt Products Database
- Full Data Preview 2 (adds the `visit_images` and `difference_images`)
- Full Prompt `visit_images` and `difference_images`

Appendix

An analysis of one Claude commit message:



Jim Bosch Jul 2nd at 11:09 AM

My ideal commit message for a complex change is probably a one-sentence top-line and 2-5 bullet points. That's almost always enough to describe even a pretty complex change. But the problem with these isn't necessarily that they're long; it's that they're low S/N.

To make this more concrete, here's one from that PR:

> **Add LsstHierarchicalBlendData, a decoupled hierarchical blend type**

First two words are fine. After that those words could describe any struct in this package, except for "decoupled", which just begs the question of what it's decoupled from.

> Introduces a meas-owned blend data class that carries the detected-parent footprint as typed, version-controlled attributes -- ``span_array``, ``origin`` and a ``legacy_spans`` provenance flag -- instead of stuffing ``spans``/``origin`` into scarlet_lite's ``ScarletBlendData.metadata`` dict.

Everything in this package is "meas-owned" by definition, and "typed, version-controlled" is a great example of how AIs (especially those using spec-driven development skills) love to repeat back bits of a *general* spec everywhere they've applied it. Probably the single biggest complaint I've got from reading Tim's commits these days too (and I should be calling those out more than I have; I don't like singling you out for this at all).
And the list of attributes is just too much detail for a commit message IMO.

Same commit continued

> That dict-based approach (audit finding IO-12) was a cross-package schema split: the keys were written by meas but lived on a scarlet_lite class that neither package could version, an ``updateCatalogFootprints`` crashed with KeyError/TypeError on any blend whose metadata was absent.

An "audit finding" ID has no place here; that's more spec/planning noise.

I'm not sure about the report about what happened before here; that would be appropriate in a fix for a bug we actually saw in production, but if it was hypothetical until your audit I'm not convinced.

> The class inherits directly from ``ScarletBlendBaseData`` -- NOT from scarlet_lite's ``HierarchicalBlendData`` -- and owns an independent schema lineage (``blend_type="lsst_hierarchical"``, schema 1.0.0). It deliberately does not override scarlet_lite's ``hierarchical`` registry entry or migration lineage, so scarlet_lite stays free to evolve ``HierarchicalBlendData`` on its own. The sole contact point is ``convert_from_hierarchical``, a one-time read of the frozen legacy on-disk format wired up by a later commit in this series.

This is mostly about paths not taken. Sometimes that's appropriate, but I don't think AIs generally have good judgement about which paths-not-taken are relevant to mention. We want to mention seemingly obvious alternate paths that are actually traps for the unwary (or that might be worth exploring, but weren't), but AI harnesses and SDD schemas work well in part because they tell the AI to consider so many possibilities in enough detail (and with enough extra context repeated along with them!) that that *substitutes* for actual judgement. And then those possibilities creep into the comments and commit messages and docs that it writes unless you're very vigilant.

> This commit only adds and registers the class; nothing produces or consumes ``lsst_hierarchical`` blends yet, so it is additive and the test suite stays green. The span-mask codec is reused from ``io/source_data.py``.

This is just more information about what could have been done but wasn't; its S/N is similarly low, and I suspect it's another case of the AI basically thinking out loud instead of quietly. (edited)