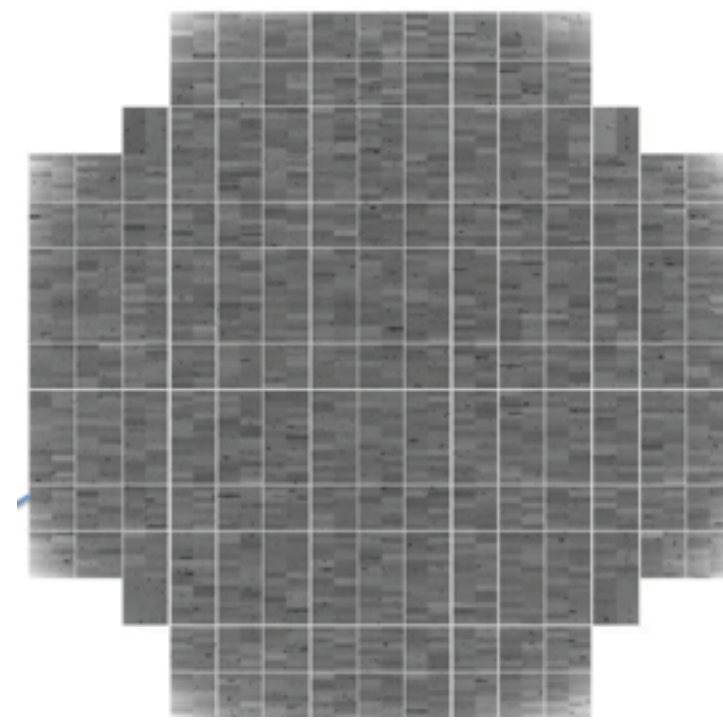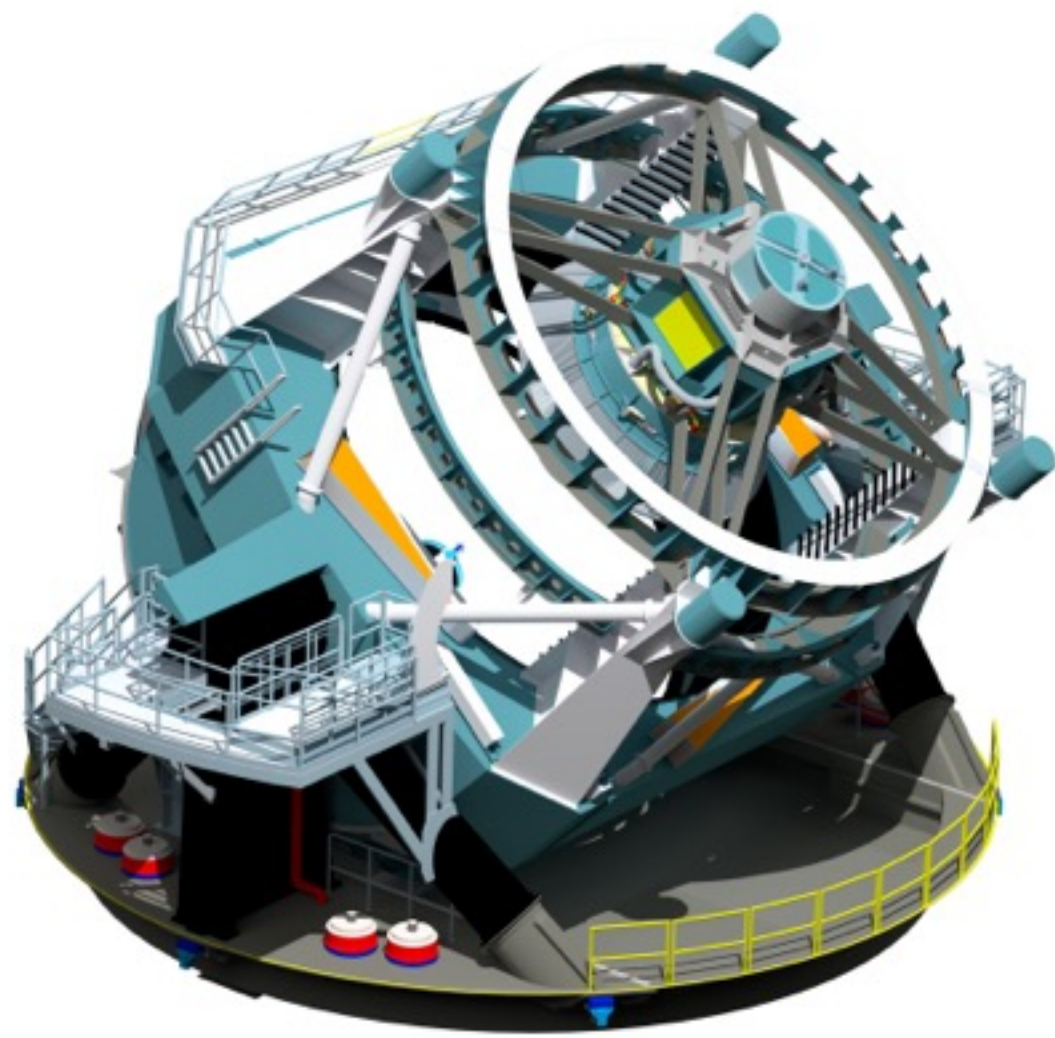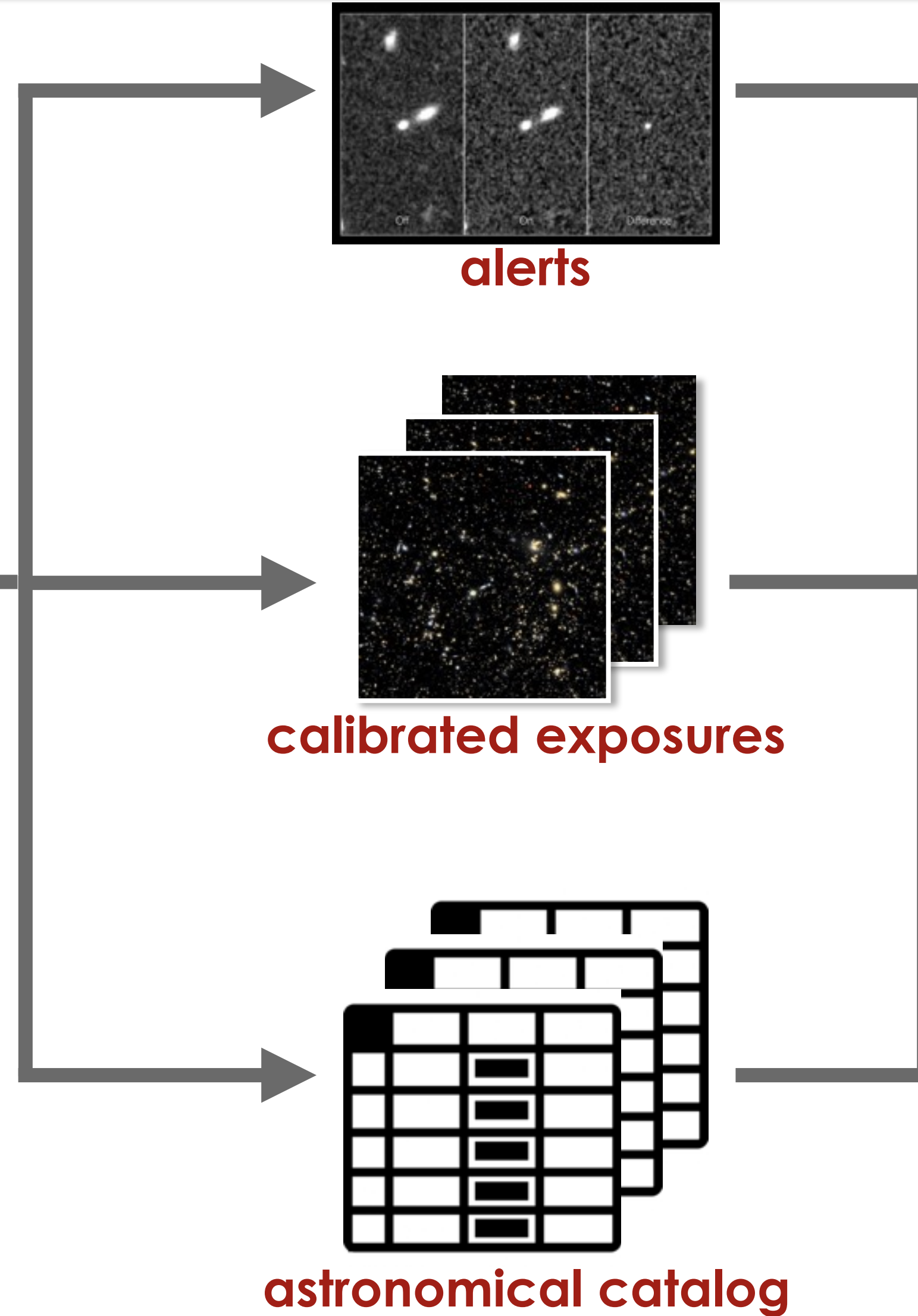# Kubernetes-based services for LSST

fabio hernandez

LARGE SYNOPTIC SURVEY TELESCOPE

raw images

alerts

calibrated exposures

astronomical catalog

scientific discoveries

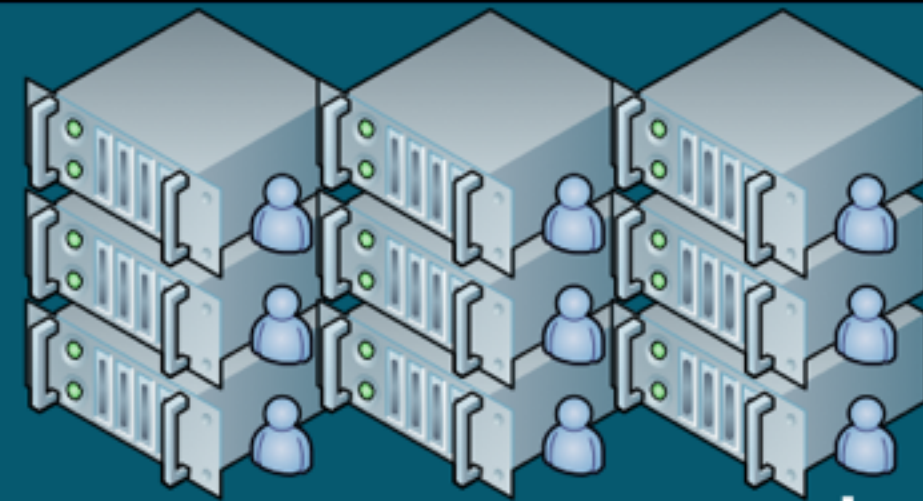**LSST aims to deliver a catalog of 20 billion galaxies and 17 billion stars with their associated physical properties**

ENVISIONED ARCHITECTURE

(preliminary figures)

22k → 122k CPU cores
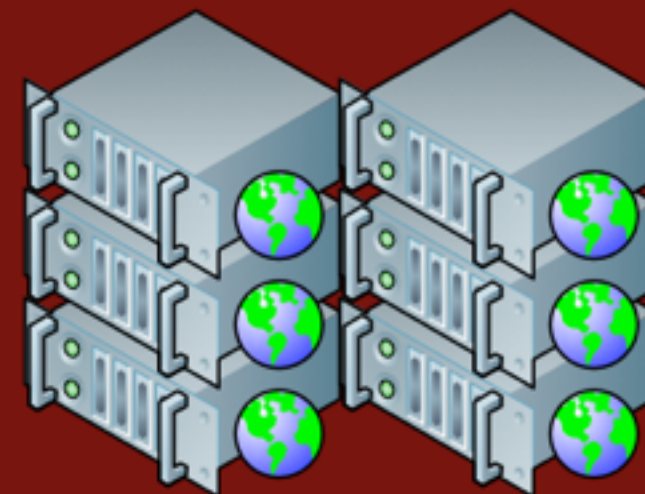
10 → 81 PB
130 → 550 GB/s

10 → 55 PB
63 → 293 GB/s
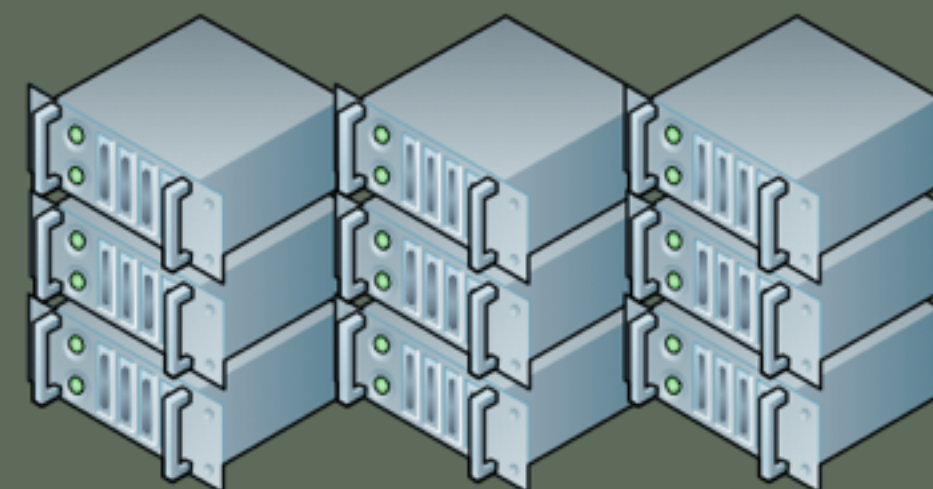
9 → 121 PB
7 → 30 GB/s

2022 → 2032

WAN

LAN

Batch farm

Login farm

Data transfer nodes

Application servers
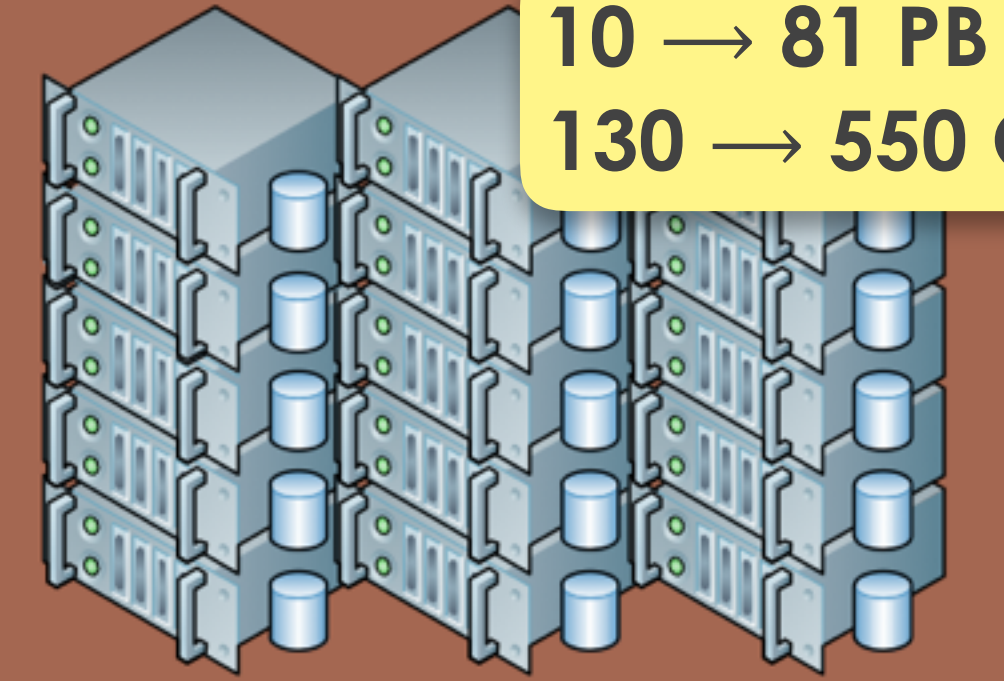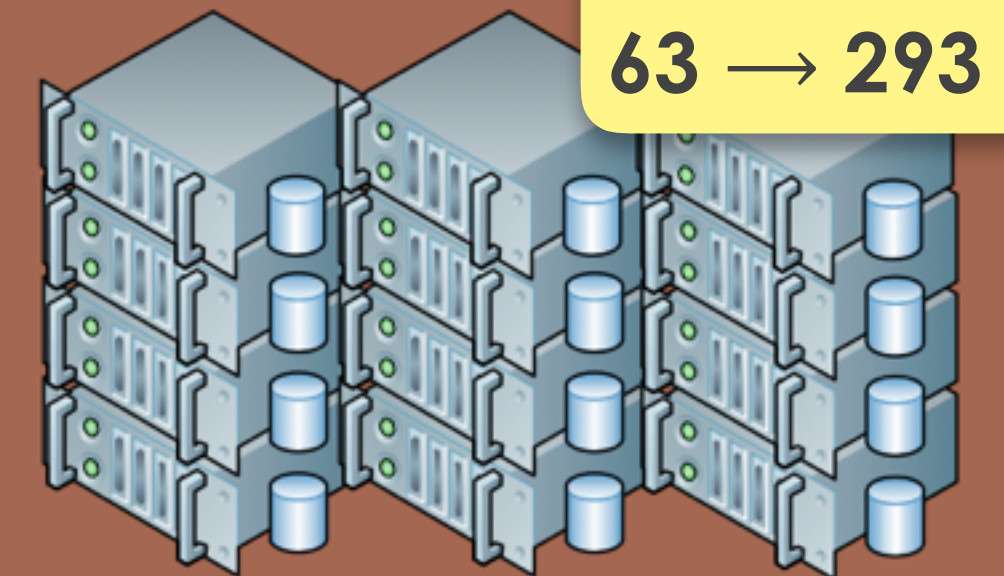
Catalog database

Disk storage

Mass storage

LSST SCIENCE PLATFORM

LSST Users

Internet

LSST Science Platform

Portal	JupyterLab	Web APIs

Data Releases	Alert Streams	User Databases	User Files	User Computing	Software Tools
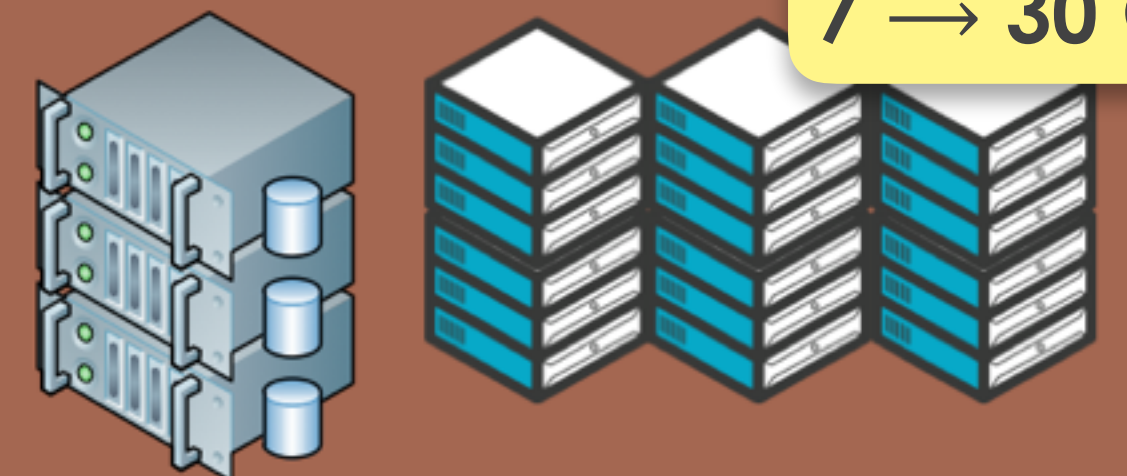
Source: M. Juric, LSST
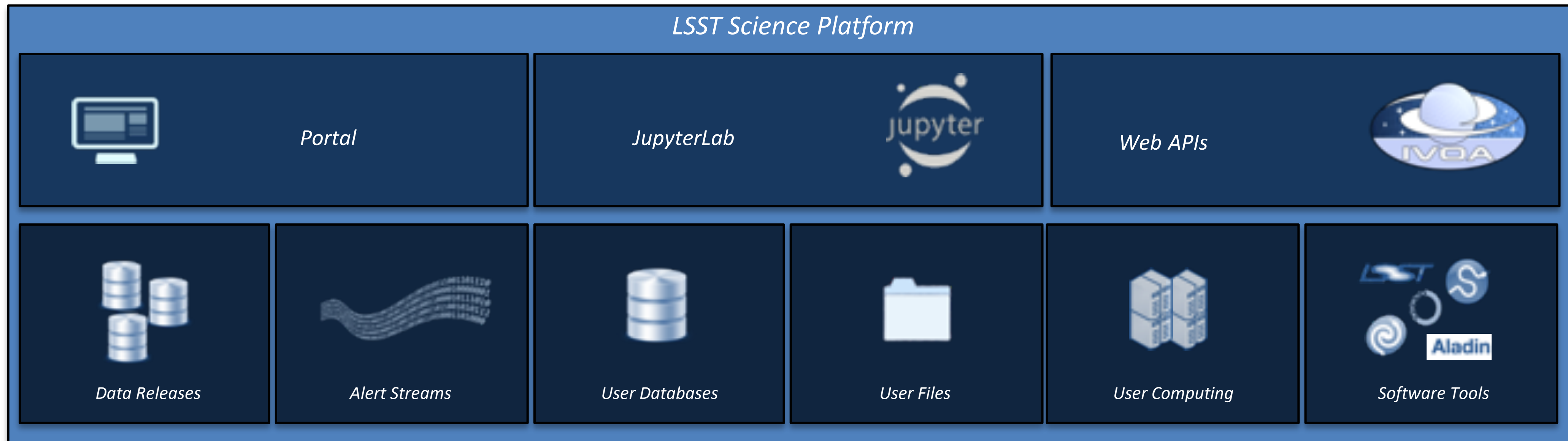
Set of integrated web applications and services, through which the scientific community will **access**, **visualize**, **subset** and perform **next-to-the-data analysis** of the data

USE CASES

○ Software for the LSST science platform is packaged as containers, to be run by Kubernetes

examples: catalog database, notebook service, visualisation service

mainly on bare metal, but exercises of the catalog database ongoing on Google cloud

CATALOG DATABASE

○ Qserv

○ Running under Kubernetes on bare metal at CC-IN2P3

   several instances, including one for developers and one for scientists

   on bare metal

○ Qserv at CC-IN2P3 will likely run on bare metal, on a set of dedicated servers

   appropriately configured as database servers

NOTEBOOK SERVICE

○ Currently using nodes in the login farm

notebook server launched by the end user through a SSH tunnel

server is destroyed when SSH session is closed, either normally or abnormally

no elasticity

no dedicated servers

○ Foreseen evolution

creation of a customised Jupyter notebook server and visualisation server for each user, on demand

likely on bare metal on servers well configured for the these roles

access from the container to datasets storage (i.e. /sps/lsst) and notebook storage (/pbs/home and /pbs/throng/lsst) required

could benefit of a generic Kubernetes service

Notebook Service (cont.)

○ Benefits

increasing or decreasing the number of containers following demand

make easier resuming a work session