Connecting Science Platforms

Jesús Salgado

IVOA GWS chair and
SKA Regional Centres Network Architect

Open Science and AI

Harmonisation
Transparent Data Access
Combined Computing Resources

Science Enabling Applications

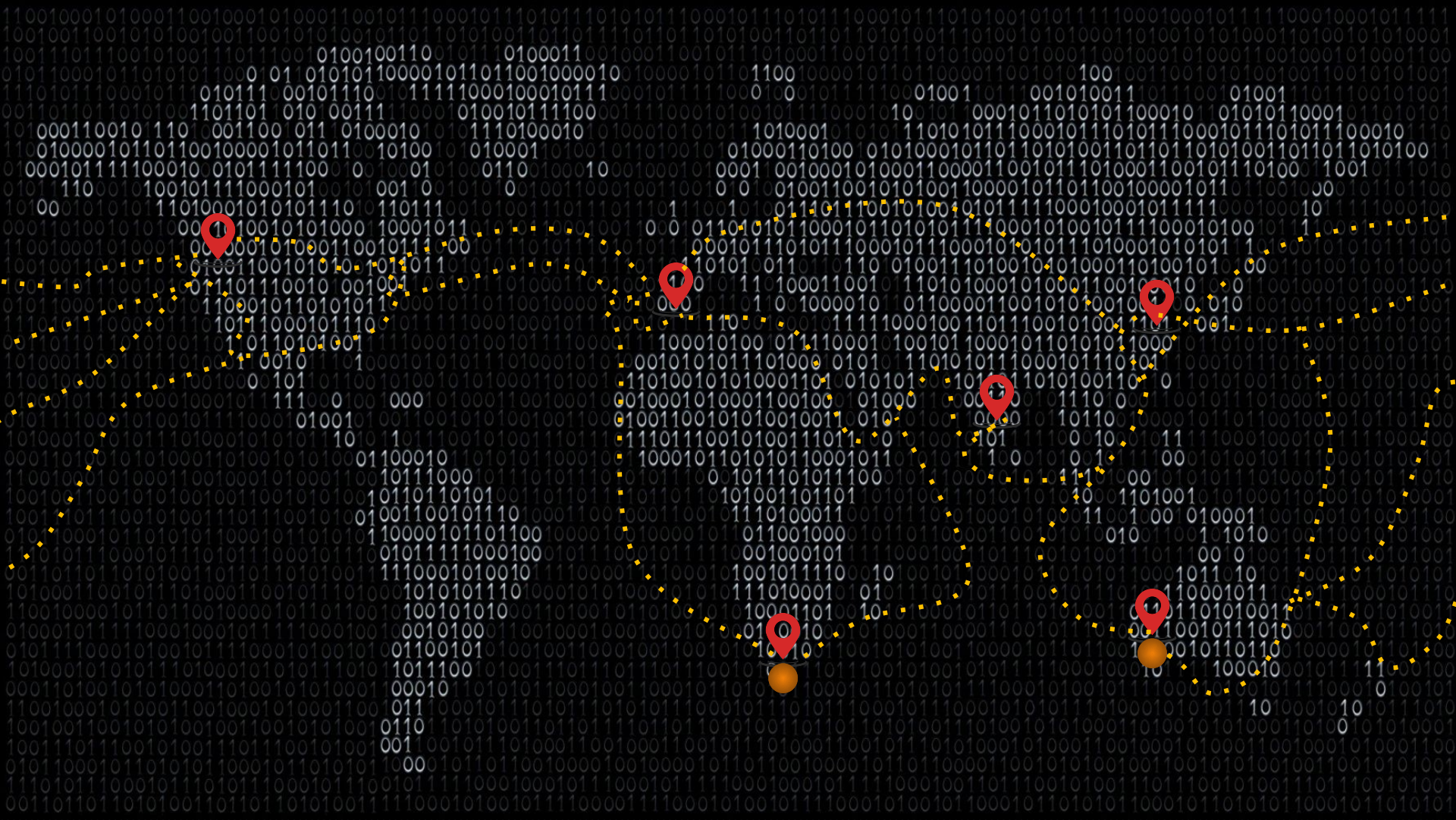
Astropy and Astroquery
Notebooks
Users environments

Discovery And Access Services

Cone Search
SSAP, SIAP
TAP

InterOperability and Federation

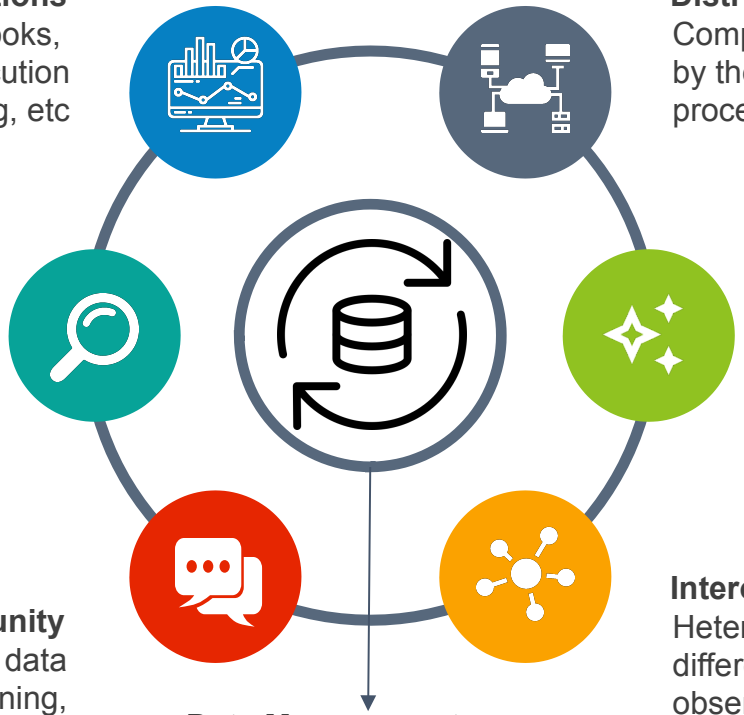
Federated Authentication and Distributed Processing Platforms interconnected Data Lakes



SKA Regional Centre Capabilities Blueprint

Science Enabling Applications
Analysis Tools, Notebooks, Workflows execution Machine Learning, etc

Distributed Data Processing
Computing capabilities provided by the SRCNet to allow data processing



Data Discovery
Discovery of SKA data from the SRCNet, local or remote, transparently to the user

Visualization
Advanced visualizers for SKA data and data from other observatories

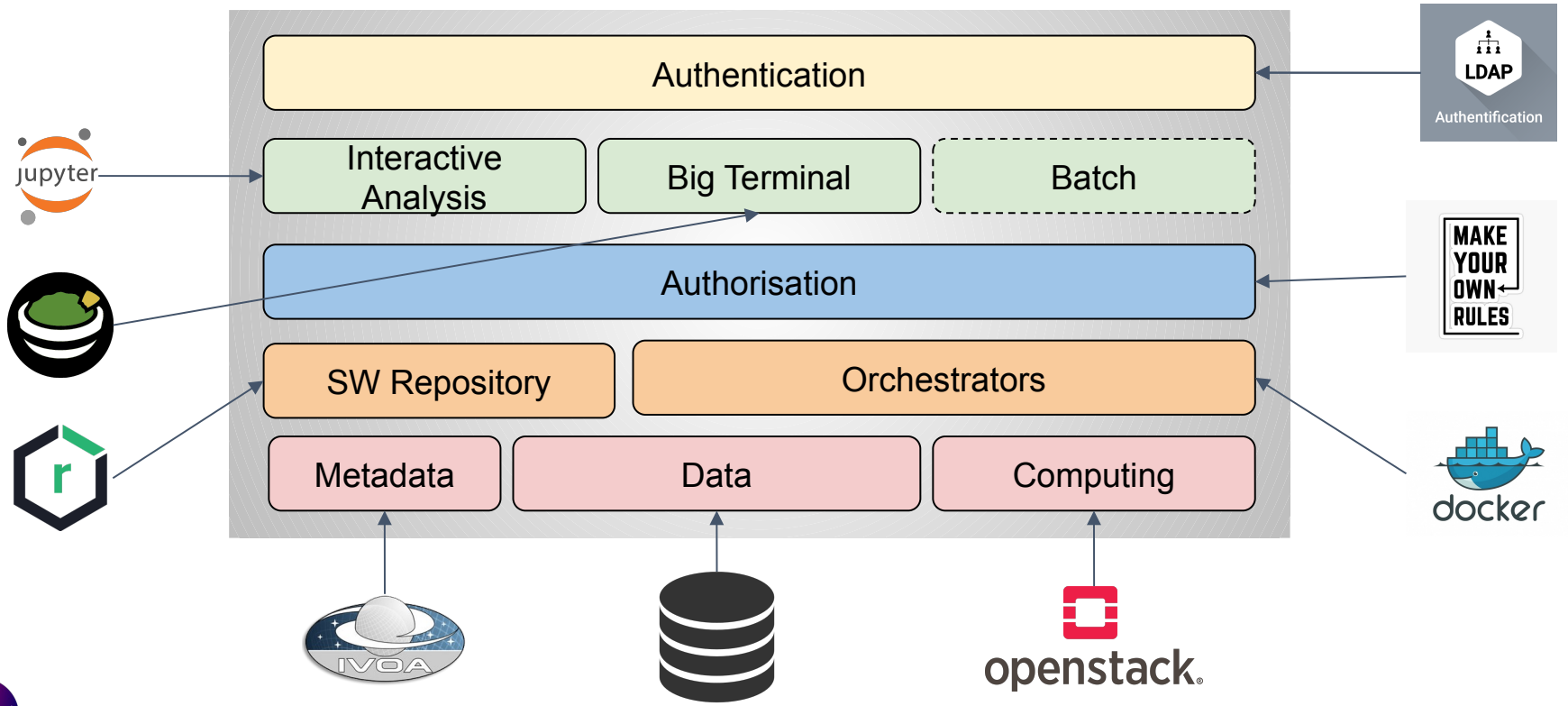
Support to Science Community
Support community on SKA data use, SRC services use, Training, Project Impact Dissemination

Interoperability
Heterogeneous SKA data from different SRCs and other observatories

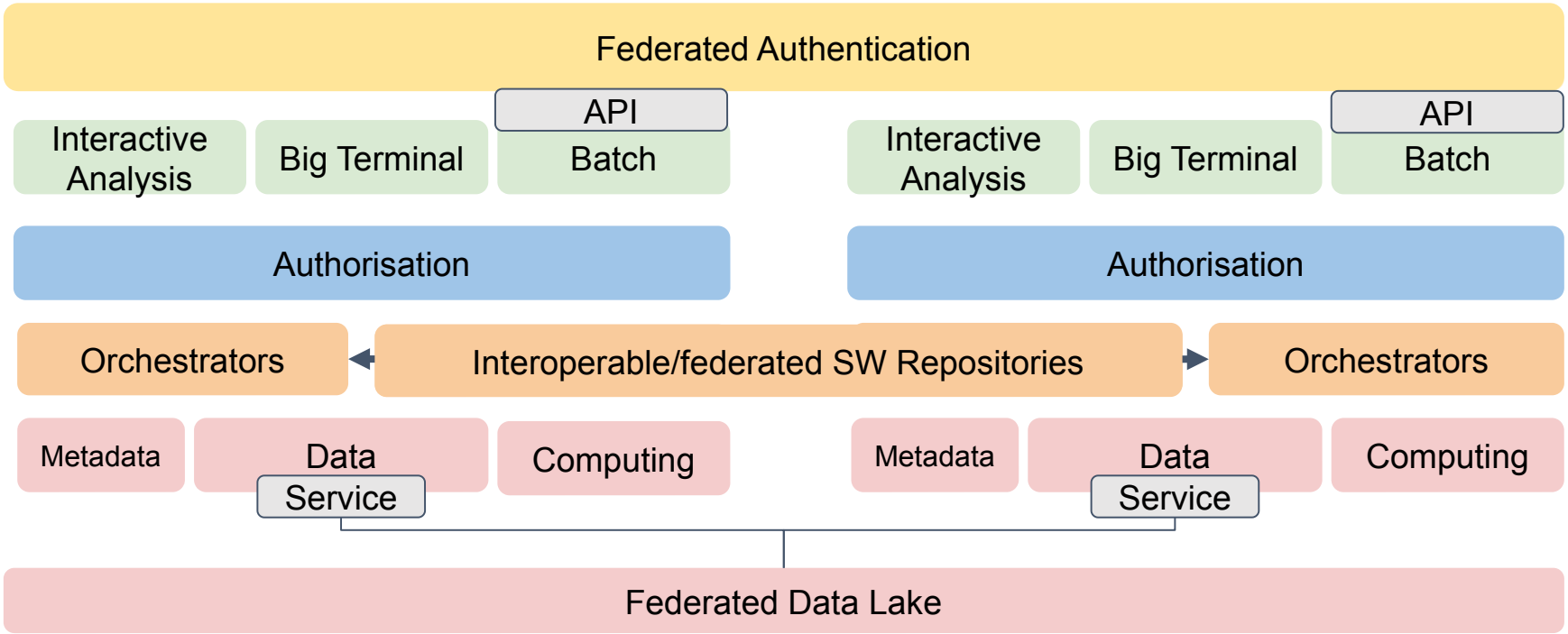
Data Management
Dissemination of Data to SRCs and Distributed Data Storage

SRC Network global capabilities

Science platforms

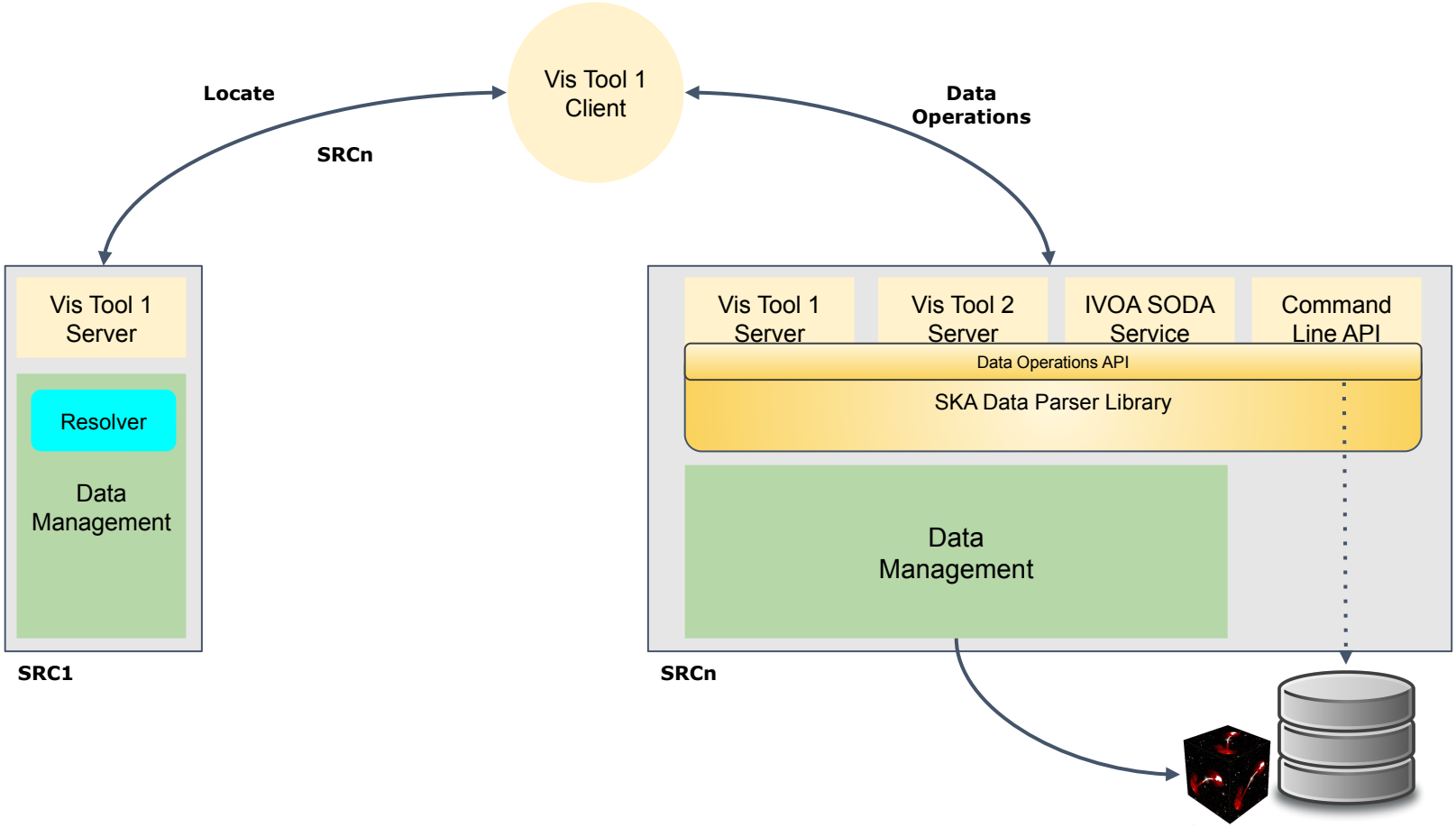


Science Platforms Interoperability

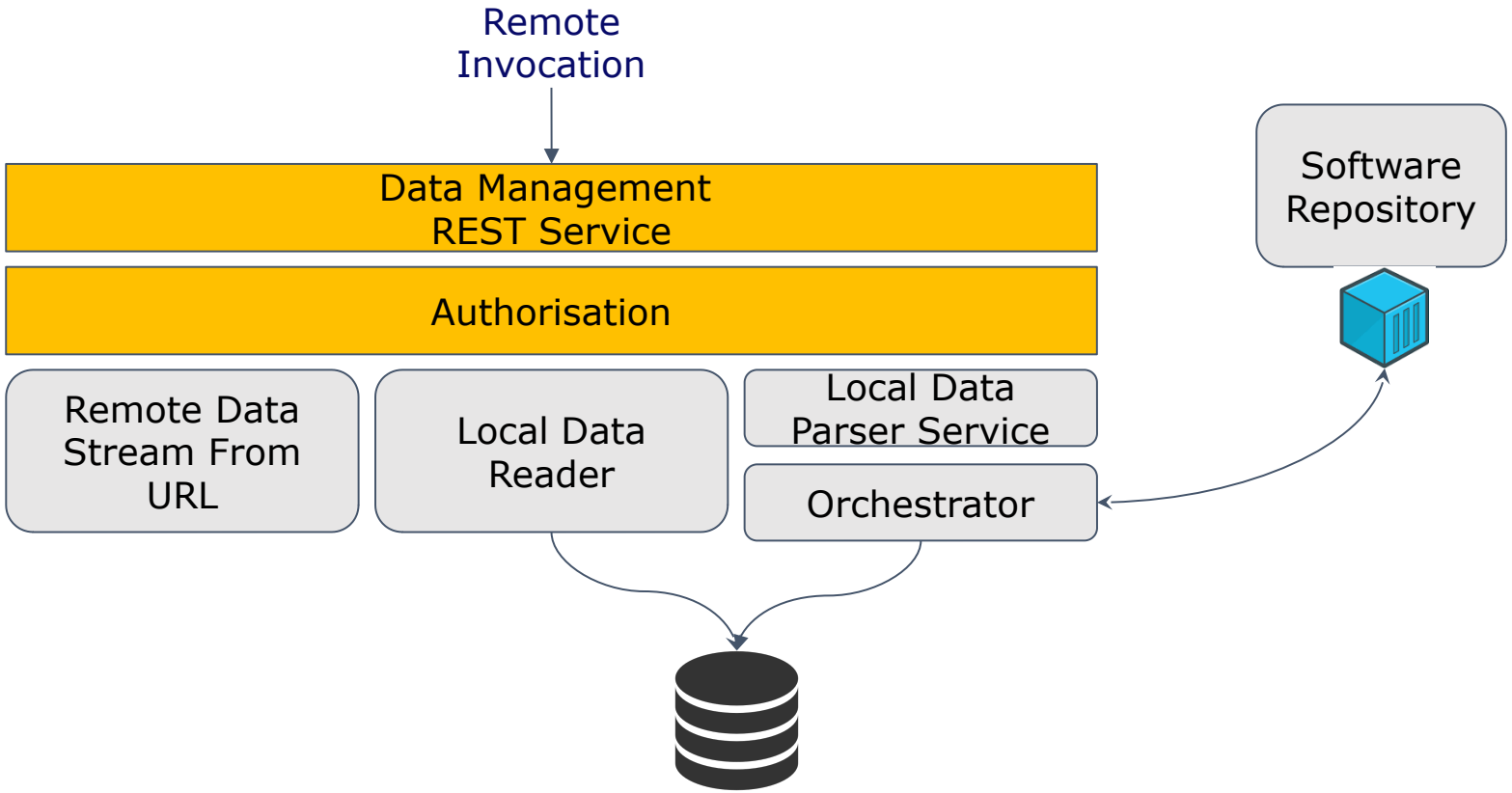


Federated Execution

Visualisation API



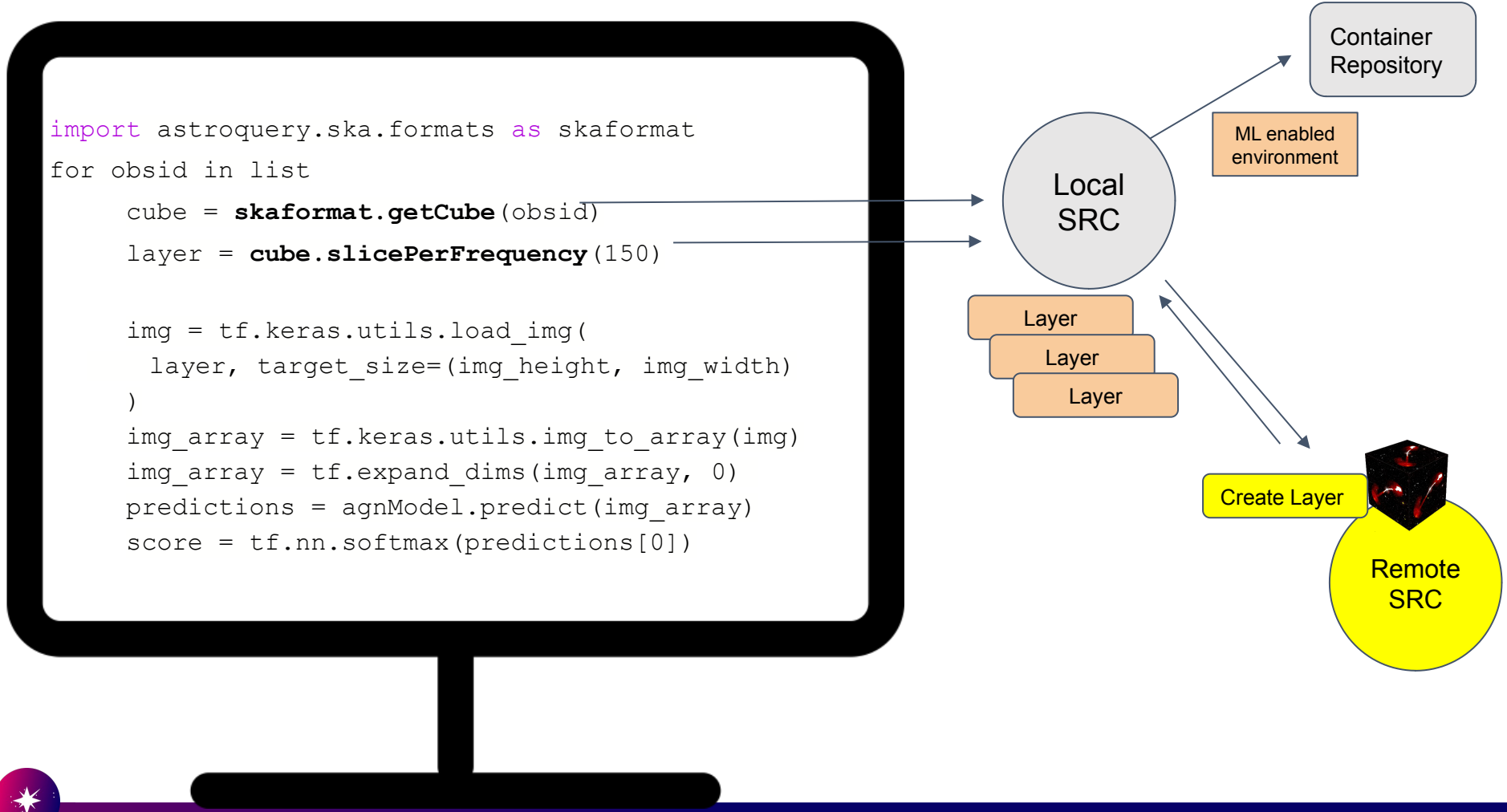
Remote “Data Parsing” Operation



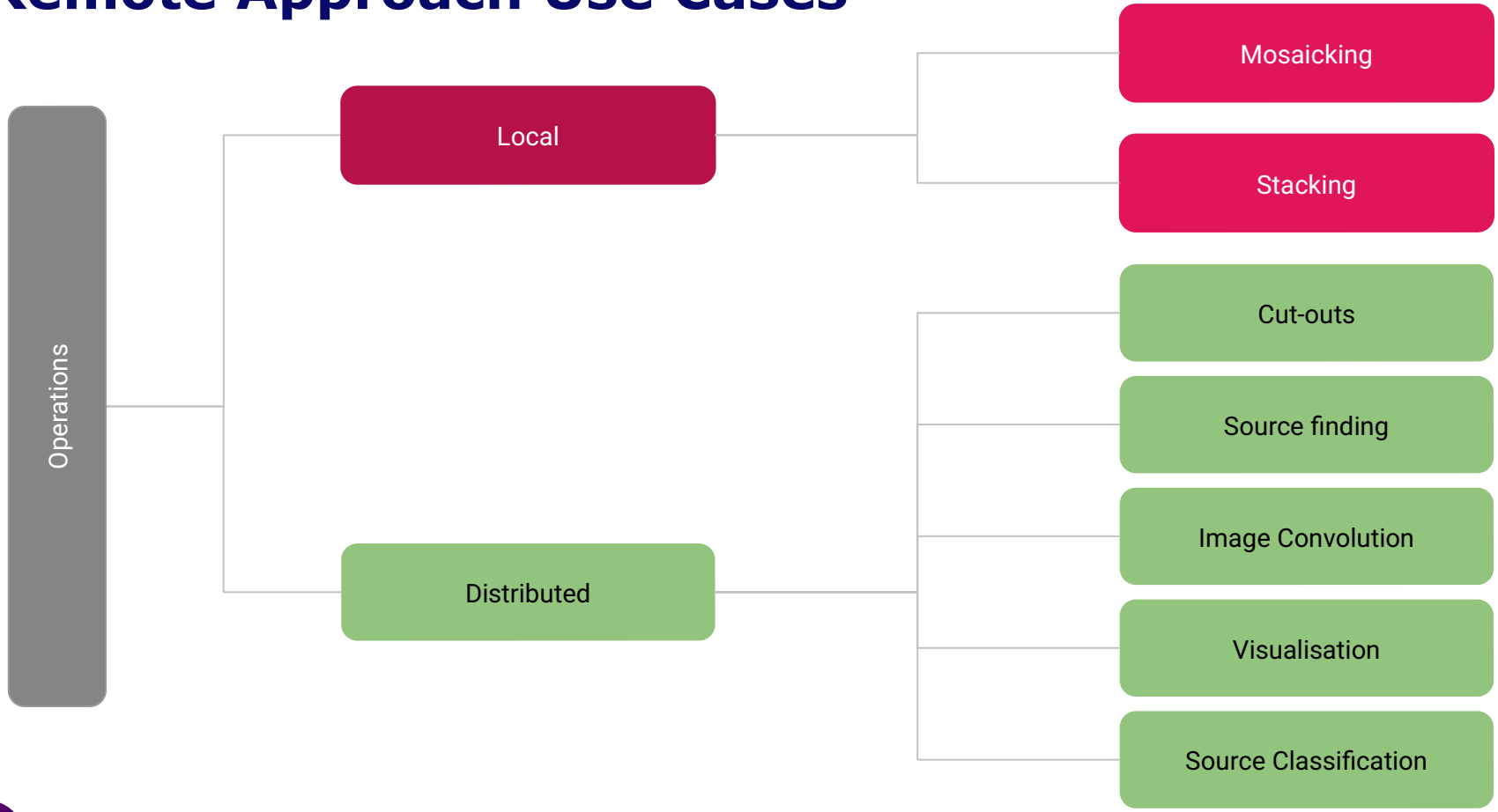
Some possible data mesh services

Data Type	Operation	Input	Output
Any Type	Get Stream	ID	Input Stream
Data Cube	Cut-out	ra, dec, size, resolution	Data Cube
Data Cube	Get Spectra	ra, dec, size	Spectrum
Data Cube	Get Time Series	ra, dec, size	Time Series
Data Cube	Get Slice	wavelength	Image
Image	Change Resolution	ra, dec, size, resolution	Image (FITS to HiPS)
Image	Source Extraction	ID, algorithm params	Source Catalogue
Spectrum native	Convert to VO	ID	Spectrum VO
Source Catalogue	Similar Source	Source ID	Source Catalogue

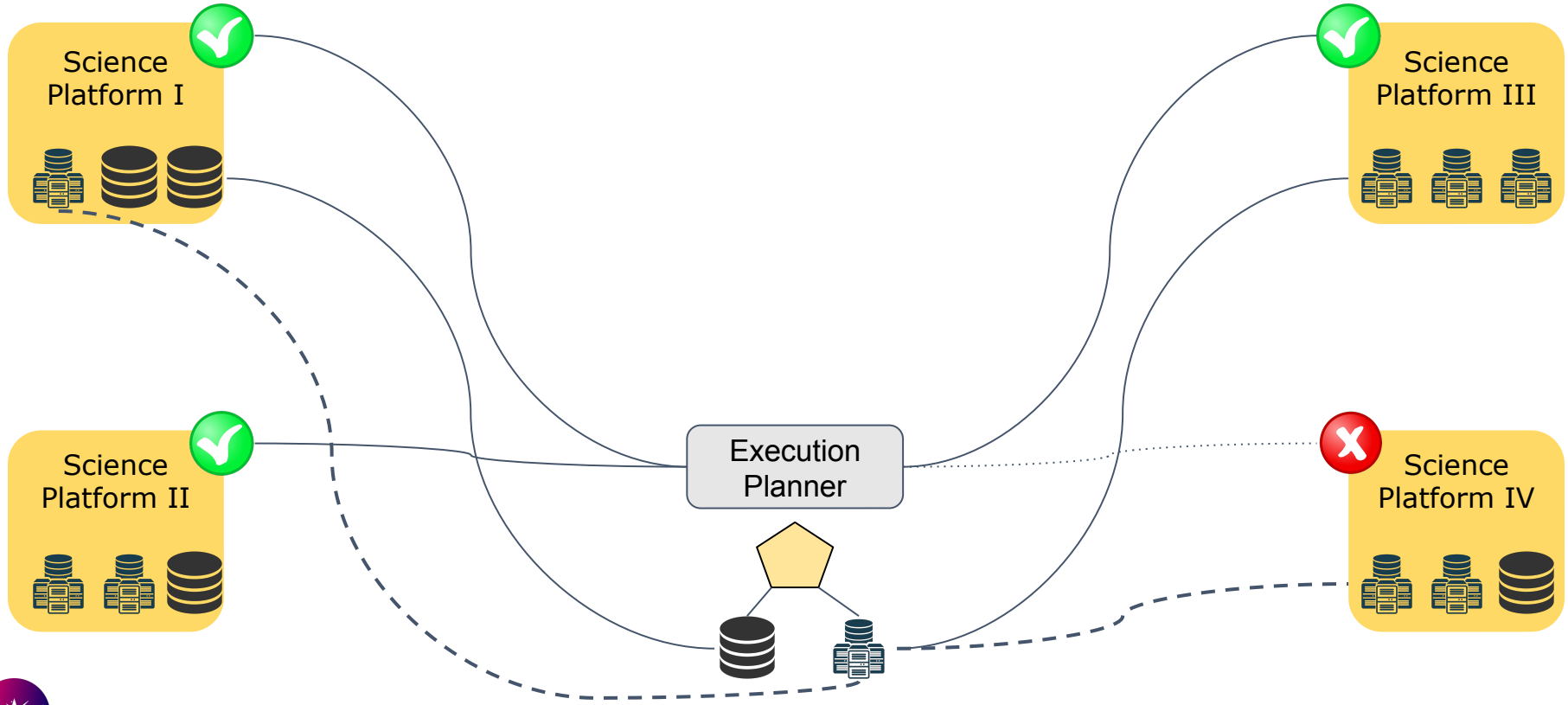
```
import astroquery.ska.formats as skaformat
for obsid in list
      cube = skaformat.getCube(obsid)
      layer = cube.slicePerFrequency(150)

      img = tf.keras.utils.load_img(
        layer, target_size=(img_height, img_width)
      )
      img_array = tf.keras.utils.img_to_array(img)
      img_array = tf.expand_dims(img_array, 0)
      predictions = agnModel.predict(img_array)
      score = tf.nn.softmax(predictions[0])
```



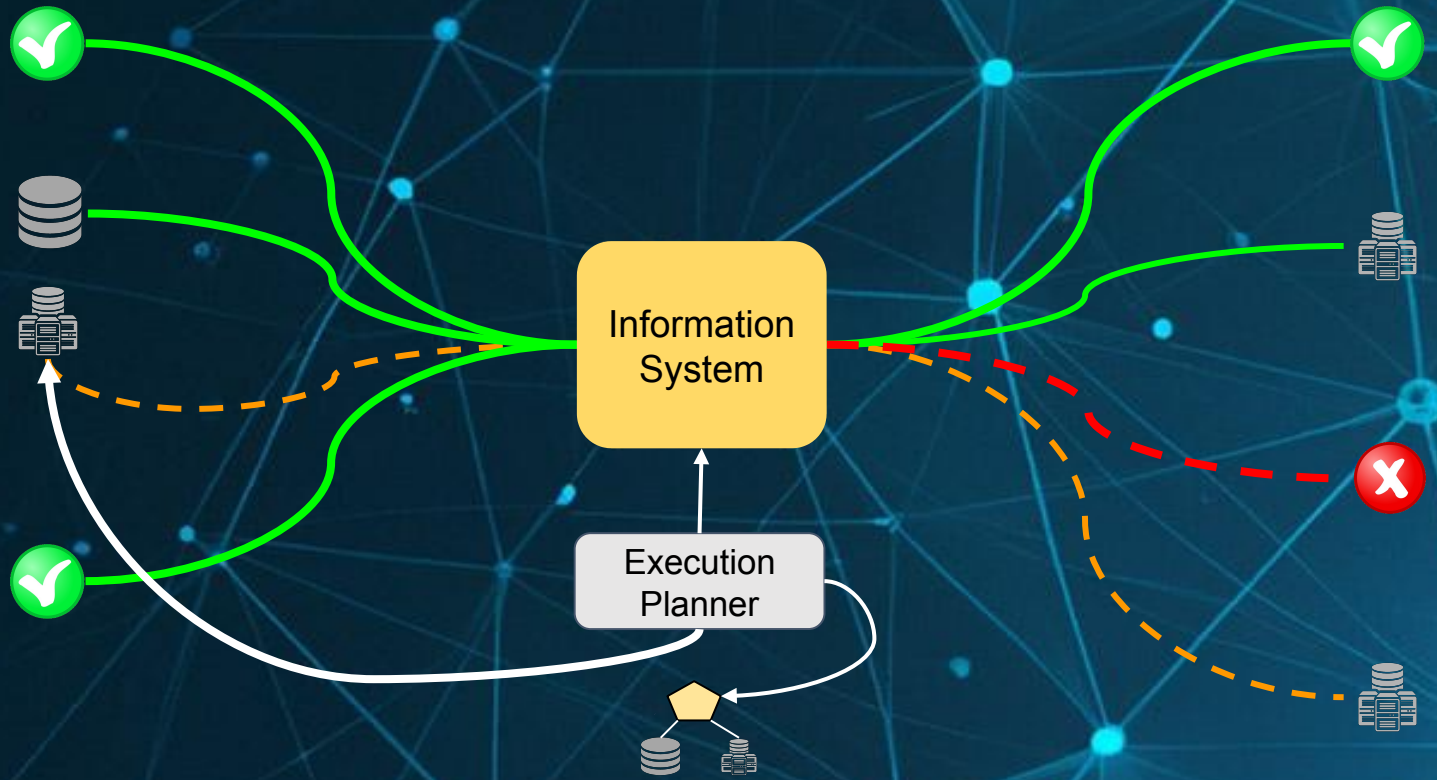
Remote Approach Use Cases

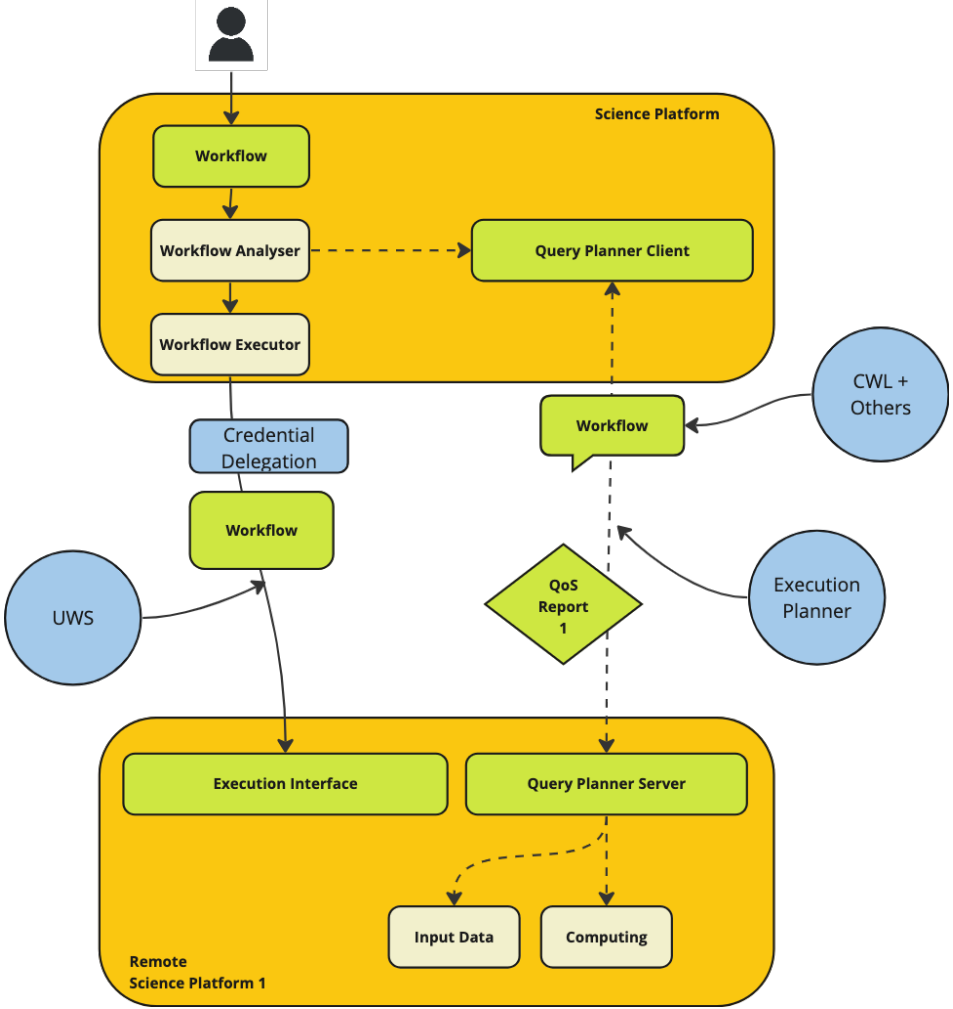


Execution Planner



Solving the Topology




Science Platform
Workflow
Workflow Analyser
Query Planner Client
Workflow Executor
CWL +
Others
Workflow
Credential
Delegation
Workflow
UWS
QoS
Report
1
Execution
Planner
Execution Interface
Query Planner Server
Input Data
Computing
Remote
Science Platform 1

Summary

- Possible “interoperable science platform” new phase with:
  - Federated Authentication Protocols
  - Improved data access
  - Remote operations
  - (Simplified) federated execution
    - Execution planner
    - Topologies
    - Software characterisation
    - Workflows

PROMOTE/ENDORSE

NEW? NEEDED?

EXTEND?

NOTE TO STANDARD

DM INSIDE EXEC.PLAN.?

NEW? NEEDED?

- Today objectives
  - Agree on roadmap
  - Identification of interested parties

Thanks for your attention