Remote Visualization of Big Data: VisIVO as a Visualization Prototype for SKA Regional Centres

Giuseppe Tudisco*, Fabio Vitello, Eva Sciacca, Cilliers Pretorius
giuseppe.tudisco@inaf.it
INAF - Osservatorio Astrofisico di Catania

INAF USCVIII - Calcolo Critico
16/06/2023

SKA Regional Centres (SRC)

Each year the SKAO will generate around 700 PB of data.

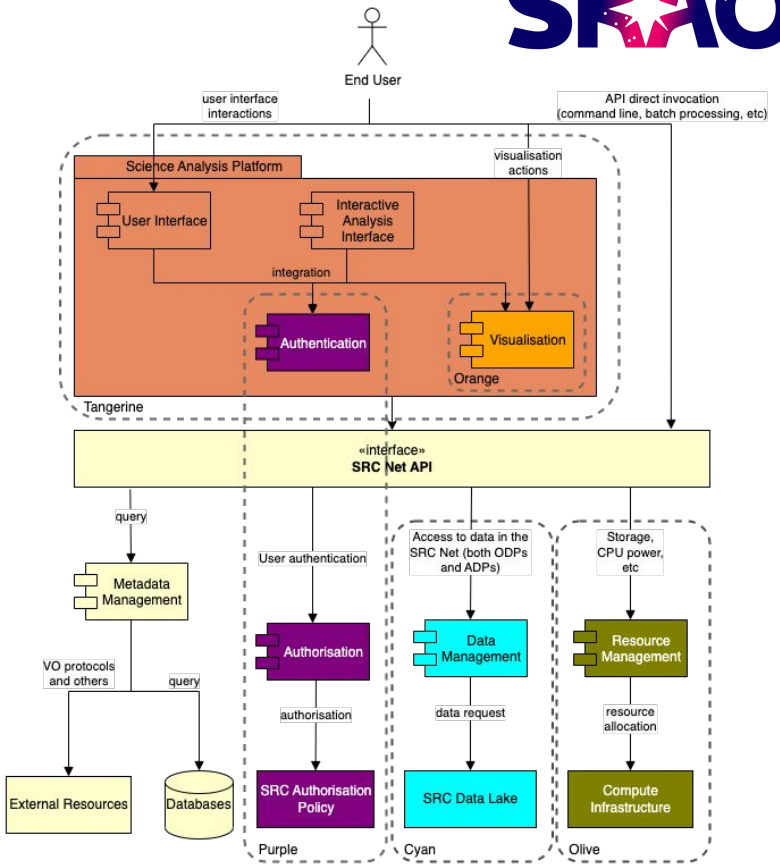
The data volumes are so large that direct delivery to end users is not practical.

Functionalities to find, access, manipulate and visualize SKA data products needs to be made available on shared computational resources.

An SKA Regional Centres (SRC) is a virtual entity that provides:

- access to data products
- platforms for advanced scientific research
- a place for development of software tools (analysis, modelling, visualization)

A global network of SRCs distributed around the world is being developed by SRCNet Agile Teams.



INAF contribution in Orange Team

Started working on Prototype 4: Visualization in PI15 (June 2022).

- Visualization Tools review (dependencies, interfaces)
- Contributing to the definition of visualization use cases for SRCNet
- Collection of data products and data formats from precursors and pathfinders
- Testing and deployment of visualization tools and data access services into SRC nodes
- Adapting Visualization Tools to address use cases and work with SRC architecture and its data lake

Given Name	Family Name	Role
Fabio	Vitello	Product Owner
Giuseppe	Tudisco	Scrum Master
Eva	Sciacca	Team Member
Andrea	Lorenzani	Team Member
Alessandra	Zanichelli	Team Member
Vincenzo	Galluzzi	Team Member
Marco	Molinaro	Team Member
Robert	Butora	Team Member

SRCNet Visualisation Use Cases

As a user visualising data, I want to be able to:

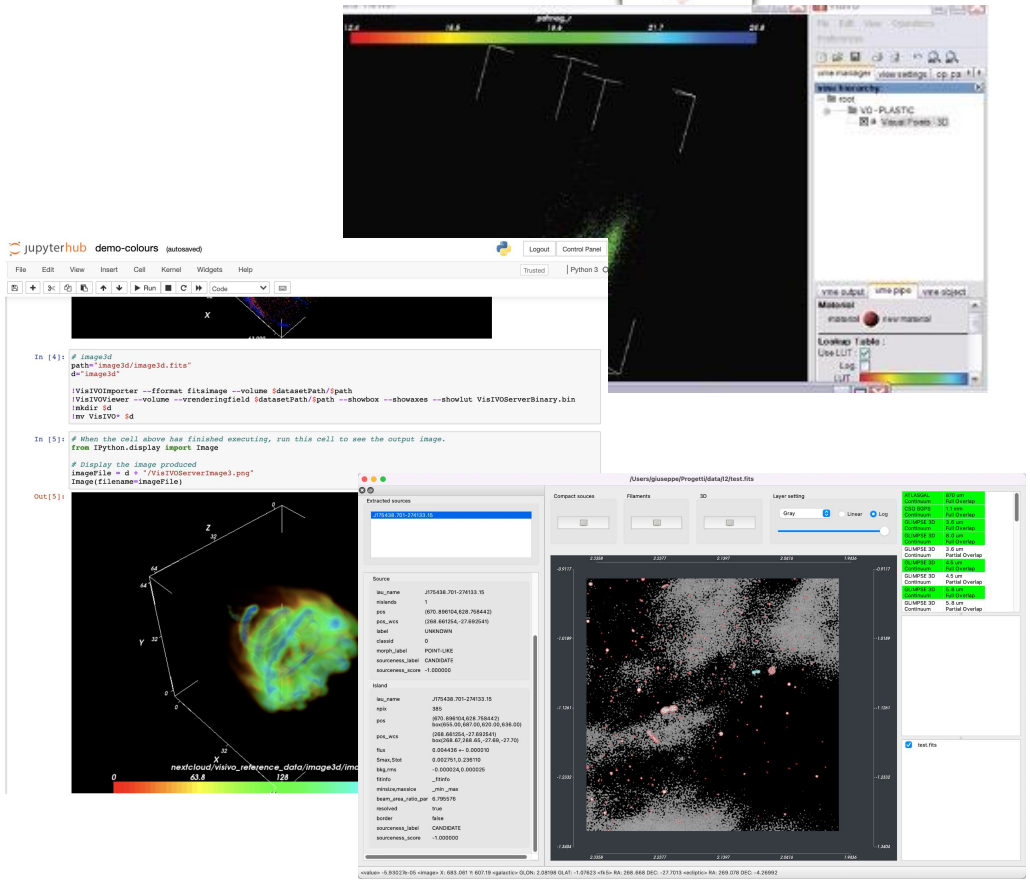
- visualise image and catalogue data in an interactive way, including overlaying other wavelength images and source lists
- visualise spectra and time series, either through beamformed observations or at locations in an image cube
- split out subsets of datasets by selecting sky areas in images or freq/time sections in a spectrum or time series plot
- fit profiles to sources or emission lines
- perform mathematical operations per pixel for analysis including polarisation, spectral index, emission lines or temperatures
- create publication quality plots such that it is reproducible by accessing the code (python, matplotlib)

VisIVO Framework

The VisIVO Framework (Visualization Interface for the Virtual Observatory) is a set of tools and services for multi-dimensional data analysis, maintained by Astrophysical Observatory of Catania.

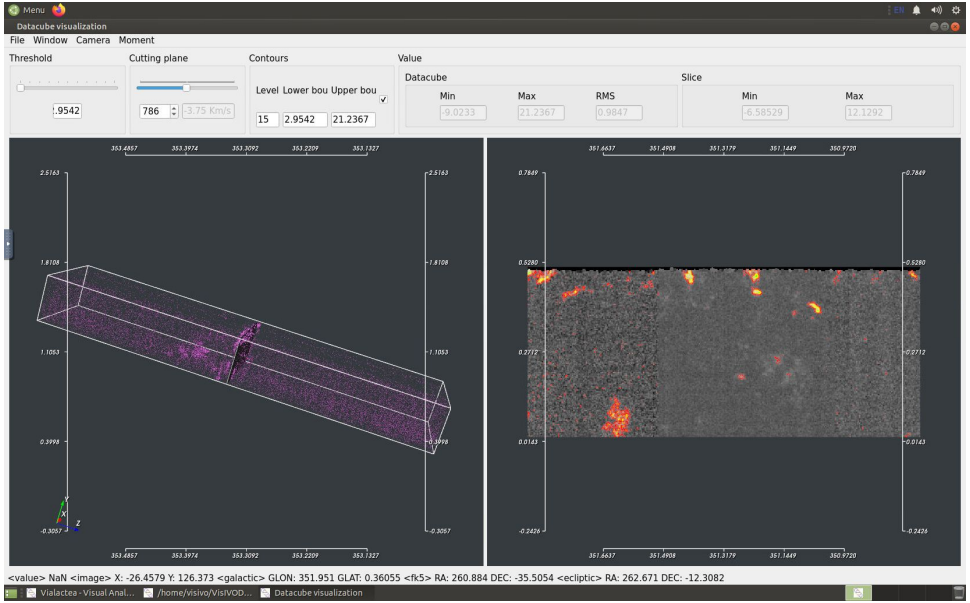
VisIVO Framework consists of:

- **VisIVO Server,** a platform for high performance visualization.
- **VisIVO Gateway**, VisIVO Server on a Jupyter Notebook.
- **VisIVO ViaLactea Visual Analytics**, provides a visual analytic environment to analyze the correlation between different data, for example 2D intensity images with 3D molecular spectral cubes.



VisIVO ViaLactea Visual Analytics

- Development started in 2015
- Open-source desktop application that offers a visual analytics environment to conduct research activities on two-dimensional regions of space and three-dimensional datacubes.
- Official client of the VLKB which includes 2D and 3D surveys, numerical model outputs, point-like and diffuse object catalogues.



DOI 10.5281/zenodo.8010708

https://github.com/VisIVOLab/ViaLacteaVisualAnalytics

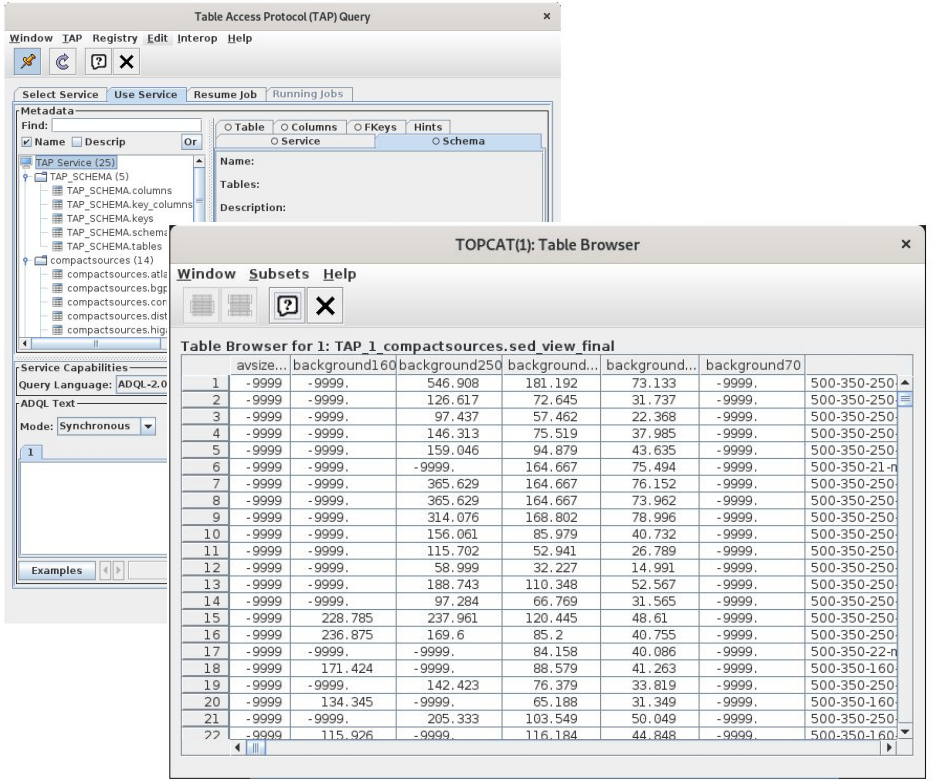
ViaLactea Knowledge Base (VLKB)

The ViaLactea Knowledge Base (VLKB) identifies a set of data collections, catalogues and services enabling discovery and access on them.

It consists of:

- a TAP service to deploy the catalogue data contents, and
- a dataset search, cutout and merge combined service API for accessing 3D radial velocity cubes and 2D images.

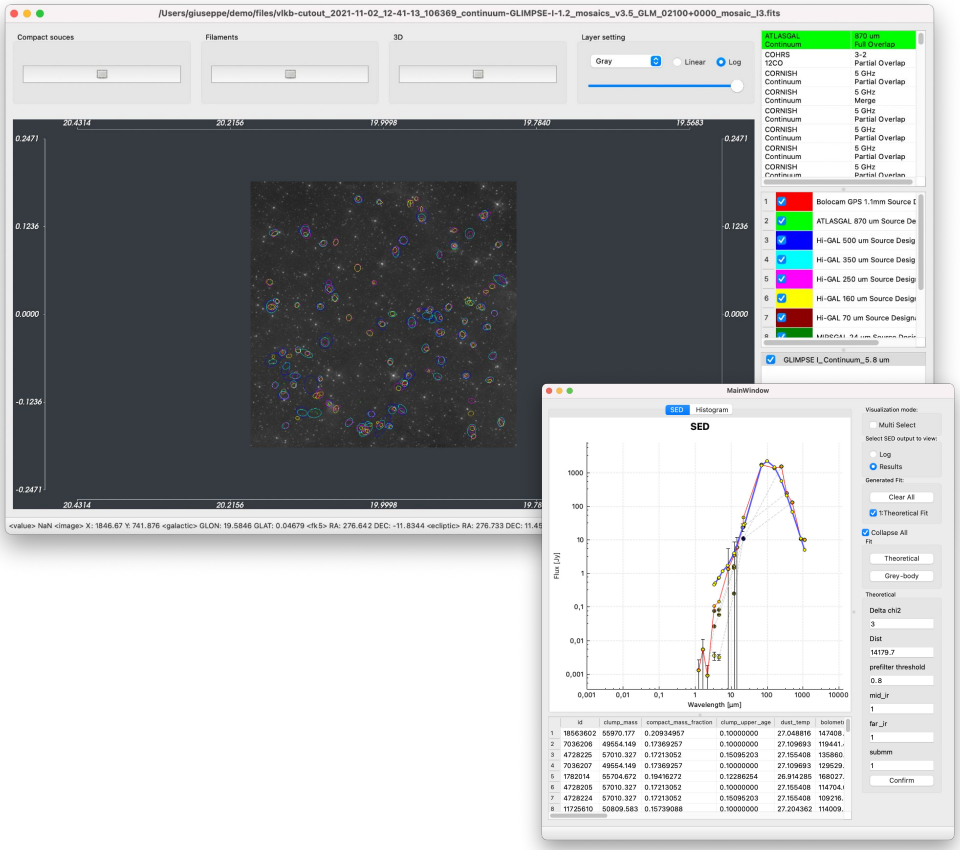
Orange Team is also working on adapting these server-side services for data access to work within SKA data lake architecture.



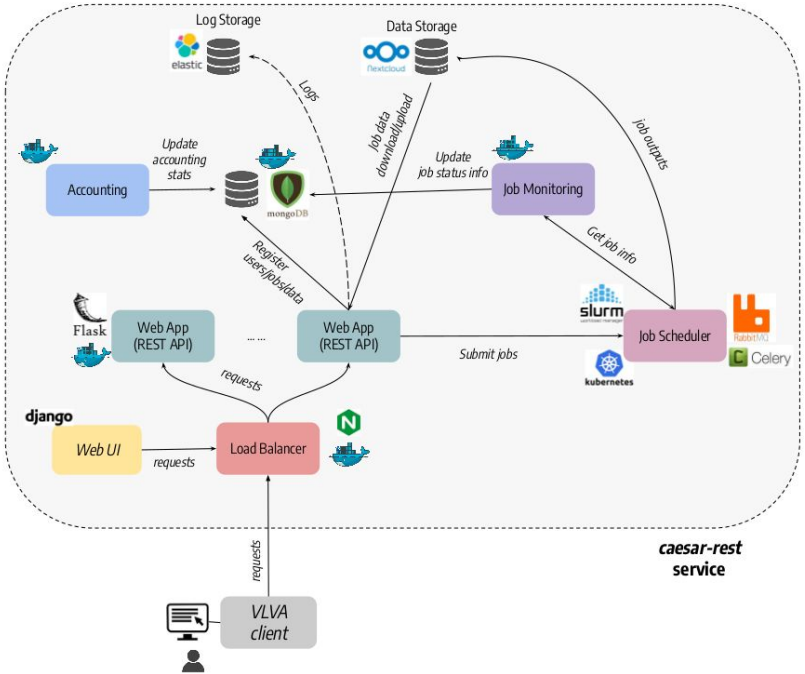
	avsize...	background160	background250	background...	background...	background70	
1	-9999	-9999.	546.908	181.192	73.133	-9999.	500-350-250
2	-9999	-9999.	126.617	72.645	31.737	-9999.	500-350-250
3	-9999	-9999.	97.437	57.462	22.368	-9999.	500-350-250
4	-9999	-9999.	146.313	75.519	37.985	-9999.	500-350-250
5	-9999	-9999.	159.046	94.879	43.635	-9999.	500-350-250
6	-9999	-9999.	-9999.	164.667	75.494	-9999.	500-350-21-n
7	-9999	-9999.	365.629	164.667	76.152	-9999.	500-350-250
8	-9999	-9999.	365.629	164.667	73.962	-9999.	500-350-250
9	-9999	-9999.	314.076	168.802	78.996	-9999.	500-350-250
10	-9999	-9999.	156.061	85.979	40.732	-9999.	500-350-250
11	-9999	-9999.	115.702	52.941	26.789	-9999.	500-350-250
12	-9999	-9999.	58.999	32.227	14.991	-9999.	500-350-250
13	-9999	-9999.	188.743	110.348	52.567	-9999.	500-350-250
14	-9999	-9999.	97.284	66.769	31.565	-9999.	500-350-250
15	-9999	228.785	237.961	120.445	48.61	-9999.	500-350-250
16	-9999	236.875	169.6	85.2	40.755	-9999.	500-350-250
17	-9999	-9999.	-9999.	84.158	40.086	-9999.	500-350-22-n
18	-9999	171.424	-9999.	88.579	41.263	-9999.	500-350-160
19	-9999	-9999.	142.423	76.379	33.819	-9999.	500-350-250
20	-9999	134.345	-9999.	65.188	31.349	-9999.	500-350-160
21	-9999	-9999.	205.333	103.549	50.049	-9999.	500-350-250
22	-9999	115.926	-9999.	116.184	44.848	-9999.	500-350-160

Image Visualization

- Downloadable data (cutouts / merge services) from the VLKB
- Layers stack
- Compact sources
- Filament structures
- SED Plot
- SED Fitting
- Integration with external source finders (CAESAR)



SFinder integration



- A REST-ful web service for running CAESAR source finding jobs on HPC systems
- Upload images to the SFinder service
- Source finder selection and job configuration
- Job monitoring and job output download from VLVA
- Manual source refinement/filtering

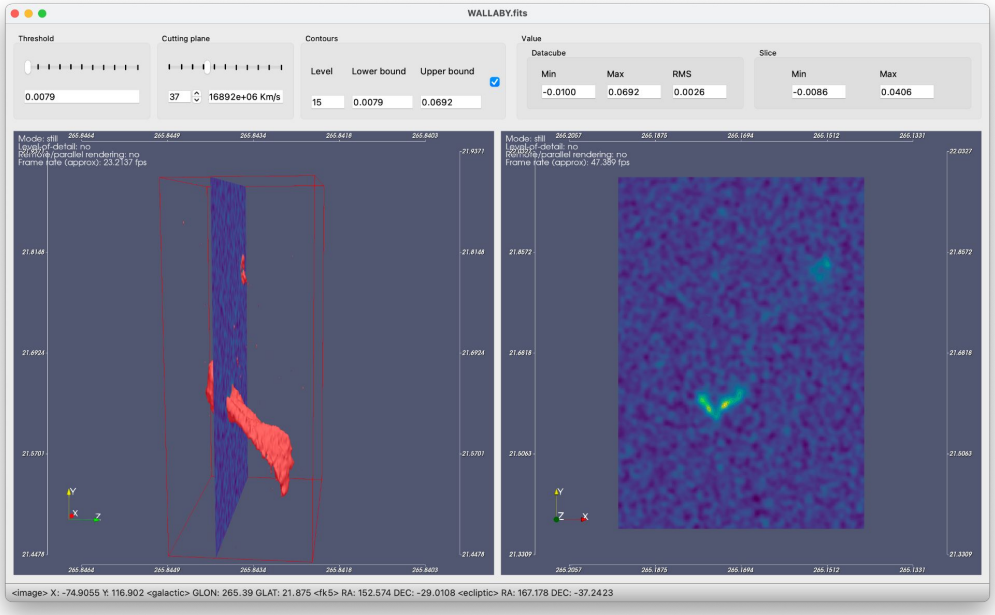
https://github.com/SKA-INAF/caesar-rest

Simone Riggi (INAF-OACT)

Datacube Visualization

Supported Data Representation:

- Volume rendering
- Slice visualization
- Isocontours
- Moment maps



Remote visualization

Remote visualization refers to any visualization where some or all of the data is on a different machine from the one used to look at images.

Motivations

- Storage: the data to be visualized exceeds by several orders of magnitude the typical storage and memory available in traditional personal computers
- Bandwidth: need to avoid as much as possible data transfers
- Computing: personal computers do not have the computing power to handle such a large amount of data

Adapting VisIVO for SRCs

Explored two approaches:

1. Containerize the tool as is and run it remotely
2. Moving from a desktop only application to a client-server based application

Containerize the application as is

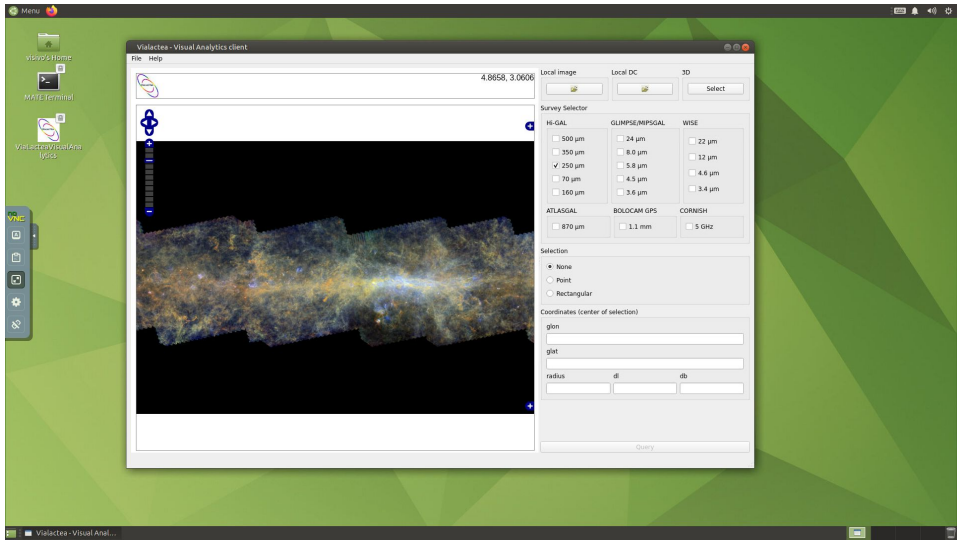
Docker image based on NVIDIA Container Toolkit

Pros:

- Accessible through VNC via web browser
- Easy to deploy
- No need to change code

Cons:

- Vertical scalability only



Client-server based application

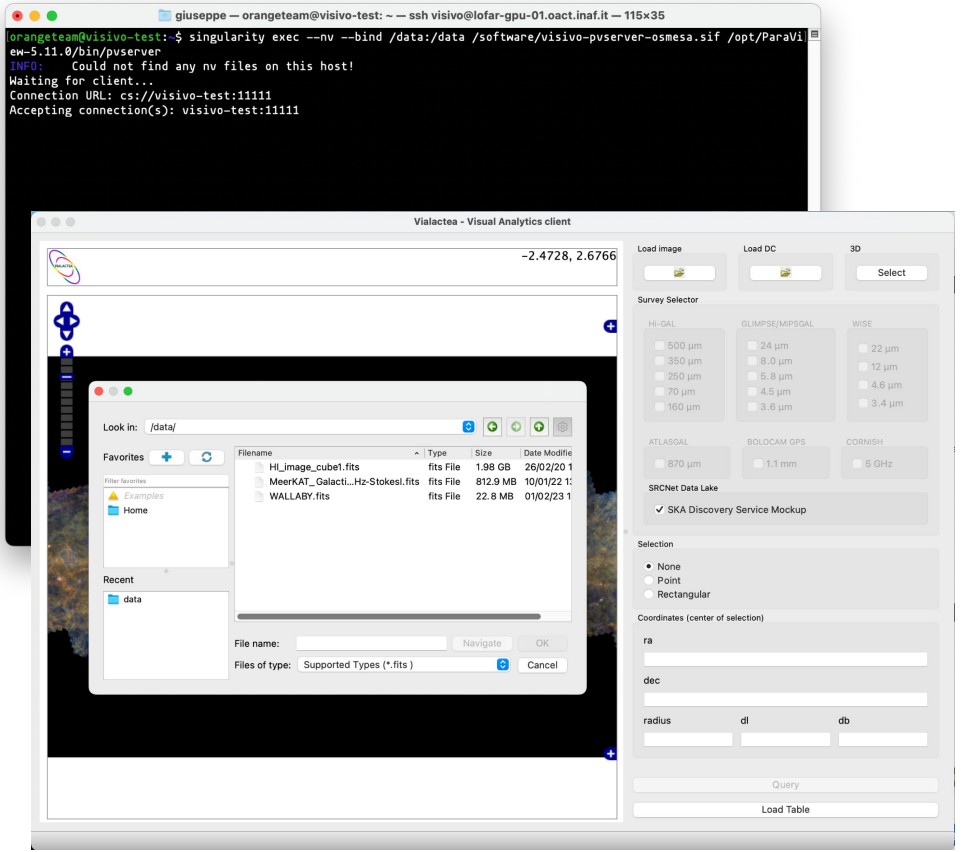
Client-server application based on Kitware ParaView

Pros:

- Only the server needs to be deployed
- Both vertical *and* horizontal scalability

Cons:

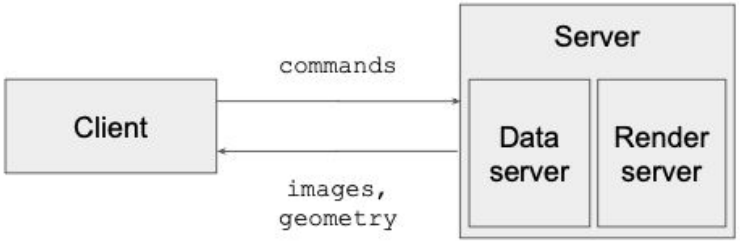
- Need to rewrite parts of the application



VisIVO based on ParaView

Three main logical components:

- a client, responsible for the user interface,
- a data server, to read and process data sets to create final geometric models,
- a render server, which renders that final geometry.



Parallel Visualization

Processing the data in parallel, simultaneously using multiple workers. Workers can be different processes running on a multicore machine or on several nodes of a cluster.

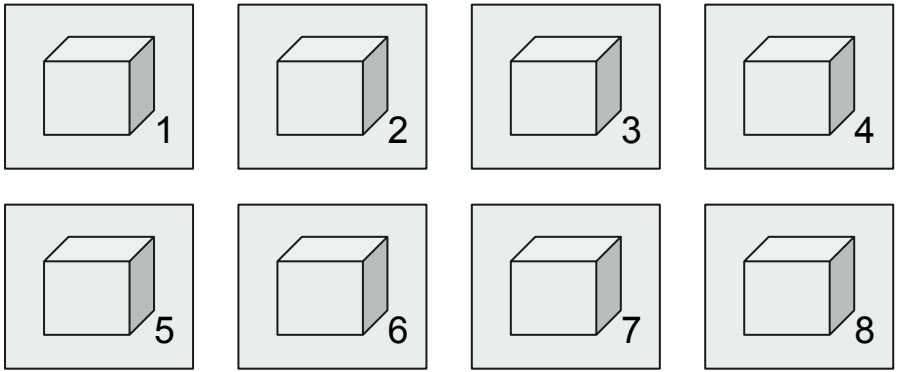
The data server and the render server in this case are a set of processes that communicate with MPI.

Each render server receives geometry from data servers in order to render a portion of the screen.

Parallel Visualization

Steps:

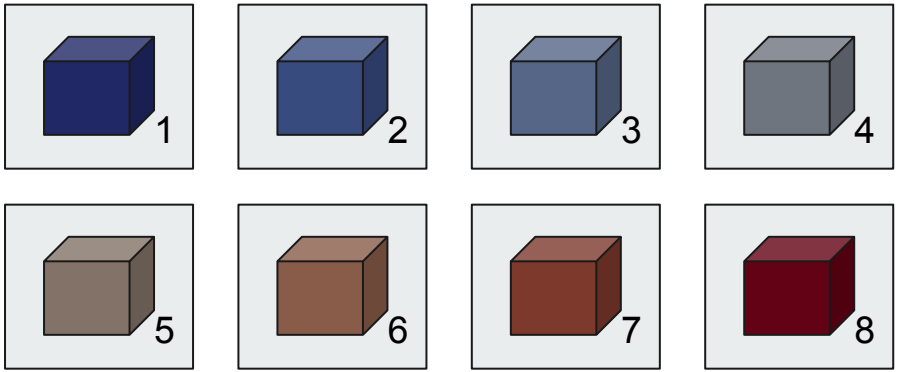
1. Partitioning input



Parallel Visualization

Steps:

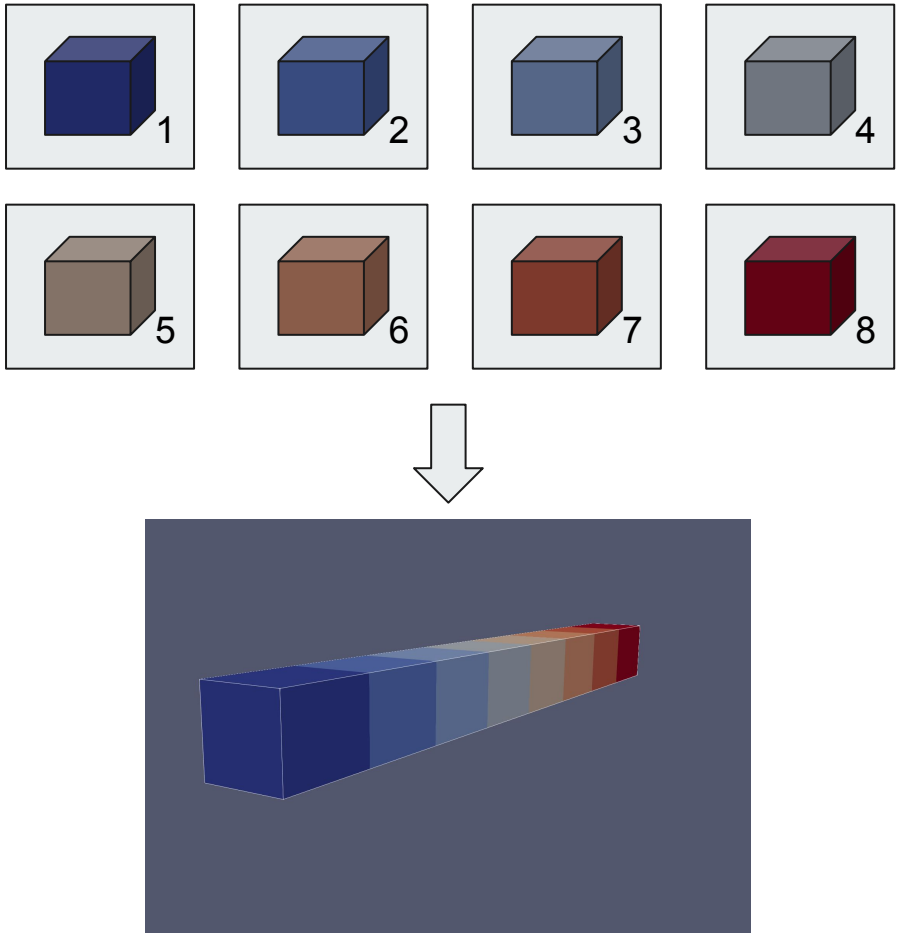
1. Partitioning input
2. Each worker works on its own chunk of data and produces partial results



Parallel Visualization

Steps:

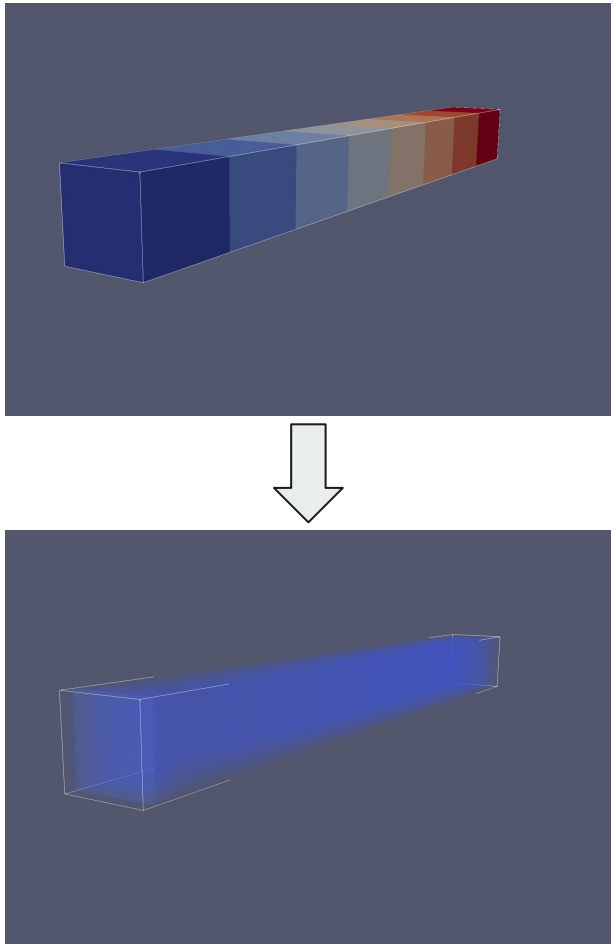
1. Partitioning input
2. Each worker works in its own chunk of data and produces partial results
3. Construct the final output



Parallel Visualization

Steps:

1. Partitioning input
2. Each worker works in its own chunk of data and produces partial results
3. Construct the final output
4. Send the result to the client



Conclusions

- The client-server approach, despite requiring more rewriting work, is the most efficient and suitable for use in the SRCNet architecture.
- This VisIVO flavour is currently being developed on two heterogeneous SRC nodes (ChinaSRC & SpainSRC).
- In planning further development and testing of VisIVO on a prototype miniSRC that will be developed with INAF resources distributed across Catania, Bologna, and Trieste.