

Analysis and Visualization with yt

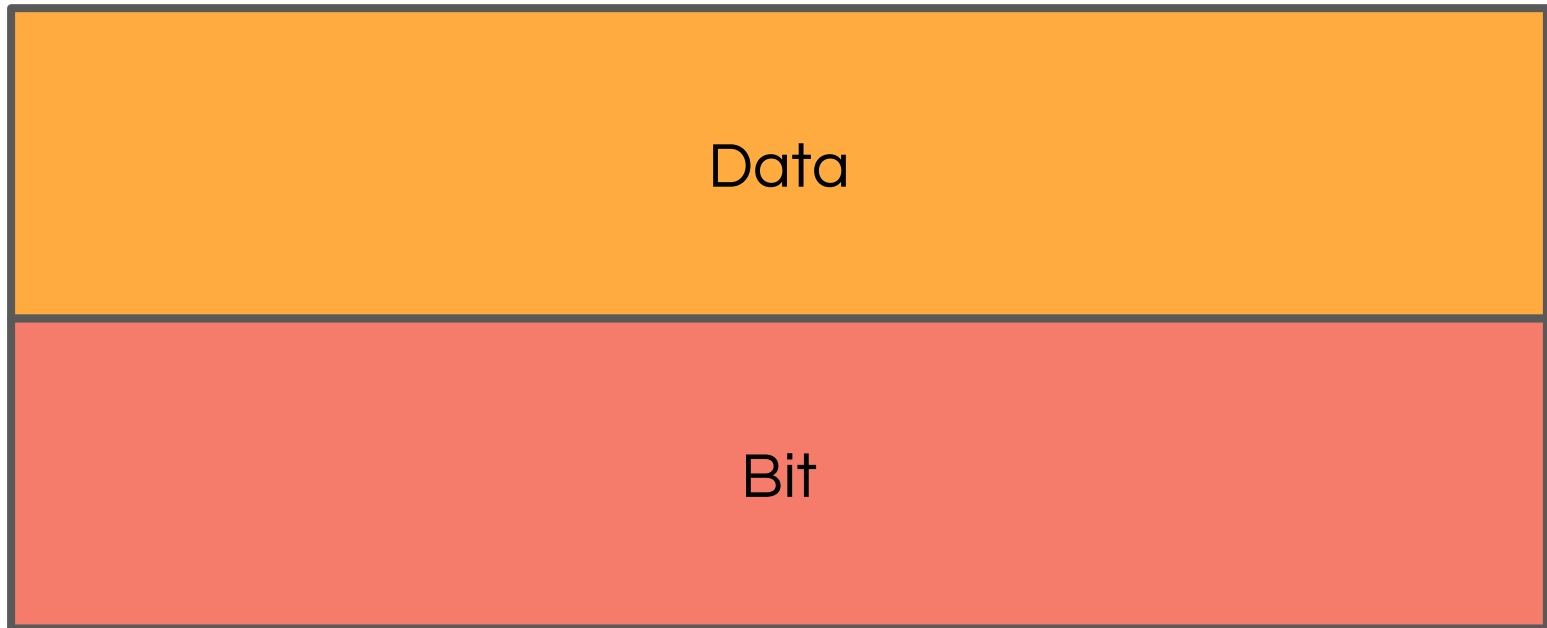
Matthew Turk
UIUC School of Information Sciences
UIUC Astronomy

yt-project.org
hub.yt

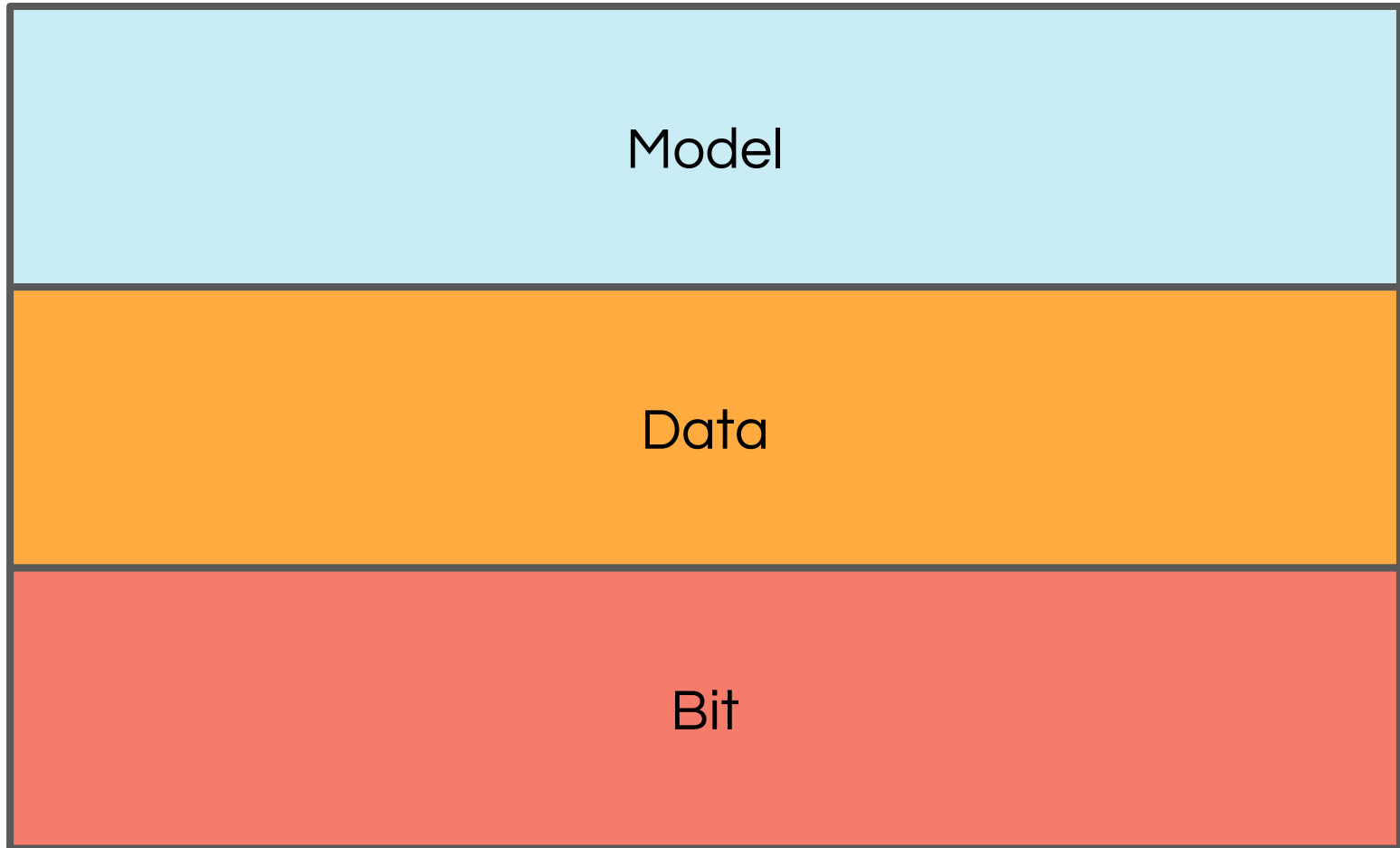
Layers of Representation

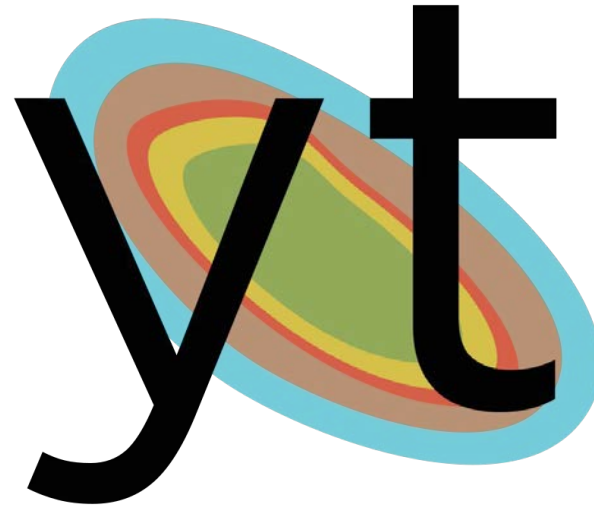
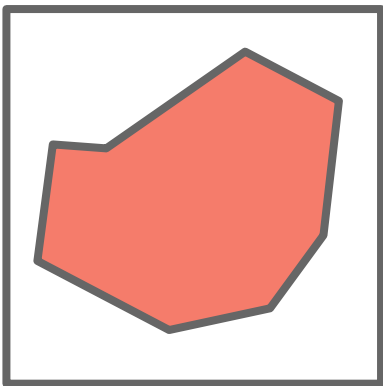
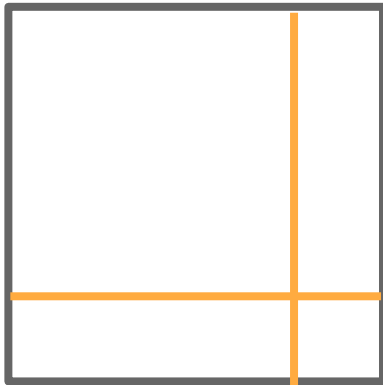
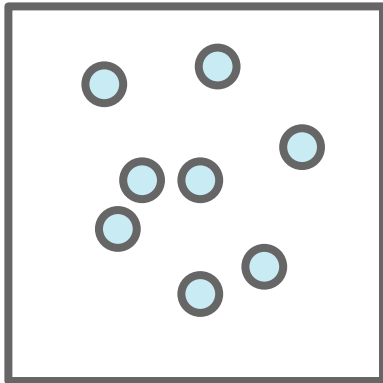


Layers of Representation



Layers of Representation





yt-project.org

Volumetric analysis and visualization

NUMFOCUS

OPEN CODE = BETTER SCIENCE

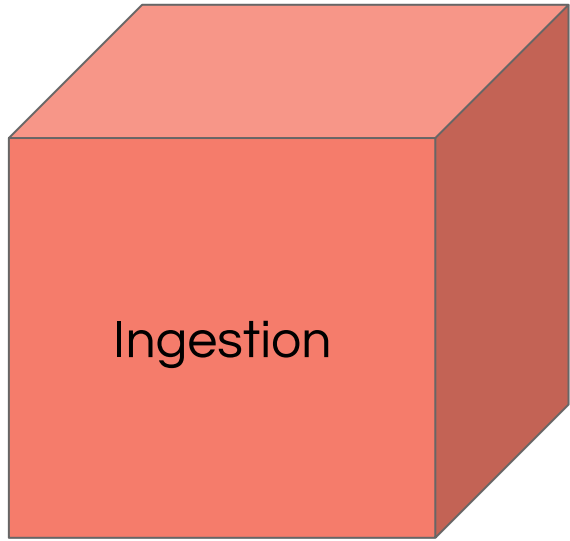


ytini

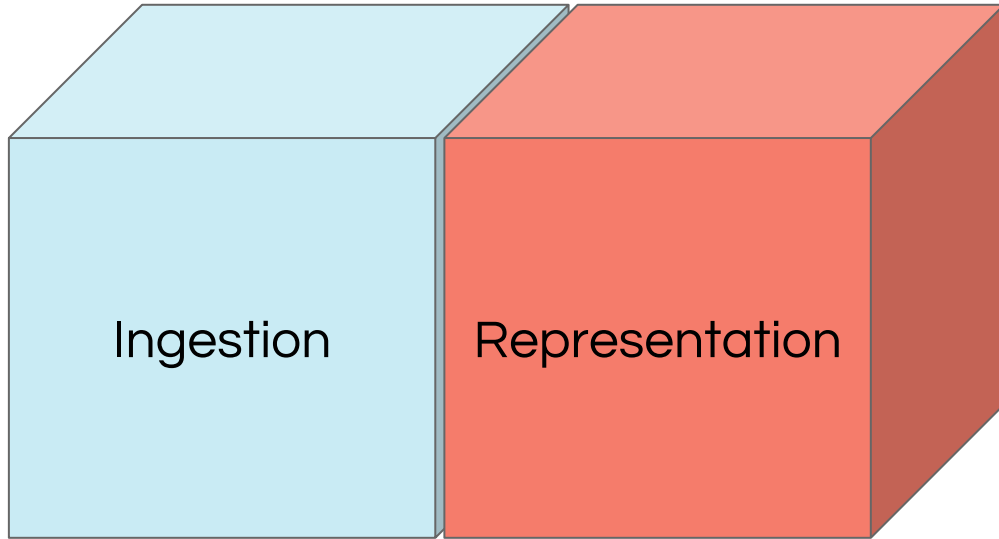
Jill Naiman, AJ Christensen, Kalina Borkiewicz
ytini.com

the yt project

- Python-based (C, Cython, etc)
- Community developed
 - NumFOCUS FSA
 - Code of Conduct
 - Governance structure
 - 100+ contributors
 - Volumetric and non-spatial data
- Used in nearly 300 papers
- Grids, particles, octrees, and unstructured meshes
- Arbitrary geometric representations
- Minimize time to inquiry

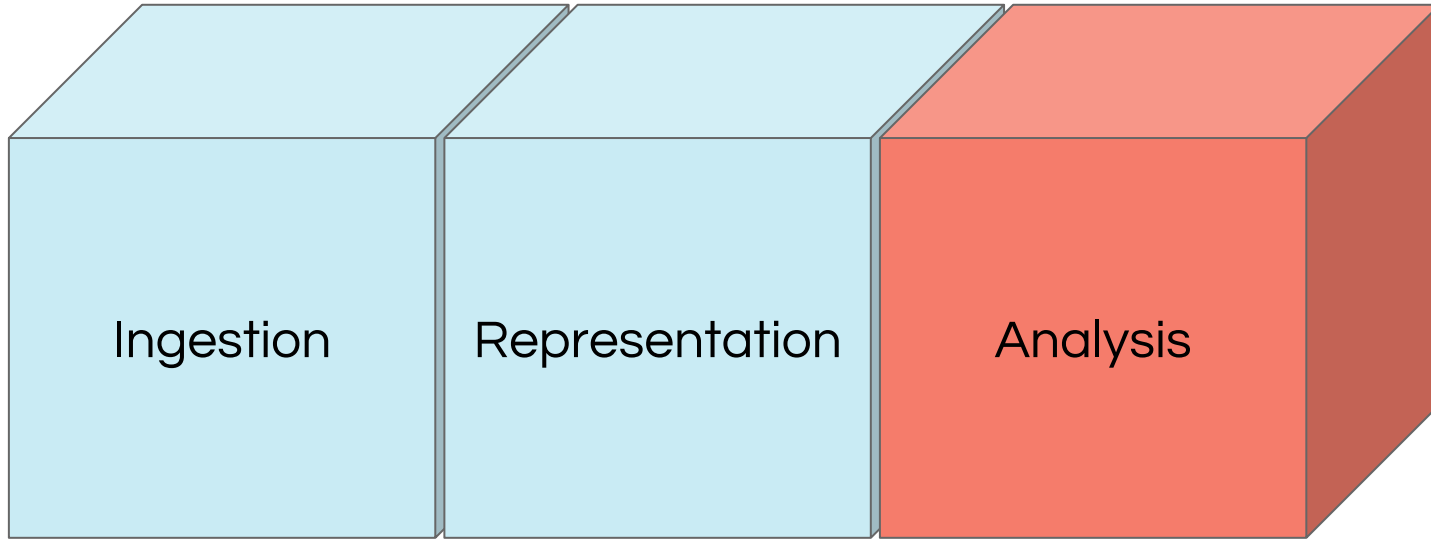


Ingestion



Ingestion

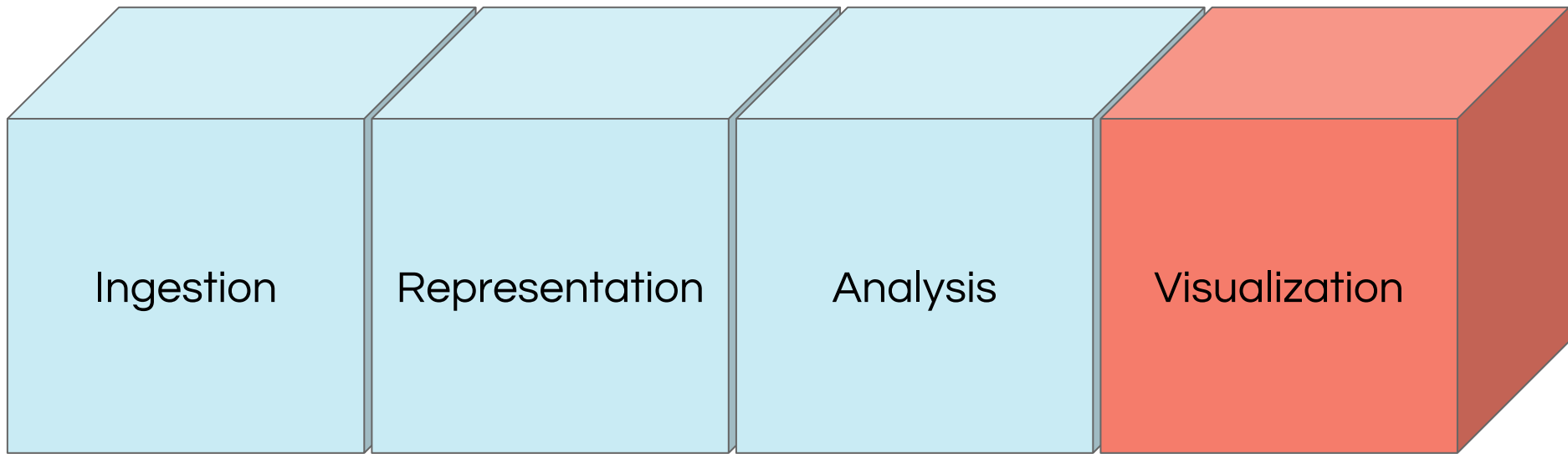
Representation



Ingestion

Representation

Analysis





Community

Ingestion

Representation

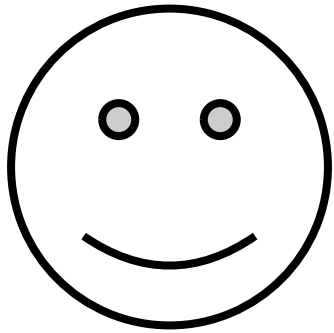
Analysis

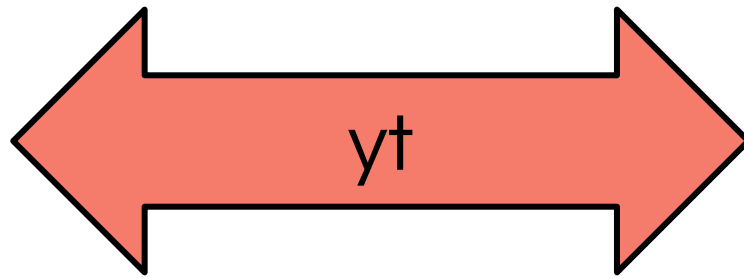
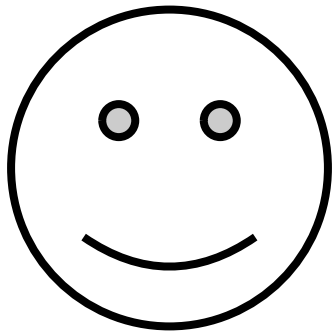
Visualization

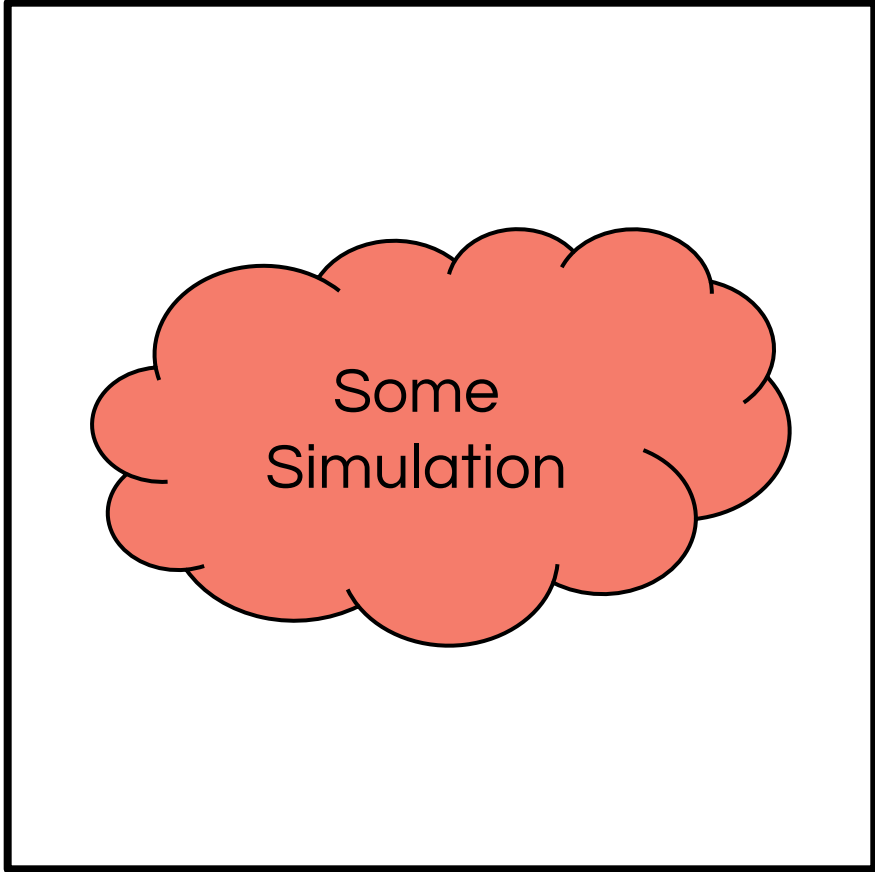
Project Members

| | |
|-----------------|-----------------|
| Kenza Arraki | Andrew Myers |
| Corentin Cadiou | Jill Naiman |
| Brian Crosby | Jeff Oishi |
| Bili Dong | Brian O'Shea |
| Hilary Egan | Douglas Rudd |
| Nathan Goldbaum | Anthony Scopatz |
| Cameron Hummels | Sam Skillman |
| Suoqing Ji | Stephen Skory |
| Allyson Julian | Britton Smith |
| Ben Keller | Casey Stark |
| Kacper Kowalik | Matthew Turk |
| Sam Leitner | John Wise |
| Alex Lindsay | Michael Zingale |
| Chris Malone | John ZuHone |

About 100 contributors...

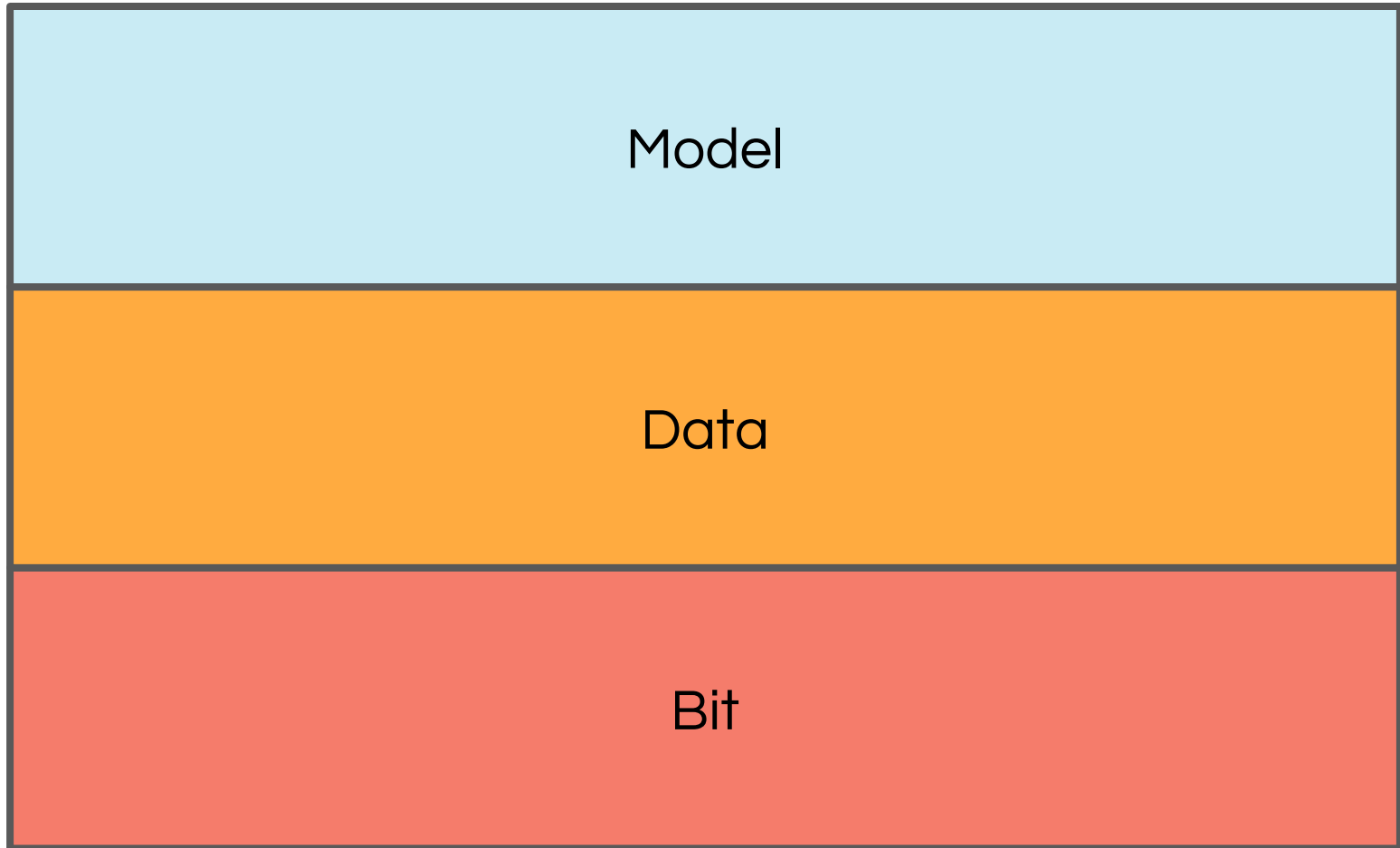






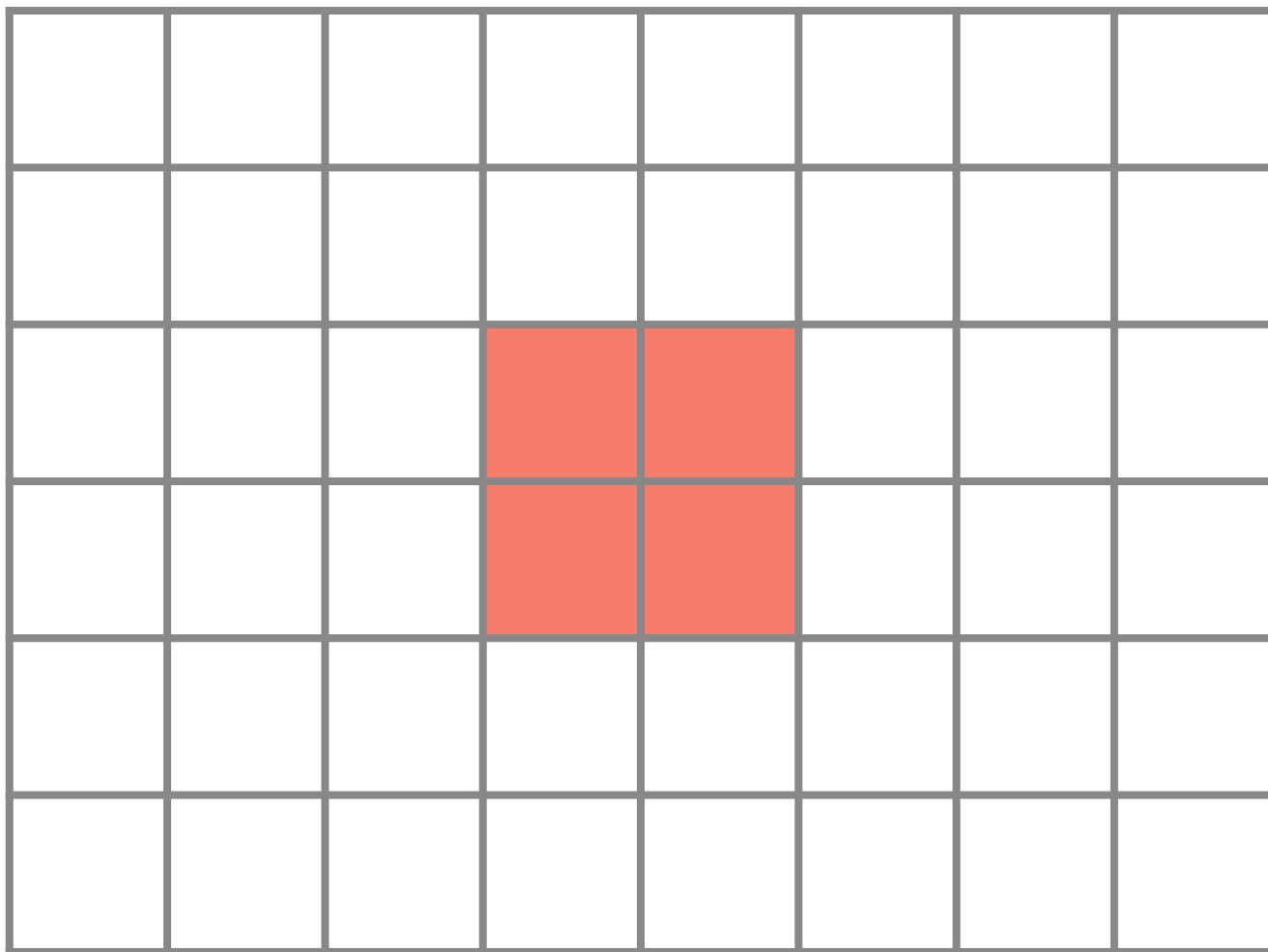
Some
Simulation

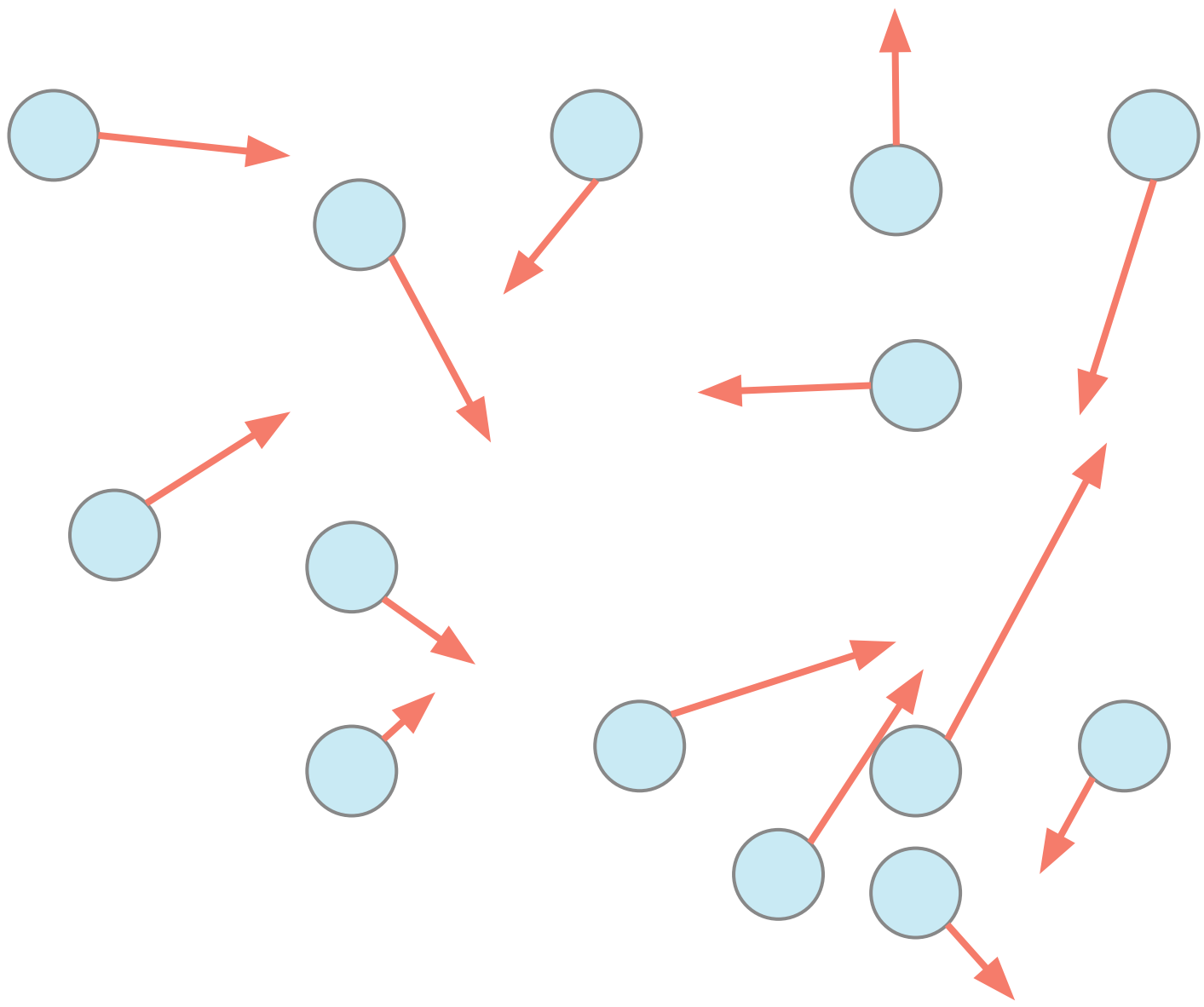
Layers of Representation

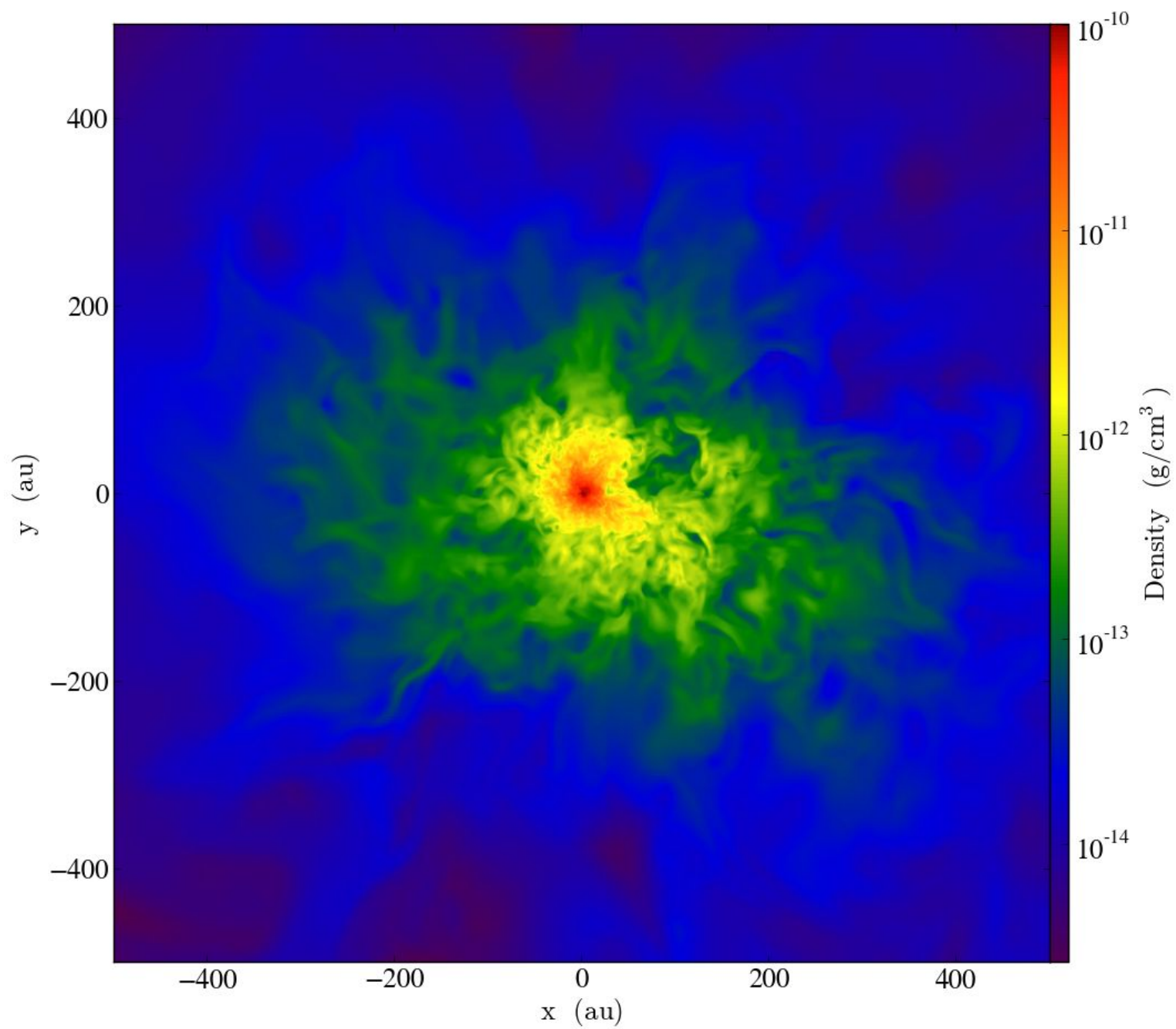


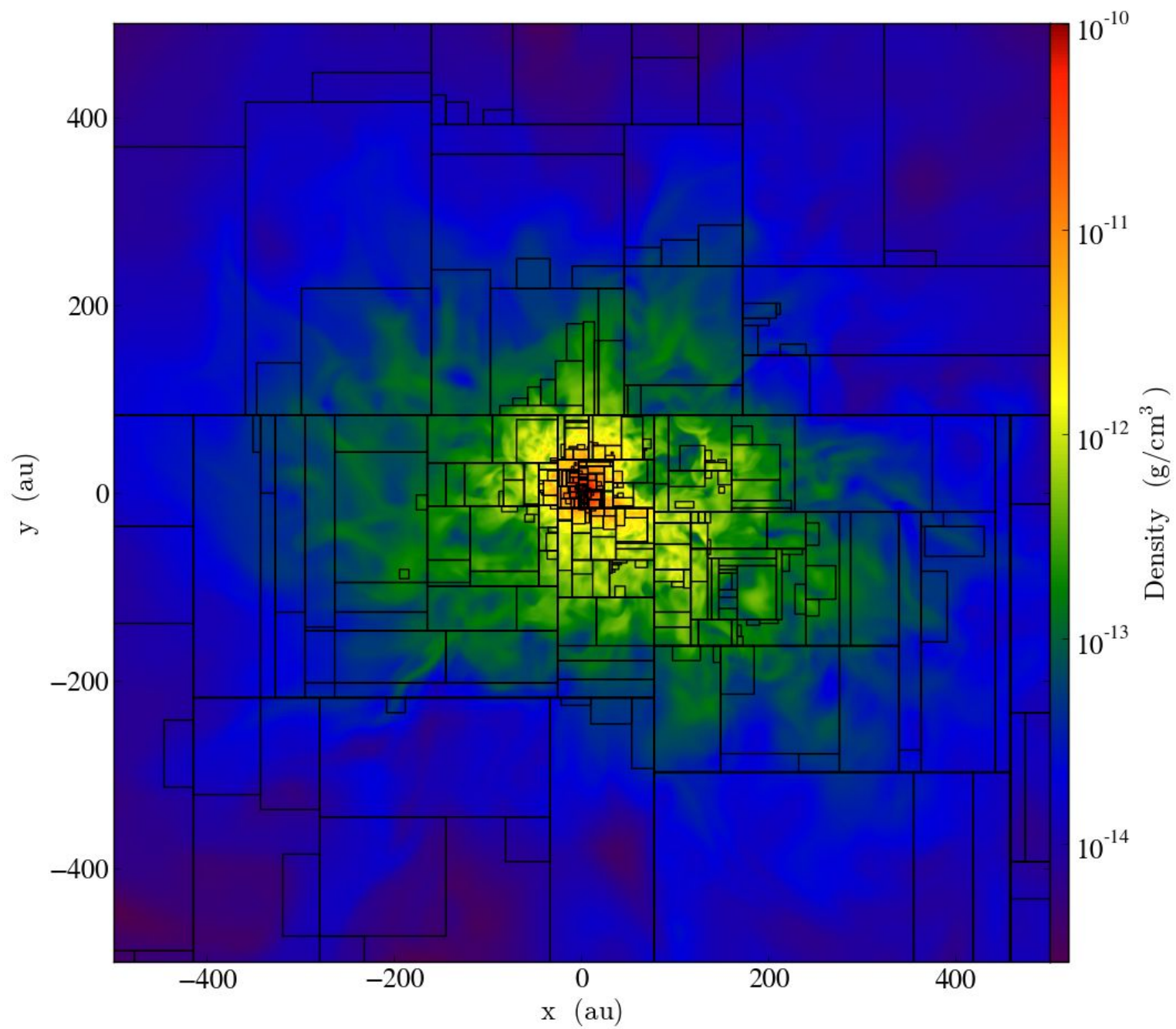
| | | | | | | | |
|--|--|--|--|--|--|--|--|
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

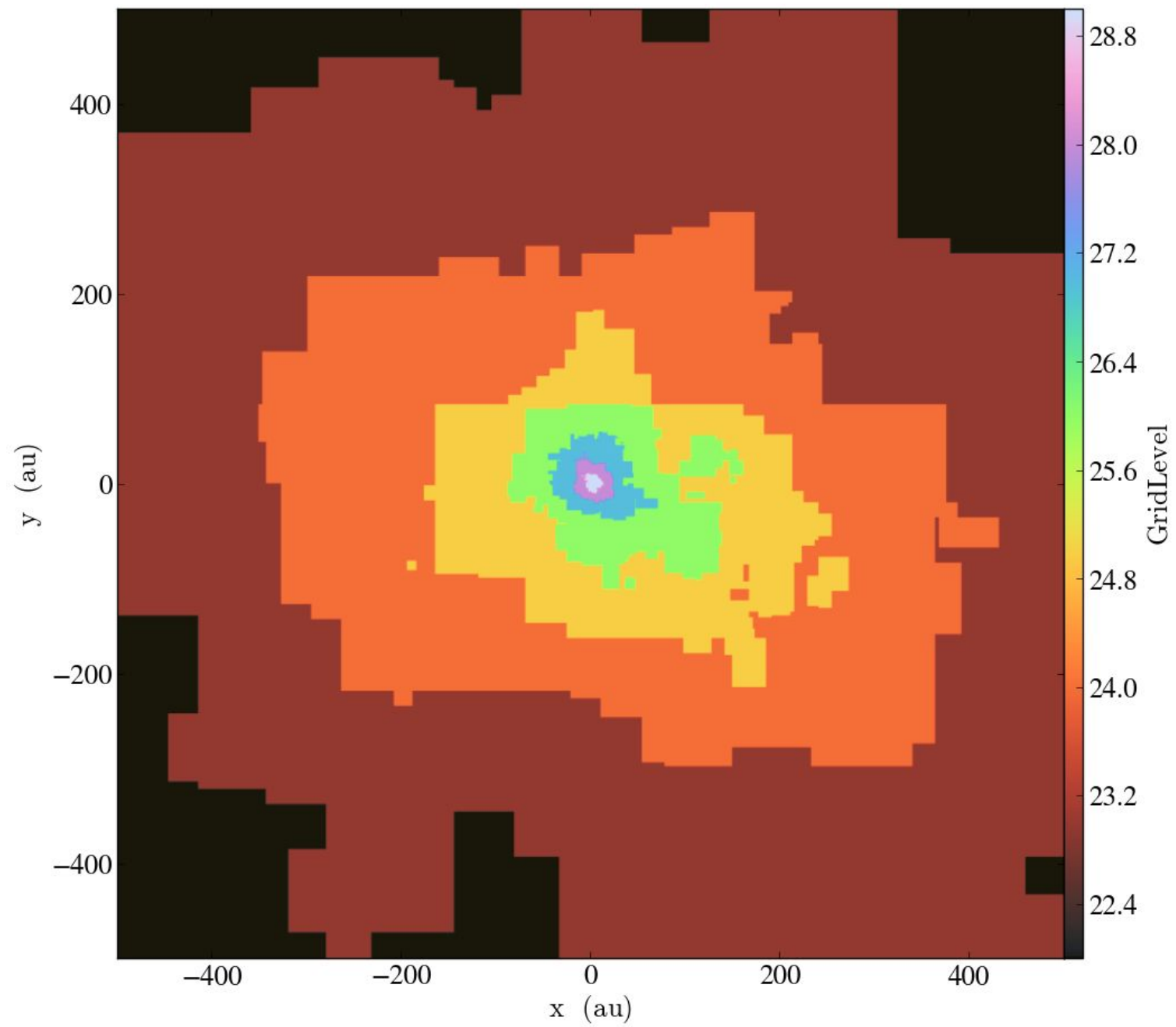












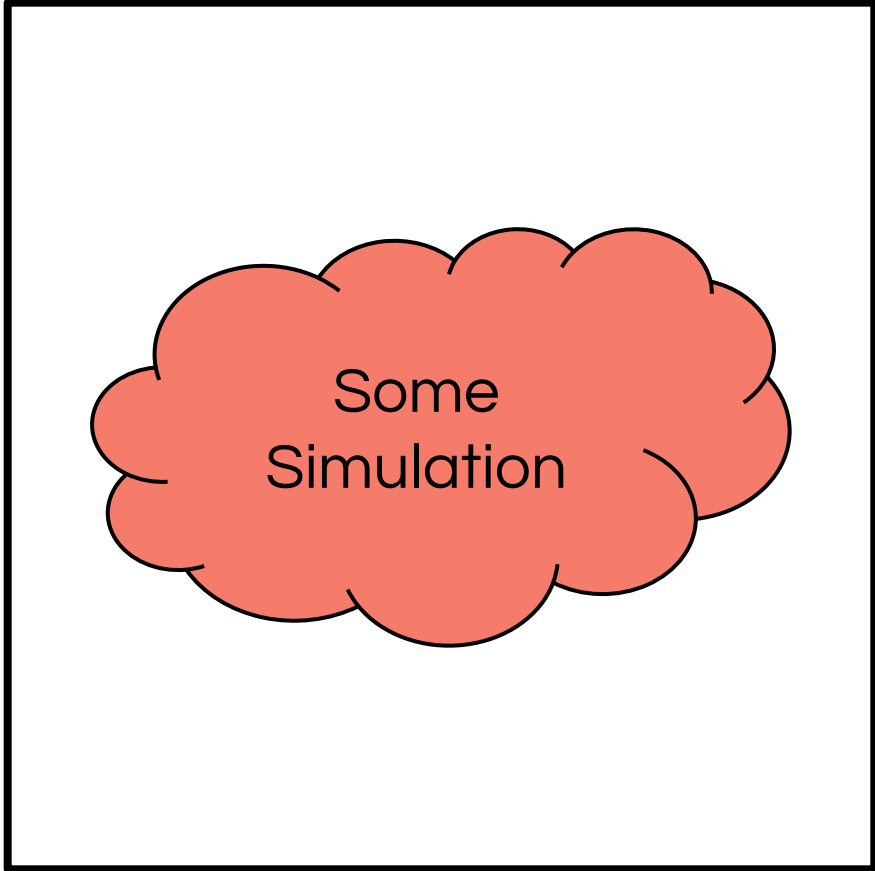
| | |
|------------|---------------------|
| ART | Maestro |
| ARTIO | MOAB |
| Athena | Nyx |
| Carpet | OWLS |
| Castro | OWLS-Subfind |
| Chombo | PKDGrav |
| Eagle | PLUTO |
| Enzo | RAMSES |
| ExodusII | Rockstar |
| FITS | SDF |
| FLASH | Stream HTTP |
| Gadget | Stream Grids |
| Gadget-FOF | Stream Octree |
| Gasoline | Stream Particles |
| GDF | Stream Unstructured |

| | |
|------------|----------------------|
| ART | Maestro |
| ARTIO | MOAB |
| Athena | Nyx |
| Carpet | OWLS |
| Castro | OWLS-Subfind |
| Chombo | PKDGrav |
| Eagle | PLUTO |
| Enzo | RAMSES |
| ExodusII | Rockstar |
| FITS | SDF |
| FLASH | Stream HTTP |
| Gadget | Stream Grids |
| Gadget-FOF | Stream Octree |
| Gasoline | Stream Particles |
| GDF | Stream Unstructured |

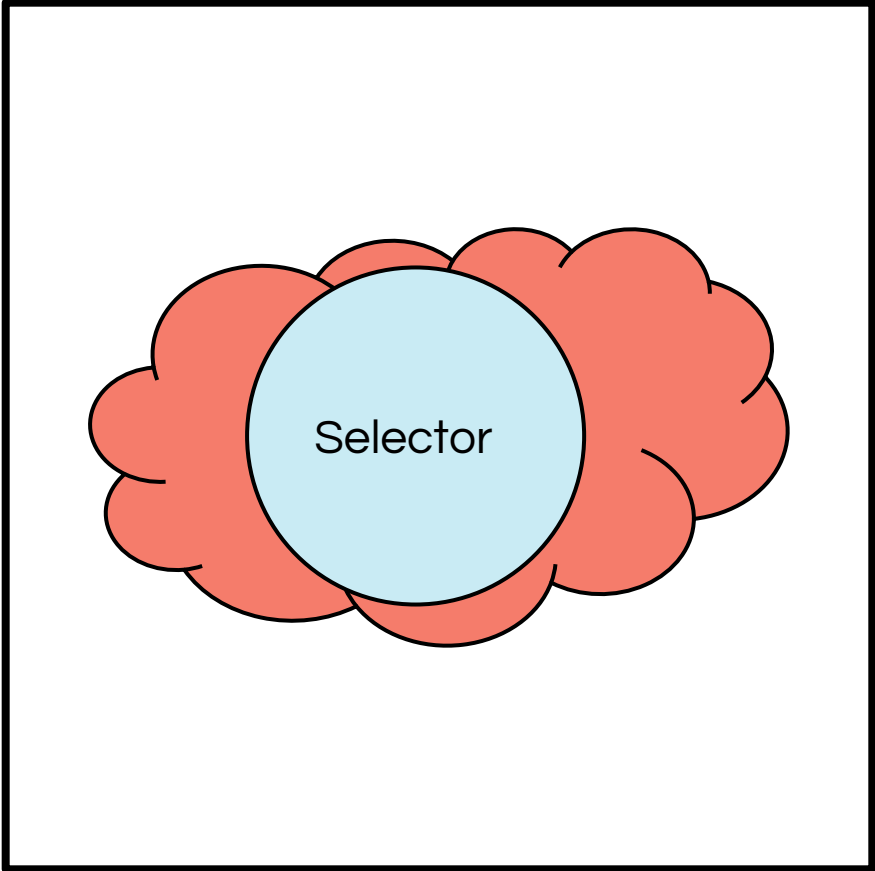
| | |
|------------|---------------------|
| ART | Maestro |
| ARTIO | MOAB |
| Athena | Nyx |
| Carpet | OWLS |
| Castro | OWLS-Subfind |
| Chombo | PKDGrav |
| Eagle | PLUTO |
| Enzo | RAMSES |
| ExodusII | Rockstar |
| FITS | SDF |
| FLASH | Stream HTTP |
| Gadget | Stream Grids |
| Gadget-FOF | Stream Octree |
| Gasoline | Stream Particles |
| GDF | Stream Unstructured |

| | |
|--------------|---------------------|
| ART | Maestro |
| ARTIO | MOAB |
| Athena | Nyx |
| Carpet | OWLS |
| Castro | OWLS-Subfind |
| Chombo | PKDGrav |
| Eagle | PLUTO |
| Enzo | RAMSES |
| ExodusII | Rockstar |
| FITS | SDF |
| FLASH | Stream HTTP |
| Gadget | Stream Grids |
| Gadget-FOF | Stream Octree |
| Gasoline | Stream Particles |
| GDF | Stream Unstructured |

| | |
|-----------------|----------------------------|
| ART | Maestro |
| ARTIO | MOAB |
| Athena | Nyx |
| Carpet | OWLS |
| Castro | OWLS-Subfind |
| Chombo | PKDGrav |
| Eagle | PLUTO |
| Enzo | RAMSES |
| ExodusII | Rockstar |
| FITS | SDF |
| FLASH | Stream HTTP |
| Gadget | Stream Grids |
| Gadget-FOF | Stream Octree |
| Gasoline | Stream Particles |
| GDF | Stream Unstructured |

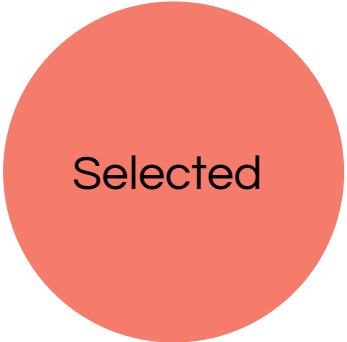


Some
Simulation

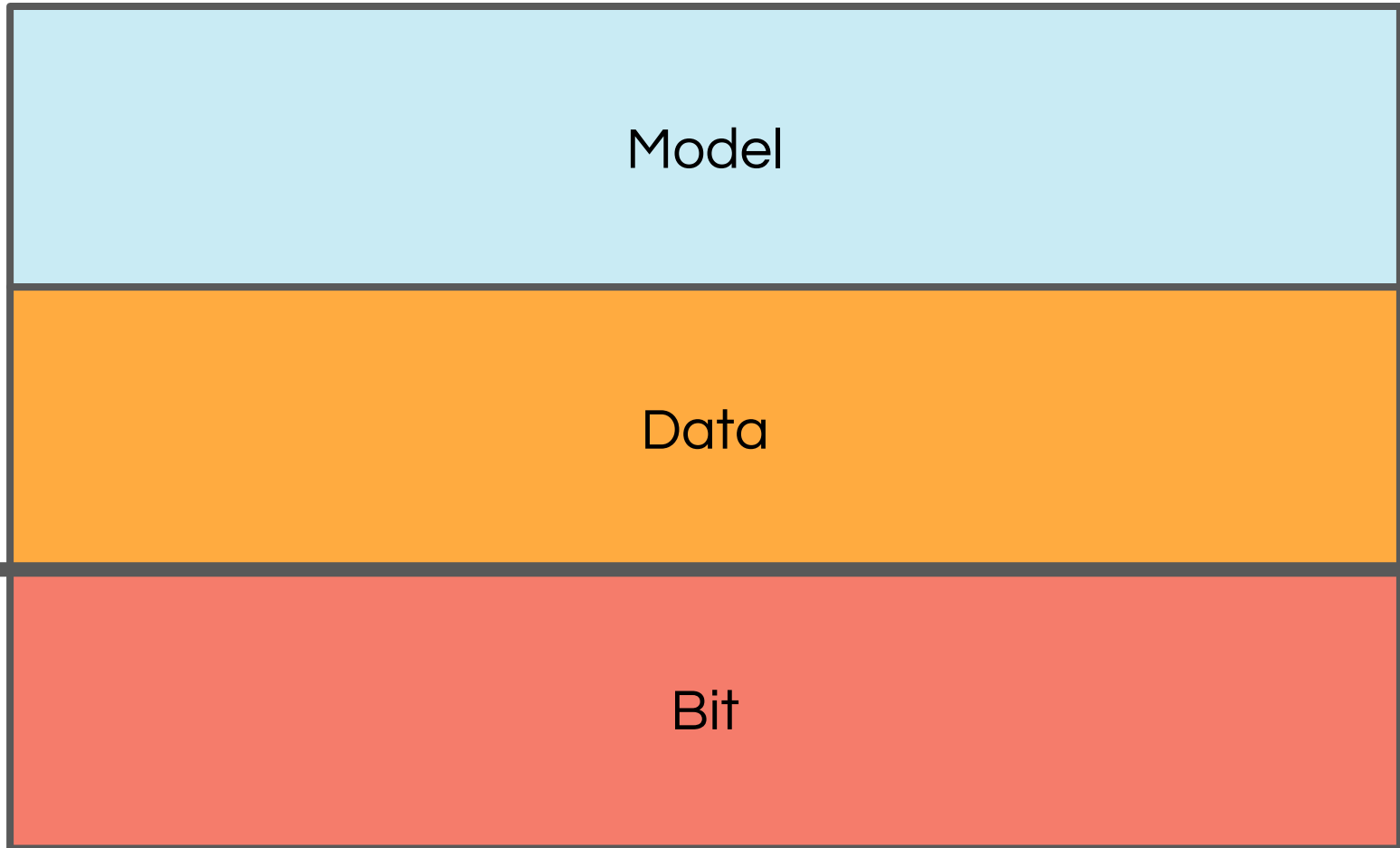




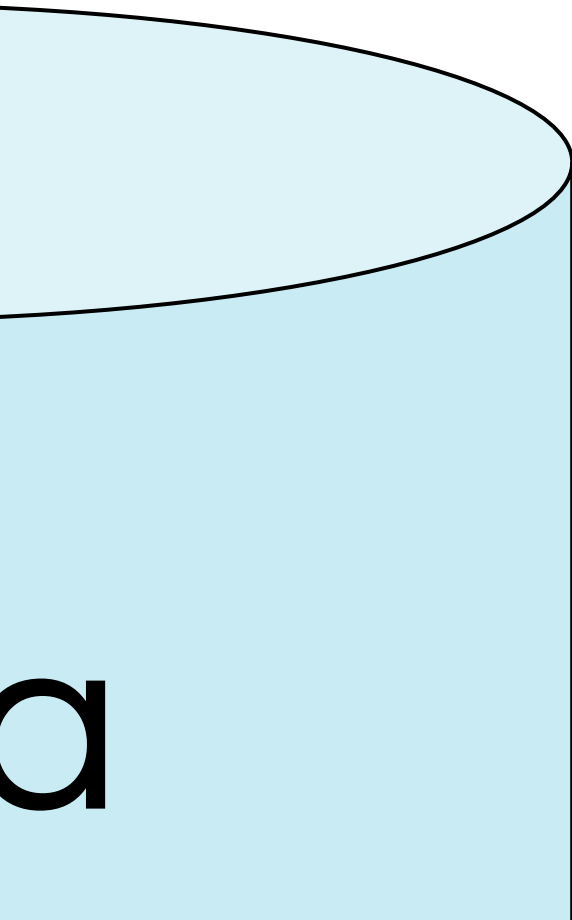
Selected



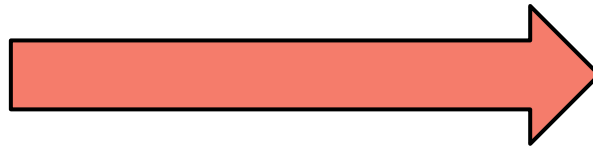
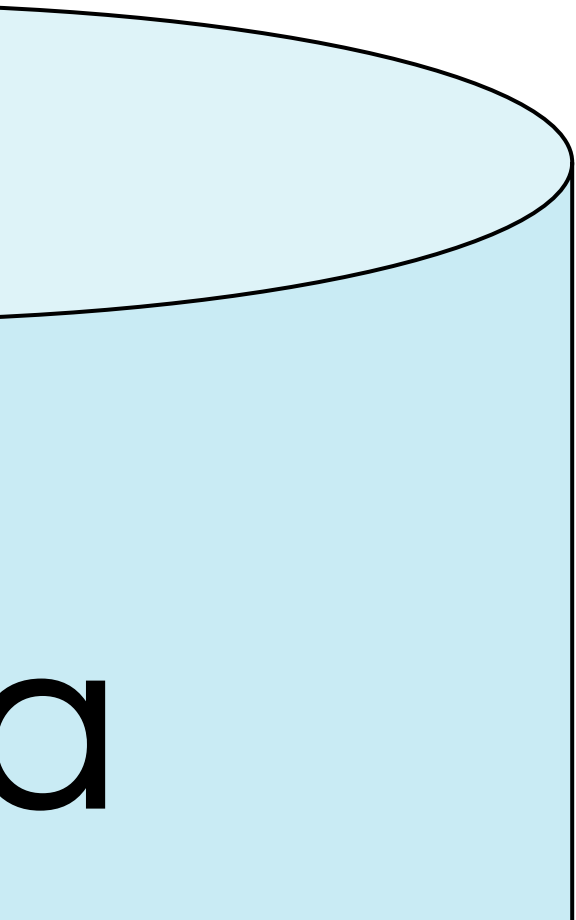
Layers of Representation



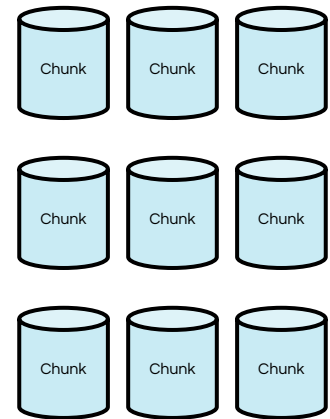
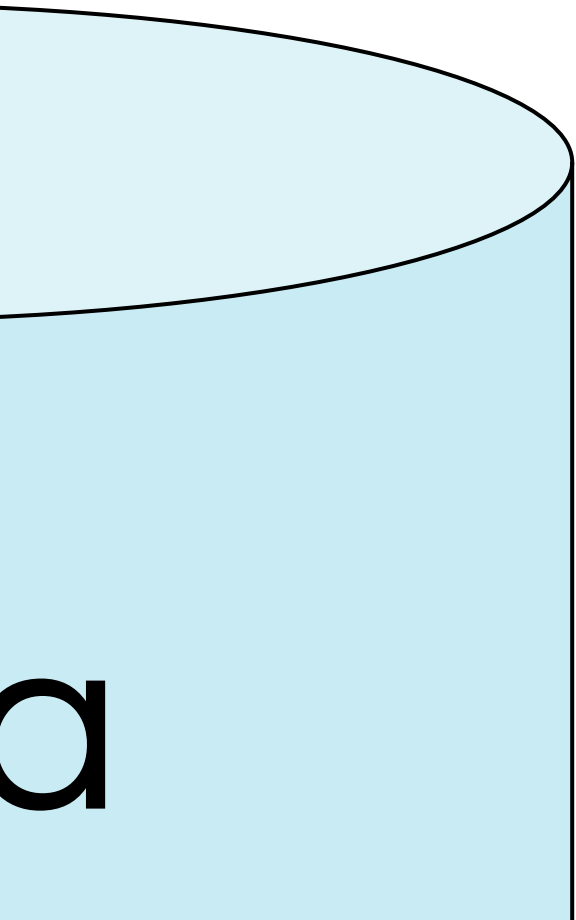
Low-Level Operations



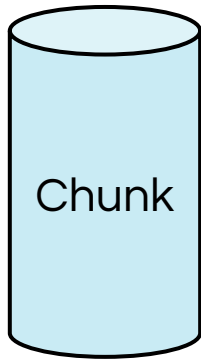
Low-Level Operations



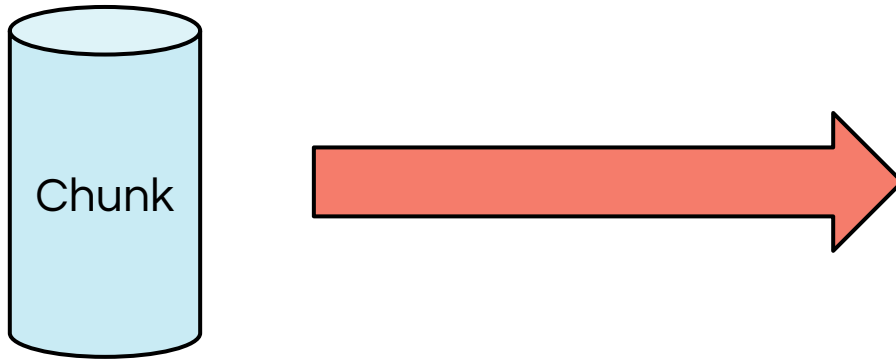
Low-Level Operations



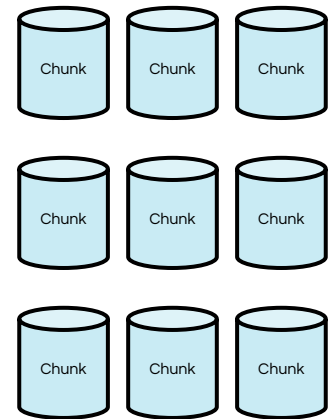
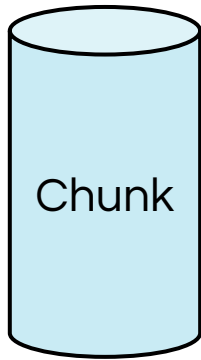
Low-Level Operations



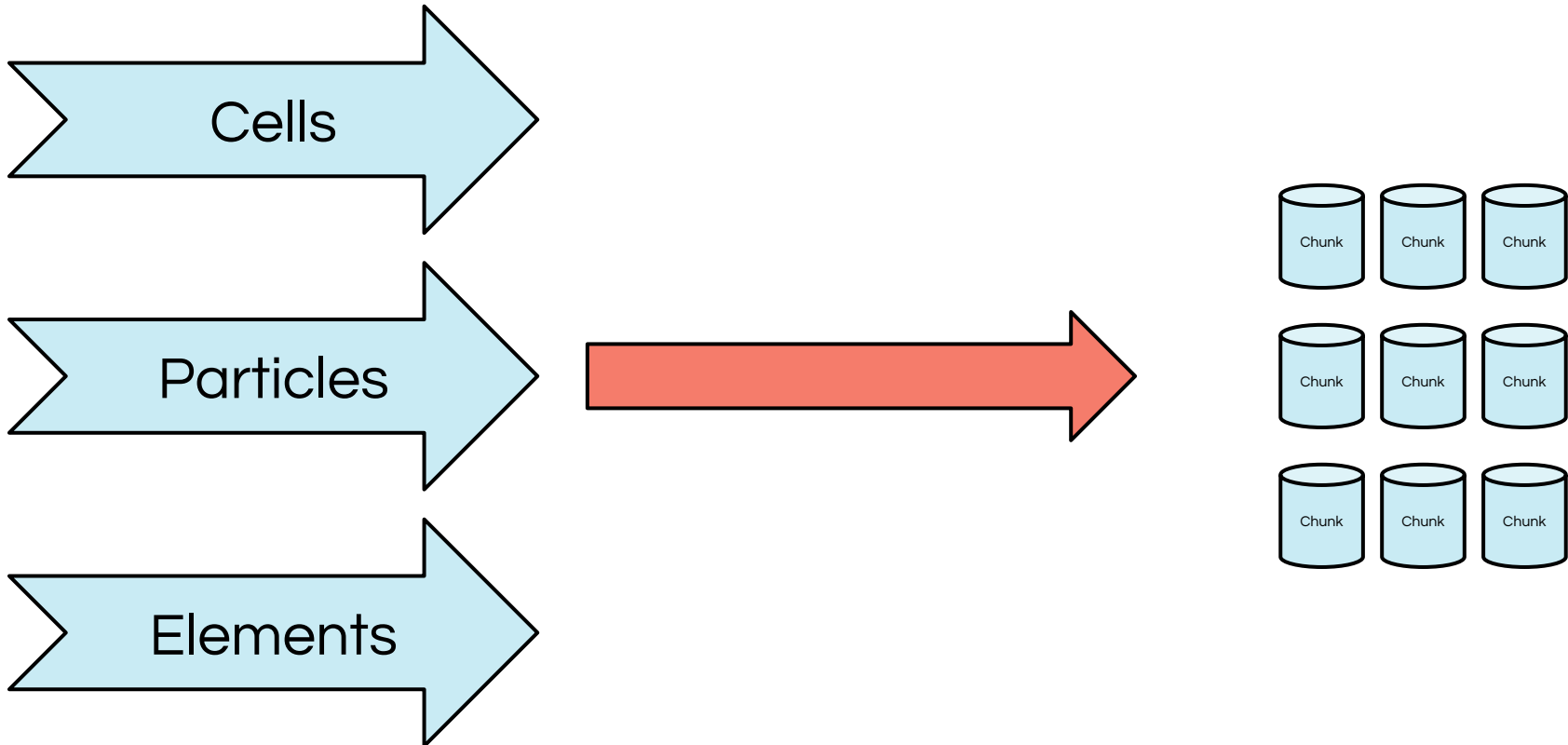
Low-Level Operations



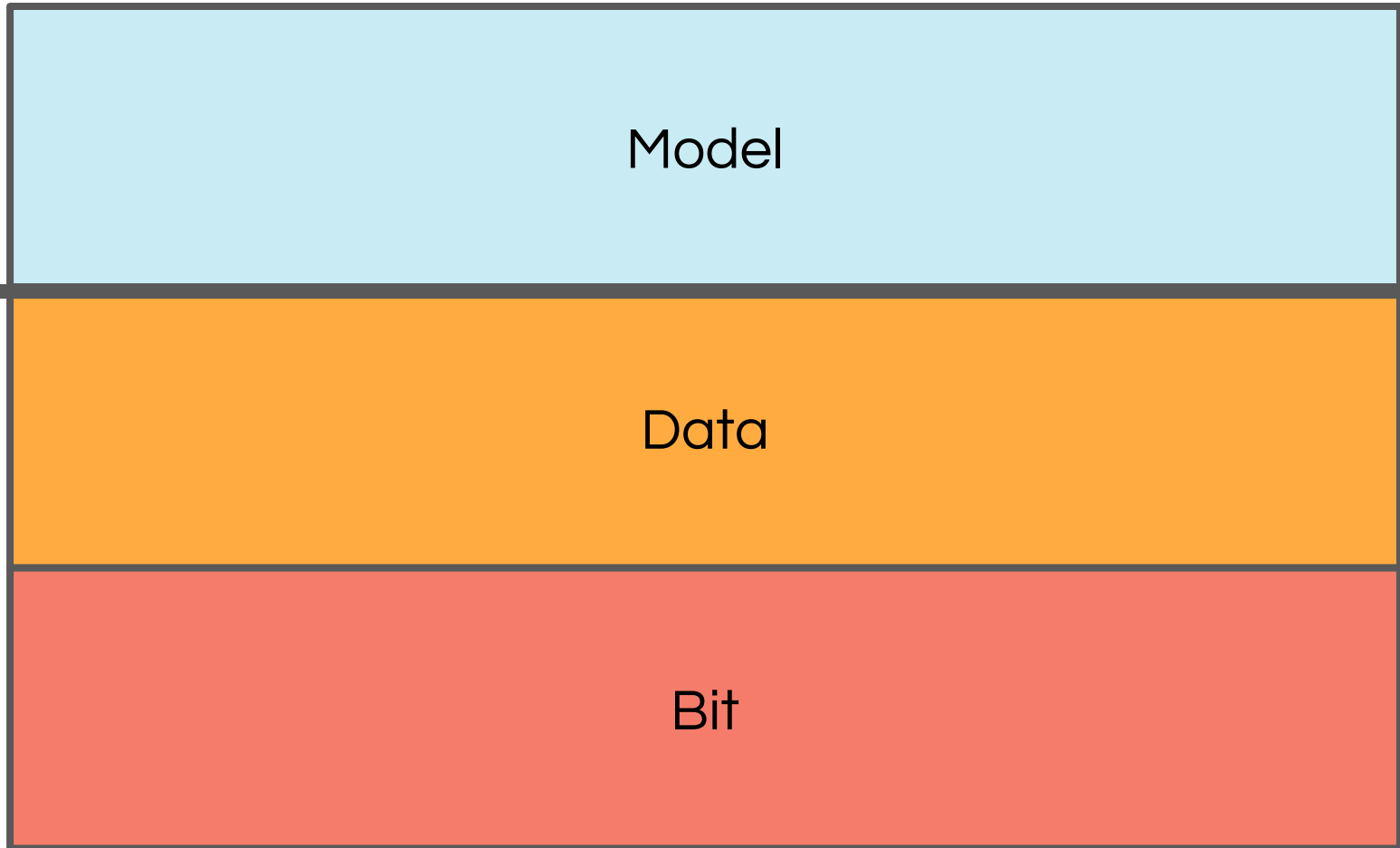
Low-Level Operations



Low-Level Operations



Layers of Representation



Data Representation

- Coordinate Handling
 - Cartesian
 - Cylindrical
 - Spherical (geographic, tomographic)
- Symbolic Units
- Derived fields
 - Dependency calculation
 - Arithmetic and spatial

“Fields”

Representation of state.

“Fields”

Representation of state.

- Name
- Units
- Context
- Prescription

"Fields"

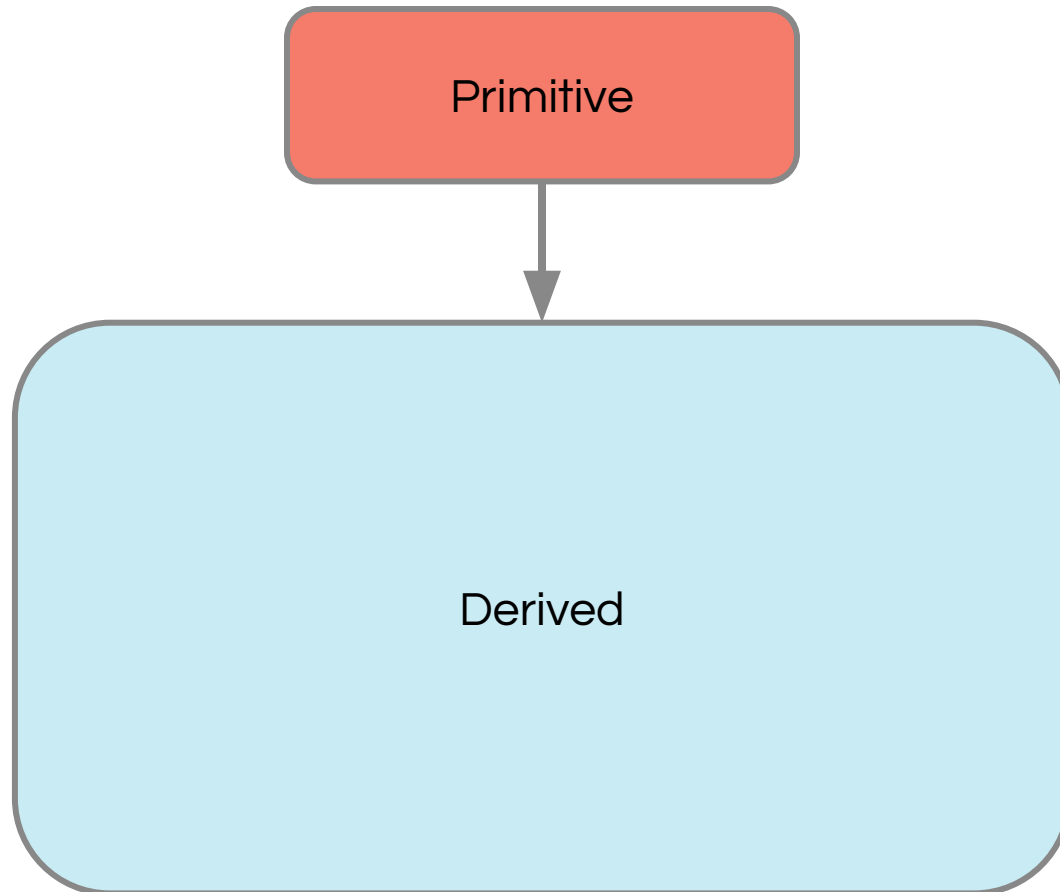
Representation of state.



Primitive

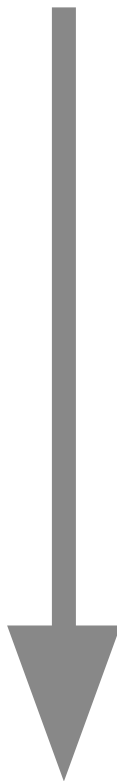
“Fields”

Representation of state.



"Fields"

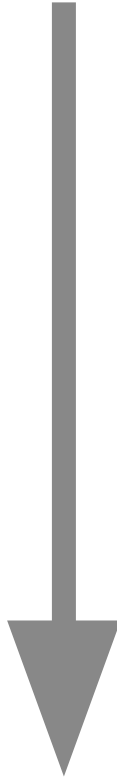
Arithmetic Operations



$$E = \frac{mv^2}{2}$$

"Fields"

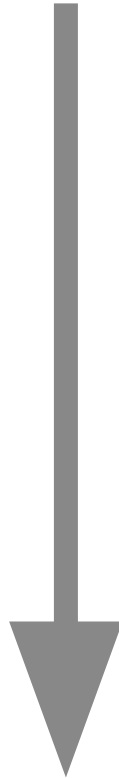
Arithmetic Operations



```
@derived_field("energy", units="erg")
def energy(field, data):
    E = 0.5 * (data["mass"] *
              data["velocity_magnitude"]**2)
    return E
```

"Fields"

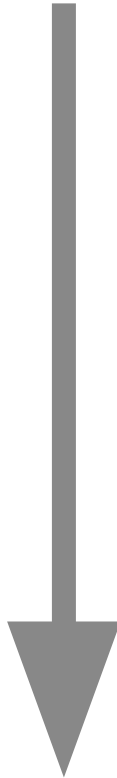
Spatial Operations



$$\operatorname{div} V = \frac{\delta V_x}{\delta x} + \frac{\delta V_y}{\delta y} + \frac{\delta V_z}{\delta z}$$

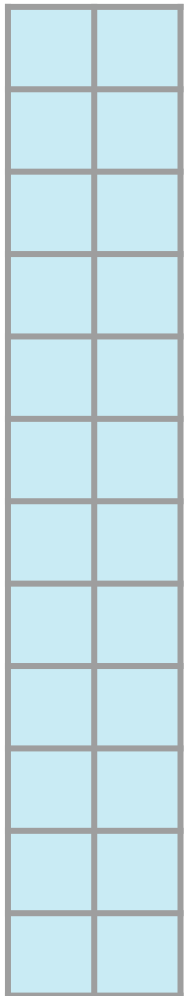
"Fields"

Spatial Operations

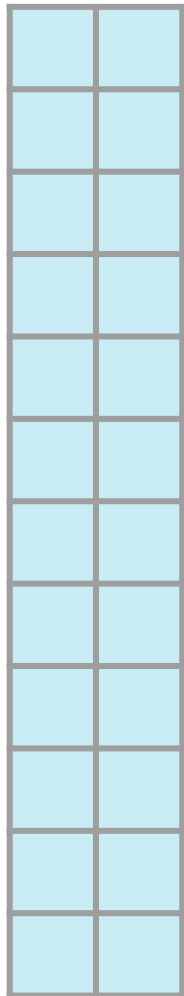


```
@derived_field("velocity_divergence", units="1/s",
               validators=[ValidateSpatial(1)])
def divergence(field, data):
    dx = data["index", "dx"]
    dy = data["index", "dy"]
    dz = data["index", "dz"]
    f = data["velocity_x"][s1_r, 1:-1, 1:-1]/ds
    f -= data["velocity_x"][s1_l, 1:-1, 1:-1]/ds
    f += data["velocity_y"][1:-1, s1_r, 1:-1]/ds
    f -= data["velocity_y"][1:-1, s1_l, 1:-1]/ds
    f += data["velocity_z"][1:-1, 1:-1, s1_r]/ds
    f -= data["velocity_z"][1:-1, 1:-1, s1_l ]/ds
    return f
```

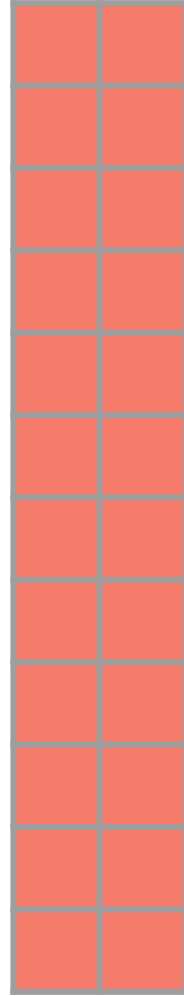
Mid-Level Operations



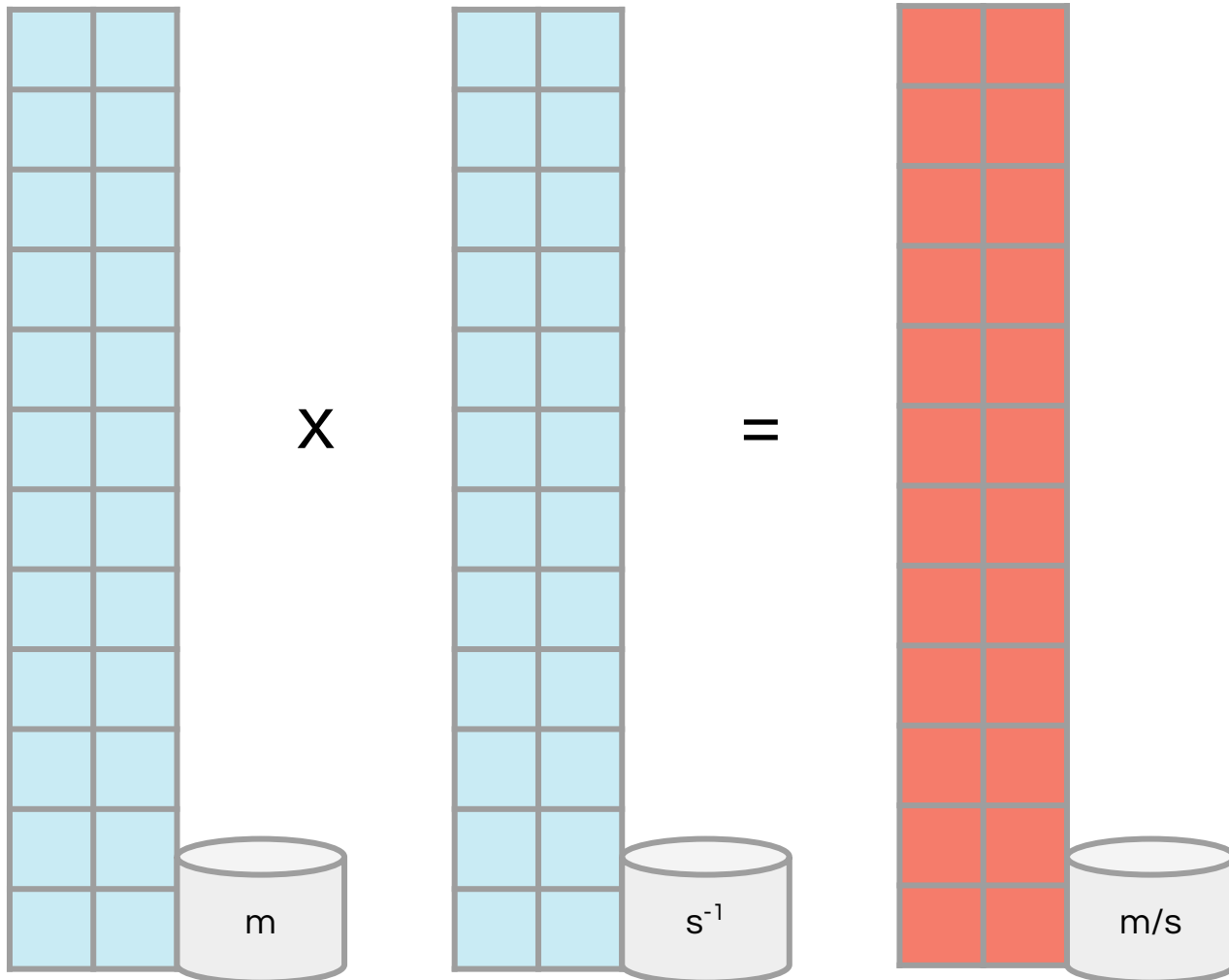
x



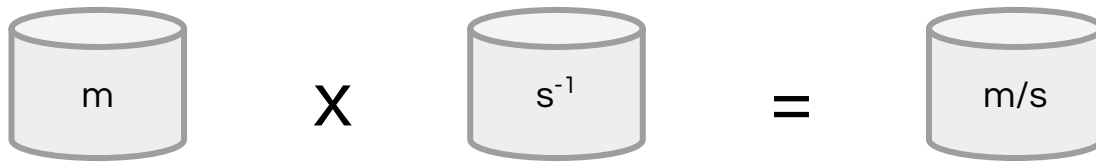
=



Mid-Level Operations

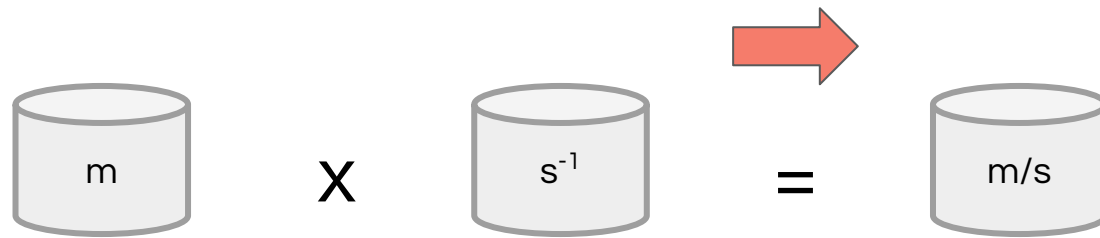


Mid-Level Operations



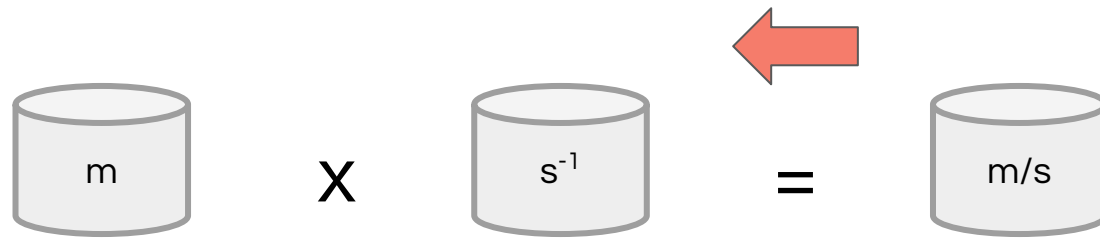
Symbolic manipulation and pragmatic ontologies

Semantically-meaningful Data

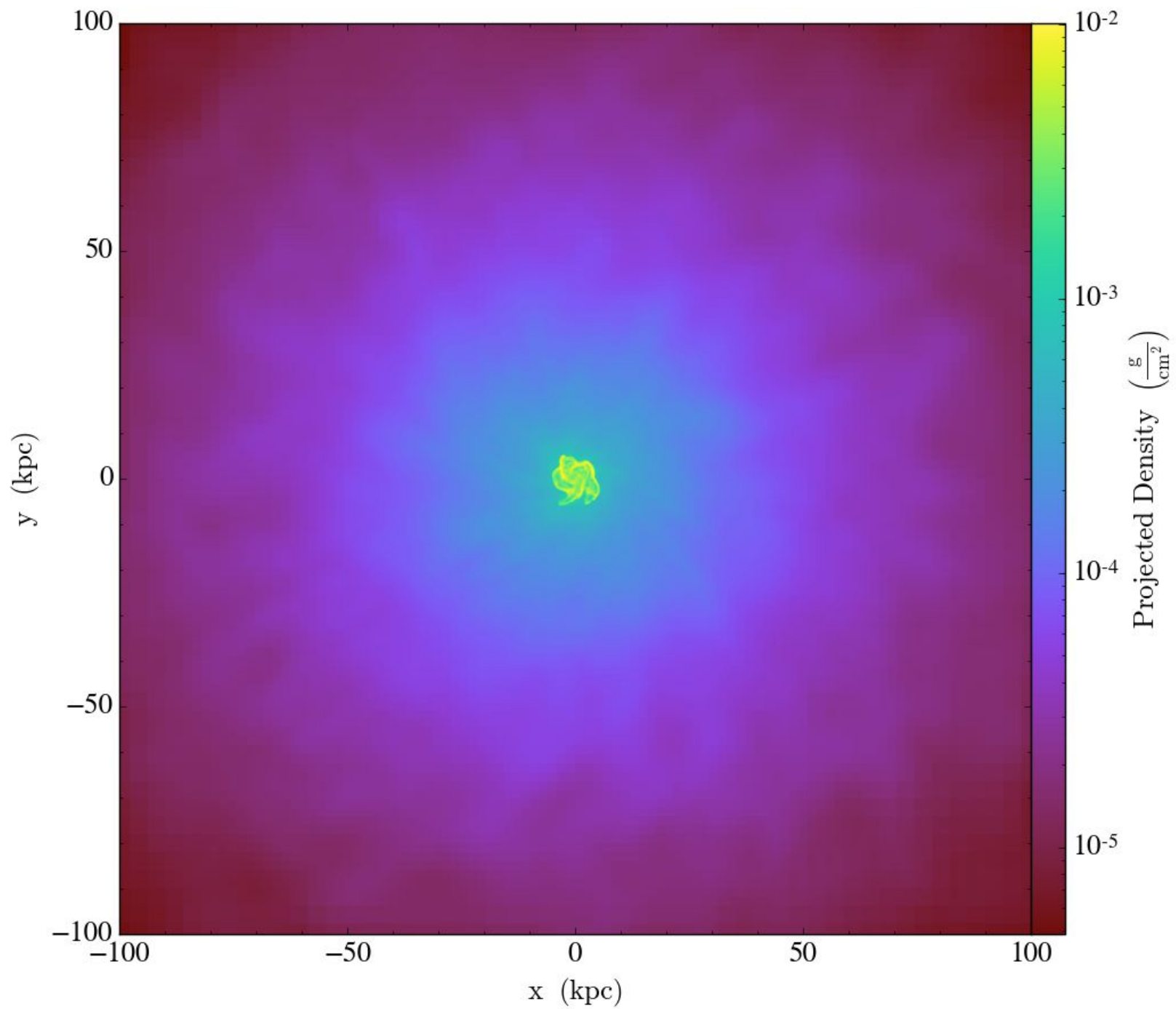


Symbolic manipulation and pragmatic ontologies

Semantically-meaningful Data



Symbolic manipulation and pragmatic ontologies

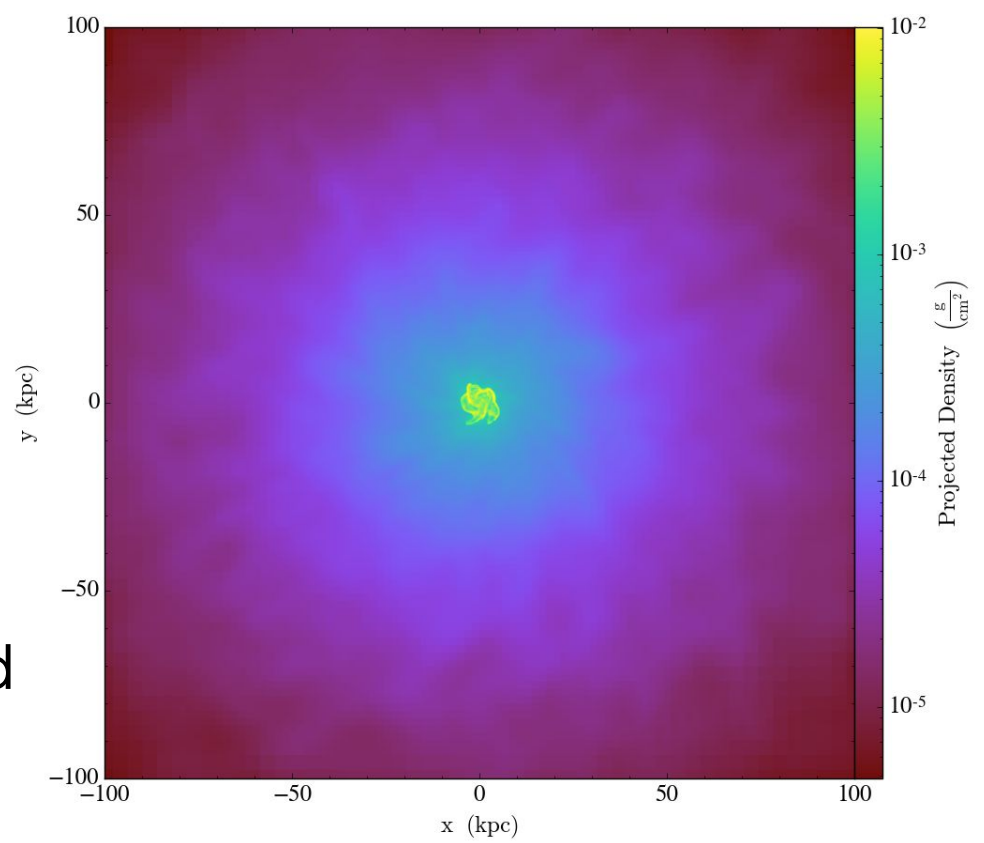
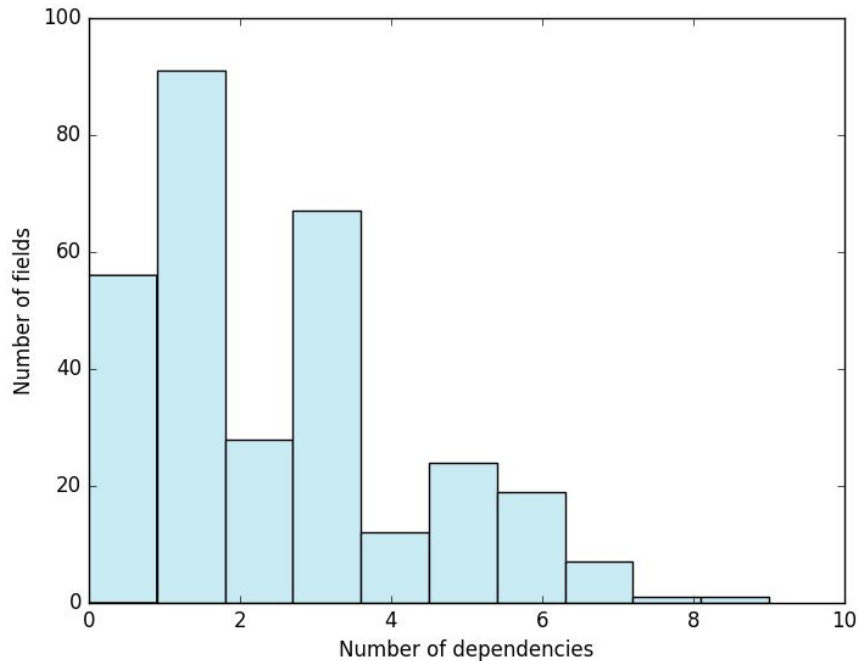


43 primitive fields

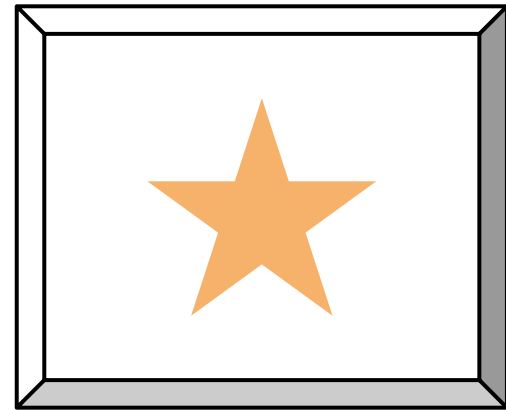
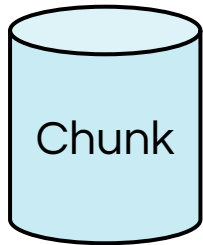
363 derived fields

35 distinct units

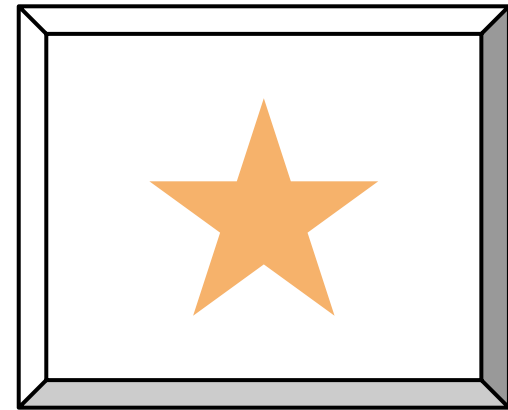
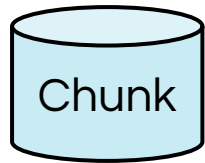
2.5 primitive fields per derived



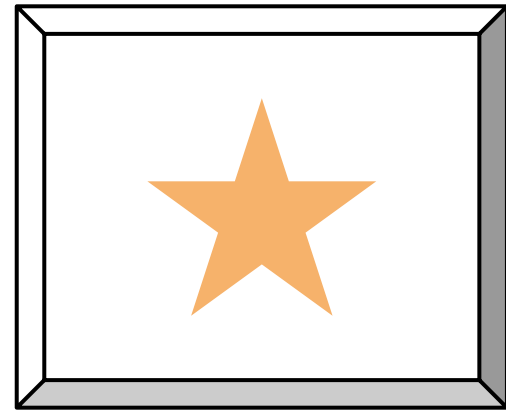
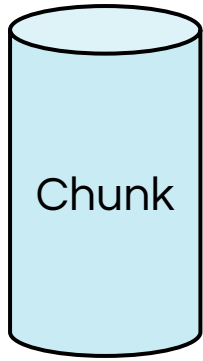
High-Level Operations



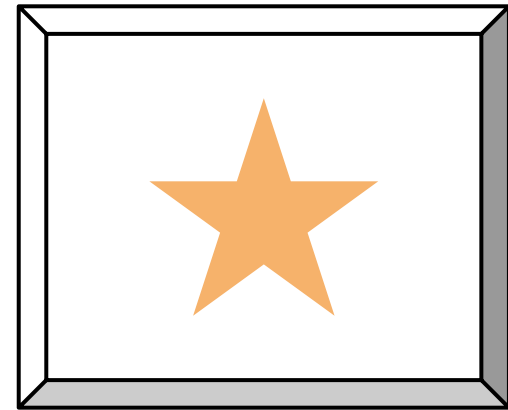
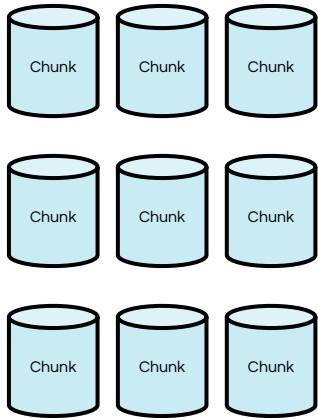
High-Level Operations



High-Level Operations

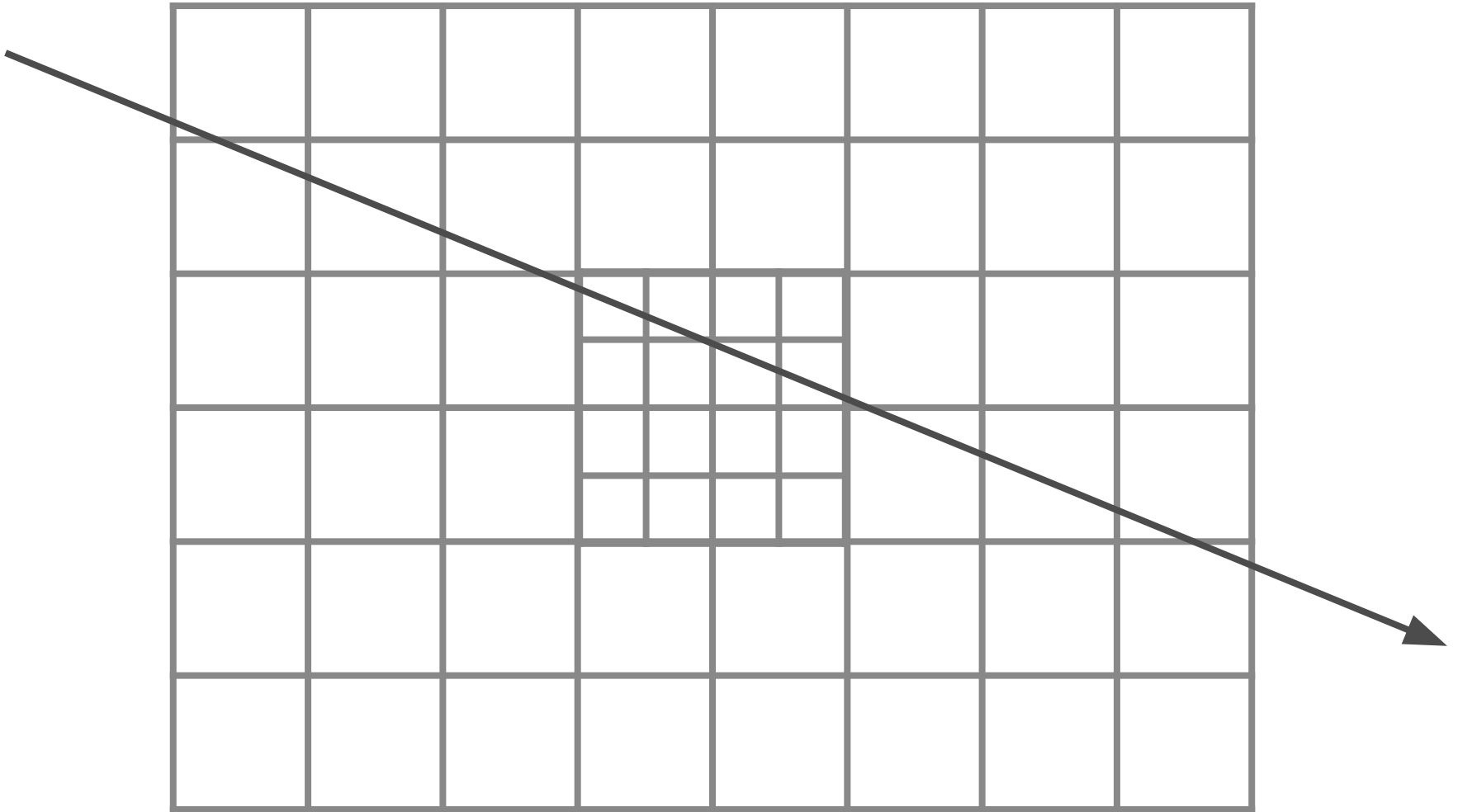


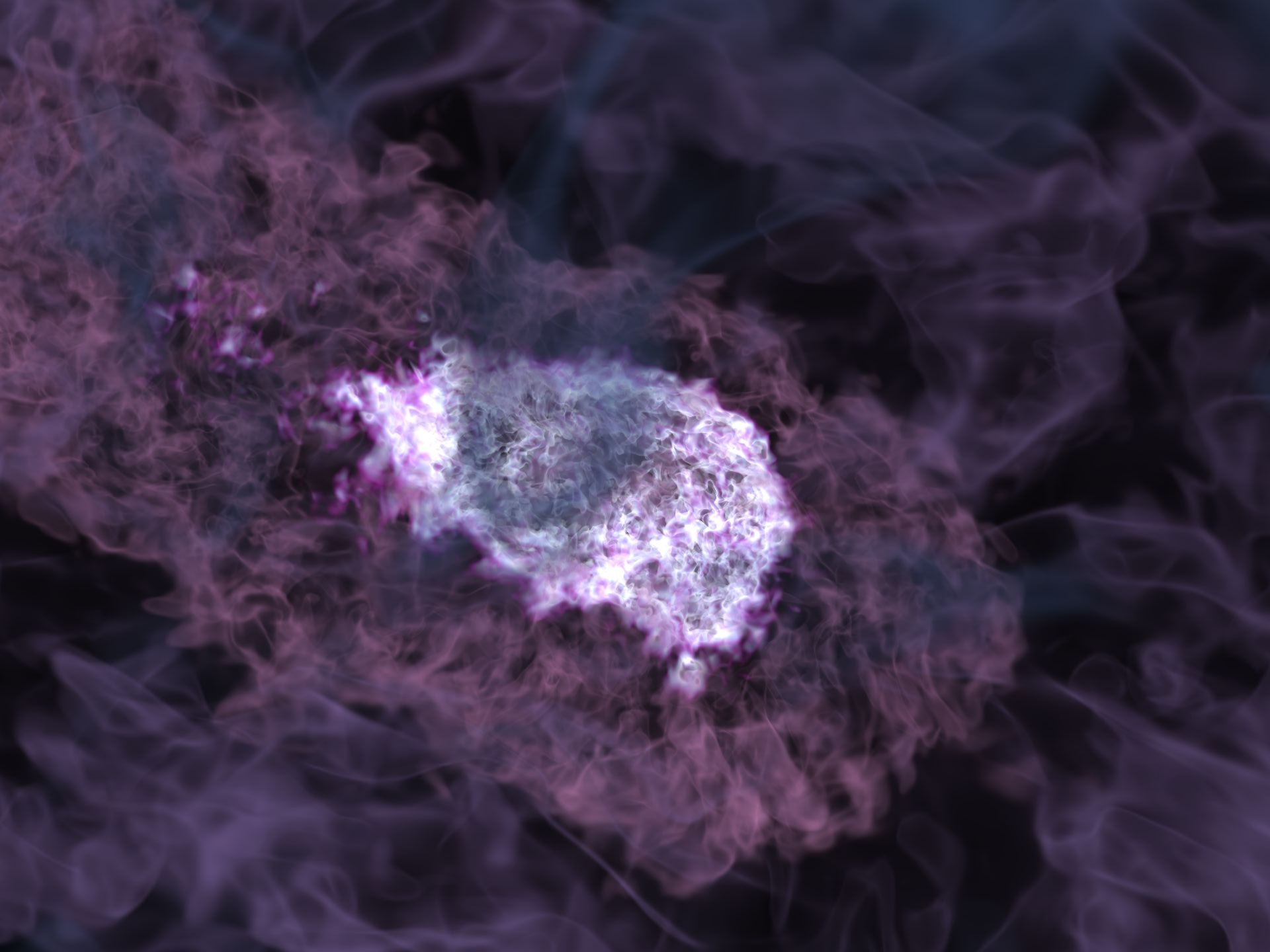
High-Level Operations

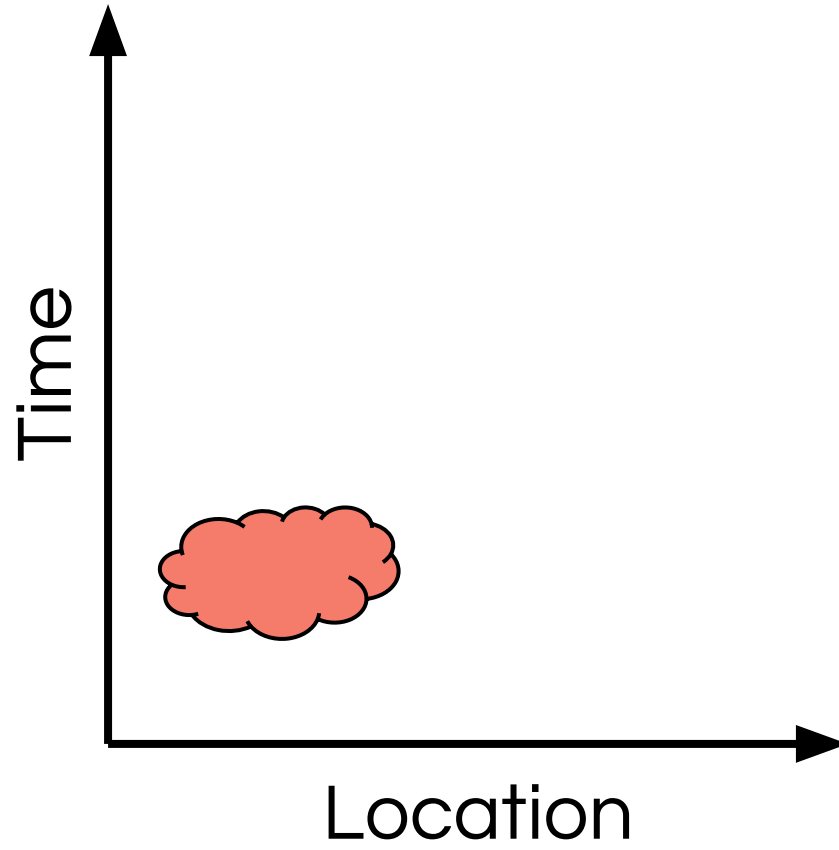


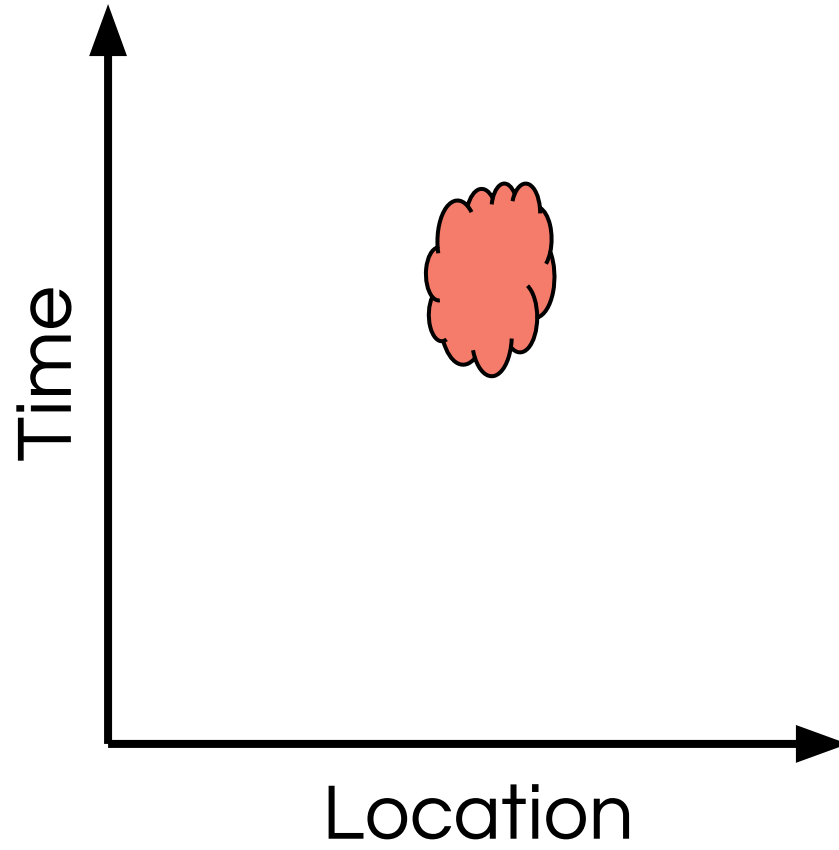
Imaging and volume tools

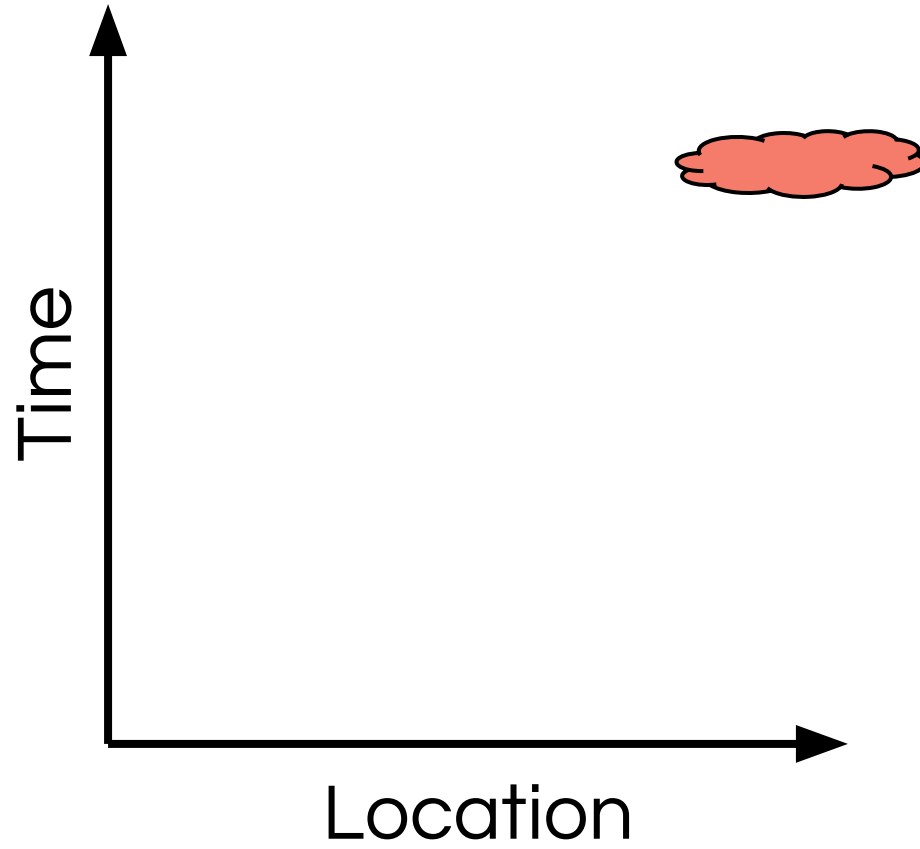
- Volumetric segmentation
 - Parallel
 - Irregular resolution data
- Marching cubes
- Ray-tracing
 - Radiative transfer
 - Volume rendering
- Rasterization / pixelization
 - Coordinate systems
 - Discretization

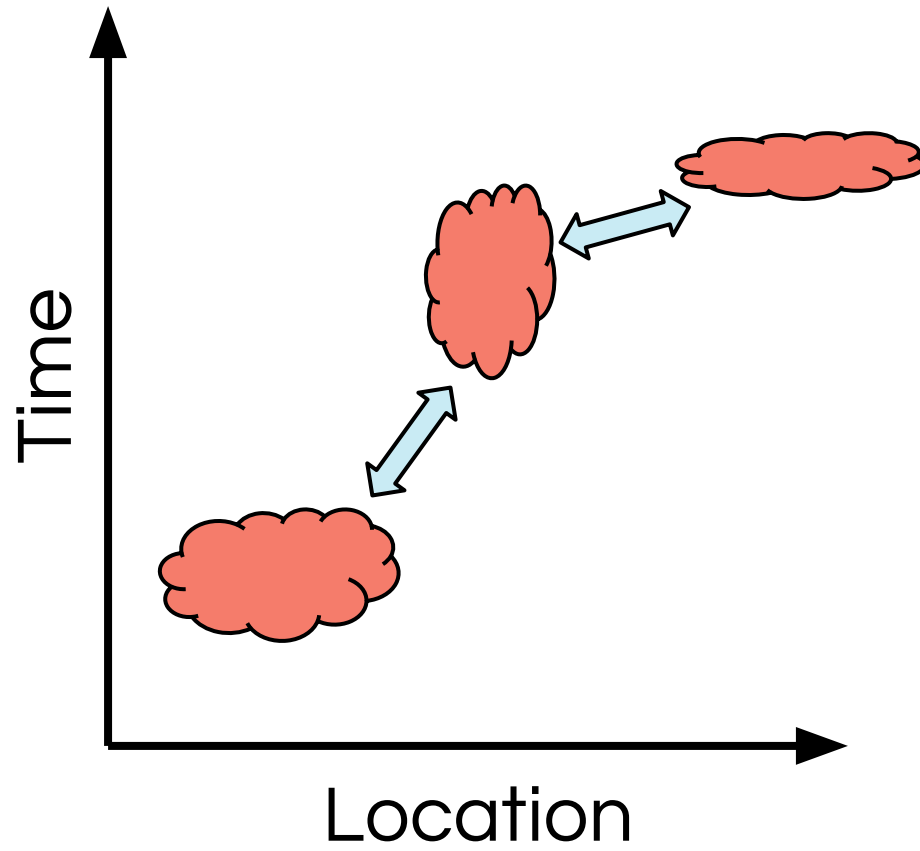


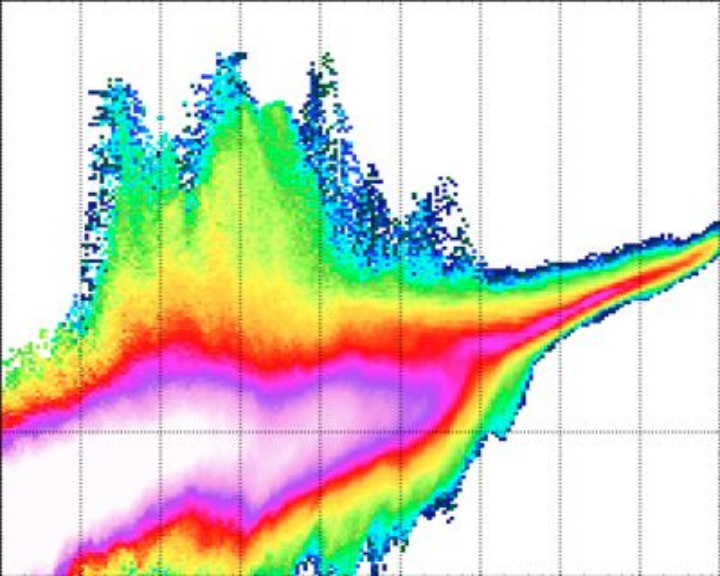






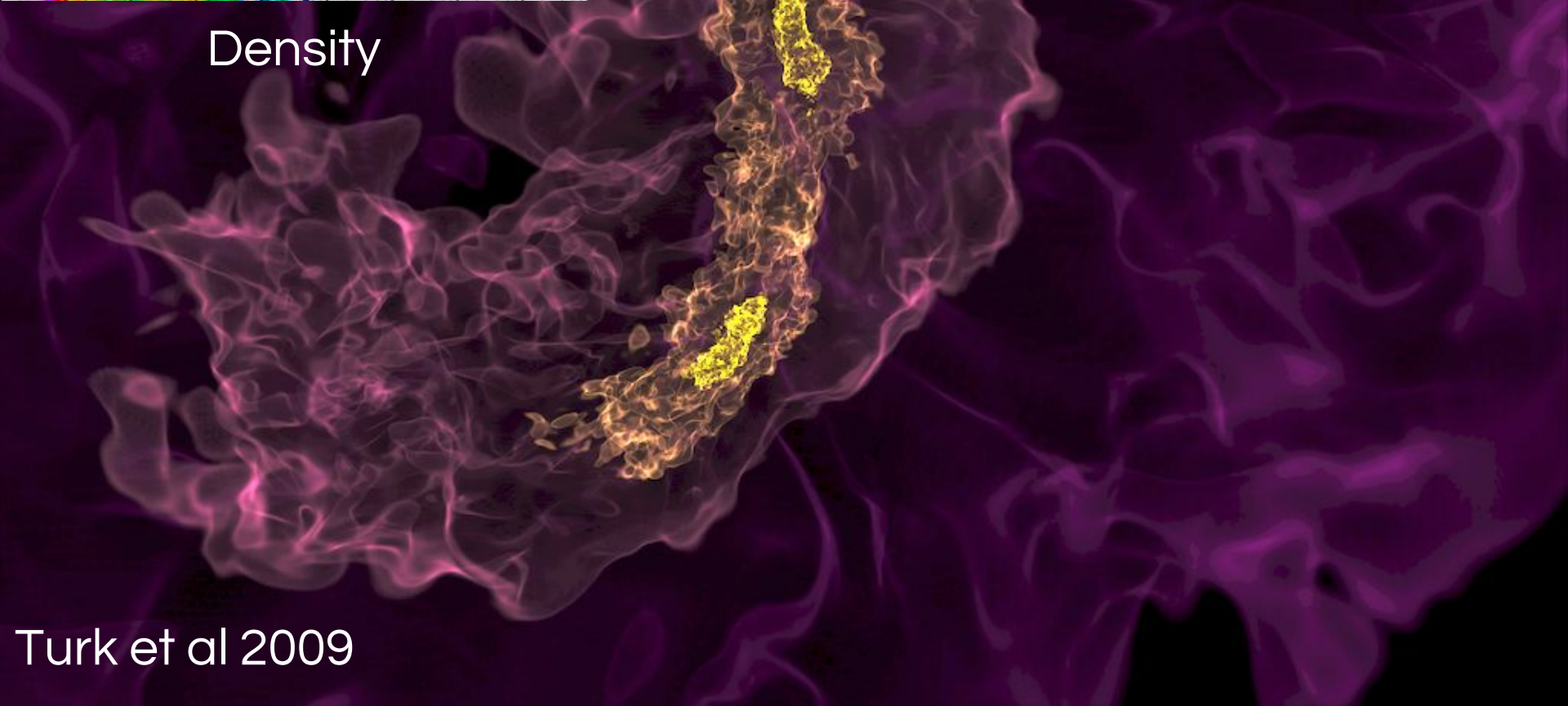




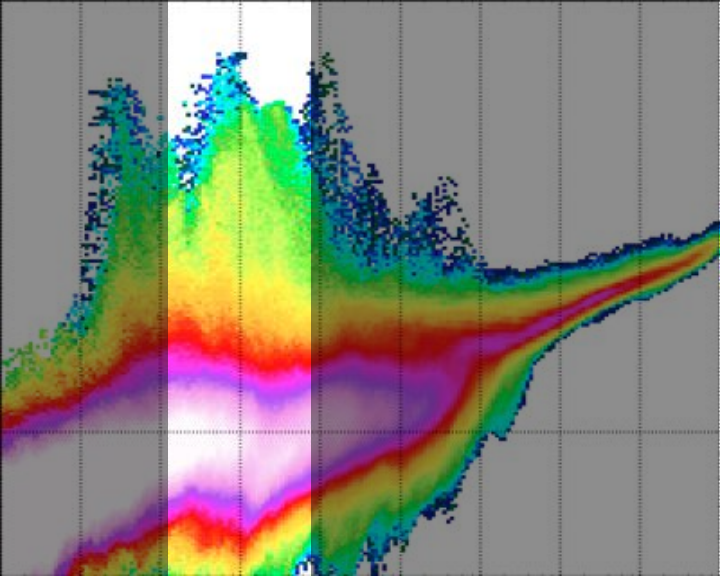


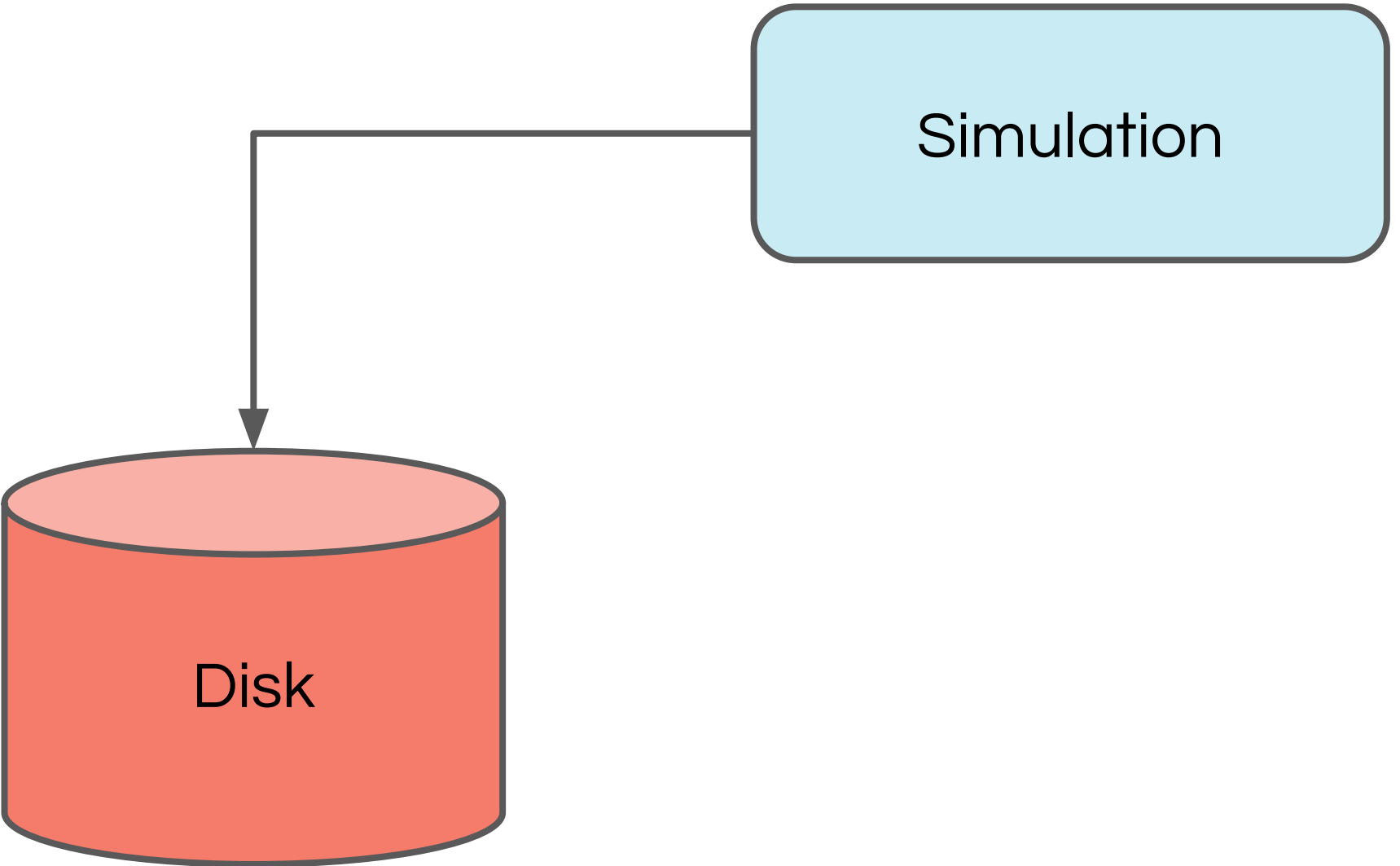
Temperature

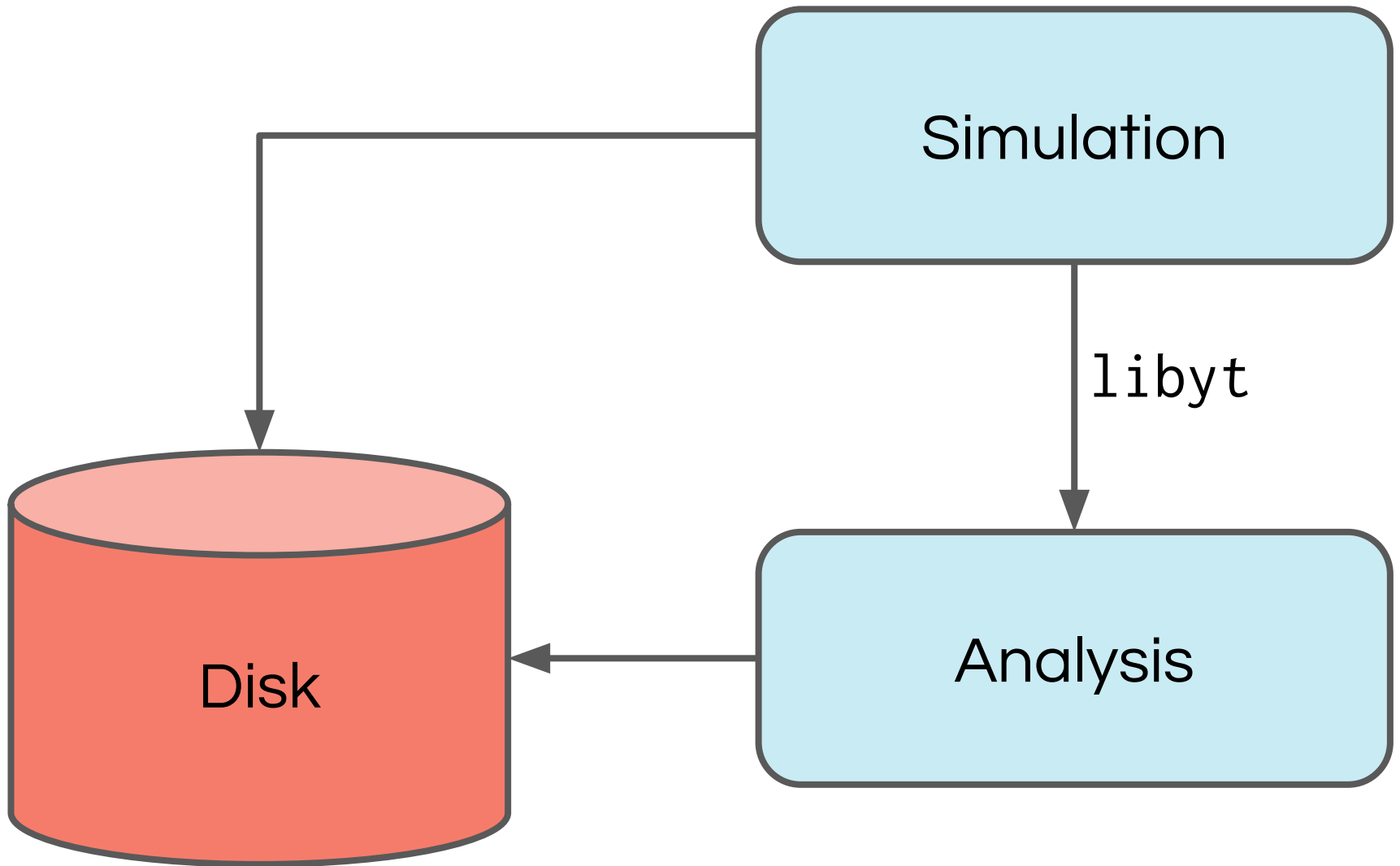
Density



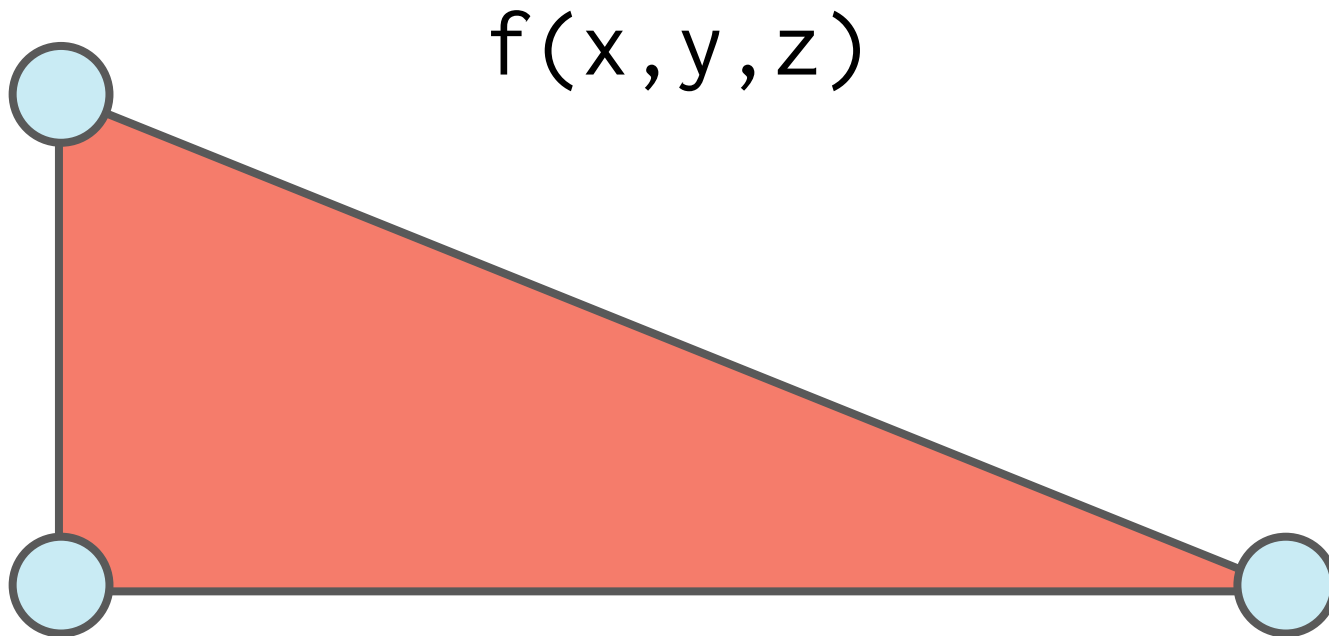
Turk et al 2009



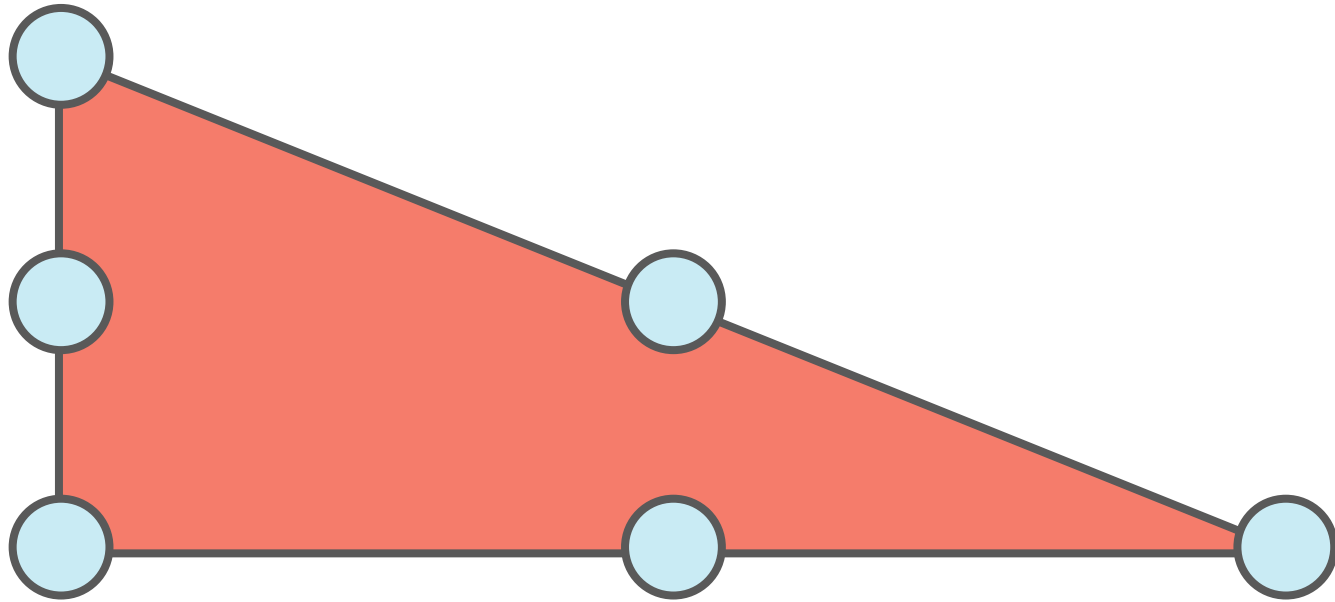




Hsi-Yu Schive

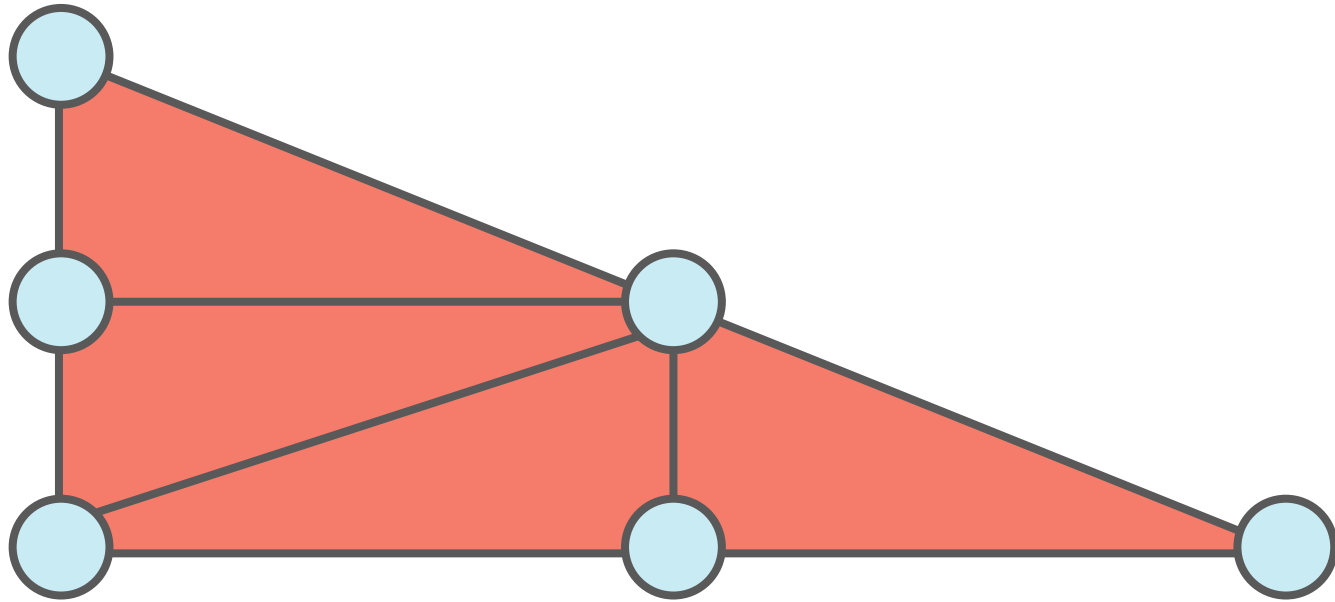


First Order



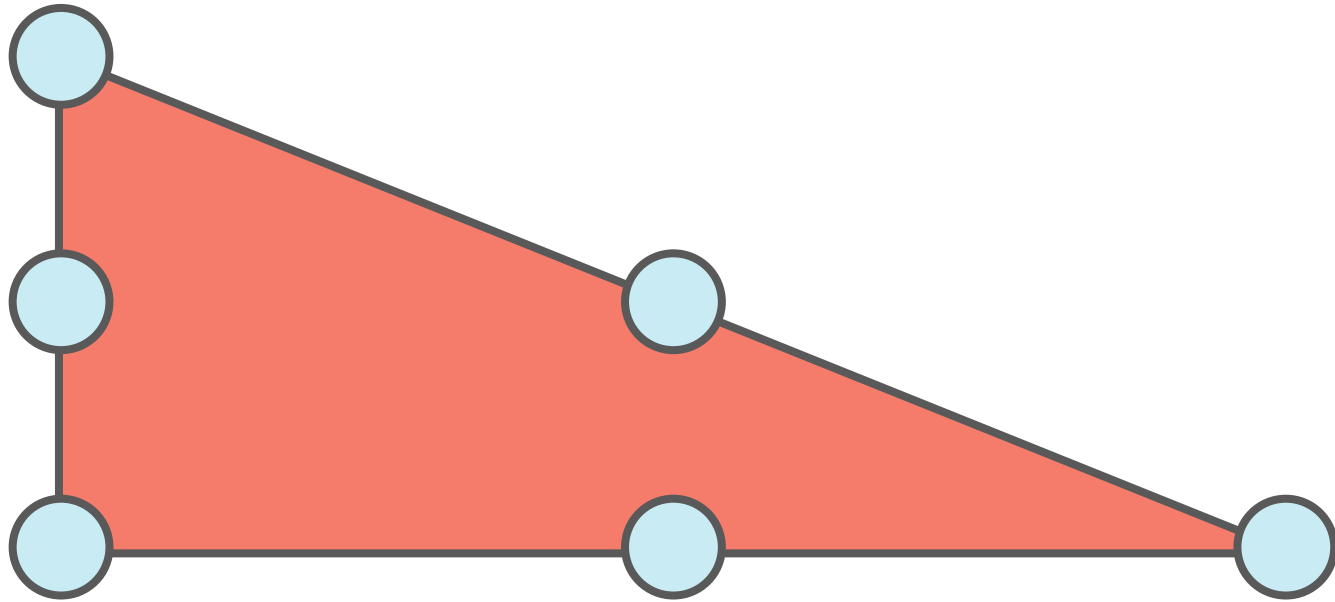
Second Order

Alex Lindsay, Andrew Myers



Approximate Second Order

Alex Lindsay, Andrew Myers



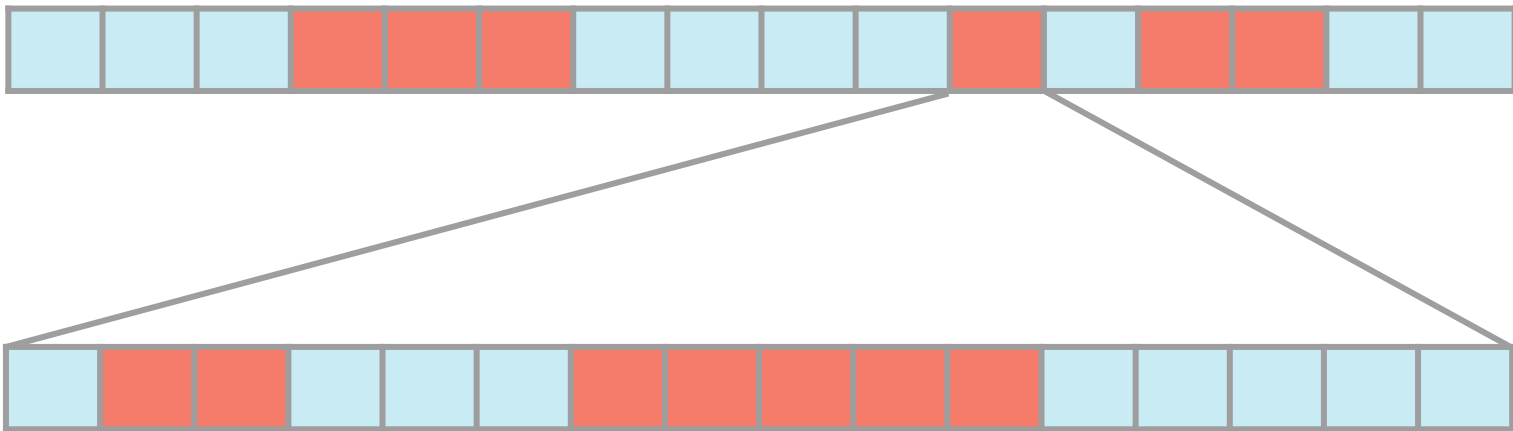
Approximate Second Order

Alex Lindsay, Andrew Myers

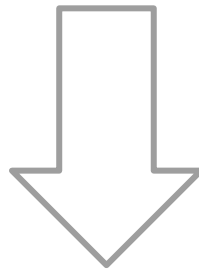
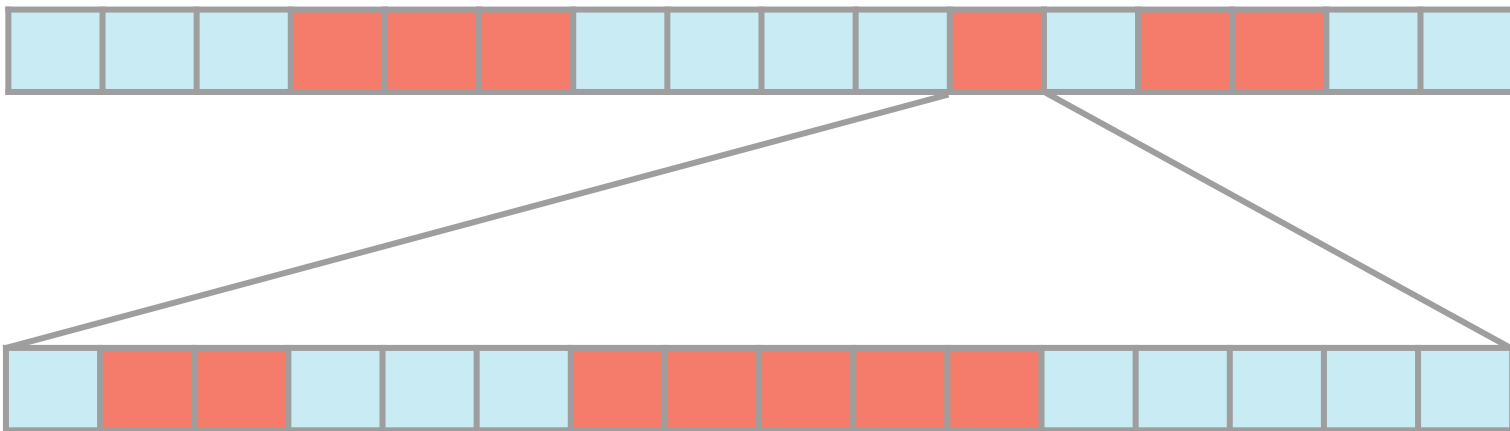
Compressed bitmap indices



Compressed bitmap indices

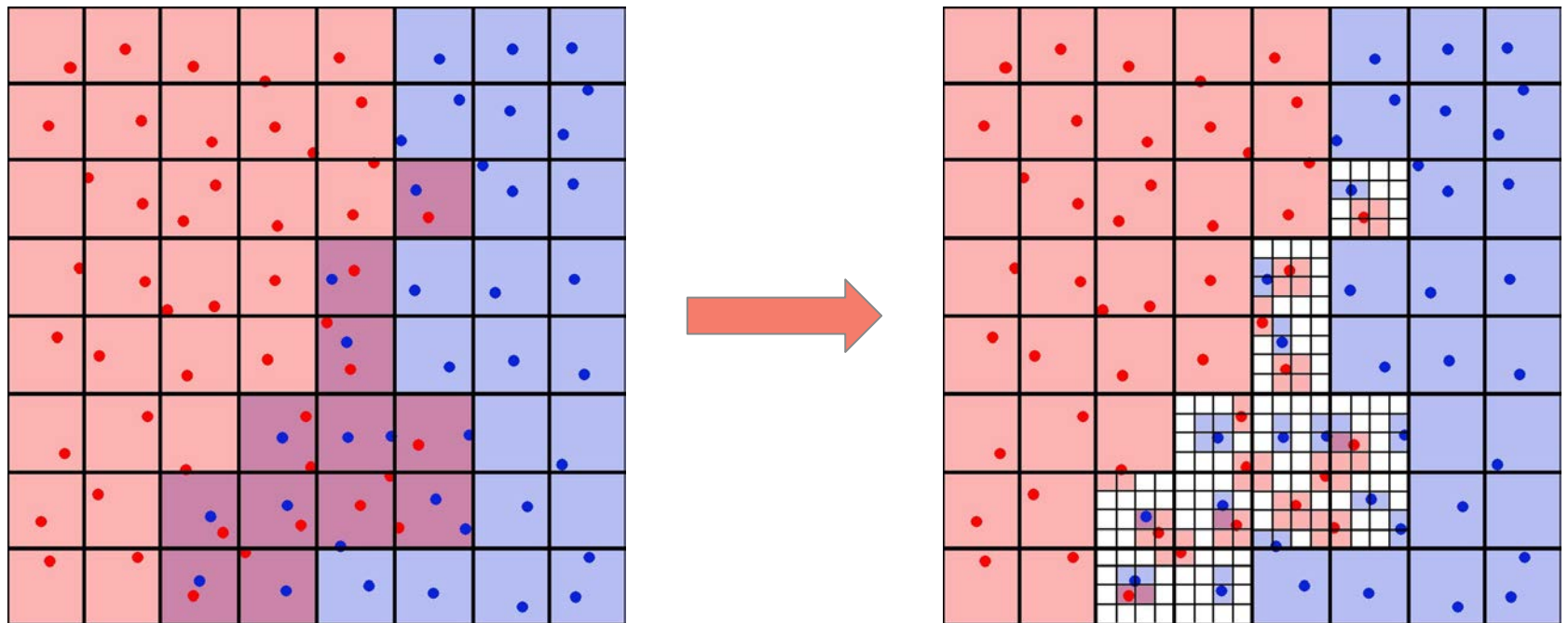


Compressed bitmap indices



Spatial Tree

Compressed bitmap indices



Community

- Not the biggest, not the smallest, but active
 - 375 on the “users” mailing list
 - 115 on the “dev” mailing list
- Code
 - Peer review
 - Mentorship
 - Continuous testing system

Community

- How can we increase diversity?
- How can we foster careers?
- How can we lower barriers?

What are our core values?

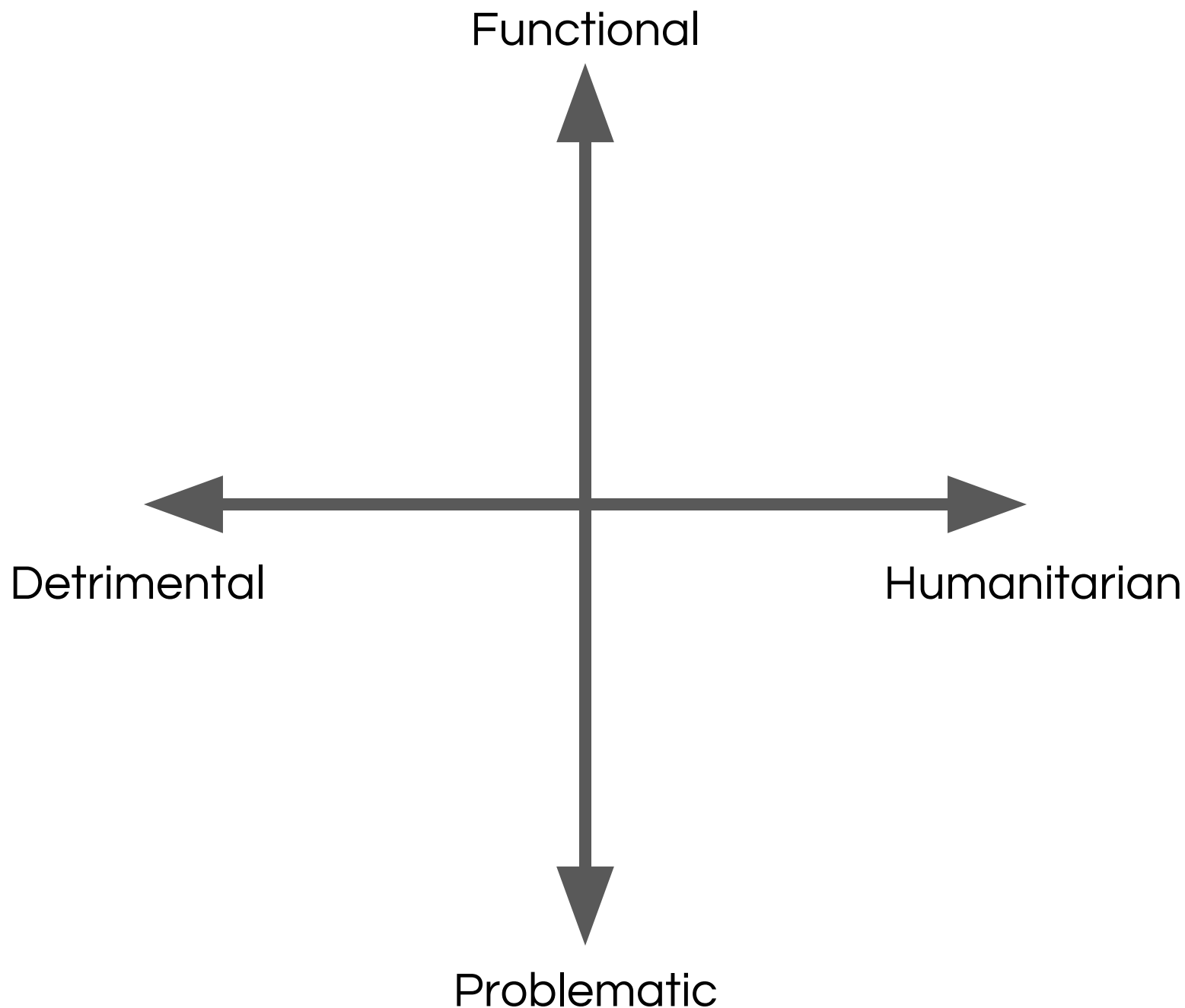
Technically Easy



Socially Challenging

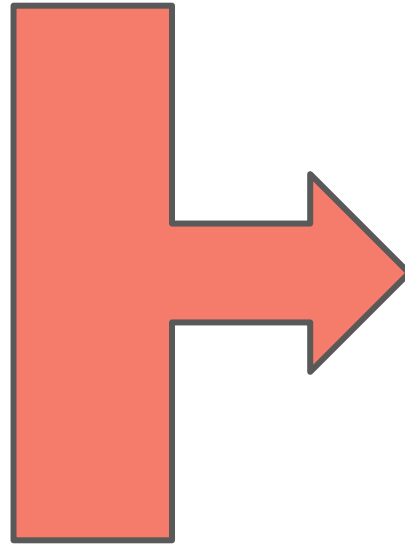
Socially Easy

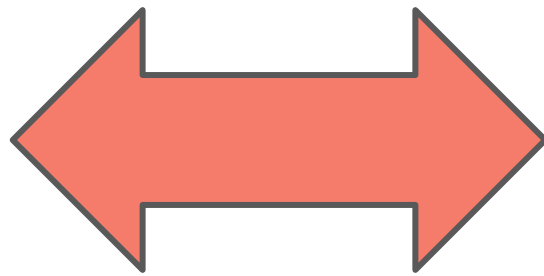
Technically Challenging



product versus project

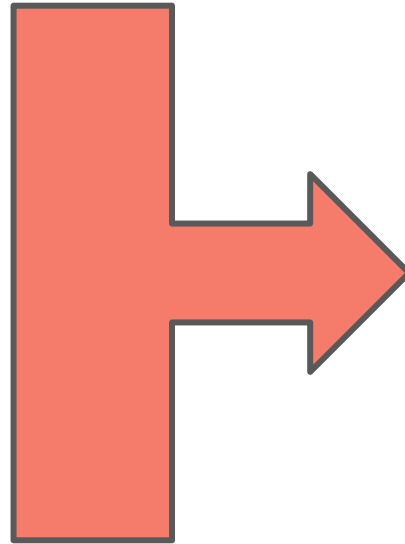
product



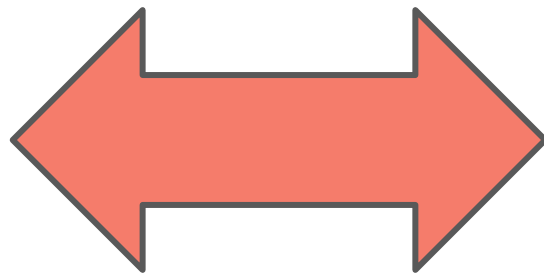


project

product



“the thing”



project

“the people”

Thank you.

mjturk@illinois.edu

<http://yt-project.org/>

<http://dxl.ncsa.illinois.edu/>

<http://sites.google.com/site/matthewturk/>

Three Options:

```
$ docker pull xarthisius/ythub-jupyter
```

```
$ docker run --rm -ti -p 8888:8888 xarthisius/ythub-jupyter
```

```
$ conda install -c conda-forge yt
```

yt-project.org and click on "Get yt"