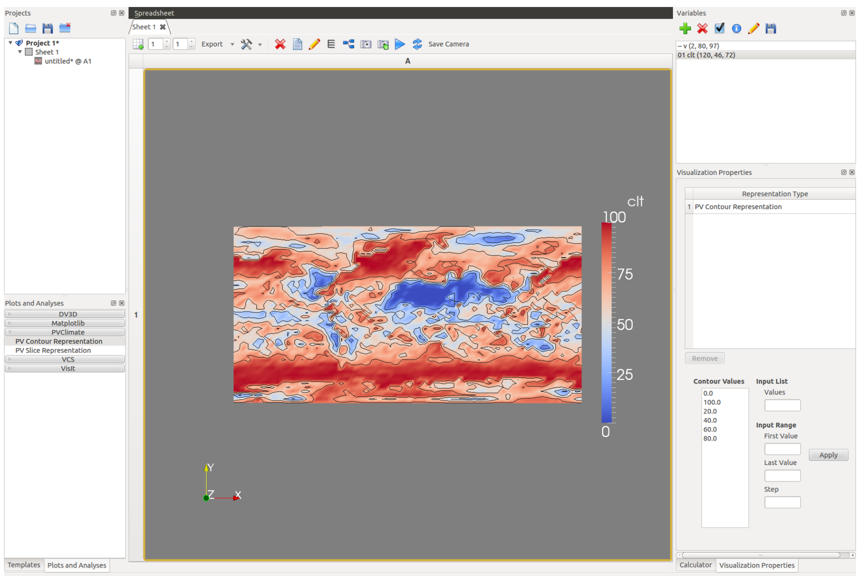
ParaView

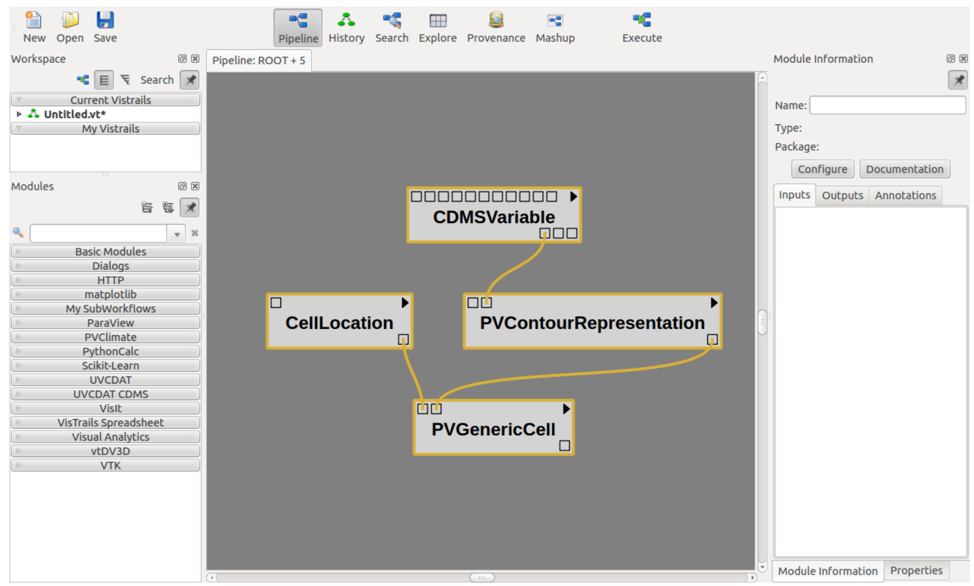
UV-CDAT ParaView Integration

- Tight Coupling
  - ParaView within VisTrails workflow
    - Created ParaView's VisTrails module such as PVContourRepresentation and PVGenericCell that can have 1 or more representations on its input.
    - ParaView pipeline helper builds the plot pipeline and create instances of ParaView modules
  - Provenance
    - Tightly coupled integration benefits from VisTrails provenance
  - Custom interface for Climate Scientists
    - Simple / domain specific interface
    - Easy to create new representations
  - Supports CDMS Variable

UV-CDAT ParaView Integration

- New Readers and Filters
  - Unstructured POP Reader
  - MOC Reader
  - Project Sphere Filter
- Serial Integration





UV-CDAT ParaView Integration

- Parallel Integration
  - Uses pvserver for remote file browsing, pvbatch (ParaView python interpreter specialized for batch processing) and MoleQueue (remote job submission application)
  - Workflow (For a particular use case)
    - User creates a visualization
    - User selects input / output location, dataset, and job queue
    - User then submits the job
    - User analyzes the output when MoleQueue notifies the user at the completion of the job

MOC and MHT

- Meridional Overturning Circulation (MOC) – measure of overturning within ocean currents
- Meridional Heat Transport (MHT) – transport of heat from low to high latitudes
- Was not computed by ocean scientists because of costly compute time
- Created parallel implementations as ParaView filters within UV-CDAT
  - Running time decreased from a few hours to a couple of minutes

Spatio-Temporal Pipeline: UV-CDAT Use Case 1

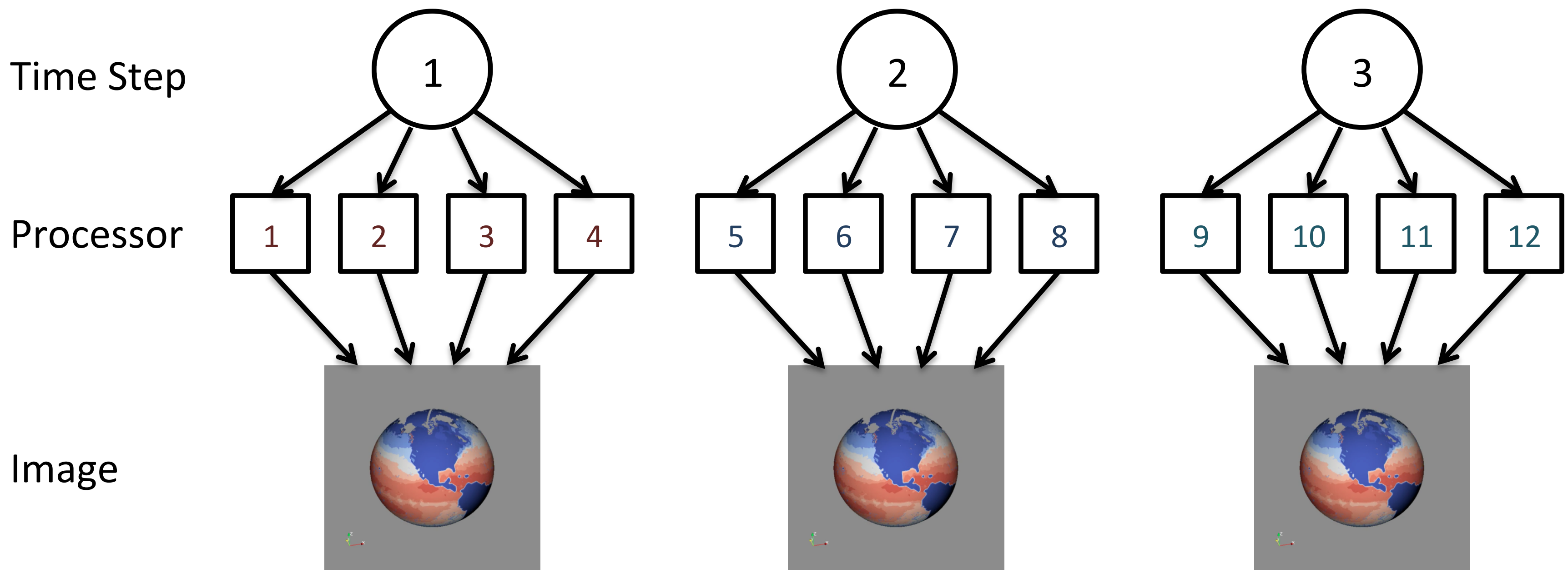
Use Case 1: High spatial resolution, high temporal resolution, image sequence production

Problem

Large datasets exist with many timesteps and high temporal resolution. UV-CDAT must be capable of handling these datasets. Existing tools do not support high temporal resolution well.

Solution

Added capability within UV-CDAT ParaView to partition within time to allow for multiple timesteps to be processed in parallel. Processors are divided into "time compartments", and each file is processed by a time compartment.



Use Case 1 Performance Results: Mustang Test

Number of Timesteps	Number of Processes (P)	Time Compartment = P (seconds)	Time Compartment = 8 (seconds)
2	16	46.96	21.76
4	32	81.84	21.47
8	64	159.77	21.16
16	128	235.61	26.85
32	256	1,103.00	23.13
64	512	2,365.89	25.02
128	1024	8,128.92 (~2 hrs)	30.15
256	**2048**	**28,862.55 (~8 hrs)**	**62.83**

- Measured on Mustang supercomputer, 8 cores per node
- Each time step is 1.4 GB
- Panasas parallel file system
- Testing having all processors read each time step versus having eight processors read each time step

Spatio-Temporal Pipeline: UV-CDAT Use Case 2

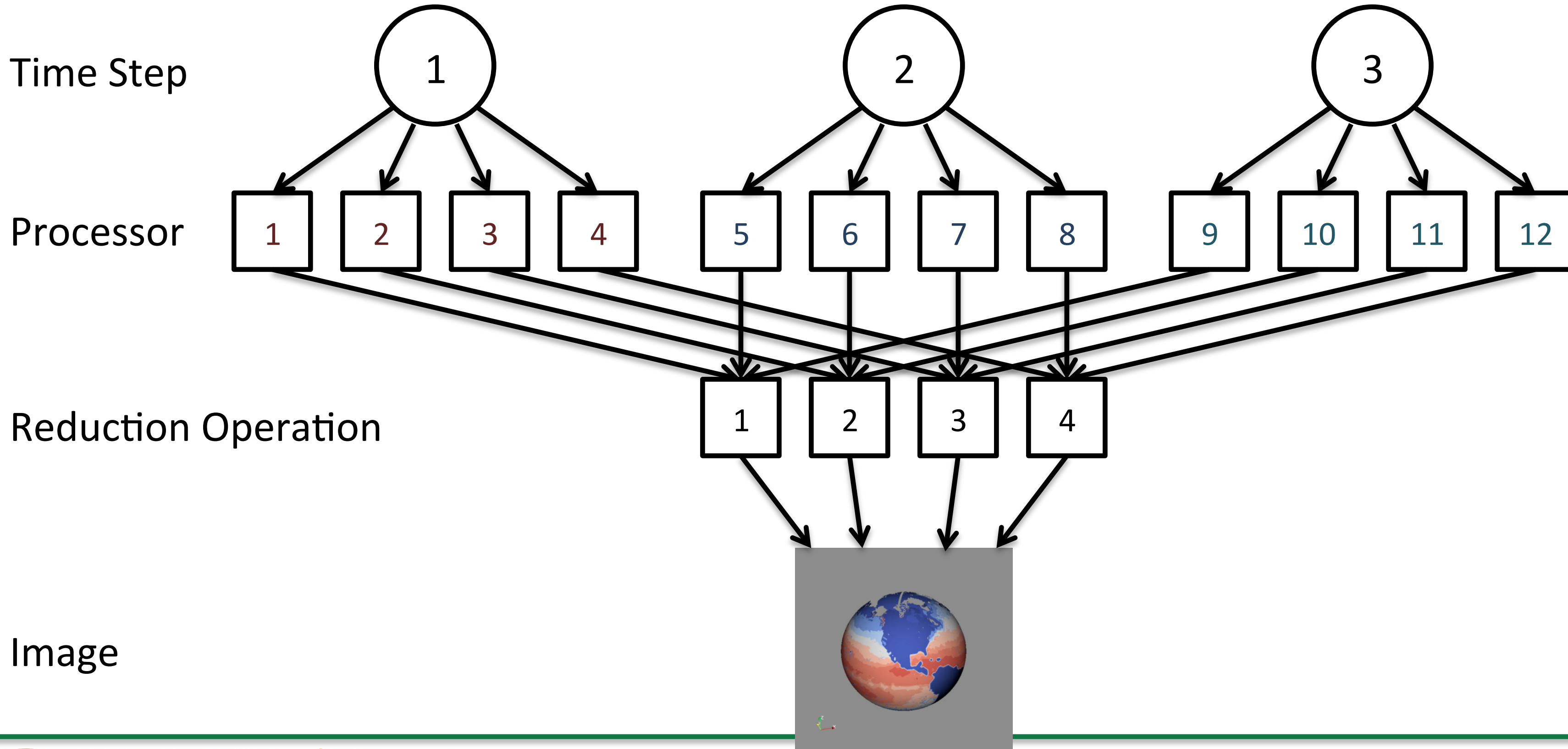
Use Case 2: High spatial resolution, high temporal resolution, time average

Problem

Multiple timesteps need to be averaged together to produce a data product based on the results.

Solution

Added capability within UVCDAT ParaView to take multiple timesteps and compute various statistics (average, min, max, standard deviation) using the spatio-temporal pipeline.



Use Case 2 Performance Results: Hopper Tests

Time Compartment Size	Total Time (seconds)
1	145
2	278
12	**93**
48	151
96	244
240	525
480	1204

- Measured on Hopper supercomputer
- 480 cores, 20 nodes
- Analysis of Michael Wehner's climate data
- 324 timesteps, total data size is 20 GB
- Calculate yearly statistics from monthly data
    - Min, max, average, standard deviation
- Lustre parallel file system