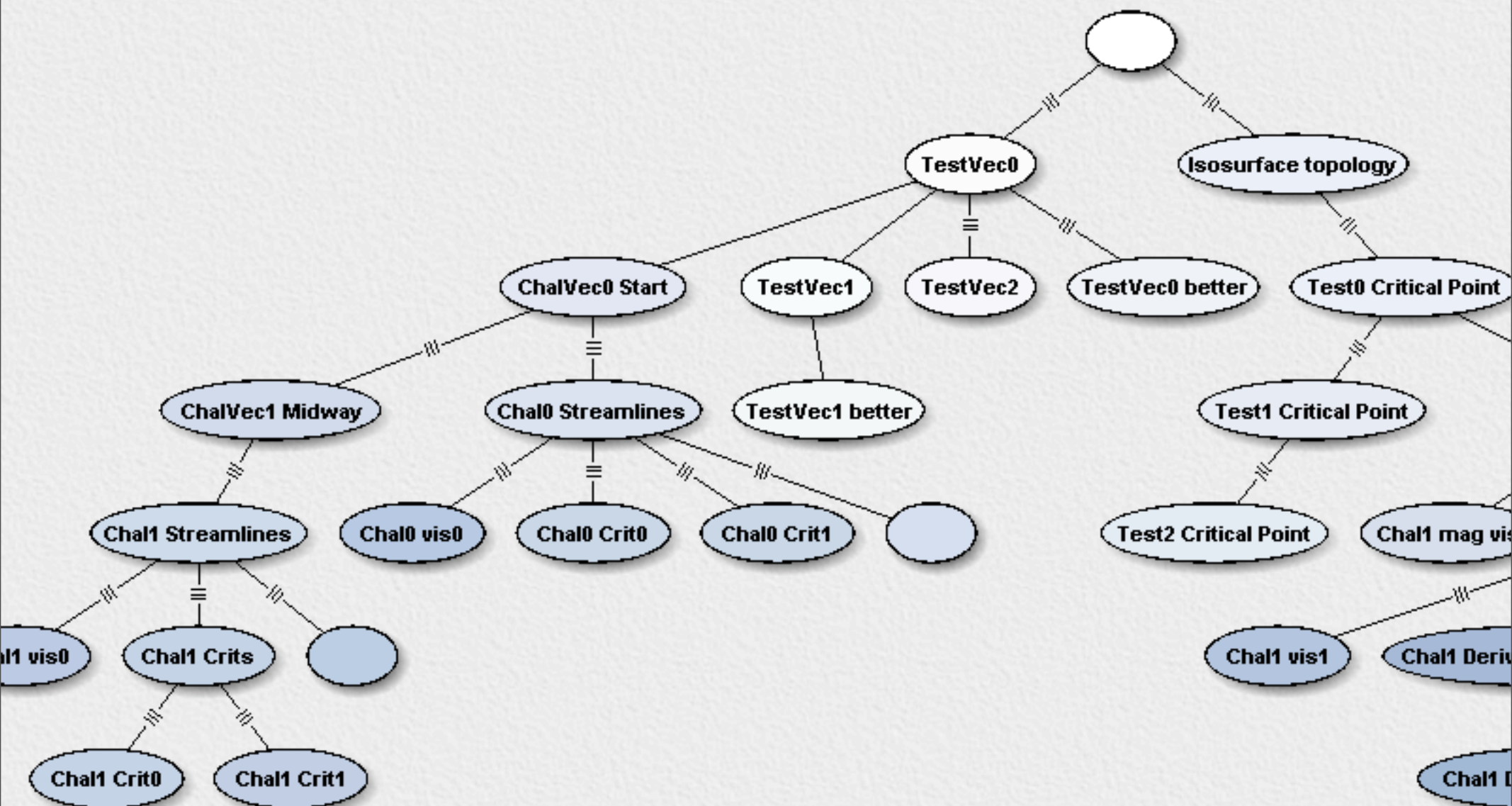# VisTrails: Process Provenance for SciVis
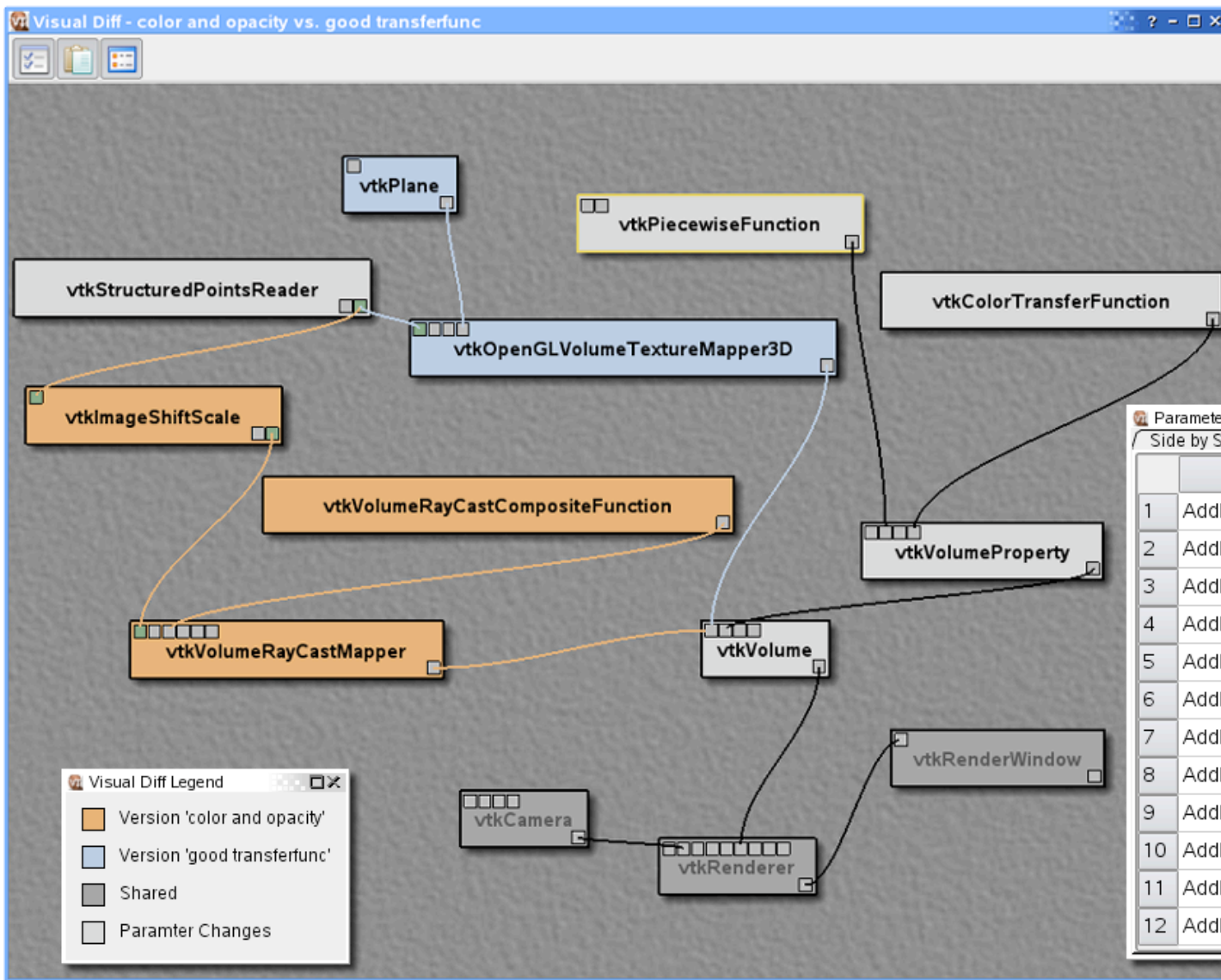
GGF18 Provenance Challenge Entry

Scientific Computing and Imaging Institute, School of Computing - University of Utah

Erik Anderson, Steve Callahan, Juliana Freire, Emanuele Santos, Cláudio Silva, Carlos Scheidegger, Huy Vo

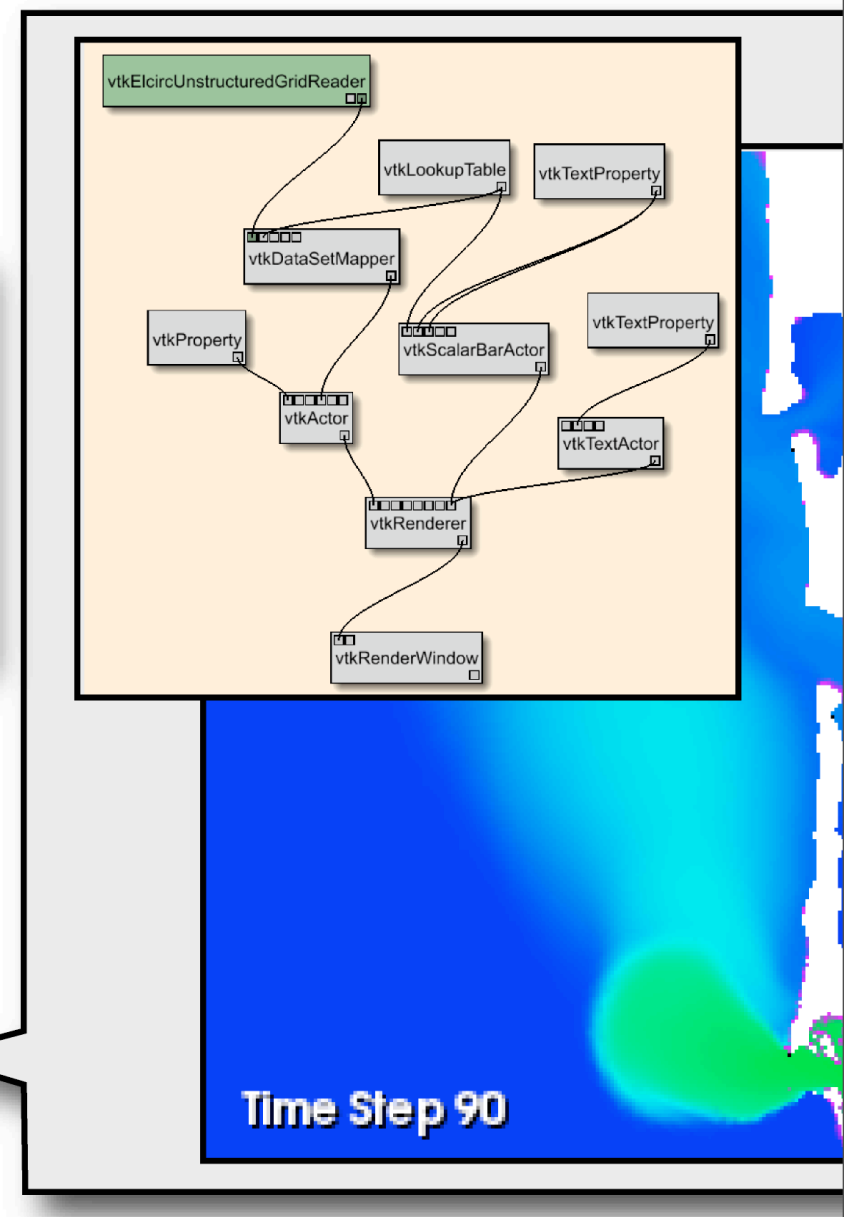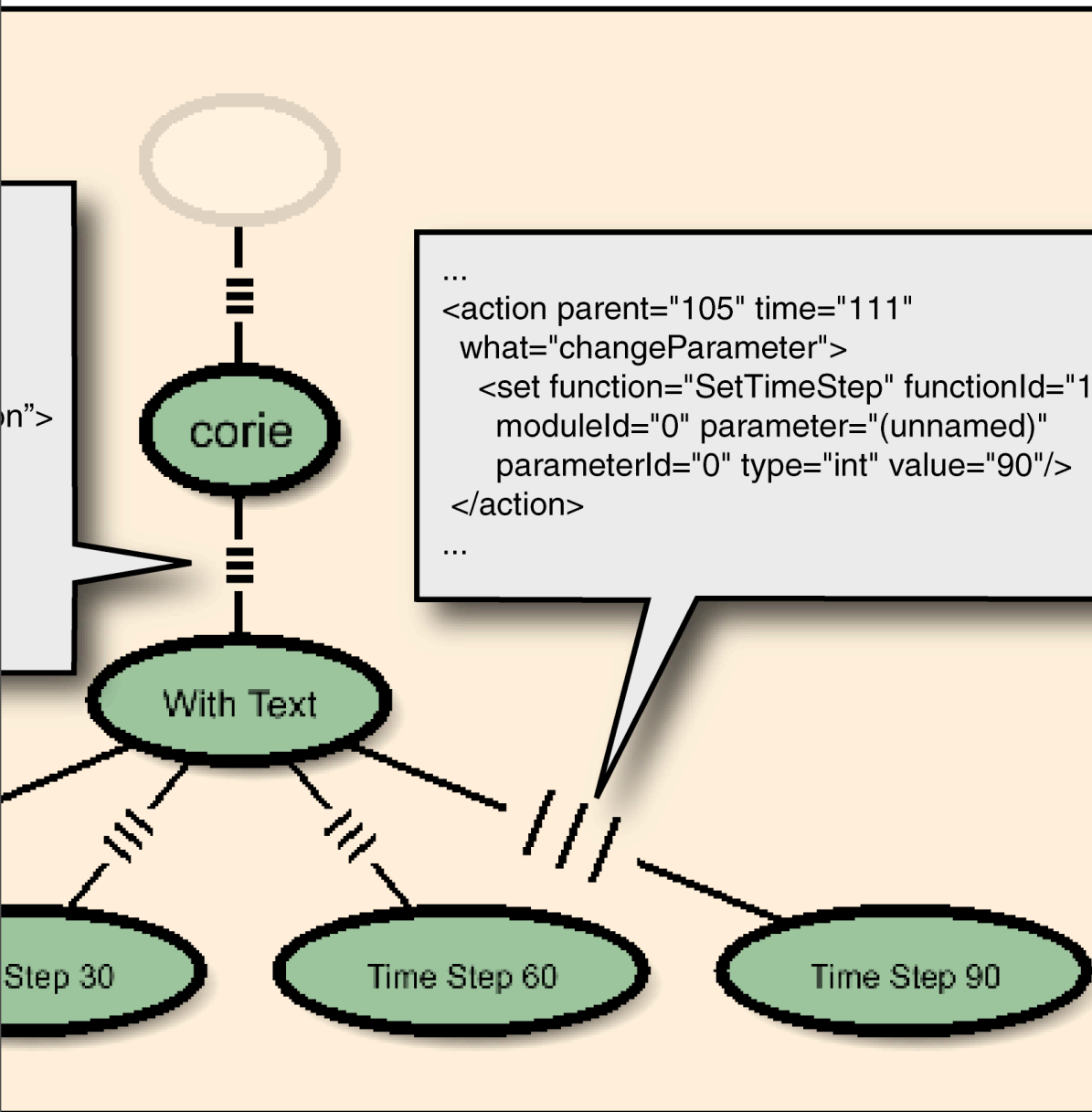Process Provenance

Process Evolution
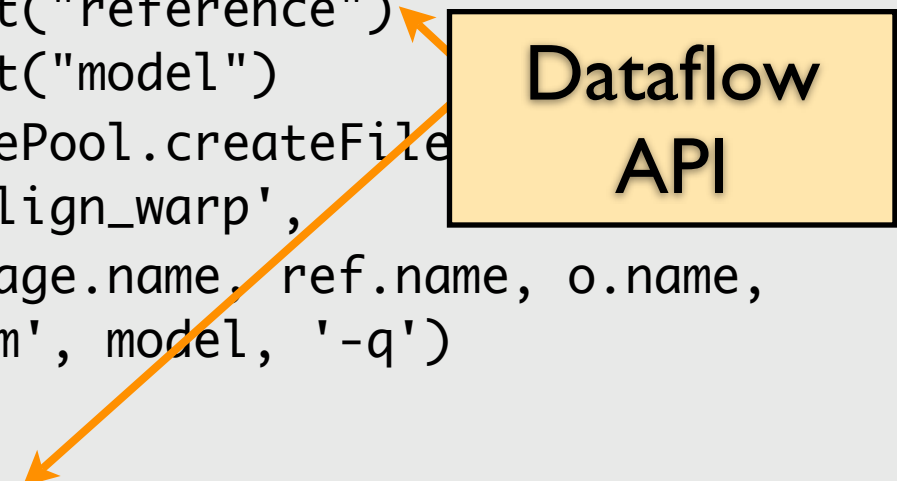
Version Tree, Workflows

The VisTrails Plugin architecture

```python
class AlignWarp(ProvenanceChallenge):

    def compute(self):
        image = self.getInputFromPort("image")
        ref   = self.getInputFromPort("reference")
        model = self.getInputFromPort("model")
        o     = self.interpreter.filePool.createFile         )
        cmd   = self.air_cmd_line('align_warp',
                                  image.name, ref.name, o.name,
                                  '-m', model, '-q')

        self.run(cmd)
        self.setResult("output", o)
```
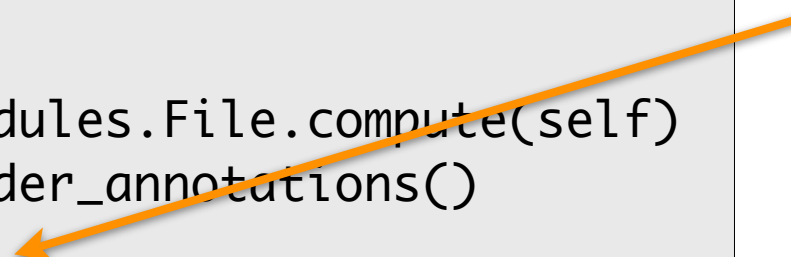
Dataflow API

Each module is just
Python source

The VisTrails Plugin architecture

```
class AIRHeaderFile(modules.basic_modules.File):

    def get_header_annotations(self):
        ....

    def compute(self):
        modules.basic_modules.File.compute(self)
        d = self.get_header_annotations()
        self.annotate(d)
```

Annotation API

Each module is just Python source
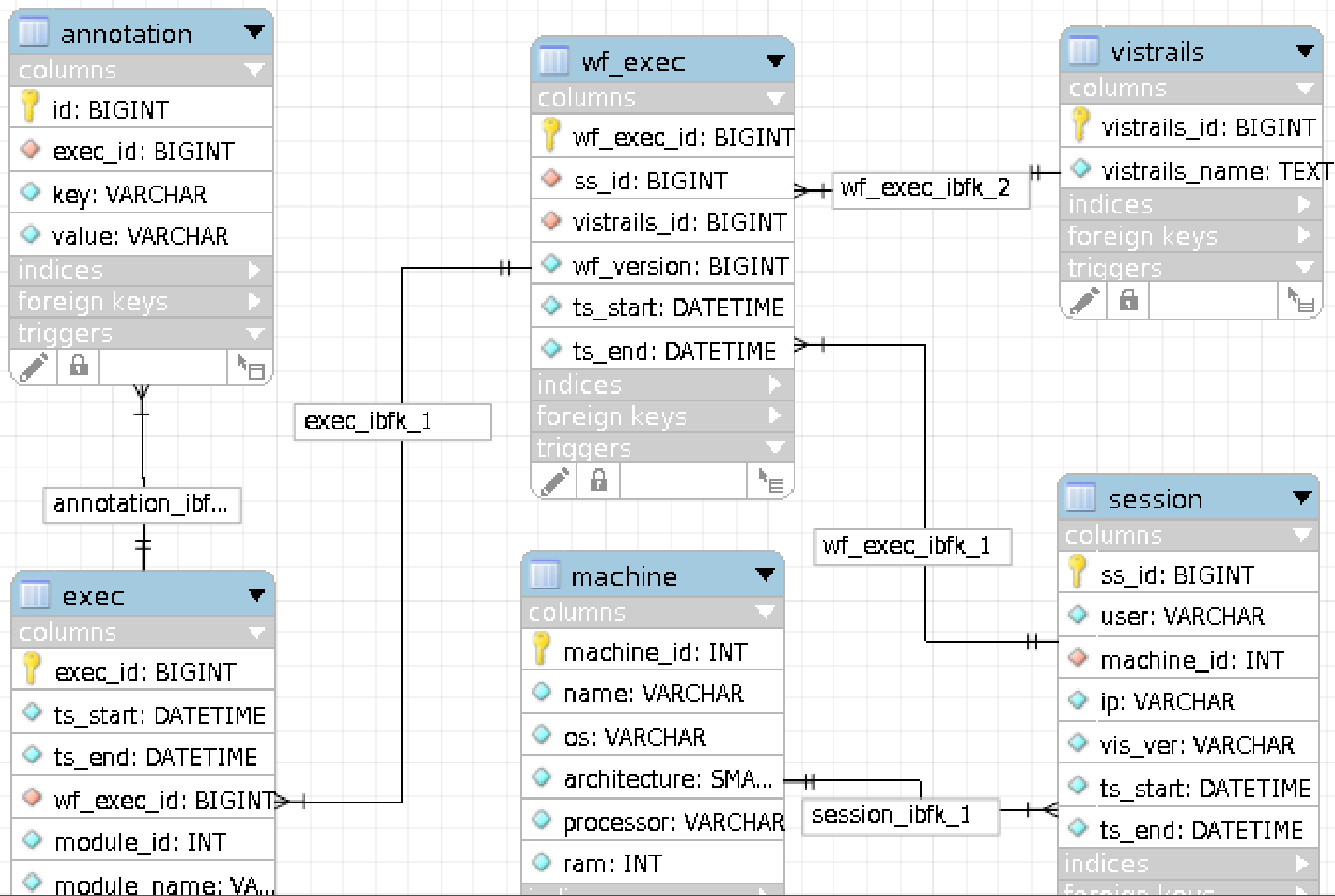
The VisTrails Plugin architecture

```
addModule(AlignWarp)
addInputPort(AlignWarp, "image", ...)
addInputPort(AlignWarp, "image_header", ...)
addInputPort(AlignWarp, "reference", ...)
addInputPort(AlignWarp, "reference_header", ...)
addInputPort(AlignWarp, "model", ...)
addOutputPort(AlignWarp, "output", ...)
```

Registering
a module
within VisTrails

Each module is just
Python source

DB for runtime info

annotation
columns
- 🔑 id: BIGINT
- 🔶 exec_id: BIGINT
- 🔷 key: VARCHAR
- 🔷 value: VARCHAR

indices
foreign keys
triggers

wf_exec
columns
- 🔑 wf_exec_id: BIGINT
- 🔶 ss_id: BIGINT
- 🔶 vistrails_id: BIGINT
- 🔷 wf_version: BIGINT
- 🔷 ts_start: DATETIME
- 🔷 ts_end: DATETIME

indices
foreign keys
triggers

vistrails
columns
- 🔑 vistrails_id: BIGINT
- 🔷 vistrails_name: TEXT

indices
foreign keys
triggers

wf_exec_ibfk_2

exec_ibfk_1

annotation_ibf...

exec
columns
- 🔑 exec_id: BIGINT
- 🔷 ts_start: DATETIME
- 🔷 ts_end: DATETIME
- 🔶 wf_exec_id: BIGINT
- 🔷 module_id: INT
- 🔷 module_name: VA...

machine
columns
- 🔑 machine_id: INT
- 🔷 name: VARCHAR
- 🔷 os: VARCHAR
- 🔷 architecture: SMA...
- 🔷 processor: VARCHAR
- 🔷 ram: INT

wf_exec_ibfk_1

session_ibfk_1

session
columns
- 🔑 ss_id: BIGINT
- 🔷 user: VARCHAR
- 🔶 machine_id: INT
- 🔷 ip: VARCHAR
- 🔷 vis_ver: VARCHAR
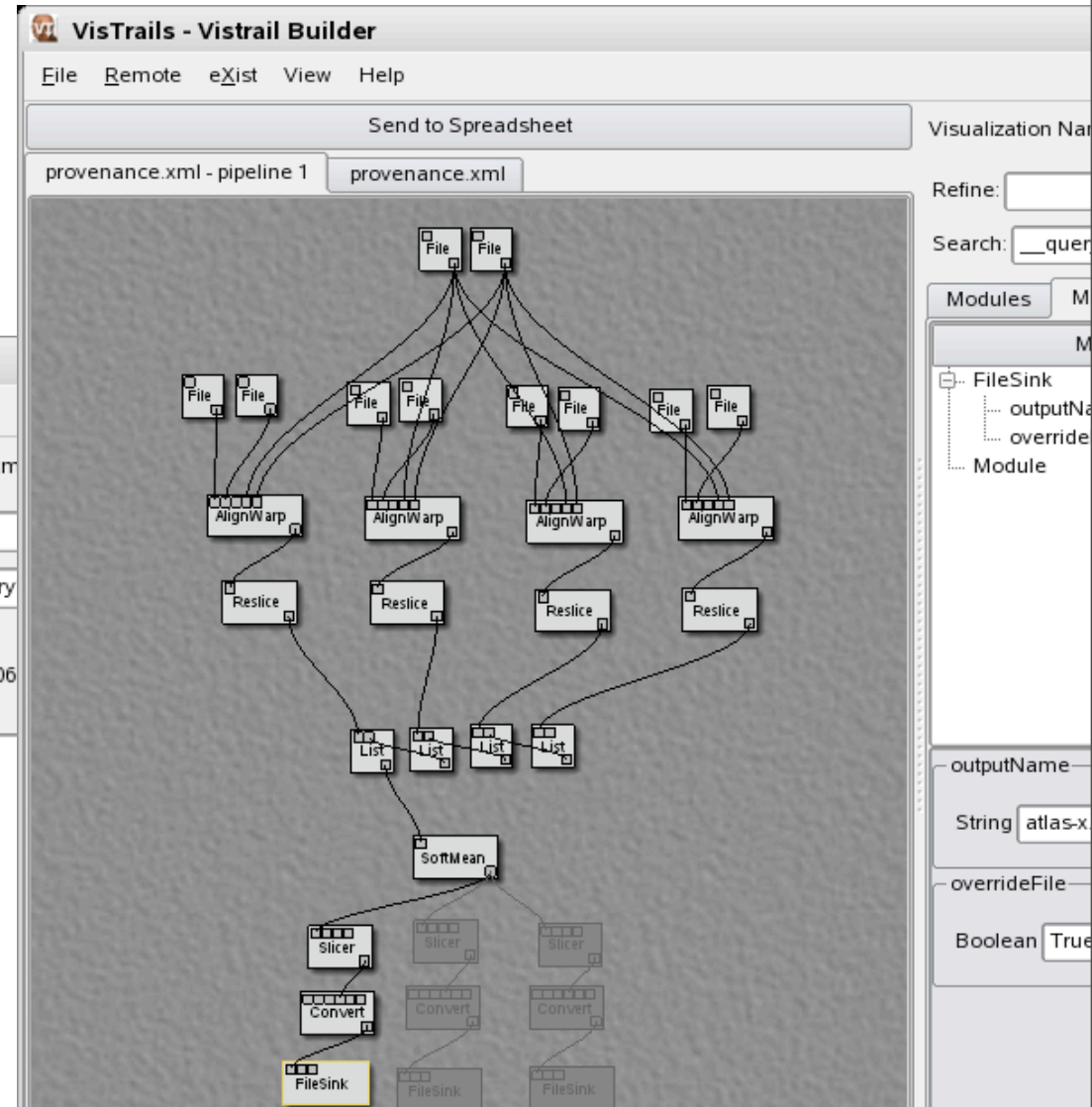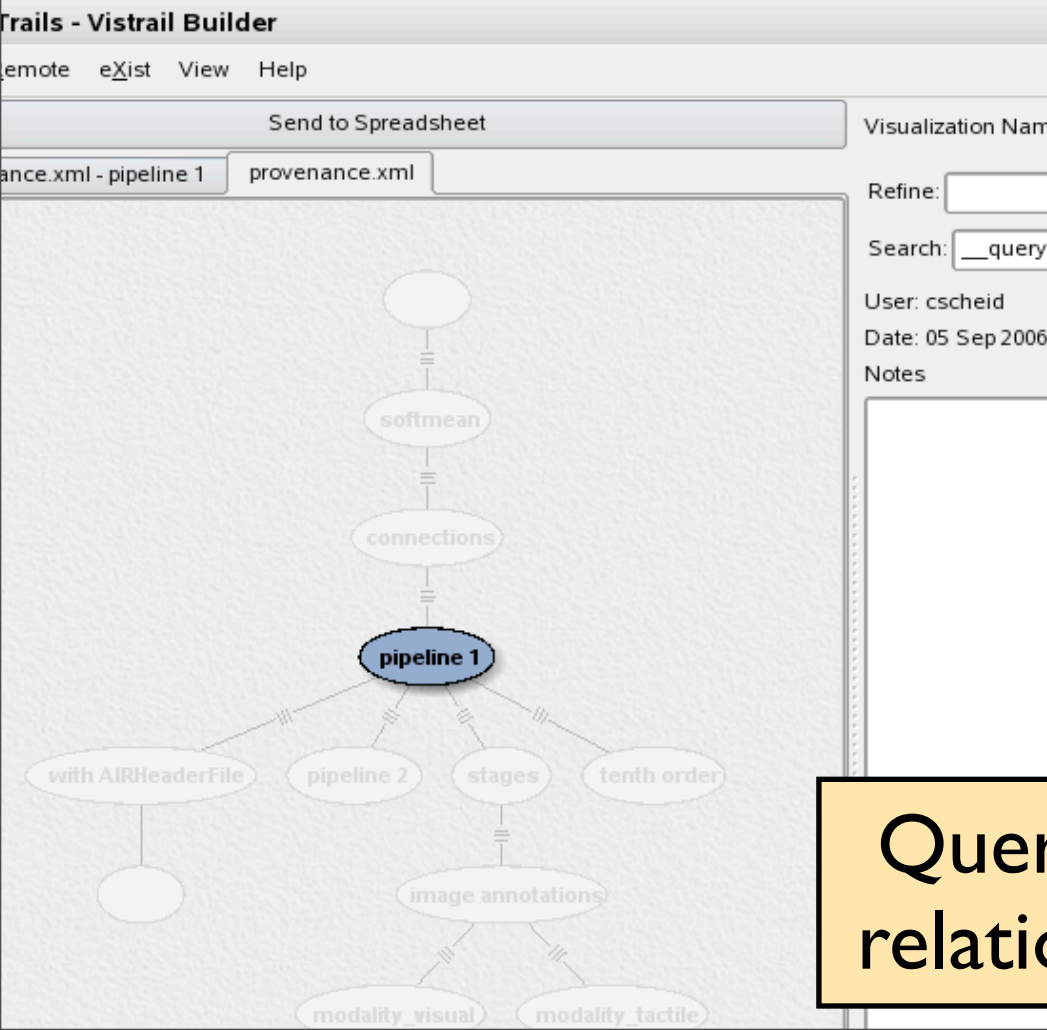- 🔷 ts_start: DATETIME
- 🔷 ts_end: DATETIME

indices

Provenance Trace

- Version tree induces a relation (version, modules)

- We extended the provenance with new relations that capture executions of the workflows and modules
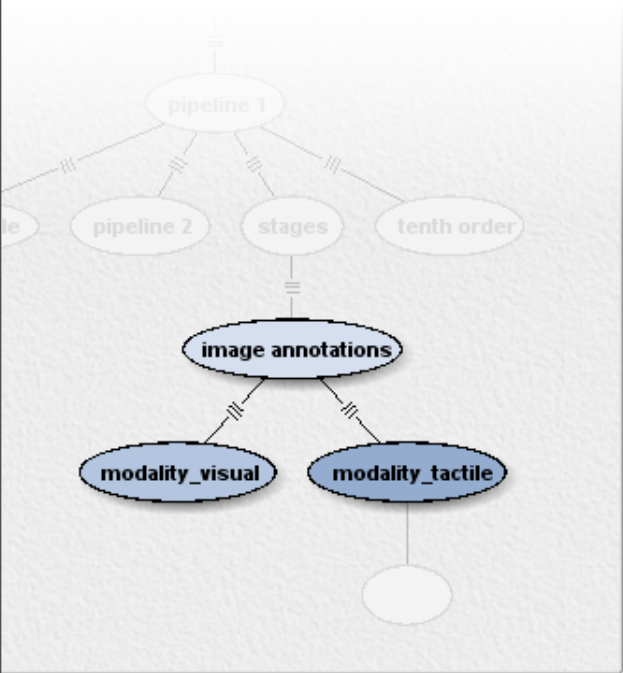
- How do we query these?

Provenance Trace

```
wf: upstream(x) union x where
    x.name = FileSink and
    x.parameter('name') = 'atlas-x.gif'
    and executed(x)
```


Query language provides a unified relational view for provenance data

Provenance Trace

```
wf: upstream(x) union x
where x.module = AlignWarp
and y in inputs(x)
and y.annotation('center') =
          'UChicago'
```


Query language provides a unified relational view for provenance data

The data logging problem

- Users really want not only execution statisticts, but also the (intermediate) results

- Sufficiently general modules require (at least) kernel-level control of file creation

  - Undecidable otherwise!

- Our solution: provide an API modules can use at runtime to store metadata for further querying

Acknowledgments

- Workshop organizers

- VisTrails team