OptIPortal Collaborations

Requirements and Progress Examples Pt 2

Christoph Willing
QCIF/UQueensland

Bernard Meade
UMelbourne

Continuing today ...

1. Collaboration:
   - what is required?
   - visualcasting
2. Issues arising:
   - pixel count
   - shared control
3. Where to now

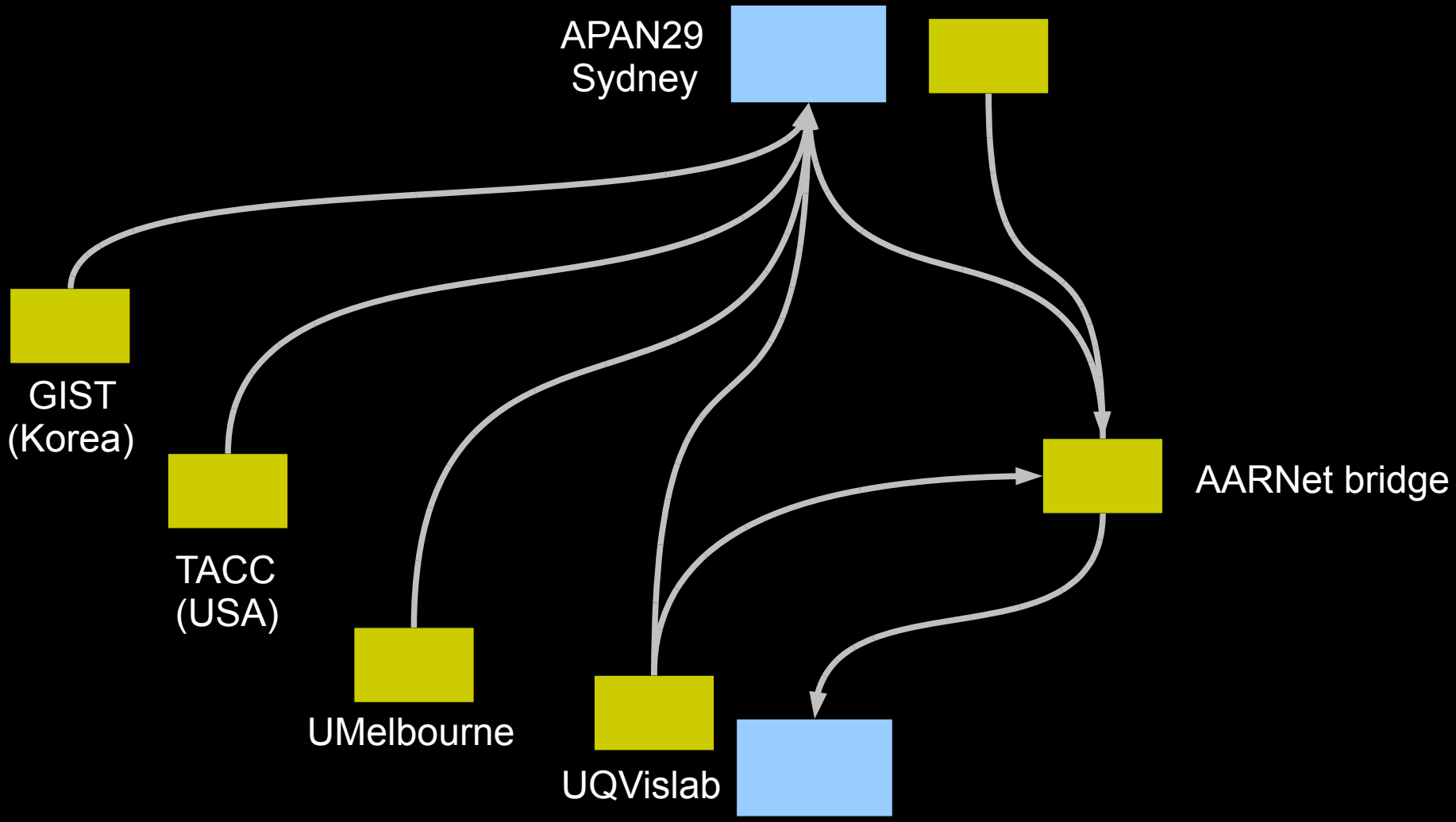
Collaboration

Requirements?

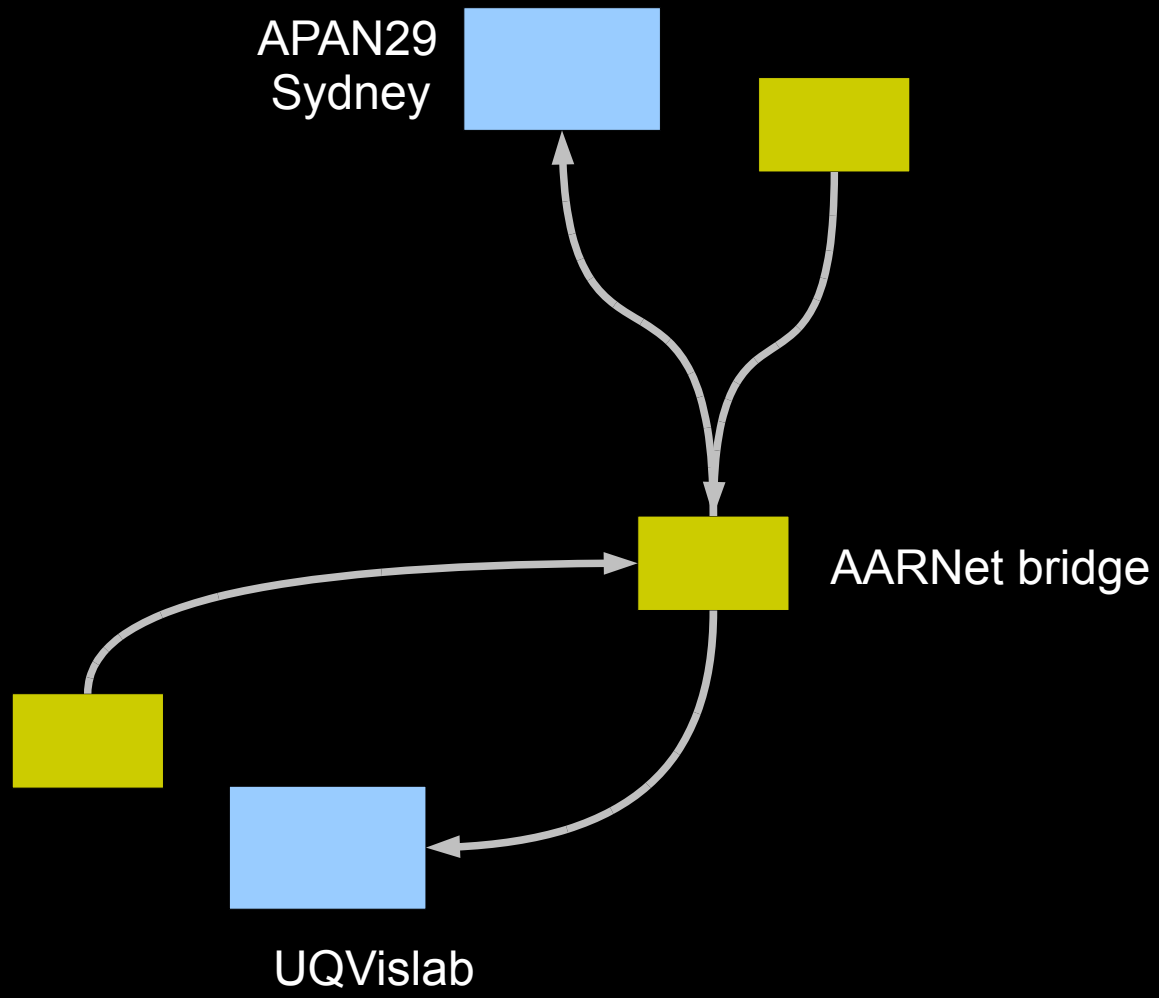
Previously (APAN29):

ability to simultaneously view dynamic content at different display sites

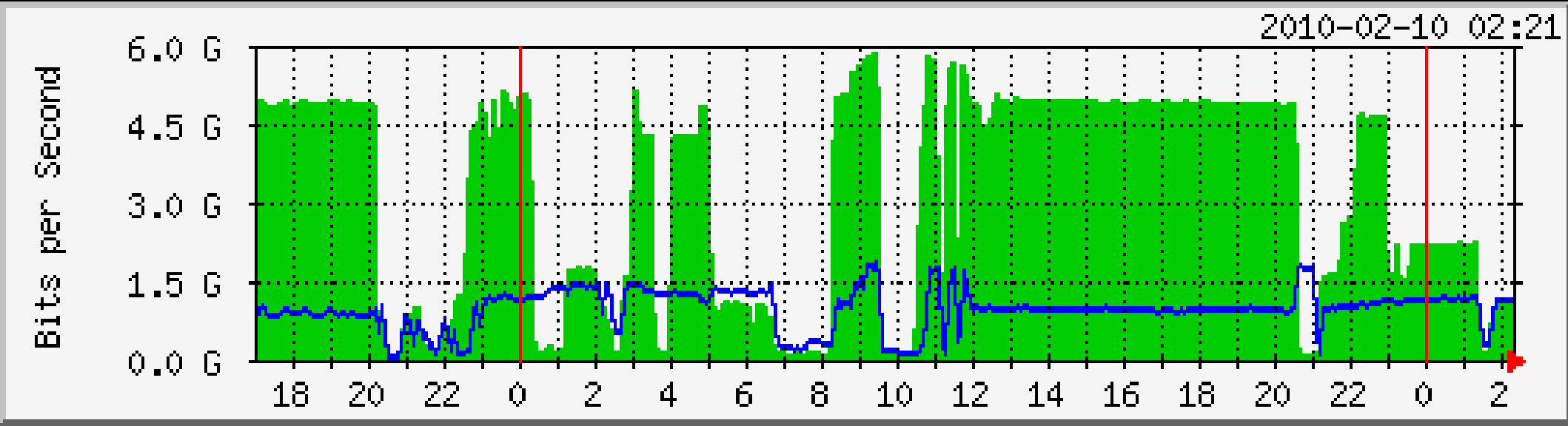
APAN29 – multiple input sites



APAN29 – visualcasting



APAN29 - traffic



Collaboration

Recently?

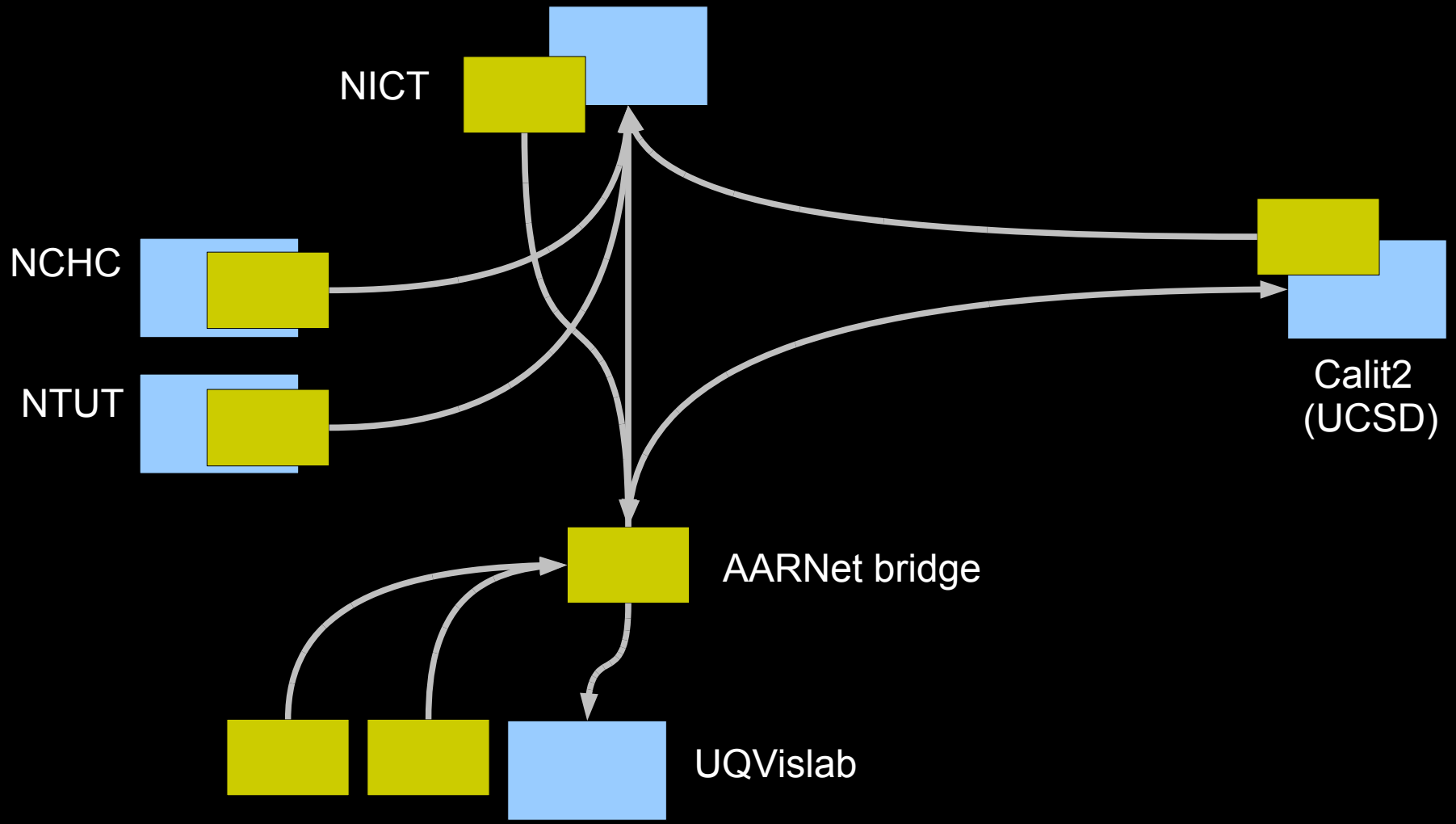
Nathan Faggian's MRI example:

- ability to control a specific application

Osaka “Knowledge City”:

- audio + multidirectional streams via bridges

Osaka “Knowledge City”



Collaboration

Requirements?

Previously:

ability to simultaneously view dynamic content at different display sites

Now:

ability to interact with remote display sites

and simultaneously control applications

Issues arising ...

Faggian's MRI app addressed 2 main issues:

1. total image size > screen size
2. need to control multiple instances
3. need to share control

Pixel Count

Application needs to write to a SAGE buffer
- not always easy

OpenGL wrapper
- limited to size of workstation screen

Pixel Count

Solution: Xvfb

however:

- slow for large sizes (still hw accelerated?)
- how many on single machine?
- GLViewport size limit

i.e. not scalable

- also: no keyboard/mouse access in Xvfb

Control

Need:
- control of single or multiple Xvfb instances
- by single or multiple local/remote sites

News flash!
- these are not requirements exclusive to Xvfb
- these are general collaboration requirements

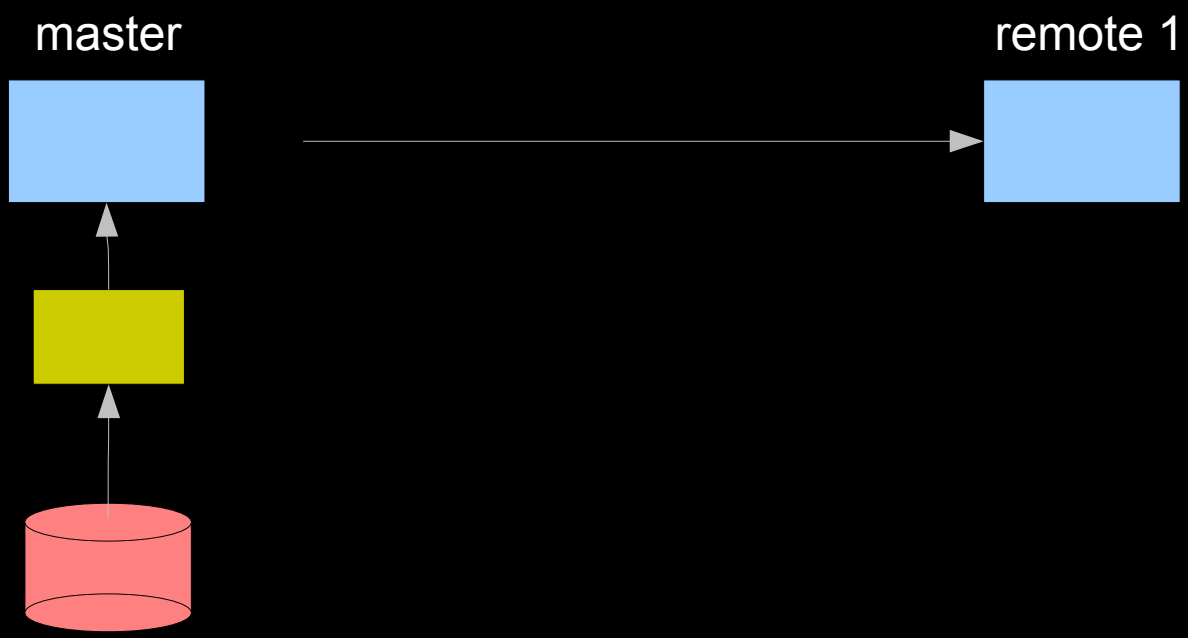
Collaborative Vis Styles

Viewing and interaction between multiple sites

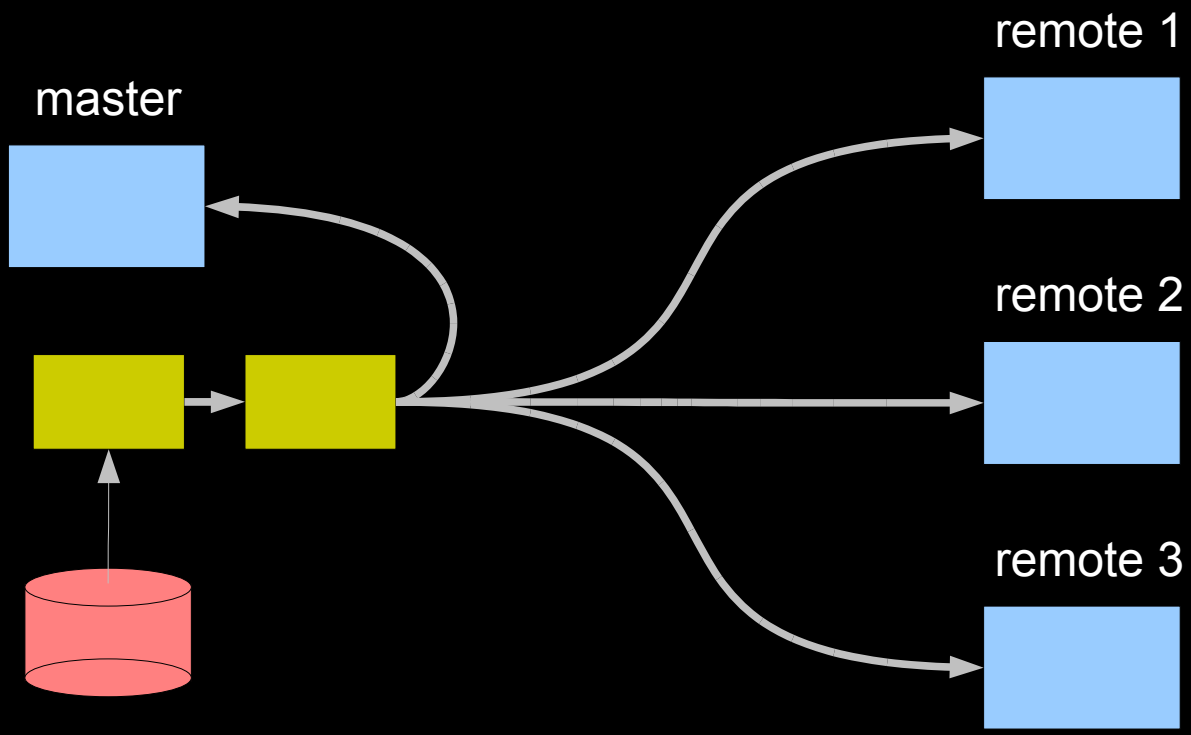
Processing, rendering at:

1. master site
2. multiple sites

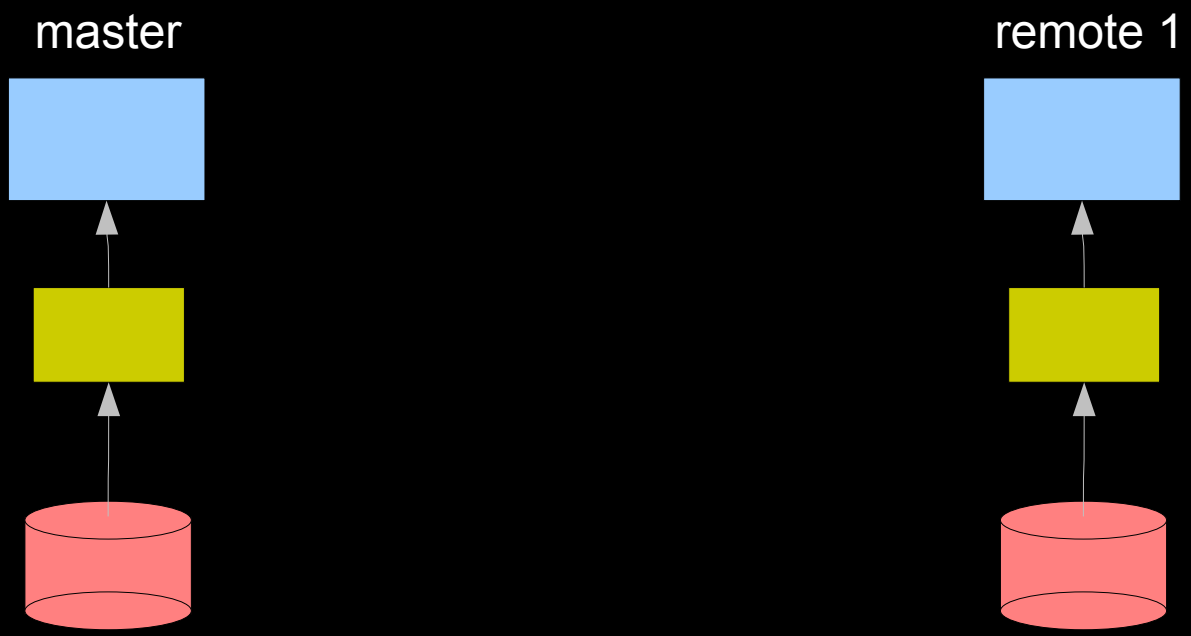
Render at master site



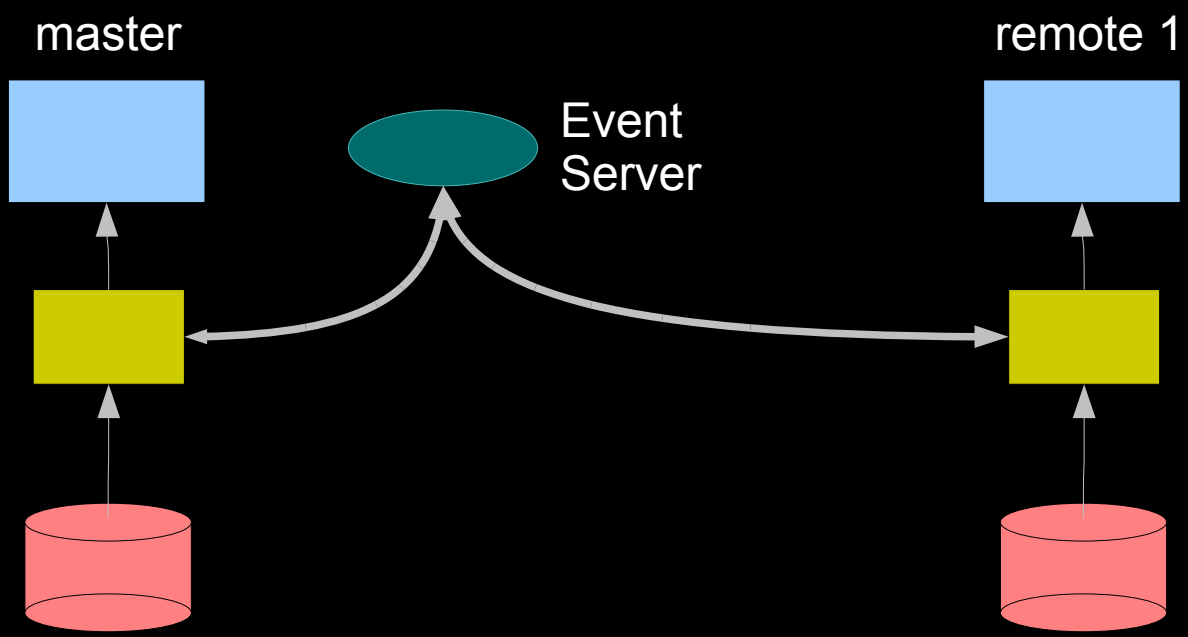
Visualcasting



Render at all sites



Render at all sites



Where to now

- control of multiple app instances by multiple sites
- increased pixel count

Control

generalised event server prototype

- python based
- Enthought Tool Suite scripting via mlab
- *mayavi2* gui frontend to VTK vis toolkit

Pixel count

- Xvfb OK within limitations, esp. GLViewport size:

```
optiputer:~ $ glxinfo -l | grep DIM
    GL_MAX_VIEWPORT_DIMS = 8192, 8192
```

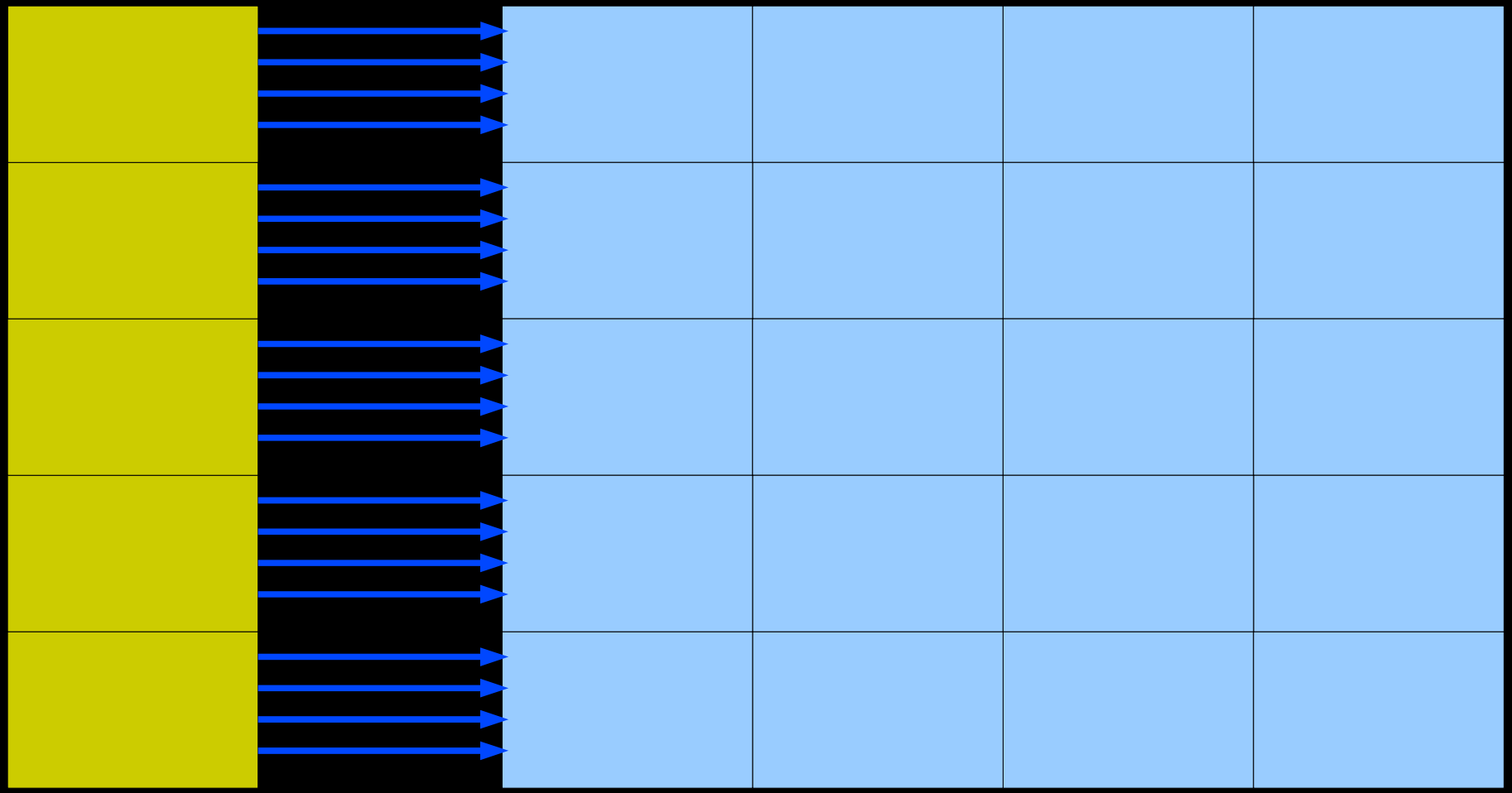
Pixel count

Solution?

Multiple instances
- rendering directly to display
- 1 instance per tile

Minimal MPI
- launch all instances with mpiexec
- each instance uses rank to render partial output
- occasional MPI_Barrier() for sync

20 instances 1920x1200 = 7680x6000 @ QCIF OptIPuter, UQVislab



atlantis

atlantis

7680x6000 @ 90fps

add glReadPixels(): 30fps

sageInf.swapBuffer(): 20fps

roller

Conclusions

- have elements for OptIPortal collaboration
  - interaction
  - application control
- mainly as prototypes (want real applications)
- simple MPI enables ultimate resolution in SAGE
  i.e. equivalent to CGLX
  but retaining SAGE's network-centric base