

Massive Remote Batch Visualizer

NAG DCSE Technical Meeting

23-24 September 2009, Oxford

George Leaver , Martin Turner
Research Computing Services
University of Manchester

Contents

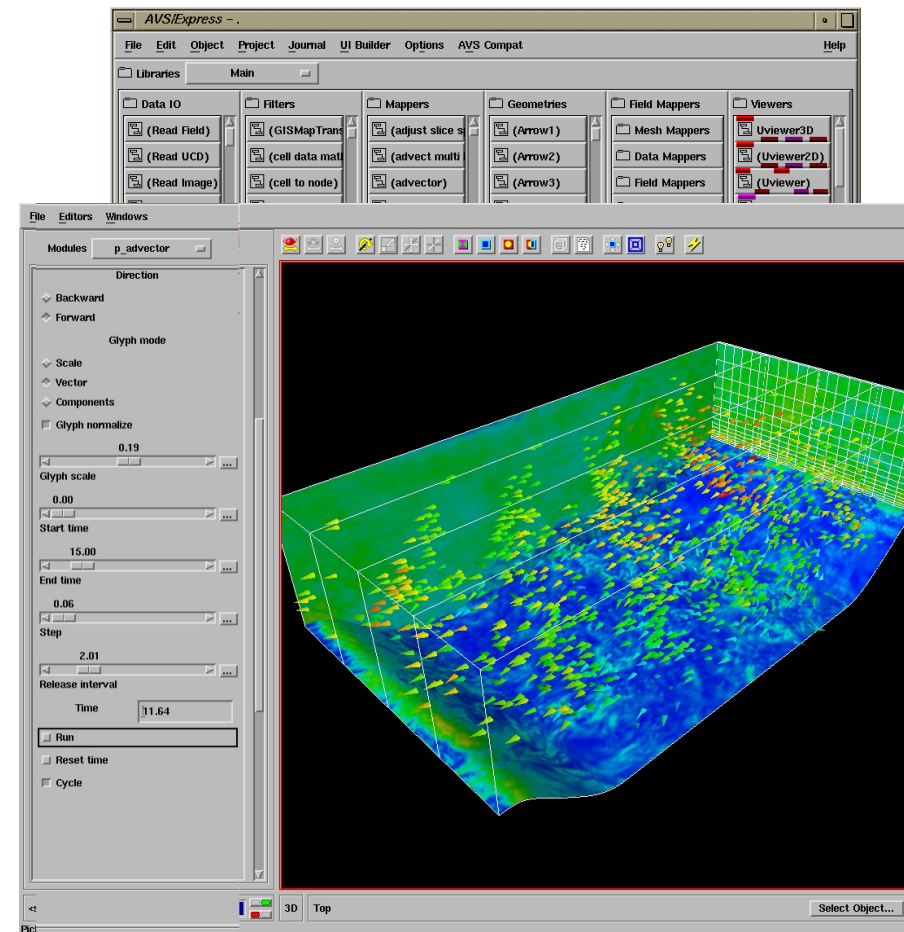
- Overview
- AVS/Express
- Porting / development
- Future work

Overview

- Port AVS/Express to HECToR
- Develop viz apps for specific user groups
 - Materials Science
 - Life Sciences
- Batch vs Interactive
- 10 months DCSE + 2 months NAG optimization
 - May 2009 – Feb 2010
 - MS1: Port / build components
 - MS2: Viz Networks
 - MS3: Parallel IO Module
 - MS4: Optimization
 - MS5: User base testing / results

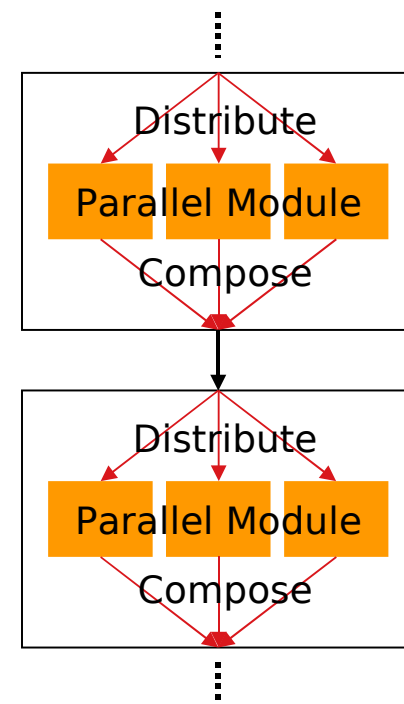
AVS/Express

- Rapid visualization pipeline construction
 - Visual programming
 - Easy to build GUI system
- Extensive visualization tools
- User extensible (C/C++)
- Portability
 - V code description language



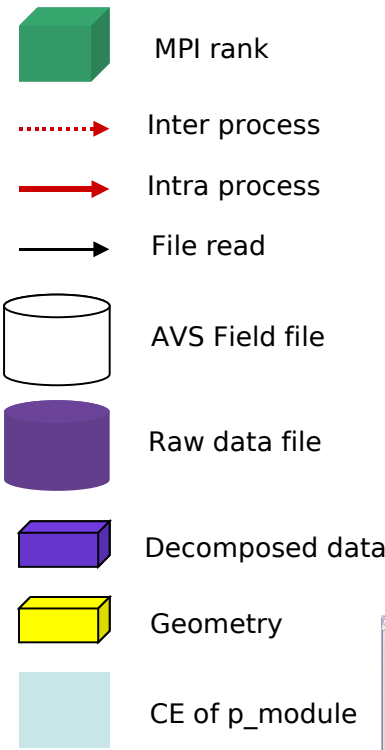
AVS/Express Editions

- Standard
 - Serial compute & rendering
 - Parallelize at user-module level (hidden from object manager)
- Multipipe Edition: parallel rendering
 - Adds new renderer to S/W, OpenGL, ...
- Parallel Edition: parallel modules
 - Adds API, framework, **p_** modules
- Distributed Data Renderer: PE + distributed rendering
- MPI (+ pthreads in renderer)
 - Clusters, interactive



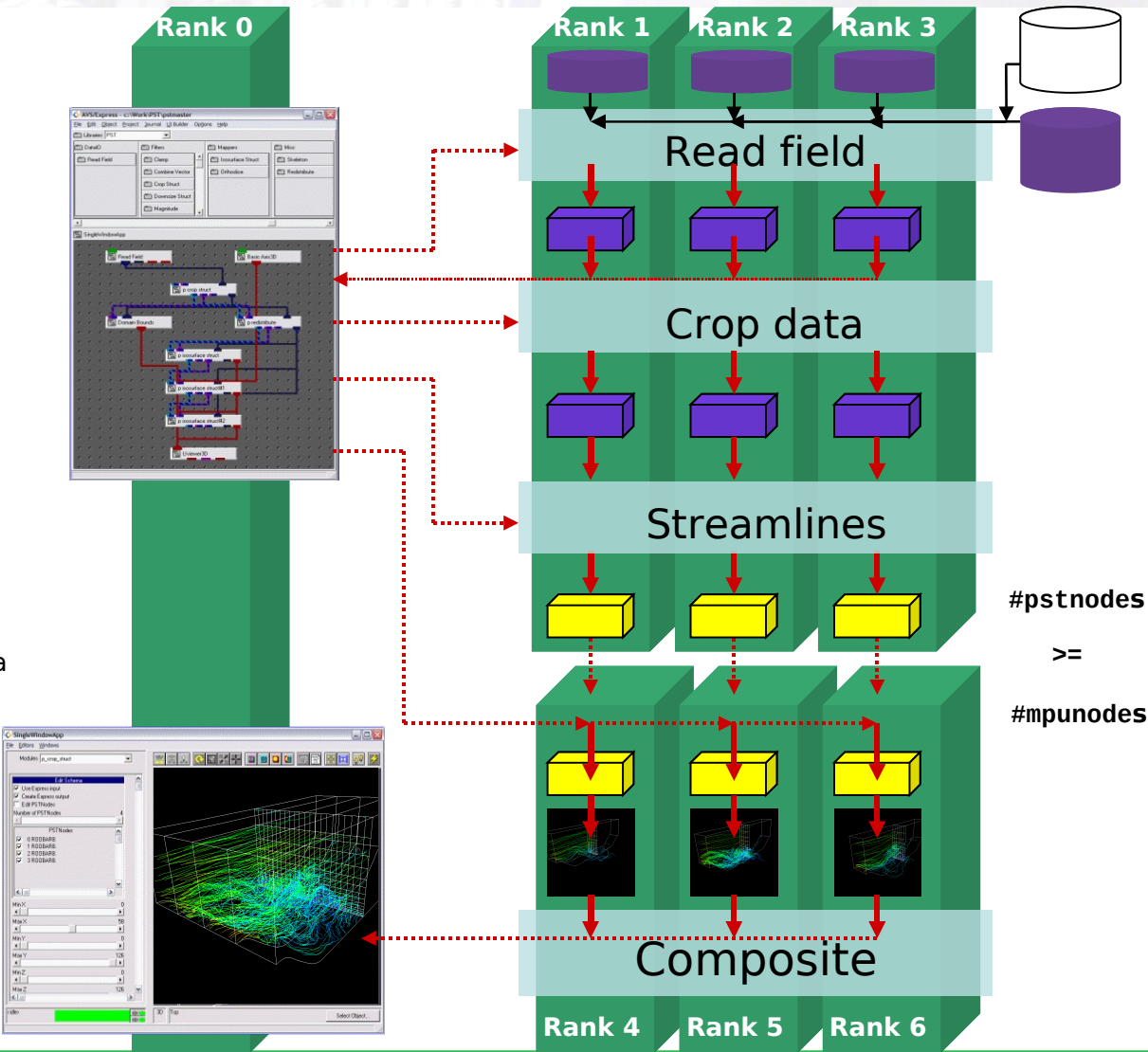
DDR Architecture

Before HECToR

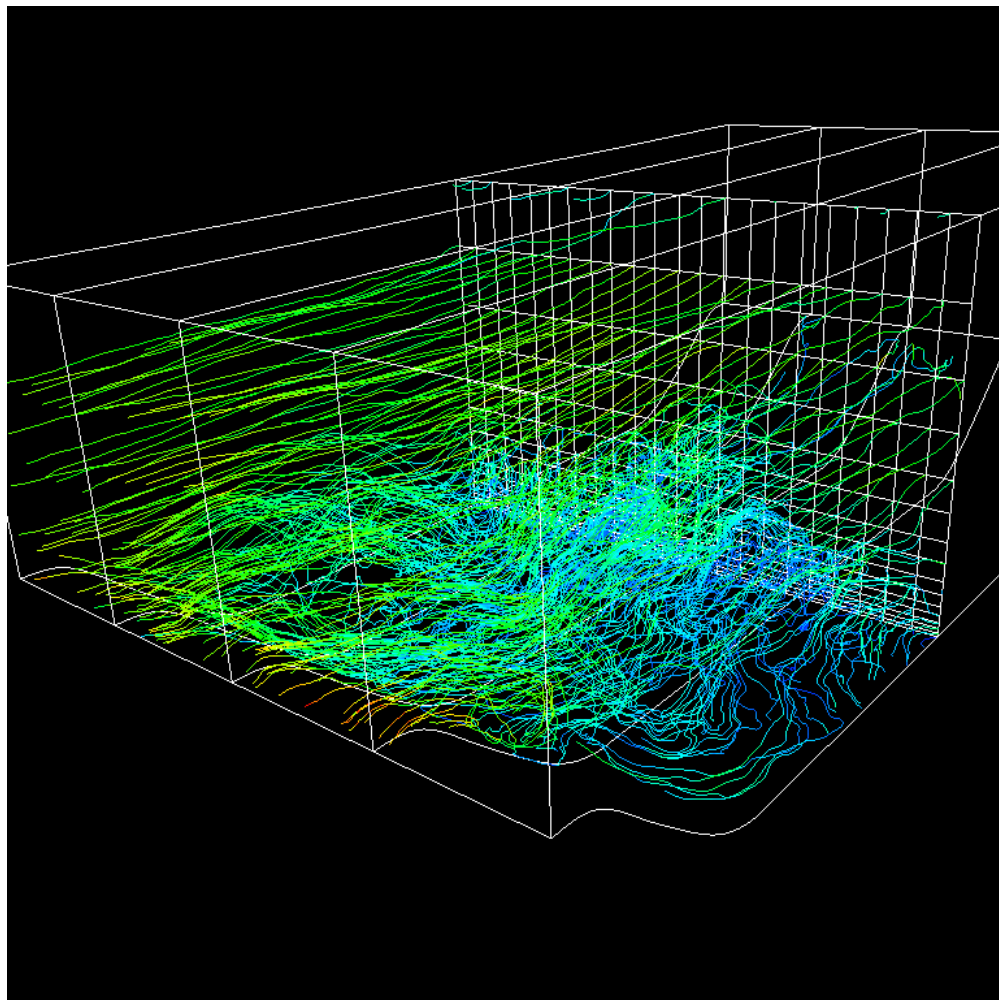
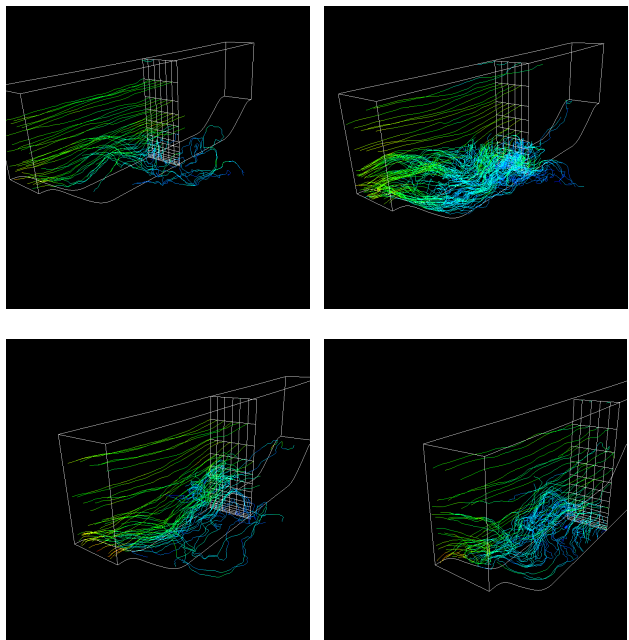


Rank 0

- Compute vs login



DDR Streamlines



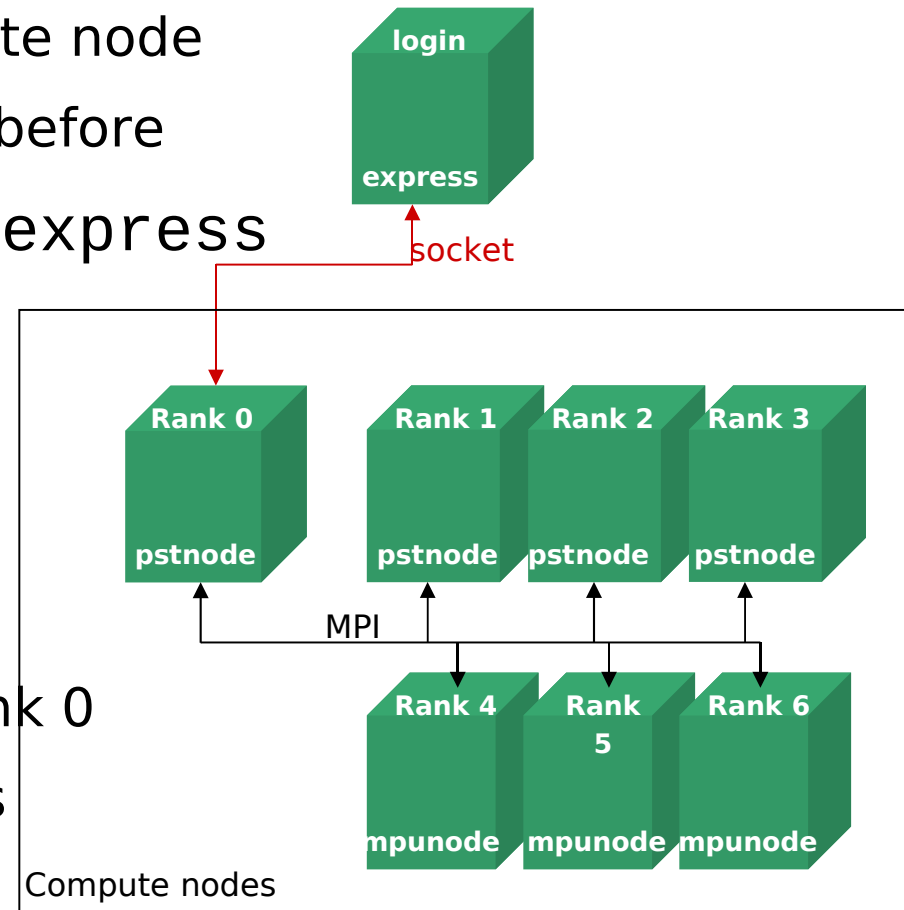
Data Courtesy Lionel Temmerman and Michael Leschziner Imperial College London Department of Aeronautics

Porting to HECToR

- Express on login node
 - No MPI
- Express on compute node
 - No X server
 - `express -offscreen` badly implemented
- Remove MPI
 - Two versions of code to maintain
- Replace Vendor MPI
 - Forward MPI from login to compute

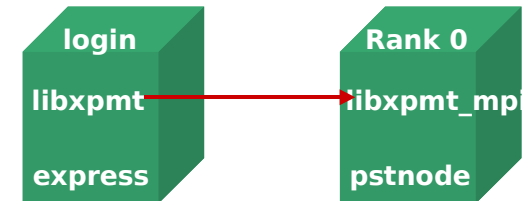
MPI Forwarding

- express was always rank 0
 - Pose as express on compute node
 - `pstnode(>0)`, `mpunode` as before
- `pstnode(0)` connects to express
 - Express submits MPI job
 - Pass login ip via env
 - Express listens
- Usual Express/DDR init
 - Processes register with rank 0
 - Comm initiated by express



XPMT Libraries

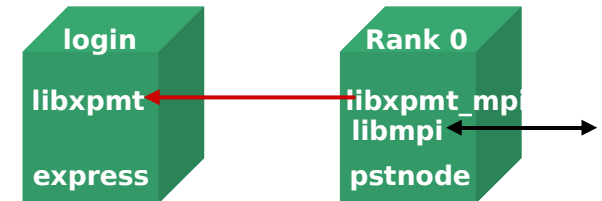
- XPMT C libs developed
 - express links libxpmt.so
 - pstnode links libxpmt_mpi.a + libmpi.a
 - mpunode links libmpi.a
- libxpmt
 - Subset of MPI functionality
 - xpmt_mpi.h not mpi.h
 - All MPI objects represented as indices
 - `MPI_abc( ... )` packs args, data, indices → rank 0
 - Awaits return values



XPMT Libraries

■ libxpmt_mpi

- Message recv / function dispatch loop
- xMPI_abc(void) Receives packed xpmt args, data
- Indices lookup real MPI objects
 - *Comm_table, Request_table, Op_table, ...*
- Call real MPI function MPI_abc(...)
- Store new MPI objects → new indices
- Send results, data, indices, err flags to libxpmt



■ MPI_Status objs non-standard

- Persist in libxpmt_mpi: pstnode(rank0)
- Require explicit deletion from libxpmt side: express

Compositing

- Last 'to-do' to complete port
- Currently AVS use HP's Paracomp
 - Parallel Pipeline model (Lee, et al, IEEE TVCG 96)
 - No MPI support
 - Hostnames of compositing nodes reqd
 - 4 pthreads per node
- Compositing core extracted
 - Supply rgba images (+depth) for blending
 - Requires MPI image transport scheme
 - *2-3 Swap. Yu, Wang, Ma, SC08*
 - Currently file-based direct-send compositing

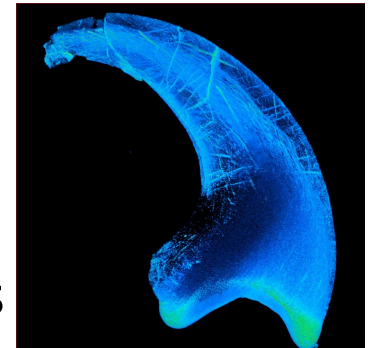
User Cases

■ Materials Science

- Dr Paul Mummery
- Henry Moseley X-Ray Imaging Facility (EPSRC, NWDA, UofM)
- CT scanning equipment (2k x 2k x 16bit x n slices)
- JEEP Beamline I12 (~4k x 2k x n slices)
- 50-100Gb volumes
- Volume render (rgba=4xRaw data), isosurface ($\sim 2 \times 10^9$ triangles)

■ Life Sciences

- Dr Bill Sellers
- High Res CT dinosaur/footprint fossil scans
- 40Gb claw

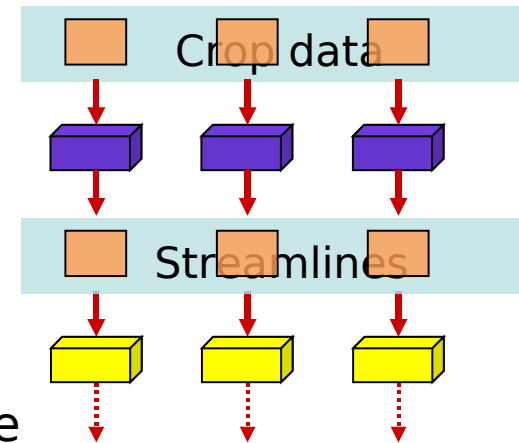


Future Work

- MS1: Build / port
 - Complete compositor MPI transport
- MS2: Viz Network
 - Common viz networks complete
 - Month 6 about to begin (integration of apps with desktop AVS)
 - Scalability tests
- MS3: Parallel IO module
 - Month 7 File decomposition tool + file striping tests
- MS4: Optimization
 - Month 8 (next slide)
 - Month 9 (compositing scheme)

Optimization

- p_module C.E. code
 - Currently serial within MPI process
 - Ideal for mixed-mode OpenMP/MPI
 - Isosurface, orthoslice
- p_volume_render
 - pstnode passes data straight to mpunode
 - mpunode converts intensity data to rgba
 - MesaGL loads rgba data as texture
 - SysV shmem technique
- Express MPI messaging
 - Many point-to-point



Thanks to James Perrin, Paul Mummery

george.leaver@manchester.ac.uk

<http://www.manchester.ac.uk/researchcomputing>