NPACI
Programming Tools and Environments

- **Interdisciplinary problems that require coupling of multiple application programs and data sources**
- **Software support for problems that require multiple time and space scales**
- **Applications use combination of sensor and simulation data**
  - Distributed data sources
  - Preprocessing should be near data -- distributed compute resources

Projects

- **Runtime support libraries**
  - Adaptive multiresolution data structures -- throughout storage hierarchy
  - Support for parallelization and combined associative query and processing
  - Several projects -- Baden, Scott, Brown, Saltz
- **Tools for coupling parallel libraries and programming environments (MetaChaos "type of parallel I/O")**
  - Knows about data distribution conventions
  - Layered on top of metacomputing/grid software (Globus, Legion)
- **Compiler development**
  - Titanium (support for explicitly parallel codes), support for compiling out-of-core

Projects

- **Client server agent for use of library based software (Netsolve)**
  - Provide computational libraries over internet
  - Linear algebra, neural simulations
- **Applications collaborations**
  - Applications projects defined so that each uses multiple coupled runtime support libraries. Applications projects use information from one or more data repositories. Composite applications drive compiler research

Titanium

- Language and system for high-performance parallel scientific computing.
  - Uses Java as base adds
    - immutable classes,multi-dimensional arrays, an explicitly parallel SPMD model of computation with a global address space, and zone-based memory management
    - current driving application: a three-dimensional adaptive mesh refinement parallel Poisson solver by Phillip Colella and Luigi Semenzator of NERSC/LBNL
    - a few thousand lines of code; handles multiple levels of refinement and both concave and convex boundaries

Titanium Accomplishments

- Fully functional 3-D AMR Poisson solver
- Improved usability, broadened user base
  - **Tutorial given at the NPACI Parallel Computing Institute in August '97; demonstration of Titanium was given at SC'97.**
  - **Spring semester 1998, Titanium was used in the graduate parallel computing course at UC Berkeley**
  - **Developed a benchmarking suite with over a dozen different codes, which will be used in regression testing new versions of the compiler and in tutorials**
  - **Produced online documentation to explain how to use the compiler**

Titanium Accomplishments

- Performance of sequential Titanium code is quite competitive with C/C++ or Fortran.
- Benchmarking studies on small application kernels indicate that sequential Titanium code written with the Titanium multidimensional array construct is sometimes faster than C/C++ or Fortran, and is never more than a factor of two slower
- Prototype debugger for sequential Titanium code

Plans

- **Expand documentation**
- **Develop new applications**
  - Additional benchmark applications
- **New supported platforms**
  - Berkeley NOW, T3E
- **Improve compiler**
  - Generate C rather than C++ code
  - Extend synchronization model to include mutual exclusion
- **Compare performance and expressiveness with other systems**
  - Implement 3-D AMR Poisson solver in KeLP

Linear Algebra

- **J Demmell -** Develop and release four numerical packages for general use; help integrate them into several NPACI applications
  - **Packages:**
    - **Distributed SuperLU, a scalable direct solver for sparse nonsymmetric linear systems**
    - **Prometheus, a parallel multigrid solver for FE problems arising in solid mechanics and elsewhere**
    - **PBody, a parallel N-body code that incorporates many useful variations (Barnes-Hut, FMM, etc.)**
    - **RelEigs, a new fully parallel symmetric eigensolver.**

Linear Algebra

- NPACI applications
  - Earthquake simulation (with G. Fenves),
  - E-Beam lithography (with A. Neureuther),
  - Finite element modelling (L. Demkowicz).

Accomplishments

- Modified ScaLAPACK kernel sustained 684 Gflop on + 9000 processors of ASCI-Red: overall application sustained 605 Gflops (Gordon Bell finalist).
  - **Material science application -- 40K by 40K double complex**
- Distributed SuperLU was accepted to Supercomputing 98: "Making sparse Gaussian Elimination scalable by static pivoting," by J. Demmel and X. Li.
  - **Solved set of large sparse matrices from applications -- 8 Gflop on 512 Cray T3E processors**

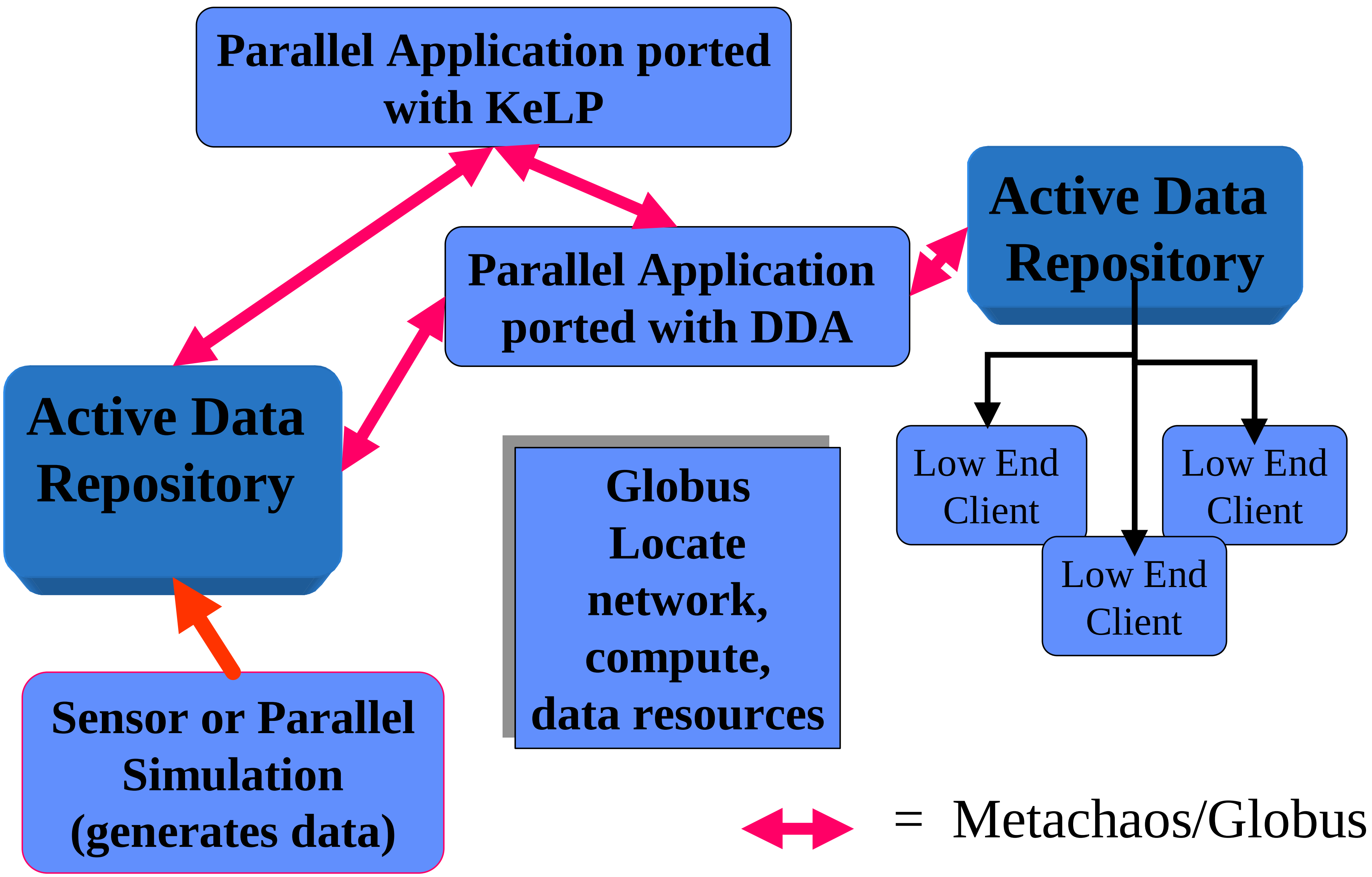
Accomplishments

- Parallel multigrid solver - Prometheus
  - **User specifies distributed assembled matrix and x,y,z coordinates**
  - **Generates coarse meshes automatically**
  - **Paper on the the maximal-independent-set-based mesh coasening algorithm -- published in the 5th Copper Mountain Conference on Iterative Methods -- won the best student paper prize**
  - **runs scalably on the NOW, a Cray T3E and ASCI Blue**
  - **gets about 60% scaled parallel efficiency on the Cray on a difficult finite element test problem with two very different materials side-by-side**

Milestones

- Pbody: Release library, integrate into NPACI Partner code
- Prometheus: design library interface
- Distributed SuperLU: Incorporate better prepivoting for stability, Better parallelization of triangular solve, PETSc interface, help integrate into NPACI partner code
- New release of Distributed SuperLU (Release v 1.0)

Application Scenarios



KeLP

- Scott Baden, Larry Carter
- Provide high level run-time support for block structured applications with irregular data decompositions
- Formalisms for handing structured but irregular data transfers
- Runtime communication analysis

Accomplishments

- **Multiblock, multicomponent U Texas (ESS) mortar space code being ported using KeLP**
- **Collaboration with Titanium project -- performance comparisons**
- **Collaboration with Maryland to develop and implement KeLP/MetaChaos interface**
- **KeLP hardened to carry out runtime error checking**

Accomplishments

- KeLP version 1.0 has been installed on the Cray T3E at SDSC and on the IBM SPs at U.Texas and SDSC
- KeLP data model separated from the control model.
    - **permits the user to invoke KeLP from their application. Earlier releases versions of KeLP required that KeLP execute the "main" program**
- porting various multi-level and multiblock benchmarks -- will be used in performance studies with the Titanium project at UC Berkeley.

Plans

- Extend KeLP to handle multiple levels of parallelism -- "KeLP^n"; develop prototype
  - **tolerate latency on SMP-based multicomputers**
  - **hierarchical tiling**
- Complete KeLP hardening on IBM SP2 and Cray T3E.

Plans

- Performance study comparing KeLP and titanium on Multiblock codes
- Demonstrate interoperability between KeLP, ADR using MetaChaos
- Develop KeLP based load balancer for ESS application
- Begin work on neuroscience application (brain bending and blending - collaboration with Miller)

Pfortran and PC Compiler

- Ridgeway Scott
- P languages -- explicit parallelism, both global and local name spaces
  - **Compilers exist for Fortran and C based P languages**
- Augment P languages to provide support for unstructured computations

Accomplishments

- **Project is essentially a new start**
- **Last year:**
  - Recoded runtime support to handle arbitrary numbers of processors (rather than meshes and hypercubes)
  - Produced documentation, bug fixing
  - Unification of PC and Pfortran test suites

Plans

- **Implement ASTRO-3D, UHBD using P languages**
- *ASTRO-3D is being used by the ASCI project at U. Chicago, and UHBD is being developed jointly in NPACI by UCSD and U. Houston (molecular science)*
- *ASTRO-3D*
  - Phase 1: Solve laplace’s equation on logically rectangular 3-D mesh
  - Phase 2: Particles do brownian walk through space. Irregular access of shared data structure

Plans

- **Install compilers on NPACI machines**
- **Implement simple web based installation methods for propagating P language software**
- **Integrate P language code with libraries (KeLP, DDA)**
  - Direct integration and use of MetaChaos

Plans

- **Demonstrate MetaChaos based interoperability between P languages and KeLP**
  - Use ASTRO-3D and UHBD

Netsolve

- **Jack Dongarra**
- Develop network-enabled Problem Solving Environments (PSEs) for NPACI application areas that build on NetSolve
- Software environment for networked computing that allows users to call procedures that reside on remote compute servers

Accomplishments

- Mcell performs three dimensional Monte Carlo simulations of cellular microphysiology.
  - **MCell users had manually started MCell jobs on different machines, manually distributed the input files and gathered the output files.**
  - **Such manual operation is impractical for runner hundreds of computations in parallel and provides no mechanisms for fault tolerance or load balancing.**
- NetSolve approach: user passes a list of all MCell simulations to be performed to the driver, which submits the requests to NetSolve
  - **NetSolve runs the jobs in parallel on available computational servers -- assures load balancing and fault tolerance**

Accomplishments

- NetSolve has been installed and tested on NPACI machines.
  - **Netsolve supports ARPACK, FitPack, ItPack, MinPack, FFTPACK, LAPACK, QMR, and ScaLAPACK (linear algebra, optimization, FFTs)**

Plans

- Install NetSolve, support MCell on the SDSC Cray T3E
- Modify MCell so that it calls NetSolve internally (increase efficiency by distributing only compute intensive portions of Mcell -- reduce data transfer overheads)
- Integrate NetSolve with Globus and Legion
- Identify new NPACI applications that can be supported by NetSolve

MetaChaos

- **Joel Saltz**
- Meta-Chaos supports data exchange between runtime libraries
  - **Libraries have different conventions for supporting distributed data structures**
  - **Meta-Chaos tracks distribution conventions and supports parallel data transfers**
    - **Can couple libraries in a single program or can link libraries in multiple programs**
    - **Will employ Globus/Nexus for low level data transfer and to launch programs**

Accomplishments

- Hardened MetaChaos and produced documentation
- Developed and implemented MetaChaos, KeLP and ADR APIs
- Used MetaChaos to couple Maryland ESS application with ADR
- MetaChaos implemented using Globus/Nexus (still in testing)

Plans

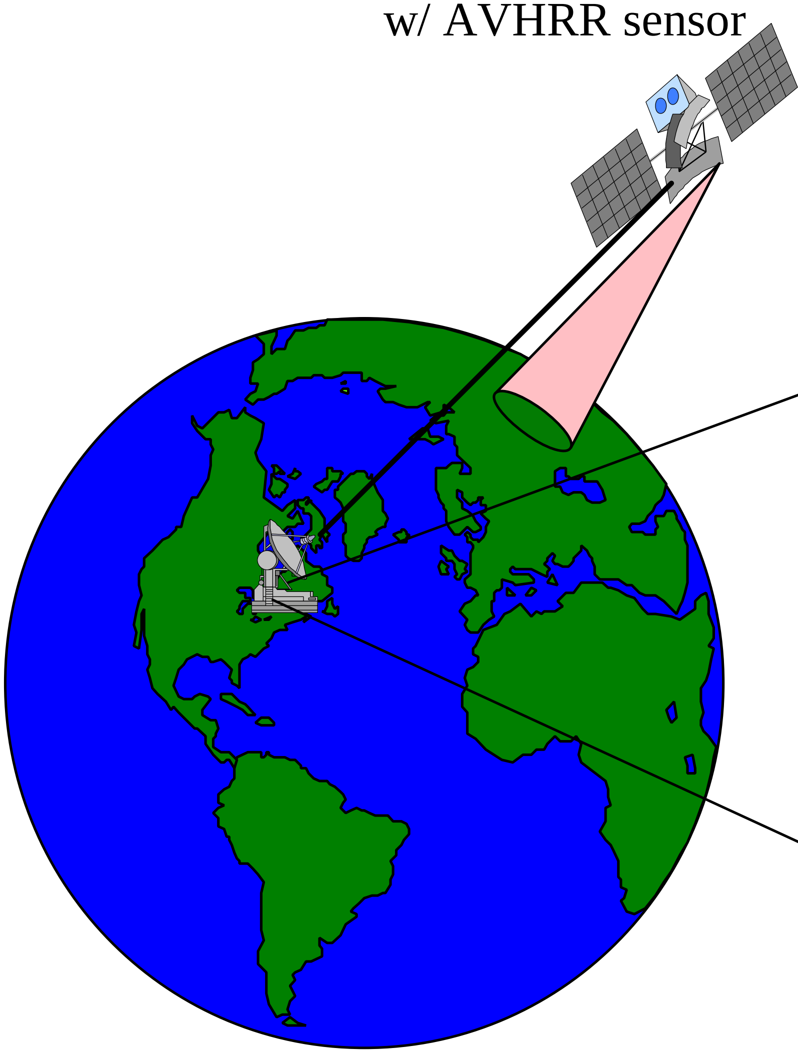
- Implement coupled application codes in U. Texas bay and estuary simulation project
  - **Flow simulator (PADCIRC) coupled to a chemical transport simulator (CE-QUAL-ICM) via a projection code (UTPROJ)**
  - **Flow data stored in ADR**
  - **MetaChaos queries retrieve and process flow data**
    - **specify time period, calculate time differences of elevation and interpolated velocity values, and compute average elevation differences and velocity values**
  - **Projection code post-processes velocity values**
  - **MetaChaos to couple projection code to chemical transport simulator**

Plans

- Complete integration of MetaChaos and Globus
- Integrate KeLP, P languages, DDA libraries into MetaChaos framework
  - **Employ MetaChaos to allow different blocks in KeLP ported mortar space code to execute on different platforms**
  - **Define and implement application scenarios that involve use of MetaChaos to couple programs written using KeLP, P languages, DDA libraries, ADR**
- Use of MetaChaos in coupling ADR to land cover classification and vegetation quantification programs
- Release of two versions of increasingly hardened MetaChaos code, generation of tutorial with working examples that demonstrate how MetaChaos can be used to link existing or new parallel libraries

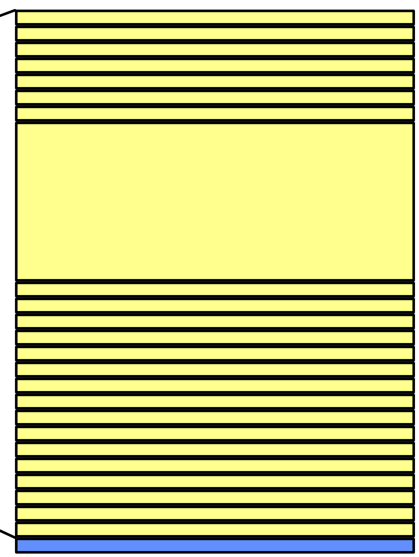
Remotely Sensed Data

NOAA Tiros-N
w/ AVHRR sensor



AVHRR Level 1 Data

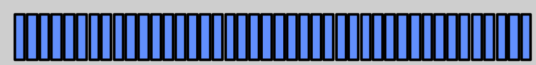
- As the TIROS-N satellite orbits, the *Advanced Very High Resolution Radiometer* (AVHRR) sensor scans perpendicular to the satellite's track.
- At regular intervals along a scan line measurements are gathered to form an *instantaneous field of view* (IFOV).
- Scan lines are aggregated into Level 1 data sets.



A single file of *Global Area Coverage* (GAC) data represents:

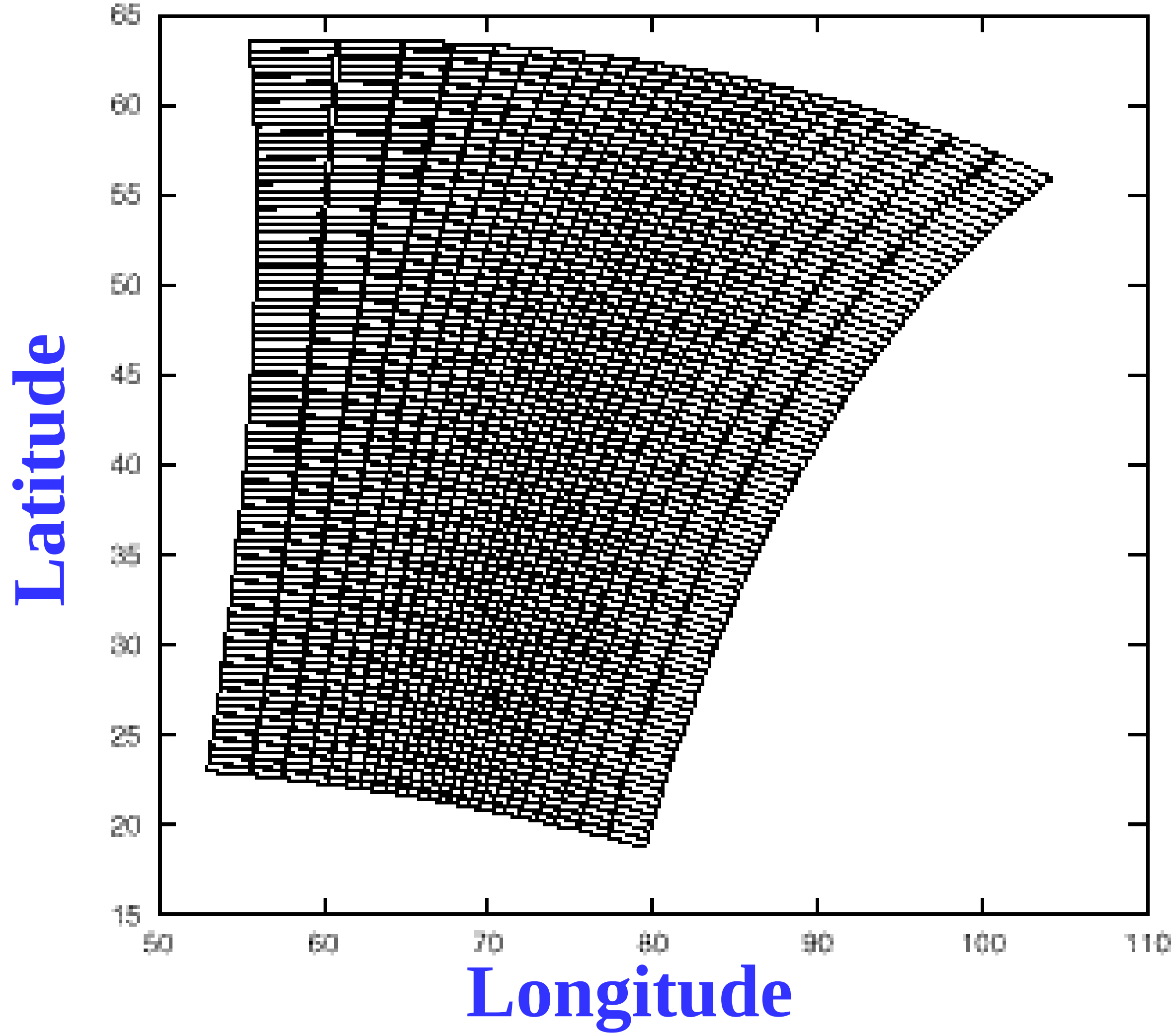
- ~one full earth orbit.
- ~110 minutes.
- ~40 megabytes.
- ~15,000 scan lines.

One scan line is 409 IFOV's



Spatial Irregularity

AVHRR Level 1B NOAA-7 Satellite 16x16 IFOV blocks.



Data Fusion

- **Jim Browne**
  - **Fusion and integrated analysis of tera-scale data deriving from both computation and experiment**
  - **integrate four ongoing threads of research: the HDDA/SDDA infrastructures for parallel implementations of adaptive computations, the Shasta interactive visualization package, the Active Data Repository and the Globus/Legion distributed computation management system**
  - **Merge HDDA/SDDA into DDA infrastructure able to carry out combined block structured/unstructured computations**

Plans

- *This is a new start*
- Driven by collaborations with composite materials (where the data is CT scan data), by reservoir modeling (where the data is seismic data) and by studies of the impact of electromagnetic fields on tissues (MRI data).
  - **Computational modeling of the composite materials research is already built on the SDDA infrastructure.**
  - **HDDA and SDDA merged into single DDA implementation**
  - **Interface DDA to MetaChaos**
  - **Participate in scenarios in which DDA, ADR, KeLP and P languages, coupled via MetaChaos are used to implement application**

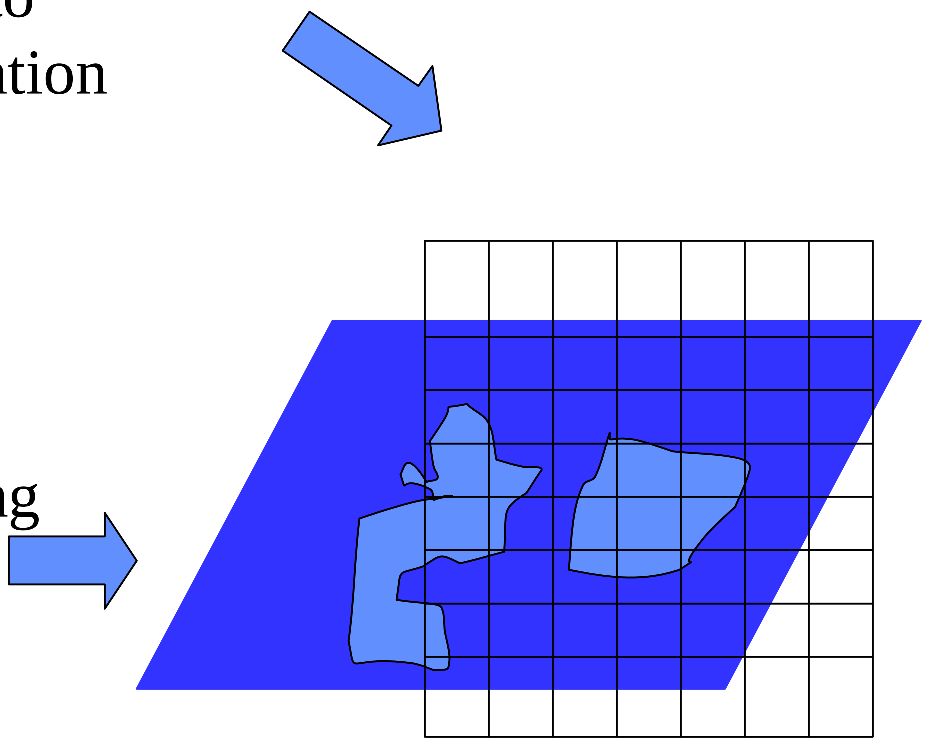
Active Data Repository

- Joel Saltz
- Infrastructure for building parallel database systems that provide integrated storage, retrieval and processing of sparse, irregular and multiresolution multi-dimensional data sets.
- Provides support for common operations including index generation, associative data retrieval, memory management, scheduling of processing across a parallel machine, and support for carrying out user-specified computational routines on associatively defined portions of multi-dimensional datasets.

Example Projection Query

Output grid onto which interpolation is carried out

Specify portion of raw sensor data corresponding to some search criterion



Accomplishments

- **Core ADR work has been funded under DICE**
  - Developed ADR from ADR prototype (Titan), generated documentation
  - Customized ADR for driving applications --
    - Bay and Estuary simulation - store, retrieve, process fluid velocity information (U Texas)
    - Virtual Microscope - storage, retrieval and processing of microscopy data (Johns Hopkins medicine, Neuroscience)
    - Storing and processing data continuously acquired from satellites. Current and future data products include: NDVI analysis of AVHRR, TM data, Landsat Pathfinder Tropical Deforestation Project, Coastal Marsh Loss Project, Land Cover Products for Global and Regional Modelers, Modeling of Primary Production, MODIS Science Team Products, Landsat 7 Science Team Products

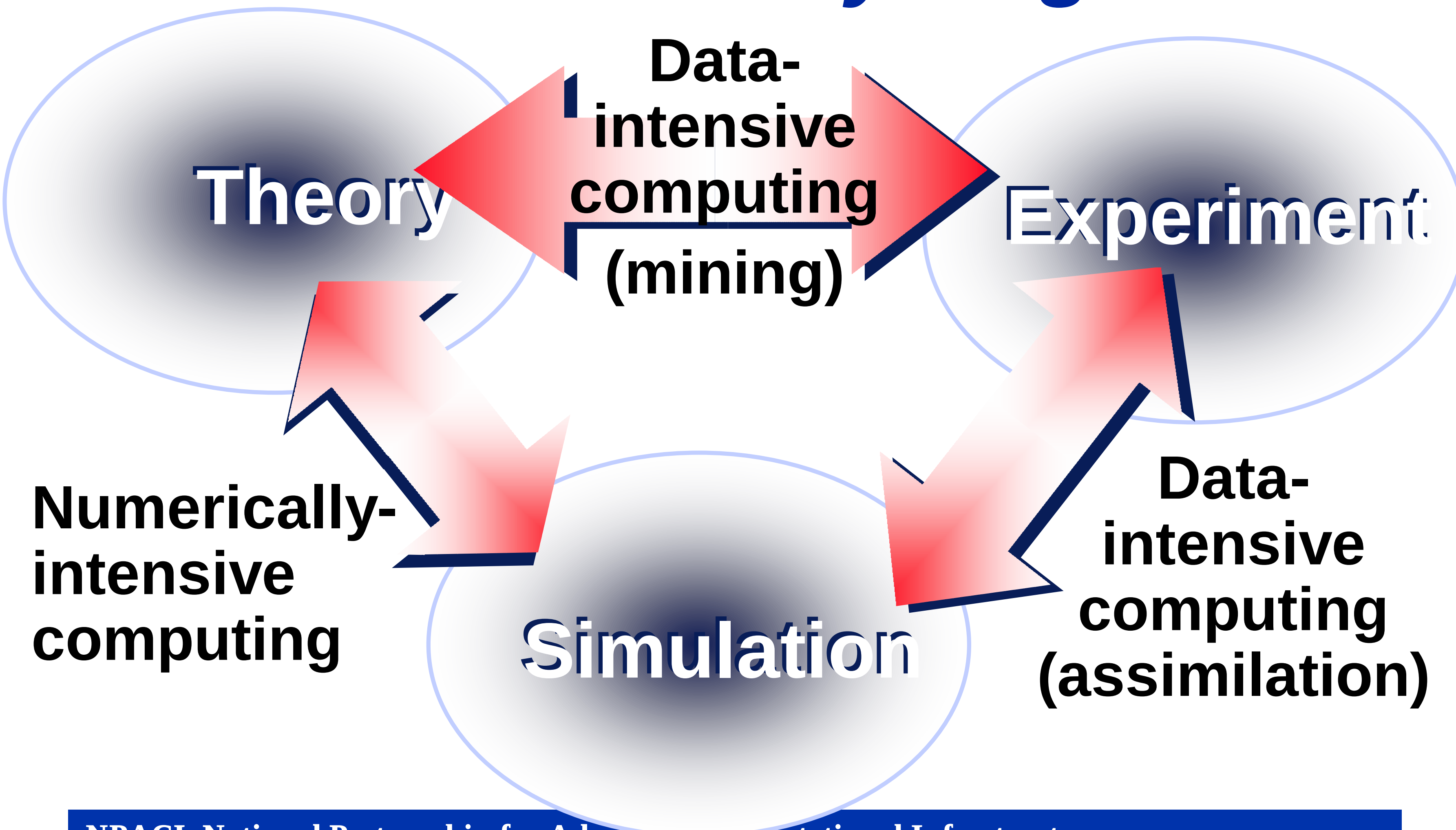
PTE Plans

- **Development of multi-resolution data structures**
  - Store, retrieve, process databases consisting of patches of data with varying resolutions, spatial locations
  - Microscopy, satellite sensor data, adaptive scientific computations
- **Integration of ADR into object/relational framework**
  - **ADR runtime support will be used to develop object-relational database abstract datatypes**
  - **Use compiler-based query optimization methods to optimize both processing and data retrieval**

Software Repository

- A software repository for the Programming Tools and Environments (PTE) thrust area has been set up at http://www.nhse.org/rib/repositories/npaci_pte/catalog/
  - **Uses Repository in a Box (RIB) toolkit that has been developed as part of the National High-performance Software Exchange (NHSE) project**
  - **Repository will be maintained by Leading Edge Site PTE thrust support person in coordination with Jack Dongarra**

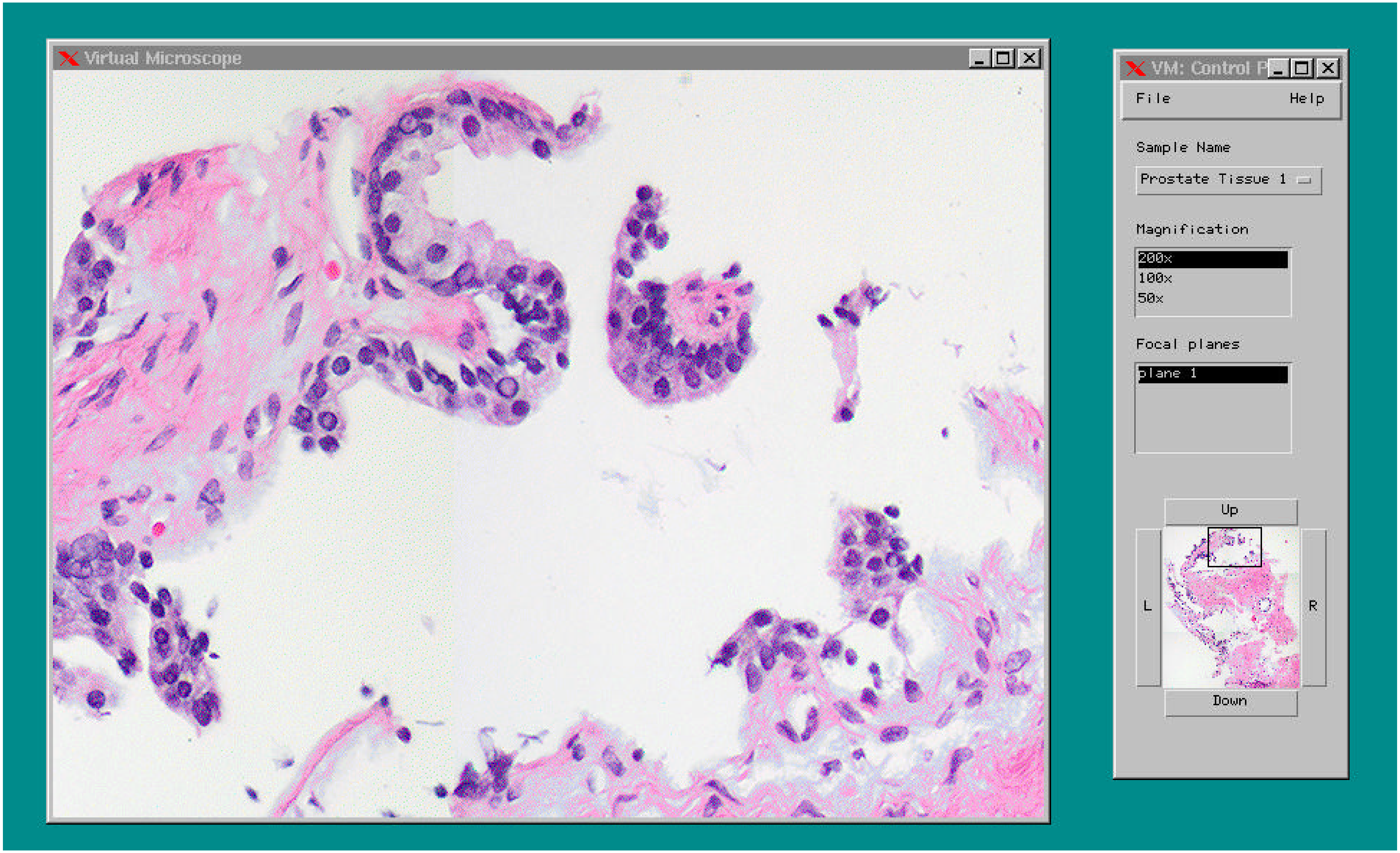
Simulation is Synergistic



Goals and Observations

- **Development of robust supported tools**
  - Organized process for developing prototypes and choosing tools to support
- **Combinations of data processing and coupled applications**
  - Basic science
  - Medicine -- Pathology and Radiology (may be market that could be nurtured)
- **Multiresolution**
  - Frontier for science
  - Nobel prizes in medicine
  - Use in data storage/retrieval/processing in Pathology, Radiology

Virtual Microscope Client



- **Common Adaptive Runtime Support Library (Baden, Saltz, Scott)**
  - Collaboration with Mary Wheeler -- use of KeLP to port block structured groundwater code (PARSSIM), KeLP/AVS interface (Genetti)
  - Workshop to carry out requirements analysis for support of AMR, unstructured applications
  - Adaptive Runtime Support (Ridgway Scott)
    - PC, Pfortran
    - Euler GROMOS
  - Metachaos used to couple libraries

- **Active Date Repository (Saltz)**
  - integrate with Meta-Chaos, Globus/Nexus, KeLP
  - Surface Water and Environmental Modeling (ESS, UT Austin), PV3 to visualize ground water and surface water, Brain Bending and Blending, Microscopy (Neuroscience, UCSD)
- **Netsolve (Dongarra)**
  - integrate with Legion/Globus
  - client server agent for use of library based software
  - Mcell (Salk)

- **Meta-Chaos (Saltz)**
  - Interaction with KeLP, Globus/Nexus
  - Surface water, environmental modeling,, ground water (later) couple ground water, surface water, wetlands (UT Austin)
- **Out-of-core compiler (Kennedy)**
  - Interaction with Active Data Repository
  - Use of Maryland ESS benchmark, possibly UT Austin

- **Titanium (Graham)**
  - Generation of code with functionality equivalent to KeLP and/or embedding of KeLP runtime support; Y2 possible language/compiler front end for ADR
  - applications ported with KeLP
- **Benchmark suite (Yelick, Baden, Dongarra)**
  - Collection of small applications to test and motivate tools and performance studies
  - Leverage Dongarra’s Parkbench work and software repository

- **Parallel FFT Library (Carter)**
  - Will carry out parallel port of FFTW (MIT-Jeremy Johnson) -- FFTW uses a tunable divide and conquer approach
  - 3-D and out-of-core problems will be addressed but time schedule for this is currently undecided
  - Collaborators - Mark Ellisman, Ben Rosen, Tom Prince, Scott Baden, John Weir

Parallel Tools and Environments