CCA
Common Component Architecture

CCA Status and Plans

CCA Forum Tutorial Working Group
http://www.cca-forum.org/tutorials/
tutorial-wg@cca-forum.org



CCTTSS Research Thrust Areas and Main Working Groups

- Scientific Components
  - Scientific Data Objects

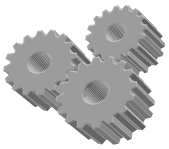
  Lois Curfman McInnes, ANL (curfman@mcs.anl.gov)
- "MxN" Parallel Data Redistribution

  Jim Kohl, ORNL (kohlja@ornl.gov)
- Frameworks
  - Language Interoperability / Babel / SIDL
  - Component Deployment / Repository

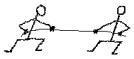
  Scott Kohn, LLNL (skohn@llnl.gov)
- User Outreach

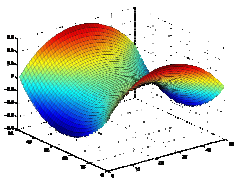
  David Bernholdt, ORNL (bernholdtde@ornl.gov)

Scientific Components



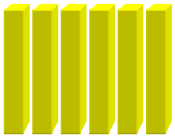
- Abstract Interfaces and Component Implementations
  - Mesh management
  - Linear, nonlinear, and optimization solvers
  - Multi-threading and load redistribution
  - Visualization and computational steering
- Quality of Service Research
- Fault Tolerance
  - Components and Frameworks





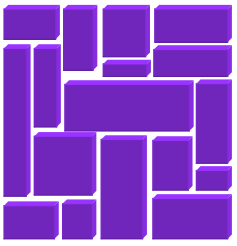
Scientific Components Extended R&D Agenda

- Complete development of abstract interfaces and base component prototypes
- Advanced component development
  - Second-level component extensions
  - Application-specific components for chemistry and climate
- Implement fault tolerance and recovery mechanisms
- Develop quality of service models for numerical components
  - Integrate QoS system into repository
- Develop interfaces and implementations for multi-level nonlinear solvers and hybrid mesh management schemes
  - Collaboration with TOPS and TSTT centers



Scientific Data Objects & Interfaces

- Define “Standard” Interfaces for HPC Scientific Data
  - Descriptive, Not (Necessarily) Generative…
- Basic Scientific Data Object
  - David Bernholdt, ORNL
- Structured & Unstructured Mesh
  - Lori Freitag, ANL
  - Collaboration with SciDAC TSTT Center
- Structured Block AMR
  - Phil Colella, LBNL
  - Collaboration with APDEC & TSTT



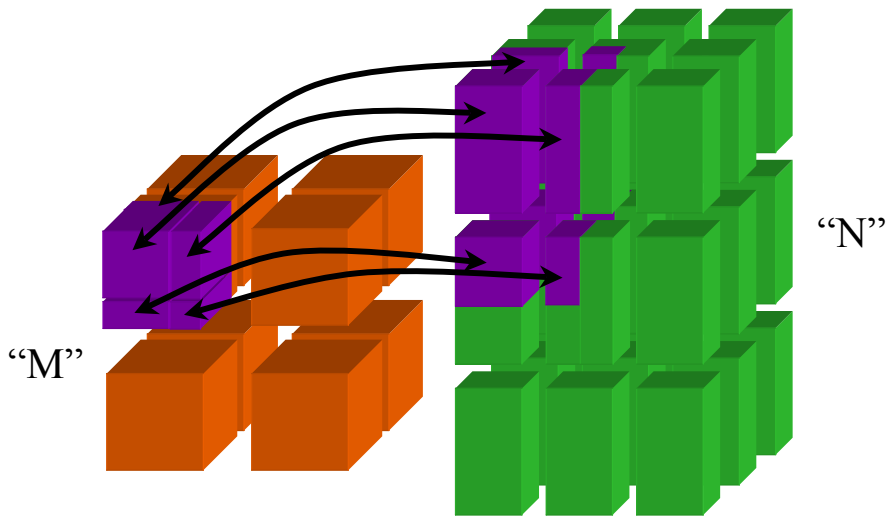
Scientific Data Interfaces

- Low Level, Raw Data
  - Supports high performance access to memory
  - Based on IOVec
    - Assumes a contiguous memory block
    - Supports basic data types such as integer, float, double
    - No topology information
- Local & Distributed Arrays
  - Abstract interfaces for higher-level data description
    - 1D, 2D, 3D dense arrays
    - Various distribution strategies
      - HPF-like decomposition types (Block/Cyclic…)

Mesh Interfaces

- Unstructured Meshes
  - Abstract interfaces for mesh and geometry access and modification
    - Supports geometry and topology access via iterators, arrays, worksets
    - Separates structured and unstructured mesh access for performance
- Block Structured AMR
  - Abstract interfaces for allowing block structured AMR packages to exchange data

"MxN" Parallel Data Redistribution: The Problem…



"MxN" Parallel Data Redistribution: The Problem…

- Create complex scientific simulations by coupling together multiple parallel component models
  - Share data on "M" processors with data on "N"
    - M != N ~ Distinct, Pronounced "M by N"…
  - Model coupling, e.g., climate, solver / optimizer
  - Collecting data for visualization ("Mx1")
- Define "standard" interface
  - Fundamental operations for any parallel data coupler
    - Full range of synchronization and communication options

Hierarchical MxN Approach

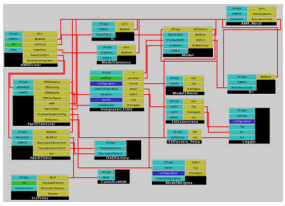
- Basic MxN Parallel Data Exchange
  - Component implementation
  - Initial prototypes based on CUMULVS & PAWS
    - Interface generalizes features of both
- Higher-Level Coupling Functions
  - Units, time & grid Interpolation, flux conservation
- "Automatic" MxN Service via Framework
  - Implicit in method invocations, "parallel RMI"

http://www.csm.ornl.gov/cca/mxn/

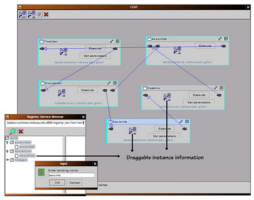
CCA Frameworks

- Component Containers & Run-Time Environments
- Research Areas:
  - Integration of prototype frameworks
    - SCMD/parallel with distributed
    - Unify framework services & interactions…
  - Language interoperability tools
    - Babel/SIDL, incorporate difficult languages (F90…)
    - Production-scale requirement for application areas
  - Component deployment
    - Component repository, interface lookup & semantics

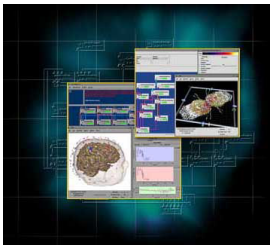
CCA Framework Prototypes



- Ccaffeine
  - SPMD/SCMD parallel
  - Direct connection



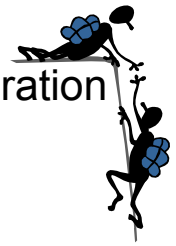
- CCAT / XCAT
  - Distributed
  - Network connection



- SCIRun
  - Parallel, multithreaded
  - Direct connection

- Decaf
  - Language interoperability via Babel

Outreach and Applications Integration



- Not Just “Thrown Over The Fence”…
- Several Outreach Efforts:
  - General education and awareness
    - Tutorials, like this one!
    - Papers, conference presentations
  - Strong liaison with adopting groups
    - Beyond superficial exchanges
    - Real production requirements & feedback
  - Chemistry and cimate work within CCTTSS
    - Actual application development work ($$$)
- SciDAC Emphasis
  - More vital ***applied*** advanced computing research!

Current CCA / CCTTSS Status

- CCA Specification at Version 0.5
- Several Working Prototype Frameworks
- Functional Multi-Component Parallel and Distributed Demonstration Applications
- Draft specifications for
  - Basic scientific data objects
  - MxN parallel data redistribution
- Demonstration Software ***Available for Download***
  - 4 different “direct connect” applications, 1 distributed
  - 31 distinct components, up to 17 in any single application, 6 used in more than one application

CCA Tutorial Summary

- Go Forth and Componentize…
  - And ye shall bear good scientific software
- Come Together for Domain Standards
  - Attain true interoperability & code re-use
- Use The Force:
  - http://www.cca-forum.org/tutorials/
  - http://www.cca-forum.org/software.html
  - *tutorial-wg@cca-forum.org*
  - *cca-forum@cca-forum.org*