Review of Major Concepts

Richard Sinnott
Colin Perkins
John Watt

http://csperkins.org/teaching/2004-2005/gc5/

UNIVERSITY
of
GLASGOW

Aims of This Module

Grid Computing is a developing area:

- Our understanding of the problem space is still evolving
- Systems and standards change frequently
- Many open research issues
- No-one has all the answers

Accordingly, we aim to:

- Produce Grid-savvy individuals
- Encourage critical thinking about Grid-related technology
- Pique your interest in Grid-related research issues

Intended Learning Outcomes

- Thorough grounding in the architecture of the Grid, and exposure to various implementations of the infrastructure
- Experience in using one particular implementation to construct a Grid-based application
  - Competence in Grid programming
  - Exposure to large-scale cluster computing facilities
- Awareness of current open research issues relating to the Grid architecture and infrastructure
  - How do they differ from distributed systems?
  - What are the challenges in making Grids work?

Material Covered

- Introduction to Grid Computing
- Scalability and Heterogeneity
- Open Standards and Architectures
- Implementations of the Grid Architecture
- Resource Discovery/Information Services
- Web Services
- Technologies for Building Grids
- Grid Security Concepts
- Virtual Organizations
- Security in Practice
- Job Scheduling and Management
- Workflow Management
- Data Access, Integration & Management
- Data Provenance and Curation
- Bulk Data Transfer
- Peer-to-Peer Communication
- Tools for Collaboration
- Sample Applications
- The Future of Grid Computing

Introduction to Grid Computing

Infrastructure for Internet-scale Distributed Systems

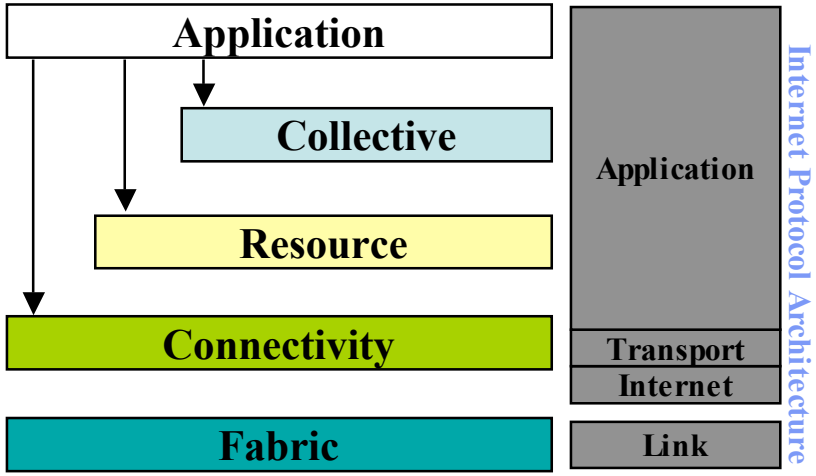
- "A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities"
- Main challenge in building a Computational Grid: to integrate the disparate resources a virtual organization needs:
  - Service and resource management protocols
  - Security solutions
  - Scheduling and workflow management
  - Data management/Provenance services
  - Scalable and efficient data transfer services
  - Tools for collaborative work

Scalability and Heterogeneity

- The two biggest challenges to designing a computational grid are **heterogeneity** and **scalability**
  - Heterogeneous of users, networks and organizations
  - Scalability of Data Storage & Distribution
  - Scalable Scheduling
  - Naming, Addressing and Middleware
  - Robustness and Fault Tolerance
  - System Configuration Management
- These distinguish grids from traditional distributed systems

Open Standards and Architectures

- Driven by technology
  - Globus toolkit v1, v2
- Move to OGSA
  - Aims for usability, extensibility, site autonomy, no central mgt, …
  - Pre-March 2004 OGSI Grid services
    - Stateful, persistent/transient services, service data …
  - Post-March 2004, move towards more pure web service
- OGSA and associated technologies moving targets
  - Work in progress; Standards in complex area
    - Challenges in software engineering, paradigm shifts
  - Technologies; Globus toolkit v3, v4
  - Challenges in understanding scientific domain shaping software/standards
  - OGSA standard vs. set of guidelines
    - Issues of conformance, compliance, consistency to OGSA architecture?



Implementations of Grid Architecture

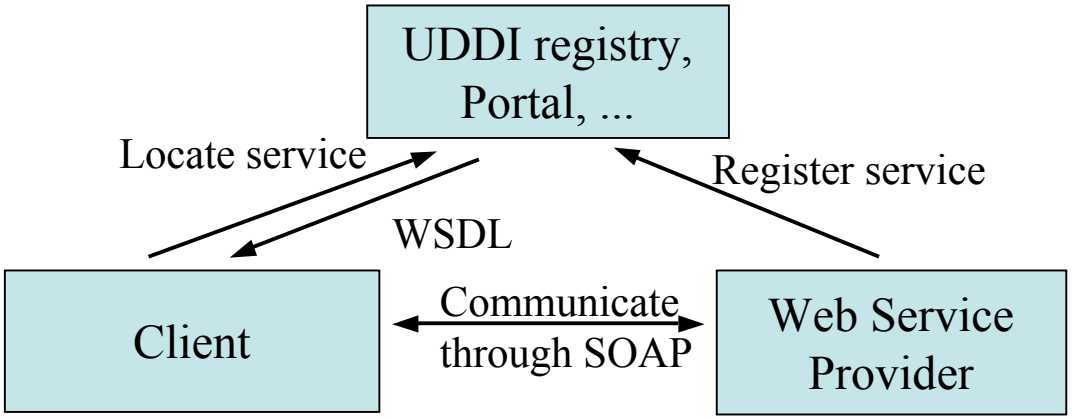
- Apple X-Grid, .NET, 'Condor', Globus
- Globus Alliance produces Globus Toolkit + helps define grid standards. – GT "'de facto' grid technology" (New York Times)
- GT3 is a useable implementation of OGSI (hence OGSA)
- Three pillars:
  - Resource management, information services, data management
  - All pillars use GSI at the connection layer
- Resource Management Requirements
  - Create job environment, stage files, submit to a scheduler, monitor and send notifications, stream jobs output
  - GT2 – GRAM
  - GT3 – Managed Job Service
- GT3 Job submission based on Grid Service Factory model
  - Service creation, invoke operation, control job

Resource Discovery/Information Management

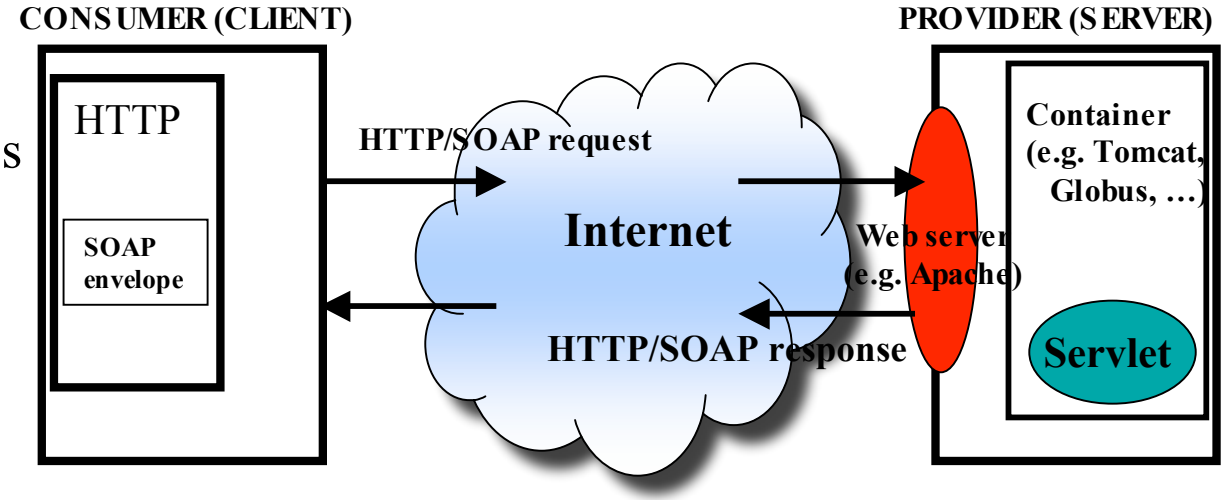
- System information - critical to operation of grid
  - What resources are available? What is the ‘state’ of the grid? How to optimise resource use (tailor to specific app)
- Any information infrastructure should provide mechanisms for:
  - Discovery, monitoring, planning, adapting application behaviour
  - Solve problems of distribution and diversity, failure management, security
- Two entities: information providers, and index services
- Implementations:
  - MDS2 – Meta Directory Service based on LDAP
    - GRIS – ‘White’ and ‘yellow’ pages lookup
    - GIIS – cache like a web search engine
  - MDS3 – Uses XML serviceData to publish information by query or subscription
    - xPath queries on XML, notifications
    - GT3 Index Service – ALL info published through ONE service

Web Services

Web Services Overview

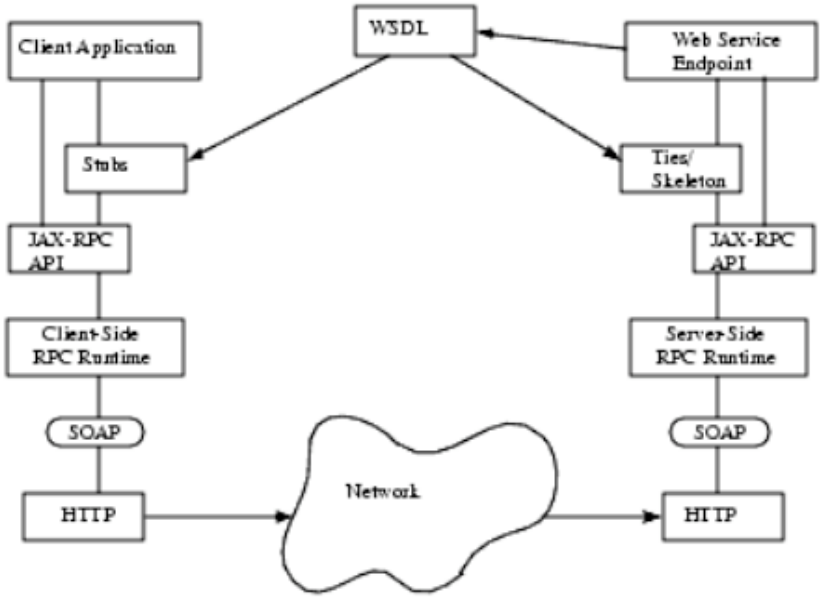


- Technologies associated with web services
  - Extensible Mark-Up Language (XML)
  - XML schema
  - XML namespaces
  - Simple Object Access Protocol (SOAP)
  - Web Services Description Language (WSDL)



Technologies for Building Grids

- Major Java environment packages for writing and deploying Java services and clients
  - J2EE, JWSDP, …
- Basic concepts of containers
  - Axis, Tomcat
- JAX-RPC basics
- Basics of deploying to containers
- Introduction to the ANT tool



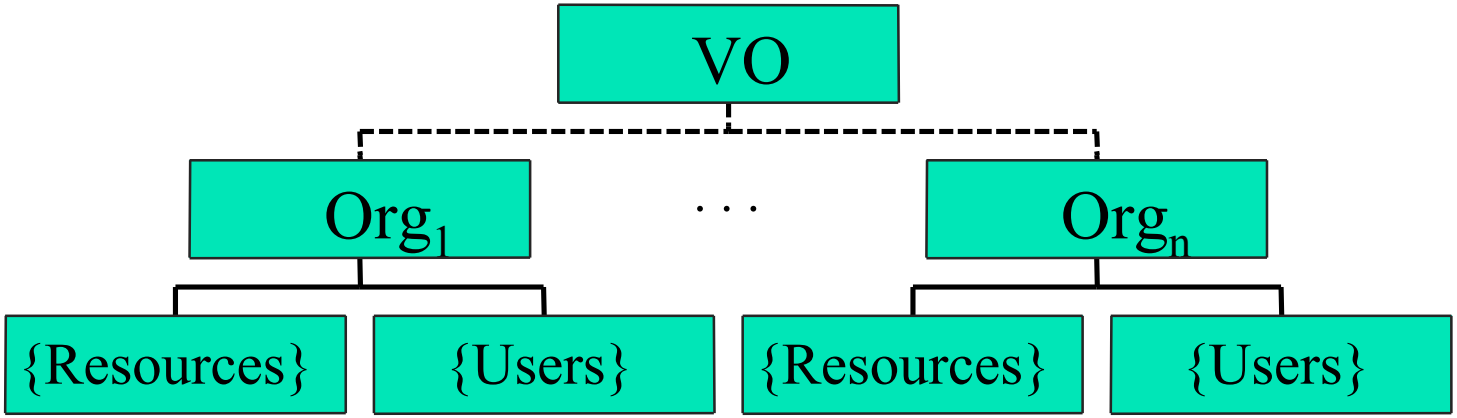
- Various examples of seeing these technologies being used

Grid Security Concepts

- Why is Grid security so important?
- The Challenge of Grid Security
- Technical challenges
  - Technologies to help make Grids secure
    - Public Key Infrastructures
  - Security always depends on the weakest link
- Concepts
  - Authentication
  - Authorisation
  - Audit/accounting
  - Confidentiality
  - Privacy
  - Integrity
  - Fabric management
  - Trust
- Social challenges
  - Educating users in security issues
- Manageability
  - Systems must be easily configurable, changeable when security threats arise / have arisen
- Usability
  - Systems must be usable by non-computer scientists
- Scalability
  - Must allow for a multitude of different classes of user

Virtual Organizations

- VO - dynamic collection of distributed resources shared by dynamic collection of users from one or more organizations



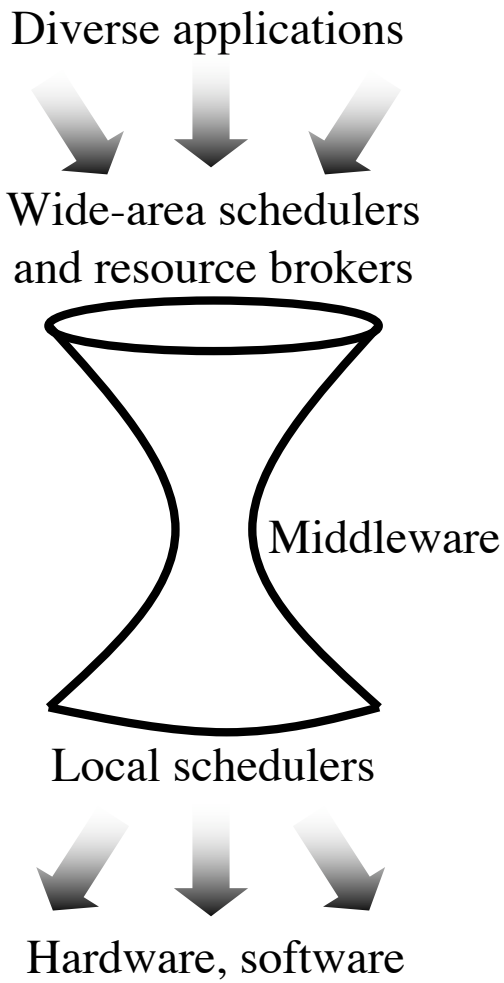
- VO local vs global policies
  - Broad array of requirements from applications
    - Security, data management, high throughput computing…
- Technologies for VOs
  - Authorisation, rules ala PERMIS
  - Generic Grid solutions GGF SAML AuthZ
    - Must deal with potentially huge number of users, resources

Security in Practise

- Focus on Grid Security Infrastructure (GSI)
  - Secure comms, site policy control, single sign-on
- Digital Certificates
  - X509, Certification Authority, digital signatures, mutual authentication
  - Delegation
    - Allows single sign-on to grid resources through proxy certificates
- GSI Implementation
  - Authentication: grid-proxy-init
  - Authorization: grid-mapfile
  - Improved security model - GRIM
    - Services connected to the network run under restricted account – worst case scenario – denial of service
- Others
  - RBAC (e.g. PERMIS – ACs, Roles)

Job Scheduling and Management

- Job Scheduling and Management Concepts
  - Resource Discovery
  - Job Allocation and scheduling; planning *vs*. scheduling
  - Code and Data Distribution
  - Need for middleware
  - Need for predictable, secure, robust and fault-tolerant execution environment
- Implementations
  - Condor, OpenPBS, Sun Grid Engine, Xgrid
  - GRAM, Condor-G
- Fundamental issues to consider:
  - Communication latency
  - Organization policy
  - Robustness and fault tolerance



Workflow Management

- New field. No standards. Try to bring the Grid to the user, not user to the Grid.
  - Represented by an XML-based workflow definition language
    - BPEL4WS, XPDL, SWFL…
  - Use graphical representation to construct the XML description
    - Use VSCE – visual connection on a ‘canvas’ e.g. Taverna
  - Submit constructed workflow to enactment engine
- Challenges:
  - Semantic compat. – links services with similar meaning different names
  - Syntactic compat. – links services with common data types
  - Problem of determining and comparing behaviours of interacting services
    - Additional metadata on service provenance and semantics
    - Services in workflow may not bind to specific service instances at runtime
      - Compile ‘n link
- Issues – parameter constraint, IDs, security, engineering

Data Access, Integration and Management

- Requirements:
  - Wish to access data directly from sources and resources (telescope, DB)
  - Want to be able to access ALL types of data in EVERY format
  - Access data in non-application specific repositories
  - Data publishing, operations, access control, provenance
- OGSA-DAI
  - Middleware to grid-enable existing databases
  - Aims to provide a component library to allow
    - Common interface to data resources, integration of distributed queries
  - Uses grid data service factory (GDSF) model
  - Projects: FirstDIG, ODDGenes
- Underlying technologies (GT based)
  - GridFTP, RFT, Replica Location, XIO API

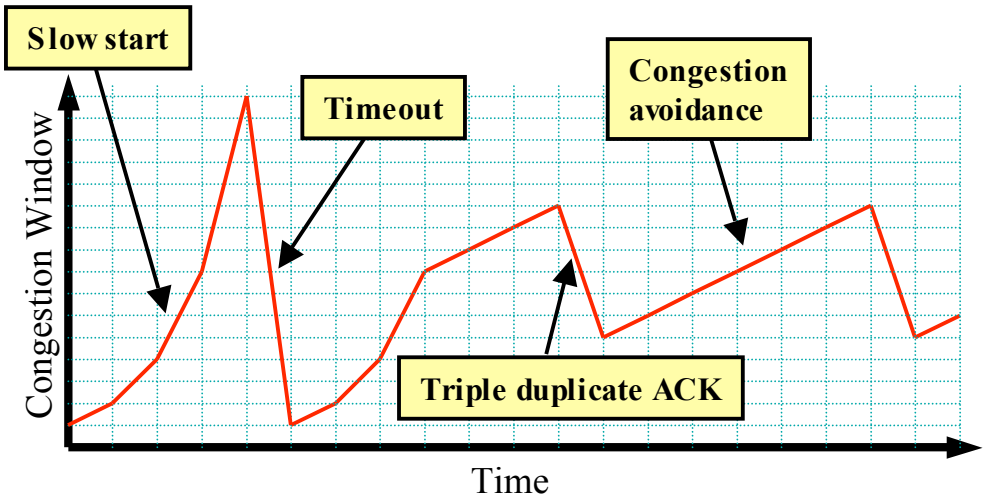
Data Provenance and Curation

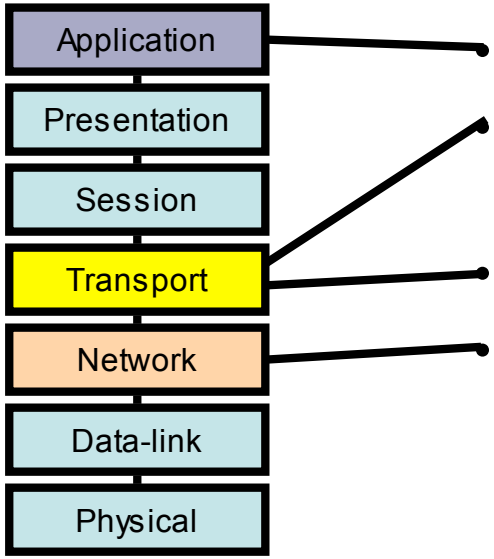
- Longevity?
  - what to keep? we can't keep it all ... we don't even want to keep it all...
  - therefore what to mark with provenance information
- Keeping track of this information in a Grid environment
  - copies of copies of copies,
  - then the issues of keeping track of provenance for these copies - merging result sets?
  - provenance information likely to add huge bloat over time - how to manage/control this?
- How do we do this anyway
  - what technologies XML?
- Do we need provenance/curation services,
  - what would/should these look like?
  - who does the mark-up?
  - how do they do this?
    - e.g. applications processing data automatically include provenance information on results
    - can we leave it for centres such as NDCC to provide services we can all use?
  - Is curation/provenance something that the Grid community needs to address?
  - Can we leave it for others, e.g. DB community?
- How can we make it all usable?
  - "…run job X with data Y (all sorts of provenance info on data goes here) on some Grid infrastructure (possibly 1000+ resources) and put results on computer Z" to do this right, we need to ensure that all information on what job/what location/what machine architecture/what OS/what time/... etc can be obtained from results so that others could in principle check our results/repeat our experiment etc etc

Bulk Data Transfer

- TCP congestion control
- Limitations of TCP/IP for high performance networks

$$T = \frac{s}{R\sqrt{\frac{2p}{3}} + 3p(1+32p^2)\cdot T_{rto}\sqrt{\frac{3p}{8}}}$$





Parallel streams

Modify the TCP congestion response

- E.g. High Speed TCP, Scalable TCP, H-TCP

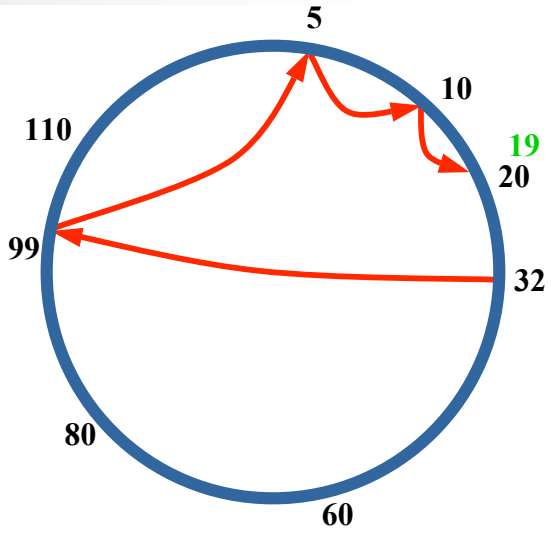
Replace TCP (e.g. XCP)

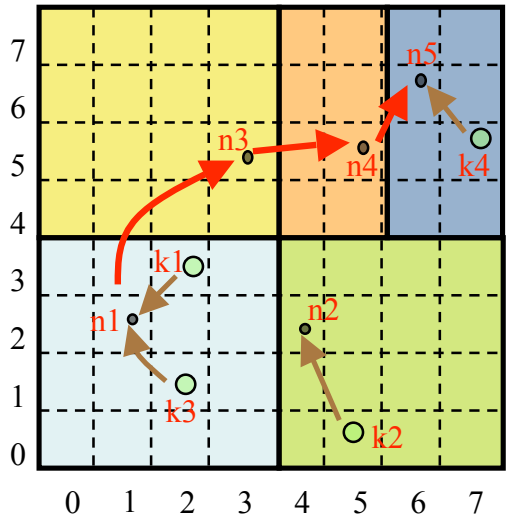
Modify the IP layer to avoid congestion

- Integrated and Differentiated Services
- MPLS and Optical Switching

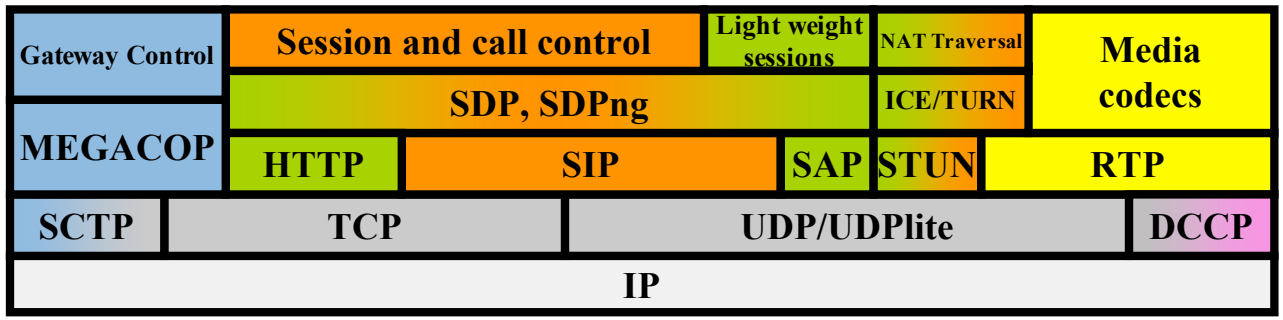
Peer-to-Peer Communication

- How peer-to-peer might be used by Grid computing systems
- Operation of distributed hash tables
- Uses of distributed hash tables
  - Object location systems
  - File sharing applications
  - Publish/subscribe event notification systems
- Distributed monitoring and aggregation systems
  - e.g. Astrolabe
- How NAT and firewalls affect peer-to-peer application deployment





Tools for Collaboration

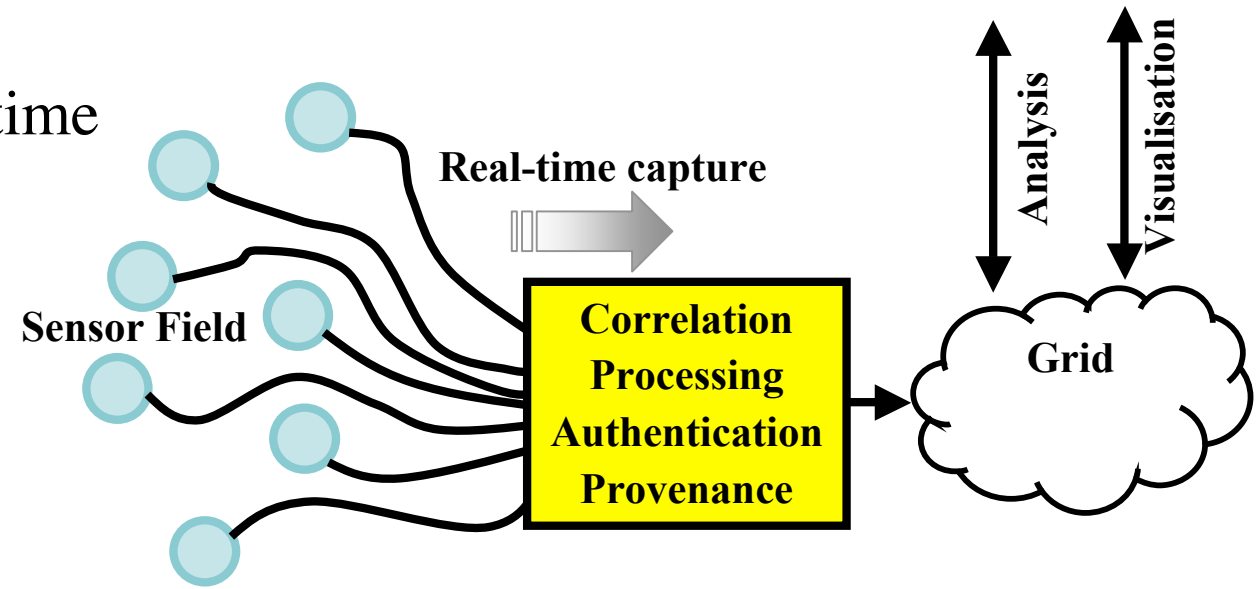


Multimedia protocol stack: separate media + signalling

Components that form a collaborative work system

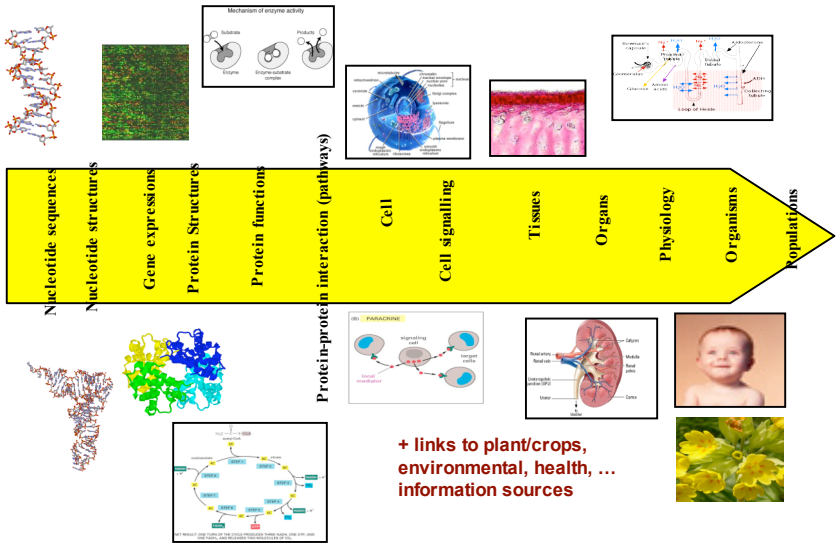
Architecture for real-time data dissemination

Re-use of multimedia protocol components



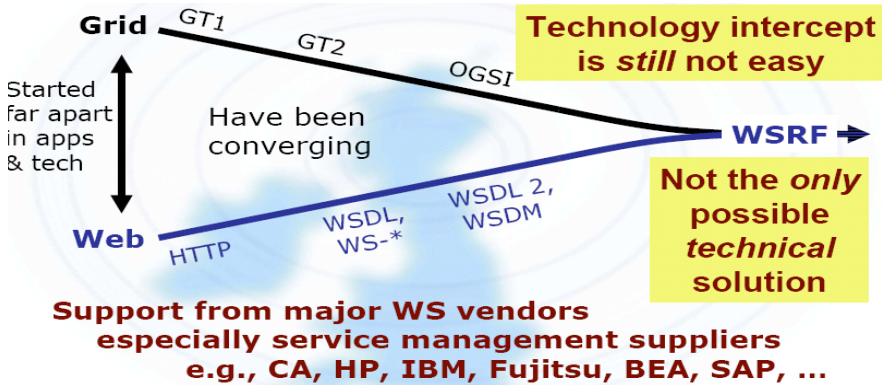
Sample Applications

- Physics
  - Particle physics, gravitational waves, astrophysics, …
- Engineering
  - Electronic, civil, mechanical, aerospace, …
- Computer science
  - Simulation, information retrieval, text mining, …
- Life sciences
  - Exponential data growth
    - Genomes everywhere
    - Structure DBs
    - Clinical data sets
    - …
  - Expanding application base
    - Amazing demonstration of BRIDGES
    - It really did work on Friday morning!



The Future of Grid Computing

- Classification of Grids
  - Compute Grids, Data Grids, Complexity Grids,
    Campus Grids (focused on Glasgow), Enterprise Grids, Semantic Grids,
    Lightweight Grids, Collaboration Grids, Autonomic Grids
- OGSA Future
  - WSRF
- Future application drivers
  - All domains
    - Social sciences
    - Environmental sciences
    - Physical sciences
      - Production level services for particle physics in 2007
    - Arts,
    - ...
  - Life sciences biggest driver for future?
    - Data data everywhere, linkage to clinical, medical, genomic, ...
    - In-silico research, big business, ...



Borrowed from Malcolm Atkinson talk

Wrap-Up

- Exam will take place after Easter
  - Sample paper available next semester
  - Revision lecture will likely be in April
  - (*dates to be confirmed*)

- Programming assignment due tomorrow at 5pm
- Tutorial tomorrow: “Q&A”…any interest? Or should we cancel?

- Please hand in any remaining module questionnaires to the teaching office