FutureGRID: A Program for long term research into GRID Systems Architecture

Jon Crowcroft, Steve Hand, Tim Harris, Ian Pratt

The Computer Laboratory, University of Cambridge

+ Andrew Herbert, Director, Microsoft Research Cambridge
0. Introduction

- Program of work between the Computer Lab, and Microsoft Research
- Builds on existing collaborations
- Designed as a set of loosely coupled basic research projects
- Common elements to projects, which lead to understanding
- Later, full systems architecture will emerge for a Future GRID.
- PhD studentships efficient use of funds (and to be honest, we have more good applicants than money 😊)
1. Who, where, how, what

- Collaborative tools based on Scribe and Pastry instead (or as well as) IP multicast (P2P CSCW) (existing RFC on PGM etc)
- Search based on locality and on partial content matching (publications this month)
- Computation based on large scale systems and massively redundant partition of computational problems (a.k.a. spread spectrum)
- Extension of Pasta work on mutable, persistent P2P storage (publications)
3. Peer-peer networking == GRIDng
P2P-GRID networking

Focus at the application level
4. Microsoft Grid Investments

- **Aims**
  - Equal opportunities for our platforms
  - Alignment of Grid with industry web services standards (SOAP, WDSL, etc)

- **Approx $1M grant to Globus project**
  - Port Globus to Windows platform
  - Develop OGSA and align with MS evolving web services architecture (GXA)

- **“Rotor” Common Language (.Net) Runtime**
  - Shared source for academic use
MSRC Portfolio

r Peer-to-Peer systems
  m Pastry: best of breed overlay network
  m Scribe: scalable event notification (= multicast)
  m PAST: archival file system
  m SQUIRREL: distributed cooperative data caching
  m OVERLOOK: dynamic DNS (=discovery)

r Economic models for resource sharing
  m Main focus: network congestion avoidance, especially for streamed A/V
  m Also: disc scheduling, OS buffer cache management

r Trustworthy distributed computing
  m Efficient Byzantine fault tolerance
MS Corporate Interest

- Evolution of web services towards computing utilities
  - Passport, .Net My Services as first attempts to offer infrastructure components and services
  - PNRP in OS as network extension of Universal Plug and Play

- Evolution of Office personal productivity suite towards support for collaboration across virtual organizations
  - Sharepoint portal, investment in Groove Networks as first steps
Paths to exploitation

- Basic research in P2P and resource management technology “mostly done”
- MSRC now searching out compelling applications to stress test implementations and demonstrate benefits
- Opportunities
  - Collaborative (Access Grid) results -> MS Office
  - P2P middleware results -> MS GXA evolution
    - E.g., Pastry as a P2P “aspect” in VS.Net GXA framework
    - E.g., Pastry protocol built into Windows OS
    - E.g., P2P (re-)implementations of core system components (Domain Controller)
  - Resource management results -> OS “scaling out” facilities
- Standards
  - Co-evolution of MS GXA and GGF OGSA architecture
Commitment to FutureGrid

- Director level support (Andrew Herbert)
- Funding for 1-2 research students
  - Awaiting confirmation of FY03 budgets...
- Participation of researchers
  - Ant Rowstron, Miguel Castro, Anne-Marie Kermarrec (P2P, Gossip Multicast)
  - Peter Key, Richard Black, Richard Mortier (Resource management)
  - Jim Gemmell [MS BARC]
- Early access to GXA
5. The Four Projects

- PhDs + Some level of RA
- Note also effort at Microsoft Research
- And later, exploitation in E-Science program...
IP Multicast - Project 1

- No duplicate packets
- Highly efficient bandwidth usage

Key Architectural Decision: Add support for multicast in IP layer
Concerns with IP Multicast

- Scalability with number of groups
  - Routers maintain per-group state
  - Analogous to per-flow state for QoS guarantees
  - Aggregation of multicast addresses is complicated

- Supporting higher level functionality is difficult
  - IP Multicast: best-effort multi-point delivery service
  - End systems responsible for handling higher level functionality
  - Reliability and congestion control for IP Multicast complicated

- Inter-domain routing is hard.

- No management of flat address space.

- Deployment is difficult and slow
  - ISP’s reluctant to turn on IP Multicast
End System P2P Multicast

Overlay Tree
Why is self-organization hard?

- Dynamic changes in group membership
  - Members join and leave dynamically
  - Members may die

- Limited knowledge of network conditions
  - Members do not know delay to each other when they join
  - Members probe each other to learn network related information
  - Overlay must self-improve as more information available

- Dynamic changes in network conditions
  - Delay between members may vary over time due to congestion

- Use Pastry/Scribe P2P system as it provides precisely these characteristics...
Overlay Multicast Architecture
P2P Search: basics - Project 2

retrieve \( K_1 \)
**Vector Space Search**

- Existing systems use flat unstructured keys
  - Let’s extend this to a virtual multi-dimensional space

- Entire space is partitioned amongst all the nodes
  - Every node “owns” a zone in the overall space
  - Self-stabilizing mechanisms manage nodes entering and exiting from the system

- Abstraction:
  - Keys can be represented as “points” in the space (perhaps with associated values)
  - Messages can be routed for a particular key to the node that owns that “point”
Vector Space Search: applications

- Resource discovery:
  - Points represent resource requirements of jobs and resource availability of machines
  - Nodes act as brokers between jobs and systems that can host them

- Network position could be reflected in the broker’s co-ordinates
  - Promote scalability through disjoint operation of user communities when requests are satisfied by local facilities
Distributed resource location

1. Determine machine locations and resource availability
2. Translate to locations in a multi-dimensional search space
3. Partition/replicate the search space
4. Queries select portions of the search space
Spread Spectrum Computing - Project 3

- Use redundancy coding *ideas*
- For code and data,
- Dissemination uses high degrees of replication
- Collection of responses is
  - Distributed (P2P)
  - Fault tolerant (like SETI@Home and the set of ideas in a lot of cryptanalysis work recently)
- Highly Optimised Tolerance (c.f. John Doyle’s work at CalTech).
Global Storage - Project 4

- Available anywhere, anytime - and fast!

- Must cope with node and network failures
  - Use replication, information dispersal codes

- Must cope with `flash crowds`
  - Automatic load balancing and distribution

- Must allow local caching for performance
  - Challenge of maintaining consistency

- Must provide `hands free' administration
  - Self-organizing system
Global Storage with *Pasta*

- Uses P2P Distributed Hash Table techniques
  - More complex structures necessary? B* trees?
- Aims to provide traditional file-system like semantics (incl. efficient mutability, quotas)
- Also, wider look at shared workspaces to support ad-hoc collaboration
  - Not all participants fully trusted...
  - Need versioning, `views` and `overlapping`
  - Object-specific locking and atomicity enforced by storage system
Related publications

- “Xen and the art of virtualization” – under submission to the ACM Symposium on Operating Systems Principles
- “Managing trust and reputation in the XenoServer Open Platform” – 1st International Conference on Trust Management
- “Controlling the XenoServer Open Platform” – 6th IEEE OPENARCH Conference
- “Storage, mutability and naming in Pasta” – 2002 International Workshop on Peer-to-Peer Computing
Status

- Hot off press: have another person from Microsoft
- Have started to deploy in related (EPSRC ProgNet Program project: Xenoservers) on Intel Planetlab
- Next - plans to deploy in eScience centers? Probably NOT until around 2005 (due to support effort (lack of)) at end of project.
- (diff between eSci center and Planetlab, is that we are allowed to break planetlab😊}