The Global Computer

Robin Sharp

Informatics and Mathematical Modelling

Technical University of Denmark (DTU)

DK-2800 Kgs. Lyngby, Denmark

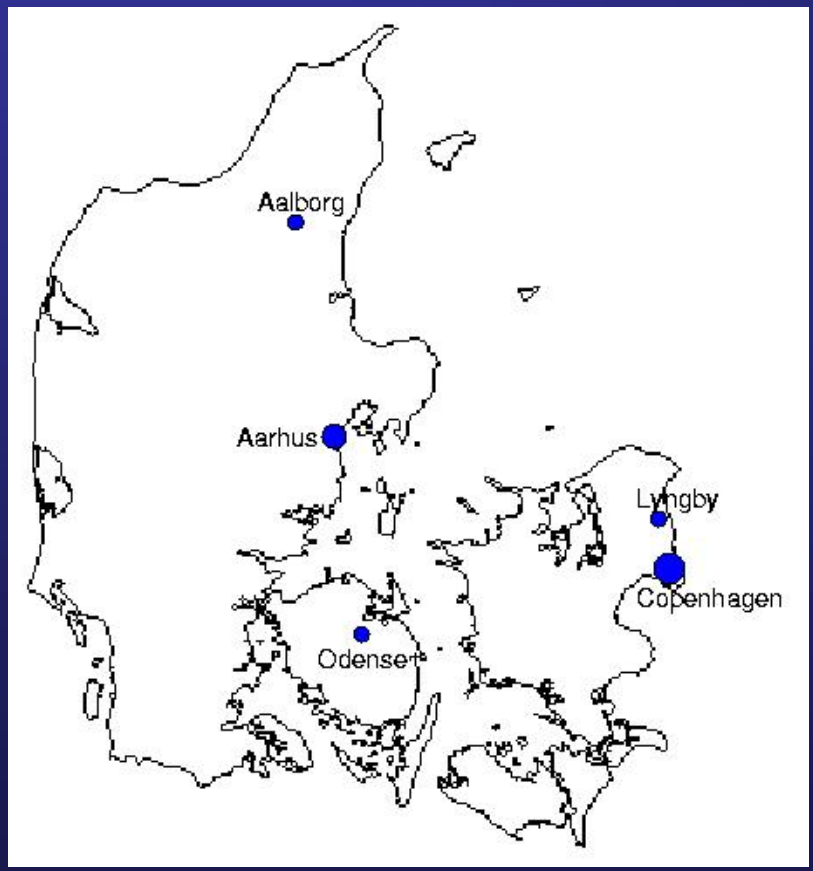
e-mail: robin@imm.dtu.dk

The Global Computer

A GRID project with focus on:

- Computer Science & ICT aspects of very large scale distributed systems, including:
  - Scheduling and load balancing
  - Security and access control
  - Caching
  - Low-latency networking
  - Mobility
- Applications of data-intensive computing

Participants



- Copenhagen University
- University of Southern Denmark (Odense)
- Technical University (DTU, Lyngby)
- Aalborg University
- Copenhagen Business School (HHK)
- UNI-C

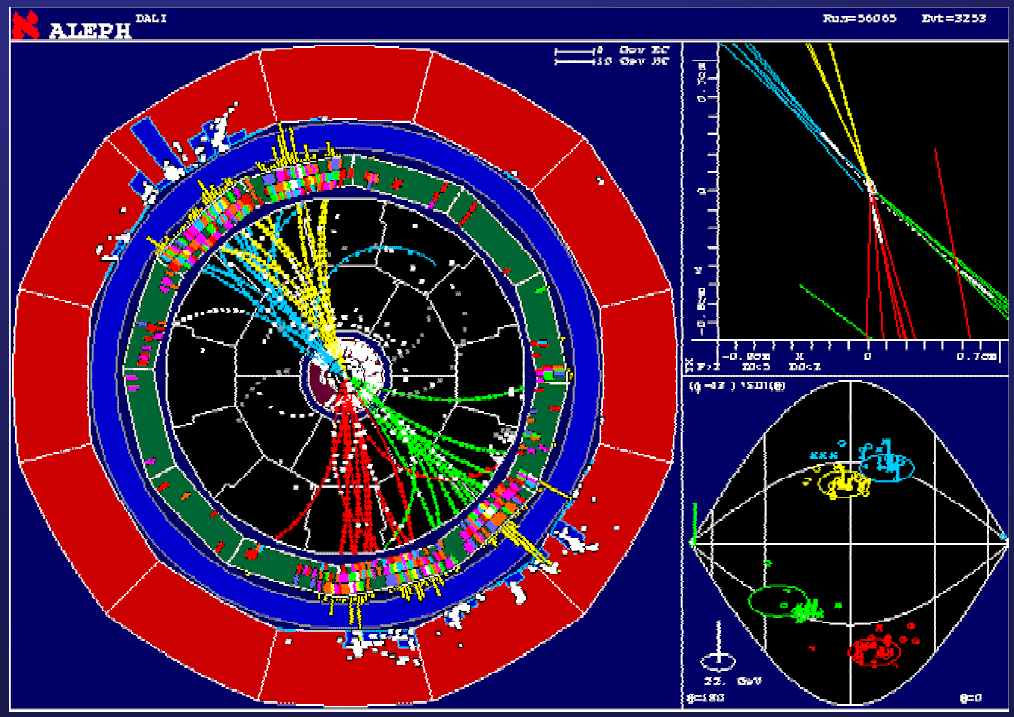
Applications

- Quantum chemistry
- Molecular dynamics
- High Energy Physics
- Model checking
- Experimental mathematics
- 3D Global Illumination
- Global Biodiversity
- … and many more!

Large Applications (3)

- High-energy physics

  CERN particle detectors produce 1-10TB/month



This *might* be a picture of a Higgs boson!

Technical challenges

Large scale distributed systems are characterised by:

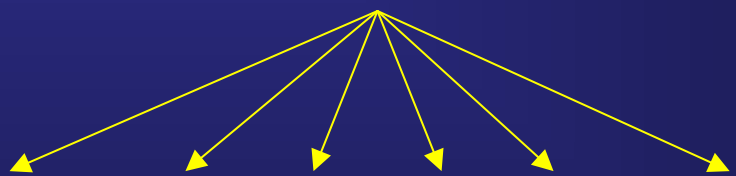
- Dynamic arrival and departure of resources, especially if mobile systems are involved
- Significant network latency
- Need for scalability
- Need for an integrated security policy
- Need for an integrated accounting policy

Aim is to achieve seamless integration of parts.

Simple solutions

Simple solutions such as “screen-saver algorithms” work well when:

- Simple partitioning: identical subtasks
- Simple task dependency graph



- No special scheduling order (or deadlines)
- No inter-subtask communication
- Security similar to ordinary PC applications

More difficult cases

Example: load balancing depends on three main factors:

- Task cost spectrum

Equal — Different — Unknown

- Task dependency spectrum

None — Static — Dynamic

- Locality (data dependency) spectrum. Communication pattern:

None — Predictable — Unpredictable

A general problem...

The Global Computer's highly dynamic environment means that activities such as:

- Scheduling
- Load balancing
- Resource allocation
- Communication
- Security

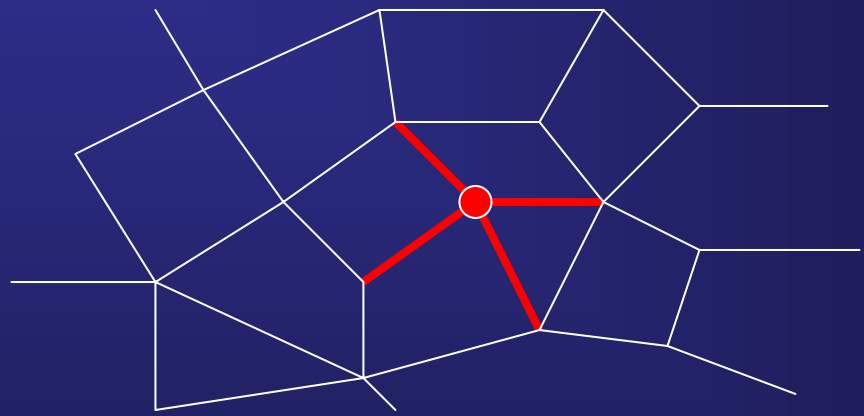
fall into the most difficult part of the spectrum

Scheduling

- Aims to satisfy resource requirements of jobs while optimising system performance.
- Well-known to be computationally hard, even in systems with static resources.
- In the GC, we consider multiple levels:
  - Local cluster (use preferred local strategy?)
  - Global, with job schedule adapted to resource dynamics (load, availability, cost, price,…).
  - Planning level, for longer term scheduling.

Resource allocation

- Most current GC systems use first-fit allocation, starting in the neighbourhood of the initiator of the task.



- This does not necessarily give the optimal global allocation of resources.
- Design of better algorithms for use in dynamic environments is a major research challenge.

Resource allocation (2)

- It is a difficult problem to keep track of available resources in a large, dynamic distributed system.
- Current solutions do not scale well: It may not be possible to efficiently maintain a consistent, up-to-date snapshot of the system.
- Even if we find resources, they may disappear or be withdrawn before we can use them!
- Resource allocation time may be so unpredictable that we have to pre-allocate resources.
- Approach: cost minimisation to achieve scalability.

Communication

- Main aim is to ensure connectivity and capacity required, and to minimise latency.
- Usability of a resource may well depend on latency involved in reaching it!
- This requires us to think more about the physical structure (and technology…) of the network than in traditional Internet usage.
- Open question: which topologies are best?

Caching

- As data rates and CPU power rise, network latency continues to be a bottleneck.
- Caching of internal data and files is needed.
- Aim: to combine caching with a simple shared memory abstraction for ease of application programming.
- Important that this should be scalable for a useful range of applications.

Security

Characteristics of large cluster environments:

- Large, dynamically changing user group
- Dynamically varying pool of resources
- Local and global security policies
- Users may have different status at different sites
- Possibly international collaboration

… all make it hard to find a good solution.

Authentication

- Need to ensure that a user and a resource can each be certain of the other's identity.
- Many open questions in practice:
  - How often to perform authentication?
  - Who should we trust?
  - Will global solution be compatible with local one, giving seamless integration?
  - Legal problems?
  - Is the technical solution scalable to megaclusters?
  - Does solution also work for mobile computing?

Confidentiality

- For some applications, only the parties involved should know about the data.
- This may even include knowing that a certain user has searched a certain (area of a) public database!
- One approach might be to use, say, a secret sharing protocol. Can this be done efficiently?
- Is anonymity compatible with proper authorisation?
- Are the solutions scalable to megaclusters?

The immediate future

- Work is currently going on to set up testbeds for the various aspects of this project.
- Collaboration is hoped for with other Nordic and EU projects.
- Many applications are waiting in the wings.

All we need is more funding...

Thank you for your attention

Nordunet, April 2002