

Grids, Web Services & Peer-to-Peer Applications

**Edinburgh University IT Conference
9th April 2003**



Paul Anderson
School of Informatics
University of Edinburgh
<dcspaul@inf.ed.ac.uk

Grids, Web Services, & Peer-to-Peer Applications:



Web Services.



Grid Computing.



Peer-to-Peer Applications.

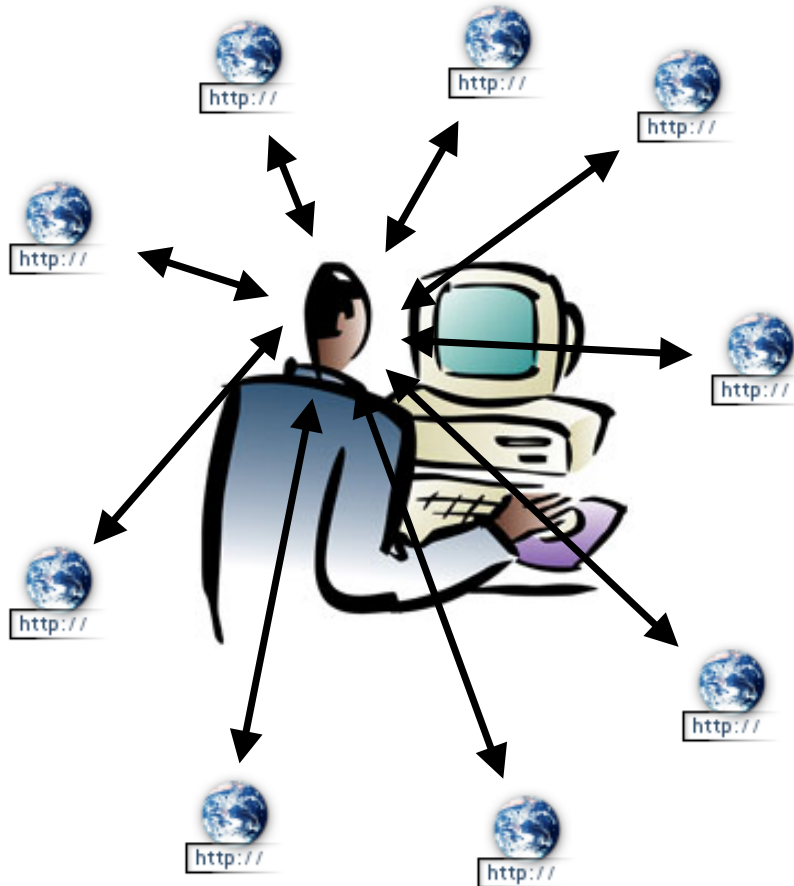


Common Issues.



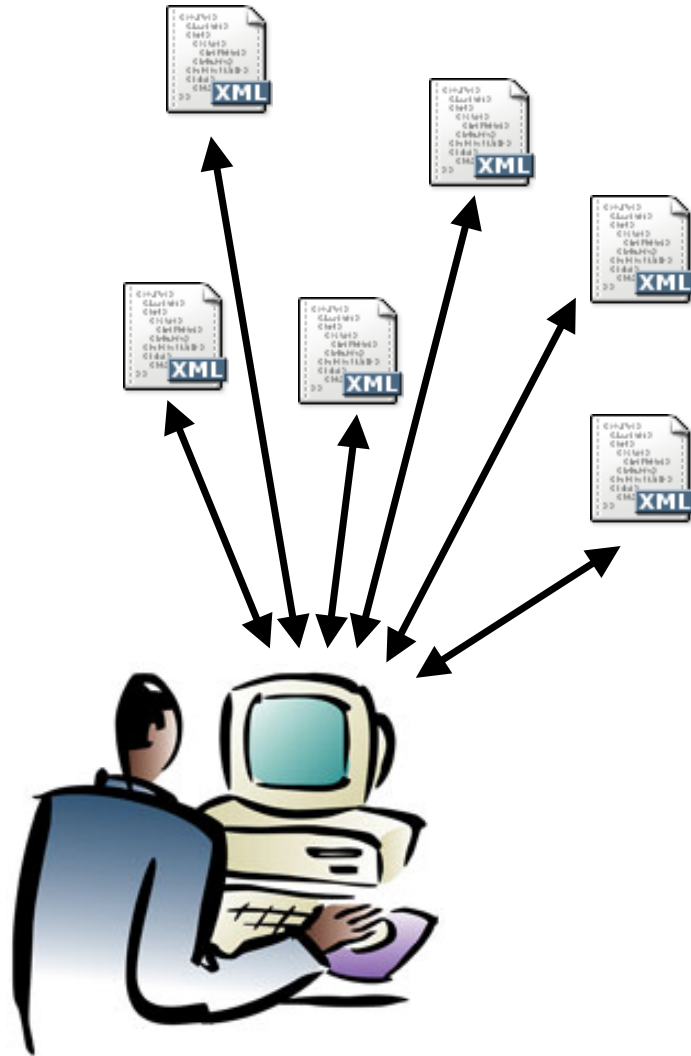
Some Stargazing.

Traditional Use of the Web



- Traditional web usage involves direct interaction between the user and separate remote resources.
- Any collation of the information, or interaction must be performed manually.
- The HTML language and current browsers are designed to support this type of interaction.

Web Services



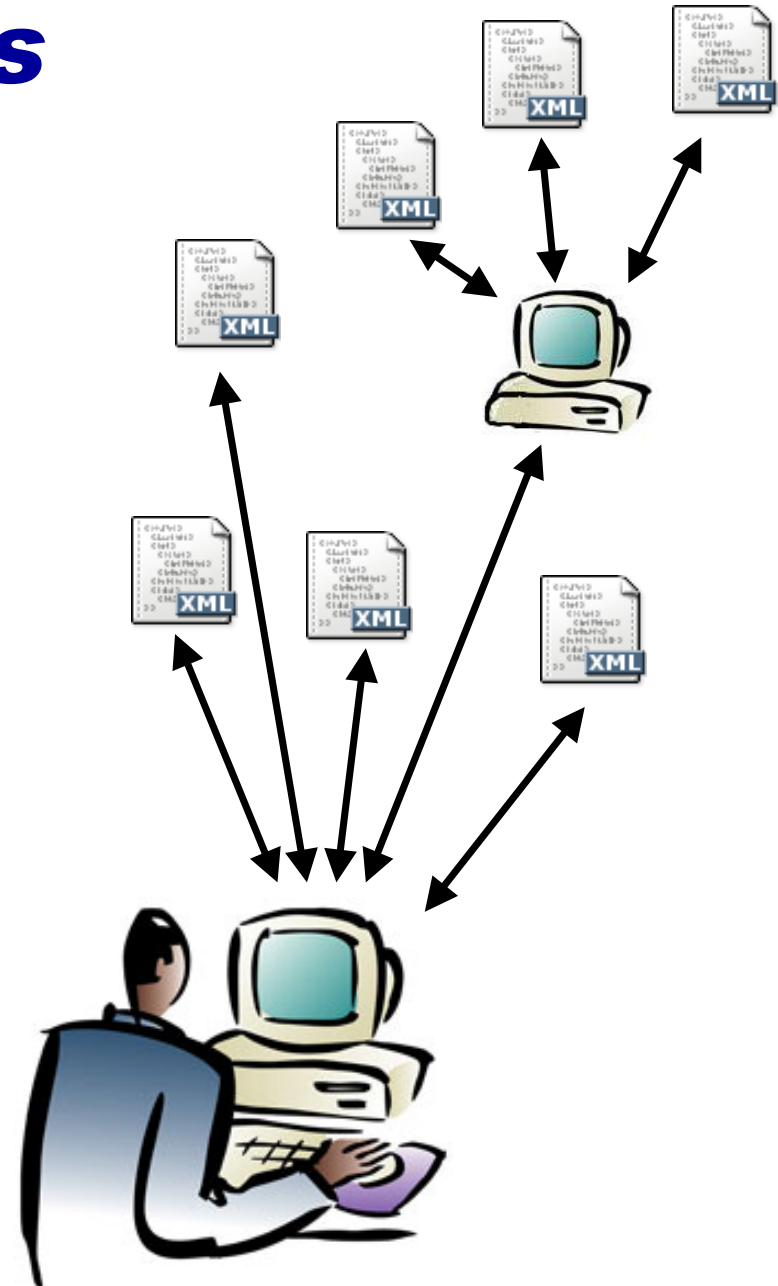
- “Web Services” are designed for interaction between two programs.
- A program on the user's local machine can interact with the remote resources and collate information for presentation to the user.
- XML is used instead of HTML. “SOAP” is the most common protocol, but this usually still uses HTTP.

Web Service Examples

- Some Simple examples:
 - My home finance program uses a web service to fetch the exchange rates from a remote server.
 - A word processor might use a web service to query an online dictionary or a citation index.
 - An address book program might use a web service to look up a postcode.
- Some simple free experimental services:
 - <http://www.xmethods.com/>
- Some of these things are possible using other technologies:
 - but standard web service technology makes it possible to "mix and match".

Web Service Issues

- Location services enable a client to locate services of a particular type.
- Services may make use of other services (proxies).
- Services can be composed to form composite services.
- Services may be charged.
- These enable more complex examples such as the outsourcing of a whole business application to several providers.



Grid Computing

- Grid Computing is not (just) about creating supercomputers by connecting lots of small machines.
- It is about sharing resources:
 - A piece of scientific equipment.
 - A database of information.
 - A medical scanner.
 - Compute power.
- Increasing network bandwidth makes sharing more economic.
- "Federation" of resources is the key:
 - The resources are owned by different people.

Resource Sharing

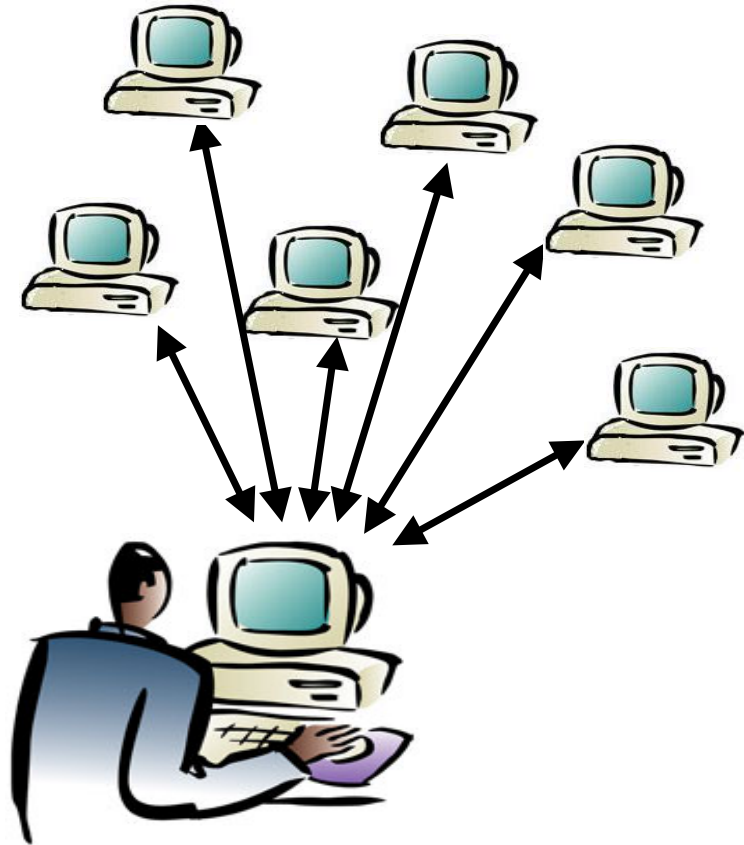
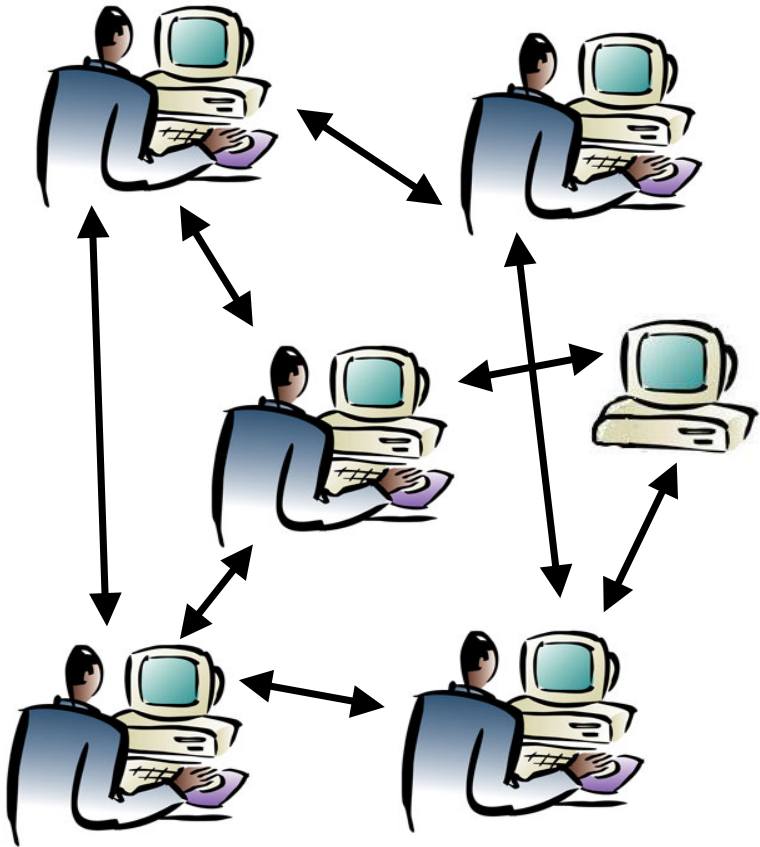
- Why share resources?
 - Why rent rather than buy?
 - You want a big resource for a short time.
 - You don't want the hassle of managing it.
 - Compute resources don't keep!
- Examples
 - A medical scanner might use very large amounts of compute power off the Grid for a short time to enable 3-D images to be rendered in realtime. Maybe the results are stored in a remote Grid database where they can be shared by other users.
 - A genetics program which requires a lot of compute power might make use of workstations in a student laboratory during the night.

Grid Technologies

- The "Globus Toolkit" is the original Grid software:
 - Very high-speed file transfer.
 - Support for delegated authentication (proxies).
 - Service advertisement.
 - Others ...
- Grid and Web Service technology are converging:
 - OGSA - Open Grid Services Architecture.
 - Takes advantage of existing Web infrastructure.
- "Utility Computing"
 - Data centres provide various resources which can be purchased as required.

Peer-to-Peer

- Any machine may act as a client or a server.



- A traditional "client/server" model makes a distinction between:
 - "Servers" which provide services.
 - "Clients" which make use of those services.

Peer-to-Peer Applications

- For example: Napster
 - Central index, P2P file sharing.
 - Newer filesharing applications have no central server at all (e.g.. Gnutella).
- Motivation
 - Robustness (against censorship).
 - Resource sharing (disk space).
 - Simple scalability (just add more nodes).
- Current Research Interest
 - Files systems.
 - Multiple redundant copies, encrypted locations.
 - Backups.

Common Requirements

- Good connectivity is assumed.
- Open Protocols.
 - Are necessary to enable different implementations to communicate.
- Cryptography is heavily used:
 - For authentication.
 - Prove who you are.
 - For confidentiality.
 - Keep my data secret.
 - Integrity.
 - Don't mess with my data.
 - For integrity.

Implications

- Applications are composed from services supplied by multiple “untrusted” providers.
 - Users have freedom to choose service providers and to automatically select these according to their own criteria.
 - This introduces competition.
- Users are often both providers and consumers of some service.
 - The distinction between service provider and consumer blurs.
- Some Benefits
 - Scaling, Robustness, Cost effective?
- Requires a different organizational approach.

Some Random Stargazing

- No more file servers.
 - Files are distributed across the spare disk space on all the desktops.
- Renting spare cycles.
 - At night, the compute power of all the local desktops is rented to a local firm for their big computing jobs.
- Printing.
 - A student wanting to print a thesis can locate the cheapest printer - or the nearest, or the fastest.
 - This could include University (central or school) printers as well as services supplied by local "copy shops".

Some More Stargazing

- Modular MIS services.
 - MIS supply modular services that can be composed and combined in different ways with school-specific, or external modules.
 - Student data?
 - Payment services?
- Less local software.
 - Services may be rented more, rather than buying and installing local software which is used only occasionally.
 - E.g.. Language translation integrated with a local wordprocessor.

Some References

- Peer-to-Peer : Harnessing the Power of Disruptive Technologies
 - Andy Oram, O'Reilly & Associates
- The Anatomy of the Grid
 - <http://www.globus.org/research/papers/anatomy.pdf>
- The Grid: Blueprint for a New Computing Infrastructure
 - Ian Foster, Carl Kesselman, Morgan Kaufmann
- Web Services
 - <http://www.w3.org/2002/ws/>
- Distributed Applications with XML-RPC, SOAP, UDDI & WSDL
 - Ethan Cerami, O'Reilly & Associates