UNIVERSITY OF SOUTHAMPTON

DEPARTMENT OF ELECTRONICS AND COMPUTER SCIENCE

AN ARCHITECTURE TO SUPPORT AN OPEN DISTRIBUTED HYPERMEDIA SYSTEM

Stuart Goose, Jonathan Dale, Gary J. Hill,

David C. DeRoure and Wendy Hall

University of Southampton

ECS Research Journal November 1995

An Architecture to Support an Open Distributed Hypermedia System

Stuart Goose, Jonathan Dale, Gary J. Hill, David C. DeRoure and Wendy Hall

Multimedia Research Group

Department of Electronics and Computer Science

University of Southampton

UK

{sg93r, jd94r, gjh, dder, wh}@ecs.soton.ac.uk

1. Introduction

It is now widely accepted that hypertext functionality can perform a useful role as an underlying component, or "link service" of an information system. This type of system, which is closer to the original perception of hypertext envisioned by Nelson [5], has been described by various authors [6,1], and the applicability of such an approach illustrated [4].

The majority of hypermedia systems have failed to address the needs of information sharing and interchange on a global scale. As a consequence, users of current systems have been conditioned to operate in isolation, thus creating an archipelago of information islands.

Contemporary systems, such as the World Wide Web (WWW), have illustrated how crucial global access to information is to the Internet community. For many, the WWW also served as an introduction to the concept of hypermedia; demonstrating its potential in providing an integrated and unifying paradigm for information organisation. As demands upon the WWW grew, it became apparent that the simplicities in its hypertext model were proving inadequate as the requirements moved from pure information access to incorporate information management, discovery and navigation. The WWW community are now looking towards open hypermedia strategies to help solve these problems.

In general, Microcosm [2] meets the requirements of an open hypermedia system, but is weak in its support for distributed operation, although Hill [3] describes initial experimentation with a networked version of Microcosm. This article outlines significant advances on Hill's contribution through the provision of a heterogeneous and distributed framework, above which a new open hypermedia system has been layered. The new system retains the Microcosm philosophy but offers greater scaleability, improved process management and more efficient communication. We anticipate that users will benefit from the superior model for open hypermedia that this widely distributed version of Microcosm provides.

2. A Distributed Hypermedia Model

As has been mentioned previously, one of the major limitations of the WWW is the simplicity of its hypertext model. To take into account distributed information exchange and management, a new hypermedia model is required to encompass the additional functionality. This model must allow users to share data easily, whilst maintaining the structure of the information in such a way that different interpretations can be imposed upon it in different contexts.

To facilitate such a model, the concept of a *hypermedia application* (figure 1) has been introduced. This provides mechanisms for encapsulating closely related documents and limiting the scope of a resource base to a defined and manageable entity. A hypermedia application, then, is the binding of a suite of processes (with their associated configuration), a collection of documents and an arbitrary amount of link data. Due to their modular nature, applications exhibit the properties of transitivity and may, in turn, comprise many other applications.

A *session* is a collection of hypermedia applications. By allowing users to build multiple sessions, they can import hypermedia applications from remote and disparate sources to form larger logical

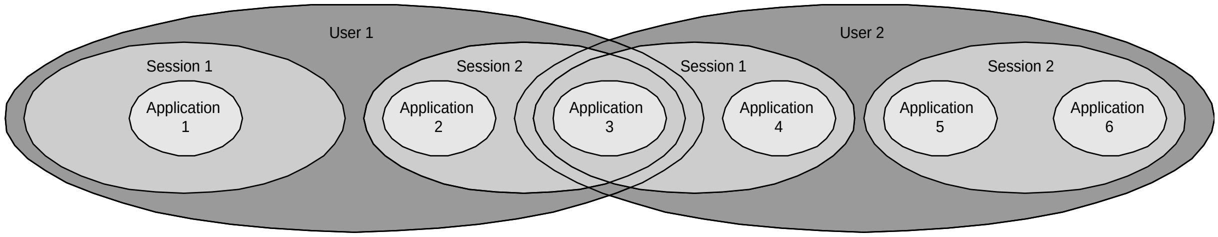

Figure 1 : Application Encapsulation and Information Exchange between Users

information spaces that are possibly related to their interests or work areas.

Additionally, to accommodate user-level access control, an application has ownership and a set of access permissions. Application owners can publish them, allowing other users to access them according to their permission fields. The binding between applications and their sessions are as loose or as tight as the permissions fields specify.

3. Distributed Communication

Wilkins and Heath [7] propose a scaleable message passing solution, called the Direct Communication Model (DCM), above which Microcosm could be layered. It provides several mechanisms by which peer entities executing on a single machine may communicate with one another. The new model adapts the DCM and enhances it to function effectively within a distributed heterogeneous multi-user environment.

For direct communication to take place one process must be able to uniquely identify or address the process it wishes to communicate with. This rudimentary requirement gave rise to a novel process addressing scheme using a familiar metaphor borrowed from computer filing systems.

The process addressing scheme forms the foundation upon which the support for message passing between processes executing on different machines is based. The first major benefit of the addressing scheme is that it can be customised by the architects of the host system to reflect the context in which it operates. This means that a process can send a message identifying the intended recipients using terms that have meaning within that context. The second key benefit is the wildcard attribute, which aids a process in disseminating a message to a wider audience. By placing a wildcard in one or more positions within the destination process address, a message can be targeted at a specific community of processes.

For example, one host system may require all process registrations to conform to the following custom addressing template:

/ProcName/UniqueProcId/Document/Service

Using this template, two link database processes could each register two service entries in the following style:

/linkbase/152.78.64.64.13/lion.raw/follow.link
/linkbase/152.78.64.64.13/lion.raw/create.link
/linkbase/152.78.64.48.11/tiger.raw/follow.link
/linkbase/152.78.64.48.11/tiger.raw/create.link

A viewer process could then direct a message to all link database processes that service follow link requests by quoting the following destination process address, where asterisk represents a wildcard:

/linkbase/*/*/follow.link

4. Architecture of the Heterogeneous Communication Model

The *router* (figure 2) acts as a bureau where processes, using the addressing scheme as the vehicle, can dynamically advertise and withdraw their services and also post messages to other registered service providers. A *process manager* is also started with each session to govern load balancing across the network and the distributed invocation of processes. Via its user interface, processes can be remotely configured and managed.

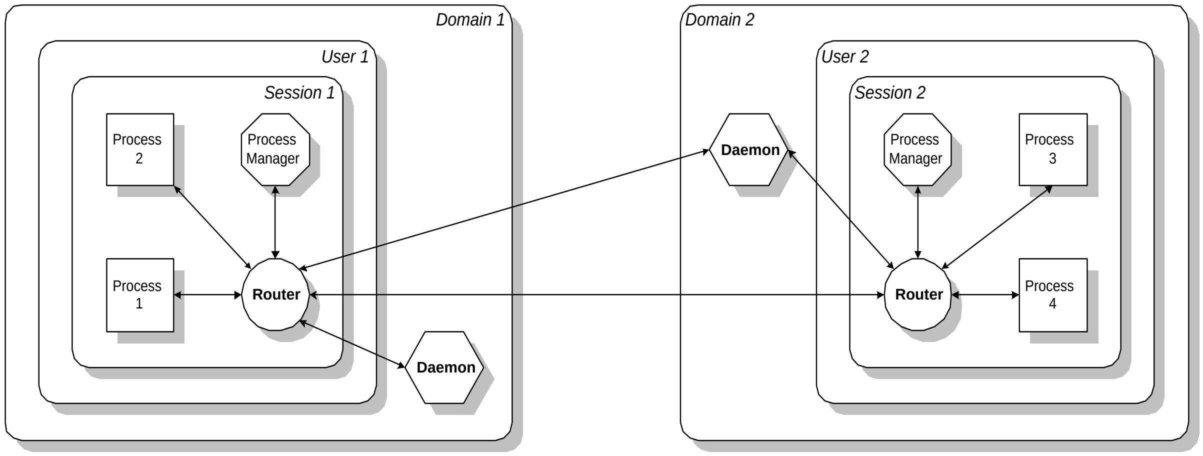

Figure 2 : Widely Distributed Multi-User Environment

For the router and participating processes to dynamically configure their network connections, a supporting piece of infrastructure proved necessary. A single *daemon* process to serve the local network domain[1], providing the sole fixed point of contact, helped achieve this.

A crucial characteristic of a scaleable architecture is that the addition of processes to the system has minimal impact upon efficiency. Another vital scaleability issue specific to the communication architecture is that the number of intermediate steps travelled by a message remains constant regardless of the number of processes present within the system. Both of these properties are exhibited by the HCM; the latter can be appreciated in figure 2, as all of the processes satellite the router, resulting in a message only ever travelling two steps to its destination.

Interactions between user sessions are also illustrated in figure 2. Here, user 1 in domain 1 initiates an action causing the daemon of domain 2 to be contacted, requesting a connection between their router and the router belonging to user 2. This currently means that the daemon in domain 1 is aware of the presence of the daemon in domain 2. Once a virtual connection has been established, negotiation between the two routers occurs. The process registration entries declared as *published* (that is, services that are available to other users) are then exchanged between the two routers. This activity alleviates superfluous message passing between routers. The scaleable framework for providing widely distributed, peer-to-peer communication can now be exploited.

5. Microcosm: The Next Generation Prototype

The Microcosm TNG prototype developed using the HCM has taken the core components of Microcosm and modelled them as discrete, distributed processes. These processes (linkbases, viewers, available links, linkers, docuverses, etc.) are described in terms of the messages that they send and receive and the processing that they perform. Processes are considered to be peer entities due to the fact that they can communicate with other processes asynchronously. The distribution of both processes and messages is handled by the HCM subsystem. In this way, the naming of a process is kept independent to the location of the process.

Resources are distributed through the application entity to which they belong. Once a user has authored an application, they can publish it to the world at large. The Microcosm TNG system provides a mechanism for allowing remote applications to be transparently included within a user's session. Also, since an application can comprise both data and processes, this inclusion not only increases the functionality of the user's session, but also increases their available dataspace; all actions (for example, link queries) will be automatically forwarded to all of the user's local *and* remote applications.

These concepts are illustrated in figure 3. In this diagram the two users, Jonathan and Stuart, both have sessions running in their respective domains. In a strict local session, all link queries and link creations only apply to the applications that are bound to a particular session (for Jonathan, this is the "Tigers"

---

[1] A domain is an administrative term that is used to describe a set of logical machines where HCM processes can execute.

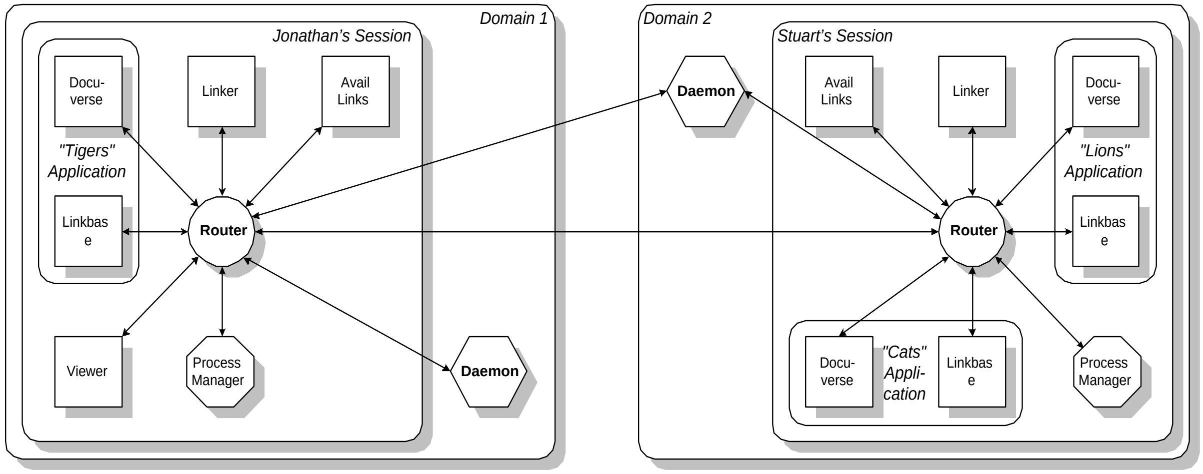

Figure 3 : Instances of the Microcosm TNG Prototype Communicating across a Widely Distributed Environment

application). Jonathan and Stuart can create their applications in isolation, importing documents and media, and making links between them.

However, imagine that Stuart wishes to create a new application, called "Cats", which is composed of both the "Tigers" and "Lions" applications. This new application exists in his local session and he makes a connection to the "Tigers" application through Jonathan's domain address (which he obtained previously through an agent, for example). Once connected, Stuart has access to both applications and he can create links between them and import new documents as appropriate.

Upon completion, the "Cats" application can then be published for other people to connect to and used in a similar fashion as indicated previously. It is important to note that when users access this application, connection to the "Tigers" and "Lions" applications and hypertext links are handled transparently.

This prototype has demonstrated that distributing an open hypertext system above the HCM shows great promise for the future. The implementation has demonstrated wide-scale distribution of users, data and processing, open hypertext functionality and integration capabilities for co-operative working environments. The next phase of development will reinforce and extend the system described here to produce an industrial-strength, distributed open hypertext system.

6. Future Work

The HCM was designed to accommodate collaboration and as such broadcasts notification events upon users beginning and terminating sessions. This information was subsequently used to build an awareness utility, showing users within the domain. But current research examining the potential of Microcosm to support advanced CSCW features will undoubtedly impact upon the future development of the Microcosm TNG system.

When disclosing resources for remote users to peruse, security becomes a genuine consideration. Authentication mechanisms are also required if users are to be charged for the time/resources they use. Preliminary work in this area has been conducted but further effort is required to develop these ideas.

Wilkins et al. [8] discuss how communities of co-operating intelligent agents can greatly assist with a variety of tasks within Microcosm. The role of agents within an open hypertext system can be sub-divided into three categories:

- Resource location and discovery.
- Maintaining information integrity.
- Navigation assistance.

It has been recognised that the HCM and Microcosm TNG provide an ideal framework for further experimentation with agent technology and the Internet.

7. Conclusions

The client/server model, the predominant architecture among systems using the Internet, can restrict the development of truly distributed systems. This paradigm makes it very difficult for current information services to take the initiative in delivering fresh information to the user. Furthermore, some processes need to act as both client and server depending upon whom they are interacting with. The open model of the HCM allows systems to benefit from a more flexible peer-to-peer based alternative.

The rapid development of the HCM-based hypertext system described in this paper illustrates the ease with which the HCM allows distributed systems to be built, and as such enables Microcosm TNG to provide a degree of flexibility not found in other distributed hypertext systems. As one would expect, collaboration between users is promoted together with strong support for the sharing of information. In particular, the ability to encapsulate a group of distributed processes with their respective resources and publish them as a single entity provides a welcome degree of abstraction. The hypertext application conceals configuration details and provides a scaleable mechanism for building and composing new applications.

References

[1] DAVIS, H. C., HALL, W., HEATH, I., HILL, G. J. and WILKINS, R. J., Towards an Integrated Information Environment with Open Hypermedia Systems. *In:* Lucarella, D., Nanard, J., Nanard, M. and Paolini, P., *Eds.*, ECHT '92, Proceedings of the Fourth ACM Conference on Hypertext, Milan, Italy (November), ACM Press, pages 181-190, 1992.

[2] HILL, G. J., WILKINS, R. J. and HALL, W., Open and Reconfigurable Hypermedia Systems: A Filter Base Model, Hypermedia, 5(2), pages 103-118, 1993.

[3] HILL, G. J. and HALL, W., Extending the Microcosm Model to a Distributed Environment. *In:* ECHT '94 Proceedings, Edinburgh, Scotland (September), ACM Press, pages 32-40, 1994.

[4] MALCOLM, K. C., POLTROCK, S. E. and SCHULER, D., Industrial Strength Hypermedia: Requirements for a Large Engineering Enterprise. *In:* Hypertext '91, Proceedings of Third ACM Conference on Hypertext, San Antonio, Texas, ACM Press, pages 13-25, 1991.

[5] NELSON, T., Literary Machines 87.1, published by the author, Mindful Press, 1987.

[6] PEARL, A., Sun's Link Service: A Protocol for Open Linking. *In:* Hypertext '89 Proceedings (November), Pittsburgh, USA, pages 137-146, 1989.

[7] WILKINS, R. J., HEATH, I., and HALL, W., A Direct Communication Model for Process Management in an Open Hypermedia System., CSTR 93-14, University of Southampton, UK, 1993.

[8] WILKINS, R. J., DeROURE, D. C., HALL, W. and DAVIS, H. C., The Role of Agents in Multimedia Information Systems. *In:* Proceedings of the Intelligent Agents and the Next Information Revolution, Manchester, UK (May), pages 14-23, 1995.