Building Scalable Ad Hoc Collaboration Networks

Krishna Kant, Vijay Tewari, Ravi Iyer

Intel Corporation, OR

1 General Information

Presenter contact information: {krishna.kant | vijay.tewari | ravishankar.iyer}@intel.com

Aims/Learning objectives:

1. Gain knowledge about the emerging ad-hoc computing paradigm
2. Understand the major research issues in ad-hoc computing
3. Understand performance and architectural issues for ad-hoc computing.

Duration: Targeted for 3 hours. Can be shortened somewhat as necessary.

Keywords: ad-hoc computing, wireless networks, peer-to-peer systems, file sharing, collaboration, intelligent searching, performance analysis

Target audience: Computer scientists interested in gaining a basic understanding of scalability, content location and performance of ad-hoc networks.

Prerequisite knowledge of audience: General computer science background with some appreciation of distributed systems.

Tutorial history: This tutorial is related to our earlier tutorial on peer-to-peer (P2P) computing given at Tools 2002 conference in London in April 2002. That tutorial concentrated on performance issues of P2P networks on the basis of a tool we had developed. An earlier version of the tutorial was intended for ICNP-2001 conference and included a significant amount of material on P2P applications, middleware and taxonomy.

2 Extended Abstract

In the last several years, we have seen a significant amount of work on ad-hoc and peer-to-peer networks. The use of ad-hoc networks occur in a variety of scenarios, but have gained prominence primarily in communications between mobile devices. These include building sensor networks (to monitor activities of various sorts), military communications, vehicular networks, epidemic communication networks [16], and collaboration between mobile devices. The explosive and continuing growth of 802.11 based wireless networks provides the most current impetus for understanding ad-hoc collaboration networks because of its high practical utility. Ad-hoc networks have also been examined for wired (LAN/WAN) scenarios [4, 8, 14, 19] generally in the context of peer-to-peer (P2P) computing and more recently in the context of grid computing [18]. For example,

research in peer-to-peer (P2P) file-sharing technologies [4, 10] has contributed substantially to ad-hoc network research. This tutorial is intended to examine some of the more sophisticated emerging ad-hoc / P2P networks, their needs and their performance issues.

As the power requirements, cost and networking ability of mobile devices drop substantially and they span an extremely wide spectrum (from laptops all the way down to micro-particulate or paintable sensors), ad-hoc networks are expected to play a dominant role in almost every aspect of daily life (from microsensors "floating" in pipes to monitor chemical composition/reaction to collaborative human networks). The tutorial shall start with a brief introduction to mobile ad-hoc computing. The much studied issues of routing in ad-hoc networks and minimizing power consumption will be touched upon briefly. We will primarily discuss the topics and issues related to service architectures and collaboration mechanisms (service discovery protocols [9], proximity issues [3], usage models for collaboration [2], capability/heterogeneity issues, etc.).

The tutorial will then introduce peer-to-peer computing paradigm via a number of usage models including file-sharing (Gnutella/Freenet) [4], processor cycle sharing (seti@home), storage sharing [6], collaborative editing, telemedicine [1], video-conferencing [5], streaming media distribution, etc. The tutorial will also briefly introduce a number of projects that have attempted to establish the infrastructure necessary for arbitrary P2P computing on a WAN scale. This includes Globe and Legion WAN operating systems[8], the open-source Globus toolkit [7] intended for building computing "Grids", low-level features supported by Sun's JXTA^{TM} initiative, P2P support in Microsoft's .NET^{TM} initiative, etc. A serious obstacle to WAN-level ad-hoc computing is the frequent use of firewalls/NATs in both business and home environments. Several controlled access mechanisms (like the use of rendezvous points) have been proposed and will be discussed. A related problem is the lack of DNS entries for most of the peers that would be interested in participating in the ad-hoc network, which makes addressing them difficult. Security is also a crucial issue in WAN interaction, but traditional methods of generating and distributing encryption keys do not work very well in a fully distributed environment with mutually suspicious agents. The tutorial shall discuss these and several other unique issues essential for designing scalable, Internet wide ad-hoc networks.

Based on the various classes of ad-hoc / P2P environments discussed above, we identify a common set of requirements for collaborative ad-hoc computing applications. We will attempt to map these requirements on to some middleware services (categorized into basic and advanced services) that would help the development of technologies / applications for this paradigm. Our earlier work on the taxonomy of peer-to-peer applications [11] as it applies to various aspects of ad-hoc computing will also be briefly discussed. One of the important aspects in ad-hoc computing is the location of desired resources and services that will participate in the collaboration. When the network is not too large and the participating nodes and their resources/hosted services are fairly static, good solutions can be built using indexing structures or hashing combined with appropriate migration policies. For large networks, aggregation schemes such as Bloom filters have been proposed to reduce the amount of information to be communicated or stored. However, dealing with a large, highly dynamic network still remains an open problem. One extreme here is full search (as in Gnutella [10]), although this can be augmented with techniques such as local caching of content and/or its location, fuzzy matching, and automatic migration of content (as in Freenet [4]). The cost of searches can be further reduced by using limited forms of indexing mechanisms such as "distributed hash tables" [20], although the flexibility appears to suffer significantly. The tutorial shall discuss these schemes as they apply to the various environments presented in the taxonomy and also present some possible extensions.

The last part of the tutorial shall briefly discuss the performance issues related to a dynamic ad-hoc network. It will review some of our previous work on the topic including an analytic random graph model [13] and also describe a simulation tool [12] that we have designed. A number of results based on the simulation tool will be discussed for both small and large collaboration networks. It will also discuss the need for a number of architectural features that could enhance performance in a highly dynamic collaboration environment.

3 Speaker Biographies

Krishna Kant received his Ph.D. degree in Computer Science from The University of Texas at Dallas in 1981. From 1981-1984 he was an assistant professor of computer science at Northwestern University, Evanston, IL. From 1985-1989, he was an assistant professor, and from 1989-1991 an associate professor of computer science at the Pennsylvania State University, University Park, PA. In 1988 he served with the Teletraffic Theory division at AT&T Bell Labs, Holmdel, NJ and in 1991 with the Integrated Systems Architecture division at BellCore, Piscataway, NJ. From 1992-1997, he was with Network Services Performance and Control group at Bellcore, Red Bank, NJ where he worked on a variety of narrowband and broadband signalling performance and congestion control issues. Since May 1997 he has been with the Server Architecture Lab at Intel Corp, where he works on performance issues for Internet servers. His current research interests are in the areas of traffic characterization, performance modeling of Internet servers, and peer to peer computing [11, 12, 13]. He organized a Panel session on P2P computing at ICNP-2001 and gave a tutorial on P2P performance at Tools 2002 conference. He is the author of a book titled *Introduction to Computer System Performance Evaluation* (New York: McGraw-Hill, 1992).

Vijay Tewari is currently a Senior Software Engineer with the Systems Software Lab in Intel Corporation. Previously, he was a member of the Peer-to-Peer architecture team working on issues related to middleware services for P2P computing. He co-authored a few papers on peer-to-peer computing and also co-developed a tutorial on peer-to-peer computing for the Tools 2002 conference held in London.

Vijay got his Masters in Computer Science from the University of Minnesota, Twin Cities. During his Masters Vijay worked as an Intern in the Enterprise Architecture Labs for Intel corp. working on Web server performance. His interests include the areas of Networking, Internet and Distributed Computing. He has ten years of experience handling large communication systems for the Armed Forces in India. These include both static and mobile systems including wireless solutions.

Ravi Iyer is currently a performance analyst in the Enterprise Architecture Laboratory at Intel Corporation. He is currently working on characterizing internet server workloads and on architectural / performance issues for front-end and back-end servers. His reasearch interests include peer-to-peer computing [11, 12] where he codeveloped a tool for P2P performance modeling and co-authored a tutorial on the subject for the Tools 2002 conference held in London. Previously, he worked for the Computer Systems Laboratory at Hewlett Packard Laboratory and in the Server Engineering Group at Intel Corporation. He received his PhD in Computer Science from Texas

A&M University in 1999. His research interests include internet computing / protocols, computer architecture, parallel processing and performance evaluation. He is currently serving as a member of the program committee for the Workshop on Workload Characterization (WWC-2002). He is a member of the IEEE.

References

[1] T. Bui and S. Sankaran, "Group Decision and Negotiation in Telemedicine: An application of Intelligent mobile agents as nonhuman teleworkers", Proc. of 30th Hawaii Intl. conference on system sciences, Vol 4, 1997, pp120-129.

[2] Bjrk, S. "Designing Mobile Ad Hoc Collaborative Applications: Scenario experiences with Smart-Its," Mobile Ad Hoc Collaboration Workshop at CHI 2002.

[3] M. Castro, P. Druschel, et. al., "Exploiting Network Proximity in Distributed Hash Tables".

[4] I. Clarke, "A Distributed Decentralized Information Storage and Retrieval system." M.S. Thesis, Division of Informatics, Univ of Edinburgh, UK, 1999.

[5] S.T. Chanson, A. Hui and E. Siu, "OCTOPUS – A scalable global multiparty video conferencing system", Proc. of 8th Intel. conference on computer communications and networks, 1999, pp97-102.

[6] P. Druschel and A. Rowstron, "PAST: A large-scale, persistent peer-to-peer storage utility", HotOS VIII, Schoss Elmau, Germany, May 2001.

[7] I. Foster, C. Kesselman, J.M. Nick and S. Tuecke, http://www.globus.org/research/papers/ogsa.pdf

[8] A. Grimshaw, et al., "Wide-Area Computing: Resource sharing on a large scale", IEEE Computer, May 1999, pp1-9.

[9] T. D. Hodes, R. H. Katz, E. Servan-Schreiber and L. A. Rowe,"Composable Ad-hoc Mobile Services for Universal Interaction," Proceedings of The Third ACM/IEEE International Conference on Mobile Computing, Budapest, Sept. 1997.

[10] M.A. Jovanovic, F.S. Annexstein and K.A. Berman, "Scalability issues in large peer to peer networks – A case study of Gnutella", Tech. Report, Univ. of Cincinati, 2001.

[11] K. Kant, R. Iyer and V. Tewari, "On the Potential of Peer-to-Peer Computing: Classification and Evaluation", Proc. of CCGrid 2002, Berlin, Germany, available at http://kkant.ccwebhost.com/download.html

[12] K. Kant and R. Iyer, "A tool for performance evaluation of peer-to-peer file-sharing systems", submitted for publication, available at `kkant.ccwebhost.com/download.html`.

[13] K. Kant, "An Analytic Model for Peer to Peer File Sharing Networks", accepted for ICC 2003, (available at http://kkant.ccwebhost.com/download.html).

[14] J. Kubiatowicz, D. Bindel et al., "OceanStore: An architecture for global-scale persistent storage", Proc. of the 9th Intl. conf. on arch. support prog. languages and OS (ASPLOS 2000) (Boston, MA, November 2000), pp. 190.

[15] R. Kumar, P. Raghvan, et. al., "The web as a graph", ACM POD conference, Dallas, TX, 2000.

[16] C. Moore and M. Newman, "Epidemics and percolation in small-world networks".

[17] G. Pandurangan, P. Raghvan and E. Upfal, "Building low-diameter P2P networks".

[18] T. Phan, L. Huang, C. Dulan, "Integrating mobile devices into the computational grid", Proc. of 8th Intl conf. on Mobile computing and networking, 2002 , Atlanta, Georgia, USA.

[19] M. Ripeanu, "Peer-to-Peer Architecture Case Study: Gnutella", Proc. of 2001 Intl Conf on Peer-to-peer Computing (P2P2001), Linkoping Sweden, 27-29 August 2001.

[20] A. Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems". IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Heidelberg, Germany, pages 329-350, November, 2001.