Configuration Challenges for Smart Spaces

John Heidemann, Ramesh Govindan, Deborah Estrin

University of Southern California/Information Sciences Institute

USC Technical Report 98-677
July 15, 1998

ABSTRACT

Smart spaces will be composed of thousands of computing elements interacting with the physical world. We argue that automatic configuration and resource discovery remains an important unsolved problem in smart-space deployment.

1. INTRODUCTION

Trends in CPU performance and packaging now allow very powerful devices to be embedded on single chips. Today a car may have a half-dozen processors on-board that are monitoring brakes, improving fuel efficiency, and even regulating climate. We are poised to live in a world of *smart spaces* where most things around us (from bandages to books) have computing power and interact with the world.

Until recently smart spaces have been limited by computing power, price, and energy. Communication between nodes has been extremely limited. Moore's law has attacked computing power and price, a variety of initiatives are attacking energy. Finally, the growth of wireless networking hardware and explosion of the Internet promises COTS networking hardware and protocols. Or do they?

Although many challenging impediments to smart spaces have been overcome, we believe direct application of COTS networking protocols will not prove satisfactory. Today's networking infrastructure has many features important in connecting smart spaces: Internet protocols and implementations are open, widely implemented, support multiple physical layers, and are relatively efficient and lightweight. All of these characteristics are important for smart spaces.

In spite of these advantages, existing networking approaches fail to address the defining feature of smart spaces: smart spaces have *hundreds of heterogeneous, networked computers per person*. Smart spaces have hundreds of computers per person because each piece of equipment will be tagged, each object visible output will provide sensor data to the network, and each object with controls will be manipulatable from the network.

Too frequently, existing protocols require manual intervention to configure and to use. Smarts spaces will have hundreds of often heterogeneous objects which frequently change configuration (as they move, are deployed, and wear out) and may be physically inaccessible. The implication of hundreds of nodes per person under these conditions is that failure to autoconfigure must be considered complete failure; failure of self-organization must be considered undeployability, and failure of automatic resource discovery must be considered effective inaccessibility. A corollary of large numbers of nodes is that smart spaces must employ algorithms which function in the face of partial operation and heterogeneity.

Although we believe that coping with large numbers of nodes is the fundamental challenge behind smart spaces, two other issues are also important. As "spaces" suggests, smart spaces will be part of the physical world. This implies that they must understand their place (physical location and electronic neighborhood) and be able to influence other objects. Also smart-space nodes will often be resource limited because of constraints of size, weight, mobility, or cost.

The remainder of this paper explores these issues further. We examine the implications of large numbers of nodes for networking protocols, application design and operation, and security.

2. CHALLENGES IN NETWORK CONFIGURATION

If the fundamental change of smart spaces is a hundredfold increase in the number of networked computers per person, then this change is sure to have effects up and down the network stack.

At the physical layer, many nodes imply a need for very flexible network connectivity. Wireless technologies are the obvious choice here. Substantial progress has been made both in COTS wireless (for example standardization and widespread deployment of IEEE-802.11 [5]) and in the research community (for example, DARPA's GloMo efforts). We believe that flexible wired connections are also important. Wired connections can provide both high-speed connectivity and (if accompanied by power) can eliminate energy constraints.

At the network layer fully automated Internet address configuration is a requirement. IPv6 holds some promise here with a carefully considered autoconfiguration [11], but it assumes that failures can be manually resolved, it bases configuration on globally unique link-layer addresses, and its protocols require a very large address space. Approaches to relax these constraints are important if smart spaces are to apply to a range of link layers and to interoperate with IPv4 systems.

At the transport layer, smart spaces suggest wide use of multicast protocols [1, 6]. Multicast protocols allow groups of devices or a user and a group of sensors to interact efficiently. Multicast group addresses decouple the data sender and list of recipients. With thousands of nodes where weak connections and node failure may be common, flexible group membership is critical. Finally, announce/listen-style multicast-based protocols and soft-state simplify failure recovery.

Protocols which self-adapt to congestion are vital for smart spaces.

Floyd argues that it is the use of end-to-end congestion control and particularly TCP's additive-increase/multiplicative-decrease algorithms that have allowed the Internet to cope with orders of magnitude in growth [2]. If smart spaces are to adapt from the very high node densities of a conference room to a relatively sparse city street, similar protocols will be required.

3. CHALLENGES IN APPLICATION CONFIGURATION

Successful network autoconfiguration will allow nodes in smart spaces to talk with each other, and will allow them to do so without overwhelming the network. In addition, application-level approaches are required to make sense out of this communication.

Since smart spaces involve the physical world, absolute or relative physical location is an important concept. Node locations must be automatically configured; manual configuration will be both inaccurate and time consuming. GPS is an obvious candidate for location determination, but several factors argue the need of additional approaches: GPS accuracy is limited, it requires an antenna which may be too large for some nodes, reception is limited indoors, and cost remains a factor. Supplemental protocols are important to offset these problems. Accuracy of centimeters or tens of centimeters is important to place a node on one side of a wall or the other. The use of alternate protocols may be desired in buildings [14]. Finally, a location-equivalent of the Network Time Protocol [8] is needed to allow nodes to benefit from nearby GPS receivers.

Although physical location is important, we have argued elsewhere that *logical* location is often more important to to real world applications [4]. (After all, do you care that you're at latitude 33.97988N, longitude 118.43994W, or that you're at work and that you're in your office?) Improving mappings between raw physical location and logical location are key to effective interaction between smart spaces and the environment.

Resource rendezvous is the process of matching clients and servers based on physical location and other attributes. Automated rendezvous allows applications to share data at high-levels without user involvement. Automated rendezvous includes both yellow-pages services (this light switch controls the northwest bay of lights) and more complex attribute-based queries (this light switch requests 25% illumination around the podium). Traditional approaches such as broadcast and expanding-ring search apply to resource discovery in smart spaces, but new opportunities exist to take advantage of physical location and device-to-device communication.

In addition to low-level congestion control, coordination at the application level is important to controlling network usage. We expect that devices in smart spaces will self-organize into *functional clusters*. Each cluster will performing a higher-level coordinated action. For example, devices attached to individual lighting elements of a light panel may coordinate to dynamically vary their overall intensity based on environmental factors. Self-organization protocols build on the basic announce/listen protocols; announcements allow frequent cluster self-organization and reorganization. While existing clustering protocols [3] can be adapted to smart spaces, two features distinguish clustering in in this environment:

- Clusters for smart spaces frequently overlap. Functionally related devices (e.g., lighting elements) may not be topologically contiguous, and different uses may overlap (full room lighting vs. podium lighting). Classical clustering protocols usually focus on the formation of disjoint clusters.
- Some smart-space devices may be power-constrained. Moreover, these power-constrained devices may communicate with tethered devices that are not subject to such constraints. Clustering protocols will need to be aware of energy constraints, where possible, in order to limit communication.

A final implication of large numbers of devices is application heterogeneity. Smart spaces will be composed of multiple generations of smart nodes; applications must cope with many different protocol versions. Two very different approaches address this problem. On one hand we can construct nodes to be field-upgradable. Active Networks initiatives offer promise here [10]. An alternate approach is based on the principle of delay and discard. Nodes interact with very flexible protocols (for example, information buses [9]), delaying the need for frequent change. Nodes are designed to be cheap enough that they can simply be discarded when they no longer interoperate.

4. SECURITY CHALLENGES

Just as configuration must scale to support hundreds of nodes per person, so must security protocols. Existing protocols for key distribution are important, but more understanding is needed for ways to get good security for thousands of nodes instead of perfect security for tens of nodes. With many nodes, some will be compromised, so approaches to identify, isolate, and respond these nodes during continued operation are important [7, 13]. Finally, for very small nodes, new lightweight security protocols may be of increasing importance [12].

5. CONCLUSIONS

In this paper we have argued that the key problem facing use of smart spaces are the implications of configuring and using hundreds of computers per person. We have examined these issues for network communications, application interaction, and security implications. Addressing these problems are important if deployment of smart spaces is to become feasible.

Acknowledgements

The authors would like to thank the members of the Simple Systems ISAT study (chaired by Deborah Estrin) for their discussions on issues related to smart spaces.

This paper originally appeared in the DARPA/NIST Smart Spaces Workshop, July 30-31, 1998, Gaithersburg, Maryland, USA.

References

1. Stephen E. Deering and David R. Cheriton. Multicast routing in datagram internetworks and extended lans. *ACM Transactions on Computer Systems*, 8(2):35–110, May 1990.
2. Sally Floyd and Kevin Fall. Promoting the use of end-to-end congestion control in the internet. Sumibtted to IEEE/ACM ToN, February 1998.
3. Ramesh Govindan, Cengiz Alaettinoğlu, and Deborah Estrin. Self-configuring active network monitoring (SCAN). White paper, February 1997.
4. John Heidemann and Dhaval Shah. Experiences with user-configurable, location-aware scheduling. Technical Report 98-675, University of Southern California, April 1998. submitted for publication.

5. IEEE. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (IEEE 802.11)*. IEEE, 1997.

6. Van Jacobson. Multimedia conferencing on the internet. Tutorial at SIGCOMM '94; published as a special issue of the SIGCOMM Newsletter, August 1994.

7. Leslie Lamport, Robert Shostak, and Marshall Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982.

8. Dave L. Mills. Network time protocol (version 2) specification and implementation. RFC 1119, Internet Request For Comments, September 1989.

9. Brian Oki, Manfred Pfluegl, Alex Siegel, and Dale Skeen. The information bus—an architecture for extensible distributed systems. In *Proceedings of the 14th Symposium on Operating Systems Principles*, pages 58–68, Asheville, North Carolina, USC, December 1993. ACM.

10. David L. Tennenhouse, Jonathan M. Smith, W. David Sincoskie, David J. Wetherall, and Gary J. Minden. A survey of active network research. *IEEE Communications Magazine*, 35(1):80–86, January 1997.

11. S. Thomson and T. Narten. Ipv6 stateless address autoconfiguration. RFC 1971, Internet Request For Comments, August 1996.

12. Gene Tsudik and Brian Tung. On constructing optimal one-time signatures. Submitted for publication, November 1997.

13. Brian Tung. Crisis: Critial resoruce allocation and intrusion response for survivable information systems. http://gost.isi.edu/projects/crisis/, January 1998.

14. Andy Ward, Alan Jones, and Andy Hopper. A new location technique for the active office. *IEEE Personal Communications Magazine*, 4(5):8–15, October 1997.