

Major Findings Resulting from Project Activities

Through our work completed thus far, we have identified several points that we feel are important in general to the emerging area of ubiquitous computing.

- Security is very important and is a difficult problem for ubiquitous computing. Users are mobile and may require different security policies based on external factors, such as location. Therefore, security is context dependent.
- The dynamic nature of Active Spaces requires more research in toolkits, learning from the environment and past actions of users. Current context toolkits are a step in the right direction. However, they must incorporate more dynamic methods for determining context through the use of artificial intelligence.
- Active Spaces should be built from generic components that can be combined to build complex systems. Decoupling components and services (i.e., through an event service) enables services to be built independently. Standard protocols are equally important to allow different service implementations to communicate.
- Notifications are different from events. Our event service is really a notification system. Small delays are tolerable, as the data does not consist of continuous media. Events, such as mouse movements, need to be fast for real-time distribution.
- Scaling an event/notification service cannot be achieved through standard federation. A scalable and efficient event mechanism has similar characteristics to group multicast routing, which must only send events to registered listeners.
- It is feasible to build small object request brokers for small handheld devices that possess limited CPU and memory. We have used dynamic method invocation and have found the performance to be excellent.
- Remote administration of server machines is important, since there may be many machines to support a given space. The ability to remotely start components simplifies administration.
- Fine-grained location detection remains difficult. Although we have been typically working on the granularity of rooms, some applications may require accuracy to the foot. There are few solutions that can address this.
- CORBA works well for method invocations. However, it is less efficient for data transfer, due to memory copies. Application control or customization of ORB memory handling internals will help to make data transfer more efficient. More of the ORB internal structure, even to the operating system layer (i.e., socket buffer sizes), needs to be exposed to allow custom configurations.
- A scripting language that is able to communicate with services can greatly increase the speed of application construction and can facilitate administration in the Active Space environment.
- Although HDTV is visually far superior to other displays, commercial support for HDTV over IP is not mature (hardware and software).
- A new application model is required for ubiquitous applications. Applications running in an Active Space are much more dynamic and can make use of several input/output devices, which may not be known *a priori*. Standard MVC can be

extended to support distributed application construction. Adaptors are needed to support the wide array of devices and user preferences.

- Wireless TCP raises performance problems. Due to true losses in wireless transmissions, new protocols are required for TCP over wireless links to increase throughput.
- There is a core set of services that applications will require to run in a physical space. Although we feel we have identified several, more work is necessary in the application space to identify a more complete list.