

Application and Desktop Sharing

Omer Boyaci, Henning Schulzrinne

March 15, 2007

Application and Desktop Sharing (ADS) enables to share a single application or the whole desktop with remote users. ADS consists of a server and a number of clients. The shared application runs only on the server and the screen updates are transmitted to clients via bitmap images and some commands like MoveRect. The Clients are thin-clients and they do not need the application itself. They just display the screen image of the remote application and transmit keyboard and mouse actions back to the server.

Application sharing differs from desktop sharing because the server has to transmit only the shared application. Shared application can open other windows and these new windows (preferences, fonts, settings, new document, ...) have to be transmitted to the clients. ADS should be careful and not to transmit any other application's window which stays on top the shared application.

Currently ADS supports only Windows for the server side and any java supporting operating system for the clients. A unix server has written by Jonathan Lennox but needs some modification in order to support new protocol between clients and server. The Windows server has two parts

- Mirror Driver (in kernel space)
- Server Code (in user space)

Mirror Driver is the most efficient way to capture screen updates in Windows. It behaves like a regular video driver and gets all the screen update commands from the OS. The user space code and the mirror driver communicates via shared memory. Developing the mirror driver was the hardest part of the

project due to inadequate documentation and restrictions of the kernel space both in terms of libraries and software model.

ADS transmits screen updates as PNG images encapsulated in RTP packets. Framing RTP Packets over Connection-Oriented Transport is used for the TCP connections. Scrolls are transmitted as MoveRect commands in order to decrease bandwidth usage and delay. If the shared application window changes continuously (video playback, animation,...) ADS has to adjust its update generation speed according to the fastest client. For example, it can generate a frame in every 100ms or 500ms for a video playback sharing session. ADS uses a different algorithms for the slower clients. BFCP is used for the floor control. Client application captures KM events and transfers them to the server. At the server these events are executed and new screen images are transmitted back to all clients.

ADS has some extra features:

Recording The clients can record the sharing session and then a separate player can play these files locally or remotely.

Multicasting Both the player and the sharing server supports multicasting. Retransmission mechanisms are developed due to lossy nature of UDP .

Listening Client The client can wait for the incoming requests and the sharing server can connect to these clients. This system can be used as a RGB cable replacement during presentations where the participants stream their application images to the display computer.

Small PNG ADS can generate small PNG images (around 1.5KB) which consist of a few scanlines. This feature is very useful for the lossy UDP transmissions and also decrease the update delay in the client side.