A Thin Client Architecture for Mobile Users Running Multiple Applications

Pieter Simoens*, Lien Deboosere
Department of Information Technology (INTEC), Ghent University, IBBT
Pieter.Simoens@intec.ugent.be, Lien.Deboosere@intec.ugent.be
Supervisors: Filip De Turck, Bart Dhoedt, Piet Demeester

* Research Assistant for the Fund of Scientific Research (FWO)

1 Introduction

In the thin client computing paradigm, the applications of a user are executed on a network server, instead of locally on the user’s device. User events, like keystrokes and mouse movements, are sent over a network connection to the server, that performs all calculations and renders the corresponding graphical data. The client device only decodes this graphical data and presents it to the user. Since all hardware that is not related to user input and output can be removed from the device, thin clients are lightweight and energy efficient, which makes them very attractive for mobile users.

When a user strokes a key, this event is sent to the application server. The graphical results are sent back to the device for presentation. There is an inherent latency of at least one round-trip time before the effect of a user event can appear on the screen. When a user becomes mobile and moves further away from his server, the latency increases and the Quality of Experience (QoE) might degrade. Our research is oriented toward the deployment of a migration mechanism to move the application to a server nearer to the user.

In current thin client deployments, the rendering of graphical data and the actual application calculations happen at the same location. Since an emerging number of applications is available on the Internet and application servers have limited processing power, it has become impossible for one service provider to offer all applications. Our architecture can solve this scalability constraint by decoupling the graphical rendering and the actual application execution. We introduce the concept of a thin client provider, acting as a single portal for the user and responsible for the cooperation between applications of different providers. Also, specific communication protocols must be developed.

In this paper, we first present a general overview of our thin client architecture. We then detail our two main research challenges: the migration of applications between servers and the optimization of the communication protocols.

2 Architecture

An overview of our architecture is presented in Figure 1.

Our architecture is composed of two main parts: the access network and the core network. The access networks connects users, via public or private access points, to their thin client provider (TCP). The applications are executed in the core network, on the servers of the application providers (AP).

The TCP acts as a portal to a range of applications, ensures a correct inter-working between applications offered by different APs and forwards user events to the correct application server. Every user has his own personal ‘desk’ top and storage space on the TCP. In this way, every time a user logs in, he finds his desktop in exactly the same state as he had left it.
3 Application Migration for mobile users

When a user moves around, the latency between the device and the application server might become too high. In order to maintain a satisfying user experience, the application should make a parallel movement and migrate to a server near to the user. The application is stalled, all its state is saved and transferred to the new server, where the execution is continued. Application migrations are however costly operations in terms of network resource usage and application downtime. Our research is oriented toward the development of algorithms for the selection of the most appropriate server to migrate the application to, in order to minimize the number of migrations. In [1], we presented a theoretical model to find the most optimal locations to install application servers in the network. The follow-up paper [2] discussed the number of migrations and proposed three mechanisms to perform a seamless migration between application servers. Finally, in [3] several heuristics are presented that take the user mobility into account to minimize the number of migrations. All previous work was focused on migrations between application servers. Currently, we are researching the migration mechanism between thin client providers.

4 Multi-application computing

The thin client provider plays a central role in the interworking between applications of different APs. It merges the graphical output data into one output stream for the client. There are two options for the format of the graphical output. In most of thin client protocols, graphic commands, e.g. to draw a combo box or a button, are sent to the client, but also protocols with lower level operations exist. However, these protocols cannot meet the requirements for multimedia- and 3D-applications. In [4], we propose an alternative approach for this type of applications, whereby the graphical output is sent as a video stream to the thin client.

To optimize the network stream, monitor probes can be placed along the connection between client and TCP. For example, the parameters of the video-codec can be adapted to a less energy-consuming codec when the battery of the device is running low. Other optimizations can be done on the wireless interface. By splitting up the thin-client traffic in different priority-classes and sending the corresponding packets at different bitrates, energy can be saved in future wireless network cards. Low-priority traffic could also be delayed, to allow the network card to remain in sleep mode as long as possible, thus saving energy.

5 Conclusion

In this paper, we presented our research on the development of a thin client architecture for mobile users running multiple applications. The thin client provider acts as a portal to the users. By migrating the applications to a server near to the user, also mobile users can have a low client-server latency. A second research challenge is the optimization of the communication protocols between the different parties in our architecture. The thin client provider merges graphical data and forwards user events to the appropriate server. Installing monitor probes on the thin client device and along the client-server connection allows to optimize the parameters of the graphical output stream.

References


