IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Hershey PA 17033-1117, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITB8026

Chapter XVIII

Supporting Electronic Commerce of Software Products Through Pay-Per-Use Rental of Downloadable Tools

Giancarlo Succi and Raymond Wong
University of Alberta, Canada

Eric Liu
University of Calgary, Canada

Carlo Bonamico and Tullio Vernazza
DIST – Università di Genova, Italy

Luigi Benedicenti
University of Regina, Italy

The Internet supports the development of software tools that can be downloaded on demand by users, software tools on-demand. *These tools cannot be purchased like products, because they do not reside on the user's machine. Rather, they can be used as "services."*

In this chapter, we propose a new paying mechanism for electronic commerce of software tools-on-demand, that charges users according to how much they have used a given tool: pay-per-use *rental. We discuss the benefits of pay-per-use for users and producers, and we evidence the critical issues in designing a system to support pay-per-use.*

Then we introduce WebMetrics, our pay-per-use system that supplies software metrics collection and analysis tools—on demand. WebMetrics integrates pay-per-use in a client/server Java application. It is based on the idea of prepaid "virtual cards" similar to rechargeable prepaid calling cards for long distance telephone calls.

We conclude with a discussion on the open issues: security, reliability, availability, and standards.

INTRODUCTION

The pervasiveness of Internet connectivity and the wide diffusion of Java-capable browsers foster innovative techniques for software distribution. In this chapter, we propose a new model for the electronic commerce of software tools based on a pay-per-use rental policy.

Pay-per-use rental of downloadable tools is the natural exploitation of Java applets that can be transferred on demand to the user's machine and executed dynamically inside a browser. While software rental is not a new idea (Flamnia and McCandless, 1996), at present no example of a standard pay-per-use rental mechanism for downloadable software tools exists.

This approach benefits from the advantages of central management of tools and zero maintenance for users typical of Java applets, together with a new way to pay for their use. Software rental presents several advantages to producers and users. Pay-per-use rental is particularly suited to Web-based applications, because they are offered to a very heterogeneous and dynamic user population (Bakos and Brynjolfson, 1997).

This chapter describes advantages and issues related to pay-per-use, and explains how to add it to Web-based systems, by presenting the example of pay-per-use integration in WebMetrics, a Web-based system providing distributed collection, management, and analysis of source code metrics.

This chapter is organized as follows. Section 2 discusses tools-on-demand. Section 3 presents the role of pay-per-use. Section 4 introduces WebMetrics, our prototype pay-per-use application. Section 5 describes the architecture of WebMetrics. Section 6 presents a list of open issues. Section 7 draws some conclusions.

TOOLS-ON-DEMAND

The Web already supports two mechanisms for electronic software distribution:

- Free software can be downloaded directly.
- Commercial, shrink-wrapped software can be purchased on-line and then downloaded.

In both cases users have to install the software on their computers.

In the last two years, another distribution mechanism has become popular: tools that are downloaded on demand from the developer's server and executed inside a browser to avoid installation on the user's machine. This approach presents several benefits (Yourdon, 1996):

- The tools are immediately available to any Internet-connected computer, providing a set of computing services available appealing to telecommuters, mobile users, and consultants.
- The tools can run on any hardware platform with a Java-capable Web browser.
- Since the tools are downloaded from a central server, users always get their latest version (Gosling and McGilton, 1996).
- There is no installation, so managing a large user base becomes more viable.
- Maintenance costs are significantly reduced (Gupta *et al.,* 1998).