

BITS C461 Software Engineering

First Semester 2003-2004

Project Requirements:

Infrastructure for Software Service Provider (SSP)

Project Sponsor: Aditya P. Mathur

Department of Computer Science

Purdue University, West Lafayette, IN, USA

Original: Aug 4, 2003. Revision: None.

1 Long Term Goal

The long term goal of this project is to create a commercially available, open source, infrastructure that allows independent vendors to submit their software capsules into a library and subscribers of the library to download the capsules on-demand whenever needed. The library will be managed by a Software Service Provider (SSP). A capsule is a fully functional piece of software that supports a subset of application features. The infrastructure will contain the following minimal set of components: (i) Capsule Specification and Validation (CSV) to verify if the capsule submitted satisfies the interface and reliability requirements imposed by the SSP, (ii) Capsule Server (CS), (iii) Library and Financial Accounts Manager (LFAM), and (iv) on-line Library Management Interface (LMI).

The library will allow software application users to subscribe to library services and pay for software on a monthly or per-use basis. This mechanism will free users from the need to buy software and save it on their local disks. Instead, users will simply log in to the library and begin using the software they need. Applications will be composed of several capsules. Each capsule will support a small subset of application features. Interface definition and strict enforcement of adherence to these definitions will allow developers from all over the world to submit capsules into the library. Developers will be paid for their capsules by the library owner depending on the amount of usage of the capsules.

The software rental model proposed here will free the software users from the financial and other inconveniences of having to buy software, upgrade software, and frequently upgrade their hardware. This model is also nature-friendly as it is likely to lead to tremendous savings in materials that go into the manufacture of hard drives, memories, and other computer hardware.

2 What is to be done ?

Selected teams of students of BITS C461 will be required to complete the following tasks related to this project:

1. Understand and analyze project requirements. The product of this activity will be a list of high level and low level use cases, system sequence diagrams, and a domain model expressed in UML.
2. Complete a design of the various components of the infrastructure and of the entire library itself. The product of this activity will be a complete design of the library and its individual components. The design must be expressed in UML.
3. A prototype of the library. The prototype will demonstrate how the library will work. It will have only a selected subset of the features that a full fledged library might have. Teams working on the project will decide what features to select for prototyping.

3 A User's Perspective of SSP

Any one with access to the internet will be able to subscribe to the services of the SSP through a monthly subscription fee. Membership with an SSP will entitle a member to the use of a set of software applications and a set of application features. The amount of use, the applications available, and the features available will depend on the membership level.

A member will be able to access the SSP through an SSP client running on the user's machine. This client, known as SSP Agent (SSPA) will allow a user to log in to the SSP through secure shell. Upon successful entry into the SSP, the SSPA will offer the list of services available to the user. The user can now begin using any of the available applications. When the user first begins using an application, an appropriate capsule, say C , is made available. C supports only a limited set of features most frequently used by this user. When the user wants to use a feature not supported by C , a *feature interruptis* said to occur. This causes the SSPA to communicate with the Capsule Server to obtain the appropriate capsule. Feature interrupts and their handling are totally transparent to the user.

All of user's data will be saved in the user's local machine or at the SSP site

depending on the nature of the membership subscription. The user will be billed monthly depending on the nature of the membership and the amount of software used.

4 A Developer's Perspective of SSP

Any software developer can contribute a set of capsules to the SSP for validation and use. A capsule is a fully functional piece of software that supports a subset of application features. For example, a spreadsheet could consist of many small capsules. The aggregate of all capsules constitutes an application. The sets of features in any two capsules might not be disjoint. Further, one capsule might belong to more than one application. For example, a capsule to sort numbers might belong to the spreadsheet and a tax application.

Development of capsules is facilitated by the SSP. For this purpose the SSP makes a complete design of a applications available to potential developers. Interfaces to the application are clearly defined. The developers decide what portions of the applications they wish to *capsulize*. Developers develop capsules and submit to the SSP who then subjects each capsule to rigorous testing and validation against the specified interface and design. Once a capsule passes the various tests, it is included in the SSP's capsule library and made available to its subscribers.

The on-line Library Management Interface allows developers and users to browse the library of capsules. Thus, for example, developers can find out whether or not there exists a capsule that supports a given feature. AS another example, a user could find out the reliability of various capsules that support a given feature.

5 The SSP Management's Perspective

The SSP management uses special CSV to manage the development, testing, and use of capsules. The CS module interacts with the SSPA to complete the login procedure and allows users to download capsules. Note that capsules may execute on user machine or the subscriber machine depending on factors such as capsule size,

processing requirements, and the type of membership. All financial requirements are met by LFAM which is responsible for account maintenance, tracking usage of capsules, and regular billing.

The requirements given here are incomplete and ambiguous. Teams will need to resolve ambiguities and, where necessary, complete the requirements. Help will be available from the instructors.