

Sammanfattning

Denna rapport beskriver utvecklingen av en Instant Messaging klient som implementerats som en språkwrapper. Klienten använder ett känt öppen källkods-bibliotek vid namn libGaim för att få tillgång till Instant Messaging tjänster. Klienten fungerar som en språkwrapper till detta bibliotek, och låter andra applikationer nyttja biblioteket genom ett databasgränssnitt. Arbetet har utförts åt och hos Evolve i Karlstad. Systemet som utvecklats klarar att hantera vanliga Instant Messaging-funktioner så som uppkoppling av konto, nedkoppling, ändring av status, visa kompisikoner, och hämtning av information angående kompisar.

I beskrivandet av arbetet med detta system ges bakgrundsinformation angående wrappers och Instant Messaging, och dessutom angående de verktyg och metoder som använts likt byggverktyget SCons och utvecklingsmetoden eXtreme Programming. Valet av lösning och design presenteras.

Som en del av arbetet undersöks språkwrappers med hjälp av ett antal tester. Dessa tester visade på vilka skillnader som kan väntas beroende på valet av språkwrapper. En språkgenerisk språkwrapper (som skapar ett gränssnitt mot flera språk) behöver inte utgöra någon större prestandaoverhead jämfört med en språkstatisk språkwrapper (som skapar ett gränssnitt mot *ett* språk).

Resultatet av arbetet blev ett system som Evolve blev nöjda med och som uppfyllde alla krav som ställdes, antingen fullständigt eller delvis.

The development of an Instant Messaging client as a language wrapper

This report describes the development of an Instant Messaging client that is implemented as a language wrapper. The client uses a well known open source library, known as libGaim, to access Instant Messaging services. The client works as a language wrapper for this library, and allows other applications to use the library by means of a database interface. The work has been carried out at and for Evolve in Karlstad. The system that has been developed can handle common Instant Messaging operations such as connecting a user account, disconnecting, changing an accounts status, showing buddy icons, and collecting information regarding buddies.

In the description of the work with this system, background information is given regarding wrappers and Instant Messaging, and also regarding the tools and methods used, such as the build tool SCons and the development method eXtreme Programming. The choice of solution and design is presented.

As a part of the work, language wrappers are evaluated using a number of tests. These tests show the differences that can be expected depending on the choice of language wrapper. A language generic language wrapper (which creates an interface for several languages) need not imply any significant performance overhead compared to a language static language wrapper (which creates an interface towards *one* language).

The result of the work we have done was a system that Evolve was pleased with and

that satisfies all the requirements, either completely or partially.