Bauhaus-Universität
Weimar

CML://PRIMI Base

Platform for Research on Instant Messaging Infrastructures

PRIMI is the top-level name for a series of projects aiming to create an **open and extensible research platform** in the field of computer-mediated communication esp. instant messaging. Central research issues are **communication protocols**, **graphical user interfaces**, **basic and advanced services**, and **application integration** (see Figure 1). Client and server side aspects are considered.

PRIMI-Base is the first subproject. Its goal was to develop the **core platform** and services needed in future projects. Among these are high and low level services. High level services offer bundled functionality or part aspects of lower level services to other parts of the application. PRIMI-Base implements services for the general administration of plugins (**plugin service**), the administration of communication protocol plugins (**connector service**), and application-wide multilevel logging (**log service**). These services can be configured at runtime using the application's configuration window (see Figure 2).

The **plugin architecture** makes it possible to add your own and third party graphical user interfaces and connector plugins. The latter implement communication protocols. One may install plugins locally in the application's plugin directory or via the network, simply by providing its URL. The plugin is downloaded and deployed at runtime. All of this allows plugging together a multiprotocol instant messenger with a user interface of your choice (see Figure 2).

A core feature distinguishing PRIMI from all other instant messaging applications is its **extensive logging capability**. The log service offers a multilevel logging facility based on Sun's Java Logging API. The detail of logging can be configured at runtime. Local and remote logging (using XML-RPC) is supported. However, plugin developers do not have to consider logging in their plugins. The platform services provide containers based on Gamma et al's (1994) Proxy Pattern to do the trick (see Figure 5).

Besides PRIMI's core platform we also implemented **two reference plugins**. We chose to implement a **communication protocol plugin** based on the Extensible Messaging and Presence Protocol (**XMPP**) defined in RFCs 3920-3923. It is also known as the Jabber protocol. The plugin was developed using the Adapter Pattern (Gamma et al. 1994) to integrate a third party library (Smack-Lib) with the platform's connector API. Thus PRIMI can be used as a regular Jabber client.

Last but not least we provide **graphical user interface** reference **plugin**. It offers multiprotocol support and allows the administration of multiple accounts. Besides that it provides standard presence awareness and chat features, yet utilizing our platform services. The plugin is able to control numerous connector plugins within one GUI (see Figure 3 and 4).

For further information see:
http://cml.medien.uni-weimar.de/primi

Prof. Dr. Tom Gross
Christoph Oemig, M.Sc.

Yunlu Ai
Robert Gerling
Andre Kunert
Kai Riege

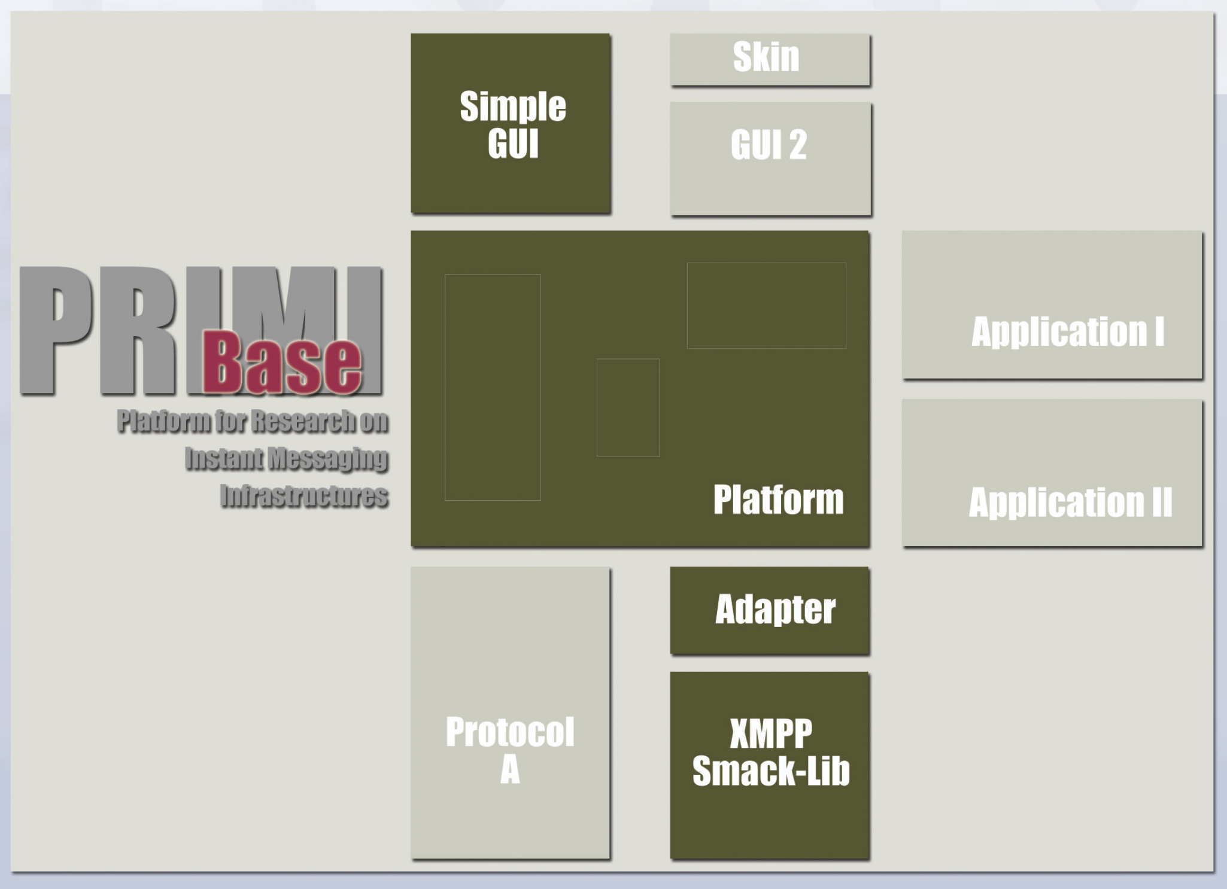
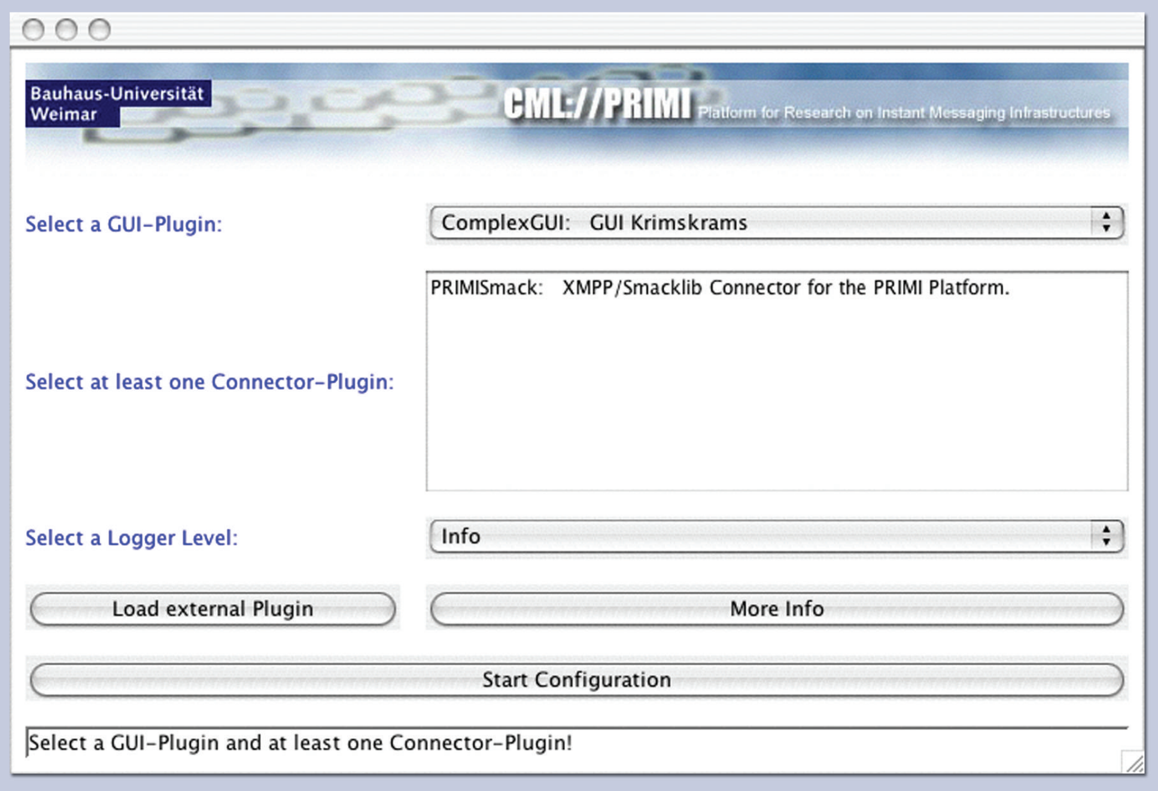


Figure 1: A general overview. The PRIMI-Base project realized the core platform, a GUI and a communication protocol plugin.



Bauhaus-Universität Weimar CML://PRIMI Platform for Research on Instant Messaging Infrastructures

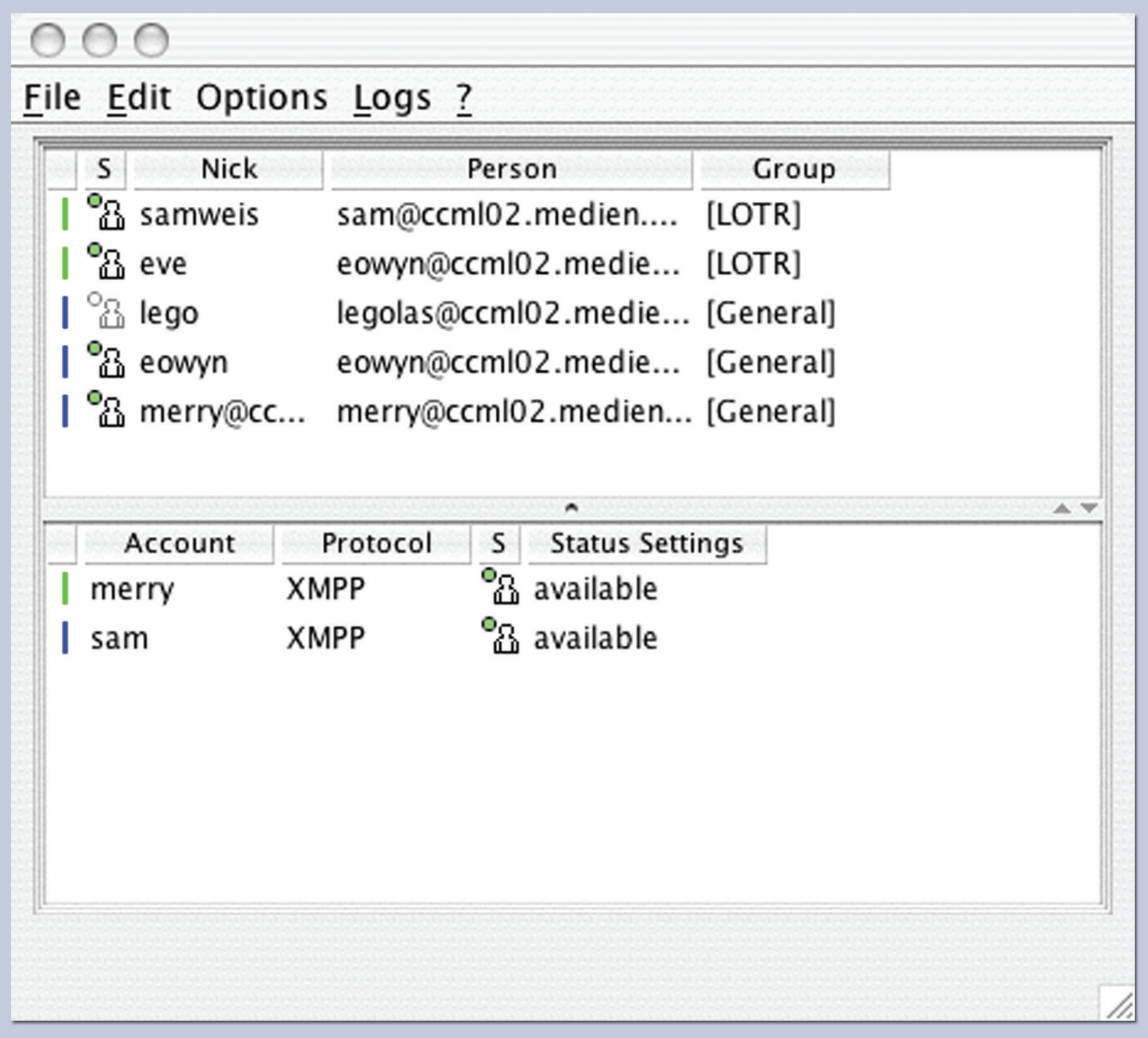
Select a GUI-Plugin: ComplexGUI: GUI Krimskrams

Select at least one Connector-Plugin: PRIMISmack: XMPP/Smacklib Connector for the PRIMI Platform.

Select a Logger Level: Info

Select a GUI-Plugin and at least one Connector-Plugin!

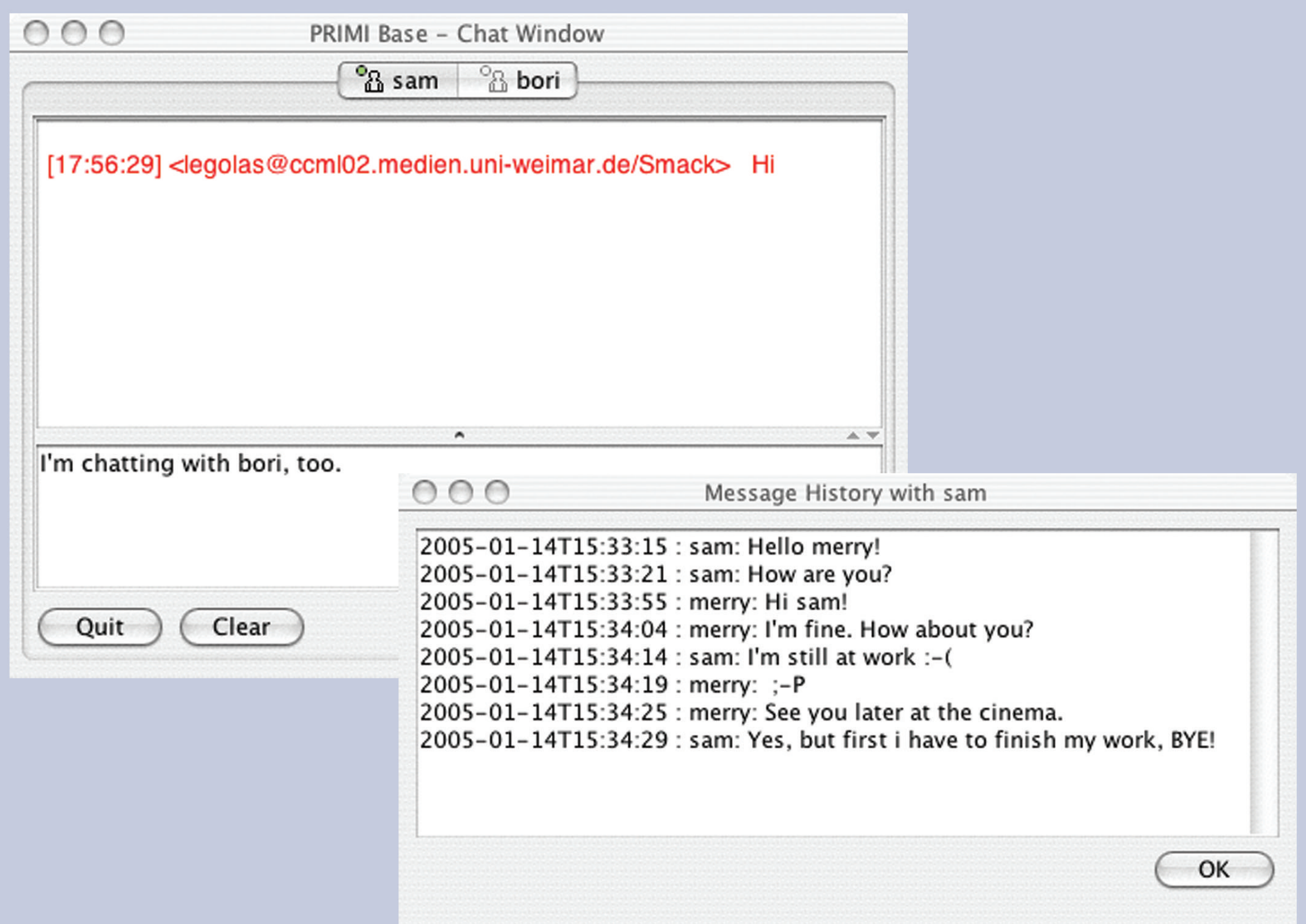
Figure 2: The configuration window. It allows to select and configure GUI and connector plugins as well as the log level. Once a configration is chosen the application can be launched.



S	Nick	Person	Group
	samweis	sam@ccml02.medien....	[LOTR]
	eve	eowyn@ccml02.medie...	[LOTR]
	lego	legolas@ccml02.medie...	[General]
	eowyn	eowyn@ccml02.medie...	[General]
	merry@cc...	merry@ccml02.medien...	[General]

Account	Protocol	S	Status Settings
merry	XMPP		available
sam	XMPP		available

Figure 3: Our GUI reference plugin offers a multiprotocol instant messenger interface. Multiple user accounts are supported.



PRIMI Base – Chat Window

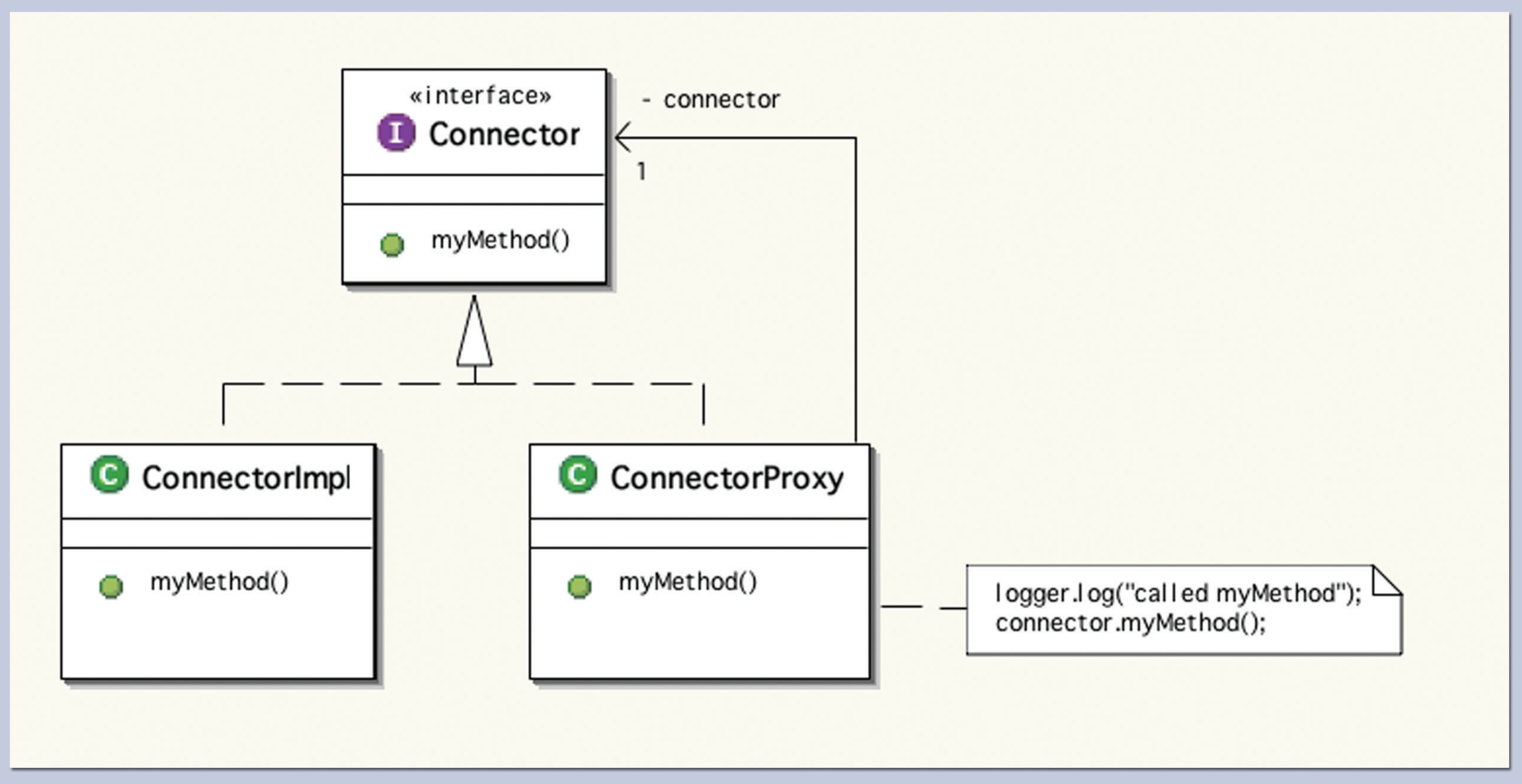
[17:56:29] <legolas@ccml02.medien.uni-weimar.de/Smack> Hi

I'm chatting with bori, too.

Message History with sam

```
2005-01-14T15:33:15 : sam: Hello merry!
2005-01-14T15:33:21 : sam: How are you?
2005-01-14T15:33:55 : merry: Hi sam!
2005-01-14T15:34:04 : merry: I'm fine. How about you?
2005-01-14T15:34:14 : sam: I'm still at work :-(
2005-01-14T15:34:19 : merry:  ;-P
2005-01-14T15:34:25 : merry: See you later at the cinema.
2005-01-14T15:34:29 : sam: Yes, but first i have to finish my work, BYE!
```

Figure 4: Concurrent chats appear tabbed in a single window. The clients local history shows just one aspect of the platforms logging capabilities.



```
logger.log("called myMethod");
connector.myMethod();
```

Figure 5: The Proxy Pattern (Gamma et al. 1994). Connectors do not have to provide any logging code nor do plugin programmers need to know about the logging architecture. The platform wraps these objects into connector proxies which perform the logging actions according to the platform's needs.

Bauhaus-University Weimar, Faculty of Media, Cooperative Media Lab

WS 2004/2005