



SIP Communicator

Emil Ivov

FOSDEM, February 25th 2007



Table of Contents

- What is SIP Communicator?
- SIP Communicator Overview – Currently supported and planned features.
- Deployment, customization and maintenance – Extensibility, ease of maintenance and deployment.
- Creating plug-ins for SIP Communicator



SIP Communicator Overview

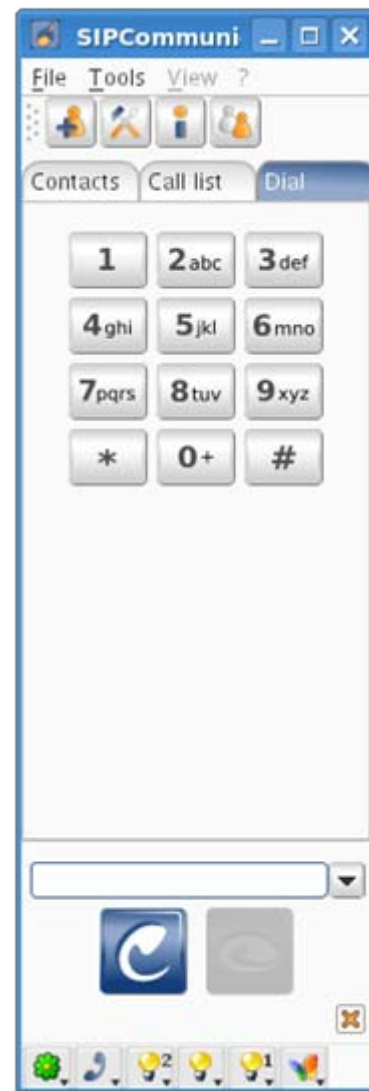
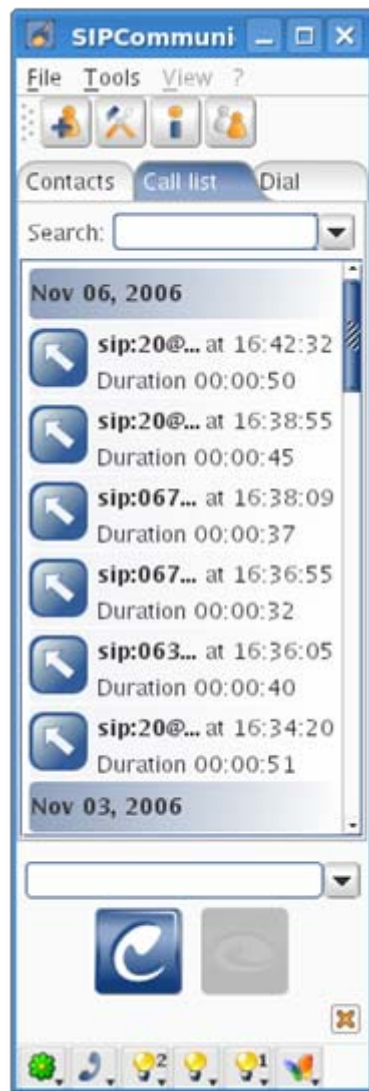


SIP Communicator is an open source (LGPL) Audio/Video software phone and instant messenger. Among others, we currently support:

- *Audio and Video calls with SIP*
- Instant messaging with Jabber, ICQ/AIM, Yahoo! Msngr and MSN
- IPv6 support for SIP and Jabber
- Support for multiple accounts and meta contacts
- Basic NAT & Firewall Traversal with STUN.
- Modularity, extensibility, and flexibility with OSGi
- Platform specific installers for Windows, Debian, Fedora, Mac OS X visit <http://sip-communicator.org> for more details



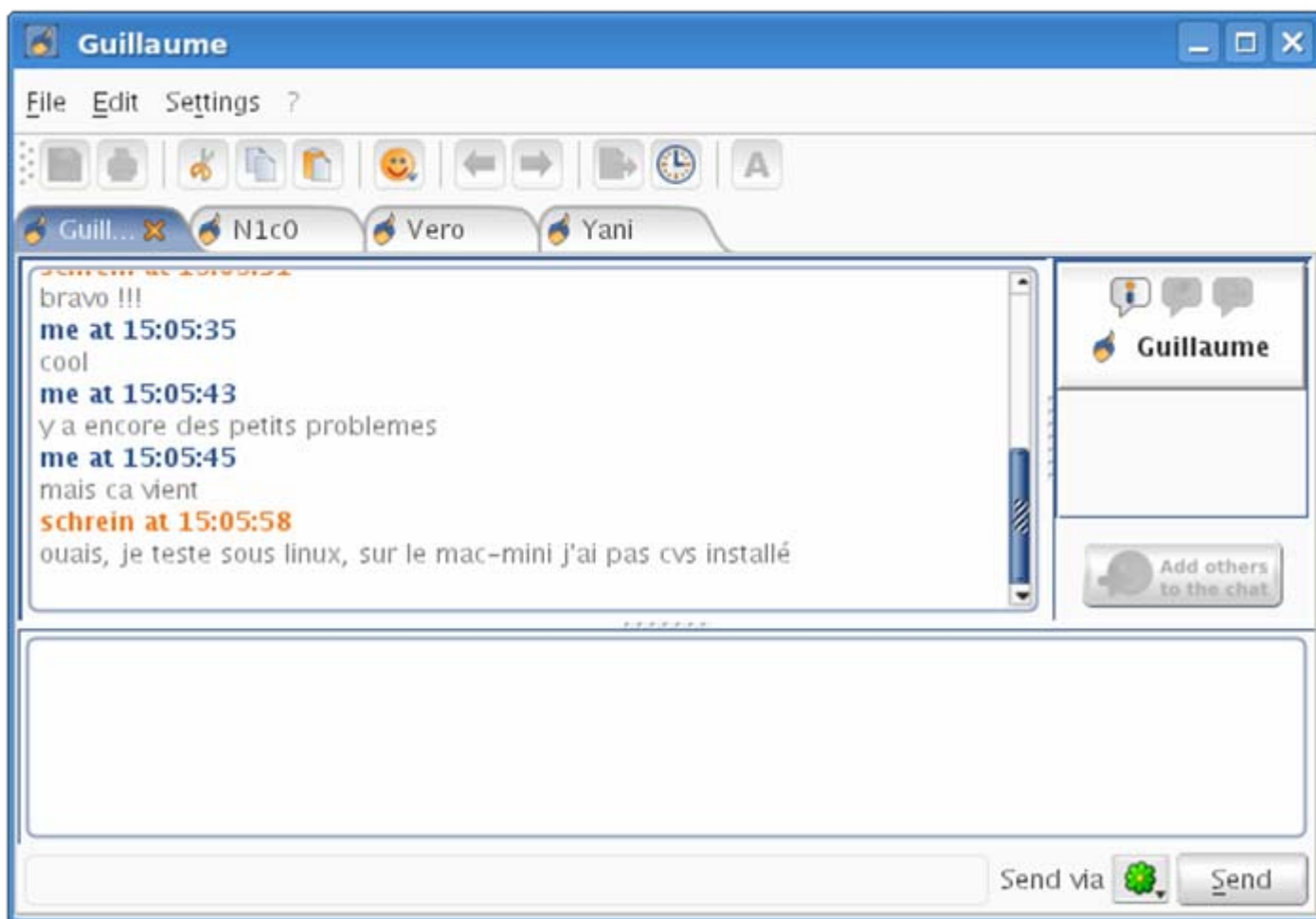
SIP Communicator Overview





SIP Communicator Overview

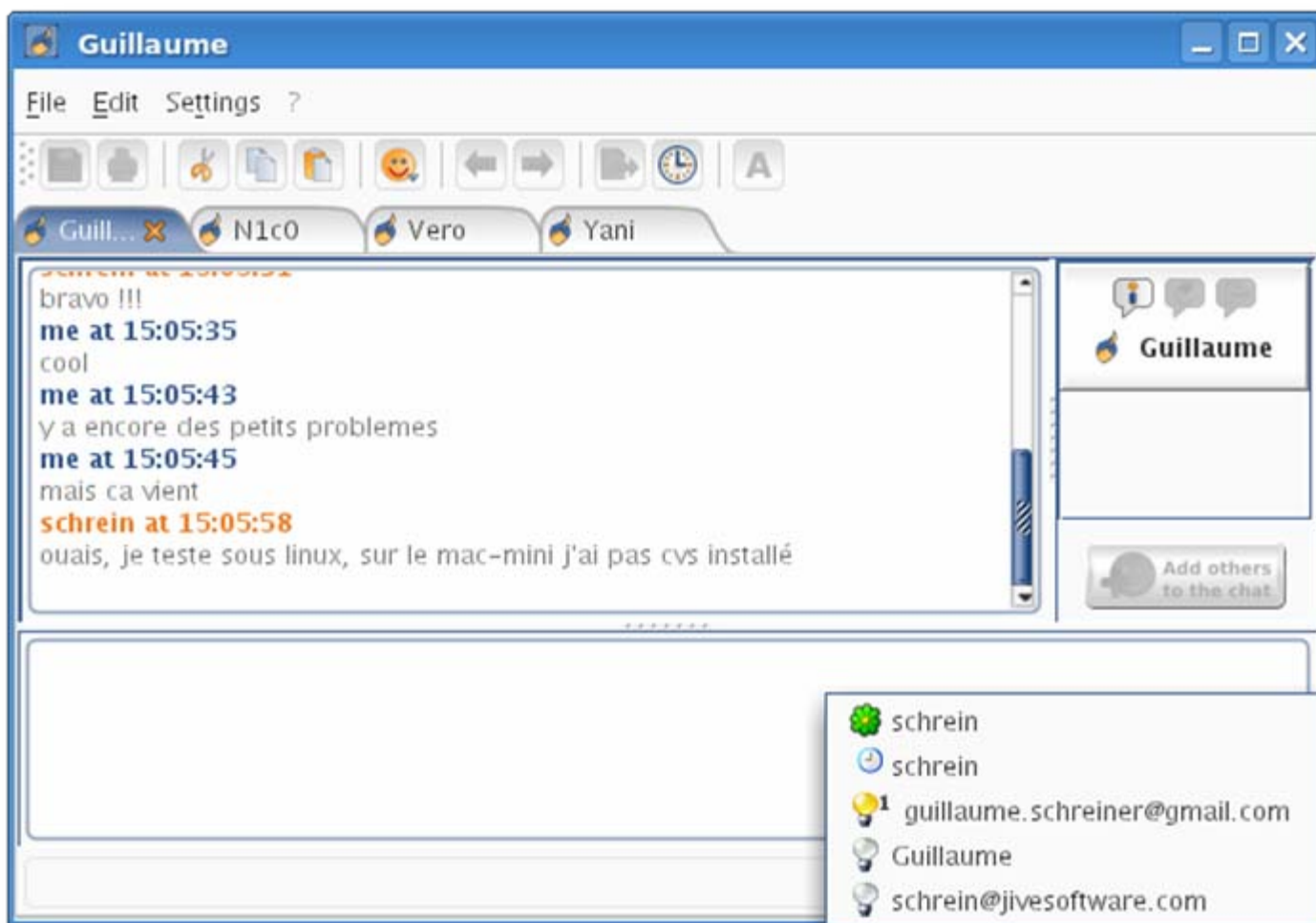
Instant Messaging





SIP Communicator Overview

Instant Messaging





SIP Communicator Overview

Instant Messaging History

The screenshot shows a window titled "History - Guillaume" with a search bar and a list of messages. The messages are dated from September 15, 2006, to November 7, 2006. The messages include timestamps and names like "schrein" and "me".

History

Search: Search

Sep 15, 2006
Sep 21, 2006
Sep 28, 2006
Sep 29, 2006
Oct 02, 2006
Oct 04, 2006
Oct 06, 2006
Oct 09, 2006
Oct 10, 2006
Oct 11, 2006
Oct 13, 2006
Nov 03, 2006
Nov 07, 2006

drop cool
schrein at 15:05:28
c dla balle
me at 15:05:29
ca marche?
schrein at 15:05:31
bravo !!!
me at 15:05:35
cool
me at 15:05:43
y a encore des petits problemes
me at 15:05:45
mais ca vient
schrein at 15:05:58
ouais, je teste sous linux, sur le mac-mini j'ai pas cvs installé

51%



Features Currently in Progress

- **Powerful and scalable firewall support** – Combine technologies like TURN, STUN, ICE and P2P in order to achieve powerful, secure, and scalable firewall support.
- **Robust and Seamless IPv6 support** – We aim to achieve more than a mere “support for IPv6”. We would like to guarantee its transparent and seamless usage. In other words, usage of IPv6 should not require any special configuration and should not cause any problems.
- **Security** – Encrypt all media. Usage of P2P for firewall traversal imposes reliable security of all media transmitted by the SIP Communicator in order to prevent relaying nodes from eavesdropping.
- **Automatic updates and one-click plug-in installation** - Provide a user interface and an online repository for SIP Communicator plug-ins.
- **Other plug-ins** - IRC, shared whiteboards, multiparty video and chat conferencing.



A Look Inside – The OSGi Core

OSGi Framework

Jabber

SIP

MSN

Media

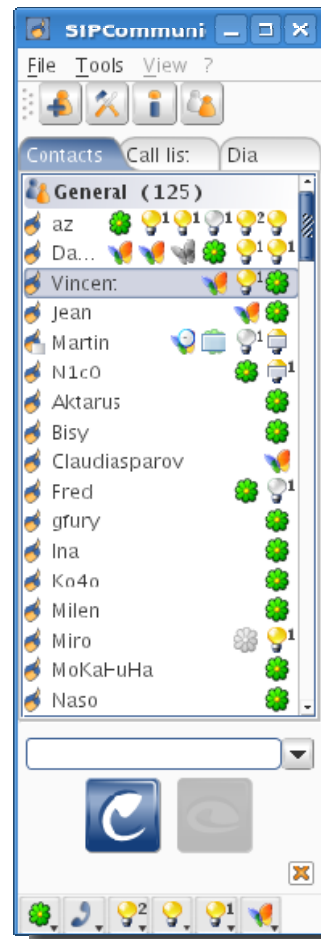
User Interface

Message History

Meta Contact List

Firewall Traversal

ICQ





A Look Inside – The OSGi Core

Jabber

SIP

MSN

Media

User Interface

Message History

Meta Contact List

Firewall Traversal

ICQ

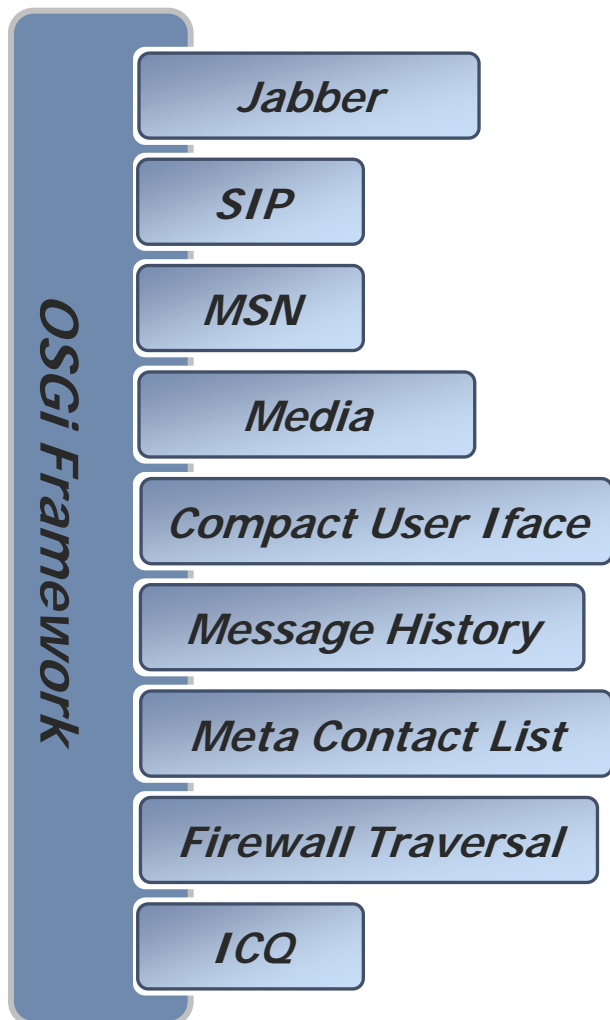
OSGi Framework

SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

- **Modularity & Flexibility** – All components of SIP Communicator are implemented as separate, replaceable modules. It is possible to run the application with different sets of features and functionalities depending on the intended user or target platform.
- **Extensibility** – It is very easy to implement additional features in the form of plug-ins. Developers that are new to SIP Communicator could easily start developing for it since they only need to get acquainted with existing APIs and not the entire source code.
- **Ease of maintenance and deployment** – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.



A Look Inside – Modularity & Flexibility



SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

- **Modularity & Flexibility** – All components of SIP Communicator are implemented as separate, replaceable modules. It is possible to run the application with different sets of features and functionalities depending on the intended user or target platform.
- **Extensibility** – It is very easy to implement additional features in the form of plug-ins. Developers that are new to SIP Communicator could easily start developing for it since they only need to get acquainted with existing APIs and not the entire source code.
- **Ease of maintenance and deployment** – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.



A Look Inside – Modularity & Flexibility

Jabber

SIP

Media

Compact User Iface

Meta Contact List

Firewall Traversal

OSGi Framework

SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

- **Modularity & Flexibility** – All components of SIP Communicator are implemented as separate, replaceable modules. It is possible to run the application with different sets of features and functionalities depending on the intended user or target platform.
- **Extensibility** – It is very easy to implement additional features in the form of plug-ins. Developers that are new to SIP Communicator could easily start developing for it since they only need to get acquainted with existing APIs and not the entire source code.
- **Ease of maintenance and deployment** – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.



A Look Inside – Extensibility

Jabber

SIP

Voice Recognition

Media

Compact User Iface

GPS Navigation

Meta Contact List

Firewall Traversal

OSGi Framework

SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

- **Modularity & Flexibility** – All components of SIP Communicator are implemented as separate, replaceable modules. It is possible to run the application with different sets of features and functionalities depending on the intended user or target platform.
- **Extensibility** – It is very easy to implement additional features in the form of plug-ins. Developers that are new to SIP Communicator could easily start developing for it since they only need to get acquainted with existing APIs and not the entire source code.
- **Ease of maintenance and deployment** – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.



A Look Inside – Deployment and Maintenance

Jabber

SIP

Voice Recognition

Media

User Interface

GPS Navigation

Meta Contact List

Firewall Traversal

OSGi Framework

SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

Web Repository

IRC

Zero Conf

Spell Checker

- Ease of maintenance and deployment – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.



A Look Inside – Deployment and Maintenance

SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

Jabber

SIP

http

Voice Recognition

Media

User Interface

GPS Navigation

Meta Contact List

Web Repository

IRC

Zero Conf

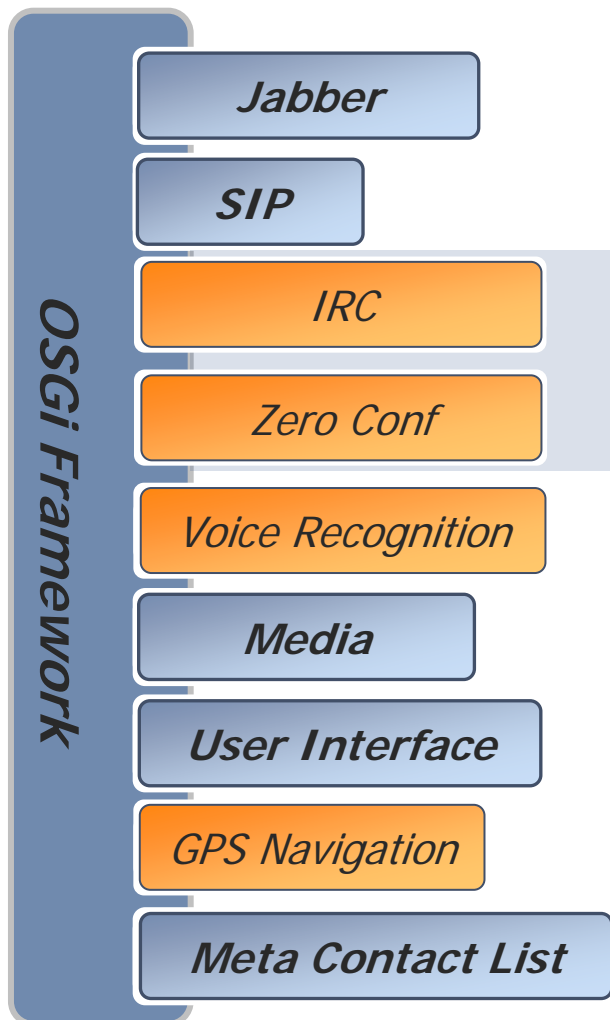
Spell Checker

- Ease of maintenance and deployment – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.

OSGi Framework



A Look Inside – Deployment and Maintenance



SIP Communicator is built upon the Apache Felix implementation of the OSGi framework. This helps us provide qualities such as:

http

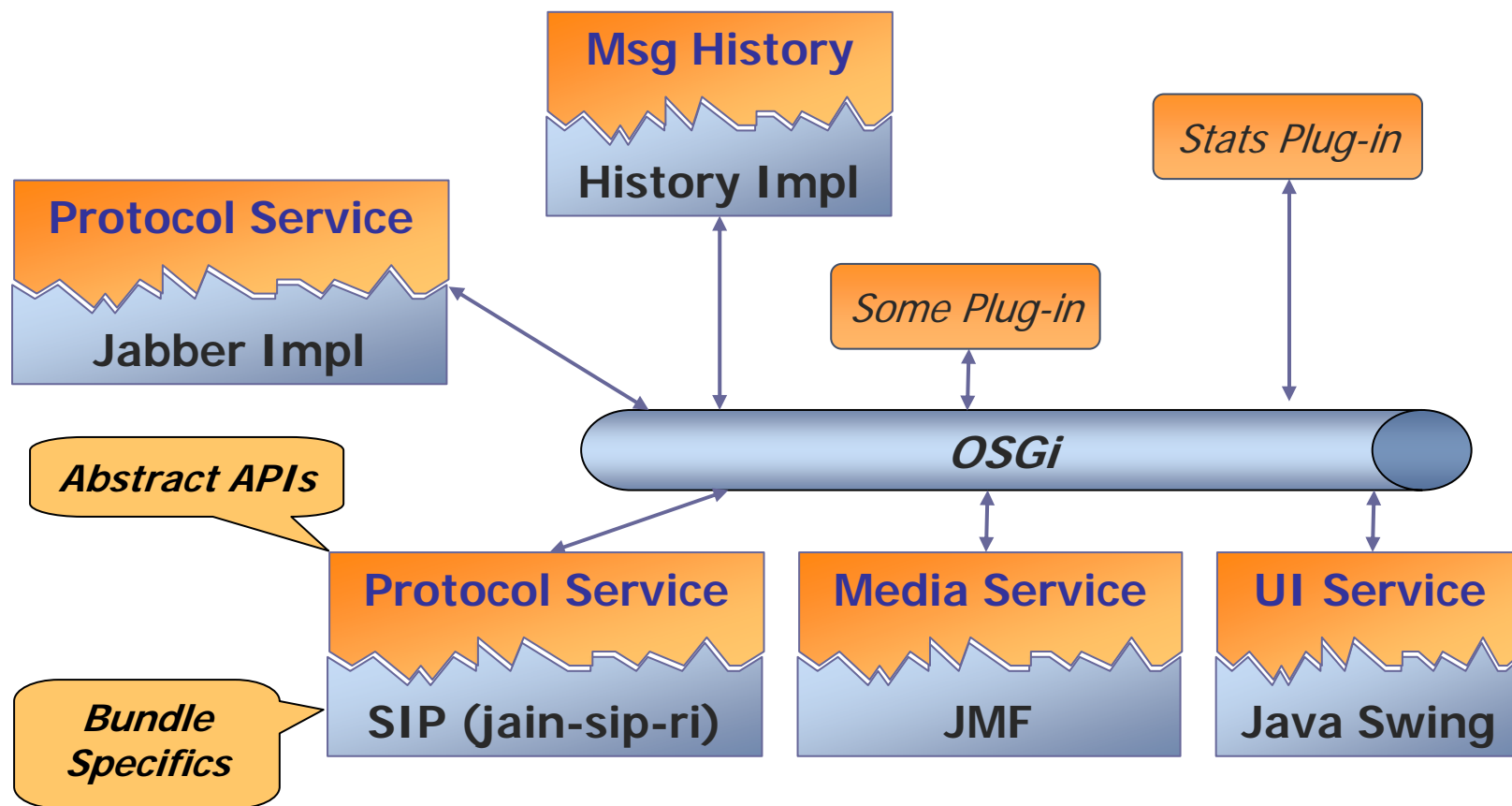
Web Repository

Spell Checker

- Ease of maintenance and deployment – The concept of an OSGi Bundle Repository allows SIP Communicator users to download, install and configure new plug-ins with a few clicks.

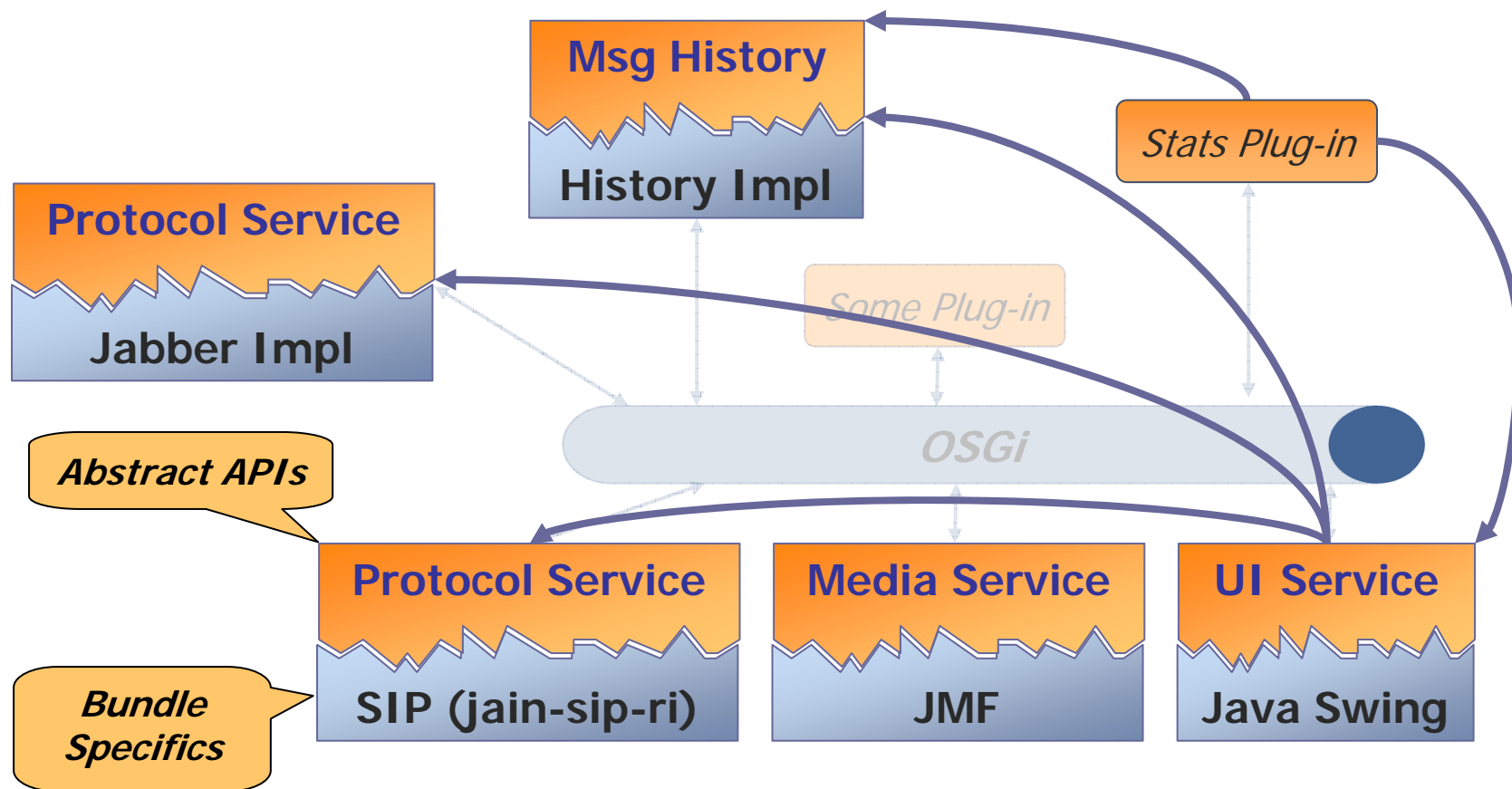


SIP Communicator 1.0 – Application Design





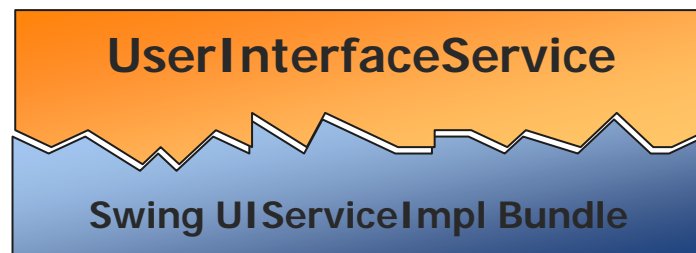
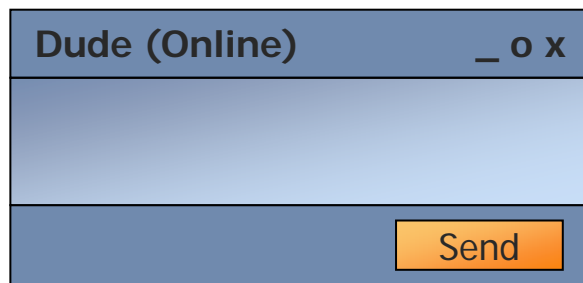
SIP Communicator 1.0 – Application Design





SIP Communicator 1.0 Architecture

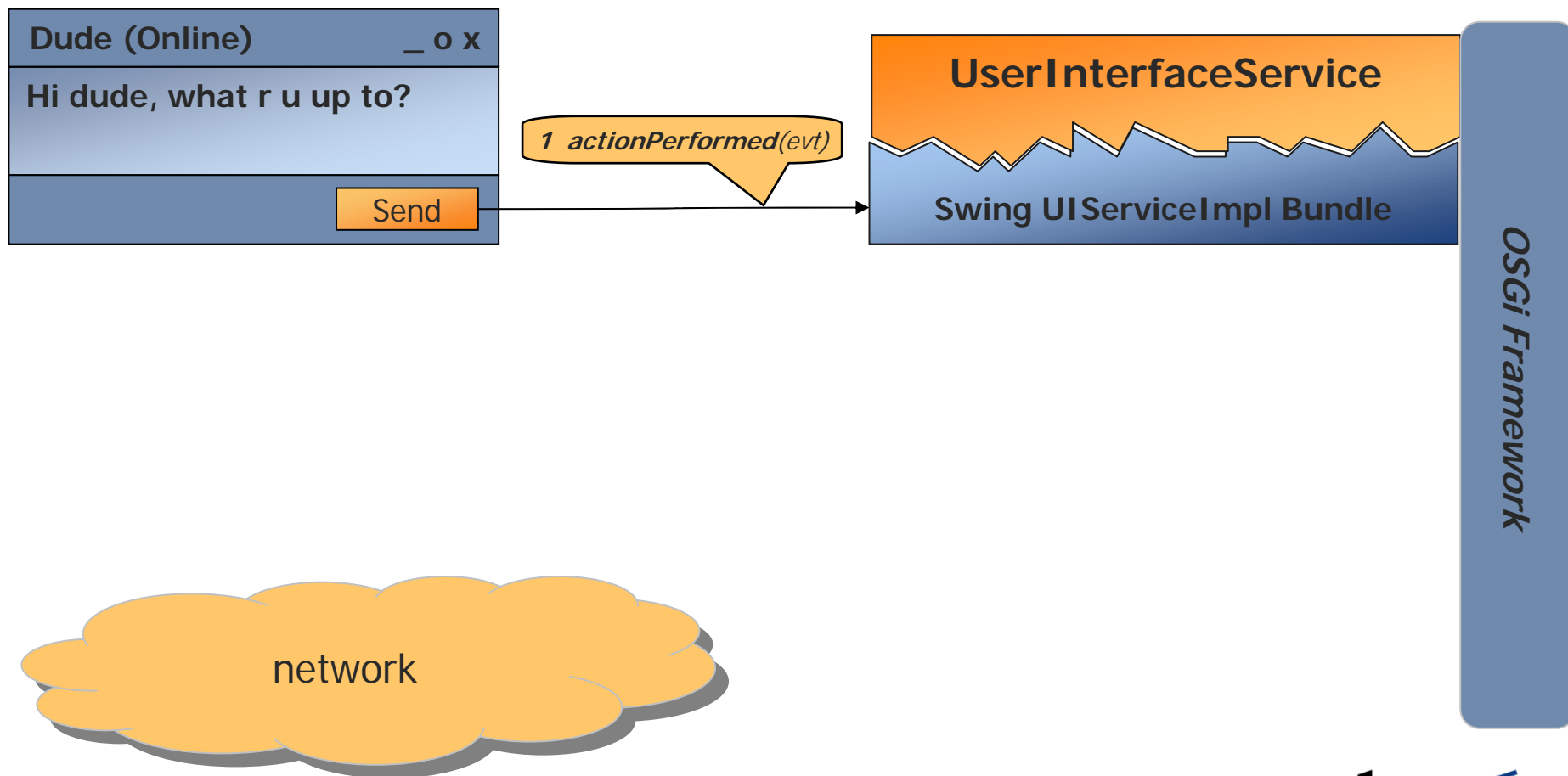
A Simple Scenario





SIP Communicator 1.0 Architecture

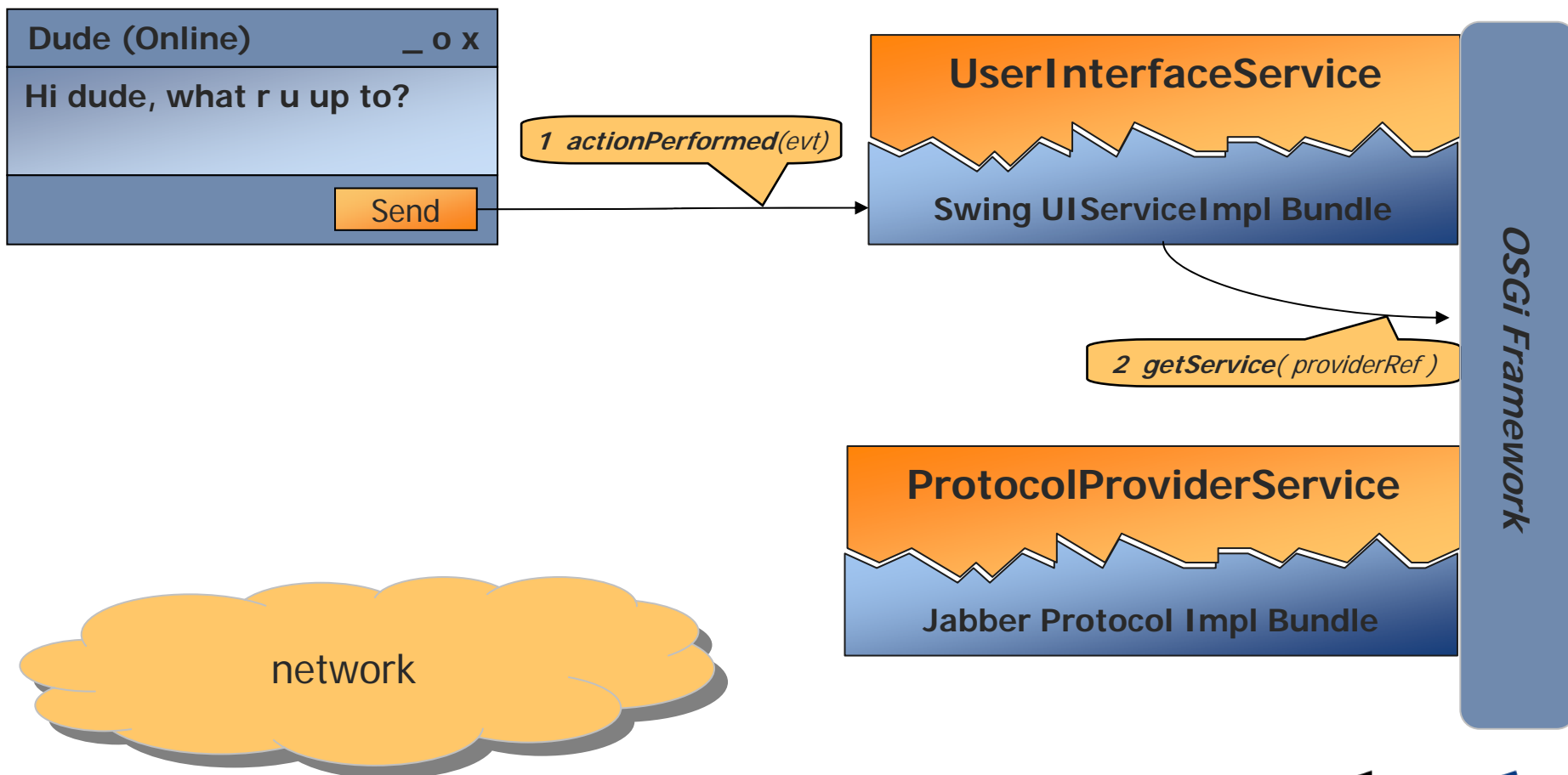
A Simple Scenario





SIP Communicator 1.0 Architecture

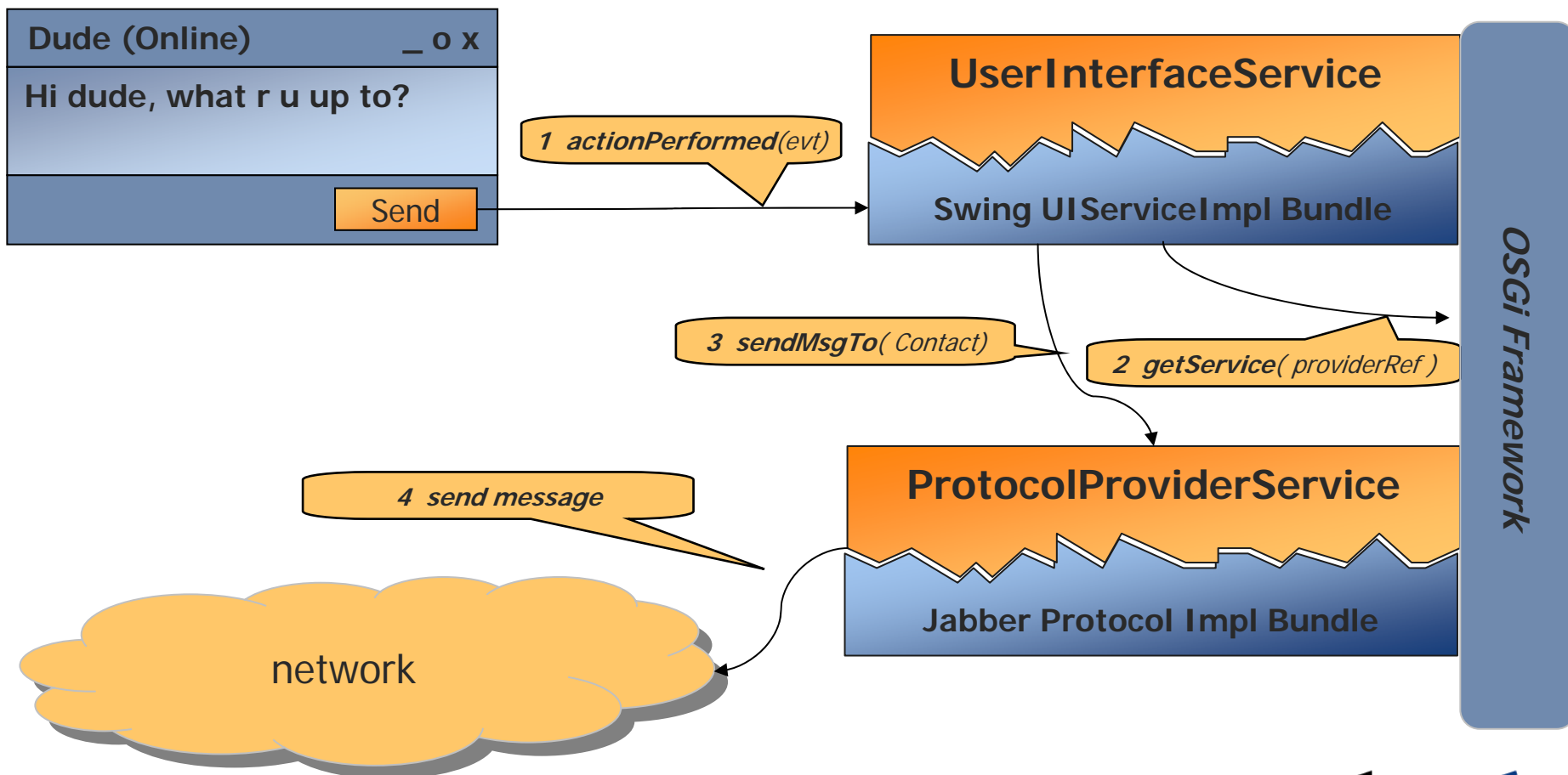
A Simple Scenario





SIP Communicator 1.0 Architecture

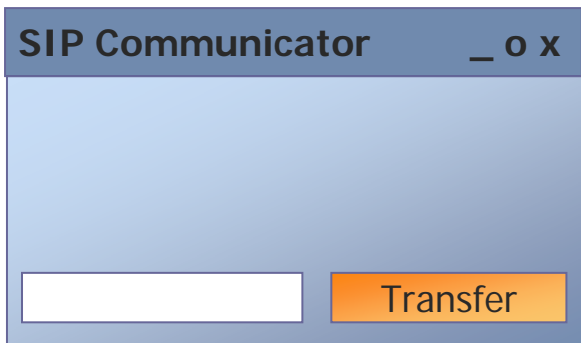
A Simple Scenario





SIP Communicator 1.0 Architecture

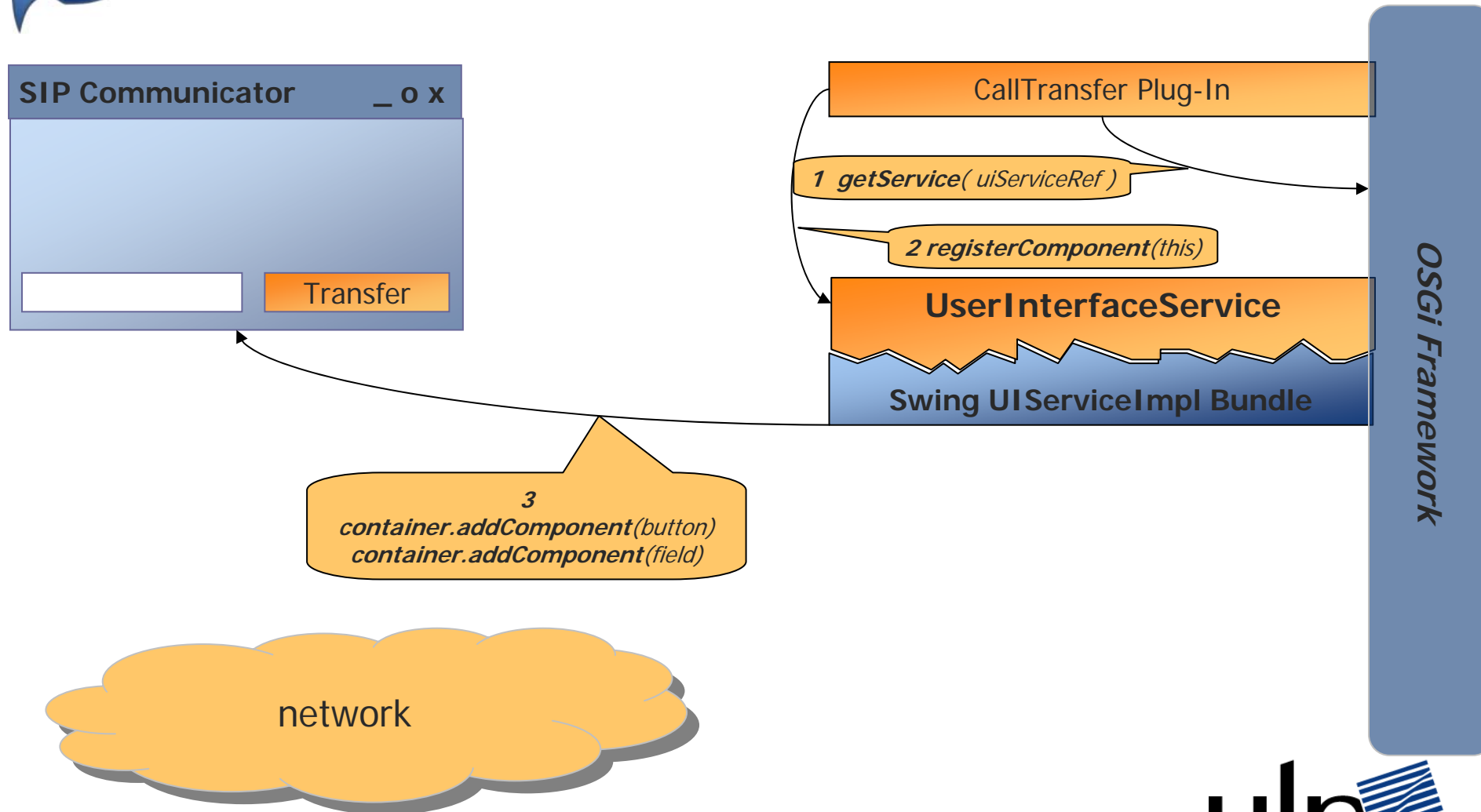
A Slightly More Complex Scenario (1)





SIP Communicator 1.0 Architecture

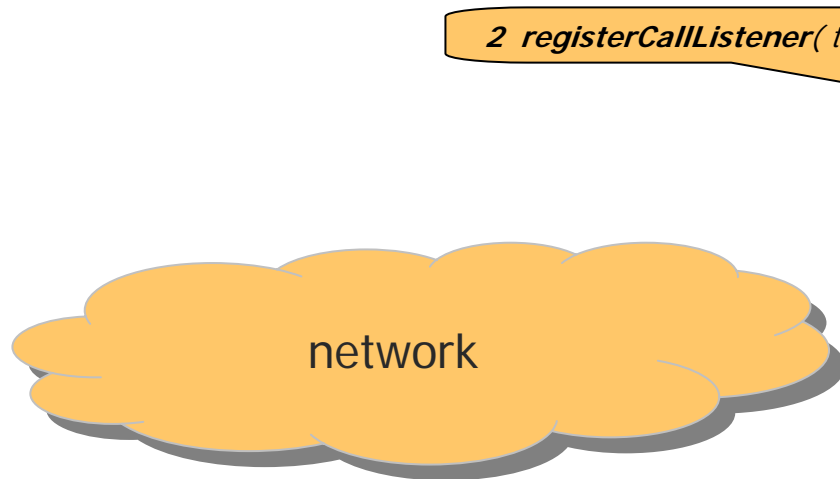
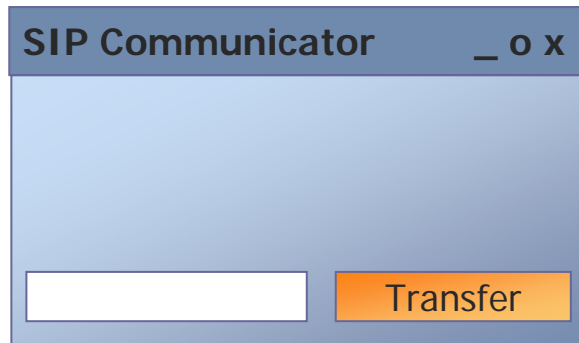
A Slightly More Complex Scenario (1)





SIP Communicator 1.0 Architecture

A Slightly More Complex Scenario (2)



2 registerCallListener(this)

CallTransfer Plug-In

UserInterfaceService
Swing UIServiceImpl Bundle

1 getService(providerRef)

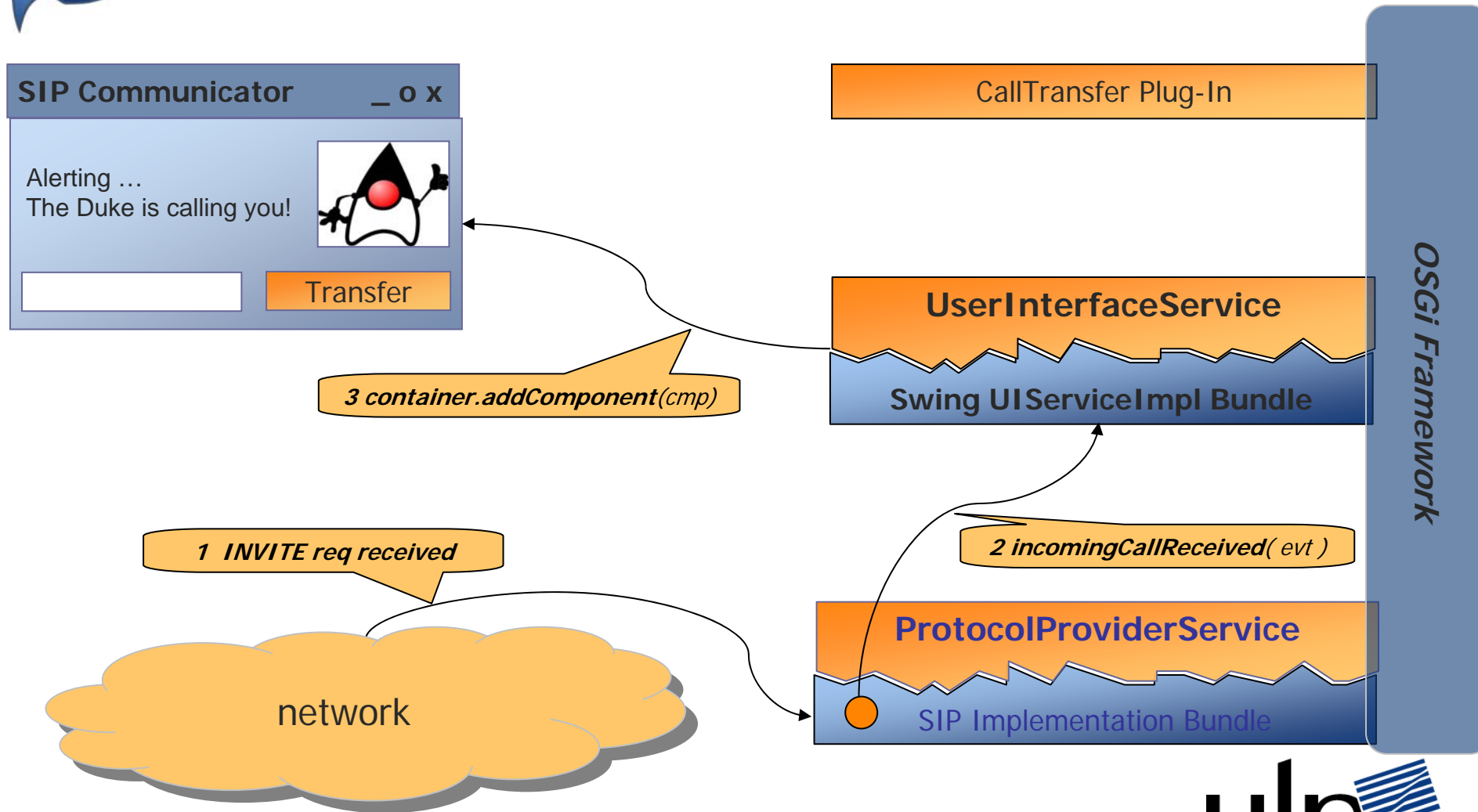
ProtocolProviderService
SIP Implementation Bundle

OSGi Framework



SIP Communicator 1.0 Architecture

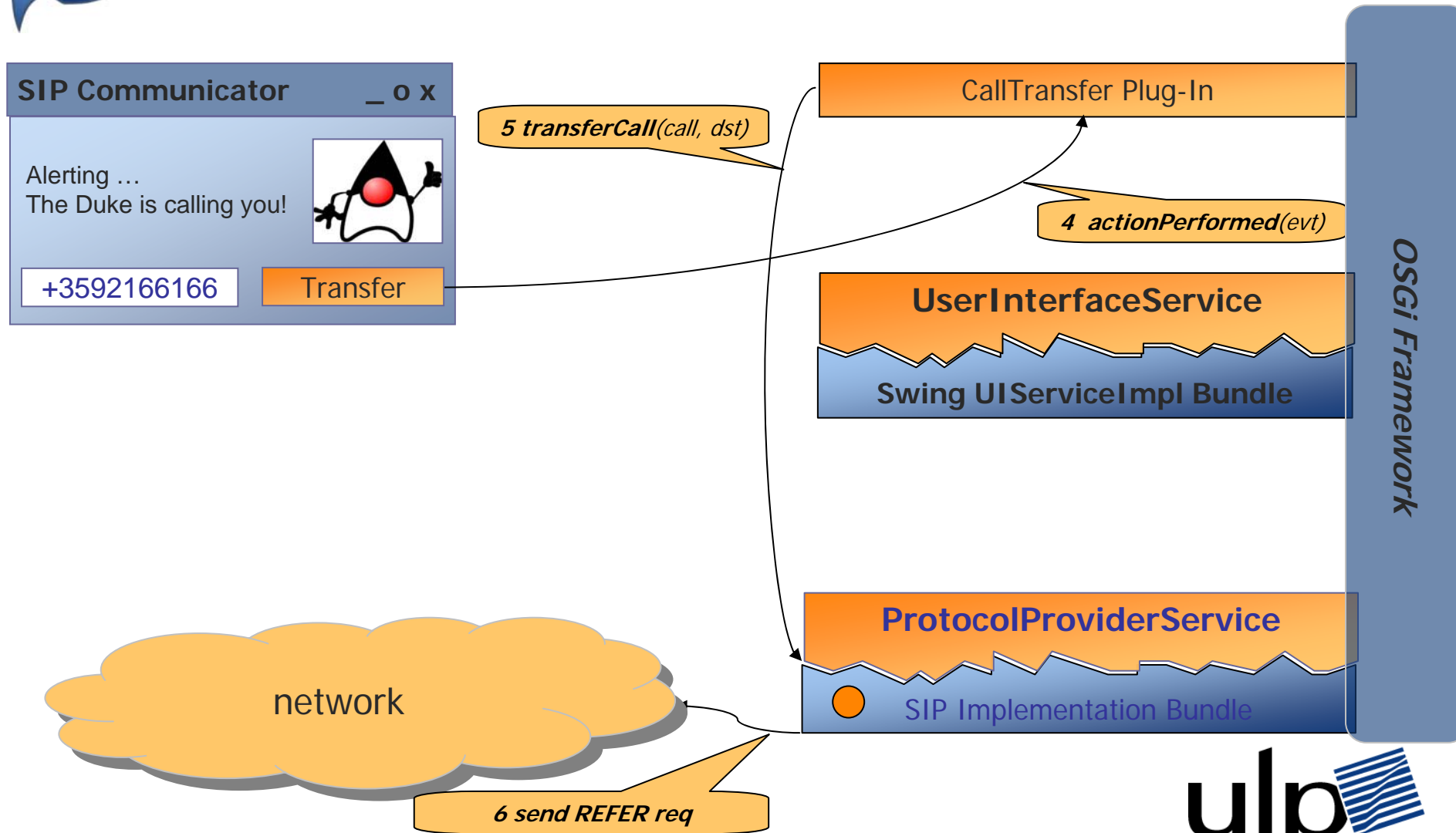
A Slightly More Complex Scenario (3)





SIP Communicator 1.0 Architecture

A Slightly More Complex Scenario (4)



<http://sip-communicator.org>



SIP Communicator tutorials:

How to create OSGi Services for SIP Communicator:

<http://www.sip-communicator.org/index.php/Documentation/CreatingServices>

How to write GUI plugins:

<http://www.sip-communicator.org/index.php/Documentation/HowToWriteGuiPlugins>

How to implement support for your favorite protocol in SIP Communicator:

<http://www.sip-communicator.org/index.php/Documentation/HowToImplementProtocols>

How to create and maintain a SIP Communicator installer:

<http://www.sip-communicator.org/index.php/Documentation/HowToBuildAnInstaller>