

Businessintegration GForge

Agenda

- What is GForge?
- Goal of the PoC.
- Kind of integration.
- Problems during integration
- Summary

What is GForge?

- GForge is a software collaboration tool which comprises of the following components:
 - Tracker (Bug, Issue etc.)
 - Forums, Tasks, Mailinglists, Surveys
 - Wiki
 - Version control

What is GForge?

- Pro's:
 - Simple to use (Web application)
 - Good overview about Bugs, Requests etc. of the project with email notification about Bugs etc.
 - Good documentation possibilities if you are using the Wiki-PlugIn.

What is GForge?

- Con's:
 - No real integration between components, no linking between Wiki, version control and tracking possible.
 - Complex installation process. Many components depend on each other.

What is GForge?

- Con's:
 - It is not possible to use an other database e.g. MySQL, Oracle etc. (no use of an DB abstraction layer).
 - No using of a template system like Smarty.

What is GForge?

- Con's:
 - No integrated fine grained permission handling for the Subversion repository.
 - Sometimes this is a knockout criterion.

What is GForge?

- Con's:
 - Within business integration some functions are a problem of privacy (e.g. peer ratings). They can not simply turned off. A little bit hands-on is needed.

What is GForge?

- Technically GForge comprises of:
 - PostgreSQL
 - PHP (4.X)
 - Apache Web-Server (or may be an other PHP capable Web-Server but with restrictions).
 - Subversion/CVS (ViewVC)

How to test GForge?

- If you just want to test GForge you can use an VMWare distribution.
 - A little drawback it's using an old Subversion (1.1.4).

Reasons to integrate

- Using of existing bug-, test-tracking tools etc.
- Transfer of GForge informations into other systems and v.v.
- Migration from test to production system.

Goal of the PoC

- Extract information out of GForge.
 - Login/Logout (authentication).
 - Get a list of projects.
 - Get a list of tracker entries etc.
 - Get track entries with attachments.
 - etc.

Goal of the PoC

- Put information into GForge.
 - Create track entries.
 - Create task entries.
 - Create new supplemental fields.
 - Create new Release entries.
 - etc.

How to integrate?

- Using the Web-Interface as the user of GForge does.
- Using the SOAP Interface of GForge

Web Interface Integration

- Pro's:
 - Very simple to do.
 - You can use many programming languages whereas the majority without special frameworks.

Web Interface Integration

- Con's:
 - Very time consuming, cause you have to simulate the user.
 - Every page flow etc.
 - Every input field must be filled with the particular values.

Web Interface Integration

- Con's:
 - More or less simple to put information into GForge, but more complex to extract information.
 - e.g. If the layout/field namings etc. will be changed the code has to be changed too.
 - Extracting mean to parse the HTML output of GForge to get the information.

The better Way

- GForge has an SAOP (1.1) Interface (WSDL), which can be used for integration with other systems.

<http://server/soap/index.php?wsdl>

The SOAP Integration

- How to implement the integration?
 - Can be done by hand.
 - Using a Framework.

The SOAP Integration

- How to implement the integration?
 - By hand:
 - Time consuming
 - Error prone.
 - ...

The SOAP Integration

- How to implement the integration?
 - Using a Framework:
 - Apache Axis Framework in Java for SAOP communication.

Apache Axis Framework

- Java Framework for Web-Service-Standard SOAP.
- Development of Client and Server is supported.
- The Release 1.3. has been used to do the PoC.

Apache Axis Framework

- Development tools like TCP monitor exist to support developers work e.g. for debugging.
- Automatic code generation based on WSDL etc.
- And much more.

First Steps with Axis

- Create the SOAP code from the WSDL:

```
WSDL2Java -t --output src --all URL
```

- This will create the complete SOAP access code.

First Steps with Axis

- `WSDL2Java -t --output src --all URL`
 - `-t Create`
 - Unit Tests
 - `--output`
 - Where to put the created source code.
 - `--all`
 - Generate code for all elements, even unreferenced ones.

Usable Code?

- After you have created the code you get running code, but not very „nice“ code.
- If you really like to use it, you should wrap it with your own code, to make it „usable“.

Created Code

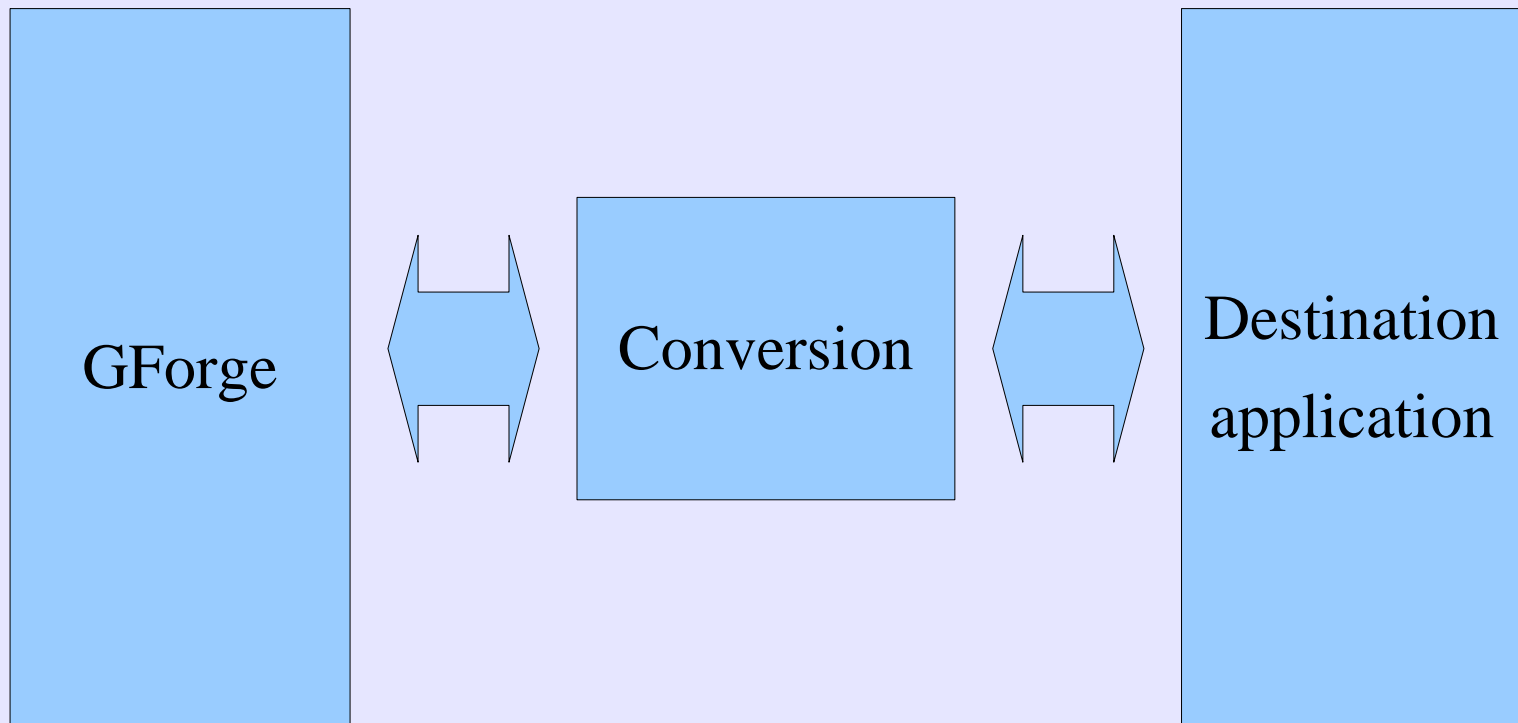
```
public interface GForgeAPIPortType extends java.rmi.Remote {  
    public java.lang.String login(java.lang.String userid, java.lang.String passwd);  
    public java.lang.String logout(java.lang.String session_ser);  
    public org.gforge.Group[] getGroups(java.lang.String session_ser, int[]  
        group_ids);  
    public org.gforge.Group[] getGroupsByName(java.lang.String session_ser,  
        java.lang.String[] group_names);  
    public java.lang.String[] getPublicProjectNames(java.lang.String session_ser);  
    ....
```

Hand written Code

- Good API to really work with GForge:

```
public class GfrogeAPI {  
    private String session_ser;  
  
    public void Login(String username, String password);  
    public void Logout();  
    public void ...  
    ....  
}
```

Integration Problems



Integration Problems

- How to handle fields in applications which do not exist in GForge?
 - Create supplemental fields, which is supported by GForge.

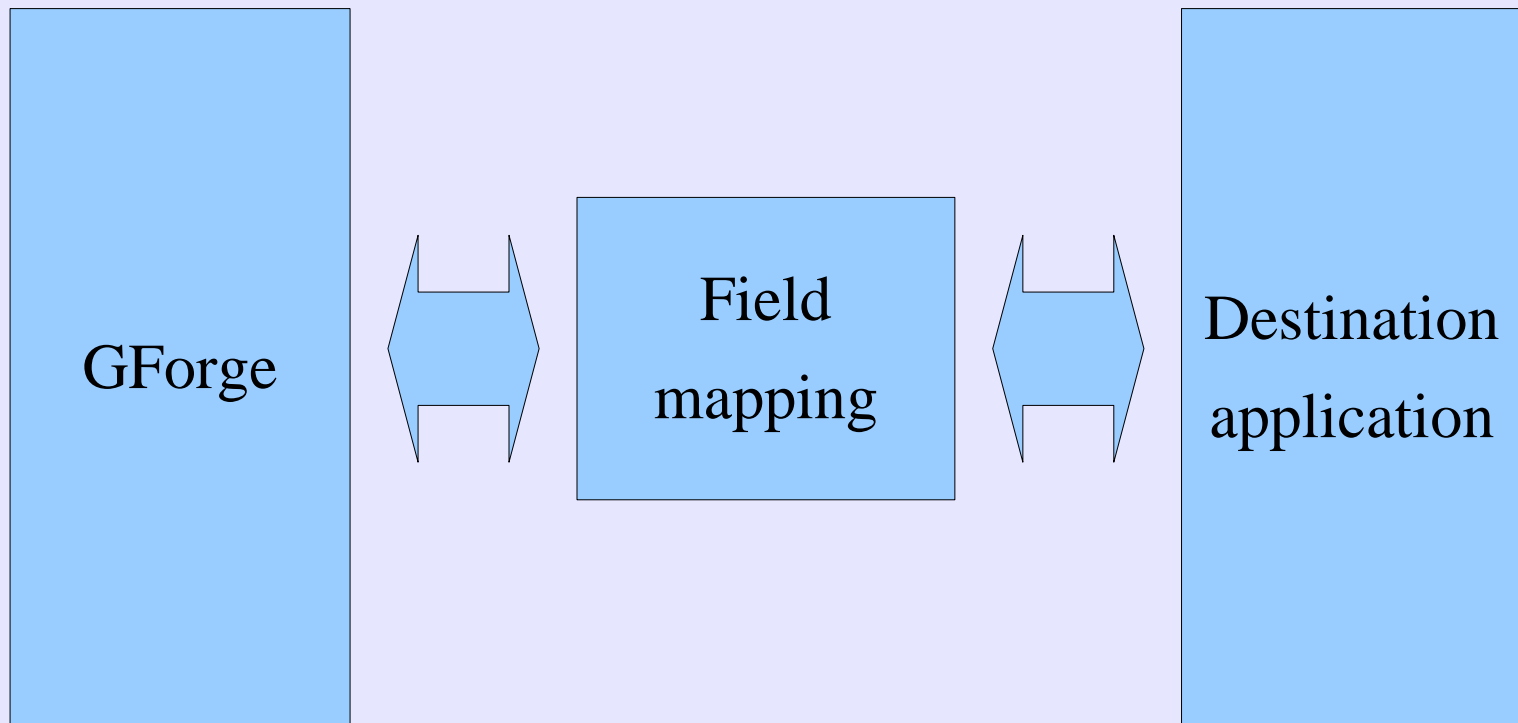
Integration Problems

- How to synchronize between GForge and other application?
 - What about the user information in GForge and the other application?
 - Must be stored in an other way.

Integration Problems

- How to synchronize between GForge and other application?
 - What about the Bug# in GForge and other applications?
 - The Bug# must be transferred to and from the application to get synchronized.

Integration Problems



Integration Problems

- How to synchronize between GForge and other application?
 - What about the Bug# of an other Bug-Tracking system?
 - The Bug# of the other tracking system can be put into an extra field.

Integration Problems

- How to synchronize between GForge and other application?
 - What about the different names of the field, which might have the same meaning?
 - You have to use a kind of mapping table etc.

Integration Problems

- If we need information of version control system(Subversion)? (Was not part of the PoC).
 - We could use the JavaSVN library of tmate.

Summary

- It had been proved:
 - CRUD (Create/Read/Update/Delete)
 - Tracker entries (Bugs, Support, Patches, Feature Requests).
 - Task entries
 - Documentation entries
 - Releases
 - etc.

Summary

- Effort for the PoC:
 - Within a week you can get a working programm which can do some of the needed procedures to do an integration.

Summary

- The SAOP way seemed to be the simplest and effective way to do an integration.
 - Stable against changes in layout etc.
 - It's a tool based development. You really don't need to write everything yourself.

Online Resources

- GForge Homepage

<http://www.gforge.org>

- Axis Framework

<http://ws.apache.org/axis/>

Questions?

- Thank for your attention.

Contact: phpcon@soebes.de