



Google Web Toolkit

a Technical Return on
Experience

www.kitry.lu

GWT in short...

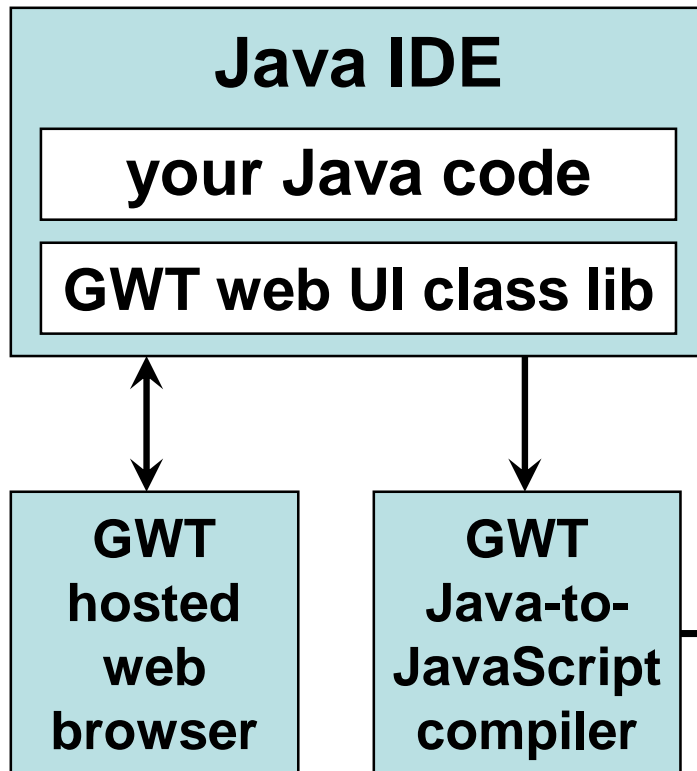


- Google / Open Source
- Java/Swing Development Framework
 - XHTML/AJAX compilation
- Key advantages
 - Java objects exchange server/Web interface
 - Cross-browser compatible
 - Productivity Gains (Java vs. JavaScript)
 - Reuse existing skills

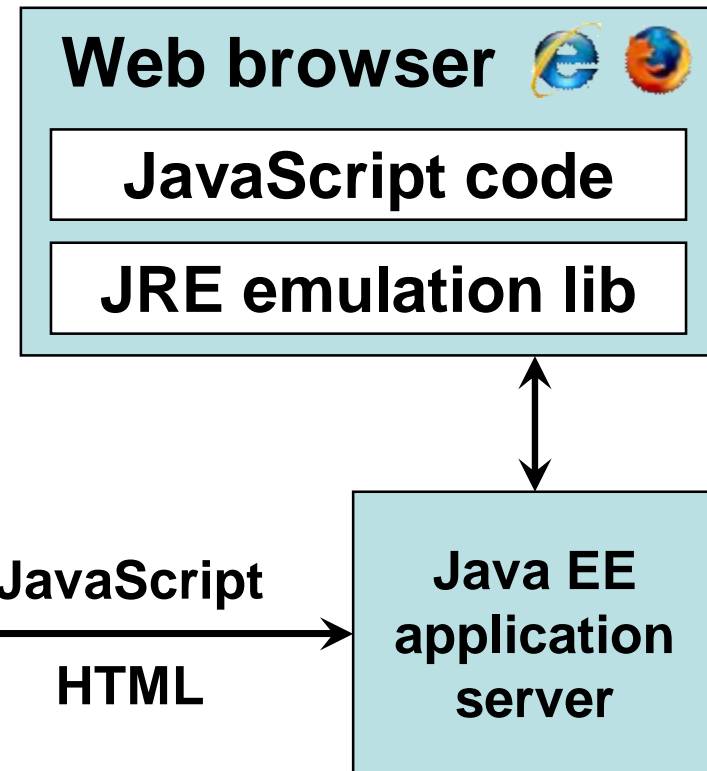


Architecture

Development



Production



Java language support



- GWT compiles Java source up to 1.4.2
- Data types : byte, char, short, int, long, float, double, Object, String, arrays (but long à JS floating point)
- Exceptions : try, catch, finally, user-defined exceptions
- Multithreading : not supported
- Reflection : only **GWT.getTypeName(Object)**
- Finalization : not supported
- Floating-point : browser-dependent

Runtime library support



JRE emulation in JavaScript :

- supports only a subset of `java.lang` & `java.util`
- syntax of Java regular expressions is similar but not identical to JavaScript
- JRE emulation support is checked in hosted mode `à` run early and often !

RPC : the server side



```
public interface MyService
    extends RemoteService {

    public String myMethod(String param);
}
```



RPC : the server side (2)

```

public class MyServiceImpl
    extends RemoteServiceServlet
    implements MyService {

    public String myMethod(String param) {
        // Do something here on the server.
        return someResult;
    }
}

```



RPC : the server side (3)

```
public interface MyServiceAsync {  
    public void myMethod(String param,  
                          AsyncCallback cb);  
}
```

RPC : the client side



```

AsyncCallback callback = new AsyncCallback() {
    public void onSuccess(Object result) {
        // do some UI updates to show success
    }
    public void onFailure(Throwable caught) {
        // do some UI updates to show failure
    }
};

```

RPC : the client side (2)



```

MyServiceAsync service =
    (MyServiceAsync) GWT.create(MyService.class);

((ServiceDefTarget)service).setServiceEntryPoint(
    GWT.getModuleBaseURL() + "aSmartName");

service.myMethod(someString, callback);
    
```

Entry points





The IC/REM experience

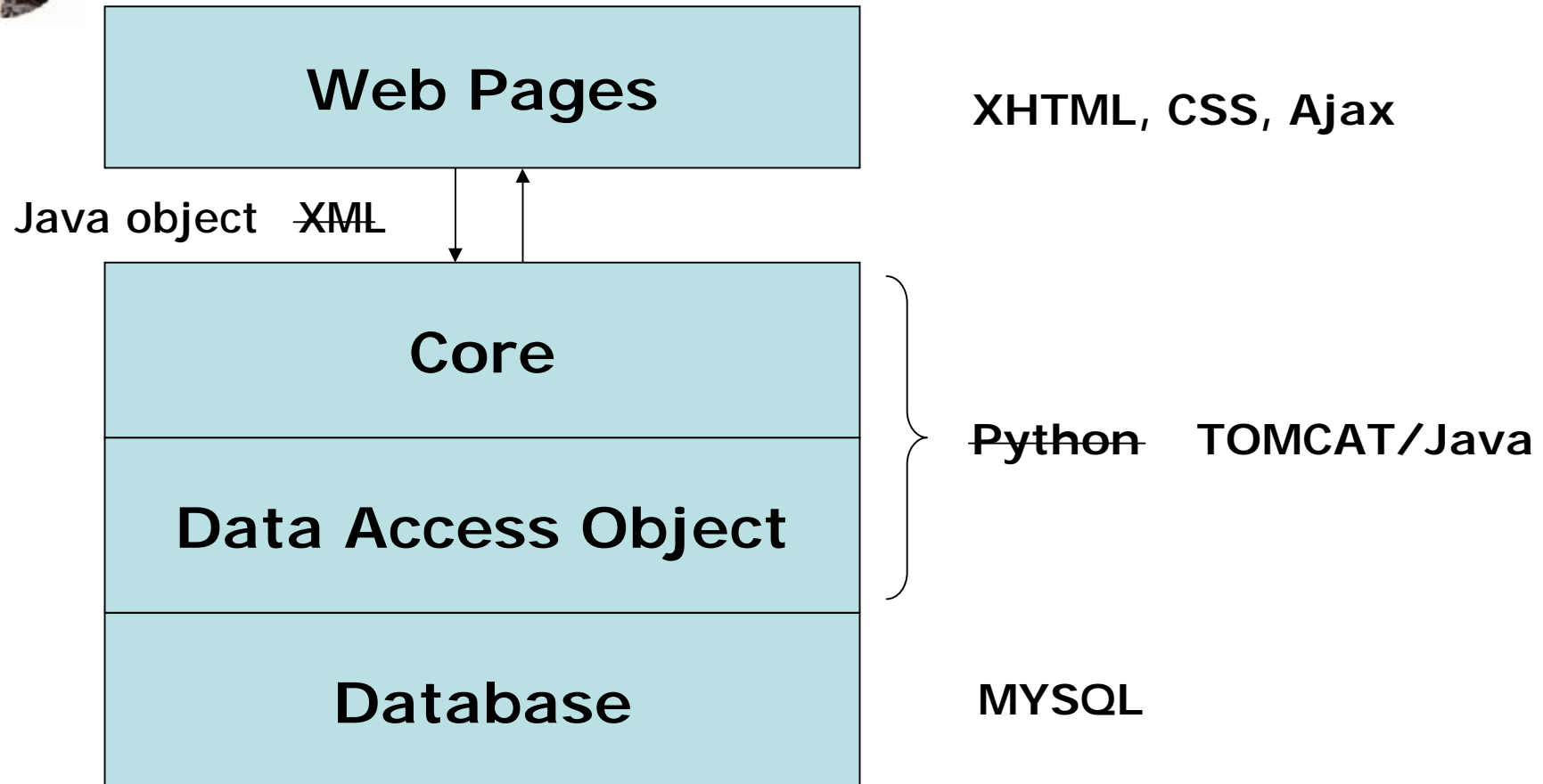
www.kitry.lu

IC/REM



- Improvement Control
 - Regulation and Equipment Management
- Business application
- LuxControl as partner
- Kitry: 2006 / Lotus Domino (R.A.D) / 15 d
- E.S.I.A.L: Web & OpenSource
 - 2 school years (2 * 80 j.h)

Architecture evolution



Screen shot



	IC/REM Improvement Control / Regulation and Equipment Management Accueil - Déconnexion	
<p>Contrôles périodiques</p> <p>Configuration</p> <p>Sites Appareils & Contrôles Rapports Responsables Organismes de contrôle</p>	Organismes de contrôle Création d'un nouvel organisme Nom de l'organisme <input type="text"/> Type d'organisme <input type="text" value="Selectionnez un type d'organisme.."/> Personne de contact <input type="text"/> Téléphone <input type="text"/> Fax <input type="text"/> E-mail <input type="text"/> <input type="button" value="Créer"/> <input type="button" value="Retour"/>	
		



Conclusions



www.kitry.lu



Metrics

Development time (DAO + GUI)	GWT + Python (2006/2007)	"pure" GWT (2007/2008)
Configuration module	80 days (XML grammar)	10 days (rewrite existing module)
Utilisation Module	Not enough time	70 days



Conclusions

- Learning curve
- Existing frameworks integration (DAO Pattern, Hibernate)
- Reuse existing competencies
- Quick results
- Pas 100% cross-browser/multimédia ?

- Automatic tests with "Selenium"
- **Off-Line Web application: Google Gears**