Rapid Application Development for GWT App Engine using EMF

Ed Merks
Cloudsmith, itemis
Accelerating up the Learning Curve

• Learning a new technology is always intimidating
  – New things generally seem complex and unintuitive
    • Why are all these things needed?
    • Why is it done like this?
    • Why isn’t it done in this old familiar way?
• Good documentation helps, but real developers don’t read it; they just dive right in
• Examples are essential
  – They provide a working starting point illustrating how it all hangs together
Swiss Army Knife

- Expressive languages
- Powerful frameworks
- Excellent tools
- A wealth of documentation
  - Introductory overviews
  - Reference material
  - Tutorials
  - Samples
- Helpful forums
Google Web Toolkit

- “Google Web Toolkit (GWT) is a development toolkit for building and optimizing complex browser-based applications.”
- It’s based on Java!
  - Reuse existing language skills
  - Reuse existing tool skills
  - Reuse some existing libraries
  - Learn new libraries
  - Learn new programming paradigms
    - Asynchronous client server communication
GWT for Eclipse

• GWT provides fantastic integration with Eclipse’s Java Development Tools
• Getting started is so easy!
  – Create a project
  – Launch it
  – Load it the browser
  – Debug client and server logic interactively
  – Deploy it to App Engine

• Demo...
Now What?

• It’s little better than a “Hello World” example...
  – How to implement a real application?
  – How to design and store the data?
  – How to access the data asynchronously?
  – How to update the browser’s views based on asynchronous arrival of data?

• There are more interesting examples...

• But it would be better to generate an application based on data from the programmer’s domain of interest
Eclipse Modeling Framework

- Provides a simple way of describing data models
- Generates fully functional applications from those models
  - Eclipse Integrated Development Environment (IDE)
  - Eclipse Rich Client Platform (RCP)
  - Eclipse Rich Ajax Platform (RAP)
  - GWT

- Sports a powerful core runtime based on high performance abstract reflection
The Advantages of EMF

• Focus on designing the essential aspects of the data
  – Produce simple natural APIs rather than stilted
denormalized forms
• Minimize time spent on repetitive coding
• Data instances are GWT serializeable
  – They can be used in service APIs
• EMF’s Resource framework is based on
  Representational State Transfer, i.e., REST
  – Transparent integration with App Engine datastore
But I Hate Modeling!

• No you don’t
  – You model all the time, but call it something else
  – You just hate UML and equate the two

• I’ll show you what I mean
  – Demo time...
Conclusions

- GWT is awesome
- EMF makes it more awesome
  - Generates an interesting fully functional application based on your specific domain model
  - Let’s you focus on designing a natural data model
  - Supports REST
    - Transparent integration with datastore
    - A plastic alternative/complement to service APIs
  - Gets your specialized application up and running rapidly
References

- [http://www.eclipse.org/emf](http://www.eclipse.org/emf)
- [http://code.google.com/webtoolkit/](http://code.google.com/webtoolkit/)
- [https://appengine.google.com/](https://appengine.google.com/)