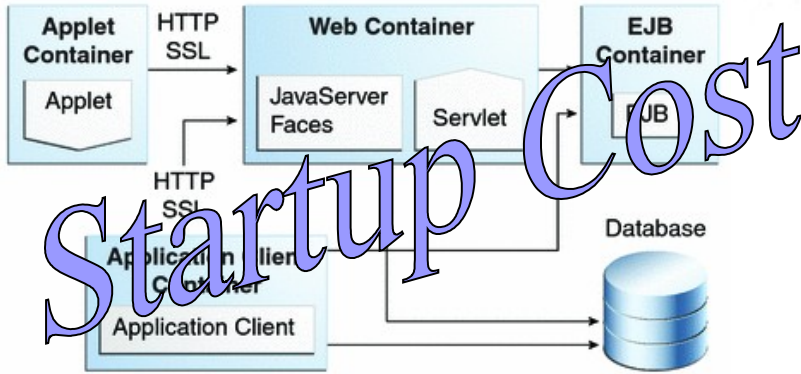




Forging a Bond between Eclipse and the Command-line

Daunting? What Do You Mean?



Startup Cost

Details

| | | |
|-------------------------------------|------------------|----------------|
| Application Client Container | Java Persistence | Java SE |
| | Management | |
| | WS Metadata | |
| | Web Services | |
| | Application | |
| Application | JAX-WS | SAAJ |
| | JAX-RPC | |
| | | |

| | | | |
|----------------------|------------------|----------------|-------------|
| Web Container | JSR 330 | Java SE | |
| | Interceptors | | |
| | Managed Beans | | |
| | JSR 299 | | |
| | Bean Validation | | |
| | EJB Lite | | |
| | EL | | |
| | JavaMail | | |
| | JSP | | |
| | Connectors | | |
| Servlet | Java Persistence | Java SE | |
| | JMS | | |
| | Management | | |
| | WS Metadata | | |
| | Web Services | | |
| | JACC | | |
| | JSP | | |
| | JAX-WS | | SAAJ |
| | JAX-RPC | | |

| | | | |
|----------------------|------------------|----------------|-------------|
| EJB Container | JSR 330 | Java SE | |
| | Interceptors | | |
| | Managed Beans | | |
| | JSR 299 | | |
| | Bean Validation | | |
| | EJB | | |
| | Java Persistence | | |
| | JTA | | |
| | Connectors | | |
| | JMS | | |
| EJB | Management | Java SE | |
| | WS Management | | |
| | Web Services | | |
| | JACC | | |
| | JASPIC | | |
| | JAXR | | |
| | JAX-RS | | |
| | JAX-WS | | SAAJ |
| | JAX-RPC | | |

Integration

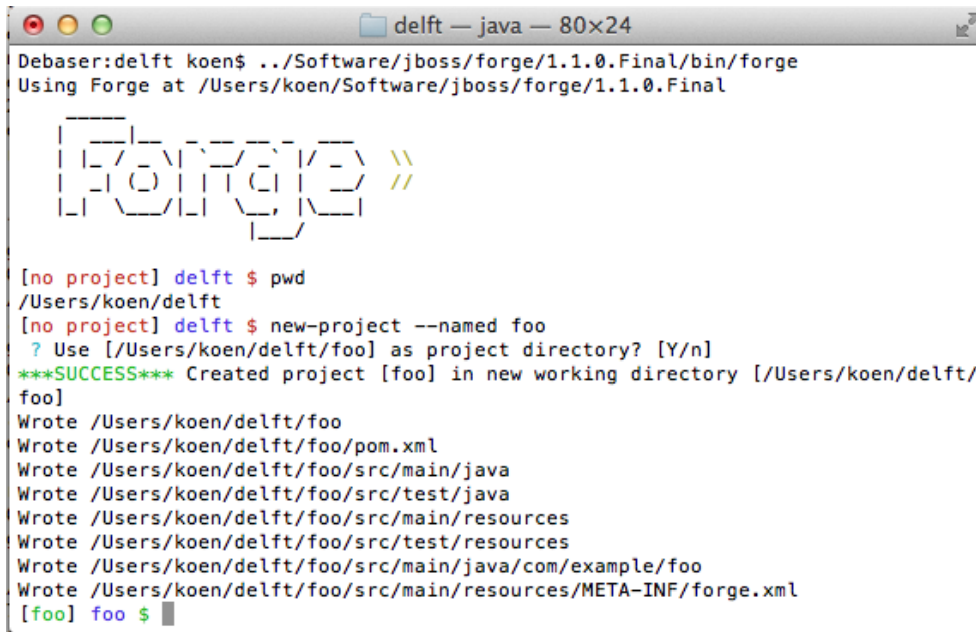
Gotchas

<http://jboss.org/forged>



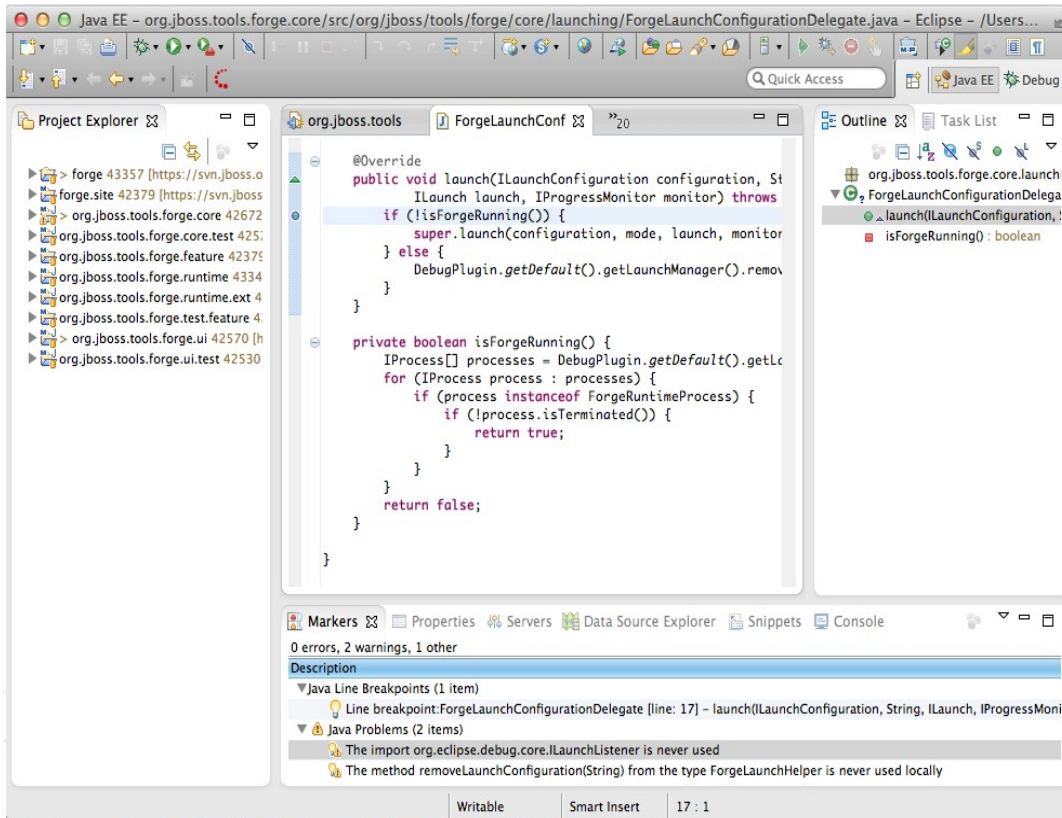
Carve out a project, work it into shape

Real Programmers Do Command-line Development

A terminal window titled 'delft — java — 80x24' showing the execution of the Forge command-line tool. The user runs './Software/jboss/forge/1.1.0.Final/bin/forge' and 'Using Forge at /Users/koen/Software/jboss/forge/1.1.0.Final'. The Forge logo is displayed. The user then runs 'pwd' and 'new-project --named foo'. A prompt asks 'Use [/Users/koen/delft/foo] as project directory? [Y/n]'. The user responds 'Y', and the tool creates a project 'foo' in the directory '/Users/koen/delft/foo'. It then writes several files: 'pom.xml', 'src/main/java', 'src/test/java', 'src/main/resources', 'src/test/resources', 'src/main/java/com/example/foo', and 'src/main/resources/META-INF/forge.xml'. The prompt changes to '[foo] foo \$'.

- Typing is faster than Clicking!
- Tab completion!
- Scripts!

But, I Want My IDE !



- Incremental builds!
- Workspace navigation!
- Outline view!
- Debugging!
- Form based editing!

Have Your Cake and Eat It!

The screenshot displays the Eclipse IDE interface. The main editor shows the `Customer.java` file with the following code:

```
package com.example.foo.model;

import javax.persistence.Entity;
import java.io.Serializable;
import javax.persistence.Id;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Column;
import javax.persistence.Version;
import java.lang.Override;

@Entity
public class Customer implements Serializable
{
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    @Column(name = "id", updatable = false, nullable = false)
    Long id = null;

    @Version
    private @Column(name = "version")
    int version = 0;

    @Column
    private String firstName;

    public Long getId()
    {
        ...
    }
}
```

The Project Explorer on the left shows the project structure, including `src/main/java`, `src/main/resources`, `src/test/java`, and `src/test/resources`. The Outline view on the right shows the class structure of `Customer` with fields `id` and `version`, and methods `getId()`, `setId(Long)`, `getVersion()`, `setVersion(int)`, `equals(Object)`, `hashCode()`, `getFirstName()`, `setFirstName(String)`, and `toString()`.

The Forge Console at the bottom shows the output of the `fi` command:

```
Wrote /Users/koen/delft/workspace/foo/src/main/java/com/example/foo/model/Customer.java
[foo] Customer.java $ fi
fingerprint find field
[foo] Customer.java $ field
oneToMany manyToMany oneToOne int string temporal number boolean long
manyToOne custom
[foo] Customer.java $ field string --named firstName
Added field to com.example.foo.model.Customer: @Column private String firstName;

Wrote /Users/koen/delft/workspace/foo/src/main/java/com/example/foo/model/Customer.java
[foo] Customer.java $
```

Questions?



<http://jboss.org/forged>