

What's my Modularity



@bobbpaulin

Bob Paulin

Email

bob@bobbypaulin.com Or
bob@apache.org

Twitter

[@bobbypaulin](https://twitter.com/bobbypaulin)

GitHub

<https://github.com/bobbypaulin>

CJUG

Meetup

<http://www.meetup.com/ChicagoJUG/>

Twitter

[@cjug](https://twitter.com/cjug)

Devoxx4Kids Chicagoland

Website

<http://devoxx4kids.org/usa/chicago/>
<http://www.meetup.com/Devoxx4Kids-Chicagoland-Area/>

Twitter

[@devoxx4kids_chi](https://twitter.com/devoxx4kids_chi)

Off Heap Java Pubhouse

Website

<http://www.javaoffheap.com/>
<http://www.javapubhouse.com/>

Twitter

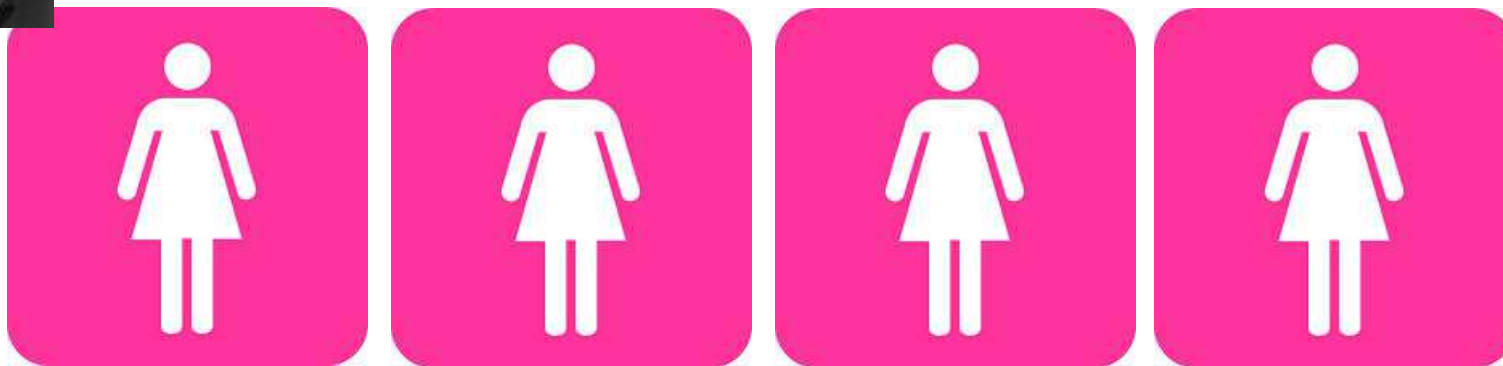
[@offheap](https://twitter.com/offheap)



<http://www2.pictures.zimbio.com/gi/Super+Rugby+Rd+12+Rebels+v+Reds+-eWcZZxaaJVI.jpg>



“The bearing of a child takes nine months, no matter how many women are assigned”
- Fred Brooks

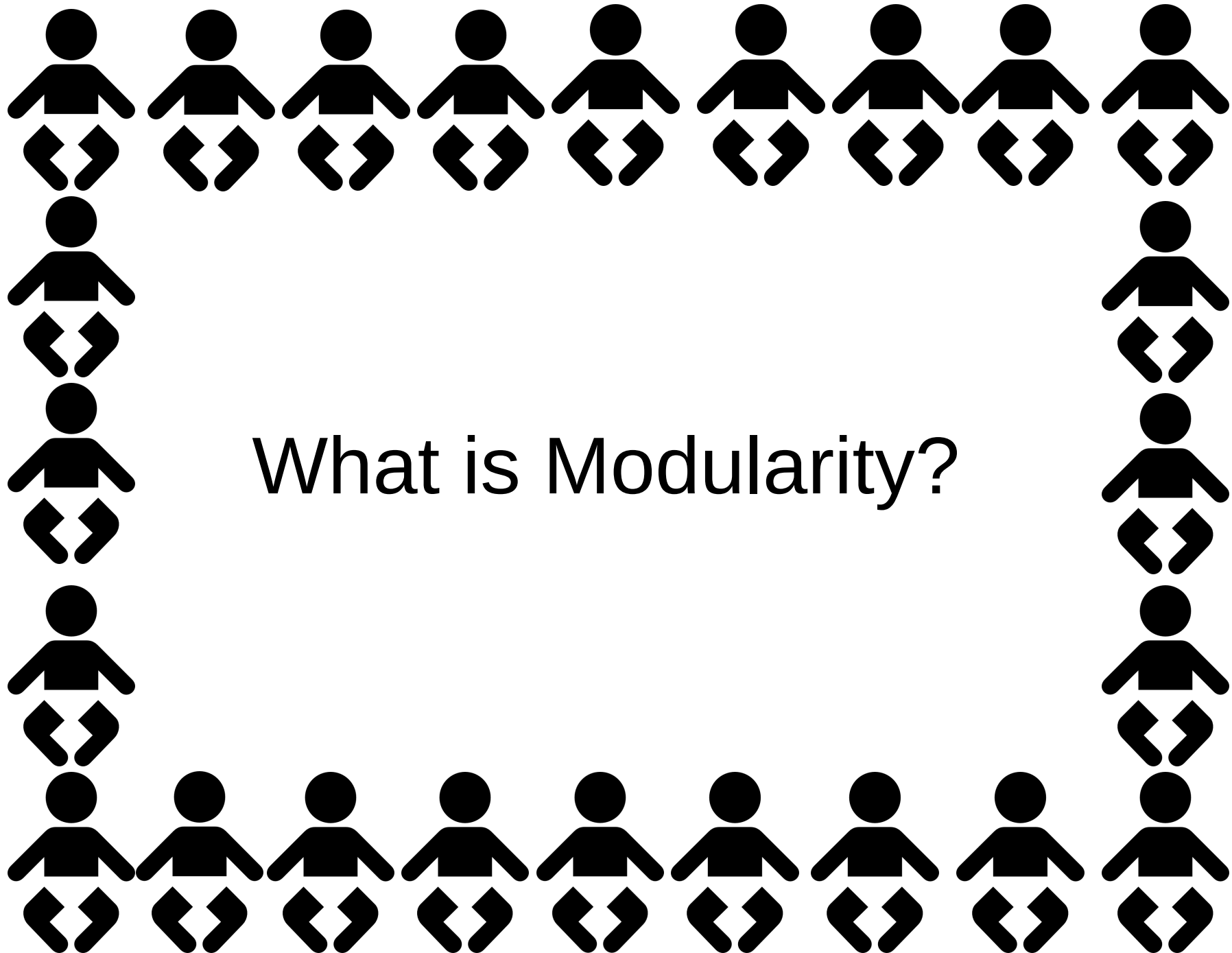


But could we make 9 babies in 9
Months....



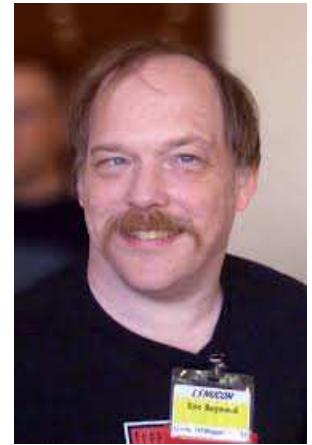


http://www.smh.com.au/ffximage/2005/07/17/wrestling_wideweb__430x250.jpg



What is Modularity?

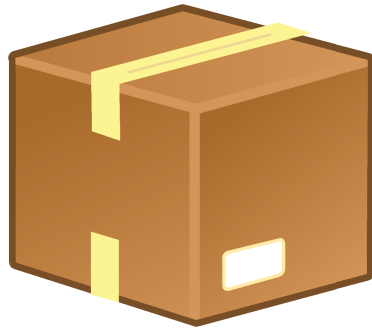
UNIX Rule of Modularity



Developers should build a program out of simple parts connected by well defined interfaces, so problems are local, and parts of the program can be replaced in future versions to support new features. This rule aims to save time on debugging code that is complex, long, and unreadable.

- Eric Raymond

What makes a module?

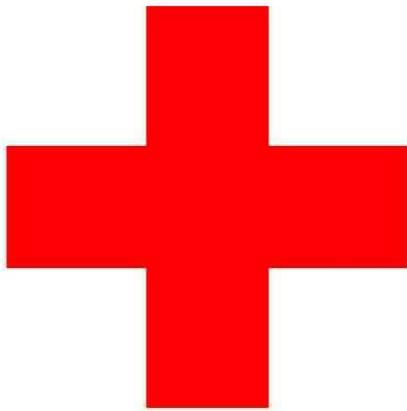


Testable
Deployable
Manageable
Natively Reusable
Composable
Stateless



Java Application Architecture
Kirk Knoernschild

Designing Modules



The modularizations include the design decisions which must be made before the work on independent modules can begin.

- David Parnas



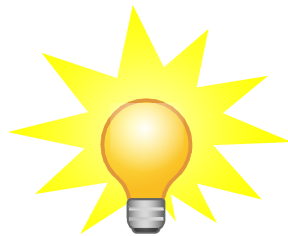
Changeability



Independent Development

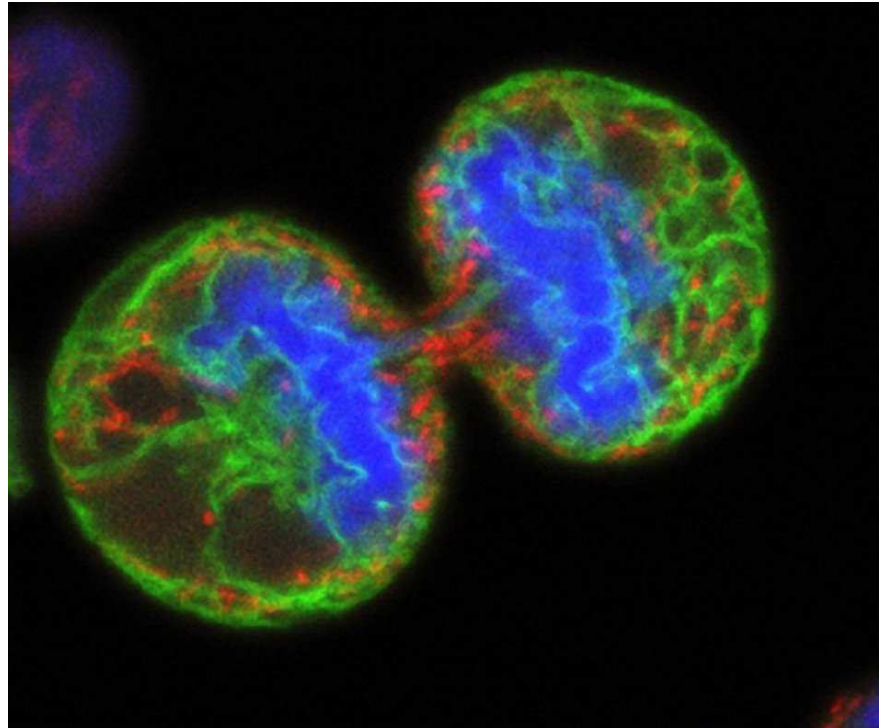


Comprehensibility



Don't like to read?

Evolvable



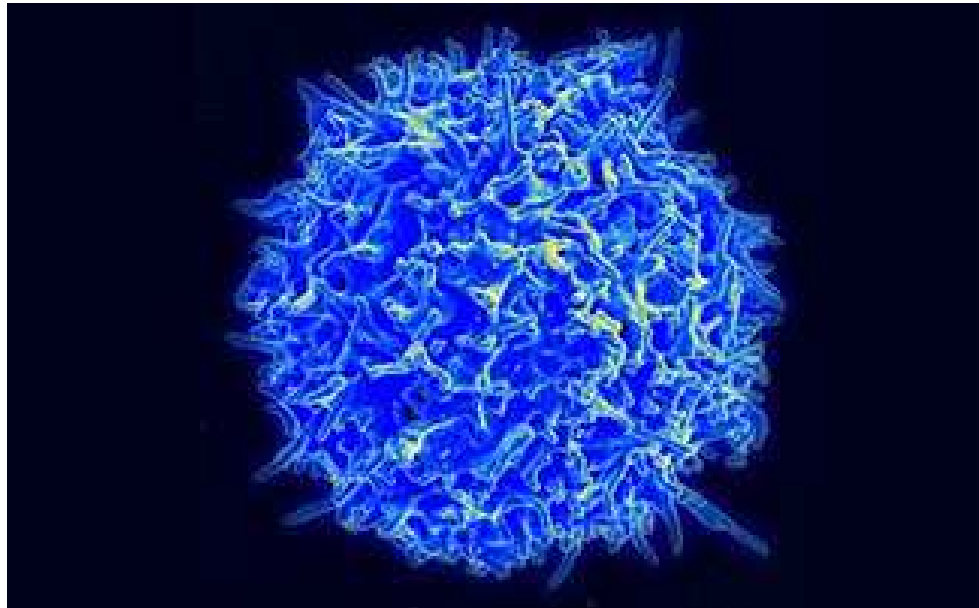
<http://a66c7b.medialib.glogster.com/media/0c/0cdf43b19a5b5ae9f51f46208517c80434e08093c52c679bb996022c27d9e0b8/cell-division-background.jpg>

Transplantable



<http://www.thinkgeek.com/product/e72e/>

Containable



<https://www.niaid.nih.gov/SiteCollectionImages/topics/immuneyesystem/humanTCellSlide.jpg>

Java Out of the Box



What's Missing in Java?



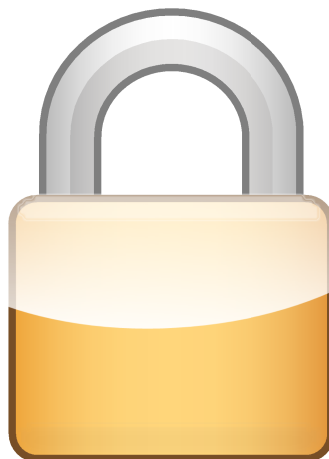
Versioning

1.0.0

Protection for Internal Classes

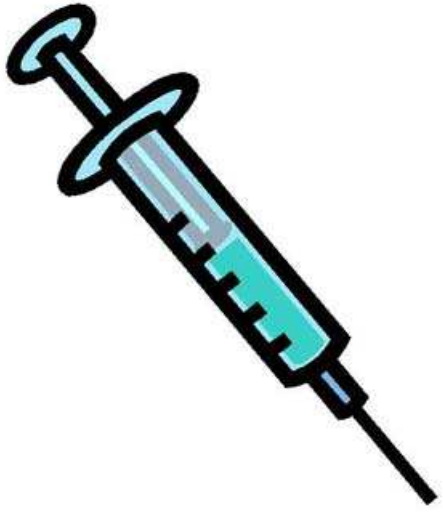


Security from unwanted classes



How Do We Deal?





Dependency Injection

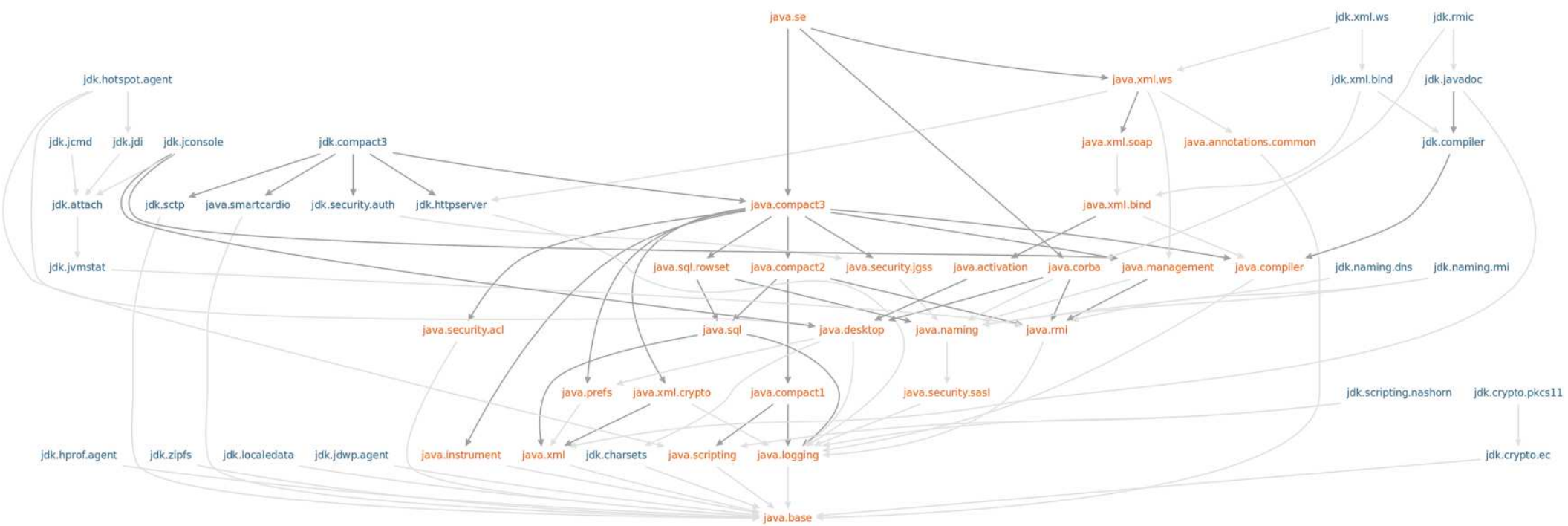
Dependency Management

A good start but....

It doesn't actually stop us from doing
bad things

Project Jigsaw

- JEP 200: The Modular JDK
- JEP 201: Modular Source Code
- JEP 220: Modular Run-Time Images
- JSR 376: Java Platform Module System



Early Release Stuff working!

<https://jdk9.java.net/jigsaw/>

Inside a Module

module-info.java

```
module java.security.sasl {  
    requires java.logging;  
  
    exports javax.security.sasl;  
    exports com.sun.security.sasl.util to  
        jdk.security.jgss;  
    provides java.security.Provider with  
        com.sun.security.sasl.Provider;  
}
```

No more JRE vs JDK Distinction

jlink

Build your own minimalist runtime!

Promised but not yet delivered



<https://www.nlm.nih.gov/medlineplus/magazine/issues/fall10/articles/fall10pg5.html>



OSGi



OSGi provides runtime modularity

Inside a Bundle

Manifest-Version: 1.0

Bundle-SymbolicName: com.bobpaulin.modularity.control.module

Bundle-Version: 1.0.0

Bundle-Name: control-module

Bundle-ManifestVersion: 2

Bnd-LastModified: 1455719427000

Private-Package: com.bobpaulin.modularity.control.internal

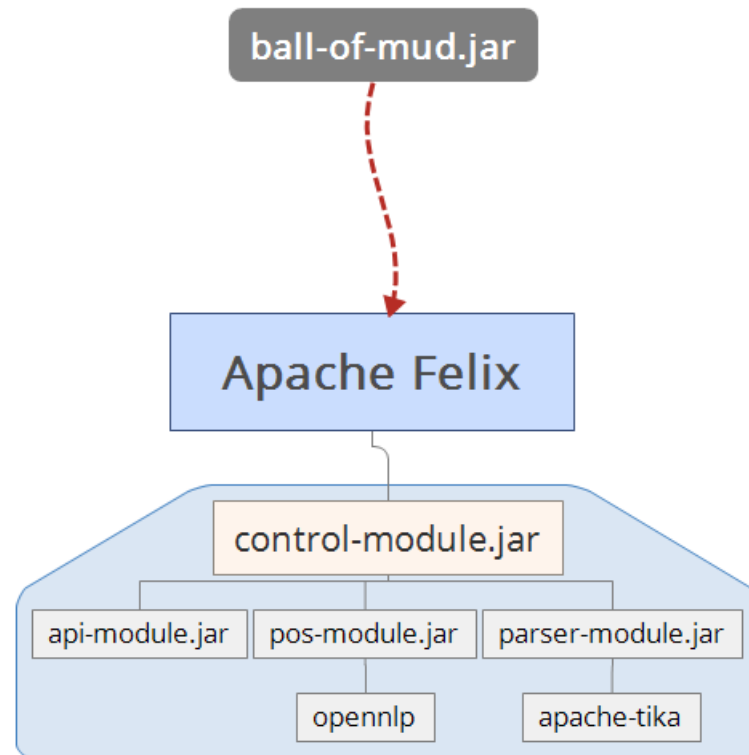
Bundle-Activator: com.bobpaulin.modularity.control.internal.Activator

Import-Package: com.bobpaulin.modularity.api,org.osgi.framework;version
n="[1.7,2)"

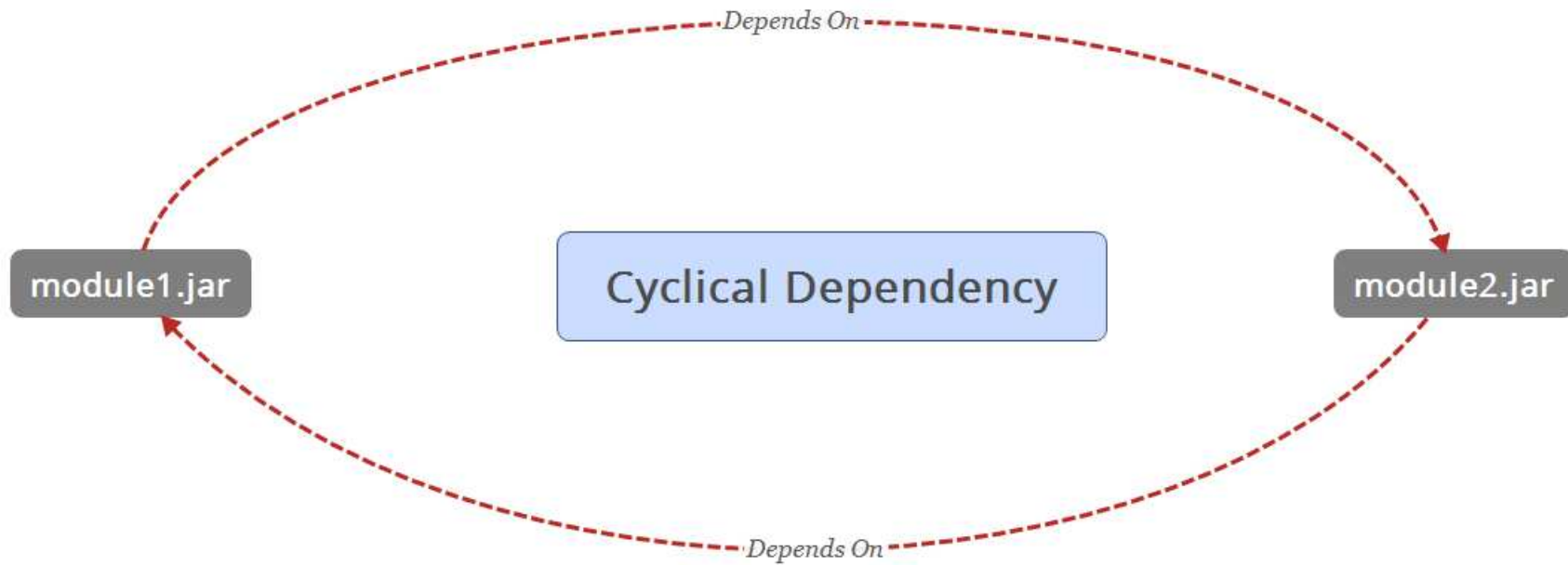
Created-By: 1.8.0_45 (Oracle Corporation)

Tool: Bnd-2.1.0.20130426-122213

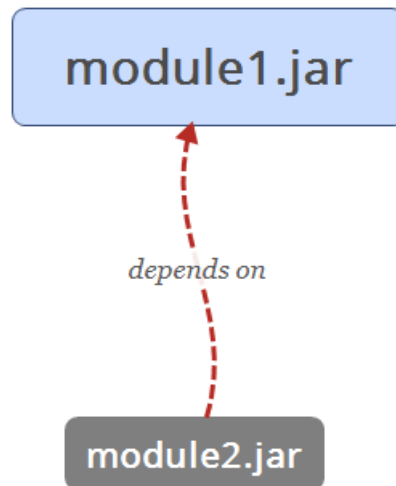
Converting the Ball Of Mud to OSGi



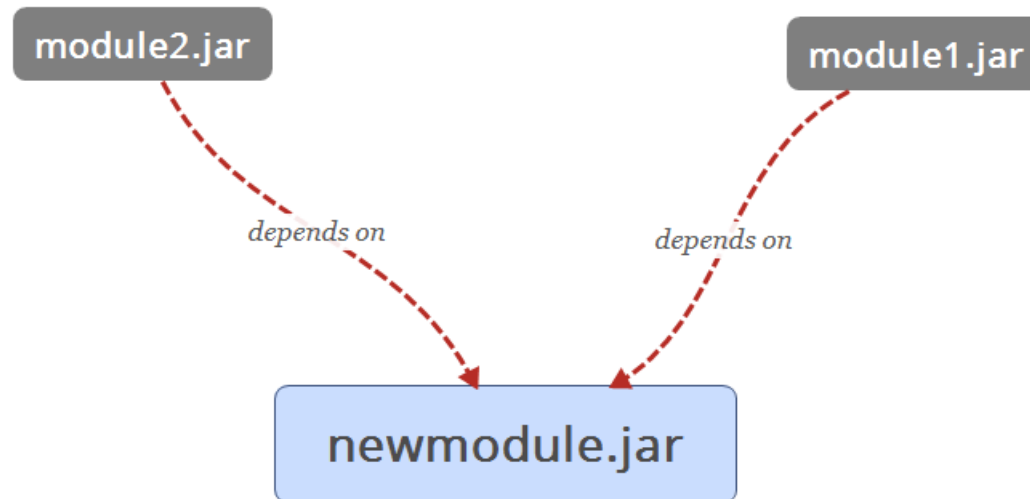
Break Cyclical Dependencies



Consolidate cycle to one Jar



Consolidate cycle to separate Jar



Create BundleActivator classes

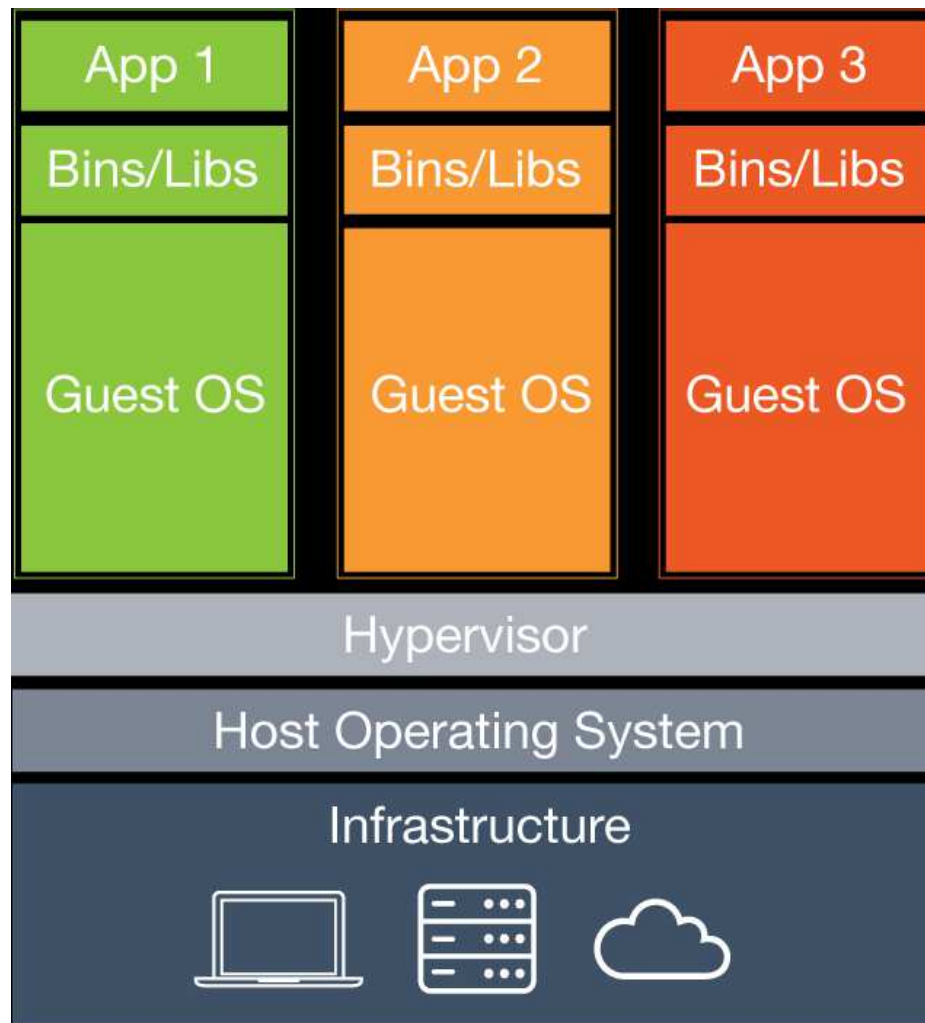
Register Services.

Import Services

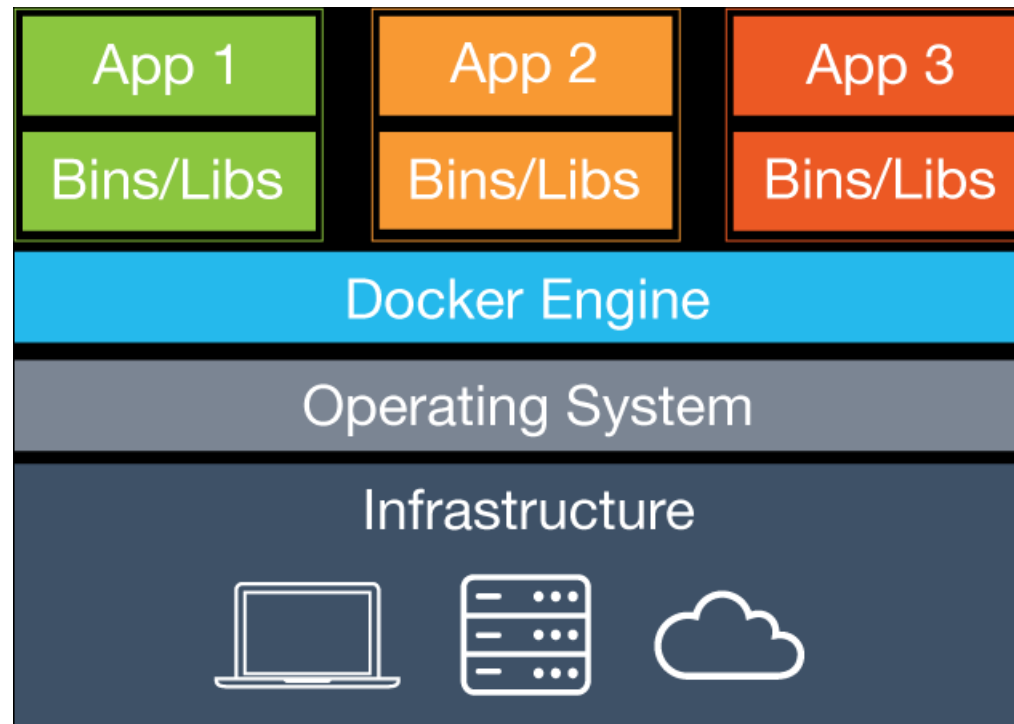


docker

Virtualization

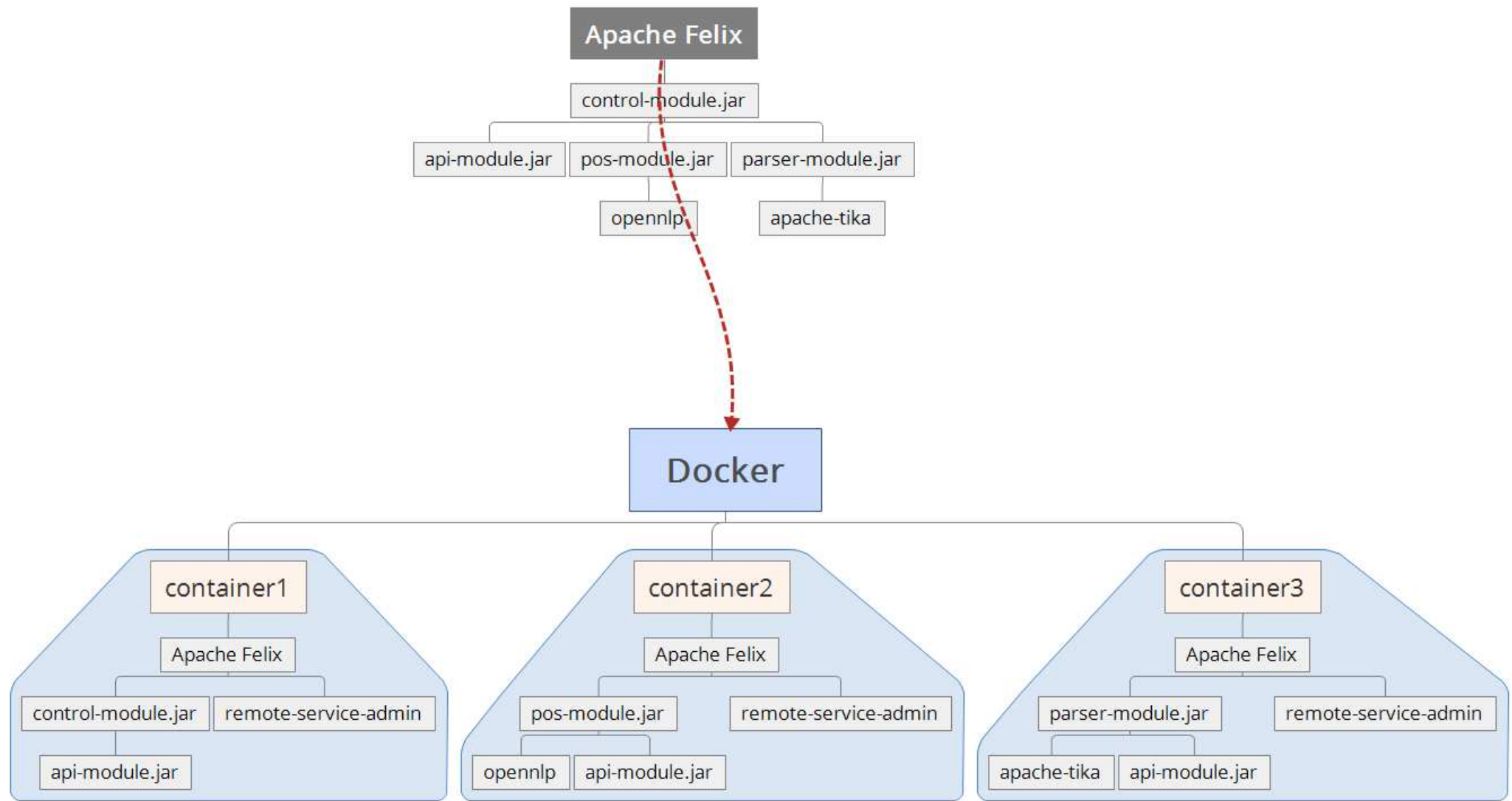


Docker



Demo

Converting single JVM OSGi to Containers

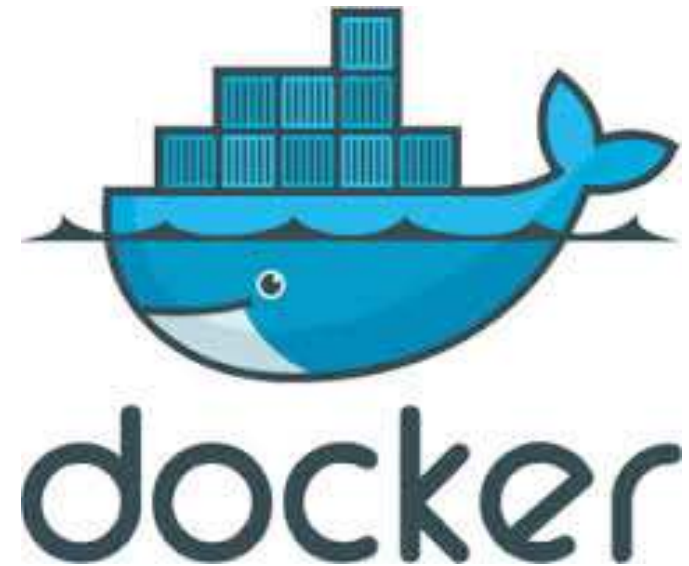


Add Docker File Generation to Gradle

Add RemoteServiceAdmin

Build Docker Images and Start

When to use....



Parting thoughts for good modular
code....

No cyclical dependencies

Know how your modules need to
scale

Keep External Interfaces simple

Just in case you do like to read...

References

Parnas, D. L. On the criteria to be used in decomposing systems into modules. Commun. ACM 15, 12 (December 1972), 1053-1058.

DOI=10.1145/361598.361623 <http://doi.acm.org/10.1145/361598.361623>

Knoernschild, Kirk. Java Application Architecture: Modularity Patterns with Examples Using OSGi Addison-Wesley Professional 2012

Brooks, Fred. The Mythical Man-Month: Essays on Software Engineering Addison-Wesley 1975

Raymond, Eric S. The Art of Unix Programming Addison-Wesley 2003

Perlis, Alan. Epigrams in Programming ACM SIGPLAN Notices 17 (September 1982) <http://www.cs.yale.edu/homes/perlis-alan/quotes.html>

Links

- Sonar <http://www.sonarqube.org/>
- Sonar Source Dashboard <http://nemo.sonarqube.org/dashboard/index>
- Docker <https://www.docker.com/>
- Apache Felix <http://felix.apache.org/>
- Amdatu Remote <http://www.amdatu.org/components/remote.html>
- Gradle <https://gradle.org/>
- Gradle Templates <https://github.com/townsfolk/gradle-templates>
- Apache Tika <https://tika.apache.org/>
- Tesseract OCR <https://github.com/tesseract-ocr/tesseract>
- Project Jigsaw <http://openjdk.java.net/projects/jigsaw/>

Modularity is not required



The Walking Dead AMC

Survival is not mandatory