



Modularity and API Design

Jaroslav Tulach (@JaroslavTulach)

Lukas Stadler

Thomas Wuerthinger (@thomaswue)

January 2015
Johannes Kepler University Linz, Austria
<http://www.jku.at>

Goals


- Why is modularity essential?
 - How to scale the development of a large software system?
 - Why are APIs like stars?
 - How to design an API that is extensible without breaking backwards compatibility?
 - What modularity and API design concepts are relevant in languages other than Java?
 - What are modularity concepts in NetBeans?
-
- **Project**

Don't reinvent the wheel!




PHASE 1: REINVENT THE WHEEL

YES, THERE ARE A LOT OF LIBRARIES I CAN USE, BUT NONE OF THEM QUITE FIT MY NEEDS.




PHASE 2: IN BETWEEN

THIS DATABASE WRAPPER I WROTE WORKS FOR EVERY PROJECT I'LL EVER TOUCH! THE REST HOWEVER MUST BE CUSTOM.



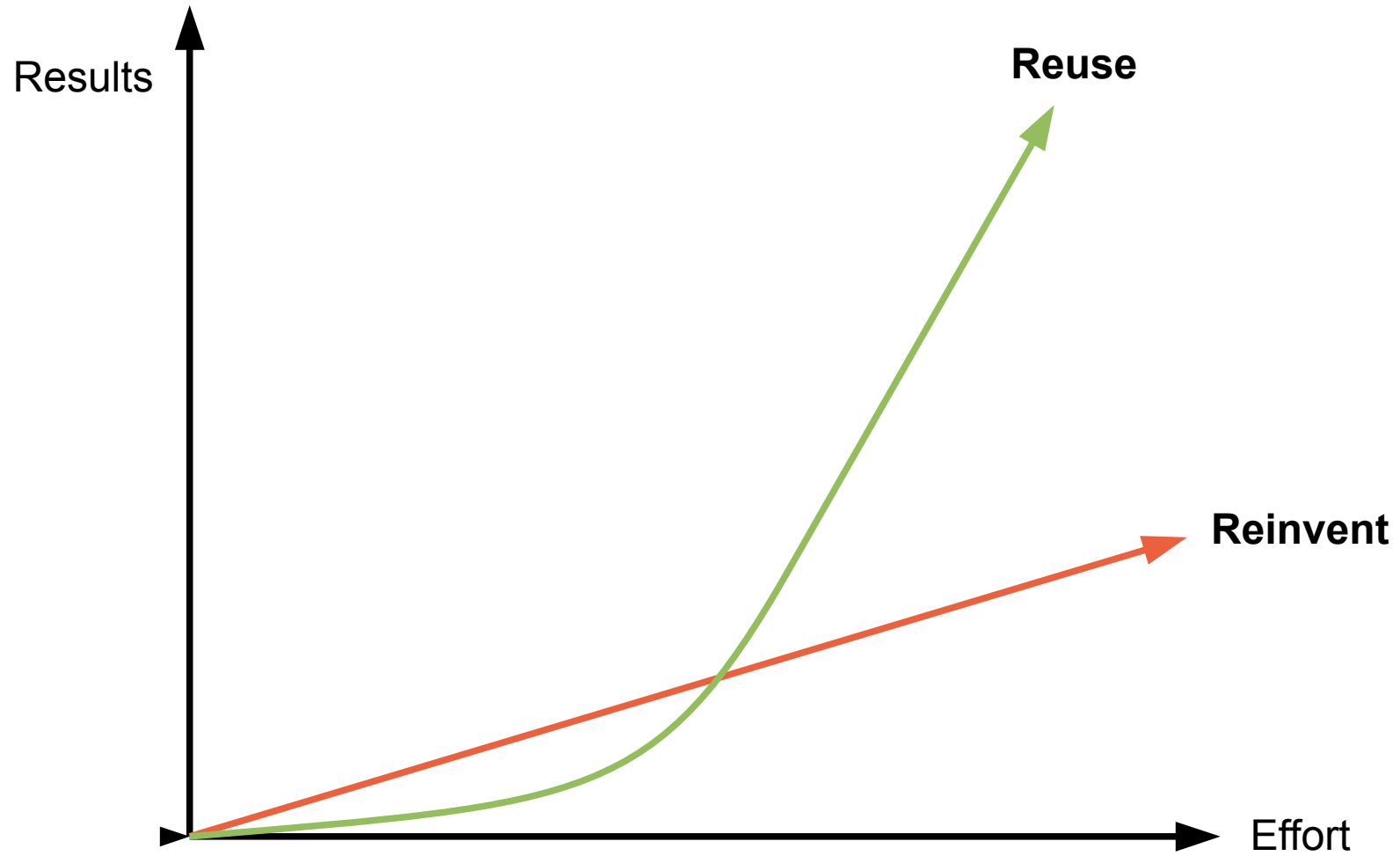
PHASE 3: CONVERTED

I'VE COMPILED A LIST OF FRAMEWORKS WE WILL USE ON THIS PROJECT. THIS PROJECT WILL PRACTICALLY WRITE ITSELF!

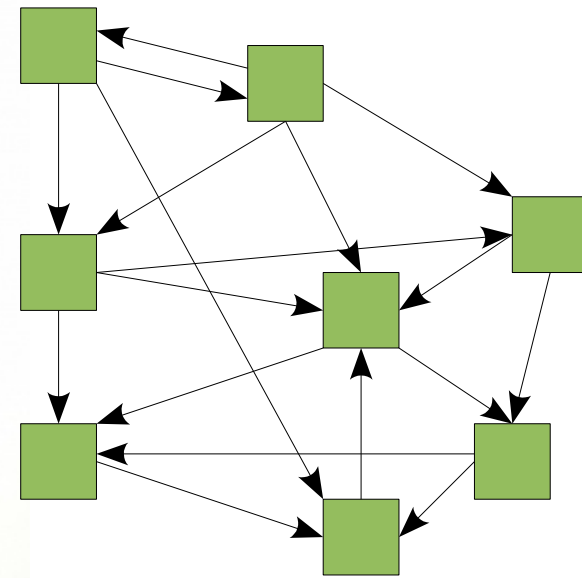


toonlet.com/creator/rjbrooksjr

Learning Curve

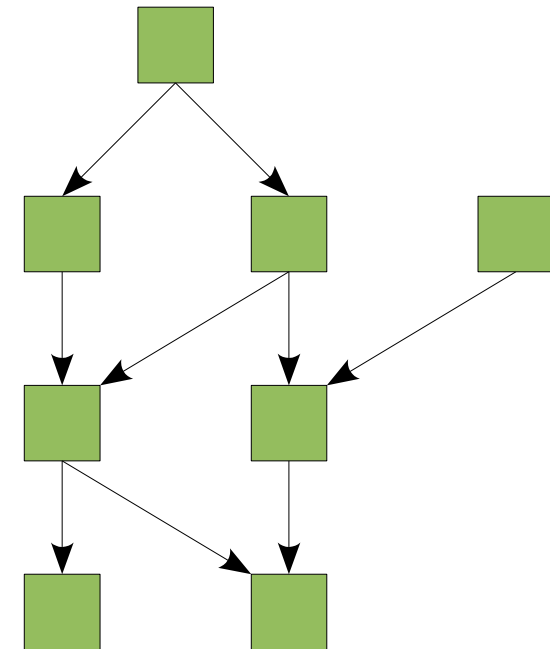


Object Oriented Spaghetti Code



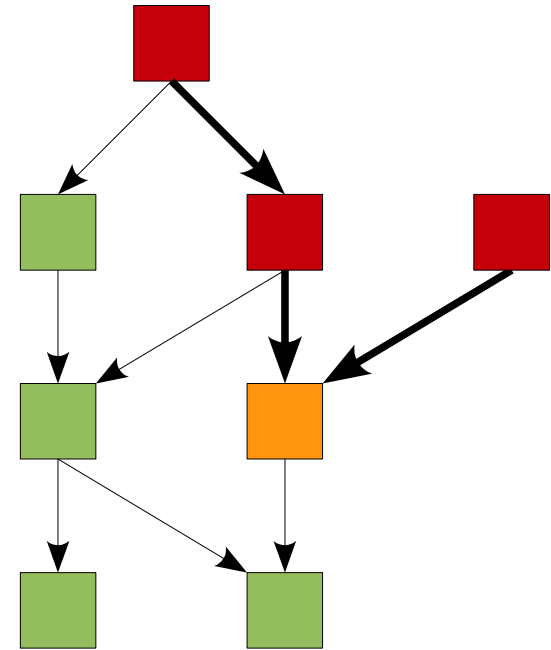
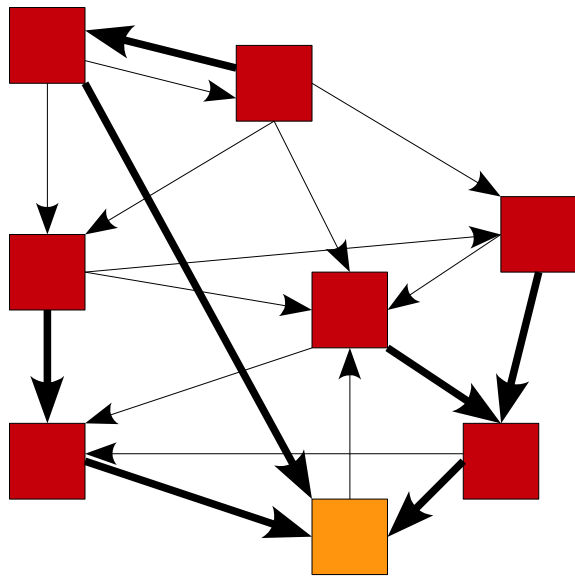
$\approx O(n^2)$ dependencies

Module System

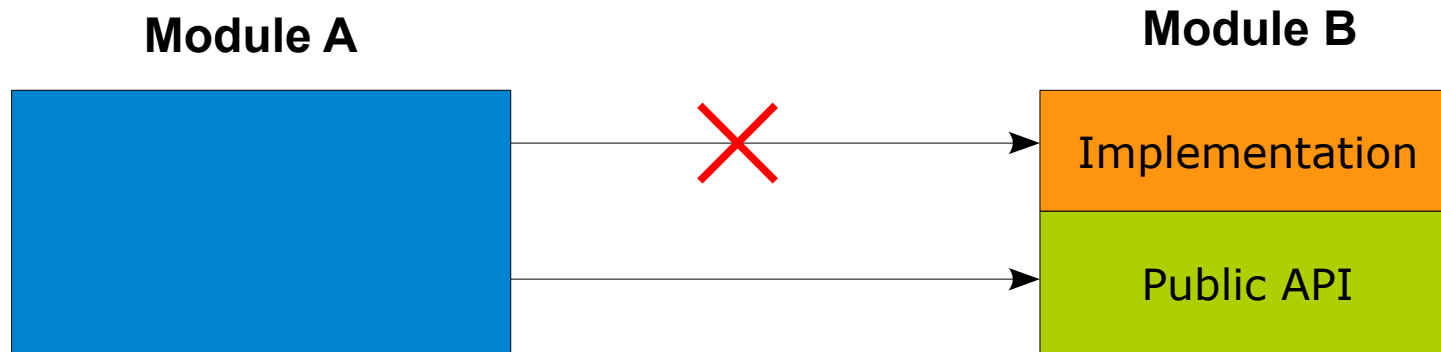


$\approx O(n)$ dependencies

Tracking impacts



Information Hiding

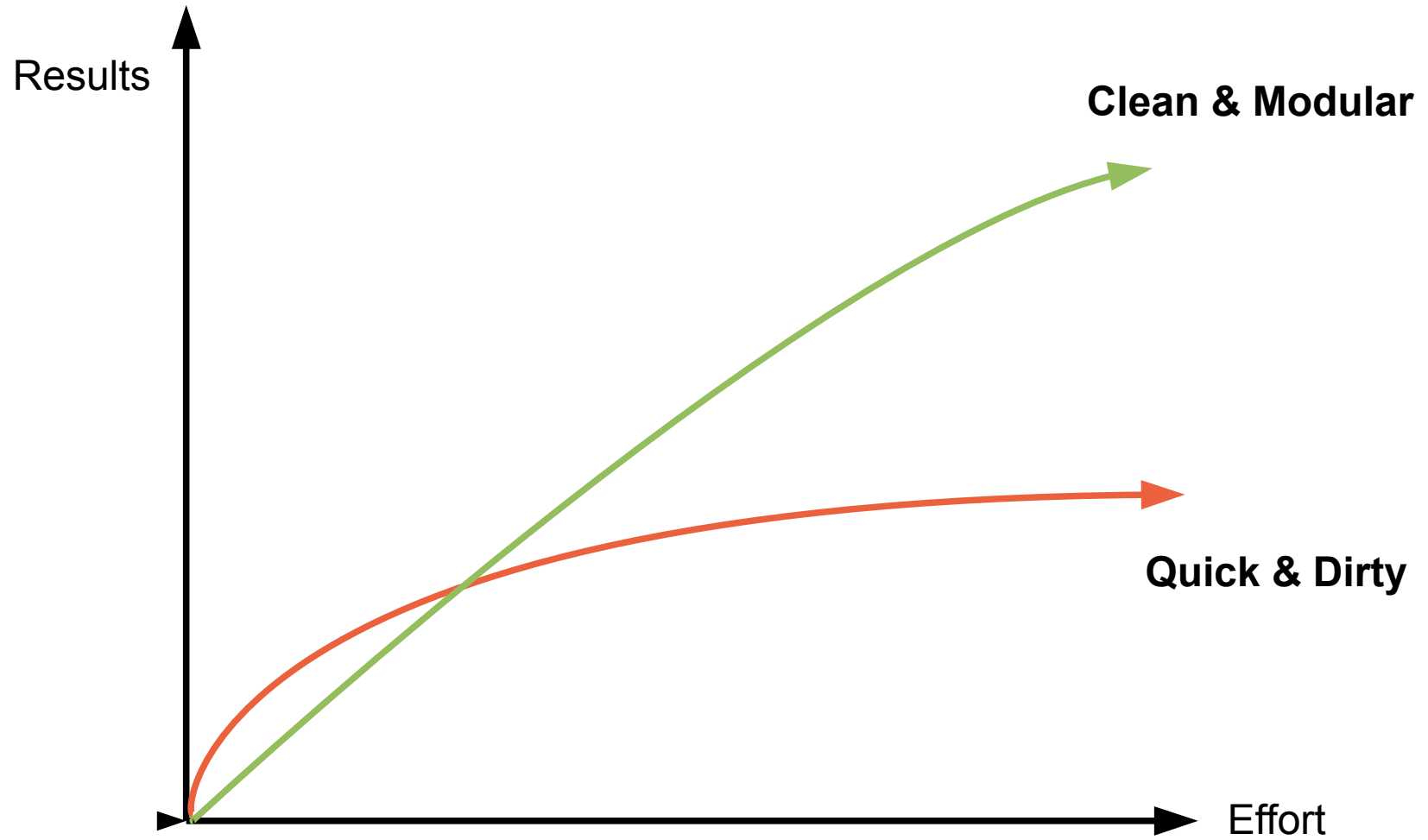


Limits are your friend!

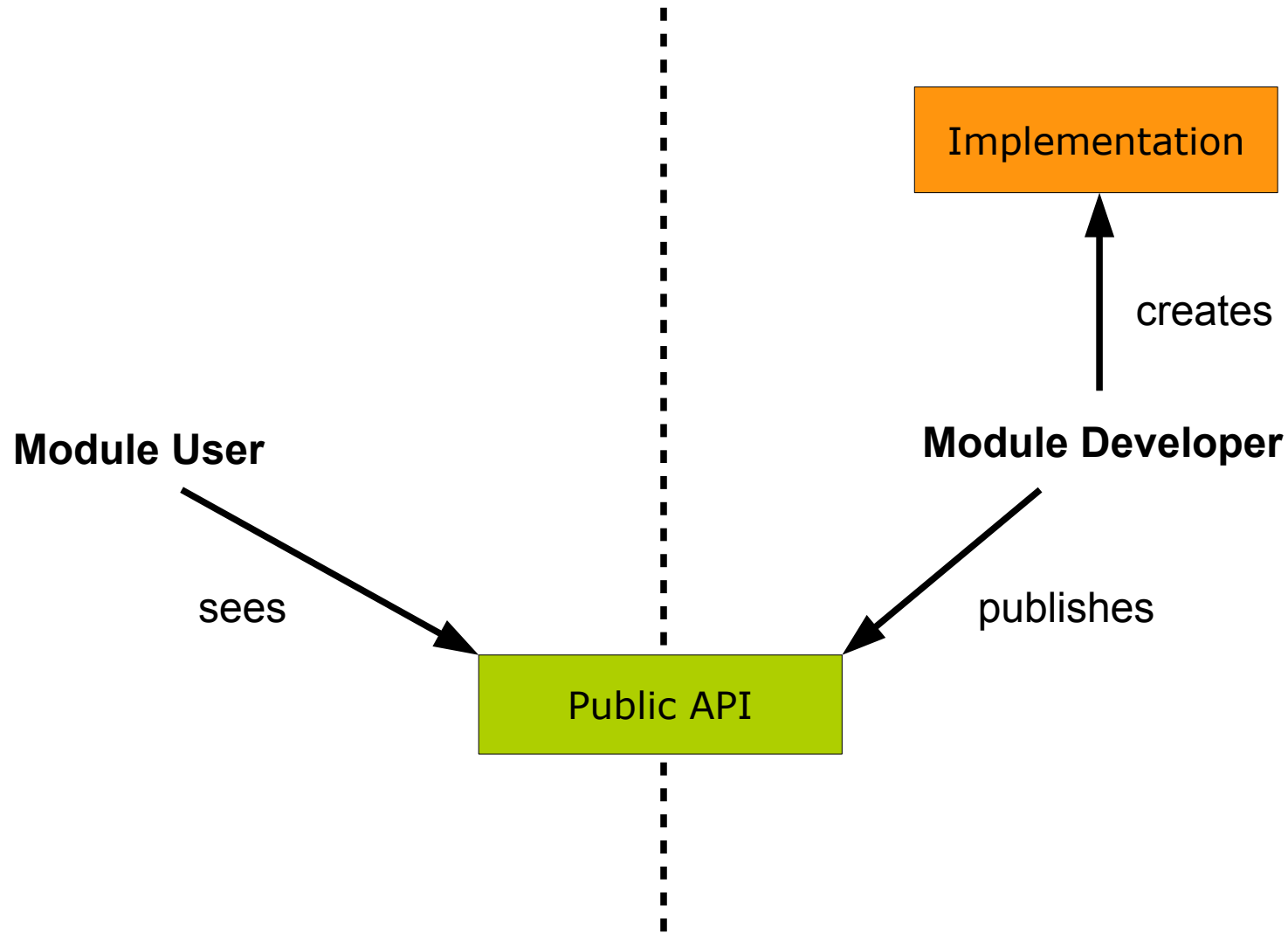
Restrict dependencies

Small public API

Clean Structure



Distributed Development



Mechanisms in Java

Access modifiers only at class level -

Custom class loaders!

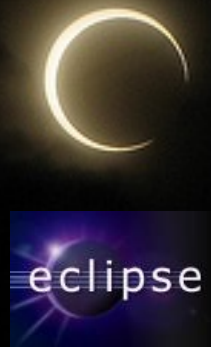
JDK9: New **modularized** JDK

GOTO vs IF, WHILE, ...

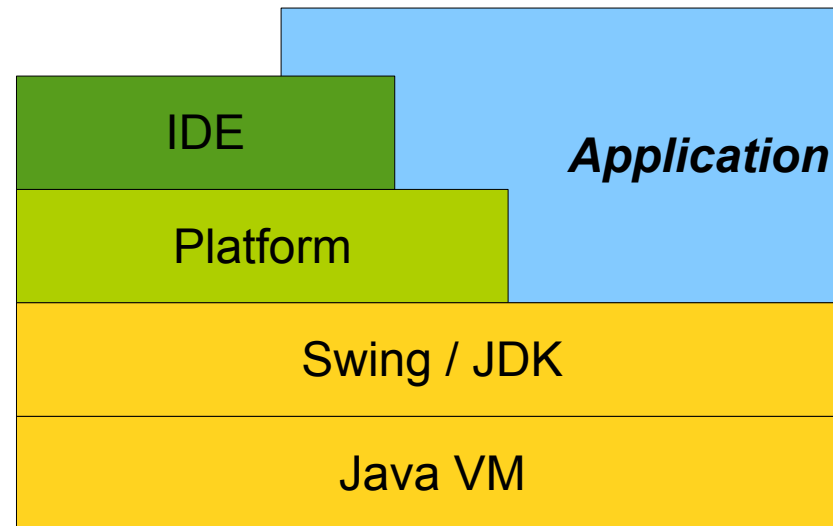
=

unrestricted OOP vs modular programming

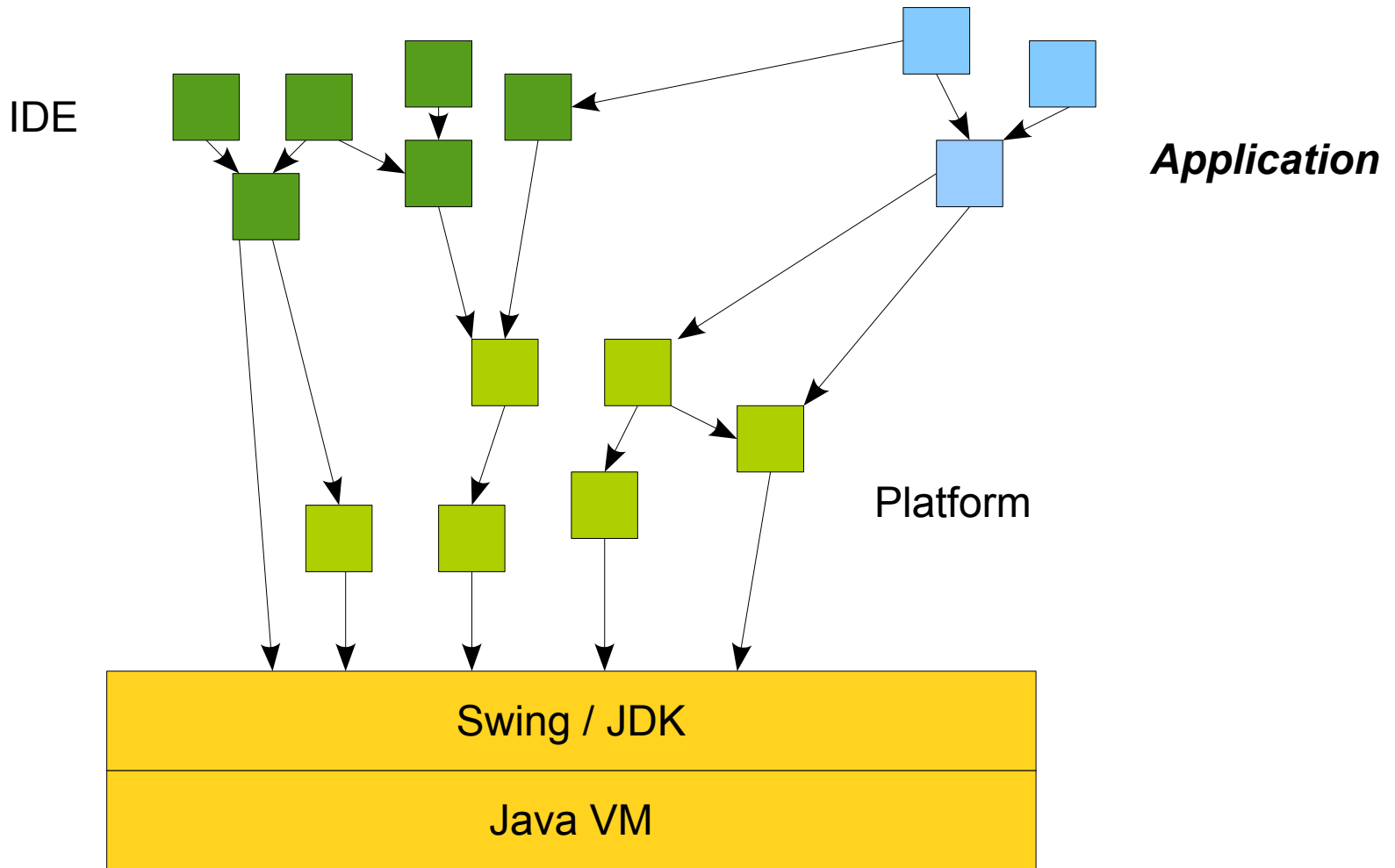
?



Architecture Eclipse/NetBeans



Detailed Architecture



- **Standalone:** Deployment including required platform / IDE modules
- **Plugin:** Deployment only with user-defined modules