

FROM NOTHING TO COMPLETE ENVIRONMENT WITH MAVEN, OOMPH & DOCKER

Max Bureck, 21. June 2017

FROM NOTHING TO COMPLETE ENVIRONMENT

1. Disclaimer
2. Motivation
3. Demo
4. Recap, Conclusion, and Future Possibilities

FROM NOTHING TO COMPLETE ENVIRONMENT



FROM (ALMOST) NOTHING TO COMPLETE ENVIRONMENT

Disclaimer: Prerequisites are Maven, Oomph & Docker 🙄

- Maven \geq 3.0
- Latest Eclipse installer
- Docker

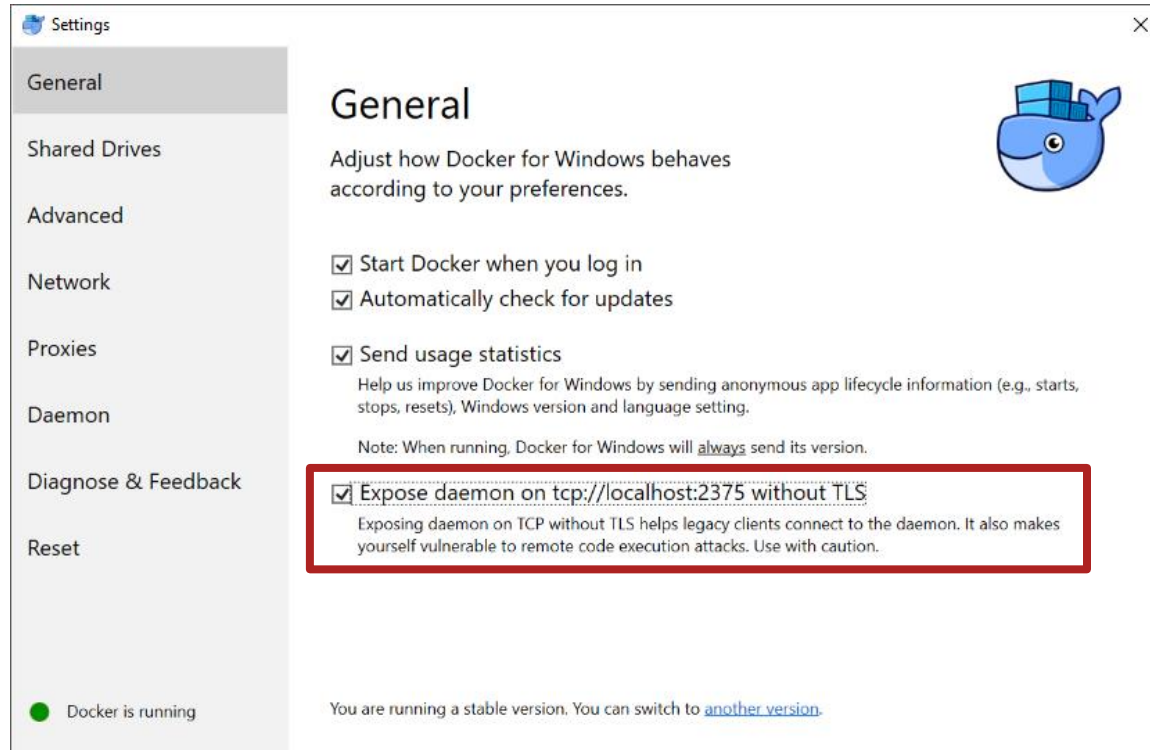
Maven™

eclipseinstaller by Oomph



FROM NOTHING TO COMPLETE ENVIRONMENT

Windows 10: Easiest Way to Make Docker Daemon Accessible From Eclipse



The screenshot shows the Windows Settings application with the 'Docker for Windows' settings page open. The 'General' tab is selected, and the 'Expose daemon on tcp://localhost:2375 without TLS' option is checked and highlighted with a red box. The Docker logo is visible in the top right corner of the settings window.

Settings

General

Shared Drives

Advanced

Network

Proxies

Daemon


Diagnose & Feedback

Reset

Docker is running

General

Adjust how Docker for Windows behaves according to your preferences.



- Start Docker when you log in
- Automatically check for updates
- Send usage statistics

Help us improve Docker for Windows by sending anonymous app lifecycle information (e.g., starts, stops, resets), Windows version and language setting.

Note: When running, Docker for Windows will always send its version.

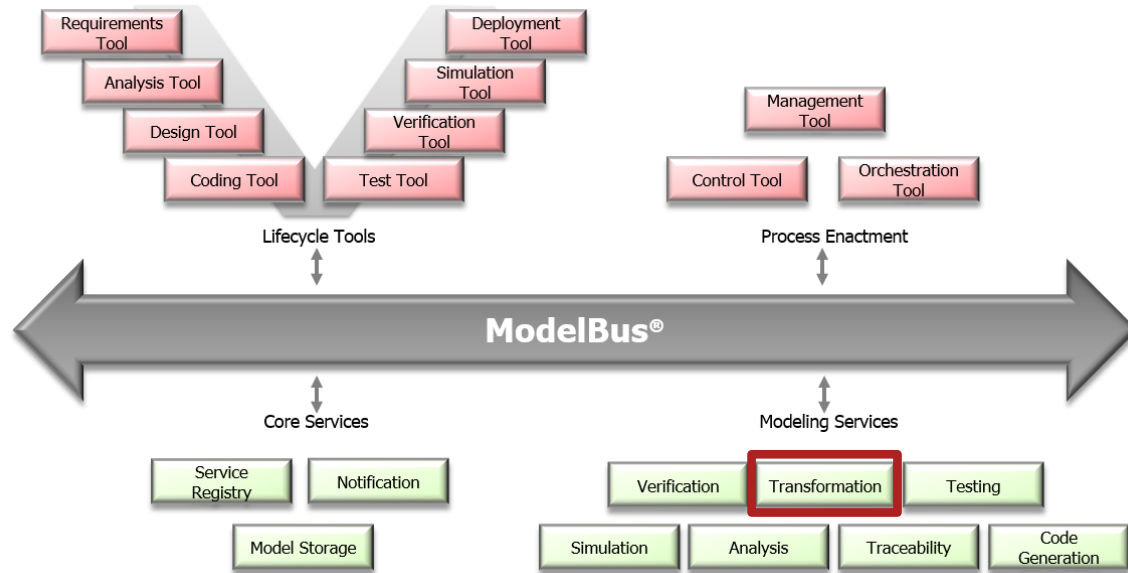
- Expose daemon on tcp://localhost:2375 without TLS**

Exposing daemon on TCP without TLS helps legacy clients connect to the daemon. It also makes yourself vulnerable to remote code execution attacks. Use with caution.

You are running a stable version. You can switch to [another version](#).

FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Target Domain Introduction



Writing a transformation service

FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Target Domain Introduction

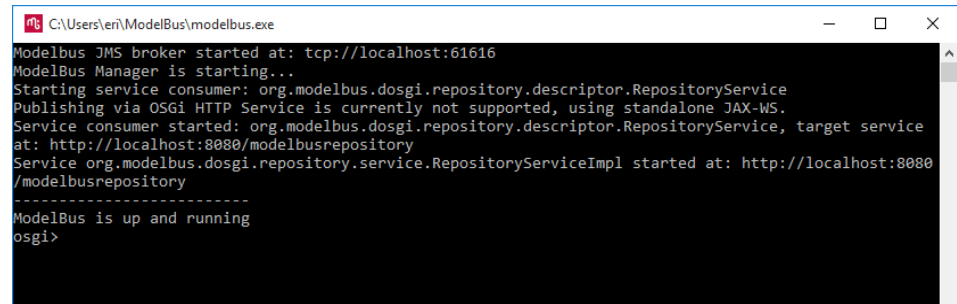
- Developing ModelBus services should be simple
- Usually first task for new team member:
 - Read ModelBus user guide
 - Create example ModelBus services



FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience, Step 1

- Setup Action:
 - Install ModelBus server
 - Configure ModelBus server
- Problems:
 - User forgets to set mandatory environment variable
 - User forgets to add SVN plugins (for legal reasons not part of ModelBus distro)

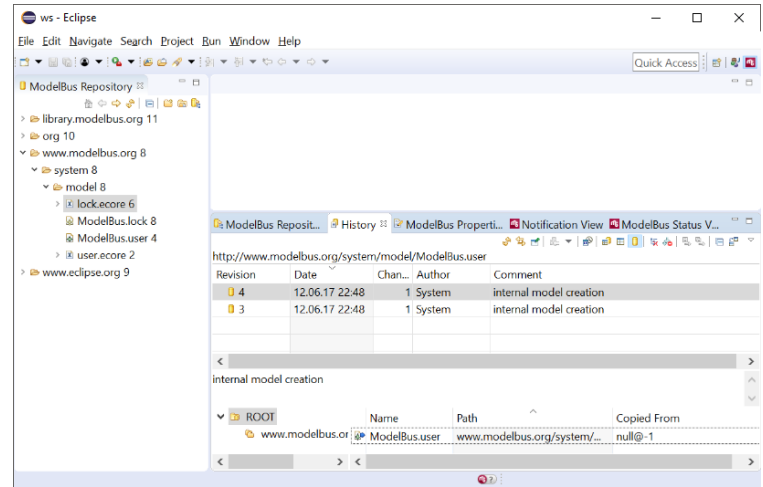


```
C:\Users\eni\ModelBus\modelbus.exe
ModelBus JMS broker started at: tcp://localhost:61616
ModelBus Manager is starting...
Starting service consumer: org.modelbus.dosgi.repository.descriptor.RepositoryService
Publishing via OSGi HTTP Service is currently not supported, using standalone JAX-WS.
Service consumer started: org.modelbus.dosgi.repository.descriptor.RepositoryService, target service
at: http://localhost:8080/modelbusrepository
Service org.modelbus.dosgi.repository.service.RepositoryServiceImpl started at: http://localhost:8080
/modelbusrepository
-----
ModelBus is up and running
osgi>
```


FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience, Step 2

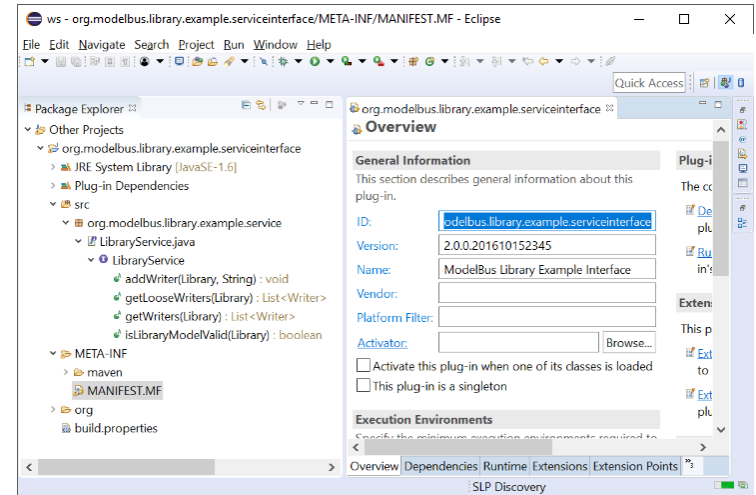
- Setup Actions:
 - Install Eclipse
 - Install ModelBus TeamProvider plugins
 - Configure TeamProvider and target platform
- Problem:
 - User does not add plugins for standalone services to target platform



FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience, Step 3

- Setup Action:
 - Create projects for service interface, provider - and consumer service
- Problem:
 - User forgets to add mandatory plugin dependency

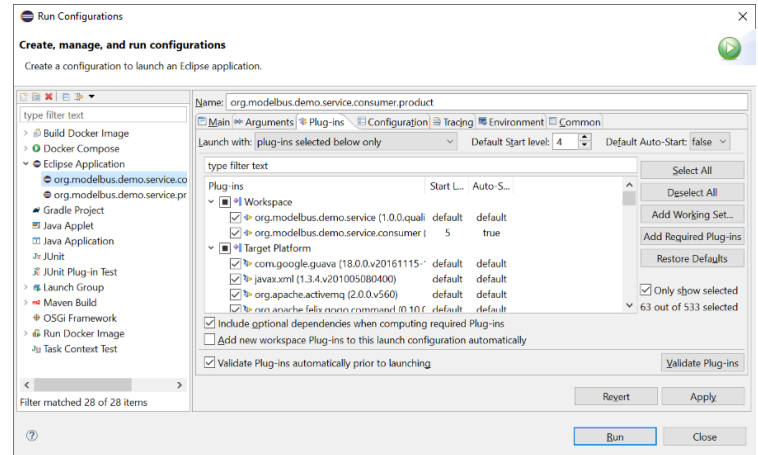


FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience, Step 4

- Setup Action:
 - User creates launch configurations for provider and consumer projects

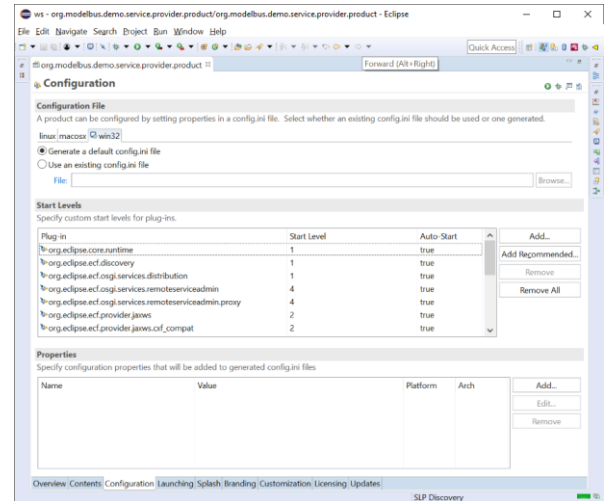
- Problem:
 - User forgets to set one or more start-levels for plugins



FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience, Step 5

- Setup Action:
 - User creates product definition for services (for standalone executable)
- Problems:
 - User forgets to set one or more start-levels for plugins (again)
 - Forgotten plugins in product definition



FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience, Step 6

- Bonus Action:
 - Create Maven Tycho build scripts
- Problems:
 - **Bundle IDs / Feature IDs have to match POM artifactIds**



```
<code><?xml version='1.0' encoding='UTF-8' ?>
<project xmlns='http://maven.apache.org/POM/4.0.0' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance' xsi:schemaLocation='http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd'>
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.eclipse.osgi</groupId>
  <artifactId>org.eclipse.osgi</artifactId>
  <version>3.10.0</version>
  <packaging>bundle</packaging>
  <name>org.eclipse.osgi</name>
  <description>Eclipse OSGi Framework</description>
  <url>http://www.eclipse.org/osgi/</url>
  <properties>
    <property name='osgi.bundle.symbolicname' value='org.eclipse.osgi' />
    <property name='osgi.bundle.namespace' value='org.eclipse.osgi' />
    <property name='osgi.bundle.namespace.version' value='3.10.0' />
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.osgi</groupId>
      <artifactId>org.osgi.core</artifactId>
      <version>6.0.0</version>
    </dependency>
    <dependency>
      <groupId>org.osgi</groupId>
      <artifactId>org.osgi.enterprise</artifactId>
      <version>6.0.0</version>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.1</version>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-jar-plugin</artifactId>
        <version>3.1</version>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-bundle-plugin</artifactId>
        <version>5.1.2</version>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-tycho-plugin</artifactId>
        <version>0.29.0</version>
      </plugin>
    </plugins>
  </build>
  <profiles>
    <profile>
      <id>tycho</id>
      <activation>
        <activeByDefault>true</activeByDefault>
      </activation>
      <build>
        <plugins>
          <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-tycho-plugin</artifactId>
            <version>0.29.0</version>
          </plugin>
        </plugins>
      </build>
    </profile>
  </profiles>
</project></code>
```

FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Setup Experience Conclusion



<https://imgflip.com/i/1qsphq>

FROM NOTHING TO COMPLETE ENVIRONMENT

Motivation: Improving Setup Experience

- Use automation to get rid of error prone manual tasks of
 - Installation of ModelBus server
 - Installation and configuration of Eclipse + plugins
 - Creation of service interface, provider, and consumer plugins
 - Creation of launch configs and product definition
 - Creation of Maven Tycho build scripts

FROM NOTHING TO COMPLETE ENVIRONMENT

Demo

FROM NOTHING TO COMPLETE ENVIRONMENT

Recap, Part 1

- We used Maven archetype to generate
 - Projects (service interface, provider, consumer, features, products)
 - Maven build scripts
 - Oomph setup file
 - Dockerfiles for ModelBus server, provider service, consumer service
 - Launch configs for build, run, debug

FROM NOTHING TO COMPLETE ENVIRONMENT

Recap, Part 2

- We used generated Oomph setup to
 - Install Eclipse
 - Install plugins into Eclipse
 - Configure Eclipse + plugins
 - Configure target platform

FROM NOTHING TO COMPLETE ENVIRONMENT

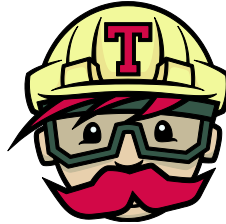
Conclusion

- If your platform is based on OSGi / Eclipse tech: Consider Maven archetypes
- Use Oomph for (reproducible) Eclipse setups
- Consider Docker if your Eclipse based platform integrates with network services

FROM NOTHING TO COMPLETE ENVIRONMENT

Going Even Further (When Possible)

- Install prerequisites via package managers (apt, chocolatey, homebrew, etc.)
 - Single shell script call for setup, literally starting from nothing
- Starting eclipse installer with Oomph setup file from command line
- Generating CI configuration (Jenkinsfile, .travis.yml, .gitlab-ci.yml, etc.)



FROM NOTHING TO COMPLETE ENVIRONMENT

Please Vote and Leave Comments



Evaluate the Sessions

Sign in and vote at eclipsecon.org

-1

0

+1

Thanks!

Questions?

CONTACT

Fraunhofer FOKUS
Kaiserin-Augusta-Allee 31
10589 Berlin, Germany
www.fokus.fraunhofer.de

Max Bureck
Senior Researcher
max.bureck@fokus.fraunhofer.de
Tel. +49 (0)30 3463-7321