SpringSource
dm Server v2.0

Agenda

- dm Server v1.0.x recap

- dm Server v2.0

  - provisioning: plans, repositories

  - web application support

  - regions

  - medic

 - other enhancements

- Supporting tools: bundlor, STS

dm Server v1.0

- Open source, GPL licensed. Commercial license available

- OSGi-based kernel + subsystems + application programming model[1]

  - built on Equinox

- Provisioning from local repository

- Enhanced serviceability & problem diagnostics

- Existing enterprise library compatibility

- Supports deployment of:

  - any OSGi bundle

  - war files (embedded tomcat)

  - web-module bundles

  - par files

(1) Spring Dynamic Modules programming model is Apache Licensed

dm Server 1.0 – par files

- set of bundles
  - unit of packaging
  - unit of deployment
  - runtime scope
    - type visibility
    - service visibility
    - synthetic context

SpringSource Confidential. Do not distribute without express permission

Slide 4

dm Server 1.0 – web support

SpringSource Confidential. Do not distribute without express permission

Slide 5
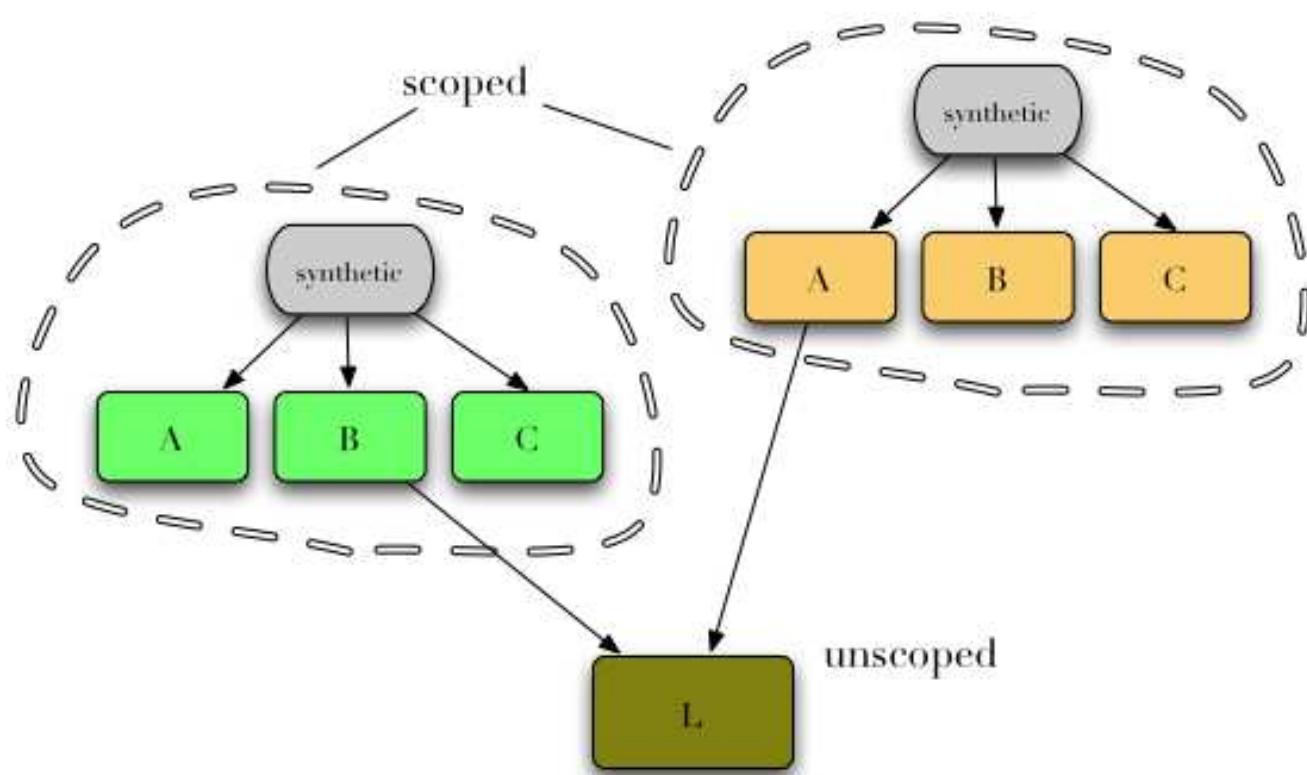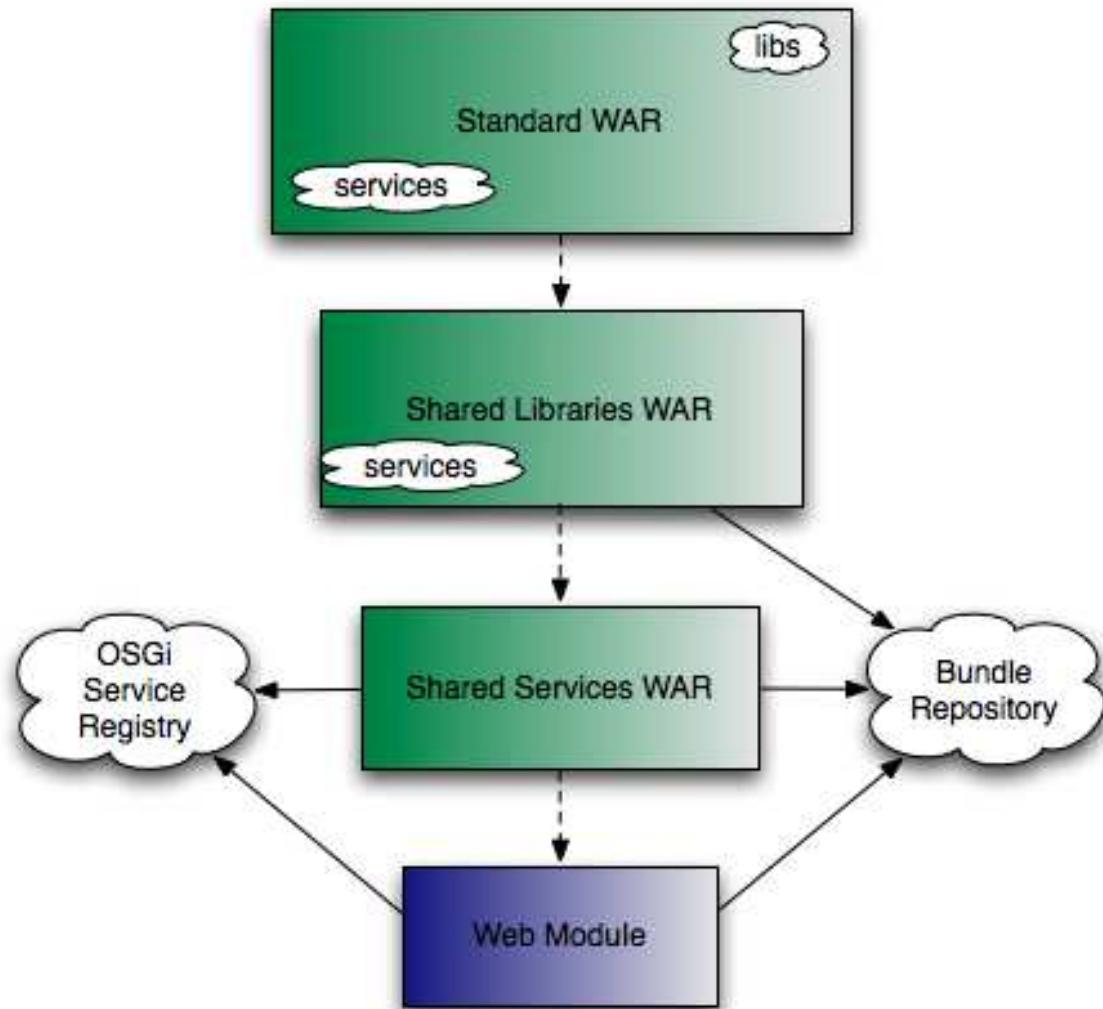
dm Server 2.0

Plan files

- generalize par file concept
 - (optionally) atomic lifecycle (deploy, start, stop, ...)
  - (optionally) scoped
  - more than just bundles

```xml
<plan name="greenpages.bundles" version="1.0.0" abstract="true">
  <artifact type="bundle" name="greenpages.db" version="1.0.1"/>
  <artifact type="bundle" name="greenpages.app" version="1.0.0"/>
  <artifact type="bundle" name="greenpages.jpa" version="1.0.3"/>
  <artifact type="bundle" name="greenpages.web" version="1.0.0"/>
</plan>

<plan name="greenpages.dev" version="1.0.0" scoped="true" atomic="true">
  <artifact type="properties" name="greenpages.jdbc.dev" version="1.0.0"/>
  <artifact type="plan" name="greenpages.bundles" version="1.0.0"/>
</plan>

<plan name="greenpages.prod" version="1.0.0" scoped="true" atomic="true">
  <artifact type="properties" name="greenpages.jdbc.prod" version="1.0.0"/>
  <artifact type="plan" name="greenpages.bundles" version="1.0.0"/>
</plan>
```

spring source

Plans

- Plans are just artefacts in a repository

- as is everything they refer to

- properties files provisioned from a repository are deployed as configuration admin dictionaries

- configure dm Server with an *initial plan*

# Provisioning: bundle dependencies

- Bundle dependencies:
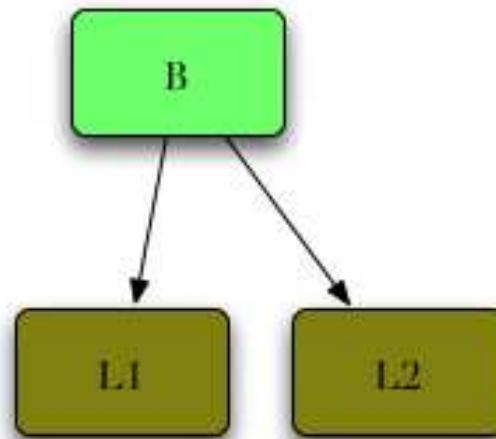
- Attempted installation

-> fail to resolve!

SpringSource Confidential. Do not distribute without express permission

Slide 9

Provisioning

- dm Server detects missing dependencies and automatically provisions bundles from repository to satisfy
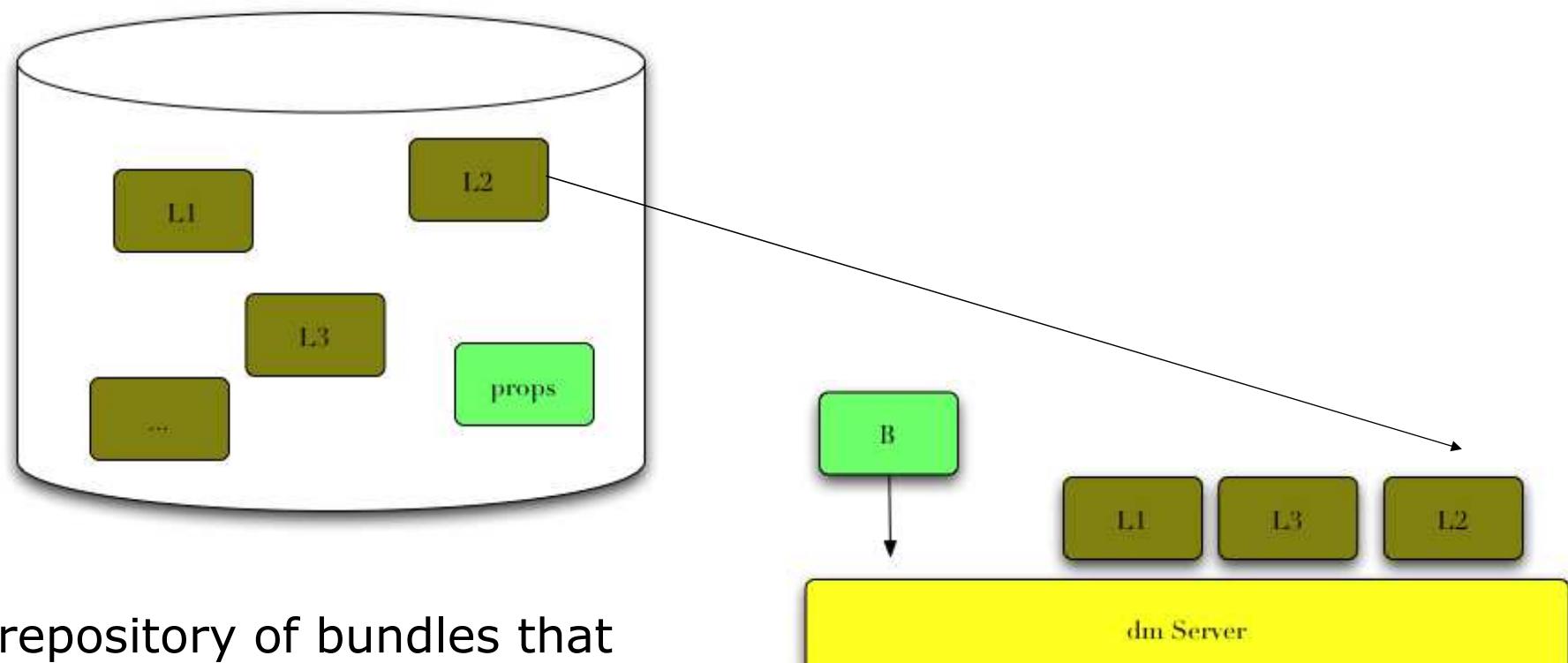
repository of bundles that *might* be installed
- disk footprint only

minimal runtime footprint maintained

dm Server 2.0

- Provisioning from chained repositories

  - may be local

  - or remote

```
{
    ...

    "remote-repo" : {
        "type" : "remote",
        "uri" : "http://my-
host:8080/com.springsource.server.repository/my-hosted-repo",
        "indexRefreshInterval" : 30,
    },
    "repositoryChain" : [
        "bundles-subsystems",
        "bundles-ext",
        "bundles-usr",
        "libraries-ext",
        "libraries-usr",
        "remote-repo"
    ]
}
```

Corporate approved libraries
Shared Repo

Project
Shared Repo

dm Server

Local
Repo
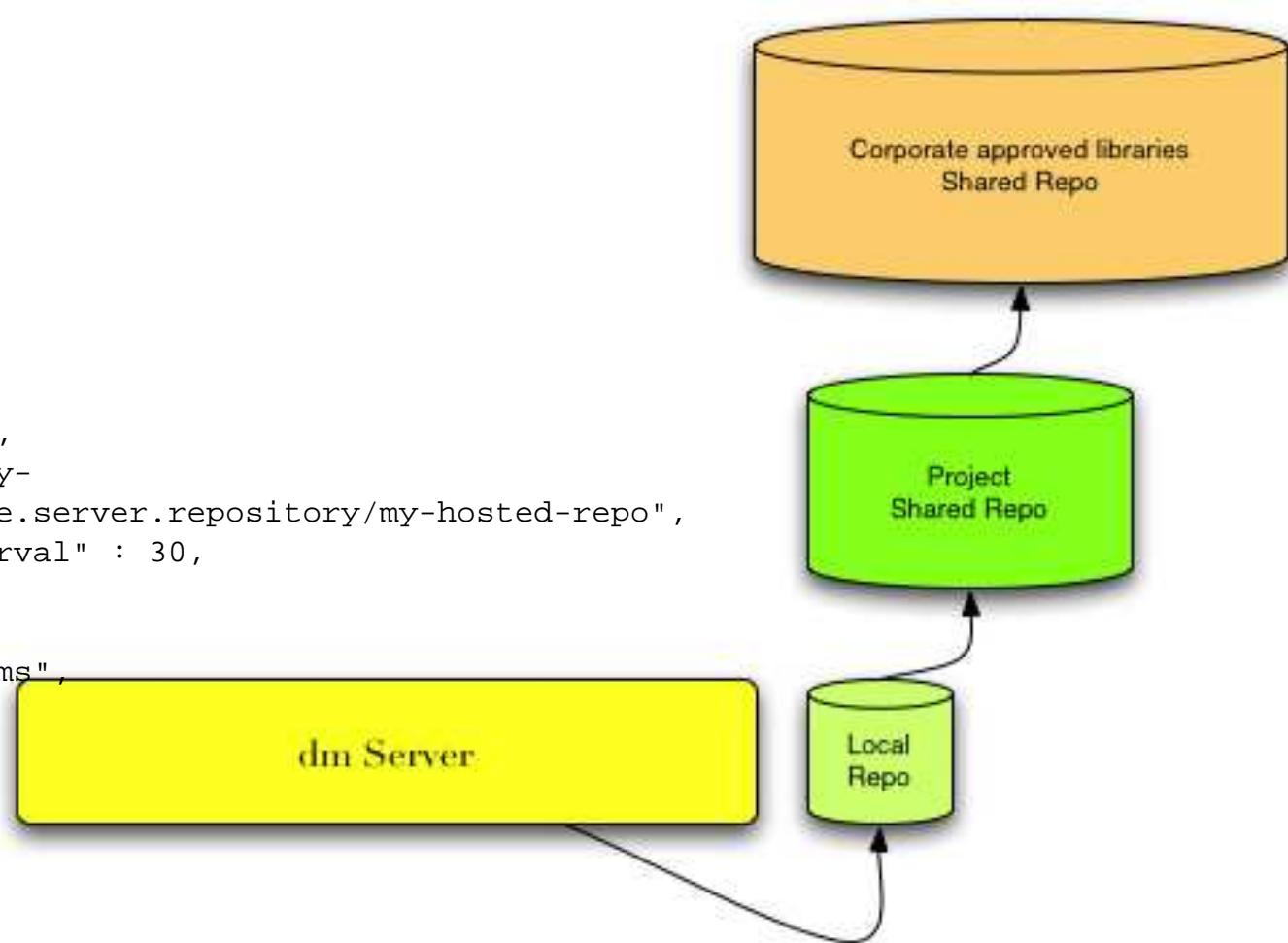
Web application support

- web modules deprecated

- replaced by RFC-66 Reference Implementation

- + standard tomcat configuration files

- standard dm Server deployment + URL protocol support

  - `install webbundle:`file:formtags.war`[?Web-ContextPath=...]`

    - adds necessary manifest entries

      - symbolic name, version, imports etc.

    - sets up `Bundle-Classpath` to include `WEB-INF/classes` and all jars in `WEB-INF/lib`

- all dm Server war options supported (shared services, shared library)

On the criteria for decomposing systems into modules...

horizontal decomposition

SpringSource Confidential. Do not distribute without express permission

Slide 13
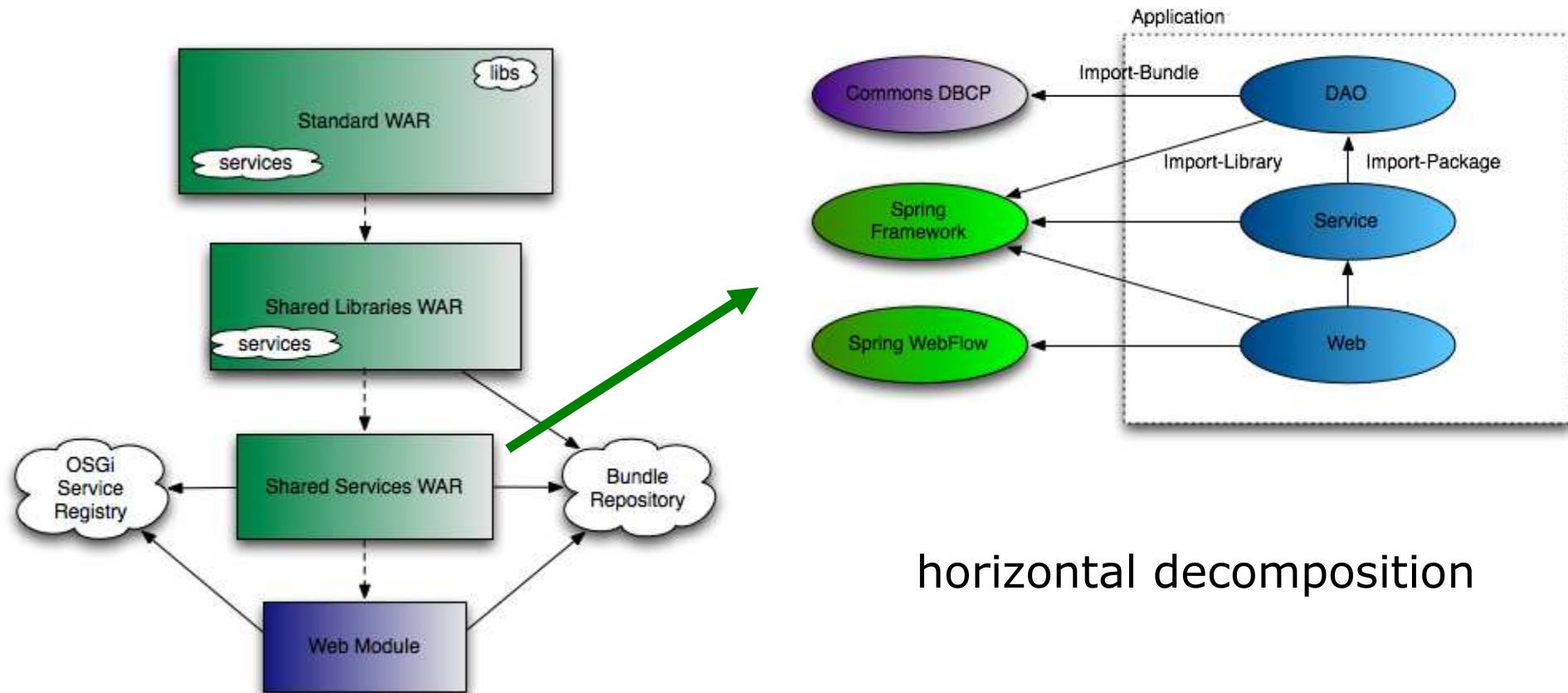
On the criteria for decomposing systems into modules...

- We want **vertical decomposition** by business function as primary decomposition mechanism

web bundle

← can't partition web layer!

infra-structure bundle

function A

function B

function C

spring source

Web slices

SpringSource Confidential. Do not distribute without express permission

Slide 15
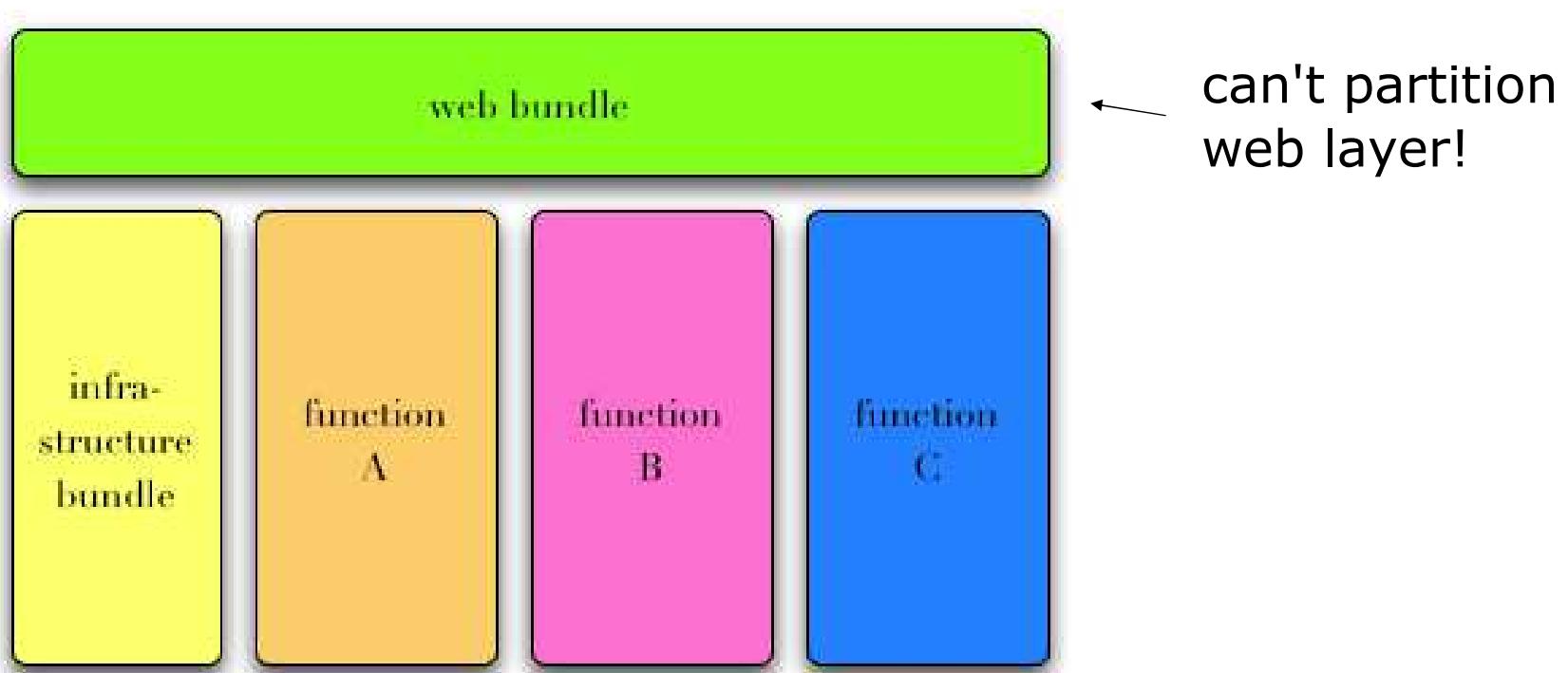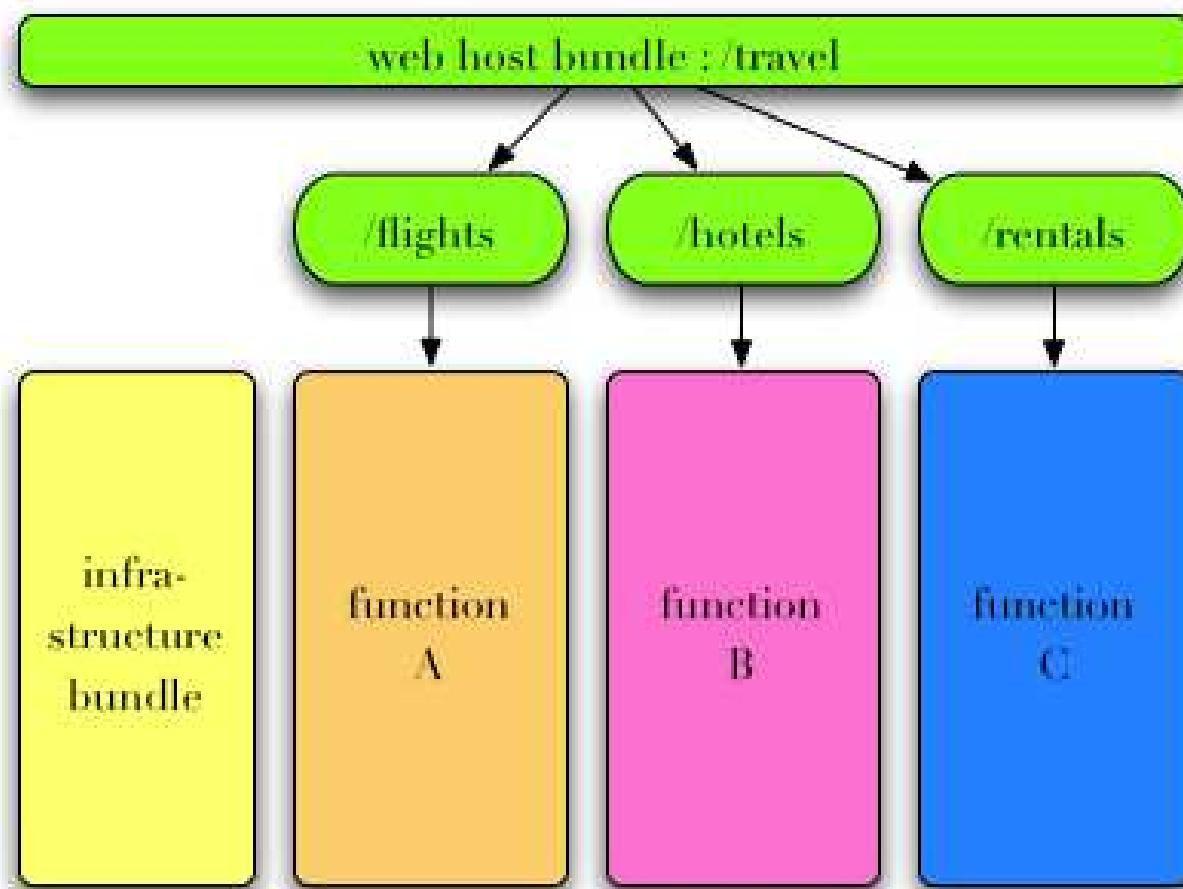
Version isolation

This works in OSGi
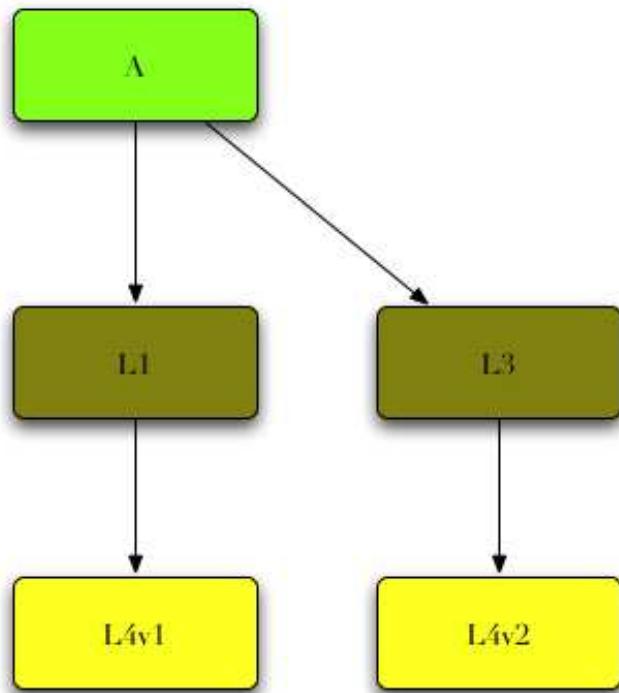
This doesn't
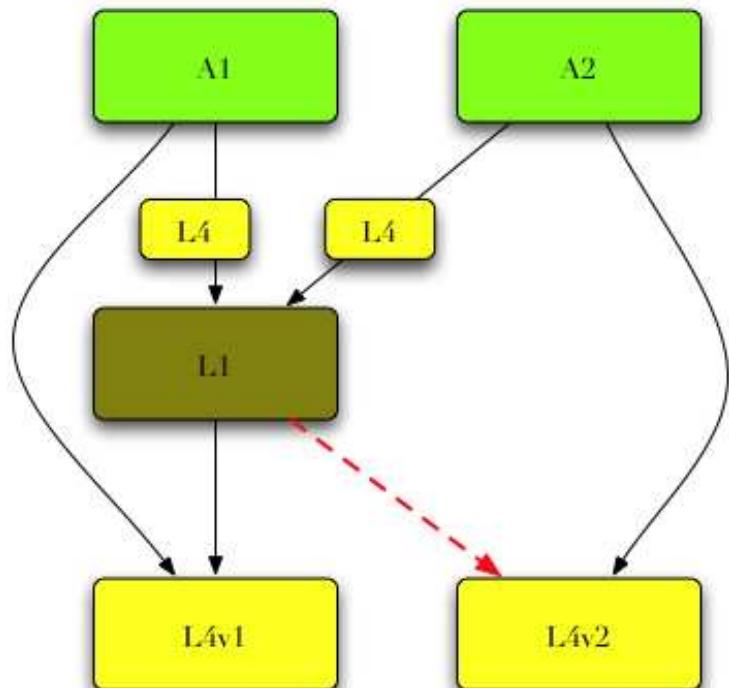
Regions

- Regions isolate user code from dm server kernel code

- Kernel and users can use different versions of Spring

SpringSource Confidential. Do not distribute without express permission

Slide 17
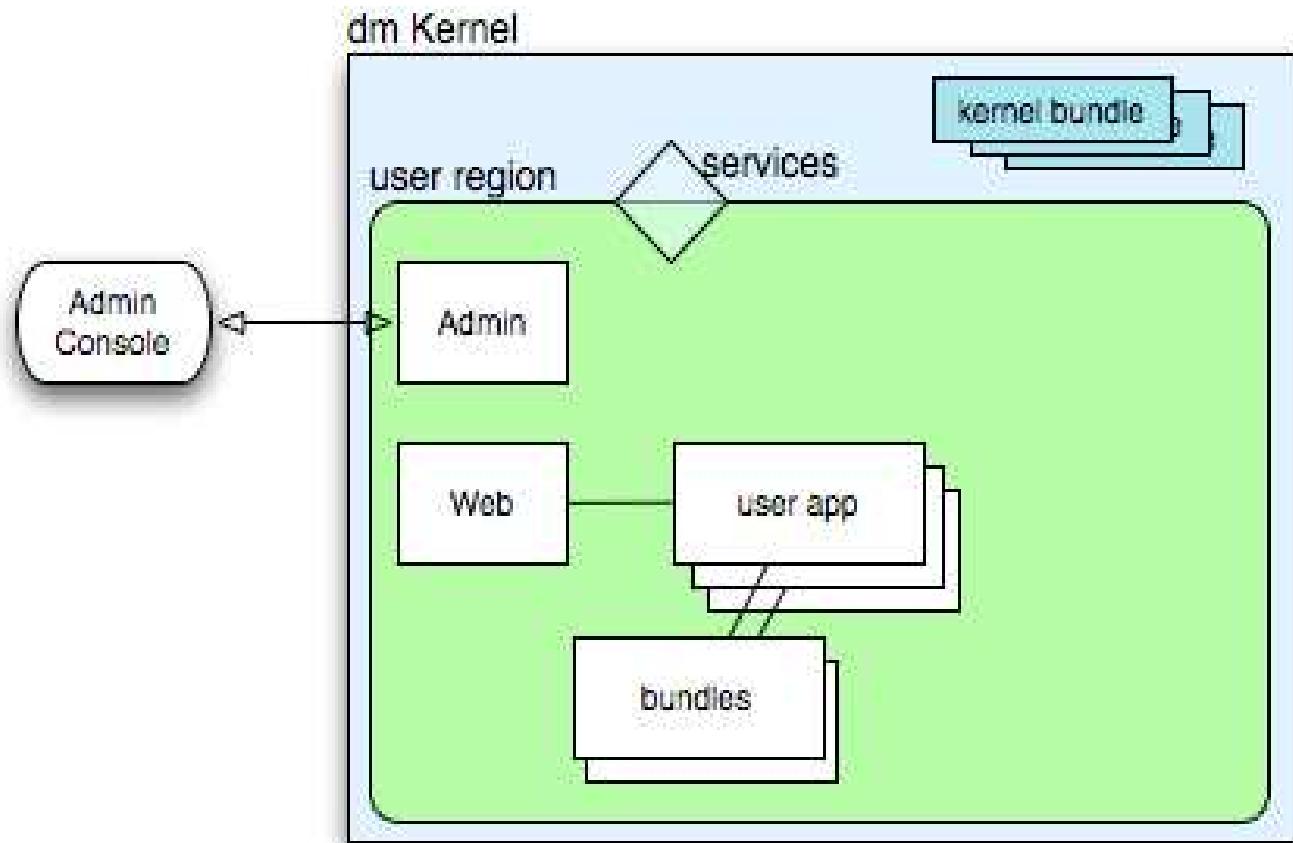
Version isolation

- Application A
  - Spring 2.5.6
  - Hibernate 3.3.2

- Application B
  - Spring 2.5.6
  - Hibernate 3.2.6

- *Spring bundle(s) can't simultaneously bind to both Hibernate 3.3.2 and Hibernate 3.2.6*
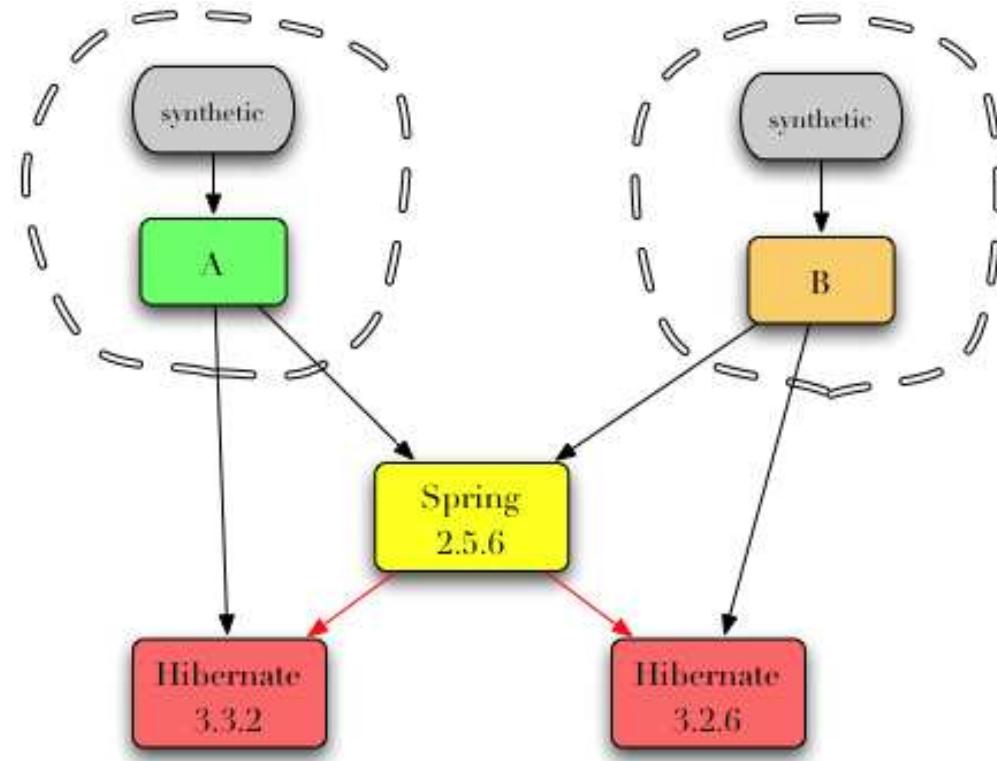
Cloning

- Cloning enables
  - maximum flexibility, minimal footprint
  - also solves "shared static" issues
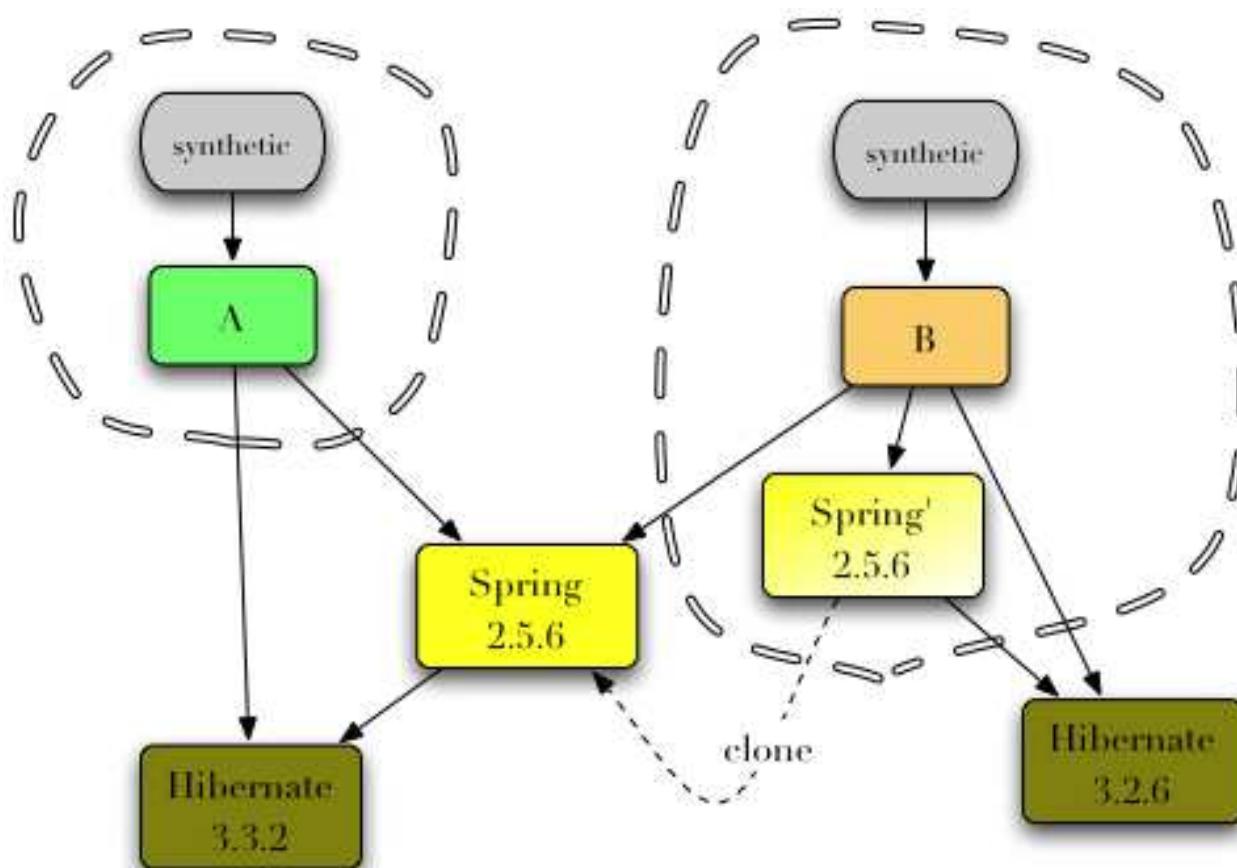
medic

- dm Server serviceability suite

  - tracing

  - logging

  - dumping

- available for application use

  - logback configuration

  - control format, destination for any bundle or group of bundles

    - independent configuration

Admin console improvements

Admin console improvements

Blueprint Service

- Support for OSGi Blueprint Service (4.2 specification)
  - aka "RFC 124"

- Standardization of Spring Dynamic Modules + core of Spring container as an OSGi Compendium Service

- Track the Spring Dynamic Modules 2.0 release

tools

springOne

Bundlor

- Consistent management of manifest across ant, maven, command-line, and your IDE

SpringSource Confidential. Do not distribute without express permission

Slide 25
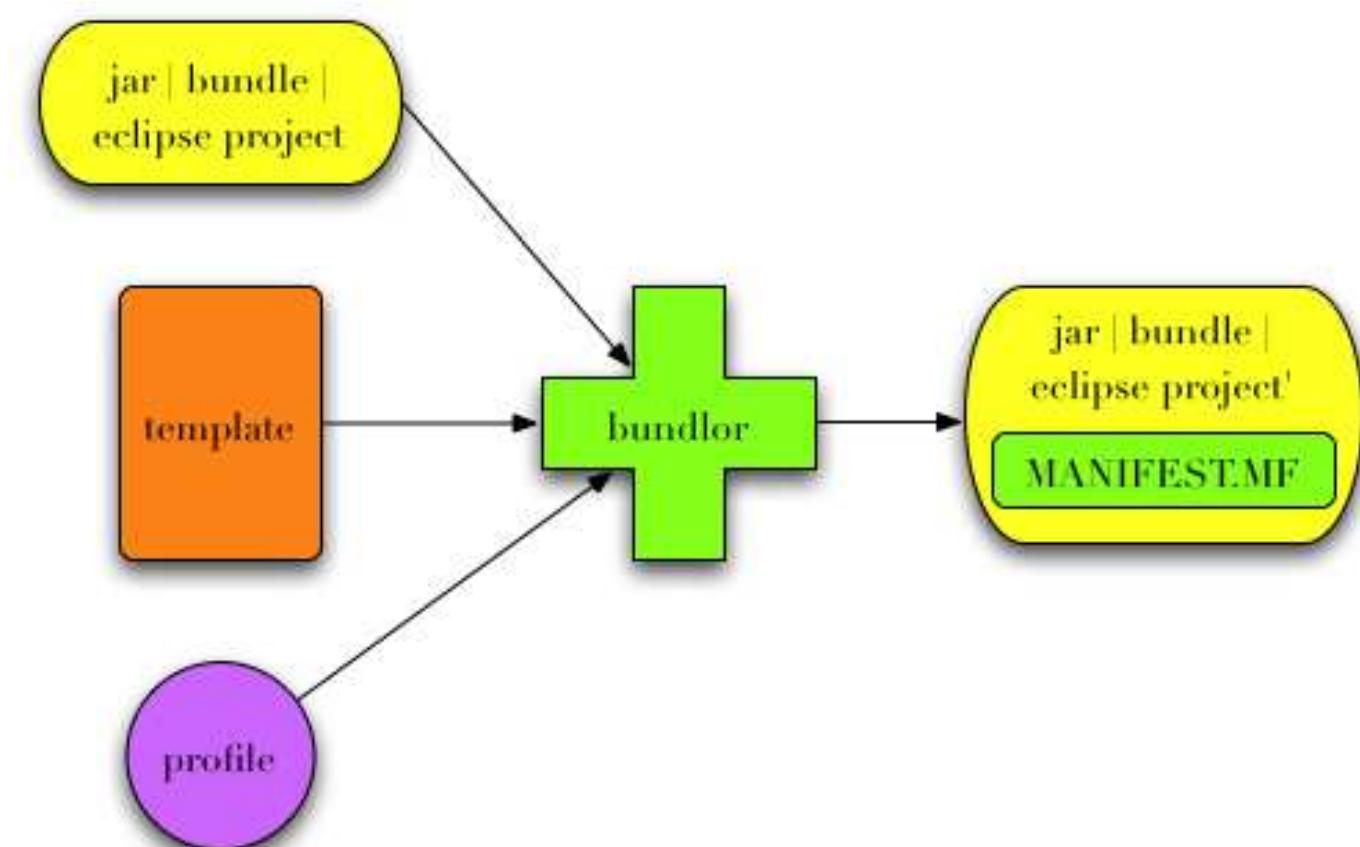
Bundlor benefits

- Automatic management of imports based on scanning project / jar

  - + Spring config, Hibernate, JPA, properties, *web.xml*, ...

- Import-Template supports specification of version information and optional imports without redundancy

  - `Import-Template:`

    `org.springframework.*;version="[2.5.4.A,3.0.0)"`

- Automatic management of *versioned* exports based on bundle version

  - can provide explicit version info with Export-Template

  - can restrict with Excluded-Exports

- Property substitution

- Version-range policies

SpringSource Confidential. Do not distribute without express permission

Slide 26

Bundlor benefits

- Automated manifest generation inside Eclipse

  - removes common developer pain point
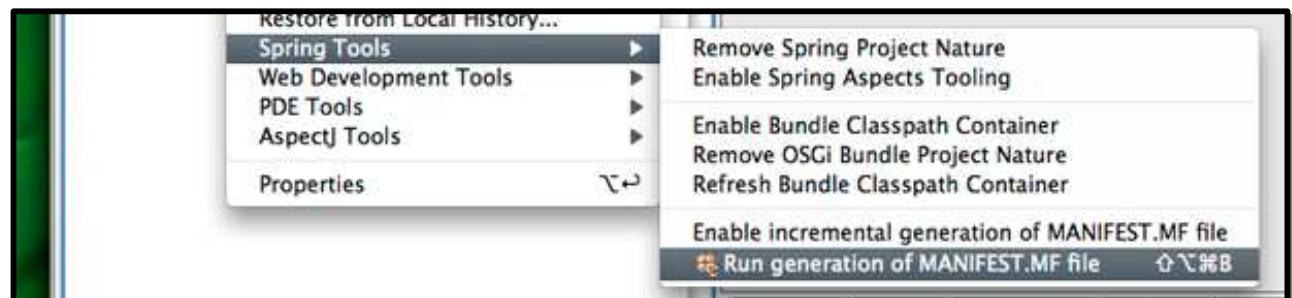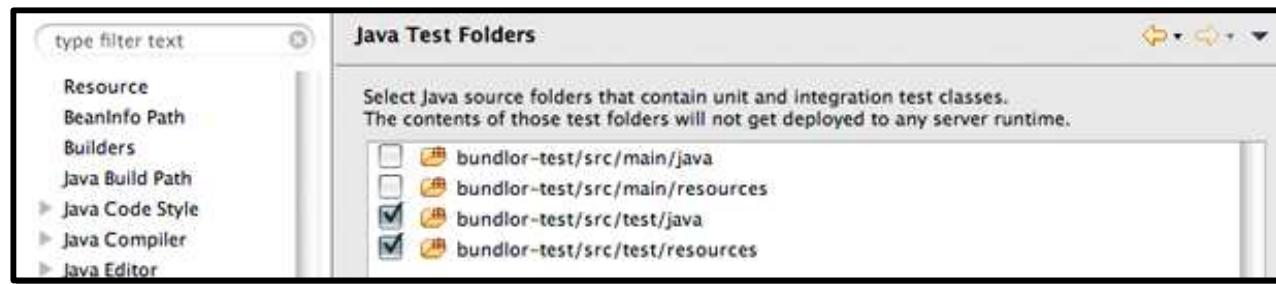
SpringSource Tool Suite

- OSGi development support
  - Bundle Classpath Container (inc. test.mf support)
  - Automated manifest creation using Bundlor
  - dm Server support in WTP
    - drag-and-drop deployment
    - repository management
    - console access
    - bundle, package, service visualizations
    - bundle inspection
    - cluster deployment on EC2, VMware LabManager

SpringSource Tool Suite
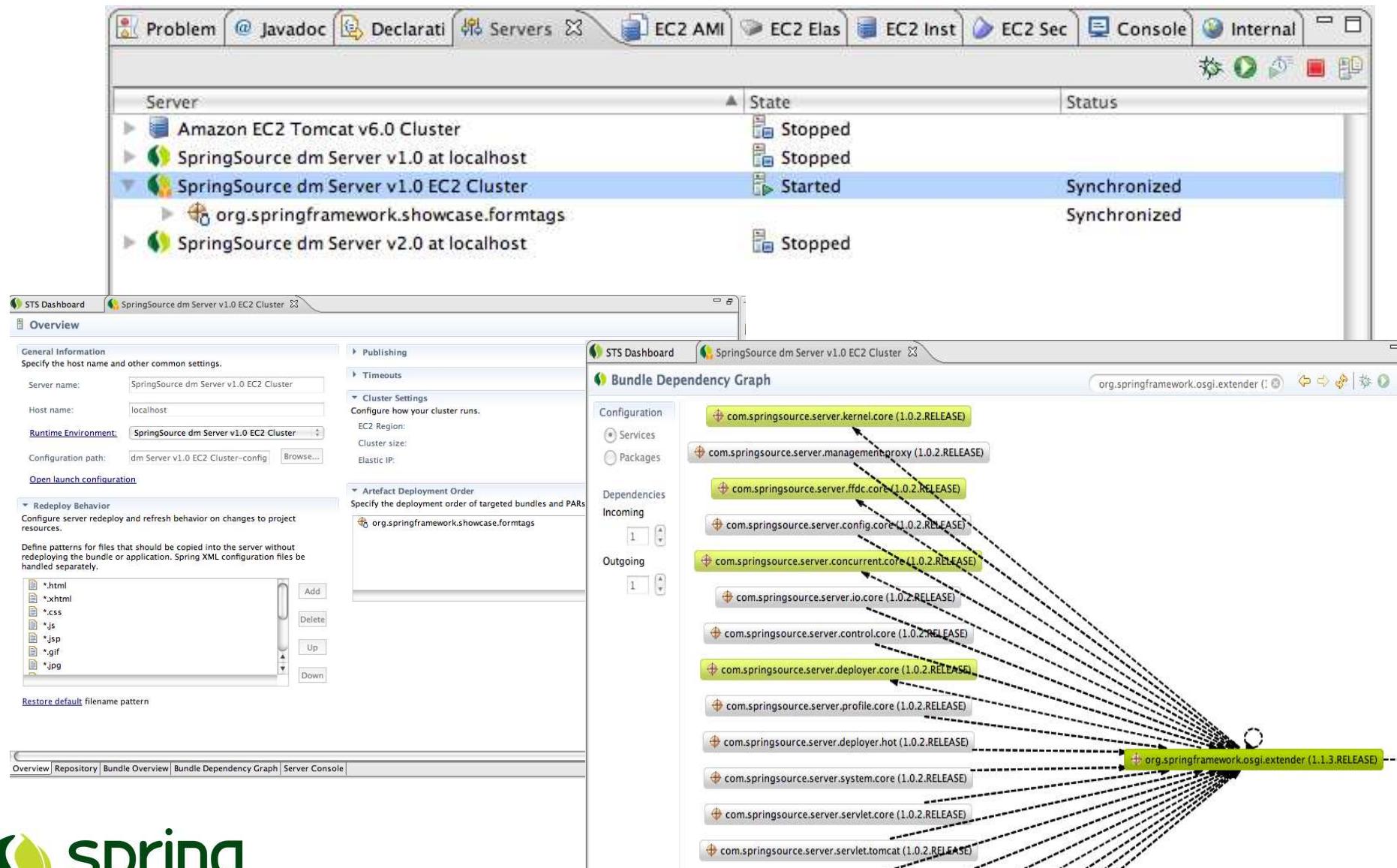
Summary

- dm Server v2.0
  - provisioning: plans, repositories
  - web application support
  - full version isolation
  - medic
  - blueprint service, other enhancements

- Supporting tools: bundlor, STS

- Keep up at blog.springosurce.com