

# **Nexus for Component Management**

## Contents

<b>1</b>	<b>Nexus for Component Management</b>	<b>1</b>
<b>2</b>	<b>Nexus for Component Management</b>	<b>1</b>
<b>3</b>	<b>Nexus for Component Management</b>	<b>1</b>
<b>4</b>	<b>But What About the Top Ten Tips?</b>	<b>2</b>
<b>5</b>	<b>And What About My Questions?</b>	<b>2</b>
<b>6</b>	<b>So What is a Component?</b>	<b>2</b>
<b>7</b>	<b>So What is a Component?</b>	<b>2</b>
<b>8</b>	<b>Development Lifecycle Integration?</b>	<b>2</b>
<b>9</b>	<b>Getting Started - Nexus OSS</b>	<b>3</b>
<b>10</b>	<b>What is a Repository?</b>	<b>3</b>
<b>11</b>	<b>Component Manager Tasks</b>	<b>3</b>
<b>12</b>	<b>Repository Concepts</b>	<b>3</b>
<b>13</b>	<b>Security</b>	<b>4</b>
<b>14</b>	<b>Search &amp; Component Information</b>	<b>4</b>
<b>15</b>	<b>Component Management</b>	<b>4</b>
<b>16</b>	<b>Component Management</b>	<b>4</b>
<b>17</b>	<b>Component Management</b>	<b>5</b>
<b>18</b>	<b>Component Management</b>	<b>5</b>
<b>19</b>	<b>Component Management with Nexus OSS</b>	<b>5</b>
<b>20</b>	<b>Managing Nexus</b>	<b>6</b>
<b>21</b>	<b>Nexus Professional - More Information</b>	<b>6</b>
<b>22</b>	<b>Nexus Professional - More Security</b>	<b>6</b>
<b>23</b>	<b>Sonatype Nexus as Central Hub</b>	<b>6</b>

---

---

<b>24</b>	<b>Scaling Nexus Deployments</b>	<b>7</b>
<b>25</b>	<b>Scaling Nexus Deployments</b>	<b>7</b>
<b>26</b>	<b>Release Process</b>	<b>7</b>
<b>27</b>	<b>Release Process</b>	<b>7</b>
<b>28</b>	<b>Nexus Professional CLM Edition</b>	<b>8</b>
<b>29</b>	<b>Sonatype CLM</b>	<b>8</b>
<b>30</b>	<b>Integrating with Nexus</b>	<b>8</b>
<b>31</b>	<b>Resources</b>	<b>9</b>

---

## 1 Nexus for Component Management

Join me on a journey

- from no component manager usage
- to full software development lifecycle integration



### Warning

Hold on tight. This is going to be a whirlwind trip.

---

## 2 Nexus for Component Management

We will move ...

- from using unknown jar files
- dependencies not being understood
- and brittle version updates

to

- enforced policies
- well understood component usage
- avoided security and license issues

## 3 Nexus for Component Management

We will move ...

- from using e.g. jars checked into version control
  - or just using Maven, Gradle, ... out of the box
  - via Nexus OSS
  - to Nexus Professional
  - and onwards to Nexus Professional CLM Edition
  - to Sontype CLM with all its tools
    - IDE integration,
    - CI integration,
    - and more
-

## 4 But What About the Top Ten Tips?

Don't worry. I got this covered. There will be lots of tips and tricks.

— Manfred Moser *Right Here (And Very Soon)*

## 5 And What About My Questions?

Go right ahead - ask any time.

But first, let me set the stage...

## 6 So What is a Component?

- All the stuff you use to create your applications
- The plumbing you don't want to write yourself
- Logging, IoC, persistence layer, ORM, ...
- And that you get easily by declaring dependencies with Maven, Gradle, nuget, npm...
- Comprises 80-90% of your application



### Warning

And you are responsible for **all** of them.

---

## 7 So What is a Component?

- But also all the parts you create and use
  - Including the application deployment packages
- JARs, WARs, EARs..
  - but also rpm and npm packages or tar.gz files

## 8 Development Lifecycle Integration?

It is an endless circle of activities:

- Research what components to use
- Implement usage of components in development
- Check component usage in QA and release process
- Monitor applications and components used in production
- Go back to the start and change versions or entire components



### Warning

It only ends when any production deployments are decommissioned and **not** when development stops.

---

## 9 Getting Started - Nexus OSS

- Installation is simple
  - Extract archive and run
- Configuration
  - `conf/nexus.properties`
  - `bin/nexus(.bat)`
- Run as a service (JSW)
  - `bin/jsw/conf/wrapper.conf`

---

**Tip**

Let's have a look at these files.

---

→ It's easy and the benefits are immediate.

## 10 What is a Repository?

- Organized storage for components
- Uses e.g. GAV coordinates for structure

→ A **Component Manager** helps with administration and usage of components stored in repositories

## 11 Component Manager Tasks

- Proxy and manage access to public repositories
- Store components that are not in public repositories
- Manage internal releases and development components

→ Facilitate internal collaboration across components and teams

## 12 Repository Concepts

- Proxy and hosted repositories
- Repository groups for easy access/aggregation
- Maven 2, NuGet and NPM

---

**Tip**

We explore all that in the user interface.

---

## 13 Security

- Privileges, roles and users
- External role mappings e.g. LDAP
- Repository targets
- Authentication from tools
  - settings.xml and more

---

**Tip**

Understanding security and possible setups is crucial to control component usage.

---

## 14 Search & Component Information

- Different search methods
- Find components in different repositories
- Find available versions - not yet used
  - Useful for component selection
- GAV coordinates and repository location

---

**Tip**

Demo time!

---

## 15 Component Management

*Supply change management for components*

- Security and authentication
- Repositories and repository targets
- Component information

→ Set the stage for first repository and component management

## 16 Component Management

Who can read?

- other teams (QA, operations)
  - remote offices
  - contractors, partner companies, public access
-

## 17 Component Management

Who can write?

- Build master
- Release engineers
- CI servers
- SNAPSHOTS vs. release

## 18 Component Management

What information is available to whom?

- development
- QA
- security
- legal
- ...

## 19 Component Management with Nexus OSS

- Internal caching and storing of components → enables collaboration
- Reduced dependency on external repositories
- One component storage location for backup, audit, control...
- Highly performant
- Reduced bandwidth usage and costs
- Efficient search
- Repository Health Check
- Some meta data
- Formats include Maven, NuGet, NPM, site, Yum and JRuby/Gems

→ Way better than manual management or ignoring the need



### **Important**

Yet easy to implement...

---



## 20 Managing Nexus

- Logging
- Support tools
- Scheduled tasks
- System feeds
- JMX
- External monitoring software

## 21 Nexus Professional - More Information

Lots more component information:

- Archive browser and Maven dependency
- Age & popularity
- Security information
- License information
- Repository health check with component details

---

**Tip**

You gotta see this!

---

## 22 Nexus Professional - More Security

More security features:

- User token
- Enterprise LDAP, Atlassian Crowd
- SSL management
- Maven settings management

## 23 Sonatype Nexus as Central Hub

images/nexus-tool-suite-integration.png

→ Nexus is a key component of your enterprise development infrastructure

---

## 24 Scaling Nexus Deployments

- One server
- Bigger server
- With failover setup
- Multiple servers for read, single write
- With load balancer

## 25 Scaling Nexus Deployments

With Nexus Professional → Smart proxy

images/nexus-smart-proxy.png

---

**Tip**

This is used for sync to OSSRH and Central from e.g. JBoss, Apache,...

---

## 26 Release Process

- Security controls write access
- Configurable with repository targets
  - Disallow deployments outside company GAV
  - Specific GAV's allowed per team
  - Or only CI server can deploy
  - Or build master

## 27 Release Process

Dedicated tooling for release process in Nexus Professional - Staging

- Temporary repository
- Rule validation upon deployment
- Configurable for repository targets
- Security controlled stages
- Releases can be dropped, if desired
- Used as input funnel for Central Repository - OSSRH

---

**Tip**

Check out the [free video training course on staging](#).

---

## 28 Nexus Professional CLM Edition

In a nutshell:

- Configurable component policies - **very** powerful
- Managed on the Sonatype CLM server
- Tied into Nexus staging

---

**Tip**

Demo time!

---

## 29 Sonatype CLM

Expands Nexus Professional CLM Edition

- Manual analysis in CLM server
- Eclipse IDE integration
- Continuous Integration Server support
- SonarQube support
- Command line scanning

## 30 Integrating with Nexus

- Lots of build tools can integrate with Maven repositories
  - Ant/Ivy, Gradle, SBT, Grails Leiningen, ...
- Lots of other repository formats supported
  - Maven, NuGet, NPM, Yum, Site, ...
- All functionality is available in REST API
- Java client for REST API available
- Plugin architecture with examples to create your own

---

**Tip**

Check out the community chapter in the Nexus book.

---

## 31 Resources

- [Repository Management with Nexus](#) - free book
  - [The Nexus Community](#)
    - with articles, videos, mailing Lists, Nexus Live and more
  - [Sonatype Support](#)
  - [Nexus Professional website](#)
  - [Nexus Professional Trial Bundle and Guide](#) - some examples can be used with Nexus OSS as well
  - [Talk to the developers/support](#) - HipChat
  - [Training classes](#)
-