{metæffekt} GmbH

# DITA Tool Chain

Reference Document

{metæffekt} GmbH

# Imprint

**{æ} DITA Toolchain**

Version:                0.18.0

**{æ} Common Content**

Version:                0.1.0
Date:                   2019-03-09

**{æ} Reference Document**

Version:                1.0
Status:                 Initial
Classification:         Public
Date:                   2019-03-09

**{metæffekt} GmbH**

https://metaeffekt.com

{metæffekt} GmbH

# Notice

All company names, organization names, license names, and product names mentioned in this documentation are used for identification purposes only. A trademark is explicitly identified as registered or unregistered trademark only if required by appropriate guidelines or license terms.

# Contents

# 1 Introduction

Getting started with *Darwin Information Typing Architecture (DITA)* and the *DITA Open Toolkit (DITA-OT)* is quite a challenge. Advantages over other documentation approaches can only be achieved once a complete infrastructure - a tool chain - is defined, established, and constantly managed.

This document is part of the {metæffekt} DITA Tool Chain. It demonstrates how content can be organized and transformed to different outputs, such as *Hypertext Markup Language, version 5 (HTML5)* and *Portable Document Format (PDF)*.

The content of this document is used as reference to demonstrate the features and capabilities of the tool chain. As such the content is often artificial.

This short introduction already illustrates the power of the toolchain to a great extend. Mind the references to the glossary. The glossary is context specific to the document that is produced and based on existing term definitions.

Glossary terms use key references. This means a reference that does not point directly to an existing file, but an identifier (key) that is then mapped to a file from the ditamap.

## 1.1 Context

The {metæffekt} DITA Tool Chain envisions to lower the hurdle for an organization to access a consistent and conceptually sound configuration that can be adapted to ogranization-specific requirements more easily.

Once the fundamentals are established the tool chain can be integrated deeper in the procedure and processes of the organization. By generating DITA content from other sources such as a issue tracker, development resources (such as Maven POMs or other project definitions) generated content can be integrated into the documents.

The ultimate goal of the {metæffekt} DITA Tool Chain is to automate the complete release documentation of a software development project. This supports incremental software development approaches.

## 1.2 Scope

The document scope is two-fold. On a general level the motivation for the {metæffekt} DITA Tool Chain is conveyed.

Second, and more prominent, features of the tool chain are explained and highlighted. The document is regarded as a reference and example for what can be done with tool chain.

## 1.3 Target Audience

This document is intented for technical writers and developers at the same sime. It is regarded essential that in particular these two groups better collaborate on writing technical documentation about the software project in focus.

# 2 Reference

The document serves as reference for selected features of DITA and the metaeffekt DITA Tool Chain. Primarily it shows how DITA content can be organized. As the result can be rendered to a PDF document and HTML5 the reference also serves as test and demonstration of the resulting document styling.

## 2.1 Tables

There are several ways to create a table using *DITA-OT*. The simpletable is quite easy to use:

| Column 1 | Column 2 |
|----------|----------|
| Monday | Get up early. |
| Tuesday | Get up late. |
| Wednesday | Stay in bed. |

However, the CALS Table is much more flexible and gives you more options.

| Column 1 | Column 2 |
|----------|----------|
| Thursday | Visit Uncle Joe |
| Friday | Get back on track. |
| Saturday | Gain faith. |

**Table 1: CALS Table Title**

Please note that the CALS table has a title. This title can be included in the table references in the backmatter of the document. Check it out.

In case you are using an WYSIWYG[1] editor there is no reason to not use the CALS table.

Please also mind that both tables are rendered differently. Actually the {metæffekt} DITA-OT customization does currently not pay particular attention to using simpletable.

## 2.2 Notes

Notes enable to highlight information.

**Note:** This is a note without further specified attributes.

Using a paragraph in a note is also possible and gives a nice result.

Yet another paragraph.

The importance attribute is currently not evaluated by the toolkit. However, the type of the note influences the graphics and the label.

---

[1] WYSIWYG: What You See Is What You Get.

**Attention:** This is note with type attention.

You should define a clear semantics of what the different types mean in your organization.

**CAUTION:** This is note with type caution.

You should define a clear semantics of what the different types mean in your organization.

**DANGER:** This is note with type danger.

You should define a clear semantics of what the different types mean in your organization.

**Fastpath:** This is note with type fastpath.

You should define a clear semantics of what the different types mean in your organization.

**Important:** This is note with type attention.

You should define a clear semantics of what the different types mean in your organization.

## 2.3 References

When writing DITA content it is possible to use all kinds of references. For example it is possible to point to a link in the web using a href. For example http://metaeffekt.org. You can also point to another chapter. For example Introduction on page 5.

Really useful are key-based references (keyref). Basically you use an alias for the reference. This alias must be mapped within the ditamap. For example Introduction on page 5. This can be explicitly useful when content is made reusable. The reusable piece want's to introduce a reference. But the reference is dangling until the content is reused in a ditamap. Then the reference must of course exist. The ditamap has to specify how the link is to be resolved.

## 2.4 Figures

This chapter demonstrates the usage of figures.

{metæffekt} GmbH
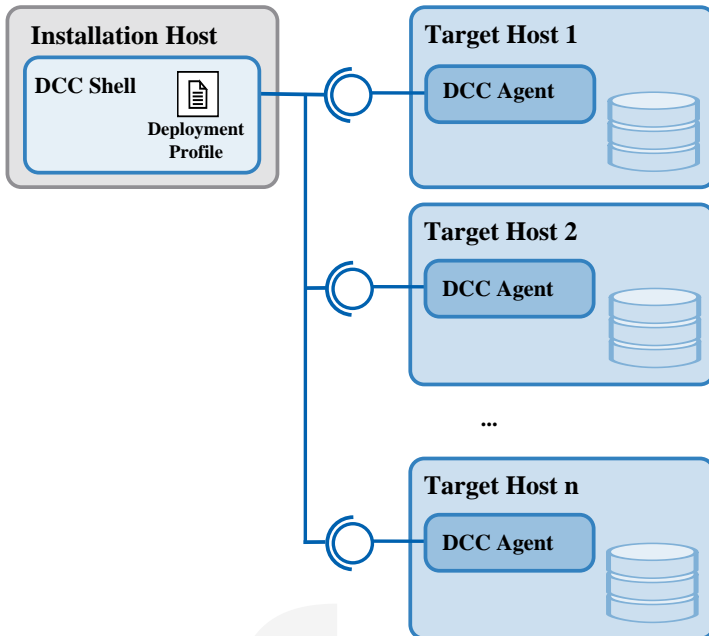


**Figure 1: Using an SVG as figure.**

SVGs - as these are XML files - can use placeholders as well. This way context specific replacements can be performed. In addition, this approach represents an option combine a custom layout with content without the need to write XSLT.

SVGs can be scaled as required. Giving best results with any print output.



**Figure 2: Using an PNG as figure.**

Non-scalable image formats have a defined resolution. On screen the resolution may still be sufficient (72dpi), but when zooming into or when printing low resolution pictures show artifacts.

{metæffekt} GmbH



**Figure 3: Using an JPG as figure.**

The features of SVG are striking in combination with the toolkit. In particular, arbitrary content can be placed as SVG figure. Transparency and alignments are working out great.

Left Aligned                    Right Aligned
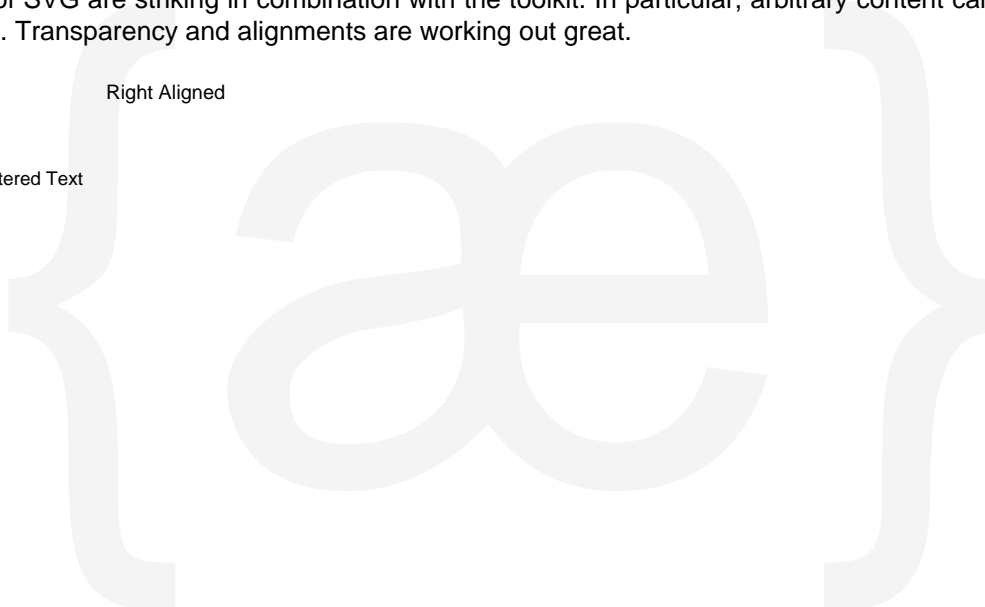
              Centered Text

**Figure 4: Testing text alignments in SVGs.**

## 2.5 Lists

1. Item 1
2. Item 2

- Item 1
- Item 2

## 2.6 Sections

**Title One**

First content.

**Title Two**

Second content.

## 2.7 Samples

This chapter shows how to embed snippets and code samples.

```
    ____  _____  _____  ____  __        __ __
   / __ \ / ___// ___/  / ___/ / /_   __   / // /
  / / / // /   / /      \__ \ / __ \ / _ \ / // /
 / /_/ // /___ / /__   ___/ // / / // __// // /
/_____/ \____/ \____/  /____//_/ /_/ \___//_//_/

Welcome to the DCC Shell.
Hit TAB to see a list of options, enter [help] for help and [exit] to exit.
dcc>
```

**Figure 5: Showing the output of a command line tool.**

```
<pluginManagement>
    <plugins>
        <plugin>
            <groupId>org.metaeffekt.dita</groupId>
            <artifactId>ae-dita-maven-plugin</artifactId>
            <version>0.18.0</version>
        </plugin>
    </plugins>
</pluginManagement>
```

**Figure 6: Showing some XML content.**

Unfortunately no syntax-highlighting is intergrated yet.

```
</**
 * A {@link Binding} is capable of connecting two capabilities (usually a provided with a required
 * {@link Capability}) with each other. Please note that the compatibility of the {@link Capability}
 * (shared {@link CapabilityDefinition}) is not enforced. The implementation however expects that
 * the attribute keys of the target are covered by the source {@link Capability}.
 *
 * @author Karsten Klein
 */
public class Binding extends AbstractElement {
    ...
}
```

**Figure 7: Showing some JAVA content.**

# 3 Reference Bill of Material

The following chapters illustrates how content of a Bill of Material is rendered. The content was extracted from another metaeffekt project. The content is not relevant for the reference project, but is only used here for illustration purposes.

## 3.1 {æ} Common Content 0.1.0

The *software distribution* covered by this document contains *subcomponents* and *components* restricted by individual license terms.

This document is part of the *software distribution annex* which addresses the obligations deduced from the *effective licenses*. The software distribution is made available or transferred only in corresponding combination with the software distribution annex.

The component licenses and notice files are provided as part of the software distribution annex. Licenses may be omitted in case the terms are restricting the distribution.

License obligations of the individual components, as far as these can be addressed in documentation, are detailed in section Copyright Notices on page 16.

### 3.1.1 {metæffekt} GmbH Proprietary Components

Proprietary components included in the covered software distribution developed and supplied by {metæffekt} GmbH are made available - if not explicitly marked otherwise - under Apache License 2.0.

### 3.1.2 Third-Party Proprietary Components

The covered software distribution contains proprietary third-party components. These are generally licensed individually under vendor-specific license terms, which further apply towards the licensee of {æ} DITA Toolchain.

License and contract information regarding proprietary licenses are covered in section Copyright Notices on page 16. License texts are included in the corresponding software distribution annex when permitted by the terms of the license and/or contract.

### 3.1.3 Open Source Components

In general, for each and every open source component the individual terms and conditions of the *associated license* apply. The licensee of an open source component is granted a non-exclusive right of use by the respective right holders restricted by the stipulated terms and conditions defined in the associated license. In particular, liability and warranty regulations of the individual open source license apply.

Specific liability and warranty definitions of {æ} DITA Toolchain, if any, are subject to the contractual agreement under which the software distribution is made available. Specific liability definitions are unaffected by the terms of the individual open source licenses.

In case a license demands the provision of source code for a component, the source code is made available by adding source code artifacts to the software distribution annex or by making the source code available as individually outlined in the Copyright Notices on page 16.

### 3.1.4 Public Domain Components

Public domain components have generally waived all property rights. Nevertheless, public domain components are associated with terms that need to be addressed in compliance.

{metæffekt} GmbH

## 3.1.5 Import / Export Information

The country, the software distribution is supplied to, may have restrictions on the import, possession, and/or use of the software. Furthermore, the export or re-export of the software distribution to other countries may be restricted or even prohibited.

Before using or re-exporting this distribution check the country specific laws, regulations and policies. For more information, see also http://www.wassenaar.org.

This software distribution does not include any component classified as restricted to export/import.

Please note that {metæffekt} GmbH does not take any responsibility for the correctness, actuality, and content of the above external links.

## 3.1.6 Import / Export Information

This software distribution includes components, which are subject to export/import control. The country, the software distribution is supplied to, may have restrictions on the import, possession, and/or use of the software. Furthermore, the export or re-export of the software distribution to other countries may be restricted or even prohibited.

Before using or re-exporting this distribution check the country specific laws, regulations and policies. For more information, see also http://www.wassenaar.org.

The following table provides export/import details on specific software component included in this software distribution:

| Component Name | Import / Export Classification Details |
|---|---|
| Bouncy Castle | The Bouncy Castle artifacts contain cryptographic software and include additional crypto providers. Classification details can be found online: http://www.bouncycastle.org/wiki/display/JA1/Frequently+Asked+Questions. |
| Jasypt | Java Simplified Encryption API (Jasypt) uses the Java Cryptography Extension API and as such is classified for export control. See details online: http://www.jasypt.org/faq.html#what-is-jasypts-export-classification. |
| Oracle JRE | The Oracle Java Runtime Environment (Oracle JRE) contains cryptographic software. For more details consult the Oracle Binary Code License agreement on http://www.oracle.com/technetwork/java/javase/terms/license/index.html. For further details on the Oracle Global Trade Compliance check http://www.oracle.com/us/products/export/index.html. |

**Table 2: Import / Export Classification**

Please note that {metæffekt} GmbH does not take any responsibility for the correctness, actuality, and content of the above external links.

# 3.2 Bill of Materials

Subsequently all included third-party components and individual *artifacts* are listed with references to the licenses and/or contracts.

## 3.2.1 Third-Party Artifacts

| Group Id | Artifact Id | Version | License |
|---|---|---|---|
| | bootstrap-theme (css) | 3.3.2 | MIT License |
| | bootstrap (css) | 3.3.2 | MIT License |
| | bootstrap (js) | 3.3.2 | MIT License |
| | glyphicons-halflings (eot) | 3.3.2 | MIT License |
| | glyphicons-halflings (svg) | 3.3.2 | MIT License |
| | glyphicons-halflings (ttf) | 3.3.2 | MIT License |
| | glyphicons-halflings (woff) | 3.3.2 | MIT License |

{metæffekt} GmbH

| Group Id | Artifact Id | Version | License |
|---|---|---|---|
| | glyphicons-halflings (woff2) | `3.3.2` | MIT License |
| | jquery (js) | `1.11.1` | MIT License |
| ant-contrib | ant-contrib | `1.0b3` | Apache License Version 1.1 |
| cglib | cglib | `3.2.4` | Apache License Version 2.0 |
| com.google.guava | guava | `19.0` | Apache License Version 2.0 |
| commons-codec | commons-codec | `1.10` | Apache License Version 2.0 |
| commons-collections | commons-collections | `3.2.2` | Apache License Version 2.0 |
| commons-fileupload | commons-fileupload | `1.3.2` | Apache License Version 2.0 |
| commons-io | commons-io | `2.5` | Apache License Version 2.0 |
| commons-lang | commons-lang | `2.6` | Apache License Version 2.0 |
| jline | jline | `2.12.1` | BSD 3-Clause License |
| net.jcip | jcip-annotations | `1.0` | Creative Commons Public Domain |
| org.apache.ant | ant-commons-logging | `1.9.7` | Apache License Version 2.0 (with-subcomponents) |
| org.apache.ant | ant-launcher | `1.9.7` | Apache License Version 2.0 (with-subcomponents) |
| org.apache.ant | ant | `1.9.7` | Apache License Version 2.0 (with-subcomponents) |
| org.apache.camel | camel-core | `2.16.2` | Apache License Version 2.0 |
| org.apache.camel | camel-restlet | `2.16.2` | Apache License Version 2.0 |
| org.apache.camel | camel-spring | `2.16.2` | Apache License Version 2.0 |
| org.apache.commons | commons-exec | `1.3` | Apache License Version 2.0 |
| org.apache.commons | commons-lang3 | `3.4` | Apache License Version 2.0 |
| org.apache.httpcomponents | httpclient | `4.5.2` | Apache License Version 2.0 |
| org.apache.httpcomponents | httpmime | `4.5.2` | Apache License Version 2.0 |
| org.apache.httpcomponents | httpcore | `4.4.5` | Apache License Version 2.0 |
| org.apache.james | apache-mime4j-core | `0.7.2` | Apache License Version 2.0 |
| org.apache.logging.log4j | log4j-api | `2.7` | Apache License Version 2.0 |
| org.apache.logging.log4j | log4j-core | `2.7` | Apache License Version 2.0 |
| org.apache.logging.log4j | log4j-slf4j-impl | `2.7` | Apache License Version 2.0 |
| org.apache.velocity | velocity | `1.7` | Apache License Version 2.0 |
| org.bouncycastle | bcpkix-jdk15on | `1.55` | Bouncy Castle License |
| org.bouncycastle | bcprov-jdk15on | `1.55` | Bouncy Castle License |
| org.ow2.asm | asm | `5.0.4` | BSD 3-Clause License |
| org.restlet.jee | org.restlet.ext.fileupload | `2.3.1` | Apache License Version 2.0 |
| org.restlet.jee | org.restlet.ext.httpclient | `2.3.1` | Apache License Version 2.0 |
| org.restlet.jee | org.restlet.ext.servlet | `2.3.1` | Apache License Version 2.0 |
| org.restlet.jee | org.restlet | `2.3.1` | Apache License Version 2.0 |
| org.slf4j | jcl-over-slf4j | `1.7.21` | MIT License |
| org.slf4j | jul-to-slf4j | `1.7.21` | MIT License |

{metæffekt} GmbH

| Group Id | Artifact Id | Version | License |
|---|---|---|---|
| org.slf4j | slf4j-api | `1.7.21` | MIT License |
| org.springframework.shell | spring-shell | `1.1.0.`<br>`RELEASE` | Apache License Version 2.0 |
| org.springframework | spring-aop | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |
| org.springframework | spring-beans | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |
| org.springframework | spring-context-support | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |
| org.springframework | spring-context | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |
| org.springframework | spring-core | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |
| org.springframework | spring-expression | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |
| org.springframework | spring-tx | `4.3.2.`<br>`RELEASE` | Apache License Version 2.0 (with-subcomponents) |

## 3.2.2 Third-Party Components

| Component | File(s) | License |
|---|---|---|
| Apache Ant Contrib | `ant-contrib-1.0b3.jar` | Apache License Version 1.1 |
| Apache Ant | `ant-commons-logging-1.9.7.jar`<br>`ant-launcher-1.9.7.jar`<br>`ant-1.9.7.jar` | Apache License Version 2.0 (with-subcomponents) |
| Apache Camel | `camel-core-2.16.2.jar`<br>`camel-restlet-2.16.2.jar`<br>`camel-spring-2.16.2.jar` | Apache License Version 2.0 |
| Apache Commons Codec | `commons-codec-1.10.jar` | Apache License Version 2.0 |
| Apache Commons Collections | `commons-collections-3.2.2.jar` | Apache License Version 2.0 |
| Apache Commons Exec | `commons-exec-1.3.jar` | Apache License Version 2.0 |
| Apache Commons File Upload | `commons-fileupload-1.3.2.jar` | Apache License Version 2.0 |
| Apache Commons IO | `commons-io-2.5.jar` | Apache License Version 2.0 |
| Apache Commons Lang | `commons-lang-2.6.jar`<br>`commons-lang3-3.4.jar` | Apache License Version 2.0 |
| Apache HTTP Components | `httpclient-4.5.2.jar`<br>`httpcore-4.4.5.jar`<br>`httpmime-4.5.2.jar` | Apache License Version 2.0 |
| Apache James Mime4J | `apache-mime4j-core-0.7.2.jar` | Apache License Version 2.0 |
| Apache Log4J | `log4j-api-2.7.jar`<br>`log4j-core-2.7.jar`<br>`log4j-slf4j-impl-2.7.jar` | Apache License Version 2.0 |
| Apache Velocity | `velocity-1.7.jar` | Apache License Version 2.0 |
| ASM | `asm-5.0.4.jar` | BSD 3-Clause License |

| Component | File(s) | License |
|-----------|---------|---------|
| Bootstrap | `bootstrap-theme-3.3.2.min.css`<br>`bootstrap-3.3.2.min.css`<br>`bootstrap-3.3.2.min.js`<br>`glyphicons-halflings-regular.eot`<br>`glyphicons-halflings-regular.svg`<br>`glyphicons-halflings-regular.ttf`<br>`glyphicons-halflings-regular.woff`<br>`glyphicons-halflings-regular.woff2` | MIT License |
| Bouncy Castle | `bcpkix-jdk15on-1.55.jar`<br>`bcprov-jdk15on-1.55.jar` | Bouncy Castle License |
| GCLib | `cglib-3.2.4.jar` | Apache License Version 2.0 |
| Google Guava | `guava-19.0.jar` | Apache License Version 2.0 |
| JCip Annotations | `jcip-annotations-1.0.jar` | Creative Commons Public Domain |
| JLine | `jline-2.12.1.jar` | BSD 3-Clause License |
| JQuery | `jquery-1.11.1.min.js` | MIT License |
| Restlet | `org.restlet.ext.fileupload-2.3.1.jar`<br>`org.restlet.ext.httpclient-2.3.1.jar`<br>`org.restlet.ext.servlet-2.3.1.jar`<br>`org.restlet-2.3.1.jar` | Apache License Version 2.0 |
| SLF4J | `jcl-over-slf4j-1.7.21.jar`<br>`jul-to-slf4j-1.7.21.jar`<br>`slf4j-api-1.7.21.jar` | MIT License |
| Spring Framework | `spring-aop-4.3.2.RELEASE.jar`<br>`spring-beans-4.3.2.RELEASE.jar`<br>`spring-context-support-4.3.2.RELEASE.jar`<br>`spring-context-4.3.2.RELEASE.jar`<br>`spring-core-4.3.2.RELEASE.jar`<br>`spring-expression-4.3.2.RELEASE.jar`<br>`spring-tx-4.3.2.RELEASE.jar` | Apache License Version 2.0 (with-subcomponents) |
| Spring Shell | `spring-shell-1.1.0.RELEASE.jar` | Apache License Version 2.0 |

## 3.3 Licenses in Effect

| License |
|---------|
| Apache License Version 1.1 |
| Apache License Version 2.0 |
| Apache License Version 2.0 (with-subcomponents) |
| BSD 3-Clause License |
| Bouncy Castle License |
| Creative Commons Public Domain |
| MIT License |

## 3.4 Copyright Notices

### Apache License Version 1.1

| Component | Group Id / Artifact Id | Version | Notice |
|---|---|---|---|
| Apache Ant Contrib | ant-contrib / ant-contrib | 1.0b3 | This product includes software developed by the Ant-Contrib project (http://sourceforge.net/projects/ant-contrib). |

### Apache License Version 2.0 (with-subcomponents)

| Component | Group Id / Artifact Id | Version | Notice |
|---|---|---|---|
| Apache Ant | org.apache.ant / ant | 1.9.7 | This distribution includes Apache Ant. Apache Ant is provided under the terms and conditions of the Apache License Version 2.0 with further ammendments covering licenses of sub-components contained in the Apache Ant binaries. Apache Ant uses artifacts provided under the W3C Software Notice and License; SAX2 (public domain). Please check the details in the license and notice files provided with this distribution. |
| Spring Framework | org.springframework / spring-core | 4.3.2.RELEASE | This distribution includes the Spring Framework. The Spring Framwork is provided under the terms and conditions of the Apache License Version 2 with further ammendments of licenses of sub-components contained in the distributed package. Spring Framework uses ASM 4.0 (BSD 3-Clause License) Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved; CGLIB 3.0 (Apache License Version 2). |

### BSD 3-Clause License

| Component | Group Id / Artifact Id | Version | Notice |
|---|---|---|---|
| JLine | jline / jline | 2.12.1 | This distribution includes JLine which is provided under the terms and conditions of the BSD 3-Clause License. Copyright (c) 2002-2016, the original author or authors. All rights reserved. |
| ASM | org.ow2.asm / asm | 5.0.4 | This distribution includes ASM which is provided under the terms and conditions of the BSD 3-Clause License. Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved. |

### Bouncy Castle License

| Component | Group Id / Artifact Id | Version | Notice |
|---|---|---|---|
| Bouncy Castle | org.bouncycastle / bcpkix-jdk15on | 1.55 | This distribution includes Bouncy Castle provided under the Bouncy Castle License. Copyright (c) 2000-2016 The Legion of the Bouncy Castle Inc. (http://www.bouncycastle.org). |

{metæffekt} GmbH

## MIT License

| Component | Group Id / Artifact Id | Version | Notice |
|---|---|---|---|
| Bootstrap | bootstrap | 3.3.2 | This distribution includes Bootstrap provided under the MIT License. Copyright (c) 2011-2016 Twitter, Inc. |
| Bootstrap | glyphicons-halflings | 3.3.2 | This distribution includes Bootstrap provided under the MIT License. Copyright (c) 2011-2016 Twitter, Inc. |
| JQuery | jquery | 1.11.1 | This distribution includes JQuery provided under the MIT License. Copyright (c) 2005, 2014 jQuery Foundation, Inc. |
| SLF4J | org.slf4j / jcl-over-slf4j | 1.7.21 | This distribution includes SLF4J. SLF4J is provided under the terms and conditions of the MIT License. SLF4J is Copyright (c) 2004-2014 QOS.ch. All rights reserved. |

# Part

# I

# 4 Part Introduction

This is the introduction of the part.

## 4.1 Part Content

This is a the content of a part. Usually there would be much more content, such that it is adequate to organize the document is parts.

## 4.2 Markdown Test

This test content was taken from https://github.com/jelovirt/dita-ot-markdown and is used under Apache License 2.0.

Paragraph Text *test* and **list**:

- hyphen
- list
- item
- asterix
  - list
- item
  - nested
1. ordered
2. list
3. item
   1. nested
   2. list

**Codeblock**

Code example on `for` loop:

```
for i in items:
      println(i)
```

Fenced block:

```
items.foreach(println)
```

{metæffekt} GmbH

## Tables

| First Header | Second Header | Third Header |
|---|---|---|
| First row | Data | Very long data entry |
| Second row | **Cell** | *Cell* |

**Table 3: simple_table**

| | Grouping | |
|---|---|---|
| **First Header** | **Second Header** | **Third Header** |
| Content | *Long Cell* | |
| Content | **Cell** | Cell |
| New section | More | Data |

**Table 4: Prototype table][reference_table**

## Links

- Markdown Test on page 18
- External

## Keys

This is Markdown Test on page 18 reference-style link.

This is Markdown Test on page 18 reference-style link.

This is Markdown Test on page 18 reference-style link.

This is Markdown Test on page 18 reference-style link.

This is an example reference-style link.

This is  reference-style link.

# A Appendix

## A.1 Maven Support

Apache Maven is the supported way to use the {metæffekt} DITA Toolchain. No further software prerequisites are required. A customized DITA-OT Toolkit is used implicitly when using the metaeffekt DITA Plugin.

### A.1.1 Installing Apache Maven

Apache Maven is the primary prerequisite for using the {metæffekt} DITA Tool Chain.

1. Download the latest Apache Maven 3.3 release from http://maven.apache.org/download.cgi.

---

**Note:** JAVA Required.

JAVA 1.7 is required to run Apache Maven 3.3. In general Oracle JAVA 1.8 is required by the {metæffekt} DITA Tool Chain.

In case JAVA is not available, it can be downloaded from https://java.com/en/download/.

---

2. Follow the instruction provided on http://maven.apache.org/install.html
   Test the installation was successful by executing the command

```
mvn -version
```

on the console. When Apache Maven was successfully installed the command provides the following output:

```
Apache Maven 3.3.9 ...
```

### A.1.2 Maven Configuration

Maven uses XML project model descriptors also known as POM files. The structure of these POM files is defined on LINK.

To enable the metaeffekt DITA Plugin specify the following plugin:

The configuration expects a ditamap at the location `src/main/dita/${project.artifactId}`

## A.2 DITA Conventions

The {metæffekt} DITA Tool Chain uses numerous conventions to structure content within a documentation project.

---

{metæffekt} GmbH

# List of Tables

# Glossary

**PDF**
Portable Document Format developed by Adobe®.

**Artifact**
An artifact is a physical file, such as an executable binary, an archive, a document, a text or configuration file.

**Associated Licenses**
An artifact can be distributed by a provider under different licenses depending on the context or the channels in which the artifact is made available or transferred. These licenses are associated with the artifact when downloading or receiving the artifact and are referred to as associated licenses. In contrast to the associated licenses, the effective licenses are the licenses under which the artifact is further distributed. The associated and effective licenses of an artifact can differ, but are restricted by the terms of the associated licenses.

**Subcomponent**
A subcomponent is a component embedded in a larger component. A subcomponent may have dedicated associated licenses.

**Darwin Information Typing Architecture (DITA)**
OASIS defined standard promoting the creation of standard information types and domain specific markup. DITA defined content can be aggregated in a DITA map and rendered to different output formats such as PDF, HTML or RTF.

**DITA Open Toolkit**
The DITA-OT supports to render DITA-based content to different output formats. The DITA-OT is an open source endeavor that supports all available versions of the DITA specification.

**Effective Licenses**
There are various cases where a component is made available by a provider under multiple associated licenses. The effective licenses are the selected licenses relevant for the software distribution:

- A component is provided with a dual or multiple licensing option where either one or the other license can be used for distribution. In this case a single effective license needs to be determined.

- A component is provided with a license in a specific or later version. In this case a version of the license has to be selected as effective license.

- For a component, which contains subcomponents the license of the component and the individual licenses of the subcomponents are effective.

- The associated license of a component allows to sublicense the component under a different license. In this case the chosen license is the effective license.

**HTML5**
Hypertext Markup Language, version 5

**Software Distribution**
A software distribution is packaged for delivering the software to a recipient. Depending on context and relationship between the provider and recipient of a distribution the license terms of the covered components have to be addressed. A software distribution may be an executable, an archive or a filesystem image.

**Software Distribution Annex**
The software distribution annex aggregates all information required to fulfill the obligations when making available or transferring the software distribution. It covers the following elements:

- The associated and effective license terms and notices of the individual components.

- The source code of components if stipulated by the effective license.

- Documentation including the bill of materials of all third-party components with appropriate license and notice information.

**Component**

{metæffekt} GmbH

A component is a group of one or more artifacts. While artifacts are physical files, the component can be regarded as logical aggregate. Licensing information applies on component-level and is consistent within the aggregated artifacts.