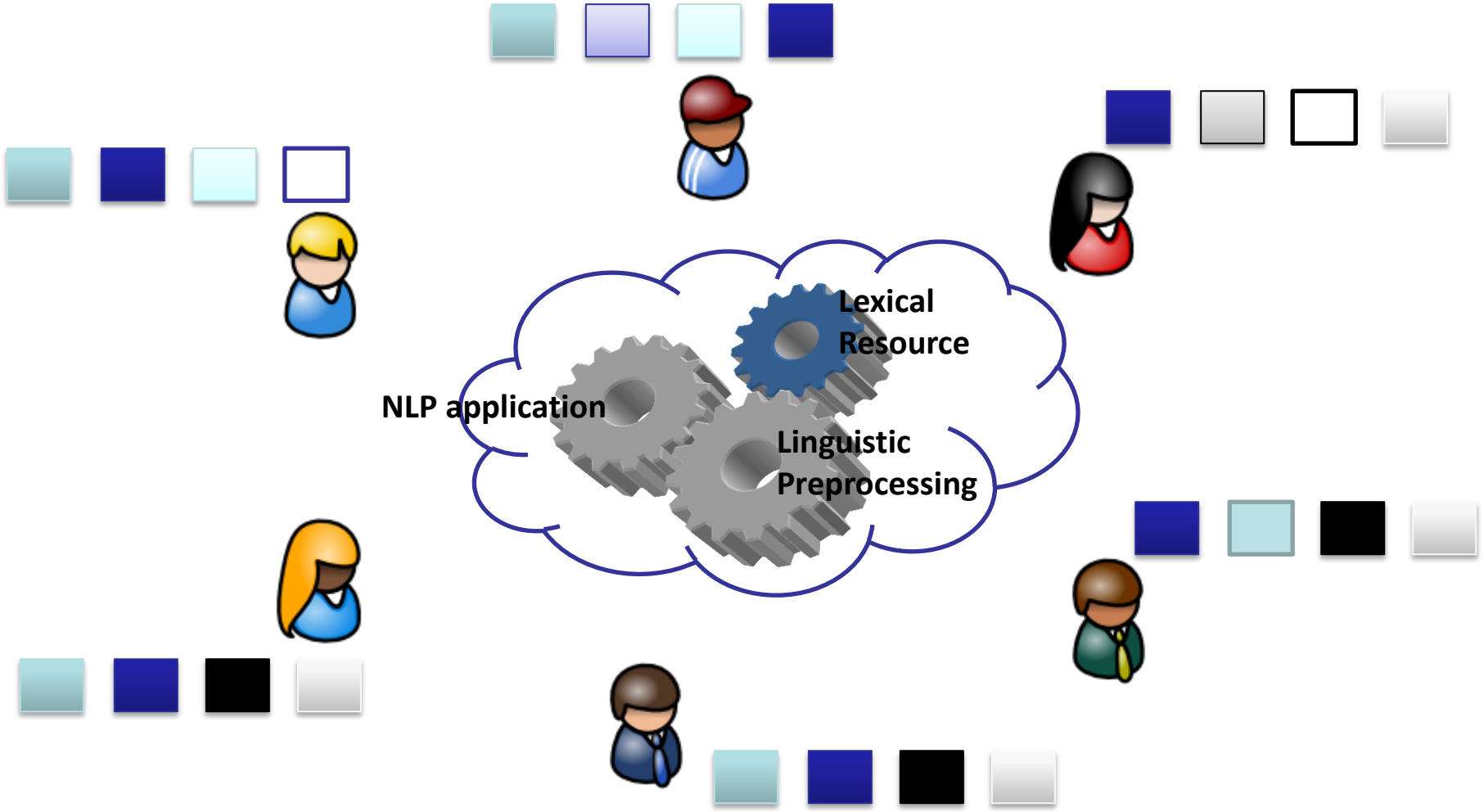
Darmstadt Knowledge Processing Repository: Collaborative software development with agile practices

TECHNISCHE UNIVERSITÄT DARMSTADT

Dr. Judith Eckle-Kohler, Richard Eckart de Castilho

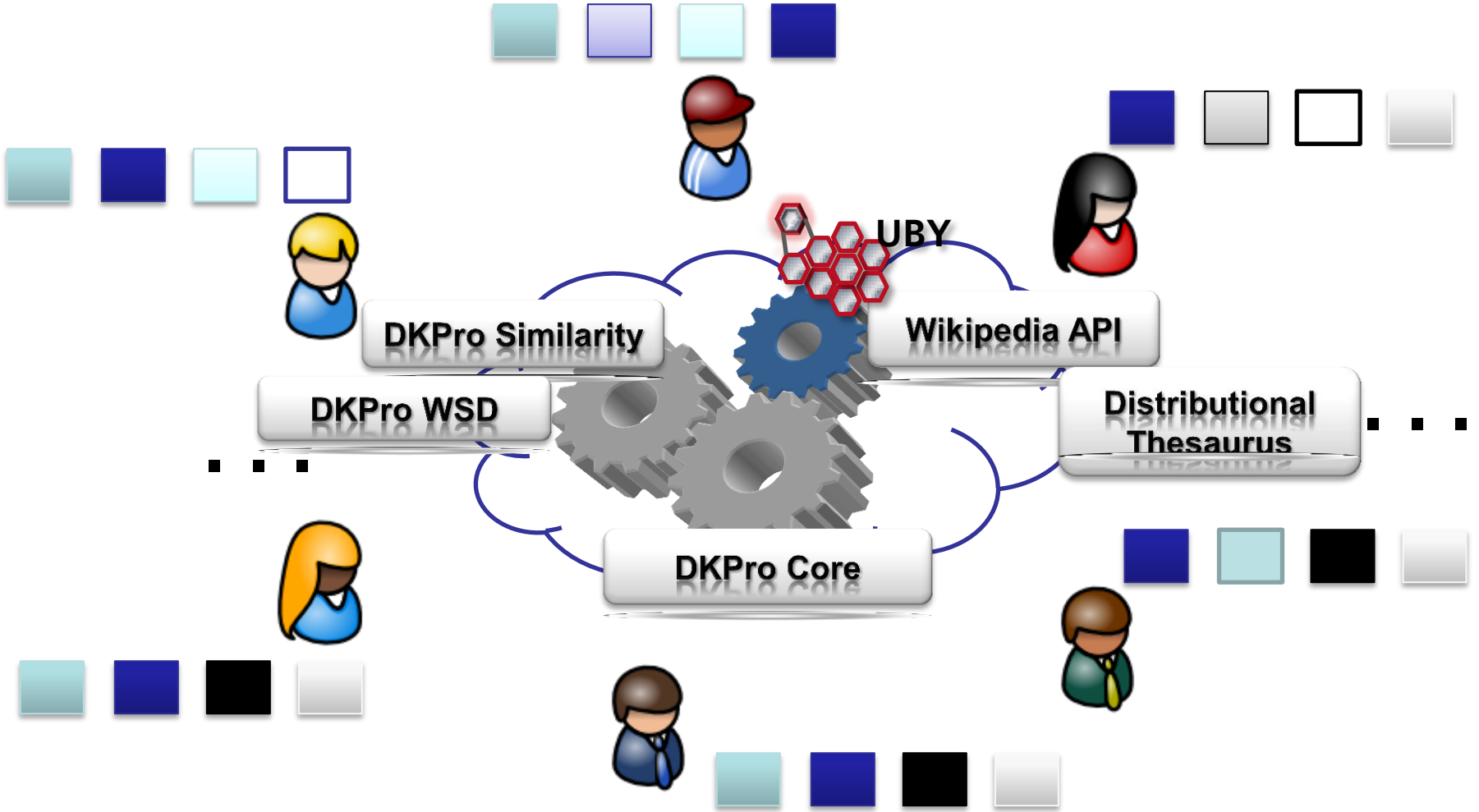
Darmstadt Knowledge Processing Repository

TECHNISCHE UNIVERSITÄT DARMSTADT

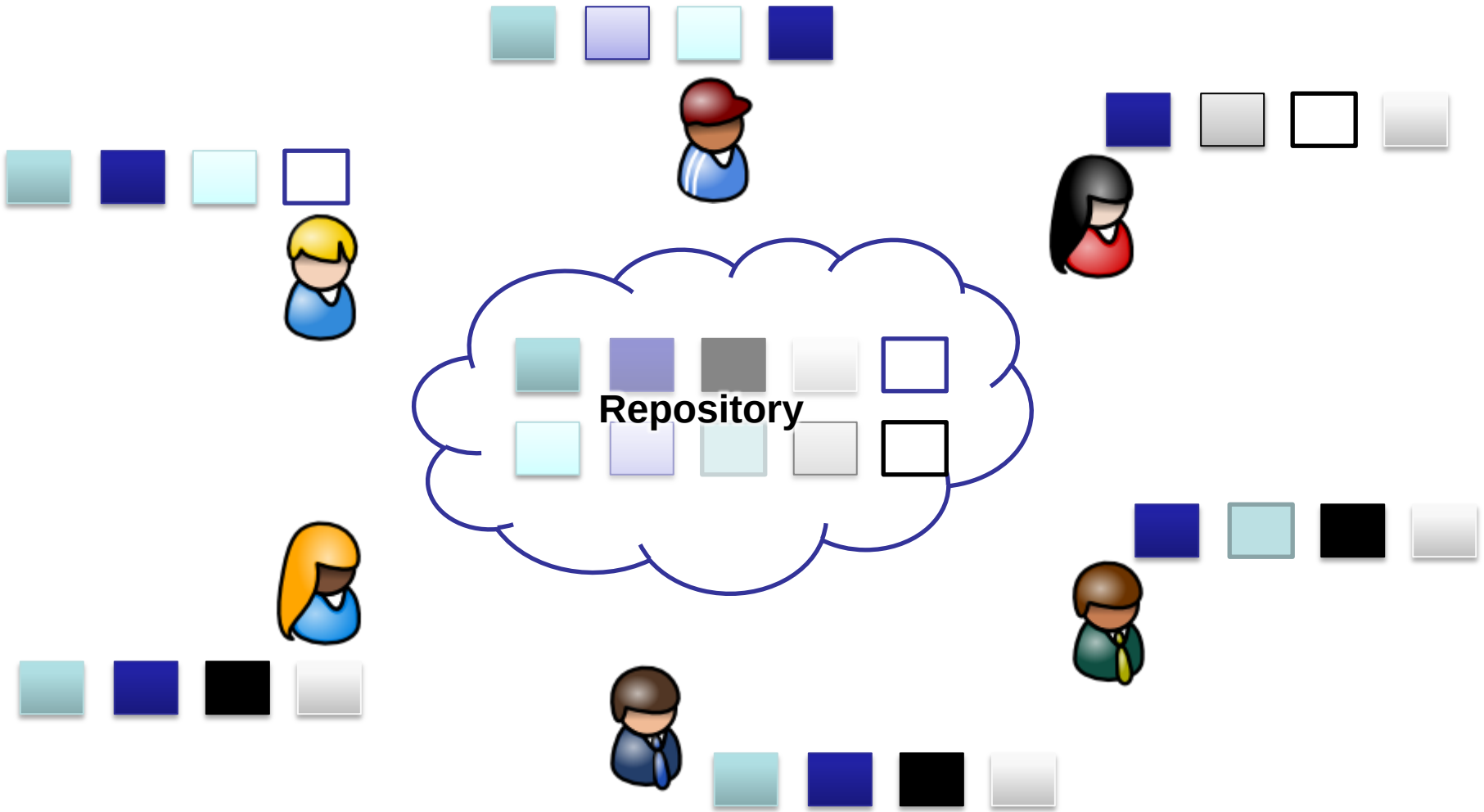


Darmstadt Knowledge Processing Repository

TECHNISCHE UNIVERSITÄT DARMSTADT



Darmstadt Knowledge Processing Repository



What is agile?

TECHNISCHE UNIVERSITÄT DARMSTADT

Wiktionary: "*Having the faculty of quick motion in the limbs; apt or ready to move; nimble; active*"

Agile Manifesto (2001):

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

… should not be taken too literally …

- **But: there are agile practices that support collaborative software development and quality assurance**
- **Some of these agile practices are used in DKPro**

Agile practices

TECHNISCHE UNIVERSITÄT DARMSTADT

- Fundamental:
  - The team
  - **Iterative / incremental** software development
- Agile practices include:
  - Version control
  - Automated build
  - Unit Tests
  - Continuous Integration
  - Refactoring
  - Pair Programming
  - Test Driven Development
  - …

Why bother with quality assurance?

That‘s why:

NLP-research heavily builds on a large NLP infrastructure (pre-processing components, lexical resources, NLP tasks and applications …)

At UKP, this infrastructure is collaboratively developed and maintained.

The more and the better components a joint repository contains, the more every single researcher can profit from these components in her individual research.

Collaborative software development

TECHNISCHE UNIVERSITÄT DARMSTADT

… requires (open) communication:

- Issue Tracker:
  - Internal Issue Tracker
  - Google Code (DKPro Core etc.) or Apache UIMA issue tracker

- Mailing Lists:
  - internal mailing lists and commit mailing Lists
  - Google Groups mailing lists: dkpro-core-developers, dkpro-core-users, …
  - External Mailing Lists: UIMA, OpenNlp, ClearTK, Stanford CoreNLP …
  - stackoverflow.com (Q&A site for programmers)

- Face-to-Face Meetings

Collaborative software development

TECHNISCHE UNIVERSITÄT DARMSTADT

… requires (open) documentation:

- Internal documentation (Wiki, …)
- Open documentation:
  - UKP Open Source Projects @ Google Code
    - DKPro Core
    - DKPro Similarity
    - UBY
    - …
    - ... and DKPro Tutorials

Agile practices@UKP

TECHNISCHE UNIVERSITÄT DARMSTADT

- Development environment
- Version control
- Unit Tests
- Automated builds
- Artifact repository
- Continuous Integration = automated builds **and** unit testing
- Refactoring requires unit testing

Agile practices@UKP – Tools

TECHNISCHE UNIVERSITÄT DARMSTADT

- Development environment — Eclipse 
- Version control — Subclipse 
- Unit Tests — JUnit
- Automated builds — Maven  
- Artifact repository — Artifactory 
- Continuous Integration — Jenkins 
- Refactoring — JUnit, Eclipse

Continuous Integration

TECHNISCHE UNIVERSITÄT DARMSTADT

Continuous Integration (CI)

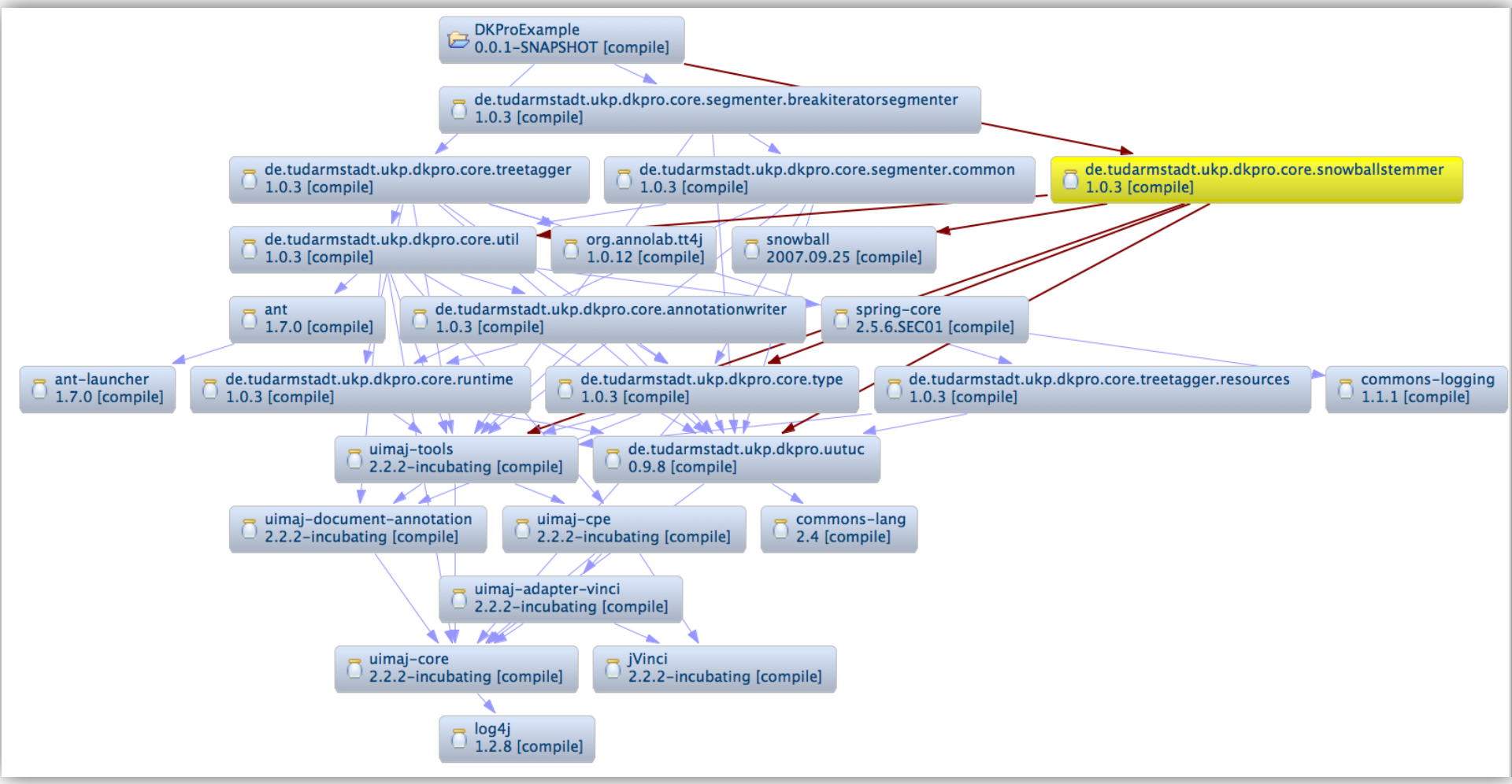
= automated builds **and** unit testing

Tools required for CI:

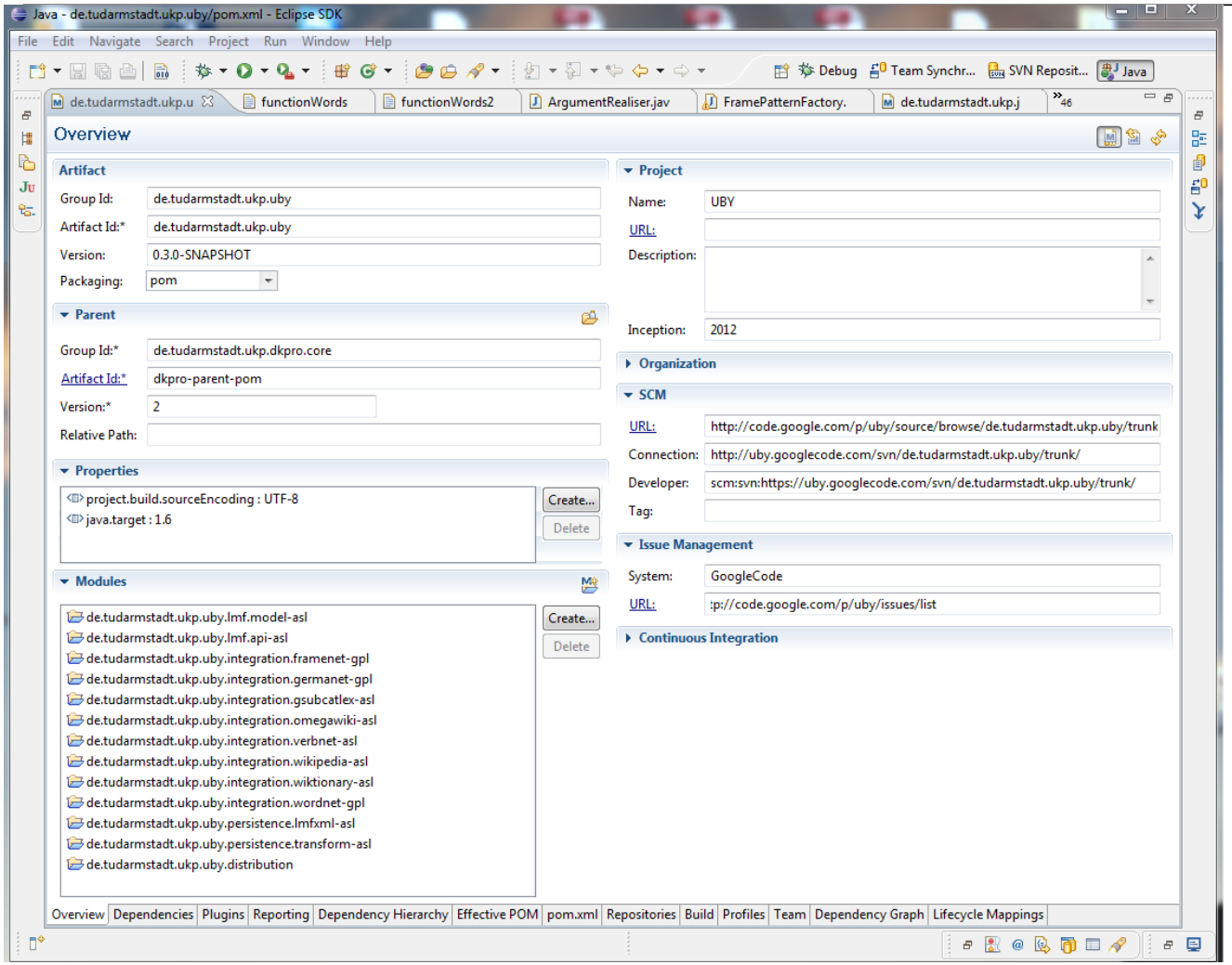
- Version control
- Testing tools
- Build management tools for automated builds

Why build automation?
Dependencies between software components

TECHNISCHE UNIVERSITÄT DARMSTADT



Build-Management with Maven



Project Object Model
`pom.xml`

Build-Management with Maven

TECHNISCHE UNIVERSITÄT DARMSTADT

„convention over configuration“

maven

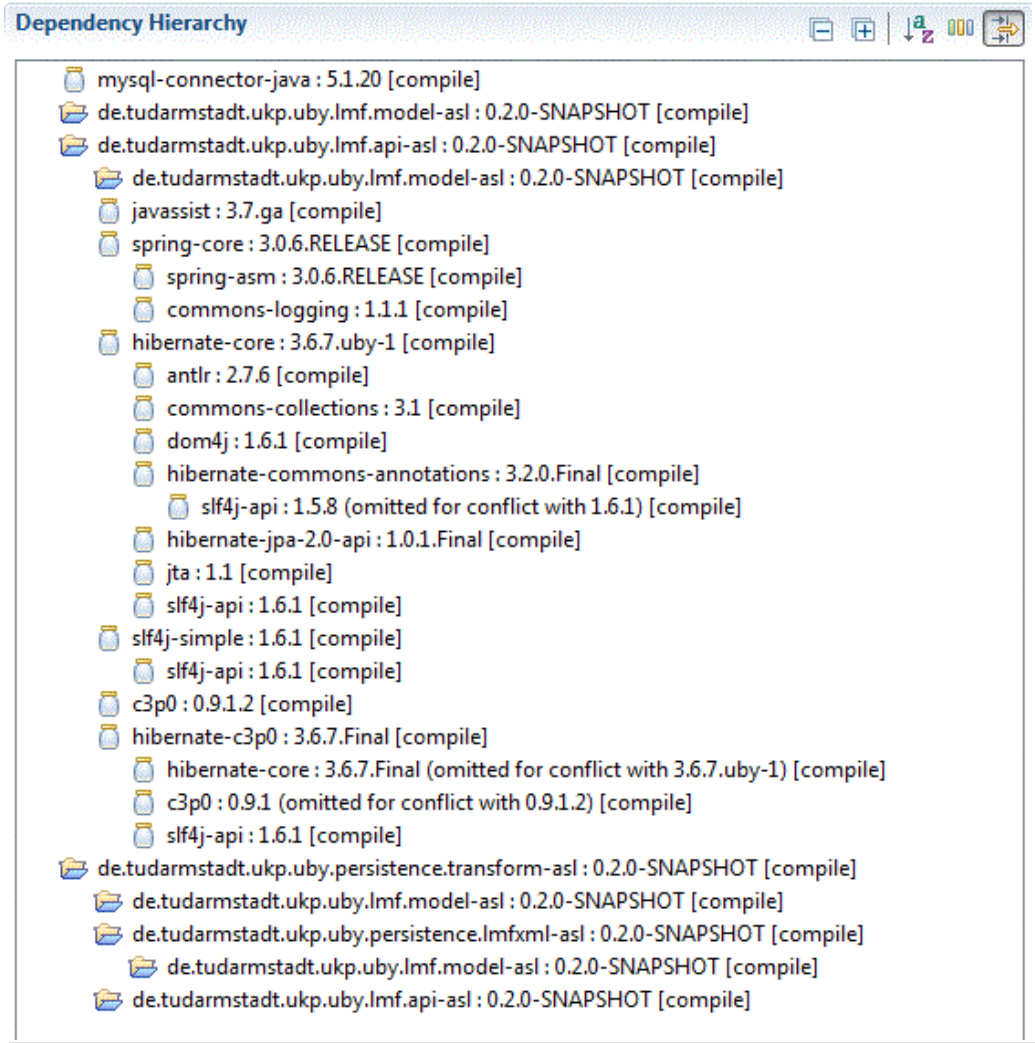
Example:

Maven Project: standardized directory structure

```
/de.company.department.hello/src/main/java
/de.company.department.hello/src/main/resources
/de.company.department.hello/src/test/java
/de.company.department.hello/src/test/resources
/de.company.department.hello/src/target
```

Build-Management: Maven

TECHNISCHE UNIVERSITÄT DARMSTADT



maven

Automatic dependency management

Maven Artifacts

TECHNISCHE UNIVERSITÄT DARMSTADT

Built Maven Project -> Artifacts

Types of artifacts:

- POM (project description in Maven XML)
- JAR (compiled classes)
- test-JAR (compiled test classes)
- javadoc
- sources (of JAR)
- test sources (sources of test-JAR)

Maven Artifact Repositories

TECHNISCHE UNIVERSITÄT DARMSTADT

Deployment of artifacts in repository:

- Apache Maven central artifact repository („Maven Central“)
- Artifact repositories at UKP
  - UKP uses **artifactory** as repository management tool
  - public artifactory on zoidberg
  - private artifactory

artifactory

- Local artifact repository: ./m2/repository

Jenkins

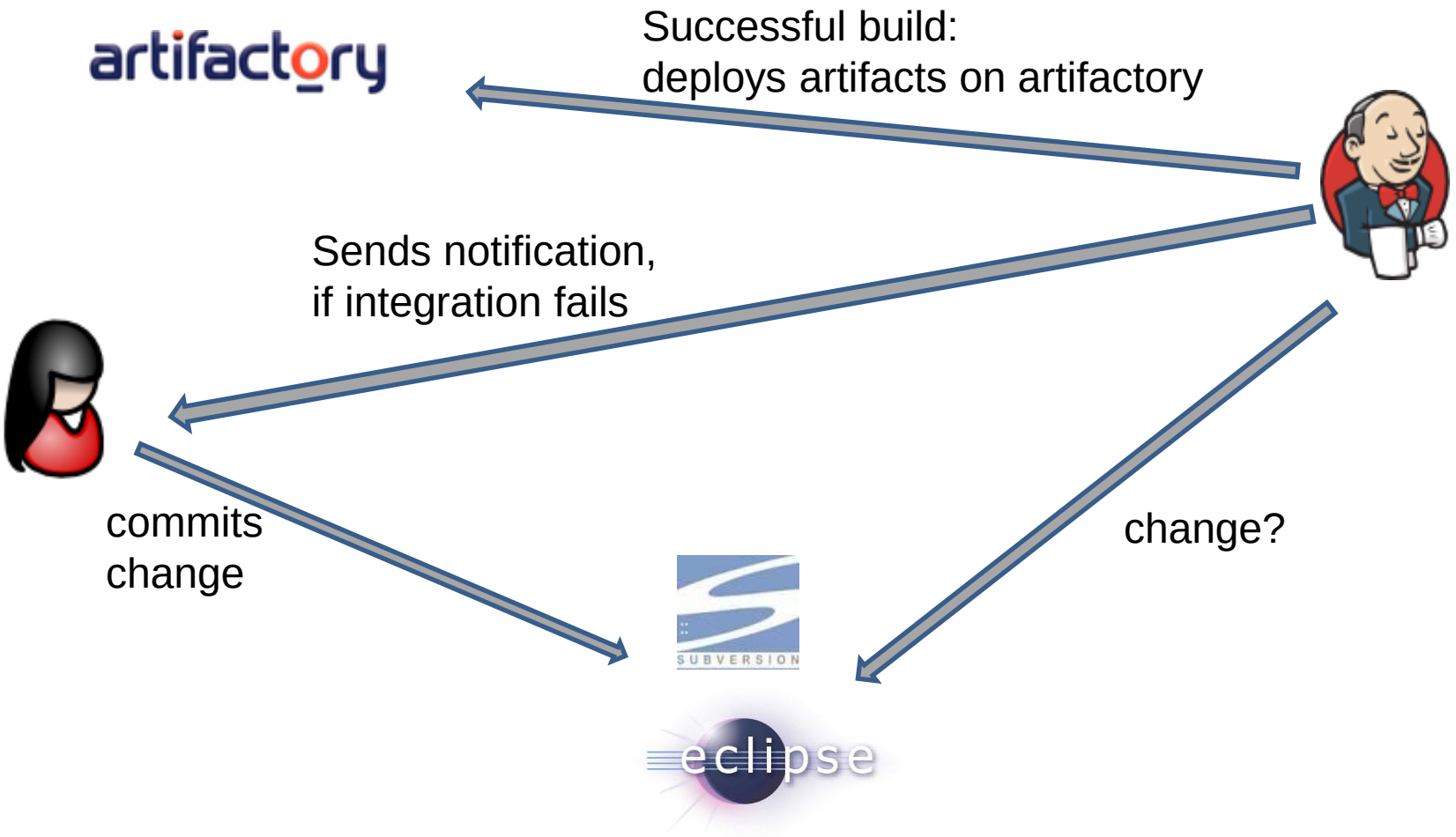
Continuous Integration Server (http://jenkins-ci.org/)

Automated

- Builds
- Tests
- Software Metrics
- Notifications

Configuration of Jenkins: Maven, JUnit, artifactory …

Simple Example



UKP Artifact repository
Developer publishes component

TECHNISCHE UNIVERSITÄT DARMSTADT

Overview

Artifact

Group Id: de.tudarmstadt.ukp.dkpro.core

Artifact Id:* de.tudarmstadt.ukp.dkpro.core.treetagger

Version: 1.0.3

Packaging:

Parent

Group Id:* de.tudarmstadt.ukp.dkpro.core

Artifact Id:* de.tudarmstadt.ukp.dkpro.core

Version:* 1.0.3

Relative Path:

Properties

Project

Name: DKPro AE TreeTagger Wrapper

URL:

Description:

Inception:

Organization

SCM

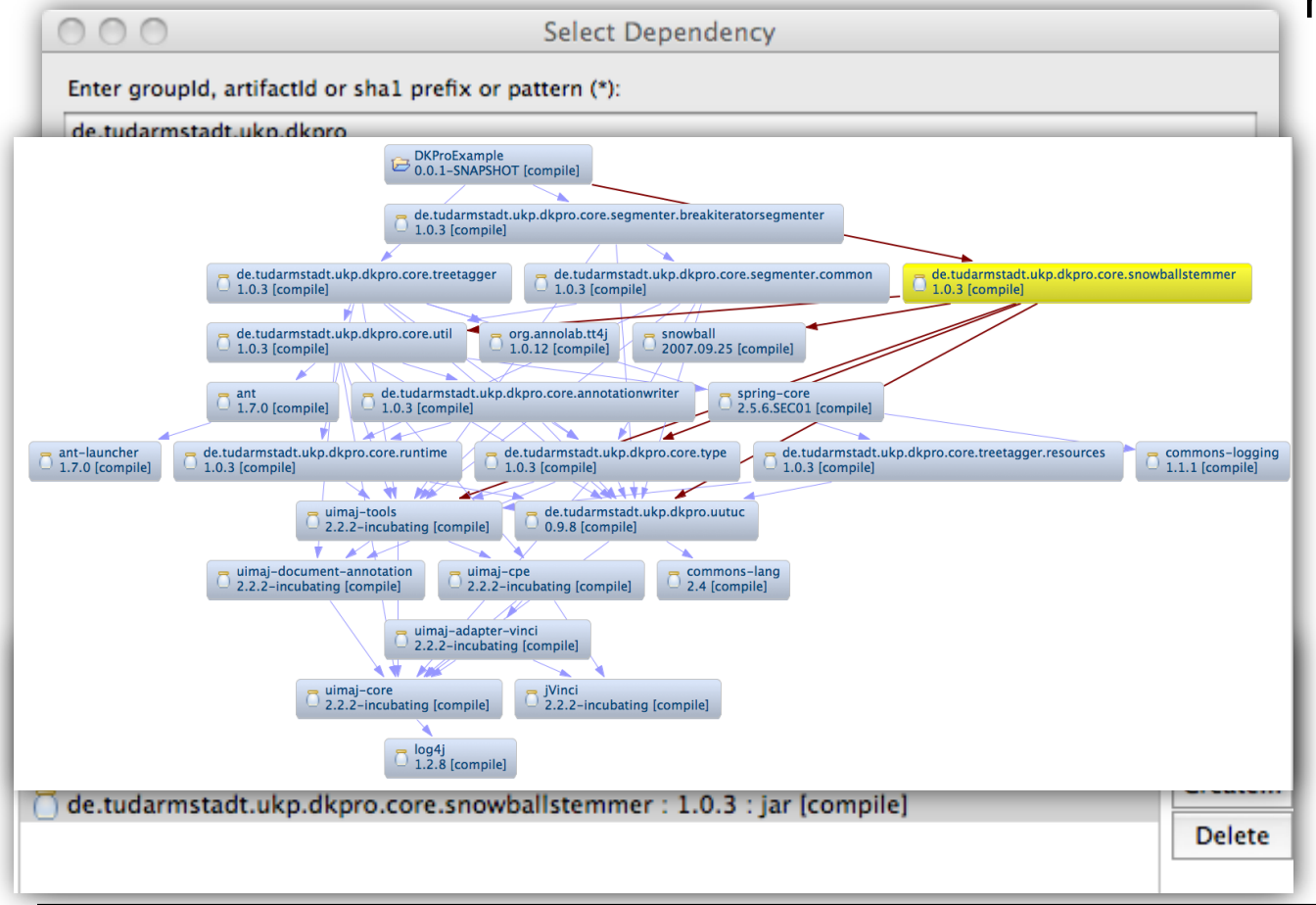
Issue Management

Continuous Integration

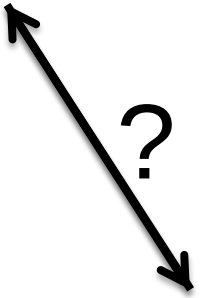
- Current development snapshot or
- Stable release version

UKP Artifact repository
Retrieving components

TECHNISCHE UNIVERSITÄT DARMSTADT



Artifact repository



UKP Artifact Repository

TECHNISCHE UNIVERSITÄT DARMSTADT

Source Version Control System

Continuous Integration Server (Automatic Building & Testing)

Artifact repository

- Current development snapshots
- Stable release versions
- Searchable via web interface
- Seamless integration with development environment

DKPro – Why participate?

TECHNISCHE UNIVERSITÄT DARMSTADT

Benefits:

Understanding the development infrastructure
and the software components available

agile implementation of research experiments