Enterprise Architect

Workshop for KUKA

Scratchbook

Javacream

21.3.2019

Resources

- Download additional resources from
https://GitHub.com/Javacream/training/enterprisearchitect
  - branch kuka_21.3.2019

Agenda

- Add new components, packages, classes
- Reverse engineering (Java, C++)
- Create a sequence diagram
- Exporting views
- Restore deleted elements
- Opening multiple models
- Version control
- Model Review
- Comparing Models
- Merging/Cherry Picking
- Creating templates
  - View
  - color schmes
- Migrating EA13 to EA14

Copyright and Impressum

© Javacream, 2019

Javacream

Dr. Rainer Sawitzki

Alois-Gilg-Weg 6
81373 München

Table of Contents

inspire
qualify
change

1

BASIC USE CASES

1.1

Add new components, packages, classes

Workflow

- **start with a structured project layout**
  - wizards are available, but create very verbose elements
  - by hand seems to be the better alternative

- **Create elements using**
  - ribbon
  - context menu
  - keyboard shortcuts

1.2

Reverse engineering (Java, C++)
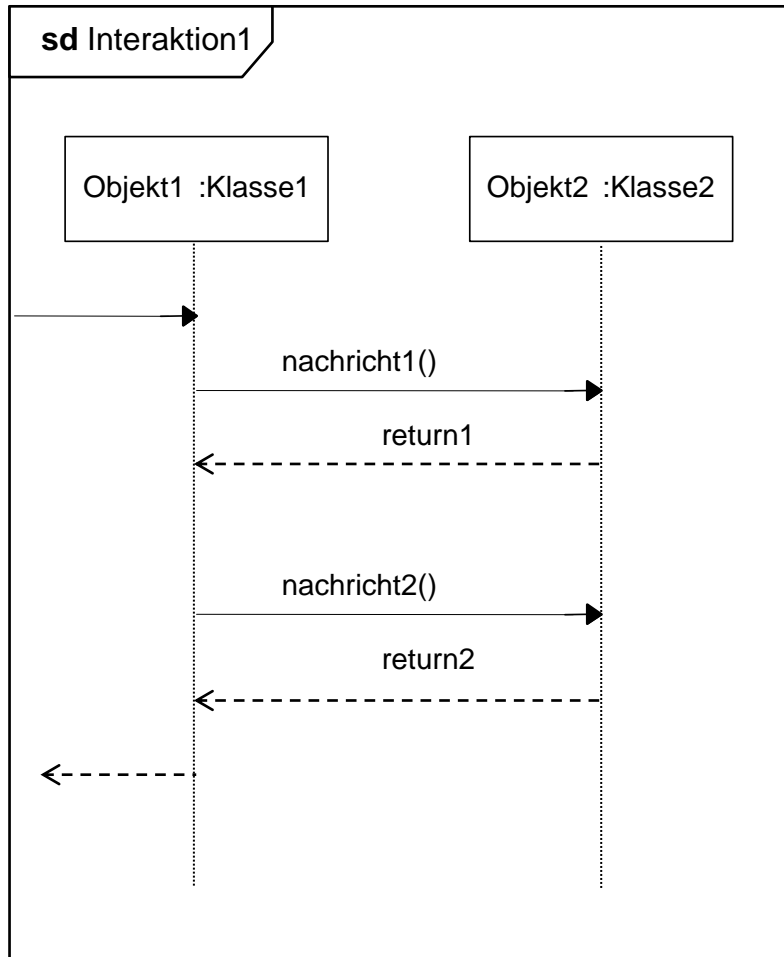
From source to model: theory

- EA has wizards for reverse engineering exiting sources into a model
  - supports different languages
    - Java
    - C++
    - …
- EA generates class and sequence models
  - especially Java source code is structured quite well
- generated model is linked to the source files

From source to model: praxis

- **reverse engineering is very limited**
  - sources tend to represent the developers thoughts, not a strict model
  - even a readable source code does not necessary create a readable model

- **changing the generated model and re-generate souces is risky**
  - not recommended

- **EA generates unconnected views for every package**
  - the "big picture" is missing
  - overview must be generated manually

- **eventually reverse engineering is just a "one shot"**
  - overview
  - documentation

- **there exists lots lightweight tools that do the same**
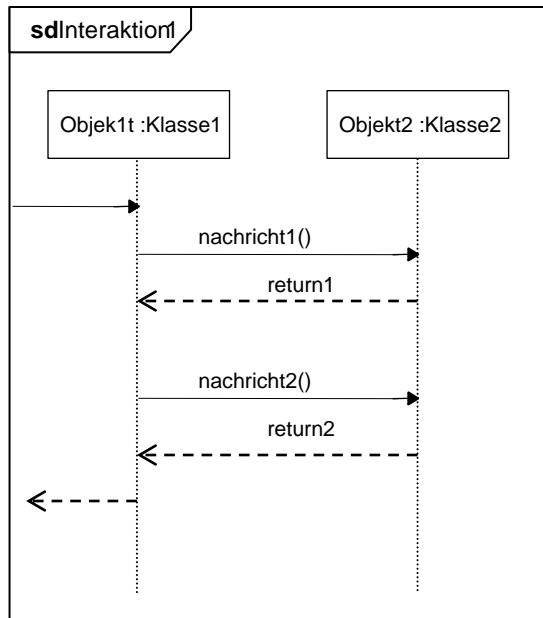 - EA might be overkill

Hands on

- the provided resources contain the Java project "BooksWarehouse"
- create class diagrams that show the dependencies, hierarchie

1.3

Create a sequence diagram

sequence diagram

sd Interaktion1

Objekt1 :Klasse1

Objekt2 :Klasse2

nachricht1()

return1

nachricht2()

return2

- Sequenzdiagramm
  - zeigt eine Reihe von Nachrichten, die
  - ... eine ausgewählte Menge von Kommunikationspartnern (Objekte, Akteure, Systeme, ...) in
  - ... einer zeitlich begrenzen Situation austauschen
- grundlegende Struktur
  - Kommunikationspartner mit einer vertikalen Lebenslinie
  - Nachrichten als Pfeile zwischen Lebenslinien
  - Zeitablauf von oben nach unten
  - Rahmen mit Namen des Diagramms

Kommunikationspartner

- Beispiele für Kommunikationspartner und ihre Kommunikation:
  - Akteure, die untereinander natürlichsprachliche Nachrichten und Dokumente austauschen, z.B. im Rahmen der Geschäftsprozessmodellierung
  - Akteure, die mit dem System kommunizieren, z.B. um den Ablauf eines Anwendungsfalls zu illustrieren
  - Instanzen von Klassen des Systems (Objekte), die untereinander programmiersprachliche Nachrichten austauschen
  - physikalische Geräte, die Ereignisse senden und empfangen

1.4

Exporting

Images

- Supported formats
    - Windows bitmap
    - GIF image
    - Windows Enhanced Metafile (standard metafile)
    - Windows Placeable Metafile (older style metafile)
    - PNG format
    - JPG
    - TGA

Reports

- https://sparxsystems.com/enterprise-architect/document-generation-reporting/document-generation-reporting.html

1.5

Restore deleted elements

"Undo delete": not yet implemented…

- Deleting views etc. from the model cannot be undone
  - backing data model in EA is quite complex, so deleting a view is much more than deleting a file

- it is your responsibility to manage backups
  - see "version control"

1.6

Opening multiple models

Not yet implemented

- Workaround
  - Open multiple EA instances

inspire
qualify
change

2

Team Collaboration

2.1

Version control

Features

- Baselines
  - a comparable zipped file of your model or a package
  - baselines are stored within your model

- Integrating source code management systems (SCM)
  - cvs
  - Subversion = svn
  - Microsoft Team Foundation Server
  - the very popular GIT is not supported
  - versions are stored inside the SCM

Limitations

- Never (!) use native SCM Clients to checkout different versions
  - use EA functions or you may get a corrupted model

- Showing diffs is OK, but merging is not supported out of the box
  - for possible workflows and workarounds the chapter "Merging/Cherry Picking"

2.2

Model Review

Great tutorial

- https://prolaborate.sparxsystems.com/resources/documentation/review-enterprise-architect-models

2.3

Comparing Models

Workflows

- Compare models using
  - Baselines
  - Version Control
  - export the model to compare as XMI

inspire
qualify
change

2.4

Merging/Cherry Picking

Model Exchange

- Export and import elements from one project to another using XMI
  - Caution
    - Neither merge nor cherry pick!
    - Elements are copied, nothing else

- Export and import elements from one project to another using CSV

- Data Transfer
  - Basically transfer data from one data storage to another
    - File to File
    - DBMS to File
    - File to DBMS
    - DBMS to DBMS
  - mainly used for Backup

3

Advanced

3.1

Creating templates

Custom Stereotypes and Shape Scripts

- https://www.sparxsystems.com.au/enterprise_architect_user_guide/14.0/modeling_tools/shape_editor_2.html

3.2

Migrating EA13 to EA14

What's new?

- Perspectives help you focus on your role

- Inline diagram reader

- Model Wizards are much more verbose