Christian-Albrechts-Universität zu Kiel
Technische Fakultät

A Hierarchical Eclipse-based Editor for System Dependency Graphs

Dean Finkes

16. November 2016

Gliederung

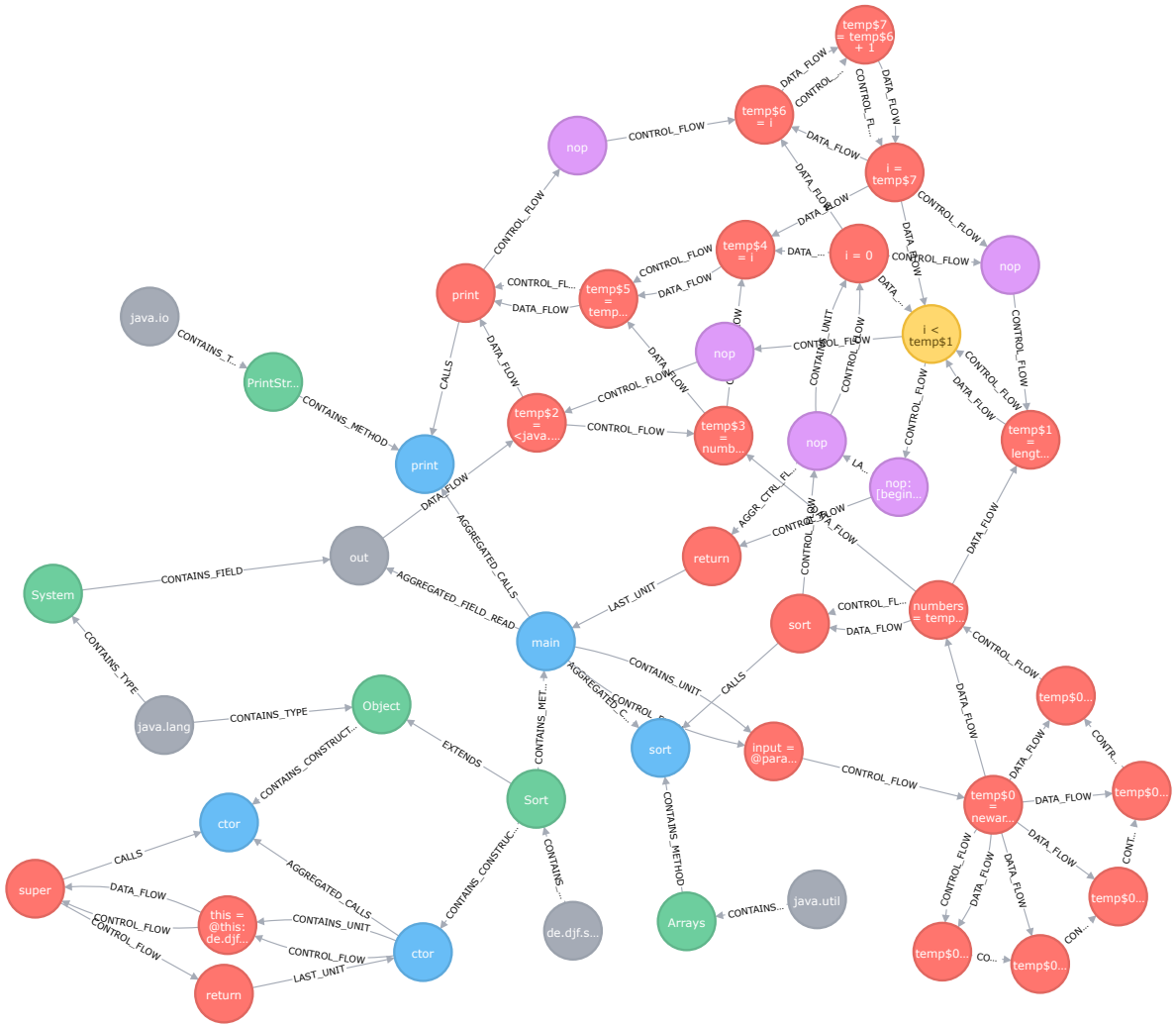


Abbildung: Systemabhängigkeitsgraph (SDG) in Neo4j

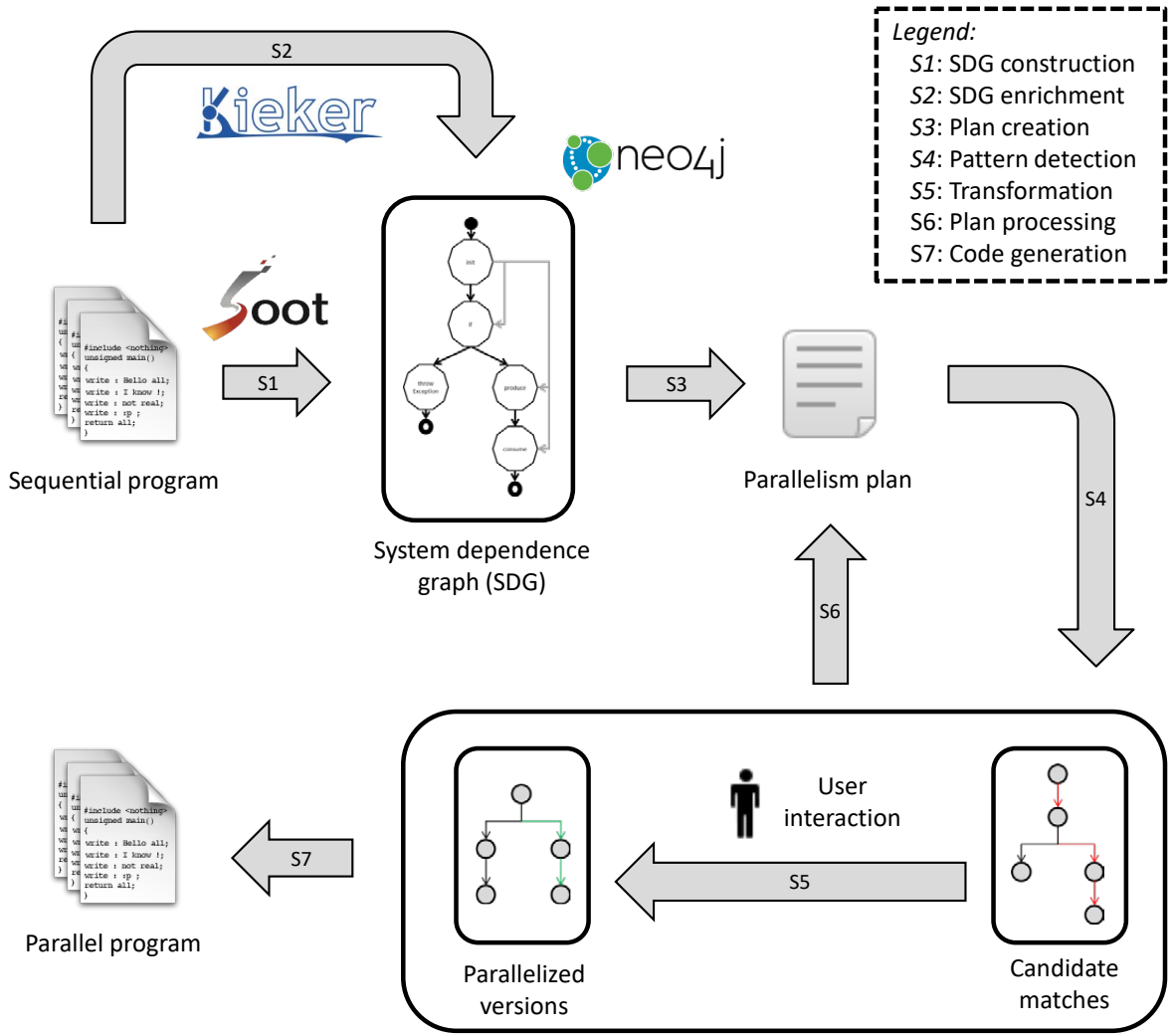


Abbildung: Semiautomatischer Parallelisierungsansatz von C. Wulf [3]

Gliederung

Christian-Albrechts-Universität zu Kiel
Technische Fakultät

- Beliebiger Wechsel zwischen flacher und hierarchischer Darstellung eines Graphen
- Ein- und Ausklappen von Elternknoten
- Berücksichtigung der Kanten bezüglich eingekapselter Knoten

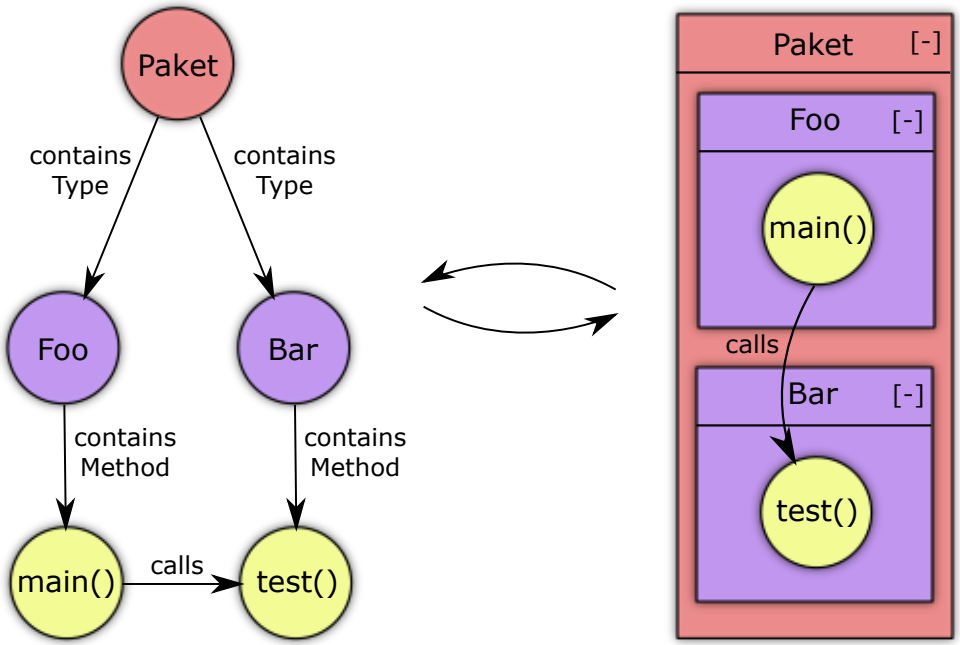


Abbildung: Umschalten zwischen der flachen (links) und hierarchischen (rechts) Darstellung

C A U
Christian-Albrechts-Universität zu Kiel
Technische Fakultät

- Import und Export beliebiger Graphen über das lokale Dateisystem und URLs
- Auswahl der als Hierarchie darzustellenden Kanten beim Import
- Optionen zum Zuweisen der Neo4j-spezifischen Knotenlabels und Kantentypen beim Export

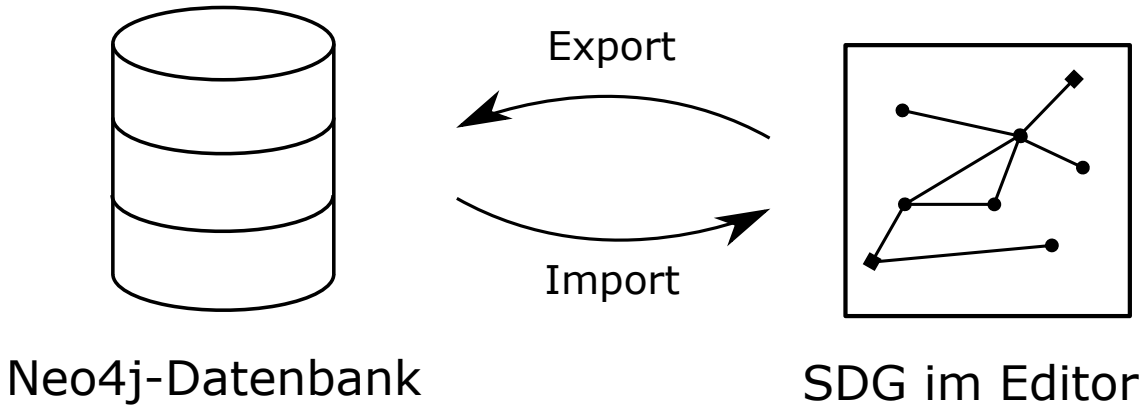


Abbildung: Import und Export eines Graphen in bzw. aus dem Grapheditor

Christian-Albrechts-Universität zu Kiel
Technische Fakultät

- Z3: Anpassungen an das Kontextmenü
  - Menüeinträge zum Einfügen von Kindknoten
  - Dynamische Anpassung an die Menge der fokussierten Objekte
- Z4: Erweiterungen bezüglich der Bedienbarkeit des Editors
  - Z4.1: Löschen von hierarchischen Knoten
  - Z4.2: Erweiterung der Optionen zum Einfärben eines Graphen
  - Z4.3: Periodisches Speichern der Graphen
- Z5: Evaluation des Grapheditors
  - Evaluation der Importfunktionalität
  - Evaluation der Exportfunktionalität
  - Evaluation der hierarchischen Konzepte

- Die Graphen:
  - Der Eigenschaftsgraph
  - Der Java-spezifische Systemabhängigkeitsgraph (SDG)

- Die Entwicklungsumgebung *Eclipse*

1

- Das *Eclipse Modeling Framework* (EMF)

2

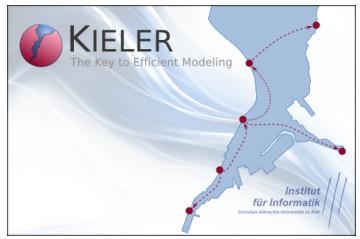
[1] https://eclipse.org/downloads/packages/sites/all/themes/solstice/_themes/solstice_packages/ logo.png

[2] https://www.eclipse.org/modeling/emf/images/emf_logo.png

- Die Graphdatenbank *Neo4j*

1

- Das *Visualisierungs*-Framework *KLighD*



2

- Die Programmiersprache *Xtend*

3

[1] http://neo4j.com/wp-content/themes/neo4jweb/assets/images/neo4j-logo-2015.png

[2] http://rtsys.informatik.uni-kiel.de/confluence/download/attachments/9471056/splash.png?version=1&modificationDate=1403687464000&api=v2

[3] http://zeroturnaround.com/wp-content/uploads/2013/01/xtend-logo.png

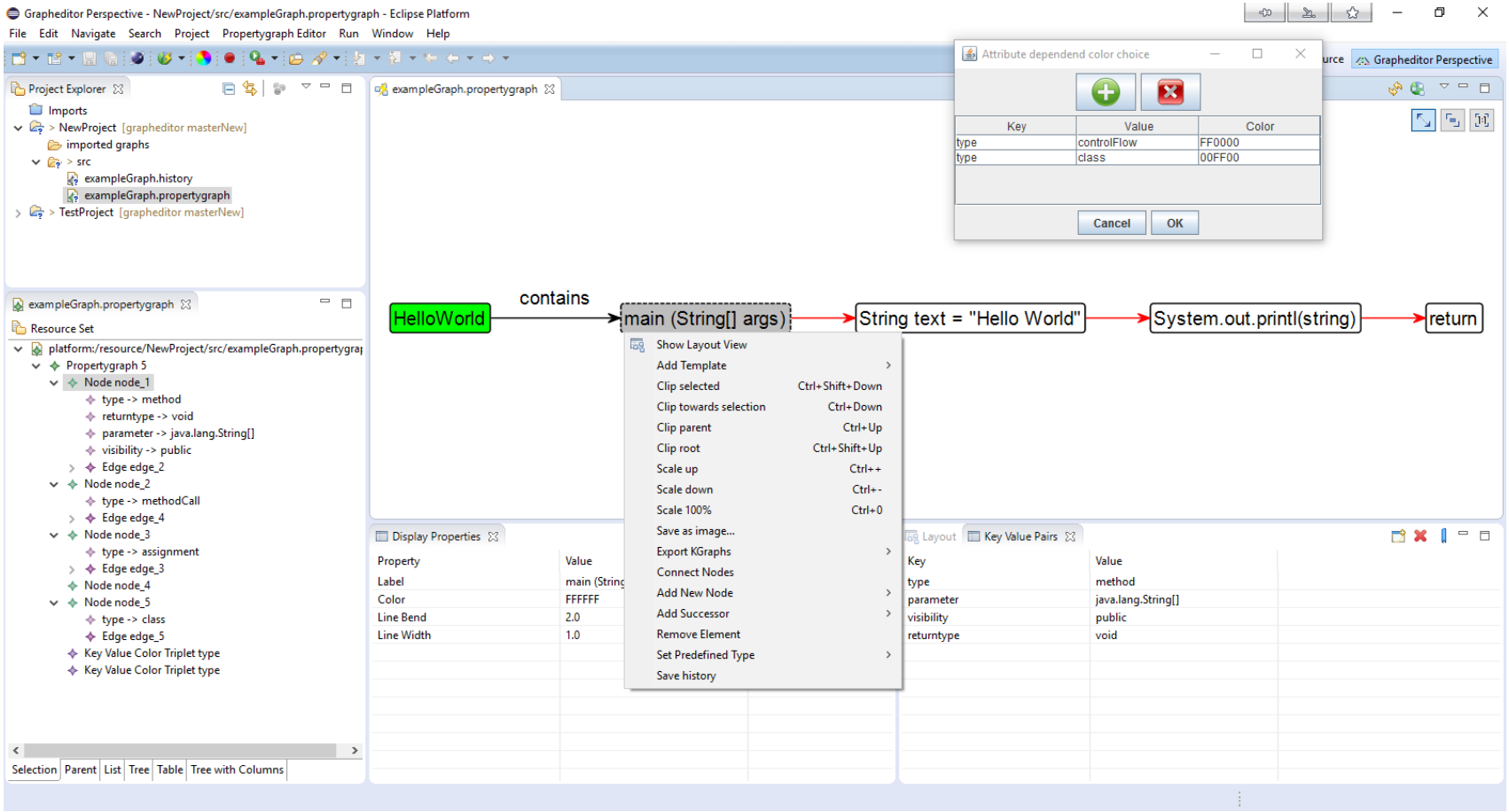


Abbildung: Screenshot des bisherigen Grapheditors [1] [2]

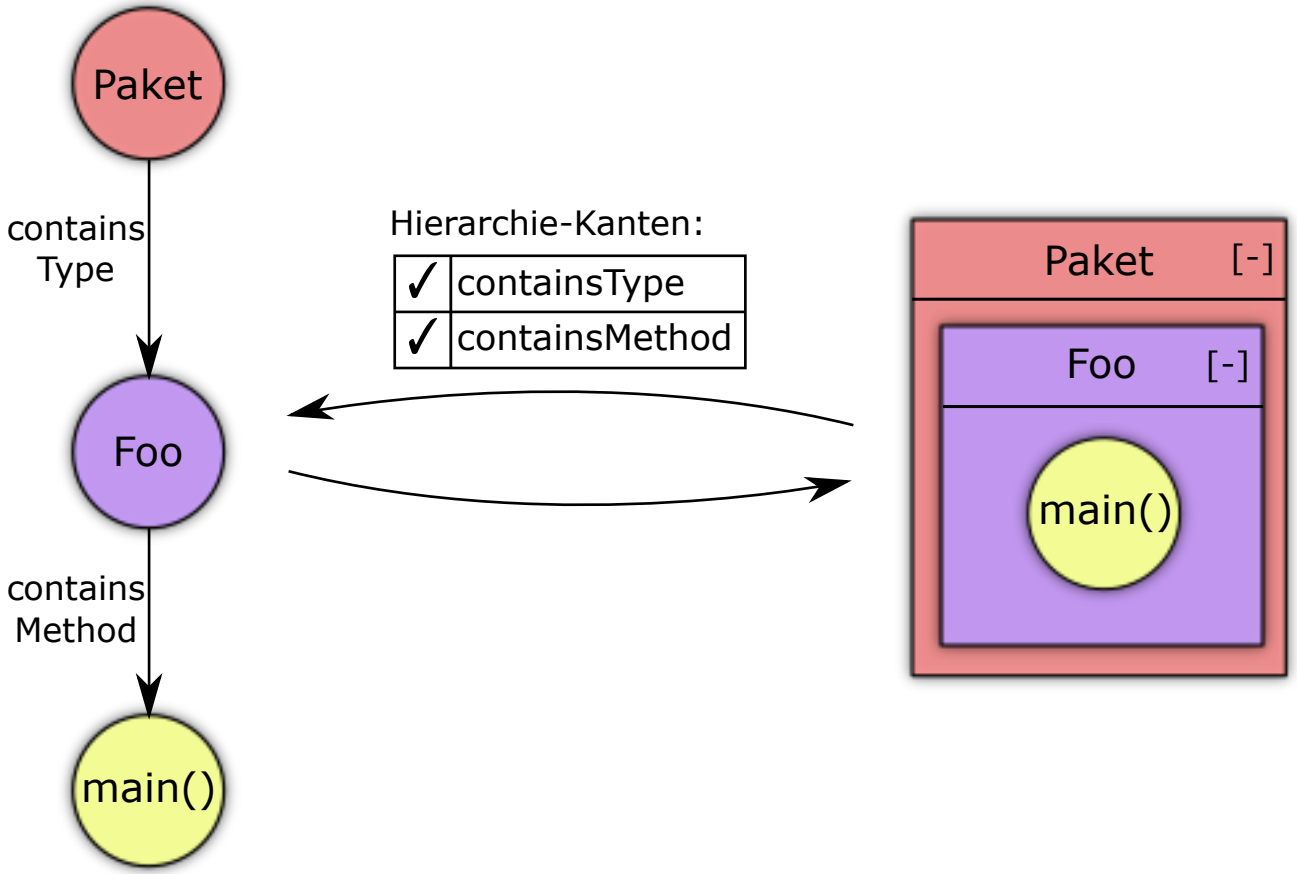


Abbildung: Transformation zwischen der flachen (links) und hierarchischen (rechts) Ansicht eines Graphen mittels *Hierarchie-Kanten*

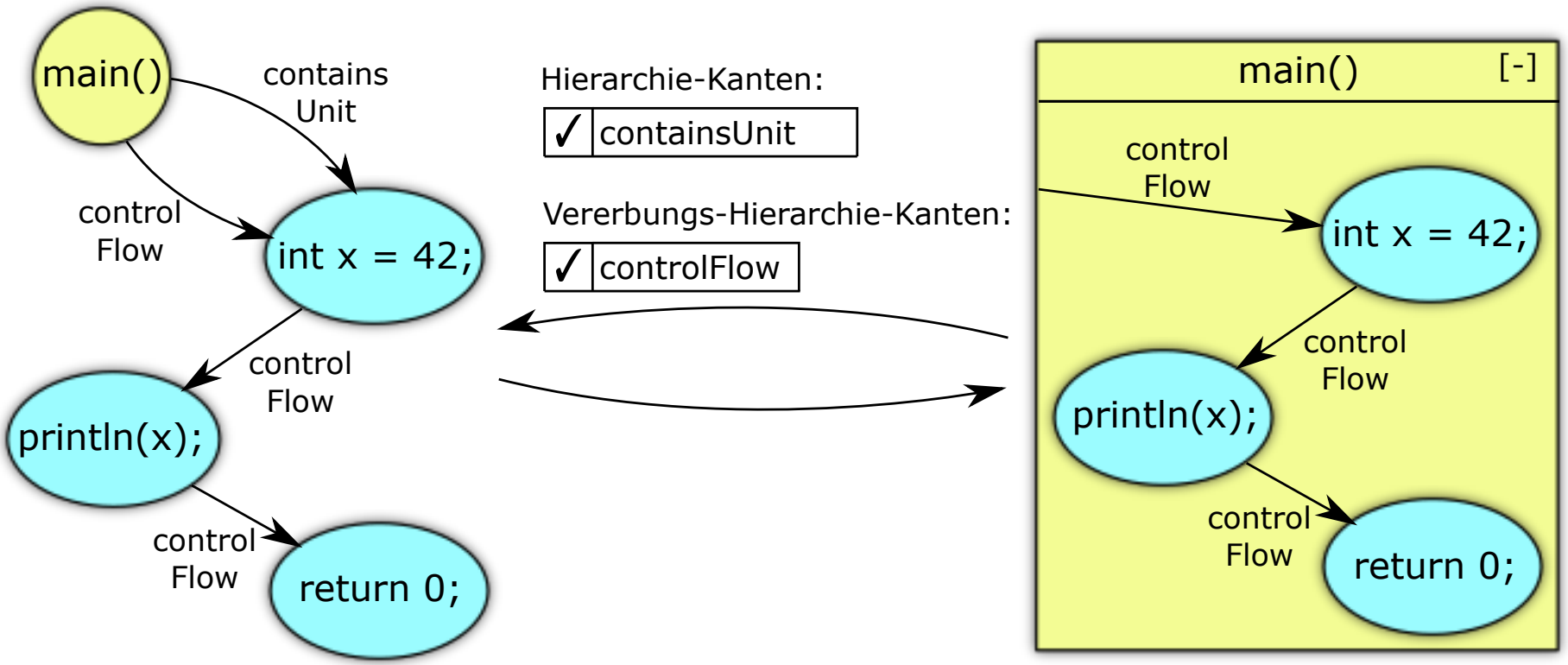


Abbildung: Verwendung der *Vererbungs-Hierarchie-Kanten*

Christian-Albrechts-Universität zu Kiel
Technische Fakultät

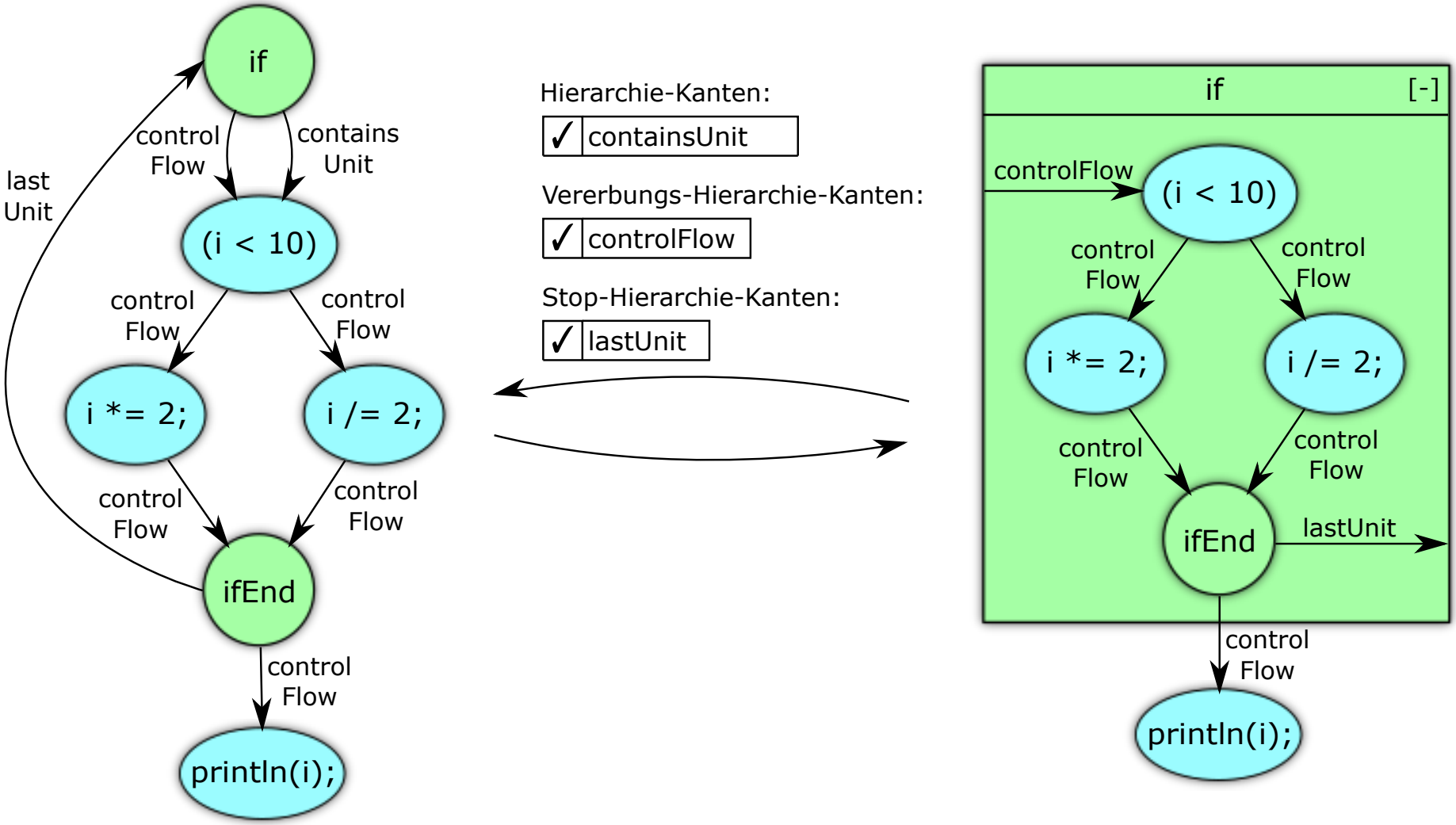
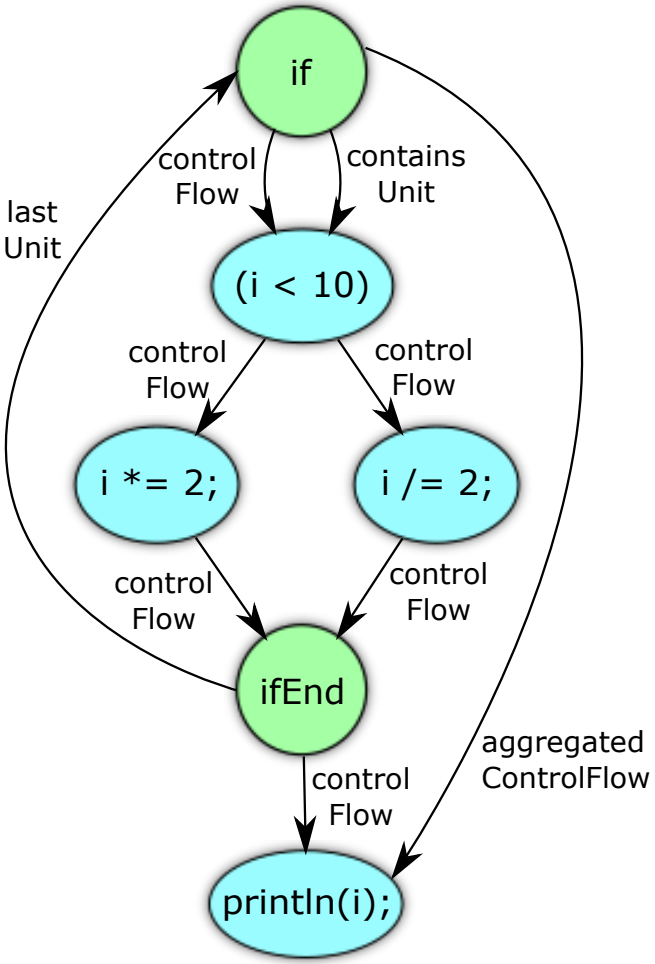


Abbildung: Verwendung der *Stop-Hierarchie-Kanten*

Christian-Albrechts-Universität zu Kiel
Technische Fakultät



Hierarchie-Kanten:

✓	containsUnit

Vererbungs-Hierarchie-Kanten:

✓	controlFlow
✓	aggregatedControlFlow

Stop-Hierarchie-Kanten:

✓	lastUnit

Aggregations-Tupel:

✓	aggregatedControlFlow
	controlFlow

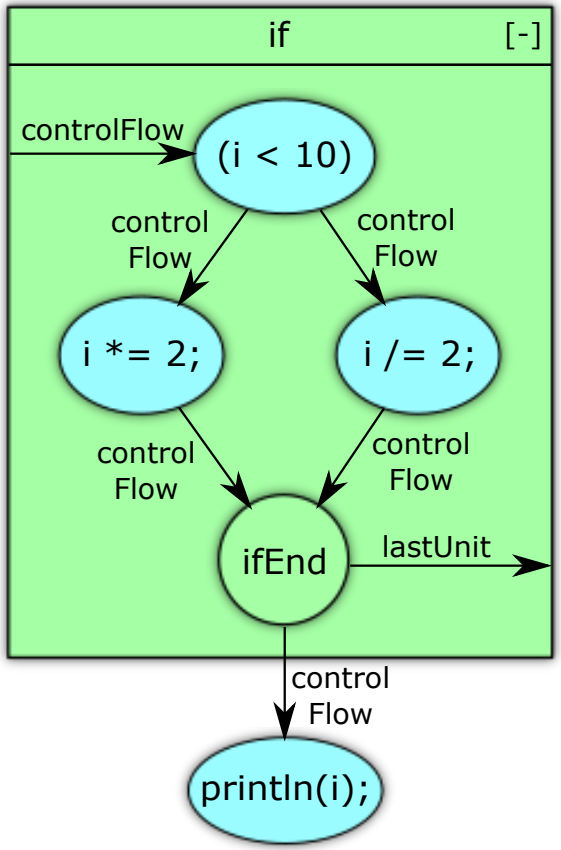


Abbildung: Verwendung der *Aggregations-Tupel*

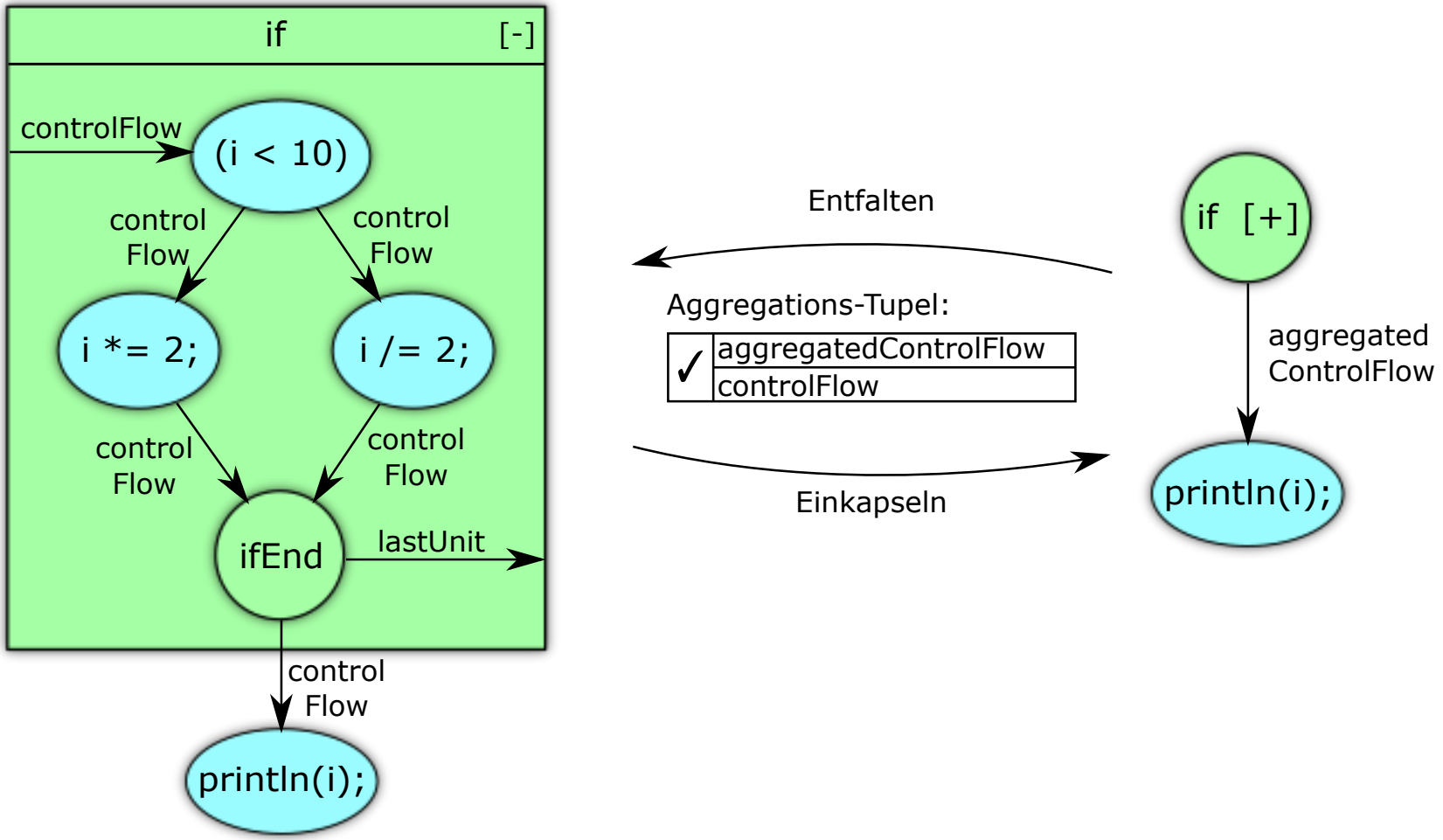


Abbildung: Dynamische Anpassung der Kanten mittels *Aggregations-Tupeln*

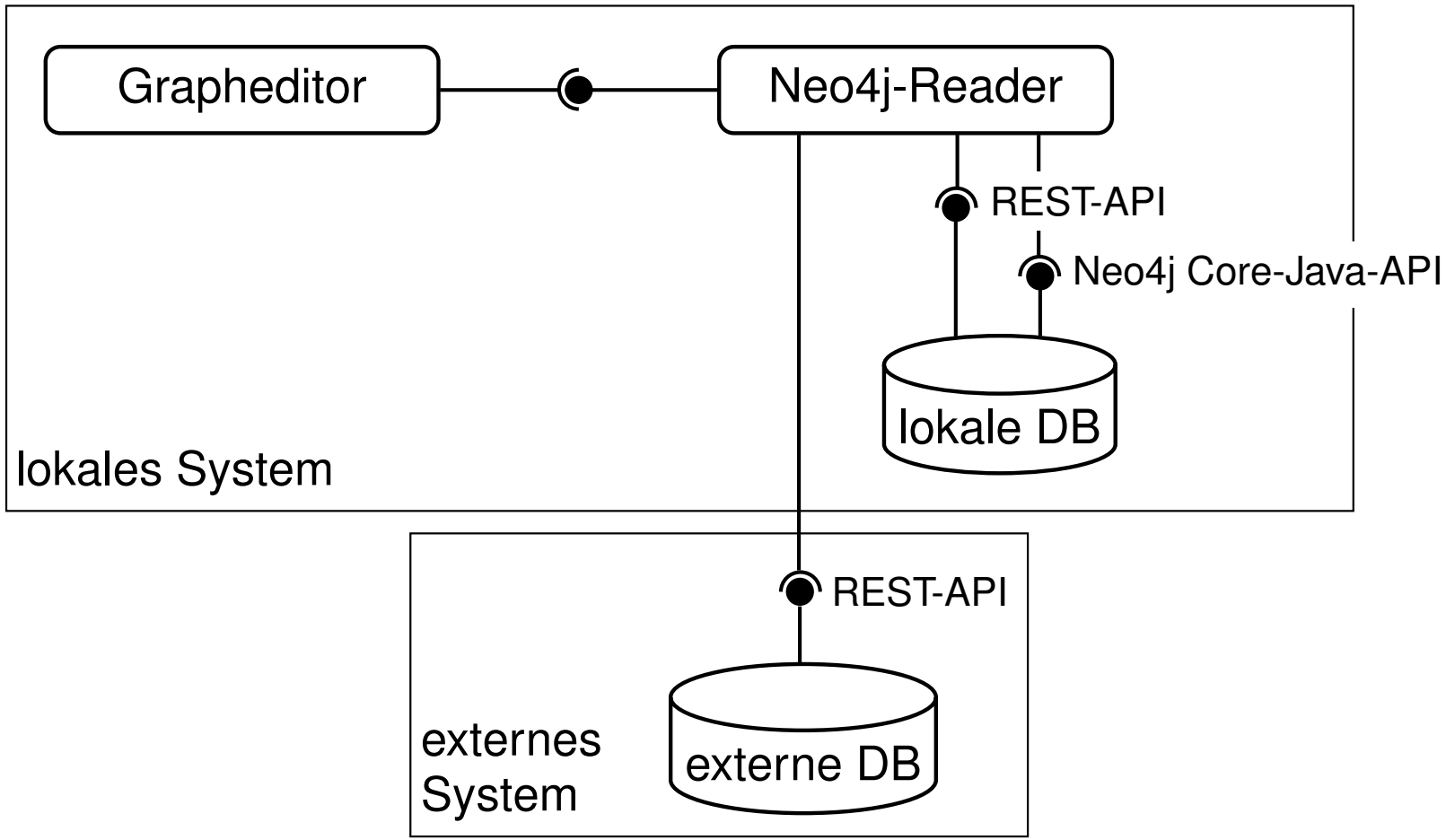


Abbildung: Zusammenhang der einzelnen Komponenten beim Im- und Exportieren eines Graphen

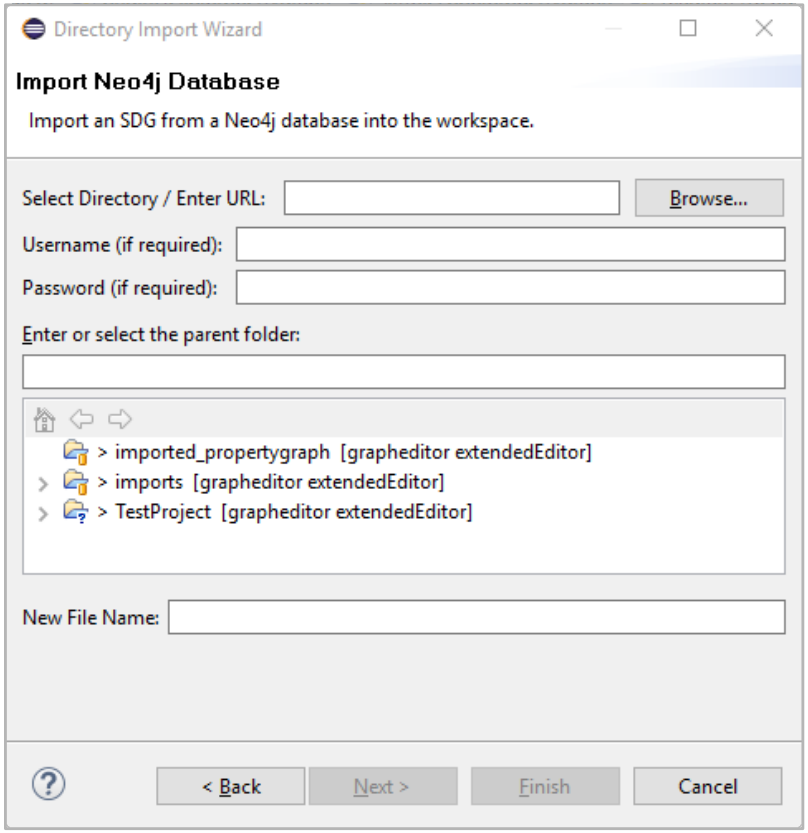


Abbildung: Dialog zur Auswahl der zu importierenden Datenbank

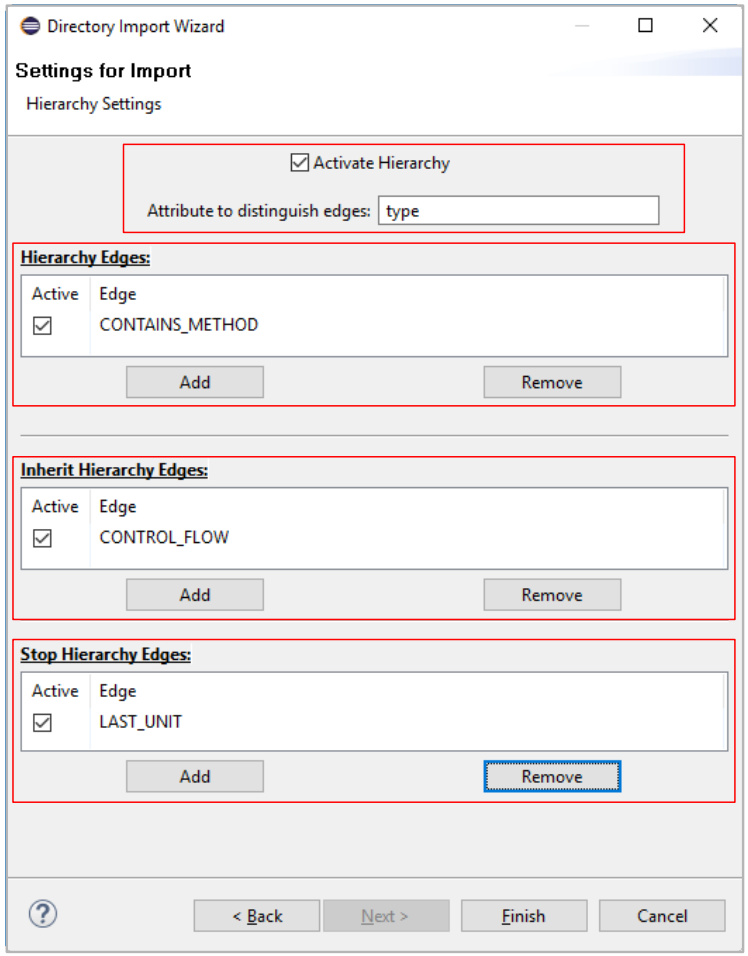


Abbildung: Dialog zur Typisierung der vorhandenen Kanten einer Neo4j-Datenbank

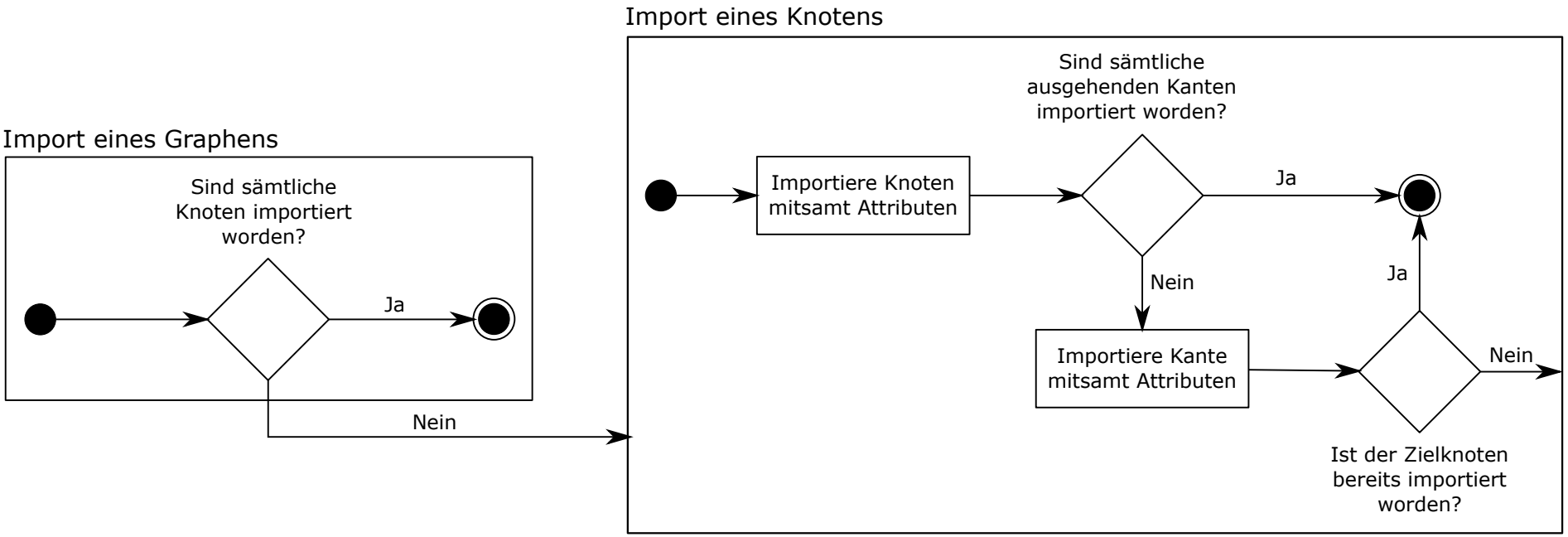


Abbildung: Import eines Graphen über die Neo4j-Core-Java-API

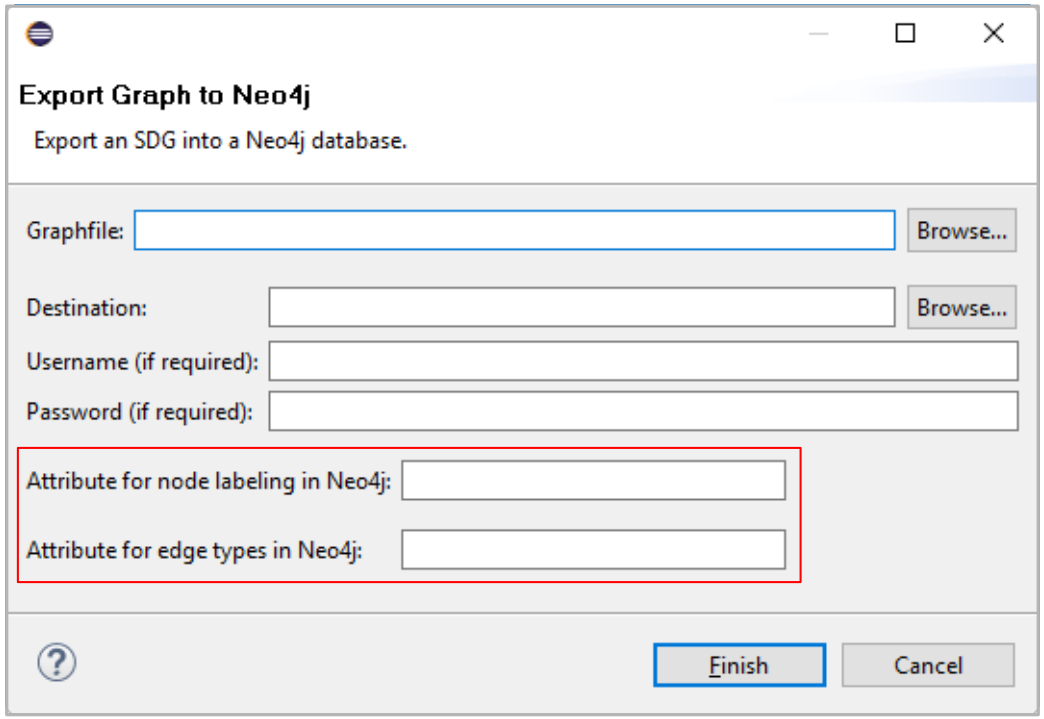


Abbildung: Dialog zum Exportieren eines beliebigen Graphen

Listing 1 Generisches Muster zum Exportieren eines Knotens

```
1  CREATE (n: <nodeType> {<properties>})
2  RETURN <node.ID> as oldID, id(n) as newID
```

Listing 2 Generisches Muster zum Exportieren einer Kante

```
1  MATCH (a),(b) WHERE id(a) = <sourceNodeID>
2                   AND id(b) = <targetNodeID>
3  CREATE (a)-[r: <edgeType> {<properties>} ]->(b)
```

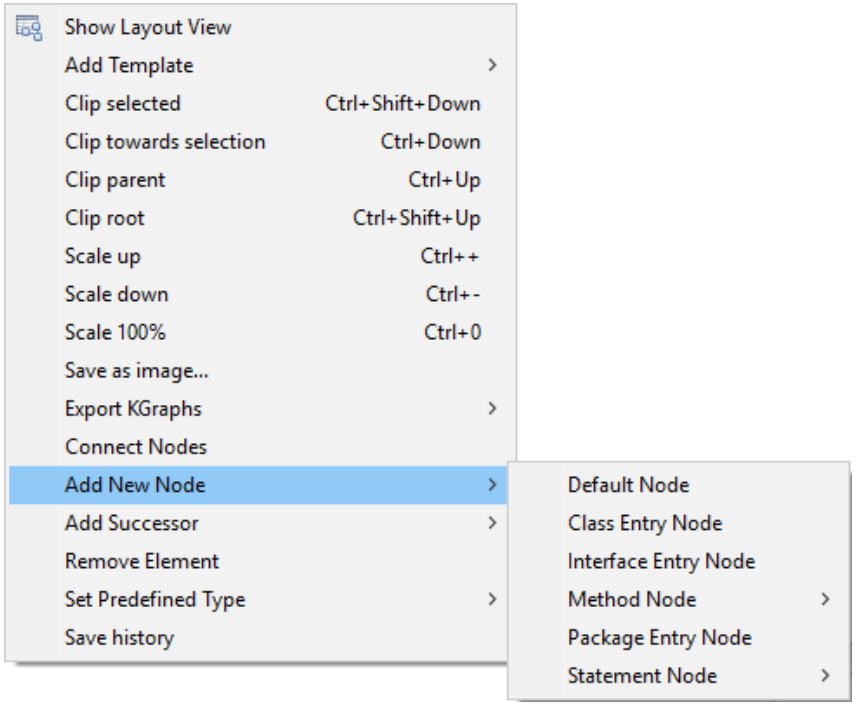


Abbildung: Bisheriges Menü

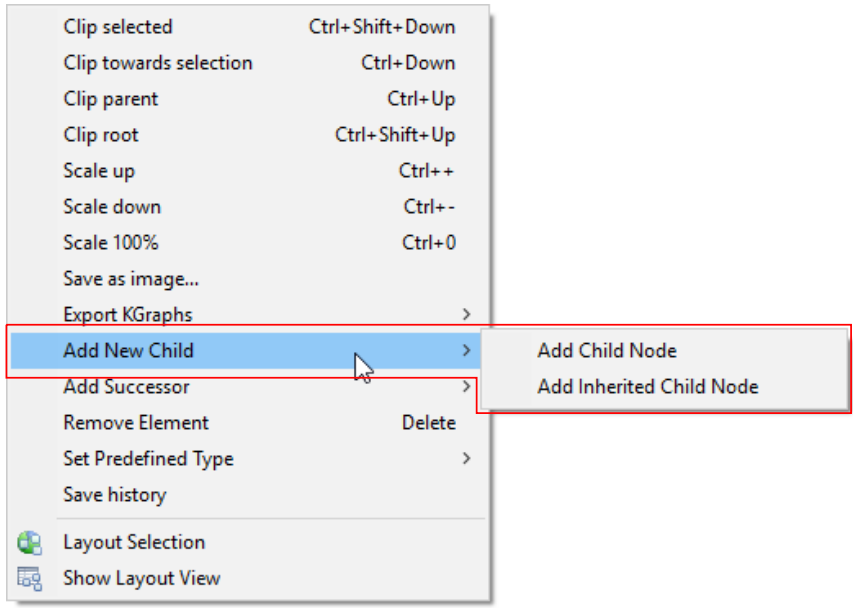


Abbildung: Dynamisches Menü

Christian-Albrechts-Universität zu Kiel
Technische Fakultät

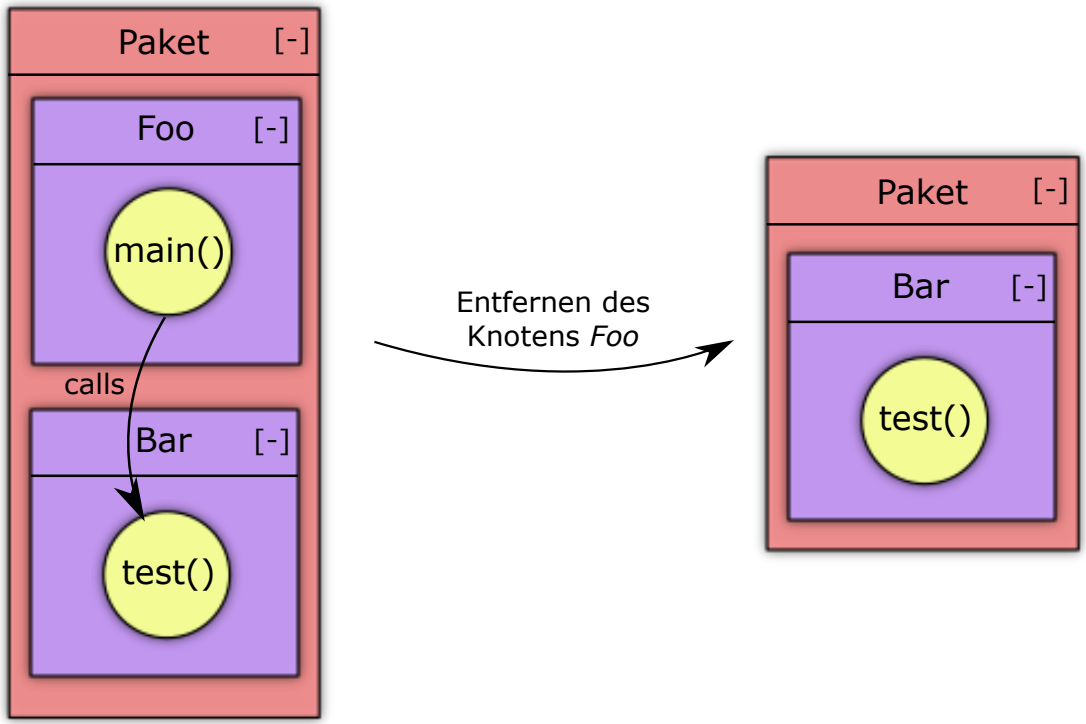


Abbildung: Löschen eines Elternknotens in der hierarchischen Ansicht

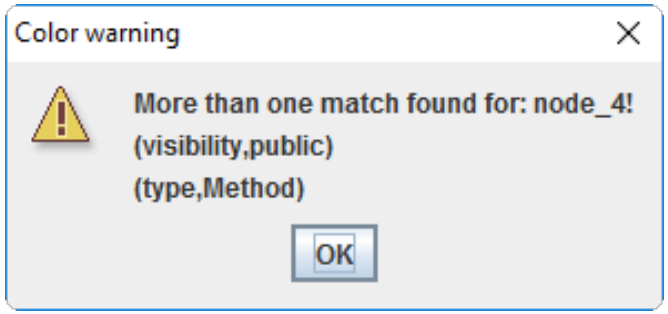


Abbildung: Farbkonflikt zwischen den Tupeln *(type,Method)* und *(visibility,public)* bei dem Knoten mit der ID *node_4*

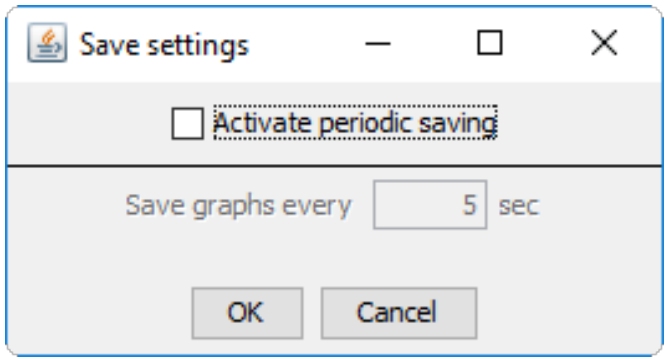


Abbildung: Der Dialog zum Aktivieren des periodischen Speicherns

C A U
Christian-Albrechts-Universität zu Kiel
Technische Fakultät

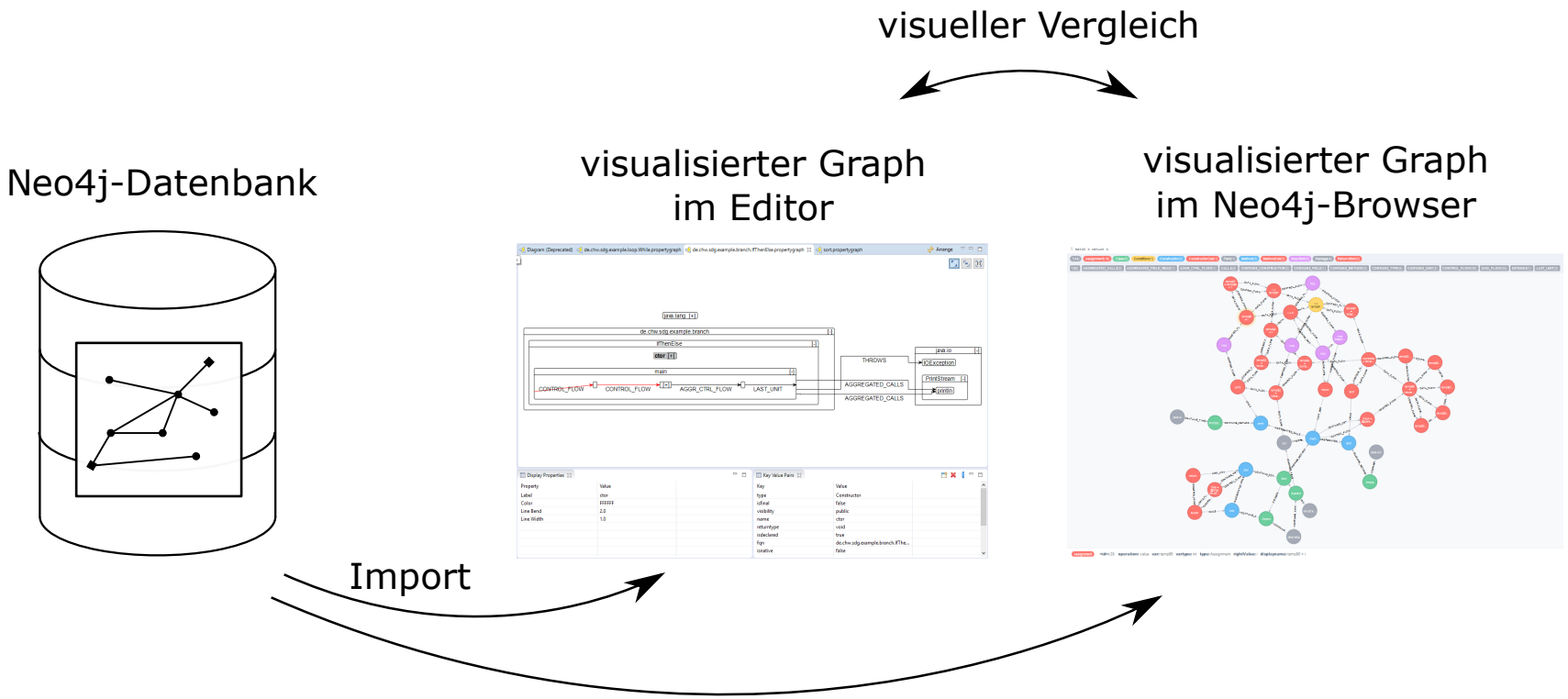


Abbildung: Evaluation der Importfunktionalität

	Smartphone			Medizin			Autohersteller			SDG		
	Orig.	Datei	URL	Orig.	Datei	URL	Orig.	Datei	URL	Orig.	Datei	URL
Wurden sämtliche Knoten importiert?												
Anzahl der Knoten	30	30	30	48	48	48	27	27	27	44	44	44
Wurden sämtliche Kanten importiert?												
Anzahl der Kanten	70	70	70	47	47	47	90	90	90	80	80	80
Korrekte Start- und Endknoten an den Kanten	(✓)	✓	✓	(✓)	✓	✓	(✓)	✓	✓	(✓)	✓	✓
Wurden die Attribute korrekt importiert?												
⋮		…			…			…			…	

Tabelle: Ermittelten Werte bei der Evaluation des *Neo4j-Readers*

C A U
Christian-Albrechts-Universität zu Kiel
Technische Fakultät

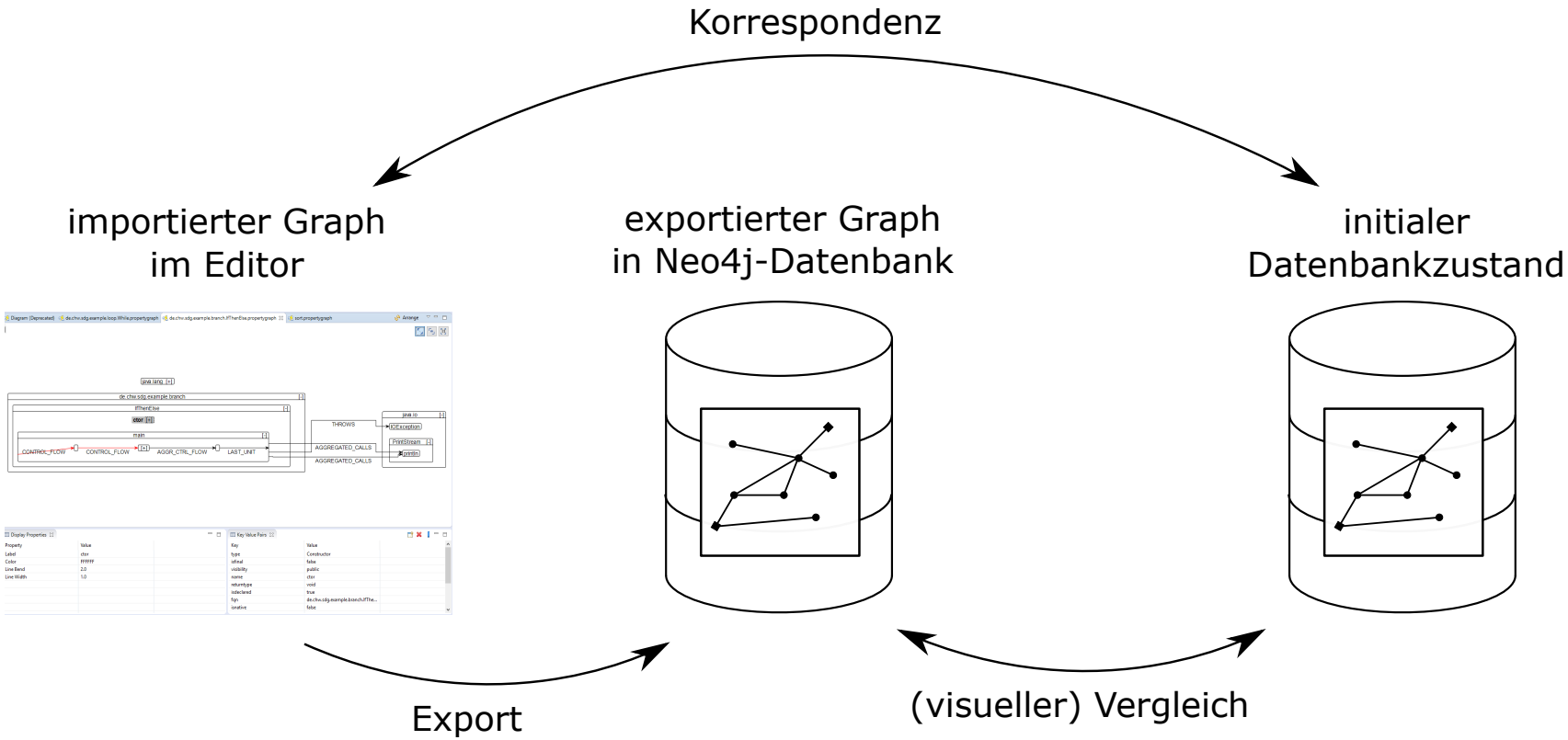


Abbildung: Evaluation der Exportfunktionalität

	Smartphone			Medizin			Autohersteller			SDG		
	Orig.	D2D	D2U	Orig.	D2D	D2U	Orig.	D2D	D2U	Orig.	D2D	D2U
		U2D	U2U		U2D	U2U		U2D	U2U		U2D	U2U
Wurden sämtliche Knoten exportiert?												
Anzahl der Knoten	30	30	30	30	30	30	30	30	30	30	30	30
		30	30		30	30		30	30		30	30
Wurden sämtliche Kanten exportiert?												
Anzahl der Kanten	70	70	70	47	47	47	90	90	90	80	80	80
		70	70		47	47		90	90		80	80
Korrekte Start- und Endknoten an den Kanten	(✓)	✓	✓	(✓)	✓	✓	(✓)	✓	✓	(✓)	✓	✓
		✓	✓		✓	✓		✓	✓		✓	✓
⋮		…			…			…			…	

Tabelle: Ermittelten Werte bei der Evaluation des *Neo4j-Writers*

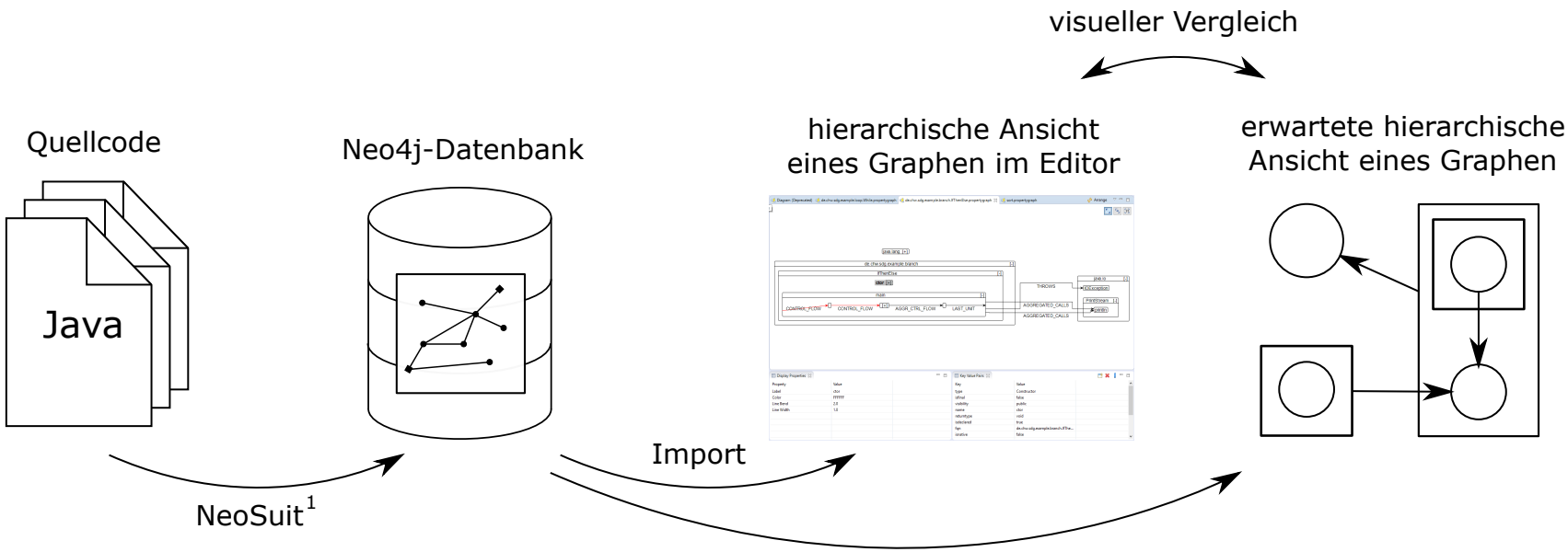


Abbildung: Evaluation der implementierten hierarchischen Konzepte

[1] https://build.se.informatik.uni-kiel.de/chw/integration-project

C A U
Christian-Albrechts-Universität zu Kiel
Technische Fakultät

	if-then-else		loops		nested		method-call	
	Erw.	Gem.	Erw.	Gem.	Erw.	Gem.	Erw.	Gem.
Korrekte Darstellung von Hierarchien?								
Anzahl der Elternknoten	13	13	12	12	15	15	39	39
Anzahl der Kindknoten	51	51	48	48	68	68	164	164
Korrekte Zuordnung zwischen Eltern- und Kindknoten	(✓)	✓	(✓)	✓	(✓)	✓	(✓)	✓
Einblendung von Kanten sichtbarer Kindknoten	(✓)	✓	(✓)	✓	(✓)	✓	(✓)	✓
Ausblendung von Kanten nicht sichtbarer Kindknoten	(✓)	✓	(✓)	✓	(✓)	✓	(✓)	✓
Korrektes Verhalten bei Kanten-Aggregationen?								
⋮	…		…		…		…	

Tabelle: Ermittelten Werte bei der Evaluation der hierarchischen Konzepte

- Hierarchischen Konzepte
  - Kantenorientierter Ansatz
  - Flexible Erzeugung verschiedenster Hierarchien
  - Kein intuitives Verständnis
- Neo4j-Reader und -Writer
  - Im- und Export unabhängig vom Modell des Graphen
  - Datenbankanbindung über das lokale Dateisystem oder eine URL
- Ausblick
  - Berücksichtigung der Neo4j-Knotenlabels und -Kantentypen
  - Automatische Erkennung von hierarchischen Kanten
  - Export von Metainformationen des *Propertygraphen*, wie etwa der Einfärbung


Yvan Benekov. „Entwicklung eines Eclipse-Plugins zur Aufzeichnung von Benutzerinteraktionen mit Eigenschaftsgraphen“. Bachelorarbeit. Institut für Informatik, Oktober 2015.


Lars Erik Blümke. „Konzeption und Implementierung eines Eclipse-Plugins zur Erstellung von graphbasierten Quellcodemustern“. Bachelorarbeit. Institut für Informatik, Oktober 2015.


Christian Wulf. „Pattern-based Detection and Utilization of Potential Parallelism in Software Systems“. In: *Proceeding of the Software Engineering 2014 Conference*. Feb. 2014.