

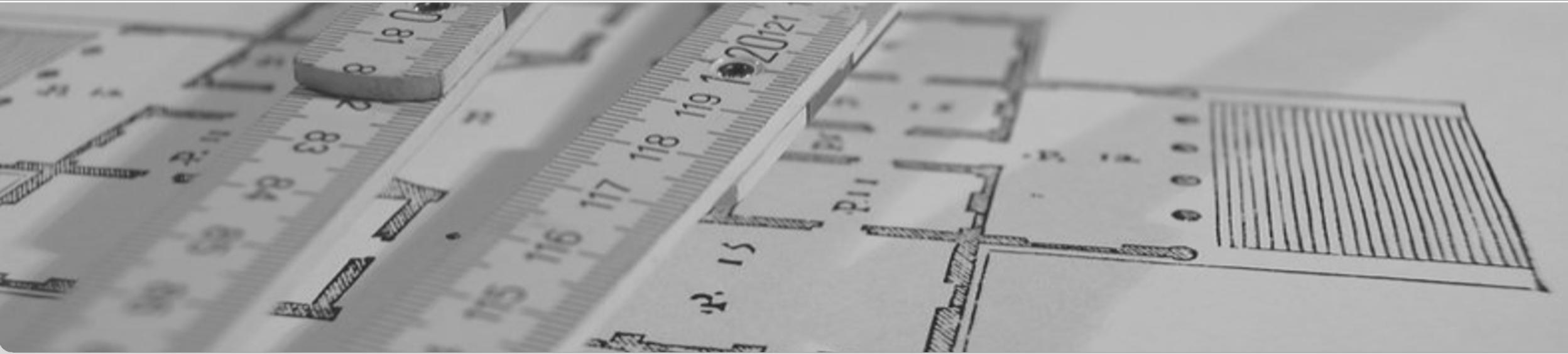
Sirius-Editor für Datenflussdiagramme

Praktikum Ingenieurmäßige Softwareentwicklung

Simon Schwarz

Betreuer: Stephan Seifermann

SOFTWARE-ENTWURF UND -QUALITÄT,
INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION, KIT-FAKULTÄT FÜR INFORMATIK



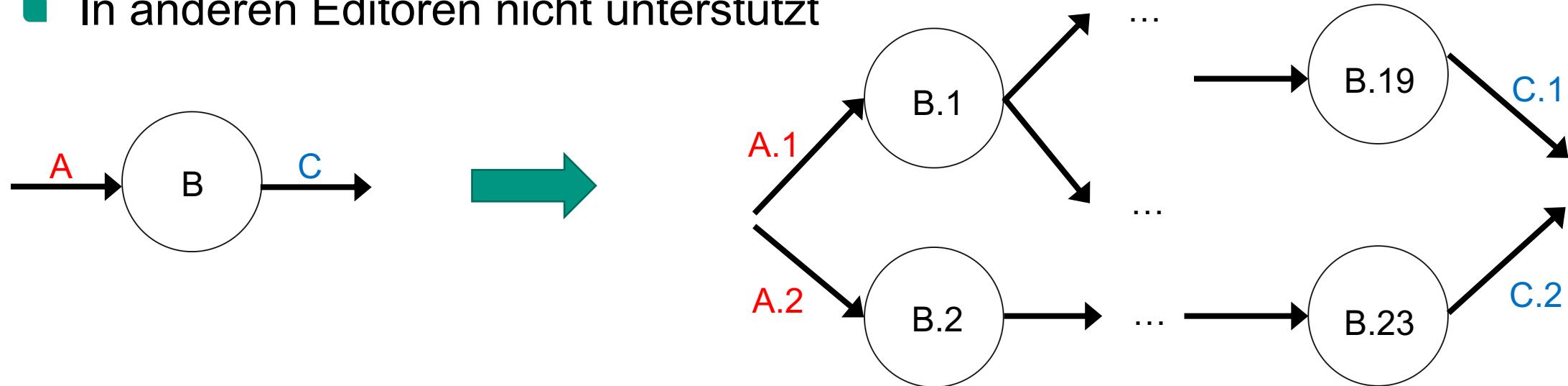
Motivation (1)

- Datenflussdiagramme: Verwendung in vielen Szenarien
 - Requirements Engineering
 - Sicherheitsanalyse
- Bisher: Kein Standard-Editor vorhanden
 - Unterschiedliche Werkzeuge im Industrieeinsatz
 - Nur unzureichende Editoren im Eclipse-Umfeld
- Zeichenprogramme möglich, aber:
 - Semantik geht verloren
 - Keine semantische Unterstützung bei Erstellung

Motivation (2)

■ Datenflussdiagramme: Hierarchisierung möglich

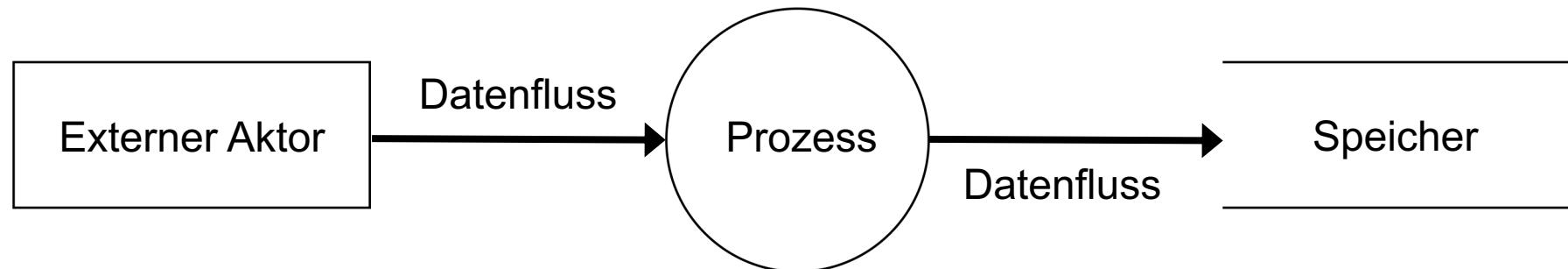
- Versch. Abstraktionsebenen
- Konsistenzbedingungen
- In anderen Editoren nicht unterstützt



Ziel: Implementierung eines solchen Editors

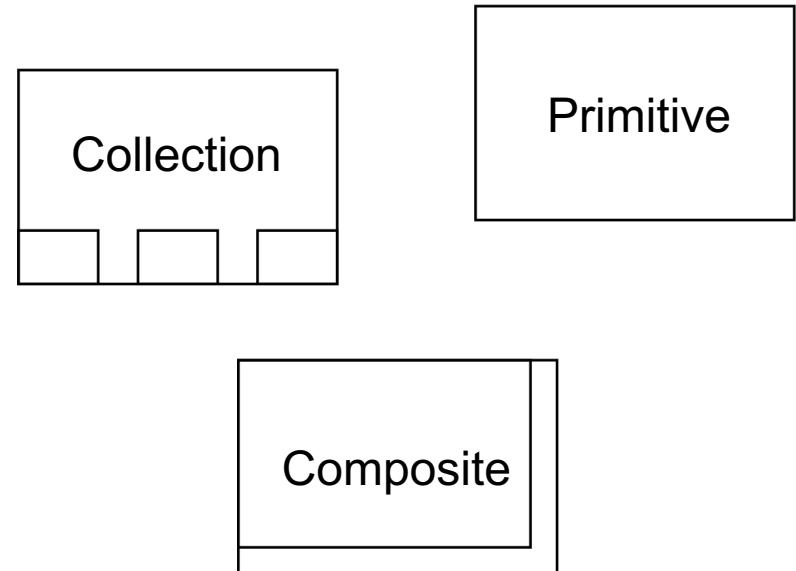
Datenflussdiagramme

- Datenorientierte Darstellung von Systemen
- 4 Entitäten
- Verfeinern in Subdiagramme möglich
- Data Dictionary enthält Informationen über Datentypen

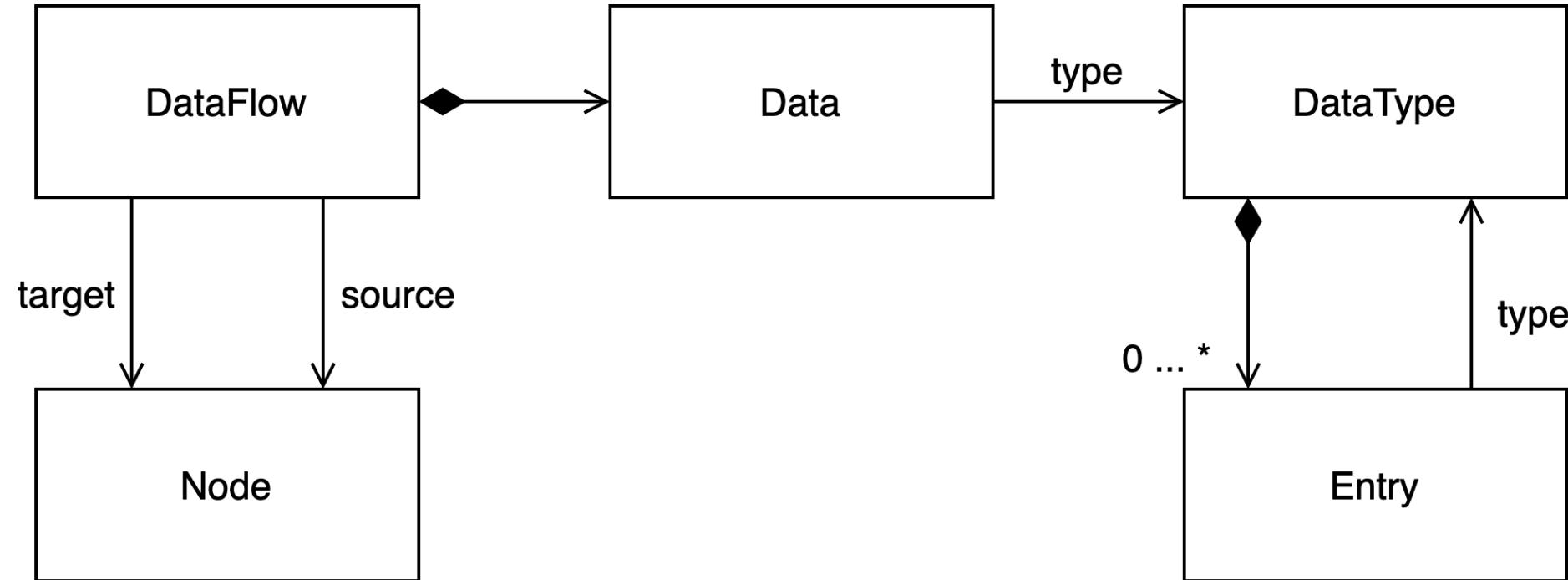


Data Dictionary

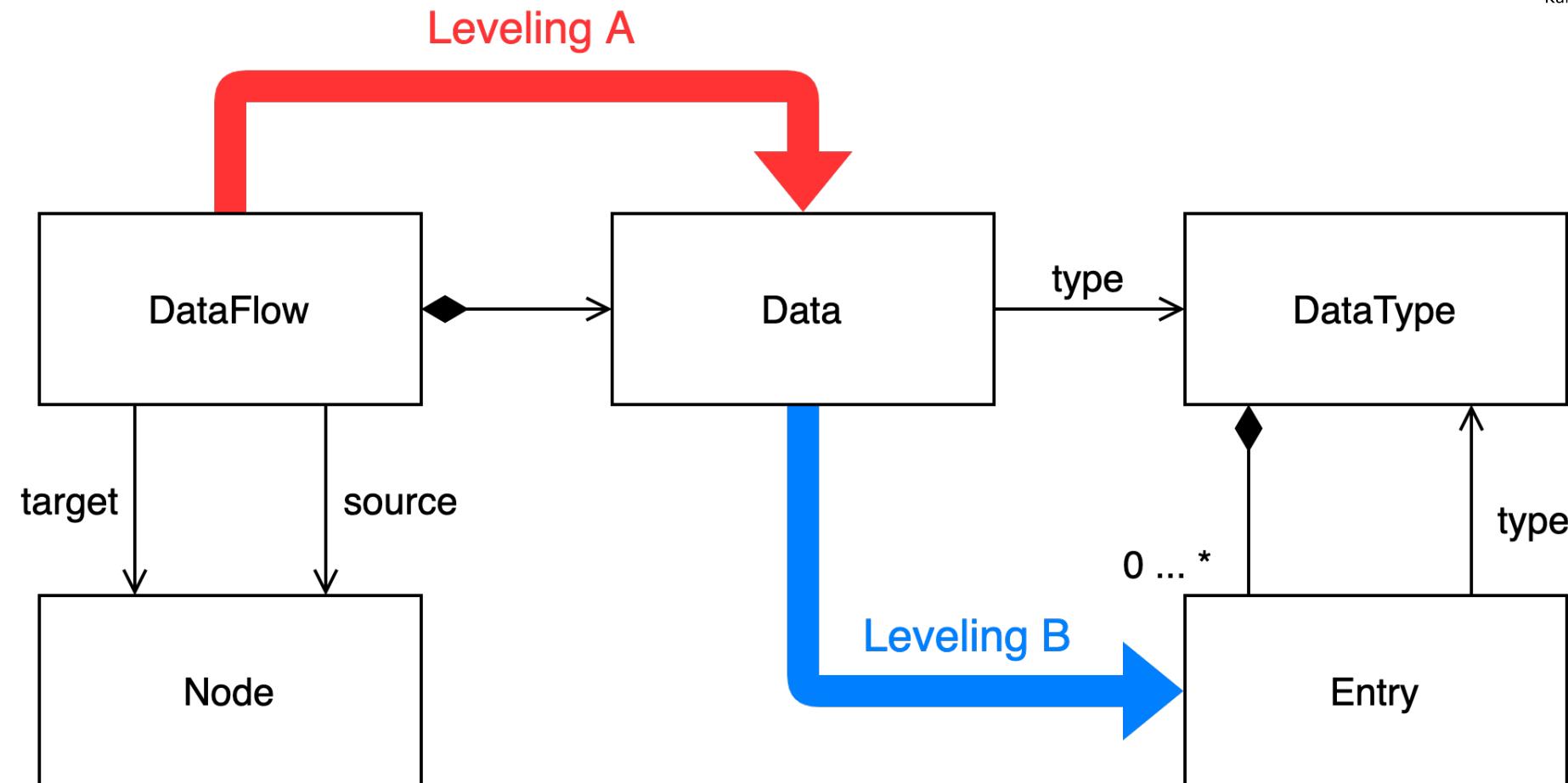
- Definition von Datentypen
- 3 Entitäten
- Grundlage für
 - Hierarchisierung
 - Prüfung von Konsistenzbedingungen



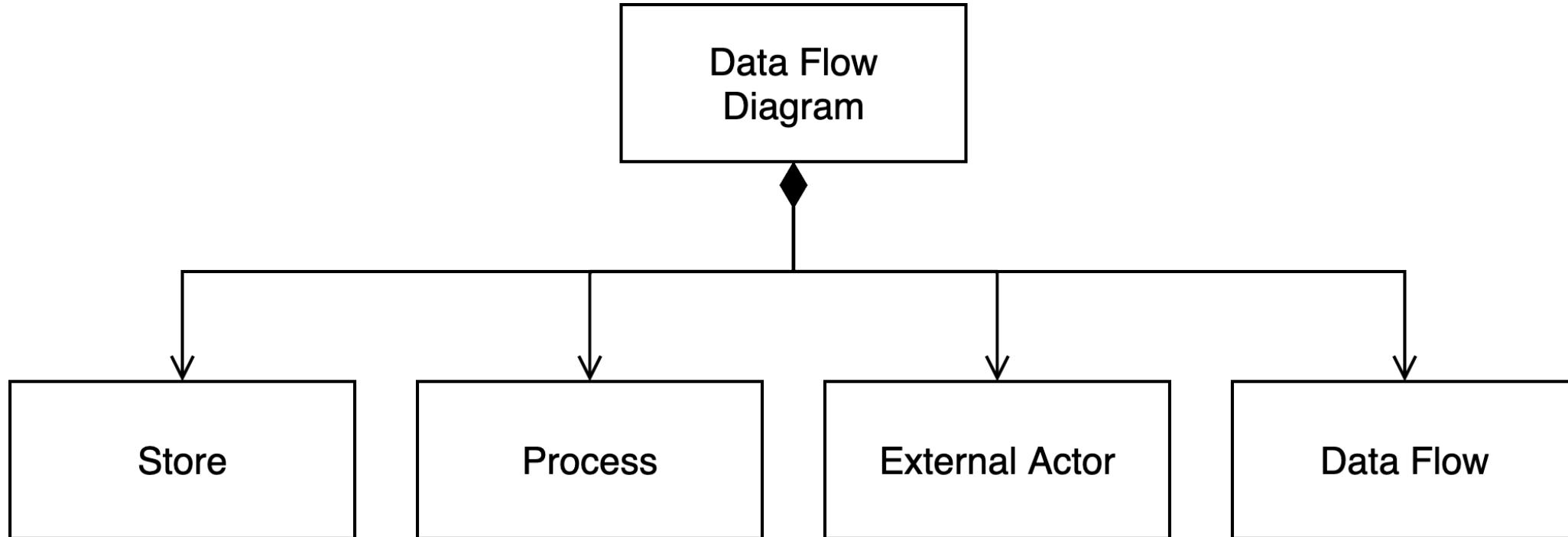
Hierarchisierung (1)



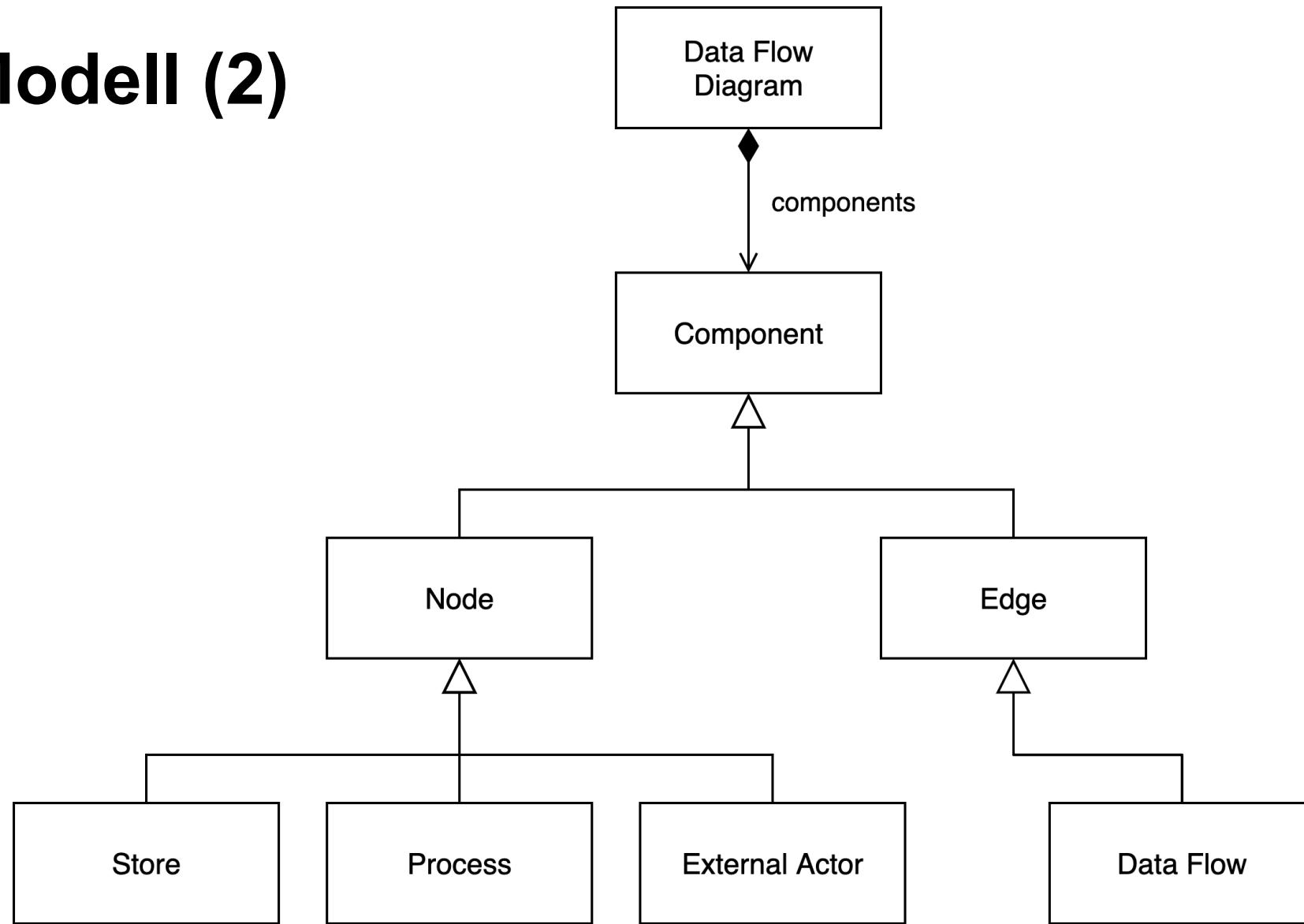
Hierarchisierung (2)



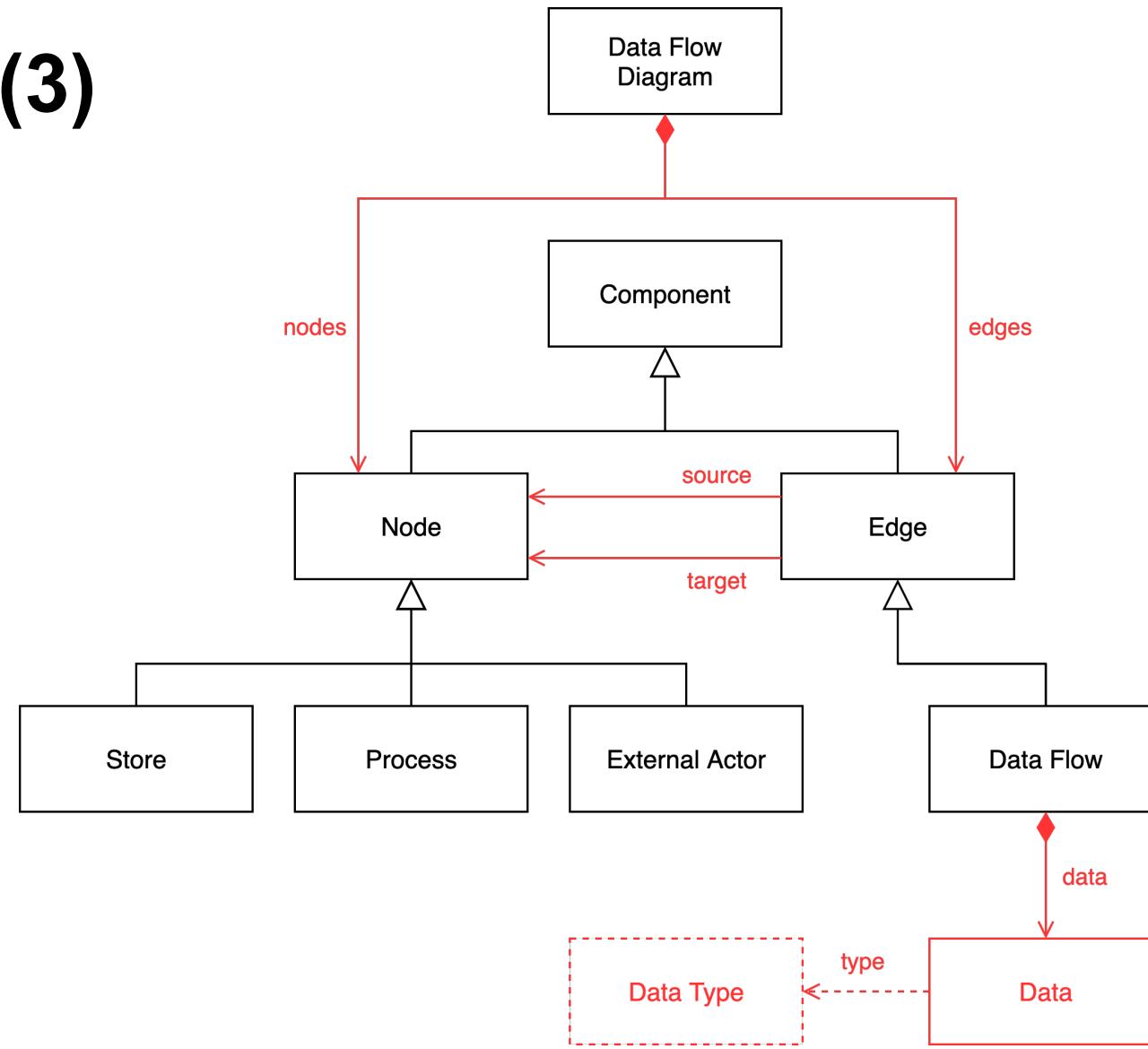
Meta-Modell (1)

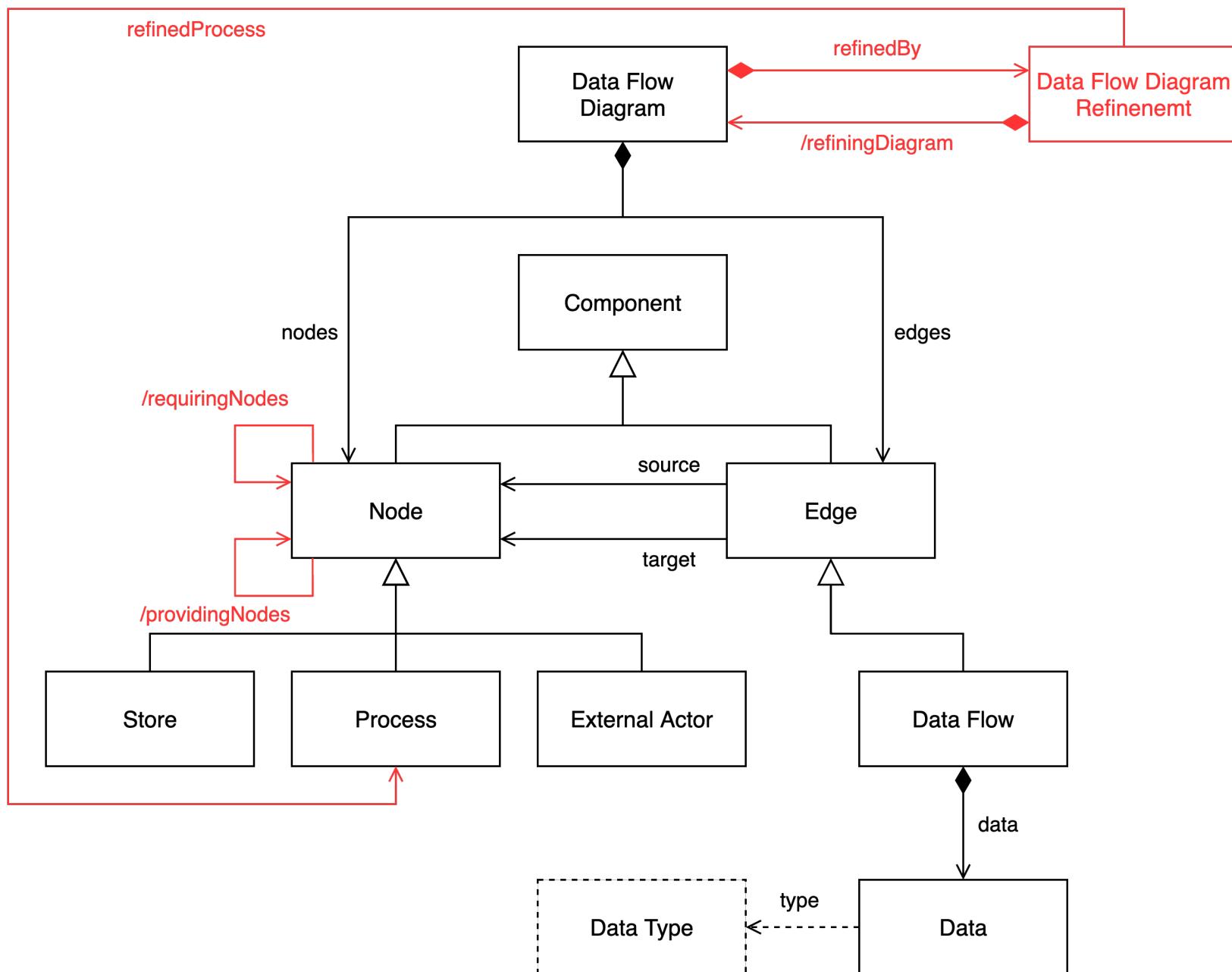


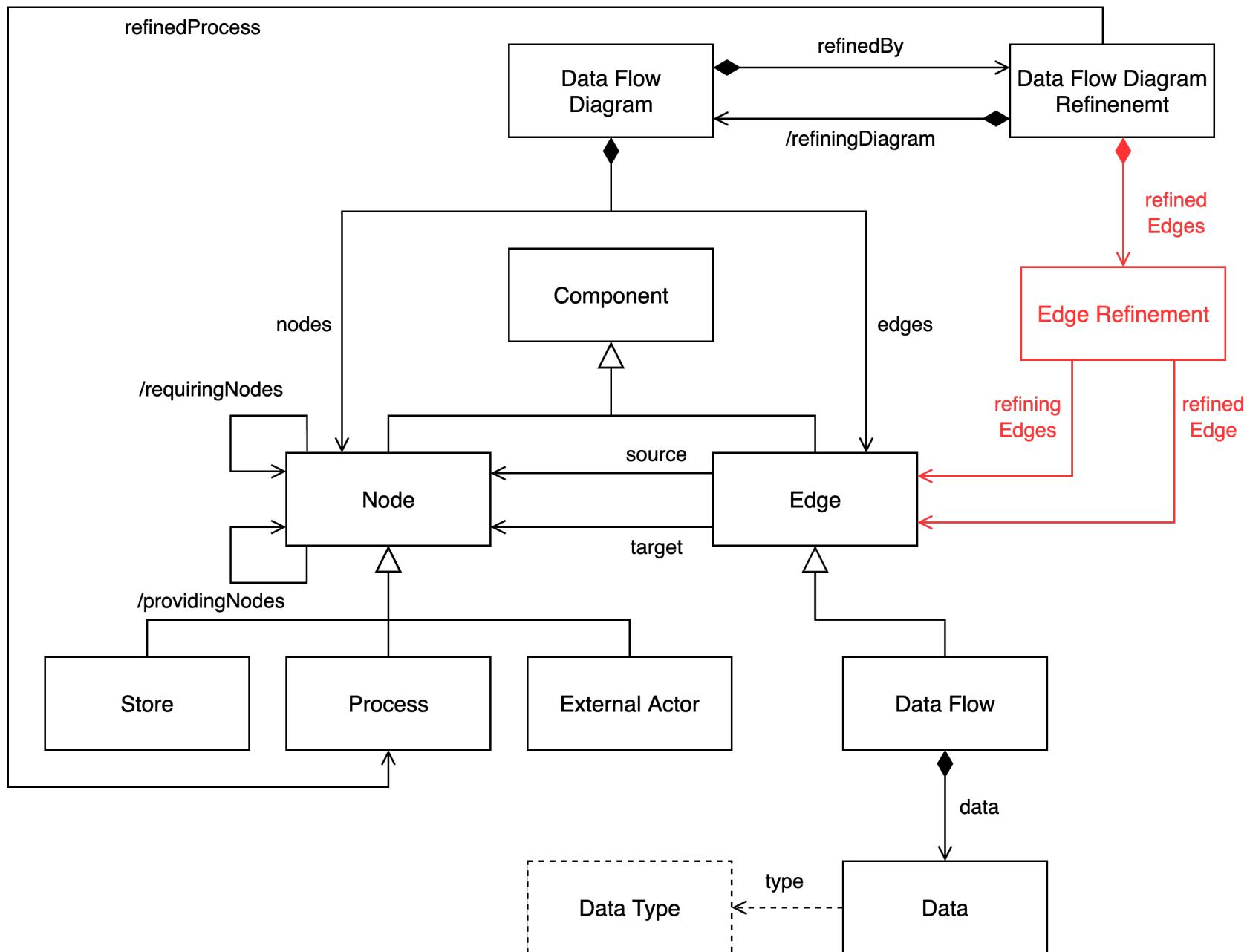
Meta-Modell (2)



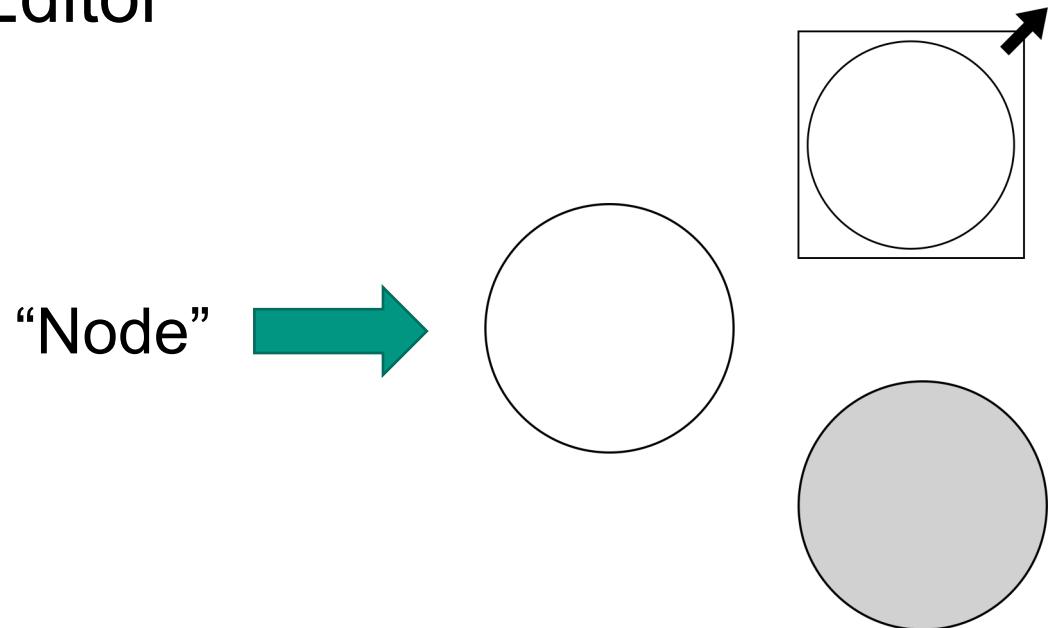
Meta-Modell (3)



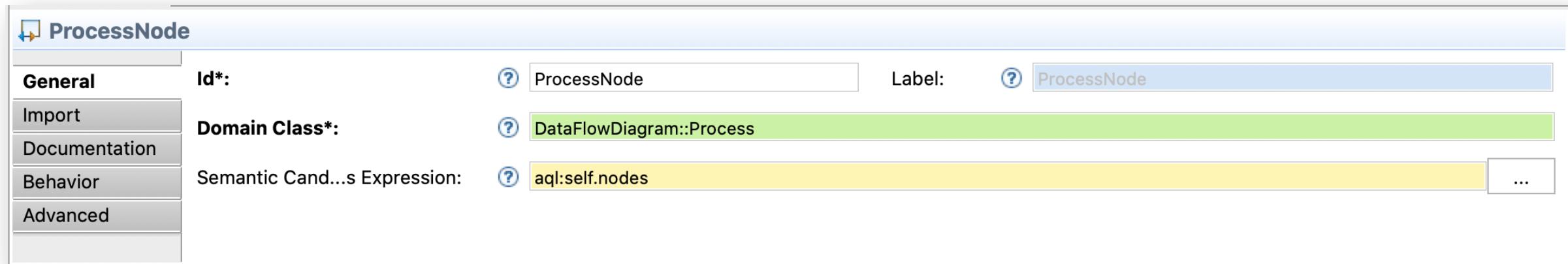
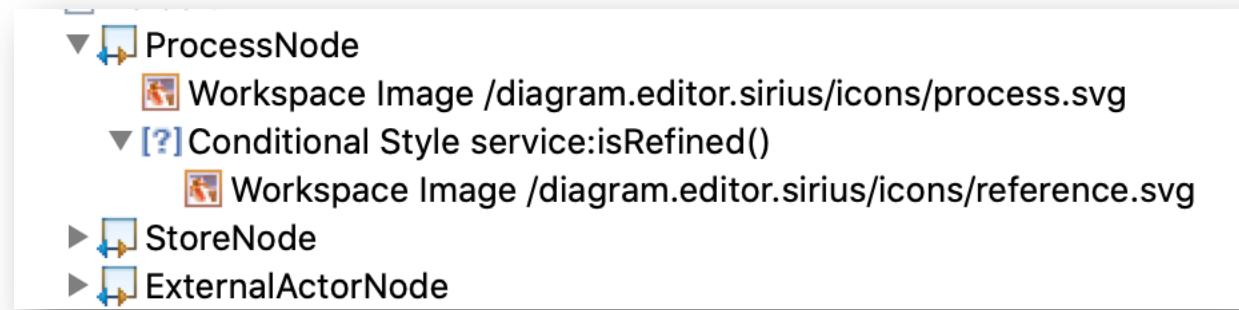




- Framework für Erstellung von Editoren
 - Verfügbare Elemente gegeben durch Meta-Modelle
 - Darstellung frei konfigurierbar
- Logische Trennung von Modell und Editor
 - Modell enthält Daten (in .xmi-Datei)
 - Editor als GUI für dessen Modifizierung



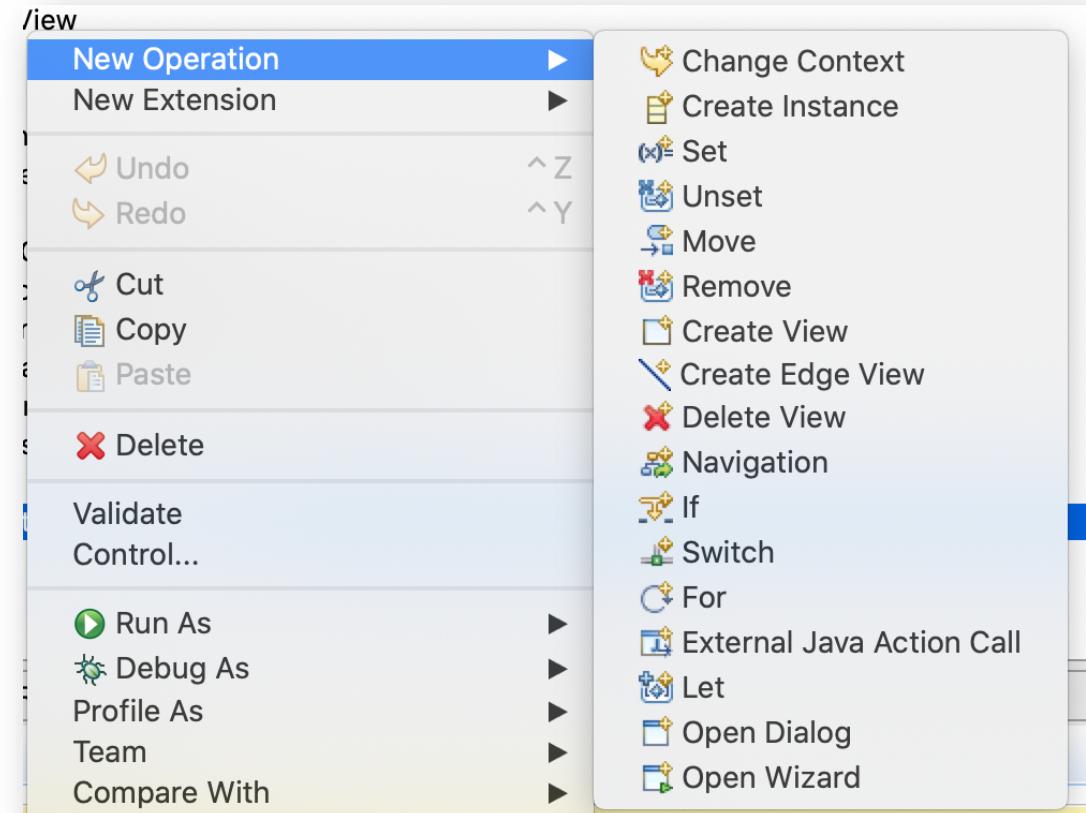
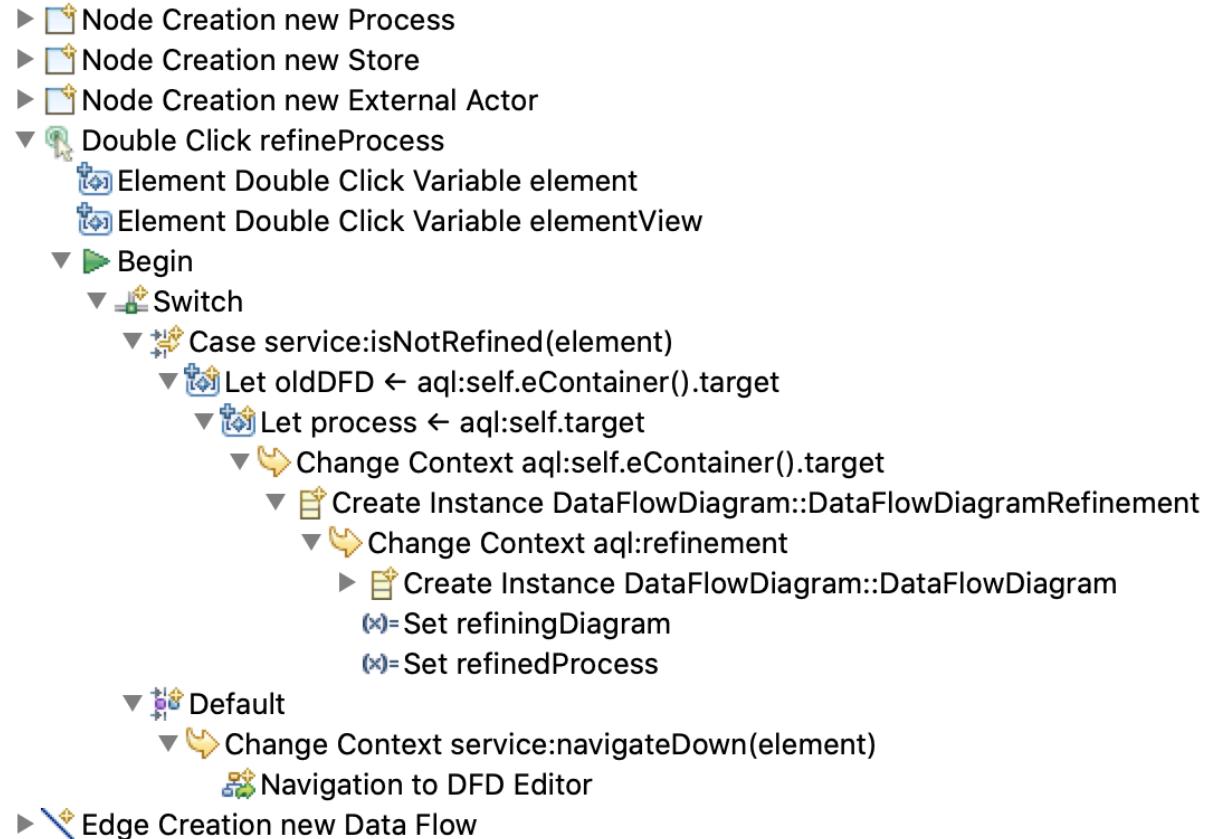
Eclipse – Sirius: Elemente



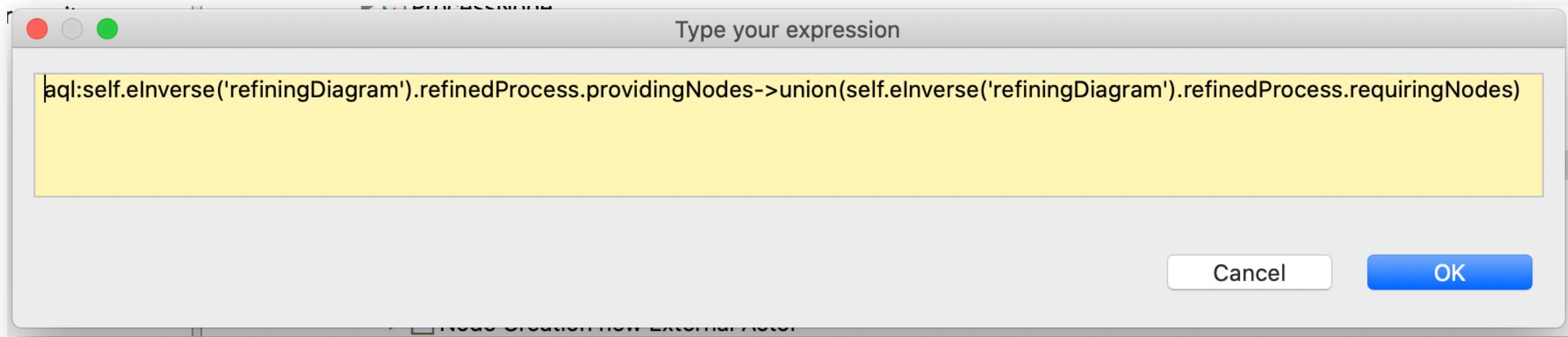
The screenshot shows the Sirius Eclipse interface with the **ProcessNode** selected in the tree view. The properties view displays the following settings:

General	Id*:	(?) ProcessNode	Label:	(?) ProcessNode
Import	Domain Class*:	(?) DataFlowDiagram::Process		
Documentation	Semantic Candidates Expression:	(?) aql:self.nodes		
Behavior				
Advanced		...		

Eclipse – Sirius: Operationen



Eclipse – Sirius: AQL



Eclipse – Sirius: Java-Services

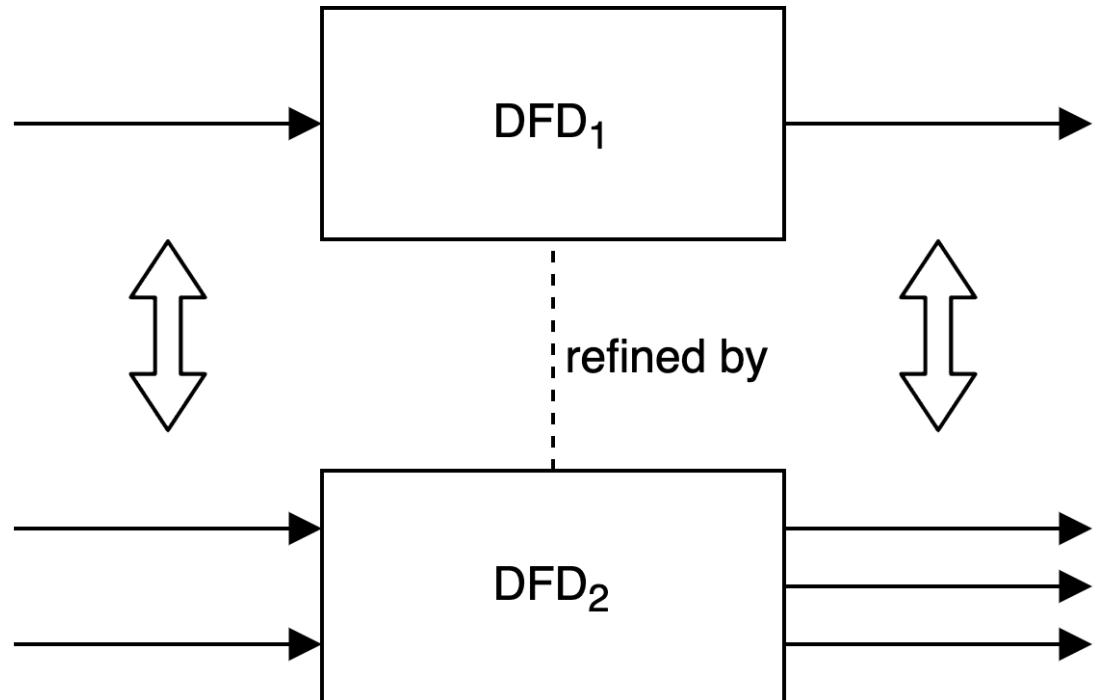
```
public void loadResources(EObject self) {
    ResourceDialog r = new ResourceDialog(Display.getCurrent().getActiveShell(), "Load Data Dictionary",
        SWT.SINGLE);
    r.open();
    r.setBlockOnOpen(true);
    Session session = SessionManager.INSTANCE.getSession(self);
    for (URI uri : r.getURIs()) {
        if (!DFDTypeUtil.uriAlreadyLoaded(uri, session))
            session.addSemanticResource(uri, new NullProgressMonitor());
    }
}
```

Validierung (1)

■ Frage: „Ist eine gegebene Verfeinerung konsistent?“

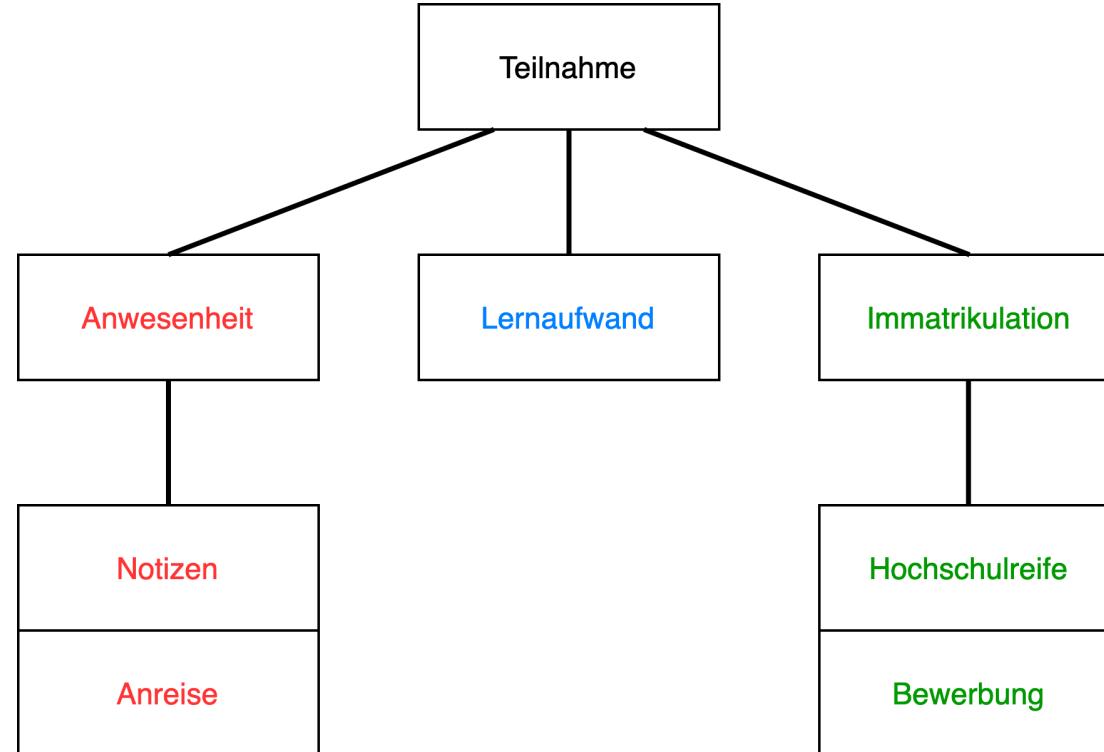
■ Probleme:

- Automatische Hierarchisierung
- Abbildung der Datenflüsse
- Nutzereingabe

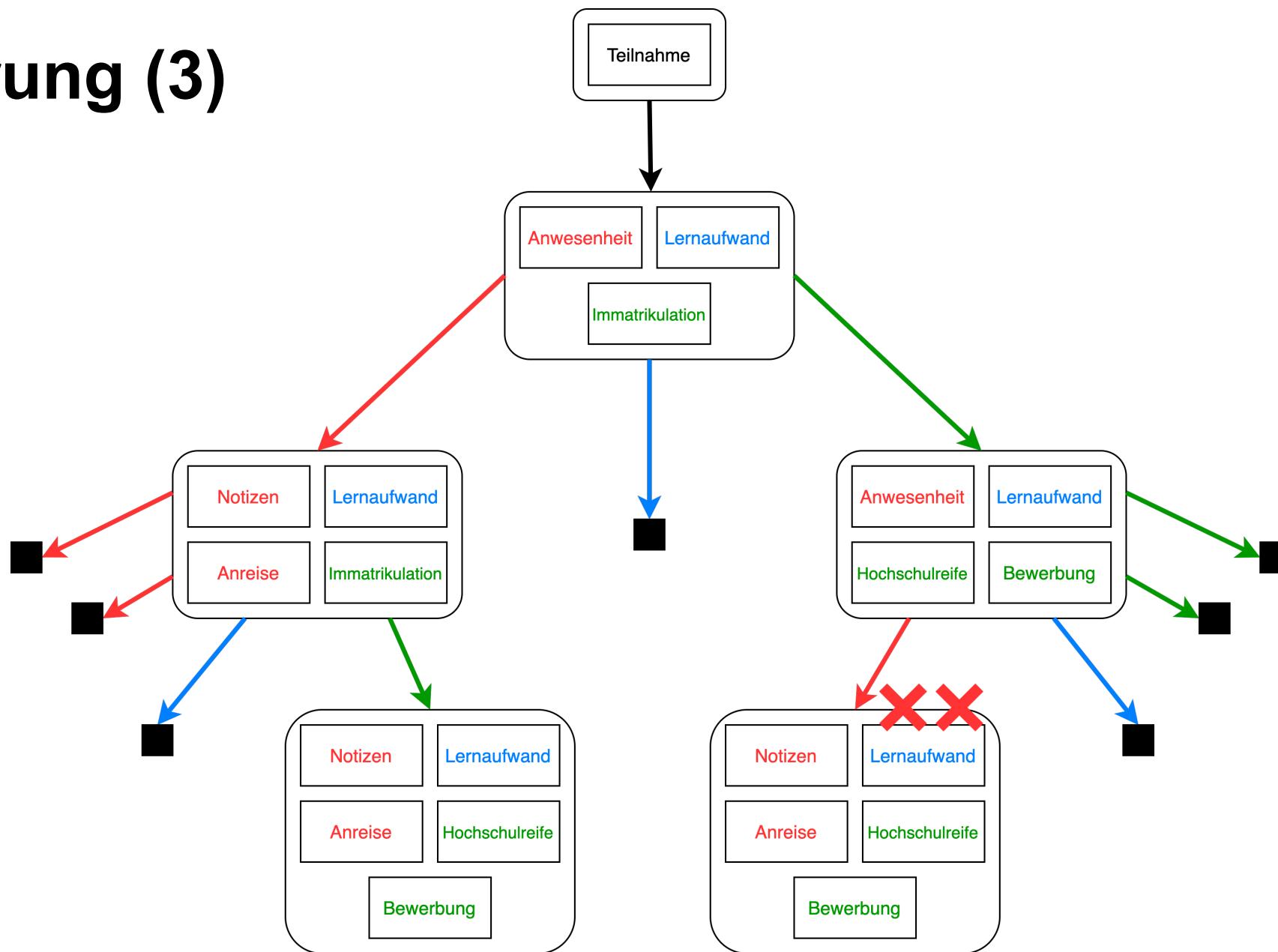


Validierung (2)

■ Beispiel: Teilnahme an Lehrveranstaltung



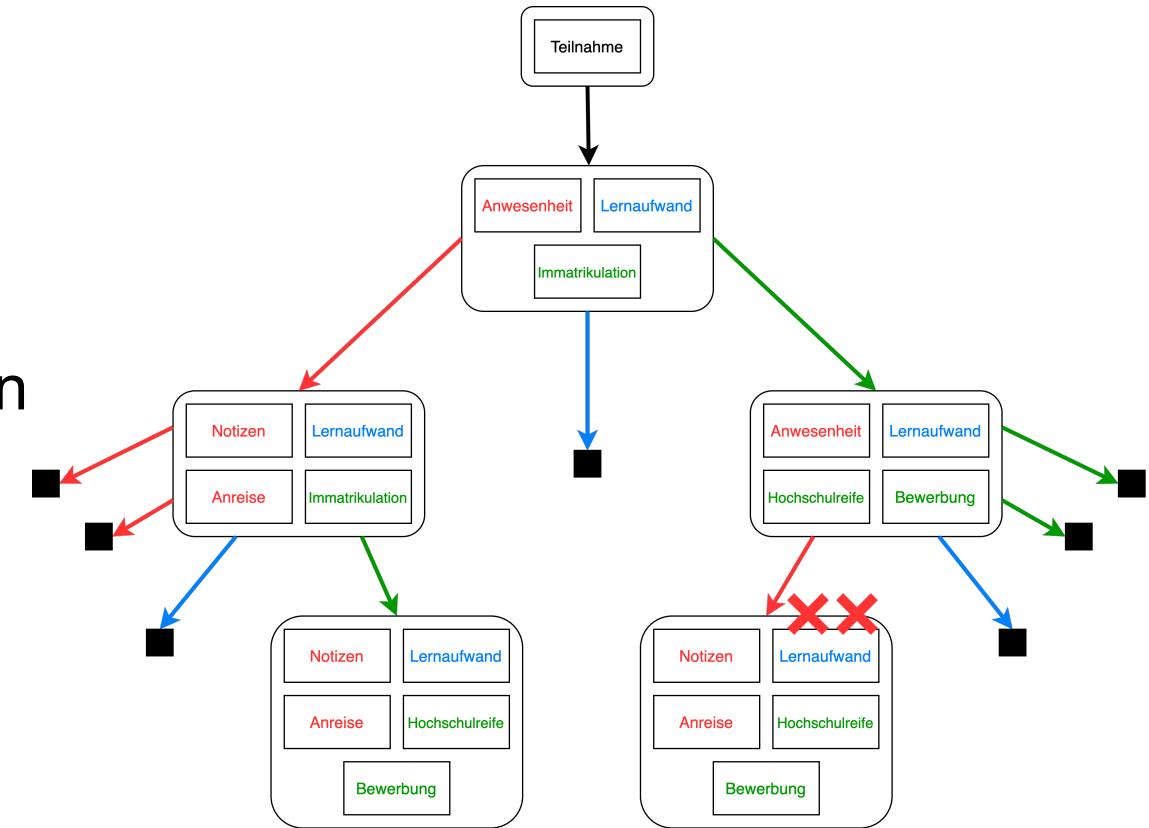
Validierung (3)



Validierung (4)

■ Future Work:

- Early Pruning
- Schrittweite
- Vorberechnung
- Notwendige vs. hinreichende Kriterien
- „Greedy“-Strategie möglich?
- Erklärbarkeit



■ Szenario: Teilnahme an Lehrveranstaltung

Motivation ➤ Grundlagen ➤ Meta-Modell ➤ Implementierung ➤ **Demo** ➤ Zusammenfassung

Zusammenfassung

- Datenflussdiagramme: Verwendung in vielen Bereichen
 - Datenorientierte Sicht
 - Versch. Abstraktionsebenen
- Bislang existierte kein geeigneter Editor
- Hierarchisierung konzeptionell schwerster Aspekt
 - Genau zu definierende Semantik
 - Validierung der Konsistenz
- Mithilfe von Eclipse Sirius: Modellorientierte Entwicklung