(Pro-)Seminar Softwaresystemtechnik (SST)

“Modellbasierte Qualitätssicherungsmaßnahmen”

(SS 14, Proseminar 2 CP | Seminar 4 CP)

TECHNISCHE UNIVERSITÄT DARMSTADT

Einführungsveranstaltung

Johannes Bürdek

johannes.buerdek@es.tu-darmstadt.de
Tel.+49 6151 16 76089

ES Real-Time Systems Lab
Prof. Dr. rer. nat. Andy Schürr
Dept. of Electrical Engineering and Information Technology
Dept. of Computer Science (adjunct Professor)

www.es.tu-darmstadt.de

17.04.2014

Herzlich Willkommen!

TECHNISCHE UNIVERSITÄT DARMSTADT

Als Proseminar (mit reduzierten Ansprüchen, 2 CP):

- B.Sc. ETiT (5. Sem.)

Als Seminar (4 CP):

- B.Sc. (5. Sem.) und M.Sc. Informatik (2. Sem.)
- Dipl. Informatik
- B.Sc. Informationssystemtechnik (5. Sem.)
- Dipl. ETiT (DT, Hauptstudium)
- entsprechende Wirtschaftsstudiengänge
- Sonstige passende Fachrichtung

Was wir von den Teilnehmern erwarten…

TECHNISCHE UNIVERSITÄT DARMSTADT

- Interesse am Thema + Motivation
- Wille zur Zusammenarbeit mit
  - Betreuer
  - Kommilitonen bei einer Gruppenarbeit
- Wissenschaftliches Vorgehen (unter Anleitung)
- Fristgerechte Abgabe der geforderten Arbeiten
- Teilnahme an *allen* Pflichtveranstaltung

ACHTUNG: Das Seminar ist inhaltlich und vom Umfang her anspruchsvoll!
→Wir geben uns Mühe bei der Betreuung, und erwarten im Gegenzug von allen Teilnehmern ebenfalls **vollen Einsatz**!!!

Ziele

TECHNISCHE UNIVERSITÄT DARMSTADT

- Grundfertigkeiten zur Erstellung einer **wissenschaftlichen Arbeit**
  - Selbständiges Erarbeiten eines Themengebietes (unter Anleitung)
  - Literaturrecherche
    → finden, lesen, verstehen, bewerten
  - Wissenschaftliches Schreiben
    → Gliedern, Zitieren, Formulieren

- Mitwirken am **Reviewprozess**
  - Verwendbares Feedback zu fremden Arbeiten geben
    → Gegenseitige Unterstützung, Schwachstellen identifizieren

- **Präsentation**
  - Aufbereiten, bewerten der Ergebnisse
  - Vorstellen der Ergebnisse
    → Techniken, Stil, Zeiteinteilung, Reden vor der Gruppe

Organisatorisches

TECHNISCHE UNIVERSITÄT DARMSTADT

- Heute
  - Themenvorstellung und Auswahl
  - Themenvergabe durch uns

- Während des Semesters
  - Erstellen einer Ausarbeitung
    → (Auf-)Schreiben von (Zwischen-) Ergebnissen
  - Vortrag vorbereiten
  - Regelmäßige Absprachen mit Betreuer!
    → Individuelle Absprachen
    → Fortschritt, Fragen, Feedback, Tipps

- Am Ende des Semesters **(11. Juli 2014)**
  - Vortrag im Blockseminar → Präsentation + Ausarbeitung liegen bereits vor

Plagiatshinweis – „Abschreiben“ verboten!

TECHNISCHE UNIVERSITÄT DARMSTADT

- Wir messen der Einhaltung der Grundregeln der wissenschaftlichen Ethik großen Wert bei.

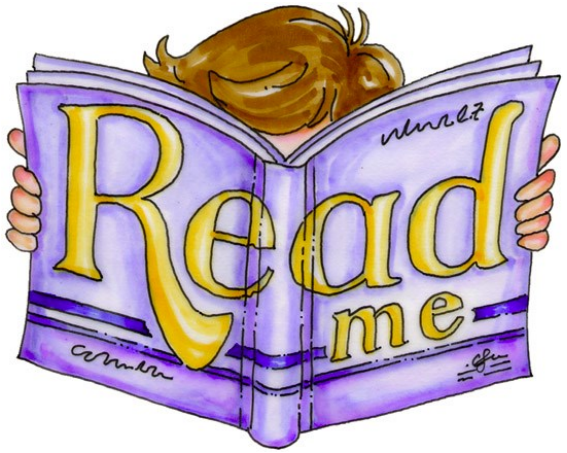
- Mit der Abgabe einer Lösung (Hausaufgabe, Programmierprojekt, Diplomarbeit, etc.) bestätigen Sie, dass (Sie/Ihre Gruppe) (der alleinige Autor/die alleinigen Autoren) des gesamten Materials sind. Falls Ihnen die Verwendung von Fremdmaterial gestattet war, so müssen Sie dessen Quellen deutlich zitiert haben.

- Weiterführende Informationen unter http://www.es.tu-darmstadt.de/lehre/plagiat/

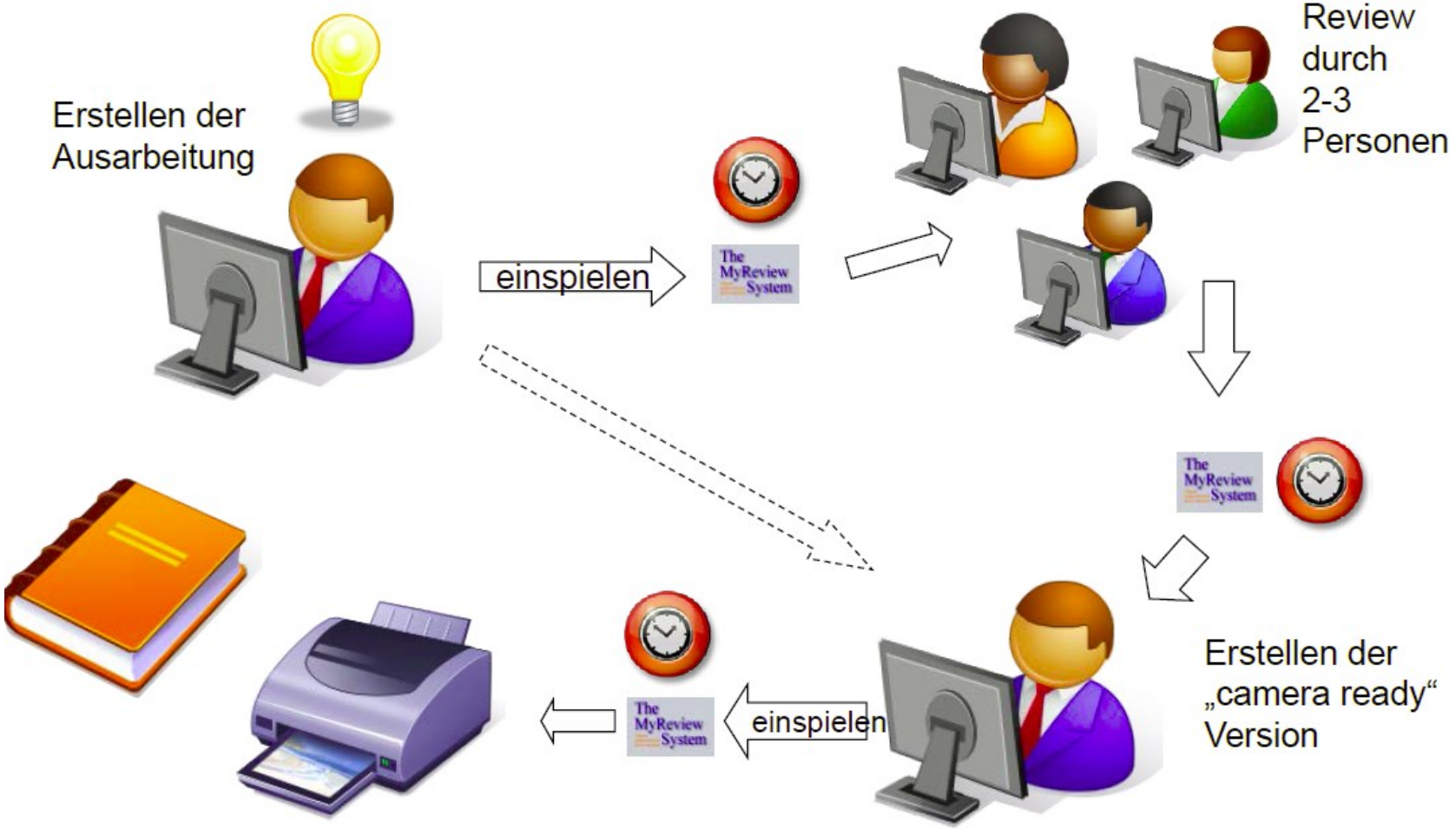
Wichtige Links zur Veranstaltung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Mailingliste
  - Jeder Teilnehmer bitte Eintragen
  - Verwenden für Fragen, Diskussionen, …
  - http://liste.es.e-technik.tu-darmstadt.de/mailman/listinfo/sst
- Seminarrichtlinien
  - http://www.es.tu-darmstadt.de/fileadmin/download/lehre/Seminarrichtlinien.pdf
- Plagiatshinweise
  - http://www.es.tu-darmstadt.de/lehre/plagiat/



Übersicht Review-Prozess



Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Qualitätssichernde Maßnahmen in der Softwaretechnik
  - Vergleich des Produktes auf seine Spezifikation
  - Verbesserung der Qualität und Wartbarkeit
  - Sicherheit garantieren
  - …

- Kennenlernen moderner Verfahren für Qualitätsicherung mit Hilfe von
  - Testen
  - Testfallgenerierung
  - Graphtransformation
  - Model Checker
  - Spieltheorie
  - und mehr…

Zeitplan

TECHNISCHE UNIVERSITÄT DARMSTADT

17.04.2014	Vorbesprechung / Themenauswahl
09.05.2014	Einarbeitung beendet, erste Gliederung
01.06.2014	Erste Fassung der Ausarbeitung (5-6 doppelspaltige Seiten)
05.06.2014	Einspielen der Ausarbeitung ins Reviewsystem
15.06.2014	Reviews eingereicht
25.06.2014	Endfassung der Ausarbeitung
01.07.2014	Erste Fassung der Vortragsfolien
07.07.2014	Endfassung der Vortragsfolien
11.07.2014	Abschlussvortrag (20 Min., mit anschl. Diskussion)

Themenübersicht

TECHNISCHE UNIVERSITÄT DARMSTADT

Thema	PS
Comparison of two different model transformation testing approaches	
Recursive Techniques in Graph Transformation Approaches	
Satisfiability Modulo Theorie Solver	
Visuelle Spezifikation von strukturellen und temporalen Eigenschaften: Eine Sprachvergleichsstudie	
Tree-Diff Algorithms	X
Model-checking using GROOVE	X
Test Case Generation with Model Checking	X
Evolution of Software Product Lines	X
Testing with Reachability Games	

TECHNISCHE UNIVERSITÄT DARMSTADT

Comparison of two different model transformation testing approaches

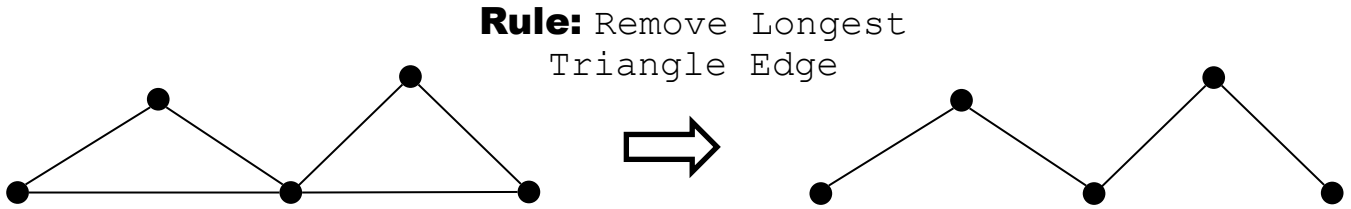
SST Thema – SS 14

Géza Kulcsár

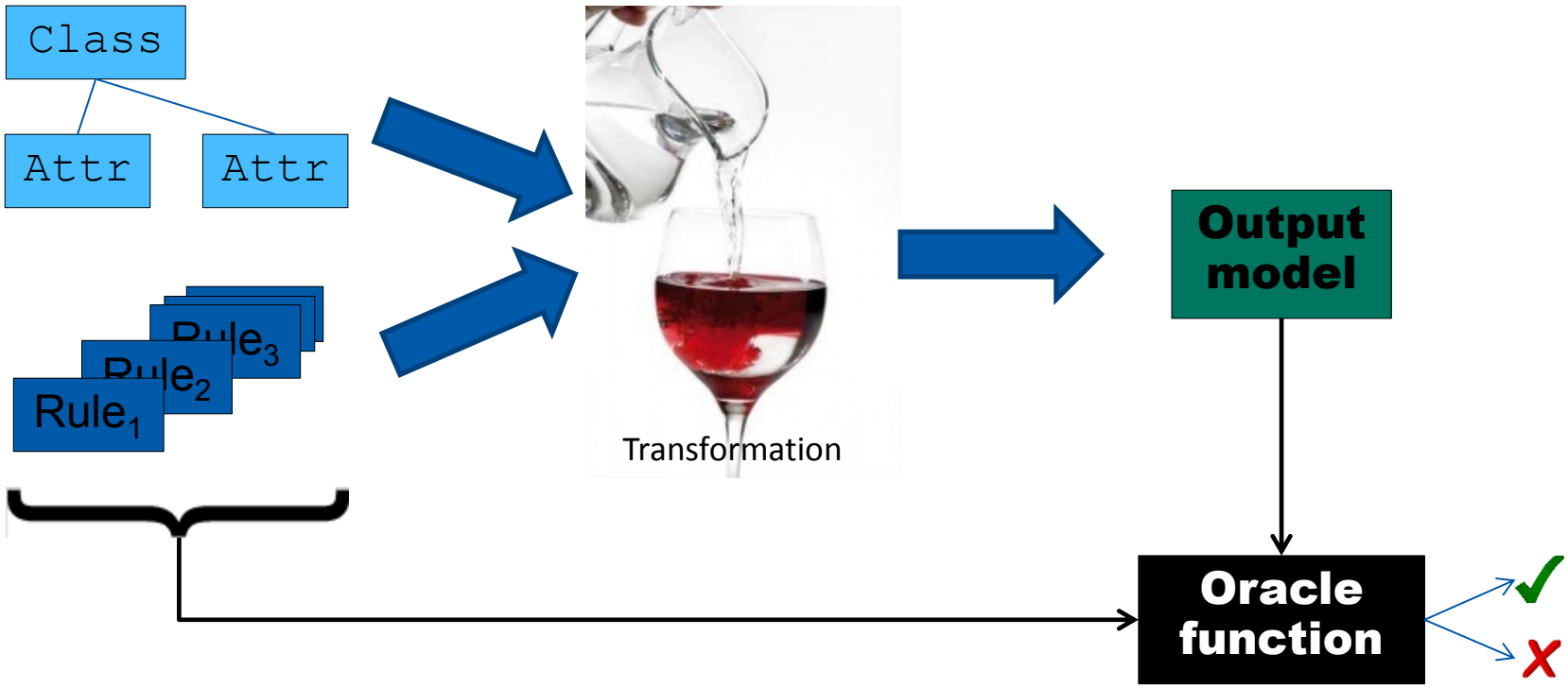
Graph transformations

TECHNISCHE UNIVERSITÄT DARMSTADT

- Systems represented by typed graphs
- Declarative rules how to modify graphs
- Graph transformation: a sequence of rule applications



Model transformation verification



- Aim of this work: analyzing and comparing two different verification approaches with as many aspects as possible

TECHNISCHE UNIVERSITÄT DARMSTADT

Rekursive Techniken in Graphtransformations-ansätzen

Gergely Varro

Rekursive Techniken in Graphtransformations-ansätzen

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation:

- Effiziente und skalierbare Techniken sind ohne Zweifel nötig, um die Verbreitung von modellgetriebenen Technologien in einem industriellen Kontext voranzutreiben. Verschiedene Szenarien (wie z.B. die Überprüfung von komplexen Anwendungsbedingungen in regelbasierten Modelltransformationswerkzeugen oder Konsistenzvalidierung) können als Graphmustersuche beschrieben werden. Deshalb spielt ihre effiziente Implementierung eine wichtige Rolle. Im Rahmen dieser Aufgabenstellung sollen solche existierende Algorithmen und Ansätzen untersucht werden, die strukturelle Wiederholungen (Rekursion) in den Musterspezifikationen erlauben.

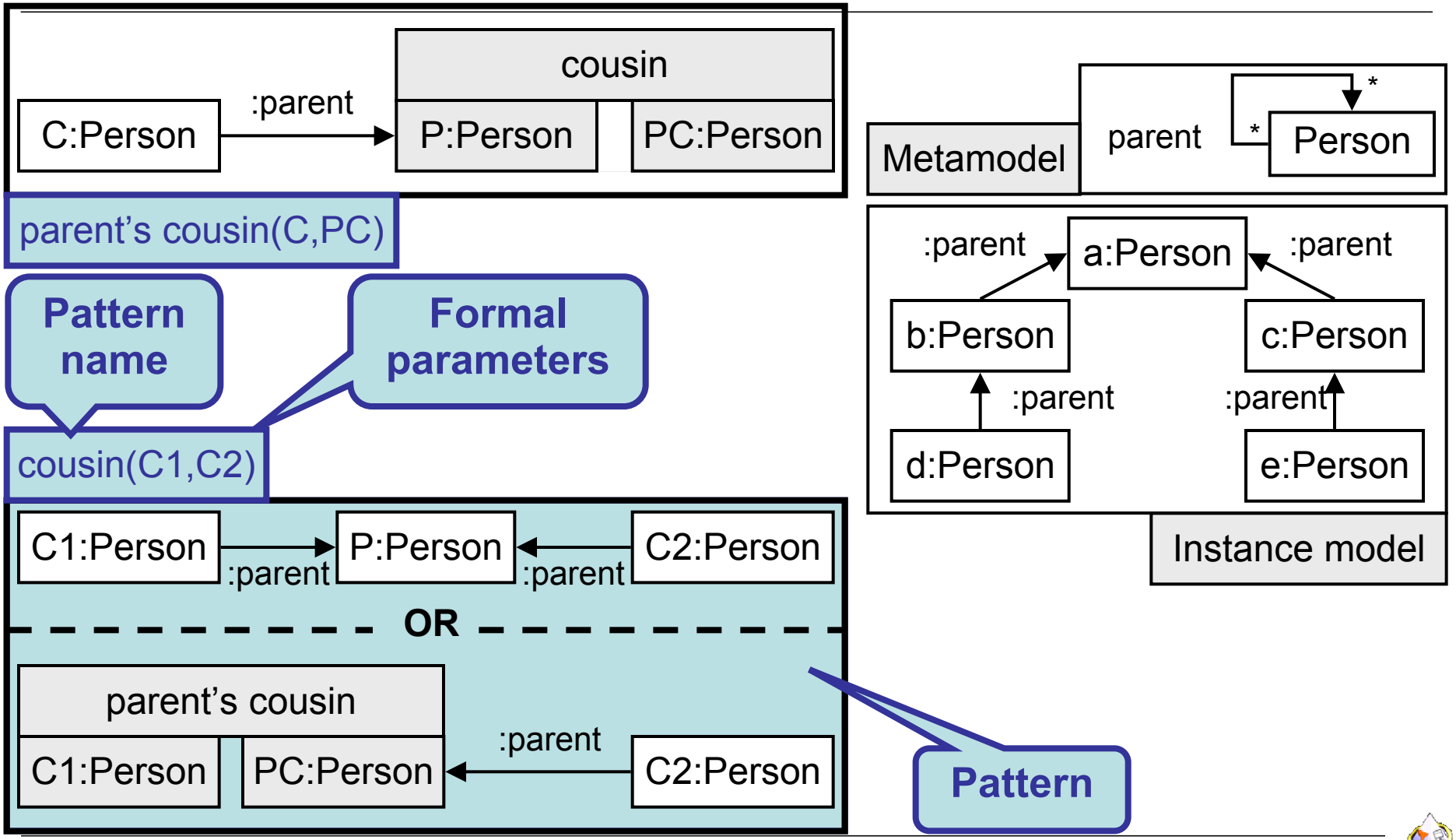
Aufgabe:

- Das Konzept komplexer und rekursiver Muster durch Lesen und Verstehen der passenden Publikationen kennenzulernen. Hierzu ist es erforderlich, englische Fachtexte lesen und verstehen zu können.
- Die existierenden Ansätze zusammenzufassen, zu vergleichen und vorzutragen.

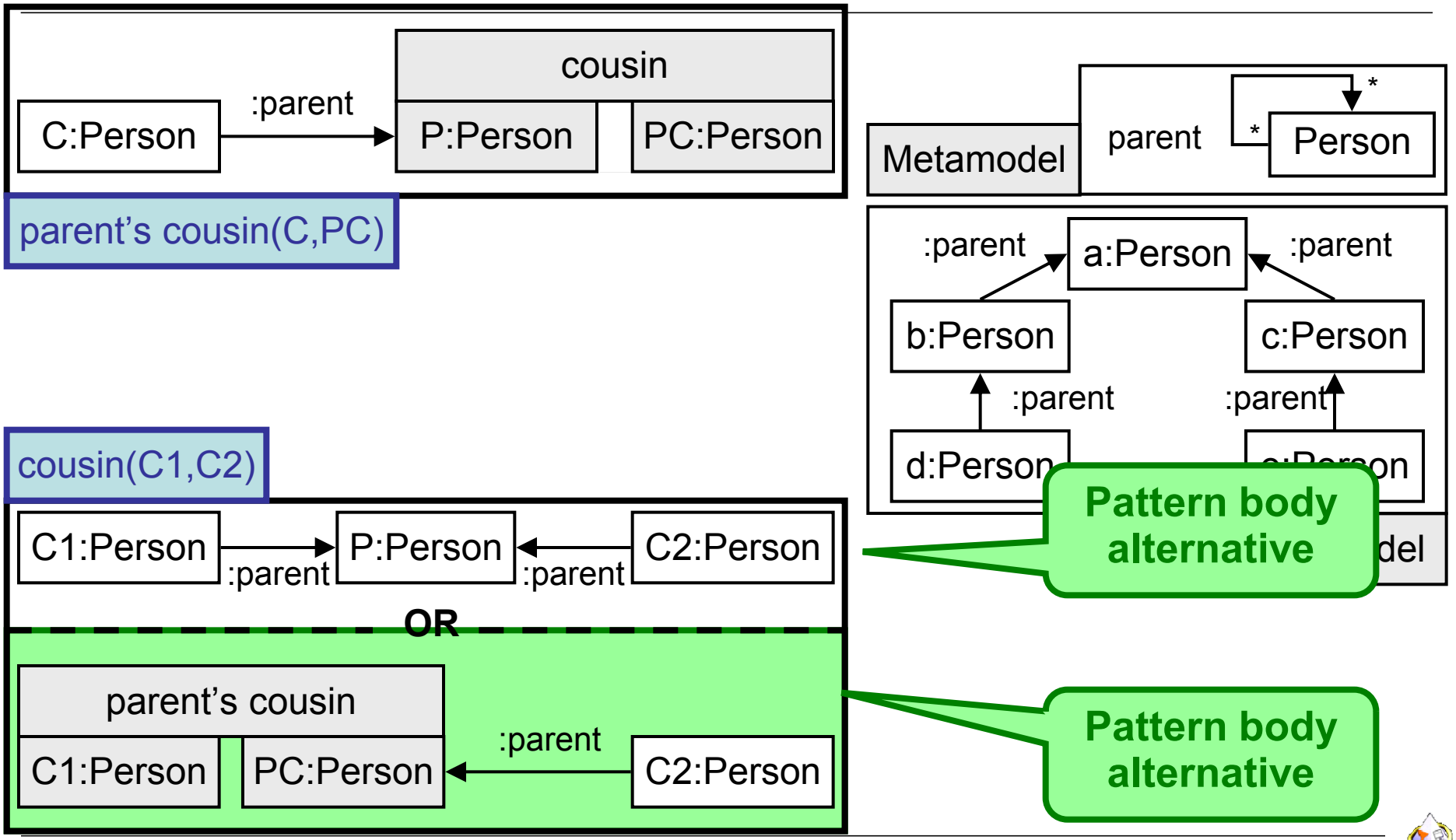
Links/Literatur:

- Graph Pattern Matching in PROGRES
- A Query Language with the Star Operator
- Modeling Successively Connected Repetitive Subgraphs
- Reusable Idioms and Patterns in Graph Transformation Languages
- Practical Declarative Model Transformation with Tefkat
- Recursive Graph Pattern Matching with Magic Sets and Global Search Plans

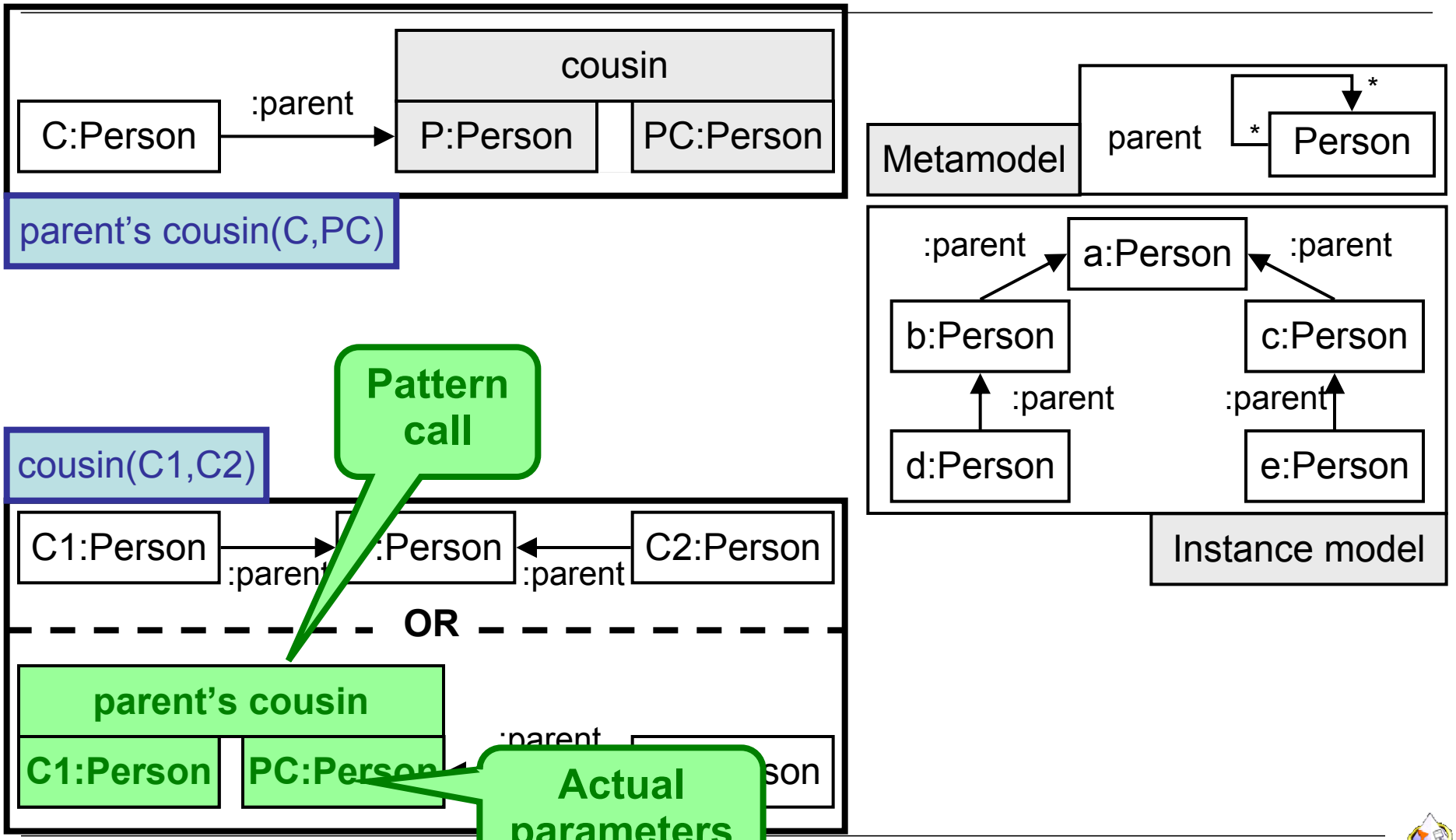
Rekursive Techniken in Graphtransformations-ansätzen



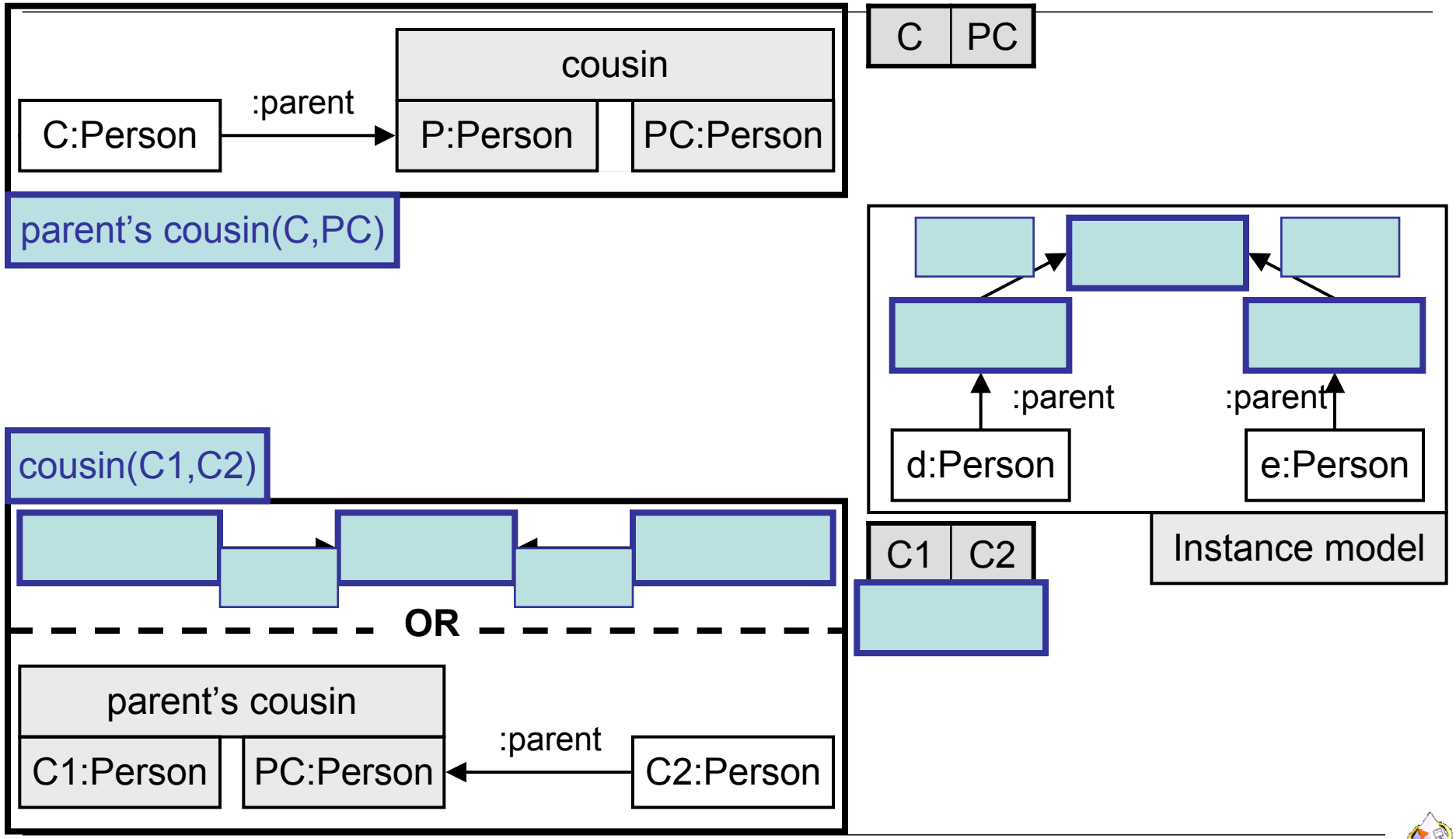
Rekursive Techniken in Graphtransformations-ansätzen



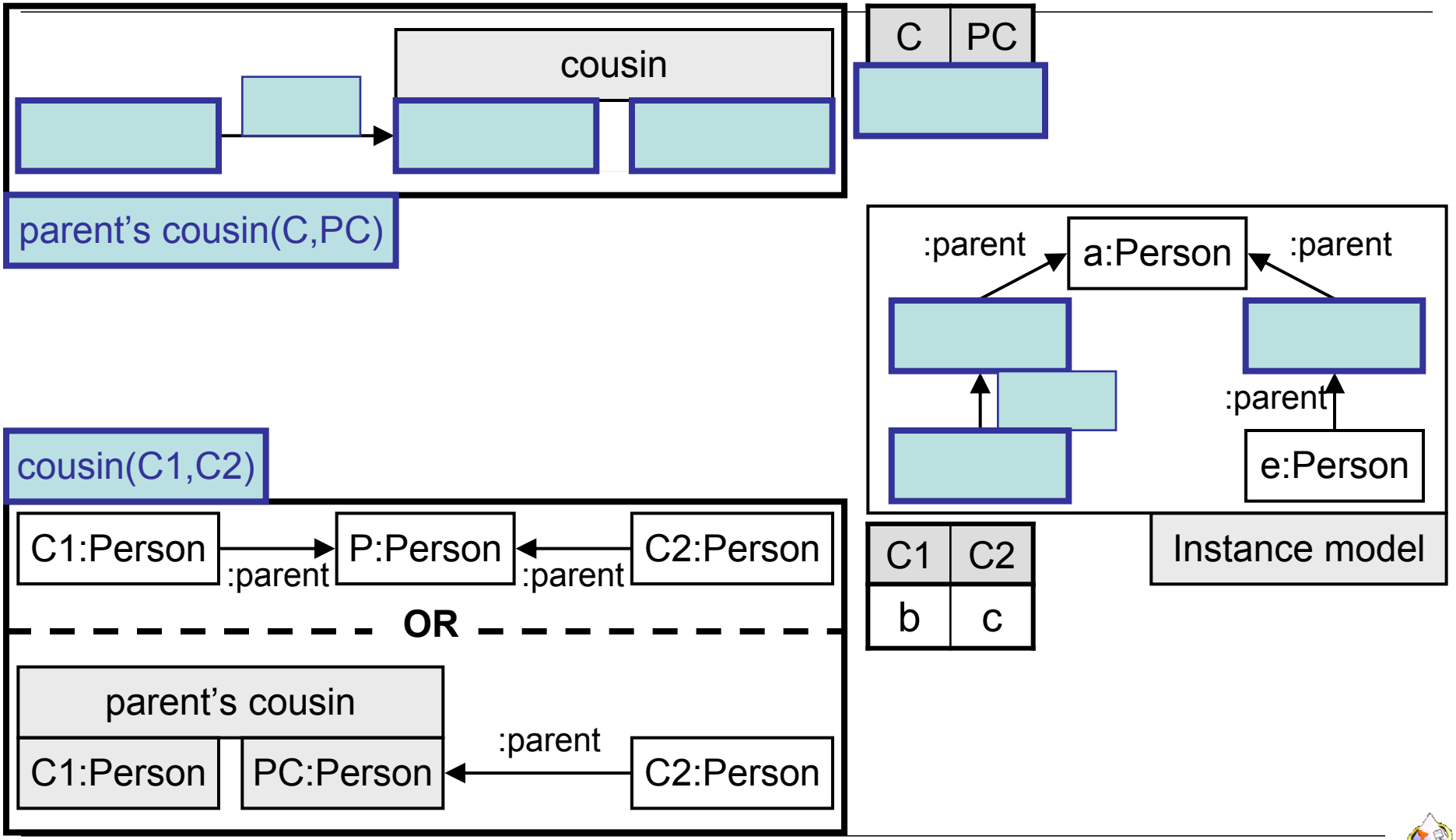
Rekursive Techniken in Graphtransformations-ansätzen



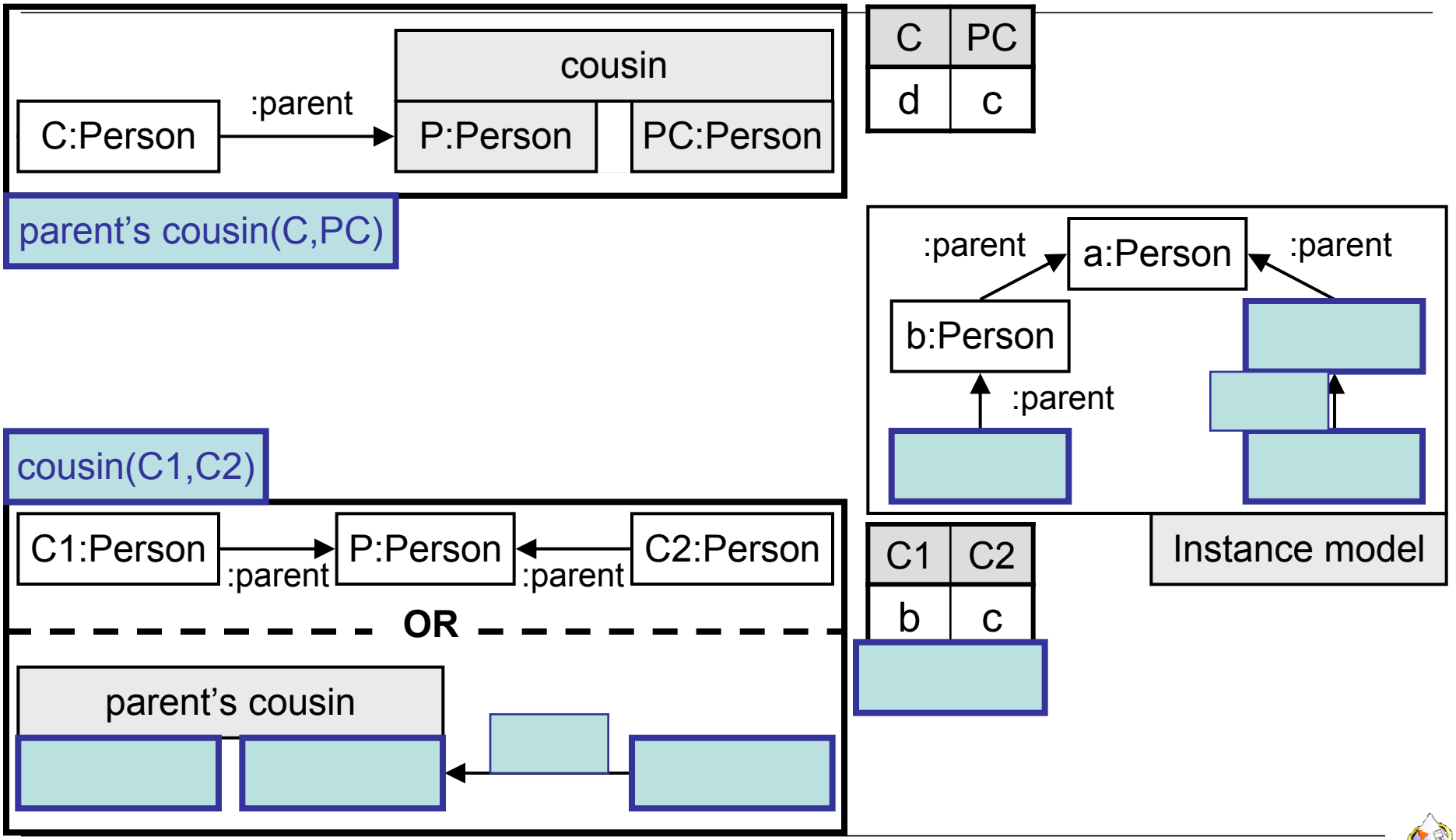
Rekursive Techniken in Graphtransformations-ansätzen



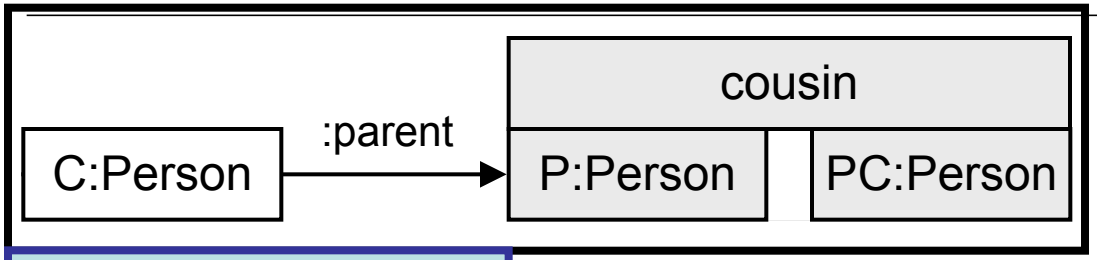
Rekursive Techniken in Graphtransformations-ansätzen



Rekursive Techniken in Graphtransformations-ansätzen

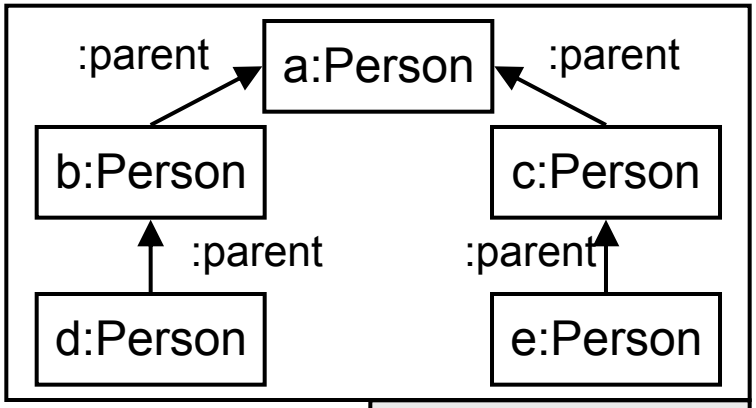


Rekursive Techniken in Graphtransformations-ansätzen



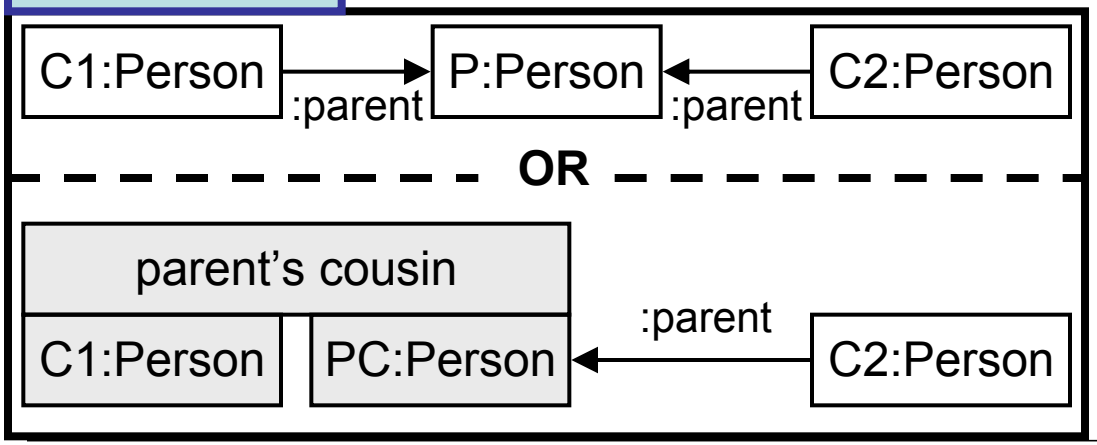
C	PC
d	c
e	b

parent's cousin(C,PC)



Instance model

cousin(C1,C2)



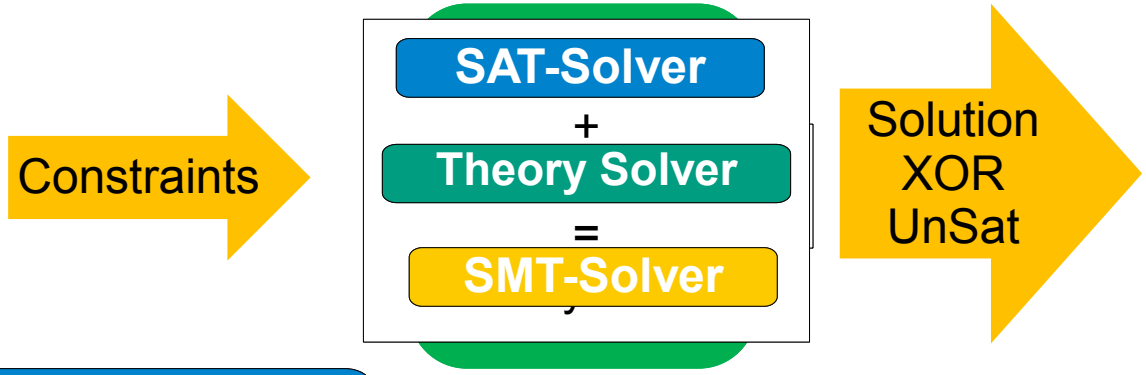
C1	C2
b	c
d	e
c	b
e	d

TECHNISCHE UNIVERSITÄT DARMSTADT

Satisfiability Modulo Theorie Solver

Frederik Deckwerth

Satisfiability Modulo Theories (SMT)



SAT-Solver

- $(a \vee b) \wedge (\bar{a} \vee c)$

Linear-Algebra Solver

- $i + n < z$

SMT-Solver

- $(a \vee b) \wedge (\bar{a} \vee c) \wedge (i + n < z)$

Satisfiability Modulo Theories (SMT)

TECHNISCHE UNIVERSITÄT DARMSTADT

Workhorse for many Tasks in Software Engeneering

Scheduling / Planning

- Find valid schedule

Test Case Generation

- Find execution trace to test a property

Software Verification

- Do the defined properties always hold

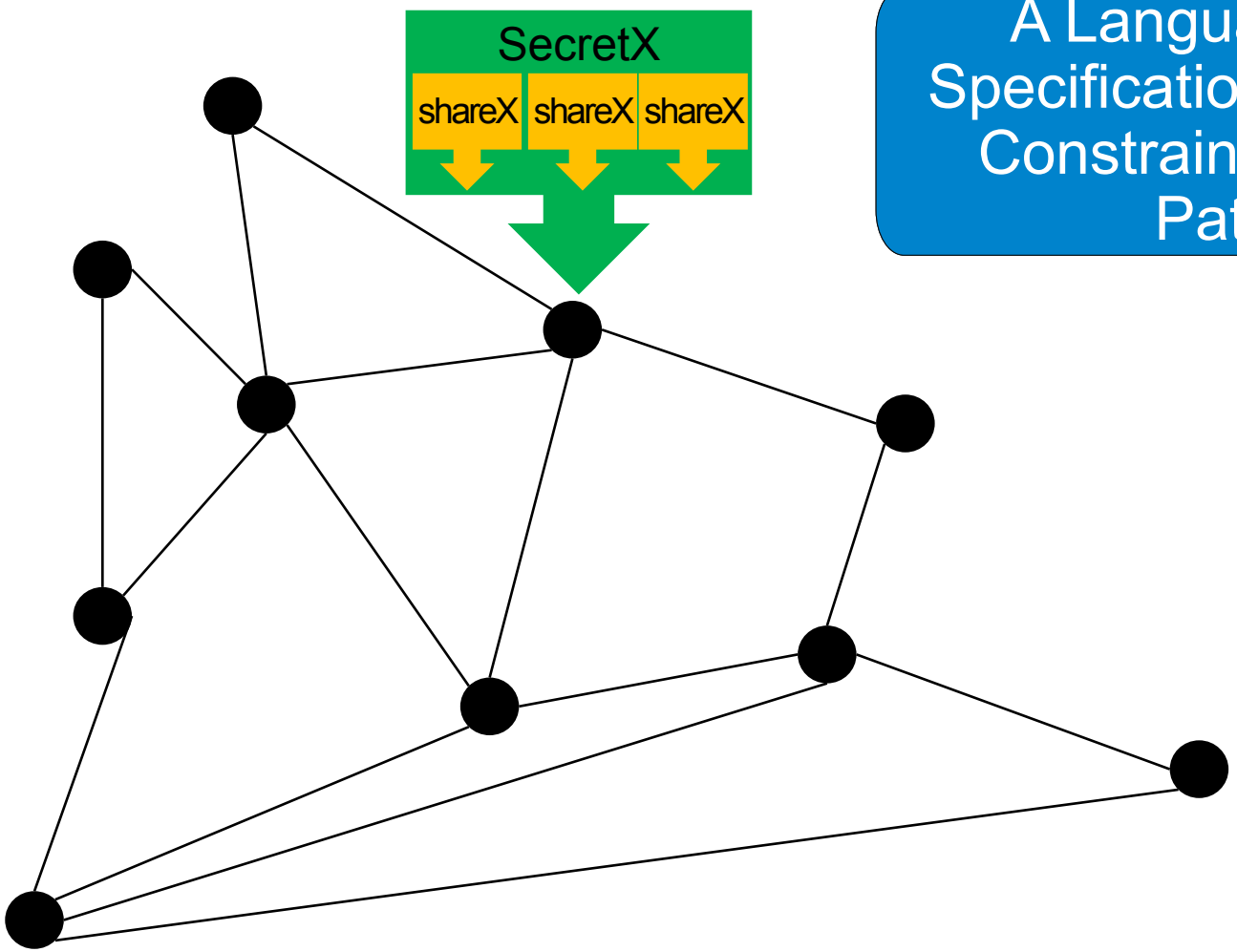
■ ■ ■

TECHNISCHE UNIVERSITÄT DARMSTADT

Visuelle Spezifikation von strukturellen und temporalen Eigenschaften

Frederik Deckwerth

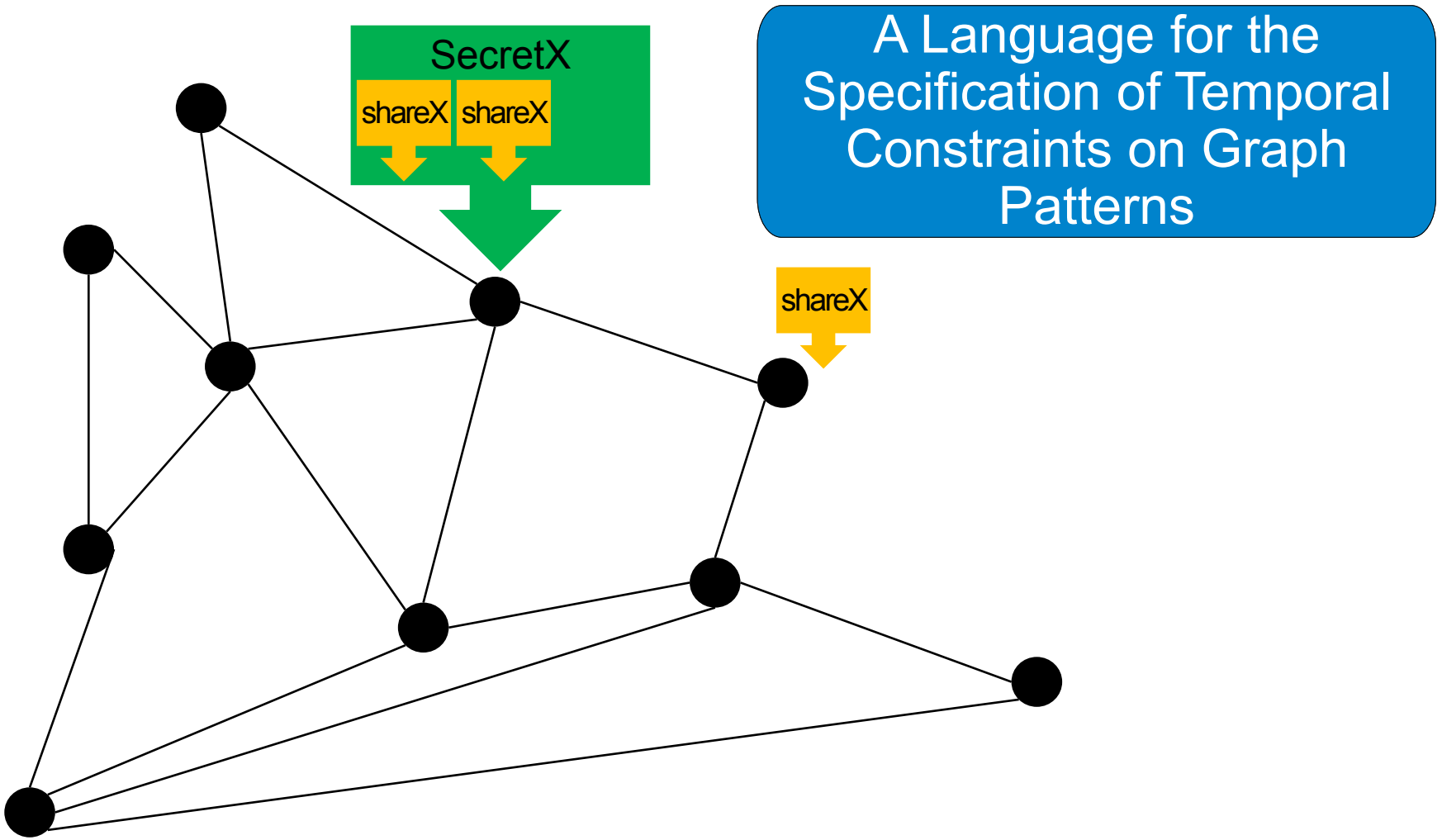
Visual Specification of Structural and Temporal Properties



A Language for the Specification of Temporal Constraints on Graph Patterns

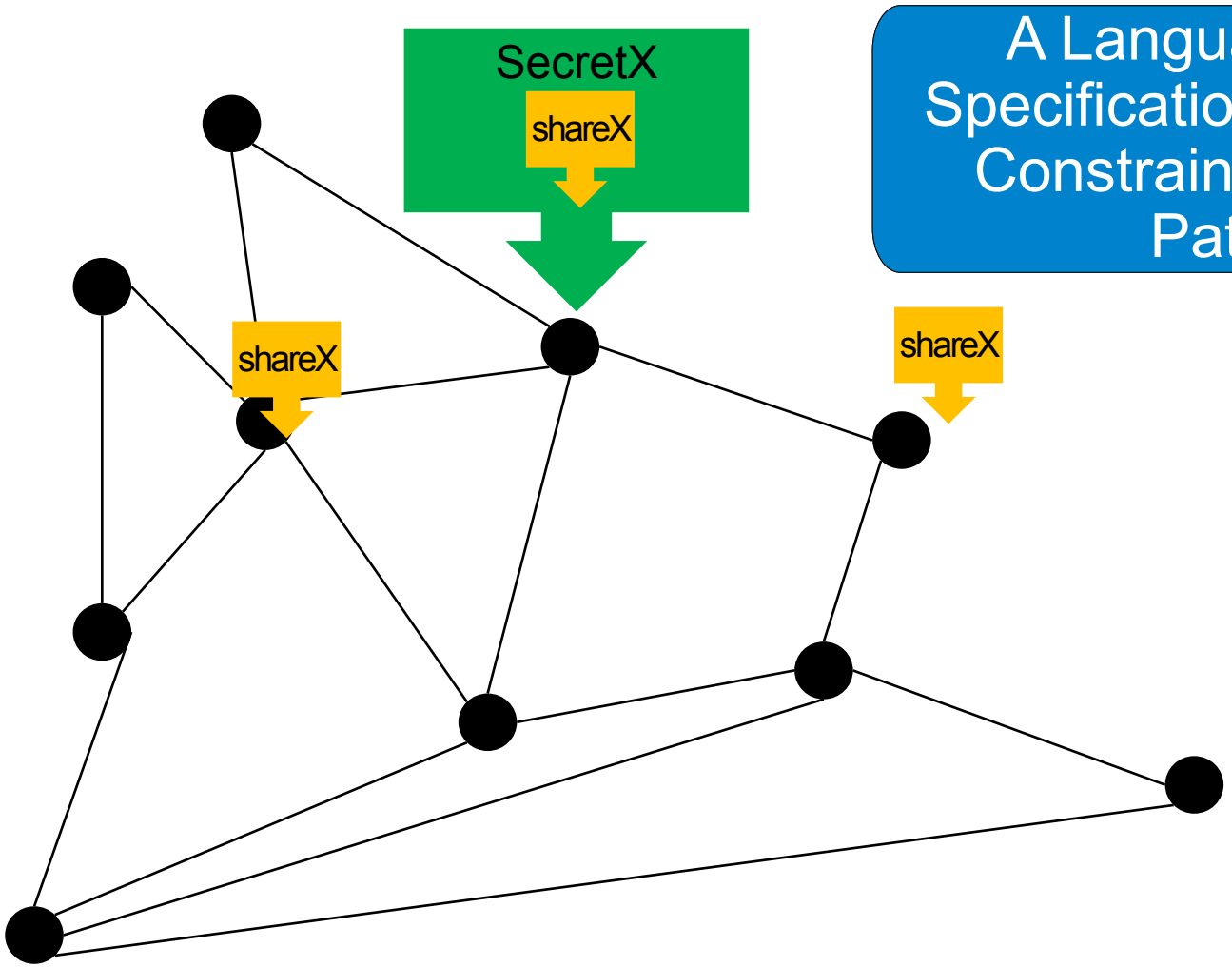
Visual Specification of Structural and Temporal Properties

TECHNISCHE UNIVERSITÄT DARMSTADT



Visual Specification of Structural and Temporal Properties

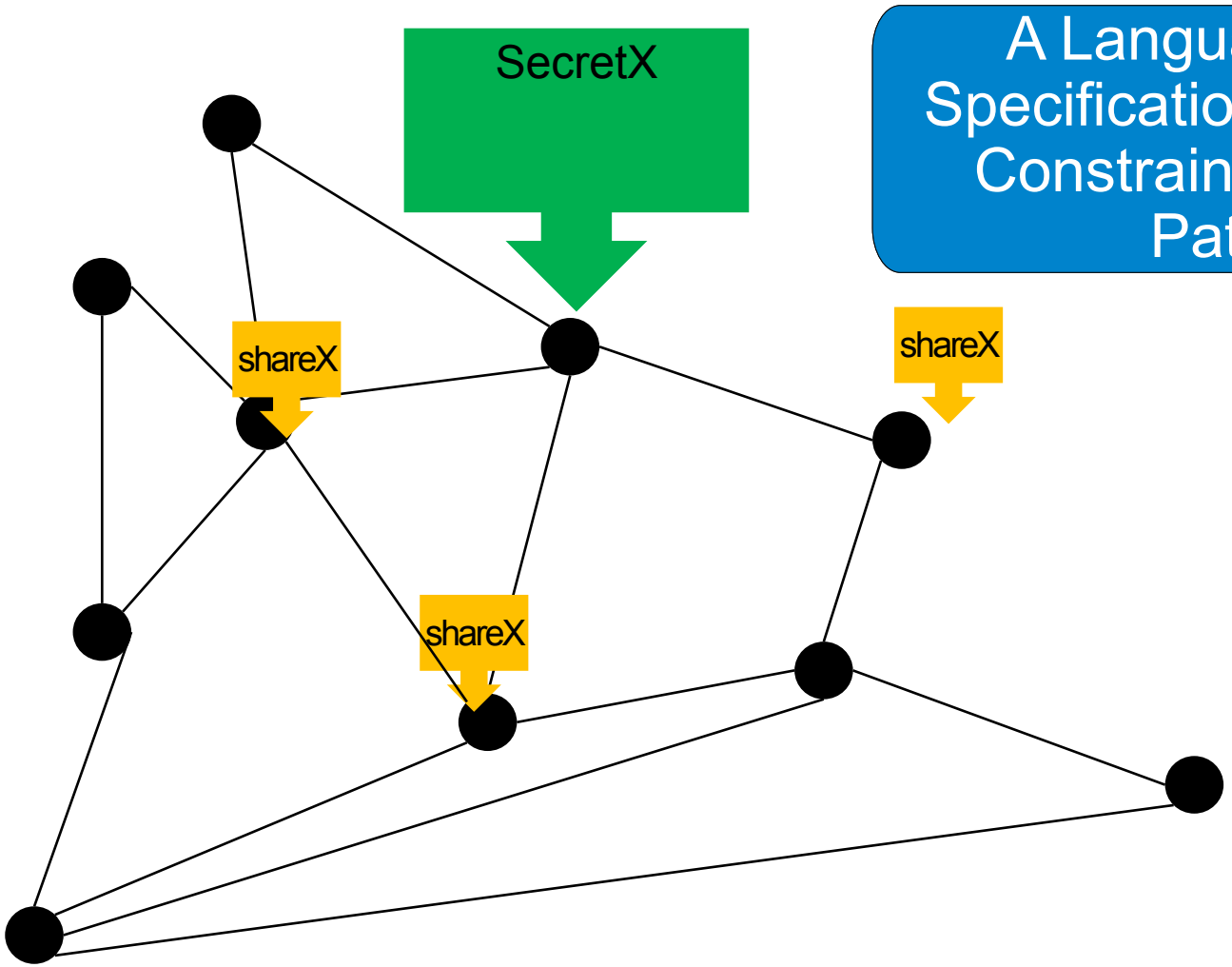
TECHNISCHE UNIVERSITÄT DARMSTADT



A Language for the Specification of Temporal Constraints on Graph Patterns

Visual Specification of Structural and Temporal Properties

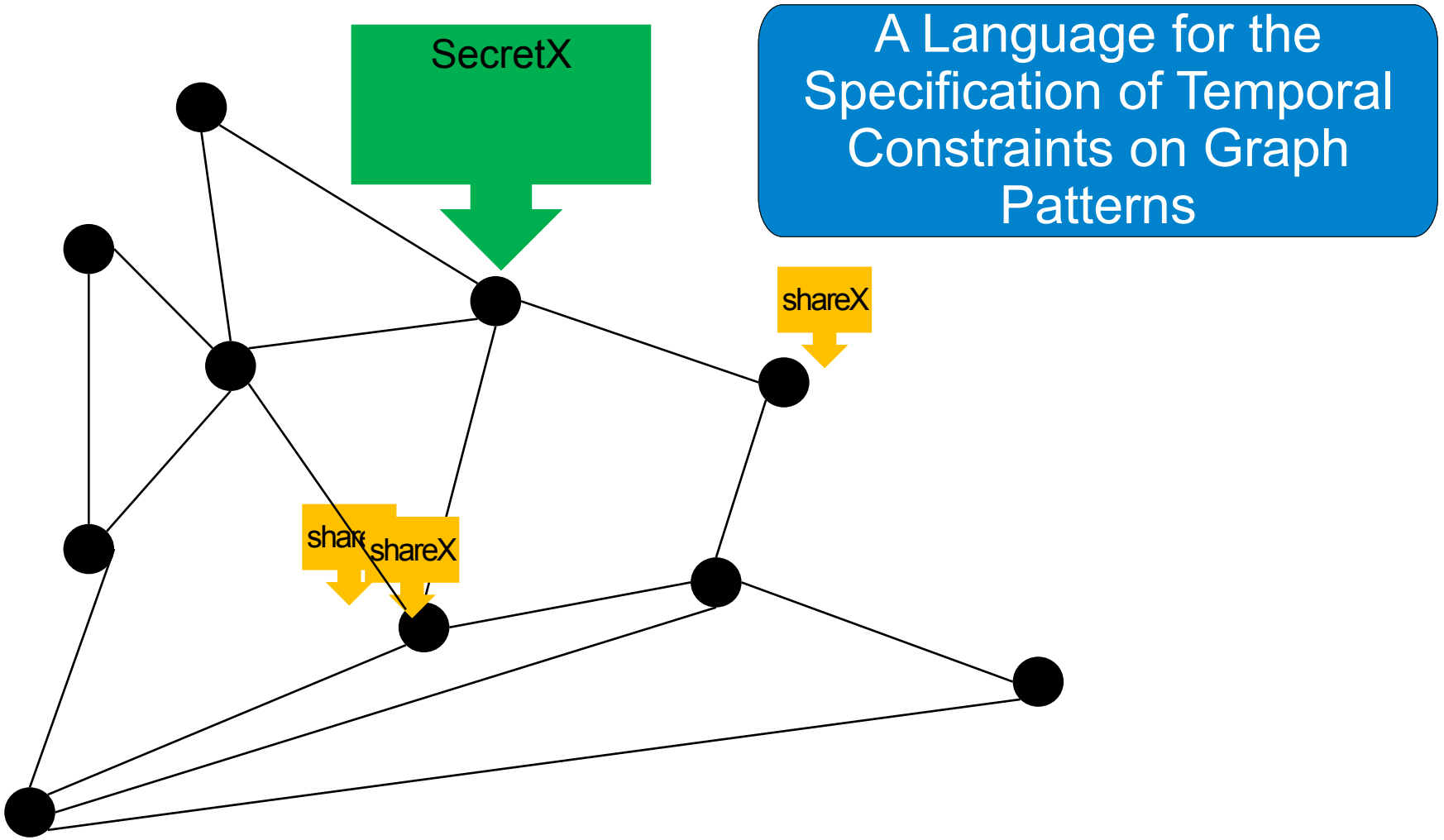
TECHNISCHE UNIVERSITÄT DARMSTADT



A Language for the Specification of Temporal Constraints on Graph Patterns

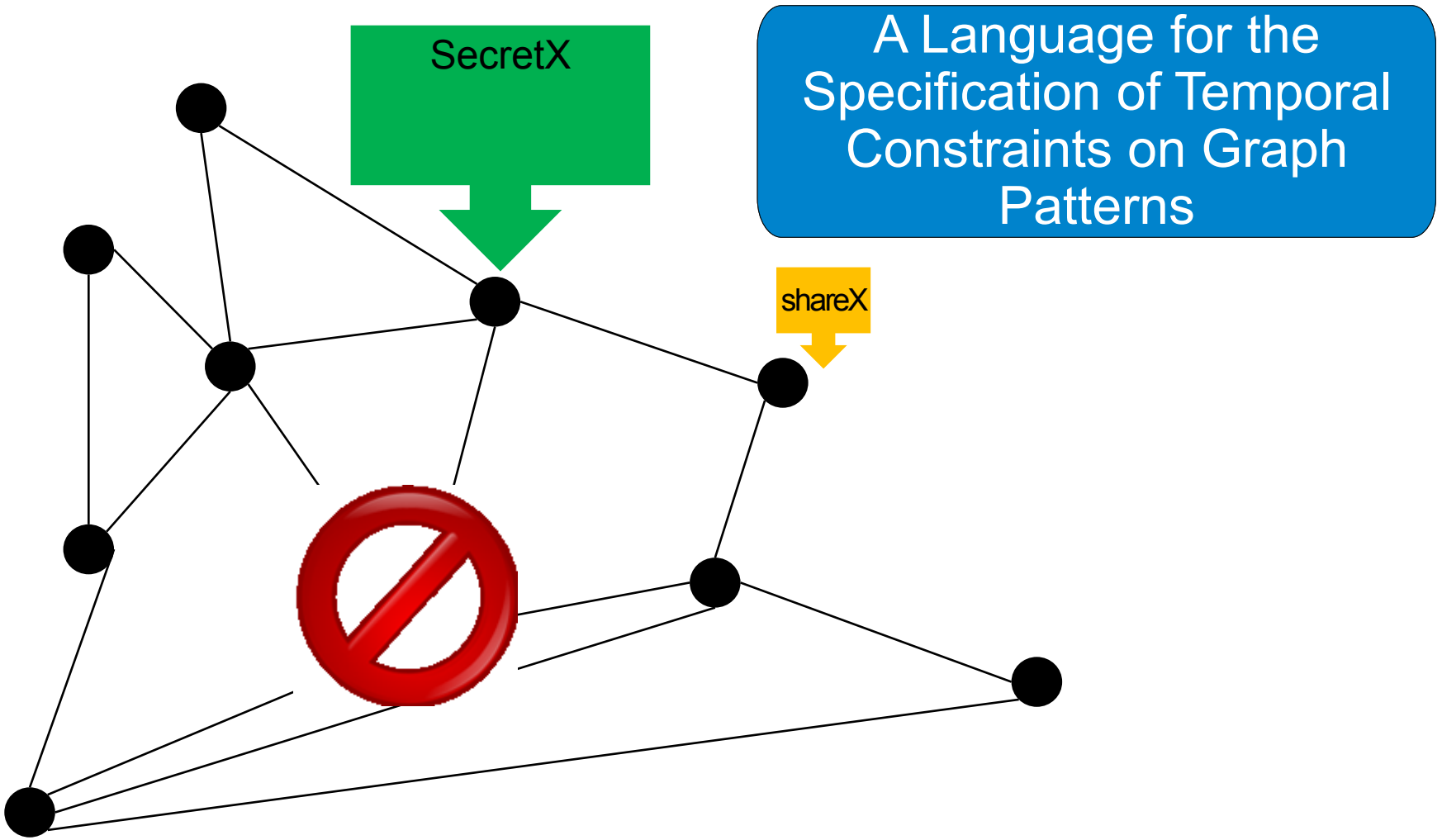
Visual Specification of Structural and Temporal Properties

TECHNISCHE UNIVERSITÄT DARMSTADT



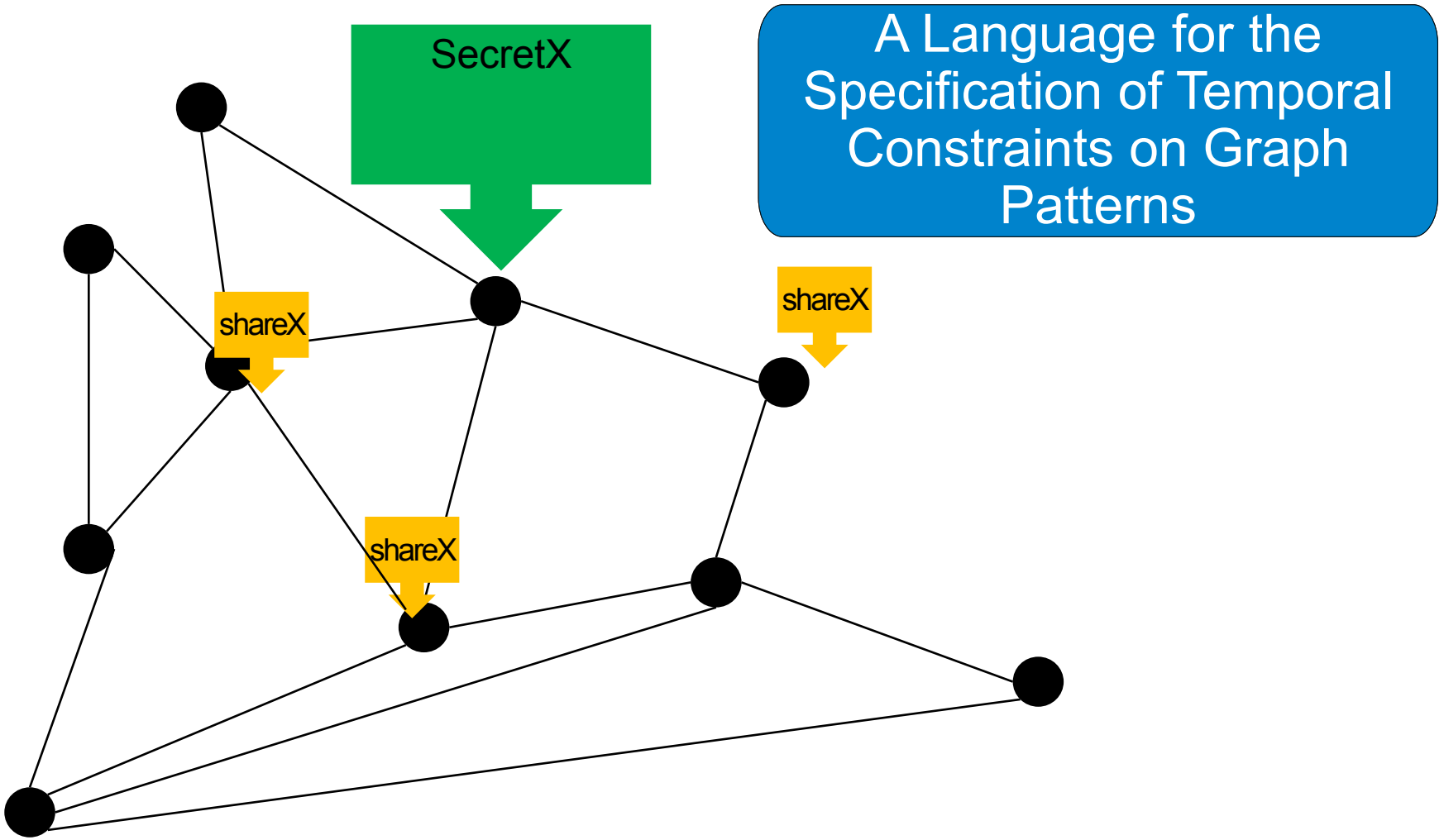
Visual Specification of Structural and Temporal Properties

TECHNISCHE UNIVERSITÄT DARMSTADT



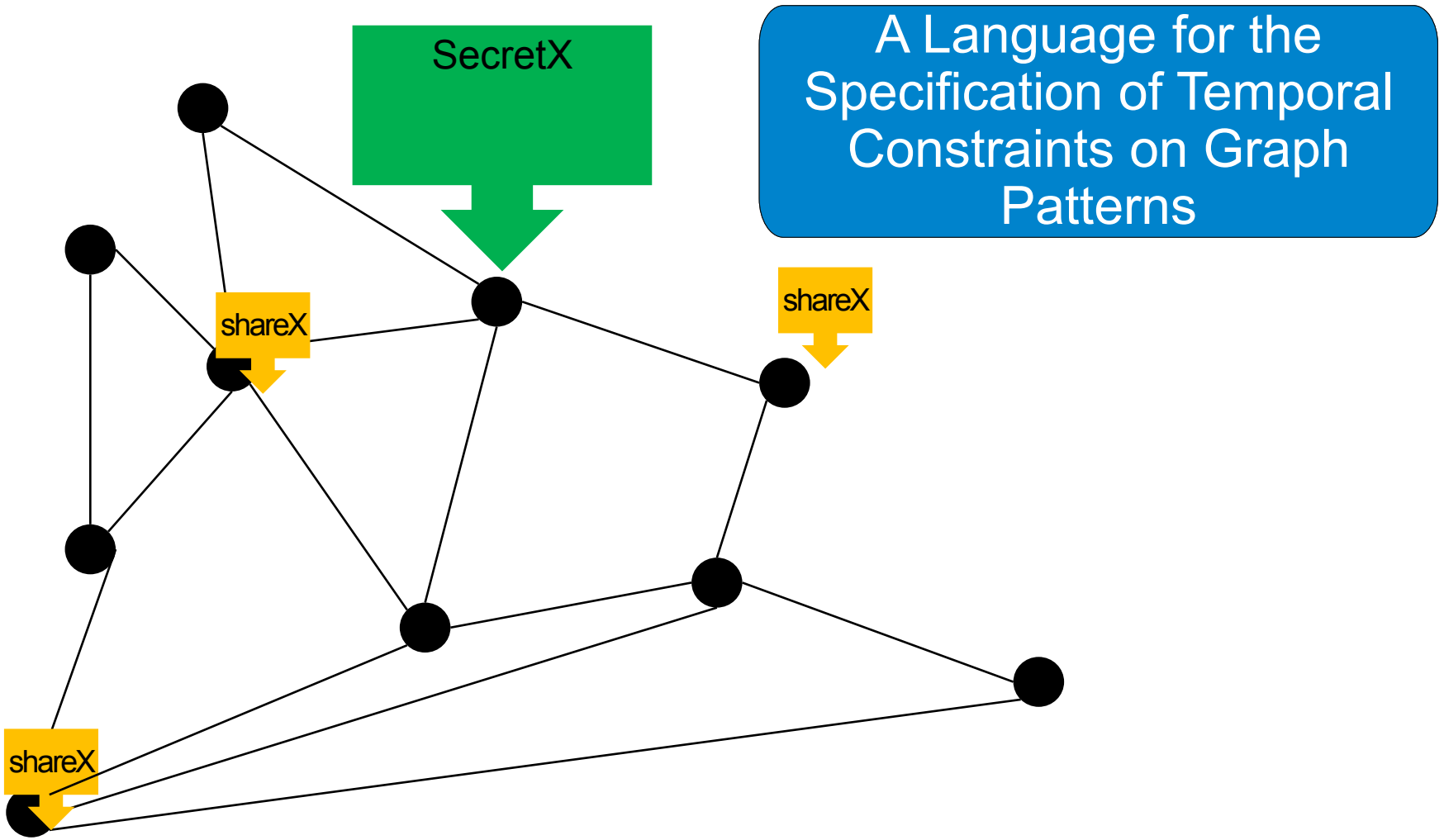
Visual Specification of Structural and Temporal Properties

TECHNISCHE UNIVERSITÄT DARMSTADT



Visual Specification of Structural and Temporal Properties

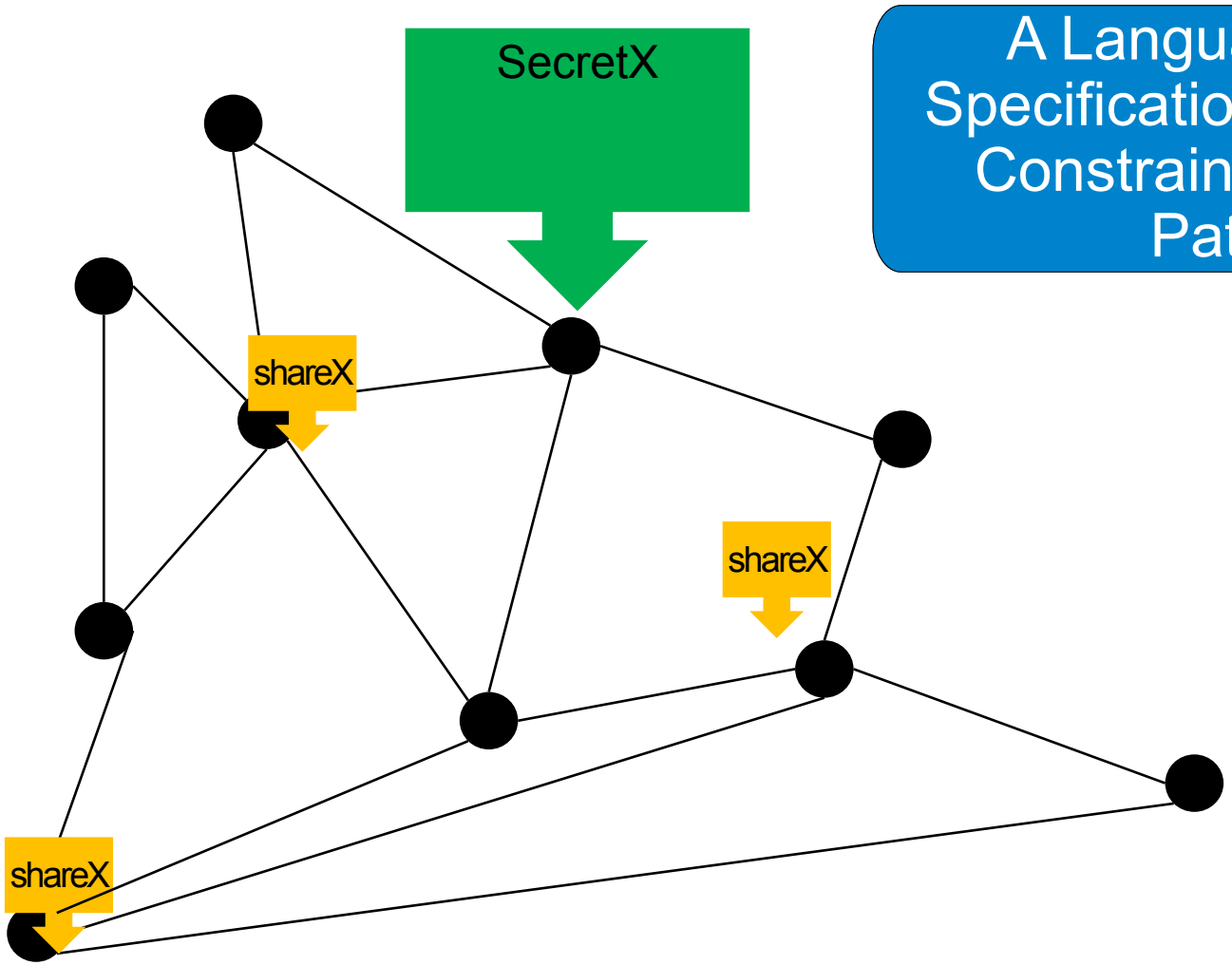
TECHNISCHE UNIVERSITÄT DARMSTADT



A Language for the Specification of Temporal Constraints on Graph Patterns

Visual Specification of Structural and Temporal Properties

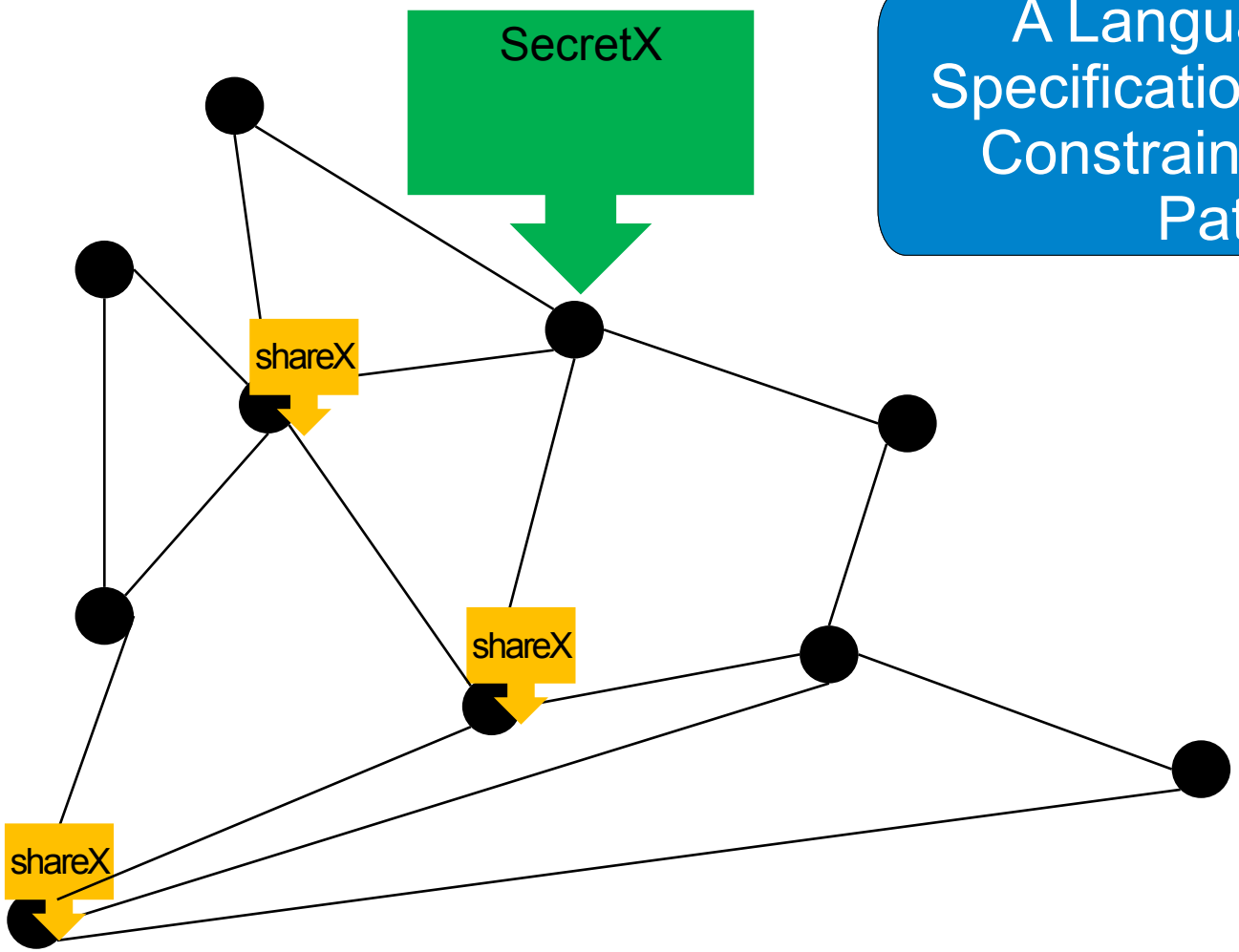
TECHNISCHE UNIVERSITÄT DARMSTADT



A Language for the Specification of Temporal Constraints on Graph Patterns

Visual Specification of Structural and Temporal Properties

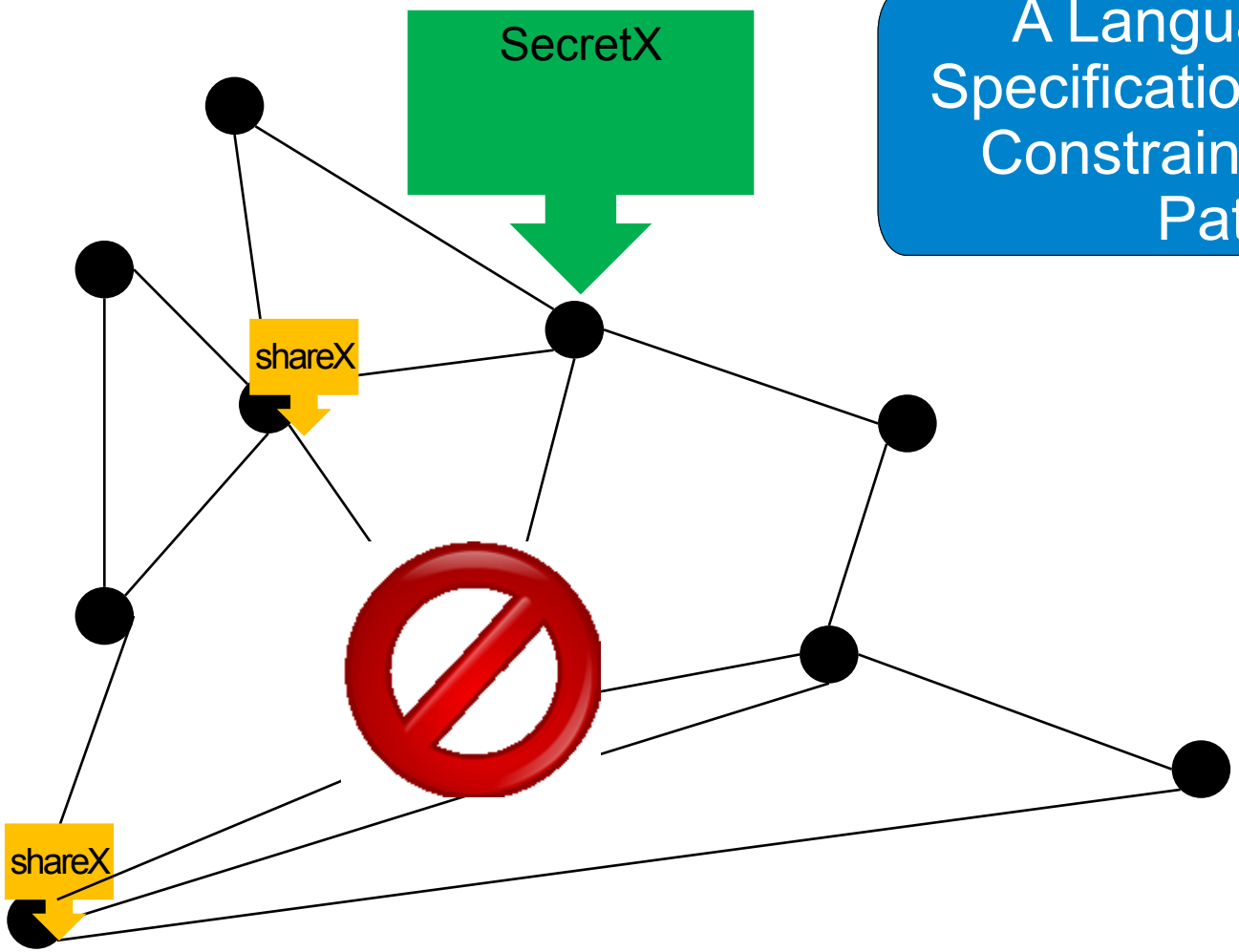
TECHNISCHE UNIVERSITÄT DARMSTADT



A Language for the Specification of Temporal Constraints on Graph Patterns

Visual Specification of Structural and Temporal Properties

TECHNISCHE UNIVERSITÄT DARMSTADT



A Language for the Specification of Temporal Constraints on Graph Patterns

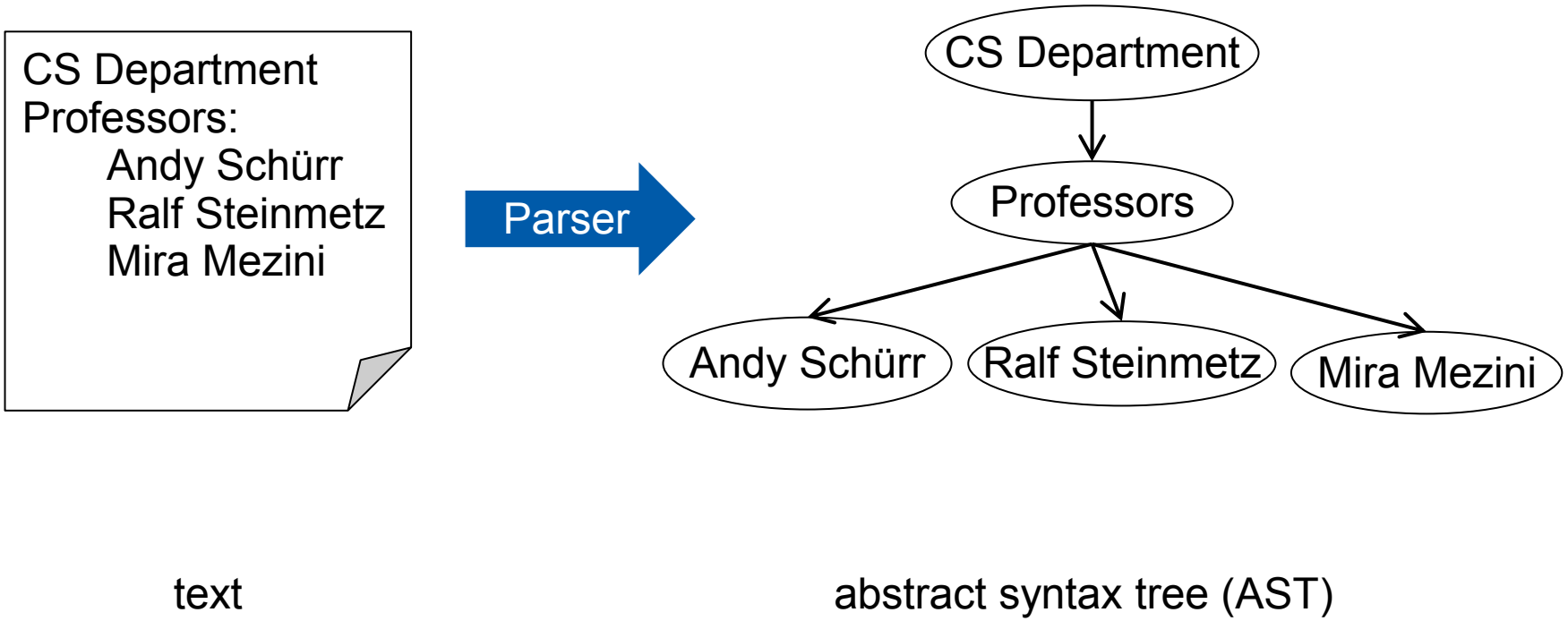
Tree-Diff Algorithms

SST Seminar – SS 14

(Proseminar geeignet)

Erhan Leblebici

Tree-Diff Algorithms



text

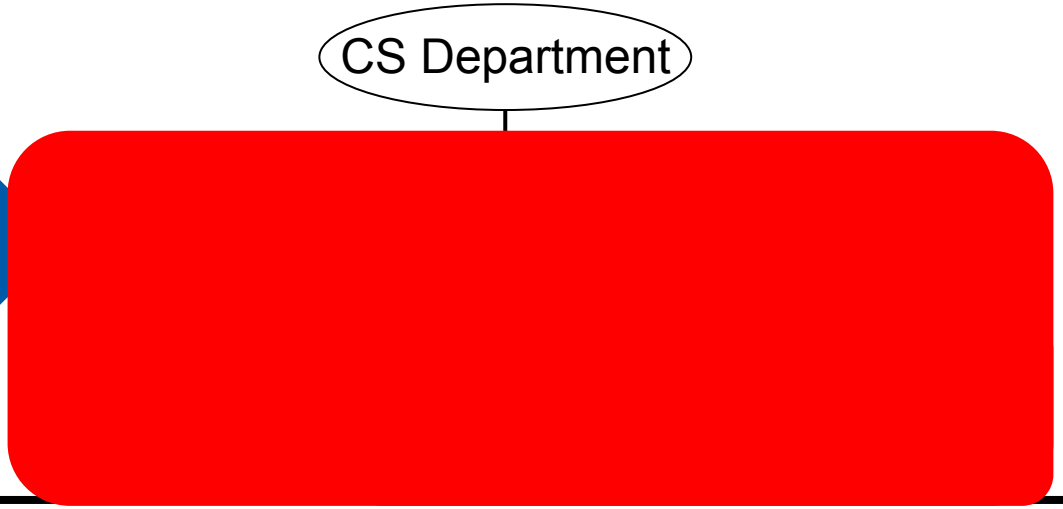
abstract syntax tree (AST)

Tree-Diff Algorithms

TECHNISCHE UNIVERSITÄT DARMSTADT

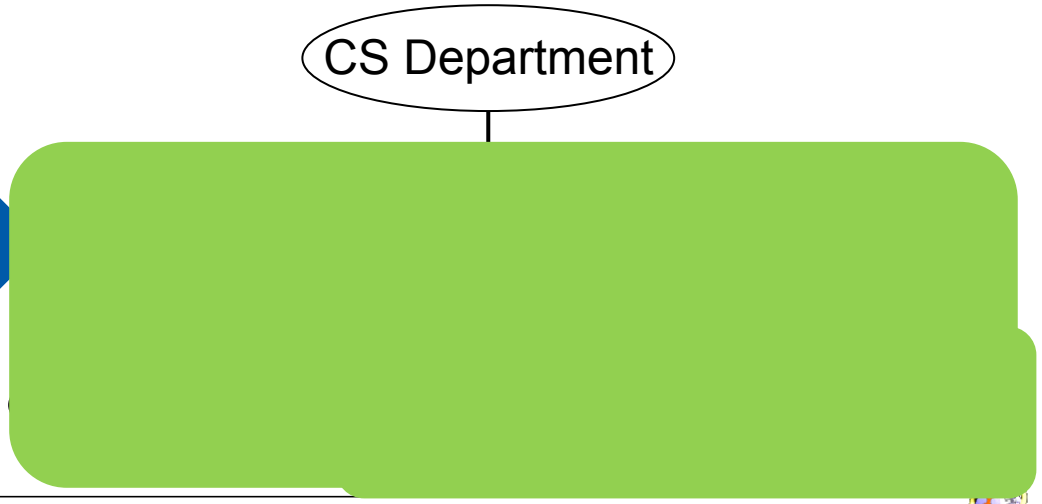
CS Department
Professors:
Andy Schürr
Ralf Steinmetz
Mira Mezini

Parser



CS Department
Profs:
Andy Schürr
Mira Mezini
Ralf Steinmetz

Parser



Anhand Literaturrecherche…

TECHNISCHE UNIVERSITÄT DARMSTADT

- Verschiedene Tree-Diff Ansätze untersuchen
- Stärken und Schwächen identifizieren
- Überblick über vorhandene Implementierungen und Evaluationen liefern

TECHNISCHE UNIVERSITÄT DARMSTADT

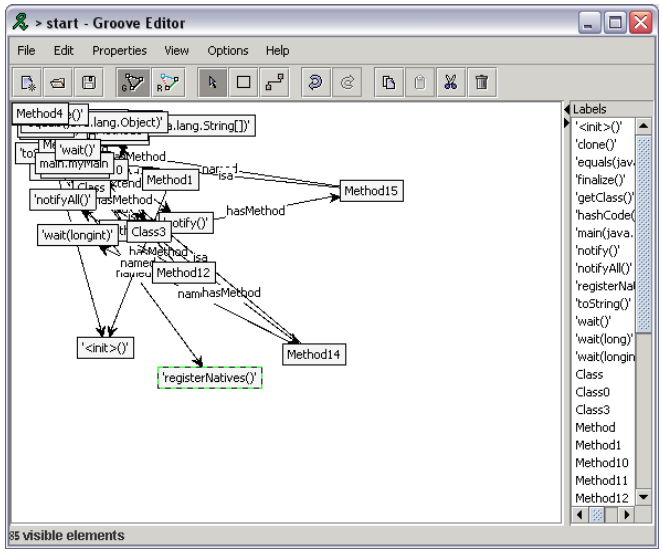
Model-checking using GROOVE

SST Thema – SS 14

(Proseminar geeignet)

Géza Kulcsár

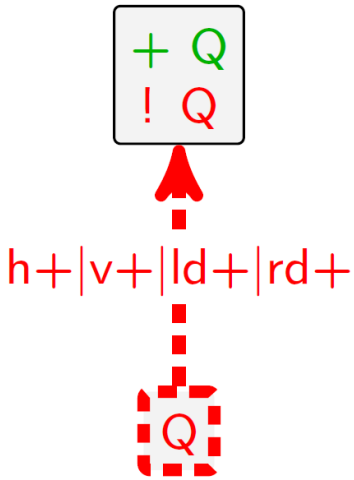
- A general purpose graph transformation tool set including:
  - a simulator: GUI for rules and graphs
  - a generator
  - a model checker

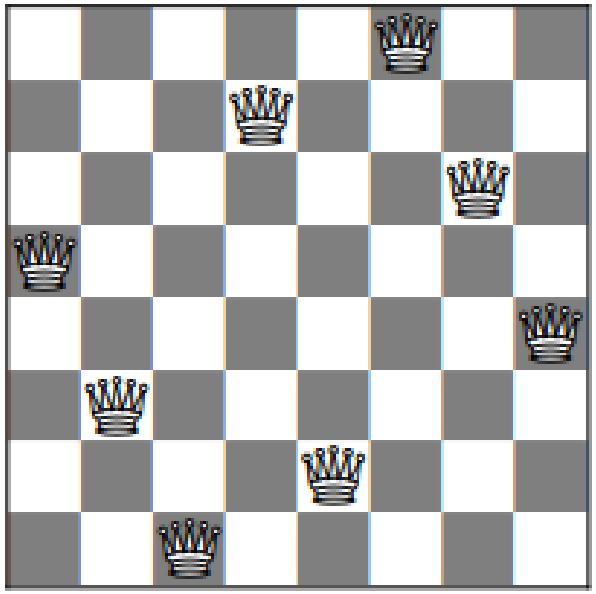


Example: 8-queen problem

TECHNISCHE UNIVERSITÄT DARMSTADT

- Place 8 queens on a chess board so that none of them attacks another
- Can be solved easily using graph transformation





Aim of this work

TECHNISCHE UNIVERSITÄT DARMSTADT

- Get acquainted with model transformation concepts
- Implement any logical puzzles using GROOVE
  - all ideas are welcome
- Verify it using GROOVE model checker

TECHNISCHE UNIVERSITÄT DARMSTADT

Test Case Generation with Model Checking

SST Seminar – SS 14

(Proseminar geeignet)

Johannes Bürdek

Test Case Generation with Model Checking (Proseminar geeignet)

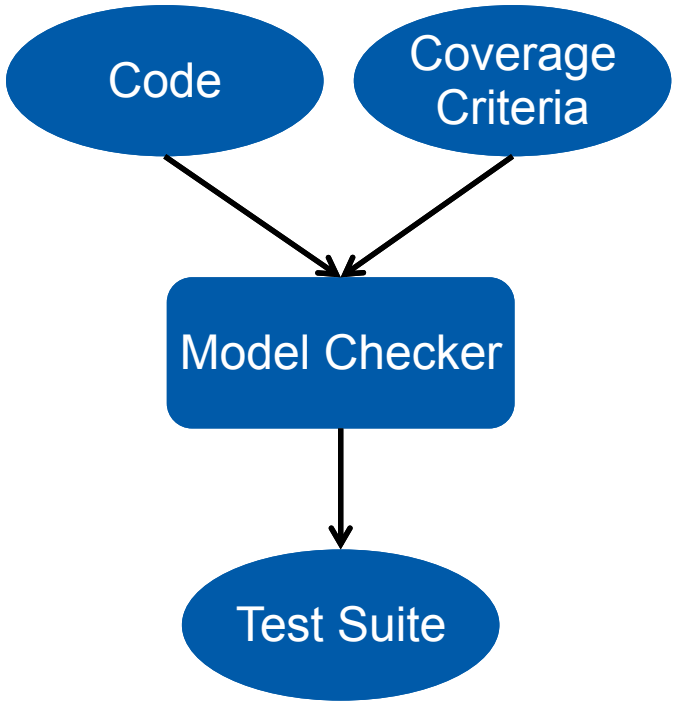
TECHNISCHE UNIVERSITÄT DARMSTADT

- Testen sehr zeit- und kostenintensiv

  => Testfallgenerierung automatisieren

- Technik: Model-Checking

  => Aus Gegenbeispiel folgt ein Testfall



- Aufgabe:
  - Überblick über Techniken zum Generieren von Testfällen mit Hilfe von Model Checkern geben

TECHNISCHE UNIVERSITÄT DARMSTADT

Evolution of Software Product Lines

SST Seminar – SS 14

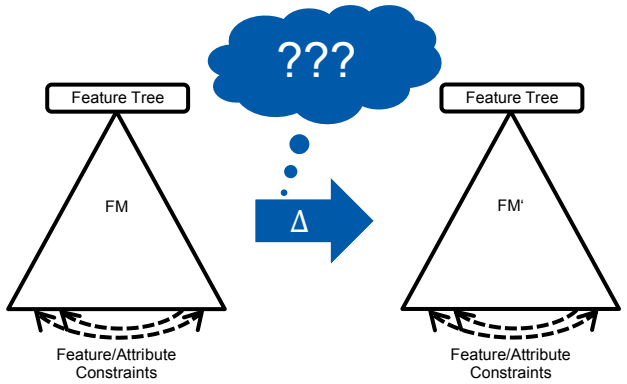
(Proseminar geeignet)

Johannes Bürdek

Evolution of Software Product Lines (Proseminar geeignet)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Software verändert sich über die Zeit
  - Hinzufügen, Löschen, Verändern von Features
  - Änderungen automatisch syntaktisch analysieren
  - Was heißt das Semantisch?



- Problem ist schwerer für Software-Produktlinien (SPL)

- Ansätze zum Kategorisieren von Änderungen

- Aufgabe:
  - Ansätze aufzeigen mit denen Änderungen von SPLs analysiert werden können

TECHNISCHE UNIVERSITÄT DARMSTADT

Testing with Reachability Games

SST Seminar – SS 14

Malte Lochau

Testing with Reachability Games

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation

- Model-based Testing is a crucial Technique for Quality Assurance of Complex Software Systems
- Various Approaches for Automated Derivation of Test Cases from a System Specification that satisfy a test goal
  => Reachability Problem
- Reachability Games provide a Game Theoretic Approach to Automated Test Case Derivation for Systems with Non-deterministic Behaviors and Environments

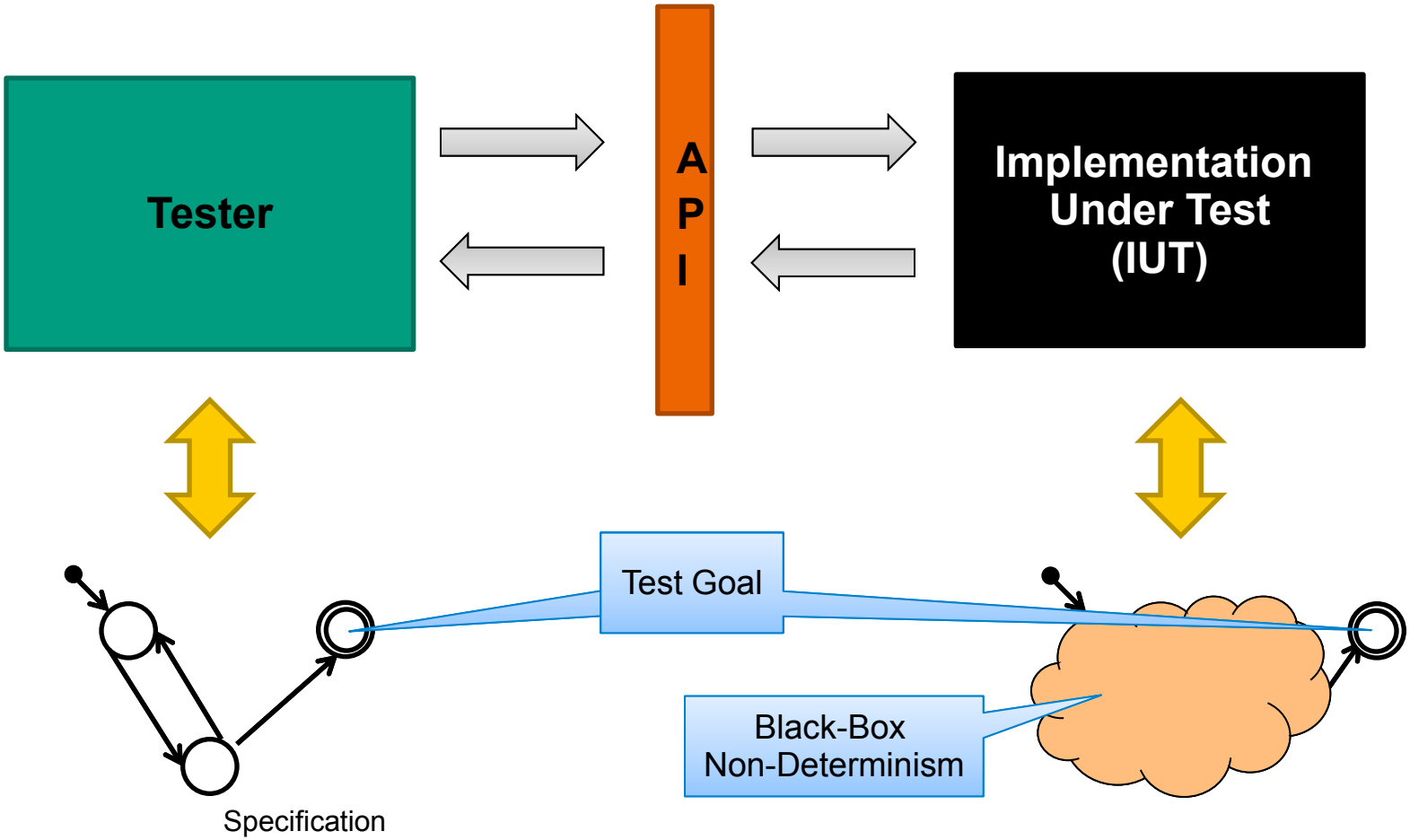
Goal

- Read and Understand Standard Literature on Testing with Reachability Games (english)
- Summarize, Compare and Evaluate Selected Approaches and Tools

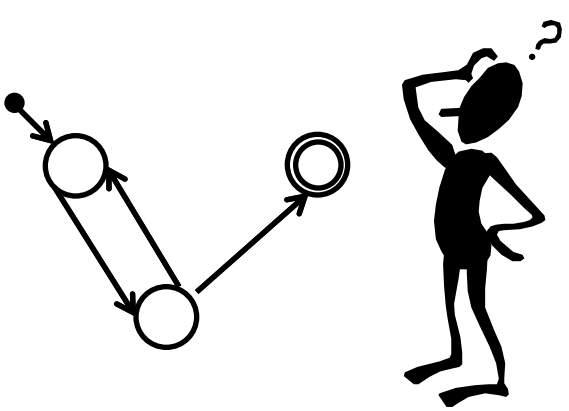
Literature

- J. Y. Halpern*: Computer Science and Game Theory: A Brief Survey.* 2007.
- A. Blass, Y. Gurevich, L. Nachmanson, M. Veanes: *Play to Test.* 2005.
- L. Nachmanson, M. Veanes, W. Schulte, N. Tillmann, W. Grieskamp: *Optimal Strategies for Testing Nondeterministic Systems.* 2004.
- R. Alur, C. Courcoubetis, M. Yannakakis: *Distinguishing tests for nondeterministic and probabilistic machines.* 1995.
- N. Kukreja, W. G. J. Halfond, M. Tambe: *Randomizing Regression Tests Using Game Theory.*

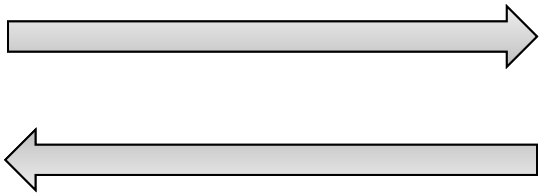
Background: Model-based Testing

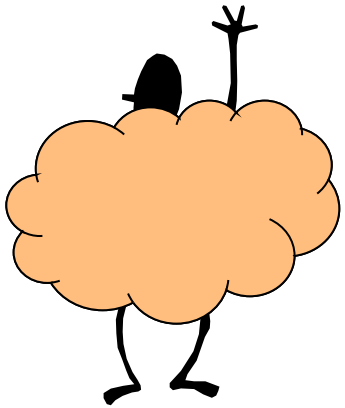


Reachability Games for Testing



Tester





IUT

Player 1: Tester

- Find Input Strategy to reach Test Goal

Player 2: IU

- Makes Random R
  with Probability Distribution

Themenauswahl

(jetzt geht's los)

Wie geht es weiter…!?

TECHNISCHE UNIVERSITÄT DARMSTADT

- Unsere Aufgaben:
  - Wir verteilen die Themen schnellstmöglich auf die Interessenten
  - Geben das Ergebnis auf bekannt (→ Homepage, E-Mail)
  - Bereiten alles vor, damit anschließend die Bearbeitung unmittelbar starten kann

- Eure Aufgaben:
  - Warten auf Ergebnisse der Zuteilung
  - Anschließend meldet ihr euch bitte unmittelbar beim Betreuer Termin für ein erstes, persönliches Treffen
  - Anschließend: Start Einarbeitung → Zeitplan beachten

Noch Fragen?