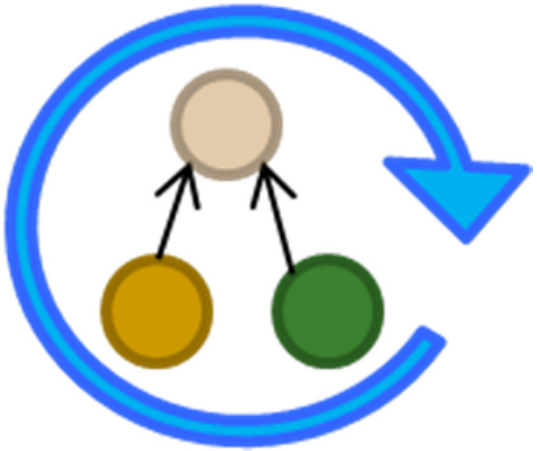
Ontologie-Management

Kapitel 7: Erweiterte Verfahren

Dr. Michael Hartung

Wintersemester 2012/13

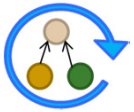
Universität Leipzig
Institut für Informatik
http://dbs.uni-leipzig.de



Inhalt

- **Erkennung (in)stabiler Ontologieregionen**
  - Motivation / Problematik
  - Ontologieregion und zugehörige Metriken
  - Algorithmus
  - Anwendung und Evaluierung

- **Merging von Ontologien**
  - Ontology Merging Prozess
  - Arten von Merge
  - Algorithmus



Entwicklung großer Ontologien

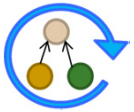
- **Große Ontologien**
  - > 10.000 Konzepte: GO, NCI Thesaurus, …
  - Kollaborative Entwicklung: „einer kann nicht alles“
  - Jeder trägt zu Teilen bei, indem seine Expertise liegt
  - Konsortium legen Designziele fest, z.B. Finalisieren eines Gebietes bis zum Ende des Jahres
- **Probleme**
  - Anwender, Entwickler möchten sich über Fortgang informieren
  - Zeitaufwendig, manuelles Vorgehen inakzeptabel

Änderungen zwischen Ontologieversionen

- **Lineare Folge veröffentlichter Versionen**
  - $O_1, \ldots, O_{j-1}, O_j, O_{j+1}, \ldots, O_n$
- **Mögliche Änderungen**
  - Basis-Änderungstypen: *add*, *del*, *upd*
  - Elemente die sich ändern können: *Konzepte*, *Beziehungen*, *Attribute*

concept		relationship		attribute		
add	*del*	*add*	*del*	*add*	*del*	*upd*

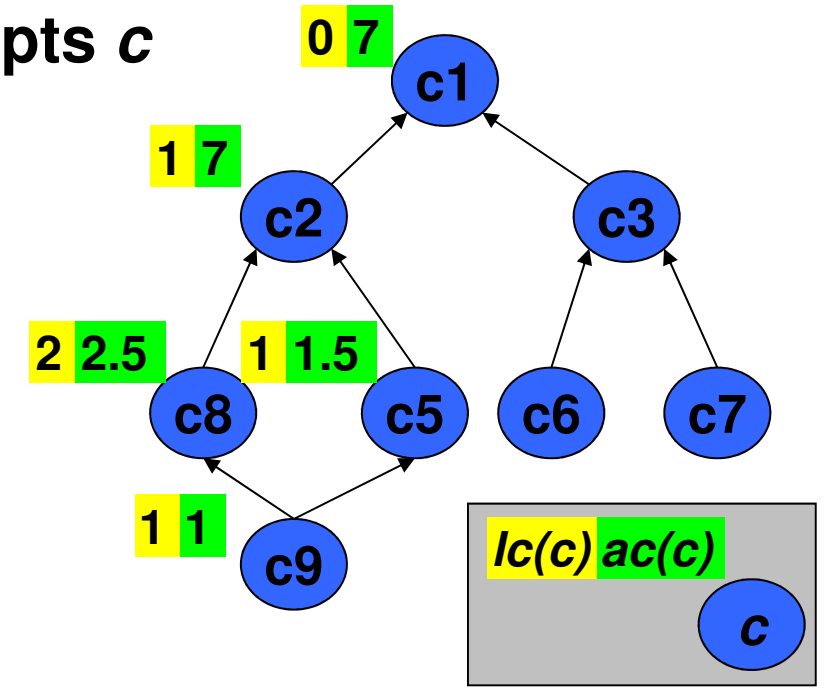
- **Beispiele**
  - Einfügen eines Konzepts: *addConcept*(GO:0015075)
  - Beziehung löschen: *delRel*(GO:0015075, is_a, GO:0005215)
  - Attribute update: *updAtt*(GO:0015075, obsolete, ‘false’, ‘true’)



Änderungskosten

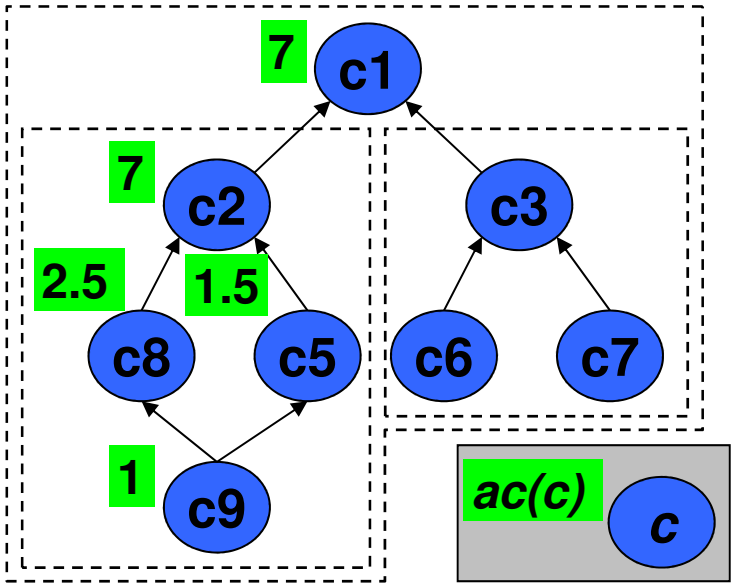
- **Kosten für Ontologieänderungen**
  - Angabe des Einflusses auf die Ontologie
    change → impactValue
  - Beispiel: *delConcept* → 2, *addConcept* → 1

- **Kosten eines Ontologiekonzepts *c***
  - Lokale Kosten ***lc(c)***
    - Änderungen mit direktem Einfluss auf ***c***
  - Aggregierte Kosten ***ac(c)***
    - Änderungen in den is_a Nachfolgern von ***c***

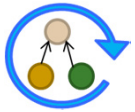


Regionen und zugehörige Metriken

- **Ontologieregion *OR***
  - Teilgraph einer Ontologie mit Wurzelkonzept ***rc***
  - Umfasst alle Konzepte im *is_a* Subgraphen von ***rc***
- **Metriken zur Bewertung**
  - Ziel: Änderungsintensität bewerten
  - Verschiedene Aspekte
    - Absolute / relative Größe
    - Absolute Änderungskosten
    - Durchschnittl. Änderungskosten
    - Kombinationen möglich



region	*abs_size*	*rel_size*	*abs_costs*	*avg_costs*
c1	8	8/8=1	7	7/8=0.875
c2	4	4/8=0.5	7	7/4=1.75
c3	3	3/8=0.375	0	0/3=0



Berechnung aggregierter Kosten für zwei Versionen

- **Eingabe:** zwei Ontologieversionen O_{old} und O_{new}, Kostenmodell σ
- **Ausgabe:** O_{new} mit aggregierten Kosten (*ac*)

computeAggregatedCosts **(O_{old}, O_{new}, σ)**

diff (ΔO_{old}-O_{new} := O_{old}, O_{new})

assignLocalCosts (ΔO_{old}-O_{new}, σ, O_{old}, O_{new})

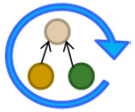
O_{old} := **aggregateCosts** (O_{old})

O_{new} := **aggregateCosts** (O_{new})

transferCosts (O_{old}, O_{new})

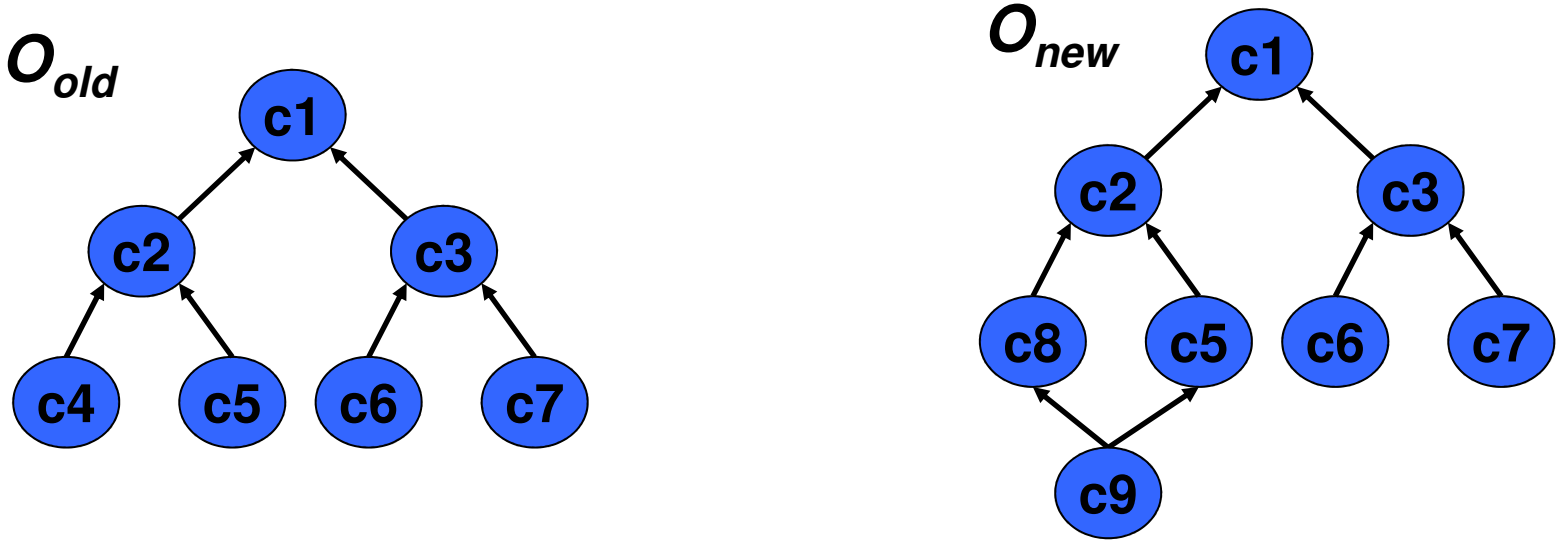
return O_{new}

Hartung, M., Groß, A., Kirsten, T., Rahm, E.: Discovering Evolving Regions in Life Science Ontologies. In Proc. Data Integration in the Life Sciences (DILS), 2010



Änderungserkennung - diff(O_{old},O_{new})

- **Änderungserkennung**
  - Ausnutzung der accession numbers von Konzepten
  - Ergebnis: Menge von ***add***/***del***/***upd*** Änderungen



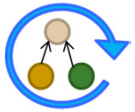
ΔO_{old}-O_{new}:

delConcept(c4)

delRel((c4,c2))

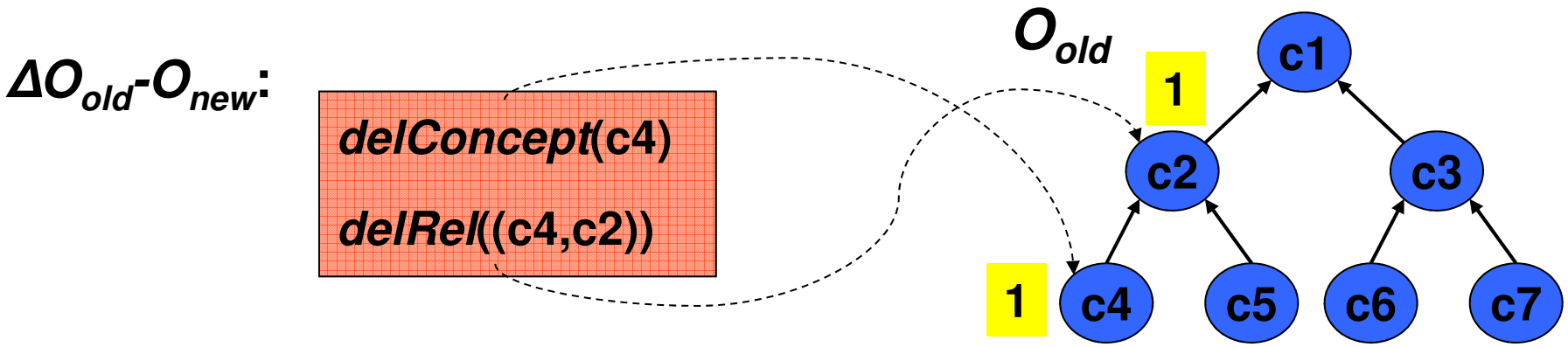
addConcept(c8, c9)

addRel((c8,c2), (c9,c5), (c9,c8))



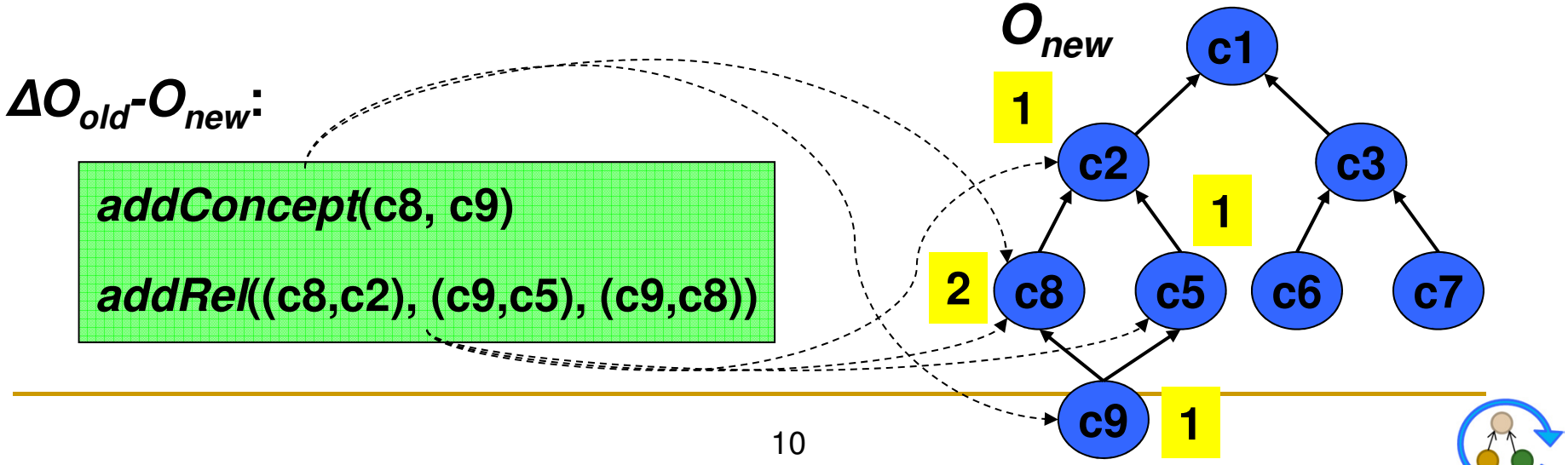
Zuweisung lokaler Kosten - assignLocalCosts(ΔO_{old}-O_{new},σ,O_{old},O_{new})

- Zuweisung basiert auf Kostenmodell und Änderungen
  - ***add***/***upd*** → Erfassung in O_{new}
  - ***del*** → Erfassung in O_{old}
  - Konzept / Attribut-Änderungen → ***lc*** des betreffenden Konzepts
  - Beziehungen → ***lc*** eines oder beider betroffener Konzepte
- Beispiel: Einheitskosten von ***1***, bei Beziehungen nur Target



Zuweisung lokaler Kosten - assignLocalCosts(ΔO_{old}-O_{new},σ,O_{old},O_{new})

- Zuweisung basiert auf Kostenmodell und Änderungen
  - ***add***/***upd*** → Erfassung in O_{new}
  - ***del*** → Erfassung in O_{old}
  - Konzept / Attribut-Änderungen → ***lc*** des betreffenden Konzepts
  - Beziehungen → ***lc*** eines oder beider betroffener Konzepte
- Beispiel: Einheitskosten von ***1***, bei Beziehungen nur Target

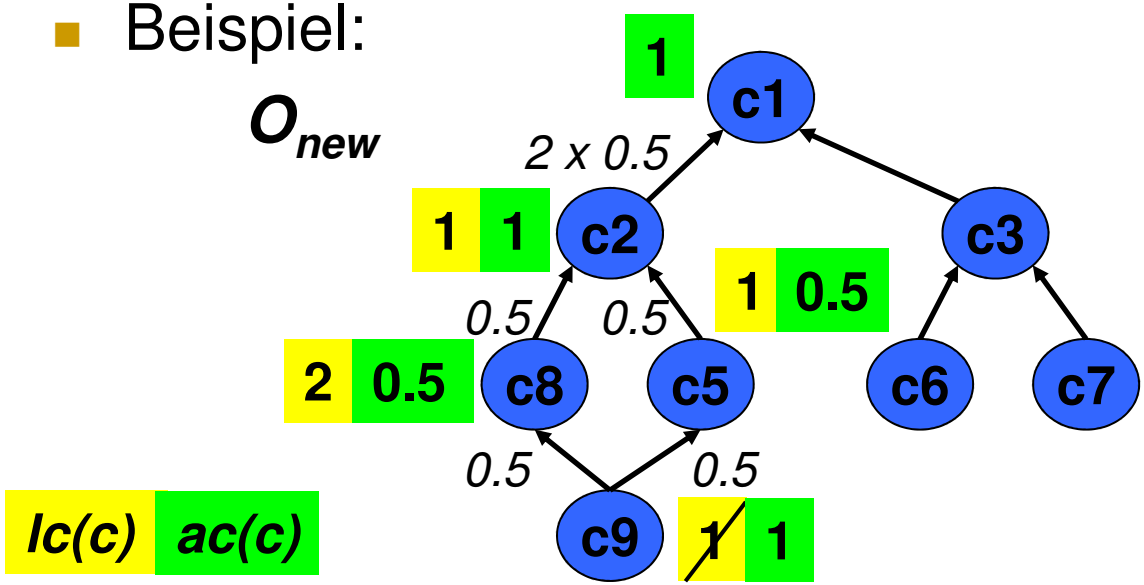


Kostenpropagierung - aggregateCosts(O_v)

- Propagierung lokaler Kosten ***lc*** zur Berechnung von ***ac***
  - **Regel:** „***ac***(***c***) eines Konzepts ***c*** ist die gewichtete Summe der ***ac***‘s aller Kinder plus die eigenen lokalen Kosten ***lc***(***c***)“

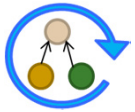
$$ac(c) = \sum_{\text{direct children c' of c}} \frac{ac(c')}{|\ parents(c')\ |} + lc(c)$$

- Beispiel:



propagation of *lc*(c9)

- ***ac*(c9)+=*lc*(c9)**
- ***ac*(c8)+=*lc*(c9)/2**
- ***ac*(c5)+=*lc*(c9)/2**
- ***ac*(c2)+=*lc*(c9)/2+*lc*(c9)/2**
- ***ac*(c1)+=*lc*(c9)/2+*lc*(c9)/2**

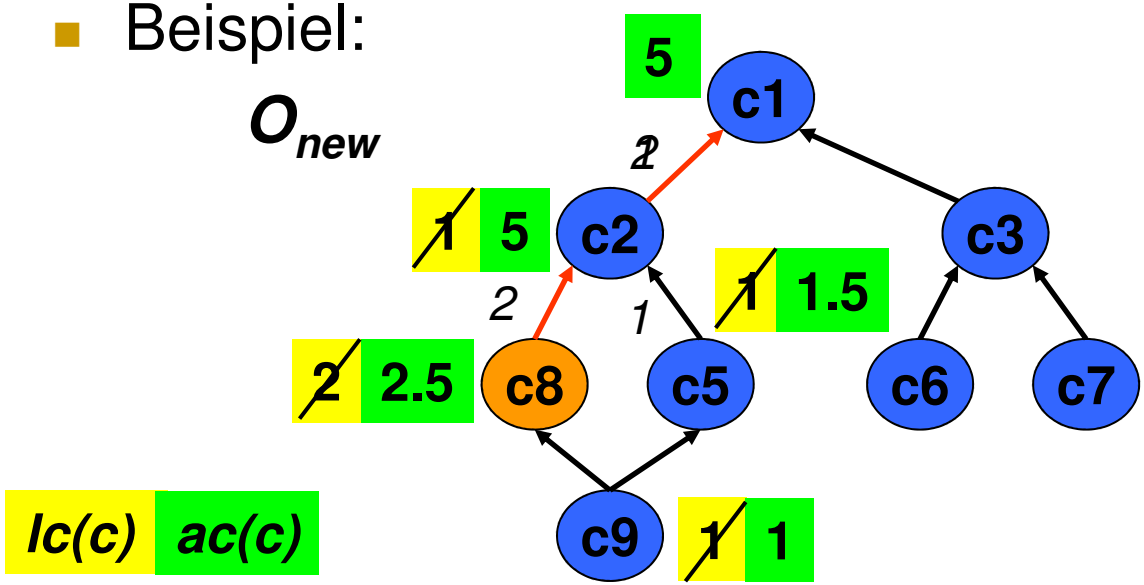


Kostenpropagierung - aggregateCosts(O_v)

- Propagierung lokaler Kosten ***lc*** zur Berechnung von ***ac***
  - **Regel:** „***ac***(***c***) eines Konzepts ***c*** ist die gewichtete Summe der ***ac***‘s aller Kinder plus die eigenen lokalen Kosten ***lc***(***c***)“

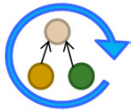
$$ac(c) = \sum_{\text{direct children c' of c}} \frac{ac(c')}{|\ parents(c')\ |} + lc(c)$$

- Beispiel:



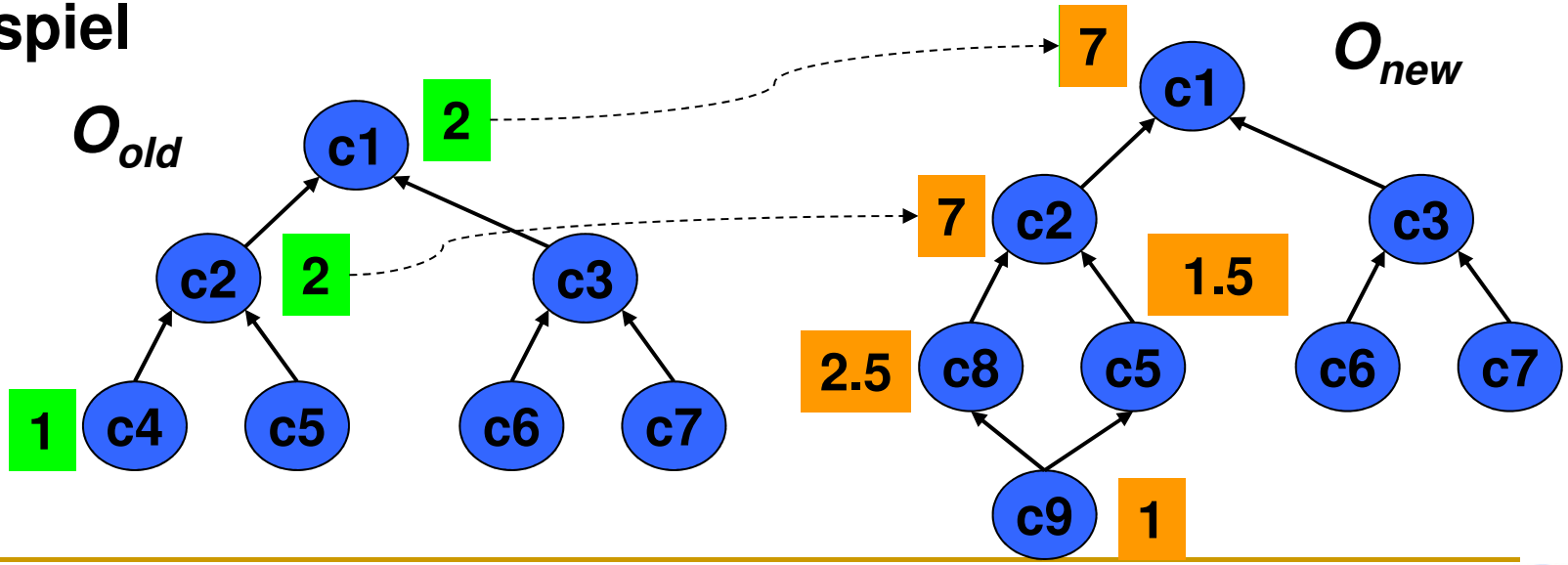
propagation of *lc*(c9)
- ***ac*(c9)+=*lc*(c9)**
- ***ac*(c8)+=*lc*(c9)/2**
- ***ac*(c5)+=*lc*(c9)/2**
- ***ac*(c2)+=*lc*(c9)/2+*lc*(c9)/2**
- ***ac*(c1)+=*lc*(c9)/2+*lc*(c9)/2**

propagation of *lc*(c8)
propagation of *lc*(c5)
propagation of *lc*(c2)



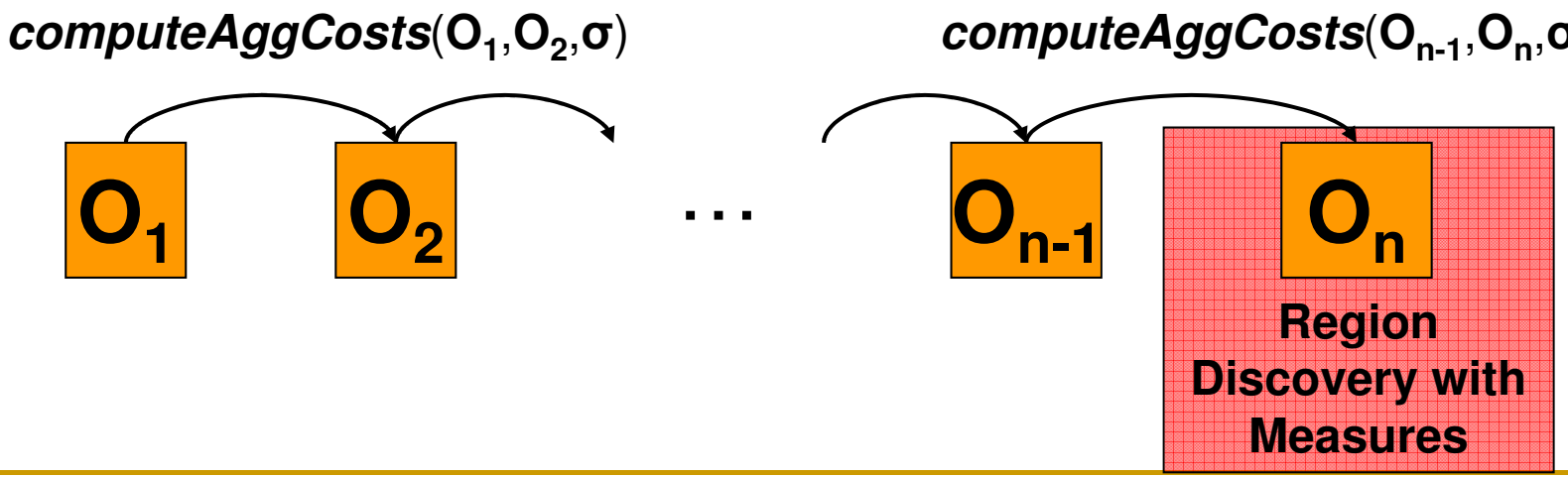
Kostentransfer - transferCosts(O_{old},O_{new})

- **Transfer aggregierter Kosten von alte in neue Version**
  - Erkennung von Regionen auf neuester Version → erfasste aggregierte Kosten in alter Version ebenfalls einbeziehen
    - Kosten von ***del*** Änderungen
  - **Regel:** "aggregierte Kosten gleicher Konzepte werden zusammengefasst"
- **Beispiel**



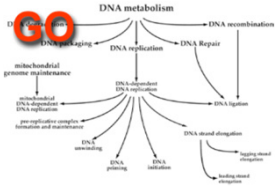
Genereller Algorithmus für *n* Versionen

- Reuse von ***computeAggregatedCosts*** für 2 Versionen
  - Sukzessive Anwendung und Transfer aggregierter Kosten in die neueste Ontologieversion
  - Erkennung von Regionen auf neuester Version
- **Eingabe:** Ontologieversionen $\mathbf{O_1}$, …, $\mathbf{O_n}$, Kostenmodell $\boldsymbol{\sigma}$
- **Ausgabe:** $\mathbf{O_n}$ mit aggregierten Kosten aller Versionen



Evaluierung

- Zwei große Ontologien
  - Gene Ontology (GO)
  - NCI Thesaurus (NCIT)
  - Versionen zwischen 2004 und 2009

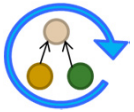


National Cancer Institute

- Kostenmodell:

concept		relationship		attribute		
add	*del*	*add*	*del*	*add*	*del*	*upd*
1	***2***	***1***	***2***	***0.5***	***0.5***	***0.5***

- Drei ausgewählte Analysen
  - Gesamtstabilität und Stabilitätsverteilung
  - Filterung der (in)stabilsten Regionen
  - Tracking der Stabilität einzelner Regionen

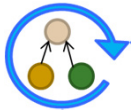


Gesamtstabilität

- Annahme: komplette Ontologie ist eine Region
  - Wurzel der Ontologie = Wurzel der Region

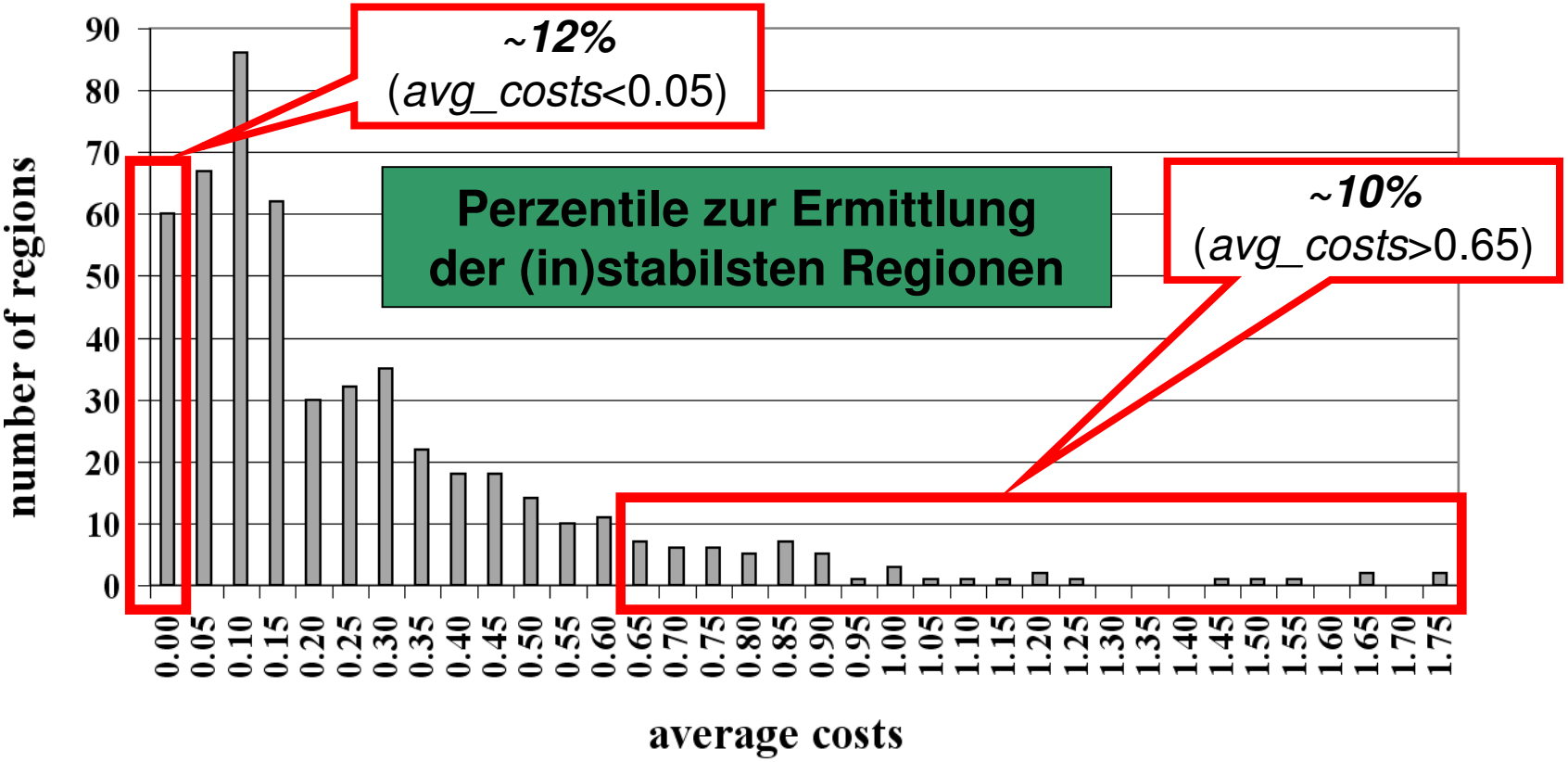
	abs_size(root)		*abs_costs(root)*		*avg_costs(root)*	
	2008	*2009*	*2008*	*2009*	*2008*	*2009*
GO	27,799	30,304	24,242	19,412	0.87	0.64
– MF	9,205	9,459	4,636	3,002	0.50	0.32
– BP	16,231	18,108	17,594	14,557	1.08	0.80
– CC	2,363	2,737	2,011	1,854	0.85	0.68
NCIT	71,337	77,455	23,165	36,562	0.32	0.47

  - ***abs_size:*** Zunahme in beiden Ontologien
  - ***abs_costs:*** bei GO höher in 2008, NCIT umgekehrt
  - ***avg_costs:*** im Durchschnitt GO instabiler
    - Biologische Prozesse (BP) als änderungsintensivste Subontologie



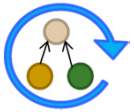
Verteilung der Stabilitäten

- Verteilung der Regionen bzgl. ***avg_costs***
  - Minimale ***rel_size*** = 0.3%
  - Beispiel: GO-BP in 2009 (***abs_size*** > 50 Konzepte)



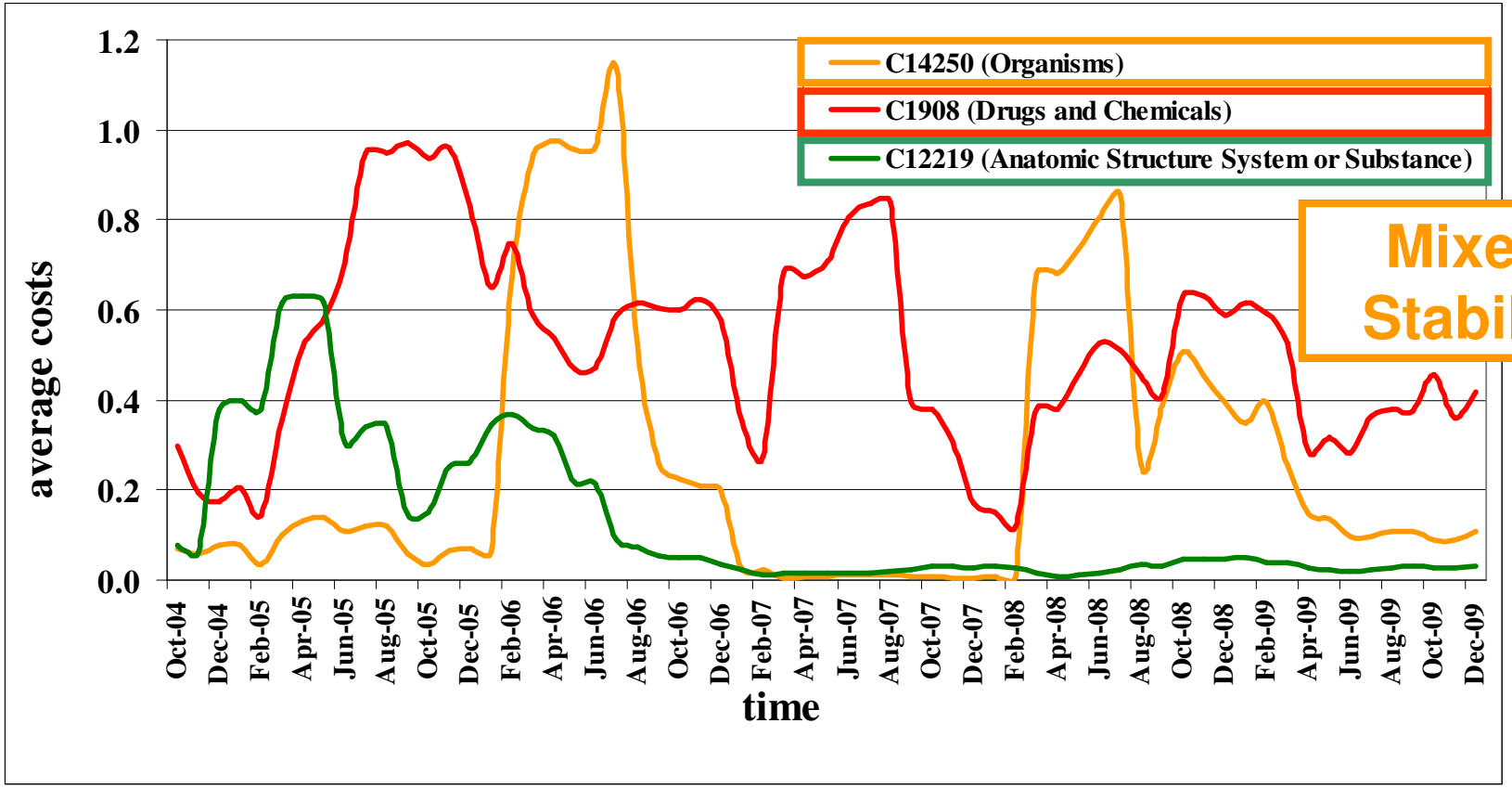
(In)stabilste Regionen in 2009

		accession	name	abs_size	rel_size	avg_costs
GO	unstable	GO:0005102	receptor binding	408	4.31%	0.95
		GO:0009653	anatomical structure morphogenesis	583	3.22%	1.22
		GO:0048856	anatomical structure development	566	3.13%	0.91
		GO:0033643	host cell part	77	2.81%	1.90
		GO:0003676	nucleic acid binding	241	2.55%	0.86
		GO:0048646	anatomical structure formation involved in morphogenesis	253	1.40%	0.92
	stable	GO:0031300	intrinsic to organelle membrane	36	1.32%	0.000
		GO:0030054	cell junction	31	1.13%	0.000
		GO:0050865	regulation of cell activation	184	1.02%	0.012
		GO:0075136	response to host	181	1.00%	0.019
		GO:0000151	ubiquitin ligase complex	25	0.91%	0.000
		GO:0016860	intramolecular oxidoreductase activity	71	0.75%	0.000
NCIT	unstable	C28428	Retired Concept	3,264	4.21%	3.49
		C53791	Adverse Event Associated with Infection	1,186	1.53%	2.36
		C45678	Industrial Aid	889	1.15%	1.40
		C74944	Clinical Pathology Procedure	747	0.96%	0.84
		C66892	Natural Product	708	0.91%	1.35
		C53543	Rare Non-Neoplastic Disorder	504	0.65%	1.22
	stable	C64389	Genomic Feature Physical Location	1,026	1.32%	0.000
		C23988	Mouse Neoplasms	886	1.14%	0.000
		C48232	Cancer TNM Finding	742	0.96%	0.000
		C53798	Adverse Event Associated with Surgery & Intra-Operative Injury	707	0.91%	0.000
		C43877	American Indian	555	0.72%	0.000
		C53832	Infection Adverse Event with Unknown Absolute Neutrophil Count	386	0.50%	0.000

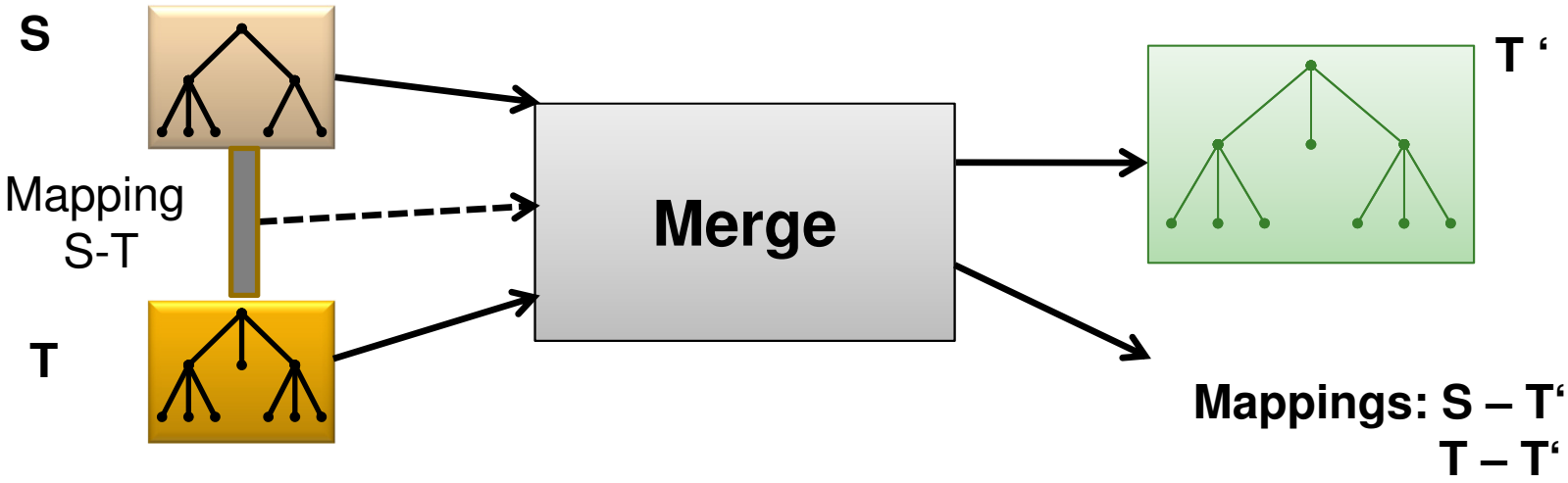


Tracking von Änderungsintensitäten

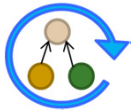
- NCIT mit 20 Hauptkategorien
  - Sliding Window der Länge 6 Monate zwischen 2004 und 2009
- Drei generelle Evolutionspatterns



Ontology Merging

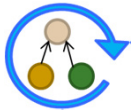


- Prozess der 2 (n) Ontologien zu einer integrierten (gemergten) Ontologie zusammenzufasst
  - Eingabe: 2 oder mehrere Ontologien, optional Mappings zwischen den Eingabeontologien
  - Ausgabe: integrierte (gemergte) Ontologie
- Varianten
  - Symmetric Merge
  - Target-driven Merge



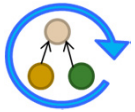
Verwandte Arbeiten

- Zahlreiche Arbeiten im Bereich Schemaintegration
  - Adressieren meist beides: Match und Merge
  - Oftmals hoher manueller Anteil, gerade bei komplexen Lösungen
  - Siehe VL Datenintegration (Top-Down vs. Bottom Up Schemaintegration)
- Wenige Arbeiten im Bereich Ontology Merging
  - PROMPT (1999-2000), Chimaera (2000)
  - FCA-Merge (2001)
  - Ebenfalls oftmals hoher manueller Aufwand erforderlich
  - Symmetric Merge
    - Bewahrung aller Inhalte aus beiden Eingabeontologien
- Hier in VL
  - Match-based Ontology Merging
  - Target-driven Merge → ATOM System

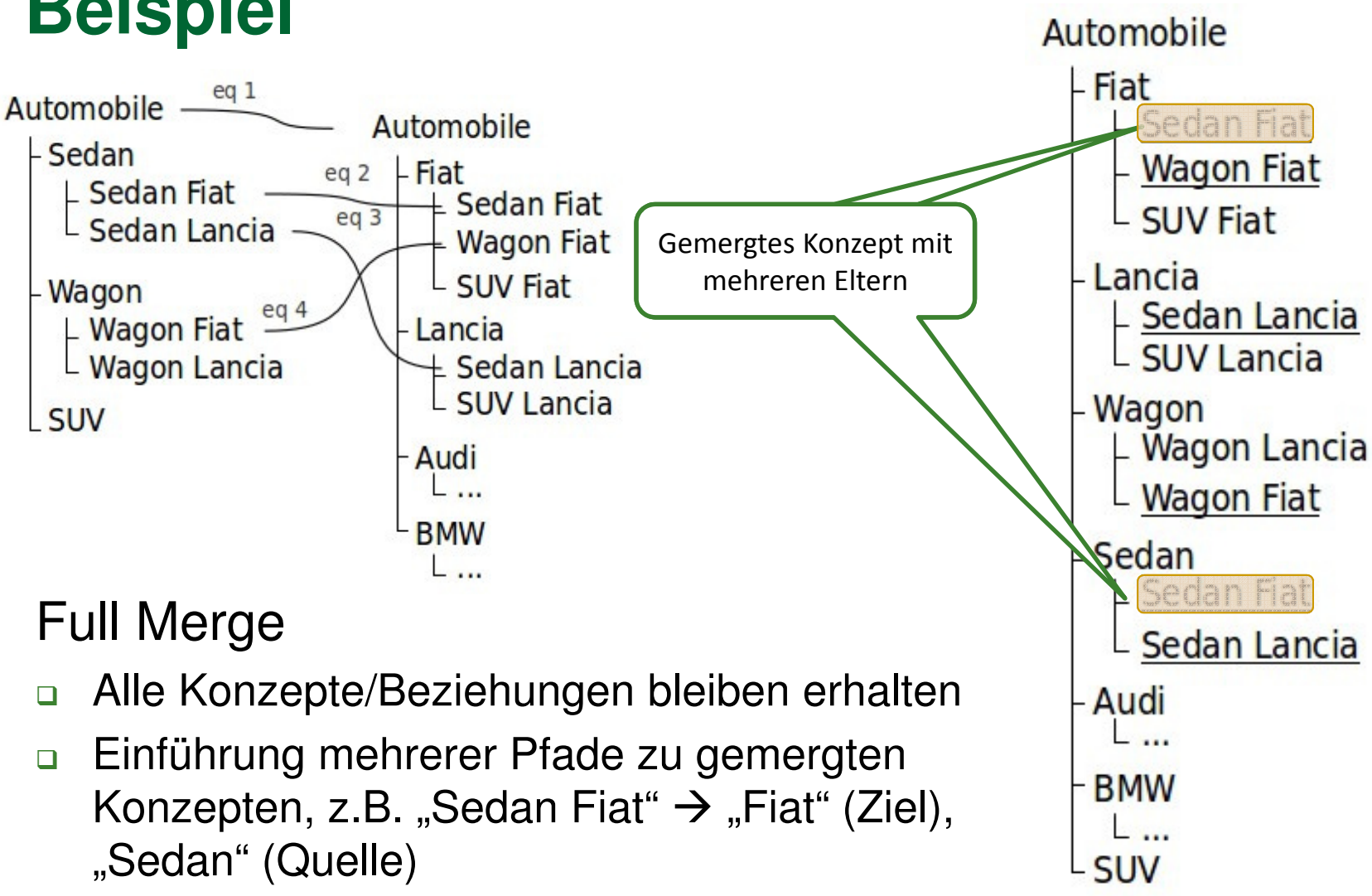


Symmetric Merge

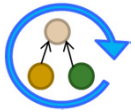
- Generelles Prinzip
  - Fasst alle äquivalenten Konzepte zusammen
  - Erhält zudem alle weiteren Konzepte und Beziehungen aus den Eingabeontologien
  - *Full Merge*
- Probleme
  - Informationen (z.B. ein Konzept) werden auf verschiedene Art und Weise innerhalb der Ontologie angeordnet
    - Reduzierte Verständlichkeit
    - Unnötige Redundanz (*semantic overlap*)
    - Z.B. mehrere Pfade zu ein und der selben Information
  - Reduzierte Stabilität
    - Präferierte Eingabeontologie (Mediatorontologie)
    - Z.B. Produktkatalog in einem Preisvergleichsportal, akzeptierte generelle Annatomieontologie für mehrere Spezies



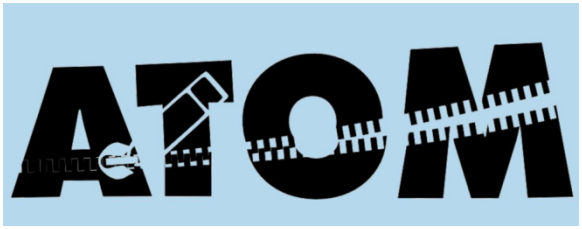
Beispiel



- Full Merge
  - Alle Konzepte/Beziehungen bleiben erhalten
  - Einführung mehrerer Pfade zu gemergten Konzepten, z.B. „Sedan Fiat“ → „Fiat“ (Ziel), „Sedan“ (Quelle)
  - Reduzierte Verständlichkeit aufgrund des Vermischens vers. Kategorisierungen

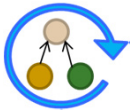


ATOM *

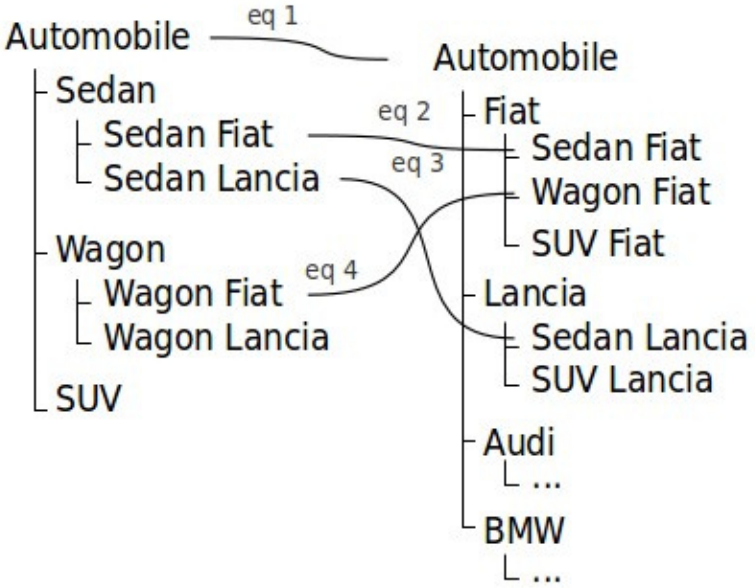


- Automatic Target-Driven Ontology Merging
  - Asymmetrischer, zielorientierter (target-driven) Merge-Ansatz
  - Reduzierung von „semantic overlap“ in der integrierten Ontologie
    - Erhaltung der Zielontologie
    - Vermeidung von Konzepten / Beziehungen aus der Quellontologie welche Redundanz einführen
  - Nutzung eines Ontologie-Mapping zwischen Eingabeontologien
    - Basisversion: Äquivalenz-Korrespondenzen
    - Optional: weitere Korrespondenz-Typen wie is_a / inverse-is_a
  - Semi(automatisch)
    - Ergebnis kann durch Nutzer verändert / angepasst werden

* *Raunich, S., Rahm, E.: ATOM: Automatic Target-driven Ontology Merging, Proc. ICDE 2011*

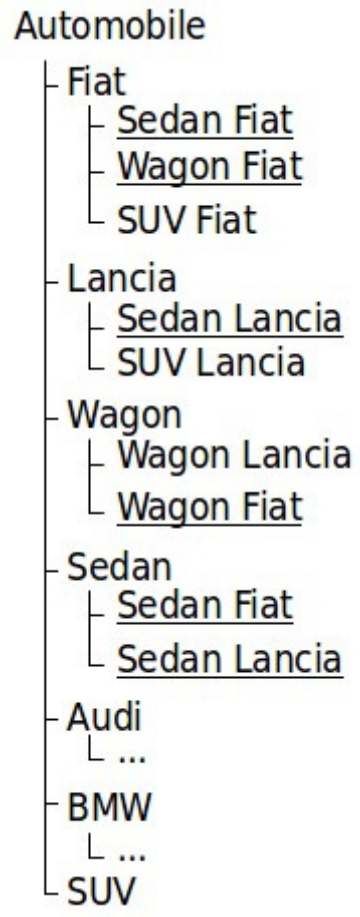


ATOM vs. Full Merge

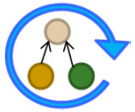


ATOM

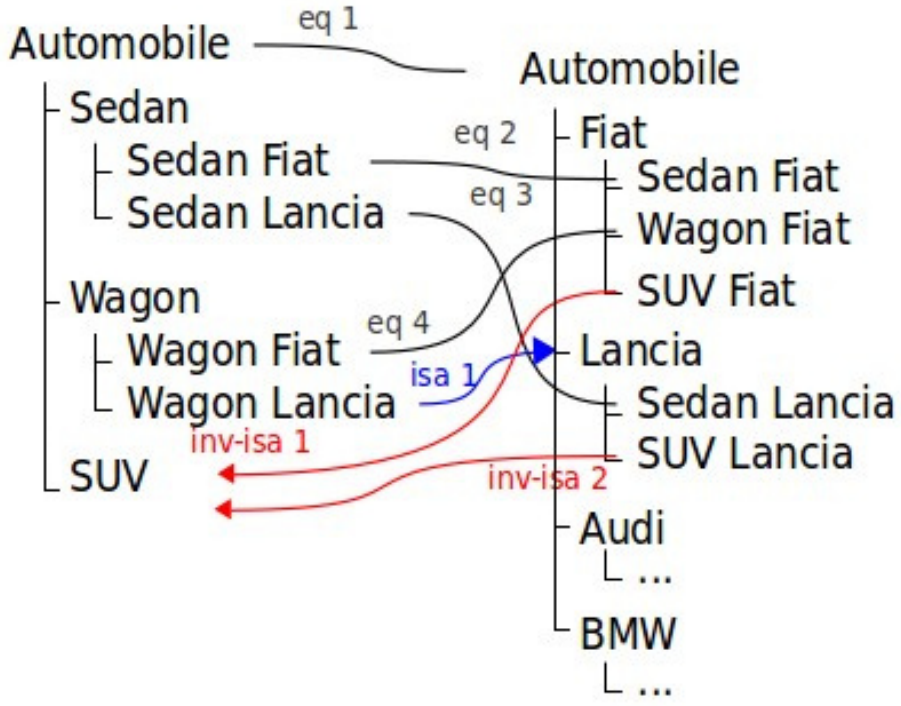
Full Merge

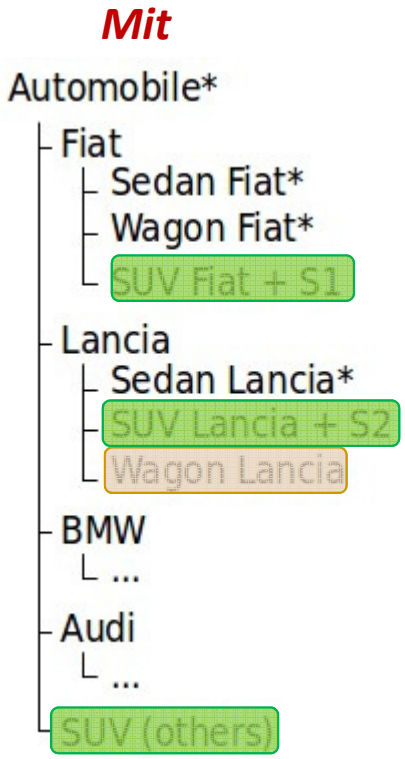


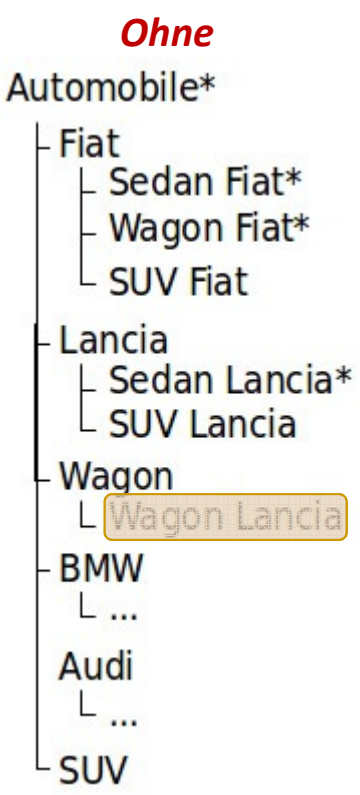
- ATOM Ergebnis
  - Erhaltung der Zielontologie
  - Kompakter als Full Merge, keine Mehrfachvererbung
- Aber
  - „Semantic overlap“ nur teilweise reduziert
    - Teilweise bessere Platzierung möglich (z.B. Wagon Lancia), Überlappung zwischen generellem SUV Konzept und SUV Fiat / SUV Lancia
  - Mehr Semantik im Ontologie-Mapping → weitere Verbesserung möglich



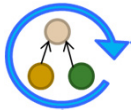
ATOM mit erweitertem Ontologie-Mapping





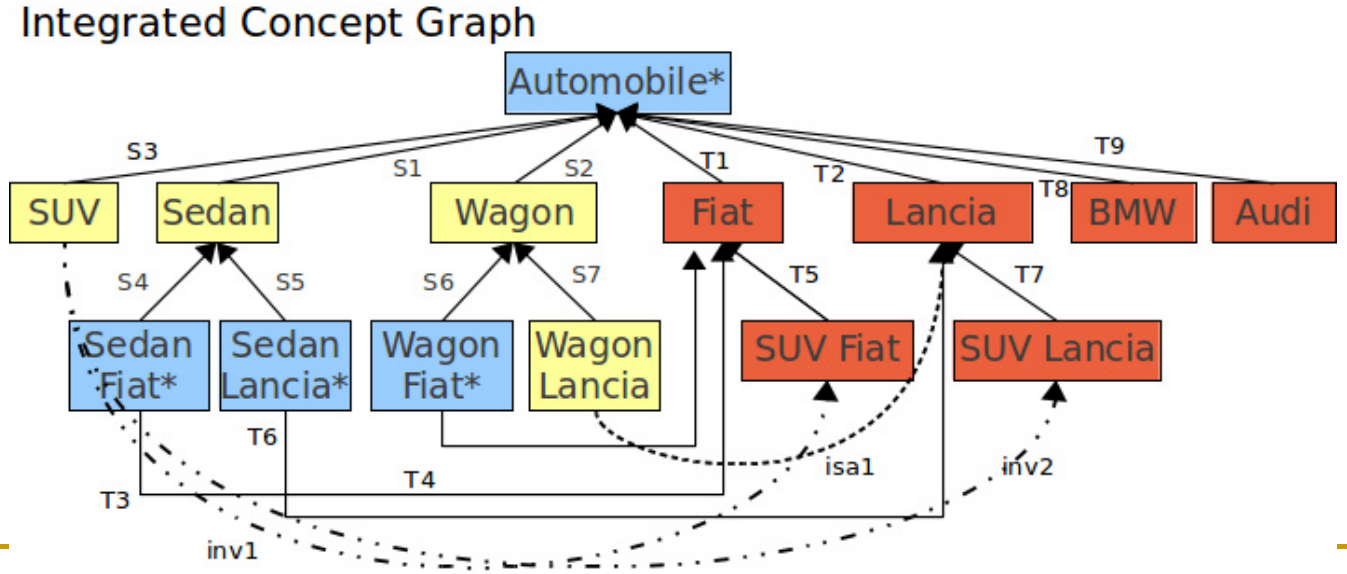


- Erweitertes Ontologie-Mapping
  - is_a und inverse-is_a Korrespondenzen in Ergänzung zu Äquivalenzen (eq)
  - Kategorie *Wagon Lancia* nun besser platziert
  - Keine Überlappung zwischen genereller *SUV Kategorie* und spezielleren *SUV Fiat* / *SUV Lancia* Kategorien

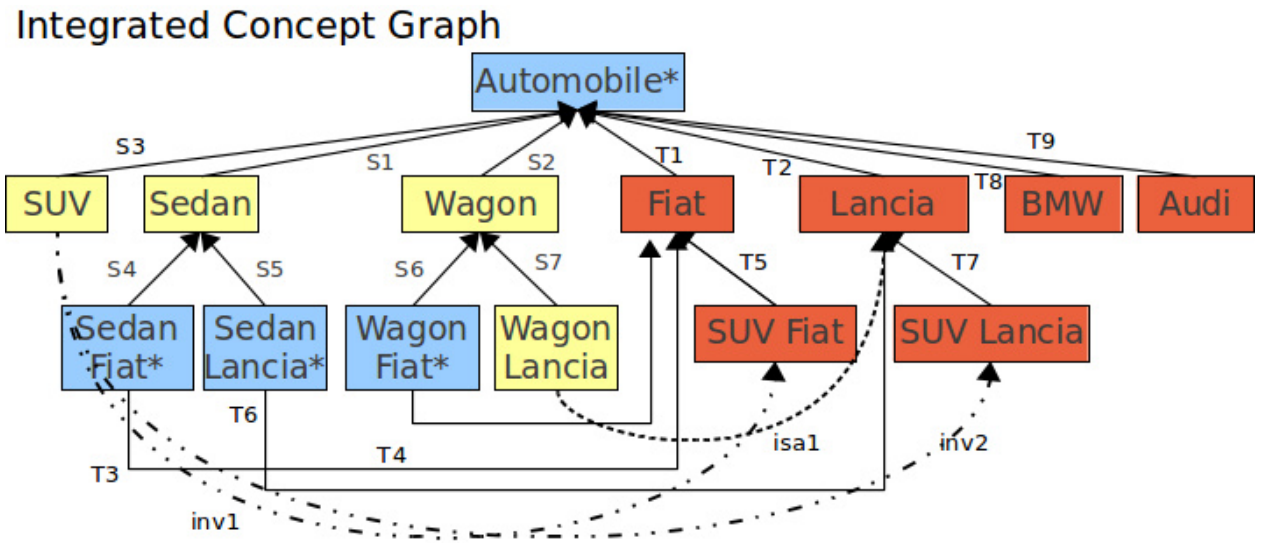


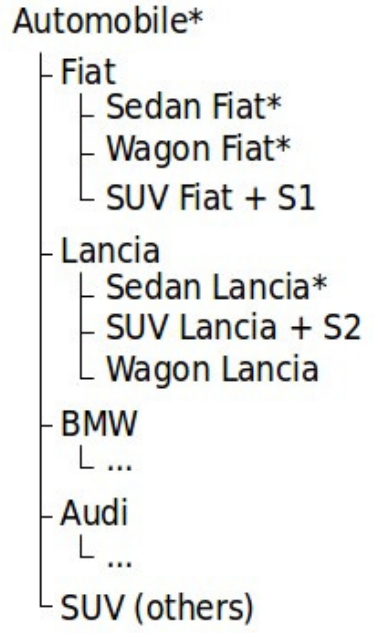
Merge Algorithmus (1)

- Vorphase
  - Verwendung der Eingabeontologie sowie des Ontologie-Mapping zum Aufbau eines *Integrated Concept Graph*
    - enthält alle Konzepte / Beziehungen aus S bzw. T
  1. Übernahme aller Konzepte der Eingabeontologien, Zusammenfassen äquivalenter Konzepte
  2. Eine gelabelte Kante für jede Beziehung aus S bzw. T
  3. Eine gelabelte Kante für jede is_a / inverse-is_a Korrespondenz



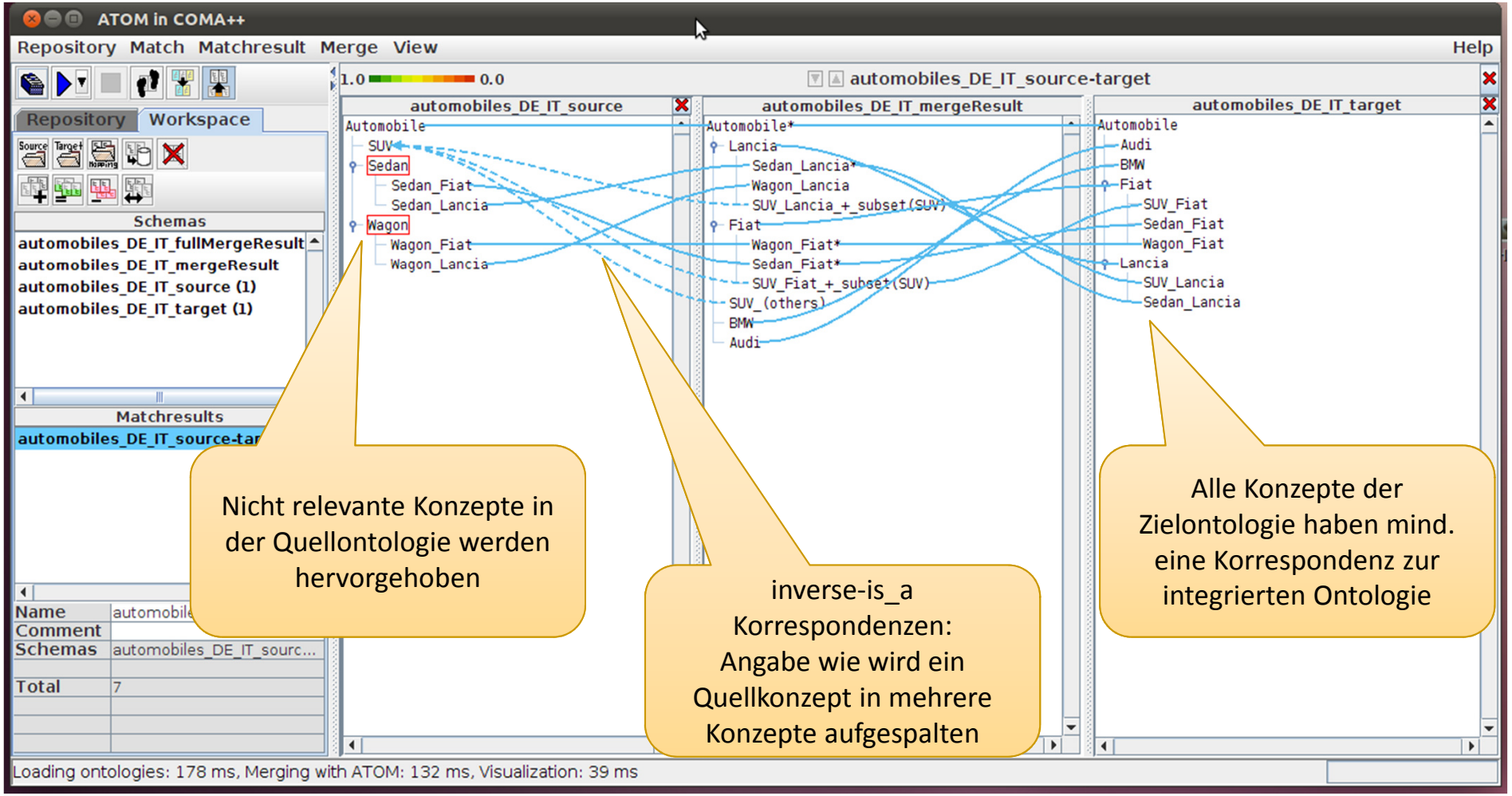
Merge Algorithmus (2)





- Hauptphase
  - Übernehme alle Konzepte / Beziehungen der Zielontologie in das finale Ergebnis (*target preservation*)
  - Übernehme alle Blattkonzepte sowohl aus der Quell- als auch Zielontologie (*instance preservation*)
  - Übernehme nur innere Konzepte, welche keine zusätzliche Redundanz einführen (*control of semantic overlap*)
  - Nutzung der is_a / inverse-is_a Korrespondenzen zur Verbesserung des Ergebnisses

Integration in COMA++



Zusammenfassung

- **Erweiterte Verfahren**
  - Komplexere Algorithmen / Verfahren, welche im Bereich Ontologie-Management eingesetzt werden
  - Lösung einer komplexen Aufgabe/Fragestellung
  - Reduzierung von manuellen Aufwand
- **Erkennung (in)stabiler Ontologieregionen**
- **Merging von Ontologien**
- **Weitere Verfahren**
  - Adaptierung von Mappings unter Evolution
  - Erkennung von Ontologiemodulen für Reuse
  - Term Enrichment Analysen in der Bioinformatik
  - ...