Ontology-driven Business Intelligence for Comparative Data Analysis

Michael SCHREFL

& the SemCockpit Team:
Neuböck, Neumayr, Schütz, …

26.06.2013

Overview

1. Introduction: Comparative Data Analysis
2. Multi-Dimensional Ontology
3. Ontology-based Measures and Scores
4. Comparative OLAP
5. BI Analysis Graphs
6. Guidance, Judgment, and Analysis Rules
7. Conclusion

1. Introduction

- Setting: SemCockpit („Semantic Cockpit“)
- Is-reporting vs. is-to-target-comparison vs. *is-to-is-comparison*
- Use case: DM2-Analysis project (of an Austrian Health Insurer)
- Comparative Data Analysis: Aims & Characteristics
- The wish list of analysts – and how to meet it
- The technical „ingredients“: md-ontology, (generic) measures & scores, BI analysis graphs, guidance/judgment/analysis rules
- Steps to run a comparative analysis project („The SemCockpit process“)
- Underlying DWH schema (MDO-base)

Setting: The SemCockpit Project

An

- ontology-driven
- interactive
- Business Intelligence tool
- for comparative data analysis

Partners:

Solvistas (BI solutions developer and provider)

Johannes Kepler University of Linz

OÖGKK, DAK (Austrian and German Health Insurers)

Funded by: Austrian Ministry of Transport, Innovation & Industry

OLAP (Online Analytical Processing) in BI

- *Is*-reporting: business monitoring
- *Is-to-target* comparison: performance management
- *Is-to-Is* comparison: *comparative data analysis*

Example: DM2-Analysis Project (ÖOGKK)

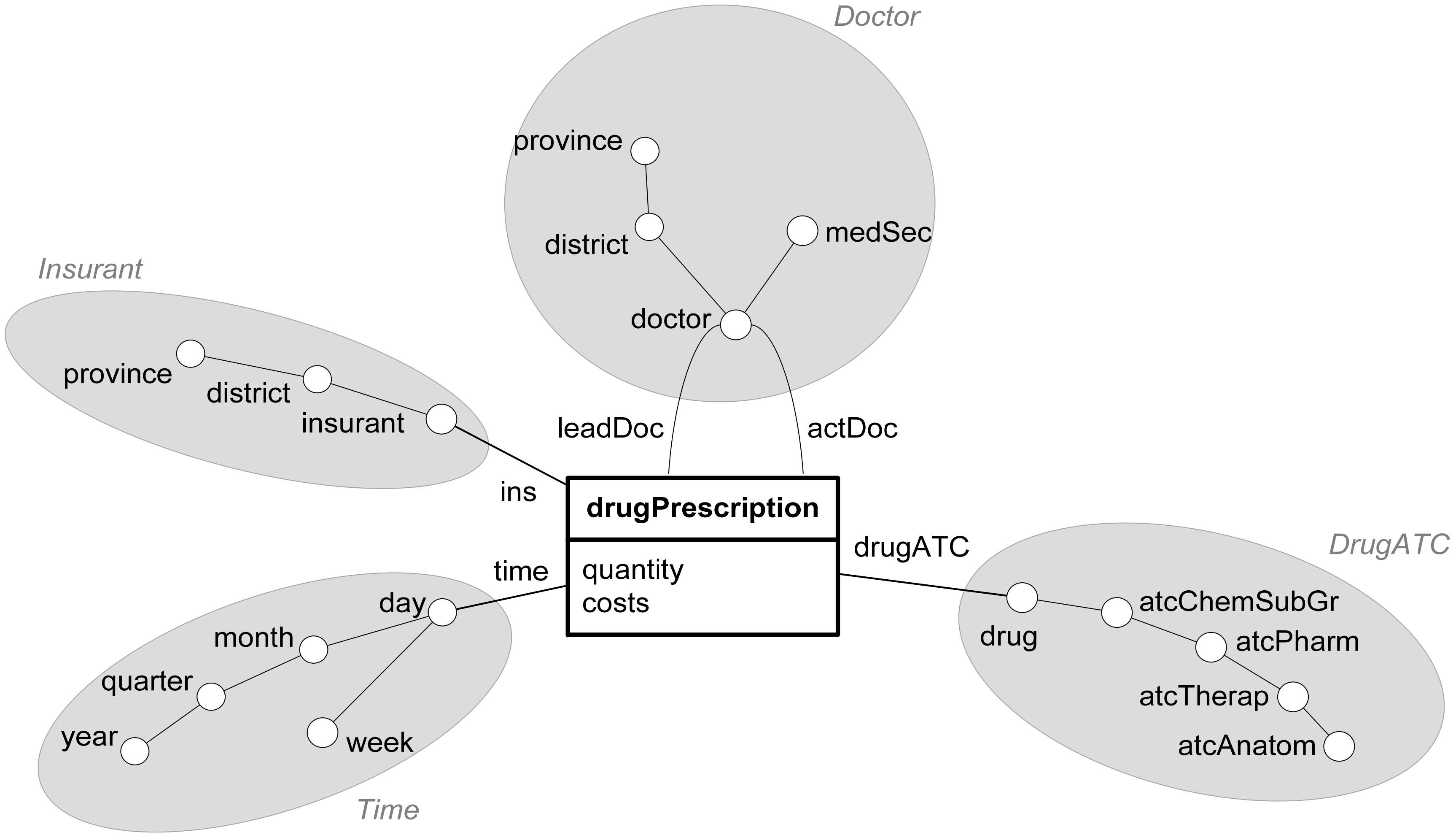
Our insurance organisation pays a high amount for insurants with Diabetes Mellitus of Type 2 (DM2).

Could costs be reduced?

What can we learn by comparison? What should we compare with what?

What specific questions to ask statisticians or data miners?

Example: DM2-Analysis (ÖOGKK, simplified)



Data Warehouse (simple, nothing new)

- Schema
  - Dimension: Levels, rolls-up relation between levels (lattice)
  - Fact class: Dimension roles, Measures
- Extension:
  - Fact: Multi-dimensional point (of dimension nodes) + measures
  - Dimension node: Each node belongs to one level
  - Rolls-up relation *rollsUp(sub,sup)*; transitive,reflexive: *rollsUp**
- Analysis:
  - Roll-up facts and cubes with “derived” measures
  - Granularity: Levels of nodes of fact, denoted by *[L1,…Ln]*

Example: Search for meaningful comparisons

Any striking differences in comparing costs for DM2 patients?

What are meaningful comparison groups?
Patients of different insurers
Patients of different provinces
Patients of rural vs. urban areas

Avg drug cost per patient,
Drug costs prescribed by GPs for regular patients

What drugs to consider?
… oral antidiabtic drugs, insulin, metaformin, …

Example: Relevant concepts … meaning?

What is an urban district?

What is a country doctor?

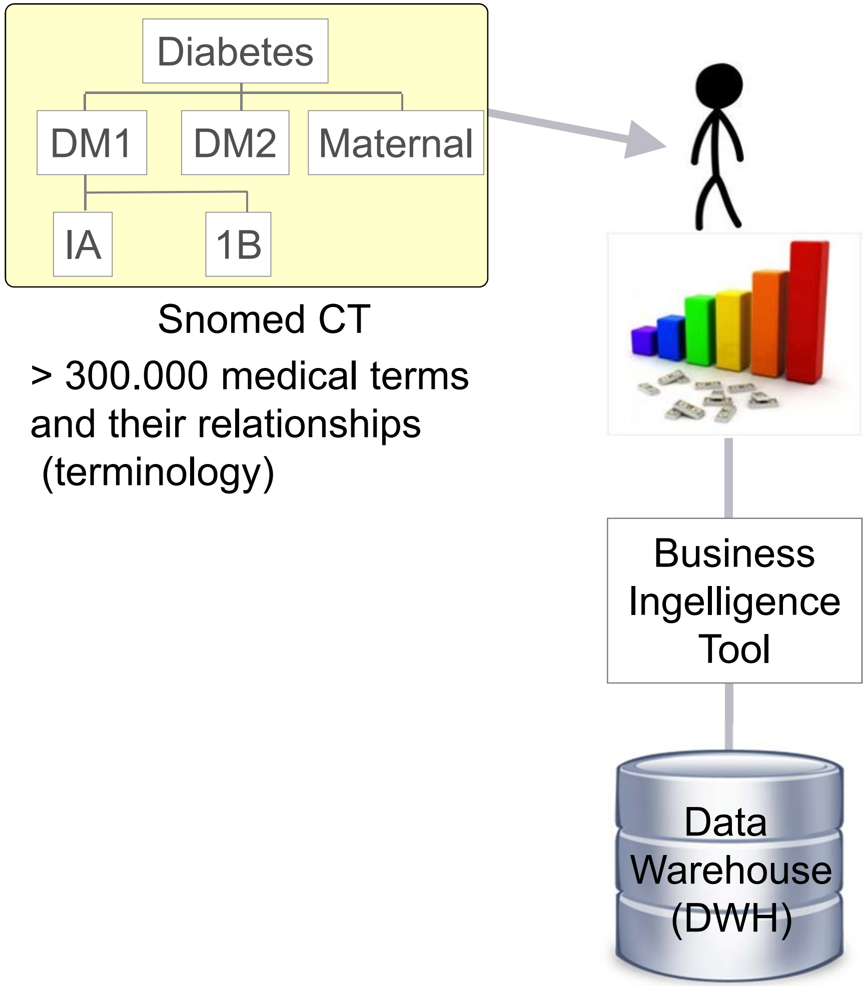
What is a rural district?

What are “cheap“ oral antidiabetic drugs?

What is a regular patient of a doctor?

What are DM2-Patients?

SemCockpit – An Ontology-Driven, Interactive Business Intelligence Tool for Comparative Data Analysis



Diabetes

DM1

DM2

Maternal

IA

1B

Snomed CT

> 300.000 medical terms and their relationships (terminology)

Business Ingelligence Tool

Data Warehouse (DWH)

Common BI-Tool

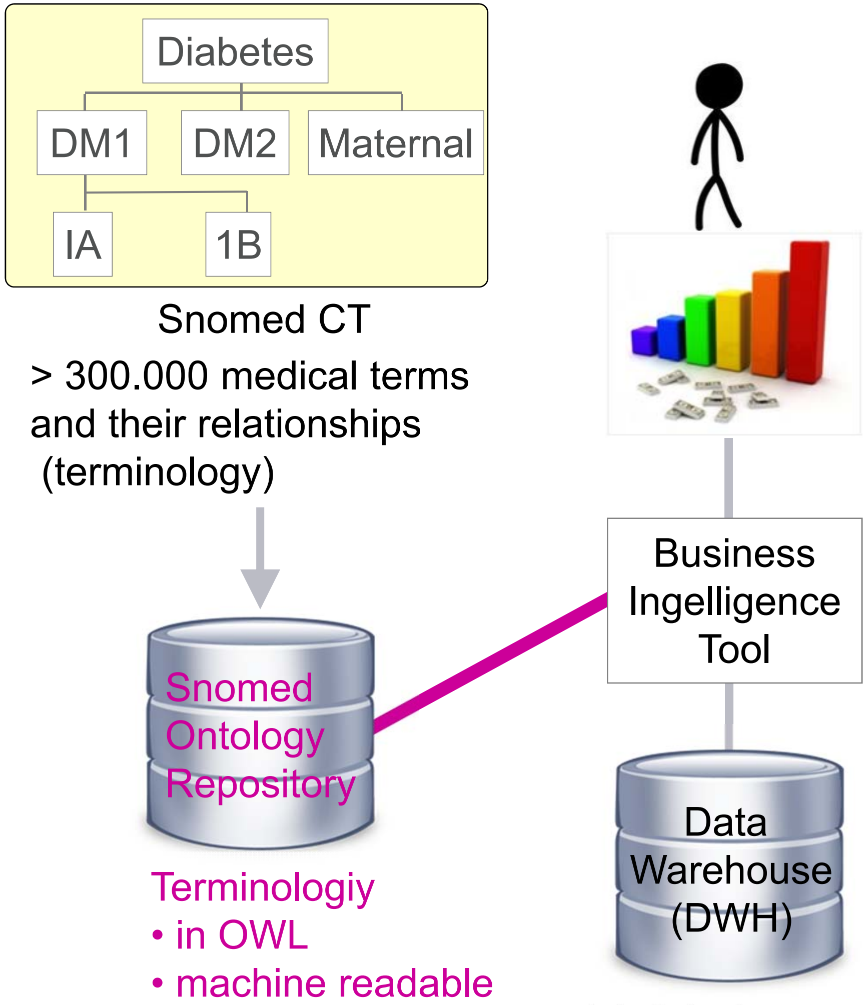
- „Diagnoses Codes“ in DWH, associated medical terminology not available in BI tool

⇩

Comparative Data Analysis

- ✘ targeted
- ✘ in business terminology

SemCockpit – An Ontology-Driven, Interactive Business Intelligence Tool for Comparative Data Analysis



Diabetes

DM1

DM2

Maternal

IA

1B

Snomed CT

> 300.000 medical terms
and their relationships
(terminology)

Snomed Ontology Repository

Business Ingelligence Tool

Data Warehouse (DWH)

Terminologiy
• in OWL
• machine readable

SemCockpit

- Meaning (semantics) of „Diagnosis Codes“ available in BI tool via ontology

⇩

Comparative Data Analysis

✓ targeted

✓ in business terminology

+ simpler, more efficient

Comparative Data Analysis in BI: *Aims*

- Understand your business
- Know where to investigate further
- Detect striking differences between comparable groups of entities/facts
- Suggest plausible explanations
- Conjecture or refute possible cause-effect relationships through further comparisons
- Pre-cursor to data mining

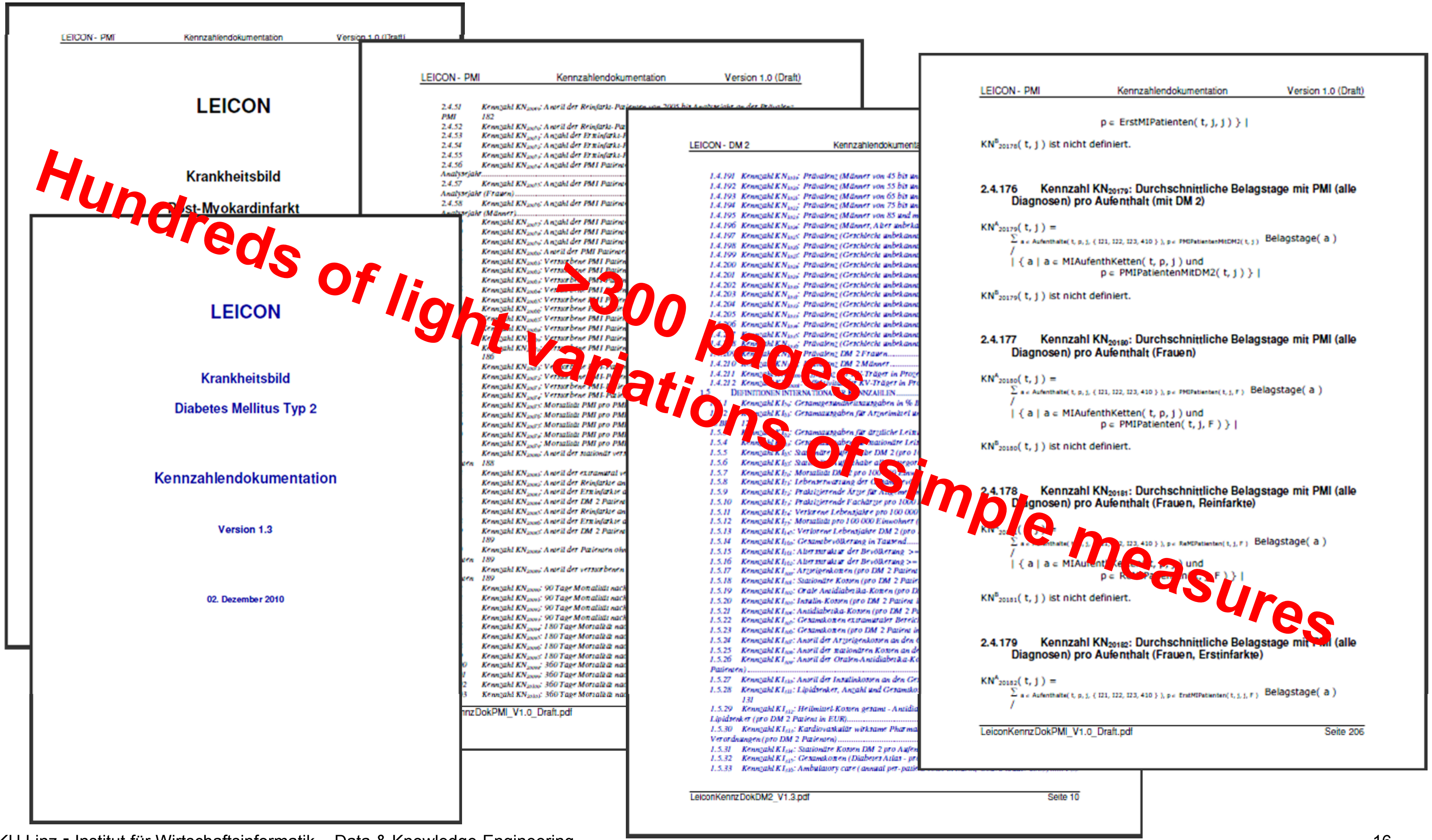
Comparative Data Analysis: *Characteristics*

- Initially vague analysis question
- What are relevant measures and comparison groups?
- Specific analysis questions crystallise over time
- Interactive, exploratory, iterative
- Applications of OLAP-operations (on two groups in parallel)
- Repetitive: once questions clear, comparative analysis process re-applied for similar comparison situations (e.g. one year later)
- Two phases:
  1. Search for relevant measures and comparison groups
  2. Proper comparative analysis (repeated)

Example: From vague to more specific

- Specific analysis questions crystallise over time
  - Compare the prescription costs for oral anti-diabetic drugs (OAD) of DM2 patients who are treated by doctors in rural areas with ones of urban areas.
- Interactive, exploratory, iterative
  - Compare patients in Upper Austria with patients in Austria ➔ compare patients in urban districts of Upper Austria with patients in urban districts of Austria ➔ …
  - *Hundreds of light variations of simple measures (>300 pages list)*
- Applications of OLAP-operations (on two groups in parallel)
  - Compare costs of Upper Austria with Austria per year ➔ compare costs of Upper Austria with Austria per month

Example: DM2-Analysis, Measure Catalogue



Hundreds of light variations of simple measures

>300 pages

LEICON

Krankheitsbild

Post-Myokardinfarkt

LEICON

Krankheitsbild

Diabetes Mellitus Typ 2

Kennzahlendokumentation

Version 1.3

02. Dezember 2010

LEICON - PMI Kennzahlendokumentation Version 1.0 (Draft)

p ∈ ErstMIPatienten(t, j, j) } |

KN^{B}_{20178}(t, j) ist nicht definiert.

2.4.176 Kennzahl KN_{20179}: Durchschnittliche Belagstage mit PMI (alle Diagnosen) pro Aufenthalt (mit DM 2)

KN^{A}_{20179}(t, j) =
$\sum_{a \in Aufenthalte(t, p, j, \{121, 122, 123, 410\}), p \in PMIPatientenMitDM2(t, j)}$ Belagstage(a)
/
| { a | a ∈ MIAufenthKetten(t, p, j) und
p ∈ PMIPatientenMitDM2(t, j) } |

KN^{B}_{20179}(t, j) ist nicht definiert.

2.4.177 Kennzahl KN_{20180}: Durchschnittliche Belagstage mit PMI (alle Diagnosen) pro Aufenthalt (Frauen)

KN^{A}_{20180}(t, j) =
$\sum_{a \in Aufenthalte(t, p, j, \{121, 122, 123, 410\}), p \in PMIPatienten(t, j, F)}$ Belagstage(a)
/
| { a | a ∈ MIAufenthKetten(t, p, j) und
p ∈ PMIPatienten(t, j, F) } |

KN^{B}_{20180}(t, j) ist nicht definiert.

2.4.178 Kennzahl KN_{20181}: Durchschnittliche Belagstage mit PMI (alle Diagnosen) pro Aufenthalt (Frauen, Reinfarkte)

$\sum_{a \in Aufenthalte(t, p, j, \{121, 122, 123, 410\}), p \in ReMIPatienten(t, j, F)}$ Belagstage(a)
/
| { a | a ∈ MIAufenthKetten(t, p, j) und

KN^{B}_{20181}(t, j) ist nicht definiert.

2.4.179 Kennzahl KN_{20182}: Durchschnittliche Belagstage mit PMI (alle Diagnosen) pro Aufenthalt (Frauen, Erstinfarkte)

KN^{A}_{20182}(t, j) =
$\sum_{a \in Aufenthalte(t, p, j, \{121, 122, 123, 410\}), p \in ErstMIPatienten(t, j, j, F)}$ Belagstage(a)
/

LeiconKennzDokPMI_V1.0_Draft.pdf Seite 206

LeiconKennzDokDM2_V1.3.pdf Seite 10

Example: Two Phases

1. Search for relevant measures and comparison groups

□ Which patients to include when comparing groups of doctors?
- regular patients only (e.g., minimum of 4 contacts per year)
but: regular is different for GPs (6 contacts) and oculists (2 contacts)

□ Analyst recognizes that it is meaningful to compare prescription costs of oral anti-diabetic drugs of patients in urban versus patients in rural areas

2. Proper comparative analysis (repeated)

□ Once questions clear: comparative analysis process re-applied for similar comparison situations (e.g. one year later)

□ Compare drug prescription costs of urban and rural patients: apply same comparisons in other years or for other diseases

The *wish list* of business analysts

1. "*Business terms*" as first-class citizens in BI tool
2. Support in definition and organisation of business terms
3. Use business terms in defining
   a) measures
   b) comparison groups
   c) analysis steps and analysis paths
4. Reuse of existing terminologies of the domain
5. "*Comparison*" *(*between a group of interest and group of comparison) as first-class modelling primitive

The *wish list* of business analysts (ctd.)

6. Comparison explicitly captured and not left to the “human eye” through visual comparison by diagram inspection
7. Intelligent “*guidance*” on how to proceed in analysis
8. Capture and exploit “*judgement knowledge*”, i.e., knowledge about striking differences between compared groups
9. *Context-sensitive support* on which business terms, measures, comparison groups, or analysis steps are applicable in a given (i.e., already partially specified) analysis situation
10. Automatic mapping of high-level analysis to data warehouse

What current BI tools provide

"Surf and save" BI tools, e.g., Tableau

+ perfect for is-reporting, multi-granular, multi-perspective
- no support for business terms
- no direct support for comparison
- no guidance

Data Warehouse Products of mega vendors

+ perfect for complex analysis tasks
- no direct support for comparison (but workarounds)
- low IT-skilled analysts complain about complex use and rely on IT-department "programming" cycles

What SemCockpit provides

Modelling and reasoning support to meet the analysts wish list

1. *Multi-dimensional ontology* (MDO) to capture “business terms” in data warehouse context
2. MDO-concepts are *organised* in subsumption hierarchies along levels, dimension hierarchies, and dimension spaces through reasoning; concepts may be defined for some context and be “overridden” in a subcontext
3. Use of MDO-concepts to define measures and comparison groups
4. Re-use of existing domain ontology as “*semantic dimension*”

What SemCockpit provides (ctd.)

5. *Comparative analysis situations* to express the comparison of a group of interest against a group of comparison
6. *Scores* (complementing measures) to explicitly capture comparison results
7. Guidance through *analysis graphs* and associated *guidance rules*
8. *Judgment rules* capture judgment knowledge
9. Reasoning support to determine concepts, measures, and comparisons applicable in a current analysis situation
10. Automatic mapping of high-level analysis to SQL.

Complementary Work (Selection)

- Ontology-based Data Warehouse Design
  - Multi-dimensional design from ontologies (Oscar Romero et.al.)
  - Concept-based design of data warehouses (Jarke et. al.)
- Ontology-based Integration
  - Ontology-based information extraction (e.g. Saggion, et.al.)
  - Ontology-based querying (of multiple sources) (e.g. Spahn, et.al.)
- Extracting semantic web data into a multi-dimensional data warehouse
  - Multi-dimensional Integrated Ontologies: Designing Semantic Data Warehouses (e.g., Nebot/Berlanga/Perez/Aramburu/Pederson)
- SemCockpit
  - *a-posterior enrichment* of given DWH
  - specific support for comparative data analysis

Comparison: Comparative Facts & Scores

Fact

- md-point
  - Point of interest
- Measure
  - describes a md-point
  - <ins:Vienna,time:2010>. totalCosts

Comparative Fact

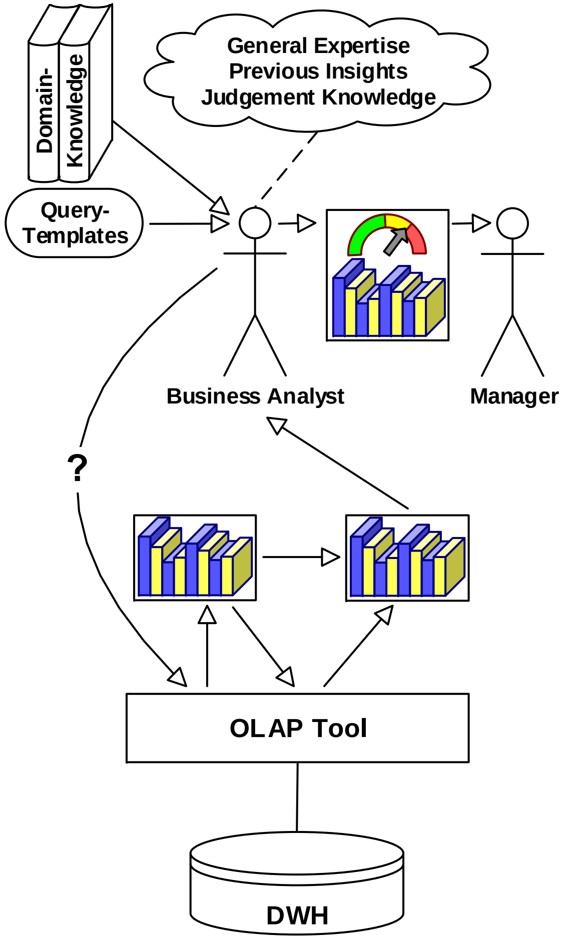
- 2 md-points
  - Point of interest
  - Point of comparison
- Score
  - describes relationship between 2 md-points
  - (<ins:Tyrol,time:2011>, <ins:Austria,time:2011>). MeanPercentilRkOfCosts-PerPatient

Kinds of Comparisons

- Group of Interest (GoI) and Group of Comparison (GoC):
  - Identified by point of interest & comparison (or more, see later)
  - Members: Facts
- Groups involved:
  - Group vs. Peer Group
  - Group vs. Super Group
- Scores: Typically based on comparing measure of 2 groups by
  - Simple arithmetic: Ratio or percentage difference
  - Median/Mean PercentileRank (typically for group vs. super group)
    - of median/average "member" of group of interest
    - considered as "member" of group of comparison

BI: Conventional vs. SemCockpit

Conventional Comparative Data Analysis



Domain-Knowledge

General Expertise
Previous Insights
Judgement Knowledge

Query-Templates

Business Analyst

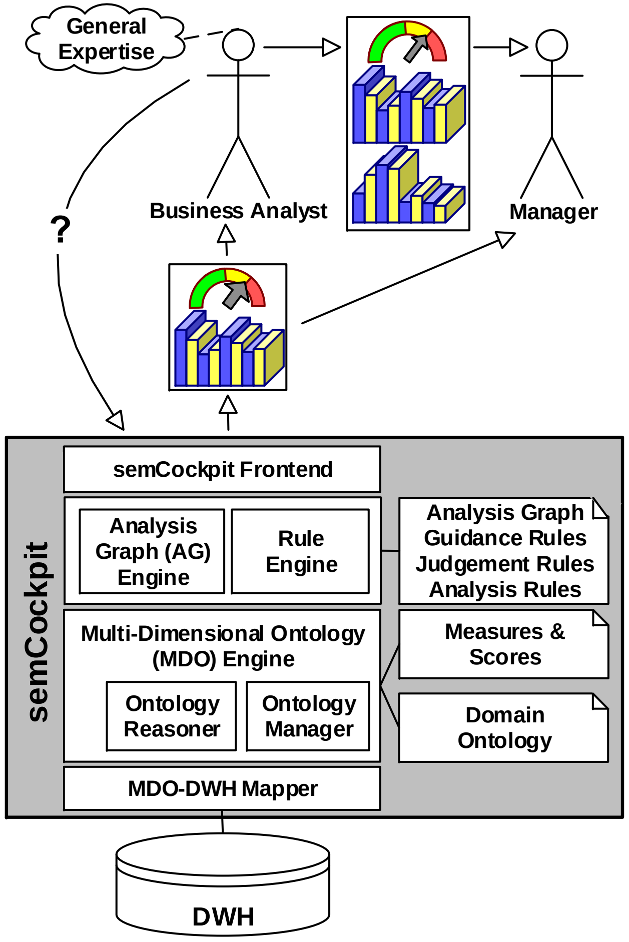
Manager

?

OLAP Tool

DWH

Comparative Data Analysis with semCockpit



General Expertise

Business Analyst

Manager

?

semCockpit

semCockpit Frontend

Analysis Graph (AG) Engine

Rule Engine

Analysis Graph
Guidance Rules
Judgement Rules
Analysis Rules

Multi-Dimensional Ontology (MDO) Engine

Ontology Reasoner

Ontology Manager

Measures & Scores

Domain Ontology

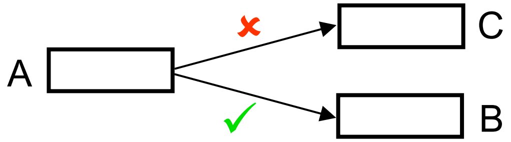
MDO-DWH Mapper

DWH

SemCockpit: Stack

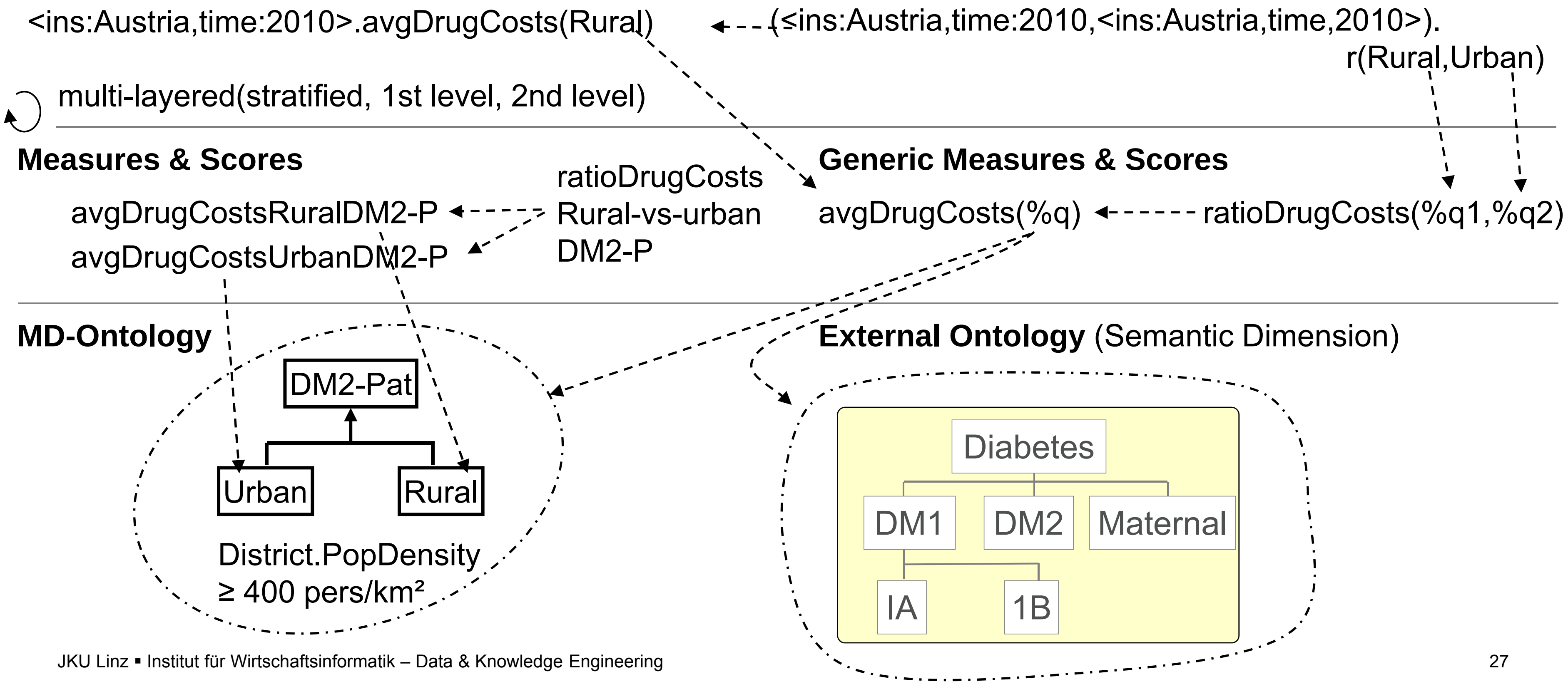
Context-Sensitive Semantic Guidance

Analysis Graphs & Judgement Rules



Judgement Rule:

FOR „comp. of (Internist,DM2-Pat; GP, DM2-Pat)“
IF ratioDrugCosts > 1.4
DO Explanation: Internists better trained for DM2



Measure & Score Application

<ins:Austria,time:2010>.avgDrugCosts(Rural)

(≤ins:Austria,time:2010,<ins:Austria,time,2010>).
r(Rural,Urban)

multi-layered(stratified, 1st level, 2nd level)

Measures & Scores

avgDrugCostsRuralDM2-P
avgDrugCostsUrbanDM2-P

ratioDrugCosts
Rural-vs-urban
DM2-P

Generic Measures & Scores

avgDrugCosts(%q)

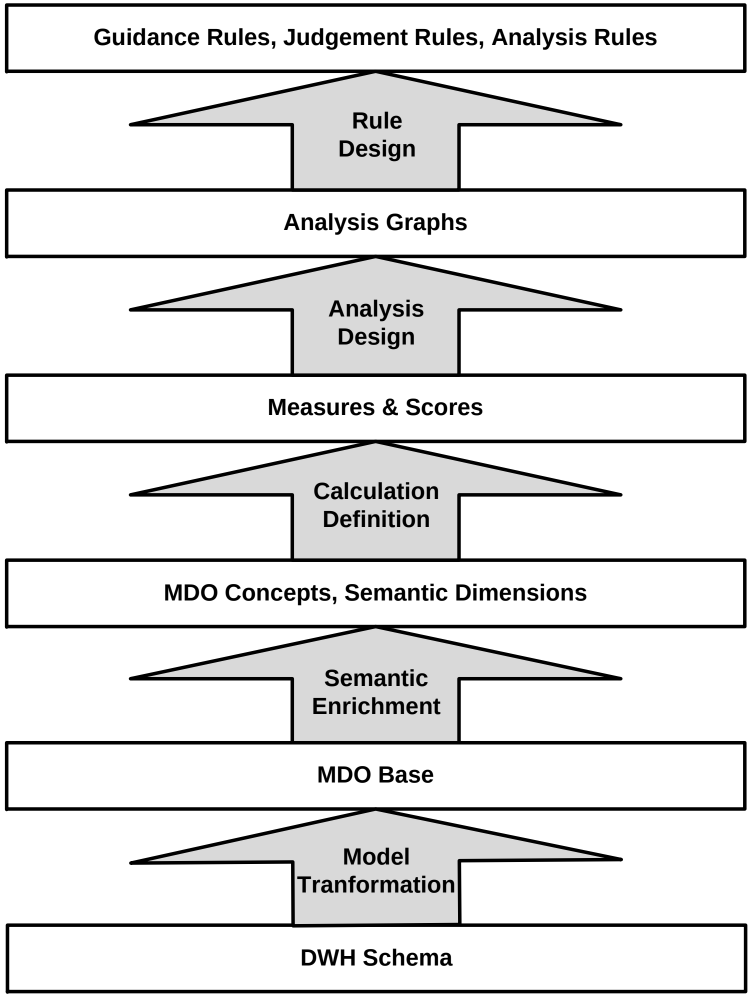
ratioDrugCosts(%q1,%q2)

MD-Ontology

District.PopDensity
≥ 400 pers/km²

External Ontology (Semantic Dimension)

The SemCockpit Process



Guidance Rules, Judgement Rules, Analysis Rules

Rule Design

Analysis Graphs

Analysis Design

Measures & Scores

Calculation Definition

MDO Concepts, Semantic Dimensions

Semantic Enrichment

MDO Base

Model Tranformation

DWH Schema

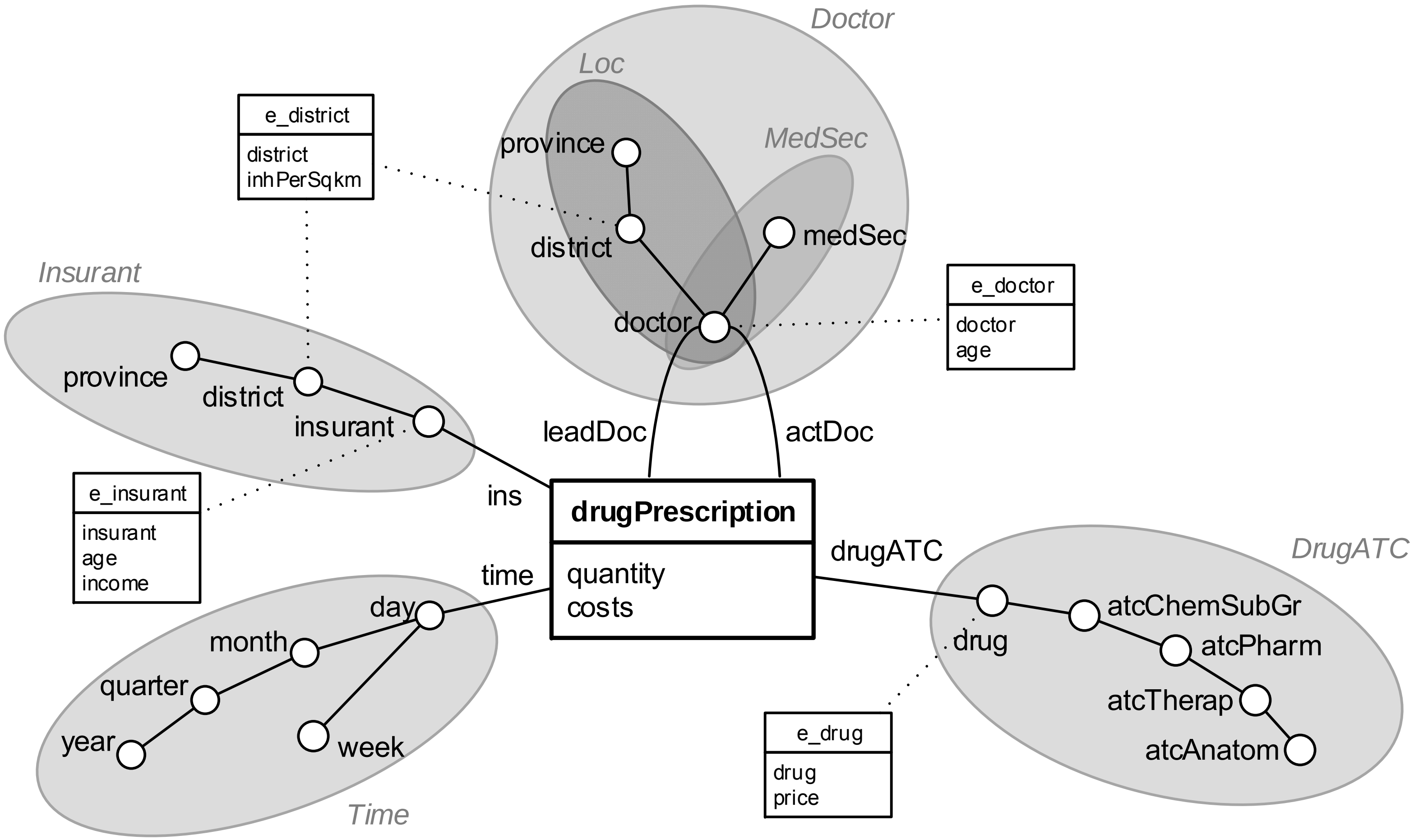
Development strategy for tool support

- Focus on primary analysis needs of business partners
- Provide support for most frequent analysis tasks
- Keep it simple (for users and to make reasoning tractable)
- Possibly ignore specific, complicated cases (seldom anyway!)
- Provide extension hooks such that complicated concepts, measures, scores, and analysis queries can, if needed, still be defined by IT-department and be provided as “built ins”
- “Built-ins” should be usable like non-built-ins (but can excluded from certain reasoning tasks)
- Initially basic, e.g., no ragged hierarchies, etc.

MDO-Base: Canonical Model for DWH

- Background: *Data Cube*
  - □ Comprises all „roll-up“ points + derived measures
  - □ Analyst selects points (of some granularity) + measure
  - □ Point: exactly *one* node for *each* dimension role of data cube, (Motivation: avoid „nested structures“ in dimension columns)
- MDO-Base: Like Dimensional-Fact-Model, but
  - □ Non-dimensional attributes extracted into „entity classes“ (for re-use and for later definition of entity concepts)
  - □ Unique role assumption for dimension roles to facilitate „drill across“, and (explained later) concept definition and use
  - □ Named hierarchies and hierarchy-specific dimension roles
  - □ Dimension space: set of dimension roles + granularity/range

DWH schema (MDO-Base)



Model Transformation: DWH -> MDO-Base

- Transformation steps
  1. If the intended analysis within a dimension is „multi-dimensional“ (e.g., doctor location, doctor medical-section), use hierarchies
  2. Extract non-dimensional attributes of a level into *entity classes* (for sharing across dimensions and for later definition of entity concepts)
  3. Define a *universal dimension space* consisting of the dimension roles (of dimensions) used to define fact classes and data cubes
  4. Define *dimension spaces*, each consisting of a set of dimension roles and a granularity (or granularity range), that are used to define the domain of fact classes (or cubes), concepts, …

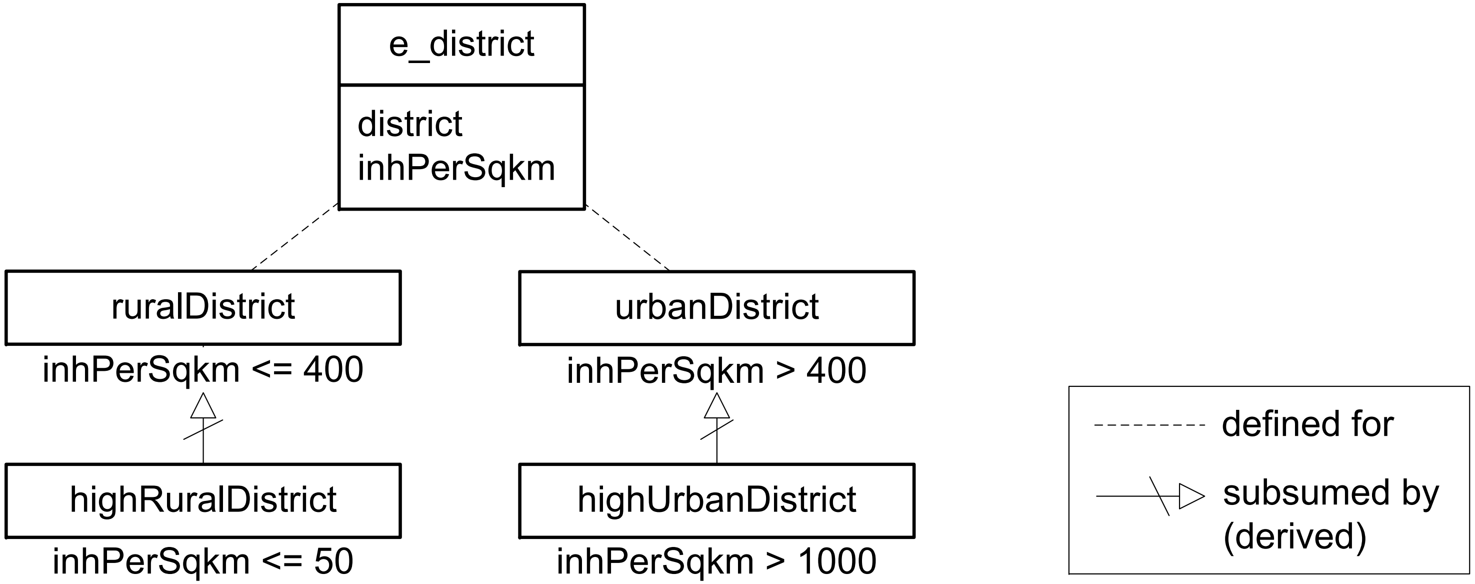
2. Multi-dimensional Ontology (md-ontology)

- Enriches given dimensional model from underlying data warehouse with “business terms” in analysis context
- Concepts defined over entities, nodes, points, pairs of points
- Flat vs. hierarchical concepts
- Context-specific and contextualized concepts
- Semantic dimensions (incorporating domain ontology)
- Mapping into OWL: for concept organization
- Mapping into SQL: for execution of OLAP in DWH

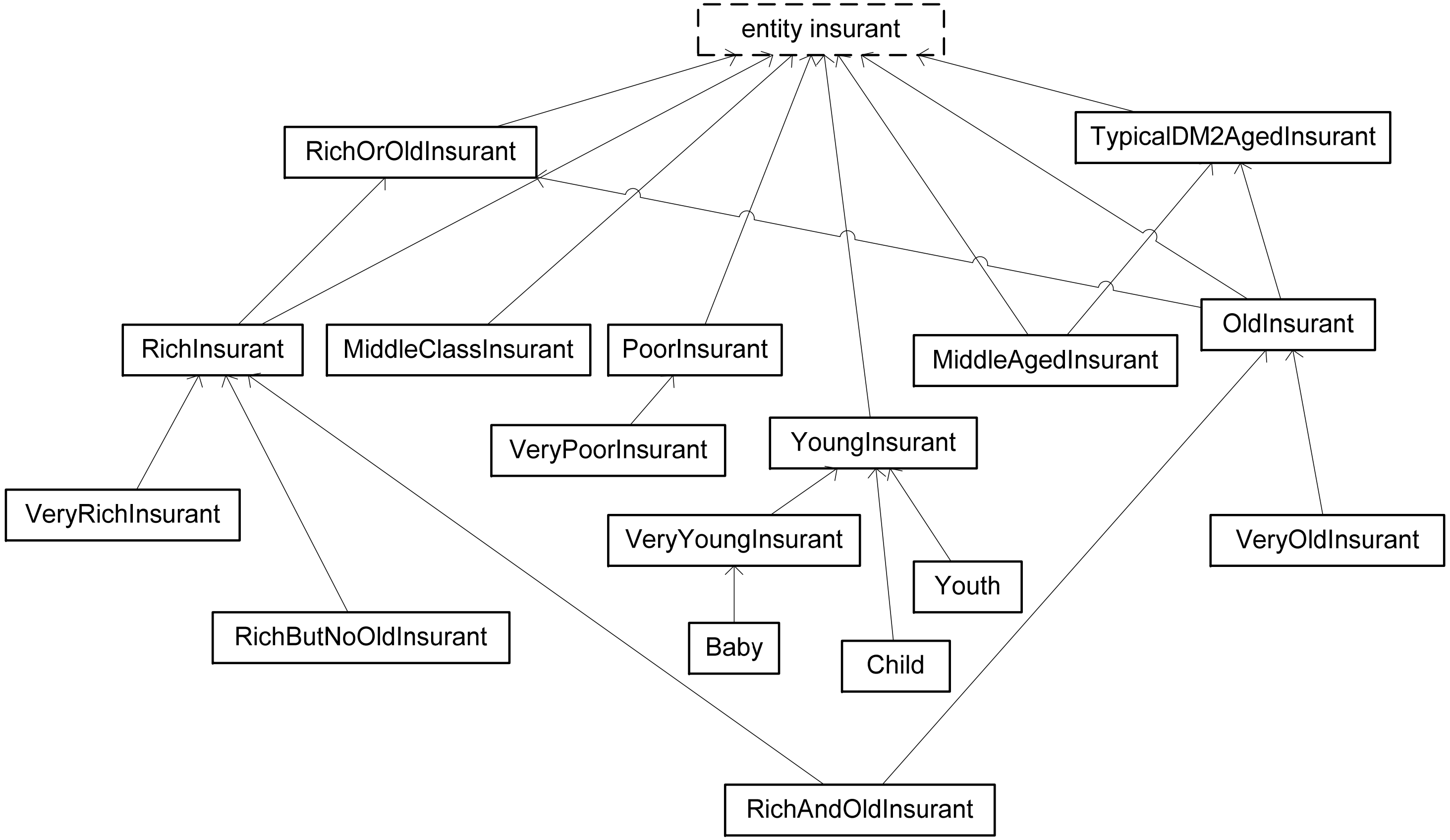
MDO Concepts

- Defined by: Signature + concept expression
- Signature: Set of individuals for which concept is defined
- Concept expression: Boolean expressions over “properties” of
  1. Entities (entity concept)
  2. Nodes of a dimension (dimensional concept)
  3. Points of a dimension space (md-concept)
  4. Pairs of points (comparative md-concept)
- For later use in
  - □ Definitions of measures and scores
  - □ Specifications of groups of interest & comparison (via parameters, explained later) in measure & score applications

Example: Entity Concepts of e_district



Example. Entity Concepts of e_insurant



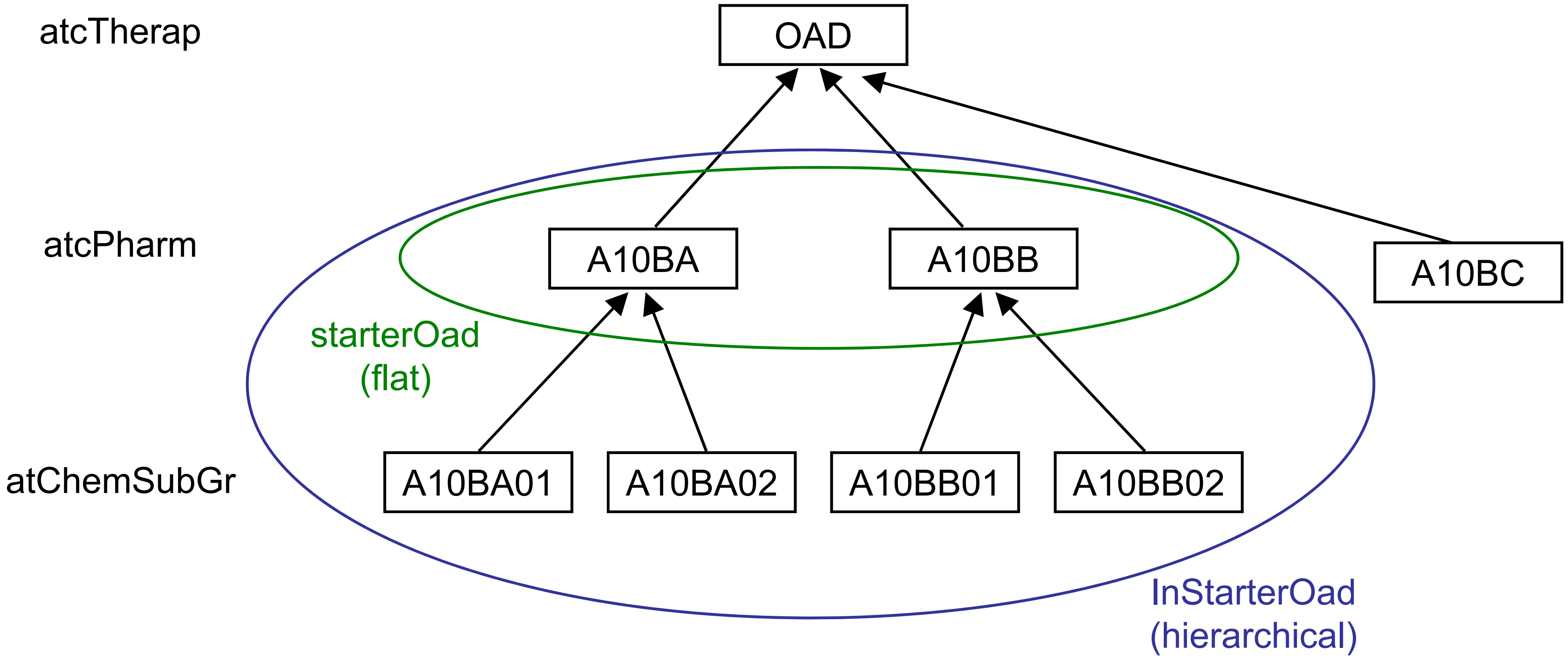
Dimensional Concepts

- Signature
  - Dimension
  - Level (flat concept) or level range (hierarchical concept)
    Naming convention: flat with small initial, hierarchical with big or ‘In’
- Concept Expression
  - entity concept reference
  - hierarchical expansion
  - level or level-range restriction
  - union, intersection, complement (Prequisite: same level ranges)
  - <node> concept [levelOrLevelRange] – expression
  - SQL built-in

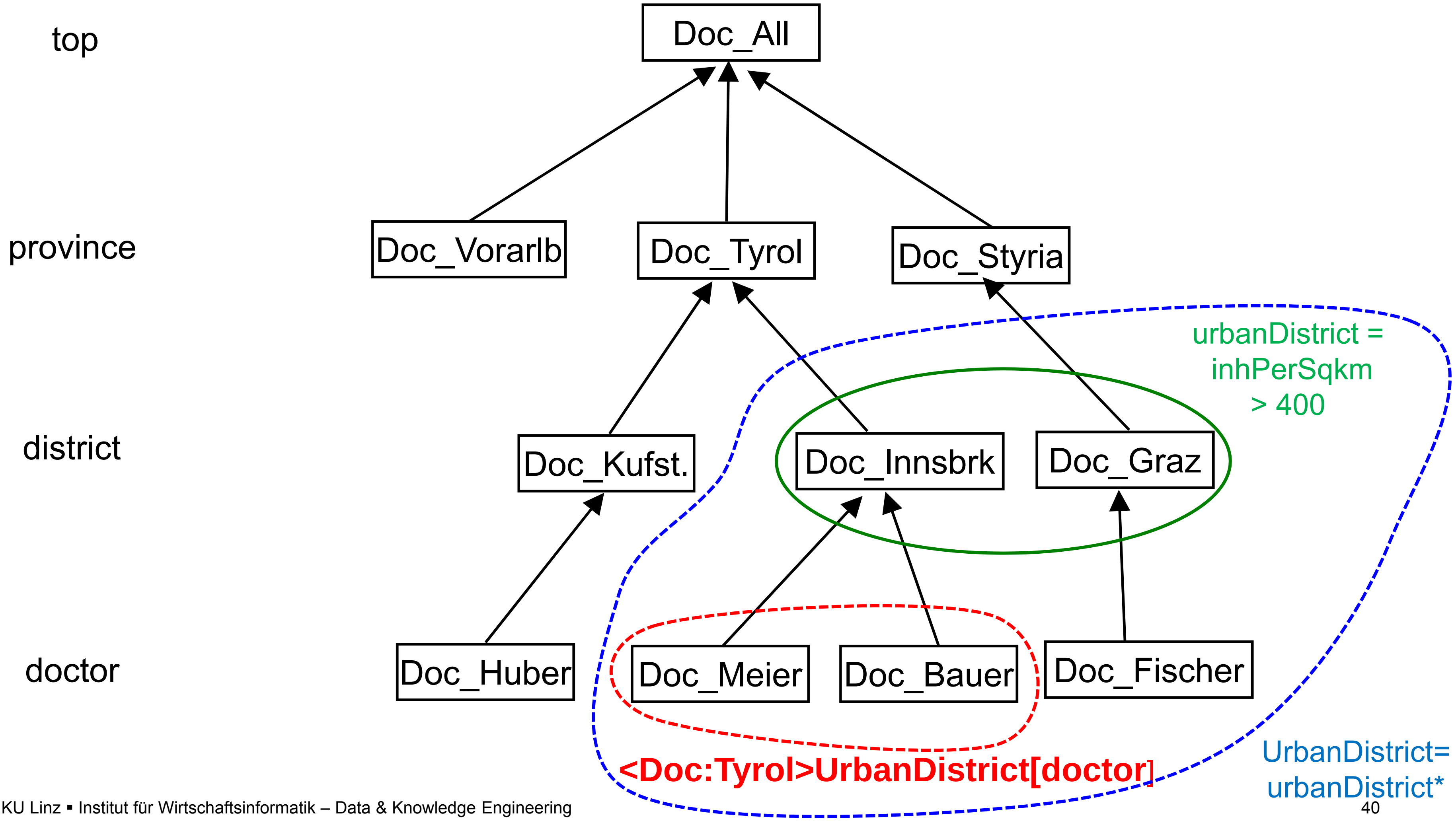
Hierarchical Concept

- Motivation:
  - Make concepts available for multiple levels / granularities
  - Avoid need for roll-ups of nodes or points in use of concepts
- The general principle of hierarchical concepts
  - Individuals organized in a hierarchy (semi-lattice)
  - *Hierarchy property*: If an individual is in the interpretation of a hierarchical concept, all descendants are in the interpretation as well.
- Definition:
  - By hierarchical expansion: *

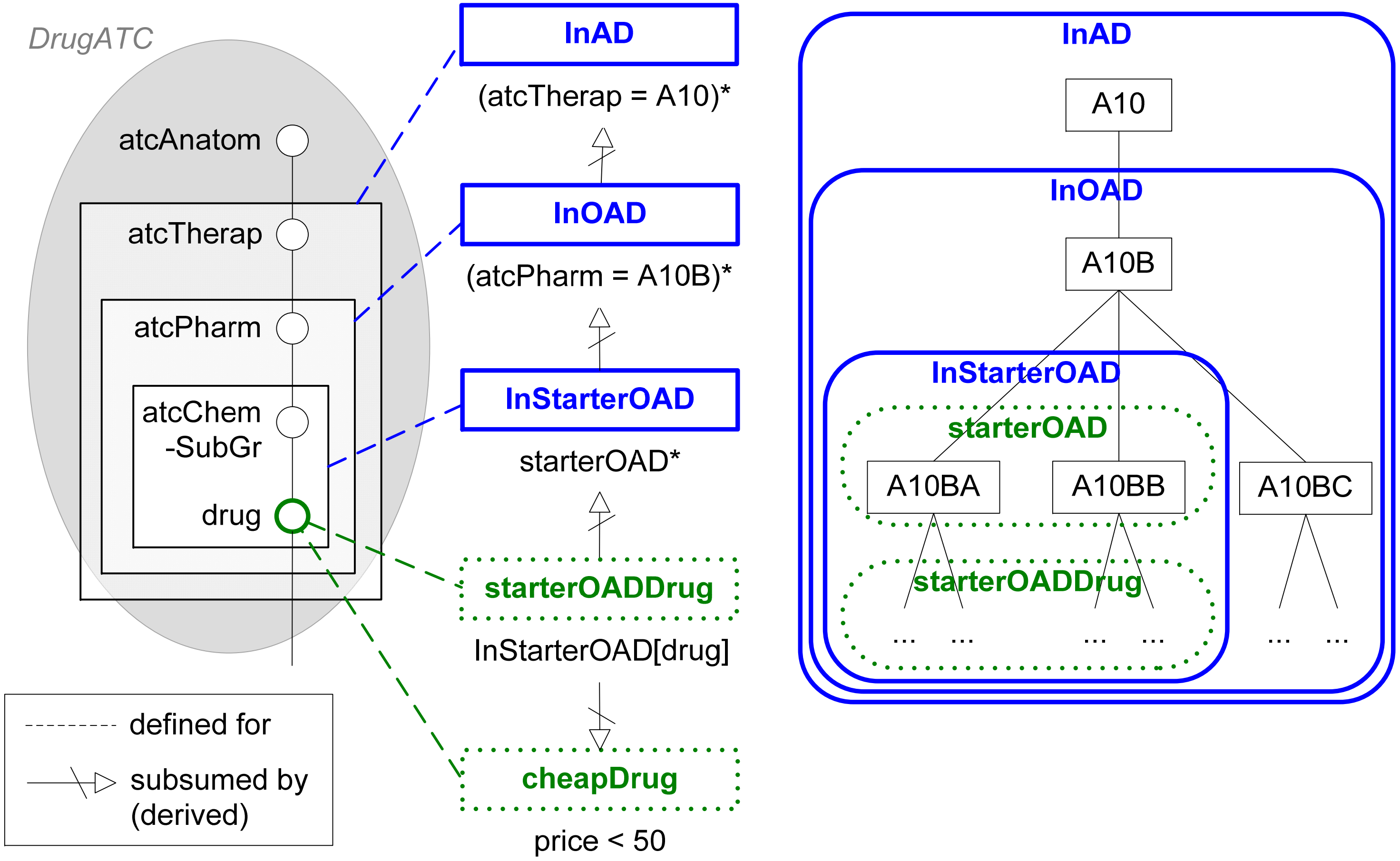
Example: Flat vs. Hierarchical Concept



Example: Node & Level Restriction



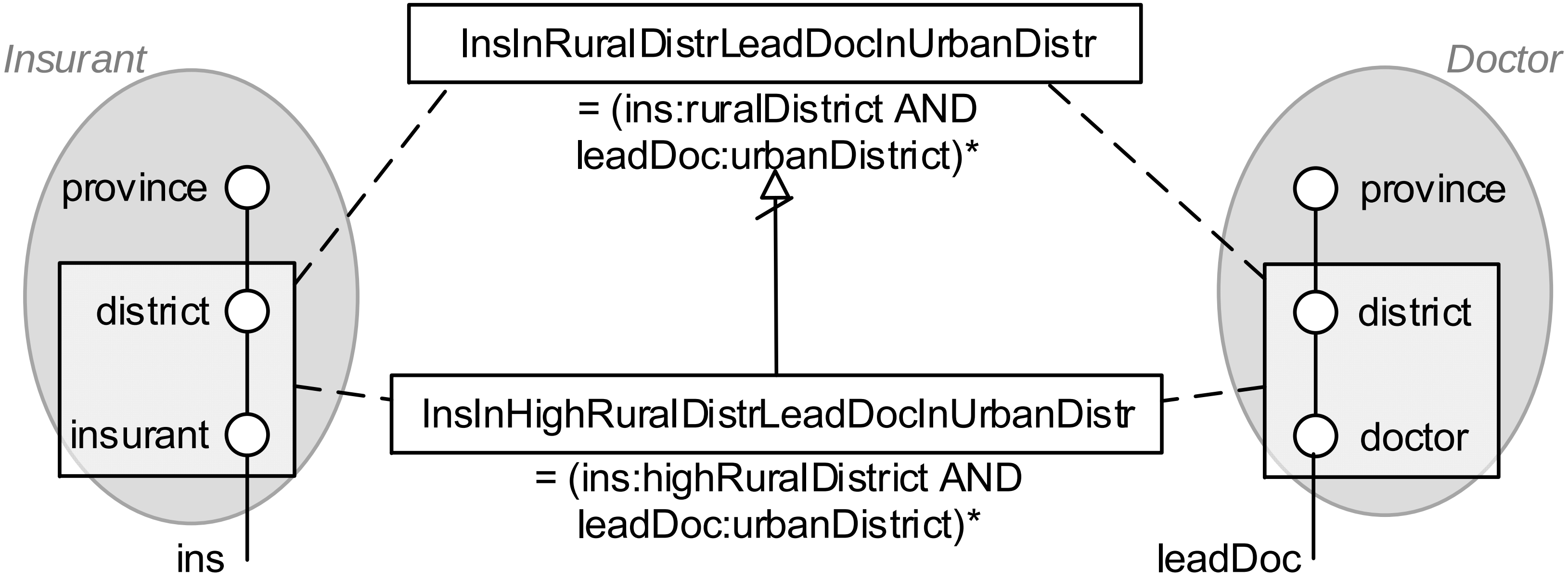
Example: Dimensional Concepts



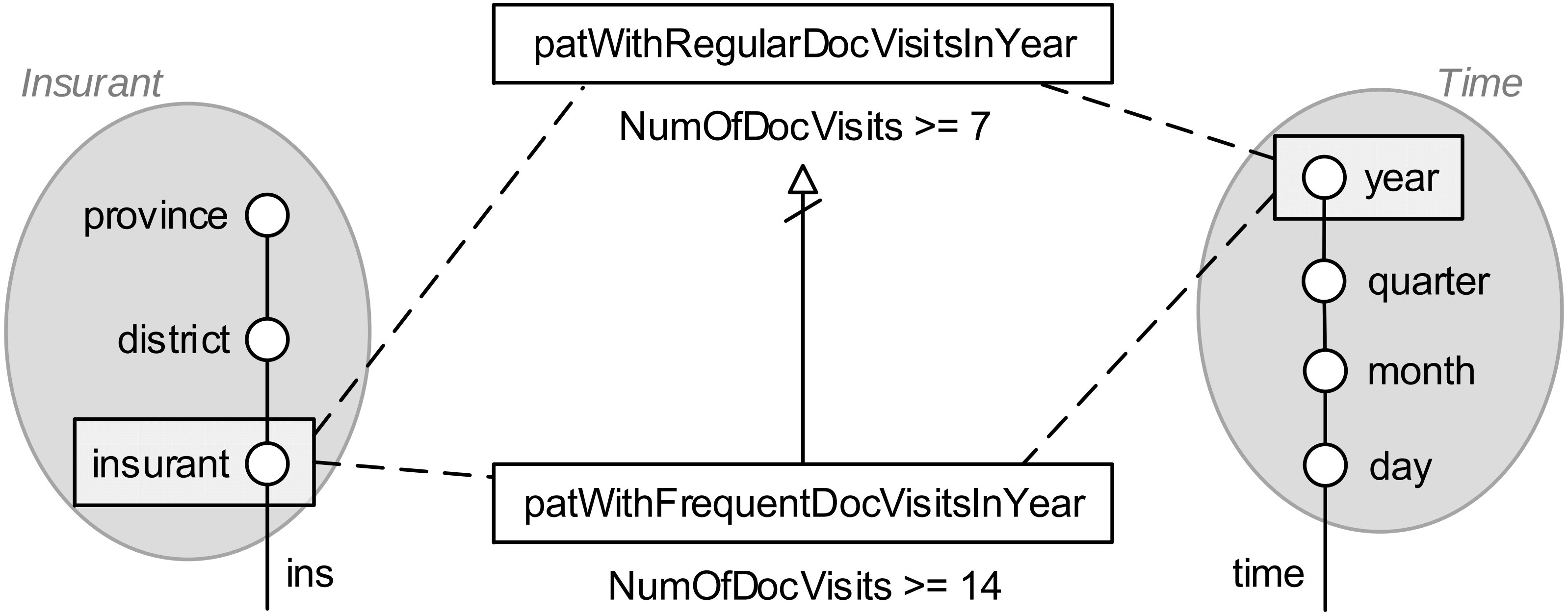
Multi-Dimensional Concepts (md-Concepts)

- Signature (or „domain“ of concept)
  - Dimension space (set of dimension roles + granularity/range)
- Concept Expression
  - dimensional concept reference
  - hierarchical expansion
  - granularity or granularity-range restriction
  - union, intersection, complement (Prereq.: same granurarity/range)
  - <point> concept [granulartiyOrRange] – expression
  - fact-based
  - SQL-built-in

Example: Multi-Dimensional Concepts


Insurant
InsInRuralDistrLeadDocInUrbanDistr
= (ins:ruralDistrict AND leadDoc:urbanDistrict)*
Doctor
province
district
insurant
ins
InsInHighRuralDistrLeadDocInUrbanDistr
= (ins:highRuralDistrict AND leadDoc:urbanDistrict)*
province
district
doctor
leadDoc

Example: Fact-Based md-Concepts



Corner-stone concept expression: <d>(c)[g]

- <dice-point>(external-slice-cond)[roll-up-granularity/range]
- A point p is in the interpretation iff
  - p rolls-up to the dice-point d (if specified)
  - p satisfies the external slice condition c (if specified)
  - p is at the indicated granularity g or within the indicated granularity range g
- Used later to
  - Define cube space (i.e., points of a cube)
  - To define or query a cube
- Analogous for comparison (see later)

Ex.: <p>(c)[g]-expression

- <actDoc:GP, time:2012>(ins:Youth)[time:month]
- <actDoc:GP, time:2012,leadDoc:all,ins:all,drugAtc:all>
(ins:Youth)
[time:month,leadDoc:medSection,ins:top,drugAtc:top]

Note: For simplicity, shorthand notation used with missing dimension roles – relative to considered dimension space – in point assumed to be "all"; missing dimension roles in granularity assumed to be at level of point in <p>(c)[g]-expression (supporting tool completes missing info by default at frontend)

Context-Specific Concepts

- Problem:
  - Concepts have different definitions for different nodes / points
  - Ex.: Who is regarded to be a regular patient of a doctor in some year? (Motivation: only those should be considered in analysis)
- Context-specific concepts
  - Signature is extended by a “context specification” (indicating a subset of nodes of a dimension or points of a dimension space)
  - Context is specified by a concept, typically in form of a <p>(c)[g]-expression

Example: Context-Specific Concept

- Context-specific concept: *regularPatient* (Who counts as regular patient of an acting GP in a rural district in a year?)
- Signature
  - Dimension Space: [ins:insurant, actDoc:doctor, time: year]
  - Context: @<actDoc: GP>(actDoc:RuralDistrict)[actDoc:doctor]
- Concept Expression:
  - noOfvisits ≥ 3

Contextualized Concepts

- Aims and issues
  - Polymorphic use of a concept signature (like methods in object-oriented systems)
  - Inheritance and overriding along concept hierarchies
  - Consistency checking: unique definition
- Approach:
  - Contextualized concepts defines a signature (without concept expression)
  - Several context-specific concepts of the same name provide different concept expressions for different nodes / points in the domain of the contextualized concept

Example: Contextualized Concept

- Contextualized concept: *regularPatient* (Who counts as regular patient of a doctor in a year?)
- Signature: [patLoc:insurant,actDoc:doctor,time:year]
- Context-specific concepts.
  - □ regularPatient@<actDoc:GP>(actDoc:RuralDistrict)[actDoc:doctor] := noOfvisits ≥ 3
  - □ regularPatient@<actDoc:Oculist>[actDoc:doctor]:= noOfvisits ≥ 2

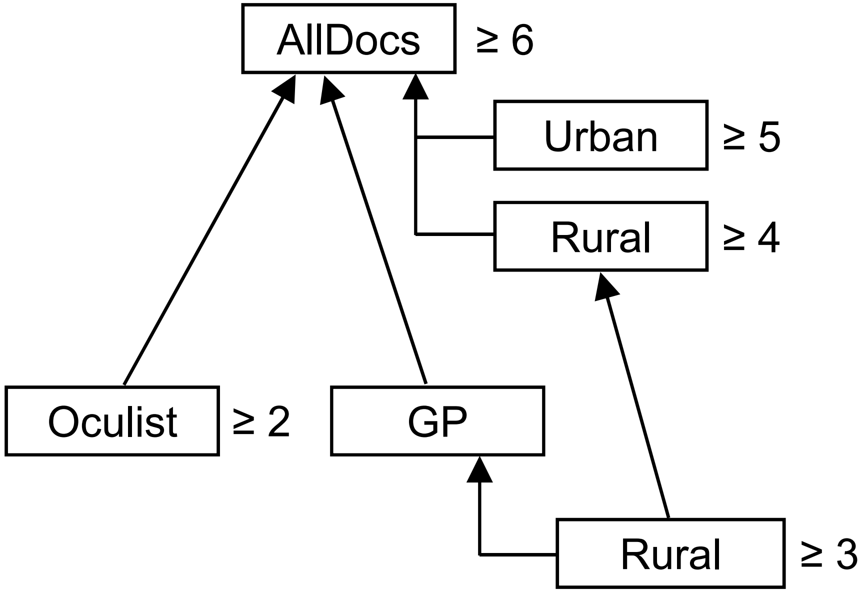
Context-Specific Contexts: Unique Definition

- Problem
  - □ Identify relevant “context concept” for concept evaluation
- General approach (like for “cooperation contracts”)
  - □ Use most specific “context concept”
  - □ Inconsistency: If more than one most specific contexts exists
- Check for global unique definition of concept
  1. Consider subsumption hierarchy of “context concepts”
  2. Apply “abstract super class rule” from oo-modelling:
     For each inner context concept A with sub contexts A1 to An add a context concept A0 = A \ (A1 U … U An)
  3. Check if no two leaf nodes “overlap”, i.e., concepts are disjoint

Context-Specific Concepts: Evaluation

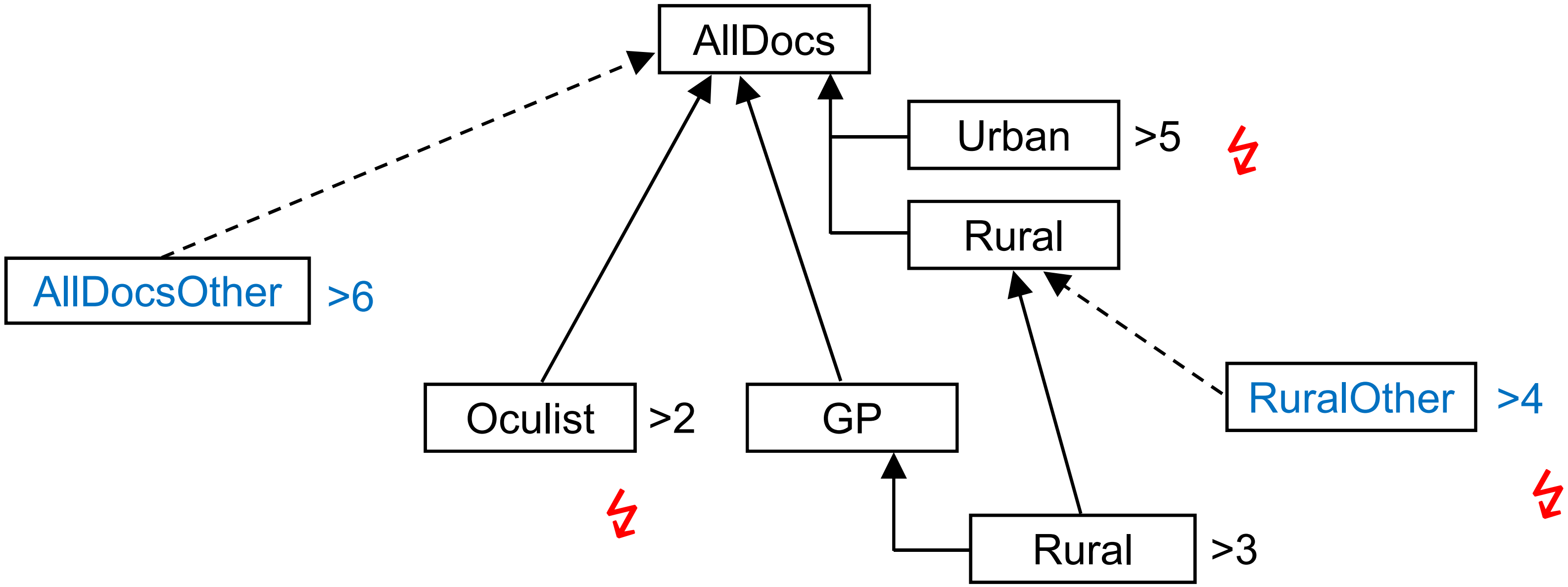
- Aim
  - □ Polymorphic use: 1 concept name
  - □ Evaluate concept by a single predicate (or SQL view as built-in)
- Approach
  1. Ensure unique definition
  2. For each inner context concept A change c@A to c@A0
  3. Let X be the set of contexts for which c is defined:
     Define c as union over x in X: c@x.

Example: Context-Specific Concepts



Number of contacts to count as regular patient of a doctor in a year

Example: Context-Specific Concepts



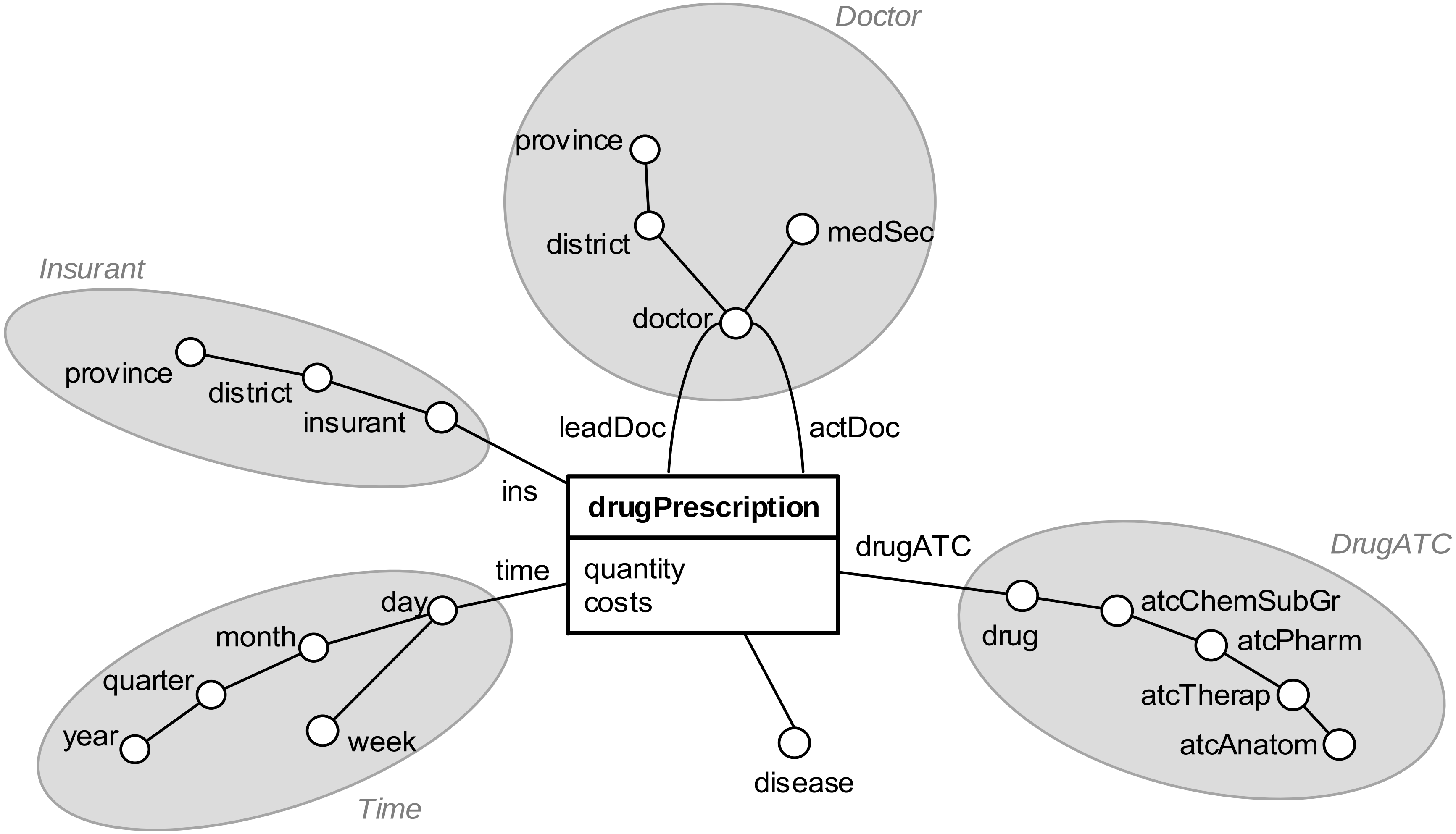
Comparative Concepts

- Signature: Comparative dimension space (2 dimension spaces)
  - group-of-interest (GoI) dimension space
  - group-of-comparison (GoC) dimension space
- Concept expression:
  - Basic form of expression:
    - md-concept for GoI, md-concept for GoC
    - join-condition: relating points of GoI and GoC by some comparison function (e.g., equality, next, previous)
  - Optionally fact-based: Boolean expression of score-value comparisons
  - Plus operators known for md-concepts, such as union, …, built-in

Semantic Dimensions

- Reuse existing terminologies of domain, e.g, SNOMED
  - Terminology: subsumption hierarchy of concepts (OWL)
  - Facts refer to these concepts in “semantic dimension”
  - Meaning “c only” if non-leaf concept c is referred to
- Use of semantic dimension [ER2012]
  - existing concepts are considered nodes of dimension hierarchy
  - subsumption hierarchy is interpreted as “rollsUp”
  - new concepts may be defined upon existing ones (post-coordination)
  - levels may be introduced “locally” (summarizability checked)
  - pre-coordinated concepts can be used like “nodes”
  - post-coordinated concepts can be used like “mdo-concepts”

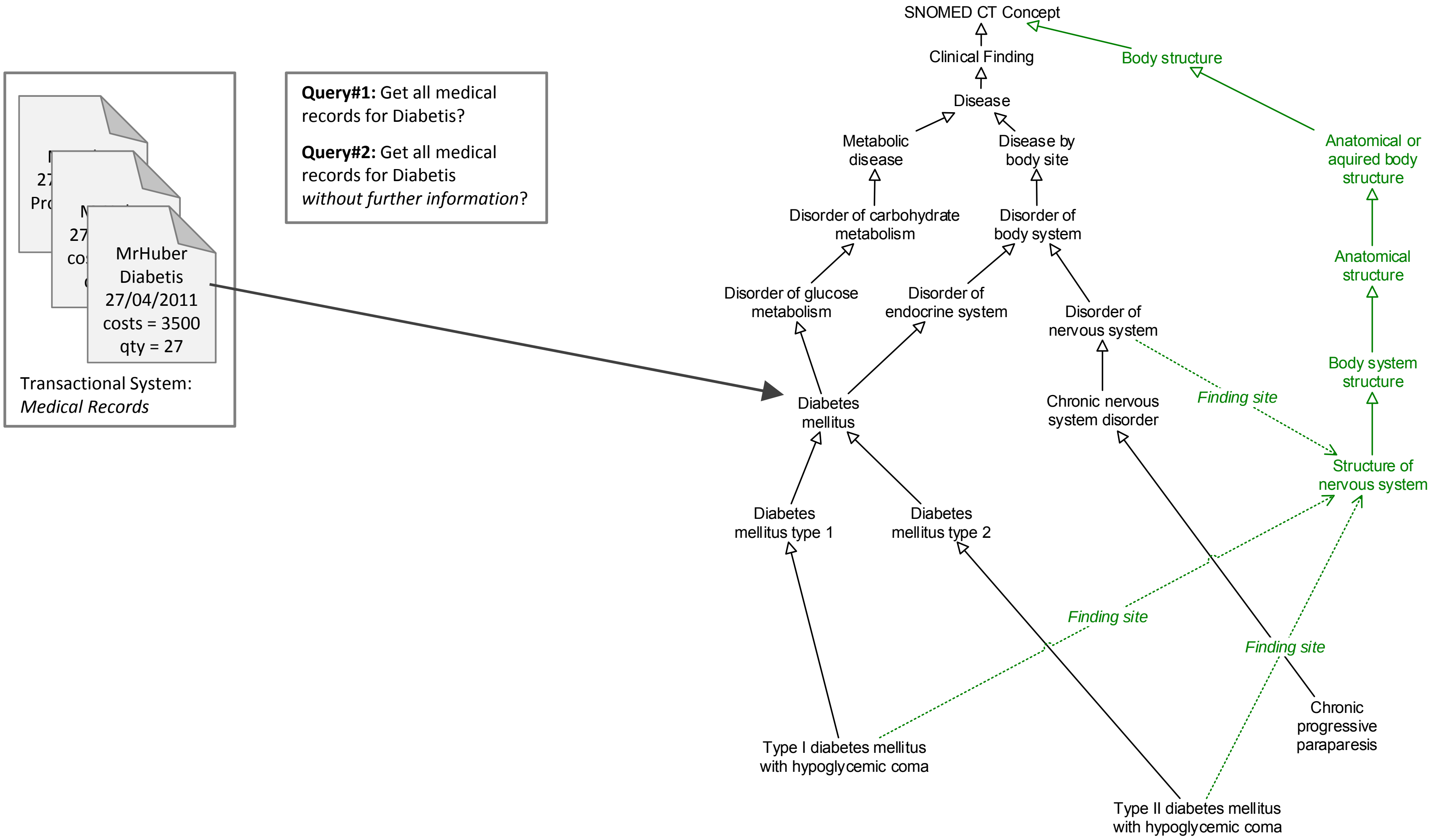
Example: DM2-Analysis (ÖOGKK, simplified)



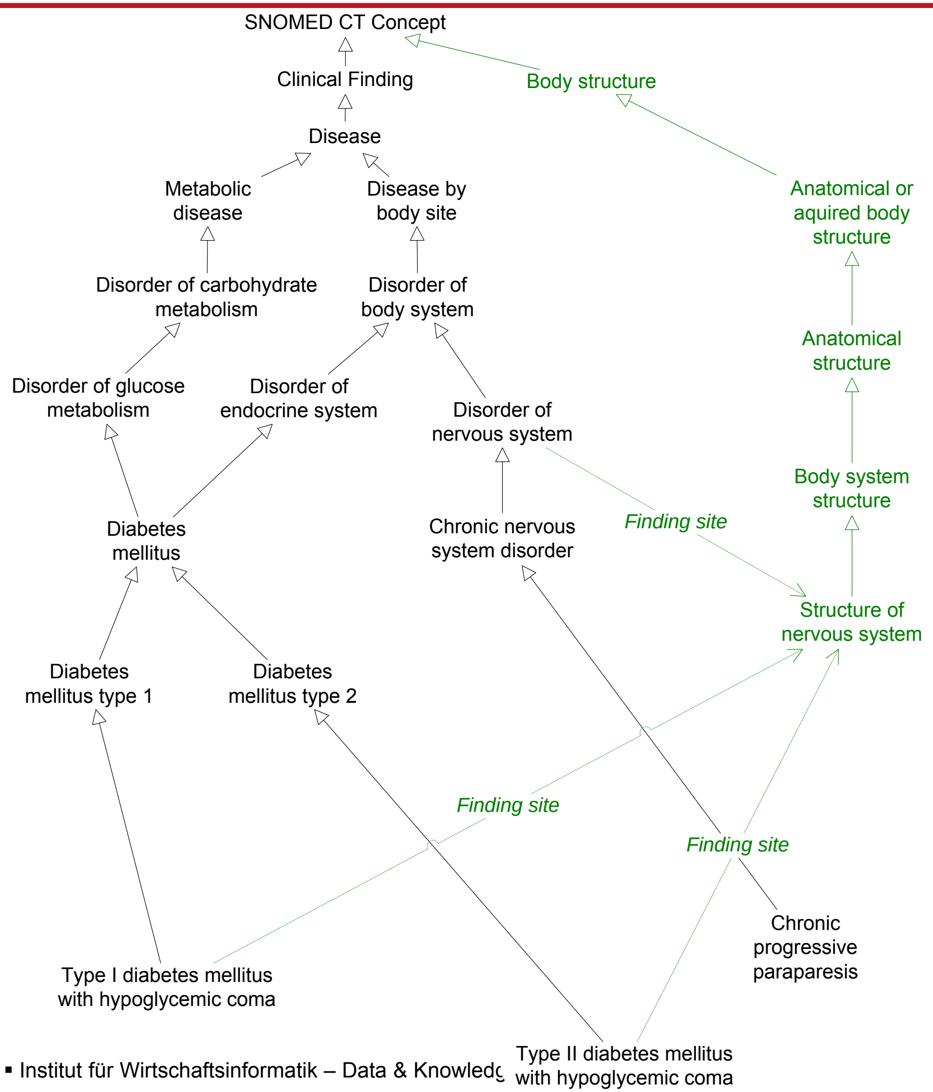
Example: Semantic Attribute

Query#1: Get all medical records for Diabetis?

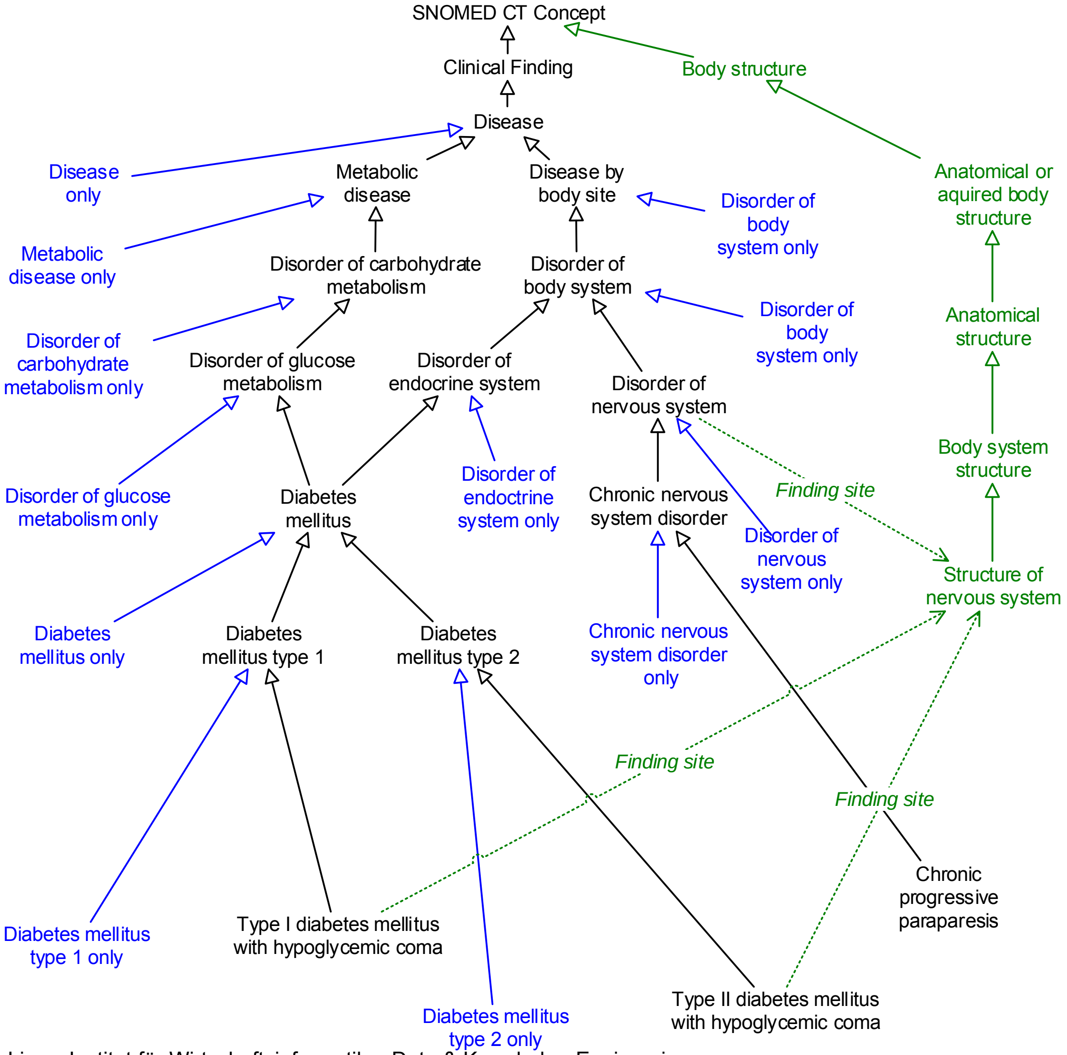
Query#2: Get all medical records for Diabetis *without further information*?


MrHuber
Diabetis
27/04/2011
costs = 3500
qty = 27
Transactional System:
Medical Records
SNOMED CT Concept
Clinical Finding
Body structure
Disease
Metabolic disease
Disease by body site
Anatomical or aquired body structure
Disorder of carbohydrate metabolism
Disorder of body system
Anatomical structure
Disorder of glucose metabolism
Disorder of endocrine system
Disorder of nervous system
Body system structure
Finding site
Diabetes mellitus
Chronic nervous system disorder
Structure of nervous system
Diabetes mellitus type 1
Diabetes mellitus type 2
Finding site
Finding site
Chronic progressive paraparesis
Type I diabetes mellitus with hypoglycemic coma
Type II diabetes mellitus with hypoglycemic coma

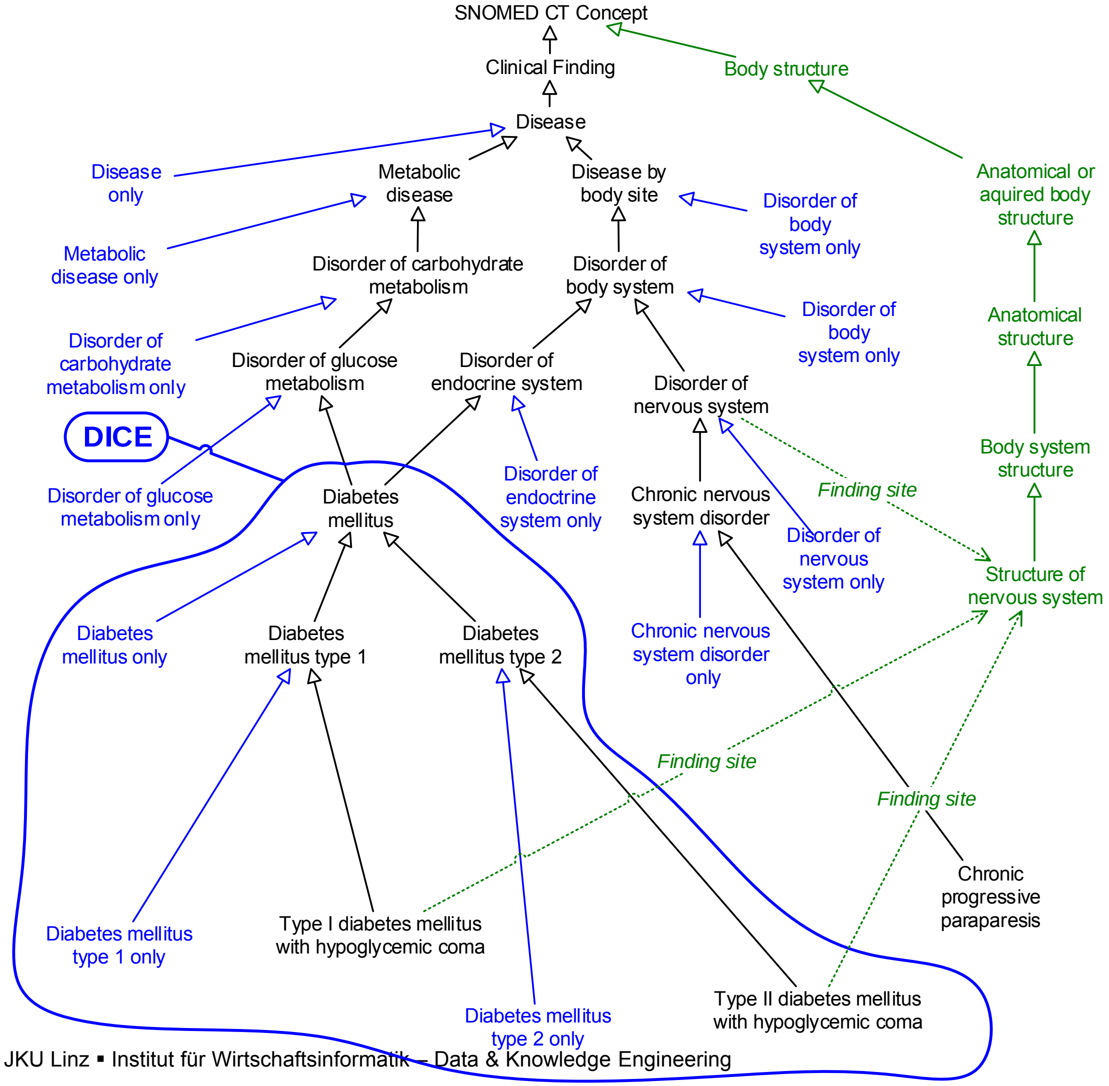
Ex.: SNOMED ontology (selection)


SNOMED CT Concept
Clinical Finding
Body structure
Disease
Metabolic disease
Disease by body site
Anatomical or aquired body structure
Disorder of carbohydrate metabolism
Disorder of body system
Anatomical structure
Disorder of glucose metabolism
Disorder of endocrine system
Disorder of nervous system
Body system structure
Finding site
Diabetes mellitus
Chronic nervous system disorder
Structure of nervous system
Diabetes mellitus type 1
Diabetes mellitus type 2
Finding site
Finding site
Chronic progressive paraparesis
Type I diabetes mellitus with hypoglycemic coma
Type II diabetes mellitus with hypoglycemic coma

Ex.: Semantic Dimension augmented by “-only”-nodes

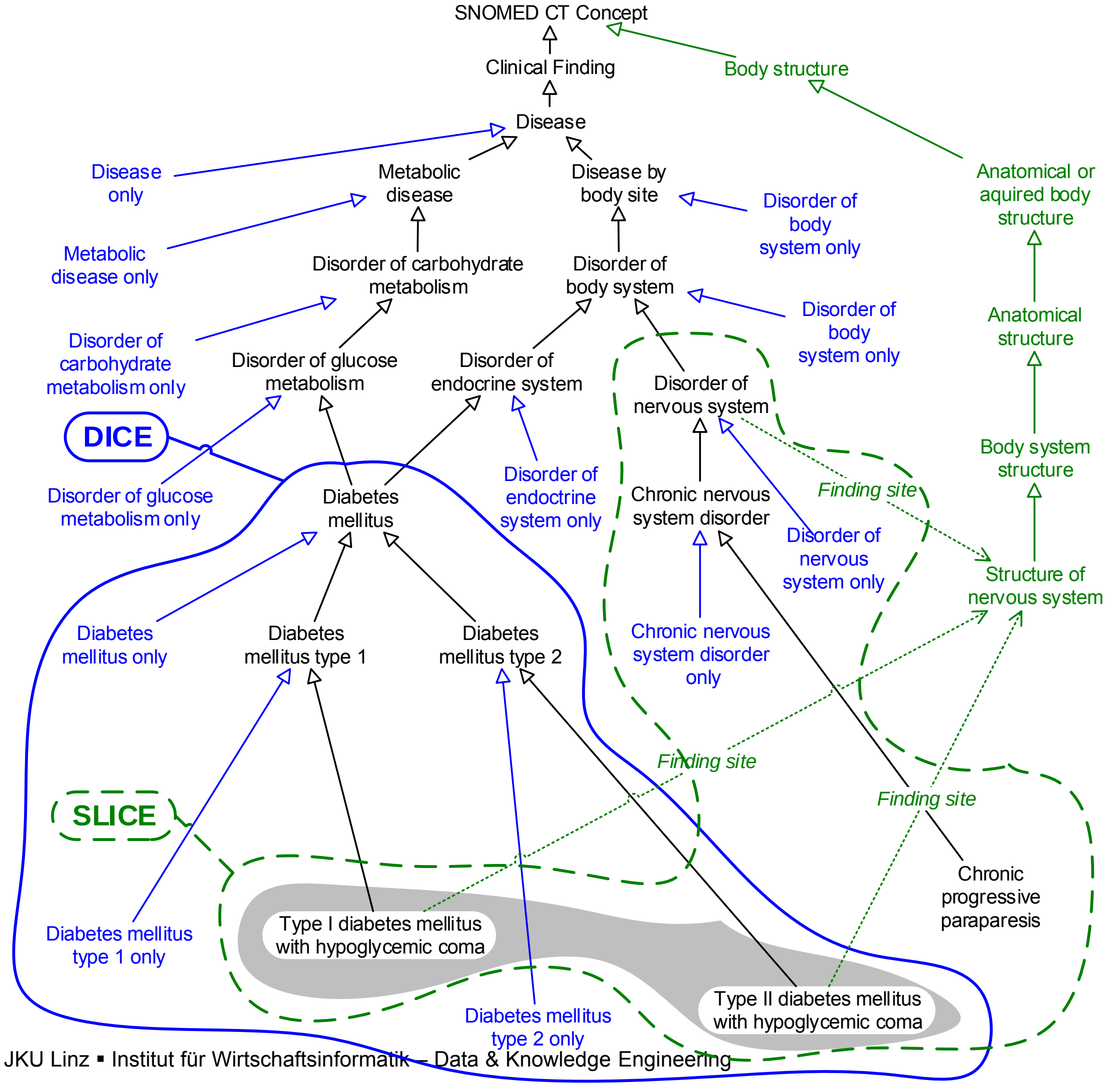

SNOMED CT Concept
Clinical Finding
Body structure
Disease
Disease only
Metabolic disease
Disease by body site
Disorder of body system only
Anatomical or aquired body structure
Metabolic disease only
Disorder of carbohydrate metabolism
Disorder of body system
Disorder of body system only
Anatomical structure
Disorder of carbohydrate metabolism only
Disorder of glucose metabolism
Disorder of endocrine system
Disorder of nervous system
Body system structure
Disorder of glucose metabolism only
Diabetes mellitus
Disorder of endoctrine system only
Chronic nervous system disorder
Finding site
Disorder of nervous system only
Structure of nervous system
Diabetes mellitus only
Diabetes mellitus type 1
Diabetes mellitus type 2
Chronic nervous system disorder only
Finding site
Finding site
Diabetes mellitus type 1 only
Type I diabetes mellitus with hypoglycemic coma
Chronic progressive paraparesis
Diabetes mellitus type 2 only
Type II diabetes mellitus with hypoglycemic coma

Ex.: DM


SNOMED CT Concept
Clinical Finding
Body structure
Disease
Disease only
Metabolic disease
Disease by body site
Disorder of body system only
Anatomical or aquired body structure
Metabolic disease only
Disorder of carbohydrate metabolism
Disorder of body system
Disorder of body system only
Anatomical structure
Disorder of carbohydrate metabolism only
Disorder of glucose metabolism
Disorder of endocrine system
Disorder of nervous system
DICE
Body system structure
Finding site
Disorder of glucose metabolism only
Diabetes mellitus
Disorder of endocrine system only
Chronic nervous system disorder
Disorder of nervous system only
Structure of nervous system
Diabetes mellitus only
Diabetes mellitus type 1
Diabetes mellitus type 2
Chronic nervous system disorder only
Finding site
Finding site
Chronic progressive paraparesis
Diabetes mellitus type 1 only
Type I diabetes mellitus with hypoglycemic coma
Type II diabetes mellitus with hypoglycemic coma
Diabetes mellitus type 2 only

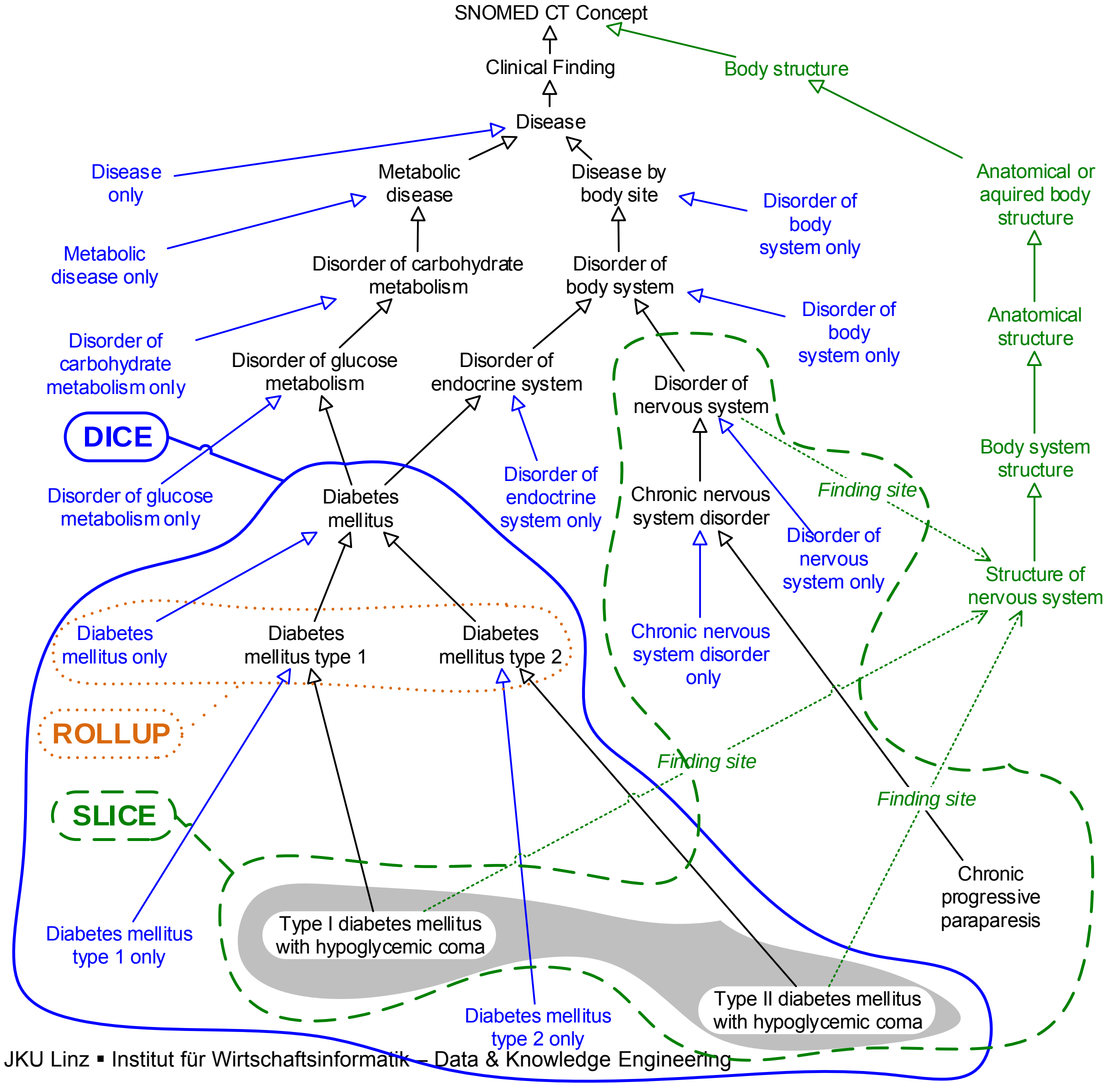
<disease: Diabetes_mellitus>

Ex.: DM found in „nervous system”


SNOMED CT Concept
Clinical Finding
Body structure
Disease
Disease only
Metabolic disease
Disease by body site
Disorder of body system only
Anatomical or aquired body structure
Metabolic disease only
Disorder of carbohydrate metabolism
Disorder of body system
Disorder of body system only
Anatomical structure
Disorder of carbohydrate metabolism only
Disorder of glucose metabolism
Disorder of endocrine system
Disorder of nervous system
DICE
Disorder of endocrine system only
Finding site
Body system structure
Disorder of glucose metabolism only
Diabetes mellitus
Chronic nervous system disorder
Disorder of nervous system only
Structure of nervous system
Diabetes mellitus only
Diabetes mellitus type 1
Diabetes mellitus type 2
Chronic nervous system disorder only
Finding site
Finding site
SLICE
Chronic progressive paraparesis
Diabetes mellitus type 1 only
Type I diabetes mellitus with hypoglycemic coma
Diabetes mellitus type 2 only
Type II diabetes mellitus with hypoglycemic coma

<disease: Diabetes_mellitus>
(Finding_site: some Structure_of_nervous_system)

Ex.: DM found in „nervous system“, by DM1 and DM2


SNOMED CT Concept
Clinical Finding
Body structure
Disease
Disease only
Metabolic disease
Disease by body site
Disorder of body system only
Anatomical or aquired body structure
Metabolic disease only
Disorder of carbohydrate metabolism
Disorder of body system
Disorder of body system only
Anatomical structure
Disorder of carbohydrate metabolism only
Disorder of glucose metabolism
Disorder of endocrine system
Disorder of nervous system
DICE
Disorder of glucose metabolism only
Diabetes mellitus
Disorder of endocrine system only
Chronic nervous system disorder
Finding site
Disorder of nervous system only
Body system structure
Structure of nervous system
Diabetes mellitus only
Diabetes mellitus type 1
Diabetes mellitus type 2
Chronic nervous system disorder only
ROLLUP
Finding site
Finding site
SLICE
Chronic progressive paraparesis
Diabetes mellitus type 1 only
Type I diabetes mellitus with hypoglycemic coma
Type II diabetes mellitus with hypoglycemic coma
Diabetes mellitus type 2 only

<disease: Diabetes_mellitus>
(Finding_site: some Structure_of_nervous_system)
[disease: {Diabetes_mellitus_only, Diabetes_mellitus_type_1, Diabetes_mellitus_type_2}]

Summarizability: Disjointness and completness (with respect to dice point) of listed concepts used as “roll-up level instances” checked by reasoner

Unifying Native & Semantic Dimensions

- Nodes:
  - Native nodes: correspond to „entity“
  - Concept nodes: correspond to „entity concept“ (domain concept)
- Levels:
  - Native level: consists of native nodes (from given dimensional-fact model)
  - Concept level: consists of concept nodes
- Overall picture:
  - Native level relates to entity class (maybe an external ontology)
  - May add concept levels (for concepts of the same entity class) above, and local to some other concept node
  - Which entity concepts (domain concepts) are included as concept nodes is a design decision; nodes are potential entries in „data cube“

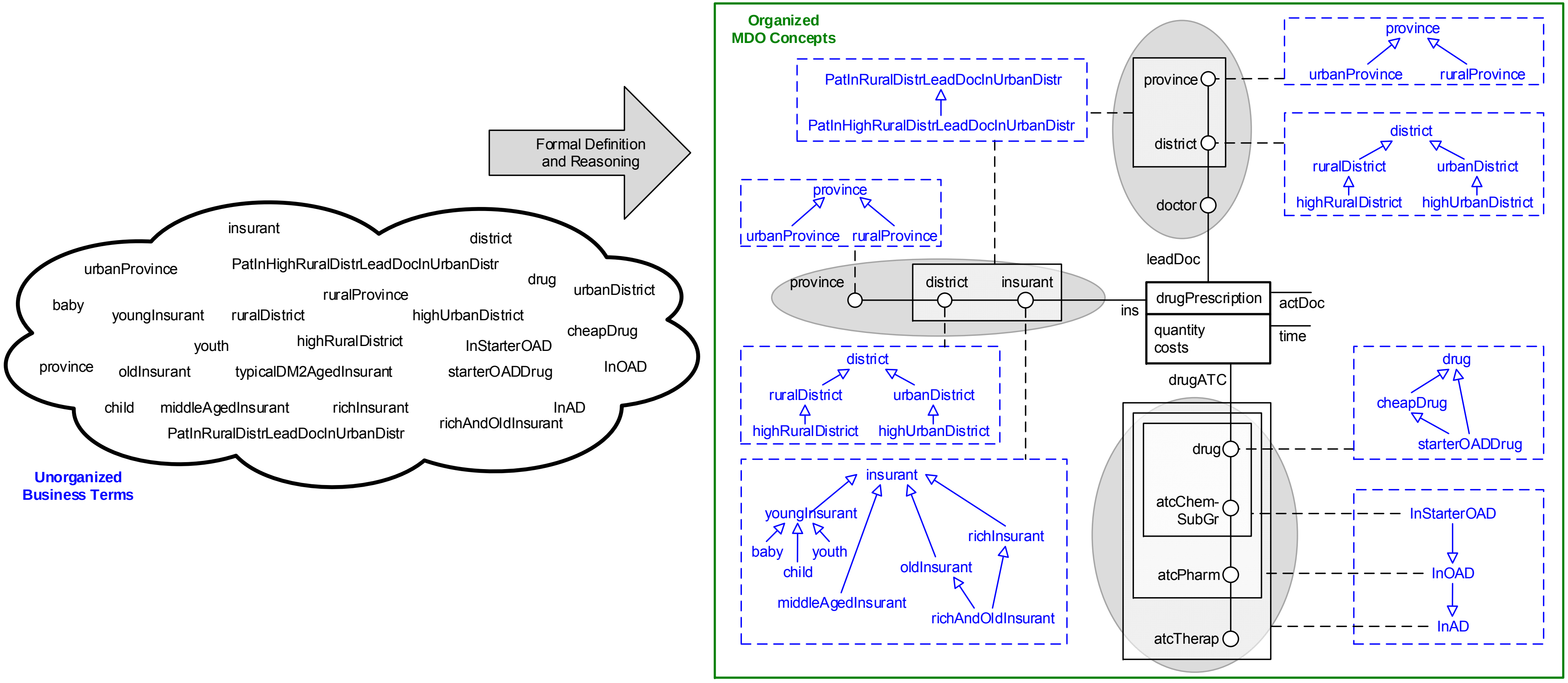
Subsumption Hierarchies and Reasoning

- Aim: Subsumption hierarchy ($\sqsubseteq$) of concepts
  - Guide users in narrowing analysis along concept hierarchies
- Mapping to OWL: open world reasoning, md-concepts + individuals (used in concept definitions) + domain concepts
  - Fact-based concepts and built-ins considered as “primitive”
  - Limited expressiveness: *rollsUpTo** cannot be “functional”
  - Workaround: redundant “rollsUpToLevel*_*X*”
- Mapping to SQL: Closed world reasoning for selected dimensions (time, drugAtc); “subset checking”, built-ins considered

Example: MDO concept in OWL

- MDO: InsInRuralDistrictLeadDocInUrbanDistr[leadDoc:doctor] (see above).
- OWL: ∃ins.∃rollsUpTo_district.∃entity.ruralDistrict ∏ ∃leadDoc.(∃atLevel.{doctor} ∏ ∃rollsUpTo_district. ∃entity.urbanDistrict)

Example: Concept Organisation



MDO concept in SQL: Relational View

- Predefined views over DWH:
  - □ dimRole*X*rollUp(dimRole*X*,dimRole*X*Sup) *Note:transtive,reflexiv rollUp*
  - □ dimRole*X*atLevel(dimRole*X*,level)
  - □ dimSpace*X*RollUp(dimRole1,dimRole1Sup,.. dimRolek,dimRolekSup)
  - □ ….
- MDO-concepts as “concept views”, naming conventions
  - □ entity-concept(entityClassName)
  - □ dimConceptName(dimName)
  - □ mdConceptName(dimRole1Name,…,dimRolekName)
  - □ compConcName(GoI_dimRole1Name,.., GoI_dimRolekName, GoC_dimRole1Name,…GoC_dimRolekName)
- MDO-concepts as SQL-built-in:
  - □ Use predefined-views and concept views to define new concept view

Example: MDO to SQL (1)

- Mapping rule for *INTERSECTION OF:*
  - □ MDO: INTERSECTION OF (C1, C2)
  - □ SQL: SELECT * FROM C1 NJ C2

- *Example:*
  - □ MDO:
    INTERSECTION OF (insRuralDistrict, leadDocUrbanDistrict)
  - □ SQL:
    SELECT * FROM insRuralDistrict NJ leadDocUrbanDistrict

NJ … NATURAL JOIN

Example: <p>(c)[g] in SQL (2)

- <dim1: n1, … dimk: nk>(concept)[dim1:l1,…dimk:lk]
  - SQL-Interpretation:

    ```
    SELECT * FROM
      (SELECT dim1,..,dimk
      FROM dim1RollUp NJ …. dimnRollUp
      WHERE dim1Sup=´n1´ AND … dimkSup=´nk´) NJ
      concept NJ
      (SELECT dim1,…, dimk
      FROM dim1RollUp NJ dim1AtLevel … NJ dimkAtLevel
      WHERE dim1AtLevel.level=`l1`  AND …. =`lk`)
    ```

3. Ontology-based Measures & Scores

- Md-concepts used
  - to select the data to be included in the calculation of derived measures & scores
  - to define the context of context-specific measures & scores
- Generic measures & scores
  - to avoid repeated definitions of similar measures & scores
  - „generic parameters“ for md-concepts
- Templates for measure & score definitions, and SQL-builtIns
- Applied to point or set of points (defined by md-concepts)
  - <point>.measure or <point>(concept)[granularity]..measure
  - (PntGoI,PntGoC).score, …
- Measure and scores represented by measure and score views

Measure & Score Definitions

Measure

- Signature
  - Set of points for which measure is defined
  - Indicated by (granularity-restricted) dimension space
- Instructions
  - Function: point to value
  - Pre-defined templates
  - Built-in (by SQL view)

Score

- Signature
  - Set of pairs of points for which score is defined
  - Indicated by (granularity-restricted) comparative dimension space
- Instructions
  - Function: 2 points to value
  - Pre-defined templates
  - Built-in (by SQL view)

Measure and score application

- *md-concept.measure*
  - □ SQL-interpretation:
    SELECT measure
    FROM concept NATURAL
    JOIN measure_view
- *concept..measure*
  - □ SQL-Interpretation:
    SELECT *
    FROM concept NATURAL
    JOIN measure_view

measure_view (data cube):

contains each point for which measure value exists + measure value

- *comparativeConcept.score*
  - □ SQL-interpretation:
    SELECT score
    FROM comparativeConcept
    NATURAL JOIN score_view
- *compConcept..score*
  - □ SQL-interpretation
    SELECT *
    FROM comparativeConcept
    NATURAL JOIN score_view

score_view (data cube):

contains each point-pair (flattened) for which score value exits + score value

Ex.: Sample Measure / Score Applications

- <ins:Tyrol,time:2011>.drugCostsOfRuralIns
- <ins:Austria,time:2011>[ins:province].drugCostsOfRuralIns
- (<actDoc:Vienna,ins:NÖ,time:2010>(drugAtc:InOAD)[time:month] (GoI.time=GoC.time) <actDoc:NÖ,ins:Vienna,time:2010>(drugAtc:InOAD)[time:month] ..ratioOfTotalDrugCosts

Note: For simplicity, shorthand notation used with missing dimension roles – with respect to signature of measure/score – in point assumed to by "all"; missing dimension roles in granularity assumed to be at level of point in <p>(c)[g]-expression (supporting tool completes missing info by default at frontend)

Measure & Score Instructions: Templates

Measures

- Simple Arithmetic
  - +,-, (,), *, / over md-point
- Aggregation
  - Base measure m
  - Qualifier q (a concept)
  - Aggregation function f
- 2-Step Aggregation
  - Roll-up granularity g
  - First level aggr. measure m

Scores

- Ratio, MeanPercentileRank
  - GoI base measure mi
  - GoC base measure mc
- 2 Step: Aggregation + Score
  - Roll-up granularity g
  - GoI aggr. measure mi
  - GoC aggr. measure mc

Example : Aggregation Measure

Aggregation Measure:	**drugCostsOfRuralIns**
Signature:	drugPrescriptionSpace
Measure:	drugPresciptionCosts
Qualifier:	ins:RuralDistrict
Aggregation function:	sum

MDO Notation:

```
sum(<self>(ins:RuralDistrict).. drugPrescriptionCosts)
```

SQL-Interpretation (measure view)

```
SELECT dimRole1Sup,…dimRoleKSup, sum(drugPrescriptionCosts)
FROM drugPrescriptionCosts_view NJ “ins:RuralDistrict”
NJ drugPrescriptionSpaceRollUp
GROUP BY dimRole1Sup,…, dimRolekSup
```

Example: 2-Step Aggregation Measure

2-Step-Aggregation:	**avgDrugCostsPerRuralIns**
Signature:	DrugPrescriptionSpace[ins:insurant..top]
Roll-up granularity:	(ins:insurant,leadDoc:doctor,time:month,drugAtc:drug)
Measure:	drugCostsOfRuralIns
Aggregation function:	avg

- MDO notation:

  avg(<self>DrugPrescriptionSpace[ins:insurant, leadDoc:doctor, time:month, drugAtc:drug]..drugCostsOfRuralIns)

- Sample application:

  <ins:Austria,leadDoc:GP,time:2012,drugAtc:all>..avgDrugCostsPerRuralIns

Note: Granularity indications omitted, if not restricted

Example Score: Ratio

Ratio:	**costRatioOfInsurantsMigration**
Signature:	DrugPrescriptionCompSpace
GoI measure:	drugCostsOfRuralIns
GoC measure:	drugCostsPrescByRuralDoctor

- MDO notation:
  - RATIO(<GoI.self>.drugCostsOfRuralIns, <GoC.self>.drugCostsPrescByRuralDoctor)
- Sample application:
  - (<actDoc:Vienna,time:2010>,<ins:Vienna,time:2010>).r
  - Rslt:= (<actDoc:Vienna,time:2010>.drugCostsOfRuralIns / <ins:Vienna,time:2010>.drugCostsPrescByRuralDoctor)

Generic Measures & Scores

- Aims
  - Avoid repeated definitions of same calculations
  - Provide for flexibility in measure & score use
  - Enable reasoning by providing common structures
- Parameters for measures & scores
  - Set on “instantiating” generic measure
- Generic measure
  - Qualifier as generic parameter
- Generic score
  - Parameter for qualifier of generic measure of GoI
  - Parameter for qualifier of generic measure of GoC

Example: Generic Aggregation Measure

Aggregation Measure:	**totalCosts(%q)**
Signature:	DrugPrescriptionCompSpace
Measure:	drugPrescriptionCosts
Qualifier:	%q
Aggregation function:	sum

- MDO notation:
  - sum(<self>(%q)..drugPrescriptionCosts)
- Sample application and interpretation:
  - <ins: Tyrol,actDoc: GP,time: 2012>.totalCosts(ins:RuralDistrict)
  - totalCosts(ins:RuralDistrict) == drugCostsOfRuralIns *(see before)*

Example: Aggregation-AvgPercentileRank

AggregationAvgPercentileRank:	**MnPctLRkCostsPerIns(%qi,%qt) s**
Signature:	DrugPrescriptonCompSpace[ins:insurant … top]
Roll-up granularity:	ins:insurant
GoI measure:	totalCosts(%qi) tc
GoC measure:	totalCosts(%qc) tc

Sample application and interpretation:

- (<actDoc:Tyrol,time:2011>,<actDoc:Austria,time:2011>. s(%qi=(ins:RuralDistrict,%qc=(actDoc:UrbanDistrict))
- percentileRank (avg (<actDoc:Tyrol,time:2011>[ins:insurant]..tc(actDoc:RuralDistrict), <actDoc:Austria,time:2011>[ins:insurant]..tc(actDoc:UrbanDistrict).

Measure & Score Drivers Hierarchy

- Change of value of one score may change value of other score
  - □ Known from definition of measures and scores
  - □ Background knowledge (e.g., relationships of base measures)
- “Influence hierarchy” captured explicitly
  - □ drives(measure1,measure2)

    Eg.: drives(ambulantTreatmentCosts,overallCosts)
  - □ drives(measure, score)
  - □ drives(score1,score2)
- Knowledge used later to guide analysis process

SemCockpit: Built-in Measures & Scores

- Principles
  - Function defined as view table with columns for each dimension role of md-point + measure-value
  - For scores: view table with 2 points + score-value
  - Based on dynamic SQL
  - Point, qualifier, and possible other parameters are %placeholders
  - Cubes, levels, dimension nodes , rollsUp* , mdo-concepts (column names are dimensions) assumed to exist as tables/views (naming conventions: e.g., ‘dimRole:dimConceptName’)
- Use of built-in measures & scores
  - The same way as non-built-ins

4. Comparative OLAP

- Build *cubes* for selected roll-up facts, using mdo-concepts to define points of cubes and to instantiate generic measures
- Build likewise *comparative cubes* for comparative facts
- Define (comparative) *analysis situations* with some elements of a (comparative cube) such as node of a point, level of a granularity, or qualifier, fixed and some elements variable
- *Explorative analysis* („try and see“) by varying variables of (comparative) analysis situations

Cubes and Facts

- Cube defined by “MDO-query”:
  - □ Cube space: <point>(concept)[granularityOrRange]
  - □ List of measures m1(q1),…mn(qn), whose measure domains subsume cube space, with qi instantiated (i=1..n)
  - □ Optionally: m(q)/v to indicate that a NULL-value is to be substituted by v
  - □ Cube contains all facts for which one measure is not NULL
- Fact: (<x1,…xn, v1.. vn)
- Comparative cubes and comparative facts:
  - □ Cube: 2 cube spaces (GoI, GoC) + join condition + score
  - □ Join condition: GoC.Di =GoI.Dj in general: any predefined comparison function / comparative concept

Example: Cube and comparative cube

(<time:2012, ins:UpperAustria>,
<time:2011, ins:UpperAustria>)
..RatioOfDrugCosts(InOAD,InOAD)

(<time:2012, ins:UpperAustria>[ins:district],
GoI:ins:district = GoC:ins:district,
<time:2011, ins:UpperAustria>[ins:district])
..RatioOfDrugCosts(InOAD.InOAD)

Comparative Analysis Situation

Groups of Interest (GoI)	*Groups of Comparison (GoC)*
Point (pntGoI)	Point (pntGoC)
Condition (conceptGoI)	Condition(conceptGoC)
Granularity (granGoI)	Granularity (granGoC)
Join Condition (joinCond)	
Qualifier (qGoI)	Qualifier (qGoC)
Score (%qGoI, %qGoC)	

One comparison per (pntGoI,pntGoC)-pair according to Join Condition.

E.g:
(<actDoc:GP,time: 2012>[actDoc:medSection,time:month], GoI.time=GoC.time
<actDoc:Oculist,time:2012>[actDoc:medSection,time:month])
.ratioAvgDrugCostsPerIns(InOAD,InOAD)

Variation along “semantic relationships”

Correlated Change

- Coordinates of points
  - □ Move down, up, aside, …
- Granularities
  - □ Drill down, roll up, …
- Qualifiers
  - □ Narrow, broaden, aside
- Score
  - □ Refocus
- Dimension
  - □ Split, merge

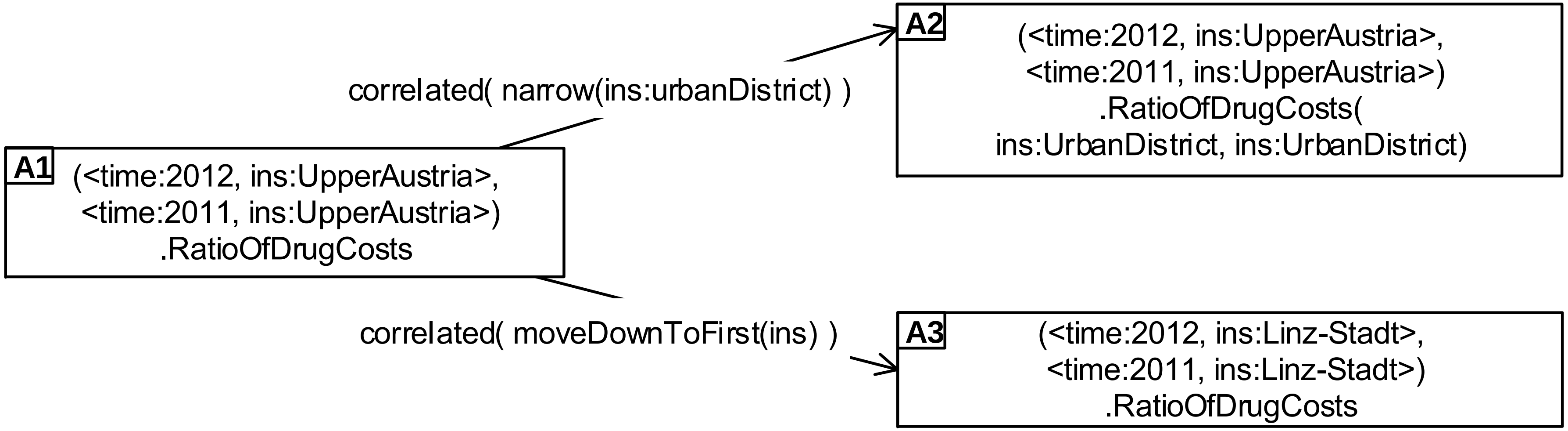
Change of GoI or GoC

- Coordinates of points
- Granularities
- Qualifiers

Change GoI-GoC pairing

- Join conditions
  - □ Shrink, expand, re-join
- External slice
  - □ Filter, extend, shift

Example: Explorative Analysis



A1 (<time:2012, ins:UpperAustria>, <time:2011, ins:UpperAustria>) .RatioOfDrugCosts

correlated(narrow(ins:urbanDistrict))

A2 (<time:2012, ins:UpperAustria>, <time:2011, ins:UpperAustria>) .RatioOfDrugCosts(ins:UrbanDistrict, ins:UrbanDistrict)

correlated(moveDownToFirst(ins))

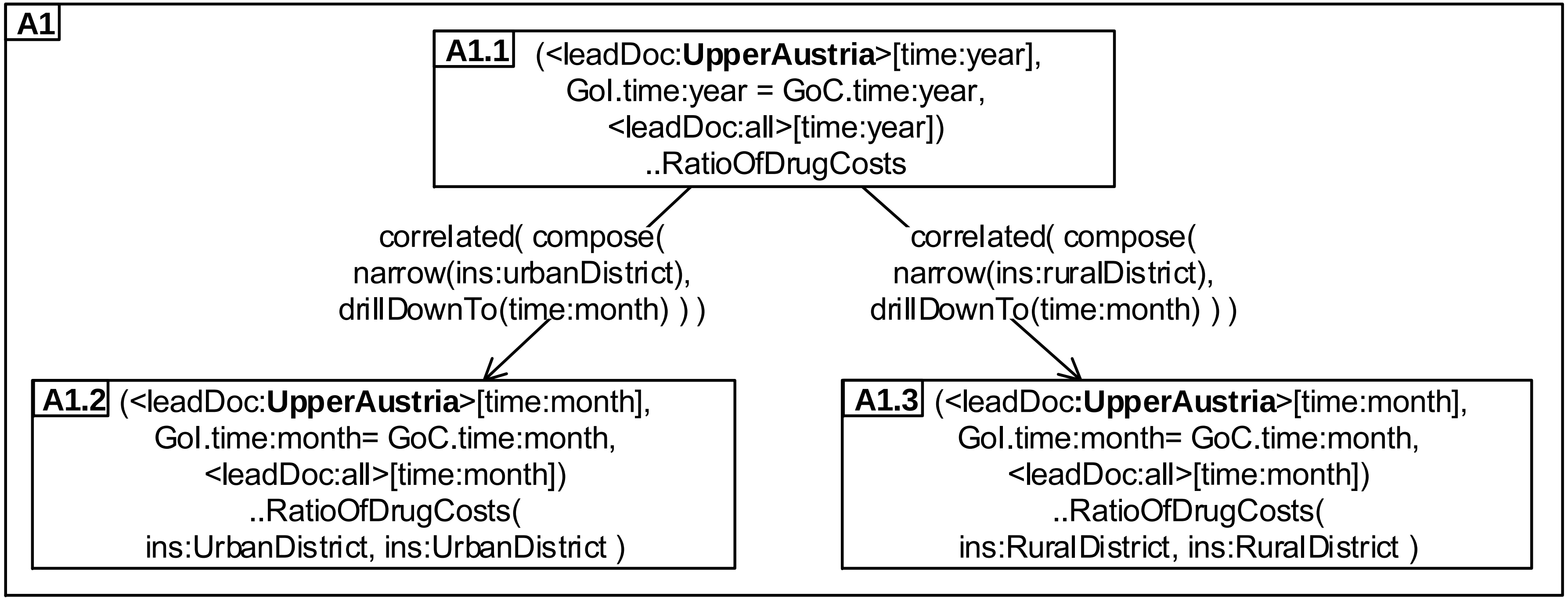
A3 (<time:2012, ins:Linz-Stadt>, <time:2011, ins:Linz-Stadt>) .RatioOfDrugCosts

Composite Analysis Situation

- Consists of multiple analysis and comparative analysis situations (e.g., show scores, drill down to raw measures)
- Linked by semantic relationships (e.g, drill-down to city)
- Relationships form a tree; successors depend on predecessors
- Analysis situations are parameterized
- Changes in parameters propagated down along relationships
- Composite analysis situation may be depicted in one diagram or in multiple dependent diagrams
- Global “picture” of aggregation & details; coherent change

Used in analysis phase (1): Search for measures and groups.

Example: Composite Analysis Situation



A1

A1.1 (<leadDoc:**UpperAustria**>[time:year],
GoI.time:year = GoC.time:year,
<leadDoc:all>[time:year])
..RatioOfDrugCosts

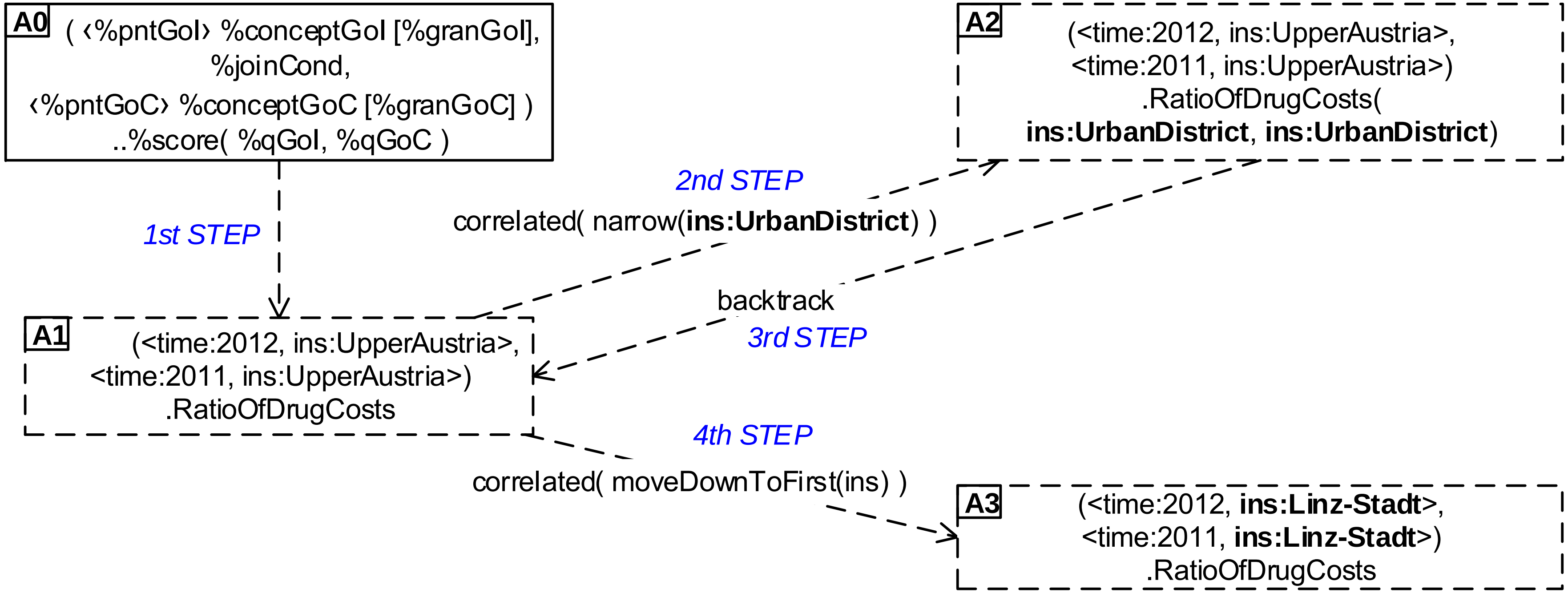
correlated(compose(
narrow(ins:urbanDistrict),
drillDownTo(time:month)))

correlated(compose(
narrow(ins:ruralDistrict),
drillDownTo(time:month)))

A1.2 (<leadDoc:**UpperAustria**>[time:month],
GoI.time:month= GoC.time:month,
<leadDoc:all>[time:month])
..RatioOfDrugCosts(
ins:UrbanDistrict, ins:UrbanDistrict)

A1.3 (<leadDoc:**UpperAustria**>[time:month],
GoI.time:month= GoC.time:month,
<leadDoc:all>[time:month])
..RatioOfDrugCosts(
ins:RuralDistrict, ins:RuralDistrict)

Example: Analysis History



A0 (‹%pntGoI› %conceptGoI [%granGoI],
%joinCond,
‹%pntGoC› %conceptGoC [%granGoC])
..%score(%qGoI, %qGoC)

1st STEP

A1 (<time:2012, ins:UpperAustria>,
<time:2011, ins:UpperAustria>)
.RatioOfDrugCosts

2nd STEP

correlated(narrow(**ins:UrbanDistrict**))

A2 (<time:2012, ins:UpperAustria>,
<time:2011, ins:UpperAustria>)
.RatioOfDrugCosts(
ins:UrbanDistrict, **ins:UrbanDistrict**)

backtrack

3rd STEP

4th STEP

correlated(moveDownToFirst(ins))

A3 (<time:2012, **ins:Linz-Stadt**>,
<time:2011, **ins:Linz-Stadt**>)
.RatioOfDrugCosts

5. BI Analysis Graphs

- Represent promising analysis paths (sequence of OLAP-steps, variations of controls on initial comparative analysis situation)
- Nodes
  - □ Analysis situations or comparative analysis situations
  - □ Parameterized
  - □ Simple or composite analysis situations
- Directed Arcs:
  - □ OLAP navigation between source and target analysis situation
  - □ Express how parameters of source and target analysis situation relate in terms of “semantic relationships” (roll-up, subsumption, but also “sibling”, “next”, …)
  - □ Optionally parameterized

Analysis Graphs: Inspired by WebML [Ceri,et.al.]

WebML

- Content Model
  - Classes & attributes
- Navigation Model: Unit
  - Associated to class
  - Contains object(s) of class
  - Parameterized SQL query over class
- Navigation Model: Links
  - Navigation between units
  - Transport information, navigate relationships

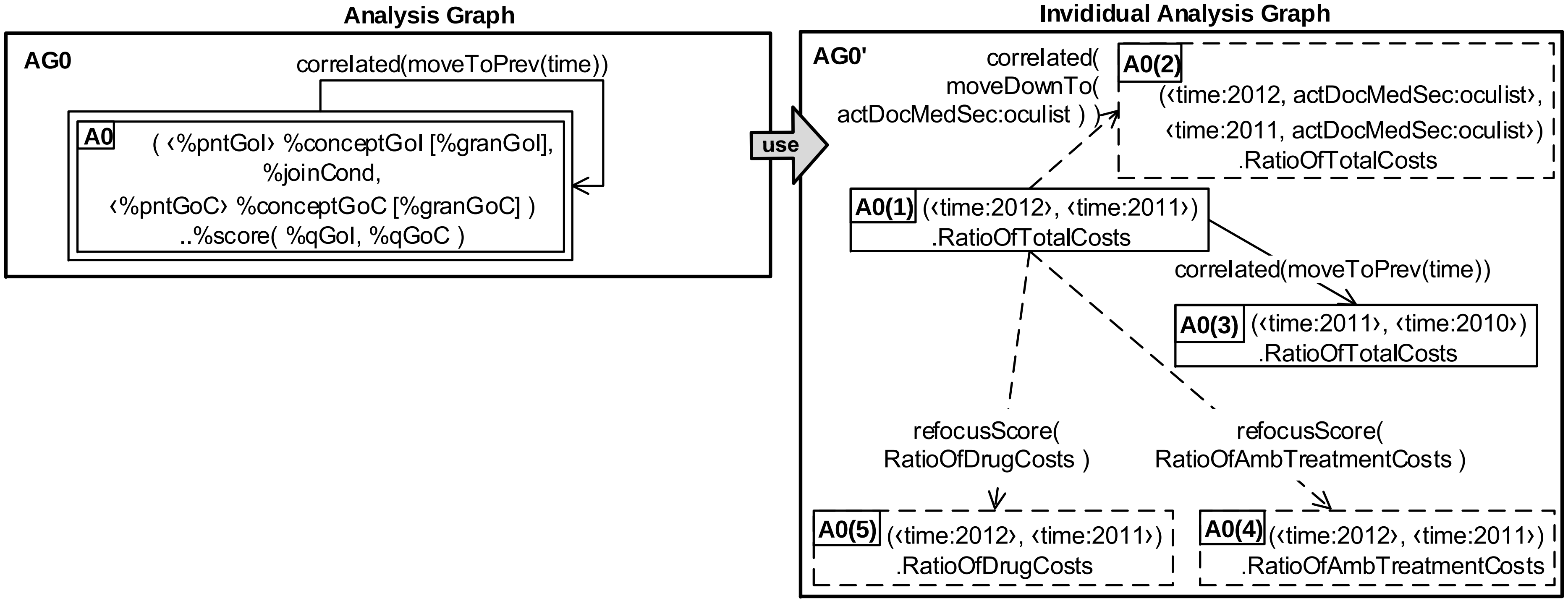
Analysis Graphs (AGs)

- Content Model
  - Cubes & measures
- AGs: Analysis Situation
  - Associated to cube
  - Contains (rollUp) facts
  - Parameterized MDO-query over cube
- AGs: Arcs
  - Navigation between ASs
  - Transport information, navigate “relationships”

Analysis Graphs: Design

- Typical analysis session is characterised by
  1. Determining an initial comparative analysis situation
  2. Visually inspect the result
  3. Modify parameters (usually based on semantic relationships) to move to a new individual analysis situation
  4. Inspect result: (a) stop if satisfied, or (b) continue with (3), or (c) backtrack to previous analysis situation, or (d) explore new steps
- Analysis graphs capture this “analyses process knowledge” for later analysis to provide guidance
  - Design graph incrementally, capturing each promising path
  - Generalize/restrict variable domains, if variation is helpful or not

Example: BI Analysis Session



Analysis Graph

AG0

correlated(moveToPrev(time))

A0 (‹%pntGoI› %conceptGoI [%granGoI], %joinCond, ‹%pntGoC› %conceptGoC [%granGoC]) ..%score(%qGoI, %qGoC)

use

Invididual Analysis Graph

AG0'

correlated(moveDownTo(actDocMedSec:oculist))

A0(2) (‹time:2012, actDocMedSec:oculist›, ‹time:2011, actDocMedSec:oculist›) .RatioOfTotalCosts

A0(1) (‹time:2012›, ‹time:2011›) .RatioOfTotalCosts

correlated(moveToPrev(time))

A0(3) (‹time:2011›, ‹time:2010›) .RatioOfTotalCosts

refocusScore(RatioOfDrugCosts)

refocusScore(RatioOfAmbTreatmentCosts)

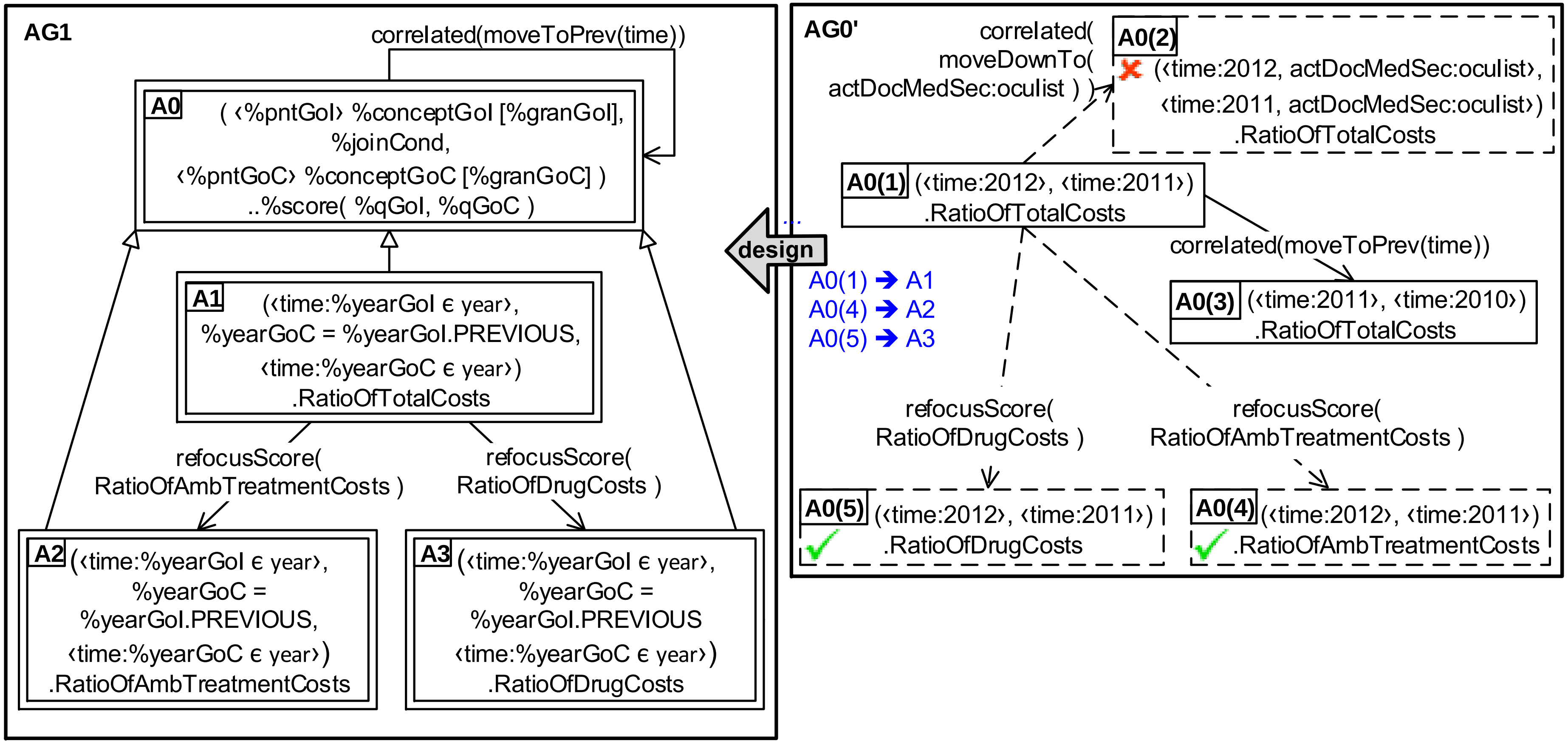
A0(5) (‹time:2012›, ‹time:2011›) .RatioOfDrugCosts

A0(4) (‹time:2012›, ‹time:2011›) .RatioOfAmbTreatmentCosts

Specialization of Analysis Situations

- Based on variable domains of analysis situations:
  - □ Point-variables (pntGoI, pntGoC): md-concept
  - □ Concept-variables (conceptGoI, conceptGoC, qGoI, qGoC): set of concepts, ANY, ^ concept
  - □ Level-variables (granGoI, granGoC): level or range
  - □ Join-condition: set of join-conditions (comparative concepts)
  - □ Score variable: set of scores, ANY, ^ score
- An analysis situation *specializes* another analysis situation, if each variable domain is the same or more restrictive
- Individual analysis situation: all variables bound

Example: BI Analyis Graph (Design)



AG1

correlated(moveToPrev(time))

A0 (‹%pntGoI› %conceptGoI [%granGoI], %joinCond, ‹%pntGoC› %conceptGoC [%granGoC]) ..%score(%qGoI, %qGoC)

A1 (‹time:%yearGoI є year›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC є year›) .RatioOfTotalCosts

refocusScore(RatioOfAmbTreatmentCosts)

refocusScore(RatioOfDrugCosts)

A2 (‹time:%yearGoI є year›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC є year›) .RatioOfAmbTreatmentCosts

A3 (‹time:%yearGoI є year›, %yearGoC = %yearGoI.PREVIOUS ‹time:%yearGoC є year›) .RatioOfDrugCosts

design

AG0'

correlated(moveDownTo(actDocMedSec:oculist))

A0(2) ✗ (‹time:2012, actDocMedSec:oculist›, ‹time:2011, actDocMedSec:oculist›) .RatioOfTotalCosts

A0(1) (‹time:2012›, ‹time:2011›) .RatioOfTotalCosts

correlated(moveToPrev(time))

A0(3) (‹time:2011›, ‹time:2010›) .RatioOfTotalCosts

A0(1) → A1
A0(4) → A2
A0(5) → A3

refocusScore(RatioOfDrugCosts)

refocusScore(RatioOfAmbTreatmentCosts)

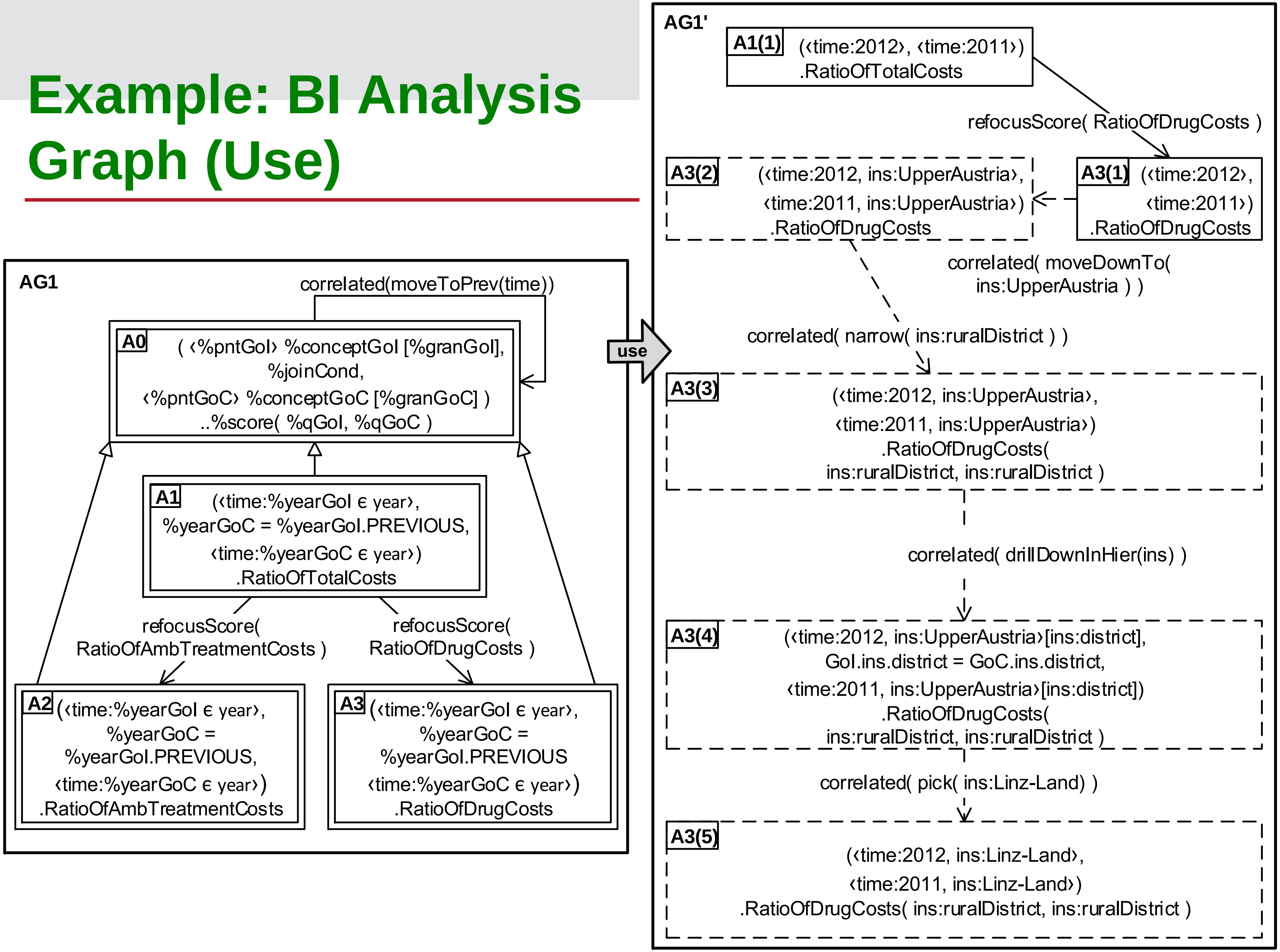
A0(5) ✓ (‹time:2012›, ‹time:2011›) .RatioOfDrugCosts

A0(4) ✓ (‹time:2012›, ‹time:2011›) .RatioOfAmbTreatmentCosts

Analysis Graphs: Use

- Typical use in analysis phase (2): proper analysis (repeated)
  1. Jump to predefined initial analysis situation or any other one
  2. Set open parameters for this analysis situation
  3. Evaluate the analysis situation and inspect the result
  4. Choose an outgoing arc to move to another analysis situation
  5. Set open parameter of the arc, if any
  6. Evaluate & inspect the result; stop if satisfied or continue with 4.
- SemCockpit support
  - Reasoning to check for specilisation of analysis situations and to determine possible values for parameters
  - Guidance rules to open/close paths and choose parameters

Example: BI Analysis Graph (Use)



AG1

correlated(moveToPrev(time))

A0 (‹%pntGoI› %conceptGoI [%granGoI], %joinCond, ‹%pntGoC› %conceptGoC [%granGoC]) ..%score(%qGoI, %qGoC)

A1 (‹time:%yearGoI ϵ year›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC ϵ year›) .RatioOfTotalCosts

refocusScore(RatioOfAmbTreatmentCosts)

refocusScore(RatioOfDrugCosts)

A2 (‹time:%yearGoI ϵ year›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC ϵ year›) .RatioOfAmbTreatmentCosts

A3 (‹time:%yearGoI ϵ year›, %yearGoC = %yearGoI.PREVIOUS ‹time:%yearGoC ϵ year›) .RatioOfDrugCosts

use

AG1'

A1(1) (‹time:2012›, ‹time:2011›) .RatioOfTotalCosts

refocusScore(RatioOfDrugCosts)

A3(1) (‹time:2012›, ‹time:2011›) .RatioOfDrugCosts

A3(2) (‹time:2012, ins:UpperAustria›, ‹time:2011, ins:UpperAustria›) .RatioOfDrugCosts

correlated(moveDownTo(ins:UpperAustria))

correlated(narrow(ins:ruralDistrict))

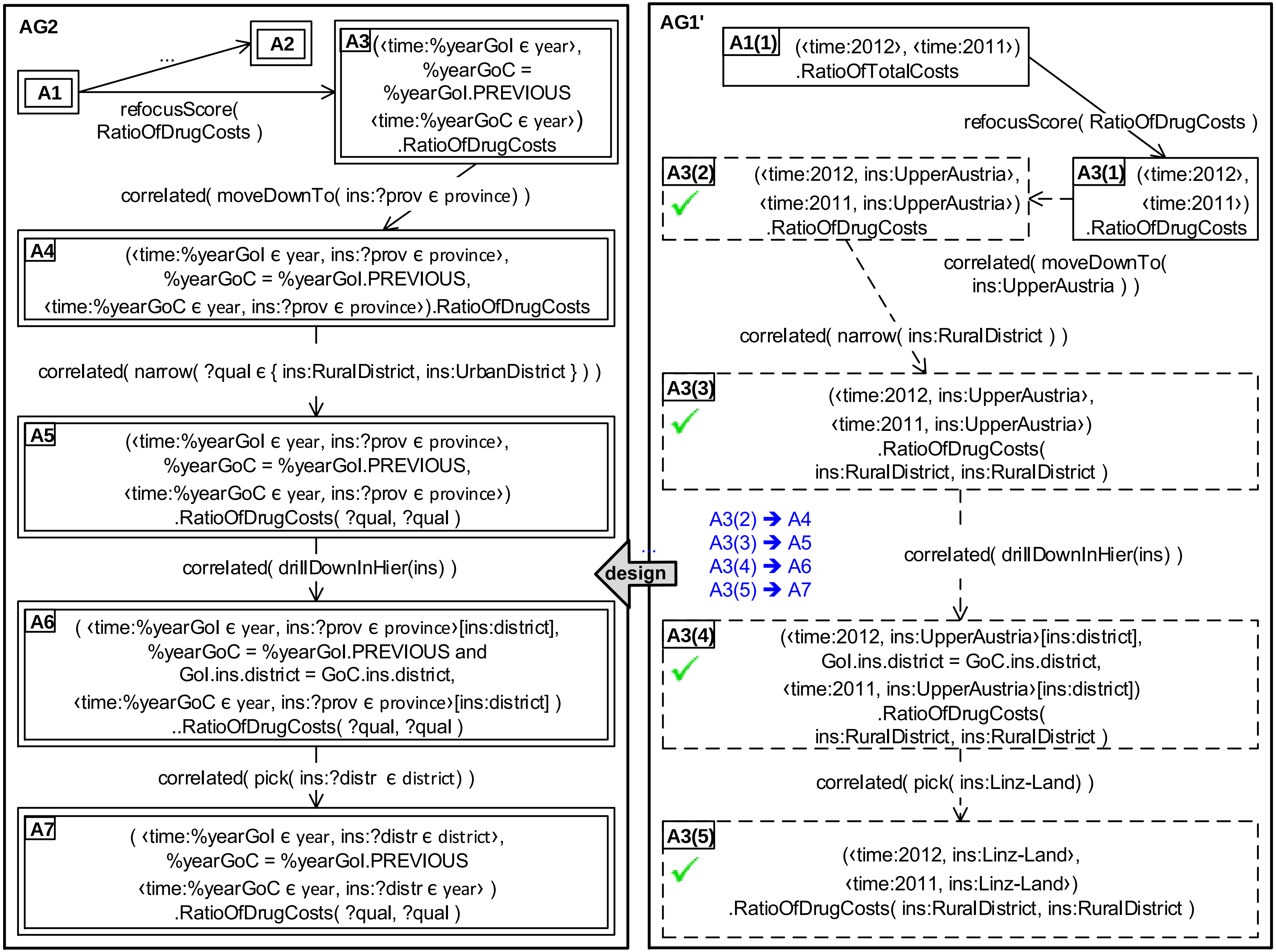
A3(3) (‹time:2012, ins:UpperAustria›, ‹time:2011, ins:UpperAustria›) .RatioOfDrugCosts(ins:ruralDistrict, ins:ruralDistrict)

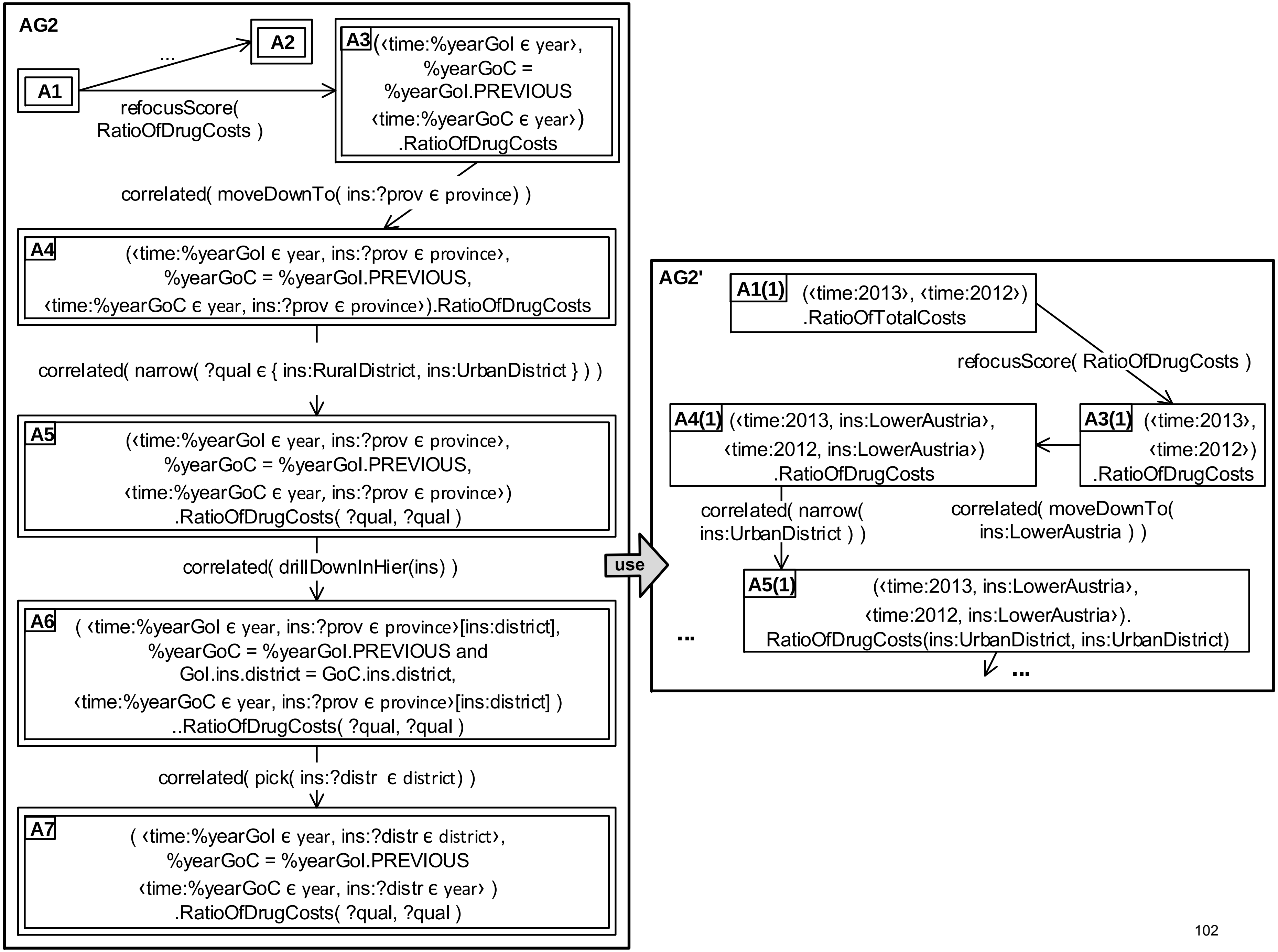
correlated(drillDownInHier(ins))

A3(4) (‹time:2012, ins:UpperAustria›[ins:district], GoI.ins.district = GoC.ins.district, ‹time:2011, ins:UpperAustria›[ins:district]) .RatioOfDrugCosts(ins:ruralDistrict, ins:ruralDistrict)

correlated(pick(ins:Linz-Land))

A3(5) (‹time:2012, ins:Linz-Land›, ‹time:2011, ins:Linz-Land›) .RatioOfDrugCosts(ins:ruralDistrict, ins:ruralDistrict)


AG2
A1
A2
...
refocusScore(RatioOfDrugCosts)
A3 (‹time:%yearGoI ϵ year›, %yearGoC = %yearGoI.PREVIOUS ‹time:%yearGoC ϵ year›) .RatioOfDrugCosts
correlated(moveDownTo(ins:?prov ϵ province))
A4 (‹time:%yearGoI ϵ year, ins:?prov ϵ province›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC ϵ year, ins:?prov ϵ province›).RatioOfDrugCosts
correlated(narrow(?qual ϵ { ins:RuralDistrict, ins:UrbanDistrict }))
A5 (‹time:%yearGoI ϵ year, ins:?prov ϵ province›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC ϵ year, ins:?prov ϵ province›) .RatioOfDrugCosts(?qual, ?qual)
correlated(drillDownInHier(ins))
A6 (‹time:%yearGoI ϵ year, ins:?prov ϵ province›[ins:district], %yearGoC = %yearGoI.PREVIOUS and GoI.ins.district = GoC.ins.district, ‹time:%yearGoC ϵ year, ins:?prov ϵ province›[ins:district]) ..RatioOfDrugCosts(?qual, ?qual)
correlated(pick(ins:?distr ϵ district))
A7 (‹time:%yearGoI ϵ year, ins:?distr ϵ district›, %yearGoC = %yearGoI.PREVIOUS ‹time:%yearGoC ϵ year, ins:?distr ϵ year›) .RatioOfDrugCosts(?qual, ?qual)
design
AG1'
A1(1) (‹time:2012›, ‹time:2011›) .RatioOfTotalCosts
refocusScore(RatioOfDrugCosts)
A3(1) (‹time:2012›, ‹time:2011›) .RatioOfDrugCosts
A3(2) ✓ (‹time:2012, ins:UpperAustria›, ‹time:2011, ins:UpperAustria›) .RatioOfDrugCosts
correlated(moveDownTo(ins:UpperAustria))
correlated(narrow(ins:RuralDistrict))
A3(3) ✓ (‹time:2012, ins:UpperAustria›, ‹time:2011, ins:UpperAustria›) .RatioOfDrugCosts(ins:RuralDistrict, ins:RuralDistrict)
A3(2) → A4
A3(3) → A5
A3(4) → A6
A3(5) → A7
correlated(drillDownInHier(ins))
A3(4) ✓ (‹time:2012, ins:UpperAustria›[ins:district], GoI.ins.district = GoC.ins.district, ‹time:2011, ins:UpperAustria›[ins:district]) .RatioOfDrugCosts(ins:RuralDistrict, ins:RuralDistrict)
correlated(pick(ins:Linz-Land))
A3(5) ✓ (‹time:2012, ins:Linz-Land›, ‹time:2011, ins:Linz-Land›) .RatioOfDrugCosts(ins:RuralDistrict, ins:RuralDistrict)



AG2

A1

A2

...

refocusScore(RatioOfDrugCosts)

A3 (‹time:%yearGoI ϵ year›, %yearGoC = %yearGoI.PREVIOUS ‹time:%yearGoC ϵ year›) .RatioOfDrugCosts

correlated(moveDownTo(ins:?prov ϵ province))

A4 (‹time:%yearGoI ϵ year, ins:?prov ϵ province›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC ϵ year, ins:?prov ϵ province›).RatioOfDrugCosts

correlated(narrow(?qual ϵ { ins:RuralDistrict, ins:UrbanDistrict }))

A5 (‹time:%yearGoI ϵ year, ins:?prov ϵ province›, %yearGoC = %yearGoI.PREVIOUS, ‹time:%yearGoC ϵ year, ins:?prov ϵ province›) .RatioOfDrugCosts(?qual, ?qual)

correlated(drillDownInHier(ins))

A6 (‹time:%yearGoI ϵ year, ins:?prov ϵ province›[ins:district], %yearGoC = %yearGoI.PREVIOUS and GoI.ins.district = GoC.ins.district, ‹time:%yearGoC ϵ year, ins:?prov ϵ province›[ins:district]) ..RatioOfDrugCosts(?qual, ?qual)

correlated(pick(ins:?distr ϵ district))

A7 (‹time:%yearGoI ϵ year, ins:?distr ϵ district›, %yearGoC = %yearGoI.PREVIOUS ‹time:%yearGoC ϵ year, ins:?distr ϵ year›) .RatioOfDrugCosts(?qual, ?qual)

use

AG2'

A1(1) (‹time:2013›, ‹time:2012›) .RatioOfTotalCosts

refocusScore(RatioOfDrugCosts)

A3(1) (‹time:2013›, ‹time:2012›) .RatioOfDrugCosts

correlated(moveDownTo(ins:LowerAustria))

A4(1) (‹time:2013, ins:LowerAustria›, ‹time:2012, ins:LowerAustria›) .RatioOfDrugCosts

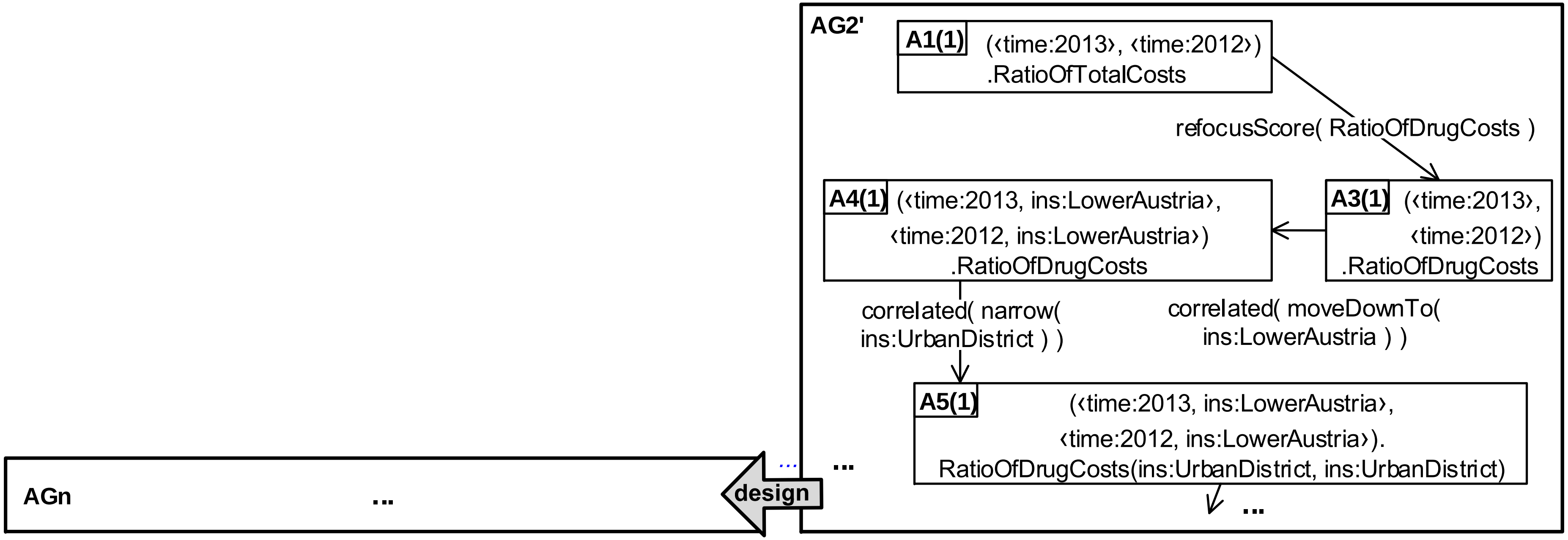
correlated(narrow(ins:UrbanDistrict))

A5(1) (‹time:2013, ins:LowerAustria›, ‹time:2012, ins:LowerAustria›). RatioOfDrugCosts(ins:UrbanDistrict, ins:UrbanDistrict)

...

...

Example: BI Analyis Graph (Design)


AG2'
A1(1) (‹time:2013›, ‹time:2012›) .RatioOfTotalCosts
refocusScore(RatioOfDrugCosts)
A4(1) (‹time:2013, ins:LowerAustria›, ‹time:2012, ins:LowerAustria›) .RatioOfDrugCosts
A3(1) (‹time:2013›, ‹time:2012›) .RatioOfDrugCosts
correlated(narrow(ins:UrbanDistrict))
correlated(moveDownTo(ins:LowerAustria))
A5(1) (‹time:2013, ins:LowerAustria›, ‹time:2012, ins:LowerAustria›). RatioOfDrugCosts(ins:UrbanDistrict, ins:UrbanDistrict)
...
...
...
AGn
...
design

6. Guidance, Judgment, and Analysis Rules

- Representation of former tacit knowledge of analyst
  - □ Process-oriented: Guidance in use of BI analysis graph (*guidance rules*)
  - □ Static: Possible explanations of striking score values (*judgment rules*)
  - □ Analysis: Analyse selected comparative facts (*analysis rules*) and report (reporting rules) interesting facts or initiate some action (action rules), e.g., start an analysis in some BI analysis graph
- Defined upon analysis situations or comparative cubes
- Inheritance and overriding along „specialization“ hierarchies of analysis situations and comparative cubes
- „Multi-granular“ evaluation along roll-up hierarchies of points

Guidance, Judgement and Analysis Rules

- Guidance rule
  - Process-oriented knowledge on how to best proceed in analysis
  - Tied to generic analysis situation
  - Action: recommendation where to look next
- Judgement rule
  - Static knowledge about comparative fact, process-independent
  - Applies to all facts of indicated cube, “semantic attachment”
  - Action: judgement why a score may be low/high, recommendation
- Analysis rule
  - Evaluated for part of DWH, e.g. for data loaded in last ETL-cycle
  - Action: “Management Summary”, or “Action invocation”

Definition & Evaluation of Guidance Rules

- A guidance rule is *defined* by
  - a name
  - a *domain*, given by a generic analysis situation
  - an *action*, given by a navigation operator (plus variable binding or domain restriction)
  - a *condition* over a comparative fact (*fact-oriented rule*) or over all facts of an analysis situation (*set-oriented rule*)
- A guidance rule is *evaluated*
  - when an individual analysis situation in the rule´s domain is met
  - for each fact / once for the set of facts
  - and the action is recommended, if the condition is met

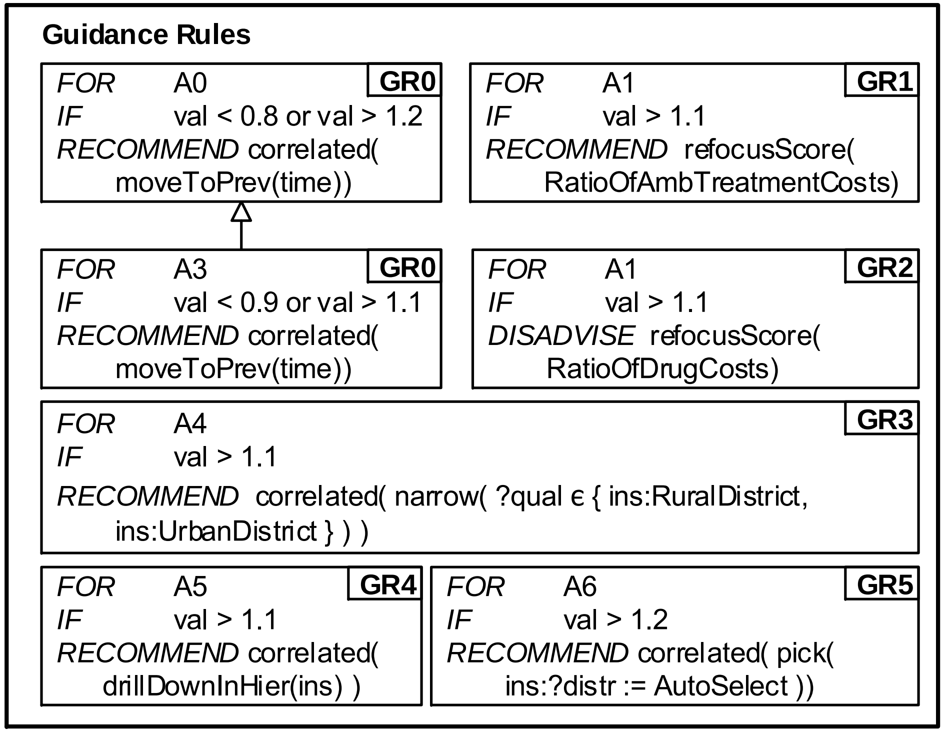
Example: Guidance Rule

GuidanceRule:	**AvgCostRatioPerRuralInsInAustria**
Point of Interest:	∈ (ins: Austria, actDoc: GP)*
Point of Comparison:	∈ (ins: Austria, actDoc: GP)*
GoI Qualifier:	^ (ins:RuralDistrict)
GoC Qualfier:	^ (ins:RuralDistrict)
Join condition:	GoI.time=GoI.time
Score:	avgCostRatioPerInsurant s
Rule Condition:	> 1.3
Rule Action:	Recommend correlated narrow “actDoc:RuralDistrict”

Rule applies to:

(<ins:UpperAustria,actDoc:GP>,<ins:Styria,actDoc:GP>).s(ins:RuraDistrict,ins:Rural District)=1.4

Example: Guidance Rules of Use Case



Guidance Rules

GR0
FOR A0
IF val < 0.8 or val > 1.2
RECOMMEND correlated(moveToPrev(time))

GR1
FOR A1
IF val > 1.1
RECOMMEND refocusScore(RatioOfAmbTreatmentCosts)

GR0
FOR A3
IF val < 0.9 or val > 1.1
RECOMMEND correlated(moveToPrev(time))

GR2
FOR A1
IF val > 1.1
DISADVISE refocusScore(RatioOfDrugCosts)

GR3
FOR A4
IF val > 1.1
RECOMMEND correlated(narrow(?qual є { ins:RuralDistrict, ins:UrbanDistrict }))

GR4
FOR A5
IF val > 1.1
RECOMMEND correlated(drillDownInHier(ins))

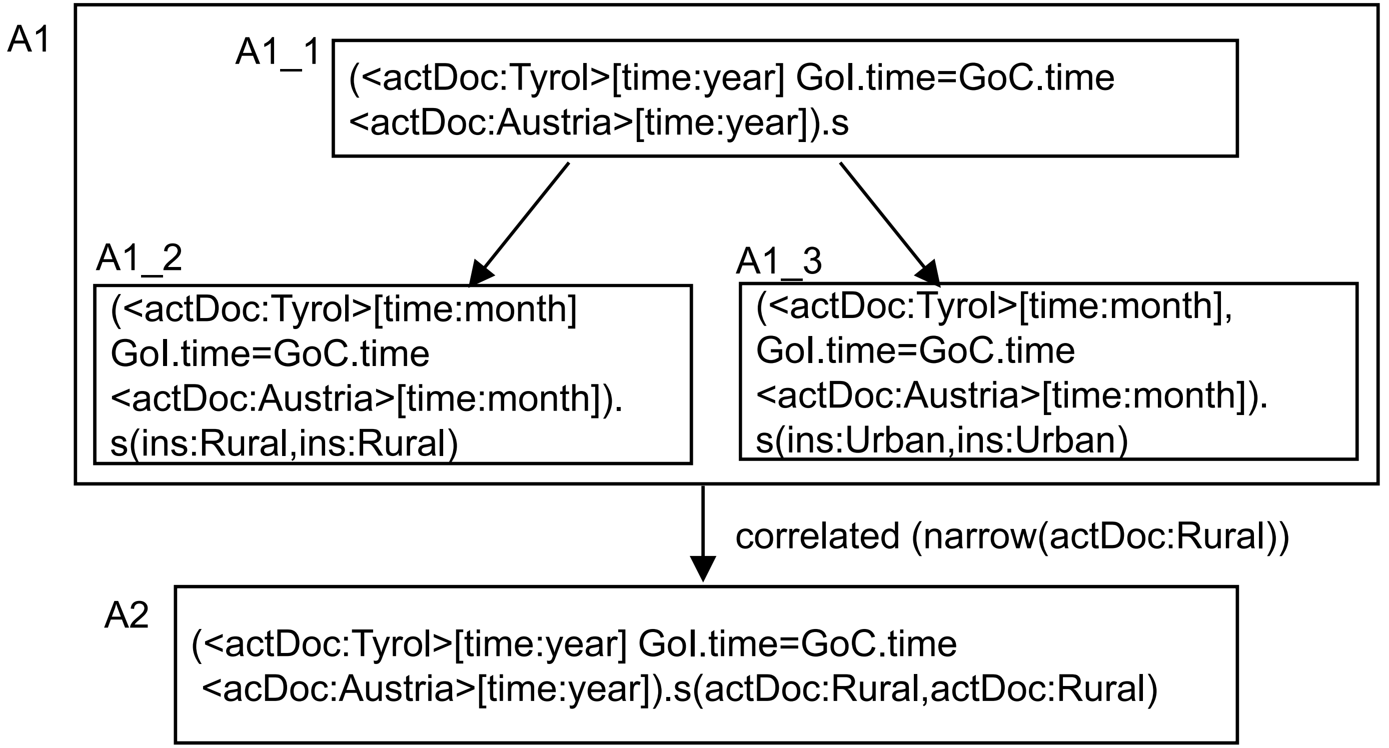
GR5
FOR A6
IF val > 1.2
RECOMMEND correlated(pick(ins:?distr := AutoSelect))

Guidance Rules: Conflict Resolution

- Conflicts: Multiple guidance rules with different guidance
  a) Apply for an analysis situation (set-oriented rule)
  b) Apply for a comparative fact (fact-oriented rule)
- Strategies for rules with different names:
  - □ Follow all, parallel in cloned tasks or iteratively in a work stack
  - □ Use rule priorities
  - □ Use priorities for guidance actions
- Strategies for rules with the same name:
  - □ Use inheritance and overriding, based on subsumption of rule domains (i.e., specialization of analysis situations)

Guidance Rules over composite ASs

- Considers multiple, related facts in rule condition
- Tied to the root AS, e.g., such a rule may guide from A1 to A2



A1

A1_1
(<actDoc:Tyrol>[time:year] GoI.time=GoC.time
<actDoc:Austria>[time:year]).s

A1_2
(<actDoc:Tyrol>[time:month]
GoI.time=GoC.time
<actDoc:Austria>[time:month]).
s(ins:Rural,ins:Rural)

A1_3
(<actDoc:Tyrol>[time:month],
GoI.time=GoC.time
<actDoc:Austria>[time:month]).
s(ins:Urban,ins:Urban)

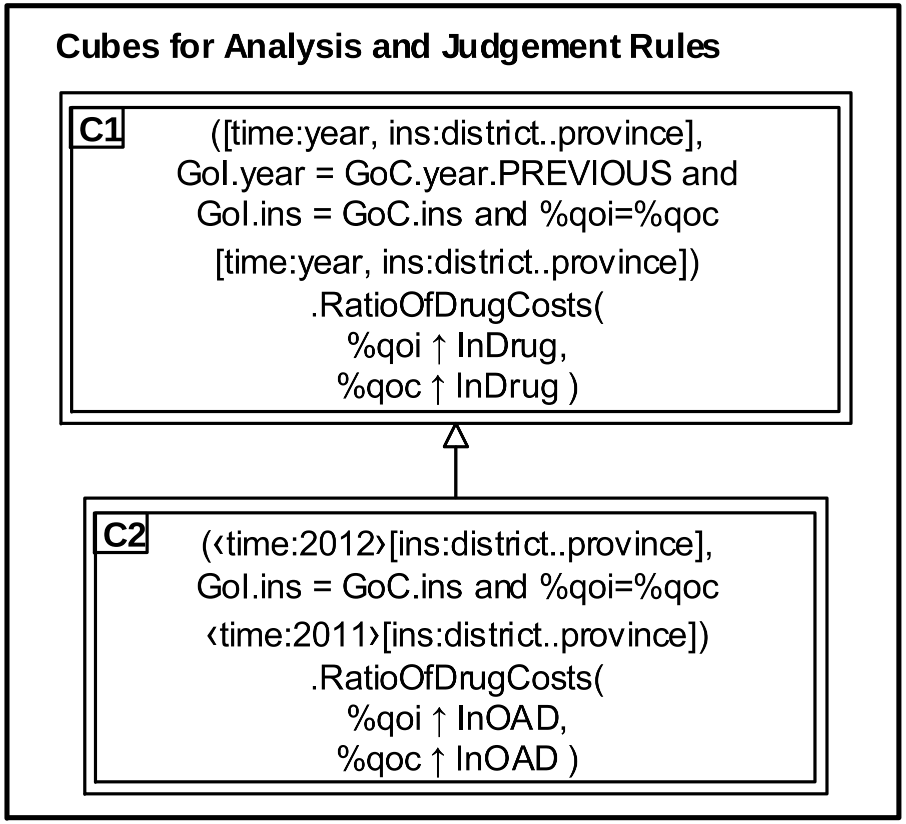
correlated (narrow(actDoc:Rural))

A2
(<actDoc:Tyrol>[time:year] GoI.time=GoC.time
<acDoc:Austria>[time:year]).s(actDoc:Rural,actDoc:Rural)

Judgement Rules

- Capture knowledge what a comparative fact *may* mean
- Defined for generic comparative cubes, i.e. comparative cubes with a generic score; condition over score, action is some text
- Specialization hierarchy for generic comparative cubes: C´ is *more specific* then C, if
  - □ the comparative cube space of C´ is subsumed by that of C
  - □ each qualifier of the generic score at C´ has the same or less restrictive domain than it has at C
- A judgment rule r is evaluated for a comparative fact f
  - □ when f in the domain of r is retrieved and no more specific judgment rule of the same name exists for f
  - □ and the fact is annotated by the judgment if the condition is met

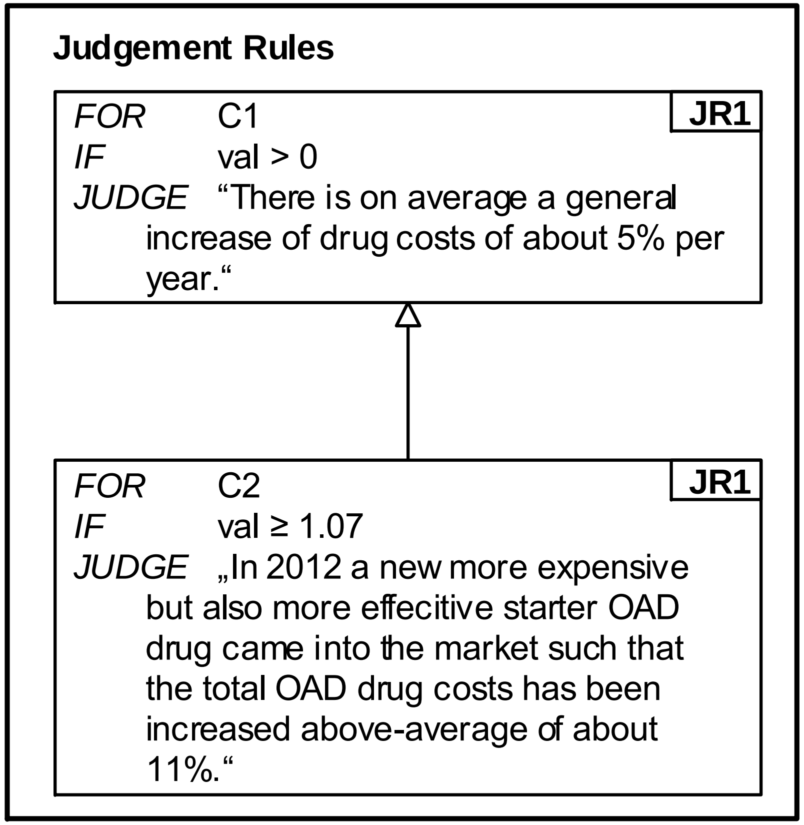
Example: Judgement Rules



Cubes for Analysis and Judgement Rules

C1 ([time:year, ins:district..province],
GoI.year = GoC.year.PREVIOUS and
GoI.ins = GoC.ins and %qoi=%qoc
[time:year, ins:district..province])
.RatioOfDrugCosts(
%qoi ↑ InDrug,
%qoc ↑ InDrug)

C2 (‹time:2012›[ins:district..province],
GoI.ins = GoC.ins and %qoi=%qoc
‹time:2011›[ins:district..province])
.RatioOfDrugCosts(
%qoi ↑ InOAD,
%qoc ↑ InOAD)



Judgement Rules

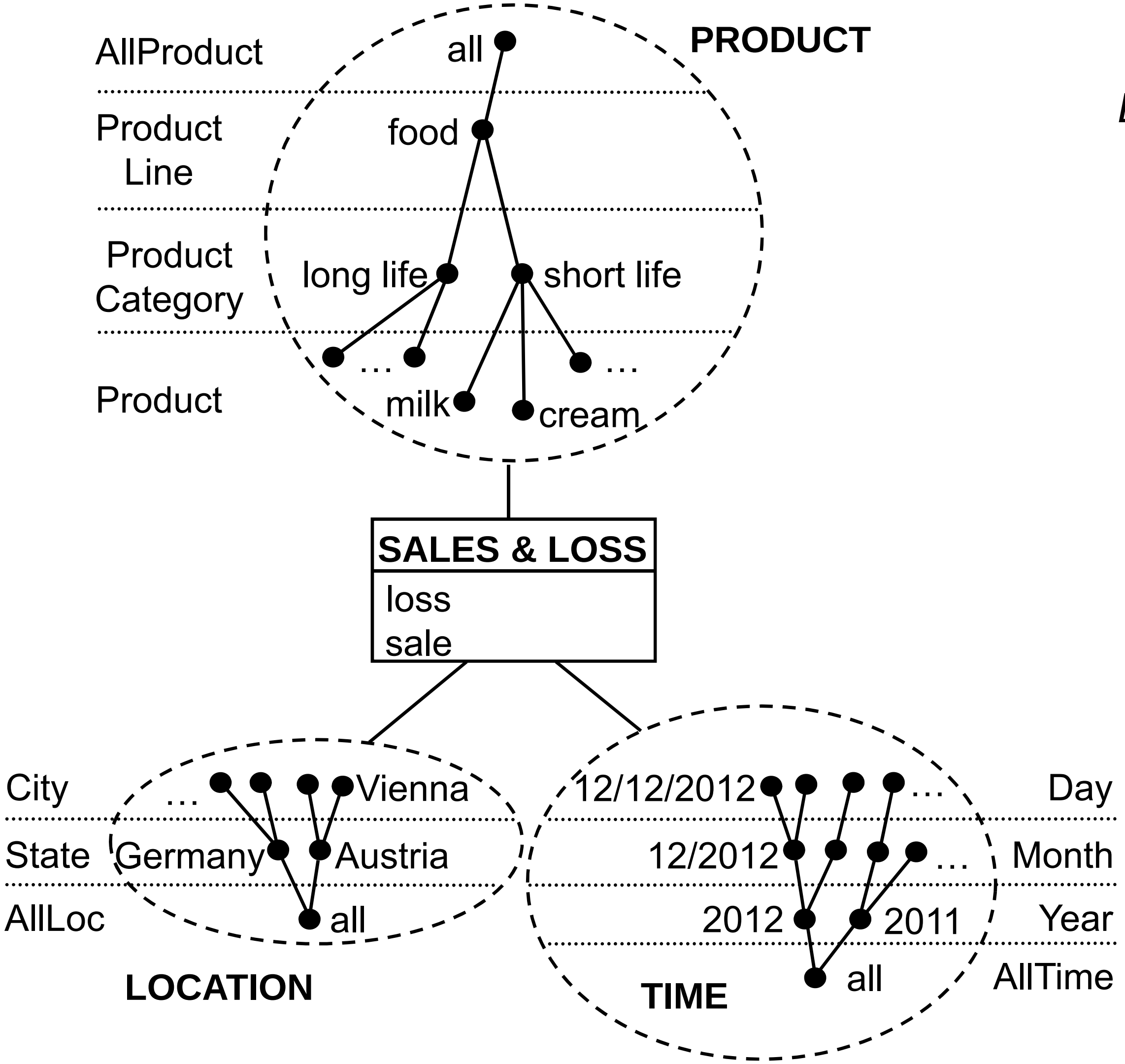
JR1
FOR C1
IF val > 0
JUDGE “There is on average a general increase of drug costs of about 5% per year.“

JR1
FOR C2
IF val ≥ 1.07
JUDGE „In 2012 a new more expensive but also more effective starter OAD drug came into the market such that the total OAD drug costs has been increased above-average of about 11%.“

Analysis Rules

- Defined like judgment rules, but
  - Evaluated only for indicated set of facts (e.g., those of last ETL-cycle)
  - Rule action is report (reporting rules) or “real” (actionable knowledge)
  - Positive and negative activation condition (“decision scope approach”)
- Multi-granular evaluation
  - Consider roll-up relationships between facts in rule evaluation
  - Rule evaluation strategies: prerogative vs. presumed
  - Explained for non-comparative case first

Ex.: Data Warehouse SALES & LOSS



Ex.: Base fact and roll-up fact

(shortLife,Austria,2012,800,2700)

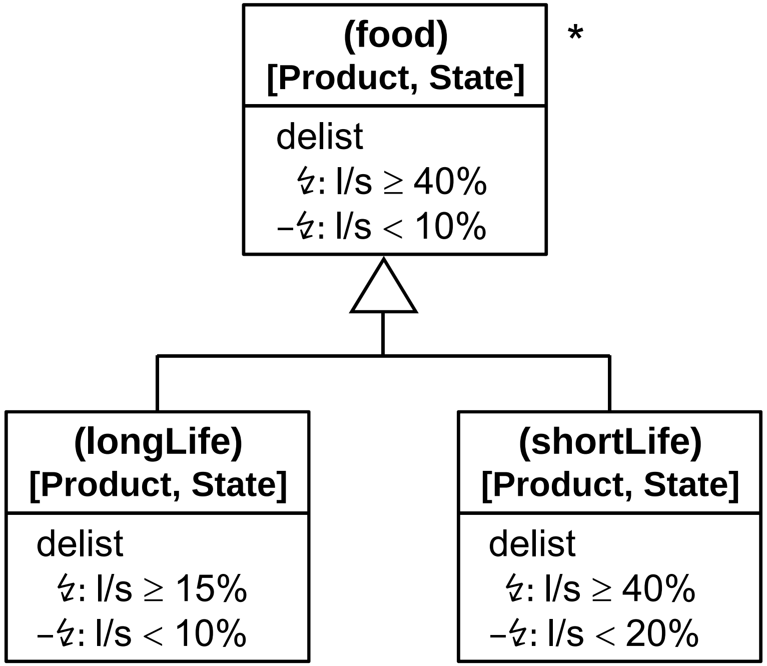
(Milk,Vienna,28-11-2012,5,20)

Ex.: Analysis rule

(food) **[Product, State]** [*]
delist ↯: l/s ≥ 40% –↯: l/s < 10%

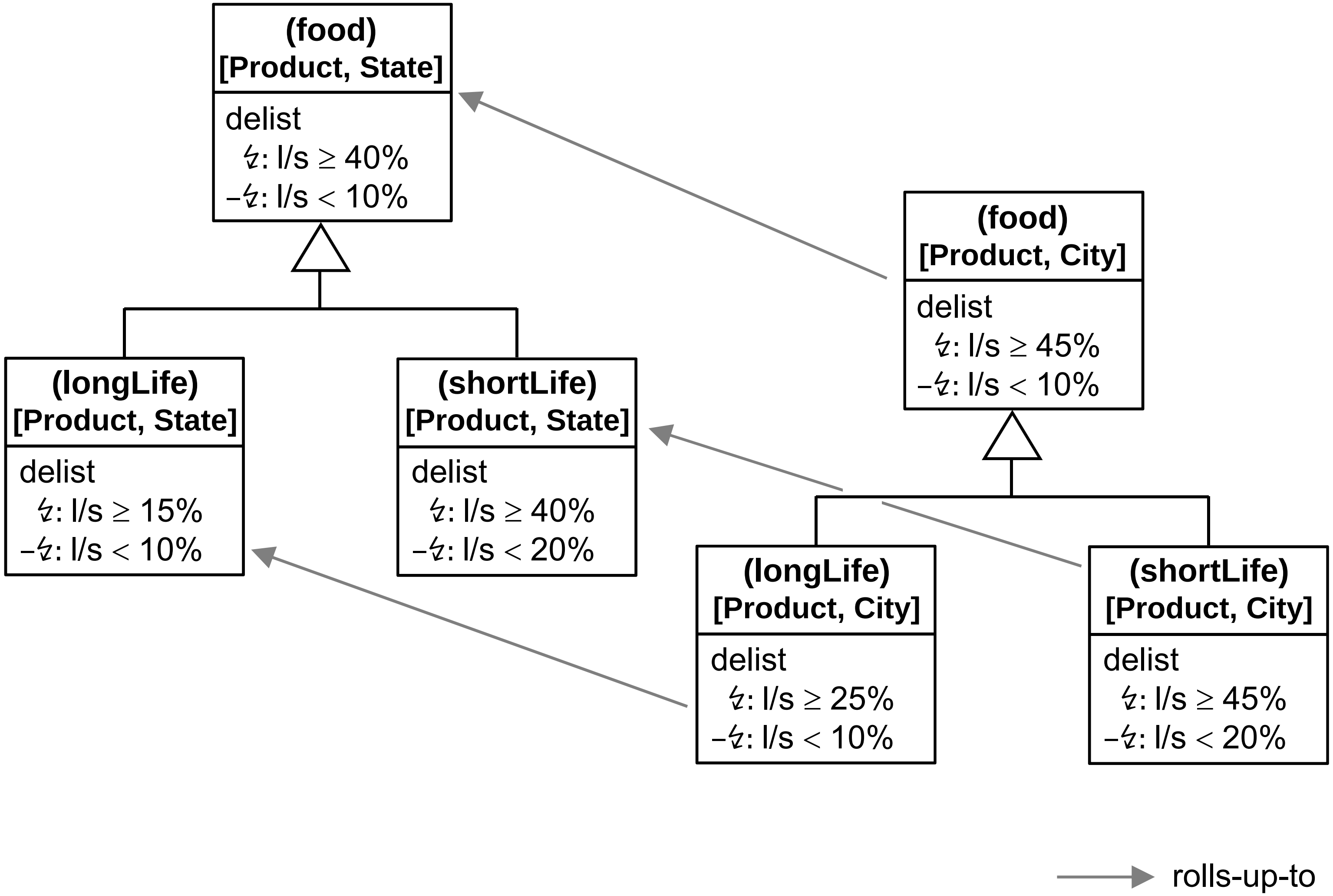
[*] set of descendent points of point (product:food, location:all) at granularity [Product, State]

Ex.: Mono-Granular Analysis Rule „Delist“



* set of descendent points of point
(product:food, location:all)
at granularity [Product, State]

Ex.: Multi-Granular Analysis Rule „Delist“



(food)
[Product, State]
delist
↯: I/s ≥ 40%
–↯: I/s < 10%

(longLife)
[Product, State]
delist
↯: I/s ≥ 15%
–↯: I/s < 10%

(shortLife)
[Product, State]
delist
↯: I/s ≥ 40%
–↯: I/s < 20%

(food)
[Product, City]
delist
↯: I/s ≥ 45%
–↯: I/s < 10%

(longLife)
[Product, City]
delist
↯: I/s ≥ 25%
–↯: I/s < 10%

(shortLife)
[Product, City]
delist
↯: I/s ≥ 45%
–↯: I/s < 20%

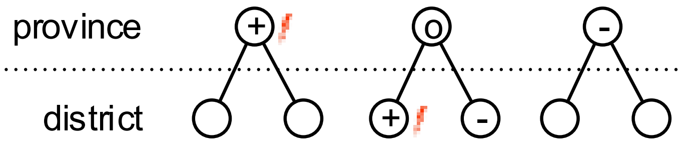
rolls-up-to

Evaluation Strategies for Multi-Granular Rules

- Two rules
  - Same name (regarded together as “one” analysis rule)
  - Domains at different granularities in “roll-up”-relationship
  - Rules evaluated “top-down”
- *Prerogative Evaluation* (for “action rules”)
  - If rule applies positively or negatively for a roll-up fact, the rule is not checked for its drill-down facts.
- *Presumed Evaluation* (for “reporting rules”)
  - If a rule applies positively or negatively for a roll-up fact, the rule is checked for its drill-down facts, but only reported if the decision would be different (e.g., “Beer sales are down, except Heineken”)

Evaluation Strategies: Comparison

prerogative evaluation of AR



⊕ pos. activation condition ist true

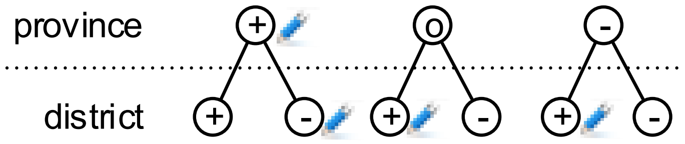
⊖ neg. activation condition ist true

ⓞ pos. and neg. activation condition is false

○ activation condition is not evaluated

⚡ action is performed

presumed evaluation of AR



⊕ pos. activation condition ist true

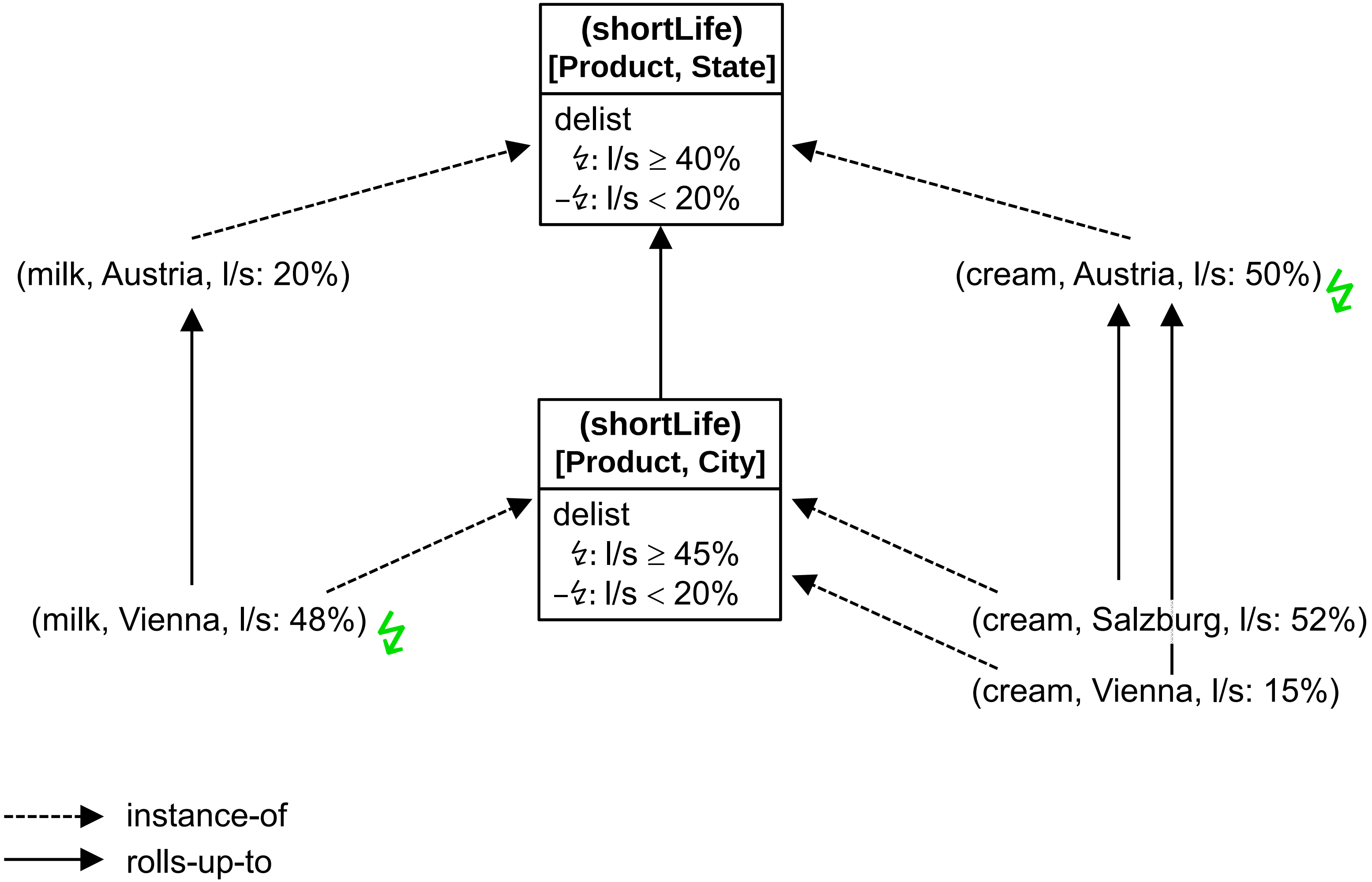
⊖ neg. activation condition ist true

ⓞ pos. and neg. activation condition is false

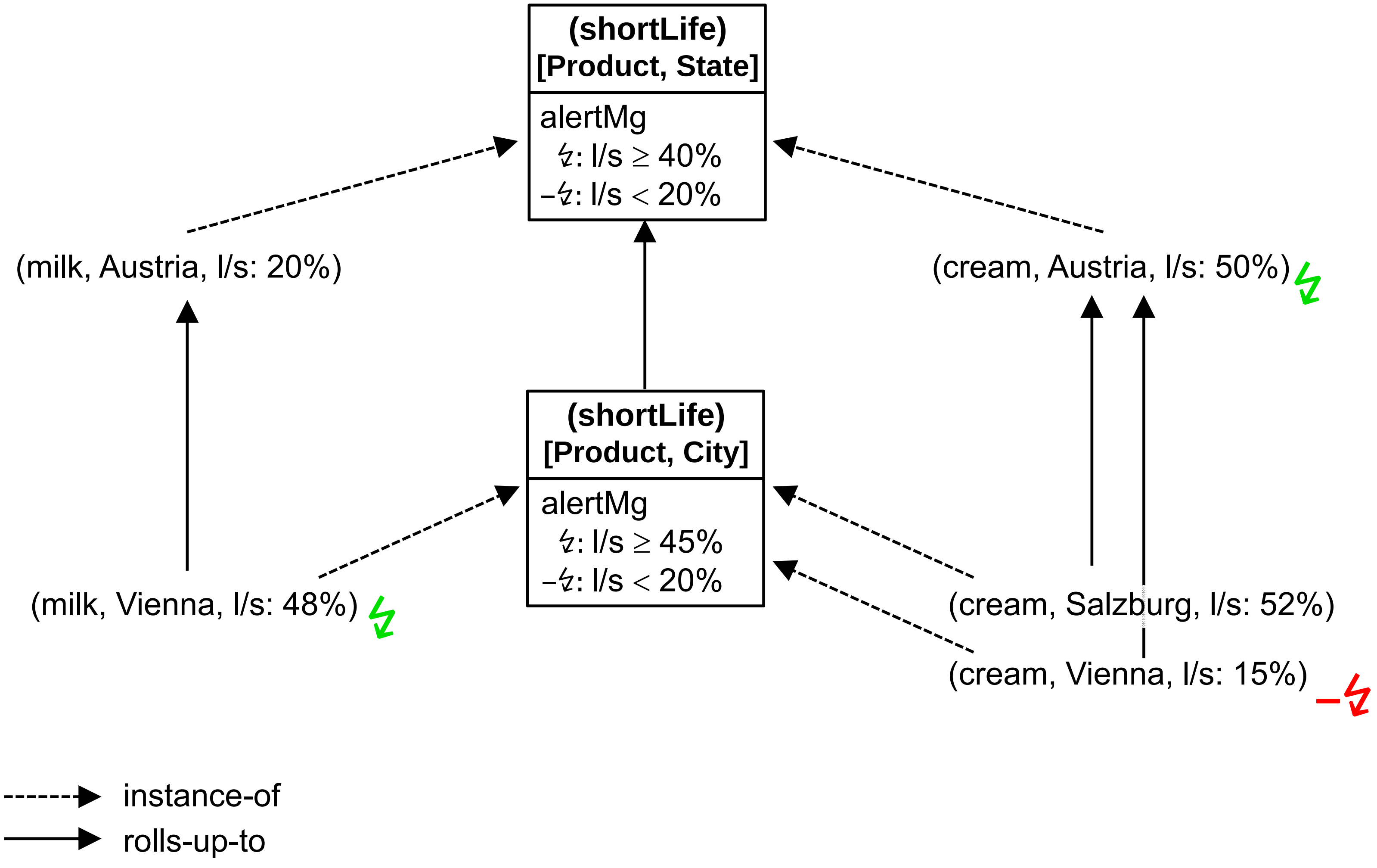
○ activation condition is not evaluated

✎ point is reported

Ex.: Multi-Granular, Prerogative Evaluation



Ex.: Multi-Granular, Presumed Evaluation



(shortLife)
[Product, State]

alertMg
↯: l/s ≥ 40%
–↯: l/s < 20%

(shortLife)
[Product, City]

alertMg
↯: l/s ≥ 45%
–↯: l/s < 20%

(milk, Austria, l/s: 20%)

(cream, Austria, l/s: 50%) ↯

(milk, Vienna, l/s: 48%) ↯

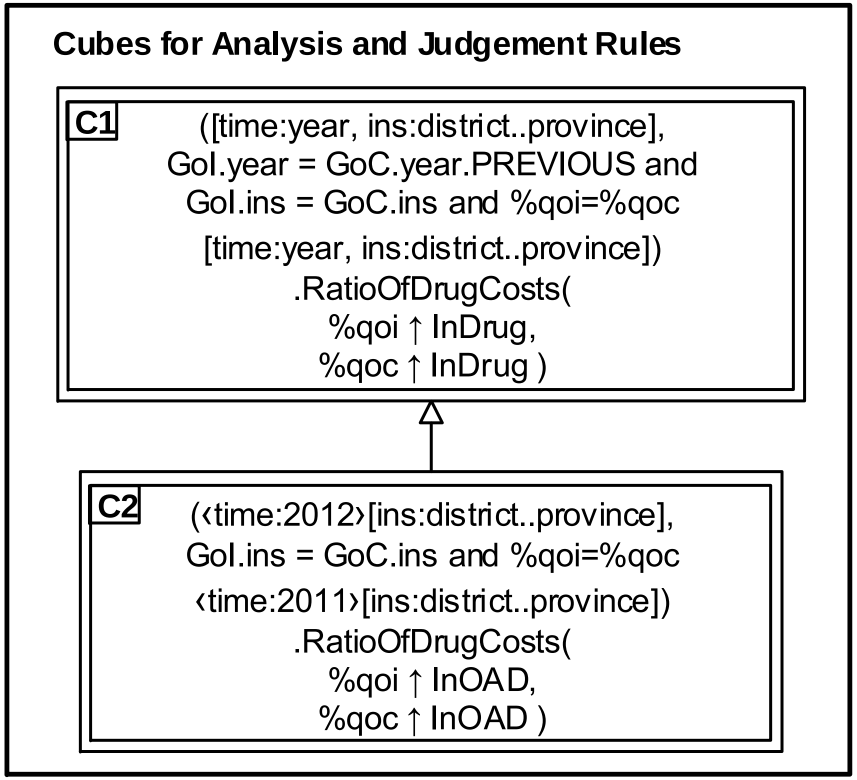
(cream, Salzburg, l/s: 52%)

(cream, Vienna, l/s: 15%) –↯

instance-of

rolls-up-to

Example: Analysis Rules (Use Case)



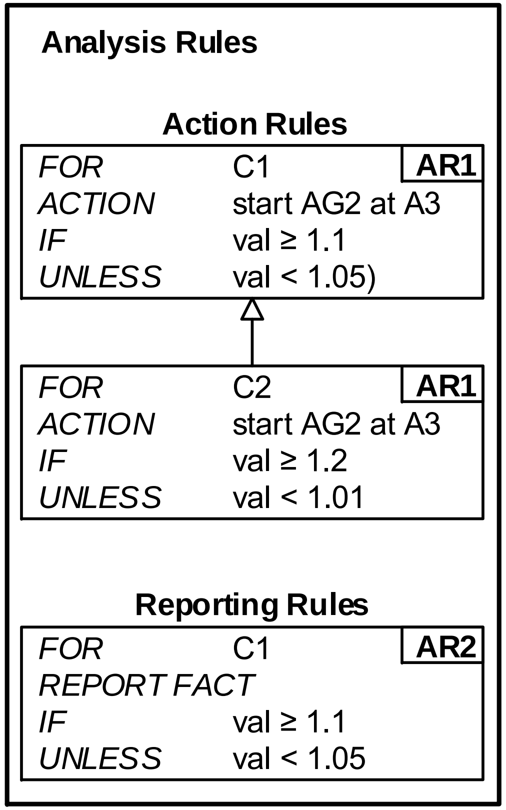
Cubes for Analysis and Judgement Rules

C1

([time:year, ins:district..province],
GoI.year = GoC.year.PREVIOUS and
GoI.ins = GoC.ins and %qoi=%qoc
[time:year, ins:district..province])
.RatioOfDrugCosts(
%qoi ↑ InDrug,
%qoc ↑ InDrug)

C2

(‹time:2012›[ins:district..province],
GoI.ins = GoC.ins and %qoi=%qoc
‹time:2011›[ins:district..province])
.RatioOfDrugCosts(
%qoi ↑ InOAD,
%qoc ↑ InOAD)



Analysis Rules

Action Rules

FOR	C1	**AR1**
ACTION	start AG2 at A3	
IF	val ≥ 1.1	
UNLESS	val < 1.05)	

FOR	C2	**AR1**
ACTION	start AG2 at A3	
IF	val ≥ 1.2	
UNLESS	val < 1.01	

Reporting Rules

FOR	C1	**AR2**
REPORT FACT		
IF	val ≥ 1.1	
UNLESS	val < 1.05	

Conclusion

- Comparative Data Analysis
  - *Is-to-is* comparison, exploratory, pre-cursor to data mining
  - Hundreds of light variations of simple measures / scores
  - Previous experience (knowledge) guides later analysis
- Approach
  - Define business terms through ontology; reuse existing ontology
  - Define generic measures & scores based on concepts of ontology
  - Map concepts and measures & scores to SQL
  - Explorative analysis: use concepts and instantiated scores to compare particular group of interest vs. group of comparison
  - Capture process knowledge by analysis graphs and guidance rules, and previous insights on comparisons by judgment and analysis rules