Judgement and Analysis Rules for Ontology-driven Comparative Data Analysis in Data Warehouses

Dieter Steiner **Bernd Neumayr** **Michael Schrefl**

Department of Business Informatics - Data & Knowledge Engineering
Johannes Kepler University Linz, Austria
Email: {steiner, neumayr, schrefl}@dke.uni-linz.ac.at

Abstract

Online analytical processing tools facilitate interactive inspection of aggregated measures of groups of data in data warehouses. In comparative data analysis, business analysts assess measures of a group of interest against measures of a group of comparison. Judgement rules annotate comparisons with background knowledge otherwise tacit to the business analyst. Analysis rules specify comparisons at different granularities and result in recommendations for further analyses. Judgement and analysis rules build on multidimensional ontologies, which simplify definition, reuse, and sharing of comparisons, and on comparative scores, which make explicit the results of comparisons. In this paper, we introduce conceptual modelling of judgement and analysis rules together with their organisation and multidimensional contextualisation in rule families, explain different rule evaluation strategies, and briefly report on the implementation of the approach.

Keywords: Ontology-driven Business Intelligence, OLAP, Rule Specialisation

1 Introduction

Business analysts use online analytical processing (OLAP) tools to interactively inspect and analyse data in data warehouses. Data warehouses organise data as multidimensional facts which typically represent business events. Facts are identified by dimensions and quantified by measures. OLAP operations allow for the interactive grouping of facts along dimension hierarchies, the aggregation of measures, and the selection of different groups of facts. To draw conclusions from aggregated measures it is often necessary to compare them with measures of some group of comparison. Using OLAP and visual data analytics tools, the interpretation of such comparisons are left to the human eye and depend on the intuition and experience of the business analyst. It is thus difficult to reuse and share the findings of comparative data analysis among business analysts.

This work was partially funded by the Austrian Ministry of Transport, Innovation, and Technology in program FIT-IT Semantic Systems and Services under grant FFG-829594 (Semantic Cockpit: an ontology-driven, interactive business intelligence tool for comparative data analysis).

The *Semantic Cockpit (semCockpit)* approach (Neuböck et al., 2013) to comparative data analysis has been developed in a joint research project of academia, industry, and prospective users from public health insurers. It extends OLAP with multidimensional ontologies (MDOs) and comparative scores and lifts comparative data analysis to the level of conceptual modelling with the goal to make more explicit and comprehensible the process and the results of comparative data analysis. Concept definitions in multidimensional ontologies complement dimensions and facts by capturing relevant business terms which are then used to specify groups of business events to be compared. Comparative ontologies treat comparisons as first-class citizens and describe them by comparative concepts which are organised in subsumption hierarchies. Comparative facts make explicit the result of particular comparisons between a group of interest and a group of comparison and quantify the comparison result by a score value.

Business analysts use *judgement rules* (Neumayr et al., 2011) to represent background knowledge that is relevant for the interpretation of comparative facts, especially to support novice business analysts and to provide possible explanations for exceptional score values. When a business analyst inspects comparative facts, the system annotates these facts automatically with knowledge encoded in judgement rules. The definition of rule scope and condition together with rule contextualisation along subsumption hierarchies of comparative concepts give the senior business analyst fine-grained control over these automatic annotations.

Business analysts use *analysis rules* (Neuböck et al., 2013) to automate routine comparative data analysis tasks, especially to compile meaningful analysis reports of noticeable comparative facts. The multi-granular evaluation of analysis rules extends the decision-scope approach (Schrefl et al., 2013) and its rule evaluation strategies to comparative data analysis.

In this work, we discuss in detail conceptual modelling of comparative rules, that is, judgement and analysis rules, and elaborate on their organisation and multidimensional contextualisation in rule families. In addition we explain the contextualised evaluation of judgement rules and apply the decision-scope approach to the contextualised and multi-granular evaluation of analysis rules. We make the paper self-contained by introducing a specification of the structure of comparative multidimensional ontologies.

The remainder of the paper is organised as follows. In Sect. 2, we introduce a metamodel for multidimensional comparative ontologies and for comparative querying of data warehouses through comparative cubes. In Sect. 3, we introduce the conceptual

modelling of comparative rules and their organisation in rule families. In Sect. 4, we explain the contextualised evaluation of judgement and analysis rules and give concrete examples of the results of rule evaluations depending on different rule evaluation strategies. In Sect. 5, we describe the implementation of the approach as part of the *semCockpit* prototype. We briefly review related work, in Sect. 6, and conclude the paper, in Sect. 7, with a summary and an outlook on future work.

2 Ontology-driven Comparative Data Analysis in Data Warehouses

In this section, as a prerequisite for judgement and analysis rules, we introduce an approach for specifying ontologies for comparative data analysis in data warehouses. We use the term *ontologies* synonymously to: shared conceptual domain models with richly defined concepts organised in subsumption hierarchies. We explain the approach along a simplified showcase of a public health insurance company that uses a data warehouse to keep track of and to analyse its history of business events. Business events of interest are, for example, drug prescriptions, ambulant treatments, and hospitalisations. Figure 1 shows a fragment of the multidimensional schema and data of this data warehouse.

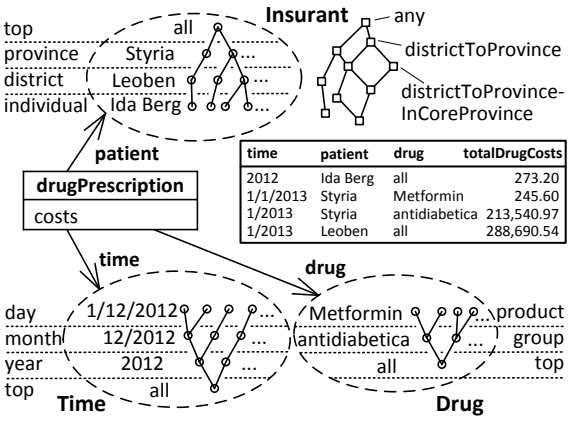

Figure 1: Multidimensional schema and data enriched with a subsumption hierarchy of dimensional concepts

The remainder of this section is structured as follows.

In Sect. 2.1 we give an overview of the structure of multidimensional ontologies (Neumayr et al., 2013) which treat dimension nodes (representing groups of entities) and multidimensional points (representing groups of business events) as first-class citizens, and enrich the data warehouse with dimensional concepts, e.g., districtToProvince-InCoreProvince of dimension Insurant in Fig. 1, and multidimensional concepts.

We extend, in Sect. 2.2, multidimensional ontologies (MDOs) to comparative ontologies which treat comparisons between groups of business events as first-class citizens and represent sets of comparisons with specific shared properties as comparative concepts, e.g., DrugPrescriptionComparison-VsPreviousYear in Fig. 2. The structure of multidimensional and comparative ontologies (the *MDO metamodel*) is partially specified in Fig. 3 as UML class diagram.

The formulation of queries in terms of cubes and comparative cubes is described in Sect. 2.3. A cube consists of facts, e.g., the facts in the table in Fig. 1, whereas a comparative cube consists of comparative

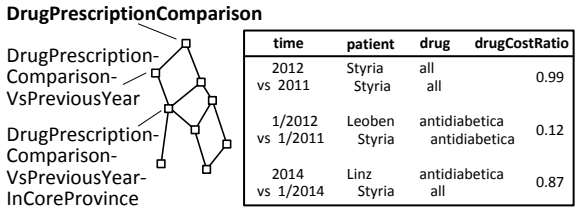

Figure 2: A subsumption hierarchy of comparative concepts and some sample comparative facts

facts, e.g., the three comparative facts in the table in Fig. 2. When formulating queries, the business analyst picks existing concepts from concept hierarchies to easily select the relevant groups of business events.

As illustrating example we specify a comparative cube comprising the ratios of drug prescription costs for specific groups of insurants (i.e., insurants that live in a core province of Austria grouped by district or province) of a given year compared to the corresponding costs of the previous year.

2.1 Overview of Multidimensional Ontologies

A *node* of a dimension hierarchy represents both an individual entity as well as a group of entities in the context of the dimension. For example, node Styria of dimension hierarchy insurant represents the province of Styria as well as the group of insurants that live in Styria.

Nodes are organised in dimension hierarchies. A *dimension hierarchy* consists of a roll-up hierarchy of *levels* forming a lattice with one *top* level and one *bottom* level. A *node* is at exactly one level and rolls up, for each parent level of its level, to one node at the parent level. The *all* node of a dimension is the only node at the top level of the dimension. Levels define *attributes* which are instantiated by its nodes.

A *level range* ranges *from* a finest level over intermediate levels *to* a coarsest level of a dimension hierarchy. It represents the nodes at these levels. For example, level range districtToProvince of dimension Insurant ranges from level district to level province. A single level may be used as a level range.

A *dimensional concept* represents a set of nodes of a dimension hierarchy with specific shared properties. Level ranges are regarded as a kind of dimensional concepts. Concepts are either primitive, or defined by a membership condition. For space limitations we do not cover the definition of membership conditions and we only give simplistic examples of concepts, we refer the interested reader to (Neumayr et al., 2013). The *domain* of a dimensional concept is given by a *level range* and restricts the applicability of the dimensional concept to the nodes of this level range. For example, dimensional concept districtToProvince-InCoreProvince has level range districtToProvince of dimension Insurant as domain and represents nodes (such as Styria and Leoben) in dimension insurant which, in turn, represent (a part of) a core province of Austria and are at level district or province.

A dimension hierarchy may play different dimension roles, e.g., dimension hierarchy insurant plays dimension role patient. Dimension roles are treated as first-class citizens, similar to roles in Description Logics, which facilitates an easy integration among different fact classes and the reuse of parts of multidimensional queries. In this paper we use the term *dimension* to refer both to dimension hierarchies as well as to dimension roles, the meaning should be clear from the context.

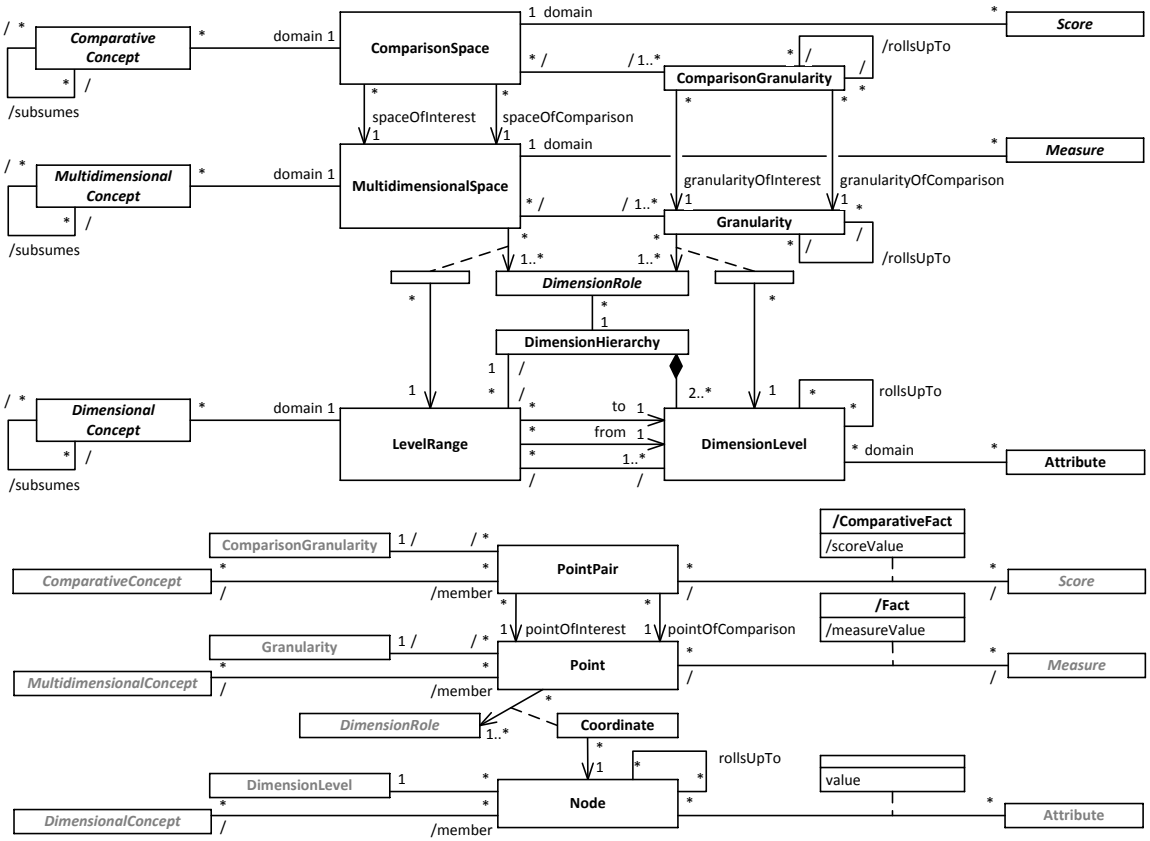

Figure 3: Overview of comparative multidimensional ontologies from a schema perspective (top) and an instance perspective (bottom), covering the most relevant parts of the MDO metamodel.

A set of dimensions together with a node for each dimension specify and identify a *point*, e.g, (time:1/1/2013, patient:Styria, drug:Metformin). This specification of a point is also referred to as the *coordinates* of a point. A *point* represents a group of business events, e.g., the group of Metformin prescriptions on day 1/1/2013 for patients in Styria.

A set of dimensions together with a dimension level for each dimension specify and identify a *granularity*. The dimension levels of the coordinates of a point give the granularity of the point. For example, point (time:1/1/2013, patient:Styria, drug:Metformin) has granularity (time:day, patient:province, drug:product).

A set of dimensions together with a level range for each dimension specify and identify a *multidimensional space* (or simply *space*). A point belongs to a space if it has the same dimensions and its coordinates are in the level ranges specified by the space. A space comprises one or more granularities. For example, space DrugPrescription-PerYearDistrictToProvince is specified as (time:year, patient:districtToProvince, drug:top) and comprises granularities (time:year, patient:district, drug:top) and (time:year, patient:province, drug:top).

A *measure*, e.g., totalDrugCosts, provides a means to quantify groups of business events. The domain of a measure is given by a space and specifies to which points it may be applied. A *fact*, e.g., (time:1/1/2013, patient:Styria, drug:Metformin; totalDrugCosts:245.60), represents the application of a measure to a point and quantifies the represented group of business events by a *measure value*.

A *multidimensional concept* represents a set of points with specific shared properties. The *domain* of a multidimensional concept is given by a space and restricts the applicability of the multidimensional concept to the points of this space. For example, multidimensional concept drugPrescription-PerYearDistrictToProvince-InCoreProvince has space DrugPrescription-PerYearDistrictToProvince as domain and represents points within this domain that have a patient coordinate from dimensional concept districtToProvince-InCoreProvince.

The MDO metamodel depicted in Fig. 3 is complemented as follows: Level ranges and multidimensional spaces are themselves regarded as concepts. The domain of a concept may be specified, or be derived from the membership condition, and is considered a part of the membership condition. Points are in a roll-up hierarchy, derived from the roll-up hierarchy of the nodes referred to by its coordinates.

2.2 Extending Multidimensional Ontologies to Comparative Ontologies

A *point pair* relates two points, a *point of interest (poi)* and a *point of comparison (poc)*, which represent two groups of business events, a *group of interest* and a *group of comparison*. In our example, point pair (poi:(time:2012, patient:Styria, drug:all), poc:(time:2011, patient:Styria, drug:all)) represents the comparison between drug prescriptions in the year 2012 for patients in Styria as group of interest and drug prescriptions in the year 2011 for patients in Styria as group of comparison.

A *score* represents a kind of comparison, e.g., score drugCostRatio compares the total costs of two groups of business events. A *comparative fact* represents the application of a score to a point pair and quantifies the comparison between group of interest and group of comparison by a *score value*. For example, comparative fact (poi:(time:2012,patient:Styria,drug:all), poc:(time:2011, patient:Styria, drug:all), drugCostRatio:0.99) represents a drug cost decrease of 1 % from 2011 to 2012 in the province of Styria.

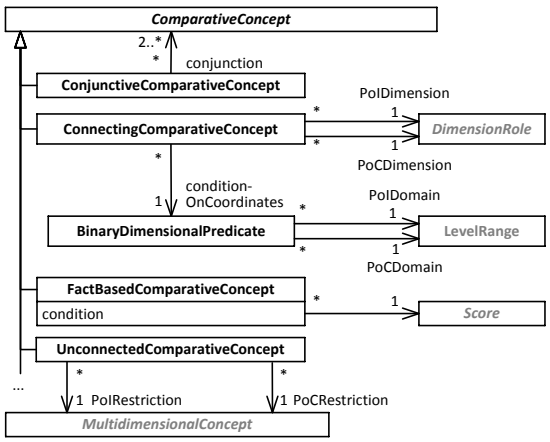

Figure 4: Definition of Comparative Concepts

A *comparative concept* represents a set of point pairs (each point pair representing a comparison) with specific shared properties. For example, comparative concept DrugPrescriptionComparison-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear represents the set of comparisons between groups of drug prescriptions of some year for insurants in core provinces grouped to district or province (as groups of interest) to similar groups but for the previous year (as groups of comparison).

The applicability of a score or a comparative concept to point pairs is restricted by its *domain*, which is given by a comparison space. A *comparison space* is given by two spaces, a *space of interest* as domain for the point of interest and a *space of comparison* as domain for the point of comparison.

A comparative concept is either primitive or defined. The membership condition of a defined comparative concept is specified in one of the following ways (see also Fig. 4).

- An *unconnected comparative concept* is defined by a reference to two multidimensional concepts, one restricting the points of interest and the other one restricting the points of comparison. For example, unconnected comparative concept DrugPrescriptionComparison-InCoreProvince is defined by multidimensional concept DrugPrescriptionInCore-Province acting both as restriction on the points of interest and the points of comparison.
- A *connecting comparative concept* selects point pairs by a condition over the relation between a coordinate for a specific dimension (poi-dimension) of the point of interest and a coordinate for a specific dimension (poc-dimension) of the point of comparison. The concept specifies the condition that must hold between the two coordinates by a *binary dimensional predicate* which takes the two coordinates as input and decides whether they meet the condition. Two level ranges, the poi-domain and the poc-domain, act as domain of the binary dimensional predicate. For example, concept VsSamePatients selects point pairs where point of interest and point of comparison have the same coordinate for dimension patient; concept VsPreviousYear selects point pairs where the time-coordinate of the point of interest is one year after the time-coordinate of the point of comparison. The definition of binary dimensional predicates is beyond the scope of this paper.
- A *conjunctive comparative concept* selects point pairs that belong to the intersection of a set of given concepts. For example, concept DrugPrescriptionComparison-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear is the conjunction of concepts DrugPrescriptionComparison-InCoreProvince and DrugPrescriptionComparison-PerYearDistrictToProvince-VsPreviousYear.
- A *fact-based comparative concept* is defined by a condition on the value of a specific score. It is interpreted by point pairs that are associated with comparative facts that fulfil this comparison.

2.3 Ontology-based Querying and Definition of Measures and Comparative Scores

An *ontology-based cube* (or simply *cube*) represents multiple measure applications to a set of points and results in a view on a subset of the asserted and derived facts in the data warehouse. The business analyst defines a cube by one or more measures, together with a multidimensional concept to specify the points to which the measures should be applied (see top part of Fig. 5). The domain of each measure needs to subsume the domain of the multidimensional concept. A cube evaluates to a collection of *composite facts*, one composite fact for each member point of its multidimensional concept. A composite fact consists of a fact for each measure associated with the cube. Such a fact may be empty, for example in case of missing base facts. Cubes may be given a name (for later re-use) or be defined in an ad-hoc manner as part of queries on the data warehouse.

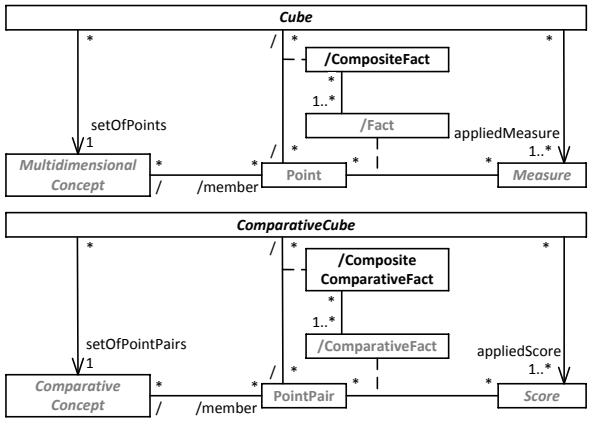

Figure 5: Specification and Evaluation of Ontology-based Cubes and Comparative Cubes

A measure is either a *base measure* (with facts being asserted) or a *derived measure* (with measure values being calculated from asserted or derived facts) (see top part of Fig. 6).

An *arithmetic measure* is defined by a simple arithmetic expression over one or more measures. When applied to a point (referred to as *this-point*), the arithmetic measure takes the facts associated with the this-point and applies the arithmetic expression on measure values of these facts.

An *aggregation measure* is defined by an aggregation function (such as *SUM* or *AVG*) together with a to-be-aggregated measure and, optionally, a multidimensional concept acting as qualifier. When applied to a point (the this-point), the aggregation measure selects all facts of the to-be-aggregated measure associated with points that roll up to the this-point, and, if a qualifier is given, that belong to the qualifier.

An *ontology-based comparative cube* (or simply *comparative cube*) represents multiple score applications to a set of point pairs. The business analyst

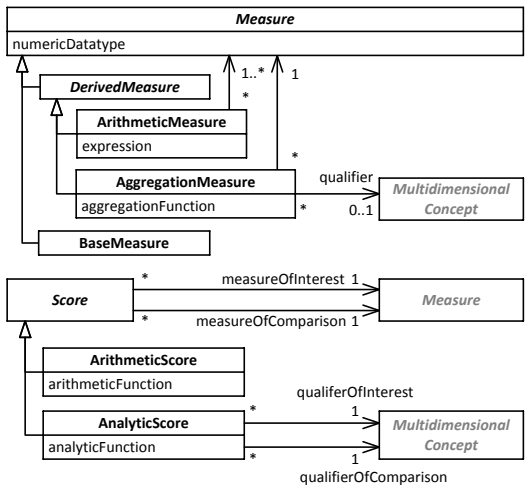

Figure 6: Specification of Measures and Scores

defines a comparative cube by a comparative concept and one or more scores (see bottom part of Fig. 5). The domain of each score must subsume the domain of the comparative concept. The scores are applied to every point pair that belongs to the comparative concept. This results in comparative facts, one for each point pair and score, which are grouped into one *composite comparative fact* for each point pair.

A score is either an *arithmetic score* or an *analytic score* (see bottom part of Fig. 6).

An *arithmetic score* is defined by a *measure of interest* and a *measure of comparison* together with an arithmetic function. When applied to a point pair (point of interest and point of comparison) it uses the arithmetic function (such as *ratio*) to quantify the relation between the fact associated with point and measure of interest and the fact associated with point and measure of comparison. For example, score drugCostRatio employs arithmetic function ratio to quantify the comparison between the totalDrugCosts (used both as measure of interest and of comparison) of point of interest and point of comparison.

An *analytic score* is defined by a measure of interest and a measure of comparison together with two multidimensional concepts, acting as *qualifier of interest* and *qualifier of comparison*. When applied to a point pair (this-point of interest, this-point of comparison) the analytic score selects the *facts of interest* which are given by the facts associated with the measure of interest and associated with points that belong to the qualifier of interest and roll up to the this-point of interest, and the *facts of comparison* which are given analogously. It then uses an analytic function (such as *median percentile rank*) to quantify the comparison between facts of interest and facts of comparison.

Aggregation, arithmetic, and analytic functions in measures and scores may be parameterised to specify their treatment of empty facts.

At this point the comparative cube mentioned at the beginning of Sect. 2 can be modelled. Comparative cube drugCostRatio-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear applies score drugCostRatio to the members of comparative concept DrugPrescriptionComparison-PerYear-DistrictToProvince-InCoreProvince-VsPreviousYear. The score captures the ratio of drug prescription costs, while the comparative concept specifies the comparative points, i.e., the yearly aggregates for core provinces and districts of core provinces compared to the previous year, for which the score is evaluated. Table 1 shows the evaluation result of this comparative cube.

point of interest			point of comparison			drugCostRatio
time	patient	drug	time	patient	drug	
2012	Styria	all	2011	Styria	all	0.99
2012	Leoben	all	2011	Leoben	all	1.08
2012	Murau	all	2011	Murau	all	1.21
2012	Weiz	all	2011	Weiz	all	0.89
2012	Lower Austria	all	2011	Lower Austria	all	1.10
2012	Melk	all	2011	Melk	all	0.97
2012	Korneuburg	all	2011	Korneuburg	all	1.07
2012	Horn	all	2011	Horn	all	1.03
2012	Upper Austria	all	2011	Upper Austria	all	1.02
2012	Linz-Stadt	all	2011	Linz-Stadt	all	1.05
2012	Wels-Stadt	all	2011	Wels-Stadt	all	0.98
2012	Eferding	all	2011	Eferding	all	0.93

Table 1: Example evaluation of a comparative cube

3 Modelling of Comparative Rules

In this section, we introduce the modelling of two kinds of comparative rules, judgement rules and analysis rules, and their organisation in rule families. We first explain the modelling of comparative rules in general and then look at the specificities of *judgement rules* and of *analysis rules* and their corresponding rule families.

Comparative rules build on comparative concepts, scores and comparative cubes. The structure of comparative rules is specified in Fig. 7.

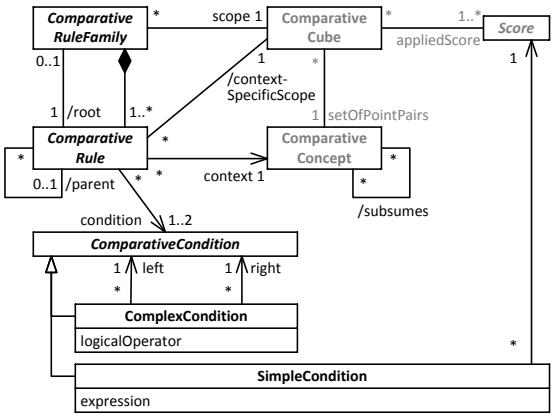

Figure 7: Comparative Rule

A *comparative rule family* is a collection of related, contextualised rules that are evaluated as single entity. Each rule family specifies its scope in terms of a comparative cube. This cube defines the scores that can appear in conditions of rules of the rule family and a comparative concept defining the point pairs for which contextualised rules can be defined.

A *comparative rule* belongs to a rule family and defines conditions over and behaviour for the *composite comparative facts* (or *facts* for short) in its context-specific scope. The *context-specific scope* of a rule is a comparative cube that is specified by a comparative concept (the *context* of the rule) and the scores specified with the rule family. A rule defines one (judgement rule) or two (analysis rule) conditions over score values to be tested on facts in its context-specific scope. A simple condition is expressed by a score-value comparison. Multiple conditions can be combined by conjunction and disjunction.

When a rule family is evaluated over the facts of a comparative cube (see Sect. 4 for details), for each fact only the most-specific rule (whose context contains the point pair of the fact) is evaluated. For this purpose, the system derives a *rule hierarchy* for each

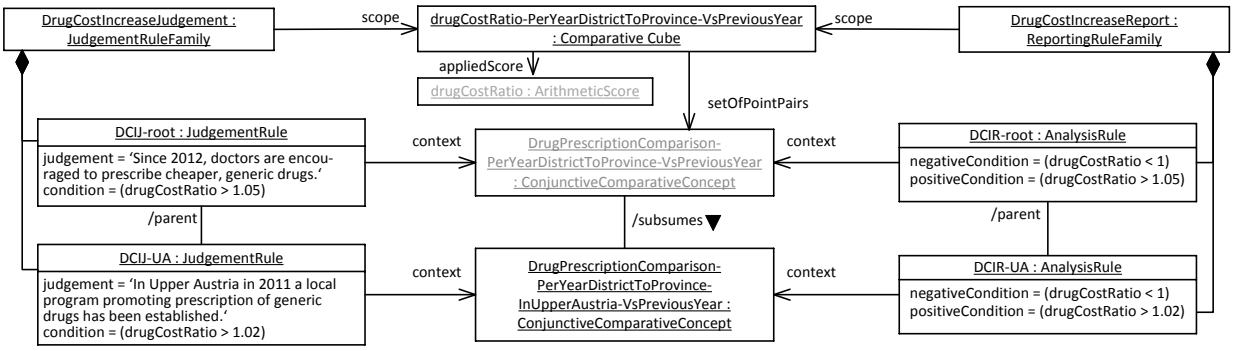

Figure 8: A judgement rule family (left) and an analysis rule family (right)

rule family based on the subsumption hierarchy of the rules' contexts. This hierarchy has to define a single *root* rule whose context subsumes the context of all other rules of the rule family. In order to ensure that there is one most-specific rule for each fact, contexts of sibling rules (rules with the same parent rule) need to be disjoint.

For each kind of rule we exemplify (see Fig. 8) the definition of rule families and rules on top of comparative cube drugCostRatio-PerYearDistrictToProvince-VsPreviousYear, which is restricted by comparative concept DrugPrescriptionComparison-PerYearDistrictToProvince-VsPreviousYear, and subsumes the previously introduced comparative cube drugCostRatio-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear. We model a judgement rule family that informs analysts about the status of programs promoting the prescription of generic drugs for areas with relatively high cost increases. We model an analysis rule family that results in a report consisting of districts and provinces with exceptional cost increases in order to identify areas that would benefit from future cost reduction programs. These example rules are further explained in the remainder of this section, and their evaluation is explained in Sect. 4.

3.1 Judgement Rules

Business analysts use judgement rules to represent specific, otherwise tacit, knowledge about groups of business events that is relevant in specific situations of comparative data analysis. Judgement rules annotate comparative facts with this knowledge, especially to support novice business analysts in the interpretation of data. For example, the judgement rule family depicted in Fig. 8 informs about an aspired shift from prescription of brand drugs to cheaper, generic drugs. An information which is especially interesting when discovering high increases in drug costs, which is contrary to the shift to generic drugs. Figure 9 depicts the model of judgement rules as a specialisation of comparative rules.

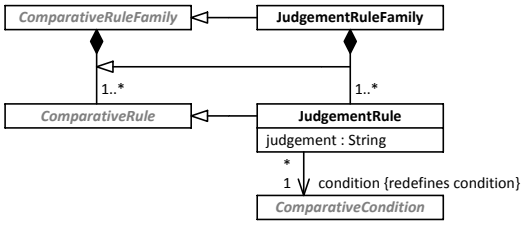

Figure 9: Judgement Rule

A *judgement rule family* is a special comparative rule family as it is a collection of judgement rules.

A *judgement rule* is the contextualisation of a judgement rule family to a particular context given by a comparative concept. A rule is further defined by a single condition over score values of a fact and a judgement that is annotated to the fact if the condition is met. A judgement is a textual annotation providing some additional information for the business analyst. Judgement rules of the same rule family override each other according to the derived rule hierarchy so that for each fact only the most specific rule is evaluated, with the most-specific condition and the most-specific judgement.

The left part of Fig. 8 shows judgement rule family DrugCostIncreaseJudgement with its two member rules DCIJ-root and DCIJ-UA. Rule family DrugCostIncreaseJudgement has comparative cube drugCostRatio-PerYearDistrictToProvince-VsPreviousYear as scope. The judgement of rule DCIJ-root expresses some general knowledge about a desired shift from prescription of brand drugs to cheaper, generic drugs. This information is displayed for facts with a drug cost increase of more than 5 % which is contrary to the desired shift to cheaper drugs. Rule DCIJ-UA expresses more detailed information for Upper Austria, where a more effective program for the promotion of cheaper drugs has been established. This information should be shown for facts with a drug cost increase of more than 2 %.

The latter rule is contextualised to drug prescription comparisons for the province of Upper Austria and its districts through comparative concept DrugPrescriptionComparison-PerYearDistrictToProvince-InUpperAustra-VsPreviousYear. Rule DCIJ-root constitutes the root rule of rule family DrugCostIncreaseJudgement with rule DCIJ-UA lying below rule DCIJ-root in the rule hierarchy. Rule DCIJ-root is applicable to facts in its scope that are not in the context-specific scope of rule DCIJ-UA.

3.2 Analysis Rules

Analysis rules provide a means for hierarchical analysis of comparative facts, from coarse-grained to fine-grained, and result in recommended actions or in a report of noticeable facts. The business analyst uses analysis rules to model and automate routine and semi-routine comparative data analysis and decision tasks. Figure 10 depicts the model of analysis rules as a specialisation of comparative rules.

An *analysis rule family* is either a *reporting rule family* or an *action rule family*. Reporting rule families report the result of their evaluation to the business analyst. Action rule families additionally recommend an action for facts with positive activation. The action defined by an action rule family could be executed automatically. Automatic action execution is not within the scope of this work and requires additional specifications. Actions are specified with the

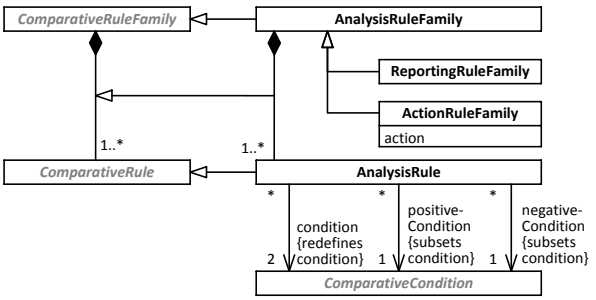

Figure 10: Analysis Rule

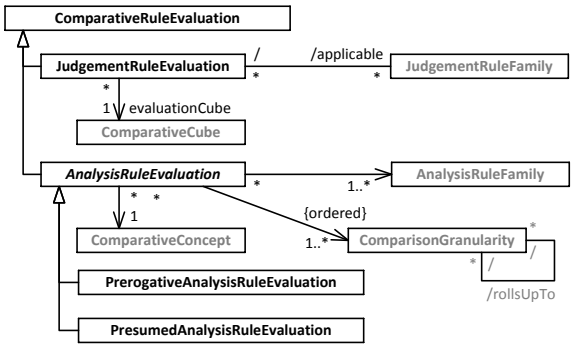

Figure 11: Comparative Rule Evaluation

rule family and are not contextualised with the rules (this is contrary to judgements which are contextualised with the rules). The dynamic (and context-specific) binding of actions depending on the point pair is subject to future work.

An *analysis rule* is the contextualisation of a rule family (either a reporting or an action rule family) and defines a positive activation condition and a negative activation condition that are specific to the context of the rule. The gap between positive and negative activation condition (if any) leaves open a decision scope for subsequent rule evaluation at finer granularities (this will be explained in the next section). The hierarchical contextualisation of rule families and the hierarchical decision-scope based rule evaluation are orthogonal to each other.

The right part of Fig. 8 shows reporting rule family DrugCostIncreaseReport which consists of two analysis rules, DCIR-root and DCIR-UA. Rule DCIR-root is the root rule and has comparative concept DrugPrescriptionComparison-PerYear-DistrictToProvince-VsPreviousYear as its context. It specifies a positive activation for facts with a cost increase of more than 5 % and a negative activation for facts with a cost decrease. It thereby leaves open a decision scope for facts at finer granularities that roll up to facts with a slight cost increase of up to 5 %. Rule DCIR-UA is contextualised to comparative concept drug-CostRatio-PerYearDistrictToProvince-InUpperAustria-VsPreviousYear. It specifies a positive activation for facts with a cost increase of more than 2 % and a negative activation for facts with a cost decrease. It thereby leaves open a decision scope for facts at finer granularities that roll up to facts with a slight cost increase of up to 2 %.

4 Evaluation of Comparative Rules

In this section, we first explain the contextualised evaluation of comparative rules in general and then look in detail at specific rule evaluation strategies for judgement rules and for analysis rules. We exemplify each rule evaluation strategy by an evaluation of a rule family modelled in Fig. 8 over comparative cube drugCostRatio-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear shown in Table 1.

Rules are not evaluated in isolation but as part of rule families. Given a comparative fact, the most specific rule for this comparative fact is selected from the rule hierarchy and applied to the comparative fact.

The triggering and evaluation of judgement and analysis rule families varies significantly (see Fig. 11). When the business analyst poses a comparative query by a comparative cube, the system automatically derives potentially applicable judgement rule families and applies them to each fact of the comparative cube. For analysis rules, the business analyst triggers explicitly the evaluation of one or more rule families and restricts the facts on which the rules should be applied to by a comparative concept and a set of comparison granularities which are in a roll-up path. Comparative rule evaluations could also be deployed as a kind of standing query (not further discussed in this paper) to be automatically evaluated over new comparative facts derived from facts that just entered the data warehouse via ETL.

4.1 Contextualised Evaluation of Judgement Rules

A *judgement rule evaluation* is associated with a given comparative cube, referred to as *evaluation cube*, which represents a query of the business analyst. The system automatically detects the set of judgement rule families that are potentially applicable to the facts in the evaluation cube. A judgement rule family is applicable to an evaluation cube if the rule family's scope, given by a comparative cube, defines the same or a subset of the scores of the evaluation cube and the sets of point pairs of the two cubes overlap. The execution of a judgement rule evaluation on an evaluation cube comprises the evaluation of all applicable judgement rule families.

Judgement rule family evaluation leads to an augmented query of the evaluation cube with judgements based on the rule definitions. Each fact in the evaluation cube is also part of the augmented evaluation, with judgements offering additional information. For those facts for which a judgement is available, i.e., the condition of the most specific judgement rule of an applicable rule family is satisfied, the judgement together with the responsible rule is reported alongside the conventional fact. For each fact of a judgement rule evaluation, multiple judgements from different rule families might be triggered and annotated in the analysis report.

For example, during the judgement rule evaluation for comparative cube drugCostRatio-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear the system, first, derives all applicable judgement rule families (in our example, there is only one rule family DrugCostIncreaseJudgement) and, second, applies to each fact the most-specific rule of each rule family. Table 2 shows the simplified result of such a judgement rule evaluation. The full comparative facts, including coordinates of the points of comparison, have been shown in Table 1.

The applicability of judgement rule family DrugCostIncreaseJudgement to this evaluation cube is derived as follows. Judgement rule family DrugCostIncrease-Judgement of Fig. 8 defines its scope through comparative cube drugCostRatio-PerYearDistrictToProvince-VsPrevious-Year which is specified by score drugCostRatio and comparative concept DrugPrescriptionComparison-PerYearDistrictTo-Province-VsPreviousYear. The evaluation cube drugCost-Ratio-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear is also

specified by score drugCostRatio and by a comparative concept DrugPrescriptionComparison-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear that is subsumed by the concept of the rule family's scope. Thus, there are overlapping facts and the judgement rule family applies to the evaluation cube

Table 2 shows the evaluation of judgement rule family DrugCostIncreaseJudgement over the facts of cube drugCostRatio-PerYearDistrictToProvince-InCoreProvince-Vs-PreviousYear. Judgement rule DCIJ-root applies to the provinces of *Styria* and *Lower Austria* and its districts and has as condition a drug cost increase of more than *5 %*. The comparative facts for *Styria*, *Weiz*, *Melk*, and *Horn* indicate a drug cost decrease or an increase below *5 %* and, thus, do not satisfy the rule condition. The comparative facts for *Leoben*, *Murau*, *Lower Austria*, and *Korneuburg* indicate a drug cost increase of more than *5 %* and are, thus, annotated with the judgement of rule DCIJ-root. Judgement rule DCIJ-UA applies to the province of *UpperAustria* and its districts and has as condition a drug cost increase of more than *2 %*. The comparative facts for *Upper Austria*, *Wels-Stadt*, and *Eferding* indicate a drug cost decrease or an increase not above *2 %* and thus, do not satisfy the rule condition. The comparative fact for *Linz-Stadt* indicates a drug cost increase of more than *2 %* and is, thus, annotated with the judgement of rule DCIJ-UA informing about a program in Upper Austria that promotes the prescription of generic drugs.

poi.time	poi.patient	poi.drug	Ratio	Rule	Judgement
2012	Styria	all	0.99	(null)	(null)
2012	Leoben	all	1.08	DCIJ-root	Since 2012, doctors ...
2012	Murau	all	1.21	DCIJ-root	Since 2012, doctors ...
2012	Weiz	all	0.89	(null)	(null)
2012	Lower Austria	all	1.10	DCIJ-root	Since 2012, doctors ...
2012	Melk	all	0.97	(null)	(null)
2012	Korneuburg	all	1.07	DCIJ-root	Since 2012, doctors ...
2012	Horn	all	1.03	(null)	(null)
2012	Upper Austria	all	1.02	(null)	(null)
2012	Linz-Stadt	all	1.05	DCIJ-UA	In Upper Austria in ...
2012	Wels-Stadt	all	0.98	(null)	(null)
2012	Eferding	all	0.93	(null)	(null)

Table 2: Judgement rule evaluation

4.2 Decision-Scope-based Contextualised Evaluation of Analysis Rules

Schrefl et al. (2013) propose the decision-scope approach to specialisation of business rules, which is inspired by a principle commonly applied in organisational contexts and in law: entities at higher organisation levels set a decision scope within which entities at lower organisation levels may operate. Schrefl et al. also show how to apply the decision-scope approach to rules in data warehousing.

In this subsection we extend the decision-scope approach to comparative data analysis and explain the adapted forms of the *presumed* and the *prerogative* rule evaluation strategies.

An *analysis rule evaluation* is specified by a set of to-be evaluated analysis rule families, a list of comparison granularities, a comparative concept, and a choice between prerogative and presumed evaluation. The comparative concept specifies, together with the scores of the analysis rule families, the comparative facts over which the rules are to be evaluated. The list of comparison granularities specifies a top-down evaluation path of granularities, from coarse-grained to fine-grained. The choice between prerogative and presumed evaluation specifies how to deal with exceptional cases for points at finer granularities that are contrary to decisions for points at coarser granularities. This difference is explained later and summarised in Fig. 12.

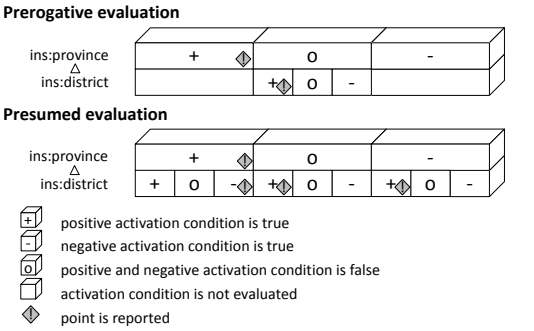

Figure 12: Analysis Rule Evaluation Strategies

The basic idea of the extension of the decision-scope approach to comparative rule evaluation is that if a point pair satisfies the positive or negative activation condition, the same activation is implied for point pairs at finer comparison granularities that roll up to this point pair. If neither activation condition is true for a given point pair, the conditions for point pairs at the next finer granularity of the evaluation path are evaluated.

Application of the decision-scope approach leads to two independent hierarchies during rule evaluation. The rule hierarchy represents knowledge about different conditions valid in different contexts; the evaluation path, on the other hand, defines a hierarchy of precedence during analysis rule evaluation.

4.2.1 Prerogative Evaluation

The *prerogative evaluation strategy* (Schrefl et al., 2013) states that rules defined on a higher or more general level always precede rules on lower levels and, therefore, constitutes a top-down evaluation approach. This principle is adapted to comparative rule evaluation as follows. If either the positive or negative activation condition is satisfied by a point pair on some comparison granularity on the evaluation path (starting with point pairs at the coarsest granularity), evaluation for this point pair as well as for point pairs that roll up to this point pair stops. On the next finer comparison granularity, as defined by the evaluation path, rules are only evaluated for point pairs that roll up to a previously undecided point pair.

Table 3 shows the result of the prerogative evaluation of rule family DrugCostIncreaseReport along dimension patient on the comparative facts of comparative cube drugCostRatio-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear, which represents the application of the score of the analysis rule family to the point pairs defined by comparative concept DrugPrescriptionComparison-PerYearDistrictToProvince-InCoreProvince-VsPreviousYear. The evaluation path consists of two comparison granularities that differ in the level of dimension patient, the first has level district and the second has level province for dimension patient, and the other dimensions are fixed to levels year and top. Highlighted tupels constitute the actual information contained in the analysis report, which is presented to the business analyst. The fact for province *Styria* satisfies the negative activation condition, it is thus not part of the analysis report and the rule is not evaluated for districts in *Styria*. The

poi.time	poi.patient	poi.drug	Ratio	Rule	Act.	Report
2012	Styria	all	0.99	DCIR-root	-	
2012	Leoben	all	1.04	not evaluated		
2012	Murau	all	1.21	not evaluated		
2012	Weiz	all	0.89	not evaluated		
2012	**Lower Austria**	all	**1.10**	**DCIR-root**	+	x
2012	Melk	all	0.97	not evaluated		
2012	Korneuburg	all	1.07	not evaluated		
2012	Horn	all	1.06	not evaluated		
2012	Upper Austria	all	1.02	DCIR-UA	o	
2012	**Linz-Stadt**	all	**1.05**	**DCIR-UA**	+	x
2012	Wels-Stadt	all	0.98	DCIR-UA	o	
2012	Eferding	all	0.93	DCIR-UA	-	

Table 3: Prerogative analysis rule evaluation

poi.time	poi.patient	poi.drug	Ratio	Rule	Act.	Report
2012	Styria	all	0.99	DCIR-root	-	
2012	Leoben	all	1.04	DCIR-root	o	
2012	**Murau**	all	**1.21**	**DCIR-root**	**+**	x
2012	Weiz	all	0.89	DCIR-root	-	
2012	**Lower Austria**	all	**1.10**	**DCIR-root**	**+**	x
2012	**Melk**	all	**0.97**	**DCIR-root**	**-**	x
2012	Korneuburg	all	1.07	DCIR-root	+	
2012	Horn	all	1.03	DCIR-root	o	
2012	Upper Austria	all	1.02	DCIR-UA	o	
2012	**Linz-Stadt**	all	**1.05**	**DCIR-UA**	**+**	x
2012	Wels-Stadt	all	0.98	DCIR-UA	o	
2012	Eferding	all	0.93	DCIR-UA	-	

Table 4: Presumed analysis rule evaluation

province of *Lower Austria* satisfies the positive activation condition and is therefore included in the analysis report. As in the case of *Styria* evaluation stops at this granularity. The province of *Upper Austria* satisfies neither activation condition of the contextualised rule DCIR-UA. Therefore, a detailed analysis of districts in *Upper Austria* has to be conducted. On the district level, *Linz-Stadt* does satisfy the positive activation condition and is included in the analysis report. The rule evaluation results in an analysis report containing *Lower Austria* and *Linz-Stadt* as areas that might be suitable targets of future cost reduction programs.

4.2.2 Presumed Evaluation

The *presumed evaluation strategy* (Schrefl et al., 2013) is best suited for reports as it allows to augment the results of prerogative evaluation with exceptional cases on finer granularities. The underlying idea is that results on finer granularities are reported if they contradict a previously reported evaluation on a coarser granularity. This behaviour enables detailed insights for business analysts, while preventing information overload.

Table 4 shows the same analysis rule evaluation as Table 3 but employing the presumed evaluation strategy. The comparative fact for the province of *Styria* satisfies the negative activation condition and is, thus, not part of the analysis report due to its negative activation. Nevertheless, Styrian districts *Leoben*, *Murau*, and *Weiz* are checked for a contradicting positive activation. *Murau* satisfies the positive activation condition and is added to the analysis report. Province *Lower Austria* satisfies the positive activation condition of rule DCIR-root and is therefore part of the analysis report. Consequently, the Lower Austrian districts *Melk*, *Korneuburg*, and *Horn* are checked for a negative activation. District *Melk* satisfies the negative activation condition and is therefore also included in the analysis report.

The final evaluation result of the example set contains the facts for *Murau*, *Lower Austria* and *Linz-Stadt*, which are reported as positive activation, and the fact for *Melk*, which is reported as negative activation as it contradicts the positive activation for *Lower Austria*.

5 Prototype Implementation

In the semCockpit project, we developed a research prototype for ontology-driven comparative data analysis with the following abstract architecture: The semCockpit data warehouse (semDWH) contains multiple OLAP cubes and provides relational views on dimensions, facts, and MDO concepts. The multidimensional ontology, containing concept and rule definitions as presented in this work, is implemented in a relational database called MDO DB. A mapping component, implemented as stored procedures within the MDO DB, computes translations of MDO concept definitions to corresponding SQL and OWL statements. These translations are persisted in separate tables within the MDO DB. Neumayr et al. (2013) introduce the representation of MDO concepts in SQL and OWL. SQL representations of MDO concepts are used for the querying of OLAP cubes in the semDWH. A reasoning component uses the OWL representation of MDO concepts in order to derive subsumption hierarchies by using the off-the-shelf OWL reasoner HermiT. Comparative concepts go beyond the expressiveness of OWL and thus can only be partially mapped to OWL and subsumption reasoning is sound but incomplete. The prototype system is accessed through a web-based user interface.

The base workflow of the prototype system is as follows. The business analyst creates new MDO elements such as concepts, scores, or cubes through the frontend; the frontend persists the elements inside the MDO DB by executing the necessary SQL-statements; other components of the prototype are notified of changes in the MDO through a set of triggers and adapt to the new state of the MDO. This results, for example, in new inserts or updates to the tables holding the OWL and SQL representations. The updated translations are used to create or replace relational views in the semDWH and to derive subsumption hierarchies. Finally, the created views are queried in the semDWH and displayed to the user.

Comparative rules, like other MDO concepts, are represented as relational views in the semDWH. A judgement rule view consists of the facts of its comparative cube that fulfil the rule condition together with the annotation of the defined judgement and the identifier of the rule. The semDWH representation of an analysis rule consists of two relational views, one containing the points with positive activation and one those with negative activation.

The relational representation of comparative rule families is defined based on the rule views of its member rules. Analysis rule families, just like analysis rules, are represented by two relational views in the semDWH. A comparative rule family view is constructed as the union of its member rule views, where each rule view is restricted to the comparative facts for which it is the most specific rule in the rule hierarchy.

The rule evaluation component of the prototype is implemented as stored procedures within the MDO DB. Implicit or explicit execution of a rule evaluation triggers a procedure call and leads to the creation of an SQL-query based on the involved rules and concepts which can be executed in the semDWH

in order to retrieve the analysis report. Steiner (2014) provides further details on the implementation of the semCockpit prototype, especially the implementation of comparative rules and the rule evaluation component.

6 Related Work

Ontology-based BI has received considerable attention and different applications and approaches have been proposed. Ontologies can be utilised during the data warehouse design process for automating schema generation tasks (Romero and Abelló, 2007; Khouri and Ladjel, 2010; Sciarrone et al., 2009; Nebot et al., 2009) or used as so-called Semantic Dimensions (Anderlik et al., 2012) for OLAP. Nebot et al. (2009) provide a framework for designing multidimensional analysis models over the semantic annotations stored in a semantic DWH. Their approach allows the analysis of data by using traditional OLAP operators (Nebot and Llavori, 2012; Nebot et al., 2009). For further discussion of related work also see Neuböck et al. (2013) and Abelló et al. (2014).

Neuböck et al. (2013) provide a high-level overview of the ontology-driven business intelligence approach developed in the semCockpit project, which was first described from a requirements perspective by Neumayr et al. (2011), with examples given in an ad-hoc notation, but lacking a precise specification of the structure of comparative multidimensional ontologies and lacking a detailed treatment of rule modelling and rule evaluation. Neumayr et al. (2012) report on preliminary work on multidimensional ontologies, then with a transformation of concept definitions to Datalog. Neumayr et al. (2013) describe dimensional and multidimensional concepts and their mapping to SQL and OWL, together with entity concepts which are used in the definition of dimensional concepts.

Current business intelligence solutions provide some rule functionality (Browne et al., 2010; Greenwald et al., 2007). For example, Oracle databases support a feature called Delivers by defining alerts that trigger based on user-specified conditions and lead to, e.g., an email notification (Greenwald et al., 2007, p. 242). As a different example, IBM Cognos supports a sophisticated comment function that can be used to annotate judgements to specific reports (Browne et al., 2010, p. 212). Note, however, that this functionality is different to judgement rules in that annotations have to be created and maintained by the user and are not automatically generated based on rule definitions.

7 Conclusion

In this paper we introduced a metamodel for the representation of comparative multidimensional ontologies, including comparative scores, comparative concepts, and comparative cubes. Building on these constructs we introduced the modelling of judgement and analysis rules, their organisation in rule families, their specialisation along subsumption hierarchies of comparative concepts, and their contextualised evaluation according to different rule evaluation strategies.

Ongoing and future work includes mechanisms for an automatic triggering of analysis rules together with an action execution model. Further, we are working on generic rules, that is, rules that can be parameterised by multidimensional and comparative concepts, and on guidance rules, that is, rules that guide the business analyst through the comparative data analysis process.

References

Abelló, A., Romero, O., Pedersen, T., Berlanga Llavori, R., Nebot, V., Aramburu, M. and Simitsis, A. (2014), Using semantic web technologies for exploratory OLAP: A survey. *IEEE Transactions on Knowledge and Data Engineering*, PrePrint, IEEE Computer Society.

Anderlik, S., Neumayr, B. and Schrefl, M. (2012), Using domain ontologies as semantic dimensions in data warehouses. *ER 2012*, LNCS Vol. 7532, Springer.

Browne, D., Desmeijter, B., Dumon, R. F., Kamal, A., Leahy, J., Masson, S., Rusak, K., Yamamoto, S. and Keen, M. (2010), IBM cognos business intelligence v10.1 handbook, IBM.

Greenwald, R., Stackowiak, R. and Stern, J. (2007), *Oracle Essentials: Oracle Database 11g*, O'Reilly & Associates, Inc.

Khouri, S. and Ladjel, B. (2010), A methodology and tool for conceptual designing a data warehouse from ontology-based sources. *DOLAP 2010*, ACM.

Nebot, V., Berlanga, R., Pérez, J., Aramburu, M. and Pedersen, T. (2009), Multidimensional integrated ontologies: A framework for designing semantic data warehouses. Journal on Data Semantics XIII, Springer.

Nebot, V. and Llavori, R. B. (2012), Building data warehouses with semantic web data. *Decision Support Systems* **52**(4).

Neuböck, T., Neumayr, B., Schrefl, M. and Schütz, C. (2013), Ontology-driven business intelligence for comparative data analysis. *eBISS 2013*, LNBIP Vol. 172, Springer.

Neumayr, B., Anderlik, S. and Schrefl, M. (2012), Towards ontology-based olap: datalog-based reasoning over multidimensional ontologies. *DOLAP 2012*, ACM.

Neumayr, B., Schrefl, M. and Linner, K. (2011), Semantic cockpit: An ontology-driven, interactive business intelligence tool for comparative data analysis. *ER 2011 Workshops*, LNCS Vol. 6999, Springer.

Neumayr, B., Schütz, C. and Schrefl, M. (2013), Semantic enrichment of olap cubes: Multidimensional ontologies and their representation in sql and owl. *ODBASE 2013*, LNCS Vol. 8185, Springer.

Romero, O. and Abelló, A. (2007), Automating multidimensional design from ontologies. *DOLAP 2007*, ACM.

Schrefl, M., Neumayr, B. and Stumptner, M. (2013), The decision-scope approach to specialization of business rules: Application in business process modeling and data warehousing. *APCCM 2013*, CRPIT Vol. 143, Australian Computer Society.

Sciarrone, F., Starace, P. and Federicit, T. (2009), A business intelligence process to support information retrieval in an ontology-based environment. *ISDA 2009*, IEEE Computer Society.

Steiner, D. (2014), Implementing judgement and analysis rules for comparative data analysis in oracle, Master's thesis, Johannes Kepler University Linz.