Visual Query Examples

Mini-Hospital data schema

The visual query environment needs a data schema defining the available class and property vocabulary, as well as applicability of properties to classes and cardinalities.

The following visualization shows a mini-hospital data schema used in further examples.

The center of the data schema is the Patient class; each patient can have a number of hospital episodes (HospitalEpisode), and a number of outpatient episodes (OutpatientEpisode). There can be a number of treatments in wards (TreatmentInWard) for each hospital episode, numbered starting by 1 in the orderNo attribute. The attribute isLastTr (values “true” or “false”) marks each treatment in ward as the last one or non-last one for its episode. Further on, every treatment in ward can have a number of associated manipulations (Manipulation).

There are cardinality marks (e.g. [0..1] or *) at some attributes or links. The attributes or links not having any cardinality mark are assumed to have cardinality 1 that means existence of exactly one required value corresponding to the attribute or the link.

The classes CManipDescr, CPhysician and CDiagnosis contain classifier values of manipulation descriptions, physicians and diagnoses respectively. There are links to the classifier classes from the main data classes. In the query building process these links can be traversed both in the direction from the data class to the classifier class, as well as in the reverse direction.

For every hospital episode there can be several admission and several discharge diagnoses specified; one of these diagnoses is marked as the main admission or discharge diagnosis respectively (accessible from the episode by link admissionDiagnosis_main or dischargeDiagnosis_main), all episode diagnoses that are not main admission or discharge diagnoses are extra diagnoses (accessible from the episode by the link admissionDiagnosis_extra or dischargeDiagnosis_extra). The links admissionDiagnosis and dischargeDiagnosis can be used to reach all admission or discharge diagnoses (including the main ones and the extra ones) for an episode, respectively.

Similarly, for an outpatient episode the outpatientDiagnosis link lists all diagnoses, outpatientDiagnosis_main lists the main diagnosis and outpatientDiagnosis_extra lists the extra diagnoses.
Simple and attribute-based statistics

**count(::) OutpatientEpisode**
- Count all outpatient episodes

**count(::) OutpatientEpisode**
- patient gender = "male"
- Count outpatient episodes costing at least 50 Euro, of male patients

**count(::) OutpatientEpisode**
- visitCost >= 50
- Count outpatient episodes costing at least 60 Euro, grouped by patient's gender

**Patient**
- exists(hospitalEpisode)
- not exists(outpatientEpisode)
- Count (distinct!) patients having a hospital episode and not having an outpatient episode

**count(::) TreatmentInWard**
- For each ward, count the number of treatments in the ward

**count(::) OutpatientEpisode**
- M=month(visitDate)
- order by M
- avg(visitDuration)
- For each month count and calculate the average (duration of outpatient episodes) in the month. Order by the month.
Subqueries: Statistics around data items (class instances)

**Subquery for Patient**
- For every patient with an outpatient episode, count the outpatient episodes. Show patient’s name, surname and outpatient episode count, order by the outpatient episode count.
- `C count( )` on `OutpatientEpisode`

**Subquery for CPhysician**
- Find the count and average outpatient episode visit cost for every physician serving at least one outpatient episode.
- `C count( ) A avg(visitCost)` on `OutpatientEpisode`

**Subquery for CPhysician**
- Count physicians that have at least 10 outpatient episodes that cost at least 50 Euros each.
- `count( ) CPhysician` on `OutpatientEpisode`

**Subquery for Patient**
- List all patients that have at least 10 outpatient episodes.
- `count( ) Patient` on `OutpatientEpisode`

**Subquery for OutpatientEpisode**
- List all patients that have at least one outpatient episode. The subquery is used not to list the patients with several episodes several times.
- `count( ) Patient` on `OutpatientEpisode`

**Subquery for CDiagnosis**
- For every group B diagnosis count the outpatient episodes with the diagnosis as the main outpatient episode diagnosis. Include also diagnoses without any outpatient episodes.
- `C count( )` on `OutpatientEpisode`

**Subquery for HospitalEpisode**
- Count patients with at least 3 hospital episodes running over at least 5 treatments in ward each.
- `T count( )` on `TreatmentInWard`
Advanced examples: expressions, condition links, free links and free nodes

**HospitalEpisode**
- `E >= 2`
- `caseRecordNo`
- `order by C DESC`
- `treatmentInWard`
- `G <= count()`
- `order by G`
- `CA` count each episode once, `CC` count each episode as many times as diagnoses from this group listed as admission diagnoses for the episode.

**TreatmentInWard**
- `hours(arrivalTime-E.admissionTime) <= 24`

**CDiagnosis**
- `A` count each episode once, `D` count each episode as many times as discharge diagnoses.
- `E >= 2`
- `G <= count()`
- `order by G`
- `CA` count each episode once, `CC` count each episode as many times as diagnoses from this group listed as admission diagnoses for the episode.

**HospitalEpisode**
- `dischargeDiagnosis`
- `caseRecordNo`
- `order by C DESC`
- `treatmentInWard`
- `G <= count()`
- `order by G`
- `CA` count each episode once, `CC` count each episode as many times as diagnoses from this group listed as admission diagnoses for the episode.

**OutpatientEpisode**
- `days(H.admissionTime-visitDate) <= 30`
- `visitDate <= H.admissionTime`
- `H >= count()`
- `order by H DESC`
- `personCode`
- `order by C DESC`
- `physician`
- `outpatientDiagnosis`