Cooking with CQL Qs&As – Session 45

Thursday, June 25, 2020

# **Measure Logic in CQL**

**Q:** Regarding the Troponin Continuous Variable measure (which refers to critical value reporting timeliness) and the use of the CQL calculated sort order within the logic expression where the relatedTo events appear first, will this function calculate the combination that has the minimum or least number of minutes between the two?

|  |  |
| --- | --- |
| define function "Minutes between results and communication by id THEN timing"(Encounter "Encounter, Performed"): | |
|  | Min( |
|  | ["Laboratory Test, Performed": "Cardiac Troponin"] T |
|  | let |
|  | Communication: First( |
|  | ["Communication, Performed": "Lab Communications"] C |
|  | where C.sentDatetime during Encounter.relevantPeriod |
|  | and C.sentDatetime after T.resultDatetime |
|  | and (not exists (C.relatedTo) or C.relatedTo includes T.id) |
|  | sort by if exists (relatedTo) then 0 else 1, sentDatetime |
|  | ) |
|  | return minutes between T.resultDatetime and Communication.sentDatetime |
|  | ) |

**A:** Yes, the calculation performed would select the communication associated with this lab test and choose the least number of minutes between the two. For example, if you have a lab test with related communications that span 60 minutes and then you have a lab test with unrelated communications that span 50 minutes, the return would be 60 minutes between T.resultDatetime and the Communication.sentDatetime. The lab test that had related communications would return 60 minutes and the lab test that had no related communications would return 50 minutes. Therefore, the calculation performed would select the related results first, then the minimum number of minutes out of that value set. The point of the exists (relateTo) is to find related items first, with fallback to unrelated only if related results are not present.

**Q:** With respect to the Troponin Continuous Variable measure (which refers to critical value reporting timeliness) and the logic expression defining the function of “Minutes between results and communication by id THEN timing”, should the relatedTo in the expression actually be written as C.relatedTo for sorting purposes?

|  |  |
| --- | --- |
| define function "Minutes between results and communication by id THEN timing"(Encounter "Encounter, Performed"): | |
|  | Min( |
|  | ["Laboratory Test, Performed": "Cardiac Troponin"] T |
|  | let |
|  | Communication: First( |
|  | ["Communication, Performed": "Lab Communications"] C |
|  | where C.sentDatetime during Encounter.relevantPeriod |
|  | and C.sentDatetime after T.resultDatetime |
|  | and (not exists (C.relatedTo) or C.relatedTo includes T.id) |
|  | sort by if exists (relatedTo) then 0 else 1, sentDatetime |
|  | ) |
|  | return minutes between T.resultDatetime and Communication.sentDatetime |
|  | ) |

**A:** No, this is a really important point. The reason we have a C.relatedTo in the previous line is because this portion of the expression is performing a query that is potentially about having multiple items (e.g., from, clauses, with).

|  |
| --- |
| define “Communications Performed”: |
| ["Communication, Performed": "Lab Communications"] C |
| where C.sentDatetime during Encounter.relevantPeriod |
| and C.sentDatetime after T.resultDatetime |
| and (not exists (C.relatedTo) or C.relatedTo includes T.id) |

By the time you get to the sort function of the expression, the query is completed and the shape of the result is established. The alias does not apply anymore. You can refer to things in the sort that happen in the return. Since relatedTo is an attribute of the communication performed returned from the query, it can be used in an expression to determine a sort order. Therefore, the expression of sort is only in terms of the result and does not happen during the query, it happens during the final result of the query.

**Q:** With respect to the Troponin Continuous Variable measure (which refers to critical value reporting timeliness) and the logic expression defining the function of “Minutes between results and communication by id THEN timing”, what is the measure intent and what is the highlighted portion (not exists (C.relatedTo) of the expression looking for?

|  |  |
| --- | --- |
| define function "Minutes between results and communication by id THEN timing"(Encounter "Encounter, Performed"): | |
|  | Min( |
|  | ["Laboratory Test, Performed": "Cardiac Troponin"] T |
|  | let |
|  | Communication: First( |
|  | ["Communication, Performed": "Lab Communications"] C |
|  | where C.sentDatetime during Encounter.relevantPeriod |
|  | and C.sentDatetime after T.resultDatetime |
|  | and (not exists (C.relatedTo) or C.relatedTo includes T.id) |
|  | sort by sentDatetimeif exists (relatedTo) then 0 else 1, sentDatetime |
|  | ) |
|  | return minutes between T.resultDatetime and Communication.sentDatetime |
|  | ) |

**A:** The intent of this measure is to drive standardization of practice with communication from the pathologist to the provider. The challenge was to find the difference in time between tests and their related communication, but account for the fact that the test may not capture the related communication. Therefore, that information may not be available. We came up with two approaches to find the difference in time between tests and their related communication.

* The first approach is to relate the results and communication by the id.
* The second approach is to relate them by the sentDatetime.

We say, if the relationship is not explicitly characterized using relatedTo, then we default to a temporal relationship where we pick the first communication after the result was sent and we will say that is the communication. This is not 100% accurate, but it is a communication that came from the same encounter so it is likely relevant. We are using this as a fallback in case the communication does not capture the relatedTo. Then, we expressed the two approaches as a combined result that said we are doing both at the same time by putting in this list of communications performed and either the communication does not specify the relationship so we need to relate it by a temporal relationship or it explicitly specifies a relationship and we need to use that. If the communication explicitly states that it is related to some lab test, then for those communications, we only want to consider the communications that are related to this lab test. When testing scenarios, we found that in some cases unrelated communication occurred before the related communication. The unrelated communication would override and the unrelated communication would be considered first because the sorting function used was sentDatetime. The not exists (C.relatedTo) function expands that solution to order by related first then sorted by the sentDatetime. This accounts for both related and unrelated communications, but gives priority to the communications that we know are related to this lab test.