Natural language template selection for temporal constraints — Template options

C. Maria Keet

Department of Computer Science, University of Cape Town,
South Africa, mkeet@cs.uct.ac.za

1 Introduction

These introductory notes describe the task, a refresher on the notation of the semantics of the constraints, and some explanatory notes on how to read the sample sentences.

Task For each constraint, fill in the corresponding section in the spreadsheet. The fields that require a value are colour-coded in red. You may either select from the drop-down list in column E or type 'yes', 'sort of', or 'no' in that column. If you have a preferred sentence from the options available—where there is more than one option to choose from—please indicate that with a '1' in column F. Any comments you may have can be added in column G.

Refresher notes on notation of the semantics We rely on the formal foundations of ER_{VT} and EER^{++}_{VT} (with some adjustments to harmonise notation and the extension with mandatory constraints), being the semantics of $\mathcal{DLR}_{\mathcal{US}}$, so as not to clutter this presentation with too much repetition and syntax notation. Considering usual model-theoretic semantics, we use here a *temporal interpretation* of the signature of a conceptual data model $\mathcal{M}$ for this. This is a structure of the form: $\mathcal{I} = \big((\mathbb{Z}, <), \Delta^{\mathcal{I}}, \{\cdot^{\mathcal{I}(t)} \mid t \in \mathbb{Z}\}\big)$, where $(\mathbb{Z}, <)$ is the set of integers denoting the intended *flow of time*, $\Delta^{\mathcal{I}} \neq \emptyset$ is the *interpretation domain* divided into $\Delta^{\mathcal{I}}_C$ over classes and $\Delta^{\mathcal{I}}_D$ over data types, and $\cdot^{\mathcal{I}(t)}$, for $t \in \mathbb{Z}$, is the *interpretation function* which assigns a set $C^{\mathcal{I}(t)} \subseteq \Delta^{\mathcal{I}}$ to each entity type $C \in \mathcal{C}$, a set $R^{\mathcal{I}(t)}$ of tuples over $\Delta^{\mathcal{I}}_C \times \Delta^{\mathcal{I}}_C$ to each relation $R \in \mathcal{R}$ and a set $A^{\mathcal{I}(t)}$ of tuples over $\Delta^{\mathcal{I}}_C \times \Delta^{\mathcal{I}}_D$ to each attribute $A \in \mathcal{A}$. While $\mathcal{DLR}_{\mathcal{US}}$ permits n-ary relations, we consider here just the case for binaries.

Explanatory notes on the sentences Each element/constraint has an abbreviation and a name. For most constraints, the semantics are given, with the others following the same pattern as the others of the same type. Several constraints also have an example from some hypothetical universe of discourse of a temporal conceptual model, to help filling in the slots of the template sentences. So, for instance in the first case, snapshot class (Sc), a sample entity type is Person, which would then be plugged in into the template at the place of the "..$\mathsf{C_1}$.." slot, resulting in, e.g., "*Each* Person *is always a* Person." for sentence template (g).

Regarding relationships, while the semantics doesn't constrain the classes (does not mention specific entity types), they are always considered in conceptual data models, mainly because there is a practice of reusing labels for naming of relationships (e.g., multiple times 'has') so that such context is necessary in the verbalisation of conceptual models (cf. ontologies). Therefore, we include the entity types in the templates.

1.1 Basics: snapshot and temporal

(SC) Snapshot class, $o \in C^{\mathcal{I}(t)} \rightarrow \forall t' \in \mathcal{T}.o \in C^{\mathcal{I}(t')}$; for instance, Person:

(a) *If an object is an instance of entity type* ..$\mathtt{C}_1$.. , *then it is a(n)* ..$\mathtt{C}_1$.. *at all times.*

(b) ..$\mathtt{C}_1$.. *is an entity type whose objects will be a(n)* ..$\mathtt{C}_1$.. *for their whole existence.*

(c) ..$\mathtt{C}_1$.. *is an entity type whose objects will be a(n)* ..$\mathtt{C}_1$.. *at all times.*

(d) ..$\mathtt{C}_1$.. *is an entity type whose objects will always be a(n)* ..$\mathtt{C}_1$.. .

(e) ..$\mathtt{C}_1$.. *is an entity type whose objects are necessarily an instance of* ..$\mathtt{C}_1$.. .

(f) *Each* ..$\mathtt{C}_1$.. *is a(n)* ..$\mathtt{C}_1$.. *at all times.*

(g) *Each* ..$\mathtt{C}_1$.. *is always a(n)* ..$\mathtt{C}_1$.. .

(TC) Temporary class, $o \in C^{\mathcal{I}(t)} \rightarrow \exists t' \neq t.o \notin C^{\mathcal{I}(t')}$; for instance, Undergraduate Student (assuming they graduate and end up as alumni or drop outs):

(a) *If an object is an instance of entity type* ..$\mathtt{C}_1$.. , *then there is some time where it is not a(n)* ..$\mathtt{C}_1$.. .

(b) ..$\mathtt{C}_1$.. *is an entity type whose objects are, for some time in their existence,* not *instances of* ..$\mathtt{C}_1$.. .

(c) ..$\mathtt{C}_1$.. *is an entity type of which each object is not a(n)* ..$\mathtt{C}_1$.. *for some time during its existence.*

(d) *All instances of entity type* ..$\mathtt{C}_1$.. *are not a(n)* ..$\mathtt{C}_1$.. *for some time.*

(e) Each ..$\mathtt{C}_1$.. *is not a(n)* ..$\mathtt{C}_1$.. *for some time.*

(f) Each ..$\mathtt{C}_1$.. *is for some time not a(n)* ..$\mathtt{C}_1$.. .

(SR) Snapshot relationship, $r \in R^{\mathcal{I}(t)} \rightarrow \forall t' \in \mathcal{T}.r \in R^{\mathcal{I}(t')}$. For instance, Producer produces Movie.

(a) *Each fact of* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *persists while the respective participating instances of* ..$\mathtt{C}_1$.. *and* ..$\mathtt{C}_2$.. *exist.*

(b) *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *endures indefinitely.*

(TR) Temporal relationship, $r \in R^{\mathcal{I}(t)} \rightarrow \exists t' \neq t.r \notin R^{\mathcal{I}(t')}$. For instance, Person married-to Person in a conceptual model for a census database (as there is assumed to be either a divorce or widowhood at some point).

(a) *The objects in the facts in* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *do, at some time, not relate through* ..$\mathtt{R}_1$.. .

(b) *The objects participating in a fact in* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *do* not *relate through* ..$\mathtt{R}_1$.. *at some time.*

(SA) Snapshot attribute, $o \in C^{\mathcal{I}(t)} \wedge \langle o, d\rangle \in A^{\mathcal{I}(t)} \rightarrow \forall t' \in \mathcal{T}.\langle o, d\rangle \in A^{\mathcal{I}(t')}$; For instance, Network Card has MAC Address `String`, where there attribute $\mathtt{A}_1$ is has MAC Address. Note that the template would drop '*has*' if it is already in the attribute's name, or it is substituted with the verb in the attribute's name:

(a) *Each object in entity type* ..$\mathtt{C}_1$.. *having attribute* ..$\mathtt{A}_1$.. *has* ..$\mathtt{A}_1$.. *at all times.*
(b) *Each* ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. *and this holds always.*
(c) *Each* ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. *and objects in* ..$\mathtt{C}_1$.. *always have attribute* ..$\mathtt{A}_1$.. .
(d) ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. *and* ..$\mathtt{C}_1$..s *always have a(n)* ..$\mathtt{A}_1$.. .
(e) *A(n)* ..$\mathtt{C}_1$.. *always has a(n)* ..$\mathtt{A}_1$.. .

(TA) Temporal attribute, $o \in C^{\mathcal{I}(t)} \wedge \langle o, d\rangle \in A^{\mathcal{I}(t)} \rightarrow \exists t' \neq t.\langle o, d\rangle \notin A^{\mathcal{I}(t')}$. For instance, the temporal counterpart for this constraint with Employee receives Bonus `Integer` together with option (e) would generate the sentence 'An Employee receives a Bonus, but not always':

(a) *Each object in entity type* ..$\mathtt{C}_1$.. having attribute ..$\mathtt{A}_1$.. *does not have a(n)* ..$\mathtt{A}_1$.. *at some time.*
(b) *Each* ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. , *but not for some time.*
(c) *Each* ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. , *but objects in* ..$\mathtt{C}_1$.. *do not always have a(n)* ..$\mathtt{A}_1$.. .
(d) ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. , *but* ..$\mathtt{C}_1$..s *do not always have a(n)* ..$\mathtt{A}_1$.. .
(e) *A(n)* ..$\mathtt{C}_1$.. *has a(n)* ..$\mathtt{A}_1$.. , *but not always.*

2 Dynamic constraints

2.1 Dynamic constraints for classes

(DEX) Dynamic extension in the future, i.e., $o \in \textsc{Dex}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C}_1{}^{\mathcal{I}(t)} \wedge o \notin \mathtt{C}_2{}^{\mathcal{I}(t)} \wedge o \in \mathtt{C}_2{}^{\mathcal{I}(t+1)})$. For instance, Employee may extend dynamically to Manager:

a. *A(n)* ..$\mathtt{C}_1$.. *may also become a(n)* ..$\mathtt{C}_2$.. .
b. *A(n)* ..$\mathtt{C}_1$.. *may also be a(n)* ..$\mathtt{C}_2$.. *at a later time.*
c. *A(n)* ..$\mathtt{C}_1$.. *may also become a(n)* ..$\mathtt{C}_2$.. *at a later time.*
d. *A(n)* ..$\mathtt{C}_1$.. *may also be a(n)* ..$\mathtt{C}_2$.. *some time in the future.*

(DEXM) Mandatory DEX, i.e., DEX and 'source total transition' apply, meaning that *all* objects instantiating C_1 will also instantiate C_2 at some point in time in the future: $o \in \textsc{DexM}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C}_1{}^{\mathcal{I}(t)} \rightarrow \exists t' > t.o \in \textsc{Dex}^{\mathcal{I}(t')}_{C_1,C_2})$. For instance, all Top-1 songs of the weekly charts will be added as a AlbumTrack on the 'best of' album of that year:

a. *Each* ..$\mathtt{C}_1$.. *must also become a(n)* ..$\mathtt{C}_2$.. .
b. *Each* ..$\mathtt{C}_1$.. *also has to become a(n)* ..$\mathtt{C}_2$.. .
c. *Each* ..$\mathtt{C}_1$.. *has to become a(n)* ..$\mathtt{C}_2$.. *as well.*
d. *Each* ..$\mathtt{C}_1$.. *will also become a(n)* ..$\mathtt{C}_2$.. .

e. *Each ..*$\mathtt{C}_1$*.. will also be a(n) ..*$\mathtt{C}_2$*.. at a later time.*
f. *Each ..*$\mathtt{C}_1$*.. will also become a(n) ..*$\mathtt{C}_2$*.. at a later time.*
g. *Each ..*$\mathtt{C}_1$*.. will also be a(n) ..*$\mathtt{C}_2$*.. some time in the future.*

(DEX$^-$) Dynamic extension in the past, i.e., $o \in \text{DEX}^{-\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \wedge o \notin \mathtt{C_2}^{\mathcal{I}(t)} \wedge o \in \mathtt{C_2}^{\mathcal{I}(t-1)})$. For instance, a Manager at a company was already an Employee at that company:

a. *A(n) ..*$\mathtt{C}_1$*.. may have been already a(n) ..*$\mathtt{C}_2$*.. .*
b. *A(n) ..*$\mathtt{C}_1$*.. may have been a(n) ..*$\mathtt{C}_2$*.. before.*
c. *A(n) ..*$\mathtt{C}_1$*.. may have been a(n) ..*$\mathtt{C}_2$*.. some time earlier.*

(DEXM$^-$) Mandatory dynamic extension, past: $o \in \text{DEXM}^{-\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \rightarrow \exists t' < t.o \in \text{DEX}^{\mathcal{I}(t')}_{C_1,C_2})$. For instance, one can only be a PhD supervisor if one is a Professor, and all professors are going to have to supervise PhD students, according to university management.

a. *Each ..*$\mathtt{C}_1$*.. must be already a(n) ..*$\mathtt{C}_2$*.. .*
b. *Each ..*$\mathtt{C}_1$*.. was already a(n) ..*$\mathtt{C}_2$*.. .*
c. *Each ..*$\mathtt{C}_1$*.. was already a(n) ..*$\mathtt{C}_2$*.. before.*
d. *Each ..*$\mathtt{C}_1$*.. was already a(n) ..*$\mathtt{C}_2$*.. some time earlier.*

(DEV) Dynamic evolution, future, optional: $o \in \text{DEV}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \wedge o \notin \mathtt{C_2}^{\mathcal{I}(t)} \wedge o \in \mathtt{C_2}^{\mathcal{I}(t+1)} \wedge o \notin \mathtt{C_1}^{\mathcal{I}(t+1)})$. For instance, a Caterpillar is expected to turn into a Butterfly (just not all do in praxis).

a. *A(n) ..*$\mathtt{C}_1$*.. may evolve to become a(n) ..*$\mathtt{C}_2$*.. ceasing to be a(n) ..*$\mathtt{C}_1$*.. .*
b. *A(n) ..*$\mathtt{C}_1$*.. may evolve to ..*$\mathtt{C}_2$*.. afterward, ceasing to be a(n) ..*$\mathtt{C}_1$*.. .*

(DEVM) Mandatory dynamic evolution, future: $o \in \text{DEVM}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \rightarrow \exists t' > t.o \in \text{DEV}^{\mathcal{I}(t')}_{C_1,C_2})$. For instance, at a university where each Student eventually will receive a certificate and thus become an Alumnus (and never study there again), the constraint for the database may be:

a. *Each ..*$\mathtt{C}_1$*.. must evolve to ..*$\mathtt{C}_2$*.. ceasing to be a(n) ..*$\mathtt{C}_1$*.. .*
b. *Each ..*$\mathtt{C}_1$*.. will evolve to ..*$\mathtt{C}_2$*.. ceasing to be a(n) ..*$\mathtt{C}_1$*.. .*
c. *Each ..*$\mathtt{C}_1$*.. will evolve to ..*$\mathtt{C}_2$*.. some time in the future, ceasing to be a(n) ..*$\mathtt{C}_1$*.. .*

(DEV$^-$) Dynamic evolution, past, optional: $o \in \text{DEV}^{-\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \wedge o \notin \mathtt{C_2}^{\mathcal{I}(t)} \wedge o \in \mathtt{C_2}^{\mathcal{I}(t-1)} \wedge o \notin \mathtt{C_1}^{\mathcal{I}(t-1)})$. For instance, for a university database: while it is common now for an Academic to move between universities, some stay at the same place, or return to, where they have been a PhD student.

a. *A(n) ..*$\mathtt{C}_1$*.. may have been a(n) ..*$\mathtt{C}_2$*.. before, but is not a(n) ..*$\mathtt{C}_2$*.. now.*
b. *If ..*$\mathtt{C}_1$*.. , then ..*$\mathtt{C}_1$*.. may have been a(n) ..*$\mathtt{C}_2$*.. before, but is not a(n) ..*$\mathtt{C}_2$*.. now.*

(DevM$^-$) Mandatory dynamic evolution, past: $o \in \text{DevM}^{-\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \rightarrow \exists t' < t.o \in \text{Dev}^{\mathcal{I}(t')}_{C_1,C_2})$. For instance, Butterfly and the Caterpillar it used to be.

a. *Each* ..$\mathtt{C_1}$.. *must have been a(n)* ..$\mathtt{C_2}$.. *, but is not a(n)* ..$\mathtt{C_2}$.. *anymore.*
b. *Each* ..$\mathtt{C_1}$.. *was a(n)* ..$\mathtt{C_2}$.. *before, but is not a(n)* ..$\mathtt{C_2}$.. *now.*
c. *If* ..$\mathtt{C_1}$.. *, then* ..$\mathtt{C_1}$.. *was a(n)* ..$\mathtt{C_2}$.. *before, but is not a(n)* ..$\mathtt{C_2}$.. *anymore.*

(PDex/PDev) Persistent extension or evolution; for instance, once a conference paper has been published in print, that publication is there for posterity in, say, Springer's system. Add at the end of the sentence either one of:

a. <selected Dex/Dev option>*, and this remains so.*
b. <selected Dex/Dev option>*, and this must remain so.*
c. <selected Dex/Dev option>*, and this remains so indefinitely.*
d. <selected Dex/Dev option>*, which does not change.*

For quantitative extension and evolution, we need a specific number, n for counting and, implicitly, some time unit to be able to construct, e.g., 'after at least 3 years'. The number is denoted in the formalisation as n and the variable in the template is denoted as " ..$\mathtt{D_1}$..".

(Qex) Quantitative extension, future, optional, where here and in the following variants, $n \in \mathbb{Z}$ and $t + n \in \mathcal{T}_p$, and for Qex then: $o \in \text{Qex}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow \exists (t+n) > t.(o \in \mathtt{C_1}^{\mathcal{I}(t)} \wedge o \notin \mathtt{C_2}^{\mathcal{I}(t)} \wedge \mathtt{C_2}^{\mathcal{I}(t+n)})$. For instance, an Employee may be promoted to Manager at that company after 3 years of service.

a. *A(n)* ..$\mathtt{C_1}$.. *may also become a(n)* ..$\mathtt{C_2}$.. *after* [at least/at most/exactly] ..$\mathtt{D_1}$.. .
b. *If* ..$\mathtt{C_1}$.. *for* [at least/at most/exactly] ..$\mathtt{D_1}$.. *, then* ..$\mathtt{C_1}$.. *may become a(n)* ..$\mathtt{C_2}$.. *as well.*

(QexM) Quantitative extension, future, mandatory: $o \in \text{Qex}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow (o \in \mathtt{C_1}^{\mathcal{I}(t)} \rightarrow \exists (t+n) > t.o \in \text{Qex}^{\mathcal{I}(t+n)}_{C_1,C_2})$. For instance, all Students have to do some Volunteer service in the second year of their study.

a. *Each* ..$\mathtt{C_1}$.. *will also become a(n)* ..$\mathtt{C_2}$.. *after* [at least/at most/exactly] ..$\mathtt{D_1}$.. .
b. *If* ..$\mathtt{C_1}$.. *for* [at least/at most/exactly] ..$\mathtt{D_1}$.. *, then* ..$\mathtt{C_1}$.. *becomes a(n)* ..$\mathtt{C_2}$.. *as well.*

(Qex$^-$) Quantitative extension, past, optional, where the past counterpart is similar to the future, but then $t - n$.

a. *A* ..$\mathtt{C_1}$.. *may already have been a(n)* ..$\mathtt{C_2}$.. *for* [at least/at most/exactly] ..$\mathtt{D_1}$.. .

(QexM$^-$) Quantitative extension, past, mandatory, where the past counterpart is similar to the future, but then $t' < t$:

a. *Each* ..$\mathtt{C}_1$.. *was already a(n)* ..$\mathtt{C}_2$.. *for* [at least/at most/exactly] ..$\mathtt{D}_1$.. .

(QEV) Quantitative evolution, future:
$o \in \text{QEV}^{\mathcal{I}(t)}_{C_1,C_2} \rightarrow \exists(t+n) > t.(o \in \mathtt{C_1}^{\mathcal{I}(t)} \wedge o \notin \mathtt{C_2}^{\mathcal{I}(t)} \wedge o \in \mathtt{C_2}^{\mathcal{I}(t+n)} \wedge o \notin \mathtt{C_1}^{\mathcal{I}(t+n)})$. For instance, a non-tenured prof may have met the criteria and thus be confirmed as a Associate Prof after 6 years.

a. *A* ..$\mathtt{C}_1$.. *may progress to a(n)* ..$\mathtt{C}_2$.. *after* [at least/at most/exactly] ..$\mathtt{D}_1$.. , *ceasing to be a(n)* ..$\mathtt{C}_1$.. .

(QEVM) Quantitative evolution, future, mandatory): For instance, the fantasy world where all non-tenured profs will get that coveted job of associate prof after 6 years in the tenure track.

a. *Each* ..$\mathtt{C}_1$.. *must progress to a(n)* ..$\mathtt{C}_2$.. *after* [at least/at most/exactly] ..$\mathtt{D}_1$.. *, ceasing to be a(n)* ..$\mathtt{C}_1$.. .

(QEV$^-$) Quantitative evolution, past, optional:

a. *A(n)* ..$\mathtt{C}_1$.. *may have been a(n)* ..$\mathtt{C}_2$.. *before for* [at least/at most/exactly] ..$\mathtt{D}_1$.. *, but is not now.*

(QEVM$^-$) Quantitative evolution, past, mandatory:

a. *Each* ..$\mathtt{C}_1$.. *was a(n)* ..$\mathtt{C}_2$.. *before for* [at least/at most/exactly] ..$\mathtt{D}_1$.. *, but is not now.*

2.2 Dynamic constraints for relationships

(RDEX) – Dynamic extension of a relationship, $\langle o, o' \rangle \in \text{RDEX}^{\mathcal{I}(t)}_{R_1,R_2} \rightarrow (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \rightarrow \exists t' > t.\langle o, o' \rangle \in \mathtt{R_2}^{\mathcal{I}(t')})$. For instance, a Person has made a booking for a Flight, and then that person does a check-in for that flight some time afterward.

a. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .

(RDEXM) – Dynamic extension for relationships, mandatory,
$\langle o, o' \rangle \in \text{RDEXM}^{\mathcal{I}(t)}_{R_1,R_2} \rightarrow (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \rightarrow \exists t' > t.\langle o, o' \rangle \in \text{RDEX}^{\mathcal{I}(t')}_{R_1,R_2})$

a. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *is followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .
b. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *will be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .
c. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *must be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .

(RDEX$^-$) – Dynamic extension for relationships, past, optional, with semantics:
$\langle o, o' \rangle \in {\text{RDEX}^-_{R_1,R_2}}^{\mathcal{I}(t)} \rightarrow (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \rightarrow \exists t' < t.\langle o, o' \rangle \in \mathtt{R_2}^{\mathcal{I}(t')})$.

a. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may be preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .

(RDEXM$^-$) – Dynamic extension for relationships, past, mandatory:
$\langle o, o' \rangle \in {\text{RDEXM}^-_{R_1,R_2}}^{\mathcal{I}(t)} \rightarrow (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \rightarrow \exists t' < t.\langle o, o' \rangle \in {\text{RDEX}^-_{R_1,R_2}}^{\mathcal{I}(t')})$. For instance, every passenger who boards a flight must have checked in.

a. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *is preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .

b. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *must be preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .

(RDev) – Dynamic evolution for relationships, future, optional: $\langle o, o' \rangle \in \text{RDev}^{\mathcal{I}(t)}_{R_1,R_2} \to (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \to \exists t' > t.\langle o, o' \rangle \in \mathtt{R_2}^{\mathcal{I}(t')} \wedge \langle o, o' \rangle \notin \mathtt{R_1}^{\mathcal{I}(t')})$. For instance, a marriage between two persons may end up in a divorce of those two persons.

a. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may be followed sequentially by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .
b. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. , *ending* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. .

(RDevM) – Dynamic evolution for relationships, future, mandatory: $\langle o, o' \rangle \in \text{RDevM}^{\mathcal{I}(t)}_{R_1,R_2} \to (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \to \exists t' > t.\langle o, o' \rangle \in \text{RDev}^{\mathcal{I}(t')}_{R_1,R_2})$. For instance, there may be a business rule that every CEO that works for a Company is going to be an Advisor who advises the Board of that company once they step down as CEO.

a. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *must be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *successively.*
b. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *is followed sequentially by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .
c. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *must be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. , *ending* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. .
d. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *will be followed by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. , *terminating the* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *relation.*

(RDev$^-$) – Dynamic evolution for relationships, past, optional: $\langle o, o' \rangle \in {\text{RDev}^-_{R_1,R_2}}^{\mathcal{I}(t)} \to (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \to \exists t' < t.\langle o, o' \rangle \in \mathtt{R_2}^{\mathcal{I}(t')} \wedge \langle o, o' \rangle \notin \mathtt{R_1}^{\mathcal{I}(t')})$. For instance, books on restricted loan from a library may have been on regular loan from that library (e.g., a book is a high-in-demand textbook for a course in the semester, or it is from a special collection to begin with).

a. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may have been sequentially preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .
b. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may have been preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *and they are not in that* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *relation now.*
c. ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *may have been be preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. , *observing they do not occur concurrently for any pair of objects.*
d. *If* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. , *then it may have been preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. , *but* ..$\mathtt{C}_1$.. *and* ..$\mathtt{C}_2$.. *are not in that* ..$\mathtt{R}_2$.. *relation anymore.*

(RDevM$^-$) – Dynamic evolution for relationships, past, mandatory: $\langle o, o' \rangle \in {\text{RDevM}^-_{R_1,R_2}}^{\mathcal{I}(t)} \to (\langle o, o' \rangle \in \mathtt{R_1}^{\mathcal{I}(t)} \to \exists t' < t.\langle o, o' \rangle \in {\text{RDev}^-_{R_1,R_2}}^{\mathcal{I}(t')})$. For instance, the check-in to a flight surely occurred before boarding, and any pair of humans who are divorced were married before that.

a. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *is strictly preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. .
b. *Each* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. *is preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *and they are not in that* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *relation anymore.*
c. *If* ..$\mathtt{C}_1$.. ..$\mathtt{R}_1$.. ..$\mathtt{C}_2$.. , *then it was preceded by* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *and they are not in that* ..$\mathtt{C}_1$.. ..$\mathtt{R}_2$.. ..$\mathtt{C}_2$.. *relation now.*

d. *Each* ..$\texttt{C}_1$.. ..$\texttt{R}_1$.. ..$\texttt{C}_2$.. *must have been preceded by* ..$\texttt{C}_1$.. ..$\texttt{R}_2$.. ..$\texttt{C}_2$.. *and they are then not in that* ..$\texttt{R}_2$.. *relation anymore.*
e. *If* ..$\texttt{C}_1$.. ..$\texttt{R}_1$.. ..$\texttt{C}_2$.. , *then it must have been preceded by* ...$\texttt{C}_1$.. ..$\texttt{R}_2$.. ..$\texttt{C}_2$.. , *but* ..$\texttt{C}_1$.. *not* ..$\texttt{R}_2$.. ..$\texttt{C}_2$.. *anymore.*

(SRDEX/SRDev) – Persistence, optional or mandatory: same four options as for persistence with classes.

2.3 Dynamic constraints for attributes

(FREEZ) "frozen" attribute. For instance, the Date of Birth of a person or her Social security number which are 'frozen'/persists once that data is added to the database.
a. *Once the value for* ..$\texttt{A}_1$.. *is set, it cannot change anymore.*
b. *Once the value for* ..$\texttt{A}_1$.. *is set, it must remain that value.*
c. *Once the value for* ..$\texttt{A}_1$.. *is set, it must remain the same.*

(AQEV) Quantitative evolution, where a is a binary relation between a class and a data type, $a \in \text{AQEV}^{\mathcal{I}(t)}_{A_1,A_2} \rightarrow \exists (t+n) > t.(a \in \texttt{A}_1{}^{\mathcal{I}(t)} \wedge a \notin \texttt{A}_2{}^{\mathcal{I}(t)} \wedge a \in \texttt{A}_2{}^{\mathcal{I}(t+n)} \wedge a \notin \texttt{A}_1{}^{\mathcal{I}(t+n)})$ where $n \in \mathbb{Z}$. For instance, an employee receives a inflation-adjusted salary increase every two years.
a. *Each* ..$\texttt{C}_1$..*'s* ..$\texttt{A}_1$.. *is updated after* [at least/at most/every] ..$\texttt{D}_1$.. to ..$\texttt{A}_2$...
b. *Each* ..$\texttt{C}_1$..*'s* ..$\texttt{A}_1$.. *changes after* [at least/at most/every] ... to ..$\texttt{A}_2$...
c. *The* ..$\texttt{A}_1$.. *of each* ..$\texttt{C}_1$.. *hast to be updated with* ..$\texttt{A}_2$.. *after* [at least/at most/every] ..$\texttt{D}_1$.. .