ORM Normative Abstract Syntax and Semantics: non-normative glossary

ORM.net Proposed Recommendation

Version: Public BETA 1 - 6 March 2020
Editors: Enrico Franconi, Terry Halpin

Abstract

Object-Role Modeling (ORM) is a rigorous approach to modeling and querying at the conceptual level the information semantics of arbitrary domains. This glossary document lists key terms and symbols used in ORM, and briefly explains their meaning by means of examples. It shows examples of the main graphical conceptual model constructs - namely declarations, constraints, and derivation rules - together with their corresponding abstract syntactic expressions, and their semantics specified as closed first-order logic formulas. This non-normative document makes use of the definitions specified in the normative document defining the abstract syntax and formal semantics of ORM conceptual models. The semantics of an ORM conceptual model is defined by transforming the model to first-order logic axioms, whose finite models denote the legal abstract information structures of the conceptual specification.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of the revisions of this technical report can be found in the ORM.net Technical Recommendations index at <https://gitlab.com/orm-syntax-and-semantics/orm-syntax-and-semantics-docs.git>.

This document is part of the ORM document suite. It summarizes the abstract syntax of the main graphical symbols used in ORM by means of examples. The companion document "ORM Abstract Syntax and Semantics: normative specifications" formally defines the core ORM concepts. Both documents of the ORM document suite can be found at <https://gitlab.com/orm-syntax-and-semantics/orm-syntax-and-semantics-docs.git>.

This document is published on ORM.net as a Proposed Recommendation. If you wish to make comments regarding this document, please send them to <orm-semantics@googlegroups.com>, after having registered at <https://groups.google.com/group/orm-semantics>. All comments are welcome.

Once this document becomes an ORM.net Recommendation, it will be a stable document and may be used as reference material or cited from other documents. ORM.net's role in making the Recommendation is to draw attention to the specification and to promote its widespread deployment. This enhances the functionality and interoperability of data models based on ORM or other fact-based modeling approaches.

Change History

None.

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*

Signature: Entity type name

Signature:

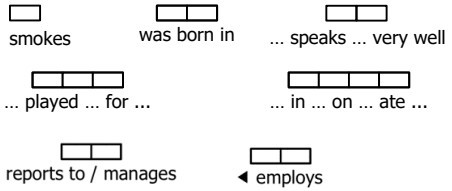
Entity Type name: `Country`

Signature: Value type name

Signature:

Value Type name: `CountryCode`

Signature: Predicate name



Signature:

Unary predicate name: `smokes`

Binary predicate names: `wasBornIn`, `?speaks?veryWell`, `reportsTo`, `employs`

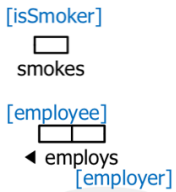
Ternary predicate name: `?played?for?`

Quaternary predicate name: `?in?on?ate?`

Alternate predicate name:

`AlternatePredicate``(reportsTo, manages (2 1))`

Signature: Role name



Signature:

Role identifier for the unary predicate `smokes`*:*
`smokes.1`

Role identifiers for the binary predicate `employs`*:*
`employs.1`, `employs.2`

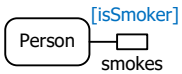
Role names:

`RoleNaming``(smokes.1, smokes.isSmoker)`

`RoleNaming``(employs.1, employs.employer)`

`RoleNaming``(employs.2, employs.employee)`

Unary fact type

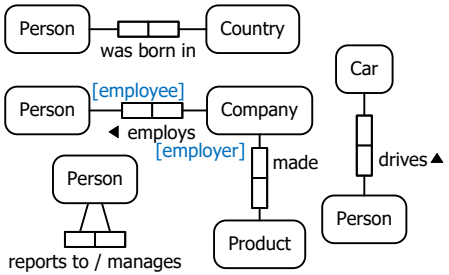


`FactType``(smokes (Person))`

$\forall x.\ \texttt{smokes}\ x \rightarrow \texttt{Person}\ x$

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*

Binary fact type



FactType(wasBornIn (Person Country))

$\forall xy.\ \texttt{wasBornIn}\ x\ y \rightarrow \texttt{Person}\ x\ \&\ \texttt{Country}\ y$

FactType(employs (Company Person))

$\forall xy.\ \texttt{employs}\ x\ y \rightarrow \texttt{Companu}\ x\ \&\ \texttt{Person}\ y$

FactType(made (Company Product))

$\forall xy.\ \texttt{made}\ x\ y \rightarrow \texttt{Company}\ x\ \&\ \texttt{Product}\ y$

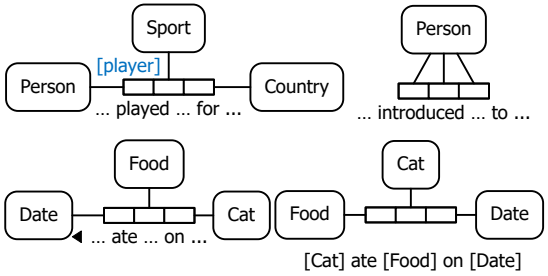
FactType(drives (Person Car))

$\forall xy.\ \texttt{drives}\ x\ y \rightarrow \texttt{Person}\ x\ \&\ \texttt{Car}\ y$

FactType(reportsTo (Person Person))

$\forall xy.\ \texttt{reportsTo}\ x\ y \rightarrow \texttt{Person}\ x\ \&\ \texttt{Person}\ y$

Ternary fact type



FactType(?played?for?
(Person Sport Country))

$\forall xyz.\ \texttt{?played?for?}\ x\ y\ z \rightarrow \texttt{Person}\ x\ \&\ \texttt{Sport}\ y\ \&\ \texttt{Country}\ z$

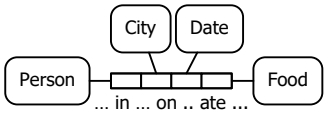
FactType(?introduced?to?
(Person Person Person))

$\forall xyz.\ \texttt{?introduced?to?}\ x\ y\ z \rightarrow \texttt{Person}\ x\ \&\ \texttt{Person}\ y\ \&\ \texttt{Person}\ z$

FactType(?ate?on? (Cat Food Date))

$\forall xyz.\ \texttt{?ate?on?}\ x\ y\ z \rightarrow \texttt{Cat}\ x\ \&\ \texttt{Food}\ y\ \&\ \texttt{Date}\ z$

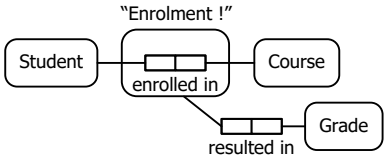
Quaternary fact type



FactType(?in?on?ate? (Person City Date Food))

$\forall xyz.\ \texttt{?in?on?ate?}\ x\ y\ z\ k \rightarrow \texttt{Person}\ x\ \&\ \texttt{City}\ y\ \&\ \texttt{Date}\ z\ \&\ \texttt{Food}\ k$

Objectification



FactType(enrolledIn (Student Course))

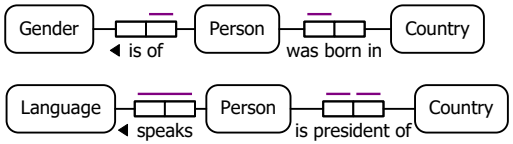
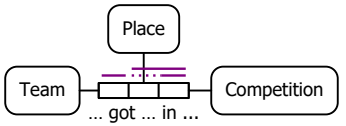
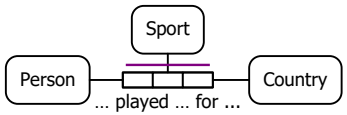
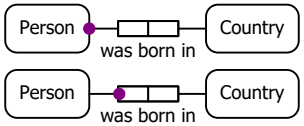
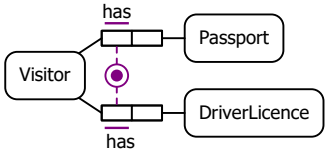
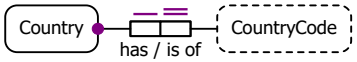
$\forall xy.\ \texttt{enrolledIn}\ x\ y \rightarrow \texttt{Student}\ x\ \&\ \texttt{Course}\ y$

Objectifies(Enrolment enrolledIn)

$\forall xy.\ \texttt{enrolledIn}\ x\ y \leftrightarrow \texttt{Enrolment}\ (\ell_{\texttt{enrolledIn}}(x\ y))$

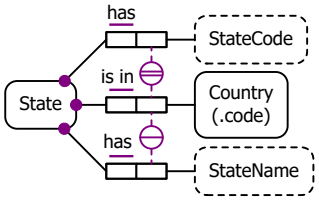
FactType(resultedIn (Enrolment Grade))

$\forall xy.\ \texttt{resultedIn}\ x\ y \rightarrow \texttt{Enrolment}\ x\ \&\ \texttt{Grade}\ y$

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*
UCs on a binary fact type		
	**Unique**(isOf.1) **Unique**(wasBornIn.1) **Unique**(speaks.1 speaks.2) **Unique**(isPresidentOf.1) **Unique**(isPresidentOf.2)	$\forall x_1 x_2.\ \texttt{isOf}\ x_1 x_2 \rightarrow \exists^{=1} y.\ \texttt{isOf}\ x_1 y$ $\forall x_1 x_2.\ \texttt{wasBornIn}\ x_1 x_2 \rightarrow \exists^{=1} y.\ \texttt{wasBornIn}\ x_1 y$ $\forall x_1 x_2.\ \texttt{speaks}\ x_1 x_2 \rightarrow \texttt{speaks}\ x_1 x_2$ $\forall x_1 x_2.\ \texttt{isPresidentOf}\ x_1 x_2 \rightarrow \exists^{=1} y.\ \texttt{ispresidentOf}\ x_1 y$ $\forall x_1 x_2.\ \texttt{isPresidentOf}\ x_1 x_2 \rightarrow \exists^{=1} y.\ \texttt{isPresidentOf}\ y\ x_2$
UCs on ternaries		
	**Unique**(?got?in?.1 ?got?in?.3) **Unique**(?got?in?.2 ?got?in?.3)	$\forall x_1 x_2 x_3.\ \texttt{?got?in?}\ x_1 x_2 x_3 \rightarrow \exists^{=1} y.\ \texttt{?got?in?}\ x_1 y\ x_3$ $\forall x_1 x_2 x_3.\ \texttt{?got?in?}\ x_1 x_2 x_3 \rightarrow \exists^{=1} y.\ \texttt{?got?in?}\ y\ x_2 x_3$
	**Unique**(?played?for?.1 ?played?for?.2 ?played?for?.3)	$\forall x_1 x_2 x_3.\ \texttt{?played?for?}\ x_1 x_2 x_3 \rightarrow \texttt{?played?for?}\ x_1 x_2 x_3$
Simple mandatory role constraint		
	**Mandatory**(Person wasBornIn.1)	$\forall x.\ \texttt{Person}\ x \rightarrow \exists y.\ \texttt{wasBornIn}\ xy$
Inclusive-or constraint		
	**Mandatory**(Visitor hasPassport.1 hasDriverLicence.1)	$\forall x.\ \texttt{Visitor}\ x \rightarrow (\exists y.\ \texttt{hasPassport}\ xy) \vee$ $(\exists y.\ \texttt{hasDriverLicence}\ xy)$
Preferred internal UC		
	**Identification**(Country has.1 (has.2))	$\forall x_1 x_2.\ \texttt{has}\ x_1 x_2 \rightarrow \exists^{=1} y.\ \texttt{has}\ x_1 y$ $\forall x.\ \texttt{Country}\ x \rightarrow \exists y.\ \texttt{has}\ xy$ $\forall x_1 x_2.\ \texttt{has}\ x_1 x_2 \rightarrow \exists^{=1} y.\ \texttt{has}\ y\ x_2$ *well-founded*(has)

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*

External UC



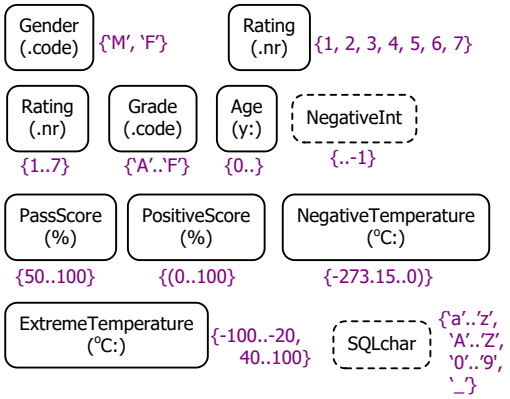
ExternalIdentification(State
(hasStateCode.2 isIn.2))

$\forall x_1x_2x_3.\ \texttt{JP1}\ x_1x_2x_3 \leftrightarrow \exists y.\ \texttt{hasStateCode}\ x_3x_1\ \&\ \texttt{isIn}\ x_3x_2$
$\forall x_1x_2x_3.\ \texttt{JP1}\ x_1x_2x_3 \rightarrow \exists^{=1}y.\ \texttt{JP1}\ x_1x_2y$
$\forall x_1x_2x_3.\ \texttt{JP1}\ x_1x_2x_3 \rightarrow \exists^{=1}y_1y_2.\ \texttt{JP1}\ y_1y_2x_3$
$\forall x.\ \texttt{State}\ x \rightarrow \exists y_1y_2.\ \texttt{K1}\ y_1y_2\ x$
well-founded(hasStateCode) *well-founded*(isIn)

ExternalUnique(hasStateName.2 isIn.2)

$\forall x_1x_2x_3.\ \texttt{JP2}\ x_1x_2x_3 \leftrightarrow \texttt{hasStateName}\ x_3\ x_1\ \&\ \texttt{isIn}\ x_3\ x_2$
$\forall x_1x_2x_3.\ \texttt{JP2}\ x_1x_2x_3 \rightarrow \exists^{=1}y.\ \texttt{JP2}\ x_1x_2y$

Object type value constraint

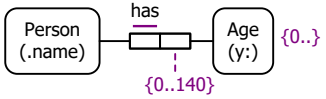


ValuesOf(GenderCode (M F))

…

$\forall x.\ \texttt{GenderCode}\ x \rightarrow x = \texttt{M} \vee x = \texttt{F}$

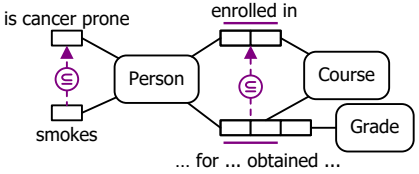
Role value constraint



ValuesOf(has.2 (0 … 140))

$\forall x_1x_2.\ \texttt{has}\ x_1x_2 \rightarrow x_2 = 0 \vee \ldots \vee x_2 = 140$

Subset constraint



Subset((smokes.1 isCancerProne.1))

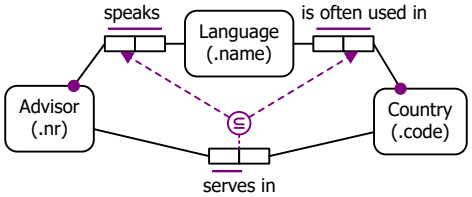
$\forall x.\ \texttt{smokes}\ x \rightarrow \texttt{isCancerProne}\ x$

Subset((?for?obtained?.1 enrolledIn.1)
(?for?obtained?.2 enrolledIn.2))

$\forall x_1x_2x_3.\ \texttt{?for?obtained?}x_1x_2x_3 \rightarrow \texttt{enrolledIn}\ y_1x_2$

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*

Join subset constraint



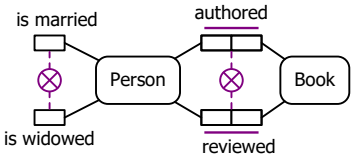

```
JoinPath(P (speaks.1 speaks.2)
           (isOftenUsedIn.1 isOftenUsedIn.2))
```

$\forall x_1 x_2.\ \mathtt{P}\ x_1 x_2 \leftrightarrow \exists y.\ \mathtt{speaks}\ x_1 y\ \&\ \mathtt{isOftenUsedIn}\ y\ x_2$

```
Subset((servesIn.1 P.1)(servesIn.2 P.2))
```

$\forall xy.\ \mathtt{servesIn}\ xy \rightarrow \mathtt{P} xy$

Exclusion constraint



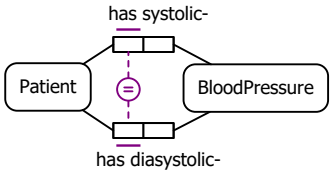

```
Exclusive((isWidowed.1 isMarried.1))
```

$\forall x.\ \mathtt{isWidowed}\ x \rightarrow \sim\mathtt{isMarried}\ x$

```
Exclusive((reviewed.1 authored.1)
          (reviewed.2 authored.2))
```

$\forall xy.\ \mathtt{reviewed}\ xy \rightarrow \sim\mathtt{authored}\ xy$

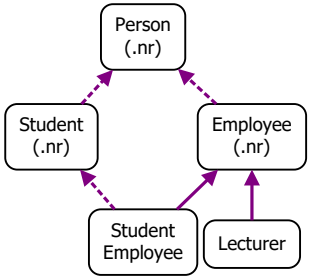
Equality constraint




```
Equal((hasSystolic.1 hasDiasystolic.1))
```

$(\forall xy.\ \mathtt{hasSystolic}\ xy \rightarrow \exists z.\ \mathtt{hasDiasystolic}\ xz) \wedge$
$(\forall xy.\ \mathtt{hasDiasystolic}\ xy \rightarrow \exists z.\ \mathtt{hasSystolic}\ xz)$

Subtyping




```
Subtype(Lecturer Employee)
```

$\forall x.\ \mathtt{Lecturer}\ x \rightarrow \mathtt{Employee}\ x$

```
Subtype(Employee Person)
```

$\forall x.\ \mathtt{Employee}\ x \rightarrow \mathtt{Person}\ x$

```
Subtype(Student Person)
```

$\forall x.\ \mathtt{Student}\ x \rightarrow \mathtt{Person}\ x$

```
Subtype(StudentEmployee Student)
```

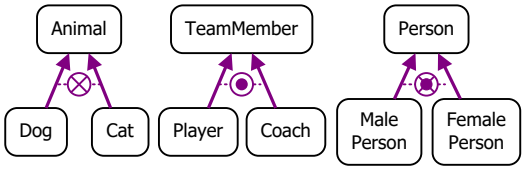
$\forall x.\ \mathtt{StudentEmployee}\ x \rightarrow \mathtt{Student}\ x$

```
Subtype(StudentEmployee Employee)
```

$\forall x.\ \mathtt{StudentEmployee}\ x \rightarrow \mathtt{Employee}\ x$

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*

Subtyping constraints



ExclusiveSubtypes((Dog Cat) Animal)

$(\forall x.\,\texttt{Dog}\ x \rightarrow \texttt{Animal}\ x\ \&\ \sim\texttt{Cat}\ x)\ \&\ (\forall x.\,\texttt{Cat}\ x \rightarrow \texttt{Animal}\ x)$

ExhaustiveSubtypes((Player Coach) TeamMember)

$(\forall x.\,\texttt{Player}\ x \rightarrow \texttt{TeamMember}\ x)\ \&$
$(\forall x.\,\texttt{Coach}\ x \rightarrow \texttt{TeamMember}\ x)\ \&$
$(\forall x.\texttt{TeamMember}\ x \rightarrow \texttt{Coach}\ x\ \texttt{V}\ \texttt{Player}\ x)$

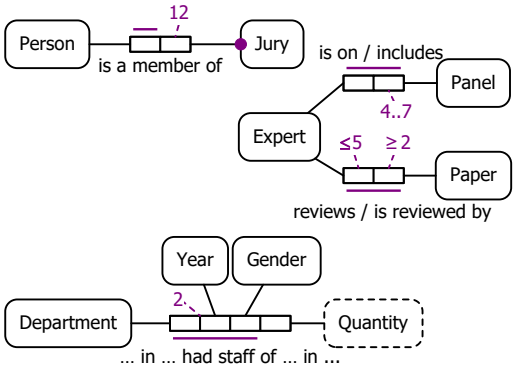
ExclusiveSubtypes((MalePerson FemalePerson) Person)

$(\forall x.\,\texttt{MalePerson}\ x \rightarrow \texttt{Person}\ x\ \&\ \sim\texttt{FemalePerson}\ x)\ \&$
$(\forall x.\,\texttt{Femaleperson}\ x \rightarrow \texttt{Person}\ x)$

ExhaustiveSubtypes((MalePerson FemalePerson) Person)

$(\forall x.\,\texttt{MalePerson}\ x \rightarrow \texttt{Person}\ x)\ \&$
$(\forall x.\,\texttt{FemalePerson}\ x \rightarrow \texttt{Person}\ x)\ \&$
$(\forall x.\texttt{Person}\ x \rightarrow \texttt{FemalePerson}\ x\ \texttt{V}\ \texttt{MalePerson}\ x)$

Internal frequency constraint



Frequency(isAMemberOf.2 (12))

$\forall x_1x_2.\,\texttt{isAMemberOf}\ x_1x_2 \rightarrow \exists^{=12}y.\,\texttt{isAMemberOf}\ x_1y$

Frequency(isOn.2 (4, 7))

$\forall x_1x_2.\,\texttt{isOn}\ x_1x_2 \rightarrow \exists^{\geq 4,\leq 7}y.\,\texttt{isOn}\ x_1y$

Frequency(reviews.1 (..5))

$\forall x_1x_2.\,\texttt{reviews}\ x_1x_2 \rightarrow \exists^{\leq 5}y.\,\texttt{reviews}\ yx_2$

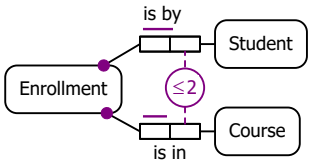
Frequency(reviews.2 (2..))

$\forall x_1x_2.\,\texttt{reviews}\ x_1x_2 \rightarrow \exists^{\geq 2}y.\,\texttt{reviews}\ x_1y$

Frequency(?in?hadStaffOf?in?.1 ?in?hadStaffOf?in?.2 (2))

$\forall x_1x_2x_3x_4.\,\texttt{?in?hadStaffOf?in?}\ x_1x_2x_3x_4 \rightarrow$
$\exists^{=2}y_1y_2.\,\texttt{?in?hadStaffOf?in?}\ y_1y_2x_3x_4$

External frequency constraint

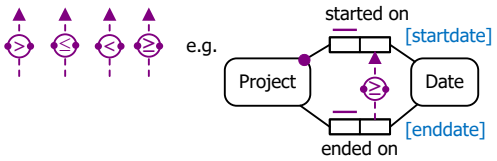


ExternalFrequency(isBy.2 isIn.2 (..2))

$\forall x_1x_2x_3.\,\texttt{JP}\ x_1x_2x_3 \leftrightarrow \exists y.\,\texttt{isBy}\ x_3x_1\ \&\ \texttt{isIn}\ x_3x_2$

$\forall x_1x_2x_3.\,\texttt{JP}\ x_1x_2x_3 \rightarrow \exists^{\leq 2}y_1y_2.\,\texttt{JP}\ y_1y_2x_3$

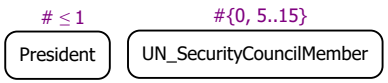
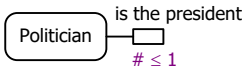
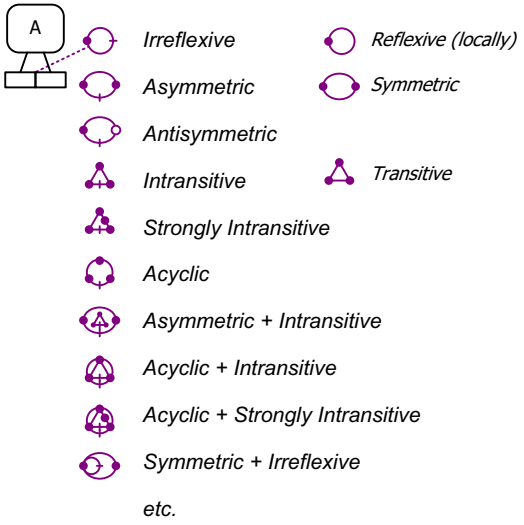
Value-comparison constraint



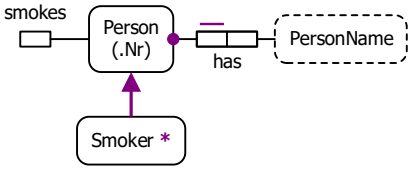
≥(endedOn.2 startedOn.2)

$\forall x_1x_2x_3.\,\texttt{JP}\ x_1x_2x_3 \leftrightarrow \exists y.\,\texttt{startedOn}\ x_3x_1\ \&\ \texttt{endedOn}\ x_3x_2$

$\forall x_1x_2x_3y_1y_2y_3.\,Px_1x_2x_3\ \&\ Py_1y_2y_3 \rightarrow \gamma_{\texttt{Date}}(x_2) \geq \gamma_{\texttt{Date}}(y_1)$

Construct and Examples	*Normative Abstract Syntax of Examples*	*Normative Semantics of Examples*
Object cardinality constraint 	**TypeCardinality**(President (0, 1))	$\exists^{\leq 1} x.\ \texttt{President}\ x$
Role cardinality constraint 	**RoleCardinality**(isThePresidentOf (0, 1))	$\exists^{\leq 1} x.\ \texttt{isThePresidentOf}\ x$
Ring constraints 	**LocallyReflexive**(P.1 P.2) *etc.*	$\forall x_1 x_2.\ \mathrm{P} x_1 x_2 \rightarrow \mathrm{P} x_1 x_1$ *etc.*

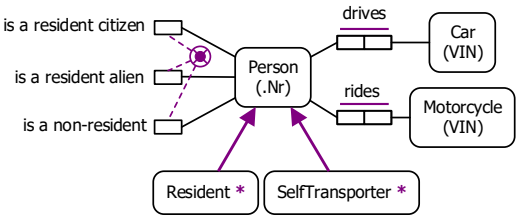
Derivation Rules



*** Each** Smoker **is a** Person **who** smokes.

```
SubTypeRule(Smoker (Person ∧ smokes))
```

$$\forall x.\ \mathtt{Smoker}\ x \leftrightarrow \mathtt{Person}\ x\ \&\ \mathtt{smokes}\ x$$



*** Each** Resident **is a** Person **who** is a resident citizen **or** is a resident alien.

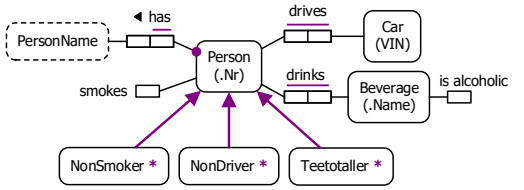
*** Each** SelfTransporter **is a** Person **who** drives **a** Car **or** rides **a** Motorcycle.

```
SubTypeRule(Resident
  (Person ∧ (isAResidentCitizen ∨
             isAResidentAlien))
```

$$\begin{aligned}&\forall x.\ \mathtt{Resident}\ x \leftrightarrow\\ &\quad \mathtt{Person}\ x\ \&\ (\mathtt{isAResidentCitizen}\ x \vee\\ &\qquad\qquad \mathtt{isAResidentAlien}\ x)\end{aligned}$$

```
SubTypeRule(SelfTransporter
  (Person ∧
    ((drives.1 ➤ [drives.2 ⋈ Car]) ∨
     (rides.1 ➤ [rides.2 ⋈ Motorcycle]))))
```

$$\begin{aligned}&\forall x.\ \mathtt{SelfTransporter}\ x \leftrightarrow\\ &\quad (\mathtt{Person}\ x\ \&\\ &\qquad ((\exists y.\ \mathtt{drives}\ xy\ \&\ \mathtt{Car}\ y) \vee\\ &\qquad\ (\exists y.\ \mathtt{rides}\ xy\ \&\ \mathtt{Motorcycle}\ y)))\end{aligned}$$



*** Each** NonSmoker **is a** Person **where it is not true that that** Person smokes.

*** Each** NonDriver **is a** Person **who** drives **no** Car.

*** Each** Teetotaller **is a** Person **who** drinks **no** Beverage **that** is alcoholic.

```
SubTypeRule(NonSmoker (Person \ smokes))
```

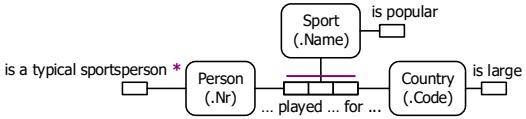
$$\forall x.\ \mathtt{NonSmoker}\ x \leftrightarrow \mathtt{Person}\ x\ \&\ \sim\mathtt{smokes}\ x$$

```
SubTypeRule(NonDriver
  (Person \ (drives.1 ➤ [drives.2 ⋈ Car]))
```

$$\begin{aligned}&\forall x.\ \mathtt{NonDriver}\ x \leftrightarrow\\ &\quad (\mathtt{Person}\ x\ \&\ \sim(\exists y.\ \mathtt{drives}\ xy\ \&\ \mathtt{Car}\ y)\end{aligned}$$

```
SubTypeRule(TeeTotaller
  (Person \
    (drinks.1 ➤ [drinks.2 ⋈
                 (Beverage ∧ isAlcoholic)]))
```

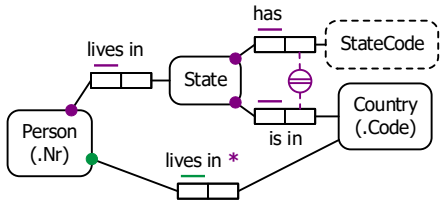
$$\begin{aligned}&\forall x.\ \mathtt{TeeTotaller}\ x \leftrightarrow\\ &\quad (\mathtt{Person}\ x\ \&\\ &\qquad \sim(\exists y.\ \mathtt{drinks}\ xy\ \&\\ &\qquad\quad \mathtt{Beverage}\ y\ \&\ \mathtt{isAlcoholic}\ y))\end{aligned}$$



*** Person is a typical sportsperson iff that** Person played **a** Sport **that** is popular for **a** Country **that** is large.

```
FactTypeRule(isATypicalSportsPerson
  (Person ∧ ?played?for?.1 ➤
    [?played?for?.2 ⋈ (Sport ∧ isPopular)]
    [?played?for?.3 ⋈ (Country ∧ isLarge)]))
```

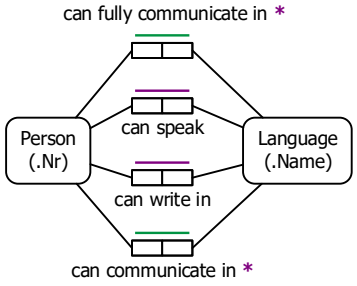
$$\begin{aligned}&\forall x.\mathtt{isATypicalSportsPerson}\ x \leftrightarrow\\ &\quad (\mathtt{Person}\ x\ \&\\ &\qquad \exists yz.\ \mathtt{?played?for?}\ xyz\ \&\ \mathtt{Sport}\ y\ \&\ \mathtt{isAlcoholic}\ y\ \&\\ &\qquad\qquad\qquad\qquad \mathtt{Country}\ z\ \&\ \mathtt{isLarge}\ z)\end{aligned}$$



* Person lives in Country **iff**
that Person lives in **a** State **that** is in **that** Country.

```
FactTypeRule(livesInCountry
  (Person ∧ livesInState.1 ➤
    [livesInState.2 ⋈ (State ∧ isIn.1 ➤
      [isIn.2 ⋈ (Country ∧ ?x)])])
  (Country ∧ ?x))
```

$$\forall xy.\ \texttt{livesInCountry}\ xy \leftrightarrow (\texttt{Person}\ x\ \&\ \exists z.\ \texttt{livesInState}\ xz\ \&\ \texttt{State}\ z\ \&\ \texttt{isIn}\ zy\ \&\ \texttt{Country}\ y)$$



* Person can fully communicate in Language **iff**
that Person can speak **that** Language
and can write in **that** Language.

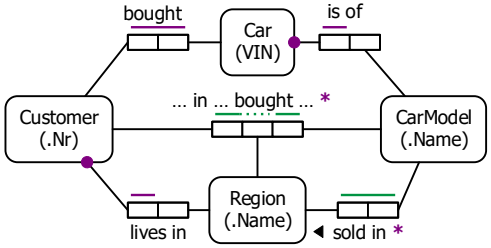
* Person can communicate in Language **iff**
that Person can speak **that** Language
or can write in **that** Language.

```
FactTypeRule(canFullyCommunicateIn
 (Person ∧
  (canSpeak.1 ➤ [canSpeak.2 ⋈ (Language ∧ ?x)]) ∧
  (canWrite.1 ➤ [canwrite.2 ⋈ (Language ∧ ?x)]))
 (Language ∧ ?x))
```

$$\forall xy.\ \texttt{canFullyCommunicateIn}\ xy \leftrightarrow (\texttt{Person}\ x\ \&\ \texttt{canSpeak}\ xy\ \&\ \texttt{canWrite}\ xy\ \&\ \texttt{Language}\ y)$$

```
FactTypeRule(canCommunicateIn
 (Person ∧
  ((canSpeak.1 ➤ [canSpeak.2 ⋈ (Language ∧ ?x)]) ∨
   (canWrite.1 ➤ [canwrite.2 ⋈ (Language ∧ ?x)])))
 (Language ∧ ?x))
```

$$\forall xy.\ \texttt{canFullyCommunicateIn}\ xy \leftrightarrow (\texttt{Person}\ x\ \&\ (\texttt{canSpeak}\ xy \vee \texttt{canWrite}\ xy)\ \&\ \texttt{Language}\ y)$$



* CarModel sold in Region **iff**
some Customer lives in **that** Region
and bought **a** Car **that** is of **that** CarModel.

* Customer in Region bought CarModel **iff**
that Customer lives in **that** Region
and bought **a** Car **that** is of **that** CarModel.

```
FactTypeRule(soldIn
 (CarModel ∧ ?x)
 (Region ∧
   (livesIn.2 ➤ [livesIn.1 ⋈ Customer ∧
     (bought.1 ➤ [bought.2 ⋈ Car ∧
       (isOf.1 ➤ [isOf.2 ⋈ (CarModel ∧ ?x)])])])))
```

$$\forall xy.\ \texttt{soldIn}\ xy \leftrightarrow (\texttt{CarModel}\ x\ \&\ \texttt{Region}\ y\ \&\ \exists z.\ \texttt{livesIn}\ zy\ \&\ \texttt{Customer}\ z\ \&\ \exists k.\ \texttt{bought}\ zk\ \&\ \texttt{Car}\ k\ \&\ \texttt{isOf}\ kx)$$

```
FactTypeRule(?in?bought?
 (Customer ∧
   (livesIn.1 ➤ [livesIn.2 ⋈ (Region ∧ ?x)]) ∧
   (bought.1 ➤ [bought.2 ⋈ (Car ∧
     (isOf.1 ➤ [isOf.2 ⋈ CarModel ∧ ?y])])))
 (Region ∧ ?x)
 (CarModel ∧ ?y))
```

$$\forall xy.\ \texttt{?in?bought?}\ xyz \leftrightarrow (\texttt{Customer}\ x\ \&\ \texttt{livesIn}\ xy\ \&\ \texttt{Region}\ y\ \&\ \exists k.\ \texttt{bought}\ xk\ \&\ \texttt{Car}\ k\ \&\ \texttt{isOf}\ kz\ \&\ \texttt{CarModel}\ z)$$