

ORM Normative Abstract Syntax and Semantics: non-normative glossary

ORM.net Proposed Recommendation

Version (draft): 29 February 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>>.

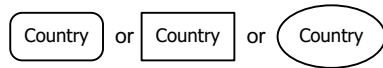
This document is part of the ORM document suite. It summarizes the abstract syntax of the main graphical symbols used in ORM by means of examples. The companion document “ORM Abstract Syntax and Semantics: normative specifications” formally defines the core ORM concepts. Both documents of the ORM document 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

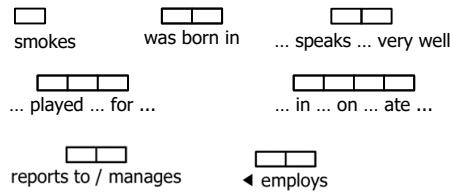
None.

Signature: Entity type nameSignature:

Entity Type name: Country

Signature: Value type nameSignature:

Value Type name: CountryCode

Signature: Predicate nameSignature:

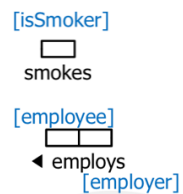
Unary predicate name: smokes

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:

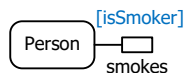
Role identifier for the unary predicate smokes:

smokes.1

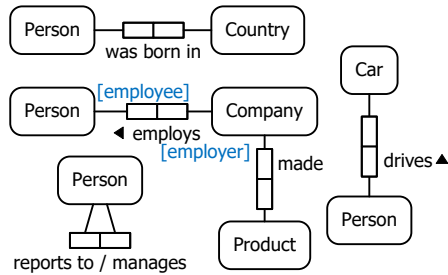
Role identifiers for the binary predicate employs:

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)) $\forall x. \text{smokes } x \rightarrow \text{Person } x$

Binary fact type



FactType(wasBornIn (Person Country))

FactType(employs (Company Person))

FactType(made (Company Product))

FactType(drives (Person Car))

FactType(reportsTo (Person Person))

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

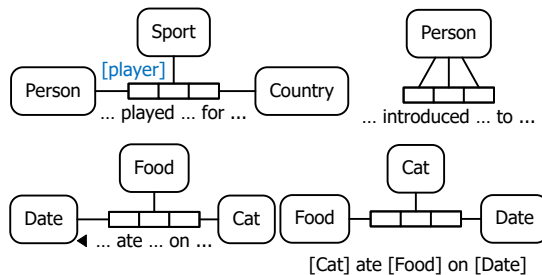
$\forall xy. \text{employs } x \ y \rightarrow \text{Company } x \ \& \ \text{Person } y$

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

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

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

Ternary fact type



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

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

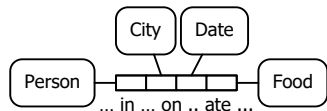
FactType(?ate?on? (Cat Food Date))

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

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

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

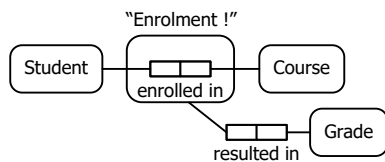
Quaternary fact type



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

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

Objectification



FactType(enrolledIn (Student Course))

Objectifies(Enrolment enrolledIn)

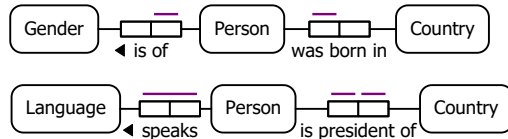
FactType(resultedIn (Enrolment Grade))

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

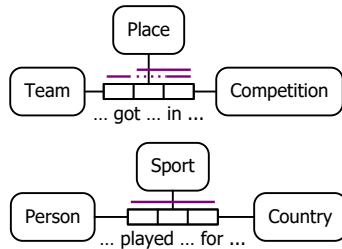
$\forall xy. \text{enrolledIn } x \ y \leftrightarrow \text{Enrolment } (\ell_{\text{enrolledIn}}(x \ y))$

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

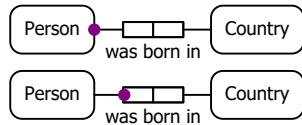
UCs on a binary fact type

**Unique**(isOf.1)**Unique**(wasBornIn.1)**Unique**(speaks.1 speaks.2)**Unique**(isPresidentOf.1)**Unique**(isPresidentOf.2) $\forall x_1 x_2. \text{isOf } x_1 x_2 \rightarrow \exists^1 y. \text{isOf } x_1 y$ $\forall x_1 x_2. \text{wasBornIn } x_1 x_2 \rightarrow \exists^1 y. \text{wasBornIn } x_1 y$ $\forall x_1 x_2. \text{speaks } x_1 x_2 \rightarrow \text{speaks } x_1 x_2$ $\forall x_1 x_2. \text{isPresidentOf } x_1 x_2 \rightarrow \exists^1 y. \text{isPresidentOf } x_1 y$ $\forall x_1 x_2. \text{isPresidentOf } x_1 x_2 \rightarrow \exists^1 y. \text{isPresidentOf } y x_2$

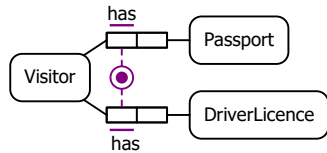
UCs on ternaries

**Unique**(?got?in?.1 ?got?in?.3)**Unique**(?got?in?.2 ?got?in?.3) $\forall x_1 x_2 x_3. ?\text{got?in? } x_1 x_2 x_3 \rightarrow \exists^1 y. ?\text{got?in? } x_1 y x_3$ $\forall x_1 x_2 x_3. ?\text{got?in? } x_1 x_2 x_3 \rightarrow \exists^1 y. ?\text{got?in? } y x_2 x_3$ **Unique**(?played?for?.1 ?played?for?.2
?played?for?.3) $\forall x_1 x_2 x_3. ?\text{played?for? } x_1 x_2 x_3 \rightarrow ?\text{played?for? } x_1 x_2 x_3$

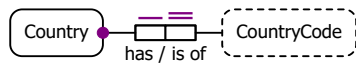
Simple mandatory role constraint

**Mandatory**(Person wasBornIn.1) $\forall x. \text{Person } x \rightarrow \exists y. \text{wasBornIn } xy$

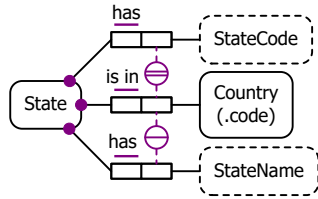
Inclusive-or constraint

**Mandatory**(Visitor
hasPassport.1 hasDriverLicence.1) $\forall x. \text{Visitor } x \rightarrow (\exists y. \text{hasPassport } xy) \vee (\exists y. \text{hasDriverLicence } xy)$

Preferred internal UC

**Identification**(Country has.1 (has.2)) $\forall x_1 x_2. \text{has } x_1 x_2 \rightarrow \exists^1 y. \text{has } x_1 y$ $\forall x. \text{Country } x \rightarrow \exists y. \text{has } xy$ $\forall x_1 x_2. \text{has } x_1 x_2 \rightarrow \exists^1 y. \text{has } y x_2$ $\text{well-founded}(\text{has})$

External UC



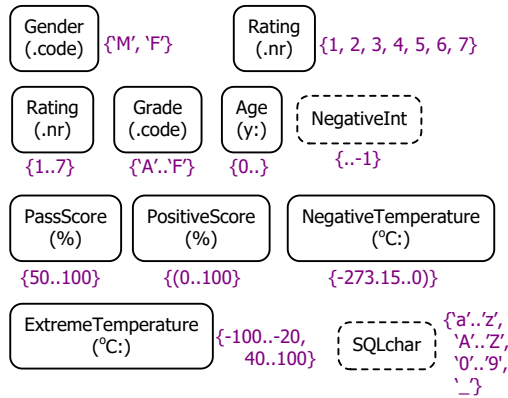
ExternalIdentification(State
(hasStateCode.2 isIn.2))

$$\begin{aligned} \forall x_1 x_2 x_3. \text{JP1 } x_1 x_2 x_3 &\leftrightarrow \exists y. \text{hasStateCode } x_3 x_1 \ \& \ \text{isIn } x_3 x_2 \\ \forall x_1 x_2 x_3. \text{JP1 } x_1 x_2 x_3 &\rightarrow \exists^{=1} y. \text{JP1 } x_1 x_2 y \\ \forall x_1 x_2 x_3. \text{JP1 } x_1 x_2 x_3 &\rightarrow \exists^{=1} y_1 y_2. \text{JP1 } y_1 y_2 x_3 \\ \forall x. \text{State } x &\rightarrow \exists y_1 y_2. \text{K1 } y_1 y_2 x \\ \text{well-founded}(\text{hasStateCode}) &\ \text{well-founded}(\text{isIn}) \end{aligned}$$

ExternalUnique(hasStateName.2 isIn.2)

$$\begin{aligned} \forall x_1 x_2 x_3. \text{JP2 } x_1 x_2 x_3 &\leftrightarrow \text{hasStateName } x_3 x_1 \ \& \ \text{isIn } x_3 x_2 \\ \forall x_1 x_2 x_3. \text{JP2 } x_1 x_2 x_3 &\rightarrow \exists^{=1} y. \text{JP2 } x_1 x_2 y \end{aligned}$$

Object type value constraint

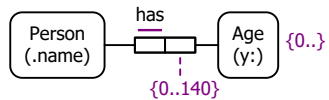


ValuesOf(GenderCode (M F))

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

...

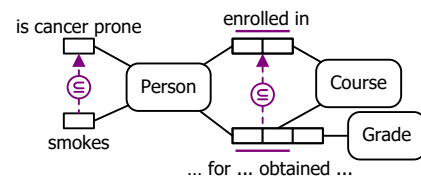
Role value constraint



ValuesOf(has.2 (0 ... 140))

$$\forall x_1 x_2. \text{has } x_1 x_2 \rightarrow x_2 = 0 \vee \dots \vee x_2 = 140$$

Subset constraint

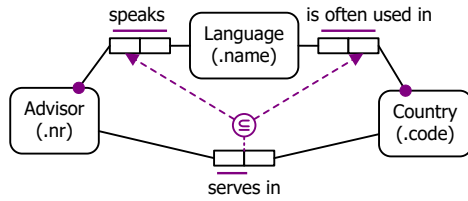


Subset((smokes.1 isCancerProne.1))

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

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

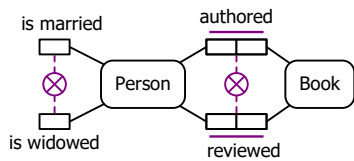
$$\forall x_1 x_2 x_3. ?\text{for?obtained?} x_1 x_2 x_3 \rightarrow \text{enrolledIn } y_1 x_2$$

Join subset constraint

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

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

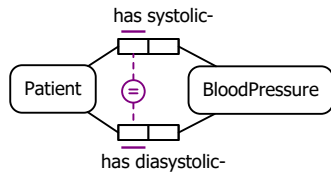
$$\forall x_1 x_2. P x_1 x_2 \leftrightarrow \exists y. \text{speaks } x_1 y \ \& \ \text{isOftenUsedIn } y x_2$$

$$\forall xy. \text{servesIn } xy \rightarrow Px y$$
Exclusion constraint

Exclusive((isWidowed.1 isMarried.1))

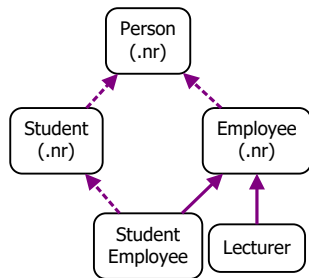
Exclusive((reviewed.1 authored.1)
(reviewed.2 authored.2))

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

$$\forall xy. \text{reviewed } xy \rightarrow \sim \text{authored } xy$$
Equality constraint

Equal((hasSystolic.1 hasDiasystolic.1))

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

$$(\forall xy. \text{hasDiasystolic } xy \rightarrow \exists z. \text{hasSystolic } xz)$$
Subtyping

Subtype(Lecturer Employee)

Subtype(Employee Person)

Subtype(Student Person)

Subtype(StudentEmployee Student)

Subtype(StudentEmployee Employee)

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

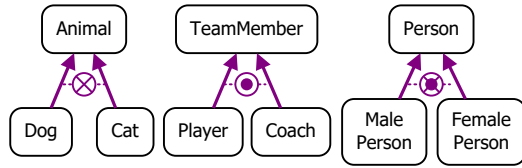
$$\forall x. \text{Employee } x \rightarrow \text{Person } x$$

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

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

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

Subtyping constraints



ExclusiveSubtypes((Dog Cat) Animal)

ExhaustiveSubtypes((Player Coach) TeamMember)

ExclusiveSubtypes((MalePerson FemalePerson) Person)

ExhaustiveSubtypes((MalePerson FemalePerson) Person)

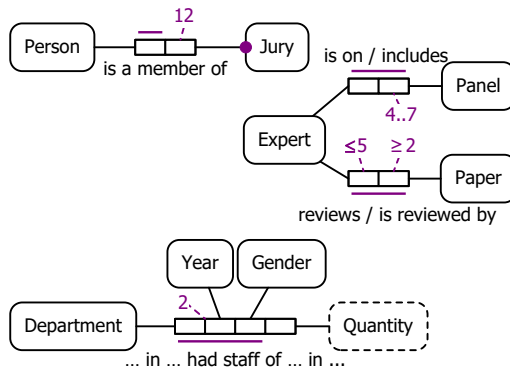
$(\forall x. \text{Dog } x \rightarrow \text{Animal } x \ \& \ \sim \text{Cat } x) \ \& \ (\forall x. \text{Cat } x \rightarrow \text{Animal } x)$

$(\forall x. \text{Player } x \rightarrow \text{TeamMember } x) \ \& \ (\forall x. \text{Coach } x \rightarrow \text{TeamMember } x) \ \& \ (\forall x. \text{TeamMember } x \rightarrow \text{Coach } x \vee \text{Player } x)$

$(\forall x. \text{MalePerson } x \rightarrow \text{Person } x \ \& \ \sim \text{FemalePerson } x) \ \& \ (\forall x. \text{FemalePerson } x \rightarrow \text{Person } x)$

$(\forall x. \text{MalePerson } x \rightarrow \text{Person } x) \ \& \ (\forall x. \text{FemalePerson } x \rightarrow \text{Person } x) \ \& \ (\forall x. \text{Person } x \rightarrow \text{FemalePerson } x \vee \text{MalePerson } x)$

Internal frequency constraint



Frequency(isAMemberOf.2 (12))

Frequency(isOn.2 (4, 7))

Frequency(reviews.1 (..5))

Frequency(reviews.2 (2..))

$\forall x_1 x_2. \text{isAMemberOf } x_1 x_2 \rightarrow \exists^{=12} y. \text{isAMemberOf } x_1 y$

$\forall x_1 x_2. \text{isOn } x_1 x_2 \rightarrow \exists^{\geq 4, \leq 7} y. \text{isOn } x_1 y$

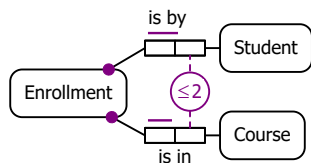
$\forall x_1 x_2. \text{reviews } x_1 x_2 \rightarrow \exists^{\leq 5} y. \text{reviews } y x_2$

$\forall x_1 x_2. \text{reviews } x_1 x_2 \rightarrow \exists^{\geq 2} y. \text{reviews } x_1 y$

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

$\forall x_1 x_2 x_3 x_4. ?in?hadStaffOf?in? x_1 x_2 x_3 x_4 \rightarrow \exists^{=2} y_1 y_2. ?in?hadStaffOf?in? y_1 y_2 x_3 x_4$

External frequency constraint

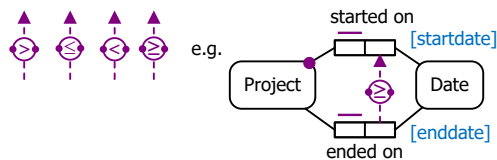


ExternalFrequency(isBy.2 isIn.2 (..2))

$\forall x_1 x_2 x_3. \text{JP } x_1 x_2 x_3 \leftrightarrow \exists y. \text{isBy } x_3 x_1 \ \& \ \text{isIn } x_3 x_2$

$\forall x_1 x_2 x_3. \text{JP } x_1 x_2 x_3 \rightarrow \exists^{\leq 2} y_1 y_2. \text{JP } y_1 y_2 x_3$

Value-comparison constraint



$\geq(\text{endedOn.2 startedOn.2})$

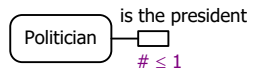
$\forall x_1 x_2 x_3. \text{JP } x_1 x_2 x_3 \leftrightarrow \exists y. \text{startedOn } x_3 x_1 \ \& \ \text{endedOn } x_3 x_2$

$\forall x_1 x_2 x_3 y_1 y_2 y_3. P x_1 x_2 x_3 \ \& \ P y_1 y_2 y_3 \rightarrow \gamma_{\text{Date}}(x_2) \geq \gamma_{\text{Date}}(y_1)$

Object cardinality constraint

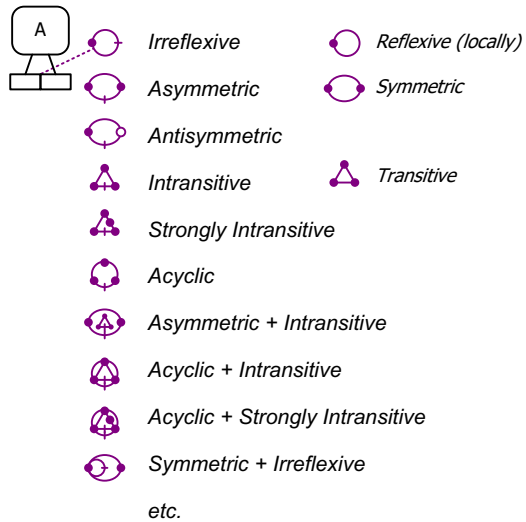
TypeCardinality(President (0, 1))

$\exists^{\leq 1} x. \text{President } x$

Role cardinality constraint

RoleCardinality(isThePresidentOf (0, 1))

$\exists^{\leq 1} x. \text{isThePresidentOf } x$

Ring constraints

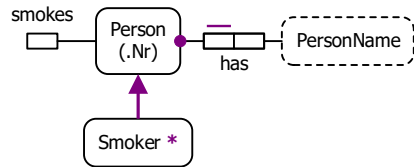
LocallyReflexive(P.1 P.2)

etc.

$\forall x_1 x_2. P x_1 x_2 \rightarrow P x_1 x_1$

etc.

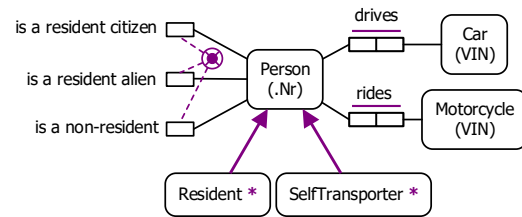
Derivation Rules



* Each Smoker is a Person who smokes.

SubTypeRule(Smoker (Person \wedge smokes))

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



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

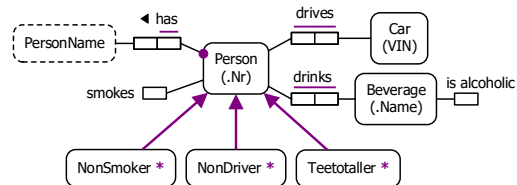
* Each SelfTransporter is a Person who drives a Car or rides a Motorcycle.

SubTypeRule(Resident
(Person \wedge (isAResidentCitizen \vee
isAResidentAlien)))

$\forall x. \text{Resident } x \leftrightarrow$
Person $x \ \& \ (\text{isAResidentCitizen } x \ \vee$
isAResidentAlien $x)$

SubTypeRule(SelfTransporter
(Person \wedge
((drives.1 \triangleright [drives.2 \bowtie Car]) \vee
(rides.1 \triangleright [rides.2 \bowtie Motorcycle]))))

$\forall x. \text{SelfTransporter } x \leftrightarrow$
(Person $x \ \&$
($(\exists y. \text{drives } xy \ \& \ \text{Car } y) \vee$
($\exists y. \text{rides } xy \ \& \ \text{Motorcycle } y)$)))



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

* Each NonDriver is a Person who drives no Car.

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

SubTypeRule(NonSmoker (Person \setminus smokes))

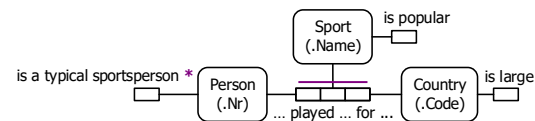
$\forall x. \text{NonSmoker } x \leftrightarrow \text{Person } x \ \& \ \sim \text{smokes } x$

SubTypeRule(NonDriver
(Person \setminus (drives.1 \triangleright [drives.2 \bowtie Car])))

$\forall x. \text{NonDriver } x \leftrightarrow$
(Person $x \ \& \ \sim(\exists y. \text{drives } xy \ \& \ \text{Car } y)$)

SubTypeRule(TeeTotaler
(Person \setminus
(drinks.1 \triangleright [drinks.2 \bowtie
(Beverage \wedge isAlcoholic)])))

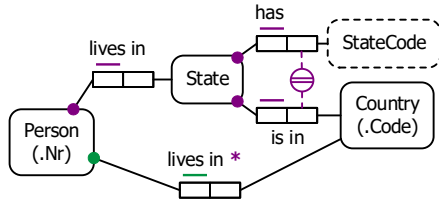
$\forall x. \text{TeeTotaler } x \leftrightarrow$
(Person $x \ \&$
 $\sim(\exists y. \text{drinks } xy \ \&$
Beverage $y \ \& \ \text{isAlcoholic } y)$)



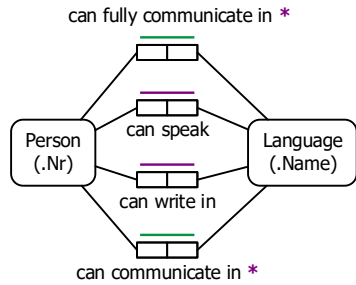
* Person is a typical sportsperson iff
that Person played a Sport that is popular for a Country that is large.

FactTypeRule(isATypicalSportsPerson
(Person \wedge ?played?for?.1 \triangleright
[?played?for?.2 \bowtie (Sport \wedge isPopular)]
[?played?for?.3 \bowtie (Country \wedge isLarge)]))

$\forall x. \text{isATypicalSportsPerson } x \leftrightarrow$
(Person $x \ \&$
 $\exists yz. \text{?played?for? } xyz \ \& \ \text{Sport } y \ \& \ \text{isAlcoholic } y \ \&$
Country $z \ \& \ \text{isLarge } z)$

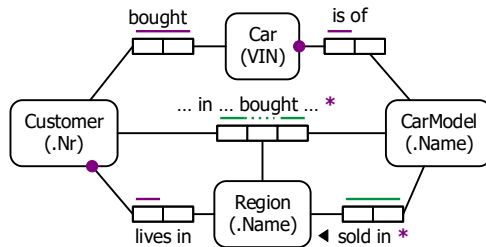


* Person lives in Country **iff**
that Person lives in a State that is in that Country.



* 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.



* 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(livesInCountry
(Person \wedge livesInState.1 \triangleright
[livesInState.2 \bowtie (State \wedge isIn.1 \triangleright
[isIn.2 \bowtie (Country \wedge ?x)])])
(Country \wedge ?x))

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

FactTypeRule(canFullyCommunicateIn
(Person \wedge
(canSpeak.1 \triangleright [canSpeak.2 \bowtie (Language \wedge ?x)]) \wedge
(canWrite.1 \triangleright [canwrite.2 \bowtie (Language \wedge ?x)]))
(Language \wedge ?x))

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

FactTypeRule(canCommunicateIn
(Person \wedge
((canSpeak.1 \triangleright [canSpeak.2 \bowtie (Language \wedge ?x)]) \vee
(canWrite.1 \triangleright [canwrite.2 \bowtie (Language \wedge ?x)]))
(Language \wedge ?x))

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

FactTypeRule(soldIn
(CarModel \wedge ?x)
(Region \wedge
(livesIn.2 \triangleright [livesIn.1 \bowtie Customer \wedge
(bought.1 \triangleright [bought.2 \bowtie Car \wedge
(isOf.1 \triangleright [isOf.2 \bowtie (CarModel \wedge ?x)])])])])

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

FactTypeRule(?in?bought?
(Customer \wedge
(livesIn.1 \triangleright [livesIn.2 \bowtie (Region \wedge ?x)]) \wedge
(bought.1 \triangleright [bought.2 \bowtie (Car \wedge
(isOf.1 \triangleright [isOf.2 \bowtie CarModel \wedge ?y)])])
(Region \wedge ?x)
(CarModel \wedge ?y))

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