Business Rule Modality

Terry Halpin

Neumont University

terry@neumont.edu
www.orm.net

Contents

- Introduction
- Modal Operators and Rule Verbalization
- Embedded, Static Rules
- Dynamic Rules
- Conclusion

Neumont University

Introduction

Business Rules determine what states and state transitions are possible or permitted for a given business domain, and may be of alethic or deontic modalities.

Alethic rules

- impose necessities (e.g. implied by physical laws)
- cannot be violated, even in principle

e.g. **Each** Person was born on **at most one** Date.
No Component includes **itself**.

Deontic rules

- impose obligations
- ought to be obeyed, but may be violated

e.g. **It is obligatory that each** Person is husband of **at most one** Person.
It is forbidden that any Person smokes in **any** Office.

Information modeling approaches such as ER, UML and ORM typically restrict their coverage of rules to alethic rules.

In practice, many business rules are deontic,
and it is often important to know when they are violated
so that actions can be taken to discourage future violations
(whether or not support for this is automated).

In recognition of this need as well as for rule exchange etc.,
the OMG is finalizing the SBVR[1]
(Semantics of Business Vocabulary and Rules) proposal
to specify a business semantics
layer on top of its software-oriented layers.

---

[1] Interim SBVR specification URL: www.omg.org/cgi-bin/doc?dtc/06-03-02.

This presentation is based on
the author's logical formalization work for SBVR,
and work on NORMA[1]
an open-source tool for ORM 2 (2nd generation ORM)
which supports deontic and alethic rules.

The ideas discussed could be adapted for other approaches
such as ER and UML.

In fact, the NORMA tool is currently being extended to
to support ER and UML notations as views of the underlying ORM model.

---

[1] http://sourceforge.net/projects/orm

Neumont University

Modal Operators and Rule Verbalization

Alethic		*Deontic*	
Reading	*Symbol*	*Reading*	*Symbol*
It is necessary that	□	It is obligatory that	***O***
It is possible that	◊	It is permitted that	***P***
It is impossible that	~◊	It is forbidden that	***F***

Modal negation rules

$\sim\Box p \equiv \Diamond\sim p$ $\qquad$ $\sim \mathbf{O}p \equiv \mathbf{P}\sim p$

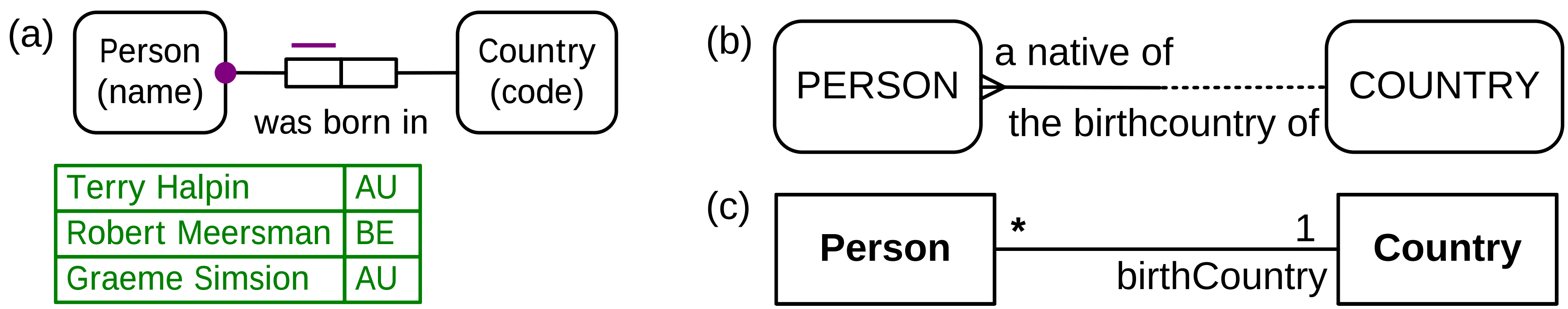
$\sim\Diamond p \equiv \Box\sim p$ $\qquad$ $\sim \mathbf{P}p \equiv \mathbf{O}\sim p \equiv \mathbf{F}p$

$\sim \mathbf{F}p \equiv \mathbf{P}p$

□ = true in all possible worlds ◊ = true in some possible world

For static constraints, a possible world is a state of the business domain.

NEUMONT UNIVERSITY



NORMA displays alethic constraints in violet
and verbalizes constraints in positive and negative forms.

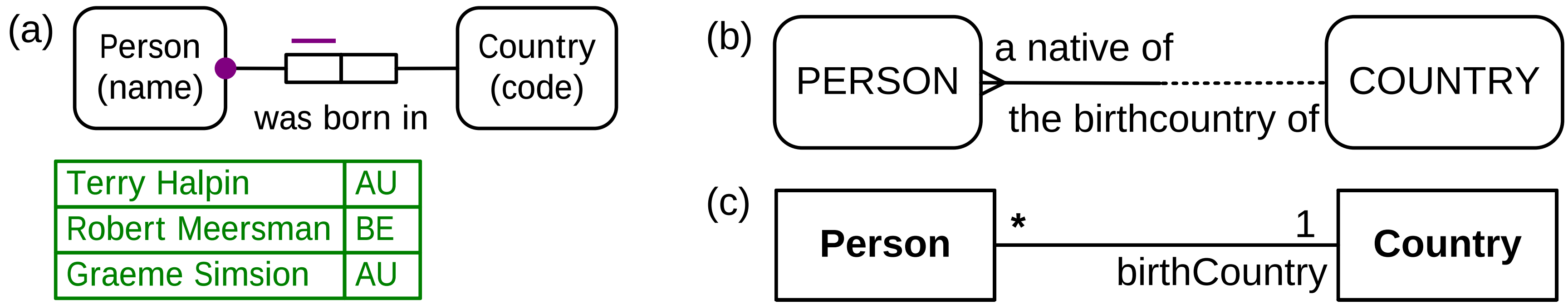
The mandatory and uniqueness constraints in positive, combined form:

Each Person was born in **exactly one** Country.

(2 constraints: **exactly one** = **some**, and **at most one**).

This is read as a necessity. As an option, this may be made explicit:

It is necessary that each Person was born in **exactly one** Country.

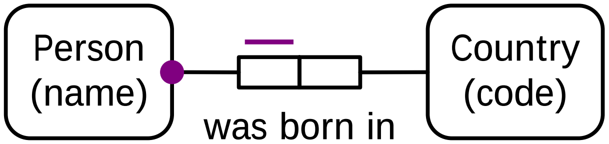


In positive form, NORMA also verbalizes
the lack of a uniqueness constraint on the right-hand role:

It is possible that
more than one Person was born in **the same** Country.

Satisfying populations help illustrate constraints.

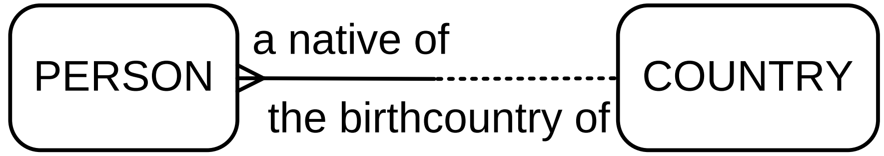
(a)



Terry Halpin	AU
Robert Meersman	BE
Graeme Simsion	AU

Terry Halpin	AU
Terry Halpin	US

(b)



(c)

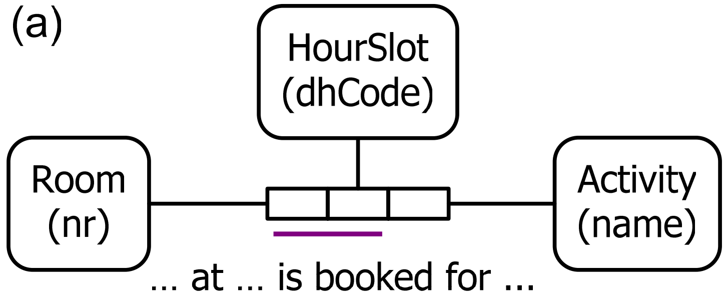
The mandatory and uniqueness constraints verbalized separately in negative form:

For each Person, **it is impossible that**
that Person was born in **more than one** Country.

It is impossible that any Person was born in **no** Country.

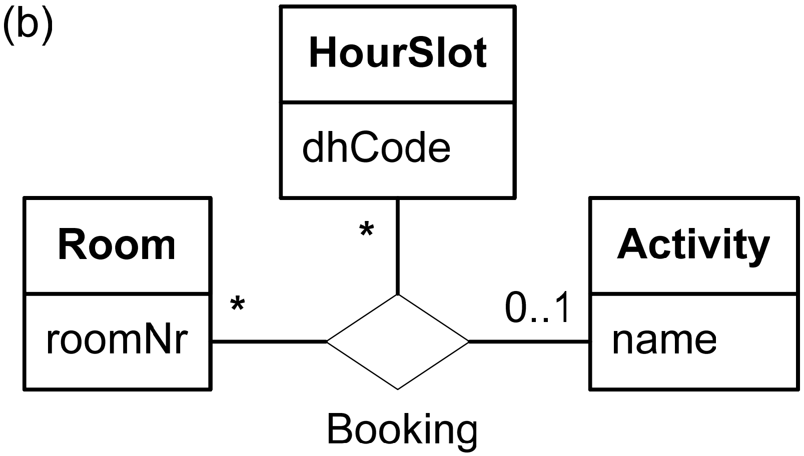
NORMA is being extended to support counterexamples to illustrate constraint violation and generate test cases.

(a)



20	Mon 9 am	ORM class
20	Mon 4 pm	CQ demo
20	Tue 2 pm	ORM class
33	Mon 9 am	CQ demo
33	Fri 5 pm	Party

(b)



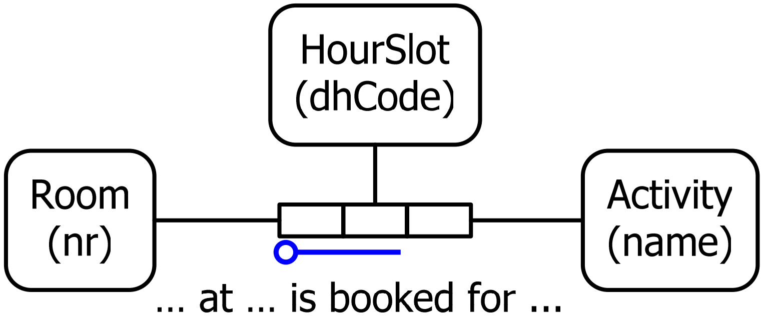
The uniqueness constraint might be alethic (shown above) or deontic.

Alethic +ve:

[It is necessary that] **For each** Room **and** HourSlot,
that Room at **that** HourSlot is booked for **at most one** Activity.

Alethic –ve:

It is impossible that
the same Room at **the same** HourSlot
is booked for **more than one** Activity.



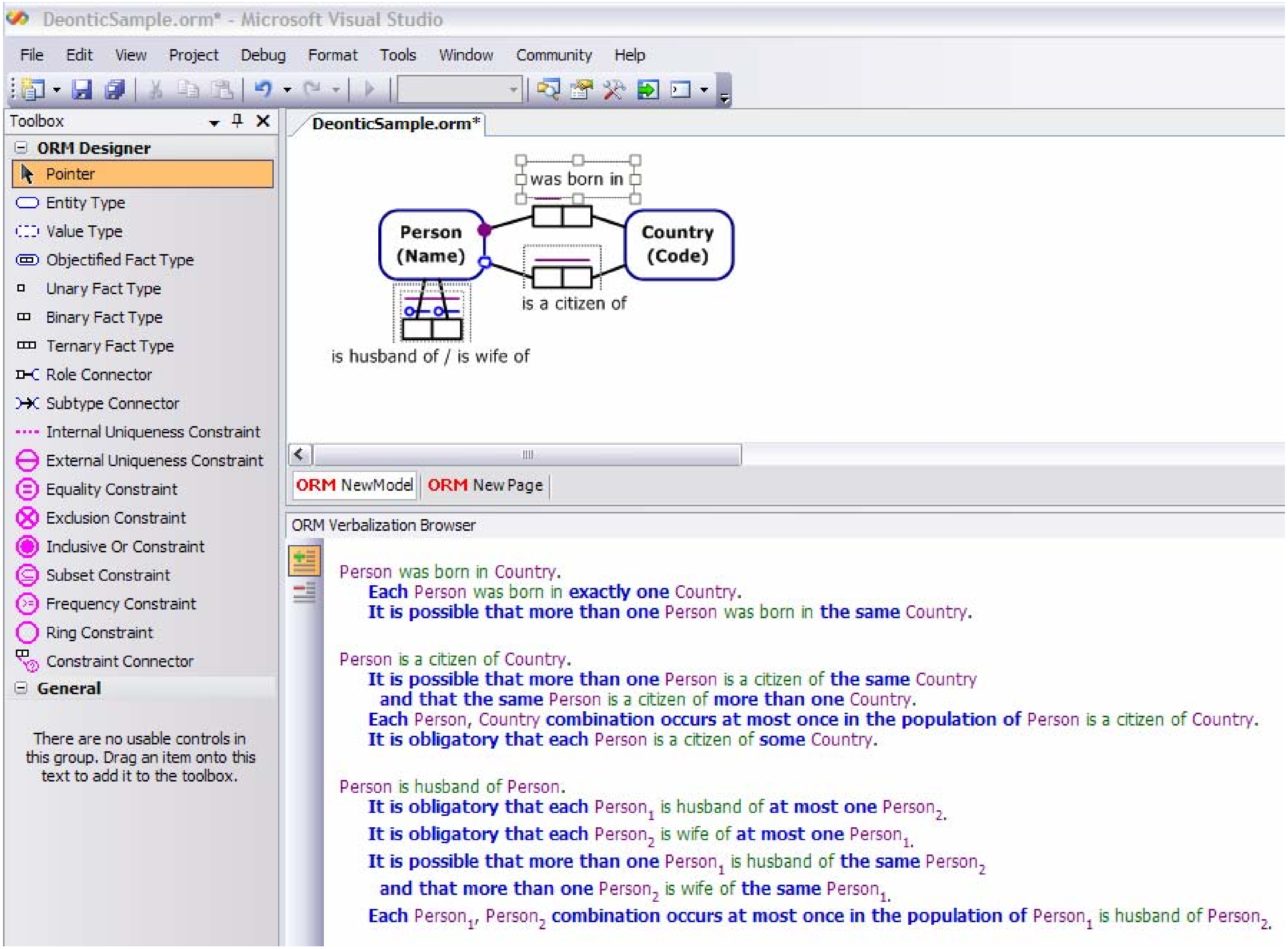
NORMA display deontic rules in blue,
typically with an “o” for “obligatory”.

Deontic modality:

+ve: **It is obligatory that for each** Room **and** HourSlot,
that Room at **that** HourSlot is booked for **at most one** Activity.

-ve: **It is forbidden that**
the same Room at **the same** HourSlot
is booked for **more than one** Activity.

NORMA screenshot



NORMA currently allows modal operators only as the main operator of the rule expression.

Some allowed SBVR formulations that violate this restriction may be transformed into an equivalent NORMA expression by applying modal negation rules and the Barcan formulae and their converses, i.e.

$$\forall x \, \Box F x \equiv \Box \forall x \, F x$$
$$\exists x \, \Diamond F x \equiv \Diamond \exists x \, F x$$

e.g.

For each Person,
it is necessary that that Person was born in **at most one** Country.

transforms to

It is necessary that each Person was born in **at most one** Country.

We also accept the following deontic variant of the Barcan formulae, allowing $\forall$ and ***O*** to commute.

$$\forall x\, \boldsymbol{O}\, \mathrm{F}x \equiv \boldsymbol{O}\, \forall x\, \mathrm{F}x$$

This allows the following rule

For each Person,
it is obligatory that that Person is a husband of **at most one** Person.

to be transformed to

It is obligatory that each Person is a husband of **at most one** Person**.**

Neumont University

By normalizing rules to restrict modal operators to the main operator, the only impact on tagging a rule as a necessity or obligation is on rule enforcement.

For static rules, no commitment to a specific modal logic is required.

Enforcement of a necessity rule never allows violations.

Enforcement of an obligation rule allows violations,
but takes some remedial action
(e.g. display a message to a relevant authority
indicating violation of the rule
including verbalization of the rule).

Embedded, Static Rules

SBVR allows modal operators to be embedded anywhere in rule expressions. In rare cases, these rules cannot be transformed into rules where the only modal operator is the main operator.

To support such cases, we have two alternatives

- adopt a specific modal logic[1]
- replace the embedded modal operators by domain-level predicates

For the first alternative, we prefer S4[2].
For the second alternative, we proceed as for embedded deontics (see later).

[1] There are many normal modal logics (e.g. K, K4, KB, K5, DT, DB, D4, D5, T, Br, S4, S5) as well as non-normal modal logics (e.g. C2, ED2, E2, S0.5, S2, S3).

[2] Schema evolution may seem to violate S4 (its accessibility relationship between possible worlds in transitive), but we resolve this by treating such evolution at the metametalevel.

Formally, we treat a model as an interpretation
where each *non-deontic* formula evaluates to true.

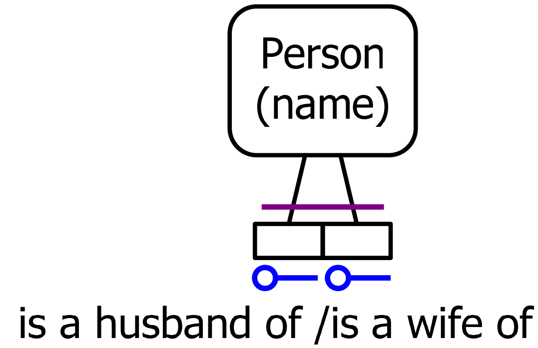
A model is a permitted model
if the *p* in each deontic formula of the form ***O**p* evaluates to true
otherwise it is a forbidden model (though still a model).

This approach avoids the need to assign a truth value
to expressions of the form ***O**p*.

Various metarules apply between alethic and deontic rules, e.g.

the argument (role set) of a deontic uniqueness constraint must be a proper subset of the argument of an alethic uniqueness constraint (explicit or implicit).

This example satisfies this metarule:



However, if the marriage predicate were alethically 1:1,
then no deontic uniqueness constraint may be added
(if something is already necessary,
it makes no sense to declare it obligatory).

Some SBVR formulae (e.g. $\boldsymbol{OP}p$) are illegal in some deontic logics, and deontic logic itself is rife with controversies.

If a deontic model operator is embedded, we first try to normalize the formula using transformation rules such as $p \supset \boldsymbol{O}q \;.\!\equiv\!.\; \boldsymbol{O}(p \supset q)$[1] or deontic counterparts to the Barcan formulae.

In rare cases, embedded deontics cannot be so normalized. Rather than choosing a specific modal logic (a risky option), where possible we transform such cases into first-order formulae with no modalities by *replacing the modal operators by predicates at the business domain level.*

Such predicates (e.g. “is forbidden”, “is permitted”) are reserved and given additional semantics
(e.g. exclusion constraints between forbidden/permitted predicates).

[1] In contrast to our approach, some versions of deontic logic reject this equivalence on the basis that agent control is restricted to the scope of modal operators.

As a simple example, consider the following SBVR rule.

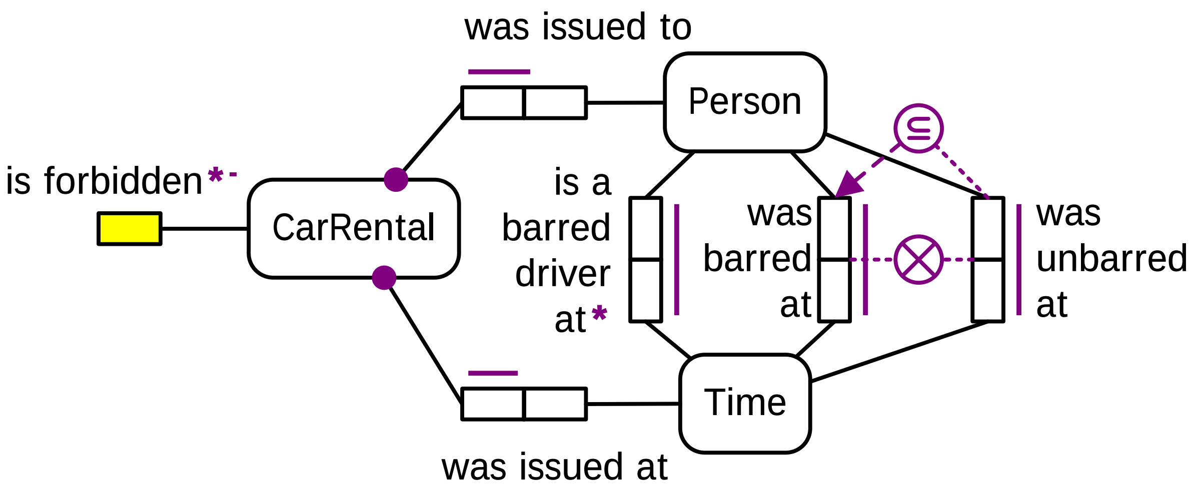
Car rentals ought not be issued to people
who are barred drivers at the time the rental was issued.

This deontic rule may be captured by the following ORM derivation rule for the partly derived domain fact type CarRental is forbidden:

CarRental is forbidden **if**
CarRental was issued at Time **and**
CarRental was issued to Person **and**
Person is a barred driver at Time.

The following slide shows the full ORM diagram for this case.

Neumont University



* Person is a barred driver at $Time_1$ **iff**
Person was barred at **a** $Time_2$ <= $Time_1$ **and**
Person was **not** unbarred at **a** $Time_3$ **between** $Time_2$ **and** $Time_1$

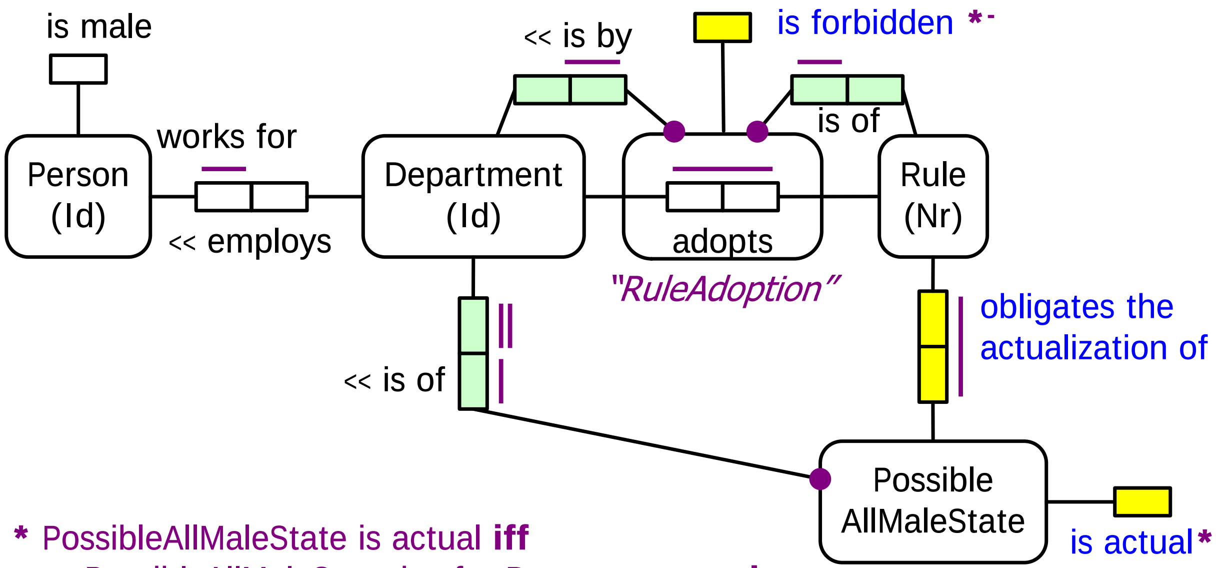
*- CarRental is forbidden **if**
CarRental was issued at Time **and**
CarRental was issued to Person **and**
Person is a barred driver at Time

SBVR allows rules that embed possibly non-factual propositions. However, there does not appear to be any simple solution to providing explicit, formal semantics for such rules.

As a nasty example, consider the following business rule:

> It is not *permitted* that some department adopts a rule
> that *says* it is *obligatory* that
> each employee of that department is male.

This example includes the mention (rather than use) of an open proposition in the scope of an embedded deontic operator.
One possible, though weak, solution is to rely on
reserved domain predicates to carry much of the semantics implicitly,
as shown in the following ORM schema.



* PossibleAllMaleState is actual **iff**
PossibleAllMaleState is of **a** Department **and**
each Person **who** works for **that** Department is male

*- RuleAdoption is forbidden **if**
RuleAdoption is by **a** Department
and is of **a** Rule
that obligates the actualization of **a** PossibleAllMaleState
that is of **the same** Department

Some support for reification in the sense of propositional nominalization is being added by Pat Hayes and Chris Menzel to the IKL language, while retaining first-order. When available, this may offer other options for dealing with such cases.

Dynamic Rules

Dynamic rules restrict the possible transitions between business states.

The rule may simply compare one state to the next
(e.g. Salaries should never decrease)

or the rule may compare states separated by a given period
(e.g. Invoices ought to be paid within 30 days of being issued).

The invoice rule might be formally expressed thus,
assuming the relevant fact types exist in the conceptual schema:

For each Invoice, **if that** Invoice was issued on $Date_1$
then it is obligatory that
that Invoice is paid on $Date_2$
where $Date_2 <= Date_1 + 30$ days.

This rule might now be normalized to the following formulation, moving the deontic operator to the front:

It is obligatory that
each Invoice **that** was issued on $Date_1$
is paid on $Date_2$
where $Date_2$ <= $Date_1$ + 30 days.

This transformation requires an equivalence rule such as

$p \supset \boldsymbol{O}q \;.\!\equiv\!.\; \boldsymbol{O}(p \supset q)$.

While this formula is actually illegal in some deontic logics, it does seem intuitively acceptable.

In principle, such formal transformation issues might be ignored, so long as the domain expert confirms that the normalized formulation agrees with his/her intended semantics.

While it is obvious how to implement this rule in a database system, capturing the formal semantics in an appropriate logic
(e.g. a temporal or dynamic logic) is a harder task.

One possibility is to provide a temporal package that may be imported into a domain model, in order to provide a first-order logic solution.

Another possibility is to adopt a temporal modal logic
(e.g. treat a possible world as a sequence of accessible states).

We prefer a first-order solution where possible.

Conclusion

Many business rules are deontic rather than alethic.

This presentation discussed one way of
modeling and verbalizing such rules in ORM 2,
as supported by the open-source NORMA tool.

NORMA currently supports static rules where
the only modal operator is the main operator.

Rules that cannot be transformed into such cases
are challenging to adequately formalize and support,
especially those involving embedded deontics or dynamic rules.

Some approaches were suggested to deal with such cases,
but further research is needed to adequately address these complexities.