EXAMPLE 8

```
1: *thief(x) ⇒ *punishable(x)
3: *minor(x) ⇒ *~punishable(x)
─────────────────────────────────
c = {L; 9: *minor(x) ⇒
  *exception(*thief(x) ⇒ punishable(x))}
```

Rule 9 holds that if somebody is a minor, the rule that thieves are punishable does not apply to him.

The rules 1 and 3 by themselves are logically inconsistent. Inclusion of rule 9 in the background makes that if the conditions of rule 3 are satisfied, there is an exception to rule 1, which takes the rule conflict away. As a consequence, the rules 1 and 3 are consistent relative to a background that contains rule 9.

Exceptions can also make a consistent set of rules inconsistent[20]:

EXAMPLE 9

```
1: *thief(x) ⇒ *punishable(x)
3: *minor(x) ⇒ *~punishable(x)
c = {9: *minor(x) ⇒
       *exception(thief(x) ⇒ *punishable(x));
10: *second_offender(x) ⇒ *exception(rule-9)}
```

We have seen in example 8 that the rules 1 and 3 are consistent with respect to a set of constraints that includes rule 9. The addition of rule 10 to the background makes that there is no guarantee anymore that, in case of a minor, there is an exception to rule 1. This is illustrated by the case that John is not only a thief and a minor, but also a second offender. In that case there is an exception to rule 9, and presumably no exception to rule 1. This illustrates that if rule 10 is added to the background, there are possible cases in which the conditions of both the rules 1 and 3 are satisfied, and in which these two rules are in actual conflict.

7. MODEL THEORY FOR RULES

Model-theoretic semantics for logic specifies the meanings of logical operators by means of the truth conditions of sentences in which these

[20] In the formalization of rule 10, rule 9 is referred to by 'rule-9'.

operators occur. Since rules are assumed[21] to have no truth values, and therefore also no truth conditions, this kind of semantics does not work for rules. It can, however, be adapted to rules by specifying what must be true if a particular rule exists, or is valid. Because the formalism that I develop for this purpose is strongly inspired by the usual model-theoretic semantics for predicate logic, I call it a model theory for rules.

Model-theoretic semantics traditionally focuses on the truth values of sentences. I will present the model theory in such a way that the emphasis is on the states of affairs that obtain. For instance, a world only counts as logically possible if it satisfies the constraint that if a state of affairs of the form `*a & b` obtains, the states of affairs of the forms `*a` and `*b` must also obtain, and vice versa. There is a close connection between this approach and traditional model-theoretic semantics, because a sentence that expresses a state of affairs is true if and only if this state of affairs obtains. I prefer the emphasis on states of affairs, because the effects of rules are in the first place that states of affairs obtain and only in the second place that particular sentences are true.

A second difference in presentation, in line with the first, is that I do not take the notion of a possible world for granted. Traditionally model-theoretic semantics specifies which relations exist between the truth values of sentences in a possible world. I turn this around and specify which relations between states of affairs must hold for a world to be a possible one. In this way the function of rules as constraints on possible worlds is highlighted. This makes it also easier to distinguish different notions of possibility as defined by different sets of constraints that are taken into account. We have seen in section 4 how variants of rule consistency depend on various notions of possibility and compatibility. These distinctions can be treated more naturally in a theory that focuses on possibility rather than on relations between truth values.

Central in the model theory for rules is the notion of a constraint. Logical constraints hold in general for all *logically possible worlds*. These are, in the present context, the constraints of predicate logic, augmented with one additional constraint that characterizes the logic of rules. Together, these constraints on all logically possible worlds are called the constraints of *Rule Logic*.

I will present the model-theoretic characterization of Rule Logic in two stages. First I disregard that exceptions should be minimized. This leads to a

[21] In the present paper I will not argue for this assumption, except for pointing out that the lack of a truth value immediately follows from the treatment of rules as logical individuals.

relatively simple characterization that has the drawback that exceptions to rules are possible, even if there are no reasons for their presence. Then, in section 11, I will formulate an additional constraint that takes the minimization of exceptions into account.

8. CONSTRAINTS

The notion of rule consistency will be formalized by model-theoretic means, namely in terms of possible worlds. Intuitively, a set of rules is inconsistent if there is a possible world in which the conditions of all the rules are satisfied and in which there is no exception to either one of the rules, while there is no possible world in which the conclusions of all the rules obtain. Variations on the notion of consistency are realized by different characterizations of possible worlds.

Some kinds of states of affairs tend to go together, while other ones exclude each other. For instance, the states of affairs that *x* kisses *y* tends to go together with the state of affairs that *x* touches *y*[22], and the state of affairs that *x* is a circle tends (very strongly) to exclude the state of affairs that *x* is a square. These relations between (usually generic) states of affairs are called *constraints* on possible worlds.

Rules, including legal rules, are a special kind of constraints. The rule that thieves are punishable makes that the state of affairs that somebody is a thief goes together with the state of affairs that this person is punishable. The unconditional rule that it is forbidden to steal, makes that every state of affairs goes together with the state of affairs that it is forbidden to steal. The power-conferring rule that the government and the parliament together are competent to make laws, makes that the states of affairs that these bodies are the government and parliament go together with the state of affairs that these bodies are competent to make laws.

The symbol $\Rightarrow$ is used to denote constraints in general and rules in particular. The predicate `Valid` serves to express that a rule exists, or – what boils down to the same thing – is valid. It is defined by the following sentence:

$$\texttt{Valid}(\mathit{rule}) \equiv_{\texttt{def.}} \exists x(x = \mathit{rule})$$

[22] This example stems from Barwise and Perry 1983, 12.

Finally, there is an one-place predicate `Exception` that ranges over instantiated rules and expresses that there is an exception to the rule in question for the case to which the rule is instantiated.[23] For instance:

`Exception(*thief(john) ⇒ *punishable(john))`

The first model-theoretic characterization of worlds that are possible according to Rule Logic runs as follows:

CONSTRAINTS ON WORLDS THAT ARE LOGICALLY POSSIBLE ACCORDING TO L_{RL}

Let L_{RL} be the language of Rule Logic. L_{RL} = {`S1, S2, ..., Sn`}, where `S1 ... Sn` are all the well-formed closed sentences of L_{RL}.

Let `Si` be a sentence in L_{RL}, and let `*sa`$_i$ denote the state of affairs that is expressed by `S`$_i$. `*sa`$_i$ is then a state of affairs that is *possible relative to* L_{RL}.[24]

Let the set SA be the set of all states of affairs that are possible relative to L_{RL}, and let W be the power set (the set of all subsets) of SA. Intuitively, W stands for the set of all worlds, the content of which is expressible in L_{RL}. Every $w \in W$ is a subset of SA.

There are no other constraints on the states of affairs that are elements of the worlds in W. There are, for instance, worlds in W in which the state of affairs `*p & q` obtains, but in which the state of affairs `*q` does not obtain. Such worlds are possible relative to L_{RL}, but they are not logically possible according to Rule Logic.[25] Worlds that are logically possible are subject to a number of additional constraints. The set of these logically possible worlds is denoted by W_{RL}.

23 More about this predicate and its relations to the predicates `Applicable` and `Applies` in chapter 3, section 5.4.

24 L_{RL} may be thought of as the conceptual scheme by means of which worlds are 'captured'.

25 One may argue for the position that only logically possible worlds are really possible, thereby excluding possible worlds in which both the states of affairs *a & b and *~a obtain. My reason for taking the worlds that are expressible by means of some language as basic is that I want to emphasize that logical constraints do not take a special position, but are ‘ordinary’ constraints, just like the physical ones, the mathematical ones, and the legal ones.

CONSTRAINTS ON WORLDS THAT ARE LOGICALLY POSSIBLE ACCORDING TO RULE LOGIC

1. if *p ∈ w then *~p ∉ w, if *~p ∈ w, then *p ∉ w, if *p ∉ w, then *~p ∈ w, and if *~p ∉ w, then *p ∈ w.
2. *p & q ∈ w if and only if both *p ∈ w and *q ∈ w.
3. *p ∨ q ∈ w if and only if either *p ∈ w, or *q ∈ w, or both.
4. *p → q ∈ w if and only if either *p ∉ w, or *q ∈ w, or both.
5. *p ≡ q ∈ w if and only if either both *p ∈ w and *q ∈ w, or both *p ∉ w and *q ∉ w.

These constraints correspond to the traditional constraints of propositional logic stated in terms of relations between states of affairs.

6. *∃x(r(x)) ∈ w if and only if there is an individual a in w, such that *r(a) ∈ w.
7. *∀x(r(x)) ∈ w if and only if there is no individual a in w, such that *r(a) ∉ w.

These constraints give the traditional meaning of the quantifiers, again stated in terms of states of affairs.[26]

A constraint that is characteristic for Rule Logic is that if the conditions of an existing rule are satisfied and there is no exception to this rule, the conclusion of this rule obtains. Let *conditions/σ and *conclusion/σ denote the states of affairs expressed by respectively the conditions and the conclusion of a rule with their variables instantiated according to instantiation σ. Then the above mentioned constraint becomes:

8. If*Valid(*conditions ⇒ *conclusion) ∈ w, and
   *conditions/σ ∈ w, and
   *exception(*conditions/σ ⇒ conclusion/σ)
   ∉ w, then
   *conclusion/σ ∈ w.

Finally there is a constraint to guarantee that terms that denote states of affairs expressed by logically equivalent sentences are co-referential:

[26] To gain simplicity at the cost of precision, the formulations of the constraints 6 and 7 do not deal with compound formulas, or the use of quantifiers or function expressions within the scope of the quantifiers.

9. If and only if for all worlds $w \in W_{RL}$ it holds that `*p ≡ q` ∈ w, then `*p = *q`.

The constraints of Rule Logic are abbreviated as c(RL).

9. COMPATIBILITY OF STATES OF AFFAIRS

Given the model theory for Rule Logic, it is possible to give a formal characterization of rule consistency. The starting point is the characterization of compatible states of affairs:

RELATIVE COMPATIBILITY OF STATES OF AFFAIRS

Let c be a set of constraints, and let W_c be the set of worlds $w \in W_{RL}$, such that for every constraint $c_i \in c \cup c(RL)$, it holds that `*Valid(ci)` ∈ w. The states of affairs in a set s are then said to be compatible relative to the set of constraints c if and only if there is some set of states of affairs $s' \in W_c$ such that $s \subseteq s'$.

LOGICAL COMPATIBILITY OF STATES OF AFFAIRS

The states of affairs in a set s are logically compatible if and only if they are compatible relative to the set of constraints c(RL): $s \subseteq s'$, where $s' \in W_c^{(RL)}$.

Let me illustrate this by means of the following examples:

EXAMPLE 10

`s = {*punishable(john), *~punishable(john)}`

The states of affairs in this set are logically incompatible, because of the first constraint on logically possible worlds.

EXAMPLE 11

`s = {*thief(john), *~punishable(john)}`

The states of affairs in this set are logically compatible, because there is no constraint on logically possible worlds that prevents the co-occurrence of these states of affairs.

We have seen that it is also possible to define a notion of compatibility that treats constraints, including rules, as a kind of background relative to which compatibilities are judged. If such a background contains the rule that thieves are punishable, the states of affairs that somebody is a thief and that he is not punishable, are incompatible relative to this background. The compatibility can then be restored by adding the presence of an exception to the rule that thieves are punishable to the set of states of affairs that is evaluated with regard to its compatibility, or to the background of constraints. This is illustrated by the following two examples:

EXAMPLE 12

```
s = {*thief(john), *~punishable(john)}
c = {*thief(x) ⇒ *punishable(x)}
```

The states of affairs in s are logically compatible, but incompatible relative to c because, barring exceptions, there is no world possible relative to c, in which a thief is not punishable.

If the set s is to be compatible relative to c, c should contain an exception to the rule that thieves are punishable. The following set s is compatible relative to c, because c contains the necessary exception:

EXAMPLE 13

```
s = {*thief(john), *~punishable(john), *minor(john)}
c = {*thief(x) ⇒ *punishable(x),
    *minor(x) ⇒ *exception(thief(x)⇒ punishable(x))}
```

10. THE CONSISTENCY OF RULES

By means of the notions of logical compatibility of states of affairs and compatibility of states of affairs relative to a set of constraints, it is possible to give a formal characterization of rule consistency.

Let r = $\{r_1 \ldots r_n\}$ be a finite set of n rules,
where r_i = *$conditions_i$ ⇒ *$conclusion_i$, for i = 1 to n.

Let s = $\{\sigma_1 \ldots \sigma_n\}$ be a set of n instantiations for the variables that occur in r, where σ_i is applied to the variables in r_i. For instance, let r_3 be *thief(x) ⇒ *punishable(x), and let σ_3 be {x -> john}. Then the instantiation of r3 by means of σ3, r_3/σ_3), is

```
*thief(john) ⇒ *punishable(john)
```

Let $I_{conditions}$(r, σ) be the set of the instantiations by means of σ of the conditions of all rules in r. That is:

$I_{conditions}$(r, σ) =
 {*$conditions_1/\sigma_1$), ... $conditions_n/\sigma_n$}.

Let $\mathtt{I}_{\mathtt{conclusion}}(\mathtt{r},\ \sigma)$ be
$\{$*conclusion$_1/\sigma_1$), ... *conclusion$_n/\sigma_n\}$.

Let $\mathtt{I}_{\mathtt{\sim exception}}(\mathtt{r},\ \sigma)$ be
$\{$*~exception(conditions$_1/\sigma_1$) $\Rightarrow$ conclusion$_1/\sigma_1$) ...
*~exception(conditions$_n/\sigma_n$) $\Rightarrow$ conclusion$_n/\sigma_n$)$\}$.

Then the following definition captures the notion of rule consistency relative to a set of constraints:

RELATIVIZED RULE CONSISTENCY:

The rules in the set s are consistent relative to a set of constraints c, if and only if it is not so that there is a set s' $\subseteq$ s and a set of instantiations σ, such that

a. the states of affairs in the set $\mathtt{I}_{\mathtt{conditions}}(\mathtt{s'},\ \sigma) \cup \mathtt{I}_{\mathtt{\sim exception}}(\mathtt{r},\ \sigma)$ are compatible relative to $\mathtt{c} \cup \mathtt{s} \cup \mathtt{c(RL)}$,
b. the states of affairs in the set $\mathtt{I}_{\mathtt{conclusion}}(\mathtt{r},\ \sigma)$ are incompatible relative to $\mathtt{c} \cup \mathtt{s} \cup \mathtt{c(RL)}$.

The compatibility of the joint rule conditions and conclusions and the absence of exceptions to the rules is judged against the background of both the set of constraints and the rules themselves, because the rules that are evaluated with regard to their consistency also are constraints on the world in which they exist.

LOGICAL CONSISTENCY OF RULES

The rules in the set s are logically consistent, if and only if it is not so that there is a set s' $\subseteq$ s and a set of instantiations σ, such that

a. the set $\mathtt{I}_{\mathtt{conditions}}(\mathtt{s'},\ \sigma) \cup \mathtt{I}_{\mathtt{exception}}(\mathtt{r},\ \sigma)$ is compatible relative to $\mathtt{s} \cup \mathtt{c(RL)}$,
b. the set $\mathtt{I}_{\mathtt{conclusion}}(\mathtt{r},\ \sigma)$ is incompatible relative to $\mathtt{s} \cup \mathtt{c(RL)}$.

Let me re-use some examples of the sections 3 and 4 to illustrate these definitions:

EXAMPLE 1

```
1: *thief(x) ⇒ *punishable(x)
2: *thief(x) ⇒ *~punishable(x)
```

The rules 1 and 2 are logically inconsistent, as is illustrated by the set of instantiations $\{\sigma1,\ \sigma2\}$, where $\sigma1 = \sigma2 = \{x \rightarrow \mathtt{john}\}$.

EXAMPLE 3

```
1: *thief(x) ⇒ *punishable(x)
4: *minor(x) ⇒ *protected(x)
5: *protected(x) ⇒ *~punishable(x)
```

That the rules 1, 4 and 5 are logically inconsistent is illustrated by the set of instantiations {σ1, σ4, σ5}, where
σ1 = σ4 = σ5 = {x → john}.

EXAMPLE 4

```
1: *thief(x) ⇒ *punishable(x)
4: *minor(x) ⇒ *protected(x)
-----------------------------------------
c = {5: *protected(x) ⇒ *~punishable(x)}
```

The rules 1 and 4 are also inconsistent relative to c because there can be no instantiation σ that makes the states of affairs `*punishable(x)/σ` and `*protected(x)/σ` co-obtain in a world in which the constraint in c exists.

EXAMPLE 5

```
1: *thief(x) ⇒ *punishable(x)
3: *minor(x) ⇒ *~punishable(x)
6: *minor(x) ⇒ *~thief(x)}
```

The rules 1, 3 and 6 are logically inconsistent, because it is logically possible that somebody is both a thief and a minor, while it is logically impossible that somebody both is and is not punishable. Notice that rule 6 has no influence on the consistency of the set as a whole. The inconsistency is caused by the rules 1 and 3, and cannot be removed by rule 6.

EXAMPLE 6

```
1: *thief(x) ⇒ *punishable(x)
3: *minor(x) ⇒ *~punishable(x)
-----------------------------------------
c = {6: *minor(x) ⇒ *~thief(x)}
```

We have seen that the three rules taken together are *logically* inconsistent. However, the conditions of the rules 1 and 3 are not compatible relative to c, because there can be no instantiation σ that makes the states of affairs `*thief(x)/σ` and `*minor(x)/σ` co-obtain in a world in which the constraint in c exists. Therefore the rules 1 and 3 are consistent against the background of c.

EXAMPLE 8

```
   1: *thief(x) ⇒ *punishable(x)
   3: *minor(x) ⇒ *~punishable(x)
---------------------------------------------
   c = {*minor(x) ⇒
          *exception(thief(x) ⇒ punishable(x))}
```

The rules 1 and 3 are consistent relative to c, because there can be no instantiation σ of *x* that makes the state of affairs `*minor(x)/σ` obtain in a world in which the constraint in c holds, and in which the state of affairs `*exception(thief(x)/σ ⇒ punishable(x)/σ)` does not obtain.

11. MINIMIZING EXCEPTIONS

Arguably there are no exceptions to rules whose conditions are satisfied, unless there is a special reason for it. Such a reason consists of facts that are made into an exception by some other rule. It is possible to modify the constraints on worlds that are possible according to Rule Logic to take this into account. The result of such a modification is that the number of exceptions is minimized to those that are necessary because of the other facts and rules that obtain in the world, namely to the so-called 'grounded' exceptions. Minimization of exceptions is a logical technique that is widely employed in the study of so-called non-monotonic logics.[27] In this chapter I will present a technique for minimizing exceptions that is based on the observation that exceptions to rules are reason-based in the sense that there cannot be an exception without a reason for its existence. To this purpose I will built on the way exceptions have been dealt with in chapter 3, sections 5.4 and 7.

According to the analysis presented there, there is an exception to a rule when the rule is applicable and nevertheless not applied. Since applicability is a contributive reason to apply a rule, non-application must either be based on contributive reasons against application that outweigh the reasons for application (including the rule's applicability), or on a decisive reason against application. In other words, RBL requires that exceptions are based on reasons. Moreover, RBL requires that contributive reasons are based on abstract reasons, thereby preventing that there are free floating contributive reasons. By making the additional demand that decisive reasons against the application of a rule are based on rules that apply:

[27] Cf. Lukaszewicz 1990, chapter 6.

$$\forall {*r}, rule(\mathrm{Dr}({*r}, {*\mathrm{exception}}(rule)) \equiv \exists {*c}(\mathrm{Applies}({*c} \Rightarrow {*\mathrm{exception}}(rule))))$$

it is safeguarded that there are no free floating exceptions. If the counterparts of this sentence and the axioms of RBL are added to the constraints on possible worlds according to Rule Logic, the occurrence of free floating exceptions in these worlds is prevented.