Dynamics of Legal Provisions and its Representation

Jacek Martinek  
Poznan University of Technology  
Institute of Control and Information Engineering  
Pl. M. Sklodowskiej-Curie 5  
60-965 Poznan, Poland  
jacek.martinek@put.poznan.pl

Jolanta Cybulka  
Poznan University of Technology  
Institute of Control and Information Engineering  
Pl. M. Sklodowskiej-Curie 5  
60-965 Poznan, Poland  
jolanta.cybulka@put.poznan.pl

ABSTRACT
Legal provisions are considered as interrelated fragments of a text with some structural relations which hold between them. Some provisions are treated as meta-provisions, in case they are used to enact, repeal or amend the substantial provisions. The meaning of a meta-provision is described by meta-norms. Every meta-norm is conditioned by a certain event, and it describes the action which should be executed in order to obtain the current properties of the provision, such as its current text content or the current structural relations holding between the provisions. The presented model of the legal provisions dynamics is based on the event calculus, which is used to represent how the provision properties change in time. The model can be easily implemented in Prolog.

KEYWORDS
Legal provision, legal provisions change in time, legal provisions dynamics model, meta-norm, event calculus.

1. INTRODUCTION
It is well known that legal provisions change, in some cases very frequently as it is, for instance, with the Polish statute on social security, which was amended seventeen times during the period of 1998-2001. So, the efficient information system dedicated to represent the provisions should cover their dynamics, and should be able to establish the current version of any required provision of the considered domain. It appears that the application of the event calculus [Kowalski and Sergot [4], Kesim and Sergot [3]] to the knowledge base modelling gives an interesting and a practically valuable possibility to express the changes in time of the provisions. This fact inspired our research, as a result of which the model of such a knowledge base was created. Additionally, the model is expressed in the Prolog language as to make the knowledge-based system implementation easier.

The event calculus was used to formalise the norm dynamics by Marin and Sartor [5]. Furthermore, Gangemi and others [2] created an ontological approach to represent the norms changes, in which legal texts should be carefully preprocessed in order to establish the set of ontological concepts used in the provisions. However, it seems that following this approach requires much work to be done, as it is, for example, in case of using the frames of norms, actions and concepts (Visser et al. [7]) to conceptually characterize legal texts. For this reason, we suppose that the ontological approach can be only applied to the selected pieces of legal texts in a certain domain.

In the presented model, legal provisions are considered as the interrelated pieces of text, which have been isolated from the whole legal text. Such a preprocessing enables to establish the structural relations between them and to indicate these meta-provisions which are used to enact, repeal or amend the substantial provisions. In our approach we introduce the meta-norms which determine the meaning of meta-provisions. This idea was inspired by the concept of a norm frame used by Visser et al. [7]. Marin and Sartor [5] consider the norms which are conditioned by events, and we have observed that our meta-norms are also conditioned by some specific events. Recently, Boer et al. [1] have been introducing a set of legislative events and studying their influence on a lifecycle of legislation. Their considerations are based on the Dutch legislation. The meta-norms introduced in our approach describe certain actions which should be executed in order to obtain the current properties of provisions such as their current text content, or the current structural relations holding between the provisions. We propose a set of meta-norms actions and we describe their component operations. The aim of our project is to obtain such a knowledge base which is able to answer the questions about the provision properties at any arbitrary point of time.

In Section 2 the principles of the Polish legislative technique are described. They serve as a source material on which we base our solution. The main ideas of the solution are given in Section 3 and its details are described in Section 4.

2. RULES OF LEGISLATIVE TECHNIQUE
The rules concerning the creation of legal texts in the Polish legislation are given in the resolution of the Council of Ministers, which was published on the 5th of November 1991 [6]. In accordance with these rules, there are three types of statutes: a main, an amending and an introductory statute (the last one is used to introduce the main statute). The main statute is this entity which may be changed as a consequence of the amendment. As to the amendments and the introductory statutes, we may assume that they are changeless when taken as a whole. Nevertheless, some of their provisions may change. Such changes, however, have a reference to the main statute, so one may consider that only the main statute changes. The amendment statute usually consists of repealing, supplementary, transitional and adaptation provisions. When the editing process of the main statute is done via the insertion of new legal articles or sections, the old numbering of the structural elements of this statute must be retained. The numbers of the newly inserted pieces of text are given the letter suffixes (for example, the number “1” is followed...
by the number “1a”). For a repealed provision, a textual "tombstone" is used which states that the provision has been repealed.

If a provision is amended, its new textual contents should be inserted as a whole in the amending statute. The considered legislative rules also define the way in which provisions are interrelated. The notion of a basic provision is introduced, the contents of which may be extended by some additionally specifying provisions. Moreover, there are some provisions basically used to generalise the other ones (the generalising provisions) or to show the way of interpreting other provisions (the interpretative provisions).

3. MAIN IDEAS OF THE MODEL

We admit a legal provision to be a piece of the statutory text, which forms a separated and marked unit (or can be separated and marked). Usually, a provision is encoded either as a single article without sections or as a section. In some cases, however, a section is (or can be) split into minor items and some of them may be regarded as provisions. The provisions are divided into substantial provisions (concerning the normalized domain matters) and meta-provisions, which are of the control nature and are used to enact, repeal, amend or interpret other provisions.

The knowledge base used to represent the dynamics of legal provisions contains some data about the events, such as: the statute promulgation, the statute enactment and others, which are described in the sequel. We suggest assigning the special meta-norms to the provisions. The meta-norms are conditioned by the enactment of the events and their role is to order some actions to be done. The actions may be instantaneous or durative (among the durative actions there are, for example, a transitional suspension of a provision and a transitional provision prolongation). For the durative action there exists a special limitary event which terminates its activity (or in some cases it begins it), and then it conditions the meta-norm which orders some instantaneous action to be performed. For example, bringing into force such a meta-provision that orders the suspension of some indicated provision for some indicated period, may be regarded as introducing a meta-norm which orders the execution of the "suspension" action during this indicated time. When the limitary event that represents the ending of the suspension action happens, the instantaneous action enacting the suspended provision should be executed. In a special case of the retroactive provision enactment, the ordered action is also a durative one, but its limitary event is the beginning of this action execution in some point of time in the past.

Except the data about the events which happened in the past, the knowledge base keeps a chronologically ordered list of the forthcoming events. First of all, this list contains the data about limitary events of the actions which began and last until now. It also contains the messages concerning the statute promulgation events. Initially, however, the considered list should contain the data about the promulgation of the main statute. Having in mind all these assumptions, there are two kinds of activities which may be potentially executed in the knowledge base: either forcing the first event from the list to occur or enlisting the data about the newly promulgated statute (e.g. the amending statute) in the chronologically proper place. Forcing the occurrence of an event results in executing the actions of all meta-norms which are conditioned by this event. It is possible for an amending statute to contain a provision which changes the date of an anticipated event that has not happened yet. This change may concern the date of the statute temporary force termination or of the provision suspension. Therefore, it is justifiable that there exists the meta-norm which forces the execution of the action of the date change of the limitary event. This action execution results in obtaining the properly chronologically ordered list of the forthcoming events.

We also suggest that the knowledge base should represent the information about the structural relations that hold between provisions, as it was mentioned in section 2. They are: specifying, generalising and interpreting relations. It is worth noticing that the data about relations (the structural facts) may change due to the provisions amendments.

In order to speed up the search for provisions in the knowledge base an index is introduced, in which the identifiers of the provisions are matched with short names representing the subjects of provisions. The index may concern the basic provisions only. This pair – an identifier and a name of a provision – is time invariable, while the whole index may be changed due to a sequence of amendments.

4. DETAILS OF THE PROVISION DYNAMICS MODEL

In the present part of the report we discuss the problem of identifying the presented model entities and forming their descriptions in the knowledge base. The facts concerning entities are divided into the facts which are invariable in time (the constant facts), and the facts which have the starting and terminating dates assigned to them. The constant facts are described in the knowledge base by means of the clauses (the atomic logical formulas), while the non-constant facts are expressed by terms, which are the arguments of special clauses used to represent their starting and terminating dates caused by the occurrence of an event.

The statute is identified by the number Nr, with which the following constant data are connected: Kind (one element of the set {main, amending, introductory}) and Name (stating the name of the statute). The following clause gathers all this data:

\[ \text{statute(Nr,Kind,Name).} \]

The relations between the amending (or the introductory) and the main statutes are described as:

\[ \text{amends(Nr,MainStatuteNr)} \]
\[ \text{introduces(Nr,MainStatuteNr).} \]

The substantial provision is identified by a term of the form:

\[ p(\text{StatuteNr,ArticleNr,SectionNr,ProvisionNr}) \]

in which StatuteNr is always the main statute number, regardless of the provision origin which may be in the main or in the amending statute. ArticleNr represents the number of the article which is the provision source, SectionNr constitutes the section number or zero if the article is not divided into sections, and ProvisionNr is the number of the provision. The provisions have numbers assigned during the preprocessing phase of the statute text, in which provisions are isolated and marked by a person who describes the statute (the scope of numbers proceeds within an article or a section).
The meta-provision is identified by a term of the form:

\[ mp(StatuteNr, ArticleNr, SectionNr, ProvisionNr) \]

in which \(\text{StatuteNr}\) is the number of the statute containing the provision, and the other arguments have the same meaning as the arguments of the \(p\) term above.

The textual contents \(\text{Text}\) is assigned to the provision identifier \(\text{ProvisionId}\) by means of the following term (which stands for a non-constant fact):

\[ \text{text}(\text{ProvisionId}, \text{Text}) \]

It is noteworthy that the amendments may influence on the textual contents of a provision.

The identifier of an event is built of the event’s name and some event specifying arguments (e.g. \(\text{statute_promulgation}(\text{StatuteNr})\) or \(\text{statute_enactment}(\text{StatuteNr})\)). Additionally, the event is specified by means of three clauses:

- \(\text{happens} (\text{EventId}, \text{Time})\)
- \(\text{initiates} (\text{EventId}, \text{FactList})\)
- \(\text{terminates} (\text{EventId}, \text{FactList})\)

in which \(\text{EventId}\) is the event identifier, \(\text{Time}\) is the day on which the event happened and \(\text{FactList}\) is the list of facts which the event initiates (or terminates); one event may cause the beginning or the termination of many facts.

The meta-norms assigned to provisions identified by \(\text{ProvisionId}\) are specified by the clauses of the form:

- \(\text{mn}(\text{ProvisionId}, \text{EventId}, \text{InstantaneousAction})\)
- \(\text{mn}(\text{ProvisionId}, \text{EventId}, \text{DurativeAction}, \text{LimitaryEventId}, \text{LimitaryEventTime})\)

We propose the following instantaneous actions of meta-norms. They are expressed via eight types of terms:

- \(\text{provision_enactment}(\text{ProvisionId}, \text{Text}, \text{NewStructFacts})\)
- \(\text{suspended_provision_enactment}(\text{ProvisionId}, \text{Text}, \text{NewStructFacts})\)
- \(\text{provision_change}(\text{ProvisionId}, \text{Text})\)
- \(\text{statute_repealing}(\text{StatuteNr})\)
- \(\text{provision_repealing}(\text{ProvisionId})\)
- \(\text{prolonged_provision_repealing}(\text{ProvisionId})\)
- \(\text{limitary_event_time_change}(\text{LimitaryEventId}, \text{NewTime})\)
- \(\text{retroactivity_recording}(\text{ProvisionId})\)

The durative actions of meta-norms are represented by five terms (accompanied by information about the limitary event):

- \(\text{temporary_holding_in_force}(\text{StatuteNr})\)
- \(\text{provision_prolongation}(\text{ProvisionId})\)
- \(\text{provision_suspension}(\text{ProvisionId})\)
- \(\text{retroactive_provision_enactment}(\text{ProvisionId})\)
- \(\text{vacatio_legis}(\text{StatuteNr})\)

The details concerning the operations, which are constituent operations of actions, are given in Table 1. In this table there are the following pairs: an action (a clause in italic letters) and its description under the thin line. We assume that the action is conditioned by the event identified by \(\text{EventId}\).

<table>
<thead>
<tr>
<th>Table 1. Meta-norms actions and their constituent operations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
In Table 1 we introduce additional terms:

- Adjoining the fact prolonged(ProvisionId) to the list of facts which are initiated by the event identified by EventId (the limitary event provision_prolongation_ends(ProvisionId) causes the execution of the action prolonged_provision_repealing(ProvisionId)).

- No operations are executed (the limitary event statute_temporal_force_ends(StatuteNr) causes the execution of the action statute_repealing(StatuteNr)).

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>provision_prolongation</td>
<td>function to prolonged(ProvisionId) to the list of facts which are terminated by the event identified by EventId (the limitary event provision_prolongation_ends(ProvisionId) causes the execution of the action prolonged_provision_repealing(ProvisionId)).</td>
</tr>
<tr>
<td>provision_suspension</td>
<td>function to suspended(ProvisionId) to the list of facts which are initiated by the event identified by EventId (the limitary event provision_suspension_ends(ProvisionId) causes the execution of the action provision_suspension_ends(ProvisionId,Text,NewStructFacts)).</td>
</tr>
<tr>
<td>retroactive_provision_enactment</td>
<td>function to retroactive(ProvisionId) to the list of facts which are terminated by the event identified by EventId (the limitary event retroactivity_period_starts(ProvisionId) causes the execution of the action retroactivity_recording(ProvisionId)).</td>
</tr>
<tr>
<td>vacatio_legis</td>
<td>function to the list of structural facts concerning the specific, actual state of a statute which should be inserted into the knowledge base.</td>
</tr>
</tbody>
</table>

In Table 1 we introduce additional terms:

- to express some facts concerning the specific, actual state of a provision.
- Adjoining the fact retroactive(ProvisionId) to the list of facts which are terminated by the event identified by EventId (the limitary event retroactivity_period_starts(ProvisionId) causes the execution of the action retroactivity_recording(ProvisionId)).
- No operations are executed (the limitary event statute_enactment(StatuteNr) will cause the execution of actions of meta-norms which are conditioned by the event).
holds_at(Fact,Time):-
  happens(Event,TimeI),
  TimeI =< Time,
  initiates_fact(Event,Fact),
  not(broken(Fact,TimeI,Time)).

initiates_fact(Event,Fact):-
  initiates(Event,FactList),
  member(Fact,FactList).

broken(Fact,TimeI,Time):-
  happens(Event,TimeE),
  TimeI < TimeE, TimeE =< Time,
  terminates_fact(Event,Fact).

terminates_fact(Event,Fact):-
  terminates(Event,FactList),
  member(Fact,FactList).

5. SHORT EXAMPLE

Let us consider a short example illustrating how the provisions
may be described by means of our representation method. The
example is based on the Polish statute on social security, to which
we assign the number 1. The first article of the statute contains a
definition of the social security as an institution. By means of the
predicate index we attach the name ‘Social security definition’
to the provision identified by p(1,1,1,1):

index(p(1,1,1,1),’Social security definition’).

For technical reasons the auxiliary text identifier t11 is introduced
together with the fact:

full_text(t11,”Art.1.1. The social security is an institution of
the social politics of the state, ... ”).

To the considered provision we assign the meta-norm of the form:

mn(p(1,1,1,1),statute_enactment(1),
  provision_enactment(p(1,1,1,1),t11,[basic(p(1,1,1,1))])).

So the provision is enacted when the whole statute is enacted.

One of the last articles (the article number 60) formulates
conditions of the statute enactment. We describe it in the
following way:

index(p(1,60,0,1),’The statute enactment’).

full_text(60,”Art. 60. The statute is enacted 30 days after its
promulgation ”).

mn(p(1,60,0,1),statute_promulgation(1),
  provision_enactment(p(1,60,0,1),t60,[basic(p(1,60,0,1))])).

mn(p(1,60,0,1),statute_promulgation(1),vacatio_legis(1),
  statute_enactment(1),T):-
  happens(statute_promulgation(1),T1),
  time_distance(T1,T,30).

To the article number 60 two meta-norms were assigned. The first
of them states that the event statute_promulgation(1) should enact
the considered provision (identified by p(1,60,0,1)). In the second
meta-norm we introduced the durative action vacatio_legis(1)
with statute_enactment(1) as a limitary event. The time T of the
limitary event depends on the time T1 of the statute promul-
gation. Therefore we introduced a conditional clause with the predicate
time_distance(T1,T,D), which helps to calculate the proper date T
if the date T1 and the distance D, between T1 and T, are given.

6. CONCLUSION

The provision dynamics model presented in the paper covers a
wide spectrum of situations which may be described by the meta-
provisions. The model can be easily implemented in Prolog. The
description of provisions may be done first in some markup
language, such as XML, and then translated into the Prolog
representation proposed in the paper.

The research was supported by Politechnika Poznanska, the grant

7. REFERENCES

Management System based on an Event-based Model of
Version Management Information in Legislation. In T.
Gordon (ed.), Legal Knowledge and Information Systems.
Jurix 2004: The Seventeenth Annual Conference.

Ontology Framework to Represent Norm Dynamics. In
Proceedings of Second International Workshop on Legal

for Modelling Temporal Objects. IEEE Transactions on
Knowledge and Data Engineering, 8, 5 (Oct. 1996), 724-741.


event-calculus. In Proceedings of the Seventh International
Conference on Artificial Intelligence and Law (ICAIL 99)
(Oslo, Norway, June 14-18, 1999). ACM Press, New York,
1999, 90-100.

w sprawie zasad techniki prawodawczej. Monitor Polski z 16
grudnia 1991 r., M.P.91.44.310.

In Proceedings of the Sixth International Conference on
Artificial Intelligence and Law (ICAIL ’97) (Melbourne,
Australia, June 30 - July 3, 1997). ACM Press, New York,