Document Structuring à la SDRT

Laurence Danlos, Bertrand Gaiffe, Laurent Roussarie

▶ **To cite this version:**

Laurence Danlos, Bertrand Gaiffe, Laurent Roussarie. Document Structuring à la SDRT. Proceedings of the European Workshop on Generation, ACL, Toulouse, 2001, Toulouse, France. halshs-00082836

HAL Id: halshs-00082836
https://shs.hal.science/halshs-00082836

Submitted on 10 Jul 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Document Structuring à la SDRT

Laurence Danlos
LATTICE – LORIA
U. Paris 7, Case 7003
2, pl. Jussieu
75251 Paris Cedex 05
France
danlos@linguist.jussieu.fr

Bertrand Gaiffe
LORIA
Campus scientifique
BP239
54506 Vandœuvre Cedex
France
gaiffe@loria.fr

Laurent Roussarie
LATTICE
U. Paris 7, Case 7003
2, pl. Jussieu
75251 Paris Cedex 05
France
roussari@linguist.jussieu.fr

Abstract

In this paper, the issue of document structuring is addressed. To achieve this task, we advocate that Segmented Discourse Representation Theory (SDRT) is a most expressive discourse framework. Then we sketch a discourse planning mechanism which aims at producing as many paraphrastic document structures as possible from a set of factual data encoded into a logical form.

1 Introduction

Using the terms of (Reiter and Dale, 2000), we consider that the Document Planner architecture is pipelined: first the content determination task does its work, and then the document structuring task takes the result and build a document plan. Following the work of (Roussarie, 2000), we adopt SDRT (Asher, 1993; Asher and Lascarides, 1998), which was designed first for text understanding, for the document structuring task[1].

The input to the document structuring component is a set of factual data encoded into a logical form, as in (1).

(1) $\exists\, e_1, e_2, x, y\ (e_1$–leave$(x)\ \wedge\ e_2$–fit-of-tears$(y) \wedge$ *cause*$(e_1, e_2)\ \wedge$
$x =$Fred $\wedge\ y =$Mary $\wedge\ e_1 < now\ \wedge\ e_2 < now)$

This level of representation is supposed to be language independent, although we use English-like predicates for the sake of simplification. (1) includes a conceptual (language independent) relation, i.e. *cause*, between the events e_1 and e_2.

A document plan is a SDRS. Our goal is to produce a wide range of paraphrases from the same factual data. For example, from the logical form in (1), we want to produce at least all the texts in (2). These texts have different communicative structures and so correspond to different communicative goals. However, these issues will not be addressed here.

(2) a. Fred left. Therefore, Mary burst into a fit of tears.
b. Mary burst into a fit of tears because Fred left.
c. Fred left. His leaving brought Mary into a fit of tears.
d. Mary burst into a fit of tears. This is due to Fred's leaving.

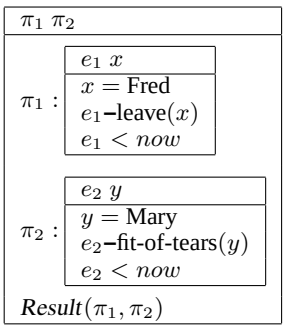
To produce paraphrases, we start by producing several document plans (i.e. SDRSs) from the same factual data. The SDRS underlying (2a) is in (3a) in which the discourse relation *Result* between π_1 and π_2 expresses the predicate *cause*(e_1, e_2). The SDRS underlying (2b) is similar to (3a) except that *Explanation*(π_2, π_1) is involved instead of *Result*(π_1, π_2). The SDRS underlying (2c) is in (3b). It includes the discourse relation *Commentary*[2] defined in (Asher, 1993). To ensure the cohesion of texts, we add the following constraint to his definition: *Commentary*(π_1, π_2) requires that one element in π_2 is coreferent with one element in π_1, as it is the case in (3b) with $e_3 = e_1$. In (3b), the causal relation has been reified as the discourse referent *f* (see section 5). This discourse referent is ex-

[1] As far as we know, Roussarie is the first author who has adopted SDRT for text generation. The work presented here differs from his work in the following: content determination and document structuring are pipelined here, while they are interleaved in his work.

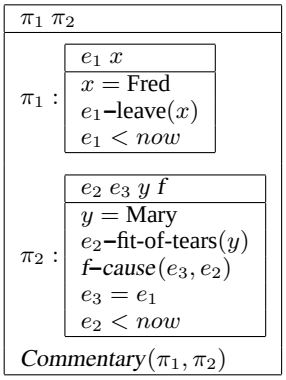
[2] The discourse relation in (2c) is not *Result* since the second sentence denotes both the cause and the effect.

pressed through the verb *bring into*[3]. The SDRS underlying (2d) is similar to (3b).

(3) a.



b.



When provided as input to a "tactical component" (microplanner and surface realizer), a given SDRS leads to zero, one or several texts. It leads to nothing when there is a lexical (or syntactic) gap in the target language. For example, if there is no verbal predicate semantically equivalent to *be due to* in the target language, the SDRS underlying (2d) leads to nothing. Similarly, if a SDRS includes a discourse relation which cannot be realized in the target language (e.g. volitional-Result proposed in (Mann and Thompson, 1987) cannot be linguistically realized in French (Danlos, 1998)), it leads to nothing[4]. A given SDRS leads to several texts when there are several lexicalizations for at least one predicate.

Thanks to the use of SDRT, we are able to give a formal background to the following assumptions generally used in bottom-up document-structuring approaches:

- "The content determination mechanism has produced a set of messages which are required to be included in the final document plan" (Reiter and Dale, 2000, p. 114). In formal terms, it translates as follows: a SDRS π built from a logical form LF is such that the logical form derived from π is logically equivalent to LF. For example, the logical forms derived from the SDRSs in (3a) and (3b) are equivalent to that in (1) modulo axioms which will be presented in section 4.
- "The NLG system has a means of determining what discourse relation (if any) can be used to link two particular messages or component document plans" (Reiter and Dale, 2000, p. 114). Our formal approach is based on reversing the SDRT conditions to establish discourse relations. As an illustration, in SDRT for text understanding, there is the Axiom in (4) for *Narration*. This axiom states that if *Narration* holds between two SDRSs π_1 and π_2, then the main event (*me*) of π_1 happens before the main event of π_2.

(4) $\Box(\textit{Narration}(\pi_1, \pi_2) \rightarrow \textit{me}(\pi_1) < \textit{me}(\pi_2))$

For text generation, this axiom is reversed in the rule in (5) which is domain and language independent. (5) is taken from (Roussarie, 2000, p. 154).

(5)
- If k and k' are DRS the main eventualities of which are not states,
- and if the main event of k occurs before the main event of k',
- then *Narration*(π, π') is a valid relation, where π and π' respectively label k and k'.

This paper is organized as follows. Section 2 gives a crash course in SDRT. Section 3 compares our approach to document structuring to those based on RST. Section 4 explains the axioms needed to lay down the logical equivalence of SDRSs such that (3a) and (3b). Section 5 explains the process for building SDRSs. Section 6 sketches how to generate a text from a SDRS. Section 7 illustrates the document structuring strategy on examples.

[3]In the generation community, causative verbal predicates such as *bring into* or *provoke* are considered as elementary ones, although it should not be so. For example, *Elixir provokes an allergic reaction* is not analyzed and so is simply represented as (allergic-reaction (Elixir)) in (Bouayad-Agha et al., 2000). Whereas, it should get a representation translating *x's taking Elixir causes x's having an allergic reaction* with a causal relation between two events.

[4]We adopt the position that there exists a set of discourse relations which are likely to be language independent.

2 Crash course in SDRT

2.1 Introduction

SDRT (Segmented Discourse Representation Theory) was introduced in (Asher, 1993) as an extension of DRT (Discourse Representation Theory, (Kamp and Reyle, 1993)) in order to account for specific properties of discourse structure.

The original motivation for developing SDRT can be found in Asher's study of the reference to abstract objects in discourse. Asher argues that a sound discourse theory has to cope with some anaphora whose antecedents turn out to be text segments larger than a clause or a sentence. Moreover, it is necessary to reveal a hierarchical discourse structure which makes appear the sites available for anaphora–antecedent binding. Consider the example in (6) taken from (Asher, 1993, p. 318):

(6) (1) After 38 months, America is back in space. (2) The shuttle Discovery roared off the pad from Cape Kennedy at 10:38 this morning. (3) The craft and crew performed flawlessly. (4) Later in the day the TDRS shuttle communication satellite was successfully deployed. (5) **This** has given a much needed boost to NASA morale.

The pronoun *this* (6.5) can only refer to the whole trip or (possibly) to the last mentioned event (TDRS launch). Consequently, the structure of (6) must be such that : i) there exists a constituent which semantically encompasses the whole story (6.1–4), and ii) neither (6.2) nor (6.3) correspond to available constituents for the anaphora resolution when computing the attachment of (6.5) in the context. *Availability* (or openness) of constituents is a formal property that can be accounted for by the use of discourse relations.

2.2 DRSs as formal discourse units

SDRT can be viewed as a super-layer on DRT whose expressiveness is enhanced by the use of discourse relations. Thus the DRT structures (Discourse Representation Structures or DRS) are handled as basic discourse units in SDRT.

Formally, a DRS is a couple of sets $\langle$U,Con$\rangle$. U (the universe) is the set of discourse referents. Con is a set of conditions which describe the meaning of the discourse in a truth-conditional semantics fashion. For instance, the DRS representing the sentence (7a) is given in (7b).

(7) a. Fred left.

b.
$x\ e$
$x =$ Fred
e–leave(x)
$e < now$

Note that in addition to individual referents (x), U includes event referents (e). DRT adopts a Davidsonian approach (Davidson, 1967): it considers that events have to be denoted by singular terms in the logical form of sentences. In the semantic model, events are handled as world immanent entities, and event referents (e) can occur in argumental slots of certain predicates (like *f-cause*(e_3, e_2) in (3b)). The statement e–leave(x) is a predicative notational variant and stands for "e is a leaving of x".

DRSs do not correspond to linguistic categories but are formal units: from the SDRT point of view, one should see them as (intensional) meaning structures. This is why some discourse abstract objects (such as facts, situations, propositions...) can be referred to by discourse referents (we will say that they are reified) and semantically characterized by (sub-)DRS. (8) is an example of a fact reading, where $\approx$ is the characterization predicate (Asher, 1993, p. 145).

(8) a. The fact that Fred left abruptly upset Mary.

b.
$x\ y\ e'\ f$	
$x =$Fred	
$f \approx$	e
	e–leave(x)
	abrupt(e)
$y =$Mary	
e'–upset(f, y)	

2.3 Discourse Relations and SDRSs

A SDRS is a couple of sets $\langle$**U**,**Con**$\rangle$. **U** is a set of labels of DRS or SDRS which may be viewed as "speech act discourse referents" (Asher and Lascarides, 1998). **Con** is a set of conditions on labels of the form:

- $\pi : K$, where π is a label from **U** and K is a (S)DRS (labelling);

- $R(\pi_i, \pi_j)$, where π_i and π_j are labels and R a discourse relation (structuring).

The set of SDRT relations includes *Narration* (for temporal sequence), *Background* (for temporal overlap), *Elaboration* (for whole-part or topic-development), *Explanation* and *Result* (for causation), *Commentary* (for gloss).

According to (Asher, 1993, p. 319), (9) sketches the SDR-theoretic analysis of (6) where each k_i stands for the DRS representing the content of the i^{th} sentence in (6).

(9)
$\pi_1\ \pi_1'\ \pi_5$
$\pi_1 : k_1$
π_1' : [$\pi_2\ \pi_3\ \pi_4$ | $\pi_2 : k_2 \quad \pi_3 : k_3 \quad \pi_4 : k_4$; *Commentary*$(\pi_2, \pi_3)$; *Narration*$(\pi_2, \pi_4)$]
Elaboration(π_1, π_1')
$\pi_5 : k_5$ *Commentary*(π_1, π_5)

SDRSs are built by means of non monotonic rules that encodes discourse properties and world knowledge. For instance, one rule states that if a discourse constituent β may be connected to a discourse constituent α in the context, then *normally* the relation *Narration*(α, β) holds. Another rule states that if β may be connected to α and if the main event of β, i.e. me(β), is known as a cause of me(α), then *normally* the relation *Explanation*(α, β) holds.

3 Comparison with RST

As nearly everybody in the NLG community uses RST (Rhetorical Structure Theory, (Mann and Thompson, 1987)) as a discourse framework, it is generally considered that the task of document structuring is to produce a tree in the RST style. Since RST is a descriptive theory without any formal background, there exists a wide range of interpretations and several notions of Rhetorical Structure. For some authors, e.g. (Marcu et al., 2000), the Rhetorical Structure is very surfacic: it is an ordered tree isomorphic to the linearized structure of the text and a rhetorical relation can be viewed as a nickname for a small set of cue phrases. For other authors, the rhetorical structure is more abstract: it aims at representing meaning. For example, in (RAGS Project, 1999; Bouayad-Agha et al., 2000), the Rhetorical Structure is an unordered tree in which terminal nodes represent elementary propositions, while non terminal nodes represent rhetorical relations which are abstract relations such as *cause*. This rhetorical representation is mapped into a Document Representation which is an ordered tree reflecting the surfacic structure of the text.

Our approach is closer to the RAGS'one if we consider our logical form as equivalent to their Rhetorical Structures. However, we differ basically on the following point: their Rhetorical Structure is a tree, while our logical form, when graphically represented, is a (connex) graph and not a tree. Let us justify our position by considering the discourses in (10).

(10) a. Fred run the vacuum cleaner$_A$ while Sue was sleeping$_B$ in order to bother her$_C$.
b. Fred run the vacuum cleaner$_A$ while Sue was sleeping$_B$ in order to please her$_C$.

They can be given various meanings, however we focus on the following:

- for (10a), running the vacuum cleaner is supposed to be noisy and Fred attempts to bother Sue by making something noisy exactly when she is sleeping,
- for (10b), running the vacuum cleaner is supposed to be an awful chore and Fred attempts to please Sue by relieving her of a chore. It just happens that he run the vacuum cleaner while she was sleeping.

In RST, both (10a) and (10b) are given the tree representation in (11), in which CIRC abbreviates CIRCUMSTANCE.

(11)
PURPOSE (N, S)
CIRC (N, S) C
A B

The semantic interpretation of a rhetorical tree is given by the "nuclearity principle" (Marcu, 1996): whenever two large spans are connected through a rhetorical relation, that rhetorical relation holds also between the most important parts of the constituent spans. In (11), the nuclearity principle amounts to saying that there is only

one interpretation, namely that in which the nucleus argument of PURPOSE is A, which is the nucleus argument of CIRCUMSTANCE. This is the right interpretation for (10b). However, (11) cannot represent the meaning of (10a) for which the first (nucleus) argument of PURPOSE is the sub-tree rooted at CIRCUMSTANCE. In conclusion, a RST tree structure is too poor: it cannot account for the expressiveness of texts. This can be accounted for by the use of representations which correspond graphically to (connex) graphs. The graphical representations of (10a) and (10b) and their equivalent in pseudo logical forms are respectively shown in (12a) and (12b)[5].

(12) a.

$$\overset{\text{PURPOSE}}{\underset{1\quad 2}{\left[\begin{array}{cc}\multicolumn{2}{c}{\text{CIRC}}\\ 1 & 2\\ A & B\end{array}\right]_D}}\quad C$$

$$\exists\, A, B, C, D\ (D\text{–}\textsc{circ}(A, B) \wedge \textsc{purpose}(D, C))$$

b.

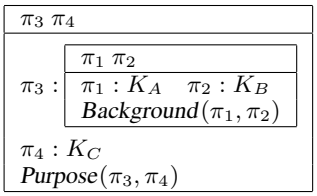
$$\overset{\text{PURPOSE}}{\underset{2}{C}}\qquad \overset{\text{CIRC}}{\underset{1\ 1}{A}}\qquad \underset{2}{B}$$

$$\exists\, A, B, C\ (\textsc{circ}(A, B) \wedge \textsc{purpose}(A, C))$$

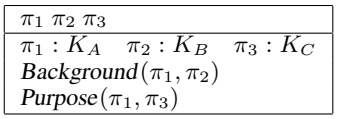
(12a) is a tree in which the first argument of PURPOSE is D, the sub-tree rooted at CIRCUMSTANCE. It is the interpretation of the RST tree in (11) *without* the nuclearity principle. (12b) is a graph in which A is part of two relations[6]. This graph corresponds to the interpretation of the RST tree in (11) given by the nuclearity principle. This principle makes that A is part of both the relation PURPOSE with C and the relation CIRCUMSTANCE with its satellite B.

The SDRSs underlying (10a) and (10b) are shown respectively in (13a) and (13b) (the notation K_A stands for the DRS representing A and so on). Here we replace CIRCUMSTANCE by the SDRT relation *Background* for temporal overlap[7].

(13) a.



b.



In (13a), the first argument of *Purpose* is π_3 which groups K_A and K_B which are linked through *Background*. In (13b), π_1 is part of two discourse relations. The graphical representations of (13a) and (13b) (in which $R(\pi_1, \pi_2)$ is represented as a tree rooted at R) have the same topology as (12a) and (12b) respectively.

In summary, in document structuring approaches based on RST, a rhetorical structure is always a tree, whenever understood as abstract representation or a more surfacic one. This cannot be maintained. First, if the rhetorical structure is an abstract conceptual representation closed to a logical form, its graphical representation is a connex graph (and not always a tree). Second, if the rhetorical structure is a discourse representation, as it is the case for SDRS, its graphical representation is also a connex graph.

This criticism is not the only one against RST. This discourse framework has already been criticized in the generation community (de Smedt et al., 1996). So we advocate the use of SDRT. This theory presents the following advantages :

- it is a formalized theory which benefits of all the progress in formal semantics most often realized in the understanding perspective around DRT or SDRT.
- adopting SDRT for text generation by "reversing" the rules (see (4) reversed in (5)) allows us to have reversible systems: the same linguistic data can be used for both text understanding and generation.
- as it will be shown in section 5, the document structuring component à la SDRT gives hint on referring expressions: it indicates when a

[5] The arguments of a binary semantic predicate are noted as 1 and 2 after the fashion of MTT (Meaning to Text Theory, (Mel'čuk, 1988)) and not as Nucleus and Satellite in the RST tradition.

[6] This graph can be annotated to mark the element(s) which are focused on.

[7] Actually, the SDR-theoretical representations of (13) should be more complex with a pseudo-topic that would span the *Background*-related constituents. See (Asher and Lascarides, 1998) for details.

discourse referent should be expressed as an anaphoric NP.

- a SDRS (i.e. a document plan) can be given to existing microplanners and surface realizers with perhaps some modifications (see section 6). For example, a SDRS can be given as input to G-TAG (Danlos, 2000) implemented in CLEF (Meunier and Reyes, 1999) provided small fits are realized.

In conclusion, we think that SDRT is a better discourse framework than RST (for both text generation and understanding).

4 Equivalence between logical forms

Recall that we want to compute both the SDRS in (3a) with *Result* and the SDRS in (3b) with *Commentary* from the logical form in (1). Let us show that the logical forms derived from these SDRSs are equivalent. In SDRT, there is an axiom for *Result* from which one can entail the rule in (14), which is similar to the axiom in (4) for *Narration*.

(14) $\mathit{Result}(\pi_1, \pi_2) \rightarrow \mathit{cause}(me(\pi_1), me(\pi_2))$

Therefore, the logical form derived from (3a) is (1) repeated in (15) without the temporal information.

(15) $\exists\, e_1, e_2, x, y$ (e_1–leave(x) $\wedge$ e_2–fit-of-tears(y) $\wedge$ *cause*(e_1, e_2) $\wedge$ $x =$Fred $\wedge$ $y =$Mary)

The discourse relation *Commentary* per se does not add information. Therefore, the logical form derived from (3b) is (16).

(16) $\exists\, e_1, e_2, f, x, y$ (e_1–leave(x) $\wedge$ e_2–fit-of-tears(y) $\wedge$ *f–cause*$(e_1, e_2) \wedge x =$ Fred $\wedge\, y =$ Mary)

The difference between (15) and (16) consists in considering the causal relation between the two events either as only a predicate or as a variable plus a predicate. However, the axioms in (17a) and (17b) can be laid down. With these axioms, (15) and (16) are equivalent since they are both equivalent to (18), in which the causal relation gets represented twice. In other words, we have the following logical entailments: (15), (17a) $\vdash$ (18), and (16), (17b) $\vdash$ (18).

(17) a. $\forall x, y$ *cause*$(x, y) \rightarrow \exists z$ z–*cause*(x, y)
b. $\forall x, y, z$ z–*cause*$(x, y) \rightarrow$ *cause*(x, y)

(18) $\exists\, e_1, e_2, f, x, y$ (e_1–leave(x) $\wedge$ e_2–fit-of-tears(y) $\wedge$ *f–cause*$(e_1, e_2) \wedge$ *cause*$(e_1, e_2) \wedge x =$ Fred $\wedge\, y =$ Mary)

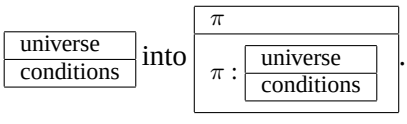
Let us underline that the content determination task may arbitrarily result in (15), (16) or even (18). Therefore, the document structuring task has to produce SDRS such as (3a) and (3b) from one of these logical forms.

There is a an important difference between SDRSs and logical forms. SDRSs represent discourses and their variables are discourse referents. Logical forms represent meanings and their variables are pure logical variables. To compute a SDRS from a logical form, one has to decide which variables from the logical form become discourse referents, as explained in the next section.

5 Building SDRSs

5.1 Basic principles

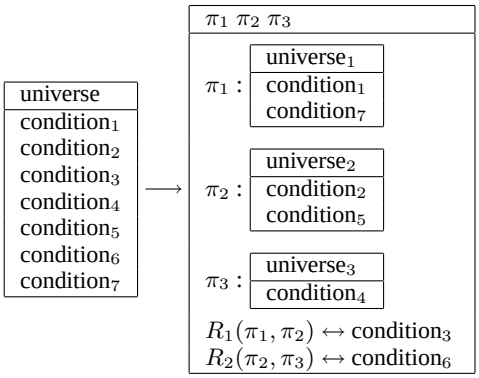
To get a recursive process, first, we translate the logical form into a DRS[8]. In case of a purely existential formula such as those we have been dealing with so far, this just amounts to putting all the variables into the universe of the DRS and split the formula into elementary conjoined conditions[9]. The document structuring task amounts then in building a SDRS from a DRS. The simplest way to do that is simply to transform :



More complex structures are obtained by splitting the DRS into sub-DRSs as illustrated below.

[8]This DRS is considered as a logical representation. It is not yet a discourse representation.

[9]More complex formulas are not considered here.



$R_1(\pi_1, \pi_2) \leftrightarrow \text{condition}_3$ means that the discourse relation R_1 to be established between π_1 and π_2 must have condition_3 among its consequences: no other element is in charge of expressing condition_3.

In SDRT for text understanding, the conditions are not ordered. However, in text generation, a document plan indicates the order of its components. As a consequence, when a document plan is a SDRS, its components (labelled π_i) have to be ordered. In the pseudo SDRS above, it is supposed that π_1 precedes π_2 which precedes π_3.

Let us examine the principles governing the splitting of the conditions and the universes. For the splitting of the conditions, the whole content of the factual database has to be verbalized. Therefore all the conditions in the DRS have to be expressed in the SDRS. Two cases appear:

- either a condition in the DRS appears as a condition in one of the sub-DRS; that is the case for $condition_1$ in the DRS labelled π_1;
- or it is expressed through a discourse relation; that is the case for $condition_3$. One of the criteria for choosing an appropriate discourse relation is that its consequences have to contain the condition involved. For example, the condition *cause*(e_1, e_2) can be expressed through *Result*(π_1, π_2) when π_1 and π_2 label the sub-DRSs that contain the descriptions of e_1 and e_2 respectively.

Let us now look on how to determine the universes of the sub-DRSs, i.e. the discourse referents. First, there are technical constraints, namely:

- the arguments of any condition in a sub-DRS must appear in the universe of this DRS;
- the universes of all the sub-DRSs have to be disjoint. This constraint is the counterpart of the following constraint in understanding: "partial DRSs introduce new discourse referents" (Asher, 1993, p. 71).

These two constraints are not independent. Assuming that the first constraint is respected, the second one can be respected with the following mechanism: if a variable x already appears in a preceding sub-DRS labelled π_x, then a brand new variable y is created in the universe of the current sub-DRS labelled π_y and the condition $y = x$ is added into the conditions of π_y. The discourse referent y will be generated as an anaphora if π_x is *available* to π_y (see section 2.1), otherwise it will be generated as a definite or demonstrative NP.

Secondly, as mentioned in section 4, it has to be decided which variables become discourse referents. When we have for instance $\exists f, e_1, e_2$ *f–cause*(e_1, e_2), we can decide to apply axiom (17b), and then remove the variable *f* and every condition having *f* as an argument (in particular the condition *f–cause*(e_1, e_2)). In order for such an operation to be valid, we have to ensure that no information is lost. In practice, this supposes that no other condition than *f–cause*(e_1, e_2) has *f* as an argument. We call this operation dereification. Conversely from such a formula as $\exists e_1, e_2$ *cause*(e_1, e_2), we can apply axiom (17a), and then remove the condition *cause*(e_1, e_2). We call this operation reification. Contrarily to dereification, no information can be lost. These two operation are a mix between something which is pure logic (that adds information) and a discourse operation that deals with discourse referents. As our objective is to build as much dicourse plans as possible, reification and de-reification are systematically performed whenever possible.

The process is recursive: once all this is done (splitting the conditions, universes determination (including reification and de-reification) and choice of discourse relations), the process can apply recursively on each of the embedded DRSs (this is the reason why the logical form is first translated into a DRS).

5.2 Algorithm

A naive solution to implement these principles will be first described. Next some refinements will be proposed.

The naive solution amounts to considering all the possible splittings of the set of conditions. If there are n conditions, the number of sub-SDRSs ranks from 1 to n. In the hypothesis of a splitting into p sub-SDRSs, each condition may be put in any of the p sub-SDRSs or in any of the $p-1$ sets of conditions to be expressed by a discourse relation[10]. Next the universes of the sub-SDRSs are built according to the principles described above. This leads to availability constraints (e.g. π_x is available to π_y) to be checked later on. In the next step, the possible discourse relations are computed according to their consequences. At this step, a lot of hypotheses are ruled out. For example, any hypothesis assuming that a condition such as e–leave(x, y) is to be expressed through a discourse relation will be ruled out. Finally, the availability constraints have to be checked using the same rules as in understanding.

With this naive solution, a lot of document plans will be rejected by the linguistic component. As an illustration, each sub-SDRS has to be verbalized as a clause (see section 6). Therefore, any sub-SDRS that does not include an eventuality or a fact will be rejected by the linguistic component.

This naive solution is theoretically valid, however it is not usable in practice. A lot of possible failures can be foreseen. For example, the conditions that can be expressed through a discourse relation, e.g. *cause*(e_1, e_2), should be considered first. If it is decided that such a condition is indeed expressed by a discourse relation, e.g. *Result*(π_1, π_2), then the sub-SDRSs π_1 and π_2 are created with the conditions concerning e_1 and e_2 respectively.

To sum up, the process of splitting the conditions should not be blind. The content of the conditions has to be taken into account in order to guide the splitting and avoid thereby failures that can be foreseen. However, the details of this optimization will not be presented here.

[10] In SDRT, any element in the universe of a SDRS must be linked to another element. Therefore, a SDRS with p sub-SDRSs must include (at least) $p-1$ discourse relations.

6 Generating a text from a SDRS

A SDRS, i.e. a document plan, is given to a microplanner and surface realizer which computes one or several texts. It is the topic of another paper to explain in detail this process. Here we will only give the basic principles which guide the choices to be made in the tactical component.

The process to generate a text from a SDRS ⟨**U**,**Con**⟩ is basically recursive:

- an element π_i in **U** has to be generated as a clause if π_i labels a DRS and recursively as a text (possibly a complex sentence) if π_i labels a SDRS.
- a condition ***R***(π_i, π_j) in **Con** has to be generated as a text "S_i. Cue S_j." or as a complex sentence "S_i Cue S_j.", where S_i generates π_i, S_j π_j, and Cue is a cue phrase lexicalizing ***R*** (Cue may be empty).
- a condition $\pi : K$ in **Con** where K is a DRS ⟨U,Con⟩ has to generated as a clause according to the following constraints:
  - in analysis, a discourse referent is the trace of either a determiner or an inflexion mark. Therefore, in generation, a discourse referent has to be generated as an NP or a tensed verb (noted V). Such an information is noted as e.g. e_1:NP/V;
  - the conditions guide the lexical choices. The conditions x = Fred correspond to proper names which is noted as x:PN[Fred]. The equality conditions between discourse referents (e.g. $x = y$) give rise to (pronominal or nominal) anaphora which is noted as x:ANA[y]. The other conditions are associated to lexical predicates.

  With these constraints, an element which is reified, e.g. *f–cause*(e_1, e_2), gives rise to an NP or a verb (*a cause of, provoke*) and an element which is not reified, e.g. *cause*(e_1, e_2), gives rise to a modifier on e_1 or e_2 with e_1 and e_2 generated either as verbs or NPs.

  This process results in a list such as:

  - e_1:NP/V[e_1–leave(x)],

– x:PN[Fred(x)],
– f:NP/V[f–*cause*(e_1, e_2)],
– e_2:NP/V[e_2–fit-of-tears(y)],
– y:PN[Mary(y)].

Such a list guides the lexical choices and syntactic realization performed by the micro-planner.

7 Illustration on examples

Let us show how to compute the SDRSs in (3a) and (3b) from the logical form in (1). First, this formula is translated in the DRS in (19), in which the conditions are numbered for the sake of convenience.

(19)

$e_1\ e_2\ x\ y$
cond$_1$: e_1–leave(x)
cond$_2$: e_2–fit-of-tears(y)
cond$_3$: *cause*(e_1, e_2)
cond$_4$: $x =$ Fred
cond$_5$: $y =$ Mary
cond$_6$: $e_1 < now$
cond$_7$: $e_2 < now$

¿From (19), one possibility is to express cond$_3$ through *Result*(π_1, π_2) in which π_1 and π_2 label the sub-DRSs grouping the conditions on e_1 and e_2 respectively. Therefore, π_1 has to group cond$_1$ and cond$_6$. As cond$_1$ introduces the variable x, cond$_4$ has to figure also in π_1[11]. The universe of the DRS labelled by π_1 is $\{e_1, x\}$. Similarly for π_2, its universe is $\{e_2, y\}$, its conditions are cond$_2$, cond$_5$ and cond$_7$. All the conditions of (19) are therefore expressed in (3a) which is a well-formed document plan. Following the rules sketched in section 6, (3a) will be generated in (2a) by the linguistic component, if *Result* is lexicalized as the cue phrase *therefore* which links two sentences.

¿From (19), another possibility is to split all the conditions into two sub-DRSs: the first one labelled π_1 grouping the conditions on e_1 (as in the previous possibility), the second one labelled π_2 grouping all the other conditions. cond$_3$ in π_2 has e_1 as argument. This variable already appears in π_1. Therefore a brand new variable e_3 is created in the universe of π_2 and the condition $e_3 = e_1$ is added in π_2. As all the conditions are split into the sub-DRSs, π_1 and π_2 have to be linked with a discourse relation which adds no information (i.e. which has no consequence). *Commentary* is such a discourse relation, and it is valid here since its constraint (one element in π_2 has to be coreferent with one element in π_1, see section 1) is respected with the coreference relation $e_3 = e_1$. At this step, the SDRS in (20) has been built.

¿From (20), one possibility is to transmit this SDRS as it is to the tactical component. If *Commentary* is lexicalized as an empty cue phrase linking two sentences, (20) will be generated in (21) where *cause*(e_3, e_2), which is not reified, is expressed through the modifier *because of*.

(20)

$\pi_1\ \pi_2$	
π_1 :	$e_1\ x$ $x =$ Fred e_1–leave(x) $e_1 < now$
π_2 :	$e_2\ e_3\ y$ $y =$ Mary e_2–fit-of-tears(y) *cause*(e_3, e_2) $e_3 = e_1$ $e_2 < now$
Commentary(π_1, π_2)	

(21) Fred left. Mary burst into a fit of tears because of that.

In text understanding, (21) is likely to be analyzed with the discourse relation *Narration*(π_1, π_2), which has for consequence $e_1 < e_2$. This condition is compatible with those in π_2 since *cause*(e_3, e_2) with $e_3 = e_1$ implies $e_1 < e_2$. So, there is no conflict between the understanding and the generation of (21).

¿From (20), another possibility is to reify *cause*(e_3, e_2) in π_2. The SDRS in (20) becomes that in (3b). If f–*cause* is lexicalized as *bring into* (a colloquial variant of *cause* when the second argument is *a fit of tears*), (2c) will be generated.

8 Conclusion

We have dealt with the document structuring task, considering that it should be able to produce several outputs so that it can cope, among other things, with real lexical gaps in the target language (and also actual gaps in a realistic surface

[11] This is an optimization: if cond$_4$ were not included in π_1, the surface realizer would fail on π_1 and the hypothesis would be ruled out.

realizer). We therefore aim at producing as much document plans as possible.

We suppose that the content determination component provides a logical form encoding the factual data to be linguistically verbalized. Axioms may apply on this logical form which enable reifications and de-reifications. As a consequence, some predicates may be realized either as a verb, an NP or a modifier.

The document structuring component is based on SDRT, a formalized discourse framework which can account for the expressiveness of texts, contrarily to RST. A document plan is a SDRS. This level of representation is likely to be language and domain independent and can be provided to existing surface realizers. Building SDRSs from a logical form is a recursive process for which a basic strategy has been presented and exemplified.

No implementation has been realized yet, however we foresee to do it and to interface it with the tactical component CLEF (Meunier and Reyes, 1999).

References

N. Asher and A. Lascarides. 1998. The semantics and pragmatics of presupposition. *Journal of Semantics*, 15(3):239–300.

N. Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer, Dordrecht.

N. Bouayad-Agha, R. Power, and D. Scott. 2000. Can text structure be incompatible with rhetorical structure ? In *Proceedings of INLG'2000*, pages 194–201, Mitzpe Ramon, Israel.

L. Danlos. 1998. Event coreference in causal discourses. In P. Bouillon, editor, *Meaning of Word*. Cambridge University Press.

L. Danlos. 2000. G-TAG: A lexicalized formalism for text generation inspired by Tree Adjoining Grammar. In A. Abeillé and O. Rambow, editors, *Tree Adjoining Grammars: formalisms, linguistics analysis and processing*, pages 343–370. CSLI Publications, Stanford.

D. Davidson. 1967. The logical form of action sentences. In *Essays on Actions and Events*, pages 105–148. Oxford Press.

K. de Smedt, H. Horacek, and M. Zock. 1996. Architectures for natural language generation: Problems and perspectives. In G. Adorni and M. Zock, editors, *Trends in Natural Language Generation. Proceedings of the 4th European Workshop, EWNLG'93, Pisa*. Springer-Verlag.

H. Kamp and U. Reyle. 1993. *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht, The Netherlands.

W. C. Mann and S. A. Thompson. 1987. Rhetorical structure theory: Description and construction of texts structures. In G. Kempen, editor, *Natural Language Generation*, pages 85–95. Martinus Nijhoff Publisher, Dordrecht.

D. Marcu, L. Carlson, and M. Watanabe. 2000. An empirical study in multilingual natural language generation: What should a text planner do? In *Proceedings of INLG'2000*, pages 17–23, Mitzpe Ramon, Israel.

D. Marcu. 1996. Building up rhetorical structure trees. In *The Proceedings of the 13th National Conference on Artificial Intelligence*, pages 1069–1074, Portland, Oregon.

I. Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press, New York.

F. Meunier and R. Reyes. 1999. La plate forme de développement de générateurs de textes CLEF. In *Actes du 2ème Colloque Francophone sur la Génération Automatique de Textes, GAT'99*, Grenoble.

RAGS Project. 1999. Towards a reference architecture for natural language generation systems. Technical Report ITRI-99-14, Information Technology Research Institute (ITRI), University of Brighton.

E. Reiter and R. Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press.

L. Roussarie. 2000. *Un modèle théorique d'inférences de structures sémantiques et discursives dans le cadre de la génération automatique de textes*. Thèse de doctorat en linguistique, Université Denis Diderot, Paris 7.