

Guidelines for using ISO standard 24617-2

February 2017

Harry Bunt

Tilburg University

`harry.bunt@uvt.nl`

Overview

This document provides practical information for using the ISO 24617-2 standard for dialogue annotation. The document is organized as follows. Section 1 discusses some general issues in dialogue act annotation. Section 2 explains the segmentation of a dialogue into the segments of communicative behaviour that express a dialogue act (possibly more than one). Section 3 shows the ISO 24617-2 metamodel and the corresponding components of an ISO 24617-2 annotation. The sections 4-7 contain guidelines for applying the concepts of the ISO 24617-2 metamodel: dimensions, communicative functions, qualifiers, functional dependences, feedback dependences, and rhetorical relations. Section 8 introduces the use of alternative representation formats of ISO 24617-2 annotations in the Dialogue Act Markup Language (DiAML) as developed for the DialogBank, which is introduced in Section 9. Section 10 discusses the possibilities for customizing the ISO 24617-2 annotation scheme to specific domains and purposes. Section 11, finally discusses the limitations of the current version of ISO 24617-2, and possible revisions in a future version. The document ends with a list of references, an appendix that contains summary definitions of the dimensions, communicative functions, and qualifiers of ISO 24617-2, and an appendix that contains summary definitions of the core set of rhetorical relations specified in the ISO 24617-8 standard for the annotations of discourse relations, which are recommended for use also in dialogue annotation.

1 General issues in DA annotation

1.1 Preliminaries

A dialogue has been defined as “a spoken, typed or written interaction in natural language between two or more agents” (DAMSL Revised Manual, p. 1). The term ‘agent’ in this characterization is intended to cover both human and artificial participants. The ISO 24617-2 standard is intended to apply to dialogues in a wider sense, where the participants not only use natural language but also nonverbal means, such as gestures and facial expressions, in the case of human participants and embodied conversational agents, and means like highlighting, blinking, and beeping in the case of interactive computer systems such as virtual assistants and helpdesks.

The prototypical setting of human dialogue is that of face-to-face communication, where speech is combined with other vocal sounds (laughs, sighs, heavy breathing, etc.), facial expressions, gaze direction, and other physical activities including head-, hand-, arm-, and shoulder gestures, forms of touching (stroking, caressing, hugging, shaking hands, patting on the shoulder, etc.), and body posture changes. All these verbal and nonverbal activities may have a communicative meaning which can be made explicit in terms of dialogue acts. ISO 24617-2 has a general emphasis on its use for creating interoperable *language* resources, but it has been successfully applied also to nonverbal and multimodal behaviours.

1.2 Dialogue settings and participants

Dialogue act annotation schemes have been developed mostly for situations involving two people in a spoken interaction, with or without visual contact, or involving several people in a setting where they see and hear each other. In either type of situation there is much of the time one participant who occupies the speaker role, i.e. “has temporary control of the dialogue and speaks for some period of time” (DAMSL Revised Manual, Preliminaries, p. 1). This participant, the ‘speaker’, speaks either to the single other participant in the case of a two-person dialogue, or to one or more participants in the case of multi-party dialogue. These participants are the addressees of the dialogue acts performed by the speaker.

There are certain formalized interactive situations where the role of addressee does not coincide with the person(s) that the speaker is in fact addressing. For example, in debates in the British House of Commons the person who occupies the speaker role is formally addressing the Speaker of the House, but his words are in fact aimed at a particular representative or cabinet member, or at a group of representatives. Another type of dialogue setting where the role of addressee is not straightforward is that of a televised interview in front of an audience. In this case the interviewee will typically speak as if addressing the interviewer, while his words are in fact intended primarily for the audience in the studio, or for the viewers at home. Communicative functions are defined in ISO 24617-2 as the way in which the speaker intends to affect the information state(s) of the addressee(s), hence in such situations the annotation of the speaker’s utterances should be determined by considering whose information states the speaker is principally trying to influence.

1.3 Annotation purposes and unusual annotation situations

ISO 24617-2 is intended for use by human annotators and by automatic annotation systems. It has proved useful in both situations and (see e.g. Petukhova PhD, 2011; Keizer et al, 2011, Petukhova and Bunt, 2014; Malchanau et al., 2016, Bunt et al., 2016). If the primary aim of an annotation effort is to achieve high accuracy, then the annotators should use all the available sources of information. For a multimodal dialogue, where speech is used in combination with nonverbal behaviour, this means that not only the recorded speech should be available to annotators, but also a video recording of the nonverbal behaviour, or at least an accurate transcription of that behaviour. Similarly, in the case of a dialogue over the telephone, annotators should not only have the transcribed speech at their disposal but also the original sound recording (or at least an accurate transcription of the prosody and the relevant nonlinguistic sounds that occur), for being able to interpret the intonation, speech tempo, and nonlinguistic vocal sounds. One important source of information for annotators,

when deciding on the identification or annotation of a given functional segment, may be the recording of how the dialogue continued *after* the segment under consideration. Therefore, if the purpose is to obtain the most accurate possible annotation, annotators should be allowed to look ahead in the dialogue.

1.4 Explicit and implicit, implied and indirect functions

A functional segment has a communicative function for one of two reasons: 1) by virtue of having certain linguistic or nonverbal features which, in the context in which the segment occurs (especially in view of the preceding dialogue), are indicators of that function; or 2) by implication of having a certain other function. In the first case it is common to say that the segment has that communicative function *explicitly*; in the second case that it has that function *implicitly*. The following example illustrates both cases:

- (1) 1. A: Would you like to have some coffee?
2. B: Some coffee would be great, thanks.

A's utterance may be taken to be an Offer, due to the fact that this linguistic form is conventionally used for that purpose (but since this form is also conventionally used to express a question, there is an ambiguity here); B's response is an Accept Offer by virtue of its linguistic form and the fact that it occurs immediately after an Offer. Since an offer can only be accepted when it has been understood, B's response by implication also has a positive auto-feedback function. (Note that the Accept Offer in this example has a functional dependence relation to the preceding Offer.) This feedback function is implicit and (therefore) implied.

A functional segment that expresses a dialogue act DA2 which has a functional dependence relation to a previous dialogue act DA1 always has an implied positive auto-feedback function relating DA2 to DA1. Such pairs DA1 - DA2 may occur when DA1 is a dialogue act that puts a certain pressure on the addressee to perform an action of the type DA2. Examples of such DA1 dialogue acts are all types of questions, offers, requests, initial greetings, and apologies. The corresponding 'responsive' DA2-acts, i.e. all types of answers, acceptance and rejection of offers and requests, return greetings, and acceptance of apologies, thus have an implied positive auto-feedback function.

More generally, the following types of implicit communicative functions can be distinguished:

1. A communicative function F_2 is *logically entailed* by another function F_1 because F_1 is a special case of F_2 . This happens in hierarchies of communicative functions like the general-purpose functions of the ISO standard, where for instance a Confirm is a special case of an Answer, and a Correction is a special case of a Disagreement, which in turn is a special case of an Inform. So every confirmation is (also) an answer; every correction is (also) a disagreement, and so on.
2. A communicative function F_1 may have another function F_2 as a *conversational implicature*, i.e. if a functional segment has the function F_1 it also has the function F_2 , unless the F_1 -act occurs in situations where there is evidence to the contrary. For example, a thanking act (like *Thank you!*) will normally be understood as also a signal of positive feedback.

Besides implicit positive feedback acts, another class of implicit acts has to do with turn-taking. Every time someone starts speaking, this may be interpreted as the performance of a turn-taking act; every time someone stops speaking this may be interpreted as a turn-release act; and every time a speaker goes on speaking this may be interpreted as a turn-keeping act. These phenomena are discussed below in Section 5.2.

Should implicit communicative functions be annotated? Annotating logically entailed functions would clearly be redundant, since by their very nature such functions can be inferred from explicit functions, and could thus be added automatically afterwards. For conversationally implicated functions the situation is a little different, since these functions can be inferred in most but not in all contexts. The annotation of all implicated positive feedback functions would be very impractical, and the annotation of all implicated turn management functions would even be impossible for implicit turn-keeping functions. Implicated communicative functions should therefore not be annotated either. If the purpose of an annotation campaign is such that it would be important to recognise implicated functions, then this can be done post-hoc by checking whether the dialogue context allows the inference of these functions and add these functions accordingly. For more details about types of implicit functions and strategies for how to deal with them see Bunt (2011).

Indirect speech acts are mostly regarded in standard speech act theory as just another *form* of the same communicative act as the direct form. By contrast, ISO 24617-2 incorporates the view that indirect forms signal subtly different packages of beliefs and intentions than direct ones. For example, the direct request *Tell me what time it is please* carries the assumption that the addressee knows what time it is, whereas the indirect request *Do you know what time it is?*, or *Can you tell me what time it is?*, does not carry that assumption (it does at least not *express* that assumption; in fact it questions it), and is best interpreted as *Please tell me what time it is, if you know/can*.

This example shows that an indirectly formulated request may have a conditional character: the speaker is expressing a request to do something under the condition that the addressee is able to perform the requested action. In this case the annotator may therefore make use of the option to annotate the utterance as having a qualified Request function, with the attribute ‘conditionality’ having the value ‘conditional’. This can be represented in DiAML as follows, where the notation `target="#fs1"` with a hash symbol is used, following the convention for referring to entities that have been defined elsewhere, for instance at another layer of annotation, or in the metadata of a given document (like the sender and the addressee in this example).

```
(2) <dialogueAct xml:id="da1" target="#fs1"
      sender="#s" addressee="#a" dimension="task"
      communicativeFunction="request" conditionality="conditional"/>
```

1.5 General advice for annotators

Dialogue act annotation is about indicating the kind of intention that the speaker had; what was he trying to achieve? When participating in a dialogue, this is what an addressee always

tries to figure out. The following general advice for dialogue act annotators derives from this.

1. **Do as an addressee would do.**

When assigning annotation tags to a dialogue utterance (more precisely, a ‘functional segment’ - see below), put yourself in the position of the participant(s) to whom the utterance was addressed, and imagine that you try to understand what the speaker is trying to achieve. Why does he say what he says? What are the purposes of the utterance? What assumptions does the speaker express about the addressee? Answering such questions should guide you in deciding which annotation tags to assign, regardless of how exactly the speaker has expressed himself. Use all the available information that you would have if you were the actual addressee, and like the addressee, try to understand the speaker’s communicative behaviour. (As mentioned in Section 1.3 depending on the purpose of the annotation, it may also be an option for you to look ahead in the dialogue.)

2. **Think functionally, not formally.**

The linguistic form of an utterance often provides vital clues for choosing an annotation tag, but such clues can also be misleading; in choosing your tags you should of course use the linguistic clues to your advantage, but don’t let them fool you - the true question is not what the speaker says but what he means.

For example, Set Questions are questions where the speaker wants to know which elements of a certain domain have a certain property. In English, such questions often contain a word beginning with ”wh”, such as *which* as in *Which books did you read on your holidays?* or *where* in *Where do your parents live?* In other languages this is not the case. Moreover, in English not all sentences of this form express a Set Question: *Why don’t you go ahead* is for instance typically a suggestion rather than a question.

Similarly, Propositional Questions are questions where the speaker wants to know whether a certain statement is true or false. Such questions are typically expressed by interrogative sentences, such as *Is The Hague the capital of the Netherlands?* or *Do you like peanut butter?* But not all sentences of this form express a Propositional Question; for example, *Do you know what time it is?* is most often used as an indirect way of requesting to tell the time. Similarly, *Would you like some coffee?* is most likely an offer, rather than a question, and *Shall we go?* a suggestion.

3. **Be specific**

Among the communicative functions that you can choose from, there are differences in specificity, corresponding to their relative positions in hierarchical subsystems of the taxonomy. For instance, a Check Question is more specific than a Propositional Question, in that it additionally carries the expectation that the answer will be positive. Similarly, a Confirm act is more specific than an Answer, in that it carries the additional assumption that the addressee expects the answer to be positive.

In general, try to be as specific as you can. But if you’re in doubt about whether to use a more or a less specific function, and you don’t really have evidence for choosing the more specific one, then use the less specific one.

2 Segmentation

According to ISO 24617-2, dialogue acts correspond to *functional segments*, defined as *minimal stretches of communicative behaviour that have a communicative function*, ‘minimal’ in the sense of not including material that does not contribute to the expression of the function or the semantic content of the dialogue act. The requirement of being ‘minimal’ has been added in order to ensure that communicative functions are assigned as accurately as possible to those stretches of behaviour which express the dialogue acts with these functions. Consider the following example (from a Map Task dialogue):

- (3) S: ... then go direction that moon lander, that thing on those legs

This stretch of behaviour could be marked up as expressing an Instruct act (*then go direction that moon lander*) followed by an Inform act (*that thing on those legs*) which explains the term “moon lander” in the instruction. In order to mark this up accurately it is best to segment this stretch into two functional segments, *fs1* = “then go direction that moon lander” and *fs2* = “that thing on those legs”, and to assign the Instruct function to segment *fs1* and the Inform function to segment *fs2*, rather than assigning both functions to the entire utterance.

Note that the more fine-grained segmentation also allows one to indicate the fact that the Inform act in *fs2* is an elaboration of the Instruct act in *fs1*, by marking up a rhetorical relation between the two dialogue acts involving the *Elaboration* relations, as in (4):

- ```
<dialogueAct xml:id="da1" target="#fs1"
 speaker="#s" addressee="#a" dimension="task"
 communicativeFunction="instruct"/>
(4) <dialogueAct xml:id="da2" target="#fs2"
 speaker="#s" addressee="#a"
 communicativeFunction="inform" dimension="task"/>
<rhetoricalLink dact="#da2" rhetoAntecedent="#da1"
 rhetorel="elaboration"/>
```

The minimality of functional segments is also illustrated nicely by hesitations, as in example (5):

- (5) Can you tell me what time the train to *ehm*,... Viareggio leaves?

The speaker interrupts himself while formulating a request for information since he needs a bit of time to choose the name of a destination. The small interrupting segment *ehm* does not contribute to the expression of the request, so according to the minimality condition it does not belong to the functional segment that corresponds to the request. The expression *Can you tell me what time the train to ehm,... Viareggio leaves?* should thus be analysed as consisting of two functional segments: the discontinuous segment *Can you tell me what time the train to [ ] Viareggio leaves?*, corresponding to a request, and the functional segment *ehm* corresponding to a Stalling act. This can be annotated in DiAML as follows:

```

 <dialogueAct xml:id="da1" target="#fs1"
 speaker="#s" addressee="#a" dimension="task"
 communicativeFunction="request" conditionality="conditional"/>
(6) <dialogueAct xml:id="da2" target="#fs2"
 speaker="#s" addressee="#a"
 communicativeFunction="stalling" dimension="timeManagement"/>

```

Note that in this example the yes-no question of the form *Can you tell me...* has been interpreted as a conditional request, i.e. as: *Please tell me, if you're able to,...*

There are cases where the determination of the minimal stretch of behaviour that forms a functional segment is not obvious, in particular when a longer stretch could be said to express a dialogue act with a certain function while it consists of parts that could also be said to express dialogue acts with that same communicative function. Example (7) (from a Map Task dialogue), illustrates this:

- (7) 1. E: and then you go up and around that, a little to the right  
 2. A: slightly northeast?  
 3. E: yeah, slightly northeast

Utterance 3 could be said to constitute a Confirm act, but each of its two parts (*“yeah”* and *“slightly northeast”*) could also be said to each constitute a separate Confirm act. Larson (1998) has recommended to take a ‘maximal’ approach to segmentation and as a rule choose the longest possible stretch of behaviour as the unit of annotation. By contrast, the use of functional segments rather suggests to take a *minimal* approach. Which of these strategies is to be preferred may be determined by the overall purpose of an annotation effort; one consideration is that the minimal approach is more fine-grained.

A functional segment is most often a part of what is contributed by the participant who occupies the speaker role, but it may happen that a dialogue act is spread over multiple turns, as in the following example, where the utterances in turns 6, 8, 11, and 13 together form the functional segment that contains B’s answer to the question in turn 5:

- (8) 1. A: I’ve skied in Colorado, and we usually go to New Mexico because it’s a little cheaper —  
 2. B: Ooh,  
 3. A: — you know  
 4. B: Uh-huh  
 5. B: Where in Colorado?  
 6. A: I’ve been to Telluride, which is on the West side,  
 7. B: Yes  
 8. A: and, uh, Copper  
 9. A: Copper is kind of my favorite up there  
 10. B: Really?  
 11. A: Breckenridge —  
 12. B: Uh-huh  
 13. A: — and Keystone

This example forms a tricky case for segmentation and dialogue act annotation, for although the answer isn’t complete until turn 13, participant B provides intermediate feedback in the

turns 7, 10, and 12, and participant A provides an intermediate assessment of the answer part in turn 8. See also Section 6.2.

### 3 Annotation components

Dialogue annotation according to ISO 24617-2 assumes a three-level architecture, consisting of:

1. a primary source, which may correspond to a speech recording, textual transcription or any further low-level annotation thereof;
2. the marking of functional segments from the primary source (a.k.a. ‘markables’);
3. the dialogue act annotation associated with a functional segment.

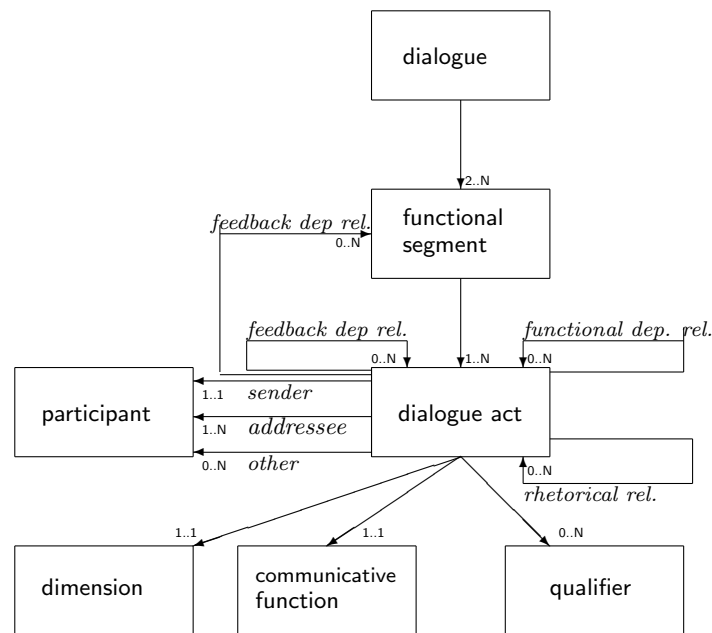


Figure 1: ISO 24617-2 Metamodel

The ISO 24617-2 metamodel, displayed in Figure 1, shows these three levels clearly: the box ‘dialogue’ at the top represents the primary source of level 1; the box ‘functional segment’ contains the markables of level 2, and the rest of the model forms the dialogue act annotations associated with these markables.

The metamodel shows that a dialogue act has a sender, one or more addressees, possibly other participants, a semantic content category (or ‘dimension’), a communicative function,



functional and feedback dependence relations, possibly one or more qualifiers, and possibly one or more rhetorical relations to other dialogue acts. Note that according to the model a feedback dependence relation relates a dialogue act (notably a feedback act) either to another dialogue act or to a functional segment. This point is discussed in further in Section 6.2.

According to the metamodel, the following ingredients must or may occur in an ISO 24617-2 dialogue at annotation:

1. the sender; every dialogue act has exactly one sender who is ‘responsible’ for the act, even though more than one speaker may contribute; see example (9)

- (9) 1. A: and then should I specify the uhm, uhm,  
2. B: budget code

In this example, A is struggling to formulate a question and B helps by providing the term that A was looking for. B’s utterance is itself a dialogue act with the communicative function *Completion*, in the Partner Communication Management dimension. The functional segment *and then should I specify the budget code* made up by parts of what A and B say expresses a question for which A is ‘responsible’ and is considered the sender.

2. the addressee(s); in a two-person dialogue the addressee is just the one who is not the sender; in multiparty dialogues, such as those of the AMI corpus, all the participants who are not the sender are addressees, unless the speaker picks out one of them (in which case the other participants form the ‘other participants’). To keep things simple, the formulations in this document assume just one addressee.
3. any other participants (if any);
4. the communicative function;
5. the dimension;
6. functional dependence relations (if any);
7. feedback dependence relations (if any)
8. qualifiers [if any; optional]
9. rhetorical relations [if any; optional]

For a given functional segment in a dialogue, the sender and addressee roles are usually easy to assign (as is the ‘other participants’ role, in case any such participants are present). For the assignment of other components see Section 4 for dimensions; Section 5 for communicative functions; Section 6 for functional and feedback dependences, and Section 7 for rhetorical relations.

## 4 Annotating dimensions

For assigning dimensions, the decision to be made is which kind of information or action is addressed. Is it

- (1) concerning the underlying task/activity (the **‘Task’** dimension); or
- (2) concerning the speaker’s processing of previous utterances (**Auto-Feedback**); or
- (3) concerning the addressee’s processing of previous utterances (**Allo-Feedback**); or
- (4) concerning the allocation of the speaker role (**Turn Management**); or
- (5) concerning the time needed to continue the dialogue (**Time Management**); or
- (6) concerning the editing of what the speaker is saying (**Own Communication Management**); or
- (7) concerning the editing of what the addressee is saying (**Partner Communication Management**); or
- (8) concerning the structure of the dialogue (**Discourse Structuring**); or
- (9) concerning social obligations (**Social Obligations Management**)?

The formal definitions of these dimensions in the ISO 24617-2 standard (in the form of so-called ‘data categories’) can be found in the document ‘Data categories for dialogue acts’ (Bunt, 2012), available at the DialogBank website: [https://dialogbank.uvt.nl/wp-content/uploads/tdb/2015/12/DiAML-ISO24617-2\\_data\\_categories.pdf](https://dialogbank.uvt.nl/wp-content/uploads/tdb/2015/12/DiAML-ISO24617-2_data_categories.pdf). Summaries of these definitions are provided in Appendix A.

## 5 Annotating communicative functions

### 5.1 General-purpose functions

This section first provides guidelines for the use of general-purpose functions (Section 5.1), subdivided into information-transfer functions and action-discussion functions, followed by guidelines per dimension for the use of dimension-specific functions (Section 5.2).

#### 5.1.1 Information transfer functions

All dialogue acts with an information transfer function have the purpose of making certain information available to the addressee (acts with an Inform function or a function dominated by Inform in the hierarchy shown in Figure 2) or of obtaining certain information from the addressee (the ‘Information-seeking functions’ in Figure 2). The information to be obtained or made available can be of any kind, relating to the underlying task or activity, or relating to the interaction.

In order to decide whether a segment of dialogue has an information transfer function, an annotator should thus decide whether the segment has one of these two purposes. If so,

the annotator can use the subtrees of the Information-providing and Information-seeking functions in Figure 2 as decision trees, going systematically left-right through the functions at the next level down and checking the defining conditions that distinguish each of these functions from their ancestor and from each other. Since the functions at one level in a subtree are mutually exclusive, at most one of them applies. If one is found that applies, then go down one level to the functions dominated by this function, and repeat the process. Keep doing this until hitting a level where none of the functions apply. In that case choose the function that dominates the functions at that level.

### 5.1.2 Action discussion functions

All action discussion functions have in common that their semantic content describes an action, possibly with specifications of manner or frequency of performance. The actions under discussion can be of any kind: actions for moving the underlying task forward, or actions for managing the interaction, or actions for dealing with social obligations.

This class of communicative functions falls apart into the classes of Commissives and Directives, familiar from speech act theory. Commissive acts all have as their common property that the sender expresses a commitment to performing an action, while directive acts are characterised by the sender having the goal that the addressee commits himself to performing an action. In order to decide whether a segment of dialogue has a commissive or a directive function, an annotator should decide whether the segment has the purpose of expressing or trying to impose such a commitment. If so, the annotator can use the subtrees of Commissives and Directives (see Figure 2) as decision trees, in the same way as for choosing an information transfer function.

## 5.2 Dimension-specific functions

Dimension-specific functions are often expressed by means of specific fixed forms and formulaic expressions, more so than general-purpose communicative functions, and can therefore often be recognised by the occurrence of these forms and formulas.

### 5.2.1 Auto- and Allo-Feedback

Feedback acts have the purpose of providing or eliciting information about the processing of utterances in dialogue. Both auto- and allo-feedback providing functions are divided into positive and negative ones. Positive feedback signals that the sender believes that the utterances under consideration were processed successfully; negative feedback that there was some problem or some uncertainty regarding the processing of these utterances. Positive feedback is very often expressed implicitly, and should in such a case most probably not be encoded, as argued above in Section 1.4. Negative feedback is virtually always explicit, and as such easy to recognise. Some of the frequently used fixed forms for negative auto-feedback are *Huh?*, *What?* and equivalent expressions in other languages, as well as nonverbal signals such as raising eyebrows, frowning, or cupping a hand behind an ear.

Repetitions and paraphrases are common forms of auto-feedback. A distinction can be made between the case where the speaker literally repeats (part of) what was said before ('echo') and the case where he paraphrases (part of) what was said. For example:

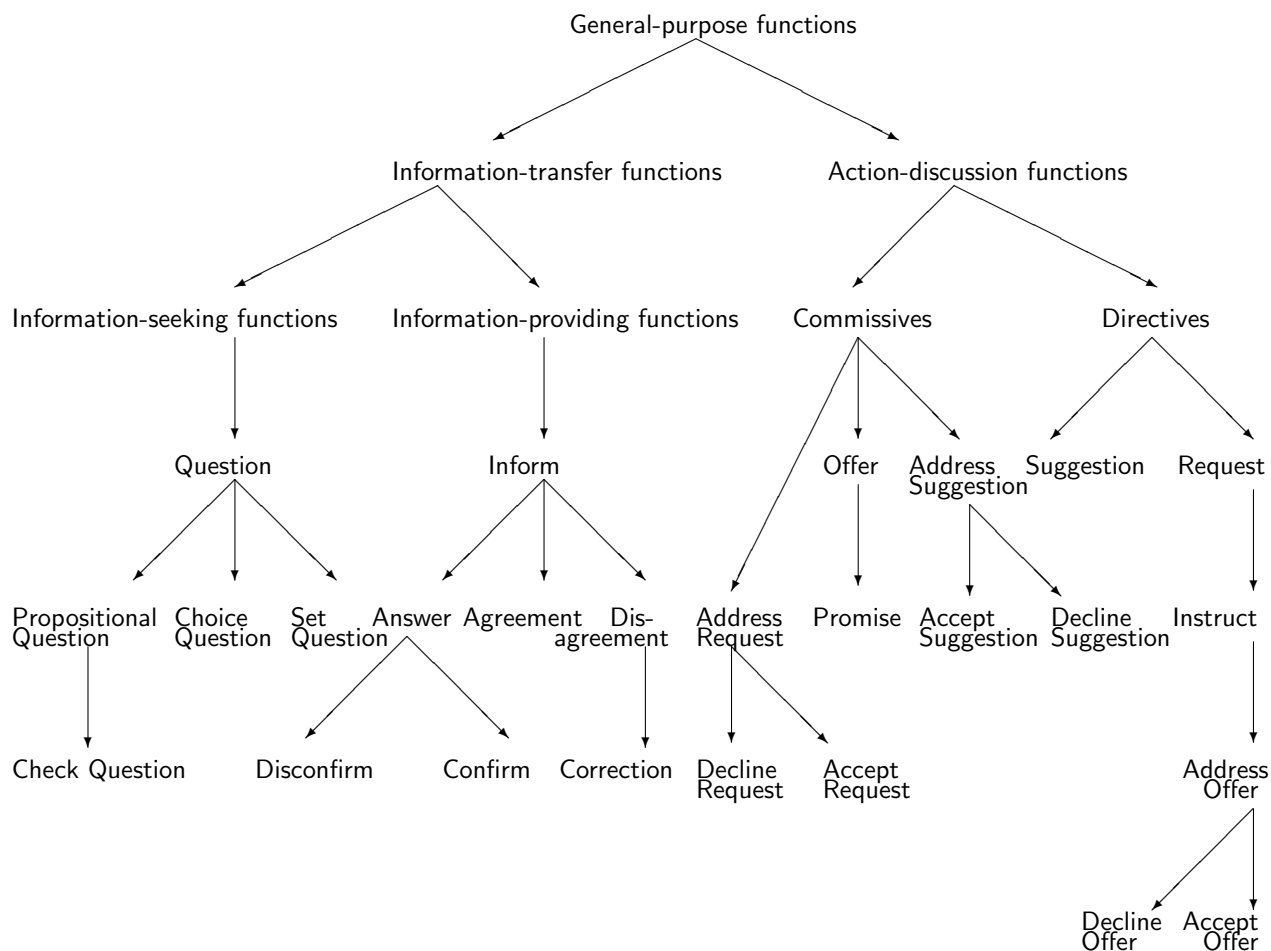


Figure 2: General-purpose functions

- (10) 1. A: I would like to travel next Saturday, in the afternoon.  
2. B: Next Saturday in the afternoon I have a flight leaving at 16:10.  
3. B: On Saturday May 8 after 12 p.m. I have a flight leaving at 16:10.

In (10.2) B literally repeats part of A's question, thereby displaying what he perceived what A said. In (10.3), by contrast, B paraphrases parts of A's question, and this can be taken to indicate not only what B heard but also how B interpreted what A said (which in this example may be particularly relevant for the interpretation of '*next Saturday*').

The most common form of explicit positive feedback is by means of fixed forms like *OK*, *Yes*, *Sure*, etc. which may be taken to express overall successful processing of what was said.

While positive feedback is very often implicit, negative feedback is mostly explicit and expressed by means of general-purpose communicative functions, as in the following examples:

- (11) 1. Did you say Tuesday?  
2. I didn't say Thursday.  
3. This Saturday I mean.  
4. What did you say?

It may be noted that there is a systematic relation between auto- and allo-feedback acts. This is for the following reason. A dialogue act in the Auto-Feedback dimension is concerned with the sender's processing of a previous utterance, e.g.:

- (12) A: Twelve-thirty?  
B: Uh-huh

In his response, B is talking about the addressee's (A's) processing, hence this is an act in the Allo-Feedback dimension.

The reverse is also true. Consider the following example:

- (13) A: No, Tuesday  
B: Ah, I thought you said Thursday

Speaker A in his utterance corrects a misunderstanding on B's part, hence this is a dialogue act in the Allo-Feedback dimension. In his response, B is talking about his own processing of a previous utterance, hence the response is an act in that participant's Auto-Feedback dimension. This is more generally the case: the response to an Allo-Feedback act is usually an Auto-Feedback act, and conversely.

### 5.2.2 Turn Management

Turn management functions are characterised by the sender having the goal to obtain, to keep, or to hand over the speaker role. For an annotator, the issue to decide on is thus whether the sender's behaviour expresses such a goal. Consider, for example, the case of a question-answer pair:

- (14) 1. A: Do you know what time it is?  
2. B: It's nearly twelve fifteen.

Does B, in answering A's question, have the goal to occupy the speaker role? Clearly, B's primary aim is to answer A's question, and in order to do so he must take the speaker role. This suggests that B did not have a separate goal to have the speaker role. It therefore seems best to characterize B's behaviour as involving an *implied* turn-taking act, which as such does not need to be annotated. to not assign a separate turn-taking function to B's utterance. A guideline for annotators is thus to not annotate a turn-taking function to a functional segment that is used to answer a question. The same goes for other responsive dialogue acts, such as those that respond to a request, an offer, a suggestion, or an apology.

Similarly, does A, by asking a question, express that he wants B to occupy the speaker role next? The answer is *No*, since A can continue for a while occupying the speaker role after asking the question, as in the following example:

1. A: Do you know what time it is? I need to catch the twelve seventeen train.  
(15) Oh dear it's already too late, I see.
2. B: Yes, it's twelve fifteen now.

This raises the question whether continuing to speak indicates the speaker's goal to keep the turn, and a turn-keeping function should thus be assigned to nearly everything that a speaker says. This would be very impractical, if not downright impossible.

For turn-taking and turn-keeping functions, and more generally for all turn-management functions, it is recommended to only assign such a function to those stretches of communicative behaviour where the sender explicitly indicates his goal to obtain, to keep, or to get rid of the speaker role. Just starting to speak, continuing to speak, or ceasing to speak should not be annotated as expressions of Turn Management functions.

### 5.2.3 Time Management

Time management functions are concerned with the sender buying some time. ISO 24617-2 distinguishes two cases:

1. the speaker is unable to say immediately what he intended to say (Stalling);
2. the speaker suspends the dialogue for a while (Pausing).

In both cases there may be several reasons why the sender wants to buy some time. In the first case this is most probably because he is looking for the right words to express what he wants to convey or that he is gathering (or calculating) the information that he wants to convey. In the second case this may be because he is aware that collecting/computing the relevant information requires more time than is reasonable to take in on-line communication, or it may be that something more urgent came up, which momentarily takes priority over continuing to contribute to the dialogue.

Stalling acts often take the form of filled pauses (*ehm, let me see, well,..*), together with slowing down and short silences. Pausing acts explicitly claim or request some time: *Just a minute, Wait a second, I'll be right back, etc..* Fully explicit requests like *Please wait while I check the current status* should not be marked as Pausing acts, but as requests in the Time Management dimension, using the general-purpose function **Request**.

#### 5.2.4 Own and Partner Communication Management

In Own Communication Management (OCM) acts the speaker is editing his own speech. The speaker interrupts himself, being aware that he said something wrong, and retracts something that he just said (*Oh sorry no,...; Or no wait,..*) or replaces something he just said by something else (*I want to travel on Tuesday Thúrursday*).

Partner Communication Management (PCM) acts similarly edit the speech of another participants, who at that moment occupies the speaker role. Two important cases are the correction of the addressee/current speaker (Correct Misspeaking), used to correct what is perceived as a slip of the tongue, and the completion of what the addressee/current speaker is struggling to say (Completion). In both cases the sender of the PCM act barges in and grabs the turn, or takes the turn which has become available because the addressee/current speaker is hesitating.

OCM acts and PCM acts have in common with feedback acts that they refer back to something that was said earlier in the dialogue. They thus have a similar kind of dependence on an ‘antecedent’ as feedback acts. The feedback dependence relation is therefore used not only with feedback acts but also with OCM and PCM acts.

#### 5.2.5 Discourse Structuring

Discourse Structuring acts are concerned with the explicit structuring of the dialogue. Such acts occur frequently at the beginning and near the end of a dialogue. A dialogue needs to be opened in some way, the communicative function Opening is appropriate for labelling a functional segment with that purpose. There are conventional ways of opening a dialogue; in multi-party dialogue an expression that is frequently used to open the dialogue is *Okay!* The same utterance is often used (though with a different intonation) to indicate that a dialogue can be closed, signalling positive feedback concerning the entire preceding dialogue.

There do not seem to exist dialogue acts that have no other function than closing a dialogue; conventionally, a dialogue is considered closed when the participants have exchanged farewell greetings.

Utterances that signal the closing or opening of a new topic, that indicate the speaker’s intention to come to a closure, or other structure-indicating purposes should be assigned the communicative function Interaction Structuring, since the ISAO 24617-2 annotation scheme does not include more fine-grained functions for discourse structuring.

#### 5.2.6 Social Obligations Management (SOM)

The kind of social obligations that should be annotated depends on the kind of dialogue. Welcome and farewell greetings that play a role in starting and ending a dialogue are domain-independent, however, as are apologies for misunderstanding, acts for introducing oneself, and thanking acts and their acceptances. All of these acts have conventional forms (‘formulas’) in every language. They tend to come in pairs: an initial greeting puts pressure on the addressee to send a response greeting; introducing oneself puts pressure on the addressee to also introduce him- or herself; an apology puts pressure on the addressee to accept the apology; a thanking puts pressure on the addressee to downplay what he is thanked for (like in *It was nothing; It was my pleasure*); and a farewell greeting puts pressure on the addressee to produce a response farewell greeting.

SOM acts can also be performed by means of general-purpose functions. For instance, *I'm extremely grateful for your help* and *I hope to see you next year in Hong Kong* are Informs in the SOM dimension, that have the same effect as thanking and saying goodbye, respectively.

### 5.3 Dialogue act qualifiers

Dialogue act qualifiers are available in DiAML for encoding various ways in which a speaker can specify certain conditions, qualifications, or feelings accompanying a dialogue act. For the encoding of certainty and conditionality DiAML has binary-valued attributes, one of which is the default value. For the encoding of feelings the `sentiment` attribute is available which has an open class of values and no default value; if no value of the attribute is specified in an annotation this means that no such information is present.

**a. Certainty** The sender of a dialogue act can express certainty or uncertainty about the correctness of the information provided in an information-providing act, or about his commitment to perform an action in a commissive act. This is illustrated in (16) for information-providing acts, where the expressions *"I have a hunch that"*, *"probably"*, *"might"*, and *"I'm not sure if"* are indicators of the speaker's uncertainty. When these expressions are omitted, as in (17), the resulting sentences no longer contain any suggestion that the speaker is uncertain about the correctness of what he says. This indicates that the default value, corresponding to the unmarked case, is *certain*.

- (16) 1. A: Do you know who'll be coming tonight?  
2a. B: I have a hunch that Mary won't come.  
2b. B: Peter, Alice, and Bert will probably come.  
2c. B: I heard that Tom and Anne might come.  
2d. B: I'm not sure if Bill will come.

- (17) 1. A: Do you know who'll be coming tonight?  
2a. B: Mary won't come.  
2b. B: Peter, Alice, and Bert will come.  
2c. B: I heard that Tom and Anne [will] come.  
2d. B: Bill will come.

Speakers may also signal being *quite certain*, as exemplified in (18). For such cases, the DiAML encoding with `certainty="certain"` is recommended,

- (18) 1. Mary will definitely not come.  
2. Peter, Alice, and Bert will come for sure.  
3. I certainly agree with that.

Certainty and the lack thereof are not only indicated by verbal expressions, but also by prosody, by gaze direction, and by several types of gestures. Prominent nonverbal expressions of uncertainty include gaze aversion, head waggles, lip pouting, lowering eyebrows, and self-touching - and combinations of these.



Warning: verbal expressions of uncertainty, in particular adverbs, should sometimes be interpreted as part of the semantic content of a dialogue act, rather than as a dialogue act qualifier. The following examples illustrate this:

- (19) 1. I'll probably come around eight o'clock.  
 2. I'll definitely come before nine.

In these examples, *probably* and *definitively* apply to the time that is mentioned, not to the sender's certainty about his commitment to come.

For deciding whether to use a certainty qualifier in the annotation of a functional segment, the decision tree shown in Figure 3 can be used.

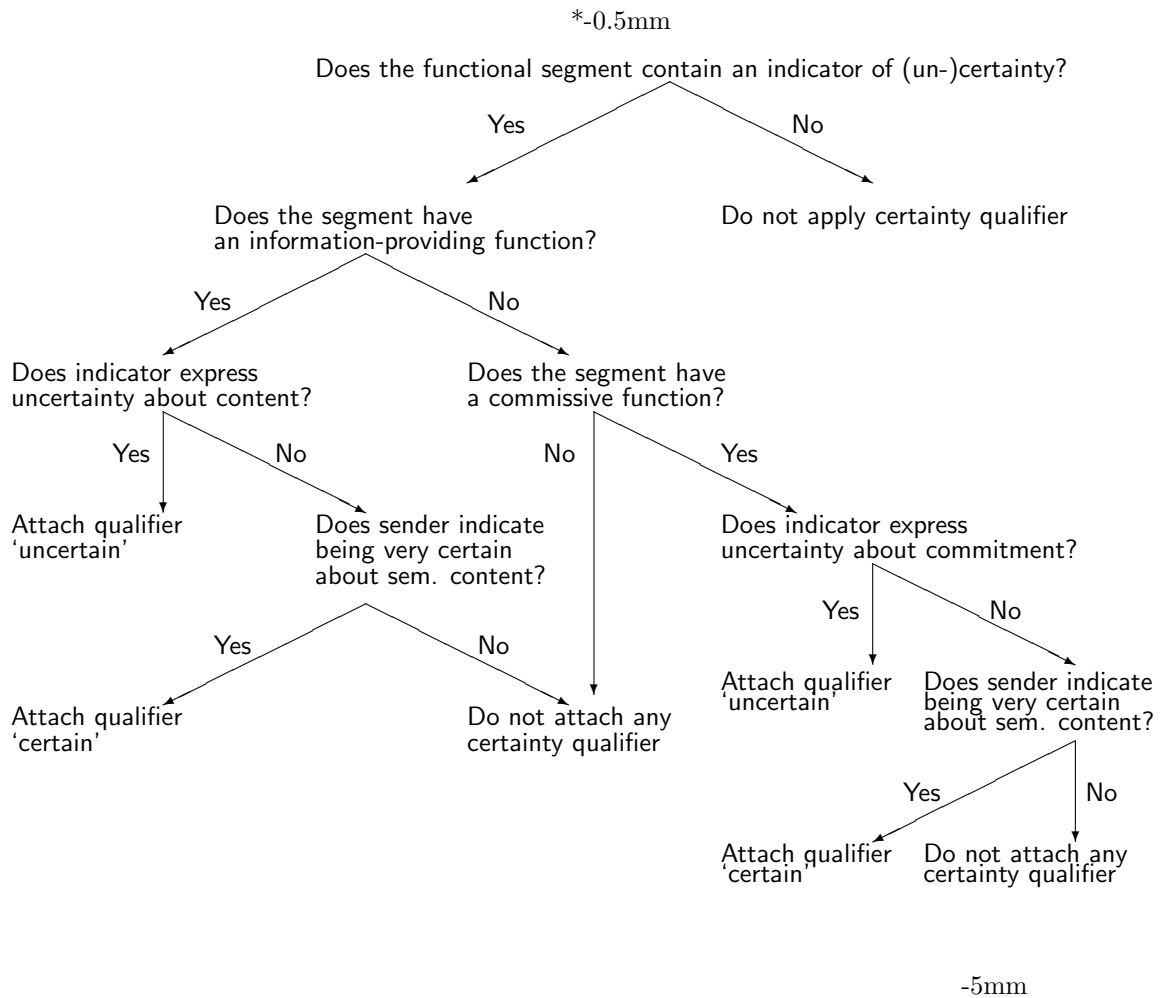


Figure 3: Decision tree for applying certainty qualifiers

**b. Conditionality** Conditionality refers to the possibility (with respect to ability and power), the necessity, or the willingness to perform an action; the qualifiers *conditional* and *unconditional* can therefore be attached to action-discussion functions. The following examples illustrate this phenomenon.

- (20) a. A: Would you like to have some coffee?  
B: Thanks, only if you have it ready.
- b. A: Can you to the presentation, if you're ready?  
B: I can do that if you like.
- c. A: I'll send you an email if you give me your address.
- d. A: Can we just go over that again?  
B: Just very quickly. I have to hurry you on here.  
C: I don't think we have time for that, unless you make it very short.
- e. A: I can make the buttons larger.  
B: No, only if we want basic things to be visible.

In (20a) we see the conditional acceptance of an offer; in (20b) a conditional request, with a conditional acceptance; in (20c) a conditional promise; in (20d) a two conditional acceptances of a request; and in (20e) a conditional rejection of a suggestion.

Similar to the case of certainty qualifiers, omission of the expressions indicating a condition leads to expressions that signal unconditional dialogue acts, hence the default value is *unconditional*, and does not need to be marked up.

Conditional dialogue acts can often be recognised by the use of conditional expressions such as *if ...* or *unless*, and *just* (as in (20d), first case) but just like in the case of certainty, these expressions can also be part of the semantic content rather than qualifiers. For deciding whether to add a conditionality qualifier to the annotation of a dialogue act, the decision tree can be used that is displayed in Figure 4.

**c. Sentiment** A particular sentiment associated with the performance of a dialogue act may be annotated if the sender indicates a certain emotion or attitude concerning the semantic content or the addressee (verbally or nonverbally, or both). Nonverbal expressions of sentiment exist in abundance and in great variety, including for instance smiling (happiness), eyebrow raising (surprise), pressing lips together (angst), and sighing (sadness). Guidelines for sentiment annotation cannot be given here, since ISO 24617-2 does not specify a particular set of sentiment qualifiers.

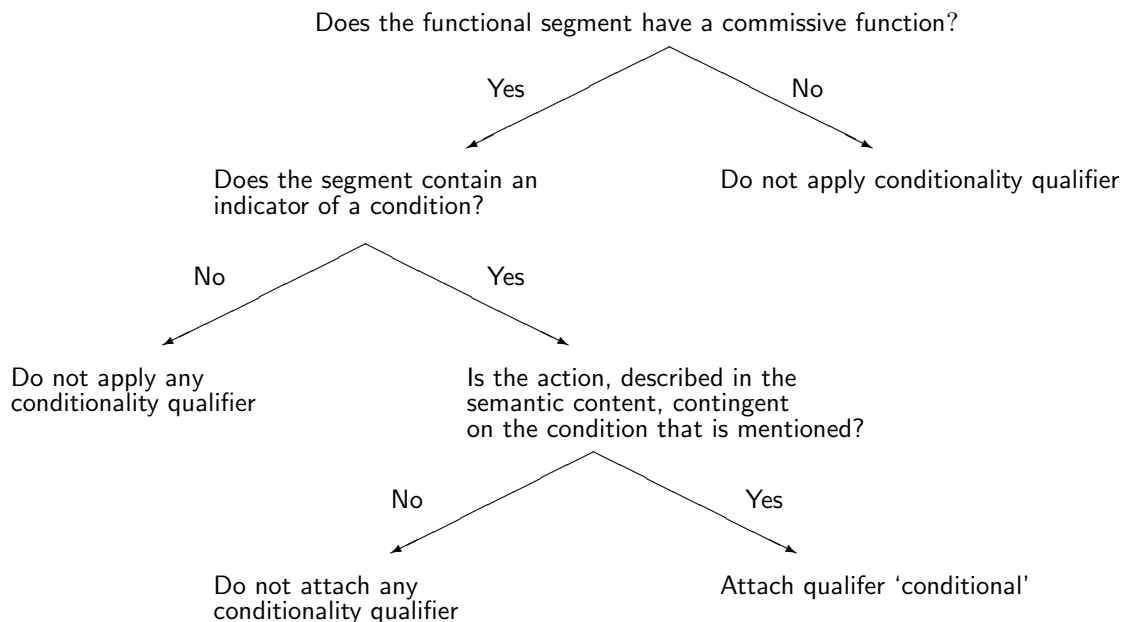


Figure 4: Decision tree for applying conditionality qualifiers

## 6 Functional and feedback dependences

### 6.1 Functional dependence

A dialogue act A1 is functionally dependent on a previous dialogue act A2 (its ‘functional antecedent’), if its communicative function by its very nature responds to another dialogue act, such as an answer being dependent on a question that it answers. This is the case for the following communicative functions defined in ISO 24617-2:

- (21) - Answer, Confirm, Disconfirm;
- Agreement, Disagreement, Correction;
- Address Request, Accept Request, Decline Request;
- Address Suggestion, Accept Suggestion, Decline Suggestion;
- Address Offer, Accept Offer, Decline Offer;
- Turn Accept;
- Return Greeting, Return Self-introduction, Accept Apology, Accept Thanking, Return Goodbye

To encode a functional dependence relation, one has to identify the functional antecedent and link the two dialogue acts. The identification of a functional antecedent is not straightforward if the current dialogue act does not respond to a single dialogue act but to a combination of dialogue acts, as in (22).

- (22) 1. U: Can you tell me what time there are trains from Harwich to York?  
 2. S: What day would you like to travel?  
 3. U: Tomorrow morning.  
 4. U: On Tuesday morning there are trains at 6:45, 70:30,...(etc.)

In (22), utterance 4 forms a functional segment with function Answer, which responds to the question formed by the dialogue acts expressed by utterances 1 and 3 together. In such a case it is recommended to mark functional dependence relations to both these dialogue acts, as in the following example:

(23) <dialogueAct xml:id="da4" ... functionalDependence="#da1 #da3/>

## 6.2 Feedback dependence

Feedback acts are about the processing of something that was said before; this ‘something’ is their semantic content. The nature of this ‘something’ depends on the kind of feedback. Feedback by means of expressions like *OK*, *Uh-huh*, or *Really?* says something about a previous dialogue act, while feedback by means of *Tuesday?* or *THIS Saturday?* is about a particular word or utterance segment. The ISO 24617-2 annotation scheme therefore allows two types of antecedents for feedback dependence relations: dialogue acts and dialogue segments. This is represented in the metamodel shown in Fig. 1 by the two arrows labelled ‘feedback dep. rel.’.

The ISO scheme is in fact not quite correct at this point, since segment-related feedback dependences are not necessarily related to a *functional* segment; they may relate to *any* previous segment, functional or not, such as a single word or a sequence of words within a functional segment. If the latter case arises, the best thing to do according to the ISO standard is to mark up a feedback dependence relation to the functional segment containing the expression that the feedback act refers to. The same goes for marking up the ‘feedback antecedents’ of speech editing acts in the Partner Communication Management (PCM) dimension.

The DBOX dialogues in the DialogBank deviate in this respect from the ISO standard, since for feedback dependences special non-functional segments have been introduced. In a future revision, the ISO standard should include this possibility to allow more accurate markup of feedback dependences, notably also for speech editing acts in the Own Communication Management (OCM) dimension.

## 6.3 Exclusiveness of dependence relations

Functional dependence relations exist only for dialogue acts with a responsive communicative function. such as answer to a question. Feedback dependence relations, on the other hand, exist only for dialogue acts in one of the two feedback dimensions. This leaves the question what to do in case of a responsive dialogue act in a feedback dimension, as in the following example:

- (24) B: (... Tuesday ...) so I would like to travel in the evening, but not too late.  
 A: On Tuesday, you said?  
 B: That’s right.

Participant A uses a Check Question in the Auto-Feedback dimension to make sure that he correctly understood what B said. B’s response is a Confirm act in his Allo-Feedback dimension. Being a responsive dialogue act, B’s Confirm act has a functional dependence relation with A’s Check Question. Being a feedback act, it could be said to also have a feedback dependence relation with that Check Question. This would be redundant to annotate, however, since a responsive act by implication *always* has an entailed feedback dependence relation with the dialogue act that it responds to. (For instance, answering a question entails that the speaker has understood the question he answers.) As it is in general not recommended to annotate entailed dialogue acts, no feedback dependence relation should be marked up in such cases.

Note that A’s Check Question in (24) does have a feedback dependence relation with B’s dialogue act in which “Tuesday” occurred. When occurring in other dimensions (other than Auto-Feedback, Allo-Feedback, OCM and PCM), notably in the Task dimension, questions do not have a feedback dependence relation.

## 7 Rhetorical relations

Many of the relations which may occur between units in a written text, such as Cause, Contrast, or Elaboration, which in the linguistic literature are called ‘rhetorical relations’ or ‘discourse relations’, may also occur between dialogue acts. So-called ‘discourse connectives’; like *also*, *but*, *because*, *for example*, *so* may signal such relations as Elaboration, Contrast, Cause, Exemplification, and Expansion, but these relations may also be implicit. ISO 24617-2 does not require the marking up of rhetorical relations, and does not specify any particular set of rhetorical relations that could be used; it only specifies *how* a rhetorical relation may be marked up as relating two dialogue acts.

Recently, ISO standard 24617-8 has been established for the annotation of rhetorical relations in discourse. This standard, called “ISO DR-core”, defines a set of 18 ‘core’ relations that are commonly used in the annotation of discourse relations. It is recommended to use this set of relations also for marking up rhetorical relations between dialogue acts; see Appendix B for more information.

The rhetorical relations defined in ISO 24617-8 are all assumed to have two arguments, an assumption which is very common in studies of discourse structure. For example, a Cause relation has two arguments, one called “Reason” and one called “Result”. In the annotation scheme of ISO DR-Core one should mark up both relations and argument roles, as shown in (25) in the DiAML-XML format.

- (25) a. John pushed Tim. He fell on the ground.  
`<drArg xml:id="a1" target="#s1" type="event"`  
`<dRel xml:id="r1" rel="cause"`  
`<drArg xml:id="a2" target="#s2" type="event"`  
`<drLink rel="r1" reason="#a1" result="#a2"`
- b. Tim fell on the ground, because John pushed him.

```

<drArg xml:id="a1" target="#s1" type="event"
<dRel xml:id="r1" target="#s2" connective="because" rel="cause"
<drArg xml:id="a2" target="#s3" type="event"
<drLink rel="r1" reason="#a2" result="#a1"

```

For marking up rhetorical relations between dialogue acts, ISO 24617-2 provides just a single slot for specifying a relation, as illustrated in (26)

- (26) A: Have you seen Pete today?  
 B: He didn't come in; he has the flu.

```

<dialogueAct xml:id="da1" target="#fs1" sender="#a"
 addressee="#b" dimension="task"
 communicativeFunction="propositionalQuestion" />
<dialogueAct xml:id="da2" target="#fs2" sender="#b"
 addressee="#a" dimension="task"
 communicativeFunction="answer"
 functionalDependence="#da1">
<dialogueAct xml:id="da3" target="#fs3" sender="#b"
 addressee="#a" dimension="task"
 communicativeFunction="inform" >
<rhetoricalLink dact="#da3" rhetoAntecedent="#da2" rhetoRel="cause"/>

```

In some of the annotations in the DialogBank this limitation has been addressed by marking up a relation plus an argument role in strings of the form ‘Cause:Reason da2’, which is intended to mean that the current dialogue act is the reason in a causal relation with dialogue act ‘da2’.

## 8 DiAML Representation Formats

The definition of the Dialogue Act Markup Language DiAML, which forms part of the ISO 24617-2 standard, observes the distinction between *annotations* and *representations* that is made in the ISO Linguistic Annotation Framework (LAF, see Ide & Romary, 2004). According to this distinction, the term ‘annotation’ refers to the linguistic information that is added to a stretch of primary data (such as an audio-video recording or a speech transcription), independent of the form in which it is represented. The term ‘representation’, by contrast, refers to the format in which an annotation is cast. ISO standard 24617-6, Principles of Semantic Annotation (see also Bunt, 20??) specifies a methodology for defining annotation schemes that implements this distinction through the introduction of an *abstract syntax* and a *concrete syntax*. The abstract syntax of a markup language specifies the structure of annotations as pairs, triples, and other set-theoretic constructs, called ‘annotation structures’; a concrete syntax specifies a way of representing these structures. The ISO 24617-2 document includes a pivot representation of DiAML-annotation structures in XML, and allows other representation formats provided that these satisfy the requirements of being *complete*, i.e. every annotation structure can be represented in this format, and *unambiguous*, i.e. every expression in this format is the encoding of exactly one annotation structure. Bunt (2010)

shows that any two representation formats with these properties are equivalent, and allow a meaning-preserving conversion from one format to the other.

DiAML annotations in XML (using the 'DiAML-XML' format), can be produced with the ANVIL annotation tool, and are perfect for machine processing, but other formats are more convenient for human inspection, correction or modification. For this reason, two tabular representation formats have been defined and used in the DialogBank, called DiAML-TabSW and DiAML-MultiTab; Wijnhoven (2016) has shown that these formats are complete and unambiguous, and has constructed a conversion program for conversions between the three formats.

## 9 Creating your own annotations

Since you can use each of the three equivalent forms, DiAML-XML, DiAML-TabSW and DiAML-MultiTab to represent ISO 24617-2 annotations, you have several choices for how to create your own annotations.

One possibility that is often used is to choose a particular annotation tool, such as ANVIL, and create annotations in the DiAML-XML format. This possibility is particularly appealing when you have audio-video registrations of the dialogues that you want to annotate, since ANVIL requires the use of an AV file. This is on the other hand a slight drawback if you only have an audio registration or a transcription, in which case you have to make a dummy AV file. For information about the use of ANVIL see Bunt et al. (2012), Petukhova et al. (2012), and the ANVIL website ([www.anvil-software.de](http://www.anvil-software.de))

The tabular formats DiAML-TabSW and DiAML-MultiTab have been shown by Wijnhoven (2016) to be more convenient for human inspection and correction than the XML-based format, therefore an attractive alternative is to create annotations in one of these formats, of which DiAML-TabSW seems the most convenient one. This can be done by creating a table in Excel, with a tokenization file and a segmentation file as simple text files, using the templates that are available in the DialogBank.

## 10 Customizing the ISO 24617-2 annotation scheme

The ISO 24617-2 annotation scheme is intended to be domain-independent, in the sense that the dimensions and communicative functions defined in the standard are applicable to any kind of dialogues about any task domain. It may be noted that the standard therefore does not define any communicative functions that are specific for a particular task domain; the Task dimension is in fact the only dimension for which no dimension-specific functions are defined. The standard is however 'open' in the sense of allowing extension with additional concepts, provided that the fundamental principles of ISO 24617-2 are respected. Similarly, for a specific annotation project not all dimensions are necessarily of interest, and one may prefer to use a restricted version of ISO annotation scheme. The official ISO 24617-2 document contains guidelines for customizing the standard to specific needs, which are summarized in the rest of this section.

## 10.1 Principles underlying ISO 24617-2

The most important design principles underlying the ISO 24617-2 standard, which should be taken into account when defining extensions or restrictions of the annotation scheme, are the following:

1. Dialogue act annotation associates communicative functions and other information to functional segments, which may be discontinuous, overlap, spread over multiple turns, and include parts contributed by different participants. A functional segment may have more than one communicative function.
2. Dimensions are defined as distinct types of communicative activity, each of which is concerned with a particular category of information (such as the processing of utterances; the allocation of participant roles; the task; and social obligations). Dimensions can therefore also be viewed as categories of semantic content.
3. Dimensions are independent or ‘orthogonal’ in the sense that a functional segment can have a communicative function in one dimension independent of the functions that it may have in other dimensions.
4. The set of communicative functions is divided into (a) sets of dimension-specific functions, one for each dimension, and (b) a set of general-purpose functions which can be applied to any sort of information and form a dialogue act in any of the dimensions. A dimension-specific communicative function, by contrast, can only be combined with semantic content of the category that is characteristic for that dimension.
5. The set of general-purpose communicative functions is semantically ‘connected’ in the sense that any two functions are either mutually exclusive alternatives or one is a specialisation of the other. This is reflected in Figure 2 (page 12), where any two functions either have a dominance relation or are alternatives with a common ancestor. For each dimension, the set of dimension-specific communicative functions for that dimension is also semantically connected.
6. The semantic connectedness of the communicative functions that can be used in any given dimension has the advantage that a functional segment never needs to be annotated with more than one function per dimension, assuming that for each dimension in which the segment has a communicative function, the most specific function is chosen for which there is sufficient evidence. Given the orthogonality of the dimensions, this has the consequence that a functional segment is annotated with maximally as many functions as there are dimensions.

## 10.2 Schema extension

The design of the ISO 24617-2 annotation scheme makes it easily extensible in the following ways:

- a. Addition of dimensions: Dimensions can freely be added as long as the above principles 2 and 3 remain intact. A property that is particularly important for an additional dimension is that of being orthogonal to the dimensions already present, in order to avoid



redundancy and ambiguity in annotations. For example, Contact Management, which is one of the dimensions of the DIT<sup>++</sup> annotation scheme, was noted by Petukhova and Bunt (2009a; b) to be a possible additional dimension, being orthogonal to the nine dimensions of this standard, being theoretically justified, empirically observed, and recognizable with acceptable precision by human annotators and by automatic annotation programs.

- b. Addition of communicative functions: Communicative functions may be added to the functions defined in this standard, provided that the above principles 4 and 5 remain intact. The DIT<sup>++</sup> and LIRICS taxonomies contain several examples of communicative functions that satisfy these requirements and that could be added to the ISO 24617-2 functions, in particular in the Auto- and Allo-Feedback dimensions and in the Discourse structuring dimension.
- c. Addition of dialogue act qualifiers: For the sentiment qualifier attribute, values may freely be introduced. Additional qualifier attributes and values may be introduced provided that they leave the set of these attributes ‘orthogonal’, in the sense of dealing with non-overlapping aspects of qualification, and for each attribute the set of values should preferably be ‘semantically connected’ in order to ensure that a uniquely determined most specific value can always be chosen for the attribute.
- d. Specification of rhetorical relations: Rhetorical relations may freely be added to the ISO 24617-2 annotation scheme. Use of the rhetorical relations defined in ISO 24617-8 is recommended but may well be extended, and other sets of relations can also freely be introduced.

### 10.3 Schema restriction

While it may happen that some task domains or annotation purposes may require extensions of the ISO 24617-2 scheme, it may also happen that for a certain annotation purpose not every dimension is considered relevant. Subschemes of the ISO 24617-2 annotation standard scheme can be defined relatively easily, by leaving out certain ingredients in the following ways.

A dimension and the corresponding set of dimension-specific communicative functions may be freely omitted; by virtue of the orthogonality of the set of dimensions. Whether a particular dimension is included or not has no influence on the remaining dimensions.

Communicative functions for which there is a less specific function present in the annotation scheme may freely be omitted, since in that case the remaining set of communicative functions is still semantically connected.

It is not recommended to omit a communicative function for which the scheme contains more specific functions while maintaining the more specific functions, since this limits the possibilities for an annotator to use a less specific functional tag in the case of lack of evidence for a more specific one.

Dialogue act qualifiers may freely be omitted. For qualifier attributes for which a default value is defined (such as certainty and conditionality), omitting a value is semantically equivalent to using the default value; for qualifier attributes for which no default value is defined (such as sentiment), omitting a value is equivalent to leaving that aspect underspecified.

## References

- [1] Bunt, Harry (2015) On the principles of semantic annotation. In *Proceedings of ISA-11*, London, pp. 1-13.
- [2] Bunt, Harry (2011) Multifunctionality in dialogue. *Computer Speech and Language* 25: 222-245.
- [3] Bunt, Harry, Volha Petukhova, David Traum, and Jan Alexandersson (2016) Dialogue Act Annotation with the ISO 24617-2 Standard. In *Multimodal Interaction with W3C Standards: Towards Natural User Interfaces to Everything*, Deborah Dahl, editor. Springer, Berlin, pp. 109-135.
- [4] Bunt, Harry, Kipp, Michael, and Petukhova, Volha (2012). Using DiAML and ANVIL for multimodal dialogue annotation, in *Proceedings of the 8th International Conference on Language Resources and Systems (LREC 2012)*, Istanbul. Paris: ELRA, 1301?1308.
- [5] Bunt, Harry and Rashmi Prasad (2016) ISO DR-Core (ISO 24617-8): Core Concepts for the Annotation of Discourse Relations. In *Proceedings 12th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-12)*, Portoroz, Slovenia. ELDA, Paris, pp. 45-54.
- [6] Ide, Nancy and Romary, Laurent (2004) International Standard for a Linguistic Annotation Framework. *Natural Language Engineering* 10: 221-225.
- [7] Keizer, Simon, Volha Petukhova and Harry Bunt (2011) Multidimensional Dialogue Management. In *Interactive Multimodal Question Answering*, Antal van den Bosch and Gosse Bouma, editors. Springer, Berlin, pp. 57-86.
- [8] Kipp, Michael (2001) Anvil - A Generic Annotation Tool for Multimodal Dialogue. *Proceedings Eurospeech 2001*, 1367-1370.
- [9] Larsson, Staffan (1998) Coding Schemes for Dialogue Moves. Technical Report from the S-DIME Project, University of Gothenburg.
- [10] Malchanau, Andrei, Volha Petukhova, Harry Bunt, and Dietrich Klakow (2016) Multidimensional dialogue management for tutoring systems In *Proceedings 7th Language and Technology Conference (LTC 2015)*, Poznan.
- [11] Petukhova, Volha and Harry Bunt (2014) In *Computing Meaning, Vol. 4*, Harry Bunt, Johan Bos and Stephen Pulman, editors. Springer, Berlin, pp. 235-256.
- [12] Petukhova, Volha , and Harry Bunt (2012) The coding and annotation of multimodal dialogue acts. . In *Proceedings 8th International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul. ELDA, Paris.

- [13] Petukhova, Volha and Harry Bunt (2010) Towards an integrated scheme for semantic annotation of multimodal of multimodal dialogue data. In *Proceedings 7th International Conference on Language Resources and Evaluation (LREC 2010)*, Malta. ELDA, Paris, pp. 2556-2563.

## Appendix A. Summary definitions of dimensions, communicative functions, and qualifiers

For more details about the concepts defined in this appendix see the official ISO 24617-2 document, the unpublished document ‘Data categories for dialogue acts’ (Bunt, 2012), available in the DialogBank (see Annotation Schemes/ISO 24617-2), or the website <http://dit.uvt.nl>.

### 10.4 Dimensions

**Task:** Category of dialogue acts whose performance contributes to pursuing the task or activity that motivates the dialogue.

**Auto-Feedback:** Category of dialogue acts where the sender discusses or reports on his processing of previous dialogue contributions.

**Allo-Feedback:** Category of dialogue acts where the sender discusses the addressee’s processing of previous dialogue contributions.

**Turn Management:** Category of dialogue acts whose performance is meant to regulate the allocation of the speaker role.

**Time Management:** Category of dialogue acts which concern the allocation of time to the participant occupying the speaker role.

**Own Communication Management:** Category of dialogue acts where the speaker edits his own speech within the current turn.

**Partner Communication Management:** Category of dialogue acts which are performed by a dialogue participant who does not have the speaker role, and edits the speech of the participant who does occupy that role.

**Discourse Structuring:** Category of dialogue acts which explicitly structure the interaction.

**Social Obligations Management:** Category of dialogue acts performed for taking care of social obligations such as greeting, thanking, and apologizing.

### 10.5 Communicative functions

Note: the definitions of communicative functions in this annex are formulated in terms of intended effects on the information state of a single addressee. For a dialogue act which has multiple addressees, it is understood that the intended effects are the same for each addressee.

### 10.5.1 General-purpose functions

#### 11.3.1.1 Information-seeking functions

**Question:** Communicative function of a dialogue act performed by the sender, S, in order to obtain the information specified by the semantic content; S assumes that the addressee, A, possesses this information; S puts pressure on A to provide this information.

**Propositional Question** Communicative function of a dialogue act performed by the sender, S, in order to know whether the proposition, which forms the semantic content, is true. S assumes that A knows whether the proposition is true or not, and puts pressure on A to provide this information

**Set Question** Communicative function of a dialogue act performed by the sender, S, in order to know which elements of a given set have a certain property specified by the semantic content; S puts pressure on the addressee, A, to provide this information, which S assumes that A possesses. S believes that at least one element of the set has that property.

**Check Question** Communicative function of a dialogue act performed by the sender, S, in order to know whether a proposition, which forms the semantic content, is true, S holds the uncertain belief that it is true S. S assumes that A knows whether the proposition is true or not, and puts pressure on A to provide this information

**Choice Question** Communicative function of a dialogue act performed by the sender, S, in order to know which one from a list of alternative propositions, specified by the semantic content, is true; S believes that exactly one element of that list is true; S assumes that the addressee, A, knows which of the alternative propositions is true, and S puts pressure on A to provide this information.

#### 11.3.1.2 Information-providing functions

**Inform** Communicative function of a dialogue act performed by the sender, S, in order to make the information contained in the semantic content known to the addressee, A; S assumes that the information is correct.

**Agreement:** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S assumes a given proposition to be true, which S believes that A also assumes to be true.

**Disagreement** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S assumes a given proposition to be false, which S believes that A assumes to be true.

**Correction** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that certain information which S has reason to believe that A assumes to be correct, is in fact incorrect and that instead the information that S provides is correct.

**Answer** Communicative function of a dialogue act performed by the sender, S, in order to make certain information available to the addressee, A, which S believes A wants to know; S assumes that this information is correct.

**Confirm** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that certain information that A wants to know, and concerning which A holds an uncertain belief, is indeed correct.

**Disconfirm** Communicative function of a dialogue act performed by the sender, S, in order to let the addressee, A, know that certain information that A wants to know, and concerning which A holds an uncertain belief, is incorrect.

### 11.3.1.3 Commissive functions

**Promise** Communicative function of a dialogue act by which the sender, S, commits himself to perform the action, specified in the semantic content, in the manner or with the frequency or depending on the conditions that he makes explicit. S believes that this action would be in A's interest.

**Offer** Communicative function of a dialogue act by which the sender, S, indicates his willingness and ability to perform the action, specified by the semantic content, conditional on the consent of addressee A that S do so.

**Address Request** Communicative function of a dialogue act by which the sender, S, indicates that he considers the performance of an action that he was requested to perform.

**Accept Request** Communicative function of a dialogue act by which the sender, S, commits himself to perform an action that he has been requested to perform, possibly depending on certain conditions that he makes explicit.

**Decline Request** Communicative function of a dialogue act by which the sender, S, indicates that he refuses to perform an action that he has been requested to perform, possibly depending on certain conditions that he makes explicit.

**Address Suggest** Communicative function of a dialogue act by which the sender, S, indicates that he considers to perform an action that was suggested to him, possibly depending on certain conditions that he makes explicit.

**Accept Suggest** Communicative function of a dialogue act by which the sender, S, commits himself to perform an action that was suggested to him, possibly with certain restrictions or conditions concerning manner or frequency of performance.

**Decline Suggest** Communicative function of a dialogue act by which the sender, S, indicates that he will not perform an action that was suggested to him, possibly depending on certain conditions that he makes explicit.

#### 11.3.1.4 Directive functions

**Request** Communicative function of a dialogue act performed by the sender, S, in order to create a commitment for the addressee, A, to perform a certain action in the manner or with the frequency described by the semantic content, conditional on A's consent to perform the action. S assumes that A is able to perform this action.

**Instruct** Communicative function of a dialogue act performed by the sender, S, in order to create a commitment for the addressee, A, to carry out a named action in the manner or with the frequency specified by the semantic content; S assumes that A is able and willing to carry out the action.

**Address Offer** Communicative function of a dialogue act performed by the sender, S, in order to indicate that he is considering the possibility that A performs the action.

**Suggest** Communicative function of a dialogue act performed by the sender, S, in order to make the addressee, A, consider the performance of a certain action, specified by the semantic content,. S believes that this action is in A's interest, and assumes that A is able to perform the action.

**Accept Offer** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that S would like A to perform the action that A has offered to perform, possibly with certain conditions that he makes explicit.

**Decline Offer** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that S does not want A to perform the action that A has offered to perform, possibly depending on certain conditions that he makes explicit.

#### 10.5.2 Feedback functions

**Auto-Positive** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S believes that S's processing of the previous utterance(s) was successful.

**Allo-Positive** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S believes that A's processing of the previous utterance(s) was successful.

**Auto-Negative** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S's processing of the previous utterance(s) encountered a problem.

**Allo-Negative** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S believes that A's processing of the previous utterance(s) encountered a problem.

**Feedback Elicitation** Communicative function of a dialogue act performed by the sender, S, in order to know whether A's processing of the previous utterance(s) was successful.

### 10.5.3 Turn management functions

**Turn Accept** Communicative function of a dialogue act performed by the sender, S, in order to signal his willingness to take the speaker role, as requested by the previous speaker.

**Turn Assign** Communicative function of a dialogue act performed by the sender, S, in order to signal that he wants the addressee, A, to take the turn.

**Turn Grab** Communicative function of a dialogue act performed by the sender, S, in order to take the speaker role away from the participant who currently occupies it.

**Turn Keep** Communicative function of a dialogue act performed by the sender, S, in order to keep the speaker role.

**Turn Release** Communicative function of a dialogue act performed by the sender, S, in order to give other dialogue participants the opportunity to occupy the speaker role.

**TurnTake** Communicative function of a dialogue act performed by the sender, S, in order to have the speaker role, which is available at that moment.

### 10.5.4 Time management functions

**Stalling** Communicative function of a dialogue act performed by the sender, S, in order to have a little extra time to construct his contribution.

**Pausing** Communicative function of a dialogue act performed by the sender, S, in order to suspend the dialogue for a short while.

### 10.5.5 Own and partner communication management functions

**Completion** Communicative function of a dialogue act performed by the sender in order to assist the addressee in the completion of an utterance.

**Correct Misspeaking** Communicative function of a dialogue act performed by the sender, S, in order to correct (part of) an utterance by the addressee, A, assuming that A made a speaking error.

**Self- Error** Communicative function of a dialogue act performed by the sender, S, in order to signal to the addressee, A, that he (S) has made a mistake in speaking.

**Retraction** Communicative function of a dialogue act performed by the sender, S, in order to withdraw something that he just said within the same turn.

**Self-Correction** Communicative function of a dialogue act performed by the sender, S, in order to correct an error that he just made, or to improve on an infelicitous formulation that he just used, within the same turn.

### 10.5.6 Discourse structuring functions

**Interaction Structuring** Communicative function of a dialogue act performed by the sender, S, in order to explicitly indicate to the addressee, A, the function or topic of his next contribution(s).

**Opening** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that S is ready and willing to engage in a dialogue with A.

### 10.5.7 Social obligations management functions

**Initial Greeting** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A that S is present and aware of A's presence; S puts pressure on A to acknowledge this.

**Return Greeting** Communicative function of a dialogue act performed by the sender, S, in order to acknowledge that S is aware of the presence of the addressee, A, and of A having signalled his presence to S.

**Initial Self-Introduction** Communicative function of a dialogue act performed by the sender, S, in order to make himself known to the addressee, A; S puts pressure on A to acknowledge this.

**Return Self-Introduction** Communicative function of a dialogue act performed by the sender, S, in order to make himself known to the addressee, A in response to a self-introduction by A.

**Apology** Communicative function of a dialogue act performed by the sender, S, in order to signal that he wants the addressee, A, to know that S regrets something; S puts pressure on A to acknowledge this.

**Accept Apology** Communicative function of a dialogue act performed by the sender, S, in order to mitigate, the feelings of regret that the addressee, A, has expressed.

**Thanking** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that S is grateful for some action performed by A; S puts pressure on A to acknowledge this.

**Accept Thanking** Communicative function of a dialogue act performed by the sender, S, in order to mitigate to the feelings of gratitude which the addressee, A', has expressed.

**Initial Goodbye** Communicative function of a dialogue act performed by the sender, S, in order to inform the addressee, A, that S intends the current utterance to be his final contribution to the dialogue; S puts pressure on A to acknowledge this.

**Return Goodbye** Communicative function of a dialogue act performed by the sender, S, in order to acknowledge his awareness that the addressee, A, has signalled his final contribution to the dialogue and S signals in return his agreement to end the dialogue; S has been pressured to respond to an initialGoodbye by A.



## 10.6 Qualifiers

**Certainty:** Class of predicates which can be associated with a dialogue act to express whether the sender of a dialogue act with that function is certain or uncertain about the correctness of the information that he provides, or about the commitment that he takes on to perform a certain action .

**uncertain** Predicate which can be associated with a communicative function to express that the sender of a dialogue act with that function is uncertain about the correctness of the information that he provides, or about the commitment that he takes on to perform a particular action .

**certain** Predicate which can be associated with a communicative function to express that the sender of a dialogue act with that function is certain about the correctness of the information that he provides, or about the commitment that he takes on to perform a particular action .

**Conditionality:** Class of predicates which can be associated with a dialogue act with an action-discussion function to express whether the sender of a dialogue act with that function is considering the performance of the action under discussion subject to certain conditions.

**conditional** Predicate which can be associated with most action-discussion functions to express that the sender of a dialogue act with that function is considering the performance of the action under discussion subject to certain conditions.

**unconditional** Predicate which can be associated with an action-discussion function to express that the sender of a dialogue act with that function is considering the performance of the action under discussion without any conditions.

**Sentiment:** Class of predicates which can be associated with a dialogue act to express an emotional stance of the sender of a dialogue act with that function towards the semantic content of the dialogue act, or to express a mental attitude towards the addressee.

## Appendix B. Summary definitions of the rhetorical relations of ISO 24617-8 (ISO DR-Core')

ISO DR-core deals with *semantic* relations in discourse, at the exclusion of e.g. presentational relations. Semantic rhetorical relations are called 'discourse relations', and are viewed as relations between the meanings of discourse units such as sentences, clauses, or paragraphs in written text, and utterances or sequences of utterances in spoken dialogue. Such meanings are events, actions (including dialogue acts), states, propositions, beliefs or facts, and are designated by the umbrella term 'situation'. The definitions in Table 1 below are therefore all in terms of semantic relations between two situations, called Arg 1 and Arg2.

Of the 18 discourse relations of ISO 24617-8, 12 are asymmetric in the sense that, if the relation R holds between the arguments a and b in that order, then it does not hold in the reverse order. For example, if a causes b, then typically b does not cause a: if Cause(Arg1,Arg2) then not Cause(Arg2,Arg1). The two arguments of asymmetric relations

| Relation           | Definition                                                                                                                                                                                                                                                                           |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Cause              | Arg1 is an explanation for Arg2.                                                                                                                                                                                                                                                     |
| Condition          | Arg1 is an unrealized situation which, when realized, would lead to Arg2.                                                                                                                                                                                                            |
| Negative Condition | Arg1 is an unrealized situation which, when not realized, would lead to Arg2.                                                                                                                                                                                                        |
| Purpose            | Arg1 enables Arg2.                                                                                                                                                                                                                                                                   |
| Manner             | Arg1 specifies how Arg2 comes about or occurs.                                                                                                                                                                                                                                       |
| Concession         | An expected causal relation between Arg1 and Arg2 is cancelled or denied by Arg2.                                                                                                                                                                                                    |
| Contrast           | One or more differences between Arg1 and Arg2 are highlighted with respect to what each predicates as a whole or to some entities they mention.                                                                                                                                      |
| Exception          | Arg2 indicates one or more circumstances in which the situation(s) described by Arg1 does not hold.                                                                                                                                                                                  |
| Similarity         | One or more similarities between Arg1 and Arg2 are highlighted with respect to what each predicates as a whole or to some entities they mention.                                                                                                                                     |
| Substitution       | Arg1 and Arg2 are alternatives, with Arg2 being the favored or chosen alternative.                                                                                                                                                                                                   |
| Conjunction        | Arg1 and Arg2 bear the same relation to some other situation evoked in discourse. Their conjunction indicates that they both hold with respect to that situation.                                                                                                                    |
| Disjunction        | Arg1 and Arg2 bear the same relation to some other situation evoked in the discourse, explicitly or implicitly. Their disjunction indicates that they are alternatives with respect to that situation, with the disjunction being non-exclusive so that both Arg1 and Arg2 may hold. |
| Exemplification    | Arg1 describes a set of situations; Arg2 describes an element of that set.                                                                                                                                                                                                           |
| Elaboration        | Arg1 and Arg2 are the same situation, but Arg2 contains more detail.                                                                                                                                                                                                                 |
| Restatement        | Arg1 and Arg2 describe the same situation, but from different perspectives.                                                                                                                                                                                                          |
| Synchrony          | Some degree of temporal overlap exists between Arg1 and Arg2.<br>All forms of overlap are included.                                                                                                                                                                                  |
| Asynchrony         | Arg1 temporally precedes Arg2.                                                                                                                                                                                                                                                       |
| Expansion          | Arg2 is a situation involving some entity/entities in Arg1, expanding the narrative of which Arg1 is a part, or expanding on the setting relevant for interpreting Arg1. The Arg1 and Arg2 situations are distinct.                                                                  |

Table 1: Definitions of discourse relation in ISO 24617-8

| Relation           | Arg1 role               | Arg2 role            |
|--------------------|-------------------------|----------------------|
| Cause              | Result                  | Reason               |
| Condition          | Consequent              | Antecedent           |
| Negative Condition | Consequent              | Negated-antecedent   |
| Purpose            | Enablement              | Goal                 |
| Manner             | Achievement             | Means                |
| Concession         | Expectation-raiser      | Expectation-denier   |
| Exception          | Regular                 | Exclusion            |
| Substitution       | Disfavoured-alternative | Favoured-alternative |
| Exemplification    | Set                     | Instance             |
| Elaboration        | Broad                   | Specific             |
| Asynchrony         | Before                  | After                |
| Expansion          | Narrative               | Expander             |

Table 2: Argument roles of asymmetric discourse relations in ISO 24617-8

have different roles; for example, the argument of a Cause relation have the roles of Reason and Result. Table 2 lists the argument roles of the asymmetric rhetorical relations defined in ISO24617-8.

The six symmetric rhetorical relations of ISO 24617-8 due their symmetry to the fact that there are no differences between the two argument roles of the two arguments. This is most obvious for the relations Similarity and Contrast, which express that two situations are similar or are different in certain respects. For these relations no argument roles are defined; the designations ‘Arg1’ and ‘Arg2’ are used to identify the argument that occurs first resp. second in the discourse.