# The Universal Networking Language (UNL) Specifications, version 1.0

## April, 1998



# Introduction

The Universal Networking Language (UNL) is a metalanguage, or representational system, for describing, summarizing, refining, storing and disseminating information in a machine- and natural-language-independent form. The present document characterizes the notation, syntax and semantics of the UNL representation system.

### *Intended Audience*

This document is intended as a consistent and detailed definition of the UNL's conceptual basis and notation. The primary audience of this document consists of developers of UNL decoders, encoders and editors for different languages, as well the grammar and lexicon developers who work with them. The authors assume some familiarity with basic concepts of linguistics and semantics but attempt to clarify technical concepts and terminology whenever possible.

This document is not intended as a detailed guide about the representation of specific-language structures in UNL, although this is undeniably important and raises many interesting and relevant questions. In the same way that the specification of a programming language gives little information about specific algorithms and programming techniques, the present specification will give little information about specific linguistic problems or situations. Further documentation appropriate for each specific language will have to be developed to deal with these questions. Only a few general guidelines have been included here because the philosophy behind the UNL is not to dictate norms of encoding for all languages, rather to provide tools for discovering which strategies will be most appropriate for each.

### *Motivation to define the UNL*

The UNL is an effort to achieve a simple basis for representing the most central aspects of information and meaning in a machine- and human-language-independent form. Other, comparable, systems exist for annotating sentence meaning, leading to a serious problem: the existing systems vary greatly in detail and conception, depending in many cases on the natural language they were developed for, thus generating fundamental incompatibilities among them and inadequacies in dealing with different languages.

A language-independent metalanguage can circumvent this problem, permitting the coding, storage, dissemination and retrieval of information independently of the original language in which it was expressed. In this sense, UNL seeks to provide the tools for overcoming the language barrier in a systematic way.

The current effort has targeted 15 different languages in order to demonstrate the language independence and generality of the approach.

## *Goals and scope of the UNL*

The UNL can be seen as a kind of mark-up language which represents not the formatting but the core information of a text. As HTML annotations can be realized differently in the context of different applications, machines, displays, etc., so UNL expressions can have different realizations in different human languages.

It is important to note that at this point in time it would be foolish to state it possible to represent the "full" meaning of any word, sentence or text for any language. Subtleties of intention and interpretation make the "full meaning", whatever concept we might have of it, too variable and subjective for any systematic treatment. The UNL avoids the pitfalls of trying to represent the "full meaning" of sentences or texts, targeting instead the "core" or "consensual" meaning that is most often attributed to them.

In this sense, much of the subtlety of poetry, metaphor, figurative language, inuendo and other complex, indirect communicative behaviors is beyond the current scope and goals of the UNL. Instead, the UNL targets direct communicative behavior and literal meanings as a tangible, concrete basis for much or most of human communication in practical, day-to-day settings.

## *Extensions and modifications to the present specifications*

The present specifications forsee the development of extensions and modifications to the UNL representational system, while at the same time constraining them is specific ways. Three basic mechanisms are employed in the UNL: labeled links, "universal words" and attributes. Universal words and attributes can be added freely to the system, within the limits of the guidelines given below. Only the inventory of link labels is closed, not admitting additions, deletions or modifications. Once a year, the inventory of link labels will be evaluated to identify any major problems that might warrant modifications.

The metalanguage defined in this document is intended to be a stable basis for developing UNL tools and systems. This document, therefore, defines the core of the Universal Networking Language and supercedes existing, preliminary versions.

## *System architecture*

### The UNL system

The UNL system is a set of interrelated modules for the extraction, storage, retrieval and expression of information.

Encoding is the step where natural language texts are converted into UNL documents. Encoders are necessary for each human language used. For humans who produce UNL documents, a UNL Editor for that language will combine encoding and decoding modules that will provide the person with feedback about how accurate his UNL document is and provide tools to modify it until it is precise enough for the user's needs.

UNL documents accumulate in a UNL Document Base of human-language-independent information. A variety of tools are under development for exploring the UNL Document Base: search engines, tools for checking the well-formedness of the UNL expressions, tools that generate the full list of UWs and make them available to developers (the UW Gate), tools that explore the relations between the most important concepts in the Document Base to generate the Conceptual hierarchy or ontology, and tools that allow developers to view all of the Links in the Document Base (organized in different ways) – the Knowledge Base. Further tools, for automatic addition and elimination of Links, for example, can be imagined.

A UNL Viewer is available for accessing the Document Base and the different human-language versions of a given document. The Viewer makes on-line or off-line use of the Decoders developed for each natural language: these modules convert UNL statements into human-language sentences.

Thus, the UNL System is conceived of as an architecture, a metalanguage and a suite of tools for developing human-language-independent knowledge bases.

### The UNL metalanguage

The UNL represents information and meaning sentence by sentence for each sentence of a given text. Sentence information is represented as a list of interrelated semantic Links (or binary relations), each between two of the concepts present in the sentence.

Concepts are represented as character-strings called "UWs". Rather than provide a fixed, static definition for each UW, they are used to index the part of the knowledge base where they appear. Since the

knowledge base evolves as information is added, so the information associated with a given UW – its "meaning" – also evolves.

UWs can be annotated with Attributes which provide further information about how the concept is being used in the specific sentence where it was found.

The semantic Links that build structures out of UW concepts are signalled in natural language texts by different grammatical means: word order, suffixes, agreement, etc. for different languages. The UNL tools for each language define a systematic mapping between the grammatical clues of that language and the UNL relations that they signal. The Links can also be interrelated in complex ways to represent very complex relations between concepts or groups of concepts.

A UNL document, then, will be a long list of Links between the concepts cited in the natural-language text it was generated from, independently of the specific language it was in or of the specific grammatical mechanisms used for their expression.

It is important to understand that the UNL does not provide a single way of representing a given meaning. Rather, it provides tools and an environment for exploring different alternatives for semantic representations that are adequate for a wide variety of languages. During the development effort, sub-languages or "dialects" of the UNL will surely arise. The best of them will become *de facto* standards for the development community.

### The Role of English in the UNL

The role of English in the UNL is very limited, but has led to a great deal of misunderstanding. English-language labels are used for the LinkLabels, UWs and Attributes of the system, leading some people to believe that the UNL is somehow based on English, a form of disambiguated English or an effort to make English the universal language of the InterNet. None of these are quite true.

For the simple reason that almost all possible developers of the UNL will have access to English-language dictionaries, English is used as the language of communication for the project. Many of the Link labels and UWs denote things that are not at all common in the English language or in Anglo culture. The English labels are simple mnemonic devices, not intended to represent English-word concepts at all. Just to give a single example, it would be perfectly acceptable in the UNL to represent the special concept of "light making it possible for something to be seen" as an event with light as the agent as in the UW see(agt>light, exp>human), although this concept is quite alien to English speakers.

In the UNL, then, we use English to speak about concepts that are for the most part not at all language specific.

## *Overview of this document*

This document is a revised and exapanded version of the preliminary specifications first circulated in 1996[1] and since then in the form of several provisional manuscript versions. The present document is the result of intensive development efforts led by Hiroshi Uchida and the specifications that were made explicit by the members of the UNL Specifications Task Force: Hiroshi Uchida (United Nations University Institute for Advanced Studies, Tokyo – UNL Center), Igor Boguslavsky (Russian Academy of Sciences, Moscow, Russia – UNL Russia), Christian Boitet (Université Joseph Fourier, Grenoble, France – UNL France), Mike Dillinger (Federal University of Minas Gerais, Brazil – UNL Brazil), and Jörg Schütz (IAI, Saarbrücken, Germany – UNL Germany), with the help of other researchers participating in the UNL Project, in particular Meiying Zhu (UNU/IAS, Tokyo – UNL Center) and Oliver Streiter (IAI, Saarbrücken, Germany – UNL Germany). The final version of the specifications was written by Mike Dillinger.

The document has the following sections:

### Introduction

This Introduction provides information about the motivation, goals and scope of UNL, as well as an overview of its functioning.

[1] UNL Center. 1996. *UNL (Universal Networking Language): An electronic language for communication, understanding and collaboration* [Chapter 4]. Tokyo: United Nations University Institute for Advanced Studies.

### Links

The section on Links defines the closed set of conceptual relations that are the backbone of the UNL representation.

### Arguments: UWs and Scopal Units

This section defines the UWs and Scopal Units which are the elements or arguments that are related by conceptual relations to build Links. The set of UWs forms the basic inventory of concepts. The inventory is open, so guidelines for adding new UWs are presented.

### Attributes of Arguments

This section defines Attributes of arguments and presents the inventory of existing Attribute labels. Attributes are used to provide further information about how a particular UW or Scopal Unit is used in a given sentence.

# Links

Links, sometimes called "binary relations", are the building blocks of UNL documents. They are made up of a conceptual relation and two arguments, with some added mechanisms for making notations on the relation or arguments. Links often stand alone, but just as often can be grouped together in different ways. This section deals with the definition and interpretation of the types of conceptual relations that are used as the basis of the UNL.

Because of their similarity in name and function to "case relations" and "arguments" or "valences" in linguistics, and their close relation in practice to some grammatical structures, it may seem that the labels used for these conceptual relations are different names for special grammatical functions. This is emphatically not the case. The intention is that the labels used denote specific ideas rather than grammatical structures: the idea of "something that initiates an event," or "agent" for example, is quite different from "grammatical subject of a sentence", even though many times the subject of a sentence will indicate the agent of the event. The agent of an event may also appear as an adjective or noun modifier, with the preposition "by" or embedded in nouns with "er" suffixes. The whole point of the conceptual relations is to have a name for these very different grammatical structures which are conceptually quite the same. Thus, the conceptual relations used here are much more abstract than the grammatical relations found in sentences.

The conceptual relations between arguments in links have different labels according to the different concepts they represent. These LinkLabels are listed and defined below. Conventions for notating syntax are found in Appendix 1; fundamental terms from the conceptual hierarchy that appear in the definitions are defined in Appendix 2.

## *Internal structure of Links*

### Syntax

Links are made up as follows:

Link :: LinkLabel (ScopeNumber) "(" Argument "," Argument ")"

These elements will be defined below.

Example Links are:

mod:01(area(icl>place):02).@indef, strategic)
obj(designate(icl>event).@entry.@pred.@may, :01)
ppl(read(icl>event), home)

### LinkLabels

LinkLabels are strings of three lower-case alphabetic characters taken from the closed inventory listed below. Examples are the elements in bold face type below:

**mod**:01(area(icl>place):02).@indef, strategic)
**obj**(designate(icl>event).@entry.@pred.@may, :01)
**ppl**(read(icl>event), home)

### ScopeNumbers

ScopeNumbers are digits (":" followed by two digits) used to define groups of links (called "ScopalUnits") so that they can be referred to as a unit. Examples are the elements in bold face type below. The first example is an instance of ScopeNumbers being used to define a unit; the second example is an instance of ScopeNumbers being used to cite or refer to a ScopalUnit previously defined.

mod**:01**(area(icl>place):02).@indef, strategic)
obj(designate(icl>event).@entry.@pred.@may, **:01**)
ppl(read(icl>event), home(icl>place))

Note that the ":02: in the first example is NOT a ScopeNumber: ScopeNumbers are either attached directly to LinkLabels or appear alone, as arguments. See InstanceNumbers for further information.

### Arguments

Arguments can be UWs or ScopalUnits. Examples are the six elements in bold face type below. Non-standard formatting has been used to make them clearer.

mod:01( **area(icl>place):02).@indef**, **strategic**)
obj( **designate(icl>event).@entry.@pred.@may**, **:01**)
ppl( **read(icl>event),** **home(icl>place))**

# Conceptual relations

Conceptual relations and arguments are components of informational structures called events, states, facts, assertions, etc., which can be represented by one or more UNL Links. Conceptual relations are informationally distinct and represent identifiable general, recurring relations between the arguments cited in sentences. In the UNL, conceptual relations are represented as three-character strings called "LinkLabels" and are defined as specified below. For ease of exposition, we group discussion of these conceptual relations under the topics Events, States and Other relations, with other subtopics introduced as necessary.

## *Events*

Events are situations in which some affected thing (marked with obj, cob, exp or opl) changes over time with respect to its location or some of its other characteristics. In other words, the affected thing has an initial state which can be described in terms of its time initial characteristics (src), and this initial state changes over time (the event describes this change) until it reaches a final state, which can be described in terms of its final characteristics (gol). More details are given in the section about Affected things.

Events can be provoked, initiated or conditioned by things different from the affected thing, and these are called "event initiators" (marked as agt, cag, ptn or con). More details are given in the section about Event initiators.

Events can be delimited by citing the initial or final characteristics of the changing object (obj) or by citing a relevant time (tmf) or place (plf) to mark the beginning of the event and similarly with the time (tmt) or place (plt) that mark the end of the event. Further information is given in the section about Delimiters of events.

The global place and time of an event can be indicated with tim and ppl or lpl. These relations are described under States.

Events can also be described in terms of the characteristics or manner in which it occurs. Thus, we can speak of a specific manner, method or instrument (met), a characteristic of the event as a whole (man), and/or the purpose that the agent has in mind (pur). Further information is given in the section about Characteristics of events.

Details about all of these conceptual relations are provided below, one by one.

## Affected things

Events describe the changes that occur in what we call "affected things" and several kinds can be distinguished: the thing that changes in the main or focussed event (obj), a different thing that may change during an interrelated, non-focussed event (cob), something (usually human or human-like) that undergoes some subjective experience (exp) and places that are seen to be affected directly by the event (opl) in addition to other things (obj).

Because this relationship between the affected thing and the event remains the same whether the event initiator (agent, condition, etc.) is mentioned or not, different sentence structures with or without the initiator should lead to similar analyses.

## obj (changing or focussed thing)

***Definition.***

"Changing object" is defined as the relation between:
Argument1 – an event, and
Argument2 – a (concrete or abstract) thing,
where:

- Argument2 is not a place, and
- Argument2 is thought of as changing its characteristics or location as described by Argument1, or
- Argument2 is what Argument1 is about or refers to, when Argument1 is a "symbolic event" of perception, cognition, emotion, or communication.

***Conceptual examples.***

Situations of spontaneous or induced movement or change in state in some thing (obj).
Mental events that refer to concrete or abstract things (obj).

***Related concepts.***

Changing object is different from cob in that the obj is in focus, whereas the cob is related to a second, non-focussed event.

Changing object is different from exp in that obj is the topic of a symbolic event, whereas exp is the human (or human-like thing) where the symbolic event occurs.

Changing object is different from opl in that obj is not seen as a place, whereas opl is seen as a place.

***Syntax.***

obj(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

obj(move(icl>event), table(icl>object))
obj(melt(icl>event), snow(icl>substance))
obj(think(icl>event), Mary(icl>human))

***Readings.***

[Argument2] describes the changes that happened to [Argument1], or
[Argument1] was about [Argument2]
Examples:

The table moved.
[John] moved the table.
The snow melted.
[John] thought of Mary.

## cob (co-object)

***Definition.***

"Co-object" is defined as the relation between:
Argument1 – an event, and
Argument2 – a (concrete or abstract) thing,
where:

- Argument2 is not a place, and
- Argument2 is thought of as changing its characteristics or location as described by a usually implicit, non-focussed event that is different from Argument1 and considered to be its counterpart.

***Conceptual examples.***

Situations of exchange in which one thing (obj) is given by someone (agt) (in exchange) for another thing (cob), given by another person (cag). Which of the two things involved will be obj or cob is a question of focus.

***Related concepts.***

See the related concepts of obj, opl and exp.

***Syntax.***

cob(ScopeNumber) "(" Argument1 "," Argument2 ")"

Examples:

cob(get(icl>event), money(icl>object))
cob(give(icl>event), time(icl>object))

***Readings.***

[Argument2] changed in order for [Argument1] to happen, or
In exchange for [Argument2], [Argument1] changed

Examples:

...get [something in exchange] for money
...give [something in exchange] for time

## exp (experiencer)

***Definition.***

"Experiencer" is defined as the relation between:
Argument1 – an event or state, and
Argument2 – a human or non-human, seen-as-cognitive entity,
where:

- Argument1 is a subjective or physiological event or state, and
- Argument2 is thought of as experiencing, feeling or perceiving Argument1, or
- Argument2 is thought of as the reference, perspective or point of view for defining Argument1, or
- Argument2 is thought of as indirectly affected by Argument1, as victim or beneficiary, for example.

***Conceptual examples.***

Situations of subjective perceiving or feeling by someone (exp).
Situations in which subjective evaluations or characterizations are made in terms of a person (exp) used as a point of reference.

***Related concepts.***

obj
cob
opl

***Syntax.***

exp(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

exp(feel(icl>event), sick(icl>state))
exp(think(icl>event), Mary(icl>human))
exp(difficult(icl>state), John(icl>human))
exp(die(icl>event), Alex(icl>human))

***Readings.***

[Argument2] felt or perceived [Argument1], or
[Argument2] indirectly affected [Argument1], or
[Argument2] is the basis for describing [Argument1]

Examples:

[John] felt sick
Mary thought [about something]
[the test] was difficult for John.
[his mother] died on Alex

## opl (affected place)

***Definition.***

"Affected place" (or "obj-like place") is defined as the relation between:
Argument1 – an event, and
Argument2 – a place or thing defining a place,
where:

- Argument2 is the specific place where the change described by Argument1 is directed, or
- Argument2 is a place that is seen as being modified during the event.
- Argument2 is usually a part of the thing cited as obj; both the obj and the opl are modified during the event.

***Conceptual examples.***

Situations in which a place (opl) is seen as being affected by the event.
Situations in which a place (opl) is cited as part of the changing object (obj).

***Related concepts.***

Affected place is different from changing object, co-object and experiencer in that what is affected by the event is a place rather than other kinds of things.

Affected place is different from physical or logical place in that the Affected place is modified during the event, while the physical and logical place define the environment in which the event happens.

***Syntax.***

opl(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

opl(look(icl>event), eyes(icl>thing))
opl(pat(icl>event), shoulder(icl>thing))
opl(cut(icl>event), middle(icl>place))

***Readings.***

[Argument2] is the specific place toward which [Argument1] was directed, or
[Argument2] is the part of the obj that [Argument1] modified, or
[Argument2] is a place that was modified during [Argument1].
Examples:

...look [him] in the eyes
...pat [her] on the shoulder
...cut [the paper] in the middle

## Event Initiators

One set of conceptual relations focuses on how events are initiated, in particular, on factors that cause or condition how events come about.

Informally, we can talk about humans (or things similar to humans) who make events happen, and call their role in the event "agent". In situations of competition or collaboration, different agents can be distinguished and called "co-agents" and "partners".

Finally, when a state or event plays a more indirect role in making an event happen, perhaps making it possible or only modifying how it happens rather than causing it directly, it can be seen as external to the main event or playing a secondary role. This role is called "condition" and identifies these indirectly relevant factors.

### agt (agent)

***Definition.***

"Agent" is defined as the relation between:
Argument1 - an event, and
Argument2 - a human or non-human, seen-as-volitional thing
where:
•Argument 2 is thought of as having a direct role in making Argument 1 happen.

***Conceptual examples.***

Situations in which someone or something (agt) directly causes an event to happen or begin.

***Related concepts.***

Agent is different from co-agent in that agent initiates the event in focus, whereas the co-agent initiates a different, secondary event.

Agent is different from partner in that agent is the focussed initiator of the event, whereas the partner is a non-focussed initiator.

Agent is different from condition in that agent is the focussed initiator of an event whereas condition is an indirect, usually unfocussed influence on the event.

***Syntax.***

agt(ScopeNumber) "(" Argument1 "," Argument2 ")"

Examples:

agt(break(icl>event), John(icl>human))
agt(save(icl>event), computer(icl>machine))
agt(tell(icl>event), machine(icl>object))

***Readings.***

[Argument2] made [Argument1] happen, or
[Argument2] initiated [Argument1]

Examples:

| | |
|---|---|
| John made the breaking happen... | John broke... |
| the computer initiated the saving event... | the computer saved... |
| a machine initiated the telling event... | a machine told [me that]... |

## cag (co-agent)

***Definition.***

"Co-agent" is defined as the relation between:
Argument1 - an event, and
Argument2 - a human or non-human, seen-as-volitional thing
where:

- Argument2 is thought of as having a direct role in making Argument1 happen,
- Argument1 is a different kind or instance of the event that is initiated by the Agent, and
- both arguments of the cag relation are seen as not being in focus (as compared to the agent's event).

***Conceptual examples.***

Situations in which two people (agt and cag) are performing something in a coordinated, but independent way which is presented by focussing on one of them (agt).

***Related concepts.***

Co-agent is different from agent in that different, independent events occur for the agent and the co-agent. Moreover, the agent and its event are in focus, while the co-agent and its event are not in focus.

Co-agent is different from the partner in that the co-agent initiates an event that is independent of the agent's event, whereas the partner initiates the same event together with the agent.

Co-agent is different from condition in that the co-agent initiates a non-focussed event, whereas the condition is an indirect influence on the focussed event.

***Syntax.***

cag(ScopeNumber) "(" Argument1 "," Argument2 ")"

Examples:

cag(walk(icl>event), John(icl>human))
cag(live(icl>event),aunt(icl>human))
cag(talk(icl>event), machine(icl>object))

***Readings.***

[Argument2] made [Argument1] happen, or
[Argument2] initiated [Argument1]
Examples:

[Someone] walked with John
[He] is living with his aunt
[Someone] talked with a machine

## ptn (partner)

***Definition.***

"Partner" is defined as the relation between:
Argument1 - an event, and
Argument2 - a human or non-human, seen-as-volitional thing
where:

- Argument2 is thought of as having a direct role in making Argument1 happen,
- Argument1 is the same, collaborative event as that initiated by the Agent, and
- both arguments of the ptn relation are seen as not being in focus (as compared to the agent's event).

***Conceptual examples.***

Situations in which two people (agt and ptn) are jointly carrying out the same event and which are presented from the point of view of only one of the two (agt).

Situations like competing and sharing in which without either one of the participants there can be no joint event, that are presented focussing on only one of them (agt).

***Related concepts.***

Partner is different from agent in that the agent and its event are in focus, while the partner and its event are not in focus.

Partner is different from co-agent in that the co-agent initiates an event that is independent of the agent's event, whereas the partner initiates the same event together with the agent.

Partner is different from condition in that the partner initiates the same event as the agent does whereas the condition is only an indirect influence on that event.

***Syntax.***

ptn(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

ptn(compete(icl>event), John(icl>human))
ptn(share(icl>event),poor(icl>human))
ptn(collaborate(icl>event), machine(icl>object))

***Readings.***

[Argument2] made [Argument1] happen together with the agent, or
[Argument2] initiated [Argument1] together with the agent

Examples:

[Someone] competed with *John*
He shared the food with the *poor*.
[Someone] collaborated with a *machine*

## con (condition or influence)

***Definition.***

"Condition" is defined as the relation between:
Argument1 – an event or state, and
Argument2 – an event or state,
where:

- Arguments 1 and 2 are different and
- Argument2 is thought of as having an indirect role in making Argument1 happen, that is as some conditioning or possibilitating (or inhibiting) factor (real or hypothesized) which influences whether or when Argument1 can happen.

***Conceptual examples.***

Situations in which an event occurs only when some other event or situation (con) happens first or at the same time.

Situations citing requirements, permission, conditions, etc.

***Related Concepts.***

See the related concepts of agent, co-agent and partner.

***Syntax.***

con(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

aoj:01(green(icl>color), light (icl>object))
con(go(icl>event), :01)

agt:01(arrive(icl>event), Mary(icl>human))
agt:02(collaborate(icl>event), team(icl>human)
con(:02, :01)

***Readings.***

[Argument2] made it possible for [Argument1] to happen, or
[Argument2] was necessary for [Argument1] to happen, or
If [Argument2] is the case, then [Argument1] can or will happen.
Examples:

If the light is green, you can go.
Because Mary arrived, it was possible for the team to collaborate.

## Delimiters of Events

We identify as delimiters of events environmental or contextual times and places at the beginning of the event (tmf, plf) and at the end of the event (tmt, plt). These are distinct from the initial state (src) and final state (gol) of the changing object (obj) involved in that event, although src and gol can also be used to delimit events. In more complex examples, events can be delimited both in tems of the environment and in terms of the changing object, as in this example from English:

"I (agt) repeatedly poured water (obj) from the bottle (src) to the glass (gol), from here (plf) to the corner (plt)"

The water changes place, repeatedly going from the bottle to the glass and this complex event is further delimited in terms of the place where it started (here) and the place where it ended (corner). Similarly, delimitation in terms of starting and ending times is also possible in English:

"I repeatedly poured water from the bottle to the glass, from 10 o'clock (tmf) until noon (tmt)"

Further examples from English are pairs such as:

I filled the cup to here (plt).
I moved the cup to here (gol).

We talked from New York (plf) to Boston (plt).
We went from New York (src) to Boston (gol).

## tmf (initial time or time-from)

***Definition.***

"Initial time" is defined as the relation between:
Argument1 – an event, and
Argument2 – a time,
where:
•Argument2 specifies the time at which Argument1 started.

***Conceptual examples.***

Situations in which an event is delimited by stating when it started (tmf).

***Related concepts.***

Initial time is different from tim in that tmf expresses the time at the beginning of the event whereas tim expresses a time for the event taken as a whole.

Initial time is different from src in that tmf expresses the time at the beginning of the event whereas src expresses characteristics of the obj at the beginning of the event.

Initial time is different from tmt in that tmf expresses the time at the beginning of the event whereas tmt expresses the time at the end of the event.

***Syntax.***

tmf(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

tmf(look(icl>event), morning(icl>time))
tmf(think(icl>event), Sunday(icl>time))
tmf(cut(icl>event), minutes(icl>time))

***Readings.***

[Argument2] is the time at which [Argument1] began, or
[Argument1] has been going on since [Argument2]

Examples:

...looking since morning
...started thinking on Sunday
...cutting since [some] minutes [ago]

## plf (initial place or place-from)

***Definition.***

"Initial place" is defined as the relation between:
Argument1 – an event, and
Argument2 – a place or thing defining a place,
where:
•Argument2 is the specific place where Argument1 started.

***Conceptual examples.***

Situations delimited by characterizing the beginning of the event in terms of some place (plf).

***Related concepts.***

Initial place is different from ppl and lpl in that ppl and lpl describe events taken as wholes, whereas plf describes only the initial part of an event.

Initial place is different from plt in that plt describes the final part of an event, whereas plf describes the initial part of an event.

***Syntax.***

plf(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

plf(go(icl>event), home(icl>place))
plf(talk(icl>event), New York(icl>place))
plf(cut(icl>event), edge(icl>place))

***Readings.***

[Argument2] is the specific place from which [Argument1] started, or
[Argument2] is the place where [Argument1] started.
Examples:

...go from home to ...
...talk from New York [until Boston]
...cut [something] from one edge to ...

## src (initial characteristics)

***Definition.***

"Initial characteristics" (or "source state") is defined as the relation between:
Argument1 – an event, and
Argument2 – a state,
where:
•Argument2 is the specific state describing the obj of Argument1 at the beginning of Argument1.

***Conceptual examples.***

Situations in which the beginning of an event are described in terms of the initial characteristics of the changing object (obj).

***Related concepts.***

Initial characteristics is different from tmf and plf in that src describes qualitative characteristics and *not* time or place.

Initial characteristics is different from gol in that gol describes the characteristics of the obj at the *final* state of the event.

***Syntax.***

src(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

src(go(equ>change), sad(icl>characteristic))
src(change(icl>event), red(icl>color))
src(transform(icl>event), weak(icl>characteristic))

***Readings.***

[Argument2] is the state of the obj at the beginning of [Argument1].

Examples:

...[she] went from sad [to happy]
...[the light] changed from red [to green]
...[the drink] was transformed from weak [to very strong]

## tmt (final time or time-to)

***Definition.***

"Final time" is defined as the relation between:
Argument1 – an event, and
Argument2 – a time,
where:
•Argument2 specifies the time at which Argument1 ended.

***Conceptual examples.***

Situations in which an event is delimited by stating when it ended (tmt).

***Related concepts.***

Final time is different from tim in that tmt expresses the time at the end of the event whereas tim expresses a time for the event taken as a whole.

Final time is different from gol in that tmt expresses the time at the end of the event whereas gol expresses characteristics of the obj at the end of the event.

Final time is different from tmf in that tmt expresses the time at the end of the event whereas tmt expresses the time at the beginning of the event.

***Syntax.***

tmt(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

tmt(think(icl>event), morning(icl>time))
tmt(look(icl>event), Sunday(icl>time))
tmt(cut(icl>event), noon(icl>time))

***Readings.***

[Argument2] is the time when [Argument1] ended, or
[Argument1] continued until [Argument2]

Examples:

...think until morning
...stopped looking on Sunday
...cutting until noon

## plt (final place or place-to)

***Definition.***

"Final place" is defined as the relation between:
Argument1 – an event, and
Argument2 – a place or thing defining a place,
where:
•Argument2 is the specific place where Argument1 ended.

***Conceptual examples.***

Situations delimited by characterizing the end of the event in terms of some place (plf).

***Related concepts.***

Final place is different from ppl and lpl in that ppl and lpl describe events taken as wholes, whereas plt describes only the final part of an event.

Final place is different from plf in that plt describes the final part of an event, whereas plf describes the initial part of an event.

***Syntax.***

plt(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

plt(go(icl>event), home(icl>place))
plt(talk(icl>event), Boston(icl>place))
plt(cut(icl>event), edge(icl>place))

***Readings.***

[Argument2] is the specific place where [Argument1] ended, or
[Argument1] continued until the obj was at [Argument2].
Examples:

...go home [from ...]
...talk [from New York] until Boston
...cut [something from the middle] to the edge ...

## gol (final characteristics)

***Definition.***

"Final characteristics" (or "goal state") is defined as the relation between:
Argument1 – an event, and
Argument2 – a state,
where:
•Argument2 is the specific state describing the obj of Argument1 at the end of Argument1.

***Conceptual examples.***

Situations in which the end of an event is described in terms of the final characteristics of the changing object (obj).

***Related concepts.***

Final characteristics is different from tmf and plf in that gol describes qualitative characteristics and *not* time or place.

Final characteristics is different from src in that gol describes the characteristics of the obj at the *final* state of the event.

***Syntax.***

gol(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

gol(go(equ>change), sad(icl>characteristic))
gol(change(icl>event), red(icl>color))
gol(transform(icl>event), strong(icl>characteristic))

***Readings.***

[Argument2] is the state of the obj at the end of [Argument1].
Examples:

...[she] went [from happy] to sad
...[the light] changed [from greed] to red
...[the drink] was transformed [from weak] to strong

## Characteristics of Events

Besides delimiting events (described in the previous section), we can also describe the characteristics of an event or the general manner in which it is carried out (man), including more specific information about the method, means or instruments used (met), and the purpose that the agent has in mind or the objective he wants to pursue (pur).

The global time and place of an event can be described using tim and ppl or lpl, which are detailed in the section on States.

## met (method, means or instrument)

***Definition.***

"Method, means or instrument" is defined as the relation between:
Argument1 – an event, and
Argument2 – an (abstract or concrete) thing,
where:

•Argument2 specifies the (abstarct or concrete) thing which is used or the steps carried out in order to make Argument1 happen.

***Conceptual examples.***

Situations in which a thing, or use of a thing, is cited as a part of an event.
Situations in which specific steps or procedures are cited as part of an event.
Situations in which a particular method is used to carry out an event.

***Related concepts.***

Method, means or instrument is different from man in that ...

***Syntax.***

met(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

met(look(icl>event), telescope(icl>object))
met(solve(icl>event), algorithm(icl>method))
met(separate(icl>event), cutting(icl>event))

***Readings.***

[Argument2] is the means or method by which [Argument1] was carried out, or
[Argument1] was done using [Argument2].
Examples:

...look [at something] with a telescope
...solve [the problem] using an algorithm
...separate [two things] by cutting [them]

## pur (agent's purpose or objective)

***Definition.***

"Agent's purpose or objective" is defined as the relation between:
Argument1 – a (concrete or abstract) thing, and
Argument2 – a (concrete or abstract) thing,
where:

- The arguments are different, and
- Argument2 specifies the thing (object, state, event, etc.) that the agent desires to attain by carrying out Argument1, or
- Argument1 is done so that the agent can get/receive/acquire Argument2, or
- Argument2 is what Argument1 is to be used for.

***Conceptual examples.***

Situations in which an event or thing is characterized in terms of the objectives some agent wishes to reach.

Situations in which the purpose of some event or thing is described.

***Related concepts.***

Agent's purpose or objective is different from gol in that pur describes the desires of the agent, whereas gol describes the state of the obj at the end of the event.

Agent's purpose or objective is different from man and met in that pur describes the reason why the event is being carried out, while man and met describe how it is being carried out.

***Syntax.***

pur(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

pur(come(icl>event), see(icl>event))
pur(work(icl>event), money(icl>thing))
pur(budget(icl>money), research(icl>event))

***Readings.***

[Argument1] will be used to do [Argument1], or
[Argument2] is the purpose or objective associated with [Argument2].
Examples:

...come to see [me]
...work for money
...the budget for research

## man (manner)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

Situations where an event is described in terms of its duration or some distance travelled or changed.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)
man(sleep(icl>event), hours(icl>time))

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cut [something] with [his] hand in [his] pocket
...[Steven] slept for hours

### *States*

Situations can be described without referring to any of the changes that might be taking place. Such descriptions most often cite things, places and characteristics, but may cite events, usually by using grammatical means to remove their explicitly temporal characteristics, such as tense and aspect. These descriptions are called states and point to things and their characteristics without talking about specific changes. A state expressed by a noun phrase, for example, can also be combined with descriptions of events to characterize more complex situations.

Abstract or concrete things can be described in terms of their explicitly stated characteristics (aoj) or their pressuposed characteristics (cnt) and in terms of their position or location, either literally (ppl) or metaphorically (lpl). They can be associated with a point or period of time (tim) or with some range of phenomena (fmt), as well as described in terms of their parts (pof) or quantities (qua).

Furthermore, things can be described in relation to other things, by making comparisons (bas), describing proportions or distributions (per), or by chacterizing their associated times as sequential (seq) or simultaneous (coo).

These conceptual relations are described in the following section.

## tim (time)

***Definition.***

"Time" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a (point or interval of) time,
where:

- Argument1, taken as a whole, occurs at the time indicated by Argument2.

***Conceptual examples.***

Characteristics and changes in characteristics (events) can be described in terms of a time at which they occur.

***Related concepts.***

Time is different from tmf and tmf in that time characterized the event or state as a whole, whereas tmf and tmt describe only parts of the event.Time is different from coo and seq in that time does not describe states and events relatively, with respect to each other, but with respect to certain points in time.

Time is different from Attributes such as @past, @present, etc. in that these Attributes describe the situation with respect to the time at which the speaker is communicating, whereas time characterizes states and events with respect to other times, not the speaker's communicative act.

Duration of events is described using man.

***Syntax.***

tim(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

tim(look(icl>event), Tuesday(icl>time))
tim(red(icl>event), morning(icl>time))
tim(cut(icl>event), o'clock(icl>time))

***Readings.***

[Argument2] describes when [Argument1] is happening, or
[Argument1] was the case at the time described by [Argument2]
Examples:

...[I will] look on Tuesday
...[it was] red in the morning
...[he] cut [his finger] at [9] o'clock

## ppl (physical place)

***Definition.***

"Physical place" is defined as the relation between:
Argument1 – a (concrete or abstract) thing,
Argument2 – a concrete thing understood as a place,
where:

- The arguments are different, and
- Argument1 is or happens in a place characterized by Argument2.

***Conceptual examples.***

In situations where things are described in terms of where they are or where they occur, ppl is the relation between the things and the places.

Because abstract things can be events or characteristics, these can also be located with ppl.

***Related concepts.***

Physical place is different from lpl in that the reference place for ppl is concrete, whereas for lpl it is abstract or metaphorical.

Physical place is different from plf and plt or src and gol in that ppl describes a place with respect to an event as a whole, whereas these other relations describe position with respect to parts of an event.

Physical place is different from opl in that ppl is not seen as being modified by an event, merely a reference point for characterizing it, whereas opl is seen as being modified.

Ppl is used for absolute position in general. More specific place relations are best represented as UWs and used in structures such as:

ppl(man(icl>human), at(icl>place))
mod(at(icl>place), corner(icl>object))

Relative position can best be expressed using bas in structures such as:

ppl(book(icl>object), under(icl>place))
bas(under(icl>place), table(icl>object))

***Syntax.***

ppl(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

ppl(cook(icl>event), mountain(icl>object))
ppl(red(icl>characteristic), bottom(icl>object))
ppl(hand(icl>object), pocket(icl>object))

***Readings.***

[Argument2] describes where [Argument1] is or is happening.
Examples:

...cook [something] on a mountain
...[it is] red on the bottom
...[her] hand is in [her] pocket

## lpl (logical place)

***Definition.***

"Logical place" is defined as the relation between:
Argument1 – a (concrete or abstract) thing,
Argument2 – an abstract or metaphorical thing understood as a place,
where:

- The arguments are different, and
- Argument1 is or happens in a place characterized by Argument2.

***Conceptual examples.***

In situations where things are described metaphorically in terms of where they are or where they occur, lpl is the relation between the things and the places.

Because abstract things can be events or characteristics, these can also be located with lpl.

***Related concepts.***

Logical place is different from ppl in that the reference place for ppl is concrete, whereas for lpl it is abstract or metaphorical.

Logical place is different from plf and plt or src and gol in that lpl describes a place metaphorically, with respect to an event as a whole, whereas these other relations describe position with respect to parts of an event.

Logical place is different from opl in that lpl is not seen as being modified by an event, merely a reference point for characterizing it, whereas opl is seen as being modified.

Lpl is used for absolute position in general. More specific place relations are best represented as UWs and used in structures such as:

lpl(man(icl>human), at(icl>place))
mod(at(icl>place), ease(icl>characteristic))

Relative position can best be expressed using bas in structures such as:

lpl(woman(icl>object), under(icl>place))
bas(under(icl>place), stress(icl>characteristic))

***Syntax.***

lpl(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

lpl(cook(icl>event), under (icl>place))
mod(under(icl>place), pressure(icl>characteristic))
lpl(win(icl>characteristic), competition(icl>event))
lpl(doctor(icl>human), duty(icl>event))
lpl(surf(icl>event), internet(icl>object))

***Readings.***

[Argument2] describes metaphorically where [Argument1] is or is happening.
Examples:

...cook [something] under pressure
...win [something] in a competetion
...the doctor is on duty
...surf on the InterNet

Characteristics of things

***Introduction***

**fmt (range from-to)**

## aoj (attribute of things)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

## pof (part-of)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

## cnt?

non-focussed equivalence

***measures***

## qua (quantity)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

## per (proportion, rate or distribution)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

***comparisons***

## bas (basis of comparison)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

### *Other conceptual relations*

***Conjunction and disjunction***

## and (conjunction)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

## or (disjunction)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

***Indefinite relations***

## mod (somehow-related, indeterminate modification)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

## smd (not semantically related)

***Definition.***

"Manner" is defined as the relation between:
Argument1 – an event or state,
Argument2 – a state or characteristic,
where:

- The arguments are different, and
- Argument1 is done in a way characterized by Argument2, or
- Argument2 is a state associated (and simultaneous) with Argument1.

***Conceptual examples.***

In situations where events or states are described in terms of their characteristics, man is the relation between the events or states and their descriptions or characteristics, such as the way the event happens, its frequency, etc.

***Related concepts.***

Manner is conceptually quite similar to Attributes of "aspect" such as @progress, @complete, and @repeat. These attributes are specfic instances of manner that are so widespread that they have simply been given more convenient formatting conventions. It would be just as possible, in the UNL, to express them as manner statements, by representing the attribute as a UW.

Manner is different from met in that met describes how an event is carried out in terms of the instruments or component steps of the event, whereas man describes other quantitative or qualitative characteristics of the event as a whole.

***Syntax.***

man(ScopeNumber)"(" Argument1 "," Argument2 ")"

Examples:

man(look(icl>event), quickly(icl>manner))
man(think(icl>event), often(icl>frequency))
ppl:01(hand(icl>object), pocket(icl>object))
man(cut(icl>event), :01)

***Readings.***

[Argument2] describes how [Argument1 is happening, or
[Argument1] is happening in the way described by [Argument2]

Examples:

...look quickly
...think often [of someone]
...cutt [something] with [his] hand in [his] pocket

# Arguments: UWs and ScopalUnits

***Introduction***

***UWs***

***ScopalUnits***

# Attributes of Arguments

# Appendix 1: Conventions for notating syntax

# Appendix 2: Basic terms of the conceptual hierarchy