The Universal Networking Language (UNL)
Specifications
Version 3.0

UNL Center
UNDL Foundation
November 1, 2001

Introduction

The Universal Networking Language (UNL) is an electronic language for computers to express and exchange every kind of information.
The UNL represents information, i.e. meaning, sentence by sentence. Sentence information is represented as a hyper-graph having concepts as nodes and relations as arcs. This hyper-graph is also represented a set of directed binary relations, each between two of the concepts present in the sentence.
Concepts are represented as character-strings called "Universal Words (UWs)". UWs can be annotated with attributes that provide further information about how the concept is being used in the specific sentence.
A UNL document, then, will be a long list of relations between concepts.
Conventions for syntax notation are found in Appendix 3.

Chapter 1: UNL Expression

Binary relations are the building blocks of UNL sentences. They are made up of a relation and two UWs. This section deals with the definition and interpretation of the binary relations of the UNL.

There are two forms for expressing the UNL binary relations, one is table form and the other is list form. Table form of UNL expression is more readable than list form, but list form of UNL expression is more compact than table form. Here, only the table form is explained and the list form is shown in Chapter 5.

Any component such as a word, a phrase or a title and, of course, a sentence of a native language can be represented with UNL expressions. An UNL expression therefore consists of an UW or a (set of) binary relation(s). In the UNL documents, a set of UNL expressions for a sentence is enclosed by the tags {unl} and {/unl} inside [S] and [/S]. In case of an UW, the UW need to be enclosed further by the tags [W] and [/W].

Thus, the UNL expressions of a sentence are the following:

```
{unl}
<Binary Relation>
...
{/unl}
```

or,

```
{unl}
[W]
<UW>
[/W]
{/unl}
```

Syntax of binary relation

A binary relation is made up as follows:

<Binary Relation>	::= <Relation Label> [":"<Compound UW-ID>] "(" {{ <UW$_1$> [":" <UW-ID$_1$>]} | { ":" <Compound UW-ID$_1$> }}[<Attribute List>] "," {{ <UW$_2$> [":" <UW-ID$_2$>]} | { ":" <Compound UW-ID$_2$> }}[<Attribute List>] ")"
<Relation Label>	::= a relation, see "Chapter 2: Relations"
<UW>	::= an UW, see "Chapter 3: Universal Words"
<Attribute List>	::= { "." <Attribute> } …
<Attribute>	::= an attribute, see "Chapter 4: Attributes"
<UW-ID>	::= two characters of '0' – '9' and 'A' – 'Z'
<Compound UW-ID>	::= two-digit decimal number (00 – 99) 00 is used for representing the main sentence, which can be omitted.

Compound UW-IDs are strings of two digits used to identify each instance specified by Compound UWs. Compound UWs are groups of binary relations (so-called "Scope-Nodes") that can be referred to as a UW.

For instance, the following shows an example of UNL expressions of the sentence "I can hear a dog barking outside".

```
{unl}
aoj(hear(icl>perceive(agt>person,obj>thing)).@entry.@ability, I)
obj(hear(icl>perceive(agt>thing,obj>thing)).@entry.@ability, :01)
agt:01(bark(agt>dog).@entry, dog(icl>mammal))
plc:01(bark(agt>dog).@entry, outside(icl>place))
{/unl}
```

In the above UNL expressions, "**aoj**", "**agt**" and "**obj**" are the **relation labels**, "**hear(icl>perceive(agt>person,obj>thing))**", "**I**", "**bark(agt>dog)**", "**dog(icl>mammal)**" and "**outside(icl>place)**" are the **UWs**, "**:01**" appeared three times shows the Compound UW-ID. The Compound UW-ID in the position of an UW, so-called the "**scope-node**", is used to cite or refer to a Compound UW previously defined. The binary relations indicated by the Compound UW-ID show or define the contents (of UNL expressions) of the scope. A scope-node always begin with ":" followed by the two digits of a Compound UW-ID.

The UW-IDs are omitted from the above UNL expressions. It is possible when an UW is distinguishable from the others without the UW-ID.

The UW-ID is used to indicate some referential information: that there are two or more different occurrences of the same concept (they are not co-referent). Normally, if the same UW occurs more than once, it is in all cases understood to refer to the same entity or occurrence. For example, if one man greeted another man, the same UW would be used twice -- "man(icl>male person)" and we could distinguish one from the other with UW-IDs:

man(icl>male person):01 for the first and
man(icl>male person):02 for the other, to make it clear that the first man did not greet himself.

Chapter 2: Relations

This section deals with the definition and interpretation of the relation labels of the UNL.
The relations between UWs in binary relations have different labels according to the different roles they play. These Relation-Labels are listed and defined below.

Relation Labels

A relation label is represented as strings of 3 characters or less.

There are many factors to be considered in choosing an inventory of relations. The principles for choosing relations are as follows.

Principle-1 Necessary Condition
When an UW has relations between more than two other UWs, each relation label should be set as to be able to identify each relation on the premise that we have enough knowledge about the concept of each UW expressed.

Principle-2 Sufficient Condition
When there are relations between UWs, each relation label should be set so as to be able to understand the role of each UW only by referring to the relation label.

The following are the relations defined according to the above principles.

agt (agent)

Agt defines a thing that initiates an action.

agt (do, thing)

Syntax
agt [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed Definition
Agent is defined as the relation between:
UW1 - do, and
UW2 - a thing
where:
• UW2 initiates UW1, or
• UW2 is thought of as having a direct role in making UW1 happen.

Examples and readings

agt(break(agt>thing,obj>thing), John(icl>person))	John breaks …
agt(translate(agt>thing,gol>language,obj>information,src>language), computer(icl>machine))	computer translates …
agt(run(icl>act(agt>volitional thing)), car(icl>vehicle))	car runs …
agt(break(agt>thing,obj>thing), explosion(icl>event))	explosion breaks …

Related Relations
Agent is different from **cag** in that agent initiates the action, whereas the co-agent initiates a different, accompanied action.
Agent is different from **ptn** in that agent is the focused initiator of the action, whereas the partner is a non-focused initiator.
Agent is different from **con** in that agent is the focused initiator of the action, whereas condition is an indirect, usually unfocussed, influence on the action.

and (conjunction)

And defines a conjunctive relation between concepts.

and (*, *)

Syntax
and [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Conjunction is defined as the relation between:
UW1 – a concept, and
UW2 – another concept,

where:
- The UWs are different, and
- UW1 and UW2 are seen as grouped together, and
- what is said of UW1 is also said of UW2.

Examples and readings

and(quickly, easily)	… easily and quickly
and(sing(agt>person), dance(agt>person))	… singing and dancing
and(Mary(icl>person), John(icl>person))	… John and Mary

Related Relations

Conjunction is different from **or** in that with **and** we group things together to say the same thing about both of them, whereas with **or** we separate them to say that what is true about one is not true about the other.
Conjunction is different from **cag** in that when agents are conjoined both are initiating an explicit event, whereas with **cag**, the co-agent initiates an implicit event.
Conjunction is different from **ptn** in that when agents and partners are conjoined both are in focus, whereas with **ptn**, the partner is not in focus (as compared to the agent).
Conjunction is different from **coo** and **seq** in meaning, although many times the same expressions can be used for both. Conjunction only means that terms are grouped together; no information about time is implied. **Coo**, on the other hand, means that the terms are in the same time, whether or not they are considered to be grouped together. In turn, **seq** means that the terms are ordered in time, one after the other.

aoj (thing with attribute)

Aoj defines a thing that is in a state or has an attribute.

aoj (*(aoj>thing), thing)
aoj (thing, thing)
aoj (be, thing)

Syntax

aoj [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition

Thing with attribute is defined as the relation between:
UW1 – a state or a thing which represent a state, and
UW2 – a thing,
where:
- UW1 is an attribute or state of UW2, or
- UW1 is a state associated with UW2.

Examples and readings

aoj(red(aoj>thing), leaf(pof>plant))	... leaf is red.
aoj(available, information)	This information is available for …
aoj(nice, ski(agt>person))	Skiing is nice.
aoj(teacher(icl>occupation), John(icl>person))	John is a teacher.
aoj(have(aoj>thing,obj>thing), I)	I have a pen.
obj(have(aoj>thing,obj>thing), pen(icl>writing instrument))	
aoj(know(aoj>thing,obj>thing), John(icl>person))	John knows …
aoj:01(difficult(aoj>thing,obj>thing), it)	It is difficult for John.
aoj(:01, John(icl>person))	

Related Relations

Thing with attribute is different from **mod** in that **mod** gives some restriction, whereas **aoj** gives a state or characteristic.
Thing with attribute is different from **ben** in that a beneficiary is quite independent from an focused event or state but this event or state can be considered to give a good or bad influence, whereas **aoj** has more close relation and

can be considered to describe a state or characteristic.
Thing with attribute is different from **obj** in that **obj** defines a thing which is directly affected by action or phenomenon, whereas, **aoj** defines a thing in a state.

bas (basis for expressing degree)

Bas defines a thing used as the basis (standard) for expressing degree.

bas (degree, thing)

Syntax
bas [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Basis is defined as the relation between:
UW1 – a degree, and
UW2 – a thing,
where:
• UW1 is a degree expressing similarity or difference, such as “more”, “most”, “less”, “same”, “similar”, “as much as”, “at least” etc., and
• UW2 is some thing used as the basis for evaluating characteristics or quantity of some other (focused) thing.

Examples and readings

bas(more(aoj>thing), 7)	Ten is three more than seven.
bas(more(icl>how), Jack(icl>person))	Betty weighs more than Jack (does).
bas(same(icl>how), girl(icl>female person).@pl)	We treat boys exactly the same as girls
bas(at least, :01) qua:01(dollar(icl>money).@pl, 500)	It’ll cost at least 500 dollars.
man(beautiful, more(icl>how)) bas(more(icl>how), rose(icl>flower))	tulip is more beautiful than rose
aoj(:01, John(icl>person)) man:01(quiet(aoj>thing), more(icl>how)) bas:01(more(icl>how), shy(aoj>thing,mod<thing))	John is more quiet than shy.

ben (beneficiary)

Ben defines a not directly related beneficiary or victim of an event or state.

ben (occur, thing)
ben (do, thing)
ben ((aoj>thing), thing)

Syntax
ben [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed Definition
Beneficiary is defined as the relation between:
UW1 – an event or state, and
UW2 – a thing,
where:
• UW2 is thought of as indirectly affected by UW1, as beneficiary or victim.

Examples and readings

ben(give(agt>thing,gol>thing,obj>thing)), Mary(icl>person))	To give … for Mary.
ben(good(aoj>thing,mod<thing)), John(icl>person))	It is good for John to …

Related Relations
Beneficiary is different from **aoj** in that **aoj** has close relation and can be considered to describe a state characteristic, whereas a beneficiary is quite independent from an focused event or state, but this event or state can be considered to give a good or bad influence.

cag (co-agent)

Cag defines a thing not in focus that initiates an implicit event that is done in parallel.

cag (do, thing)

Syntax
cag [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition
Co-agent is defined as the relation between:
UW1 - an action, and
UW2 - a thing
where:
• There is an implicit action that is independent of, but accompanies, UW1, and
• UW2 is thought of as initiating the implicit action, and
• UW2 and the implicit action are seen as not being in focus (as compared to the agent's action).

Examples and readings

cag(walk(icl>do), John(icl>person))	To walk with John
cag(live(icl>do), aunt(icl>person))	To live with … aunt

Related relations
Co-agent is different from **agt** in that differing independent actions occur for the agent and the co-agent. Moreover, the agent and its action are in focus, while the co-agent and its action are not in focus.
Co-agent is different from the **ptn** in that the co-agent initiates an action that is independent of the agent's action, whereas the partner initiates the same action together with the agent.
Co-agent is different from **con** in that the co-agent initiates a non-focused action, whereas the condition is an indirect influence on the focused action.

cao (co-thing with attribute)

Cao defines a thing not in focus, is in a state in parallel.

cao ((aoj>thing), thing)
cao (thing, thing)

Syntax
cao [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition
Co-thing with attribute is defined as the relation between:
UW1 – a state or a thing which represents a state, and
UW2 – a thing,
where:
• There is an implicit state that is independent of, but accompanies, UW1, and
• UW2 is in an implicit state, or
• UW2 is associated with an implicit state.

Examples and readings

cao(exist(icl>be), you)	… be with you

Related relations
Co-thing with attribute is different from **aoj** in that there is a different, independent state for the thing with attribute and co-thing with attribute, respectively.

cnt (content)

Cnt defines an equivalent concept.

cnt (thing, thing)

Syntax
cnt [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed Definition
Content is defined as the relation between:
UW1 – a thing, and
UW2 – a thing,
where:
• UW2 is a content or explanation of UW1.

Examples and readings

cnt(UNL(icl>Universal Networking Language), Universal Networking Language)	UNL, Universal Networking Language
cnt(Internet(icl>communication network), amalgamation(icl>harmony))	The Internet: an amalgamation
cnt(language generator, deconverter.@double_quotation)	a language generator "deconverter"…

cob (affected co-thing)

Cob defines a thing that is directly affected by an implicit event done in parallel or an implicit state in parallel.

cob (occur, thing)
cob (do, thing)
cob ((aoj>thing,obj>thing), thing)

Syntax
cob [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed Definition
"Co-object" is defined as the relation between:
UW1 – an event or state, and
UW2 – a thing,
where:
• UW2 is thought of as directly affected by an implicit event done in parallel or an implicit state in parallel.

Examples and readings

cob(die(obj>living thing), Mary(icl>person))	… dead with Mary
obj(injure(icl>hurt(agt>thing,obj>living thing), John(icl>person)) cob(injure(icl>hurt(agt>thing,obj>living thing), friend(icl>comrade).@pl) pos(friend(icl>comrade).@pl, he)	John was injured in the accident with his friends

Related Relationss
Co-object is different from **obj** in that the **obj** is in focus, whereas the **cob** is related to a second, non-focused implicit event or state.

con (condition)

Con defines a non-focused event or state that conditions a focused event or state.

con (occur, occur)
con (occur, do)
con (occur, (aoj>thing))
con (do, occur)
con (do, do)
con (do, (aoj>thing))
con ((aoj>thing), occur)
con ((aoj>thing), do)
con ((aoj>thing), (aoj>thing))

Syntax
con [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Condition is defined as the relation between:
UW1 – a focused event or state, and
UW2 – a conditioning event or state,
where:
• UW1 and UW2 are different and
• UW2 is thought of as having an indirect or external role in making UW1 happen, that is as some conditioning or inhibiting factor (real or hypothesized) which influences whether or when UW1 can happen.

Examples and readings

aoj:01(tired(aoj>thing,mod<thing), you) con(go(icl>move(agt>thing,gol>place,src>place)), :01)	If you are tired, we will go straight home.

coo (co-occurrence)

Coo defines a co-occurrent event or state for a focused event or state.

coo (occur, occur)
coo (occur, do)
coo (occur, (aoj>thing))
coo (do, occur)
coo (do, do)
coo (do, (aoj>thing))
coo ((aoj>thing), occur)
coo ((aoj>thing), do)
coo ((aoj>thing), (aoj>thing))

Syntax
coo [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Co-occurrence is defined as the relation between:
UW1 – a focused event or state, and
UW2 – a co-occurrent event or state,
where:
• UW1 and UW2 are different, and
• UW1 occurs or is true at the same time as UW2.

Examples and readings

coo(run(icl>act(agt>volitional thing)), cry(icl>weep(agt>volitional thing))	… was crying while runing

coo(red(aoj>thing,mod<thing), hot(aoj>thing,mod<thing)) … is red while … is hot

Related Relations
Co-occurrence is different from **seq** in that **seq** describes events or states that do not occur at the same time, but one after the other, whereas **coo** describes events that occur simultaneously.
Co-occurrence is different from **tim** in that **coo** relates the times of events or states with other events or states, whereas **tim** relates events or states directly with points or intervals of time.

dur (duration)

Dur defines a period of time during which an event occurs or a state exists.

dur (occur, period)
dur (occur, event)
dur (occur, state)
dur (occur, occur)
dur (occur, do)
dur (occur, *(aoj>thing))
dur (do, period)
dur (do, event)
dur (do, state)
dur (do, occur)
dur (do, do)
dur (do, *(aoj>thing))
dur (*(aoj>thing), period)
dur (*(aoj>thing), event)
dur (*(aoj>thing), state)
dur (*(aoj>thing), occur)
dur (*(aoj>thing), do)
dur (*(aoj>thing), *(aoj>thing))

Syntax
dur [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Duration is defined as the relation between:
UW1 – an event or a state, and
UW2 – a period that the event or state continues.

Examples and readings
dur(work(agt>person), hour(icl>period)) … work nine hours (a day)
qua(hour(icl>period), 9)
dur(talk(icl>express(agt>thing,gol>person,obj>thing), meeting(icl>event)) … talk … during meeting
dur(come(icl>move(agt>thing,gol>place,src>place), absence(icl>state)) … come during (my) absence

fmt (range: from-to)

Fmt defines a range between two things.

fmt (thing, thing)

Syntax
fmt [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Range (from-to) is defined as the relation between:
UW1 – a range-initial thing, and
UW2 – a range-final thing,
where:
• The UWs are different, and
• UW2 describes the beginning of a range and UW1 describes the end.

Examples and readings

fmt(a(icl>letter), z(icl>letter))	the alphabets from a to z
fmt(Osaka(icl>city), New York(icl>city))	the distance from Osaka to New York
fmt(Monday(icl>day), Friday(icl>day))	the weekdays from Monday to Friday

Related Relations
Range is different from **src** and **gol** in that for **src** and **gol** the initial and final states of some **obj** are characterized with respect to some event, whereas **fmt** makes a similar characterization but without linking the endpoints of a range to some event.
Range is different from **plf** and **plt** or **tmf** and **tmt** in that **fmt** defines endpoints of a range without reference to any sort of event, whereas **plf**, **plt**, **tmf** and **tmt** delimit events.

frm (origin)

Frm defines an origin of a thing.

frm (thing, thing)

Syntax
frm [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Origin is defined as the relation between:
UW1 – a thing, and
UW2 – an origin of the thing,
where:
• UW2 describes the origin such as original position of UW1.

Examples and readings

frm(visitor(icl>person), Japan(icl>country))	a visitor from Japan

gol (goal: final state)

Gol defines a final state of object or a thing finally associated with object of an event.

gol (occur(gol>thing), thing)
gol (do(gol>thing), thing)

Syntax
gol [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Final state is defined as the relation between:
UW1 – an event, and
UW2 – a state or thing,
where:
• UW2 is the specific state describing the **obj** (of UW1) at the end of UW1, or
• UW2 is a thing that is associated with the **obj** (of UW1) and the end of UW1.

Examples and readings

gol(change(gol>thing,obj>thing,src>thing), red(aoj>thing,mod<thing))	the lights changed from green to red
gol(deposit(icl>save(agt>thing,obj>thing)), account(icl>record))	millions were deposited in Swiss bank account

Related Relations

Final state is different from **tmf** and **plf** in that **gol** describes qualitative characteristics and not time or place. Final state is different from **src** in that **gol** describes the characteristics of the **obj** at the final state of the event.

ins (instrument)

Ins defines an instrument to carry out an event.

ins (do, concrete thing)

Syntax

ins [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

Instrument is defined as the relation between:
UW1 – an event, and
UW2 – a concrete thing,
where:
• UW2 specifies the concrete thing that is used in order to make UW1 happen.

Examples and readings

ins(look(agt>thing,obj>thing), telescope(icl>optical instrument)	look at stars through [with] a telescope
ins(write(icl>express(agt>thing,obj>thing)), pencil(icl>stationery))	write [draw] with a pencil
ins(cut(agt>thing,obj>thing)), scissors(icl>cutley))	He cut the string with a pair of scissors

Related Relations

Instrument is different from **man** in that **man** describes an event as a whole, whereas **ins** characterizes one of the components of the event: the use of the instrument.
Instrument is different from **met** in that **met** is used for abstract things (abstract means or methods), whereas “ins” is used for concrete things.

man (manner)

Man defines a way to carry out event or characteristics of a state.

man (occur, how)
man (do, how)
man ((aoj>thing), how)

Syntax

man [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

“Manner” is defined as the relation between:
UW1 – an event or state, and
UW2 – a manner,
where:
• The UWs are different, and
• UW1 is done or exists in a way characterized by UW2.

Examples and readings

man(move(agt<thing,gol>place,src>place), quickly)	move quickly
man(visit(agt>thing,obj>thing)), often)	I often visit him.
man(beautiful, very(icl>how))	it is very beautiful.

Related Relations

Manner is different from **ins** or **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.

met (method or means)

Met defines a means to carry out an event.

met (do, abstract thing)
met (do, do)

Syntax

met [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

“Method or means” is defined as the relation between:
UW1 – an event, and
UW2 – an abstract thing or an action,
where:

- UW2 specifies the abstract thing used or the steps carried out in order to make UW1 happen.

Examples and readings

met(solve(icl>resolve(agt>thing,obj>thing)), dynamics(icl>science))	… solve … with dynamics
met(solve(icl> resolve(agt>thing,obj>thing)), algorithm(icl>method))	… solve … using … algorithm
met(separate(agt>thing,obj>thing,src>thing)), cut(agt>thing,obj>thing)	… separate … by cutting …

Related Relations

Method or means is different from **man** in that **man** describes an event as a whole, whereas **met** characterizes the component steps, procedures or instruments of the event.
Method or means is different from **ins** in that **met** is used for abstract things (abstract means or methods), whereas **ins** is used for concrete things.

mod (modification)

Mod defines a thing that restricts a focused thing.

mod (thing, thing)
mod (thing, (mod>thing))

Syntax

mod [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

“Modification” is defined as the relation between:
UW1 – a focused thing, and
UW2 – a thing that restricts UW1 in some way.

Examples and readings

mod(story(icl>tale), whole(mod<thing))	the whole story
mod(plan(icl>idea), master(mod<thing))	a master plan
mod(part(pof>thing), main(aoj>thing))	the main part
qua(block(icl>concrete thing), 3)	… three blocks of ice …
mod(ice(icl>solid), block(icl>concrete thing))	

Related Relations

Modification is different from **aoj** in that **aoj** describes a state or characteristic of a thing, whereas **mod** merely indicates a restriction, which might indirectly suggest some characteristics of the thing described. Most **mod** relations require a paraphrase introducing some implicit event to become clearer, and even then many possibilities are usually available.

Modification is different from **man** in that **man** describes a way to carry out event or characteristics of a state.

nam (name)

Nam defines a name of a thing.

nam (thing, thing)

Syntax

nam [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition

Name is defined as the relation between:
UW1 – a thing, and
UW2 – a thing used as a name,
where:
• UW2 is a name of UW1.

Examples and readings

Nam(tower(icl>building), Tokyo(icl>city))	Tokyo tower

obj (affected thing)

Obj defines a thing in focus that is directly affected by an event or state.

obj (occur, thing)
obj (do, thing)
obj (be, thing)
obj ((aoj>thing,obj>thing), thing)

Syntax

obj [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed Definition

Affected thing is defined as the relation between:
UW1 – an event or state, and
UW2 – a thing,
where:
• UW2 is thought of as directly affected by an event or state.

Examples and readings

obj(move(gol>place,obj>thing,src>place), table(icl>furniture))	the table moved.
obj(melt(gol>thing,obj>thing), sugar(icl>seasoning))	the sugar melts into …
obj(cure(agt>thing,obj>thing), patient(icl>person))	to cure the patient.

obj(have(aoj>thing,obj>thing), pen(icl>writing instrument))	I have a pen.

Related Relations

Affected thing is different from **cob** in that **obj** is in focus, whereas **cob** is related to a second, non-focused implicit event or state.

opl (affected place)

Opl defines a place in focus where an event affects.

opl (do, place)

Syntax

opl [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed Definition

Affected place is defined as the relation between:
UW1 – an event, and
UW2 – a place or thing defining a place,
where:

- UW2 is the specific place where the change described by UW1 is directed, or
- UW2 is a place that is seen as being affected during the event.

Examples and readings

opl(pat(icl>do(obj>thing)), shoulder(icl>limb))	… pat … on shoulder
opl(cut(icl>do(obj>thing)), middle(icl>place))	… cut … in middle

Related Relations

Affected place is different from **obj** and **cob** in that what is affected by the event is a place rather than other kinds of things.
Affected place is different from **plc** in that the affected place is characterized by the event, while the physical and logical place defines the environment in which the event happens.

or (disjunction)

Or defines disjunctive relation between two concepts.

or (thing, thing)

Syntax

or [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition

Disjunction is defined as the relation between:
UW1 – a thing, and
UW2 – a concept,
where:

- The UWs are different, and
- Some description is true for either UW1 or UW2 (but not both), or
- Some description is true for either UW1 or UW2 (and perhaps both).

Examples and readings

or(stay(icl>do), leave(icl>do))	Will you stay or leave?
or(red(icl>color), blue(icl>color))	Is it red or blue?
or(John(icl>person), Jack(icl>person))	Who is going to do it, John or Jack?

Related Relations

Disjunction is different from conjunction that the items of disjunction are grouped in order to say that something is true for one or the other, whereas in conjunction they are grouped to say that the same is true for both. Disjunction in formal logic permits three situations for a disjunction to be true: 1) it is true for UW1, 2) it is true for UW2, 3) it is true for both. On the other hand, conjunction only permits the third situation.

per (proportion, rate or distribution)

Per defines a basis or unit of proportion, rate or distribution.

per (thing, thing)

Syntax

per [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

Proportion, rate or distribution is defined as the relation between:
UW1 – a quantity, and
UW2 – a quantity, or a thing seen as a quantity,
where:

- UW1 and UW2 form a proportion, where UW1 is the numerator and UW2 is the denominator, or
- UW2 is the basis or unit for understanding UW1, or
- Each UW expresses a different dimension, of size, for example.

Examples and readings

per(hour(icl>period), day(icl>period)) qua(hour(icl>period), 8)	eitgh hours a day
per(time(icl>unit), week(icl>period)) qua(time(icl>unit), 2)	… twice a week

plc (place)

Plc defines a place an event occurs or a state is true or a thing exists.

plc (occur, thing)
plc (do, thing)
plc ((aoj>thing), thing)
plc (thing, thing)

Syntax

plcl [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

Place is defined as the relation between:
UW1 – an event, state, or thing, and
UW2 – a place or thing understood as a place.

Examples and readings

plc(cook(icl>do), kitchen(pof>building))	… cook … in the kitchen
plc(sit(icl>do), beside(icl>relative place))	… sit beside me
plc(cool(aoj>thing), here(icl>how))	It’s cool here.

Related Relations

Place is different from **plf** and **plt** or **src** and **go**l in that **plc** describes a place with respect to an event as a whole, whereas these other relations describe position with respect to parts of an event.
Place is different from **opl** in that **plc** is not seen as being modified by an event, merely a reference point for characterizing it, whereas **op**l is seen as being modified.

plf (initial place)

Plf defines a place an event begins or a state becomes true.

plf (occur, thing)
plf (do, thing)
plf ((aoj>thing), thing)

Syntax
plf [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
“Initial place” (or “place-from”) is defined as the relation between:
UW1 – an event or state, and
UW2 – a place or thing defining a place,
where:
- UW2 is the specific place where UW1 started, or
- UW2 is the specific place from where UW1 is true.

Examples and readings

plf(come(icl>do), home(icl>place))	… come from home
plf(deep(aoj>thing), there(icl>how))	The sea is deep from there to here.

Related Relations
Initial place is different from **plc** in that **plc** describes events or states taken as wholes, whereas **plf** describes only the initial part of an event or state.
Initial place is different from **plt** in that **plt** describes the final part of an event or state, whereas **plf** describes the initial part of an event or state.
Initial place is different from **src** in that **plf** describes the place where the event began, whereas **src** describes the initial state of the object.

plt (final place)

Plt defines a place an event ends or a state becomes false.

plt (occur, thing)
plt (do, thing)
plt ((aoj>thing), thing)

Syntax
plt [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Final place is defined as the relation between:
UW1 – an event or state, and
UW2 – a place or thing defining a place,
where:
- UW2 is the specific place where UW1 ended, or
- UW2 is the specific place where UW2 becomes false.

Examples and readings

plt(travel(icl>do), Boston(icl>city))	I’m travelling until Boston
plf(deep(aoj>thing), there(icl>how))	The sea is deep from there to here

Related Relations
Final place is different from **plc** in that **plc** describes events or states 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 or state, whereas **plf** describes the

initial part of an event.
Final place is different from **gol** in that **plt** describes the place where an event or state ended, whereas **gol** describes the final state of the object.

pof(part-of)

Pof defines a concept of which a focused thing is a part.

pof (thing, thing)

Syntax
pof [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Part-of is defined as the relation between:
UW1 – a partial thing, and
UW2 – a whole thing,
where:
• UW1 is a part of UW2.

Examples and readings

Pof(wing(icl>limb), bird(icl>animal))	Bird’s wing.

pos (possessor)

Pos defines the possessor of a thing.

pos (thing, volitional thing)

Syntax
pos [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Possessor is defined as the relation between:
UW1 – a thing or a place, and
UW2 - a human or non-human, seen-as-volitional thing
where:
• UW2 is a possessor of UW1.

Examples and readings

pos(dog(icl>aminal), John(icl>person))	John’s dog
pos(book(icl>concrete thing), I)	my book

ptn (partner)

Ptn defines an indispensable non-focused initiator of an action

ptn (do, thing)

Syntax
ptn [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Partner is defined as the relation between:
UW1 - an action, and
UW2 - a human or non-human, seen-as-volitional thing
where:
• UW2 is thought of as having a direct role in making an indispensable part of UW1 happen, and

• UW1 is the same, collaborative event as that initiated by the agent, and
• UW2 is seen as not being in focus (as compared to the agent).

Examples and readings

ptn(compete(icl>do), John(icl>person))	… compete with John
ptn(share(icl>do(obj>thing)), poor(icl>person))	… share … with the poor
ptn(collaborate(icl>do), he)	… collaborate with him …

Related Relations

Partner is different from **agt** in that the agent and its event are in focus, while the partner and its event are not in focus.
Partner is different from **cag** 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 **con** in that the partner initiates the same event as the agent does, whereas the condition is only an indirect influence on that event.

pur (purpose or objective)

Pur defines the purpose or objective of an agent of an event or a purpose of a thing that exist.

pur (occur, occur)
pur (occur, do)
pur (do, occur)
pur (do, do)
pur (occur, thing)
pur (do, thing)
pur (thing, occur)
pur (thing, do)
pur (thing, thing)

Syntax

pur [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition

Purpose or objective is defined as the relation between:
UW1 – a thing or an event, and
UW2 – a thing or an event,
where:
• The UWs are different, and
When UW1 is an event:
• UW2 specifies the agent's purpose or objective, or
• UW2 specifies the thing (object, state, event, etc.) that the agent desires to attain by carrying out UW1, or
When UW1 is not an event:
• UW2 is what UW1 is to be used for.

Examples and readings

pur(come(icl>do), see(icl>do(obj>thing)))	… come to see you
pur(work(icl>do), money(icl>do))	... work for money
pur(budget(icl>expense), research(icl>do))	our budget for research

Related Relations

Purpose or objective is different from **gol** in that **pur** describes the desires of the agent, whereas **gol** describes the state of the object at the end of the event.
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.

qua (quantity)

Qua defines quantity of a thing or unit.

qua (thing, quantity)

Syntax
qua [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Quantity is defined as the relation between:
UW1 – a thing, and
UW2 – quantity,
where:
• UW2 is the number or amount of UW1.

Examples and readings

qua(cup(icl>tabelware), 2)) mod(coffee(icl>beverage), cup(icl>tableware))	Two cups of coffee
qua(kilogram(icl>unit), many(aoj>thing))	many kilograms
qua(dog(icl>animal), 2)	two dogs

Related Relations
Quantity is different from **per** in that quantity is an absolute number or amount, whereas **per** is a number or amount relative to some unit of reference (time, distance, etc.).
Quantity is also used to express iteration, or number of times an event or state occurs.

rsn (reason)

Rsn defines a reason that an event or a state happens.

rsn (occur, thing)
rsn (do, thing)
rsn (occur, occur)
rsn (occur, do)
rsn (do, occur)
rsn (do, do)
rsn (occur, (aoj>thing))
rsn (do, (aoj>thing))
rsn ((aoj>thing), occur)
rsn ((aoj>thing), do)
rsn ((aoj>thing), thing)
rsn ((aoj>thing), (aoj>thing))

Syntax
rsn [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Reason is defined as the relation between:
UW1 – an event or state, and
UW2 – a reason of an event or state,
where:
• UW2 is a reason that UW1 happens.

Examples and readings

rsn(go(icl>do), rain(icl>weather))	... didn't go because of the rain
agt:01(arrive(icl>do), Mary(icl>person)) rsn(start(icl>do(obj>thing)), :01)	They can start because Mary arrived.

rsn(known(aoj>thing), beauty(icl>abstract thing)) mod(city(icl>region), known(aoj>thing)) mod(beauty(icl>abstract thing), city(icl>region))	a city known for its beauty

scn (scene)

Scn defines a virtual world where an event occurs or state is true or a thing exists.

scn (do, thing)
scn (occur, thing)
scn ((aoj>thing), thing)
scn (thing, thing)

Syntax
scn [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
Scene is defined as the relation between:
UW1 – an event or state or thing, and
UW2 – an abstract or metaphorical thing understood as a place,
where:
• The UWs are different, and
• UW1 is or happens in a place characterized by UW2.

Examples and readings

scn(win(icl>do(obj>thing)), contest(icl>event))	… win a prize in a contest
scn(apear(icl>occur), program(icl>thing))	… appear on a TV program
scn(play(icl>do), movie(icl>entertainment))	… play in movie

Related Relations
Scene is different from **plc** in that the reference place for **plc** is in real world, whereas for **scn** it is abstract or metaphorical world.

seq (sequence)

Seq defines a prior event or state of a focused event or state.

seq (occur, occur)
seq (occur, do)
seq (do, occur)
seq (do, do)
seq (occur, (aoj>thing))
seq (do, state)
seq ((aoj>thing), occur)
seq ((aoj>thing), do)

Syntax
seq [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition
“Sequence” is defined as the relation between:
UW1 – a focused event or state,
UW2 – a prior event or state,
where:
• The UWs are different, and
• UW1 occurs or is true after UW2.

Examples and readings

seq(leap(icl>do), look(icl>do))	Look before you leap.
seq(red(aoj>thing), green(aoj>thing))	It was green and then red.
seq(take off(icl>do(obj>thing)), come in(icl>do))	She came in and took her coat off.

Related Relations

Sequence is different from **coo** in that **seq** describes events or states that do not occur at the same time, but one after the other, whereas **coo** describes events that occur simultaneously.
Sequence is different from **bas** in that **seq** describes events or states in terms of order in time, whereas **bas** describes things or states in terms of qualitative differences or similarities.

src (source: initial state)

Src defines the initial state of an object or thing initially associated with the object of an event.

src (occur, thing)
src (do, thing)

Syntax

src [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition

Initial state is defined as the relation between:
UW1 – an event, and
UW2 – a state or thing,
where:
- UW2 is the specific state describing the object of UW1 at the beginning of UW1, or
- UW2 is a thing that is associated with the object of UW1 at the beginning of UW1.

Examples and readings

src(change(icl>occur), red(aoj>thing))	The lights changed from green to red.
src(withdraw(icl>do(obj>thing)), stove(icl>furniture))	I quickly withdrew my hand from the stove.

Related Relations

Initial state is different from **tmf** and **plf** in that src describes qualitative characteristics and not time or place.
Initial state is different from **gol** in that **gol** describes the characteristics of the object at the final state of the event.

tim (time)

Tim defines a time an event occurs or a state is true.

tim (occur, time)
tim (do, time)
tim ((aoj>thing), time)

Syntax

tim [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition

Time is defined as the relation between:
UW1 – an event or state, and
UW2 – a time,
where:
- UW1, taken as a whole, occurs at the time indicated by UW2.

Examples and readings

tim(leave(icl>do), Tuesday(icl>time))	… leave on Tuesday

tim(do(obj>thing), o'clock(icl>time)) … do … at … o'clock
tim(start(icl>do), come(icl>do)) Let's start when … come

Related Relations
Time is different from **tmf** and **tmt** in that time characterizes 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.

tmf (initial time)

Tmf defines a time an event starts or a state becomes true.

tmf (occur, time)
tmf (do, time)
tmf ((aoj>thing), time)

Syntax
tmf [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition
Initial time is defined as the relation between:
UW1 – an event or state, and
UW2 – a time,
where:
• UW2 specifies the time at which UW1 starts, or
• UW2 specifies the time at which UW1 became/becomes true.

Examples and readings
tmf(work(icl>do), morning(icl>time)) … work from morning to [till] night
tmf(change(icl>occur), live(icl>do)) … has changed … since I have lived here.

Related Relations
Initial time is different from **tim** in that **tmf** expresses the time at the beginning of the event or state 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 or state whereas **src** expresses characteristics of the object at the beginning of the event.
Initial time is different from **tmt** in that **tmf** expresses the time at the beginning of the event or state whereas **tmt** expresses the time at its end.

tmt (final time)

Tmt defines a time an event ends or a state becomes false.

tmt (occur, time)
tmt (do, time)
tmt ((aoj>thing), time)

Syntax
tmt [":"<Compound UW-ID>] "(" {<UW1>|":"<Compound UW-ID>} "," {<UW2>|":"<Compound UW-ID>} ")"

Detailed definition
Final time is defined as the relation between:
UW1 – an event or state, and
UW2 – a time,
where:
• UW2 specifies the time at which UW1 ends, or
• UW2 specifies the time at which UW1 became/becomes false.

Examples and readings

tmt(work(icl>do), night(icl>time)) … work from moning to [till] night

tmt(full(aoj>thing), tomorrow(icl>time)) … be full till tomorrow

Related Relations

Final time is different from **tim** in that **tmt** expresses the time at the end of the event or state, 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 or state, whereas **gol** expresses characteristics of the object at the end of the event.

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

to (destination)

To defines a destination of a thing.

to (thing, thing)

Syntax

to [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

Destination is defined as the relation between:
UW1 – a thing, and
UW2 – a destination of the thing,
where:

- UW2 describes the destination such as final position of UW1.

Examples and readings

to(train(icl>vehicle), London(icl>city)) a train for London

to(letter(icl>message), you) a letter to you

via (intermediate place or state)

Via defines a intermediate place or state of an event.

via (occur(gol>thing,src>thing), thing)
via (do(gol>thing,src>thing), thing)

Syntax

via [“:”<Compound UW-ID>] “(“ {<UW1>|“:”<Compound UW-ID>} “,” {<UW2>|”:”<Compound UW-ID>} “)”

Detailed definition

Intermediate place or state is defined as the relation between:
UW1 – an event, and
UW2 – a place or state,
where:

- UW2 is the specific place or state describing the object of UW1 at some time in the middle of UW1,
- UW2 is a thing that describes a place or state that the object of UW1 passed by or through during UW1.

Examples and readings

via(go(icl>do), New York(icl>city)) … go … via New York

via(bike(icl>do), Alps(icl>place)) … bike … through the Alps

via(drive(icl>do), tunnel(icl>topography)) … drive … by way of tunnel

Related Relations

Intermediate place or state is different from **src**, **plf** and **tmf** in that these all refer to the beginning of an event,

whereas **via** describes the middle of an event.
Intermediate place or state is different from **gol**, **plt** and **tmt** in that these all refer to the end of an event, whereas **via** describes the middle of an event.

Chapter 3: Universal Words

A UW (Universal Word) represents simple or compound concepts. There are two classes of UWs:

• simple, unit concepts called “UWs” (Universal Words), and

• compound structures of binary relations grouped together and called “Compound UWs”. These are indicated with Compound UW-IDs, as described below.

3.1 UWs

3.1.1 Syntax of UW

A UW is made up of a character string (an English-language word) followed by a list of constraints. The meaning and function of each of these parts is described in the next section, on Interpretation.
The following expressions provide a more formal statement of the syntax of UWs.

```
<UW>                ::= <Head Word> [<Constraint List>]
<Head Word>         ::= <character>…
<Constraint List>   ::= “(“ <Constraint> [ “,” <Constraint>]… “)”
<Constraint>        ::= <Relation Label> { “>” | “<” } <UW> [<Constraint List>] |
                        <Relation Label> { “>” | “<” } <UW> [<Constraint List>]
                        [ { “>” | “<” } <UW> [<Constraint List>] ] …
<Relation Label>    ::= “agt” | and” | “aoj” | “obj” | “icl” | ...
<character>         ::= “A” | ... | “Z” | “a” | ... | “z” | 0 | 1 | 2 | ... | 9 | “_” | ” “ | “#” | “!” | “$” |
                        “%” | “=” | “^” | “~” | “|” | “@” | “+” | “-“ | “<” | “>” | “?”
```

3.1.2 Interpretation

HeadWord

The Head Word is an English word/compound word/phrase/sentence that is interpreted as a label for a set of concepts: the set made up of all the concepts that may correspond to that in English. A Basic UW (with no restrictions or Constraint List) denotes this set. Each Restricted UW denotes a subset of this set that is defined by its Constraint List. Extra UWs denote new sets of concepts that do not have English-language labels.
Thus, the headword serves to organize concepts and make it easier to remember which is which.

Constraints or Restrictions

The Constraint List restricts the interpretation of a UW to a subset or to a specific concept included within the Basic UW, thus the term “Restricted UWs”.
The Basic UW “drink”, without a Constraint List, includes the concepts of “putting liquids in the mouth”, “liquids that are put in the mouth”, “liquids with alcohol”, “absorb” and others.
The Restricted UW “drink(icl>do,obj>liquid)” denotes the subset of these concepts that includes “putting liquids in the mouth”, which in turn corresponds to verbs such as “drink”, “gulp”, “chug” and “slurp” in English.
The restrictions of Restricted UWs, their Constraint Lists, are Constraints. The Constraints that use the Relation Labels defined above can be seen as an abbreviated notation for full binary relations: drink(icl>do,obj>liquid) is the same as obj(drink(icl>do),liquid) which means something like “cases of drinking where the ‘obj’ is a liquid”.
Every constraint in the Constraint List should use the Relation Labels listed in Appendix 2 and each of them should be sorted in alphabetical order.
Such a relation label can be omitted when it repeats the same one in its left side and for restricting the subset concept. For instance, an UW like “xxx(icl>change(icl>occur))” can be defined as “xxx(icl>change>occur)” simply.

3.1.3 Types of UW

UWs, then, are character strings (words or expressions) that can be given specifications, attributes and

Instance-IDs. Their function in the UNL system is to represent simple concepts. The three types of UWs, in order of practical importance are:

- Basic UWs, which are bare Head Words with no Constraint List, for example:

  go
  take
  house
  state

- Restricted UWs, which are Head Words with a Constraint List, for example:

  state(icl>express)
  state(icl>country)
  state(icl>abstract thing)
  state(icl>government)

- Extra UWs, which are a special type of Restricted UW, for example:

  ikebana(icl>flower arrangement)
  samba(icl>dance)
  soufflé(icl>food)

Basic UWs

Basic UWs are character strings that correspond to an English word. A basic UW denotes all the concepts that may correspond to that in English. They are used to structure the knowledge base and as a fallback method for establishing correspondences between different language words when more specific correspondences cannot be found.

Restricted UWs

Restricted UWs are by far the most important. Each Restricted UW represents a more specific concept, or subset of concepts. The Constraint List restricts the range of the concept that a Basic UW represents.
The Basic UW "drink", with no Constraint List, includes the concepts of "putting liquids in the mouth", "liquids that are put in the mouth", "liquids with alcohol", "absorb" and others.
The Restricted UW "drink(icl>do(obj>liquid))" denotes the subset of these concepts that includes "putting liquids in the mouth", which in turn corresponds to verbs such as "drink", "gulp", "chug" and "slurp" in English.

Consider again the examples of Restricted UWs given above:

> **state(icl>express)** is a more specific concept (arbitrarily associated with the English word "state") that denotes situations in which humans produce some information, or express something.
> **state(icl>country)** is a more specific sense of "state" that denotes a nation or country.
> **state(icl>abstract thing)** is a more specific sense of "state" that denotes a kind of condition that persons or things are in. This UW is defined as a more general concept that can be referred to when defining other synonymous UWs like "situation" or "condition".
> **state(icl>government)** is a more specific sense of "state" that denotes a kind of government.

The information in parentheses is the Constraint List and it describes some conceptual restrictions; that's why these are called Restricted UWs. Informally, the restrictions mean "restrict your attention to this particular sense of the word". Thus, the focus is clearly the idea and not the specific English word.
It often turns out that for a given language there is a wide variety of different words for these concepts and not, coincidentally, all the same word, as in English.
Notice that by organizing these senses around the English words, we can simplify the task of making a new UW/Specific Language dictionary: we can use a bilingual English/Specific Language dictionary and proceed from there, specifying the number of different concepts necessary for each English word.

This of course does not mean that we're translating English words; we're just using the English dictionary to remind us of the concepts that we will want to deal with and thus to organize work more efficiently.

Extra UWs

Extra UWs denote concepts that are not found in English and that have to be introduced as extra categories. Foreign-language words are used as Head Words using English (Alphabetical) characters. Consider again the examples given above:

> **ikebana(icl>flower arrangement)** is "a kind of flower arrangement" for the meaning of "something you do with flowers",
> **samba(icl>dance)** is "a kind of dance", and
> **soufflé(icl>food)** is "a kind of food".

To the extent that these concepts exist for English speakers, they are expressed with foreign-language loanwords and don't always appear in English dictionaries. So, they simply have to be added if we are going to be able to use these specific concepts in the UNL system. Notice that the Constraint List or restrictions give the idea of what kind of concept is associated with these Extra UWs and the Constraints provide binary relations between this concept and other concepts already present and more general (action, dance, food, etc.).

3.2 Compound UWs

Compound UWs are a set of binary relations that are grouped together to express a complex concept. A sentence itself is considered a compound UW. This allows us to deal with situations like:
Women who wear big hats in movie theaters should be asked to leave.
Without Compound UWs, we wouldn't be able to build up complex ideas like "women who wear big hats in move theaters" and then relate them to other concepts.

Compound UWs denote complex concepts that are to be interpreted as unit concepts, understood as a whole so that we can talk about their parts all at the same time. Consider again the example given above.
[Women who wear big hats in movie theaters] should be asked [to leave].
~~The example does not mean that [women] or [women who wear big hats] should be asked to leave. Only when we group the structure together and talk about it as a whole unit do we get the correct interpretation.~~
~~[zhu changed above as follows:]~~
The all parts in square brackets are what should be asked. Only when we group them together and talk about it as a whole unit we do get the correct interpretation.
Just as we can relate such complex units to other concepts with conceptual relations, we can attach attributes to them to express negation, speaker attitudes, etc., which are usually interpreted as modifying the main predicate within the Compound UW.

3.2.1 The way to define a Compound UW

A Compound UW is defined by placing a Compound UW-ID immediately after the Relation Label in all of the binary relations that are to be grouped together. Thus, in the example below, ":01" indicates all of the elements that are to be grouped together to define Compound UW number 01.

```
agt:01(wear(icl>do(obj>thing)), woman(icl>person).@pl)
obj:01(wear(icl>do(obj>thing)), hat(icl>clothes))
aoj:01(big(aoj>thing), hat(icl>clothes))
plc:01(wear(icl>do(obj>thing), theater(icl>facilities))
mod:01(theater(icl>facilities), movie(icl>entertainment))
agt:01(leave(icl>do).@entry, woman(icl>person).@pl)
```

After this group has been defined, wherever the Compound UW-ID, for instance "01" in the above example, can be used to cite the Compound UW. The way to cite a Compound UW is explained in the next section.
A Compound UW is considered as a sentence or sub-sentence, so in the definition of a Compound UW one entry node marked by @entry is necessary.

3.2.2 The way to cite a Compound UW

Once defined, a Compound UW can be cited or referred to by simply using the Compound UW-ID as an UW. The way is to indicate the Compound UW-ID following a colon ":". The reference to a Compound UW is also called Scope-Node. The Scope-Node has the following syntax:

<Scope-Node>	::= ":" <Compound UW-ID> [<Attribute List>]
<Compound UW-ID>	::= two digits of a number 00 – 99
<Attribute List>	::= { "." <Attribute Label> } …
<Attribute Label>	::= "@entry" | "@may" | "@past" | ...

To complete the example above, we could continue with:

```
obj(ask(icl>do(obj>thing)).@should, :01)
gol(ask(icl>do(obj>thing)).@should, woman.@pl)
```

Again, ":01" is interpreted as the whole set of binary relations defined above. It means that ":01" should be understood as all of these binary relations. Compound UWs can be cited within other Compound UWs.

Chapter 4: Attributes

Attributes of UWs are used to describe the subjectivity of sentences. They show what is said from the speaker's point of view: how the speaker views what is said. This includes phenomena technically called "speech acts", "propositional attitudes", "truth values", etc. Conceptual relations and UWs are used to describe the objectivity of sentences. Attributed of UWs enrich this description with more information about how the speaker views these states-of-affairs and his attitudes toward them. Such attributes play the role of bridging the conceptual world represented by UWs and relations, and the real world. Or, such attributes bring the concept defined by UWs and relations into the real.

4.1 Time with respect to the speaker

Where does the speaker situate his description in time, taking his moment of speaking as a point of reference? A time before he spoke? After? At approximately the same time? This is the information that defines "narrative time" as past, present or future. These Attributes are attached to the main predicate.
Although in many languages this information is signaled by tense markings on verbs, the concept is not tense, but "time with respect to the speaker". The clearest example is the simple present tense in English, which is not interpreted as present time, but as "independently of specific times".
Consider the example: The earth is round.
This sentence is true in the past, in the present and in the future, independently of speaker time, so although the tense is "present" it is not interpreted as present time.

@past	happened in the past	ex) It <u>was</u> snowing yesterday
@present	happening at present	ex) It'<u>s raining</u> hard.
@future	will happen in future	ex) He <u>will arrive</u> tomorrow

4.2 Speaker's view of Aspect

A speaker can emphasize or focus on a part of an event or treat it as a whole unit. This is closely linked to how the speaker places the event in time. These Attributes are attached to the main predicate.
The speaker can focus on the beginning (@begin) of the event, looking forward to it (@begin.@soon), or backward to it (@begin.@just).
He can also focus on the end (@end) or completion (@complete) of the event, looking forward to it (@end.@soon or @complete.@soon), or backward to it (@end.@just or @complete.@just).
~~Degree of forwardness or backwardness (@soon, @just).~~
He can focus on the middle (@progress) or continuation (@continue) of the event.
The speaker can choose to focus on the lasting effects or final state of the event (@state) or on the event as a repeating unit (@repeat), experience (@experience) or custom (@custom).
He can also focus on the incompleteness or not yet happen of an event using @yet.

@begin	beginning of an event or a state	Ex) It <u>began to</u> work again. work.@begin.@past
@complete	finishing/completion of a (whole) event.	Ex) <u>I've looked through</u> the script. look.@entry.@complete
@continue	continuation of an event	Ex) He <u>went on talking</u>. talk.@continue.@past
@custom	customary or repetitious action	Ex) I <u>used to visit</u> [I <u>would often go</u>] there when I was a boy. visit.@custom.@past
@end	End/termination of an event or a state	Ex) I <u>have done</u> it. do.@end.@present
@experience	Experience	Ex) Have you ever visited Japan? visit.@experience.@interrogation Ex) I have been there. visit.@exterience
@progress	an event is in progress	Ex) I <u>am working</u> now. work.@progress.@present

@repeat	repetition of an event	Ex) He is jumping. jump.@entry.@present.@repeat
@state	final state or the existence of the object on which an action has been taken	Ex) It is broken. break.@state

The following attributes are used to modify the attributes above.

@just	Immediateness backward to the beginning, end or completion of an event	Ex) He has just come. come.@complete.@just
@soon	Immediateness forward to the beginning, end or completion of an event	Ex) The train is about to leave. leave.@begin.@soon
@yet	feeling of not yet begin, end or complete	Ex) I have not yet done it. do.@complete.@not.@yet

4.3 Speaker's view of Reference

Whether an expression refers to a single individual, a small group or a whole set is often not clear. The expression "the lion" is not sufficiently explicit for us to know whether the speaker means "one particular lion" or "all lions". Consider the following examples:
The lion is a feline mammal.
The lion is eating an anti-lope.
In the first example, it seems reasonable to suppose that the speaker understood "the lion" as "all lions", whereas in the second example as "one particular lion".
The following Attributes are used to make explicit what the speaker's view of reference seems to be.

@generic	generic concept	Ex) The dog is a faithful animal.
@def	already referred	Ex) the book you lost
@indef	non-specific class	Ex) There is a book on the desk.
@not	complement set	Ex) Don't be late!
@ordinal	ordinal number	Ex) the 2^{nd} door

These attributes are usually attached to UWs that denote things.

4.4 Speaker's view of Emphasis, Focus and Topic

The speaker can choose to focus or emphasize the parts of a sentence to show how important he thinks they are in the situation described. This is often related to sentence structure.

@emphasis	The emphasized UW	Ex) I do like it.
@entry	The entry or main UW of a sentence or a scope	Ex) He promised (entry of the sentence) that he would come(entry of the scope)
@qfocus	The focused UW of a question	Ex) Are you painting the bathroom blue? To this question, the answer will be "No, I'm painting the LIVING-ROOM blue"
@theme	Instantiates an object from different class	Ex)
@title	Title	Ex)
@topic	Topic	Ex) He(@topic) was killed by her. Ex) The girl(@topic) was given a doll. Ex) This doll(@topic) was given to the girl.

One UW marked with "@entry" is essential to each UNL expression or in a Compound UW.

4.5 Speaker's Attitudes

The speaker can also express, directly or indirectly, what his attitudes or emotions are toward what is being said or who it is being said to. This includes respect and politeness toward the listener and surprise toward what is being said.

@affirmative	**Affirmation**
@confirmative	**Confirmation** Ex) “You won't say that, will you?” Ex) “sou desu ne?” (In Japanese)
@exclamation	**Feeling of exclamation** Ex) “kirei na!”(“How beautiful (it is)!” in Japanese) Ex) “Oh!, look out!”, “Ow!”
@imperative	**Imperative** Ex) “Get up!” Ex) “You will please leave the room.”
@interrogative	**Interrogation** Ex) “Who is it?”
@invitation	**Inducement to do something** Ex) “Will / Won’t you have some tea?” Ex) “Let’s go, shall we?”
@polite	**Polite feeling** Ex) “Could you (please)...” Ex) “If you could … I would …”
@request	**Request** Ex) “please don’t forget…”
@respect	**Respectful feeling** Ex) “o taku”(“(your) house” in Japanese) Ex) “Good morning, sir.”
@vocative	**Vocative** Ex) “Boys(@vocative), be ambitious!”

4.6 Speaker’s viewpoint (??)

~~The variety of possibilities for view point reflects degrees of belief, emphasis, and the extent to which what is said should be interpreted as a suggestion or order, as well as many other social factors such as the relative status of the speakers.~~
The following attributes are used to clarify the speaker’s viewpoint information.

~~**@ask-back**~~	~~**ask back**~~
~~**@doubt**~~	~~**Have doubt**~~
~~**@induce**~~	~~**Induce to do**~~

Judgment	
@ability	**Ability, capability of doing something** Ex) The child can 't walk yet. Ex) He can speak English but he can’t write it very well.
@admiration	**Admiration** Ex) kirei naa! (How beautiful (it is)! In Japanese)
Assumption and Consequence	
@unreal	**Unrealistic** Ex) If we had enough money, we could buy a car.
@consequence	**a natural (logical) result or consequence** Ex) The city is situated near the sea and consequently enjoys a healthy climate.
@unexpected-consequence	**result or consequence contrary to a wish or expectation** Ex)
Consent	
@grant	**To give/get consent/permission to do something** Ex) Can I smoke in here? Ex) You may borrow my car if you like.

@grant-not	**Not to give consent to do something** Ex) You {mustn't/are not allowed to/may not} borrow my car.
Necessity, obligation	
@need	**Feel necessity of doing something** Ex) You needn't finish that work today.
@obligation	**Obligation to do something according to (quasi-) law, contract, or …** Ex) The vendor shall maintain the equipment in good repair.
@obligation-not	**Obligation not to do something, forbid to do something according to (quasi-) law, contract or …** Ex) Cars must not park in front of the entrance. Ex) No smoking
@should	**to do something as a matter of course** Ex) You should do as he says. Ex) You ought to start at once.
Possibilities	
@inevitability	**Logical inevitability that something is true of happens** Ex) There must be a mistake. Ex) They should be home by now.
@possibility	**Logical possibility that something is true or happens** Ex) Anybody can make mistakes.
@probability	**(practical) probability that something is true or happens** Ex) That would be his mother. Ex) He must be lying.
@may	**Practical possibility that something is true of happens** Ex) It may be true. Ex) It could be.
Expectation and regretfulness	
@expectation	**Expect something to happen** Ex) Children ought to be able to read by the age of 7.
@regret	**regretful feeling** Ex) I'm sorry to hear that you have to leave.
@wish	**wishful feeling, wish something was true or had happened** Ex) If only I could remember his name! (~I `do wish I could remember his name!) Ex) You might have just let me know.
Intention	
@insistence	**Strong will to do something** Ex) He will do it, whatever you say.
@intention	**intention on something or to do something** Ex) He shall get this money. (speaker's intention) Ex) We shall let you know our decision.
@will	**will to do something** Ex) I'll write as soon as I can. Ex) We won't stay longer than two hours.

4.7 Convention

Typical UNL structures can be expressed by attributes, to avoid the complexity of enconverting and deconverting. These attributes do not express speaker's information.

@pl	Plural	Ex) These (this.@pl) are the wrong size.
@angle_bracket	< > is used	
@double_parenthesis	(( )) is used	
@double_quotation	" " is used	
@parenthesis	( ) is used	
@single_quotation	' ' is used	
@square_bracket	[ ] is used	

Chapter 5: Format of UNL

5.1 UNL Document

Information is provided in UNL documents. The UNL document has the following format.

<UNL document> ::= "[D:" <dinf> "]" { "[P]" { "[S:" <number> "]" <sentence> "[/S]" }... "[/P]" }... "[/D]"

<dinf> ::= <document name> "," <owner name> ["," <document id> "," <date> "," <mail address>]

<document name> ::= "dn=" <character string>

<owner name> ::= "on=" <character string>

<document id> ::= "did=" <character string> /* defined by system */

<date> ::= "dt=" <character string> /* defined by system */

<mail address> ::= "mid=" <character string> /* defined by system */

<sentence> ::= "{org:" <l-tag> ["=" <code>] "}" <source sentence> "{/org}" "{unl" [":" <uinf>] "}" <UNL expression> "{/unl}" "{" <l-tag> ["=" <code>] [":" <sinf> "]" <generated sentence> "{/" <l-tag> "}"
/* necessary information about one sentence */

<l-tag> ::= "ab" | "cn" | "de" | "el" | "es" | "fr" | "id" | "hd" | "it" | "jp" | "lv" | "mg" | "pg" | "ru" | "sh" | "th" /* language flag */

<code> ::= <character code name>

<character code name> ::= <character string>

<source sentence> ::= <character string>

<generated sentence> ::= <character string>

<uinf> ::= <system name> "," <post editor name> "," <reliability> ["," <date> "," <mail address>]

<sinf> ::= <system name> "," <post editor name> "," <reliability> ["," <date> "," <mail address>]

<system name> ::= "sn=" <character string>

<post editor name> ::= "pn=" <character string>

<reliability> ::= "rel=" <digit>

<number> ::= <digit> /* sentence number */

Tags used in above definition are the following.

[D:<dinf>]	indicates the Beginning of a document and the necessary information about the document
[/D]	indicates the End of a document
[P]	indicates the Beginning of a paragraph
[/P]	indicates the End of a paragraph
[S:<number>]	indicates the Beginning of a sentence and the sentence number
[/S]	indicates the End of a sentence
{org:<l-tag>=<code>}	indicates the Beginning of an original/source sentence, language and character code, "=<code>" can be omitted.
{/org}	indicates the End of an original sentence
{unl:<uinf>}	indicates the Beginning of the UNL expressions of a sentence and necessary information, ":<uinf>" can be omitted.
{/unl}	indicates the End of the UNL expressions of a sentence

About <UNL expression> see the following section.

5.2 UNL Expression

UNL expressions of a sentence are identified with the following tags: {unl} and {/unl}.

There are two forms for expressing the UNL binary relations, one is table form and the other is list form. Table form of UNL expression is more readable than list form, but list form of UNL expression is more compact than table form.

Any component such as a word, a phrase or a title and, of course, a sentence of a native language can be represented with UNL expressions. An UNL expression therefore consists of an UW or a (set of) binary relation(s). In the UNL documents, a set of UNL expressions for a sentence is enclosed by the tags {unl} and {/unl} inside [S] and [/S]. In case of an UW, the UW need to be enclosed further by the tags [W] and [/W].

Thus, the UNL expressions of a sentence are the following:

```
{unl}
<Binary Relation>
...
{/unl}

or,

{unl}
[W]
<UW>
[/W]
{/unl}
```

Each tag and binary relation should end with a return code: “0x0a”.

5.2.1 Table form of UNL expression

Syntax of binary relation

<Binary Relation>	::= <Relation Label> [“:”<Compound UW-ID>] “(“ {{ <UW$_1$> [":" <UW-ID$_1$>]} | { “:” <Compound UW-ID$_1$> }}[<Attribute List>] “,” {{ <UW$_2$> [":" <UW-ID$_2$>]} | { “:” <Compound UW-ID$_2$> }}[<Attribute List>] “)”
<Relation Label>	::= a relation, see “Chapter 2: Relations”
<UW>	::= an UW, see “Chapter 3: Universal Words”
<Attribute List>	::= { “.” <Attribute> } …
<Attribute>	::= an attribute, see “Chapter 4: Attributes”
<UW-ID>	::= two characters of ‘0’ – ‘9’ and ‘A’ – ‘Z’
<Compound UW-ID>	::= two-digit decimal number (00 – 99) 00 is used for representing the main sentence, which can be omitted.

Compound UW-IDs are strings of two digits used to identify each instance specified by Compound UWs. Compound UWs are groups of binary relations (so-called “Scope-Nodes”) that can be referred to as a UW.

For instance, the following shows an example of UNL expressions of the sentence “I can hear a dog barking outside”.

```
{unl}
aoj(hear(icl>be(obj>thing)).@entry.@ability, I)
obj(hear(icl>be(obj>thing)).@entry.@ability, :01)
agt:01(bark(icl>do).@entry, dog(icl>animal))
plc:01(bark(icl>do).@entry, outside(icl>how))
{/unl}
```

In the above UNL expressions, “**aoj**”, “**agt**” and “**obj**” are the **relation labels**, “**hear(icl>be(obj>thing))**”, “**I**”, “**bark(icl>do)**”, “**dog(icl>animal)**” and “**outside(icl>how)**” are the **UWs**, “**:01**” appeared three times shows the Compound UW-ID. The Compound UW-ID in the position of an UW, so-called the “**scope-node**”, is used to cite or refer to a Compound UW previously defined. The binary relations indicated by the Compound UW-ID show or define the contents (of UNL expressions) of the scope. A scope-node always begin with “:” followed by the two digits of a Compound UW-ID.

The UW-IDs are omitted from the above UNL expressions. It is possible when an UW is distinguishable from the others without the UW-ID.

The UW-ID is used to indicate some referential information: that there are two or more different occurrences of the same concept (they are not co-referent). Normally, if the same UW occurs more than once, it is in all cases understood to refer to the same entity or occurrence. For example, if one man greeted another man, the same UW would be used twice -- "man(icl>person)" and we could distinguish one from the other with UW-IDs:

man(icl>person):01 for the first and
man(icl>person):02 for the other, to make it clear that the first man did not greet himself.

5.2.2 List form of UNL expression

List form of UNL expression consists of UW set and encoded binary relations of a sentence.

```
{unl}
[W]
{<UW> | {":"<Compoun UW-ID>}}":"<Node-ID>    /* node identification */
…
[/W]
[R]
<Encoded Binary Relation>
…
[/R]
{/unl}
```

Tags used above have the following meanings.

[W]	indicates the Beginning of Node identification
[/W]	indicates the End of Node identification
[R]	indicates the Beginning of encoded binary relations
[/R]	indicates the End of encoded binary relations

Each tag, encoded binary relation, node identification should end with a return code: "0x0a".

Syntax of encoded binary relation

```
<Encoded Binary Relation>    := <Node1-ID><Relation Label>[":"<Compound UW-ID>]<Node2-ID>
<Node-ID>                    := two characters of '0' – '9' and 'A' – 'Z'
```

For instance, the following shows an example of list form of the UNL expressions of the sentence "I can hear a dog barking outside".

```
{unl}
[W]
I:01
hear(icl>be(obj>thing)).@entry.@ability:02
dog(icl>animal):03
bark(icl>do).@entry:04
outside(icl>how):05
:01:06
[/W]
[R]
02aoj01
02obj06
04agt:0103
04plc:0105
```

[/R]
{/unl}

Appendix 1: List of Relation Labels

agt	Agent	a thing in focus which initiates an action
and	conjunction	a conjunctive relation between concepts
aoj	Thing with attribute	a thing which is in a state or has an attribute
bas	Basis	a thing used as the basis(standard) for expressing degree
ben	beneficiary	a not directly related beneficiary or victim of an event or state
cag	co-agent	a thing not in focus which initiates an implicit event which is done in parallel
cao	co-thing with attribute	a thing not in focus, as in a state in parallel
cnt	content	an equivalent concept
cob	effected co-thing	a thing which is directly effected by an implicit event done in parallel or an implicit state in parallel
con	condition	a non-focused event or state which conditions a focused event or state
coo	co-occurrence	a co-occurrent event or state for a focused event or state
dur	duration	a period of time during which an event occurs or a state exists
fmt	Range	a range between two things
frm	Origin	an origin of a thing
gol	goal/final state	the final state of an object or the thing finally associated with an object of an event
ins	Instrument	the instrument to carry out an event
man	Manner	the way to carry out an event or characteristics of a state
met	Method	the means to carry out an event
mod	Modification	a thing which restrict a focused thing
nam	Name	a name of a thing
obj	effected thing	a thing in focus which is directly effected by an event or state
opl	effected place	a place in focus where an event takes effect
or	Disjunction	a disjunctive relation between two concepts
per	proportion, rate of distribution	a basis or unit of proportion, rate of distribution
Plc	Place	the place an event occurs or a state is true or a thing exists
plf	initial place	the place an event begins or a state becomes true
Plt	final place	the place an event ends or a state becomes false
pof	part-of	a concept of which a focused thing is a part
pos	Possessor	the possessor of a thing
ptn	Partner	an indispensable non-focused initiator of an action
pur	purpose or objective	the purpose or objective of an agent of an event or the purpose of a thing which exists
qua	quantity	quantity of a thing or unit
rsn	Reason	a reason that an event or a state happens
scn	Scene	a virtual world where an event occurs or state is true or a thing exists
seq	Sequence	a prior event or state of a focused event or state
src	Source/initial state	the initial state of an object or thing initially associated with the object of an event
tim	Time	the time an event occurs or a state is true
tmf	initial time	the time an event starts or a state becomes true
tmt	final time	the time an event ends or a state becomes false
to	destination	a destination of a thing
via	intermediate place or state	an intermediate place or state of an event

Appendix 2: List of Attribute Labels

(to be prepared)

Appendix 3: syntax definition notation

Symbol	Definition
::=	to indicate the left is defined as the right
|	to indicate two disjunctive elements: “or”
[ ]	to indicate an optional element
{ }	to indicate an alternative element
…	to indicate repetition of the previous element, 0 or more than 1 time
“ ”	to enclose a string of literal characters
< >	to indicate a variable name