

[from (Dec 2005): <http://www.unl.org/publications/gm/index.htm>]

[United Nations University, dated 1999]

---

# A Gift for a Millennium

Hiroshi Uchida

Meiying Zhu

Tarcisio Della Senta

---

The United Nations University is an autonomous academic institute which was established by the United Nations General Assembly in 1975.

The University carries out works on the pressing global problems of human survival development and welfare through a network of research and postgraduate training centers and cooperating institutions in both industrialized and developing countries.

Established in 1995, the UNU/Institute of Advanced studies (UNU/IAS) is an advanced research and education institution with a flexible and multi-thematic programme orientation concerned with the interactions of social and natural systems. For the initial years, the UNU/IAS programme is directed at sustainable development issues. To this end, the UNU/IAS is currently active in the following relevant areas of research and education: Eco-restructuring for Sustainable Development; Megacities and Urban Development; Multilateralism and Governance; and Science, Technology and Society.

---

The UNL, a Gift for a Millennium  
November, 1999

**ISBN: 4-906686-06-0 C**

Institute of Advanced Studies  
The United Nations University  
53-67. Jingumae 5-chome, Shibuya-ku, Tokyo 150-8304, Japan  
Tel: 81-3-5467-5547 Fax: 81-3-5467-2324

UNL home page <http://www.unl.ias.unu.edu/>  
Inside the cover: The United Nations University Headquarters in Tokyo.

### Copy rights

All right reserved. No part, or the totality, of this publication can be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the prior written permission of the UNU/ IAS, nor be otherwise circulated in any form of binding or cover other than that in which is published and without a similar condition being imposed on the subsequent purchaser.

## Introduction : Invitation to a Global Solidarity Endeavor

---

Sharing information and providing a common educational environment for all is an often-stated ideal. Language is an essential factor for this. Sharing of information among countries is a crucial part of the work of international organizations such as the United Nations, UNESCO, EU, and many others. These organizations are concerned about providing equal opportunities to all member states. Many nations, however, have much less opportunities to learn about the activities of the international organizations, despite the great significance of these activities for them. At present, most information materials, scientific, technical and educational, are written in English or in few other languages. While this benefits millions of people, millions more are denied access to these information materials because they do not speak the required languages.

The smoothest way to communicate with other people and obtain information and education, is in one's own mother language. Smooth communication among people with different language will improve mutual understanding.

For this purpose, we are introducing the UNL. UNL will provide a common communication environment for different language. Furthermore, UNL will expand education and business opportunities around the world. Mutual understanding among different cultures is one of the ultimate goals of UNL.

The UNL is still at an early stage, but the infrastructure and the architectural design is available for a collaborative work of scholars, developers and providers from any language. The development of the UNL offers a unique opportunity for a genuine global solidarity endeavor. Our common well-defined purpose of working together for a goal that is beneficial to all. To achieve it, depends on the vast intellectual resources available in all languages. This is an invitation for universal win-win collaboration that meets the ideals of the UN, UNESCO, EU, and many other international organizations that care about equal opportunities for all peoples. It is a call for a global partnership for a very targeted, attainable, tangible and long-lasting communication facility.

The UNDL Foundation will provide the means to accomplish such an ambitious collaborative win-win endeavor. Starting in January 2000, the UNL resources will be made available to all those who are interested in joining the collective task. The UNL resources will be provided free of charge. This is an invitation to undertake a collaborative research and development to offer all peoples of the world a gift for a millennium.

Hiroshi Uchida  
Meiyong Zhu  
Tarcisio G. Della Senta  
UNU/IAS

Tokyo, November 1999



## Preface : Acknowledgements

---

Launching a major endeavor such as the UNL requires the combined knowledge and talent of many people. Behind these people, there are a number of research institutions and traditional universities. In introducing this ambitious initiative to the public, and bringing the Specifications of the UNL system into the open, I would like to acknowledge the value of years of scientific research. I would like to thank the research teams for their highly motivated commitment. The following institutions joined the UNL initiative as partners in the first hour, in November 1996.

The Royal Scientific Society (Jordan)  
Research Center of Computer & Microelectronics Industrial Development (CCID),  
Ministry of Information Industry (MII) Peoples Republic of China  
GETA-CLIPS, Universite Joseph Fourier & CNRS (France)  
Institute for Applied Information Sciences (Germany)  
Indian Institute of Technology, Bombay (India)  
Directorate for Technology in Informatics and Electronics, Agency for The Assessment and Application of Technology (BPP Teknologi) (Indonesia)  
Istituto di Linguistica Computazionale del CNR, Pisa (Italy)  
Institute of Mathematics and Computer Science of the University of Latvia (Latvia)  
Mongolian State Pedagogical University of Mongolia (Mongolia)  
University of Sao Paulo, at Sao Carlos (Brazil)  
Russian Academy of Sciences, Institute for Information Transmission Problems (Russia)  
STAR SPB Ltd, Saint Petersburg (Russia)  
Universidad Politecnica de Madrid (Spain)  
National Electronics and Computer Technology Center (NECTEC) (Thailand).

A special Task Force on the UNL Specifications met several times before consolidating the first version of the Specifications that are presented in this edition. The Task Force included: Hiroshi Uchida (UNL Center, head of Task Force), Pushpak Bhattacharyya (UNL-India), Christian Boitet (UNL-France), Igor Boguslavsky (UNL-Moscow, Russia), Christian Boitet (UNL-France), Mike Dillinger (UNL-Brazil), David Escorial (UNL-Spain), Daoud Maher (UNL-Jordan), Luis Iraola Moreno (UNL-Spain), Irina Prodanof (UNL-Italy), Joerg Schuetz (UNL-Germany), Virach Sornlertlamvanich (UNL-Thailand), Oliver Streiter (UNL-Germany), M. Tomokiyo (UNL, France), Meiyong Zhu (UNL Center).

I am particularly grateful to Dr. Kazuhiko Nishi for his vision of the treasure for humankind that UNL represents. Thanks are also due to Prof. Takenori Makino, the brain that produced the NEC supercomputer and a computational linguistics pioneer who was associated with the launching of this initiative. He also drafted parts of this publication.

My special gratitude goes to the many scientists and partners who believe that UNL is possible and engage their competence to realize it.

Tarcisio G. Della Senta  
Director UNU/IAS

# 1 Global Internet Infrastructures

Only a few scientists and government officials could use a small text-base computer network in early 1980. An estimated 150 million Internet users worldwide are supported by interconnected networks in 1999.

The growing Internet infrastructures are dramatically changing society, business and individual life styles. Electronic mail has become a popular communication facility, ranking with telephone. Thanks to the Internet, the e-mail has become a new fashion in human relations. Shopping malls on the Internet are jammed with customers. In the stock market, many investors perform transactions through the Internet. These communication facilities could not be imagined just a few years ago.

## 1.1 Expansion of Internet Infrastructures

In 1999, more than 97% of all Internet hosts were in developed countries, corresponding to only 16% of the world population. The international structure of the Internet has been US-centric. Moves toward expanding international Internet networks, however, are growing in each region of the globe. The European Internet infrastructure has become interconnected with high-capacity bandwidth. Asian countries began to investigate the conditions to form an intra-Asia network. Such moves will change Internet infrastructure dramatically over a short period.

In developing countries, one common problem is the high cost of Internet services and its physical setup. Wireless local networks are often mentioned as the most economical and feasible solution to the severe shortage of communication infrastructure in these countries. Furthermore, opto-electronic technology and optical networking will become the key enablers of the future communications infrastructure by eliminating the severe restrictions imposed by traditional communication systems.

As network technologies have been improving and expanding, the Internet has begun to spread throughout many nations. In some countries, it is maintained under the control of central governments. In most countries, however, the Internet is growing boosted by open competition and market forces. Although the Internet structure has been initially US-centric, regional and local networks are interlinked worldwide. Internet infrastructures of the next generation have been strengthened with governmental support, market demand, and state-of-the-art technologies.

## 1.2 Reshaping Community Boundaries Worldwide

Fostered by the growth of the Internet, information networks are transforming the world communications scenario. In the past, these networks were used primarily to carry electronic mail. Initially, networks such as ARPA-NET

were used by university researchers to exchange scientific data. Companies then began to use these networks, not only for their internal communications, but also for communicating with other companies, thus creating a common working environment, which became a necessary part of the way of doing business. They now include content services, on-line shopping, and others services that, so far, have been provided through Value Added Networks (VANs). Next, the networks entered family homes, where they are used for recreational and domestic purposes. The global village is becoming more interconnected, and community boundaries are being re-defined.

Today, the Internet is providing a larger platform for communication among people across national frontiers. Individuals send their information to the world independent of the traditional mass media. Product developers can sell their products to end-users without salespeople and stores. People can download their desired books from digital libraries around the world, from their homes. They can read digital newspapers using information filtered on their computer. New forms of relationships are being established among people who have never met each other. Certainly, the Internet has broken various barriers in communication. The remaining barriers that subsist are between different languages. Although demand for e-mail applications is still large, the uses of the Internet are becoming more diverse.

As networks are expanding around the world, and as the physical barriers to their development are surmounted, the language barriers will increase in prominence. Even now, a flood of information is provided through the networks, written in a number of languages. But, the language barrier hinders access to this information. Most of the information in the world is written in the author's mother tongue. There is relatively little information written for multiple languages readers. Information for worldwide use is often written in English or in a few other well-disseminated languages. The United Nations and other international organizations produce their documentation in 6 official languages: Arabic, Chinese, English, French, Russian and Spanish. For millions of people, however, whose mother tongue is not one of these languages, it is difficult to take full advantage of the information produced by these organizations. Conversely, information that is not written in one of the dominant languages is seldom made available in these six languages, much less in others.

## 2 Breaking down the Language Barrier

In this century, English is regarded as the most common communication language worldwide. Non-English speakers spend money and time to learn it. Nevertheless, they hardly ever acquire perfect English as a native speaker. Thus, they remain somewhat handicapped in communicating in English. In addition, English itself does not reach billions of people because they do not have the resources for learning it. Hence, although English has become



popular in business, entertainment, science and government-related matters, a communication gap persists for non-English speakers.

To make things worse, the reverse is also true: a gap exists between English speakers and the information and culture in other peoples. They have little chance to have access to languages, such as, Chinese, Hindi, Japanese, Arabic, and many other cultural and otherwise rich languages.

In order to overcome the language barrier, many attempts have been made in the past. In the colonization era, metropolitan powers forced language education in colonized territories. Local people had to learn foreign languages in order to survive and develop. Several major languages spread worldwide in that era, Spanish, Portuguese, French, English and others. Many nations in these territories lost their traditional culture, their own language, and with that, their identity. A language is the basis for culture and social identity. Every language is a "local" language. To communicate with others, people have to learn the other language. Esperanto was the first attempt at intermediating communication among local languages and becoming a world common language. It was accepted by only a few people because it is just one local language itself.

Professional translators have been bridging such a communication gap. The quantity of translation by humans, however, is rather small as compared to the required communication needs among different languages. The main reason for such a limitation is high costs. In addition, the number of translators for minor languages is rather small. Translation by humans, thus, has its limitation in terms of cost and human resources.

Since the first electronic computer was developed in 1946, many hoped that human intelligence could produce translation through machines. Translations with computers emerged as an interesting research theme. Georgetown University and IBM demonstrated a small computer translation system between Russian and English in the early 1960's. Since then, there is a long history of research on computer translation. Some companies developed commercial computer translation systems that are available on PC.

## **2.1 Computer Translation Systems**

Computer translation systems made significant progress. Some of them are now being incorporated in network browsers. The demand for these systems indicates how large the language problem is among Internet users. Computer translation systems are useful under limited conditions. For instance, the user can evaluate and modify a translated document in one's own language, but seldom in the other language. However, after translating with a computer, the user has to work to edit the output document. In addition, it would require language knowledge to edit the translation of the document in the other language. In sending information throughout the world, the sender normally does not know the language of the recipient. In this case, the sender is bound to use a computer translation system blindly, because he can't check whether the translated results are correct or not. This is a serious limitation in current computer translation systems, and explains, in part, their limited acceptance.

The difficulty in these translation systems lies in the language analysis process to be performed by the computer. In analyzing a sentence in its semantic representation, the computer has to discriminate the lexical and syntactic ambiguities, and then, derive the correct semantic representation. There are many problems to be solved in these processes. On the other hand, a language synthesis from the semantic representation also has a few problems, although some of the most difficult ones are related to the generation of elegant sentences.

This situation will change dramatically with the UNL.

## **2.2 A Breakthrough with UNL**

UNL is an electronic language for computers. It intermediates understanding among different natural languages. UNL represents sentences in the form of logical expressions, without ambiguity. These expressions are not for humans to read, but for computers. It would be hard for users to understand, and they would not need to, unless they are UNL experts. Thus, UNL is an intermediate language to be used through the Internet, which allows communication among people of different languages using their mother tongue.

Adding UNL to the network platforms will change the existing communication landscape. The purpose of introducing UNL in communication networks is to achieve accurate exchange of information between different languages. Information has to be readable and understandable by users. Information expressed in UNL can be converted into the user's native language with higher quality and fewer mistakes than the computer translation systems. In addition, UNL, unlike natural language, is free from ambiguities.

## **2.3 UNL: a Unique Opportunity for All**

Sharing information and providing a common educational environment for all is an often-stated ideal. Language is an essential factor for this. At present, however, most information, scientific and educational materials are written in English or in few other languages. This benefits millions of people. Yet, non-speakers of these languages have to learn them to access these materials. However, it would be rather smoother to get education in one's own mother language.

Sharing of information among countries is a crucial part of the work of international organizations such as the United Nations and UNESCO. The staff working in these organizations are concerned about all peoples of the world. But these have few opportunities to learn about the activities of the international organizations, despite the great significance of these activities for them. One of the reasons for this lack of information is the language barrier.

Providing equal economic opportunities is essential for business. Economic interaction among countries has increased with the globalization of the

economy. A crisis in Asia shook the economies all over the world. In business, it is important to quickly know what is happening worldwide. Today, various kinds of customers from different languages are accessing Internet shops. These Internet shops would have great advantage in preparing information in many languages at once. Customers can freely access various web-pages worldwide to buy different goods. They would prefer to read web-pages written in different languages in their own.

UNL will provide a common educational environment to different languages. Furthermore, UNL will expand business opportunities immensely around the world. Mutual understanding among different cultures is one of the ultimate goals of UNL. Smooth communication through UNL among people with different languages will support and improve mutual understanding. UNL will provide a common educational environment to different languages.

# 3 UNL-The Universal Networking Language-

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 as 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 which 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.

## 3.1 UNL Expression

Binary relations are the building blocks of UNL expressions. They are made up of a relation and two UWs. This section deals with the definition and interpretation of the relations that are used as the basis 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.

### 3.1.1 Internal Structure of Binary Relations

Binary relations are made up as follows:

```
<Binary relation> ::= <Relational Label> [ ":" <Compound UW-ID>
]
      "(" { <UW1> ":" <UW-ID1> } | ":" <Compound
UW-ID1> } ","
      {<UW2> ":" <UW-ID2> } | ":" <Compound
UW-ID2> } ")"
```

These elements will be defined in the paragraphs below.

Example binary relations are:

```
mod:01(area(icl>place):02.@indef, strategic)
obj(designate(icl>do).@entry.@may, :01)
plc(read(icl>do), home)
```

<b>Meta-symbols for description:</b>	
< >	non-technical symbol / variable

" "	enclosed string is literal characters
::=	... is defined as ...
	disjunction, "or"
[ ]	optional element
{ }	alternative element
{ }...	to be repeated more than zero times

### 1)Relation Labels

Relation labels are strings of two or 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>do).@entry.@may, :01)
plc(read(icl>do), home)

```

### 2)Compound UW-IDs

Compound UW-IDs are strings of two digits used to identify each Compound UWs. Compound UWs are groups of binary relations (called "Hyper-Nodes") which can be referred to as an UW. Examples are the elements in bold face type below. The first example is an instance of compound UW-IDs being used to define a unit; the second example is an instance of Compound UW-IDs being used to cite or refer to a Compound UW previously defined. See 3.2.2 for further information.

```

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

```

Note that the ":02" in the first example is not a Compound UW-ID but an UW-ID. A Compound UW-ID is either attached directly to Relation Labels or appears alone, as UWs. See 3.2.1 for further information.

### 3)UWs

UWs can be UWs or compound UWs. 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)
plc(read(icl>do), home(icl>place))
obj(designate(icl>do).@entry.@may,:01)

```

## 3.1.2 Format of UNL

### 1)UNL Document

The structure of a UNL document is expressed using the following tags.

[D] Beginning of document  
 [/D] End of document  
 [P] Beginning of paragraph  
 [/P] End of paragraph  
 [S] Beginning of sentence  
 [/S] End of sentence

UNL documents are generally constructed in the following manner.

[D] Beginning of document  
 [P] Beginning of paragraph  
 [S] Beginning of sentence  
 ... UNL expressions  
 [/S] End of sentence  
 ... Repetition of [S]...[/S]  
 [/P] End of paragraph  
 ... Repetition of [P]...[/P]  
 [/D] End of document  
 ... Repetition of [D]...[/D]

## 2)UNL Expression

An UNL expression is identified with the following tags:

{unl} Beginning of UNL expression  
 {/unl} End of UNL expression

There are two kinds of UNL expression, 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. In UNL expression, there are three types of information, namely binary relations, UWs, and encoded binary relations. The following tags are used to distinguish this information.

[W] Beginning of UW set  
 [/W] End of UW set  
 [R] Beginning of binary relations  
 [/R] End of binary relations

<Binary Relation> ::= <Relation Label> [ ":" <Compound UW-ID> ]  
 "(" { <UW1> ":" <UW1-ID> } | { ":" <Compound  
 UW-ID1> } ", "  
 { <UW2> ":" <UW2-ID> } | { ":" <Compound  
 UW-ID2> } ")"

<UW> ::= <Head Word> [ <Constraint List> ] [ ":" <UW-ID> ] [ "."  
 <Attribute List> ]

<Encoded Binary Relation> := { <UW-ID> | <Compound UW-ID> }

<Relation Label> [":" <Compound UW-ID>] {<UW-ID> |  
<Compound UW-ID>}

### 3) Table form of UNL expression

The table form of UNL expression consists of binary relations or only one UW.

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

or

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

### 4) List form of UNL expression

The list form of UNL expression consists of UWs and encoded binary relations.

```
{unl}  
[W]  
...  
[/W]  
[R]  
...  
[/R]  
{/unl}
```

Each tag, binary relation, UW, and encoded binary relation should be separated with carriage return (0x0a, or 0x0d 0x0a).

Sample of UNL expression in list form.

#### Ex.1) Monkey eats bananas.

```
[S]  
{unl}  
[W]  
eat(icl>do).@present.@entry:00  
monkey(icl>animal).@generic:01  
banana(icl>food).@generic:02  
[/W]  
[R]  
00agt01  
00obj02  
[/R]  
[/S]
```

**Ex.2) UNL is a common language that would be used for network communications.**

```
[S]
{unl}
[W]
language(icl>abstract thing).@present.@entry:00
UNL(icl>language).@topic:01
common(aoj>thing):02
use(icl>do).@present:03
language(icl>abstract thing).@present.@entry:04
communication(icl>action).@pl:05
network(icl>thing):06
[/W]

[R]
00aoj01
00mod02
03obj04
03pur05
05mod06
[/R]
[/S]
```

### 3.2 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.

#### 3.2.1 Syntax of UW

UWs are made up of a character string (an English-language word) followed by a list of constraints and a list of attributes. These can also be followed by an Instance ID. 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>] [ ":" <IUW-ID> ] [ "."
<Attribute List> ]
<Head Word> ::= <character>...
<Constraint List> ::= "(" <Constraint> [ "," <Constraint> ]... ")"
<Attribute List> ::= <Attribute Label> [ "." ]...
<UW-ID> ::= { <upper case alphabetical character> | <digit> }
{ <upper case alphabetical character> | <digit> }
<Constraint> ::= <Relation Label> { ">" | "<" } <UW> [<Constraint List>]
```



```

|
<Relation Label> { "<" | ">" } <UW> [<Constraint List>]
[ { ">" | "<" } <UW> [<Constraint List>] ] ...
<Attribute Label> ::= "@volitional" | "@reason" | "@past" | ....
<Relation Label> ::= "agt" | "and" | "aoj" | "obj" | "icl" | ...
<digit> ::= 0 | 1 | 2 | ... | 9
<upper case alphabetical character> ::= "A" | ... | "Z"
<character> ::= "A" | ... | "Z" | "a" | ... | "z" | " " | " " | " " | "#" | "!" | "$" | "%" |
"=" | "^" | "~" | "|" | "@" | "+" | "-" | "<" | ">" | "?" | ""

```

### 1) Head Word

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. An 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.

### 2) 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", 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.

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".

Constraints can use Relation Labels. Each constraint in the Constraint List should be sorted in alphabetical order.

When the relation label is omitted, it is assumed that the left most relation is omitted. For example, xxx(icl>change(icl>do)) can be replaced with xxx(icl>change>do).

### 3) Attributes

The Constraint List can be followed by a list of attributes, defined in section 3.4, which provide information about how the concept is being used in a particular sentence.

### 4) UW-ID

A UW can include an UW-ID. The UW-ID is simply used to indicate some referential information: that there are two 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.

### 3.2.2 Types of UWs

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(agt>person,obj>information)  
state(equ>nation)  
state(icl>situation)  
state(icl>government)

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

ikebana(icl>action,obj>flowers)  
samba(icl>dance)  
soufle(icl>food,pof>egg)  
murano(icl>glass,aoj>colorful)

#### 1)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 fall-back method for establishing correspondences between different language words when more specific correspondences cannot be found.

#### 2)Restricted UWs

Restricted UWs are by far the most important. Each Restricted UW denotes a subset of the concept that may correspond to that in English defined by its Constraint List. Each Restricted UW represents a more specific concept. Consider again the examples of Restricted UWs given above:

state(agt>person,obj>information) is a more specific concept (arbitrarily associated with the English word "state") that denotes situations in which humans produce some information, or state something.

state(equ>nation) is a more specific sense of "state" that denotes a nation.

state(icl>situation) is a more specific sense of "state" that denotes a kind of situation.

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 is 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.

### 3)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 below:

ikebana(icl>activity, obj>flower) "something you do with flowers"

samba(icl>dance) "a kind of dance"

souffle(icl>food, pof>egg) "a kind of food made with eggs"

murano(icl>glass, aoj>colorful) "a kind of colorful glass"

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 already give some idea of what concept is associated with these Extra UWs, and the Constraints provide a binary relation between this concept and other concepts already present (activity, flower, egg, food, etc.).

### 4)Compound UWs

Compound UWs are a set of binary relations that are grouped together to express a 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 movie theaters" and then relate them to other concepts.

Compound UWs are indicated by Compound UW-IDs, which are made up of a colon ":" followed by two digits. Compound UW-IDs can also be followed by an Attribute List.

More formally, their syntax can be described as follows:

```
<Compound UW> ::= ":" <Compound UW-ID> [ "." <Attribute List> ]  
<Compound UW-ID> ::= { <upper case alphabetical character> | <digit> }  
{ <upper case alphabetical character> | <digit> }  
<Attribute List> ::= <Attribute Label> [ "." <Attribute Label> ] ...  
<Attribute Label> ::= "@imperative" | "@may" | "@past" | ...  
<digit> ::= 0 | 1 | 2 | ... | 9  
<upper case alphabetical character> ::= "A" | ... | "Z"
```

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.

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.

## 5) How do define Compound UWs

Compound UWs are 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), woman(icl>person).@pl)  
obj:01(wear(icl>do), hat(icl>thing))  
aoj:01(big(aoj>thing), hat(icl>thing))  
plc:01(wear(icl>do), theater(icl>place))  
mod:01(theater(icl>place), movie(icl>thing))
```

After this group has been defined, wherever ":01" is used as an UW, it means that the UW should be understood as all of these Binary relations.

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 (see 3.4.1 Speaker's focus) is necessary.

## 6) How to cite Compound UWs

Once defined, Compound UWs can be cited or referred to by simply using the Compound UW-ID as an UW. To complete the example above, we could continue with:

```
agt(ask(icl>do).@should, :01)
obj(ask(icl>do), leave(icl>do))
```

Again, ":01" is interpreted as the whole set of binary relations defined above. Compound UWs can be cited within other Compound UWs.

### 3.3 Relations

In the UNL, binary relations are represented as strings of 3 or less characters called "Relation-Labels".

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 a relation label.

The UNL relations are defined as specified below.

**Agt** defines a thing which 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(icl>do), John(icl>person)) John breaks

agt(translate(icl>do), computer(icl>machine)) computer translates ...

agt(run(icl>do), car(icl>thing)) car runs ...

agt(break(icl>do), 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 focussed initiator of the action, whereas the partner is a non-focussed initiator.

Agent is different from **con** in that agent is the focussed 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(dream(icl>do), think(icl>do)) ... to think and to dream

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 which is in a state or has an attribute.

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

**aoj (thing, 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 represents 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(icl>thing)) leaf is red  
aoj(available(aoj>thing), book(icl>thing)) book is available  
aoj(nice(aoj>thing), ski(icl>event)) Skiing is nice  
aoj(teacher(icl>thing), 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>thing))  
aoj(know(aoj>thing,obj>thing), John(icl>person)) John knows ...  
aoj(can(aoj>thing,obj>thing), I) I can ...  
aoj:01(difficult(aoj>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 focussed 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 thing which is directly affected by action or phenomenon, whereas, **aoj** defines thing in a state.

**bas (basis for expressing degree)**

**Bas** defines a thing used as the basis 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", "like", "as much as", "at least", etc., and
- UW2 is some thing used as the basis for evaluating characteristics or quantity of some other (focussed) thing.

**Examples and readings**

bas(more, rat(icl>thing)) ...er than rat; more ... than rat  
bas(like, star(icl>thing)) ... like star  
bas(same, b(icl>thing)) ... the same as b  
bas(at least, 12) ... at least 12  
aoj(beautiful(aoj>thing), tulip(icl>thing)) tulip is more beautiful than rose  
man(beautiful(aoj>thing), more)  
bas(more, rose(icl>thing))  
aoj(:01, John(icl&t;person)) John is more quiet than shy  
man:01(quiet(aoj>thing), more)  
bas:01(more, shy(aoj>thing))

## 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(icl>do), Mary(icl>person)) John give ... for Mary.  
agt(give(icl>do), John(icl>person))  
ben(good(aoj&g;thing), John) 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 a 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 which initiates an implicit event which 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)) ... walk with John

cag(live(icl>do),aunt(icl>person)) ... lives 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, as 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

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(aoj>thing), you) ... is here 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>language), universal networking language(icl>language) UNL,  
Universal Networking Language

cnt(internet(icl>network), amalgamation(icl>thing)) Internet: an  
amalgamation

cnt(language generator(icl>tool), deconverter(icl>tool).@double\_quotation) a  
language generator "deconverter"...

## **cob (affected co-thing)**

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

### **Examples and readings**

cob(die(icl>occur), Mary(icl>person)) ... dead with Mary

cob(have(aoj>thing,obj>thing), pencil(icl>thing)) ... have a pen with a pencil

obj(have(aoj>thing,obj>thing), pen(icl>thing))

### **Related relations**

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 which 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(green(aoj>thing), light (icl>thing)) If light is green, ... go  
con(go(icl>do), :01)

## 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>do), cry(icl>do)) ... run while crying  
 coo(red(aoj>thing), hot(aoj>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 (do, period)**

**dur ((aoj>thing), period)**

*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 state, and

UW2 - a period that the event or state continues.

*Examples and readings*

dur(work(icl>do), hour(icl>period)) ... work nine hours

qua(hour(icl>period),9)

dur(talk(icl>do), meeting(icl>event)) ... talk during meeting

dur(come(icl>do), absence(icl>state)) ... come during ... 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)) ... from a to z

fmt(Osaka(icl>place), New York(icl>place)) ... from Osaka to New York

fmt(Monday(icl>time), Friday(icl>time)) ... 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(man(icl>person), Japan(icl>country)) ... man from Japan

## gol (goal:final state)

**Gol** defines the final state of an object or the thing finally associated with an 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(go(gol>place), sad(aoj>person)) ... go ... to sad

gol(change(gol>thing), red(aoj>thing)) ... change ... to red

gol(transform(gol>thing), strong(aoj>thing)) ... is transformed ... to strong

gol(post(gol>thing), account(icl>place)) ... post ... to 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 the 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 which is used in order to make UW1 happen.

#### *Examples and readings*

ins(look(icl>do), telescope(icl>thing)) ... look ... with telescope

ins(solve(icl>do), pencil(icl>thing)) ... solve ... using pencil

ins(separate(icl>do), knife(icl>thing)) ... separate ... with knife

#### *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 the way to carry out an 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(look(icl>do), quickly) ... look quickly  
 man(think(icl>do), often) ... think often ...  
 man(beautiful(aoj>thing), very) 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 of means)**

**Met** defines the means to carry out an event.

**met (do, abstract thing)**

***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,  
 where:

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

***Examples and readings***

met(solve(icl>do), dynamics(icl>abstract thing)) ... solve ... with dynamics  
 met(solve(icl>do), algorithm(icl>abstract thing)) ... solve ... using algorithm  
 met(separate(icl>do), cut(icl>do)) ... 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 which 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 which restricts UW1 in some way.

### **Examples and readings**

mod(story(icl>thing), whole) whole story

mod(plan(icl>thing), master) master plan

mod(part(icl>thing), main) main part

qua(block(icl>thing), 3)) three blocks of ice

mod(ice(icl>thing), block(icl>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 an 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>thing), Tokyo(icl>thing)) Tokyo tower

## **obj (affected thing)**

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

### **obj (occur, thing)**

### **obj (do, 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(icl>do), table(icl>thing)) table moves

obj(melt(icl>occur), snow(icl>thing)) snow melts

obj(cure(icl>do), patient(icl>person)) cure patient

obj(have(aoj>thing,obj>thing), pen(icl>thing)) ... 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), shoulder(icl>thing)) ... pat ... on shoulder

opl(cut(icl>do), 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 define 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)) ... stay or leave

or(red(icl>color), blue(icl>color)) ... red or blue

or(John(icl>person), Jack(icl>person)) ... John or Jack

### *Related Relations*

Disjunction is different from conjunction in 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(2, day(icl>period)) ... two ... per day  
 per(time(icl>unit), week(icl>period)) ... twice a week  
 qua(time(icl>unit), 2)

## plc (place)

**Plc** defines the 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*

plc["<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(icl>thing)) ... cook ... in kitchen  
 plc(sit(icl>do), beside(icl>relative place)) ... sit beside ...  
 plc(red(aoj>thing), bottom(icl>thing)) ... red on bottom

*Related Relations*

Place is different from **plf** and **plt** or **src** and **gol** 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 **opl** is seen as being modified.

## plf (initial place)

**Plf** defines the 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(go(icl>do), home(icl>place)) ... go from home ...  
plf(call(icl>do), new york(icl>place)) ... call from New York  
plf(cut(icl>do), edge(icl>place)) ... cut ... from edge ...  
plf(beautiful(aoj>thing), side(icl>place)) ... is beautiful from side ...

#### *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 the 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(talk(icl>do), boston(icl>place)) ... talk ... until Boston  
plt(cut(icl>do), edge(icl>place)) ... cut ... to edge  
plt(beautiful(icl>state), fence(icl>thing)) ... is beautiful up to fence

#### *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*

pof(wing(icl>body), bird(icl>animal)) Bird's wing.

## **pos (possessor)**

**Pos** defines 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>thing), John(icl>person)) John's dog

pos(book(icl>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), poor(icl>person)) ... share ... with poor

ptn(collaborate(icl>do), machine(icl>thing)) ... collaborate with machine

### **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 a purpose or objective of an agent of an event or a purpose of a thing which 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)) ... come to see

pur(work(icl>do), money(icl>do)) ... work for money

pur(budget(icl>money), research(icl>do)) ... 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***

```
que["<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(block(icl>thing), 3)) three blocks of ice

mod(ice(icl>thing), block(icl>thing))

qua(kilo(icl>unit), many(aoj>thing)) many kilos ...

qua(truckload(icl>unit), 7) seven truckloads ...

#### ***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), illness(icl>thing)) go because of illness  
agt:01(arrive(icl>occur), Mary(icl>person)) Because Mary arrives, team  
collaborate  
agt:02(collaborate(icl>do), team(icl>person))  
rsn(:02, :01)
```

## **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), competition(icl>event)) ... win ... in competition

scn(appear(icl>occur), program(icl>thing)) ... appear on TV program

mod(program(icl>thing), TV(icl>thing))

scn(play(icl>do), movie(icl>thing)) ... play in movie

### **Related Relations**

Scene is different from **plc** in that the reference place for **plc** is in the real world, whereas for **scn** it is an 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, and

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>dot), look(icl>do)) ... look before leaping

seq(green(aoj>thing), red(aoj>thing)) ... was red before ... was green

### ***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 (sorce: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(go(icl>change), sad(aoj>thing)) ... go from sad ...

src(change(icl>occur), red(aoj>thing)) ... change from red

src(transform(icl>do), weak(aoj>thing)) ... is transformed from weak ...

src(steal(icl>do), bank(icl>thing)) ... steal ... from bank

### ***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 the 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(look(icl>do), Tuesday(icl>time)) ... look on Tuesday

tim(cut(icl>do), o'clock(icl>time)) ... cut ... at ... o'clock

tim(start(icl>do), come(icl>do)) ... 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(look(icl>do), morning(icl>time)) ... look since morning

tmf(full(aoj>thing), noon(icl>time)) ... is full at noon

### ***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 **tmt** expresses the time at the beginning of the event or state whereas **tmt** expresses the time at its end.

## tmt (final time)

**Tmt** defines the 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(think(icl>do), morning(icl>time)) ... think until morning

tmt(cut(icl>do), noon(icl>time)) ... cut until noon

tmt(full(aoj>thing), tomorrow(icl>time)) ... be full until 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 **tmt** 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>thing), London(icl>city)) ... train for London

## via (intermediate place or state)

**Via** defines an 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>place)) ... go ... via New York

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

via(drive(icl>do), tunnel(icl>thing)) ... 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.

### 3.4 Attributes

Attributes of UWs are used to describe 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 objective things, events and states-of-affairs in the world. Attributes of UWs enrich this description with more information about how the speaker views these states-of-affairs and his attitudes toward them.

#### 3.4.1 Types of Attributes

## 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

**@present** happening at present

**@future** will happen in future

## 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 of the event, looking forward to it (**@begin-soon**), or backward to it (**@begin-just**).

He can focus on the middle of the event (**@progress**).

He can also focus on the end of the event, looking forward to it (**@end-soon**) or backward to it from nearby (**@end-just**) or from farther away (**@complete**).

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**).

**@begin-soon**

**@begin-just**

**@progress**

**@end-soon**

**@end-just**

**@complete**

**@state**

**@repeat**

## 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  
**@def** already referred  
**@indef** non-specific class  
**@not** complement set  
**@order** ordinal number

These Attributes are usually attached to UWs that denote things.

#### 4) Speaker's Focus

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**  
**@entry** entry point or main UW of whole UNL expression or in a hyper-node  
**@qfocus**  
**@theme** instantiates an object from a different class  
**@title**  
**@topic ex)** He(@topic) was killed by her.

One UW marked with "@entry" is essential to each UNL expression and each Compound UW.

#### 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**  
**@confirmation**  
**@exclamation** feeling of exclamation  
**@imperative** imperative  
**@interrogative**  
**@invitation** inducement to do  
**@politeness** polite feeling  
considered as the way to speak. "Please, could you..."  
**@respect** feeling of respect  
considered as a particle to show the respect ("Dear sir: ...").  
**@vocative ex)** Dear(@vocative) , please hurry up !

#### 6) Speaker's viewpoint

The variety of possibilities 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 labels are used to clarify the speaker's viewpoint information.



**@ability** ability, capability of doing things

**ex)** Mary can speak French but she can not write it very well.

**@ability-past** ability in the past

**ex)** John never could play the piano.

**@apodosis-real** apodosis: reality in the first person

**ex)** We should (would) love to go fishing if we had the chance.

**@apodosis-unreal** apodosis: A supposed result from a supposition contrary to reality

**ex)** If I had more money, we could build a house.

**@apodosis-cond** apodosis: A supposed result from an assumed condition

**ex)** He would work too much if I did not stop him.

**@conclusion** He is her husband; she is his wife.

**@custom** habitual action: habit in the past

**ex)** Every morning he would take a bath.

**@expectation** expectation to other's

**ex)** He'll help her if you ask him.

Will you have another cup of coffee?

**@grant** to give consent to do

**ex)** Can I drink beer ?

You may drive my car if you like.

**@grant-not** to not give consent to

**ex)** You may not break the regulations.

**@insistence** strong will to do

**ex)** You shall go as you decide.

He shall be punished.

**@intention** will, intention to do

**ex)** He shall get this prize.

He will do it, whatever we do not want.

**@inevitability** supposition that something is inevitable

**ex)** She should be school by now.

The fight will be finished by now.

**@may** supposition of actual possibility

**ex)** The bridge may be blocked.

**@obligation** to oblige someone

**ex)** The company shall maintain the product in good repair.

**@obligation-not** forbid to do

**ex)** You need not go by 9 o'clock.

**@possibility** assume reasonable possibility

**ex)** I can speak English.

**@probability**

assume probability

**ex)** That would be his father.

**@should** to feel duty

**ex)** You should do it.

**@unexpected-presumption** presumption contrary to a wish or expectation

**ex)** It is strange that you should go there.

**@unexpected-consequence** consequence contrary to a wish or expectation

**ex)** I made a draft, but it still needs another work.

**@will** will to do

**ex)** We shall overcome language barrier..

## 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.

**@angle\_bracket** < > is used  
**@double\_parenthesis** ( ( ) ) is used  
**@double\_quotation** " " is used  
**@parenthesis** ( ) is used  
**@pl** plural  
**@single\_quotation** ' ' is used  
**@square\_bracket** [ ] is used

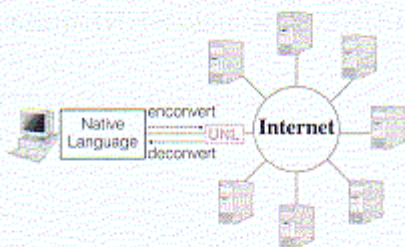
## 3.5 Knowledge Base

The UNL Knowledge Base stores possible binary relations between UWs (Universal Words). The knowledge base is a set of knowledge base entries. The format of knowledge base entries is as follows.

```
<Knowledge Base entry> ::= <Binary relations> "=" <degree of certainty>  
<Binary Relation> ::= <Relation Label> [ ":" <Compound UW-ID> ] "("  
{ <UW1> | ":" <Compound UW-ID1> } "," { <UW2> | ":" <Compound UW-  
ID2> } ")"  
<degree of certainty> ::= "0" | "1" | ... | "255"
```

When the degree of certainty is "0", it means the relation between two UWs is false. When the degree of certainty is more than "1", it means the relation between two UWs is true, and the bigger the number is, the more the credibility is.

The UW system has been introduced to reduce the number of knowledge base entries. For this purpose also the "icl" relation was introduced to make it possible to inherit properties from upper UW's.



## 4 UNL System

### -Supporting Communication among Different Languages-

The UNL system allows people to communicate with peoples of different languages in their mother tongue. The UNL is a common language to exchange information through computers which can deal with natural languages. The UNL system basically consists of language servers, UNL editors and UNL viewers.

A conversion system from native languages into UNL is called "enconverter", and one that deconverts from UNL into native languages is called "deconverter". Information "enconverted," from any language is exchanged in UNL format via networks. Information represented in UNL is "deconverted" into each native language on the terminal network.

#### 4.1 Language Server

A Language Server consists of a deconverter and an enconverter. The processes of "enconversion" and "deconversion" are provided by a Language Server which resides in the network of the Internet. The "enconverter" and "deconverter" are responsible for converting a particular language into UNL, and vice versa. The "enconverter" "enconverts" a language into UNL, while the "deconverter" "deconverts" UNL into a native language.

##### 4.1.1 Deconverter

A "deconverter" is a software that automatically deconverts UNL into native languages. It is important to achieve a high quality and correct results. It is also important that the basic architecture of the "deconverter" is widely shared throughout the world, in order to treat all languages with the same quality and precision standards. Technology developed for a language can be applied to other languages as long as the architecture is shared.

UNU/IAS developed a software for deconversion called "DeCo" which constitutes a deconverter together with a word dictionary, co-occurrence dictionary and conversion rules for a language. This "DeCo" is a language independent software that is applicable for any languages.

A "Deconverter", which generates natural language from UNL, plays a core role in the UNL system. It is very significant that "deconverter" is capable of expressing UNL information with very high accuracy. It follows that

information, once composed in UNL, can be understood in any language as far as there is a "deconverter" of the language.

#### 4.1.2 Enconverter

An "enconverter" is a software that automatically or interactively enconverts natural language text into UNL. UNU/IAS developed a software for enconversion called "EnCo" which constitutes an enconverter together with a word dictionary, co-occurrence dictionary and conversion rules for a language. This "EnCo" is a language independent software which is applicable for any languages.

An "enconverter", as it generates UNL from natural languages, enables people to make UNL documents without any knowledge about UNL. It means that users of the UNL system do not need to learn UNL. This makes UNL quite different from Esperanto, for instance.

#### 4.1.3 Language Dictionary and Rules

A word dictionary stores information for a language. It stores information concerning what kinds of UWs (concepts), words of the language express and where those words can be used. A word dictionary stores the following items:

- 1) Universal words for identifying concepts
- 2) Word headings for words that can express concepts
- 3) Information on the syntactical behavior of words

A word dictionary provides information for computers to understand natural language, and express information in natural language. A dictionary entry consists of a correspondence between a concept and a word, and information concerning syntax properties of a word when that correspondence was established.

The following shows the text format of word entries:

```
<word entry> ::= <word heading> <word id> <universal word> <syntax
attribute>
<other> ";"
<word heading> ::= "[" <character string> "]"
<word id> ::= "{" <number> "}"
<universal word> ::= "" <character string> ""
<syntax attribute> ::= "(" { <character string> "," }... ")"
<others> ::= "<" <language id> "," <frequency> "," <priority> ">:"
<language id> ::= "E"
<frequency> ::= <number>
<priority> ::= <number>
```

There are two kinds of conversion rules. One is an enconversion rule and another is a deconversion rule. The enconversion rule is used to analyze sentences, and the deconversion rule is used to generate sentences.

## 4.2 UNL Editor and Viewer

UNL editor is used to make UNL documents. UNL editor is linked to a language server equipped with an "enconverter" and a "deconverter" for a natural language. As the author writes a document, e-mail or any other text, in his/her language, UNL editor "enconverts" it into UNL documents. In this process, UNL expressions are produced automatically or interactively with the author.

There are four kinds of UNL editor according to the method of enconversion.

- 1) full automatic enconversion for natural language texts
- 2) full automatic enconversion for controlled or tagged language texts
- 3) interactive enconversion for natural language texts
- 4) word by word input method

The correctness of generated UNL is increased from 1) to 4), but the cost for making UNL documents is also increased from 1) to 4). Users can choose the enconversion method according to the purpose of the UNL documents that he/she wants to make.

UNL editor also shows the input in a UNL document in the author's native language, showing how the UNL editor understands the original document; hereby, the author can check the correctness of the "enconversion". In this verification, the high accuracy of "deconversion" counts a lot. When it is found that the result is not correct enough, the author can either rewrite the original document or modify UNL interactively according to the guidance that is provided by the editor. Then the author can produce a UNL document as correct as is desired.

UNL viewer is used to see UNL documents in a user's native language; UNL viewer utilize a language server when it deconvert, the UNL documents into the user's native language.

### **4.3 UNL Document Base**

In the Internet communication, the dominant text format, HTML, is capable of holding many links with other documents, enabling readers to refer to various kinds of related documents. In summary, an electronic document contains various supplementary information in it, which contributes to increased usability. UNL information is to be equally treated in the network.

One of the HTML merits is that it allows production of the whole document in plain text. In general, information contained in an electronic document is divided into text and embedded instruction. In HTML, however, even embedded instruction is also described in plain text. This characteristic gives HTML a universal adaptability to any editing system, as well as the other advantages of hyper-text. Furthermore, in HTML, description format for embedding is open to the public. HTML conventions are still expanding and developing. Conventions to treat UNL information are expected to be regarded as one of the extensions in HTML.

In order to achieve this universality, it is proposed that the description format for UNL expression is considered as an extension of HTML convention. UNL information can be embedded in a HTML document with tags attached at its beginning and end, which specify the UNL information. Extensions of

conventions should conform to the existing HTML so that it enables UNL expressions to be handled like other documents, without damaging the HTML hyper-text structure. In order to conform to the HTML conventions, descriptions in UNL will be in plain text.

The UNL document base consists of a document management system and a UNL document embedded in HTML.

### 4.3.1 UNL Document Management System

The UNL document base management system manages the UNL document base.

The document base management system consists of:

- 1) Document management system
- 2) Key-concept management system
- 3) Information retrieval system

The Document management system manages UNL documents registered in a UNL document base. For this purpose, it stores the following information for a registered document.

- 1) Document name
- 2) Document ID
- 3) Address of document

### 4.3.2 UNL Document

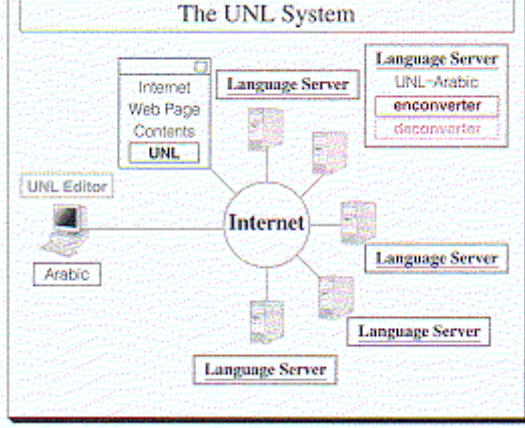
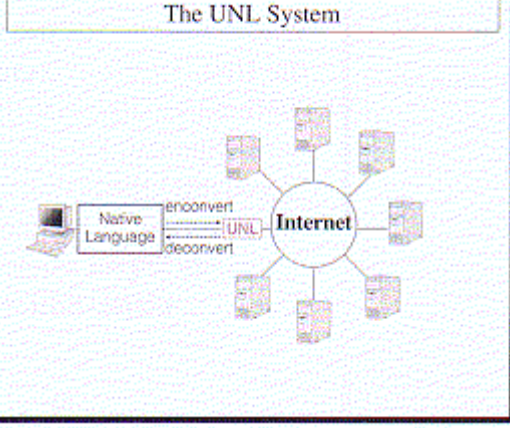
Format of a UNL document is as follows.

```

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

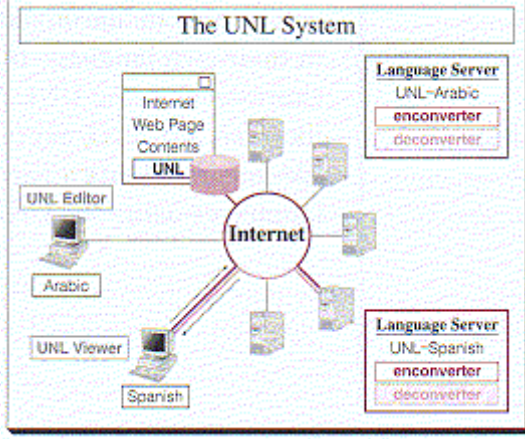
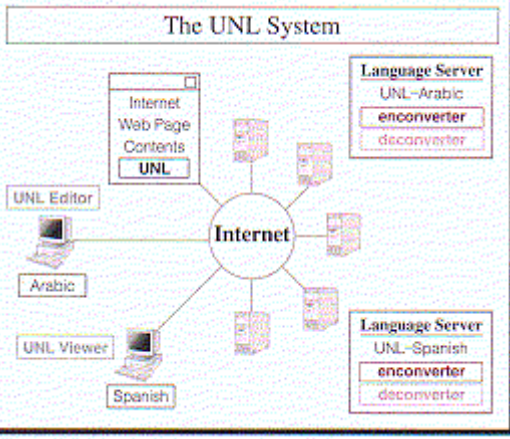
```



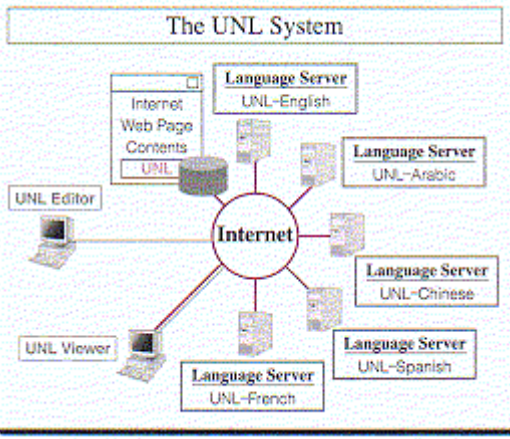
<post editor name>::= "pn=" <character string>  
 <reliability>::= "rel=" <digit>  
 <number>::= <digit>

**4.4 How UNL System works**

Any person with access to the Internet will be able to "enconvert" text written in their own language into UNL expressions using UNL editor. And likewise, any UNL expressions can be "deconverted" into a variety of native



languages using the UNL viewer.



The processes of "enconversion" and "deconversion" are provided by a Language Server which resides in the network of the Internet. The "enconverter" and "deconverter" are responsible for converting a particular language into UNL, and vice versa. The "enconverter" "enconverts" a language into UNL, while the "deconverter" "deconverts" UNL into a native language.

The illustrations on the left shows the case that a home page will be developed in Arabic, through UNL, and we will see this page in Spanish. The Arabic Language Server and the Spanish Language Server provide the conversion service.

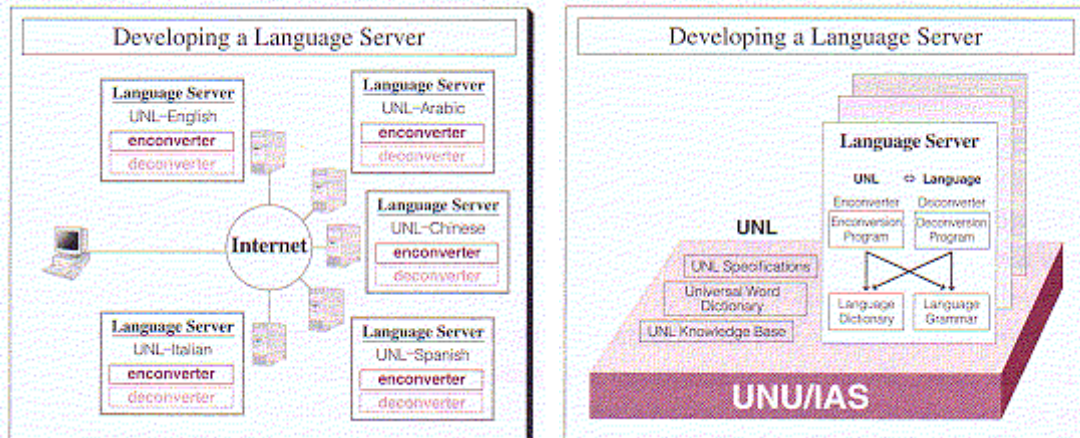
When home pages are developed in Arabic, the UNL Editor recognizes the contents as Arabic and sends a request to the Arabic Language Server to "enconvert" the text. Once the Arabic text is "enconverted" to UNL, the Arabic Language Server sends the results back to the UNL Editor. Home page designers can now embed UNL into their pages.

When we read this page in Spanish, the UNL Viewer recognizes the contents as UNL and sends a request to the Spanish Language Server to "deconvert" the text. Once UNL is "deconverted" to Spanish, the Spanish Language Server sends the results back to the UNL Editor.

The text - once converted to UNL - may be converted to many different languages. For example, home pages can be designed in one's native language

and then "enconverted" to UNL before being uploaded. Once a home page is expressed in UNL, it can be read in a variety of languages.





## 5 Developing and Managing UNL Resources

The UNL system is developed for languages, but it requires expansion and improvement. Scientists in computer linguistics and program developers around the world are invited to join the development of the UNL system. They can improve, or create new, facilities related to the UNL. There are a number of opportunities coming up ranging from the general architecture, new language modules, applications to specific domains, to operational tools.

### 5.1 Developing a Language Server

At least one language server for each native language is necessary. A language server has two software programs residing in it: an "enconverter" and a "deconverter."

During the processes of "enconversion" and "deconversion," two files are mainly used. The first file is a dictionary that lists correspondence between the Universal Words of UNL and the words of a native language. The second file lists grammatical rules. The "enconversion" and "deconversion" programs, together with these two files, constitute the "enconverter" and "deconverter." Each is specific to a particular language server and should be developed according to the UNL Specifications, the UW dictionary and the UNL Knowledge Base.

The UNL Center provides language independent enconversion software (parser) called "EnCo" and deconversion software (generator) called "DeCo". With this basic architecture in place, anyone can participate in the development of a language server.

For example, several different firms could develop language servers and introduce them into the market. With competition among different servers, users could then choose the language server that best meets their needs.

An "enconverter" and "deconverter" can also be developed by interfacing existing systems with UNL. Needless to say, UNL rather than eliminating existing machine translation technologies, builds upon them, and opens a totally new dimension for them.

### 5.2 Developing Dictionaries and Rules

To develop a new language server, it is necessary to develop dictionaries and rules for a new language or a new domain.

When developing a word dictionary, the developer should do the following things.

- 1) Link each head word of a language to UW
- 2) Set grammatical attributes to the word when it expresses the meaning specified by the UW.

When setting a UW for a word, steps are:

- 1) Imagining the set of UWs for a word
- 2) Check that the imagined UW is already registered in the UW dictionary (utilize UW gate)
- 3) If the imagined UW is not found in the UW dictionary, register the imagined UW to the UW dictionary, and also position the new UW in the UW system (hierarchy), (utilize KB gate).

Based on the dictionary developed, conversion rules will be developed. The word dictionary can be used both for enconversion and deconversion when DeCo and EnCo are used.

### **5.3 Maintenance of UNL**

The UNL, as a language, is defined by UWs, relations, attributes and the knowledge base. The UNL can not always cover every domain.

In nature, it is necessary to extend UNL to cover new domains such as medicine, engineering, etc.. In that case, developers of language servers and UNL related application programs should add new UWs, new attributes and knowledge base entries for new UWs through UW gate and KB gate.

## 6 Invitation to a Global Solidarity Endeavor

The UNL offers a unique opportunity for a genuine global solidarity endeavor. Our common well-defined purpose is working together for a goal that is beneficial to all. To achieve it, depends on the vast intellectual resources available in all languages. This is an invitation for universal win-win collaboration that meets the ideals of the UN, UNESCO, EU, and many other international organizations that care about equal opportunities for all peoples. It is a call for a global partnership for a very targeted, attainable, tangible and long-lasting communication facility; a gift for a millennium.

The UNU/IAS will provide the means to accomplish such an ambitious collaborative win-win endeavor. Starting in January year 2000, the UNL resources will be made available to all those who are interested in joining the collective task. The UNL resources will be provided free of charge. This includes the basic documentation and the technical specifications of the UNL system which will be open to researchers, developers, providers and potential users. It will also provide access to the UNL platform, the UNLnet, to all those willing to join the development and improvement of the UNL system, or take advantage of its facilities.

In fact, UNL resources provide great opportunities in all fields: communication, business, governance, education, medicine, entertainment, cultural and academic activities, and many others. Motivations and interests may vary. For instance, many may be attracted by the theoretical foundations of linguistics or by the computational treatment of linguistics that supports UNL, and would like to dedicate their research to advancing the frontiers of knowledge in this field. Other, who are more pragmatic, would prefer to contribute to the development of software applications of the UNL system. Certainly many more people would like just to use it in their business, personal communication or professional work, or provide the UNL resources to customers. All these diverse interests may be accommodated under an umbrella society interested in the UNL resources that, for the sake of identification, can be identified as UNL Society.

### 6.1 An Operational Organization

To manage the UNL resources, while accommodating such a broad variety of interests, a very simple and flexible operational organization will be put in place in early 2000. Its mission will be to expand, develop further, constantly improve the UNL resources, and maintain the quality of the interface among different languages.

The Society will have a broad representation, from representatives of international organizations of member states, to representatives of specific Languages. It will be open for membership to UNL researchers, computer program developers, and service providers from all nations and languages. The Society will be supported by an Operational Center.

Inspired in the model inaugurated by the Internet consortium, the UNL Society provides a framework for a very active and organized involvement of many different partners, without rigid, bureaucratic or costly structures. Its ultimate goal is to catalyze the participation of multiple people and organizations moved by diverse motivations. Eventually, in the future, the society may recognize the need for establishing specific "chapters" of members, which may be grouped according to broad motivations, such as basic research, R&D, and service providers. These categories may comprise the core of the center, which may be constituted by societies.

The Operational center has coordinating functions as well as quality control responsibility of the UNL resources. It will be operating from the UNL Center, at UNU/IAS, as its initial basis. It will operate with distributed local language centers. A minimal administrative structure supports the management and the center.

Given the diverse motivations or interests, there will be an orderly way of entering, benefiting, and contributing to, the "Society".

## 6.2 UNL Assets

The UNL is endowed with valuable resources. These are constituted by "structural" facilities, "enabling" facilities, supporting tools and applications.

The UNL structural facilities are

- the UNL system, the overall language architecture design
- the UNL set of interconnected software (eco, deco and dictionary system)
- the UNL System Network (UNLnet) connecting language servers

The UNL language enabling facilities are

- the UNL editors
- the UNL words dictionary
- the UNL viewer

A number of UNL supporting tools are already available, and many others will come

- UNL word processing
- UW and KB

UNL applications will multiply rapidly with the development of specialized treatment UWs and KB for specific domains of knowledge. Examples:

- UNL "org explorer"
- UNL e-commerce

## 6.3 Protecting and Sharing the UNL Resources

To ensure equal opportunity to all languages, and free access for all, the UNU/ IAS is, and shall remain at all times, the sole and exclusive owner of the UNL resources. This includes the system design and architecture, the computer programs, and the enabling facilities, and respective technical documentation.

UNU/IAS shall have all the property rights of the UNL Language specifications, the Universal Words and Knowledge Base, including linguistic and non-linguistic data which together form the package known as the UNL System.

UNU/IAS shall also have the property rights on further developments of the UNL, in particular all that is related to structural facilities and enabling facilities.

Any commercial use of these resources shall be agreed upon by UNU/IAS, and the terms and conditions defined in a specific and case-by-case contractual agreement.

The UNL resources will be managed, initially, by the UNU/IAS, and then by a well -established "Society", open to UNL researchers, computer program developers, and service providers from all nations and languages.

#### **6.4 Joining the UNL Society**

To be associated with UNU/IAS, interested partners may apply for a membership (form is available on the UNU/IAS web site). After confirmation of acceptance, membership is made effective upon signing the "Terms of Commitment" between UNU/IAS and the applicant. (An outline of the "Terms of Commitment" is available on the UNU/IAS web site).

Fifteen research institutions with more that 100 linguistics and computational linguistics researchers and computer developers, who joined the UNU/IAS in the first hour for the creation of the UNL, are the founding partners of the future UNL Society. Many other research institutes, university departments, laboratories in companies, and individuals involved in Linguistics, Computational Linguistics or related sciences, manifested interest in joining the UNL work. Now, with the available UNL resources, all those willing to join the global endeavour are welcome. UNL/Center has started to accept applications for joining the UNL Society.

#### **6.5 Privileges of UNL Society members**

UNU/IAS will share with the members, free of charge, the following resources:

1. UNL system enabling facilities
  - UNL Specifications
  - Universal Words Dictionary (UWs)
  - Knowledge Base (KB)

2. Access to UNL system facilities (UNLnet), through the Language Server, with its

- Enconversion Programme ( software) which allows connection with the Dictionary and Grammar of UNL for producing native language text into UNL expressions
- Deconversion Programme (software) which allows connection with the Dictionary and Grammar of UNL for reading text in native languages.

The documentation may be downloaded from the Internet with a key to access the UNL facilities. The key will be provided after the "Terms of Commitment" are signed. This opens access to

- Basic documentation on UNL,
- Continuing flow of general information,
- Interface with other researchers.

For those engaged in R&D related to the development of UNL resources (new languages modules, applications, tools, etc,) the UNL center will provide

- Technical assistance
- On-line access to UNLnet for the purpose of developing UNL products;

To repeat:

This is an invitation for universal win-win collaboration that meets the ideals of the UN, UNESCO, EU, and many other international organizations that care about equal opportunities for all peoples. It is a call for a global partnership for a very targeted, attainable, tangible and long-lasting communication facility; a gift for a millennium.