P. S. Pankov,
Manager of Chair, International University of Kyrgyzstan,

P. S. Dolmatova,
Student, Kyrgyz-Russian Slavic University,

Algorithmic Language and Classification of Verbs for Computer-Based Presentation[*]

At the International Conference on Electronics and Computer held in Almaty in 2007 we proposed the following hypotheses (Pankov and Dolmatova 2007):

Hypothesis 1. A human's genuine understanding of a text in a natural language can be elucidated by means of observing the human's actions in real situations corresponding to this text.

Hypothesis 2. An up-to-date computer equipment is sufficient to model situations necessary to detect genuine understanding of main notions in natural languages.

Vygotsky and Saharov (1981) demonstrated similar things having various attributes together with calling them in any artificial "language" (with nouns and adjectives only) to children. If a child called other things in this "language" properly then s/he was asked why s/he had used these words. Winograd (1972) proposed giving commands to a robot with such words as *table*, *box*, *block*, *pyramid*, *ball*, *grasp*, *move to*, *ungrasp*.

By using these ideas, we proposed (Pankov 1992) fulfilling some actions corresponding to a notion and proposed (Pankov and Alimbay 2005):

Definition 1. Let any "notion" (word of a language) be given. If an algorithm acting at a computer performs (generating randomly) sufficiently large amount of situations covering all essential aspects of the "notion" to the user; gives a command involving this "notion" in each situation; perceives the user's actions and performs their results clearly; detects whether a result fits the command, then such algorithm is said to be a computer interactive presentation of the "notion". (Certainly, commands are to contain other words too. But these words must not give any definitions or explanations of the "notion").

Further, we proposed a general scheme of such kind of software (Pankov, Alaeva and Kutsenko 2006).

A hint to the language to present the user's actions (the only example) was in the work of Kustova and Paducheva (1994): "X moves towards the Place" is a pair <"at t1 X is not in the Place", "at t2 X is in the Place">.

A draft of algorithmic language for the unified presentation of notions was proposed by us (Pankov and Dolmatova 2007). For brevity, we denoted it as *NotiLang*. We describe it in details below. In *NotiLang,* definition of any notion involves some *Entities.* We propose to consider the minimal number of *Entities* as the attribute of a notion.

[*] Доклад представлен автором на Республиканской научно-практической конференции «Образование и наука в современном университете: опыт, теория, практика» (г. Бишкек, 13 марта 2008 г.).

Section 1 contains a non-formal description of *NotiLang*. Section 2 describes classification of notions (mainly, of verbs) ensuing by *NotiLang*. Examples of descriptions of some notions in *NotiLang* are in Section 3.

1. Description of *NotiLang*

1.1. Mathematical base for *NotiLang:*

- a rectangle (display) D, each point (pixel) of it has a color, for output;
- controlling any point on D, together with discrete choice of modes (a mouse with two buttons) for input.

Remark 1. Modern displays are formally discrete but they are perceived as continuous. So, we can implement continuous motion.

Remark 2. Virtual objects can also be performed in such way; we demonstrated natural interactive performing of abstract spaces (Pankov and Bayachorova 1996).

One of the colours is one of background, others are ones of objects. A simple object is a family of connected subsets of *D* so that they are constant or varying continuously with time.

An object has a position and can also have other “attributes”: colour, size, direction (angle of rotation), image, caption, sound etc. An object can both be changing itself and be changed (shifted, rotated, pushed etc.) by the user. Objects can overlap each other.

1.2. Content of *NotiLang:*

- a formalized subset of a natural language (we shall define it as *NatLang*);
- statements describing environment: entities; relations among them;
- statements describing the user’s opportunities;
- statements describing objects’ opportunities (possible actions);
- statements describing conditions to meet a statement in the natural language (in temporal order).

Remark 3. We do not consider insonation of commands written in *NatLang* because it is well-known.

1.3. Possible List of Entities

The following Entities are necessary.

Time (as a sequence of necessary actions); *Future; Past; Space*.

(Single) Cursor (moved by mouse) and *Grasping Cursor* (moved by mouse with left button pressed).

Remark 4. According to *Windows* software customs, *Grasping Cursor* implements parallel shift of *Thing;* shift with rotation are proposed (Pankov and Dolmatova 2007); for other mathematical transformations, *Avatar, Tool* and *Magic Wand* can be used.

Avatar (an object affected by and identified with the user).

Thing (a *Thing* itself is an object moved by the user (with *Grasping Cursor*); a *Thing* with other Entities is an object affected by the user).

(Kinds of *Thing):*

Specific Thing (will be described or drawn for any case especially);

Moving (in itself) *Thing*;

Part (a *Thing* with relation to another *Thing*);

*Composite Thing (*made of some *Things);*

Tool (a *Thing* affecting another *Thing*);

Magic (Forcing) Wand (a *Thing* affecting another *Thing* due to its properties);

Transformability (in the form of *Transformable Thing*): *Flexibility* etc.;

Place (a subset of the display *D,* or an immobile *Thing*):

Specific Place (will be described or drawn for any case especially);

Place-of-Thing (related to a *Thing*).

Motion (in the form of *Moving Thing* without the user's actions).

Animate (an *Animated Thing* which can perceive; by the way, some devices connected with computers are animate in up-to-date speech).

Couple; Plural (some notions mean *couples;* some verbs demand some homogeneous *Things*).

Gravitation (some verbs and prepositions demand it).

Irreducible Adjectives such as *Color.*

Overlapping.

Thing's Actions such as *Sound; Sing; Light.*

Entities can be subdivided to concrete (*Thing, Place, Part, Tool)* and abstract. Concrete Entities participate in the Initial Environment (see below).

Define the function Name for all objects except *Avatar* and *Cursors.*

Also, to distinguish features of any notion an environment with a finite number of entities as a generalization of an environment with a finite number of conservative laws as we proposed (Pankov and Baryktabasov 2004).

1.4. Definitions of Notions

A Condition is a statement about existence and non-existence of common points of objects considered as subsets of the display *D* during the user's actions. Switching from *(Single) Cursor* to *Grasping Cursor* is denoted as Grasp; reverse switching is denoted as Ungrasp.

An *Action* (*Ringing, Singing,...*) is performed by computer's multimedia when any Condition is fulfilled.

(It is meant that *Actions* are made by *Things* forced by the user).

Three techniques are proposed for the user's guessing:

G1) uniqueness of the action, (or the sequence of actions) which subdues the command and the situation naturally;

G2) similarity (some objects have the same property and this property is mentioned twice);

G3) alternation (new Notion and new word appear together as an alternative to preceding Notions).

Definition 2 of a Notion includes:

DE) List of Entities;

DP) List of preceding Notions (if it is necessary);

DI) Description of the Initial Environment (if it is necessary);

DC) Command (written in *NatLang*);

DS) Sequence of Conditions (in temporary order) with operations AND, OR, XOR, NOT and Actions.

If all Conditions are fulfilled then the announcement *Yes* outputs (the Notion has been understood).

If any Condition is violated then the announcement *No* outputs. Also, in learning mode, according to *Windows* software customs: if the user tries to take a wrong *Thing* then the computer permits a small shift only and returns it to its starting-position; if the user tries to move a fitting *Thing* onto a wrong spot then the computer also returns it to its starting-position (number of attempts is not bounded).

Due to Definition 1, some objects (and sometimes their positions) are taken randomly, as long as every implementation of the Notion; they will be denoted with (R). Grammar rules of *NotiLang* will not be described in details; they will be understood from examples below.

2. Classification of Notions in *NotiLang*

By this approach, the set of "notions" is semi-ordered by precedence: of each two "notions" either one must be introduced before other or they can be introduced simultaneously or they can be introduced independently.

Also, the analysis of definitions constructed due to 1.4 confirms the well-known opinion that the verb is the main part of speech, and their classification is definitive for the whole language. A detailed survey of different approaches of verbs classification is made by Sanfilippo (1996). These classifications are ascertaining, while we propose a constructive one. Also, the demand of constructivity distinguishes essential mathematical-physical features that were neglected because of ascertaining of mentioned classifications.

First of all, our classification develops (with some additions) the well-known valence of verbs but with specifications of objects and addition of latent circumstances. The closest is Lexical Conceptual Structure which "is mainly organized around the notion of motion, other semantic/cognitive fields being derived from motion by analogy (e.g. change of possession, change of property)."[1]

For constructive purposes, we cannot use analogy. We are based on the user's concrete *"actions"*.

In this paper, we propose the following classification of verbs (as it is known, some verbs are polysemantic and can fall into different sections). Also, some verbs are introduced with the minimal number of Entities but can be used with additional (facultative) Entities further. For example: *ring the bell; ring the bell with a stick.*

The first subdivision is: imperative and non-imperative verbs. If a verb can be performed naturally by its imperative mood and the user's corresponding action then it is imperative. The following verbs are non-imperative: *see, can, understand.* Their possible performance will be given below.

In their turn, imperative verbs may be subdivided as follows:

Direct verbs: the user acts with *Cursor* (*move, shift, take, put, find, hide, show*).

Tool verbs -"- with *Tool* or other *Thing* (*paint, cut, put, find, hide*).

Avatar verbs -"- by means of *Avatar (go, turn, push, pull).*

Forced verbs -"- by means of *Magic Wand (run, ring, sing, light, jump, sit down, stand up).*

Due to precedence ordering of notions, transitive verbs can be subdivided into "independent" of essence of direct objects (*move, take, put, find, hide*) and "dependent" (*write, paint, flex, read*).

[1] A. Sanfilippo et al. *Preliminary Recommendations on Semantic Encoding. Interim Report.* (The EAGLES Lexicon Interest Group / Verb Semantic Classes, 1996)

3. Examples of Verbs in *NotiLang*

We shall not describe *NotiLang* formally; we shall give outlines of each notion only. Also, one of the ways to implement G1 is constructing such environment that the only natural action is evident. Then corresponding *Specific Things* and *Specific Places* will be described for a programmer non-formally.

Examples (for English and corresponding Kyrgyz words) are given due to Definition 2 or non-formally. One of the ways of guessing (G1, G2, G3) and number of entities are given. Also, some simple *Things* (*square, circle, rectangle*) are necessary.

3.1. Verb *put="кой"* (G1), Entities=5.

DE) *Time; Topological Space; Cursor*; (R)*Thing*; (R)*Place*.

DC) "Put"+Name(*Thing*)+"into"+ Name(*Place*)+"!"

DS) Grasp in *Thing*. Ungrasp in *Place*.

3.2. Verb ***make="жаса"*** (G1), Entities=6.

DE) *Time; Topological Space; Cursor*; (R)*Thing*(-with-out-*Part*); (R)*Part*; *Gap*.

DC) "Make"+Name(*Thing*)+"!"

DS) Grasp in *Part*. Ungrasp in *Gap*.

3.3. Verb *show="көрсөт"* (G1), Entities=5.

DE) *Time*; *Topological Space*; *Cursor*; (R)*Thing*; *Plural*.

DC) "Show"+Name(Thing)"!"

DS) *Cursor* is in *Thing*.

3.4. Verb *cover="жап"* (G1), Entities=6.

DE) *Time; Topological Space; Cursor*; (R)*Thing1*; (R)*Thing2* (larger than *Thing1* and *Overlaps* with respect of *Thing1*).

DP) *Initial nouns*.

Step 1.

DI) *Cursor* is in *Thing2*.

DC) "Cover"+Name(*Thing1*)+"!"

DS) Grasp in *Thing2*. Ungrasp in *Thing1* [i.e. *Thing1* is covered by *Thing2*].

Step 2. (The preposition *with* is also introduced).

DE) *Time; Topological Space; Cursor;* (R)*Things1;* (R)*Thing2 (larger than Things1); Plural.*

DC) "Cover"+Name(*Thing1*)+"with"+Name(*Thing2*)+"!"

DS) Grasp in *Thing2*. Ungrasp in *Thing1*.

3.5. Verb ***find="тап"*** (G1), Entities=6.

DE) *Time*; *Topological Space*; *Cursor*; *(R)Things1*; *(R)Thing2* (larger than *Things1*); *Plural*.

DP) Initial nouns; verb *Cover* (or *Close*, *Hide*...).

DI) Brief description. All *Things1* are covered with copies of *Thing2*.

DC) "Find"+Name(*Thing1*)+"and touch it!"

DS) *Cursor* in *Thing1* [after shifting *Things2* from some or all *Things1*].

3.6. Verb *push="түрт"* (G1), Entities=5.

DE) *Time; Geometrical Space; Avatar;* (R)*Thing;* (R)*Place.*

DP) Initial nouns.

DI) [If *Avatar* touches *Thing* then *Thing* moves along the opposite direction].

DC) Push"+Name(*Thing*)+"into"+Name(*Place*)+"!"

DS) The center of *Thing* is in the interior of *Place*.

3.7. Verb *symmetrize="симметрияла"* (G1), we noted its naturalness (3).

DI) Slightly non-symmetrical object.

DC) "Symmetrize!"

DS) The user is to make the object symmetrical.

3.8. The scheme for *Forced verbs* (for instance, *sing="ырда"*)

DE) *Plural*; (R)*Things* (one of them can sing, others can sound, tremble etc.); *Magic Wand*.

DP) Some nouns.

Step 1.

DC) "Touch"+ Name(Thing)+"with"+"Magic Wand+"!".

DS) *Magic Wand* is in Thing: *Sing*.

Remark 5. Such technique (with *Click* instead of *Magic Wand*) is used in some teaching software to enliven environment.

Step 2.

DE) *Plural;* (R)*Things; Magic Wand;* (R)*Place*.

DC) "Touch things with Magic Wand and put a thing which sings into "+Name(*Place*)+"!".

DS) (*Magic Wand* is in the *Thing*: *Sing*) OR (*Magic Wand* is in other *Things*: Other *Actions*). The *Thing* is in *Place*.

3.9. Non-imperative verb *see="көр"*.

DI) A Thing is in the center of a circle with a gap; Animals are around the circle; one of them is at the gap.

DC) "Show the animal seeing Name(Thing)!"

DS) *Cursor* is in the Animal.

3.10. Non-imperative verb *can="Stem-of-verb+"A" ал"*.

DI) A high pole with Thing on its top; Giraffe and other Animals are near the pole.

DC) "Show the animal which can take Name(Thing)!"

DS) *Cursor* is in Giraffe.

Conclusion

We do not pretend to an adequate description of a language as whole in this paper. Nevertheless, we hope that further research with permanent feedback of results treating corresponding software would distinguish important features of natural languages and yield an objective base for comparison of notions of different languages.

By our experience, some 7- or 8-year-old children having a good command of using a computer mouse can adequately understand commands in an unknown language, can learn dozens of words and pass a test without mistakes. Adults are less careful and they try to ask: "*I have guessed what I am to do, but how can this word be translated?*"

We hope that such software would be able to be used as an introduction to learning languages by means of distributing CD-ROMs with corresponding software. Besides, the test mode can be used in various kinds of examinations and competitions for all students learning foreign languages. For advanced students, writing definitions of various notions in *NotiLang* would be also useful.

References

1. Kustova, G.I., Paducheva, E.V. 1994. Semantic Dictionary as a Lexical Database. *EURALEX, Proceedings.* Amsterdam: 479-485.
2. Pankov, P.S., Aidaraliyeva, J.Sh., Lopatkin, V.S. 1996. Active English on computer. *Conference "Improving Content and Approach in the Teaching of English Language in the Context of Educational Reform"*, Bishkek: 25-27.
3. Pankov, P.S. 1996. Independent learning for Open Society. *Collection of papers as results of seminars conducted within the frames of the program "High Education Support".* Bishkek: Foundation "Soros-Kyrgyzstan", Issue 3: 27-38.
4. Pankov, P.S., Alimbay, E. 2005. Virtual Environment for Interactive Learning Languages. *Human Language Technologies as a Challenge for Computer Science and Linguistics: Proceedings of 2nd Language and Technology Conference*, Poznan: 357-360.
5. Pankov, P.S., Alaeva, S.A., Kutsenko, V.A. 2006. Algorithmic Interactive Presentation of Notions. *Proceedings of the 11th International Conference "Speech and Computer" SPECOM'2006.* Saint-Petersburg: 478-480.
6. Pankov, P.S., Bayachorova, B.J. 1996. Using computers to perform non-Euclidean topological spaces. *The 6th International conference and exhibition on computer graphics and visualization "Graphicon-96"*, Vol. 2, Saint-Petersburg.
7. Pankov, P.S., Dolmatova, P.S. 2007. Algorithmic language for computer-based presentation of notions. *4th International Conference on Electronics and Computer.* Almaty: Suleyman Demirel University: 274-279.
8. Sanfilippo, A. et al.1996. Preliminary Recommendations on Semantic Encoding. *Interim Report. The EAGLES Lexicon Interest Group / Verb Semantic Classes.*
9. Winograd, T. 1972. *Understanding Natural Language.* New York: Massachusetts Institute of Technology.
10. *Выготский Л.С., Сахаров Л.С.* Исследование формирования понятий: методология двойного стимулирования // Хрестоматия по общей психологии. Психология двойного стимулирования. – Москва: изд. МГУ, 1981.
11. *Панков П.С.* Обучающая и контролирующая программа по словоизменению в кыргызском языке на ПЭВМ. – Бишкек: Мектеп, 1992.
12. *Панков П.С., Барыктабасов К.К.* Инварианты при бесконечномерных геометрических преобразованиях и их компьютерная реализация // Вестник КГПУ им. И.Арабаева. Серия 3, выпуск 2: Педагогика. Психология, 2004. – С. 56-60.
13. *Панков П.С., Долматова П.С.* Построение дифференциальных уравнений для гибкого управления объектами с помощью компьютерной мыши // Исследования по интегро-дифференциальным уравнениям, вып. 37. – Бишкек: Илим, 2007. – С. 18-23.