E-membranes to detect Users' Behavior Patterns

Dr. JUAN CHAMERO
Artificial Intelligence Lab
CAECE University
Buenos Aires
ARGENTINA
juan.chamero@intag.org , http://www.intag.org, http://www.caece.edu.ar

Abstract: - Darwin-FIRST, is an Artificial Intelligence procedure to automatically retrieve intelligence diluted in huge databases. FIRST stands for Full Information Retrieval System on Thesaurus, one expert system aided by agents and Darwin stands for Distributed Agents to Retrieve the Web Intelligence, a network of FIRST nodes. FIRST algorithms and agents extract from databases their *inherent intelligent skeletons* throughout *e-membranes*. These e-membranes will allow Internet users to locate Cognitive Objects directly, like it they were exposed in supermarket stands. They behave like special high resolution magnifier "glasses" that enabled them to "see" and to retrieve those Cognitive Objects precisely.

E-membranes are like real bio membranes, they have their own *e-endoderm*, *e-mesoderm*, and *e-ectoderm*, virtual matchmaking interfaces that separate servers from their users, cognitive offer from cognitive demand. In the same way that e-membranes could be settled and tuned up to extract inherent intelligence skeletons from databases they could be settled and tuned up to extract the *people's inherent intelligence skeleton* as well.

The ectoderm is enabled to detect *users' behavior patterns* that at users' side are analog to keywords at servers' side. E-endoderm takes care of messages from e-mesoderm to server's side and vice versa. Content and its structure evolution are controlled by special messages. E-mesoderm takes care of these special messages generation related to evolution and learning processes. In theory once adequately settled and tuned up e-membranes could be left to work autonomously. However the whole process, and the "permeability" of the e-membrane layers is supervised by a *Desktop*, controlled in its turn by a human being, the *Chief Editor*.

E-membranes may work in three modes: a) extracting intelligence, for instance to build a proprietary intelligent content to attract users; b) as an i-matchmaking interface between servers and their users, continuously optimizing offer versus demand, autonomously freezing/killing unused offer and/or searching new offer for unsatisfied demand; c) at users' behavior pattern detection mode, learning as much as possible from users interactions

With these e-membranes we may build super search engines, working over the Web space in first mode first, and then creating a *Web Thesaurus*, the inherent intelligence skeleton of the Web. We may also recover the inherent intelligence of any huge data reservoir to facilitate its overall security and daily management. In this case we make reference to corporations that along years have nurtured huge databases without taking care of saving its history indexed by their inherent database intelligence.

A prototype of this e-membrane is now working in Darwin-FIRST Demo site hosted in www.intag.org. Its procurement agent worked in the first mode to extract the inherent intelligence skeleton from the Web for a single discipline: Computing. This skeleton has the form of a Thesaurus of 53,000 keywords related to a Logical tree of 1600 branches/leaves and a set of nearly 6,000 Authorities. Now the prototype is working in modes 2 and 3.

Key-words:-E-membrane, Knowledge management, Knowledge representation, Web thesaurus, Data mining, People's pattern behavior, Agents, Behavior patterns, Search engines, Man-machine

1 The apparent chaos of huge data Reservoirs

The fig. 1 depicts how *i-Webs*, Webs with built-in *e-membranes*, would perform their content procurement and retrieval tasks. Black region represents the Web space that hosts more than 8,000 million documents. Search Engines, like the one depicted in blue, aid world wide users to locate what they need [16]. Being this space so vast, some ranking procedure is needed. However, the Websites competence to "rank high" is so fiery to encourage all kind of dubious ethic marketing strategies and many times unethical habits oriented to deceive the Search Engines robots that continuously browse the Web space to update their ranked lists. For this reason two equally important Websites like the ones shown may look extremely different concerning their rank: one like a bright star on the Top and the other like a pale star among millions.

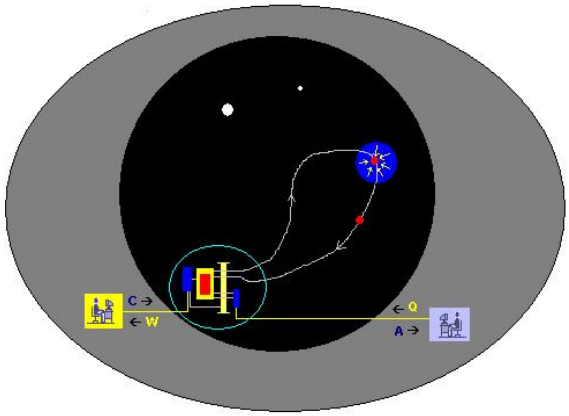

Fig. 1

This problem discourages general users unable to find what they look for to satisfy their "curiosity", their information needs. We used to work as content experts, retrieving "difficult-to-get" information for others. We learnt how to dig and to see within the apparent chaos. Our second step was trying to build and train agents to aid us in our task but previously we tried to behave and to "think" like robots [10] [3].

2 Physiology of Procurement throughout e-membranes

These "procurement" and "retrieve" processes are shown by the loop in gray that goes from an i-Website (magnified to see the data process physiology) to a Search engine and then back to the i-Website carrying the content procured/retrieved, represented by a red dot. The agents play a fast convergent "game" (resembling an interception game) versus the search engine content [4] [16].

In this way we may retrieve any kind of *Cognitive Content* synthesizing any kind of *Info maps* [15], Human Knowledge Maps, and Commercial and Industrial Catalogues. This type of content attracts users as it's depicted in the figure. Users and i-Website *Chief Editors* are located in the Cyberspace, grey region, each one connected via their respective local ISP, Internet Service Provider.

As time passes by i-Website cognitive content shown as red rectangle becomes obsolete and experiments degradation. The same loop that generates the initial content serves to maintain and update it continuously. So many complex and diverse tasks require a Command Board, defined as *Darwin-FIRST Desktop* depicted as the blue rectangle connected to the Chief Editor.

Users communicate with i-Website either directly or through a personal platform that could also become intelligent, something like a *User Desktop*, depicted as a small blue rectangle.

Now we are ready to introduce FIRST, the *Expert System* that learns from *man-machine* interactions. Users interact with i-Website content through Q, Queries (man action) receiving A, Answers, as reaction (machine re-action). In order to learn from users FIRST needs to know how users act and react. FIRST solves the easiest part of the problem indirectly, (how users react) from *virtual chips* attached to each cognitive unit of its content that log their use history "from cradle to grave" [13].

We have now clearly defines the two sides that globally interact in the Web space, users and *owners*. Users act individually and owners act represented by their Chief Editor. We may imagine then a virtual interface between them, depicted as the "e-membrane", a "Match Making" interface shown as a yellow wall in the figure above.

Chief Editors control the whole flow of communications throughout their Desktop via W, Warning "be-aware-of" messages received from agents and re-acting controlling the whole system via C, Command messages.

FIRST could extend the scope of its primal content (in red) by attaching "similar cognitive content" (yellow) [8]. FIRST similar agents may bring a set of

similar for each cognitive unit stored as primal content. Where rests the difference?. Primal Content is somehow "certified" information meanwhile *Extended Content* is up to users. However FIRST could be adjusted to measure their user's preferences in order to enrich its Primal Content [14].

3 E-membrane roles within FIRST architecture

Content generation starts (Fig. 2) with the LT, *Logical Tree* of a given discipline, Catalogue of Parts, or any suitable collection of Objects. Agents (a) guided by LT structure go to crawl the Web collecting a huge amount of data where from by a depuration algorithm "straw from wheat" is separated. Wheat becomes the TH, *Thesaurus* of the given discipline [5] [9]. In parallel, a "reasonable good" collection of documents, considered *authorities* for the given subject, is mapped onto *Content Maps*.

Now the *Triad [LT, TH and Maps]* represents the *Inherent Intelligence Skeleton* of the given discipline. This procedure repeated for all possible LT's of the Human Knowledge would become a basic *Human Map Knowledge. Clones* of these maps installed in i-Websites, and empowered by FIRST engines, will behave like super search engines, providing *You Get What You Want in only one click interfaces.* A *virtual chip* is attached to each Cognitive Object (an intelligible summary of the Authority pointed) to register its life cycle in detail, from cradle to grave. .

The e-membrane is depicted dividing the local cyberspace in two, the *Website realm* where the Content is hosted, and the *users' realm* where people as users interact with the Website content. Agents (a) transport messages from one side to the other and perform missions under FIRST supervision but managed at the upper level by the Chief Editor. Green regions resulting from users' interactions will make evolve TH, Map, and even LT, the content structure. New keywords and new authorities are continuously suggested by agents and users, exploring the Cyberspace guided by the Triad.

FIRST via a family of *welcome agents* [1] continuously detects and classifies *users' keywords*, and *threads* of words and keywords, pieces of *users' Jargon* and components of their "speech". The black region hosts *users' speeches* and their inherent intelligence as well, enabling the continuous monitoring and matchmaking of the e-membrane operation.

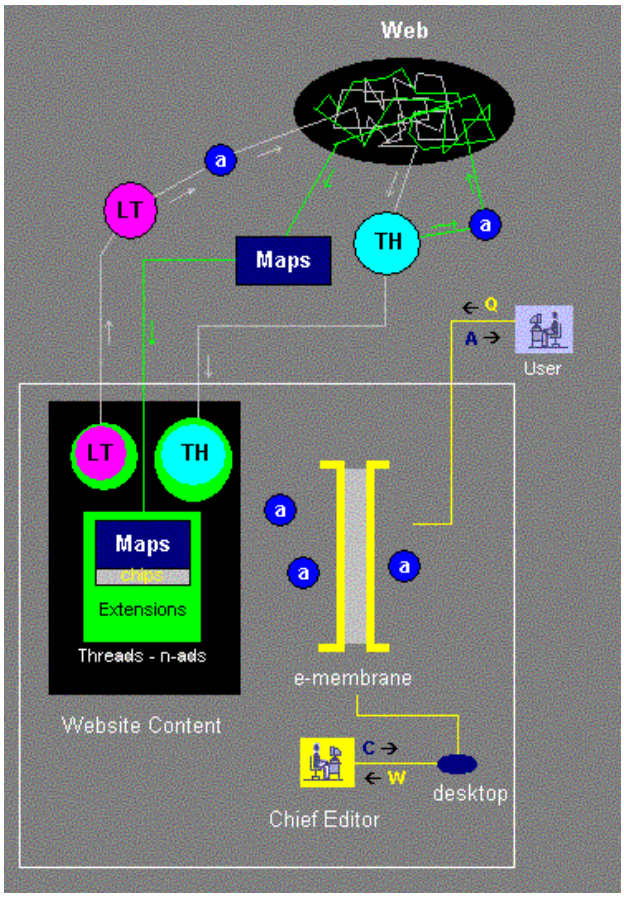

Fig. 2

4 A Word about Inherent Intelligence Skeletons

What are databases in essence? Collections of data related to the human being, its nature and its activities. Documents are pieces of knowledge directly or indirectly issued by humans and most of them are actually stored in databases [2] [9]. A virtual representation of the actual Human Knowledge is hosted in the Web space represented by billion of documents. Those documents are indexed by search engines robots that continuously crawls the Web space trying to facilitate their retrieval. However, not only data are stored "up there" but intelligence, under the form of meaningful messages we have to unveil.

Any collection of data could be ordered in as many ways as we imagined but it has its inherent ordering

schemes, one and sometimes more than one. Let's talk for instance about human documents. They deal with "subjects", and they are written in a given language and in a given jargon as well. To express their ideas humans use sequential strings of "common words" and "keywords" belonging to a given language and to a given jargon. Letting common words aside, a document becomes then a string of keywords and at last a weighted string of non repeated keywords. Let's compare two documents that deal with the same subject in the same language and in the same jargon. We may define a metric of "similarity" between documents that goes from 0 to 1. One meaning that both documents have the same *fingerprint* – they have exact the same weighted string of keywords-. Zero, on the contrary, stands for null coincidence [8].

Documents could then be abstracted by an intelligence skeleton of subjects and fingerprints closely interrelated and referred to logical trees. Any collection of Cognitive Objects has its own inherent intelligence skeleton. Once this skeleton is known the whole collection could be seen as perfectly ordered and consequently directly retrieved. What does it mean?. You may retrieve what you need or at least a sure track to satisfy your need in only one "move", being equivalent to know the "coordinates" where some specific data is stored.

Any data reservoir could be assimilated to this knowledge model [5]. What's important is that the triad [LT, TH, Map] Logical Tree, Thesaurus, Map, behaves like an interrelated entity where: once known two of them you may obtain the third, and even known one of them you may infer possible configurations of the other two [11].

5 How Humans retrieve solutions?

In fig. 3 we depicted a set of *users' query strategies*. Each user has his/her own uncertainty, ignorance about something he/she wants to know, represented by circles and ovals where colors and sizes would represent different levels of relative uncertainty and skills to locate what they need. The user that appears on top has held a *session* of 8 questions. We, as observers, normally ignore how many searches a user performs, that is the same as to say that we ignore how many needs he/she is trying to satisfy in each session. Let's take a close look at his/her possible reasoning mechanic. In red we were trying to represent the brain reasoning-gram: Given a need he/she issues, through a complex reasoning we are not going to analyze here, depending of the knowledge this user has, his/her temper and state of mind, and many other individual and contextual factors, keyword k1. Once the *Cognitive Offer* existent at the owners' side of the *e-membrane* gives its answer, it is receipted by the user's brain, analyzed and pondered whether the need has been thoroughly satisfied or not. If it has not been yet satisfied user proceeds issuing another keyword k2, and so on and so forth either till satisfaction or to the end of session.

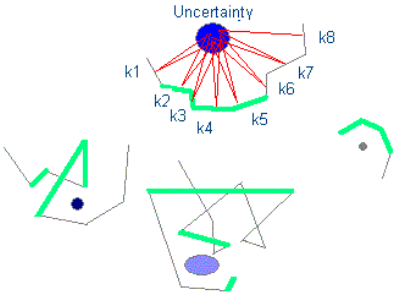

Fig. 3

Note: We were talking along our reasoning about people questioning, either explicitly or implicitly, by pairs [k, s] where k stands for keyword and s by subject, namely by "acceptations" or meanings for a given keyword. So a search query strategy has the form of a string of pairs [k, s] instead of k's. To make the things close to reality pairs could pertain to different disciplines instead of running within a single one.

6 How Robots would retrieve solutions?

How could we envisage a simplified digital approach to interpret "how a human proceeds to look for solutions through questioning a search engine"? [7] [8]. It was the first question we made ourselves before designing the *people's behavior pattern algorithm*.

Humans have an inherent need of *solutions* to solve their working and existential problems. Let's imagine then a very simplified version of a robotized *humanbot*. It, instead of he/she, has its knowledge defined as a set of *solutions*. It also has a collection of *subjects* or *activities* [10]. These activities are organized in Logical Trees for humanbot playing different roles, and each activity will have an associated set of keywords. So our humanbot has at hand (or has to build first) the following resources:

- Professional and social activities hierarchically organized as LT's per role, for instance, as a Real State professional, as a father, as a husband, as member of a club, as member of a political party, etc.
- A set of Solutions for each activity, under the form of "how to perform". Probably there is more than one approach about the best how to perform a given activity, let's say more than one solution approach for each activity.
- A set of keywords associated to each activity. These *users' keywords* are meaningful concepts that are stored in its long range memory.

Note: Be the keywords "Tea infusion" and "floor inspection". Surely humans have in their minds both concepts clearly defined but probable have not yet identified them with specific words and/or symbols. They know pretty well what a tea infusion is and perhaps how to prepare it and in the case of a real state expert he knows pretty well how a floor inspection has to be performed but perhaps he do not know yet its precise expression to share it with similar humans [8]. The only way real human of flesh and blood and humanbots have to search information related to these two concepts is to use "established" standard keywords, for instance *tea, infusion, tea infusion, floor, inspection, floor inspection*. Our humanbot will find at large what it needs throughout a thread of established standard keywords. One of our strong conjectures is that within homogeneous communities and for each human activity appears frequent threads – or search strategies- that could be associated to *users keywords*. We reiterate this conjecture to avoid confusions: we all know what a word is, but a keyword makes reference to a concept that could be formally defined by a word or by a sequence of words. That happens in the establishment side of the e-membrane, arbitrarily in our drawing to the left of it. To get something, our humanbot, located "on the other side", at the right of the e-membrane, is by de facto obliged to query by these keywords, the only standard it knows. A thread or search strategy would be then a string of the following nature: [(w, w) w w (w, w, w)] where w stands for a single word. This thread has four standard keywords, namely (w, w), w, w, and (w, w, w). But concerning the people's side the whole thread behaves like a *user keyword:* it belong to an activity and it points to a given solution or set of solutions.

Let's go back to our initial purpose: How our humanbot queries and learns. It wants to know as much as possible about a given subject s' (to differentiate from s, the subject at the establishment side). The only way it has to learn is to query via standard keywords. We imagine this complex process this way [1]:

- Step 1: s' [k], where [k] is the associated set of standard keywords related to activity s'.
- Step 2: at random or perhaps triggered by its particular "mood" at this moment (we may simulate moods) our humanbot extracts one k out of [k].
- Step 3: It Queries search engine by k
- Step 4: it obtains a list of n References. It first checks n against its "experience". If it is too low "perhaps" it decides to go to make another query with a different k. It "learns" a little from this experience. This is a very complex step because our humanbot has to decide if the "searching game" against the search engine is going on well. It will arrive to this step several times obtaining for a given s' a sequence like: n1, n2, ……, nj, where n1<<n2<<……..<<nj. It also learns about navigation techniques along this step.
- Step 5: it "perhaps" decides to review the Top h References, and/or to inspect some documents. It learns too much of this review, and on account of this learning "perhaps" it decides to modify [k] and even s'. If it feels satisfied it probably stops, collects solutions and pieces of solutions. Eventually it goes to initiate another search with another s'.
- Step 6: It decides whether or not to continue trying with another k

Imagine an Artificial Neural Network to emulate this process and the difficulties to match hypothesis with real humans. Our agent design strategy is to consider a universe of billion [k] session tracks and extract patterns from them. Of course we ignore the particular [s'], we only know sequences of [k].

This log will look like {[{r, s'} k k k | {r, s'} k| {r, s'} k k k k k | ……]; [{r, s'} k|k|]; [{r, s'} k k k k k k | {r, s'} k k | {r, s'} k| {r, s'} k| {r, s'} k k k k k k k ...]; ……..},

Where | bar represents the stop of a search. {r, s'} k k k k, represents a search of solutions for subject s' of role r. Of course we ignore the pairs {r, s'} either. Red | represents successful search, user has found a reasonable solution but we also ignore these worthy references. Our raw information only consists of strings of keywords per session.

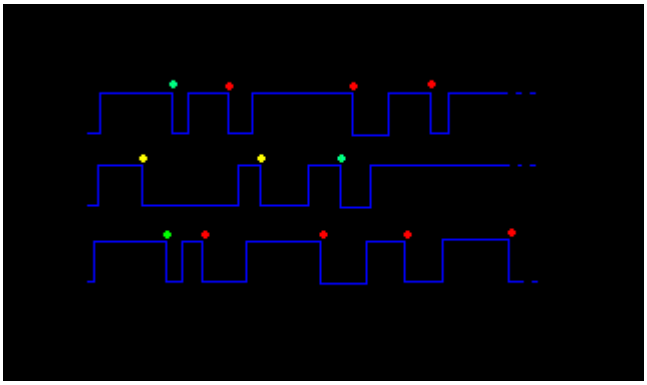

Fig. 4

Fig. 4 depicts a *search-o-gram*, expressed like waves of k sequences. At the end of each sequence

we mark the humanbot level of satisfaction: red ⇔ poor, yellow ⇔ medium, and green ⇔ high. In the future the people's realm should be carefully investigated either via humanbots or by humans, playing specific roles and with specific talent and knowledge. It will be crucial for instance to know the thread size distribution to succeed querying conventional and/or thematic search engines [11]. The parameters of that type of research would be:

User, Type of user, Query ID, Alleged subject, Search mode, Initial number of References, Final number of References, Thread size, Level of satisfaction.

Because it is important to know not only about thread but also how fast the search process converges towards solutions, for activities, roles, etc. Search mode makes reference to use of search engines facilities, for example, by words, by keywords, by advanced search (zooming by adding words/keywords without change the thread ancestors), etc.

7 How do we proceed without interfering with users interactions?

As we ignore almost everything about users and our approach is a non interfering policy, we initially ignore everything about *users' activities*, *users' keywords* and *users' solutions*. The only we know is the *establishment inherent intelligence skeleton* and the *users' tracks* that query the establishment to know as much as possible about solutions [12]. So the only intelligible data we have at hand are users' sessions and their tracks. This track is a sequence of keywords and navigations instances. Let's see how we design the inference algorithm to detect users' behavior patterns.

For a given span of possible membership to a thread, let's say span=4, we have for each k of the user track three more possible associated search membership. We compute then all possible n-ads for each k. Be our user track the following: [1 2 3 4 5 6], and taking a span of 4 we have the span sequences [1 2 3 4] [2 3 4 5] [3 4 5 6] [4 5 6] [5 6] [6].And for each possible search membership we have to consider all combinations without taking care of the relative order (it's another conjecture to be tested):

[1 2 3 4]: [1] [2] [3] [4]; [1 2] [1 3] [1 4]; [2 3] [2 4]; [3 4]; [1 2 3] [1 2 4] [2 3 4]
[2 3 4 5]: ~~[2] [3] [4]~~ [5]; ~~[2 3] [2 4]~~ [2 5]; ~~[3 4]~~ [3 5]; [4 5]; ~~[2 3 4]~~ [2 3 5] [3 4 5]
[3 4 5 6]: ~~[3] [4] [5]~~ [6]; ~~[3 4] [3 5]~~ [3 6]; ~~[4 5]~~ [4 6]; [5 6]; ~~[3 4 5]~~ [3 4 6] [4 5 6]
[4 5 6]: ~~[4] [5] [6]; [4 5] [4 6] [5 6]; [4 5 6]~~
[5 6]: ~~[5] [6]; [5 6]~~
[6]: ~~[6]~~

Note: we are considering here triads as the maximum size thread, based in our searching experience. More than three k leads to either lack of references or a rather redundant search strategy. It means that high frequency triads are probable users' behavior patterns (two in less extent).
.

Monads: 6: [1] [2] [3] [4] [5] [6]
Dyads: 12: [1 2] [1 3] [1 4] [2 3] [2 4] [2 5] [3 4] [3 5] [3 6] [4 5] [4 6] [5 6]
Triads: 7: [1 2 3] [1 2 4] [2 3 4] [2 3 5] [3 4 5] [3 4 6] [4 5 6]

8 N-ads Generation

As we will see our n-ads generation algorithm will look like a data mining algorithm but instead of working with millions of "ex post" transactions it works on a differential strategy, processing transactions at the right moment they are generated. In each user session we may distinguish two kinds of events: inquiries and "instances". An inquiry is made by pairs (keyword, subject) and instances represent all possible navigation circumstances. So the first step is to split the session in two strings, one for the sequence of queries and other for the sequence of instances. In fig. 5 we present the n-ads generation from queries strings. An n-ad is a set of pairs (keywords, subject). In some extent these strings are like "Tarzan conversations" with a virtual Oracle, but anyhow conversations!

Once the string of k's splits in n-ads, they are inserted and accounted in its corresponding nk-Thesaurus. An nk-Thesaurus is a vivid collection of typical nk inquiries. In general terms, we may state that the larger the "power" (n) of the inquiry, the higher the certainty of getting a valuable answer.

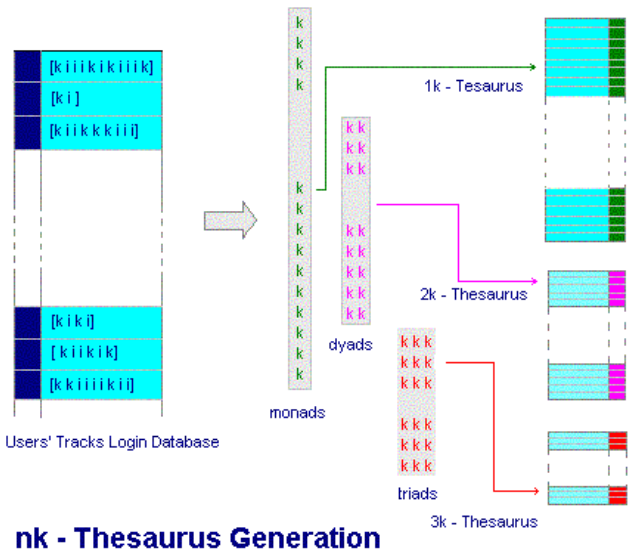

Fig. 5

9 PAG's, Potential Affinity Groups

Our thesis is that PAG's, Potential Affinity Groups could be suggested based on Top nk-ads of order equal or higher than two. It means for instance, that Top 3k-ads tell us that users using those triads have a high probability of having interests in common. The vertical column, it is shown at right in the nk-Thesauruses, accounts for the popularity of nk-ads. The blue column at left, in the *Users' Tracks Login Database*, corresponds to the *ID's Tags of sessions*.

So we may correlate TOP nk-ads to users ID's and those correspondences lead us to suggest PAG's. Now FIRST knows "something" about users' behavior patterns, and from now on, each time a welcome agents detect suspected behavior patterns they may, under Desktop permission, ethically interfere with those users inviting them to integrate autonomous PAG's for their own sake. The e-membrane suggests and induces a win-win scenario to both sides of it.

References:

[1] *Logical Approach to Building Agents, Active Databases and Workflows: Representing and Reasoning about Actions*, by Chitta Baral, Jorge Lobo, and Richard Scherl. PDF format, from Michel Chitta Baral, from the Computer Science Department of University of Texas at El Paso, USA.

[2] *Semantic Networks For Conceptual Analysis*, SENECA, by Ernesto García Camarero, J. García Sanz y M.F. Verdejo of the Centro de Cálculo, Universidad Complutense de Madrid and Universidad Politécnica de Madrid, Spain 1980.

[3] *Web Searching Agents*, from AAAI, http://www.aaai.org/AITopics/html/webagent.html

[4] *Search Engines: Evolution and Diffusion*, a PDF document of 174KB, from Stephen Arnold and Harry Collier, talks about the industry of searching.

[5] *Web semantics*, from Tim Berners Lee, The Semantic Web, by Tim Berners-Lee, James Hendler and Ora Lassila.

[6] *Artificial Intelligence Tutorial Review*, from University of Toronto, CA, developed and compiled by Eyal Reingold and Johnathan Nightingale.

[7] *Overview Map of CIRL*, Reasoning, Knowledge Representation, and Applications, from University of Oregon, CA, 1999.

[8] *Papers on Analogy and Similarity*, http://www.qrg.northwestern.edu/papers/papers.htm, from Qualitative Reasoning Groups, from Northwestern University.

[9] John F. Sowa, *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, http://www.jfsowa.com/krbook/, Brooks Cole Publishing Co., Pacific Grove, CA, ©2000. Actual publication date, 16 August 1999.

[10] *Chaos and Complexity in Social Systems*, http://www.hehd.clemson.edu/complex/Cmplxdex.htm, from *University of Clemson.*

[11] *Knowledge Discovery In Databases*: Tools and Techniques, by Peggy Wright. This work is funded by U.S. Army Corps Engineers Waterways Experiment Station, Vicksburg, MS 39180

[12] *Darpa puts thought into cognitive computing*, http://www.eet.com/at/news/OEG20021209S0062, Advanced Technology, Dec 2002, by R. Colin Johnson.

[13] *Recent Expert Systems Applications*, from AAAI, 2003.

[14] *How Darwin-FIRST Agents work*, Intelligent Agents Internet Corp, First Human Knowledge Prototype, Oct 2003.

[15] Infomaps, *Some Information Maps Models*, Intelligent Agents Internet Corp, First Human Knowledge Prototype, Oct 2003.

[16] *The Web as a Global Market*, FIRST Desktop, Intelligent Agents Internet Corp, First Human Knowledge Prototype, Oct 2003.