Usability of the professional e-services: what information sciences can learn from neighboring sciences

Gábor Vitályos

Vitályos Consuting, Budapest, Hungary
consulting@vitalyos.hu

Abstract- **The lack of a strict conceptual basis, terminological disorders and misunderstandings, difficulties in documentation -- these all are obstructing more and more portal building in all phases: design, implementation, application and application support, audit and training.**

In this paper we continue our research in the usability of professional e-services. We assume that usability is an important chapter in the Cognitive Info-Communication science, because the client should be able to recognize without much effort and waste of time what he/she sees on the screen: the structure of the portal, the content table, data panels, the forms, the dialogue states.

This paper gives a survey of the neighboring sciences, and gives an outline of the not researched, unelaborated areas the research of which would be useful for the area of professional e-services.

So this paper is not about the results of a research in the HCI, but it outlines further research.

I. Introduction

Keywords: e-government, HCI (Human Computer Interaction), usability, IA (Information Architecture), software technology, Internet, ontology, OO (Object Oriented) technology, DFU (Design for Usability), CMS, constructed languages

Terminology

Portal: 'electronic content', 'e-content', 'electronic service', 'e-service', 'Internet service' are used here as equivalent terms. We use the term *portal* for all of them.

URM (Usability Reference Model): model published in [3]. Its more correct name is *IARM* (Information Architecture Reference Model). For the simplicity we use the URM.

URM philosophy: we often refer the '*URM philosophy'* or 'requirements'. The consistent summarization of this philosophy and its 'requirements' will be in a subsequent paper.

IConS (Interactive Contents & Services): the term for work of the planned, hypothetic new SMC technology complying with the URM our requirements.

Virtual space and *object*: the structure of a *portal*, and the things what the user sees in the virtual space as we defined in [1].

II Backgrounds

The Usability Reference Model

The [3] describe the V0 of the Usability Reference Model. It can be a base of a conceptual basement i.e. the ontology of the HCI. To make easy the reader's work, here is a survey of the high level of the Model:

We divide the interactive information systems into *layers*, from the point of view of man-machine interaction. The Table 2 shows the top level hierarchy of HCI in the URM philosophy, i.e. as they may be built on each other *in our mind*.

7.	Conduciveness layer	Is the mission useful for the intended audience?
6.	Process control layer	Is the business or service logic correct?
5.	Synopsis layer	Are the connections of the outer world clear?
4.	Domain semantics layer	Is the logical structure of the domain clear?
3.	ICT semantics layer	Is the logical structure of the portal clear?
2.	Simple objects' layer	Are the simple objects operable correctly?
1.	Perceivability layer	Are the objects in the screen recognizable?
0.	Physical ergonomics layer	How I feel myself using the device?

Table 1: The top level of the URM model hierarchy

Each *layer* refers to some sort of the functionality – on one hand to human cognitive function, on the other to the software functions. The main considerations to establish the boundary between *layers* as follows:

A. The layers identify the different areas of *human perception requirements*, the *software technologies* and the service *responsibilities*, and demarcate them from each other.

B. The different software layers are stratified on and separated from each other. The main separation is between the 3-4 layers, the ICT platform and the application *domain*.

C. The requirements and compliancy check list of the layers can be independent from each other.

D. We are intended to establish the requirements in that structured and a strict way that it can be the base of a strict layer based audit and DFU methodology, and a layer based portal building CMS software.

E. The URM-based audit methodology must work from down to up. E.g. if user has problems in the layer 1 in recognizing the objects, this causes problems in using the layer 2 in using them, whether the layer 2 complies or not.

This last point makes the URM similar to Maslow hierarchy of general human needs. If the need of a lower level does not comply, the higher level needs may lose their importance.

Note, that the layers don't mean any sequence of the time. The user may recognize a well known logical structure of the *portal* in the 3rd layer, than the wrong-formed logo in the 1st layer. Moreover: general requirement that the user has to recognize the conduciveness in the 7th layer before the complicated structure in the 3rd layer of the *portal*.

Emphasis, that this model is not a predictive one, which would be to be verified by experiments. It is a descriptive and definitive model, intended to be used for constructive works. The proof of its usefulness will be the successful construction of the audit methodology, the DFU methodology, and finally a CMS technology.

Areas where we have scientifically correct solution

It will be very instructive to have a survey of the scientific disciplines of the ICT produced software functions which is wide-spread around us as part of the desktop platforms. Based on the [3] we give a survey here:

The 3rd generation programming languages, in the 1960s. ALGOL, C, PASCAL. Note, that the countless languages born later on - some of them called 4th generation language - are not exact construction, or are not wide-spread in any platform.

The inter-process communication, established by Dyksta, in 1968.

The relation database management, established by the Codd model, in 1969.

The communication networks, based on OSI model from the 1970s.

The cryptography, the digital certifications, the 1970s. The RSA is from 1976.

We can see that these areas of ICT have professional and scientifically established software product, having the next features:

- have strict mathematical bases
- are standard part of the platforms, either being integrated into a platform, or being a product
- have a well known product name and responsible support
- are de facto technical standard, some are canonized in ISO.

On the other hand in the HCI area we don't have such scientifically established solutions. We only have something can be told as:

The "windows standard", from the 1970s. This is the ad hoc name used in [3] because it has no widely-known name. It is an ad hoc construction, containing the

- windows management,
- the menu philosophy,
- the pointing device,
- the low level communication tools as radio buttons, writable fields, etc.

Moreover, there is no project to develop scientifically established solution for the HCI. The software developing actors all over the World work either based on their own ad-hoc HCI solution, or based on that 40 years old "windows standard" of poor abilities, without strict scientific base we sow above.

III. What ICT Can Learn From Other Sciences

1) Lessons from the psychology - the ergonomics

The ergonomics is the most researched and elaborated chapter of the HCI. It deals with the screen and the input devices. Some requirements are fixed by the W3C consortium.

The system designers already are familiar with its principles, the ICT have learnt from it.

However, we lack of its systematic establishment. The [1] gives an axiomatic approach of the Object Permanency Principle; which was an unelaborated and unknown part of the ergonomics.

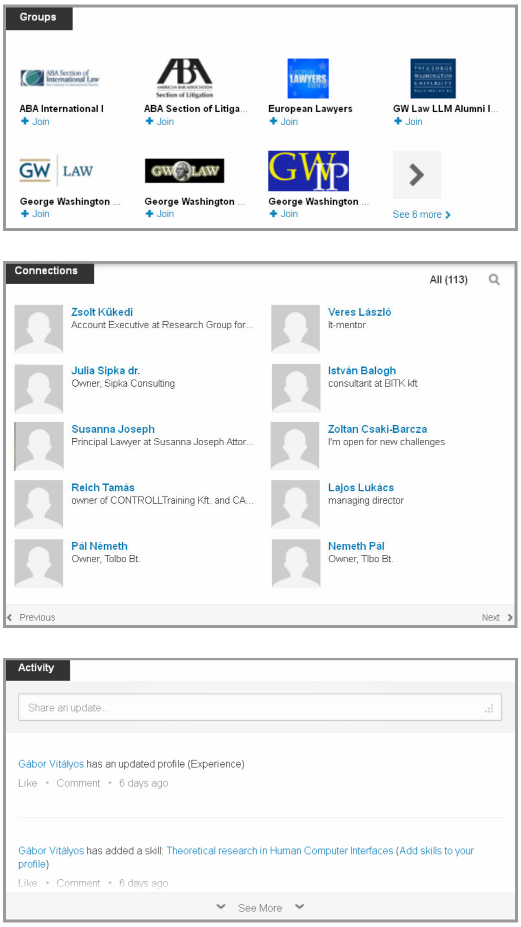
2) Lessons from the linguistics - the dialects have equal rights

In the Fig. 1. we see that an interactive operation – the positioning the windows before a greater object, i.e. paging, scrolling, shifting, etc. – is implemented in 3 different way in the same portal.

We consider these implementations as dialects. All the three implementation can work, but their mixing in the same portal such a way may disturb the user.

By the opinion of the system designers this mixing gives variety to the screen, so helps the user to recognize the screen, to find its way around the portal. And the URM philosophy says, that the variety and the recognizeability must be offered via the variety of the background and the frame of the windows, not by such kind of the mixing.

Then, the URM 2 level suggests that for the positioning operations – and for all oprations – we have to develop standard dialects, which can be selected, but can't be mixed within the same portal.



Groups

ABA International I
ABA Section of Litiga...
European Lawyers
GW Law LLM Alumni I...
George Washington ...
George Washington ...
George Washington ...
See 6 more

Connections

All (113)

Zsolt Kükedi
Account Executive at Research Group for...

Veres László
It-mentor

Julia Sipka dr.
Owner, Sipka Consulting

István Balogh
consultant at BITK kft

Susanna Joseph
Principal Lawyer at Susanna Joseph Attor...

Zoltan Csaki-Barcza
I'm open for new challenges

Reich Tamás
owner of CONTROLLTraining Kft. and CA...

Lajos Lukács
managing director

Pál Németh
Owner, Tolbo Bt.

Nemeth Pál
Owner, Tlbo Bt.

Previous
Next

Activity

Share an update...

Gábor Vitályos has an updated profile (Experience)
Like • Comment • 6 days ago

Gábor Vitályos has added a skill: Theoretical research in Human Computer Interfaces (Add skills to your profile)
Like • Comment • 6 days ago

See More

Fig. 1. The 3 different dialects in one portal

3) Lessons from the linguistics - HCI as 'a priori constructed language'

See a survey of the constructed languages in [4]. The HCI will be a special one intended to interact with the virtual objects and virtual actors. (NB: users operate on the virtual objects and communicate with the virtual actors, cf. [2].)

Let's consider the two basic types of the languages: the nominative and the ergative languages (see [5]). We can find some analogy between the nominative languages and the object oriented HCI, or between the ergative languages and the command oriented HCI.

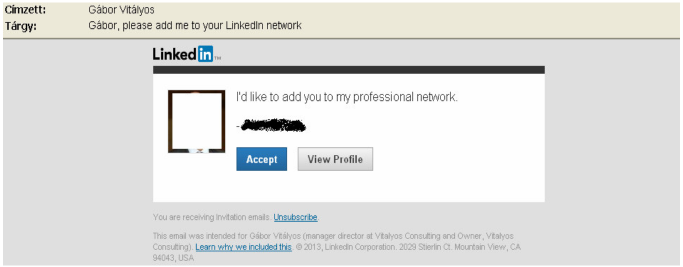
We can expect that there can be more analogy. What is the interactive *sentence*, what are the *paragraph* and the *chapter* in a HCI dialogue? It would be useful to find the analogs of these concepts in the HCI.

(Here we establish the hypothesis that the following definition will be suitable for the notion *sentence* in the HCI: the *sentence* is the sequence of the interactions while the completion software performs a *transaction*. Consequently the *sentence* is uninterruptible: it can be closed by COMMIT if the interaction succeeded or must be aborted by ROLLBACK if it failed[1]. This hypothesis must be verified in the future, if it complies with the URM philosophy or not.)

It is well known that to express complicated, complex things (here: dialogues) we need some sort of language as a tool. The *"windows standard from the 1970s"* is not a proper language. Therefore we expect that the practice or the theory of the constructed languages would enrich the HCI discipline.

4) Lessons from the pragmatics - who speaks to whom

A tentative formulation of pragmatic principles has been given in [2] illustrated by examples. Let us take Fig. 2.



Címzett: Gábor Vitályos
Tárgy: Gábor, please add me to your LinkedIn network

I'd like to add you to my professional network.

- [redacted]

You are receiving Invitation emails. Unsubscribe.

This email was intended for Gábor Vitályos (manager director at Vitalyos Consulting and Owner, Vitalyos Consulting). Learn why we included this. © 2013, LinkedIn Corporation. 2029 Stierlin Ct. Mountain View, CA 94043, USA

Fig. 2. The disturbing sentence in the dialog box

We point out 2 problems in it.

a) In the *message object* the sender asks me to add her/him to my network. But in the dialog box somebody would like to add me (or whom?) to her/his network. What do I finally accept?
The latter sentence is unnecessary, moreover, disturbing. It would be better to have a 3rd button: 'Decline'.

b) What can I unsubscribe? If it means the invitation e-mails – as is suggested by the text and the situation – the correct text

[1] COMMIT and ROLLBACK are functions of the completion software. Naturally, the user will have to recognize both the beginning and the end of a sentence.

of the command would be: ‘Unsubscribe them’ or ‘Stop receiving invitation e-mails’.

More generally, the verb forms read on the screen may cause confusion. Reading the imperative, sometimes the user doesn’t know, who calls upon whom. Reading the conjugated verb, the user has to find out who are ‘we’.

If the *portal* designer keeps the axioms of pragmatics, these problems can be avoided.

5) Lessons from the semiotics – the world of icons and signs

The software designers have constructed innumerable icons in the different applications, used as commands by clicking. There are some well known smilies or emoticons, making the texts more expressive.

We know a convention of generating a pointer icon from a simple icon, as the Fig. 3. shows it.

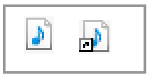


Fig 3. A convention for generating new icons

There are unexploited possibilities for the HCI in the developing conventions - namely grammar - for the generating new signs/icons for both in the texts and in the commands.

6) Lessons from public administration

Contemporary digital administration soon will be limited to what people can see on the screen and they can do on their terminal. If e-service would fail, public administration would fail. Because the primary function of e-services is to fasten and facilitate administration.

But computer science is a technological discipline at the same time, for which logical conceptualization, exact terminology and transparent execution of service processes are equally important. These values must also be transmitted to the public. Professional users of domains as well as non-professional users of the services should be trained for that. That is the way by which they can contribute to making complex and complicated cases simpler and easily manageable.

But the effect nowadays seems to be almost the opposite: complicated problems become even more complicated when we try to solve them by technological systems of poor usability.

7) Lessons from the library sciences - the ontology

The library sciences developed classification methodologies, used already before the computers. These methods are widely used in software industries as some kind of database, thesauri, semantic technologies as the ontology, etc. for classify, search and manage large amount of data.

But these methods are hardly used to manage the object of the HCI itself, the objects, we have to recognize on the screen. The [3] gives an experimental classification of the subset of the HCI objects, i.e. e subset of the HCI ontology.

8) Lessons from the sociology – popular vs. professional user behavior

The professional e-services are not for everybody, but it is for the people of high qualification, the decision-makers, etc., who use it as day-to-day working tool. We lack the research on the difference of the professional and the popular behavior of the users.

9) Lessons from the sociology – usability as the generation gap

Nowadays *younger and younger people develop working tools for older and older people.* It is far larger problem than that of the e-inclusion for the retired senior citizen, and differs from that for the people with disabilities or handicap. These are everywhere discussed well known problems.

Here we mean instead that a manager of 40s uses a tool, designed and made by a teenager. We don’t see researches assigned to this.

10) Lessons from the activity theory - activity, action, operation, etc

The idea of this theory comes from Leontiev [6].

It is the more promising and unelaborated area the HCI would exploit. The final vision of the URM is a platform, integrated by an activity manager. It runs in the client’s environment as part of the operating system, dealing with the *portal* independent levels of the URM, as is shown in the Picture 8. in [3]. It can be regarded as the successor of the personal assistant applications.

The usability of a portal is not its intrinsic quality. *A portal complies with the usability requirements of the URM if (and only if) the client’s workstation with that manager can connect to and work with it.*

IV. Summary, Conclusions

What the neighboring sciences have to understand from ICT

Neighboring sciences ought to understand that professional ICT is governed by business, which cannot wait for elaborated results of scientific research, it does not require too much conceptual consideration of technologies. It follows its own logic and it is polluting scientific clarity of our virtual environment with superficial, messy and faulty e-services. So sciences ought to be more initiative in this problem. E.g. it should press EU legislation to place recommendations on a proper scientific foundation.

Further on sciences must be ready to compromise. Scientists for instance like to say that measuring comes first and then you can induce the notions of usability from the results.

This is a misunderstanding: computational science is not a natural science, and it works just the other way round. For measurement you need pre-determined categories and tools exactly calibrated. And they are not provided just because of missing software technology. Exact measuring can be done only in limited specific fields, e.g. in ergonomics. "In vitro" measuring portals and experimental settings for some specific research have already been prepared. Also there are specific measuring points placed in some portals for "in vivo" observations, but the usability of a real service cannot be measured exactly. A clear cut conceptually founded technology of a service, which we are just urging here, could just serve as a subject to exact measurements.

Although the study of HCI is not an exact science, because experimental results – for the time being or in the near future, and probably never -- can lead to correct abstractions for the requirements of usability, still software technology should be as precise and elaborate, as possible.

Exactness and conceptual elaboration are certainly among the most important requirements for technologies of professional portals. This is the biggest challenge for computational science nowadays.

Present day practice, when services are built up according to ad hoc ideas from adolescents of 140 IQ, who never read a textbook or a standard, or following primitive CMS technologies provided by providers, when younger and younger people are preparing software tools for aging users, but according to their own skills and tastes, is carrying a severe sociological risk.

NOTE

Present paper is part of a greater research project, intended to elaborate a formal ontology of HCI discipline of the professional use.

The 1st paper was the [1] on OPP in the HCI.
The 2nd paper was the [2] on the principles on the pragmatics.
The 3rd one was on the Usability Reference Model that clears the way to build up HCI ontology.
This is the 4th paper, on the required further researches in the neighboring sciences.
The next one is planned to be on Pragmatics and the Activity Theory in the HCI.

REFERENCES

[1] G. Vitályos: *The Object Permanency Principle in the Usability discipline*. Conference on Cognitive InfoCommunication 2011, Budapest.

[2] A. S. Szőlősy, G. Vitályos: *Pragmatics in the Usability discipline*. Conference on Cognitive InfoCommunication 2012, Budapest.

[3] G. Vitályos: *What are to be recognized on the screen?The Usability Reference Model*. Conference on Cognitive InfoCommunication 2012, Budapest. Special issue. In print.

[4] Constructed Languages, en.wikipedia.org/wiki/Construced_language

[5] Havas, Ferenc: Linguistic Typology, 2011, In Hungarian.

[6] Leontiev: *Activity Theory*. http://marxists.org/archive/leontev/works/1978/index.htm