Medical Infobahn for Europe
A. Hasman et al. (Eds.)
IOS Press, 2000

Agent-Oriented Captology for Medical Informatics

Boldur E. BĂRBAT*, Constantin B. ZAMFIRESCU**, Gabriel COSTACHE***

"Lucian Blaga" University in Sibiu, Faculty of Engineering
Dept. of Comp. Sc. and Automatic Control, 4 Emil Cioran Street 2400 Sibiu, ROMANIA
**Phone: +40-69-212523 Fax: +40-69-212716 E-mail: bbarbat@sibnet.ro*
***Phone: +40-69-422424 Fax: +40-69-212716 E-mail: zbc@acm.org*
****Oltchim S.A., Dept. of Computing Centre, 1 Uzinei St., 1000 Rm. Vâlcea, ROMANIA*
Phone: +40-093338475 E-mail: cgabi@oltchim.onix.ro

Abstract. Considering that neither captology nor agent-orientation, are applied in medical informatics, as they could be, the paper presents a broad-spectrum generic architectural framework to support developing adaptive medical applications, based on synergistic correlation between persuasive interfaces and intelligent agents. Their main features are adapted for medical informatics. Lying on this groundwork, the design space for agent-oriented persuasive applications is defined and several guidelines for its main dimensions are given. The approach is instantiated through an agent-based test-bench application, having the purpose to persuade to quit smoking.

1. Introduction

Although they concern the user quite different, both persuasive technologies and agent-orientation (AO) share the feature that, despite being wide-ranging research trends in information technology (IT), their significant potential is insufficiently used for medical informatics (MI). Their synergistic connection has been even less investigated.

Why captology? First the health-care reasons: a) the therapeutic role of persuasion [6]; b) the related, but distinct, impact of credibility [9]; c) the role of deontological ethics [2]; d). Other reasons stem from IT: d) in medical applications the computer can be tool, medium, and social actor [5]; e) the role of interfaces (chiefly, multimodal) [1]; g) the implied anthropocentric approach; h) the unanswered questions of persuasive technologies [5].

Why agent-orientation? The reasons are layered [1]: a) anthropocentric approaches become familiar in MI; b) the interface is crucial; c) its role implies increasing intelligence; d) intelligence in IT applications means, above all, agents; f) getting closer: where intentionality is involved, the agent is the most natural interactant; g) though, even if AO is a widespread trend in IT, it is not yet as pervasive and diversified in MI, as it could be.

The reminder of the paper adapts the captological design space to MI (Section 2); Section 3 outlines a generic architecture for AO captology and gives some guidelines for its design; the approach is instantiated through an agent-based application described in Section 4; conclusions are drawn and future work is hinted to in the final section.

2. Defining the captological design space

We adopt Fogg's definition for captology: "an attempt to shape, reinforce, or change behaviors, feelings, or thoughts about an issue, object, or action" and shift at architectural level, some concepts used in generative programming, such as "design space", "concern",

and "aspect" [4]. So, shaping captological requirements for the application domain aimed at, means to separate the relevant concerns, and to delimit the aspect segments.

- *Levels of analysis.* Regarding the population aimed at, the intervention levels are: individual, family, organization, and community (e.g., drinking problems are better dealt with at individual or family level, but environment ones can be handled also at community level).

- *Intentionality.* While behavioural changes may be side effects of other technologies, in Fogg's definition "an attempt to" implies intentionality (with the connotation of "planned effect"). However, from a user standpoint, it depends on the assumed stance (in Dennett's sense): if, due to system complexity, a mechanistic or functional stance is not handy, the computer is empathized as an "intentional system". Primarily in the agent age, intentionality is not trivial and has to be focused on, i.e. has to be seen as design-space dimension.

- *Credibility.* Has a noticeable impact through both its key components: *trustworthiness* and *expertise.* The types of credibility are [9]: presumed, reputed, surface, experienced.

- *Ethics.* We refer to it "as a rational, consistent system for determining right and wrong, usually in the context of specific actions or policies" [2]. The aspects express degrees of rigor, from strict deontological ethics (total intransigence), to act-based pragmatism ("pro and con" ethics); the midpoint: rule-based utilitarianism (rules are set only if always following them is beneficial). In such cases, the aspect of "being open to inspection" is crucial.

- *Persuasive strategies.* We detach the fields where theories come from (sociology, psychology, rhetoric, communication) and basic strategies [5] (e.g., positive/negative feedback, role-play, simulated experiences, surveillance, environments of discovery, personalizing).

- *Form factors.* The forms are [6]: desktop-based (usual/dedicated computers), artefact-based (portable, embedded), environment-based (integrated into user settings, distributed).

- *User profile.* The huge inter-individual diversity (e.g., depending on personality, cultural background, area of interest, age, social status, motivation, level of information, previous exposure to IT) is still amplified in sensitive situations, like those involving persuasion.

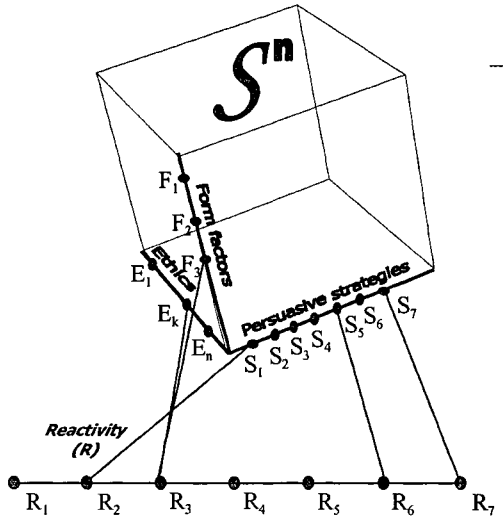
- *User history.* Related to the previous facet, but with distinctive dynamics (e.g., the role of anamnesis in therapy and diagnosis), personal history as design-space dimension has gradually varying aspects. Some of them (regarding privacy) can interfere with ethical aspects (rich history enables potential manipulation of information about the user).

Since the design (sub)space for agents has been defined for MI in [1], the agentsy features [10] will be re-examined in the next section. (The only added dimension is *credibility.*)

3. Generic architectural framework

Lying on the previous groundwork and on the generative-programming concepts stated, defining a design space for agent-oriented persuasive applications allows, instead of describing specific architectures, to set out a broad-spectrum generic architectural framework. The need for generic architectures stems from three nested ranges, all regarding flexibility: general trends in IT; the way they are mirrored in captology; support for application development (to have both a mould for individual applications and a test bed for research). Thus, the directions along which architectonic genericity is imposed are the design-space dimensions. This "meta-architecture" is the common denominator of all specific architectures it is able to create (its instances are standalone, being useless to spot the core they come from).

Generally, dimensions are placed on two (abstraction) levels: captological features and agentsy features (see Figure 1, where, to avoid overloading, only three dimensions of the



	Aspects
R_1	Static input
R_2	Interactive input
R_3	Exception handling
R_4	Transaction handling
R_5	Subsumption architecture
R_6	Rule learning
R_7	Ontology refinement

Legend: E_1 - Rule-based utilitarianism; E_k - Open-to-inspection; E_n - Strict deontological ethics; S_1 - Feedback; S_2 - Role playing; S_3 - Surveillance; S_4 – Simulated experiences; S_5 – Personalizing; S_6 - Environments of discovery; S_7 - Virtual groups; F_1 – Desktop-based; F_2 - Artefact-based; F_3 - Environment-based.

Figure 1. Design space for agent-oriented captology

captological design space, S^n, are shown). The aspects selected on the hypercube edges become requests for the (lower) level of reactivity aspects; e.g., an environment-based application (aspect *F3*), needs (at least) the reactivity reached through exception handling (*R3*), whilst applying the persuasive strategy of personalizing (*S5*) requests rule-learning (*R6*).

Other dimensions (captological as well as agentsy features) are added and handled alike.

The agentsy design-space dimensions [1], adapted to captology, become (in brackets are some design guidelines for developing persuasive applications):

- *Autonomy* and *continuity in time*. (All agents are created as independent tasks.)
- *Social ability*. Since agents must persuade the user, the interface should be comprehensive (not merely conversational, but multimedial, and even more often, multimodal).
- *Reactivity*. The first environment is the user; the second one depends on the type of application. (So as to react promptly, the agents or the applications are interrupt-driven.)
- *Pro-activity*. Specialized knowledge is where goals and initiative come from. (The agents use persuasive strategies, and adapt to preset or dynamically learned user profiles.)

The features stated above fulfil the requirements for the weak notion of agent. To approach the strong notion, some ideas to supply the "outlets" for features to be "plugged in" are:

- *Mobility*. It must be considered (only) for future research. (Despite being yet static, most agents should be coded in Java and put up in a WWW environment.)
- *Benevolence* and *rationality*. Although unclear, such concepts are relevant because of the "non nocere" principle. (Agents have to reject actions they believe potentially harmful.)
- *Credibility*. Practically, the interface agents give the overall credibility of persuasive applications. The *trustworthiness* sub-dimension is provided chiefly by the information accuracy. (All tables, graphics etc. shall be up-to-date and open to inspection.) The *expertise* sub-dimension is a corollary of reactivity and pro-activity. Credibility can be increased through *emotivity*. (Agents should simulate the feeling of satisfaction or disappointment.)

This manner of anticipating increased complexity is useful in many situations. (Although a single agent is enough to start with, it shall be cloned to prepare for multiple agents.)

Developing a diversified "agent library" in order to choose the needed building cells, would enable easily switching from a variant to another (i.e., changing aspects of one or se-

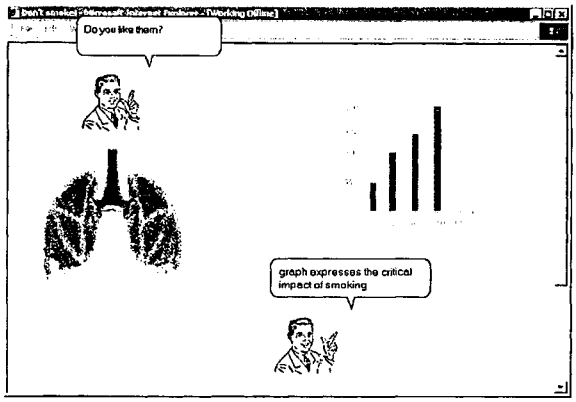


Figure 2. Snapshot of the application instance

veral dimensions). Moreover, if the granularity is refined, then the "switch" may be replaced by a "potentiometer" (carried out via dynamic priorities for the run-time threads).

4. Application instance

The approach is instantiated through an agent-based test-bench application with a multimodal interface, having the purpose to persuade a smoker to quit this habit (see Figure 2). This aim was chosen, as it is a relevant health-care application, (with many potential users), is ethically uncontroversial, has a precious arsenal of strategies/ tactics (applied also in traditional therapies), has been approached in captology only as a handheld, self-help technology [6], and, allows multimodal experiments (by means of a simple smoke detector).

The communication issues are solved through: a) a framework supporting a full range of multimedia components for a personalised content (such as XML) [8]; b) a set of programmable software services that support presenting interactive animated characters (Microsoft Agents) [7]. This package of programmable software services eases: to set in advance the interaction degree at system level (installation phase); to clone the agents in a simple manner (scenario generation phase); to add multimodal communication (interaction phase). Thus, the dialogue is not limited to usual "WIMP"- style: besides accepting mouse and keyboard input, the agents respond using synthesized speech, recorded audio, and animation. The requirements for the persuasive agents are: a) to play their role according to the script; b) to alert the user if a smoke is detected around the computer; c) to stay out of sight (and mind), when the script don't needs them; d) to be easily launched and stopped.

The scenario is based on the psychological theories of *stimulus - response*, *social judgement* and *consistency*. The agent shows the image of a middle-aged man, soberly clothed, using in its persuasive strategy both rational arguments and emotional ones. In order to stimulate the activity of each cerebral hemisphere, pictures and emotional elements are placed on the left side of the screen whereas verbal-logical information is placed on the right side (e.g., when the agent presents a graph expressing the correlation between smoking and cardiovascular diseases, it will be placed in the right top of the screen). Because the move is another contrast element that affects the attention, the agent will move on diagonal from top-right to bottom- left; in order to improve message understanding, the "Z flow" (i.e., the natural look trajectory when a page is gone through) is underlined by graphical elements.

The only controversial topic was to accept or not subliminal messages (the debate relates to the deontology of clandestine persuasive practices). Thus, for both testing purposes and ethical reasons, the subliminal message "DON'T SMOKE" (written black on yellow for simultaneous chromatic contrast with maximal effect), is presented also explicitly; there are different exposure times: very short (subliminal), short (liminal, the message is easily perceived), long (the message stays on the screen). Another (multimodal) look of the subliminal message: upon a beautiful landscape on the entire screen, and accompanied by classical music, emerges the image of a white ashtray with a red flower inside, on a light background carrying the message written with the same colour as the background, but a darker tint.

Reactivity is enhanced replying to cigarette smoke. Here the meaning of "interrupt" migrates from hardware to the very user, because, after detecting smoke, no matter what program is active, the agent with the throaty voice comes in to convey its warnings. The signal generated by the smoke detector is transmitted to a serial port of the computer – that is the agent's "nose"; a thread reads the serial port and wakes-up the agent). Other examples of implementing agentsy features: *trustworthiness* (all graphics relating lung cancer to smoking are updated and open to inspection); *expertise* (in a manner similar to "error-logging", the agents trace the dynamics of the user's smoking habit); *emotivity* (if the patient starts smoking, the agent, without being upset, shows disappointment). Although a single agent suffices, it has been cloned (the new agent is identical, to the old one, except its voice).

A more detailed description of a previous version of this application is given in [3].

5. Conclusions and future work

Since patient motivation is of paramount importance in any medical treatment, in some therapies, it is worthwhile – despite the (mainly, ethical) risks involved – to explore ways to reflect the advantages of dialectical argumentation; thus, captology is justified in MI.

From the IT standpoint, a generic architecture cannot be comprehensively evaluated after just one instantiation, no matter how relevant or successful it could be. Nevertheless, at this stage, the approach is promising for the following reasons: a) the application endorses the meta-architecture it originates from. b) Designing it according to the given guidelines was straightforward and unambiguous. c) The trend toward increasing intelligence embodied in every MI application (and mainly in captology) is here to stay. Currently, that means AO.

Future research objectives are: a) investigating the influence that the dependability of the conveyed information (exact, approximate, fuzzy, and uncertain) can have on credibility; b) refining the captological design-space for virtual groups strategies [11]; c) enhancing the interaction, by elevating the agents to the role of personal assistants.

Specific short-range targets: a) investigating the role of cloning as a means to implement polymorphic agents; b) refining the interaction by increasing the agent expertise (e.g., the agents shall be able to learn from tracing the dynamics of the user's smoking habit).

References

[1] Bărbat, B. (1999). Agent-Oriented Architectures for Medical Applications. *Conferina de informatic medical MEDINF'99*, Sibiu.

[2] Berdichevsky, D., E. Neunschwander (1999). Toward an Ethics of Persuasive Technologies. *Communications of the ACM*, **42** (5), 51-58.

[3] Costache, G., C.B. Zamfirescu, C. Sas, B. Brbat (1999). Agent-Oriented Persuasive Interface. *Conferina de informatic medical MEDINF'99*, Sibiu.

[4] Eisenecker, U.W., K. Czarnecki (1998). Generative Programmierung. In: *horizonte*, No. 13, Okt.

[5] Fogg, B. J. (1999). Persuasive Technologies. *Communications of the ACM*, **42** (5), 27-29.

[6] King, P., J. Tester (1999). The Landscape of Persuasive Technologies. *Communications of the ACM*, **42** (5), 31-38.

[7] Microsoft Corporation. http://www.microsoft.com/workshop/imedia/agent/agentdevdl.asp

[8] Pallmann, D. (1999). *Programming Bots, Spiders, and Intelligent Agents in Microsoft Visual C++*. Microsoft Press, Redmond, WA.

[9] Tseng, S., B. J. Fogg (1999). Credibility and Computing Technology. *Communications of the ACM*, **42** (5), 39-44.

[10] Wooldridge, M., N. R. Jennings (1995). Intelligent Agents: Theory and Practice. *Knowledge Engineering Review*, **10** (2), 115-152.

[11] Zamfirescu, C.B. F.G. Filip (1999). An agent-oriented approach to team-based manufacturing systems. In: H. Van Brussel and P. Valckenaers (Eds.). *Proceedings of the Second International Workshop on Intelligent Manufacturing Systems*, Katholieke Universiteit Leuven, p. 651-658.