Enacting an Agent-based Digital Self in a 24x7 Web Services World

Steve Goschnick

IDEA Lab
Department of Information Systems
University of Melbourne, 3010, Australia
gosh@staff.dis.unimelb.edu.au

Abstract. As broadband access to the Internet becomes pervasive, the need for a 24 hours a day, seven days a week (24x7) interface within the client devices, requires a level of sophistication that implies agent technology. From this situation we identified the need for a user-proxy++, something we have termed the *Digital Self* that acts for the user gathering appropriate information and knowledge, representing and acting for them when they are off-line. With these notions in mind we set about defining an agent architecture, sufficiently complex to deal with the myriad aspects of the life of a busy time-poor modern user, and we arrived at the Shadowboard architecture. For the theory, for the model of mind, we drew upon the *Psychology of Subselves*, a modern strain of Analytical Psychology. For the computation engine we drew upon Constraint Logic Programming. For the hundreds of sources of sub-agency and external intelligence needed to enact a Digital Self within the 24x7 Internet environment, we drew upon the Web Services paradigm. This paper presents the theory, the architecture and the implementation of a prototype of the Shadowboard agent system.

1 Introduction

There were two driving motivations for the research and development outlined in this paper. Firstly, our continual connection to the Internet 24x7, usually as a by-product of broadband, leaves us connected to a world of disparate services and external computational intelligence, largely under-utilized while we are either: asleep, away from our desktops, or otherwise occupied. Secondly, the time-poor lives that many of us live, in our busy modern modes of existence, where we each have numerous and increasing roles, each with associated obligations, needs and desires, often operating concurrently – eg. one may be a *parent*, a *teacher*, a *domain expert*, an *artist*, a *traveler*, an *investor* – sometimes all at once. We have set out to harness the untapped potential of the first, to improve the quality of life observed in the second.

By creating and then using a sufficiently complex agent architecture on the client device, we are able to harness external services and growing intelligence out on the web, in a systematic and synergetic way, providing a serious bid to buy back some quality time for the well connected user. We term this fully configured system for a given individual user, their *Digital Self.*

In Section 2 we outline the theory by introducing the *Psychology of Subselves* as the complex model of mind from which we inferred the Shadowboard agent architecture. In Section 3 we introduce the *Shadowboard architecture*, looking at its different components. In Section 4 we briefly look at the *methodology* still being evolved, that is designed to get the best match for an individual user, of a combination of sub-agents that will in concert, represent their Digital Self. In Section 5 we outline the development of a prototype. It includes: *ShadowFaces* an agent-metaphor interface which is more appropriate for a Digital Self operating 24x7 than the document-centric desktop metaphor; the use of CoLoG a constraint logic language as the computational engine; and the wrapping of hundreds of SOAP web services into the sub-agents that make the Digital Self viable and useful in the near term.

2 Theory

The BDI agent architecture [14] is one of a number of architectures that enact deliberative agents. BDI calls upon the mentalistic notions of *Beliefs*, *Desires* and *Intentions* from *Folk Psychology*, as abstractions to encapsulate the hidden complexity of the inner functioning of an *individual* agent. Folk psychology is the name given to everyday talk about the mind, the vocabulary we use to explain ourselves such as: *beliefs, desires, intentions, fears, wishes, hopes*. As people, the use of such language gives us an efficient method for understanding and predicting behaviour, although the model of mind it presents is very course-grained and therefore not suitable for a complex Digital Self. However BDI sets a precedent in its use of a psychology other than cognitive psychology, as fertile ground for computational agent models.

Main exponent:	**Sigmund Freud**	**Carl Jung**	**Roberto Assagioli**	**Hal Stone & Sidra Winkelman** (Psychoanalysis & Psychosynthesis)
Technique	Psychoanalysis		Psychosynthesis	Voice Dialogue
Model divisions of the human psyche	Ego	Persona	Centre	Aware Ego
		Self, self	Self	
	Super Ego	Higher Self		Protector/Controller Inner Critic
	Id *repression*	The Shadow		Several *Disowned Selves*
		Anima/ Animus	Many sub-selves	Pusher, Pleaser, Parental and many other subselves
		Archetypes	Evolved Subselves	

Fig. 1. Lineage of sub-personality exponents and some of their divisions of the psyche

Western psychology has many rich branches from which one could draw models of agency including: *analytical psychology, humanistic psychology, developmental*

psychology. Shadowboard, the agent architecture we have developed, draws upon *analytical psychology*, in particular a contemporary stream encompassing refinements of Freudian [15], Jungian [9] and Assagiolian [1] concepts. The approach is known as the *Psychology of Subselves,* an attempt to understand the whole personality of an individual - in order to model consciousness, deliberation and action [15; 17].

Figure 1 above represents a scant overview of the divisions of the psyche that different influential analytical psychologists identified and named. Beginning with Freud about 100 years ago, followed by Carl Jung these psychologists identified substrata within the psyche of an individual. As we move across the columns in Figure 1 from left to right, we move forward to contemporary theory on sub-selves within the psyche. The most easily identified sub-selves of a person often align with the *roles* in their lives – *teacher, parent, domain expert, artist, sibling, traveler,* etc. A comparative description of the different psychologies we considered and the subsequent development of the Shadowboard *architecture*, is detailed extensively elsewhere [4].

3 Architecture

Figure 2 below is a graphic overview of the Shadowboard architecture, collectively representing an individual *whole agent* made up of numerous sub-components – the structural implications are inferred from the Psychology of Subselves. In the centre of the agent is the *Aware Ego Agent* – the dominant sub-agent in the whole cluster of sub-agents. In the figure, the Aware Ego Agent is surrounded by eight first-level sub-agents, diagrammatically drawn as circles the same size as the Aware Ego Agent. Five of these example sub-agents are not nested any deeper (i.e. sub-agents can be clustered recursively), while the other three have clusters of circles within them, representing a second-level of sub-agents, grouped into numerous Envelope-of-Capability (EoC).

Each *EoC,* of which there are arbitrarily eight in the figure, represent different areas of expertise that a particular whole agent embodies. As such the whole agent could perform a number of consecutive and diverse tasks, depending on what goals via what roles it has taken on in the outer world.

Each EoC contains a number of sub-agents with similar capabilities, but each different from it neighbors in some specialized way. At the two far ends of an EoC are two diametrically skilled agents - in Figure 2 the two are adjacent, one is white the other is black, separated by a dotted radial line.

One is the *basic reactive sub-agent* a purely reactive agent [2] with a hard-wired rule-action mechanism and no deliberative capability. It is usually called upon within a particular EoC when time or other resources are severely constrained.

At the other end of the scale is an *archetypal sub-agent*, one that has maximum deliberative ability used when time and other resources are plentiful.

The *aware ego agent*, as well as each of the other sub-agents (*delegation sub-agents*) that are shaded as spheres, have *knowledge of the capabilities* of sub-agents

in their EoC. They use this knowledge to select the appropriate sub-agent to achieve the particular goal that has been sent their way from higher up in the recursive hierarchy.

When a sub-agent has been found lacking in capability to achieve a specific intention, or when an external (and available) agent matches the particular specialty better than any internal sub-agents, an external agent can be called upon as if it were an internal sub-agent. This process is termed *disowning a sub-agent*.

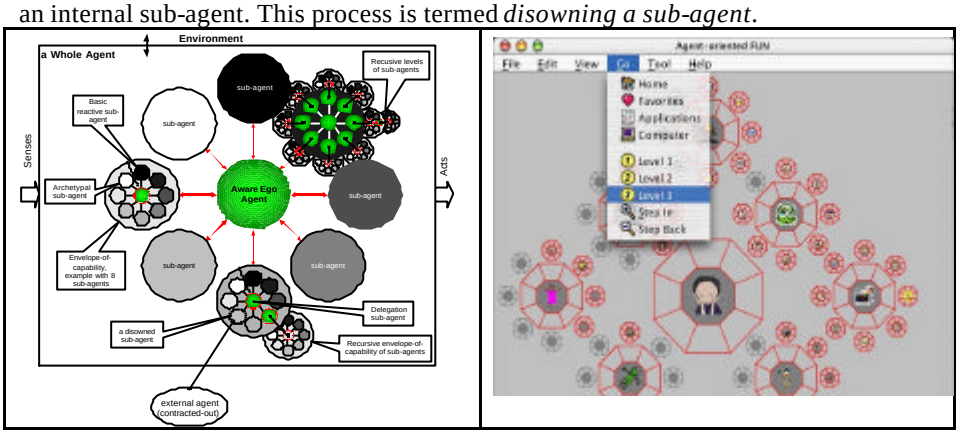

Fig. 2. The Shadowboard Architecture and the Implemented Prototype

3.1 Sub-Agents as Subselves

The mentalistic notion of a *subself* at work within the psyche of an individual, is a metaphor for the *sub-agent* of a Shadowboard agent. To broadly place this work in context of research upon multi-agent systems (MAS) - most MAS can be described as *inter*-agent systems. In contrast, the Shadowboard theory and architecture is an *intra*-agent system, one enabling the incorporation of many components that together represent one *whole* agent, albeit a very sophisticated one. Such a *whole agent* built upon the Shadowboard architecture – the *Digital Self* – seen from without, should be seen as a fully *autonomous* individual agent, one compatible with contemporary definitions of agency, such as that of Wooldridge [19].

Unlike the autonomous whole agent, the inner sub-agents are semi-autonomous or even totally subservient to the *Aware Ego Agent* - the executive controller within a Shadowboard agent. Sub-agents may themselves be sophisticated agents capable of their own semi-autonomous work, or they may be conventional application programs, expert systems, or even wrapped Web Services as we shall see further down. This is a significant relaxation on the need in most MAS systems (at least in theory) for each individual agent to be a fully functioning autonomous agent. This flexibility in capability provided within a Shadowboard agent, is attained by making all sub-agents ultimately subordinate to the *Aware Ego Agent*.

4 Methodology

Building a Digital Self involves building a comprehensive *user-proxy++* - something much more wide ranging than a user surrogate. In order to do so for the vast diversity of individuals, we are developing a methodology. It begins with the definition of a generic range of sub-agent *types* that are likely to encapsulate the sorts of EoC any given user may want to choose from to build into their Digital Self. This list of Shadowboard sub-agent types include: *DecisionMaker, Manager, Protector, PersonalAssistant, Advisor, Critic, Initiator, Adventurer, KnowledgeSeeker, Logician, Player, Teacher, Mentor, Engineer, Artist, Intuitive, Intrapersonal, ContextSituator.* Most often these sub-agent types represent an *envelope-of-capability (EoC)*, which in turn have a range of like-functioned sub-agents within it, but covering a range of capabilities and/or degrees of agencies from highly deliberative to reactive. An example of sub-agents (their generic names) within one of the EoCs above (ie. Manager) include: *Benevolent* (archetypal), *Conciliatory, Planner, Scheduler, Coordinator, Recycler, Decisive* (reactive). Similarly, there are lists of further sub-agents for each of the other EoCs above. These are fully detailed in [4], which together cover many of the roles and activities an individual may be involved in.

Other EoCs outside this starting range, may draw upon agent ontologies compiled by other people, and can occupy any level within the recursive hierarchy of Shadowboard. For example Hristozova [7] specified a range of *Middle Agents: Matchmaker Agent, Recommender Agent, Mediator Agent, Facilitator agent, Broker agent* – which establish, maintain and complete communications with end agents, in a graduated range of capability. In other words, the methodology for building a Digital Self, is very much about utilizing third party agents and ranges of agents, as sub-agents within the Shadowboard architecture framework.

5 Implementation

Enacting the Shadowboard architecture has been hastened by the use of several existing technologies at our disposal, which we have modified, enhanced and integrated. There are several levels of technology. At the base is a system called ShadowSpaces [6], which is a dynamic model of the hierarchy of sub-agents which are later intertwined with logic programming. There is an interface system called ShadowFaces, which the user uses as both an editor to create and modify their Digital Self, and as an interface to their running Digital Self system. Notifications, filtered information and those decisions that the Digital Self defers back to the user, all percolate up through the interface. There is our underlying constraint logic language parser called CoLoG, which uses constraint logic programming [13], to bind together the various sub-agents into a powerful and dynamic computational system.

5.1 ShadowSpaces - a Dynamic Object System

ShadowSpaces is not unlike the W3C DOM (Document Object Model), and with some effort could be made to be DOM compliant [18]. In our case, it is used to dynamically maintain a hierarchy of objects, which are independent of the way the objects are instantiated and stored in the language the system is created in. ie. Java. As with the W3C DOM, the ShadowSpace tree is *mutable* - able to deal with events that modify the structure of the tree.

5.2 The ShadowFaces Interface

The ShadowFaces interface we have built adopts an *agent-metaphor* [11] (in addition to it being an agent technology), rather than the *desktop metaphor* of most systems today – which we consider to be a severely challenged metaphor in the 24x7 space.

ShadowFaces displays three levels of the hierarchy of sub-agents at a time, and can be thought of as a *lens* that the user can use to navigate through the recursive structure of the Shadowboard architecture. See right-hand-side of Figure 2 above. The advantages of this interface are numerous:

- It has an agent-metaphor rather than the document-centric interface of the desktop metaphor, making it applicable non-desktop systems.
- Up to 589 individual items in the hierarchy can be displayed and are selectable at the one time, versus the use of a tree manager approach (eg. Windows Explorer).
- The shape echoes the graphic depiction of the Shadowboard architecture, reinforcing the *system image* in the user's mind – useful when introducing a new software interface [16].
- The interface can be expanded to full-screen or reduced down to the size of a handheld screen (as low as 160x160 pixels), making it an ideal interface for a number of different consumer devices – PDAs and newer mobile phones.
- It has an eight-way navigational action making it equally suitable to: the mouse, the keypad, function-key pads, and game console joystick interfaces – again making it ideal for 24x7 operation via suitability to different types of consumer devices, each applicable to different locales.
- The same interface can be used to: zoom, navigate, filter and get details-on-demand (see Figure 3 below, editing an Envelope-of-capability).

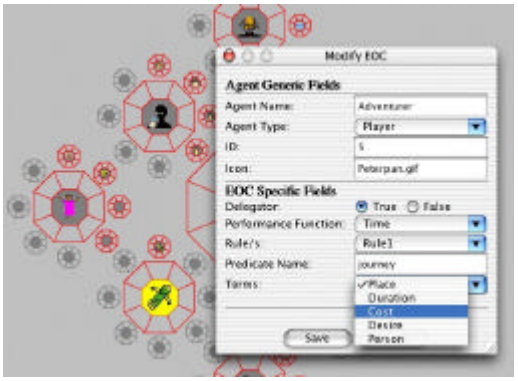

Fig. 3. Editing the current Envelope-of-Capability (EoC)

5.3 The CoLoG Computation Engine

To understand the computation engine and how it inter-operates with both individual *sub-agents* and the *EoCs* in which they are clustered, we examine a small *Constraint Logic Program*, and observe how it is carved up within the Shadowboard system. Figure 4 below is the constraint logic program called *Journey*. It has single *Rule* at the top, consisting of the *conjunction* (represented by commas) of numerous *Predicates*, together with two *constraints*, both of which use the >= operator.

Following the Rule is a series of *ground terms* which conform to the predicates *destination(…), money(…)* and *available_time(…)*, such that: *destination(…)* represents all holiday locations for which a traveler may choose, the second argument is the amount of money it costs, and the third argument is the number of days the travel/holiday package deal includes; *money(…)* represents a number of individuals and the amount of money each has available; *available_time(…)* represents the amount of days each individual has available to travel.

Here are a series of Queries/Goals that the program Journey is capable of answering, expressed in both English and in predicate logic:

- *Which people could go to Brisbane?* journey(**brisbane**, Duration, Cost, Desire, Person)?
- *What places could Jill travel to?* journey(Place, Duration, Cost, Desire, **jill**)?
- *What does it cost to go to Canberra?* destination(**canberra**, Cost, Duration)?
- *How much time does John have to travel?* available_time(**john**, Time)?

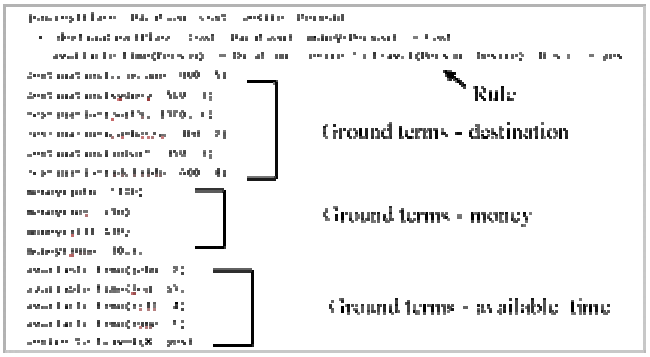

Fig. 4. Constraint Logic Program - *Journey*

The range of queries/goals that this program can answer, is typical of the flexibility of most logic languages.

A Shadowboard EoC holds *rules, constraints* and *performance functions,* with links to appropriate sub-agents next step down in the recursive hierarchy. The Journey example has only one rule, however the EoC may hold several alternative rules, the order of choice guided by the *performance-function*.

Looking at the detail in Figure 3 you will see that the *'Modify EOC'* dialog box, represents the EoC that has program *journey* as its *Predicate Name*. Predicate *journey* has five terms: *Place, Duration, Cost, Desire* and *Person*. It has a rule called *Rule1*.

In this computational system, sub-agents that are not EoCs have a single *predicate* for which they may *get*, *generate* and *manage* ground terms. In our example we have a sub-agent called the *Travel Sub-agent*, which manages just those ground terms for the predicate *destination(Place, Cost, Duration)*.

The standalone example constraint logic program *Journey* represented in Figure 4 is capable of *goal driven processing* only. Shadowboard exhibits both *goal driven* and *data driven reasoning*. Ie. The sub-agents are able to source new/updated ground terms, which in turn trigger the appropriate EoCs to recalculate – which in turn pass their derived terms up the hierarchy to their parent EoC. Although the computation engine within Shadowboard uses constraint logic programming, it is more like a generalized constraint solver in its operation [13].

5.4 Wrapping Web Services as Sub-agents

The ground terms managed by a particular sub-agent are not a static set. A sub-agent may generate them itself. A whole ShadowBoard Agent in the form of a Digital Self can accommodate multiple: Personal Assistant agents, notification, filtering [12] and other agent types, application programs and other less complex processes – given the definition of sub-agency in Shadowboard. However, there is a need for hundreds of third-party sub-agents to make a user's Digital Self, genuinely useful in the near term.

Web Services, particularly in the form of SOAP Web Services, are of interest to us as the primary source of much external intelligence. Several recent papers document other efforts to utilize web services within the agent paradigm of programming [3; 10]. There are hundreds of single/limited purpose web services already available across the Internet and that number is accelerating. Web Services offer a programmatic interface to services, many of which were only available previously as browser-oriented services. Three consolidating standards are of interest to us:

- SOAP - *Simple Object Access Protocol*
- WSDL - *Web Services Description Language*
- UDDI - *Universal Description Discovery and Integration*

In short, these represent the *wiring*, *description* and *discovery* stacks respectively, of web services. UDDI in particular, is still under-developed. However, numerous manually built and managed directories detail a growing number of services from which a user can choose, to wrap a Shadowboard sub-agent around. For example see the directories at: www.xmethods.com and www.salcentral.com.

To give the reader a flavour of what is available, the following is a short, terse description of a small sample of the web services listed there: *eBay Price Watcher; FedEx Package Tracker; BabelFish language translation; Stock Quote; Traffic Conditions; Barnes and Noble Price Quote; Amazon; Currency Exchange Rate;*

Whois by IP addresses, domain; Location by IP (longitude, latitude); Top Music Charts; etc).

The format of data that are retrievable from these by SOAP RPC for example, are lists of values which match a fixed set of attributes - very much like the ground term tuples that a sub-agent stores in the Shadowboard computational system. So far we have built into the ShadowFaces interface the ability to define a wrapper around a known SOAP RPC Web Service. We intend to extend the wrapping to other protocols including WSDL described services in the near future.

By wrapping web services individually within sub-agents we are able to orchestrate numerous independent web services into synergies of capability via the CoLoG constraint logic programming enacted at the EoC level of Shadowboard.

6 Conclusions

In recent times, since we proposed the Shadowboard architecture as a blueprint for the resulting technology [4], several agent architectures and methodologies have come forward with hierarchies of agent roles as a feature [8]. Several people have since questioned our use and continued emphasis of the Psychology of Subselves as the theory behind the Shadowboard hierarchy of sub-agents. To paraphrase the question: *'Why highlight the psychology when the ends is simply a hierarchical role structure?'*

The answer is: *the journey is easily as important as the destination* – particularly when insights are still being gained. Numerous insights have been derived from the psychology:

- *Computational implications* from the Psychology of Subselves include:
  - Notions of decomposition, of sub-agents without full-autonomy, with an overriding executive controller with full self-knowledge of the system – the *Aware Ego* agent.
  - Notions of diversity in both the sub-agents and in their capabilities – allowing lesser applications, such as web services to be wrapped as sub-agents, without diluting the theory.
  - Internalizing external services and capabilities, for pragmatic purposes.
  - Learning from external agencies as archetypes and mentors.
- *Structural implications* from the Psychology of Subselves include:
  - Hierarchy of sub-agency equating to a hierarchy of roles.
  - Application of numerous ontologies at sub-levels of a model.
  - Storage of ground terms at the sub-agent level, is akin to the memory storage/retrieval of war-stories associated with a sub-self/role in a personal history.

The psychology continues to guide the developing *Shadowboard methodology* outlined in Section 4 above. Now that the development is well advanced and the working prototype is being enhanced, new questions are coming to the fore with

respect to fertile ground for future research. We would like to address the following research questions in the near future:

- Will the use of the software validate theories of the Analytical Psychology?
- What are the consequences of externalizing aspects of Self?
- How will people be influenced and what can they learn about the real *Self*, from this model of Self with augmented functionality?

It is now obvious to us that the cross-fertilization between the fields of Computer Science and Psychology are only at the beginning.

References

[1] Assagioli, Roberto: *Psychosynthesis*. New York. Viking, (1965)

[2] Brooks, R. A.: Intelligence Without Representation, Workshop in Foundations of Artificial Intelligence, Endicott House, Dedham, MA, (1987)

[3] Buhler, P.A. and Vidal, J.M.: Toward the synthesis of web services and agent behaviors. Proceedings of the International Workshop on Challenges in Open Agent Systems, (2002)

[4] Goschnick, S.B.: Shadowboard**:** an Agent Architecture for enabling a sophisticated Digital Self, Thesis, 180 pages, Dept. of Computer Science, University of Melbourne, Sep. (2001)

[5] Goschnick S.B.: Shadowboard: A Whole-Agent Architecture that draws Abstractions from Analytical Psychology, Proceedings of PRIMA 2000, Melbourne (2000)

[6] Goschnick, S.B. & Sterling, L.: Enacting and Interacting with an Agent-based Digital Self in a 24x7 Web Services World. To appear in the Workshop on Humans and Multi-Agent Systems, at the AAMAS-2003 conference, (2003)

[7] Hristozova, M.: Middle Agents. Presentation, Agent Lab, Department of Computer Science, University of Melbourne (2002)

[8] Juan,T., Pearce,A.R. and Sterling, L.: ROADMAP: extending the gaia methodology for complex open systems. AAMAS (2002) 3-10

[9] Jung C.G.: *Man and his Symbols*. Aldus Books, (1964)

[10] Kuno, H. and Sahai, A.: My agent wants to talk to your service: personalizing web services through agents. In Proceedings of the First International Workshop on Challenges in Open Agent Systems, (2002)

[11] Lewis M.: Designing for Human-Agent Interaction**.** AI Magazine, Summer (1998) 67-78

[12] Maes P. and Kozierok R.: Learning Interface Agents. In *Proceedings of AAAI Conference*, (1993)

[13] Marriott, K. and Stuckey, P.J.: Programming with Constraints: an Introduction, MIT Press, (1998)

[14] Rao A.S. and Georgeff M.P.: An Abstract Architecture for Rational Agents. Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning, (1992) 439-449

[15] Sliker, G.: Multiple Mind - Healing the Split in Psyche and World. Shambhala Publications Inc. (1992)

[16] Shneiderman, B.: Designing the User Interface, Strategies for Effective Human-Computer Interaction, third edition, Addison-Wesley, (1997)

[17] Stone, H. and Winkelman, S.: Embracing Ourselves - the Voice Dialogue Manual. New World Library. (1989)

[18] World Wide Web Consortium Document Object Model (DOM). Documentation available at <http://www.w3.org/DOM/> (2003)

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