The LIDA architecture: Adding new modes of learning to an intelligent, autonomous, software agent

Stan Franklin
The University of Memphis

Integrated Design and Process Technology, IDPT-2006
Printed in the United States of America, June, 2006

THE LIDA ARCHITECTURE: ADDING NEW MODES OF LEARNING TO AN INTELLIGENT, AUTONOMOUS, SOFTWARE AGENT

Stan Franklin
W. Harry Feinstone Interdisciplinary Research Professor
& Professor of Computer Science
Institute for Intelligent Systems • FedEx Institute of Technology #312
The University of Memphis • Memphis, TN, 38152 • USA
http://www.cs.memphis.edu/~franklin
☎ (901) 678-1341 • (901) 678-5129 🖷
franklin@memphis.edu

And

F. G. Patterson, Jr.[1]
Visiting Scholar & Adjunct Professor of Computer Science
Institute for Intelligent Systems • FedEx Institute of Technology
The University of Memphis • Memphis, TN, 38152 • USA
☎ (703) 764-1004 • (775) 599-4319 🖷
pat.patterson@nasa.gov

ABSTRACT

This is a report on the LIDA architecture, a work in progress that is based on IDA, an intelligent, autonomous, "conscious" software agent that does personnel work for the US Navy. IDA uses locally developed cutting edge artificial intelligence technology designed to model human cognition. IDA's task is to find jobs for sailors whose current assignments are about to end. She selects jobs to offer a sailor, taking into account the Navy's policies, the job's needs, the sailor's preferences, and her own deliberation about feasible dates. Then she negotiates with the sailor, in English via iterative emails, about job selection. We use the word "conscious" in the sense of Baars' Global Workspace Theory (Baars, 1988, 1997), upon which our architecture is based.

IDA loops through a cognitive cycle in which she perceives the environments, internal and external; creates meaning, by interpreting the environment and deciding what is important; and answers the only question there is: "What do I do next?" LIDA, the learning IDA will add three modes of learning to IDA's design: perceptual learning, episodic learning, and procedural learning. LIDA will learn from experience, which may yield several lessons over several cognitive cycles. Such lessons include newly perceived objects and their relationship to already known objects and categories, relationships among objects and between objects and actions, effects of actions on sensation, and improved perception of sensory data. The LIDA architecture incorporates six major artificial intelligence software technologies: the copycat architecture, sparse distributed memory, pandemonium theory, the schema mechanism, the behavior net model, and the subsumption architecture.

NOMENCLATURE

Attention Codelet: A codelet that attempts to train attention on some particular kind of information. Examples: Expectation codelet, intention codelet (Conscious Software Research Group, 2006).

Autonomous agent: A system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future (Franklin & Graesser, 1996).

Behavior Codelet: A codelet that can execute some par-

[1] Dr. Patterson is on assignment from the National Aeronautics and Space Administration Headquarters, Office of the Chief Engineer, Washington, DC, to the University of Memphis, FedEx Institute of Technology, Institute for Intelligent Systems.

ticular, specialized task (Conscious Software Research Group, 2006).

Codelet: In the LIDA model, a special purpose, active process, the function of which is represented in a few lines of executable code.

Consciousness: Conscious cognition is implemented computationally by way of a broadcast of contents from a global workspace, which receives input from the senses and from memory (Baars 2002; Franklin, Baars, Ramamurthy, & Ventura, 2005; Franklin, 2003).

Content-addressable Memory: A memory the contents of which can be found by cuing with part of the desired content without knowing a memory address. Examples: Episodic memory in us humans, Declarative memory, Transient Episodic memory (Conscious Software Research Group, 2006).

Episodic Memory: Associative, content-addressable, memory for events, the what, the where, the when. Example: Where food was cached (Conscious Software Research Group, 2006).

Expectation Codelet: An attention codelet that watches for the result of some behavior(s) (Conscious Software Research Group, 2006).

Global Workspace theory: A theory developed by B. J. Baars, that associates conscious experience with three basic constructs: a global workspace, a set of specialized unconscious processors, and a set of unconscious contexts that serve to select, evoke, and define conscious contents (Baars, 1988).

IDA: **I**ntelligent **D**istribution **A**gent, a "conscious" software agent, based on Global Workspace theory.

Information Codelets: Codelets that carry information and participate in coalitions that compete for attention (Conscious Software Research Group, 2006).

Intention Codelet: An attention codelet that watches for information relevant to achieving some particular goal (Conscious Software Research Group, 2006).

LIDA: **L**earning **IDA**, adds three modes of learning to the IDA model: perceptual, episodic, and procedural learning.

Long-term Working Memory: Memory buffer than holds percepts and local associations for the competition of attention (Conscious Software Research Group, 2006).

Perception: The process of assigning meaning to incoming sensory data. Examples: The color red, a sound (Conscious Software Research Group, 2006).

Perceptual Memory: A perceptual memory, distinct from semantic memory but storing some of the same contents, exists in humans, and plays a central role in the assigning of interpretations to incoming stimuli. The conscious broadcast begins and updates the process of learning to recognize and to categorize, both employing perceptual memory (Conscious Software Research Group, 2006).

Procedural Memory: The storage of procedures for executing behaviors. Examples: Running, eating, mating (Conscious Software Research Group, 2006).

Sensation: In the LIDA model, the detection of primitive features in the sensory input from the environment. The capability to detect primitive features is built-in.

Transient Episodic Memory: Episodic memory with decay times measured in hours or a day or so (Conscious Software Research Group, 2006).

Working Memory Buffers: Buffers whose contents serve to cue transient episodic and declarative memory (Conscious Software Research Group, 2006).

IDA, AN INTRODUCTION TO LIDA

IDA is an intelligent, autonomous, "conscious" (Baars, 1988, 1997; Franklin, 2003) software agent that does personnel work for the U. S. Navy, using artificial intelligence software technology developed at the University of Memphis. The LIDA architecture is an extension of the IDA model. IDA research was funded by Office of Naval Research and other Navy sources.

There were two primary goals for the IDA project. The first goal, a science goal, was to create a "working" model of human cognition that could be used to suggest possible answers to questions about the human mind, and to articulate such questions in greater detail. Mechanisms of mind could be conjectured and analyzed in terms of working IDA mechanisms. The second goal, an engineering goal, was to design and develop a practical application that could do the work of a human "detailer," a person who negotiates with sailors who are near the end of their current tours of duty, in everyday English about new jobs. Acceptance testing of IDA amounted to concurrence by human detailers that IDA did her job in "about the same way" as did the detailers themselves, with reasonable, if not identical, practical outcomes for the sailors.

In the course of finding new jobs for sailors, IDA senses her environments, internal and external; creates meaning by interpreting the environment and deciding what is important; and answers the only question there is: "What do I do next?" In the detailer domain, she communicates with sailors in English using common email to understand their needs. She deliberates about jobs to offer to each sailor, taking into account the sailor's preferences, the Navy's current and projected personnel needs, estimated cost, optimal timing, detailed logistical issues, and the like. She selects jobs to offer the sailor and negotiates with the sailor over the course of several emails. The IDA architecture is shown in Figure 1.

THE LIDA COGNITIVE CYCLE

LIDA, the *learning* IDA, will add three modes of learning to IDA: perceptual learning, episodic learning, and procedural learning. Each of these types of learning will be discussed in turn as part of the LIDA cognitive cycle, which is depicted in Figure 2. The model enables a fine grained analysis of a broad range of aspects of human cognition, organized as a cycle of processes, beginning with perception, and ending in an action. There is neurobiological evidence that complete human cognitive cycles occur at an average rate of five cycles per second (D'Mello & Franklin, 2004).

The cognitive cycle is produced by the collective, coordinated actions of subsystems, that we shall describe, in turn, as Sensation, Perception, Working Memory, Episodic memory, Consciousness, Learning, and Action Selection. We have organized the actions of the cycle into three phases: LIDA samples and acts on her environment through a *perceive*-*interpret*-*act* cycle. It is noteworthy that these phases bear a striking similarity to a number of systems engineering life cycles. For example, the *formulate-analyze-interpret* cycle of Sage (1992), the *recognize-analyze-synthesize* cycle of Patterson (1999), the *plan-do-check-act* cycle of Shewhart and Deming (Sage, 1992), the *assign-perform-input-evaluate-improve* cycle of Øxnevad (Patterson, 2001), and the Boehm spiral (1990) may each be overlaid onto the cognitive cycle (with many interesting implications for the nature of systems engineering that will not be discussed here).

The planned LIDA computational model of the Baars Global Workspace (Baars, 1988, 1997; Franklin, 2005) model will generate hypothesis at the *mechanism* level. These hypotheses are typically in the areas of functional designs and their underlying requirements; independence, interdependence, and communication among modules that correspond to counterparts in neurobiology; performance issues related to timing, capacity, complexity, and the like; and a host of others – all of which are now or, given proper advances in neurophysiology, soon should be fully testable.

LIDA will learn from experience, which may yield several lessons over several cognitive cycles. For example, she can be expected to learn to recognize newly perceived objects and their relationship to already known objects and object categories, relationships among objects and between objects and actions, effects of actions on sensation, and improvements in the perception of sensory data.

The *Perceive Phase*

The sensory subsystems, whether we are referring to a living organism, a robot, or a thermostat, provide the sensory input to the *Perceive Phase* of the cognitive cycle. From a human anatomical standpoint, one may think of the photoreceptor cells[2] of the retina for vision, the chemoreceptors located in the olfactory epithelium for our sense of smell, and the sensory receptors of our other sensory systems as creating inputs from the *external* environment. We can also speak of an *internal* environment that consists of primarily of proprioception. Imagination also contributes to the internal environment. Using the stimuli from the external environment and the contents of the internal environment, the Perceive Phase is the part of the cognitive cycle that must recognize, or establish ignorance or unfamiliarity of, that which is sensed.

We describe the agent's ontology, which we can define as the list of objects and classes that LIDA can currently recognize, as organized into a slipnet, a semantic network with passing activation (Franklin, 1995; Hofstadter, 1995; Mitchell, 1993). The slipnet is, conceptually, a large-and-growing collection of codelets that can recognize a stimulus and pass activation to its predecessor in the network (Bolland, 1997). Slipnet nodes and perception codelets can be thought of as interchangeable implementation details. As new stimuli enter from the environment, codelets from the frontier of the network, known as primitive feature detectors, descend upon and evaluate the stimulus (D'Mello et al., 2004). At the front end of each cognitive cycle, those zero or more perceptual codelets that find some part of the stimulus relevant and increase their own activations, begin passing activation to those nodes in the slipnet to which they are linked, and those nodes pass activation in turn to other nodes to which they are related (linked). When activation is complete (stable), the set of slipnet nodes that have received sufficient activation, representing a subset of the total ontology, are referred to as the *percept* which is copied to preconscious working memory buffers, as shown in Figure 2.

The percept can be thought of as the set of elements of the ontology that are relevant to the stimulus. We can organize this information into a binary vector, where each field of one or more bits represents an element of the ontology. We refer to the binary vector as a *cue*, that will be used to query the content-addressable memories, *autobiographical memory* (ABM) and *transient episodic memory* (TEM), which are based closely on Kanerva's sparse distributed memory (SDM) (D'Mello, Ramamurthy, & Franklin, 2005; Franklin, 1995; Kanerva, 1990, 1993). The TEM contains information about events that the agent "remembers," including time and place information. The

[2]It has been shown that the differences between rods and cones have little or no functional significance. The functional aspects of vision are determined by the electro-physiological properties of a single functional type of *photoreceptor cell* that is associated with one of four types of chromophore. These chromophores are sensitive in the ultraviolet, the short, the medium and the long wavelength portions of the visual spectrum of light. (Fulton, J. T. 2000. *Process in Animal Vision*. Corona Del Mar, CA: Vision Concepts. Internet: URL: http://www.4colorvision.com., 2000)

ABM is populated with information that has been learned over time, including some information that begin in TEM, but which is now known independently of the when and where. As implemented with SDM, every item in ABM is capable of relating to any other object or object category in the agent's ontology. A cue is thought of as an address in the ABM. SDMs do not implement every memory address, instead implementing discrete points that are scattered throughout the high-dimensional memory space.

Cues from the preconscious buffers are used to query SDM, which is content addressable. The values of the points in the neighborhood of the point in ABM and TEM that are actually implemented in SDM are averaged and returned to become the new values of the cues. The perceptual codelets query the SDMs again using the new cue, and repeat the process until the value returned becomes stable, *i.e.*, unchanged from the previous iteration.

The *Interpret* Phase

The percept that was created and stabilized in the *Perceive Phase* is now copied by *information codelets* to Long Term Working Memory[3] (LTWM), where it joins previously copied percepts. LTWM may be likened to Jackson's pandemonium theory (Dennett, 1991; Franklin, 1995; Jackson, 1987) and may be compared to other blackboard architectures (Engelmore & Morgan, 1988; Franklin, 1995; Hayes-Roth, 1985). Attention codelets analyze the contents of LTWM, each attention codelet looking for elements of the various percepts present that are of interest to it. *Coalitions* of codelets containing an attention codelet and information codelets of interest to it are formed when possible. When the attention codelets have completed their analysis, the coalition with the highest average activation, which is thus deemed to be important, is chosen to move its information into *consciousness* (Figure 2) to be broadcast throughout the LIDA system. Information that moves through consciousness in this way is *learned.* The function of consciousness is to broadcast throughout all parts of the LIDA agent the important information from the winning coalition. In particular, the information that composes the winning coalition of codelets is broadcast to each of the learning modules for assimilation and storage.

The *Act* Phase

The *Act Phase* begins when the conscious broadcast is received by each subsystem of the LIDA agent. Our focus here is on the Procedural Memory (PM) and the Action Selection subsystem (AS), which, like the other subsystems in LIDA, are acting independently.

[3]Information in Long Term Working Memory may have a lifetime of only several seconds.

PM is an actively self-managed collection of schemes. The software technology upon which the module is based is Drescher's Scheme Net (Drescher, 1991; Franklin, 1995). A scheme can be compared to a production rule. Each scheme has a context; actions, which are chains of behaviors; and results, along with a base-level activation that estimates the likelihood of the actions producing the result when performed in the context. When the broadcasts of consciousness provide information that matches the context of one or more schemes, PM will suggest these schemes by copying instantiations of the schemes to AS.

It is the job of AS to decide what to do next. To do that, AS will look across its set of possible next behaviors (instantiated schemes). This set is composed of the next behavior to be executed in each of the schemes that it currently has in its buffers. By examining the latest conscious broadcast, updating precondition checklists, and looking at the updated total activations of coalitions of codelets that have passed through the current or a previous conscious broadcast, AS will pick this cognitive cycle's winner. LIDA then performs the behavior. Maes' behavior net (Franklin, 1995; Maes, 1990, 1991; Negatu and Franklin. 2002) is the underlying mechanism of AS.

As part of some of the behaviors residing in PM and copied to AS, an assertion may be present to be used to validate the efficacy of the action should the behavior be chosen and implemented by AS. Assessment of such a behavior using these assertions, which are referred to as *expectations* in the LIDA model, is the job of *expectation codelets*.

A COGNITIVE THEORY OF EVERYTHING

To borrow a phrase from Minsky, "The mind is simply what the brain does" (1987, 2002). As neurobiologists become better able to explain the physical processes in the brain, we will likely become able to correlate them to mental processes and products. We believe that our development of a model of cognition can help to guide such discoveries by developing processes and exploring them in terms of our cognitive model. Any hypotheses that derive from our work on LIDA should be completely testable, given the advances in neurobiology that we expect.

The LIDA cognitive cycle is a very useful tool for "explaining," or conjecturing on the basis of the mechanics of our model, how various cognitive functions might work in humans and other living organisms. The understanding that we gain by using a classical systems approach[4] is much different from the biological (or com-

[4]A particularly clear description of the systems approach, demonstrating the precedence of synthesis over analysis, is given by Ackoff (Ackoff, R. L. 1981. *Creating*

puter science) approach of decomposing the system, learning about the parts, then summarizing the knowledge. Here, with respect to the function (that is, the subsystem identified with the process responsible for the function or exhibited behavior of the system) that we want to explain, we first identify the larger system that contains the function. Our systems approach seeks to explain the behavior of the larger system in terms of the functioning of the subsystem within the larger system. The behavior of the whole directs our investigation of the parts. If our model does not exactly reproduce the parts of the organism that we are trying to understand, it does give clues about what to look for, about how such a system might work, and about the limitations upon such mechanisms within the system. Of course, if our model does accurately model the organism, then so much the better. LIDA is designed to represent the three phases, *perceive-interpret-act* using separate subsystems for each phase. Within these subsystems the functions that we will address are sensation, perception, consciousness, learning, the scheme mechanism, and the behavior net.

Sensation

Sensation refers to the front end of the Perceive Subsystem. The role of sensation is to identify objects from the environment, both internal and external. Every autonomous agent must possess built-in sensors. Perceptual codelets, representing nodes from the Perceptual Associative Memory (PAM), modeled after the slipnet architecture, recognize parts of the sensory input and increase their own activation according to strength of recognition. Activation is passed within the slipnet until slipnet stabilizes. We refer to the collection of nodes whose activation is above threshold as the percept. The percept provides a link between sensation and perception.

Perception

The Perception subsystem is also contained in the Perceive Subsystem. During the Perceive Phase, the Perceive Subsystem is able to identify objects, object categories, and relationships among object. To accomplish this, the Perception subsystem uses PAM. The percept is reformed into a binary vector referred to as a cue, which is used to query TEM and ABM. The result is a binary vector populated with averages of the associations in declarative memory. The returned vector is used as the next cue with which to query declarative memory. The process repeats until it stabilizes.

---

the Corporate Future. New York: John Wiley & Sons, Inc.1981)

Feeling and Emotion

Feelings and emotions are represented in the same way as other data in the system. They implement the value system of the agent -- whether actions do or do not achieve a desired result in the current situation, whether elements of the current environment are dangerous or foretell a desirable outcome, and the like -- according to the past experience of the agent. Drives may be built into the agent as positive emotional reactions to desired outcomes. Feelings and emotions also have an active role in modulating all types of learning. In general, the more affect, the more effective the learning, but only up to a point, as the performance curve resembles an inverted "U."

Consciousness

In the Interpret phase of the LIDA model, the information from the Perceive Phase, carried by information codelets, are copied to long term working memory. Of the percepts held in information codelets in long term working memory, attention codelets assess their importance to the particular attention codelet, often forming coalitions with several information codelets with related information. The activation of a coalition is the average of the activations of its constituents. The coalition with highest activation is chosen and broadcast throughout the cognitive model.

The conscious broadcast has two primary roles: recruitment of resources and learning. Recruitment of resources here involves the scheme mechanism and its selection of schemes for the behavior net. Learning in the LIDA model can only occur after information has been broadcast by consciousness.

The types of learning in the model are *perceptual*, *episodic*, and *procedural*. Perceptual learning is learning to recognize new objects, new categorizations, and new relationships (Conscious Software Research Group, 2006). As new objects, categories, and the relationships among them and between them and other elements of the agent's ontology are learned, nodes (objects and categories) and links (relationships) are added to PAM, but not before the conscious broadcast (Figure 2). The conscious broadcast begins and updates the process of learning to recognize and to categorize, both employing perceptual memory (Franklin et al., 2005).

Episodic learning is the encoding of information into episodic memory, the associative, content-addressable, memory for events -- the what, the where, and the when (Conscious Software Research Group, 2006). Procedural learning is the encoding of procedures for executing behaviors into Procedural Memory (Figure 2). It is the learning of new actions and action sequences with which to accomplish new tasks (D'Mello et al., 2004). Learning to perform new motor behaviors, or to improve the performance of existing behaviors (Conscious Software Research Group, 2006). These procedural skills are shaped by reinforcement learning, operating by way of conscious processes over more than one cognitive cycle (Franklin et al., 2005).

Scheme Mechanism

Schemes have a context, an action, and a result, as well as a base-level activation. The action may be complex sequences of steps organized with Boolean logic. Any schemes whose contexts are met are instantiated into the behavior net. Before being copied, variables are bound according to the conscious broadcast and the scheme receives a current activation.

Behavior Net

The behavior net will choose the next behavior from one of the instantiated schemes to which it currently has access, based on the conscious broadcast. LIDA's behavior net is closely modeled after Maes' behavior net software technology (Franklin, 1995; Maes, 1990, 1991; Negatu and Franklin. 2002). The behaviors from which to choose are those contained in the schemes that have been instantiated by the scheme net. The behaviors, of which schemes are composed, are typically simple actions that can be regarded as pre-existing features of the motor system. A behavior can be thought of as a procedure together with preconditions and postconditions and having an activation. If a behavior's preconditions are all true, then the behavior is executable.

Action selection marks the end of the cognitive cycle by enacting from among the executable behaviors whose activations are above a certain threshold the behavior with the greatest such activation. In the IDA model, the computation of activation involved possible built-in goals, called drives, that provided activation to certain types of behaviors. In the LIDA model, drives and goals are not built-in, although primitive feelings are built-in that serve the same purposes, especially regarding the well-being of the agent. The network is constructed by linking the possible behaviors in such a way that, if a node can make one or more preconditions of other nodes true, then there is a *successor* link to those other nodes. Both predecessor and successor links are established. There are also conflictor links between each node and other nodes that it can make some of its preconditions false. In each cognitive cycle, activation is spread by each behavior to its successors and from each behavior to its predecessors until the stability condition is met.

SOME POSSIBLE APPLICATIONS OF LIDA

Of the many possible applications of the LIDA technology, there are five areas that stand out. In each case, LIDA will control an autonomous agent that is performing a human-like task, but one that, for various reasons, is better accomplished by an autonomous machine.

Autonomous Space Exploration

Space exploration combines all the negative features of a hostile environment, poor communications with an external controller, routine tasks, and a general lack of many of the necessities to sustain human life. For many space missions, it seems reasonable to deploy an autonomous agent, especially for deep space missions that make remote communication, and thus remote control, difficult or impossible. On the positive side, the lessons learned by agents during prior missions can in theory be completely transferred to new agents for new missions, something very difficult for human astronauts to accomplish.

Autonomous Undersea Exploration

Undersea exploration is significantly friendlier than deep space for both human based and remote controlled machine based missions. The possibility of sequences of dives, each of short duration, enables humans to return for needed resources as required. Shipboard control of robotic tools is also possible, since distances are relatively short for communication, and agents can be returned to the surface for repair when needed. The motivation for the use of autonomous agents, therefore, seems to be issues of safety, duration, and exploration of the interior parts of undersea structures that would limit communication, such as caves, volcanoes, and sunken ships. An example of a safety issue would be exploration at extreme depths or examination of hazardous materials. An example of duration would be continuous inspection of pipes in offshore oil processing or continuous monitoring of an undersea volcano. Finally, exploration of an underwater cave, for example, would likely prohibit communication and control from the surface.

Autonomous Product Inspection (Including Environments Humans Cannot Survive)

This list of inspection environments is very large, ranging from the routine assembly line inspection of manufactured products to difficult medical testing that is carried out where no human based medical care is available, to inspection of dangerous environments, such as nuclear power plants or war zones. The combination of tedious inspections with a high cost of error is a prescription for automation. The additional need for autonomous control is found in the unpredictability of the environments, requiring planning and execution of strategies for success over multiple cognitive cycles.

Complex Planning and Scheduling (Program and Project Management)

Much of the research into management techniques in the last fifty years has been in the area of process control, usually resulting in the use of data intensive metrics and measures, followed by complicated analysis and perhaps simulation of possible remedies and outcomes, finishing with resource constrained multipart actions, the results of which must be verified, perhaps leading to a new cycle of measure, strategize, and act. While management is possibly a necessarily human activity, many, perhaps most, of the methods and tools of management are subject to automation. In addition, as much of management involves unpredictable situations, an *autonomous* agent with built-in drives and goals is needed, since there are no rules, but only guidelines, for managing the unpredictable.

Human Information Agents

IDA's success as a detailer for the Navy is well documented (Franklin, Kelemen, & McCauley, 1998). Leveraging this experience to the commercial sector opens any number of possibilities. For example, the model lends itself well to applications such as customer service agents, travel agents, and loan officers in banks (Franklin, 2001). In addition, for much more tedious, computationally intensive searches, the LIDA technology can likely outperform a human counterpart. Examples might be fingerprint analysis, where thousands of samples must be searched; photoreconnaissance, where very large numbers of photographs containing unpredictable objects must be interpreted, and many other forms of data mining. It is worth noting that in finding work for LIDA, we have moved from tasks which are lonely and dangerous to tasks on which she can simply outperform a human counterpart.

CONCLUSIONS

The LIDA architecture is the result of the on-going research of the Conscious Software Research Group at the University of Memphis. The architecture is based on the successes of previous models: the copycat model, sparse distributed memory, pandemonium theory (Dennett, 1991; Franklin, 1995; Jackson, 1987), and other blackboard-like models, the scheme mechanism, and the behavior net, together with the IDA model, developed here. Not only is the model useful for designing control systems for autonomous agents, but LIDA is also useful for generating testable conjectures about how the human mind works, using a systems approach to cognition, including a cognitive cycle and a system of systems.

ACKNOWLEDGMENTS

We wish gratefully to acknowledge the work of the members of the Conscious Software Research Group at the Institute for Intelligent Systems, FedEx Institute of Technology at the University of Memphis. The current members include Dr. Bernard Baars, Scott Brown, Dr. Glenn Colman, Sidney D'Mello, David Friedlander, Kevin Marker, Dr. Aregahegn Negatu, Dr. Uma Ramamurthy, and Yongmin Shan, together with the authors of this paper.

REFERENCES

Ackoff, R. L. 1981. *Creating the Corporate Future*. New York: John Wiley & Sons, Inc.

Baars, B. J. 1988. *A cognitive theory of consciousness*. Cambridge: Cambridge University Press.

Baars, B. J. 1997. *In the Theater of Consciousness: The Workspace of the Mind*. New York: Oxford University Press.

Boehm, B. W. 1990. A Spiral Model of Software Development and Enhancement. In M. Dorfman, & R. H. Thayer (Eds.), *System and Software Requirements Engineering*: 513-527. Los Alamitos, California, USA: IEEE Computer Society Press.

Bolland, S. 1997. Copycat Tutorial, available on URL:http://www2.psy.uq.edu.au/CogPsych/Copycat/Tutorial/index.htm. Brisbane: University of Queensland.

Conscious Software Research Group. 2006. How Minds Work: A Cognitive Theory of Everything. In S. Franklin (Ed.),*http://csrg.cs.memphis.edu/tutorial/* Vol. 2006. Memphis, TN: FedEx Institute of Technology, University of Memphis.

D'Mello, S. K., & Franklin, S. P. 2006. A cognitive architecture capable of human like learning. *In Preparation.*.

D'Mello, S. K., Ramamurthy, U., & Franklin, S. 2005. *Encoding and Retrieval Efficiency of Episodic Data in a Modified Sparse Distributed Memory System*. Paper presented at the 27th Annual Meeting of the Cognitive Science, Stresa, Italy.

Dennett, D. C. 1991. *Consciousness Explained*. Boston: Little, Brown, and Company.

Drescher, G. 1991. *Made Up Minds: A Constructivist Approach to Artificial Intelligence*. Cambridge, MA: MIT Press.

Engelmore, R., & Morgan, T. (Eds.). 1988. *Blackboard Systems*. New York: Addison-Wesley.

Franklin, S., Baars, B. J., Ramamurthy, U., & Ventura, M. 2005. The Role of Consciousness in Memory. *Brains, Minds and Media*, 1(bmm150 (urn:nbn:de:0009-3-1505)).

Franklin, S. Automating Human Information Agents. In: Z. Chen & L. C. Jain (Ed.). Practical Applications of

Intelligent Agents. Berlin: Springer-Verlag, 2001. Automating Human Information Agents

Franklin, S. 2003. "IDA: A Conscious Artifact?" *Journal of Consciousness Studies* 10: 47-66.

Franklin, S. P. 1995. *Artificial Minds*. Cambridge, Massachusetts: MIT Press.

Franklin, S. P. 2005. A “Consciousness”-Based Architecture for a Functioning Mind. In D. N. Davis (Ed.), *Visions of mind: architectures for cognition and affect*. Hershey, PA: Information Science Publishing (an imprint of Idea Group Inc.).

Franklin, S. P., & Graesser, A. C. 1996. Is it an Agent, or just a Program? Taxonomy for Autonomous Agents, *Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages*. Vienna, Austria: Springer-Verlag.

Franklin, S. P., Kelemen, A., & McCauley, L. 1998. IDA: A Cognitive Agent Architecture. In F. DiCesare, & M. A. Jafari (Eds.), *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics: Intelligent Systems for Humans in a Cyberworld*. Los Alamitos, CA: IEEE Computer Society Press.

Fulton, J. T. 2000. *Process in Animal Vision*. Corona Del Mar, CA: Vision Concepts. Available on the Internet: URL: http://www.4colorvision.com.

Hayes-Roth, B. 1985. A Blackboard Architecture for Control. *Artificial Intelligence*, 26 (3): 251–321.

Hofstadter, D. 1995. *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought*. New York: Basic Books.

Jackson, J. V. 1987. Idea for a Mind. *Siggart Newsletter*, 181: 23–26.

Kanerva, P. 1990. *Sparse Distributed Memory*. Cambridge, MA: MIT Press.

Kanerva, P. 1993. Chapter 3. Sparse Distributed Memory and Related Models. In M. H. Hassoun (Ed.), *Associative Neural Memories: Theory and Implementation*: 50-76. New York: Oxford University Press.

Maes, P. 1990. How to Do the Right Thing. *Connection Science Journal*, 1(3, Special Issue on Hybrid Systems): 291-323.

Maes, P. 1991. *A Bottom-up Mechanism for Behavior Selection in an Artificial Creature*. Paper presented at the First International Conference on Simulation of Adaptive Behavior.

Minsky, M. 1987. *The Society of Mind*. New York: Simon & Schuster.

Minsky, M. 2002. Minds Are Simply What Brains Do, *Truth* (on-line newsletter), Vol. 2006: Interview with Marvin Minsky. Addington, TX: Leadership University.

Mitchell, M. 1993. *Analogy-Making as Perception*. Cambridge: The MIT Press.

Negatu, A., and Franklin, S. 2002. An action selection mechanism for "conscious" software agents. *Cognitive Science* 2:363-386.

Patterson, F. G., Jr. 1999. Chapter 1: Systems Engineering Life Cycles: Life Cycles for Research, Development, Test, and Evaluation; Acquisition; and Planning and Marketing. In A. P. Sage, & W. B. Rouse (Eds.), *Handbook of Systems Engineering and Management*: 59-112. New York: John Wiley & Sons.

Patterson, F. G., Jr. 2001. An Engineering Life Cycle Based On Early Unification of Design Representations. In N. Callaos (Ed.), *Proceedings of the Fifth World Multiconference on Systemics, Cybernetics, and Informatics (SCI 2001)*. Orlando: International Institute of Informatics and Systemics (IIIS).

Sage, A. P. 1992. *Systems Engineering*. New York: John Wiley & Sons, Inc.

FIGURES

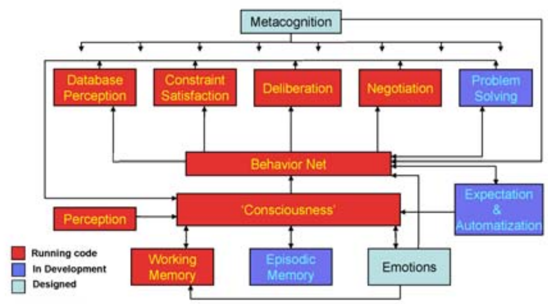

Figure 1. The IDA Architecture

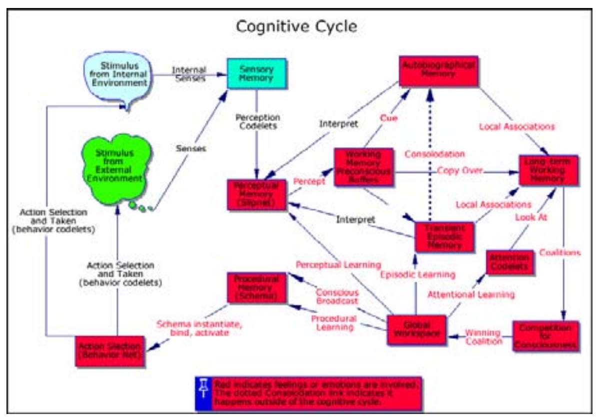

Figure 2. The LIDA Cognitive Cycle