2010

Grounded Event-Based and Modal Representations for Objects, Relations, Beliefs, Etc

R. McCall

S. Franklin

D. Friedlander

Grounded Event-Based and Modal Representations for Objects, Relations, Beliefs, Etc.

Ryan McCall[1], Stan Franklin[2], David Friedlander[3]

[1,2]Computer Science Department & Institute for Intelligent Systems, The University of Memphis
[1]FedEx Institute of Technology #403h, 365 Innovation Dr., Memphis, TN 38152
[2]FedEx Institute of Technology #312, 365 Innovation Dr., Memphis, TN 38152
[3]1111 Post Oak Blvd., Apt. 1510, Houston, TX 77056
[1]rmccall@memphis.edu, [2]franklin@memphis.edu, [3]dfriedlander@att.net

Abstract

Intelligent software agents (agents) adhering to the action selection paradigm have only one primary task that they need accomplish at any given time: to choose their next action. Consequently, modeling the current situation effectively is a critical task for any agent. With an accurate model of the current situation, actions can be better selected. We propose an *event-based* representational framework designed to provide grounded perceptual representations of events for agents. We describe how they are produced and detail their role in a comprehensive cognitive architecture designed to explain, integrate, and model human cognition. Event-based representations draw inspiration from research on thematic roles, and integrate research on event perception. Events are represented as parameterized actions, that is, nodes with thematic role links that can bind to Agent, Object, and other node types.

Introduction

Agents adhering to the action selection paradigm, (Franklin, 1995) have only one primary task they need to accomplish at any given time; that is, selecting their next action. In order to choose actions well, it is critical for the agent to effectively represent its current situation. In this paper we present basic, primitive representations for agents to represent their current situation with. We also detail the processes necessary to produce these representations and the role they play in a comprehensive cognitive architecture using the LIDA model as an example (Franklin & Patterson 2006, Franklin et al. 2007).

The LIDA model is a comprehensive, conceptual and computational architecture designed to explain, integrate, and model a large portion of human cognition. Based primarily on Global Workspace Theory (Baars 1988), the model implements and fleshes out a number of psychological and neuropsychological theories including situated cognition (Varela et al. 1991), perceptual symbol systems (Barsalou 1999, 2008), working memory (Baddeley and Hitch 1974), memory by affordances (Glenberg 1997), long-term working memory (Ericsson and Kintsch 1995), Sloman's H-CogAff architecture (1999), and transient episodic memory (Conway 2001).

The LIDA computational architecture, derived from the LIDA cognitive model, employs several modules that are designed using computational mechanisms drawn from the "new AI." These include variants of the Copycat Architecture (Hofstadter and Mitchell 1995, Marshall 2002), Sparse Distributed Memory (Kanerva 1988, Rao and Fuentes 1998), the Schema Mechanism (Drescher 1991, Chaput et al. 2003), the Behavior Net (Maes 1989, Tyrrell 1994), and the Subsumption Architecture (Brooks 1991). An initial version of a software framework for LIDA-based agents has recently been completed. The LIDA model and its ensuing architecture are grounded in the LIDA cognitive cycle.

Every autonomous agent (Franklin and Graesser 1997), be it human, animal, or artificial, must frequently sample (sense) its environment and select an appropriate response (action). More sophisticated agents, such as humans, process (make sense of) the input from such sampling in order to facilitate their action selection. The agent's "life" can be viewed as consisting of a continual sequence of these cognitive cycles, as they are called in the LIDA model. Each cycle is composed of phases of sensing, attending and acting. A cognitive cycle can be thought of as a moment of cognition - a cognitive "moment." Higher-level cognitive processes are composed of many of these cognitive cycles, each a cognitive "atom."

Just as atoms are composed of protons, neutrons and electrons, and some of these are composed of quarks, gluons, etc., these cognitive "atoms" have a rich inner structure. What the LIDA model hypothesizes as the rich inner structure of the LIDA cognitive cycle will be described briefly. More detailed descriptions are available elsewhere (Baars & Franklin 2003, Franklin et al. 2007).

During each cognitive cycle the LIDA agent first makes sense of its current situation as best as it can *by updating its representation of its world, both external and internal.*

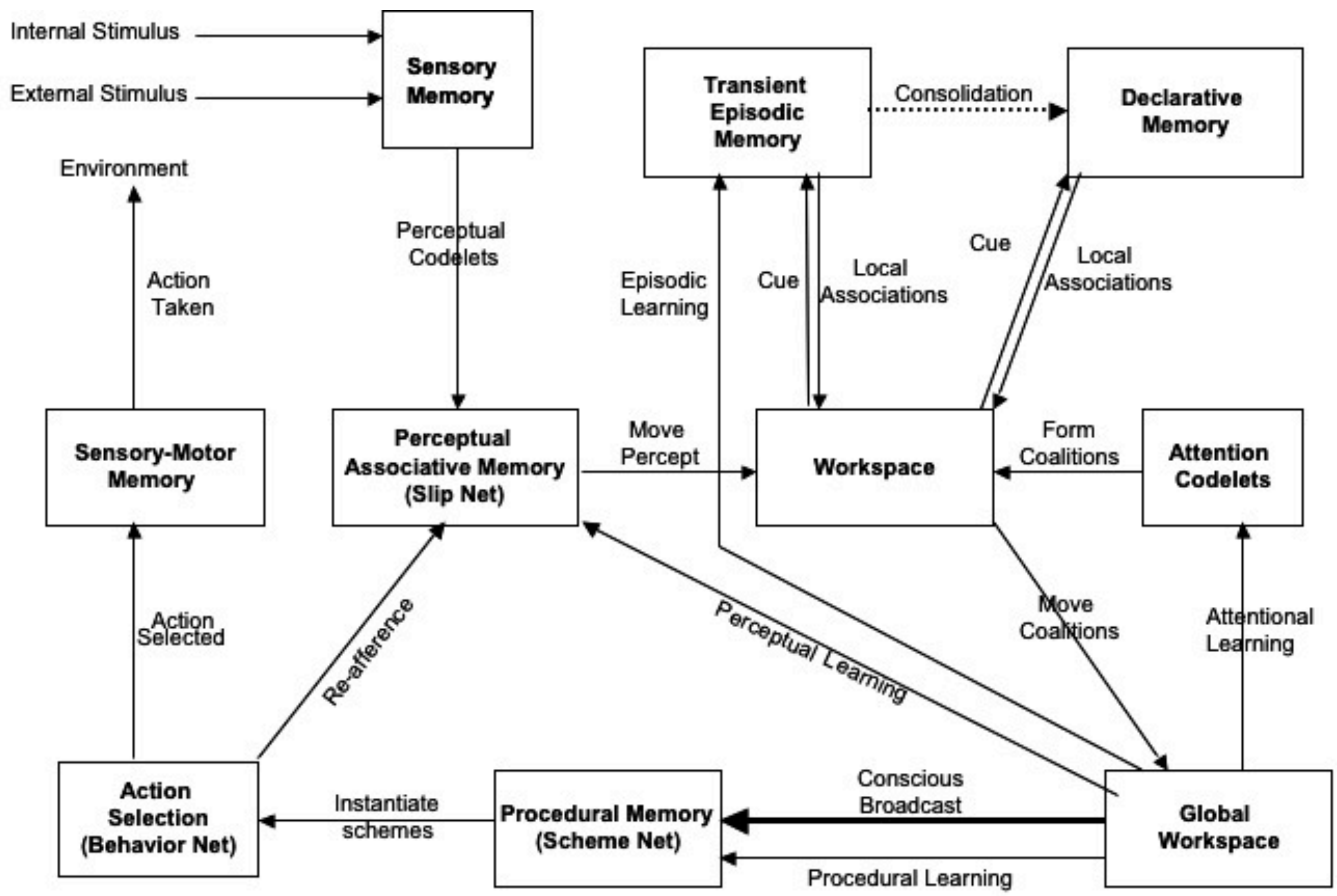

Figure 1. The LIDA Cognitive Cycle

By a competitive process, as specified by Global Workspace Theory, it then decides what portion of the represented situation is most in need of attention. Broadcasting this portion, the current contents of consciousness, helps the agent to finally choose an appropriate action and execute it. Thus, the LIDA cognitive cycle can be subdivided into three phases, the understanding phase, the attention phase, and the action selection phase. Figure 1 should help the reader follow the description. It proceeds clockwise from the upper left .

Beginning the understanding phase, incoming stimuli activate low-level feature detectors in Sensory Memory. This preprocessed output is sent to Perceptual Associative Memory where higher-level feature detectors feed in to more abstract entities such as objects, categories, actions, events, feelings, etc. The resulting percept is sent to the Workspace where it cues both Transient Episodic Memory and Declarative Memory producing local associations. These local associations are combined with the percept to generate a current situational model, the agent's understanding of what's going on right now.

Attention Codelets[1] begin the attention phase by forming coalitions of selected portions of the current situational model and moving them to the Global Workspace. A competition in the Global Workspace then selects the most salient, the most relevant, the most important, the most urgent coalition, whose contents become the content of consciousness that are broadcast globally.

The action selection phase of LIDA's cognitive cycle is also a learning phase in which several processes operate in parallel. New entities and associations, and the reinforcement of old ones, occur as the conscious broadcast reaches Perceptual Associative Memory. Events from the conscious broadcast are encoded as new memories in Transient Episodic Memory. Possible action schemes, together with their contexts and expected results, are learned into Procedural Memory from the conscious broadcast. Older schemes are reinforced. In parallel with all this learning, and using the conscious contents, possible action schemes are recruited from Procedural Memory. A copy of each such scheme is instantiated with its variables bound, and sent to Action Selection, where it competes to be the behavior selected for this cognitive cycle. The selected behavior triggers Sensory-Motor Memory to produce a suitable algorithm for the execution of the behavior. Its execution completes the cognitive cycle.

While its developers hesitate to claim that LIDA is more

[1] The term codelet refers generally to any small, special purpose processor or running piece of computer code.

general or more powerful than other comprehensive cognitive architectures such as SOAR (Laird, et al., 1987), ACT-R (Anderson, 1990), Clarion (Sun, 2007), etc., they do believe that LIDA will prove to be a more detailed and faithful model of human cognition, including several forms of learning, that incorporates the processes and mechanisms required for sophisticated decision making.

LIDA has a number of features that separate it from other cognitive architectures. There is an explicit attention mechanism (functional consciousness) to focus on a salient portion of its current situation. Feelings and emotions are used for motivation and to bias learning. LIDA incorporates the "cognitive cycle" hypothesis – that the action-perception cycle (Neisser, 1976; Freeman, 2002) can be thought of as a cognitive atom, and that all higher-level cognitive processes are composed of multiple cognitive cycles implemented using behavior streams. LIDA's Workspace provides a detailed inner structure for preconscious working memory. It includes a model of the agent's current situation (Current Situational Model). The Current Situational Model contains a perceptual scene with windows for both real and virtual (imaginary) conceptual representation (McCall, Snaider, Franklin, 2010). The Current Situational Model also contains complex structures for an even higher-level, "global" representation. The Workspace also contains an ordered queue of the recent contents of consciousness (Conscious Contents Queue) and an episodic buffer of recent local associations (Franklin et al., 2005).

A quick glance at the LIDA model, particularly its cognitive cycle diagram (Figure 1), makes the model appear modular. This interpretation is misleading. Within a single cycle, individual modules such as Perceptual Associative Memory and Action Selection are internally quite interactive. Structure-building codelets operate interactively on the Workspace. The cognitive cycle as a whole operates quite interactively, in that internal procedures happen asynchronously. For example, nodes and links in Perceptual Associative Memory instantiate grounded copies of themselves in the Workspace whenever they become sufficiently activated, without waiting for a single percept to be moved. Thus, the cognitive cycle is more interactive than would be expected by a system based on information passing between modules. The only straight information passing in the entire cycle is the conscious broadcast of a single coalition's contents and the selection of the single action to be performed. All other processes within a cycle are interactive. All higher-level, multi-cyclic processes in LIDA are quite interactive, since their cognitive cycles, occurring at a rate of five to ten per second, continually interact with one another. Though LIDA superficially appears modular, it is, in its operation, much more aligned with the interactive approach.

Background

Earlier versions of LIDA's representations consisted of only nodes and links (Franklin 2005). However, their use is impractical to represent the kind of processed information produced by complex perception. For example, representing a detailed visual image with nodes and links does not appear to be the most effective representation from which to make geometric inferences.

In many important Cognitive Architectures (e.g. SOAR, ACT-R, Clarion), symbolic representations are used (Langley 2009). While symbolic-like representations are, in our view (Franklin, 1995) necessary for cognition, they must be grounded in external reality. A major shortcoming of early artificial intelligence was its ignoring of the perceptual processes necessary for real-world cognition. Granted, understanding perception and implementing it computationally is a challenging task. This criticism of symbolic representations extends to users of pre-processed domains, e.g. virtual worlds or simulations in which agents can perceive object "for free". In the "real" world, agents must *construct* meaning from their environment. While such simulations are useful, it is imperative that agents developed for such simulations do not rely on preprocessed stimuli. The danger here is that the agent's performance will not scale to more complex or "real-world" problems.

One important kind of perceptual representation is what we have termed *event-based representations*. In this scheme events are represented as parameterized actions (Allbeck and Badler, 2003). We essentially take our definition of events from Zacks & Tversky (2001): an event is the contents of a segment of time at a given location, and that segment is perceived to have a beginning and an end. We are primarily interested in events lasting on the order of a few seconds, such as throwing a ball.

Next we describe primitive feature detectors, Perceptual Associative Memory nodes, and their role in producing perceptual representations that are ultimately grounded in the sensory stimulus. These representations retain their sensory modality groundings as they are used in cognitive processing. This ensures that our representations are modal and grounded (Barsalou 1999, 2008) and follow an embodied approach to cognition (Barsalou 1999, 2008; de Vega, Glenberg and Graesser, 2008; Varela et. al., 1991). Then we propose classes of representational nodes thought to be most primitive (fundamental). We focus in detail on a particular class, action nodes, that, with their thematic role links, are used to produce event representations. Next we propose primitive link classes analogous to the primitive node classes. Finally, we identify future directions for our research in this area.

Sensory Memory and Perceptual Associative Memory Components

Primitive feature detectors

In LIDA, all feature detectors are a part of Perceptual Associative Memory (PAM). Each feature detector consists of a PAM node and an algorithm to compute the current activation of the node. Each also has a receptive field comprised of lower level feature detectors or some subset of the agent's sensory memory. The activation depends on the activity of the entities in its receptive field, for example, a center-surround cell in human vision. Unless a primitive feature detector (PFD) is detecting a trivial pattern in the sensory stimulus there must be some processing or algorithm to compute the PFD node's activation depending on the activity within its receptive field. Regardless of mechanism used for detection, the end result of the feature detection process is the sending of activation to a PAM node or nodes. Such a processing step effectively translates raw sensory input into a higher-level node representation. It is here that the first "map" is being "drawn" of the current environmental "territory".

Nodes

Each node in PAM has base-level, a current, and a total activation. Base-level activation measures the long-term importance or usefulness of the node, and can be modified as part of perceptual learning. The current activation of a node is a measure of that node's relevance in the current situation. Current activation for a non-primitive feature detector node *A* is some function of the activation of other nodes with links that enter into *A*. Links have weights, which may factor into this computation of current activation. Current activation decays quickly for the nodes near PFD nodes and more slowly for nodes farther away from (and more abstract than) PFD nodes. Total activation is simply the sum of base-level and current activations.

In LIDA, when a PAM node goes over its threshold, it is said to be instantiated. Instantiation involves creating a copy of the original node and binding its variables. Instantiated PAM nodes ought to maintain connection to the lower level sensory representations that have activated them. If this is not done, them the node becomes an ungrounded symbol. Since every PAM node ultimately gets its activation from PFDs, all perceptual representations in LIDA are sufficiently grounded in LIDA's sensory machinery.

Proposed primitive node classes. Nodes and links from LIDA's Perceptual Associative Memory are a major kind of representation for the LIDA architecture. In particular, they play a central role in the building of structures in the preconscious Workspace. They should be thought of in terms of Barsalou's (1999) perceptual symbols. There is further discussion of additional kinds of representation thought to be critical to the success of a generally intelligent system (Wang, Goertzel, and Franklin, 2008). Major node classes thought to be primitive include *action nodes* that represent actions, *feature nodes* representing perceptual features such as red*, and class nodes* representing categories. Instance and feature nodes are linked to class nodes providing activation. *Instance (object) nodes* represent instances of a class. Attachment of instance nodes to class nodes is via links that represent membership. Finally, *feeling nodes*, implement values.

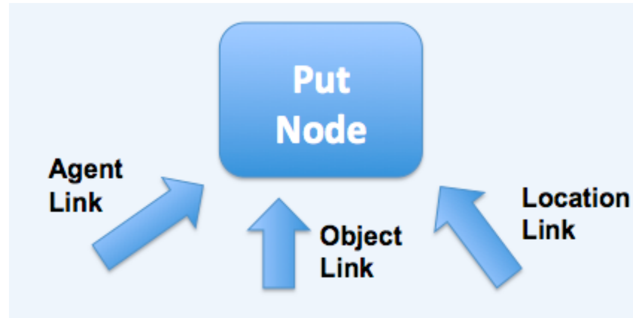

Figure 2. The action node *Put* has *Agent*, *Object*, and *Location* role links.

Action nodes (Figure 2) are central to event representation. Every action nodes has thematic role links, first introduced by Fillmore (1968). We make no claim that any particular classification of thematic roles is the best; rather we use Fillmore's as an example. The binding of thematic role links of action nodes to role filler nodes takes place in the preconscious Workspace, possibly requiring multiple cycles. If the event had been learned into PAM, i.e., if an action with specific role-fillers had been learned as a single unit, then the event could be recognized directly in one cognitive cycle. We take an action node together with bound role nodes to constitute the representation of an event (Figure 3). In spoken language there are different senses of the same action e.g. strike, as in "to cause harm" or "to impact". For these cases there must be a unique node for each sense of the action. Also note that there are endless combinations of role-fillers that could be bound to a given action depending on the environmental situation. By adding a representation for events to the LIDA model, events can come to attention (consciousness), be learned into LIDA's various memories and, once learned, be recognized directly in a single cognitive cycle. We extend the notion of action to include states of being. Thus the node for "exists" is considered to be an action node with roles links just like other action nodes. The *exists* node would be used in combination with an object to represent the presence of this object in the current situation. Structure-building codelets, which are daemon-like processes, match active nodes in the Workspace to unfulfilled roles.

Instance nodes represent instances of a class. They are attached to the class node via links that represent membership. Instances of a class are features in the sense that they have a role in defining the class. Non-instance nodes that are attached to the class node by links are called feature nodes. The activation of a feature node contributes to the activation of the class node(s) to which that feature node is linked. The instance nodes plus the feature nodes that are connected to a class node are what define the class.

Proposed primitive link classes. Each link has a source node and a sink node. There are separate links for passing activation in each direction.

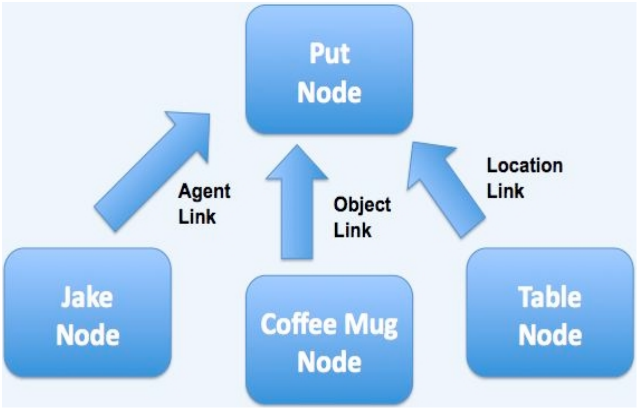

Figure 3. The event representation for the event "Jake put the coffee mug on the table"

Each link class includes its own weighting scheme for passing activation between the source node and sink node. Nodes and links may be added via perceptual learning. Links are meant to represent relationships between nodes. So the classes of links thought to be primitive include: *is-a-feature-of*, *is-a-part-of*, *spatial*, *causal*, *thematic role*, *membership,* and *kind-of*. By adding these link classes we distinguish between different kinds of relationships that occur in representation: thematic roles in events, spatial relationships for geometry, relationships of membership, etc. Two other primitive link classes have functional significance: *feeling* links associate nodes with feeling nodes, while *lateral-inhibitory* links allow one node to inhibit another. Causal links concern events on the order of 10 to 100 ms apart in time. Kind-of links represents a subset relationship e.g. polar bear –kind-of-> bear.

Spatial relationships are represented at a higher conceptual level by spatial links. At a low level, spatial relations may be represented by a geometric map of objects similar to how a political map reflects the actual shape and position of countries. A conceptual representation of a spatial relationship, e.g. *left-of,* represents all possible such relations. Thus a *left-of* link may be activated whenever such a relationship occurs. Depending on the circumstances, the relationship may be represented by one unidirectional spatial link between two objects or, alternatively, by two unidirectional links connecting the objects in both ways. Unidirectional links provide additional information about the relationship of the two nodes they connect. For example the direction of the link in *bob –left-of-> bill* says that bob is on the left and bill is on the right. Other examples of spatial links include *right-of*, *in-front-of*, *in-back-of*, *above*, *below*, and *touching*.

Common thematic roles thought to be primitive for representation in LIDA are *Agent, Attribute, Beneficiary, Source, Destination, Location, Experiencer, and Instrument* (Verbnet, 2009, Sowa, 1991). In LIDA, the 'object' role link is considered to be equivalent to an affordance link, that is, a link that represents an affordance of an object (Gibson, 1979).

Implementation and Testing. A software framework for LIDA-based agents has recently been developed. We plan to use it to replicate data from existing psychological experiments (e.g. reaction time, backward masking, etc.) Such data will be used to test and tune internal parameters of the model.

We plan to test performance by replicating previous experiments in event segmentation, for example, segmenting the movement of a ball into events (Zacks 2004).

Conclusion

We have described how grounded perceptual representations can be produced within a comprehensive cognitive architecture. A number of primitive node and link classes were proposed as fundamental representational building blocks. A particular class of nodes, action nodes, feature members with thematic role links. Action nodes with bound role links comprise the representation of an event. A major direction for future research involves addressing issues arising from event representation where multiple events at possibly different time scales are present in the agent's model of the current situation.

Acknowledgements

The authors would like to thank Javier Snaider and Sidney D'Mello for feedback and suggestions on some of the ideas presented in this paper.

References

Allbeck, J. & Badler, N. (2003). Representing and Parameterizing Behaviors, in Life-Like Characters: Tools, Affective Functions and Applications, H. Prendinger and M. Ishizuka, Eds.: Springer.

Baars, B. (1988). *A Cognitive Theory of Consciousness*. Cambridge: Cambridge University Press.

Baars, B, & Franklin, S. (2003). How conscious experience and working memory interact. Trends in Cognitive Sciences Vol. 7 No. 4.

Baddeley, A. D., & Hitch, G. J. (1974). Working memory. In G. A. Bower (Ed.), The Psychology of Learning and Motivation (pp. 47–89). New York: Academic Press.

Barsalou, L. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences, 22*, 577–609.

Barsalou, L. (2008). Grounded Cognition. *Annual Review of Psychology, 59*, 617–645.

Brooks, R.A. Intelligence without Representation. Artificial intelligence, 1991. Elsevier.

Chaput, H. H., Kuipers, B., & Miikkulainen, R. (2003). Constructivist Learning: A Neural Implementation of the Schema Mechanism. Paper presented at the Proceedings of WSOM '03: Workshop for Self-

Organizing Maps, Kitakyushu, Japan.
Conway, M. A. (2001). Sensory-perceptual episodic memory and its context: autobiographical memory. Phil Trans. R. Soc. Lond. B 356, 1375-1384.
de Vega, Glenberg and Graesser, (2008). Symbols and Embodiment: Debates on meaning and cognition. Oxford, UK: Oxford University Press.
Drescher, G.L. (1991). Made-up minds: A Constructivist Approach to Artificial Intelligence.
Ericsson, K. A., and W. Kintsch. 1995. Long-term working memory. Psychological Review 102:21–245.
Fillmore, C. (1968). The case for case. In Universals in linguistic theory, eds. Bach & Harms:1–90. New York: Holt, Rinehart & Wilson.
Franklin, S. (1995). *Artificial Minds*. Cambridge MA: MIT Press.
Franklin, S. (2005, March 21-23, 2005). *Perceptual Memory and Learning: Recognizing, Categorizing, and Relating*. Paper presented at the Symposium on Developmental Robotics: American Association for Artificial Intelligence (AAAI), Stanford University, Palo Alto CA, USA.
Franklin, S., Baars, B. J., Ramamurthy, U., & Ventura, M. (2005). The Role of Consciousness in Memory. *Brains, Minds and Media, 1*, 1–38, pdf.
Franklin, S., & Graesser, A., 1997. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, published as Intelligent Agents III, Springer-Verlag, 1997, 21-35.
Franklin, S., & Patterson, F. G. J. (2006). The LIDA Architecture: Adding New Modes of Learning to an Intelligent, Autonomous, Software Agent *IDPT-2006 Proceedings (Integrated Design and Process Technology)*: Society for Design and Process Science.
Franklin, S., Ramamurthy, U., D'Mello, S., McCauley, L., Negatu, A., Silva R., & Datla, V. (2007). LIDA: A computational model of global workspace theory and developmental learning. In AAAI Fall Symposium on AI and Consciousness: Theoretical Foundations and Current Approaches. Arlington, VA: AAAI.
Freeman, W. J. (2002). The limbic action-perception cycle controlling goal-directed animal behavior. *Neural Networks, 3*, 2249-2254.
Gibson, J.J. (1977). The theory of affordances. In R. Shaw & J. Bransford (eds.), Perceiving, Acting and Knowing. Hillsdale, NJ: Erlbaum.
Glenberg, A. M. 1997. What memory is for. Behavioral and Brain Sciences 20:1–19.
Hofstadter, D. (1995). Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought. New York: Basic Books.
Kanerva, P. (1988). Sparse Distributed Memory. Cambridge MA: The MIT Press.
Langley, P., Laird, J.E., Rogers, S. (2009), Cognitive architectures: Research issues and challenges, Cognitive Systems Research 10, 141-160.
Maes, P. 1989. How to do the right thing. Connection Science 1:291-323.
Marshall, J. (2002). Metacat: A self-watching cognitive architecture for analogy-making. In W. Gray & C. Schunn (eds.), Proceedings of the 24th Annual Conference of the Cognitive Science Society, pp. 631-636. Mahwah, NJ: Lawrence Erlbaum Associates.
McCall, R., Snaider, J., & Franklin, S., (2010 submitted). Sensory and Perceptual Scene Representation. Journal of Cognitive Systems Research.
Neisser, U. (1976). *Cognition and Reality: Principles and Implications of Cognitive Psychology* San Francisco: W. H. Freeman.
Palmer, M. Verbnet. Retrieved Nov. 6, 2009, from http://verbs.colorado.edu/~mpalmer/projects/verbnet.html
Rao, R. P. N., & Fuentes, O. (1998). Hierarchical Learning of Navigational Behaviors in an Autonomous Robot using a Predictive Sparse Distributed Memory. Machine Learning, 31, 87-113.
Sloman, A. 1999. What Sort of Architecture is Required for a Human-like Agent? In Foundations of Rational Agency, ed. M. Wooldridge, and A. Rao. Dordrecht, Netherlands: Kluwer Academic Publishers.
Sowa, John F., ed. (1991) Principles of Semantic Networks: Explorations in the Representation of Knowledge, Morgan Kaufmann Publishers, San Mateo, CA.
Tyrrell, T. (1994). An Evaluation of Maes's Bottom-Up Mechanism for Behavior Selection. Adaptive Behavior, 2, 307-348.
Varela, F. J., Thompson, E., & Rosch, E. (1991). *The Embodied Mind*. Cambridge, MA: MIT Press.
Zacks, J. M., & Tversky, B. (2001). Event structure in perception and conception. Psychological Bulletin, 127, 3-21.
Wang, P., Goertzel, B., & Franklin, S. (2008). Artificial General Intelligence 2008: Proceedings of the First AGI Conference. IOS Press: Frontiers in Artificial Intelligence and Applications.
Zacks, J. M. (2004). Using Movement to Understand Simple Events. Cognitive Science, 28, 979–1008.