An Agent Architecture Potentially Capable of Robust Autonomy

Stan Franklin
and the Conscious Software Research Group

Institute for Intelligent Systems
University of Memphis
Memphis, TN, 38152, USA
stan.franklin@memphis.edu

Abstract

Robust autonomy on the part of software agents requires, at least in part, the ability to deal intelligently with novel and unexpected situations. According to global workspace theory, dealing with such situations is one of the primary functions of consciousness in humans. Below we briefly describe two software agents that implement this psychological theory, and discuss their resulting potential for robust autonomy.

Autonomous Agents

Artificial intelligence pursues the twin goals of understanding human intelligence and of producing intelligent artifacts. Designing, implementing and experimenting with autonomous agents (Franklin & Graesser 1997) furthers both these goals in a synergistic way. In particular, designing and implementing within the constraints of a theory of cognition can further the first goal by providing conceptual and computational models of that theory. The second goal is also served. One way to get really smart agents is to model them after humans. "Really smart" autonomous agents should be robust in their ability to deal intelligently with novel and unexpected situations.

"Conscious Software Agents"

A "conscious" software agent is one that implements global workspace theory (Baars 1988, 1997), a psychological theory of consciousness and cognition. (No claim of sentience is being made.) "Conscious" software agents have the potential to play a synergistic role in both modeling cognitive theory and in producing software with more human-like intelligence.

Minds can be viewed as control structures for autonomous agents (Franklin 1995). A theory of mind constrains the design of a cognitive agent that implements and models that theory. While a theory is typically abstract and only broadly sketches a functional architecture, an implemented computational design provides a fully articulated architecture and a complete set of mechanisms. This architecture and set of mechanisms provides a richer, more concrete and more decisive theory, as well as both a conceptual and a computational model of the theory.

Moreover, every design decision taken during an implementation translates into a hypothesis about how human minds work. These hypotheses may motivate experiments with humans and other forms of empirical tests. Conversely, the knowledge gained from such experiments should motivate corresponding modifications of the architecture and mechanisms of the cognitive agent. In this way, the concepts and methodologies of cognitive science and of computer science will work synergistically to enhance our understanding of mechanisms of mind (Franklin 1997).

As we'll see below, such "conscious" software agents should be capable of more adaptive, more human-like behavior, including being capable of robust operation in the face of novel and unexpected situations.

Global Workspace Theory

Global workspace theory postulates that human cognition is implemented by a multitude of relatively small, special purpose processes, almost always unconscious. (It's a multiagent system.) Communication between them is rare and over a narrow bandwidth. Coalitions of such processes find their way into a global workspace. This limited capacity workspace serves to broadcast the message of the coalition to all the unconscious processors (bringing it to consciousness) in order to recruit relevant processors to join in handling the current novel situation, or in solving the current problem. Thus consciousness produces robustness, that is, it allows us to deal with novelty or problematic situations that can't be dealt with efficiently, or at all, by habituated unconscious processes. In particular, consciousness serves to recruit appropriately useful resources, thereby solving the relevance problem. Implementing a "consciousness" mechanism in software agents can be expected to enable more robust, more human-like software. .

CMattie

"Conscious" Mattie (CMattie) is our first attempt at implementing a "conscious" software agent (McCauley & Franklin 1998, Ramamurthy et al. 1998, Zhang et al. 1998, Bogner et al. 2000) in this case a clerical agent. She composes and emails out weekly seminar announcements, having communicated by email with seminar organizers and announcement recipients in natural language. She maintains her mailing list, reminds organizers who are late with their information, and warns of space and time conflicts. There is no human involvement other than these email messages.

CMattie's cognitive modules include perception, learning, action selection, associative memory, "consciousness," emotion and metacognition. Her emotions influence her action selection. Her mechanisms include variants and/or extensions of Maes' behavior nets (1989), Hofstadter and

Mitchell's Copycat architecture (1994), Jackson's pandemonium theory (1987), Kanerva's sparse distributed memory (1988) , and Holland's classifier systems (1986). As of this writing CMattie is almost completely coded and should be ready for experimentation within a few months. Though CMattie's domain is narrow and relatively simple, novel and unexpected situations seem likely to arise. Will her current architecture and mechanisms, described below, be capable of robust action as the theory leads us to expect? It's an open question.

IDA

IDA (Intelligent Distribution Agent) is a "conscious" software agent being developed for the US Navy (in parallel with CMattie) (Franklin et al. 1998). At the end of each sailor's tour of duty, he or she is assigned to a new billet. This assignment process is called distribution. The Navy employs some 280 people, called detailers, to effect these new assignments. IDA's task is to facilitate this process by completely automating the role of detailer.

IDA must communicate with sailors via email in natural language, understanding the content and producing life-like responses. Sometimes she will initiate conversations. She must access several databases, again understanding the content. She must see that the Navy's needs are satisfied by adhering to some ninety policies. She must hold down moving costs. And, she must cater to the needs and desires of the sailor as well as is possible. This includes negotiating with the sailor via an email correspondence in natural language. Finally, she must write the orders and start them on the way to the sailor.

Though more complex, IDA's architecture and mechanisms are largely modeled after those of CMattie. In particular IDA needs deliberative reasoning in the service of action selection (Sloman 1999), where CMattie was able to do without. At this writing the design of IDA is far along, so that she constitutes a useful conceptual model of cognition capable of producing testable hypotheses (Bogner et al. In preparation). The coding of an initial partial implementation is now running. IDA's domain is orders of magnitude more complex than that of CMattie, and should produce a variety of novel and unexpected situations that would require robust handling. Again the theory says that IDA should be capable of such robust autonomy.

Codelets

In both the CMattie and IDA architectures the processors postulated by global workspace theory are implemented by codelets, small pieces of code. These are specialized for some simple task and often play the role of demons waiting for appropriate conditions under which to act.

"Consciousness"

The apparatus for "consciousness" consists of a coalition manager, a spotlight controller, a broadcast manager, and a collection of attention codelets who recognize novel or problematic situations (Bogner 1999, Bogner et al. 2000). Each attention codelet keeps a watchful eye out for some particular situation to occur that might call for "conscious" intervention. In most cases the attention codelet is watching the workspace, which will likely contain both perceptual information and data created internally, the products of "thoughts." Upon encountering such a situation, the appropriate attention codelet will be associated with the small number of codelets that carry the information describing the situation. This association should lead to the collection of this small number of codelets, together with the attention codelet that collected them, becoming a coalition. Codelets also have activations. The attention codelet increases its activation in order that the coalition, if one is formed, might compete for "consciousness".

If the situation is sufficiently novel, there may be no attention codelet that will respond to it. This indicates that robustness may well require general-purpose attention codelets whose task is to respond to unknown situations. But how is such a codelet to recognize such situations without knowing about all the usual occurrences? It seems to be the same problem faced by the immune system, and may well require the same kind of solution.

In CMattie and IDA the coalition manager is responsible for forming and tracking coalitions of codelets. Such coalitions are initiated on the basis of the mutual associations between the member codelets. At any given time, one of these coalitions finds it way to "consciousness," chosen by the spotlight controller, who picks the coalition with the highest average activation among its member codelets. Global workspace theory calls for the contents of "consciousness" to be broadcast to each of the codelets. The broadcast manager accomplishes this.

Perception

Perception in both CMattie and IDA consists mostly of understanding incoming email messages in natural language. In sufficiently narrow domains, natural language understanding may be achieved via an analysis of surface features without the use of a traditional symbolic parser. Allen describes this approach as complex, template-based matching, natural language processing (1995). CMattie's limited domain requires her to deal with only a dozen or so distinct message types, each with relatively predictable content. This allows for surface level natural language processing. CMattie's language understanding module has been implemented as a Copycat-like architecture (Hofstad-

ter & Mitchell 1994) though her understanding takes place differently. The mechanism includes a slipnet storing domain knowledge, and a pool of codelets (processors) specialized for specific jobs, along with templates for building and verifying understanding. Together they constitute an integrated sensing system for CMattie, allowing her to recognize, categorize and understand. IDA, though more complex, perceives in much the same way.

Action Selection

Both CMattie and IDA depend on a behavior net (Maes 1989) for high-level action selection in the service of built-in drives. Each has several distinct drives operating in parallel. These drives vary in urgency as time passes and the environment changes. Behaviors are typically mid-level actions, many depending on several codelets for their execution. A behavior net is composed of behaviors and their various links. A behavior looks very much like a production rule, having preconditions as well as additions and deletions. A behavior is distinguished from a production rule by the presence of an activation. Each behavior occupies a node in a digraph. The three types of links, successor, predecessor and conflictor, of the digraph are completely determined by the behaviors.

As in connectionist models, this digraph spreads activation. The activation comes from activation stored in the behaviors themselves, from the environment, from drives, and from internal states. The more relevant a behavior is to the current situation, the more activation it's going to receive from the environment. Each drive awards activation to every behavior that, by being active, will satisfy that drive. Certain internal states of the agent can also send activation to the behavior net. This activation, for example, might come from a coalition of codelets responding to a "conscious" broadcast. Finally, activation spreads from behavior to behavior along both excitatory and inhibitory links. Call a behavior *executable* if all of its preconditions are satisfied. To be acted upon a behavior must be executable, must have activation over threshold, and must have the highest such activation. Behavior nets produce flexible, tunable action selection for these agents.

Action selection via behavior net suffices for CMattie due to her relatively constrained domain. IDA's domain is much more complex, and requires deliberation in the sense of creating possible scenarios, partial plans of actions, and choosing between them. For example, suppose IDA is considering a sailor and several possible jobs, all seemingly suitable. She must construct a temporal scenario for each of these possible billets. In each scenario the sailor leaves his or her current position during a certain time interval, spends a specified length of time on leave, possibly reports to a training facility on a certain date, uses travel time, and arrives at the new billet with in a given time frame. Such scenarios are valued on how well they fit the temporal constraints and on moving and training costs. These scenarios are composed of scenes organized around events, and are constructed in a computational workspace corresponding to working memory in humans.

Deliberation, as in humans, is mediated by the "consciousness" mechanism. The principle is that IDA should use "consciousness" whenever a human detailer would be conscious in the same situation. For example, IDA could readily recover all the needed items from a sailor's personnel record unconsciously with a single behavior stream. But, observing and questioning human detailers indicate that they become conscious of each item individually. Hence, according to our principle, so must IDA be "conscious" of each retrieved personnel data item.

Other Modules

Both CMattie and IDA employ sparse distributed memory (SDM) as their major associative memories (Kanerva 1988). SDM is a content addressable memory that, in many ways, is an ideal computational mechanism for use as a long-term associative memory. Any item written to the workspace triggers a read from associative memory returning prior activity associated with the current entry.

In both CMattie and IDA we include mechanisms for emotions (McCauley & Franklin 1998). CMattie, for example may "experience" such emotions as guilt at not getting an announcement out on time, frustration at not understanding a message, and anxiety at not knowing the speaker and title of an impending seminar. Action selection will be influenced by emotions via their effect on drives, modeling recent work on human action selection (Damasio 1994). IDA's emotions are similar, but more complex.

IDA, but not CMattie, is provided with a constraint satisfaction module designed around a linear functional. It provides a numerical measure of the suitability, or fitness, of a specific job for a given sailor. This fitness measure is use in the deliberation process described above.

Due to her quite narrow domain, CMattie generates language (email messages) simply by filling in appropriate scripts. IDA does the same, except that she chooses the appropriate script "consciously," and occasionally has to tweak it to fit the current situation.

Metacognition should include knowledge of one's own cognitive processes, and the ability to actively monitor and consciously regulate them. This would require self-monitoring, self-evaluation, and self-regulation. CMattie's metacognition module (Zhang et al. 1998) uses Holland's classifier system (1975). It serves to interrupt oscillatory behavior, to keep the agent on task, and to push her toward efficient allocation of resources.

Evaluation

Both CMattie and IDA will be evaluated on how well they perform their designated tasks. CMattie will be judged as would a human secretary responsible for seminar announcements. Does she maintain her mailing list well? Are

the announcements compete, accurate, and on time? Does she catch inconsistencies and afford organizers an opportunity to correct them? She has already been evaluated as an implementation of global workspace theory (Franklin & Graesser 1999). Evaluating IDA will prove more difficult since the US Navy has no established protocol for evaluating human detailers. Our eventual evaluation of IDA is still in the planning stage.

Future Plans

Modules capable of learning from conversations with organizers and detailers are planned (Ramamurthy et al. 1998, Negatu & Franklin 1999) for both CMattie and IDA. A development/training period is also anticipated for IDA (Franklin 2000). A paper is in preparation detailing some of the hypotheses for human cognition suggested by these agents (Bogner et al. In preparation). Future agents built on the IDA architecture are being considered with a self and the ability to report "conscious" activity.

Dealing with Novel and Unexpected Situations

Though their "consciousness" modules are designed to deal intelligently with novel, unexpected, and problematic situations, both CMattie and IDA are normally expected to deal only with novel instances of routine situations. Though its content may be different, one speaker topic message from a seminar organizer is much like another in form, even in natural language with no agreed upon protocol. Similarly, finding a new billet for one sailor will generally require much the same process as for another even thought the personnel data and job descriptions and requirements are different. Even the negotiation process between IDA and a sailor promises to be relatively routine. From analysis of a corpus of messages we've constructed a complex, but quite finite, flow chart of possible messages types and responses.

However, we expect IDA to occasionally to receive messages outside of this expected group. Can she handle such a message intelligently by virtue of her "consciousness" mechanism alone? I doubt it. Some attention codelet will be needed to bring the novel message to "consciousness." Some behavior priming codelets will be needed to instantiate an appropriate behavior stream (goal hierarchy) needed to deal with the situation (Franklin to appear). Perhaps a single, novel-situation attention codelet will bee needed to respond to a percept by default if no other attention codelet does so within a prescribed time interval. This novel-situation attention codelet would try to bring information about the novel situation to "consciousness." The broadcast would, hopefully, recruit behavior priming codelets to instantiate a behavior stream able to cope with the situation. Suppose there is no such stream? Well, we humans can't cope with every situation either. But, we try. And, we combine goal hierarchies in novel ways. This combining ability would seem a necessary ingredient if a "conscious" software agent were to be truly robustly autonomous. It also seems that learning must play a role here.

I conclude that "conscious" software agents present a promising architecture and collection of mechanisms from which to start in trying to design truly robust autonomous agents. But, clearly, there's lots of work to be done.

Acknowledgements

This work was supported in part by ONR grant N00014-98-1-0332. It includes essential contributions from the Conscious Software Research Group whose members currently include Ashraf Anwar, Arpad Kelemen, Ravikumar Kondadadi, Lee McCauley, Aregahegn Negatu, Uma Ramamurthy, Zhaohua Zhang.

References

Allen, J. J. 1995. *Natural Language Understanding*. Redwood City CA: Benjamin/Cummings; Benjamin; Cummings.

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

Baars, B. J. 1997. *In the Theater of Consciousness*. Oxford: Oxford University Press.

Bogner, M. 1999. Realizing "consciousness" in software agents. Ph.D. Dissertation. University of Memphis.

Bogner, M., U. Ramamurthy, and S. Franklin. 2000. Consciousness" and Conceptual Learning in a Socially Situated Agent. In *Human Cognition and Social Agent Technology*, ed. K. Dautenhahn. Amsterdam: John Benjamins.

Bogner, M., S. Franklin, A. Graesser, and B. J. Baars. In preparation. Hypotheses From "Conscious" Software. .

Damasio, A. R. 1994. *Descartes' Error.* New York: Gosset; Putnam Press.

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

Franklin, S. 1997. Autonomous Agents as Embodied AI. *Cybernetics and Systems* 28:499–520.

Franklin, S. 2000. Learning in "Conscious" Software Agents. In *Workshop on Development and Learning*. Michigan State University; East Lansing, Michigan, USA: NSF; DARPA; April 5-7, 2000.

Franklin, S. to appear. Conscious Software: A Computational View of Mind. In *Soft Computing Agents: New Trends for Designing Autonomous Systems*, ed. V. Loia, and S. Sessa. Berlin: Springer (Physica-Verlag).

Franklin, S., and A. C. Graesser. 1997. Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In *Intelligent Agents III*. Berlin: Springer Verlag.

Franklin, S., and A. Graesser. 1999. A Software Agent Model of Consciousness. *Consciousness and Cognition* 8:285–305.

Franklin, S., A. Kelemen, and L. McCauley. 1998. IDA: A Cognitive Agent Architecture. In *IEEE Conf on Systems, Man and Cybernetics*. : IEEE Press.

Hofstadter, D. R., and M. Mitchell. 1994. The Copycat Project: A model of mental fluidity and analogy-making. In *Advances in connectionist and neural computation theory, Vol. 2: logical connections*, ed. K. J. Holyoak, and J. A. Barnden. Norwood N.J.: Ablex.

Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press.

Holland, J. H. 1986. A Mathematical Framework for Studying Learning in Classifier Systems. *Physica* 22 D:307–317. (Also in Evolution, Games and

Learning. Farmer, J. D., Lapedes, A., Packard, N. H., and Wendroff, B. (eds.). NorthHolland (Amsterdam))

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

Kanerva, P. 1988. *Sparse Distributed Memory*. Cambridge MA: The MIT Press.

Maes, P. 1989. How to do the right thing. *Connection Science* 1:291–323.

McCauley, T. L., and S. Franklin. 1998. An Architecture for Emotion. In *AAAI Fall Symposium Emotional and Intelligent: The Tangled Knot of Cognition*. Menlo Park, CA: AAAI Press.

Negatu, A., and S. Franklin; 1999. Behavioral learning for adaptive software agents. Intelligent Systems: ISCA 5th International Conference; International Society for Computers and Their Applications - ISCA; Denver, Colorado; June 1999 .

Ramamurthy, U., S. Franklin, and A. Negatu. 1998. Leaming Concepts in Software Agents. In *From animals to animats 5: Proceedings of The Fifth International Conference on Simulation of Adaptive Behavior*, ed. R. Pfeifer, B. Blumberg, J.-A. Meyer , and S. W. Wilson. Cambridge,Mass: MIT Press.

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. : Portland Oregon.

Zhang, Z., D. Dasgupta, and S. Franklin. 1998. Metacognition in Software Agents using Classifier Systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*. Madison, Wisconsin: MIT Press.