Symbiotic Cognitive Computing

Robert Farrell, Jonathan Lenchner, Jeffrey Kephart, Alan Webb, Michael Muller, Thomas Erickson, David Melville, Rachel Bellamy, Daniel Gruen, Jonathan Connell, Danny Soroker, Andy Aaron, Shari Trewin, Maryam Ashoori, Jason Ellis, Brian Gaucher, Dario Gil

■ *IBM Research is engaged in a research program in symbiotic cognitive computing to investigate how to embed cognitive computing in physical spaces. This article proposes five key principles of symbiotic cognitive computing: context, connection, representation, modularity, and adaptation, along with the requirements that flow from these principles. We describe how these principles are applied in a particular symbiotic cognitive computing environment and in an illustrative application for strategic decision making. Our results suggest that these principles and the associated software architecture provide a solid foundation for building applications where people and intelligent agents work together in a shared physical and computational environment. We conclude with a list of challenges that lie ahead.*

In 2011, IBM's Watson competed on the game show *Jeopardy!* winning against the two best players of all time, Brad Rutter and Ken Jennings (Ferrucci et al. 2010). Since this demonstration, IBM has expanded its research program in artificial intelligence (AI), including the areas of natural language processing and machine learning (Kelly and Hamm 2013). Ultimately, IBM sees the opportunity to develop cognitive computing — a unified and universal platform for computational intelligence (Modha et al. 2011). But how might cognitive computing work in real environments — and in concert with people?

In 2013, our group within IBM Research started to explore how to embed cognitive computing in physical environments. We built a Cognitive Environments Laboratory (CEL) (see figure 1) as a living lab to explore how people and cognitive computing come together.

Our effort focuses not only on the physical and computational substrate, but also on the users' experience. We envision a fluid and natural interaction that extends through time across multiple environments (office, meeting room, living room, car, mobile). In this view, cognitive computing systems are always on and available to engage with people in the environment. The system appears to follow individual users, or groups of users, as they change environments, seamlessly connecting the users to available input and output devices and extending their reach beyond their own cognitive and sensory abilities.

We call this *symbiotic cognitive computing:* computation that takes place when people and intelligent agents come together in a physical space to interact with one another. The intelligent agents use a computational substrate of "cogs" for visual object recognition, natural language parsing, probabilistic decision support, and other functions. The term *cog* is from

 ISSN 0738-4602

Figure 1. The Cognitive Environments Lab.

CEL is equipped with movement sensors, microphones, cameras, speakers, and displays. Speech and gesture are used to run cloud-based services, manipulate data, run analytics, and generate spoken and visual outputs. Wands, and other devices enable users to move visual elements in three dimensions across displays and interact directly with data.

the book *The Society of Mind* where Marvin Minsky likened agents to "cogs of great machines" (Minsky 1988). These cogs are available to intelligent agents through programmatic interfaces and to human participants through user interfaces.

Our long-term goal is to produce a physical and computational environment that measurably improves the performance of groups on key tasks requiring large amounts of data and significant mental effort, such as information discovery, situational assessment, product design, and strategic decision making. To date, we have focused specifically on building a cognitive environment, a physical space embedded with cognitive computing systems, to support business meetings for strategic decision making. Other applications include corporate meetings exploring potential mergers and acquisitions, executive meetings on whether to purchase oil fields, and utility company meetings to address electrical grid outages. These meetings often bring together a group of participants with varied roles, skills, expertise, and points of view. They involve making decisions with a large number of high-impact choices that need to be evaluated on multiple dimensions taking into account large amounts of structured and unstructured data.

While meetings are an essential part of business, studies show that they are generally costly and unproductive, and participants find them too frequent, lengthy, and boring (Romano and Nunamaker 2001). Despite this, intelligent systems have the potential to vastly improve our ability to have productive meetings (Shrobe et al. 2001). For example, an intelligent system can remember every conversation, record all information on displays, and answer questions for meeting participants. People making high-stakes, high-pressure decisions have high expectations. They typically do not have the time or desire to use computing systems that add to their workload or distract from the task at hand. Thus, we are aiming for a "frictionless" environment that is always available, knowledgeable, and engaged. The system must eliminate any extraneous steps between thought and computation, and minimize disruptions while bringing important information to the fore.

The remainder of this article is organized as follows. In the next section, we review the literature that motivated our vision of symbiotic cognitive computing. We then propose five fundamental principles of symbiotic cognitive computing. We list some of the key requirements for cognitive environments that implement these principles. We then provide a description of the Cognitive Environments Lab, our cognitive environments test bed, and introduce a prototype meeting-support application we built for corporate mergers and acquisitions (M&A) that runs in this environment. We wrap up by returning to our basic tenets, stating our conclusions, and listing problems for future study.

Background

In his paper "Man-Machine Symbiosis," J. C. R. Licklider (1960) originated the concept of symbiotic computing. He wrote,

> Present-day computers are designed primarily to solve preformulated problems or to process data according to predetermined procedures. … However, many problems … are very difficult to think through in advance. They would be easier to solve, and they could be solved faster, through an intuitively guided trial-and-error procedure in which the computer cooperated, turning up flaws in the reasoning or revealing unexpected turns in the solution.

Licklider stressed that this kind of computer-supported cooperation was important for real-time decision making. He thought it important to "bring computing machines effectively into processes of thinking that must go on in real time, time that moves too fast to permit using computers in conven-

tional ways." Licklider likely did not foresee the explosive growth in data and computing power in the last several decades, but he was remarkably prescient in his vision of man-machine symbiosis.

Distributed cognition (Hutchins 1995) recognizes that people form a tightly coupled system with their environment. Cognition does not occur solely or even mostly within an individual human mind, but rather is distributed across people, the artifacts they use, and the environments in which they operate. External representations often capture the current understanding of the group, and collaboration is mediated by the representations that are created, manipulated, and shared. In activity theory (Nardi 1996), shared representations are used for establishing collective goals and for communication and coordinated action around those goals.

Work on cognitive architectures (Langley, Laird, and Rogers 2009; Anderson 1983; Laird, Newell, and Rosenbloom 1987) focuses on discovering the underlying mechanisms of human cognition. For example, the adaptive control of thought (ACT family of cognitive architectures includes semantic networks for modeling long-term memory and production rules for modeling reasoning, and learning mechanisms for improving both (Anderson, Farrell, and Sauers 1984).

Work on multiagent systems (Genesereth and Ketchpel 1994) has focused on building intelligent systems that use coordination, and potentially competition, among relatively simple, independently constructed software agents to perform tasks that normally require human intelligence. Minsky (1988) explained that "each mental agent in itself can do some simple thing that needs no mind or thought at all. Yet when we join these agents in societies — in certain very special ways — this leads to true intelligence."

Calm technology (Weiser and Brown 1996) suggests that when peripheral awareness is engaged, people can more readily focus their attention. People are typically aware of a lot of peripheral information, and something will move to the center of their attention, for example when they perceive that things are not going as expected. They will then process the item in focus, and when they are done it will fade back to the periphery.

We have used these ideas as the basis for our vision of symbiotic cognitive computing.

Principles of Symbiotic Cognitive Computing

Our work leads us to propose five key principles of symbiotic cognitive computing: context, connection, representation, modularity, and adaption. These principles suggest requirements for an effective symbiosis between intelligent agents and human participants in a physical environment.

The *context principle* states that the symbiosis should be grounded in the current physical and cognitive circumstances. The environment should maintain presence, track and reflect activity, and build and manage context. To maintain presence, the environment should provide the means for intelligent agents to communicate their availability and function, and should attempt to identify people who are available to engage with intelligent agents and with one another. To track and reflect activity, the environment should follow the activity of people and between people and among people, and the physical and computational objects in the environment or environments. It should, when appropriate, communicate the activity back to people in the environment. At other times, it should await human initiatives before communicating or acting. To build and manage context, the environment should create and maintain active visual and linguistic contexts within and across environments to serve as common ground for the people and machines in the symbiosis, and should provide references to shared physical and digital artifacts and to conversational foci.

The *connection principle* states that the symbiosis should engage humans and machines with one another. The environment should reduce barriers, distractions, and interruptions and not put physical (for example, walls) or digital barriers (for example, pixels) between people. The environment should, when appropriate, detect and respond to opportunities to interact with people across visual and auditory modalities. The environment should provide multiple independent means for users to discover and interact with agents. It should enable people and agents to communicate within and across environments using visual and auditory modalities. The environment should also help users establish joint goals both with one another and with agents. Finally, the environment should include agents that are cooperative with users in all interactions, helping users understand the users' own goals and options, and conveying relevant information in a timely fashion (Grice 1975).

The *representation principle* states that the symbiosis should produce representations that become the basis for communication, joint goals, and coordinated action between and among humans and machines. The environment should maintain internal representations based on the tracked users, joint goals, and activities that are stored for later retrieval. The environment should externalize selected representations, and any potential misunderstandings, to coordinate with users and facilitate transfer of representations across cognitive environments. Finally, the environment should utilize the internal and external representations to enable seamless context switching between different physical spaces and between different activities within the same physical space by retrieving the appropriate stored representations for the current context.

The *modularity principle* states that the symbiosis should be driven by largely independent modular composable computational elements that operate on the representations and can be accessed equally by humans and machines. The environment should provide a means for modular software components to describe themselves for use by other agents or by people in the environment. The environment should also provide a means of composing modular software components, which perform limited tasks with a subset of the representation, with other components, to collectively produce behavior for agents. The environment should provide means for modular software components to communicate with one another independently of the people in the environment.

Finally, the *adaptation principle* states that the symbiosis should improve with time. The environment should provide adequate feedback to users and accept feedback from users. Finally, the environment should incrementally improve the symbiosis from interactions with users and in effect, learn.

We arrived at these principles by reflecting upon the state of human-computer interaction with intelligent agents and on our own experiences attempting to create effective symbiotic interactions in the CEL. The context principle originates from our observation that most conversational systems operate with little or no linguistic or visual context. Breakdowns often occur during human-machine dialogue due to lack of shared context. The connection principle arises out of our observation that today's devices are often situated between people and become an impediment to engagement. The representation principle was motivated by our observation that people often resolve ambiguities, disagreements, and diverging goals by drawing or creating other visual artifacts. The use of external representations reduces the domain of discourse and focuses parties on a shared understanding. The modularity principle arose from the practical considerations associated with building the system. We needed ways of adding competing or complementary cogs without reimplementing existing cogs. The adaptation principle was motivated by the need to apply machine learning algorithms to a larger range of human-computer interaction tasks. Natural language parsing, multimodal reference resolution, and other tasks should improve through user input and feedback.

One question we asked ourselves when designing the principles was whether they apply equally to human-human and human-computer interaction. Context, connection, representation, and adaptation all apply equally well to these situations. The modularity principle may appear to be an exception, but the ability to surface cogs to both human and computer participants in the environment enables both better collaboration and improved human-computer interaction.

Cognitive environments that implement these requirements enable people and intelligent agents to be mutually aware of each others' presence and activity, develop connections through interaction, create shared representations, and improve over time. By providing both intelligent agents and human participants with access to the same representations and the same computational building blocks, a natural symbiosis can be supported.

It is impossible to argue that we have found a definitive set of principles; future researchers may find better ones or perhaps more self-evident ones from which the ones we have articulated can be derived. It may even be possible to create a better symbiotic cognitive system than any we have created and not obey one or more of our principles. We look forward to hearing about any such developments.

We are starting to realize these principles and requirements by building prototype cognitive environments at IBM Research laboratories worldwide.

The Cognitive Environments Laboratory

The Cognitive Environments Laboratory is located at the IBM T. J. Watson Research Center in Yorktown Heights, New York. The lab is meant to be a test bed for exploring what various envisioned cognitive environments might be like. It is more heavily instrumented than the vast majority of our envisioned cognitive environments, but the idea is that over time we will see what instrumentation works and what does not. The lab is focused on engaging users with one another by providing just the right technology to support this engagement.

Perhaps the most prominent feature of the CEL is its large number of displays. In the front of the room there is a four by four array of high definition monitors (1920 x 1080 pixel resolution), which act like a single large display surface. On either side of the room are two pairs of high definition monitors on tracks. These monitor pairs can be moved from the back to the front of the room along tracks inlaid in the ceiling, enabling fast and immediate reconfiguration of the room to match many meeting types and activities. In the back of the room there is an 84-inch touch-enabled 3840 x 2160 pixel display. The monitors are laid out around the periphery of the room. Within the room, visual content can either be moved programmatically or with the aid of special ultrasound-enabled pointing devices called "wands" or with combinations of gesture and voice, from monitor to monitor or within individual monitors.

In addition to the displays, the room is outfitted with a large number of microphones and speakers. There are several lapel microphones, gooseneck microphones, and a smattering of microphones attached to the ceiling. We have also experimented with array microphones that support "beam forming" to isolate the speech of multiple simultaneous

speakers without the need for individual microphones.

An intelligent agent we named Celia (cognitive environments laboratory intelligent agent) senses the conversation of the room occupants and becomes a supporting participant in meetings. With the aid of a speech-to-text transcription system, the room can document what is being said. Moreover, the text and audio content of meetings is continuously archived. Participants can ask, for example, to recall the transcript or audio of all meetings that discussed "graph databases" or the segment of the current meeting where such databases were discussed. Transcribed utterances are parsed using various natural language processing technologies, and may be recognized as commands, statements, or questions. For example, one can define a listener that waits for key words or phrases that trigger commands to the system to do something. The listener can also test whether certain preconditions are satisfied, such as whether certain objects are being displayed. Commands can invoke agents that retrieve information from the web or databases, run structured or unstructured data analytics, route questions to the Watson question-answering system, and produce interactive visualizations. Moreover, with the aid of a text-to-speech system, the room can synthesize appropriate responses to commands. The system can be configured to use the voice of IBM Watson or a different voice.

In addition to the audio and video output systems, the room contains eight pan-tilt-zoom cameras, four of which are Internet Protocol (IP) cameras, plus three depth-sensing devices, one of which is gimbal mounted with software-controllable pan and tilt capability. The depth-sensing systems are used to detect the presence of people in the room and track their location and hand gestures. The current set of multichannel output technologies (that is, including screens and speakers) and multichannel input technologies (that is, keyboard, speech-to-text, motion) provide an array of mixed-initiative possibilities.

People in the CEL can simultaneously gesture and speak to Celia to manipulate and select objects and operate on those objects with data analytics and services. The room can then generate and display information and generate speech to indicate objects of interest, explain concepts, or provide affordances for further interaction. The experience is one of interacting with Celia as a gateway to a large number of independently addressable components, cogs, many of which work instantly to augment the cognitive abilities of the group of people in the room.

Dependence on a single modality in a complex environment generally leads to ineffective and inconvenient interactions (Oviatt 2000). Environments can enable higher level and robust interactions by exploiting the redundancy in multimodal inputs (speech, gesture, vision). The integration of speech and gesture modalities has been shown to provide both flexibility and convenience to users (Krum 2002). Several prototypes have been implemented and described in the literature. Bolt (1980) used voice and gesture inputs to issue commands to display simple shapes on a large screen. Sherma (2003) and Carbini (2006) extended this idea to a multiuser interaction space. We have built upon the ideas in this work in creating the mergers and acquisitions application.

Mergers and Acquisitions Application

In 2014 and 2015 we built a prototype system, situated in the Cognitive Environments Laboratory, for exploring how a corporate strategy team makes decisions regarding potential mergers and acquisitions. As depicted in figure 2, one or more people can use speech and gestures to interact with displayed objects and with Celia. The system has proven useful for exploring some of the interaction patterns between people and intelligent agents in a high-stakes decision-making scenario, and for suggesting architectural requirements and research challenges that may apply generally to symbiotic cognitive computing systems.

In the prototype, specialists working on mergers and acquisitions try to find reasonable acquisition targets and understand the trade-offs between them. They compare companies side by side and receive guidance about which companies are most aligned with their preferences, as inferred through repeated interactions. The end result is a small set of companies to investigate with a full-fledged "due diligence" analysis that takes place following the meeting.

When the human collaborators have interacted with the prototype to bring it to the point depicted in figure 2, they have explored the space of mergers and acquisitions candidates, querying the system for companies with relevant business descriptions and numeric attributes that fall within desired ranges, such as the number of employees and the quarterly revenue. As information revealed by Celia is interleaved with discussions among the collaborators, often triggered by that information, the collaborators develop an idea of which company attributes matter most to them. They can then invoke a decision table to finish exploring the trade-offs.

Figure 3 provides a high-level view of the cognitive environment and its multiagent software architecture. Agents communicate with one another through a publish-and-subscribe messaging system (the message broker) and through HTTP web services using the Representational State Transfer (REST) software design pattern. The system functions can be divided into command interpretation, command execution, agent management, decision making, text and data analysis, text-to-speech, visualization, and manage-

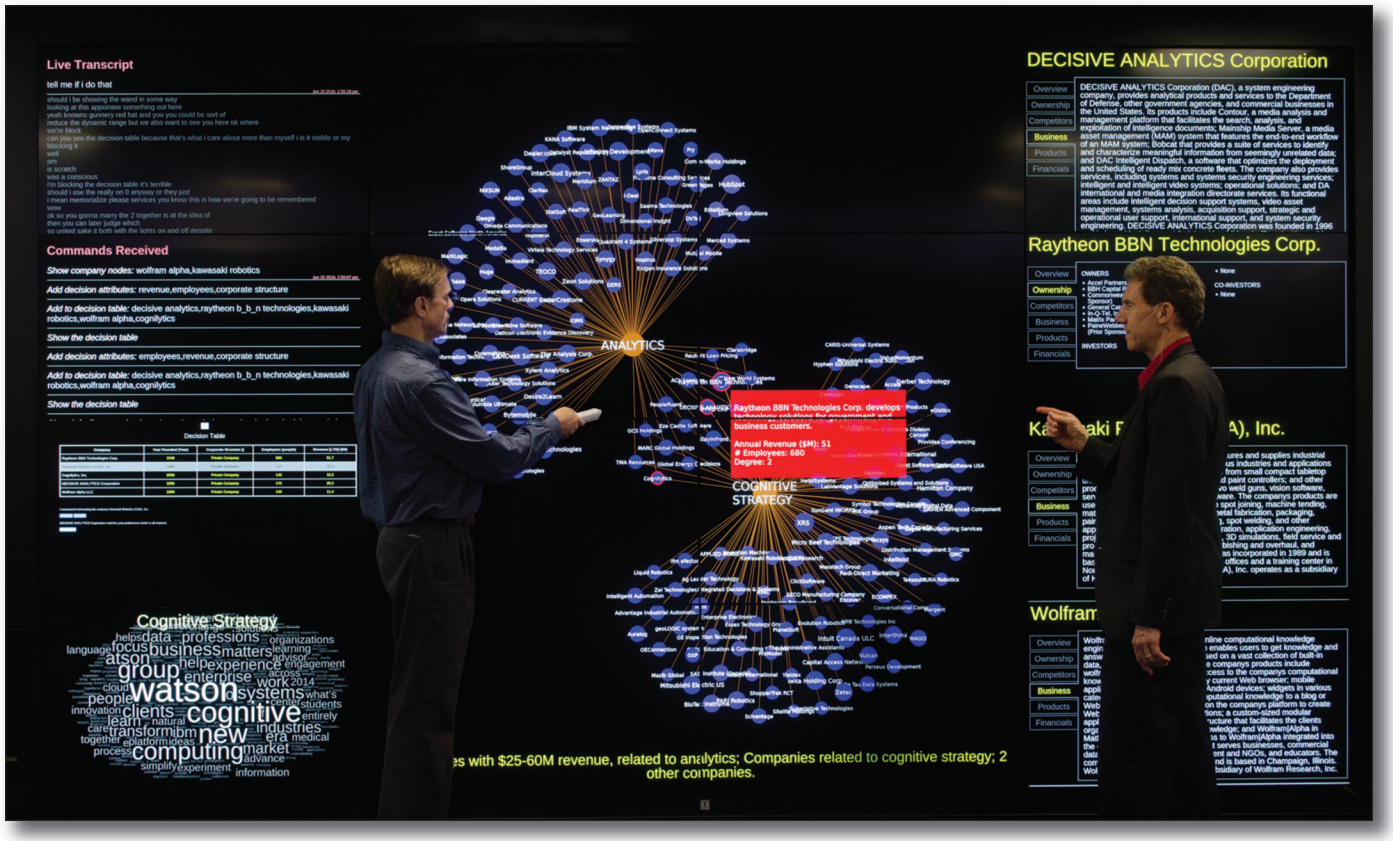

Figure 2. The Mergers and Acquisitions Prototype Application.

People working with one another and with Celia to discover companies that match desired criteria obtain detailed information about likely candidates and winnow the chosen companies down to a small number that are most suitable.

ment. We explain each of these functions in the sections that follow.

Command Interpretation

The system enables speech and gesture to be used together as one natural method of communication. Utterances are captured by microphones, rendered into text by speech recognition engines, and published to a "transcript" message channel managed by the message broker. The message broker supports high-performance asynchronous messaging suitable for the real-time concurrent communication in the cognitive environment.

We have tested and customized a variety of speech-recognition engines for the cognitive environment. The default engine is IBM Attila (Soltau, Saon, and Kingsbury 2010). It has two modes: a first that renders the transcription of an utterance once a break is detected on the speech channel (for example, half a second of silence), and a second that renders a word-by-word transcription without waiting for a pause. In the former mode there is some probability that subsequent words will alter the assessment of earlier words. We run both modes in parallel to enable agents to read and publish partial interpretations immediately.

Position and motion tracking uses output from the position and motion sensors in combination with visual object recognition using input from the cameras to locate, identify, and follow physical objects in the environment. The user identity tracking agent maps recognized people to unique users using acoustic speaker identification, verbal introduction ("Celia, I am Bob"), facial recognition upon entry, or other methods. The speaker's identity, if known, is added to each message on the transcript channel, making it possible to interleave dialogues to some degree, but further research is needed to handle complex multiuser dialogues. The persistent session information includes persistent identities for users, including name, title, and other information collected during and across sessions.

The natural language parsing agent subscribes to the transcript channel, processes text transcriptions of utterances containing an attention word (for example, "Celia" or "Watson") into a semantic representation that captures the type of command and any relevant parameters, and publishes the representation to a command channel. Our default parser is

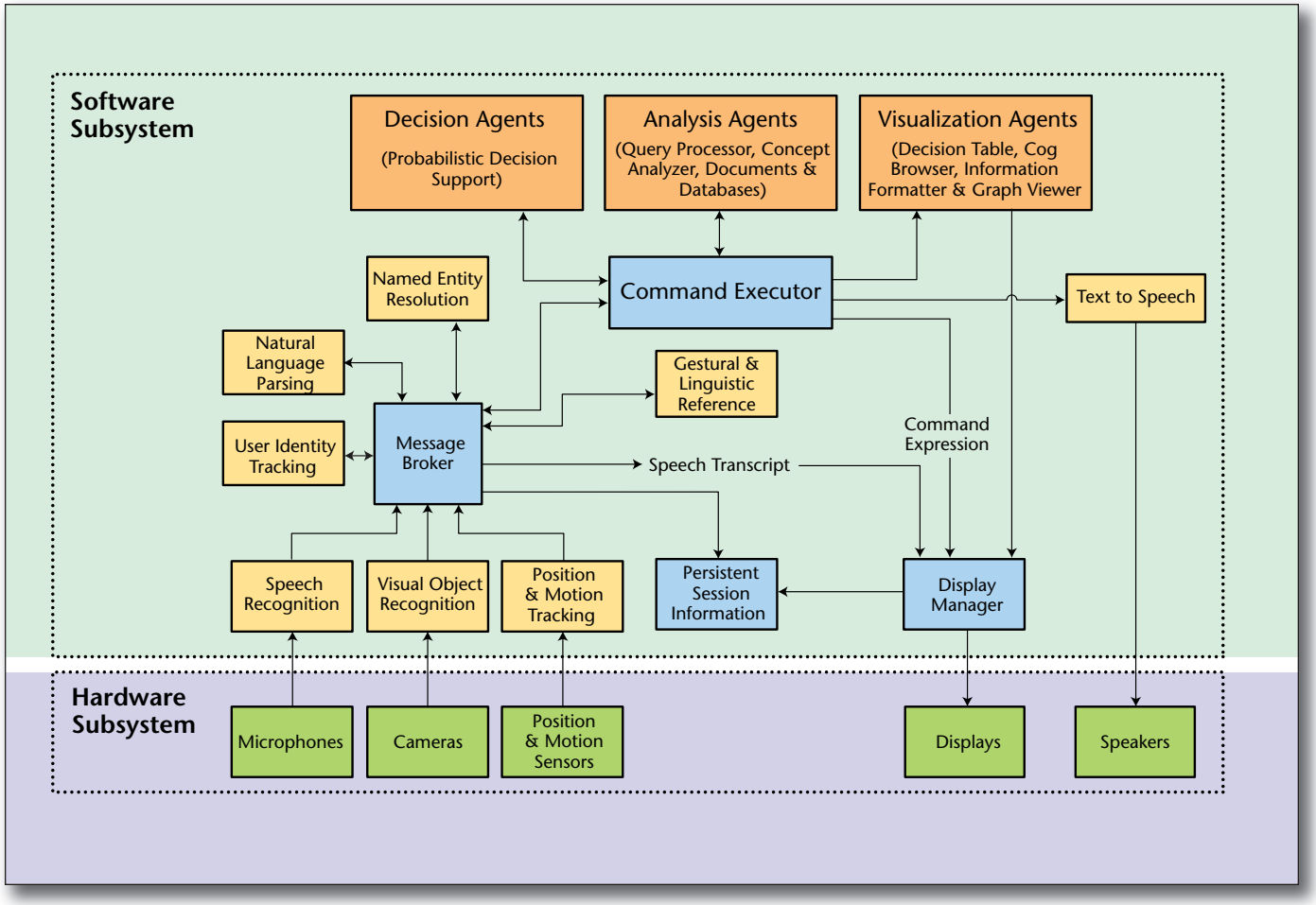

Figure 3. Architecture of the Mergers and Acquisitions Prototype.

based on regular expression matching and template filling. It uses a hierarchical, composable set of functions that match tokens in the text to grammatical patterns and outputs a semantic representation that can be passed on to the command executor. Most sentences have a subject-verb-object structure, with "Celia" as the subject, a verb that corresponds to a primitive function of one of the agents or a more complex domain-specific procedure, and an object that corresponds to one or more named entities, as resolved by the named entity resolution agent, or with modifiers that operate on primitive types such as numbers or dates. Another parser we have running in the cognitive environment uses a semantic grammar that decomposes top-level commands into terminal tokens or unconstrained dictation of up to five words. The resulting parse tree is then transformed into commands, each with a set of slots and fillers (Connell 2014). A third parser using Slot Grammar (SG) is from the Watson System (McCord, Murdock, and Boguraev 2012). It is a deep parser, producing both syntactic structure and semantic annotations on input sentences. Interrogatives can be identified by the SG parser and routed directly to a version of the IBM Watson system, with the highest confidence answer generated back to the user through text-to-speech.

An important issue that arose early in the development of the prototype was the imperfection of speech transcription. We targeted a command completion rate of over 90 percent for experienced users, but with word-recognition accuracies in the low to mid 90 percent range and commands ranging from 5 to 20 or more words in length, we were not achieving this target. To address this deficiency, we developed several mechanisms to ensure that speech-based communication between humans and Celia would work acceptably in practice. First, using a half hour of utterances captured from user interaction with the prototype, we trained a speech model and enhanced this with a domain-specific language model using a database of 8500 company names extracted from both structured and unstructured sources. The named entity resolution agent was extended to resolve acronyms and abbreviations and to match both phonetically and lexicographically. To provide

better feedback to users, we added a speech transcript. Celia's utterances are labeled with "Celia," and a command expression is also shown to reflect the command that the system processed. If the system doesn't respond as expected, users can see whether Celia misinterpreted their request, and if so, reissue it. Finally, we implemented an undo feature to support restoration of the immediately prior state when the system misinterprets the previous command.

The gestural and linguistic reference agent is responsible for fusing inputs from multiple modes into a single command. It maintains persistent referents for recently displayed visual elements and recently mentioned named entities for the duration of a session. When Celia or a user mentions a particular company or other named entity, this agent creates a referent to the entity. Likewise, when the display manager shows a particular company or other visual element at the request of a user or Celia, a referent is generated. Using the referents, this agent can find relevant entities for users or Celia based on linguistic references such as pronouns, gestures such as pointing, or both. Typically the referent is either something the speaker is pointing toward or is something that has recently been mentioned. In the event that the pronoun refers to something being pointed at, the gestural and linguistic reference agent may need to use the visual object recognition's output, or if the item being pointed at resides on a screen, the agent can request the virtual object from the appropriate agent.

Command Execution

The command executor agent subscribes to the command channel and oversees command execution. Some of its functions may be domain-specific. The command executor agent communicates over HTTP web services to the decision agents, analysis agents, and visualization agents. Often, the command executor serves as an orchestrator, calling a first agent, receiving the response, and reformulating that response to another agent. Thus, the command executor maintains state during the utterance. A single request from a user may trigger a cascade of agent-to-agent communication throughout the command execution part of the flow, eventually culminating in activity on the displays and/or synthesized speech being played over the speakers.

Decision Making

In the mergers and acquisitions application, people have high-level goals that they want the system to help them achieve. A common approach from decision theory is to specify a utility function (Walsh et al. 2004). Given a utility function defined in terms of attributes that are of concern to the user, the system's objective is to take actions or adjust controls so as to reach a feasible state that yields the highest possible utility. A particularly attractive property of this approach is that utility can be used to propagate objectives through a system from one agent to another. However, experience with our symbiotic cognitive computing prototype suggests that this traditional approach misses something vital: people start with inexact notions of what they want and use computational tools to explore the options. It is only during this sometimes-serendipitous exploration process that they come to understand their goals better. Certain companies appeal to us, sometimes before we even know why, and it can take a serious introspection effort (an effort that may be assisted by the cognitive system) to discover which attributes matter most to us or to realize we may be biased. This realization prompted us to design a probabilistic decision support agent (Bhattacharjya and Kephart 2014). This agent starts with a set of candidate solutions to the decision support problem and attributes, and a highly uncertain model of user preferences. As the user accepts or rejects its recommendations, or answers questions about trade-offs, the agent progressively sharpens its understanding of the users' objectives, which it models as a probability distribution of weights in the space of possible utility functions. The agent is able to recommend filtering actions, such as removing companies or removing attributes, to help users converge on a small number of targets for mergers and acquisitions.

Text and Data Analysis

The concept analyzer agent provides additional data for decision making by extracting concepts and relationships from documents. While IBM Watson was trained for general question answering using primarily open web sources such as Wikipedia, we anticipate that most applications will also involve harnessing data from private databases and third-party services. For the mergers and acquisitions application, we developed a large database of company annual reports. Concepts extracted from the reports can be linked to the companies displayed in the graph viewer and when a user asks for companies similar to a selected company, the system is able to retrieve companies through the concepts and relationships. The query processor agent provides a query interface to a database of company financial information, such as annual revenue, price-earnings ratio, and income.

Persistent Session Information

We have added the ability for the cognitive environment to capture the state of the interaction between users and agents either as needed or at the end of a session. It does this by creating a snapshot of what agents are active, what is being displayed, what has been said, and what commands have been completed. The snapshots are saved and accessible from any device thus enabling users to take products of the work session outside of the cognitive environment.

The session capture feature allows users to review past decisions and can support case-based reasoning (Leake 1996). It also provides continuity because users can stop a decision-making process and continue later. Finally, it allows for some degree of portability across multiple cognitive environments.

Text-to-Speech

The text-to-speech agent converts text into spoken voice. The current system has a choice of two voices: a North American English female voice or a North American English male voice (which was used by the IBM Watson system). In order to keep interruptions to a minimum, the speech output is used sparingly and usually in conjunction with the visual display.

Visualization

The visualization agents work in the cognitive environment's multiuser multiscreen networked environment. Celia places content on the 25 displays and either the visualization agents or the users then manipulate the content in three dimensions. We designed the M&A application visualizations to work together in the same visual space using common styles and behaviors. The display manager coordinates content placement and rendering using placement defaults and constraints. The cog browser enables people to find and learn about the available cogs in the cognitive environment. It displays a cloud of icons representing the society of cogs. The icons can be expanded to reveal information about each cog and how they are invoked.

Many agents have functions that can be addressed through speech commands or through gesture. For example, the information formatter can display company information (on the right in figure 2) and allows a user to say "products" or select the products tab to get more information about the company's products. In addition, many of the agents that provide building blocks for Celia's decision-making functions are cogs that are also independently addressable through visual interfaces and speech commands. For example, the graph viewer can be used by Celia to recommend companies but is also available to users for visualizing companies meeting various criteria.

Agent Management

We have implemented several management modules that operate in parallel with the other parts of the system to allow agents to find instances of other agents that are running and thereby enable discovery and communication. When an agent is first launched, it registers itself to a life-cycle manager module to advertise its REST interfaces and types of messages it publishes over specific publish-and-subscribe channels. When an agent requires the services of a second agent, it can locate an instance of the second agent by querying the lifecycle manager, thereby avoiding the need to know details of the second agent's running instance.

Exchange 1

Brian: Celia, this is Brian. I need help with acquisitions.

Celia: Hello Brian, how can I help you with mergers and acquisitions?

Exchange 2

Brian: Celia, show me companies with revenue between $25 million and $50 million and between 100 and 500 employees, pertaining to analytics.

Celia: Here is a graph showing 96 companies pertaining to biotechnology *(Celia displays the graph).*

Exchange 3

Brian: Celia, place the companies named brain science, lintolin, and tata, in a decision table.

Celia: Ok. (*Celia shows a table with the 3 companies, one per row, and with columns for the name of the company, the revenue, and number of employees).*

Celia: I suggest removing Lyntolin. Brain Sciences, Incorporated has greater revenue and greater number of employees (*Celia highlights Brain Sciences and Lyntolin).*

Figure 4. A Sample Dialogue Processed by the Mergers and Acquisitions Prototype.

The agents used in the M&A application and others are available as cogs in the CEL and work as one intelligent agent to provide access to cognitive computing services. Taken together, the agents provide a completely new computing experience for business users facing tough decisions.

A sample dialogue is shown in figure 4. To handle exchange 1, the system processes the first sentence using the speech recognition engine and sends a message with the transcribed text to the message broker on the transcript channel. The natural language parser listens on the transcript channel and publishes a semantic representation based on a dependency parse of the input that identifies the name Brian in the object role. The user identity tracking agent resolves the name against the persistent identifier for Brian and publishes a command on the command channel with a set user identifier. The command executor then requests the speech recognition agent to start using the speaker's speech model. The next sentence is processed similarly, but the gestural and linguistic reference agent resolves "I" to "Brian." The verb "help" is recognized as the main action and the actor is the persistent identifier for Brian. The command executor recognizes the representation as the initialization of the mergers and acquisitions application using a pattern rule. It calls the display manager, which saves and hides the state of the displays. The command executor then generates the response that is sent to the text to speech agent. This requires querying the user identity track-

ing agent to map the persistent identifier for Brian to his name.

To handle exchange 2, a similar flow happens, except the command executor calls the query processor agent to find companies matching the revenue and company size criteria. Upon receiving the response, the command executor calls the graph viewer, which adds nodes representing each matching biotechnology company to a force-directed graph on the display (in the center in figure 2). The command executor also calls the text-to-speech agent to play an acknowledgement over the speakers. The user can then manipulate the graph using gestures or issue further speech commands.

For exchange 3, the command executor first calls the named entity resolver three times to resolve the exact names of the companies referred to by the user; for example it might resolve "brain science" into "Brain Sciences, Incorporated." Upon receiving these responses, the command executor calls the query processor agent to obtain company information, which it then sends to the probabilistic decision support agent. This agent must interact with the decision table agent, which in turn uses the display manager to display the output to the user. While all of this is happening, the executor also calls the text-to-speech agent to acknowledge the user's request. Coordination between speech and display thus happens in the command executor. During this interaction, the transcript displayer and command displayer display the utterance and the interpreted command.

In the next section, we discuss our work to date on the cognitive environment in terms of both prior work and our original symbiotic cognitive computing principles.

Discussion

Prior intelligent decision-making environments focus primarily on sensor fusion, but fall short of demonstrating an intelligent meeting participant (Ramos et al 2010). The New EasyLiving Project attempted to create a coherent user experience, but was focused on integrating I/O devices (Brumitt et al. 2000). The NIST Meeting Room has more than 280 microphones, seven HD cameras, a smart whiteboard, and a locator system for the meeting attendees (Stanford et al. 2003), but little in the way of intelligent decision support. The CALO (Cognitive Assistant that Learns and Organizes) DARPA project includes a meeting assistant that captures speech, pen, and other meeting data and produces an automated transcript, segmented by topic, and performs shallow discourse understanding to produce a list of probable action items (Voss and Ehlen 2007), but it does not focus on multimodal interaction.

The experience of building and using the M&A application has been valuable in several respects. First, while we haven't yet run a formal evaluation, we've found that the concept of a cognitive environment for decision making resonates well with business users. To date we have now had more than 50 groups of industry executives see a demonstration and provide feedback. We are now working closely with the mergers and acquisitions specialists at IBM to bring aspects of the prototype into everyday use. Second, the prototype has helped us to refine and at least partially realize the symbiotic cognitive computing principles defined in this article, and to gain a better understanding of the nature of the research challenges. Here we assess our work on the prototype in terms of those principles.

First, how much of the symbiosis is grounded in the physical and cognitive circumstances? We have just started to explore the use of physical and linguistic context to shape the symbiosis. Some aspects of maintaining presence are implemented. For example, motion tracking and visual recognition are used to capture the presence of people in the room. However, endowing intelligent agents with the ability to effectively exploit information about individual people and their activities and capabilities remains a significant research challenge. Multiple people in the cognitive environment can interact with the system, but the system's ability to associate activities with individual users is limited. The session capture agent tracks both human commands and agent actions, but additional work is required to reflect the activity of people and agents in the environment back to participants. The linguistic and gestural reference agent maintains some useful context for switching between applications, but additional research is needed to exploit this context to enable an extended dialogue.

Second, how much does the cognitive environment support the connection principle, enabling people and intelligent agents to engage with one another? We feel that the architecture and implementation support all of the requirements, at least to some degree. First, barriers between human intention and system execution are reduced by multimodal interactions that allow users to converse with the system almost as if it were a human partner rather than having to deal with the cumbersome conventions of typical user interfaces, but the lack of affordances in speech-based interaction remains a challenge. The cog browser provides users with some understanding of the capabilities of various cogs but the system does not offer assistance. The system supports interactions across multiple environments; cogs can in effect follow the user to different physical spaces, marshaling the input and output resources that they find there — thereby reducing the time required to initiate system computations and actions when moving across cognitive environments. The cognitive environment cooperates with users in multiple ways: decision agents use elicitation techniques to develop an understanding of user goals and trade-offs and then

to guide users toward decisions that best realize them, Celia listens for commands and the command executor responds only when adequate information is available for a response. Because multiple users can see the same display and Celia has access to displayed objects through the display manager, Celia can track and respond to their coordinated actions.

Does the cognitive environment support the representation principle? The CEL and its agents maintain representations that are the basis for communication between participants and with Celia. The identity and location of users in the room, the developing conversation with Celia and recognized commands, and the state of ongoing decisions are all captured, externalized, and shared outside the environment, providing common ground between both people in the physical environment and those connected to the environment only remotely or periodically. In future work, we would like to recognize individual differences in representation preferences, and be able to conduct "private" interactions with individual users through the media of their choice.

We have realized the modularity requirements of self-description, composition, and intercomponent communication by implementing the cognitive environment as a multiagent system. Some agents are strongly human centered, providing services such as speech and gesture transcription, speech synthesis, or visualization. Others mainly serve the needs of other agents. For example, the life-cycle manager facilitates agent communication and composition by enabling agents to advertise their capabilities to one another and use one another's services. An important research challenge is to create deeper semantic descriptions of services to allow users to select and compose services as needed through natural language dialogue.

How does the cognitive environment support adaptation, improving with time? Currently most of the system's improvement is offline, not during the dialogue. For example, we trained speech models and extended the language model with custom dictionaries. Users' gestural vocabularies could also be modeled and interpreted, or we could develop individualized models of combinations of speech, gesture, and larger movements. We currently capture useful data during interactions with users that can be used to improve interactions in the future. For example, the system captures linguistic and gestural context, which can in principle be mined by other agents seeking to detect patterns that might be used to better anticipate user needs. Ultimately we would like cognitive systems to adapt to users' goals and capabilities, available I/O resources, and available cogs to maximize the effectiveness of the entire session and improve the symbiotic relationship between users and the system.

Despite our successes with engineering a symbiotic cognitive computing experience, practical applications continue to be a challenge: speech commands in a noisy room are often misrecognized and we cannot reliably identify individual speakers, the tracking of gestures is still error prone with both wands and hand gestures, and natural language inputs require domain-specific natural language engineering to map commands to the proper software services and invoke decision, analysis, and visualization agents. Despite these challenges, the cognitive environment provides a valuable test bed for integrating a variety of IBM Research cognitive computing technologies into new scenarios and business applications.

Conclusions

This article introduced our work on symbiotic cognitive computing. We outlined five principles: context, connection, representation, modularity, and adaptation, and we showed how key requirements that flow from these principles could be realized in a cognitive environment. We have started to apply this environment to real business problems, including strategic decision making for corporate mergers and acquisitions.

Reaching a true symbiosis between cognitive computing and human cognition is a significant multiyear challenge. The IBM Watson question-answering system and other intelligent agents can be embedded in physical spaces, but additional research is needed to create cognitive computing systems that can truly sense the world around them and fully interact with people to solve difficult problems.

Our future directions include detection and understanding of emotion, cognitive computing in virtual and mixed reality, simultaneous speech and gesture understanding, integration of uncertain data from sensors into real-time interaction, and machine learning to improve decisions over time. We are also interested in exploring tasks such as information discovery, situational assessment, and product design where difficult decisions require bringing together people who have complementary skills and experience and providing them with large amounts of structured and unstructured data in one collaborative multisensory multimodal environment.

The IBM Watson *Jeopardy!* system demonstrated the ability of machines to achieve a high level of performance at a task normally considered to require human intelligence. We see symbiotic cognitive computing as the next natural step in the evolution of intelligent machines: creating machines that are embedded in the world and integrate with every aspect of life.

Acknowledgements

Thanks also to Wendy Kellogg, Werner Geyer, Casey Dugan, Felicity Spowart, Bonnie John, Vinay Venkataraman, Shang Gao, Mishal Dholakia, Tomas Beren, Yedendra Shirnivasan, Mark Podlaseck, Lisa Amini, Hui Su, and Guru Banavar.

References

Anderson, J. 1983. *The Architecture of Cognition*. Mahwah, NJ: Lawrence Erlbaum Associates.

Anderson, J.; Farrell, R.; and Sauers, R. 1984. Learning to Program in LISP. *Cognitive Science* 8(2): 87–129. dx.doi.org/10.1207/s15516709cog0802_1

Bhattacharjya, D., and Kephart, J. O. 2014. Bayesian Interactive Decision Support for Multi-Attribute Problems with Even Swaps. In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence,* 72–81. Seattle, WA: AUAI Press.

Bolt, R. A. 1980. Put-That-There. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques: SIGGRAPH '80,* 262–270. New York: Association for Computing Machinery. dx.doi.org/10.1145/800250.807503

Brumitt, B. L.; Meyers, B.; Krumm, J.; Kern, A.; and Shafer, S. 2000. EasyLiving: Technologies for Intelligent Environments. In *Handheld and Ubiquitous Computing, 2nd International Symposium,* September, Lecture Notes in Computer science Volume 1927, ed P. Thomas and H.-W. Gellersen, 12–27. Berlin: Springer.

Carbini, S.; Delphin-Poulat, L.; Perron, L.; and Viallet, J. 2006. From a Wizard of Oz Experiment to a Real Time Speech and Gesture Multimodal Interface. *Signal Processing* 86(12): 3559-3577. dx.doi.org/10.1016/j.sigpro.2006.04.001.

Connell, J. 2014. Extensible Grounding of Speech for Robot Instruction. In *Robots That Talk and Listen,* ed. J. Markowitz,175–201. Berlin: Walter de Gruyter GmbH and Co.

Ferrucci, D.; Brown, E.; Chu-Carroll, J.; Fan, J.; Gondek, D.; Kalyanpur, A. A.; Lally, A.; Murdock, J. W.; Nyberg, E.; Prager, J.; Schlaefer, N.; and Welty, C. 2010. Building Watson: An Overview of the DeepQA Project. *AI Magazine* 31(3): 59–79.

Genesereth, M. R., and Ketchpel, S. P. 1994. Software Agents. *Communications of the ACM* 37(7), 48–53. dx.doi.org/10.1145/176789.176794

Grice, P. 1975. Logic and Conversation. In *Syntax and Semantics. 3: Speech Acts,* ed. P. Cole and J. Morgan, 41–58. New York: Academic Press.

Hutchins, E. 1995. *Cognition in the Wild*. Cambridge, MA: The MIT Press.

Kelly, J., and Hamm, S. 2013. *Smart Machines: IBM's Watson and the Era of Cognitive Computing*. New York: Columbia University Press.

Krum, D.; Omoteso, O.; Ribarsky, W.; Starner, T.; and Hodges, L. 2002. Speech and Gesture Multimodal Control of a Whole Earth 3D Visualization Environment. In *Proceedings of the 2002 Joint Eurographics and IEEE TCVG Symposium on Visualization,* 195-200. Goslar, Germany: Eurographics Association.

Laird, J.; Newell, A.; and Rosenbloom, P. 1987. SOAR: An Architecture for General Intelligence. *Artificial Intelligence* 33(1): 1–64. dx.doi.org/10.1016/0004-3702(87)90050-6

Langley, P.; Laird, J. E.; and Rogers, S. 2009. Cognitive Architectures: Research Issues and Challenges. *Cognitive Systems Research* 10(2): 141–160. dx.doi.org/10.1016/j.cogsys.2006.07.004

Leake, D. B. 1996. *Case-Based Reasoning: Experiences, Lessons, and Future Directions*. Menlo Park, CA: AAAI Press.

Licklider, J. C. R. 1960. Man-Computer Symbiosis. *IRE Transactions on Human Factors in Electronics* Volume HFE-1(1): 4–11. www.dx.doi.org/10.1109/THFE2.1960.4503259

McCord, M. C.; Murdock, J. W.; and Boguraev, B. K. 2012. Deep Parsing in Watson. *IBM Journal of Research and Development* 56(3.4): 3:1–3:15.

Minsky, M. 1988. *The Society of Mind*. New York: Simon and Schuster.

Modha, D. S.; Ananthanarayanan, R.; Esser, S. K.; Ndirango, A.; Sherbondy, A. J.; and Singh, R. 2011. Cognitive Computing. *Communications of the ACM* 54(8): 62–71. dx.doi.org/10.1145/1978542.1978559

Nardi, B. A. 1996. Activity Theory and Human-Computer Interaction. In *Context and Consciousness: Activity Theory and Human-Computer Interaction,* ed. B. Nard, 7–16. Cambridge, MA: The MIT Press.

Oviatt, S.; and Cohen, P. 2000. Perceptual User Interfaces: Multimodal Interfaces that Process what Comes Naturally. *Communications of the ACM* 43(3): 45-53. dx.doi.org/10.1145/330534.330538Ramos, C.; Marreiros, G.; Santos, R.; and Freitas, C. F. 2010. Smart Offices and Intelligent Decision Rooms. In *Handbook of Ambient Intelligence and Smart Environments,* ed. H. Nakashima, H. Aghajan, and J. C. Augusto, 851–880. Berlin: Springer. dx.doi.org/10.1007/978-0-387-93808-0_32

Romano, N. C., and Nunamaker, J. F. 2001. Meeting Analysis: Findings from Research and Practice. In *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*. Los Alamitos, CA: IEEE Computer Society.

Sharma, R.; Yeasin, M.; Krahnstoever, N.; Rauschert, I.; Cai, G.; Brewer, I.; MacEachren, A.; Sengupta, K. 2003. Speech-Gesture Driven Multimodal Interfaces for Crisis Management. Proceedings of the IEEE 91(9): 1327–1354. dx.doi.org/10.1109/JPROC.2003.817145

Shrobe, H.; Coen, M.; Wilson, K.; Weisman, L.; Thomas, K.; Groh, M.; Phillips, B.; Peters, S.; Warshawsky, N.; and Finin, P. 2001. The Intelligent Room. MIT AI Laboratory AFRL-IF-RS-TR-2001-168 Final Technical Report. Rome, New York: Air Force Research Laboratory.

Soltau, H.; Saon, G.; and Kingsbury, B. 2010. The IBM Attila Speech Recognition Toolkit. In 2010 IEEE Workshop on Spoken Language Technology, SLT 2010 — Proceedings, 97–102. Piscataway, NJ: Institute for Electrical and Electronics Engineers. dx.doi.org/10.1109/slt.2010.5700829

Stanford, V.; Garofolo, J.; Galibert, O.; Michel, M; and Laprun, C. 2003. The (NIST) Smart Space and Meeting Room Projects: Signals, Acquisition Annotation, and Metrics. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03),* 4, 6–10. Piscataway, NJ: Institute for Electrical and Electronics Engineers. dx.doi.org/10.1109/icassp.2003.1202748

Voss, L. L., and Ehlen, P. 2007. The CALO Meeting Assistant. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. Stroudsberg, PA: Association for Computational Linguistics 2007. dx.doi.org/10.3115/1614164.1614173

Walsh, W. E.; Tesauro, G.; Kephart, J. O.; and Das, R. 2004. Utility Functions in Autonomic Systems. In *Proceedings of the 1st International Conference on Autonomic Computing (ICAC 2004),* 70–77. Piscataway, NJ: Institute for Electrical and Electronics Engineers. dx.doi.org/10.1109/ICAC.2004.1301349

Weiser, M., and Brown, J. S. 1996. Designing Calm Technology. *PowerGrid Journal* 1.01, 1–5.

Robert Farrell is a research staff member at the IBM T. J. Watson Research Center in Yorktown Heights, NY, USA. He has a long-term research interest in the cognitive processes of human learning, knowledge representation, reasoning, and language understanding. His past work includes cognitive models, intelligent tutoring systems, and social computing applications. He is currently working on software to extract knowledge from unstructured information sources.

Jonathan Lenchner is chief scientist at IBM Research-Africa. Previously he was one of the founders of the IBM Cognitive Environments Lab in Yorktown Heights, NY. His research interests include computational geometry, robotics, AI, and game theory. His recent work includes research on humanoid robots and development of an immersive environment to help a professional sports team with trades and draft picks.

Jeffrey Kephart is a distinguished research staff member at IBM T. J. Watson Research Center, and a Fellow of the IEEE. He is known for his work on computer virus epidemiology and immune systems, self-managing computing systems, electronic commerce, and data center energy management. Presently, he serves as a principal investigator on a cognitive computing research project with a large energy company and leads work on applying intelligent agent technologies to corporate mergers and acquisitions.

Alan Webb is a senior software engineer at the IBM T. J. Watson Research Center. His present research interests are focused upon applying the principles of distributed cognition as an inspiration for pervasive cognitive environments. He is currently working on a generalized system architecture for the cognitive environment and development of the mergers and acquisitions application.

Michael Muller is a research staff member in the Cognitive User Experience group at IBM Research in Cambridge, MA. His research areas have included collaboration in health care, metrics and analytics for enterprise social software, participatory design, and organizational crowdfunding. His current work focuses on employee experiences in the workplace.

Thomas Erickson is a social scientist and interaction designer at the IBM T. J. Watson Research Center. His research has to do with designing systems that enable groups of people to interact coherently and productively in both virtual and real environments.

David Melville is a research staff member at IBM T. J. Watson Research Center. His research interests include immersive data spaces, spatial computing, adaptive physical architecture, and symbiotic experience design.

Rachel Bellamy is a principal research staff member and group manager at IBM T. J. Watson Research Center and heads the Research Design Center. Her general area of research is human-computer interaction and her current work focuses on the user experience of symbiotic cognitive computing.

Daniel Gruen is a cognitive scientist in the Cognitive User Experience group at IBM Research in Cambridge, MA. He is interested in the design of systems that let strategic decision makers seamlessly incorporate insights from cognitive computing in their ongoing deliberations and creative thinking. He is currently working with a variety of companies to understand how such systems could enhance the work they do.

Jonathan Connell is a research staff member at IBM T. J. Watson Research Center. His research interests include computer vision, machine learning, natural language, robotics, and biometrics. He is currently working on a speech-driven reactive reasoning system for multimodal instructional dialogue.

Danny Soroker is a research staff member at IBM T. J. Watson Research Center. His research interests include intelligent computation, human-computer interaction, algorithms, visualization, and software design. He is currently working on agents to support problem solving and decision making for corporate mergers and acquisitions.

Andy Aaron is a research staff member at IBM T. J. Watson Research Center. He was on the speech team for the IBM Watson Jeopardy! match. Along with his work in speech synthesis and speech recognition, he has done sound design for feature films and produced and directed TV and films.

Shari Trewin is a research staff member at IBM T. J. Watson Research Center. Her current research interests include multimodal human-computer interaction and accessibility of computer systems. She is currently working on knowledge extraction from scientific literature and interaction designs for professionals working with this extracted knowledge.

Maryam Ashoori is a design researcher at IBM T. J. Watson Research Center. She has a passion for exploring the intersection between art and computer science. Her work has resulted in several novel applications for the Cognitive Environments Laboratory, including a "Zen Garden" for unwinding after a long day and a service for sparking creativity for inventors.

Jason Ellis is a research staff member at IBM T. J. Watson Research Center. His research interests include social computing and usability. He is currently working on collaborative user interfaces for cognitive systems.

Brian Gaucher is senior manager of the Cognitive Environments Laboratory at IBM T. J. Watson Research Center. He leads teams specializing in user experience design and physical infrastructure of cognitive computing environments. His work focuses on the creation of highly interactive physical spaces designed to improve decision making through always-on ambient intelligence.

Dario Gil is the vice president of science and technology for IBM Research. As director of the Symbiotic Cognitive Systems department, he brought together researchers in artificial intelligence, multiagent systems, robotics, machine vision, natural language processing, speech technologies, human-computer interaction, social computing, user experience, and interaction design to create symbiotic cognitive computing technology, services, and applications for business.