In P. Compton, R. Mizoguchi, H. Motoda, and T. Menzies (Eds.) *Proceedings of Pacific Knowledge Acquisition Workshop PKAW'96*. October 23–25, 1996. Coogee, Australia.
See revision: *Int. J. Human-Computer Studies* (1998) 49, 831-865.

Brahms: Simulating Practice for Work Systems Design

William J. Clancey
Patricia Sachs
Institute for Research on Learning
66 Willow Place, Menlo Park, CA 94028 USA
Maarten Sierhuis
Ron van Hoof
NYNEX Science and Technology, Inc.
400 Westchester Avenue, White Plains, NY 10604 USA

ABSTRACT

The overarching context of work consists of activities, in which people are actors within a "choreographed" social interaction. Goals and problems arise within this *conceptual, social context*, in which technical, product-oriented tasks and their associated methods and evaluation criteria are defined. Brahms is a simulation tool for modeling this interactive social context, represented as the activities or "practice" of located agents. Rather than modeling cognition in detail, Brahms models focus on *what people do* where, when, and with whom. This entails modeling *social knowledge*—what people know about each other's activities and capabilities, by which collaboration is possible. Rather than modeling problem solving as disembodied puzzle manipulation, Brahms models focus on *circumstantial, interactional influences* on how work actually gets done, especially how information is shared and how participation (and hence a problem solving method) is determined. Brahms is suitable for use in work systems design, instruction, implementing software agents, and as a workbench for studying social systems.

INTRODUCTION TO BRAHMS

Brahms is a multi-agent simulation framework for modeling work practice, incorporating state-of-the-art methods from artificial intelligence research and insights about work and learning from the social sciences. Brahms was developed for use in work systems design, instruction, and as a language for software agents:

- Brahms models consist of groups of agents with context-sensitive, interactive behaviors. Agents are located, mobile, and have knowledge and changing beliefs. Groups may define job functions, teams, people at a certain location, or people with certain knowledge and beliefs.
- Brahms enables modeling *activities* of people during the day—how people spend their time—emphasizing information processing, communication in different modalities (phone, fax, voice mail, face-to-face, databases), and location-specific interaction (meetings, chance conversations, teamwork). Thus, Brahms allows modeling *a community of practice*—a group of people who participate in some shared, choreographed interaction, usually involving collaboration between individuals with different roles and experience.

- Brahms combines the functional perspective of *business process models* (orders, organizations, roles, product flow) and the knowledge perspective of *cognitive process models* (transformation of representations, flow and storage of information, error detection and problem solving) with models of active objects (e.g. fax machines, workflow systems).
- Brahms models are designed to make social processes visible by incorporating *social knowledge*—what people know about each other, relevant to assigning jobs, getting assistance, and prioritizing work.
- Brahms models incorporate generic protocols and objects, such as computer terminals, phones, and fax machines, and how to engage in a face-to-face conversation.

Thus, Brahms models provide a holistic perspective on *how work gets done*, emphasizing informal, social, *circumstantial* practices (rather than policies or procedures), while incorporating standard "task flow" views and productivity statistics.

Brahms models may be used to:

- Redesign organizations, facilities, and procedures
- Design information processing technology (e.g., use and function of personal digital assistants)
- Develop an instructional system with role playing, including coaching and practice of unusual situations.
- Develop software ("intelligent") agents in Brahms's language or models of people to be used by such agents.
- Provide a researcher workbench for developing models of organization, problem solving, and learning in social systems.

Brahms is primarily designed to be a tool for learning how to design work systems, as well as a means of embodying what we learn over time. Brahms was originally developed to replace a conventional business process modeling tool and is directly based on Nynex's and IRL's experience in studies of work practice for the purpose of systems design (Corcoran, 1992).

Conventional tools can't be used to easily or directly represent: customer reactions, troubleshooting, social conflicts, learning, and multi-tasking. These concepts are essential for understanding why people did what they did and why they weren't changing their practice spontaneously. Tools based on order flow alone or formal policies and organizations ("what should be") are of minimal value for communicating to workers and managers alike the work that needed to be done to change the system (Figure 1).

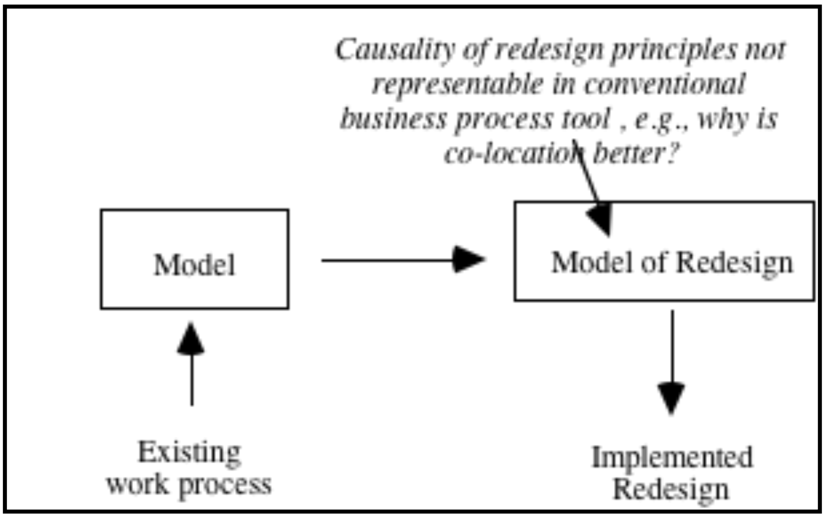

Figure 1. Business process modeling tools can be used to model the existing work process and the redesigned system. But redesign principles based on informal relationships, multitasking, and collaborative communication cannot be depicted.

When using conventional flow models, workers and managers alike become stuck in the old paradigm of job functions, cost, and time. This paradigm, so much part of the re-engineering mentality, tries to impose a manufacturing, assembly-line view of work (Davenport, 1995). Coordination is viewed therefore as something like clockwork, and work is viewed as a complex procedure. The detailed improvisations, informal assistance, and dynamic reorganizations of activities are left out. Without social concepts for improving the process, redesign would be viewed narrowly in terms of productivity measurements for an idealized system. In reality, the issue is how people can detect and cope with situations that don't fit the idealized description of how work is supposed to occur. Ethnographic study of the Nynex's work practices showed that many failures to satisfy customer needs could be tracked directly to social issues: understanding of other people, ownership, satisfaction in work, cooperation, respect, and personal needs.

Brahms is intended to be usable by workers and managers to facilitate participatory design (Greenbaum and Kyng, 1991), as well as computer scientists and social scientists who routinely study and formalize work practices. Brahms models provide a means of capturing and reproducing recurrent patterns of good designs and insights, including both social systems and formal work process. In particular, Brahms provides a *unique means for social scientists to express and present their observations in coherent formal models*, with reusable components.

Multiagent simulations of work

Many other simulation tools enable modeling organizational command and coordination policies in a geographically distributed environment. For example, Cohen, et al.'s (1989) Phoenix simulation models coordination of fire-fighting teams. Hayes-Roth, et al.'s (1995) simulations allow for improvisation in games played by the agents. Tambe, et al. (1995) describe a tool that models social interactions such as briefing sessions before military missions. In general, such multiagent simulation tools have the following characteristics:

- *Agents* are modeled as active, controlling the work flow by their beliefs, decisions & actions, not as resources passively assigned to work; agent behavior is asynchronous. Each agent is a "knowledge-based system" that monitors and controls its own behavior.

- That is, the model is based on *located, interactive behaviors of objects and agents,* not just abstracted (disembodied) process of work flow.
- *Attention* of actors at each moment in time is modeled explicitly.
- The *environment* is explicitly modeled, with its own objects, states & causal processes.
- In ecological models (e.g., SimLife) *all interactions are transactions* (because environment triggering actions is itself changing and being changed by actions).
- *Communications* between agents are modeled, which change beliefs and trigger new actions.
- Reciprocal information requirements between concurrent activities are modeled.
- *Movements* of agents are modeled.
- Agent behavior is modeled as a *problem-space (task) hierarchy*
- An *organization* model indicates work capabilities and controlling privileges (e.g., task assignments)
- *Inferences* (decision making) of agents are modeled by production rules.
- *Throughput of the work group emerges through interactive behaviors* of all agents, not modeled as equations based on numeric parameters describing the group (e.g., "amount of knowledge").

Many simulations focus specifically on modeling business "enterprises," complex coordination between agents playing different roles. For example, Levitt, et al.'s (1995) VDT models both inefficient behavior as well as idealized "intelligent" behavior. Burstein, et al. (1993) describe a tool for designing coordination strategies for efficiency at peak loads, using "coordination structures" as templates for creating models, for example, CONTRACTED-SERVICE; ADMINISTRATOR-WITH-MULTIPLE(SIMILAR)-PROCESSING-AGENTS. Dozens of programs focus specifically on modeling animal, computational, and even early human societies (e.g., see Gustavsson (1993); Gilbert and Doran (1993)).

Most existing work modeling tools and design methodologies may be viewed according to three orthogonal dimensions:

1. Hardware/Software Integration

These are tools for integrating special hardware for document databases, including methods for imaging and storing correspondence, networking computers from different vendors (e.g., Intel, DEC, Macintosh), and basic programming techniques for integrating software (e.g., integrating e-mail with spreadsheet programs).

2. Representation of Work, People, and Technology

There are two types of tools that incorporate models of work:

- *Workflow model*

  A workflow model describes the actual flow of information and representational materials (e.g., forms) through an organization. On this basis, a customized tool may be developed that integrates: on-line data, forms, manuals, and e-mail.

- *Business process model*

  A business process model describes organization and functional transformations at a high level. Business process models are often framed in terms of role relationships of people, tasks, and information:

—customer-supplier

—process-activity-(role-actor)

—conversations for action (request-negotiate-perform-close)

These systems have the following traits:

- They emphasize event-triggered interdependencies, e.g., when an order arrives on a fax, a person in a given role processes the order.
- They de-emphasize or ignore work practice, including ambiguity, exceptions, change, visuals, creative improvisation, redundancy, "out louds"
- They often separate people from business processes. For example, some business process modeling tools don't provide constructs for modeling who does what.
- These models are more suitable for clerical forms processing (in which kinds of information used and roles are relatively stable), than for design and planning tasks (in which the information and team are dynamically constructed).

3. Modeling and change methodology

Many commercial tools provide means for modeling an organization and managing change. These include:

- data analysis tools: time, cost, value measures
- representing multiple views: worker, analyst, manager, programmer, etc.

In summary, when designing Brahms, we found that vendors of work systems design tools are *preoccupied by the problem of integrating hardware and software*, but they are beginning to provide modeling tools with languages for describing the workplace. These languages vary greatly; few show sensitivity to the nuances of work practice—they emphasize business functions (e.g., accounting, sales), data, and documents, rather than people, roles, and organization.[1]

Such *work flow models* poorly describe *activities*. Generally modelers represent the idealized procedures of standard (textbook) policies and plans, rather than what people actually do. For example, a typical sales model nowhere says, "Now the salesperson takes the client out for lunch." In part, these early efforts are directed at simply constructing *some* model of work; idealized content offers at least a clean, already formalized place to begin.

In general, the notion of "social" in these tools is quite impoverished. For example, one tool models social behaviors in terms of "decreasing information-processing capability" (e.g., emotional responses) and deceit (Carley, et al., 1993, p. 3). Malone and Crowston (1991) define coordination as the "act of working together" but apply the conventional management perspective. Their models do not capture practice, but instead descriptively *abstract* coordination in terms of bidding and communicating interdependencies. Indeed, they view coordination as "the additional information processing performed when multiple, connected actors pursue goals that a single actor pursuing the same goals would not perform." That is, coordination is the *overhead* required when you can't do everything yourself!

Strikingly, the *cognitive, as well as the social, aspects of work are omitted* in most current modeling methods. Most examples provided by vendors suggest that work is clerical, involving purely mechanical manipulation of materials, with no conceptualization or meaning construction. However, some tools are intended to support strategic planning.

[1]For a recent comparative review, see (Tyo, 1995).

How is Brahms different from other current "distributed AI" or agent simulations?

In Brahms one models what agents do during a day, *activities,* not just tasks. Activities model *scoped focus of attention*, such that different "concerns" may be active at one time. Rather than viewing activities as procedure calls, every activity on a current hierarchical path remains active—just as a person is simultaneously conversing with a colleague, attending a meeting, on a business trip, representing an employer, pursuing a career, being a citizen, etc. Each level of activity establishes the *context for what is noticed in the environment, how beliefs are modified, how work is prioritized, and how communication occurs*. Activities define what problems need to be solved; goals and tasks arise in the context of activities.

For example, in Brahms models one might represent that what an agent does between 9 and 10am is quite different from the habitual practice at 4:30pm—even though the agent is doing the same "job." Similarly, a worker might engage in the activity of a coffee meeting, at which time a supervisor hands out the days' job assignments and workers bring up problems that occurred the previous day. Thus, problems are articulated and resolved in the context of activities. This context establishes what information is conveyed by whom, who participates in problem solving, and what actions are available. Activities, unlike tasks, do not have well-defined goals, whose accomplishment by definition terminates the task (e.g., "being a computer scientist" has no well-defined termination condition at which point we could say that the objective had been reached).

Brahms models therefore combine components not usually found together in a single simulation: movement of agents, communication, agents interacting with nearby agents (e.g., going to see someone at his or her desk), and agents interacting with physical objects. For example, the VDT tool incorporates phone call communication, but a phone call is something appearing in the in-box that expires in one minute if it is not attended to. In Brahms we actually model hearing the phone and, for example, deciding whether to answer the phone at a neighboring desk. Almost every other model is too idealized, forgetting that such circumstantial interactions occur and play a pivotal role in everyday life. Brahms is specifically designed to *model collaboration in practice*, that is, at the level of how actions of two or more agents become synchronized in time and place, plus the *contingent* communications and efficiencies that develop from circumstantial interactions.

A model of practice emphasizes:

- *what people actually do,* not just an official job function
- *what an agent is doing at every minute* (where they are, what they are perceiving) not just "working on a task"—allows capturing incidental, informal exchanges and the interplay of location, representations, communication.
- *daily actions* (e.g., breaks, reading mail, working during lunch hour).
- *informal information processing, circumstantial encounters, redundancy, fatigue, multitasking*
- *collaboration as an choreographed event (requiring a handshake protocol) between two or more agents* (e.g., a face-to-face conversation during a meeting, a conference call)—communication is not a stochastic event, but directly modeled as a multiple agent, synchronized, located interaction.
- *people are not interchangeable resources*, but have a personal identity:
    - behavior is sensitive to time of day;

- activity framework allows *multitasking, interrupt and resume;*
- practice determines *how jobs get assigned and why they get done when and how they do in practice* (circumstantial influences such as interruptions, meeting someone, phone call, traffic, teaming, carryover of work from day-to-day).

Although other tools emphasize organizational design, most ignore that agents belong to multiple, overlapping groups (not merely a job function hierarchy), such that each group has interests, capabilities, and behaviors that members of the group "inherit."

Other key characteristics of an activity-based model of work practice include:

- *use of representational artifacts* (e.g., fax, databases) that agents create, write, and read.
- collaboration only occurs through *shared perceptual circumstances and direct communications* (e.g., starting a conversation because you can see someone is copresent and invite them to talk).
- *actions stem from beliefs in a context,* involving subactivities, detected facts about the world, and new belief consequences (not just belief -> new belief or belief -> action, but rather: belief -> activity (detectables) -> new beliefs.
- all beliefs come from observation, being told, or inferential chaining ("thoughtframes").
- actions are *physical changes to the world,* including changes to representation, moving, and speaking to someone.

How is Brahms related to knowledge acquisition?

Most knowledge acquisition efforts are focused on problem solving: The reasoning involved in planning, designing, diagnosing, controlling, etc. some system in the world. Such reasoning requires data which is gathered from instruments by asking other agents. Models of practice place reasoning in context: *Brahms models agent behaviors rather than only inferences*. Nevertheless, standard knowledge acquisition techniques are useful: Observation, interviews, scenario definition, and case analysis.

A simple example is illustrative. In developing a medical diagnostic expert system, one might ask the physician what kinds of patients he or she sees, what information is used, etc. In developing models of practice, one would start with questions like: Where do you work? What time do you start working each day? What do you do first? Is your schedule different during the week? Whom do you work with? How do you communicate with other people and what kinds of conversations do you have?

A typical Brahms model captures a day in the life of some main character in a community of practice, or perhaps some key collaborative event in development of a work product. To make the modeling manageable, one thinks in terms of writing a play: There is a stage, main characters, a point of view, and probably some climax scene. For example, in our most elaborate model, the climax is a three-way conference call by which a "turf coordinator" brings together a service technician at a customer site and another craftsman in the central office.

Models of practice include the lowest level tasks of a corresponding business process model, but omit the cognitive modeling level of reasoning and calculation one might find in a typical expert system. For example, we might model a person as filling out a form, but not indicate the inferences necessary to do this. The information required to fill out the form and the changes to the form are only modeled to the extent necessary to represent what triggers or modifies another person’s activities. Especially, we model information and tasks that might be in error or produce error. But the distinction is subtle: A Brahms model might represent the specific inferences by which a supervisor prioritizes the day’s work and assigns jobs to particular workers. Unlike in an expert system, the simulation of this “scheduling task” might include an interruption by a co-worker, an inability to log onto the computer database, running out of time, copying information to paper, etc.

In summary, Brahms models are not as detailed as models of cognitive skills, nor are they as general as functional models of business processes. They do not describe just what people are supposed to accomplish (functional transformations of materials), nor do they describe the intricate details of reasoning or calculation.

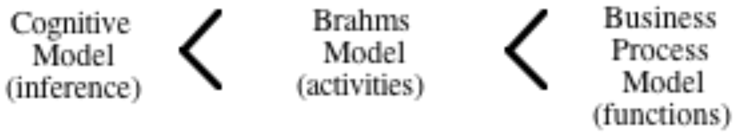

Figure 2. Relation of Brahms to other models of work.

Brahms models not just reasoning, but effectively an agent’s *situation-action knowledge within a social choreography*:

- checking up on status of a worker,
- knowing how to reach someone, to get help, to get required information,
- conveying information to about your activities to others,
- routine behaviors as well as serendipitous interactions with other agents,
- how to set up teams to satisfy customers and promote learning.

What are some typical settings for using Brahms?

Our largest model, serving as a test bed, models the activities of the “turf coordinator” for T.1 provisioning at Nynex, incorporating meetings, job assignments, a computer database for logging work, conference calls, and agent work at remote locations[2]. A second example models the work of a group in Nynex’s “enterprise” business. A third model on paper represents the physical flow of patients, physicians, and nurses through a healthcare clinic during a patient visit.

Here are some examples of design problems for which our experience indicates Brahms would be useful:

- Design workflow and information-processing technology in a healthcare clinic.
- Develop personal digital assistants (PDAs) to be used on the floor of a manufacturing plant or by a distributed work group.
- Design a new “coordinator” role for reconciling interests and priorities of different crafts in customer services (e.g., the role of coordinating work at the central offices and customer locations in provisioning new telecommunications services).

[2] A T.1 is a high-capacity circuit that is often used for point-to-point communication between business customer locations and telephone central offices. Provisioning is the process of accepting customer orders and installing and testing such circuits.

- Determine tradeoff in face-to-face meetings, multiple line phones, voice mail, and e-mail in a highly distributed work group (e.g., airlines personnel at the curb, at the check-in counters, at the gates, and servicing planes).
- Model different means by which customers find out and use information about financial or rental services (e.g., internet, agents, phone inquiry, local business office, friends) at home and on the road.
- Find ways to relate learning on the job to small team assignments (e.g., pairing novices and experts); model how practices are learned by co-location to better understand how direct interaction between people is complemented by instructional programs at the desktop.

Epistemological Stance

Brahms is effectively an application of cognitive modeling techniques to social processes. That is, the constructs of the representational language and engine are based directly on techniques used in multi-agent "knowledge-based" simulations. However, we interpret the constructs in a Brahms model in a way that is different from the interpretation of most "symbolic cognitive models." The important differences from previous work in cognitive modeling include:

- Composed action sequences are viewed as *activities*, not merely tasks or procedures, but a *socially-organized conception* of how interactions normally occur with other people and things in the world.
- An agent's goals (intentions) are represented as the activities he/she is currently doing—a set of interactive behaviors in some setting—not merely static names for "desired end states." Activities are not "things" that need to be done, but *interactional processes*.
- Situation-action rules ("workframes") are conceived as *observer descriptions* of patterns of an agent's behavior, not things literally stored in the head of the person being modeled.
- Pre- and post-conditions of workframes are either *observable* states of the world or *conscious claims* an agent makes about another agent's beliefs, not arbitrary, "objective" descriptions ("a view from nowhere").

These epistemological claims relate a Brahms model to the knowledge of the people being modeled. We say that we are modeling what people *do*, rather than what they know. We focus on a work group's choreography, not just an individual's reasoning. Brahms models thus embody our theories of work practice and knowledge:

- We emphasize how individual capacity (such as the ability to detect and resolve trouble) is a *product of individual habits and environmental processes* (including especially interpersonal relations). That is, we don't locate control in either the head or in the world.
- We emphasize that observed patterns of activity are emergent, *not scripts* that are shared (literally known) by everyone in a group. For example, there is no such thing as one restaurant script that everyone knows. There are waiter activities, chief activities, cashier activities, patron activities, etc. If these activities are properly choreographed, it appears as if the team knows what the other agents are doing. But in fact much of what other agents do is invisible to each agent, and proper coordination is the result of local interactions, not global knowledge of "the script." Thus knowledge is situated, subjective (with respect to a social role)

and inherently interactional, not a view from nowhere. Many descriptions of knowledge in AI models capture what the group accomplishes (product-oriented descriptions of group behavior), a kind of global scientific perspective, not what individuals know.

Most generally, we are pursuing the hypothesis that *relationships between people and their environment partly explain how organizations change*. Over the long term we want to model organizational knowledge and learning based on contingent, local encounters.

In summary, Brahms is specifically designed for modeling the reactive nature of human behavior. The language allows modeling how people dynamically shift their attention, interrupt and resume activities, and compose activities (having multiple interests at once).

Note that Brahms models do not represent the following:

- Agents' *models of their history and trends of their group*: history of the group, competitive pressures, management's initiatives, changes in customers.
- *Actual language* used by agents when communicating (e.g., how social conversations become task oriented)
- *Reconceptualization* (learning on the job) influencing later priorities, attitudes, judgments in handling difficult situations
- *Learning* by watching others or being told how to do something.
- *Cumulative effects of work flow*, especially interruptions and waiting (also: forgetting, variety, rhythm, fatigue, anxiety, exuberance).
- *Complex juggling and simultaneity of activities* to ensure closure, to be productive (e.g., reading while on the phone).
- *Life away from work*: breaks, vacations, family.

Representation Language Details

The most central representational unit in Brahms is called a *workframe* (Figure 3), a situation-action rule consisting of preconditions (what the agent must believe to be true), actions, detectables (what facts in the world might be noticed, with what probability and when during the actions), and consequences (changes to the world or this agent's beliefs that result). Workframes are organized hierarchically into activities. Actions in a workframe may be primitive (just indicating a name, duration, and priority) or composite (another activity). Primitive actions also include movement to another location and communication (described below). Consequences and actions are ordered and interleaved. Detectables may be indicated as "impasses" that interrupt the workframe or as "end conditions" that end the workframe or its encompassing activity.

Workframes are inherited by agents from all groups to which they belong; groups may belong to other groups. Priorities allow workframes to interrupt each other or carry out specific aspects of a more general protocol. For example, workframes at the "all groups" (top) level specify how to use a telephone and have face-to-face conversations; these have intermediate priority. Workframes that trigger conversations are most specific and have the lowest priority. Workframes that specify what to say during certain kinds of conversations have the highest priority. By this simple scheme, it is possible for one agent to initiate a conversation and for the responder to "remember" something he wanted to tell the first agent when he called; thus a give and take may ensue.

Thoughtframes model agent reasoning about implications of beliefs, leading to changes in what they do next (thus a distinction is draw between “action rules” and “thinking rules”) Thoughtframes take no time.

Changes to beliefs may occur by virtue of: broadcast (e.g., speaking outloud), transfer from agent (telling or asking), transfer from object (e.g., reading a database or a fax), detectables, and consequences.

Activities are spatially-dependent:

- *location goals* cause agent to move to location when workframe is enabled (e.g., “Move to location X.”)
- *location preconditions* depend on agent location (e.g., “Is the current agent at location X?”)

Workframe: WIRE-T.1-END-SECTION

Pre-conditions:
located at customer floor
have wire
have jack
have tools
customer is aware of your presence

Action: wire end section & install jack

Detectables:
jack wrong or broken (probability 10%)

Consequences:
jack installed (fact & belief)
end section wired (fact & belief)
need to talk to TC about overall test (belief)

Figure 3. Example of a workframe, written informally.

Objects embody stored information about the world, modeled as the “beliefs” of the object (e.g., a database). *Factframes* models object behavior, including what they detect and how they change state. Object instances may be created by an action (e.g., fax transmission creates a paper copy at the receiving station).

Facts are an eagle-eye view-from-nowhere—the outsider's view of the simulation, for example, the state of telephones, location of agents, etc. Detectables specify what facts an agent might detect during the action of a workframe. Beliefs are propositions agents believe about objects (state of the world) or other agents.

A communication may involve asking or telling. A communication may be from an agent or object to a specific agent or object, a group of agents, a class of objects, or may be broadcast. For example, a factframe for the fax object broadcasts to every agent within proximity that a fax has arrived.

Brahms currently models geography in a rudimentary way, consisting of regions, buildings, and their connections. Duration of movement is simply proportional to distance; for convenience movement between non-connected locations takes no time.

In general, descriptions of activities are associated with groups. In practice, there may only be one member of a group in a given workplace (e.g., one "physician's assistant" in a medical care module) or roles may be highly differentiated (e.g., the role of the "physician in charge"). Depending on the purpose for building the model, models may represent:

- particular people (Dr. Axelrod in Redwood City),
- types of people ("an HMO physician at CareGood"), or
- pastiches ("a typical nurse, patterned after Mr. Reno at San Joaquin Valley").

Agents that are not central to the work being modeled may be modeled as an individual representing a group. For example, an aggregate "customer" for a workgroup could generate orders.

EXAMPLE: PRACTICE IN A HEALTH CARE MODULE

As an example, I will sketch a model of the work practice in a module consisting of about ten people in one corner of an outpatient clinic. The model describes the activities of people in this module, including their communications with other modules and the hospital. The *primary activity* we will model is the patient visit. Thus, the patient is the main character and the module is the stage. The climax scene is the physician's interaction with the patient in the exam room.

This particular module was chosen to test an electronic medical record (EMR) system for a large HMO. An ethnographic pre-study of several months was followed by a week of videotaping routine work in the outpatient setting. Our overall interest was in promoting participatory design of the EMR, plus understanding the difficulties and benefits of transitioning from a paper chart to an EMR.

We had available the following information for constructing this model:

- floor plan (blueprints)
- flow chart describing typical work process
- video tapes of 24 patient visits, each filmed at four locations having wide-angle and narrow views.
- ethnographers' field notes from several months of pre-study
- hour-long, transcribed interviews of members of the module, focusing on their experience with computers and potential uses of an EMR
- matrices comparing backgrounds and work styles of the providers
- sample forms and format description of a medical chart

Using Brahms, one might start by describing the groups (Figure 4).

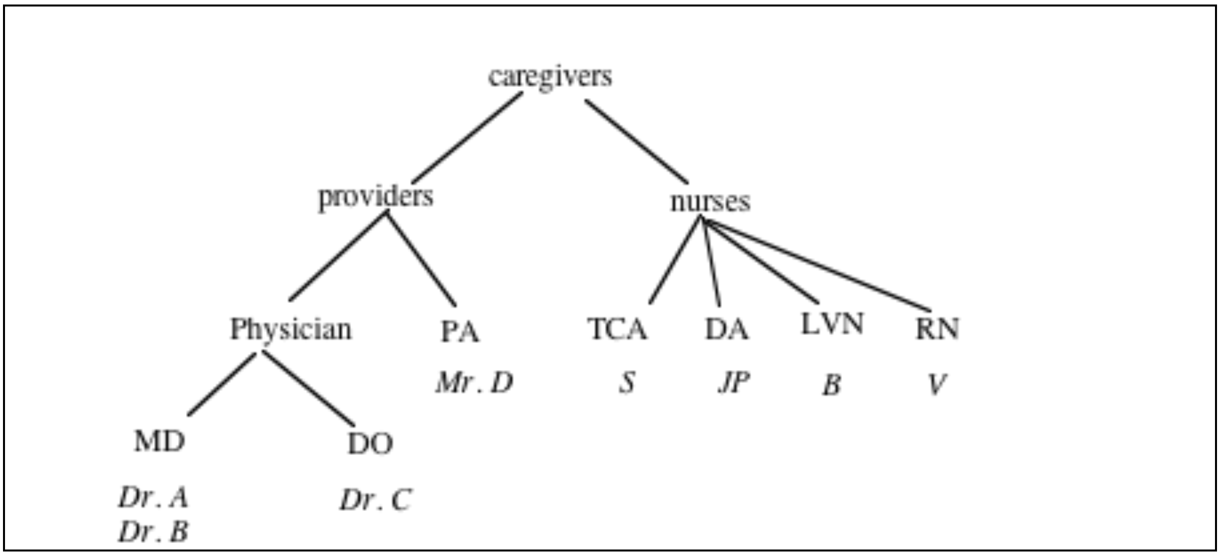

Figure 4. Groups for a health care module[3]

(Individuals who belong to each group are listed in italic.)

The most important representational artifact at this work site is *the chart*. There are also two computer systems to be modeled, one for patient scheduling and other for patient data. The chart is modeled in terms of sections and what information is written in each section (e.g., Current progress note, new orders, lab and pathology results, back page). Other forms used by the caregivers and patients include: Pathology lab order, labels, and consent forms. Details about content and use of forms and computer systems are included in the model only as they arise in the description of activities. Other representational artifacts include messages from the appointment center in the clinic, a prescription refill (such as a fax from a pharmacy), phone messages from patients, and notes from physicians to nurses to call patients. Each of these is represented insofar as it is read, written, or hand-carried during a patient visit.

Early on, one describes the "geography" of activities, including the following locations:

- the clinic
    - the module
        - the waiting room
        - the nurse processing station (central nursing station)
        - the nursing station (4)
        - the exam room (4)
        - the provider's office (4)

[3]TCA = Trained Clinical Assistant, not a nurse technically.

Numbers in parentheses indicate how many instances of this kind of location appear in the module. Other locations that communicate with this location include the registration desk, the chart room, and the hospital.

The primary activity of the module is the patient visit. When not engaged in this activity, participants are engaged in the activity of "being in the module" and what they do may be omitted.

In scoping the model, we decided to distinguish the roles of the nurses and the capabilities of the providers (MD vs. physician assistant). We also wanted to model the *mentoring activity* between a physician (Dr. A) and a PA (Mr. D) (which occurs as a scheduled meeting).

The primary actors during the patient visit are: the patient, the registration clerk, the nurse (assigned to a provider for the day), the provider (MD or PA), and assisting nurse.

The main activities of the patient visit are, in order of occurrence (included optional activities are indicated by brackets):

- patient checks in for appointment (patient, registration clerk)
- patient arrives at the module (patient, nurse)
- initial interview by nurse (patient, nurse)
  —{pre-exam lab/pathology tests (patient, nurse)}
- examination by physician (patient, provider)
  —{drug administration (patient, nurse)}
- processing orders (patient, nurse)
  —{post-exam lab/pathology tests (patient, nurse)}
- patient checks out at registration (patient, registration clerk)

The *flow of people* is as follows:

- The Patient moves from registration desk to waiting room to nursing station to exam room and back out in reverse order.
- The Nurse moves from nurse processing station to her nursing station to her provider's exam room to her provider's office and then back to the central nursing station.
- The Provider moves from his office to his exam room and back to his office.

The *movement of the chart* for a patient visit is pivotal in the choreography of the visit:

- At the beginning of the day, the charts for a provider are delivered to the *nursing station* of the nurse assigned to that provider on that day.
- At the beginning of the day, each nurse moves the provider's charts to the *nurse processing station*.
- After the patient has arrived and been checked in, the nurse places the chart on the desk in the *provider's office*.
- The provider takes the chart with him into the *exam room*.
- After the patient exam, the provider takes the chart to his *office*.
- When the provider has completed his notes, he puts the chart in his *out-basket*.
- The nurse retrieves the charts periodically and takes them to the *nurse processing station*, placing them in the *central bin*.

- At the end of the day, the charts are picked up and returned to the *chart room*.

The "basic rhythm" of the work site is described chronologically (this is the formal schedule; the team interacts informally as needed):

- The clinic opens at 7:15am to 7:30 for incoming calls; chart room is open.
- Module reception (waiting room) opens 8:15 to 8:30am.
- Nurses arrive 8 to 8:15, officially 8:30, to get and organize charts, organize same-day appointments (SDA), schedule the PA, who arrives at 8:30.
- The MDs arrive at 9 and begin seeing patients at 9am.
- Patients are scheduled in half-day shifts, from 9am to noon and 1:30-4:30 (approx), every 10 minutes. There may be double appointments.
- If "doing triage in morning," MD works through lunch; afternoon appointments in module are first 1:30, last at 3pm.
- Lunch begins for nurses at 12:15, but one covers until MDs are done.
- Each MD sees 10-14 patients during one half-day shift, 1 to 3 shifts per week.
- The PA sees patients 10 shifts per week. 7-8 shifts are SDA.

Note that providers have many other activities, including doing laboratory procedures, learning, mentoring at other locations, covering in the hospital ("team rounding"), evening emergency care, etc. This model only describes what providers do during a patient visit. That is, this is effectively a "model of the patient in the clinic" not a model of providers.

Similarly, we are aware of different ways nurses work together. For example, one nurse (RN or LVN) works with two providers on Tuesdays and Thursdays. Two or three other nurses (possibly including a TCA) work with three or four providers the other days of the week. The TCA prefers to work with Dr. A (the physician-in-charge), etc. Such interactional patterns illustrate part of what is meant by practice, as opposed to policy or procedures. We describe what people do together, how they interact with each other and objects in the environment.

In modeling the clinic, we focus on the nurses' first rule, "Keep the rooms filled." This orientations gives us the nurses' point of view, so we can expect what they might notice and helps us detect gaps in our model. This is an example of a choreography constraint, which describes what the group is trying to accomplish in their routine interactions. Accordingly, if a provider is absent on a given day, his exam rooms will be used for placing the third patient of another provider.

As an example, here is a sketch of the nurse's activity of coordinating patient visits, which occurs as a subactivity of "being a nurse in the module."[4] This sketch would be filled in by additional workframes that specify what the nurse does while waiting, etc. Or we could decide that because this model focuses on the patient, we will omit such details (corresponding to actors who leave the stage and return later).

COORDINATING PATIENT VISITS

  At the nurse processing station

  <u>If the light for provider $X appears on the console,</u>

    Subactivity:: The patient visit

[4]Key to notation: **NAMES OF WORKFRAMES IN ALLCAPS**; <u>Key preconditions underlined (beliefs determined by detectables and/or thoughtframes)</u>; **Subactivities are composite actions,** others are primities (not decomposed); **=>** indicates "go to location"; $X indicates variable binding to be referenced later; **Wait** indicates a detectable that causes an impasse until it changes state.

THE PATIENT VISIT

=> Outside the waiting room

Retrieve the CPR from the slot

=> At the nurse processing station

Locate the chart and assemble materials (5 mins)
Take the chart
Subactivity:: Pre-exam patient interview

THE PRE-EXAM PATIENT INTERVIEW

=> In the waiting room

Call the patient
Tell the patient to follow you

=> In the nursing station

Stamp the progress note with the member's and provider's cards
Record the patient's vital signs on the progress note
Place the CPR and member's card on the front of the chart
Tell the patient to follow you
Note the room assigned to this visit ($X)
Subactivity:: Bring together patient and provider
Wait for the exam to be complete

<u>When the provider indicates the exam is complete,</u>

Subactivity:: Post-exam patient visit

BRING TOGETHER PATIENT AND PROVIDER

=> In the examination room ($X)

Tell the patient to undress and stay put

=> In the provider's office

Write the number of the examination room ($X) on the progress note
Place the chart on the provider's desk

POST-EXAM PATIENT VISIT

=> In the nurse processing station

<u>If CPR indicates tests are ordered,</u>

Prepare lab requisitions (3 mins)
Instruct the patient about the tests (2 mins)
Tell the patient to follow you.
Subactivity:: Direct the patient to the lab
Wait for the patient to return

<u>If provider has requested a follow-up visit and date is less than 2 weeks,</u>

Call the appointment center
Handle consultation requests
Tell the patient to return to the registration desk.

DIRECT THE PATIENT TO THE LAB

=> In the laboratory

Tell the patient to return to the nursing station when done.

To this point, we have only considered the routine work. Within this participation framework, we will want to represent what can go wrong, who detects problems where and when, and what outside influences effect the rhythm of activity in this work system. For example, we might focus on:

- Time spent processing (and finding) paper records at multiple locations.
- Difficulty interpreting x-rays that are kept at the hospital.
- Need to call pharmacy for dosage information (patient is left waiting in the exam room).
- Nurses having to locate lab results for providers.
- Effect of nurses having to cover other modules ("floaters"); effect of union rules on number of floaters.
- Different practices in other modules.
- How previous working relationships between these people affect current work practices.

THE DESIGN STANCE IN BUSINESS ANTHROPOLOGY

A work systems design (WSD) effort presupposes what engineers call a "design stance." That is, rather than simply studying or describing a workplace, which has been the conventional approach in anthropology, we intend to use Brahms to change how work gets done. This means that we must have theories of *how different kinds of designs produce different results*, otherwise we would not be able to suggest alternative designs or reason about trade-offs. On the other hand, the very idea of participatory design means that WSD facilitators (e.g., anthropologists and model builders) need not (and could not) develop an optimal design and deliver it to clients. Instead, Brahms is conceived as a tool by which WSD facilitators in collaboration with workers and management will bring about change incrementally and iteratively. Nevertheless, using Brahms as a design tool presupposes that WSD facilitators have some understanding of how interactions between people, technology, and organizations relate to work practice.

For example, to show the benefits of alternative work system designs, we might model how beliefs about the group change because of a design. For instance, co-location facilitates face-to-face conversations, in which beliefs about the group are communicated. In contrast to communications via on-line systems, face-to-face conversations tend to include *beliefs about skills and experience*, not just beliefs about work status. We can show these relations graphically:

design of facilities, technology & organization
    => *communication* of beliefs about the group
        => activates workframes for *relating people to work* (e.g., assigning jobs, getting help, dividing up the work, teaming people)

Thus designs, beliefs, and work interact: Communications influence beliefs about other people's skills, which influences how work gets done. For instance, co-location (working in the same office) might influence what a coordinator knows about certain technicians, which influences how jobs are assigned. In a simple example lasting a few minutes of simulated time, we might observe all of the phenomena listed above.

From a cognitive modeling perspective, human behavior is explained in terms of the beliefs of an agent and work skills. Making social phenomena visible in the context of work systems design means showing *how beliefs and skills change as a result of a given design.* Although we don't currently model change in skills in Brahms, we can model change in beliefs about skills. To model change in belief, we include deductive inferences (i.e., thoughtframes) as in a cognitive model, but emphasize informal (circumstantial or serendipitous) communications in which beliefs are exchanged. In summary, modeling social phenomena means modeling how different work designs influence informal communications, and how the beliefs exchanged influence how the work gets done.

A simple way of stating our incipient understanding of work systems design is given by this causal relation:

design -> interactions -> workflow -> results

A given *design* (configuration of people, technology, and organization) causes certain everyday *interactions* in activities (both positive and negative, including antagonistic relationships and customer-focused conversations). These interactions are manifest in certain properties of the *workflow* (e.g., bottlenecks, early detection of errors, multiple error-prone hand-offs). Finally, these workflow properties are manifest in the *business results* management uses to assess productivity (e.g., average number of days to process an order).

As an example, consider the logic behind part of the redesign of the T.1 provisioning process (Figure 5).

Work System Design: co-location, turf coordinator, turf teams

—>Activity Interactions: improved communication (trading contextual information), collaborative planning & troubleshooting

—>Workflow Quality: bridges, hand-offs, no silos

—>Business Results: fewer errors, shortened interval (CD)

Figure 5: Relation of design to business results in T.1 provisioning

The design incorporated elements such as "co-location" and a turf coordinator. This combination of people, technology, and organization changed the interactions occurring during work activities to improve communication and promote collaborative planning. These changes in turn affected the workflow: Bridging front and back ends of an order and reduced hand-offs released work early, provided a single dispatch to the customer, made scheduling of jobs more efficient, etc. This in turn lowered the error rate and shortened the time required to turn up the circuit.

The example illustrates that reasoning about designs requires adopting different perspectives and arguing about how they are *causally related*. The advantage of the ethnographic perspective is that it introduces the level of "activity interactions," complementing the systems analysis perspective on workflow and business results. A whole range of social conception is thus introduced in terms of location of people, conversations, informal relationships, etc. that better explains how responsibility and attention arise and are sustained in a group. (Indeed, the advantage is to understand *motivation* and *concern* as conceived and developed with respect to a group.)

To repeat, the key perspectives that we relate when comparing and promoting change in work practice are:

- *The physical work system* of people, technology, and organizations (including information-processing tools, facilities, and formal job specifications—things that can be directly changed),
- *The emergent activities* of the groups involved in doing the work (including especially how they conceive of their roles and knowledge when planning and handling trouble),
- *The workflow* (including both the physical transformations of work artifacts such as T.1 circuits and the transformations of representations such as forms describing T.1 circuits), and
- *The business results* measured by management for assessing customer satisfaction, competitiveness, and productivity.

In short, when someone suggests a change in work practice in the context of a work systems design project, they must have, at least implicitly, an argument about how the redesign will change or sustain business results over some period of time. These causal arguments are rarely articulated in detail, and are instead often truncated by assumptions about the downstream effects. For example, a designer might have preformulated a set of "good interactions in practice," assuming that if these people-technology-organization interactions occur, the workflow and results will be acceptable or improved. Therefore, design conversations will focus only on how to bring about these interactions. But, as emphasized here, redesign requires some theory or experience that justifies the remainder of argument. If a manager on the steering committee of a redesign project is focused on the number of days to process an order, he or she will probably only find the discussion of activities to be meaningful if the relation to workflow and business results is made explicit.

A central problem in adopting a design stance based on ethnographic observation is that the concepts used to describe human culture (practice) have not traditionally been related to design. Unlike traditional ethnography, work systems design is fundamentally *evaluative*. Put another way, an observation is only interesting to a manager if it relates to business results. Adopting this design stance requires developing *a causal theory of how practice relates to workflow,* as shown in the above diagrams.

In general, ethnographic observation is required to identify situations that illustrate abstract design guidelines. For example, another change brought about in the T.1 redesign is that pressure and rancor was replaced by "shared end-to-end responsibility." How is pressure and rancor manifest in activities? In the workflow? How is "shared end-to-end responsibility" manifest in activities and the workflow? To make such abstract, but central concepts explicit, anecdotes are converted into cases, whose preconditions and consequences are modeled, just as in traditional knowledge engineering. In effect, expressing such conventional social concepts in a formal model brings to social science a change in practice that complements the change in business process modeling in adopting the social perspective.

In summary, a design stance requires a *vocabulary* for describing practice (e.g., functional hand-off, alerting, discussing context, lack of contact, developed relationships, rancor, cryptic notes, responsibility) and a *theory* of how changes in jobs and technology will transform one work practice into another. This theory is necessarily quite detailed, for it must explain how the practice will develop (e.g., how do we get from "lack of contact between people" to "developed relationships"?) and *how this changed practice will affect workflow and hence business results* (e.g., how do "developed relationships" improve the quality of the work?). This means that we must organize ethnographic characterizations to portray shifts from "dysfunctional" to "desired" practice and make explicit the causal relations to the analysis provided by other system designers who are focusing on workflow, quality, and productivity.

THEORETICAL IMPLICATIONS OF MODELING WORK PRACTICE

Designing work systems—including organization, technology, and facilities—requires understanding the *conceptual* context in which work occurs. For people, the overarching context of work consists of activities, in which people are social actors, that is agents with identities, a sense of place, and manner of interacting. Goals and problems arise and have meaning within such an *activity framework*, and it is with respect to this conceptual, social context that technical, product-oriented tasks (such as using instruments or manipulating representations) and their associated methods and evaluation criteria are defined.

Brahms is a simulation tool for modeling conceptual context of everyday work, represented as the activities or "practice" of located agents. Rather than modeling cognition in detail, Brahms models focus on what people do where and when with whom. This entails modeling *social knowledge*—what people know about each other's capabilities and areas of competence. Rather than modeling problem solving as an abstract puzzle manipulation, Brahms models focus on *circumstantial, interactional influences* on how work gets done, especially how information is shared and how participation (and hence problem solving method) is determined. Brahms is suitable for use in work systems design, instruction, implementing software agents, and as a workbench for studying social systems.

The demands of modeling work practice almost turn inside out the conventional view of knowledge engineering. Context is not something in the environment ("data") but partly conceptual and partly about other people. A social system is not just an organization, but a choreography of interaction, a set of practices for doing things in certain places at certain times. Knowledge is not just technical, but is about the group—*social* knowledge. What people know and do is organized around their roles as social actors, not plug-compatible problem solvers, but people with interests and ways of working together. Expertise includes knowing what other people know, how to get help, who is trustworthy and who is diplomatic, and how to team the patient, careful worker with the imaginative explorer.

In modeling work practice, standard AI issues of scheduling, planning, and information processing are not omitted, but are made *problematic*: How does a supervisor remember what everyone is doing during the day? How do members of a team at different locations coordinate their work day? How does informal, circumstantial encounters (such as conversations in a hallway) help align the expectations and understanding of the group about group's capabilities, how busy they are, and what they are becoming? To say that such issues are ignored in expert systems is an understatement. Indeed, social knowledge and interaction are ignored in almost all tools for designing computer systems. The most prevalent "work flow" and "business modeling" tools have the most narrow, indeed naive views of how work gets done. How can we design computer tools if we don't know what people need? That's the ultimate value of modeling work practice. In the collaboration between IRL and Nynex, this effort is only in its infancy. But the insights gained amply demonstrate that much is to be learned when cognitive and social scientists bring their insights together to create a holistic model of work.

ACKNOWLEDGMENTS

Brahms was conceived and developed by Bill Clancey and Dave Torok (formerly NYNEX S&T) in 1992-94. Our understanding of work systems design and social theories was developed through many long discussions with Gitti Jordan (IRL/Xerox-Parc), David Moore, Maddeline Ritter, and Jim Euchner at NYNEX. Technical designs were contributed by Erik Vinkhuyzen, Alwin Bakkenes, and Edgar Steenvoorden.

Brahms exists as a prototype developed in G2 on a SUN workstation. The system has a useful, but rudimentary interface, editor, and trace/debugging package. Current work includes comparative studies of tools and exploratory use on client projects. The name “Brahms” stands for “Business Redesign Agent-based Holistic Modeling System,” but it applies to any human activities.

The model of the health care module was developed with the cooperation and financial support of Southern California Kaiser Permanente. Judith Gregory provided the ethnographic data. Charlotte Linde (IRL) and Jean Gilbert (KP) were project managers.

Support for the development of Brahms has been provided by NYNEX Science and Technology, Inc.

REFERENCES

Corcoran, E. (1992). Building networks: New York Telephone rethinks how to regain lost customers. *Scientific American*, November, pp. 119-120.

Burstein, M.H., Ferguson, W., and Abrett, G. (1993). A Simulation Development Tool for Evaluating Coordination Strategies in Organizations. *AAAI Workshop on AI & Theories of Groups & Organizations: Conceptual & Empirical Research*.pp. 24-28.

Carley, D. Park, and M. Prietula. (1993). Agent honesty, Cooperation and Benevolence in an Artificial Organization. *AAAI Workshop on AI & Theories of Groups & Organizations: Conceptual & Empirical Research*. pp. 1-7.

Cohen, P. R., Greenberg, M. L., Hart, D. M., & Howe, A. E. (1989). Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine* **10**(3), 34-48.

Davenport, T.H. (1995). The fad that forgot people. *Fast Company*. November.

Gilbert, N. and Doran, J. (1993). *Simulating Societies: The computer simulation of social phenomena*. UCL Press.

Greenbaum, J., and Kyng, M. (1991). *Design at Work: Cooperative Design of Computer Systems*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Gustavsson, R. (1993). Societies of Computation: A framework. *AAAI '93 Workshop on AI & Theories of Groups & Organizations: Conceptual & Empirical Research*. pp. 96-102.

Hayes-Roth, B., Brownston, L., and Sincoff, E. (1995). Directed improvisation by computer characters. *Proceedings of the International Joint Conference on Artificial Intelligence*.

Levitt, R. E., Jin, Y., Oralkan, G.A., Kunz, J.C., and Christiansen, T.R. (1995). *Computational enterprise models: Toward analysis tools for designing organizations*. CIFE Working Paper, Stanford University, Department of Civil Engineering. February.

Malone, T.W. and Crowston, K. (1991). *Toward an interdisciplinary theory of coordination*. Technical Report CCS TR# 120, Center for Coordination Science, MIT.

Tambe, M, Johnson, W.L., Jones, R.M. Laird, J.E., Rosenbloom, P.S., and K.Schwamb. (1995). Intelligent Agents for Interactive Simulation Environments. *AI Magazine*, **16**(1):15-39,Spring.

Tyo, J. (1995). Simulation modeling tools. *Information Week*, 60-67. July 10, 1995.