Information Appliances: Relevant Artificial Intelligence Themes[1]

Reid G. Smith

Schlumberger-Doll Research
Old Quarry Road
Ridgefield, CT 06877-4108
USA

My intention is to concentrate on the potential utility of AI in the *information appliance* scenario. It is also my understanding that we are to discuss the interdependencies of the various technologies represented on the panel; that is, what each contributes to the whole, what each needs from the others, and what are barriers to progress. Hence, I have included some discussion of each of the technologies represented. I may have placed more power in the information appliance than originally intended in the strawman scenario... Nevertheless, here are some thoughts.

Artificial Intelligence:

It is apparent that the work on natural language communication - text and speech - could play a role, but I would like to concentrate on three other types of facilities that I believe will be central to the design of information appliances: *(i)* the evolving knowledge-based systems that act as *intelligent assistants* to smoothly merge machine capabilities with those of humans; *(ii) learning apprentice systems* that refine and extend their knowledge bases during normal use; *(iii)* knowledge representation languages that enable efficient and powerful encoding of a large variety of information.

Information/Data Management

An interesting trend these days is the increasing merge between work in DBMSs and in AI representation languages. Some people refer to the result as *knowledge management systems* or *expert database systems.* The AI representation languages bring a level of semantic data modeling complexity that is typically lacking from traditional DBMSs, while the DBMSs typically deal with much larger quantities of information, concurrent access, protection, and so on. I believe that systems of this sort will be central to the information appliance scenario.

Furthermore, smart access methods for these systems must be developed. This will be complicated by the fact that ideally we would like these methods to operate very close to the information storage media. The problem is that this may well place impossible loads on the *information utilities.* Resolution of this potential problem will depend on network access and bandwidth, as well as the local computing power of the appliances. Learning apprentice systems may prove helpful over time as personal information appliances develop models of their owner's modes of interaction, together with models of new and changing information utilities.

User Interface

We absolutely must have the ability to look at information in a variety of ways, including text, graphics, and voice, both interactively and for recording. This means that a variety of browsing

[1] *National Computer Conference, Panel on Trends in Computer Systems Technology*, Chicago, IL, USA, July 15, 1985. See pp. 10-11 of this document for the panel abstract, participants and a strawman "information appliance" scenario.

capabilities must be supported and that it must be possible for a user to easily define his own new methods. The point is that when we have a large quantity of information to manage, we need a lot of power to enable us to comprehend it. (We see this problem of complexity management all the time in construction of large software systems.) My experience has been that object-oriented programming (together with procedures and rules) offers a good deal of assistance in building such facilities.

From a low-level software point of view, the normal helpful utilities available on high-performance workstations (like Interlisp machines) must become normal if these appliances are to be usable for a large segment of the population (e.g., spelling correction and escape completion). In addition, the not-so-mundane things are also important, such as history maintenance, user modeling as an assist to reformulate queries, and cooperative interactions. (User-friendly means more than menus...)

The ability to personalize via reconfiguration, specialization, and extension of the appliance software is a key requirement. Knowledge-based system techniques may be useful here.

Workstation, Terminal

First of all, I assume that powerful local processors with large quantities of memory will be the norm.

Fast integrated text, graphics, and voice are a must. The resolution and speed offered by today's bitmap graphics workstations provide a lower bound to desired performance. Low-cost, high-quality hardcopy is another obvious necessity. A challenge is to construct systems that effectively orchestrate multi-mode interactions (this is already difficult enough with mouse and keyboard). Finally, better keyboards will surely someday become normal, in contrast to the sorry state we are in today. (As I sit here at my Mac I am especially sensitive to this!)

I would also like to see the information appliance environment as a natural extension to my every day programmer's environment. This might indeed simplify development of appliance facilities. I guess I am simply extolling the virtues of powerful integrated environments - both from the perspective of the system developer and user.

Networking

Appropriate protocols to support integrated text, graphics, and voice are clearly required.
Network bandwidth and access times will strongly affect the design and utility of information appliances. We get very used to LAN bandwidths (e.g., 10 mbps) and depressed when forced to return to more traditional numbers (e.g., 1200-9600 bps). How much computing is done locally (in response, say, to a query), based on information simply retrieved from an information utility, will depend on access time. If transmission is expensive and slow, then I assume more computing will have to be done at the utility, thus placing potential large loads on computing resources.

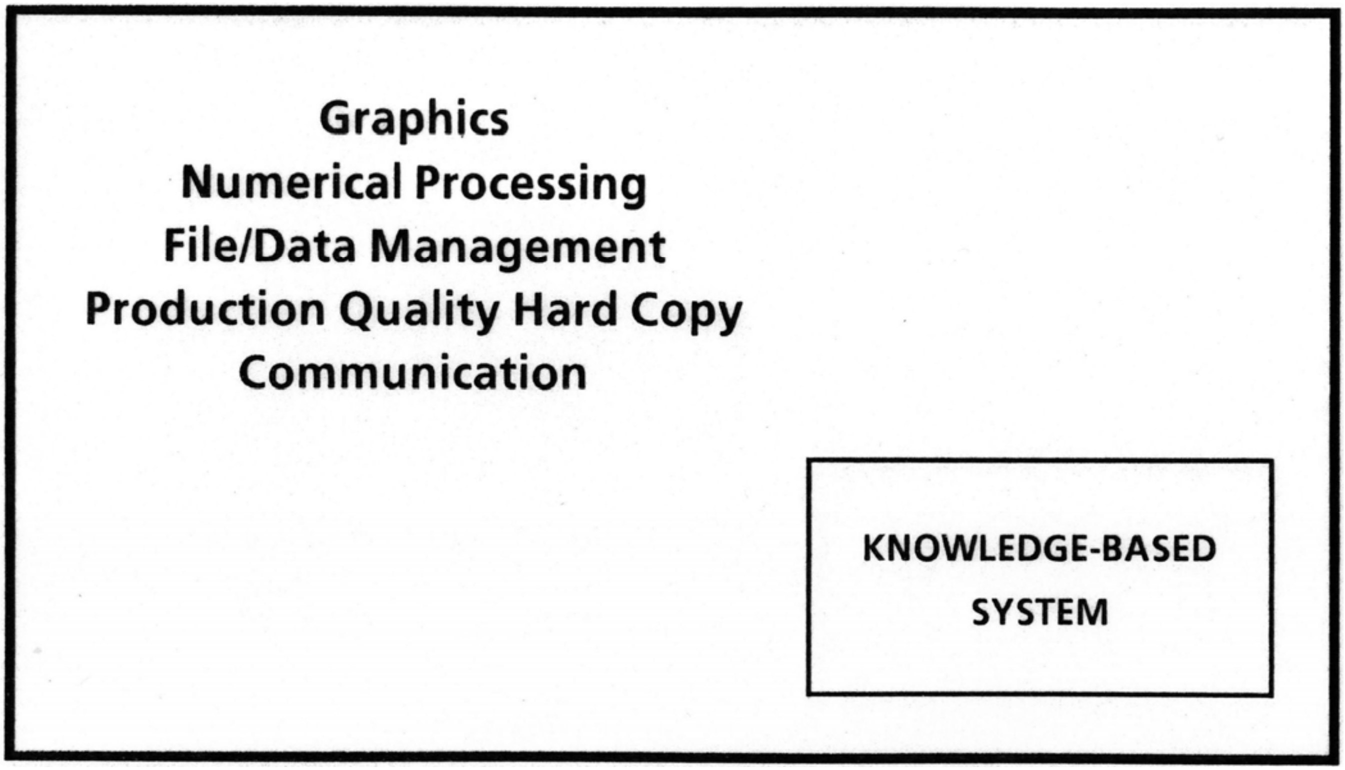
Some of the AI work on distributed problem solving may be relevant here. We may not be able to assume that the appropriate information utility will always be known by an individual information appliance. In such a case, some negotiation to find it may be required.

A small point: During a single information retrieval session, a user may need to have open several connections with a variety of information utilities. The local networking software should support this.

INFORMATION APPLIANCES:
Relevant Artificial Intelligence Themes

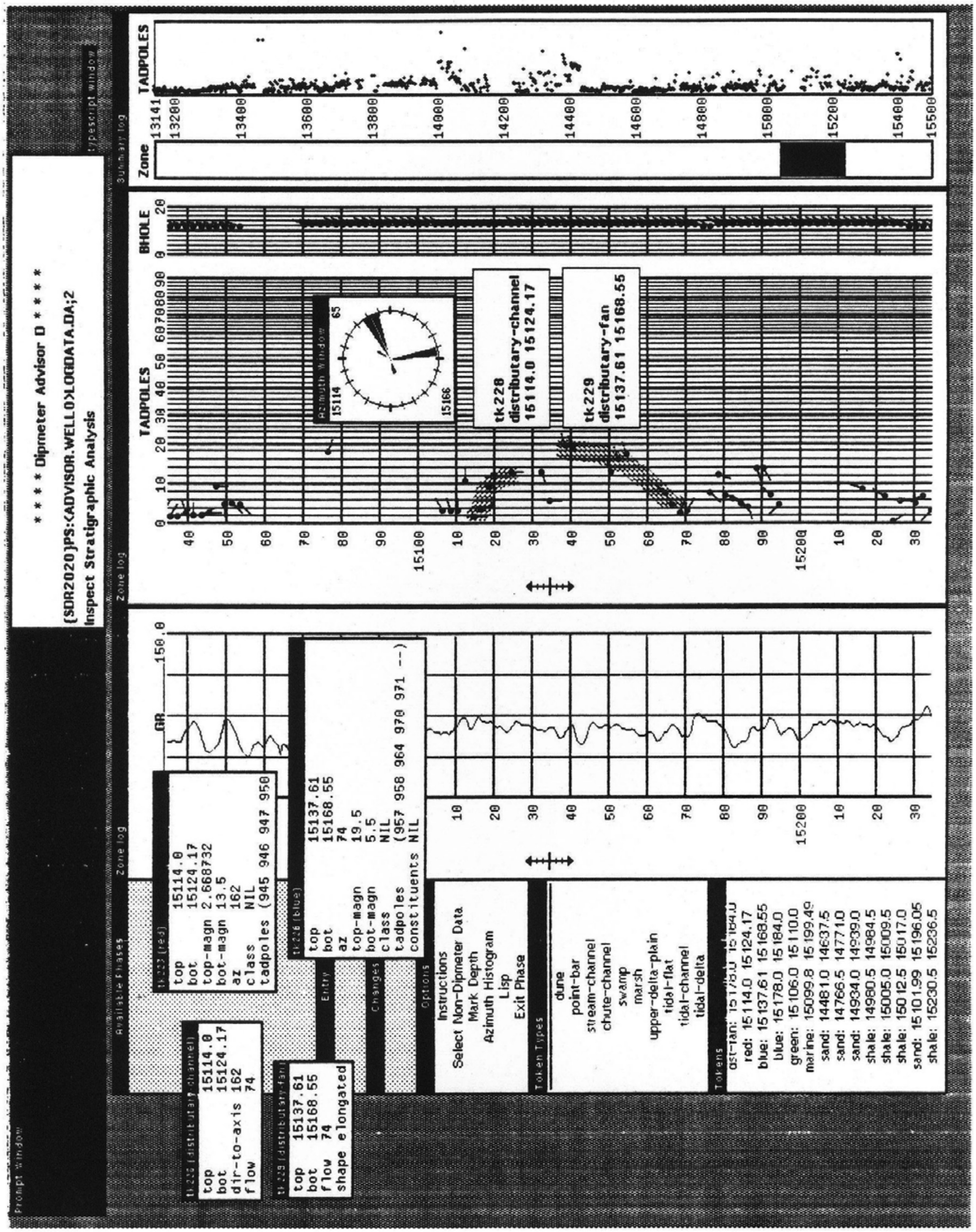
- *Natural Language Communication*
- Intelligent Assistants
- Learning Apprentice Systems
- Knowledge Representation Languages

Embedding a Knowledge-Based System: An Intelligent Assistant

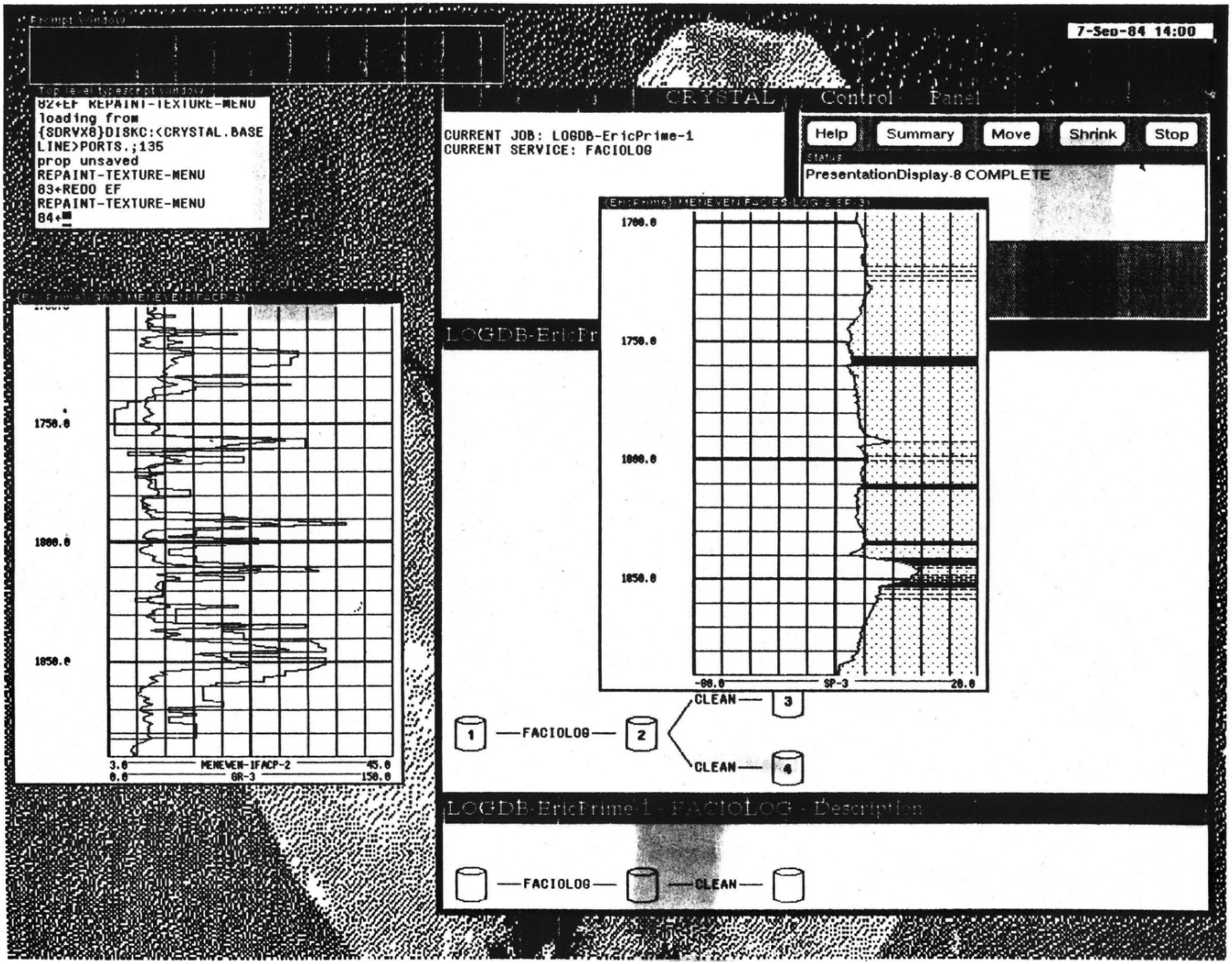


A user gets a number of advantages from using the system – one of which is symbolic inference.

In watching the system operate, an observer might never realize that any intelligence is involved.



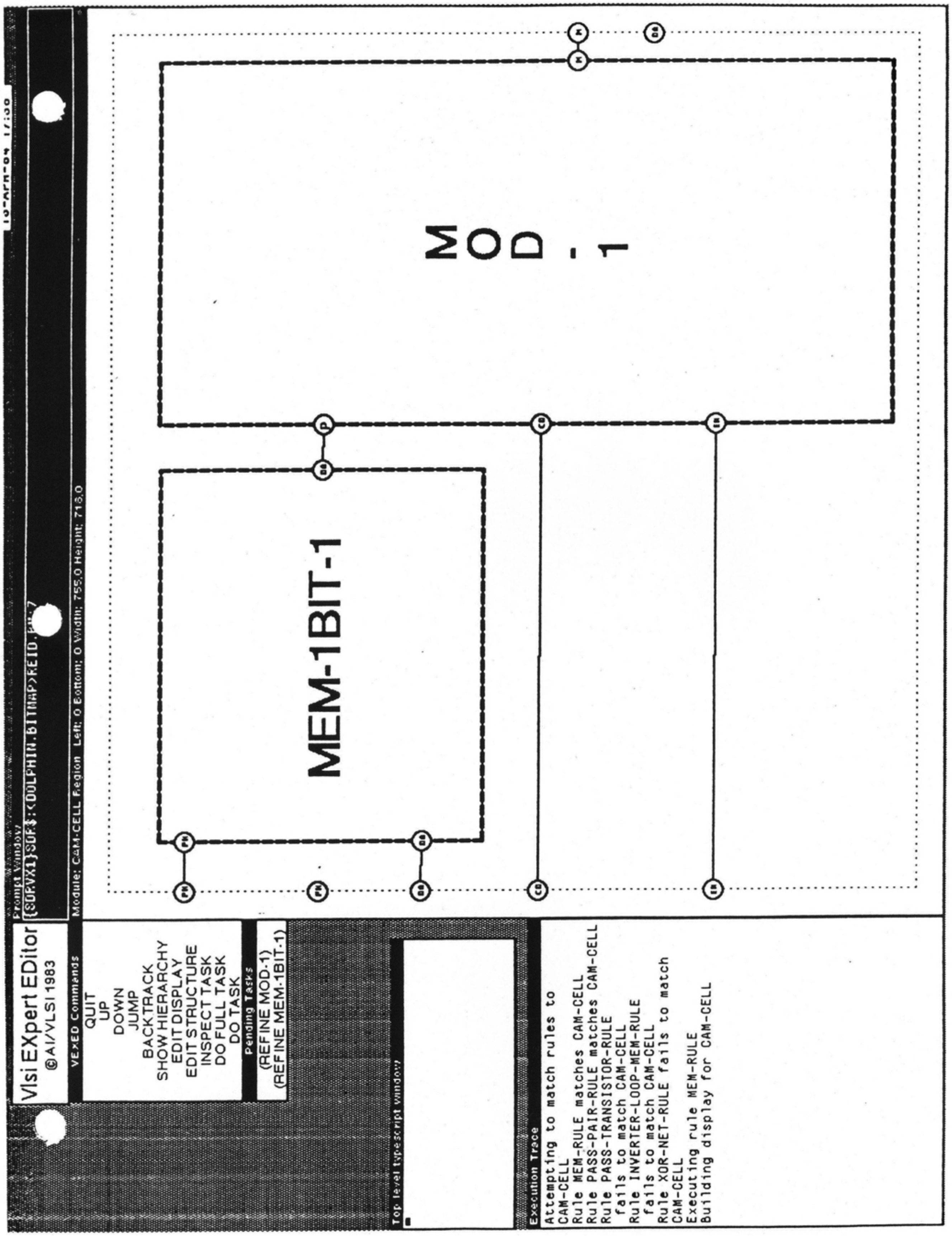
* * * * Dipmeter Advisor D * * * *
{SDR2020}PS:<ADVISOR.WELL0>LOGDATA.DA;2
Inspect Stratigraphic Analysis
TADPOLES
BHOLE
Azimuth Window
tk228
distributary-channel
15114.0 15124.17
tk229
distributary-fan
15137.61 15168.55
Zone
GR
Instructions
Select Non-Dipmeter Data
Mark Depth
Azimuth Histogram
Lisp
Exit Phase
dune
point-bar
stream-channel
chute-channel
swamp
marsh
upper-delta-plain
tidal-flat
tidal-channel
tidal-delta
Available Phases
Entry
Changes
Options
Token Types
Tokens
Zone log
Summary log
Prompt Window
typescript window

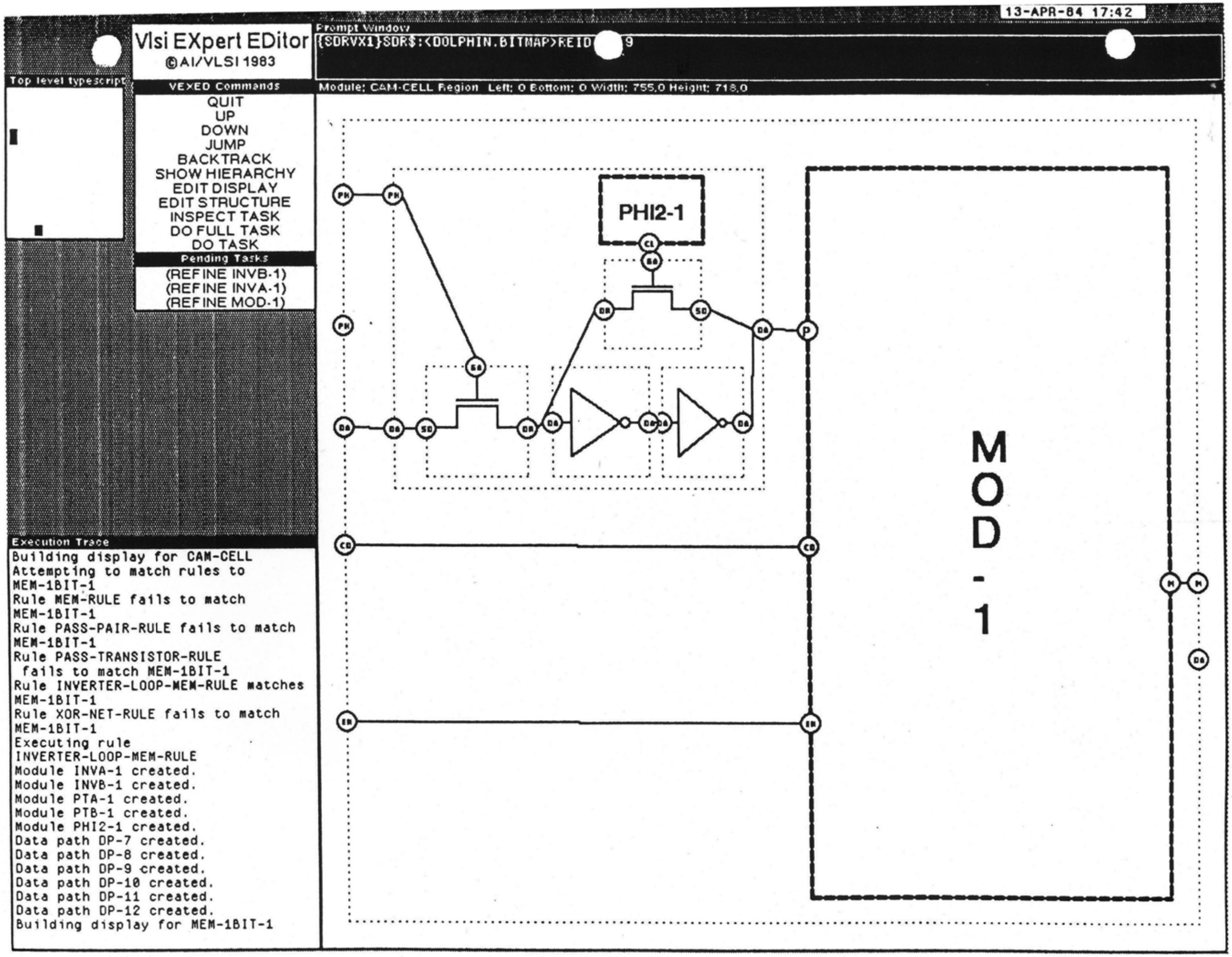

7-Sep-84 14:00
82←EF REPAINT-TEXTURE-MENU
loading from
{SDRVX8}DISKC:<CRYSTAL.BASE
LINE>PORTS.;135
prop unsaved
REPAINT-TEXTURE-MENU
83←REDO EF
REPAINT-TEXTURE-MENU
84←
CRYSTAL Control Panel
CURRENT JOB: LOGDB-EricPrime-1
CURRENT SERVICE: FACIOLOG
Help
Summary
Move
Shrink
Stop
PresentationDisplay-8 COMPLETE
1700.0
1750.0
1800.0
1850.0
-80.0
SP-3
20.0
3.0
MENEVEN-IFACP-2
45.0
0.0
GR-3
150.0
LOGDB-EricPr
FACIOLOG
CLEAN
LOGDB-EricPrime-1 - FACIOLOG - Description

Learning Apprentice Systems

(Mitchell, 1983)

A Learning Apprentice System is an interactive knowledge-based system that refines and extends its knowledge during *normal* problem solving.


Vlsi EXpert EDitor
©AI/VLSI 1983
VEXED Commands
QUIT
UP
DOWN
JUMP
BACKTRACK
SHOW HIERARCHY
EDIT DISPLAY
EDIT STRUCTURE
INSPECT TASK
DO FULL TASK
DO TASK
Pending Tasks
(REFINE MOD-1)
(REFINE MEM-1BIT-1)
Top level typescript window
Execution Trace
Attempting to match rules to
CAM-CELL
Rule MEM-RULE matches CAM-CELL
Rule PASS-PAIR-RULE matches CAM-CELL
Rule PASS-TRANSISTOR-RULE
fails to match CAM-CELL
Rule INVERTER-LOOP-MEM-RULE
fails to match CAM-CELL
Rule XOR-NET-RULE fails to match
CAM-CELL
Executing rule MEM-RULE
Building display for CAM-CELL
Prompt Window
Module: CAM-CELL Region Left: 0 Bottom: 0 Width: 755.0 Height: 718.0
MEM-1BIT-1
MOD-1


13-APR-84 17:42
Vlsi EXpert EDitor
©AI/VLSI 1983
Prompt Window
{SDRVX1}SDR$:<DOLPHIN.BITMAP>REID
Top level typescript
VEXED Commands
QUIT
UP
DOWN
JUMP
BACKTRACK
SHOW HIERARCHY
EDIT DISPLAY
EDIT STRUCTURE
INSPECT TASK
DO FULL TASK
DO TASK
Pending Tasks
(REFINE INVB-1)
(REFINE INVA-1)
(REFINE MOD-1)
Module: CAM-CELL Region Left: 0 Bottom: 0 Width: 755.0 Height: 718.0
PHI2-1
MOD-1
Execution Trace
Building display for CAM-CELL
Attempting to match rules to
MEM-1BIT-1
Rule MEM-RULE fails to match
MEM-1BIT-1
Rule PASS-PAIR-RULE fails to match
MEM-1BIT-1
Rule PASS-TRANSISTOR-RULE
fails to match MEM-1BIT-1
Rule INVERTER-LOOP-MEM-RULE matches
MEM-1BIT-1
Rule XOR-NET-RULE fails to match
MEM-1BIT-1
Executing rule
INVERTER-LOOP-MEM-RULE
Module INVA-1 created.
Module INVB-1 created.
Module PTA-1 created.
Module PTB-1 created.
Module PHI2-1 created.
Data path DP-7 created.
Data path DP-8 created.
Data path DP-9 created.
Data path DP-10 created.
Data path DP-11 created.
Data path DP-12 created.
Building display for MEM-1BIT-1

Following is the context for the preceding information.

Panel: Trends in computer systems technology

National Computer Conference, July 15, 1985, Chicago Illinois

Chair:
JOHN L. BERG, *Sperry Corporation,* St. Paul, Minnesota

Members:
SHELDON A. BORKIN, *IBM Cambridge Scientific Center,* Cambridge, Massachusetts
ALFRED W. DiMARZIO, *General Electric*, Bridgeport, Connecticut
JOHN HEFNER, National *Bureau of Standards,* Washington, D.C.
EUGENE LOWENTHAL, *Microelectronics and Computer Technology Corp.,* Austin, Texas
REID SMITH, *Schlumberger-Doll Research,* Ridgefield, Connecticut

Can the innovations in computer technology over the next 10 years be predicted on the basis of the measured trends of the last 10 years? Some analysts say no, citing microcomputers/workstations, database/information resource management, LANs, artificial intelligence applications, and user interfaces as technologies for which extrapolations are not easily made. A panel of industry analysts discusses this issue.

From John Berg:

Consider the following strawman scenario for the future of computing: imagine a world in which information appliances exist of many brands, each possibly specific to one or two applications but the appliances, in aggregate, include hundreds of thousands of different applications. Each of these can be plugged into the outlet in your home, office, school, library, etc. This action connects the appliance to a communications network that can be connected to one or a combination of information utilities. The information utility might be the highly secure company proprietary information collection or as public as the Encyclopedia Americana. In any case, the information utility collects, holds, maintains, and provides access to data. The "application specific" components of software, hardware, and display are provided by the information appliance.

The panelists represent five technologies critical to the appearance of such a future. Each was asked to present briefly the current trends and inhibitors which will affect progress as a prelude to general discussion among the audience and panel.

The analogy intended is, of course, to electric power and electric appliances. Once a water wheel provided power to a shaft running down the center of the factory. Leather drive belts connected the shaft to lathes, sewing machines, and looms. That water wheel was eventually replaced with a steam engine and now factories could be built away from water sources. Then the steam engine was replaced by a huge electric motor and the power source could be the cheaper of steam or water and far away to avoid their nuisance impact. Finally, the advent of the fractional horsepower electric put power almost anywhere and made it fit specific tasks. And each unit was cheap enough to permit hundreds to be placed in every home. In the computer industry we've clearly made the important step similar to the step to fractional horsepower motors.

Examples of information appliances: ATM, Airport seat selection terminals, CAD/CAM workstations, Library of Congress Index system, New York Times Information System terminals, Travel agency reservation terminals.

Examples of information utilities: New York Times Information System over public telephone lines, corporate interactive data systems, SABRE (airline reservation system), departmental systems with shared file servers.

Questions:

- Can one establish a network/file-server such that information appliances can access the file-server without requiring MIPS expenditure by the information utility responsible for the database? (This is apart from the issue of creating more demand for computing services.)
- Can a logical database be scattered over several file-servers?
- What is the role of a data dictionary/directory in such a scenario? To find a datum you must know the physical location and ask for its contents. Or you must know the datum's name which a name-server can translate into a physical location. A name of data may be “larger than a destroyer and sounds like ‘snow’”, or “the set of symbols in all natural languages performing the role of Arabic zero”, or “Who is our largest customer?” or “what aircraft between 1910 and the present had a configuration similar to the image on page ten of the Starwars report?”