

Printed By: SELL-KEN @PRUNE SENT: 91-05-19
15:05
FROM: NARAYAN PANKAJ @FORTY TO:
DL.ALL TANDEM @FORTY
SUBJECT: 2:Jim @ray's Vision ...

From [deleted] Mon May 6 00:38:38 1991
To: [deleted]
Subject: Jim Gray on super-server strategy for DEC

Jim Gray was recently lured out of Tandem by Digital. His conditions were that they set him up in a lab in downtown San Francisco, and that he get to have his lab work on whatever he wanted to. Here is his announcement to Digital of what he intends to work on: super-servers.

From: SFBAY::JIMGRAY "Jim Gray, Digital, 455 Market, 7th fl, SF CA 94105, 415-882-3955 dtn 542-3955 24-Feb-1991 1837" 24-FEB-1991 21:50:34.55
To: Subj [deleted I
: rambling memo on the need for clusters of super-servers

Note: This is a 'public' but drafty version of this memo. I have discussed these ideas with many people for many years. But only lately, with recent hardware and software developments, have the ideas crystalized. I hope to revise this paper in a few months based on criticisms from people like you.

A BUSINESS MODEL

By the end of the decade, boatloads of Posix boxes, complete with software, will be arriving in ports throughout the world. They will likely be 100 times more powerful than the VxX-9000, and will cost less than 10,000\$ each, including a complete NAS-like software base. No doubt they will come in a variety of shapes and sizes, but typically these new super-computers will have the form factor of a PC or VCR. These products will be inexpensive because they will exploit the same technologies (software and hardware) used by mass- market consumer products like HDTV, telephones, voice and music processors, super-FAX, and personal computers.

How can Digital and other 'computer' companies add one hundred billion dollars of value to these Posix boxes each year? Such added value is needed to keep 'computer industry' giants like IBM, Fujitsu, and Digital alive.

My theory is three-fold:

1. Provide some of the HARDWARE and SOFTWARE COMPONENTS these platforms need. Digital will no doubt continue to manufacture and sell hardware and software components. The new thing will be Digital's cooperation with "competitors" (for example we may outsource some components and OEM others).
2. SELL, SERVICE, and SUPPORT these boxes to corporations. This is Digital's traditional market. Although these boxes will be standard, corporations want to outsource the expertise to build, install, configure and operate computer systems. System integration will be a growing part of Digital's business.
3. Sell Digital-created CORPORATE ELECTRONICS (by analogy to consumer

electronics). Prepackaged systems which directly solve the problems of large corporations. The proliferation of computers into all aspects of business and society will create a corresponding demand for supporting super-servers which store, analyze, and transmit data. Super-servers will be built from hundreds of such boxes working on common problems. These super-servers will need specialized application software to exploit their cluster architecture. Database search and scientific visualization are two examples of such specialize application software.

As in the past, most revenue will come from the iron and from items I and 2: software, sales and service. But, the high margins and the high profit are likely to be in corporate electronics.

These are not a new businesses for Digital, but the business structure will be different. There will be more emphasis on using commodity (aka "outside') products where possible. The development cost of "standard' Digital products must be amortized across the maximum number of licenses. They must be marketed (OEMed) to our competitors as well as to our customers. Non-Standard development must focus on products that represent a Digital contribution. The cost of 'me too' products on proprietary platforms will be prohibitive.

This phenomenon is already visible in the PC-Workstation marketplace. In that market, standardized hardware with low margins provides the bulk of the revenue (and bulk of the profits). A few vendors dominate the high-margin software business (notably Microsoft and Novel).

3B MACHINES: SMOKING HAIRY GOLF 3ALLS

Today, the fundamental computer building blocks are cpus, memory chips, discs, print engines, keyboards and displays. Each is a commodity item. Computer vendors add value and get high margins by integrating these building blocks and by adding software to form workstations, mid-range computers, and to some extent mainframes. For example NCR, Sun, HP, Teradata, Sequent, and Tandem all use commodity components. Digital's MIPS-based products are also examples of this approach.

By the end of this decade, the basic processor building blocks will be COMMODITY BOARDS running COMMODITY SOFTWARE. The boards will likely have a 1 bips cpu (billion instructions per second), 1 GB (Giga byte) of memory, and will include a fairly complete software system. This is based on a technology forecast something like the following.

Year	I Chip CPU Speed	I Chip DRAM memory	IGB Disc Size
1990	10 MIPS	4 Mb	8"
1993	80 MIPS	16 Mb	5"
1996	500 MIPS	64 Mb	3"
1999	1000 MIPS	256 Mb	1"

This forecast is fairly conservative. It also forecasts the following costs for the various 1999 components

Year	1 Chip CPU	1 Chip DRAM memory	1GB Disc Size
1999	100\$	7\$	50\$

Given these costs, one could buy a processor, 40 memory chips, several high-speed communications chips, and ten discs, package and power them for a few thousand dollars.

Such computers are called 3B MACHINES (Billion instructions per second, Billion Bytes of DRAM storage, and a Billion bytes per second of IO bandwidth). A 4B MACHINE will support a Billion bit display, that is 3000x3000 pixels and each pixel 32 bits of shading and color.

In 1985, an engineer noted that these machines will be SMOKING-HAIRY- GOLF-BALLS. The processor will be one large chip wrapped in a memory package about the size of a golf ball. The surface of the golf ball will be hot and hairy: hot because of the heat dissipation, and hairy because the machine will need many wires to connect it to the outside world.

The software for 3B machines will contain all the elements of X/Open, Posix, DCE, SAA, and NAS. In particular it will include some standard descendents of Motif, SQL, OSI, DCE-Unix, X/Open transaction processing, and so on. Such a hardware-software package is likely to cost less than 10,000\$.

Future DISC FARMS will be built from mass-produced I² discs placed on a card much as DRAMs are placed on memory boards today. A ten-by-ten array of such discs will store about 100 GBytes. Disc array technology will give these farms very high performance and very high reliability.

These basic building blocks will be commodities. That is, the hardware will be mass produced and so will have very low unit price. Standard operating systems, window systems, compilers, database systems, and transaction monitors will have high volumes and so will also have low unit prices. This can already be seen in the workstation world. There, OS/2 Extended Edition and Open DeskTop provide a complete software system (database, network, and tools), all for less than a thousand dollars.

Some believe that in that era, machine designers will be changing the instruction set every 18 months. The only 'stable' interface 'Will be soft: the programming languages, operating system, and i/o libraries, databases, network protocols, and the like.

BUSINESS STRATEGY IN AN ERA OF COMMODITY SOFTWARE

Profit margins on manufacturing commodity hardware and software products will be relatively modest, but the volumes will be enormous. So, it will be a good business, but a very competitive one. There will continue to be a brisk business for peripherals such as displays, scanners, mass storage devices, and the like. But again, this will be a commodity business with narrow profit margins.

Why even bother with such a low-margin business? It is essential to be in the low-margin business because it is the high-volume business. The revenues and technology from this business fund the next generation. This can already be seen in the IC business where DRAM manufacturing refines the techniques needed for many other advanced devices.

There is a software analogy to this phenomenon. We already see that IBM cannot afford to do SAA and that Digital cannot afford to do NAS. These projects are HUGE; so they are being 'stretched-out" over the next decade. For either IBM or Digital to recover their costs, their efforts will have to become ubiquitous.

This is why NAS must run on 'foreign' hardware platforms. It must target the portable software market as a major new revenue source (not profit source). For example, to be profitable, RdbStar will have to be widely used in the Posix world on non-Digital platforms. Put glibly, Digital should build a database system and database tools that will put Oracle, Ingres, Informix, and the other database vendors out of business. Digital should produce a networking system that will put Novel, 3-Com, Ungermann-Bass and the other network vendors out of business. if it cannot afford to do that, it should partner with one of these vendors.

In some areas, Digital should buy technology from outside. For example most of the current Unix-DCE code body comes from outside today. We should accept this as a good thing. No company can afford to do everything. No company can have the best implementation of each standard.

SALES AND SERVICE IN A COMMODITY WORLD

I have little to say on this topic. It is one triad of the three components of Digital's future. It is Digital's traditional business. Digital's size and wide geographic distribution will be a key strength in marketing and supporting any products the company offers.

FUTURE MAINFRAMES: THE 3T MACHINES

In a classic paper Gordon Bell and X [ref??] defined the basic laws of computing. One of their key observations is that there are seven

tiers to the computer business: These tiers are roughly categorized by the dollar value of the computers:

- 10\$: wrist watch computers
- 100\$: pocket computers
- 1,000\$: portable computers
- 10,000\$: personal computers (desktop)
- 100,000\$: departmental computers (closet)
- 1,000,000\$: site computers (glass house)
- 10,000,000\$: regional computers (glass castle)

They observed that each decade, computers from one tier move down a notch or two. For example, current portables have the power and capacity approximating that of a 1970 glass-house machine. machines with the power of 1980 workstations are now appearing as portable and even pocket computers.

Bell observed that individuals can be capitalized at about 10,000\$ per person on average. That more or less defines the price of the typical workstation.

The costs of departmental, site and regional servers can be amortize over many more people, so they can cost a lot more.

So, what will the price structure look like in the year 2000? Will there be some super-expensive super-fast neural-net computer that costs ten million dollars? If future processors and discs are very fast and very cheap, how can one expect to build an expensive computer? What will a main-frame look like?

One theory is that the mainframe of the future will be 10,000\$ of hardware and 999,990\$ worth of software. Being a software guy, I like that model. This is the direction the Air Force is going. Each new fighter is smaller and lighter than its predecessor, and each new one costs much more. They way they do this is to fill the fighter with expensive software that weighs nothing and has no volume. Perhaps we can do this in a non-governmental cost-is-no-object world, but I doubt it.

OK, so the 100% software theory is blown. What else? Perhaps the customer will pay for 999,990\$ worth of maintenance on his 10,000\$ box? Probably not. He will probably just buy two, and if one breaks, discard it and use the other one.

OK, so we conclude that the mainframe itself will cost about a million dollars in hardware. What will a million dollars buy? Well it will buy (packaged and powered) about:

~ 1,000 processors = 1 tips (trillion instructions per second) or
~ 30,000 DRAMS (@256Mb) = 1 TB (one terabyte RAM) or
~ 10,000 discs (@1GB) = 10 TB (ten terabytes disc)

So, THE MAINFRAME OF THE FUTURE IS A 3T MACHINE!

WHO NEEDS A 3T SUPER-SERVER?

What would anyone do with a 3T machine? Perhaps the mainframe of the future is just a personal computer on each desk. One thousand of them would add up to the 3T 'site' computer. The system is the network!

It seems likely that each worker will have one or more dedicated 3B computers, but it also seems likely that there will be some jobs that require more storage or more processing than a single processor, even one of these super- powerful 3B ones.

Consider for example the problem of searching the ten terabyte database mentioned above looking for a certain pattern. If one processor searched through the 10 TB using a single 3B processor, and using current software (e.g. Rdb), the search would take three hours. By using a thousand 3B processors in parallel, the search would take about 10 seconds.

Similar observations apply to other applications that analyze or process very large bodies of data. Database search is prosaic compared to the data visualization algorithms which map vast quantities of data to a color image. These search and visualization problems lend themselves very nicely to parallel algorithms. By doubling the number of processors and memories, one can SCALEUP the problem (solve twice as big a problem), or SPEEDUP the solution (solve the problem twice as

fast) .

Some believe that the 3B machines spell the end of machines costing much more than 10,000\$. I have a different model. I BELIEVE THAT THE PROLIFERATION OF INEXPENSIVE COMPUTERS WILL INCREASE THE NEED FOR SUPER-SERVERS.

Many forces argue that a fraction (say 25%) of future computer expenditures will go for super-servers. The central arguments are:

Manageability: People do not want to manage their own data centers.

Yet the trends above suggest that we will all own a personal data center in 1999. Each PC and each mobile telephone will be a 3B machine. There will be a real demand for automatic archiving of data and automatic system management. This will likely be a centralized service. A simple example of this is visible today with the success of X-terminals which move management issues from the desktop to the closet. The bandwidth and data storage demands of such servers which support hundreds or thousands of 3B machines will be enormous.

Control: The proliferation of machines will make it possible, even easy, to access centralized services and resources. No longer will you go to the video store to get a videotape, you will download it. No longer will you search paper libraries for information, you will have a server do it for you. These resources (movies, libraries) will contain valuable information. Central utilities (or at least regional utilities) will want to control access to them. So they will set up super-servers that offer an RPC interface to them.

The typical strategy today is to spend half the budget on workstations, and half on print, storage, and network servers. In the end, the split may be more like 90-10 (this is the ratio of cost of ATMs to the host server in an ATM network). But the point is that servers will not disappear. Put another way: FAST CLIENTS WANT FASTER SERVERS.

WHAT ARE THE KEY PROPERTIES OF SUPER-SERVERS?

Servers must obviously have the following properties:

PROGRAMMABLE: It is easy to write applications for the server.

MANAGEABLE: It is easy to manage the server.

SECURE: The server can not be corrupted or penetrated by hackers.

DISTRIBUTED: The server can interoperate with other super-servers.

SCALEABLE: The server's power can grow arbitrarily by adding hardware.

ECONOMIC: The server is built from commodity components.

HIGHLY AVAILABLE: The server does not lose data and is always 'up'.

CLUSTERS AND CLUSTER SOFTWARE -- THE KEY TO 3T MACHINES

Servers typically need to be as powerful or more powerful than their

clients. They must serve hundreds or millions of clients. How can powerful servers with all these properties be built from commodity components? How can a collection of hundreds of 3B machines be connected to act as a single server? What kind of architecture is needed? What kind of software is needed?

Digital has it now! Digital currently offers VaxClusters which scale to hundreds of processors. The cluster has many excellent programming tools, it is a single management entity, and it is secure. Clients access ACMS servers on the cluster not knowing where the servers are running or where the data resides. So the cluster is scaleable. Processors, storage, and communications bandwidth can be added to the cluster while it is operating. VaxClusters are fault-tolerant. They mask faults with failover of discs and communications lines. VMS has the transaction concept integrated into the operating system. The VAX family is built from commodity components and is among the most economic servers available today.

Well, that is the official Digital marketing story, and there is a grain of truth to it. But, the details of the VaxCluster do not deliver on most of these promises. The VaxCluster really only scales to tens (not thousands of processors), the programming and management tools do not offer much transparency; each component is managed individually. VMS is not especially secure. Virtually none of the tools use more than one-processor-at-a time in running an application. This dramatically limits its ability to scaleup or speedup applications by adding hardware. The VaxCluster price is not especially economic when compared to PC-based servers. And, there are many single points of failure in the VaxCluster software.

But, the VaxCluster is certainly a step in the right direction. ', also the direction that most other vendors have adopted. Notable examples are:

- * Teradata builds clusters out of the Intel x86 family and proprietary software. These clusters act as back-end SQL servers to mainframes and LANS. The systems feature economy, scalability, and fault-tolerance. The largest clusters are a few hundred processors and a thousand discs.
- * Tandem builds clusters out of a proprietary hardware-software combination. Tandem systems run as network servers. The system features are exactly the super-server list above. The systems scale to about 100 processors and to a few hundred discs. Customers complain that the systems are not manageable.
- * IBM Sysplex is a cluster of up to fortyeight 390 processors. there is very little software to support this cluster At present hardware

In addition, the IBM AIX system (their Unix clone) running on the RS/6000 hardware (their RISC engines) has software to support the cluster concept. Their research group has acquired the source to the University of Wisconsin Gamma database software, and seems intent on porting it to an AIX cluster to make a SQL super-server.

- * NCR, after hiring TeradataUs chief architect (Phil Neches), has adopted the Teradata approach. It is hoping to build systems which scale from the palm to the super-computer by building clusters of Intel x86 or

Risc processors. At present, the NCR plan is in development. (See "NCRUs 486 Strategy", Moad, J., Datamation, V36.23, 1 Dec. 1990, pp. 34-38).

* Intel is building a hyper-cube of 2000 processors, called the Delta machine. The first instance of this machine (actually only about 530 processors) is now installed at CalTech. Intel has no clear idea about how to program or manage such a system. But the faculty at CalTech are working hard on the programming issue. Smaller Hypercubes are being seeded around Universities and industry.

Looked at in this light, no vendor is ready to build a thousand processor 3T machine and the associated software. Some are ahead of others. In fact, I would place Digital second behind Tandem in the race -- the VaxCluster is an excellent start. Digital's VMS staff and other engineering organizations have experience with clusters that their competitors lack. But, neither Digital nor Tandem are taking advantage of their lead to extend their clusters sizes by another order of magnitude or two.

It is important to understand the virtue of clusters. The idea is to add more discs, processors, memory, and communications lines, and get more work out of the system. This speedup and scalup should go from one processor to several thousand.

The ISO protocol stack has an elevator shaft running down the side called network management. That elevator shaft is not standard. There are implementations which are defacto standards (e.g. NetMaster, DECNet EMA, NetView), but they are not standard. ISO currently punts on issues like security and performance.

The SQL standard looks the other way about most errors (they just define the standard ones, performance (no performance monitor), utilities (no load/dump, import/export), and administration (no accounting, space management,...)).

The SO reactionaries believe that computing is fractile: there is complexity in every corner of it. Workstations hide this complexity by dealing with a single user and ignoring system management. But "real" computers will not be able to hide some of this complexity.

When building a workstation, one aims for simplicity.

Microsoft/IBM have an "open" standard MS/DOS which is a single code body (which is its Own spec). Apple Macintosh is a similar story. The Unix world has a standard 'open' application programming interface which allows many simple stand-alone programs to be easily ported from one platform to another. The CICS world has 300,000 programmers who know and love the CICS application programming interface. It is open and standard. It does not change.

But there is a separate world. There is no real open-standard operations interface for a network of PCs boxes, or for the applications that run them. All the tools to do these operations tasks are proprietary. The CICS operations interface is not Well documented, is not open, and it changes from release to release. It is the elevator shaft.

Certainly, the standards organizations have place-holder bodies that are "working" on these elevator shafts, but the SO reactionaries believe such efforts are doomed. These big-systems issues are too specific to make it into commodity standards or products.