```
 _   _   _   _   _   _   _
| | | | | | | | | | | | | |
|d| |i| |g| |i| |t| |a| |l|
| | | | | | | | | | | | | |
|_| |_| |_| |_| |_| |_| |_|
```

I n t e r o f f i c e     M e m o

TO: see "TO" DISTRIBUTION

DATE: MON 4 JUN 1984  3:56 PM EDT
FROM: KEN OLSEN
DEPT: ADMINISTRATION
EXT:  223-2301
LOC/MAIL STOP: ML10-2/A50

MESSAGE ID: 5238585527

SUBJECT: A CORPORATE EMERGENCY

When a pilot says he is in trouble and needs some help, he is asked, would he like to "declare an emergency."  He is often reluctant to do this; he will try to get by without that formal act.

When he declares an emergency, all resources are available to help him, everything in the neighborhood comes to a halt, and the highest priority in that area is to get him safely to the ground.

This is expensive, and after it is done there is a formal, legal inquiry into how it all happened.  This, of course, makes people reluctant to "declare an emergency."

I am about to declare an emergency at Digital in the world of small computers.  There are a number of reasons why we did a terrible job in the personal computers and why we do poorly in the small business.  We have discussed some of them, but they are relatively unimportant.  We built too much quality into the product, we announced them before they were done, we took orders before the products were completed, and we spent more money on advertising than we did on servicing the customer.

These problems are survivable and we have already proved this.

have a good product and the customers love them, but we are missing two things that we set about to do as initial goals and the stated goals all along the program. Management feels its job is budgets and finance, and that, miraculously, technical goals should be taken care of by someone else and management will not take part in these goals.

Corporate management spends a lot of time squeezing budgets so that on paper we make a profit, and Engineering management spends a lot of time squeezing budgets so that they can get all the projects they initiated done, even though none of those projects solve the basic Corporate needs.

We are missing two basic things that we have said many times in the several years we are going to accomplish.

We still have no plan, no techniques, and no ambition to be able to tell a small customer how he can integrate his small computers into a system for running a business, his agency, his brokerage company, his small hospital, or his store.

We still have no plan to make a system which is so simple that anyone can understand it, anyone can sell it, anyone can install it. We do not need another wheel-around computer that has more options, more freedom, more complexity than we have ever had before in any of the computers we have ever built.

Our plans are still to insist that the customer figure out if he wants to have serial or parallel communications, synchronous or asynchronous lines, buffered or unbuffered lines, RS232, RS423, RS422, RS449, or 20 ma. loops. Then, to top it off, the customer has to decide which combination of the above he wants in funny sounding modules.

We do not know the difference between distributed processing and time sharing, but we expect the customer to figure it out and order all the parts he needs to do what he wants to do. Of course, we won't have the software to help him, but we will have a vast number of pieces which our salesmen, of course, cannot figure out how the customer should use, but with all of that stuff, the customer surely should be able to figure out something.

I am about to declare an emergency.

Most of our small business marketing groups came from the OEM world, which insists on all this freedom. To them, systemness means they have never forgotton anything the customer may possibly want in some weird set of conditions or in some state of ignorance. I do not think they could sell a product which was so simple that anybody could install it and use it.

I am about to declare an emergency which will set about to develop a naive, limited, simple minded J-11 Q-BUS smart server that would integrate distributed small processors in an organization. The same box would be a disk server when ETHERNET is necessary.

I am about ready to contract outside, a small business plan that would have naive, simple minded, limited set of computers and hardware that would allow a small number of configurations that anyone could understand, and software which is naive and efficient, but readily understood by everybody and that are hooked together with our new RS423 serial lines that are so easy to use anyone can do it, but then not allow in this system, people to use any other system for interconnect. If people want a special system, they can go somewhere else or buy it through an OEM, or through special systems.

I want to be able to say that with our system we can integrate anybody's PC, but only in a very simple minded way.

Gordon insisted on dropping the Q-BUS for small computers, because installation was so expensive as compared to a personal computer. Part of this was illusion, because our marketers then offered all the software that was standard within the industry and never got around to integrating into a system necessary to run a business. I thought it was clear for years that we made a mistake, and that we should have, and that we should solve the office, the business, the organizational problem, but do it with a traditional computer system, but limit them severely, but not as much as we have when we limit ourselves to off-the-shelf, bubble-pack software. However, we are in worse shape today than when we started, because we have learned a lot more complexity in our Q-BUS machines. We have made it more complicated for people to use them, and we made the machines even more expensive than they used to be because we know more options and freedoms to build into them. In the personal computer area, we are still trying harder and harder to get more standard software off-the-shelf, but still not trying to accomplish what the customer wants, and not fully realizing what we are missing.

KHO:blk
KO3:S10.34
DICTATED BUT NOT READ


Fri 15-Jun-1984 07:59 EDT


16-JUN-84   8:36:27   S 00558   USMT
USMT MESSAGE ID: 19840615193639/2028@BRUTUS

"TO" DISTRIBUTION:

| | | |
|---|---|---|
| MARK ABBETT | ARTHUR ARONOVITZ | LYN BENTON |
| LOU DIFINIZIO | JOAN DOOLITTLE | FRED FORSYTH |
| DAVE GREENLEE | GERRY GRINCAVITCH | STEVE HANDLER |
| FRED JOHNSON | PAUL MCGAUNN | DEBBIE PAKKILA |
| ALAN SELIN | CHRIS SHATARA | MARION SHATTUCK |
| STEVE TOCMAN | | |

2-3-95

CRAIG,

HERE ARE SOME
MORE OLD MEMOES
FROM KEN OLSEN
+ GORDON BELL

HAVE FUN!

JAY

_staff mtg_

```
+---------------+
: d i g i t a l :
+---------------+
```

INTEROFFICE MEMO

TO: OPERATIONS COMMITTEE

CC: Richard Berube
    R. L. Lane

Date: 22 December 1981 .
From: Ken Olsen
Dept: Administration
MS: ML10-2/A50 '     Ext: 2301

SUBJ: THE DOWNFALL OF THE JAPANESE AND THE 25TH ANNIVERSARY OF DIGITAL

I think there are signs of serious danger in the Japanese industry.
There are signs that they are believing the propaganda which says they
are the only ones competent enough to make a quality product, in
quantity and at low price.  Once they believe that they are over the
hill.

Digital is often in grave danger. We have been going through one of
those periods during the last few years of boom time.  We get to
believe that we are the only ones who can make a quality product, in
quantity and at low price.  After this kind of period our quality goes
down, we have trouble getting the quantity out, and our prices soar.

I think the last thing we need to do is use our 25th anniversary as an
opportunity to tell ourselves how great and invincible we are, or to
tell ourselves that we have the secret of success in this market, even
though we can't tell what the secret is.

We have emerged from these periods through humility, which made us
work and change and innovate and criticize so that we have been able
to survive in this industry. The kiss of death will probably be a
coordinated effort in telling ourselves and our mothers-in-law how
great we are.

If anything, the secret of our success has been humility, fear of our
overwhelming competitor and breaking down tasks so that everyone knows
what he is doing and how he will be measured, and by not having people
who tell us what to do, but rather people who are doers.

Now we think that we have the secret of success, and everybody can
tell everybody else what to do and no one has to do anything except
brag about the miraculousness of being what we are.

I am terrified by people who want to show off to their mothers-in-law
about what Digital is, but don't have any idea what Digital really is.


KHO:ml
KOl:S8.12

*[handwritten: Some food method... Litt act on them. Jorden]*

```
+---------------+
: d i g i t a l :
+---------------+
```

JAN 20 1982

I N T E R O F F I C E   M E M O

TO: OPERATIONS COMMITTEE  *[handwritten: Ken / Copy Staff]*

Date: 22 December 1981
From: Ken Olsen
Dept: Administration
MS: ML10-2/A50    Ext: 2301

SUBJ: HERESY AT DIGITAL

Heresy has taken over the fundamental beliefs at Digital. We now believe, and we tell magazines and even write in our house organs, that the secret of our apparent success is matrix management. In matrix management everybody is boss and everyone has many bosses. We also believe that if we measure someone, he has to make all the decisions. He doesn't have to manage, he just has to make all the decisions, because it is unfair to measure him unless he decides everything. Of course this leaves no time for management. We have strayed a long way from the original theories upon which we based our organization.

A few of the tenets of our belief were:

1.  Everyone does and no one tells others what to do.

2.  He who proposes does.

3.  Everyone has a defined job which he can identify and on which he will be measured in short intervals.

4.  Short-term goals which we can see and measure the results. *[handwritten: → Really important in high technology because everything changes so fast! (The demise SDS, GE RCA, & historic]*

5.  Every manager will break down the jobs under him into pieces and assign them to individuals who will be measured.

6.  *[handwritten left margin: womb-to-tomb]* An engineer will start and finish a job and will not break it down. (Between research, development and engineering and then another engineering group and finally engineering for manufacturing and then someone who writes the manuals and then someone who makes the thing work.)

7.  An engineer can do his own writing.

8.  Marketers market.

9.  Managers manage.

10. Engineers engineer.

Joe Kurta
ML4-4/E99

11. We will keep a lean staff and give everybody several jobs.

12. We will trust and gamble on young, potentially powerful people.

The product line organization has been very successful. With it we have been able to do many more activities then companies who have one central planning or are directly run by one set of managers. The success came about because each of the groups set goals and were measured by them.

We have evolved from this original way of doing things by getting the product line managers, not to be measured, but to make all decisions and then to police all those decisions around the world. We have called this matrix management. The original goals were quite different. We gave the product line manager those resources he needed to manage the product line. He didn't make all the decisions and that was not the major goal of the product line. The result is that we have collected many, many people in the product line who really don't do anything, they police others. We now have many, many people who are not growing professionally in either technical skills or management skills and we are drastically overstaffed.

The strange thing is that we now propagate the idea that the reason for our success is this enormous overstaffing, all this enormous red tape and controls and getting involved in everybody elses management. The original goal was to make things simple and the people who were going to be measured could make the decisions or proposals without vast amounts of red tape and people and organization.

There is a lot to learn about our original product line breakdown. We assigned a multitude of jobs to individuals and measured them and we were able to accomplish things that the president could not accomplish by himself. This concept has application in many parts of the Company.

For example: We are often terrified by the Japanese skill in packaging things for shipment. You often open their products and they are magnificently packaged. You just have the feeling of quality and elegance. Then you open up a Digital package and there is blue pastic with foam inside that looks like dried up vomit.

This does not mean that we are not skilled, it is just that we have picked up the new, modern, Digital approach to management. The boss does everything and when he runs out of energy, it doesn't get done. We spend enough money in packaging. We spend enough money in engineering of packaging, but we don't assign tasks. If we would just tell our packaging people, "Your goal is to have cheaper, but more elegant packaging than the Japanese,"

# COMPANY CONFIDENTIAL

we probably would get it. But when the managers do not want to
do what I did to Digital when we were a 14 million dollar
company, we will have so many, many things fall through the
cracks because our managers do not have the energy or span of
interest to do everything themselves.


KHO:ml
KO1:S8.17

In the attached memo Ken offers some thoughts on a variety of
topics...general management theory, Company planning, the role of 'staff',
product selection, motivation, marketing etc. Very interesting reading if
you haven't seen it.

Jim


From:    NAME: Ken Olsen
         FUNC: Administration
         TEL: 223-2301                          <OLSEN.KEN AT A1 at CORA @ CORE>
Date:    21-Dec-1989
Posted-date: 21-Dec-1989
Precedence: 1
Subject: HOW TO MANAGE A HIGH-TECH COMPANY
3
To:      See Below
CC:      See Below


Everyone seems to feel that they know how a high-tech company
should be managed, and most of those think that they can do a
better job than what they see being done.  The college student,
the reporter from the NEW YORK TIMES the engineer, the janitor
all understand how it should be done.  However, their theory is
usually based on one simple concept.  This concept is usually
based on what they see missing in the management they've
observed.  Some would agree that what is needed is stability, and
others argue that the secret is quick, rapid and frequent change.
Still, others would say what's needed is strong, tough management
with firm financial controls.  Others say the secret to success
is giving freedom to everyone.  Some argue that giving P&L
responsibility to someone is the magic that automatically creates
success.  Some believe the secret is to have a very human,
sensitive organization, and others feel toughness is the one
secret to success.

Few people stop to observe that very few high-tech companies
avoid stagnation after reaching a certain level or avoid
completely disappearing.  When one looks at what has happened to
the bright stars in high tech of thirty, twenty, ten, and five
years ago, one might come to the conclusion that more thoughtful
consideration of the theory of management is in order.

In looking at the history of high tech, one might come to the
only sure conclusion that, like raising children, the only ones
sure to fail are those who know they have everything figured out
and don't have to learn anything.  However, a little more thought
raises doubt for that simple theory.

Life is filled with paradoxes and conflicts in raising children
or managing a high tech company, and integrating them all in a
balanced way.  Like raising children, love, enthusiasm,

enjoyment and fun compensates for a lot of the weaknesses in theory.

THE MAGIC OF P&L RESPONSIBILITY

When we ask someone to pilot our airplane, or we have someone do open heart surgery on us, or if we have someone design a new bridge, we give them complete responsibility. We give them freedom to make choices and judgments, and we give them great trust. However, we do not say they are free to invent their own theories, teach themselves, try out new ideas which they invented themselves that are not commonly accepted by the field. In fact, for these very critical jobs, we are very careful to make sure that their training is in line with the theories we believe in. We expect to have standards which they will follow, and we have checks and balances to make sure they don't do anything way out of line. This in no way limits their freedom, responsibility, capability for creativity, and freedom to make judgments when needed.

However, there is a tendency in running businesses to pick a person for, often irrelevant reasons, and then let him go headlong into the job without any discussion of theory, standards, or measurements, like you would expect a bridge designer, doctor or a pilot to follow.

When people are asked how they manage, they often use words from textbooks, but they don't explain what they mean, or maybe they don't understand what they mean. For example, it is so commonly believed in the circles we move in that giving someone P&L responsibility guarantees success, and the magic of those words makes everything work well. We often point out the failure of communism where entrepreneurs don't have P&L responsibility with freedom to make choices and decisions in investments, and they don't have a reward system for success and failure. So, from this we often, without much thought, conclude that simply saying the words P&L creates magic.

We, indeed, do see magic often when there is a group with one product who feel complete P&L responsibility, who can justify investments by their success, who can control their overhead, their R&D, and their expenditures on irrelevant things such as buildings, flagpoles and airplanes. They are forced to balance their income with their expenditures and forced to make a profit if they are going to grow and if they are going to have a reward.

Not all small groups with P&L responsibility survive. Those who are first interested in the accoutrements of the office, or those who can't add or subtract on the P&L statement, disappear quickly.

The thing that people find hard to understand is that, when you give P&L responsibility to someone running a third of a very large corporation, you do not give the motivation and reward

system to those individual products upon which the success of the organization is dependent.  In fact, it is much closer to the Russian system.  Like the Russians, the top man has P&L responsibility.  If this was enough, it would guarantee success of the Communist system.

When the top man has P&L responsibility and he runs everything himself, the individual group is not motivated by the need to justify future expansion by results.  There is no great motivation to cut costs when the expenditures don't balance the income.  There is no great need to be creative in doing all the things necessary to make a profit.  In an organization like this, it is clear that profit is a very unnatural thing because the pressure on expenditures is much greater than the pressure on income.

In the name of capitalism, we give P&L responsibility to the leader of a large number of groups, but we end up, from the individuals in the groups point of view, being very close to the Communist system.  Investments are made by personality, politics and appearance, and no group has a feeling of controller influence on their future, or little upon their success.  Unlike the agencies in the state of Massachusetts, the game is to politic for as much expense money as possible, because that is the secret for success and expansion.

My friends in charge of large segments would feel very hurt at this because they are very bright, competent, conscientious and economical, but even if they can't get around to admitting it, they don't have the time or energy to be the leader and take P&L responsibility for each of the multitude of units below them.

THE PLACE FOR STAFF

Staff can be very dangerous in an organization.  Staff can really set all the goals and make all the decisions.  They do so quietly, subtly, and because they are all good people, they do it unobtrusively.  They simply don't let anything into the system that they don't approve of, and that means they control everything.

Staff is also not rewarded for getting jobs done, taking risks or normal success.  Their only goal is to avoid trouble and mistakes, and it is not uncommon for them to tie the whole organization up with enormous red tape so nothing useful gets done.

However, staff is the secret to managing the Company.  Staff is the key to success and staff is the one tool management has to run the organization.

The head of a large company or the head of a sector of a large company can only spend energy on a small number of most important things.  There is a normal tendency for staff to do only those

things which help the boss, and therefore, they concentrate all their effort on those small number of things which are most important to the boss.

The first thing the staff should do is concentrate on all those things that the boss is not concentrating on to make sure that they get due emphasis and concern.

The boss gets all his financial reports designed around those things he's most interested in, or all his responsibilities as a big glob. He has no tools with which to manage the details. Everything is glommed together or allocated arbitrarily. When he has to pass judgment on the cancellation or expansion of the product, the actual results of the product get lost by averaging and arbitrary allocation.

The people responsible for a product are devastated by the fact that their future is dependent more on averaging and allocations than upon actual result, and their enthusiasm for running their unit in a business-like way is clearly discouraged. The financial person who arbitrarily allocates the expenses has complete power over the success or failure of the project.

JANUARY 1990 FIVE-YEAR PLAN

The five-year plan I have asked the Management Sciences Group to collect, from all the units of the Company, during the month of January 1990, will hopefully demonstrate how the staff can be useful in managing the Company. They all must set up a program for collecting from each of the units the plans, results and data about the unit, be they very small or reasonably large. Their goals and plans will be clearly specified. Their measurements will be specified, and the results will be clear. Little or no arbitrary allocations will be used--just what they spend money on will be included.

This collection of plans should be of little burden to the sectors because they will be just recasting the figures they have to separate all the individual units and document them.

When we have all the separate units collected by computer, it would be easy to recast them to look at products and plans and to study questions that need to be answered.

From this, we can also quickly analyze which products have done well and which ones have done poorly. Then analyze which ones have advertised and which ones have done real marketing, and see it as a correlation between advertising and results of products. We can also figure out if there really was a plan to sell a product or if it was just planned to be developed and then got into production.

From this data, we should be able to find out if all the marketers in the Corporation are part of a plan, or if each

marketing group does what's right in their own eyes.  We also
ought to find out if there are marketing groups that work for the
"state" who decide which products they will market and which ones
they will not, and if the decision is not left to the people who
supposedly have product responsibility.

We also ought to be able to pick out those products which may be
good and may be a success as far as completion, but for which we
have no plans to sell, and cancel them.  Then cancel the
development of those projects which don't include the plan to
advertise and market.

KHO:mg
KO:3646
(DICTATED 12/21/89 BUT NOT READ)




To Distribution List:

NAME: John Sims <SIMS.JOHN AT A1 at CORA @ CORE>,
NAME: Win Hindle <HINDLE.WIN AT A1 at CORA @ CORE>,
NAME: Martin Hoffmann @CORE <HOFFMANN.MARTIN AT A1 at CORA @ CORE>,
NAME: Jim Osterhoff <OSTERHOFF.JIM AT A1 at CORA @ CORE>,
JOHN WETMILLER @OGO,
STEVE FRIGAND @OGO,
NAME: Jack Smith <SMITH.JACK AT A1 at CORA @ CORE>,
NAME: Bill Demmer <DEMMER.BILL AT A1 at CORA @ CORE>,
NAME: Bob Glorioso <GLORIOSO.BOB AT A1 at CORA @ CORE>,
NAME: Dom LaCava <LACAVA.DOM AT A1 at CORA @ CORE>,
PIER CARLO FALOTTI @GEO,
NAME: Dave Grainger <GRAINGER.DAVE AT A1 at CORA @ CORE>,
NAME: DICK POULSEN <POULSEN.DICK AT A1 at CORA @ CORE>,
NAME: Don Busiek <BUSIEK.DON AT A1 at CORA @ CORE>,
NAME: DONALD ZERESKI <ZERESKI.DONALD AT A1 at CORA @ CORE>,
NAME: BILL HANSON <HANSON.BILL AT A1 at CORA @ CORE>,
NAME: BILL STRECKER <STRECKER.BILL AT A1 at CORA @ CORE>,
NAME: PETER SMITH <SMITH.PETER AT A1 at CORA @ CORE>,
NAME: Bill Johnson <JOHNSON.BILL AT A1 at CORA @ CORE>,
NAME: BRUCE J RYAN @CORE <RYAN.BRUCE J AT A1 at CORA @ CORE>,
NAME: Dick Farrahar <FARRAHAR.DICK AT A1 at CORA @ CORE>,
NAME: Willow Shire @ CORE <SHIRE.WILLOW AT A1 at CORA @ CORE>,
NAME: Ken Olsen <OLSEN.KEN AT A1 at CORA @ CORE>,
NAME: BOB PALMER <PALMER.BOB AT A1 at CORA @ CORE>,
NAME: Grant Saviers <SAVIERS.GRANT AT A1 at CORA @ CORE>

CC Distribution List:

NAME: Abbott Weiss <WEISS.ABBOTT AT A1 at CORA @ CORE>

To Distribution List:

PETER GRAHAM @MRO,
LAGERQUIST @CURIE@VAXMAIL,
SCORZONI @CURIE@VAXMAIL,
TARBI @CURIE@VAXMAIL,
CORMIER @CURIE@VMSMAIL,
LIZ ABERDALE @MLO,
STADECKER @SHARE@VMSMAIL,
WATSON @RDVAX@VMSMAIL,
HGREENFIELD @CURIE@VMSMAIL

Preliminary Draft for Comment by Digital Engineering Community

## HEURISTICS AND COMMENTS FOR BUILDING GREAT PRODUCTS
Gordon Bell, Vice President, Engineering

Product goodness is somewhat like pornography, it can't fully be
described, but we're told people know it when they see it.  If we can
agree on heuristics about product goodness and how to achieve it -
then we're clearly ahead.   Five sets of dimensions for building great
products need be attended to (roughly in order of importance):
  . a responsible, productive and creative engineering group;
  . product and design metrics (competitiveness);
  . design goals and constraints;
  . product evolution, revolution and death; and
  . the ability to get the product built and sold.

## ENGINEERING GROUP
As a company managed primarily by engineers, groups are encouraged to
form and design products.  With this right, are responsibilities.

The Team must have:
  . a chief designer/chief programmer to formulate and lead the
  resolution of the problems encountered in the design;  No matter
  how large the project, it must be lead from a "single head".  We
  often make two errors in leadership: having no clear technical
  leader/problem resolver; and abdicating to a committee.

  Committees do not do design!  They are never held responsible,
  nor are they rewarded or punished.  Committees can review.
  . Workers should demand good Management
  . management who understand the product space and who has
  engineered successful products;  The two most important jobs are:
    . making sure that everyone knows their job; and
    . setting and reviewing work on a timely basis, ie. MBO.
  . team skills and resources to implement the proposal so that we
  adhere to the cardinal rule of Digital, "He Who Proposes, Does";
  A plan must include the chief designer, team, project
  organization and resources (eg. computers).  Supporting skills
  and disciplines are essential in the respective product areas,
  eg. ergonometrics, acoustics, radiation, microprogramming, data
  bases, security, reliability.
  . an understanding of the design, design production (eg. CAD)
  processes, and manufacturing processes:  Learning curves apply
  to all processes!  The organization must be staffed with people
  who understand the product, the design process (CAD and
  management discipline) and the production introduction process.
  One or two out of three isn't enough.

behaviorally, the team must:

- do it right the first time: Being correct has the highest payoff everywhere: timeliness, quality, lack of rework, and mfg. cost.
- execute the project in a timely fashion: Virtually ALL of our projects are late because we start too late, don't get it done on time because some critical invention is required, take too long to get it introduced, etc. For the very long, very late projects, the failure is lack of planning, tools and organization. Finally, people burn out. This suggests we:
  - limit projects to two years by a small team. We often make an aggressive business plan, then hire the team. They then find out they have neither tools nor technology to do the project.
  - not predicate a project on scheduling inventions in the design, process and CAD areas. If we can't see how to do the work in 2 years, then let's not start the project! This means the product must be cut down to fit the tools, people and process. Advanced development is to insure that we can do development.

  *stay in A/D until a project can be scheduled*

  - have a written design methodology that includes: all design processes in the form of manuals, design conventions, conflict resolution, criteria for task completion, PERT structure, etc.;
- be open and have external reviews, and clearly written product descriptions for inspection: For new product areas, we require breadboards in addition to the above heuristics. When the product gestation time equals the generation time, a full advanced development effort is the only way to be successful.
- start small, be reviewed and grow on its demonstrated success;
- learn, in order to handle the increase in complexity that comes with technology. Until there's a formal sabbatical program, individuals would do well to consider taking the equivalent of a semester of technical courses each 10 years.

*How many courses have you taken?*

PRODUCT METRICS KNOWLEDGE includes:
- products for which there'll be no competitor;
- all product cost metrics (cost, cost of ownership, cost to operate and use);
- all product performance and cost/performance metrics; These are the goodness measures of a product and tell how easily it will be to sell, and if we have improved. Cost and performance is measured against a state-of-the-art line represented by the first shipment of a more advanced product. Alternatively, when there's no direct comparison, the time goodness is determined from the day the product could have shipped. For example, because of parts availability, Nebula and CT could have shipped two and three years ago based on component availability.
- reasons why the product will succeed against present and likely future competition; sure success in the market is to introduce a needed function (eg. 32-bit address) by which all other products have to be measured.
- major competitor products by cost, performance and functionality; This should cover the past and future five years.
- leading edge, innovative, small company products;
- productivity, quality and design process metrics for projects.

*Marketing = what you do when you have no products*

## DESIGN GOALS AND CONSTRAINTS

Design constraints are generally set as various kinds of standards. These are useful because they limit the choice of often trivial design decisions, and let us deal with important free choices, the goals. Goals are vitally important because they target our uniqueness.

Poor "mind-set" standards can create poor products, even though they may have made sense at one time. The historical English measures is a good case in point. Currently, the 19" rack and the metal boxes Digital makes to fit in them, and then ship on pallets to customers, act as constraints on building cost-effective PDP-11 Systems. This historical "mind set" standard often impedes the ability to produce products that meet the 20% per year cost decline curve.

- Goals and constraints must be written down and updated from the day the project starts. Virtually every product failure and period of product floundering is a result of no clear goals and constraints since everyone has a different idea of the product.
- A product can only have a few goals and constraints. The ranking is usually: it must work and have improved cost of ownership, be the shortest time to market, highest performance and lowest cost.

We must adhere to standards which we either follow or set!

- If a standard exists, follow it or change it for all! We lost the IEEE Floating Point format. It is likely we will eventually have to support it.
- If a standard is forming go all out to set it. When formed, then follow it. We didn't make DDCMP a standard. When HDLC came, we didn't use it. The result: expensive, low performance products.

Standards can be grouped into four distinct sets:

- DEC Engineering Standards; These cover most physical structures and design practice for producibility, and assimilate critical external standards, such as UL, VDE, and FCC.
- professional society, industry and area information processing standards, from EIA, CBEMA, ECMA, ANSI, ISO etc. such as Cobol '74, Codasyl, IEEE 488;
- defacto industry wide information processing and communication standards such as IBM SNA, Visicalc;
- standards implied by the architecture of existing DEC products to insure our customer software investments are preserved include:
  - architecture of computers, terminals, mass store and communications links; Our current ISP's include 6, 11's, 10/20, VAX, 8048, 8080, 8086, 68000; VT52, VT100, keyboards, Regis; MCP; HDLC, C1, N1, S1.
  - physical interconnect busses for computers and for interconnecting them CT, Q, U, N1, C1, etc. These insure that future system products can evolve from component and computer options between generations.
  - operating system interface file commands, command language, human interface, calling sequence, screen/form management, keyboard, etc.

- Products must be designed for easy translation into in any natural language since we are an international company.
- All products must have be customer installable and maintainable.
- Portability is an important goal. Personal computers must be portable! We must achieve this for all systems ASAP!

WHEN TO CREATE, WHEN TO EVOLVE AND WHEN TO STOP PRODUCTS
Engineering is responsible for designing evolutionary products in our markets AND for producing products that are natural to our tradition of supplying the most interactive, cost-effective computing. If a new product such as personal computing emerges and we do not have a product, engineering has failed, independent of being asked for it!

Given all the constraints, can we ever create a new product, or is everything just an evolutionary extension of the past? If revolutionary do we know or care where product ideas come from? The important aspect about product ideas is:
- Ideas must exist to have products! If we don't have ideas to redefine or extend a market, then we should not build a product.

It is hard to determine whether something is an evolution or just an extension. The critically successful products are likely to occur the second time around. Some examples: PDP 6,KA10,KI10,KL10,2080; Tops 10,Tenex,TOPS20; PDP5,8,8S,8I/L,8E/F/M; OS8-RT11; 11/20,40,34,44; RSX-A... M, M+; TSS-8,RSTS; various versions of Fortran, Cobol and Basic follow this; LA30,36,120; VT05,50/52,100, 101 etc.; RK05,RL01/2.
- A product tree MUST be maintained by each engineering group showing roots, gestation time and life.

Goodness and Greatness
All products whether they be revolutionary, creating a new base, or evolutionary, should:
- be elegant and high quality: Russ Doane's working definition is: "every feature contributes two benefits", like a double pun. Quality means no excess. Elegant, high quality designs, do double duty with a minimum use of resources. Quality is also the absence of errors, by being right the first time so that it doesn't have to be inspected or redone.
- offer at least a factor of two in terms of cost-effectiveness over a current product; we have classic failures because a CPU cost has been minimized, only to find the total system cost has barely changed 10% and the total cost to the customer is only 5% lower! If each product is unique then we will have funds to build good products.
- be based on an idea which will offer an attribute or set of attributes that no existing products have; For example, the goals and constraints for VAX included factor of two algorithm encoding and also offering ability to write a single program in multiple languages. VT100 got distinction by offering 132 columns and smooth scrolling.
- build in generality, and extensibility: historically we have not

been sufficiently able to predict how applications will evolve, hence generality and extensibility allow us and our customers to deal with changing needs. Extendable products also permit mid-life kickers to products. We have built several dead end products with the intent of lower product cost, only to find that no one wants the particular collection of options. In reality, even the $200 calculators offer a family of modular printer and mass storage options. For example, our 1-bit PDP-14 had no arithmetic ability, nor could it be a general purpose computer. As customers used it, ad hoc extensions were needed to count, compare, etc. and it finally evolved into a really poor, general purpose digital computer.

- be a complete system, not piece parts; The total system is what the user sees. A word processing system for example includes: memory, keyboard, tube, modems, cpu, documentation including how to unpack it, the programs, table (if there is one, if not then the method of using at the customer table), and shipping boxes.

- be a great system because the components are great; We should not depend on system markups and software functionality to cover poor components and high overhead.

- if we don't make it, buy it; We must carefully decide what components to make versus buy. It is very hard for an organization to be competitive without competing in the marketplace, hence unless we sell it, we should buy it.

## Product Evolution

A product family evolution is described on page 10 of Computer Engineering along the paths of lower cost, and relatively constant performance; constant cost and higher performance; and higher cost and performance. In looking at our successful evolutions:

- lower cost products require additional functionality too; A lower cost product, with constant performance or constant function is risky because a new customer base and new way of marketing may be required. Some other company may, however, be successful with the concept. The PDP-8, based on new technology, was radically more successful than its higher priced predecessor, the PDP-5, because it was 2/3 the price and 6 times more performance. The PDP-8/S was a failure at 2/3 the price and 15 less performance than the PDP-8. There are similar stories about the LA 34, VT50/52 and PDT as replacement products.

- constant cost, higher performance products are likely to be the most useful; Economics of use, the marketing channel and customer base are already established and a more powerful system such as the LA120 will allow higher productivity (see Computer Engineering for the understanding and economics). In the 11's there was a successful evolution: 20, 40, 34 anChied 44. Not the 60. The 11/70 was probably our greatest success; it was billed as a mid-life kicker to the 11/45-55.

## Revolutionary New Product Bases

- A new product base, such as a new ISP, physical interconnection, Operating System, approach to building Office Products, must

start a family tree from which significant evolution can occur.
The investment for a point product is so high that the product is
very likely not to payoff. In every case where we have
successful evolutionary products, the successors are more
successful than the first member of the family. Point products
with no follow-on will probably fail all roi tests.

Product Termination
- A product evolution is likely to need termination after
  successive implementations, because new concepts in use have
  obsoleted its underlying structure. All structures decay with
  evolution, and the trick is to identify the last member of a
  family, such as the 132 column card, and then not build it. This
  holds for physical components, processors, terminals, mass
  storage, operating systems, languages and applications. Some of
  the signs of product obsolescence:
    - It has been extended at least once, and future extensions
      render it virtually unintelligible.
    - Better products using other bases are available.

SELLING AND BUILDING THE PRODUCT
"Buy in" of the product can come at any time. However, if all the
other rules are adhered to, there is no guarantee that it will be
promoted, or that customers will find out about it and buy it. Some
rules about selling it:
- it has to be producible and work, AND be useful to software:
  This, although seemingly trivial rule, is often overlooked when
  explaining why a product is good or not. If it is a piece of
  hardware that requires software to support it, the hardware must
  be available to the programmers who must support it. Software
  engineers approach new hardware with much caution! The often
  ask: is it significant? is it needed? why isn't it compatible
  with the past? If a hardware is viewed with distrust by software
  engineers it may be met with the same distrust by customers!
- a business plan with orders and marketing plans from several
  marketing persons and groups needs to be in place; Just as it is
  unwise to depend on a single opinion in engineering for design
  and review, it is even more important that several different
  groups are intending to sell the product. Individual marketers
  are just as fallible as unchecked engineers. This rule can and
  must be violated for revolutionary products!
- never build a product for a single customer, although a
  particular customer may be used as an archetype user;
  predicating a product on one sale is the one sure way to fail!
  Paraphrasing a remark by former GM executive Charles Wilson: if
  it's good for General Motors, it may only be good for GM.
- it must be done in a timely fashion according to the committed
  schedule, price and functions as previously described;
- it must be understandable and easy to use. The small size,
  complete hardware books were the DEC trademark that established
  the minicomputer. We must revive these such that a particular
  user never need access more than one. Simplicity must be the

rule for our documentation.

What heuristics are missing?  What heuristics do you disagree with?

What heuristics could be removed? reordered?

Could I please have your feedback before this becomes a final draft?

3/15/82 Mon 8:47
GB3.S2.5