IT Productivity

- How to Plan it
- How to Quantify it
- How to Estimate it
- How to Measure it
- How to deliver it in practice

- One Hour session 10-11 am
- Dec 1 2010 London
- For XXXXXX
- By Tom Gilb, Gilb.com

Main Ideas

Do Not

Throw masses of *nice sounding* technology, in a 37-50 page document,

at an undefined problem of ´Productivity´

- with no consideration of their
- known and probable effects,
- their uncertainty,
- their costs, and
- their side effects

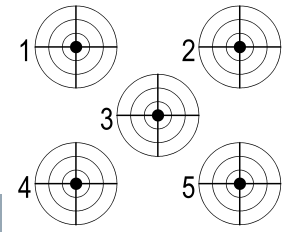
Do Well

1. Define Productivity Objectives clearly, numerically
2. Agree on these objectives
3. Select ´most effective´ strategies first
4. Stop selecting strategies when you have enough
5. Test and measure strategies evolutionarily
   - ◆ And start delivering real results now, this month and onwards

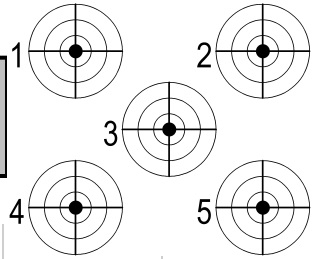
The Engineering Productivity Principles:

Here are some basic suggestions for a framework for getting control over engineering productivity:

1. **Subjective Productivity:** Productivity is someone's subjective opinion of what values we want to create for our critical stakeholders.
2. **Measurable Productivity:** Productivity can be defined as a set of quantified and measurable variables.
3. **Productivity Tools:** Productivity can be developed through individual competence and motivation, the way we organize people, and the tools we give them.
4. **Avoid Rework:** The initial attack on productivity improvement should be reduction of wasted effort
5. **Productive Output**: The next level of attack on productivity should be to improve the agreed value delivered to stakeholders.
6. **Infinite Improvement:** Productivity improvement can always be done: there are no known limits.
7. **Perfection Costs Infinity**: Increasing system performance towards perfection costs far more than increasing volume of system function.
8. **Value Varies:** Product attributes are viewed and valued quite differently even by members of the same stakeholder group.
9. **Practice Proves Productivity:** You cannot be sure how well a productivity improvement strategy will work until you try it in practice
10. **Productivity Dwindles:** Yesterday's winning productivity tactic may not continue to work as well forever.



Real (NON-CONFIDENTIAL version) example of an initial draft of setting the objectives that engineering processes must meet.



Business objective	Measure	Goal (200X)	Stretch goal ('0X)	Volume	Value	Profit	Cash
Time to market	Normal project time from GT to GT5	<9 mo	:6 mo	X		X	X
Mid-range	Min BoM for The Corp phone	<$9	3	X		X	X
Platformisation Technology	# of Technology 66 Lic. shipping > 3M/yr	4	6	X		X	X
Interface	Interface units	>11M	>13M	X		X	X
Operator preference	Top-3 operators issue RFQ spec The Corp			X		X	X
Productivity							X
Get Torden	Lyn goes for Technology 66 in Sep-04	Yes		X		X	X
Fragmentation	Share of components modified	<10%	<5%		X	X	X
Commoditisation	Switching cost for a UI to another System	>1y	>2yrs				
Duplication	The Corp share of 'in scope' code in best-selling device	>90%	>95%		X	X	X
Competitiveness	Major feature comparison with MX	Same	Better	X		X	X
User experience	Key use cases superior vs. competition	5	10	X	X	X	X
Downstream cost saving	Project ROI for Licensees	>33%	>66%	X	X	X	X
Platformisation IFace	Number of shipping Lic.	33	55	X		X	X
Japan	Share of of XXXX sales	>50%	>60%	X		X	X

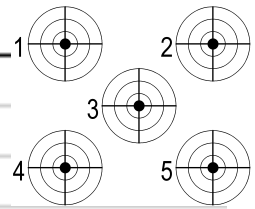
Numbers are intentionally changed from real ones

Business Objectives Quantified

Strategy Impact Estimation:
for a $100,000,000 Organizational Improvement Investment

Objectives

Technical Strategies



Viking Deliverables

Business Objective	hardware adaptation	Telephony	Reference designs	IFace	Modularity	Defend vs Technology 66	Tools	User Experience	GUI & Graphics	Security	Defend vs OCD	Enterprise
Time to market	20%	10%	30%	5%	10%	5%	15%	0%	0%	0%	5%	5%
Mid-range	15%	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	5%	10%	5%	5%	0%	0%
Platformisation Technology	25%	10%	30%	0%	[illegible]	10%	0%	5%	0%	10%	0%	5%
Interface	5%	15%	15%	0%	5%	0%	5%	0%	0%	10%	0%	10%
Operator preference	0%	[illegible]	[illegible]	[illegible]	[illegible]	20%	5%	10%	10%	20%	5%	10%
Get Torden	25%	10%	10%	-10%	0%	20%	0%	10%	-20%	10%	10%	5%
Commoditisation	20%	10%	20%	10%	-20%	25%	15%	0%	0%	5%	10%	5%
Duplication	15%	0%	10%	0%	0%	40%	0%	0%	0%	5%	20%	5%
Competitiveness	10%	15%	20%	0%	10%	20%	10%	10%	20%	10%	10%	10%
User experience	5%	[illegible]	0%	0%	[illegible]	[illegible]	0%	30%	10%	0%	0%	0%
Downstream cost saving	15%	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	10%	0%	0%	10%	5%
Platformisation IFace	10%	10%	20%	[illegible]	0%	20%	5%	[illegible]	[illegible]	0%	[illegible]	5%
Japan	10%	5%	20%	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0%
Contribution to overall result	15%	9%	17%	4%	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	5%
Cost (£M)	£ 2.85	£ 0.49	£ 3.21	£ 2.54	£ 1.92	£ 2.31	£ 0.81	£ 1.21	£ 2.68	£ 0.79	£ 0.62	£ 0.60
ROI Index (100=average)	106	358	109	[illegible]	[illegible]	[illegible]	148	107	10	152	202	174

Strategy Impacts on Objectives

"Benefits"

Cost

Benefit/Cost ratio

358 !

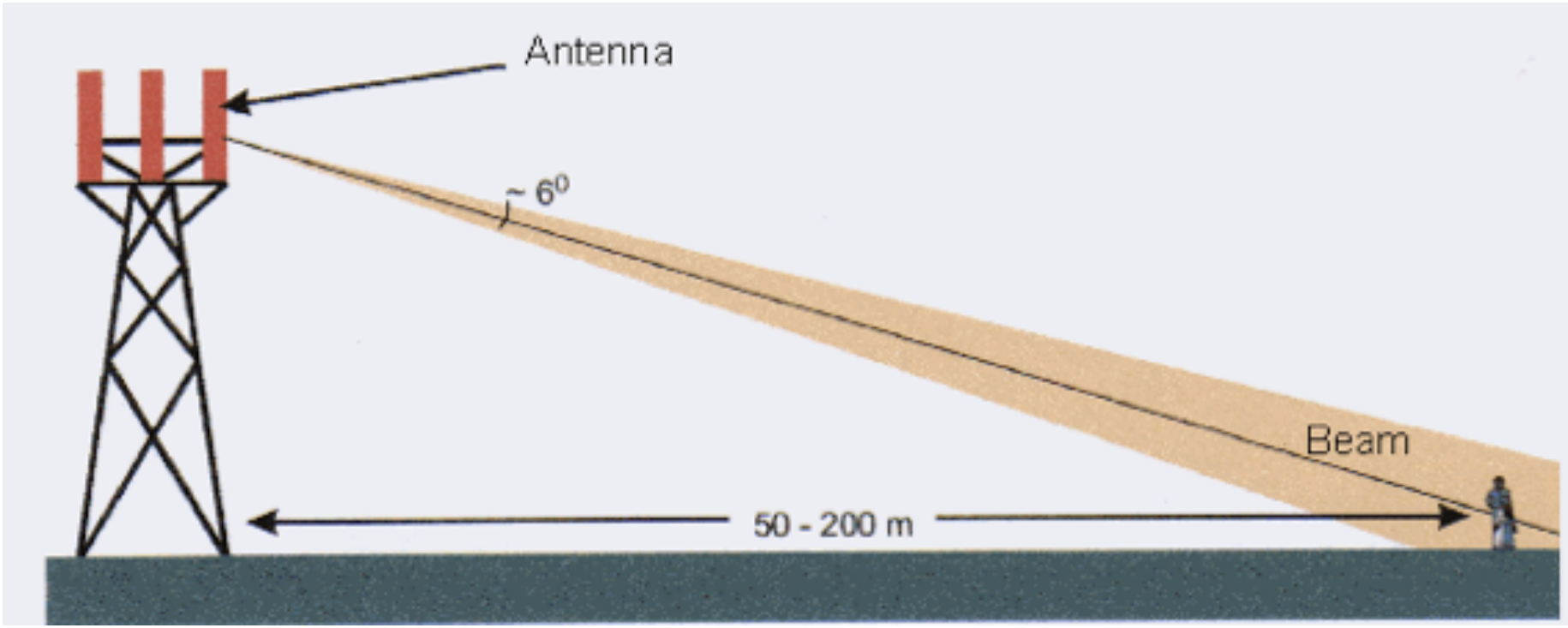
Software Engineering Productivity Study

ERICSSON

An example of setting objectives for software engineering process improvement

For 1997 with 70% software labor development content in products

Tom and Kai Gilb, Consultants to Ericsson ERA
CTO Thomas Ericsson



Main beam from a macrocell base station antenna

The problem

Great Market Growth Opportunities

Too Few Software Engineers

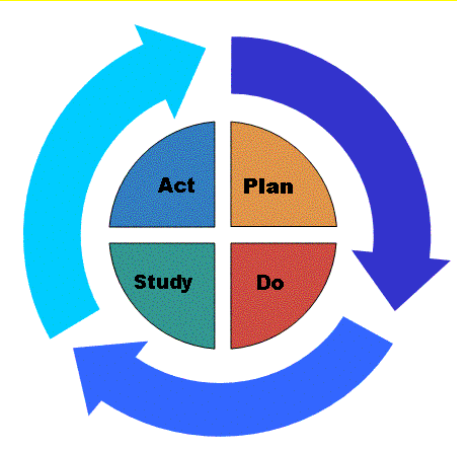
Solution:

Increase productivity of existing engineers

The One Page Top Management Summary (after 2 weeks planning)

The Dominant Goal

Improve Software Productivity in R PROJECT by 2X by year 2000



Dominant (META) Strategies

Continual Improvement (PDSA Cycles)
.DPP: Defect Prevention Process
.EVO: Evolutionary Project Management

Long Term Goal [1997-2000+]

DPP/EVO, Master them and Spread them on priority basis.

Short Term Goal [Next Weeks]

DPP [RS?]
EVO [Package C ?]

Decision: {Go, Fund, Support}

The Ericsson Quality Policy:

ERICSSON

"every company shall *define* **performance** indicators (which) ..

reflect **customer satisfaction**,

internal **efficiency**

and business **results**.

The performance indicators are used in **controlling** the operation."

Quality Policy [4.1.3]

Levels of Objectives.

Fundamental Objectives

Strategic Objectives

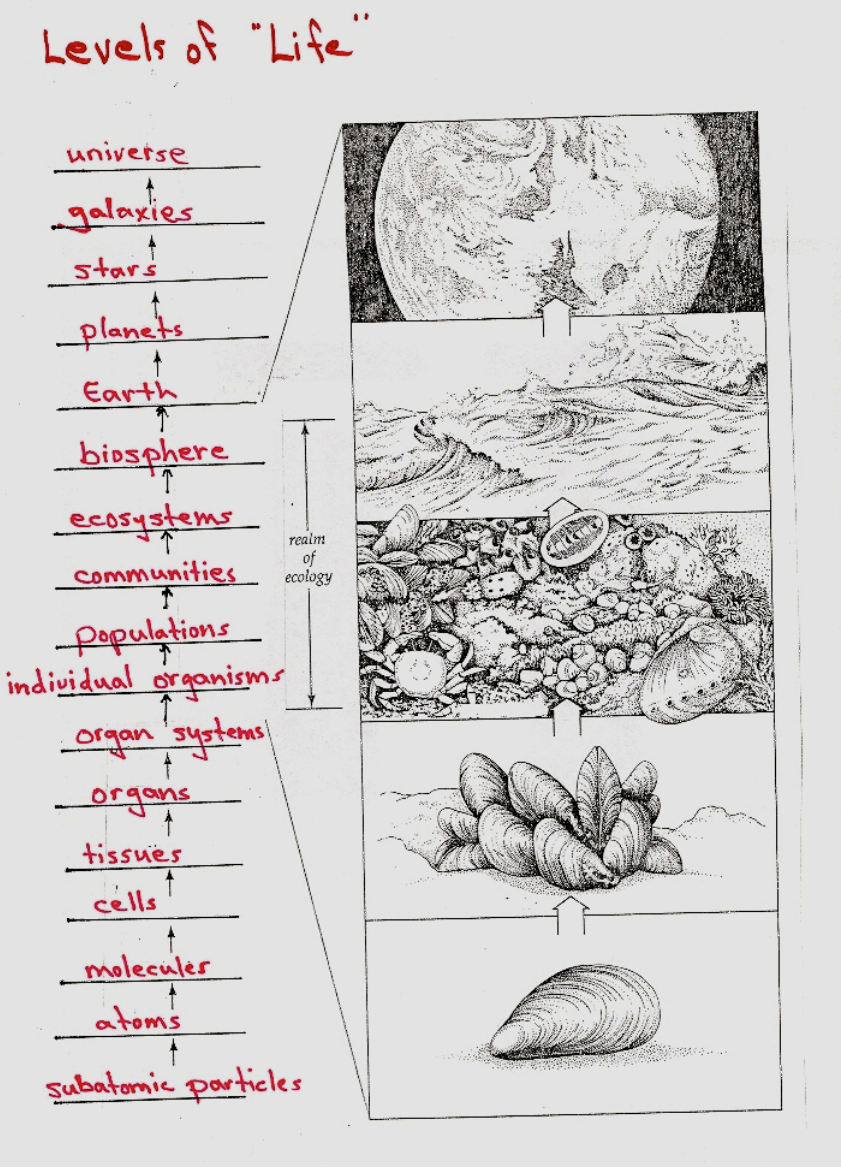
Means Objectives:

Organizational Activity Areas.

- Pre-study.
- Feasibility Study.
- Execution.
- Conclusion.

Generic Constraints

- Political Practical
- Design Strategy Formulation Constraints
- Quality of Organization Constraints
- Cost/Time/Resource Constraints



Keeney's: Levels of objectives.

1. Fundamental Objectives

(above us)

2. Generic Constraints

(our given framework)

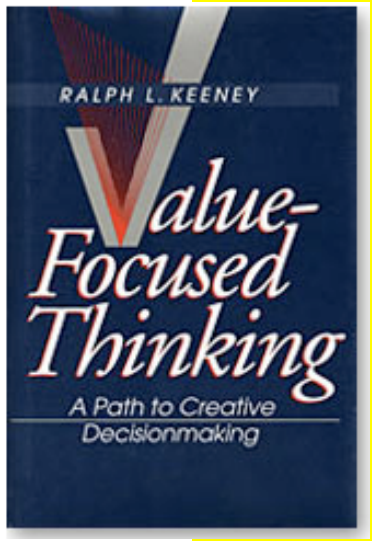
Political Practical

Design Strategy Formulation Constraints

Quality of Organization Constraints

Cost/Time/Resource Constraints

Constraints



3. Strategic Objectives

(objectives at our level)

4. Means Objectives:

(*supporting* our objectives)

The Strategic Objectives (CTO level)

Support

the **Fundamental** Objectives (Profit, survival)

Software Productivity:

Lines of Code Generation Ability

Lead-Time:

Predictability.

TTMP: Predictability of Time To Market:

Product Attributes:

Customer Satisfaction:

Profitability:

‘Means’ Objectives:

Support the **Strategic** Objectives

Complaints:
Feature Production:
Rework Costs:
Installation Ability:
Service Costs:
Training Costs:
Specification Defectiveness:
Specification Quality:
Improvement ROI:

"Let no man turn aside,
ever so slightly,
from the broad path of honour,
on the plausible pretence
that he is justified by the goodness
of his end.
All good ends can be worked out
by good means."
Charles Dickens

Strategies: (total brainstormed list) 'Ends for delivering Strategic Objectives'

Evo [Product development]:

DPP [Product Development Process]: Defect Prevention Process.

Inspection?

Motivation.Stress-Management-AOL

Motivation.Carrot

DBS

Automated Code Generation

Requirement -Tracability

Competence Management

Delete-Unnecessary -Documents

Manager Reward:?

Team Ownership:?

Manager Ownership:?

Training:?

Clear Common Objectives:?

Application Engineering area:

Brainstormed List (not evaluated or prioritized yet)?

Requirements Engineering:

Brainstormed Suggestions?

Engineering Planning:

Process Best Practices:

Brainstormed Suggestions?

Push Button Deployment:

Architecture Best Practices:

Stabilization:

World-wide Co-operation?

Principles for Prioritizing Strategies

Strategies must be well-defined

- **Not vague**

Strategies must have some relevant predictable numeric experience

- **On main effects**
- **Side effects**
- **Costs**
- **Risks - Uncertainty**

Not too big a spread of experience

"Software Productivity" =

Lines of Code Generation Ability

"Software Engineering net production in relation to corresponding costs."

Ambition: Net lines of code successfully produced per total working hours needed to produce them. A measure of the efficiency ('effective production/cost of production') of the organization in using its software staff

Scale: [Defined Volume, kNCSS or kPle

Software Development: Defined:

Productivity calculations include Work-Ho
Phase

Meter : <PQT Database and EPOS, CP

Comment: we know that real software
chosen this measure as it is available i

P1: Past [1997, ERA/AR] < to be calculated when data a ble Volume/Work Hours>

Past-R PROJECT: Past [1997, R PROJECT] < to be calculated when data available, available Volume/ Work Hours >

Past-EEI: Past [1997, Ireland, Plex] ___??__ kPLEX / Work-Hour.

<add more like LuleÂ>

Fail [end 1998, R PROJECT, Same Reliability] 1.5 x Past-R PROJECT <- R PROJECT AS 3 c " by 50%".

"50% better useful code productivity in 1.5 years overall"

Same Reliability: State: The Software Fault Density is not worse than with comparable productivity. Use official The Company Software Fault Density measures <- 1997 R PROJECT Balanced Scorecard (PA3).

Goal [Year=2000, R PROJECT, Same Reliability] 2 x Past-R PROJECT,

[Year=2005, RPL, Same Reliability] 10?? x Past-R PROJECT

Wish [Long term, vs. D pack.] 10 x Past-R PROJECT "times higher productivity" <- R PROJECT 96 1.1 c

Wish [undefined time frame] 1.5 x Past-R PROJECT <- R PROJECT AS 3 c " by 50%"

Comment: May 13 1997 1600, We have worked a lot on the Software Productivity objectives (all day) and are happy that it is in pretty good shape. But we recognize that it needs more exposure to other people.

Scale: [Defined Volume, kNCSS or kPlex] per Software Development Work-Hour.

Lead-Time:

Lead-Time:

"Months for major Packages"

***Ambition**: decrease months duration between major Base Station package release.*

Scale: Months from TG0, to successful first use for major work station package.

Note: let us make a better definition. TG

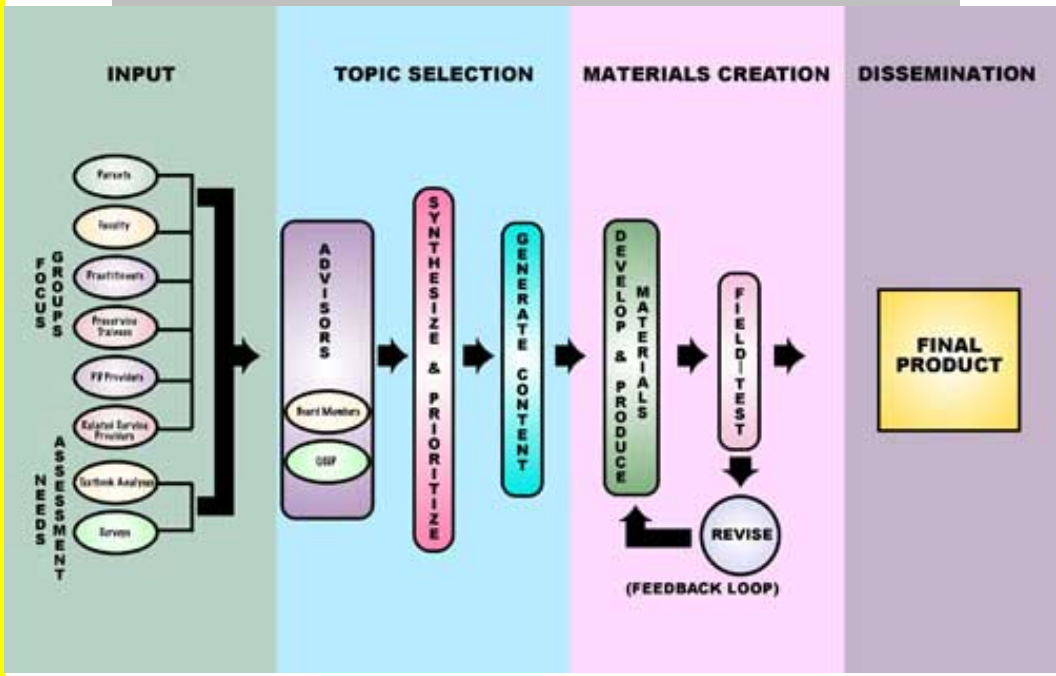
Past [C Package, 1996?] 20? Months?? <-guess tg

Goal [D-package] 18 months <- guess tg

Goal [E-package and later] 10.8 Months <- R PROJECT 96 1.1 a "40% > D"

Goal [Generally] ??? <- R PROJECT AS 3a

"10% Lead-Time reduction compared to any benchmark".



Predictability of Time To Market:

TTMP: Predictability of Time To Market:

Ambition: From Ideas created to customers can use it. Our ability to meet agreed specified customer and self-determined targets.

Scale: % overrun of actual Project Time compared to planned Project Time

Project Time: Defined: time from the date of Toll-Gate 0 passed, or other Defined Start Event, to, the Planned- or Actually- delivered Date of All [Specified Requirements], and any set of agreed requirements.

Specified Requirements: Defined: written approved Quality requirements for products with respect to Planned levels and qualifiers [when, where, conditions].
And, other requirements such as function, constraints and costs.

Meter: Productivity Project or Process Owner will collect data from all projects, or make estimates and put them in the Productivity Database for reporting this number.

Past [1994, A-package] < 50% to 100%> <- Palli K. guess.
[1994, B-package] 80% ?? <- Urban Fagerstedt and Palli K. guess

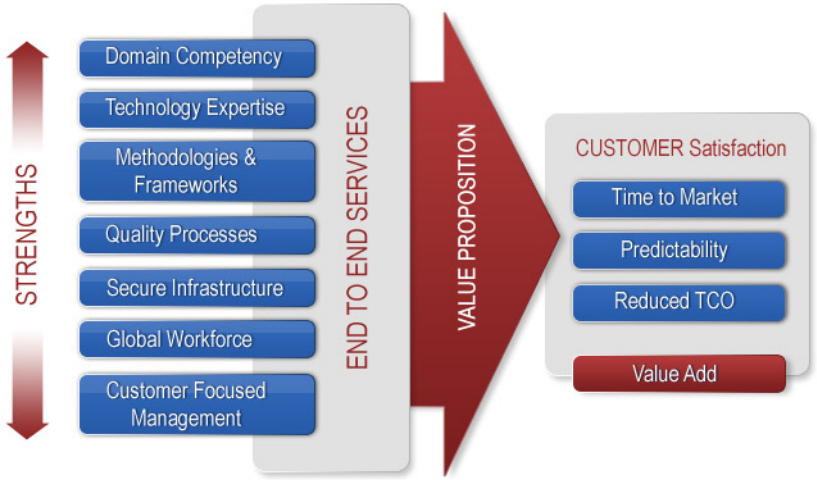
Record [IBM Federal Systems Division, 1976-80] 0% <- RDM 9.0 quoting Harlan Mills in IBM SJ 4-80

"all projects on time and under budget"

[Raytheon Defense Electronics, 1992-5] 0% <- RDE SEI Report 1995 Predictability.

Fail [All future projects, from 1999] 5% or less <- discussion level TG

Goal [All future projects, from 1999] 0% or less <- discussion level TG



Product Attributes:

Product Attributes:

"Keeping Product Promises."

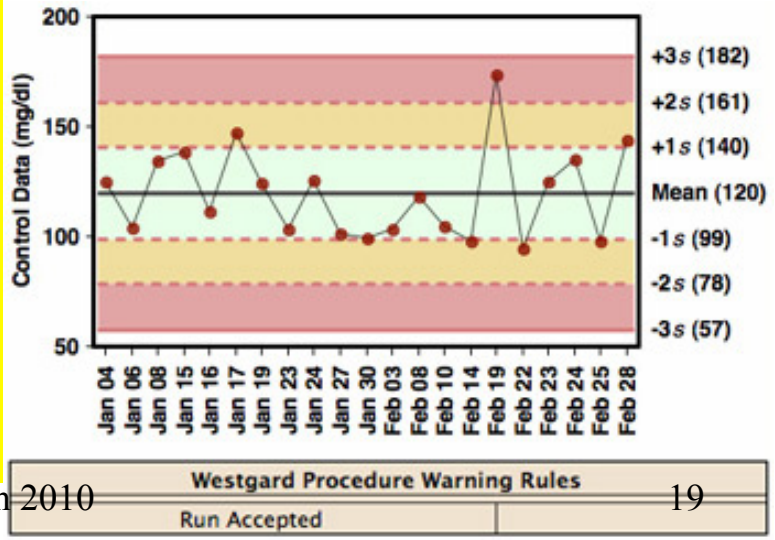
Ambition: Ability to meet or beat agreed targets, both cost, time and quality. (except TTMP itself, see above)

***Scale*: % +/- deviation from [defined agreed attributes with projects].**

***Past* [1990 to 1997, OUR DIVISION] at least 100% ???**

<- Guess. Not all clearly defined and differences not tracked. TSG

Goal [Year=2000, R PROJECT] **near 0%** negative deviation <- TsG for discussion.



Customer Satisfaction

Customer Satisfaction:
"Customer Opinion of Us"
Scale: average survey result on scale of 1 to 6 (best)
Meter: The Company Customer Satisfaction Survey
Past [1997] **4**
Goal [1998-9?] **5** <- R PROJECT 96 1.1 b

Profitability

Profitability:

"Return on Investment."

Ambition: Degree of saleable product ready for installation.

Scale: Money Value of Gross Income derived by [All R PROJECT Production OR defined products] for [Product Lifetime OR a defined time period]

Goal: <we did not complete this>

‘Means Objectives’ Samples
They use the
same *definition* process
as we use for the higher level objectives

Means Objectives

"support Strategic Objectives"

Summary:

'Means Objectives' are

not our major Strategic Objectives (ab

but each one represents areas which if improved

will normally help us achieve our Strategic Objectives.

Means Objectives have a lower priority than Strategic Objectives.

They must never be 'worked towards'

to the point where they reduce our ability to meet Strategic Objectives.

Complaints

Complaints:

"Customer complaint rate to us"

***Ambition**:*

Means Goal: for Customer Satisfaction (Strategic).

Scale: number of complaints per customer in [defined time into <operation>]

Past [Syracuse Project , 1997] ?? <bad> <- ML

Goal [Long term, software component, in first 6 months in Operation] **zero complaints** <- R PROJECT 96 1.1 b

"zero complaints on software features"

Impacts: <one or more strategic objectives>

Feature Production:

Feature Production:

"ability to deliver new features to customers"

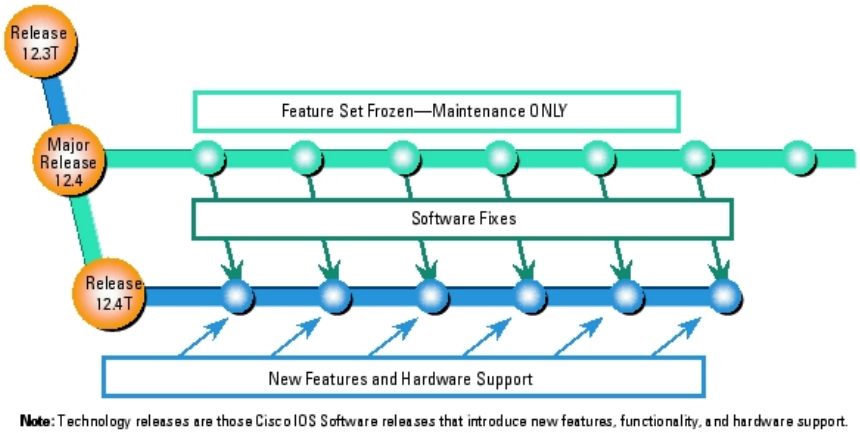
***Ambition**: reverse our decreasing ability to deliver new features <- R PROJECT AS 1.1*

Scale: Number of new prioritized <Features> delivered successfully to customer per year per software development engineer.

Too Little: **Past** [1997] ?? "estimate needed, maybe even definition of feature"

Goal [1998-onwards] **Too Little + 30% annually**?? <-For discussion purposes TsG.

"we need to drastically change our ability to effectively develop SW" <- R PROJECT AS 1.1



Note: Technology releases are those Cisco IOS Software releases that introduce new features, functionality, and hardware support.

Improvement ROI:

Improvement ROI:

"Engineering Process Improvement Profitability"

Ambition: Order of magnitude return on investment in process improvement.

Scale:

The average [annual OR defined time term] Return on Investment in Continuous Improvement as a ratio of [Engineering Hours OR Money]

Note: The point of having this objective is to remind us to think in terms of real results for our process improvement effort, and to remind us to prioritize efforts which give high ROI. Finally, to compare our results to others. <-TsG

Record

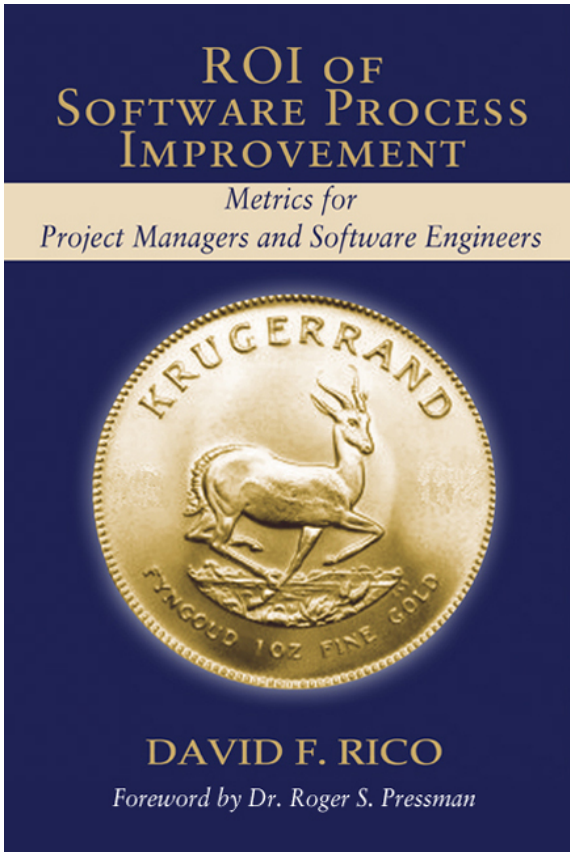
[Shell NL, Texas Instruments , Inspections] 30:1 <- Independently published papers TsG

Past

[IBM RTP, 1995, DPP Process] 13:1 <- Robert Mays, Wash DC test conference slides TsG

[Raytheon, 1993-5, Inspection & DPP] $7.70:1 <- RDE Report page 51 ($4.48 M$0.58M) Includes detail on how calculated. PK has copy.

[IBM STL, early 1990's] Average 1100% ROI (11:1) <- IBM Secrets pp32. PK has copy. NB Conservative estimate. See Note IBM ROI below.



2004

What should be *our* Productivity Objectives?

MEASURES OF IT PRODUCTIVITY
Some possibilities

At a client prospect

December 1st 2010

London

THE BRAINSTORMED SET
Measures of productivity

1. Environment management
    1. Automation
    2. Bureaucracy.
2. Time to deployment
3. Production Quality
4. Maintainability
5. Adaptability
6. Synergy.
7. Reuse.
8. Agility
9. Communication clarity
10. Developer autonomy Trust. <- Nick
11. Predictability

Environment management

Includes

- Automation
- Bureaucracy

Automation

Scale: % of potentially automated task time that is actually automated

Ideal 100%

Bureaucracy.

Scale: % of total effort due to defined Bureaucracy [Types]

Types:

- Rework
- Required Meetings
- Reporting

Time to Deployment

Scale: Time from defined [Start] to Successful Deployment [Type]

Type:

- Delivered: Value Delivered Initially and Proven
- Full: Full projected value is measured in place
- Lead: Leading indicators of success are experienced.
- Ready: the system is ready for deployment but other factors prevent actual implementation

Production Quality

Scale: Major Defect Density in defined [Stages]

Stages:

- Requirements
- Architecture
- Test Plans
- Released Systems
- Pervasive Systems

Maintainability

Scale: Calendar Time to Correctly Repair and Validate defined [Fault Types] using defined [Means]

Fault Types:

- Data Faults
- Logic Bugs
- Bad Test Plans
- Incorrect Management Presentations

Adaptability

Scale: Work Years needed to Successfully Complete and Implement defined [Change Types] using defined [Means]

Change Types:

- Legacy to Modern
- Data Integration
- Organizational Merger systems

Synergy.

Scale: not worked out yet, but we can do it!

Reuse.

Scale: not worked out yet, but we can do it!

Agility

Scale: not worked out yet, but we can do it!

One Investment Bank has quantified Agility Objectives extensively for their Agility Programme (November 2010) <-TsG

We can borrow some ideas from them.

Communication clarity

Scale: not worked out yet, but we can do it!

Developer autonomy Trust. <- Nick

Scale: not worked out yet, but we can do it!

Predictability

Scale: not worked out yet, but we can do it!

See the Ericsson Case Study

Rework <- TG

Main Ideas

Do Not

Throw masses of *nice sounding* technology, in a 37 page document,

at an undefined problem of ´Productivity´

- with no consideration of their
- known and probable effects,
- their uncertainty,
- their costs, and
- their side effects

Do Well

1. Define Productivity Objectives clearly, numerically
2. Agree on these objectives
3. Select ´most effective´ strategies first
4. Stop selecting strategies when you have enough
5. Test and measure strategies evolutionarily
   - ◆ And start delivering real results now this month and onwards

Last Slide