FUNCTION POINT METRICS: 25 YEARS OF PROGRESS

Capers Jones, Chief Scientist Emeritus

Software Productivity Research LLC
Web: www.SPR.com
email: cjones@spr.com

July 31, 2004

SOURCES OF MEASUREMENT INFORMATION

Enterprises providing data

• Fortune 500 companies	160
• Small-medium companies	120
• Federal Government agencies	20
• State-local Government agencies	10
• Universities	6
• Military services	3
• (Plus web sites and literature)	

MAJOR FUNCTION POINT USES CIRCA 2004

- **Function points are now a standard sizing metric.**
- **Function points are now a standard productivity metric.**
- **Function points are now a standard cost metric.**
- **Function points are now a standard quality metric.**
- **Function points are now a standard schedule metric.**
- **Function points are now a standard staffing metric.**
- **Function points are now a standard benchmark metric.**
- **Function points are used for outsource agreements.**
- **Function points are used for risk and value analysis.**

FUNCTION POINTS IN CONTEXT

Measures among SPR clients	Percent
• Financial measures	100%
• Software measures with function points	75%
• Sales and market measures	65%
• Warranty and Quality Measures	60%
• Shareholder measures	55%
• Balanced-scorecard measures	45%
• Supply-chain measures	25%

VALUE OF FUNCTION POINT ANALYSIS

- **Function point analysis costs about $1.50 per function point counted. What is the ROI of function point analysis?**
- **Projects with function point analysis at requirements have lower "requirements creep" than uncounted projects.**
- **Projects with function point analysis have about 15% lower cost overruns than uncounted projects.**
- **Projects with function point analysis have about 25% less schedule slip than uncounted projects.**
- **Preliminary analysis indicates that function point analysis saves $15 to $50 per function point. ROI = 15 to 1 roughly.**

WHERE FUNCTION POINTS ADD VALUE TO SOFTWARE

VALUE FACTORS	RESULTS
Defect removal improvement	50%
Scrap and rework reduction	50%
Maintenance cost reduction	40%
Poor technology acquisitions	40%
Defect prevention improvement	25%
Development cost reduction	25%
Development schedule reduction	25%
Paperwork reduction	20%

COST REDUCTION IS A KEY FUNCTION POINT STRENGTH

FUNCTION POINT VALUE OBSERVATIONS

Function Point Users		Companies Without	
On-time projects:	75%	On-time projects:	45%
Late projects:	20%	Late projects:	40%
Cancelled projects:	5%	Cancelled projects:	15%
Defect removal:	> 95%	Defect removal:	Unknown
Cost estimates:	Accurate	Cost estimates:	Optimistic
User satisfaction:	High	User satisfaction:	Low
Software status:	High	Software status:	Low
Staff morale:	High	Staff morale:	Low

Conclusion: Measurement and function points are valuable!

DEALING WITH VALUE MEASUREMENTS

• **Tangible Financial Value Factors**	
– **Cost Reduction**	**$ saved**
– **Direct Revenues**	**$ gained**
– **Indirect Revenues**	**$ gained**
– **User Effectiveness, Efficiency**	**$ saved**
• **Non- Financial Value Factors**	
– **User Satisfaction**	**Market share**
– **Employee Satisfaction, Morale**	**Turnover**
– **Competitive Advantages**	**Market share**
– **Human Life or Safety**	**?**
– **National Defense**	**?**
– **Enterprise Security**	**?**
– **Federal or regulatory mandates**	**?**
– **Enterprise Prestige**	**?**

DEALING WITH RISK MEASUREMENTS

• **Tangible Financial Risk Factors**	
– **Cancelled Projects**	**$ lost**
– **Cost Overruns**	**$ spent**
– **Schedule Delays**	**$ spent**
– **Poor Quality**	**$ spent**
– **Breach of Contract Litigation**	**$ spent**
• **Intangible Risk Factors**	
– **User Dissatisfaction**	**Market loss**
– **Employee Dissatisfaction**	**Turnover**
– **Poor quality**	**Market loss**
– **Patent, theft litigation**	**?**
– **Human Life or Safety Failures**	**?**
– **Federal or regulatory complaints**	**?**
– **Loss of Enterprise Prestige**	**?**

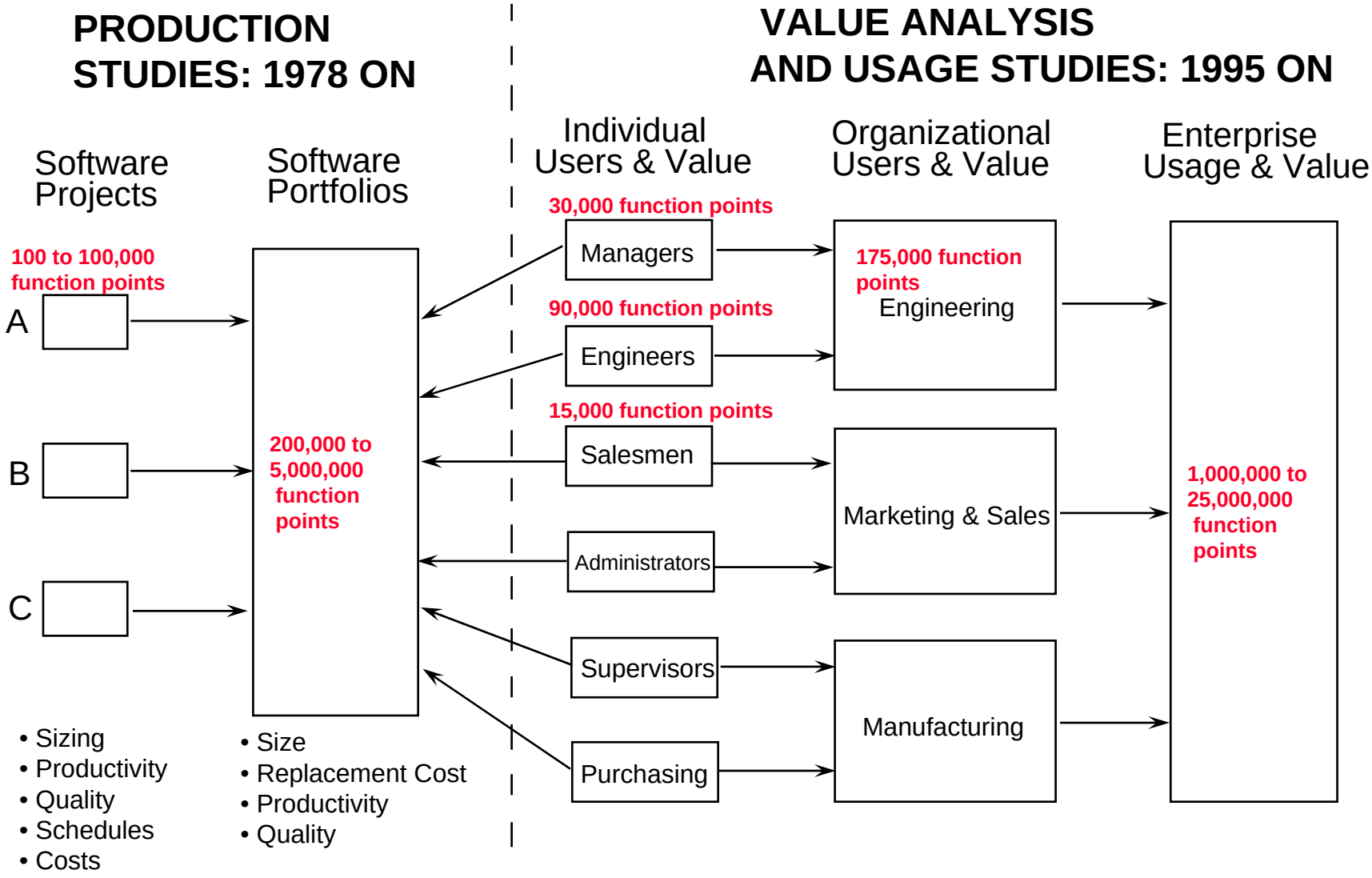
FUNCTION POINT VALUE ANALYSIS FACTORS

VALUE FACTOR	FUNCTION POINTS
Value to Cost Reduction	Useful
Value to Schedule Reduction	Useful
Value to Revenue Increase	Useful
Value to New Business Opportunities	Useful
Value to National Defense	Limited
Value to Shareholder Equity	Limited
Value to User Satisfaction	Limited
Value to Staff Morale	Limited
Value to Market Share	Not Used
Value to human life or safety	Not Used

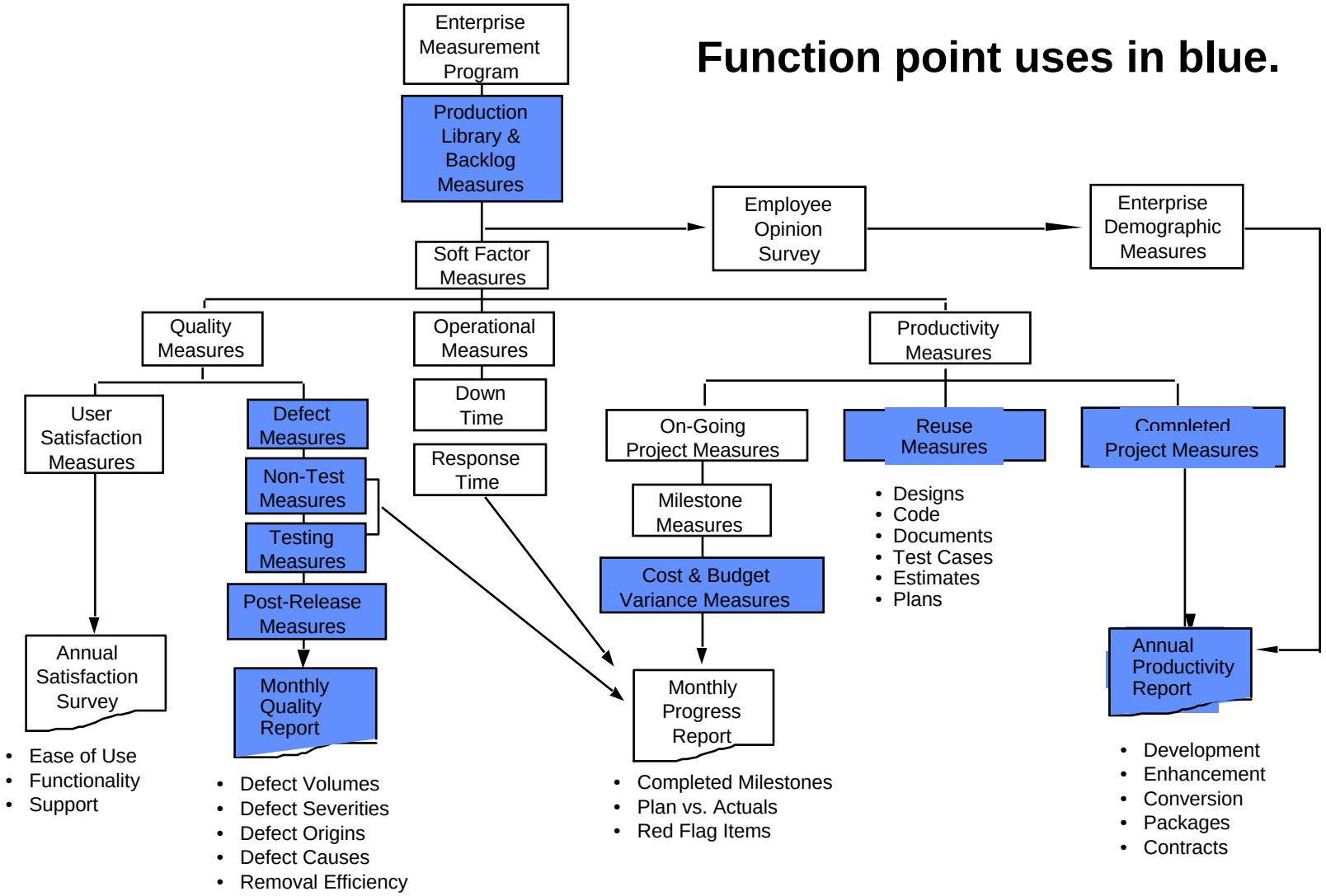
FUNCTION POINT RISK ANALYSIS FACTORS

RISK FACTORS	FUNCTION POINTS
Risk of Project Termination	Useful
Risk of Litigation for Breach of Contract	Useful
Risk of Tax Litigation	Useful
Risk of Unstable Requirements	Useful
Risk of Schedule Slip	Useful
Risk of Cost Overrun	Useful
Risk of Poor Quality	Useful
Risk of Morale Problems	Limited
Risk of Inadequate Development Processes	Limited
Risk of Staff Turnover	Not Used

FUNCTIONAL METRICS EVOLUTION



ENTERPRISE MEASURING AND REPORTING



MEASUREMENT WEB CITATIONS CIRCA 2004

Search-engine citations	Number of sites
1. Lines of code measures	8,110,000
2. IFPUG Function point metrics	6,680,000
3. Feature point metrics	6,350,000
4. Object point metrics	4,750,000
5. Benchmarks	2,890,000
6. Mark II function point metrics	210,000
7. COSMIC function point metrics	193,000

DIFFICULT MEASUREMENTS CIRCA 2004

- **Data base volumes or size**
- **Data quality**
- **Data cost of ownership**
- **Web content size**
- **Web content quality**
- **Intangible value**
- **Outsource measures and metrics**
- **Cancelled project measures**

HAZARDOUS MEASUREMENTS CIRCA 2004

Metrics and Measures Behaving Badly

• **Cost per Defect**	**(Penalizes quality)**
• **Lines of Code**	**(Ambiguous)**
• **Cost per Line of Code**	**(Penalizes new languages)**
• **Lines of Code per Month**	**(Ignores non-code work)**
• **Staff Work Hours per month**	**(Ignores non-work tasks)**
• **Industry averages**	**(Vague and ambiguous)**
• **Function point variants**	**(No benchmark data)**

METRICS AND MEASURES FOR INNOVATION

External Product Innovation

- Patents and inventions
- Research & Development spending
- Market share and New Market Growth

Internal Process Innovation

Time to Market	Process Assessments
Development costs	CMM Level > 3
Quality and Reliability	Industry Benchmarks
Customer satisfaction	Six-Sigma measures
$ per Function Point	Hours per Function Point

METRICS AND INDUSTRY LEADERSHIP

MEASURES WHERE INDUSTRY LEADERS OFTEN EXCEL:

- **Research and development**
- **Market shares and market growth**
- **Shareholder value**
- **Time to market***
- **Unit development costs of products***
- **Customer satisfaction**
- **Service and support***
- **Warranty repairs***
- **Staff morale**

(* = Function Point Usage)

FUNCTION POINTS AND COMPLEXITY CIRCA 2004

FUNCTION POINT COMPLEXITY IS DECOUPLED FROM:

- **COMBINATORIAL COMPLEXITY**
- **COMPUTATIONAL COMPLEXITY**
- **CYCLOMATIC COMPLEXITY**
- **ESSENTIAL COMPLEXITY**
- **FLOW COMPLEXITY**
- **SYNTACTIC COMPLEXITY**

FUNCTION POINTS ALONE ARE NOT ENOUGH

To support e-business and web development many metrics and measurement approaches are needed for software:

- **Accurate Effort, Cost, and Schedule Data**
- **Accurate Defect and Quality Data**
- **Accurate User-Satisfaction Data**
- **Source Code Volumes for All Languages**
- **Types and Volumes of Paper Documents**
- **Volume of Data, Information, and Web content**
- **Consistent and Reliable Complexity Information**

SOFTWARE METRICS USAGE IN 2004

Usage among SPR clients	Percent
• IFPUG function points	95%
• Backfiring (LOC to function points)	75%
• Lines of code (LOC) metrics	30%
• Other function points (COSMIC, Mark II, etc.)	10%

IFPUG function points #1 in overall usage in 2004

(International Function Point Users Group)

SOFTWARE PROJECTS MEASURED CIRCA 2004

Metrics Used for U.S. Benchmarks	Projects
• Backfiring (LOC to function points)	85,000
• IFPUG function points	45,000
• Lines of code (LOC) metrics	20,000
• Other function points (COSMIC, Mark II, etc.)	7,500

Backfiring #1 in volume of data in 2004

METRICS CONVERSION: A WEAK LINK IN 2004

Metrics Conversion is Ambiguous Between:

- **Physical lines of code and logical statements**
- **Logical statements and function points**
- **Physical lines and function points**
- **IFPUG Function points and British MK II function points**
- **IFPUG Function points and COSMIC function points**
- **IFPUG Function points and object points**
- **IFPUG complexity factors and cyclomatic complexity**
- **Calendar months and staff months**
- **Work hours, work days, work weeks, work months**
- **Current dollars and inflated dollars**
- **Burden rates when benchmarking cost data**

SOFTWARE MEASUREMENT STATUS CIRCA 2004

- **Fortune 500 companies with productivity measures: 30%**
- **Fortune 500 companies with quality measures: 45%**
- **Fortune 500 companies with complete measures: 15%**
- **Fortune 500 companies with missing measures: 85%**
- **Number of software measurement personnel 5,500**
- **Number of software projects measured: 160,000**
- **Number of software projects not measured 50,000,000**

We need to do much more than we have done!

SUCCESSFUL AND UNSUCCESSFUL PROGRAMS

SUCCESSFUL MEASUREMENT PROGRAMS

- Measurements are used as baselines for process improvement
- Measurements are used as benchmarks within industry
- Monthly reports of key measures to senior executives
- Measurements used as goals for improvement targets
- Annual productivity and quality report produced

UNSUCCESSFUL MEASUREMENT PROGRAMS

- Measurements not used for improvement programs
- Measurements not used for industry comparisons
- No reports of key measures to executives
- No goals for improvement targets
- No annual productivity and quality report produced

MEASUREMENTS AND SOFTWARE CLASSES

– Systems software	Best quality measurements
	Best software quality
– Information systems	Best productivity measurements
	Best use of function point metrics
– Outsource vendors	Best benchmark measurements
	Best baseline measurements
	Shortest delivery schedules
– Commercial software	Best user satisfaction measurements
	Best testing metrics
– Military software	Most SEI process assessments
	Best software reliability

IBM AND ITT MEASUREMENT EXPERIENCES

MEASUREMENT VALUE FACTORS	PERCENT
• Customer satisfaction on key products	+ 50%
• Software maintenance cost reductions	- 45%
• Software maintenance schedule reductions	- 35%
• Development schedule reductions	- 20%
• Development productivity rates	+ 15%
• Software staff morale	+ 15%

(Results from 4 years of software measurements)

ANNUAL "TAX" FOR SOFTWARE MEASUREMENTS

YEAR	ASSESSMENT	PRODUCTIVITY	QUALITY	TOTAL
Year 1	1.5%	2.0%	2.0%	5.5%
Year 2	1.0%	1.5%	1.5%	4.0%
Year 3	1.0%	1.5%	1.5%	4.0%
Year 4	1.0%	1.5%	1.5%	4.0%
Year 5	1.0%	1.0%	1.5%	3.5%

(Percentage of annual software staff budget)

ANNUAL ROI FOR SOFTWARE MEASUREMENTS

YEAR	ASSESSMENT	PRODUCTIVITY	QUALITY	TOTAL
Year 1	$1.25	$1.50	$1.75	$4.50
Year 2	$1.75	$2.00	$2.50	$6.25
Year 3	$2.50	$2.75	$3.50	$8.75
Year 4	$3.25	$3.25	$5.00	$11.50
Year 5	$5.00	$5.00	$8.00	$18.00

(Return for each $1.00 invested)

Measurement has an excellent ROI!

GOALS OF SOFTWARE MEASUREMENT

Apply metrics and measurement techniques that enable software projects to be managed and controlled with professional levels of performance.

Effective software measurement includes high levels of accuracy in determining quantitative factors:

- **Sizes of all deliverables**
- **Schedules of all activities**
- **Resources and costs expended**
- **Staffing levels of software specialists**
- **Defect levels and removal efficiency**
- **Demographics of software personnel**
- **Customer support and maintenance data**

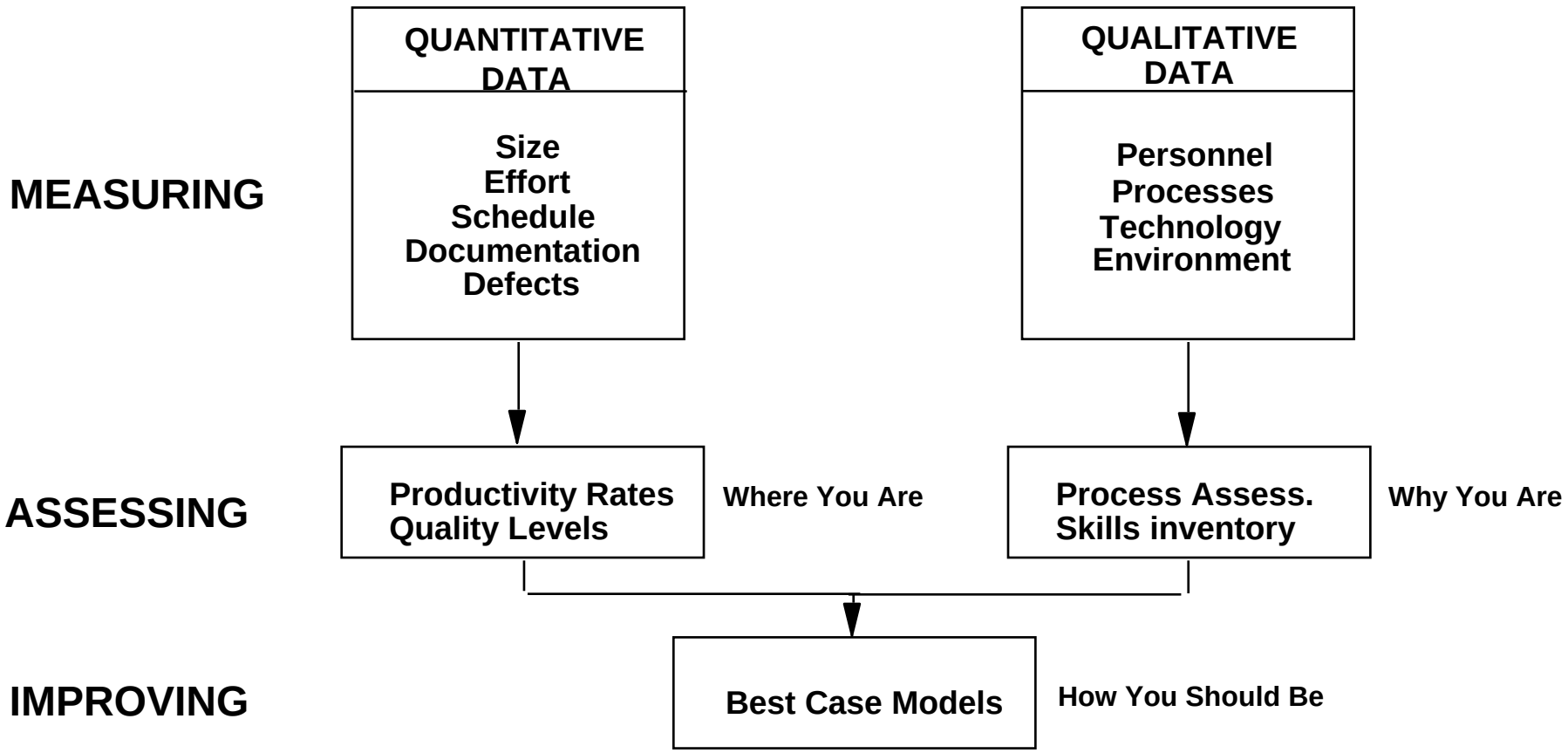
SUBJECTIVE SOFTWARE MEASUREMENT

Effective software measurement also includes important but subjective or qualitative factors:

- **SEI CMM or CMMI maturity level of project**
- **Staff specialization and experience levels**
- **Volatility of requirements change**
- **Value of processes, languages, and tools used**
- **Risk, value, and projected ROI of major projects**
- **User satisfaction after deployment**
- **Staff and management morale**

Source: Capers Jones, Applied Software Measurement McGraw Hill, 1996

QUANTITATIVE AND QUALITATIVE MEASUREMENTS



FUNDAMENTAL SOFTWARE PROBLEMS

- PROJECT MANAGERS ARE NOT WELL TRAINED
- HISTORICAL PRODUCTIVITY DATA STILL SPARSE IN 2004
- HISTORICAL QUALITY DATA REMAINS VERY SPARSE
- PROJECT MANAGEMENT TOOLS ARE NOT WELL USED
- QUALITY CONTROL IS OFTEN INADEQUATE
- REQUIREMENTS GROW AT AN AVERAGE OF 2% PER MONTH
- TECHNOLOGY CLAIMS EXCEED EMPIRICAL DATA

THE 10 MOST SERIOUS SOFTWARE RISK FACTORS

1) Inaccurate Metrics
2) Inadequate Measurements
3) Excessive Schedule Pressure
4) Canceled projects
5) Inaccurate Cost Estimating
6) Low Productivity
7) Creeping User Requirements
8) Low Quality
9) Silver bullet syndrome
10) Management disputes and malpractice

FUNCTION POINT RULES OF THUMB

- **Function points ^ 0.40 power = calendar months in schedule**
- **Function points ^ 1.15 power = pages of paper documents**
- **Function points ^ 1.20 power = number of test cases**
- **Function points ^ 1.25 power = software defect potential**
- **Function points / 150 = development technical staff**
- **Function points / 1,500 = maintenance technical staff**

NOTE: These rules assume IFPUG Version 4.1 counting rules.

U.S. SOFTWARE QUALITY CIRCA 2004

(Data expressed in terms of defects per function point)

Defect Origins	Defect Potential	Removal Efficiency	Delivered Defects
Requirements	1.00	77%	0.23
Design	1.25	85%	0.19
Coding	1.75	95%	0.09
Documents	0.60	80%	0.12
Bad Fixes	0.40	70%	0.12
TOTAL	5.00	85%	0.75

(Function points show all defect sources - not just coding defects)

BEST IN CLASS SOFTWARE QUALITY CIRCA 2004

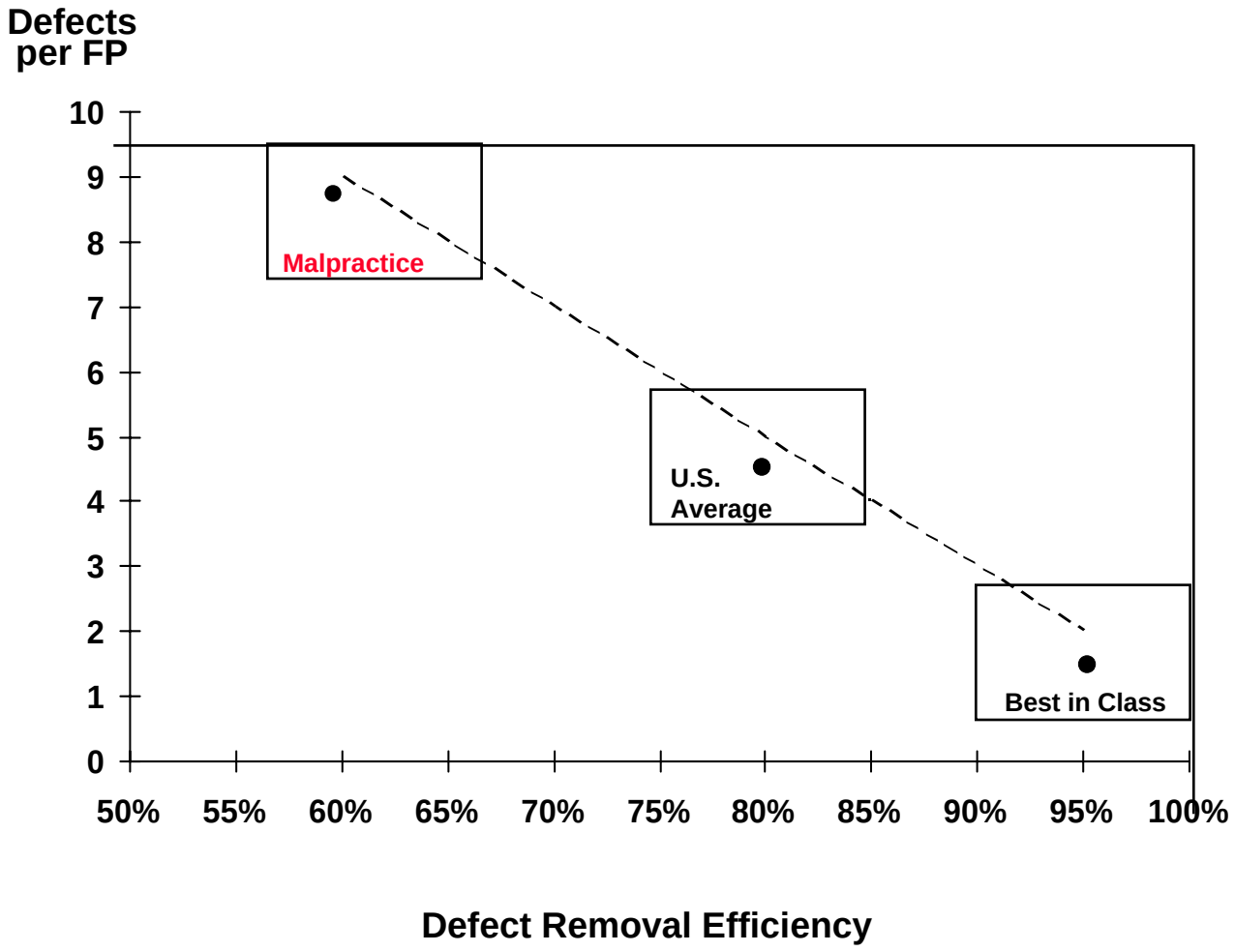
(Data expressed in terms of defects per function point)

Defect Origins	Defect Potential	Removal Efficiency	Delivered Defects
Requirements	0.40	85%	0.08
Design	0.60	97%	0.02
Coding	1.00	99%	0.01
Documents	0.40	98%	0.01
Bad Fixes	0.10	95%	0.01
TOTAL	2.50	96%	0.13

OBSERVATIONS

- **Most often found above Level 3 on the SEI CMM scale**
- **Most often found in systems and military software**

SOFTWARE QUALITY RANGES



TWELVE SOFTWARE METRICS CRITERIA

A Useful Software Metric Should:

1. **Be standardized**
2. **Be unambiguous**
3. **Have a formal user group**
4. **Have adequate published data**
5. **Have tools available for new projects**
6. **Have tools available for legacy projects**
7. **Have conversion rules for related metrics**
8. **Deal with all deliverables**
9. **Support all kinds of software**
10. **Support all programming languages**
11. **Support all sizes of applications**
12. **Support new + reused artifacts**

FUNCTION POINTS MEET MOST METRIC CRITERIA

The IFPUG Function Point Metric Meets 11 Criteria:

1. **Is standardized**
2. **Is unambiguous**
3. **Has a formal user groups**
4. **Has adequate published data**
5. **Has tools available for new projects**
6. **Has tools available for legacy projects**
7. **Has conversion rules for related metrics**
8. **Deals with all deliverables**
9. **Supports all kinds of software**
10. **Supports all programming languages**
11. **Counting inaccurate < 15 function points**
12. **Can be used with reused artifacts**

STRENGTHS AND WEAKNESSES OF FUNCTION POINTS

The main strengths of function point metrics are:

1. **Constant regardless of programming languages used**
2. **Function points are a good for full life-cycle analysis**
3. **Function points are a good for benchmarks**
4. **Function points are supported by software estimating tools**
5. **Function points can be converted into logical code statements**

The main weaknesses of function point metrics are:

1. **Counting requires certified function point specialists**
2. **Counting can be time-consuming and expensive**
3. **Counts are erratic for projects below 15 function points in size**
4. **Function point variations have no conversion rules (Mark II, COSMIC, Object Points, Engineering FP, etc.)**

TWELVE CRITERIA FOR MEASUREMENT SUCCESS

The Measurement Program Should:

1. Benefit the executives who fund it
2. Benefit the managers and staff who use it
3. Generate positive ROI within 12 months
4. Meet normal corporate ROI criteria
5. Be as accurate as financial data
6. Explain why projects vary
7. Explain how much projects vary
8. Link assessments with quantitative data
9. Support multiple metrics
10. Support multiple kinds of software
11. Support multiple activities and deliverables
12. Lead to improvement in software results

SEVEN MEASUREMENT HAZARDS

The Measurements Program Should Not:

1. Conceal the names of projects and units
2. Show only overall data without any details
3. Omit non-coding activities such as design
4. Omit “soft factors” that explain variances
5. Support only one metric such as LOC
6. Omit quality and show only productivity
7. Set ambiguous or impossible targets
    - 10 to 1 productivity improvement
    - 10 to 1 quality improvement
    - 50% improvement in 1 year

U.S. COSTS PER FUNCTION POINT CIRCA 2004

	Unburdened	Fully Burdened
Web Software	$150	$250
Information Systems Software	$600	$1,000
Outsource Software	$550	$1,500
Commercial Software	$1,000	$1,700
Systems Software	$1,200	$2,000
Military Software	$2,500	$5,000
Average	$1,000	$1,908

CREEPING REQUIREMENTS IN 2004

Domain	Average Monthly Rate of Creeping Requirements
Web software projects	4.0%
Commercial Software	3.5%
Information technology	2.5%
System, embedded software	2.0%
Military Software	2.0%
Outsourced Software	1.5%
AVERAGE	2.6%

POSSIBLE FUNCTION POINT EXPANSIONS

POTENTIAL BUSINESS METRICS

- **Function points -** **Measures software size**
- **Data points -** **Measures data base size**
- **Service points -** **Measures support size**
- **Engineering points -** **Measures hardware size**
- **Value points -** **Measures intangibles & ROI**
- **Content points -** **Measures web-site contents**

PROPOSED SUITE OF FUNCTIONAL METRICS

Function Points
- Inputs
- Outputs
- Inquiries
- Logical files
- Interfaces

Service Points
- Customers
- Inputs
- Outputs
- Inquiries
- Logical files
- Interfaces
- Constraints
- References

Data Points
- Entities
- Sets
- Attributes
- Interfaces
- Constraints

Engineering Points
- Algorithms
- Inventions
- References
- Feedback
- Constraints
- Inputs
- Outputs
- Interfaces
- Components

Value Points
- Time to market
- Cost reduction
- Revenue increase
- Market share
- Morale
- Health/Safety
- Risk reduction
- National security
- Mandates/statutes
- Customer satisfaction

Web Content Points
- Primary information
- Derivative information
- Nationalization
- Personalization
- Links
- Services
- Applets

RATIONALE FOR FUNCTION POINTS

- **Software personnel > 10% of total staffing in many large businesses.**
- **Software portfolios > 150,000 to 5,000,000 function points in many large businesses.**
- **Software is a critical corporate asset, and very hard to control.**
- **Function points can assist is solving many software business problems.**
- **Function points can assist in sizing, estimating, planning, and software quality control.**

RATIONALE FOR DATA POINTS

- **Businesses and government agencies own more data than they own software.**
- **There are no effective metrics for measuring data base size or for measuring data quality.**
- **Data is a critical corporate asset, and even harder to control than software.**
- **No current methods for economic studies of data mining, data warehousing, on-line analytical processing (OLAP), etc.**
- **Data base updates are being underestimated for Web and and E-business due to lack of metrics and lack of estimating tools that can handle data creation and modification.**

RATIONALE FOR ENGINEERING POINTS

- **Engineering personnel > 10% of total staffing in many large businesses.**
- **Many important products are hybrid and need both software and hardware engineering.**
- **Integrated estimation and measurement across the hardware software boundary is difficult.**
- **Function points and engineering points could lead to new forms of estimation and measurement tools.**

RATIONALE FOR SERVICE POINTS

- **The United States is becoming a service economy. Service personnel > 50% of employment in many enterprises.**
- **Many companies provide more services than manufactured goods.**
- **Services have no current metrics that can distinguish between knowledge-based services such as consulting or medical doctors and labor-based services such as clerical work.**
- **Service points would allow enterprises to perform economic analysis of many forms of service work.**

RATIONALE FOR CONTENT POINTS

- **Web "content" is a combination of graphics, text, services, and sounds. Content is a superset of data base information.**
- **No current metrics exist for measuring content size, quality, cost, schedules, or any other aspect. No estimating tools can handle content construction, modification, or removal.**
- **Content point metrics could integrate could bridge the gap between e-business and brick and mortar business.**
- **Function points and content points together would solve unique business problems.**
- **Economic studies of web-based applications and e-business require some form of content point metric.**

RATIONALE FOR VALUE POINTS

- **Business value is a combination of tangible financial value and intangible value. Most value metrics only assess the tangible financial portion (and don't do that very well).**
- **Software applications that improve employee morale or security need to be considered as well as applications that reduce costs.**
- **Value point metrics could integrate the tangible and intangible definitions of value.**
- **Value points would assist complex business problems.**

POTENTIAL INTEGRATED COST ANALYSIS

Unit of Measure	Size	Unit $	Total Costs
Function points	1,000	$500	$500,000
Data points	2,000	$300	$600,000
Content points	1,000	$100	$100,000
Service points	1,500	$250	$375,000
Engineering points	1,500	$700	$1,050,000
TOTAL	7,000	$375	$2,625,000
Value points	10,000	$1,000	$10,000,000
Net Value	3,000	$2,458	$7,375,000

SUMMARY AND CONCLUSIONS

- **The value of Measurement can only be shown by measurement!**
- **Good measurements yield positive ROI**
- **Bad measurements yield negative ROI**
- **Baselines are needed to track progress**
- **Assessments are needed to identify strengths and weaknesses**
- **Benchmarks are needed for industry comparisons**

Function Point Information Sources

Conflict and Litigation Between Software Clients and Developers
Jones, Capers; SPR; 2004

Models and Metrics in Software Quality Engineering
Kan, Stephen; Addison Wesley; 2003

IT Measurement: Practical Advice from the Experts
IFPUG; Addison Wesley; 2002

Function Point Analysis: Measurement Practices
For Successful Software Projects
Garmus, David and Herron, David; Addison Wesley; 2001

Software Assessments, Benchmarks, and Best Practices
Jones, Capers; Addison Wesley; 2000

Function Point Information Sources

Estimating Software Costs
Jones, Capers; McGraw Hill, 1998

Why We Should Use Function Points
Furey, S., Journal of Systems and Software; 1997; 14 (2)

Applied Software Measurement
Jones, Capers; McGraw Hill, 1996

Measuring the Software Process:
A Guide to Functional Measurement
Garmus, David and Herron, David; Prentice Hall, 1995

Reliability of Function Point Measurement:
A Field Experiment
Kemerer, C.F.; Comm. Of ACM 36(2); 1993

Function Point Information Sources

Function Point Analysis in the Estimation and Evaluation Of the Software Process
Low, G.C. and Jeffrey, D.R.; IEEE Trans. Of Soft. Eng. 16(1) 1990

Function Point Analysis
Dreger, J.B.; Prentice Hall, 1989

Software function, source lines of code, and development effort Prediction: a software science validation
Albrecht, A.J. and Gaffney J.E.; IEEE Trans. On Soft. Eng. (9)6); 1983

Measuring Application Development Productivity
Albrecht, A.J.; IBM/Share/Guide Symposium; 1979