FUNCTION POINT METRICS: 
25 YEARS OF PROGRESS

Capers Jones, Chief Scientist Emeritus
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

<table>
<thead>
<tr>
<th>Measures among SPR clients</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Financial measures</td>
<td>100%</td>
</tr>
<tr>
<td>• Software measures with function points</td>
<td>75%</td>
</tr>
<tr>
<td>• Sales and market measures</td>
<td>65%</td>
</tr>
<tr>
<td>• Warranty and Quality Measures</td>
<td>60%</td>
</tr>
<tr>
<td>• Shareholder measures</td>
<td>55%</td>
</tr>
<tr>
<td>• Balanced-scorecard measures</td>
<td>45%</td>
</tr>
<tr>
<td>• Supply-chain measures</td>
<td>25%</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>VALUE FACTORS</th>
<th>RESULTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defect removal improvement</td>
<td>50%</td>
</tr>
<tr>
<td>Scrap and rework reduction</td>
<td>50%</td>
</tr>
<tr>
<td>Maintenance cost reduction</td>
<td>40%</td>
</tr>
<tr>
<td>Poor technology acquisitions</td>
<td>40%</td>
</tr>
<tr>
<td>Defect prevention improvement</td>
<td>25%</td>
</tr>
<tr>
<td>Development cost reduction</td>
<td>25%</td>
</tr>
<tr>
<td>Development schedule reduction</td>
<td>25%</td>
</tr>
<tr>
<td>Paperwork reduction</td>
<td>20%</td>
</tr>
</tbody>
</table>

**COST REDUCTION IS A KEY FUNCTION POINT STRENGTH**
# FUNCTION POINT VALUE OBSERVATIONS

<table>
<thead>
<tr>
<th>Function Point Users</th>
<th>Companies Without</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-time projects: 75%</td>
<td>On-time projects: 45%</td>
</tr>
<tr>
<td>Late projects: 20%</td>
<td>Late projects: 40%</td>
</tr>
<tr>
<td>Cancelled projects: 5%</td>
<td>Cancelled projects: 15%</td>
</tr>
<tr>
<td>Defect removal: &gt; 95%</td>
<td>Defect removal: Unknown</td>
</tr>
<tr>
<td>Cost estimates: Accurate</td>
<td>Cost estimates: Optimistic</td>
</tr>
<tr>
<td>User satisfaction: High</td>
<td>User satisfaction: Low</td>
</tr>
<tr>
<td>Software status: High</td>
<td>Software status: Low</td>
</tr>
<tr>
<td>Staff morale: High</td>
<td>Staff morale: Low</td>
</tr>
</tbody>
</table>

**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 ?
<table>
<thead>
<tr>
<th>VALUE FACTOR</th>
<th>FUNCTION POINTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value to Cost Reduction</td>
<td>Useful</td>
</tr>
<tr>
<td>Value to Schedule Reduction</td>
<td>Useful</td>
</tr>
<tr>
<td>Value to Revenue Increase</td>
<td>Useful</td>
</tr>
<tr>
<td>Value to New Business Opportunities</td>
<td>Useful</td>
</tr>
<tr>
<td>Value to National Defense</td>
<td>Limited</td>
</tr>
<tr>
<td>Value to Shareholder Equity</td>
<td>Limited</td>
</tr>
<tr>
<td>Value to User Satisfaction</td>
<td>Limited</td>
</tr>
<tr>
<td>Value to Staff Morale</td>
<td>Limited</td>
</tr>
<tr>
<td>Value to Market Share</td>
<td>Not Used</td>
</tr>
<tr>
<td>Value to human life or safety</td>
<td>Not Used</td>
</tr>
</tbody>
</table>
# FUNCTION POINT RISK ANALYSIS FACTORS

<table>
<thead>
<tr>
<th>RISK FACTORS</th>
<th>FUNCTION POINTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk of Project Termination</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Litigation for Breach of Contract</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Tax Litigation</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Unstable Requirements</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Schedule Slip</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Cost Overrun</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Poor Quality</td>
<td>Useful</td>
</tr>
<tr>
<td>Risk of Morale Problems</td>
<td>Limited</td>
</tr>
<tr>
<td>Risk of Inadequate Development Processes</td>
<td>Limited</td>
</tr>
<tr>
<td>Risk of Staff Turnover</td>
<td>Not Used</td>
</tr>
</tbody>
</table>
FUNCTIONAL METRICS EVOLUTION

PRODUCTION STUDIES: 1978 ON

- Software Projects
  - 100 to 100,000 function points
  - A

- Software Portfolios
  - 200,000 to 5,000,000 function points
  - B

VALUE ANALYSIS AND USAGE STUDIES: 1995 ON

- Individual Users & Value
  - 30,000 function points
  - Managers
  - 90,000 function points
  - Engineers
  - 15,000 function points
  - Salesmen

- Organizational Users & Value
  - 175,000 function points
  - Engineering
  - Managers

- Enterprise Usage & Value
  - 1,000,000 to 25,000,000 function points

- 100 to 100,000 function points
- 200,000 to 5,000,000 function points
- 30,000 function points
- 90,000 function points
- 15,000 function points
- 175,000 function points

- Sizing
- Productivity
- Quality
- Schedules
- Costs

- Size
- Replacement Cost
- Productivity
- Quality

- Managers
- Engineers
- Salesmen
- Administrators
- Supervisors
- Purchasing
- Marketing & Sales
- Manufacturing
Function point uses in blue.
**MEASUREMENT WEB CITATIONS CIRCA 2004**

<table>
<thead>
<tr>
<th>Search-engine citations</th>
<th>Number of sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Lines of code measures</td>
<td>8,110,000</td>
</tr>
<tr>
<td>2. IFPUG Function point metrics</td>
<td>6,680,000</td>
</tr>
<tr>
<td>3. Feature point metrics</td>
<td>6,350,000</td>
</tr>
<tr>
<td>4. Object point metrics</td>
<td>4,750,000</td>
</tr>
<tr>
<td>5. Benchmarks</td>
<td>2,890,000</td>
</tr>
<tr>
<td>6. Mark II function point metrics</td>
<td>210,000</td>
</tr>
<tr>
<td>7. COSMIC function point metrics</td>
<td>193,000</td>
</tr>
</tbody>
</table>
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 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
<table>
<thead>
<tr>
<th>Usage among SPR clients</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IFPUG function points</td>
<td>95%</td>
</tr>
<tr>
<td>• Backfiring (LOC to function points)</td>
<td>75%</td>
</tr>
<tr>
<td>• Lines of code (LOC) metrics</td>
<td>30%</td>
</tr>
<tr>
<td>• Other function points (COSMIC, Mark II, etc.)</td>
<td>10%</td>
</tr>
</tbody>
</table>

IFPUG function points #1 in overall usage in 2004

(International Function Point Users Group)
## SOFTWARE PROJECTS MEASURED CIRCA 2004

<table>
<thead>
<tr>
<th>Metrics Used for U.S. Benchmarks</th>
<th>Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backfiring (LOC to function points)</td>
<td>85,000</td>
</tr>
<tr>
<td>IFPUG function points</td>
<td>45,000</td>
</tr>
<tr>
<td>Lines of code (LOC) metrics</td>
<td>20,000</td>
</tr>
<tr>
<td>Other function points (COSMIC, Mark II, etc.)</td>
<td>7,500</td>
</tr>
</tbody>
</table>

Backfiring #1 in volume of data 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

<table>
<thead>
<tr>
<th>MEASUREMENT VALUE FACTORS</th>
<th>PERCENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer satisfaction on key products</td>
<td>+ 50%</td>
</tr>
<tr>
<td>Software maintenance cost reductions</td>
<td>- 45%</td>
</tr>
<tr>
<td>Software maintenance schedule reductions</td>
<td>- 35%</td>
</tr>
<tr>
<td>Development schedule reductions</td>
<td>- 20%</td>
</tr>
<tr>
<td>Development productivity rates</td>
<td>+ 15%</td>
</tr>
<tr>
<td>Software staff morale</td>
<td>+ 15%</td>
</tr>
</tbody>
</table>

*(Results from 4 years of software measurements)*
## ANNUAL “TAX” FOR SOFTWARE MEASUREMENTS

<table>
<thead>
<tr>
<th>YEAR</th>
<th>ASSESSMENT</th>
<th>PRODUCTIVITY</th>
<th>QUALITY</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year 1</td>
<td>1.5%</td>
<td>2.0%</td>
<td>2.0%</td>
<td>5.5%</td>
</tr>
<tr>
<td>Year 2</td>
<td>1.0%</td>
<td>1.5%</td>
<td>1.5%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Year 3</td>
<td>1.0%</td>
<td>1.5%</td>
<td>1.5%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Year 4</td>
<td>1.0%</td>
<td>1.5%</td>
<td>1.5%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Year 5</td>
<td>1.0%</td>
<td>1.0%</td>
<td>1.5%</td>
<td>3.5%</td>
</tr>
</tbody>
</table>

(Percentage of annual software staff budget)
## ANNUAL ROI FOR SOFTWARE MEASUREMENTS

<table>
<thead>
<tr>
<th>YEAR</th>
<th>ASSESSMENT</th>
<th>PRODUCTIVITY</th>
<th>QUALITY</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year 1</td>
<td>$1.25</td>
<td>$1.50</td>
<td>$1.75</td>
<td>$4.50</td>
</tr>
<tr>
<td>Year 2</td>
<td>$1.75</td>
<td>$2.00</td>
<td>$2.50</td>
<td>$6.25</td>
</tr>
<tr>
<td>Year 3</td>
<td>$2.50</td>
<td>$2.75</td>
<td>$3.50</td>
<td>$8.75</td>
</tr>
<tr>
<td>Year 4</td>
<td>$3.25</td>
<td>$3.25</td>
<td>$5.00</td>
<td>$11.50</td>
</tr>
<tr>
<td>Year 5</td>
<td>$5.00</td>
<td>$5.00</td>
<td>$8.00</td>
<td>$18.00</td>
</tr>
</tbody>
</table>

(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
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

QUANTITATIVE AND QUALITATIVE MEASUREMENTS

MEASURING

QUANTITATIVE DATA
- Size
- Effort
- Schedule
- Documentation
- Defects

QUALITATIVE DATA
- Personnel
- Processes
- Technology
- Environment

ASSESSING

Productivity Rates
Quality Levels
Where You Are

Process Assess.
Skills inventory
Why You Are

IMPROVING

Best Case Models
How You Should Be
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)

<table>
<thead>
<tr>
<th>Defect Origins</th>
<th>Defect Potential</th>
<th>Removal Efficiency</th>
<th>Delivered Defects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>1.00</td>
<td>77%</td>
<td>0.23</td>
</tr>
<tr>
<td>Design</td>
<td>1.25</td>
<td>85%</td>
<td>0.19</td>
</tr>
<tr>
<td>Coding</td>
<td>1.75</td>
<td>95%</td>
<td>0.09</td>
</tr>
<tr>
<td>Documents</td>
<td>0.60</td>
<td>80%</td>
<td>0.12</td>
</tr>
<tr>
<td>Bad Fixes</td>
<td>0.40</td>
<td>70%</td>
<td>0.12</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>5.00</strong></td>
<td><strong>85%</strong></td>
<td><strong>0.75</strong></td>
</tr>
</tbody>
</table>

(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)

<table>
<thead>
<tr>
<th>Defect Origins</th>
<th>Defect Potential</th>
<th>Removal Efficiency</th>
<th>Delivered Defects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>0.40</td>
<td>85%</td>
<td>0.08</td>
</tr>
<tr>
<td>Design</td>
<td>0.60</td>
<td>97%</td>
<td>0.02</td>
</tr>
<tr>
<td>Coding</td>
<td>1.00</td>
<td>99%</td>
<td>0.01</td>
</tr>
<tr>
<td>Documents</td>
<td>0.40</td>
<td>98%</td>
<td>0.01</td>
</tr>
<tr>
<td>Bad Fixes</td>
<td>0.10</td>
<td>95%</td>
<td>0.01</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>2.50</strong></td>
<td><strong>96%</strong></td>
<td><strong>0.13</strong></td>
</tr>
</tbody>
</table>

**OBSERVATIONS**

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

Defects per FP

50% 55% 60% 65% 70% 75% 80% 85% 90% 95% 100%

Defect Removal Efficiency

Malpractice
U.S. Average
Best in Class

Copyright © 2004 by Capers Jones. All Rights Reserved.
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

<table>
<thead>
<tr>
<th>Software Type</th>
<th>Unburdened</th>
<th>Fully Burdened</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Software</td>
<td>$150</td>
<td>$250</td>
</tr>
<tr>
<td>Information Systems Software</td>
<td>$600</td>
<td>$1,000</td>
</tr>
<tr>
<td>Outsource Software</td>
<td>$550</td>
<td>$1,500</td>
</tr>
<tr>
<td>Commercial Software</td>
<td>$1,000</td>
<td>$1,700</td>
</tr>
<tr>
<td>Systems Software</td>
<td>$1,200</td>
<td>$2,000</td>
</tr>
<tr>
<td>Military Software</td>
<td>$2,500</td>
<td>$5,000</td>
</tr>
<tr>
<td>Average</td>
<td>$1,000</td>
<td>$1,908</td>
</tr>
</tbody>
</table>
# CREEPING REQUIREMENTS IN 2004

<table>
<thead>
<tr>
<th>Domain</th>
<th>Average Monthly Rate of Creeping Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web software projects</td>
<td>4.0%</td>
</tr>
<tr>
<td>Commercial Software</td>
<td>3.5%</td>
</tr>
<tr>
<td>Information technology</td>
<td>2.5%</td>
</tr>
<tr>
<td>System, embedded software</td>
<td>2.0%</td>
</tr>
<tr>
<td>Military Software</td>
<td>2.0%</td>
</tr>
<tr>
<td>Outsourced Software</td>
<td>1.5%</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>2.6%</td>
</tr>
</tbody>
</table>
POSSIBLE FUNCTION POINT EXPANSIONS

POTENTIAL BUSINESS METRICS

<table>
<thead>
<tr>
<th>Business Metric</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function points -</td>
<td>Measures software size</td>
</tr>
<tr>
<td>Data points -</td>
<td>Measures data base size</td>
</tr>
<tr>
<td>Service points -</td>
<td>Measures support size</td>
</tr>
<tr>
<td>Engineering points -</td>
<td>Measures hardware size</td>
</tr>
<tr>
<td>Value points -</td>
<td>Measures intangibles &amp; ROI</td>
</tr>
<tr>
<td>Content points -</td>
<td>Measures web-site contents</td>
</tr>
</tbody>
</table>
# PROPOSED SUITE OF FUNCTIONAL METRICS

<table>
<thead>
<tr>
<th>Function Points</th>
<th>Data Points</th>
<th>Value Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>Entities</td>
<td>Time to market</td>
</tr>
<tr>
<td>Outputs</td>
<td>Sets</td>
<td>Cost reduction</td>
</tr>
<tr>
<td>Inquiries</td>
<td>Attributes</td>
<td>Revenue increase</td>
</tr>
<tr>
<td>Logical files</td>
<td>Interfaces</td>
<td>Market share</td>
</tr>
<tr>
<td>Interfaces</td>
<td>Constraints</td>
<td>Morale</td>
</tr>
<tr>
<td>Service Points</td>
<td>Engineering Points</td>
<td>Health/Safety</td>
</tr>
<tr>
<td>Customers</td>
<td>Algorithms</td>
<td>Risk reduction</td>
</tr>
<tr>
<td>Inputs</td>
<td>Inventions</td>
<td>National security</td>
</tr>
<tr>
<td>Outputs</td>
<td>References</td>
<td>Mandates/statutes</td>
</tr>
<tr>
<td>Inquiries</td>
<td>Feedback</td>
<td>Customer satisfaction</td>
</tr>
<tr>
<td>Logical files</td>
<td>Constraints</td>
<td>Web Content Points</td>
</tr>
<tr>
<td>Interfaces</td>
<td>Inputs</td>
<td>Primary information</td>
</tr>
<tr>
<td>Constraints</td>
<td>Outputs</td>
<td>Derivative information</td>
</tr>
<tr>
<td>References</td>
<td>Interfaces</td>
<td>Nationalization</td>
</tr>
<tr>
<td></td>
<td>Components</td>
<td>Personalization</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Links</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Services</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Applets</td>
</tr>
</tbody>
</table>
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.
RATIONAL 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 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.
RATIONAL 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

<table>
<thead>
<tr>
<th>Unit of Measure</th>
<th>Size</th>
<th>Unit $</th>
<th>Total Costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function points</td>
<td>1,000</td>
<td>$500</td>
<td>$500,000</td>
</tr>
<tr>
<td>Data points</td>
<td>2,000</td>
<td>$300</td>
<td>$600,000</td>
</tr>
<tr>
<td>Content points</td>
<td>1,000</td>
<td>$100</td>
<td>$100,000</td>
</tr>
<tr>
<td>Service points</td>
<td>1,500</td>
<td>$250</td>
<td>$375,000</td>
</tr>
<tr>
<td>Engineering points</td>
<td>1,500</td>
<td>$700</td>
<td>$1,050,000</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>7,000</td>
<td>$375</td>
<td>$2,625,000</td>
</tr>
<tr>
<td>Value points</td>
<td>10,000</td>
<td>$1,000</td>
<td>$10,000,000</td>
</tr>
<tr>
<td><strong>Net Value</strong></td>
<td>3,000</td>
<td>$2,458</td>
<td>$7,375,000</td>
</tr>
</tbody>
</table>
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

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

Software function, source lines of code, and development effort Prediction: a software science validation

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