

N92-19432/

REUSE METRICS AND MEASUREMENT - A FRAMEWORK

Donald J. Reifer, President
Reifer Consultants, Inc.
Torrance, CA 90505

R-25902

Abstract: This presentation will describe the lessons learned and experience gleaned by those firms which have started to implement the reuse metrics and measurement framework prepared by the Joint Integrated Avionics Working Group (JIAWG) for use in controlling the development of common avionics and software for its affiliated aircraft programs (e.g., the Air Force's Advanced Tactical Fighter (ATF), the Army's LH helicopter and the Navy's A-12 fighter). The framework was developed to permit the JIAWG and Service System Program Offices (SPOs) to measure the long-term cost/benefits resulting from the creation and use of Reusable Software Objects (RSOs). The framework also monitors the efficiency and effectiveness of the JIAWG's Software Reuse Library (SRL).

The presentation will begin by defining the metrics and measurement framework which was established to allow the following six determinations and findings to be made relative to software reuse:

1. Impact of RSO creation on software cost and productivity.
2. Impact of RSO reuse on software cost and productivity.
3. Impact of RSO mining on software cost and productivity.
4. Minimum standards of quality for RSOs as they enter the SRL.
5. Efficiency and effectiveness of SRL usage.
6. Long-term cost/benefits of SRL usage.

The presentation will discuss how the following seven criteria were used to guide the establishment of the proposed reuse framework:

1. Compatible - The framework should be compatible with the software processes used by JIAWG contractors to develop avionics software products in Ada under DoD-STD-2167A.
2. Ease of Data Collection - The data needed to quantify the metric should be easy to collect and normalize.
3. Ease of Understanding - The metrics employed should be easy to understand, analyze and interpret.
4. Minimum Cost - The measurement costs (i.e., data collection, analysis and reporting) should be kept to a minimum.
5. Nonobtrusive - Collection of metrics data must not adversely impact the processes or products being measured.
6. Objective - It should be difficult to bias or distort the value of the metric.
7. Predictive - The metric should facilitate generation of accurate estimates of software cost, productivity and quality.

Next, object recapture and creation metrics will be explained along with their normalized use in effort, productivity and quality determination. A single and multiple reuse instance version of the popular COCOMO cost model will be presented which employs these metrics and the measurement scheme proposed by the Software Productivity Consortium (SPC) to predict the software effort and duration under various reuse assumptions. Investigations in using this model to predict actuals taken from the RCI database of over one thousand completed projects will be discussed along with statistical findings.

User experience with this metrics and measurement framework as part of the Air Force's Reusable Ada Avionics Software Package (RAASP) and Avionics Fault-Tolerant Software/Ada Technology Insertion Program (AFTS/ATIP) projects will be discussed next. The lessons learned with these metrics by these projects will be summarized. These two projects are conducting controlled experiments to capture measurement data that provides insight into those factors which impact software cost, quality, productivity and system performance. The RAASP effort is focusing on determining the relative impact of object-oriented methods, reuse paradigms and SRL operational policies software productivity, cost and quality. AFTS/ATIP is assessing the impact of a large number of process and product factors on overall cost and system performance.

The presentation will conclude with a summary of key points. Recommendations will be presented to help those embarking on a reuse program to improve their measurement and prediction capabilities.

**VIEWGRAPH MATERIALS
FOR THE
D. REIFER PRESENTATION**

RCI-TN-470

REUSE METRICS AND MEASUREMENT - A FRAMEWORK

28 November 1990

Prepared For:

NASA/Goddard Fifteenth Annual
Software Engineering Workshop



Reifer Consultants, Inc.

D. Reifer
RCI
Page 3 of 9

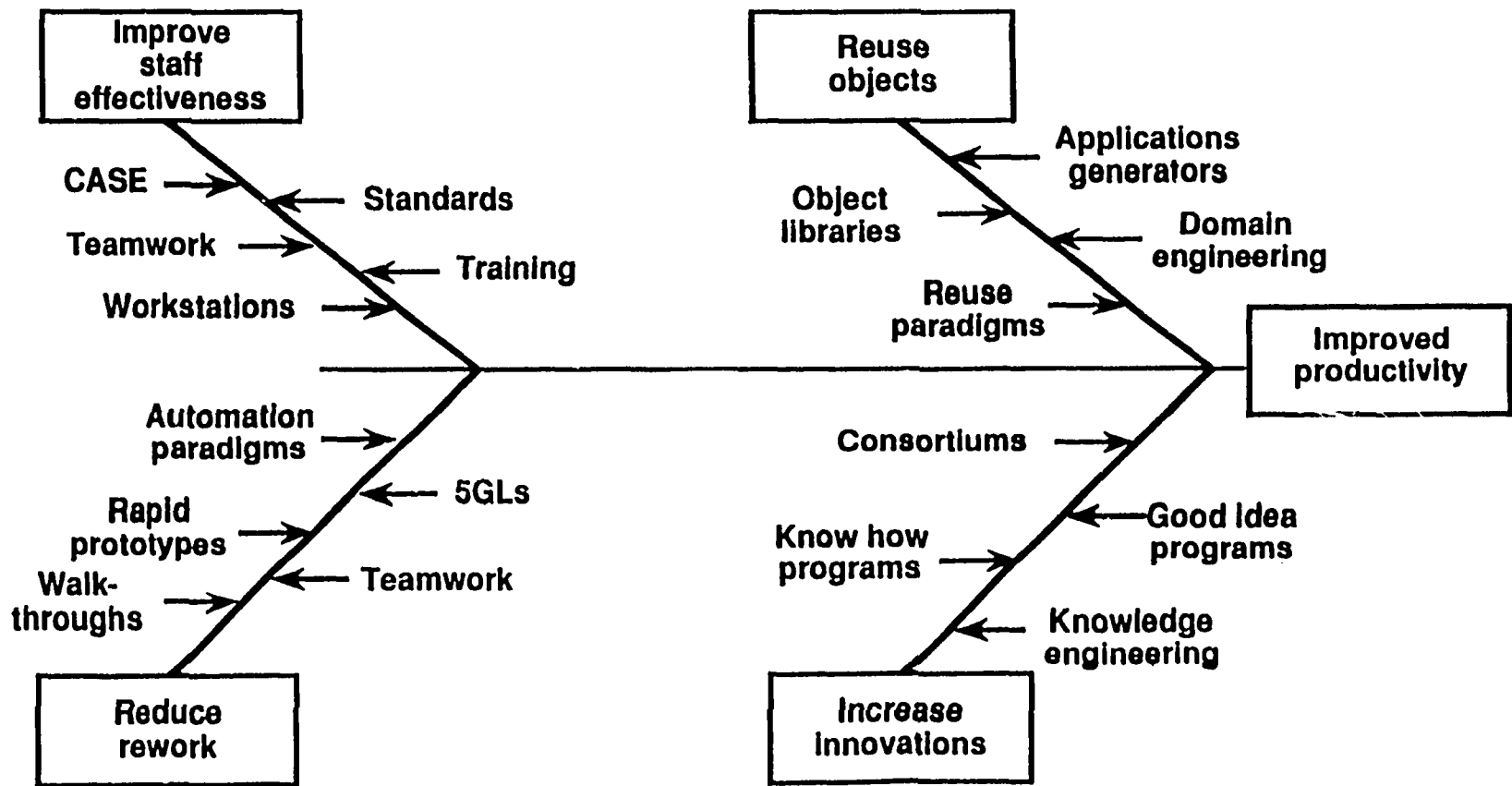
25550 Hawthorne Boulevard, Suite 112, Torrance, California 90505 / Phone: (213) 373-8720 / Fax: (213) 375-9845

PURPOSE

- Describe the reuse metrics and measurement framework created by JIAWG to make the following determinations:
 - Impact of RSO acquisition on software cost and productivity
 - Impact of RSO reuse on software cost and productivity
 - Minimum standards of quality for RSOs entering the Software Reuse Library (SRL)
 - Efficiency and effectiveness of SRL usage
 - Long-term cost/benefits of SRL usage
- Discuss implementation of the framework on the OSS and RAASP projects

Reusable Software Object (RSO) - life cycle products developed to be reused (designs, algorithms, code, tests/test cases, etc.)

PRODUCTIVITY IMPROVEMENT STRATEGIES



D. Rafter
RCI
Page 5 of 19

* Productivity must be measured from a quality viewpoint

BARRIERS TO REUSE

- Lack of incentives
- Few standards
- Limited tool support
- Champion needed
- Multiple quality levels
- NIH bias
- Needed infrastructure changes
- Few quantitative metrics

Source: RCI Reuse Survey, 8/8^o

METRICS SELECTION CRITERIA

Page 4

RCI-TN-470

- Compatible with DOD processes
- Ease of data collection
- Ease of understanding
- Minimum measurement cost
- Objective and unbiased
- Predictive of the future
- Unobtrusive as possible

D. Baker
RCI
Page 7 of 19

REUSE METRICS

OBJECT ACQUISITION RATIO

$$\text{OAR} = \sum_{i=1}^n (w_i) (a_i/A_i)$$

where: a_i = no. of RSOs acquired per collection

A_i = no. of objects in that collection

n = no. of collections

w_i = weighting factor for each collection

$$\text{and } \sum_{i=1}^n w_i = 1; a_i/A_i \geq 0$$

OBJECT REUSE RATIO

$$\text{ORR} = \sum_{i=1}^n (w_i) (r_i/R_i)$$

where: r_i = no. of reused objects in a collection

R_i = no. of objects in that collection

n = no. of collections

w_i = weighting factor for each collection

$$\text{and } \sum_{i=1}^n w_i = 1; r_i/R_i \geq 0$$

REUSE MECHANIZATION

Collection	W_i
• Requirements	0.20
• Design	0.30
• Source code	0.20
• Tests/test cases	0.30

$$\bullet \text{ OAR}_e = (0.2)(a_{1n} + a_{1r})/A_1 + (0.3)(a_{2n} + a_{2p} + a_{2r})/A_2 + (0.2)(a_{3n} + a_{3p} + a_{3r})/A_3 + (0.3)(a_{4n} + a_{4p} + a_{4r})/A_4$$

where: a_{xn} = newly created objects
 a_{xp} = purchased objects
 a_{xr} = recovered objects

$$\bullet \text{ ORR}_e = (0.2) r_1/R_1 + (0.3) r_2/R_2 + (0.2) r_3/R_3 + (0.3) r_4/R_4$$

where: r_x/R_x = reuse ratio for a collection

METRICS USAGE

REUSE VERSION COCOMO (SINGLE INSTANCE)
--

$$\bullet \text{ Effort}_r = c (1 + (\text{OAR}_x)(b_{18}) - (\text{ORR}_\theta)(b_{19})) \text{ Effort}$$

where: c = adjustment factor for domain

b_{18} = RSO cost factor ($0.10 < b_{18} < 0.36$)

b_{19} = RSO benefits factor ($0.20 < b_{19} < 0.60$)

OAR_x = expanded form of OAR

ORR_θ = effective form of ORR

Effort_r = cost in staff-months with reuse

Effort = cost in staff-months (COCOMO)

$$\bullet \text{ OAR}_x = (0.2)(a_{1n} + (0.5) a_{1r})/A_1 + (0.3)(a_{2n} + (0.2) a_{2p} + (0.4) a_{2r})/A_2 + (0.2)(a_{3n} + (0.2) a_{3p} + (0.5) a_{3r})/A_3 + (0.3)(a_{4n} + (0.3) a_{4p} + (0.6) a_{4r})/A_4$$

FACTOR RATINGS

	LOW	NOMINAL	HIGH	VERY HIGH	EXTRA HIGH
Reuse Cost Factor (b_{1g})	0.10	0.17	0.26	0.31	0.36
	Limited reuse packaging	Design and code RSO reuse packaging	Full RSO reuse packaging	Domain specific RSO reuse packaging	Extensive reuse packaging (synthesis)

	LOW	NOMINAL	HIGH	VERY HIGH	EXTRA HIGH
Reuse Benefits Factor (b_{1g})	0.20	0.25	0.34	0.48	0.60
	Planned reuse	Systematic reuse	Managed reuse	Institutionalized reuse (within and across jobs)	Optimized reuse (domain specific)

METRICS USAGE

SPC MODEL

$$\text{Cost} = (1-R) + R(B + E/N)$$

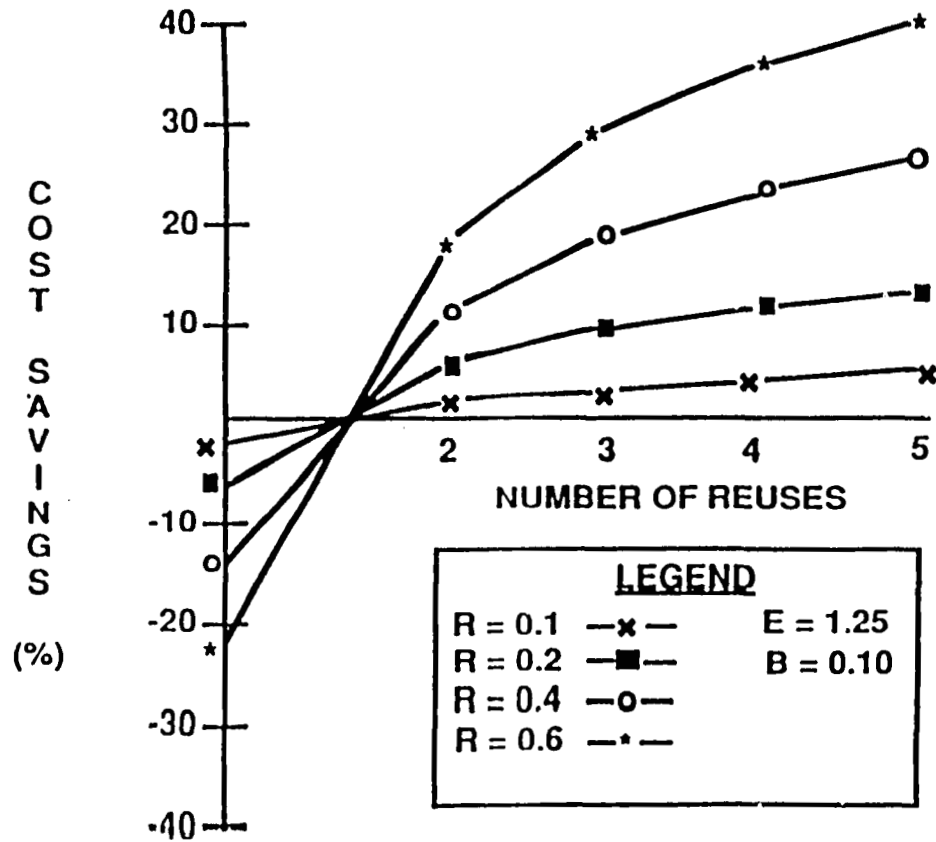
where: B = relative cost to reuse RSO
R = proportion of reused software
E = cost to develop RSO
N = number of reuses

REUSE COCOMO MODEL (MULTIPLE INSTANCES)

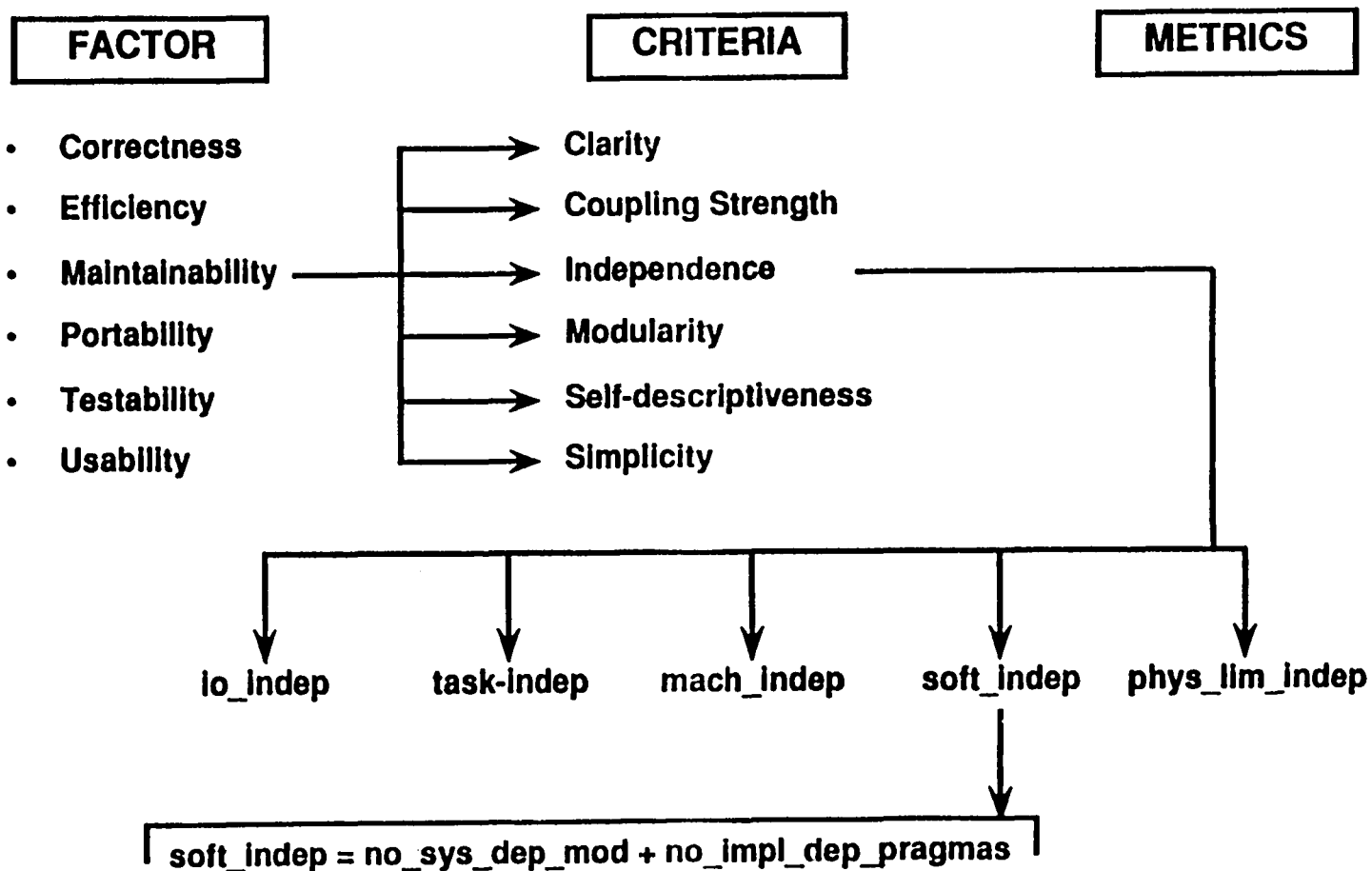
$$\text{Effort}_m = c(1 + \frac{(\text{OAR}_x)(b_{18})}{m} - (\text{ORR}_\theta)(b_{19})) \text{Effort}$$

where: c = calibration coefficient
m = number of reuses ($m > 1$)
 b_{18} = cost factor ($0.10 < b_{18} < 0.36$)
 b_{19} = benefit factor ($0.20 < b_{19} < 0.60$)
 OAR_x = Object Acquisition Ratio (average)
 ORR_θ = Object Reuse Ratio (average)

THE OSS EXAMPLE

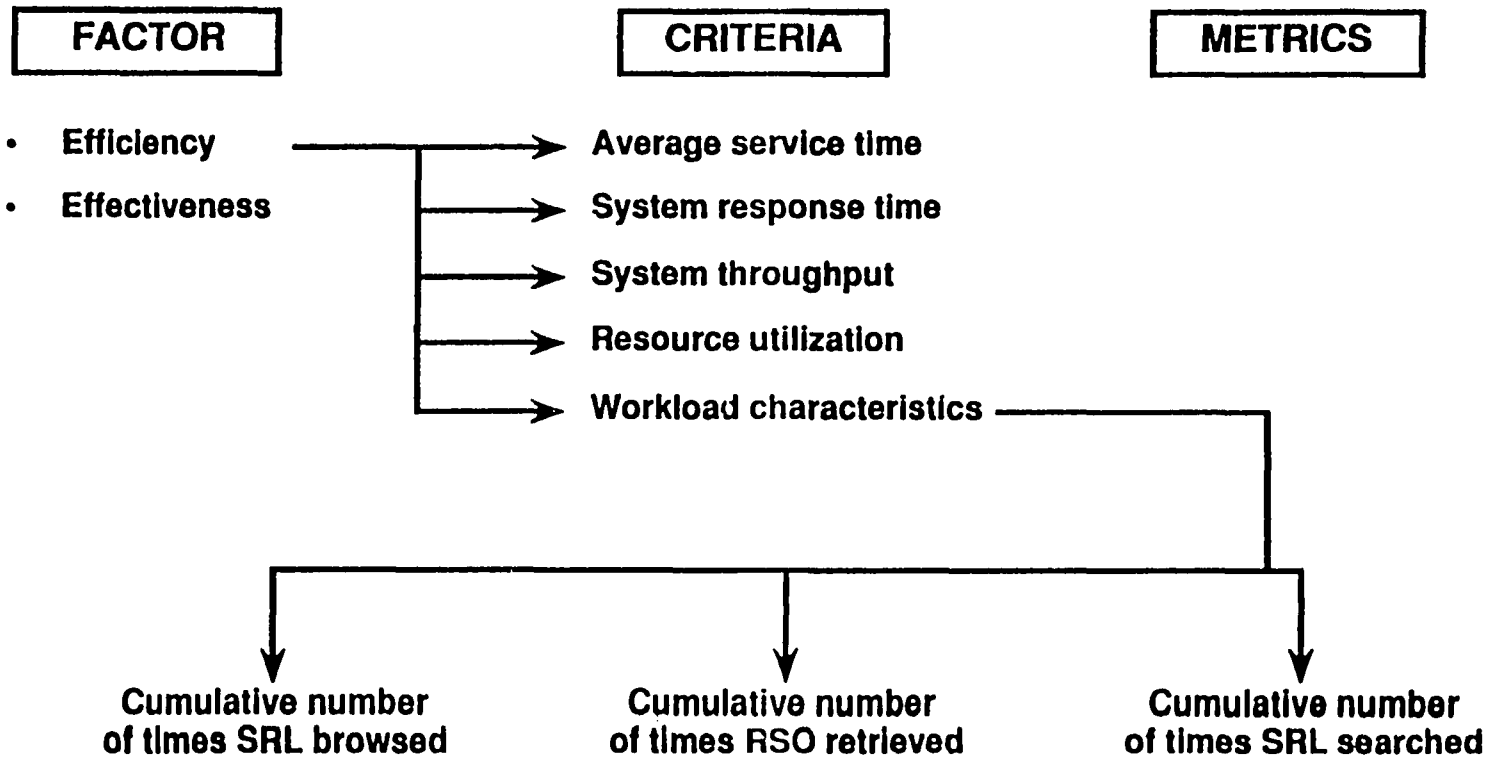


QUALITY METRICS

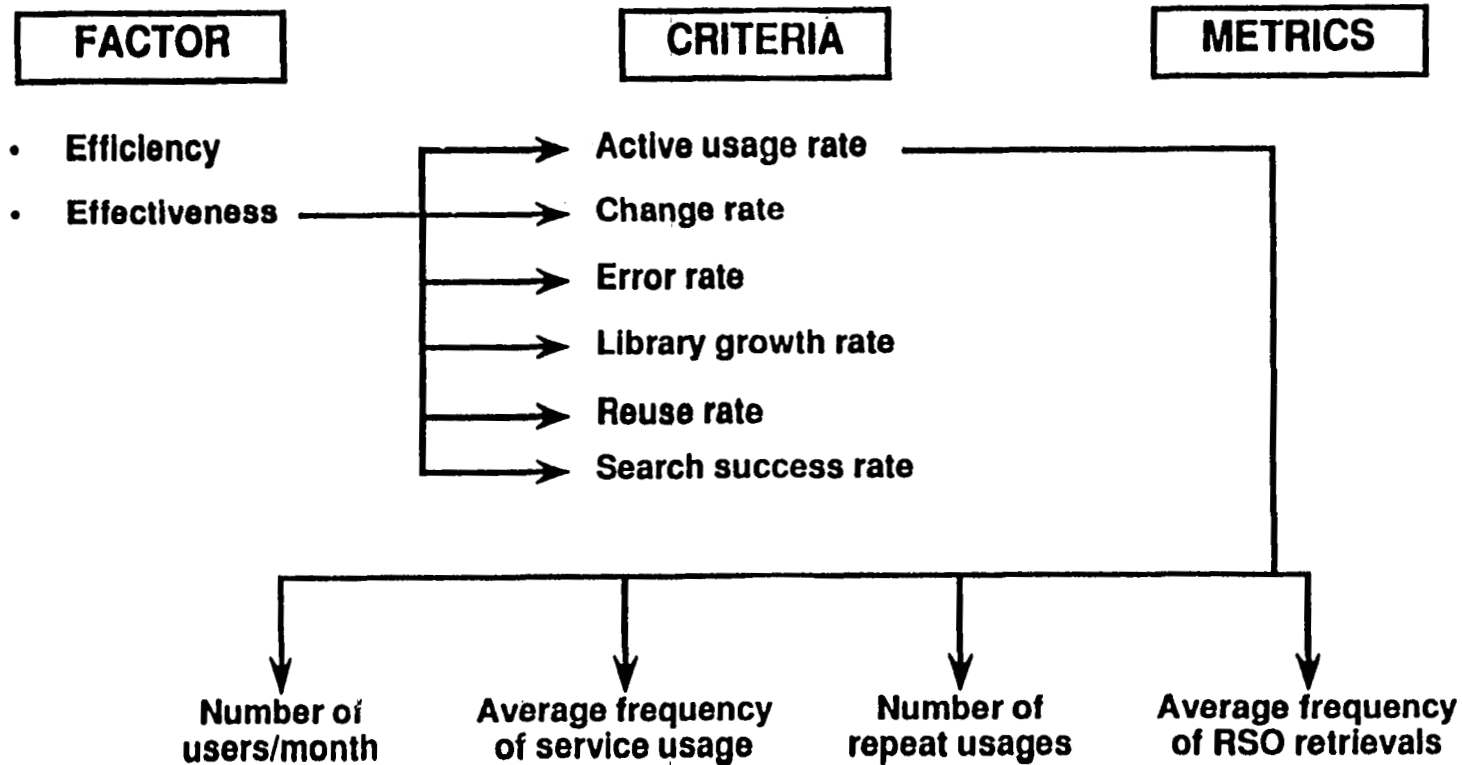


D. Paffter
RCI
Page 14 of 19

LIBRARY EFFICIENCY



LIBRARY EFFECTIVENESS



D. Rafter
RCI
Page 16 of 19

LONG-TERM COST/BENEFITS

NET PRESENT WORTH

$$NPW = \sum_{t=0}^T (B_t) (1/(1+i)^t)$$

NON-RECURRING COSTS

- Acquisition \$ _____
- Adaptation _____
- Documentation _____
- Infrastructure _____
- Training _____

COSTS \$ _____

RECURRING COSTS

- Administration \$ _____
- Maintenance _____
- Operations _____

COSTS \$ _____

TANGIBLE BENEFITS

- Cost avoidance \$ _____
- Added capability _____
- Reduced cost of quality _____
- Cost savings _____

BENEFITS \$ _____

INTANGIBLE BENEFITS

- Better customer satisfaction \$ _____
- Fitness for use _____

BENEFITS \$ _____

HYPertext LIBRARY EFFICIENCY AND EFFECTIVENESS

- No. of objects In library
- No. of links traversed/hit
- No. of items browsed/hit
- Amount of time for a hit
- No. of log in's per user
- Amount of time/user session
- No. of objects withdrawn/user session
- No. librarian actions/object
- No. of objects submitted/month
- No. of objects withdrawn/month
- No. of SPRs/object/month
- No. of SCRs/object/month

Usage Profiles

- By object
- By service
- System-wide

SUMMARY AND CONCLUSIONS

Page 16

RCI-TN-470

- We've described the JIAWG software reuse metrics and measurement framework
- We've described the pilot implementation of the framework on OSS and RAASP
- We've discussed our multiple instance reuse version of the COCOMO model
 - Needed to explore the economics of reuse
- We've just touched the surface of the issues involved
- Your thoughts, feedback and help are solicited especially if you have "hard" data to share

D. Butler
RCI
Page 19 of 19