

MEASUREMENT FRAMEWORKS

MEASUREMENT FRAMEWORKS

Measurement is not just the collection of data/metrics

calendar time

number of open problems

number of defects found in inspections

cyclomatic complexity

machine time

lines of code/module

total lines of code

severity of failures

total effort

total number of defects

lines of code/staff month

number of failures during system test

MSWE 609

2

MEASUREMENT FRAMEWORKS

Measurement must be driven by goals and models

for the processes and products

projects

organization

There are a variety of measurement frameworks that help you define the appropriate metrics and interpret the results of the measurement process

The most common frameworks in use are:

Goal/Question/Metric Paradigm (GQM)

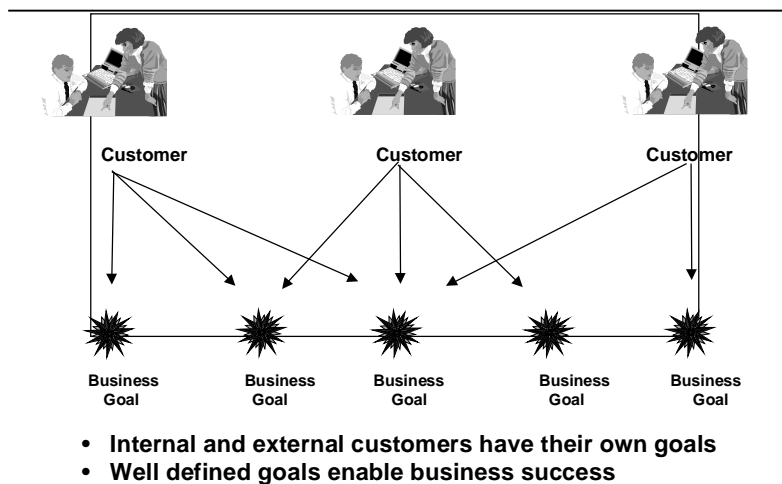
Quality Function Deployment (QFD)

Software Quality Metrics (SQM)

MSWE 609

3

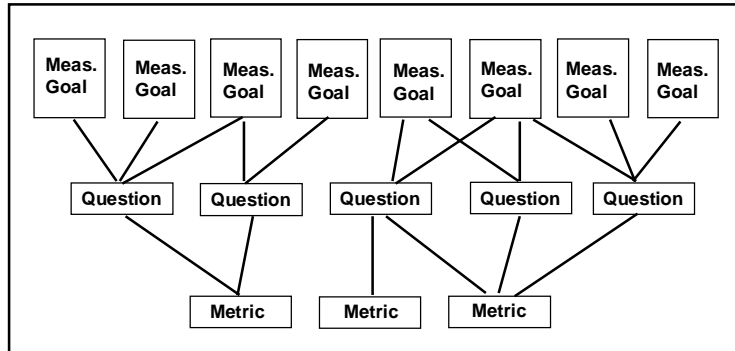
THE MEASUREMENT INFRASTRUCTURE



MSWE 609

4

THE MEASUREMENT INFRASTRUCTURE Goal Based Measurement



- Each metric supports multiple goals
- Questions focus metric selection and in-process analysis

MSWE 609

5

MEASUREMENT FRAMEWORKS

Goal/Question/Metric Paradigm

A mechanism for defining and interpreting operational, measurable goals

It uses four parameters:

a model of an **object of study**,

e.g., a process, product, or any other experience model

a model of one or more **focuses**,

e.g., models that view the object of study for particular characteristics

a **point of view**,

e.g., the perspective of the person needing the information

a **purpose**,

e.g., how the results will be used

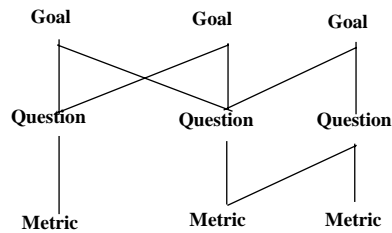
to generate a **GQM model**

relative to a **particular environment**

MSWE 609

6

GOAL/QUESTION/METRIC PARADIGM Goal and Model Based Measurement



A Goal links two models: a model of the object of interest *and* a model of the focus and develops an integrated GQM model

Goal: Analyze the final product to characterize it with respect to the various defect classes from the point of view of the organization

Question: What is the error distribution by phase of entry

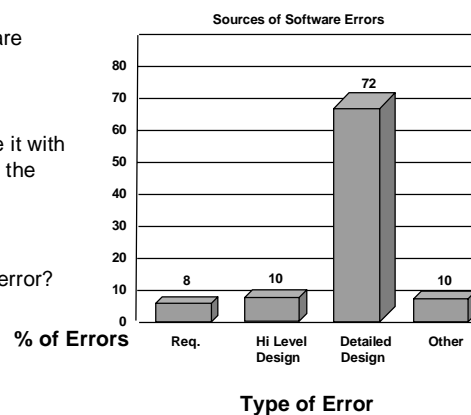
Metric: Number of Requirements Errors, Number of Design Errors, ...

MSWE 609

7

DEFINING MEASUREMENT GOALS A GOAL/QUESTION/METRIC EXAMPLE

- **Business Goal**
 - Understand problem areas in the software business
- **A Measurement Goal**
 - Analyze the final product to characterize it with respect to the various defect classes from the point of view of the organization
- **Question**
 - What is the error distribution by type of error?
- **Metrics**
 - Number of Requirements Errors, Number of Design Errors, ...



MSWE 609

8

GOAL/QUESTION/METRIC PARADIGM

Overview of the GQM Approach

Develop a set of corporate, division and project **goals** for productivity and quality, e.g., customer satisfaction, on time delivery, improved quality, developing reusable objects, reusing experiences

Generate questions (based upon models) that define those goals as completely as possible in a quantifiable way.

Specify the measures needed to be collected to answer those questions and track process and product conformance to the goals.

Develop mechanisms for data collection.

Collect, validate and analyze the data in real time to provide feedback to projects for corrective action.

Analyze the data in a postmortem fashion to assess conformance to the goals and make recommendations for future improvements.

MSWE 609

9

GOAL/QUESTION/METRIC PARADIGM

Goal Generation Template

Goals may be defined for any object, for a variety of reasons, with respect to various models of quality, from various points of view, relative to a particular environment.

Purpose: Analyze some
(objects: processes, products, other experience models)
for the purpose of
(why: characterization, evaluation, prediction, motivation, improvement)

Perspective: with respect to
(focus: cost, correctness, defect removal, changes, reliability, user friendliness,...)
from the point of view of
(who: user, customer, manager, developer, corporation,...)

Environment: in the following context
(problem factors, people factors, resource factors, process factors,...)

MSWE 609

10

GOAL/QUESTION/METRIC PARADIGM

Template parameters

Environment: defines all context variables for the study

Objects of study: processes, products, other experience models

Associated Models:

- the process itself,
- how the performance of the process will be evaluated
- the domain
- how well the domain is understood
- how well the object on which the process is being performed is understood

Indirect Models:

- training, experience, ability of the people in the process, the domain

Focus Models: cost, correctness, defect removal, changes, reliability, user friendliness,...

MSWE 609

11

GOAL/QUESTION/METRIC PARADIGM

Template parameters

Purpose of study:

understanding or characterization requires:

- model of the object and any data we collect helps provide us with some insight

evaluation requires:

- some basis for comparison (historical data base) or some absolute measure

prediction requires:

- a pattern in the data that allows us to develop a consistent model based upon factors that we have available or can reasonably estimate

motivation or improvement requires:

- an accurate model that represents a true understanding of the object or positive quality we are modeling

Point of view:

- determines such things as
 - when the information should be available
 - who needs it
 - its level of granularity
 - its acceptable accuracy, etc.

MSWE 609

12

GOAL/QUESTION/METRIC PARADIGM

Characterizing Goals

Analyze the **software products**
 in order to **characterize** them with respect to

- development error rates**
- cost in staff months**
- % of code reused**

from the point of view of the **organization** relative to the **SEL environment**

Analyze the **software processes**
 in order to **characterize** them with respect to

- effort distributions**
- classes of errors**
- source code growth**

from the point of view of the **organization** relative to the **SEL environment**

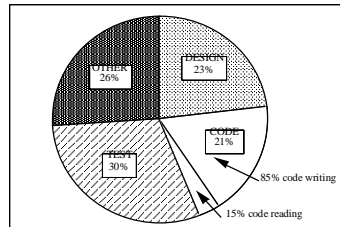
MSWE 609

13

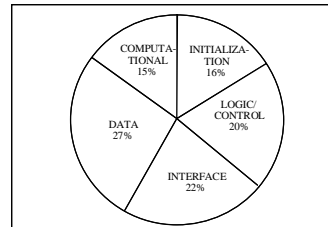
GOAL/QUESTION/METRIC PARADIGM

NASA/SEL PROCESS BASELINE EXAMPLE

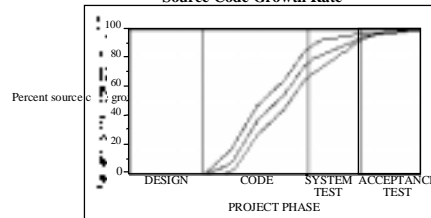
Effort Distribution*



Classes of Errors*



Source Code Growth Rate*



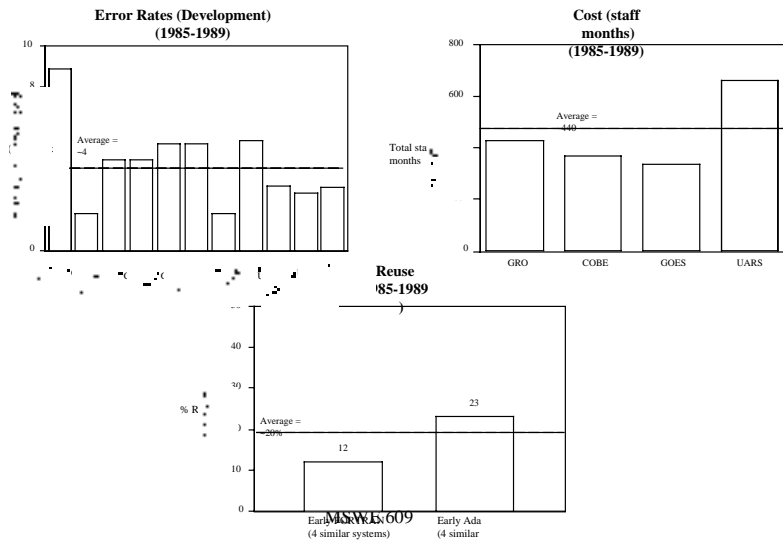
*Data from 11 Flight Dynamics projects (mid 1980s)

MSWE 609

93M1147007

GOAL/QUESTION/METRIC PARADIGM

NASA/SEL Product Baseline Example



15

DEFINING GOAL RELATIONSHIPS Example

- **Business Goal:** Cycle Time Reduction
- **Measurement Goal:** Improve Software Delivery Schedule by 10%
- **Example Support Measurement Goals:**

Goal 1:
Build a
Schedule
Baseline

Analyze **representative projects** in order to **characterize** them with respect to **calendar time & effort for each phase, activity and in total** from the point of view of **the Corporation**

Goal 2:
Determine how good your prediction capabilities are

Analyze **representative projects** in order to **characterize** them with respect to **variation from predicted schedule for each phase, activity and in total** from the point of view of **the Corporation**

MSWE 609

16

GOAL/QUESTION/METRIC PARADIGM

Process Goal: Question Guidelines

Process Conformance:

Characterize the process quantitatively and assess how well the process is performed.

Domain Understanding:

Characterize the object of the process and evaluate the knowledge of the object and its domain by the process performers.

Focus:

Analyze the output of the process according to some quality model and some viewpoint.

Feedback:

What has been learned about the process, its application, the product domain, or any other process or product?

MSWE 609

17

GOAL/QUESTION/METRIC PARADIGM

Process Goal: Example

Analyze the system test process for the purpose of evaluation with respect to defect slippage from the point of view of the corporation.

System Test Process Model:

Goal: Generate a set of tests consistent with the complexity and importance of each requirement.

Procedure: (1) Enumerate the requirements, (2) Rate importance by marketing, (3) Rate complexity by system tester, (4) ...

Defect Slippage Model:

Let **Fc** = the ratio of faults found in system test to the faults found after system test on this project.

Let **Fs** = the ratio of faults found in system test to the faults found after system test in the set of projects used as a basis for comparison.

Let **QF = Fc/Fs** = the relationship of system test on this project to faults as compared to the average in the appropriate basis set.

18

GOAL/QUESTION/METRIC PARADIGM

Simple Interpretation of Defect Slippage Model

if QF > 1 then
 method better than history
 check process conformance
 if process conformance poor
 improve process or process conformance
 check domain conformance
 if domain conformance poor
 improve object or domain training

if QF = 1 then
 method equivalent to history
 if cost lower than normal then method cost effective
 check process conformance

if QF < 1 then
 check process conformance
 if process conformance good
 check domain conformance
 if domain conformance good
 method poor for this class of project

MSWE 609

19

GOAL/QUESTION/METRIC PARADIGM

Product Goal: Question Guidelines

Product Model/Definition:

Characterize the product qualitatively independent of the perspective of interest. Aspects of interest include:

Logical/Physical Attributes:

Characterize the logical and physical attributes of the product e.g.,

logical attributes: application domain, function

physical attributes: size, complexity, interfaces

dynamic attributes: coverage, reliability

Cost:

Characterize the resources expended, e.g., effort, computer time

Changes:

Characterize the modifications associated with the product, e.g., enhancements, errors, faults, failure

Context:

Characterize the customer community and their operational profiles

MSWE 609

20

GOAL/QUESTION/METRIC PARADIGM

Product Goal: Question Guidelines

Perspective/Focus:

Analyze the product models from each perspective of interest, e.g., reliability, user friendliness, specifying the following:

Major model(s) used

Specify some perspective model/definition and viewpoint

Validity of the model for the project

Evaluate the appropriateness of the model for the project environment

Validity of the data collected

Evaluate the quality of the data

{Substantiation of the model

Given any alternate perspectives that provide support for the quality of the results}

Feedback:

What has been learned about the product, the processes that produced it, or any other product that will improve this project and future projects? 21

GOAL/QUESTION/METRIC PARADIGM

Product Goal Example

Analyze the design document for the purpose of evaluation with respect to the design inspection defects uncovered from the point of view of the project manager.

Design Inspection Process Model:

Goal: Analyze the design document for the purpose of characterization with respect to its correct and complete implementation of the requirements from the point of views of the user, developer and tester.

- Procedure:
- (1) Disseminate the appropriate part of the requirements and design documents,
 - (2) Read the document by the appropriate set of readers from the appropriate points of view,
 - (3) Report defects by various classification schemes, including omission and commission defects,
 - (4) ... MSWE 609

22

GOAL/QUESTION/METRIC PARADIGM

Product Goal Example

Design Document Product Model/Definition:

Logical/Physical Attributes:

logical attributes: application domain, function

physical attributes: size: lines of design language, complexity, interfaces

Cost:

effort by phase, activity, person

computer time

Changes:

of enhancements, errors,

of errors and faults found during design inspection

Context:

Customer community: designers, coders, system testers, maintainers

MSWE 609

23

GOAL/QUESTION/METRIC PARADIGM

Simple Document/Defect Evaluation Model

KLOD = number of thousand lines of design language

Fc = number of faults/**KLOD** found in design inspections on this project

Fs = number of faults/**KLOD** found in design inspections in the set of projects used as a basis for comparison

QF = Fc/Fs = the relationship of faults found on this project as compared to the average the appropriate basis set

if **QF** > 1 then **QF** = H (worse than history)

if **QF** <= 1 then **QF** = L (better than history)

PC = the process conformance rating on this project

= C if inspections are performed to the definition, N otherwise

DU = the domain understanding rating on this project

= S if domain understanding is satisfactory, U otherwise

<u>PC</u>	<u>DU</u>	<u>QF</u>	<u>Design-in</u>	<u>Design-out</u>	<u>Design Process</u>	<u>Inspection Process</u>
C	S	L	good	good	effective	effective
C	S	H	poor	fixed-up	not-effective	effective
N	X	X	?	?	?	?
X	U	X	?	MSWE 609	?	?

PROCESS GOAL EXAMPLE

Test Method Process Conformance

Characterize the test method experience of the test team.

(subjective rating per person)

- 0 - none
- 1 - have read the manuals
- 2 - have had a training course
- 3 - have had experience in a laboratory environment
- 4 - have used on a project before
- 5 - have used on several projects before
- x - no response

How many requirements are there?

(enumerate them)

What is the importance of testing each requirement?

(Subjective rating 0 - 5 by marketing and testers)

- 0 - not important, could be left out
- 1 - not too important, may affect some users
- 2 - mildly important, will affect some users
- 3 - important, should affect most users
- 4 - extremely important, part of the essence of the system
- 5 - critical, without this the system is useless
- x - don't know

MSWE 609

25

PROCESS GOAL EXAMPLE

Test Method Process Conformance

What is the complexity of testing each requirement?

(subjective rating 0 - 5 by tester)

- 0 - doesn't need to be tested
- 1 - easy to test, one test should do it
- 2 - reasonably easy to test, only a few ad hoc tests are needed
- 3 - not easy to test, requires carefully made up test suite
- 4 - very difficult to test, requires a lot of thought to make up a good test suite
- 5 - extremely difficult to test, requires a large, complex test suite
- x - impossible to test

What is the distribution of tests over requirements?

(number of tests/requirement)

Is the number of tests/requirement consistent with the evaluation of its complexity and importance?

- 0 - there are no tests for this requirement
- 1 - there is at least one test
- 2 - there are several tests but not nearly enough
- 3 - the number of tests are reasonable but insufficient for the requirement importance/complexity
- 4 - the number of tests are sufficient for the complexity and importance of the requirement
- 5 - the number of tests are more than adequate for the requirement importance and complexity
- x - no response

or a formula,

e.g., If importance = 5 and complexity = 5

then number of test cases = ...MSWE 609

26

PROCESS GOAL EXAMPLE
Test Object Domain Understanding

Understanding of the Domain and Requirements

How familiar is the domain?

(subject rating 0 - 5 for each tester)

- 0 - domain new to me
- 1 - have had a course in the subject domain
- 2 - have built or tested one system in this domain
- 3 - have built and tested at least one system in this product line
- 4 - have built and tested several systems in this domain
- 5 - have tested and built several systems in this product line

How understandable are the requirements for this project?

(subjective rating 0 - 5 for each requirement)

- 0 - not understandable at all
- 1 - requirement ambiguous or not sure what it means
- 2 - not sure of the full ramifications
- 3 - reasonably clear requirement
- 4 - requirement is perfectly clear
- 5 - have successfully tested this type of requirement before

27

PROCESS GOAL EXAMPLE
Test Object Domain Understanding

Quality of the Tests

How precisely are the tests (inputs, results) known in advance?

(subjective rating 0 - 5)

- 0 - there were no tests for this requirement
- 1 - will make the inputs up at terminal
- 2 - know the inputs but not the results
- 3 - know the inputs and the range of the results
- 4 - know the inputs and the results
- 5 - have simulation results for the test cases

How confident are you that the result is correct?

(subjective rating 0 - 5)

- 0 - there are no results
- 1 - the results are incorrect
- 2 - not sure the results are correct
- 3 - think they are correct
- 4 - reasonably sure they are correct
- 5 - positive they are correct

What is the evaluation of the domain conformance?

28

PROCESS GOAL EXAMPLE

Goal Focus: Cost of Use

What is the total cost of testing?

OR

(depending upon the level of granularity required)

What is the staff time to make a test?

What is the staff time to run a test and check the result?

What is the staff time to isolate the fault?

What is the staff time to design and implement a fix?

What is the staff time to retest?

What is the machine time used?

MSWE 609

29

PROCESS GOAL EXAMPLE

Goal Focus: Simple Defect Slippage Model

Simple defect slippage model questions:

What is the number of faults failures discovered during system test, acceptance test and one month, six months, one year after system release on this project?

What is the number of faults failures discovered during system test, acceptance test and one month, six months, one year after system release on the set of projects classified as similar?

What is the ratio of faults in system test on this project to faults found from system test on?

What is the ratio of faults in system test on the set of similar projects to faults found from system test on?

What is the ratio of system test performance on this project to system test performance on the set of similar projects?

MSWE 609

30

PROCESS GOAL EXAMPLE
Goal Focus: General Defect Slippage Model

What is the number of errors, faults and failures on this project
in total,
per line of code,
by various classification schemes, and
by cost to isolate, fix and overall,
discovered during each phase of development and one month, six months, one year after
system release?

What is the number of errors, faults and failures on the set of similar projects
in total,
per line of code,
by various classification schemes, and
by cost to isolate, fix and overall,
discovered during each phase of development and one month, six months, one year after
system release?

What is the ratio of weighted faults in system test on this project to faults found from system
test by various classification schemes?

What is the ratio of weighted faults in system test on the set of similar projects to faults found
from system test by various classification schemes?

What is the ratio of system test performance based upon the various slippage models on this
project to system test performance on the set of similar projects?

PROCESS GOAL EXAMPLE
Feedback

Does the system test method need to be refined or modified?

Is more or different training needed in the method or the technology?

Is more or different training needed in the application domain?

What should be automated?

What is the input to the requirements, specification, design and code
techniques, methods, and tools, and the defect detection techniques,
methods, and tools?

PROCESS GOAL EXAMPLE

Data Sources

System test tables:

System test table 1: **Nature of requirements**

(Filled out after baselining of requirements)

Req. #	How understandable is the requirement	Importance of testing (marketing)	(systest)
--------	---------------------------------------	-----------------------------------	-----------

R1	5	3	2
----	---	---	---

System test table 2: **Nature of tests**

(Filled out after test plan)

Req. #	# of tests	How well are tests known?	Difficulty of testing	Evaluation subj. stat	Are # of tests consistent with diff. & import?	Rating
--------	------------	---------------------------	-----------------------	-----------------------	--	--------

R1	5	3	2	3	3/50	4	4
----	---	---	---	---	------	---	---

System test table 3: **Results of the tests**

(Filled out after tests run)

Test #	Failure? Yes : No	How confident are you in the results?	# of Faults found
--------	-------------------	---------------------------------------	-------------------

T1	x	3 MSWE 609	1
----	---	------------	---

33

PROCESS GOAL EXAMPLE

Defect/Change Report Form

For each fault, record:

- Description of fault
- Date isolated
- Date fixed
- Time to fix
- # of components changed
- # of components that had to be examined
- # of components that were examined

Classifications:

- Error origin
- Error domain
- Detection time
- Omission/commission
- Software aspect
- Failure severity

MSWE 609

34

PROCESS GOAL EXAMPLE

Data Presentations

Slippage model data:

QEs

REs, RPEs

Es, Ea, Eo

Histograms of:

Number of faults found in each phase

The number of requirements vs. subjective ratings of

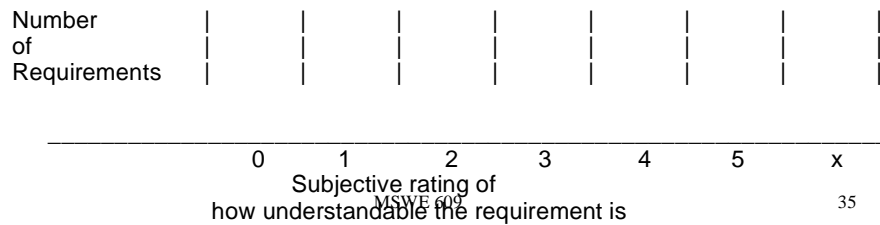
how understandable the requirement is

importance of requirement

difficulty of testing the requirement

...

Example:



GOAL/QUESTION/METRIC PARADIGM

Product Goal: Example

Analyze the **final product** for the purpose of **evaluation** with respect to **reliability** from the point of view of the **various customer classes**.

We need a model of the

final product

reliability

PRODUCT GOAL EXAMPLE

Final Product Model: Logical/Physical Attributes

Classify the application domain and solution domain of the final product?

What is the function of the final product?

What is the size of the final product with respect to:
pages or lines of user documentation,
source lines with comments,
executable statements,
program units, e.g., functions, processes, components,
the number of requirements, etc.?

What is the complexity of the product with respect to:
syntactic control (e.g., cyclomatic complexity, software science
metrics),
data use (e.g., span, data bindings), etc.
for each appropriate unit?

What is the distribution of programming language features used?

What are the run-time time and space constraints?

MSWE 609

37

PRODUCT GOAL EXAMPLE

Product Model: Cost

What is the effort by phase, activity, personnel type, used to develop the system?

How much machine time was used to develop the system; documentation as well as other aspects?

How much calendar time was used to develop the system, each component?

Presentation:

Histogram of the number of hours and the percent of time vs. the time card accounting sub-codes.

Histogram of the # of hours and the percent of time vs. the various activities (from the beginning of data collection and during the last report period).

Number of Hours							% of time
	A1	A2	A3	A4		

where A1, A2, A3, etc., are the different activities for which time is being accounted.

MSWE 609

38

Product Model: Changes

A. Enhancements

What is the number of enhancements
(normalized by calendar time, phase of project)?

What are the number of enhancements categorized by
type (requirements, specification, design, architecture, planned
enhancements, insert/delete debug code, improve clarity,
optimize: space or time, feature, enhancement, bug)?
market/external and internal needs?
size, e.g., number of lines of code, number of components affected, etc.?
disposition, e.g., rejected as a change, not relevant, under
consideration, being worked on, completed, saved for next
enhancement?
level of document changed?

How many customers are affected by the changes?

What is the Trouble Report history profile for each change?

MSWE 609

39

PRODUCT GOAL EXAMPLE

Product Model: Changes

B. Defects

What are the number of errors in total and by type, e.g., error origin, error
domain?

What are the faults in total and by type, e.g., fault entry time, fault
detection time/phase, (omission, commission), software aspect,
mechanism of discovery, product level?

What are the failures in total and by type, e.g. severity, failure detection
time/phase?

What are the number of errors, faults and failures in total and by type,
normalized by phase and calendar time?

Presentations:

Histograms of the number of changes/defects by various classes.

Graphs of # of changes/defects initiated and closed vs. calendar time

Number of errors/component for each component

MSWE 609

40

PRODUCT GOAL EXAMPLE
Product Model: Context

A. Customer community

What classes of customers are expected to use the system?

What is the matrix of functional requirements vs. customer classes?

What is the matrix of functional requirements vs. components of the system?

B. Operational profile

What percent of the system is expected to be executed by each customer class?

MSWE 609

41

PRODUCT GOAL EXAMPLE
Product Model: Data - Various Matrices

Requirements X Customer Usage Matrix, e.g.,

CUSTOMER

REQUIREMENTS		percent use	
		by customer	

Components X Requirements Coverage Matrices, e.g.,

REQUIREMENTS

COMPONENTS		percent R_i covered by X_j	= 100
		percent X_i covers R_j	>= 100

Components X Customer Coverage Matrix

Test X Components Coverage Matrix

Test X Requirements Coverage Matrix

MSWE 609

42

PRODUCT GOAL EXAMPLE

Focus Model: Reliability

Perspective: Examine the reliability of the system from the user's point of view.

Major model used: some MTTF model (e.g., Musa)

What is the MTTF for the overall system, during system test, acceptance test, and for each customer base during operation?

What is the estimated fault density for the model? target MTTF?

Presentations:

Graphs of the MTTF vs. execution time and calendar time containing actual mean-time to failure data and projected mean-time to failure data

MSWE 609

43

PRODUCT GOAL EXAMPLE

Focus Model: Reliability

Validity of the model for the project:

How many requirements are there? What is the distribution of tests/requirement?

What is the customer/requirements matrix with probabilities (usage weightings) for each requirement?

How accurately does that describe each user operational profile?

Do the test cases reflect the customer/requirements probability matrix, i.e., is the test suite made up according to the different operational scenarios of the customer bases?

Is the probability distribution of tests run during system/acceptance test based on customer OP profile?

MSWE 609

44

PRODUCT GOAL EXAMPLE

Focus Model: Reliability

Validity of the model for the project:

Were the test cases randomized and distribution based on the customer operational profile from the previous question when they were run?

Are corrections being made as failures are discovered?

Are new errors being introduced into the system during testing and debugging?

Is the model being run for each customer base with a different operational scenario?

Do the cumulative changes imply that the reliability model should be restarted?

MSWE 609

45

PRODUCT GOAL EXAMPLE

Focus Model: Reliability

Validity of data collected:

How valid is the failure data and the associated failure times?

Substantiation of the model:

Use of error profiles:

What is number of faults detected per 1000 lines and per component (one-half system test, system test, one-half certification, end of certification)?

Does this agree with previous projects' fault history and reliability ratings?

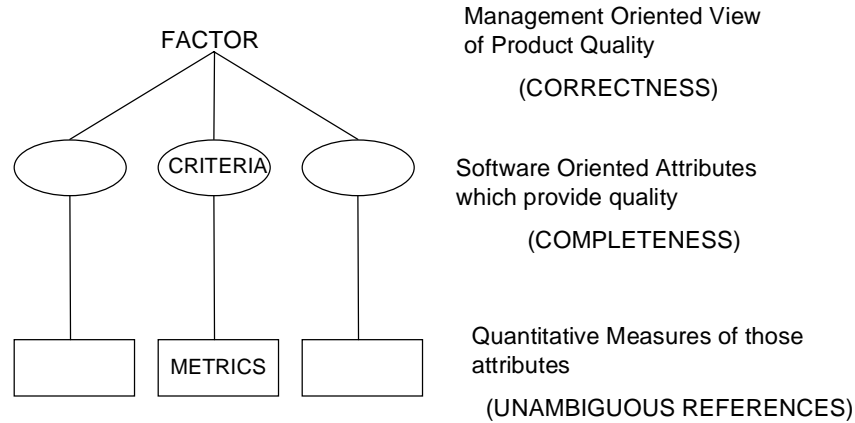
Use of coverage data:

What is the requirement, component coverage of the test plan at various points in time?

Do the requirement and component coverage correspond to the distribution in the (requirement, component) and (component, customer) matrices?

46

User's View of Quality Factor/Criteria/Metric Paradigm



- product characteristics
- customer point of view

Boehm, Thayer, Lipow (TRW)
McCall, Walters (GE)
Boehm & Harris

MSWE 609

Software Quality Metrics

Perspective: Examine the user satisfaction with the final system?

Major Model(s) used: User Response to System

- How many failures are reported by the users?
- How many clarifications are requested by the user?
- Is the use of the system growing, shrinking, staying the same?
- How many requests are there for functional enhancements?
- How many functional change requests are real versus functions already in the system?
- How many performance change requests are being submitted?

Validity of the Model for the Project: User Representation

- Has a user committee been appointed that covers all user types?
- Has the user community had any input into the requirements?
- Are there prototype screens for the users to play with?
- Is the user part of the test plan development?
- Is the user part of the test team?

Validity of the Data Collected:

- How valid is the data collected?

48

Software Quality Metrics

Substantiation of the Model: User Subjective Evaluation

How responsive is the system to user request for functionality?

How responsive is the system to user request for performance?

How does the user rate the system with respect to:

ease of use

functionality

performance

ease of understanding the documentation?

(a questionnaire should be made up)

Why did you buy it?

Did you get more, less, same as expected?

How many others have you used?

Would you recommend it to a friend? A competitor?

Would you reorder?

Which competitors did you consider?

How many requests are there for functional enhancements?

MSWE 609

49

Software Quality Metrics

Performance

How well does it Function?

User Concern	Quality Factor
How well does it utilize a resource?	Efficiency
How secure is it?	Integrity
What confidence can be placed in what it does?	Reliability
How well will it perform under adverse conditions?	Survivability
How easy is it to use?	Usability

MSWE 609

50

Software Quality Metrics

Design

How valid is the Design?

User Concern	Quality Factor
How well does it conform to the requirements?	Correctness
How easy is it to repair?	Maintainability
How easy is it to verify its performance?	Verifiability

MSWE 609

51

Software Quality Metrics

Adaptation

How adaptable is it?

User Concern	Quality Factor
How easy is it to expand or upgrade its capability or performance?	Expandability
How easy is it to change?	Flexibility
How easy is it to interface with another system?	Interoperability
How easy is it to transport?	Portability
How easy is it to convert for use in another application?	Reusability

MSWE 609

52

Software Quality Metrics

Evaluating Factors

Based upon project needs, various quality factors are chosen for monitoring, analysis, and feedback for corrective action.

Each Factor is defined by a rating formula whose data is not available till project completion

Each Factor is also defined by a set of criteria that are collected during project development and are used as predictors of the quality factors

These metrics are collected on worksheets

MSWE 609

53

Software Quality Metrics

Quality Factor Formulas

Efficiency	1 - $\frac{\text{Actual Utilization}}{\text{Allocated Utilization}}$
Integrity	1 - $\frac{\text{Errors}}{\text{Lines of Code}}$
Reliability	1 - $\frac{\text{Errors}}{\text{Lines of Code}}$
Survivability	1 - $\frac{\text{Errors}}{\text{Lines of Code}}$
Usability	1 - $\frac{\text{Labor-Days to Use}}{\text{Labor-Years to develop}}$
Correctness	1 - $\frac{\text{Errors}}{\text{Lines of Code}}$
Maintainability	1 - 0.1(Average Labor-Days to Fix)

MSWE 609

54

Software Quality Metrics

Quality Factor Formulas

Verifiability	1 -	$\frac{\text{Effort to Verify}}{\text{Effort to Develop}}$
Expandability	1 -	$\frac{\text{Effort to Expand}}{\text{Effort to Develop}}$
Flexibility	1 -	0.05 (Average Labor-Days to Change)
Interoperability	1 -	$\frac{\text{Effort to Couple}}{\text{Effort to Develop}}$
Portability	1 -	$\frac{\text{Effort to Transport}}{\text{Effort to Develop}}$
Reusability	1 -	$\frac{\text{Effort to Convert}}{\text{Effort to Develop}}$

55

Software Quality Metrics

Quality Criteria

Each factor has associated with it a set of quality criteria:

Factor	Criteria
Efficiency	Effectiveness-Communication Effectiveness-Processing Effectiveness-Storage
Integrity	System Accessibility
Reliability	Accuracy Anomaly Management Simplicity
Survivability	Anomaly Management Autonomy Distributedness Modularity Reconfigurability

56

Software Quality Metrics

Quality Criteria

Factor	Criteria
Usability	Operability Training
Correctness	Completeness Consistency Traceability
Maintainability	Consistency Modularity Self-Descriptiveness Simplicity Visibility
Verifiability	Modularity Self-Descriptiveness Simplicity Visibility

57

Software Quality Metrics

Quality Criteria

Factor	Criteria
Expandability	Augmentability Generality Modularity Self-Descriptiveness Simplicity Virtuality
Flexibility	Generality Modularity Self-Descriptiveness Simplicity
Interoperability	Commonality Functional Overlap Independence Modularity System Compatibility

58

Software Quality Metrics

Quality Criteria

Factor	Criteria
Portability	Independence Modularity Self-Descriptiveness
Reusability	Application Independence Document Accessibility Functional Scope Generality Independence Modularity Self-Descriptiveness Simplicity System Clarity

MSWE 609

59

Software Quality Metrics

Metrics

Each criteria has associated with it a set of quality metrics:

Example:

Criteria	Metrics
Completeness	CP 1: Completeness Checklist
Consistency	CS 1: Procedure Consistency CS 2: Data Consistency
Traceability	TR 1: Cross Reference

MSWE 609

60

Software Quality Metrics

Worksheets

Associated with each phase of development are worksheets that collect data that can be turned into quality metrics

For Example:

Metric Worksheet 0: Collected during Systems Requirements Analysis

Metric Worksheet 1: Collected during Software Requirements Analysis

Metric Worksheet 2: Collected during Preliminary Design

Metric Worksheet 3: Collected during Detailed Design

Metric Worksheet 4: Collected during Coding and Unit Test

MSWE 609

61

Software Quality Metrics

Evaluation

Data collected at the unit level is accumulated at the subsystem level

Data collected at the subsystem level is accumulated at the system level

Accumulation can be done by averaging the values of metrics at the lower level units

Score sheets exist for each factor

Metric values at the system level are reported for each criteria

All metric values are designed to lie between 0 and 1.

MSWE 609

62

Software Quality Metrics

Metric Definition

Example: Completeness
From Worksheet 0:

- CP.1(1) Has a common technical vocabulary with equivalent definition been established for use with this system and for use with the interoperating systems (e.g., definition and use of data item, block, record)? Y, N, N/A
- CP.1(2) a. How many data references are identified? ____, N/A
b. How many identified data references are documented with regard to source, meaning, and format? ____, N/A
c. Calculate b/a and enter score. ____, N/A
- CP.1(3) a. How many data items are defined (i.e., documented with regard to source, meaning and format)? ____, N/A
b. How many data items are referenced? ____, N/A
c. Calculate b/a and enter score. ____, N/A
- CP.1(5) Have all defined functions been referenced? Y, N, N/A

MSWE 609

63

Software Quality Metrics

Metric Definition

- CP.1(6) Have all system functions been allocated to configuration items? Y, N, N/A
- CP.1(7) Have all referenced functions been defined (i.e., documented with precise inputs, processing, and output requirements)? Y, N, N/A
- CP.1(8) Are the processing flows (algorithms) and all decision points (conditions and alternate paths) in the flows described for all functions? Y, N, N/A
- CP.1 (11) a. How many software problem reports have been recorded to date? ____, N/A
b. How many recorded software problem reports have been closed to date? ____, N/A
c. Calculate b/a and enter score ____, N/A

MSWE 609

64

Software Quality Metrics

Metric Definition

Example: Completeness - Other Worksheets

Worksheet 1: Same as Worksheet 0 except for CP.1(1)

CP.1(1) All all inputs, processing and outputs clearly defined?
Y, N, N/A

Worksheet 2: Same as Worksheet 1 except CP.1(5,7,8) are eliminated
and CP.1(9) is added

CP.1(9) Are all conditions and alternative processing options defined for
each decision point? Y, N, N/A

Worksheet 3B: Same as Worksheet 2 except CP.1(3,6,7,8) are eliminated
and CP.1(10) is added

CP.1(10) Are all parameters in the argument list used? Y, N, N/A

MSWE 609

65

Software Quality Metrics

Metric Definition

Example: Completeness - Other Worksheets

Worksheet 3A tends to summarize the results of Worksheet 3B,
i.e., CP.1(1,2,3,4,9,10,11) are summed as follows:

CP.1(1) a.How many applicable units (answer of Y or N on 3B)? _____, N/A
b.How many units with answer of Y on 3B? _____, N/A
c. Calculate b/a and enter score. _____, N/A

Worksheet 4B has CP.1(2,4,9,10)

Worksheet 4A has CP.1(2,3,4,9,10,11)

MSWE 609

66

Software Quality Metrics

Relationship between SQM and GQM

We can express the SQM in the GQM framework by writing them as goals for the following form:

Analyze the **final product** for the purpose of **assessment** with respect to **efficiency** from the point of view of the **customer**.

Analyze the **final product** for the purpose of **assessment** with respect to **integrity** from the point of view of the **customer**.

That is, they all attempt to assess the final product from the point of view of the customer relative to the various quality factors

Analyze the **final product** for the purpose of **assessment** with respect to **efficiency, integrity, reliability, survivability, usability, correctness, maintainability, verifiability, expandability, flexibility, interoperability, portability, reusability** from the point of view of the **customer**.

MSWE 609

67

Quality Function Deployment (QFD)

Structure:

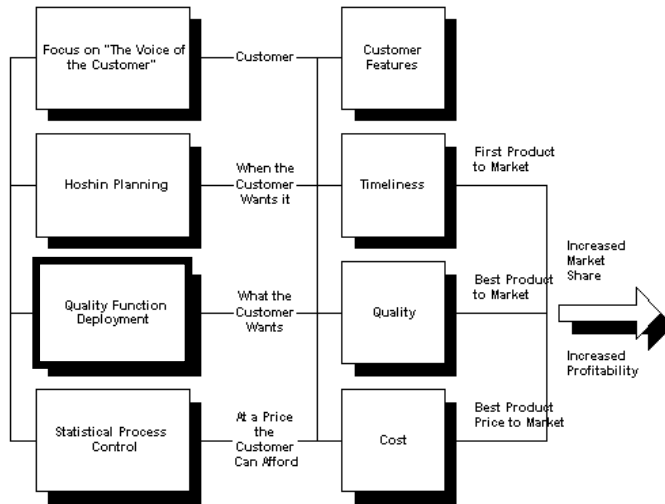
- What are the qualities the customer desires?
- What functions must the product serve and what functions must we use to provide the product or service?
- Based upon the resources we have available, how can we best provide what our customer wants?

Developed in Japan in 1970s.

MSWE 609

68

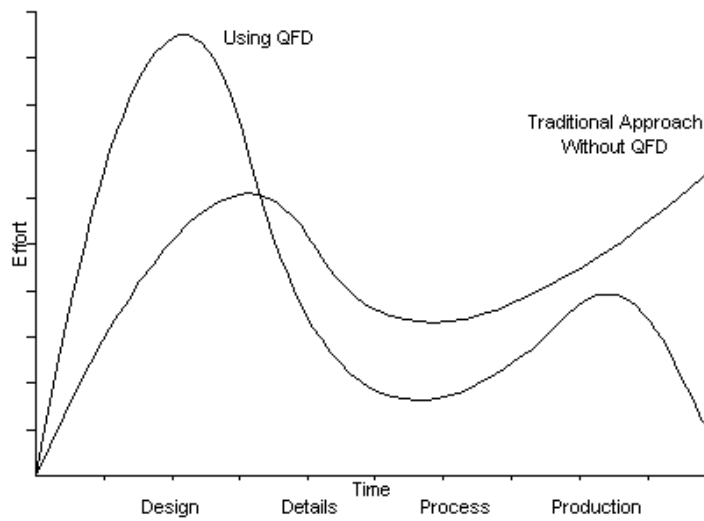
Total Quality Management (TQM) (in Japan)



MSWE 609

69

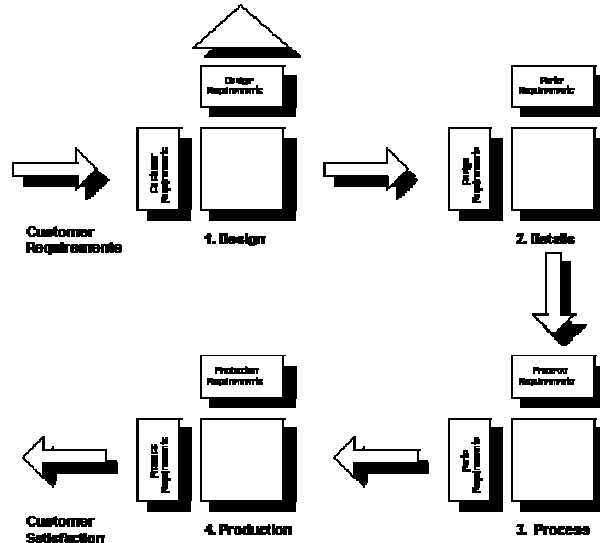
Workflow upfront with QFD



MSWE 609

70

Four Phases of QFD



MSWE 609

71

Four Phases of QFD

Design Phase - customer helps to define the product or service requirements.

Details Phase - The *How*s carried over from the Design phase become the *Whats* for the *Details Phase* and design specifications are converted into individual part details.

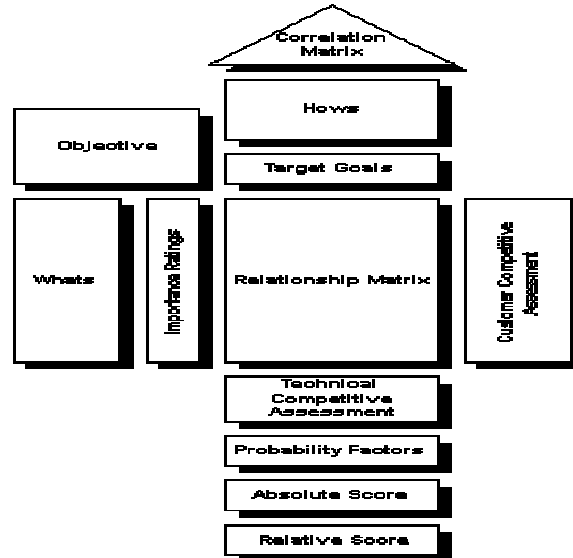
Process Phase - The processes required to produce the product are developed. The *How*s from the Details phase become the *Whats* for the Process Phase.

Production Phase - The *How*s from the Process phase become the *Whats* for the *Production Phase* and the production requirements for the product are developed.

MSWE 609

72

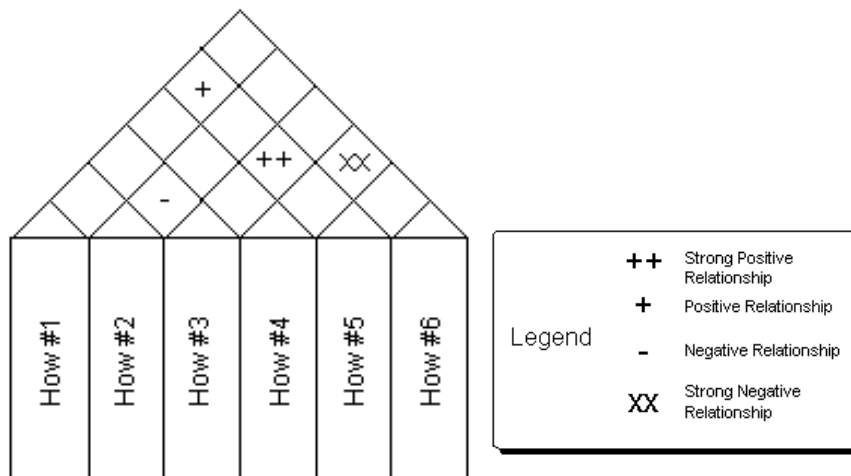
Quality measurements



MSWE 609

73

Correlation Matrix

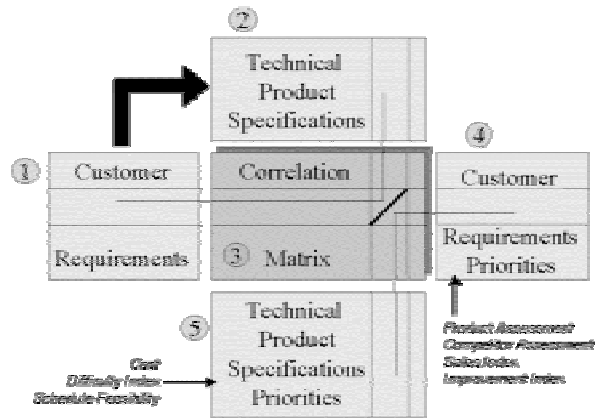


MSWE 609

74

Software QFD

Basic SQFD model



MSWE 609

75

Comparing Quality Frameworks

<u>Criteria</u>	QFD Approach	SQM Approach	GQM Approach
<u>Scope of Goals</u>			
Objects	products	final product	any
Purpose	plan (engineer, control)	certify	any
Viewpoint	customer, user	customer, user	any
<u>Identification of Metrics</u>			
Paradigm	Trace user characteristics of final product into related product/process characteristics at various stages of development	Refine factors into criteria and metrics	Refine goals into questions and metrics
Mechanism	UC: select/tailor PC: select/tailor	F: select C: select M: select	G: select/tailor Q: select/tailor M: select/tailor
<u>Scope of the Approach</u>			
Use	Quality Management	Quality Management	Quality and Project Management

MSWE 609

76