OVERCOME MEASUREMENT CHALLENGES WITH FUNCTION POINTS & CMMI^{SM} –

Presented by

Carol A. Dekkers, President
Quality Plus Technologies, Inc.

Phone (727) 393-6048
Email:dekkers@qualityplustech.com
URL: www.qualityplustech.com

BACKGROUND

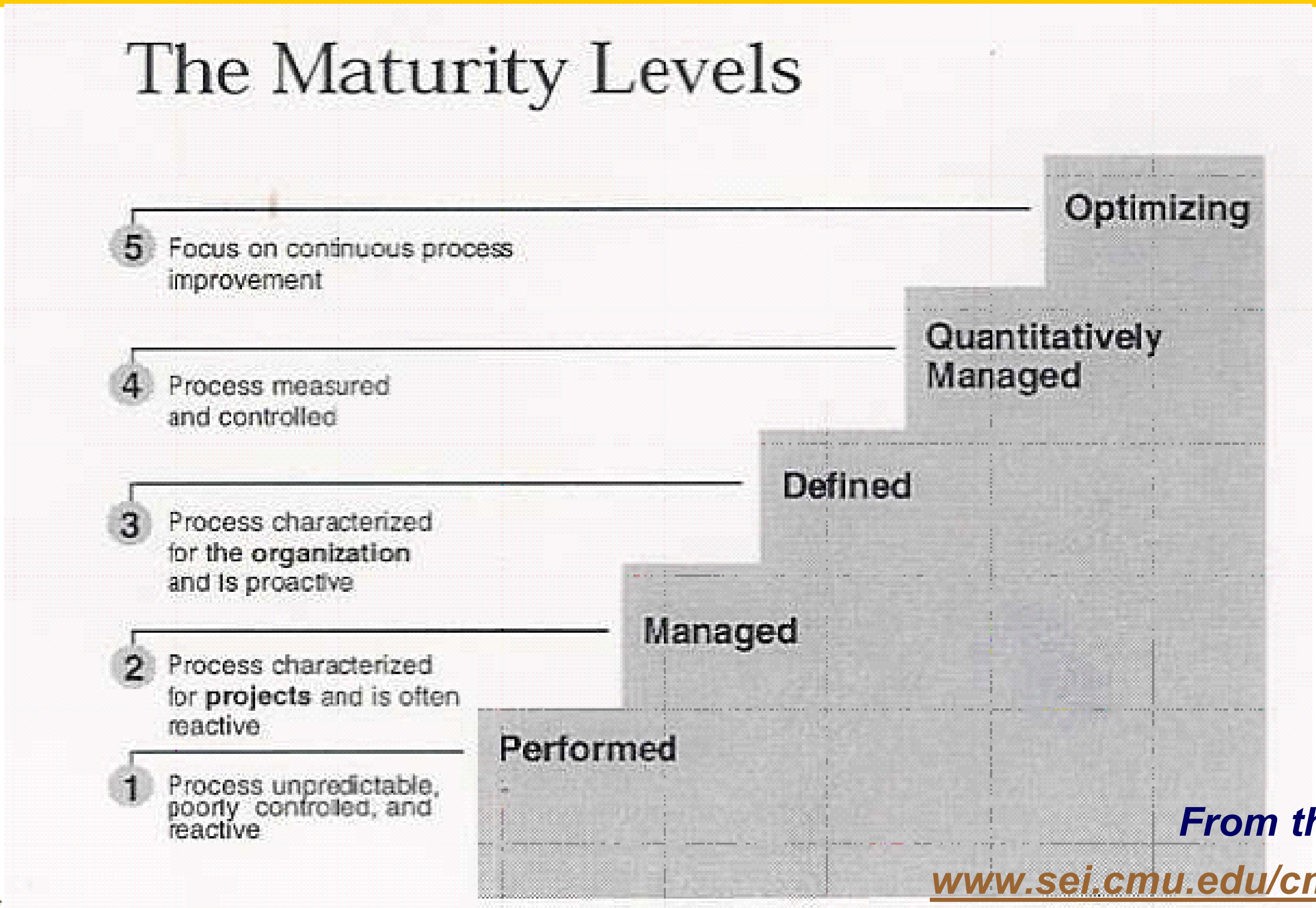
- ❑ Active in ISO/IEC JTC1 SC7 WG12 (Functional Size Measurement) since 1994:
  - ❑ ISO project editor: ISO/IEC 14143-5: Determination of Functional Domains for Use with Functional Size Measurement (TR2, 2003)
  - ❑ IFPUG Category C Liaison Representative to WG12
  - ❑ U.S. Delegate to SC7 / WG12
- ❑ Experience:
  - ❑ Past president of IFPUG, P.Eng. (Canada), S/W Development & Project Management, Function Point Analysis, ISBSG, Software and Systems Measurement. Author of >60 published articles.

AGENDA

- ❒ CMMI℠ and Measurement Data
- ❒ Terminology
- ❒ Context for Function Points & Software Size
- ❒ CMMI℠ levels 1-5
  - ▼ PA’s and sample FP-based metrics
- ❒ Summary

CMMI^SM AND MEASUREMENT DATA

CMMI^{SM} Staged Representation
5 Maturity Levels – 24 Process Areas



The Maturity Levels

5 Focus on continuous process improvement — Optimizing

4 Process measured and controlled — Quantitatively Managed

3 Process characterized for the **organization** and is proactive — Defined

2 Process characterized for **projects** and is often reactive — Managed

1 Process unpredictable, poorly controlled, and reactive — Performed

From the SEI website

www.sei.cmu.edu/cmm/cmm.html

CAPABILITY MATURITY MODEL CMMI^SM

INTEGRATED SYSTEMS/SOFTWARE ENGINEERING

1. **Performed:**

   Level 1 process is a performed process that is often ad hoc and occasionally chaotic. The competence and heroics of the people doing the work drive how the activities are performed.

   measures?

2. **Managed:**

   Level 2 process is a managed process that is planned, performed, monitored, and controlled for individual projects, groups, or standalone processes to achieve a given purpose. Managing the process achieves both the specific goals for the process as well as other goals, such as cost, schedule, and quality.

   measures?

3. **Defined:**

   Level 3 process is a defined process that is tailored from the organization's set of standard processes. Deviations beyond those allowed by the tailoring guidelines are documented, justified, reviewed, and approved.

   measures?

CAPABILITY MATURITY MODEL CMMI℠

INTEGRATED SYSTEMS/SOFTWARE ENGINEERING

4. **Quantitatively Managed:**

Level 4 process is a <u>quantitatively managed</u> process that is controlled using statistical and other quantitative techniques. Product quality, service quality, and process performance are understood in statistical terms and are controlled throughout the life cycle.

measures?

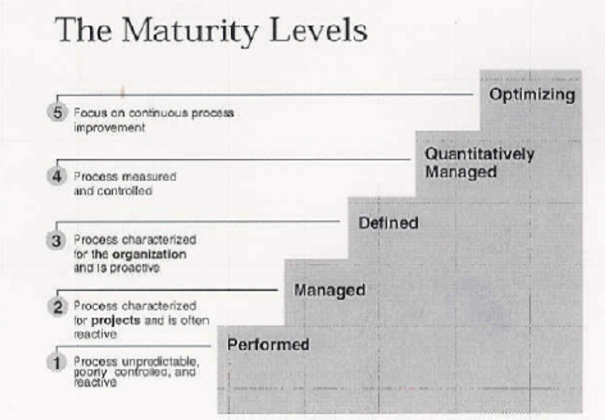
5. **Optimizing:**

Level 5 process is a <u>quantitatively managed process that is improved based on an understanding of the common causes of process variation</u> inherent in the process. An optimizing process focuses on continually improving the range of process performance through both incremental and innovative improvements.

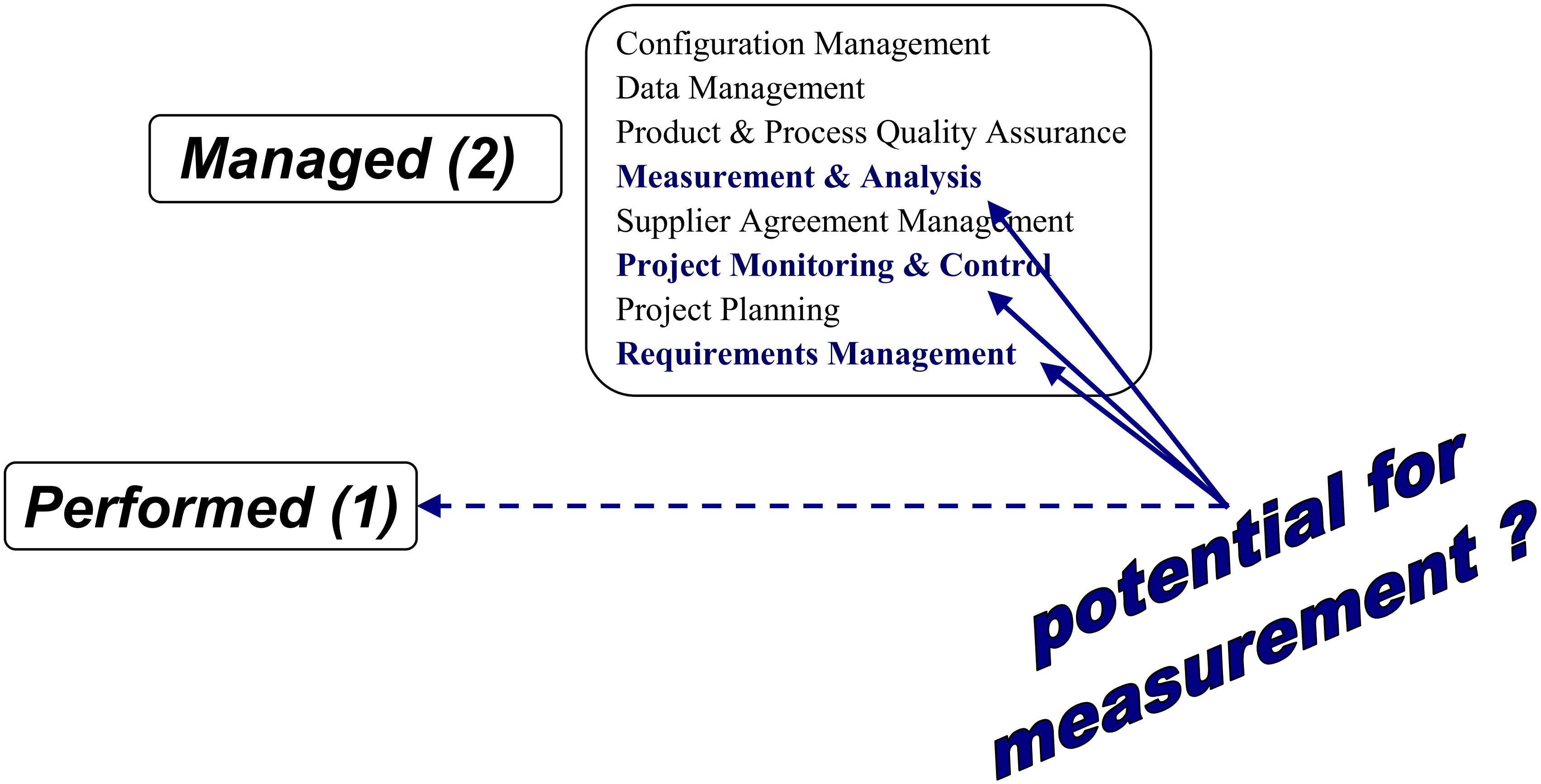
measures?

CAPABILITY MATURITY MODEL CMMI℠
INTEGRATED SYSTEMS/SOFTWARE ENGINEERING

Predictability, effectiveness, and control of an organization's software/systems *processes* are believed to improve as the organization moves up these five levels. While not rigorous, the empirical evidence to date supports this belief.


The Maturity Levels
5 Focus on continuous process improvement
Optimizing
4 Process measured and controlled
Quantitatively Managed
3 Process characterized for the organization and is proactive
Defined
2 Process characterized for projects and is often reactive
Managed
1 Process unpredictable, poorly controlled, and reactive
Performed

Capability Maturity Model CMMI℠
Integrated Systems/Software Engineering



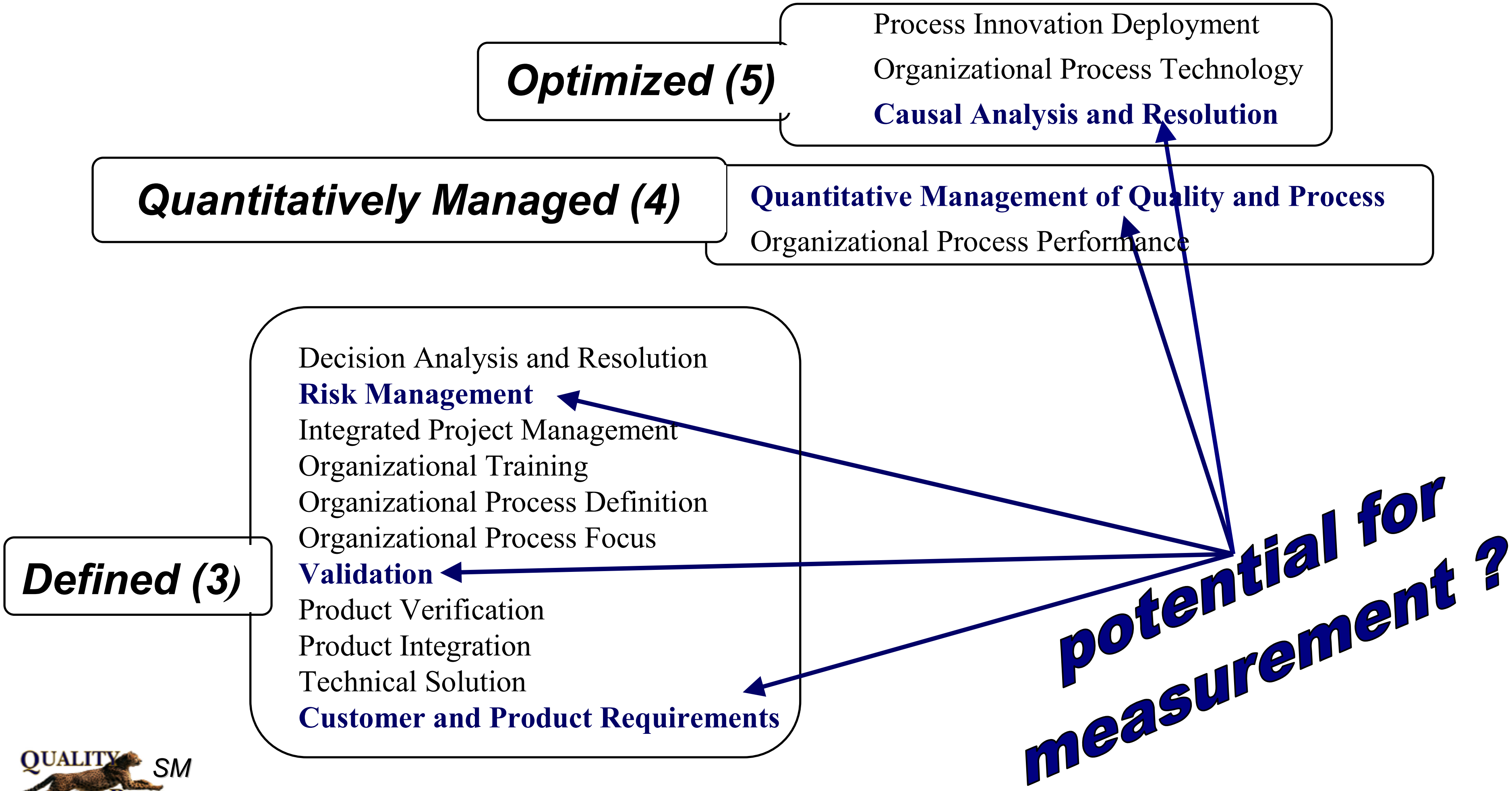
Managed (2)

Configuration Management
Data Management
Product & Process Quality Assurance
Measurement & Analysis
Supplier Agreement Management
Project Monitoring & Control
Project Planning
Requirements Management

Performed (1)

potential for measurement ?

CAPABILITY MATURITY MODEL CMMI℠
INTEGRATED SYSTEMS/SOFTWARE ENGINEERING



Optimized (5)

- Process Innovation Deployment
- Organizational Process Technology
- **Causal Analysis and Resolution**

Quantitatively Managed (4)

- **Quantitative Management of Quality and Process**
- Organizational Process Performance

Defined (3)

- Decision Analysis and Resolution
- **Risk Management**
- Integrated Project Management
- Organizational Training
- Organizational Process Definition
- Organizational Process Focus
- **Validation**
- Product Verification
- Product Integration
- Technical Solution
- **Customer and Product Requirements**

potential for measurement ?

QUALITY PLUS TECHNOLOGIES, INC. SM

TERMINOLOGY

TERMINOLOGY: "USER"

USER --

Any person, other application, department, etc. – anything with the requirement to 'interface' (i.e., send/receive data), with the system (product)

Terminology: Requirements

- ❒ Gilb

  We need to define requirements in terms of desired future end states: testable & measurable.

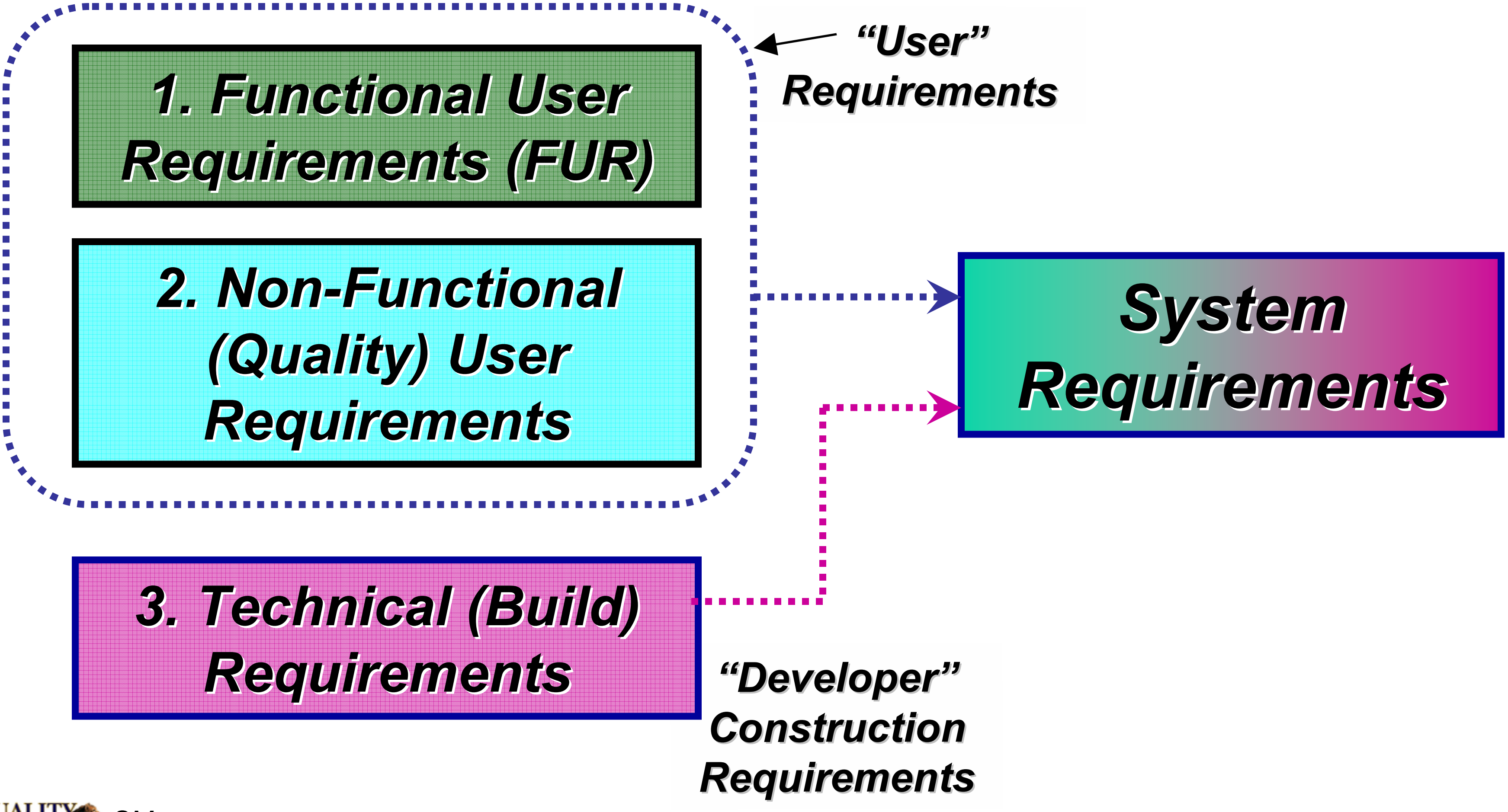
- ❒ Kowal

  Behavior models produce “true” sets of requirements that document the customer’s or user’s *expectations* and does not propose a form of the solution. The two most important aspects of requirements: names of the data, and customer expectations of system behavior.

Terminology: Requirements

- IEEE Std 610.12, IEEE Standard Glossary of Software Engineering Terminology
  1. A condition or capability needed by a user to **solve a problem or achieve an objective.**
  2. A condition or capability that must be met or possessed by a **system or system component to satisfy a contract, standard, specification, or other formally imposed documents.**
  3. A documented representation of a condition or capability as in (1) or (2).

REQUIREMENTS INCLUDE THREE DISTINCT TYPES



1. Functional User Requirements (FUR)

2. Non-Functional (Quality) User Requirements

“User” Requirements

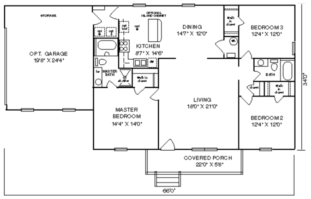
System Requirements

3. Technical (Build) Requirements

“Developer” Construction Requirements

TYPES OF "SOFTWARE" REQUIREMENTS

1. Functional (What software does)
   = FUR → Can be sized with FSM

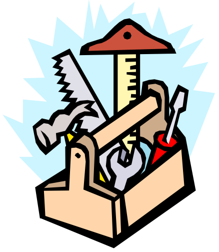


2. Non-functional (How software does)
   = Quality-"ilities"(ISO 9126)→ Not FSM

USER REQUIREMENTS

BUILDER REQUIREMENTS

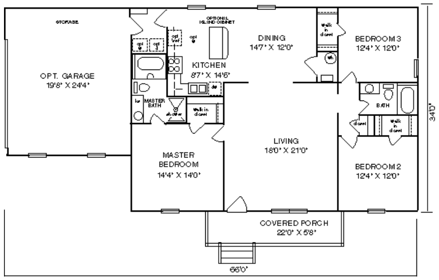
3. Technical (How software is built) →
   Building tools & techniques → Not FSM



QUALITY PLUS TECHNOLOGIES, INC. SM

www.qualityplustech.com

1. FUNCTIONAL USER REQUIREMENTS (FUR)

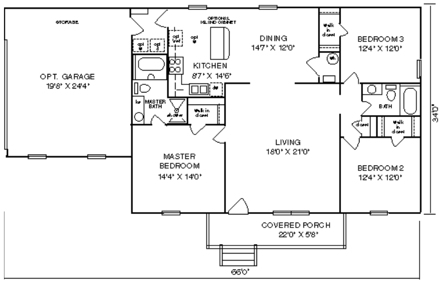


- ❑ WHAT functional processes are performed by or supported by the software. (e.g., record ambient temperature)
- ❑ Responsibility of “users” to specify/confirm
- ❑ Important to remove assumptions (document)
- ❑ The Floor Plan for software

Size of FUR = functional size (FS).
Influences project cost & effort

QUALITY PLUS TECHNOLOGIES, INC. SM

1. Functional User Requirements (FUR)



- ❑ Represent the elementary processes that must be performed by or supported by the software. (e.g., add customer, gauge altitude)
  - ⇨ WHAT the software must do/does
  - ⇨ May be documented by Use Cases, functional specs, ERD, object models, etc.
  - ⇨ Functional size can be quantified using Function Points (FP)

2. NON-FUNCTIONAL (QUALITY) USER REQUIREMENTS

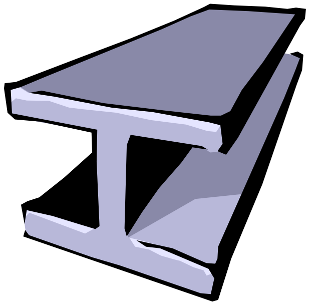
- ❑ HOW software must perform. Complexity. "ILITIES" : *(Suitability, Accuracy, Interoperability, Compliance, Security, Reliability, Efficiency, Maintainability, Portability, Quality in Use, etc.)*
- ❑ Often poorly documented or "sprinkled"
- ❑ Responsibility of "users" to specify / confirm
- ❑ Contracted Specifications for software

NOT part of FS. Influences project cost & effort

2. Non-Functional User Requirements

- Typically not documented (this is a problem) – or documented throughout "requirements" artifacts (can be aided by early discussions)
- Input to estimating model as variables that influence work effort and cost
- Represents problem complexity. Parts are addressed by FP VAF (Value Adjustment Factor).

3. Technical (Build) Requirements



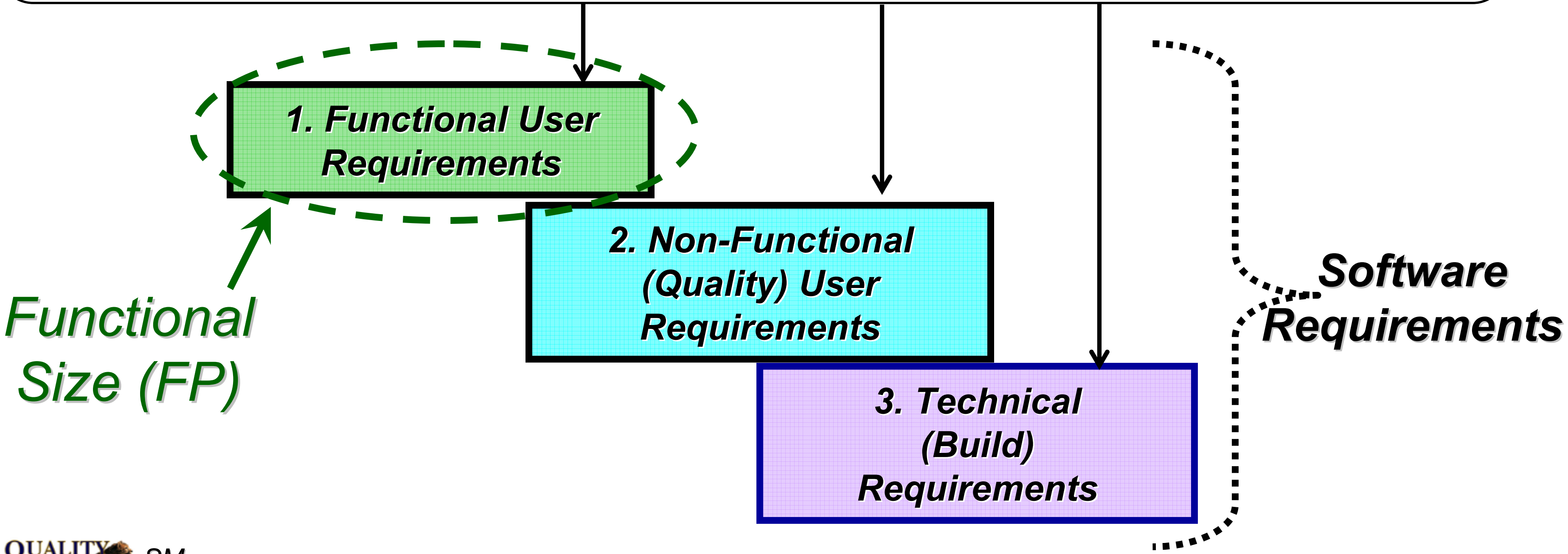
- ❑ How software is developed / built
- ❑ Includes: tools, methods, skills, language, WBS tasks, platforms, software, type of project, etc.
- ❑ Developer responsibility
- ❑ Blueprints (w/ electrical, etc.)

NOT part of FS. Influences project cost & effort

SOFTWARE COST ESTIMATING

Cost or Effort = f (Functional, Non-functional, Technical) * Risk

$$ or hours



1. Functional User Requirements

2. Non-Functional (Quality) User Requirements

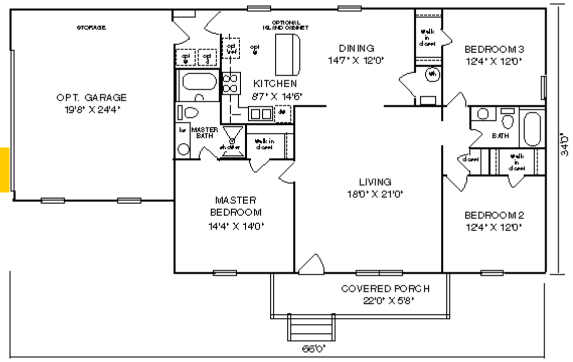
3. Technical (Build) Requirements

Functional Size (FP)

Software Requirements

CONTEXT FOR FUNCTION POINTS & SOFTWARE SIZE

Common Sizing Measure: Function Points (FP)



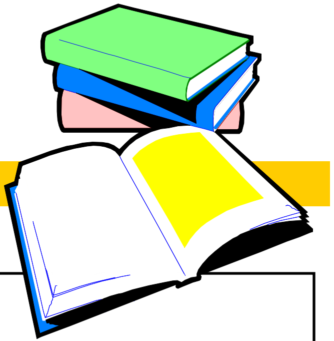
Characteristics:

- ❒ IFPUG 4.1 (unadjusted) = ISO/IEC 20926:2003
- ❒ Sizes functionality the ***user*** requests and receives (software functionality)
- ❒ Functional size = ***independent*** of quality & implementation technology (physical implementation is technical)
- ❒ IFPUG = International Function Point Users Group

\

QUALITY PLUS TECHNOLOGIES, INC. SM

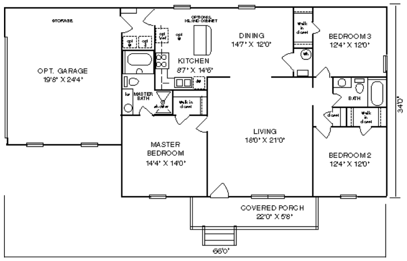
COMMON SIZING MEASURE: FUNCTION POINTS (FP)



- **Function point (FP)***.* A measure which represents the **<u>functional size</u>** of application software.
- **Function point analysis.** A standard method for measuring software development and maintenance from the customer's point of view.

International Function Point Users Group (IFPUG) Counting Practices Manual Release 4.2 (2004)

BASIC CONCEPTS: FUNCTION POINTS

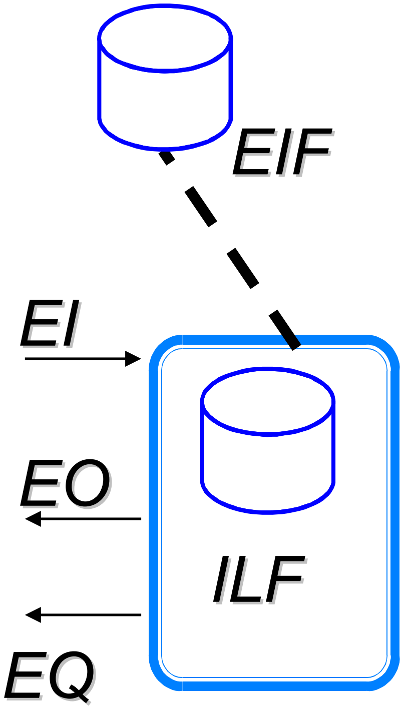


Data Functions:

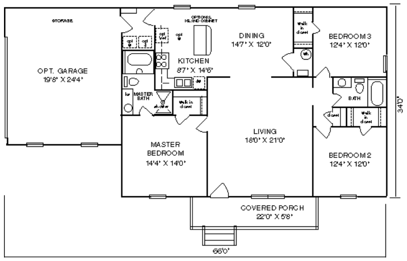
▼ **Internal Logical File**	**ILF**
▼ **External Interface File**	**EIF**

Transactional Functions:

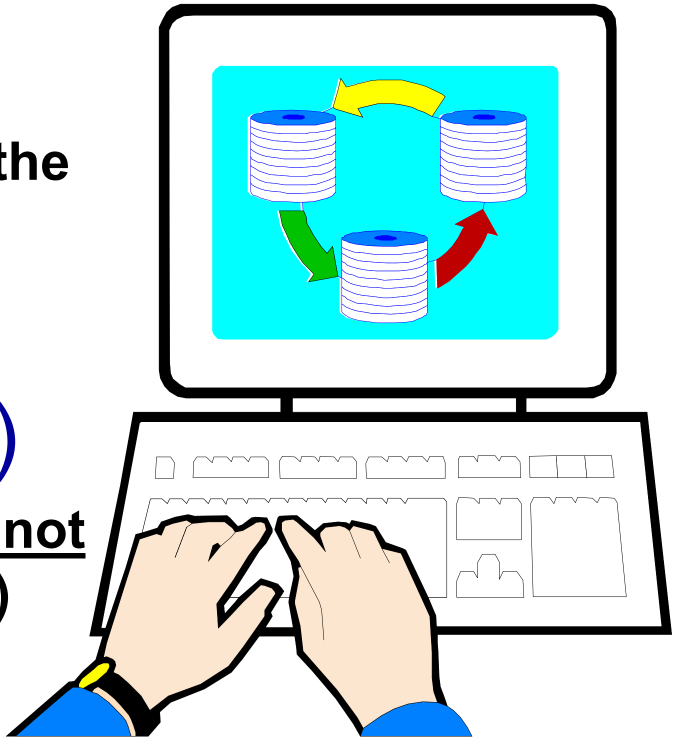
▼ **External Input**	**EI**
▼ **External Output**	**EO**
▼ **External Query**	**EQ**



FP CONCEPTS: WHAT TO COUNT

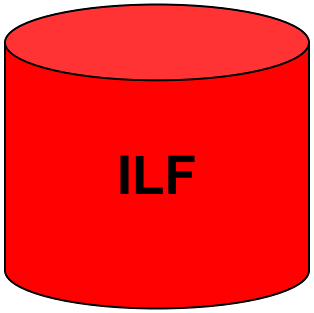


- ❒ Internal Logical File (ILF)
  - ▼ Logical group of data maintained by the application (e.g., Employee object)



- ❒ External Interface File (EIF)
  - ▼ Logical group of data referenced but not maintained (e.g., Employee schedule)

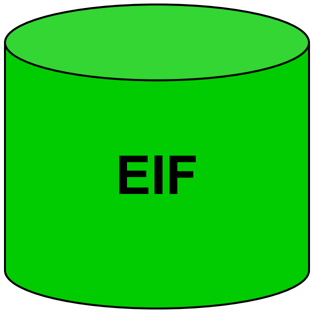
INTERNAL LOGICAL FILE



IFPUG Definition[1]: An **Internal Logical File (ILF) is** *a* ***user identifiable group of logically related data or control information maintained within the boundary of the application. The primary intent*** **of an ILF is to hold data maintained through on one or more elementary processes of the application being counted.**

[1] IFPUG Function Point Counting Practices Manual, Release 4.2

EXTERNAL INTERFACE FILE



IFPUG Definition[1]: An External Interface File (EIF) is *a user identifiable group of logically related data or control information referenced by the application, but maintained within the boundary of another application.* The *primary intent* of an EIF is to hold data referenced through on one or more elementary processes within the boundary of the application being counted. This means an EIF counted for an application must be in an ILF in another application.

[1] IFPUG Function Point Counting Practices Manual, Release 4.2

ILF & EIF COMPLEXITY

	DATA	ELEMENT	TYPES
RECORD ELEMENT TYPES:	**1 to 19 DET**	**20 to 50 DET**	**51 or more DET**
1 RET	Low	Low	Average
2 to 5 RET	Low	Average	High
6 or more RET	Average	High	High

Rate the functional Complexity of the ILF or EIF based on the above RET/DET complexity matrix for data function types.

Example ILF Counts What is the complexity (Low, Average or High) of each logical data group ("File") ?

*** Need to examine FP Matrix (DETs, RETs)

Purchase Order File - maintained

1. Purchase order number (key)
2. Item Description
3. Item Number
4. Quantity Required

LOW (7 FP)

Item File - maintained

1. Item Number (key)
2. Item Description
3. Stock on Hand
4. Stock on Order
5. Monthly Demand
6. Unit Price
7. Unit of Issue
8. Current Vendor
9. Vendor Address
10. Open Purchase Order Number(s)
11. Qty required for each open PO
12. Status for each open PO
13. Vendor Code for each open PO

LOW (7 FP)

EIF EXAMPLES

Corporate holiday file

1. Year (only key)
2. Holiday name
3. Holiday date
4. Date determination (Fixed or floating)
5. Type (National or corporate)

LOW (5 FP)

NOTE: externally maintained.

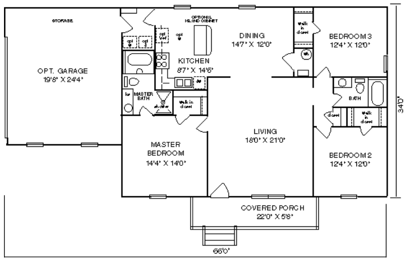
Seasonal message table

1. Seasonal message number (only key)
2. Seasonal message text
3. Type of message
4. Last user id updated

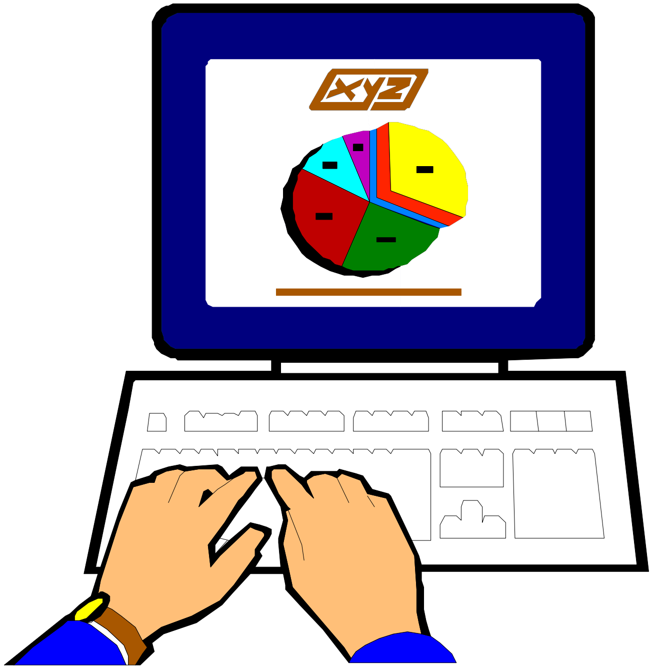
LOW (5 FP)

NOTE: externally maintained.

FP CONCEPTS: WHAT TO COUNT



- ❐ External Input (EI)
  - ▼ Maintains ILF or passes control data into the application.

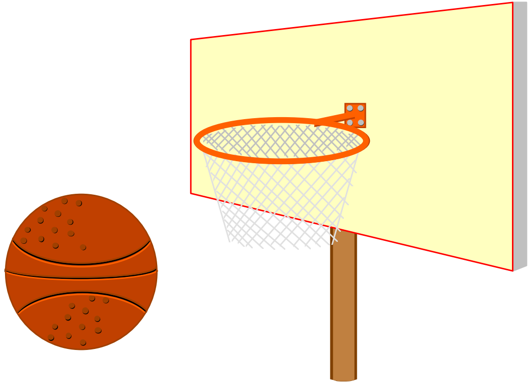


- ❐ External Output (EO)
  - ▼ Formatted data sent out of application with additional processing.
- ❐ External Query (EQ)
  - ▼ Pure data retrieval,

QUALITY PLUS TECHNOLOGIES, INC. SM

External Input

External Input (EI) – An external input (EI) is an *elementary process* that **processes data or *control information* that comes from outside the application boundary**. The *primary intent* of an EI is to maintain one or more ILFs and/or to alter the behavior of the system.



EXTERNAL INPUT

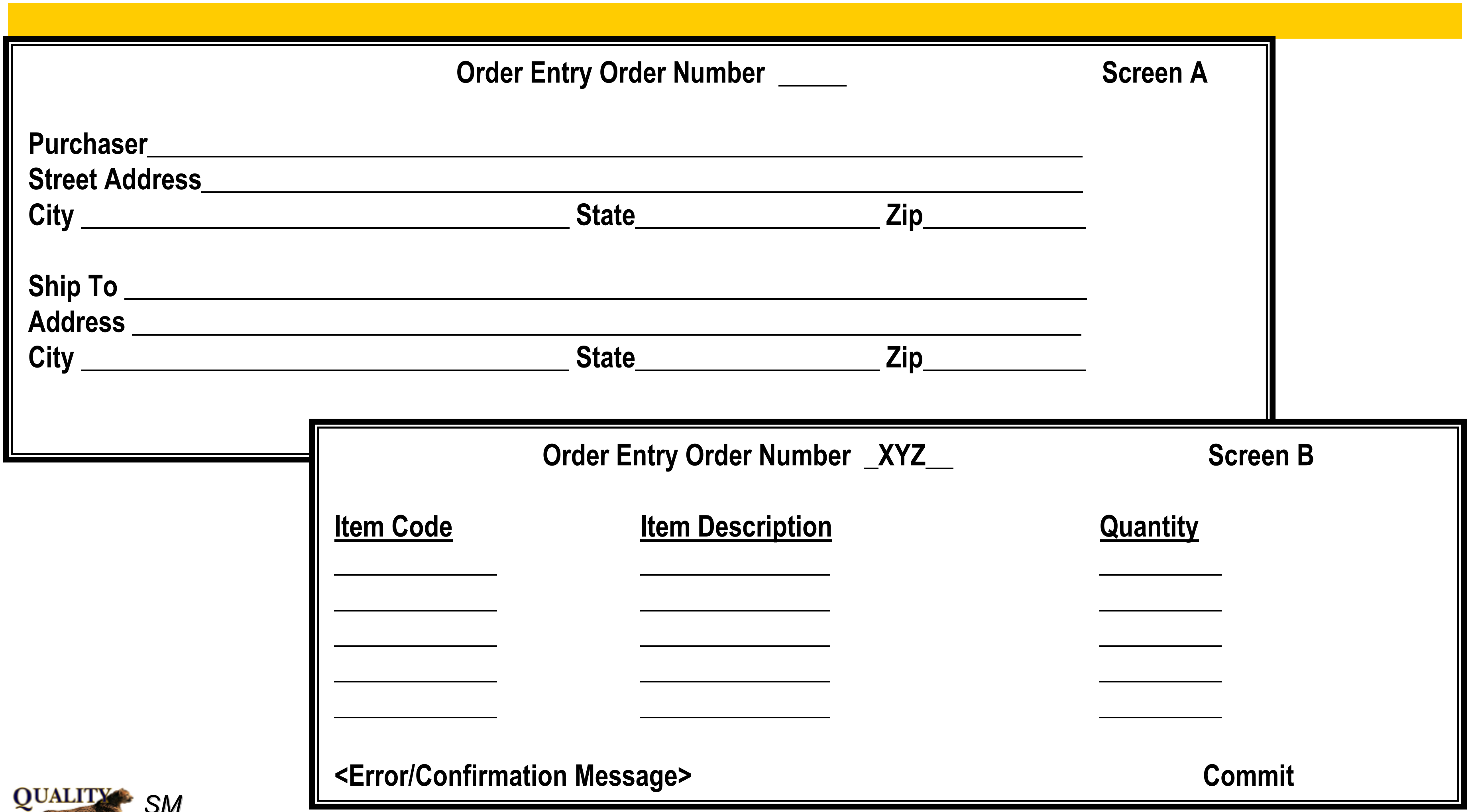
Determining the Complexity of EIs

FTR	**1 to 4 DET**	**5 to 15 DET**	**16 or more DET**
0 to 1 FTR	Low	Low	Average
2 FTRs	Low	Average	High
3 or more FTRs	Average	High	High

External Input Example

To complete an order, the user enters all of the underlined fields of data in screens A & B (next page). Both screens must be completed. Screen B automatically comes up when screen A is completed and the user enters all three fields, then presses enter to commit the order. This process uses the order ILF (referenced and maintained) and item EIF (to validate item code). This is to add an order only. Error messages are from a hard coded error file.

EI EXAMPLE



Order Entry Order Number _____ Screen A

Purchaser____
Street Address____
City ________________________________ State________________ Zip___________

Ship To ____
Address ____
City ________________________________ State________________ Zip___________

Order Entry Order Number _XYZ__ Screen B

Item Code	Item Description	Quantity
___________	_____________	________
___________	_____________	________
___________	_____________	________
___________	_____________	________
___________	_____________	________

<Error/Confirmation Message> Commit

External Output

External Output (EO) – An external output (EO) is an *elementary process* that **sends data or *control information* outside the application boundary**. The *primary intent* of an EO is to present information to a user through processing logic other than, or in addition to, the retrieval of data or control information.

The processing logic must contain at least one mathematical formula or calculation, or create derived data. An external output may also maintain one or more ILFs and/or alter the behavior of the system.

EXTERNAL QUERY

External Inquiry (EQ) – An external inquiry (EQ) is an *elementary process* that **sends data or *control information* outside the application boundary**. The *primary intent* of an EQ is to present information to a user through retrieval of data or control information from an ILF or EIF.

The processing logic contains NO mathematical formulas or calculations, and creates no derived data. NO ILF is maintained during the processing, nor is behavior of the system altered.

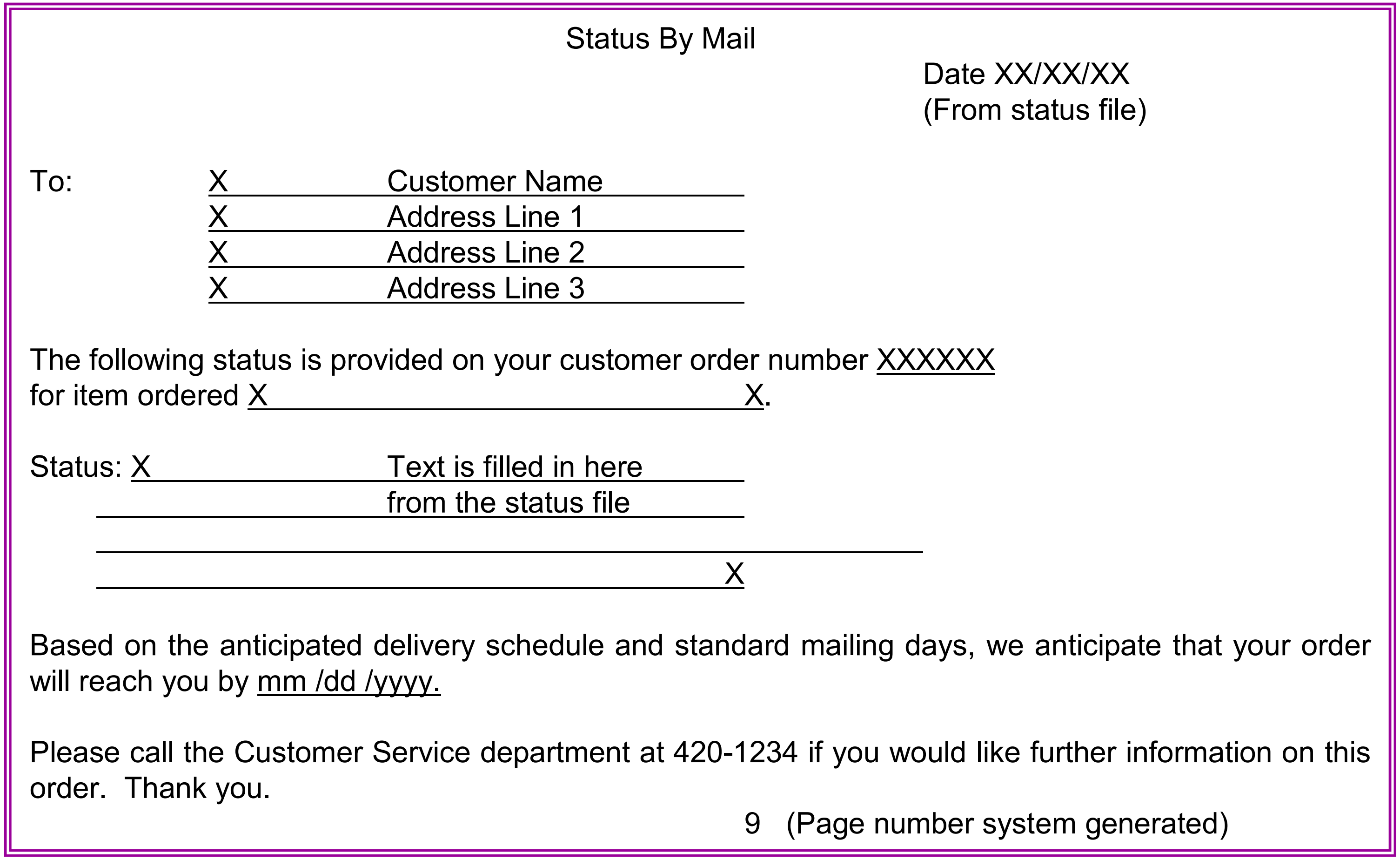
Determining the Complexity of EOs and EQs

FTR	**1 to 5 DET**	**6 to 19 DET**	**20 or more DET**
0 to 1 FTR	Low	Low	Average
2 to 3 FTRs	Low	Average	High
4 or more FTRs	Average	High	High

EXTERNAL OUTPUT EXAMPLE

The customer name/address are from the Customer ILF; the order # & item are from the Order ILF; status and status date are from the Status ILF; and the calculated date is determined by using the Order ILF and a hard coded algorithm.

System Generated Letter



Status By Mail

Date XX/XX/XX
(From status file)

To: X Customer Name
X Address Line 1
X Address Line 2
X Address Line 3

The following status is provided on your customer order number XXXXXX
for item ordered X X.

Status: X Text is filled in here
from the status file

X

Based on the anticipated delivery schedule and standard mailing days, we anticipate that your order will reach you by mm /dd /yyyy.

Please call the Customer Service department at 420-1234 if you would like further information on this order. Thank you.

9 (Page number system generated)

EXTERNAL QUERY EXAMPLE

Catalog Order

Enter customer number 99999

-

[When entering customer order number, 2 potential error messages may be returned.]

Select System Help
Retrieve Data
Exit

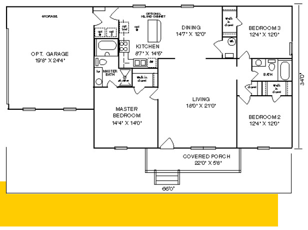
Retrieve will return the following information from the saved customer order:

Catalog Order for Customer Number 99999

Customer Name	Ship to Address	Item ordered	Customer Order Number
xxxxxxxxxx	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999
	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999
	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999
	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999
	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999
	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999
	xxxxxxxxxxxxxxx	xxxxxxxxxx	99999999999

Select System Help
Exit
Scroll through list
Select correct customer order number and return to follow-up screen

FP Concepts: Values of Each Function



❒ Function Weights (Unadjusted FP)

Function Type	Low	Average	High
EI	x 3	x 4	x 6
EO	x 4	x 5	x 7
EQ	x 3	x 4	x 6
ILF	x 7	x 10	x 15
EIF	x 5	x 7	x 10

FP VERSUS DEVELOPMENT TERMINOLOGY: “FILE”

❒FP Meaning

- ▼*Logical* data entity
- ▼*Logical* grouping of data as in ILF, EIF
- ▼*Not one to one* with physical file, dataset, database, table, flat file, etc.

❒IT Meaning

- ▼Dataset, tape, table, formatted storage of data
- ▼*Physical* Dataset in DP terms:
  - →Work/Sort file (0 FP)
  - →Master files (ILF)
  - →Reference data (EIF)
  - →Input data TX (EI’s)
  - →Tapes, files out (EO’s)

QUALITY PLUS TECHNOLOGIES, INC. SM

FP CONCEPTS: COUNTING STEPS

1. Determine Type of Count
2. Application Boundary ← SCOPE
3. Determine the Unadjusted FP Cour ← COMPLETENESS
   - *a) Data Functions (ILF, EIF)* ← CONSISTENCY
   - *b) Transaction Functions (EI,EO,EQ)*

ISO/IEC 14143-1:1998 Functional Size Measurement, Definition of Concepts

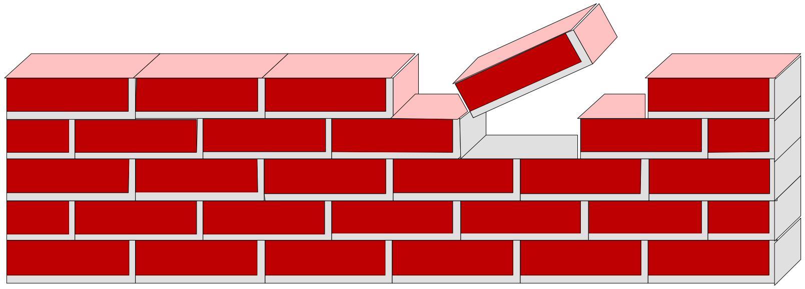
4. Calculate Value Adjustment Factor
5. Final Adjusted FP

Also: SCOPE CREEP, SIZE OF CHANGES, etc.

CMMI^{SM} PA'S AND SAMPLE FP-BASED METRICS

FP Maturity Model

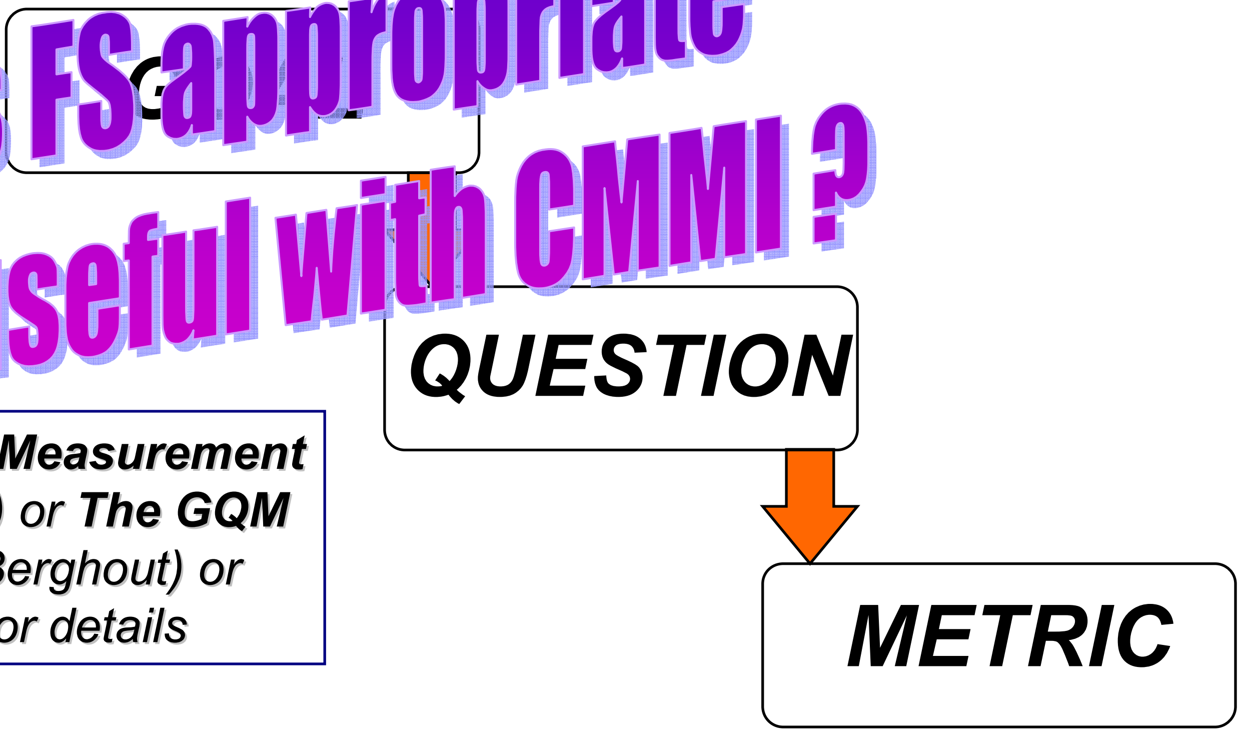
- ❒ As your organization matures, your usage of FPs typically also matures
- ❒ FPs can be used at most levels, as appropriate – determine what is appropriate for where your organization is



QUALITY PLUS TECHNOLOGIES, INC. SM

GOAL – QUESTION – METRIC (VICTOR BASILI, U OF MD)

- To determine “appropriate” metric or indicator



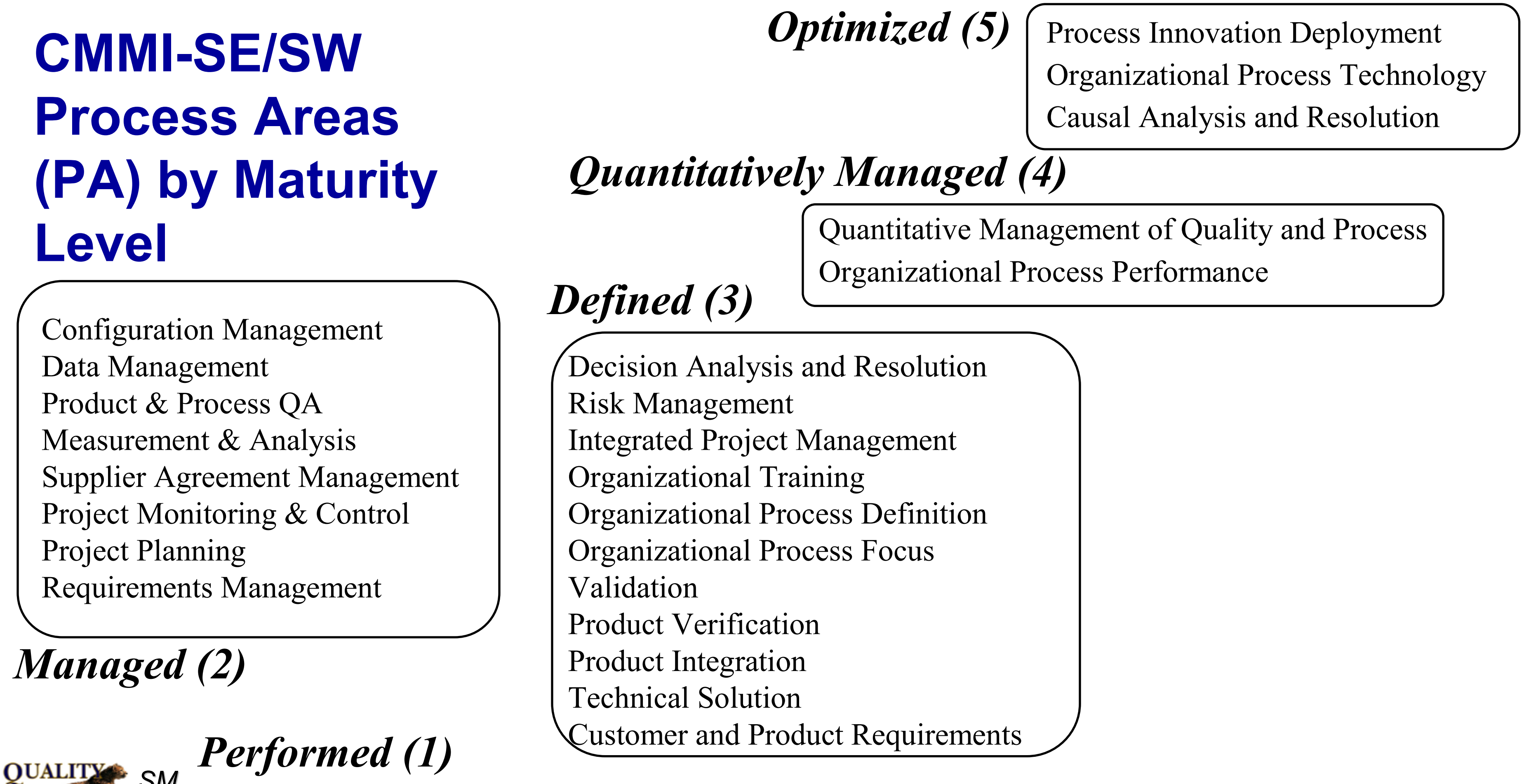
Is FS appropriate or useful with CMMI ?

Refer to: ***SEI Goal Driven Measurement*** *or* ***PSM*** *(www.psmsc.com) or* ***The GQM Method (****VanSolingen/Berghout) or writings of Vic Basili for details*

CAPABILITY MATURITY MODEL ® CMMI

INTEGRATED SYSTEMS/SOFTWARE ENGINEERING

CMMI-SE/SW Process Areas (PA) by Maturity Level



Optimized (5)

- Process Innovation Deployment
- Organizational Process Technology
- Causal Analysis and Resolution

Quantitatively Managed (4)

- Quantitative Management of Quality and Process
- Organizational Process Performance

Defined (3)

- Decision Analysis and Resolution
- Risk Management
- Integrated Project Management
- Organizational Training
- Organizational Process Definition
- Organizational Process Focus
- Validation
- Product Verification
- Product Integration
- Technical Solution
- Customer and Product Requirements

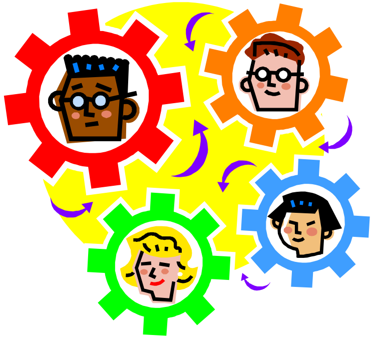
Managed (2)

- Configuration Management
- Data Management
- Product & Process QA
- Measurement & Analysis
- Supplier Agreement Management
- Project Monitoring & Control
- Project Planning
- Requirements Management

Performed (1)

CMMI Level 1 - Performed

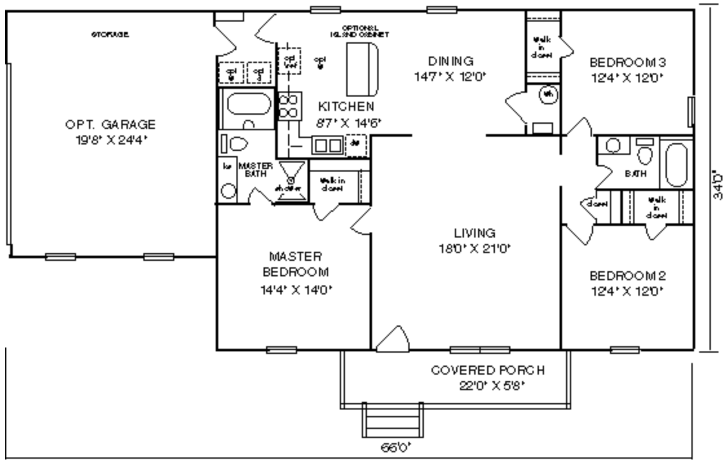
- ❐ Level 1 process is a performed process that is often ad hoc and occasionally chaotic. The competence and heroics of the people doing the work drive how the activities are performed.



CMMI - Level 1 - Performed

Sample FP measures

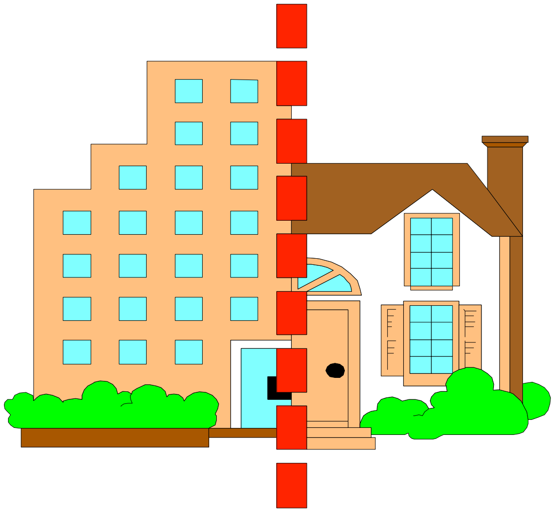
- ❐ Size of Applications or Projects underway in FPs



CMMI - Level 1 - Performed

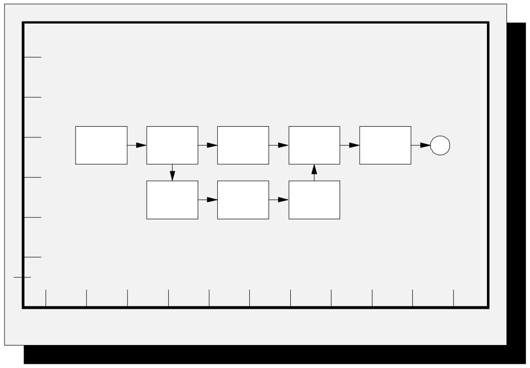
Sample <u>project</u> measures

- ❒ Size of Project (s) in FPs



CMMI - Level 2 - Managed

- Level 2 process is a managed process that is planned, performed, monitored, and controlled for individual projects, groups, or standalone processes to achieve a given purpose. Managing the process achieves both the specific goals for the process as well as other goals, such as cost, schedule, and quality.



CMMI Level 2 - Managed

- **Requirements Management** - Use FPs to quantify the size of the functional requirements
- **Project Planning** - Use FPs with other variables (non-functional, technical, risk) to estimate effort and cost
- **Project Monitoring and Control** - Use FPs to keep track of scope changes, % complete, etc.
- **Supplier Agreement Management** - Use FPs to define size of requirements (application/project being outsourced) and for monitoring supplier's progress; evaluate alternatives for outsourcing; writing service level agreements (SLAs)

CMMI Level 2 – Managed (contd)

- **Measurement & Analysis** - Collecting FP data in a central repository, reporting data to managers /PMs, understanding how FPs can/cannot be used
- **Process and Product Software Quality Assurance** - Defect tracking (defect density)
- **Data Management** - Feeds into the FPA and provides information about data stores.
- **Configuration Management** - FPs track changes to requirements & measure impact on project size

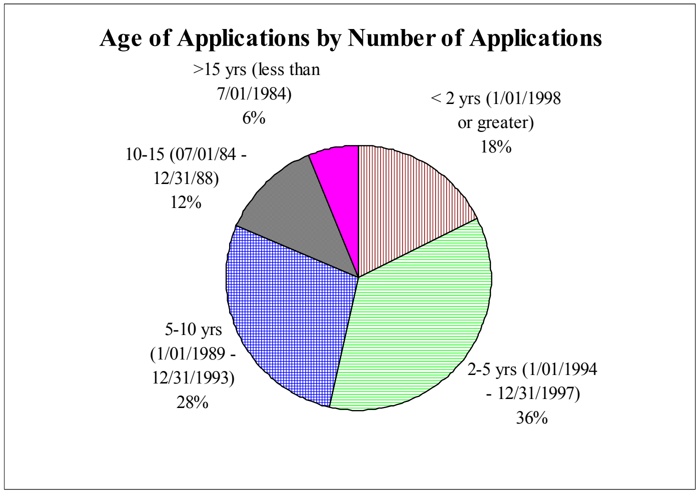
CMMI - LEVEL 2

- Sample measures:
  - (SEI) CAPABILITY MATURITY MODEL Level 2:
    - *Requirements Management:*
      - Measurements are made to assess the **status of requirements** (are they reviewed/accepted/rejected)
      - Change activity amounts (**Size of change requests**)
    - *Project Planning:*
      - **Estimates for size** of software work products or changes must be derived according to documented procedure.

CMMI - Level 2 - Managed

Sample measures

- **Portfolio Size and Growth Trends** - % increase/change between dates
- **Age of Applications** - % older than 5 years, etc.
- Application **Support Rates** - # FPs per resource
- **Portfolio breakdown** - FPs by Language, Platform: % breakdown
- **Application Churn** - # CRs (size / app size)



CMMI - Level 2 - Managed

Sample project measures

- **Delivery Rates / Duration Delivery Rates** – FPs/hour or FPs/month
- **Productivity** – Hours per FP
- **Stability Ratio, Scope creep/requirements volatility** - % FPs Added, Changed, Deleted over time
- **Project Cost per FP** - $ per FP
- **Defect Ratio** – defects per FP
- **Testing Proficiency Ratios** - % defects found pre-delivery/total defects (up to 1 month post-delivery)

CMMI - Level 3 - Defined

- Level 3 process is a defined process that is tailored from the organization's set of standard processes. Deviations beyond those allowed by the tailoring guidelines are documented, justified, reviewed, and approved.

CMMI - LEVEL 3

(SEI) CAPABILITY MATURITY MODEL LEVEL 3

→*Level 3 (Defined)*

- Includes formal software measurement program, repeatable processes

▼Many level 1 companies fail to reach level 2:

'Lack of measurement to aid project planning, tracking & oversight, requirements management. One of most often missed measures is that of the "size" of various tasks/products to be performed/produced.'

Ken Dymond on Using and Implementing the CMM (11/98) as cited by G.Rule, SMS Ltd.

CMMI Level 3 - Defined

- **Customer and Product Requirements** - FP analysis process serves as a design walk-through of requirements
- **Technical Solution** - Review FPA for design issues and for documenting interfaces and project documentation
- **Product Integration** - Use FPs to measure interface with other applications
- **Product Verification** - Update FPs based on verification process
- **Validation** - Track defect data and use FPs as denominator
- **Organizational Process Focus** - Include FPA as a standard organizational process

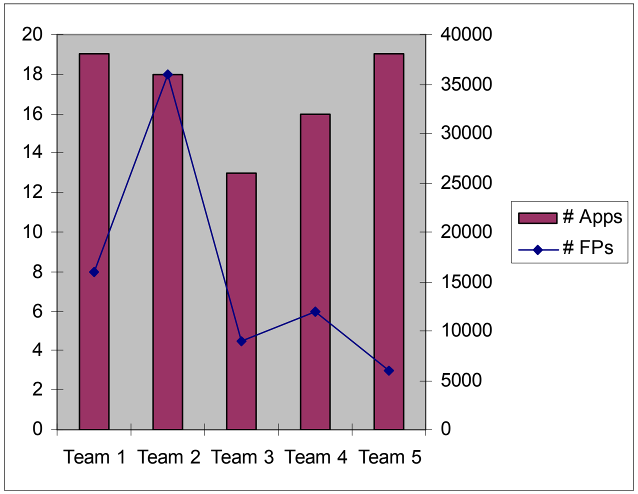
CMMI Level 3 – Defined (contd)

- **Organizational Process Definition** - Include FPs as part of the organizational measurement repository
- **Organizational Training** – n/a
- **Integrated Project Management** – n/a
- **Risk Management** - Using FPs to measure the size of the application will enable assessing risk changes when the size changes
- **Decision Analysis and Resolution** - Using FPs to assist in decision making, e.g., impact of scope changes on schedule, extend schedule, increase resources, renegotiate scope…

CMMI - Level 3 - Defined

Sample measures

- Application **Support Rates** for Organization
- Support **Activity Trends**
- Application **Maintenance Load per Person**
- Application **Maintenance Cost per FP**
- **Mean Time to Repair**



CMMI - Level 3 - Defined

Sample project metrics - compare projects

- ❒ **Delivery Rates** by type of project (development, enhancement, maintenance, language, platform, etc.) (hours per FP)
- ❒ **Duration Delivery Rates** by type of project (elapsed duration)
- ❒ Trending of delivery rates
- ❒ Testing Proficiency Ratios
- ❒ **Scope creep/requirements volatility** - comparisons between projects
- ❒ **Defect Density**

CMMI - Level 4
Quantitatively Managed

❐ Level 4 process is a quantitatively managed process that is controlled using statistical and other quantitative techniques. Product quality, service quality, and process performance are understood in statistical terms and are controlled throughout the life cycle.

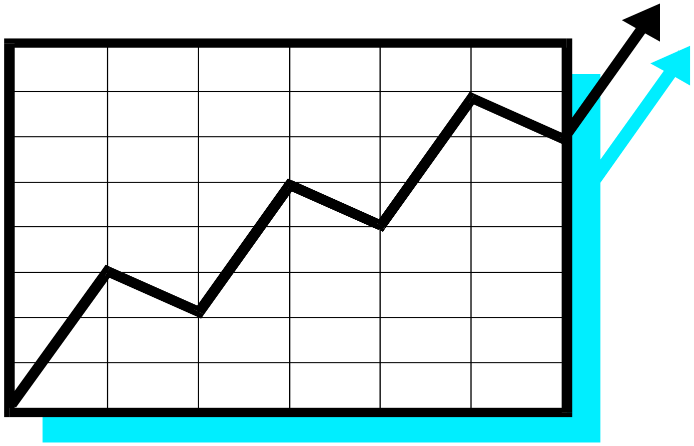
CMMI LEVEL 4
QUANTITATIVELY MANAGED

- ❐ **Quantitative Management of Quality and Process** - use FPs as the common denominator for selecting measures and statistical control of products and quality
- ❐ **Organizational Process Performance** - use FPs as one of the measure for establishing performance baselines for processes.

CMMI - Level 4
Quantitatively Managed

Sample measures

- ❐ Enterprise **Productivity Rates**
- ❐ Enterprise **Quality Rates**
- ❐ Enterprise **Cost Per FP**
- ❐ Mean Time to Failure
- ❐ Mean Time to Repair

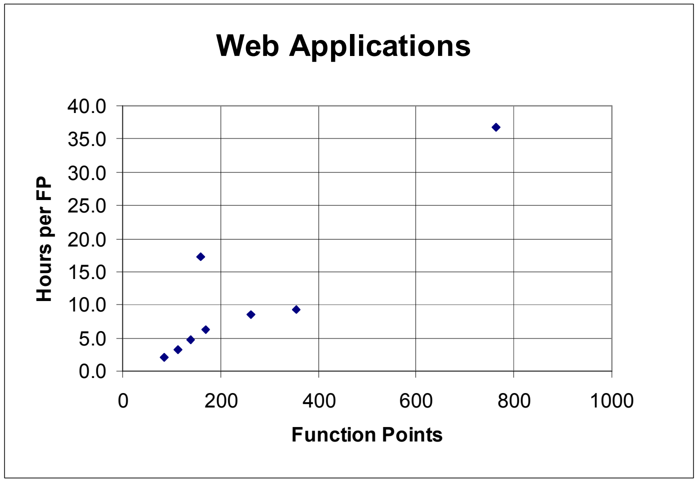


QUALITY PLUS TECHNOLOGIES, INC. SM

CMMI - Level 4 - Quantitatively Managed

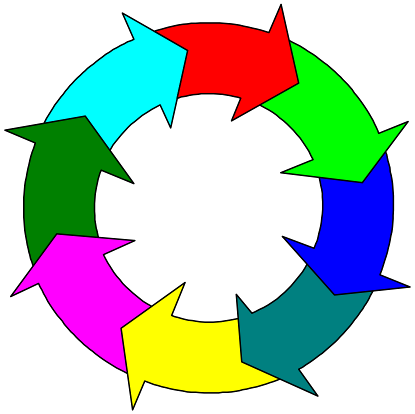
Sample project measures

- ❒ Enterprise **Project Delivery Rates**
- ❒ Enterprise **Project Quality Rates**
- ❒ Enterprise **Project types Cost Per FP**
- ❒ Statistical Process Control of Delivery Rates (System development processes)
- ❒ Trend Analysis



CMMI Level 5 - Optimized

- ❒ Level 5 process is a quantitatively managed process that is improved based on an understanding of the common causes of process variation inherent in the process. An optimizing process focuses on continually improving the range of process performance through both incremental and innovative improvements.



CMMI Level 5 - Optimized

- **Causal Analysis and Resolution** - Analyze **defect data using FPs as the common denominator** enables comparisons between projects; allows evaluating the impact of changes also by using FPs as the common denominator
- **Organizational Process Technology Innovation** - use FPs as a **basic size normalizing measure for establishing process improvement objectives**

CMMI - Level 5 - Optimized

Sample measures

- ❒ Repair **Cost Ratio**
- ❒ **Defect Density**
- ❒ **Cumulative Defects** (size as % of whole)
- ❒ **Defect Distribution** by Severity
- ❒ **Defects by Cause** (phase)
- ❒ Mean Time between Defects
- ❒ Application **Support Rate Trends** after process improvements

CMMI - Level 5 - Optimized

Sample project measures

- **Defect Detection Ratio** (by phase found) aka inspection effectiveness
- **Defect Removal Efficiency**
- **Defect Distribution by Severity**
- **Defects by Cause**
- **Delivery Rate Trends** after process improvements
- SPC of defect data
- **Cost of defect removal by inspection phase**
- SPC of process info
- Software quality - **post-release defect density**

SUMMARY

SUMMARY

What level is your organization?

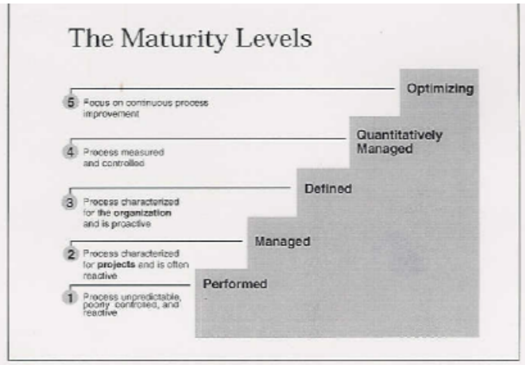

The Maturity Levels
Optimizing
5 Focus on continuous process improvement
Quantitatively Managed
4 Process measured and controlled
Defined
3 Process characterized for the organization and is proactive
Managed
2 Process characterized for projects and is often reactive
Performed
1 Process unpredictable, poorly controlled, and reactive

Figure 2: Maturity Levels

- ❐ Understanding role of FS can help establish SMART goals
  - ▼ strategic, measurable, achievable /actionable, realistic, and time bounded
- ❐ Apply **<u>appropriate metrics</u> based on goal-question-metric priorities**

IF FUNCTIONAL SIZE-BASED MEASUREMENT IS APPROPRIATE:

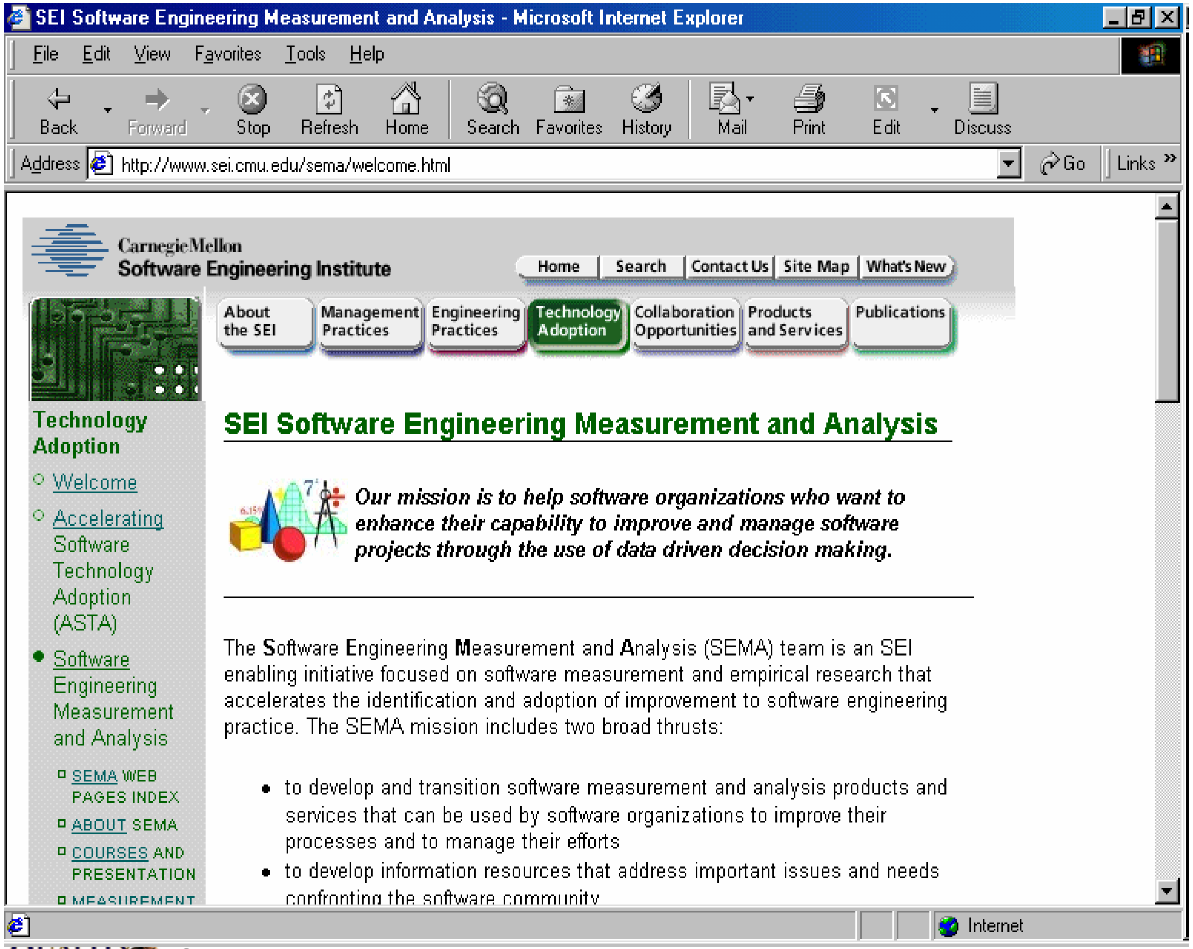
- ❒ Obtain **PSM training**
- ❒ Obtain **IFPUG Certified FP training**
- ❒ At level 1 and 2 they want to **investigate** less labor intensive FSM / FP approaches:
  - ▼ Count [illegible] FP functions [illegible] average
  - ▼ One file model (AUDIO) → Rule of 31
  - ▼ "Front firing" from SLOC

Bottom line:

Use FPs if & when appropriate

RESOURCES:

http://www.sei.cmu.edu/sema/welcome.html


SEI Software Engineering Measurement and Analysis - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss
Address http://www.sei.cmu.edu/sema/welcome.html
Go Links
Carnegie Mellon Software Engineering Institute
Home Search Contact Us Site Map What's New
About the SEI
Management Practices
Engineering Practices
Technology Adoption
Collaboration Opportunities
Products and Services
Publications
Technology Adoption
○ Welcome
○ Accelerating Software Technology Adoption (ASTA)
● Software Engineering Measurement and Analysis
▫ SEMA WEB PAGES INDEX
▫ ABOUT SEMA
▫ COURSES AND PRESENTATION
▫ MEASUREMENT
Internet

SEI Software Engineering Measurement and Analysis

Our mission is to help software organizations who want to enhance their capability to improve and manage software projects through the use of data driven decision making.

The Software Engineering Measurement and Analysis (SEMA) team is an SEI enabling initiative focused on software measurement and empirical research that accelerates the identification and adoption of improvement to software engineering practice. The SEMA mission includes two broad thrusts:

- to develop and transition software measurement and analysis products and services that can be used by software organizations to improve their processes and to manage their efforts
- to develop information resources that address important issues and needs confronting the software community

Other References

- **Guidelines to Software Measurement** – Release 2.0 (2004) – IFPUG
- **Software Measurement – Advice from the Experts** (2003) – Addison Wesley
- **Best Practices in Software Measurement (How to use metrics to improve project & process performance)** – Ebert et al (2004) ISBN:3-540-20867-4

QUALITY PLUS TECHNOLOGIES, INC. SM

OVERCOME MEASUREMENT CHALLENGES WITH FUNCTION POINTS & CMMI SM –

Thank You

Carol A. Dekkers
Quality Plus Technologies, Inc.
Phone (727) 393-6048
Email:dekkers@qualityplustech.com
URL: www.qualityplustech.com