

IFPUG

INTERNATIONAL FUNCTION POINT USERS GROUP

www.ifpug.org



The New CPM: 4.3

(Effective January 2010)

IFPUG Counting Practices Committee

ISMA – September 2009

Adri Timp, CFPS

Chair, Counting Practices Committee



Topics

- ▶ **Functional Size Measurement (FSM) Background**
- ▶ **The Next CPM - Status**
- ▶ **CPM 4.2.1 → CPM 4.3 – Overview of Changes**
- ▶ **Part 1: IFPUG FSM What It Looks Like**
- ▶ **Part 2 thru 5: Changes from 4.2.1**
- ▶ **Results from Impact Study**
- ▶ **Certification**
- ▶ **Publication**



ISO Functional Size Measurement (FSM) Background

- ▶ **ISO/IEC 14143-1 Information technology - Software measurement - FSM - Published 1998**
- ▶ **IFPUG PAS Document – CPM 4.1 Based - Published 1999 as initial conforming document**
- ▶ **CPM 4.2 Restructured – Multi-Part – 2004**
 - Part 1: Process and Rules
 - Part 2: Counting Practices
 - Part 3: Examples
 - Part 4: Appendices and Glossary
- ▶ **CPM 4.3 100% ISO compliant – Multi-Part – 2010**



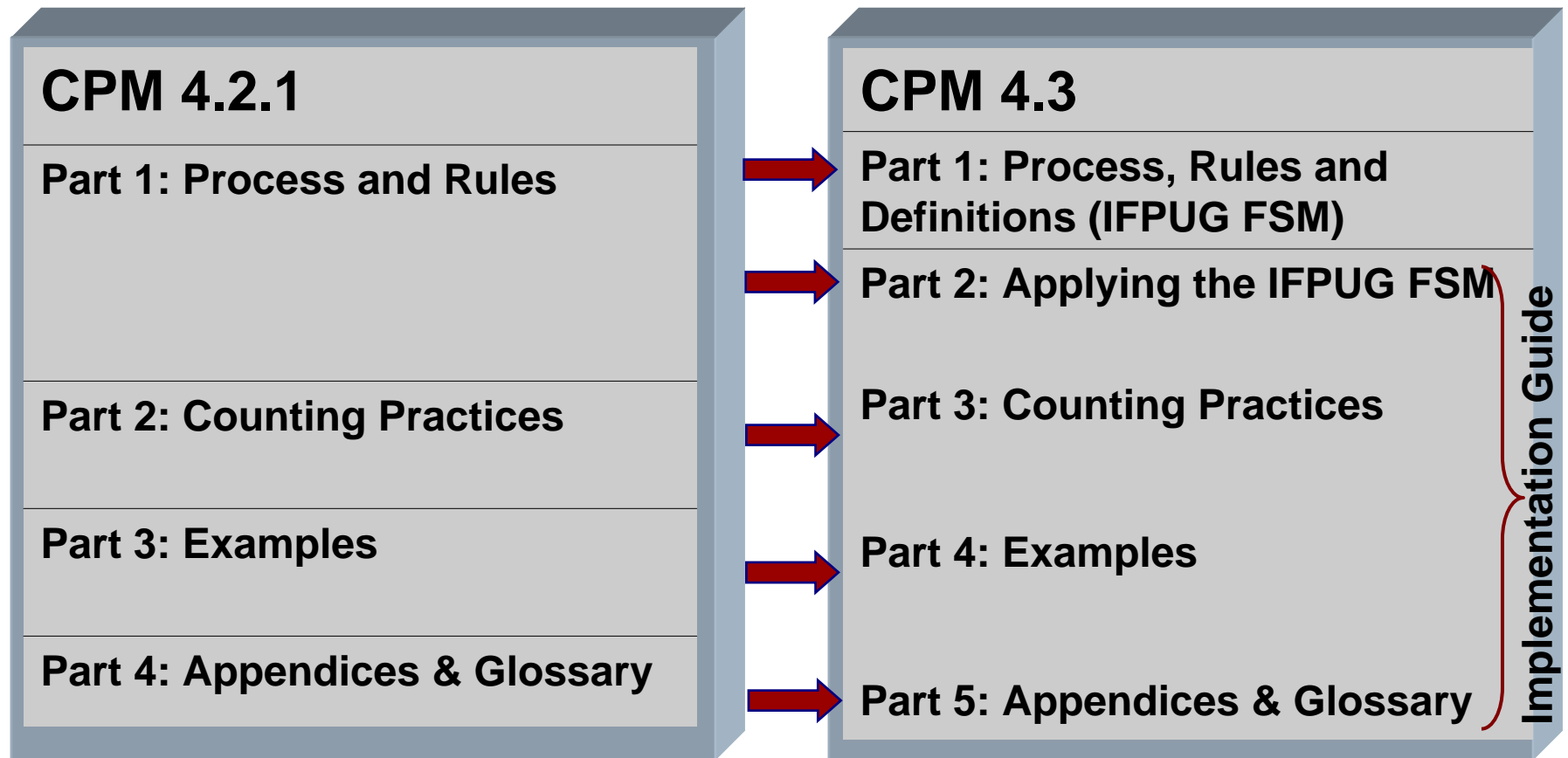
The Next CPM - Status

- ▶ **Extensive internal & external reviews**
- ▶ **Obtained IFPUG Board approval (March 2008)**
- ▶ **Submitted to ISO for Approval (May 2008)**
- ▶ **ISO Approved with Comments (December 2008)**
- ▶ **CPC Processed ISO Comments (January 2009)**
- ▶ **Final ISO Approval (May 2009)**
- ▶ **CPM 4.3 Publication (September 2009)**
- ▶ **CPM 4.3 Effective (January 2010)**



CPM 4.2.1 → CPM 4.3

Overview of Changes



CPM 4.2.1 CPM 4.3

Overview of Changes

- ▶ **Process and Rules split in to two parts (1 and 2)**
- ▶ **Part 1 (new): The ISO Standard (IFPUG FSM)**
- ▶ **Part 2 (former part 1): The Bridge - Applying the IFPUG FSM Method**
 - Retained remaining chapters from Process and Rules
 - Moved GSCs and VAF to Appendix C
 - Changed wording and sequence of actions consistent with FSM
 - Moved all discussion of Enhancement Projects to Enhancement Chapter
- **Part 3 (former part 2): Counting Practices**
 - Changed wording and sequence of actions consistent with FSM
 - Provided additional guidance and examples for enhancements
 - Added 5th Chapter for counting Conversion Activity
- ▶ **Part 4 (former part 3): Examples**
 - Changed wording and rule boxes consistent with FSM
 - Added clarifications and additional examples
- ▶ **Part 5 (former part 4): Appendices and Glossary**
 - Now contain optional GSCs and VAF



The Next CPM – Part 1 IFPUG FSM

What It Looks Like

▶ IFPUG FSM

- Follows specific document format, structure and wording
- Normative Elements
 - Terms and Definitions
 - Cannot contain requirements
 - Cannot be repeated elsewhere in the document
 - Requirements (aka Rules) must be:
 - Stated precisely
 - Non-redundant
 - Unambiguous
- Informative Elements
 - Examples and Notes
 - Used only to illustrate
 - Cannot contain requirements (i.e., rules)



The Next CPM – Part 1 IFPUG FSM

- ▶ **Part 1 is the ISO Standard “IFPUG FSM” in its entirety**
 - Is the former Part 1 Process and Rules of the CPM
 - Condensed to 21 pages and using strict ISO wording and format



The Next CPM – Part 1 IFPUG FSM What It Looks Like

▶ **Uses ISO Template**

- Different look and feel
- Wording, format and sequence of actions changed
- No repetition

▶ **Minor modifications**

- Clarifications and simplifications
- Added definitions for consistent state, self-contained, sorting and arranging
- Common set of DET and FTR rules for all transactions
- Uniqueness Test (i.e., same DETs, FTRs & Processing Logic) - removed from EI/EO/EQ rules & stated once

▶ **IFPUG FSM does not include GSCs or VAF**



The Next CPM – Part 1 IFPUG FSM

What It Looks Like – Table of Contents

Foreword

Introduction

1 Scope

2 Normative references

3 Terms and definitions

4 Abbreviated terms

5 Measurement Process

5.1 Overview

5.2 Gather the available documentation

5.3 Determine the counting scope and boundary and identify functional user requirements

5.4 Measure data functions

5.5 Measure transactional functions

5.6 Measure conversion functionality

5.7 Measure enhancement functionality

5.8 Calculate functional size

5.9 Document the function point count

5.10 Report the result of the function point count

Annex A



The Next CPM – Part 1 IFPUG FSM

What It Looks Like - Excerpt

5.4 Measure data functions

5.4.1 Overview...

To measure data functions, the following activities shall be performed

- a) identify and group all logical data into data functions in accordance with 5.4.2,
- b) classify each data function as either an ILF or an EIF in accordance with [5.4.3](#),
- c) count the DETs for each data function in accordance with 5.4.4,
- d) count the RETs for each data function in accordance with 5.4.5,
- e) determine the functional complexity for each data function in accordance with 5.4.6,
- f) determine the functional size for each data function in accordance with 5.4.7.



The Next CPM – Part 1 IFPUG FSM What It Looks Like - Excerpt

5.4.3 Classify each data function as either an ILF or an EIF

A data function shall be classified as an

- Internal Logical File (ILF) if it is maintained by the application being measured, or
- External Interface File (EIF) if it is
 - referenced, but not maintained, by the application being measured, and
 - identified as an ILF in one or more other applications.



The Next CPM – Part 1 IFPUG FSM

What It Looks Like - Excerpt

5.5 Measure transactional functions

5.5.1 Overview...

To measure transactional functions, the following activities shall be performed

- a) identify each elementary process required by the user in accordance with [5.5.2](#),
- b) classify each transactional function as an External Input (EI), External Output (EO) or an External Inquiry (EQ) in accordance with 5.5.3,
- c) count the File Types Referenced (FTRs) for each transactional function in accordance with 5.5.4,
- d) count the Data Element Types (DETs) for each transactional function in accordance with 5.5.5,
- e) determine the functional complexity for each transactional function in accordance with 5.5.6, and
- f) determine the functional size of each transactional function in accordance with 5.5.7.



The Next CPM – Part 1 IFPUG FSM What It Looks Like - Excerpt

5.5.2.1. To identify each elementary process, the following activities shall be performed

- a) Compose and/or decompose the functional user requirements into the smallest unit of activity, which satisfies all of the following:
 - 1) is meaningful to the user,
 - 2) constitutes a complete transaction,
NOTE: For users of prior versions of this International Standard, item 2) is not a change, but rather a refinement to increase the specificity to promote consistent interpretation.
 - 3) is self-contained,
 - 4) leaves the business of the application being counted in a consistent state



The Next CPM – Part 1 IFPUG FSM

What It Looks Like - Excerpt

5.5.2.1. To identify each elementary process (cont.)

EXAMPLE 1: A functional requirement may state that a function must be provided to Maintain Employee information. That requirement is decomposed into smaller units of work such as Add Employee, Change Employee, Delete Employee, Inquire about Employee.

EXAMPLE 2: Individual requirements may state the need to add different types of employee information (e.g., address, salary and dependent information), but the smallest unit of activity recognizable to the user is to Add Employee.

- b) Identify an elementary process for each unit of activity identified that meets all of the criteria in 5.5.2.1 a.



The Next CPM – Part 1 IFPUG FSM

What It Looks Like - Excerpt

5.5.2.2. To determine unique elementary processes, the following activities shall be performed

- a) When compared to an Elementary Process (EP) already identified, count two similar EPs as the same Elementary Process if they:
 - 1) require the same set of DETs, and
 - 2) require the same set of FTRs, and
 - 3) require the same set of processing logic to complete the elementary process (refer to the list below 5.5.2.2 b)

NOTE 1: The activities to measure transactional functions are depicted sequentially; however, they are in fact iterative. FTRs and DETs are identified in accordance with 5.5.4 and 5.5.5, but are necessary to compare two similar elementary processes.



The Next CPM – Part 1 IFPUG FSM

What It Looks Like - Excerpt

5.5.2.2. To determine unique elementary processes (cont.)

NOTE 2: One elementary process may include minor variations in DETs or FTRs as well as multiple alternatives, variations or occurrences of processing logic below.

EXAMPLE: When an EP to Add Employee has been identified, it is not divided into two EPs to account for the fact that an employee may or may not have dependents. The EP is still Add Employee, and there is variation in the processing logic and DETs to account for dependents.

b) Do not split an EP with multiple forms of processing logic into multiple EPs

NOTE 3: An elementary process that accepts and validates data from the user, reads and filters data from an ILF and sorts and presents the results back to the user cannot be split into multiple elementary processes.



DET Rules (CPM 4.3)

- a) **Review everything that crosses (enters and/or exits) the boundary,**
- b) **Count one DET for each unique user recognizable, non-repeated attribute that crosses (enters and/or exits) the boundary during the processing of the transactional function,**
- c) **Count only one DET per transactional function for the ability to send an application response message even if there are multiple messages,**
- d) **Count only one DET per transactional function for the ability to initiate action(s) even if there are multiple means to do so, and**
- e) **Do not count the following items as DETs:**
 - literals such as report titles, screen or panel identifiers, column headings and attribute titles;
 - application generated stamps such as date and time attributes;
 - paging variables, page numbers and positioning information; e.g., ‘Rows 37 to 54 of 211’;
 - navigation aids such as the ability to navigate within a list using “previous”, “next”, “first”, “last” and their graphical equivalents;
 - attributes generated within the boundary by a transactional function and saved to an ILF without exiting the boundary;
 - attributes retrieved or referenced from an ILF or EIF for participation in the processing without exiting the boundary.



DET Rules (EI)

CPM 4.2.1

- b) ▶ Count one DET for each user recognizable non-repeated field that enters or exits the application boundary and is required to complete the external input**
- e) ▶ Do not count fields that are retrieved or derived by the system and stored on an ILF during the elementary process if the fields did not cross the boundary**
- c) ▶ Count one DET for the capability to send a system response message outside the application boundary to indicate an error occurred during the processing, confirm that processing is complete or verify that processing should continue**
- d) ▶ Count one DET for the ability to specify an action to be taken even if there are multiple methods for invoking the same logical process**

CPM 4.3

- a) Review everything that crosses (enters and/or exits) the boundary,**
- b) Count one DET for each unique user recognizable, non-repeated attribute that crosses (enters and/or exits) the boundary during the processing of the transactional function,**
- c) Count only one DET per transactional function for the ability to send an application response message even if there are multiple messages,**
- d) Count only one DET per transactional function for the ability to initiate action(s) even if there are multiple means to do so, and**
- e) Do not count the following items as DETs:**



DET Rules (EO/EQ)

CPM 4.2.1

- b)** ▶ Count one DET for each user recognizable non-repeated field that enters the application boundary and is required to specify when, what and/or how the data is to be retrieved or generated by the elementary process
- b)** ▶ Count one DET for each user recognizable, non-repeated field that exits the boundary
- b)** ▶ If a DET both enters and exits the boundary, count it only once for the elementary process
- c)** ▶ Count one DET for the capability to send a system response message outside the application boundary to indicate an error occurred during the processing, confirm that processing is complete or verify that processing should continue
- d)** ▶ Count one DET for the ability to specify an action to be taken even if there are multiple methods for invoking the same logical process
- e)** ▶ Do not count fields that are retrieved or derived by the system and stored on an ILF during the elementary process if the fields did not cross the application boundary
- e)** ▶ Do not count literals as DETs
- e)** ▶ Do not count paging variables or system-generated stamps

CPM 4.3

- a) Review everything that crosses (enters and/or exits) the boundary,
- b) Count one DET for each unique user recognizable, non-repeated attribute that crosses (enters and/or exits) the boundary during the processing of the transactional function,
- c) Count only one DET per transactional function for the ability to send an application response message even if there are multiple messages,
- d) Count only one DET per transactional function for the ability to initiate action(s) even if there are multiple means to do so, and
- e) Do not count the following items as DETs:



The Next CPM - Changes from 4.2.1 to 4.3

Part 2: Process and Rules

- ▶ **Is the former Part 1 (Process and Rules) of the CPM**
- “The Bridge” from the FSM to the familiarity of the CPM
 - Repeats rules from the FSM for convenience
 - Contains additional guidance to easily apply the FSM rules
 - Contains chapters and text not included in the FSM
 - Changed terminology and sequence of actions to comply with the FSM
 - Moved GSCs and VAF to Appendix C



The Next CPM - Changes from 4.2.1 to 4.3

Part 2: Process and Rules

▶ **Chapter 2: Overview of Function Point Analysis**

- Changed to “IFPUG FSM Method Overview”
- Made extensive wording changes for consistency with the FSM
- Added first step in procedural diagram: “Gather Documentation”
- Replaced term “unadjusted function points” with “functional size”
- Moved discussion of “unadjusted” or “adjusted” to Appendix C

▶ **Chapter 3: User View**

- Changed to “Gather Available Documentation”



The Next CPM - Changes from 4.2.1 to 4.3

Part 2: Process and Rules

▶ **Chapter 4: Determine Type of Count**

- Revised wording for definitions:
 - development project function point count,
 - enhancement project function point count and
 - application function point count

▶ **Chapter 5: Identify Counting Scope and Application Boundary**

- Changed to “Determine Counting Scope and Boundary and Identify Functional User Requirements”
- Made minor wording changes



The Next CPM - Changes from 4.2.1 to 4.3

Part 2: Process and Rules

- ▶ **Chapter 6: Count Data Functions**
 - Changed to “Measure Data Functions”
 - Repeated rules here for convenience
- ▶ **Chapter 7: Count Transactional Functions**
 - Changed to “Measure Transactional Functions”
 - Repeated rules here for convenience
- ▶ **Chapter 8: Determine Value Adjustment Factor**
 - Moved entire chapter to Appendix C
- ▶ **Chapter 9: Calculate Adjusted Function Point Count**
 - Moved formulas to Appendix C
 - Moved Enhancement Example to Enhancement Chapter in Part 3



The Next CPM - Changes from 4.2.1 to 4.3

Part 3: Counting Practices

▶ **Chapter 1: Code Data**

- Added ISO definition of Functional Size
- Updated the definition of Functional User Requirements
- Replaced the terms Quality Requirements and Technical Requirements by the ISO term Non-Functional Requirements

▶ **Chapter 2: Logical Files**

- Changed the sequence and structure of steps consistent with the FSM
- Moved evaluation of code data to prevent premature elimination of items that may look like code data, but are in fact not

▶ **Chapter 3: Shared Data**

- Changed references to other parts and chapters of CPM



The Next CPM - Changes from 4.2.1 to 4.3 :

Part 3: Counting Practices

▶ Chapter 4: Enhancement Projects...

- Incorporated section from Part 1, Chapter 9 on Enhancement Project Count
- Updated terminology to be consistent with FSM
- Provided additional guidance and examples for forms of processing logic:
 - #3 - Equivalent values: Changed example from currency conversion in response to IFPUG Bulletin Board discussions
 - #4 - Data is filtered: Changed example to exclude counting change in value or list of values only and added new examples in response to IFPUG Bulletin Board discussions:
 - o Count a change in selection criteria from single item to list of items
 - o Count the addition of a filter as a change, not as a new elementary process



The Next CPM - Changes from 4.2.1 to 4.3 : Part 3: Counting Practices

▶ Chapter 4: Enhancement Projects...(cont.)

- Changed guidance and examples for forms of processing logic (cont.):
 - #11 - Prepare and present information: Added four new examples to reflect CPC responses to Bulletin Board issues:
 - o New application interface with existing file
 - o Changing the position of data elements on a screen, report or file
 - o Changes to characteristics of a data element
 - o Changes to help text
 - #12 - Accept data: Added two new examples to reflect CPC responses to Bulletin Board issues:
 - o Accepting an existing file from a new application
 - o Changes to characteristics of a data element



The Next CPM - Changes from 4.2.1 to 4.3 :

Part 3: Counting Practices

- ▶ **Chapter 4: Enhancement Projects...(cont.)**
 - Changed guidance and examples for forms of processing logic (cont.):
 - #13 - Sorting or arranging a set of data: Added two new examples to reflect CPC responses to Bulletin Board issues
 - In “Consideration and Hints” section:
 - Added discussion regarding deleted functions
 - Moved hints on GSC changes to Appendix C
 - In “Enhancement vs. Maintenance” section prefaced any reference to GSCs with “optional”
- ▶ **Chapter 5 (New): Data Conversion Activity**



The Next CPM - Changes from 4.2.1 to 4.3

Part 4: Examples

▶ Chapter 1: Data Function Counting Examples

- ILF Example: Audit Data for Inquiries and Reports - Removed references to Employee Security Maintenance
- ILF Example: Report Definition - Expanded explanation, addressing why Report Definition is not an instance of code data
- ILF Example: Shared Application Data - Clarified example to make it clear that the security described is not application user access security
- EIF Example: Providing Data to Other Applications - Expanded explanation, addressing why Currency Conversion is not an instance of code data
- EIF Example: Help Application - Expanded explanation, addressing why Help is not an instance of code data; also explained why Window Help and Field Help are separate data functions



The Next CPM - Changes from 4.2.1 to 4.3

Part 4: Examples

▶ Chapter 2: Transactional Function Examples

- EP Example: New Employee/Dependent Data - Expanded explanation, addressing why sending the file to the Benefits System is a separate elementary process
- EP Example: Batch Employee Data Feed - New example illustrates that producing batch error reports and statistical reports are not separate elementary processes
- EP Example: Assign Employee to Job - New example illustrates the evaluation of similar elementary processes to determine whether they are unique
- EI Example: EI with Attributes Retrieved from an EIF - New example illustrates an EI with attributes retrieved from an EIF that do not cross the boundary
- EI Example: EI Delete - New example illustrates counting DETs for a delete transaction



The Next CPM - Changes from 4.2.1 to 4.3

Part 4: Examples

▶ Chapter 2: Transactional Function Examples (cont.)

- EI Example: Add Window Security - New example illustrates counting functionality to maintain application security
- EQ Example: EQ Triggered without Data Entering the Boundary - Renamed example to eliminate confusion
- EQ Example: Additional Help Functionality - New example illustrates counting additional help functionality
- EQ Example: Security for User Access - New example illustrates treatment of application security
- EQ Example: Application Logon - New example illustrates counting the logon function



The Next CPM - Changes from 4.2.1 to 4.3

Part 5: Appendices

▶ **Appendix A: Calculation Tables**

- Removed term “Adjusted”

▶ **Appendix B: Change from Previous Version**

- Reflects all changes in the document from 4.2.1 to 4.3
- Includes the following:
 - An overview of the changes by chapter
 - The impact of the changes on CPM 4.2.1 users
 - Conversion from CPM 4.2.1 to 4.3
 - Recommendations for users changing from CPM 4.2.1 to 4.3



The Next CPM - Changes from 4.2.1 to 4.3

Part 5: Appendices

▶ **Appendix C: Reader Request Form**

- Eliminated Request Form; Readers can submit email to cpc@ifpug.org
- Now contains guidance on applying the General Systems Characteristics and Value Adjustment Factor to determine Adjusted Functional Size
- Contains all formulas

▶ **Glossary**

- Numerous new terms have been added
- Some terms have been changed to align with the FSM



The Next CPM - Results Impact Study

- ▶ **Impact study performed by 44 Volunteers, all CFPS**
- ▶ **Counted case study using both CPM 4.2.1 and CPM 4.3. Results were the same for both methods.**
- ▶ **Counted over 100 projects previously counted under CPM 4.2.1 using CPM 4.3; mix of development, application and enhancement. Results showed no difference.**
- ▶ **Conversion factor was determined to be 1.0 (i.e., no difference).**
- ▶ **Participant quote “The result of the count is the same for both releases. CPM 4.3 includes clarifications about the rules included in the old release.”**
- ▶ **Participant quote “Actually I did not found differences applying 4.2.1 and 4.3 versions of CPM, from my understanding both versions have the same set of rules hence the results are quite the same.”**



The Next CPM - Certification

- ▶ **CPM 4.2.1 => CPM 4.3 No recertification necessary**
- ▶ **CFPS exams will begin using CPM 4.3 a minimum of six months after CPM 4.3 is published in a given language**
- ▶ **Be careful in the exam preparation**
 - Although the Impact Study showed no difference in counts done in CPM 4.2.1 versus CPM 4.3, many definitions and wording have changed, so thorough preparation for the exam is necessary.



The Next CPM - Publication

Activity	Time Frame
Publish CPM 4.3	3rd QTR 2009
CPM 4.3 Becomes Effective	January 2010



The Next CPM - Summary

- ▶ **The IFPUG FSM (CPM 4.3 – Part 1) meets the requirements of the ISO FSM Standard**
- ▶ **It looks and reads significantly different from Part 1 of the CPM 4.2.1**
- ▶ **It conveys the same rules and counting process in a much more succinct and unambiguous manner**
- ▶ **The Implementation Guide (CPM 4.3 – Parts 2 thru 5) provides a bridge between the FSM and the familiarity of the “old” CPM**
- ▶ **Together, they will comprise CPM 4.3**



The Next CPM - Questions

Adri Timp, CFPS
Counting Practices Committee
cpc@ifpug.org

