PMPULS

Consulting ■ Training ■ Mentori

Software Estimation – Function Point Ninja

DAY - 2

A 3 Day Customized Course

Agenda [Day – 2]

- Estimation Bonus
- Function Point Templates

- Function Point Case-Study
- Journey From Estimation To Scheduling

- When Not To Use Function Point
- The WBS Method
- Wide-Band Delphi Method
- COCOMO – II Curtain Raiser
- Some Statistical Data

FP Extensions

FFP
Extension
For
Estimating Rules
For
Estimating
Embedded
&
Real
Time
Systems

Definition - FFP

When compared to traditional IFPUG method, the Full Function Points measurement method:

- **Adopts a FULL functional perspective spanning more than the functionality perceived by the software human users; it includes functionality interacting with other software and engineered devices,**
- **Offers a finer granularity (sub-process level) than the IFPUG method (process level), thus better capturing the size of real-time or embedded software.**
- **Can be applied to processes whose functionality does not pass data outside of the software boundary,**

Definition – Real Time / Embedded Systems

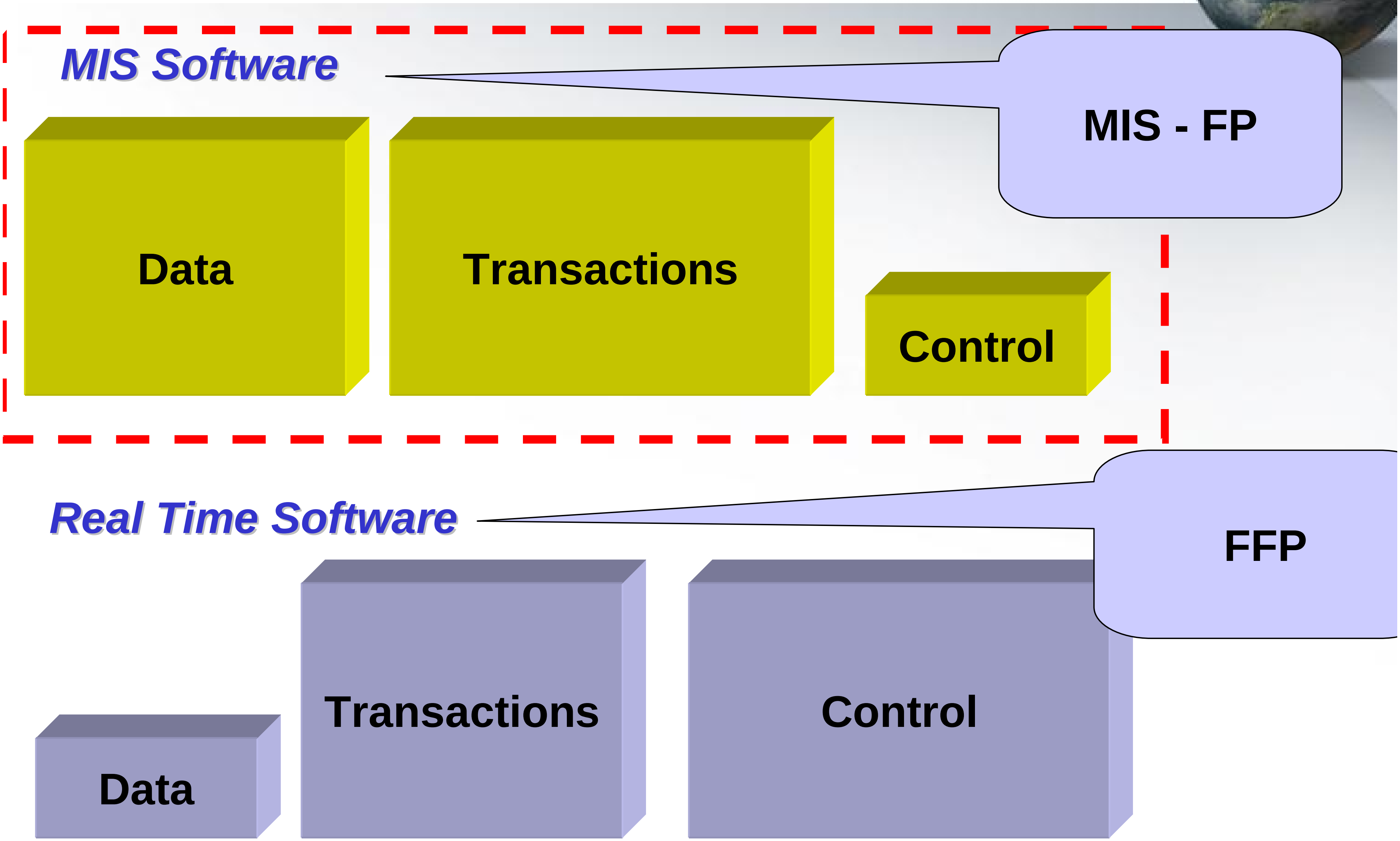
Stankovic and Ramamritham (1988):

‘Real-time systems are defined as those systems in which the correctness of the system depends not only on the result of computation, but also on the time at which the results are produced.’

The Oxford Dictionary of Computing (Illingworth 1991):

‘ Any system in which the time at which the output is produced is significant. This is usually because the input corresponds to some movement in the physical world, and the output has to relate to that same movement. The lag from input time to output time must be sufficiently small for acceptable timeliness. ’

MIS System Vs Real Time System



MIS Software

MIS - FP

Data

Transactions

Control

Real Time Software

FFP

Data

Transactions

Control

MIS System Vs Real Time System

Two important aspects of real-time software:

Timing and ***Interaction with external entities***.

The purpose of this interaction is to obtain information and/or to control external entities.

This interaction takes place with tight constraints on the choice and on the timing of tasks.

Such software incorporates dedicated components to manage these constraints; failure to achieve the timing constraints may result in a malfunction.

Limitations Of FPA

Transaction Limitations

Real-time software processes have a specific transactional characteristic in common: the number of their sub-processes varies a great deal. A real-time functional measurement technique has to take into account that some processes have only a few sub-processes, while others have a large number of sub-processes.

Using FPA rules sub-processes are disregarded and measure only the “Elementary Processes”.

Besides the FPA rules are not flexible to scale to counting different no.s of Sub- Processes.

FFP Extension

Full Function Point is simply an extension of FPA method.

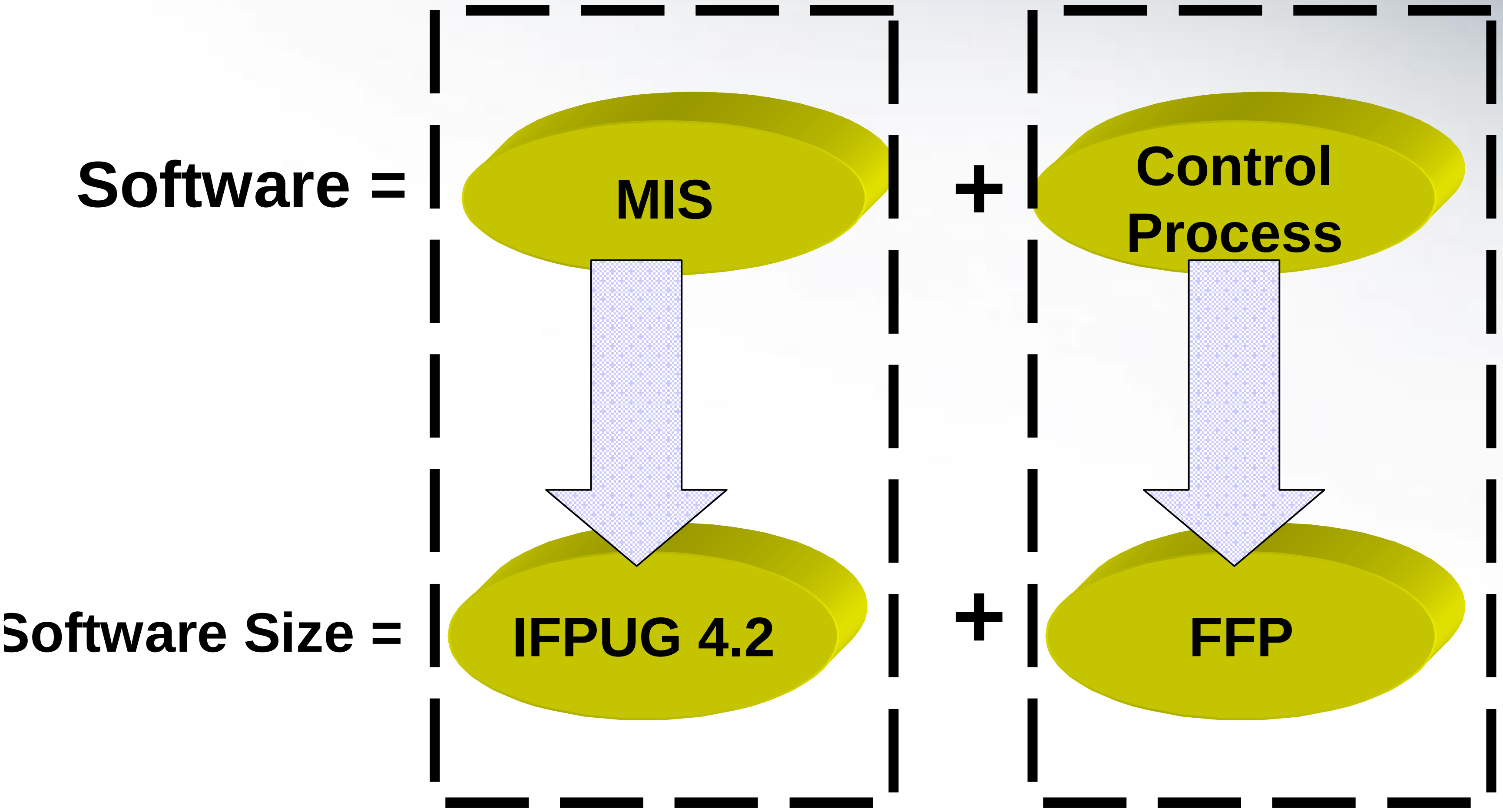
Which means that the counting output of the two could be merged to produce a total.

Only a small part of the FP counting rules (Dealing with control concepts) have been expanded.

The FFP techniques remarkably resemble the FPA method in look and feel as well as in terminology.

FFP Extension

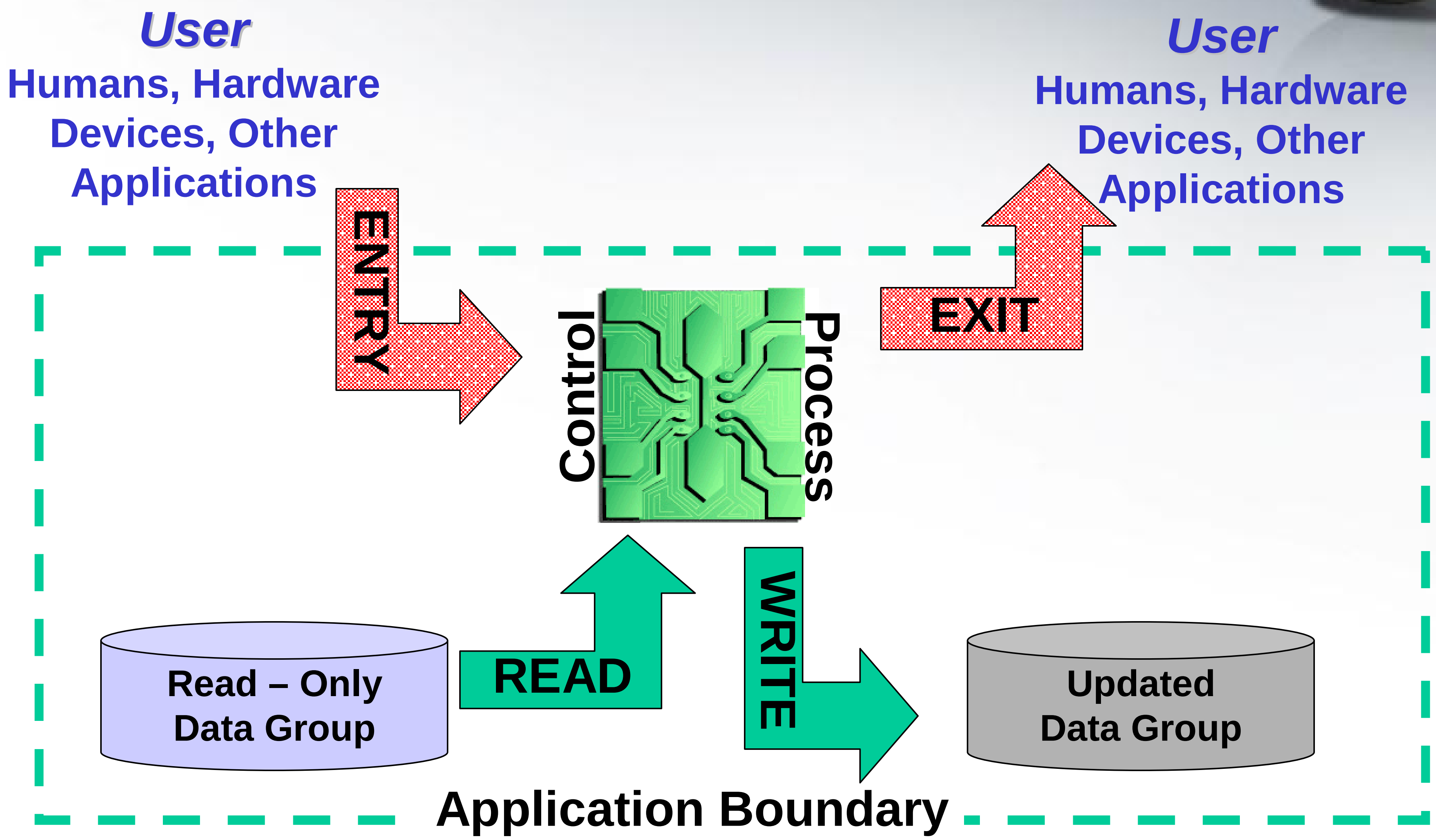
Model For MIS And Real Time Software



Software = MIS + Control Process

Software Size = IFPUG 4.2 + FFP

FFP Core Concept



User

Humans, Hardware Devices, Other Applications

ENTRY

Control

Process

EXIT

User

Humans, Hardware Devices, Other Applications

READ

WRITE

Read – Only Data Group

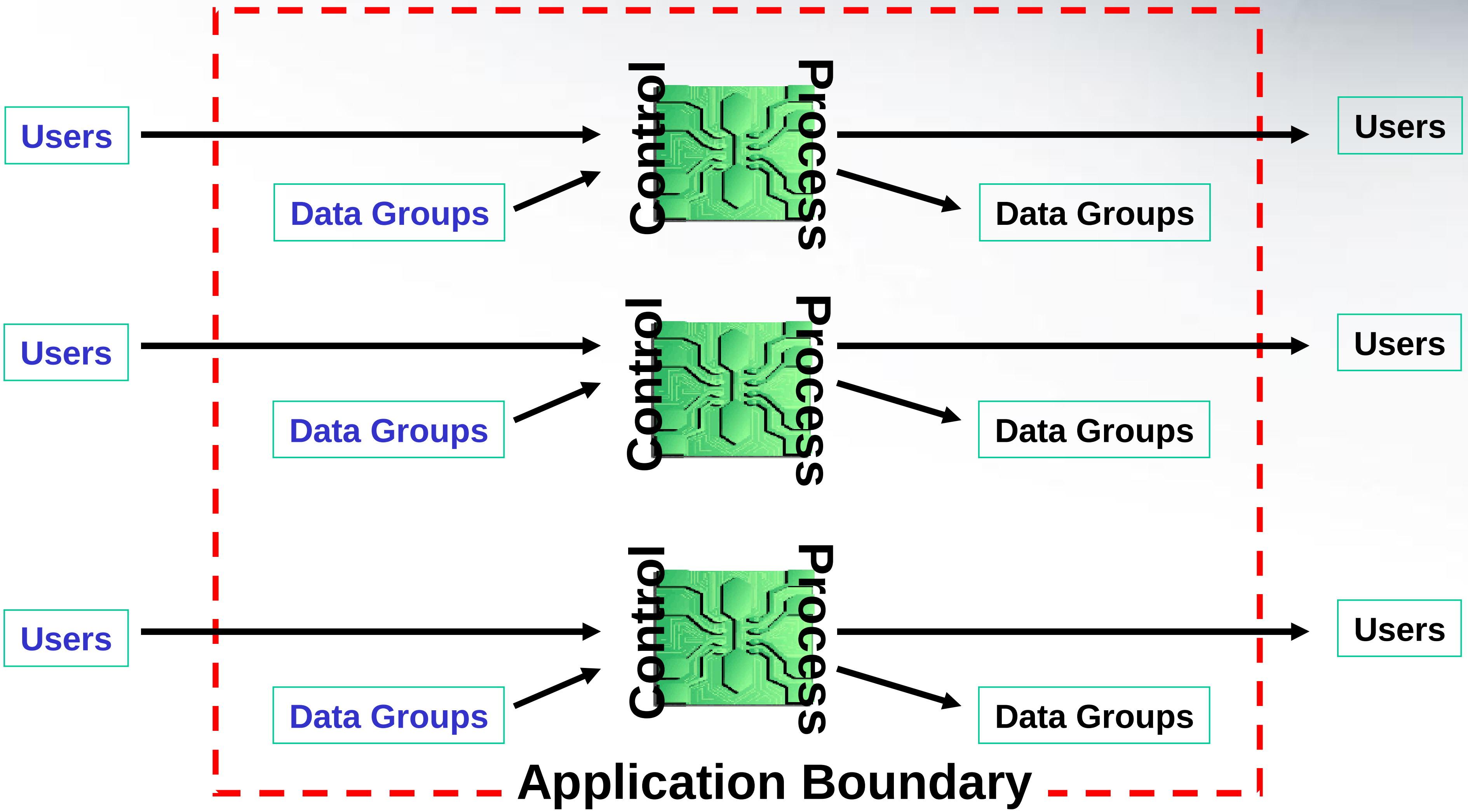
Updated Data Group

Application Boundary

FFP Core Concept

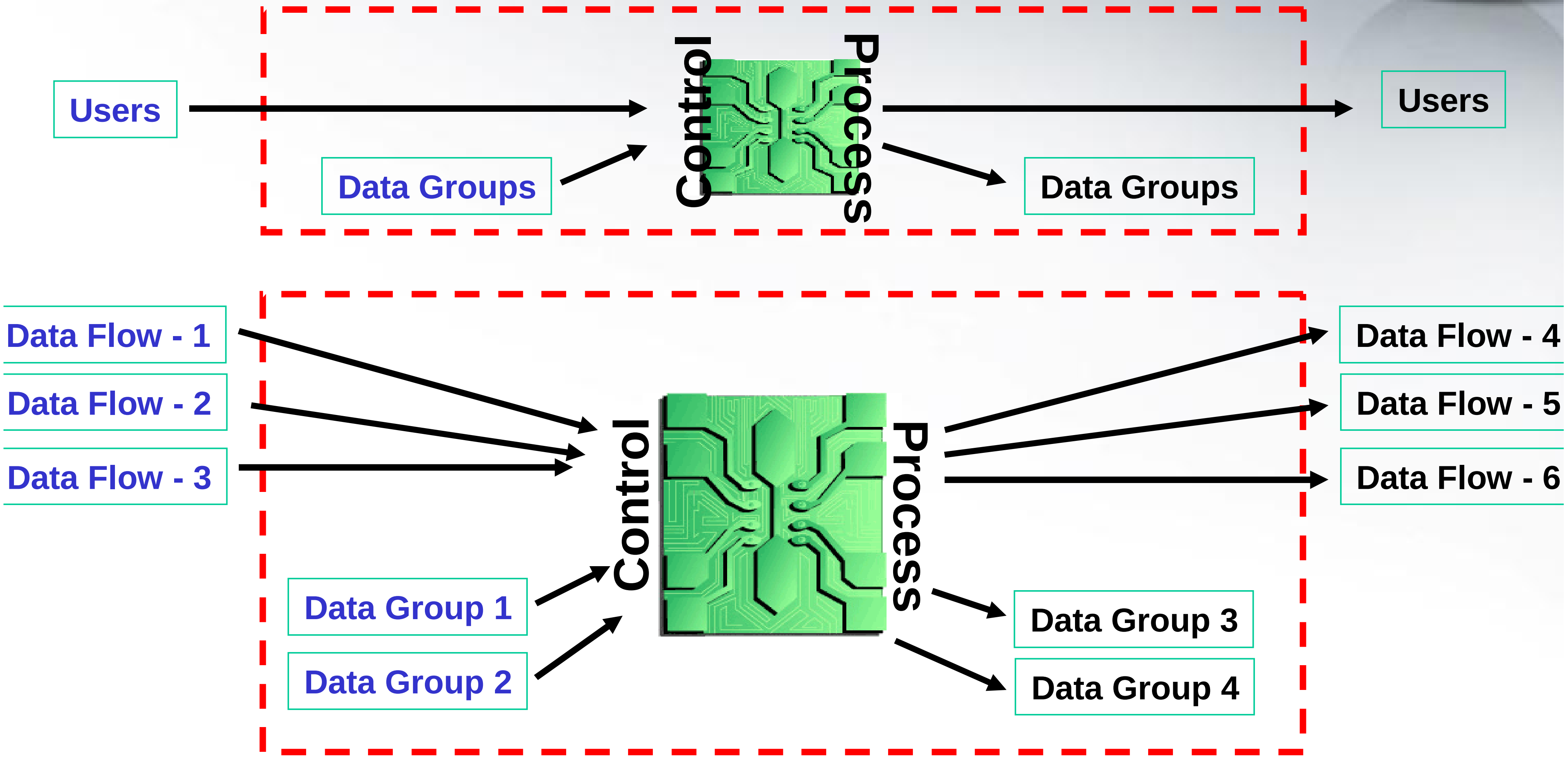
Process Level

All processing associated with a unique trigger

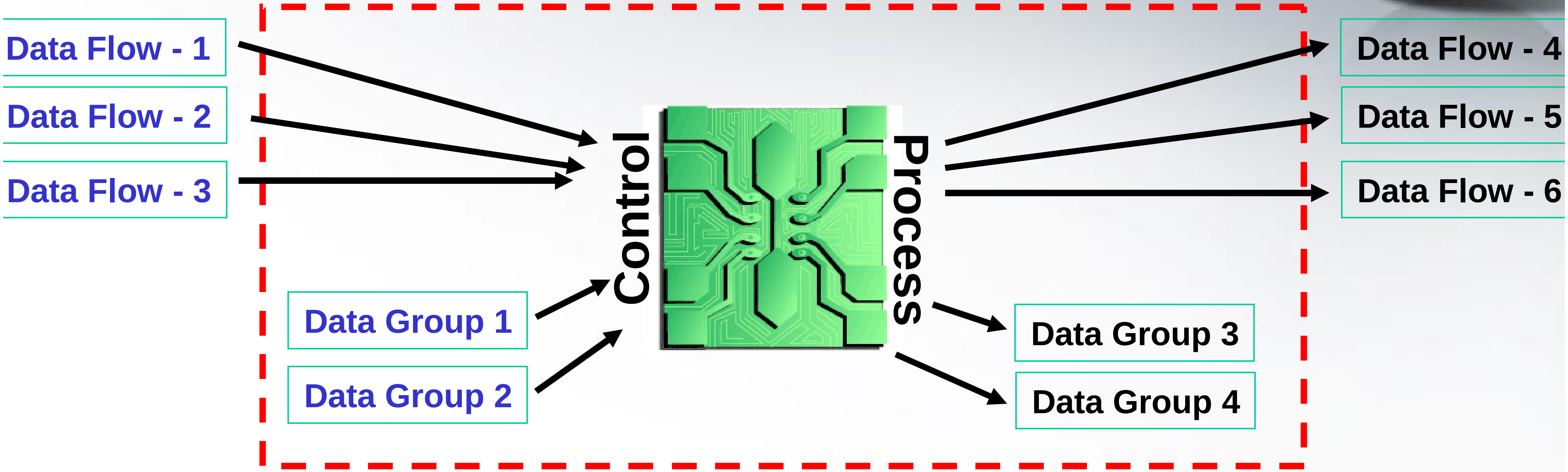


FFP Core Concept

Sub - Process Level



FFP Core Concept



Data Flow - 1

Data Flow - 2

Data Flow - 3

Control

Process

Data Flow - 4

Data Flow - 5

Data Flow - 6

Data Group 1

Data Group 2

Data Group 3

Data Group 4

Counting is done at the sub-process level.

Or

The points are assigned at the Sub-process level

FFP Function Types

Consequently, FFP introduces additional data and transactional function types: -

Data Function Types (2) :-

1. Updated Control Group (UCG)
2. Read-Only Control Group (RCG)

Control Transactional Function Types (4): -

1. External Control Entry (ECE)
2. External Control Exit (ECX)
3. Internal Control Read (ICR)
4. Internal Control Write (ICW)

FFP Function Types

Data Function Types

Updated Control Group (UCG):

A UCG is a group of control data updated by the application being counted. It is identified from a functional perspective.

Read-only Control Group (RCG):

An RCG is a group of control data used, but not updated, by the application being counted. It is identified from a functional perspective.

FFP Function Types

Control Transactional Function Types

External Control Entry (ECE):

An ECE is a unique sub-process. It is identified from a functional perspective. An ECE is processes control data coming from outside the application ’ s boundary. It is the lowest level of decomposition of a process acting on one group of data.

Consequently, if a process enters two groups of data, there are at least 2 ECEs.

ECEs exclude the updating of data, a functionality that is covered by another Control Function Type (Internal Control Write).

FFP Function Types

Control Transactional Function Types

External Control Exit (ECX):

An ECX is a unique sub-process. It is identified from a functional perspective. An ECX is processes control data going outside the application boundary. It is the lowest level of decomposition of a process acting on one group of data.

Consequently, if a process exits two groups of data, there are at least 2 ECXs.

ECXs exclude the reading of data, a functionality that is covered by another Control Function Type (Internal Control Read).

FFP Function Types

Control Transactional Function Types

Internal Control Read (ICR):

An ICR is a unique sub-process. It is identified from a functional perspective. **An ICR reads control data**. It is the lowest level of decomposition of a process acting on one group of data.

Consequently, if a process reads two groups of data, there are at least 2 ICRs.

FFP Function Types

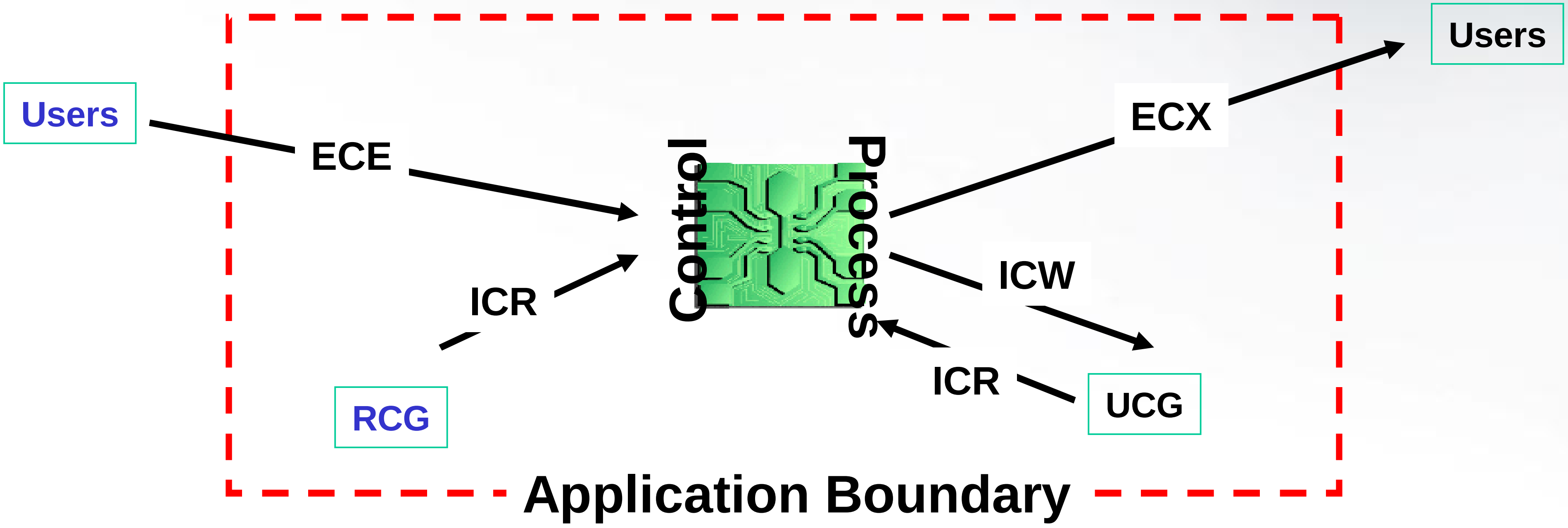
Control Transactional Function Types

Internal Control Write (ICW):

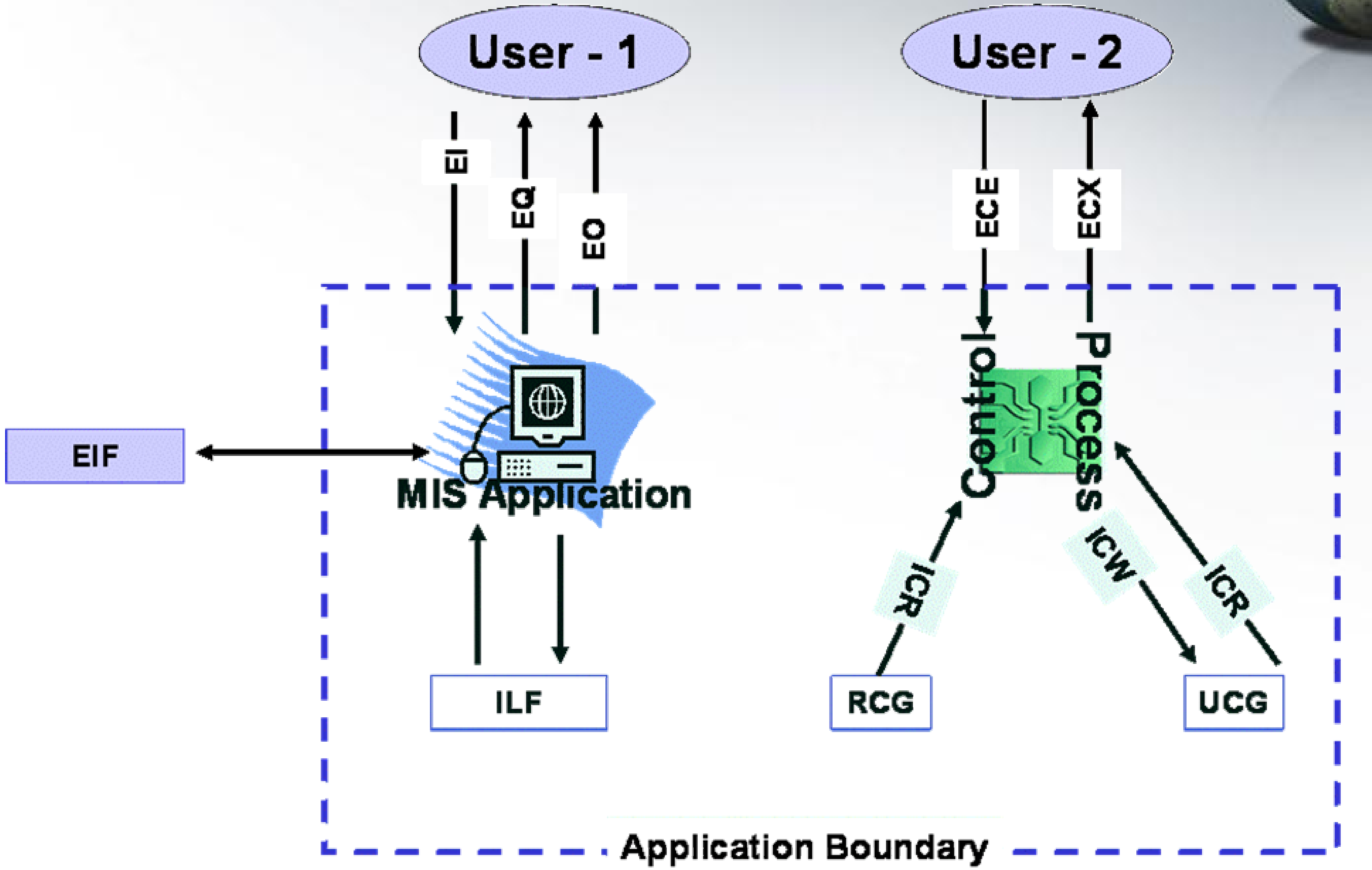
An ICW is a unique sub-process. It is identified from a functional perspective. An ICW writes control data. It is the lowest level of decomposition of a process acting on one group of data.

Consequently, if a process writes on two groups of data, there are at least 2 ICWs.

FFP Function Types



All Together – The FFP



FFP Counting Method – Point Assignments

To Data Function Types

Multi-Occurrence Groups Of Data

The rules on this are the same as the calculation of points for ILF and EIF as in FPA

There is just no difference.

All data function types are segregated into two portions – Multi-Occurrence and Single Occurrence types.

Than FPA is used for Multi-Occurrence types and FFP for Single Occurrence types

FFP Counting Method – Point Assignments

To Data Function Types

Single Occurrence Groups Of Data

Here only the DETs are calculated for each of the Data Function Type.

Then segregated into UCG and RCG.

For each UCG the formula is **[No. of DETs / 5] + 5**

For each RCG the formula is **[No. of DETs / 5]**

FFP Counting Method – Point Assignments

To Data Function Types

Single Occurrence Groups Of Data

A single occurrence **UCG** comprises all single control updated values (from a functional perspective) of the application being measured.

Since it contains all single values of the application, there can be only one of them in an application.

Consequently, an application can have more than one multiple occurrence UCG, but only one single occurrence UCG.

The same goes for **RCGs**.

FFP Counting Method – Point Assignments

To Data Function Types

Single Occurrence Groups Of Data

In typical real-time applications, the number of such single values varies from a few up to hundreds. That is why a formula is used rather than a 3-level table like the standard FPA technique. It allows FFP to consider a large range of single occurrence groups of data. The following tables present examples of the formulas:

UCG Point Assignement Example			
No. Of DETs :	10 DETs	35 DETs	50 DETs
Corresponding Points :	7	12	15

RCG Point Assignement Example			
No. Of DETs :	10 DETs	35 DETs	50 DETs
Corresponding Points :	2	7	10

FFP Counting Method – Point Assignments

To Transaction Function Types

For the transactions part there is no difference between the method of calculation among the four transaction types.

- The number of points assigned to Control Transactional Functions Types (ECE, ICW and ICR) depends on the number of DETs 9 .
- Once the number of DETs determined, the following table is used to translate DETs into points

FFP Counting Method – Point Assignments

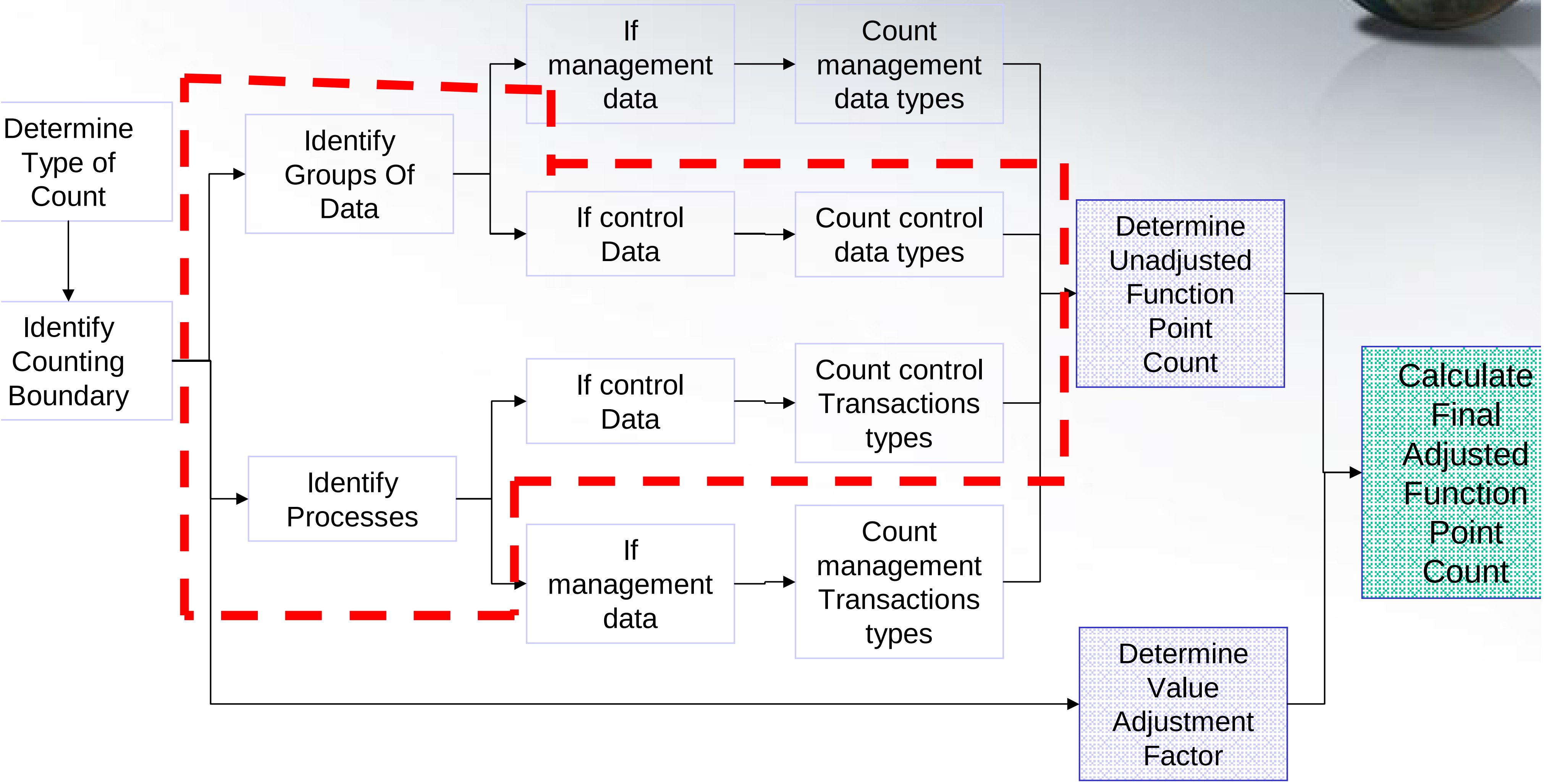
To Transaction Function Types

Control Transactional Function Type Translation Table

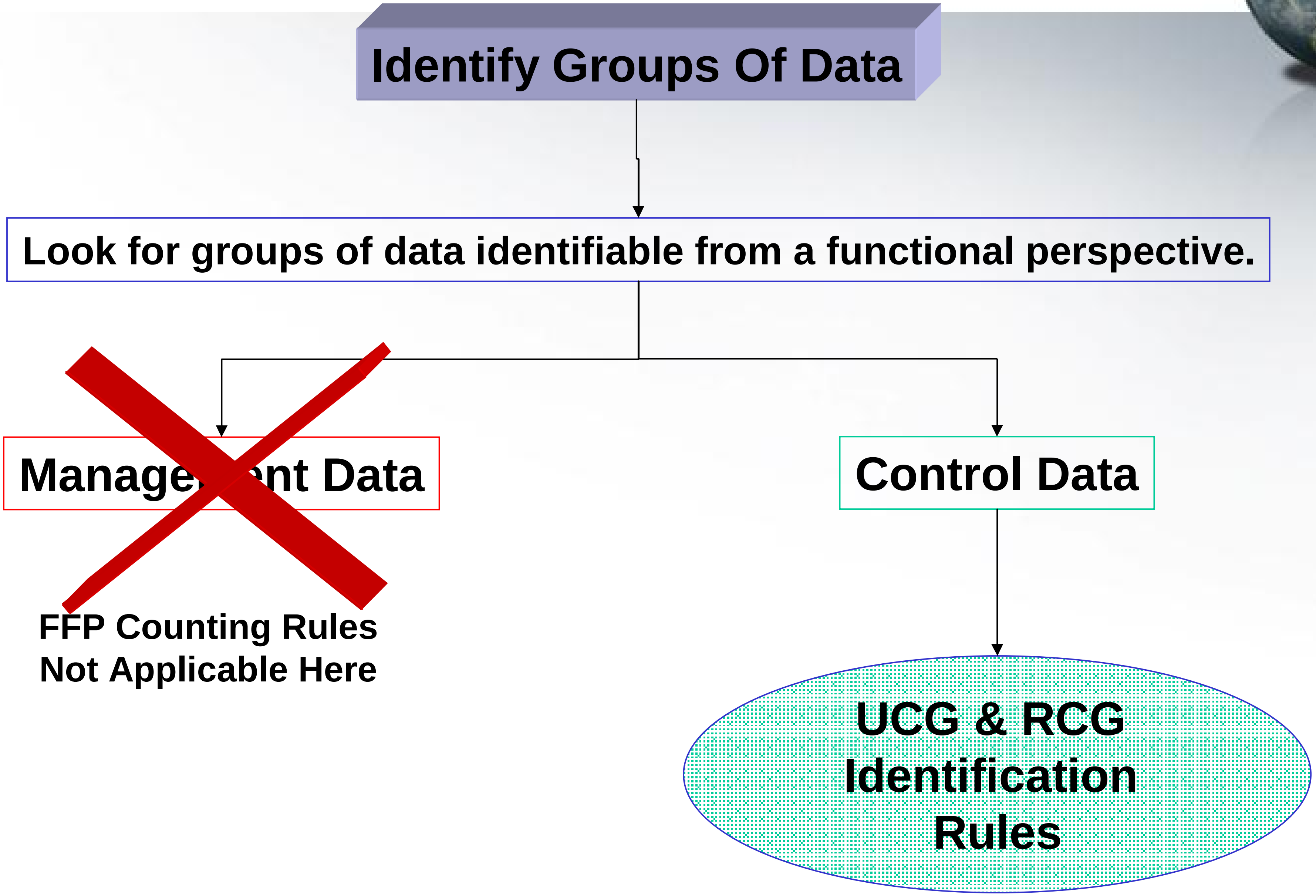
No. Of DETs :	1 to 19 DETs	20 to 50 DETs	51 + DETs
Corresponding Points :	1	2	3

These range boundaries (1 to 19, 20 to 50, 51+) were chosen in order to bring the size of Control Transactional Function Types in as close alignment as possible with FPA.

FFP Counting Procedures And Rules


Determine Type of Count
Identify Counting Boundary
Identify Groups Of Data
Identify Processes
If management data
Count management data types
If control Data
Count control data types
If control Data
Count control Transactions types
If management data
Count management Transactions types
Determine Unadjusted Function Point Count
Determine Value Adjustment Factor
Calculate Final Adjusted Function Point Count

FFP Counting Procedures And Rules



Identify Groups Of Data

Look for groups of data identifiable from a functional perspective.

~~Management Data~~

FFP Counting Rules
Not Applicable Here

Control Data

UCG & RCG
Identification
Rules

FFP Counting Procedures And Rules

UCG & RCG Identification Rules

UCG Identification: -

q The group is either a logically related group of data or a single occurrence group of data.

q The group of data is updated within the application boundary.

q The group of data lives for more than one transaction.

q The group of data identified has not been counted as an RCG, ILF or EIF for the application.

All of the above are mandatory rules.

FFP Counting Procedures And Rules

UCG & RCG Identification Rules

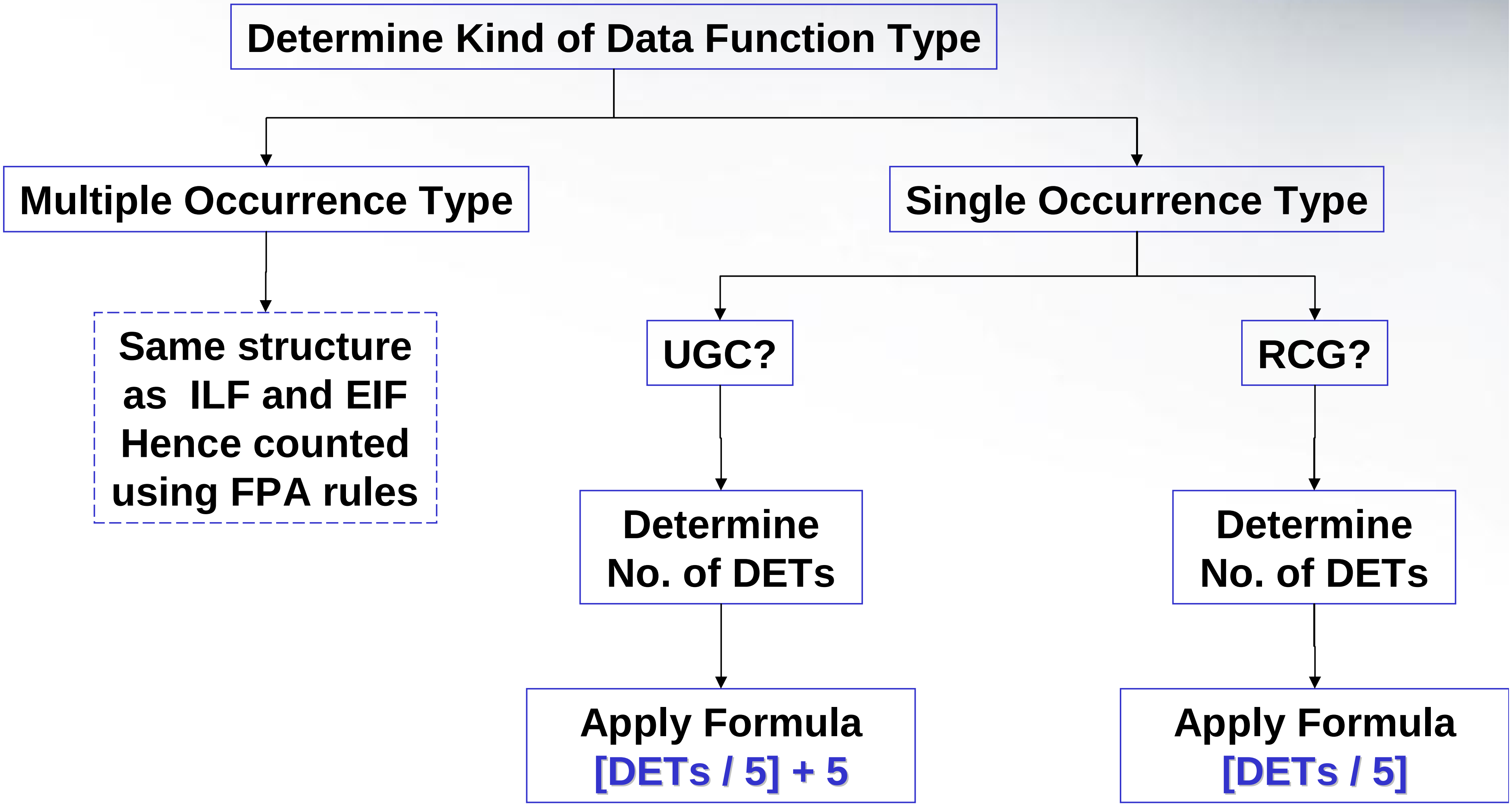
RCG Identification: -

- The group is either a logically related group of data or a single occurrence group of data.
- The group of data is not updated by the application being counted.
- The group of data is referenced by the application being counted.
- The group of data lives for more than one transaction.
- The group of data has not been counted as a UCG, ILF or EIF for the application.

All of the above are mandatory rules.

FFP Counting Procedures And Rules

Point Assignments To Data Function Types



Determine Kind of Data Function Type

Multiple Occurrence Type

Same structure as ILF and EIF Hence counted using FPA rules

Single Occurrence Type

UGC?

Determine No. of DETs

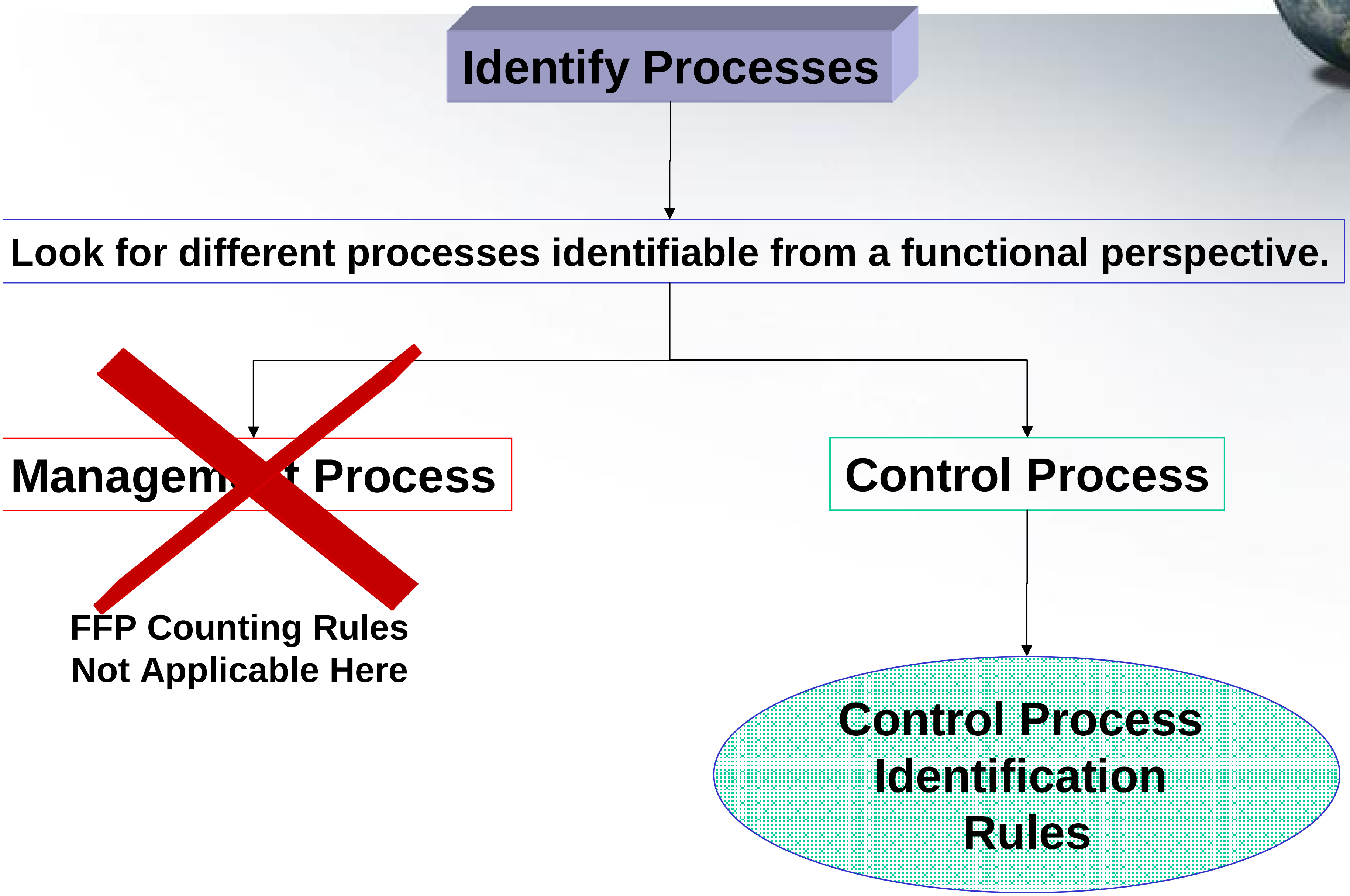
Apply Formula [DETs / 5] + 5

RCG?

Determine No. of DETs

Apply Formula [DETs / 5]

FFP Counting Procedures And Rules



Identify Processes

Look for different processes identifiable from a functional perspective.

Management Process

FFP Counting Rules
Not Applicable Here

Control Process

Control Process
Identification
Rules

FFP Counting Procedures And Rules

Processes Identification Rules

- According to the logical execution order of the sub-processes within the process, identify the first sub-process that receives, exits, reads or writes a group of control data.
- Apply the relevant ECE, ECX, ICR or ICW set of rules.
- Determine the ECE, ECX, ICR or ICW contribution (point assignment) to the unadjusted function point count.
- Again according to execution order, identify the next sub-process that enters, exits, reads or writes a group of control data.
- Repeat steps 2 to 4 until all sub-processes of the processes are identified.
- At the end of the cycle, remove all the duplicated sub-processes (same processing and same DET).

Note:* *If the same sub-process is associated with different control processes, it can be counted more than once.

FFP Counting Procedures And Rules

ECE Identification Rules

- The sub-process receives a group of control data from outside the application boundary.
- The sub-process receives only one group of data. If more than one different group of data is received, count one ECE for each group of data.
- The sub-process does not exit, read or write data.
- The sub-process is unique, that is, the processing and data elements identified are different from other ECEs associated with the same process.

All are mandatory rules

FFP Counting Procedures And Rules

ECX Identification Rules

- The sub-process sends control data external to the application ’ s boundary.
- The sub-process sends only one group of data. If more than one different group of data is sent outside the application ’ s boundary, count one ECX for each group of data.
- The sub-process does not receive, read or write data.
- The sub-process is unique, that is, the processing and data elements identified are different from other ECXs associated with the same process.

All are mandatory rules

FFP Counting Procedures And Rules

ICR Identification Rules

- The sub-process reads a group of control data.
- The sub-process reads only one group of data. If more than one different group of data is read, count one ICR for each group of data.
- The sub-process does not receive, exit or write data.
- The sub-process is unique, that is, the processing and data elements identified are different from other ICRs associated with the same process.

All are mandatory rules

FFP Counting Procedures And Rules

ICW Identification Rules

- The sub-process writes a group of control data.
- The sub-process writes only one group of data. If more than one different group of data is written, count one ICW for each group of data.
- The sub-process does not receive, exit or read data.
- The sub-process is unique, that is, the processing and data elements identified are different from other ICWs associated with the same process.

All are mandatory rules

FFP Counting Procedures And Rules

Processes Point Assignments

For ECE and ECX

- Count one DET for each unique user recognizable, non-recursive field, that crosses the boundary of the application.

For an ICR

- Count one DET for each unique user recognizable, non-recursive field that is read from an ILF, EIF, UCG or RCG, including keys.

For an ICW :

- Count one DET for each unique user recognizable, non-recursive field that is written in an ILF or UCG, including keys.

FFP Counting Procedures And Rules

Processes Point Assignments

Having counted the DETs use the following table for the final point assignments: -

Control Transactional Function Type Translation Table			
No. Of DETs :	1 to 19 DETs	20 to 50 DETs	51 + DETs
Corresponding Points :	1	2	3

When NOT To Use FP

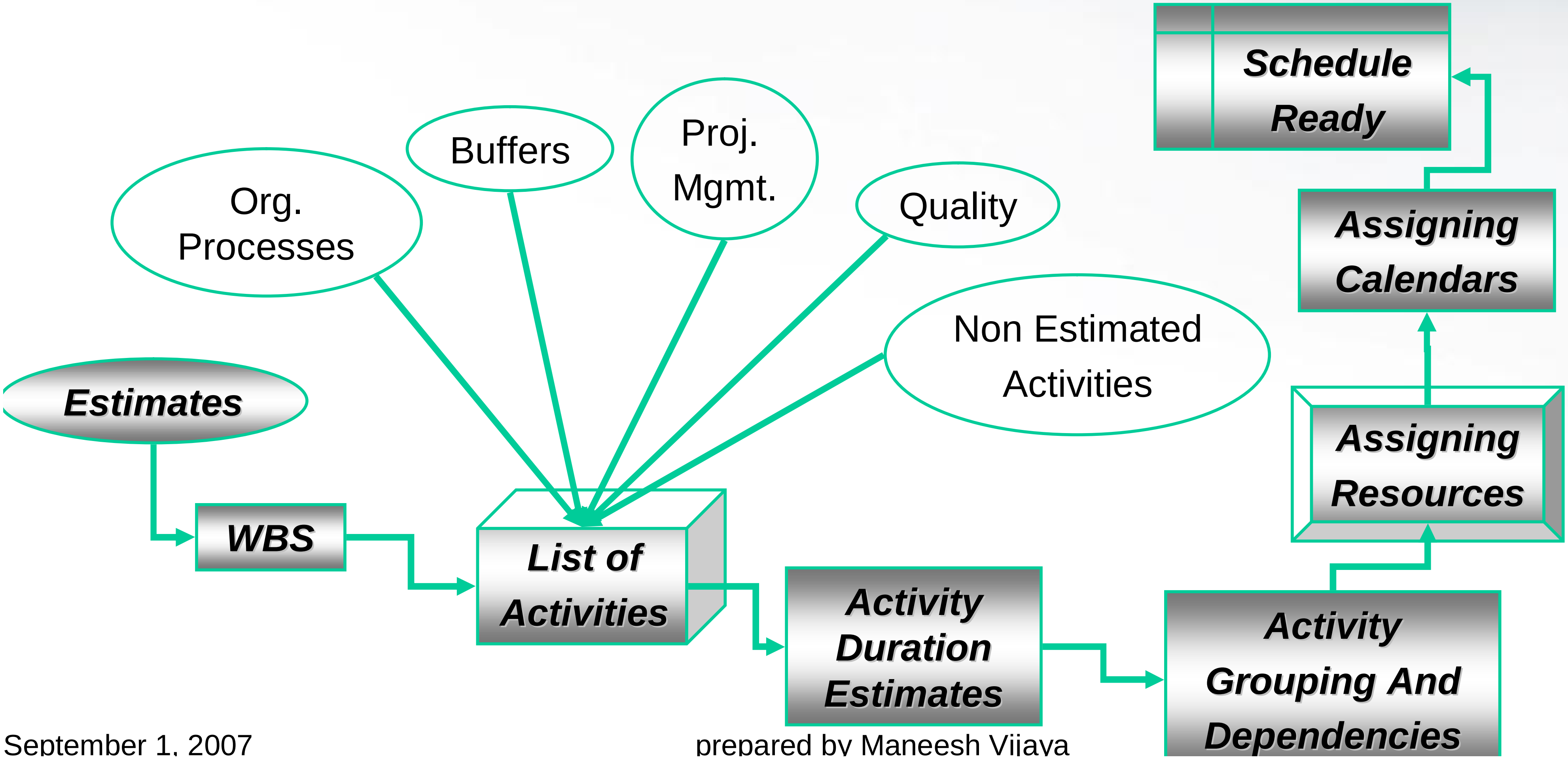
Never Use ….When

- Maintenance Problems
- Performance Issues
- Production Problems

From Estimations To Schedule

Let us first discuss "How this journey is done as of today?"

Now let us discuss "How this Journey should be?"



Wide-Band Delphi Method

What Is It?

- Its a Process
- Not a Method
- Can be used with any Method of estimation
- Can be used with various Methods of estimations done together
- Needs a very high level of professionalism
- Needs more than one estimator

Wide-Band Delphi

This Process of estimation is applicable to every situation of estimation and is popularly known as the “Middle-way” method.

It has since been adapted across many industries to estimate many kinds of tasks, ranging from statistical data collection results to sales and marketing forecasts.

It has proven to be a very effective estimation tool, and it lends itself well to software projects.

Wide-Band Delphi Overview

- The technique can help you estimate, plan and schedule – almost anything
- It’s comprised of 6 steps
- Particularly useful for high – mid level “initial” estimates – at the beginning or early phases of a project
- A basis for the estimating is gathering a set of “experts” in specific areas - then contrasting their estimates
- Estimates are usually confidential
- Team based estimate convergence and off-line schedule - materials work by the PM or facilitator
- Usually only examine a small section or component of the overall effort / project (50-100 tasks)
- Iterative, team based, collaborative estimating

Step – 1: Choose The Team

Picking a qualified team is an important part of generating accurate estimates. Each team member must be willing to make an effort to estimate each task ***honestly***, and should be comfortable working with the rest of the team.

- Steps In: Choose the team
  - The project manager selects the estimation team and a moderator. The team should consist of 3 to 7 project team members.
    - The moderator should be familiar with the Delphi process, but should not have a stake in the outcome of the session if possible.
    - If possible, the project manager should not be the moderator because he should ideally be part of the estimation team.

Step – 2: Kickoff Meeting

This discussion typically uncovers many important (but previously unrecognized) project priorities, assumptions, and tasks.
The team is much more familiar with the work they are about to undertake.
The team members need to be on the same page.

- Steps In : Kickoff Meeting
  - The project manager must make sure that each team member understands the Delphi process, has read the vision and scope document and any other documentation, and is familiar with the project background and needs.
  - The team brainstorms and writes down assumptions.
  - The team generates a WBS with 10-20 tasks.
  - The team agrees on a unit of estimation.

Step – 3: Individual Preparation

This Step is very important as from this point onwards there is no looking back.
This entails – Assumptions List, Formats and templates of estimation, Personal Arguments and Comments, Understanding of the goals and objectives of the project, technical journals and project documentation – with each of the Estimators.

q Steps In: Individual Preparation

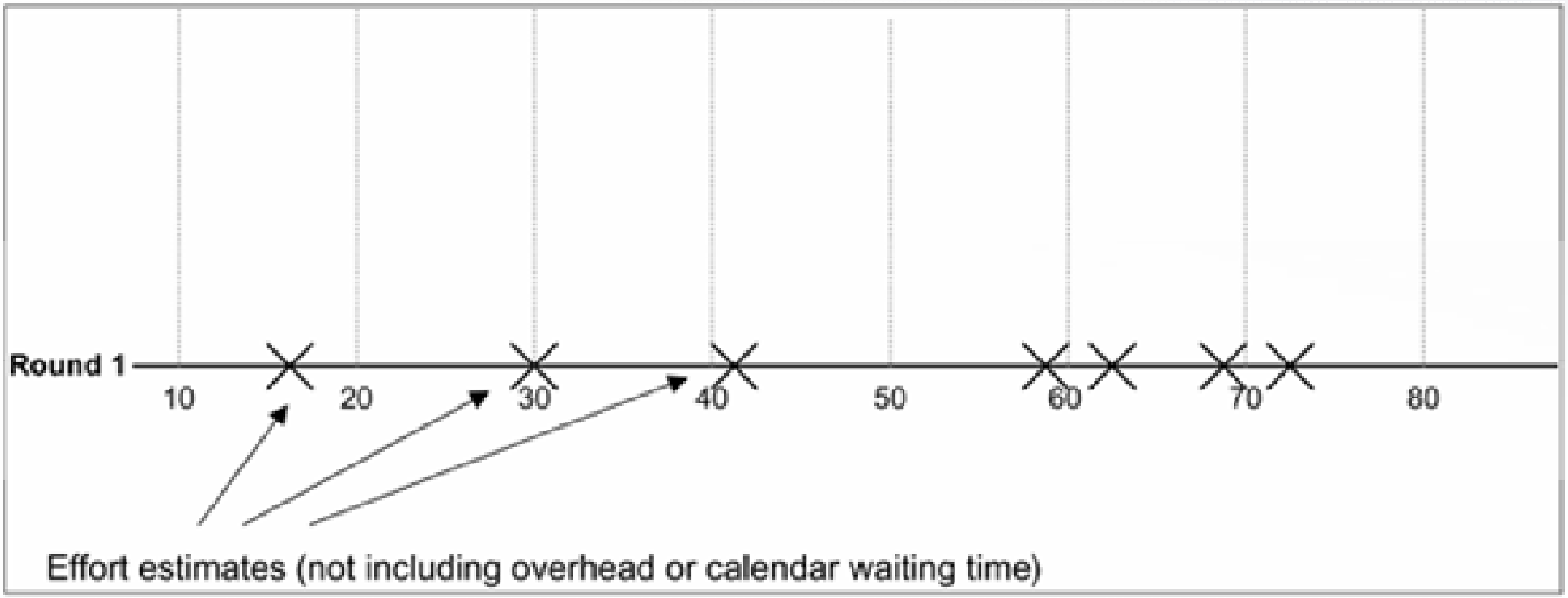
- Ø Each team member independently generates a set of preparation results.
- Ø For each task, the team member writes down an estimate for the effort required to complete the task, and any additional assumptions he needed to make in order to generate the estimate.

Step – 4: Estimation Session

- Steps in: Estimation Session
  - During the estimation session, the team comes to a consensus on the effort required for each task in the WBS.
  - Each team member fills out an estimation form which contains his estimates.
  - The rest of the estimation session is divided into rounds during which each estimation team member revises her estimates based on a group discussion. Individual numbers are not discussed

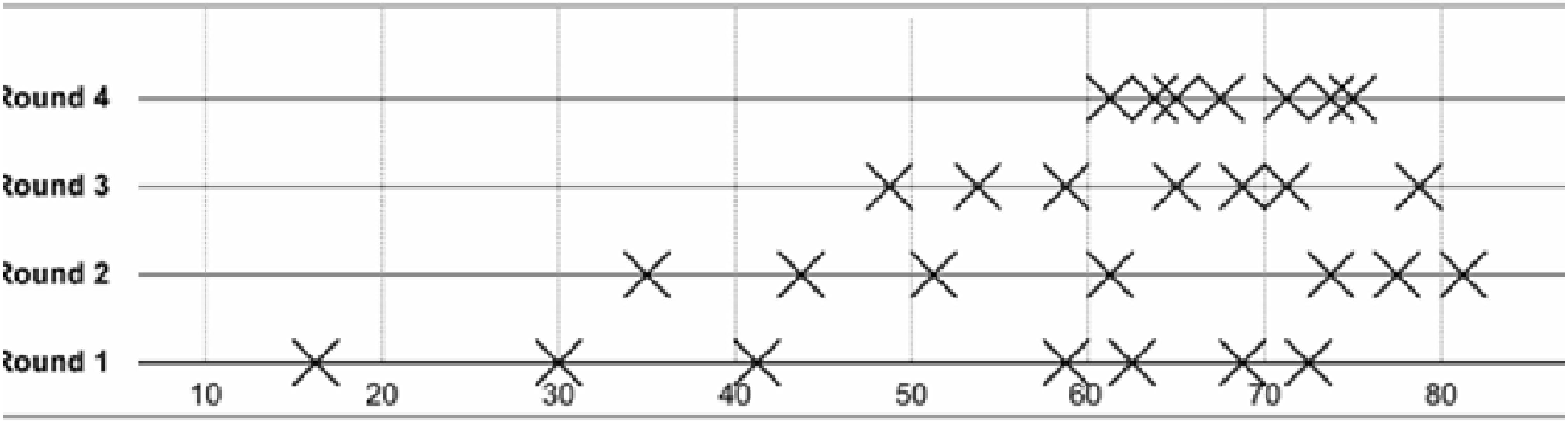
Step – 4: Estimation Session

- Step in: Estimation Session (continued)
  - The moderator collects the estimation forms and plots the sum of the effort from each form on a line:



Step – 4: Estimation Session

- Steps In: Estimation Session (continued)
  - The team resolves any issues or disagreements that are brought up.
    - Individual estimate times are not discussed. These disagreements are usually about the tasks themselves. Disagreements are often resolved by adding assumptions.
  - The estimators all revise their individual estimates. The moderator updates the plot with the new total:



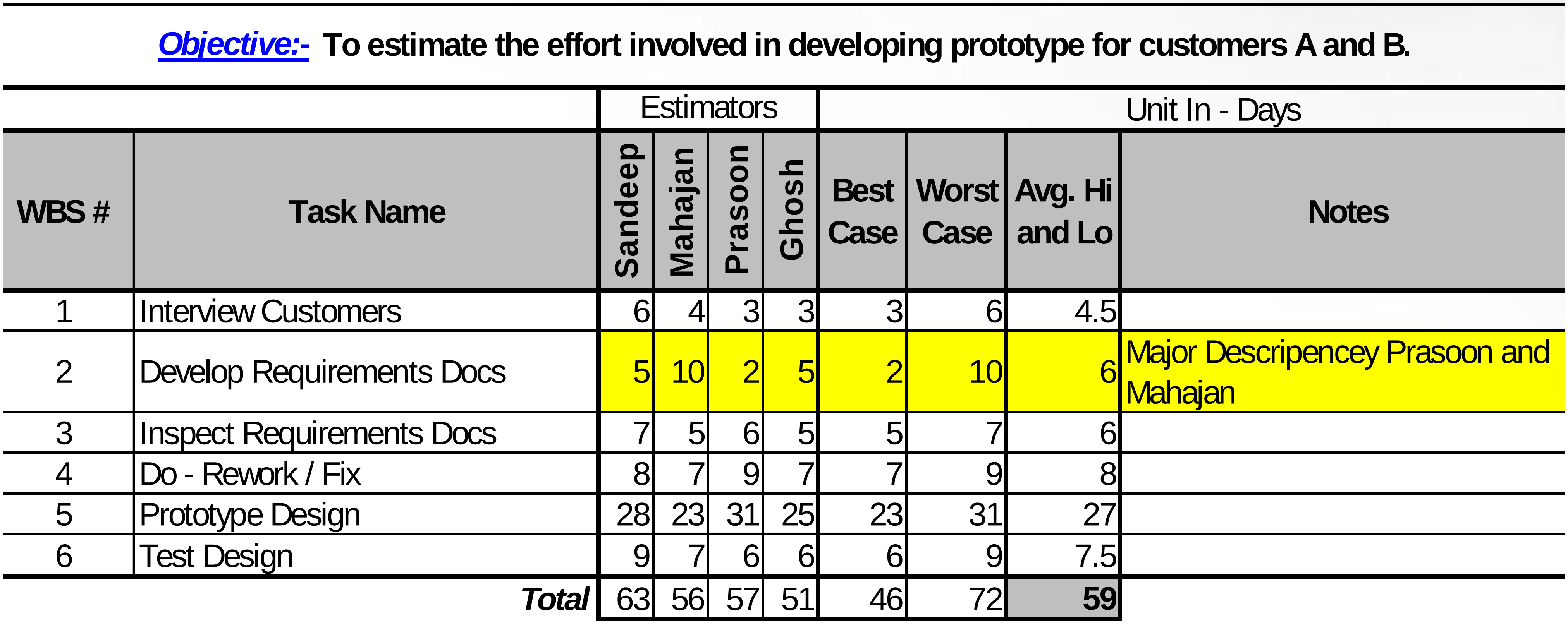
Step – 4: Estimation Session

q Steps In: Estimation Session (continued):

  Ø The moderator leads the team through several rounds of estimates to gain consensus on the estimates. The estimation session continues until the estimates converge or the team is unwilling to revise estimates.

Step – 5: Assemble Tasks

- Step 5: Assemble Tasks
  - The project manager works with the team to collect the estimates from the team members at the end of the meeting and compiles the final task list, estimates and assumptions.



Objective:- **To estimate the effort involved in developing prototype for customers A and B.**

		Estimators				Unit In - Days			
WBS #	**Task Name**	**Sandeep**	**Mahajan**	**Prasoon**	**Ghosh**	**Best Case**	**Worst Case**	**Avg. Hi and Lo**	**Notes**
1	Interview Customers	6	4	3	3	3	6	4.5	
2	Develop Requirements Docs	5	10	2	5	2	10	6	Major Descripencey Prasoon and Mahajan
3	Inspect Requirements Docs	7	5	6	5	5	7	6	
4	Do - Rework / Fix	8	7	9	7	7	9	8	
5	Prototype Design	28	23	31	25	23	31	27	
6	Test Design	9	7	6	6	6	9	7.5	
	Total	63	56	57	51	46	72	**59**	

Step – 6: Review Results

- Step 6: Review Results
  - The project manager reviews the final task list with the estimation team.

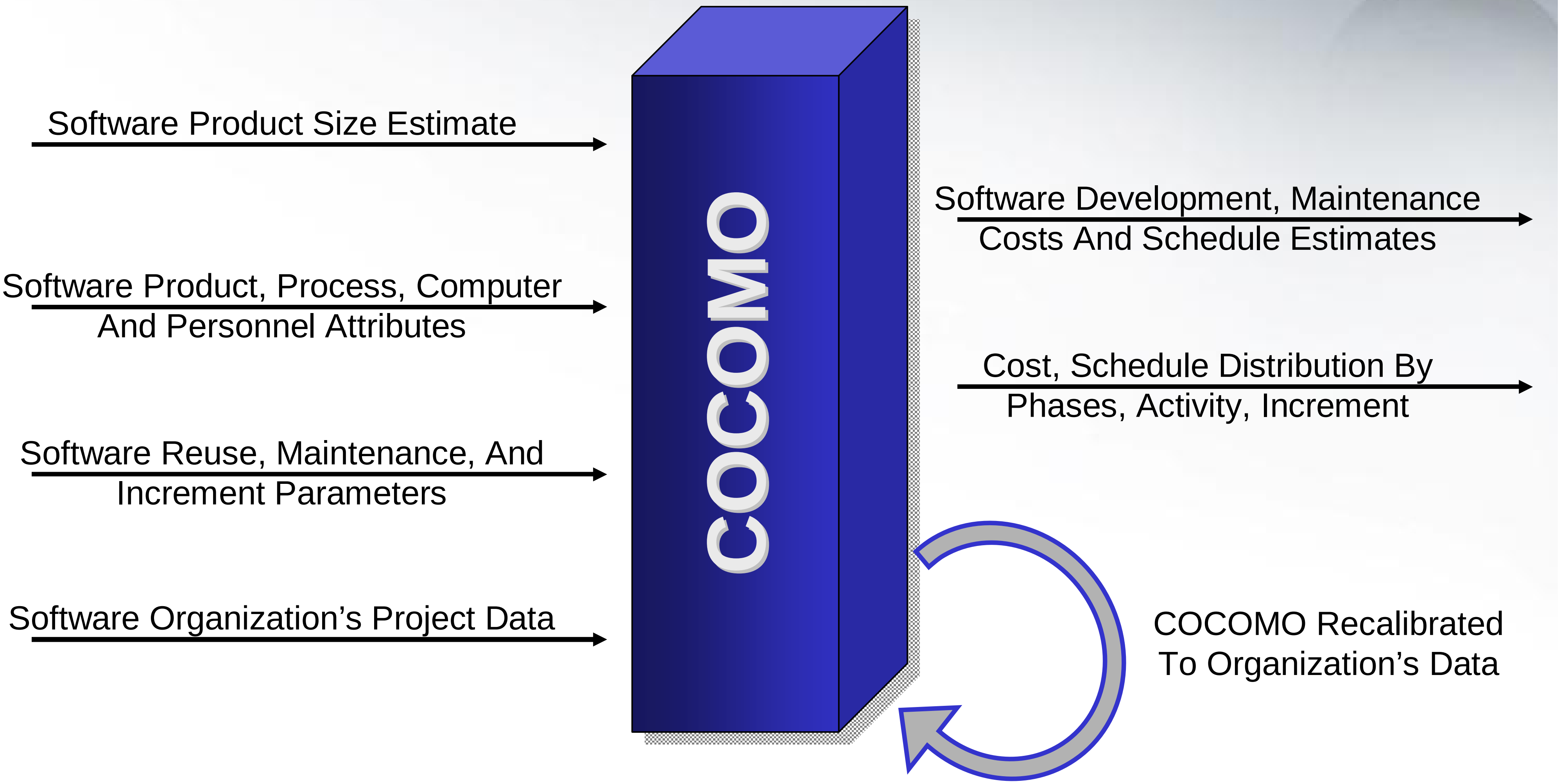
Strengths and Weaknesses

- Strengths:
  - Team based planning - collaboration and buy-in
  - Thorough generation of small task lists
  - Produces estimate data that can be used in a variety of ways
  - Creates useful project artifacts (risks, issues, assumptions, etc.)
- Weaknesses:
  - Lots of overhead (time, team involvement, planning) for a relatively small sets of tasks
  - Takes quite a few “steps” or iterations
  - How do you update the plan?

COCOMO II

A Curtain Raiser

COCOMO II [*Baseline Overview*]



Software Product Size Estimate

Software Product, Process, Computer And Personnel Attributes

Software Reuse, Maintenance, And Increment Parameters

Software Organization’s Project Data

COCOMO

Software Development, Maintenance Costs And Schedule Estimates

Cost, Schedule Distribution By Phases, Activity, Increment

COCOMO Recalibrated To Organization’s Data

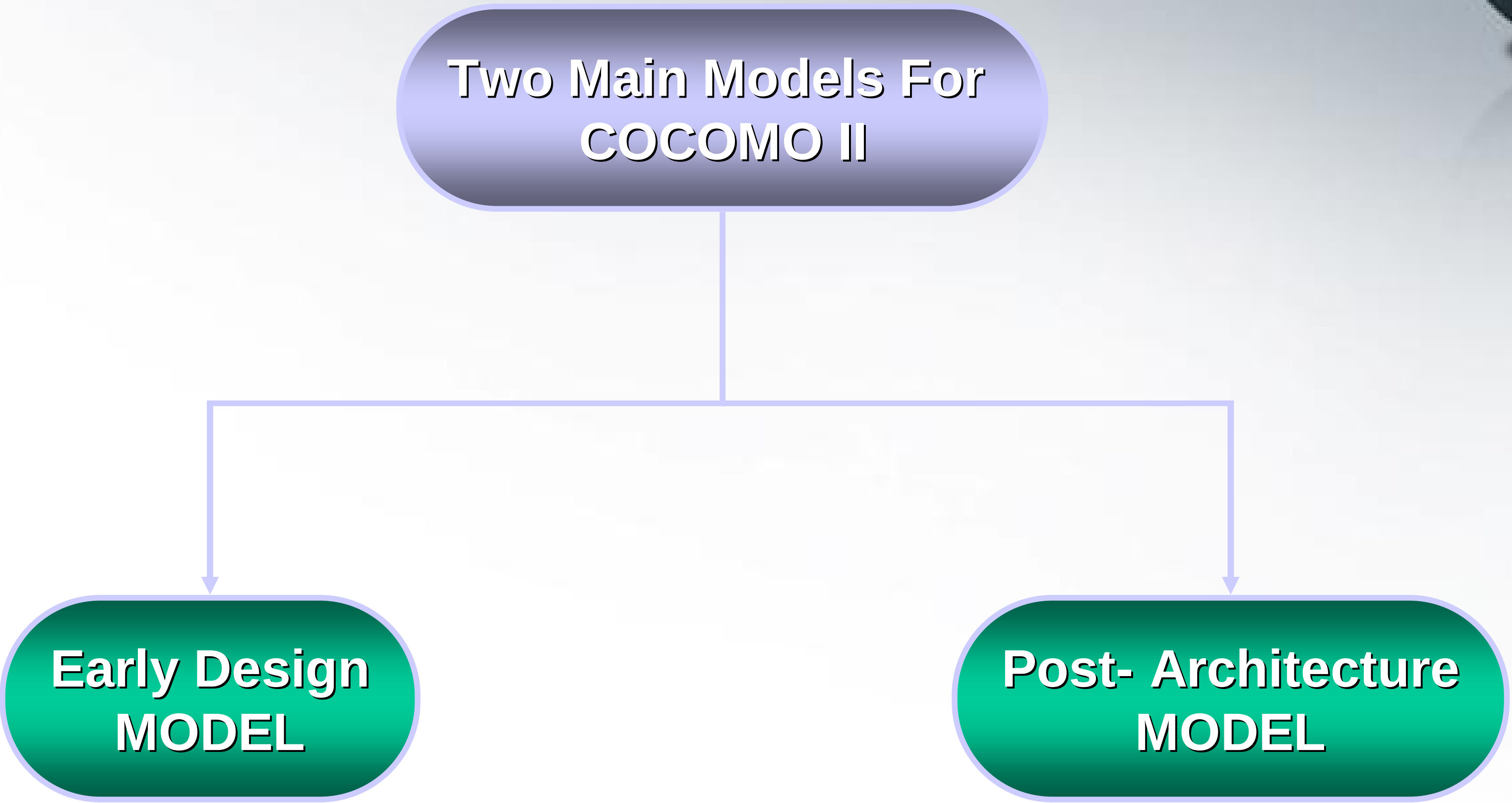
COCOMO II Objectives

- Develop Software Cost Model That Is Comprehensive and Addressing The New World Processes And Practices
- Retail COCOMO Internal And External Openness
- Develop Database Tool Support For Continuous Model Improvement
- Support Closed-loop Quantitative Project Management And Process Improvement

COCOMO II *MODEL* Objectives

- Provide accurate cost and schedule estimates for both current and likely future software projects.
- Enable organizations to easily recalibrate, tailor or extend COCMO II to better fit their unique situations.
- Provide careful, easy-to-understand definitions of the model’s inputs, outputs and assumptions.
- Provide a constructive model
- Provide a normative model
- Provide an evolving model

COCOMO II – *Model Definition*



The Basic Functional Formula For Both Are Same

COCOMO II – *Model Definition*

Base Formula

$$PM_{NS} = A \times (Size)^{E}$$

Where: -

- PM_{NS} = Person Month Effort For Nominal Schedule
- A = Constant [Indicating Normal Productivity] For COCOMO II – 2.94
- E = Measure of the total Economies / Diseconomies of Scale
- Size = **Unadjusted FP**, SLOC, KLOC or Unadjusted Usecase Count

The Person Month Thing..

Internally COCOMO II takes an average of
152 hrs Per Person Per Month
(This Includes adjustments for Weekends Off + Some Leaves)

This is an adjustable figure
there fore companies can take 170hrs / PM as their calculation value.

152hrs / PM is called the Nominal Value.

This Value Will Directly Affect The “***Duration***”

The Constant Value Of “A”

A = 2.94

(for COCOMO II – 2000)

This value cannot be altered.

Size

Could Be

Unadjusted FP, SLOC, KLOC

or

Unadjusted Usecase Count

The Variable “*E*”

$$E = B + 0.01 \times \Sigma SF$$

Where: -

- B = Constant [***0.91*** For COCMO II]
- SF = Scale Factors

Calculating Scale Factors “SF”

- There are 5 Scale Factors.

- These Scale Factors Account For The Relative Economies And Diseconomies Of Scale.

- These Scale Factors Are
  - Precedentedness (PREC)
  - Development Flexibility (FLEX)
  - Architecture / Risk Resolution (RESL)
  - Team Cohesion (TEAM)
  - Process Maturity (PMAT)

Calculating Scale Factors “SF”

Scale Factors	Very Low	Low	Nominal	High	Very High	Extra High
REC	Thoroughly Unprecedented	Largely Unprecedented	Somewhat Unprecedented	Generally Familiar	Largely Familiar	Thoroughly Familiar
	6.20	4.96	3.72	2.48	1.24	0.00
LEX	Rigorous	Occasional Relaxation	Some Relaxation	General Conformity	Some Conformity	General Goals
	5.07	4.05	3.04	2.03	1.01	0.00
ESL	Little (20%)	Some (40%)	Often (60%)	Generally (75%)	Mostly (90%)	Full (100%)
	7.07	5.65	4.24	2.83	1.41	0.00
EAM	Very Difficult Interactions	Some Difficult Interactions	Basically cooperative interactions	Largely cooperative	Highly cooperative	Semaless Interactions
	5.48	4.38	3.29	2.19	1.10	0.00
MAT	CMM L1 (Lower)	CMM L1 (Upper)	CMM L 2	CMM L 3	CMM L 4	CMM L 5
	7.80	6.24	4.68	3.12	1.56	0.00

Calculating Scale Factors “SF”

- If E < 1.0 It means the project exhibits economies of scale which means if Say project size is doubled than the effort is ***less than*** doubled.

- If E = 1.0 the economies and diseconomies of scale are in balance.

- If E > 1.0 the project exhibits diseconomies of scale

Calculating Scale Factors “SF”

- Say if all the Scale Factors were “Very Low” it would total to ***31.6***
- In this case the value of ***E = 0.91 + 0.01 X 31.6***
- i.e., ***E = 1.226***

- Say if the scale factors were “Extra high” then it would total to ***0.00***
- In this case the value of ***E = 0.91 + 0.01 X 0.00***
- i.e., ***E = 0.91***

The Base Formula

Applying the values of E in the base formula: -

Where E = 0.91
Say UFP = 100

Than the base formula is

PM = A X Size^{E}

PM = 2.94 X (100)$^{0.91}$

PM = 2.94 X 66.06

PM = 194.22

Where E = 1.226
Say UFP = 100

Than the base formula is

PM = A X Size^{E}

PM = 2.94 X (100)$^{1.226}$

PM = 2.94 X 283.14

PM = 832.43

The Nominal Schedule

$$TDEV_{NS} = C \ X \ (PM_{NS})^{F}$$

Where:

- TDEV = Time To Develop (Nominal Schedule)
- C = Constant with the value of ***3.67***
- F = ***D + 0.2 X (E – B)***
- D = Constant with value of ***0.28***
- E = Calculated by summing up the scale factors
- B = Constant with value of ***0.91***

The Nominal Schedule

So what is the nominal schedule for a PM_{NS} of 100 and E of 1.226

$$TDEV = C \text{ X } (PM)^{F\, =\, \boldsymbol{D + 0.2 \text{ X } (E - B)}}$$

$$TDEV = 3.67 \text{ X } (100)^{F\, =\, \boldsymbol{0.28 + 0.2 \text{ X } (1.226 - 0.91)}}$$

$$TDEV = 3.67 \text{ X } (100)^{\boldsymbol{0.34}}$$

$$TDEV = 3.67 \text{ X } 4.78$$

$$TDEV = 17.54$$

So What About *Enforced* Schedule?

- The dictated schedule would be different from the Nominal Schedule.

- So a ratio of SCED (Required Development Schedule) is calculated as "***EXPECTED DURATION*** / ***NOMINAL DURATION***"

- The ratio is compared with a table as under…..

SCED Descriptions	75% of Nominal	85% of Nominal	100% Of Nominal	130% Of Nominal	160% Of Nominal	Extra High
The Multiplier	1.43	1.14	1.00	1.00	1.00	N/a

Example?

If TDEV = 17.54

And Dictated Schedule is Say 11 Months.

Than the SCED = 11 / 17.54

SCED = 0.63

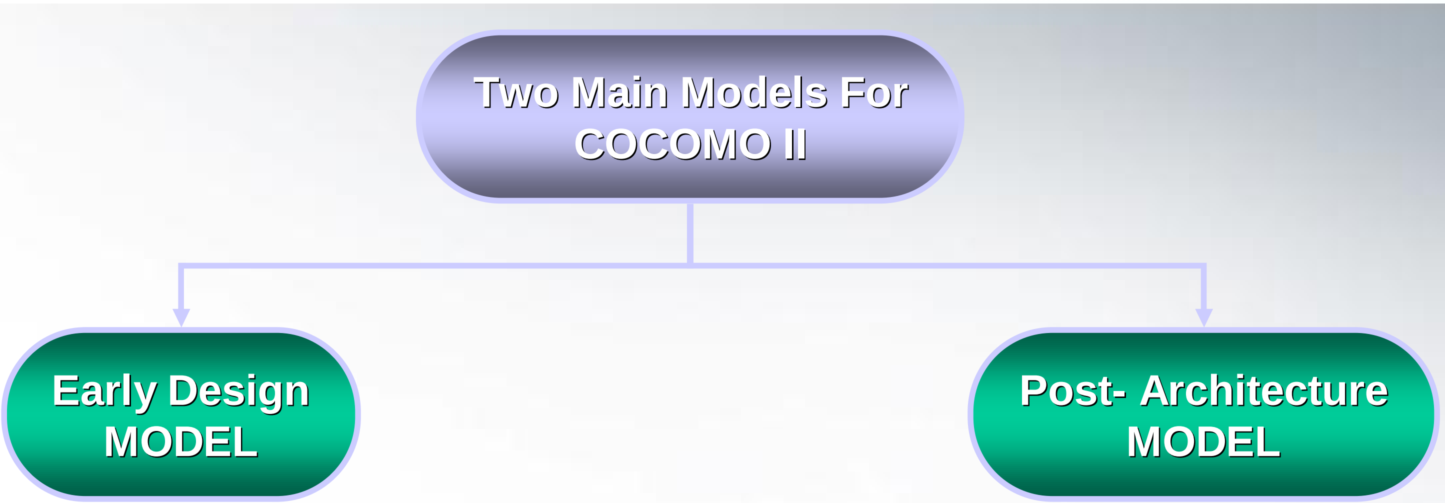
This is closest to 75%

So the applicable multiplier = ***1.43***

Therefore the PM Effort For This Asked Schedule = PM X 1.43

100 X 1.43 = 143 Person Months (adjusted)

Lets Talk About Effort Multipliers



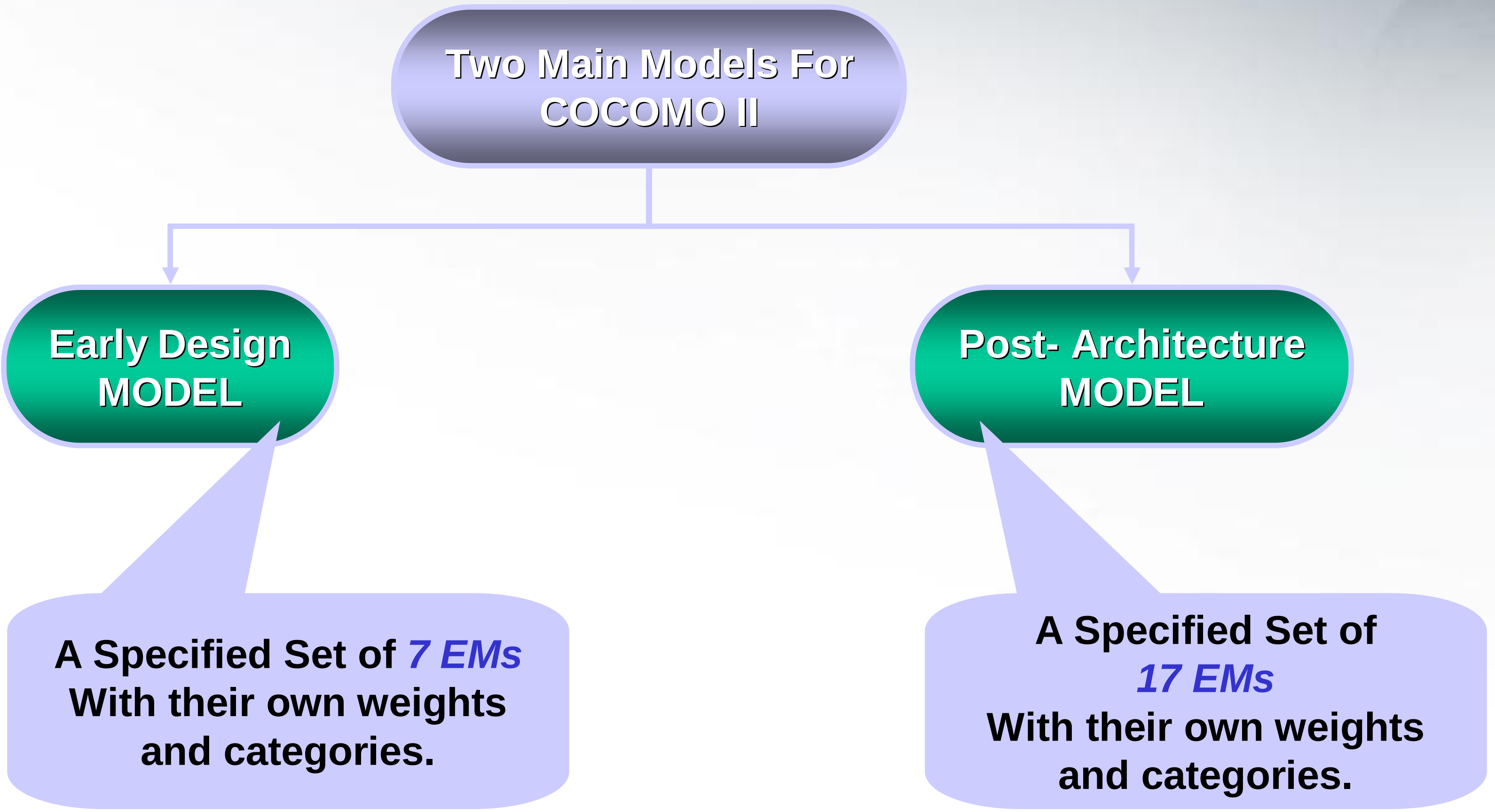
The Basic Functional Formula For Both Are Same

But what makes them really different is the concept of Cost Drivers / Effort Multipliers

Basic Formula And The EMs

$$PM_{NS} = A \times (\text{Size})^{E} \times \boxed{[\pi EM]}$$

Lets Talk About *EMs*



Two Main Models For COCOMO II

Early Design MODEL

Post- Architecture MODEL

A Specified Set of *7 EMs* With their own weights and categories.

A Specified Set of *17 EMs* With their own weights and categories.

EMs For EARLY DESIGN

This model is used in the early stages of a software project when very little may be known about the size of the product to be developed.

Hence there are only 7 EMs for the Early Design Effort Model

They are: -

1. Product Reliability And Complexity (RCPX)
2. Developed For Reuse (RUSE)
3. Platform Difficulty (PDIF)
4. Personnel Capability (PERS)
5. Personnel Experience (PREX)
6. Facilities (FCIL)
7. Required Development Schedule (SCED)

EMs For EARLY DESIGN

S. No.	EM Descriptions	Extra Low	Very Low	Low	Nominal	High	Very High	Extra High
1	Product Reliability And Complexity ***(RCPX)***	0.49	0.60	0.83	1.00	1.33	1.91	2.72
2	Developed For Reuse ***(RUSE)***		n/a	0.95	1.00	1.11	1.23	n/a
3	Platform Difficulty ***(PDIF)***			0.87	1.00	1.29	1.81	2.61
4	Personnel Capability ***(PERS)***	2.12	1.62	1.26	1.00	0.83	0.63	0.50
5	Personnel Experience ***(PREX)***	1.59	1.33	1.22	1.00	0.87	0.74	0.62
6	Facilities ***(FCIL)***	1.43	1.30	1.10	1.00	0.87	0.73	0.62
7	Required Development Schedule ***(SCED)***		1.43	1.14	1.00	1.00	1.00	n/a