



FPA and Portfolio Counting

– Size for Application Maintenance projects the easy way

Christine Green

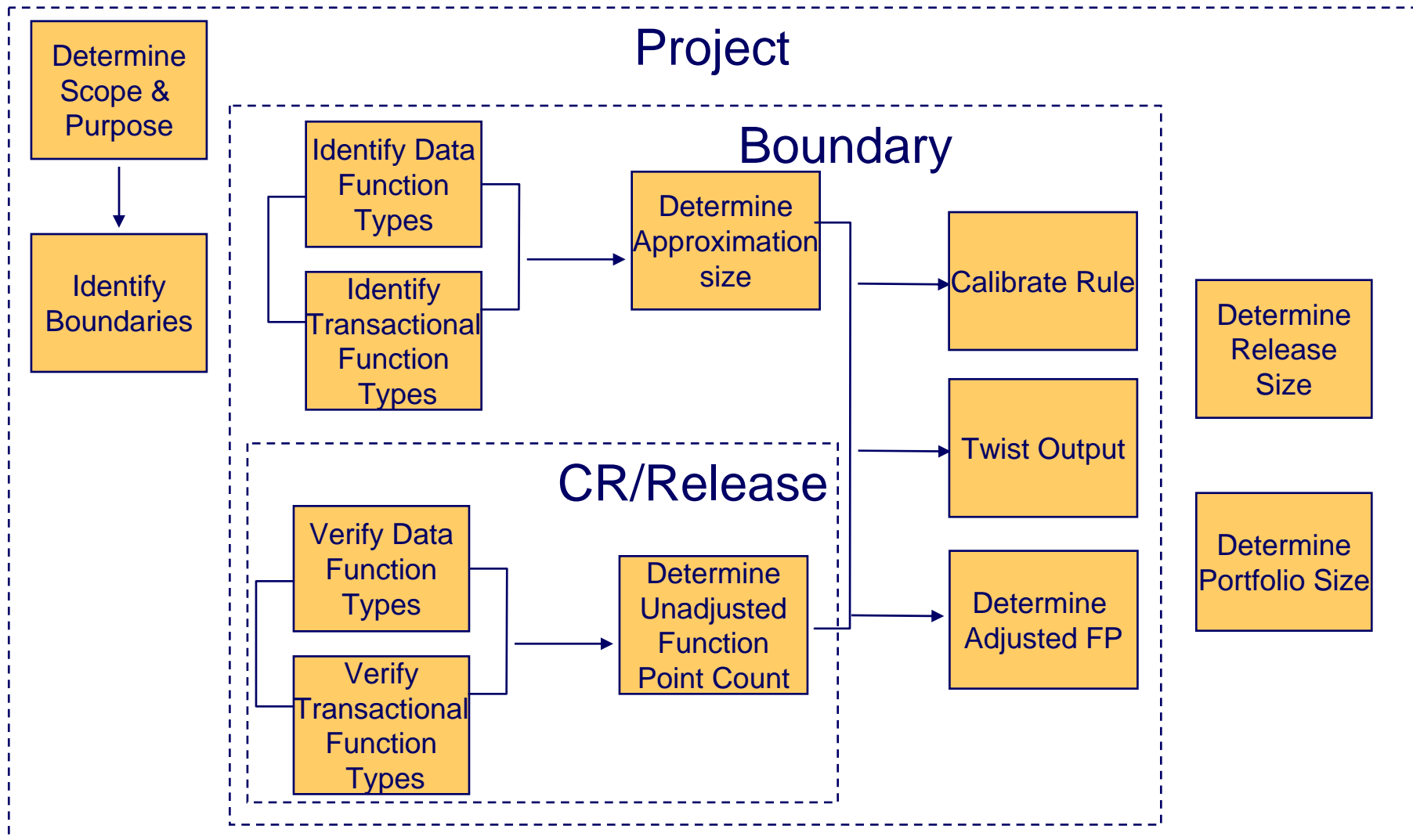
CFPS

••• Objectives

- To explain how Application Portfolio Function Point counting can be done with high ROI
- To show how FPA can provide Process improvement and estimating.
- To explain how Application FPA can be used to increase Client satisfaction
- To identify a Process where Portfolio count can be done in an easy way with a quick Application size output – with limited effort and focus on more than just SIZE



••• The process overview



⋮⋮ The history of the Project example



Hated FPA, Quality and the likes

I use the Project example as background – but only snapshots – the true story is a LOT bigger than the FPA and size measures shown in this presentation

- Major old project (actually it was a program with multiple projects)
- Many different people involved
- Issues with:
 - Estimates
 - Release Planning
 - Quality of the documentation
 - Client satisfaction
 - Client changes late in process

❖❖❖ Determine Scope and Purpose

FPA Questions

- Accuracy of a FPA
- Time used to do FPA
- What is the FPA to be used for?

Project Questions

- Level and Quality of Documentation?
- Quality of the estimating?
- Major issues within the project?
- Communicate with the Client?



❖❖❖ Determine Scope and Purpose - Example

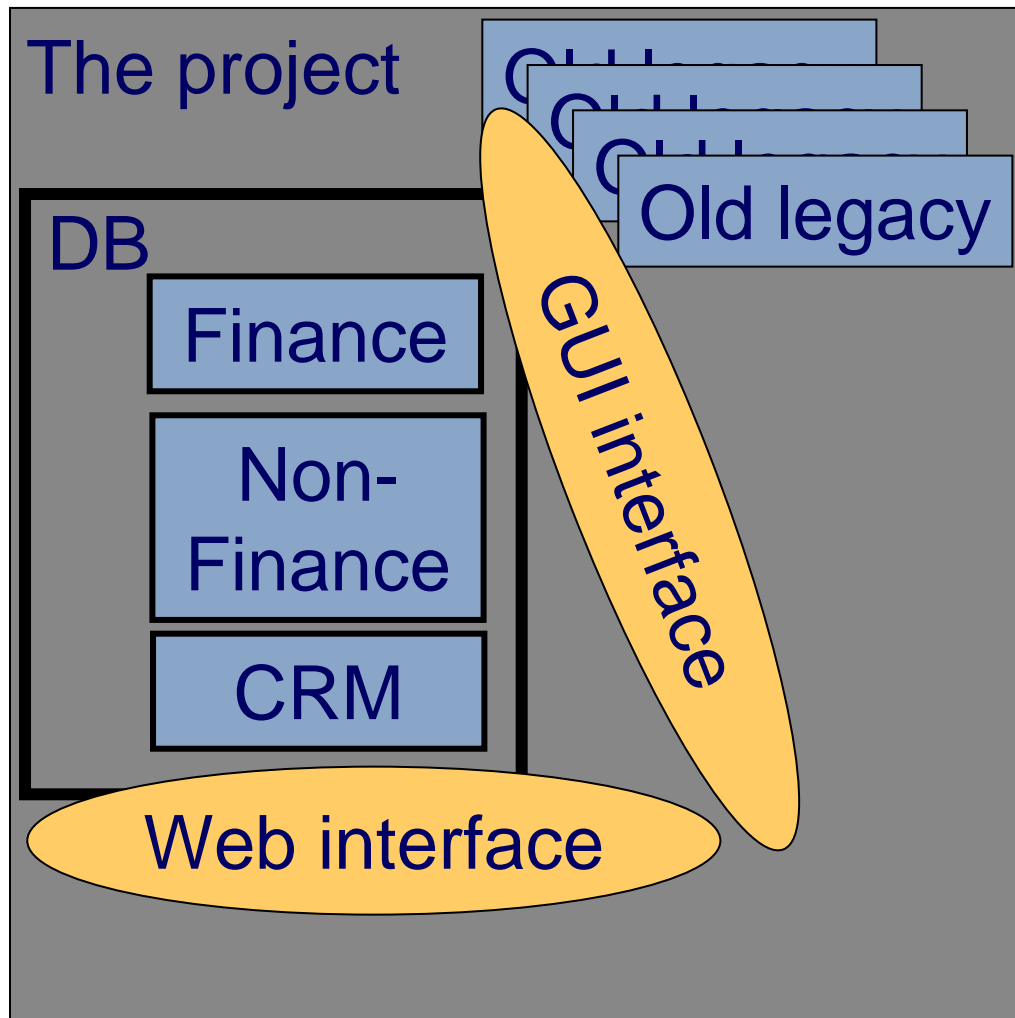
- To identify and create missing documentation
- To create size of application – as accurate as possible
- To make Application Maintenance estimate and contract based on size.
- To create a strategy for Release planning based on size
- To size each Release with an accuracy of +/-10%

Be aware that even though this was the scope for the work – it actually included several Function Point Analysis - where lower level scope and purpose exist

Identify Boundaries

- Ask for documentation that are drawings
 - Entity relationship diagram
 - Technical solutions
- Have a meeting with the project stakeholders
 - Include technical people and project manager
 - Include people that ensure estimates and evaluate Change Request
- Make a logical diagram of the project and application and technical solution – together with the stakeholders

Identify Boundaries - Example



Boundaries:

Finance

Non-finance

Old Legacy 1-?

Customer CRM

Issues:

All tables in ONE BIG
DB

No common
documentation

Identify Data Functional Types

1. Identify a list of all tables in DB
2. For all tables Identify if possible number of attributes – preferred the naming of the of the attributes
3. Together with DBA to identify naming convention
4. Identify possible duplicates and "test" and Control tables
5. Identify possible RET
6. Record the information – together with a FPA assumption of type and complexity
7. Make approximation based on the assumptions made

Identify Data Functional Types– Example

The list identified

Entity	Attributes
Employee	14
TX400	2
Organisation	5
Manager	2
Job-describtion	10

Assumption of type & complexity

Data Element	DET	RET	Type
Employee	14	1	ILF
TX400	2	1	ILF
Organisation	5	1	ILF
Country	2	1	ILF
Job-describtion	10	1	ILF

Most of the table names seem to be using names that are easy to understand – and is therefore assumed to be ILF's

When talking with the DBA – it turns out that TX is an acronym for a Legacy system, and that these tables are used to Temporary store information send from the application to Legacy



Determine Approximation Size – Example using only Data functional Types

Approx	Data Element	Multiplier	DET	RET	Type
1	Employee		14	1	ILF
0	TX400		2	1	ILF
1	Organisation		5	1	ILF
0	Country		2	1	ILF
1	Job-description		10	1	ILF

Use DCG Rules from IFPUG 2002

Rule of 28

FPC = Data Stores X 28

Rule of 4

FPC = 4 X ((10 * ILF) + (7 * EIF))

Assumption:

Less than 3 DET's is a control file

Issue:

Accuracy of the approximation might not be good enough

Identification only based on technical Solution

The approximation gives you the Possible valid data elements.

☼☼☼ Identify Transactional Functional Types

1. Identify a list of Modules, procedures, classes or other objects used
2. For all Objects identify naming convention together with developer
3. Identify FTR's if possible
4. Record the information – together with a FPA assumption of type
5. Make approximation based on the assumptions

Be aware that if multiple lists are possible you might need to chose the one list that makes most sense from a FPA point of view



Identify Transactional Functional Types – Example

Procedure
Maintain Employee INT400
Maintain Organisation List Manager

Approx	Transaction	Type
4 0	Maintain Employee INT400	EI
4 1	Maintain Organisation List Manager	EI EQ

Start by just making the list

If possible use it to identify transactions – if not include it as a reference for future use

- Make assumption about type and approximation
- Use standard assumptions – Maintain means Create, Update, Delete and Find
- List - is always identical with EQ's
- Total included in the procedure always identical with EO's



Determine Approximation – Using both Data functional and Transactional types

- For each ILF – use the identified complexity
- For each transaction use average complexity
- Compare the two result – the rule of 28 and the approximation using assumption

Remember two numbers is better then one numbers

Type	#
ILF	3
EI	8
EQ	1

Rule 28	Size
$3*28$	84

Type	Size	Comment
ILF	21	3 Low compexity
EI	32	Assumption average size
EQ	4	Assumption average size
Total	57	

⋮ FPA on CR's or Release

Guide by the Enhancement counting method

1. For each CR counted - identify and document details
2. For each Data element identified – verify that this is a user identifiable group of logically data
3. For each ILF identify what elementary process is used for maintenance
4. For each Data element – verify the DET's and RET's
5. For each transaction identify that type reflects the primary intent.
6. For each transaction identify the File Type referenced
7. For each transaction identify DET's

Adding to the documentation Add, Delete, Chg or Conv will give you even more information than my example

Verify Data Functional Types – Example

Approx	Data Element	Multip	DET	RET	Type	CR	Comment
1	Employee	1	24	2	ILF	CR1, CR2, CR3	
0	TX400	0	2	1	ILF	CR5	A temp table
1	Organisation		5	1	ILF		
0	Country	0	2	1	ILF	Control	Update request
0	Job-description	0	10	1	ILF	CR3	Employee RET

- Counting using the information from the CR's verifies the type and complexity
- The multiplier is introduced to be a part of the Function Points identified.
- The FPA verifies the assumption made about the control tables for one of the control tables and valid data elements
- For EIF's identified - don't include approximation for Rule of 28 – but you might introduce a new column and use the results to use the Rule of 4.
- EIF – record the application reference

Verify Transactional Function Type – Example

Approx	Transaction	Multipl	Type	DET	FTR	CR	FTR reference	Comment
2	Maintain Employee	2	EI	26	1	CR1, CR2, CR3	Employee	update, create
1	Maintain Employee	1	EI	2	1	CR1, CR2, CR3	Employee	Delete
1	Maintain Employee	1	EQ	26	1	CR1, CR2, CR3	Employee	Find
1	INT400	1	EO	40	3	CR5	TX400	Table is temp
2	Maintain Organisation	2	EI			CR6	Organisation, Employee	Create, Update
1	Maintain Organisation	1	EQ			CR6	Organisation, Employee	Find
1	List Manager	1	EQ	4	0			Hard-coded?

Recommendation:

- Don't forget to document all you find during the FPA
- Multiple lines if transaction change either type or complexity
- If possible document enhancement type – add, del, chg, conv
- Document references to objects, test cases etc.
- Record logical name (Client identifiable name) if needed



Approximation Size after FPA verification- Example

Type	Number
ILF	2
EI	5
EO	1
EQ	3

Rule 28	Size
2*28	56

Type	Size	Comment
ILF	17	1 Average + 1 Low
EI	20	Assumption average size
EQ	12	Assumption average size
EO	5	Assumption average size
Total	54	

- Due to the verification the numbers is now much more close
- The approximation is updated - since not all is verified
- Only use multiplier on what is verified
- Use percentage calculation to update your approximated total size



Determine Unadjusted Function Point Count - Example

Transaction	Multip	Type	DET	FTR	Compl	UFP
Maintain Employee	2	EI	26	1	A	8
Maintain Employee	1	EI	2	1	L	3
Maintain Employee	1	EQ	26	1	A	4
INT400	1	EO	40	3	H	7
Maintain Organisation	2	EI			A	4
Maintain Organisation	1	EQ			A	4
List Manager	1	EQ	4	0	L	3
Transaction size						33
Data element size						17
<i>UFP</i>						50

Since the FPA method has been used to produce this result – this is Unadjusted Function Points

••• Calibrate Rule

- Use the FPA where CR's are identified – these are verified
- Calculate the % of verified UFP – against approximated or estimated Size
- Use the verified UFP to calibrate a change of Rule 28 – by finding the average UFP per ILF verified

Recommendation:

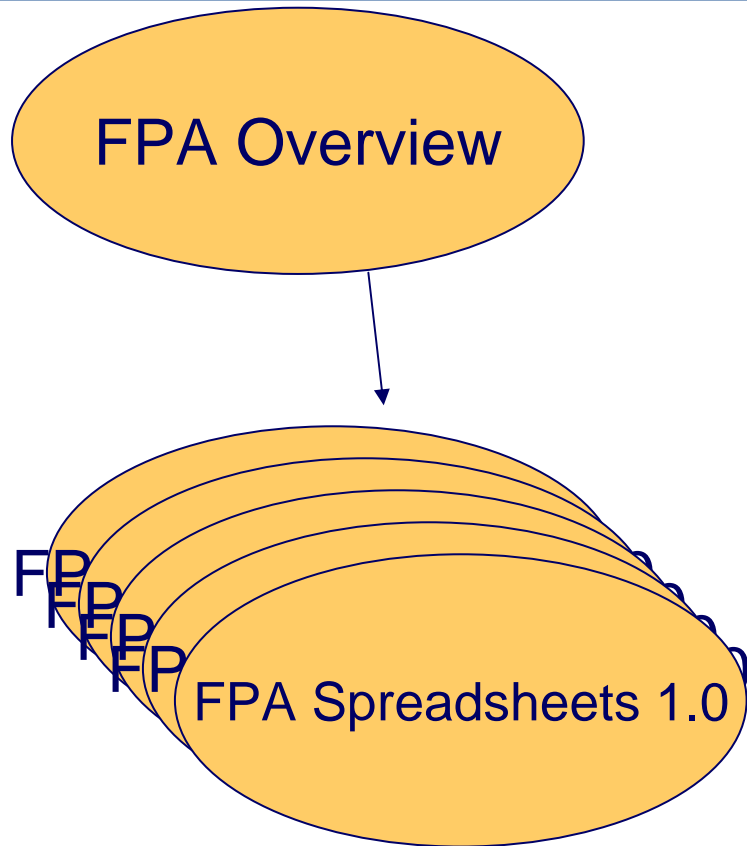
- Don't call the approximation or other size output Function Point before the verification It is a size that only MIGHT reflect the function Point size
- Only the verified transactions and Data elements has been Function Point Counted

❖❖❖ Twist Output - Example

- Twist the output around – and you will have a total new view to talk with the Client about
 - Size per CR
 - Transactions per table

CR	Transaction	Multip	Type	DET	FTR	Compl	UFP
CR1	Maintain Employee	2	EI	26	1	A	8
CR1	Maintain Employee	1	EI	2	1	L	3
CR1	Maintain Employee	1	EQ	26	1	A	4
CR1	Total CR size					15	
CR6	Maintain Organisation	2	EI			A	4
CR6	Maintain Organisation	1	EQ			A	4
CR6	Total CR size					8	

Documentation is important – What and how



FPA Overview

- Document to be used to describe all decisions made to Boundaries and all counts made for the project.

The reason for this document is the fact that most projects are covering more than one boundary and therefore consist of more than one count. This document is to be used to have the overview of all counting activities done for the project incl. any approximation calculations done

❖❖❖ FPA Overview Template



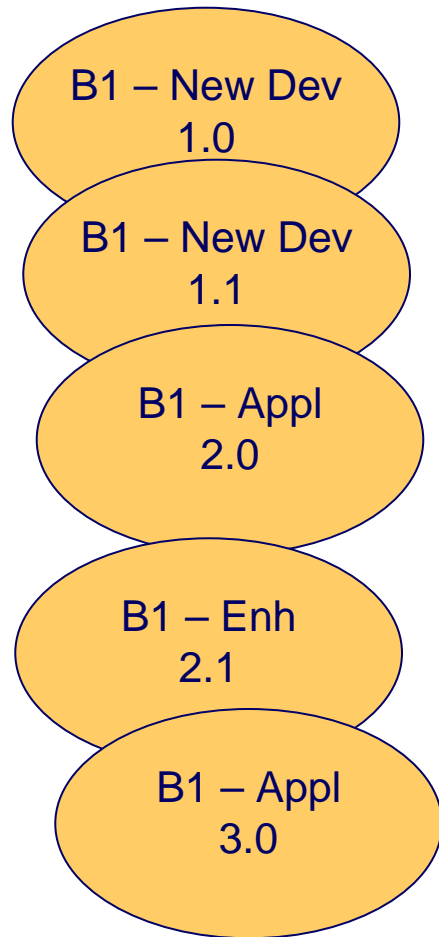
WHAT

- A high-level intro to the project being counted
- A high-level intro to all the boundaries Identified in the project – incl. the reason why the boundaries are identified as it is.
- A detailed description of each boundary (reuse wordings from and two the Scope and boundary definition in the FPA documentation)
- A high-level overview of the FPA's completed on the boundaries – such as size, type, who – comments.
- Always link or reference as a minimum validated and confirmed FPA documents to the overview

WHY

- Documentation in only FPA Spreadsheet – not good enough
- Boundaries where not linked to each other
- FPA where grouped – without consideration of boundaries
- No history for the work that had been created by the FPA Specialists
- Projects is returning to the other FPA Specialists for support
- **Documentation and overview is needed.**

❖❖❖ FPA spreadsheets example



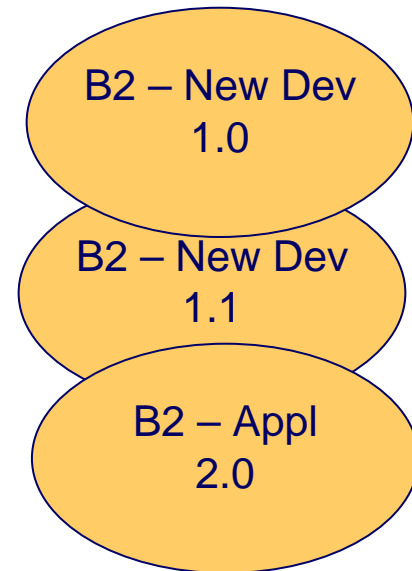
Two boundaries,
Two New Dev counts

Two boundaries,
Two Design counts

Two boundaries,
Two Baseline validation

Two boundaries,
Only B1 functionality Enhanced

Two boundaries,
Two Baseline validations





FPA – Transaction- and Data elements information

- Always put in references to documentation, work-products etc
- Put in extra lines and headings where it makes sense – give the reader overview
- Put Data elements in alphabetic order where possible
- Put transactions grouped by module, screen or other logical grouping where possible



⋮⋮ FPA – Assumptions

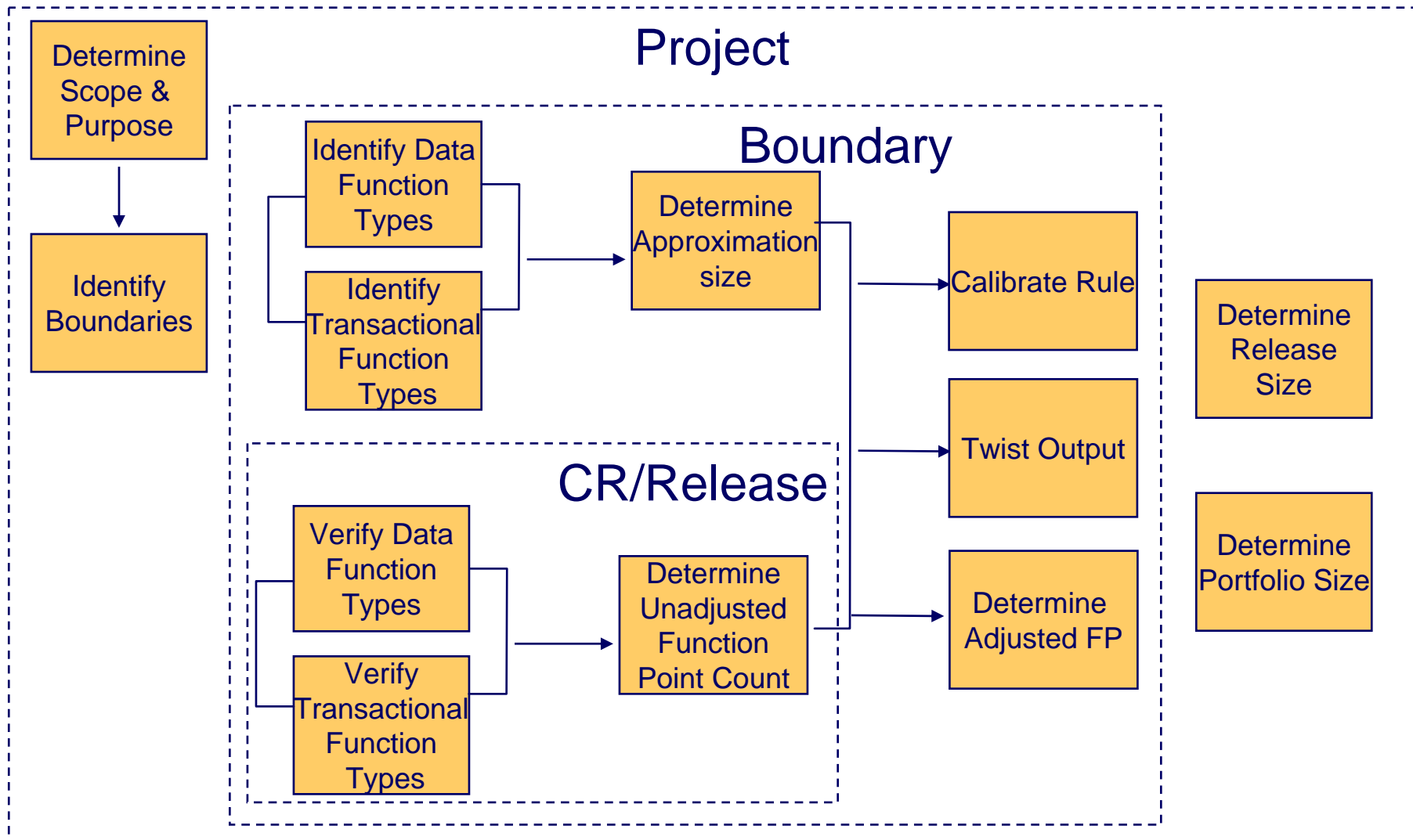


- Always mark down assumptions – all assumptions
- Try to mark down generic assumptions if possible
- Make sure that all assumptions are validated with either Business expert or FP Specialist – not left blank

⋮⋮⋮ What was delivered to the example project

- Size estimate early – that the project used to compare them self to history and benchmark information
- A list of all the procedures and the tables they referenced. This was a big missing piece in the documentation.
- A 75% UFP Analysis within 8 months – using 3 major releases CR
- Identification of which modules that where the most changed – Possible Re-engineering factor
- The twisted output – that was used by the Client to decide what CR's to go into the next release
- A procedure for sizing CR's and use the result to plan Releases together with Client
- By using the CR Size the estimating accuracy increased after two releases

••• The process overview





Lessons learned

- The more you document the more you need it later
- Twisting output is really useful
- Don't use the wording Function Points for something that is only an approximation – It can be misused
- Verify all you find during the analysis of the CR's
- Version control – and amended history AND assumptions and comments on all you do
- One size with low accuracy is better then no size
- FPA is perfect as CR peer review and analysis
- Try it out – before introducing it to the Client – you get wiser over time



eds.com

Contact information:

Christine.Green@eds.com