

The following presentation was given at:

SCAF Workshop “Project Cost Control”

Tuesday 2nd February 2016

The Royal Institution of Naval Architects, London

Released for distribution by the Author

www.scaf.org.uk/library

Cost modelling: A comparison of Project & Programme approaches

SCAF Workshop Theme: “Project Cost Control”

By Louise Carver and Neil Abbott

Tuesday 2nd February 2016

Contents

Project vs. programme approach

Key Factors

- Purpose & scope of the model
- Developing the model input
- Managing the model
- Maintaining the model

Summary

Conclusion

Project vs. Programme approach

Traditionally the MOD is focussed on project management where each project reports its progress independently of other projects

Recently there has been a move toward a more “programme-centric” approach driven by:

- Potential improved VfM, for example
 - Incremental delivery
 - Rather than “big bang”, roll out capability in smaller capability sub-sets
 - Optimise commonality, modularity and reuse

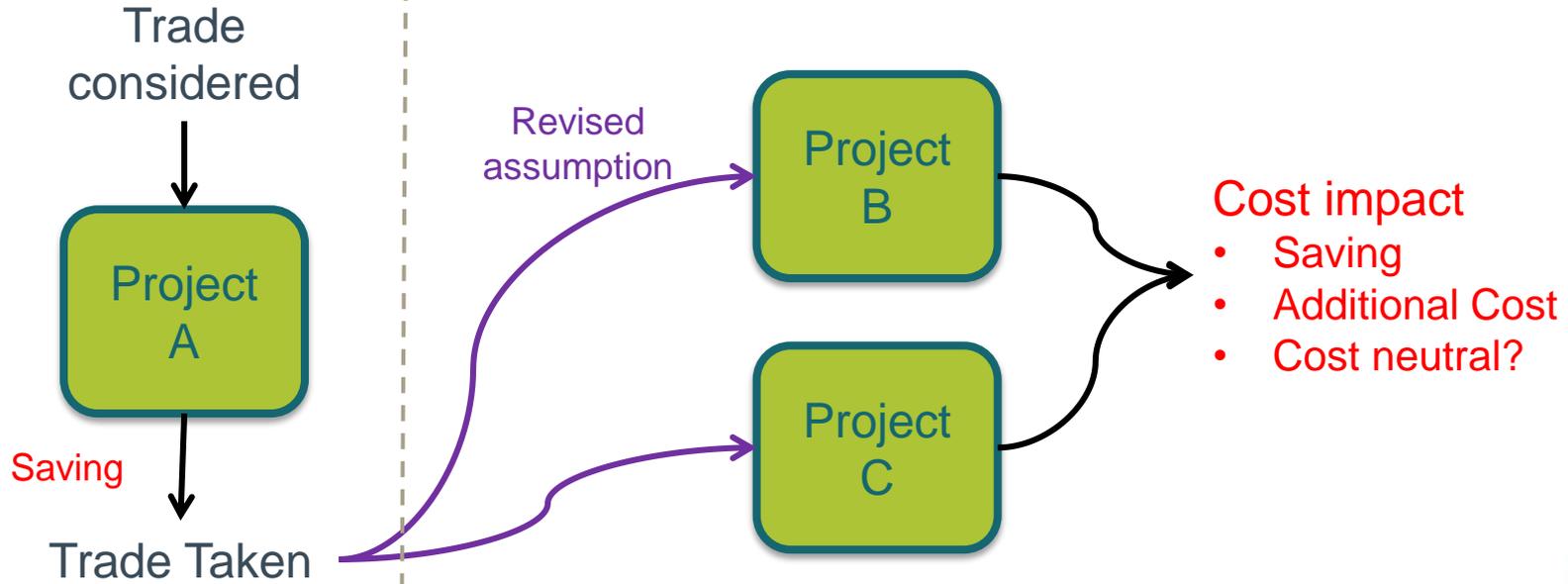
However, MOD processes and procedures are still generally project focussed.

QUESTION: How can the two views (project/programme) be maintained and monitored efficiently from a costing perspective?

Project vs. Programme approach

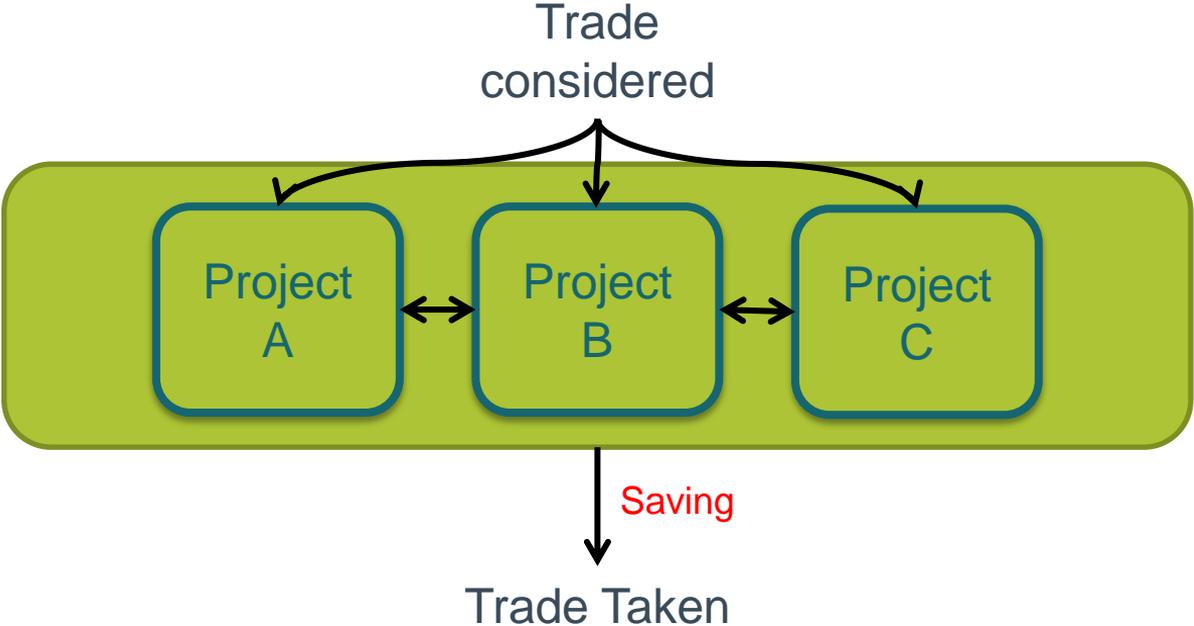
Project-centric view:

'Stovepiped' project



Project vs. Programme approach

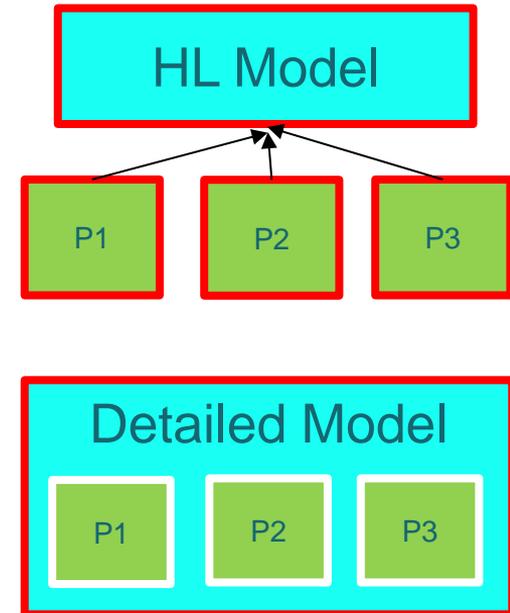
Programme-centric view:



Programme level cost modelling

In situations where cost forecasting needs to be managed at programme and project level. Two options are considered here:

1. Projects maintain their own cost models with an aggregated view uploaded to the programme level model (referred to as “High level model”) or
2. Maintain all project costing in a single model (referred to as “Detailed model”)



Key factors

Key factors to consider in selecting an approach:

- Purpose & scope of the model
- Developing the model input
- Managing the model
- Maintaining the model

Purpose & scope of the model (1)

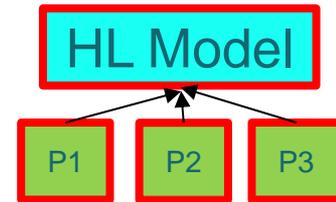
Formal reporting is typically maintained at project level i.e. for Project Approvals and Budgeting Review/Planning Rounds



- Detailed model for approvals means the model contains “excess” information not required for the approval in question

Where are the boundaries the model?

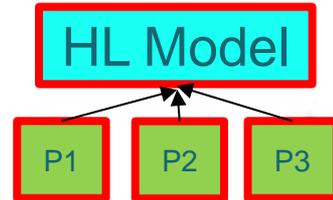
- Number of projects
- Scale
- Maturity
- Schedules alignment



Purpose & scope of the model (2)

Sensitivity analysis and trade off studies are more likely to be a requirement at the programme level

- The Detailed model maintains the dependencies and detail but may take considerable time to run
- Project level models may need to be independently run to confirm the results in the High Level model
- The following slides will consider the affect in development, maintenance and management of either option

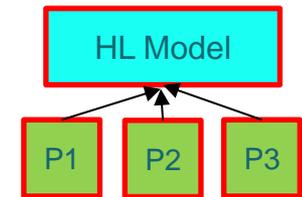


Developing the model input

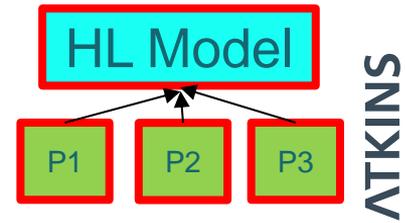
- Data collection (cost data including risk)
 - Responsibility of the project team regardless of the modelling approach applied.
 - Maintains project control and ownership
- At a programme level, data gathering processes can better manage
 - Stakeholders liaison to minimise multiple engagements on the same topics.
 - Review and alignment of common data and assumptions to ensure coherence and consistency across projects
 - Risk reviews to:
 - identify/manage programme level risks; and
 - support the development of more comprehensive project risk registers by considering whether risks identified in one project apply to another project (may be different score/impact)

Managing the model

- For the “Detailed model” this may be the responsibility of a small number of individuals, most likely within the programme team
 - A single model minimises the verification and validation effort
 - Potentially maximises use of SQEP (may be more beneficial to have 1 or more full time cost resource across the programme rather than several part time resources)
- For the “High level model”, project teams are responsible for their own project model, programme team responsible for the high level model
 - Are there benefits associated with models having similar structures?
 - Project teams can undertake their own sensitivities/analysis when required



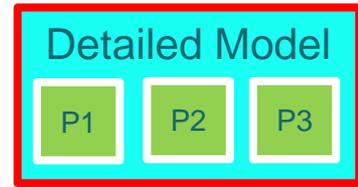
Maintaining the model (1)



Populating a high level model has a number of potential complexities that are not independent:

- Aggregation of data at the programme level – what level of fidelity is appropriate?
 - Multiple project model data sets need to be fed into the high level programme model
 - Detailed project risks or a composite view
- Need to maintain configuration control across multiple models

Maintaining the model (2)

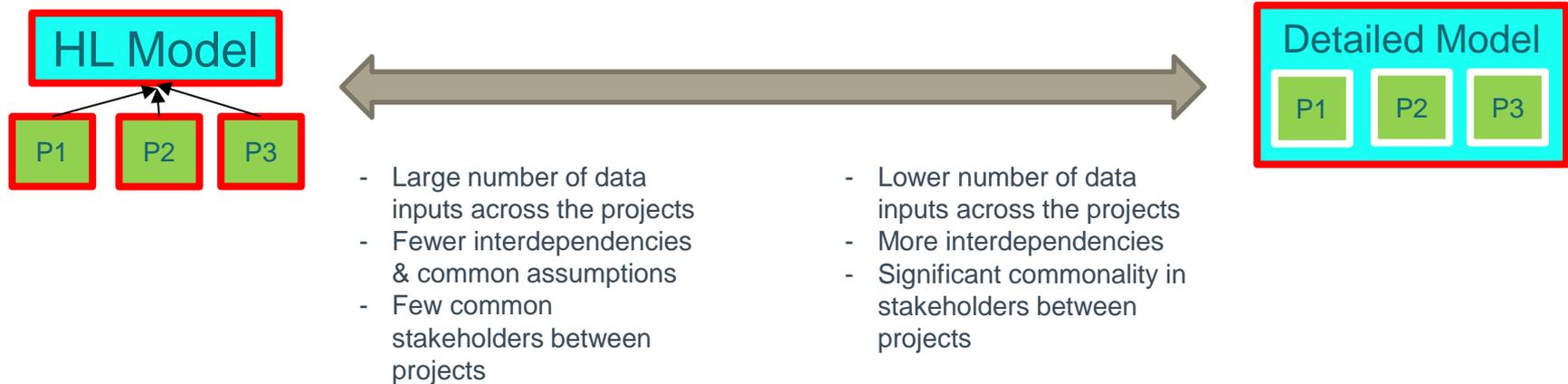


Populating a detailed level model is simpler but there are a number of other considerations:

- Larger volumes of data in one repository results in potential maintenance of data issues
 - Model tends to lend itself to smaller scale programmes
- Demands on model from different project communities requiring outputs for different purposes at the same time
 - Benefit is that all QRPC data can be produced for all projects at the same time
- Smaller group of people understand the model set up and configuration status
 - This may be a positive or negative feature!

Summary

These drivers can be plotted on a sliding scale showing which approach works best dependent on the situation



Conclusion

What are the key factors to consider in project and programme level models?

- **Purpose & scope of the model.** The level of fidelity required for sensitivities and trade off studies will influence data maintenance requirements.
- **Developing the model input.** The volume of data would influence the time invested in collecting large sets of data
- **Managing the model.** The approach to configuration control and level of V&V effort.
- **Maintaining the model.** The number of interdependencies will influence the time taken in managing model linkages.

Questions