

“A Pragmatic Approach to Large-Scale IT Program Performance and Health”

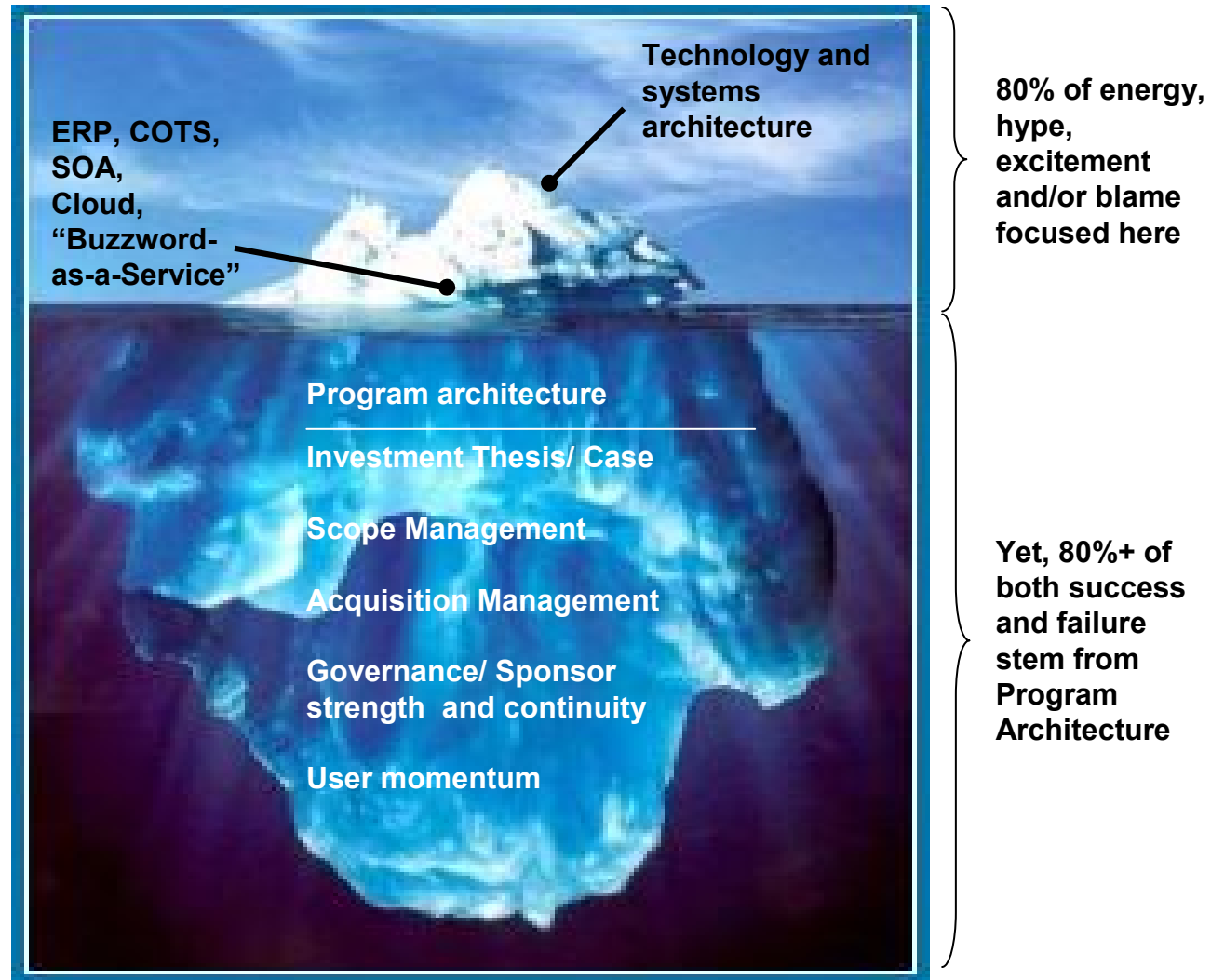
ArchitecturePlus Seminar, Washington DC
March 25, 2010

CONFIDENTIAL AND PROPRIETARY
Any use of this material without specific permission of McKinsey & Company is strictly prohibited

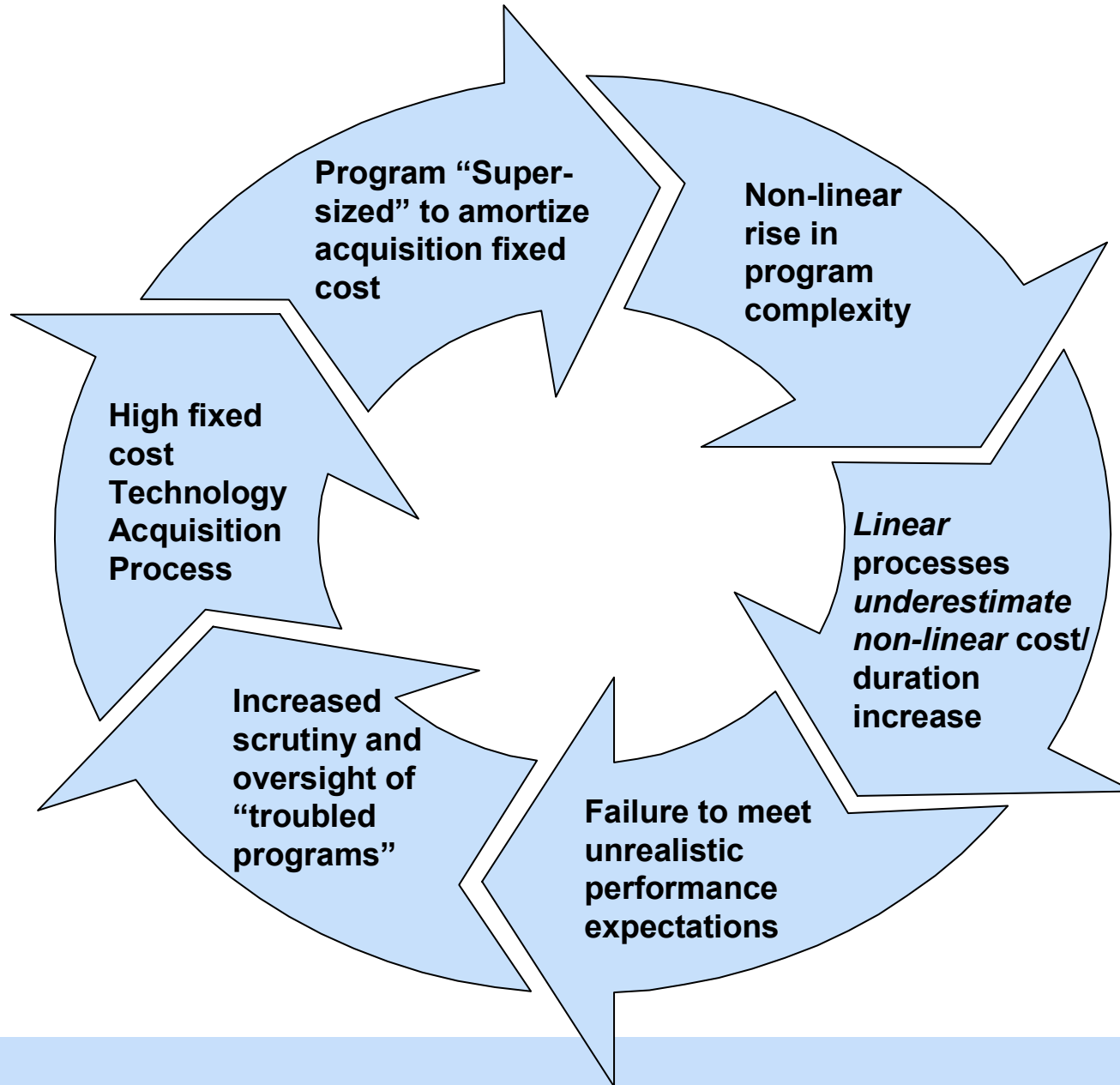
THE PUNCH-LINE

1. Sound system architecture choices are absolutely essential to good design and delivery
2. The N-squared Complexity problem: Large IT programs are *exponentially, not linearly* harder to manage than smaller Programs – INDEPENDENT of system architecture choices
3. Federal acquisition and oversight policies impose a high fixed cost on small IT. This creates a structural bias to put forward larger, and typically, more complex, and hence more “volatile” programs
4. Consequently, we believe that *non-linear complexity reduction* should be a central tenet to drive sound program architecture decisions
5. Based on sound empirical evidence and painful experience, we suggest systematic application of 5 core complexity reduction principles to improve the “performance” and “health” outcomes of large programs

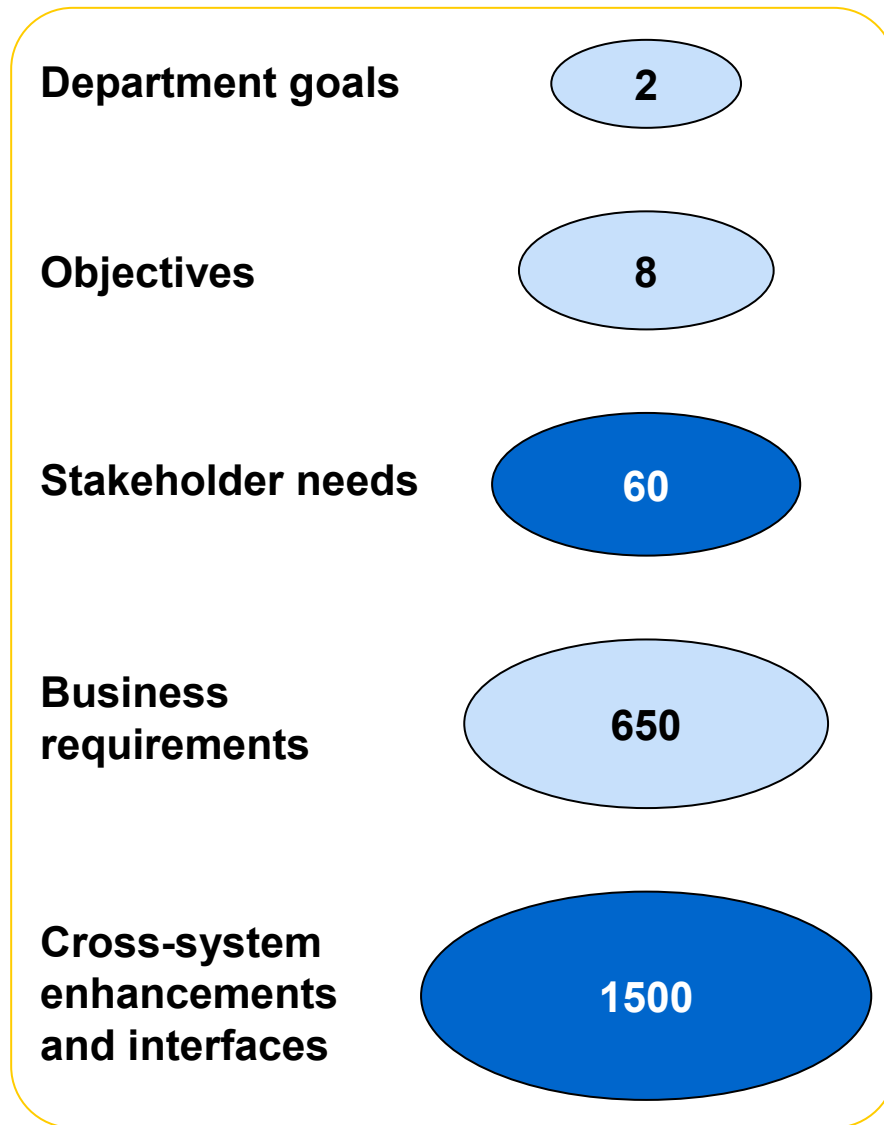
The Large Technology Program Iceberg



The Deadly Do-Loop: Challenge of Non-linear Complexity in Federal IT

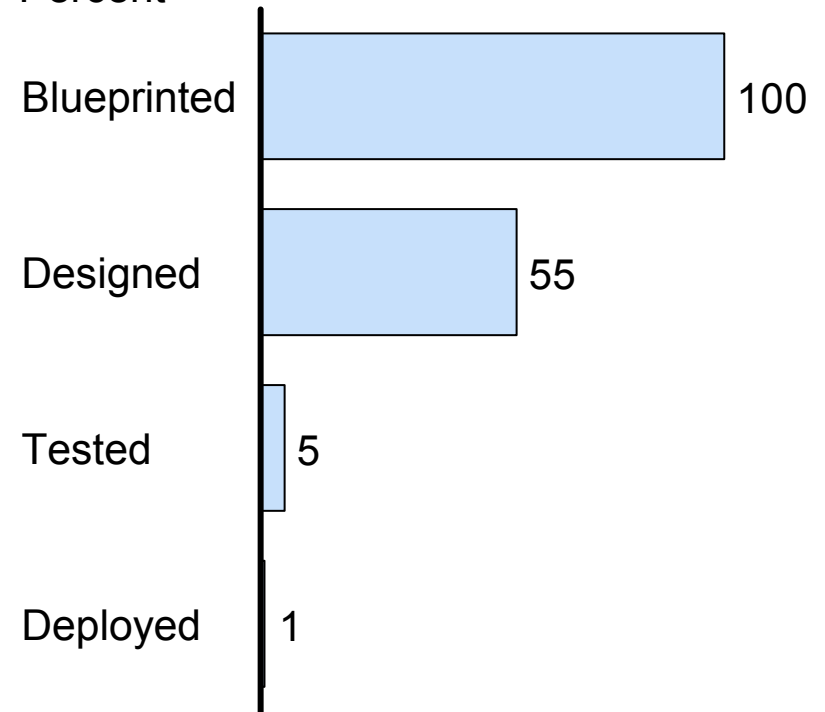


Stakeholder complexity often drives very high cost Non-linear interface complexity



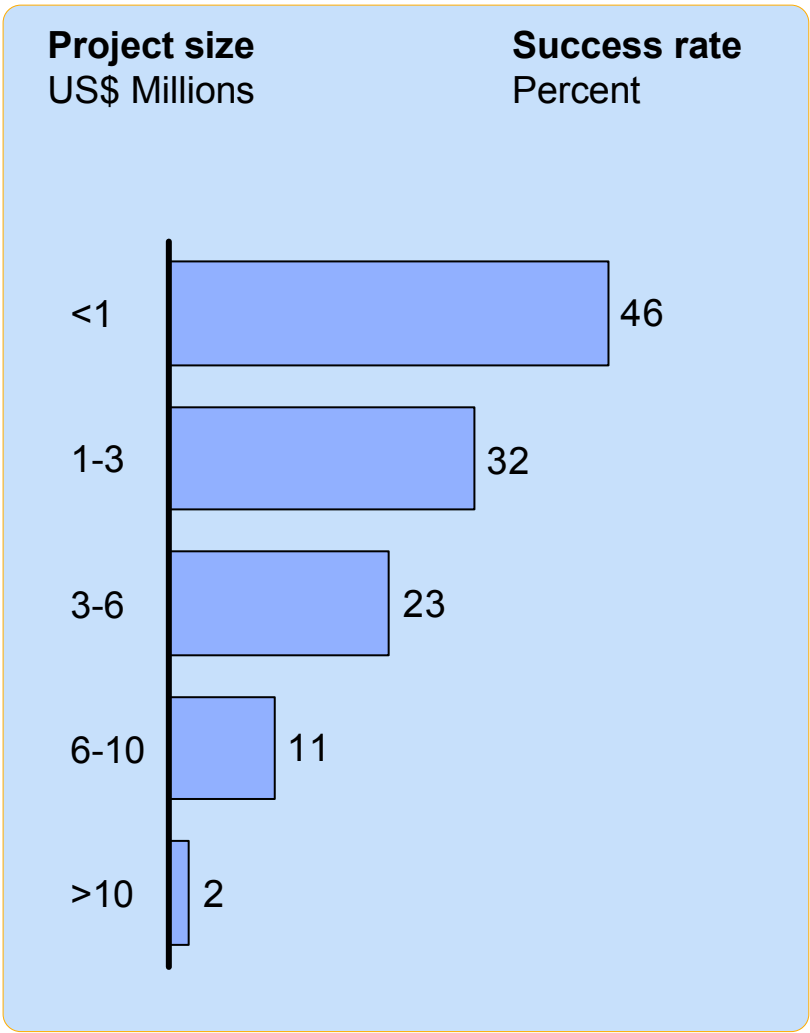
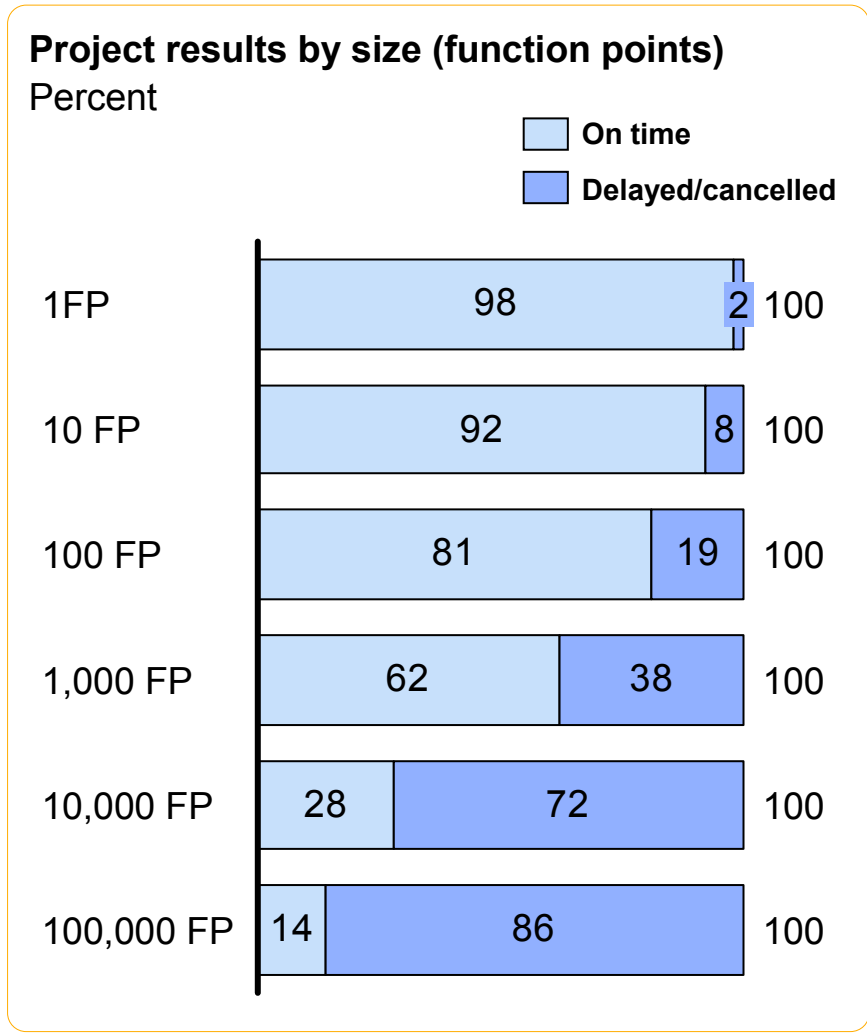
Program status 5 years from inception

Percent

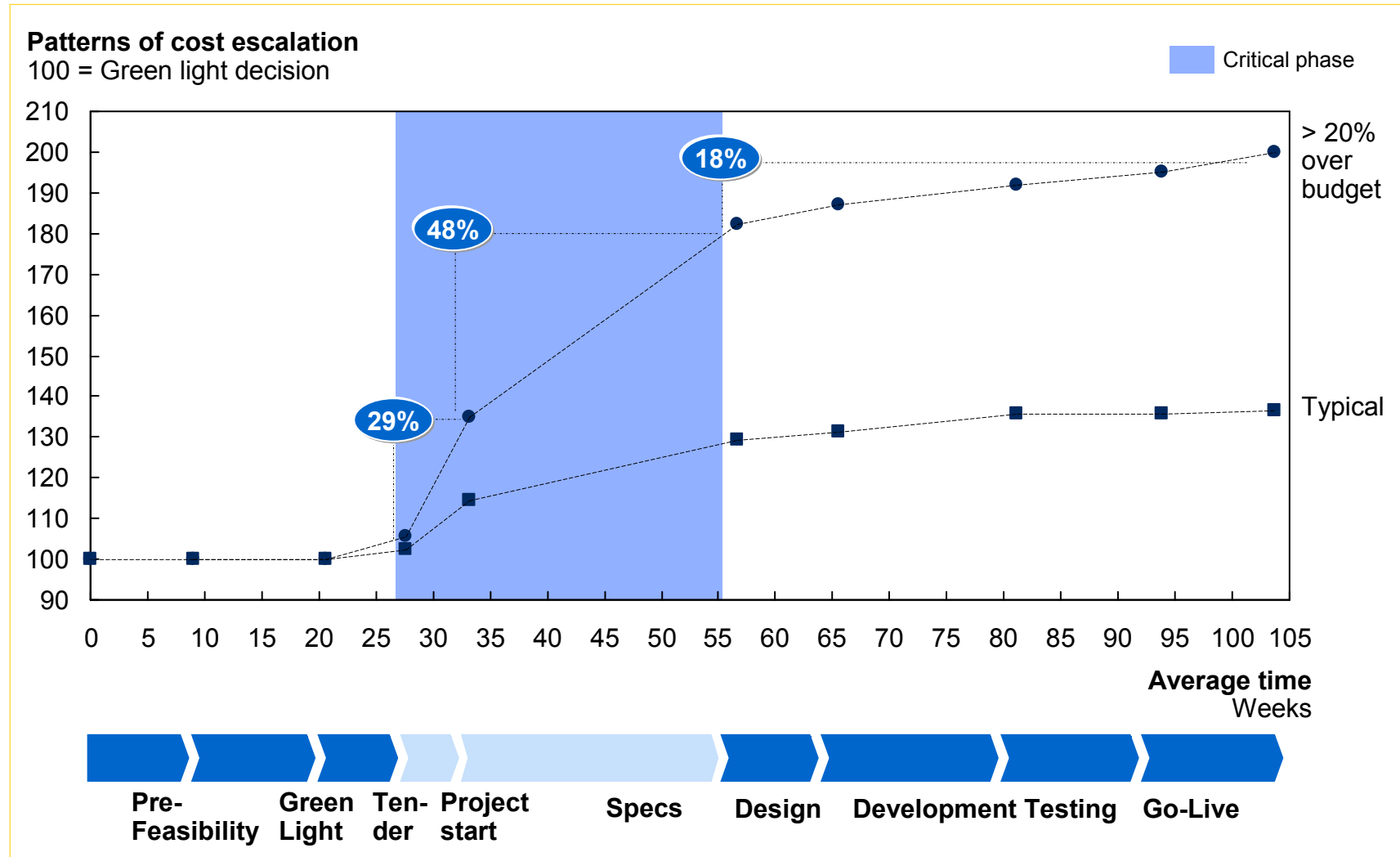


- Several hundred million dollars
- Very good documents
- Negligible deployment

Program size and complexity have non-linear impacts on timely delivery, even for modest programs...



... and for larger ones, the battle is often lost before its begun. 3/4 of total cost escalation happen virtually as soon as programs start



Managing non-linearity: Top 5 Program Architecture requirements for good “health” and “performance”

Expectations **ROBUST** to “FAILURE”

Program **MUST** be attractive **EVEN** assuming a 2-3x budget over-run and/or delay – otherwise it is not robust enough. Plan to “fail”, **BUT** ensure you still like the plan

SMALL stakeholder and sponsor set

Scope creep, veto power and lack of accountability rise *exponentially* with every incremental stakeholder

SHORT Impact Windows

Sponsor(s) and Program lead(s) must be in power for as long as it takes to deploy system, **OR** program **MUST** deliver **SOME** fielded capability every 6-12 months

Intensive and iterative **USER** involvement

New systems are like new habits. We don't like to change habits. Design to change user habits as painlessly as possible, or make it worthwhile to endure the pain

IN-SOURCED competence

Too often, agencies rely overly on vendors for Scope Governance, Technical Architecture Design and even Stakeholder Management. This creates asymmetric risk