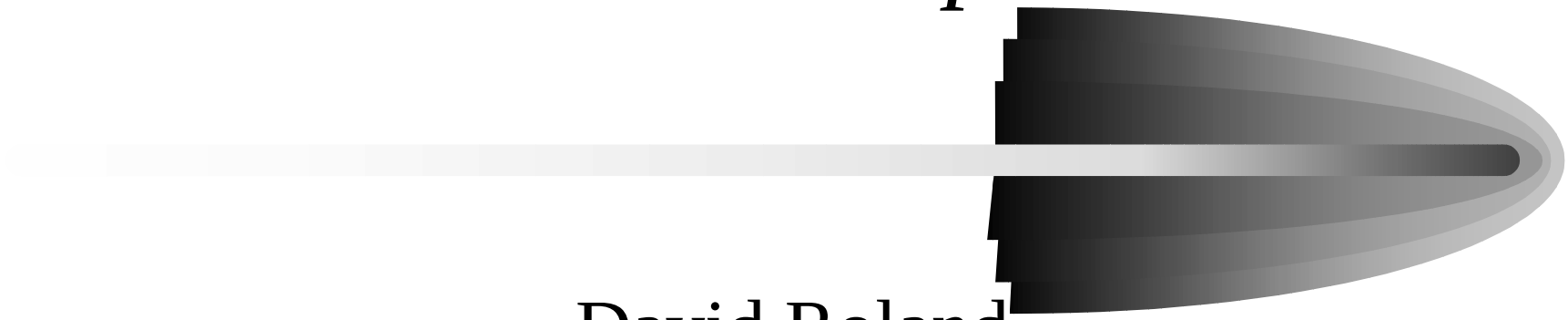
Lessons From Aerospace



David Roland

(droland@sutmyn.com)

SSQA August 11, 1998

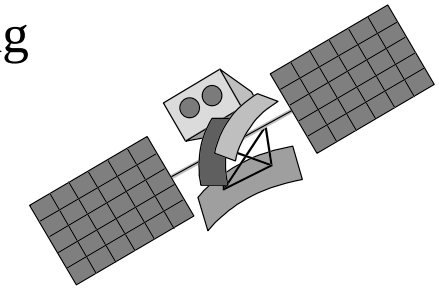
What lessons?

Benchmarking examines any related industries for best practices.

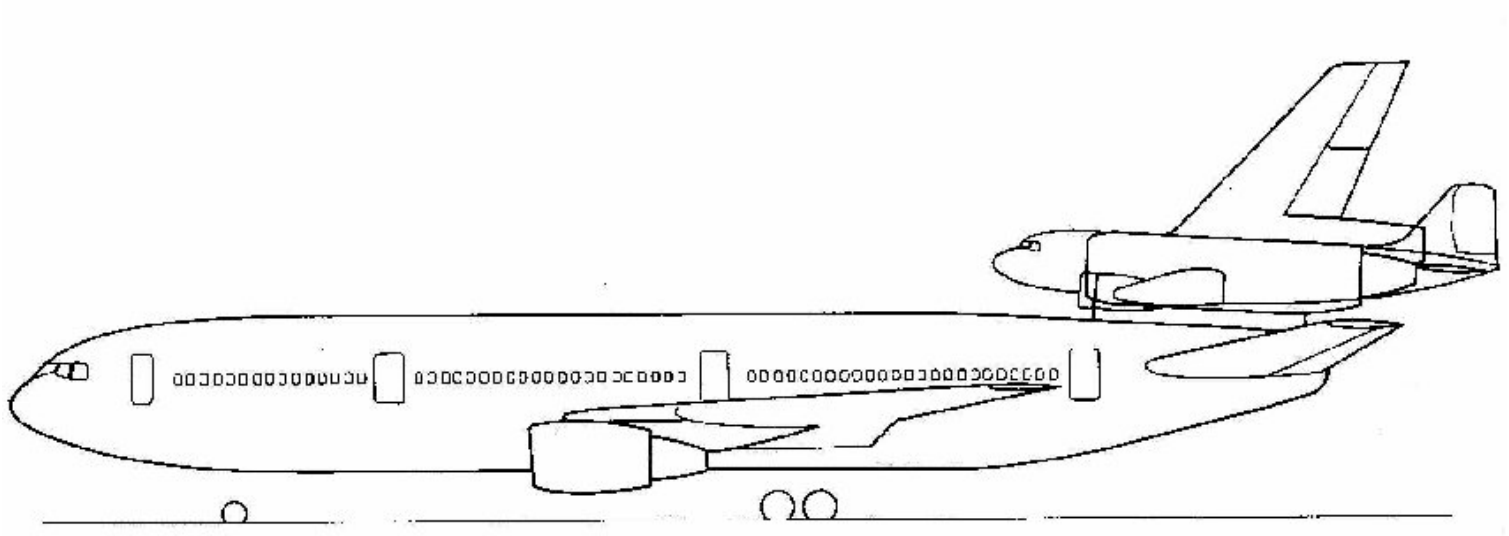
Why does the Aerospace Industry have lessons for the Software Development Community?

Similarities

- Aerospace
  - Performance growth
    - Kitty Hawk to ???
  - Increasing complexity
    - Engines/Airframe/Avionic
  - Range of project sizes
    - Rutan to Boeing
- Software
  - Performance growth
    - ENIVAC to ???
  - Increasing complexity
    - Web/DBs/Embedded
  - Range of project sizes
    - Web Pages to ATC

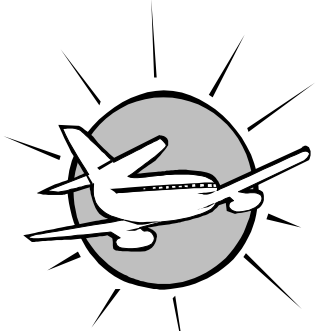


Large performance growth



Differences

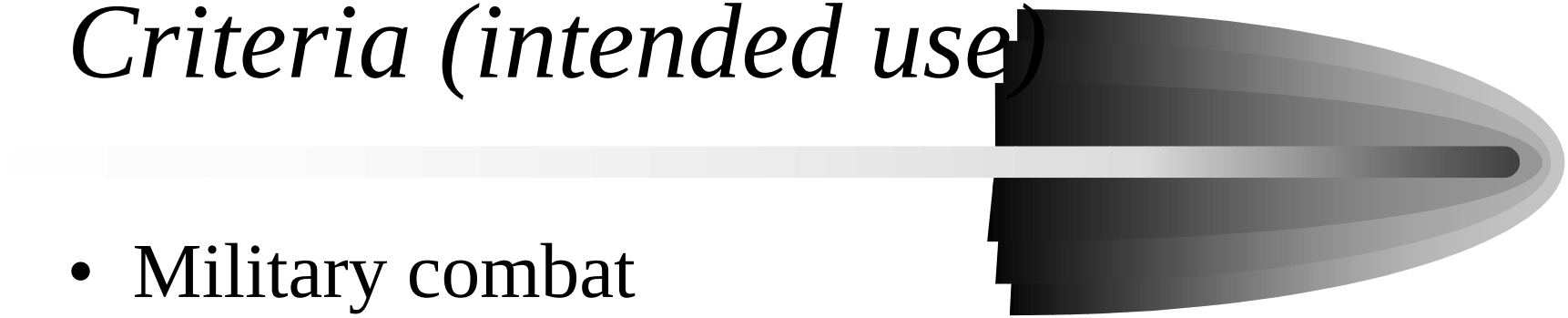
- Aerospace
  - Becoming more reliable
    - Except software
  - Regulated
    - CAB and FAA
  - Standard Practices
    - FAR Part 25 and company
- Software
  - Becoming less reliable
    - ATC project failure
  - Unregulated
    - For how long?
  - Ad hoc practices
    - Methode de jour



Aerospace practices

- Engineering criteria...
- Reuse of high value items/techniques...
- Analyze multiple configurations...
- Long lead time decisions...
- Accountability...
- Design to cost...
- Weight reduction after introduction…

Criteria (intended use)



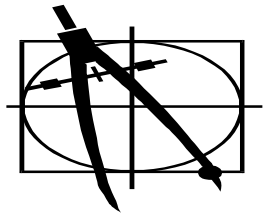
- Military combat
  - Performance overrides cost => little reuse
  - Survivability versus lethality trade-offs
- Airline
  - Safety and reliability
  - Low cost of ownership (direct and indirect)
- Software: project must define criteria

Reuse of high value items

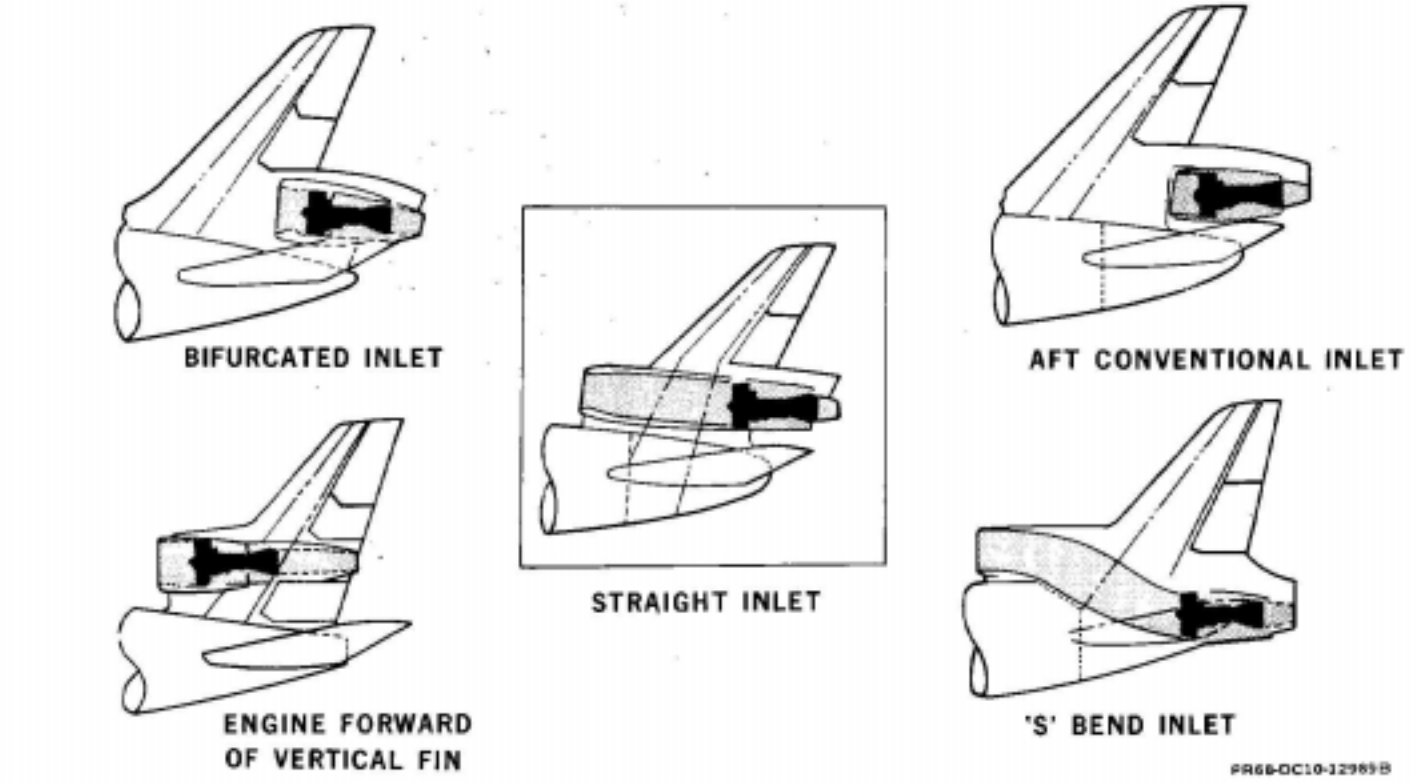
- Windscreens - require extensive testing
- Cockpits User Interfaces - B-757 and B-767
- Douglas - extrusions dating from WWII
- Software: algorithms, UI elements, code?
  - No academic experience in library searches
  - No communication vehicle - Java changing?

Analyze multiple configurations

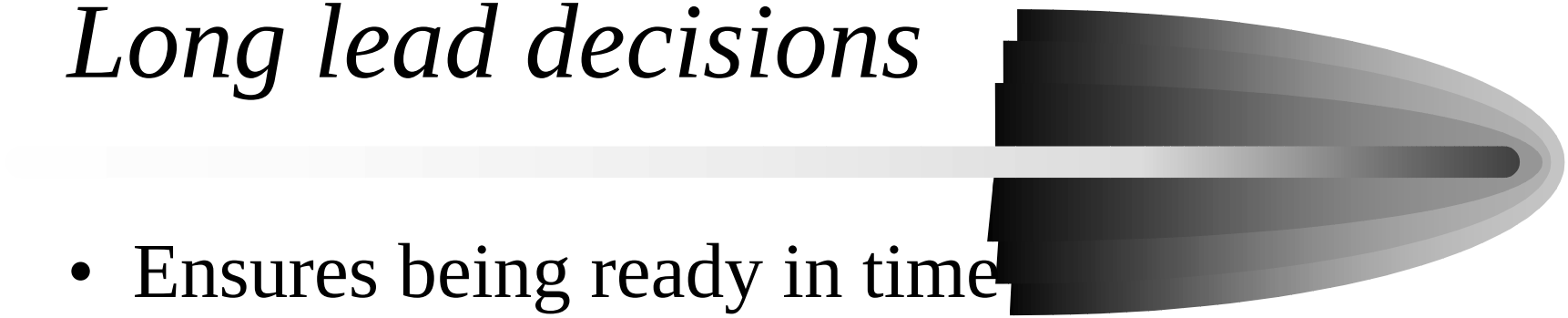
- Five different DC-10 tail arrangements...
- Wing design: perform vs fuel vs weight…
- Apply different criteria simultaneously
  - Least weight or fastest or earliest delivery
  - Marketability, risks, operating cost are metrics
- Software: soonest | fastest | smallest | robust



Multiple analysis



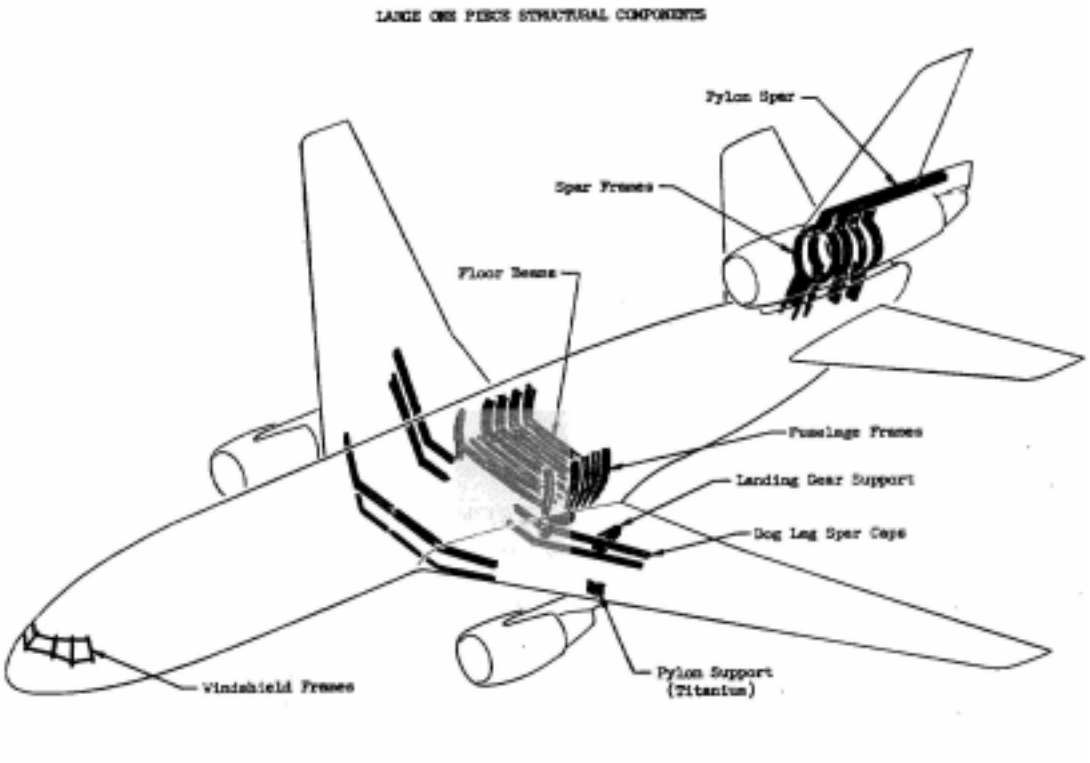
Long lead decisions



- Ensures being ready in time
  - DC-10 “banjo” forging...
  - Manuals
- Have to design around these choices
  - Landing gear (F-4)
  - Flaps (C-17)
- Software: enabling tech, large volume tasks

Long lead decisions

Major forgings requiring load lead time:

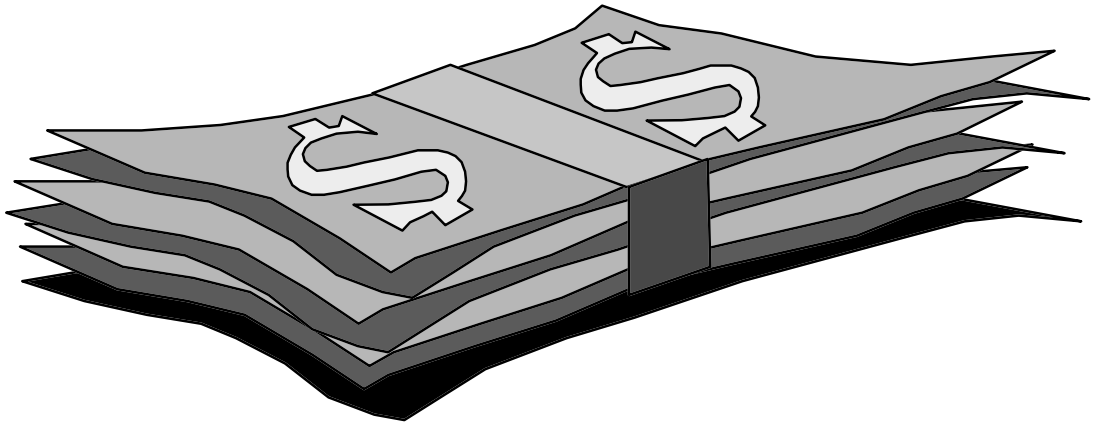


Accountability

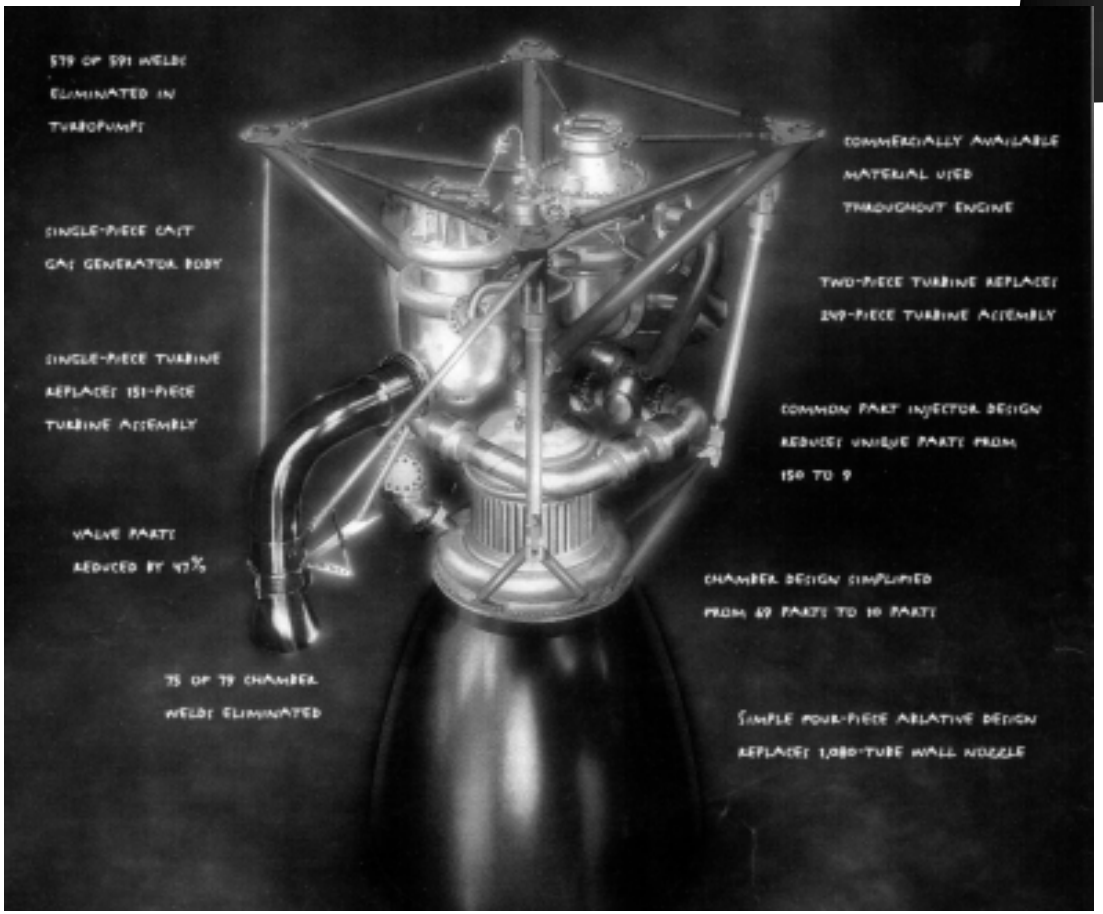
- All documents signed, checked, approved
- Material lots tracked
- Effectivity of revisions
- Root cause analysis - CF6 drive shaft
- Software: not even author’s name in source

Design to cost

- Constrains approaches allowed
- First used in military - works in commercial
- Common auto engineering technique
- Software: rationale for reuse, approaches



Design to cost



Ad points:

579 of 591 welds eliminated

Commercial materials

Single piece cast gas generator

2 pcs turbine replaces 249 pcs

Common injector: 150 to 9 pcs

Chamber: 69 to 10 parts

Turbine: 151 pcs assembly to 1

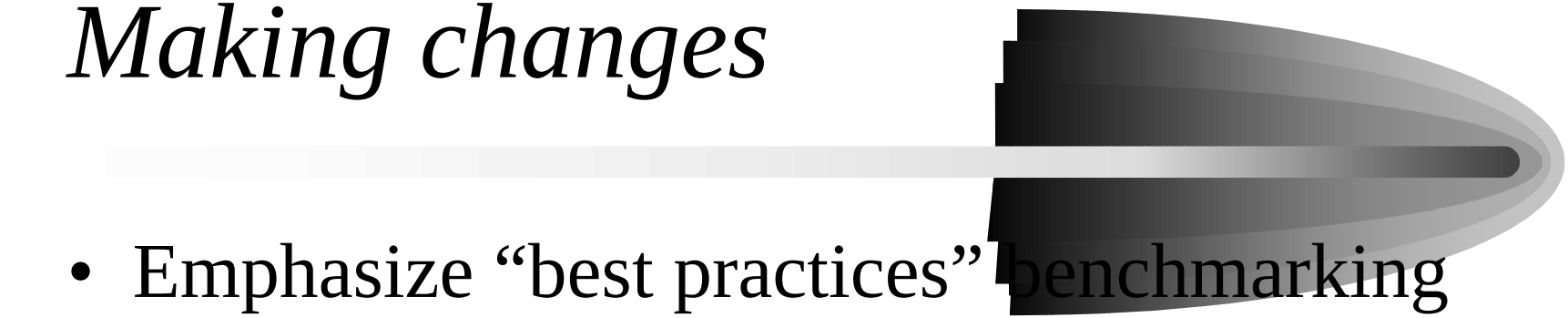
Weight reduction

- Refines initial release
- Allows less constrained development
- Doesn’t add new features
- Software: has weight? Yes!
  - E.g., similar but specialized routines, unused
  - Costs time to manipulate and research for reuse
  - More opportunities for bugs
  - Weight reduction is consolidation, not debug

Applying these lessons

- Start with one or two
  - Best payback
  - Easiest because of ??? <some local reason>
  - High visibility to gain acceptance
- Making change happen...
- Combat rebellion with techniques...

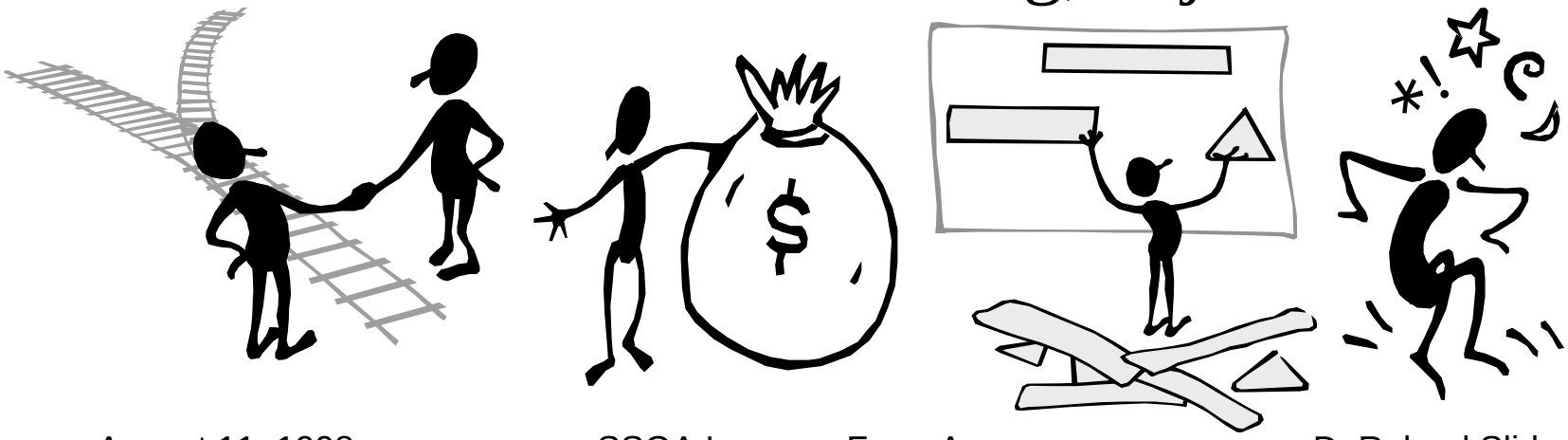
Making changes



- Emphasize “best practices” benchmarking
  - Makes change improvement, not remedial
  - Challenge to achieve same levels
- Requires management *enforcement*
  - Take responsibility for risk
  - Establish objectives to follow processes
- At least six-months project (add training)

Combat rebellion

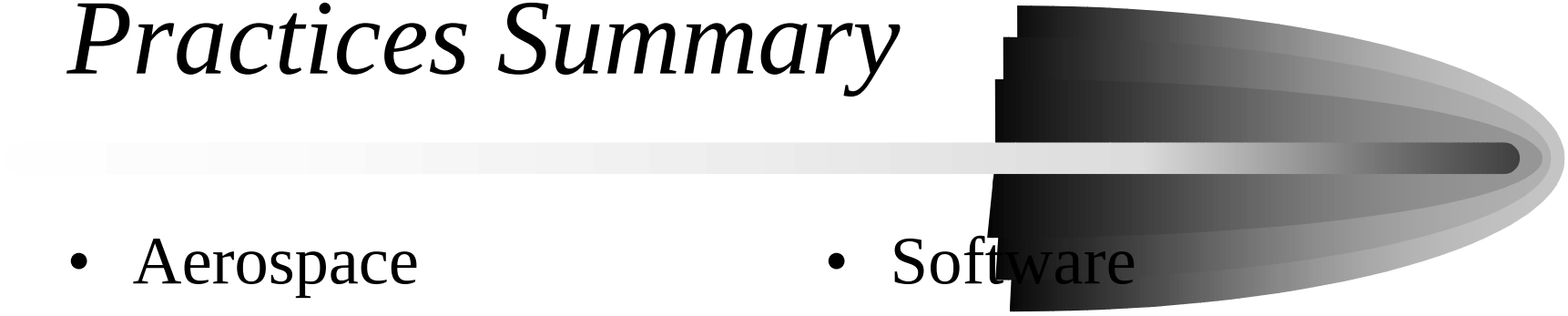
- Implicitly via self-checking processes
- Explicitly via appraisals and rewards
- Small milestones to build confidence
- Communication for venting; adjustment



Obstacles and Solutions

- No academic preparation
- Late start => large backlog of processes
- Change is challenging
- Long term payoff
- In-house training
- Pick selected, high value projects
- Rewards
- Many small victories

Practices Summary



- Aerospace
  - Engineering criteria
  - Reuse high value
  - Multiple analysis
  - Long lead items
  - Design to cost
  - Accountability
  - Weight reduction
- Software
  - Establish criteria
  - Reuse what works
  - Multiple analysis
  - Early decisions
  - Design to cost
  - Accountability
  - Weight reduction

SW takes flight...

…by following Aerospace Best Practices!