

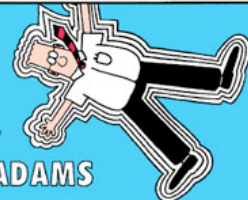
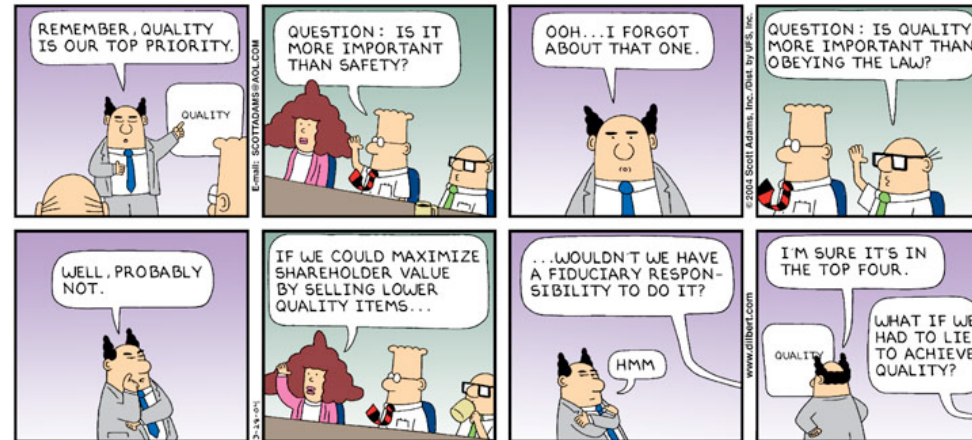
Quality plan

Alexandre Bergel
abergel@dcc.uchile.cl



DILBERT[®]

BY
SCOTT ADAMS

Sources

[Software Engineering](#), I. Sommerville, 7th Edn., 2004.

[Software Engineering](#) — A Practitioner's Approach, R. Pressman, Mc-Graw Hill, 5th Edn., 2001.

[Fundamentals of Software Engineering](#), C. Ghezzi, M. Jazayeri, D. Mandroli, Prentice-Hall 1991

Quality Control Assumption

Project Concern = Deliver on time and within budget

External (and Internal)
Product Attributes

Process Attributes

Assumptions:

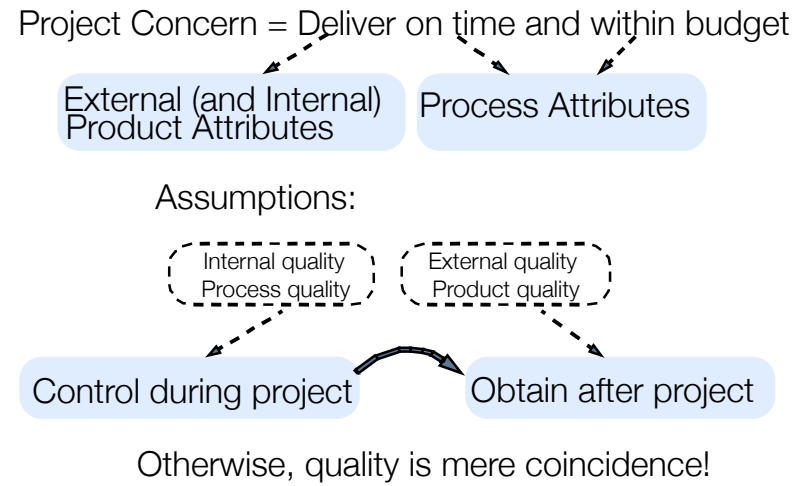
Internal quality
Process quality

External quality
Product quality

Control during project

Obtain after project

Otherwise, quality is mere coincidence!



Quality plan

A quality plan should

- set out desired product qualities and how these are assessed

- define the most significant quality attributes

define the quality assessment process

- i.e., the controls used to ensure quality

set out which organizational standards should be applied

- typically done via quality reviews after internal release

Quality Management should be [separate](#) from
project management to ensure independence

Software quality controls

Reviews

[Inspections](#) for defect removal (product)

[Progress Assessment Reviews](#) (product and process)

[Quality reviews](#) (product and standards)

Automated Software Assessment

[Measure](#) software attributes and compare to standards (e.g., defect rate, cohesion, etc.)

Types of quality reviews

A *quality review* is carried out by a group of people who carefully examine part or all of a software system and its associated documentation.

Reviews should be recorded and records maintained

Software or documents may be “signed off” at a review

Progress to the next development stage is thereby approved

Types of quality reviews

Review type	Principal purpose
Formal Technical Reviews (a.k.a. design or program inspections)	Driven by checklist - detect detailed errors in any product - mismatches between requirements and product - check whether standards have been followed.
Progress report	Driven by budgets, plans and schedules - check whether project runs according to plan - requires precise milestones - both a process and a product review

Review meetings

Review meetings should:

- typically involve 3-5 people

- require a maximum of 2 hours advance preparation

- last less than 2 hours

Review minutes

The review report should summarize:

What was reviewed

Who reviewed it?

What were the findings and conclusions?

The review should conclude whether the product is:

Accepted without modification

Provisionally accepted, subject to corrections (no follow-up review)

Rejected, subject to corrections and follow-up review

Review guidelines

1. Review the **product**, not the producer
2. Set an **agenda** and maintain it
3. **Limit debate** and rebuttal
4. **Identify problem areas**, but don't attempt to solve every problem noted
5. Take **written notes**

Review guidelines

6. Limit the number of participants and insist upon advance preparation
7. Develop a checklist for each product that is likely to be reviewed
8. Allocate resources and time schedule for reviews
9. Conduct meaningful training for all reviewers
10. Review your early reviews

Sample review checklists

Software project planning

1. Is software scope unambiguously defined and bounded?
2. Are resources adequate for scope?
3. Have risks in all important categories been defined?
4. Are tasks properly defined and sequenced?
5. Is the basis for cost estimation reasonable?
6. Have historical productivity and quality data been used?
7. Is the schedule consistent?
8. ...

Sample review checklists

Requirements Analysis

1. Is information domain analysis complete, consistent and accurate?
2. Does the data model properly reflect data objects, attributes and relationships?
3. Are all requirements traceable to system level?
4. Has prototyping been conducted for the user/customer?
5. Are requirements consistent with schedule, resources and budget?
6. ...

Sample review checklists

Design

- 1.Has modularity been achieved?
- 2.Are interfaces defined for modules and external system elements?
- 3.Are the data structures consistent with the information domain?
- 4.Are the data structures consistent with the requirements?
- 5.Has maintainability been considered?
- 6....

Sample review checklists

Code

- 1.Does the code reflect the design documentation?
- 2.Has proper use of language conventions been made?
- 3.Have coding standards been observed?
- 4.Are there incorrect or ambiguous comments?
- 5....

Sample review checklists

Testing

1. Have test resources and tools been identified and acquired?
2. Have both white and black box tests been specified?
3. Have all the independent logic paths been tested?
4. Have test cases been identified and listed with expected results?
5. Are timing and performance to be tested?

Review results

Comments made during the review should be classified

No action

No change to the software or documentation is required.

Refer for repair

Designer or programmer should correct an identified fault.

Reconsider overall design

The problem identified in the review impacts other parts of the design.



© Scott Adams, Inc./Dist. by UFS, Inc.