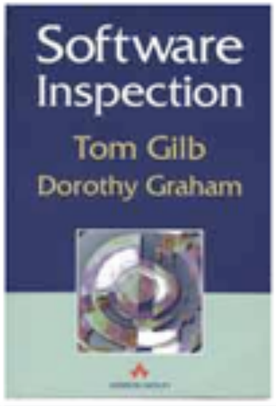


"Agile Specification Quality Control: How to do inspections on any kinds of IT Development outputs for measurement of major defects"

…based on objective and quantitative review methods.

Master

60 minute talk

Tom Gilb, and Kai Gilb

Tom@Gilb.com, kai@Gilb.com

kyoritsu-pub.co.jp

Defect Rates
in 2003 Pilot Financial Shop, London, Gilb Client
Spec QC/Extreme Inspection + Planguage Requirements

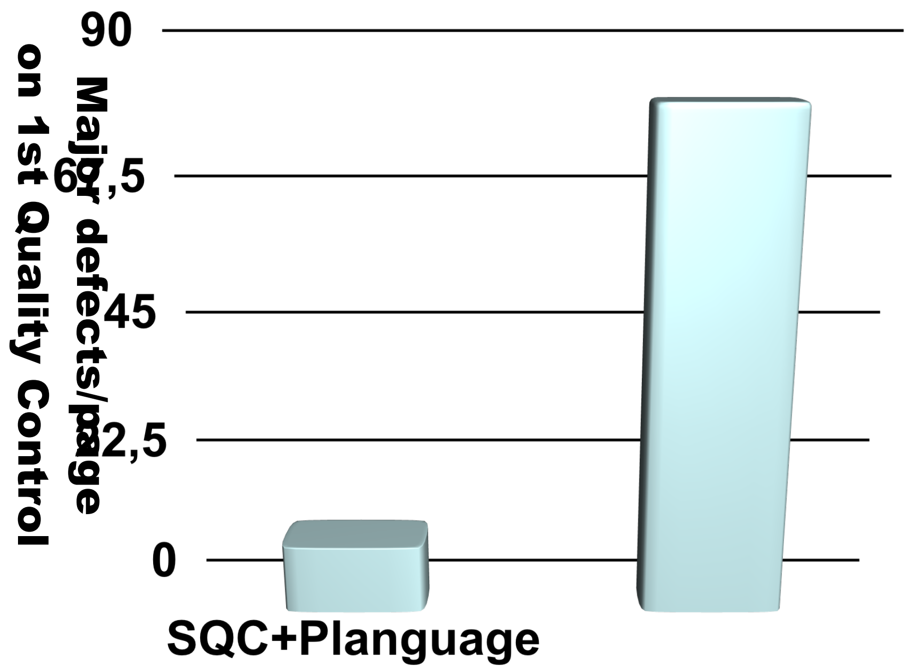
Across 18 DV (DeVelopment) Projects using the new requirements method, the average major defect rate on first inspection is 11.2.

4 of the 18 DV projects were re-inspected after failing to meet the Exit Criteria of 10 major defects per page.

A sample of 6 DV projects with requirements in the 'old' format were tested against the rules set of:

> The requirement is uniquely identifiable
> All stakeholders are identified.
> The content of the requirement is 'clear and unambiguous'
> A practical test can be applied to validate it's delivery.

The average major defect rate in this sample was 80.4.



Source Eric Simmons, erik.simmons@intel.com 25 Oct 2011
http://selab.fbk.eu/re11_download/industry/Terzakis.pdf

A Recent Example

Application of Specification Quality Control by a SW team resulted in the following defect density reduction in requirements over several months:

Rev.	# of Defects	# of Pages	Defects/ Page (DPP)	% Change in DPP
0.3	312	31	10.06	
0.5	209	44	4.75	-53%
0.6	247	60	4.12	-13%
0.7	114	33	3.45	-16%
0.8	45	38	1.18	-66%
1.0	10	45	0.22	-81%
Overall % change in DPP revision 0.3 to 1.0:				-98%

Downstream benefits:

•Scope delivered at the Alpha milestone increased 300%, released scope up 233%

•SW defects reduced by ~50%

•Defects that did occur were resolved in far less time on average

• teams typically exit with densities ranging from 5 majors per page (600 words) to 1 defect in a couple of pages.

Agile Inspection of TEST ACCEPTANCE Requirements (30 minutes, Oct 8 2009)

Rules

- 1. Clear enough to test
- 2. Unambiguous to intended readership
- 3. No Design
  - (Password -> Security)

Process

- Count probable violations
  - Forget minors (can't impact real product)
  - Use 10 minutes
- Report Majors to Tom

Page 1 & 2, Acceptance Test Basis

1. Mailhandling

We need to ensure that The system can:

Automatically create any email into a service ticket

Allow "clients and non clients" to use a mailsystem as only interface to raise requests to MACH and receive the solution via their mailbox meanwhile the workload is handled in MACH via a service tickets.

The system must be able to deal with scenarios where email address is not yet known in advance. It must be clear how we avoid spam but ensure handling of relevant mails.

We will need to see an automation af emailsending such as opening mail and solution mail.

2. Client Interface

From the client interface the structure of how to create service ticket in the new system should be visible, showing the flow for creating a serviceticket including example of choosing a product, being present with only relevant categories, and again being guided through data that needs to be included in ticket, if any. It must be shown that a connection between request and priority and KPI can be made automatically.

Client should be able to monitor(filter on parameters) and search through tickets open as well as closed and close

Client should be able to add comment to a ticket (via mail or ST system). This action must also trigger notification internally at MACH.

Client must be able to close ticket, accept a solution, decline a solution provided.

Client must be able to maintain contact information and own users.

3. Handling of input data and client contact

It must be possible to incorporate client information from all relevant MACH databases to be reused in the service ticket system so information that is already known in other MACH databases needs to be remaintained in The system.

It is possible to build flows, processes, SLA parameter on basis of input from other systems for example AXAPTA, such as client number and agreement numbers.

We must see an integration case and will suggest to use connectivity database for this.

4. Workflow

It must be visible that The system supports the processes and workflows that are used in client by having implemented a limited number of processes, that are representative for how The system will handle these.

Reports and tools must be available to monitor and control and priorize the ticket flow per person/team/management.

5. Workrequests

It must be clear that work requests can support the ways of working in client support in respect to manage and control workload, sort the priority of tickets, escalate, enable follow up routines and KPI, and ensure clear responsibilities for the task.

6. Performance

The system must perform similar well across all locations to allow CS agents to work efficient and with acceptaple response times and the system must scale to the ticket volume and numbers of users.

7. Security

It must be clear that the structure of The system allows control of who can have access to which information.

It must be possible to limit access to certain type of tickets based on clients/product etc to a smaller group of persons/teams.

It is clear that The system support the requirements to be certified by ISO as well as SAS70

It must be possible to control at client interface who by the client have access to which tickets related to a certain product/service or business unit.

8. SLA handling

It must be possible to specify a SLA for a client and ensure that tickets created can be handled in accordance with the SLA by different teams and that it is possible to extract statistical information regarding ticket handling into the SLA.

It is possible to verify that a ticket related to a product/service can be handled in accordance with a SLA defined for this producttype.

It is possible to establish measurements on time used, build related reports and set up warnings/count down functionality in accordance with timeframes.

Evaluating the Requirement

Raw Major defect count

- Pages are about 300 words (327, 287)
- 23 30? 24 23 35
- Other table
  - 6 5 14 14 10 12
- Dennis: 65, words on page 2
- There were 30 people in the class, 15 took page 1, 15 took page 2

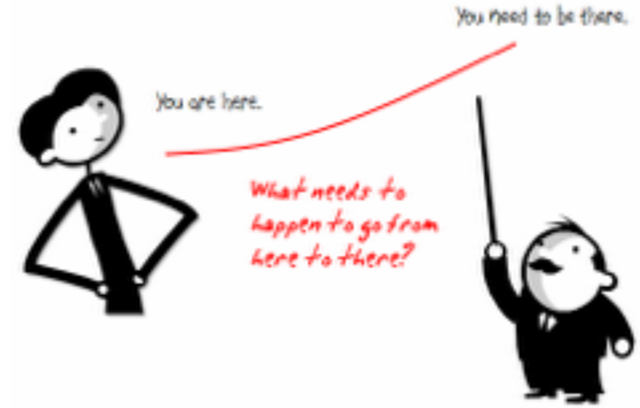
Extrapolation: real defect density.

- If Exit was Max 1.0 major
  - .
  - Estimate majors found by Team = 2 x 35 ±10 = 70
  - Inspection effectiveness is about 1/3rd   33%
  - There are about 3 x 70 **= 210 Majors** on these pages on average.
  - The total requirements has about **450 majors** in the one requirement doc (701 word)

Real Case of Agile SQC from London Sept 3 09

- How good are you at finding critical defects in requirements?

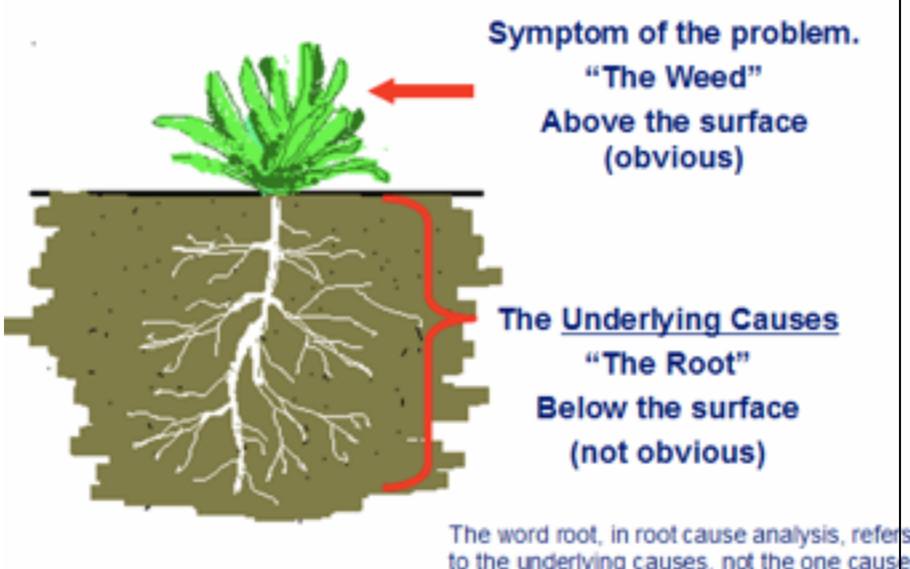
WHY are we doing this? Part of Platform Rationalisation Initiative, with below **<u>Main Objectives.</u>**



- **Rationalize into a smaller number of core processing platforms. This cuts technology spend on duplicate platforms, and creates the opportunity for operational saves. Expected 60%-80% reduction in processing cost to Fixed** Income **Business levies.**
- **International Securities on one platform, Fixed Income and Equities (Institutional and PB).**
- **Global Processing consistency with single Operations In-Tray and associated workflow.**
- **Consistent financial processing on one Accounting engine, feeding a single sub-ledger across products.**
- **First step towards evolution of "Big Ideas" for Securities.**
- ***<u>Improved development environment</u>*, leading to increased capacity to enhance functionality in future.**
- **Removes duplicative spend on two back office platforms in support of mandatory message changes, etc.**

Rules are needed

- To define *specification* defects



- **Main Objectives Defects** ***(root causes)*** lead to *potential defects* in *the next stages*
  - Architecture
  - Design
  - Testing
  - Construction
- Any of which can result in FAULTS in the final system
- Faults can result in breakdown of the real product.

QC Rules for Top Level Objectives

- **CLEAR: Every word and phrase should be clear enough to allow objective test of a delivery.** (we need to know exactly what is required and expected)
- **UNAMBIGUOUS: Every word and phrase should be unambiguous to all potential intended readers.** (no different than intended interpretations should be possible)
- **QUANTIFIED QUALITY**: all qualities (good things we want to improve) shall be expressed quantitatively.

- After we started the exercise I regretted not adding the usual rule:
- 4. **NO DESIGN: objectives shall not be expressed in terms of a design or architecture**
  - (a 'means' to reach the 'real' objective), when it is possible and is our real intent, to express the improvements in terms of quality, performance, and cost that are expected, instead.

Potential consequence of major defects in architecture specs

COUNT MAJOR 'DEFECTS' (RULES VIOLATIONS)
Rules Reminder:
1. Clear, 2. Unambiguous, 3. Quantified Qualities,
4. No Design/Architecture

- **"Rationalize into a smaller number of core processing platforms. This cuts technology spend on duplicate platforms, and creates the opportunity for operational saves. Expected 60%-80% reduction in processing cost to Fixed Income Business lines.**
- **International Securities on one platform, Fixed Income and Equities (Institutional and PB).**
- **Global Processing consistency with single Operations In-Tray and associated workflow.**
- **Consistent financial processing on one Accounting engine, feeding a single sub-ledger across products.**
- **First step towards evolution of "Big Ideas" for Securities.**
- ***Improved development environment*, leading to increased capacity to enhance functionality in future.**
- **Removes duplicative spend on two back office platforms in support of mandatory message changes, etc."**

<u>LINK WORDS</u>:
OBJECTIVE:ARCHITECTURE
RULE 4. No Design/Architecture

er number of core processing platforms.
end* on duplicate platforms, and *creates the
onal saves. Expected 60%-80% reduction in
l Income Business lines.

on one platform, Fixed Income and Equities

tency <u>with</u> single Operations In-Tray and

cessing on one Accounting engine, <u>feeding</u> a
s products.

tion of "Big Ideas" for Securities.

***vironment*, <u>leading to</u> increased capacity to**
n future.

nd on two back office platforms <u>in support</u>

Agile Spec QC Results

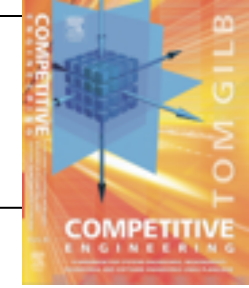
- **Reported** major defects =
- Last week: 15, 17, 21
- **Today** =18, 15, 15, 13   others less

- Estimated appx. Total defects found by a small team (2-4 people) = 36±6
  - 2x highest found.
- Estimated appx. Total Majors in the 110 words = 100±10
  - (3x group total. 30% effectiveness of team)
- Estimated approximate total defects in normalized page (300 words) = 280±20

Advanced Questions

- High Quality Level: "Maximum Majors for Exit from process" = **1.0 majors** remaining max.
- If all *found* majors removed, how many majors *remaining* per Page? =
- Predicted Bugs resulting if released now (for each such page in requirements),
- Penalty for Majors at this level (Main Objectives) = PROJECT FAILURE

How can we improve such bad specification? (‘Planguage’)



<u>Development Capacity</u>:

Version: 3 Sept 2009 16:26

Type: Main <Complex/Elementary> Objective for a project.

Ambition Level: radically increase the capacity for developers to do defined tasks. <- Tsg

Scale: the Calendar Time for defined [Developers] to Successfully carry out defined [Tasks].

Owner: Tim Fxxx

C**<u>alendar Time</u>**: defined as: full working days within the start to delivery time frame.

Past [2009, {Bxx, Lxx, Gxx}, If QA Approved Processes used, Developer = Architect, Task = Draft Architecture] **15 days** ±4 ?? <- Rob

Goal[2011, { Bxx, Lxx, Gxx }, If QA Approved Processes used, Developer = Architect, Task = Draft Architecture] **1.5 days** ± 0.4 ?? <- Rob

Justification: Really good architects are very scarce so we need to optimize their use.

Risks: we use effort that should be directed to really high volume or even more critical areas (like Main Objective).

Participant Feedback

- **Management Conclusion:**
  - **The defect density is completely unacceptable in the 'Main Objectives' section**
  - **They wondered how to improve it (see example below)**
  - **They emailed me afterward:**
  - *"Thanks for your time today Tom, very useful talking to you and perfect timing for the stage we're at in our reengineering program. There are some concepts I definitely want to take forward and will spend some time over the next few days discussing this with Pxx and Pxx , but may then get some more of your time to think through how we take things forward.*
  -
  - *Once again, thanks for your time, Kxx "*

Case:
Real Inspection

- of System Requirements Specification (SRS) of 82 pages for a major US corporation.

This presentation

shows

how we carried out a short

specification quality control

process

with senior/middle managers.

The purpose is to
make managers aware
that they play a key-role
in creating projects
delays
by approving poor
quality of requirements
specifications.

The results shown in this real-life example successfully predicted a project delay of at least 2 calendar years.

Poor quality marketing requirements documents prove time and again to be

a good predictor of project delays.

- The clue is that
- requirements documents
  - with a high defect density
  - are an indicator of
  - a truly *unprofessional* engineering culture.

Framework

- Demonstration of power of Inspection
  - 8 Managers
  - 2 hours
  - 4 real requirements specifications offered ,
    - 1 used

We Introduced best practice Rules for Requirements

- 1. **Unambiguous** to intended Readership
- 2. **Clear** enough to test.
- 3. **No** unintentional **Design**

We Explained the definition of <u>Defect</u>

- A **Specification Defect** is a *violation of a Specifciation Rule (a 'standard')*
  - Note: If there are 10 ambiguous terms in a *single* requirement
    - then there are 10 defects!

Explain the definition of Major defect

- **Major:**
  - a Defect that *potentially*

costs more

- to find and fix
- **later** in the development process
- than it would cost *now*.
- We need to get rid of it NOW!

Agree with Management on Exit level

Is 1,000 Majors per page OK

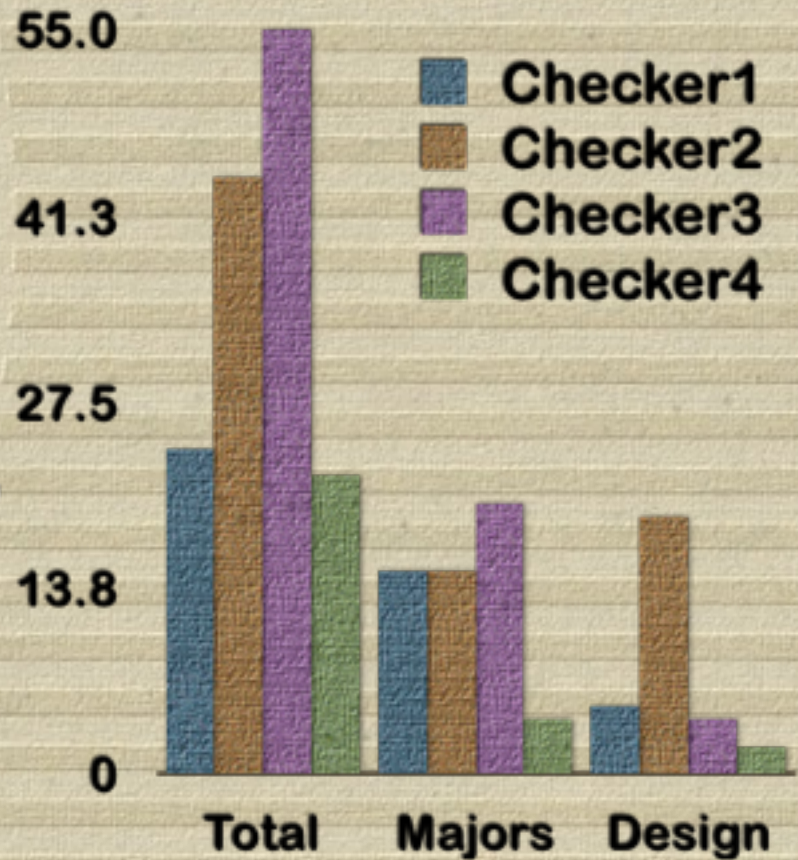
100, 10, 1

- **Exit Conditions:** (when Requirements can go forward to Design, Test etc with little risk)
  - **Maximum 1 Major Defect/ (Logical) Page**
  - **Logical Page = 300** Non commentary words.

the Job

- You have up to 30 minutes
  - check 1 sample requirements page (from an 82 page document)
- Count all *potential* Rule Violations
  - = Defects
- *Classify* Defects as Major or minor

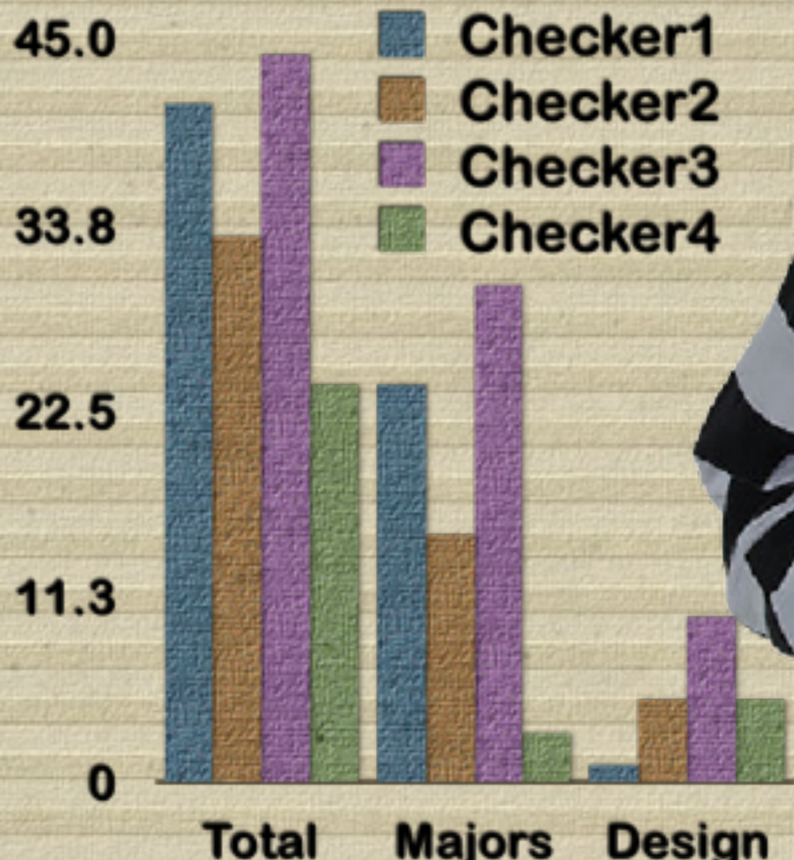
Report
Page 81

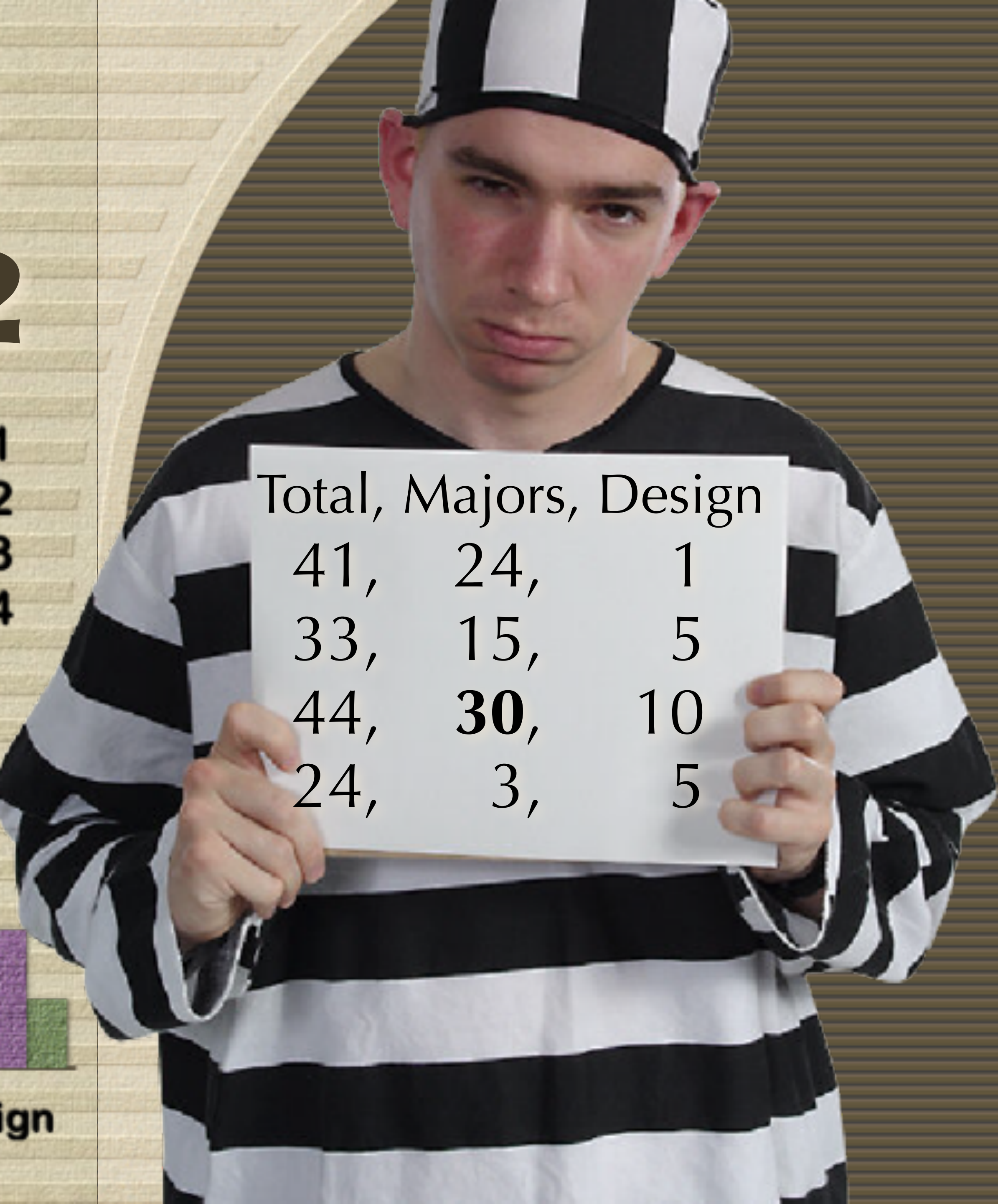


Defect Density Estimation

- Total for group (page 81)
  - 20 x 2 = 40 Majors
  - assume 40 are unique
- If 33.333% effective,
  - total in page = 3x 40= **120**
- Of which 2/3 or 80 were *not yet found.* .
- If we fix all we found (40),
  - then the estimated remainder of Majors would be 80 (not found)
  - +8 “not fixed for correctly”
  - = **88** Majors remaining.

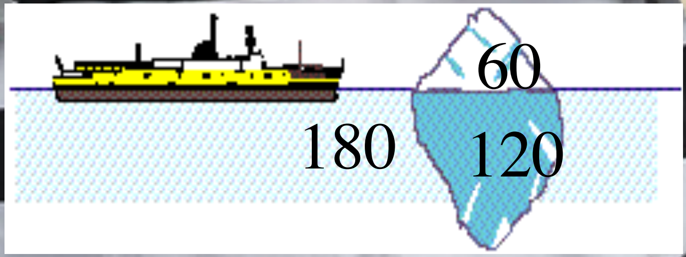
Report
Page 82





Defect Density Estimation

Total,	Majors,	Design
41,	24,	1
33,	15,	5
44,	**30**,	10
24,	3,	5



- Total for group (page 82)
  - 30 x 2 = 60 Majors
  - assume are unique.
- If 33.333% effective,
  - total in page = 3x 60 = **180**
- Of which 2/3 or 120 were not yet found.
  - If we fix all we found (60),
  - then the estimated remainder of

Majors would be 120 (not found)

- +10 “not fixed correctly”
- = **130** Majors remaining.

Conclusions

- Human *defect removal* by Inspections/reviews/SQC is
  - a hopeless cause: not worth it.
- Spec QC can be used, in spite of imperfect effectiveness,
  - to *accurately estimate* major defect level density.
- This measurement can be used to motivate engineers to
  - dramatically (100x! Over about 7 learning cycles)
    - reduce their defect insertion (rule violation)
  - to a *practical* exit level
    - (like *less than* 1.0 Majors/page)

Extrapolation to Whole Document

- **Average: 150 Majors/page**
  - Page 81: 120 majors/page
  - Page 82: 180 Majors/page
- **Total in whole document:**
  - **12,300 Majors**
    - 150 Majors/page x 82 pages.

Chapter 1

Hal Spacejock was sitting at the Black Gull's flight console, his attention riveted to a small chessboard balanced amongst the toggle switches, flashing lights and status displays. Recently he'd skimmed an article extolling the benefits of the ancient game: how playing it would sharpen his mind, improve his memory and increase his attraction to the opposite sex. Chess had been an important part of his daily routine ever since, but after two hundred and seventy-six losses in a row Hal was beginning to doubt the article's claims. He didn't feel any smarter and he couldn't remember the last time he'd spoken to a member of the opposite sex, let alone attracted one. Briefly, he wondered whether it was such a clever idea to play against the Navcom, the Black Gull's onboard computer. Underpowered and outdated, it was still more than capable of running the ship's accounts, navigation and life support systems while beating humans at simple board games. However, since Hal was the only human aboard the Black Gull, his choice of opponents was limited.

'Your turn,' said the Navcom, in a neutral female voice.

'I'm thinking.'

'While you're planning your opening move, can I tell you about a special offer?'

'What kind of offer?' asked Hal suspiciously.

'Planet Books have a chess title on sale.'

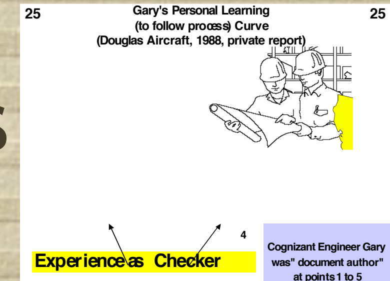
'Really? Put it on-main.'

2

Estimated Project Loss

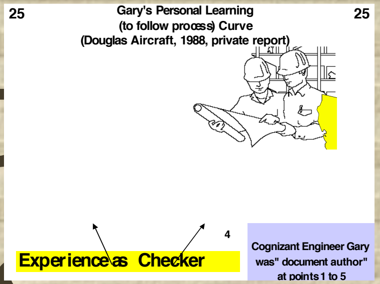
- If a Major has
  - 1/3 chance of causing loss
- And each loss caused by a Major is
  - avg. 10 hours
- then total project Rework cost is
- about ***41,000 hours loss.***
- (This project was over a year late)
- 1 year = 2,000 hours x 10 people

The DAC/Boeing Quantified Results

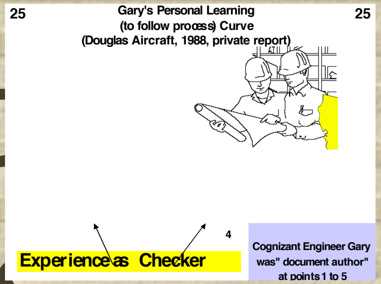


- Dale Warren was the Director of Design Engineering at that time.
- So we had to convince him that this new systems engineering process paid off.
- Before we began, DAC had done a study showing that each EO defect, causing rework to the EO/Drawing cost an average of about $2,965 (corresponding Boeing numbers in 1989 were $3,000 to $5,000, I learned).
- Our Inspection 'Major Defects' concept was removing things that threatened that time and money loss.
- The engineering staff at DAC worked up a formula for the Return on Investment for Inspection which considered this, and one day about Summer 1988, we presented it to Dale Warren
- . I sat there as he studied the result, which was approximately an ROI of 4.5 to 1. He more than accepted his staffs calculations.
- His words to me were.
  - This is conservative because it only takes engineering labor into account.
  - It would be much larger if we took materials wasted into account.
- Later Dale Warren walked into a meeting of 'our Inspection Gang' one evening and told me that I would soon be hearing from Boeing.
  - He had bragged about the results to them at a conference and given them my card.
  - The result of that was that we did a major trail of the method at Boeing
    - (130 people trained, a 12 week period, about 25 parallel projects in all engineering areas.
- Conclusion, that Inspection was indeed the most powerful method for checking engineering specs they had seen.
- Now 2012 called the PEP, Process Error Prevention Method <- Lawrence Day Boeing

The Quant
Results



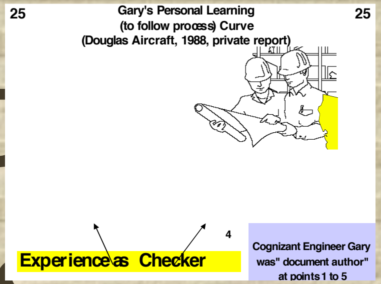
The Quant
Results



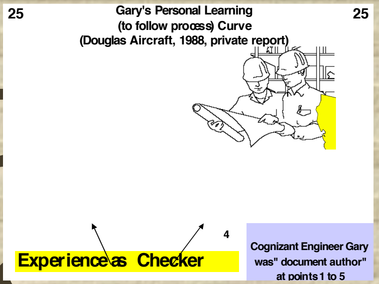
loss.

- The engineering staff at DAC worked up a formula for the Return on Investment for Inspection which considered this, and one day about Summer 1988, we presented it to Dale Warren
- . I sat there as he studied the result, which was approximately an ROI of 4.5 to 1. He more than accepted his staffs calculations.
- His words to me were.
  - This is conservative because it only takes engineering labor into account.
  - It would be much larger if we took materials wasted into account.
- Later Dale Warren walked into a meeting of ‘our Inspection Gang’ one evening and told me that I would soon be hearing from Boeing.
  - He had bragged about the results to them at a conference and given them my card.
  - The result of that was that we did a major trail of the method at Boeing
    - (130 people trained, a 12 week period, about 25 parallel projects in all engineering areas.
- Conclusion, that Inspection was indeed the most powerful method for checking engineering specs they had seen.

The Quant
Results



The Quant
Results



- Conclusion, that inspection was indeed the most powerful method for checking engineering specs they had seen.

Feedback on this "simple "formula

Tom Since returning from the QAI Conference in Orlando, I've been attempting to lay the foundation for our product team to develop clear requirements and implement productive inspections as opposed to just going through empty motions. It's definitely been an uphill effort.

One bright moment was my use of the formula that you provided me to

estimate the # of high-severity bugs still in a software product.

I applied it to our product's Test Pass 1 and then forwarded the estimated

number of remaining bugs after Test Pass 1 to the count estimated to

still be in the product when we began Test Pass 2.

This provided me with

a prediction of the number of high-severity bugs that would be found which was within 5% of the number actually found during Test Pass 2. :-)

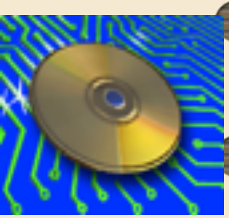
I can't tell you how much that relatively simple activity buoyed my spirits. Thank you for the time you spent with me in Orlando.

Thanks, Jeff Finn, CSTE, CQA, Microsoft SharePoint Portal Server, 425-703-4213

jfinn@exchange.microsoft.com, May 22 2001

More feedback: Intel

- We are using this with requirements documents, and have been able to double the quality of the documents with only a few hours of effort.



- " Erik Simmons, Intel, Oregon " "erik.simmons@intel.com
- January 9th 2002

Agile Inspection activities in Japan 2009

Atsushi Nagata, Sony Japan

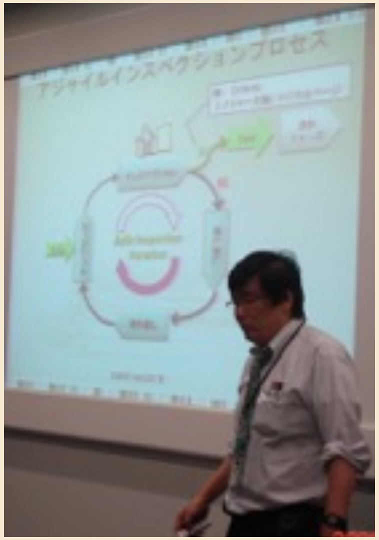
- Overview of Activities
  - Workshops
    - Promotion
  - Community Gatherings
    - Check Proper Comprehension
    - Collect Case Studies
    - Issue Guidelines
  - Deployment
    - Assist in Actual Implementation

Workshops

- Objectives
  - Introduction of Agile Inspection
    - Proper Image
    - Principles
  - Experience
    - Short exercise of Agile Inspection

      Participants feel finding many defects at one page than they expected in short time. Good motivation.
  - Measurement
    - In order to improve estimation of

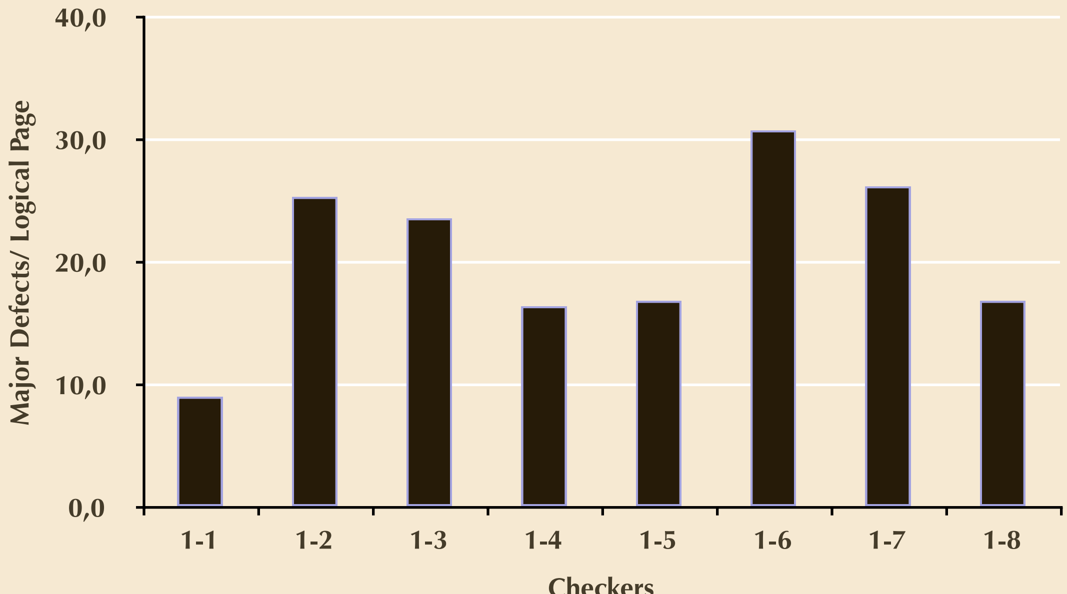
    Defect Density per logical page. Key Metric.
- Workshop were held 7 times.
- Total participants 114



A 2-hour small group session f
hands-on first time experience
agile inspections

Workshop : result example

8	Number of Checkers
61.6	Number of Unique Majors found per 300 words
33%	effectiveness
187	Actual Major per 300 words
15	Pages (300 words) in document
2800	Total Majors in Document
933	1/3 actually occur/hit/go-wrong
9	Average Cost in HOURS per Major if let through to the next stage
8400	Estimated delay in WORK HOURS caused by Majors



Community Gatherings

- Members 10
- Meeting once a month
- Check Proper Comprehension
  - 10 Principles
    - Some guidelines for engineering your engineering review processes for maximum efficiency : 2008
- **Issue Guidelines**
    - Method and Training
      - Agile Inspection Leader
      - Writer : How to improve the document.
      - Inspector: How to give good advice to writer.
    - Process

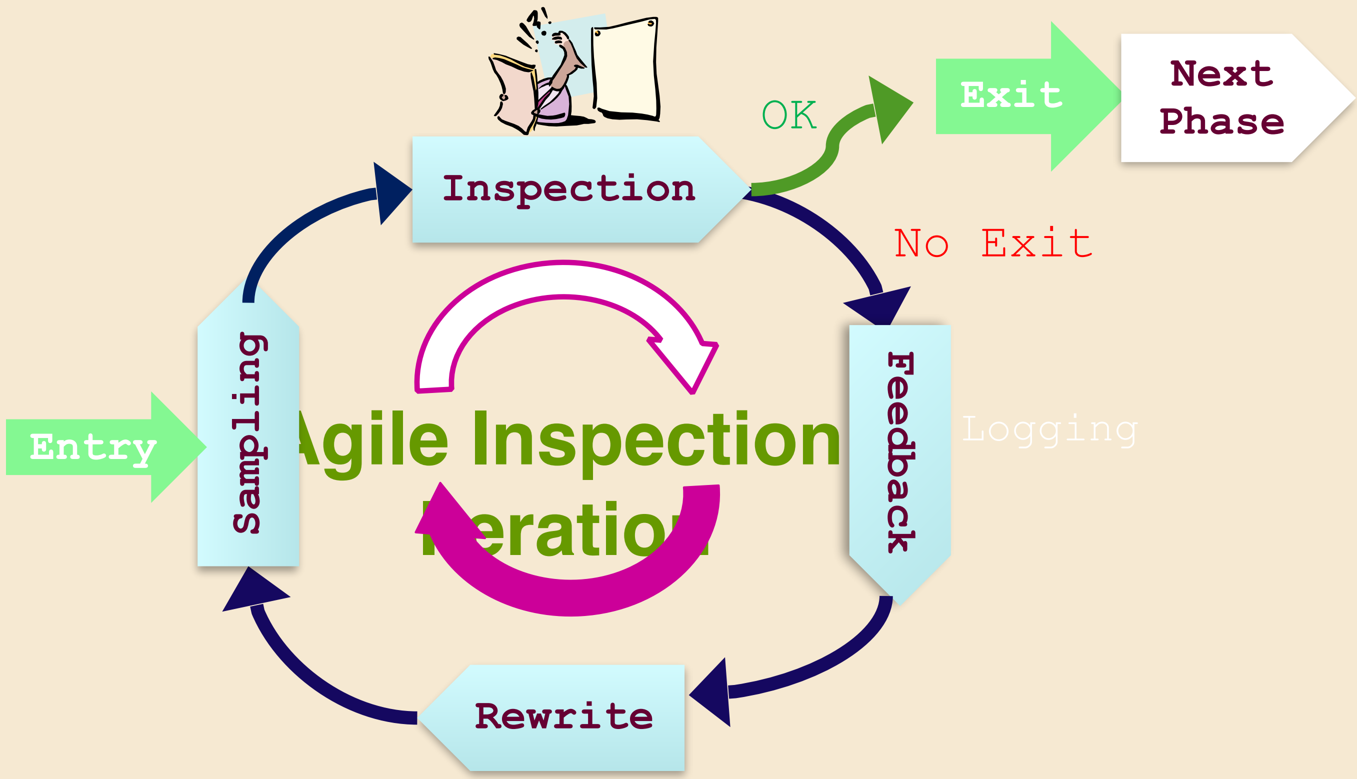
  Agile Inspection is a part of software development processes.
      - Implementation motivation
        - How to get the motivation to implement the process.
      - Improvement motivation
        - Agile Inspection process should be improved day by day

These are our key principles.

- 1. The Variation Principle:
- 2. The Efficiency Principle:
- 3. The Payoff Principle:
- 4. The Many-Purpose Principle:
- 5. The Manifold Principle:
- 6. The Prevent - don't Clean Principle:
- 7. The Teaching Principle:
- 8. The Stitch In Time Principle:
- 9. The Entry-Exit Principle:.
- 10. The Clean to be Mean Principle:

Deployment

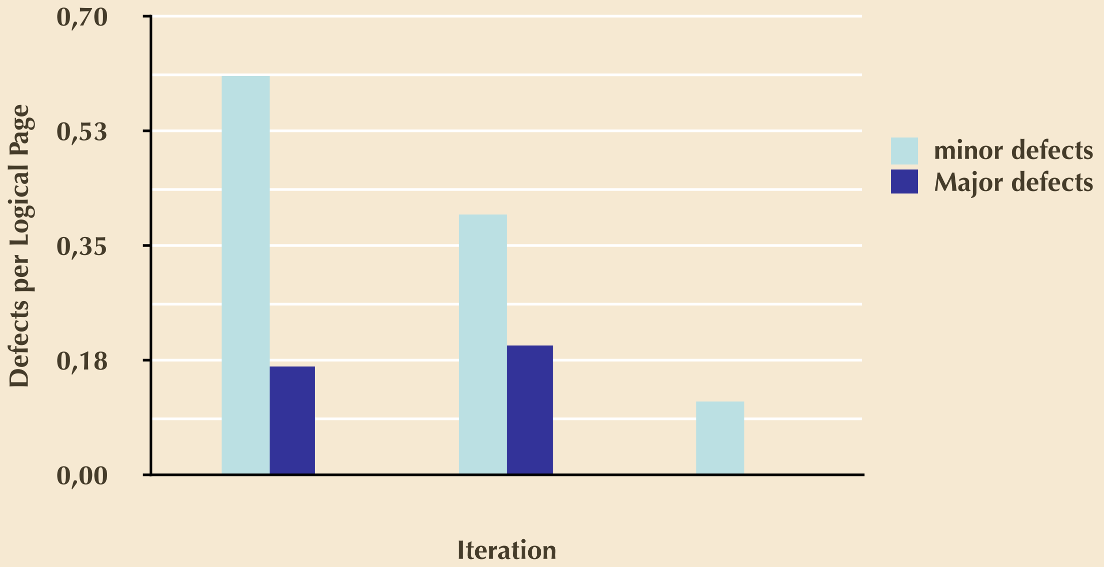
Agile Inspection Process



Collect Case Studies

User's manual inspection case

Iteration		1	2	3
Checking Rate (Hrs./Logical Page)		0.57	0.46	0.42
Defects per Logical Page	minor defects	0.61	0.40	0.11
	Major defects	0.17	0.20	0.00
Exit/Not Exit		No	No	Exit



Experience First Year

- We do not actual agile inspection process in the field, but we have some plans in this year.
- So now we could not show the case studies.
- Right now I've improved the workshop procedure to get more useful data by logging and profiling.
- I will analyze the data and write a report this year. <-Atsushi Nagata email 30.9.2009

Details of a Real Process Definition for Agile Inspection

- We do not expect to lecture with these slides. They are background information.

Authors: Tom Gilb Tom@Gilb.com & Kai Gilb Kai@Gilb.com

Intended Purpose:

Extreme Inspection <client> Variation:

a simple but powerful version of inspection (Specification Quality Control – SQC) that <CLIENT> can install immediately at low cost.

Rules

- ***The primary Rules we check against are the same Rules that writers will use when writing specifications.***
- ***Initially they will be Clarity, Unambiguousness, Consistency, Traceability, separation of requirements and solutions, and separation of Performance, Functions and Designs.***
- ***See separate document: "Rules for Specification Writers."***

Extreme Inspection Outcome

- **The outcome of this type of inspection is to give a fair measure of Major defect density.**

Intent of Outcome

- The intents of the Major defect density measure are:
- Clean: to make sure that polluted specifications do not enter the next working processes.
- Learn: to motivate specification writers to learn and follow <CLIENT> best practice specification rules.

Internal Extreme Inspection Goals

- ***"The expected effects of rigorously carrying out this process are:"***
- **Density:**

Scale: Estimated remaining Major defect density per logical page (300 Non Commentary words)

Past [December 2002] 50-100 Majors/Page <- Multiple sample inspections

Goal [Jan 2003] less than 10 Majors/Page

Goal [Jan 2004 or sooner if feasible!] less than 1 Major/ Page

External Extreme Inspection Goals

Project Efficiency

Scale: Total project time to successfully complete a project

Past [Dec 2002] ???

Goal [Dec 2003] = 70% of Past [Dec 2002]

Goal [Dec 2004] = 50% of Past [Dec 2002]

Comment:

This will be accomplished by

less back and forth,

and reviewing of requirement documents,

and by shorted coding and test times,

and by less effort when work is contracted out of country or to sub-suppliers.

More time at the requirement stage is expected.

Process Management of Extreme Inspection: 1

– ***1. Inspection Outcome Justification***

- The outcome of this variation on conventional Inspection processes is **to determine 'specification exit**' by measuring and estimating Major defect density. The outcome is NOT (as with conventional inspection) to 'clean up' bad work.
- The result of this outcome limitation is that many of the time honored conventions of Inspections (as in Gilb & Graham: Software Inspection) are NOT necessary or desirable. We only need to do whatever gives a **reasonable measure of defect density**. We only need to focus on determining that the specification is exit-able or NOT.
  - So we do not need to get maximum effectiveness by having a large team or by using one hour per page or by looking at all pages (we can sample in 10-40 minutes and use one or 2 people).
- In simple terms **if we find (checker detects) one or more Majors** in a page, it is NOT exit-able, because the real estimated quantity of majors actually there, exceeds the Exit limit of 'one per page'. If we find less than one major defect on 4 pages, it probably is *economic* to exit the spec.
- ***Economic*** is the key word. We are trying to determine **if it pays off** to exit now, or to rewrite the spec to a cleaner level now.

2. Inspection Cost Charging.

- All costs for the writer, the checker and a possible process guide, will be
  - charged to the project the writer is working on,
  - and to the QC process costs specifically.
  - *Rationale: so we can track the true costs of doing this and the degree to which it is done*.

3. Auditing this process:

- The Inspection (Spec QC) process must be regularly (monthly) audited
  - to make sure it is really conducted according to intent
  - and is not corrupted or misunderstood.
- This includes double checks on audits
  - to see if the conclusions of the check and the audit are reasonably consistent.
- Frequent audits are necessary in the beginning and with newcomers.
- Auditing will be done by the process owners.

Process Management of Extreme Inspection

- ***4. Process Improvement***
  - The process needs to be continuously updated
    - mainly in the tools kit which defines and supports the inspection process:
    - the checklists,
    - the process definitions,
    - the computer data collection support
    - by the official process owner.

5. Process Ownership

- There must be an official process owner to champion (and to manage 'local' champions),
  - spread,
  - audit,
  - and improve the process,
  - as experience and insight dictates.
- This can be a group.

6. Process Sponsorship

- The executive sponsor of this process should be official and visible

7. Confidentiality

- The checker shall *never* reveal the numeric result of an Inspection to anyone else except the writer.
  - The writer may reveal the results if they want to, but they are not obliged to do so even to their direct manager (who should not even ask!).
  - The results of an inspection, as recorded in the Specification Quality Control Database, are never to be released, revealed or reported with the name of the writer or information (such as document ID) that can lead to their identification.
- *Rationale:*
  - *to prevent fear of defamation leading to false reporting of results.*
  - *To emphasize that the process is there to help the writer reach the corporate quality level required.*
  - *It is not in any way of time to be used for personal job performance evaluation.*
  - *Evaluation should be based on EXITED specifications, and their timeliness only.*
  - *Managers need to be informed and reminded of this cultural paradigm by the process owners.*

Process Management of Extreme Inspection: 3

- ***8. Expected Effectiveness***
- We expect that the Major defect finding effectiveness of the checking process will be in the range of 10% to 35% of the actual real Majors present in a specification.
- This is quite sufficient to *estimate* the actual total number of majors actually present.
- We can then estimate with *sufficient accuracy* (say ±20%) determine levels of Majors in entire spec and in spec after correction of listed (by checkers) defects.

Defect Rates (repeat of earlier slide intentional)
Here is what really happened afterwards
in 2003 Pilot Financial Shop, London, Gilb Client
Spec QC/Extreme Inspection + Planguage Requirements

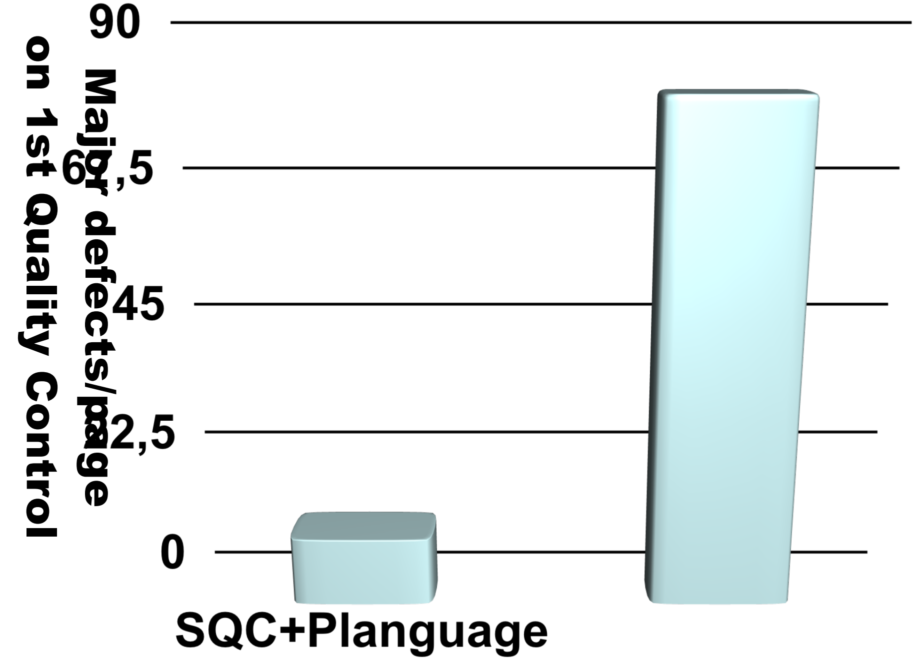
Across 18 DV (DeVelopment) Projects using the new requirements method, the average major defect rate on first inspection is 11.2.

4 of the 18 DV projects were re-inspected after failing to meet the Exit Criteria of 10 major defects per page.

A sample of 6 DV projects with requirements in the ‘old’ format were tested against the rules set of:

- The requirement is uniquely identifiable
- All stakeholders are identified.
- The content of the requirement is ‘clear and unambiguous’
- A practical test can be applied to validate it’s delivery.

The average major defect rate in this sample was 80.4.



9. True Measure of Inspection Progress.

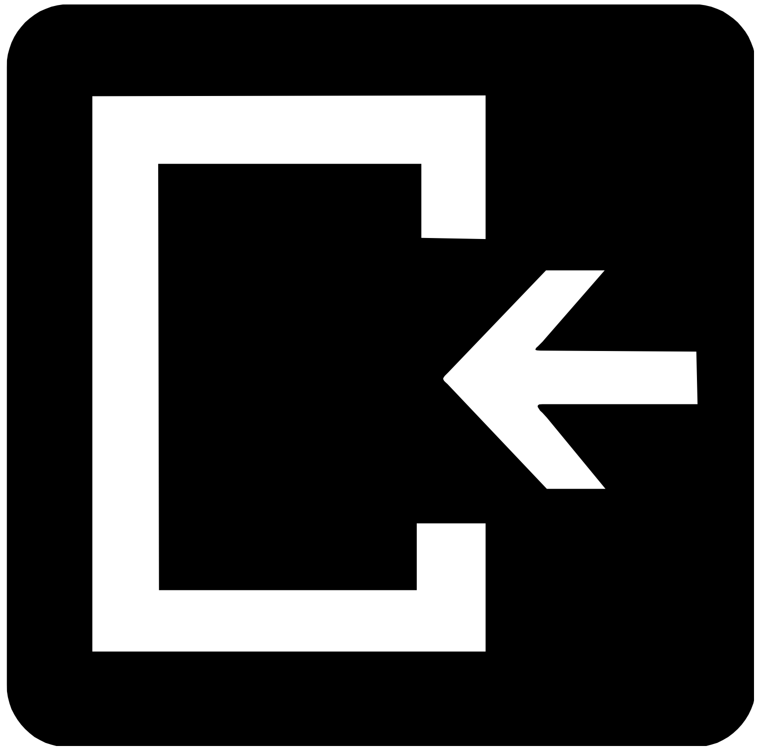
- The correct and relevant measure of how effective the Inspection process is working, is NOT as many would assume the quantity of Major defects found and fixed by an Inspection.
  - In fact we strongly recommend that this measure is well hidden from public view! (It has its uses!).
- The true measure is the average level of Major defects/Page which we can consistently release.
  - We need to move from about 100 Majors/Page down towards about less than one per page.
  - This cannot be achieved by finding and fixing defects (because we cannot find a large percentage at all)!
  - It can only be achieved in practice by *motivating* writers to reduce defects actually injected in their work, from 100, and move them down towards one maximum injected/page.
  - This is the ‘individual defect injection learning rate’.
  - Individuals seem capable of reducing their own defect injection by about half (50% fewer for each cycle of learning (write, inspect and rewrite with 50% less cycle).
- The measure of real progress is the released defect density, and it is this measure which will most closely correlate with later statistics on quality and productivity of projects.

The EI Process (Extreme Inspection):

Version:April 21, 2015, Owner: Tom@Gilb.com

- This is the formal process definition
- You should be able to print it all on a single page

El Entry Conditions



EI.E1:

- At least one of the participants
  - has done a well conducted successful inspection once before,
  - or been briefed by a competent practitioner,
  - or will be guided through the process by a competent guide (ideally an expert in this process).
- *Rationale: people need to have some reasonable sense of how to do this process, otherwise it can become corrupted. We believe we can avoid formal training in the method, but we need some knowledge and experience of it in place.*

EI.E2:

- The specification writer sincerely believes that
  - the defect level is low enough to exit.
  - They have done personal checking against the rules themselves and find no defects.
- *Rationale: the writer should*
  - *take the trouble to make sure the spec is as clean as possible before inspections.*
  - *They should not misuse people and time to compensate for sloppy work.*

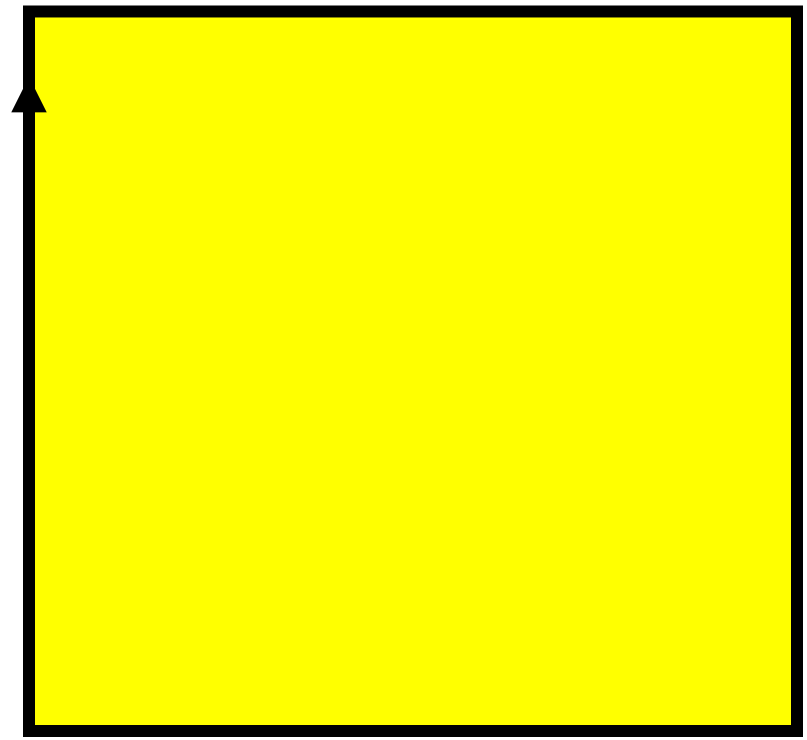
EI.E3:

- Exited copies of all source specifications are available.
  - *Rationale: there is little point in checking consistency against highly polluted source specifications.*
  - *(example by using bad Business Requirements to check new System Requirements).*

EI.E4:

- An updated 'Inspection Toolkit' (with specification Rules, Checklists (for learning to apply the rules in practice), Process descriptions, forms, electronic support, intended readership role information) is available and is understood by the participants.
  - *Rationale: This tool kit is the real definition of the Inspection process. This really determines correct use of the method.*

Ex In Procedure



EI.P1:

- The specification writer (‘writer’)
  - finds one other person (called a Checker)
  - to (help) carry out the QC (Quality Control) of their specification.

EI.P2:

- **a meeting time, with maximum duration 1.0 hour is agreed.**
- **(if the Checker is experienced, they can in fact do their checking at any time, alone, and report their results to the writer.)**

EI.P3:

- The writer makes sure the checker is knowledgeable about the following:
- the spec's intended readership and their uses of the spec.
- the specification Rules that apply (and their practical interpretation)
- The definition of Major defect, and how to spot them
- the purpose of the Spec QC process (to help the writer get to real exit-able level of defect density).

EI.P4:

- **The writer and the checker will each select the same one logical page ‘at random’ (300 Non-commentary words) sample to check.**
- **The writer is now performing the role of a ‘checker’ on their own work.**
- **They should agree that the page selected is representative of the quality of the rest of the document.**

EI.P5:

- **checking will be done individually**
  - **(but maybe in same room)**

EI.P6:

- the initial checking time will be 10 minutes.
- If NO Major defects are found by either checker.
- The checking process will continue for another 30 minutes.
- Even if no further Majors are found.

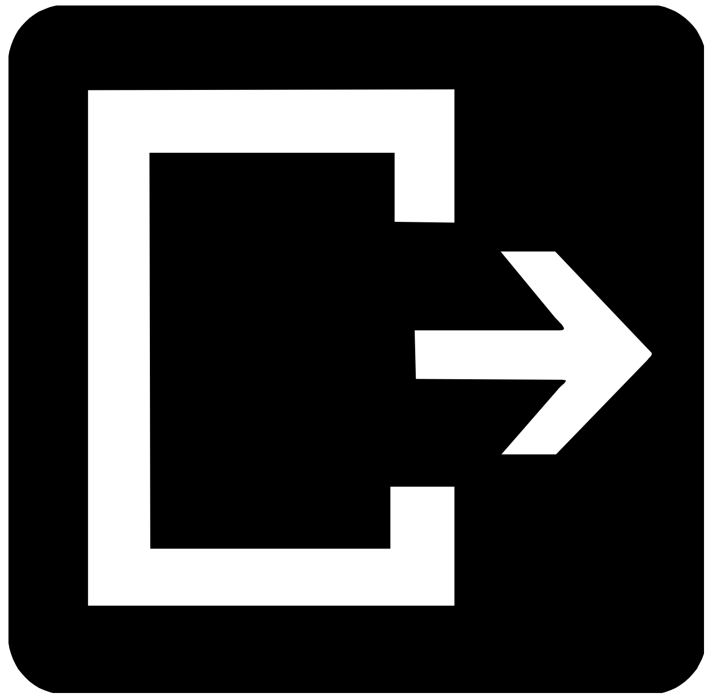
EI.P7:

- **If any Major defect is found**
  - **(and acknowledged by the writer as a real Major defect)**
  - **in the first 10 minutes of checking,**
  - **then this will be considered a sign that the spec contains many more major defects.**
  - **The writer will consider whether they want to stop the QC process and improve the spec,**
    - **or whether they want to continue for another 30 minutes to gather more Major defect cases**
      - **(to better signal what they need to rewrite).**

EI.P8:

- **At the end of the checking time,**
  - **the writer**
    - **(or the checker if they decide to take reporting responsibility)**
    - **will calculate the estimated Majors/Page in the current document**
    - **(using formulas or tools supplied)**
    - **and will report (on a form or to a database)**
      - **all time used and results**
      - **(Majors found,**
      - **Majors/page estimated,**
      - **decision to Exit or not, etc.)**

EI Exit Conditions



EI.X1: Defect Density Condition:

- **Estimated Major Defects remaining per page is less than 1 per 300 Non commentary words (initially until end 2003 10 Majors, to get a lenient start).**
- **FORMULA FOR ESTIMATION:**
- **Assume 33% effectiveness of the 2-checker checking-process.**
- **Total Unique Majors acknowledged by writer, found in the sample logical page, times 3, gives a reasonable estimate of Majors/Page. This is before writer correction of known Majors.**
- **Note: the effectiveness for a 3 checker group is slightly higher say about 40%. This figure needs to be determined by your own measurement.**

- **OPTION: we might manage the exit level at an individual writer level to gradually motivate them to improve by about 50% (defect injection) less per iteration of the write and check cycle. <- KM idea – TG likes it!**

- **NOTE: THE 33% effectiveness is based on experience, but it could vary, for example depending on the rate of checking used. The rate is controlled here because the time and the volume (a logical page) are controlled in the process.**

EI.X2:

- **Writer Veto**
- **The specification cannot exit if the spec writer wants more time to improve it.**

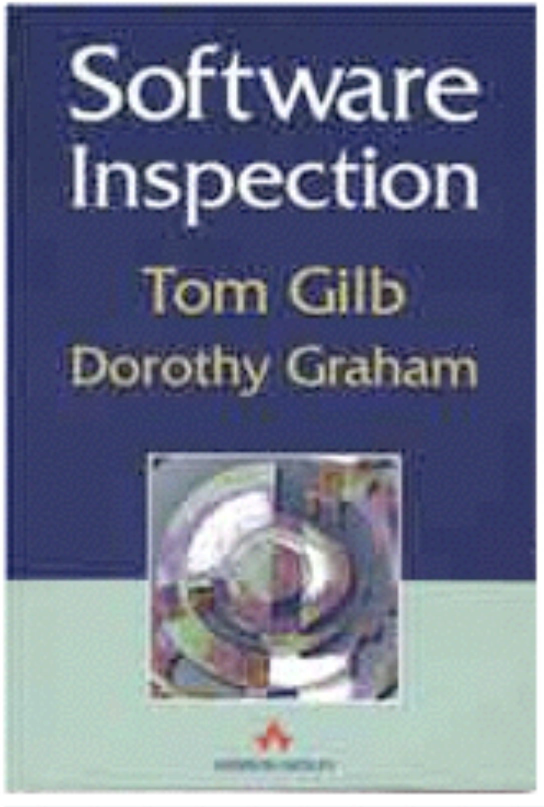
Some downloads about Agile SQC

- SQC Spec quality control paper
- agile inspection papers

- AgileCutter5p http://www.gilb.com/tiki-download_file.php?fileId=64
- INCOSE SQC… http://www.gilb.com/tiki-download_file.php?fileId=57
- Agile SQC Sl… http://www.gilb.com/tiki-download_file.php?fileId=239
- Rule Magazine http://www.gilb.com/tiki-download_file.php?fileId=192
- Eng.Rev.Pro… http://www.gilb.com/tiki-download_file.php?fileId=143
- Course Cert http://www.gilb.com/Inspection+Leader+Certification
- 2009 Test EX… http://www.gilb.com/tiki-download_file.php?fileId=264

Downloads about this Talk feel free to Tweet these!

- http://tcs.java.no/tcs/?id=E4C42322-C78C-42C0-9477-2DD742A7E837
  - A video of this Talk in Oslo Sept 2009
- ????
  - A download of a version of these slides


Software
Inspection
Tom Gilb
Dorothy Graham

Last Slide


TOM GILB
COMPETITIVE
ENGINEERING
A HANDBOOK FOR SYSTEMS ENGINEERING, REQUIREMENTS ENGINEERING, AND SOFTWARE ENGINEERING USING PLANGUAGE