Inspections : A Flexible Method to Increase Software Quality

By Sylvie Trudel
CRIM
sylvie.trudel@crim.ca

CUSEC, Concordia University

March 2002

Presentation Objectives

- Provide a level of awareness on inspection practices in the software industry
- Demonstrate flexibilty in a formal technical review method

CRIM

Agenda

- History
- Why Inspect ?
- Inspection Objectives
- Inspection Method
- Quality Assurance And Managers Role
- Return On Investment
- Leverage and Barriers

CRIM

History

- Created in the late 60's (IBM)
- Published by Fagan in the 70's
- Many methods developed over the last 10 years :
  - We will show an adaptation of Tom Gilb and Dorothy Graham's work
  - Deployed and optimized over 2000 software inspections

CRIM

Why Inspect ?

- Quality : Find and correct defects
- Productivity : reduce time-to-market
- Efficiency : Correct defects at their point of origin
- Cost :
  - up to 1,000 times cheaper to correct an error during analysis than during test
  - up to 40,000 times if the defect is correct before deployment!

CRIM

Inspection Objectives

- Find and fix defects
- Help the developer
- Reduce time-to-market
- Alleviate the « downstream » effect of defects
- Train team members (knowledge transfer)
- Process improvement
- And many more!

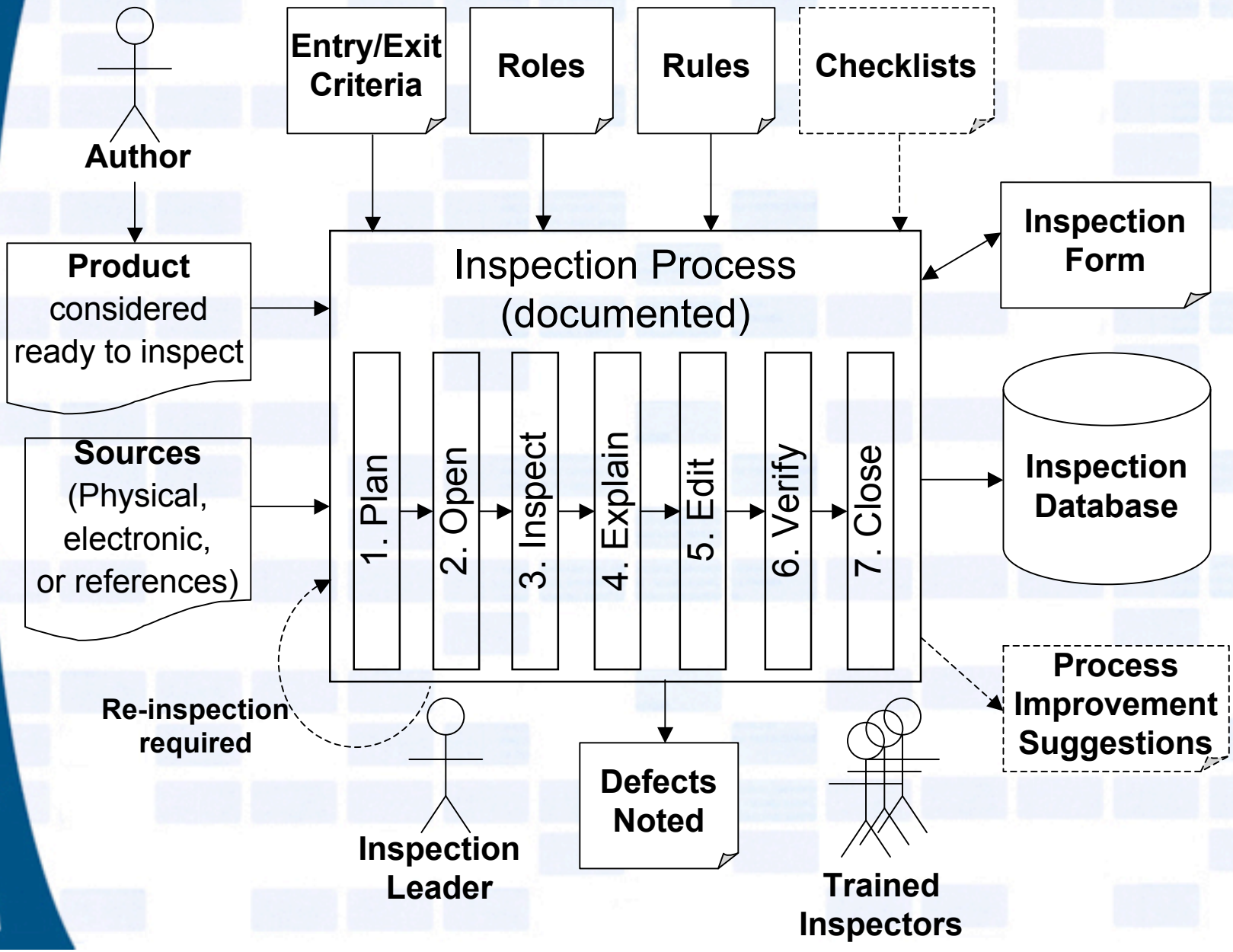
CRIM

Inspection Is Not...

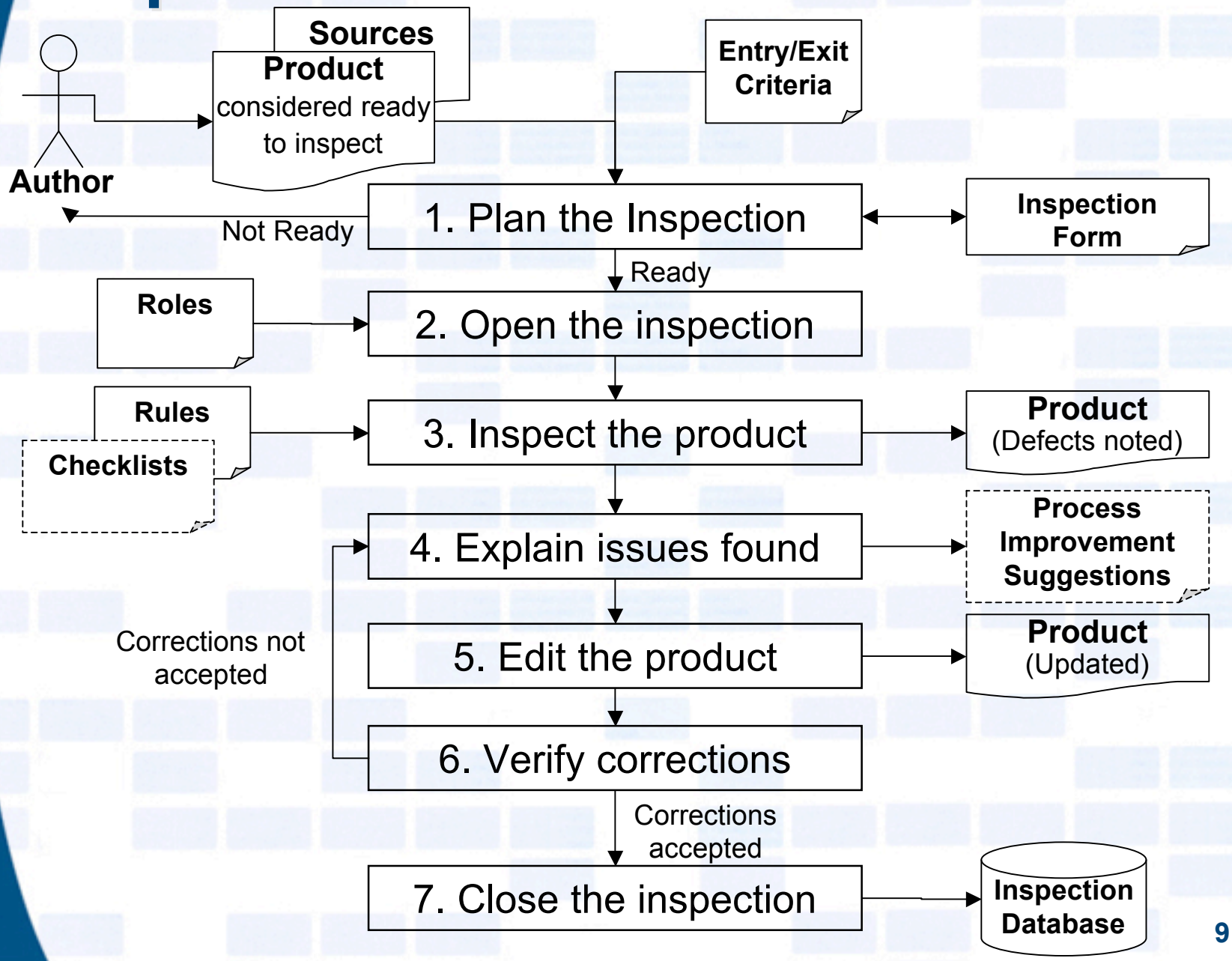
- Design Optimization
- Quality of design approval
- Level of ambition approval
- A discussion forum
- A bureaucratic routine!
- IT IS:
  - Inner consistency checking of all related documents, by peers
  - Verifying against defined best practices (rules, checklists)
  - Only done if measurably profitable

CRIM

Inspection Context



Inspection Process : Detailed View



Inspecting: The Key Points

- 2 to 5 inspectors :
  - at least 2 different roles
  - the author cannot be the inspection leader
  - the author can be an inspector with at least 2 other inspectors
- Inspect slowly to find more important defects :
  - 3 to 5 pages/hour for most documents
  - up to 20 pages/hour for code (~1,000 LOC)
- Attitude, attitude, attitude!
  - Respect, professionalism, minded to help the author, aim for excellence, give/share/receive

CRIM

Inspection Rules : Some Examples

- Generic
  - Complete
  - Clear
  - Consistent
  - Correct
  - Brief
  - Relevant
- Design
  - Loose coupling
  - High cohesion
- Requirements
  - Testable
  - High-level
  - Elementary
- Code
  - Confined
  - Complexity
  - Style

CRIM

Most Common Inspector Roles

- Logic
- Requirements
- User interface
  - Usability
  - Consistency
- Standards
- Quality
- Design
  - Coupling
  - Cohesion
- Rules
- Algorithm
- Graphics
- Financial

CRIM

Quality Assurance Role

- Audit inspection process usage
- Inspect with a role such as « standard » or « quality »
- Show « due diligence »
  - Have its own documents inspected

CRIM

Manager Role

- Support inspection process usage
- Provide resources
- **NEVER** ask for individual inspection results
- Show « due diligence »
  - Have its own documents inspected
- Show enthusiasm on positive results
- Sanction the “outlaws”
- Do not inspect subordonate’s work
  - unless they insist

CRIM

Return On Investistment

- Ratio of rework saved by inspection hour :
  - According to Gilb ⇒ 9.3 : 1
  - According to an IBM study ⇒ 18 : 1
- Account for :
  - Cost of quality (inspection effort)
  - Cost of non-quality
  - Corollaries (hard to put numbers on)
    - More performant software development teams (better productivity through enhanced communication)
    - Organization's reputation

CRIM

Leverage

- Positive attitude towards quality
- Willingness to improve at all levels
- Customer or contractual requirements
- Working climate that promotes :
  - respect
  - excellence
  - innovation

CRIM

Barriers

- Lack of resources
- Previous bad experience
- A blame culture
- Big egos, prima donna behaviours
- Opinion leaders who reject any process

CRIM

You certainly have questions ?

Lets inspect!

References

- Tom Gilb and Dorothy Graham, Software inspections, Addison-Wesley, 1993
- Tom Gilb, Document Inspection Team Leader Course material, version 8.21, 1996
- Roger S. Pressman, Software Engineering: A practitioner's Approach, 5th édition, McGraw-Hill, 2001
- Partick D'Astous, Les aspects de l'échange d'information dans un processus de génie logiciel, École Polytechnique de Montréal, 1996
- Karl E. Wiegers, Software Requirements, Microsoft Press, 2000

CRIM

Contact

Sylvie Trudel

Software Engineering Specialist

sylvie.trudel@crim.ca

(514) 840-1235 ext. 4562

CRIM Software Test Center

www.crim.ca

CRIM