Requirements validation

- Structured walkthroughs
- Why have these reviews
- When to have reviews
- Who participates in the reviews
- What procedures are helpful

Thoughtless reviews:

The process whereby the false assumptions of one member of the team become shared by the entire team

Some reasons to seek the help of reviews:

- http://www.pm2go.com/sample_research/chaos_1994_1.php

Structured Walkthroughs

- A walkthrough is a peer group review of any technical product
- Systems Theory tells us that within a system, we cannot recognize fundamental defects except from outside that system
- The review involves other
  - systems analysts who are working with you,
  - as well as users, programmers,
  - systems designers,
  - operations people,
  - and others
- Typically, does *not* include
  - your boss
  - the head of the department
  - the vice-president from the user organization.

Quality reviews

- The principal method of validating the quality of a process or of a product
- Group examines part or all of a process or system and its documentation to find potential problems
- Reviews can have different objectives
  - Inspections for defect removal (product) -- detailed inspection of the requirements, design and code (often uses check list)
  - Reviews for progress assessment -- product and process review to assess project progress against projected costs, plans, and schedule
  - Quality reviews -- faults or mismatches between requirements, specification, design, code, and documentation.
    - May also include broader issues such as adherence to standards and other quality metrics
- We don’t get much practice as students because of cheating policies :-)

Quality reviews have a role throughout the project

- Analysis walkthroughs
- Design walkthroughs
- Code walkthroughs
- Test walkthroughs

In practical terms, analysis walkthrough means that a group of systems analysts, together with other interested parties, gather together to review use-case diagrams, object models, dataflow diagrams, entity-relationship diagrams, state-transition diagrams, data dictionary entries, and process specifications, i.e., all the technical products developed by the systems analyst.

- Poor quality should not drive the schedule for reviews
  - ***regularly scheduled reviews needed*** for quality assessment as well as improvement
- Any artifacts of the development process are fair game for review
- Research says review teams should be relatively small (Wiegers)
  - and reviews should be fairly short
- Reviews should be recorded and records maintained
  - recall CMM required data
- Inspections and reviews are nearly universally considered a “best practice” for software development
  - http://www.christine.com/bestprac/bestprac.htm
  - http://www.spmn.com/16ptp/sld006.htm

Analysis walkthroughs work!

- 50% + of *all* software defects can be traced back to the requirements
  - and probably 80% of the rework is due to poor requirements (Wiegers)
- Fagan (78) claims dramatic results from inspections in general
  - do a web search on requirements, defects, inspections and find a lot of data saying inspections and walkthroughs are ***more cost effective*** than other techniques for defect discovery
  - a CMM-5 group reports inspections obviate tests for defects (Hamlet01)
  - interesting note: Porter's work in software code inspections shows good techniques equally (or more) important than team meetings in finding defects
    - this data is contrasted to older studies and is done on code products
- What would eXtreme programming say about all this?
  - they "fold in" the benefits of inspections with the "pair programming" concept: and it is proven to result in cost effective defect removal (Beck)
    - though they don't put much focus on requirements documents :-(

Analysis review objectives: Example Data Flow Diagrams

Automated checking (using an analysis tool can only go so far)

What humans might ask more effectively:

- Are there too many bubbles in the diagram?
- Have the process names been chosen in a meaningful way?
  - have the flows been labeled meaningfully?
- Has the diagram been drawn so that it is esthetically pleasing?
  - Is it likely that the user will actually read the diagram, or likely to be confused by it?
- Have elementary activities been packaged together in an intelligent way to form higher-level bubbles?
  - and dataflows packaged in a meaningful way from one level to another?
- Does the whole diagram "hang together"
  - can I understand it, explain it? accurately?

Analysis review objectives: Example Use Case Diagrams

- Is the normal course complete?
- Have all the alternative courses been discussed?
- Are includes chosen appropriately?
- Is the priority appropriate?
- Do the pre-conditions and the “course of events” necessarily lead to the post-conditions?
- Have exceptions been considered?
  - this one is often overlooked
    - what are the things you don’t want to have happen? how will the system respond?
- Are there “related requirements” implied by this Use Case that must be documented elsewhere?

Requirements validation:

Define validation checklists:

Are the requirements

- Complete? (Have you covered all necessary elements? *Hard*.)
- Consistent? (Can you get a big “enough” picture? *Hard*.)
- Comprehensible? (Think like a 205 student who reads this.)
- Precise? (Think like a programmer here.)
- Traceable? (Can you find the source of the requirement?)
- Testable? (Can you envision a simple test for each requirement?)
- According to standards?

Sample Requirements Checklists abound on the web (and in your text)

Ideally, you write your own after studying several samples

Roles at quality reviews

- **Presenter**, the person who explains to the reviewing group what the product does, what assumptions were made when it was created, and so on. In many cases, the presenter is the producer, but not always.
- **Chairperson**, the person who organizes the meeting and runs it. His or her purpose is to keep the discussion moving along in an orderly, constructive way, to prevent tangential discussions, as well as critiques of the presenter.
- **Scribe**, the person who makes a written record of the significant events of the review. Scribe writes notes about significant technical questions that were raised, bugs that were found, and suggestions for improvement or modification made by the reviewers.
- **Maintenance oracle**, a reviewer whose main orientation is the long-term maintenance of the product.
- **Standards bearer**, the role of this person is obvious: to ensure that the product is consistent with the overall standards that have been adopted by the project and/or organization.

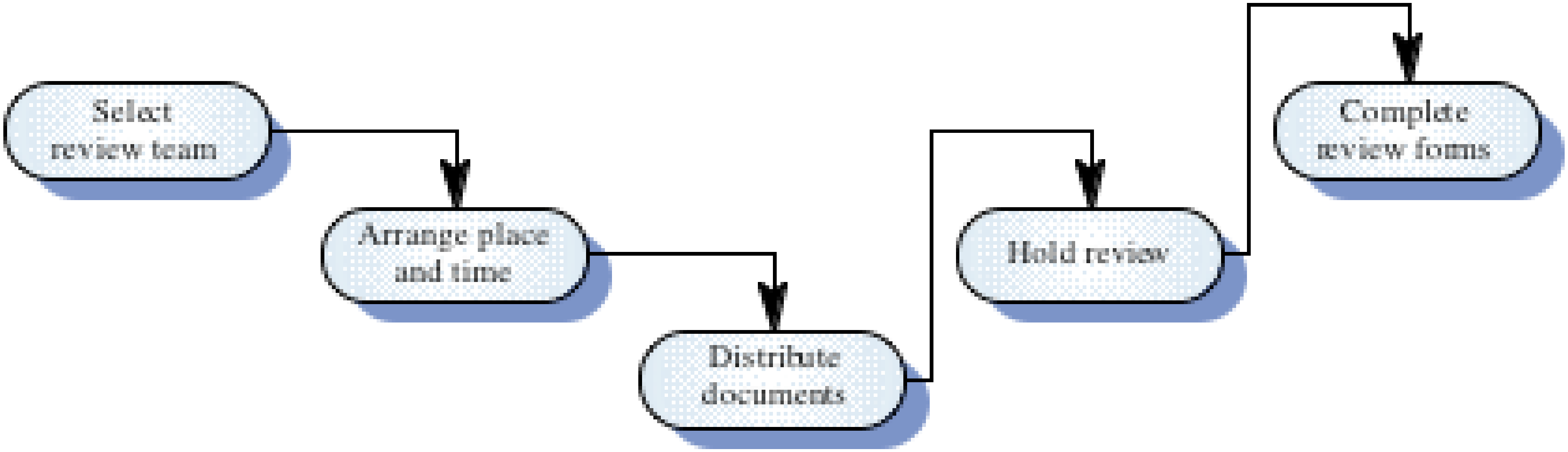
What are the benefits of reviews

- Quality function - they are part of the general quality management process
  - data shows reviews a cost effective way to discover defects (early)
- Project management function - they provide information for project managers
  - “visibility” of the development process / product are essential to good management (and vital to health of project according to Chaos Report)
- Training and communication function - product knowledge is passed between development team members
  - don’t underestimate this function, the “learning” and “communication” aspects of reviews

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 concept***. The problem identified in the review impacts other parts of the artifact. Some overall judgement must be made about the most cost-effective way of solving the problem.
    - Requirements and specification errors may have to be referred to the client.

The review process



The Technical Review Process

- Pre-Review Activities
- Review Meeting Activities
- Post-Review Activities

Pre-Review Activities (Production team)

- Prepare and make available the artifact to be reviewed
  - For example the Software Requirements Specification
  - Supporting material e.g. Vision and Scope
- Call the meeting
  - Participants
  - Time and place
  - Identification of artifact to be reviewed
  - Agenda

Pre-Review Activities (Review team)

- Inspect the artifact and note comments.
  Identify
  - Standards compliance
    - e.g., document structure (incl. cross-references, sign-offs etc.), possibility of validation,…
  - Presentation quality
    - e.g., readability, clarity of information, well-definedness, completeness…
  - Contents quality
    - Conformance – does it satisfy its requirements
      - well, “requirements” do have requirements!
    - Structure – is it technically satisfactory

Pre-Review Activities
(usually the chairperson)

- Prioritize the comments
  - High-impact
  - Low-impact
- Identify action items
- Write it up
- Deliver it to the review team

Review Meeting Activities

- Review the artifact, *not the team*
- Set an agenda, and *maintain it*
- *Limit debate* and rebuttal
- Identify problems, *defer problem resolution*
- Use a recorder

Defect report:
For each defect

- Defect ID
- Description
- Procedure to reproduce
- Platform info
- Date identified
- By whom
- Severity
- Phase
- Date corrected
- By whom
- Effort (in staff hours)
- Work product corrected
- Resolution status
- Notes

Review Meeting Activities

Agenda Item	Presenter	Duration
Introduce Producers	Joe Kelley	1 min.
Review Meeting Purpose	Joe Kelley	5 min.
Summarize Low-impact Comments	Mary Talley	5 min.
Clarify High-impact Comments	Sue Thompson	30 min.
Review Action Item List	Tom Gramaglia	5 min.
Complete Review Summary	Joe Kelley	5 min.
Process Meeting	Mary Talley	5 min.

Post-Review Activities

- Analyze the comments
- Determine course of action (action items)
  - accountability
- File
  - the review report,
  - the minutes from the meeting(s),
  - the action items
  - their follow-up

Lessons from Reviews

- Small teams, short meetings
- Ego isn’t appropriate
- Find, don’t fix
- Be methodical

Challenges to Reviews

- Boring work (compared to other work)
- Costly work
  - consider (N team members) ***X*** (M hours, meeting and prep)
- Voluminous documents
  - solution: review separable pieces
    - not always easy or possible
- Large team of reviewers
  - smaller teams more manageable
    - remember Brooks' Law and its implications for communication (argumentation) paths!
- Political pressures
  - managers who want to find scapegoats
  - team members who want to grandstand