Perspective-Based Reading (PBR)

- Researchers at Experimental Software Engineering Group at the University of Maryland, College Park, have created Perspective-Based Reading (PBR) to provide a set of software reading techniques for finding defects in English-language requirements documents

Different Perspectives - 1

- PBR operates under the premise that different information in the requirements is more or less important for the different uses of the document
- Each user of the requirements document finds different aspects of the requirements important for accomplishing a particular task

Different Perspectives - 2

- PBR provides a set of individual reviews, each from a particular requirements user's point of view, that collectively cover the document's relevant aspects
- This process is similar to constructing system use cases, which requires identifying who will use the system and in what way

Steps in PBR

- Selecting a set of perspectives for reviewing the requirements document
- Creating or tailoring procedures for each perspective usable for building a model of the relevant requirements information
- Augmenting each procedure with questions for finding defects while creating the model
- Applying procedures to review the document

Two Questions

- What information in these documents should they check?
- How do they identify defects in that information?

Benefits of Different Perspectives - 1

- Systematic
  - Explicitly identifying the different uses for the requirements gives reviewers a definite procedure for verifying whether those uses are achievable
- Focused
  - PBR helps reviewers concentrate more effectively on certain types of defects, rather than having to look for all types

Benefits of Different Perspectives - 2

- Goal-oriented and customizable
  - Reviewers can tailor perspectives based on the current goals of the organization
- Transferable via training
  - PBR works from a definite procedure, and not the reviewer's own experience with recognizing defects, new reviewers can receive training in the procedures' steps

Identifying Defects

- A series of questions are used to identify different types of requirements defects
- Requirements that do not provide enough information to answer the questions usually do not provide enough information to support the user. Thus, reviewers can identify and fix defects beforehand

Requirements Defects that PBR Helps Detect

- Missing information
- Ambiguous information
- Inconsistent information
- Incorrect fact
- Extraneous information
- Miscellaneous defects

Missing Information - 1

- Any significant requirement related to functionality, performance, design constraints, attributes, or external interface not included
- Undefined software responses to all realizable classes of input data in all realizable classes of situations

Missing Information - 2

- Sections of the requirements document
- Figure labels and references, tables, and diagrams
- Definitions of terms and units of measures

Ambiguous Information

- Multiple interpretations caused by using multiple terms for the same characteristic or multiple meanings of a term in a particular context

Inconsistent Information

- Two or more requirements that conflict with one another

Incorrect Facts

- A requirement-asserted fact that cannot be true under the conditions specified for the system

Extraneous Information

- Unnecessary or unused information (at best, it is irrelevant; at worst, it may confuse requirements users)

Miscellaneous Defects

- Other errors, such as including a requirement in the wrong section

Benefits of PBR's Detailed Questions

- Allow controlled improvement
  - Reviewers can reword, add, or delete specific questions
- Allow training
  - Reviewers can train to better understand the parts of a representation or work product that correspond to particular questions

PBR General Questions - 1

- Does the requirement make sense from what you know about the application or from what is specified in the general description?
- Do you have all the information necessary to identify the inputs to the requirement? Based on the general requirements and your domain knowledge, are these inputs correct for this requirement?

PBR General Questions - 2

- Have any of the necessary inputs been omitted?
- Are any inputs specified that are not needed for this requirement?
- Is this requirement in the appropriate section of the document?

Results of PBR Experiments - 1

- PBR provides a framework that represents an improved approach for conducting requirements reviews
- This approach will only be effective when an organization tailors the framework to its own needs and uses feedback from its reviewers to continually improve and refine the techniques

Results of PBR Experiments - 2

- PBR seems best suited for reviewers with a certain range of experience (not too little; not too much)
- Development teams that use PBR to inspect requirements documents tend to detect more defects than they do using other less- structured approaches

Results of PBR Experiments - 3

- Relatively novice reviewers can use PBR techniques to apply their expertise in other development tasks to defect detection
- Using PBR improves team meeting by helping team members build up expertise in different aspects of a requirements document

Results of PBR Experiments - 4

- It creates high-level representations of the software system, usable as a basis of work products in later stages of the development
- Each development organization can customize PBR's set of perspectives, level of detail, and types of questions

Results of PBR Experiments - 5

- PBR facilitates controlled improvements, providing a definite procedure, alterable according to projects metrics and reviewers' feedback

Summary

- Discussed defect removal and in particular inspections using, perspective-based reading

References - 1

- Software Engineering: A Practitioner's Approach by Roger S. Pressman
- A Handbook of Software Quality Assurance edited by G. Gordon Schulmeyer and James L. McManus
- Customer-Oriented Software Quality Assurance by Frank P. Ginac
- Software Quality: Analysis and Guidelines for Success by Capers Jones

References - 2

- ‘How Perspective-Based Reading Can Improve Requirements Inspections’ by Forrest Shull, Ioana Rus, & Victor Basili, IEEE Computer, July 2000, pp. 73-79