Change impact analysis - what, why, how?

Tutorial

2008. 10. 15

Seunghun Park
Contents

- Introduction
- What is the change impact analysis
- Why we perform the change impact analysis
- How to perform the change impact analysis
- Future direction of research
- Q&A
Software change is inevitable

- Reasons why still struggles with issues of software change
  - Software requirements are continually changing to accommodate user expectations, new operational environment needs, etc.
    - Changes to requirements may lead to massive software changes
  - Development lifecycle of large software system can be quite long
    - Few people stay with the project for entire system development cycle
    - Few people remember the initial requirements or the rationale behind the requirements or design
  - Software change cycle for large and complex system can be slow
    - Software changes can have unpredictable consequences that often delay their implementation
Role of impact analysis in a software change process

Difficulties of change impact analysis

- Manual change impact analysis is labor-intensive and error-prone
- Automatic change impact analysis techniques provide a limited analysis depending on the ability to
  - Create models of relationships among artifacts
  - Capture the relationships in software and associated representations
  - Trace the relationships and reasonably bound the search space

Objective of this tutorial

- Understand basic concepts of the change impact analysis
- Examine advantages and disadvantages of a variety of techniques for determining the impacts of changes
- Discuss the future direction of the change impact analysis research
Definition of change impact analysis

- There is no consensus definition
  - Evaluation of the many risks associated with the change, including estimate of the effects on resources, effort, and schedule (by S. Pfleeger)
  - Identifying the potential consequences of a change, or estimating what needs to be modified to accomplish a change (by S. Arnold)
  - Assessment of a change, to the source code of a module, on the other modules of the system (by R. Turver)
What is the change impact analysis (2/3)

- Typical change impact analysis process

1. Analyze change proposal
2. Analyze artifacts
3. Identify initial impacts
4. Determine ripple effects
5. Implement change
Results of the change impact analysis

Correct results:
- Predicted and modified
- Not predicted and not modified

Incorrect results:
- Predicted and not modified
- Not predicted and modified

Actual impact set and Predicted impact set
Why we perform the change impact analysis (1/2)

Benefits of change impact analysis

- Helps software engineers plan and accommodate changes, and trace the effects of changes
  - Makes the potential effects of changes visible before the changes are implemented
  - Makes the change management easier and more accurate
- Used as a measure of the cost of a change
  - Allows an assessment of the cost of the change and helps the management choose tradeoffs between alternative changes
  - Allows engineers to evaluate the appropriateness of a proposed change
Applications of change impact analysis

- Software development
  - Program integration
    - To determine whether the changes interfere
      » Create an integrated program that incorporates both sets of changes as well as the unchanged portions
  - Regression testing
    - To reduce the cost of regression testing
      » Determine the parts of a program that need to be re-tested after a change is made

- Software maintenance
  - To understand existing software and make changes without having a negative impact on the unchanged part

- Software quality assurance
  - To locate all code or document that contributes to the part of a specific quality attribute
Two views for classifying the change impact analysis

Types of relationships

- Traceability
- Dependency

Searching Strategies

- Exhaustive
- Semantically guided
- Heuristically guided
- Stochastically guided
- Hybrid guided
Types of relationships

Dependency analysis

- Examines detailed dependency relationships among program entities, often source codes
  - Provides a detailed evaluation of low-level dependencies in code but does not little for artifacts at other levels
- Considers the data dependency and the control dependency
  - Data dependency
    » Represents the relationships among program statements that define or use the data of program
    » Exists when a statement provides a value directly or indirectly used by another statements in a program
  - Control dependency
    » Represents the relationships among program statements that control the program execution
    » Provides information on the logical decision points in a program
Types of relationships (cont’d)

- Traceability analysis
  - Addresses the impact analysis from a broader perspective
  - Uses the two kinds of traceability relationships
    - Horizontal traceability
      - Addresses the relationships between different models or artifacts
    - Vertical traceability
      - Expresses the relationships within a model or an artifact

*M. Lindvall, “Traceability aspects of impact analysis in object-oriented systems, 1998*
How to perform the change impact analysis (4/10)

Searching strategies to determine the impacts

- Unguided/exhaustive
  - Identify impacts in a “brute force” manner

- Semantically guided
  - Directed by predefined semantics of objects and relationships

- Heuristically guided
  - Directed by predetermined rules or heuristics to suggest possible paths to examine or dubious ones to avoid

- Stochastically guided
  - Guided by probabilities for a specific situation

- Hybrid guided
  - Identify impacts using a combination of the above
Change impact analysis techniques

- Transitive closure
  - Determines all objects reachable from an object to any other through paths of length 0 or more
  - Does not need the additional information to perform the search
  - Is an easy and simple way to explain and implement the impact analysis
  - Is usually a starting point for prototyping impact analysis tools and for developing faster search techniques
Change impact analysis techniques (cont’d)

- Inferencing
  - Uses rules to characterize relationships among objects
    - Rules define how it is explored, as well as the underlying relationships among objects or properties of objects
  - Consists of a database of facts and the ways to infer new facts from previous ones
  - Performed by inference engines which provide the evaluation mechanisms to determine potential impacts

![Diagram](image)

Simple transitive rule
Calls(X,Y) and Calls(Y,Z) → Indirectly_Calls(X,Z)

Indirectly_Calls(A,C) (by Calls(A,B) and Calls(B,C))
Change impact analysis techniques (cont’d)

Two strategies of inferencing

- **Truth maintenance system**
  - A type of constraint system that uses constraint rules to characterize relationships among objects
  - Maintains the consistency in a model
  - Checks the change of the model for inconsistencies, identified them, and notifies them to users

- **Semantic inferencing**
  - Uses the semantic information which represents the knowledge of software
    - Semantics are represented as logics’ axioms for applying logic algorithms to determine the impacts of software changes
    - Semantics can include the design knowledge or dependencies
Change impact analysis techniques (cont’d)

- Program slicing
  - Identifies program statements which may potentially affect or be affected by the computation of a specific variable at a statement
    - Decomposition technique for extracting some specified parts from the program that are currently of interest
  - Produces a program slice
    - Executable program including only parts of the original program that affected by a change
    - Taken with respect to a slicing criterion \((s,v)\) such that \(s\) is a statement and \(v\) is a variable
  - Uses the control dependency and the data dependency of a program
    - Creates a program dependence graph using the dependencies
  - There are various kinds of slicing techniques
    - Static, dynamic, intra-procedural, inter-procedural, etc.
How to perform the change impact analysis (9/10)

- Program slicing
  - Example

Original program

```plaintext
program Main
    SUM := 0
    i := 1
    while i < 11 do
        SUM := SUM + i
        i := i + 1
    od
    print (SUM)
    print (i)
end
```

Program dependence graph

- Edge Key
  - ➔: Data dependence
  - ➔: Control dependence
How to perform the change impact analysis (10/10)

Classification of the change impact analysis techniques

<table>
<thead>
<tr>
<th></th>
<th>Exhaustive</th>
<th>Semantically guided</th>
<th>Heuristically guided</th>
<th>Stochastically guided</th>
<th>Hybrid guided</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dependency</td>
<td>Transitive closure</td>
<td>Semantic inferencing</td>
<td>•</td>
<td>•</td>
<td>Program slicing</td>
</tr>
<tr>
<td>Traceability</td>
<td></td>
<td></td>
<td>Truth maintenance</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Advantages and disadvantages of the techniques

<table>
<thead>
<tr>
<th></th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transitive closure</td>
<td>• Easy, simple, and intuitive to implement</td>
<td>• Slow for running</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Too many false-positives</td>
</tr>
<tr>
<td>Inferencing</td>
<td>• Flexible according to relationship rules</td>
<td>• Depending on the knowledge of experts</td>
</tr>
<tr>
<td></td>
<td>• Enable to reduce the search space</td>
<td></td>
</tr>
<tr>
<td>Program slicing</td>
<td>• Outperform without additional information</td>
<td>• Need the complete dependency analysis</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Larger part of program than actual impacted part</td>
</tr>
</tbody>
</table>
Multi perspectives of change impact analysis
Change impact analysis for NFRs
- Most strategies for impact analysis work under the assumption that changes only affect functionality.
- It is difficult to assess the impact of changes to NFRs or changes where NFRs are indirectly affected.

Change impact analysis with partial information
- Most automatic strategies for impact analysis assume that there are complete models and full traceability information.
- There is a need for more robust impact analysis strategies that can work with partial information.
Change impact analysis in new development technologies

- It needs to be adapted to the types of systems that become increasingly common today, such as SOA and COTS software
  - Data dependencies of repositories which can be shared among several distinct systems is important
  - It should consider the interoperability as well as the data and control dependencies