A System Model for Managing Requirement Trace Matrices via Statistical Artifact Change Analysis

Benjamin J. Deaver
Advisor – Dr. LiGuo Huang
Department of Computer Science and Engineering
Southern Methodist University
The Requirements Traceability Matrix (RTM) is utilized to:

◦ Link deliverables to requirements
◦ Identify Overlap
◦ Ensure fulfillment

Critical Change Management Tool
◦ Highly effective in identifying collateral impact of change

In dynamic landscapes, the RTM can easily fall out of a known validity
◦ Rapid Change
One of the major challenges facing the implementation of traceability is the cost.

- Systems grow in size and complexity
- Systems become more dynamic
- Manual generation of RTM is a tremendous undertaking
  - When the is manual generation required?
  - “After each change” may not be feasible.
  - “Allowing multiple changes may result in missed overlaps
Dynamic Landscapes
- Rapidly changing requirements
- New requirements being generated
- Concurrent development being implemented
  - Identifying Overlap before problems arise

Growth in Size and Complexity
- Over time, a system being constructed has a tendency to grow larger.
- With growth comes complexity.
- Size and Complexity can generate overlap and reuse.
What is the value of knowledge about the changing Requirements Traceability?

- We can certainly understand the cost associated with the RTM.

The value of understanding the impact of change to a System / System of Systems

The value of understanding the impact of change to a RTM
How is the problem being solved today?

- Many different areas of research focus on the effective utilization of the RTM.
- Not a significant amount of research built around determining confidence in the RTM and determining the need for regeneration.

- Automated RTM Management
- Manual Generation and Management
- IR Techniques
Automated RTM Management
- Software with the purpose of maintaining the links between requirements and deliverables.
- Effective for the mining of information from an RTM.
- Ineffective at generating the RTM.
- Highly ineffective at identifying the effectiveness of the RTM as systems evolve without additional assistance.
Manual Generation of the RTM.
- Intensely time consuming.
- Requires high levels of expertise and knowledge of the system in question.
- Based on the effort required for generation, a risk of being outdated before full generation exists.

Research is being conducted in the area of speeding the delivery of the RTM.
- Significant increases, but still significant effort.
- Value Based RTM Generation.
Related Work

Information Retrieval Methods
- Several areas and methods are being researched
- Latent Semantic Analysis of artifacts
- Clustering of artifacts
- Effectiveness varies greatly
  - ~50% precision and ~50% recall are median results.
  - Precision or recall can be increased at the expense of the other.
  - Not effective enough to make decisions regarding change management.
Identify the confidence in an RTM based on changes since generation.

- All changes will be classified according to some generated taxonomy of change.
- Based on this taxonomy of change, what changes will have the greatest impact on the RTM?
- What combinations of changes will indicate a loss of confidence in the RTM?
Taxonomy of Change

◦ All change can be broken into set categories.
◦ Taxonomy may be different for
  • Software Engineering
  • Systems Engineering
  • Systems of Systems Engineering
Requirements Engineers are familiar with
- Deliverables
- Requirements
- The RTM linking the two

Based upon the changes applied since the last generation of the RTM
- Changes are classified via their respective Taxonomy of Change
- The order of change may have an impact as well
  - Interesting follow up questions generated from this model.
Develop a tool which will mine changes from version management systems and identify changes.
  ◦ Attempt to automate the classification of change in a taxonomy of change.

Results in a system which will
  ◦ Rapidly identify the types of changes taking place
  ◦ The order of the changes taking place
  ◦ The statistical likelihood of the RTM being impacted by change
    • Confidence in the RTM is derived from this data.
Based on the number of changes seen in a system, the confidence in the RTM will diminish

- Confidence being the likelihood of correctness
- We expect different types of change to have different impact
Utilize OSS projects with a substantial version history
  ◦ Gantt
  ◦ ReactOS
  ◦ jHotDraw
Utilize validated traces as a starting point for further analysis.
Identify and classify changes between known versions.
Retrace at major (stable) releases.
Based on the impact of changes between traced versions, we will derive the statistical model.
Utilize Traces Generated by Institute for Software Engineering and Automation
  ◦ Reuse methods for creation of additional traces for additional releases of software
  ◦ Working closely with Dr. Alexander Egyed

Utilize Software Engineering Graduate Students to create Taxonomy of Change
Tool to Automatically Mine Changes and classify them.
Examine the totality of changes between versions.
- Based on the types of changes that have taken place, evaluate the total impact.
- Identify the impact of individual changes.
- Based on the types of change identified, identify the impact of changes.

Other interesting questions.
- Do some changes appear together?
- Are certain combinations more impactful?
- At some number of changes, does it no longer matter what kind of change is being made?
Requirements for tracing have been identified and documented in all three OSS projects.
Initial Traces have been collected.
Data gathering exercises have begun with Gantt versions.
  ◦ 2.0.8 → 2.0.9
  ◦ 2.0.9 → 2.0.10
Examination of change repository data for projects is being conducted.
Our research is still in the preliminary phases.

- Initial examination of results (Gantt) is scheduled for the end of 2011.
- Taxonomy of change has been identified.
- Frequency and Impact of different types of change is being analyzed.
- Original hypothesis will be tested based on collected Gantt data.
  - Results of this experiment will drive the continuation of research in this area.
Modification of taxonomy will allow for rapid scaling between:
  ◦ Software Engineering
  ◦ Systems Engineering
  ◦ Systems of Systems Engineering

Knowledge of RTM Reliability

A greater understanding of the impact of specific types of change to a System / SoS.

A greater understanding of the impact of specific types of change to the RTM
Conclusion & Future Work

- Few conclusions at this time.
  - The Taxonomy of Change has been established.
    - Initial research shows that all change can be accounted for.
    - New categories may be uncovered with additional research.

- Future Work
  - Should all change be included in the research?
    - Changes that are later reversed?
    - Delta of change to a version is 0, but what if other changes occur between the initial and reversal?
Follow up questions or comments can be directed to Benjamin Deaver

bdeaver@smu.edu