Software Re-Engineering: Technology Maturation Study

Marwan Abi-Antoun

17-939A: What Makes Good Research in Software Engineering?
Professor Mary Shaw
Spring 2005
General Model for Software Reengineering

General Model for Software Reengineering

Reverse Engineering (Abstraction)

Conceputal Requirements

Design

Implementation

Existing System

(Alteration)

re-design

Forward Engineering (Refinement)

Conceputal Requirements

Design

Implementation

Design Recovery

Target System

• Analytic Models
  – Information about programs to aid restructuring, e.g.:
  
  Program Dependency Graphs (PDG) = Control Flow Graphs (CFG) + Data Dependency Graphs (DDG)

• Tools/Notations
  – Hide complexity/formalism of analytic model
  – E.g., developers use concrete syntax and do not think

  Tools/Notations

  Data Dependency Graphs (DDG)
  + Control Flow Graphs (CFG)
  = Program Dependency Graphs (PDG)

• Analytic Models

Research Methods
Software Re-engineering

- Enforcing “Intent” into Code
- Architecture Recovery
- Design Recovery
- Resource Flow Diagrams (MIL)
- Finding Program Invariants
- System Partitioning
- Structuring Flowgraphs
- GOTO less approach
- Subroutines

- Program Transformation
- Database Re-structuring
- Re-Modularization
- Re-Structuring
- Re-Designing
- Re-Architecting
Exclusions

- Software Conversion
- Visual Programming
- Program Execution/Algorithm Animation
- Restructuring Economics
Inclusions

- Imposing Block Structure
- Transformational Programming
- Good System Structure
- Program Understanding
Papers (pre-1979)

1. Ashcroft, E. and Manna, Z. The translation of 'go to' programs to 'while' programs. 51'ANC5-71-188, Stanford, CA. 1971.

7. TBD
Papers (1990-1999)

Papers (2000 - )

3. TBD