Software Engineering 4 (F22HX3)

Cause and Effect Analysis

Andrew Ireland

School of Mathematical & Computer Sciences
Heriot-Watt University
Edinburgh

Cause-Effect Graphs

- Dates back to IBM research & development in 1970's.
- Cause-effect graphs provide a representational technique for capturing the relationship between program inputs (causes) and program outputs (effects).

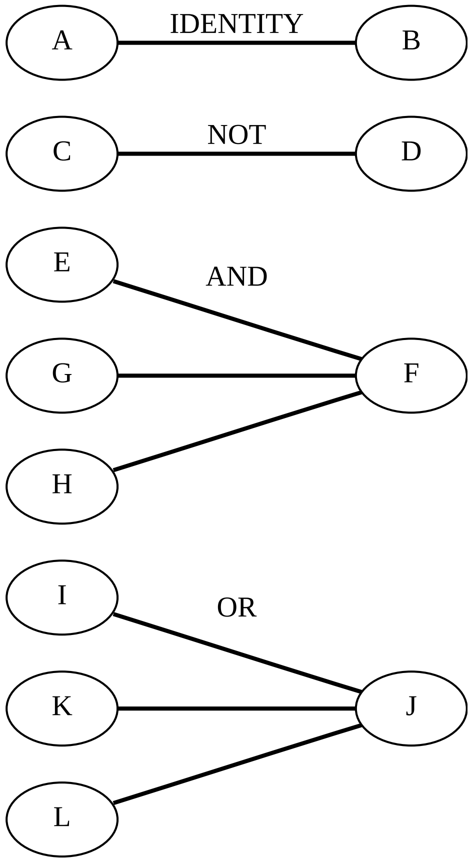
  "... gives one a different perspective on, and additional insight into, the specification ... a cause-effect graph is a good way to uncover ambiguities and incompleteness in specifications."

  Myers 1979

  "It [cause-effect graphing] is a rigorous method for transforming a natural-language specification into a formal-language specification, and exposes incompleteness and ambiguities ..."

  Kit 1997

Basic Building Blocks



An Example

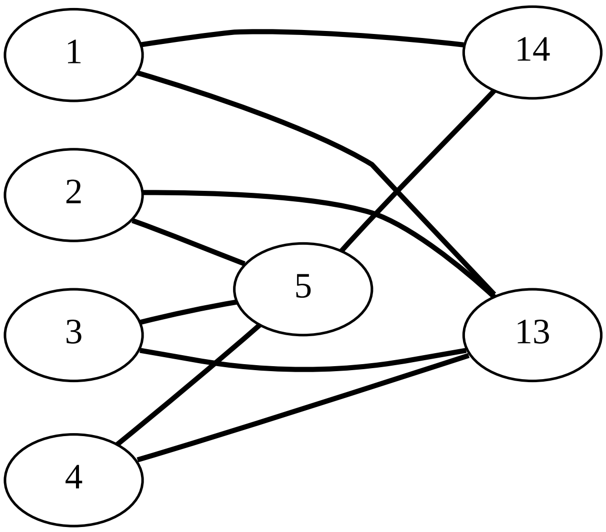
Requirements specification:

> *"If the pilot selects the reverse-thrusters and either the landing-gear is not down or not locked then the reverse-thruster command should be over-ruled. In addition, if the landing-gear is down and locked but the wheels are not spinning then the reverse-thruster command should also be over-ruled. Otherwise the reverse-thruster command should be accepted."*

Identifying Causes & Effects

- **The causes:**
  1. Reverse-thrusters selected;
  2. Landing-gear down;
  3. Landing-gear locked;
  4. Wheels spinning;
- **The effects:**
  13. Reverse-thrusters activated;
  14. Reverse-thrusters not activated;

A Cause-Effect Graph



Note: in general intermediate nodes are required.