Formal Methods at MOTOROLA's Toronto Design Center

Commercial Government Industrial Solutions Sector
MOTOROLA TORONTO DESIGN CENTER

We Don't Use Them!!

We don't build nuclear power plants or airplanes

Should we??

Is the FM 'cake' fully baked yet?

Outline

- Who we are - TDC
- What we do – CGISS products, TDC projects
- How we do requirements
- Are there opportunities for us to use Formal Methods?

TDC Background

TDC *is a regional Design Center in Motorola's* ***C****ommercial* ***G****overnment* ***I****ndustrial* ***S****olutions* ***S****ector (CGISS)*

- 150 S/W engineers
- SEI level 3 since 1995
  - **Future Goals: SEI level 4 in 2004, SEI level 5 in 2005**
- Previously: developed products for the CANADIAN market
- Now (for the last few years): in collaboration with other MOT groups contribute to our worldwide product portfolio
- Recently, existing projects have been transitioned to TDC from other Centers
  - **We inherit the project's existing processes/techniques, but have ownership of the roadmap to improve them**
- Our projects are typically 'box' s/w or subsystem s/w
- Requirements come from System Design group
  - usually natural language + MSCs
- We need to be 'cost' competitive with other Design Centers
  - defects, cycle time, $

CGISS Products

CGISS Products

We develop (radio) communications systems for private & government customers: e.g. Police, Ambulance, Fire. Including: radios, base stations, call processing 'switch', dispatcher stations.

~3K engineers, 3K staff months per system release (of new features)

System Characteristics:

- Available 99.999% of the time (unavailable 6 mins/yr):
  - > Link & box redundancy
  - > System failure scenarios important -> s/w requirements
- 'mission critical' – lives could be in danger if system fails to meet reqs (shoot/don't shoot)

Systems have long product life (~10 years)

Customers expect 'few' defects

TDC Project Background

Current types of s/w projects :

- Client GUI for radio system configuration
  - *JAVA + XML*
- Embedded real-time s/w for radios, base stations
  - *Call processing state machines*
- AutoTest tools, simulators

+ System Testing, System Design, H/W Design, & RF groups

Formal Methods?

So why don't we use formal methods?

- Ignorant of more recent work
  - Not aware of data to justify their benefits
- Not mandated by our customers
- We believe they're not applicable to our domain
  - Only for nuclear power plants, flight control, etc.
- We believe they're too complicated – require expert help to construct & expert practitioners to implement
- We believe it's faster to find defects in reviews/testing

Experiences with Semi-formal Methods

Structured Methods (Hatley Pirbhai) 10 years ago

Produced a complete, detailed requirements specification (H/W & S/W) of the MTP radio communications system

- To be used to validate customer requirements
  - in the end it was too technical for them to understand
- Very successful project – very few defects

Experiences with Semi-formal Methods

Structured Methods (Hatley Pirbhai)

- **Pros:** (i.e. Characteristics we would like to see in a Formal Method)
  - Easy to convert to an implementation model: implementation went very smoothly
  - Leveling: different levels of abstraction/detail – you could see the 'big' picture as well as the details
  - 'Fun': tools were easy to use
- **Cons:**
  - didn't know when to stop: at some point we started adding implementation to the requirements model, but we weren't aware of it

Requirements - Now

We recently completed a review of how we do requirements at TDC. To:

- improve reqs within projects
  - better techniques/methods
- improve s/w engineer competency in reqs

Some results:

S/W Engineers –

- Natural language requirements are their first choice
  - Because it's easiest (trained in English language since baby)
  - But have difficulty (because natural language is so imprecise)
- Unaware of other techniques
  - No training/schooling (only seen 'toy' examples using FMs)
- More interested in doing architecture & design

Commercial Government Industrial Solutions Sector
MOTOROLA TORONTO DESIGN CENTER

Requirements - Now

Projects

- Reqs deliverables are of inconsistent quality – reviews are important
  - Reflection of engineers' competencies
- Natural Language reqs dominate
  - Some use of UML, RUP
- Not enough time allocated to do reqs properly
  - Just using English language – shouldn't take long
- Project improvement roadmaps just being drafted
  - Desire to improve
  - Don't know what different reqs techniques are appropriate

One Outcome – TDC S/W engineer competency matrix (SWEBOK)

Formal Methods?

How do I decide if a Formal Method is the appropriate reqs technique for my project? Which Method?

Some Questions:

- What is the ROI for increased formality?
- Is there a 'spectrum' of methods with differing levels of formality that I can choose from?
- Are there case studies describing the payback for using formal methods for different problem domains?
- Is there a 'catalogue' of FMs showing the applicablity of different methods in different problem/project domains?

A Proposed 'Catalogue'

Match Project Characteristics to (Formal) Method Characteristics

Project Characteristics (ROI + S/W domain):

- Criticality (ROI)
- Product life, time to market (ROI)
- S/W size, complexity (ROI)
- Defects – cost to fix, ease of insertion (ROI)
- S/W Domain (GUI, RT, Client-Server)

A Proposed 'Catalogue'

Method Characteristics:

- Expressiveness
  - Narrow or broad domain applicability
- Degree of formality
- Maturity: industry acceptance, case studies
- Tool support
- Usability
- Development capabilities:
  - Simulatable
  - Autocode
  - Test case generation

Summary

- We (like all other companies involved in S/W development) want to use the best reqs specification technique for our product
- Are Formal Methods right for us? We don't think so, but we're not sure.
- Need:
  - Guidance & advice
  - Data to base a decision on