Improving Models for Model-based Testing using Exploratory Testing
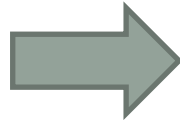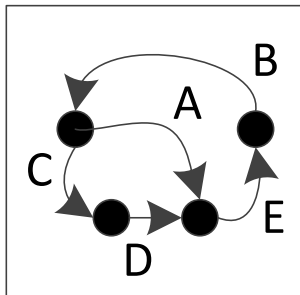
Hasan Sözer

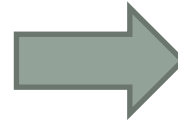together with Ceren Şahin Gebizli

Overview

- Model-based testing (MBT)
  - Automated test case generation based on models that represent the desired behavior of the system under test (SUT)

System Model

MBT Tool

Test Cases

A,E,B,A,E
A,E,B,C,D
C,D,E,B,A
C,A,E,B,A
...

- **Refinement** of models

  - by exploiting the **domain knowledge** of experienced test engineers

 - can lead to **30%** increase in the number of detected errors

Case Study: TV Software

- Types of tests performed
    - Functional tests and performance tests (4 weeks)
    - Experience based tests (2 weeks)
    - Torture tests with automation (2 weeks)

- Challenges:
    - Large code base
        - 5M LOC in total
        - ~1.5M LOC modified within 2013
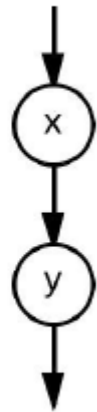   - Short time-to-market
    - Limited resources

Applied Model Based Testing Approach

- Event Sequence Graphs (ESG)
  - A formalism to express the SUT model
  - Inputs and states are represented together by assigning them to events

- Modeling Tools
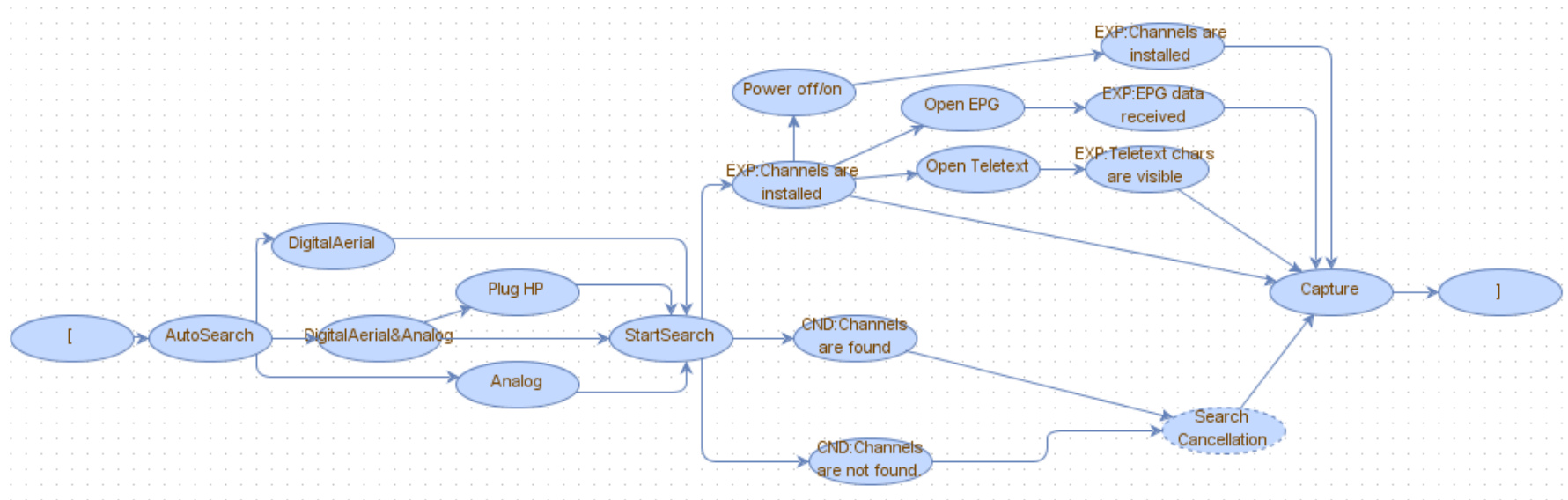 - TSD (works with ESG models)
  - MaTeLo
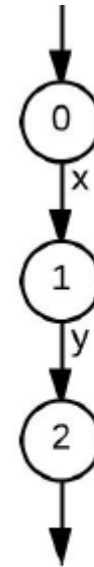
(a) FSA (b) ESG

Limitations of MBT

- Modeling is manual and error-prone
  - based on Functional Requirements Specification (FRS) documents
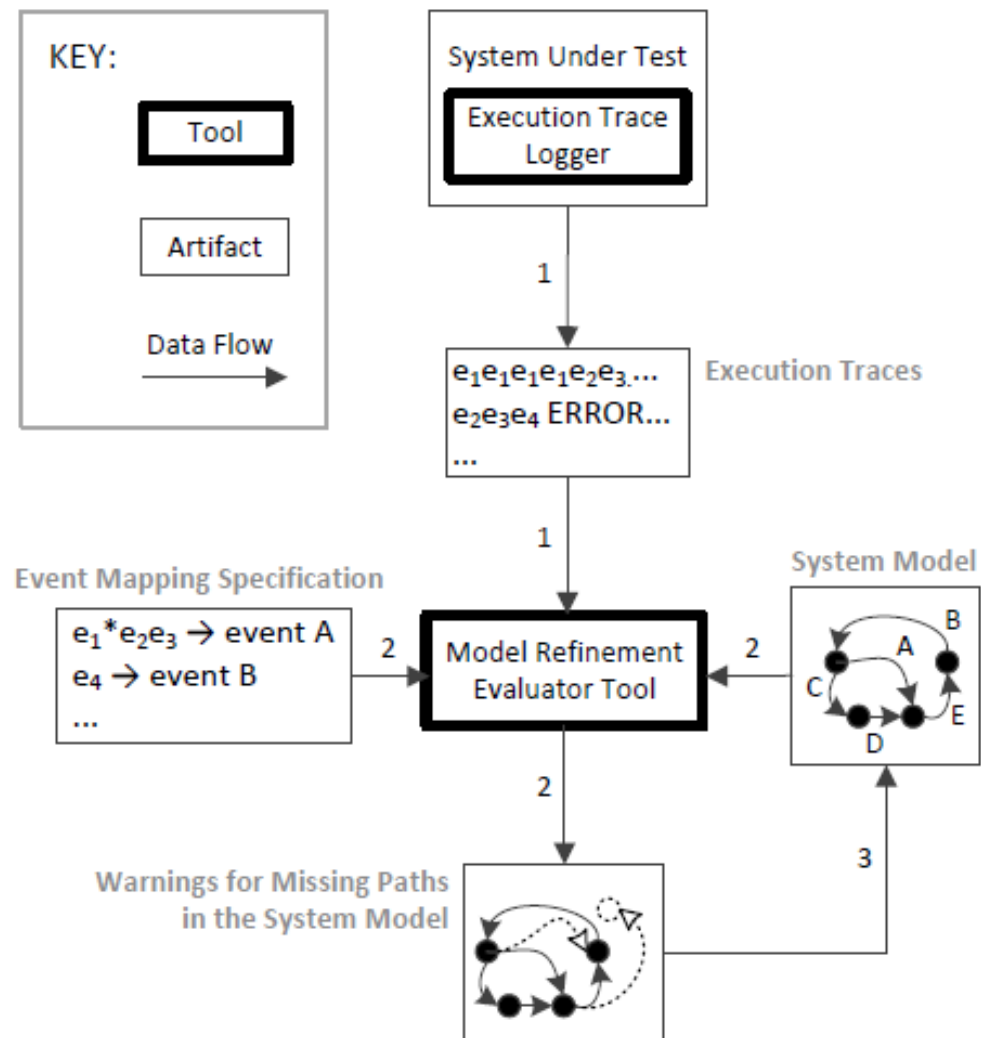
Applied Exploratory Testing Approach

- Test engineers / technicians perform manual tests

- Iterative Process
  - Learn about the product;
  - Plan the testing work to be done;
  - Design and execute the tests;
  - Report the results.

- Proved effective in detecting critical faults
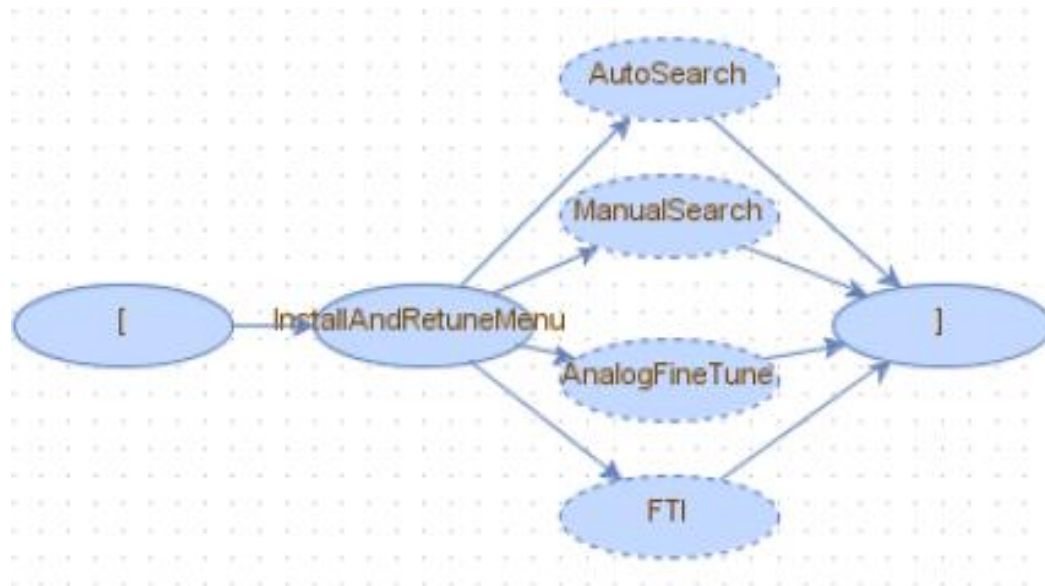- The domain knowledge and experience are not documented and shared

The Overall Approach

Industrial Case Study-1

- Digital Video Broadcasting - Terrestrial Channel Installation (DVB-TCI) module

- ESG Model of the DVB-TCI

Mapping Model Elements to Execution Traces

- Execution Traces: sequence of function calls
- Model Elements: high level events
- Mapping in the form of regular expressions

```
// <Event Sequence>        : <Model Element>
      r                    : Install And Retune;
      as                   : Auto Search;
      an,s                 : Analog;
      st                   : Start Search;
      ps                   : Press Standby;
      po                   : Press 1;
      f,l*                 : Freq=121.00MHz;
      m,p*                 : Media Video Playing;
//
```

Refinement of the DVB-TCI ESG Model

```
e4: Start search --> prgrup, prgrup : Missing destination event,
an event that
"prgrup, prgrup"
function call sequence is mapped,should be added
```

```
e4: Start search --> e1: InstallAndRetuneMenu : Missing edge,
add edge between
"e4: Start search" and "e1: InstallAndRetuneMenu"
```
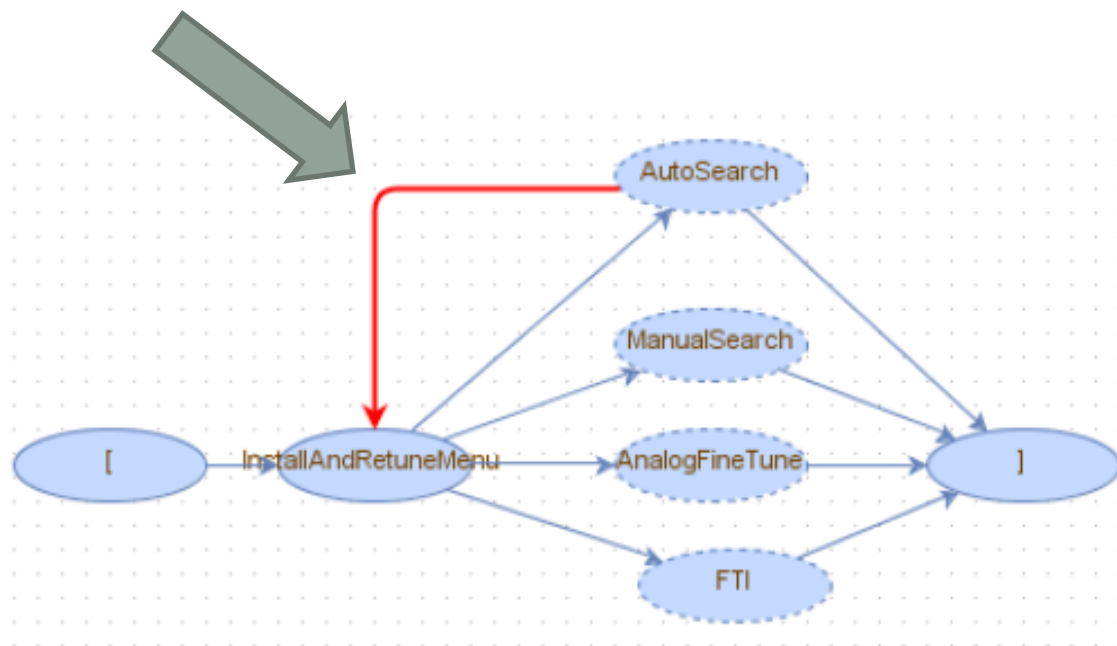
Results from Case Study-1

	# of Nodes	# of Edges	# of Test Cases	# of Faults
Before Refinement	1225	1501	217	3
Exploratory Testing	5 faults found including the 3 faults that were also found with MBT (before refinement)			
After Refinement	2012	2868	349	8

Additional (and highly critical) 3 faults revealed
- Caused the TV to reset itself
- Duplicate channels after a channel search is performed
- Caused the TV to crash

Industrial Case Study-2

- Media Browser(MB) Module

- ESG Model of the MB Module
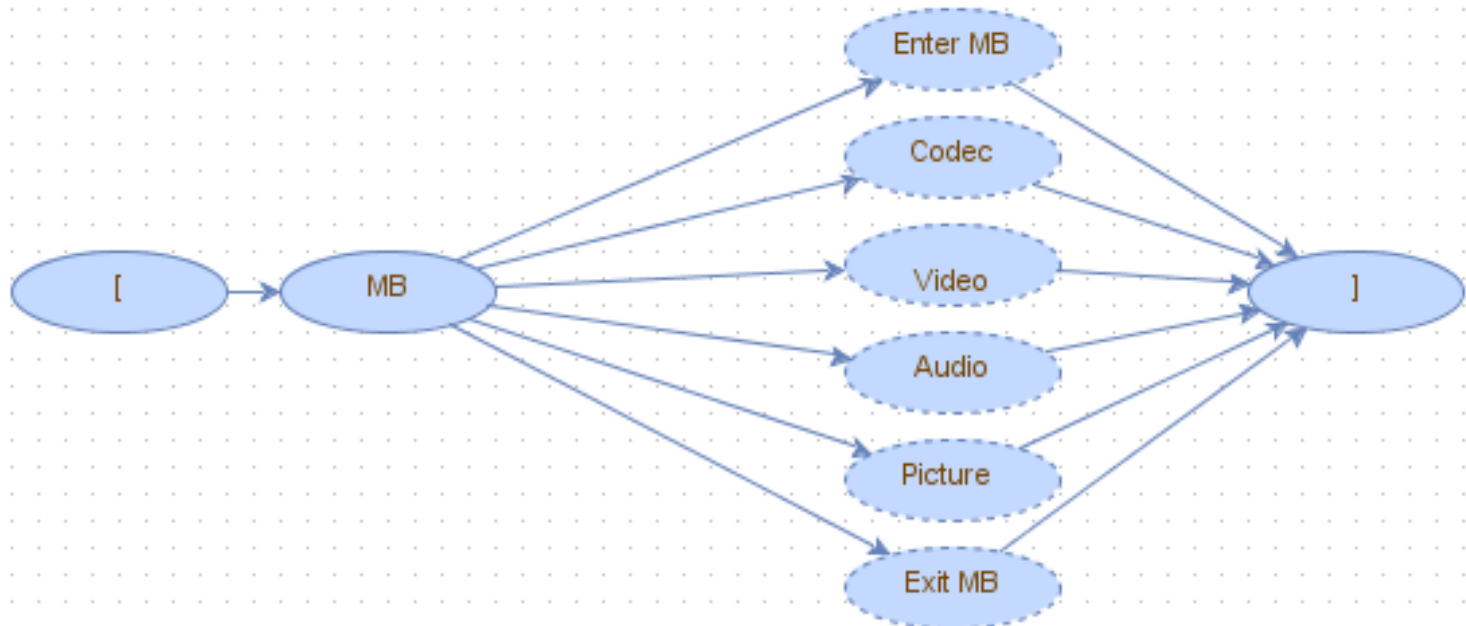
Example Warning Messages for the MB ESG Model

```
1  Tune to analog channel --> e: Enter Media Browser

2  Missing source event!

3  A new mapping must be added for

4  "Tune to analog channel"

5  function call sequence

6

7  e58: While video playing --> Press subtitle for embedded subtitle

8  Missing destination event!

9  A new mapping must be added for

10 "Press subtitle for embedded subtitle"

11 function call sequence
```

Example Refinement of the MB ESG Model
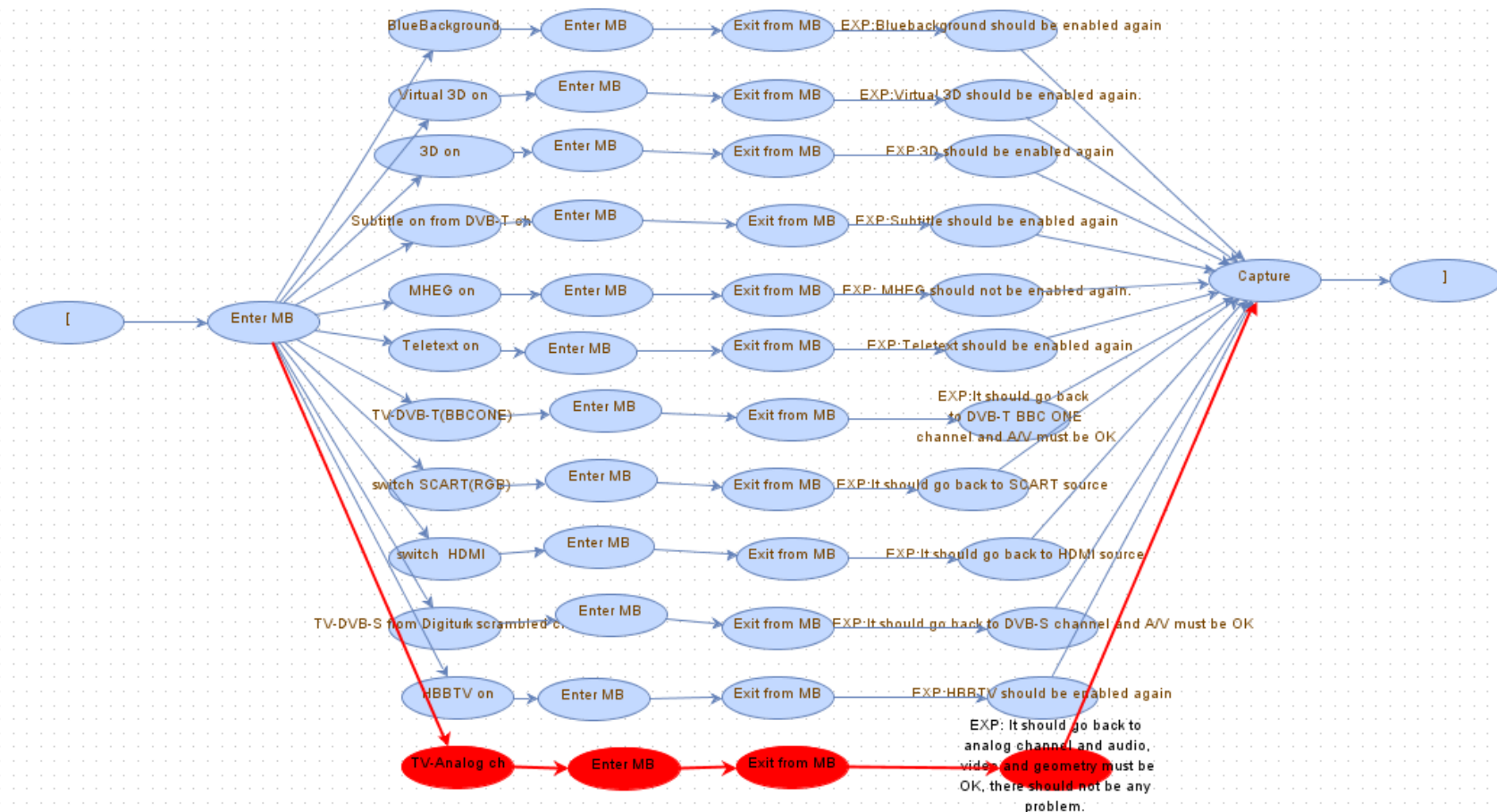
Results from Case Study-2

	# of Nodes	# of Edges	# of Test Cases	# of Faults
Before Refinement	294	424	132	36
Exploratory Testing	44 faults found including the 36 faults that were also found with MBT (before refinement)			
After Refinement	341	478	139	48

Additional (and highly critical) 4 faults revealed
- Disappear audio/video output
- Corrupted remote controllers key buffers
- Undetectable user commands
- Caused the TV to reset itself

Conclusions

- Consumer electronics domain

  - Limited resources, large code bases

- Automation is essential; MBT is employed in the industry

- Yet, critical faults are often revealed with Exploratory Testing

  - Domain knowledge and experience are valuable

- Approach: MBT coupled with Exploratory Testing activities

  - SUT models are refined based on recorded execution traces

- Real and critical faults found

  - Not detected with MBT

  - Also missed during exploratory testing