

Hierarchical Inter-Object Traces for Specification Mining

David Lo[†], Shahar Maoz[‡]

[†]Singapore Management University [‡]The Weizmann Institute of Science, Rehovot, Israel

Introduction

- Scalability and comprehensibility of results are challenges of specification mining methods
- We mine modal scenarios from execution traces [1]
- We propose the use of object hierarchies as an abstraction mechanism over event traces
- Traces at higher abstraction level are shorter and hence can be mined more efficiently
- Resultant high-level specifications can be refined to more detailed lower-level specifications at a desired abstraction level
- Result: Improved scalability & comprehensibility

Hierarchical Inter-Object Traces

An event is a triple <Caller, Callee, Method>
A trace is a series of events

An abstraction function is defined, based on the objects hierarchy. This function is then extended from objects to events to traces.

Hierarchy of Packages

```

jedi.backend.Jabber
jedi.jabber.Backend
jedi.ui.ChatWindow
jedi.ui.ChatWindows
jedi.ui.ChatSplitPane
jedi.plugins.titlescroller.Plugin
jedi.plugins.titleflash.Plugin.Flash
    
```

Focus on inter-object behavior
During abstraction: some inter-object events become intra-object and are dropped

Mining Steps

- Abstract the trace to a desired abstraction level.
- Mine LSCs from the high-level abstracted traces following the method in [1]
- Let user choose a set of mined LSCs to be refined.
- The miner refines the LSCs by adding low-level details to the mined LSCs. Order of corresponding events is preserved in the refinement.

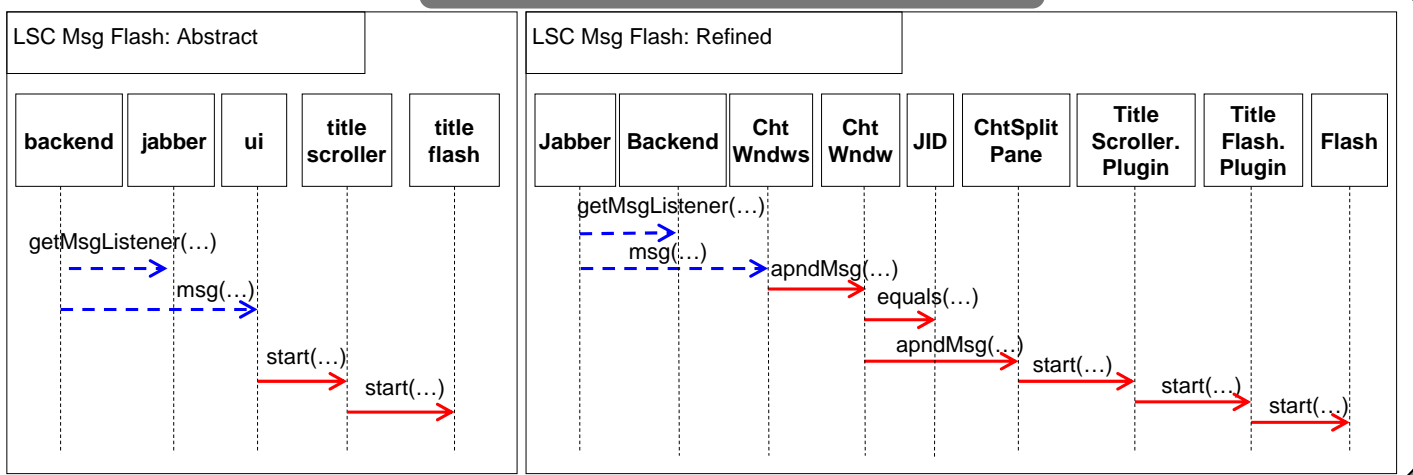
Trace Snippet – JETI Messaging App.

```

...
<backend.Jabber,jabber.Backend,getMessageListener(..)>
<backend.Jabber,ui.ChatWindows,message(..)>
<ui.ChatWindows,ui.ChatWindow,appendMessage(..)>
<ui.ChatWindow,jabber.JID,equals(..)>
<ui.ChatWindow,ui.ChatSplitPane,appendMessage(..)>
<ui.ChatSplitPane,plugins.titlescroller.Plugin,start(..)>
<ui.ChatSplitPane,plugins.titleflash.Plugin,start(..)>
<titleflash.Plugin,titleflash.Plugin.Flash,start()>
..
    
```

[variants of the above repeat many times during execution]
[object identities are implicit in the trace]

Example Mined Live Sequence Charts (LSCs)



Mined LSCs Description

- “Whenever a message is received, the chat window’s title bar starts flashing and the message’s text is scrolled in the title bar.”
- Abstract LSC: inter-package communication
- Refined LSC: inter-class communication
 - Additional events
 - More concrete participants

Future Work & Refs

- Case studies using different types of hierarchies (e.g., packages, component compositions)
 - Tool support
 - Integration with past work in [1] and [2]
- [1] D. Lo, S. Maoz and S-C. Khoo. Mining Modal Scenario-Based Specifications from Execution Traces of Reactive Systems. In ASE, 2007
[2] D. Lo, S. Maoz. Mining Scenario-Based Triggers and Effects. In ASE, 2008