Testing of inter-process communication and synchronization of ITP LoadBalancer software via model-checking

Marko van Eekelen, Stefan ten Hoedt, René Schreurs, Yaroslav S. Usenko

Aia & LaQuSo

NatTestDag 2006, ASML, Veldhoven, 17-11-2006
Outline

Aia and the Case Study

Case Analysis and Reverse-Engineering

Modeling and Analysis with the mCRL2 Toolset

Conclusions and Future Work
Aia Software BV

- Leading producer of high capability Business Document Output Solutions
- Established 1988
- Expanding partner network
- 800 customers in >25 countries
- Corporate Headquarters - Nijmegen, The Netherlands
- Regional Centers
  - Benelux
  - Germany
  - UK
  - North America
  - Switzerland
  - Australia
Aia and the Case Study

Applications of ITP

- Insurance
  - Policies
  - Endorsements
  - Renewals

- Financial Services
  - Statements
  - Correspondence
  - Contracts

- Government
  - Taxation
  - Permits
  - Correspondence

- Independent Software Vendors
Applications of ITP

- Insurance
  - Policies
  - Endorsements
  - Renewals
- Financial Services
  - Statements
  - Correspondence
  - Contracts
- Government
  - Taxation
  - Permits
  - Correspondence
- Independent Software Vendors
Applications of ITP

- Insurance
  - Policies
  - Endorsements
  - Renewals
- Financial Services
  - Statements
  - Correspondence
  - Contracts
- Government
  - Taxation
  - Permits
  - Correspondence
- Independent Software Vendors
Applications of ITP

- Insurance
  - Policies
  - Endorsements
  - Renewals
- Financial Services
  - Statements
  - Correspondence
  - Contracts
- Government
  - Taxation
  - Permits
  - Correspondence
- Independent Software Vendors

van Eekelen et. al. (Aia & LaQuSo)
ITP Document Platform

- Document Definition
- Distribution
- Model Management
- Document Assembly
- Data Access
- Integration
Intelligent Text Processing (ITP) and its Load-Balancer
Issues

- LoadBalancer does not respond at all (deadlocks)
- Free workers are not used (partial deadlocks)
- Client does not get a response (many reasons)
Artifacts

- source code in C for windows (7681 lines)
- application layer protocol documentation
- verbal information during meetings, phone and e-mail communication.
- Threads
- MutExes
- WSA
- WaitForMultipleObjects
- CallBack functions.
Artifacts

- source code in C for windows (7681 lines)
- application layer protocol documentation
- verbal information during meetings, phone and e-mail communication.

- Threads
- MutExes
- WSA
- WaitForMultipleObjects
- CallBack functions.
A Typical Use Case Scenario

Client
request to print

Client object
get data

Server object
work?

Server
work?

yes! (partners)

wake-up : get data

data

wake-up : data ready

process data
Reverse Engineering of the LoadBalancer Design

Diagram showing interactions between components C1, C2, C3, S1, S2 with socket and wake-up events.
Properties to Check

- deadlock freedom
- critical logs
- if the partner of $A$ is $B > 0$, then the partner of $B$ is $A$ or 0.
- a server may not sleep w/o a partner (except when a request is pending to it)
- limits on locking
- limits on a number of requests
mCRL2 Language

mCRL2 is based on process algebra (ACP) and algebraic (equational) data types. Specification structure:

- data types definitions (sort, func, map, rew)
- actions and communication functions definitions (act, comm)
- process definitions (proc): equations involving:
  \[ p ::= a(t) | \delta | \Upsilon(t) | p + p | p \cdot p | p \parallel p | c \rightarrow p \diamond p | c \rightarrow p + p | \sum_{d:D} p | \tau_{I}(p) | \partial_{H}(p) | \rho_{R}(p) | \nabla_{G}(p) | \Gamma_{C}(p) \]
- initial state (init).

Extensions to process algebra:

- action parameterized by data \((a(d) \mid b(e) \approx (d = e) \rightarrow c(d))\),
- \(\sum_{d:D} p\) and \(c \rightarrow x \diamond y\)
- systems of parameterized recursion equations.
The mCRL2 Toolset (www.mcrl2.org)

- **Specification Text**
  - mCRL2
    - LPE
      - formula checking
        - formcheck (eq-BDD-based prover)
        - confcheck
        - invcheck
      - simulation
        - sim
        - xsim
        - lpe2torx
    - FLTS
      - ltsmin (comparison, minimization)
      - ltsupdate (renaming of actions)
  - optimization
    - mcr122lpe
    - lpepp
  - model checking
    - lpe2pbes
    - lpe2lts
      - (state-space generation, deadlock checking, action search, confluence reduction)
  - visualisation
    - ltsview
    - noodleview
  - evaluator (CADP)
    - lperewr
    - lpeconstelm
    - lpeparelm
    - lpedataelm
    - lpesumelm
    - lpeconfelm
    - lpeinvelm
    - lpeupdate
    - lpestructelm

van Eekelen et. al. (Aia & LaQuSo)

Testing ITP LoadBalancer

17-11-2006 13/18
Experiments on a 3Ghz 32 bit machine with 4Gb RAM.

<table>
<thead>
<tr>
<th>#clients</th>
<th>#servers</th>
<th>time</th>
<th>#levels</th>
<th>#states</th>
<th>#transitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>7m 38s</td>
<td>241</td>
<td>657k</td>
<td>1.38M</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3h 01m</td>
<td>367</td>
<td>18M</td>
<td>38.5M</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>9h 55m</td>
<td>444</td>
<td>54M</td>
<td>141M</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>13h*</td>
<td>481</td>
<td>213M</td>
<td>465.5M</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>&gt;113h*</td>
<td>&gt;215</td>
<td>&gt;511M</td>
<td>&gt;1121M</td>
</tr>
</tbody>
</table>

* On a cluster of 32 64-bit machines, 1Gb each.
Detected Issues

- partner links inconsistent
  - set partner to 0 was forgotten for one of the parties
  - found by model-code comparison
  - confirmed to be a problem by model-checking

- server sleeping w/o a partner
  1. set client’s partner link to 0 before waking up the server
  2. forgotten to wake up the server
    - 1st found by model-code comparison, 2nd by model-checking

- critical logs could occur
  1. sending request for disconnect to itself happened in a wrong state (forgot to change the state)
  2. request to wake up can lead to an inappropriate state change when server disconnects (not critical)

- number of requests exceeds the limit
  - server sends request for disconnect to the client and does not break the partnership afterwards
Detected Issues

- partner links inconsistent
  - set partner to 0 was forgotten for one of the parties
  - found by model-code comparison
  - confirmed to be a problem by model-checking

- server sleeping w/o a partner
  1. set client’s partner link to 0 before waking up the server
  2. forgotten to wake up the server
     - 1st found by model-code comparison, 2nd by model-checking

- critical logs could occur
  1. sending request for disconnect to itself happened in a wrong state (forgot to change the state)
  2. request to wake up can lead to an inappropriate state change when server disconnects (not critical)

- number of requests exceeds the limit
  - server sends request for disconnect to the client and does not break the partnership afterwards
Detected Issues

► partner links inconsistent
  ▶ set partner to 0 was forgotten for one of the parties
  ▶ found by model-code comparison
  ▶ confirmed to be a problem by model-checking

► server sleeping w/o a partner
  1 set client’s partner link to 0 before waking up the server
  2 forgotten to wake up the server
    ▶ 1st found by model-code comparison, 2nd by model-checking

► critical logs could occur
  1 sending request for disconnect to itself happened in a wrong state
    (forgot to change the state)
  2 request to wake up can lead to an inappropriate state change when server disconnects (not critical)

► number of requests exceeds the limit
  ▶ server sends request for disconnect to the client and does not break the partnership afterwards
Detected Issues

- partner links inconsistent
  - set partner to 0 was forgotten for one of the parties
  - found by model-code comparison
  - confirmed to be a problem by model-checking

- server sleeping w/o a partner
  1. set client's partner link to 0 before waking up the server
  2. forgotten to wake up the server
     - 1st found by model-code comparison, 2nd by model-checking

- critical logs could occur
  1. sending request for disconnect to itself happened in a wrong state
     (forgot to change the state)
  2. request to wake up can lead to an inappropriate state change when server disconnects (not critical)

- number of requests exceeds the limit
  - server sends request for disconnect to the client and does not break the partnership afterwards
A Faulty Scenario

Client
request to print

Client object
get data

Server object
work?

Server
work?

yes! (partners)

wake-up : get data

wake-up : client went away

disconnect
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Conclusions

▶ Session layer of Load Balancer is modelled
▶ A number of properties are verified
▶ Number of issues discovered, communicated and corrected
▶ Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
▶ Case with 2 clients and 2 servers was partially analyzed
▶ Modification of the model and further analysis are possible
▶ Reverse engineering of the model took most of the time
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Conclusions

- Session layer of Load Balancer is modelled
- A number of properties are verified
- Number of issues discovered, communicated and corrected
- Cases up to 1 client and 3 servers and 2 clients and 1 server were fully analyzed
- Case with 2 clients and 2 servers was partially analyzed
- Modification of the model and further analysis are possible
- Reverse engineering of the model took most of the time
Future Work

- Certification of the LoadBalancer software
  - For the session layer: more properties
  - For the application layer: model/code generation.
Future Work

- Certification of the LoadBalancer software
- For the session layer: more properties
- For the application layer: model/code generation.
Future Work

- Certification of the LoadBalancer software
- For the session layer: more properties
- For the application layer: model/code generation.