TLAPLUS

留言

歸檔的貼文。您不能於此貼文評分或留言。

4

TLA+tools to generate test cases on implemenetation (aka Specification Driven Development Demo)

(jarjuk.github.io)

jarjuk 於 8 月前 發表

4 留言 分享 儲存 隱藏 贈送金幣 檢舉

crosspost

所有 4 則留言

排序依據: 最佳 ▾

[–] jarjuk S 1 指標 8 月前

Synopsis: I have been developing a tool called sbuilder, which produces formal models in TLA+ language representing "business IT" systems, i.e. application with interfaces, persistent state and service semantics between these two.

The idea is to use a formal model to generate "virtual system tests". The work flow:

1) Create a formal model + with correctness invariant (assisted by sbuilder translation) 2) Run model checking on formal model in the "usual" way 3) Run model checking using a configuration, which produces a counter example, which represent execution we would like to test the system with. 4) Parse the counter example and extract interface calls steps: a) state of the formal model before interface call, b) formal model parameter binding to the interface call, c) formal model interface response message, d) formal model state after the call 5) iterate each interface step in the trace and execute it as unit test a) map formal model before

本文發表於 10 Feb 2018

4 指標 (84% 好評)

shortlink: https://redd.it/7wnsyu

發表新連結

發表新文章

TLAPLUS

停止訂閱 385 讀者 1 人在這裡

在這個看板上顯示我的使用者標籤。標籤顯示為:

hengxin

/r/tlaplus is a place for discussion about TLA+ (and related topics like synchronous programming), formal methods, software specification and software correctness in general.

COMMUNITY

- The TLA+ User Group
- @tlaplus

RESOURCES

- The Hyperbook
- The TLA+ Video Course
- Specifying Systems
- learntla.com
- TLA+ Examples
- Dr. TLA+ Series
- TLA papers by Leslie Lamport
- TLA+ in Practice and Theory
- The TLA+ Homepage

GUIDELINES

state to implementation state && init SUT with that state b) map formal model interface parameter bindings to a request in implementation state && send that req. msg to SUT c) receive response msg from SUT and compare it with formal model response message mapped to implementation state d) map formal model after state top implementation && check SUT state 6) Interpret the aggregate result of executing individual unit tests as a virtual system test

As the result: - expect saving in "virtual" system testing, because executing unit test is considerable easier than trying to manage execution of a system test as a single unit - formal model and implementation conformance increased

永久連結 embed 儲存 贈送金幣

[–] **pron98** 1 指標 7 月前

Your work seems very interesting. Even though the idea of generating tests from model-checker counterexamples is not new, it's great to see it implemented using TLA$^+$. However, your text is very hard to follow. It employs both too many buzzwords, and has too little in way of explanation. If you can write in greater detail it will be easier to appreciate your work.

永久連結 embed 儲存 上層留言 贈送金幣

[–] jarjuk S 1 指標 7 月前

I know I have trouble in expressing my thoughts (failing to balance between being too verbose and too succinct ;) - especially, because, I confess, I do not fully understand the problem I am trying to solve with this sbuilder stuff.

However, yet another try: https://jarjuk.wordpress.com/2018/03/12/simple-sdd-demo/

And,sure enough - it also ends with more questions: This blog post has used “Simple ‘Specification Driven Development’ Demo” to point out several issues, which may rise, when trying to apply Specification Driven Development for real cases. It is left for further blog post(s) to cover, how to solve these issues.

永久連結 embed 儲存 上層留言 贈送金幣

[–] jarjuk S 1 指標 7 月前*

To create counter examples, referenced above, sbuilder offers "possibility" operators, which are implemented on top of TLA+tools extension feature. A possibility operator translates to an extension module, which extends base module holding formal model of the application. The extension module

- Please follow proper reddiquette.

created by pron98 社群已經成立 1 年

寄送訊息給版主

板主

pron98

關於管理員團隊 »

最近瀏覽連結

Program Synthesis with TLA+
7 指標 | 留言

Modern SAT solvers: fast, neat and underused (part 1 of N)
7 指標 | 留言

Decision Tables
5 指標 | 留言

Modeling, refinement, and verification
4 指標 | 留言

Generalized data structure synthesis
3 指標 | 留言

清除

帳號活動

CSS via /r/Structura by /u/Cereal_Addict

requires negation of "possibility" operator to hold universally. In step 2 above, we would be model checking using the base module, and in step 3, using the extension module. Refer to https://jarjuk.wordpress.com/2017/12/06/sbuilder-wallet-1/#orgheadline3 blog entry for more details.

永久連結 embed 儲存 贈送金幣