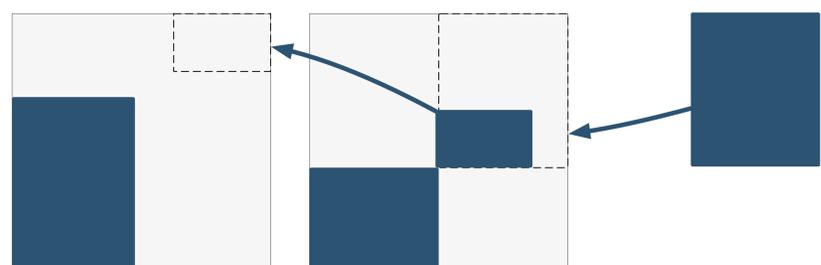


TRUSTABLE VM SCHEDULING IN A CLOUD

Fabien Hermenier
Ludovic Henrio



The scheduler deploys VM to servers according to SLOs which state through **constraints** the awaited performance, availability, placement requirements, etc. A scheduler that behaves as expected leads to low running costs and higher user confidence.



VM schedulers have defects

Peer review, unit & smoke testing do not counteract reasoning issues.

Issue trackers report un-anticipated state transitions or event ordering and partial logic understanding.

Causes & consequences

over-filtering deny solutions and reduce the hosting capacity.

crashes introduce delay and reduce user confidence.

under-filtering lead to decisions that violates constraints and reduce user confidence.

Constraint specification

- ▶ state acceptable (re)configurations
- ▶ augmented first order logic
- ▶ business functions in native code
- ▶ temporal call to reason on the history
- ▶ integration through code annotation

```
RunningCapacity(ns <: nodes, nb : int) ::=
  sum({card(running(n)). n : ns}) <= nb

MaxOnline(ns <: nodes, nb : int) ::=
  card({i. i:ns , nodeState(i)=online}) <= nb

Among(vs <: vms, parts <<: nodes) ::=
  ?(g : parts)
  {host(i). i : vs, vmState(i) = running} <: g

ShareableResource(id : string) ::=
  !(n : nodes)
  sum([cons(v, id). v : host(n)]) <= capa(n, id)
```

Implementation checker

- ▶ fuzzed test cases to avoid bias
- ▶ simulator + spec as an oracle
- ▶ implementation vs. oracle to report inconsistencies

```
@CstrTest()
public void testMaxOnline(TestCampaign c) {
  // The constraint
  c.check("maxOnline");

  // Fuzzer configuration
  c.vms(10).srcVMs(1, 9, 0).with("nb", 0, 7);

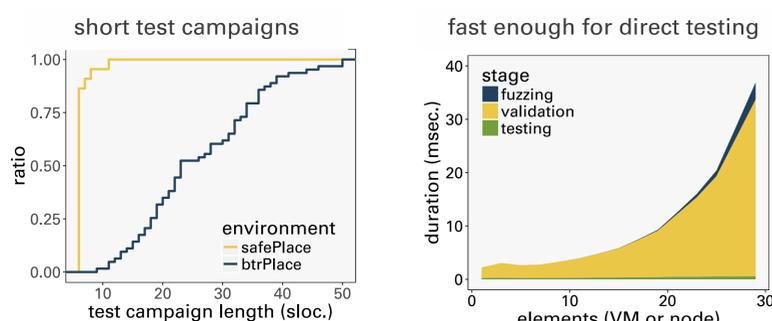
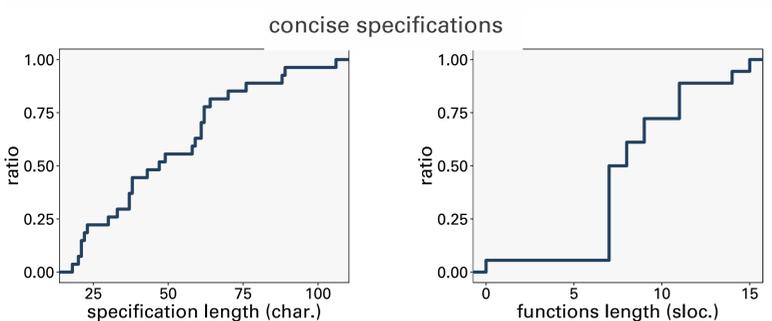
  // Scheduler configuration
  c.schedulerParams().doRepair(true);

  // Test configuration
  c.limits().tests(10000).failures(1);
}
```



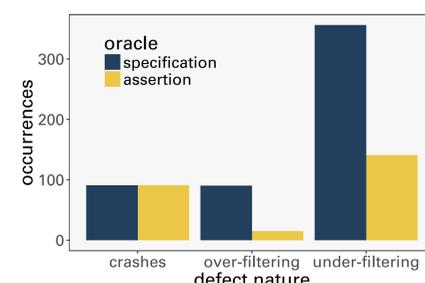
Usability

BtrPlace constraints specified. Suitable for OpenStack & VMWare DRS.



Debugging BtrPlace

DEFECT CAUSE	CONSTRAINTS	TESTS	CONSEQUENCE	CONSTRAINTS	TESTS
Initial violation in continuous mode	7	704	Under-filtering	10	938
Unexpected arguments	4	642	Crashes	3	459
Discrete filtering in continuous mode	3	45	Over-filtering	6	244
Unsupported action synchronisation	4	20			
Bad action semantic comprehension	1	16			
Unconsidered initial element state	1	4			



SafePlace outperforms BtrPlace assertion system

Read more in the paper.
Check out BtrPlace website.
Contact: fabien.hermenier@nutanix.com

