

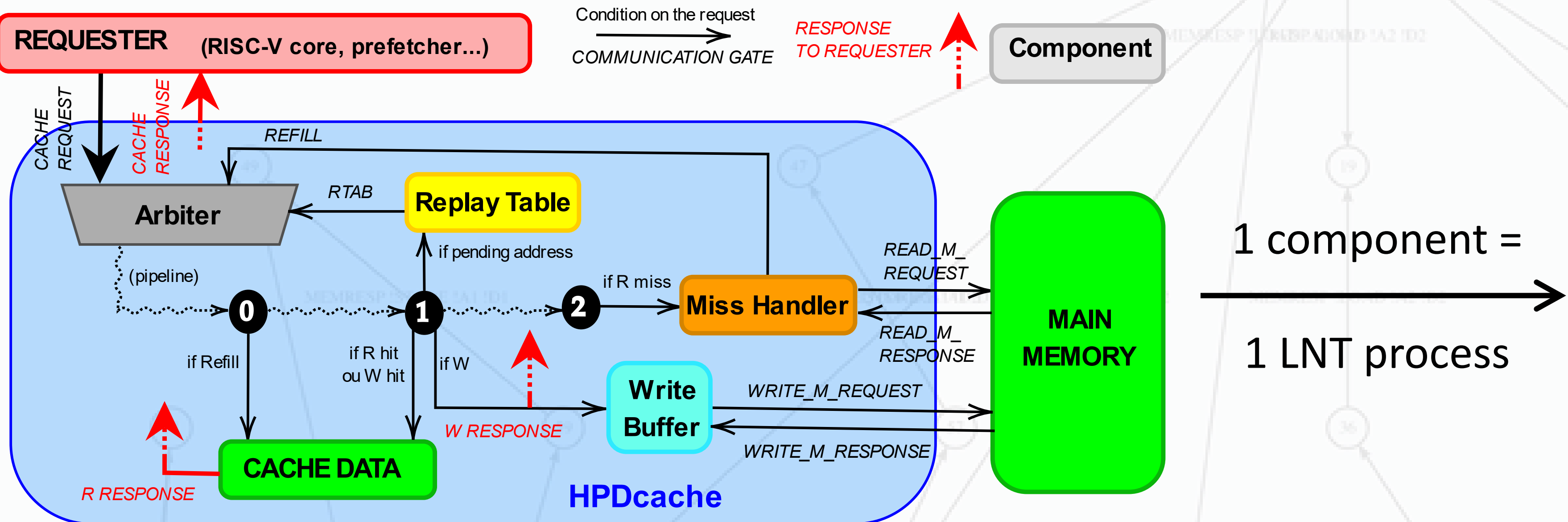
# On Benefits of Modeling the HPDcache in LNT

Zachary Assoumani — César Fuguet — Radu Mateescu — Wendelin Serwe  
Inria Center at Université Grenoble Alpes



<https://github.com/openhwgroup/cv-hpdcache/>

## LNT MODEL



```
-- The full HPDcache, with its internal components.

process HPDCACHE [CRI_REQ, CRI_RSP_R, CRI_RSP_W,
  CMI_REQ_R, CMI_REQ_W, CMI_RSP_R, CMI_RSP_W,
  RTAB_REQ, REFILL_REQ,
  CONTROLLER_CACHE, CONTROLLER_RTAB, CONTROLLER_WRITEBUFFER,
  CONTROLLER_MSHR: Wire,
  CHECK: Checkgate, WBUF_NOTIF, MSHR_NOTIF: Notifgate,
  DEBUG: any] is
  par CHECK in
    CRI_REQ, CMI_REQ_R, CMI_REQ_W, CMI_RSP_W, CMI_RSP_R,
    RTAB_REQ, REFILL_REQ, CONTROLLER_CACHE, CONTROLLER_RTAB,
    CONTROLLER_WRITEBUFFER, CONTROLLER_MSHR ->
    CONTROLLER [...]
  |
    CONTROLLER_CACHE ->
    CACHEDATA [CONTROLLER_CACHE, CRI_RSP_R, CHECK, DEBUG]
  |
    CONTROLLER_WRITEBUFFER, CMI_REQ_W, CMI_RSP_W, WBUF_NOTIF ->
    WRITEBUFFER [...]
  |
    CONTROLLER_MSHR, REFILL_REQ, CMI_REQ_R, CMI_RSP_R, MSHR_NOTIF ->
    MSHR [...]
  |
    CONTROLLER_RTAB, RTAB_REQ, CRI_REQ, WBUF_NOTIF, MSHR_NOTIF ->
    RTAB [...]
  end par
end process
```

```
int rtab_find_ready(int last) {
  int i = (last + 1) mod % RTAB_NENTRIES;
  for (;;) {
    if (rtab[i].valid && rtab[i].ll_head &&
        (rtab[i].deps == 0))
      return i;
    if (i == last)
      return -1;
    i := (i + 1) % RTAB_NENTRIES;
  }
}
```

pseudocode function...

...to LNT function

```
function rtab_find_ready(rtab:RTAB_Array, last:int): int is
  var i:int in i := (last + 1) mod RTAB_NENTRIES;
  loop
    if rtab[Nat (i)].valid and rtab[Nat (i)].ll_head and
      (rtab[Nat (i)].deps == NoDeps) then
      return i
    elsif i == last then
      return -1
    end if;
    i := (i + 1) mod RTAB_NENTRIES
  end loop
end var end function
```

## BENEFITS: FORMAL ANALYSIS WITH CADP



ETAPS Test-of-Time Tool  
Award Winner in 2023

### Model checking (EVALUATOR)

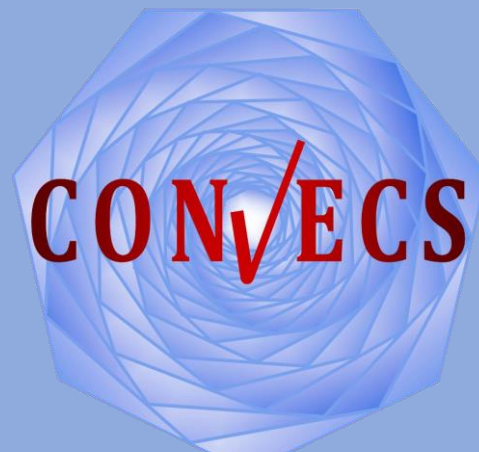
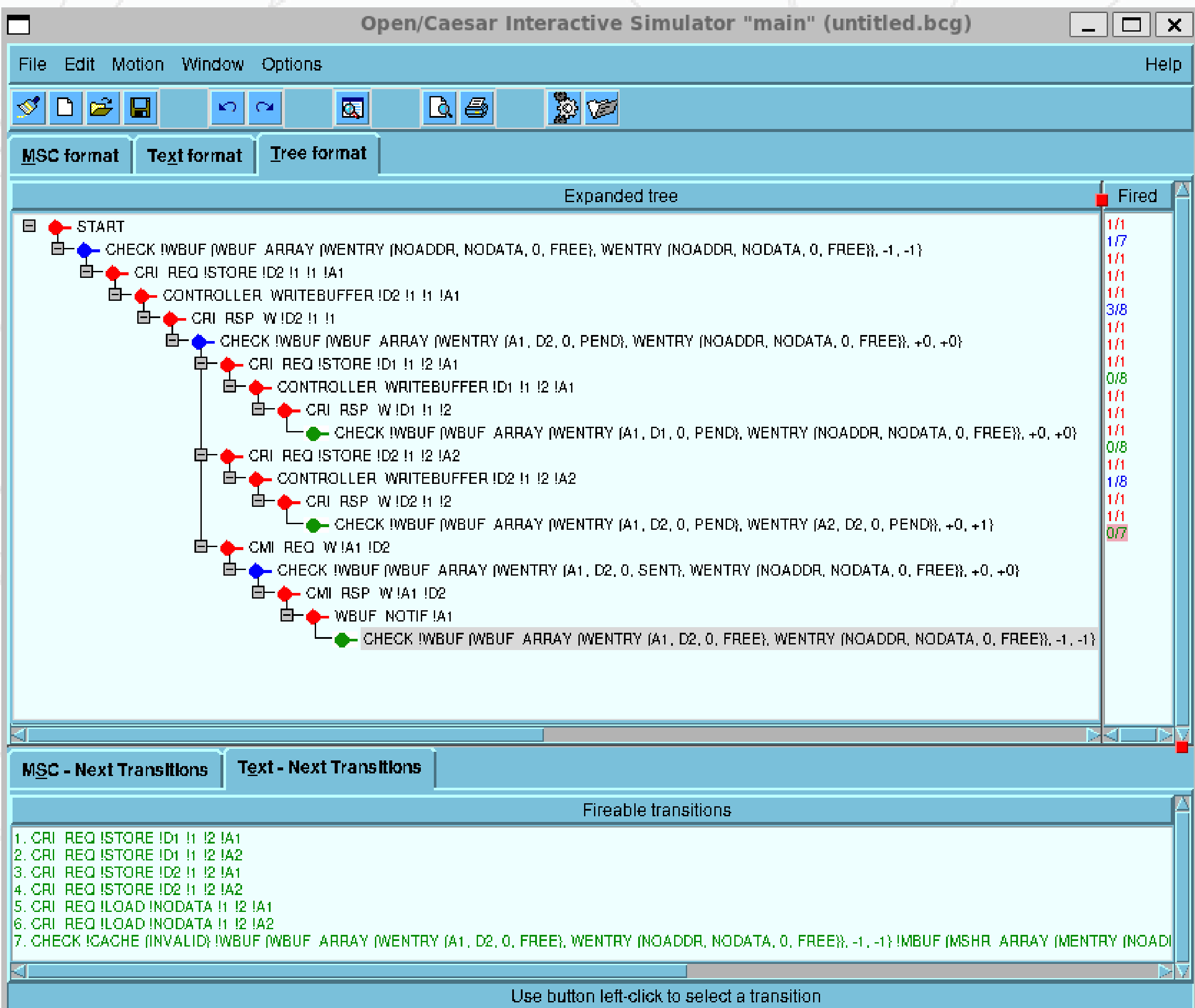
Temporal logic formula: RVWMO safety property

```
[ true*.
  {REQ ?any ?s:nat ?t1:nat ?a:string}.
  not ({RSP_W ?any !s !t1} or {RSP_R ?any !s !t1})*.
  {REQ ?any ?any !s ?t2:nat !a where t1 <> t2}.
  not ({RSP_W ?any !s !t1} or {RSP_R ?any !s !t1})*.
  {RSP_W ?any !s !t2} or {RSP_R ?any !s !t2}
] false
```

### Counterexample trace

<initial state>  
...  
"CRI\_REQ !STORE !D2 !1 !1 !A2"  
"CRI\_RSP\_W !D2 !1 !1"  
"CMI\_REQ\_W !A2 !D2"  
"CRI\_REQ !STORE !D2 !1 !2 !A1"  
"CRI\_RSP\_W !D2 !1 !2"  
"CRI\_REQ !STORE !D2 !1 !3 !A1"  
"CRI\_REQ !STORE !D2 !1 !4 !A1"  
"CMI\_RSP\_W !A2 !D2"  
"CMI\_REQ\_W !A1 !D2"  
"RTAB\_REQ !STORE !D2 !1 !3 !A1"  
"CMI\_RSP\_W !A1 !D2"  
"RTAB\_REQ !STORE !D2 !1 !4 !A1"  
"CRI\_RSP\_W !D2 !1 !4"  
<goal state>

### Interactive step-by-step simulation (OCIS)



PROGRAMME  
DE RECHERCHE  
CLOUD

