

Vulcan: Hardware Support for Detecting Sequential Consistency Violations Dynamically

Abdullah Muzahid, Shanxiang Qi, and Josep Torrellas

University of Illinois at Urbana-Champaign

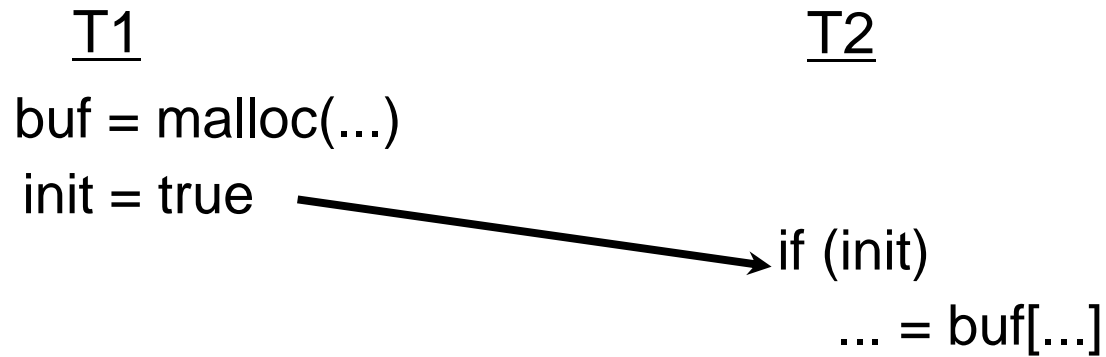
<http://iacoma.cs.uiuc.edu>



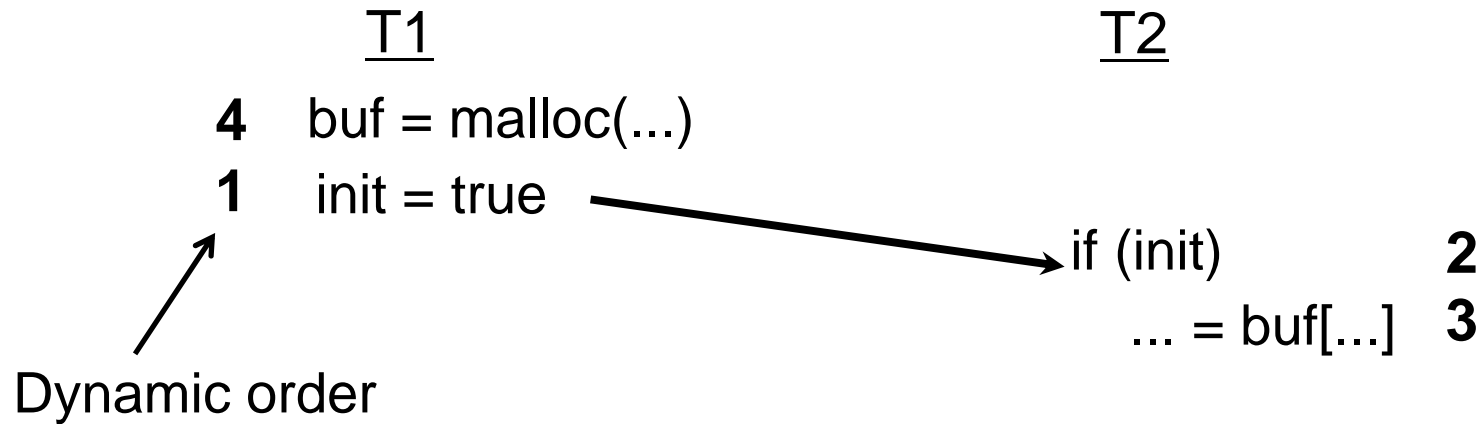
MICRO
December 2012



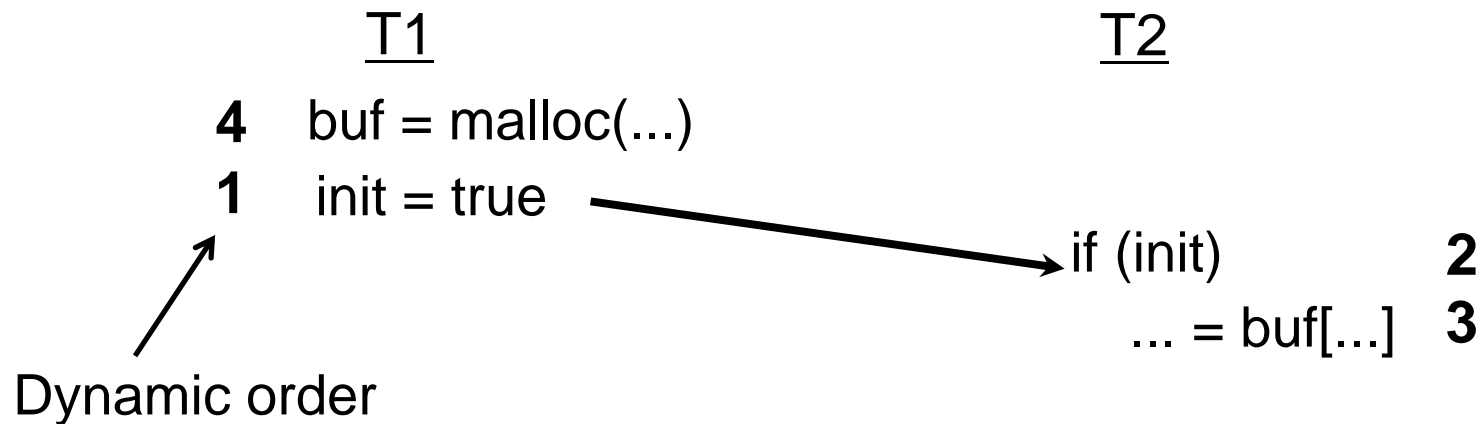
Sequential Consistency Violation (SCV)



Sequential Consistency Violation (SCV)



Sequential Consistency Violation (SCV)

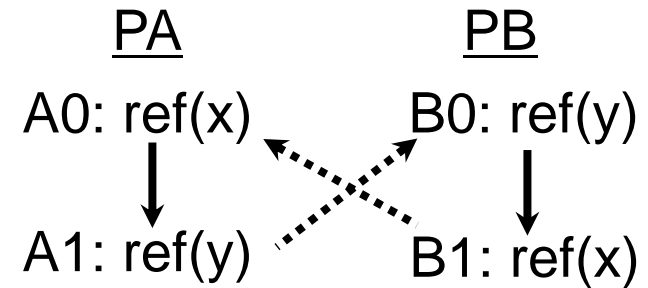


- SCV: Memory accesses get reordered unintuitively
- Virtually always a bug
- 18% of all the reported data races fall in this category

How Vulcan Detects SCVs

SCV requires:

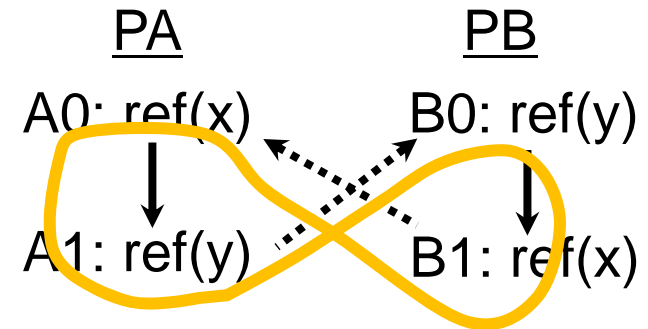
- Two or more data races overlap
- They create a cycle



How Vulcan Detects SCVs

SCV requires:

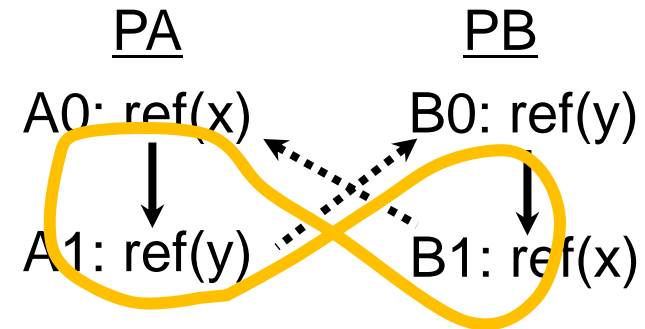
- Two or more data races overlap
- They create a cycle



How Vulcan Detects SCVs

SCV requires:

- Two or more data races overlap
- They create a cycle



Vulcan approach:

- Uses HW to check if memory dependence orders **form a cycle**
- Leverages the cache coherence protocol messages
- Detects SC violations on the fly → exception

Results

- Detects SCVs in relaxed consistency machines highly **precisely**
- Detects **3 new SCV bugs** in popular codes:
 - Pthread library, Crypt library, and fmm
- Only has 4% -- 6% execution overhead
- Usable for on-the-fly deployment
- Uses no other input than executable