BRAVO –
Biased Locking for Reader-Writer Locks

Dave Dice Alex Kogan
Oracle Labs

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Reader-Writer Locks

- Allow shared access for read-only use of a resource
- Ubiquitous in modern systems

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

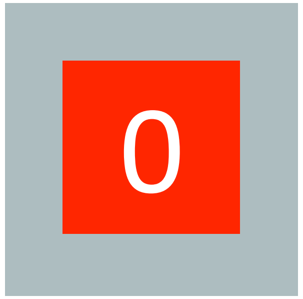
Reader-Writer Locks

- Allow shared access for read-only use of a resource
- Ubiquitous in modern systems

- Have to keep track of the presence of active readers

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

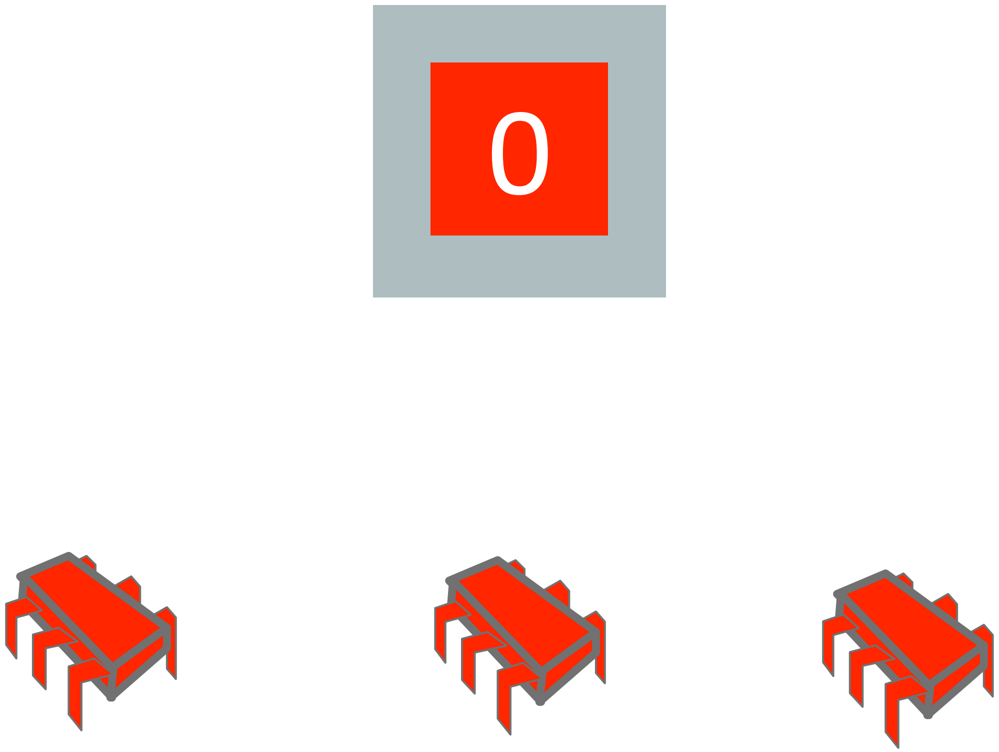
The “shared counter” approach



0

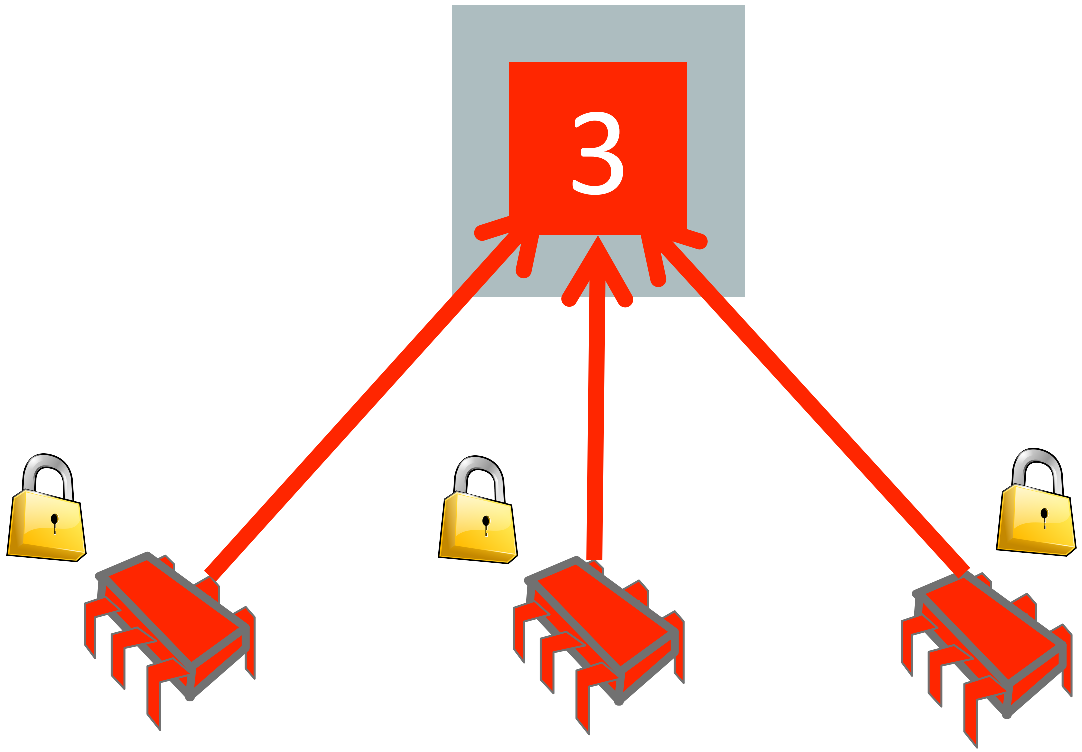
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The “shared counter” approach

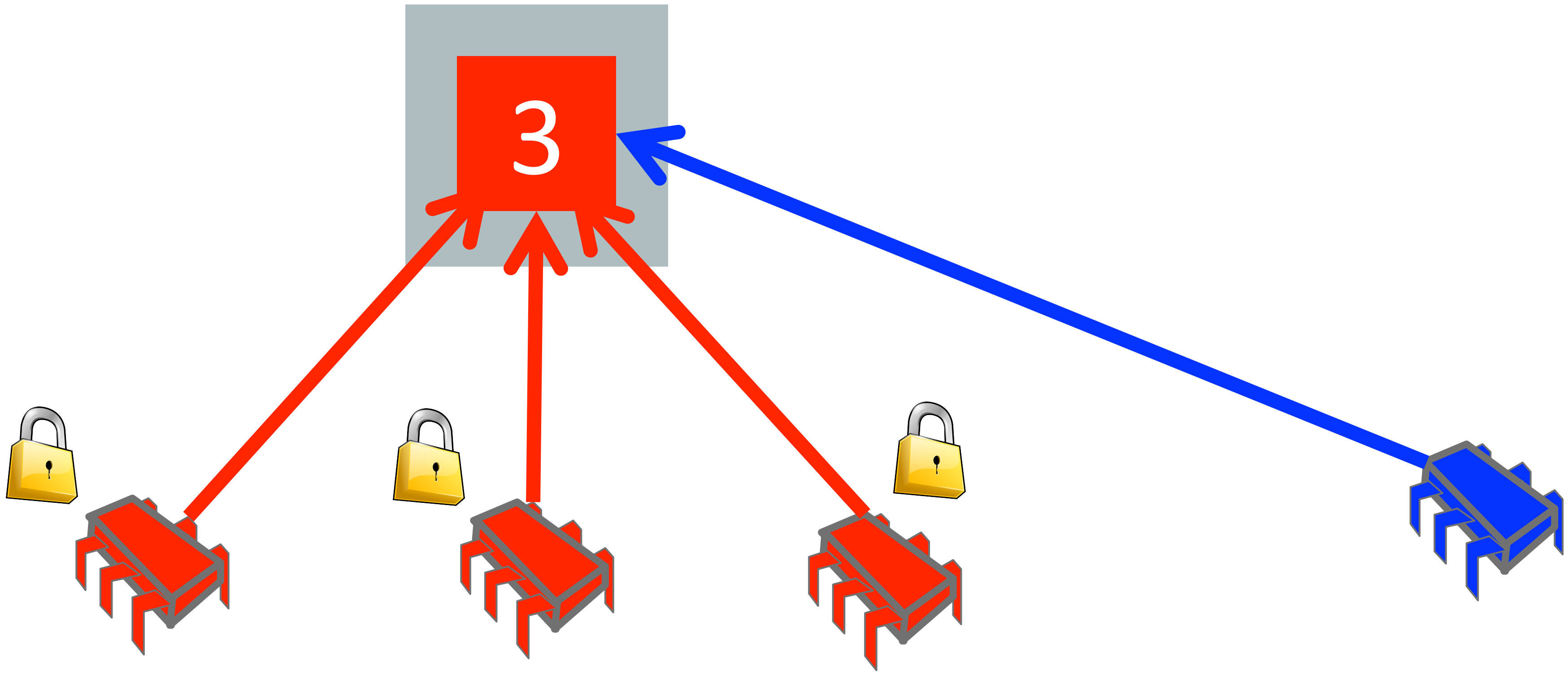


Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The “shared counter” approach

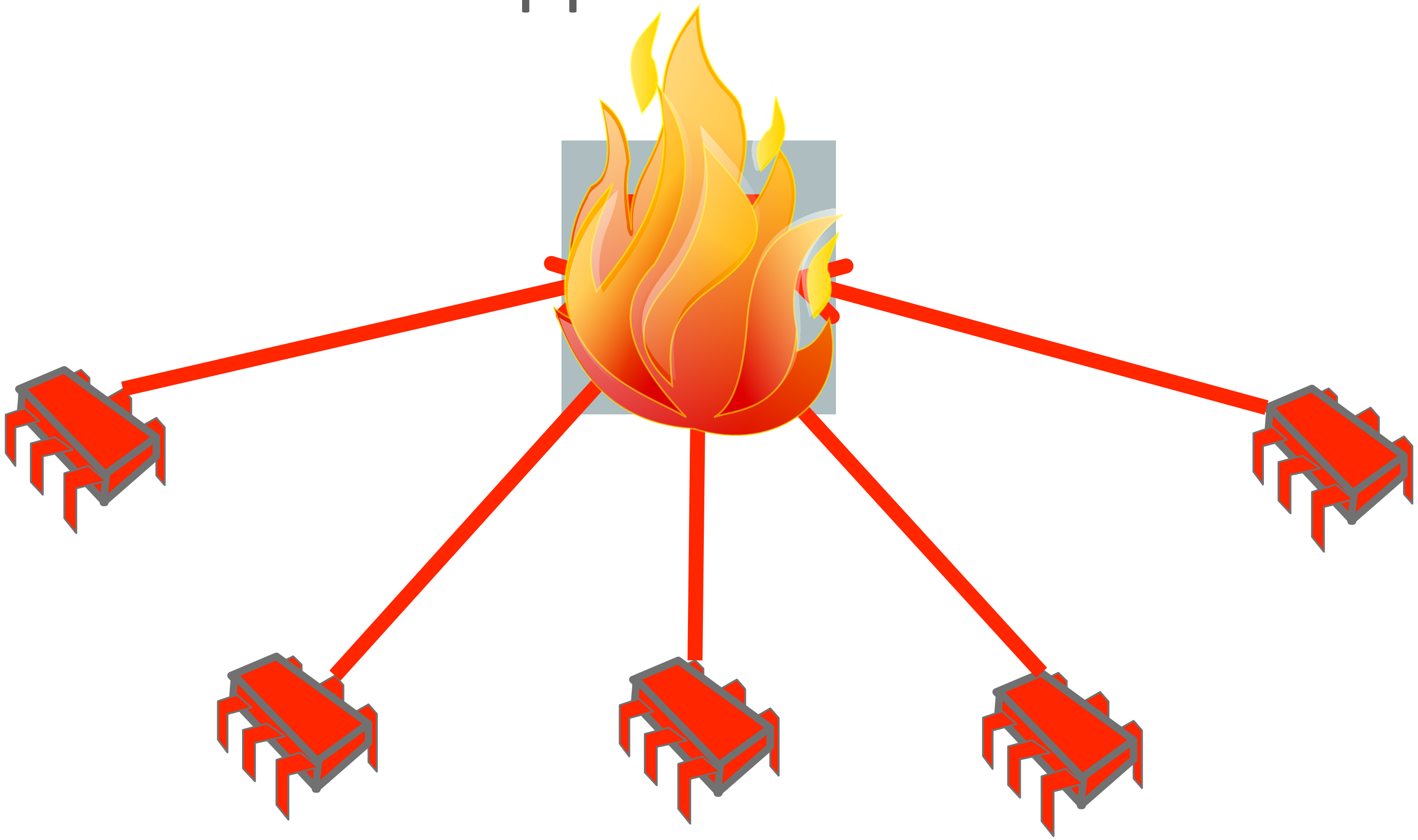


The “shared counter” approach

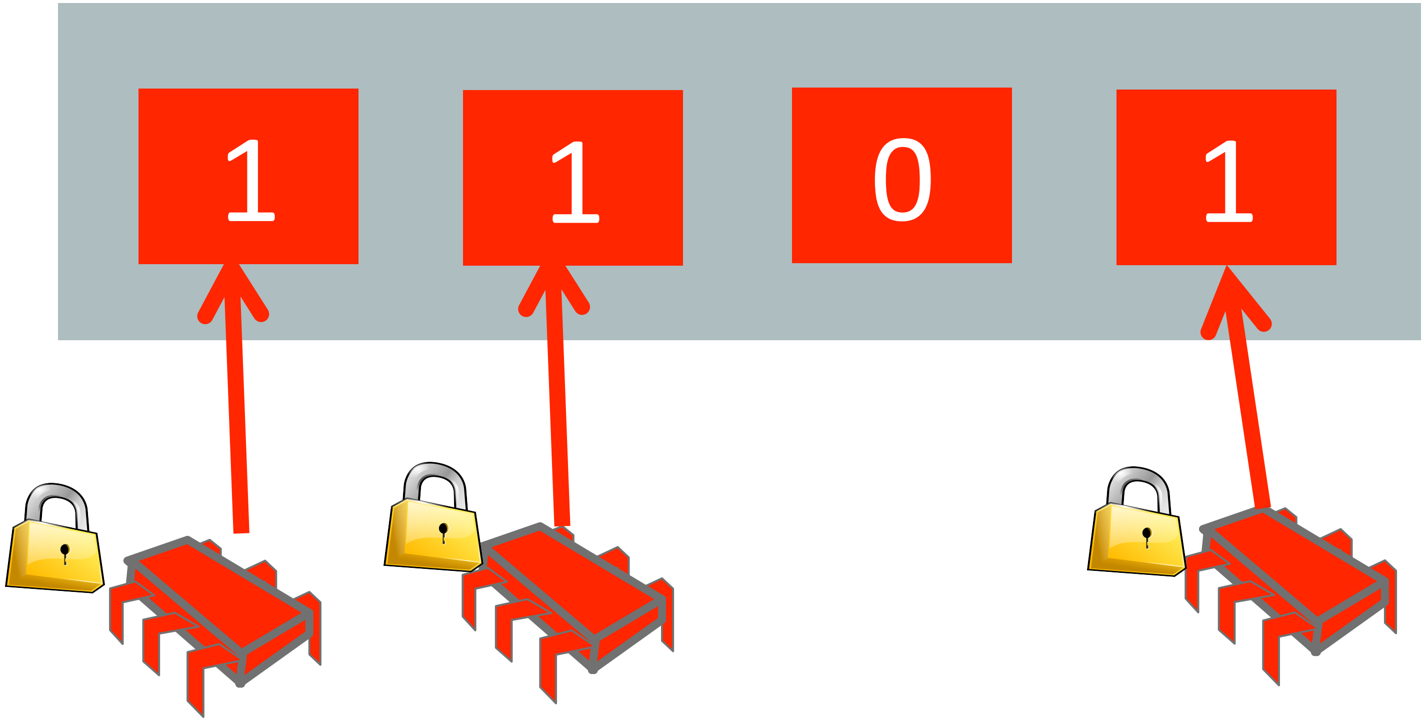


Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The “shared counter” approach

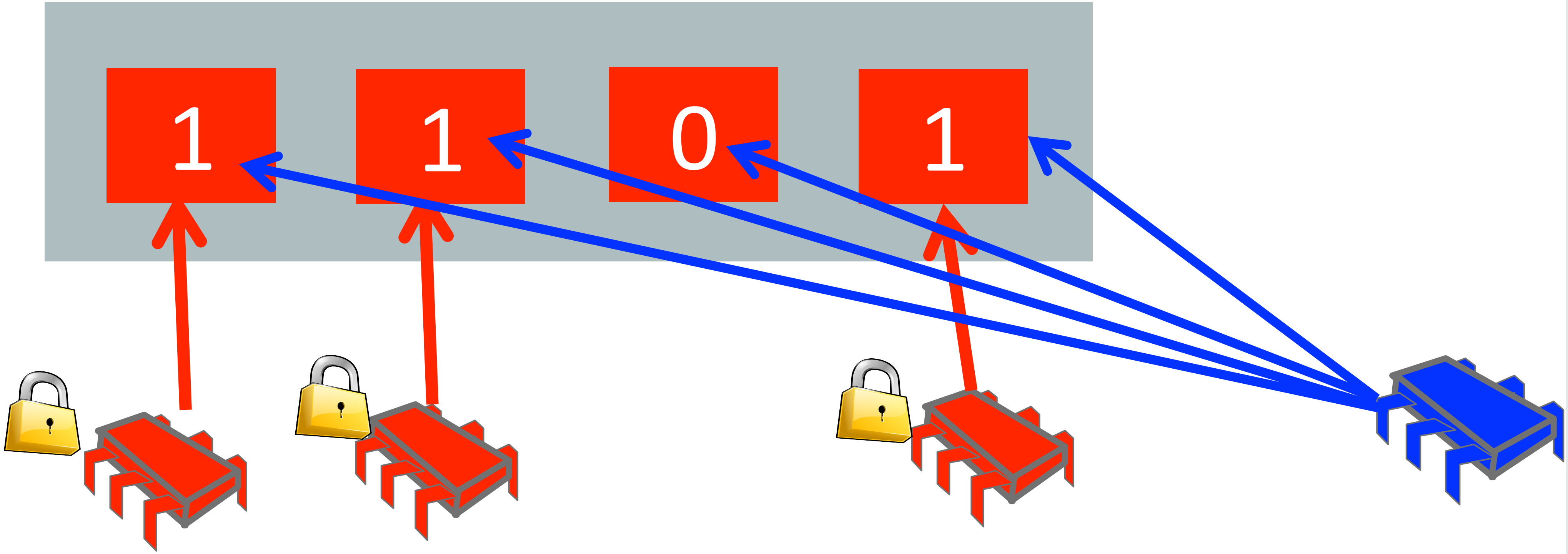


The “distributed” approach



Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The “distributed” approach



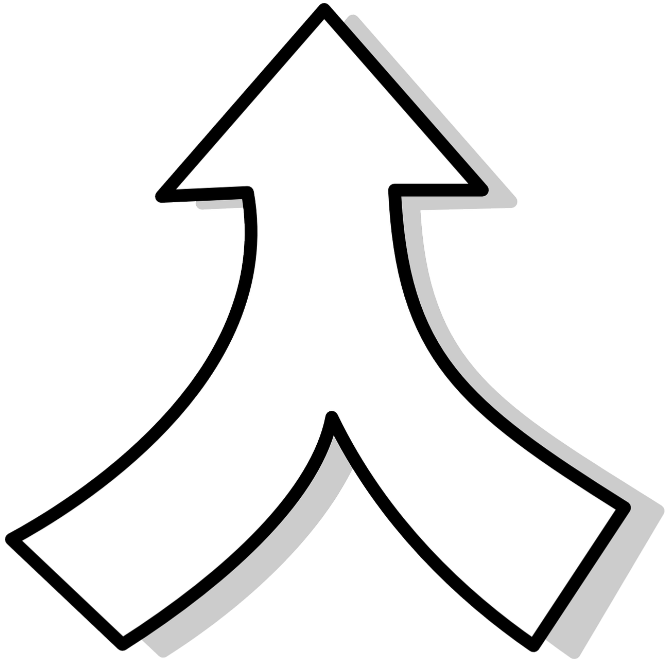
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The Scalable “Reader Indicator” Dilemma

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

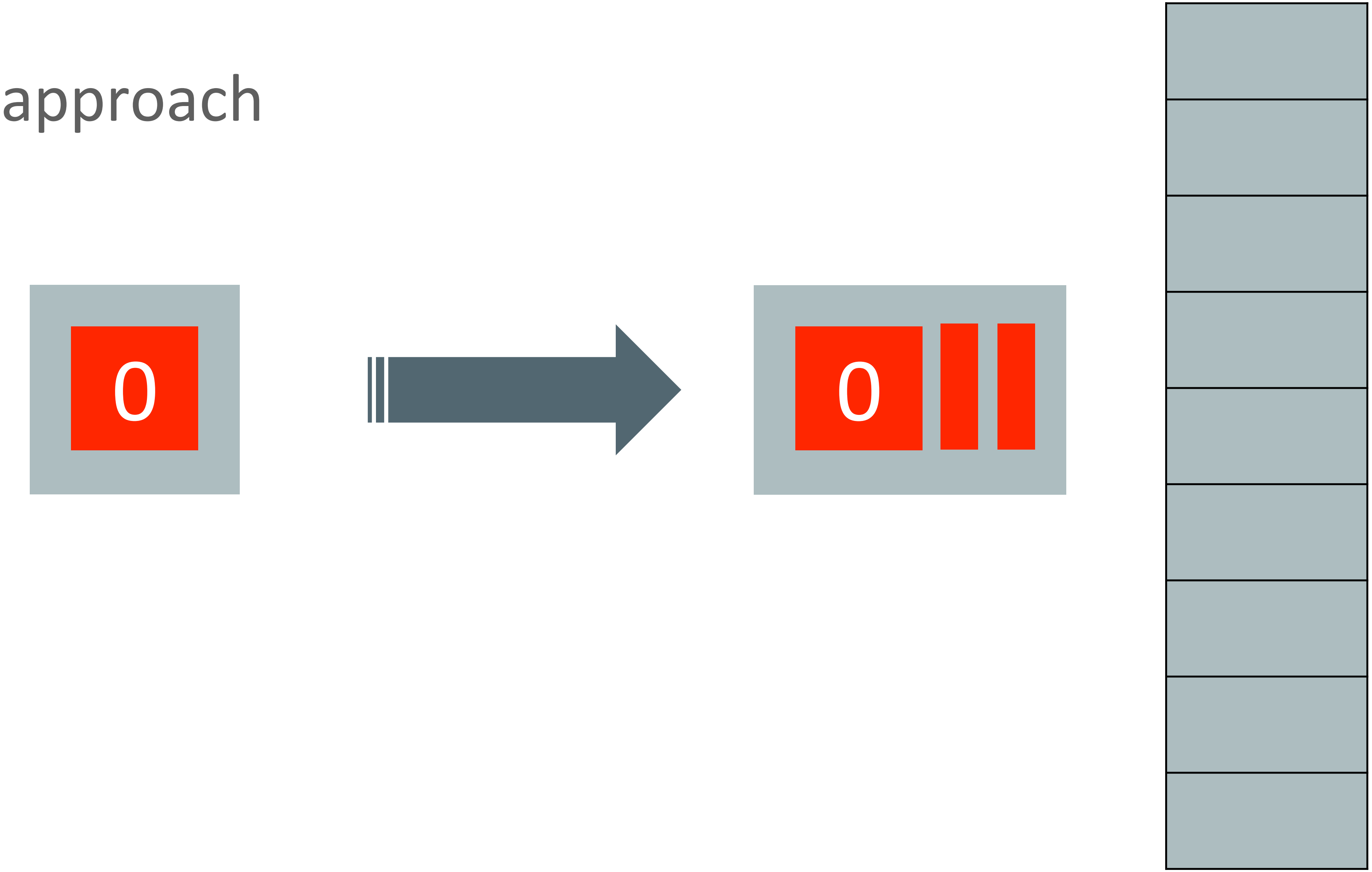
The BRAVO approach

Compact & Scalable

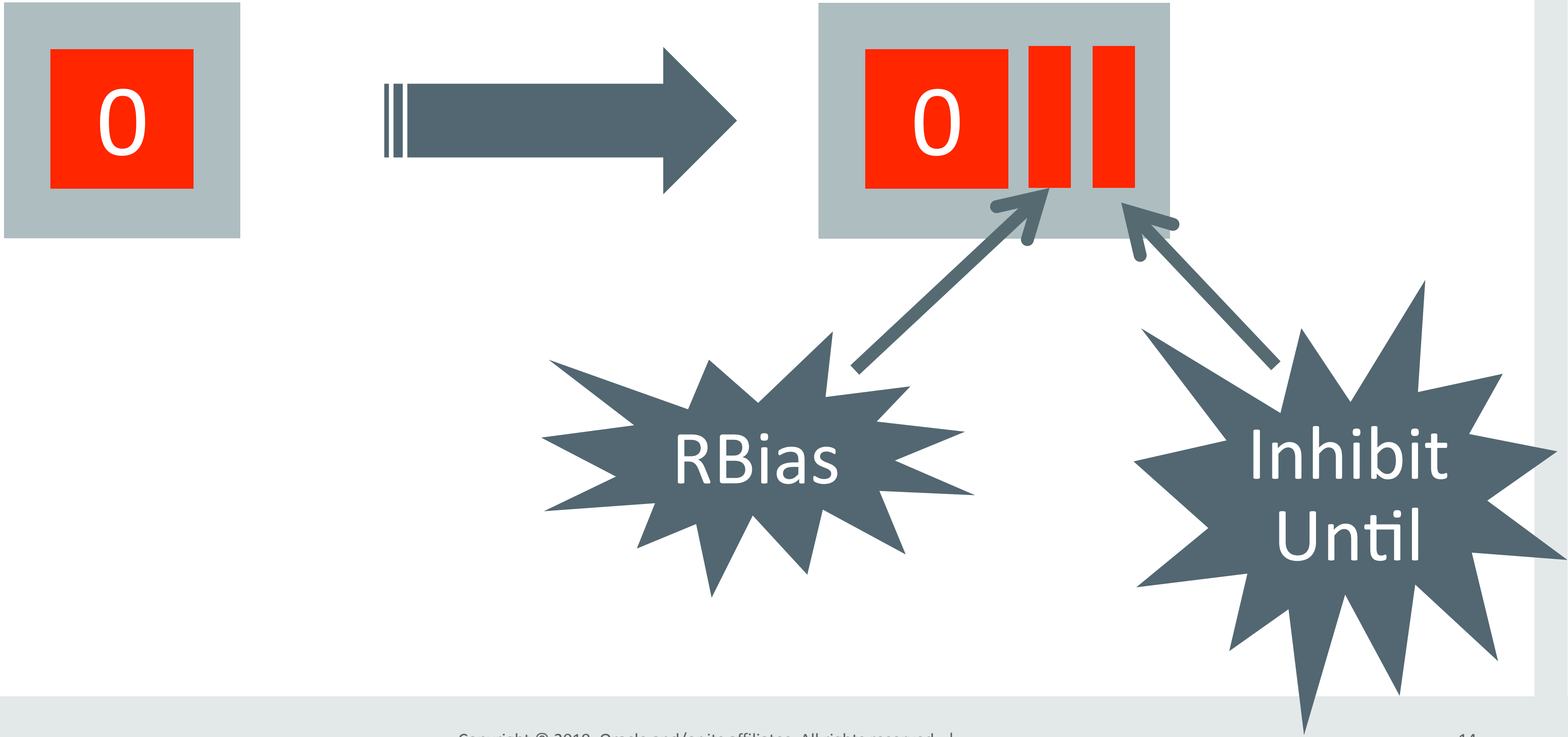


Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach

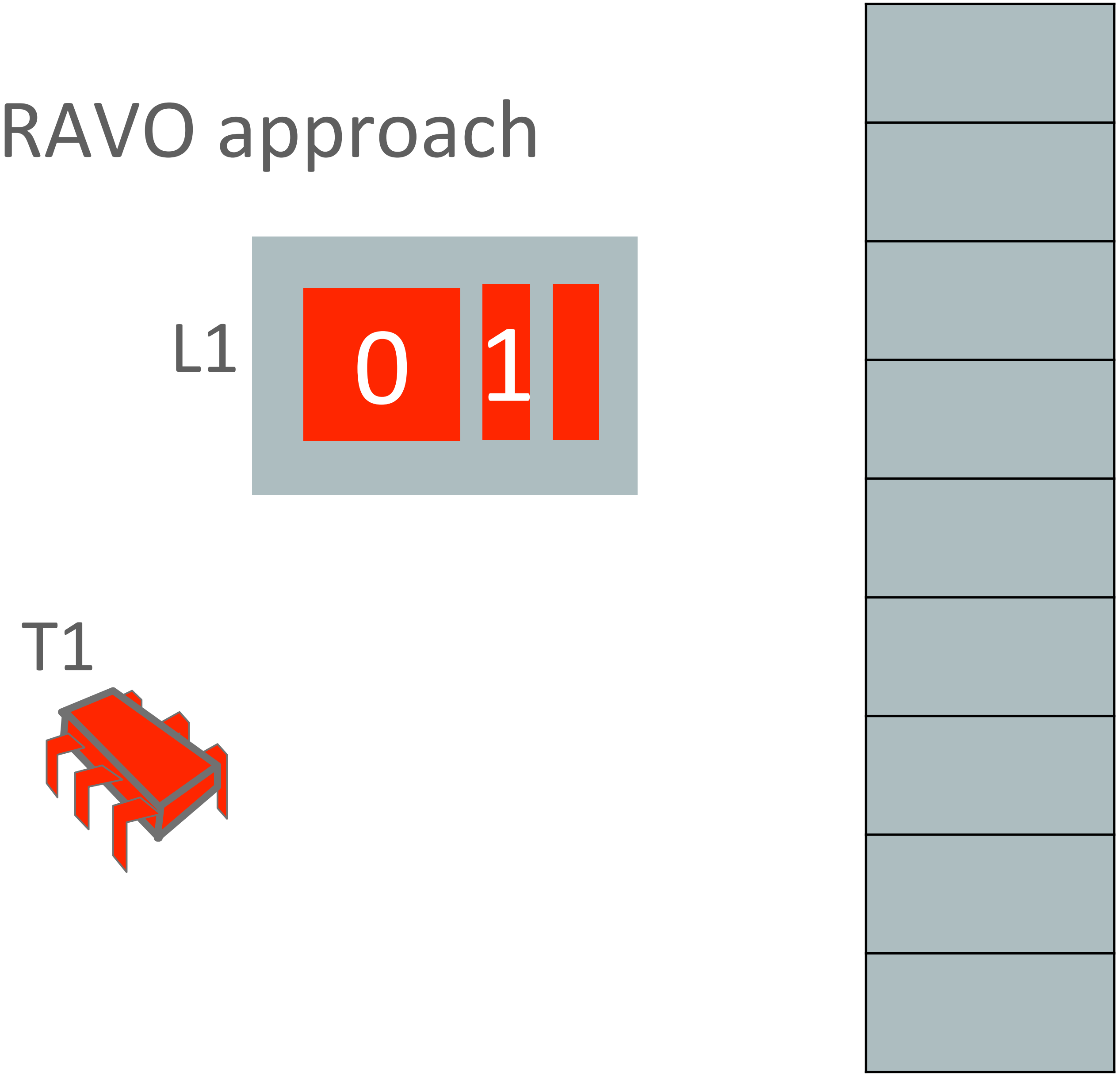


The BRAVO approach



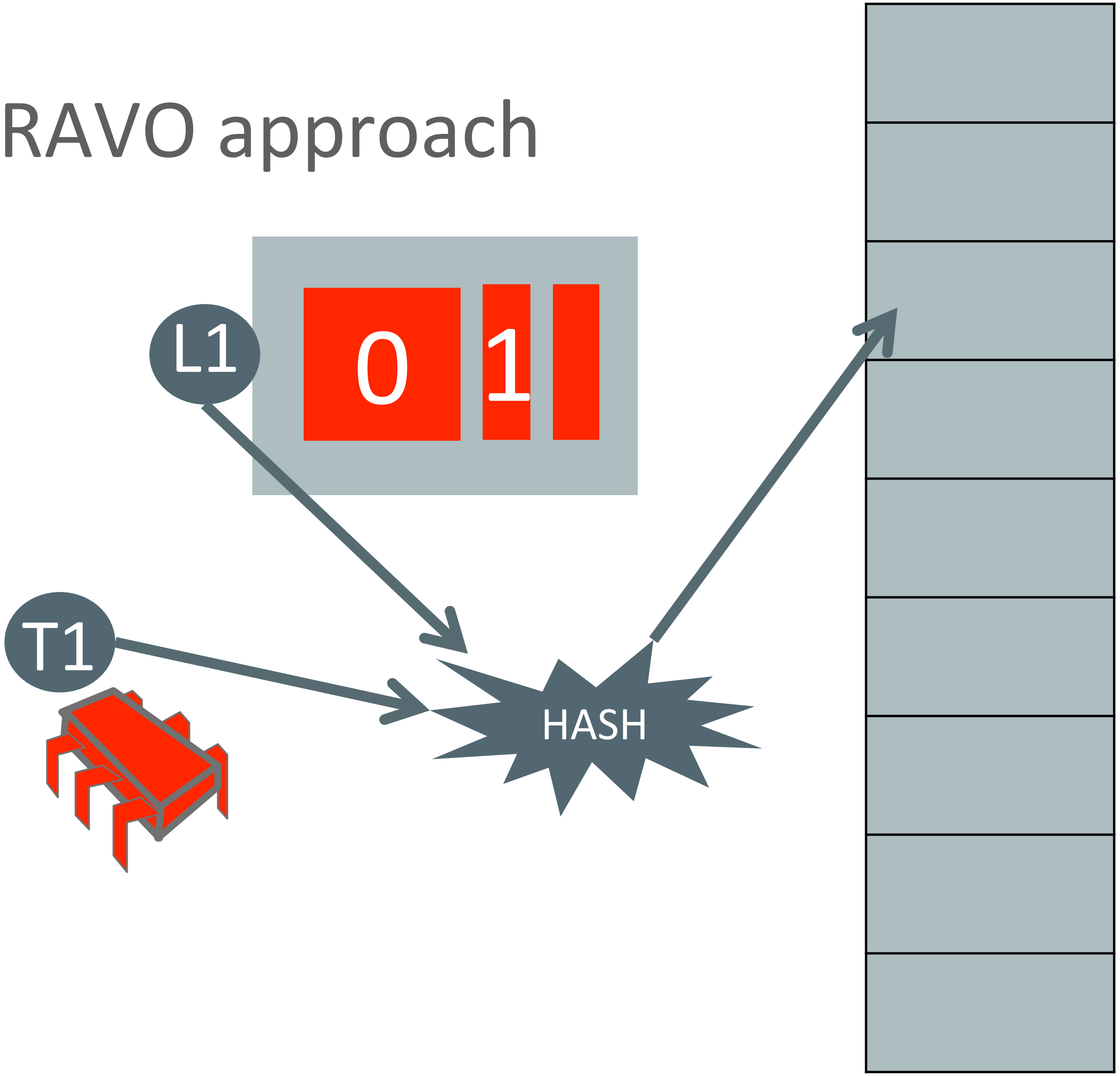
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



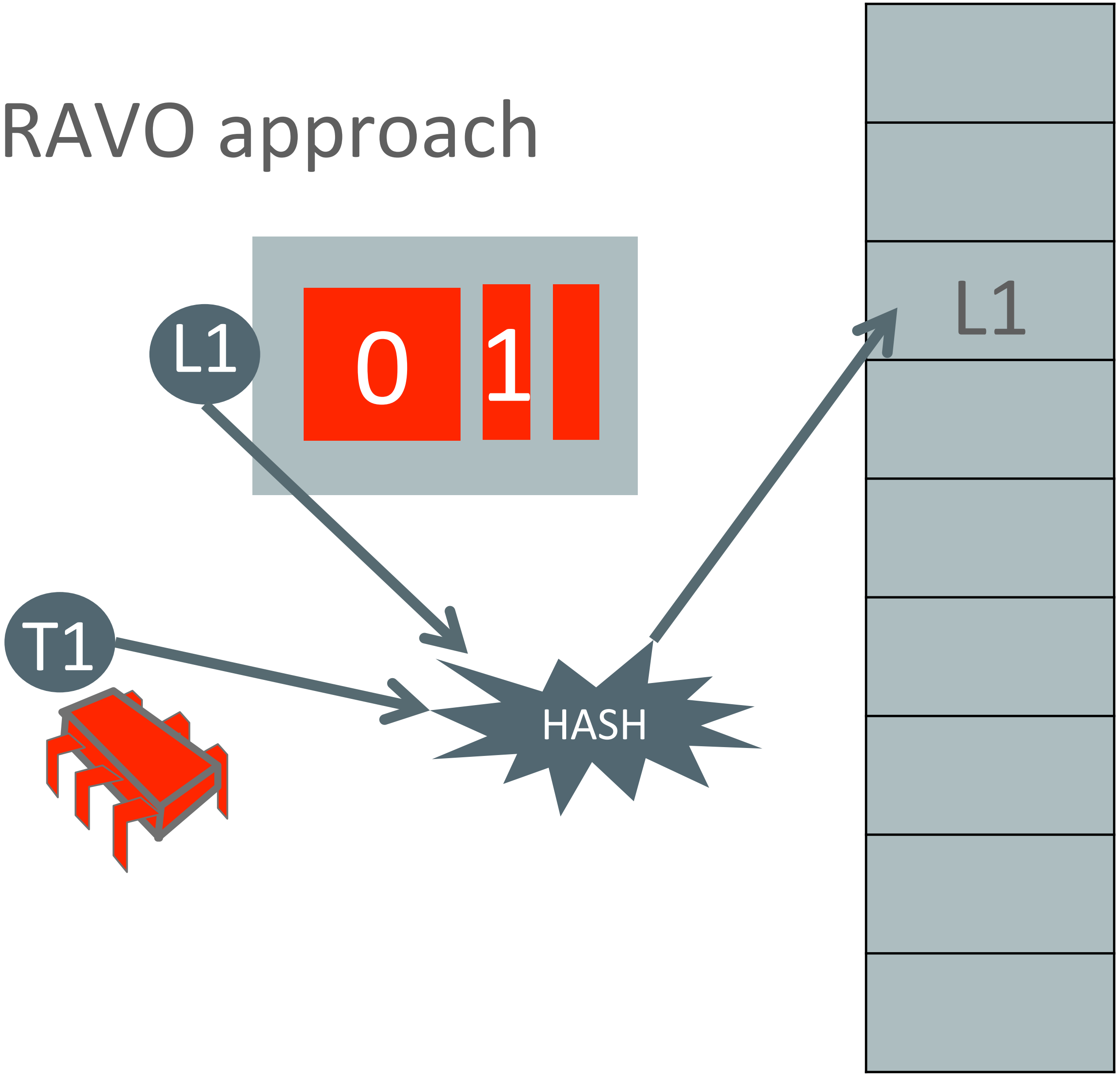
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



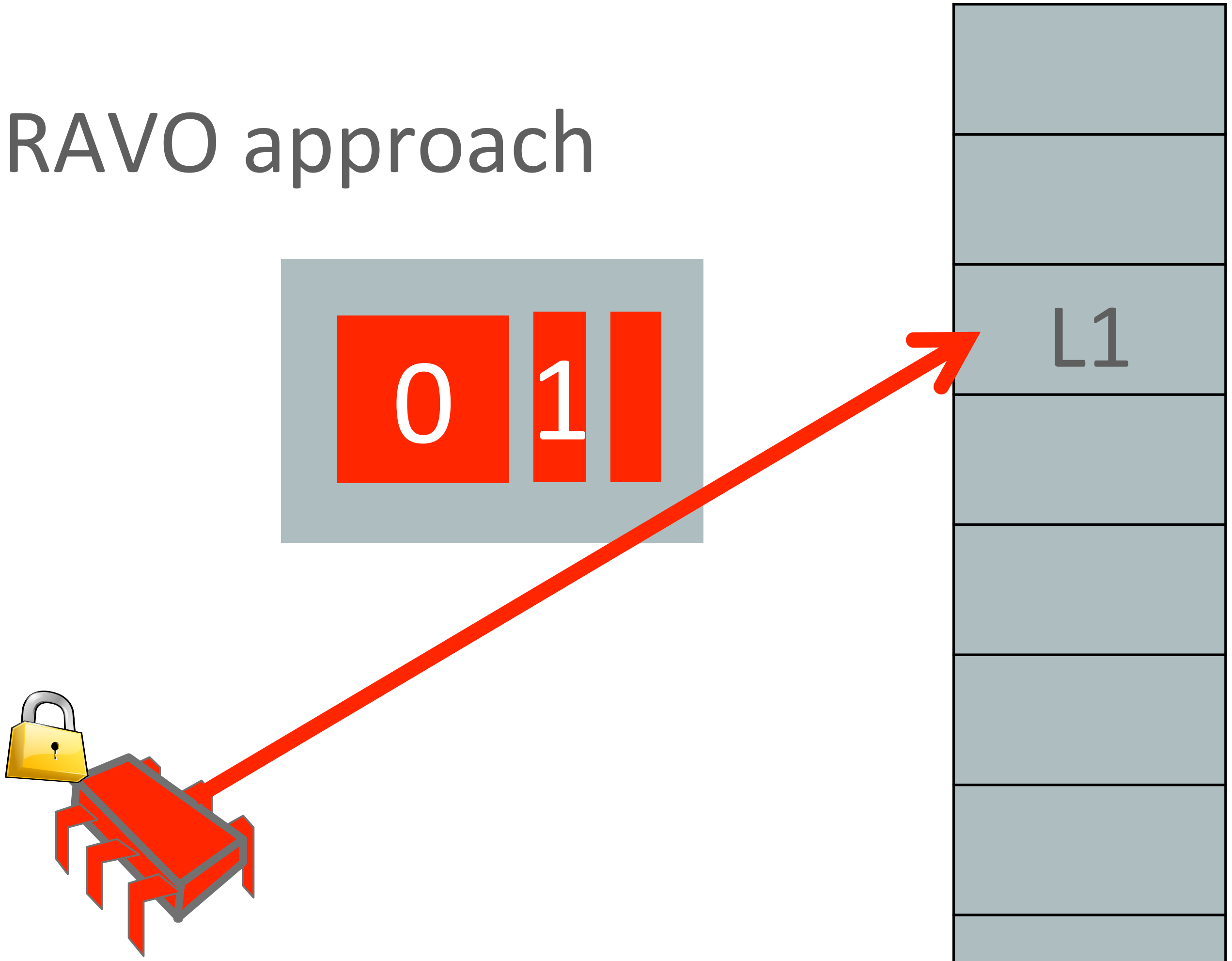
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

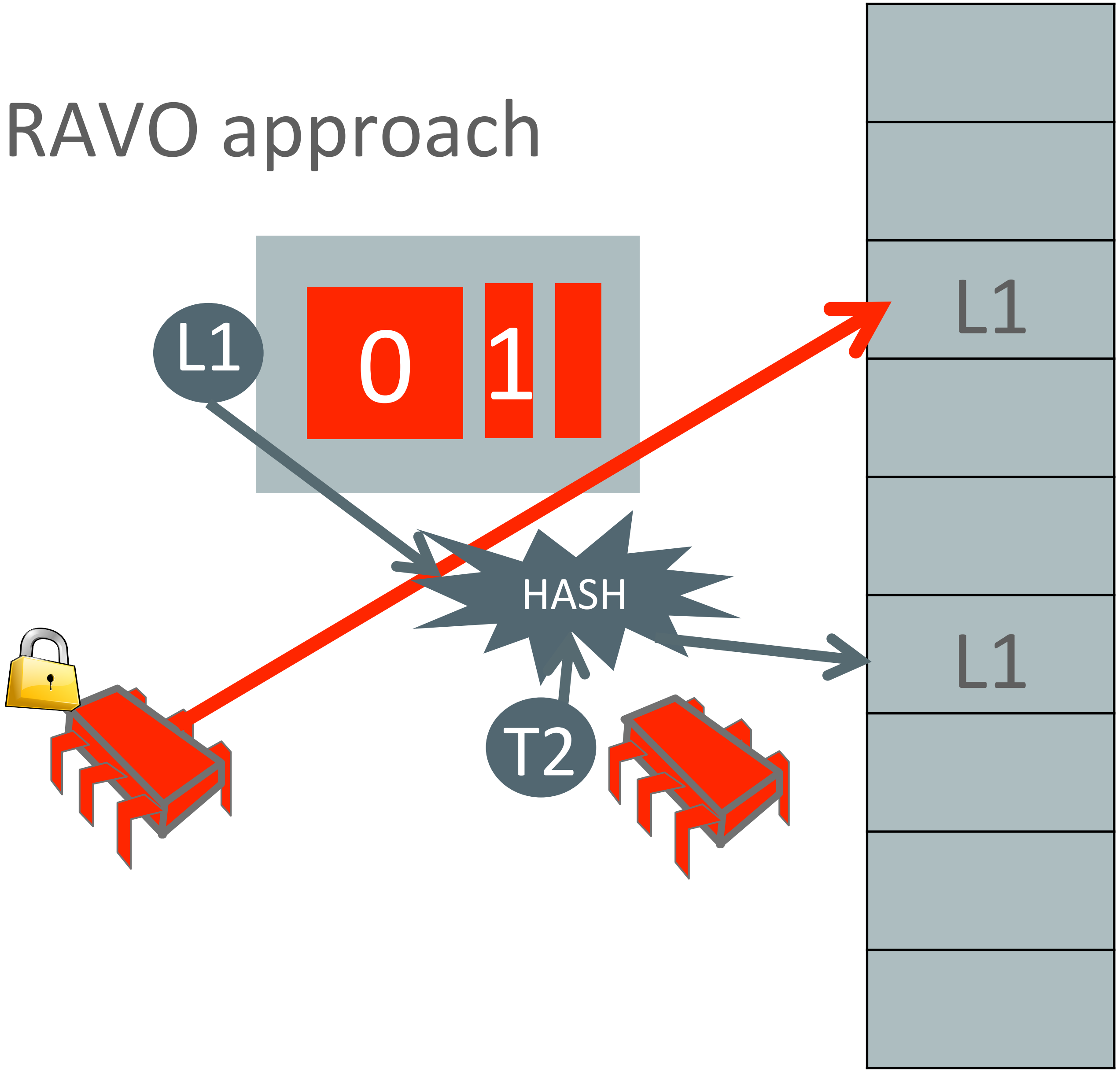
The BRAVO approach



ORACLE

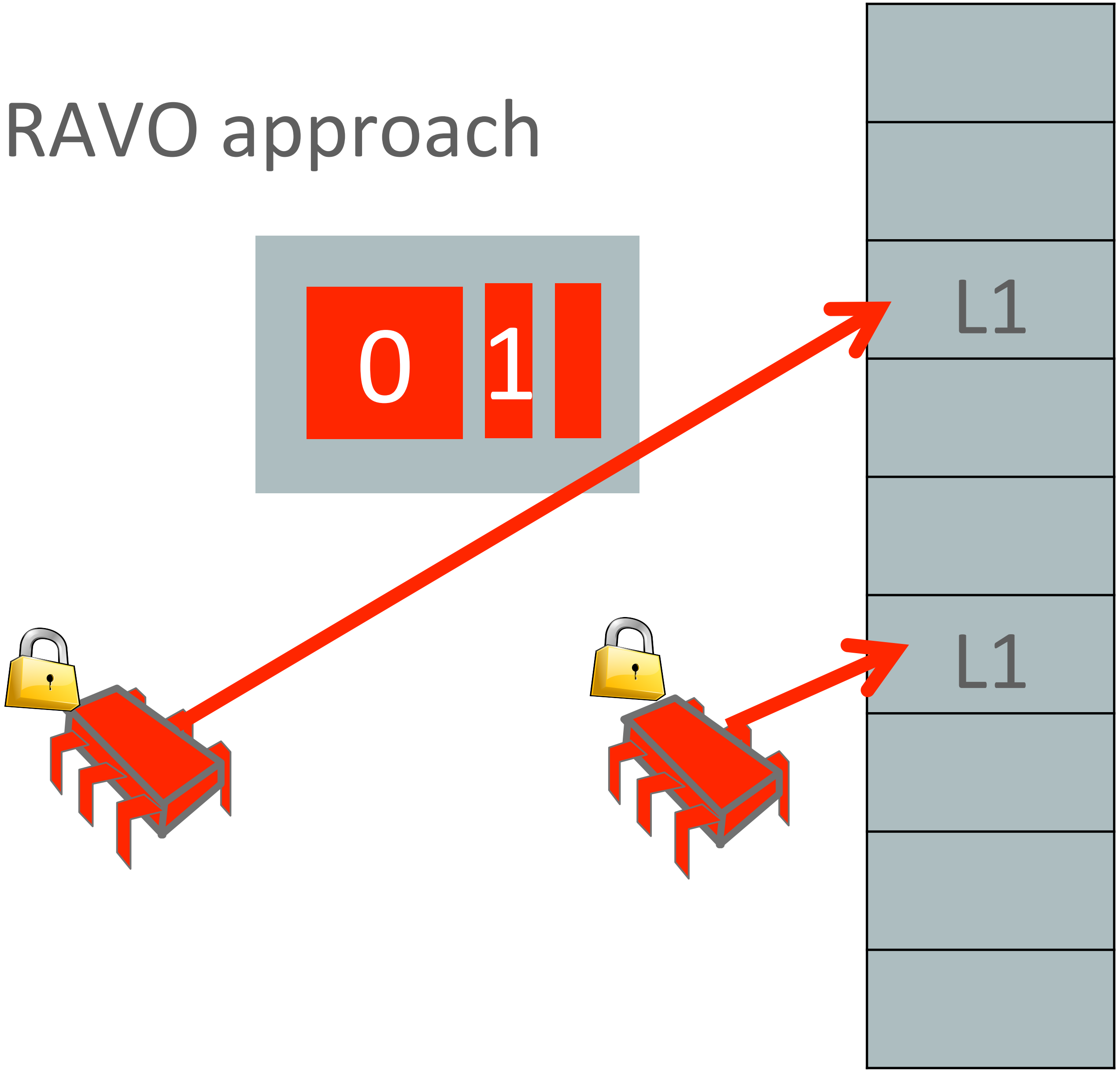
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



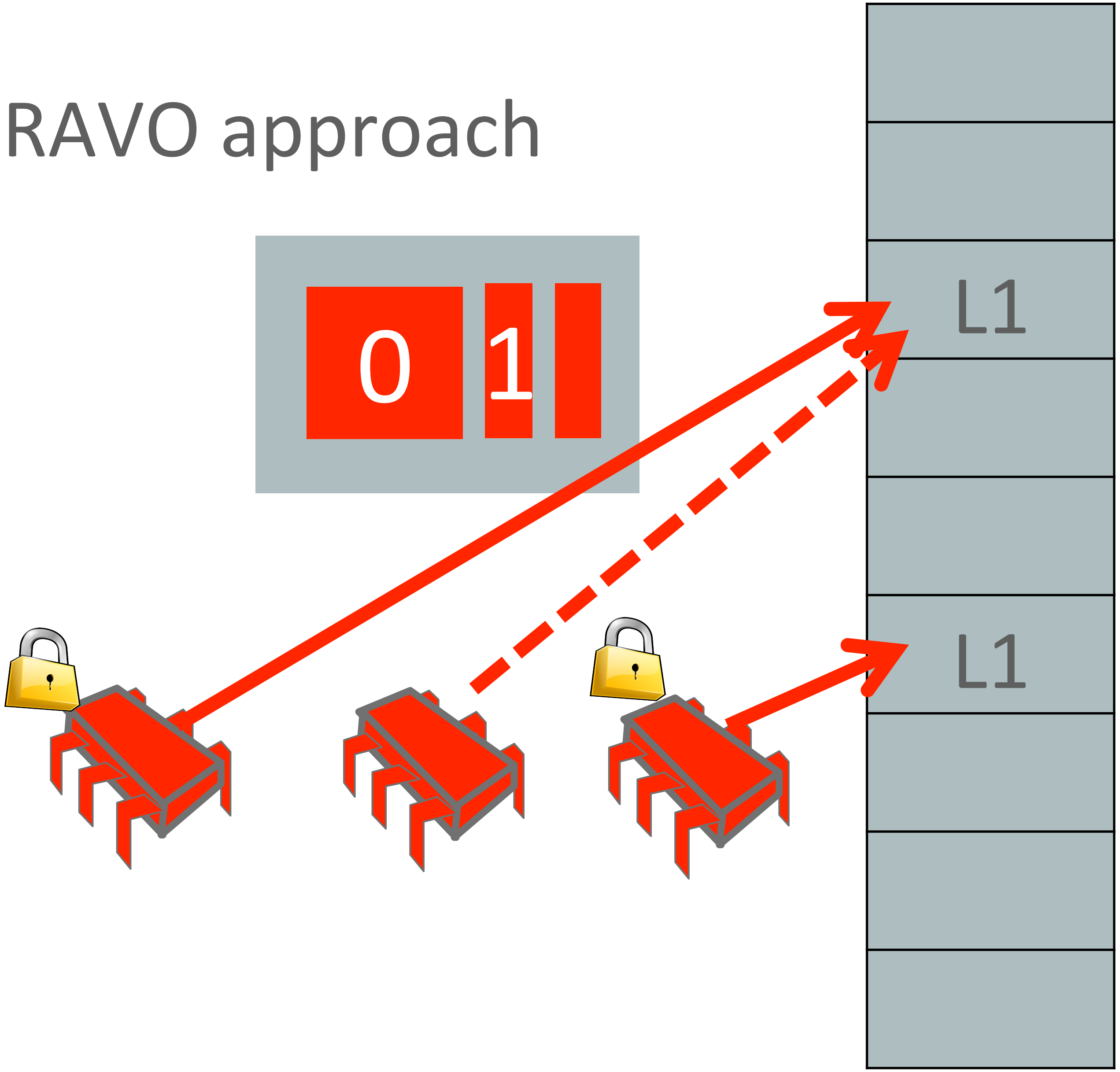
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



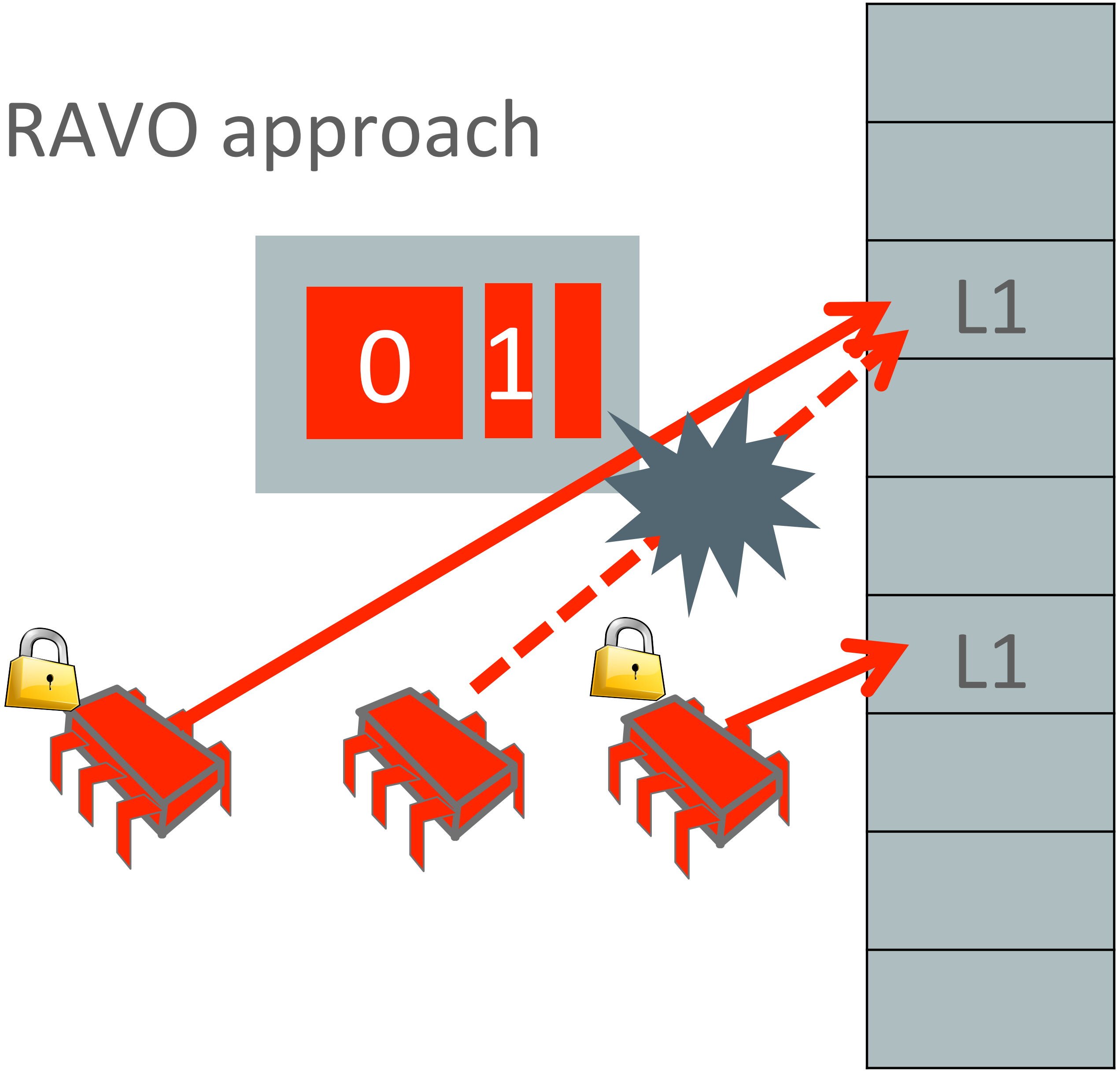
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



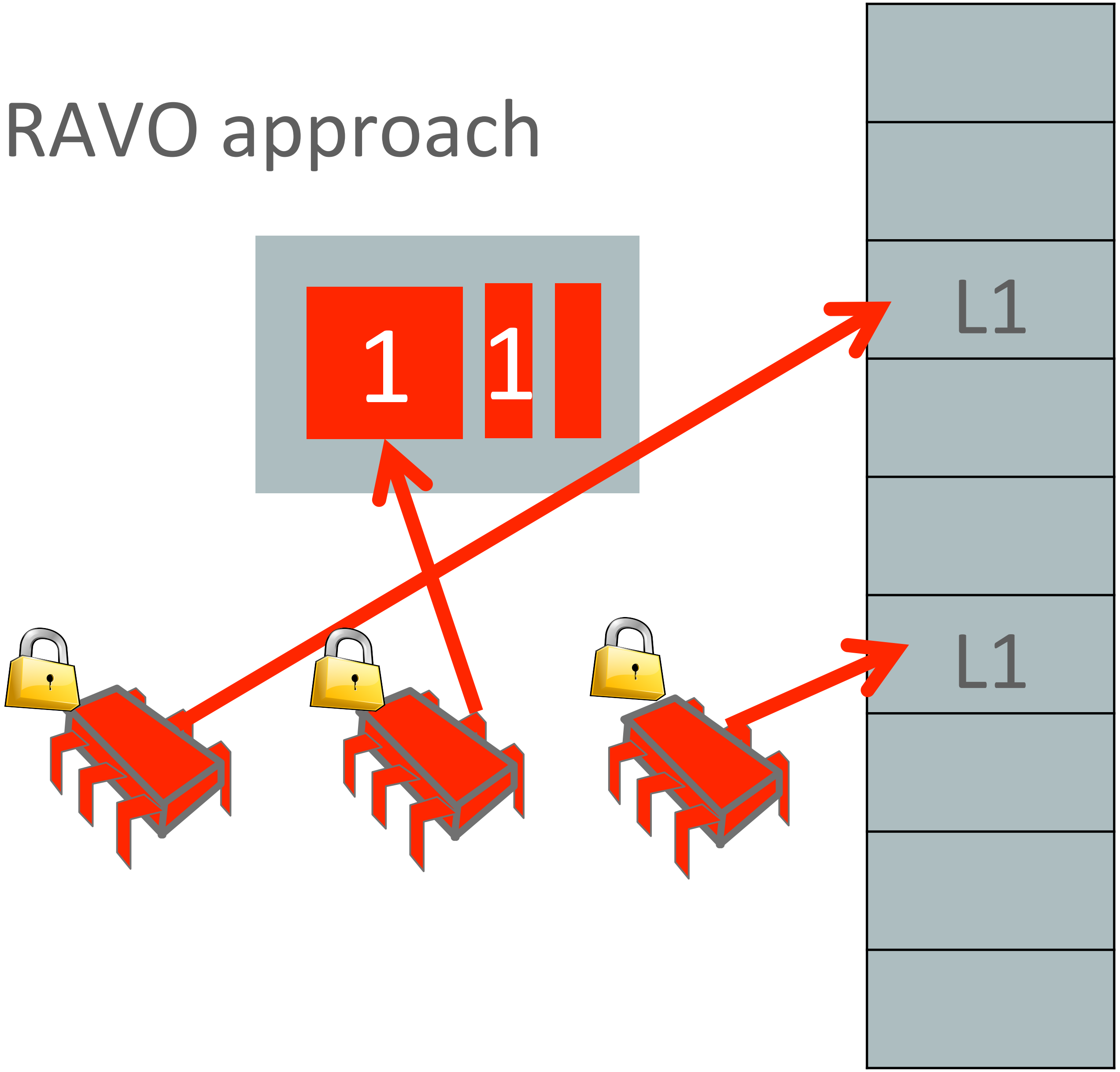
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



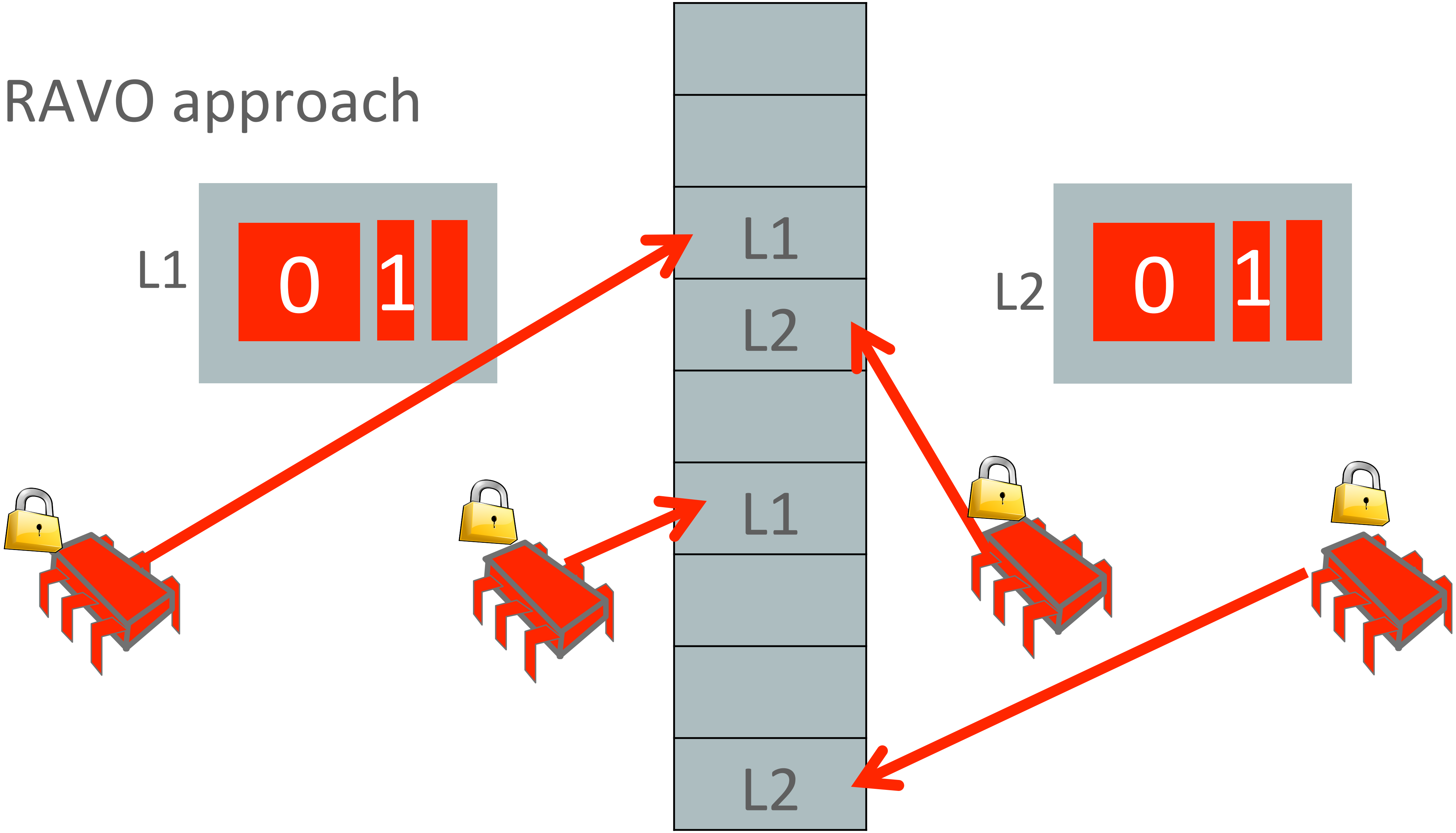
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



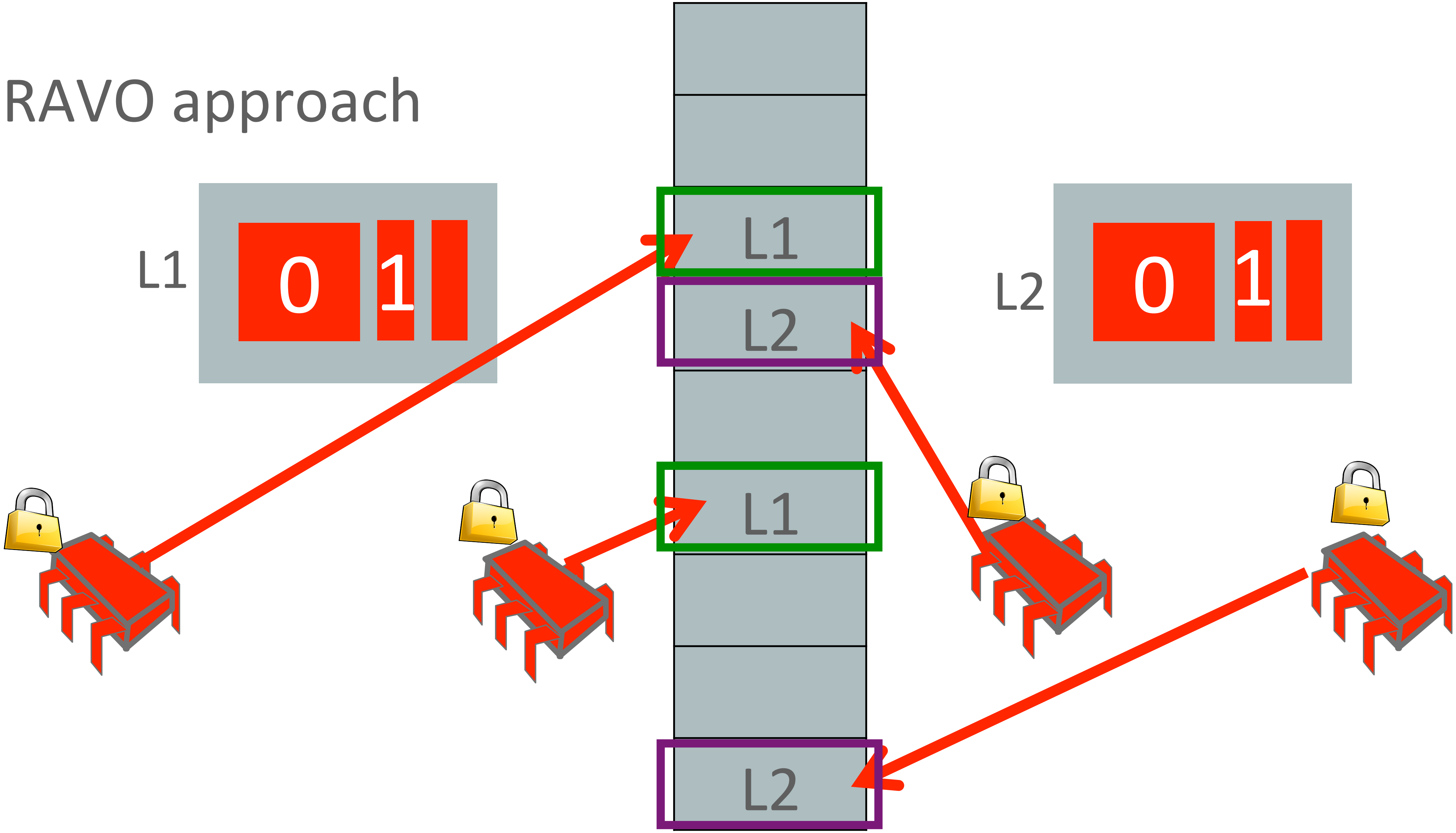
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



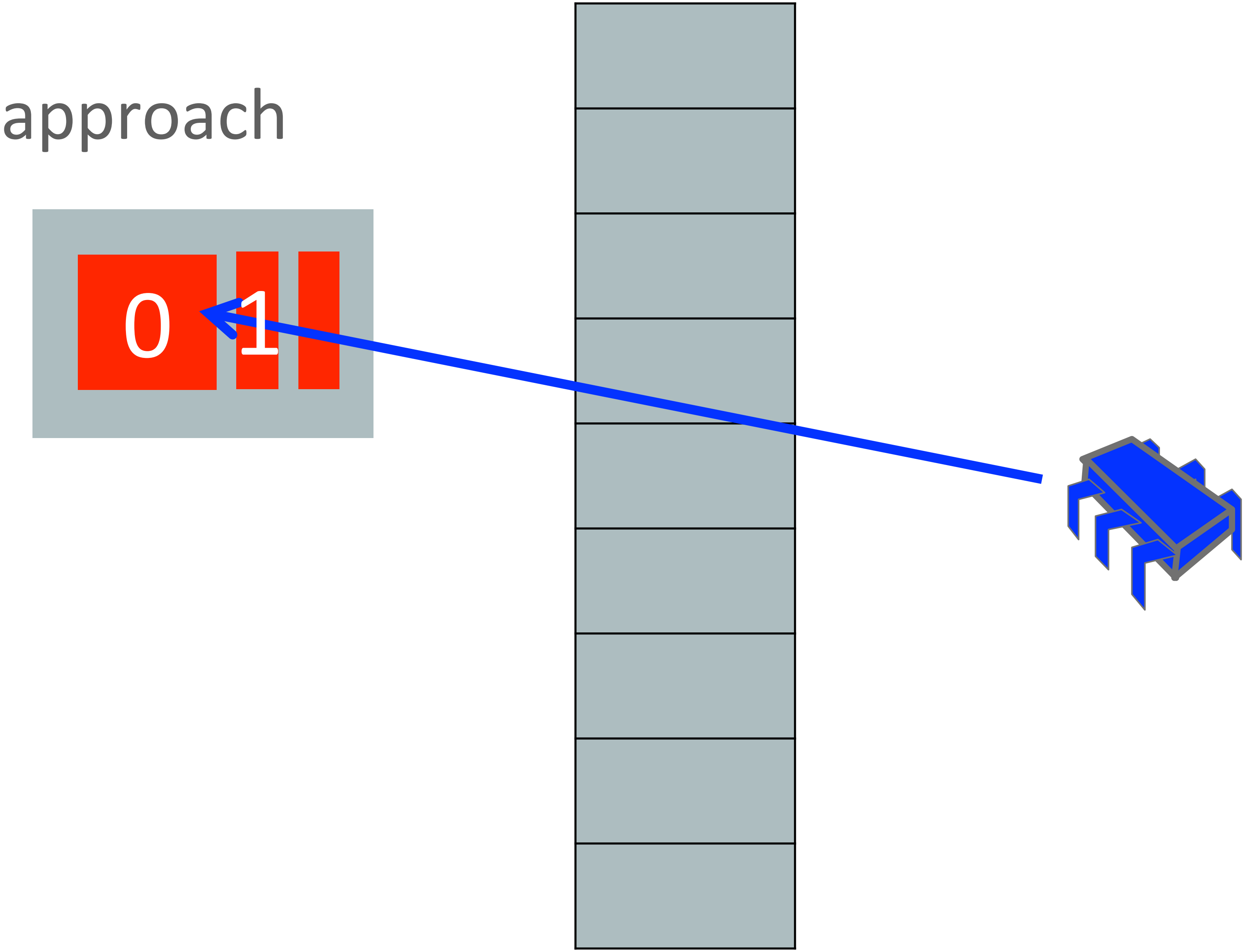
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



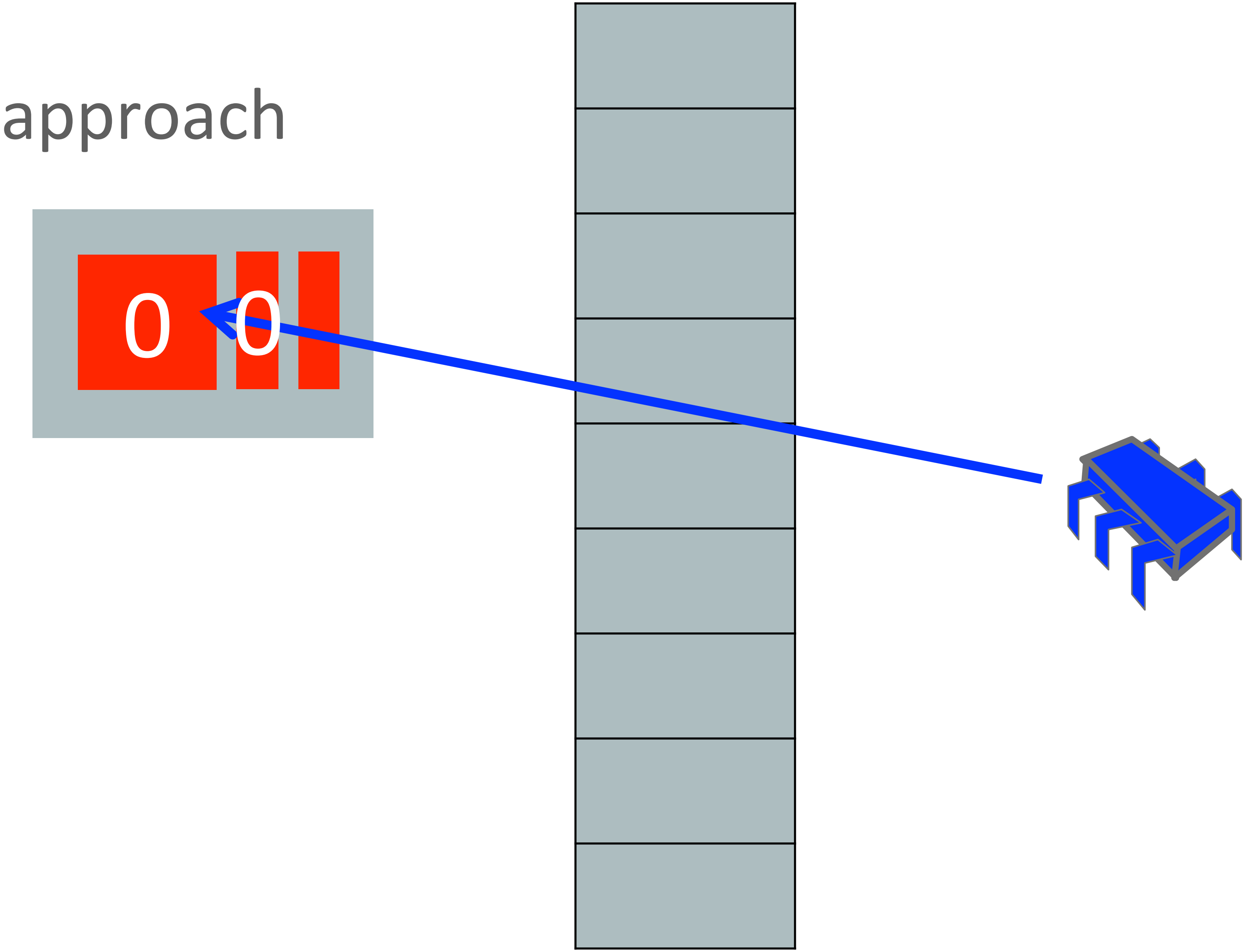
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



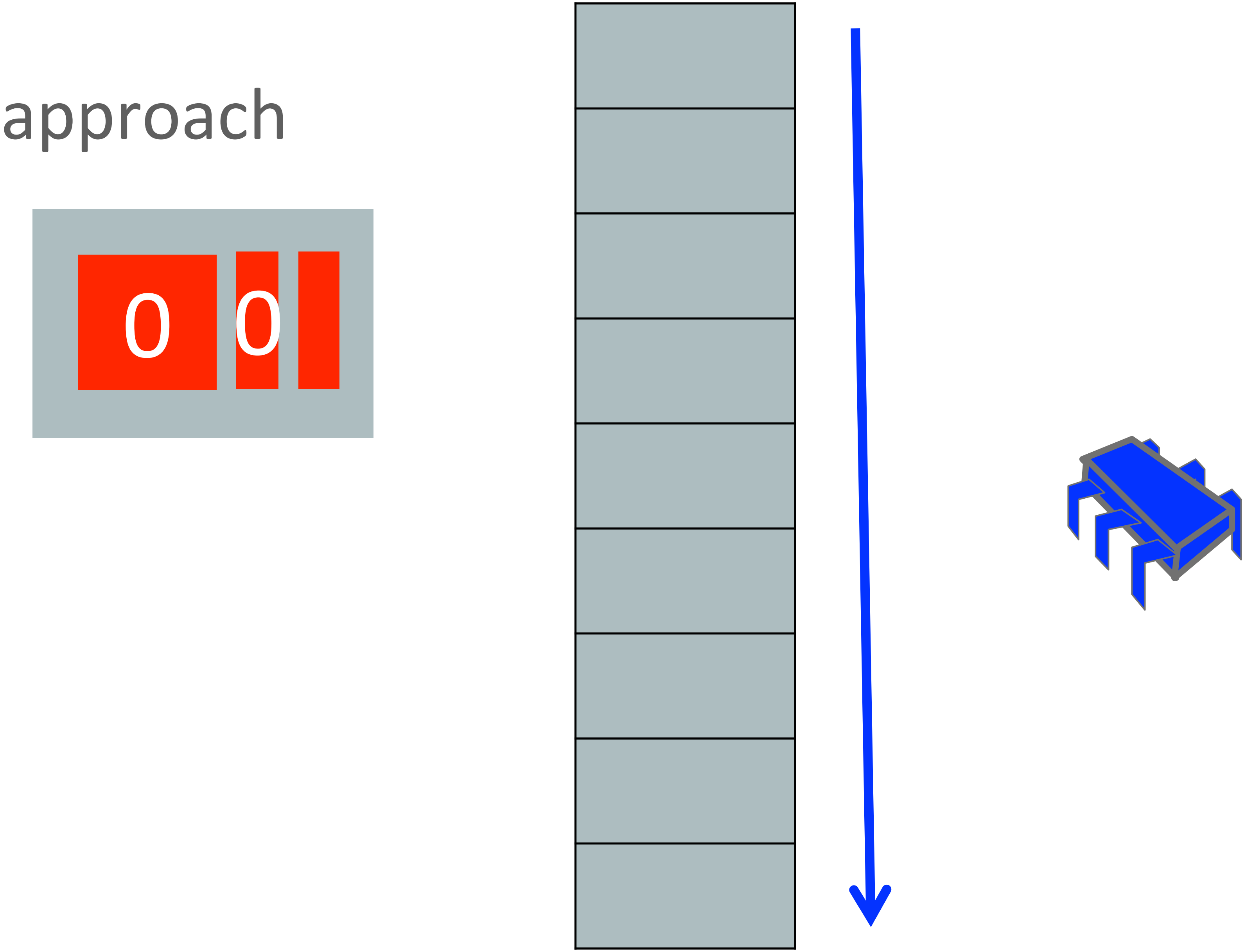
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



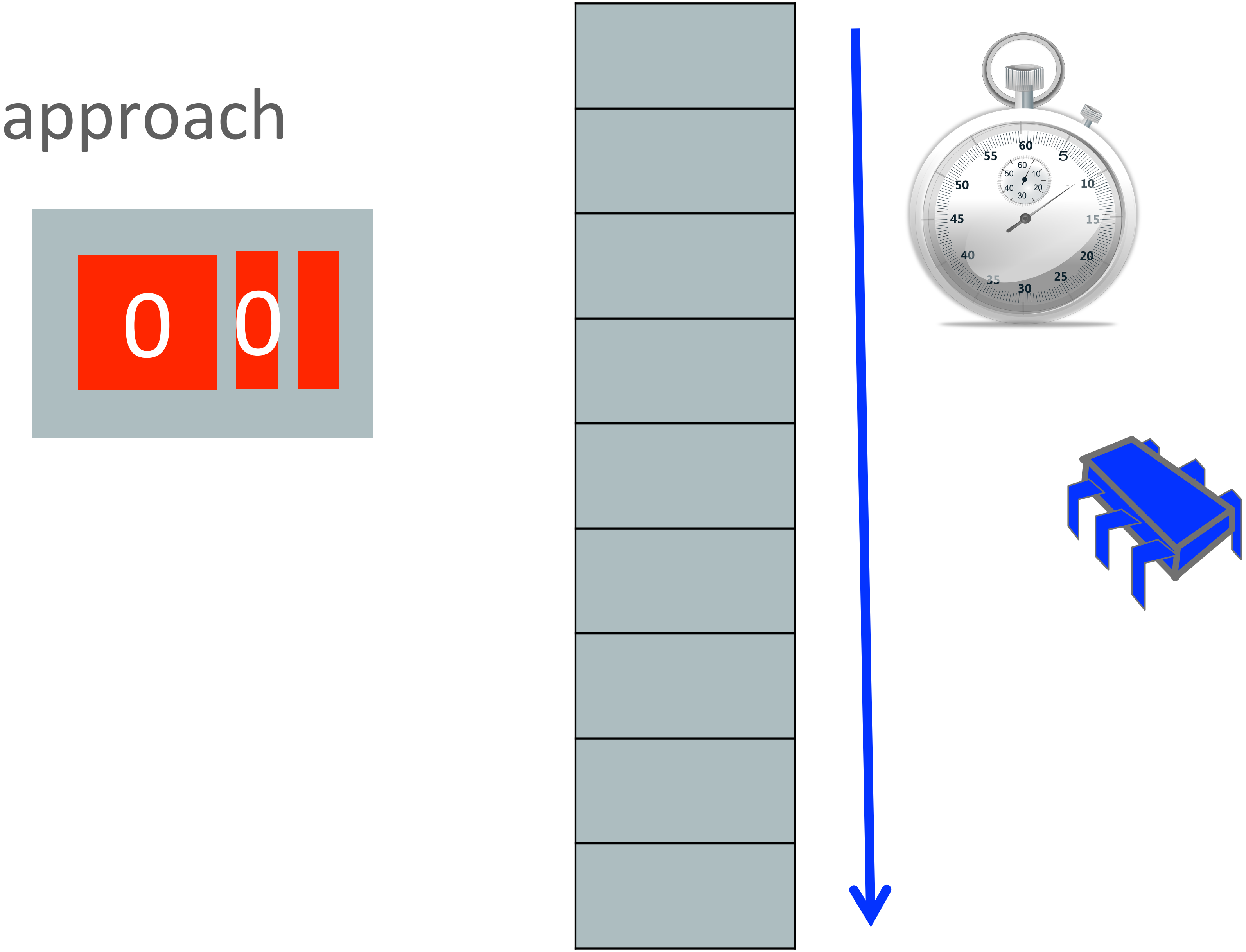
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



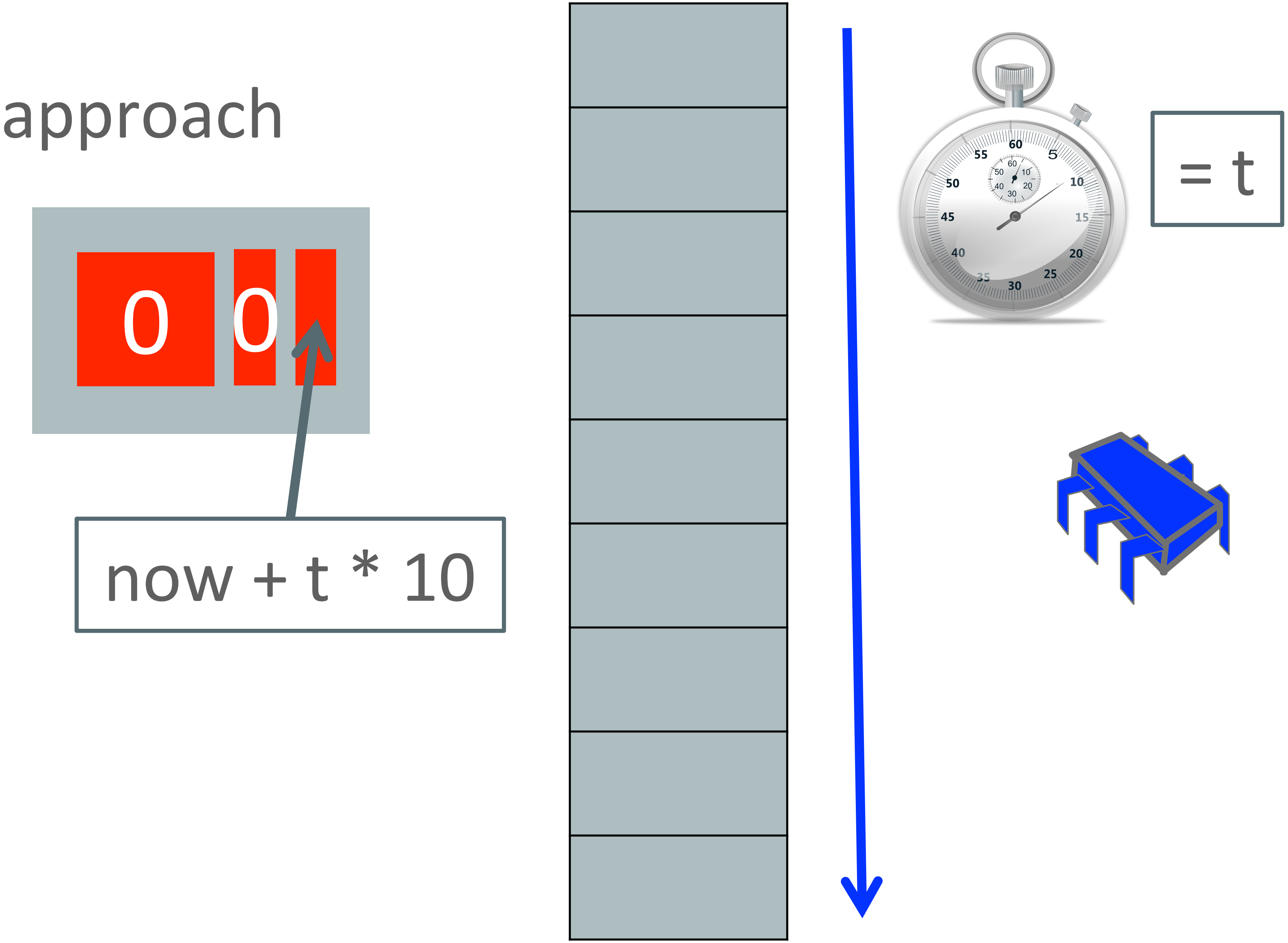
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



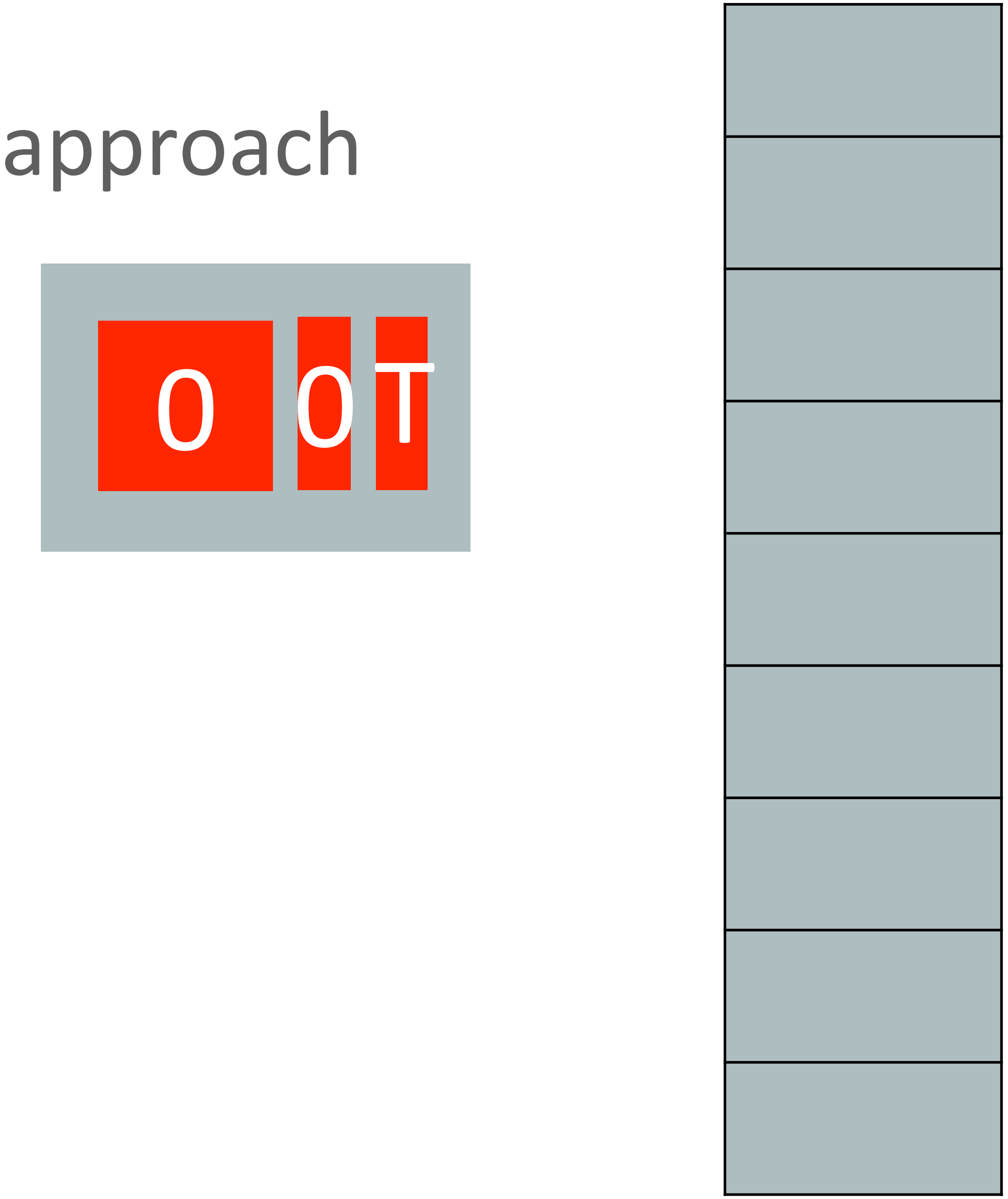
0 0

now + t * 10

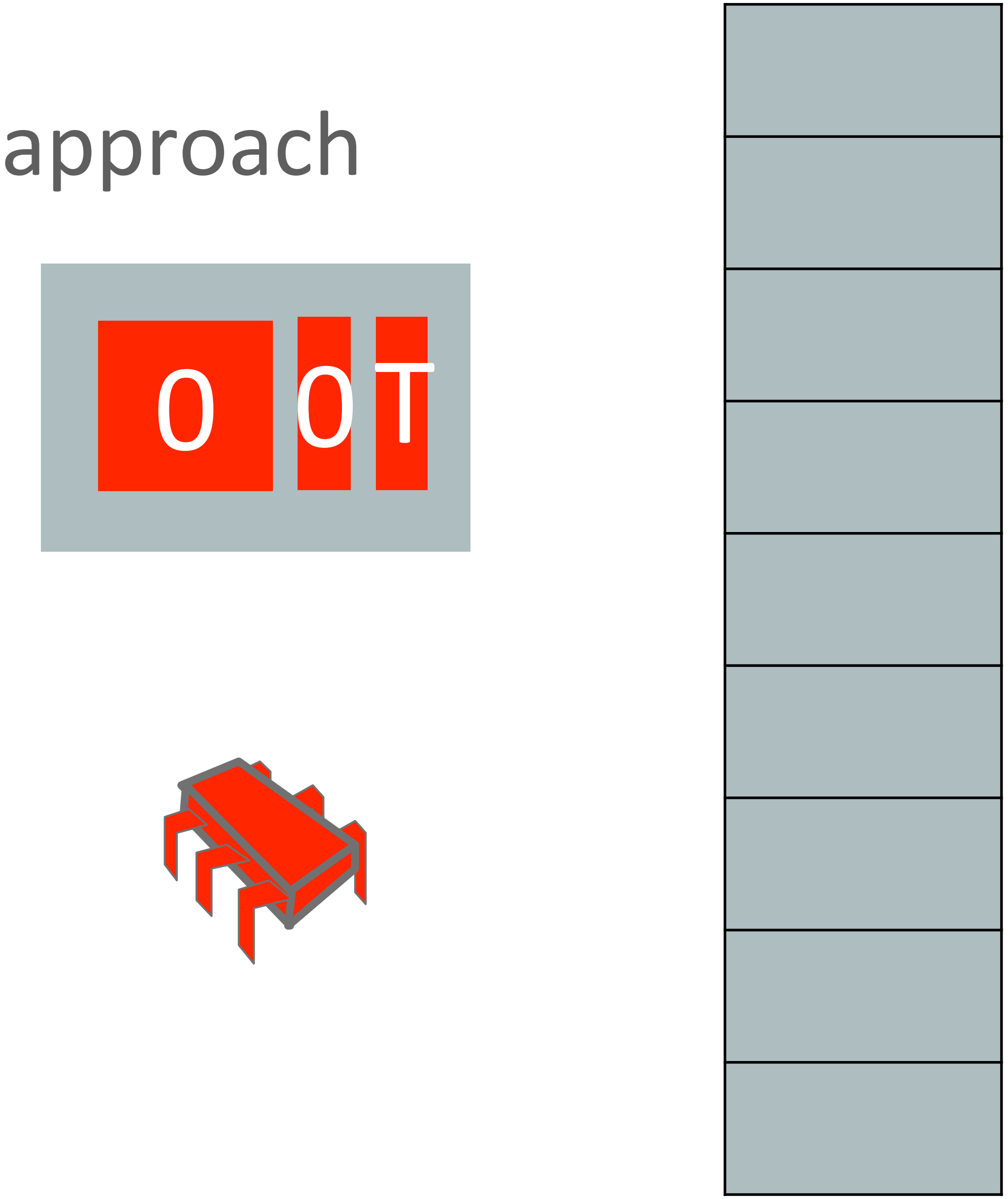
= t

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach

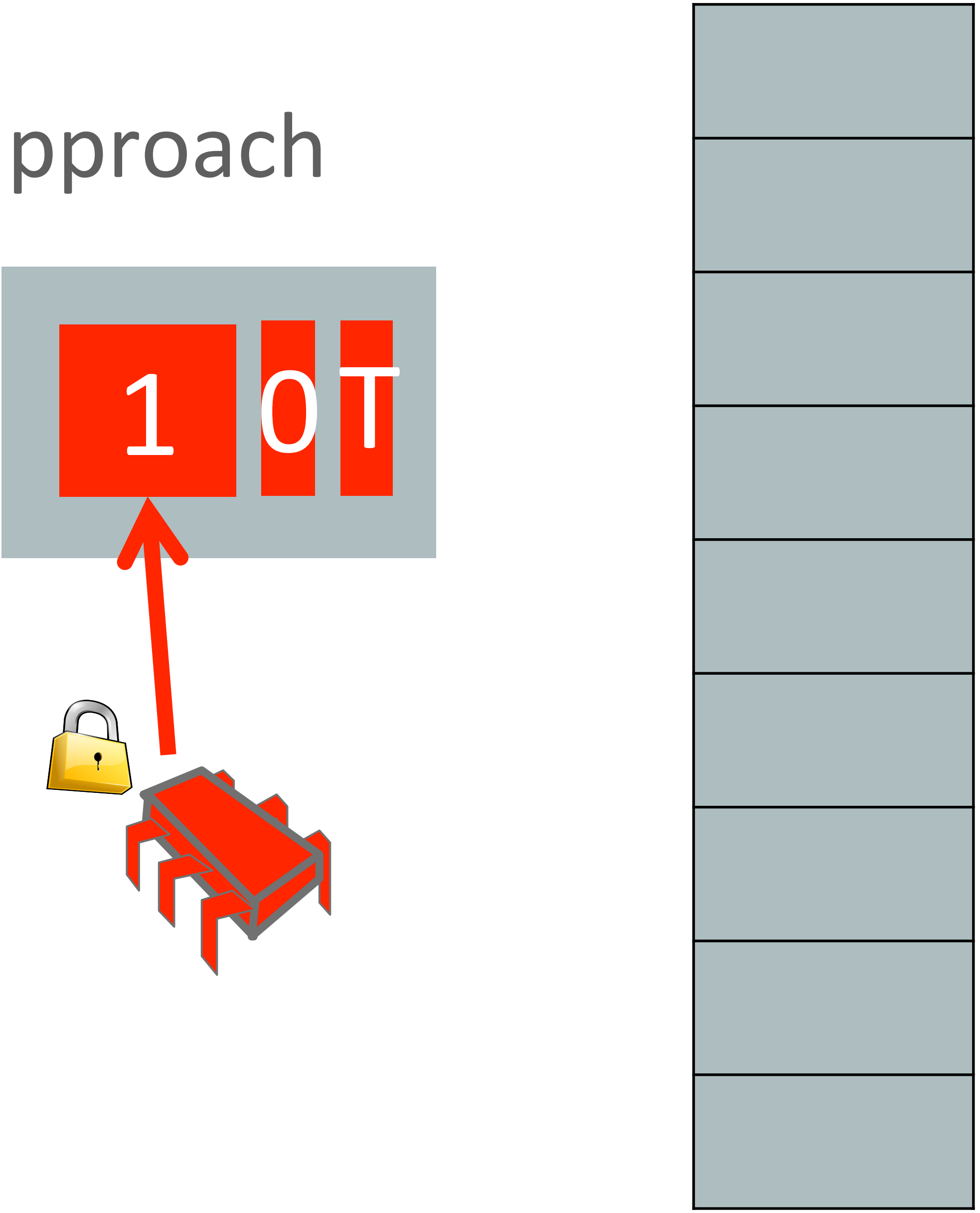


The BRAVO approach



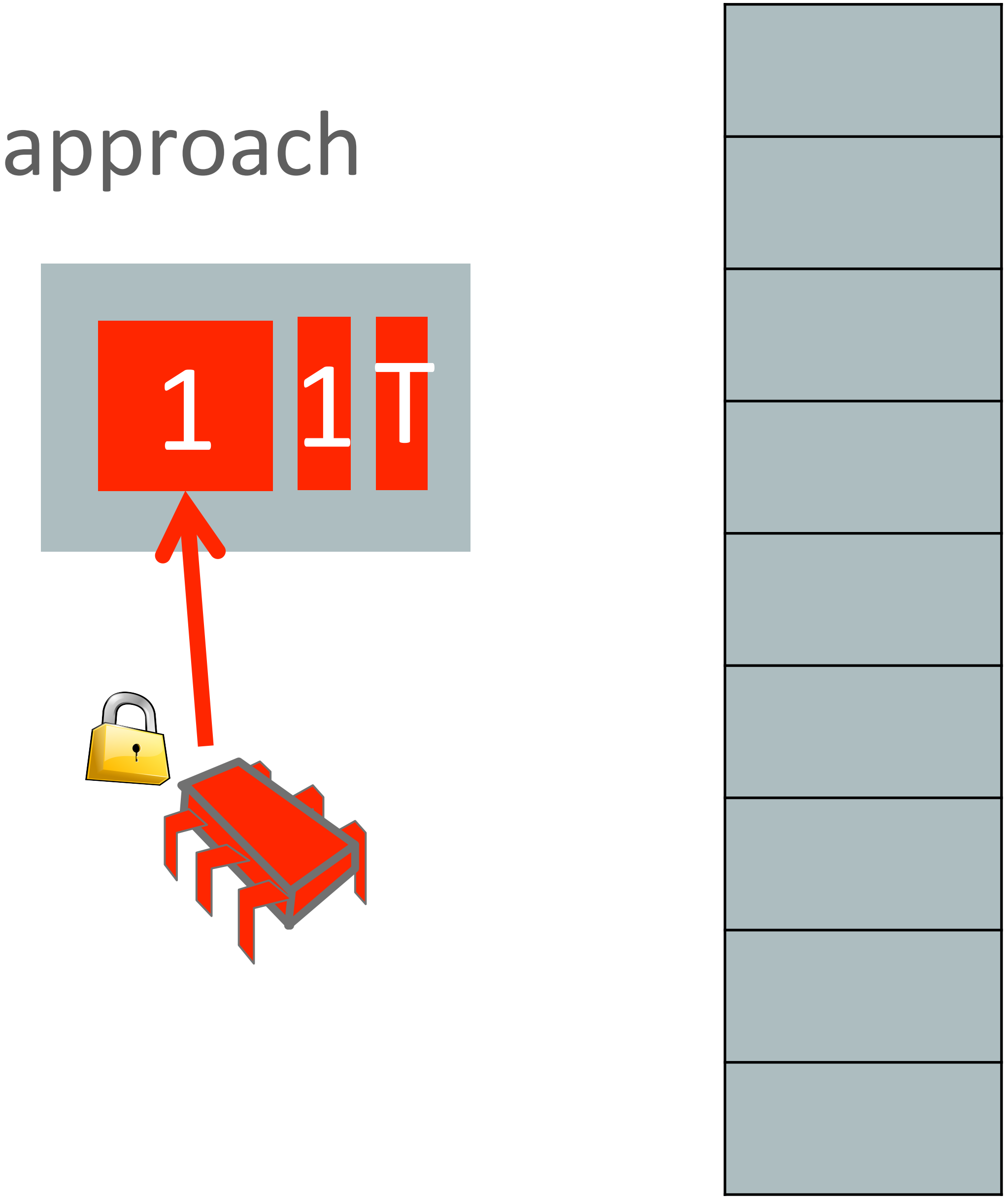
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



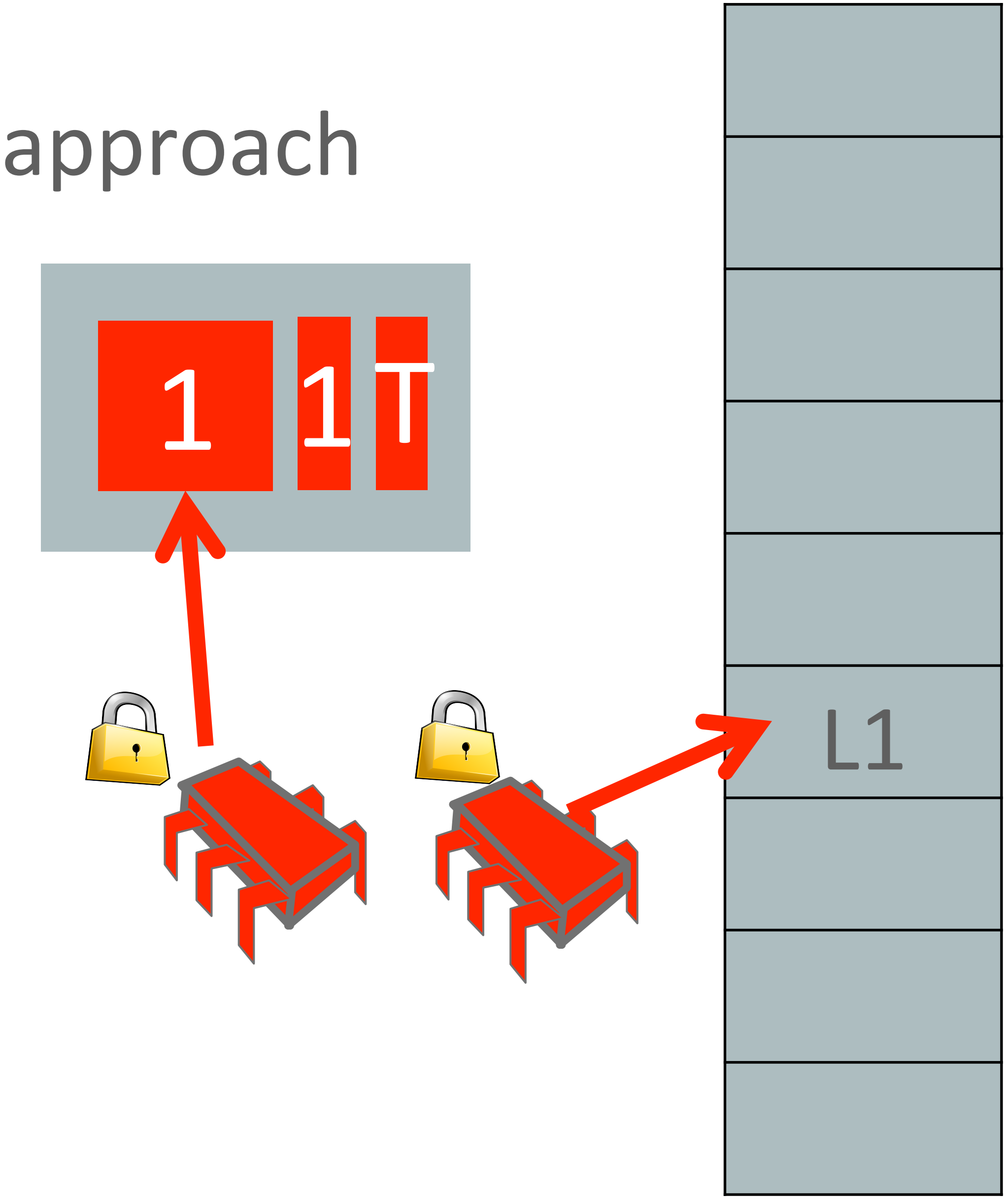
Copyright © 2019, Oracle and/or its affiliates. All rights reserved.

The BRAVO approach



Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

The BRAVO approach



ORACLE

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Evaluation

- Easy to integrate with existing locks
- Compact
- Accelerates reads
- Handles writes gracefully

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Evaluation: Easy to integrate

- Brandenburg-Anderson (BA) reader-writer lock

- POSIX Pthread reader-writer lock

- Linux kernel rwsem

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Evaluation: Compact

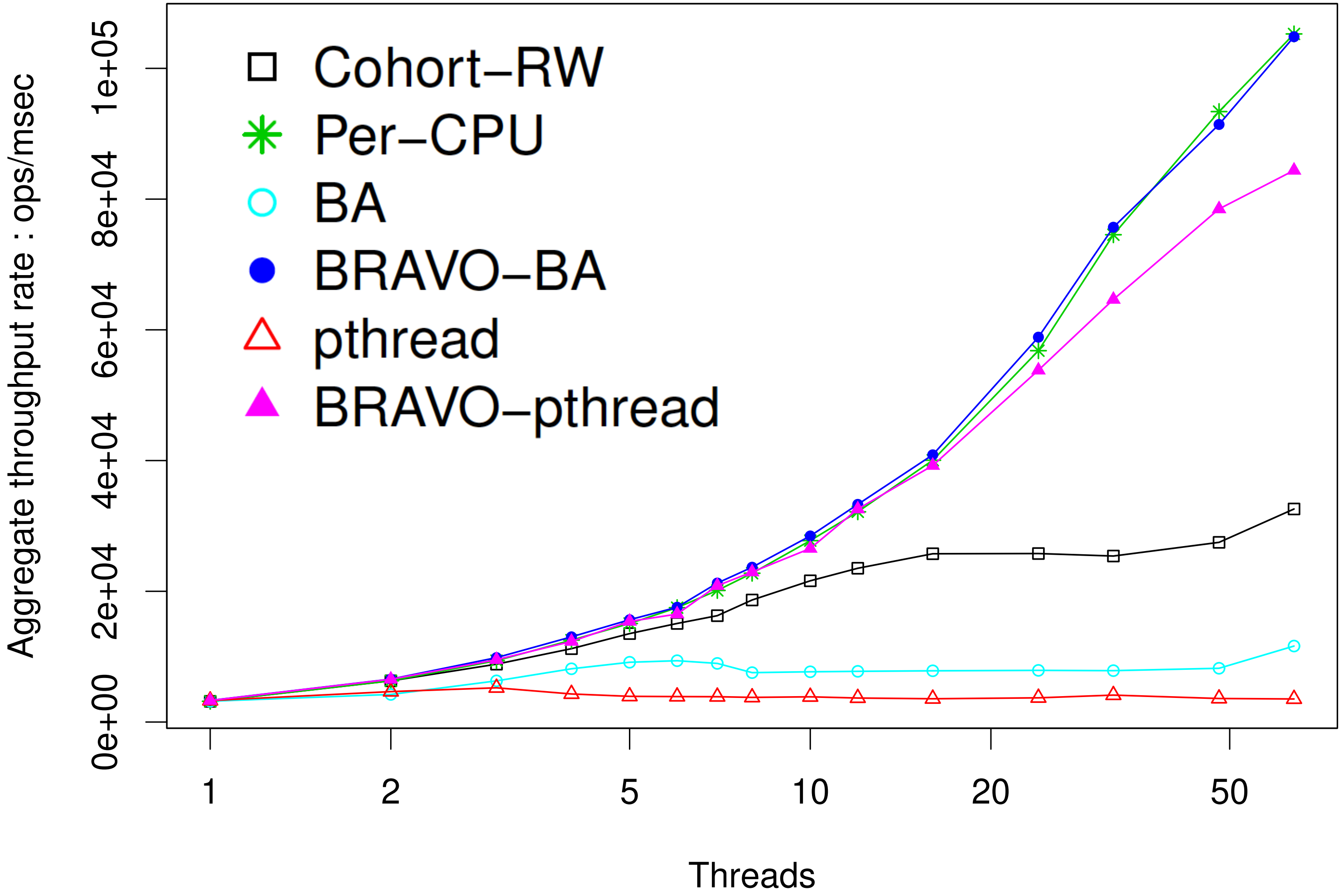
Locks	Memory footprint
BA	40
BA + BRAVO	40 + 12 + 32KB (for a table)
Per-CPU	9216 (on a system with 72 CPUs)
Cohort-RW	896 (dual-socket)

Intel Xeon E5-2699 v3 CPU
2 sockets
72 logical CPUs in total

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Evaluation: Accelerates reads

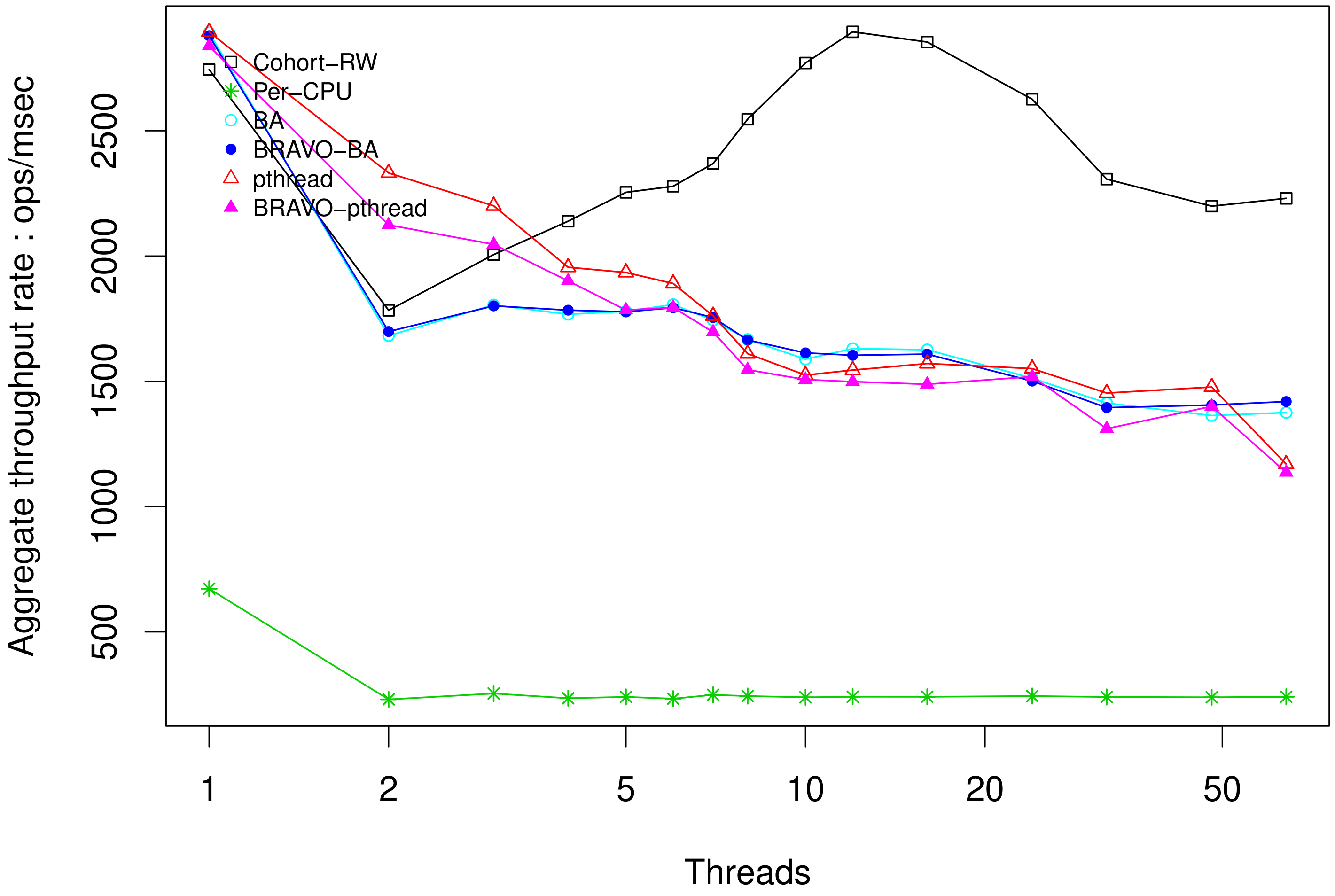
RWBench with 1 out of every 10000 are writes



Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Evaluation: Handles writes gracefully

RWBench with 9 out of every 10 are writes



Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Linux Kernel rwsem

- Counter + waiting queue protected by a spin lock
- Reader atomically increments the counter and checks its value

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Linux Kernel rwsem

- Counter + waiting queue protected by a spin lock
- Reader atomically increments the counter and checks its value
- Synchronization bottleneck in the kernel (mmap_sem)

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Linux Kernel rwsem

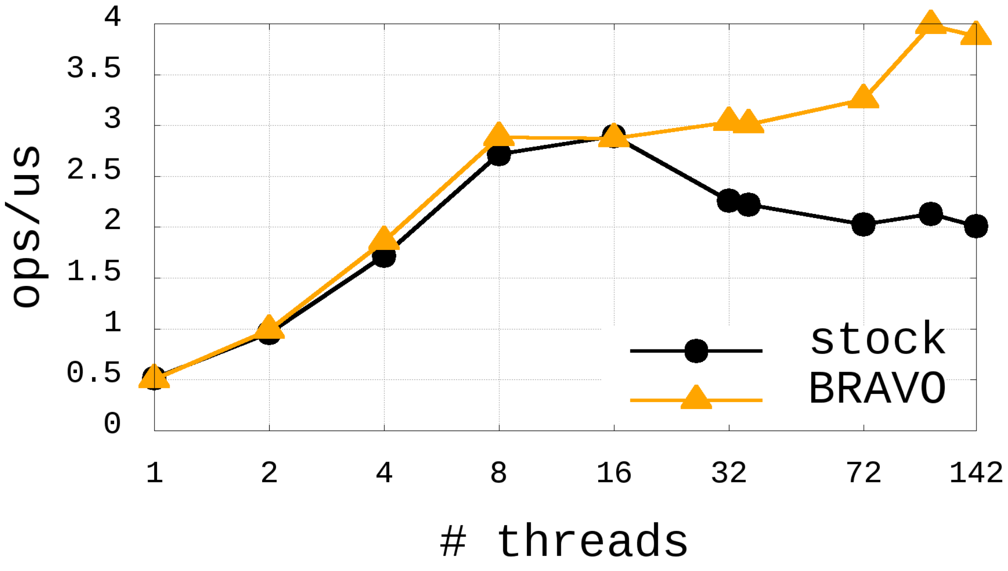
- Counter + waiting queue protected by a spin lock
- Reader atomically increments the counter and checks its value
- Synchronization bottleneck in the kernel (mmap_sem)
- Stress-test with will-it-scale: page_fault and mmap

ORACLE

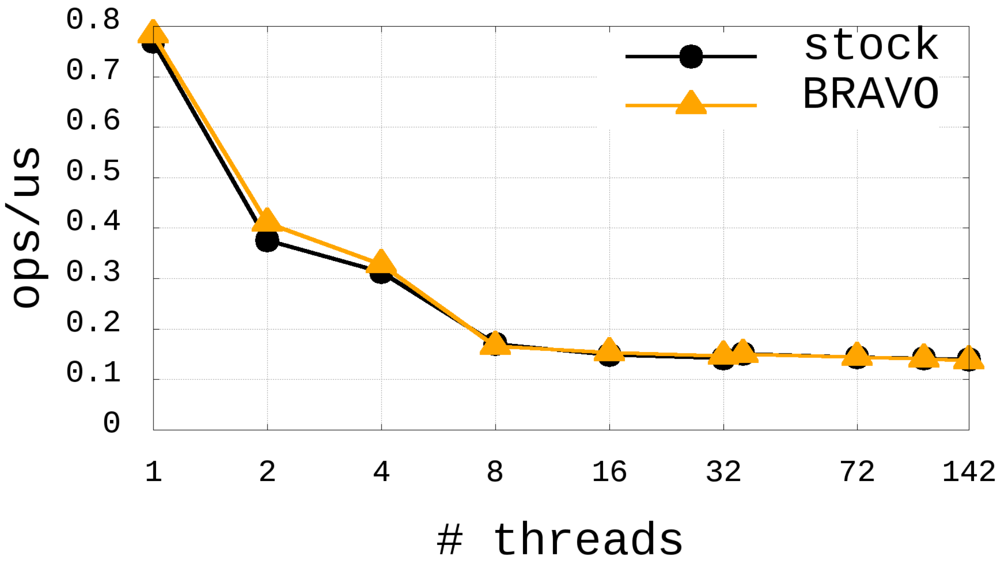
Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Evaluation with will-it-scale

Intel Xeon E7-8895 v3 CPU
4 sockets
144 logical CPUs in total



page_fault

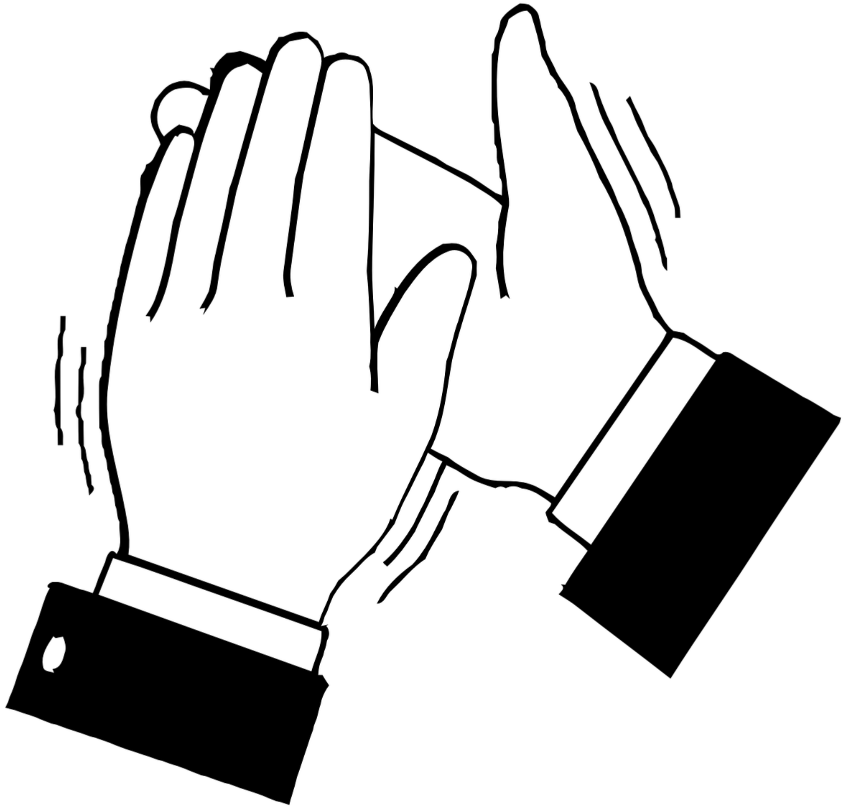


mmap

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Conclusion

- Builds into any existing lock
- Reads are accelerated
- Avoids write overhead
- Very compact
- Overall, takes the “reader indicator” dilemma away



Thank you! Questions?

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |

Future Work

- Dynamic table sizing
- Probing multiple table locations
- Adaptive policies for enabling bias
- Revocation scan via SIMD instructions and non-temporal loads

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. |