# Octet: Capturing and Controlling Cross-Thread Dependences Efficiently


Michael Bond
Milind Kulkarni — Purdue
Man Cao
Minjia Zhang
Meisam Fathi Salmi
Ohio State
Swarnendu Biswas
Aritra Sengupta
Jipeng Huang

# Parallel programming is mainstream

Shared memory with locks

Challenge:
performance & correctness

Need practical runtime support

- Help express parallelism better
- Eliminate concurrency errors
- Diagnose production bugs
- Deal with nondeterminism

Need practical runtime support

- Atomicity checking
- Data race detection
- Record & replay
- Transactional memory
- DRF/SC enforcement
- Deterministic execution

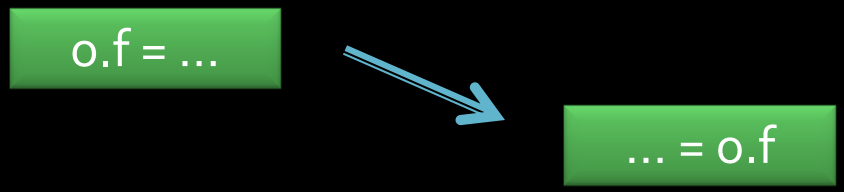
# Need practical runtime support

- Atomicity checking
- Data race detection
- Record & replay

Track dependences

- Transactional memory
- DRF/SC enforcement
- Deterministic execution

Control dependences



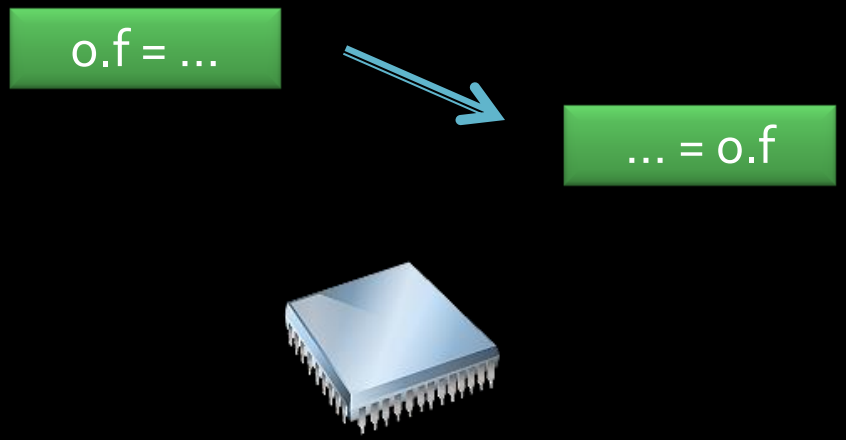
# Need practical runtime support

- Atomicity checking
- Data race detection
- Record & replay

Track dependences

- Transactional memory
- DRF/SC enforcement
- Deterministic execution

Control dependences



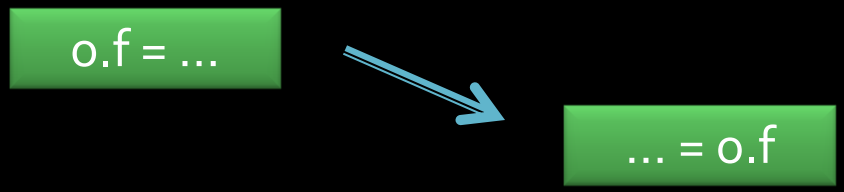
Need practical runtime support

- Atomicity checking
- Data race detection
- Record & replay

Track dependences

- Transactional memory
- DRF/SC enforcement
- Deterministic execution

Control dependences




Commodity (software-only) approaches
slow programs by several times

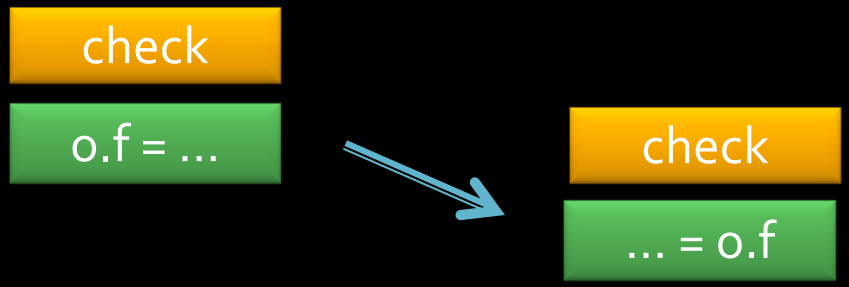
Need practical runtime support

- Atomicity checking
- Data race detection
- Record & replay

Track dependences

- Transactional memory
- DRF/SC enforcement
- Deterministic execution

Control dependences




Commodity (software-only) approaches
slow programs by several times

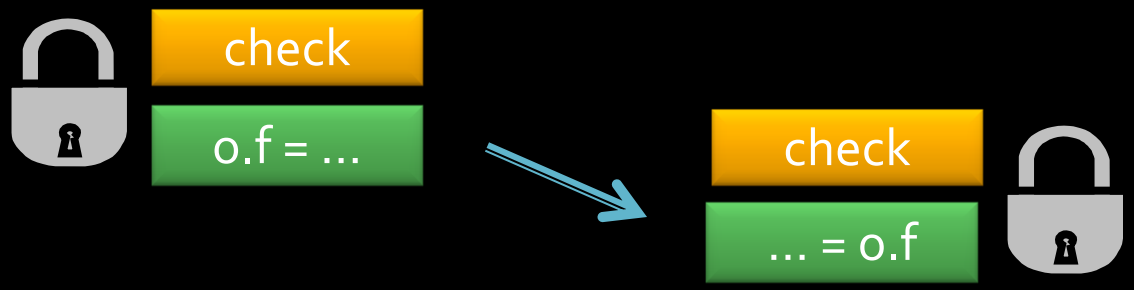
Need practical runtime support

- Atomicity checking
- Data race detection
- Record & replay

Track dependences

- Transactional memory
- DRF/SC enforcement
- Deterministic execution

Control dependences




Any access could race →
add synchronization at every access

# Octet

Framework for runtime support

HB edges → all dependences

Atomicity of analysis & access

Concurrency control mechanism

Synchronization ⇔ cross-thread dependence

→ Qualitative performance improvement

# Octet

Framework for runtime support

HB edges → all dependences

Atomicity of analysis & access


Concurrency control mechanism

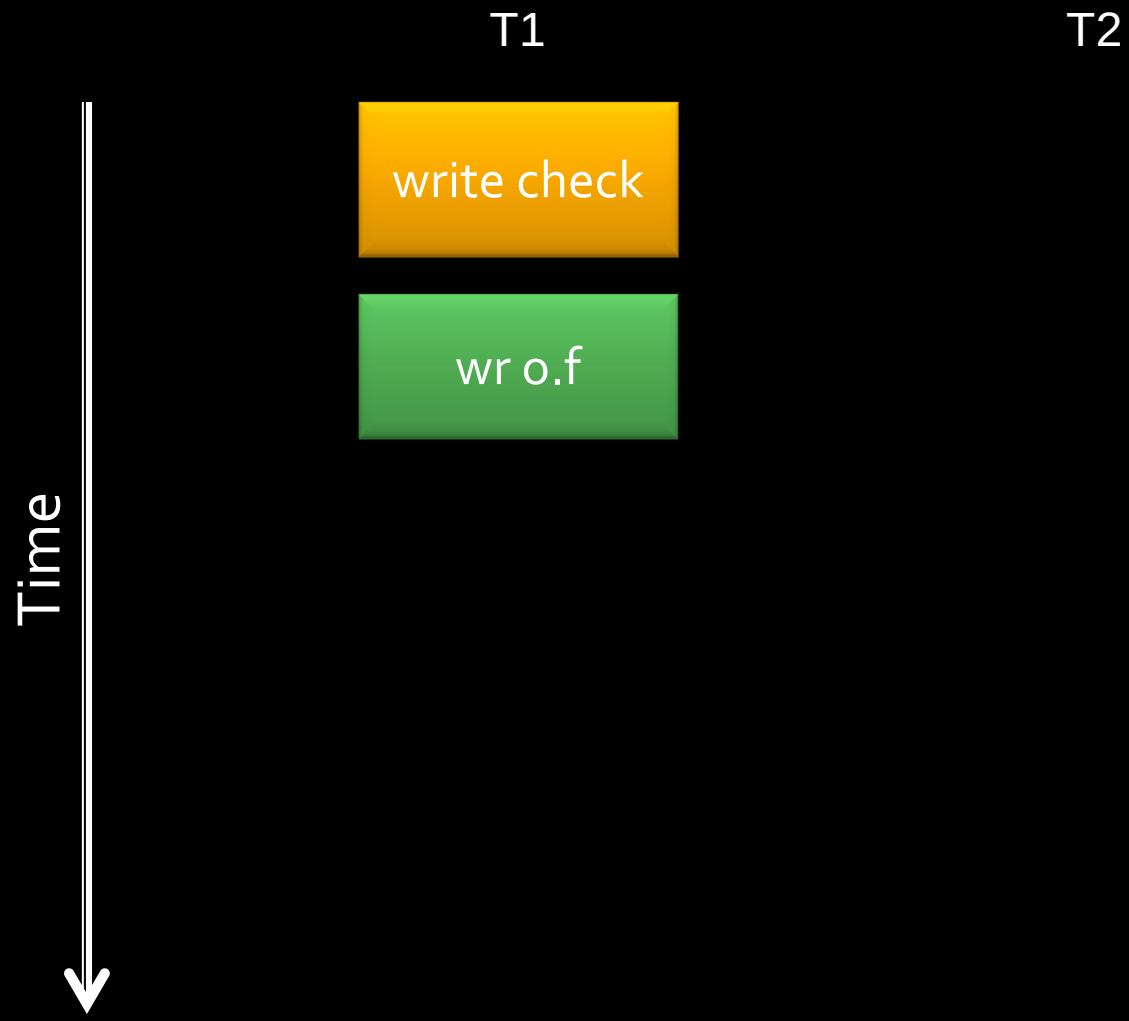
Synchronization ⇔ cross-thread dependence

→ Qualitative performance improvement

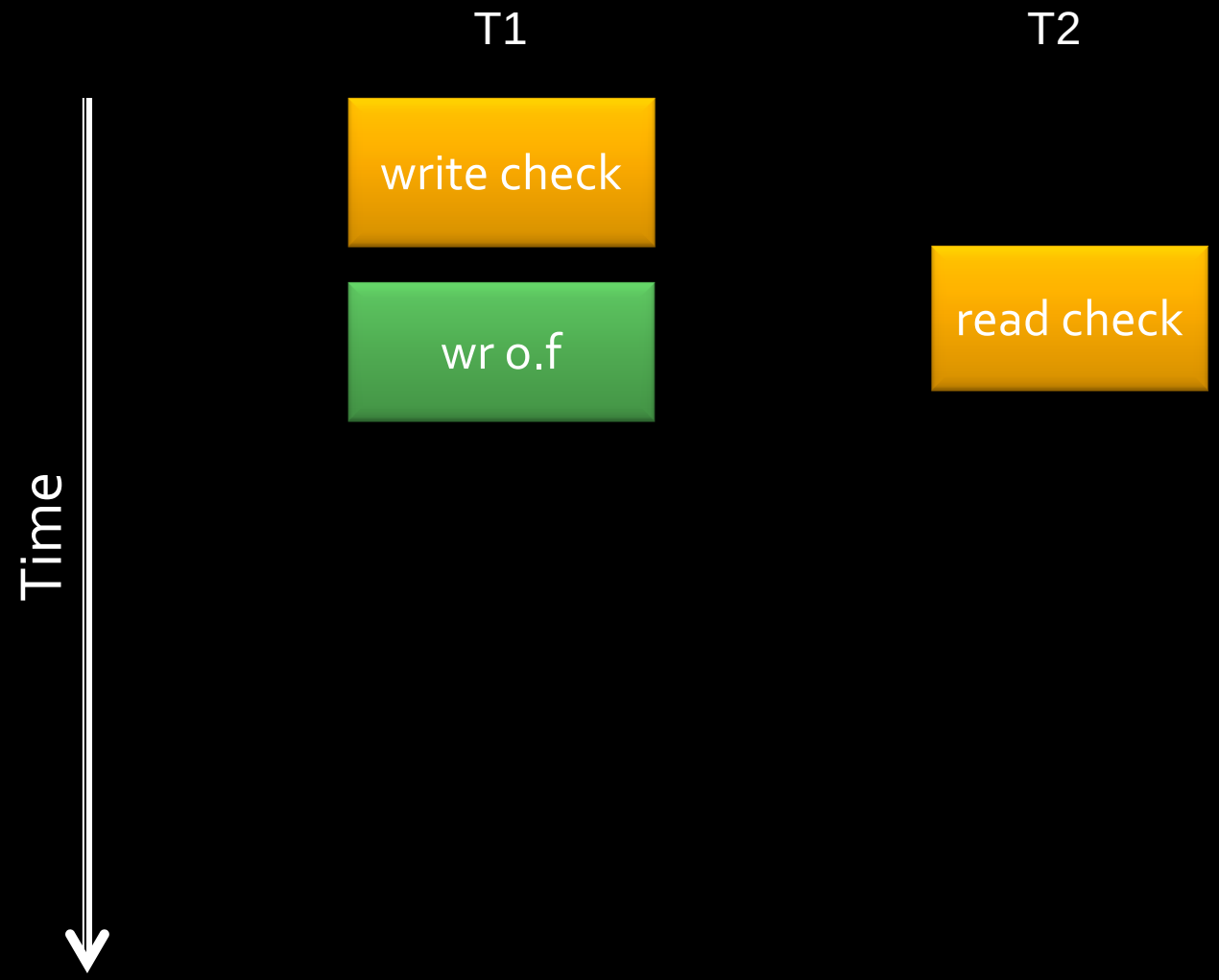
# Octet tracks ownership

Each object's **state** $\in \{ \mathrm{WrEx}_T, \mathrm{RdEx}_T, \mathrm{RdSh}_c \}$

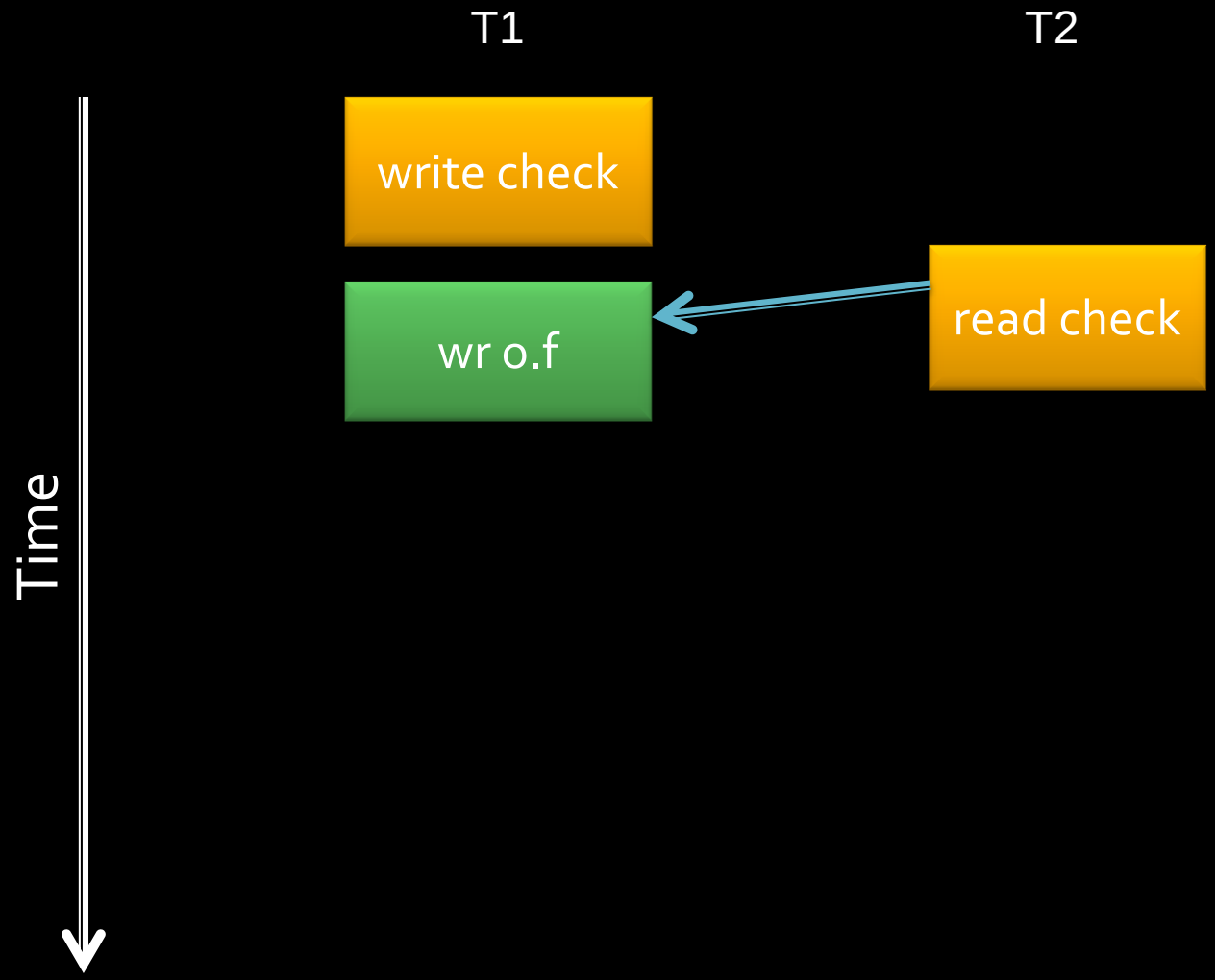
o's state = $WrEx_{T1}$



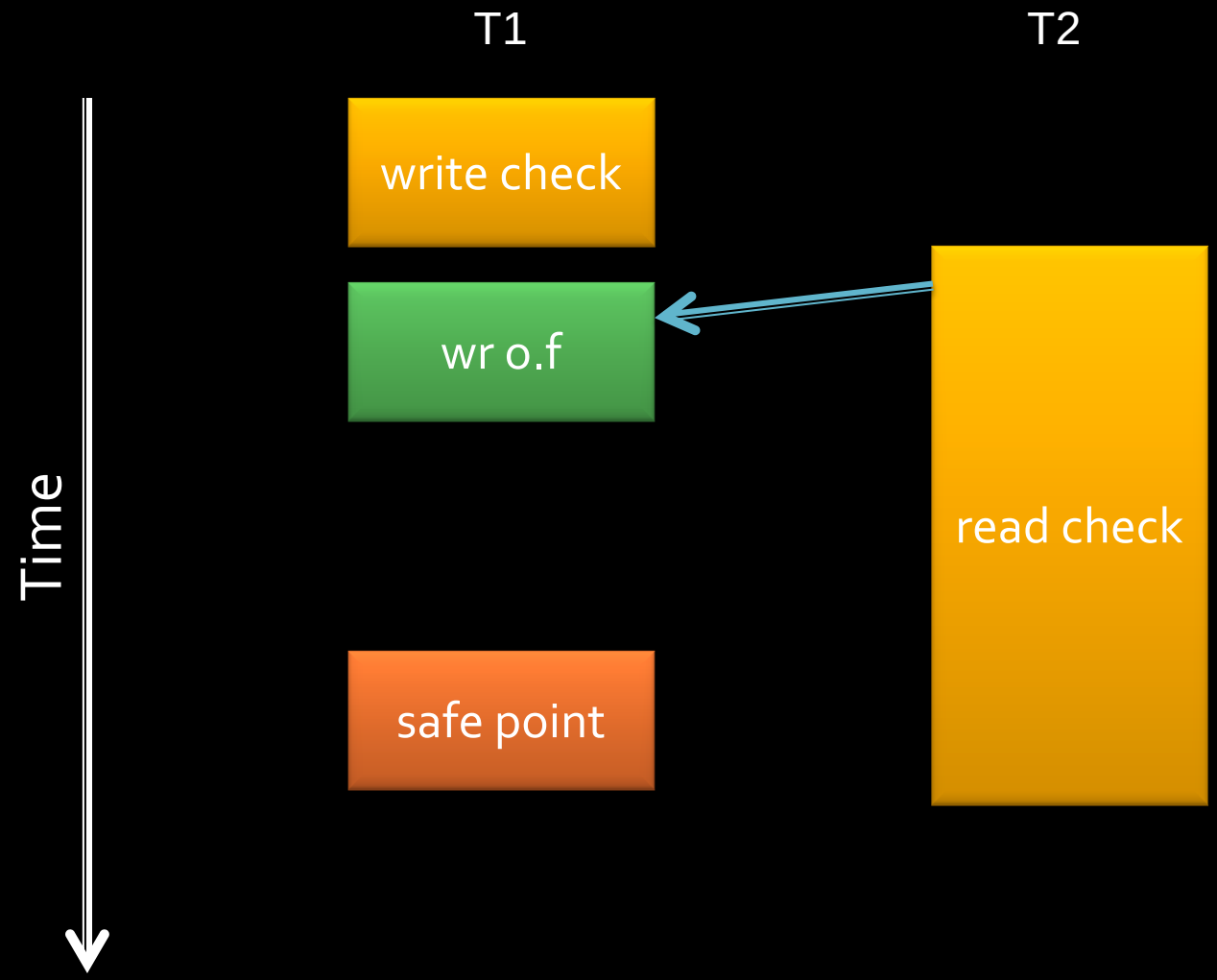
o's state = $WrEx_{T1}$



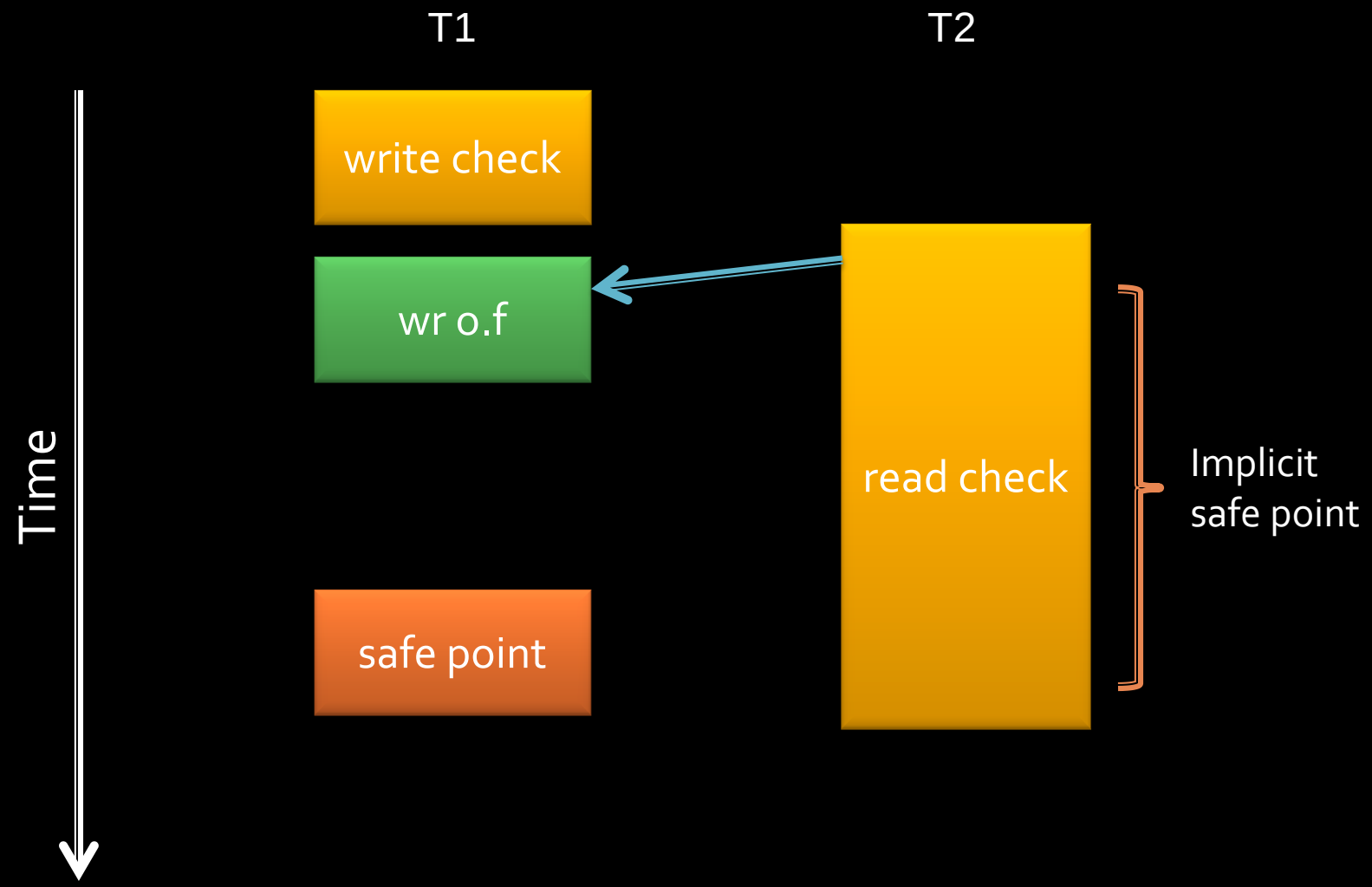
o's state = $WrEx_{T1}$



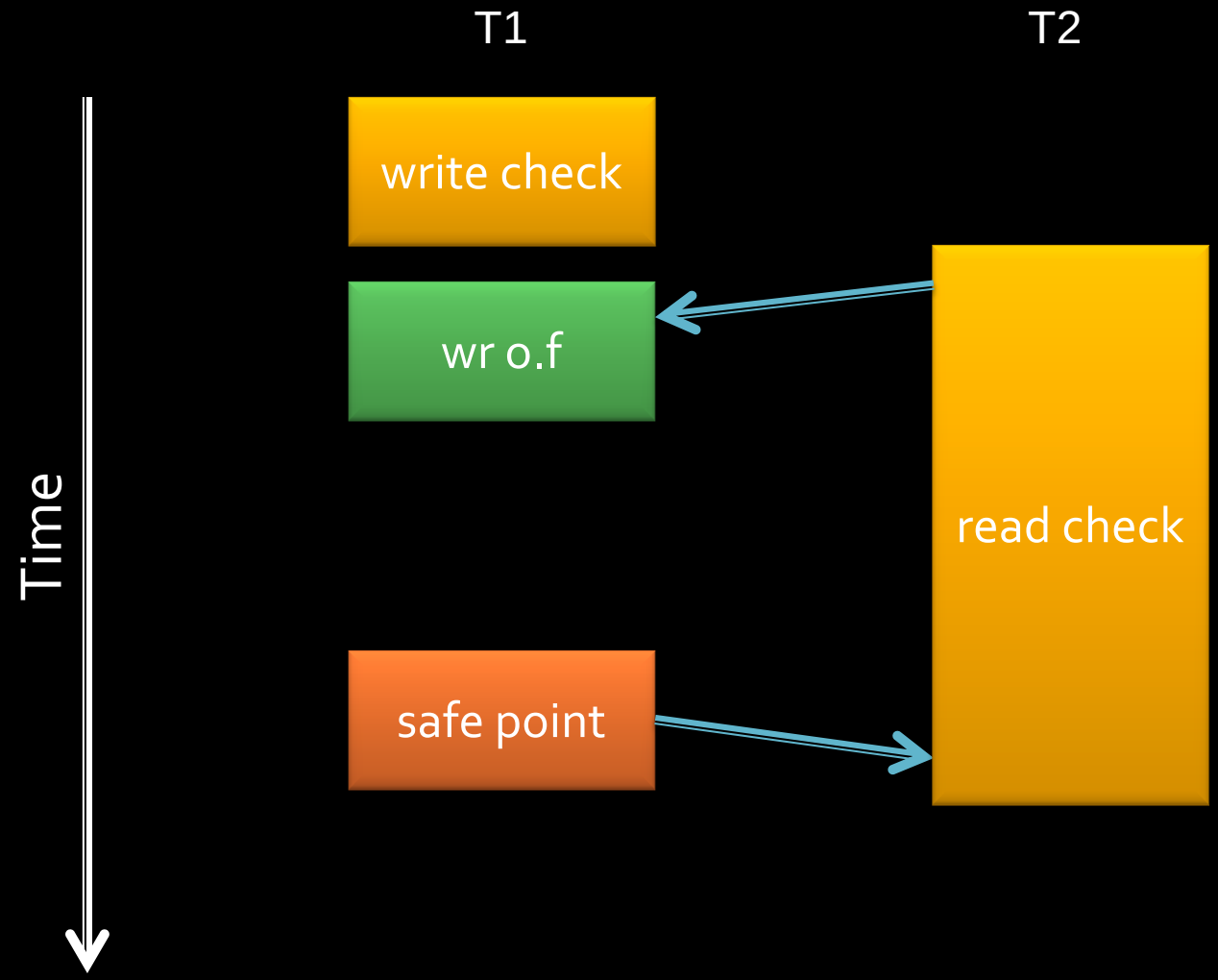
o's state = $WrEx_{T1}$



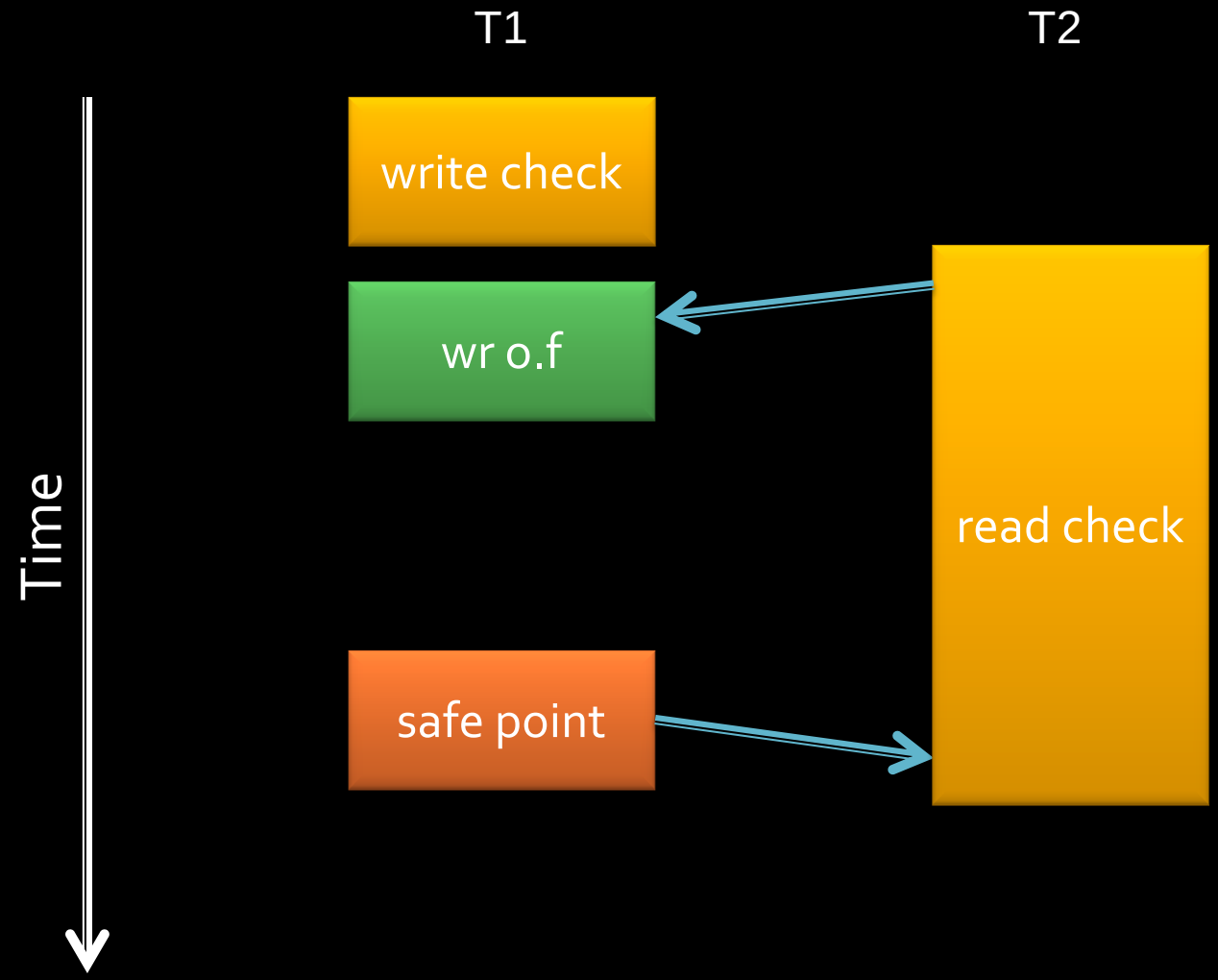
o's state = $WrEx_{T1}$



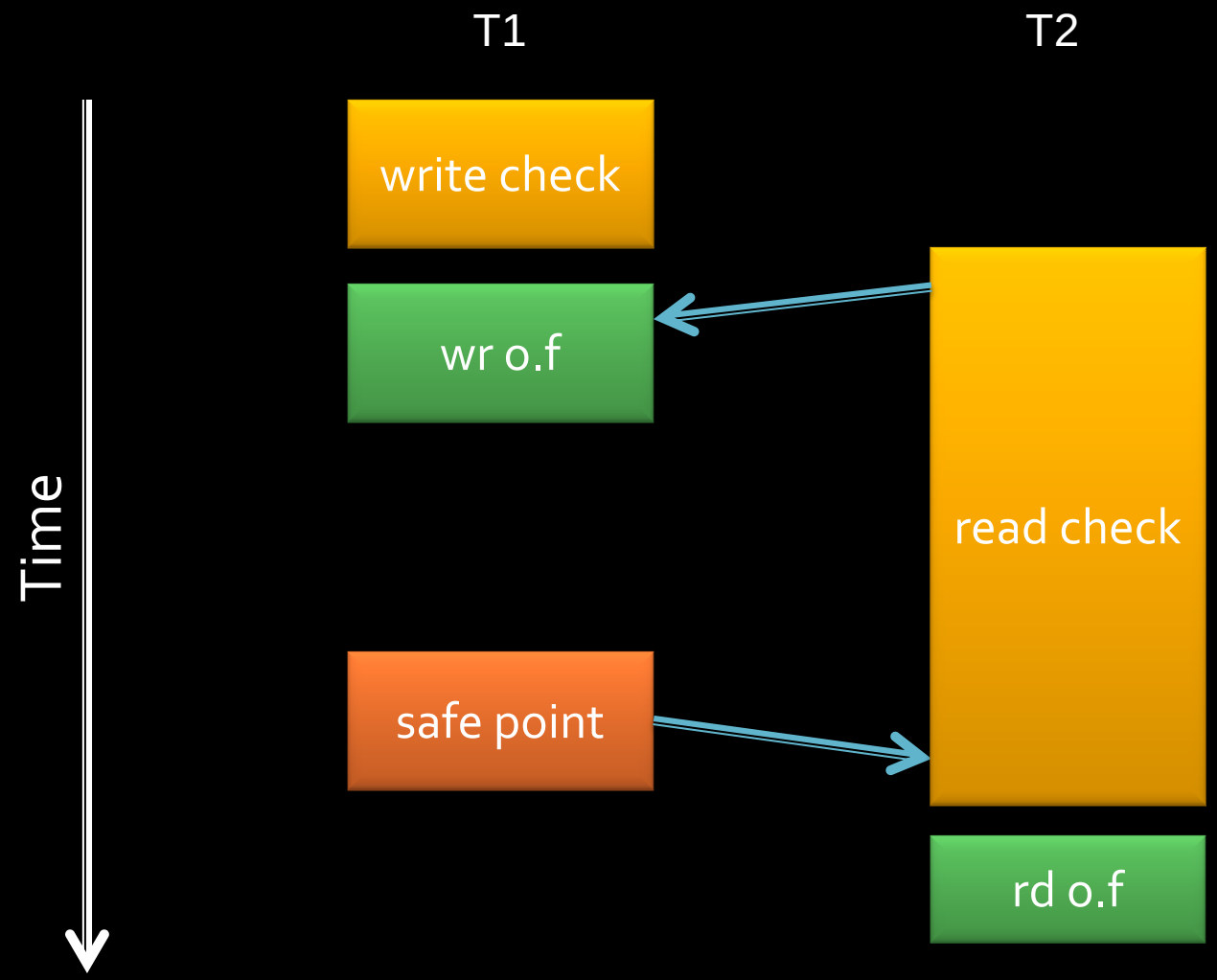
o's state = $WrEx_{T1}$



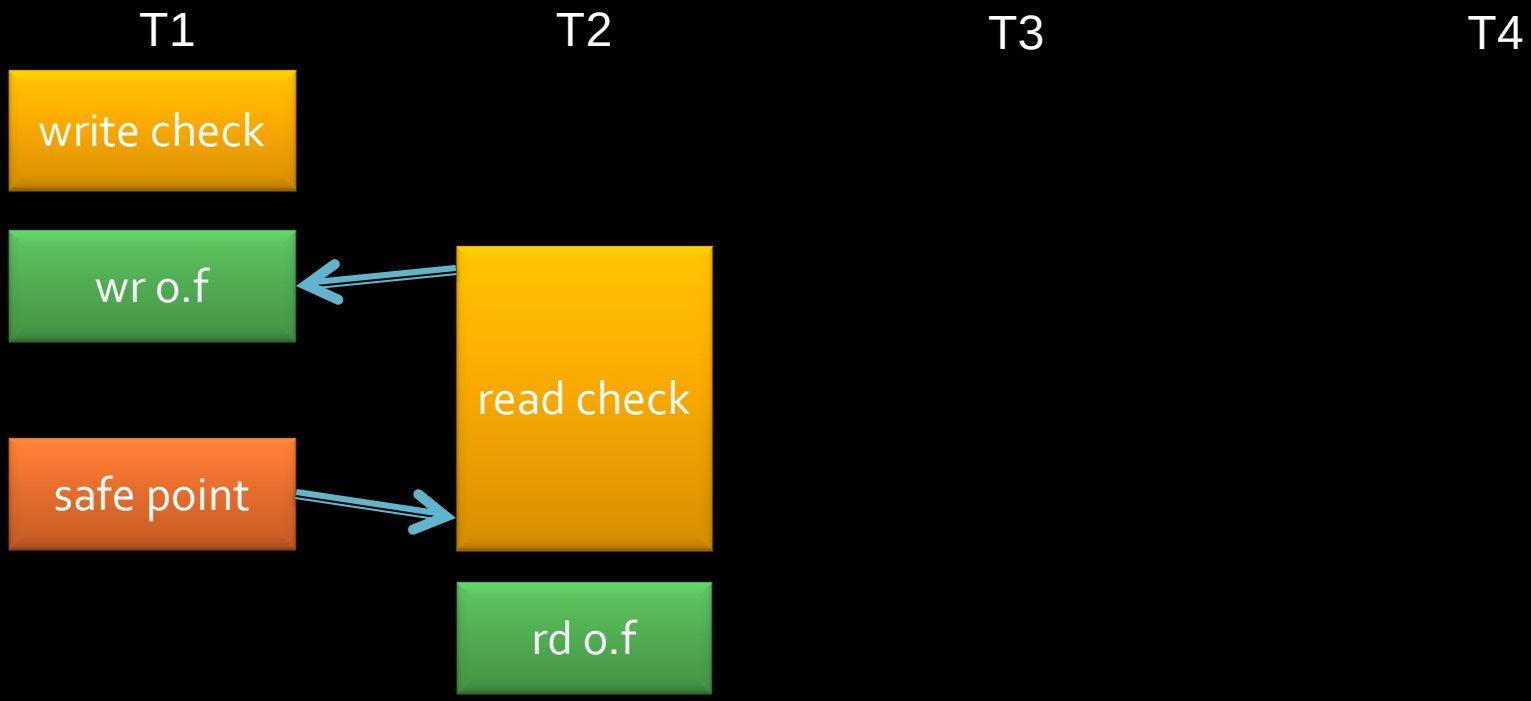
o's state = $RdEx_{T2}$



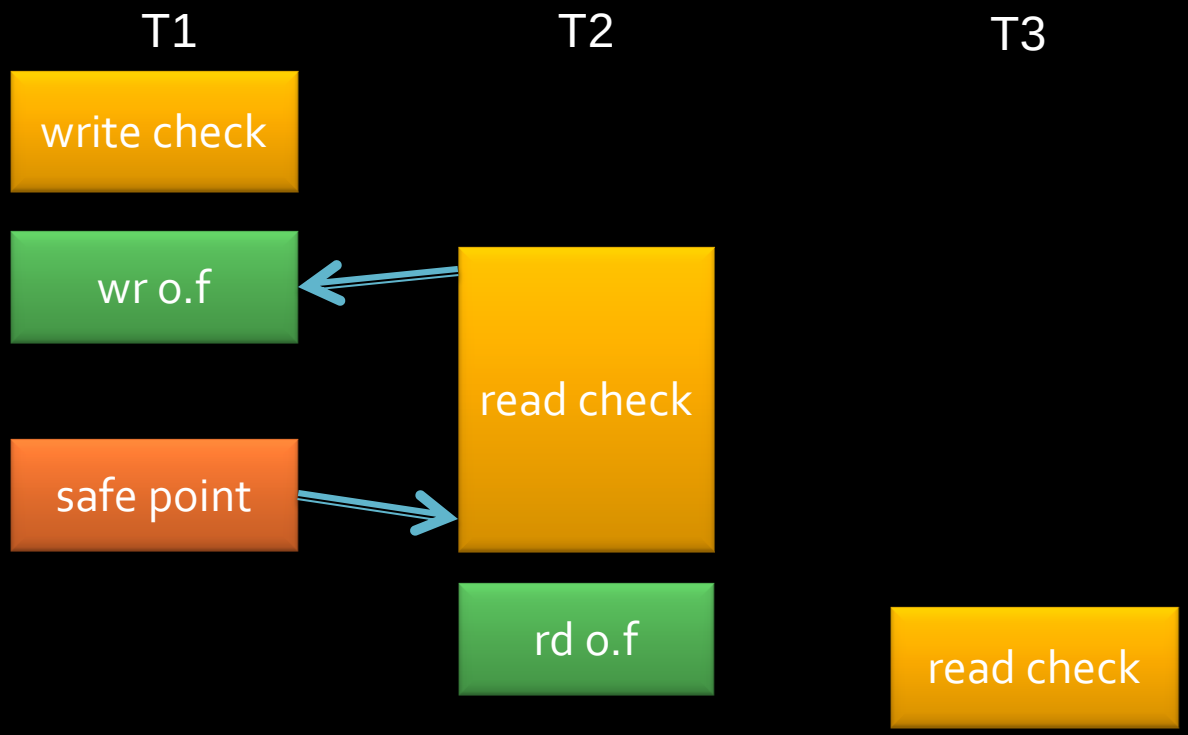
o's state = $RdEx_{T2}$



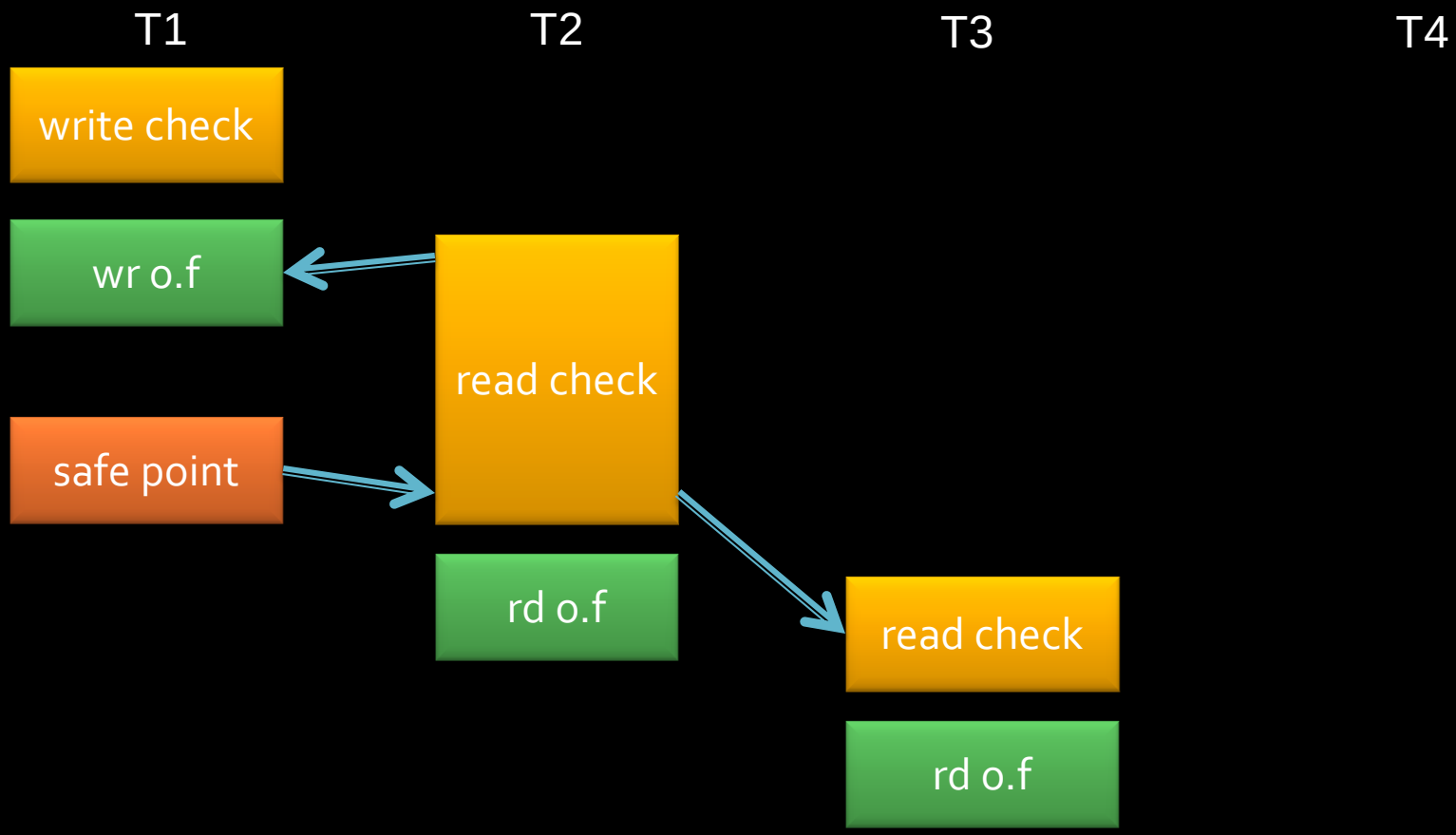
o's state = $RdEx_{T2}$



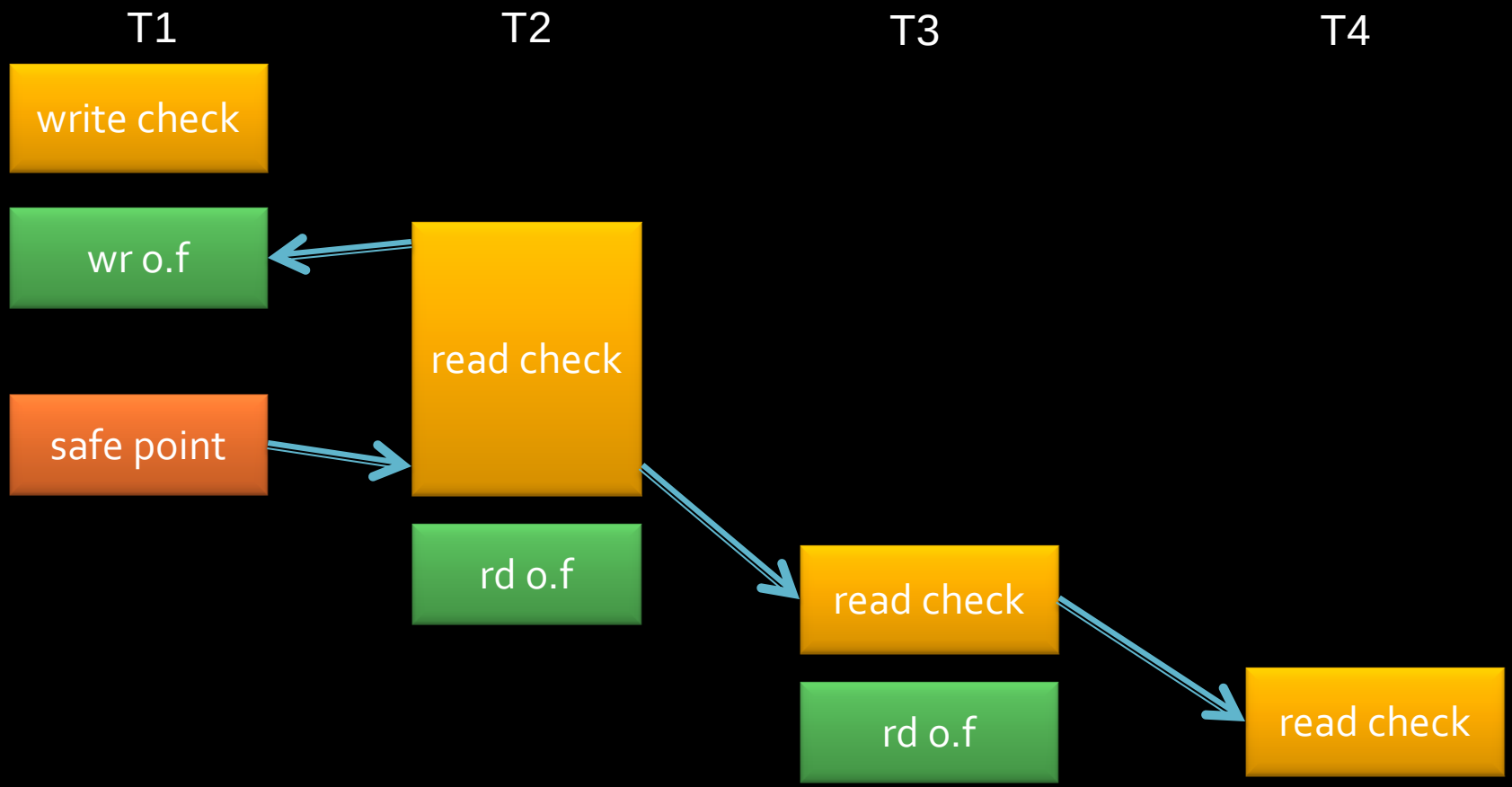
o's state = $RdEx_{T2}$



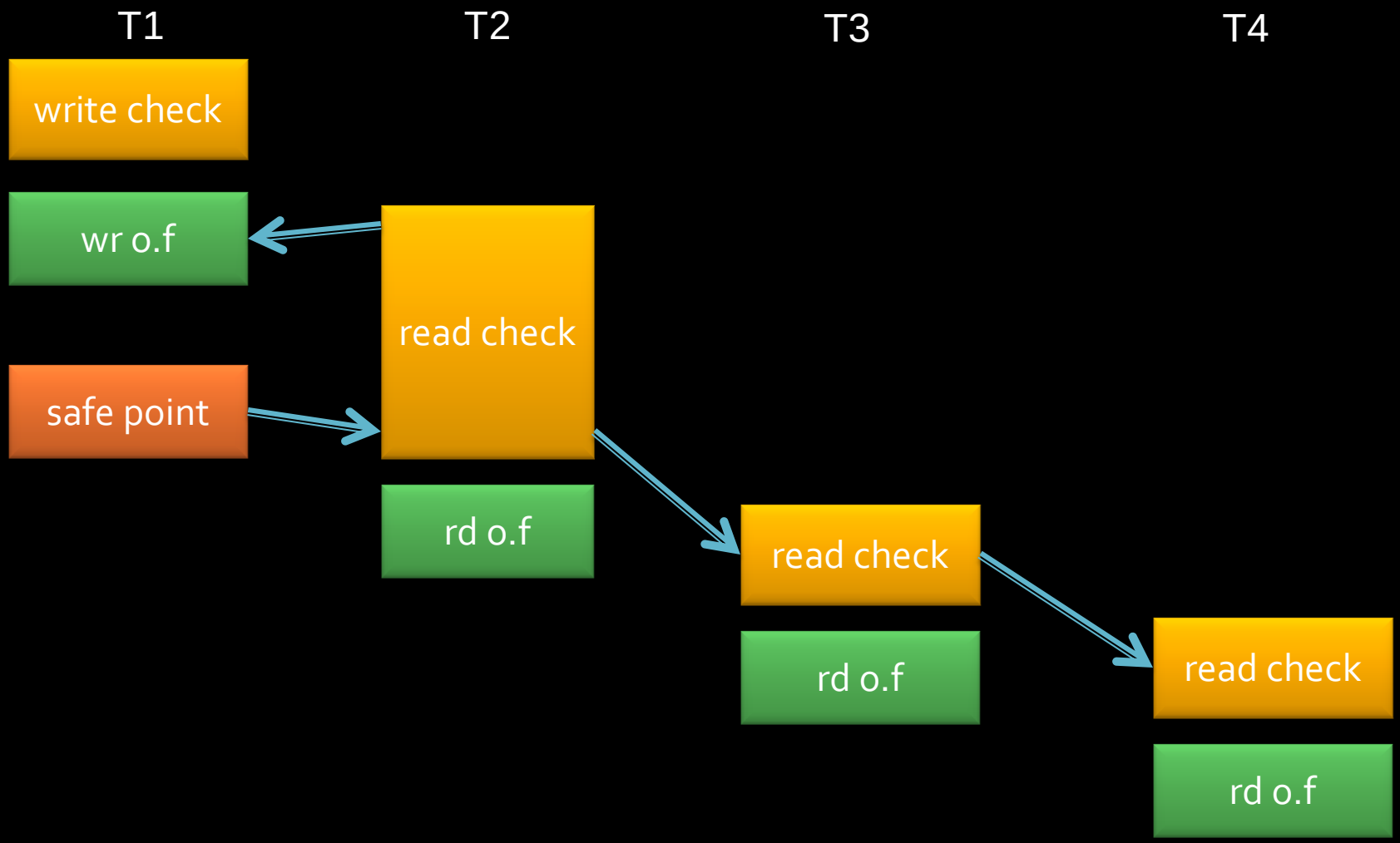
# o's state = $RdSh_c$



# o's state = $RdSh_c$



# o's state = $RdSh_c$



o's state = $RdSh_c$

T1 T2 T3 T4

write check

wr o.f

safe point

read check

read check

rd o.f

## Sharing detection

[von Praun & Gross '01]

*Comparison in our paper*

## Distributed shared memory

Shasta [Scales et al. '96]

## Biased locking

[Kawachiya et al. '02]

[Russell & Detlefs '06]

[Hindman & Grossman '06]

# Practical runtime support

- Atomicity checking
- Data race detection
- Record & replay

Track dependences

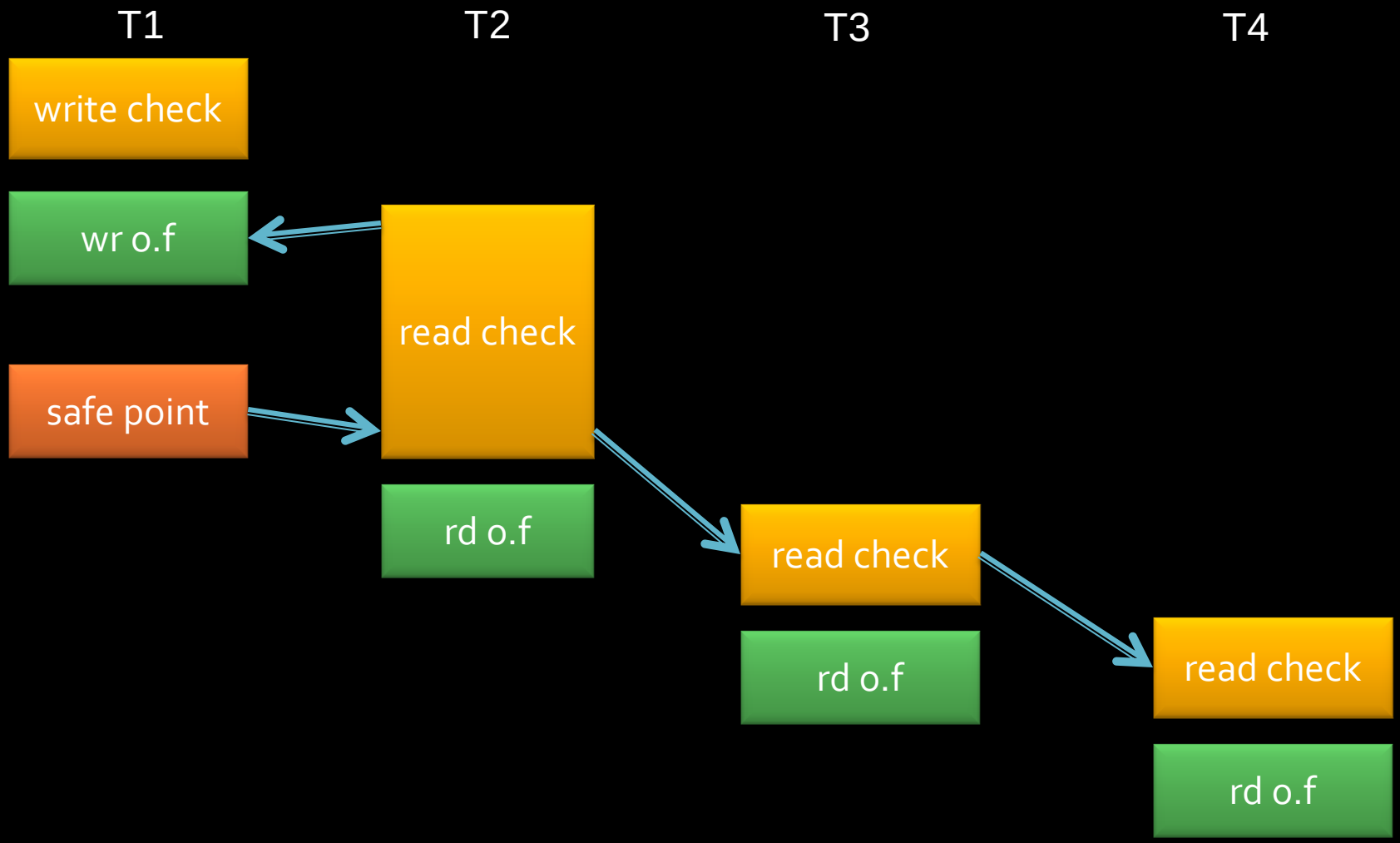
- Transactional memory
- DRF/SC enforcement
- Deterministic execution

Control dependences

Octet

Framework for runtime support
Concurrency control mechanism

# Dependence recorder records happens-before edges



## Implementation in Jikes RVM

Publicly available

http://jikesrvm.org/Research+Archive

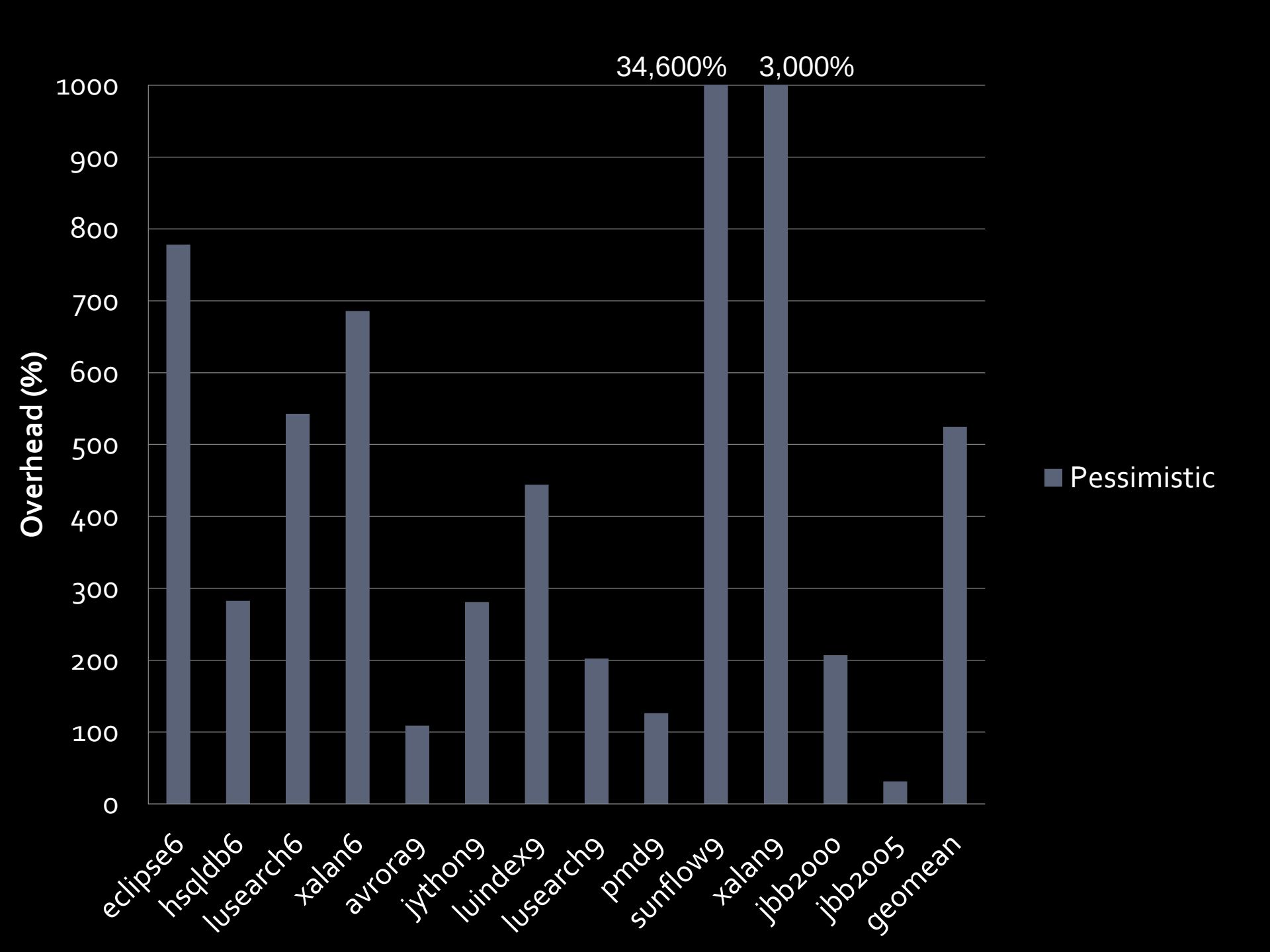
## Parallel programs

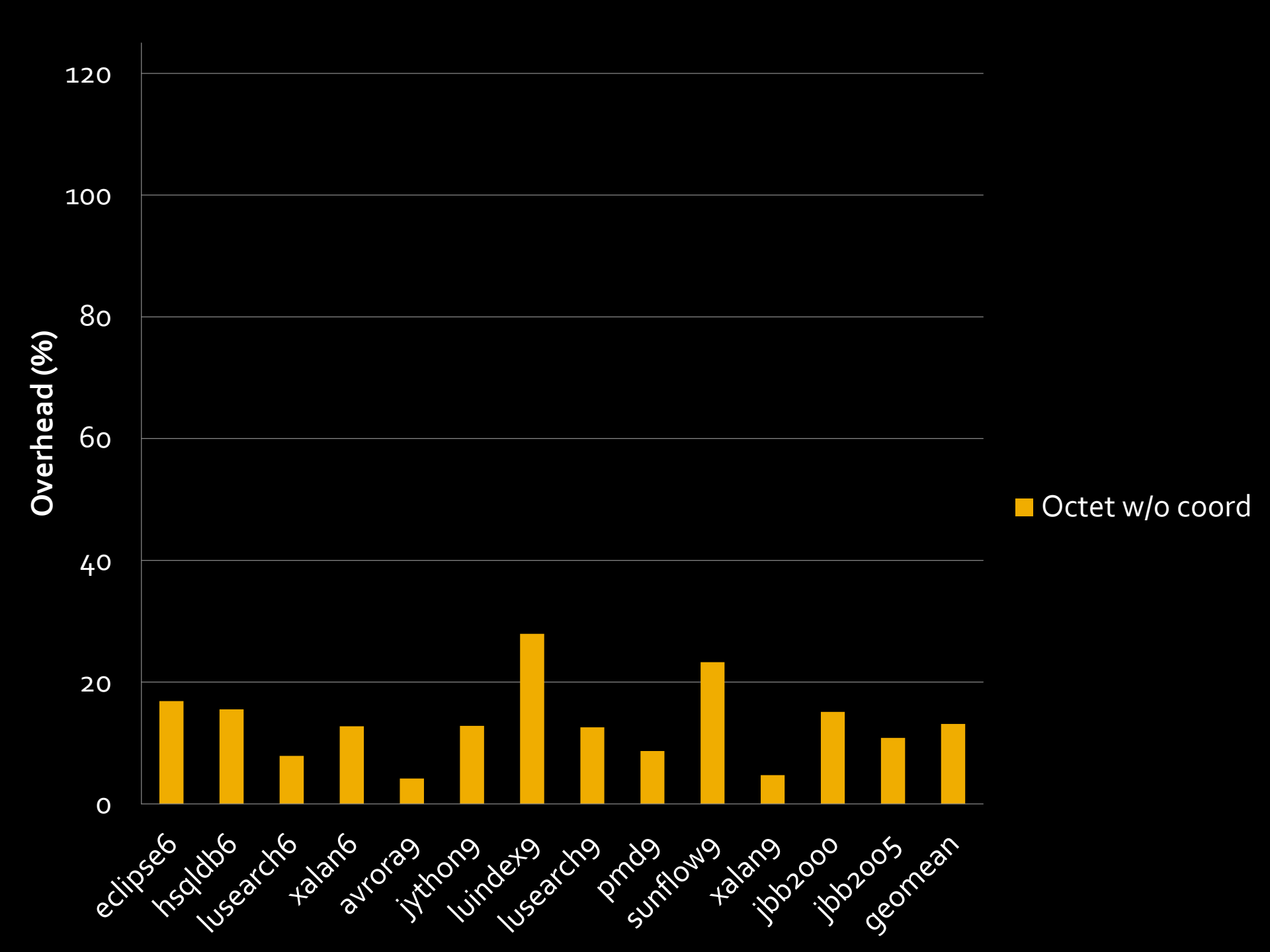
DaCapo Benchmarks 2006 & 2009

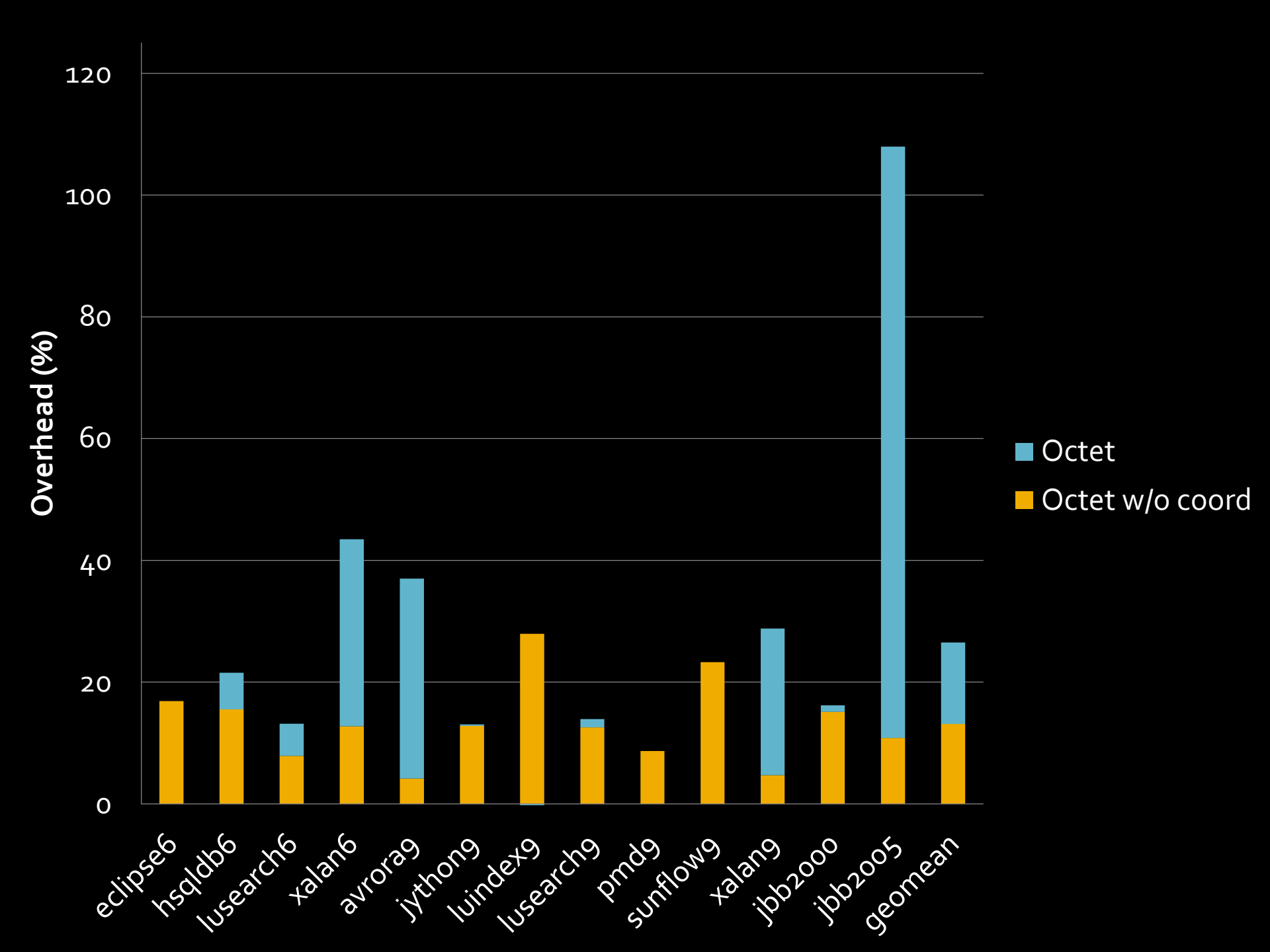
SPEC JBB 2000 & 2005

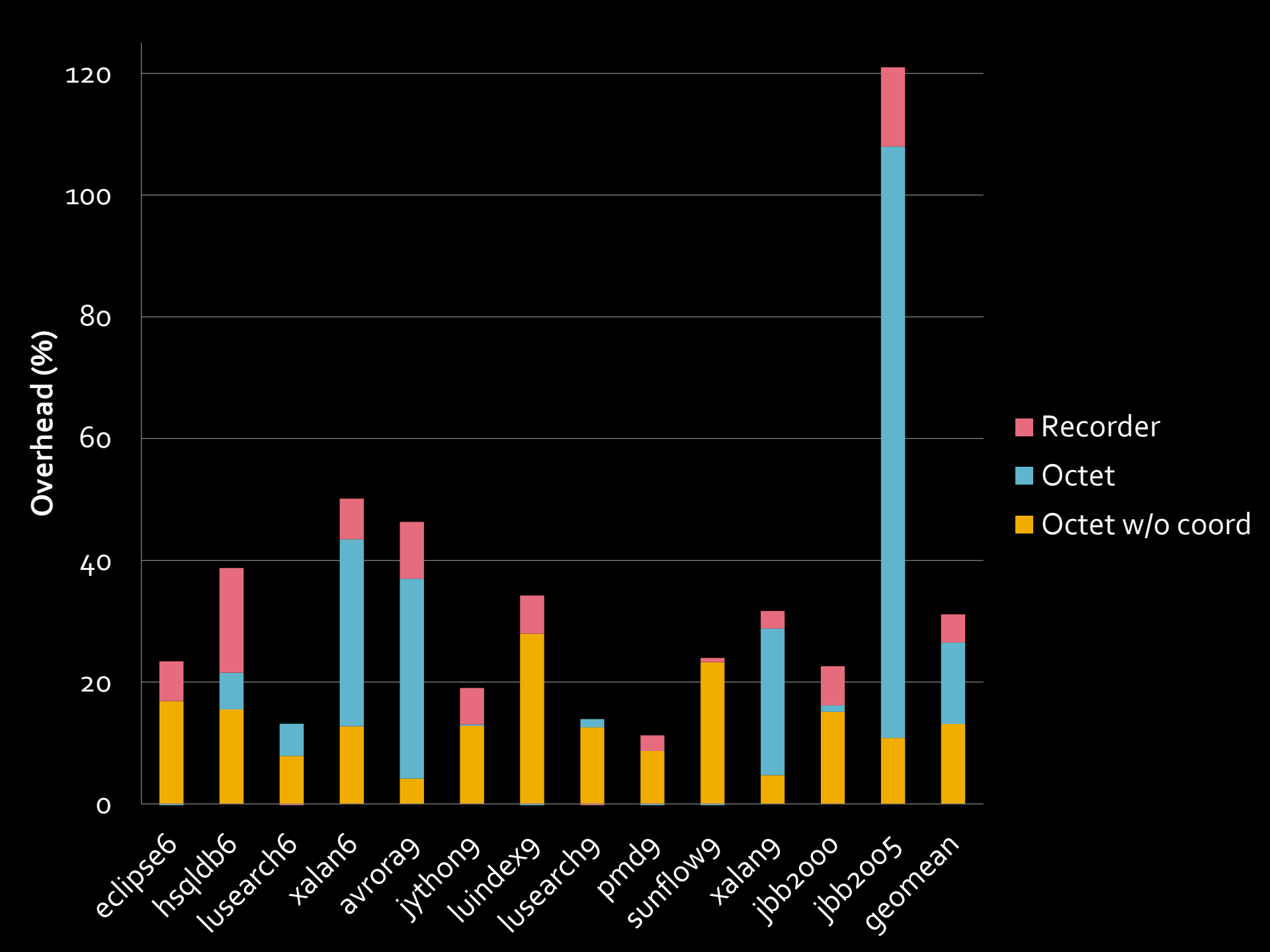
## Parallel platform

32 cores (AMD Opteron 6272)


34,600%
3,000%
1000
900
800
700
600
500
400
300
200
100
0
Overhead (%)
Pessimistic
eclipse6
hsqldb6
lusearch6
xalan6
avrora9
jython9
luindex9
lusearch9
pmd9
sunflow9
xalan9
jbb2000
jbb2005
geomean


Overhead (%)
120
100
80
60
40
20
0
Octet w/o coord
eclipse6
hsqldb6
lusearch6
xalan6
avrora9
jython9
luindex9
lusearch9
pmd9
sunflow9
xalan9
jbb2000
jbb2005
geomean


Overhead (%)
120
100
80
60
40
20
0
Octet
Octet w/o coord
eclipse6
hsqldb6
lusearch6
xalan6
avrora9
jython9
luindex9
lusearch9
pmd9
sunflow9
xalan9
jbb2000
jbb2005
geomean


Overhead (%)
120
100
80
60
40
20
0
Recorder
Octet
Octet w/o coord
eclipse6
hsqldb6
lusearch6
xalan6
avrora9
jython9
luindex9
lusearch9
pmd9
sunflow9
xalan9
jbb2000
jbb2005
geomean

Octet helps enable practical runtime support for reliable, scalable concurrency

Framework for runtime support

HB edges → all dependences

Atomicity of analysis & access

Concurrency control mechanism

Synchronization ⇔ cross-thread dependence

→ Qualitative performance improvement