Wait-Free Data Exchange in Massively Threaded VR Systems

Patrick Lange, Rene Weller, Gabriel Zachmann

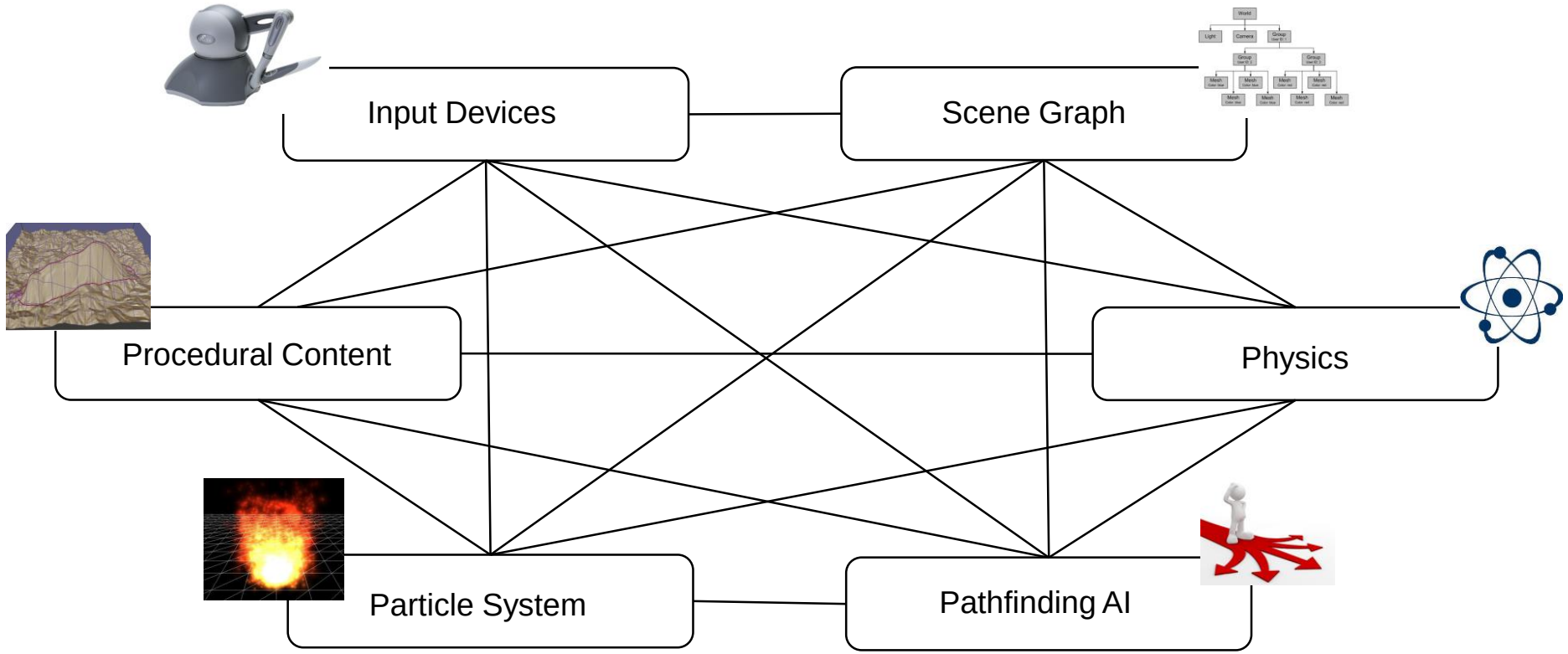
University of Bremen, Germany

cgvr.cs.uni-bremen.de

WSCG 2014, June 2014, Plzen

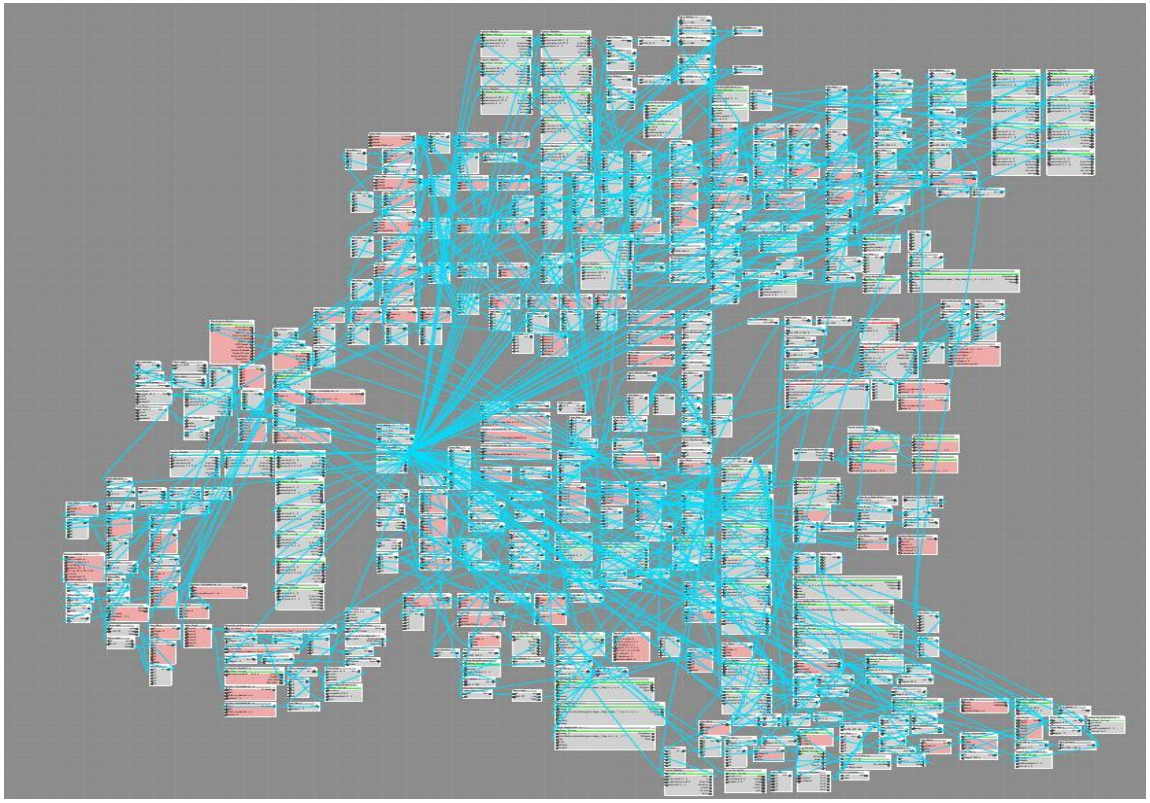
Data Flows in Classic VEs

- Modern virtual environments and games usually consist of many different components
- Classic approach: fields-and-routes-based data flow paradigm



Data Flow in Classic VEs

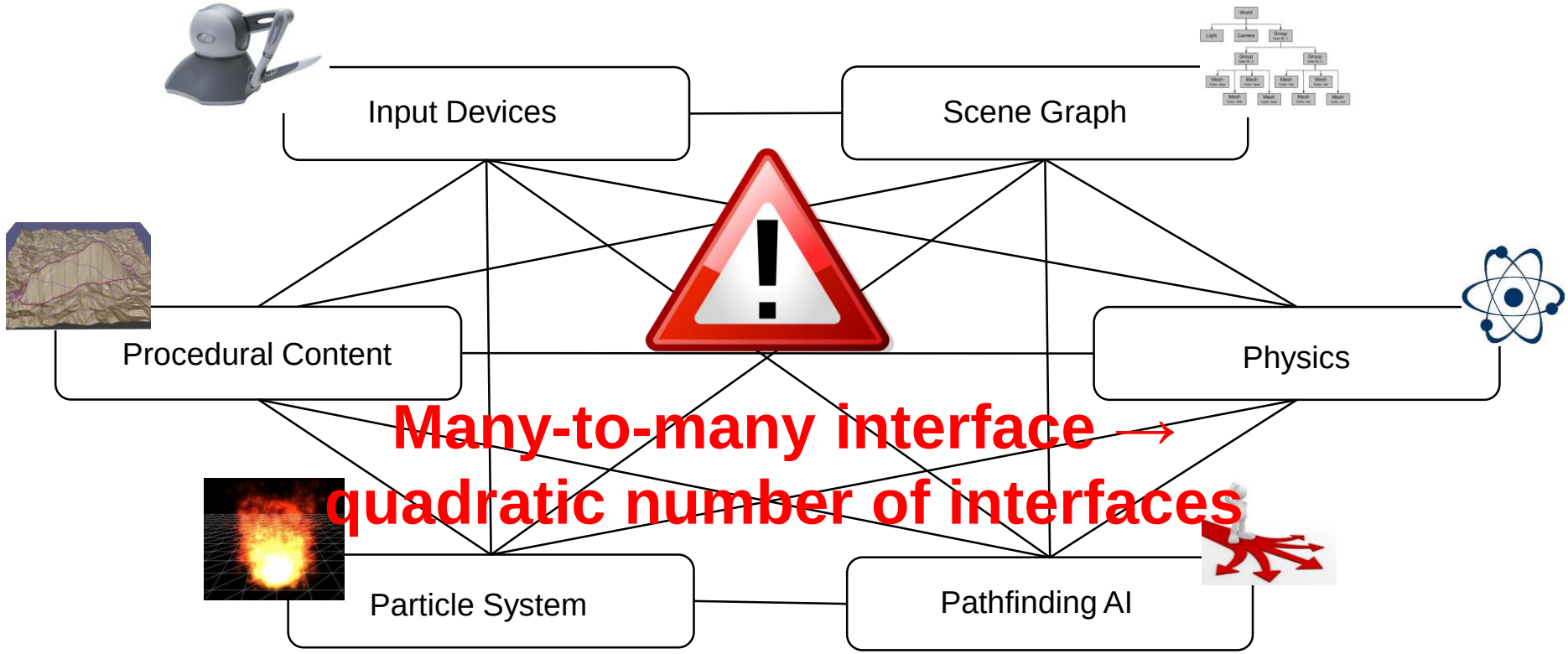
- Modern virtual environments and games usually consist of many different components
- Classic approach: fields-and-routes-based data flow paradigm



(CryEninge 3: Dynamic Player Movement, http://www.crydev.net/viewtopic.php?t=37675)

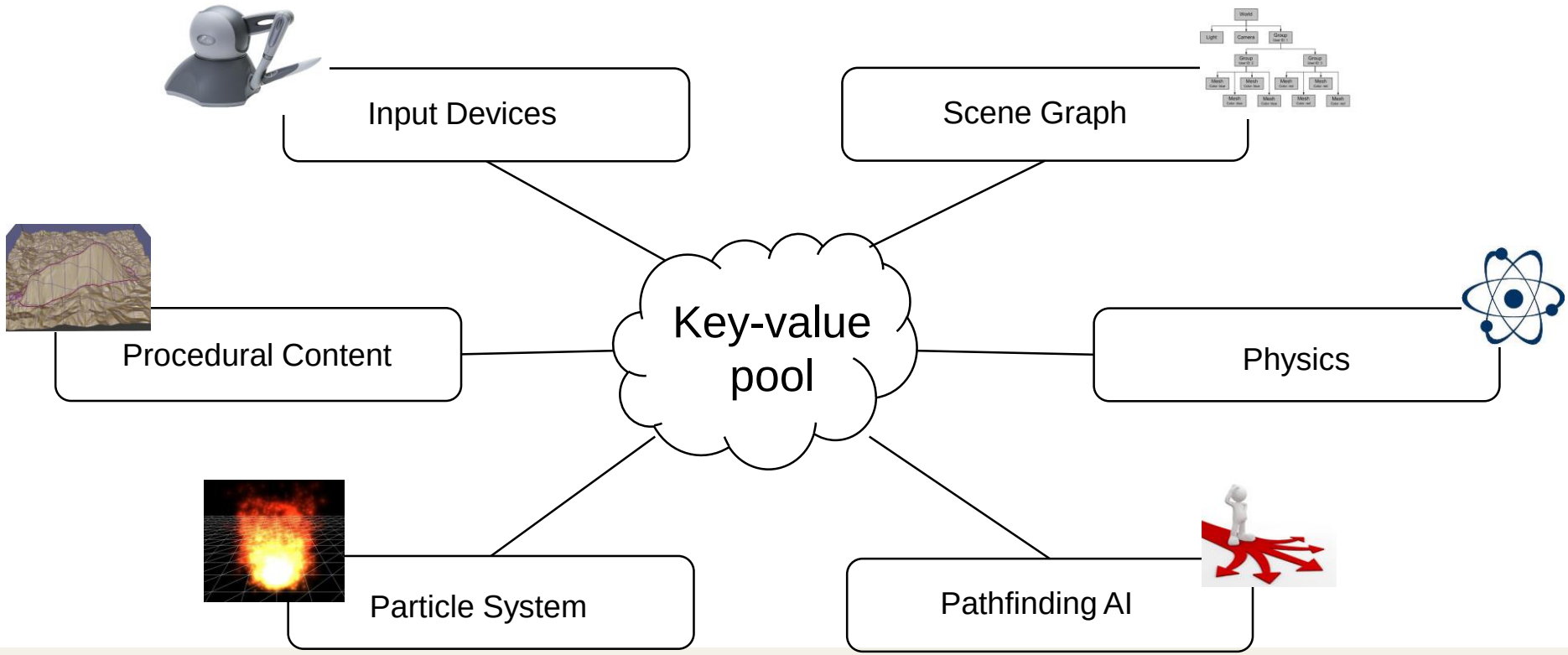
Data Flows in Classic VEs

- Modern virtual environments and games usually consist of many different components
- Classic approach: fields-and-routes-based data flow paradigm



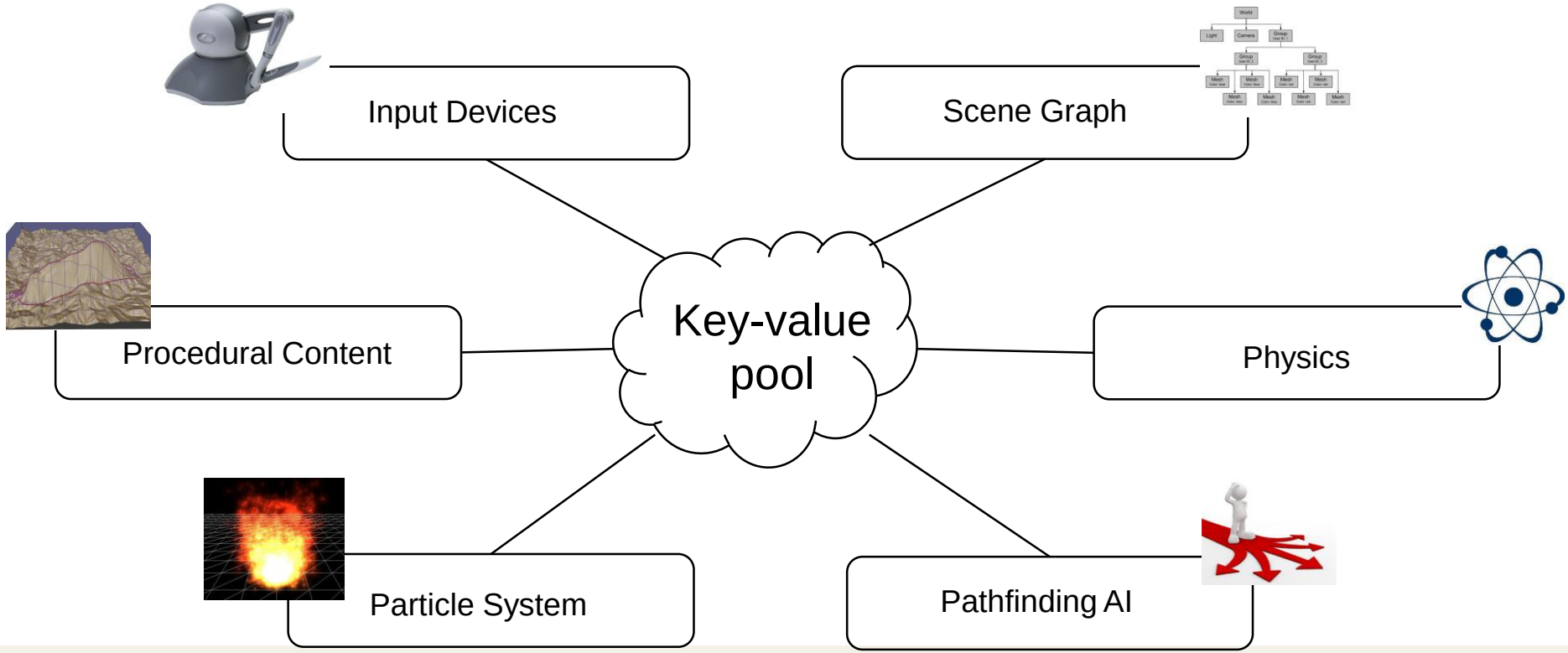
Our Novel Approach: the Key-Value Pool

- Assignment of unique key-value pair to each data packet which is exchanged between the components
- Key-value pool holds complete shared world state

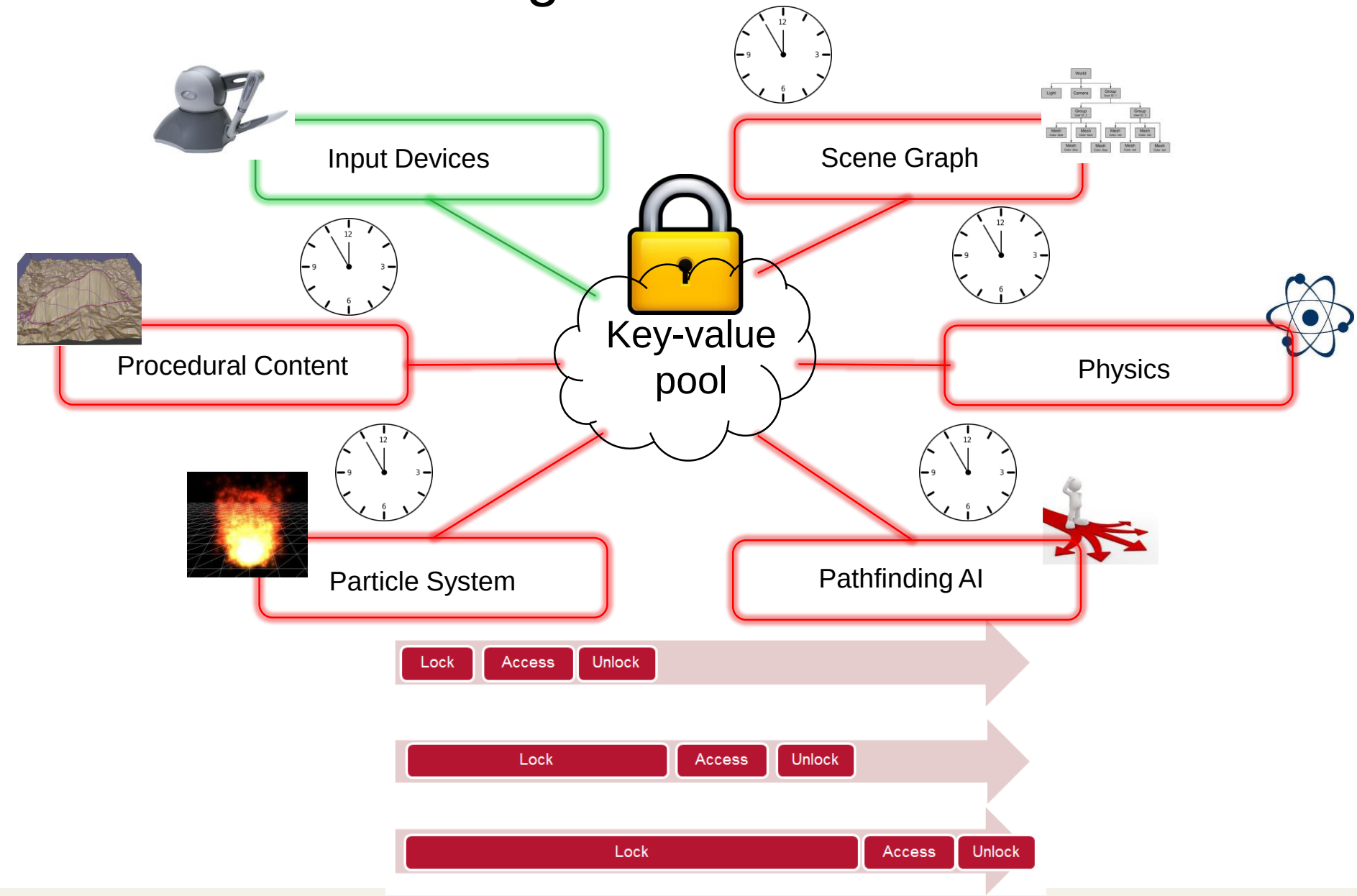


Our Novel Approach: the Key-Value Pool

- Assignment of unique key-value pair to each data packet which is exchanged between the components
- Key-value pool holds complete shared world state
  → concurrent shared data structure

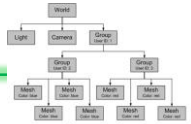


Classic Blocking Data Structures

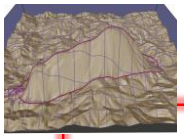


Lock-Free Non-Blocking Data Structures

Input Devices



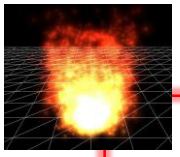
Scene Graph



Procedural Content

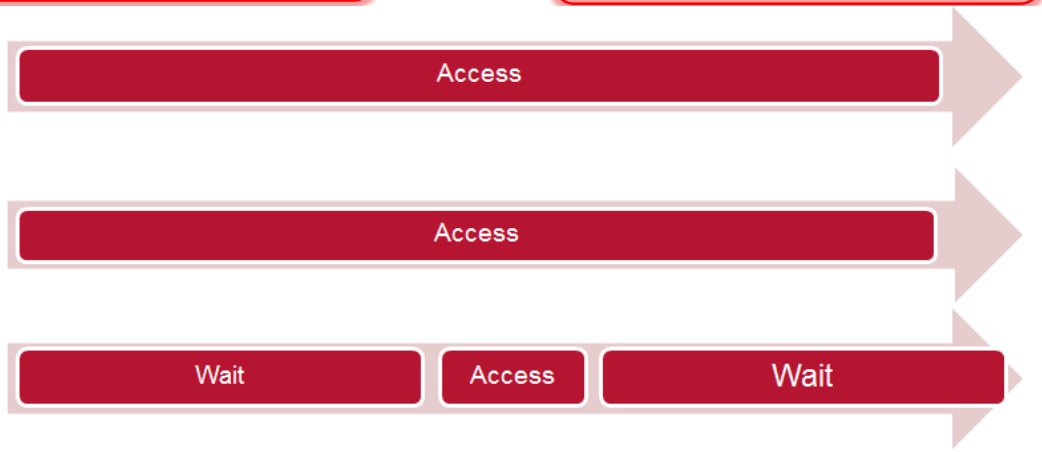
Key-value pool

Physics



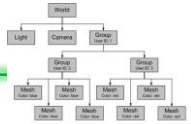
Particle System

Pathfinding AI

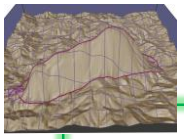


Wait-Free Non-Blocking Data Structures

Input Devices



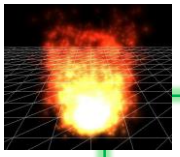
Scene Graph



Procedural Content

Key-value pool

Physics



Particle System

Pathfinding AI

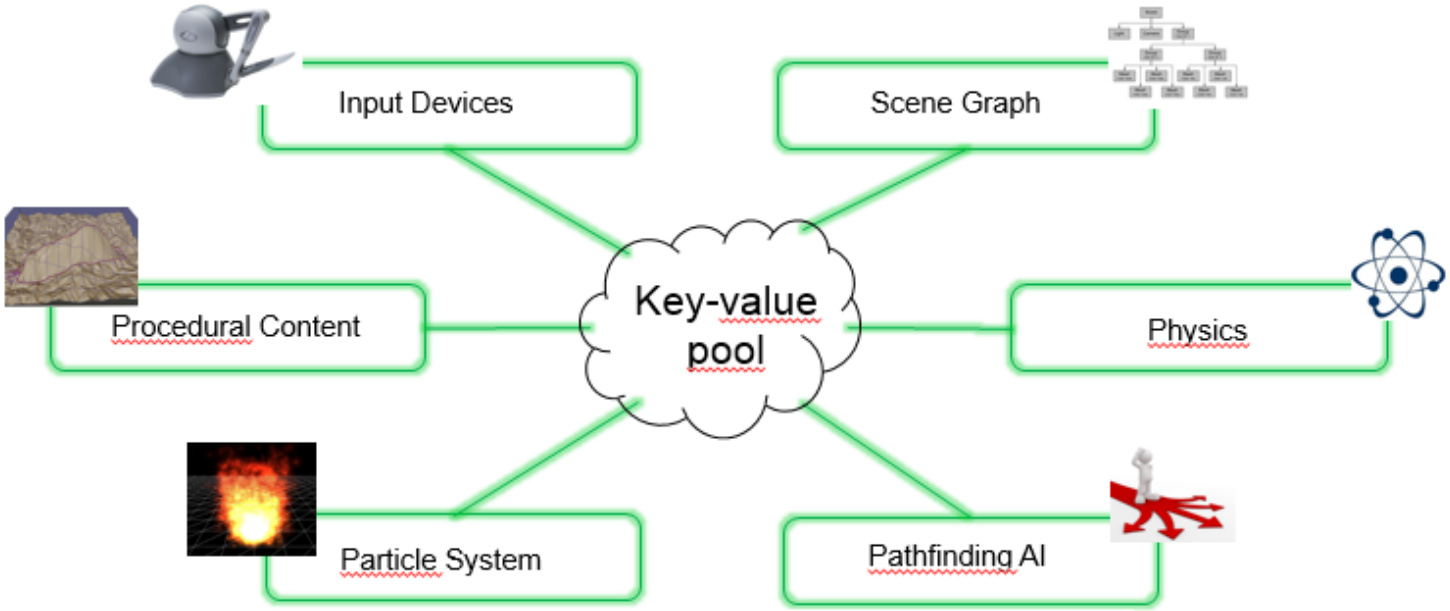
Access

Access

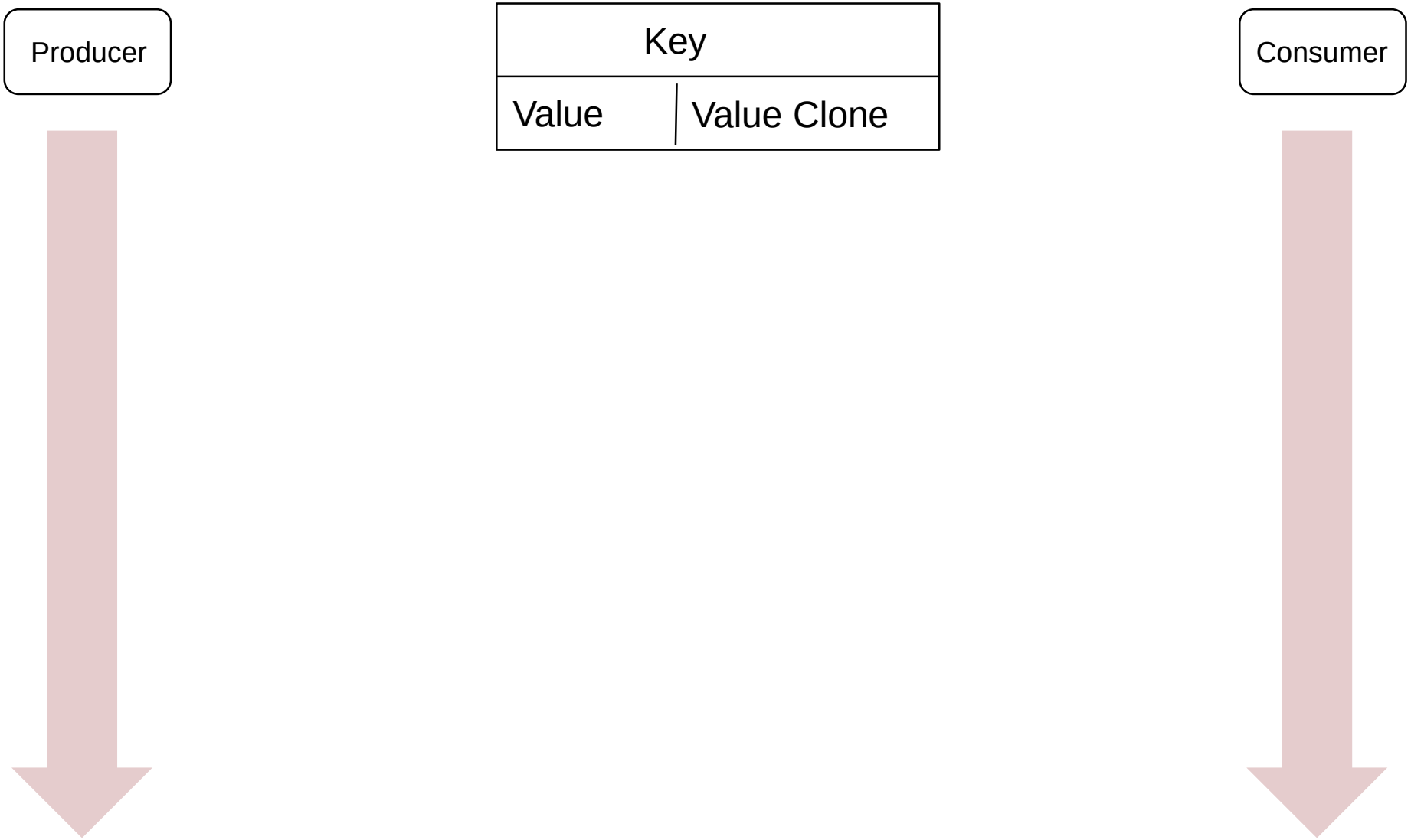
Access

Our Contribution

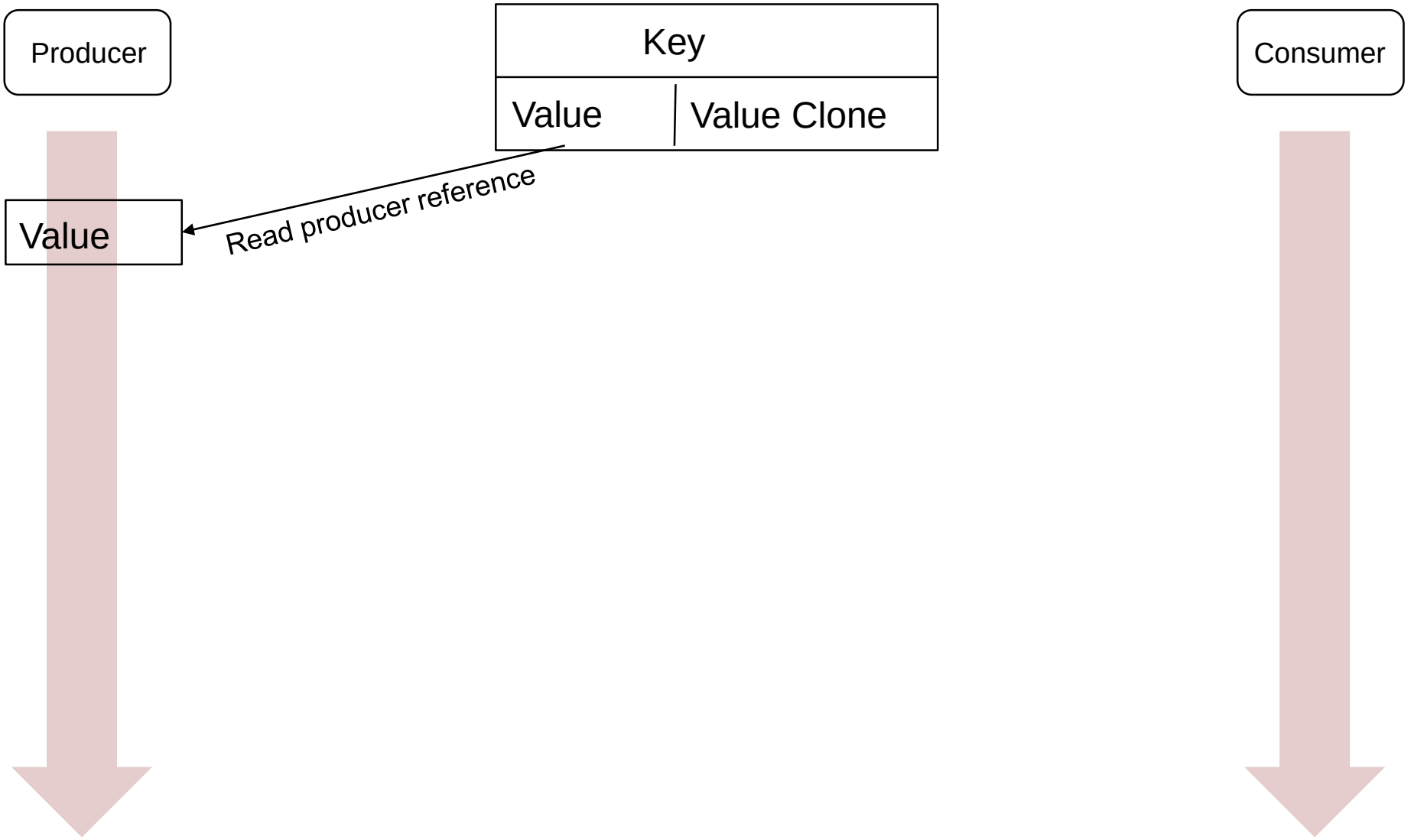
1. Novel approach to data flow in massively parallel VR system that reduces the number of interfaces from $O(n^2)$ to $O(n)$
   - Benefits better maintainability and lower synchronisation overhead
2. Novel data structure for wait-free dictionaries
   - No deadlocks, no starvation, and high performance read/write access



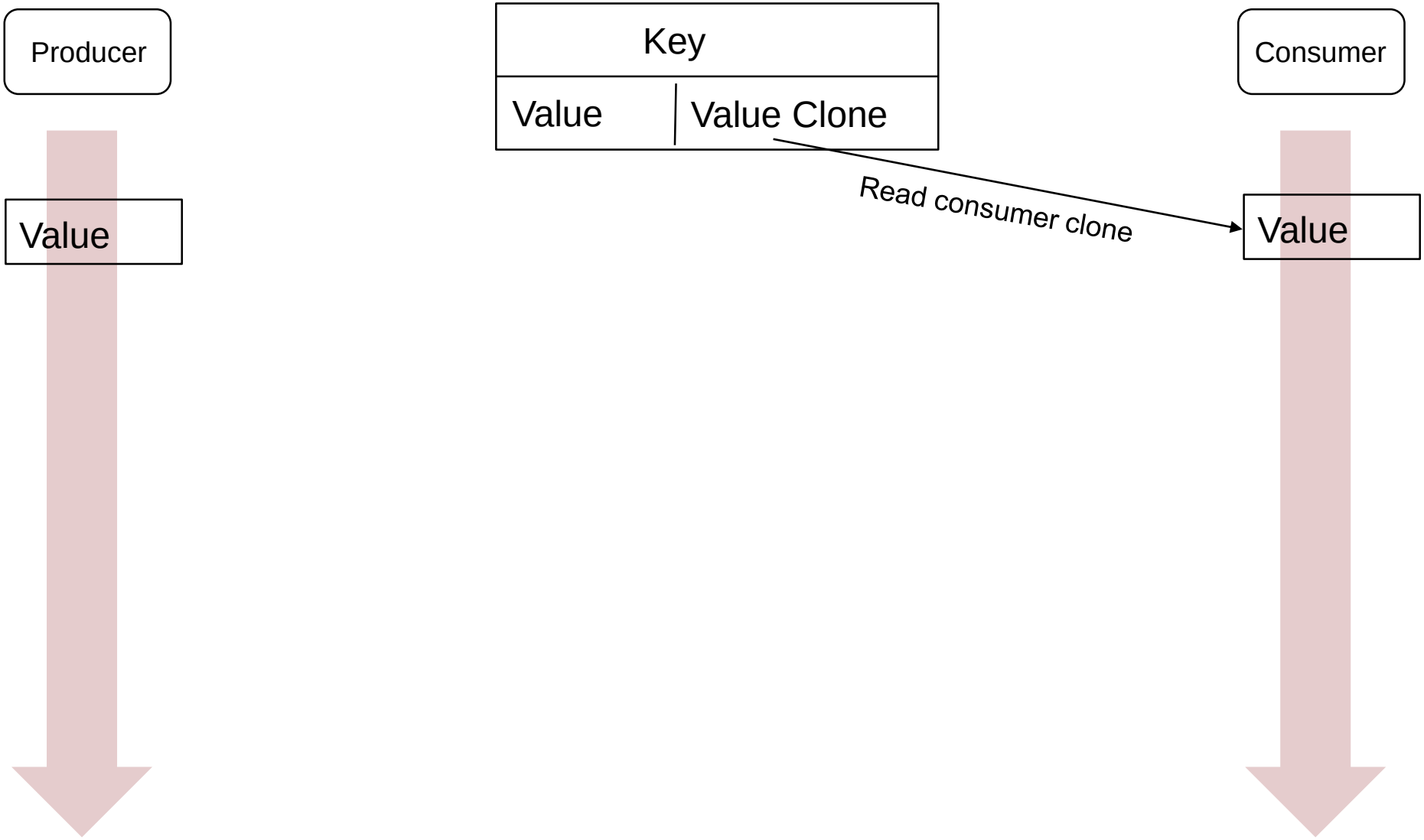
Concept Example



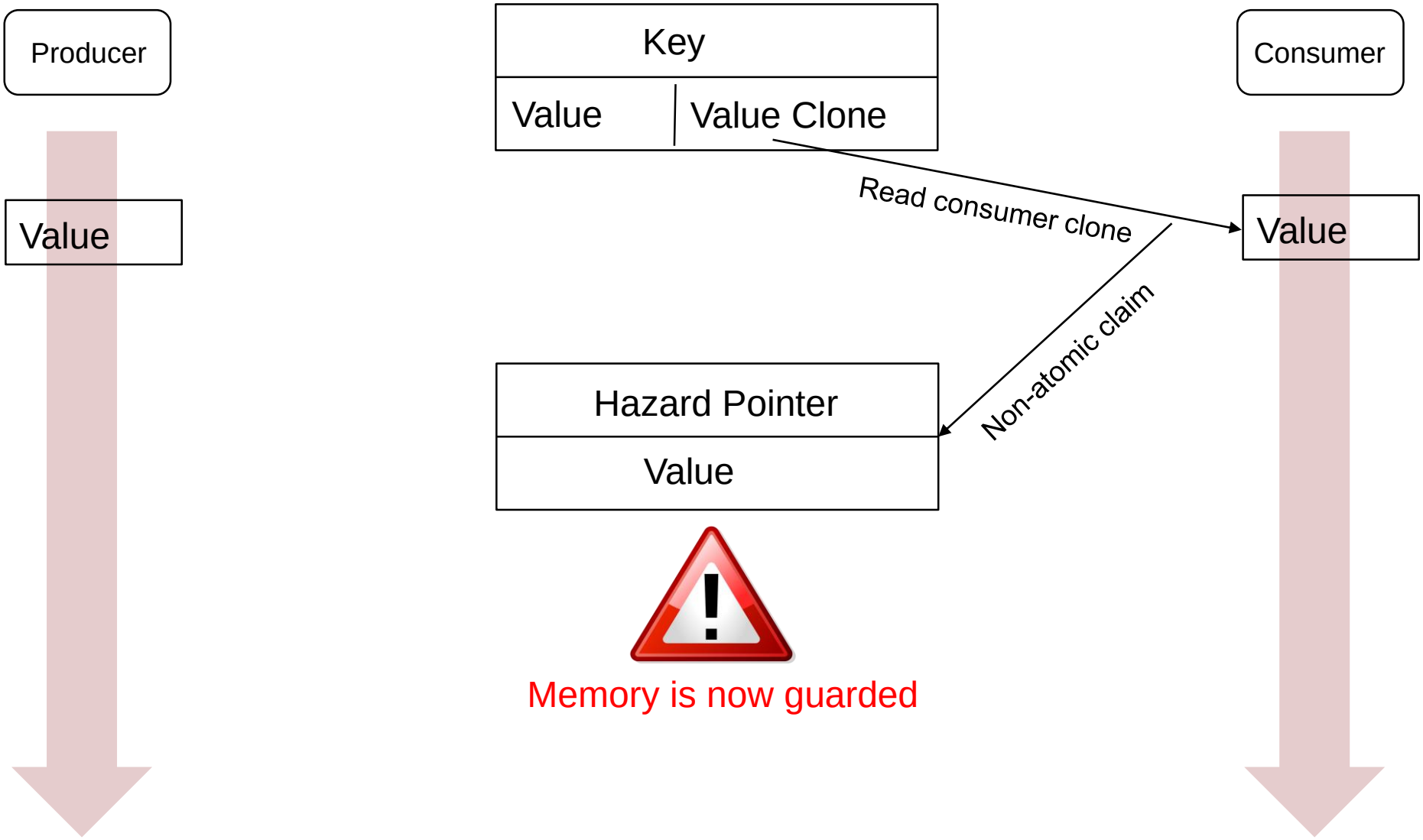
Concept Example



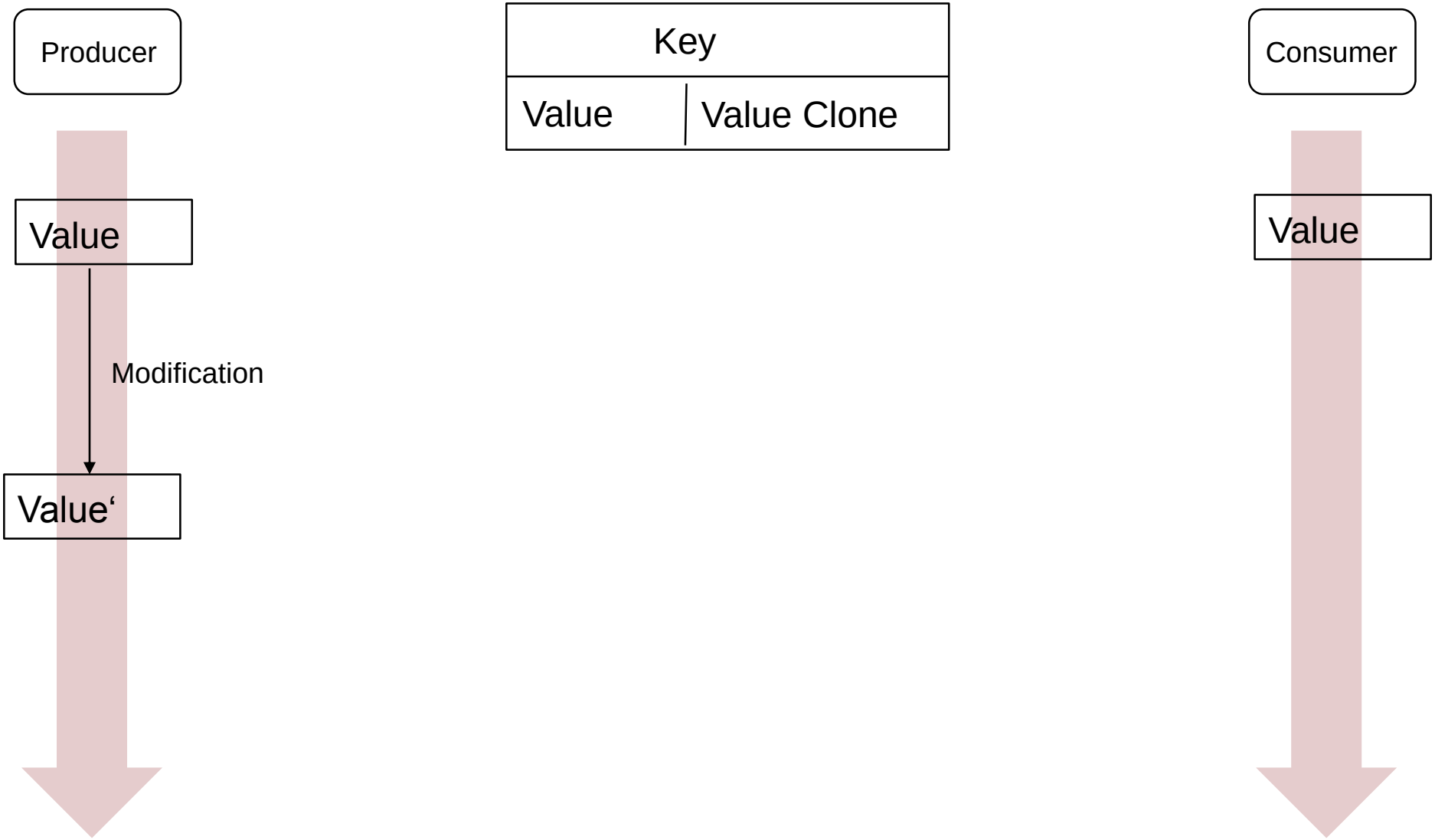
Concept Example



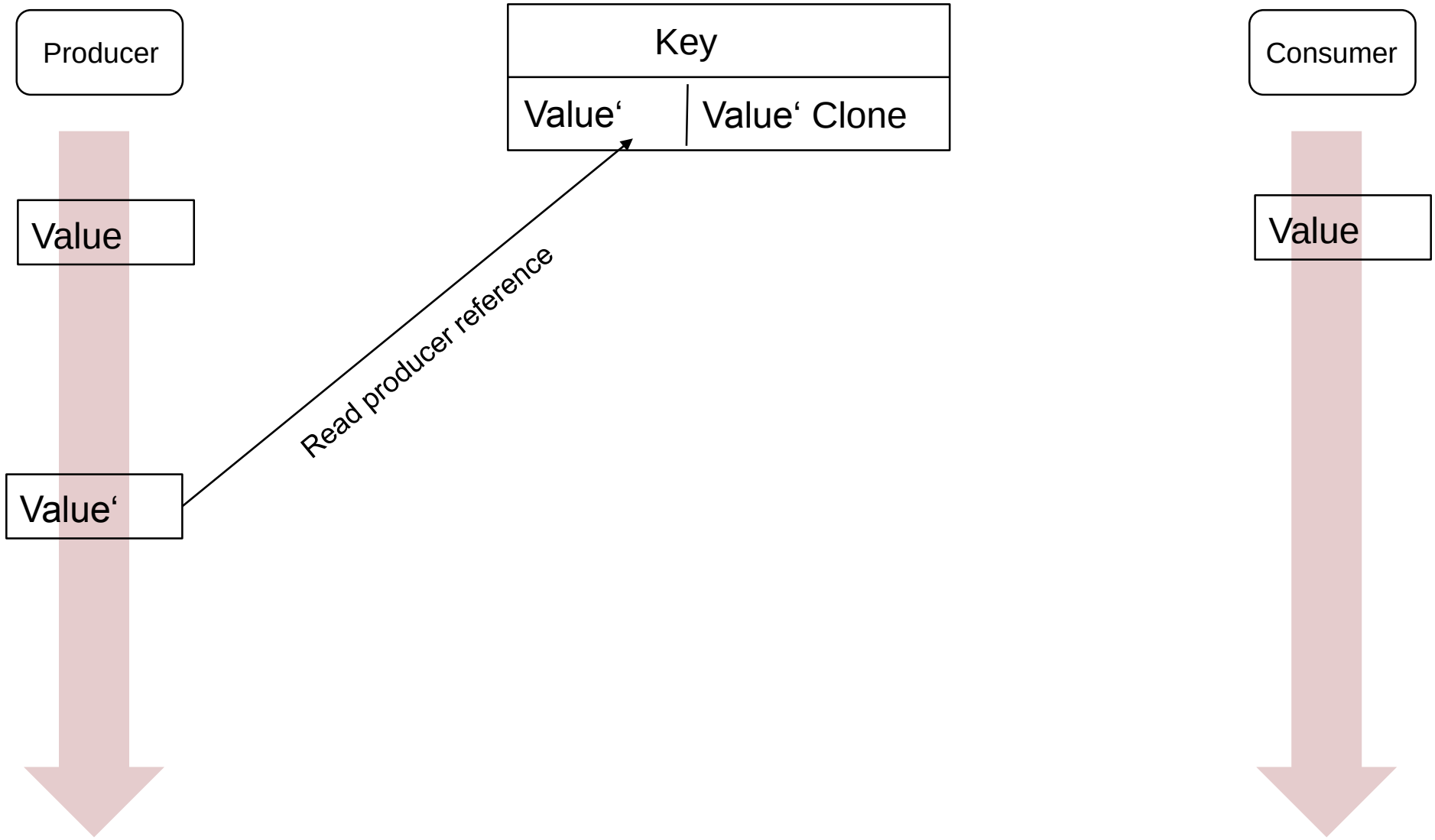
Concept Example



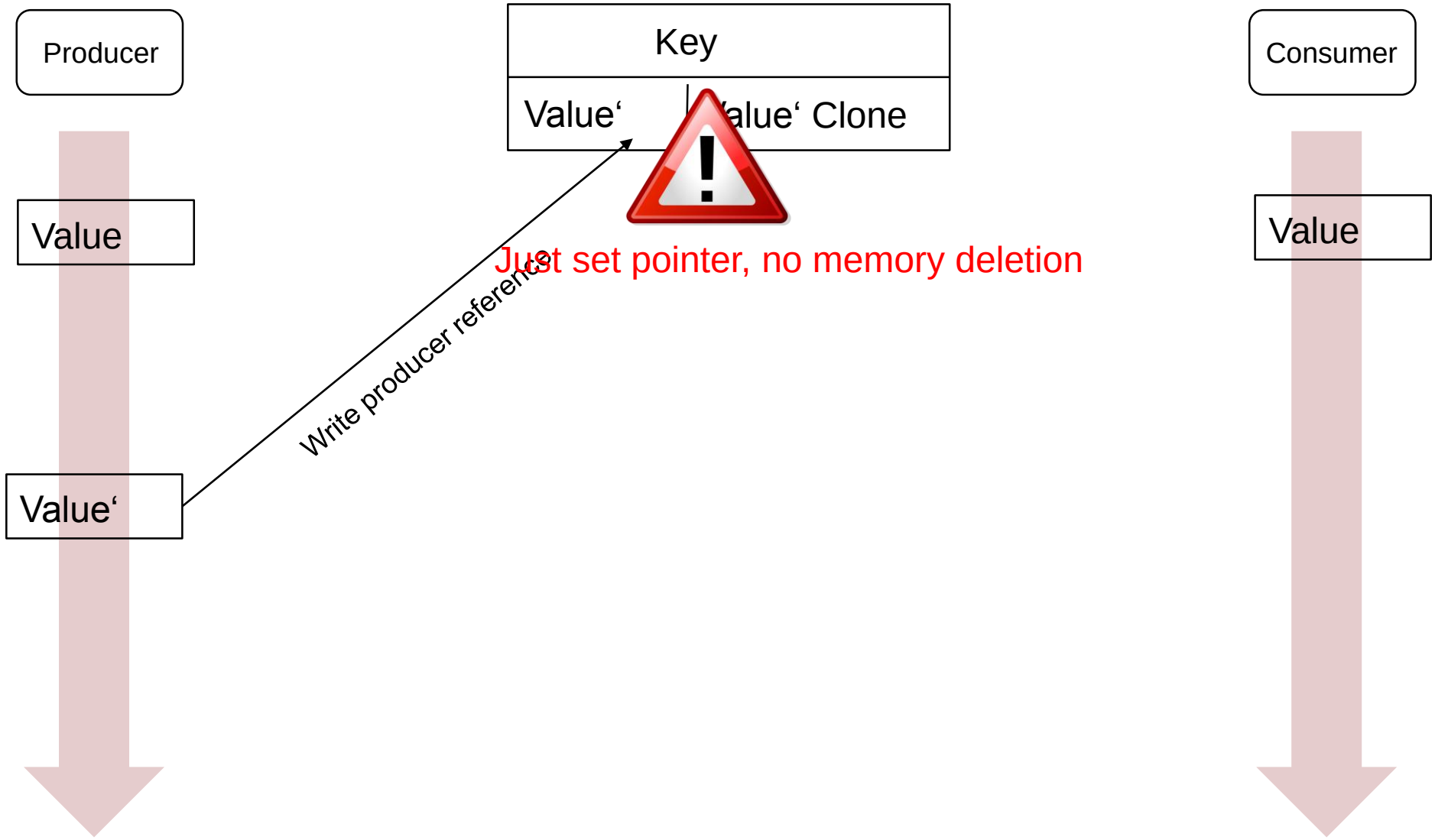
Concept Example



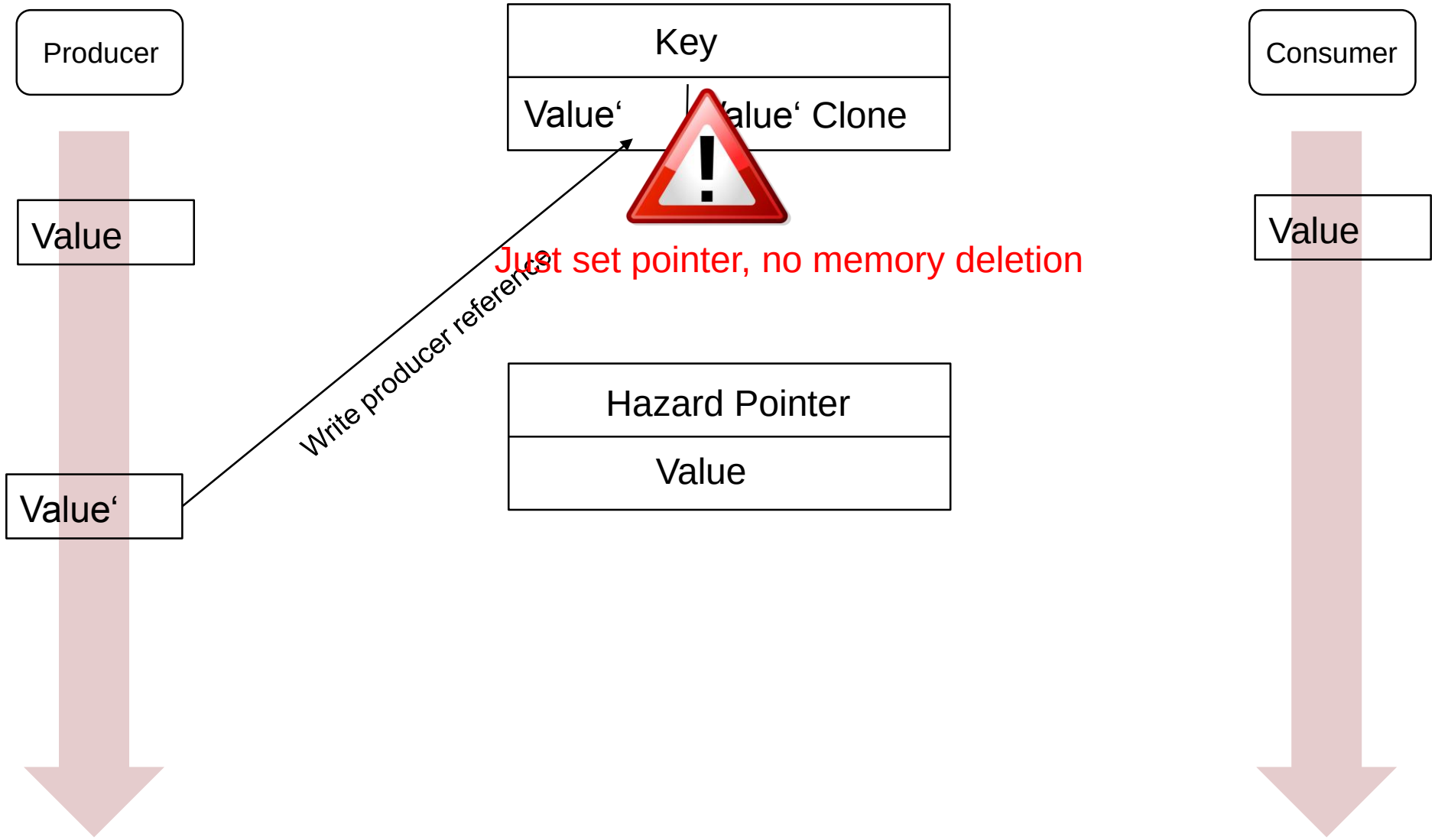
Concept Example



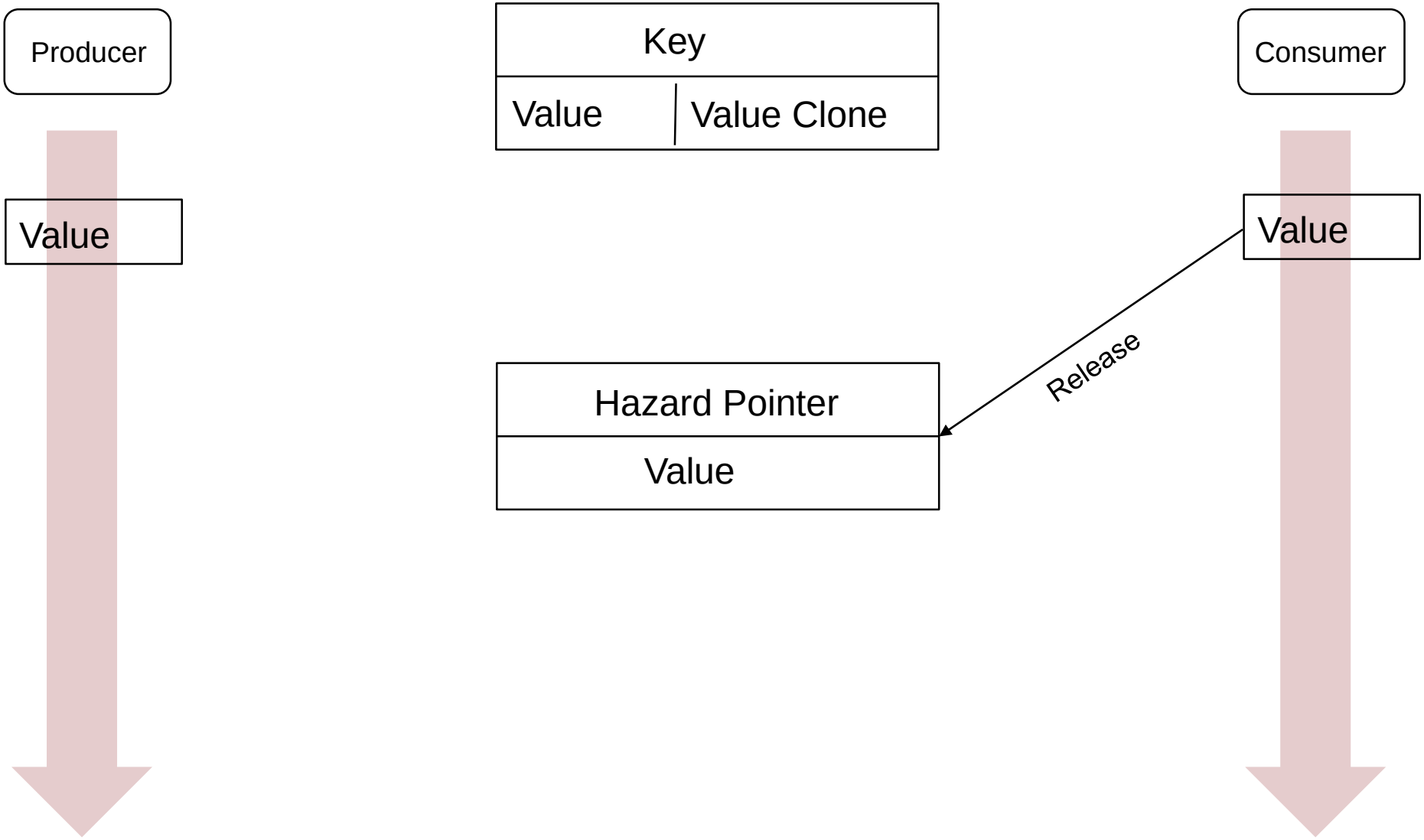
Concept Example



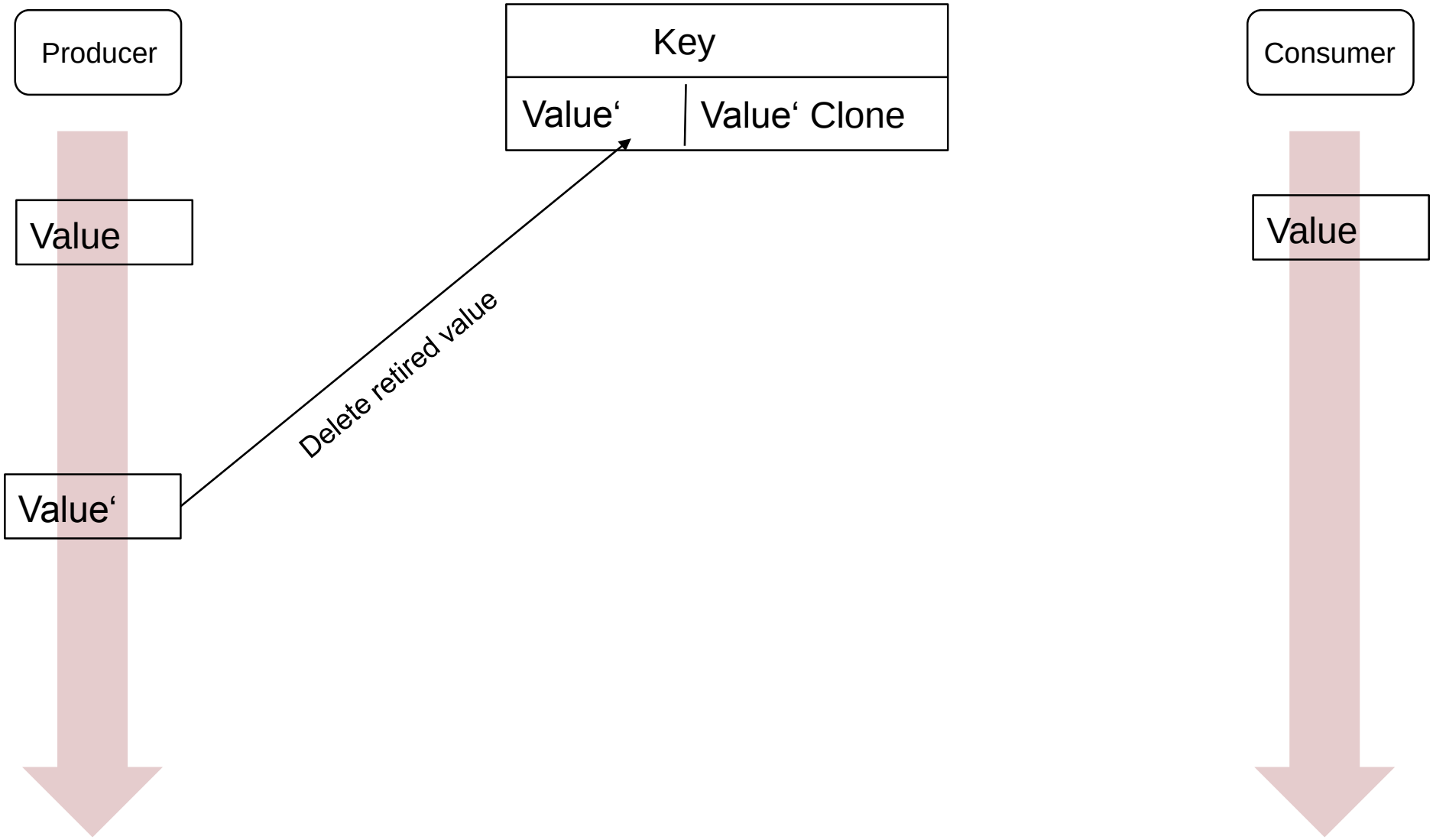
Concept Example



Concept Example



Concept Example

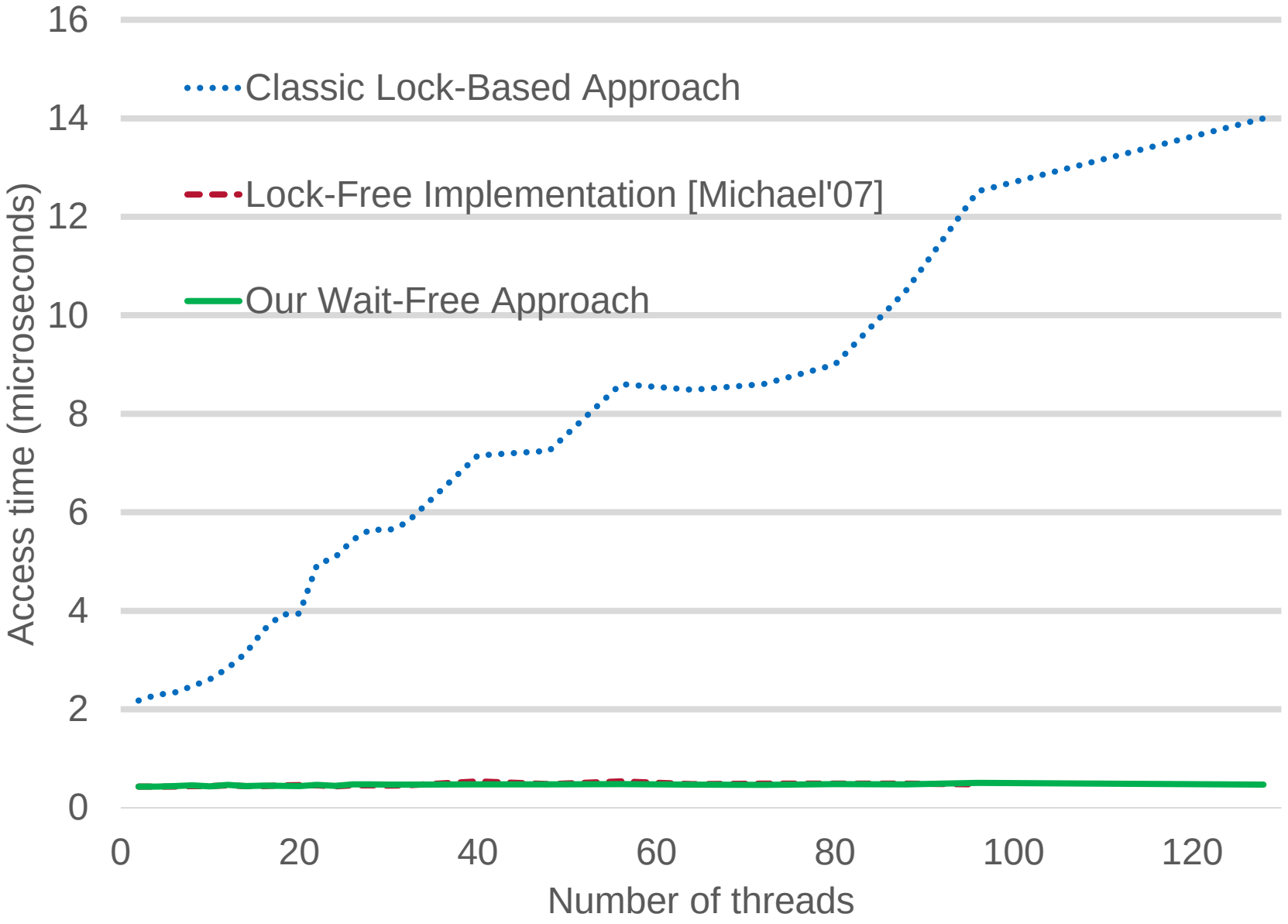


Results

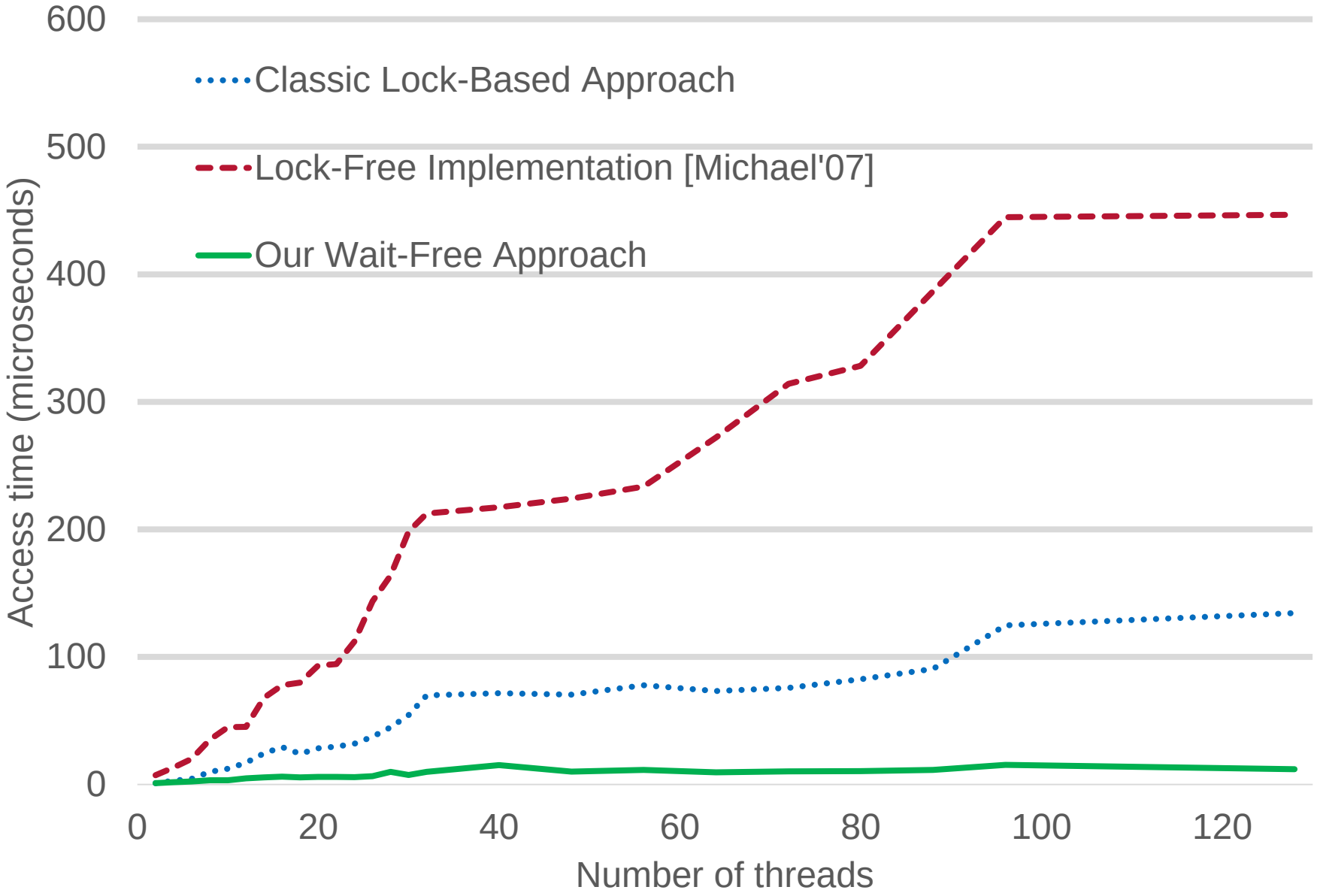
- Performance comparison with two competitors
  1. Hash map with standard locking mechanisms from the boost library
     - Read and write operations are locking
  2. Lock-free hash map based on the original Hazard Pointer scheme [Michael‘07]
     - Wait-free read and lock-free write operations

- Test configuration
  - Windows 7
  - Intel Core i7 2.4GHz, 4GB RAM
  - VC++12 Compiler with all optimizations enabled

Read Operations



Write Operations



Conclusions

1. Wait-free design pattern for data exchange (writing and reading) in massively threaded virtual environments
   - No deadlock, no starvation of producers and consumers
2. Our novel key-value pool outperforms traditional approaches by up to two orders of magnitude
3. Our novel key-value pool allows for VR systems many-to-one communication
   - Supports arbitrary non-blocking, wait-free thread cycles times even in massively threaded systems
   - Low number of interfaces leads to highly maintainable systems

Thank you for your attention

Questions?

Patrick Lange, Rene Weller, Gabriel Zachmann
{lange,weller,zach}@cs.uni-bremen.de

Funding by DLR, contract *50NA1318*