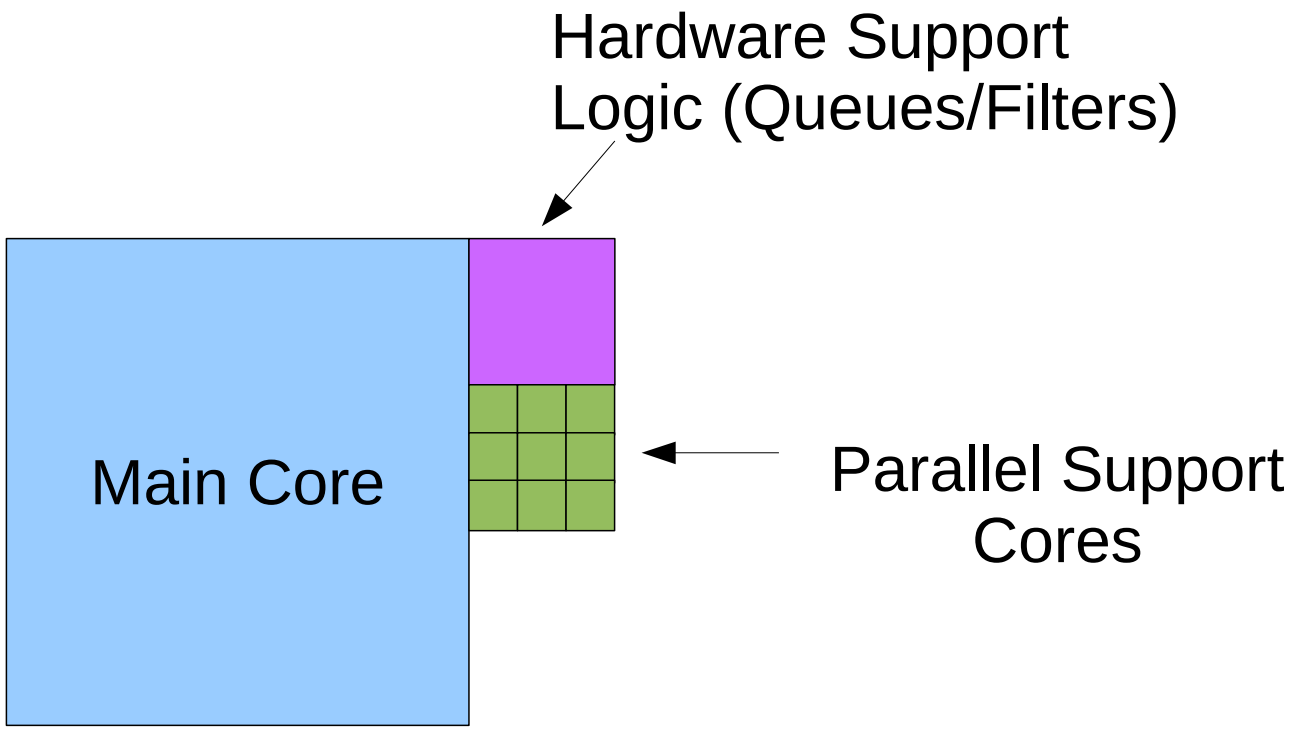
Parallel Support Cores for Performance and Reliability



Sam Ainsworth

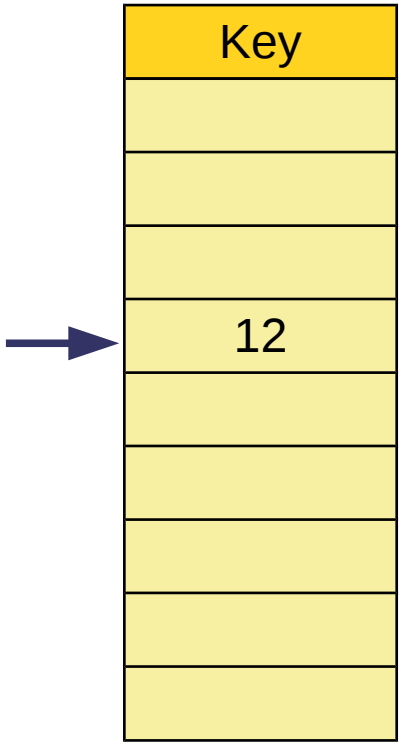
UNIVERSITY OF CAMBRIDGE
Computer Laboratory

What we'll discuss today

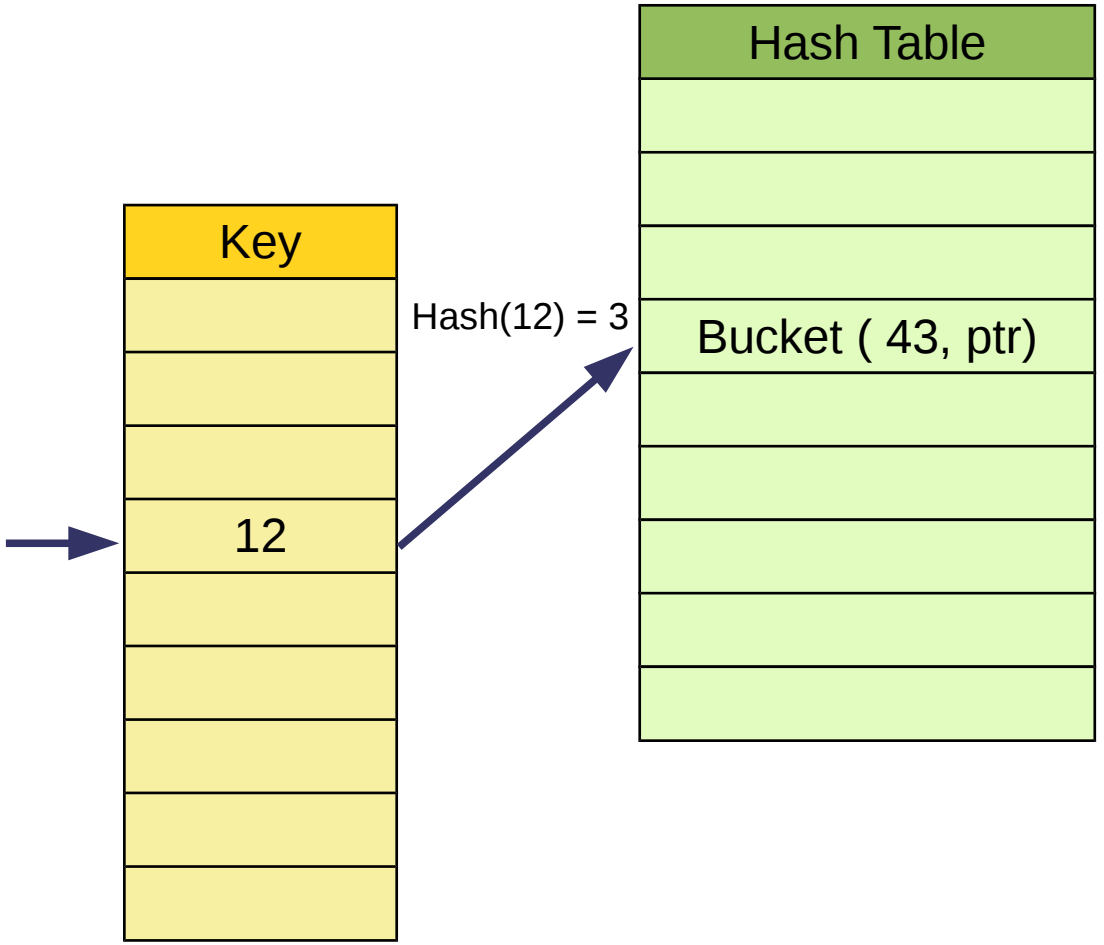
- Programmable Prefetching
- Fault Tolerance
- Security

Programmable Prefetching

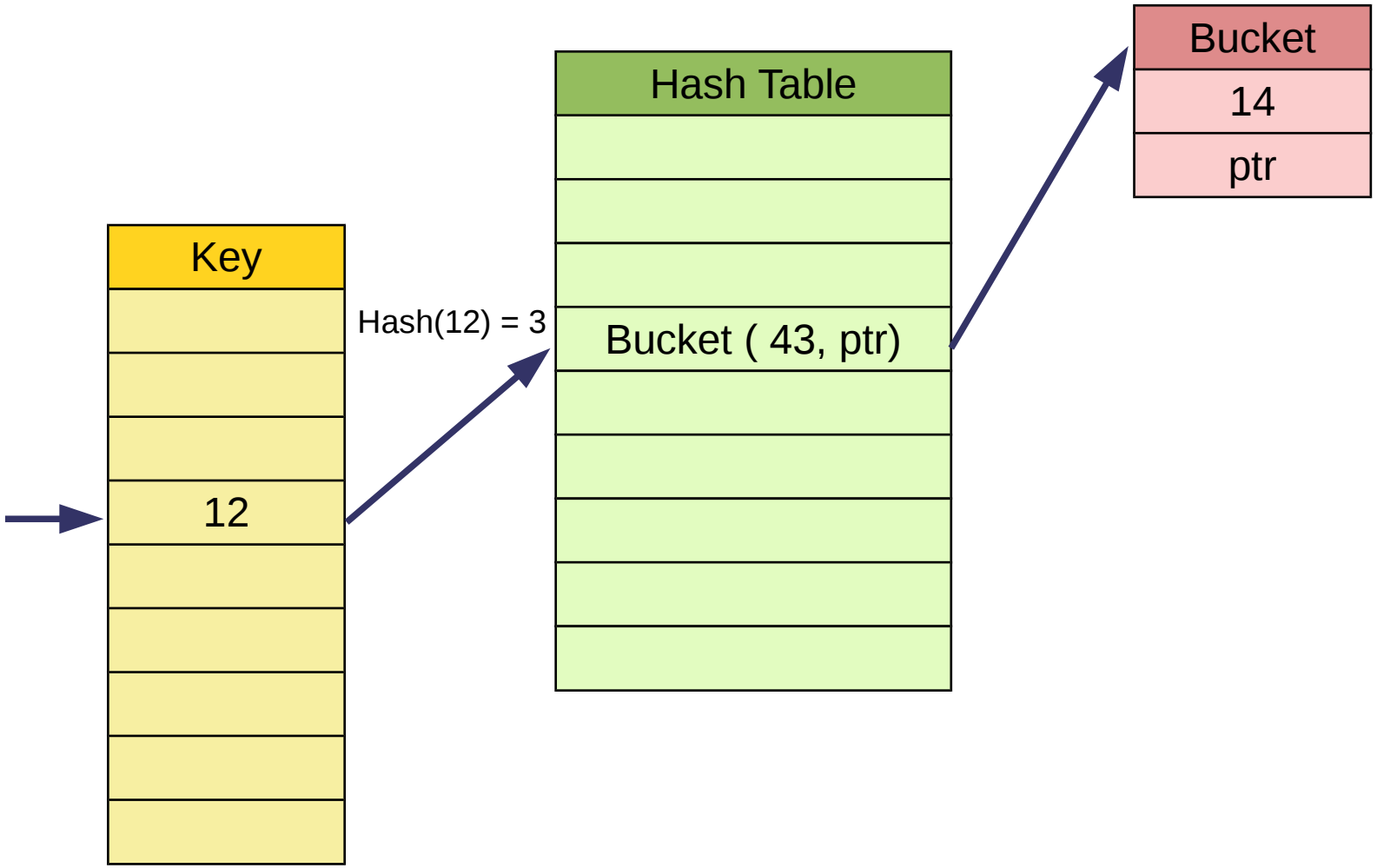
Programmable Prefetching: Example



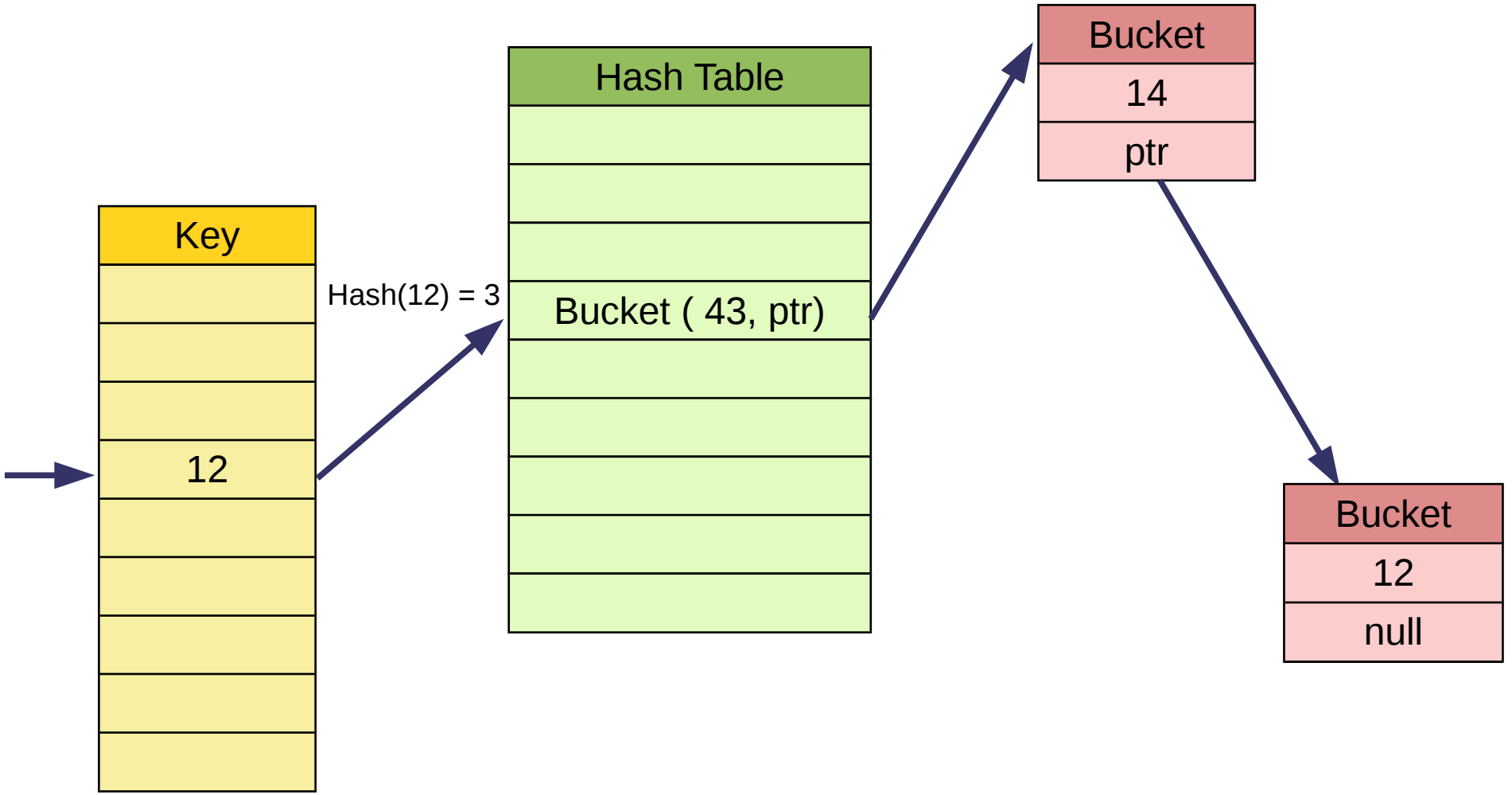
Programmable Prefetching: Example



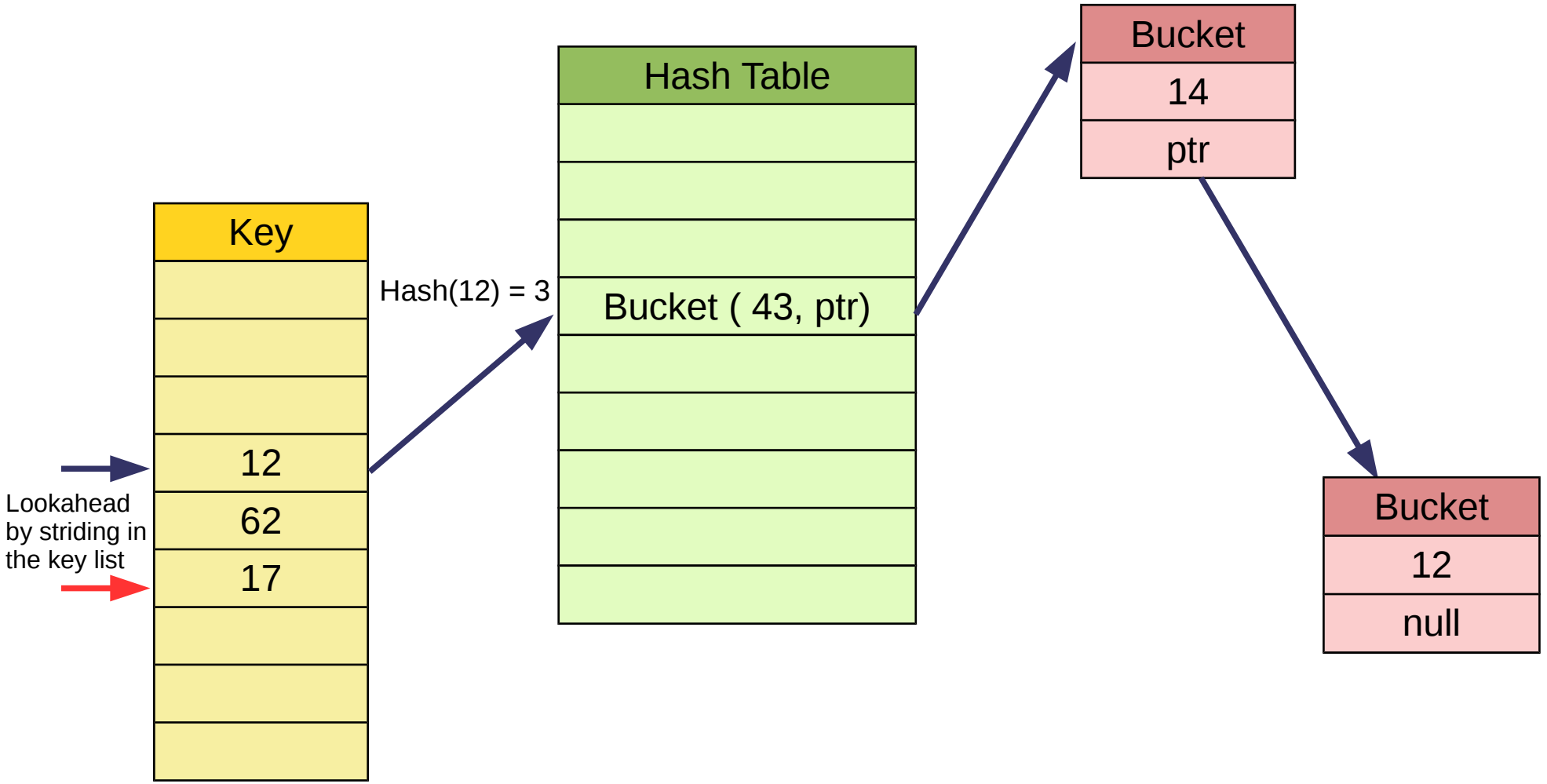
Programmable Prefetching: Example



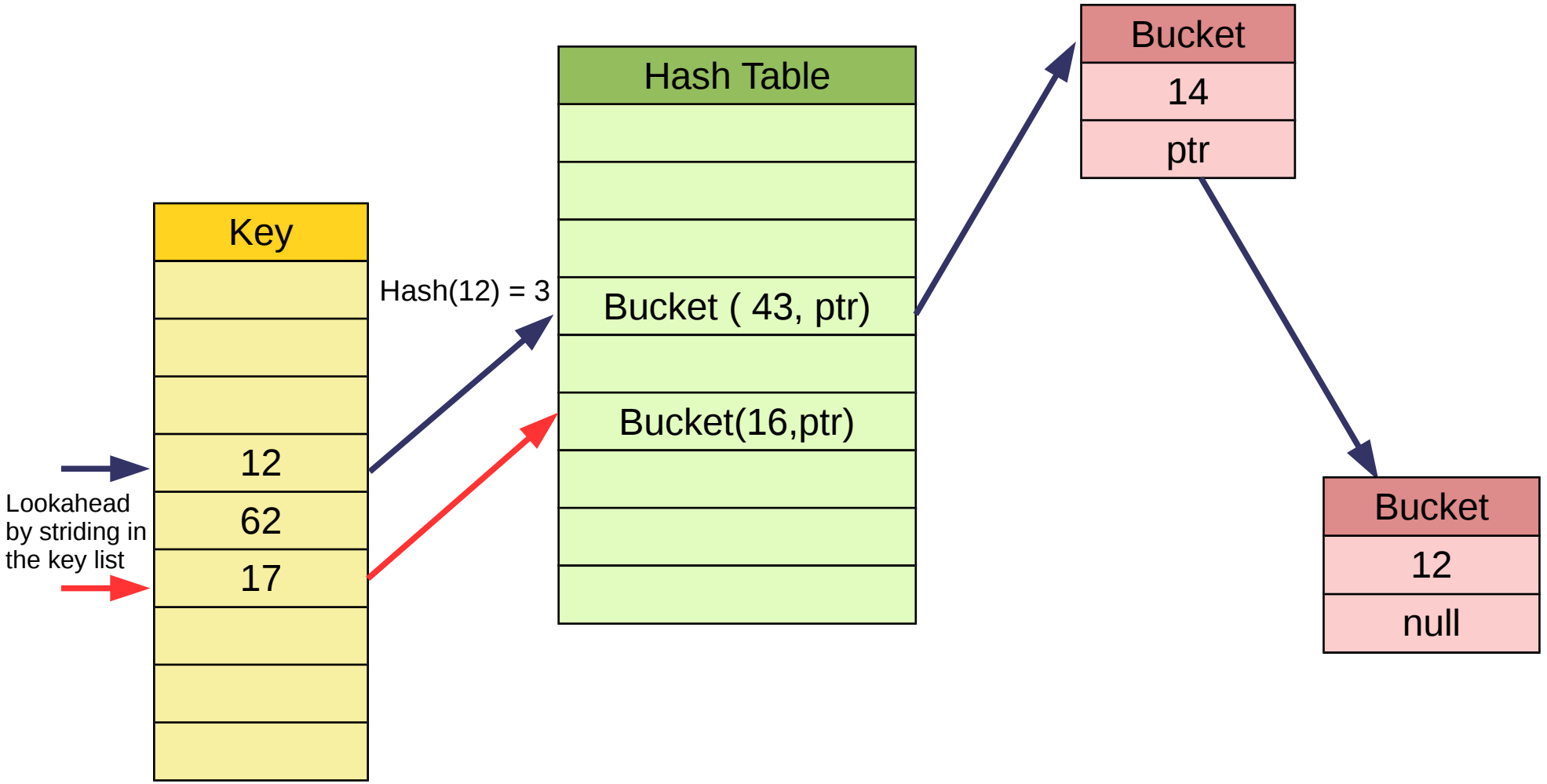
Programmable Prefetching: Example



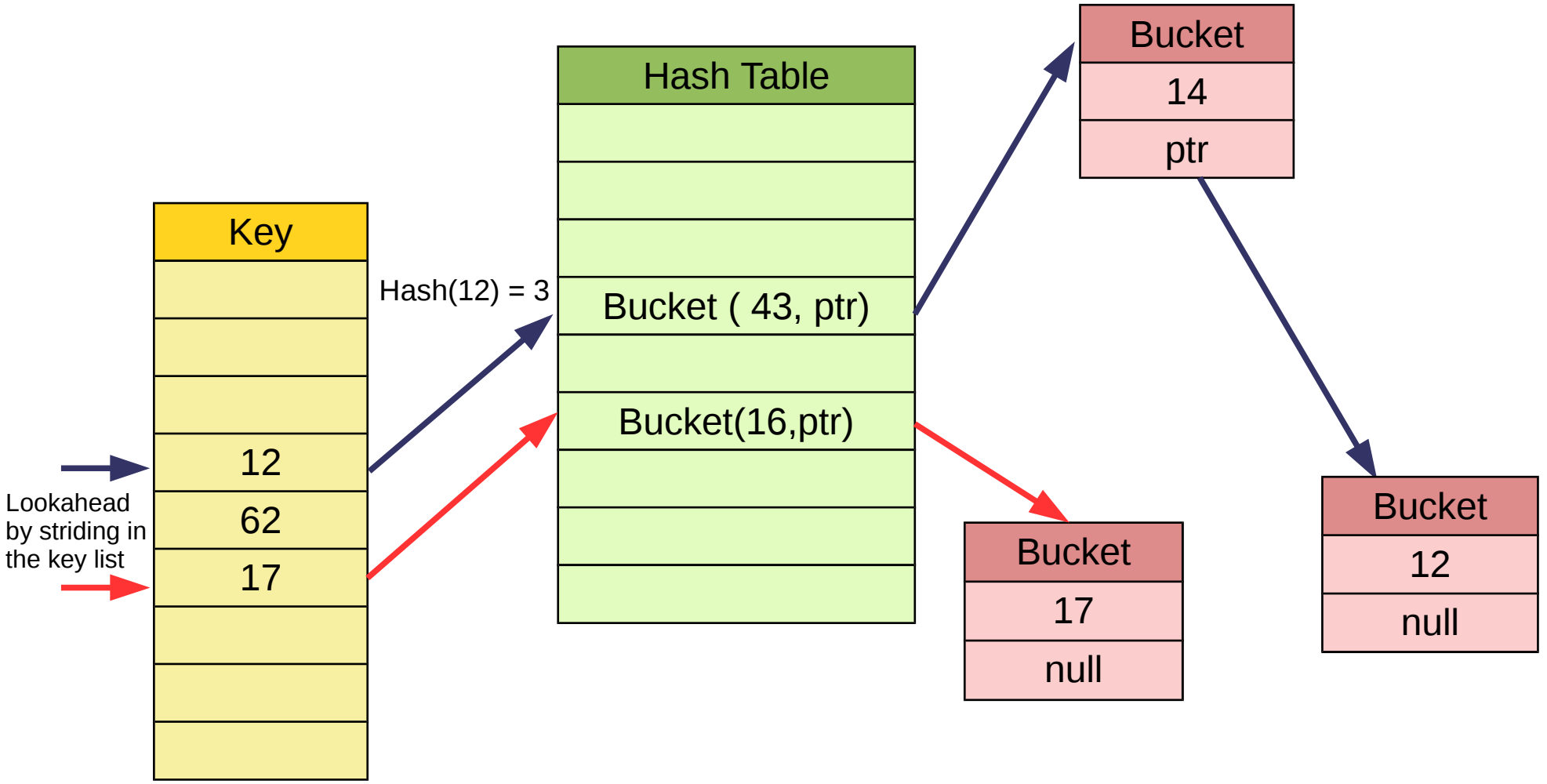
Programmable Prefetching: Example



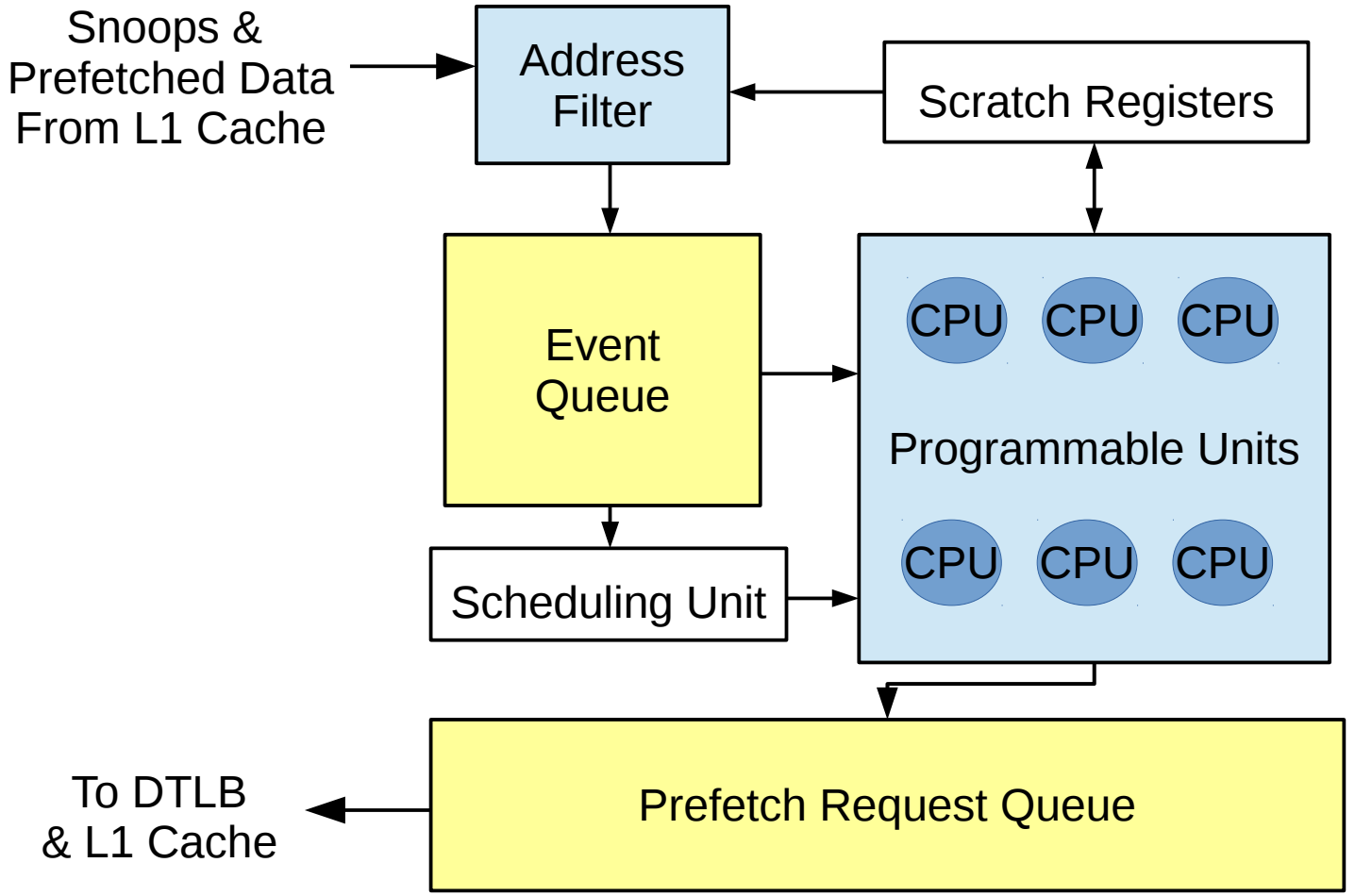
Programmable Prefetching: Example



Programmable Prefetching: Example



Programmable Prefetching: Architecture



Programmable Prefetching: Programming Model

```
int64_t acc = 0;
for (x=0; x<N; x++) {
  acc += C[B[A[x]]];
}
return acc;
```

Main Program

Programmable Prefetching: Programming Model

```
int64_t acc = 0;
for (x=0; x<N; x++) {
  acc += C[B[A[x]]];
}
return acc;
```

Main Program

```
void on_A_load() {
  <calc addr at offset in A>
  prefetch
}
```

Event-Kernel Code

Programmable Prefetching: Programming Model

```
int64_t acc = 0;
for (x=0; x<N; x++) {
  acc += C[B[A[x]]];
}
return acc;
```

Main Program

```
void on_A_load() {
  <calc addr at offset in A>
  prefetch
}
```

```
void on_A_prefetch() {
  <calculate addr in B>
  prefetch
}
```

Event-Kernel Code

Programmable Prefetching: Programming Model

```
int64_t acc = 0;
for (x=0; x<N; x++) {
  acc += C[B[A[x]]];
}
return acc;
```

Main Program

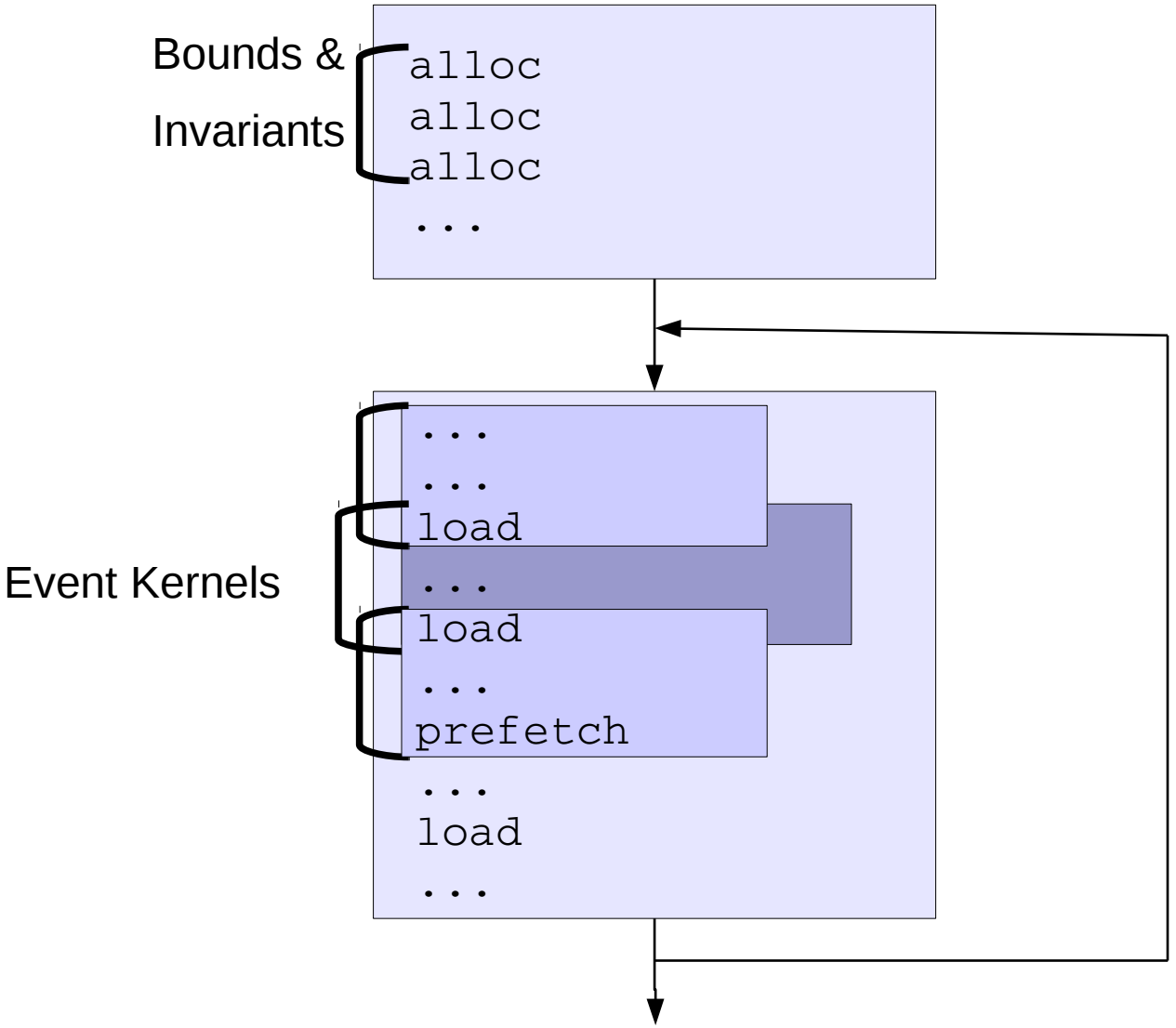
```
void on_A_load() {
  <calc addr at offset in A>
  prefetch
}
```

```
void on_A_prefetch() {
  <calculate addr in B>
  prefetch
}
```

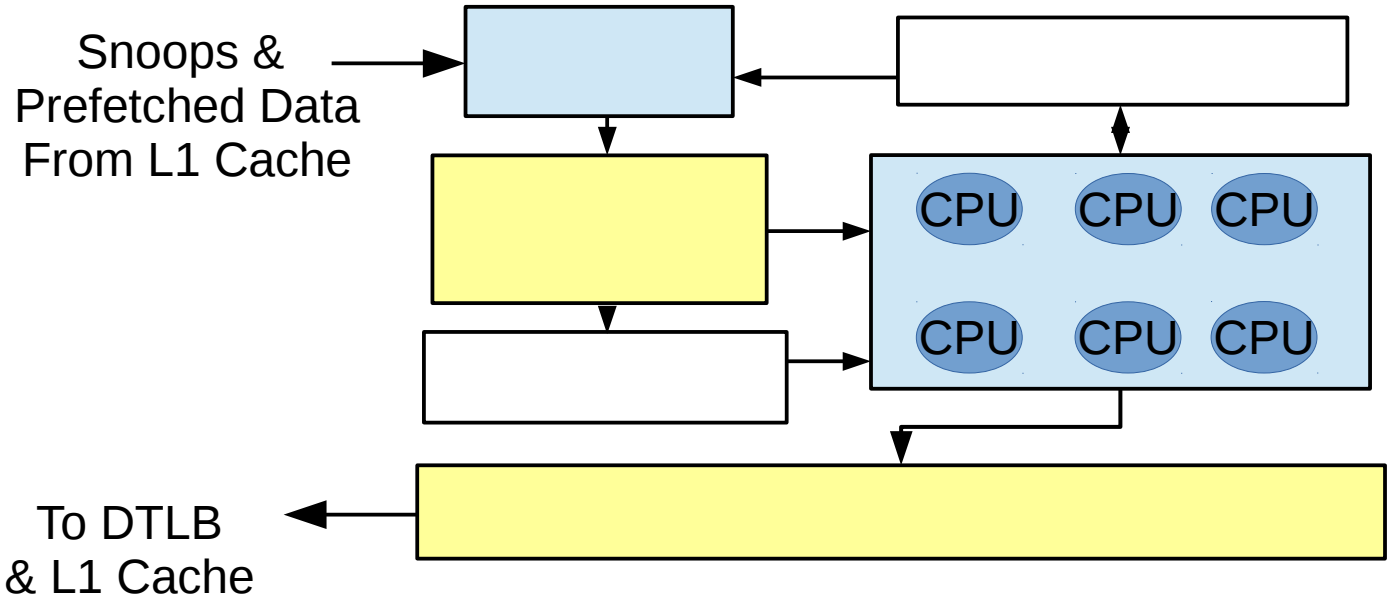
```
void on_B_prefetch() {
  <calculate addr in C>
  prefetch
}
```

Event-Kernel Code

Programmable Prefetching: Compiler Automation



Programmable Prefetching

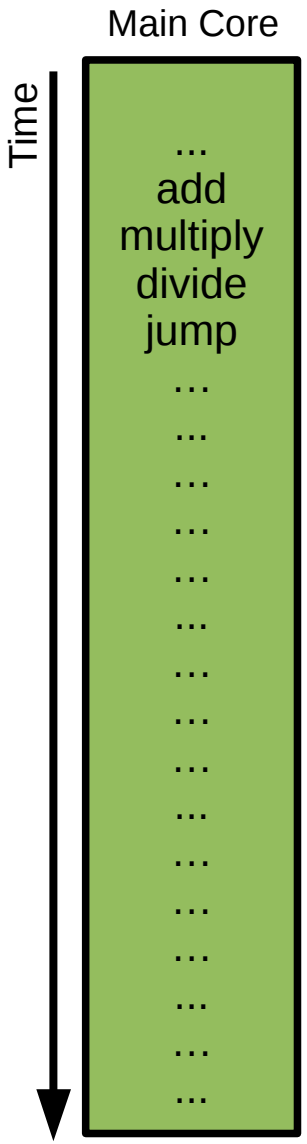


For more information, see our paper:

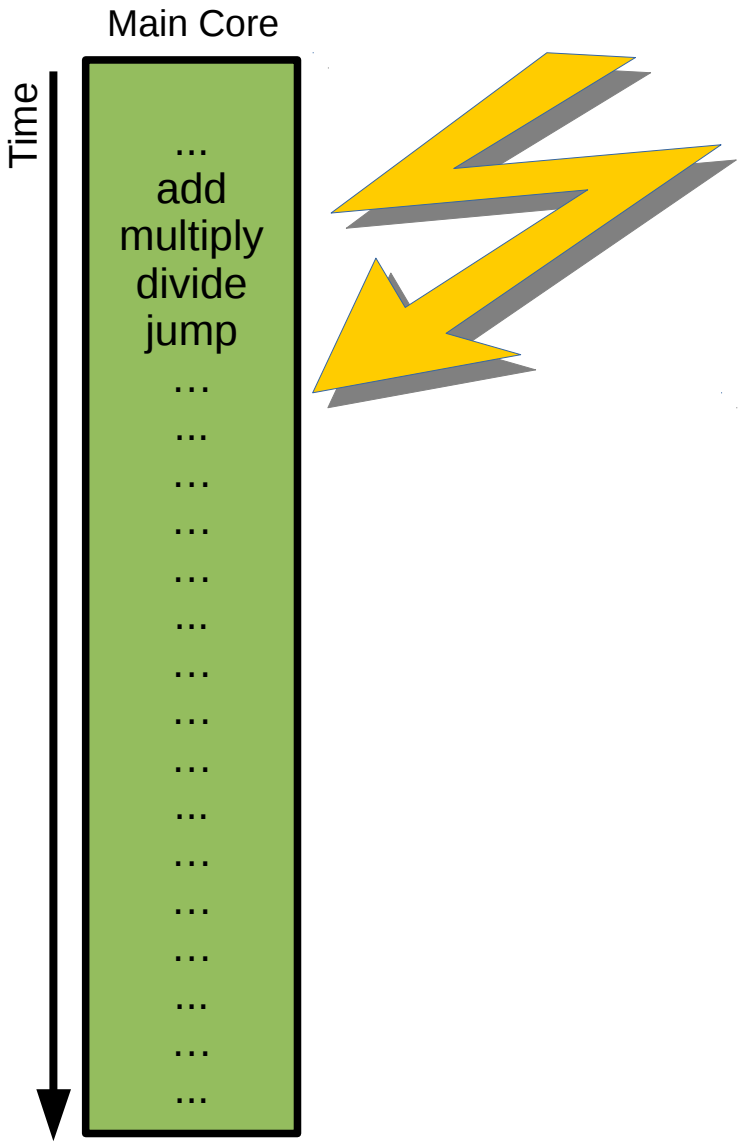
An Event-Triggered Programmable Prefetcher for Irregular Workloads, Sam Ainsworth and Timothy M. Jones, ASPLOS 2018

Processor Error Detection

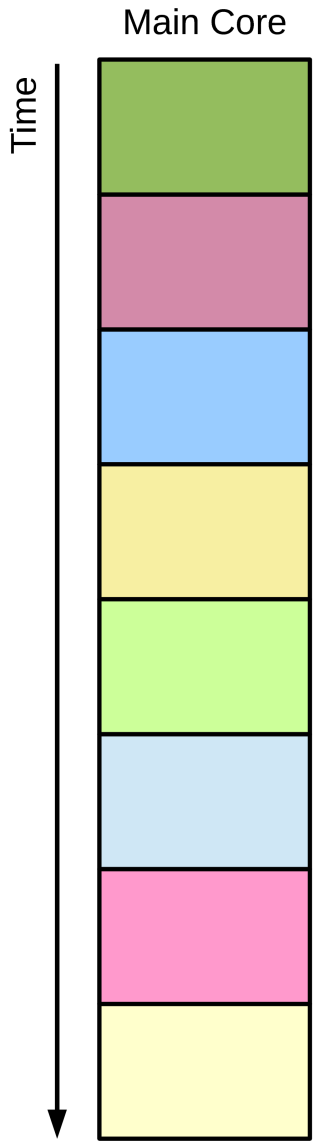
Error Detection



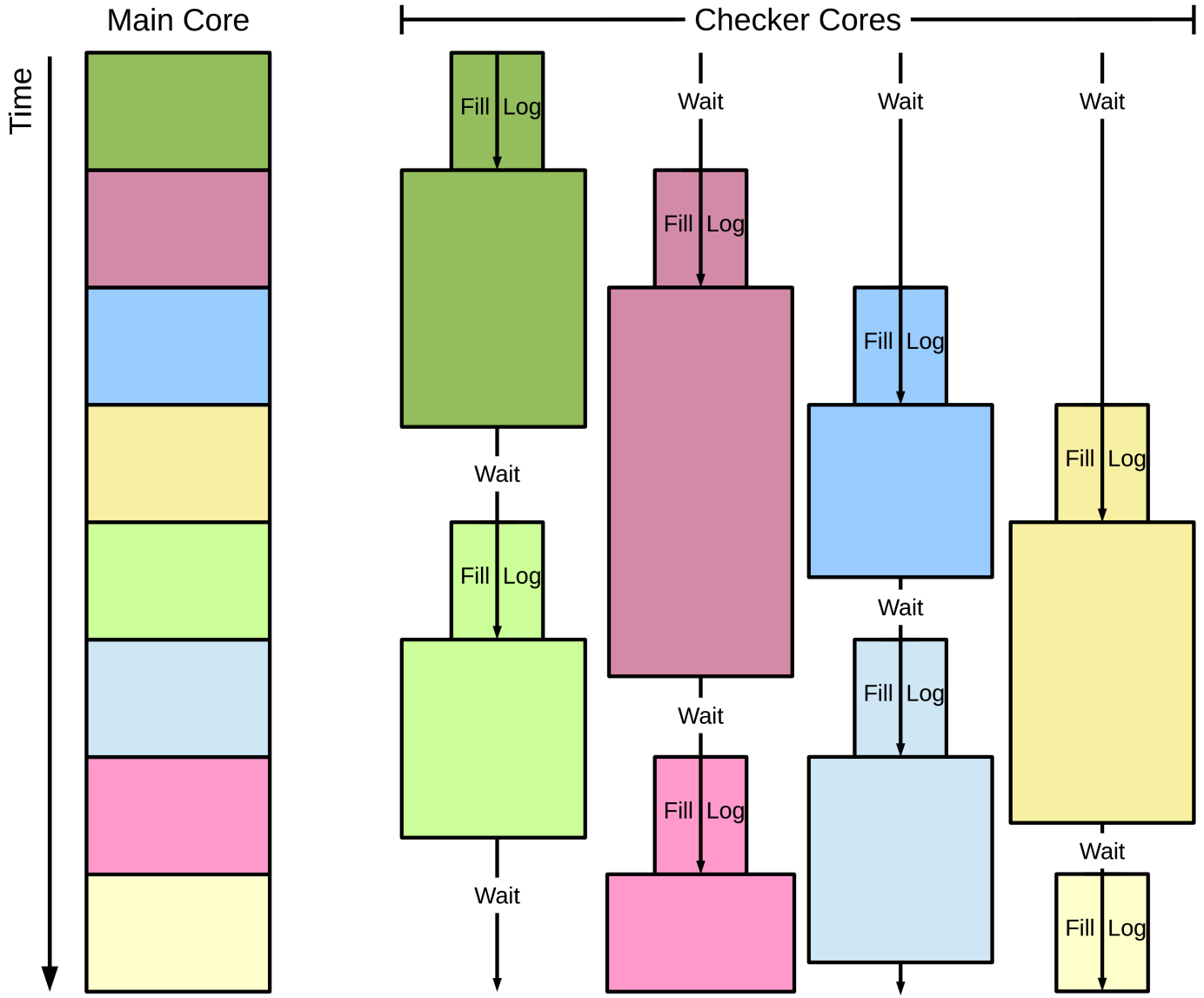
Error Detection



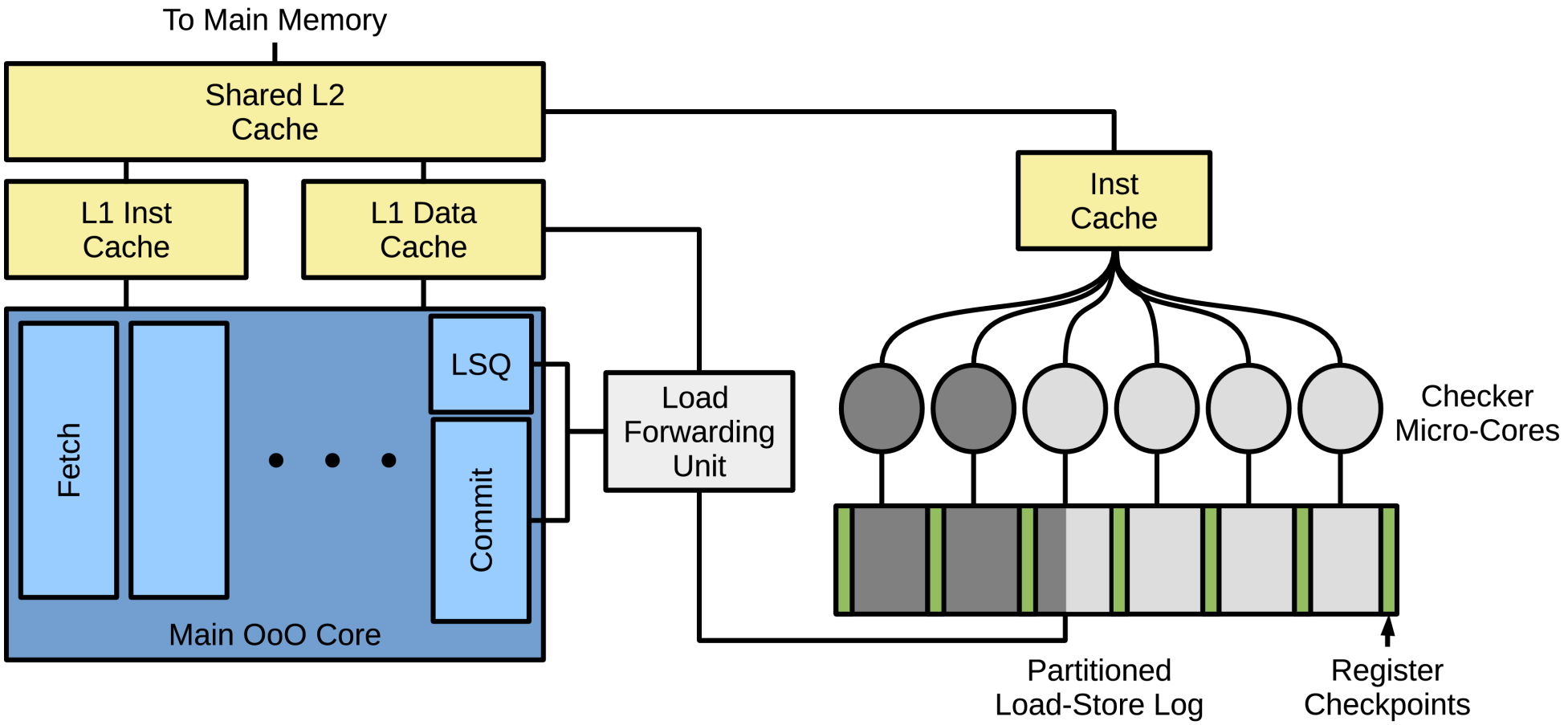
Error Detection is Parallel



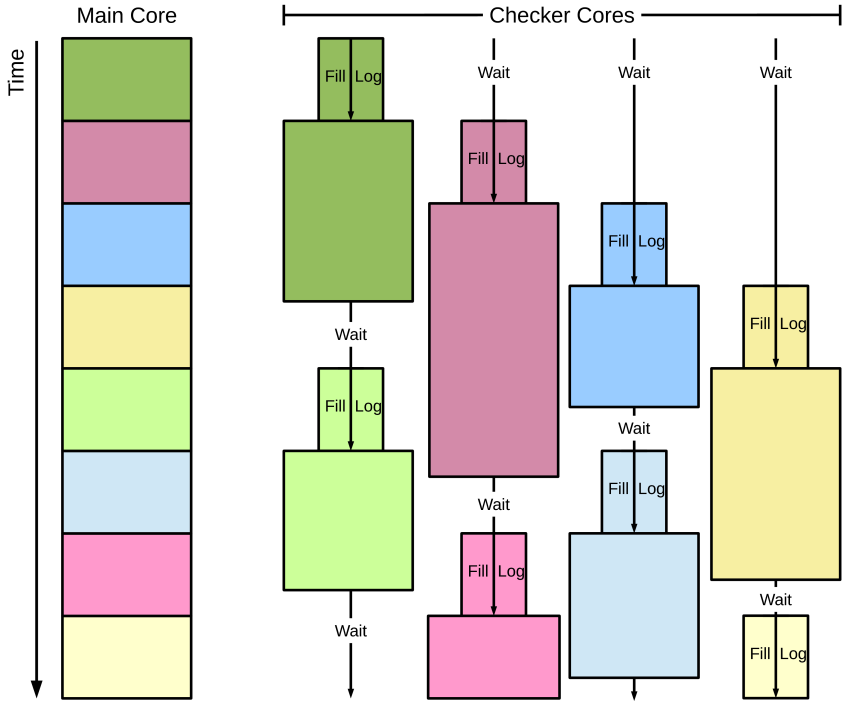
Error Detection is Parallel



Error Detection: Architecture



Error Detection

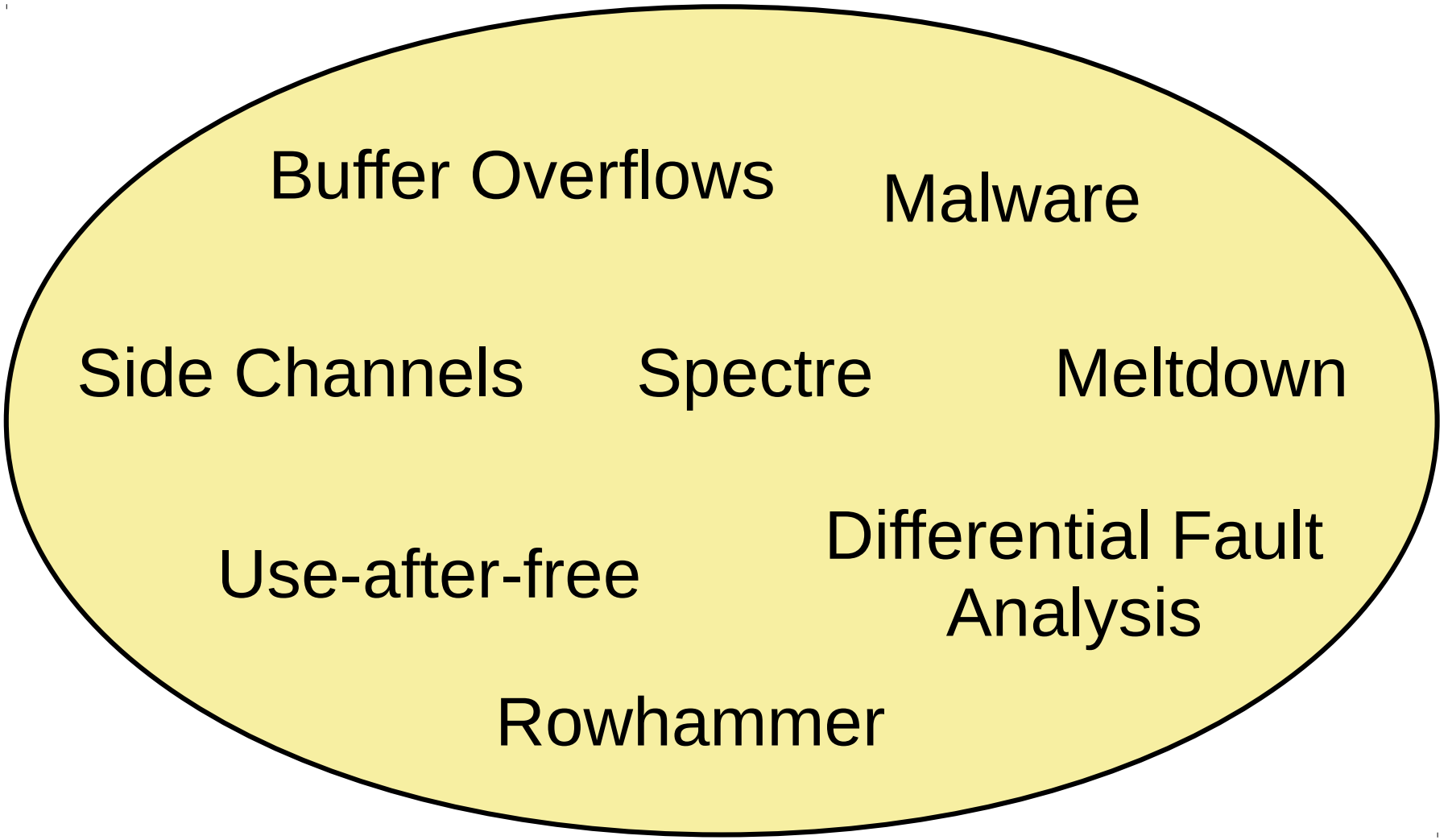


For more information, see our paper:

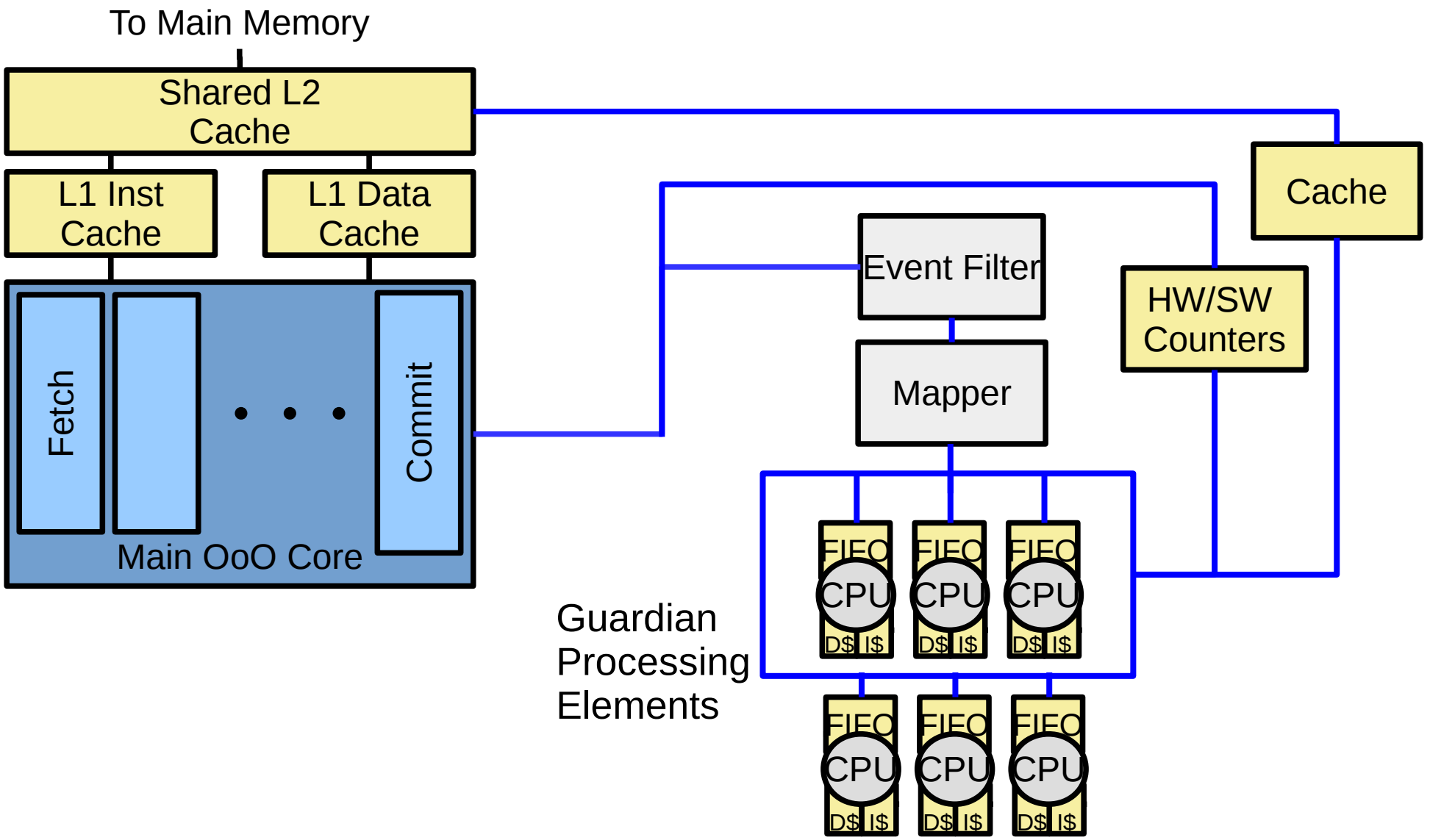
Parallel Error Detection Using Heterogeneous Cores, Sam Ainsworth and Timothy M. Jones, DSN 2018

Programmable Hardware Security

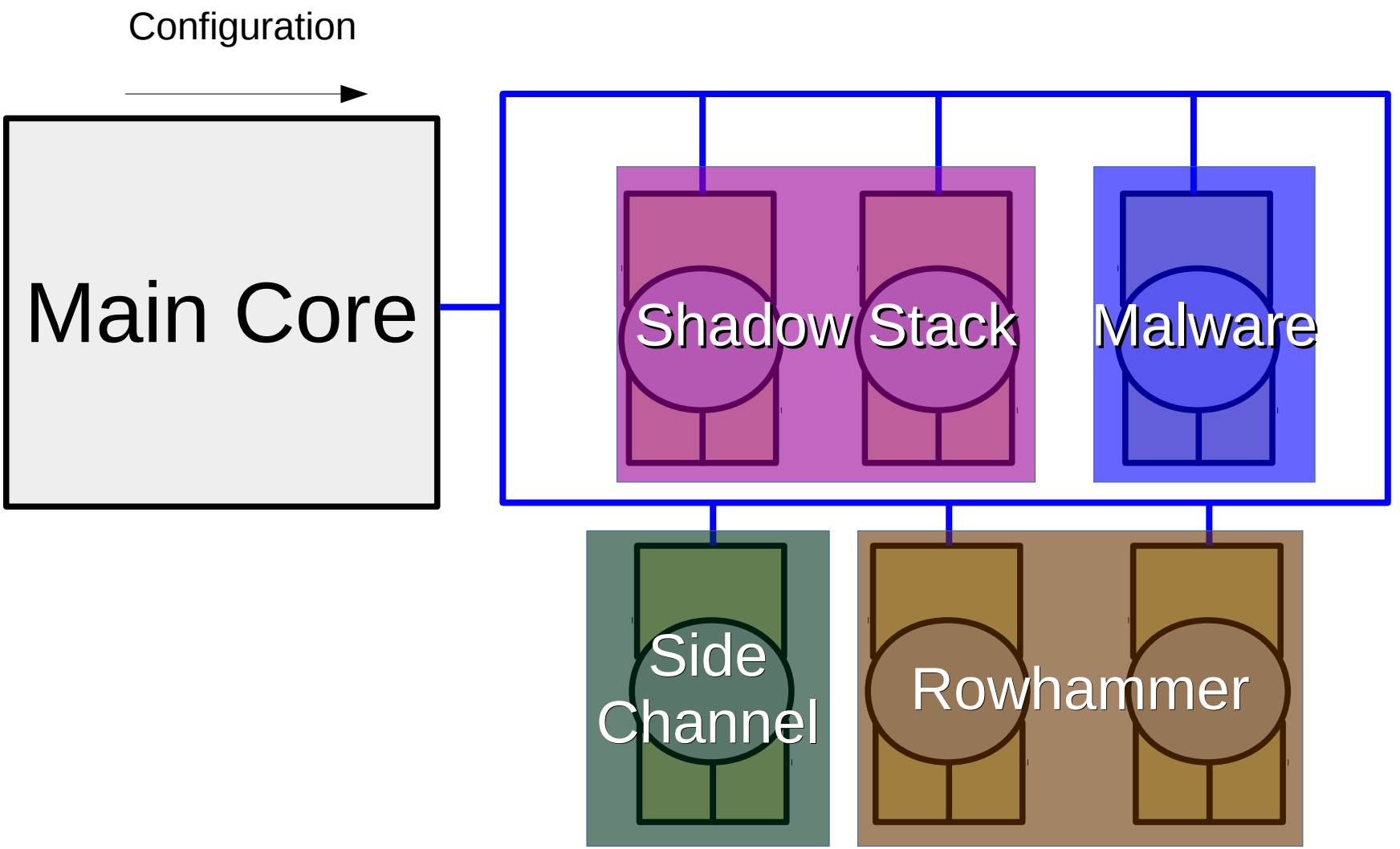
Security: Attacks



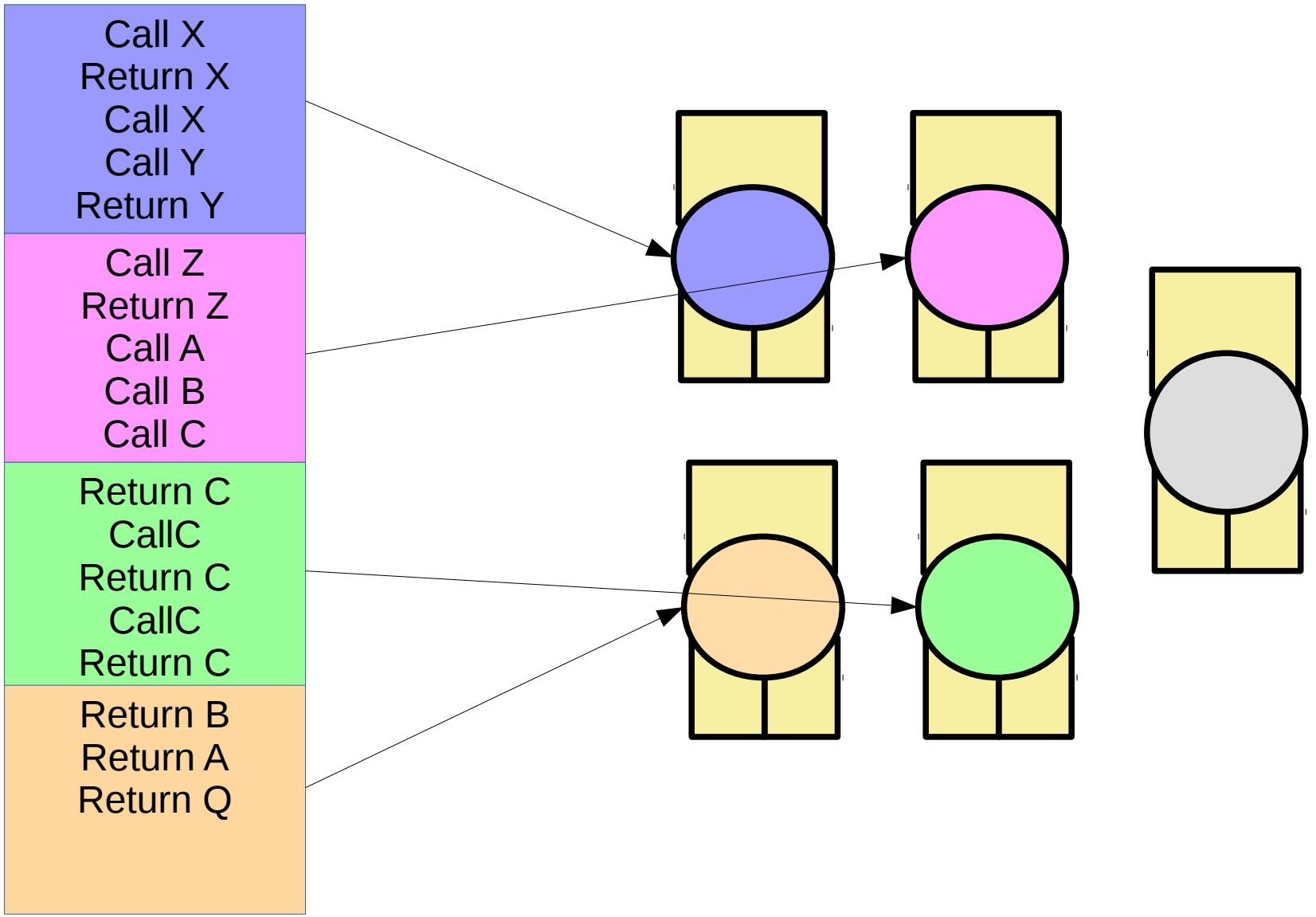
Security: Architecture



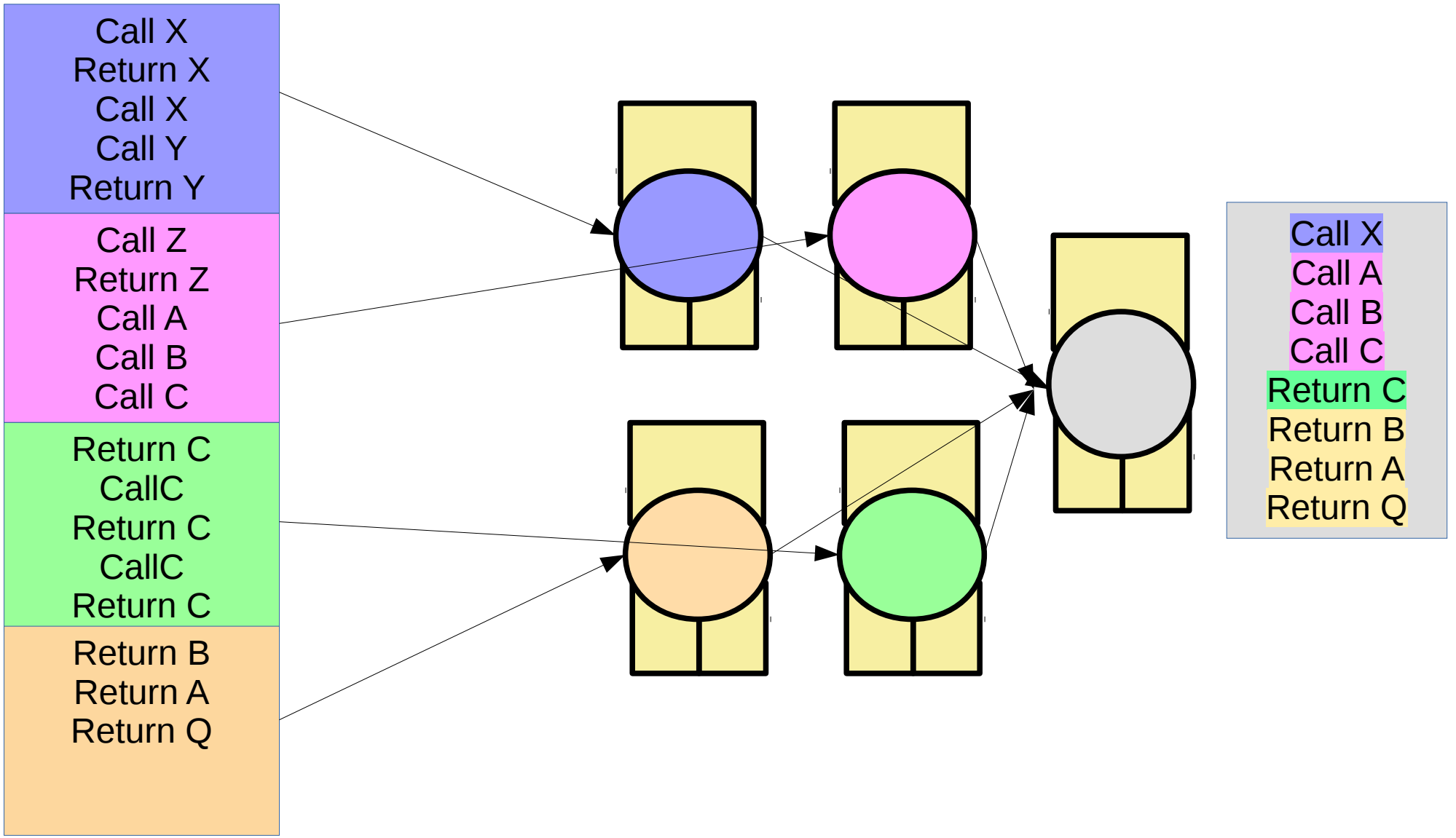
Security: Programming Model



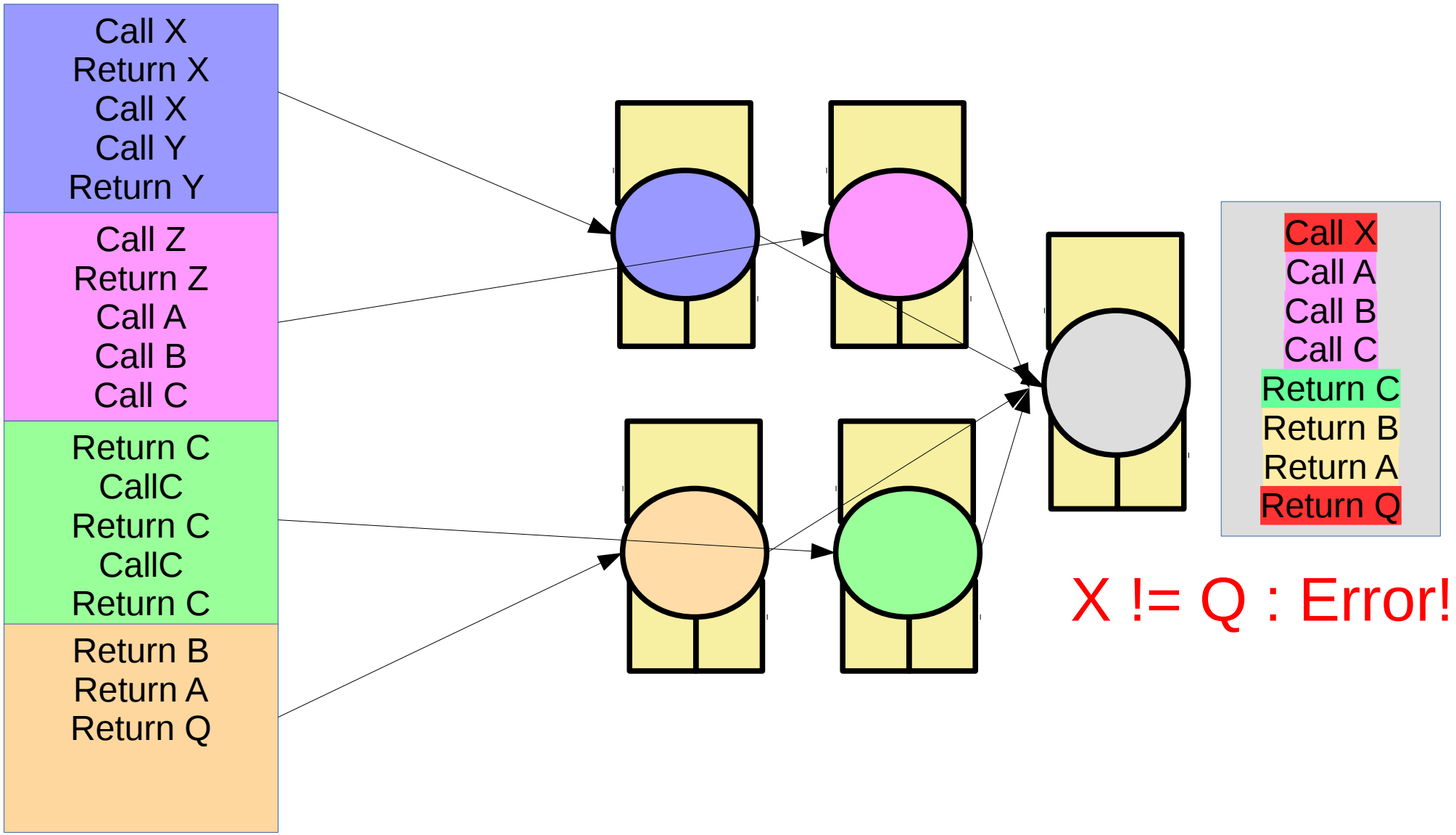
Example: Shadow Stack



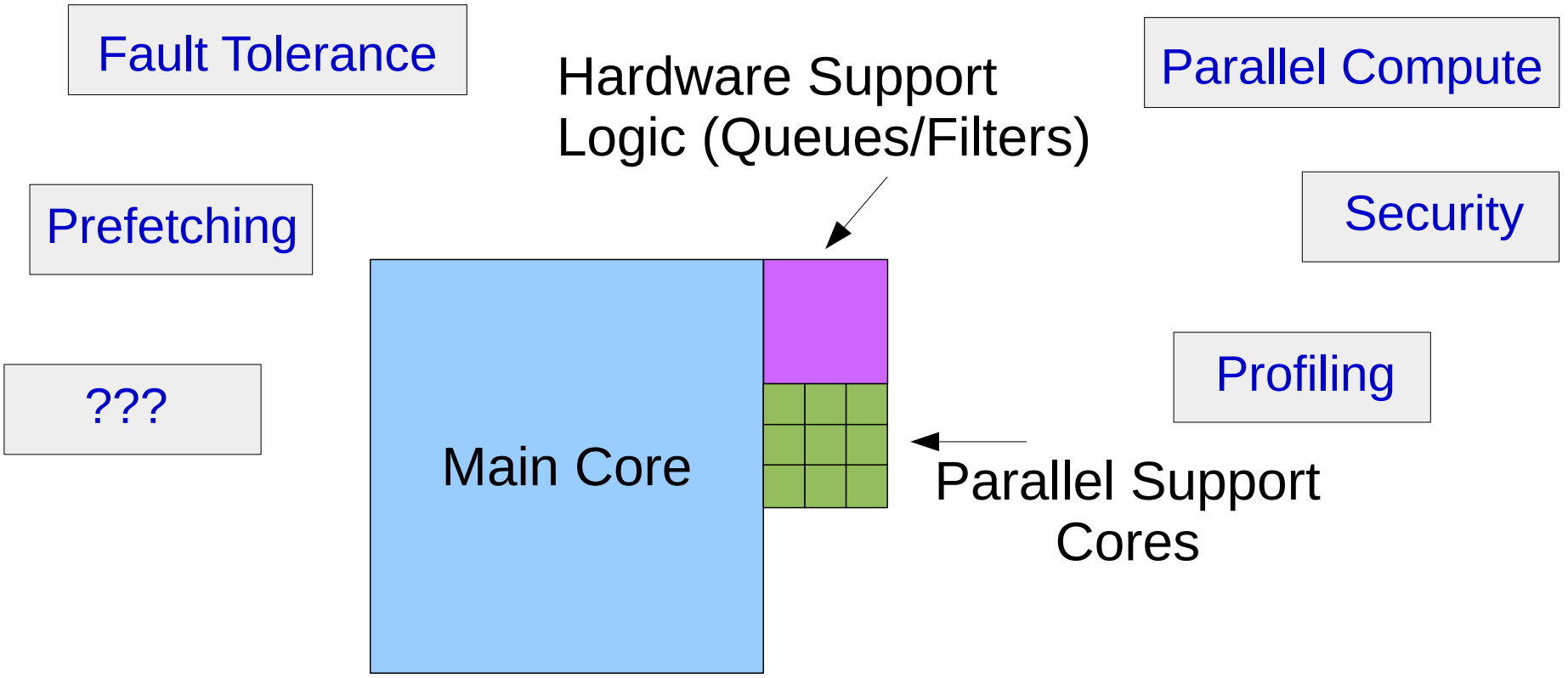
Example: Shadow Stack



Example: Shadow Stack

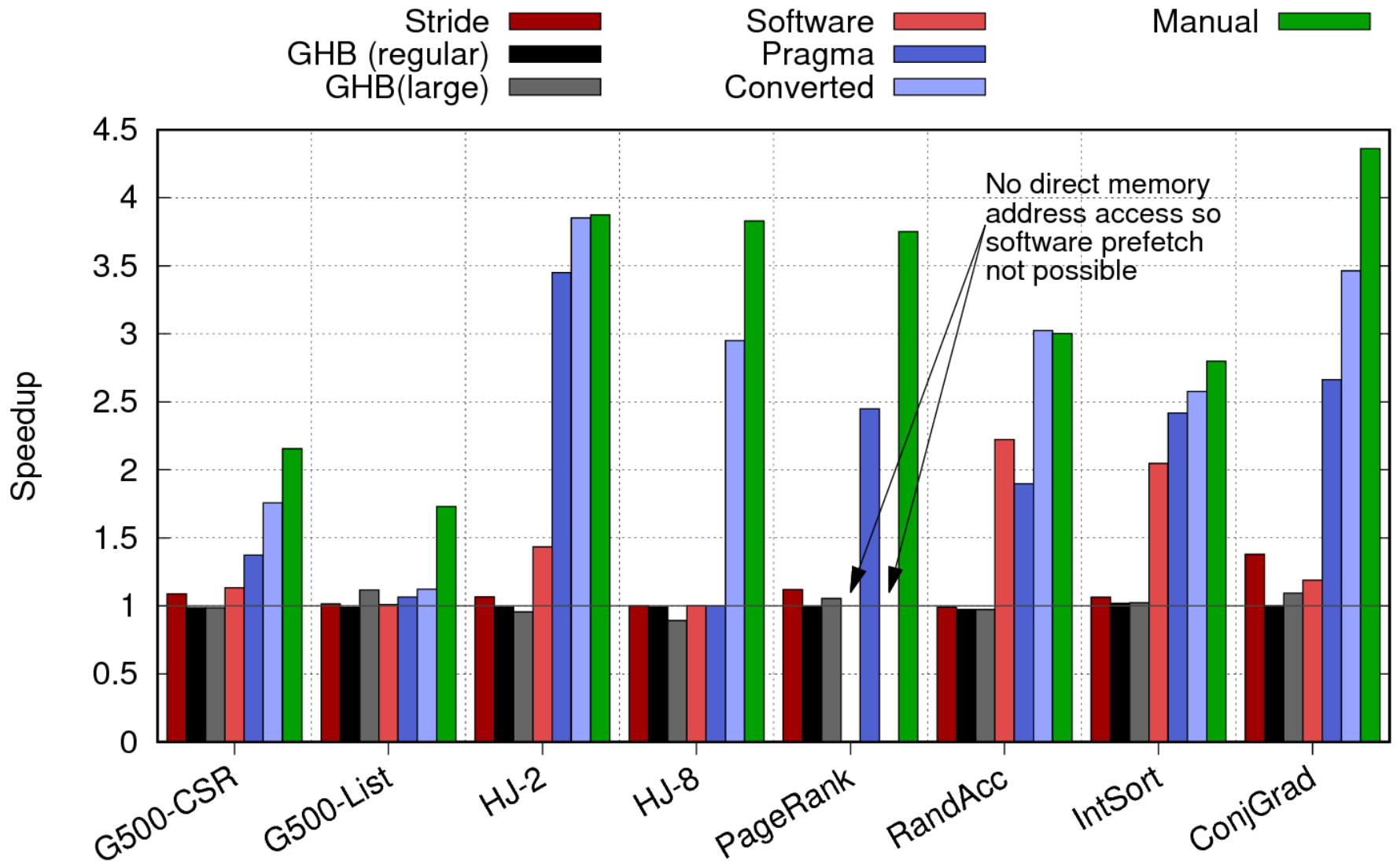


The Bigger Picture: Parallel Support Cores

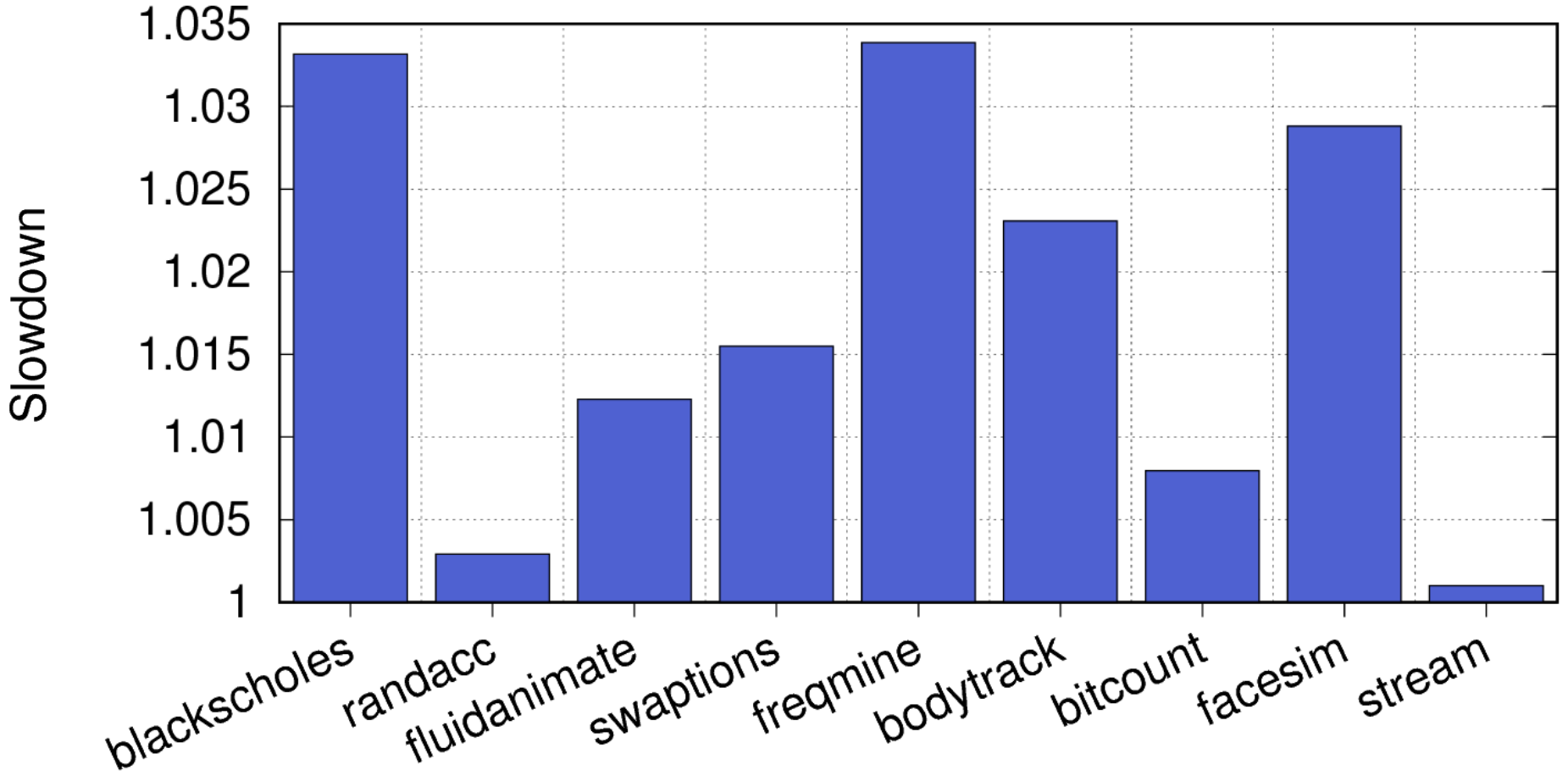


Sam Ainsworth
University of Cambridge
sam.ainsworth@cl.cam.ac.uk

Programmable Prefetching: Results



Error Detection: Slowdown



Security: Number of Small Cores Required