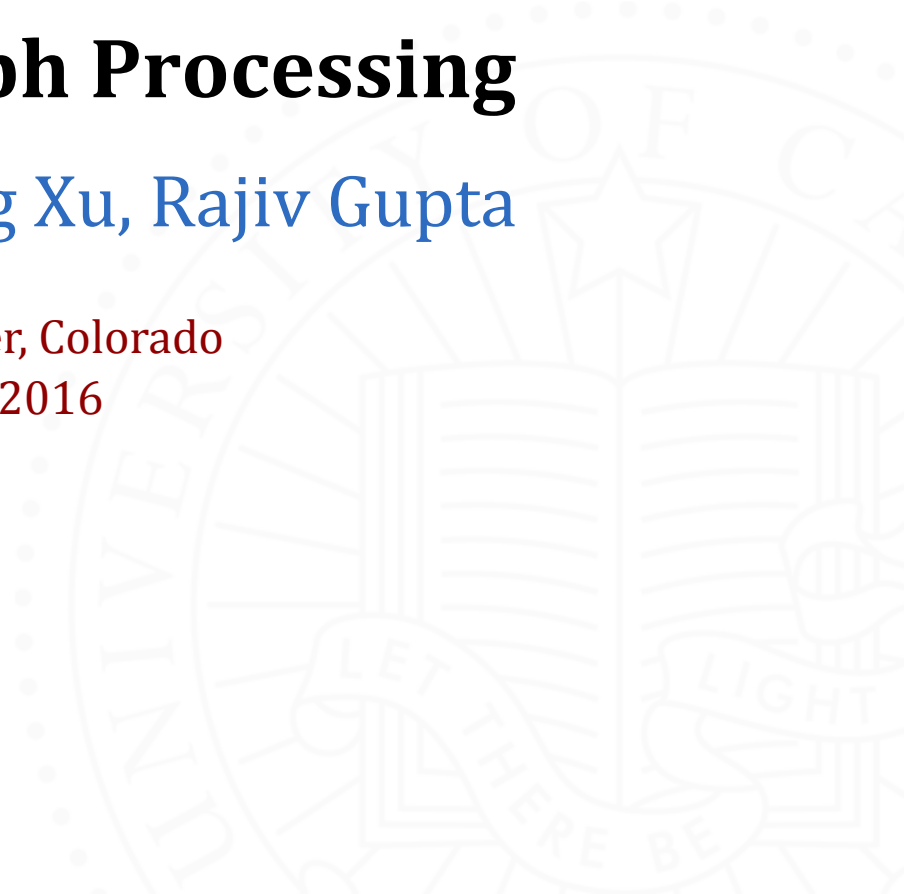


Load the Edges You Need: A Generic I/O Optimization for Disk-based Graph Processing

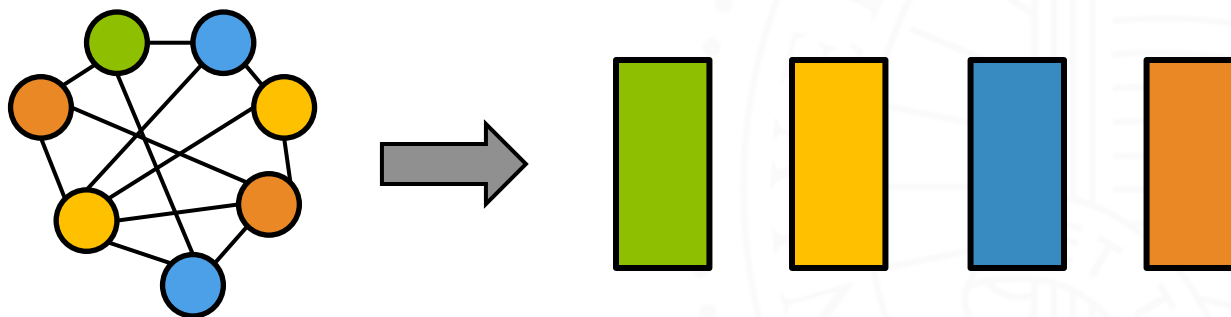
Keval Vora, Guoqing Xu, Rajiv Gupta

ATC'16 – Denver, Colorado
June 24, 2016



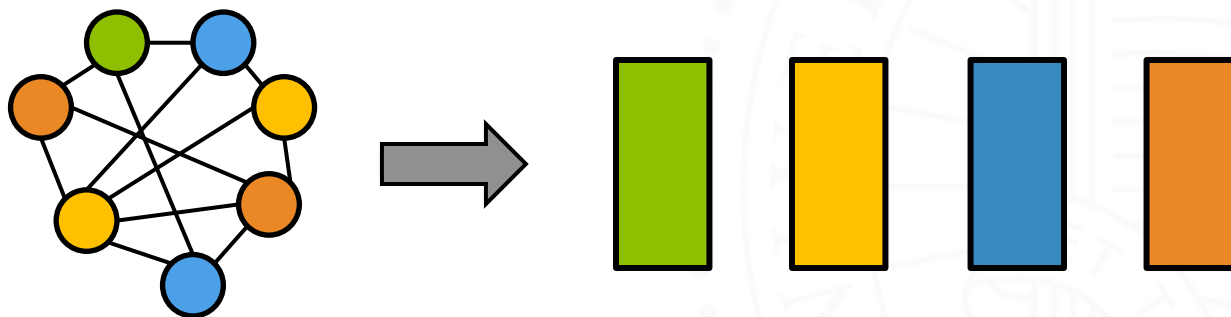
Graph Processing

- ▶ Challenging due to ever-growing graph sizes
- ▶ Out-of-core processing on a single machine
 - Cheap, feasible & competitive
 - GraphChi [OSDI'12], X-Stream [SOSP'13], etc.
- ▶ Disk friendly static partitions
 - Carefully orchestrate reads from disk



Out-of-core Graph Processing

- ▶ GraphChi [OSDI'12]
- ▶ Graph is split across multiple shards
 - Created during pre-processing
 - Remain **static** throughout processing
- ▶ Process by loading an entire shard one at a time
- ▶ Iterative processing
 - Each shard is loaded multiple times



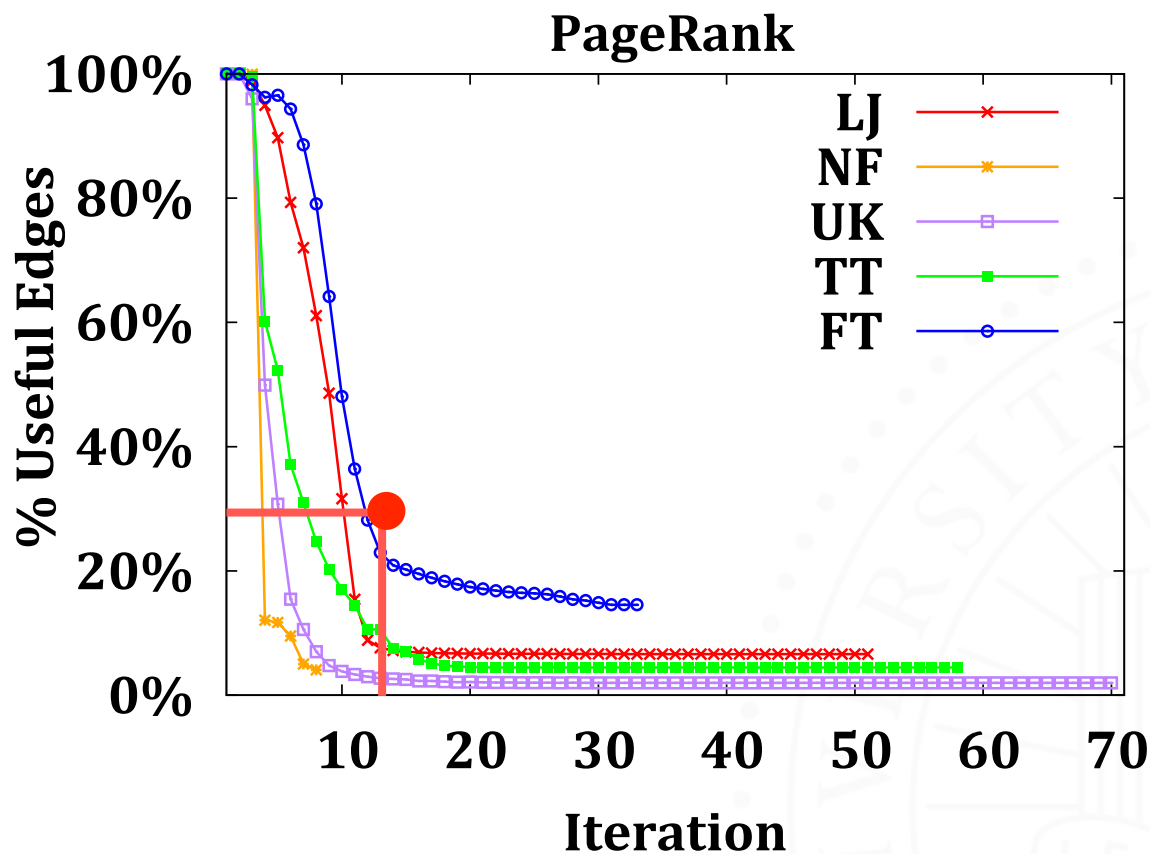
I/O Bound

- ▶ Loading static partitions is expensive
 - ▶ GraphChi spends 73-88% of the time in loading
 - ▶ X-Stream spends 73-80% of the time in loading



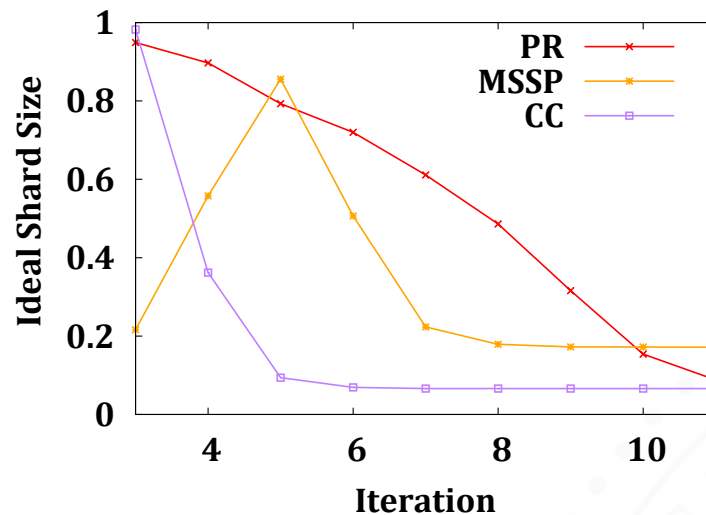
Opportunity

- ▶ Not all edges are always required



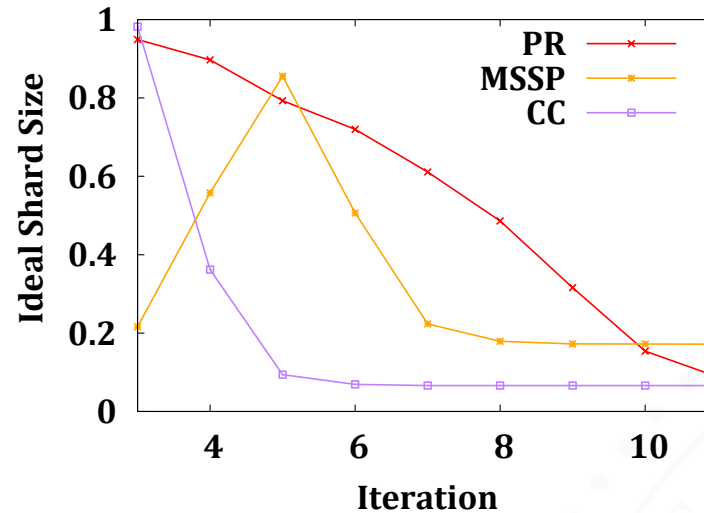
Challenges

- ▶ Different algorithms behave differently



Challenges

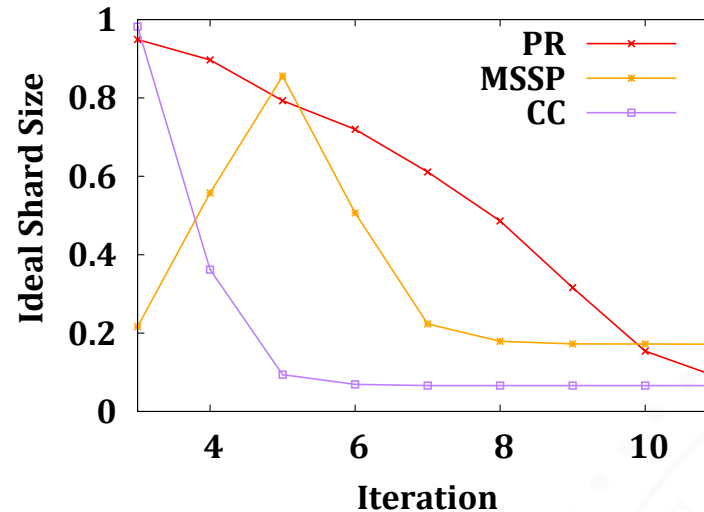
- ▶ Different algorithms behave differently



- ▶ Need **dynamic partitions** to capture working set

Challenges

- ▶ Different algorithms behave differently



- ▶ Need **dynamic partitions** to capture working set

**How to create
dynamic partitions
*on the fly?***

**How to process
dynamic
partitions?**

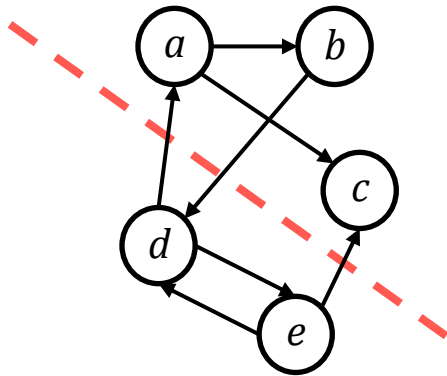
Outline

- › Dynamic Shards
- › Creating Dynamic Shards
- › Processing Dynamic Shards
- › Accumulation based Computation
- › Evaluation
- › Conclusion



Shards

- › Maintain locality while processing



Shard 0

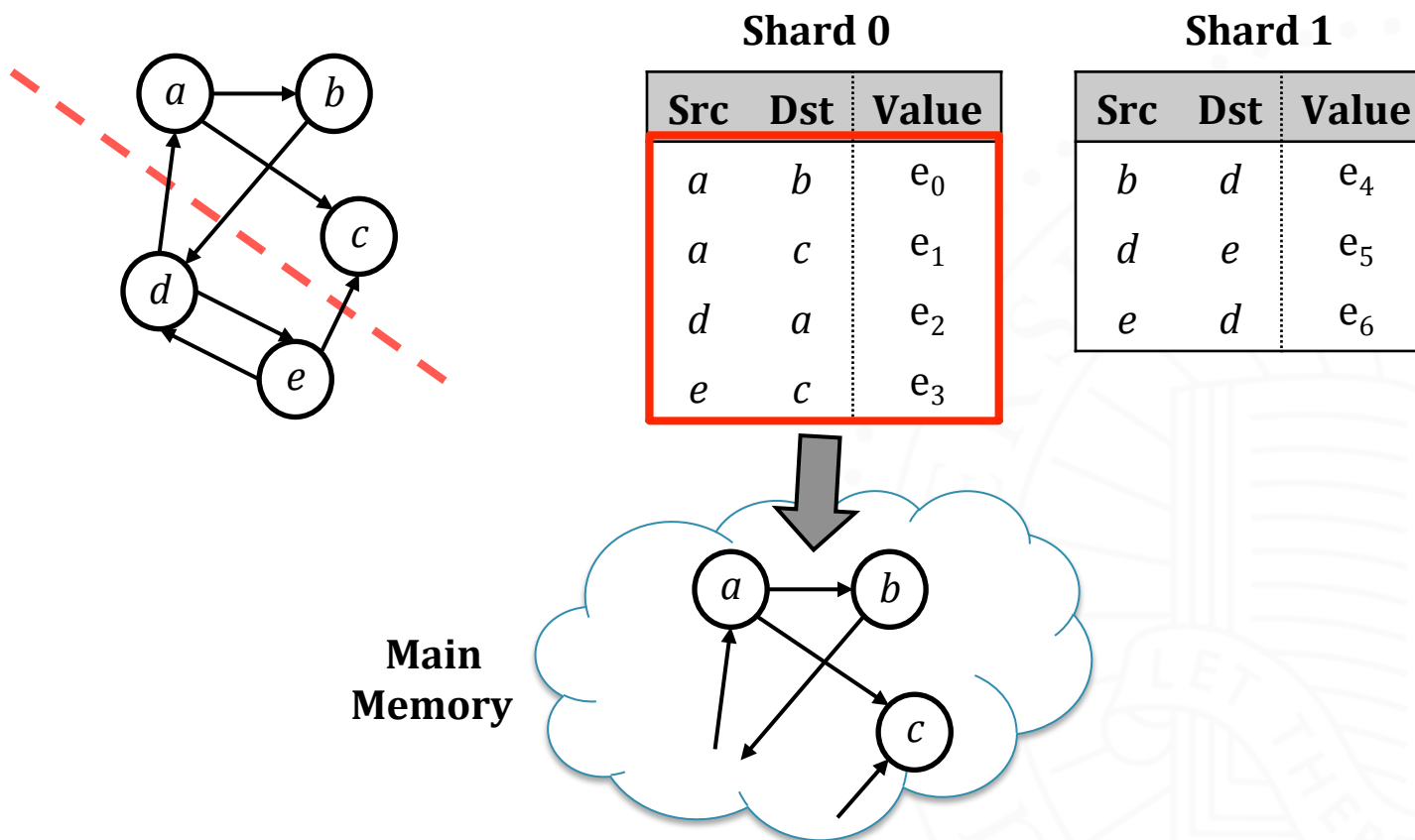
Src	Dst	Value
<i>a</i>	<i>b</i>	e_0
<i>a</i>	<i>c</i>	e_1
<i>d</i>	<i>a</i>	e_2
<i>e</i>	<i>c</i>	e_3

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	e_4
<i>d</i>	<i>e</i>	e_5
<i>e</i>	<i>d</i>	e_6

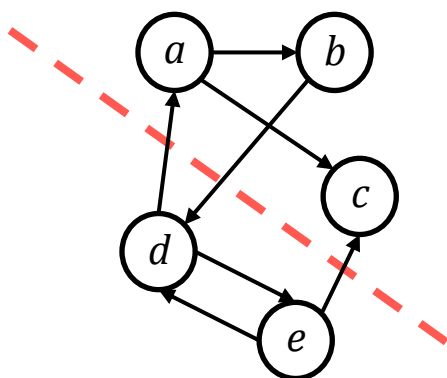
Shards

- › Maintain locality while processing
- › Load, compute, store



Shards

- Maintain locality while processing
- Load, compute, store



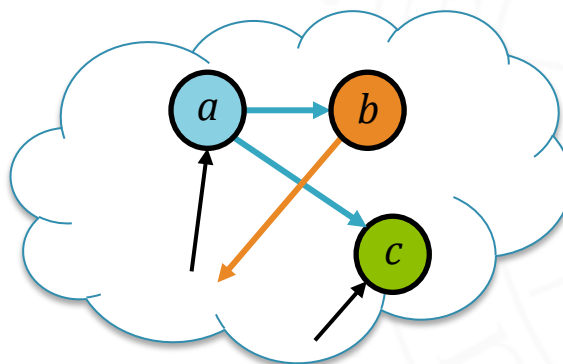
Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	e_0
<i>a</i>	<i>c</i>	e_1
<i>d</i>	<i>a</i>	e_2
<i>e</i>	<i>c</i>	e_3

Shard 1

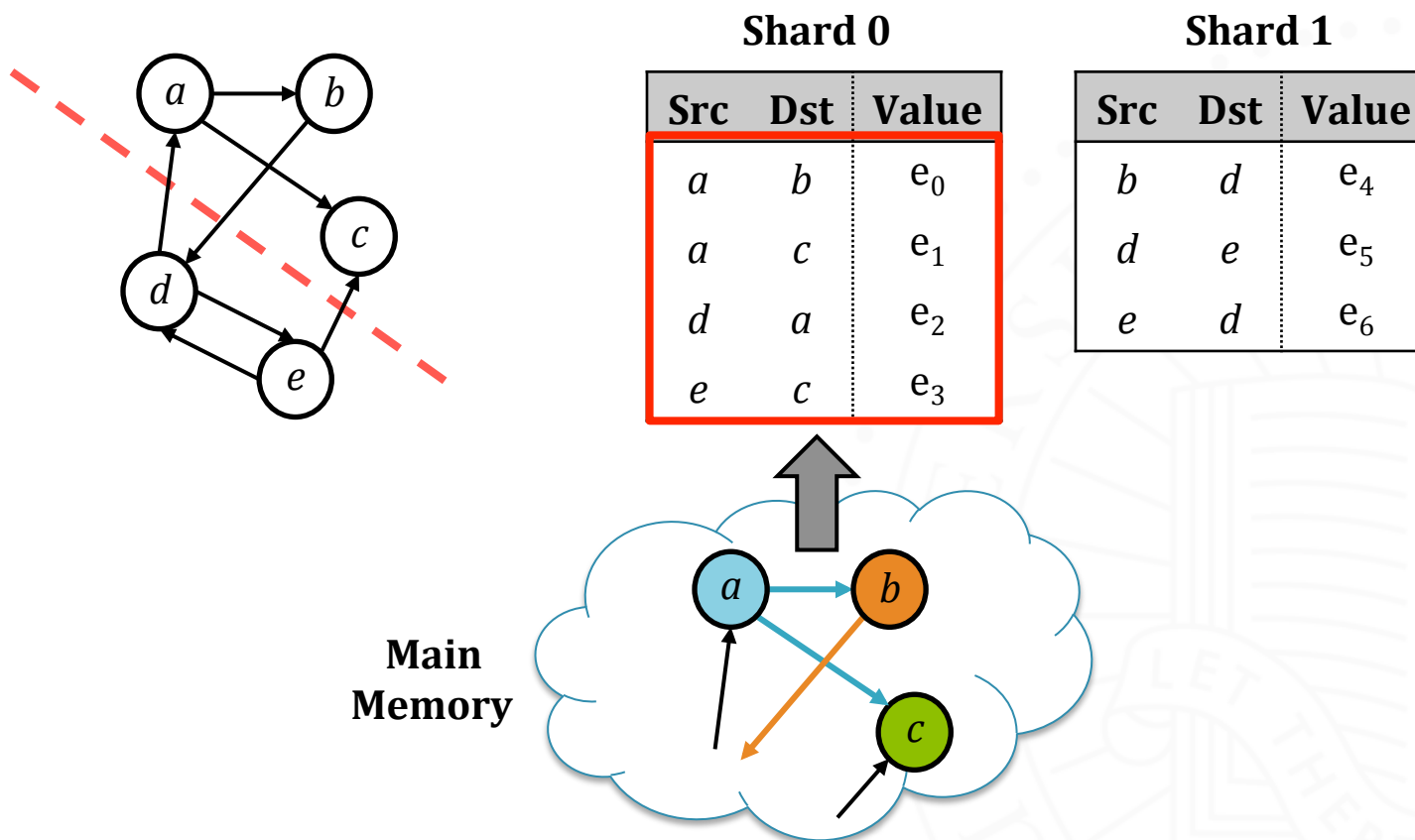
Src	Dst	Value
<i>b</i>	<i>d</i>	e_4
<i>d</i>	<i>e</i>	e_5
<i>e</i>	<i>d</i>	e_6

Main
Memory



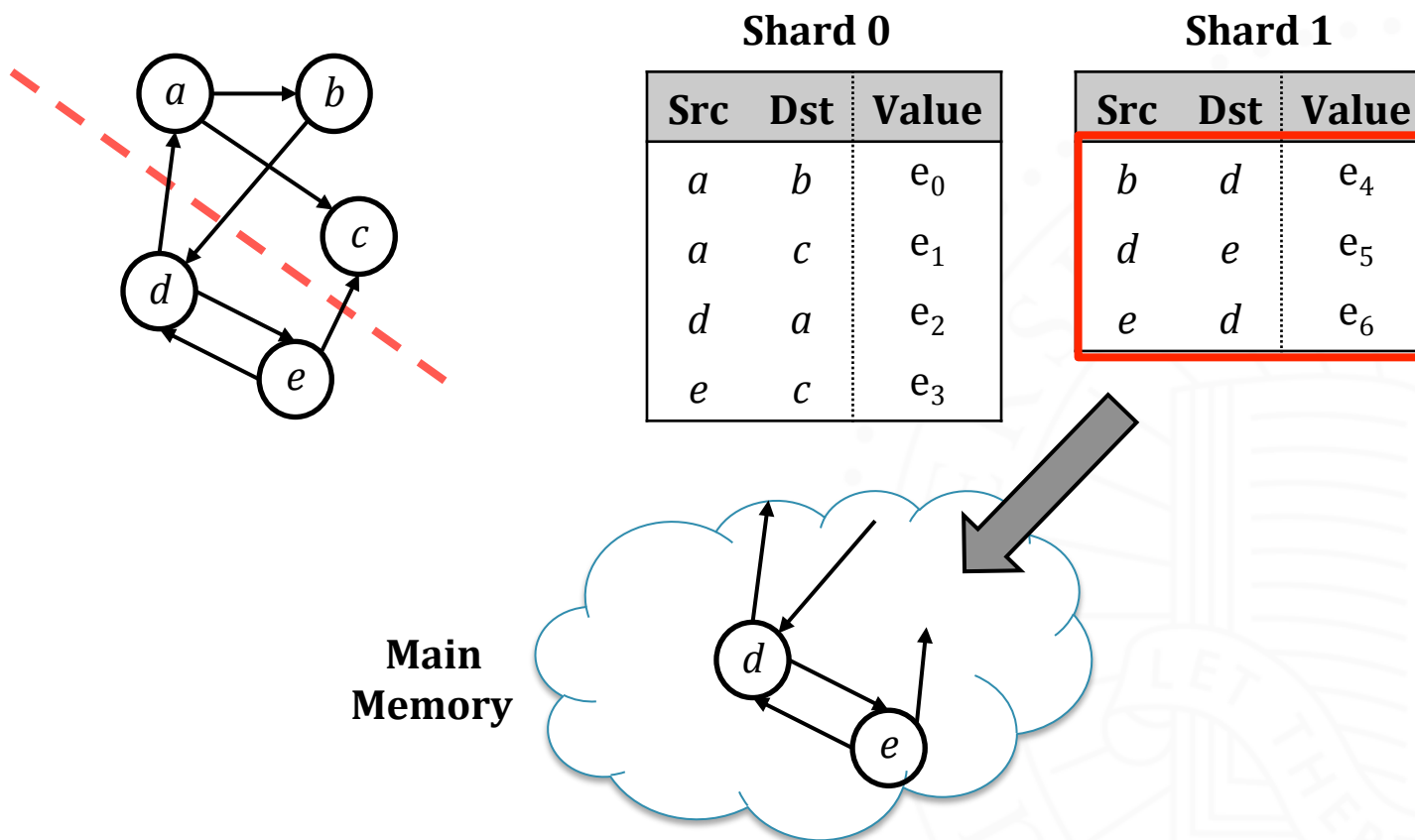
Shards

- Maintain locality while processing
- Load, compute, store



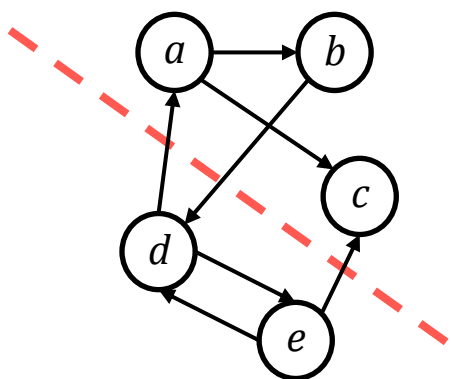
Shards

- Maintain locality while processing
- Load, compute, store



Shards

- Maintain locality while processing
- Load, compute, store



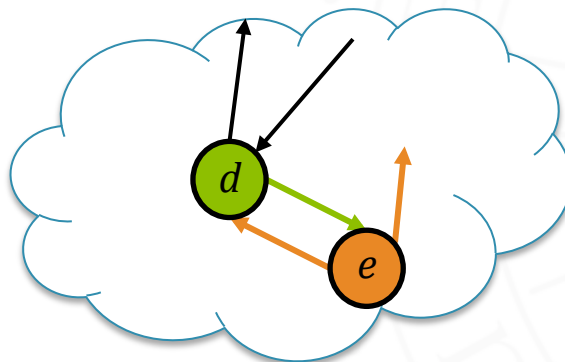
Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	e_0
<i>a</i>	<i>c</i>	e_1
<i>d</i>	<i>a</i>	e_2
<i>e</i>	<i>c</i>	e_3

Shard 1

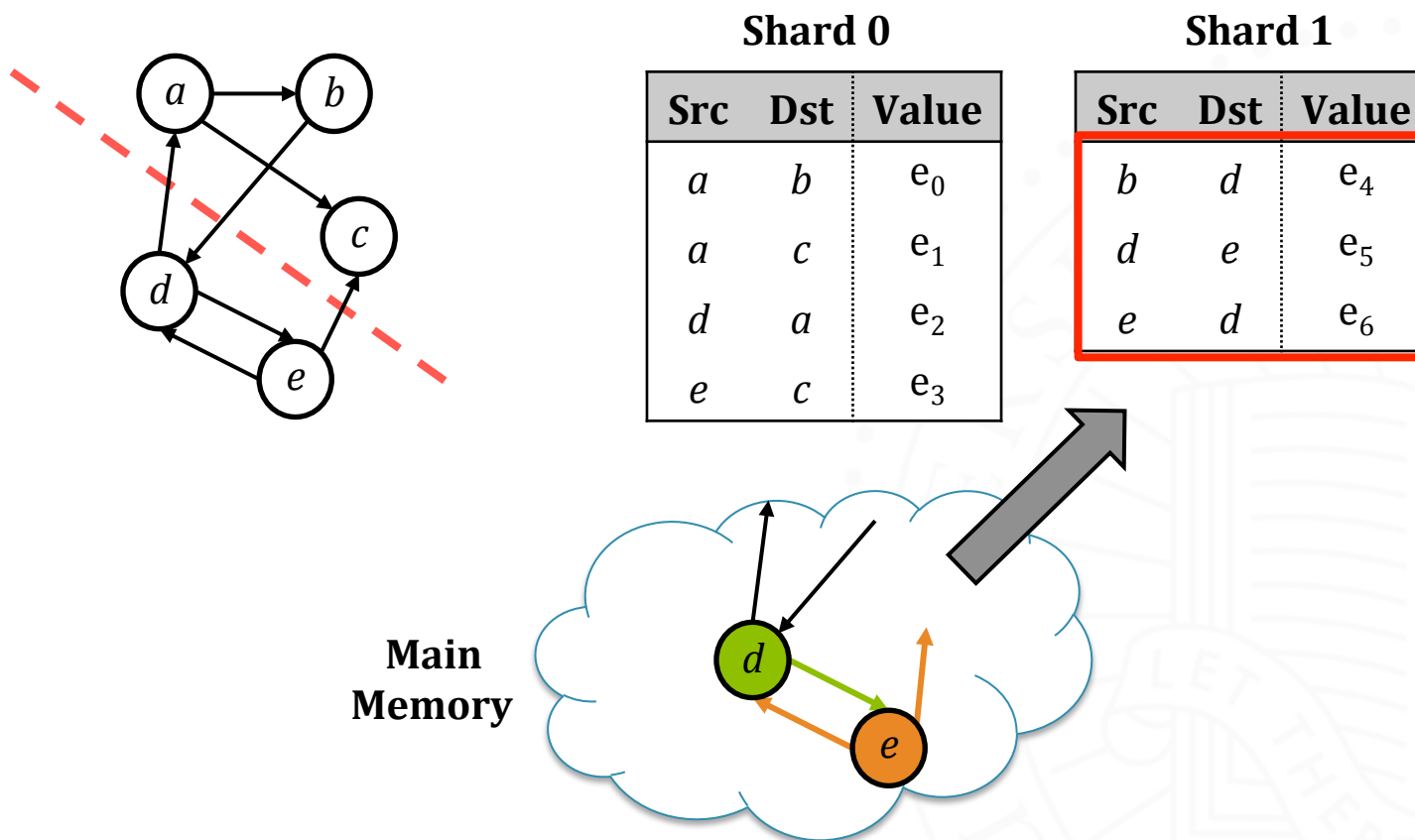
Src	Dst	Value
<i>b</i>	<i>d</i>	e_4
<i>d</i>	<i>e</i>	e_5
<i>e</i>	<i>d</i>	e_6

Main
Memory



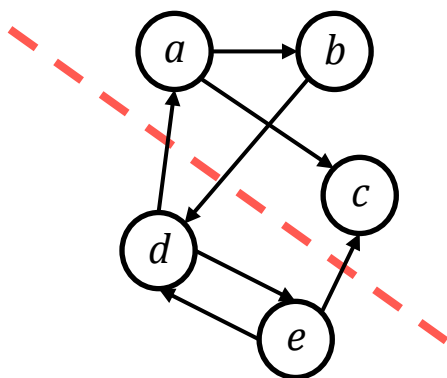
Shards

- Maintain locality while processing
- Load, compute, store



Shards

- Maintain locality while processing
- Load, compute, store
- Iterative execution



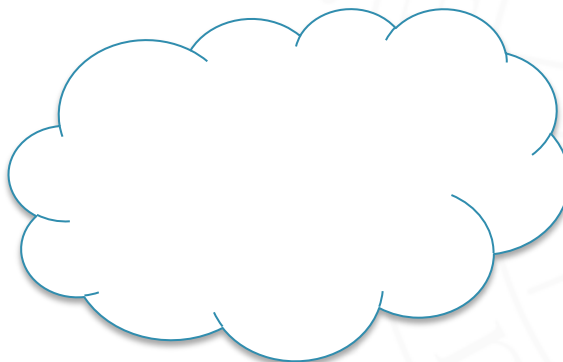
Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	e_0
<i>a</i>	<i>c</i>	e_1
<i>d</i>	<i>a</i>	e_2
<i>e</i>	<i>c</i>	e_3

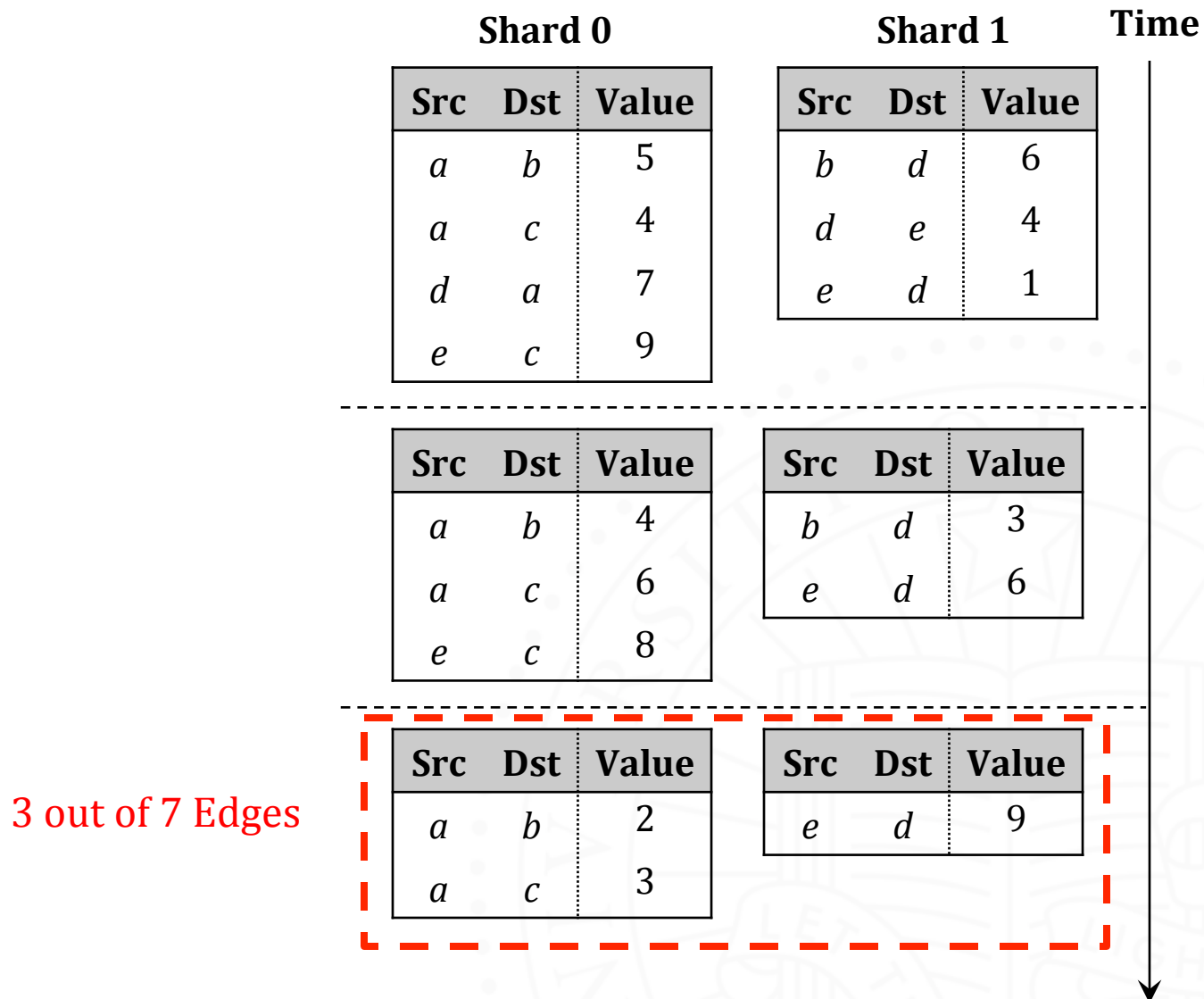
Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	e_4
<i>d</i>	<i>e</i>	e_5
<i>e</i>	<i>d</i>	e_6

Main
Memory



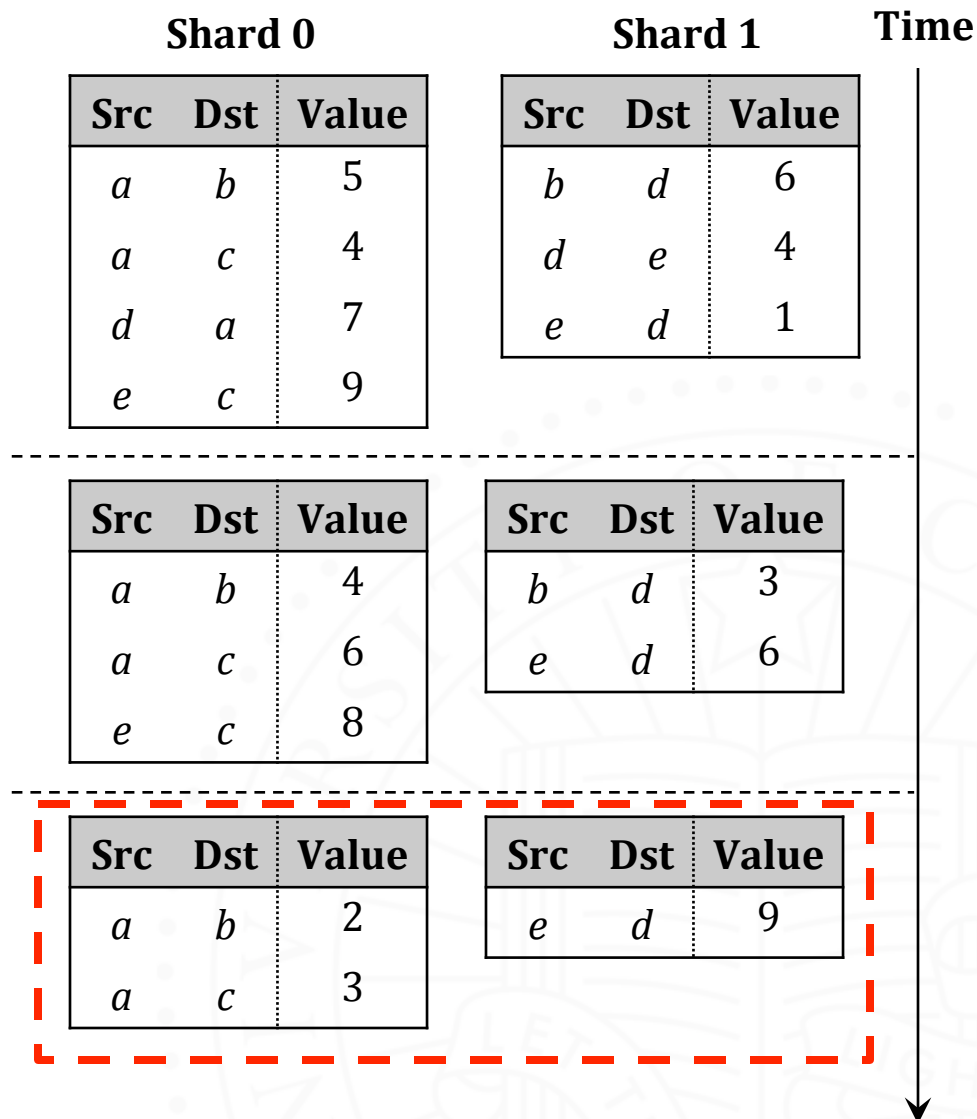
Dynamic Shards



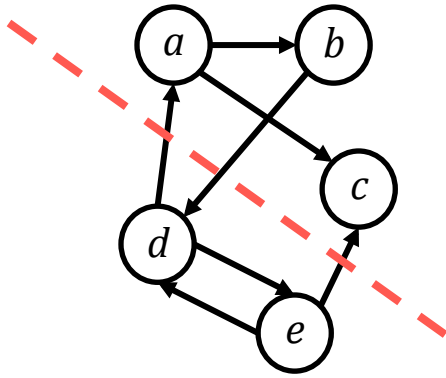
Dynamic Shards

How to efficiently
create dynamic
shards *on the fly*?

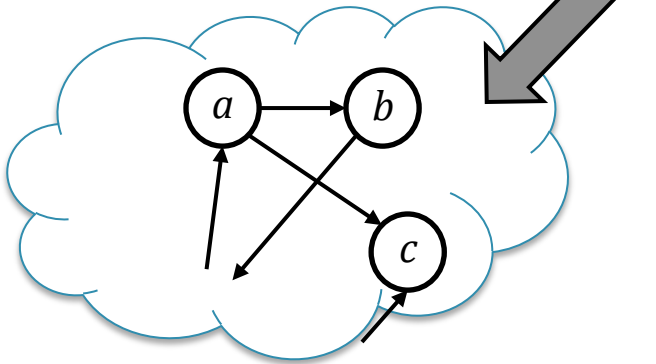
3 out of 7 Edges



Creating Dynamic Shards



**Main
Memory**



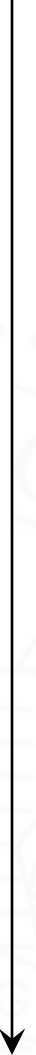
Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

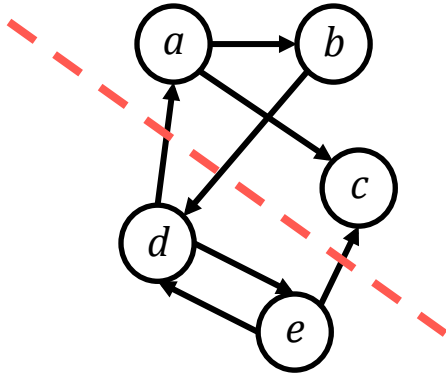
Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

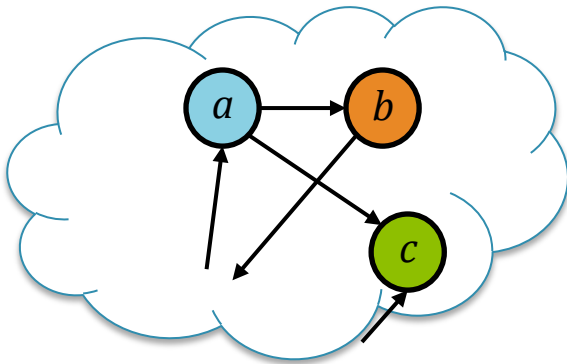
Time



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

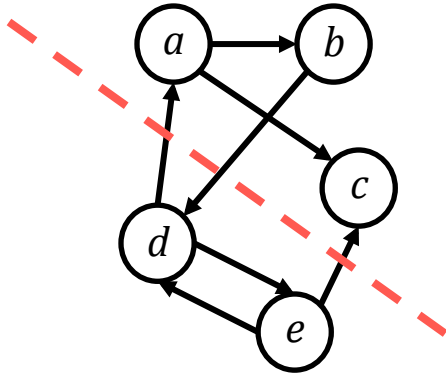
Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

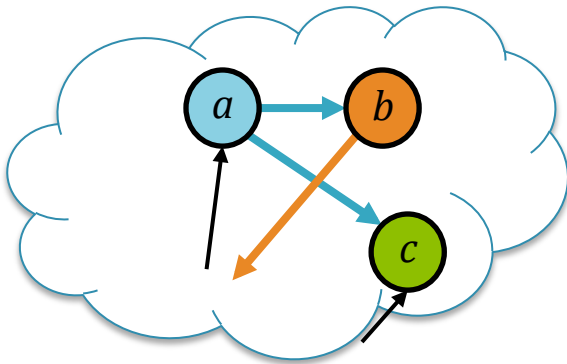
Time



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

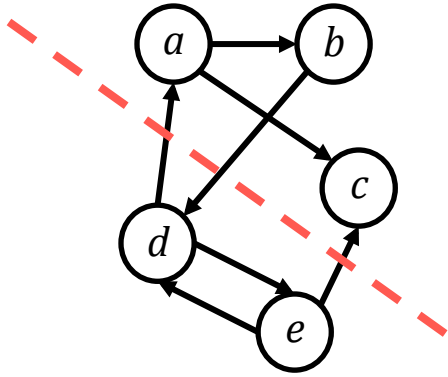
Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

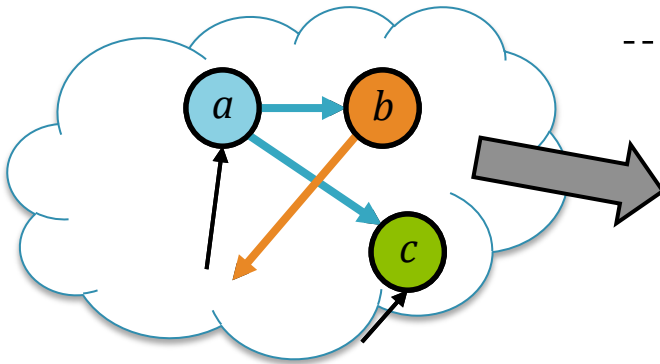
Time



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

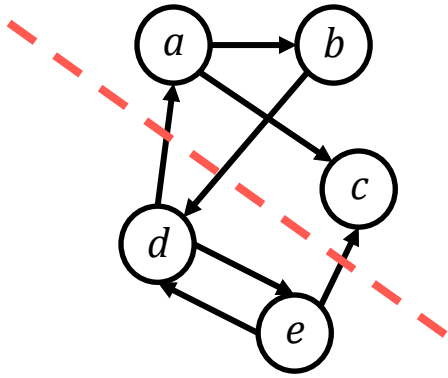
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6

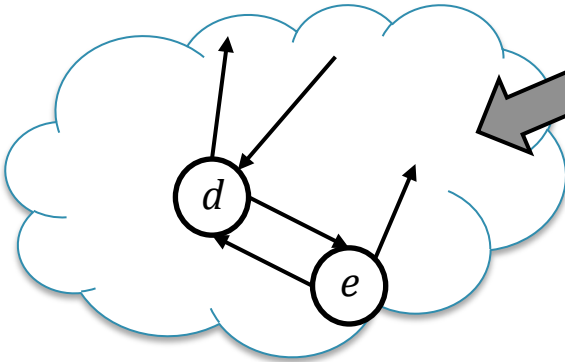
Src	Dst	Value
<i>b</i>	<i>d</i>	3



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>		9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

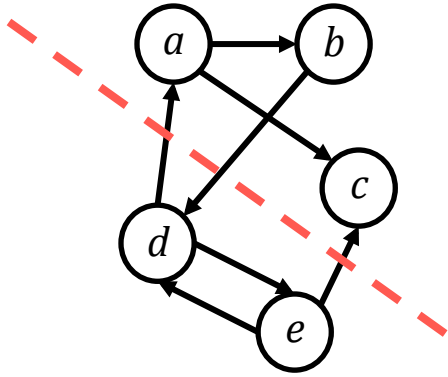
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6

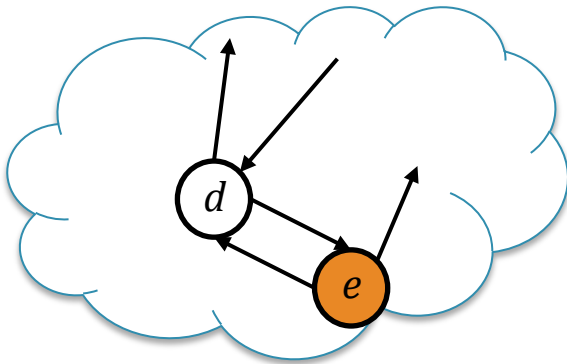
Src	Dst	Value
<i>b</i>	<i>d</i>	3



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

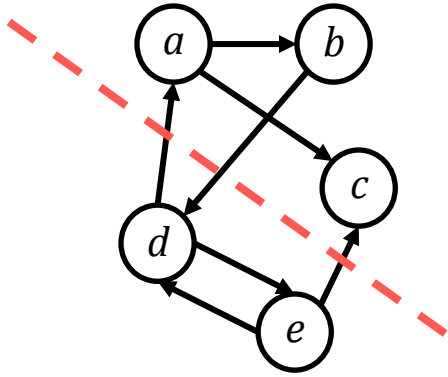
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6

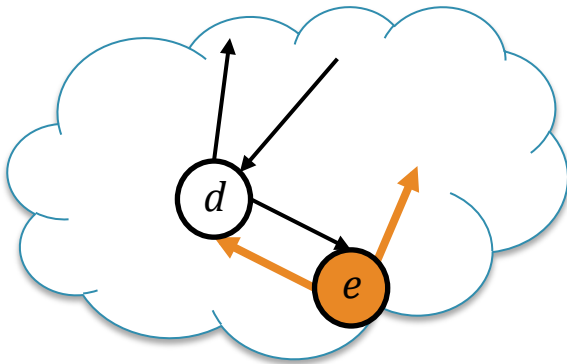
Src	Dst	Value
<i>b</i>	<i>d</i>	3



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

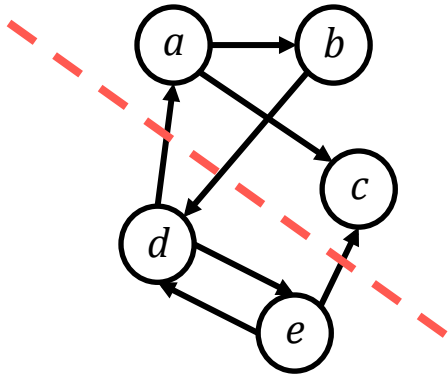
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6

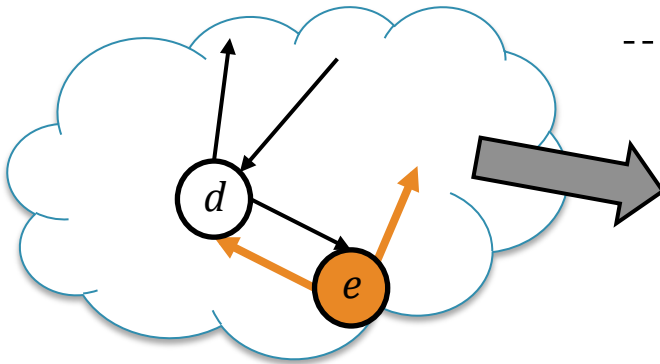
Src	Dst	Value
<i>b</i>	<i>d</i>	3



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

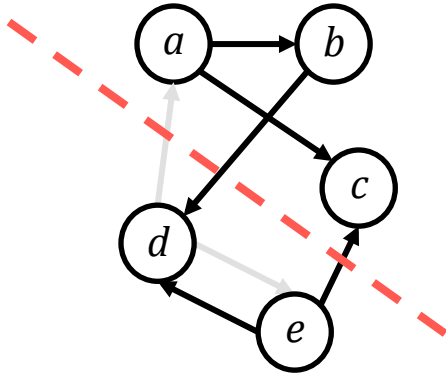
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

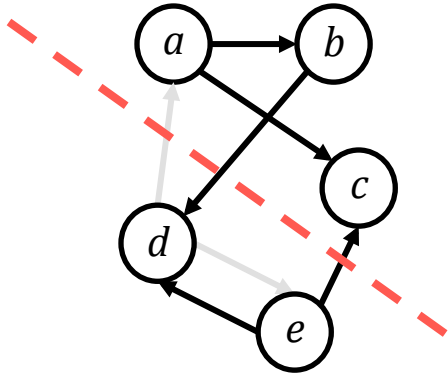
Time



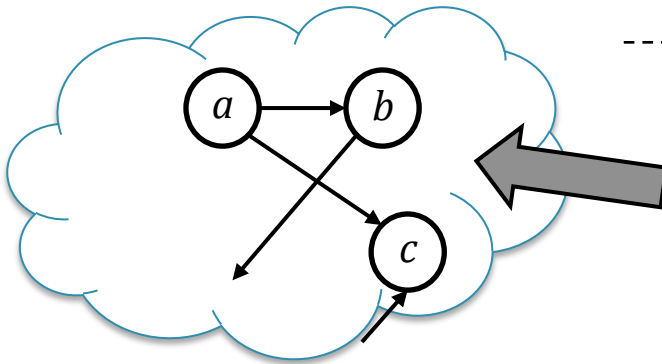
Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

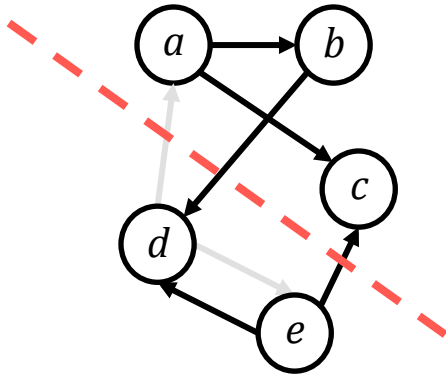
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

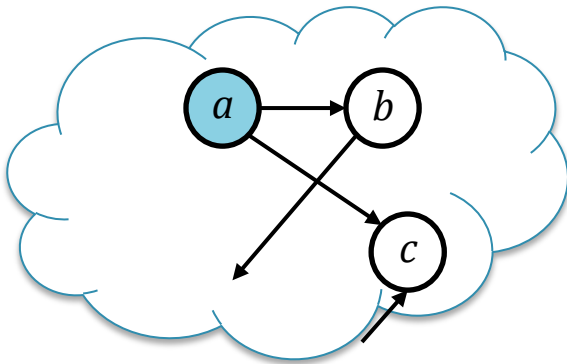
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

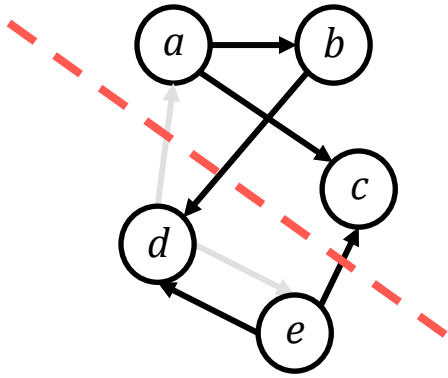
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

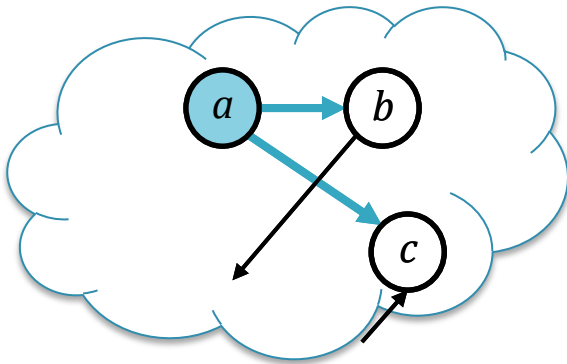
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

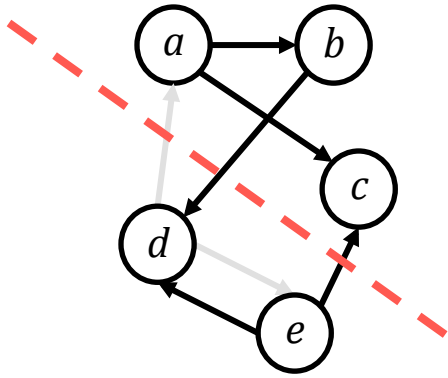
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

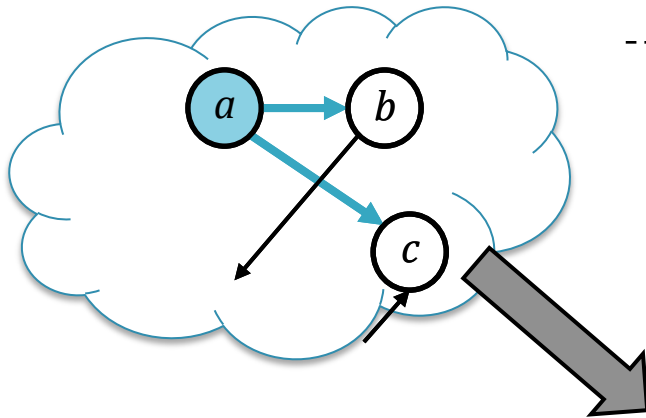
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

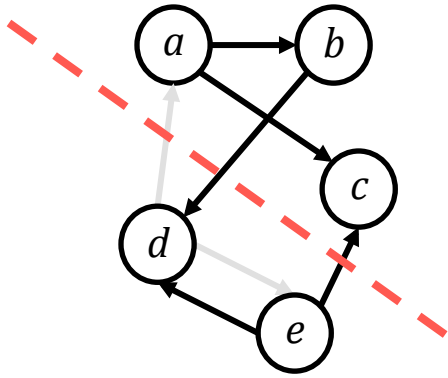
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
<i>a</i>	<i>b</i>	2
<i>a</i>	<i>c</i>	3

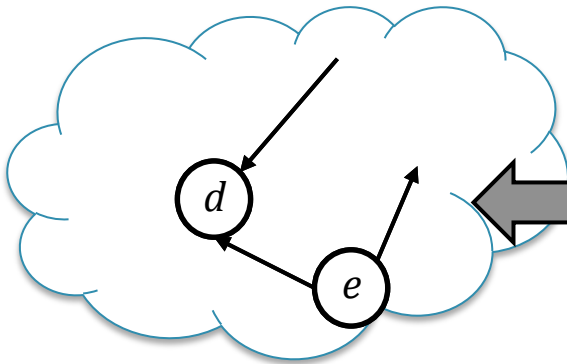
Src	Dst	Value
-----	-----	-------



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

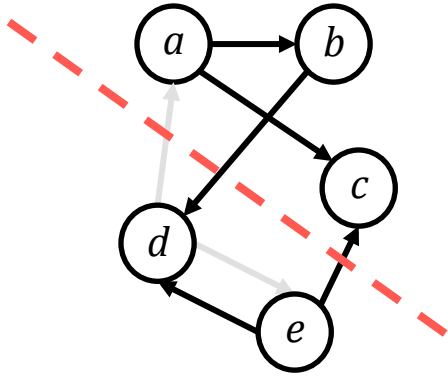
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
<i>a</i>	<i>b</i>	2
<i>a</i>	<i>c</i>	3

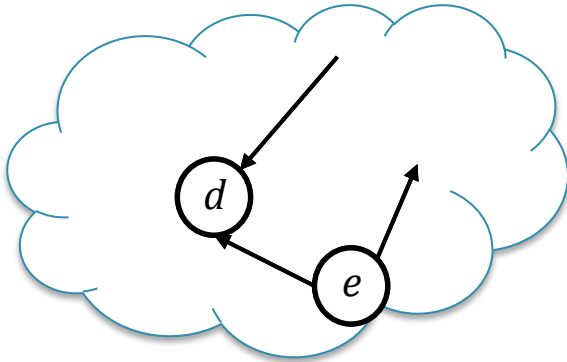
Src	Dst	Value
-----	-----	-------



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

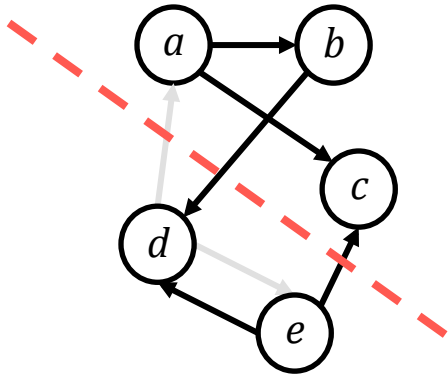
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
<i>a</i>	<i>b</i>	2
<i>a</i>	<i>c</i>	3

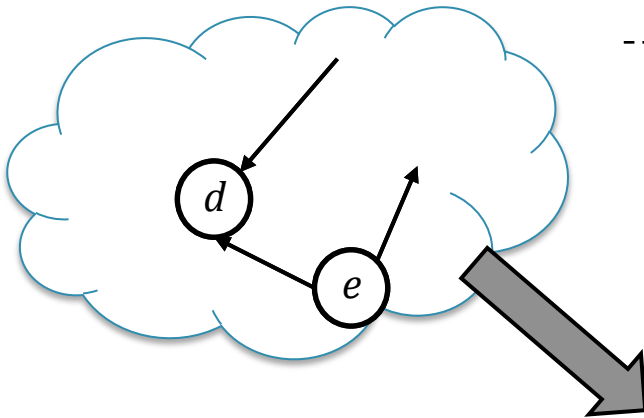
Src	Dst	Value
-----	-----	-------



Creating Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

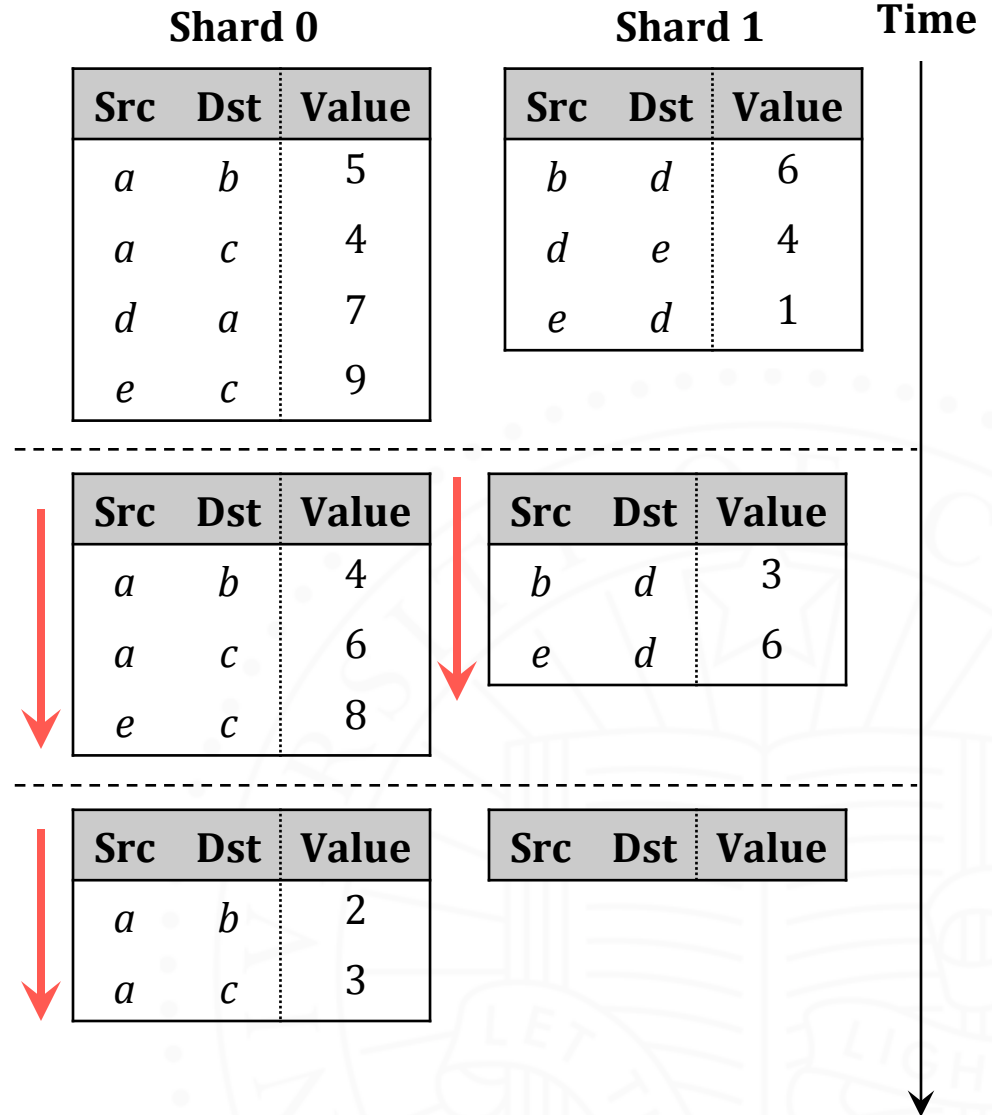
Src	Dst	Value
<i>a</i>	<i>b</i>	2
<i>a</i>	<i>c</i>	3

Src	Dst	Value
-----	-----	-------

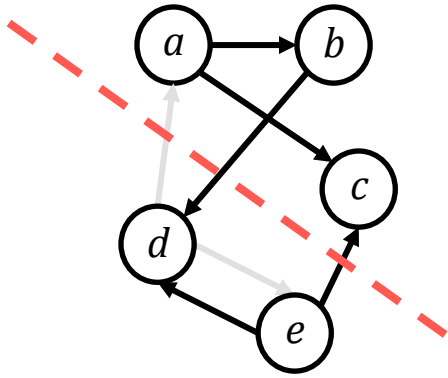


Creating Dynamic Shards

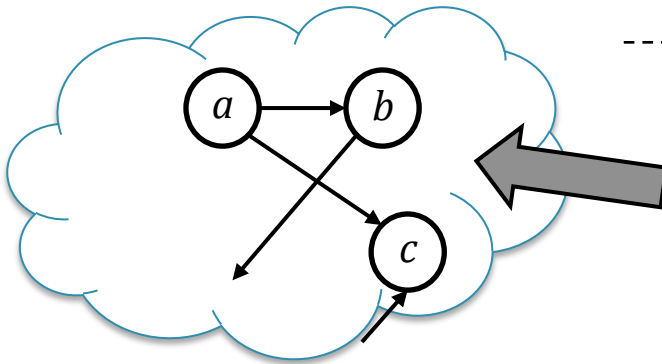
- Sequential writes
- Light-weight



Processing Dynamic Shards



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

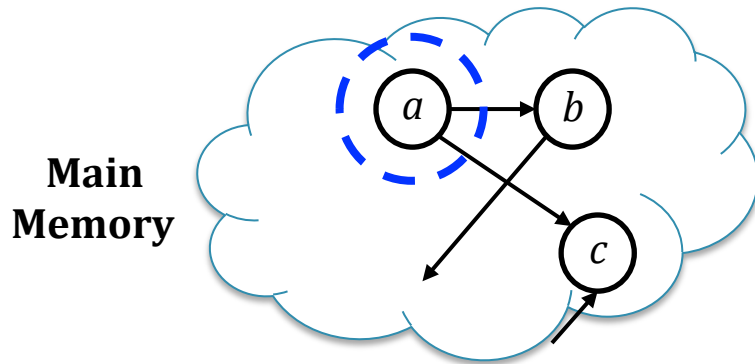
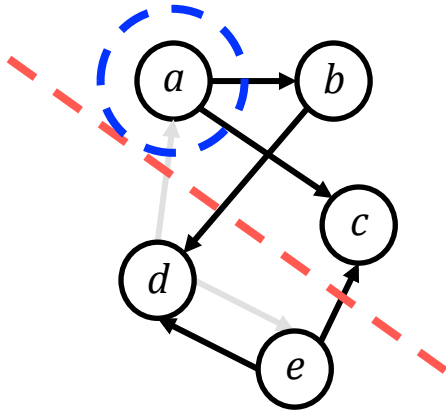
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Processing Dynamic Shards



**How to process
vertices with
missing edges?**

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

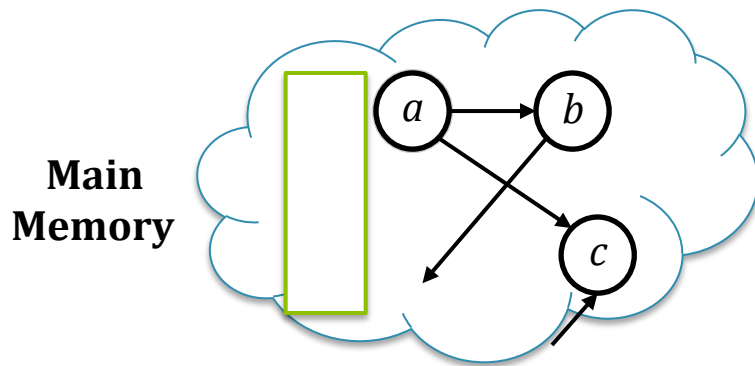
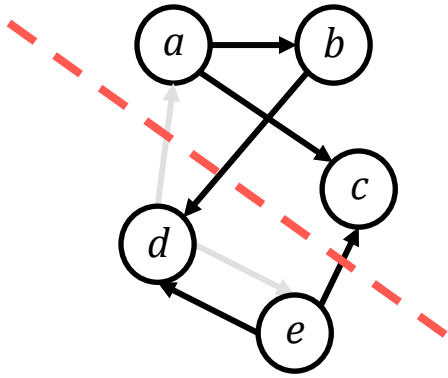
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Delay based Processing



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

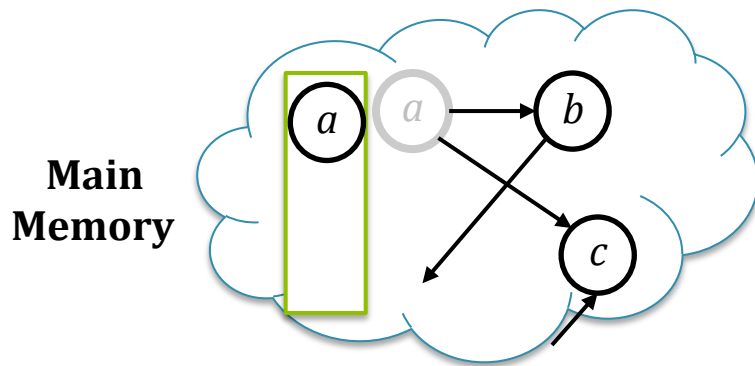
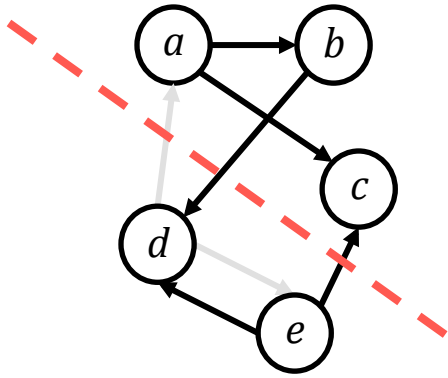
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Delay based Processing



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

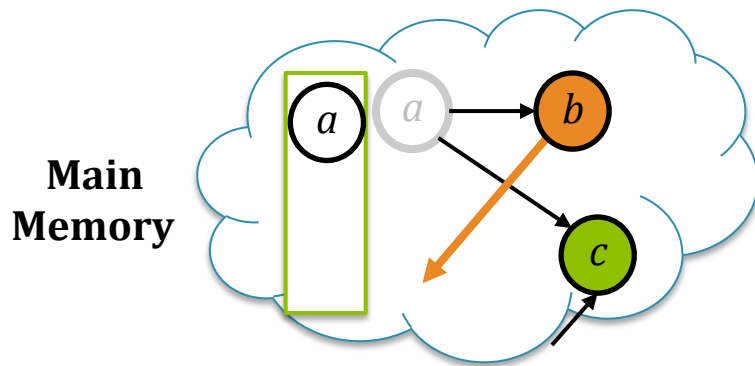
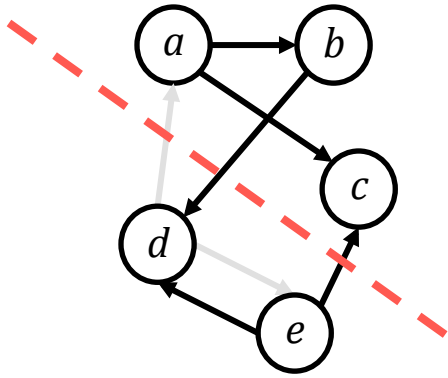
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Delay based Processing



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

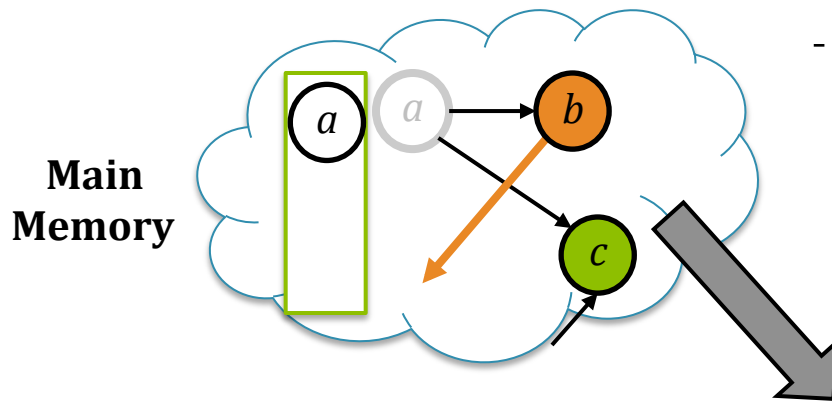
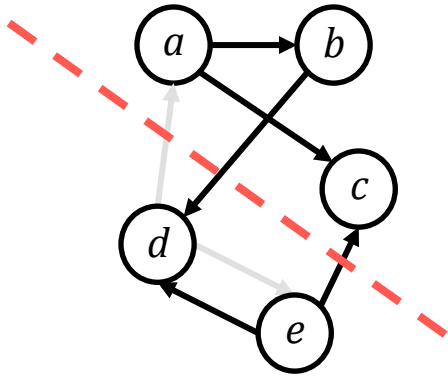
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6



Delay based Processing



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

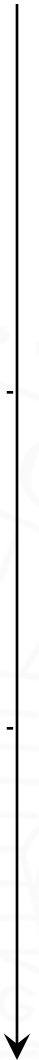
Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

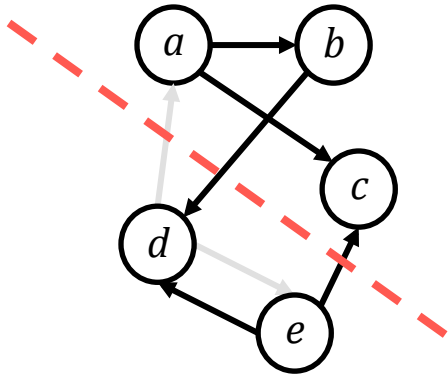
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

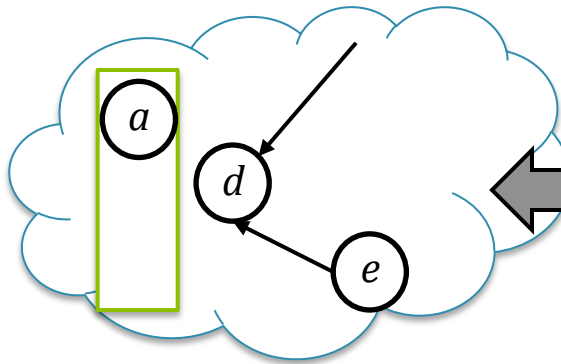
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Delay based Processing



Main Memory



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

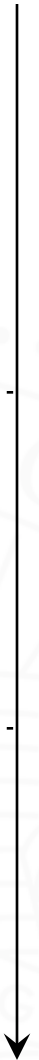
Time

Src	Dst	Value
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

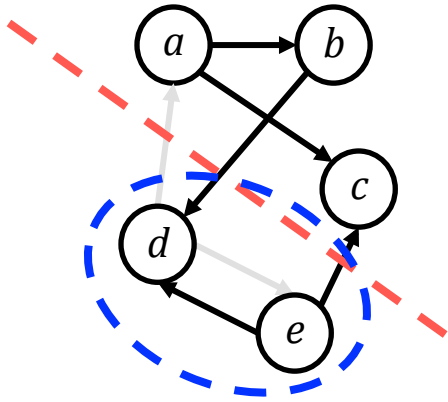
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

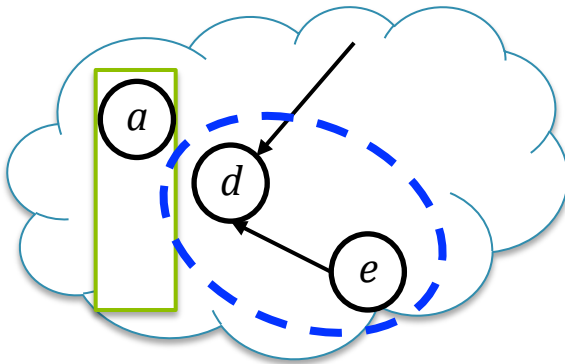
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Delay based Processing



Main
Memory



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

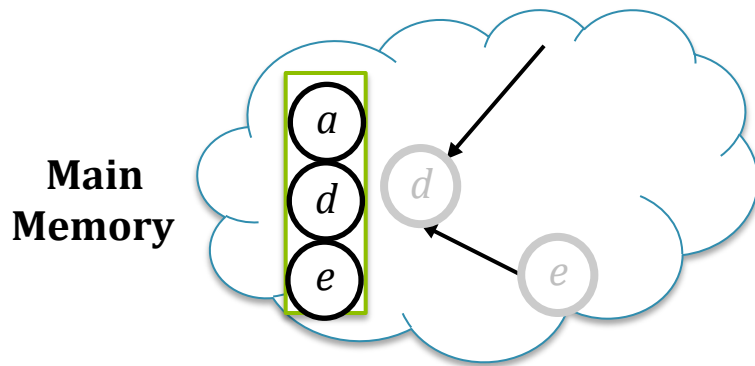
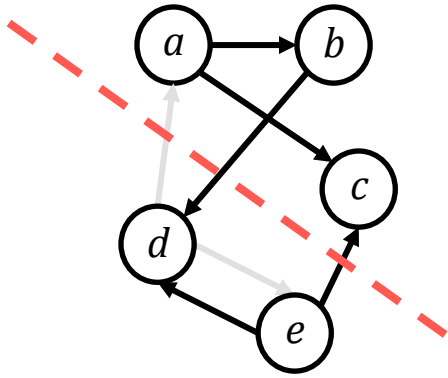
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

Src	Dst	Value
<i>b</i>	<i>d</i>	5



Delay based Processing



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

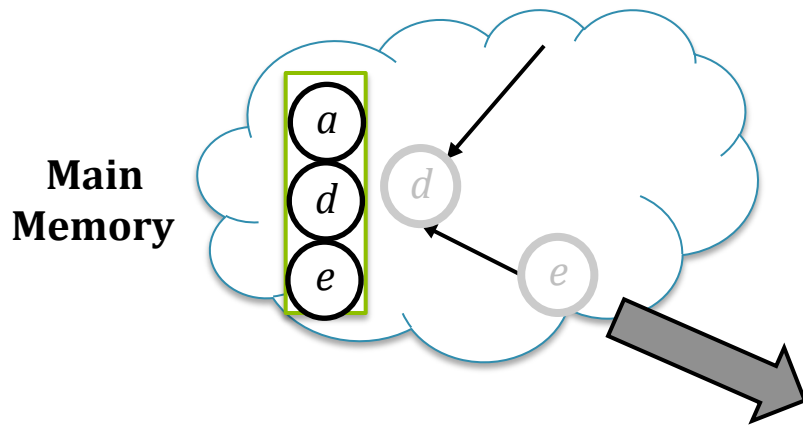
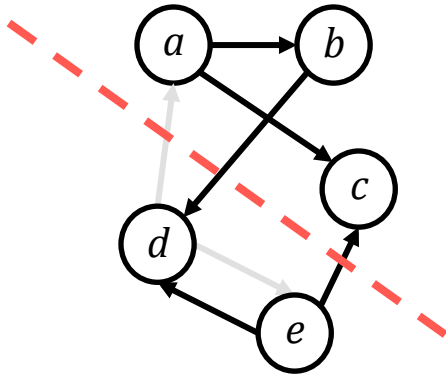
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

Src	Dst	Value
<i>b</i>	<i>d</i>	5



Delay based Processing



Delay computations

Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

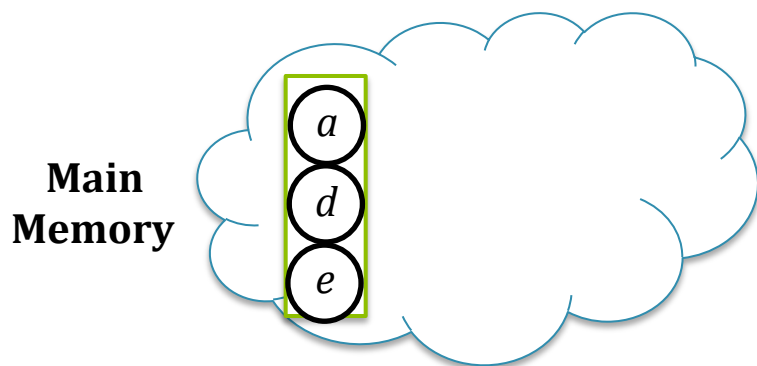
Src	Dst	Value
-----	-----	-------

Src	Dst	Value
<i>b</i>	<i>d</i>	5

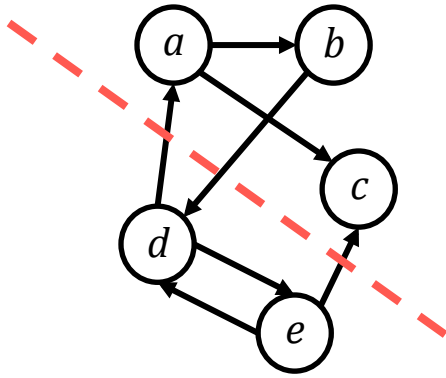


Delay based Processing

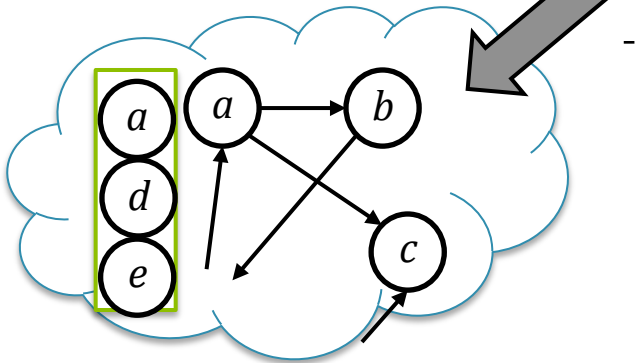
- ▶ Delayed Computations are held in-memory buffer
- ▶ Periodically process delayed vertices
 - ▶ Shadow iteration -- all edges are made available



Shadow Iteration



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

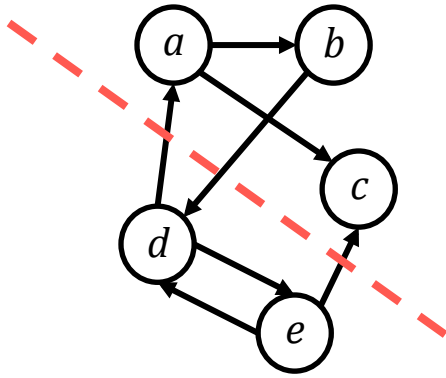
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

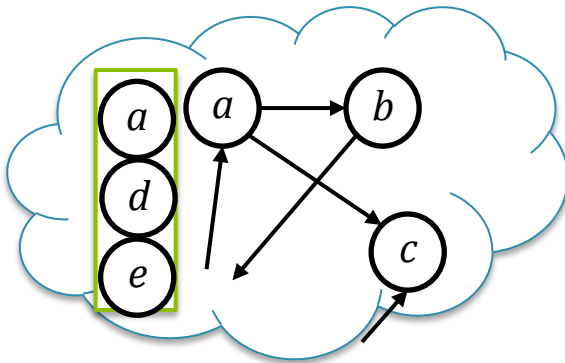
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

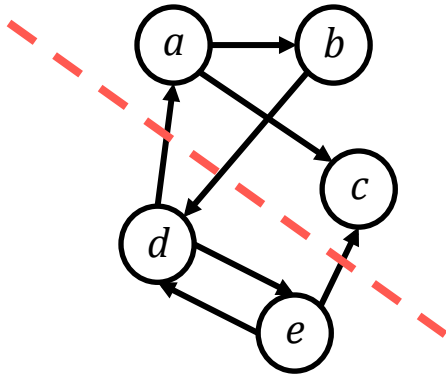
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value

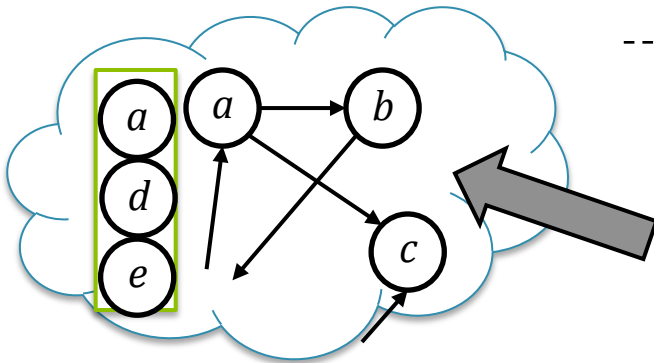
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Shard 0

Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Time

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

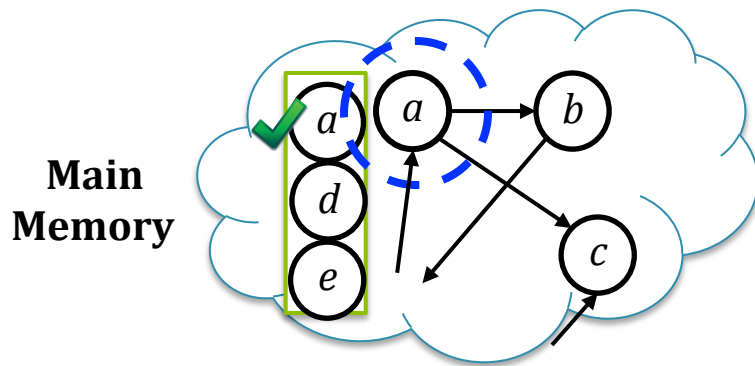
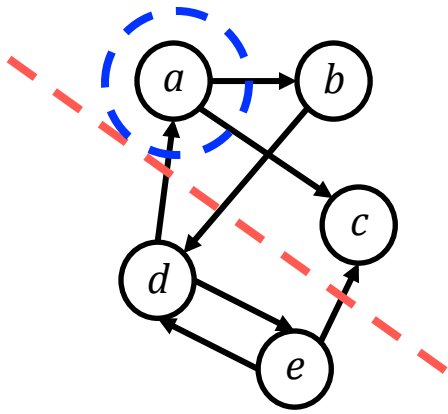
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



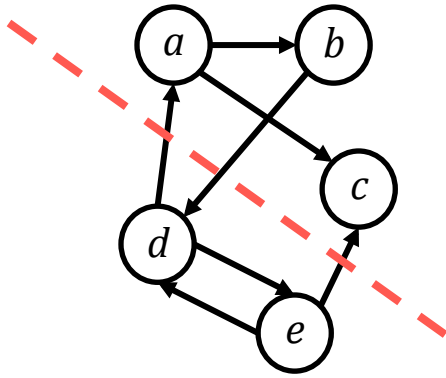
			Shard 1	Time	
Src	Dst	Value	Src	Dst	Value
<i>a</i>	<i>b</i>	5	<i>b</i>	<i>d</i>	6
<i>a</i>	<i>c</i>	4	<i>d</i>	<i>e</i>	4
<i>d</i>	<i>a</i>	7	<i>e</i>	<i>d</i>	1
<i>e</i>	<i>c</i>	9			

Src	Dst	Value	Src	Dst	Value
<i>a</i>	<i>b</i>	4	<i>b</i>	<i>d</i>	3
<i>a</i>	<i>c</i>	6	<i>e</i>	<i>d</i>	6
<i>e</i>	<i>c</i>	8			

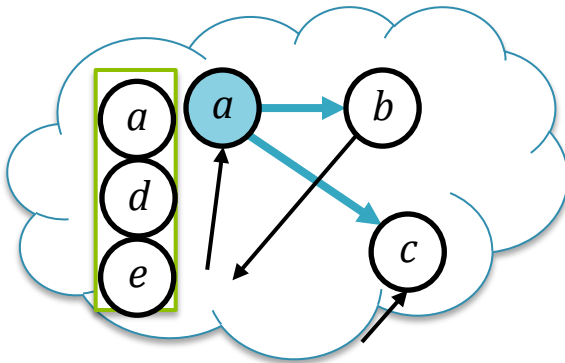
Src	Dst	Value	Src	Dst	Value
			<i>b</i>	<i>d</i>	5

A vertical arrow on the right side of the tables points downwards, indicating the progression of time.

Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1			Time
Src	Dst	Value	
<i>b</i>	<i>d</i>	6	↓
<i>d</i>	<i>e</i>	4	
<i>e</i>	<i>d</i>	1	

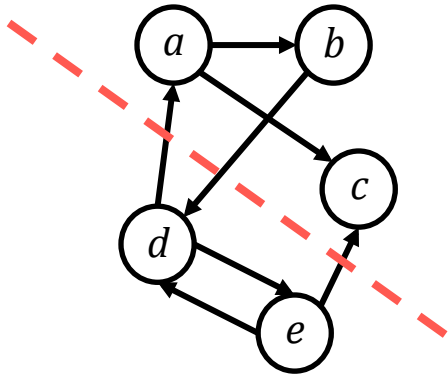
Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

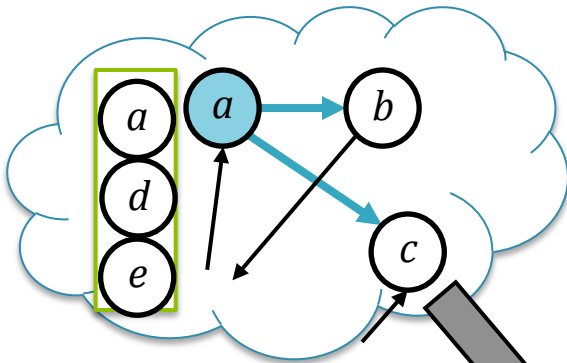
Src	Dst	Value

Src	Dst	Value
<i>b</i>	<i>d</i>	5

Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
-----	-----	-------

Shard 1

Time

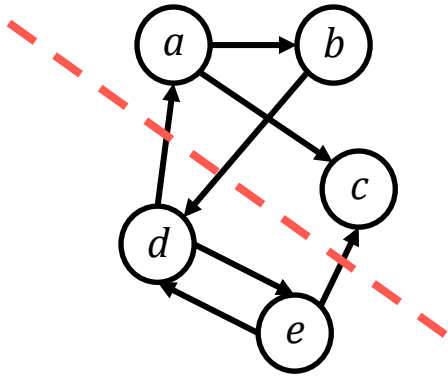
Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

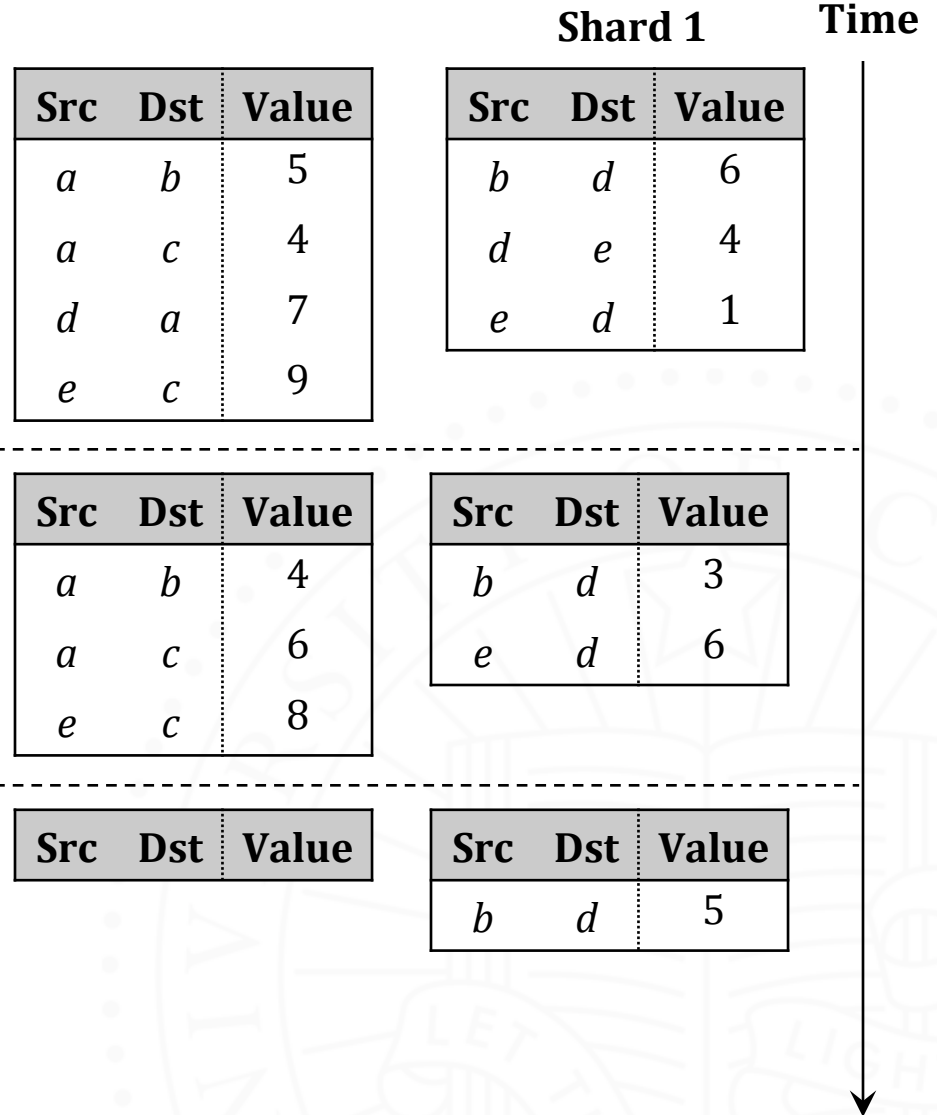
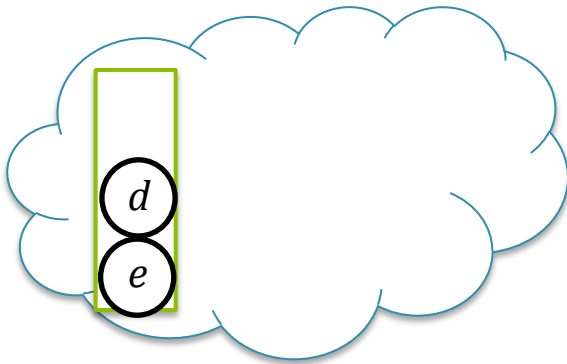
Src	Dst	Value
<i>b</i>	<i>d</i>	5



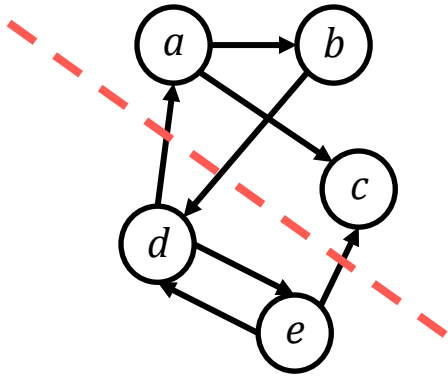
Shadow Iteration



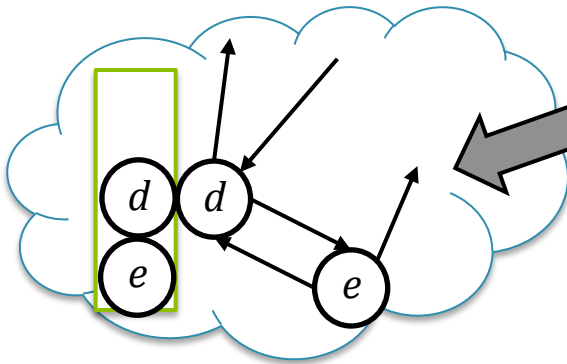
Main
Memory



Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	8

Shard 1

Time

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

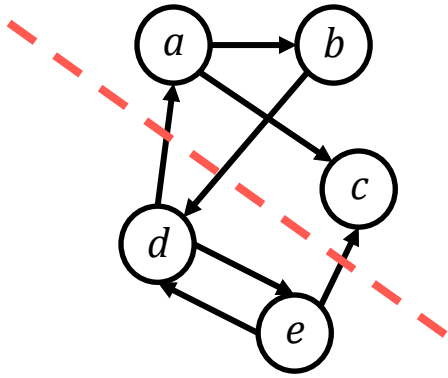
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

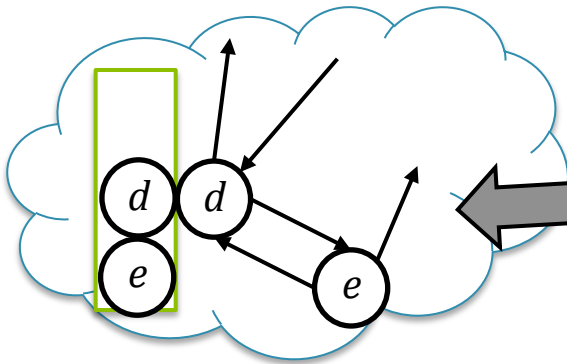
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value

Shard 1

Time

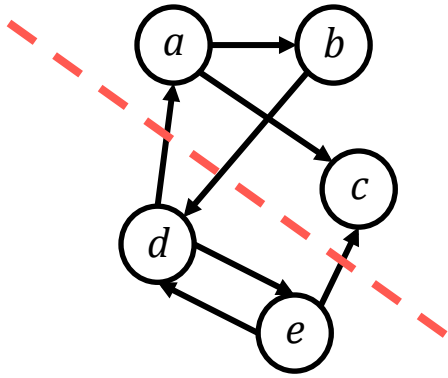
Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

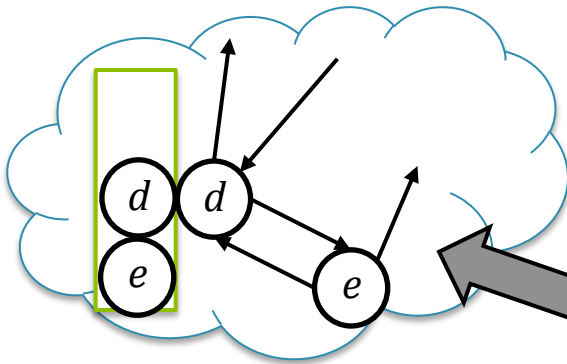
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Time

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

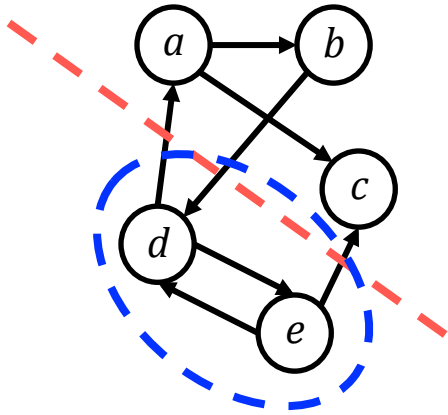
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
<i>a</i>	<i>b</i>	5

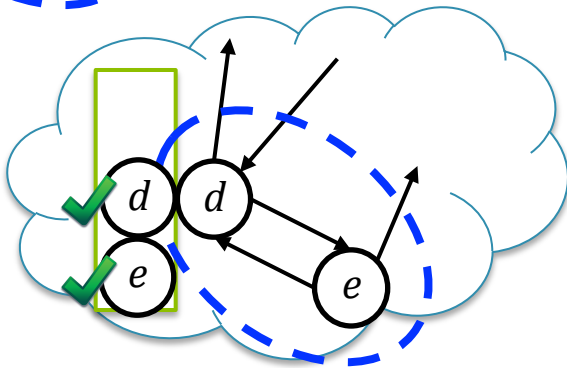
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1			Time
Src	Dst	Value	↓
<i>b</i>	<i>d</i>	6	
<i>d</i>	<i>e</i>	4	
<i>e</i>	<i>d</i>	1	

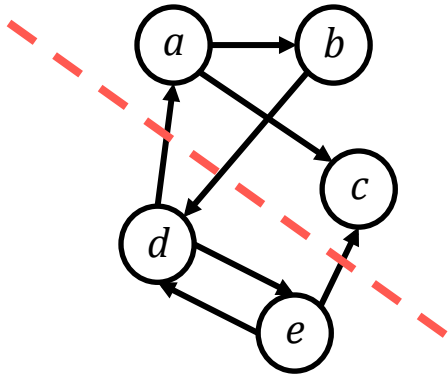
Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

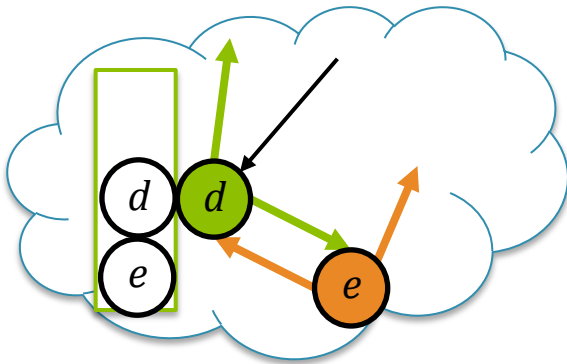
Src	Dst	Value

Src	Dst	Value
<i>b</i>	<i>d</i>	5

Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1

Time

Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

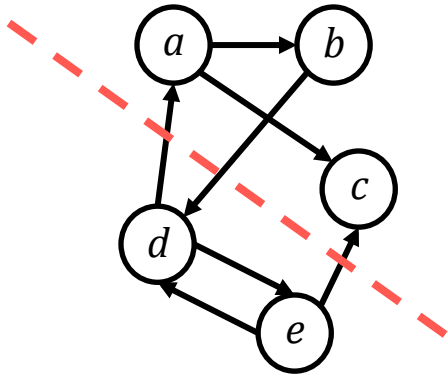
Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

Src	Dst	Value
-----	-----	-------

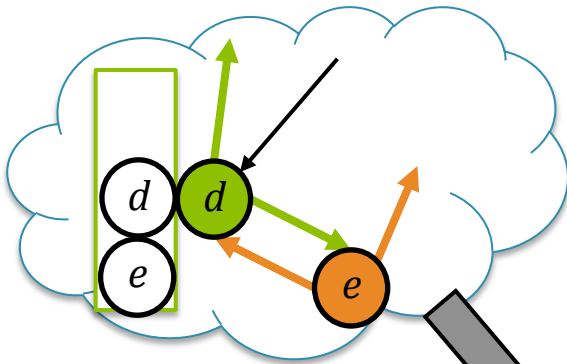
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value

Shard 1

Time

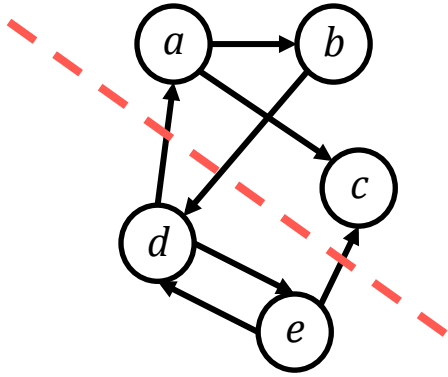
Src	Dst	Value
<i>b</i>	<i>d</i>	6
<i>d</i>	<i>e</i>	4
<i>e</i>	<i>d</i>	1

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

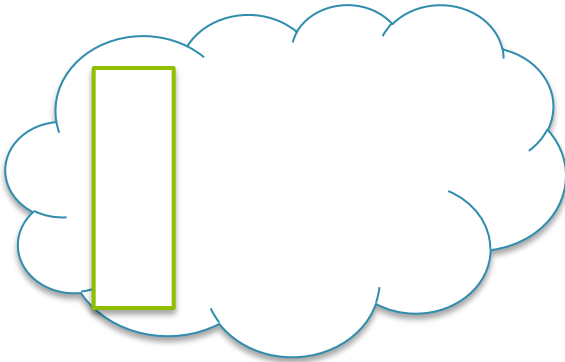
Src	Dst	Value
<i>b</i>	<i>d</i>	5



Shadow Iteration



Main
Memory



Src	Dst	Value
<i>a</i>	<i>b</i>	5
<i>a</i>	<i>c</i>	4
<i>d</i>	<i>a</i>	7
<i>e</i>	<i>c</i>	9

Shard 1			Time
Src	Dst	Value	
<i>b</i>	<i>d</i>	6	↓
<i>d</i>	<i>e</i>	4	
<i>e</i>	<i>d</i>	1	

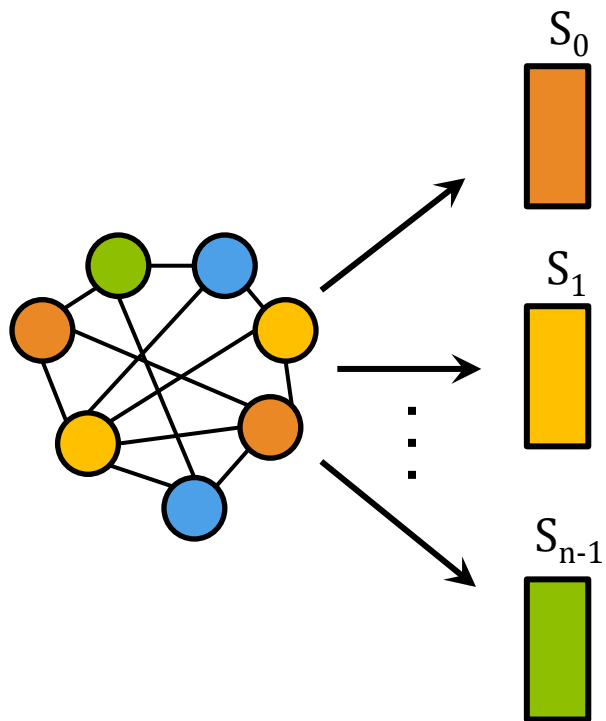
Src	Dst	Value
<i>a</i>	<i>b</i>	4
<i>a</i>	<i>c</i>	6
<i>e</i>	<i>c</i>	8

Src	Dst	Value
<i>b</i>	<i>d</i>	3
<i>e</i>	<i>d</i>	6

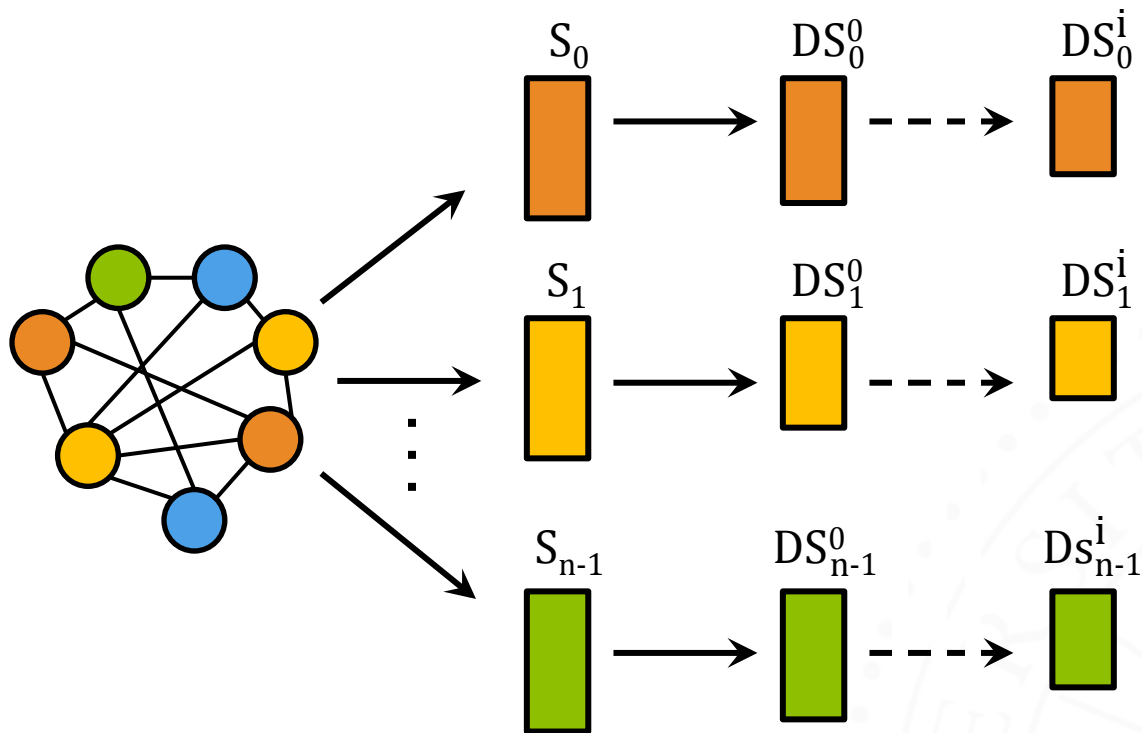
Src	Dst	Value

Src	Dst	Value
<i>b</i>	<i>d</i>	5

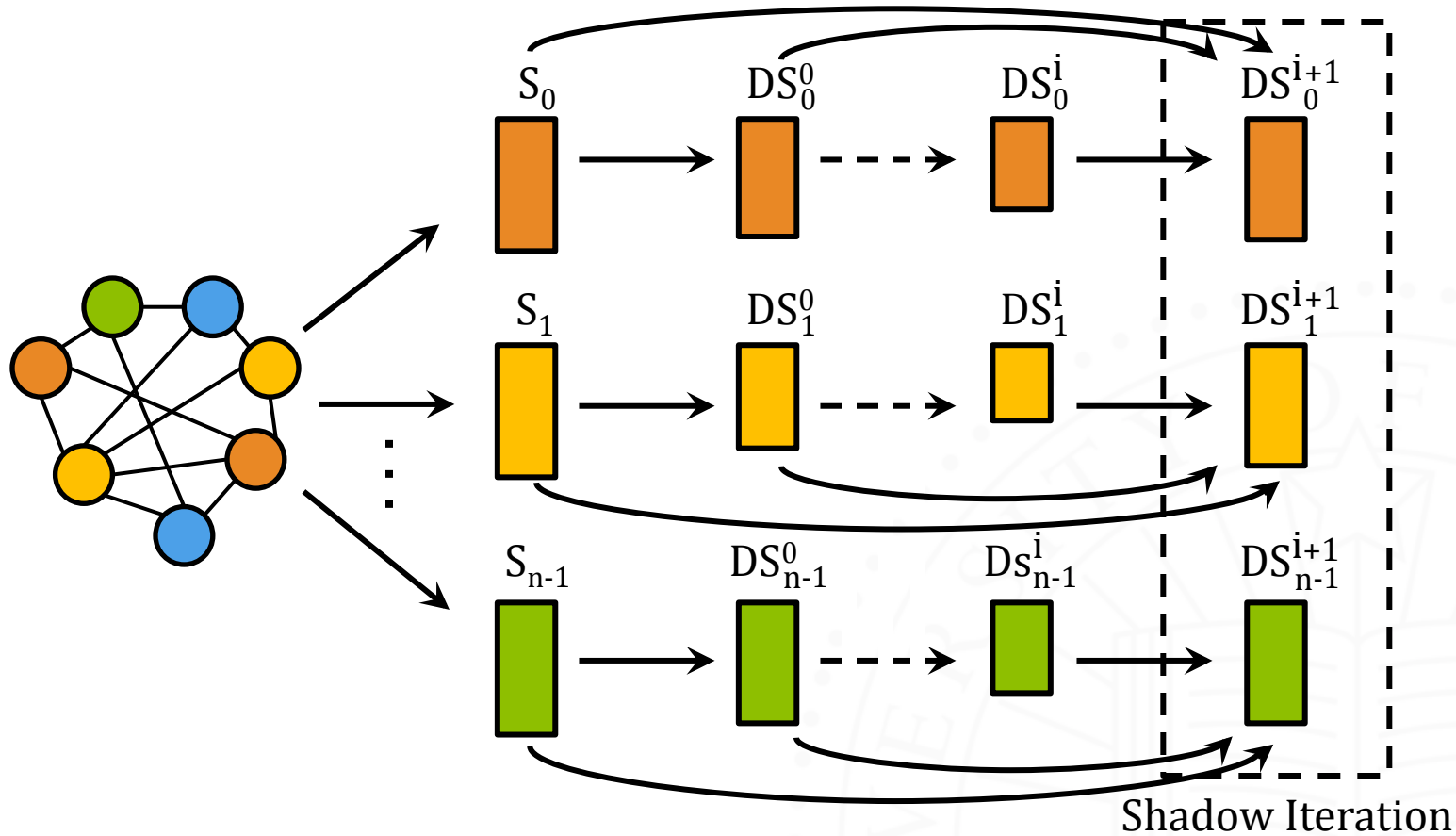
Dynamic Shards



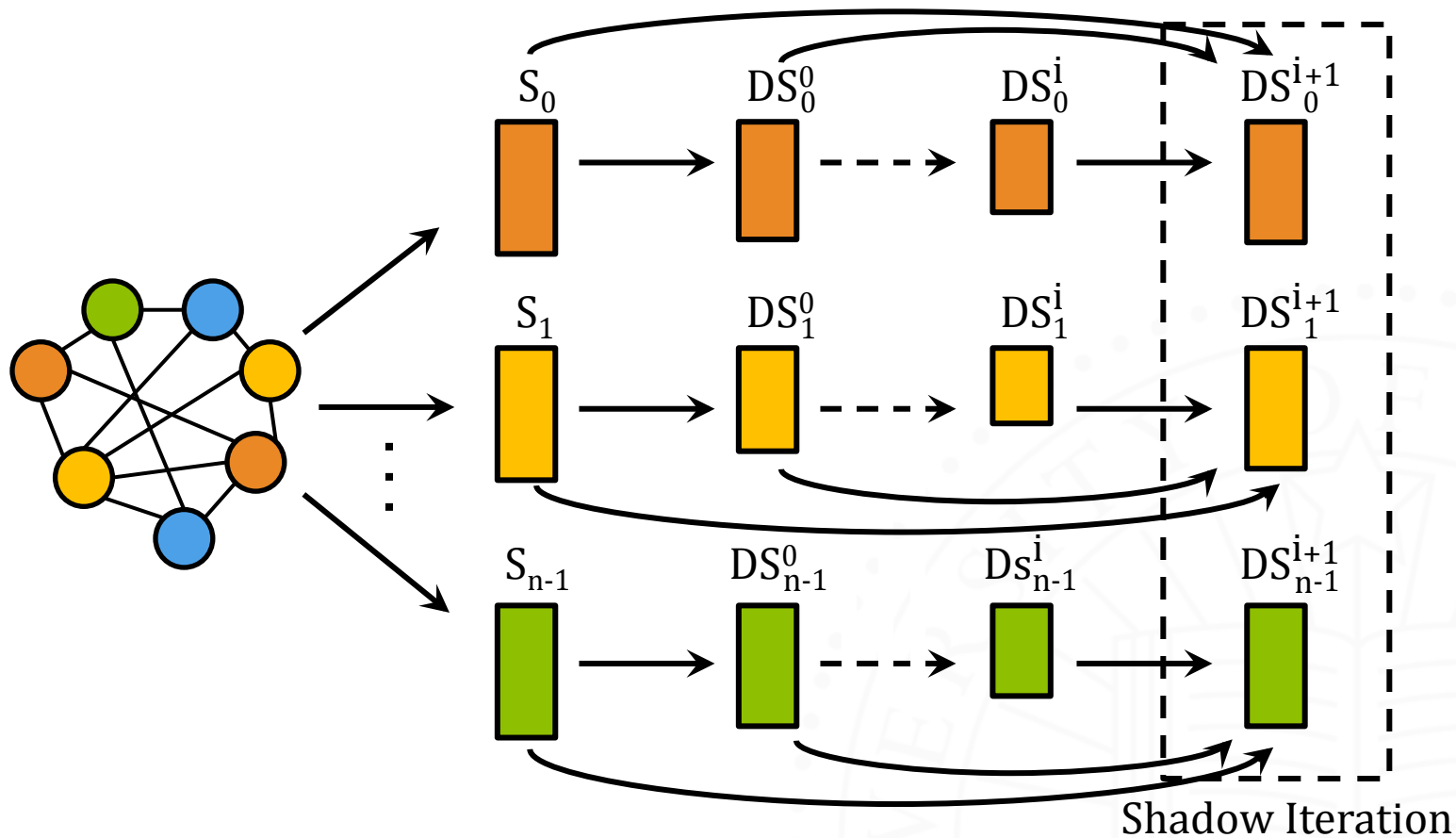
Dynamic Shards



Dynamic Shards

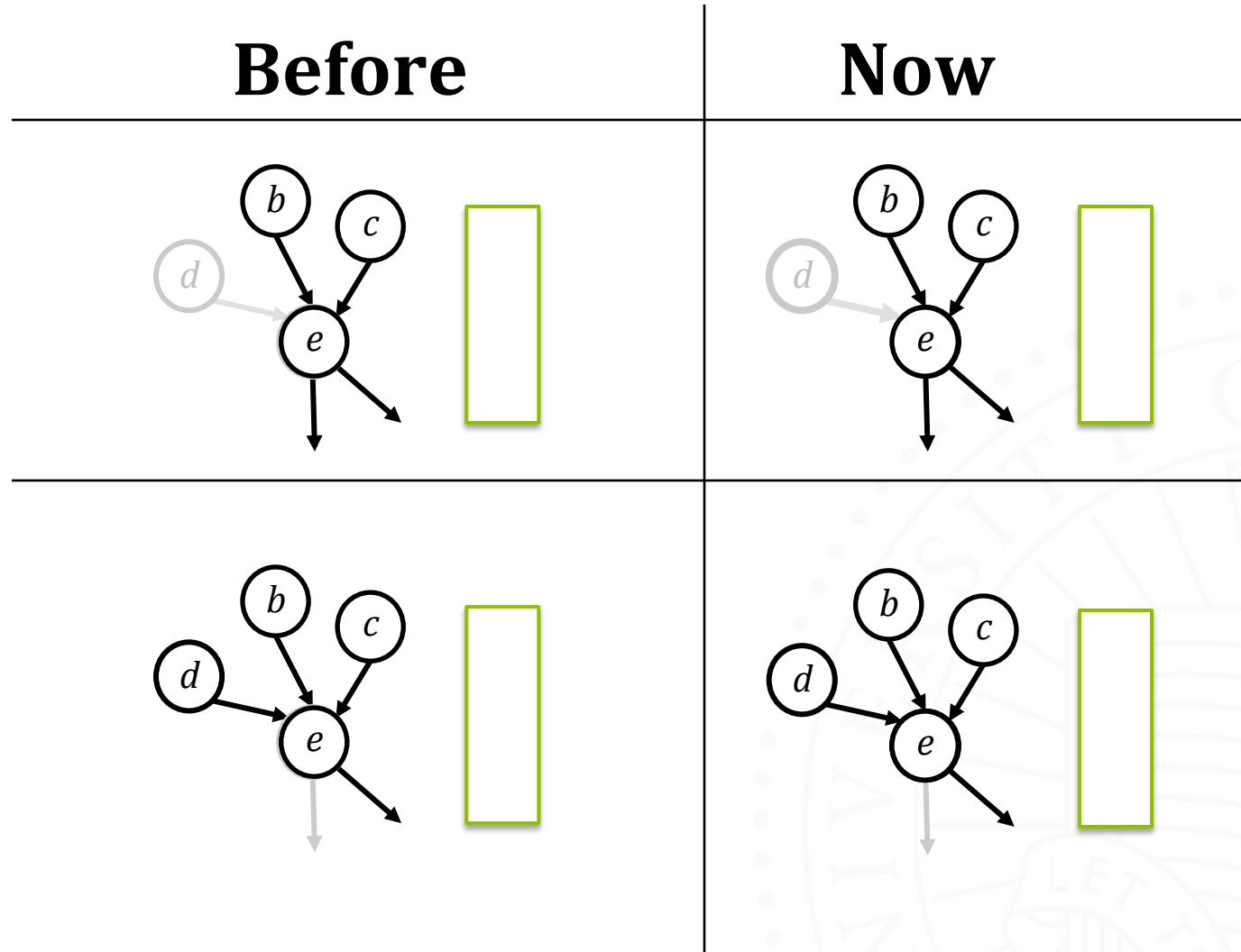


Dynamic Shards

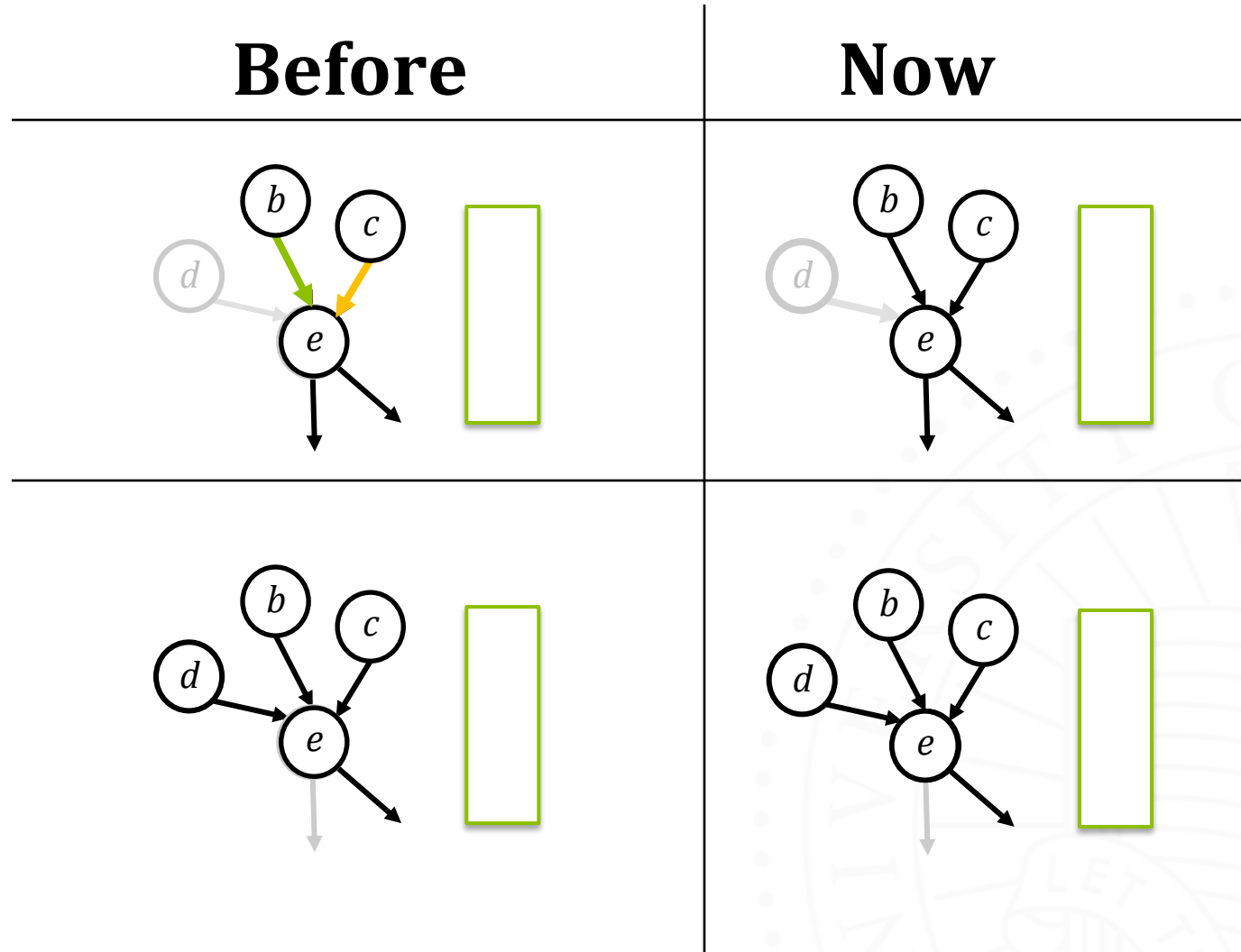


64-73% of active vertices get delayed

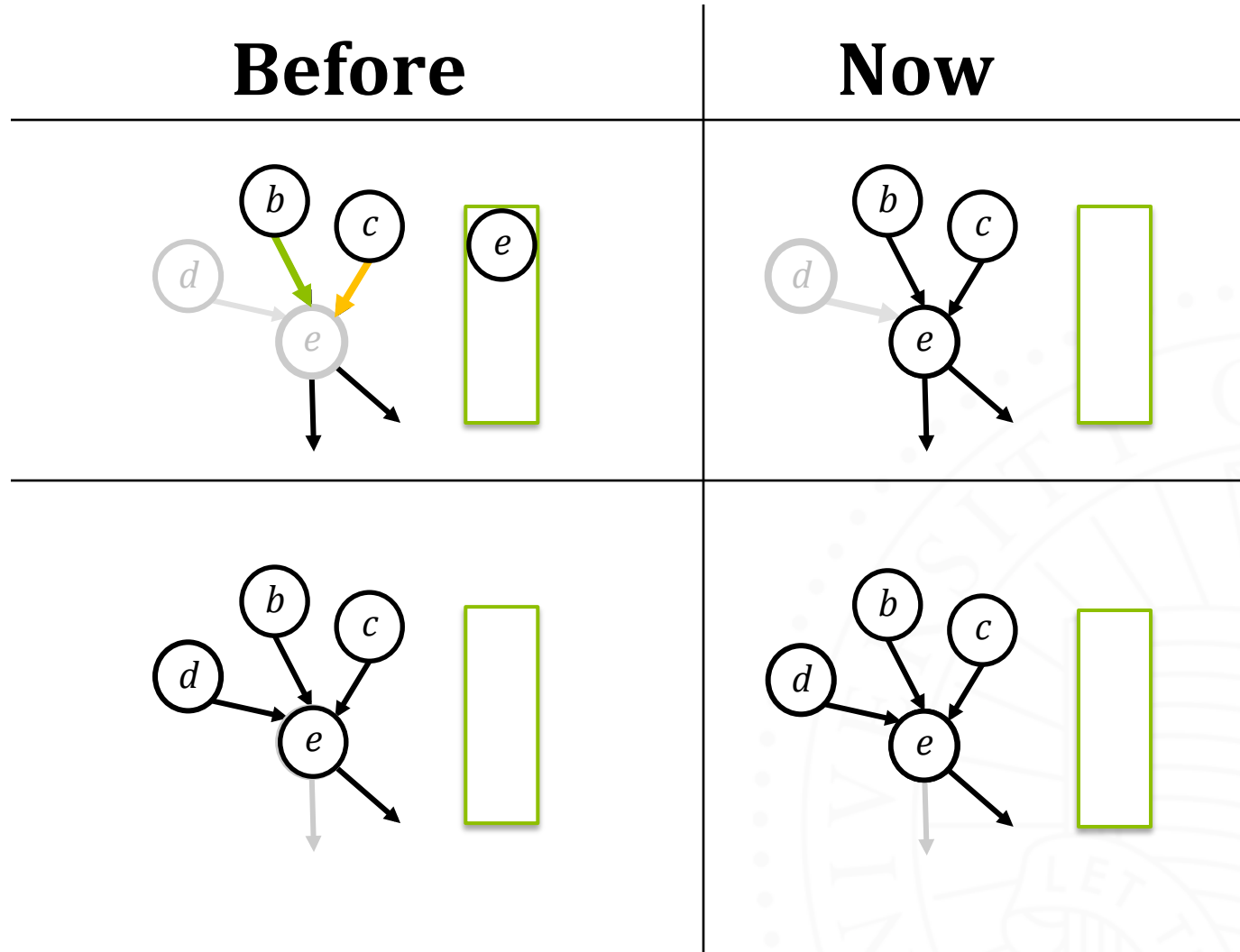
Accumulation based Processing



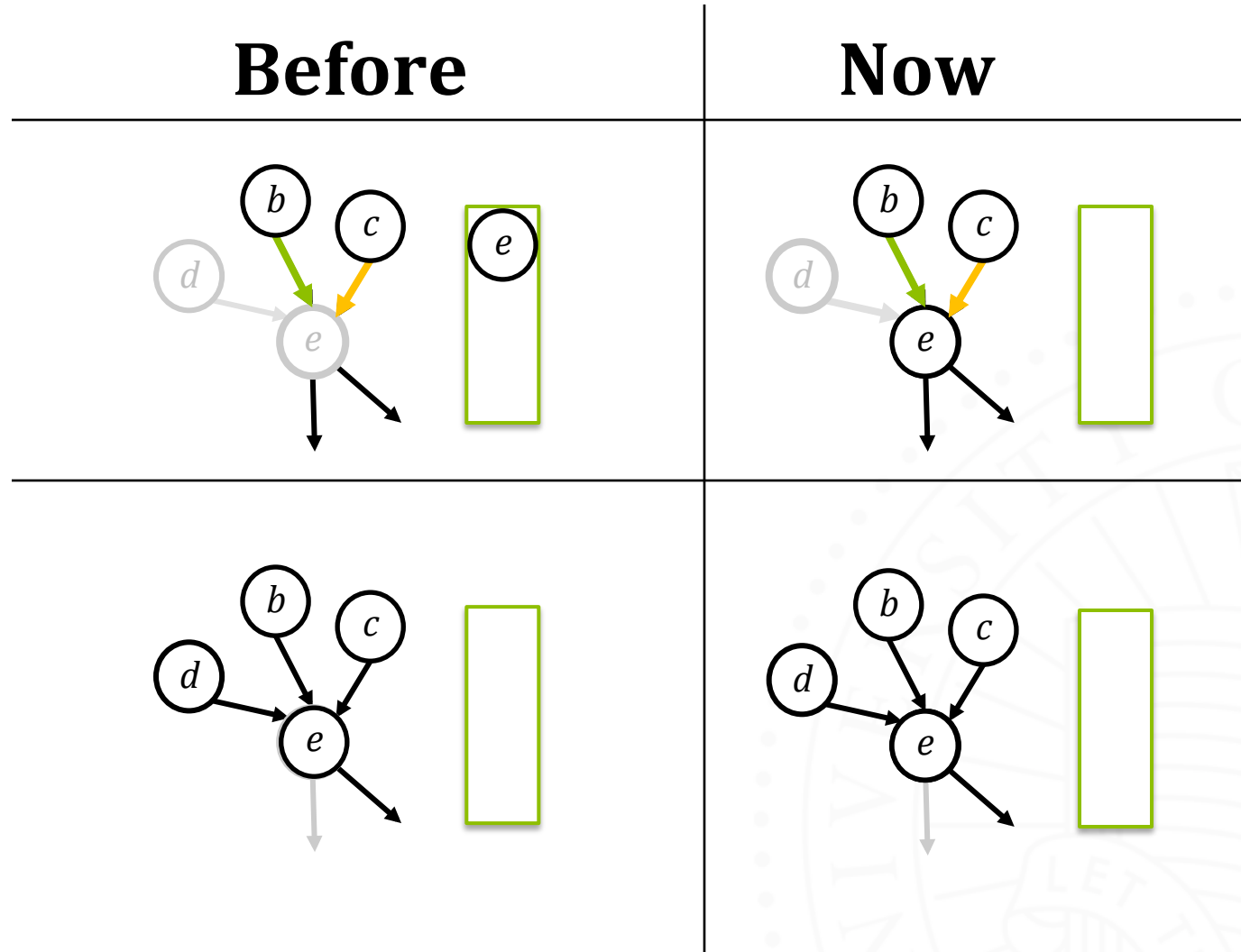
Accumulation based Processing



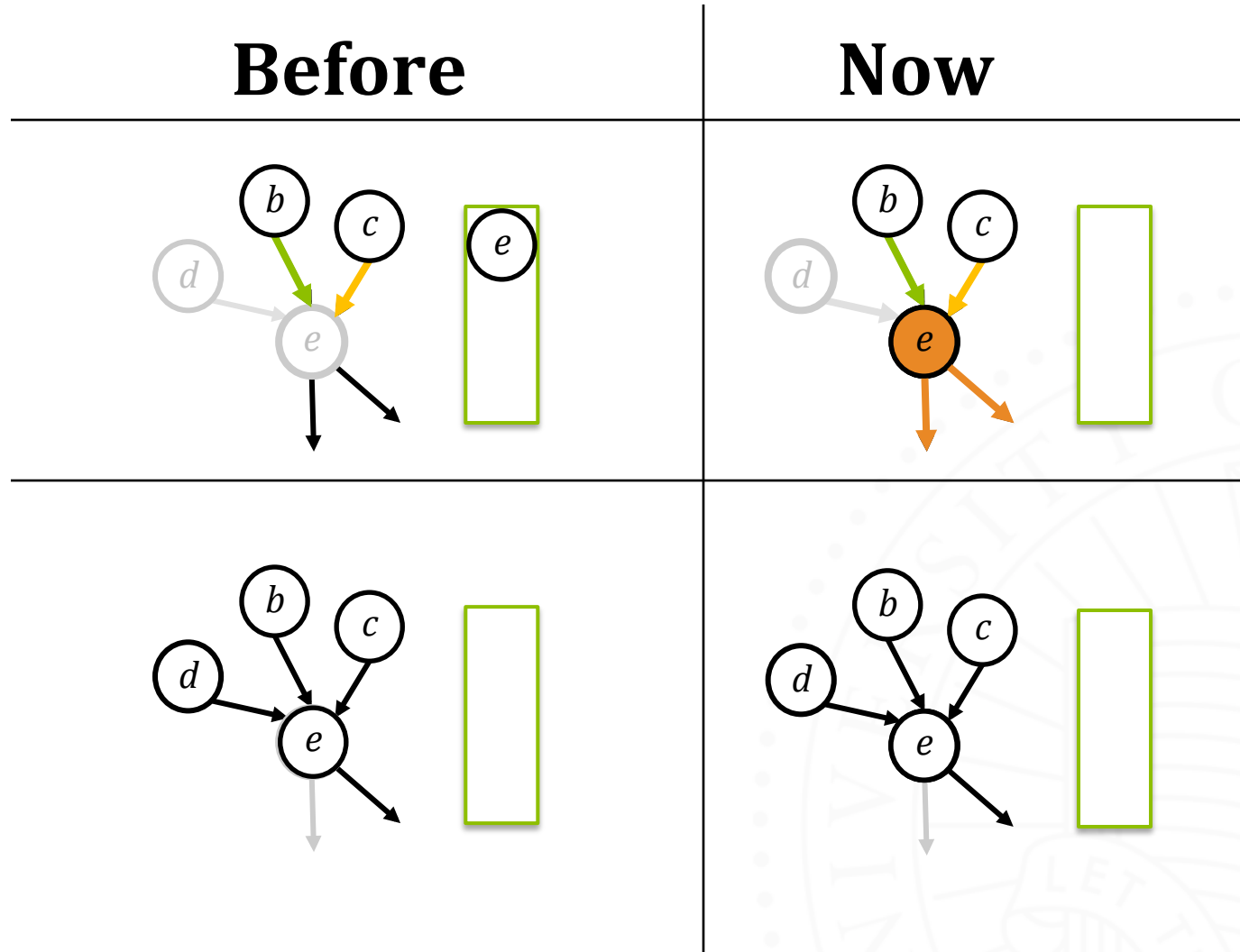
Accumulation based Processing



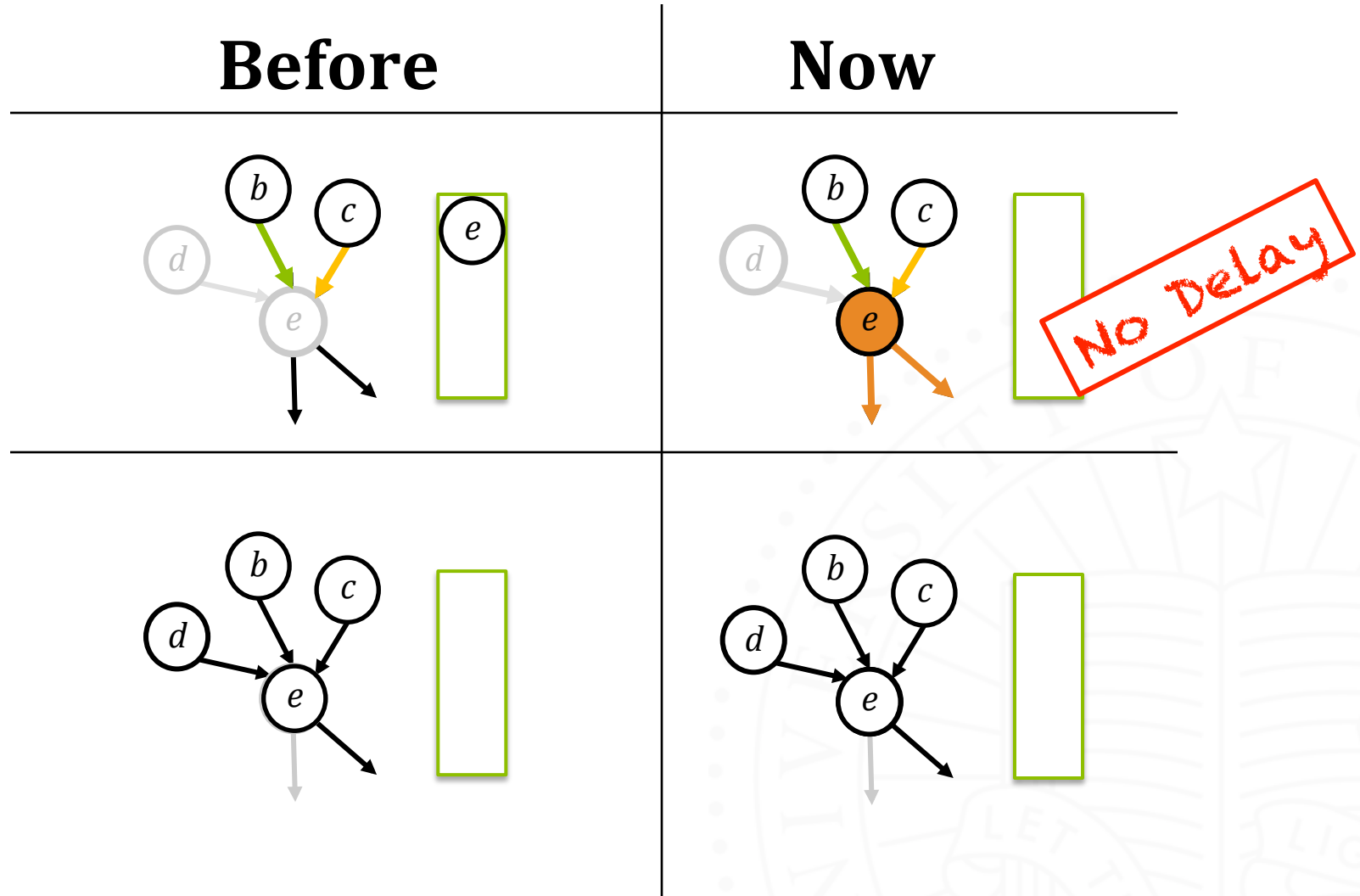
Accumulation based Processing



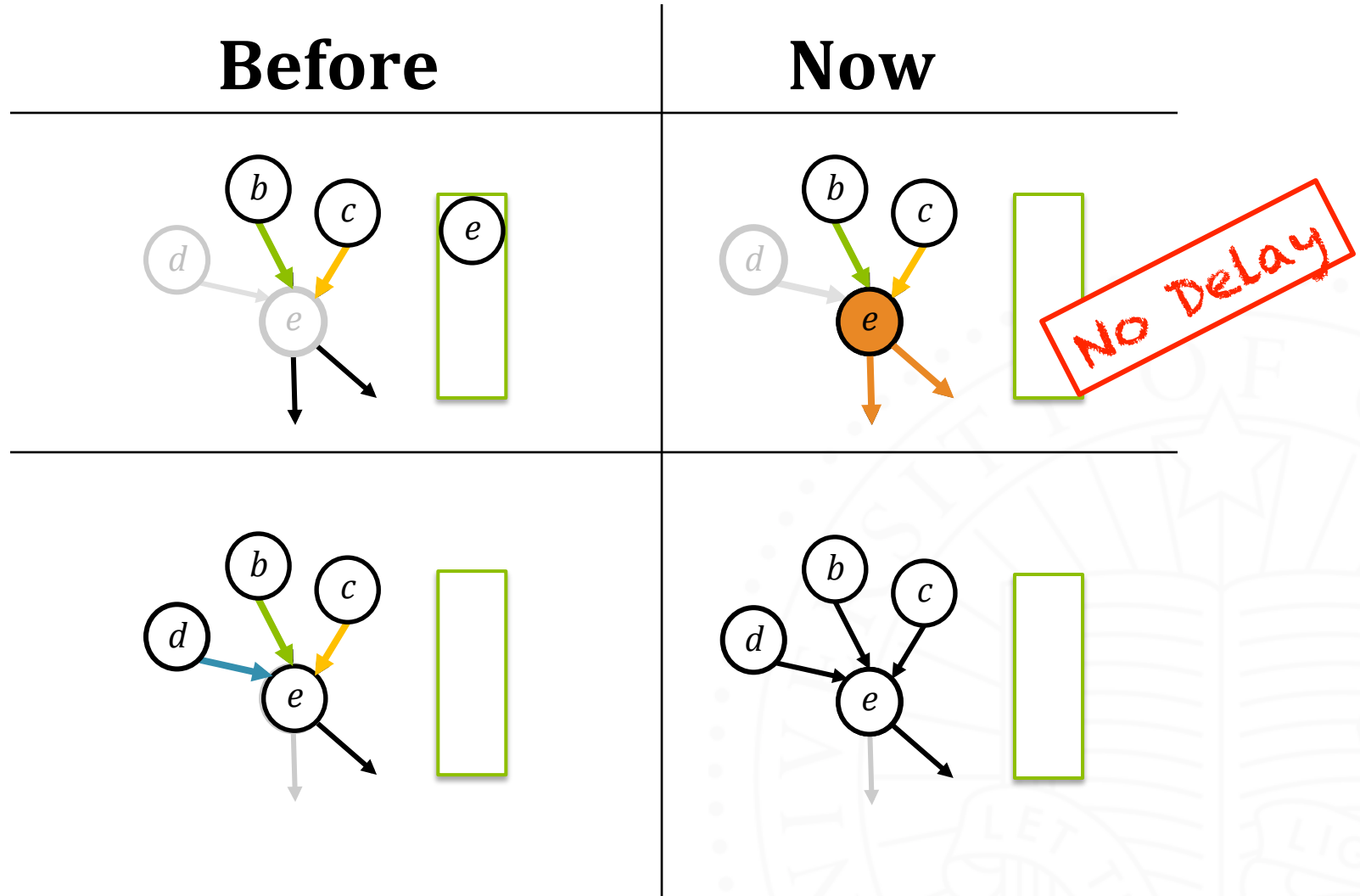
Accumulation based Processing



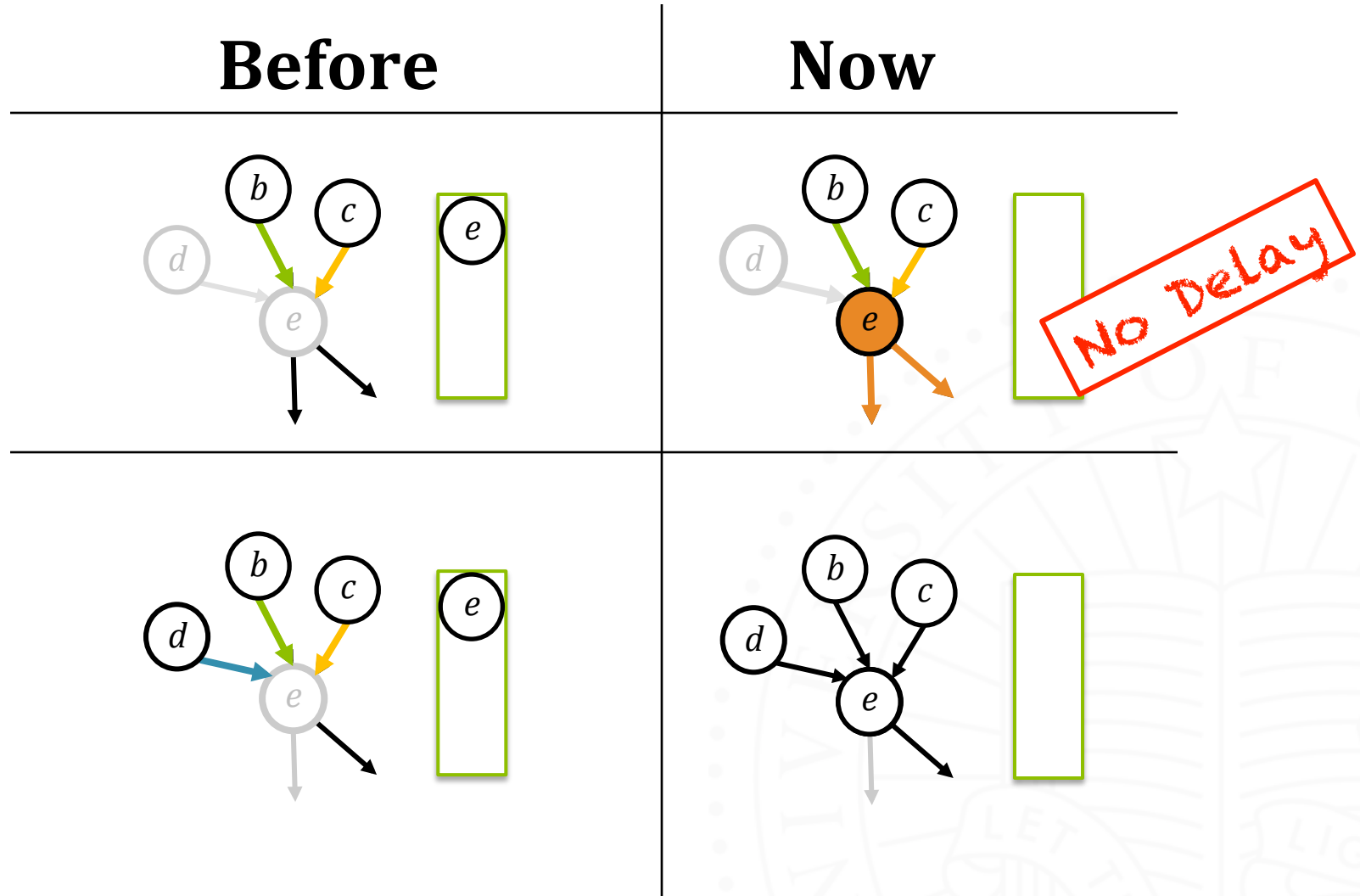
Accumulation based Processing



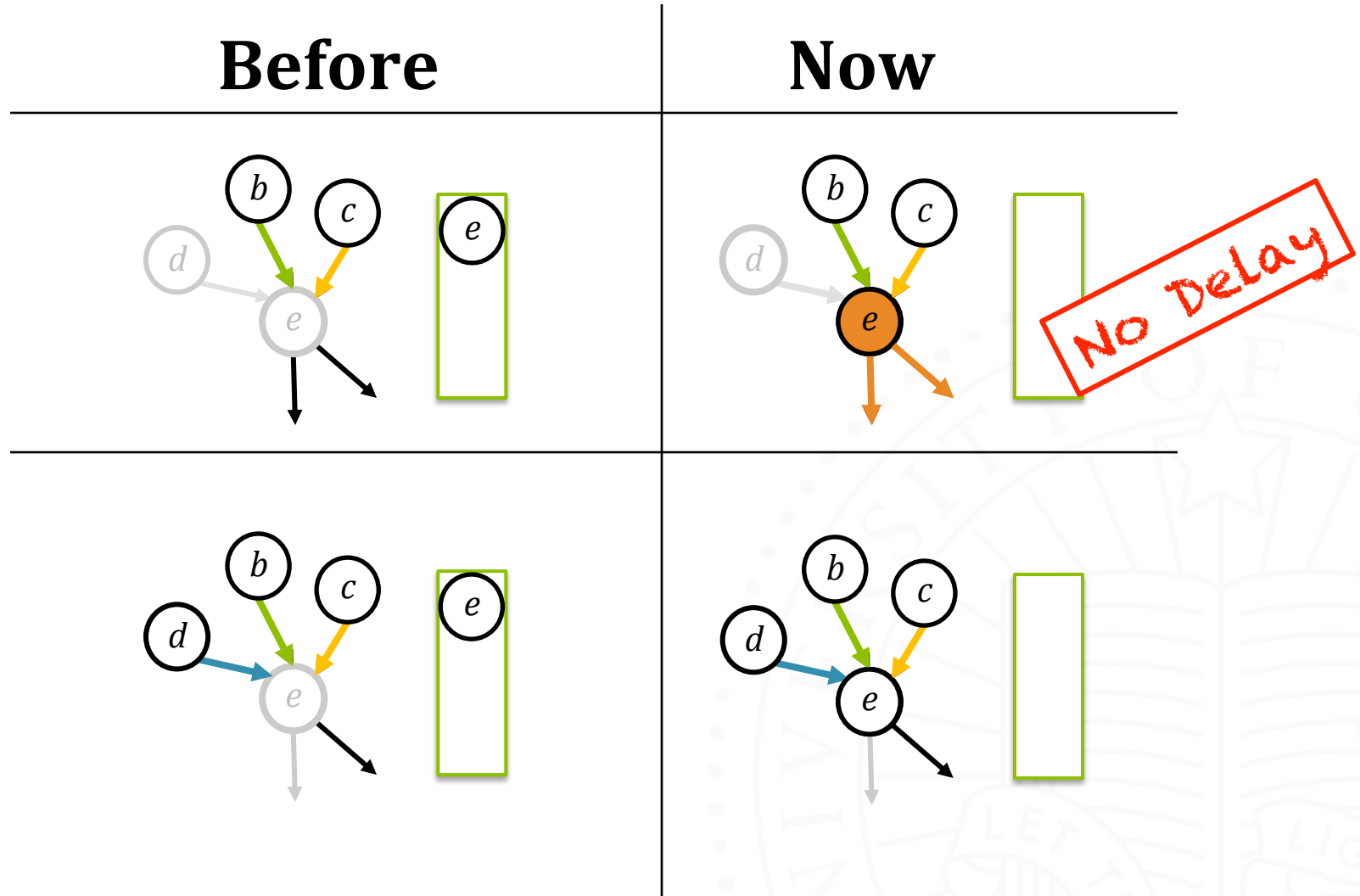
Accumulation based Processing



Accumulation based Processing

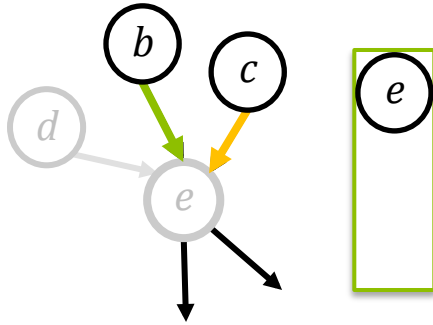


Accumulation based Processing

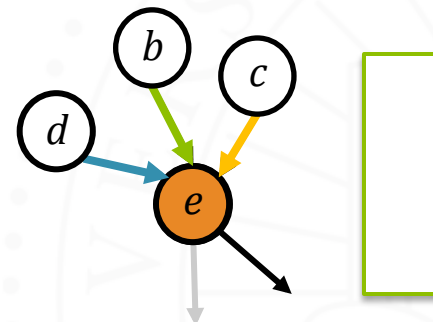
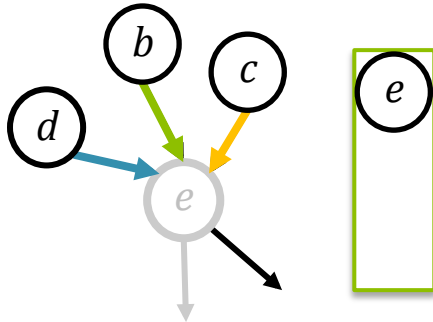
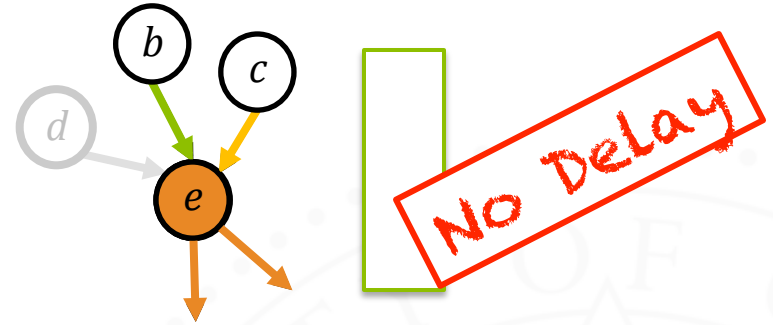


Accumulation based Processing

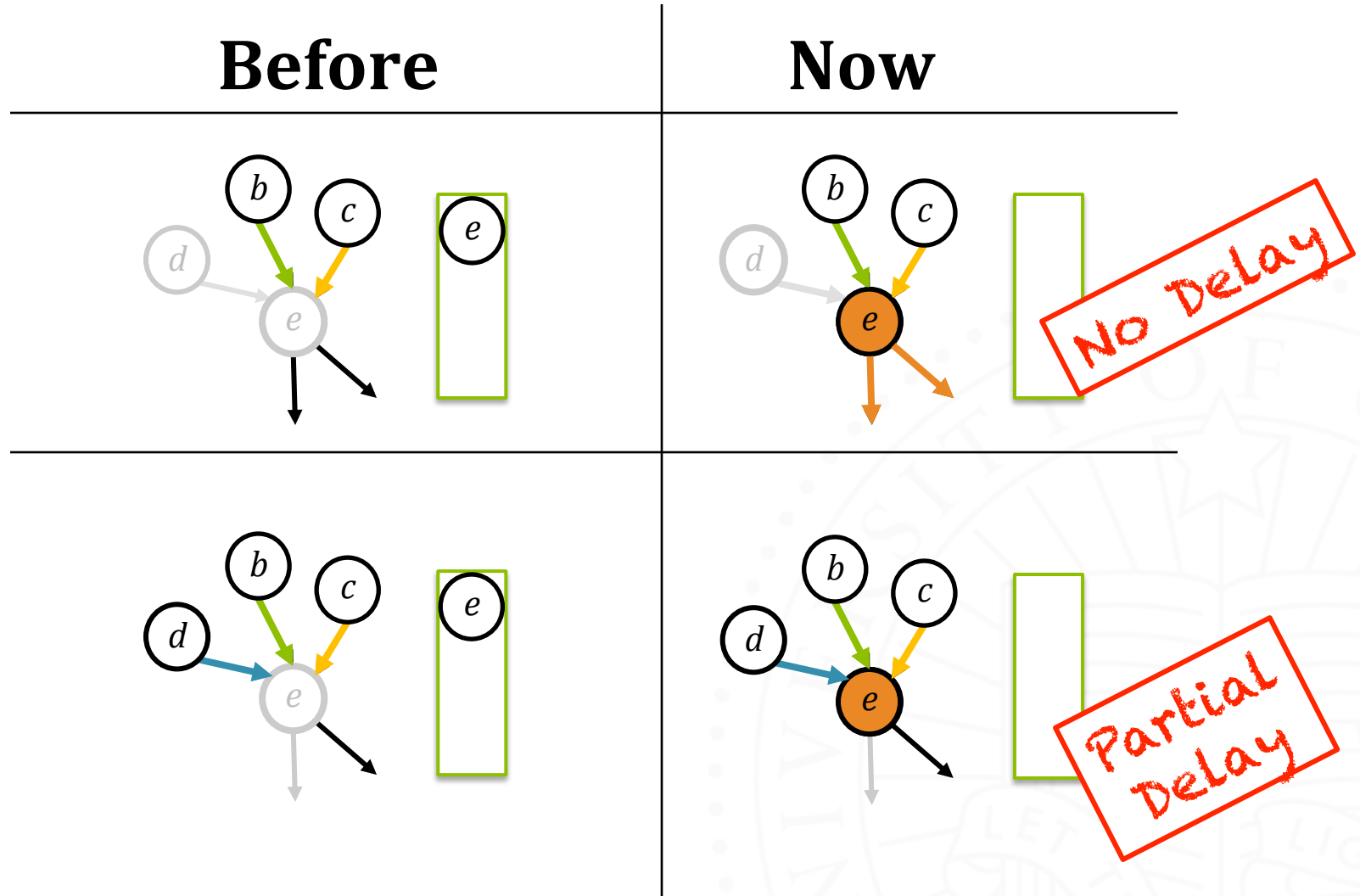
Before



Now



Accumulation based Processing

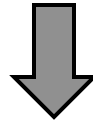


Accumulation based Processing

[GATHER] $sum \leftarrow \sum_{e \in in(v)} e.data$

[APPLY] $v.pr \leftarrow (0.15 + 0.85 * sum) /$
 $v.numOutEdges$

[SCATTER] $\forall e \in out(v) : e.data \leftarrow v.pr$



[EXTRACT] $oldsum \leftarrow (v.pr * v.numOutEdges$
 $- 0.15) / 0.85$

[GATHER] $newsum \leftarrow oldsum + \sum_{e \in in(v)} e.data$

[APPLY] $newpr \leftarrow (0.15 + 0.85 * newsum) /$
 $v.numOutEdges;$

$oldpr \leftarrow v.pr; v.pr \leftarrow newpr$

[SCATTER] $\forall e \in out(v) : e.data \leftarrow newpr - oldpr$

Evaluation Setup

- › Dell 8-core, 8 GB main memory
- › Dell 500GB 7.2K RPM HDD, Dell 400GB SSD
- › Ubuntu 14.04
- › File system caches flushed



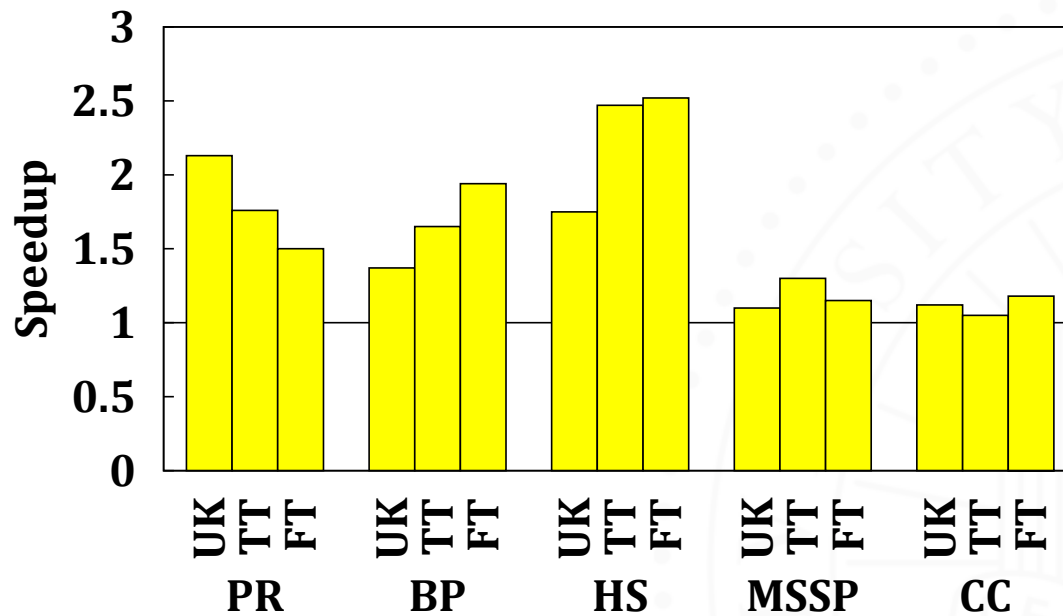
Benchmark & Inputs

- ▶ 5 graph algorithms
 - ▶ PageRank, Belief Propagation, Heat Simulation, Multiple Source Shortest Paths, Connected Components
- ▶ 6 real-world input graphs

Inputs	#Vertices	#Edges	SizeToMem
LiveJournal (LJ)	4.8M	69M	0.2×
Netflix (NF)	0.5M	99M	0.2×
UKDoman (UK)	39.5M	1.0B	2.4×
Twitter (TT)	41.7M	1.5B	5.2×
Friendster (FT)	68.3M	2.6B	10.2×
YahooWeb (YW)	1.4B	6.6B	21.6×

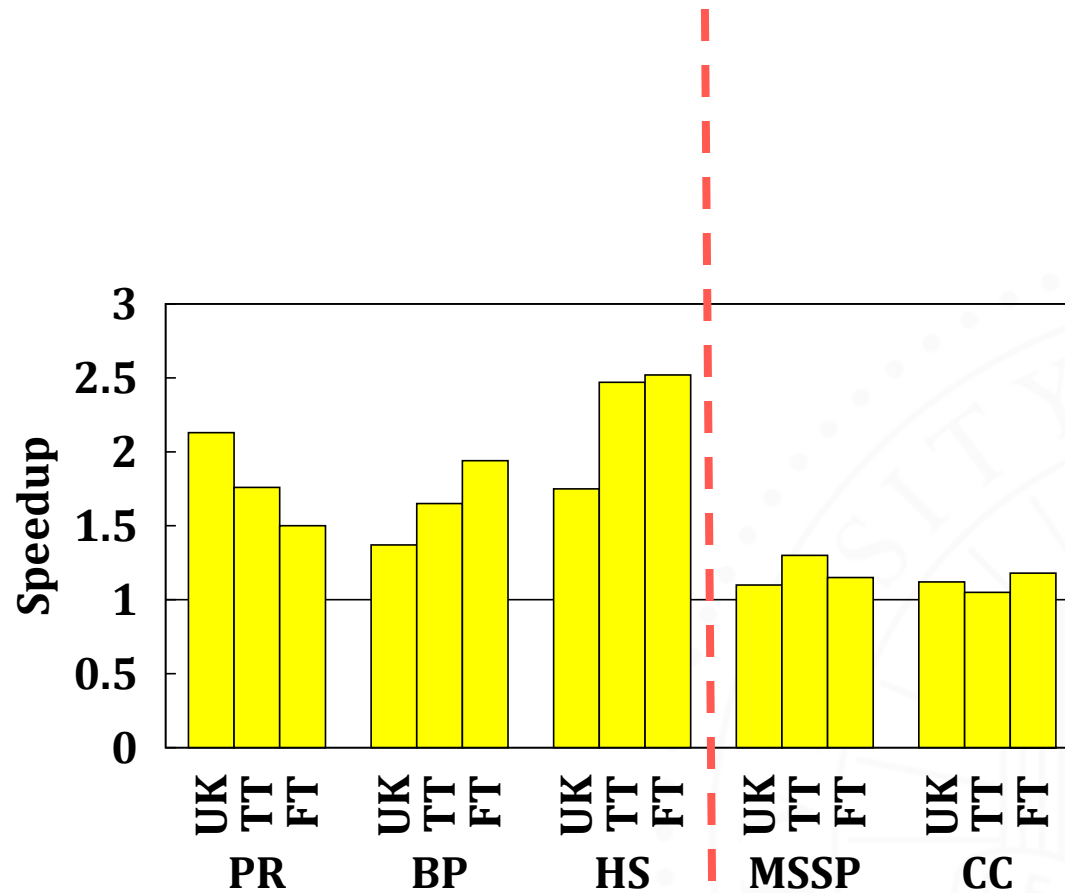
Performance

- ▶ 1.8x average speedup

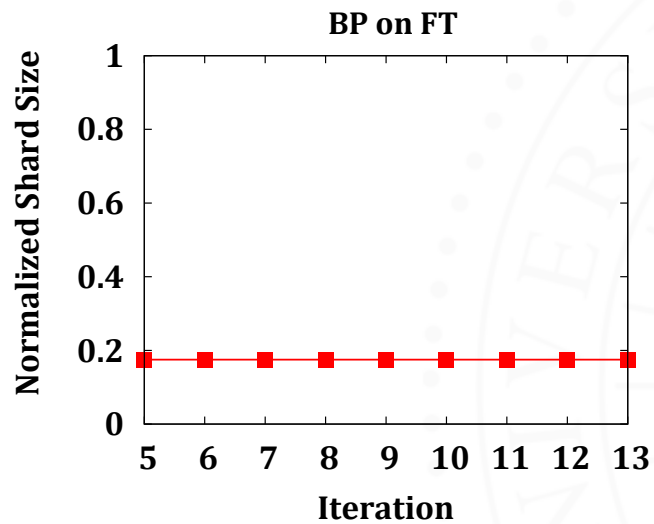
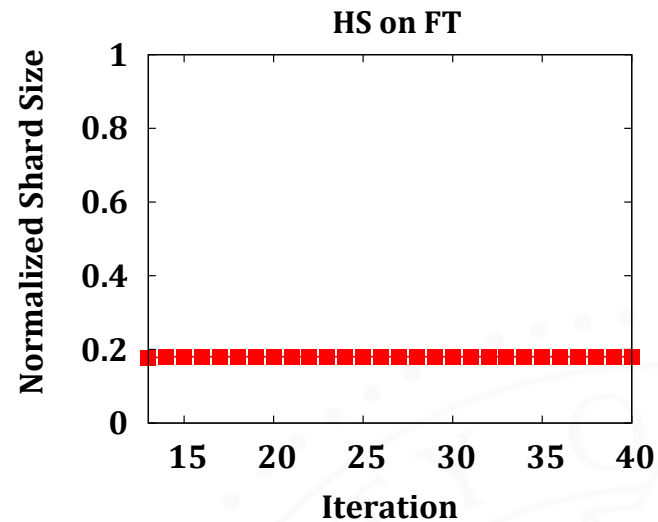
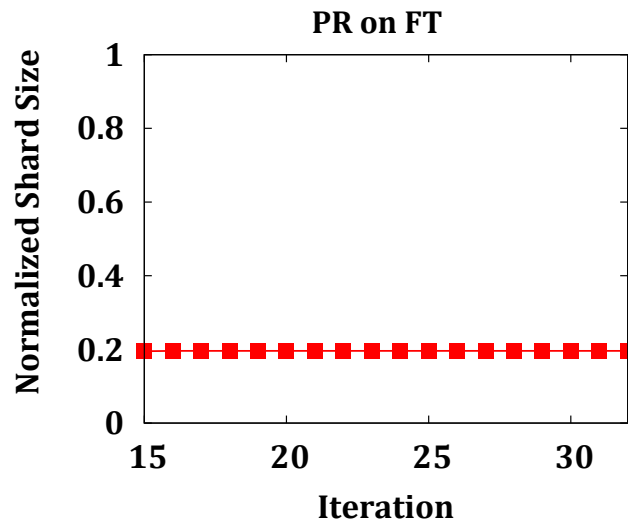


Performance

- ▶ 1.8x average speedup

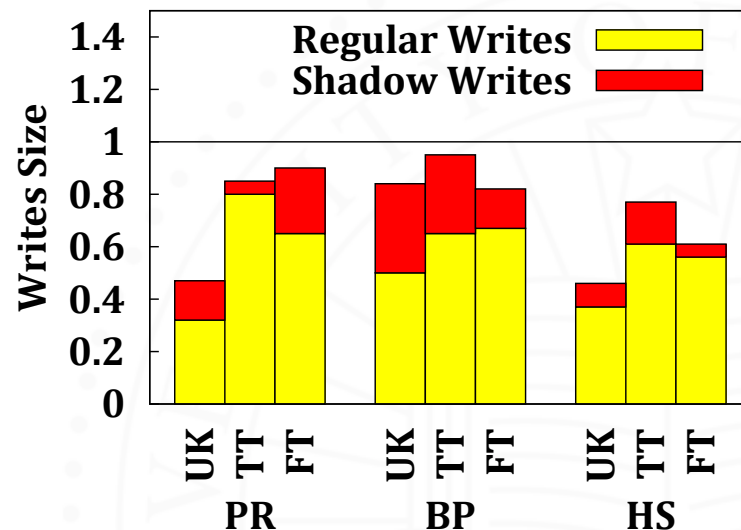
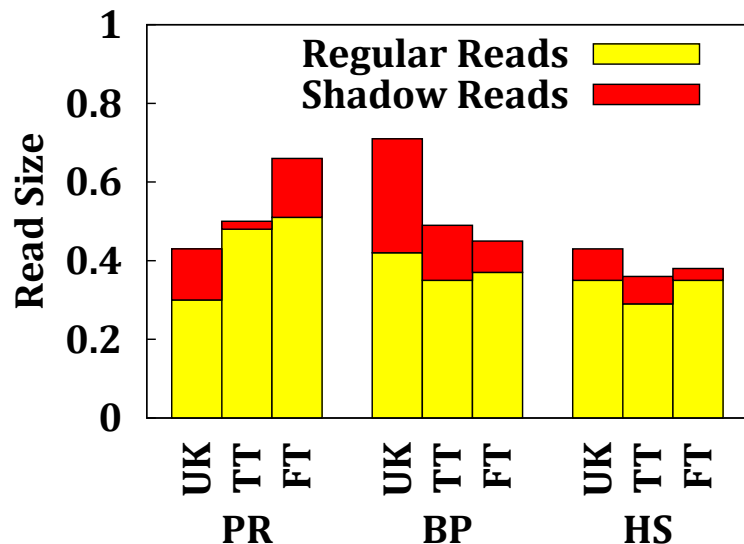


Dynamic Shard Size

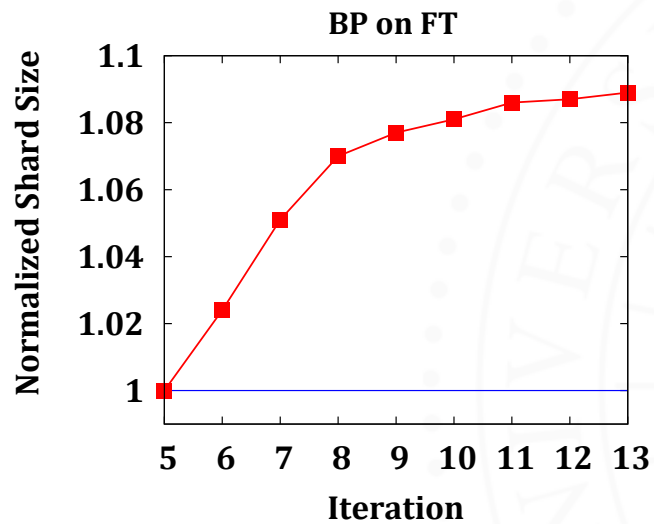
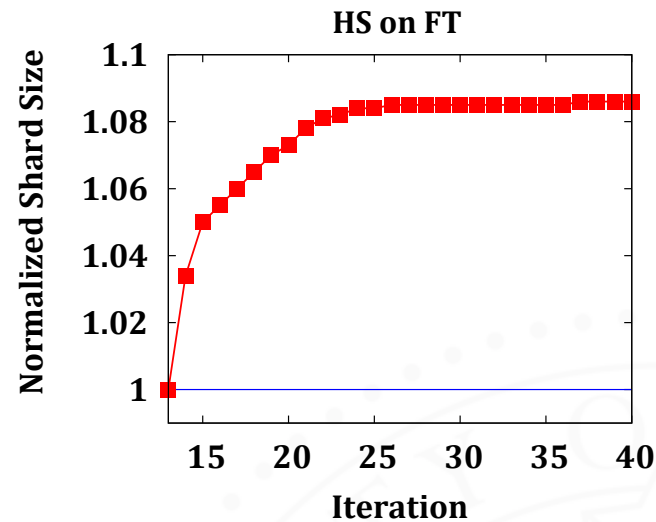
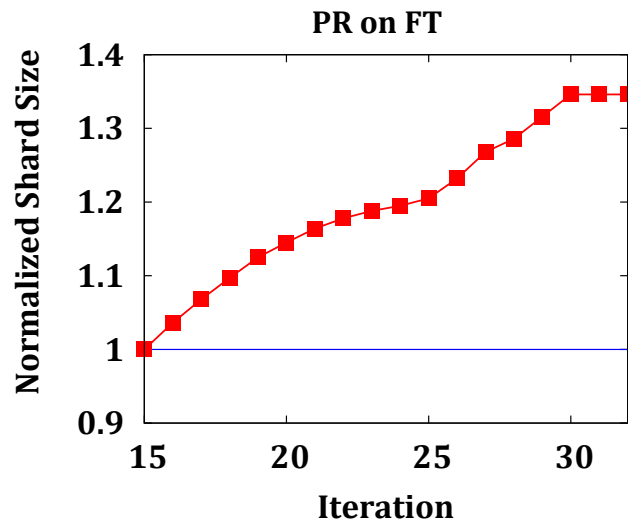


Reads & Writes

- Up to 64% reduction in data read
- Up to 54% reduction in data written
- Shadow iterations are I/O intensive

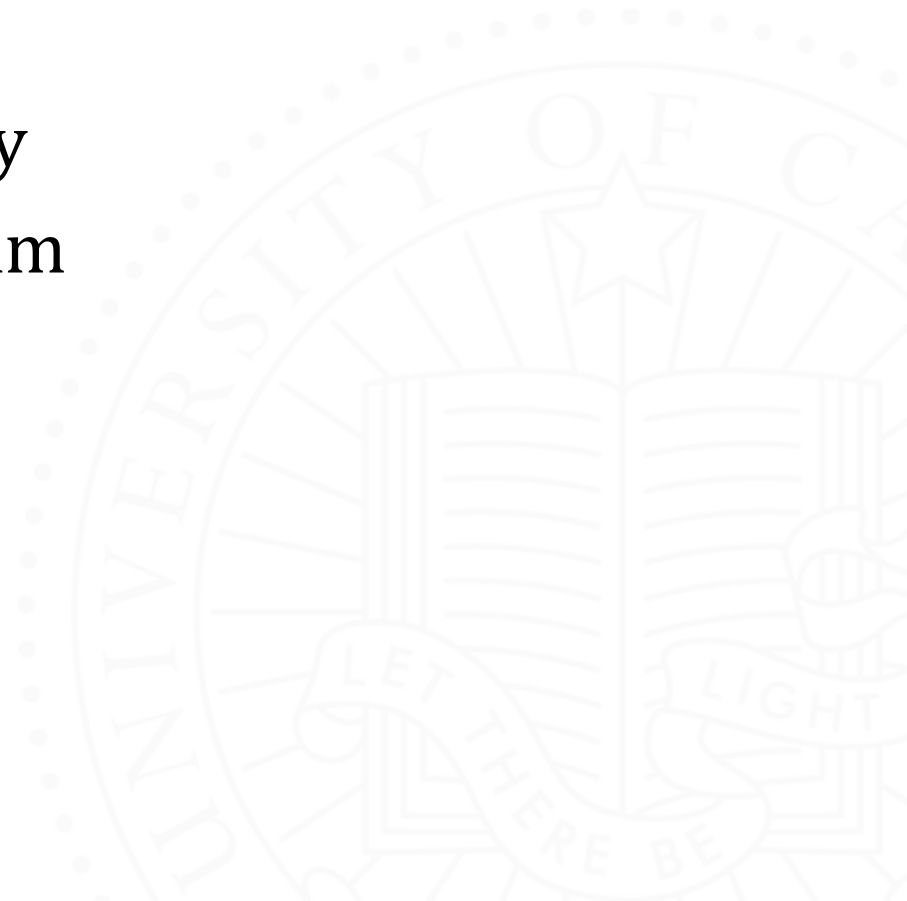


Dynamic Shard Size



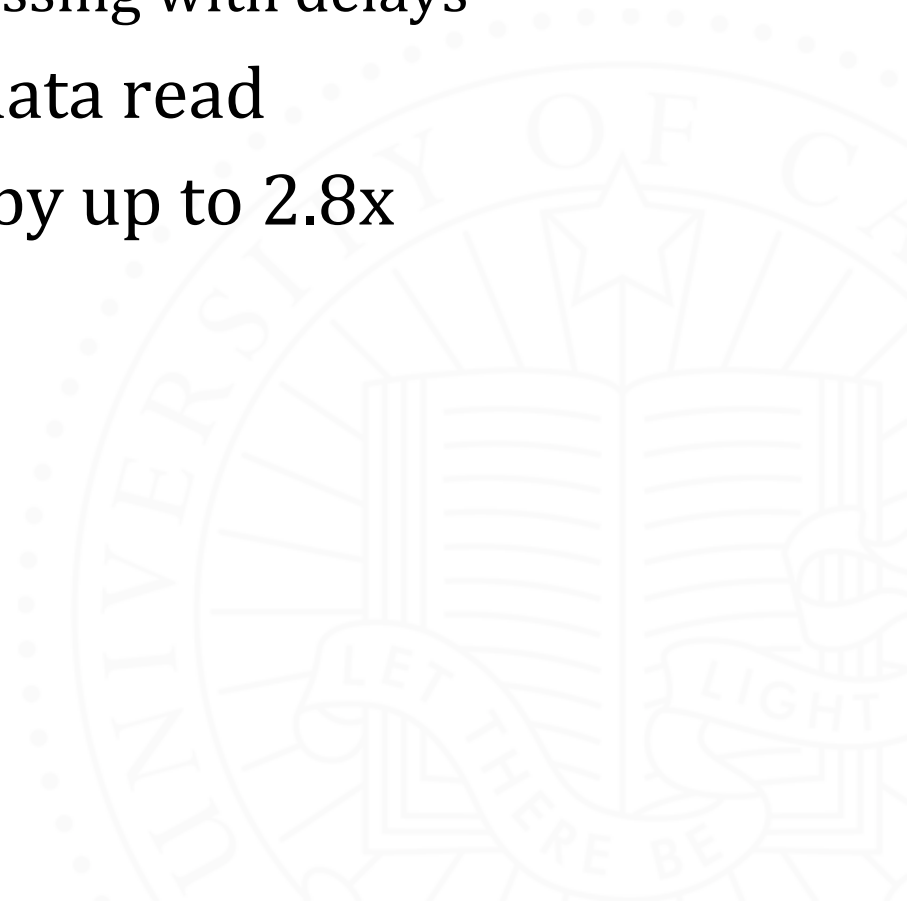
More Results

- › All benchmarks & inputs
- › Performance on SSD
- › Speedups per iteration
- › Disk space utilization
- › Edge utilization efficiency
- › Comparison with X-Stream



Conclusion

- ▶ Dynamic partitions accelerate out-of-core processing
 - ▶ Light-weight creation strategy
 - ▶ Accumulation based processing with delays
- ▶ Up to 76% reduction in data read
- ▶ Accelerate performance by up to 2.8x



Thanks

