Hybrid Merge/Overlap Execution Technique for Parallel Array Processing

Emad Soroush Magdalena Balazinska

Dept. of Computer Science and Engineering

University of Washington, Seattle, USA

{soroush,magda}@cs.washington.edu

Workshop Array Databases 2011,
March 25, 2011, Uppsala, Sweden.

Outline

- Motivation: Why Array?
- Two techniques for parallel array processing
  - Merge
  - Overlap
- Contribution: Hybrid technique
- Evaluation

SciDB*: Array DB System for Science

- Sciences are increasingly data rich.
- Existing database systems do not meet needs.
  - Relational model is ill-suited for sciences.
  - Relational operations are ill-suited for sciences.
- SciDB is a new type of database system
  - Based on a multidimensional array data model.
  - Specialized operations: regrid, matrix ops, slice, etc.
  - Parallel system for shared-nothing architecture.

*** SciDB: http://www.scidb.org/**

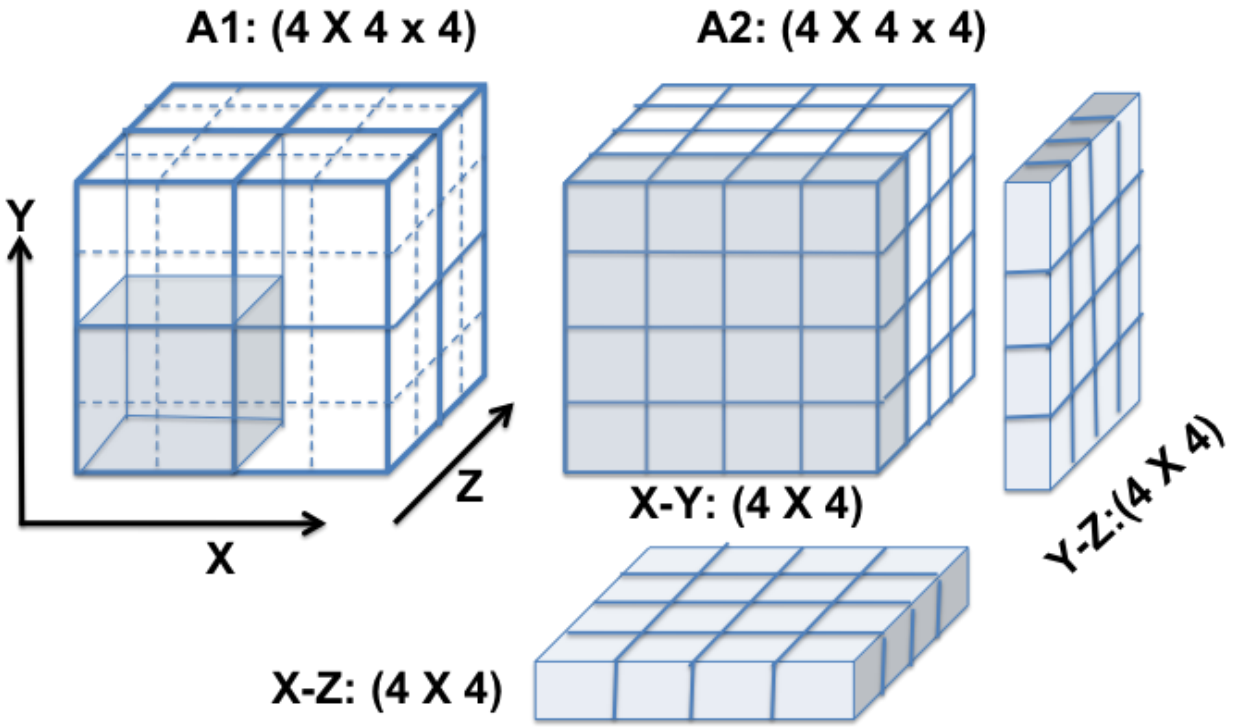
Array Engine Brief History

Many engines built to support Array	
MOLAP [2,3]	
Multi-dimensional indexing. R-Tree, KD-Tree	
App specific array systems: T2 [4], Titan[5]	
General-purpose array systems	
RDBMS-based	**Designedfrom scratch**
RIOT[10],Rasdaman[7],MAD skills[8], RAM[9]	SciDB[6]

ARRAY STORAGE IN SCIDB

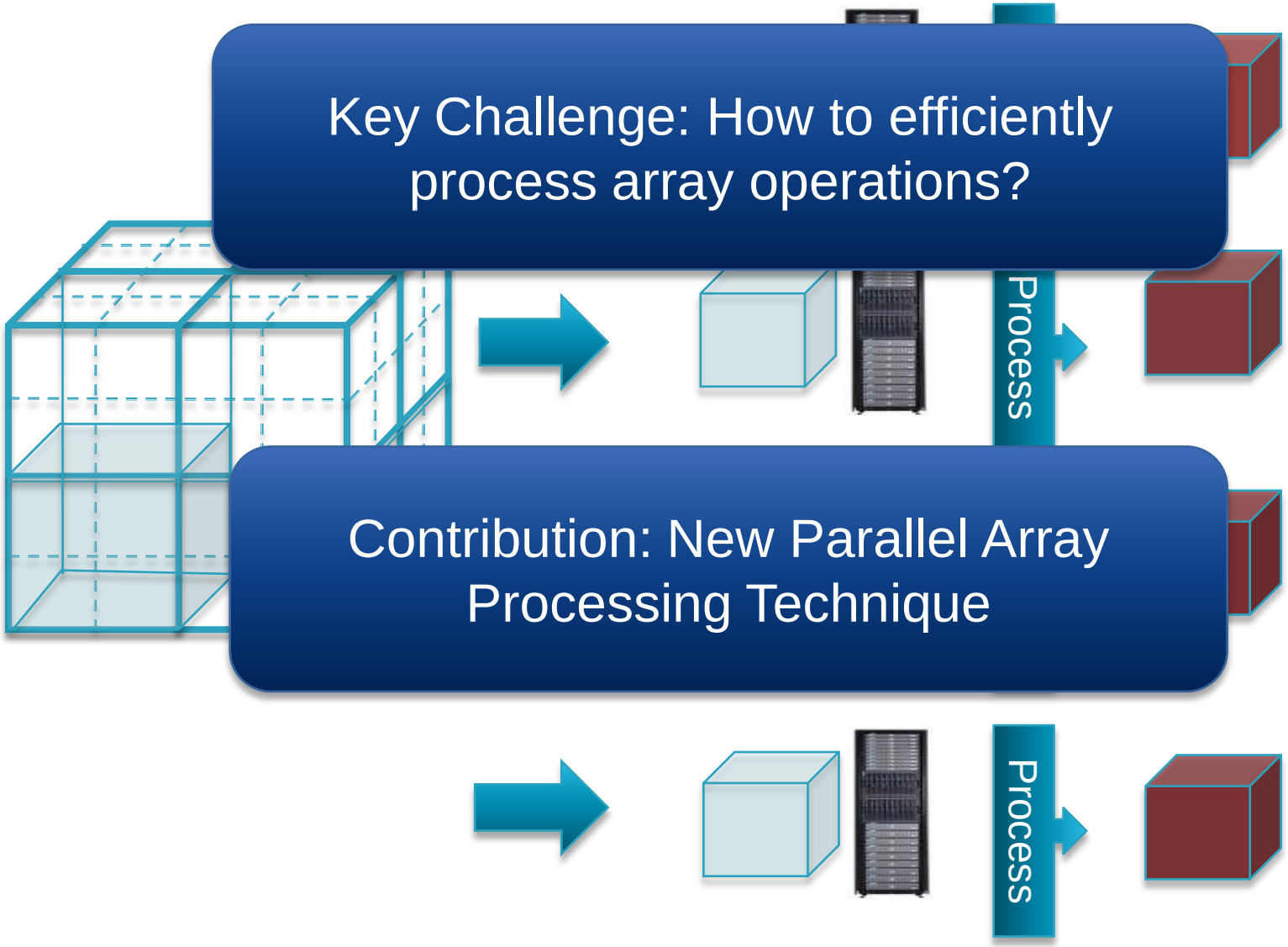
- An array is partitioned into subarrays called **chunk**.

Chunking alleviate dimension dependency



X-Z and Y-Z requires 8 blocks to read in A2 but 4 blocks in A1

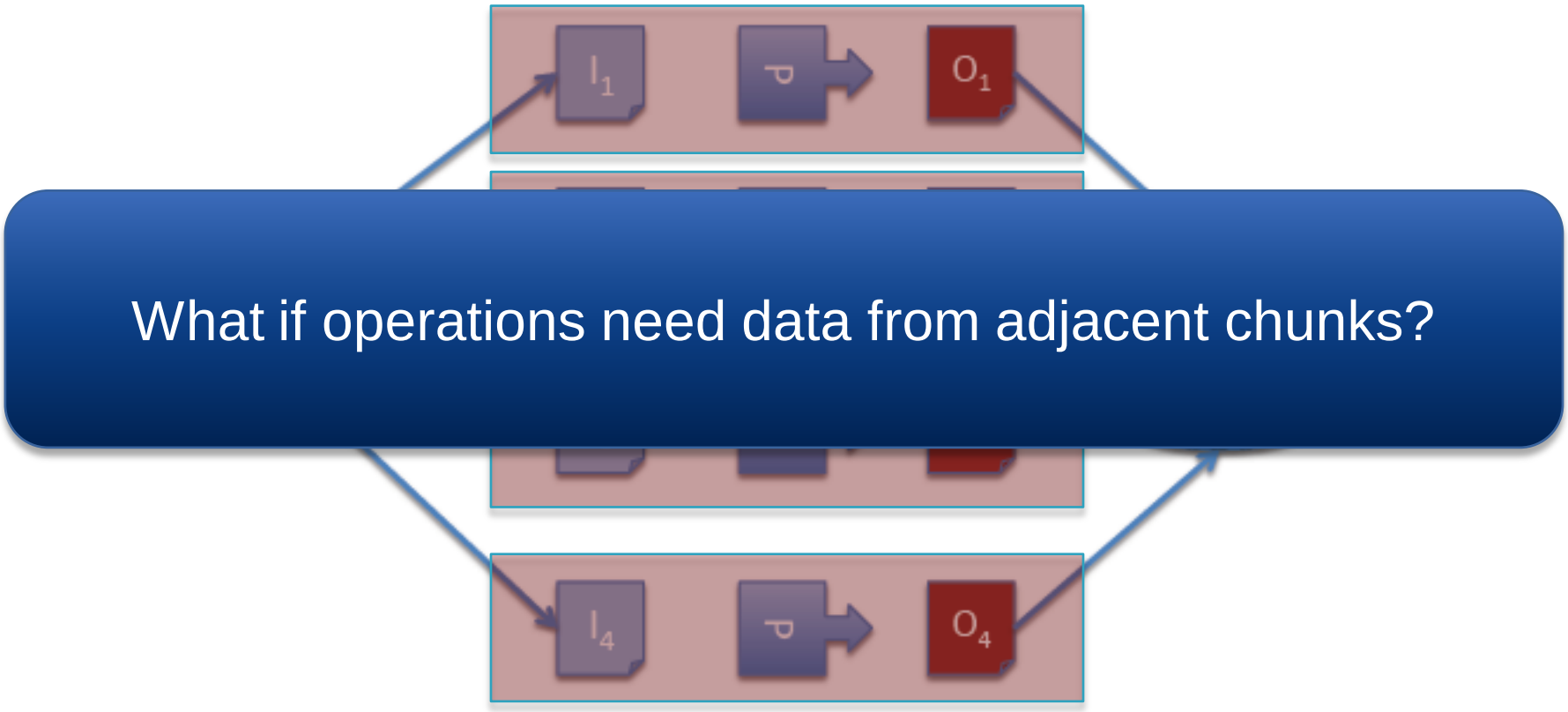
Array Partitioning Across Nodes



Outline

- Motivation: Why Array?
- Two techniques for parallel array processing
  - Merge
  - Overlap
- Contribution: Hybrid technique
- Evaluation

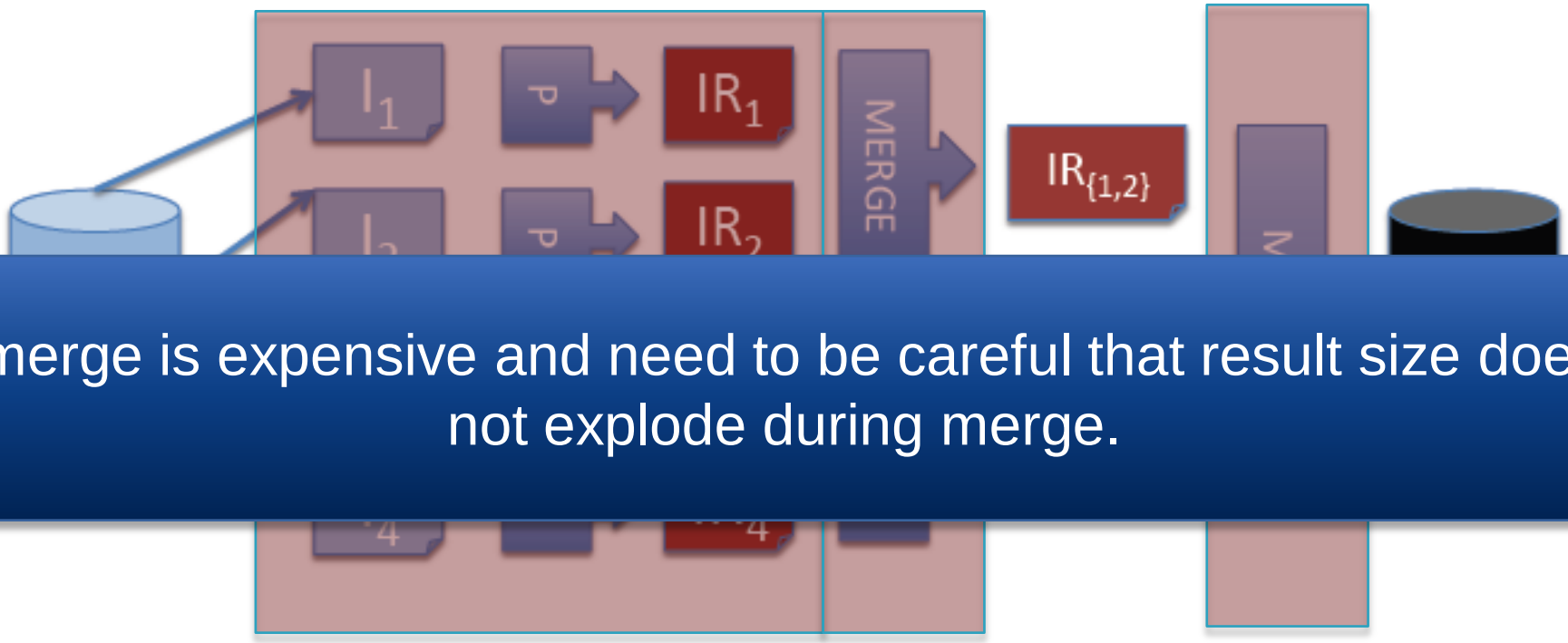
BASICAPPROACH



- Works well for *independent* operations.
  - Operations that process array cells independently.
  - Example: filter, slice.

MERGE STRATEGY

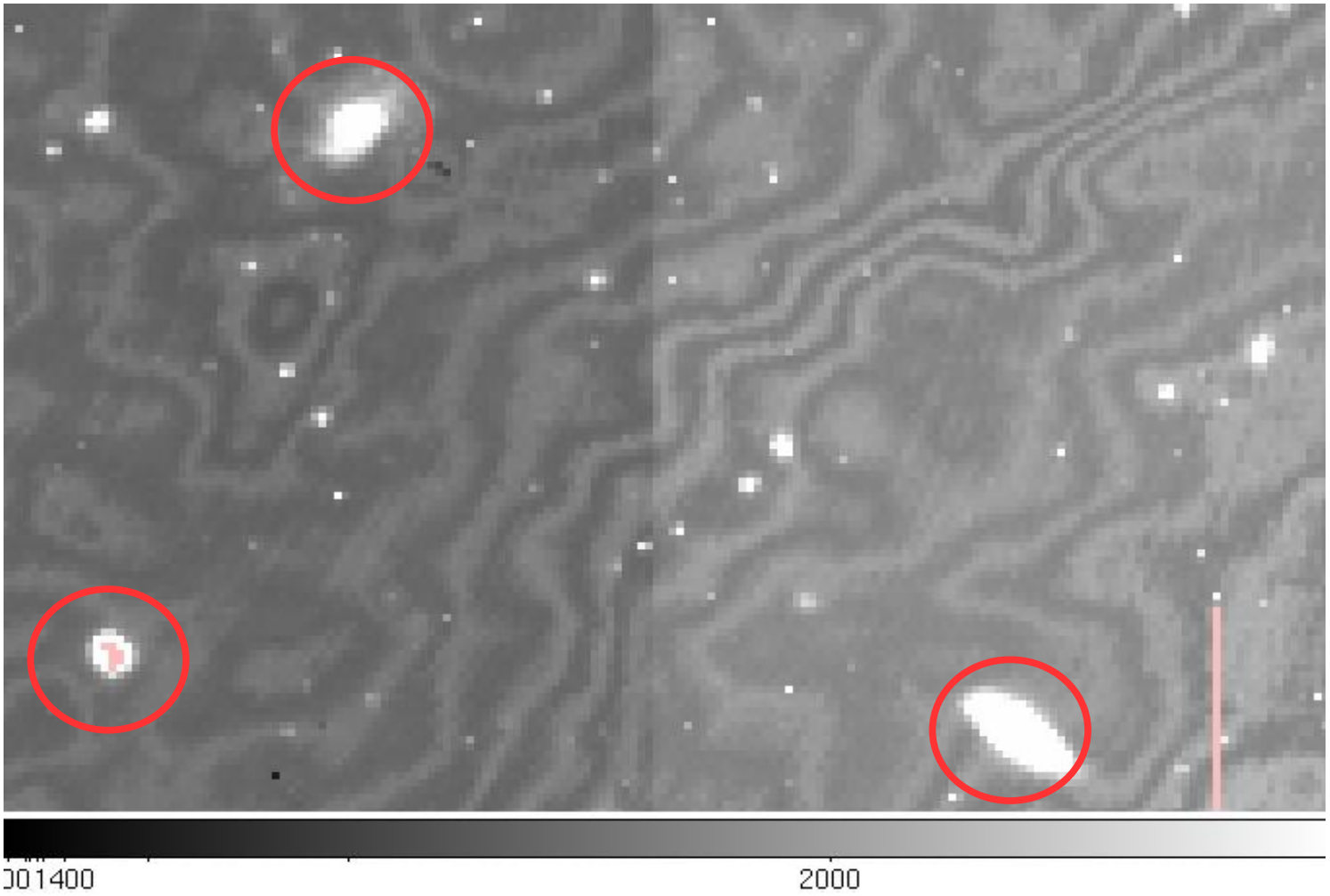
PROCESS AND MERGE



merge is expensive and need to be careful that result size does not explode during merge.

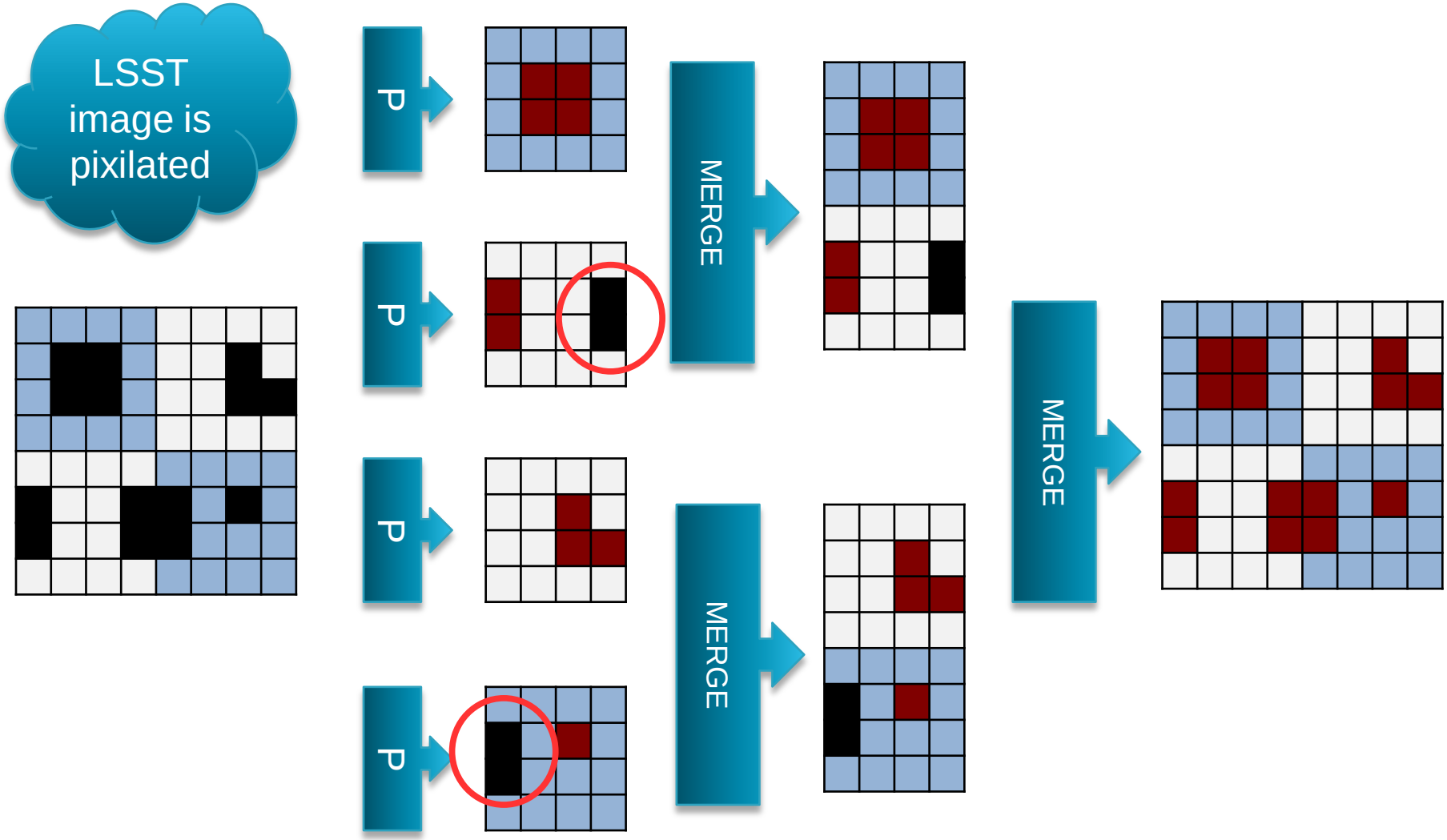
- Works well for many *dependent* array operations.
  - Example: avg(), regrid(), count()

LSST[1] Object Detection Function



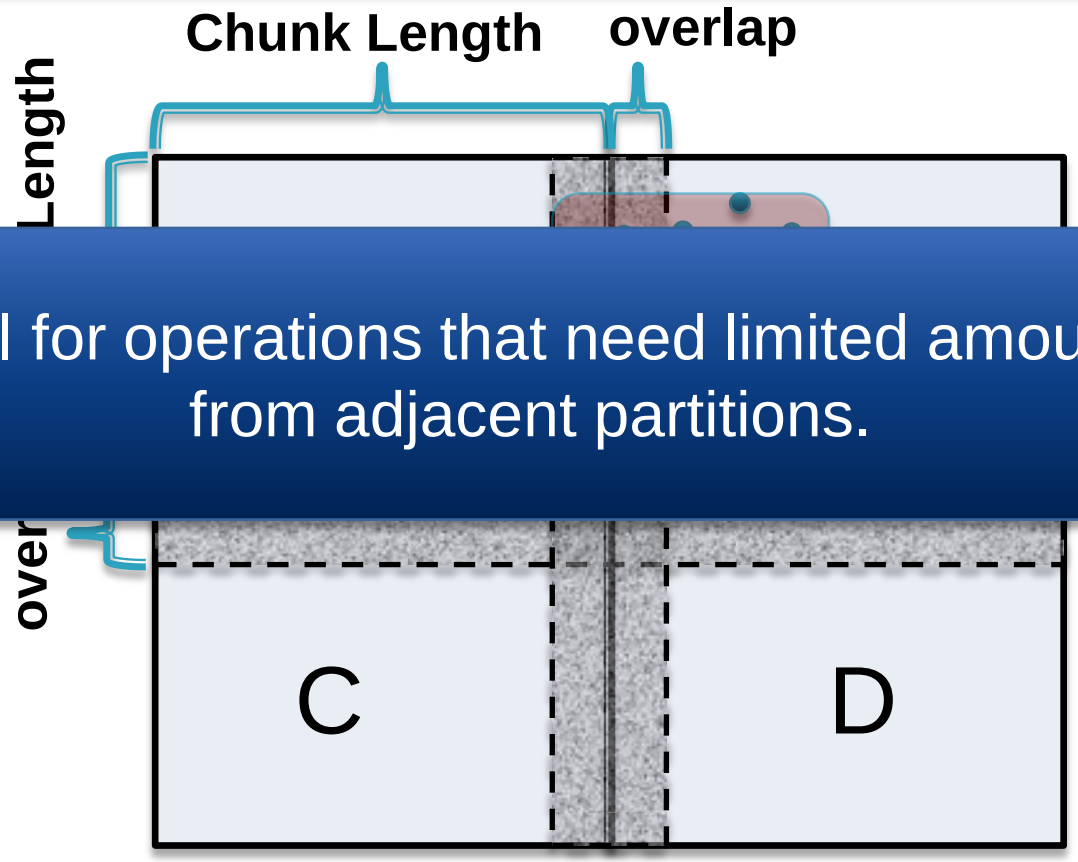
LSST: http://www.lsst.org/lsst/science/concept_data

MERGE STRATEGY (LSSTEXAMPLE)



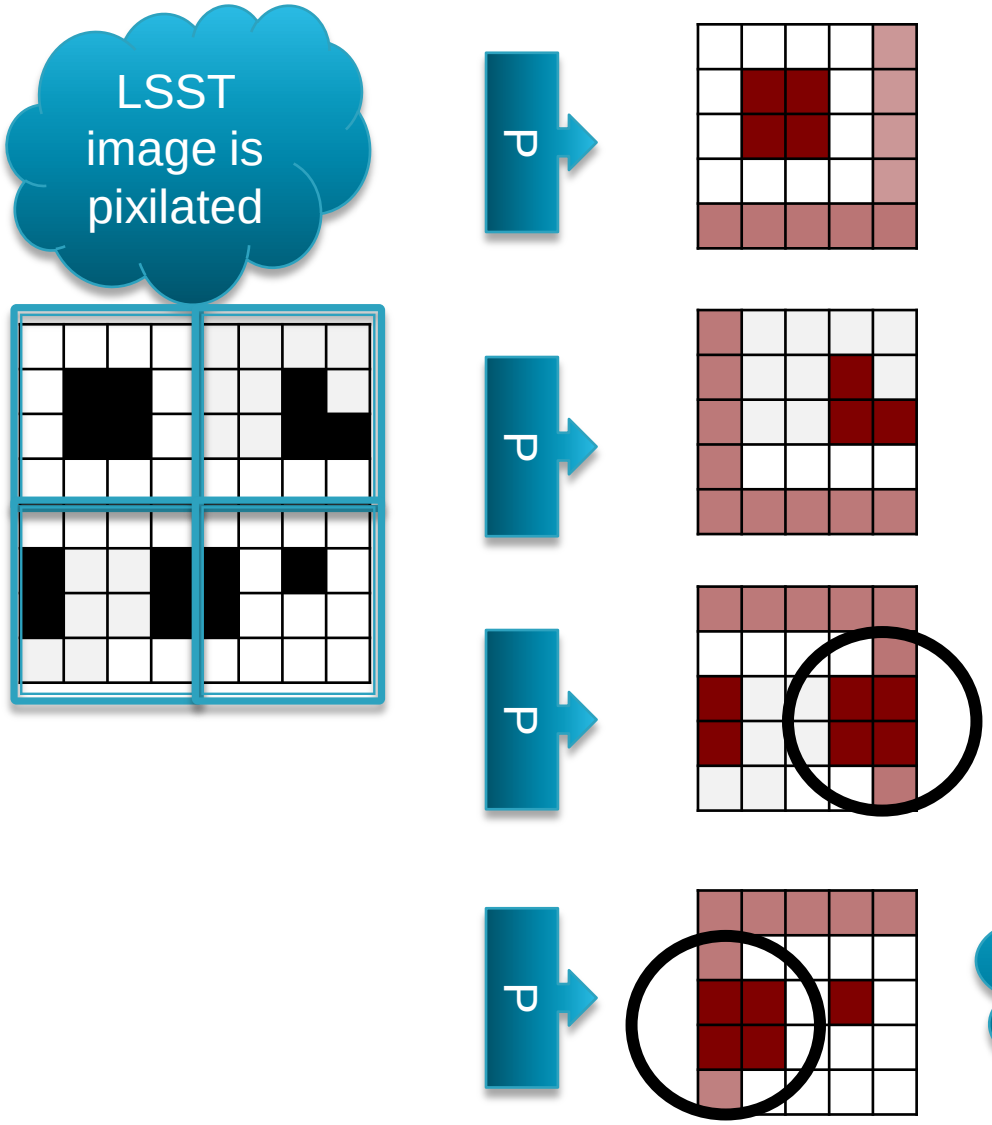
Overlap Strategy

PROCESS WITH OVERLAP



Works well for operations that need limited amount of data from adjacent partitions.

Overlap Strategy (LSSTExample)



- Duplicate results possible.
- Some duplicate resolution mechanism required.
- Cluster **centroid** to resolve duplicate for this example.

overlap length is 1 cell

Overlap vs. Merge Strategy

- Overlap Strategy:
- P1: Significant Overhead both in I/O and CPU

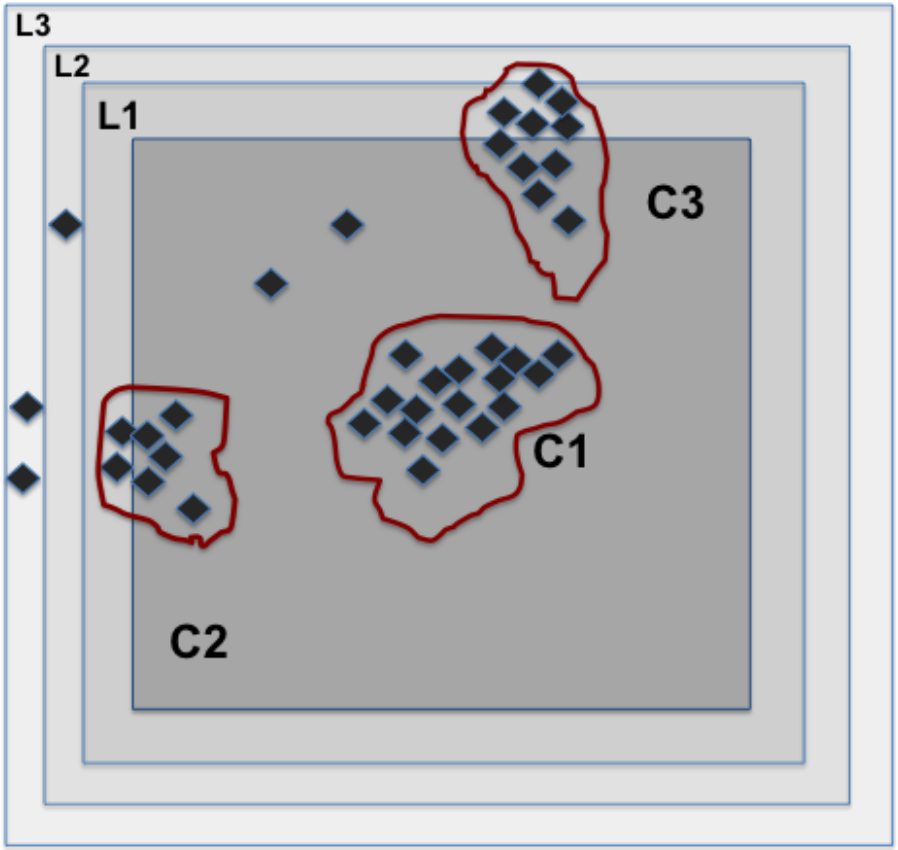
10% overlap size along each dimension		
2D	3D	6D
21%	33%	75%

What about unbounded dependent array operations?
Example: cluster spans multiple partitions

- "Merge" post processing as alternative. But "Merge" is expensive.

Maybe a hybrid (Overlap & Merge) strategy is a solution.

MULTI-LAYER OVERLAP



***ArrayStore: A Storage Manager for Complex Parallel Array Processing*, Emad Soroush, Magdalena Balazinska, and Daniel Wang. SIGMOD 2011**

Outline

- Motivation: Why Array?
- Two techniques for parallel array processing
  - Merge
  - Overlap
- **Contribution: Hybrid technique**
- Evaluation

CONTRIBUTION: **HYBRID** TECHNIQUE

Output	Function	Input
(Result_Chunk,Merge_Chunk)	process	Input_Chunk
{(Result_Chunk,Merge_Chunk)}	merge	{(Result_Chunk,Merge_Chunk)}
Merge_Chunk	filter	(Merge_Chunk,Bitmap)

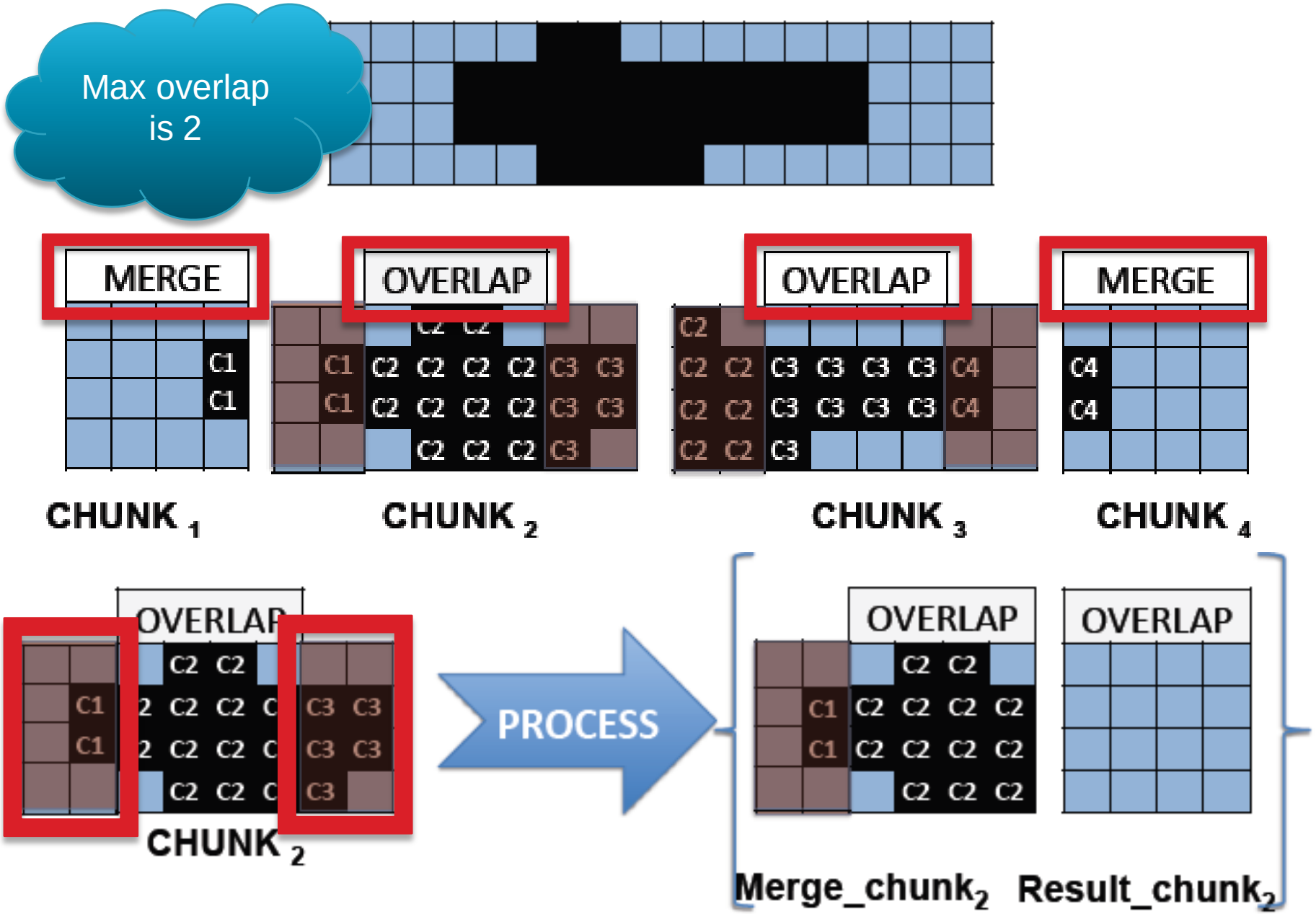
with or without overlap

filter out duplicates

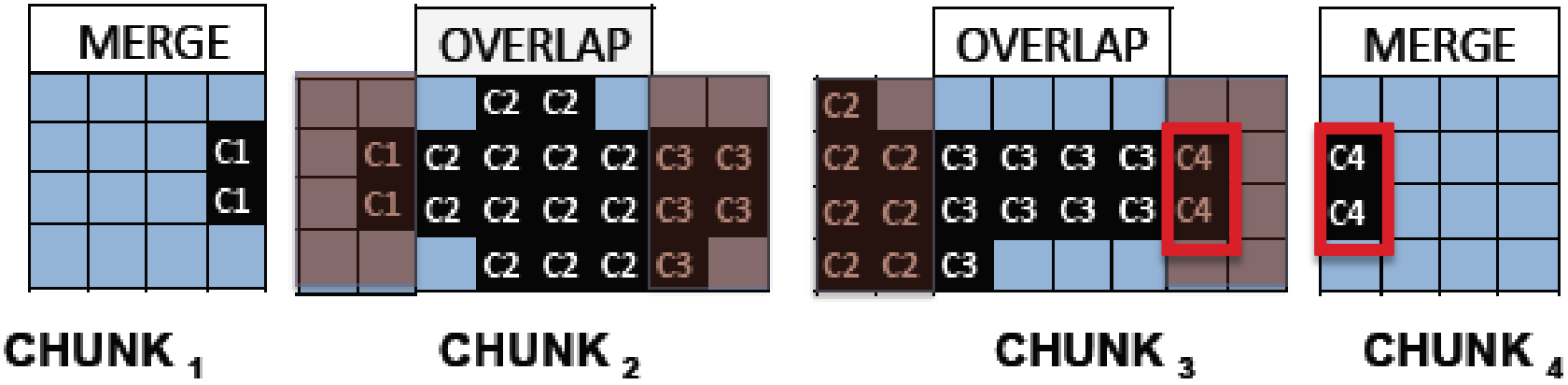
do as little as possible

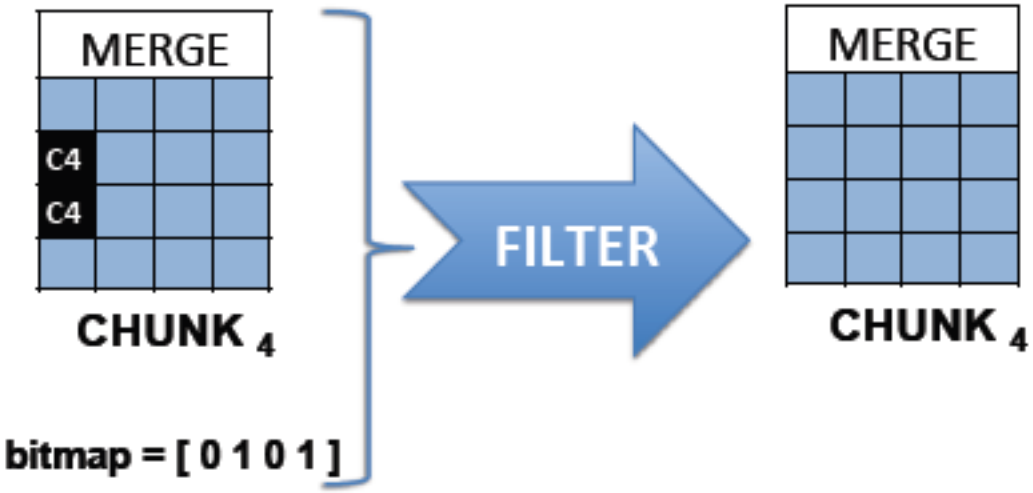
- "Merge_Chunk" contains intermediate results.
- "Result_Chunk" contains final results.
- "Bitmap" indicates chunks processed with either "overlap" or "merge".
- We do not address how "Bitmap" is generated.

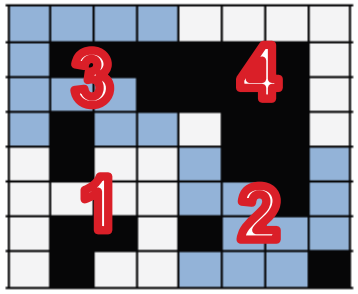
HYBRID PROCESS WITH BITMAP



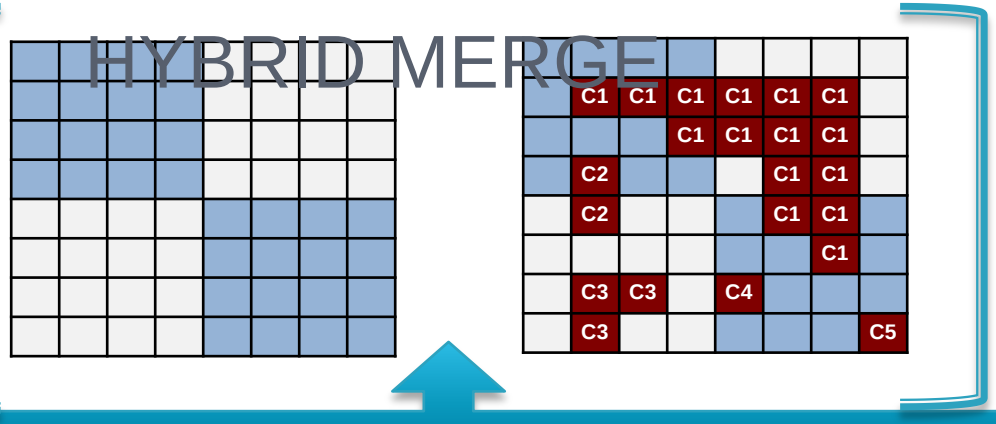
HYBRID FILTER



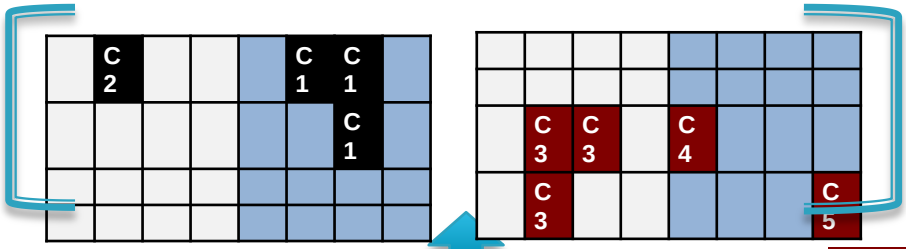


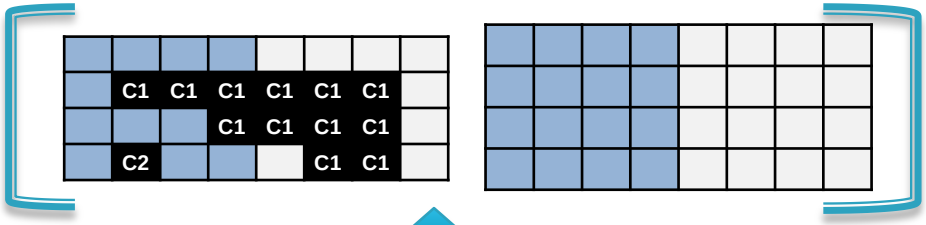


HYBRID MERGE

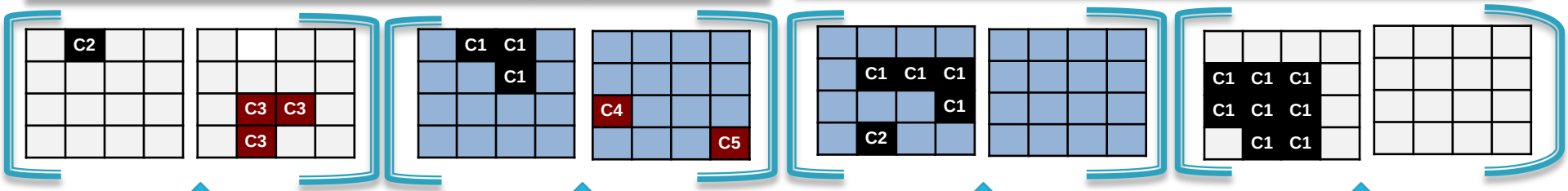


$(M_{\{4,3\}},R_{\{4,3\}})$ MERGE $(M_{\{2,1\}},R_{\{2,1\}})$ **Dim Y**





(M_4,R_4) MERGE (M_3,R_3) **Dim X** (M_2,R_2) MERGE (M_1,R_1)



Process **Process** **Process** **Process**

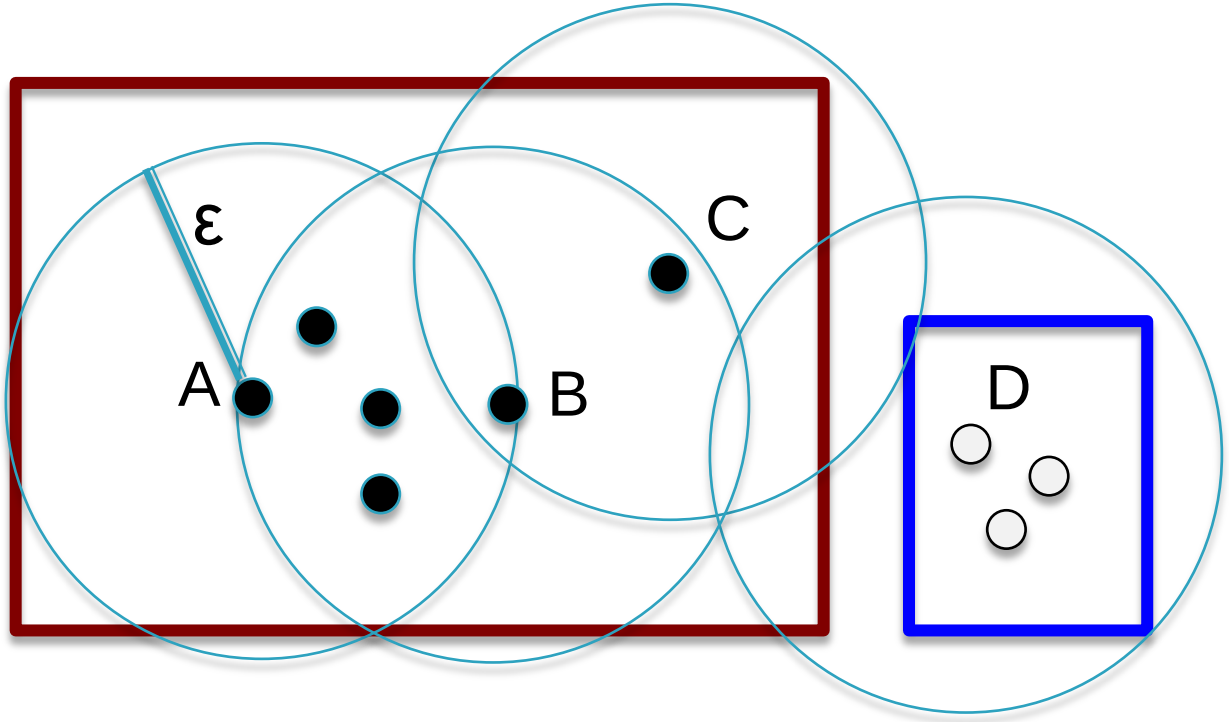
Outline

- Motivation: Why Array?
- Two techniques for parallel array processing
  - Merge
  - Overlap
- Contribution: Hybrid technique
- **Evaluation**

PRELIMINARY EVALUATION

- Experiments on 3D astronomy simulation dataset. 74 GB
- ARRAY simulation{id,vx,vy,vz,mass,phi}[X,Y,Z]
- Single Node experiments.
- Experiments for **Bounded Dependent Array Operation.**
- ARRAY has (16x16x4) number of chunks.
- 20 layers of overlap for chunk, covers 0.5 of chunk dimension length.
- Friends of Friends (FoF) application.

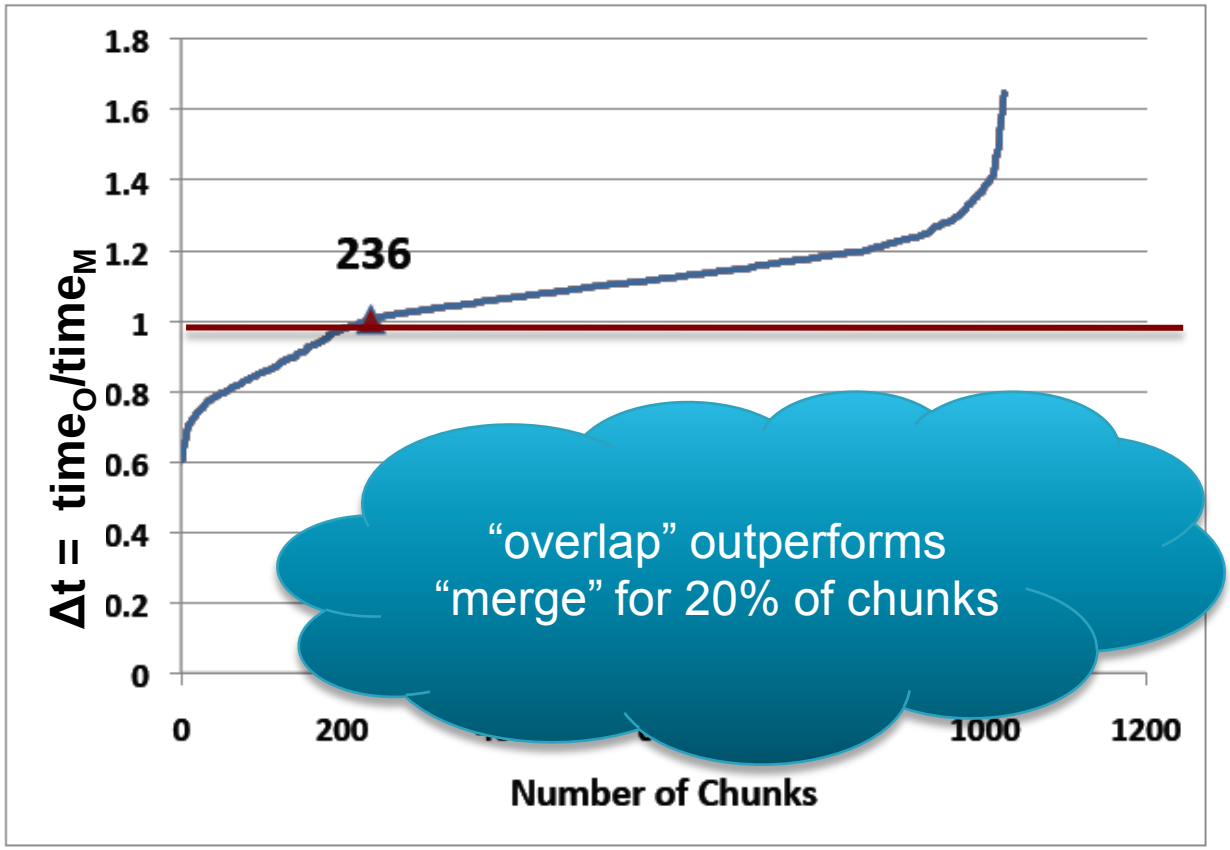
Friends of Friends (FoF)

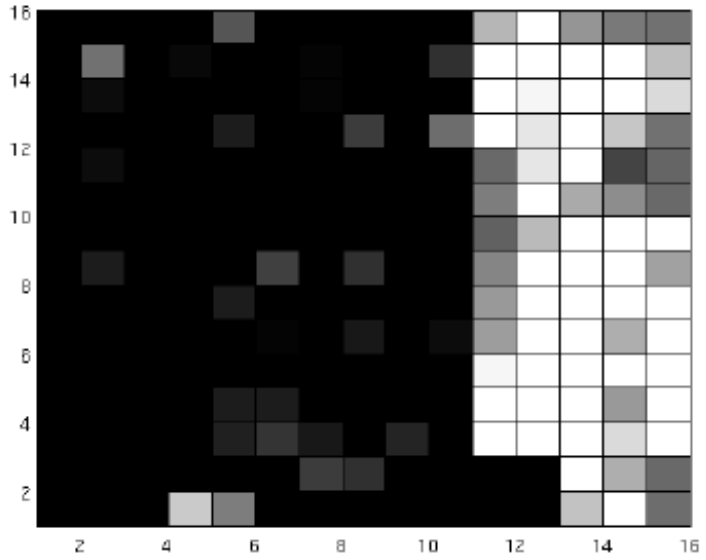


- **A** and **B** are Friends. **B** and **C** are Friends. **A** and **C** are **not** Friends.
- **A** and **C** holds **FoF**relation.

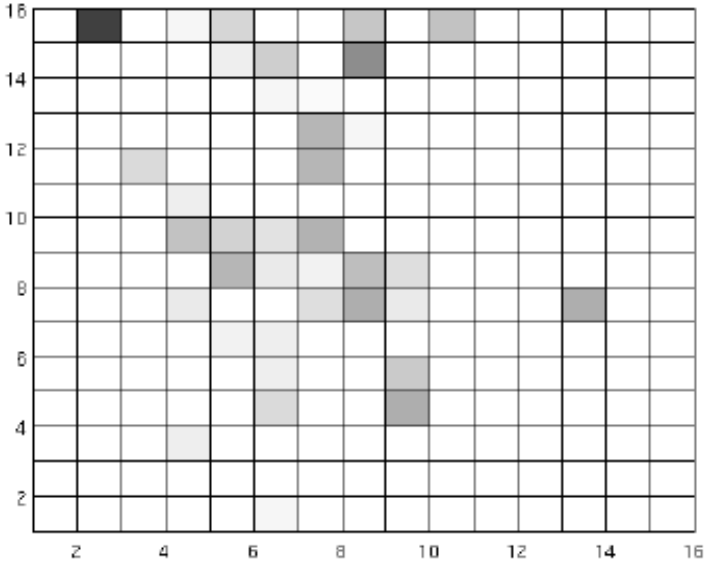
PRELIMINARY EVALUATION

time(**overlap**) >time(**merge**) >time(**hybrid**)

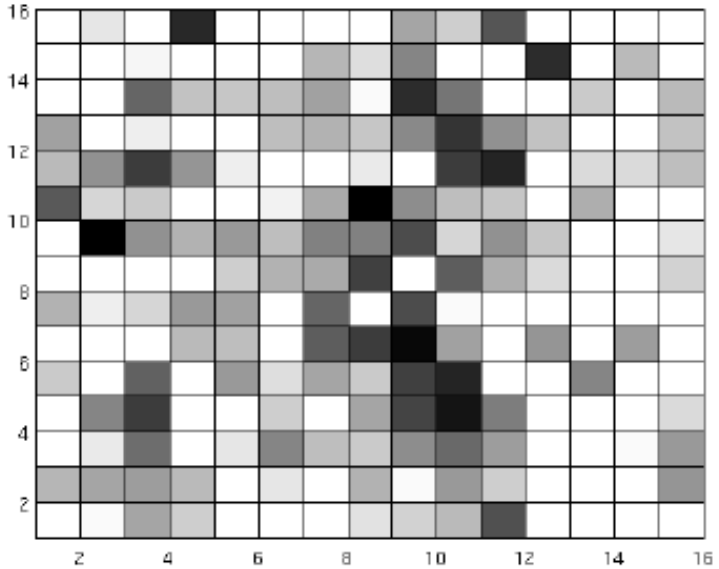




(a) X-Y slice of the array with Z=1



(b) X-Y slice of the array with Z=2



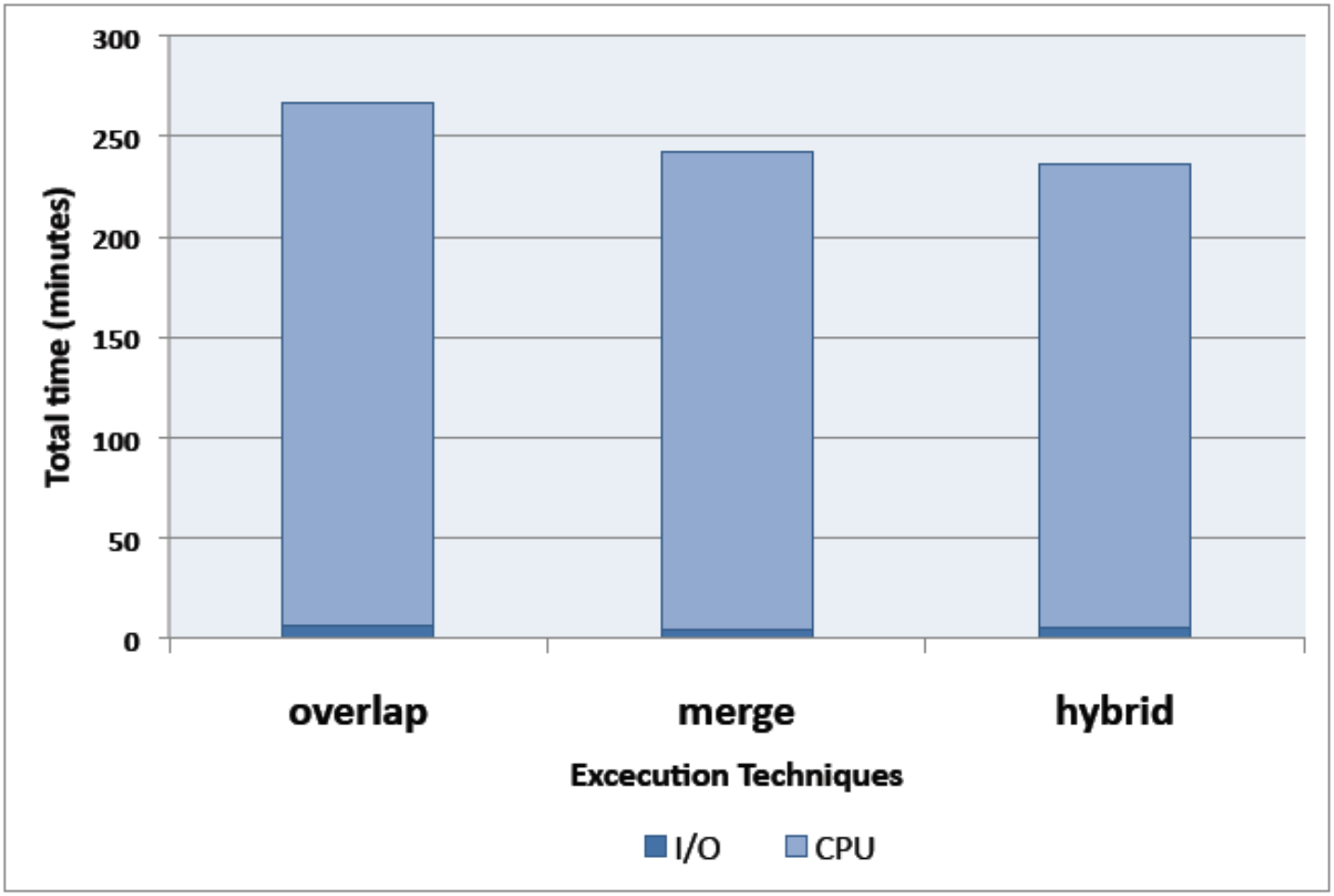
(c) X-Y slice of the array with Z=4

CONCLUSION

- Hybrid technique = “merge” + “overlap”
- API and execution method to support hybrid approach.
- Experiments show hybrid is better than either uniform one.
  - Evaluation on Bounded Dependent Array Operation.
- Future work includes:
  - Evaluation of the hybrid approach on unbounded parallel array operation.
  - Address the problem of automated selection of the execution techniques.

Thank You, Questions ?

Preliminary Evaluation



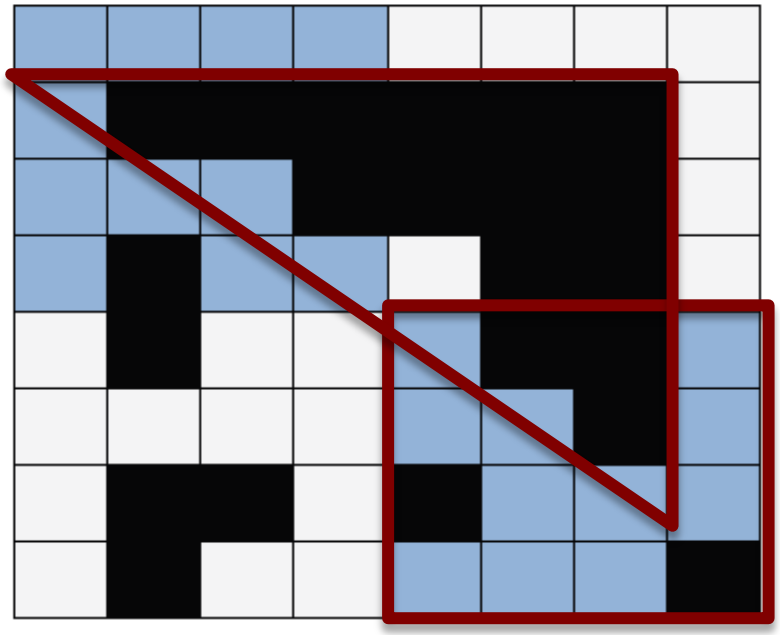
Array Operation Taxonomy

	Independent	Bounded Dependent	Unbounded Dependent
Algebraic	N/A	Regrid, Cluster Centroids (bounded-size clusters)	Cluster Centroids (no bound on size)
Holistic	Filter, Slice	Smooth, Cluster Centroids (bounded-size clusters)	Cluster Centroids (no bound on size)

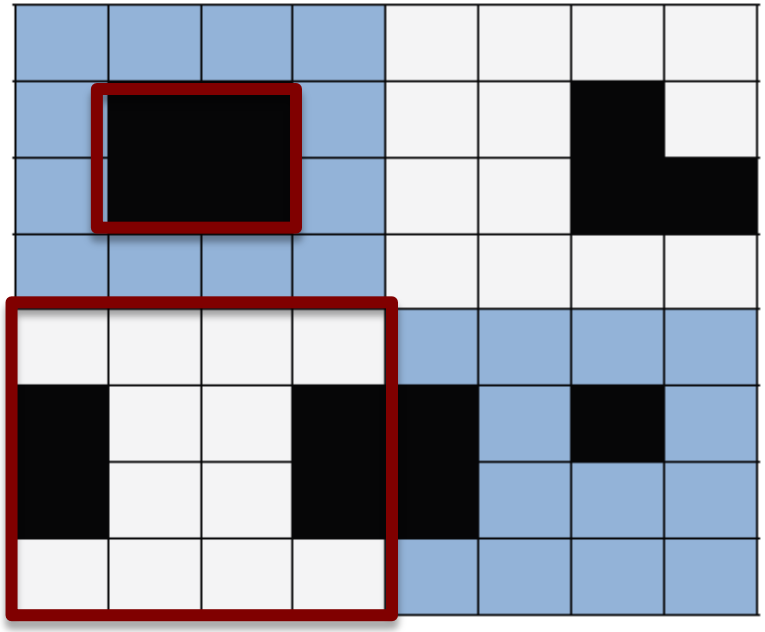
Processing Approach Applicability

	Independent	Bounded Dependent	Unbounded Dependent
Algebraic	N/A	merge& overlap	merge
Holistic	Independent	merge& overlap	merge

Bounded vs. UnboundedExample



(a) Unbounded clusters



(b) Bounded clusters

REFERENCES

- [1] http://www.lsst.org/lsst/science/concept_data
- [2]Tsuji et. al. An extendible multidimensional array system for molap. In Proc. of the 21st SAC Symp, pages 503-510, 2006.
- [3] Pedersen et. al. Multidimensional database technology. IEEE Computer, 34(12):40-46, 2001.
- [4] Chang et. al. T2: a customizable parallel database for multi-dimensional data. SIGMOD Record, 27(1):58-66,1998.
- [5] Chang et. al. Titan: A high-performance remote sensing database. In Proc. of the 13th ICDE Conf., pages 375-384,1997.
- [6]Mike Stonebraker et. al. Requirements for science data bases and SciDB. In Fourth CIDR Conf. (perspectives), 2009.
- [7] Baumann et. al. The multidimensional database system RasDaMan. In Proc. of the SIGMOD Conf.pages 575-577, 1998.
- [8]Cohen et. al. MAD skills: new analysis practices for big data. PVLDB, 2(2):1481-1492, 2009.
- [9] Ballegooij et. al. Distribution rules for array database queries. In 16th. DEXA Conf., pages 55{64, 2005.
- [10] Zhang et. al. RIOT: I/O-ecient numerical computing without SQL. In Proc. of the Fourth CIDR Conf., 2009.