Supercomputing 2009

SmartStore: A New Metadata Organization Paradigm with Semantic-Awareness for Next-Generation File Systems

Yu Hua Hong Jiang Yifeng Zhu

Dan Feng Lei Tian

Outline

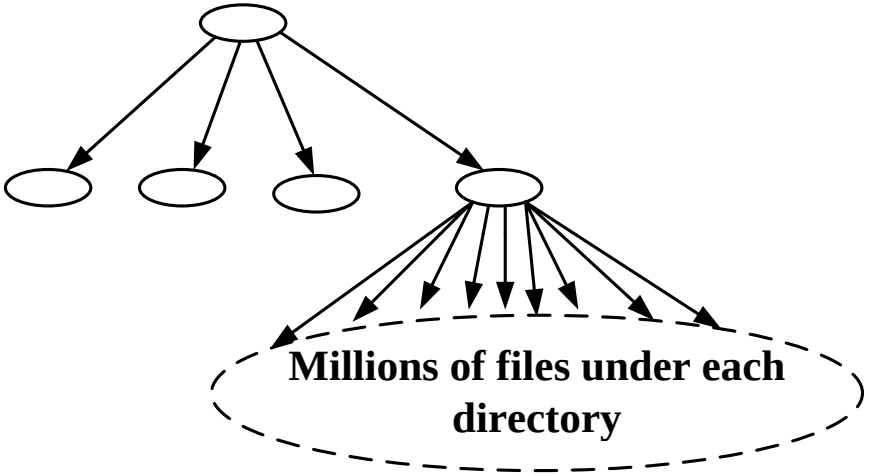
- **Motivations**
- **SmartStore System**
- **Key Issues**
- **Performance Evaluation**
- **Discussion and Conclusion**

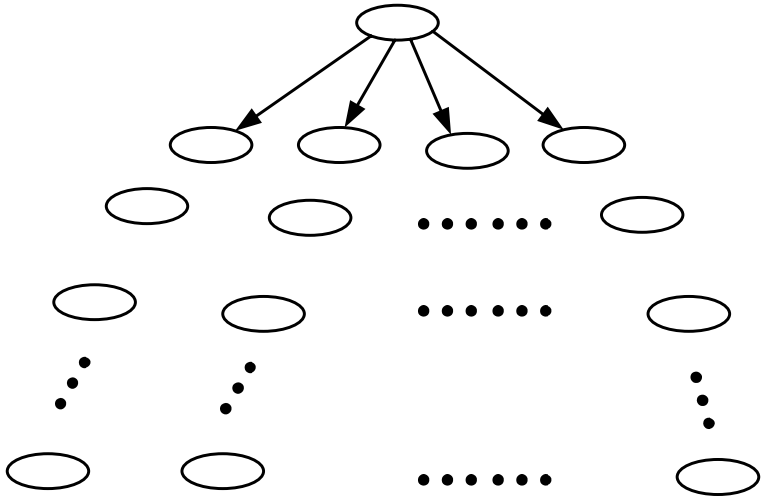
Motivations

- **Some Facts**
  - *Storage capacity → Exabyte (or even larger)*
  - *Amounts of Files → Billions*
  - *Metadata-based transactions → over 50%*
  - *Hierarchical directory tree → Performance Bottleneck*

- **Inefficiency of current file systems**
  - *Static and inflexible I/O interfaces*
  - *Linearly brute-force searching*
  - *Lack of full utilization of semantics*

Conventional Directory Trees





This tree is too FAT !

This tree is too HIGH !

Ideal Scenarios

- **User requirements**
  - *Quickly return queried results with acceptable tradeoff*
  - *Obtain interested knowledge from data ocean to guide higher-level services*
  - *Query for high-dimensional data*
- **System requirements**
  - *Scalability*
  - *Reliability*
  - *Performance improvements*

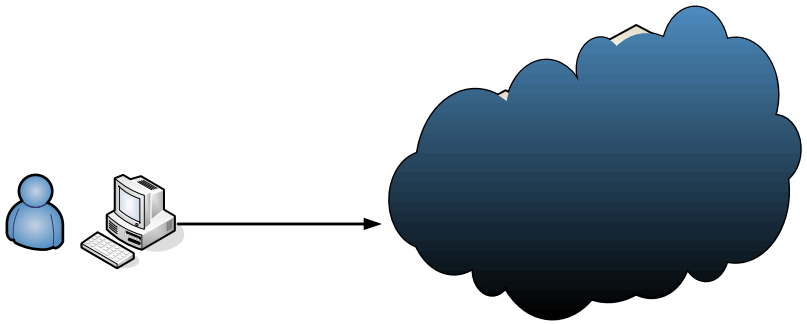
Intuition

- **Reduce search space**
  - ***Not entire large-scale file system***
- **Search correlated metadata**
  - ***Configure a context related to queries***
- **Desirable interfaces**
  - ***Such as range query and top-k query, i.e., complex queries;***

Examples: Complex Queries

Range Query:

Which files are created no more than 30 min. and larger than 2.6GB?



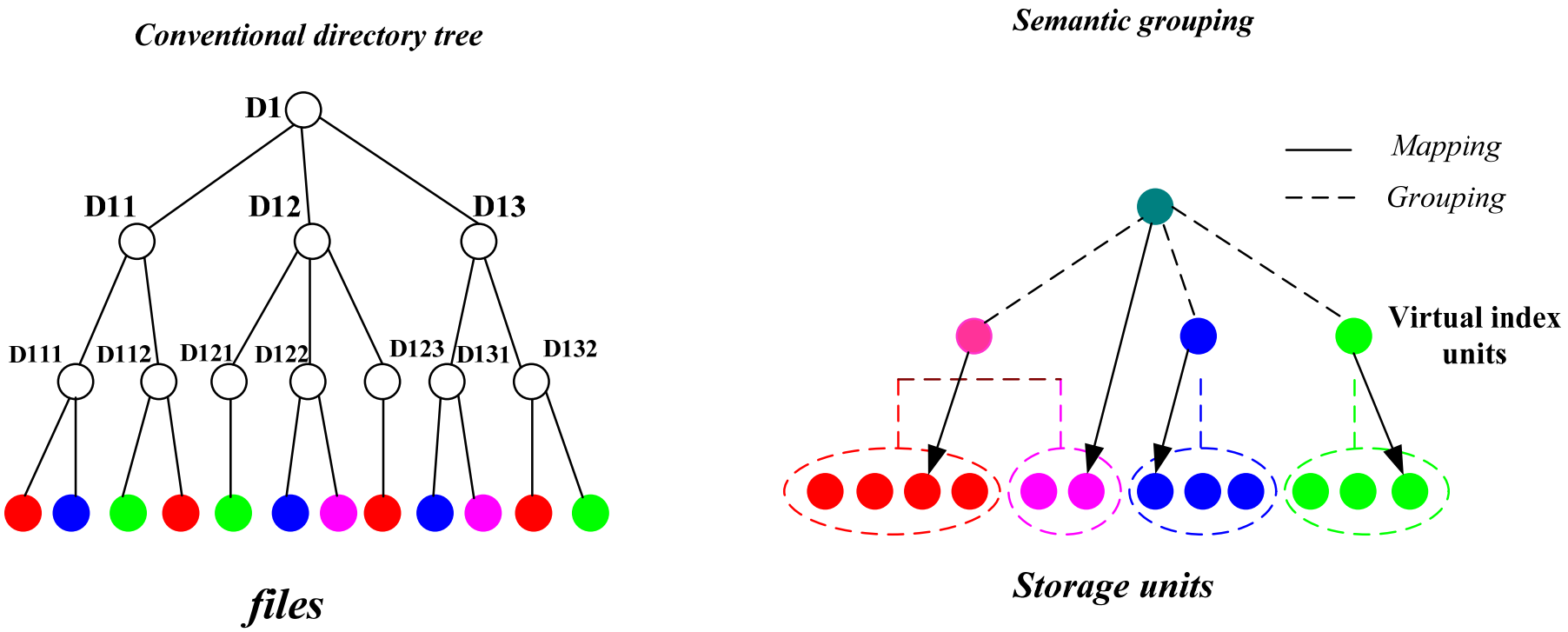
Top-k Query:

Can the system show 10 files that are closest to the description that file size is around 300MB and was last visited around Jan.1, 2008 ?

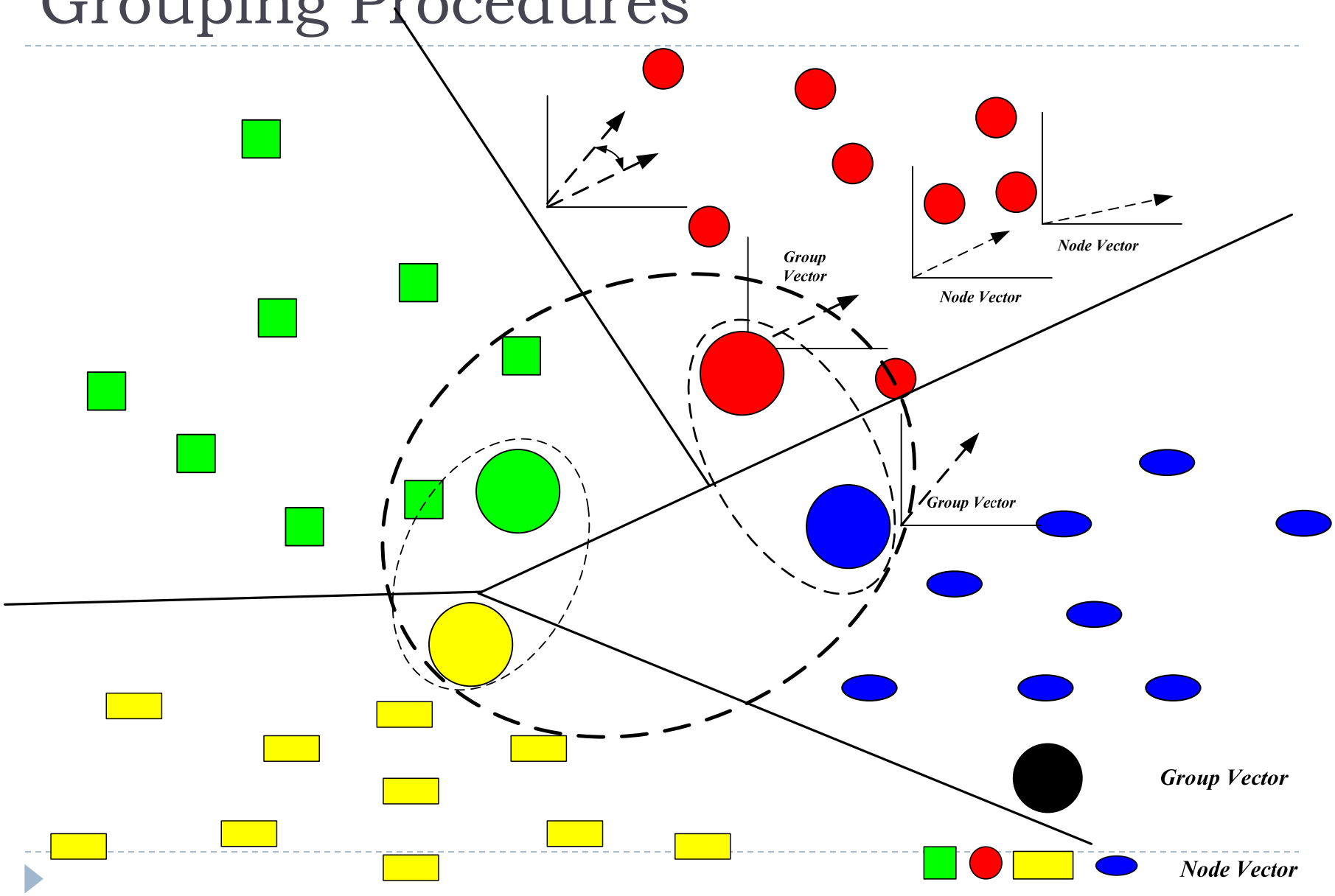
Our Approach: SmartStore

- **Basic ideas:**
  - **Semantic: correlation represented by multi-dimensional attributes of file metadata**
  - **Group files based on metadata semantic correlations by using *Latent Semantic Indexing* (LSI) tool**
  - **Query and other relevant operations can be completed *within one or a small number of such groups*.**
- **Our goal is to avoid or minimize brute-force search that is widely used in a directory-tree based file system during a complex query.**

Comparisons with Conventional File Systems



Grouping Procedures



Semantic Grouping

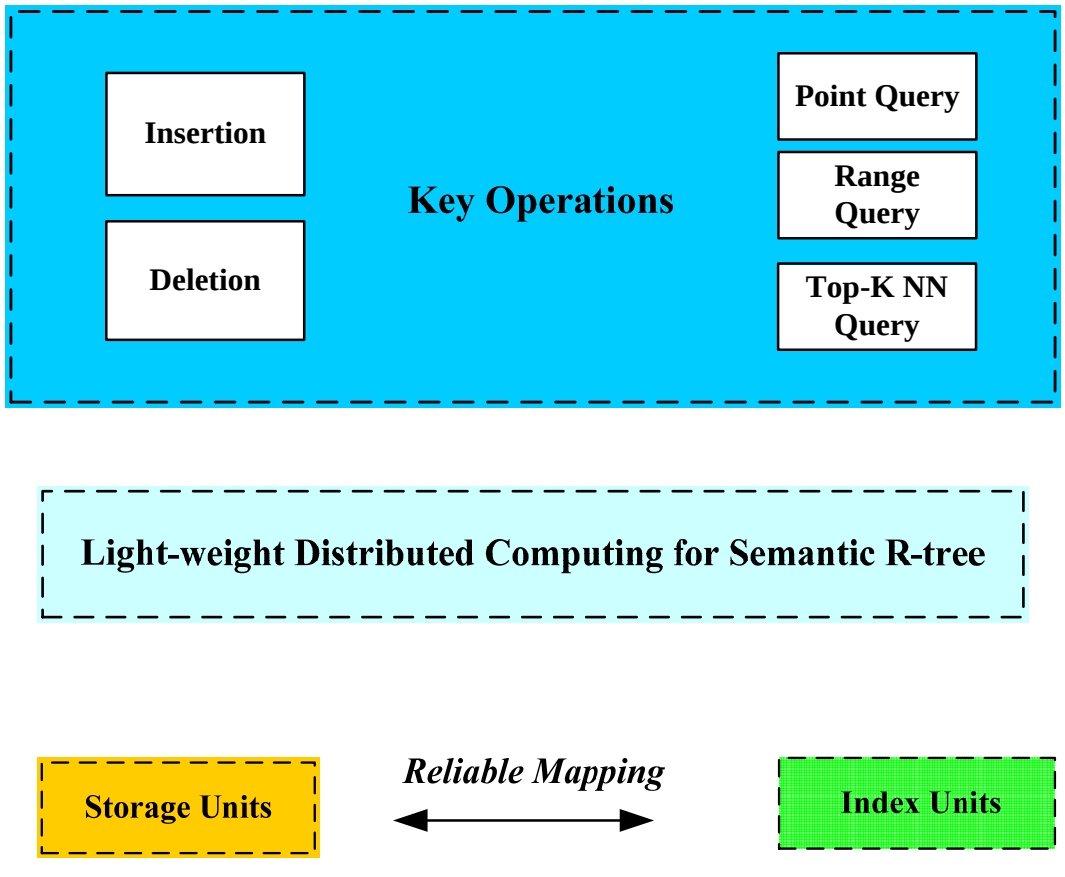
- **Design Objectives**

- ***Group sizes are approximately equal.***

- ***A file in a group has a higher correlation with other files in this group than with any file outside of the group***

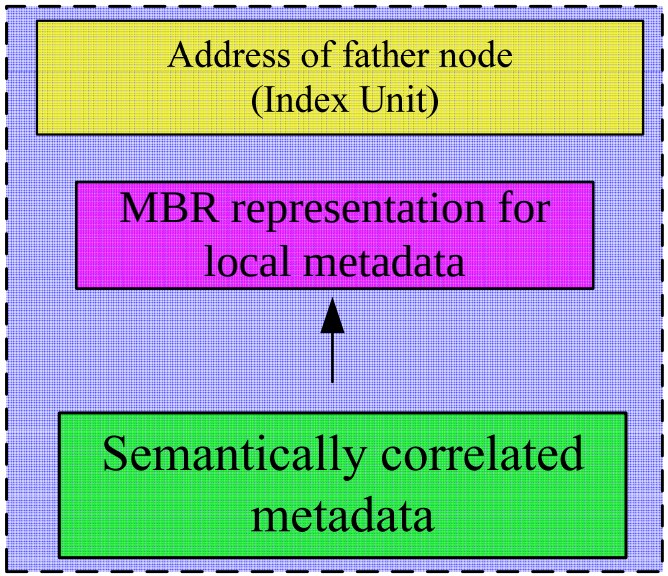
System Architecture

- **Grouping correlated metadata into storage and index units based on the LSI**
- **Construction of semantic R-trees in a distributed environment**
- **Multiple operations**

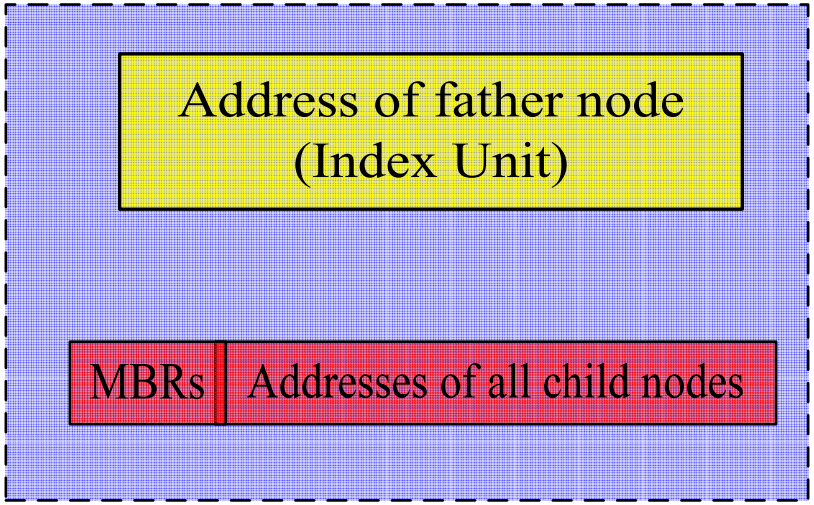


Constructing a Semantic R-tree.

- **Semantic R-tree leaf nodes as *storage units***
- **The non-leaf nodes as *index units***



Storage Unit



Index Unit

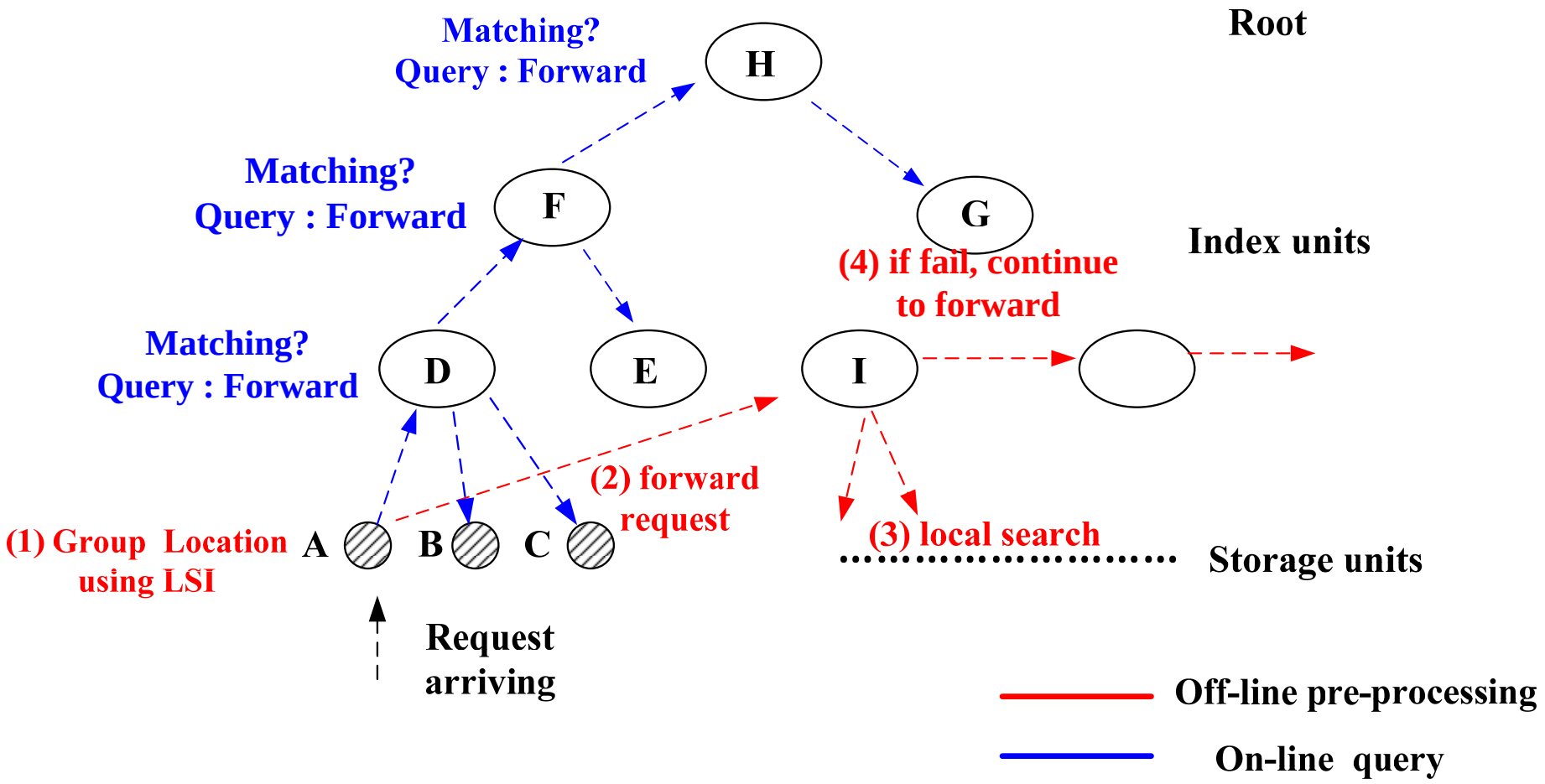
SmartStore functions

- **Insertion**
- **Deletion**
- **On-line Query Approaches**
  - ***Range Query***
  - ***Top-K Query***
  - ***Point Query***

Key issues: *on-line & off-line*

- **Accelerate queries**
  - ***Off-line pre-processing***
- **Each storage unit locally maintains a replica of the semantic vectors of all first-level index units to speed up the queries**
- **Lazy updating to deal with information staleness**

Key Issues: *on-line vs off-line*

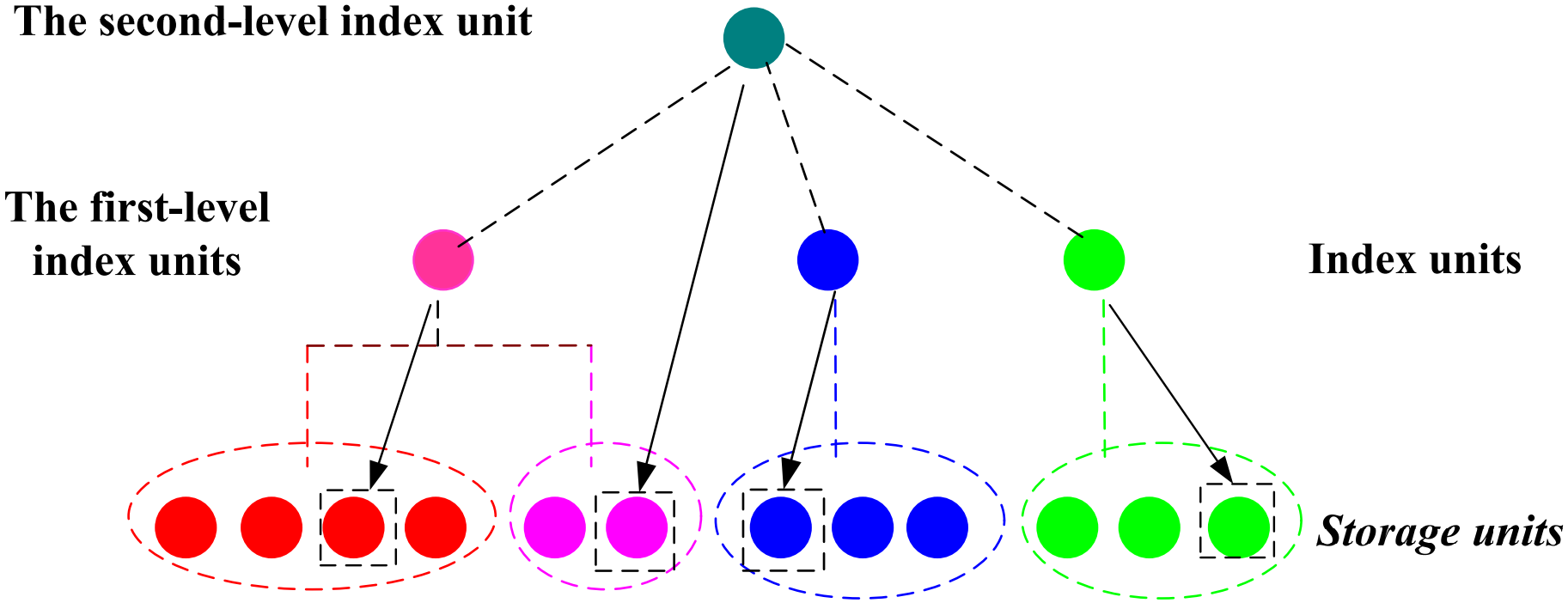


Key Issues: *Consistency Guarantee via Versioning*

- **Multi-replica technique can potentially lead to information staleness and inconsistency.**
- **Lazy Versioning:**
  - **A newly created version attached to its correlated replica temporarily contains aggregated real-time changes that have not been directly updated in the original replicas**
  - **SmartStore removes attached versions when reconfiguring index units**
  - **The frequency of reconfiguration depends on the user requirements and environment constraints**

Key issues: *Mapping of Index Units*

- **Our mapping is based on a simple bottom-up approach that iteratively applies random selection and labeling operations.**



Performance Evaluation

- **Prototype Implementation**
- **Large file system-level traces, including HP , MSN, and EECS by using *Trace Intensifying Factor***
- **Compared with typical *DBMS* and *R-tree***
  - ***Query latency reduction: 1000 times***
  - ***Space savings: 20 times***

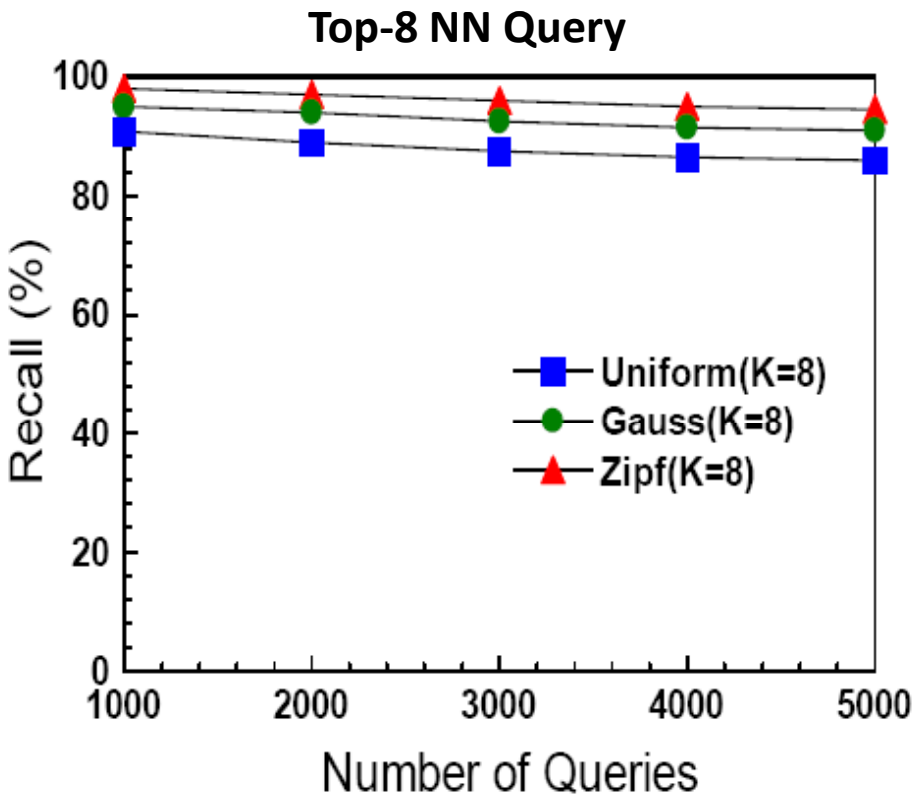
Complex Queries Latency

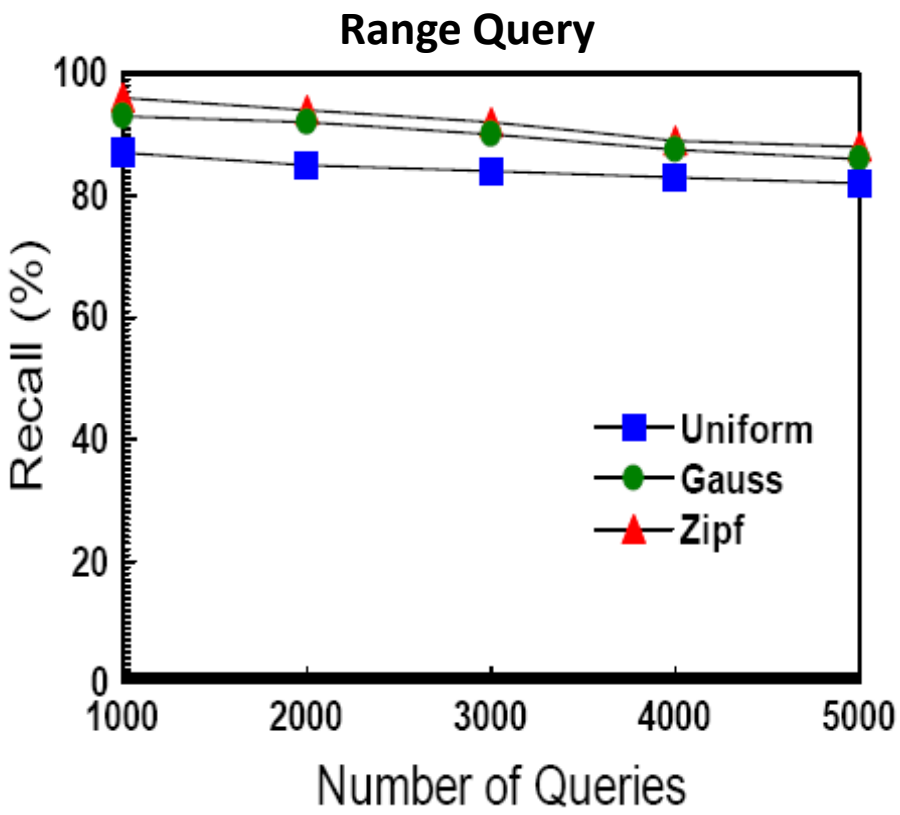
Query Types	TIF	*MSN* Trace			*EECS* Trace		
		DBMS	R-tree	SmartStore	DBMS	R-tree	SmartStore
Point Query	120	146.7	32.6	0.108	26.4	8.6	0.074
	160	378.6	122.5	0.179	168.9	42.1	0.136
Range Query	120	1516.5	242.5	1.63	685.2	126.3	1.56
	160	3529.6	625.7	3.41	1859.1	293.1	2.87
Top-k Query	120	4651.8	492.5	2.48	2076.1	196.8	2.25
	160	11524.6	1528.4	4.02	6519.3	571.7	3.47

Preliminary Simulation Results

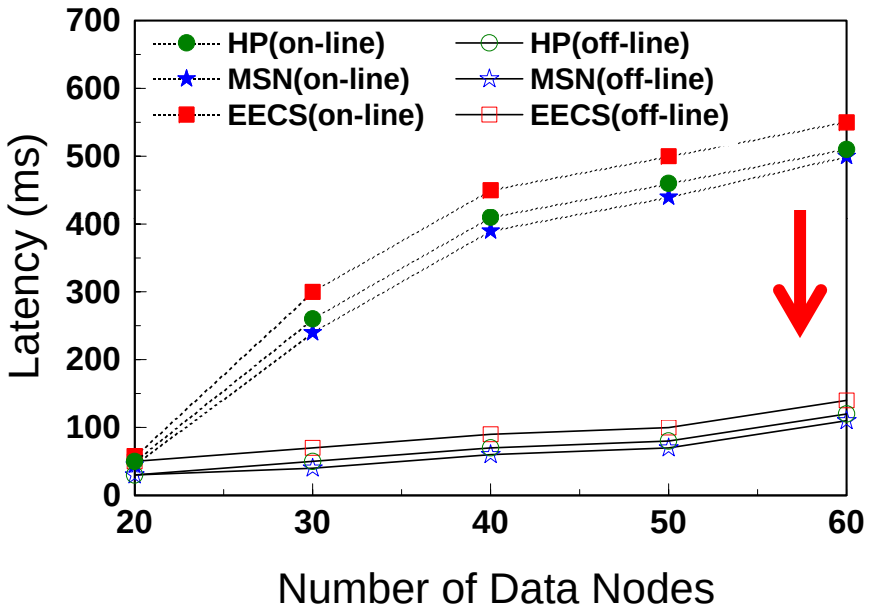
$$recall = \frac{|T(q) \cap A(q)|}{T(q)}$$

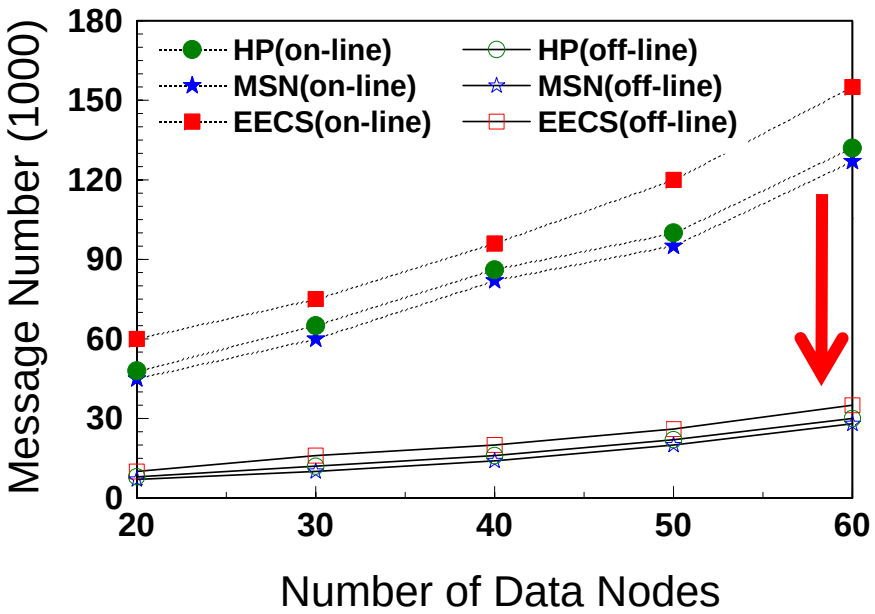
- **T(q) is the ideal answer for query q**
- **A(q) is the actual query results**





On-line & off-line

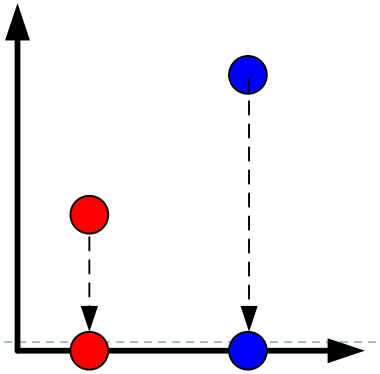




Discussion

- **SmartStore does work for:**
  - ***Pay-only-once: configuration efficiency for a long time due to complexity for semantic analysis;***
  - ***Rich semantics of multi-dimensional attributes to guarantee the groups to match access patterns well***

- **SmartStore does not efficiently work for:**
  - ***Lack of semantics, such as uniform distribution;***
  - ***Quick and dynamic evolution of semantics;***
  - ***Explicit scatter of dimension increments;***



Potential Applications

- **Users' views**
  - **Range query and top-k query**
- **System views**
  - **De-duplication**
  - **Caching**
  - **Pre-fetching**

Conclusions

- **SmartStore is a new paradigm for organizing file metadata for next-generation file systems**
  - ***Exploit file semantics***
  - ***Complex queries***
  - ***Enhance system scalability and functionality.***
- **Methodology**
  - ***Semantic aggregation***
  - ***Decrease search space***

Acknowledgement

- **This work is partially supported by**
- ***NSFC under Grant 60703046***
- ***National Basic Research 973 Program under Grant 2004CB318201***
- ***NSF CCF-0621526, NSF CCF-0937993, NSF CCF-0937988 and NSF CCF-0621493***
- ***HUST-SRF No.2007Q021B***
- ***The Program for Changjiang Scholars and Innovative Research Team in University No. IRT-0725.***

Thanks & Questions