Fast Indexing for Temporal Information Retrieval

Christian Rauch and Panagiotis Bouros

Institute of Computer Science, Johannes Gutenberg University Mainz, Germany

crauch@uni-mainz.de, bouros@uni-mainz.de

Temporal IR [1,2]

Setting

- Incorporate *temporal dynamics* in Information Retrieval
- Objects carry a $[t_{st},t_{end}]$ *time interval* and a *description d*
- *Challenges*: query analysis, ranking, clustering, *query processing and indexing*
- *Applications:* search for documents in archives, search multimedia databases, search basket data for market analysis

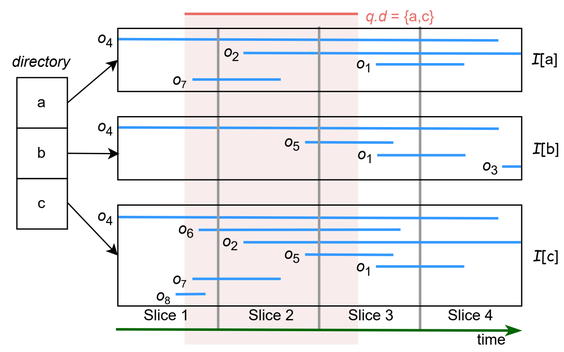
Search

- Blend *containment IR* search with *time-travel* search
- Query $q = <[q.t_{st},q.t_{end}],q.d>$
- Find all objects whose interval *overlaps* $[q.t_{st},q.t_{end}]$ and their description *contains all* elements in $q.d$

Temporal IR Indexing

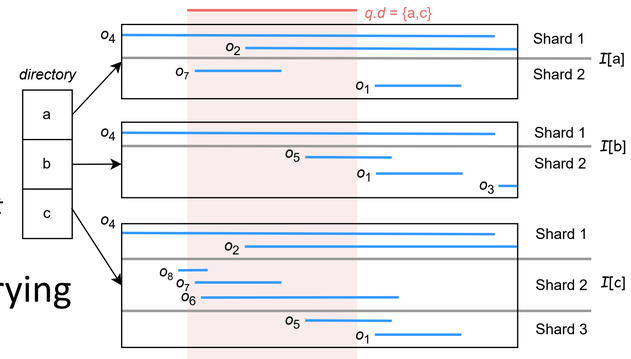
Slicing [3]

- Build a *temporal* inverted file
- *Split* time domain into *slices*
- *Partition* posting lists, *replicate* objects
- *Intersect* lists of $q.d$ elements for querying
- *Exclude* sub-lists of slices *not intersecting* query
- *Eliminate* duplicate results using [6]



Sharding [4]

- Build a *temporal* inverted file
- *Organize* posting lists into shards *sorted* by $o.t_{st}$
- Shards satisfy *staircase property*
- *Scan* shards of $q.d$ elements until $q.t_{end}$ for querying



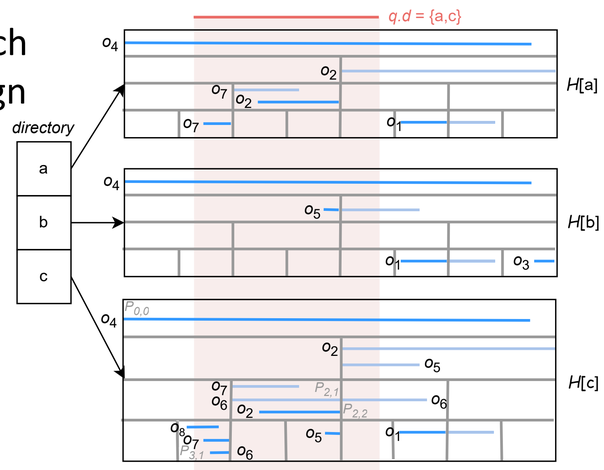
Our Goals

1. **Capitalize on modern, fast interval indexing state-of-the-art HINT [5]**
2. **Devise efficient time-first solutions**

IR-first Solution

tIF+HINT

- Build a *temporal* inverted file
- *Organize* every posting list as a HINT
- *IR-driven* query processing
- *Intersect* lists of $q.d$ elements
- *Temporal pruning* with HINT search
- Duplicate results *avoided* by design



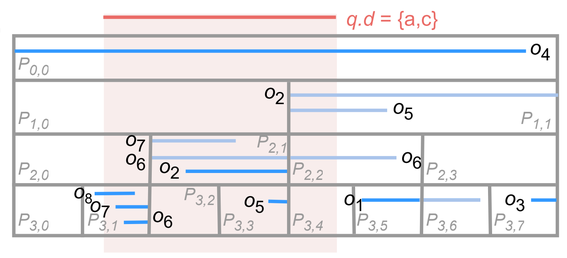
Variants

- *Binary search* or *merge-sort*
- *Hybrid* variant with Slicing

Time-first Solution

irHINT

- Build a *HINT* for the time domain
- *Organize* partitions as inverted files
- *Time-driven* query processing
- HINT *temporal* search
- *IR* search inside relevant partitions



Variants

- For *performance*
- For index *size*

element	$\mathcal{I}^O_{0,0}$	$\mathcal{I}^R_{1,1}$	$\mathcal{I}^O_{2,1}$	$\mathcal{I}^R_{2,1}$	$\mathcal{I}^R_{2,2}$
a	$\langle o_4,[\ldots]\rangle$	$\langle o_2,[\ldots]\rangle$	$\langle o_2,[\ldots]\rangle$	$\langle o_7,[\ldots]\rangle$	-
b	$\langle o_4,[\ldots]\rangle$	$\langle o_5,[\ldots]\rangle$	-	-	-
c	$\langle o_4,[\ldots]\rangle$	$\langle o_2,[\ldots]\rangle,\langle o_5,[\ldots]\rangle$	$\langle o_2,[\ldots]\rangle$	$\langle o_6,[\ldots]\rangle,\langle o_7,[\ldots]\rangle$	$\langle o_6,[\ldots]\rangle$

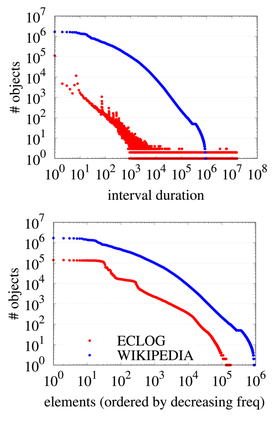
element	$\mathcal{I}^O_{3,1}$	$\mathcal{I}^O_{3,3}$	$\mathcal{I}^O_{3,5}$	$\mathcal{I}^R_{3,6}$	$\mathcal{I}^O_{3,7}$
a	$\langle o_7,[\ldots]\rangle$	-	$\langle o_1,[\ldots]\rangle$	$\langle o_1,[\ldots]\rangle$	$\langle o_7,[\ldots]\rangle$
b	-	$\langle o_5,[\ldots]\rangle$	$\langle o_1,[\ldots]\rangle$	$\langle o_1,[\ldots]\rangle$	-
c	$\langle o_6,[\ldots]\rangle,\langle o_7,[\ldots]\rangle,\langle o_8,[\ldots]\rangle$	$\langle o_5,[\ldots]\rangle$	$\langle o_1,[\ldots]\rangle$	$\langle o_1,[\ldots]\rangle$	$\langle o_7,[\ldots]\rangle$

Experiments

Setup

- Multiple query factors: *time interval extent*, *description size*, *selectivity* and *element frequency*
- Both *real* and *synthetic* datasets
- *Indexing*, *querying* and *maintenance* costs
- *Faster* tIF+HINT variant, *hybrid* with Slicing

	ECLOG	WIKIPEDIA
Cardinality	300311	1672662
Size [MBs]	171	4715
Time domain [secs]	15807599	126230391
Min. interval duration [secs]	1	1
Max. interval duration [secs]	15802098	126169456
Avg. interval duration [secs]	1325118	6587819
Avg. interval duration [%]	8.4	5.2
Dictionary size [# elements]	178478	927283
Min. description size [# elems]	1	1
Max. description size [# elems]	14399	6982
Avg. description size [# elems]	72	367
Min. element frequency	1	1
Max. element frequency	140423	1671696
Avg. element frequency	122	675
Avg. element frequency [%]	0.04	0.05



WIKIPEDIA (similar results for ECLOG)

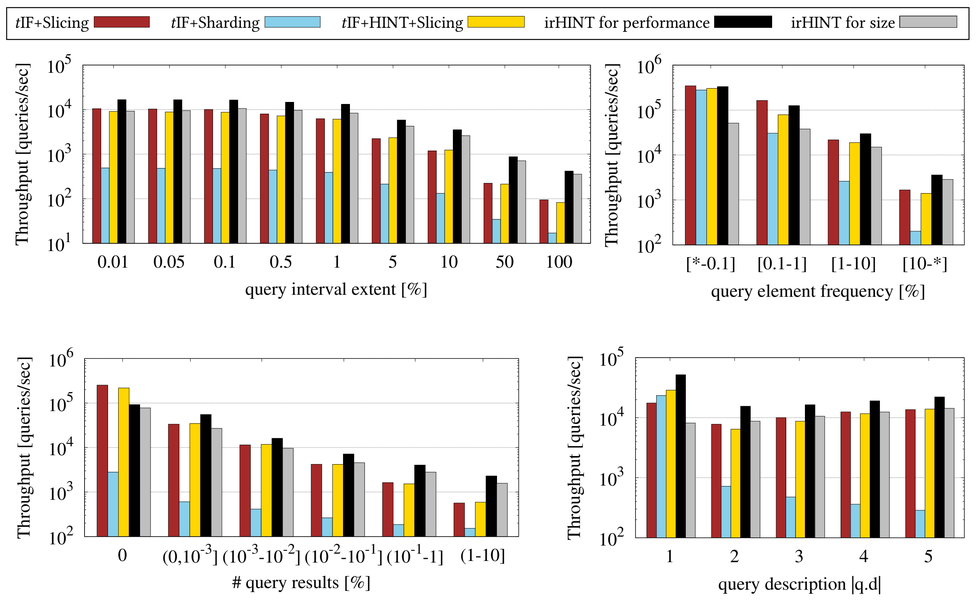
Indexing costs

index		time [secs]	size [MBs]
tIF+Slicing		86.8	19150
tIF+Sharding		288	7180
tIF+HINT	using binary search	529	21221
	using merge-sort	103	9740
	with Slicing	143	22507
irHINT	for performance	580	18022
	for size	594	7124

Maintenance costs: update time in secs

index		insertions			deletions		
		1%	5%	10%	1%	5%	10%
tIF+Slicing		1.16	6.10	11.8	6.43	32.5	65.9
tIF+Sharding		2.68	16.0	32.0	338	1707	3364
tIF+HINT	using binary search	7.23	38.5	76.2	6.62	33.8	67.7
	using merge-sort	1.85	9.97	19.7	4.77	24.2	48.3
	with Slicing	3.23	16.8	33.4	11.6	58.3	118
irHINT	for performance	3.41	17.8	34.1	9.22	46.4	96.3
	for size	5.31	28.0	54.0	15.9	78.4	156

Querying cost



References

[1] Wikipedia article, https://en.wikipedia.org/wiki/Temporal_information_retrieval

[2] N. Kanhabua, and A. Anand, Temporal Information Retrieval, ACM SIGIR, July 17-21, 2016

[3] K Berberich, S. J. Bedathur, T. Neumann, and G. Weikum, *A time machine for text search*, ACM SIGIR, July 23-27, 2007

[4] A. Anand, S. J. Bedathur, K. Berberich, and R. Schenkel, *Temporal index sharding for space-time efficiency in archive search*, ACM SIGIR, July 25-29, 2011

[5] G. Christodoulou, P. Bouros, and N. Mamoulis, *HINT: A Hierarchical Index for Intervals in Main Memory*, ACM SIGMOD, July 12-17, 2022

[6] J. Dittrich, and B. Seeger, *Data Redundancy and Duplicate Detection in Spatial Join Processing*, IEEE ICDE, February 28 – March 3, 2000

ACM SIGMOD/PODS International Conference on Management of Data, Bengaluru, India, May 31 – June 5, 2026