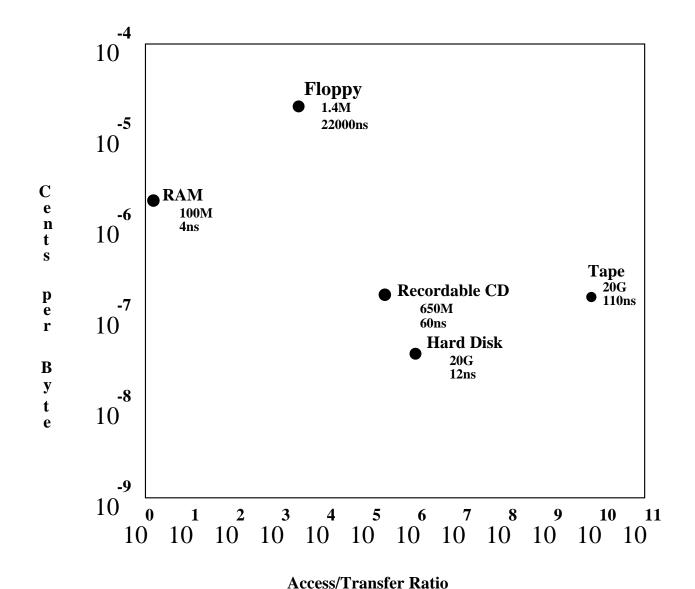
Copyright ©1998 Timothy Howard Merrett

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation in a prominent place. Copyright for components of this work owned by others than T. H. Merrett must be honoured. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or fee. Request permission to republish from: T. H. Merrett, School of Computer Science, McGill University, fax 514 398 3883.

The author gratefully acknowledges support from the taxpayers of Québec and of Canada who have paid his salary and research grants while this work was developed at McGill University, and from his students (who built the implementations and investigated the data structures and algorithms) and their funding agencies.

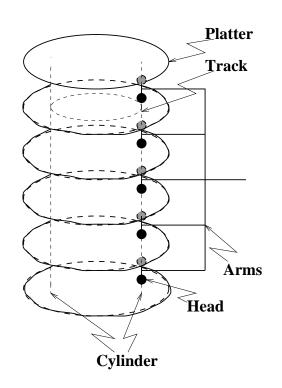
Exhibit 1.2.1 Cost vs. Access/Transfer Ratio for 1998 Memory Technology, Showing Typical Unit Capacities and Transfer Times per Byte.



The Difference Between RAM and SS

e.g., Disks

- have been around a long time
- recently improved faster than other technologies



- Latency @ 7200 rpm: $\lambda = 1/(2 \times 120)$ ~ 4 ms.
- Arm movement, say, $\mu = 8$ ms.
- Access time $\lambda + \mu = 12$ ms.
- With, say, $\beta = 100,000$ bytes/track, transfer is $1/\tau = 12$ Mbytes/sec

So access/transfer $\rho = (\lambda + \mu)/\tau = 144,000$ (bytes which could have been transferred while seeking the data).

T. H. Merrett

Exhibit 1.2.2 Specifications for Magnetic Disk Units to be Used in This Book (revised 1998).

`		au		a		σ	-
		TRANSFER		ACCESS/		ROTĂ	
		TIME/BYTE		TRANSFER		SPE	
				RATIO		0 . –	
DISK2000		8.3nsec.		1,440,000		7200	rpm
RCD2000		$0.7\mu \text{sec.}$		240,000		600-30	
FLOPPY2000		$22\mu \mathrm{sec}$.		4,000		600	
. 2011 . 2000		,		•			•
Ī		$I \qquad \lambda$		11		/	3
		AVERAGE	Λ	$_{VERAC}^{\mu}$	S E	RVT	, ==
		LATENCY		ARM DELAY		BYTES/ TRACK	
DISK2000		4.2ms		7.8ms			
RCD2000		75ms		95ms		1,000,000 140,000–280,000	
FLOPPY200	00	50ms		38ms		4608	
. 2011 1200		001110		001110		. 0	
		I				$T \cap T \Lambda$	•
		$\frac{\gamma}{1}$		ν		TOTAL	
		,		CYLINDERS		CAPACI	I Y
DICKOOO		CYLINDER		/UNIT 20000		20GB	
DISK200		10		3095		650MB	
RCD2000		$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$		160		1.4MB	
FLOPPY2000		4		100		1.41016)
		au	_	ρ	,	σ	
		RANSFER		CCESS		TAPE	
		IME/BYTE	TR	ANSFE	ER S	SPEED	
TAPE2000		12.5nsec.		20 G		80 ips	
		λ				δ	
	A'	VERAGE R	EW]	ND F	RECO	RDING	
		ATENCY	TIM			SITY	
TAPE2000		25 sec.	25se	C.	1 N	lBpi -	
ĺ		ι		ϕ	Т	OTAL	
		INTER-	Т	APE		PACITY	
	В	LOCK GAP		NGTH			
TAPE2000		0.2 in.		50 ft.	4	0 GB	
'							

T. H. Merrett

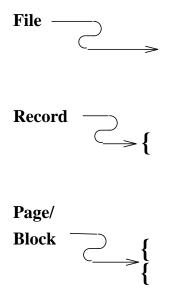
©98/9 4

Course Overview

- Access Complexity
 - Sequential (linear)
 - (Tree) Logarithmic
 - Direct (constant)
- Activity (ratio, records needed / total records)
- Volatility (add, delete, change records)
- Symmetry (all fields are equal?)

Sequential Files

Ordered



Ord#	Cust	Sales
1	GNS	E
2	NYC	N
3	L&S	E
4	PR	Н
5	NYC	Н
6	B&0	Н
7	GTRC	N
8	GNS	E

N =
8 records
n = 4 blocks
blocksize
b = 2
load factor
$\alpha = 1$
$n = \lceil \frac{N}{h\alpha} \rceil$

Unordered

Average cost of a successful search: n/2 accesses.

Sequential files are best for high activity.

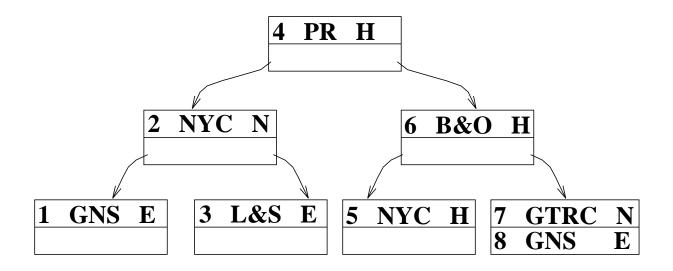
i.e. $>\sim 0.1\%$ of records accessed.

T. H. Merrett

©98/9

Logarithmic Files

e.g., B-trees



Average cost of a successful search: log n accesses.

e.g.,
$$n = 6$$

$$\lceil \log_2 n \rceil = 3$$

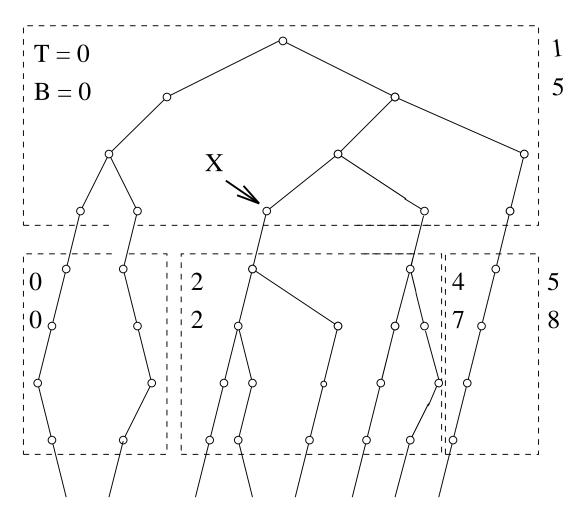
B-trees are very flexible, good for dynamic data

Tries

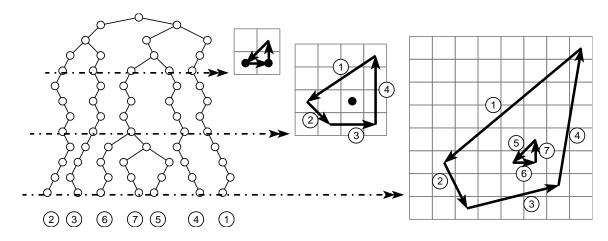
(Digital trees

Information retrieval)

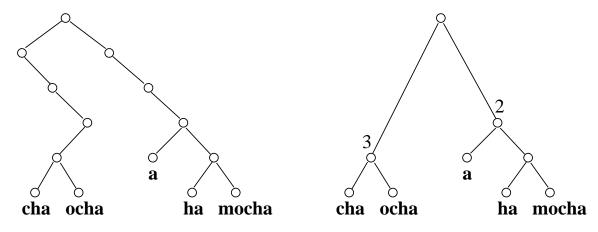
Sample data:



Kd-Tries and Variable Resolution



Truncated Tries and Text Data



1) Truncated Trie

2) PATRICIA Trie

Sample "text":

mocha: 111011011011111011000111110100011100001 with "starts" every eight bits.

Direct Access Files

e.g., Multipaging

Average cost of a successful search: 1 access.

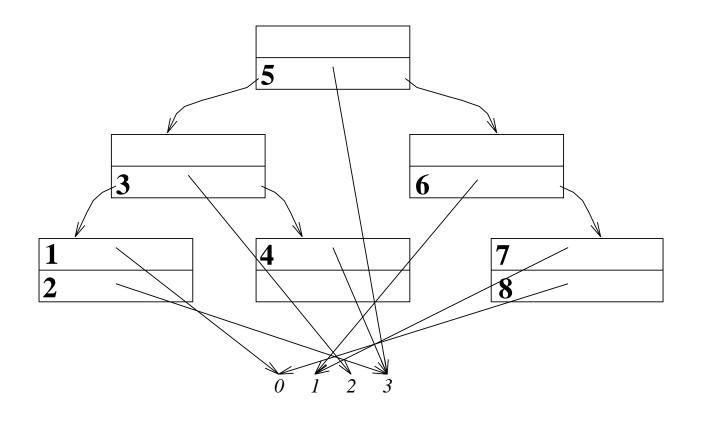
Order-preserving, thus good for high activity.

Can be built up dynamically.

B&O		6	
GNS	1,8		
GTRC			7
L&S	3		
NYC		5	2
PR		4	
	Е	Н	N

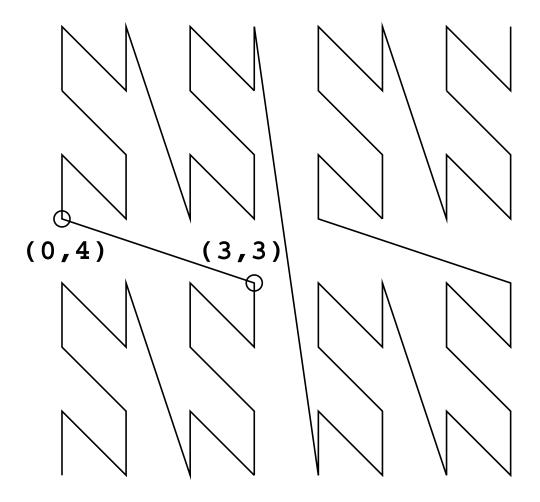
T. H. Merrett

Hybrid Files



B&O		0	6	1
GNS	1,8	3		
GTRC				7
L&S	3	2		3
NYC			5	2
PR			4	
	Е		Н	Ν

Z-Ordering



1-dimensional ordering of m-dimensional data So can use existing structures (e.g., B-tree)

Based on kd-trie, or on interleaving of bits:

$$(3,3)=(0011,0011)$$
 shuffles to $0000111 < 00010000$ unshuffles to $(0000,0100)=(0,4)$

T. H. Merrett