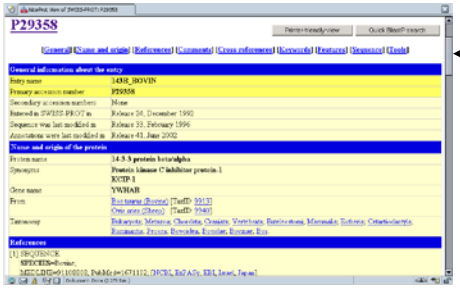
Database Operations on Modern Hardware

Ulf Leser, Stefan Sprenger

Three Tier Architectures



Servlets/
EJB

Presentation

Upwards: OO Interface
Application Server
Downwards: SQL

Application logic
“Business processes”

DBMS

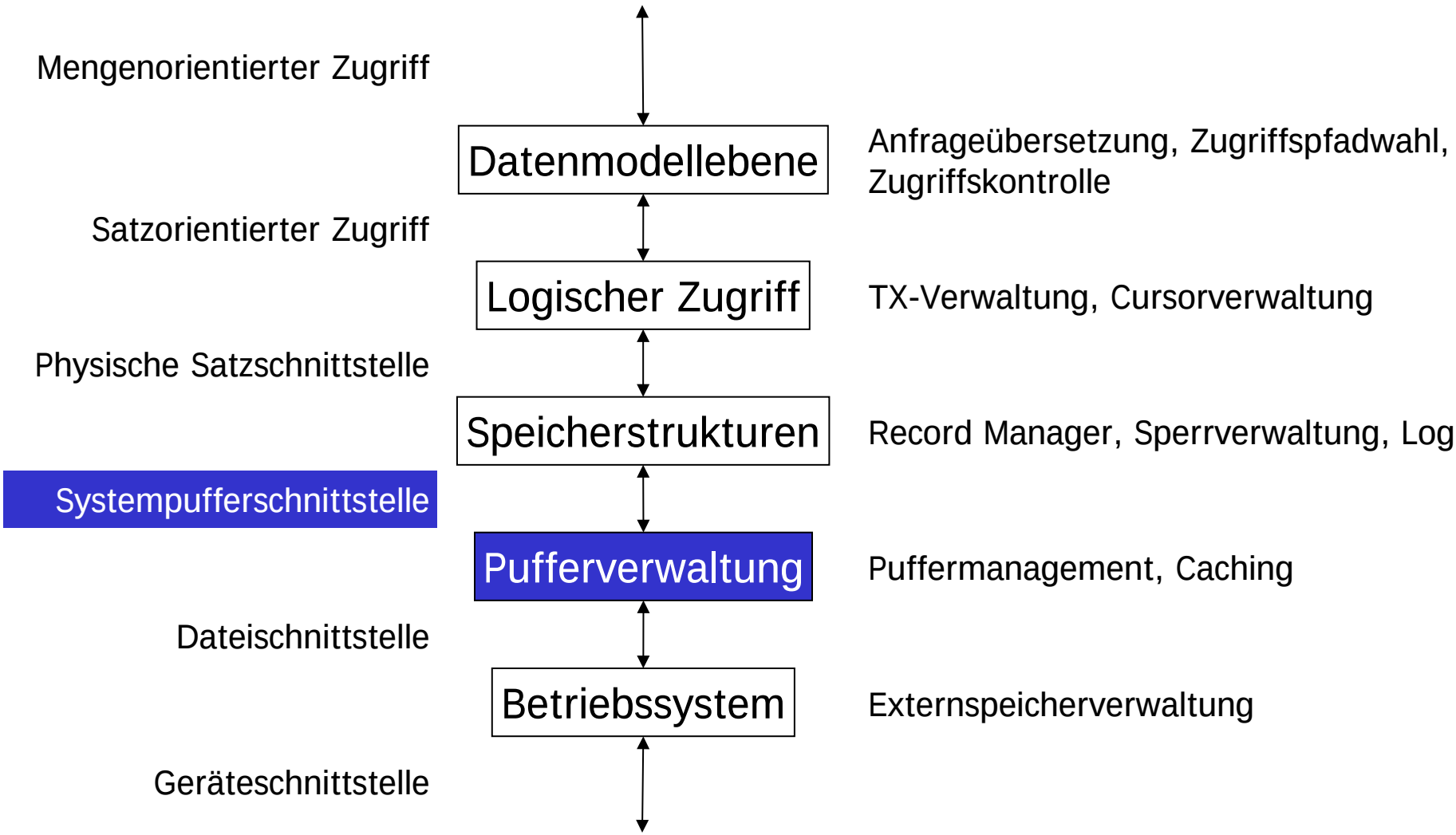
Data management:
queries, model

DB Files 1

DB Files 2

“State”: storage,
recovery, archival

5 Schichten Architektur



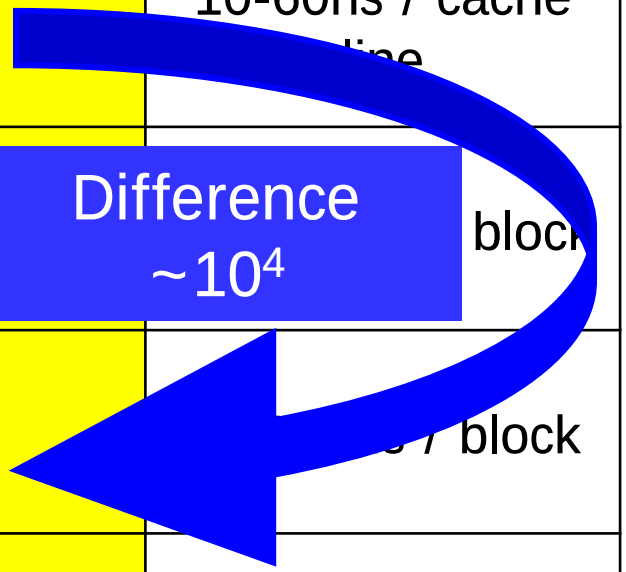
Bottlenecks

- Minimize IO – anything else is negligible cheap

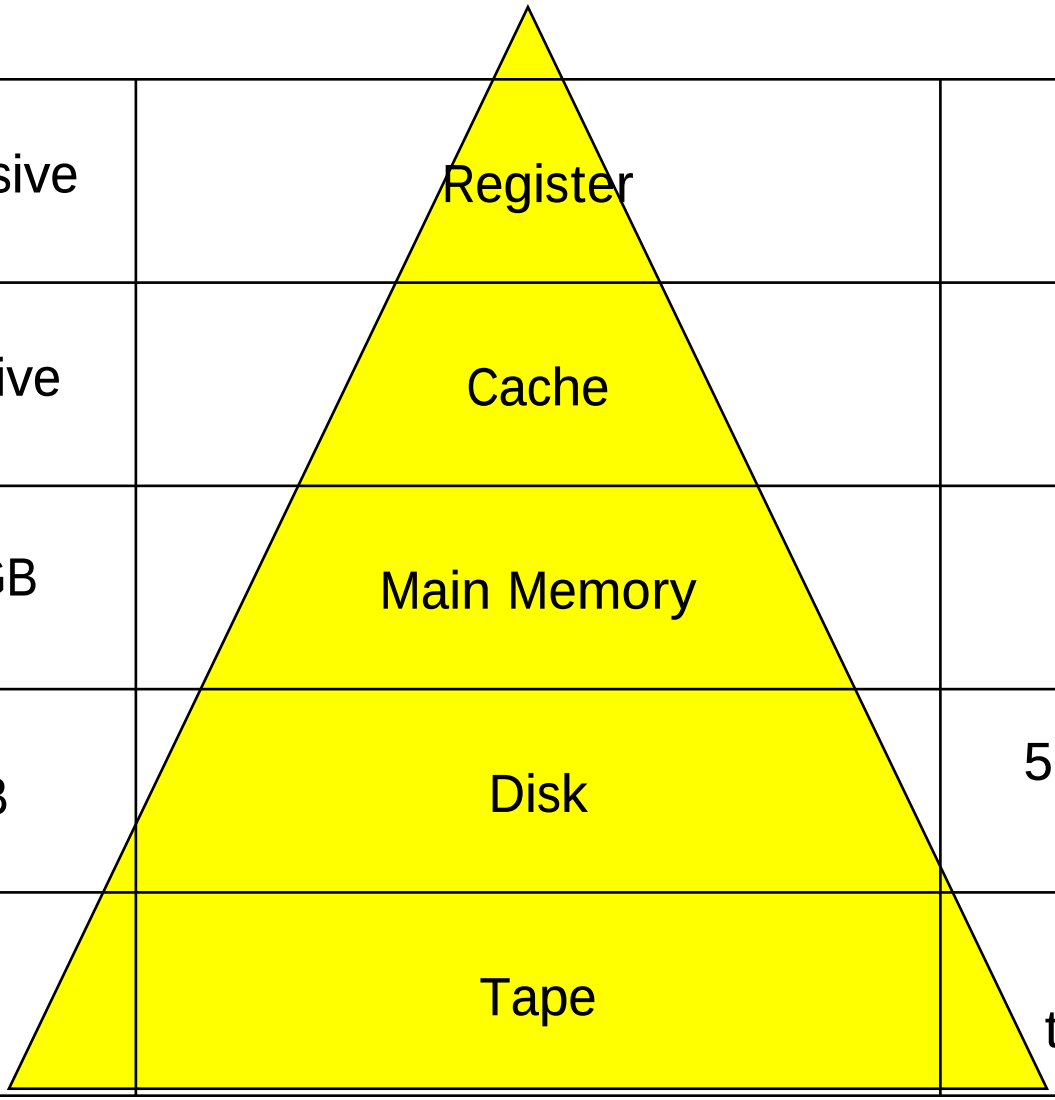
- Caching, prefetching, buffering, ...
- Optimize for small intermediate results
- Indirect memory access through page directory
- Avoid random access wherever possible
- ...

Price versus speed

Really expensive	Register	1-10ns / byte
Very expensive	Cache	10-60ns / cache
~ 200 € / GB	Main Memory	bloc
~ 1 € / GB	Disk	/ block
< 1€/GB	Tape	sec – min

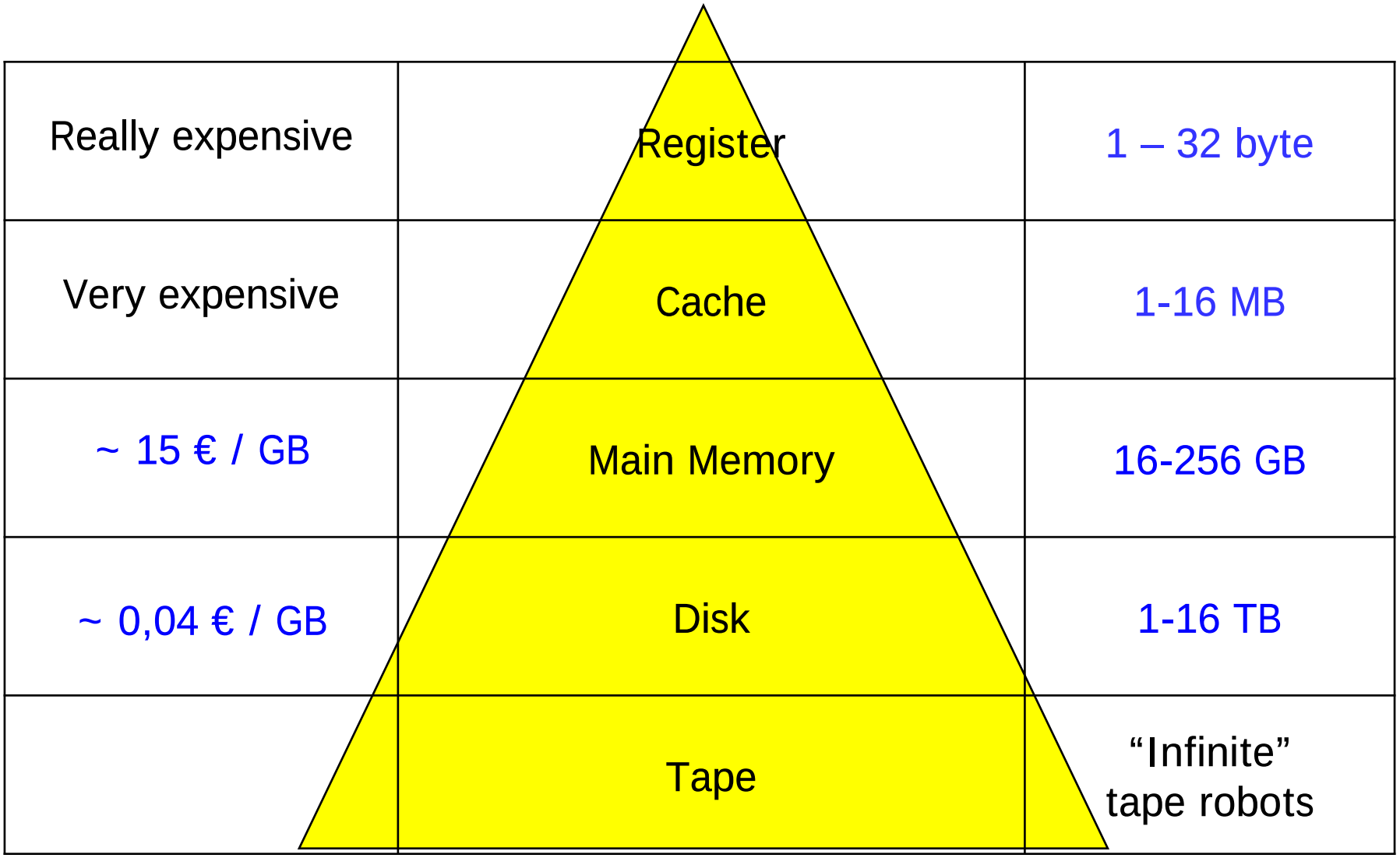


Storage Hierarchy

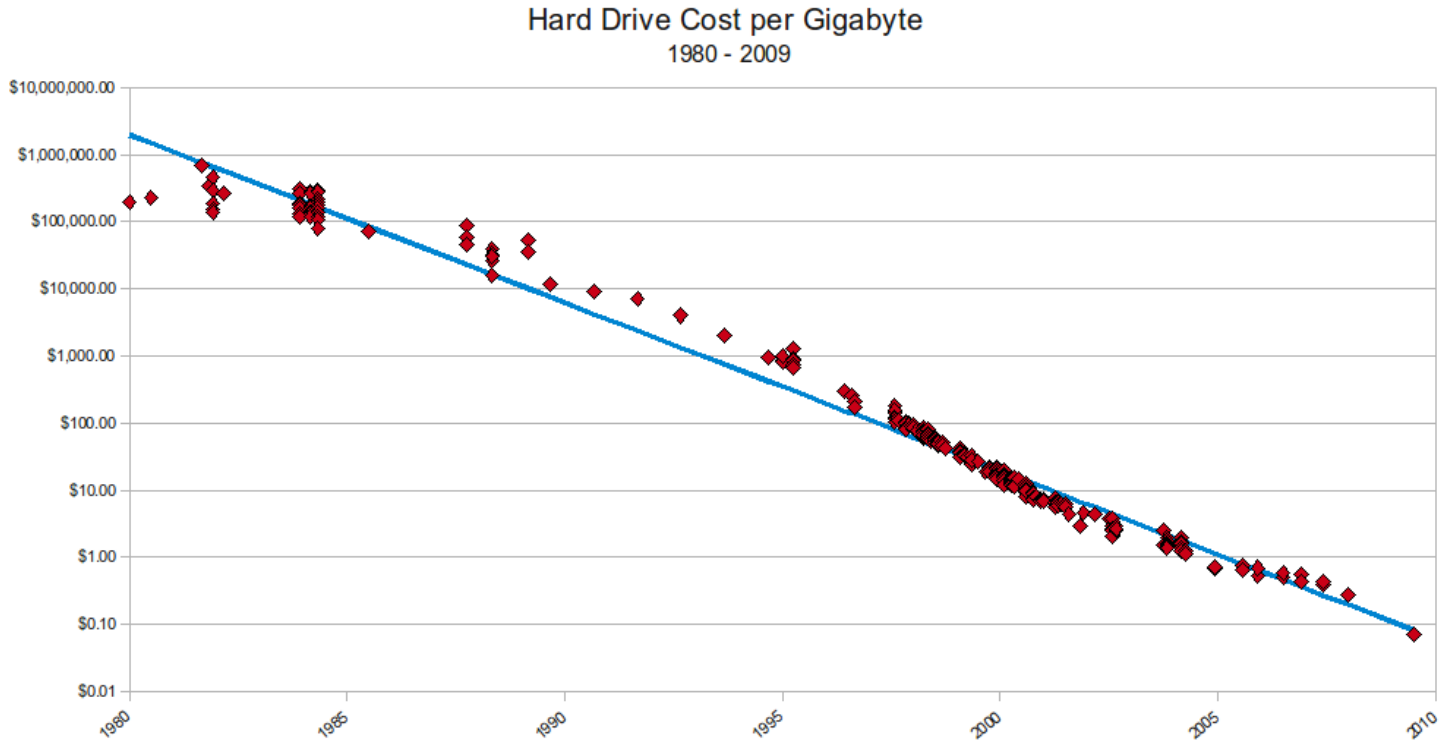


Really expensive	Register	1-4 byte
Very expensive	Cache	1-4 MB
~ 200 € / GB	Main Memory	1-16 GB
~ 1 € / GB	Disk	512GB – 1TB discs
< 1€/GB	Tape	“Infinite” tape robots

Storage Hierarchy Today

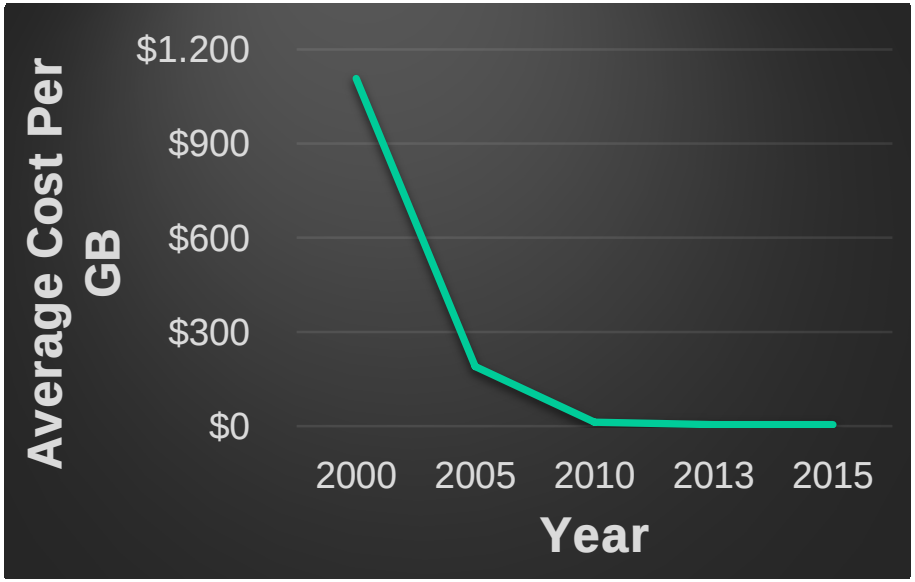


Costs Drop Faster than you Think



Source: http://analystfundamentals.com/?p=88

New: Prize of Main Memory



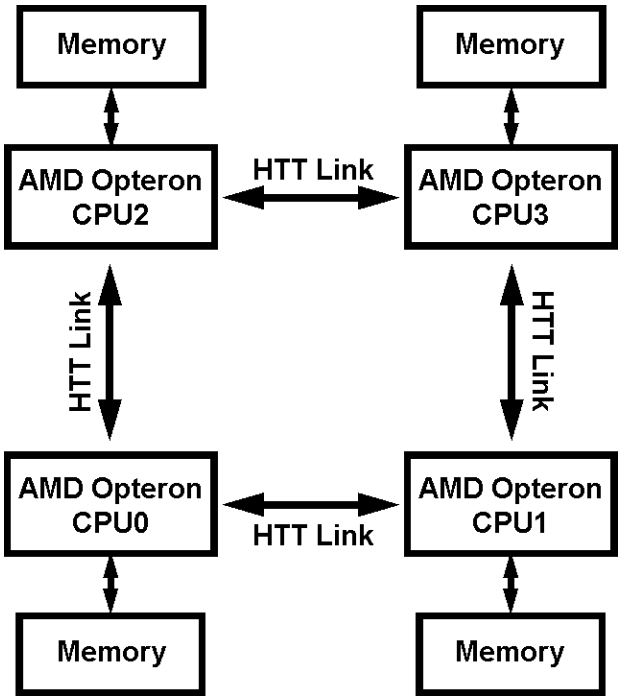
- 1TB main memory ~ 5000€
- Now: Laptops with 16GB, desktops with 32GB, servers with 128GB
- Guess 99% of all commercial databases are smaller than 100GB

New: High Performance CPUs (Here: Intel)

	Core	Techn.	Intro.	High performance MP server processor lines	Core count
Truland MP	Pentium 4 MP Prescott	90 nm	3/2005	90 nm Pentium 4 MP (Potomac)	1C
	Pentium 4 Presc.	90 nm	11/2005	7000 (Paxville MP)	2x1C
	Pentium 4 Presc.	65 nm	8/2006	7100 (Tulsa)	2x1C
Caneland MP	Core2	65 nm	9/2007	7200 (Tigerton DC) 7300 (Tigerton QC)	2C 2x2C
	Penryn	45 nm	9/2008	7400 (Dunnington)	6C
Boxboro-EX	Nehalem	45 nm	3/2010	7500 (Beckton/ Nehalem-EX)	8C
	Westmere	32 nm	4/2011	E7-4800 (Westmere-EX)	10C
Brickland	Ivy Bridge	22 nm	2/2014	E7-4800 v2 (Ivy Bridge-EX)	15C
	Haswell	22 nm	5/2015	E7-4800 v3 (Haswell-EX)	14C
	Broadwell	14 nm	??	E7-4800 v4 (Broadwell-EX)	24C
Purley	Skylake	14 nm	2017??	n.a. (Skylake-EX)	28C

New: Multi-Core with NUMA

- Modern CPUs can easily have 4-8 cores, each 2 threads
- 4 CPUs in one server is standard
- Add hyperthreading
- 128 hardware threads
- Future: Servers with 1000+ threads (exascale)
  - Network on a chip: Caching, routing, ...



Quelle: http://ixbtlabs.com/articles2/cpu/rmma-numa2.html

New: Processing beyond CPUs

- Specialized hardware is flourishing

- GPU: Fast, highly parallel, vectorized operations
- FPGA: Programmable hardware, low clock-speed
- SSD: In between DRAM and disk, fast random access
- Non-volatile RAM: Slower than DRAM, much faster than disk, highly asymmetric read/write performance
- ...

Challenges

- Utilize all available hardware
- Memory management with different types of memories and processing units
- Avoid cache misses (memory layout)
- Consider compute time (query compiler)
- Parallelize everything (programming models)
- Ensure consistency and durability (non-volatile)
- ...

Commercial Systems

	Partitioned	Multi-Versioned	Row/Columnar
Hekaton	No	Yes	Row
HyPer	No	Yes	Hybrid
SAP HANA	No	Yes	Hybrid
H-Store/VoltDB	Yes	No	Row

DB on Modern Hardware

- Old models are not adequate any more
  - Most DBs fit into main memory
  - New bottlenecks: Cache lines, computational cost of ops
  - Parallelization is the norm, not the exception
- DB on Modern Hardware
  - Modern CPU feature (pipelining, vectorization)
  - Many cores
  - Exploitation of specialized hardware
  - Main-memory index structures
  - Hardware-level transaction support

Who should be here

- Master Informatik
  - Also: Wirtschaftsinformatik, Ms.Edu, Diplominformatik
- Ability to read English papers
- Knowledge in computer architecture and operation systems
  - CPUs, memory hierarchy, caching & scheduling ...
- Good knowledge in databases
  - Cost models, optimization, index structures, ...

How it will work

- Today: Introduction and choice of topics
- Meet advisor before 30.11.16 to discuss topic and papers
- Present topic in 5min flash-presentation before Christmas
- Meet your advisor before 20.1.17 to discuss slides
- Present your topic (30-40min) at the Blockseminar
- Write seminar thesis (~15 pages) by 31.3.2017

ToC

- Introduction
- Topics
- Assignment
- Hints on presenting your topic and writing your thesis

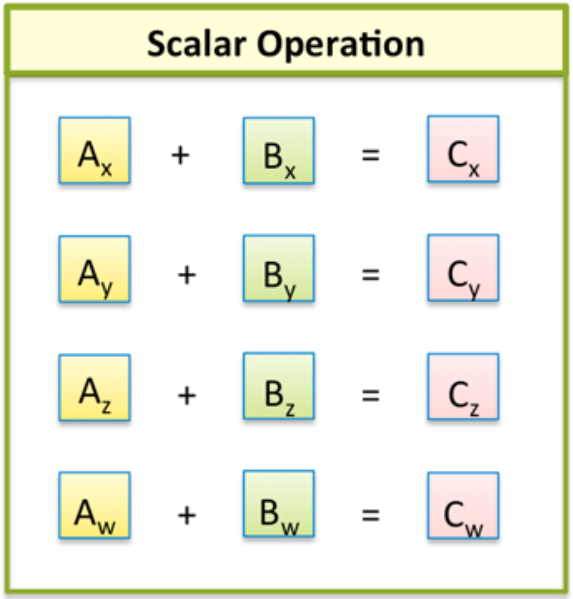
General Literature

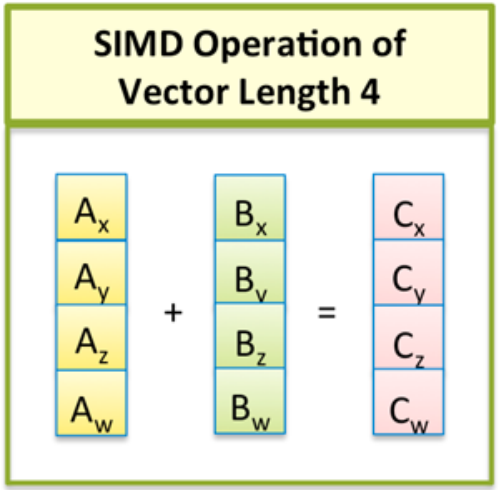
- To be read by everyone!
- A good database implementation book
- Larson, Paul and Levandoski, Justin: Modern Main-Memory Database Systems, VLDB Tutorial, 2016.
- Zhang, Hao et al.: "In-Memory Big Data Management and Processing: A Survey", IEEE Trans. Knowl. Data Eng. 27(7), 2015.

Topic	Assigned to	Supervision
Vectorized Instructions		Sprenger
Main Memory Index Structures		Leser
Many-Core CPUs		Leser
NUMA-aware algorithms		Leser
FPGAs		Sprenger
GPUs		Sprenger
Hardware transactional memory		Leser
Systems: MonetDB, HyPer, HANA		Sprenger

Vectorized Instructions (SIMD)

- Single Instruction Multiple Data
- Execute one instruction on multiple data elements in parallel





Intel® Architecture currently has SIMD operations of vector length 4, 8, 16
Quelle: https://01.org/node/1495

- Polychroniou, Orestis et al.: "Rethinking SIMD Vectorization for In-Memory Databases", SIGMOD, 2015.
- Polychroniou, Orestis and Ross, Kenneth A.: "Vectorized Bloom Filters for Advanced SIMD Processors", DaMoN, 2014.
- Polychroniou, Orestis and Ross, Kenneth A.: "High Throughput Heavy Hitter Aggregation for Modern SIMD Processors", DaMoN, 2013.
- Willhalm, Thomas et al.: "SIMD-Scan: Ultra Fast in-Memory Table Scan using on- Chip Vector Processing Units", VLDB, 2009.

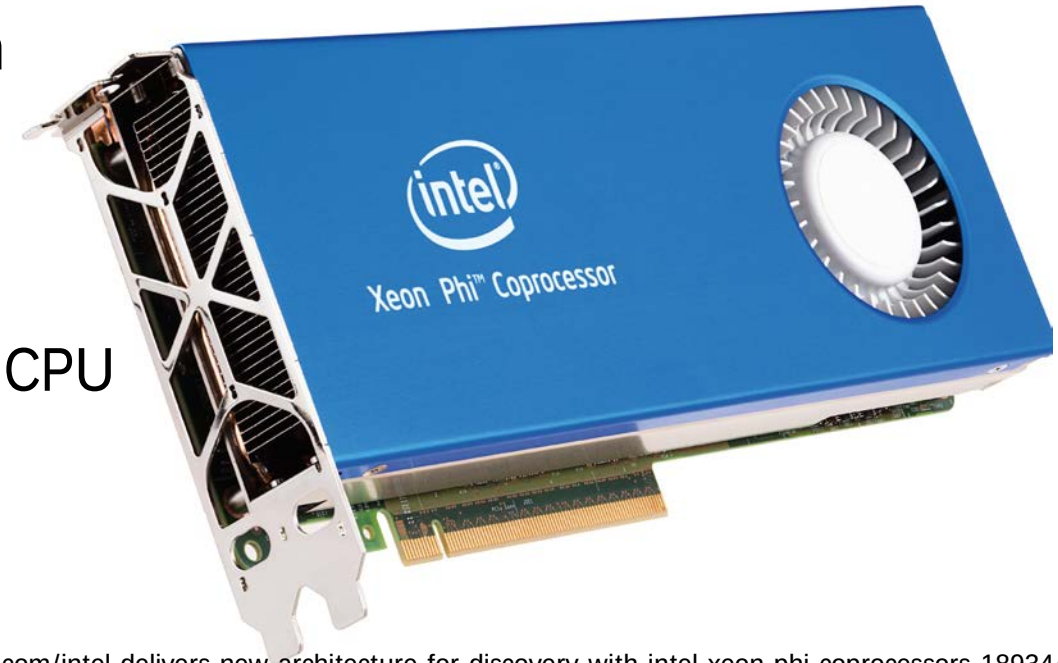
Main-Memory Index Structures

- Memory is cheap enough to hold entire databases
- Bottleneck has moved from main memory/disk up to CPU/main memory
- Optimize cache misses instead of page accesses
- Reduce wait time (CPU stalls) and maximize compute time
- Exploit features of modern CPUs
  - e.g., SIMD, Cache Lines, Pipelined Execution

- Leis, Viktor et al.: "The Adaptive Radix Tree: ARTful Indexing for Main-Memory Databases", ICDE, 2013.
- Kim, Changkyu et al.: "FAST: Fast Architecture Sensitive Tree Search on Modern CPUs and GPUs", SIGMOD, 2010.
- Rao, Jun and Ross, Kenneth A.: "Making B+-Trees Cache Conscious in Main Memory", SIGMOD, 2000

Many-Core CPUs

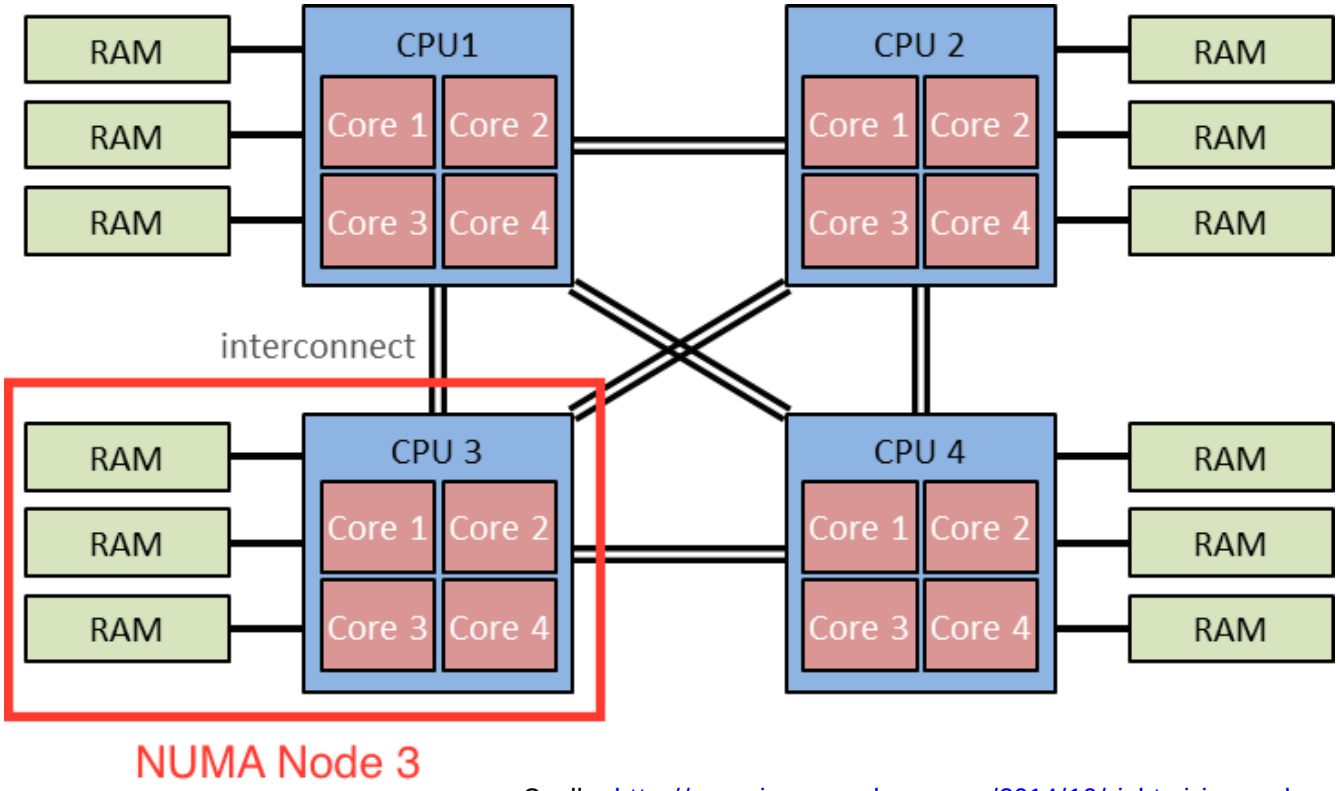
- Modern CPUs feature lots of cores
- High degree of parallelism
- Specialized coprocessors become standalone CPUs (Intel Xeon Phi, 2016)
- Up to 72 (!) cores on one CPU



Quelle: https://www.chiploco.com/intel-delivers-new-architecture-for-discovery-with-intel-xeon-phi-coprocessors-18934/

- Jha, Saurabh et al.: "Improving Main Memory Hash Joins on Intel Xeon Phi Processors: An Experimental Approach", VLDB, 2015.
- Balkesen, Cagri et al.: "Main-Memory Hash Joins on Multi-Core CPUs: Tuning to the Underlying Hardware", ICDE, 2013.
- Blanas, Spyros et al.: "Design and Evaluation of Main Memory Hash Join Algorithms for Multi-core CPUs", SIGMOD, 2011.

NUMA (Non-Uniform Memory Access)-aware algorithms

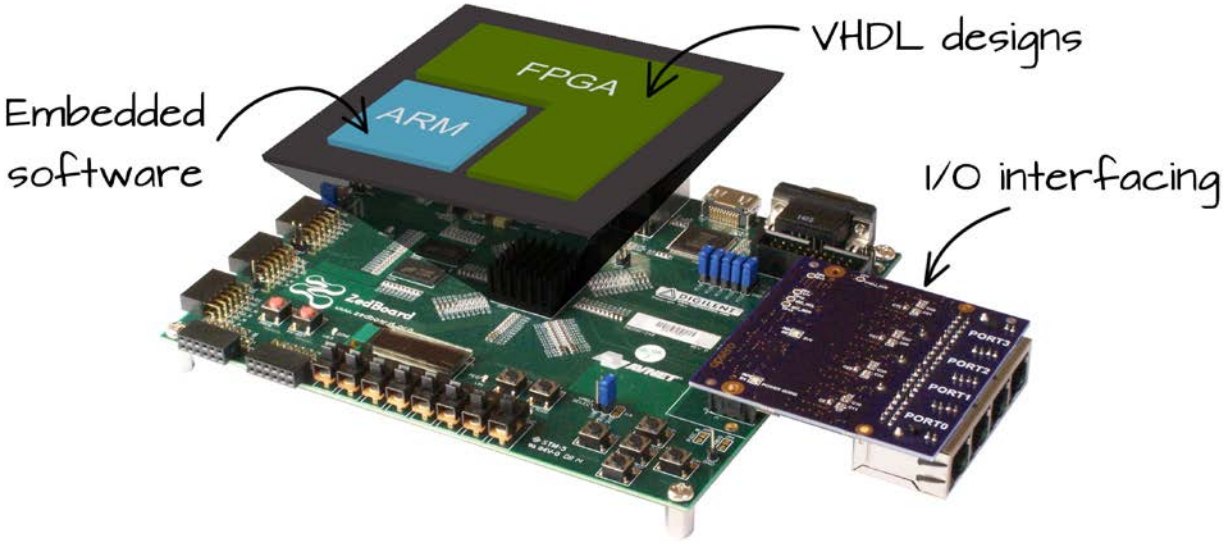


Quelle: http://www.jasongaudreau.com/2014/10/right-sizing-and-recertification-part-1.html

- Leis, Viktor et al.: "Morsel-driven parallelism: a NUMA-aware query evaluation framework for the many-core age", SIGMOD, 2014.
- Albutiu, Martina-Cezara et al.: "Massively Parallel Sort-Merge Joins in Main Memory Multi-Core Database Systems", PVLDB 5(10), 2012.
- Majo, Zoltan and Gross, Thomas R.: "Memory System Performance in a NUMA Multicore Multiprocessor", SYSTOR, 2011.

FPGAs (Field-Programmable Gate Array)

- Integrated circuit that can be configured
- Coprocessor that can be tailored to a certain application
- Chip consists of many logic gates that are wired by software



Quelle: https://opsero.com/fpga-programming/

- Halstead, Robert J. et al.: "FPGA-based Multithreading for In-Memory Hash Joins", CIDR, 2015.
- István, Zsolt et al.: "A flexible hash table design for 10GBPS key-value stores on FPGAS", FPL, 2013.
- Mueller, Rene et al.: "Data Processing on FPGAs", VLDB, 2009.
- Mueller, Rene and Teubner, Jens: "FPGA: What's in it for a Database?", SIGMOD, 2009.

GPUs

- Very high parallelism
- Up to thousands of threads
- SIMD on large data chunks
- Very high throughput
- Limited instruction set

Quelle: http://wccftech.com/nvidia-reportedly-preparing-geforce-titan-gpu-gk110/

- Sitaridi, Evangelia A. and Ross, Kenneth A.: "Optimizing select conditions on GPUs", DaMoN, 2013.
- Kaldewey, Tim et al.: "GPU join processing revisited", DaMoN, 2012.
- Bakkum, Peter and Skadron, Kevin: "Accelerating SQL Database Operations on a GPU with CUDA", GPGPU, 2010.

Hardware Transactional Memory (HTM)

- Modern CPUs, e.g., Intel's Skylake, provide transactional memory
- Beneficial for performance of atomic operations

```
Balance Transfer {
    Lock()
    A_balance -= 10
    B_balance += 10
    Unlock()
}
```

Using Global Lock

```
Balance Transfer {
    A.lock() B.lock()
    A_balance -= 10
    B_balance += 10
    B.unlock() A.unlock()
}
```

Using Fine Grained Locks

```
Balance Transfer {
     _xbegin()
    A_balance -= 10
    B_balance += 10
    xend()
}
```

Using HTM

Quelle: http://adms-conf.org/2016/HTMPresentation.pdf

- Cervini, David et al.: "Applying HTM to an OLTP system: No free lunch", DaMoN, 2015.
- Wang, Zhaoguo et al.: "Using restricted transactional memory to build a scalable in-memory database", EuroSys, 2014.
- Leis, Viktor et al.: "Exploiting Hardware Transactional Memory in Main-Memory Databases", ICDE, 2014.

Modern Database Systems

- S. Idreos et al.: "MonetDB: Two decades of research in column-oriented database architectures", IEEE Data Eng. Bull. 35(1), 2012.
- F. Färber et al.: "SAP HANA database: data management for modern business applications", SIGMOD Record 40(4), 2011.
- T. Neumann: "Efficiently compiling efficient query plans for modern hardware", VLDB, 2011.

ToC

- Introduction
- Topics
- Assignment
- Hints on presenting your topic and writing your thesis

Allgemeine Hinweise

- Dozenten sind ansprechbar!
  - Vorbesprechung des Themas
  - Folien durchgehen
  - Abgrenzung der Ausarbeitung
- Diskussion erwünscht
  - Keine Angst vor Fragen: Fragen sind keine Kritik
  - Eine Frage nicht beantworten können ist in Ordnung
- Tiefe, nicht Breite
  - Lieber das Thema einengen und dafür Details erklären
- Bezug nehmen
  - Vergleich zu anderen Arbeiten (im Seminar)

Allgemeine Hinweise

- Werten und bewerten
  - Keine Angst vor nicht ganz zutreffenden Aussagen – solange gute Gründe vorhanden sind
  - Begründen und argumentieren
  - Kritikloses Abschreiben ist fehl am Platz
- Literaturrecherche ist notwendig
  - Die ausgegebenen Arbeiten sind Anker
  - Weiterführende Arbeiten müssen herangezogen werden
  - Auch Grundlagen nachlesen

- Wir schicken eine Liste zum Abhaken rum

Wie halte ich einen Seminarvortrag

1. Wenn man nun so einen Seminarvortrag halten muss, dann empfiehlt es sich, möglichst lange Sätze auf die Folien zu schreiben, damit die Zuhörer nach dem Vortrag aus den Folienkopien noch wissen, was man eigentlich gesagt hat.

2. Während so einem Vortrag schaut sowieso jeder zum Projektor, also kann man das selbst ruhig auch tun - damit kontrolliert man gleichzeitig auch, ob der Beamer wirklich alles projeziert, was auf dem Laptop zu sehen ist. Ausserdem kann man so den Strom für das Laptop-Display sparen.

3. Übersichtsfolien am Anfang sind langweilig, enthalten keinen Inhalt und nehmen den Zuhörern die ganze Spannung. Schliesslich gibt‘s im Kino am Anfang auch keine Inhaltsangabe.

4. Powerpoint kann viele lustige Effekte, hat tolle Designs und Animationen. Die sollte man zur Auflockerung des Vortrags unbedingt alle benutzen, um zu zeigen, wie gut man das Tool im Griff hat.

5. Nicht zu wenig auf die Folien schreiben. Man weiß ja nie, ob man sie nicht doch ausdrucken muss, und man kann so wertvolle Zeit sparen, wenn man nicht weiterschalten muss.

6. Man sollte versuchen, möglichst lange zu reden. Die Zeitvorgaben sind nur für die Leute, die nicht genug wissen - eigentlich will der Prüfer sehen, dass man sich auch darüber hinaus mit dem Thema beschäftigt hat.

Bloß keine Hervorhebungen im Text – sonst müssen die Zuhörer ja gar nicht mehr aufpassen!

Hinweise zum Vortrag

- 30-40 Minuten plus Diskussion
- Klare Gliederung
- Ab und an Hinweise geben, wo man sich befindet
- Themenauswahl: Lieber verständlich als komplett
- Bilder und Grafiken; Beispiele
- Font: mind. 16pt
- Eher Stichwörter als lange Sätze
- Vorträge können auch unterhaltend sein
  - Gimmicks, Rhythmuswechsel, Einbeziehen der Zuhörer, etc.
- Adressat sind alle Teilnehmer, nicht nur die Betreuer
- Technik: Laptop? Powerpoint? Apple?

Hinweise zur Ausarbeitung

- Eine gedruckte Version abgeben
  - Selbstständigkeitserklärung unterschreiben
- Eine elektronische Version schicken
- Referenzen: Alle verwendeten und nur die
  - Im Text referenzieren, Liste am Schluss
- Korrekt zitieren
  - Vorsicht vor Übernahme von kompletten Textpassagen; wenn, dann deutlich kennzeichnen
  - Aussagen mit Evidenz oder Verweis auf Literatur versehen
- Verwendung von gefundenen Arbeiten im Web
  - Möglich, aber VORSICHT
  - Eventuell Themenschwerpunkt verschieben – Betreuer fragen

Hinweise zur Ausarbeitung –2-

- Gezielt und sachlich schreiben
- Füllwörter vermeiden (dabei, hierbei, dann, ...)
- Knappe Darlegung, präzise Sprache
- Eine gute Gliederung ist die halbe Miete
- Kommen Sie zu Aussagen
  - Vorteile, Nachteile, verwandte Arbeiten, mögliche Erweiterungen, Anwendbarkeit, eigene Erfahrungen, ...

Format

- Benutzung unserer Latex-Vorlage
- Nur eine Schriftart, wenig und konsistente Wechsel in Schriftgröße und –stärke
- Inhaltsverzeichnis
- Bilder: Nummerieren und darauf verweisen
- Referenzen:
  - [1] Yan, X., Yu, P. S. and Han, J. (2004). "Graph Indexing: A Frequent Structure-Based Approach". SIGMOD, Paris, France.
  - [YYH04] Yan, X., Yu, P. S. and Han, J. (2004). "Graph Indexing: A Frequent Structure-Based Approach". SIGMOD, Paris, France.
- Darf man Wikipedia zitieren?
  - Ja, aber nicht dauernd