



*Waste Not, Want Not! - Managing Relational Data in Asymmetric Memories*  
H. Pirk

# Waste Not, Want Not!

Managing Relational Data in Asymmetric Memories

Holger Pirk

## Abstract

Computer systems are not the monolithic machines they used to be. In the early days of computer science (until the late 70s), most computer systems included exactly one component to perform a given task: one (type of) disc for persistence, one CPU for processing and one volatile RAM to hold intermediate data. Today, the architecture has developed into a heterogeneous landscape of components: discs, SSDs, RAM, NVRAM, GPUs and CPUs with a hierarchy of caches – all working together to accomplish a given task. However, making efficient use of all of these devices is difficult: slow interconnects make communication and synchronization of these devices costly and motivate sophisticated co-operation strategies to minimize such communication. While developing an efficient cross-device co-operation strategy is far from trivial, there is a fundamental characteristic of computer hardware that can be exploited:

While memory component properties are asymmetric (fast&small vs. slow&large), so is data access (hot vs. cold, sequential vs random).

In this thesis, we study the management of relational data in modern, i.e., asymmetric computer systems. We explore different strategies to identify asymmetries in persistent data, map them to asymmetries in the memory landscape and, eventually, exploit them to increase query processing performance. To this end, we study memory conscious decomposition and storage of data at different granularities: relations, vertical partitions, single attributes as well as individual bits. In the interest of conciseness, we exclude techniques that require auxiliary data structures such as indices or horizontal partitioning which come with significant maintenance overhead.

Further, we argue that, when managing memory-resident data, the problem of optimal data placement is tightly connected to the efficiency of the query processing paradigm and can, therefore, not be studied in isolation. Consequently, we also investigate the connection between storage model and processing paradigm. In the case of decomposition at partition granularity we identify Just-in-Time compilation as the only viable query processing model. In the case of distribution at the granularity of individual bits, we develop a novel processing paradigm that efficiently exploits the asymmetries in the underlying data and memory components.

## Samenvatting

Computersystemen zijn niet de monolithische machines die ze vroeger waren. In het begin van de computerwetenschappen (tot het einde van de jaren '70) omvatten de meeste computersystemen precies n component voor elke taak: een soort disc voor het opslaan, een CPU voor het verwerken en een volatiel RAM voor het houden van intermediaire data. Tegenwoordig is de architectuur ontwikkeld tot een heterogeen landschap van componenten: discs, SSD's, RAM, NVRAM, GPU en CPU's met een hiërarchie van caches - die allemaal samenwerken om een bepaalde taak te volbrengen. Echter, efficiënt gebruik van al deze apparaten is moeilijk: trage interconnecties maken de communicatie en synchronisatie van deze apparaten duur en motiveren geavanceerde samenwerkingsstrategieën om de communicatie zoveel mogelijk te beperken. Terwijl het ontwikkelen van een efficiënt cross-device samenwerkingsstrategie niet triviaal is, is er een fundamentele eigenschap van computer hardware die kan worden geëxploiteerd:

Eigenschappen van geheugencomponenten zijn asymmetrisch (snel&klein vs. langzaam&groot), en dat geldt ook voor de toegang tot de gegevens (warm vs. koud, opeenvolgend vs. random).

In dit proefschrift onderzoeken wij het beheer van relationele data in de moderne, dat wil zeggen, asymmetrische computersystemen. Wij onderzoeken verschillende strategieën om asymmetrieën in data management applicaties te identificeren, deze met asymmetrieën in de componenten van het geheugen te matchen en deze uiteindelijk te gebruiken om de prestaties te verbeteren. Hiervoor onderzoeken wij de memory conscious decompositie en opslag van gegevens op verschillende granulariteiten: relaties, (verticale) partities, attributen en individuele bits.

Verder voeren wij aan dat het probleem van een optimale plaatsing van gegevens strak is verbonden met het paradigma van queryverwerking en om die reddend niet geïsoleerd kan worden onderzocht. Daarom hebben wij ook de samenhang onderzocht tussen het opslag-paradigma en het verwerking-paradigma. In het geval van decompositie op granulariteit van de partitie identificeren wij Just-in-Time compilatie als het enige rendabele query processing model. In het geval van een verdeling op de granulariteit van de individuele bits, ontwikkelen wij een nieuw verwerking-model dat efficiënt gebruik maakt van de asymmetrieën in de onderliggende gegevens en de componenten van het geheugen.