

# Entzauberung von Clustered Indexen

Datum: 12.03.2016

Autor: Uwe Ricken (db Berater GmbH)

# Uwe Ricken

db Berater GmbH



www: <http://www.db-berater.de>  
email: [uwe.ricken@db-berater.de](mailto:uwe.ricken@db-berater.de)  
blog: <http://www.sqlmaster.de>  
twitter: <https://twitter.com/@dbberater>  
xing: [http://www.xing.com/profile/Uwe\\_Ricken](http://www.xing.com/profile/Uwe_Ricken)

I am working with IT-systems since early 1990's and with the main focus on *Microsoft SQL Server* since version 6.0. I started with development of database applications in 1998 with a professional CRM-System based on Microsoft products (Microsoft Office and *Microsoft SQL Server*).

Since 2008 I'm focused exclusively on *Microsoft SQL Server* and since 2008 I'm working in 3rd level support teams for banks, insurances and global industries.

Since May 2013 I'm a **Microsoft Certified Master: SQL Server 2008** which was an amazing way into the depth of *Microsoft SQL Server*.

In July 2013 I have been awarded with the MVP Award for *Microsoft SQL Server*.

# Uwe Ricken

db Berater GmbH

**Microsoft**  
**CERTIFIED**

Master

SQL Server® 2008

**Microsoft**  
**CERTIFIED**

Solutions Expert

Data Platform



The Microsoft Certified Solutions Master (MCSM) and Microsoft Certified Master (MCM) programs recognize the most experienced IT professionals who demonstrate and validate their deep technical skills to successfully design and implement technical solutions in a complex business environment.

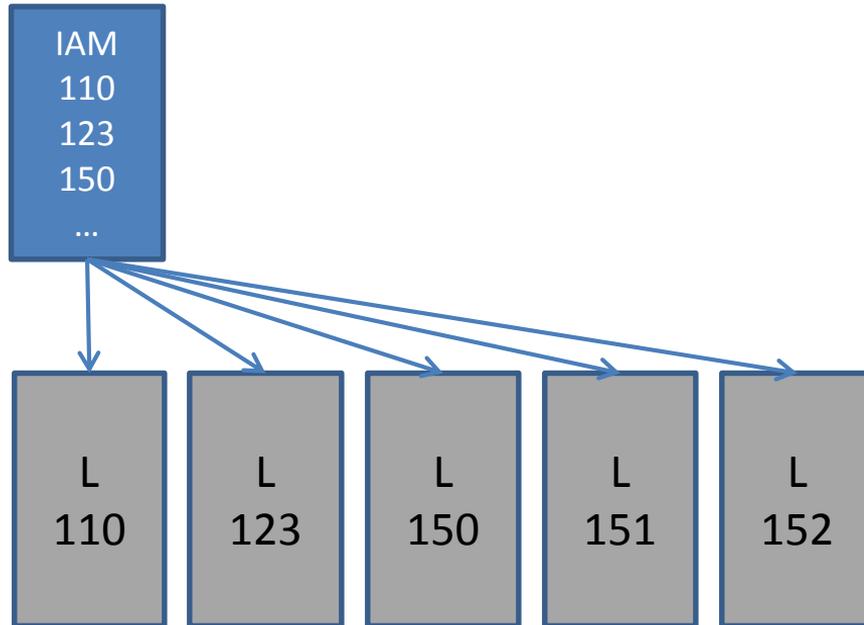
Microsoft Certified Solutions Expert (MCSE) certifications are designed to recognize IT professionals who can design and build solutions across multiple technologies, both on-premises and in the cloud.

The Microsoft Most Valuable Professional (MVP) Award is our way of saying thank you to exceptional, independent community leaders who share their passion, technical expertise, and real-world knowledge of Microsoft products with others.

# Agenda

- Was ist ein HEAP?
- Was ist ein CLUSTERED INDEX?
- Non Clustered Indexe in HEAPS und CLUSTERED INDEX
- HEAPS vs. CLUSTERED INDEX
  - INSERT, UPDATE, DELETE
  - SELECT
  - Maintenance

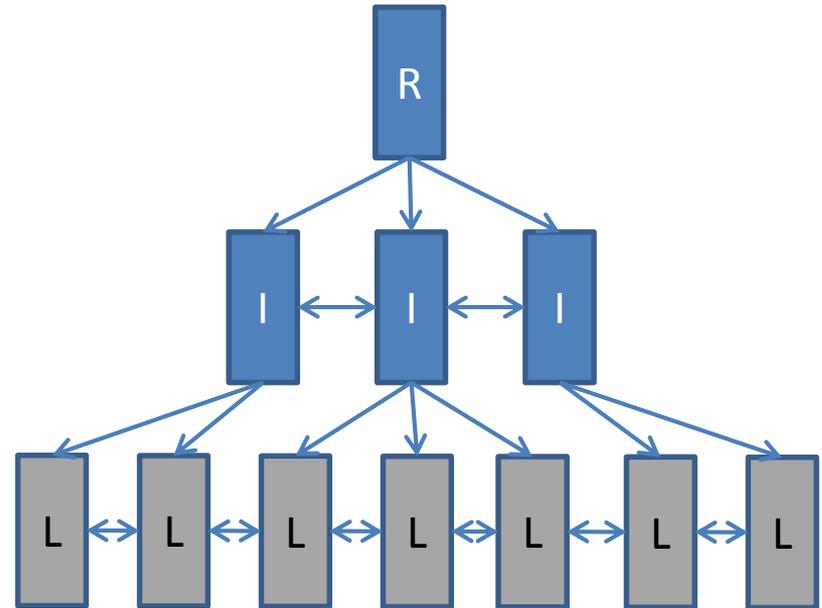
# Was ist ein Heap?



- Ein HEAP ist eine Tabelle ohne Ordnungskriterium.
- Zwischen den einzelnen Datenseiten besteht KEINE direkte Verbindung

# Was ist ein CLUSTERED INDEX

- Ein Clustered Index eine nach einem Ordnungskriterium logisch sortierte Tabelle.
- Die unterste Ebene (Leafs) beinhalten die Daten.
- Der Root Knoten (R) und die B-Tree-Ebenen (I) beinhalten Pointer auf Datenseiten, die sich in der nächsten Hierarchie befinden.
- Es kann nur EINEN Clustered Index pro Tabelle geben.



# DEMO-TIME!

# CLUSTERED vs. HEAP: SELECT

- Ein SELECT auf den CLUSTERED KEY ist effizienter, weil mittels INDEX SEEK gezielt auf die gewünschte ID gesprungen werden kann.
- Ein HEAP benötigt IMMER einen TABLE SCAN, da kein Index vorhanden ist!

# CLUSTERED vs. HEAP: SELECT

- Wird mit einem TOP-Operator gearbeitet, KANN die Suche in einem HEAP effizienter sein, wenn sich der zu suchende Wert gleich am Anfang der Tabelle befindet
- Dennoch kann hier der Clustered Index klar punkten!

# CLUSTERED vs. HEAP: SELECT

- Gruppierungsabfragen OHNE weitere Indexe führen in der Regel zu besserem Verhalten in HEAPS!
- Ein HEAP hat nur eine Ebene (Leaf), die durchlaufen werden muss
- Ein CLUSTERED INDEX muss den B-Tree mit scannen.

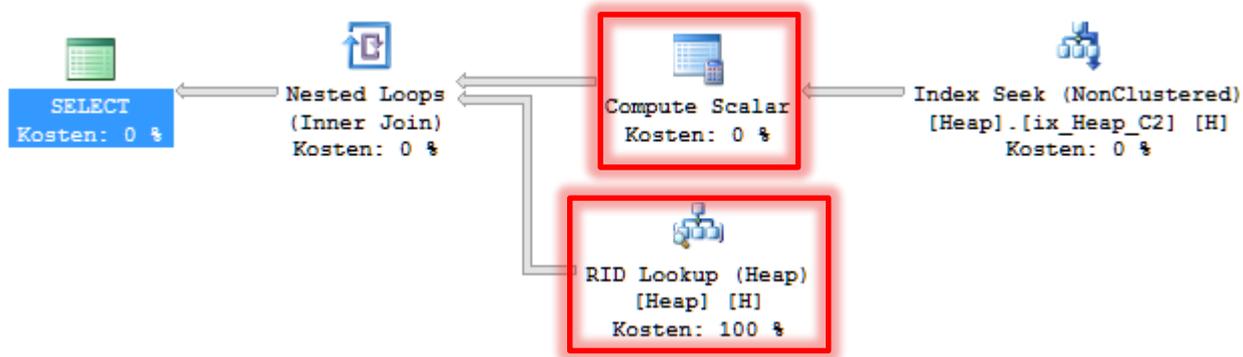
# Clustered vs. HEAP: NON-Clustered Index

- Ein NON-Clustered Index in einem Clustered Index hat immer den Clustered Key als Pointer gespeichert!
  - Die Größe des Pointers bestimmt sich aus dem verwendeten Datentypen
  - Der REBUILD eines Clustered Index impliziert keinen Neuaufbau des NON-Clustered Index
- Ein NON-Clustered Index in einem HEAP hat immer die RID (Row Locator Id) als Pointer gespeichert!
  - Die Größe des Pointers ist IMMER 8 Bytes
  - Ein REBUILD der Tabelle bedeutet immer einen Neuaufbau ALLER NON-Clustered Indexes, da sich die Position des Datensatzes ändert (neue RID).

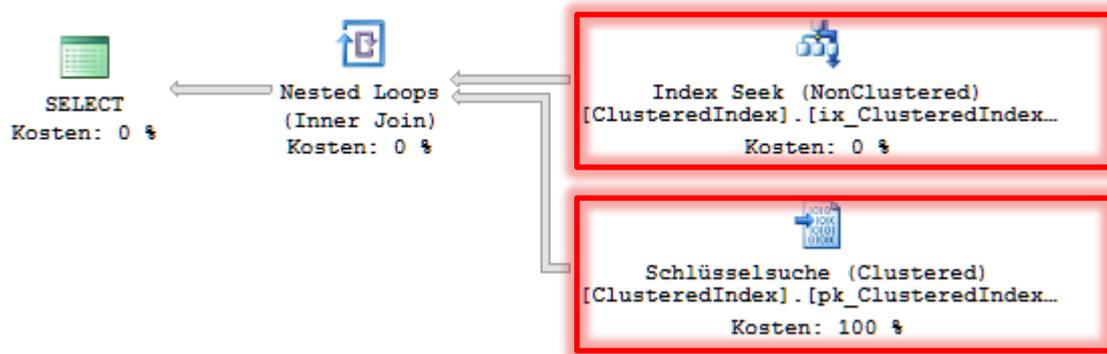
# RID-Lookup vs Key-Lookup

- Lookups werden verwendet, um Werte aus Attributen zu ermitteln, die nicht durch einen Index abgedeckt sind
- RID-Lookups erzeugen immer nur EINEN I/O!
- Key-Lookups erzeugen I/O abhängig von der Tiefe eines Index

# RID Lookup



# Key Lookup

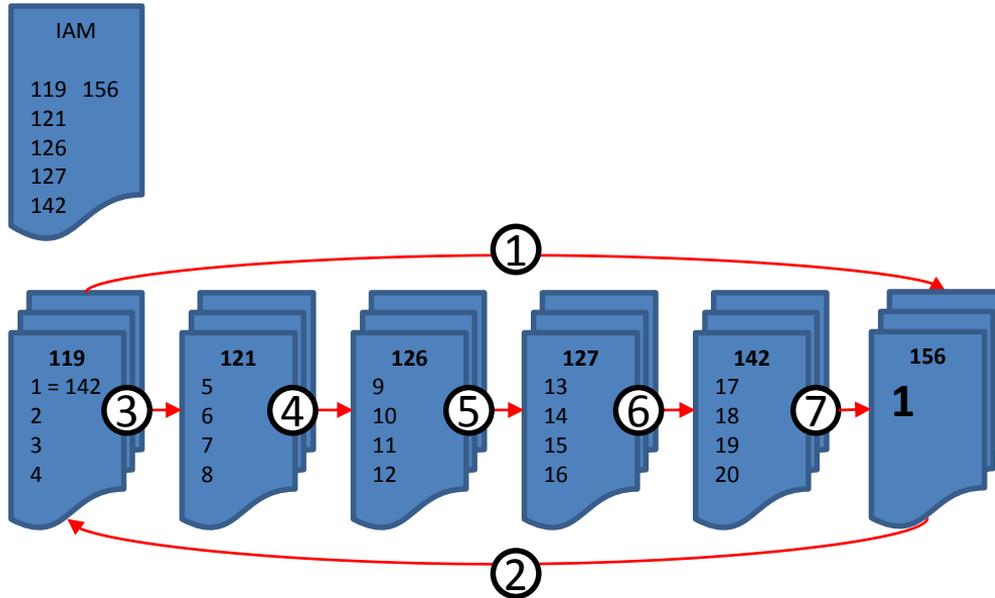


- Wie kostspielig ein Key Lookup ist, hängt von der Datenmenge im Clustered Index ab.
- Je mehr Datensätze vorhanden sind, umso mehr B-Tree-Ebenen müssen für jeden Key Lookup durchlaufen werden!

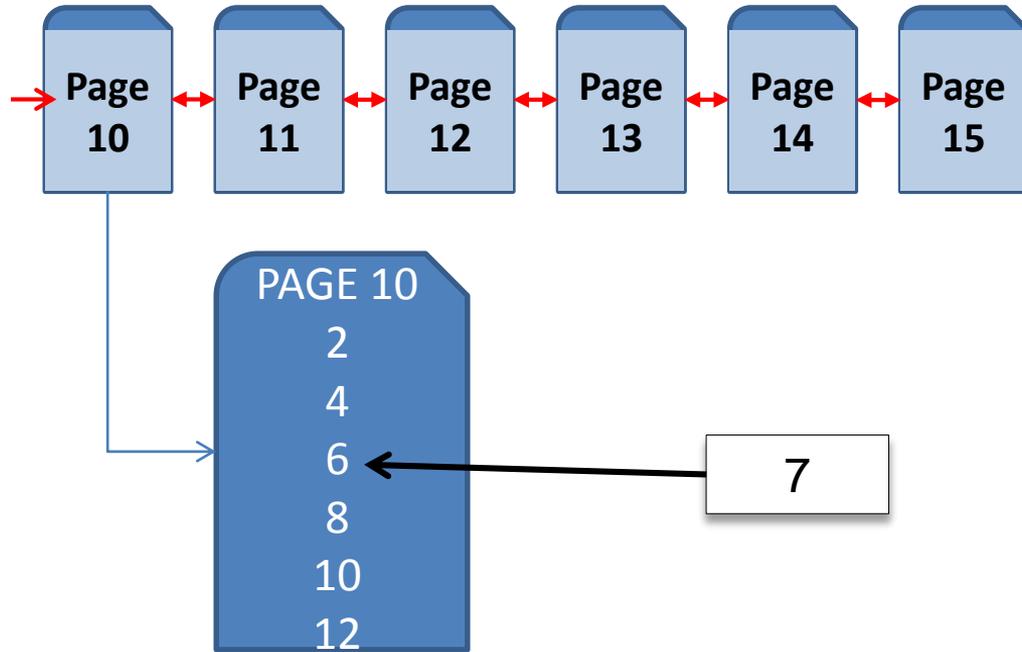
# HEAPS: Forwarded Records

- Forwarded Records können nur in HEAPS auftreten!
- Ein Forwarded Record wird generiert, wenn ein Datensatz nicht mehr auf eine Datenseite passt.
- Ein Forwarded Record hat immer nur EINEN Pointer

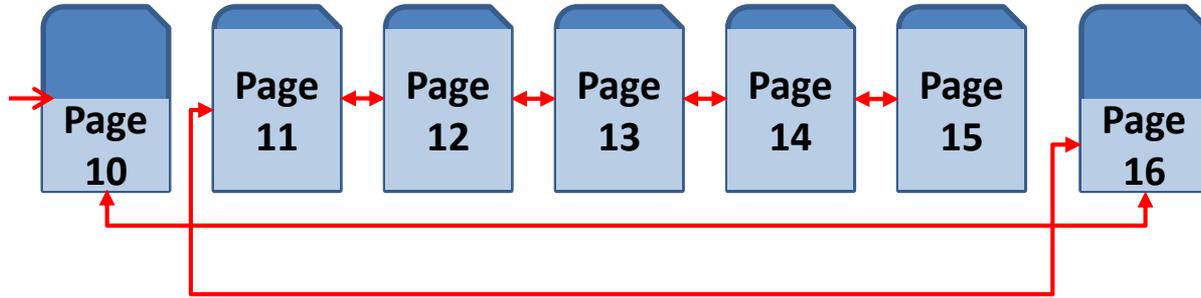
# HEAPS: IO bei Forwarded Records



# Clustered Index: Page Splits



# Clustered Index: Page Splits



# Actions of Page Split

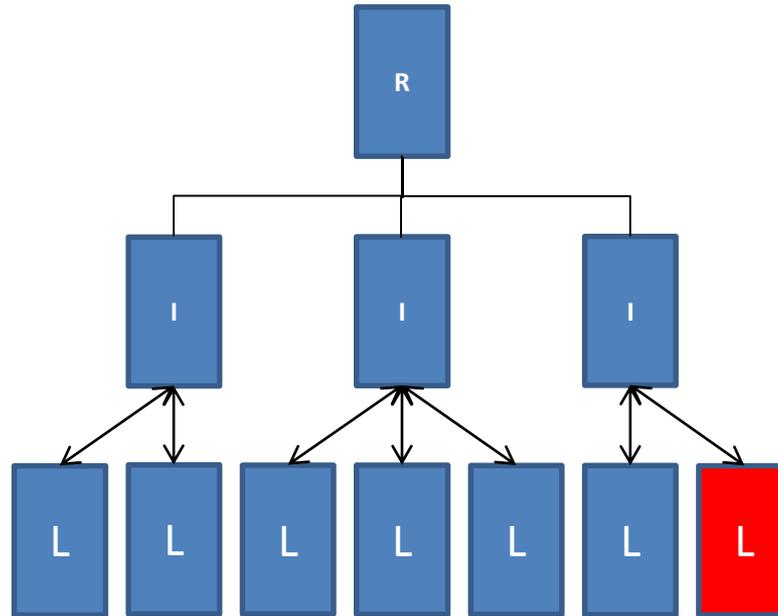
## LOP\_BEGIN\_XACT

LOP_INSYSXACT	LCX_INDEX_INTERIOR	Mark index page as part of system transaction
LOP_INSYSXACT	LCX_CLUSTERED	Mark leaf page as part of system transaction
LOP_INSYSXACT	LCX_CLUSTERED	Mark NEXT leaf page as part of system transaction
LOP_MODIFY_ROW	LCX_PFS	Set the value of free space in PFS
LOP_HOBT_DELTA		Increase the value for the amount of data pages
LOP_FORMAT_PAGE	LCX_HEAP	Zero the new page
LOP_INSYSXACT	LCX_CLUSTERED	Mark new created page as part of system transaction
LOP_INSERT_ROWS	LCX_CLUSTERED	Insert data on the new page
LOP_DELETE_SPLIT	LCX_CLUSTERED	Remove the data from the original page
LOP_MODIFY_HEADER	LCX_HEAP	Set [NEXT PAGE] on original page
LOP_MODIFY_HEADER	LCX_HEAP	Set [PREVIOUS PAGE] on original page
LOP_INSERT_ROWS	LCX_INDEX_INTERIOR	Update of the root page / B-tree level
LOP_INSYSXACT	LCX_CLUSTERED	Release NEWLY added leaf page as part of system transaction
LOP_INSYSXACT	LCX_CLUSTERED	Release OLD next page as part of system transaction
LOP_INSYSXACT	LCX_CLUSTERED	Release NEW next page as part of system transaction
LOP_INSYSXACT	LCX_INDEX_INTERIOR	Release B-Tree as part of system transaction

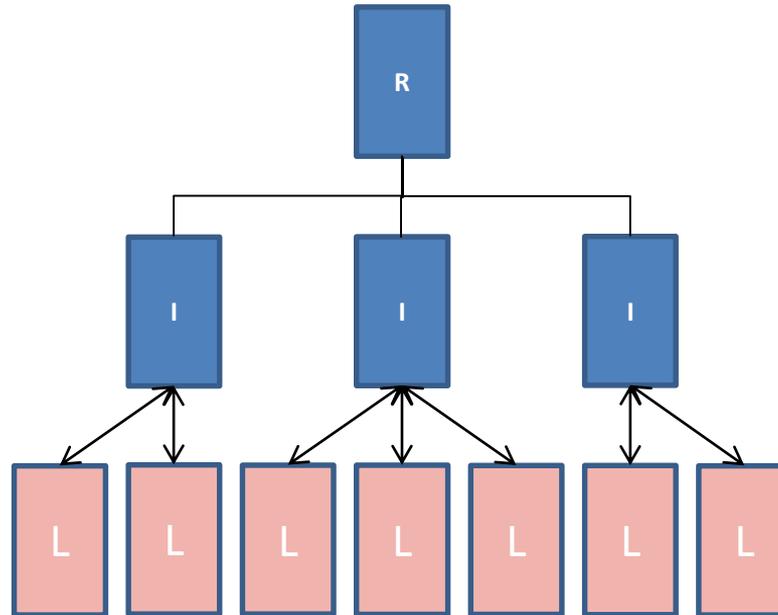
## LOP\_COMMIT\_XACT

# HIGH CONCURRENCY SYSTEMS

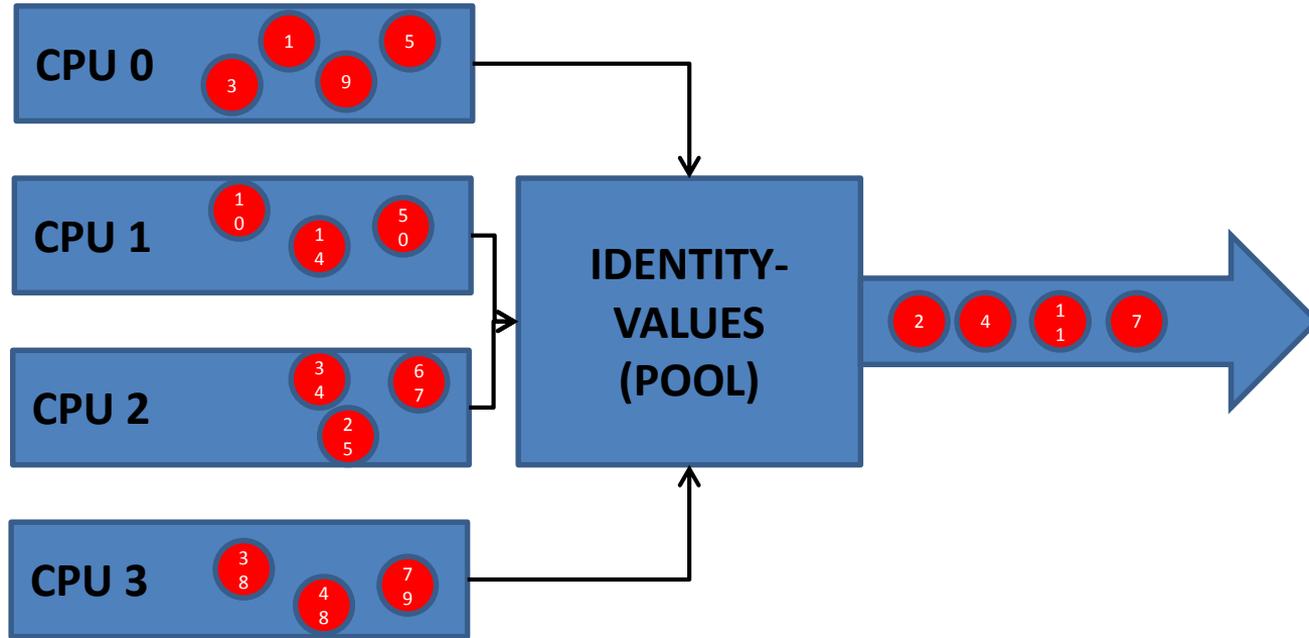
# Clustered Index – IDENTITY()



# Clustered Index – zufällig



# Insert IDENTITY Values...



# FRAGEN?

Ich freue mich auf Ihr Feedback!

[Uwe Ricken]

**VIELEN DANK!**