

Why Inserts into Partitioned Tables are faster...

Educational Article

distributed by



<http://www.IT-Checklists.com>

Table of Contents

Introduction.....	3
Background.....	3
Summary.....	3
Related Articles.....	3
Example: Database Table and Data Volume.....	4
Analysis and Evaluation.....	5
Inserting 100 records from 100 random customers into a non-partitioned database table.....	5
Comparing the insert into a non-partitioned and into a partitioned table	6
Other Options – What about	7
What about a partitioned table, but using global indexes?.....	7
What about inserting into table without indexes, and creating the indexes afterwards?.....	8
How Mercury Consulting Ltd. can help you	9
Our Services	9
Our Products.....	12
Database Independent Products.....	12
Database Specific Products.....	13
Our free White papers.....	14
About the Author.....	14

Why are inserts into a day-partitioned table faster than inserts into a not partitioned table?

Introduction

Background

Using partitioned database tables provides many benefits.

Fast remove of old data just by dropping oldest partition is the best-known advantage.

This article describes the least-known advantage: Faster INSERT's.

Summary

In case that indexes need to be active during inserting, it is for performance reasons crucial to have all index blocks related to new inserted records cached in the database cache.

In case of large tables, the indexes usually don't fit fully into the database cache.

As all or at least most inserted records are usually from one day, using partitioned indexes on (e.g. day-) partitioned tables allow full caching of that day's index partitions.

Based on an example this article derives and explains the time difference of 10 ms (partitioned) to 910 ms (not partitioned) for inserting 100 records into a large table.

Related Articles

[1]	Daily, Weekly or Monthly Partitions http://www.mercury-consulting-ltd.com/wp/daily_weekly_or_monthly_partitions.html Mercury Consulting Ltd., Schaan, 2006
-----	--

Example: Database Table and Data Volume

13 Million customers identified by customer_id, with average 5 transactions per day → 6.5 million transactions per day.

Insert transaction records into database table TRANSACTIONS continuously.

The customer_id's are random from complete range.

16 KB Database Block Size, 100 records per data block, 6.5 million transactions records per day → 1 GB per day

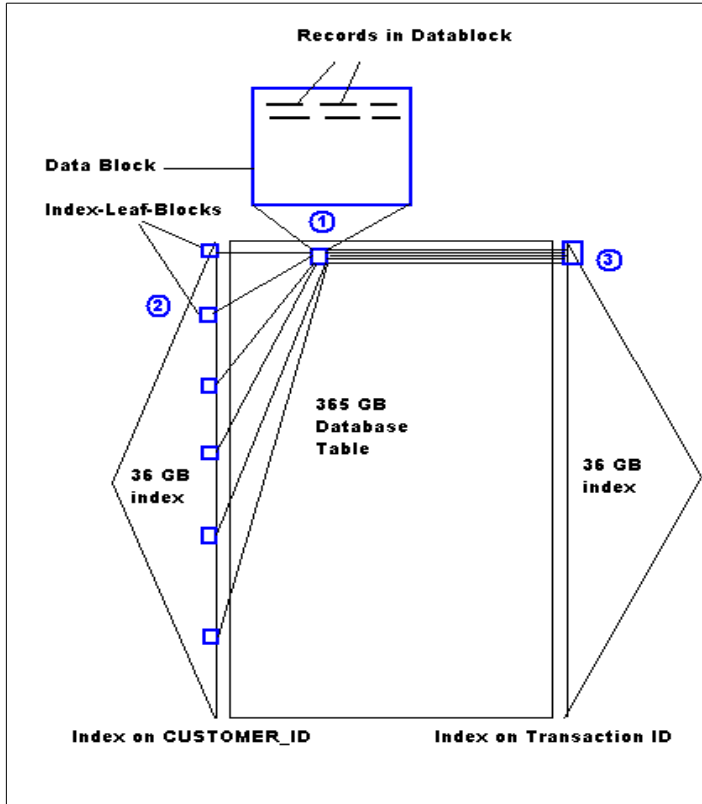
2 Indexes, each 10% of table size: 1 Index on CUSTOMER_ID, 1 Index on TRANSACTION_ID; The transaction ID is a sequence.

365 days retention period.

	Not partitioned	Partitioned by date (daily)	
Table "TRANSACTIONS"	365 GB	365 partitions x 1.0 GB = 365 GB	
Index I_TRANS_CUSTOMER_ID	36 GB	365 partitions x 0.1 GB = 36 GB	
Index I_TRANS_TRANS_ID	36 GB	365 partitions x 0.1 GB = 36 GB	
Total	437 GB	437 GB	

Analysis and Evaluation

Inserting 100 records from 100 random customers into a non-partitioned database table



Inserting 100 records from random customers:

Step	Description	Database Segment	Blocks accessed
1	All 100 records are inserted into the same database table block	Table "TRANSACTIONS"	1
2	As the 100 records are from 100 different CUSTOMER_ID's, the index-entries need to be placed in 100 different index leaf blocks .	Index I_TRANS_CUSTOMER_ID	100
3	All 100 records have sequential transaction ID's, therefore the index entries are inserted into the same index leaf block (and there is still space for many more). <i>(The fact that this index leaf block could be a point of contention in case of high concurrency - inserts from many database sessions - is out of scope for this example)</i>	Index I_TRANS_TRANS_ID	1

Comparing the insert into a non-partitioned and into a partitioned table

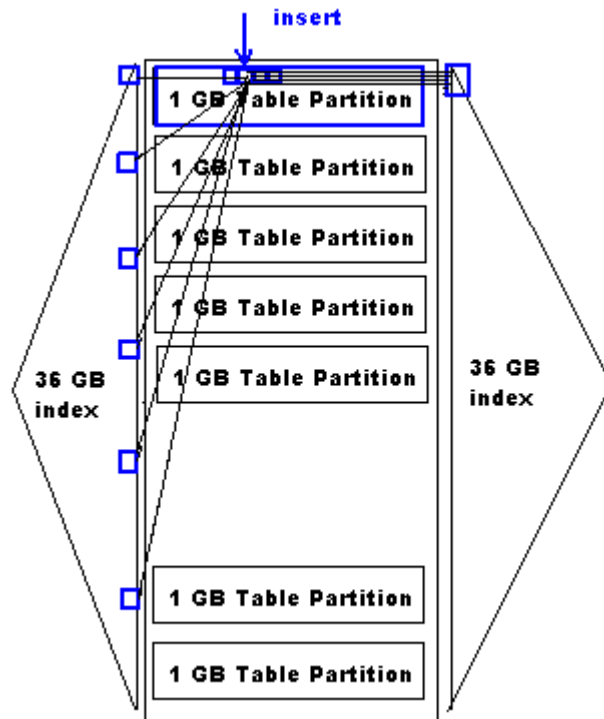
Background: From a performance point of view the main factor are disk reads, typically 6 to 9 milliseconds for a random read of a 16 KB block.

	Inserting 100 records into not partitioned table	Inserting 100 records into day-partitioned table
Table "TRANSACTIONS"	1 disk read to read empty block, 1 disk write to write back full block	1 disk read to read empty block, 1 disk write to write back full block
Index I_TRANS_CUSTOMER_ID	The 36 GB index does not fit completely into the 10 GB Database cache. As many other tables and indexes compete for the database cache, let's assume that this index gets about 1/3 rd of the database cache, that's 3.3 GB or 9% of the index size; that means that 91% of disk blocks need to be read from disk; and before ageing out written back to disk. 91 disk reads 91 disk writes	The 100 MB small index partition from today fits easily into to Database Cache, and due to continuous access of those blocks they remain in the cache. 0 disk reads very low disk writes (only at checkpoints) – assume 0.1
Index I_TRANS_TRANS_ID	0.1 disk reads 0.1 disk writes	0.1 disk reads 0.1 disk writes
Total disk reads	101.1 disk reads	1.1 disk reads
Total disk writes	101.1 disk writes	1.2 disk writes
Total elapsed time [6 ms/read]	606.6 milliseconds	6.6 milliseconds
Total elapsed time [9 ms/read]	909.9 milliseconds	9.9 milliseconds

Other Options – What about ..

What about a partitioned table, but using global indexes?

It's obvious that this will not help....



It is important that the index is partitioned.

What about inserting into table without indexes, and creating the indexes afterwards?

A very good idea, but only usable for batch loading scenarios, and not the OLTP scenario used in this article.

Scenario	Description	Recommended Approach
Batch Loading - initial	Loading data into an empty table (e.g. initial load)	Create table, but do not create the indexes Load data into empty table Create the indexes (in parallel)
Batch Loading only - append	Appending Data into existing table	Evaluate Option 1 and Option 2 below
Batch Loading append + OLTP	Appending Data into existing table, but concurrent OLTP activity	Option 2 below will most likely work.

Appending data into an existing table, already containing data:

Scenario	Phase	Non-partitioned Table	Partitioned Table
Batch Loading Option 1	Before Loading	Disable index (set index to unusable status)	Option 1: Disable index partitions for that table partition (day), where (majority of) data are loaded into.
	After Loading	Rebuild complete index on entire table	Rebuild index partitions only
	Benefit	Questionable / Might not exist – rebuilding complete index might take longer than time saved by faster loading.	Rebuilding index-partitions (of course in parallel) of (e.g. only 1 day) is expected to be shorter than the time saved on faster insert / loading.
	Issues		Concurrent SELECT's on partitioned table might suffer from unusable index-partition. (Depends if data in that partitions would be accessed or not).
Batch Loading Option 2	Loading		Option 2:
	After Loading		<ul style="list-style-type: none"> Load data into empty temporary table, create indexes on this temporary table and use partition exchange to move the temporary table into the partitioned table (DDL-operation only).
	Benefit		Same as Option 1 + no issues with SELECT's on partitioned table during loading

How Mercury Consulting Ltd. can help you

Mercury Consulting Limited (MCL) is a professional consultancy providing experience, support and training in IS/IT operations for companies during high-time-pressure startup phase and following consolidation phase, especially in the Telecom-market.

Our Services

Services

Technology Selection for Disaster Recovery and Business Continuity

- [Database Administration](#)
- [Database Tuning](#)
- [Audit of Database Installations](#)
- [Database- and Application Capacity Management and Capacity Planning](#)
- [Project Support: Requirements Definition, initial Sizing, Installation....](#)
- [Service Level Management](#) by ITIL[®] Foundation certified expert
- [Systems Documentation](#)
- [Writing your Operations Manual](#)

Database Administration through certified Database Administrators, 16 years experience

Billing DBA

Specialized on Billing System Databases
Experience in Telecommunications and Banking

We cover all aspects of Database Administration:

- System DBA
- Application DBA / Center of Excellence advising Developers
- Development DBA

[Back to top](#)

Database Tuning

We don't just change a few performance-parameters....

Analysis - WHERE is the problem ?

- Application
- Execution Plans
- Resource Contention - we know how fast your disks or storage array could / should be...
- Database Parameters

Documentation of analysis, findings and proposed changes including all captured evidence and argumentations for our proposed changes.
Implementation after your agreement and in line with your Change- and Release Management (*we do know ITIL...*).

[Back to top](#)

Auditing of Database Installations

Our Audit focuses on Stability

Related product: "[Database Health Check - Stability Assessment](#)"

[Back to top](#)

Database- and Application Capacity Management and Capacity Planning

...not only disk space but also CPU, Memory and IO-performance We strive for **MANAGING capacity - that means keeping resource requirements as low as possible and just plan for the required extension afterwards.**

Our capacity planning is not just simple extrapolation of trend lines, it is model-based.

- Definition of Performance Indicators (PI) and Key Performance Indicators (KPI)
- Establishing an initial baseline
- Implementation of automatic recordings and capacity monitoring
- Deriving Business-to-IS/IT-Usage Factors: GB per Mio Customers, CPU's per 10,000 Invoices/hour, ...
- Applying those Usage Factors to Forecasts from Business
- Capacity Monitoring and Reporting against Forecast: Checking if real IS/IT resource usage matches the Forecasted values, in case of unusual changes investigations if those are caused by a problem

[Back to top](#)

Project Support: Requirements Definition, initial Sizing, Installation....

Your business experts usually do spend significant effort on functional requirements, but unfortunately the non-functional requirements are not sufficient covered.

Feel free to choose

- to purchase our template and use it yourself or
- to use our consulting service to explain all that to your staff and customize that template exactly to your project's requirements

Our related products

- **["Nonfunctional Requirements \(more than 150 requirements\)"](#)**
- **["Enterprise Application Integration \(EAI\): Requirements, Checklist and Template for Planning, Defining and Documenting Application Interfaces"](#)**

[Back to top](#)

Service Level Management

by **ITIL® Foundation** certified expert

We do understand ITIL based Service Level Management and SLA's. As Oracle Experts we can support your Service Level Manager establishing an Operations Level Agreement (OLA) for the database team.

Our related Product: **["Database OLA / SLA"](#)**

[Back to top](#)

Systems Documentation

There are many reasons why your systems documentation is far from being complete...

OUR vision is a systems handbook which not only satisfies every auditor, but is also a major help for your daily and non-daily operations. Our consultants have in addition to experience in application support also excellent writing skills ("Document Writer") in English and German and are able to compile a systems handbook...

- by providing application support we learn how the system works and fill step by step our Handbook-Template
- by "Knowledge Engineering" - Reading available information and interviewing your application support staff
- [Read more about this service...](#)

Our related product: **["Template: Systems- and Operations Handbook"](#)**

[Back to top](#)

Our Products

you can purchase online and immediately download at our eBook-Shop



Our Checklists and Templates will help you to ensure that good or best practice is not only known but consistently applied!

Database Independent Products

Product	Benefit
270 Non functional Requirements.	Requirements Template with 70 non functional requirements for selecting or developing robust, reliable, stable and manageable applications to meet the Service Level Agreements (SLA's). For external RFP's (Request for Proposal) and for internal development.
Checklist for Data Migration	65 important questions to identify and address or exclude typical migration pitfalls in an early phase of the project, thus ensuring the confidence for keeping the time plan.
Template: Systems- and Operations Handbook	Template to establish that documentation auditors like to see for each IT Application / IT-System.
Template "Data Centre Operations Manual"	Template to create an Operations Manual for a Data Centre, focusing on Data Centre Infrastructure.
Application & Server Inventory Template	This document assists in first step of gathering and classifying information about servers, applications and software.
Interface Checklist	Those questions which you need to ask before starting the development! Requirements, Checklist and Template for Planning, Defining and Documenting Application Interfaces.
Server Upgrade / Migration Checklist	This vendor- and operating system independent checklist deals with "big picture" and helps to evaluate all potential impact of upgrading a server, e.g. adding CPU's, replacing CPU's with faster CPU's , Operating System Upgrade, adding Storage or replacing a server with a newer, bigger model
Application Upgrade / Migration Checklist	Purpose of this document is to support <ul style="list-style-type: none">• The decision making process of "should we upgrade now or later – or not at all?"• The impact analysis – identification of all internal and external dependencies.• The gathering of requirements for the upgrade.• The selection of the best upgrade approach ("In-Place Upgrade" or "New Installation").• Providing input required for the Upgrade Planning.
Checklist for Production Release and System Handover	Checklist for small and medium projects focusing on non-functional aspects for operations team.

Product	Benefit
	Simple but effective!
Template for Outage Planning	This template with in-depth explanations of tasks required to plan outages on medium to large IT Systems will avoid that important steps are forgotten.
Application Retirement – Template and Checklist	Very often "Application Retirement" discusses the process of upgrading an application to a newer version or replacing by another similar product, emphasizing the benefits of upgrading and only minor focus on the decommissioning. This document deals solely with the "end of operations" / "decommissioning" of an obsolete IT application or old version of an upgraded / replaced application and addresses IT Organizations at maturity levels between 1 and 3, with highest benefit for those on level 2 striving to achieve level 3.
Backup SLA / OLA	Operations Level Agreement (OLA) with the Backup Team.
Decision Template for selecting Disaster Recovery (DR) Technology	This template helps you to define your individual DR-requirements and after explaining the four main technology groups to select the most suitable technology.
Business Requirements for Archiving and Purging	Template with Business Requirements for Archiving & Restore & Purging. Not removing old customer data can cause conflicts with privacy laws. Business must act and clearly specify what to purge and what to archive!
Application Health Check: Stability Assessment	Using this template to check your systems - and DOCUMENT the findings might show you even more potential issues beyond invalid objects.

Database Specific Products

Product	Benefit
Database SLA / OLA Template for Database Operations Level Agreement (OLA) / SLA	If a Service Level Manager needs to offer to Business a Service Level Agreement (SLA) for an End-to-End IS/IT-Service, he sign this SLA only after he arranged within IS/IT for each system or component used to provide this service an Operations Level Agreement (OLA) with the providing team or department. This document provides a template for such an Operations Level Agreement (OLA) for Oracle Databases containing the agreed values and for QA-purpose also the measures implemented to reduce the likelihood of violations of those agreed values. It does not only deal with availability, but contains also comprehensive service catalogue of advanced DBA services and a template for the Service Level Reporting (SLR).
Database Health Check - Part 1: Stability Assessment	Stability Assessment of your Oracle Database. Most Application- and Database-Crashes can be avoided when detecting early indicators and reacting to them. Be Proactive - Check Now!

Product	Benefit
DBA and Application Support: Job Description and Self Assessment	<p>Checklist to ensure that all 60 DBA-duties are assigned: System DBA, Development DBA and Application DBA versus Application Support</p> <p>If you have just a job role "DBA", but not a dedicated job role "Application DBA" those 19 duties must be explicitly assigned to either "Application Support" or to the "DBA" - otherwise they might not be executed!</p> <p>Detect unassigned tasks before an auditor reports them !</p> <p>This product addresses disputes between System DBA and Application Support (or, if existing, dedicated Application DBA) regarding the responsibility for the application's database objects.</p>

Our free White papers

<http://www.mercury-consulting-ltd.com/wp/whitepapers.html>

White paper	Benefit and Description
The Importance of Application Level Monitoring.	Keeping your applications free from invalid objects is an important task, but does not guarantee error free operations. This free White paper explains the difference between "Application Level Monitoring" to "Database Monitoring" and "System/Server/OS"-Monitoring.
The Danger of Invalid Database Objects	An awareness paper for Operations Managers and Application Support describing the problems and potential risks caused by invalid objects in an Oracle Database.
Private or Public Synonyms – or no Synonyms at all ?	A decision support paper visualizing the pro's and contras on a single page in tabular form and evaluating the arguments.

About the Author

The Author of this white paper is an



ISO 20000 certified Consultant
and internal Auditor

with 16 years experience as DBA and 10 years experience in Telecommunications.