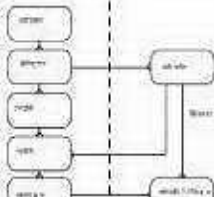


ওরাকল ডাটাবেজ অ্যাডমিনিস্ট্রেশন

মো: ইফতেখারুল আলম

ওরাকল ডাটাবেজ ব্যবস্থাপনায় ডাটাবেজের মাধ্যমে কখন করা স্টোরেজের সুচম ব্যবহার অত্যন্ত গুরুত্বপূর্ণ একটি কাজ। ডিবিএ এ কাজটি যত্নের সাথে করতে হয়। আজকের এ অধ্যয়নাময় আলোচনায় প্রায়শ প্রায়শের স্টোরেজ ব্যবস্থাপনার খুঁটিখাটি বিভিন্ন দিক পাঠকের সামনে তুলে ধরা।

ওরাকল ডাটাবেজের ৩টি উপাদান- কন্ট্রোল ফাইল, ডাটা ফাইল ও লক ফাইল রয়েছে, যা বিপাক আলোচনায় উল্লেখ করা হয়েছিল। অবশ্যই পাঠকের মনে এই প্রশ্ন আসা অব্যাহত নয় যে, এ সব ফাইল কিভাবে হার্ডড্রাইভে স্থান করে নেয়। মূলত ওরাকল ডাটাবেজ টেবিলস্পেস, ডাটাসাইল, সেগমেন্ট, এন্ট্রিটে এক ডাটা বকে বিভক্ত।



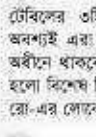
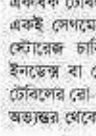
টোলেজ ব্যবস্থাপনায় মনে হিলেদন

হার্ডড্রাইভে ইন্টার কন্ট্রোল সরবরাহ করা ডাটা ডাটা বকে স্থান করে নেয়। বিপাক অ্যালোচনায় লক ফাইল ও কন্ট্রোল ফাইল ডাটা বকে কিভাবে স্থান করে নেয় তার বিশদ বর্ণনা দেয়া হয়েছিল। এবার দেখা যাক, অন্য ডাটাবেজ অবজেক্টগুলো কি এবং তা কিভাবে ম্যানেজ করা হয়ে থাকে।

সেগমেন্টের প্রকারভেদ : সেগমেন্ট ডাটাবেজে স্থান লক্ষণকারী একটি অবজেক্ট, যা ডাটা ফাইলে নিজেদের স্থান করে নেয়। এখানে সেগমেন্টের বিভিন্ন প্রকার এবং তাদের নিয়ে নানা প্রশ্নের সমাধান দেয়া হয়েছে।

টেবিল : ডাটাবেজের সবচেয়ে কখন সংরক্ষণ পদ্ধতি হলো টেবিল। টেবিল সেগমেন্ট টেবিলে অথবা ডাটা কোনোদ্রুপ ক্লাস্টার অথবা পার্টিশন না করেই সংরক্ষণ করে। মূলত কোনো প্রকার অর্ডার মেইনটেনেন্স না করেই টেবিল সেগমেন্টে ডাটা সংরক্ষণ করা হয়ে থাকে। একটি টেবিলের সব ডাটা অংশই একটি একক টেবিল স্পেসে স্টোর করতে হবে। টেবিলের রো-এর ওপর একজন ডিবিএ খুব কমই নিয়ন্ত্রণ থাকে।

টেবিল পার্টিশন : যখন কোনো টেবিলে একই সময় অনেক ইউজারের ব্যবহার বেড়ে যায়, তখন ডাটার প্রাপ্যতা একটি বড় সমস্যা হয়ে



দেখা দেয়। এ সময়ে একটি একক টেবিলে ডাটা সংরক্ষণ না করে একে বিভিন্ন পার্টিশনে বিভক্ত করে ভিন্ন ভিন্ন টেবিল স্পেসে রাখে। একে বৈকল টেবিল-পার্টিশন। যখন কোনো টেবিলকে পার্টিশন করা হয়, তখন প্রতিটি পার্টিশনকে আলাদা আলাদাভাবে একটি একক সেগমেন্ট হিসেবে ধরা হয়।

ক্লাস্টার : টেবিলের মহতাই ক্লাস্টার একটি ডাটা টাইপ সেগমেন্ট। কি কলাম ডায়াল ওপরে ভিত্তি করে ক্লাস্টারের সৌভাগ্য স্টোর হয়ে থাকে। একটি ক্লাস্টারের অধীনে একধিক টেবিল থাকতে পারে। এই টেবিলগুলো একই সেগমেন্টের আওতাধর থাকে এবং একই স্টোরেজ চারিত্রিক বৈশিষ্ট্যতা শেয়ার করে। ইনডেক্স বা হেসিং অ্যালগরিদম দিয়ে ক্লাস্টার টেবিলের রো-এর অভ্যন্তরে ডাটা নির্বাহী না এর অভ্যন্তরে থেকে উদ্ধার করা হয়।

ইনডেক্স : কোনো ইনডেক্সে সব এন্ট্রি একটি একক ইনডেক্স সেগমেন্টের আওতাধর থাকে। যদি কোনো টেবিলের ৩টি পৃথক ইনডেক্স থাকে তবে অবশ্যই এরা ৩টি পৃথক ইনডেক্স সেগমেন্টের অধীনে থাকবে। এই সেগমেন্টের মূল উদ্দেশ্য হলো বিশেষ কি (Key) স্মৃত কোনো টেবিল রো-এর সৌকরশন নির্বাহ করা।

ইনডেক্স অর্গানাইজ
টেবিল : এখানে কি (Key) ডায়াল ওপরে ভিত্তি করে ডাটা সংরক্ষণ হয়ে থাকে। এর জন্য আলাদা করে বুকআপ টেবিলের প্রয়োজন হয় না। কারণ সব ডাটাই ইনডেক্স ট্রি থেকে সরাসরি হয়ে থাকে।

ইনডেক্স পার্টিশন : একটি ইনডেক্সকে বিভিন্ন অংশে পার্টিশন করে তাদের বিভিন্ন টেবিল স্পেসের আওতাধর রাখা হয়। এই ক্ষেত্রে প্রতিটি পার্টিশন আলাদা আলাদা সেগমেন্টের আওতাধর এবং আলাদা টেবিল স্পেসের অধীনে থাকে। ইনডেক্স আইও কন্ট্রোলই এই বিশেষ ইনডেক্সের প্রাথমিক কাজ।

আর্ভ সেগমেন্ট : যখন ডাটাবেজে ডাটার কোনো পরিবর্তন ঘটে, তখন পরিবর্তন পূর্ববর্তী ডাটা ওরাকল যে সেগমেন্টে সংরক্ষণ করে তাকে বলা হয় আর্ভ সেগমেন্ট।

টোম্পোরারি সেগমেন্ট : যখন কোন ইউজার



CREATE INDEX,
SELECT DISTINCT,
SELECT GROUP BY
কমন্ড দেয়, তখন ওরাকল সার্ভার চেষ্টা করে মূল

মেমরিতেই সার্ভ অপারেশন সম্পন্ন করতে। যখন এই সার্ভ অপারেশন সম্পন্ন করার জন্য অপরো স্থানে প্রয়োজন হয়। তখন ইন্টারমিডিয়েট ফলাফল ডিস্ক-সংরক্ষণ করার প্রয়োজনীয়তা প্রকাশ পায়। এসব ইন্টারমিডিয়েট ফলাফল সংরক্ষণের জন্য টেম্পোরারি সেগমেন্টের প্রয়োজন হয়।



এগজকি সেগমেন্ট : অনেক সময় দেখা যায়, কোনো টেবিলের কলামে লার্জ অবজেক্ট, ফেমেন = টেক্সট, ডকুমেন্ট, ছবি, ভিডিও ইত্যাদি রাখতে হয়। যদি কলামে এরূপ বস্তু হয় তখন ওরাকল সার্ভার একটি পৃথক সেগমেন্ট তৈরি করে এবং ওইসব ডাটা সংরক্ষণ করতে থাকে। এইরূপ ডাটা মিউইভ করার জন্য পয়েন্টার বা লোকেটর ব্যবহার করা হয়।



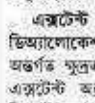
নেসটেড টেবিল : কোনো টেবিলের কলাম ইউজার ডিফাইন টেবিলকে নামক করে। এই ক্ষেত্রে কলামের অভ্যন্তরে অবস্থিত টেবিলকে বলা হয় নেসটেড টেবিল এবং এই টেবিল একটি পৃথক সেগমেন্টের আওতাধর থাকে।



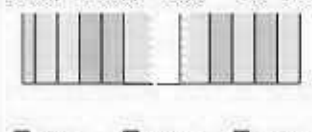
বুটস্ট্রপ সেগমেন্ট : এর আরেক নাম ক্যাশ সেগমেন্ট। ডাটাবেজ তৈরির সময় redo log ফ্রন্ট যখন স্থান করে তখন এই সেগমেন্টের সৃষ্টি হয়। যখন



কোনো ডাটাবেজ ইনস্টলেশন দিয়ে ওপেন হয় তখন এই সেগমেন্টে ডাটা ডিকশনারি ক্যাশ উন্মুক্ত থেকে সহায়তা করে।



একটেকি অ্যাগোরেশন এবং ডিঅ্যাগোরেশন : একটি টেবিল স্পেসের অন্তর্গত পৃথকভাবে অংশ হচ্ছে একটেকি। একটি একটেকি অ্যাগোরেশন হয় যখন সেগমেন্ট ক্লিয়ারে, একটেকি, অলটার্ট হয় আর

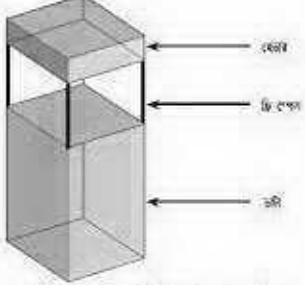


ডিঅ্যাগোরেশন হয় যখন সেগমেন্ট ক্লিয়ার, অলটার্ট এবং ট্রান্সফার্ট হয়।

একটেকি ব্যবহারের প্রক্রিয়া : যখন কোনো টেবিলস্পেসে তৈরি করা হয় তখন ওই টেবিল স্পেসের ডাটা ফাইলকে একটি অংশ এর হেডার হিসেবে ব্যবহার হয়, যা মূলত এর প্রথম ব-কোই।

হয়ে থাকে। যখন কোনো সেগমেন্ট তৈরি করা হয় তখন টেবিল স্পেসের বালি অংশে এন্ট্রিতে জমা বরাদ্দ হয়। পরস্পর কাছাকাছি অবস্থিত স্পেসের যখন সেগমেন্ট ব্যবহার করে তখন তাকে বলা হয় ইউজড এন্ট্রিতে। যখন কোনো সেগমেন্ট এই স্পেস থেকে নেমে তখন এই বালি এন্ট্রিতে জমা হয়ে ফ্রি এন্ট্রিতে পূরণ, যা পরে সেগমেন্টের ব্যবহারের জন্য বরাদ্দ দেয়া হবে।

ডাটাবেজ ব্লক : সর্বশেষ যেটা আই/ও(I/O) ইউনিট, যা এক বা একাধিক অংশে (অপারেটিং সিস্টেম) ব্লকের সমন্বয়ে গঠিত। টেবিল স্পেস তৈরির সময় একে সেটা করতে হয়। DB BLOCK SIZE দিয়ে ডিফল্ট ব্লক সাইজ নির্ধারণ করা হয়।



ডাটাবেজ ব্লকের উপাদান : এর উপাদান এটা : হেডার, ডিফেন্স ও ডাটা।

ব্লক হেডার : ব্লক হেডার ডাটাস্পেস আফ্রেন্স, টেবিল ডিফারেন্স এবং ট্রানজেকশন স্পেসের তথ্য সংরক্ষণ করে। এটা উপর থেকে নিচে সম্পর্কিত হয়।

ডাটা স্পেস : হো ডাটা দিত থেকে উপরে ইনসার্ট হয়ে থাকে।

ফ্রি স্পেস : ফ্রি স্পেস একটি ব্লকের মাঝামাঝি স্থানে অবস্থান করে যাতে হেডার এবং ডাটা স্পেস প্রয়োজনমতো সম্পর্কিত থেকে পারে।

ব্লক স্পেস ইউটিলাইজেশন প্যারামিটার : ব্লক স্পেস ইউটিলাইজেশন প্যারামিটার আমরা ডাটা ও ইন্ডেক্স সেগমেন্টের স্পেস ব্যবহার নিয়ন্ত্রণের আদে লাগতে পারে। যেসব প্যারামিটার আমরা ব্যবহার করতে পারি তা হলো INITRANS, MAXTRANS, PCTFREE, PCTUSED।

কনকারেন্ট নিয়ন্ত্রণ প্যারামিটার : INITRANS, MAXTRANS কনকারেন্ট নিয়ন্ত্রণকারী প্যারামিটার। এই দুই প্রকারের প্যারামিটার সর্বোচ্চ এবং সর্বনিম্ন সংখ্যক ট্রানজিকশন স্পেস নির্ণয়ে সাহায্য করে। কোন ব্লকে নির্দিষ্ট সময়ে কতবার পরিবর্তন হবে তার তথ্য ট্রানজিকশন স্পেস সংরক্ষণ করে। একটি ট্রানজিকশন শুধু একটি ট্রানজিকশন স্পেসেই করতে। এমনকি যদি সে বহু বো বা ইন্ডেক্স এন্ট্রির পরিবর্তন সাধন করে তবেও।

ইনিট্রান্স (INITRANS) : সর্বনিম্ন কতবার ট্রানজিকশন সংঘটিত হবে তা এই প্যারামিটার

দিয়ে নিয়ন্ত্রণ করা হয়। ডাটা সেগমেন্টের জন্য এর ডিফল্ট জায়গা ১ এবং ইন্ডেক্সের জন্য ২। উদাহরণস্বরূপ বলা যায়, যদি INITRANS ৩ সেট করা হয় তবে একই সময়ে তিনটি ট্রানজেকশন ডাটা ব্লকের পরিবর্তন সাধন করতে পারবে। যদি প্রয়োজন হয় তাহলে অতিরিক্ত স্পেস ফ্রি স্পেস থেকে অ্যালোকেশন করবে। যাতে আরও ট্রানজিকশন সংঘটিত হতে পারে।

ম্যাক্সট্রান্স (MAXTRANS) : এই প্যারামিটারের মাধ্যমে সর্বোচ্চ কতবার কনকারেন্ট ট্রানজিকশন সংঘটিত হয়ে ডাটা অথবা ইন্ডেক্স ব্লকে পরিবর্তন সাধন করতে পারবে তা নির্দিষ্ট করা যায়। এর ডিফল্ট জায়গা ২৫৫। যখন একই সেট করা হয় তখন এর জায়গা ট্রানজিকশন স্পেসের স্পেস ব্যবহারের বিধিবিধির অংশে করে যা রেও ইন্ডেক্স ডাটাবেজ ব্লক পর্যন্ত স্পেসের শিকড়তা দেয়।

পিসিটিফ্রি (PCTFREE) : এই প্যারামিটারের মাধ্যমে ডাটা ব্লকের কত শতাংশ জায়গা আপডেট সংক্রমণ কাজে বালি রাখতে হবে তার উপসং-ন করা হয়। মূলত ডাটা সেগমেন্টের ওপরে এই প্যারামিটার কাজ করে। পিসিটিফ্রি প্যারামিটারের ডিফল্ট জায়গা ১০।

পিসিটিইউজড (PCTUSED) : এই প্যারামিটারের প্রতিবিধি করে সর্বনিম্ন কত শতাংশ ব্যবহার হওয়া স্পেস ওরাকল সার্ভার একটি নির্দিষ্ট ডাটা ব্লকের জন্য বরাদ্দ রাখবে। এর ডিফল্ট জায়গা ৪০।

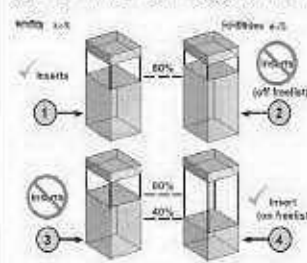
ডাটা ব্লকের ব্যবস্থাপনা : দুইটি পদ্ধতিতে ডাটা ব্লকের সৃষ্টি ব্যবস্থাপনা সম্ভব। ০১: স্বয়ংক্রিয় সেগমেন্ট স্পেস ব্যবস্থাপনা, ০২: অস্বয়ংক্রিয় ব্যবস্থাপনা। আবার সের বর্তমান অ্যালোম্যা সফটওয়্যার সেগমেন্ট স্পেস ব্যবস্থাপনা নিয়ে। স্বয়ংক্রিয় সেগমেন্ট স্পেস ব্যবস্থাপনা ওরাকল ৩ এর একটি বিশেষ ফিচার, যা আপরে ওরাকল ডাটাবেজেতে বিদ্যমান ছিল না।

স্বয়ংক্রিয় সেগমেন্ট স্পেস ব্যবস্থাপনা : ডাটাবেজ সেগমেন্টের অভ্যন্তরে ফ্রি স্পেস এই ব্যবস্থাপনার মাধ্যমে করা হয়। এই পদ্ধতিতে ফ্রি অথবা ইউজড এক বিশেষ প্রতিরোধ রাখা হয়। যাকে বলা হয় বিটিম্যাপ। বিটিম্যাপ সেগমেন্ট একটি বিশেষ প্রকারের ম্যাপ রাখতে করে, যা বিটি (০-১) দিয়ে তৈরি। এই বিটিম্যাপ প্রতিটি ব্লক স্ট্যাটাস নির্দেশ করে। এই ম্যাপ পৃথক স্টেটের বিটিম্যাপ ব্লক ধারণ করে। যখন কোনো নতুন ব্লক বা ডাটা ব্লকে ইনসার্ট হয় তখন সার্ভার ম্যাপ দেখে নির্ধারণ করে কোথায় এই মুহুর্তে ফ্রি স্পেস আছে। যখন কোনো নতুন ইনসার্ট সংঘটিত হয় তখন এই ম্যাপে তার প্রতিফলন ঘটে।

স্বয়ংক্রিয় সেগমেন্ট স্পেস ব্যবস্থাপনা কনফিগারেশন। শুধু টেবিল স্পেস লেবেলে একে আনবেশ করা যায়।
 CREATE TABLESPACE data02
 DATAFILE 'u01/oradata/data02.dbf'
 SIZE 5M
 EXTENT MANAGEMENT LOCAL
 UNIFORM SIZE 64K
 SEGMENT SPACE MANAGEMENT

AUTO:
ব্লক স্পেসের ব্যবহার : একটি উদাহরণের মাধ্যমে আমরা দেখব কিভাবে ব্লক স্পেসের ব্যবহার হয়ে থাকে। বলা যাক, পিসিটিফ্রি = ২০ এবং পিসিটিইউজড = ৪০ নিচে সেগমেন্ট থেকে নেওয়া হলো কিভাবে ইনসার্ট এবং আপডেট সংঘটিত হয়।

স্টেপ-১ : ব্লকে হো ততক্ষণ পর্যন্ত ইনসার্ট হবে যতক্ষণ না ফ্রি স্পেস ২০%-এর কম বা



সময় না হয়। ব্লকে তখন হো ইনসার্ট হবে না যখন ৮০% ব্লক স্পেস হো দিয়ে পূর্ণ হয়ে যাবে।

স্টেপ-২ : অবশিষ্ট ২০% ব্যবহার হবে যখন হো-এর সাইজ বাড়ে। যেমন এন্ট্রি কলাম মাল (NULL), তাকে কোন জায়গা দিয়ে আপডেট করতে হবে তখন আপডেট প্রসারের জন্য ৮০% স্পেসের বাইরে অংশ ব্যবহার করবে।

স্টেপ-৩ : যদি কোনো ব্লকে ডিফল্ট করা হয় (ব্লক থেকে) তাহলে ব্লক ইউটিলাইজেশন ৮০%-এর নিচে নেমে যাবে। কিন্তু ততক্ষণ পর্যন্ত না কোনো ইনসার্ট অপারেশন সংঘটিত হবে যতক্ষণ না পিসিটিইউজ জায়গার (৪০) নিচে না নেমে যাবে।

স্টেপ-৪ : যখন ব্লকের ব্যবহার পিসিটিইউজের নিচে নেমে যাবে তখন ব্লক ইনসার্ট অপারেশনের উপযোগী হবে।

নিম্নলিখিত ডিউকলে পেরি আমরা স্টোরেজ সংক্রমে তথ্য জানতে পারি। DBA TABLESPACES, DBA DATA FILES, DBA SEGMENTS, DBA EXTENTS, DBA FREE SPACE

নিম্নলিখিত কোয়ারিগুলো দিয়ে আমরা একটি নির্দিষ্ট সেগমেন্টে কতগুলো এন্ট্রিতে এবং ব্লক বরাদ্দ হলো তা জানতে পারব।

```
SQL> SELECT segment_name, tablespace, name, extents, blocks
2 FROM dba_segments
3 WHERE owner = 'HR';
SEGMENT_NAME TABLESPACE
EXTENTS BLOCKS
```

```
REGIONS SAMPLE 1 8
LOCATIONS SAMPLE 1 8
DEPARTMENTS SAMPLE 1 8
JOBS SAMPLE 1 8
EMPLOYEES SAMPLE 1 8
JOB_HISTORY SAMPLE 1 8
5 rows selected.
```