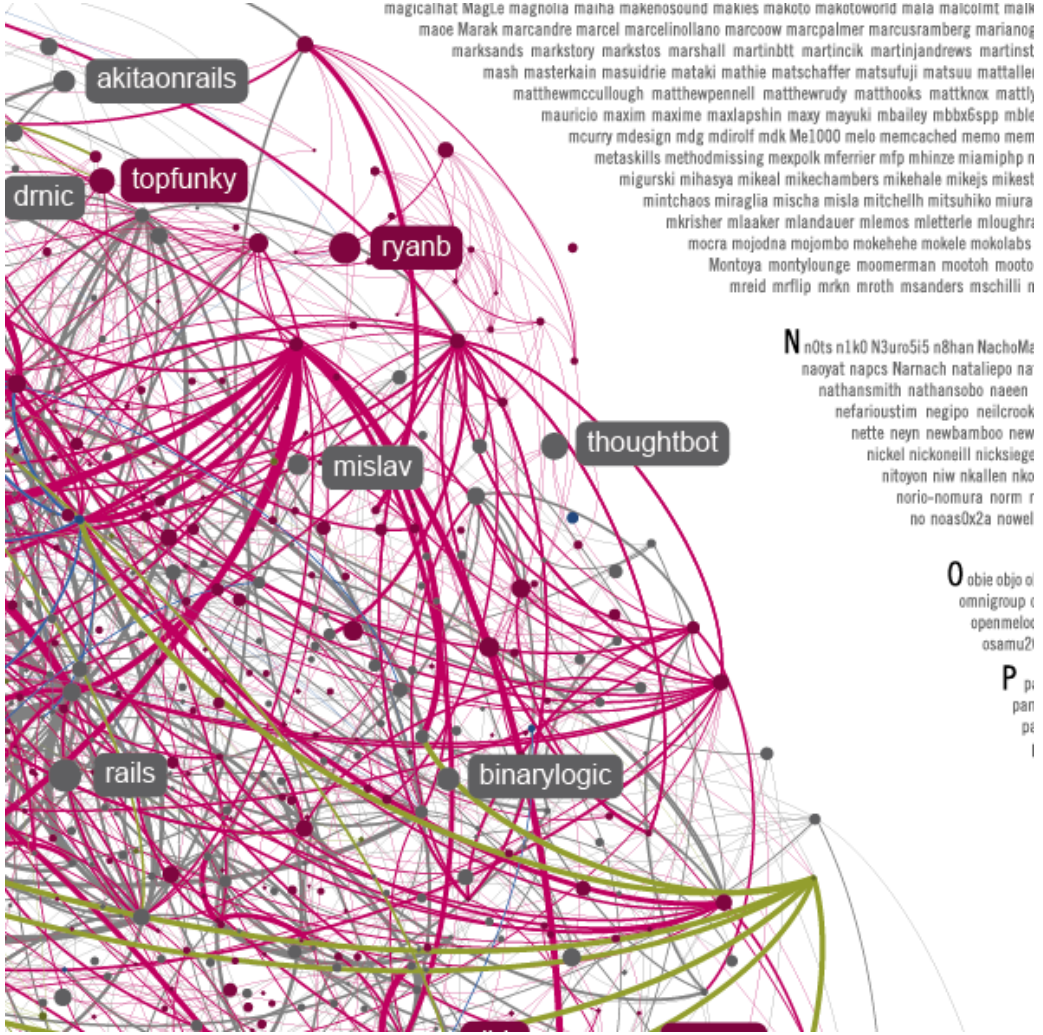
DB2设计与性能优化

第10周



法律声明

【声明】本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

http://edu.dataguru.cn

议程

- 建库
- 日志
- 表空间
- 缓冲池
- 表设计
- 索引设计
- 分区设计

建库（1）

- “CREATE DATABASE mydb”：
  - 自动 runstats 被激活
  - 内存自调优Self Tuning Memory Management 被激活
  - 数据库编码为Unicode
  - Automatic Storage使用单一路径作为容器路径（DBM CFG DFTDBPATH 定义）
    - 可以使用ON 字句定义多个容器路径
  - 元数据和日志文件位于和表空间相同的路径下面（DBM CFG DFTDBPATH）
    - 使用DBPATH ON子句定义不同的路径

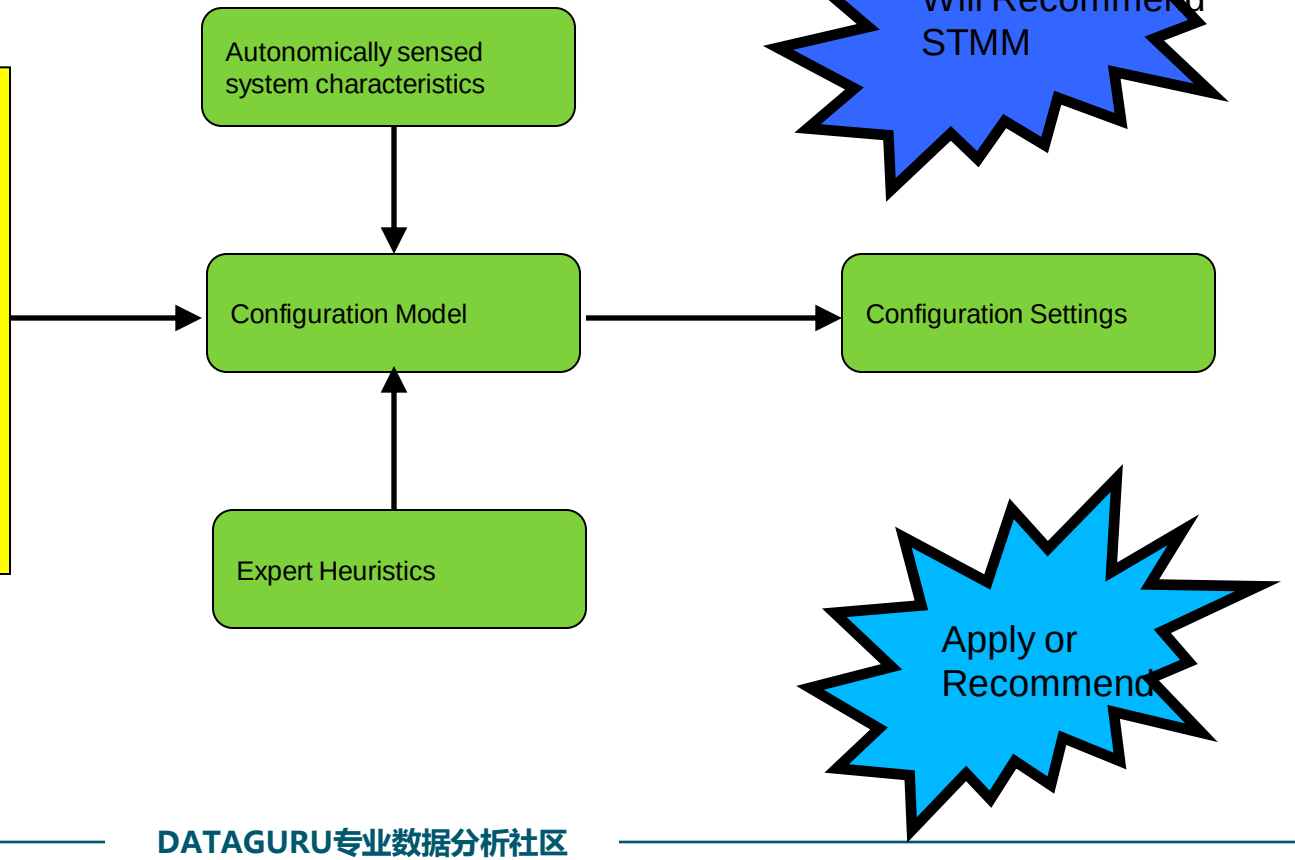
建库（2）

- “CREATE DATABASE mydb” ：
  - 默认页大小为4K
  - 默认表空间:
    - SYSCATSPACE
    - TEMPSPACE1
    - USERSPACE1 （DMS大型表空间）
  - IBMDEFAULTBP缓冲池初始大小为1000页，启用STMM自调优
    - 可能需要多个缓冲池
    - 建议为临时表空间创建独立的缓冲池
  - Currently Committed 被激活
  - DB2DETAILDEADLOCK 事件监控器被创建
    - 可以删除它

自动配置（AUTOCONFIGURE）

数据库创建后，运行该工具

```
DB2 AUTOCONFIGURE USING
  MEM_PERCENT 60
  WORKLOAD_TYPE MIXED
  NUM_STMTS 20
  TPM 500
  ADMIN_PRIORITY BOTH
  IS_POPULATED YES
  NUM_LOCAL_APPS 0
  NUM_REMOTE_APPS 20
  ISOLATION RR
  BP_RESIZEABLE YES
APPLY NONE
```



日志文件

- LOGPATH 或 NEWLOGPATH (DB CFG) – 定义事务日志路径
  - update db cfg for mydb using NEWLOGPATH </path/.../>
  - 或者建库的时候使用DBPATH ON选项
  - 存放日志的磁盘I/O吞吐要好，避免日志访问成为系统瓶颈
  - 要和表空间路径分开，避免磁盘竞争
- LOGBUFSZ (DB CFG) – 定义日志缓冲大小 (以4KB 为单位)
  - 需要定义一个合适的范围
- DIAGSIZE (DBM CFG) – 定义诊断日志文件的总大小 (以M 为单位)

  下面的命令会产生10个诊断日志文件，这些文件会循环使用，每个文件大小为100M（100 = 1024/10）
  - db2 update dbm cfg using DIAGSIZE 1024

表空间管理

- System Managed Space (SMS)
  - 目录容器
  - 使用操作系统的标准I/O访问数据
  - 空间随需分配
  - 管理和监控成本低
  - 适用于小的临时表空间
- Database Managed Space (DMS)
  - 空间预分配
  - 由DB2负责数据访问，提升了性能
  - 增加了管理和监控开销
  - 为了I/O并行性，支持将表数据、LOB数据和索引放置于不同的表空间中
  - 支持大型表空间 (Large RIDs)以提升表容量

自动存储（Automatic Storage）

- 存储完全由DB2管理
- 默认使用Automatic Storage
  - 默认，也是使用单路径
- 使用CREATE DATABASE命令中的ON字句指定多个路径来存放表空间数据
  - 路径可以增加或者删除
- 创建表空间时，不再需要指定容器路径
- DB2负责在指定的路径上创建表空间所需要的容器

如果不使用自动存储

- 使用表空间 –数据、索引和LOB存放于不同的DMS大型表空间中
  - 减少I/O冲突
  - 提升预取性能
- 临时表空间 –大规模数据，使用大型DMS表空间；小规模数据，使用SMS表空间

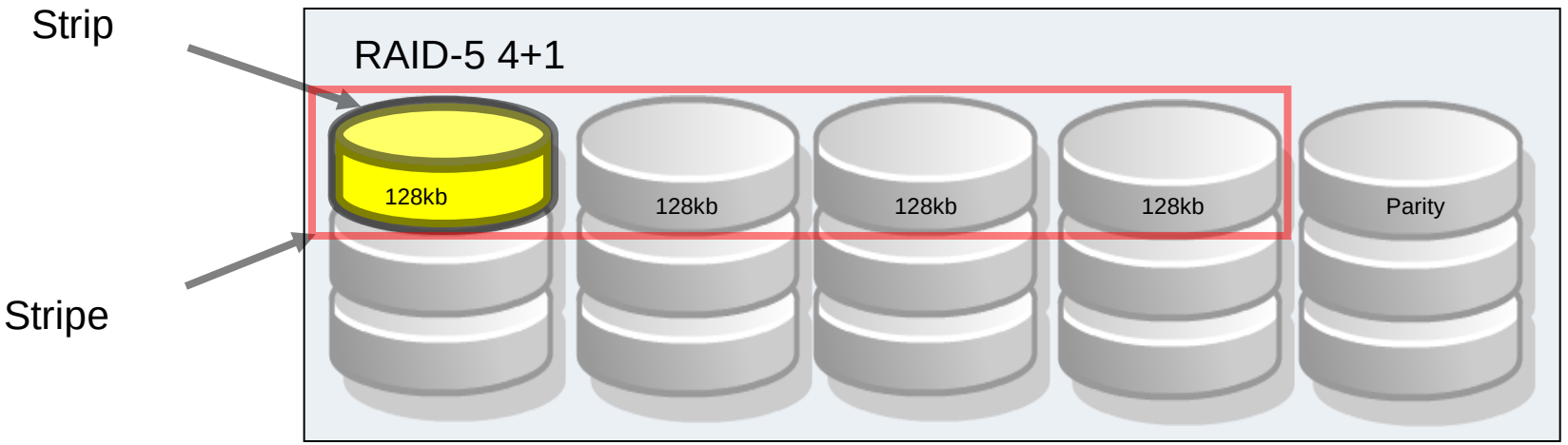
表空间和页大小

- 从DB2 V9开始引入大型DMS表空间
- 根据记录尺寸选择相应的页大小
  - 小尺寸记录的表集中放在小页面的表空间中
  - 大尺寸记录的表集中放在大页面的表空间中

选择扩展数据块大小（Extent sizes）

- 在表空间创建时指定，后续无法改变
- 原则:
  - 大部分情况下，32个页面是最佳选择
  - BI/OLAP 类负载采用更大配置
  - OLTP类负载采用较小配置

根据存储来设定

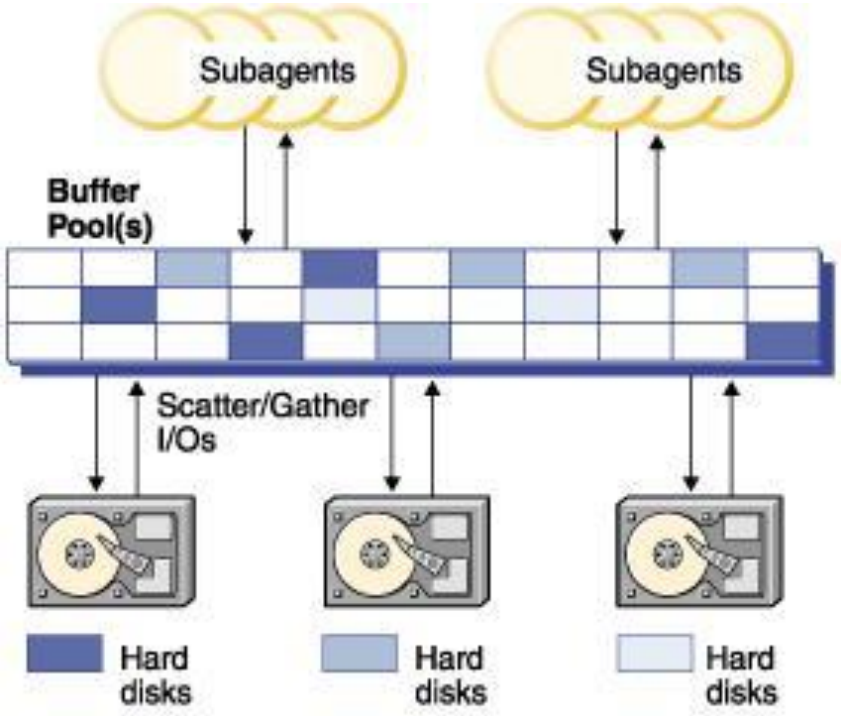


Stripe = 4 (# of data disk) x 128 (Strip Size KB) = 512 KB

$$\text{EXTENTSIZE} = \frac{\text{Stripe (KB)}}{\text{Page Size (KB)}}$$

缓冲池

- 用于缓存表和索引
- 减少访问数据所需要的时间
- 减少I/O竞争
- 每种页面至少需要一个缓冲池对应
- 可以使用STMM来管理



缓冲池创建

- CREATE BUFFERPOOL mybuf
  - 如果STMM被激活, 创建初始大小为1000个页面的缓冲池
  - 使用配置顾问，对创建的缓冲池进行再次配置
- CREATE BUFFERPOOL mybuf SIZE 250
  - 显式制定缓冲池的尺寸
  - 非自调整的缓冲池可以改为自调整的

缓冲池设计

- OLTP – 通常占用可用内存的 75%
  - 根据表空间用途的不同，创建多个不同配置的缓冲池可以提升性能
- BI/OLAP –通常占用可用内存的50%
  - 为常规表空间和用户临时表空间，创建一个缓冲池
  - 为系统临时表空间，创建一个缓冲池

表设计

- CREATE/ALTER TABLE选项
  - 压缩（**COMPRESSION**）：**可以提升I/O密集型负载的性能**
    - 从磁盘或者数据是最慢的操作
    - 更少的I/O 读写可以获取同样多的数据
  - **APPEND ON：能提升大量插入的性能，可以避免搜索可用空间的开销**
    PCTFREE：保留可用空间，以供后续插入、LOAD和REORG
    - 默认是10，针对集群索引或者大量插入类的负载，可以增大到 20-35
    - 如果开启APPEND ON，设定PCTFREE为0
- 分区表（ PARTITION BY RANGE ）：支持快速滚入和滚出，支持分区排除，支持全局索引和分区索引

MDC表

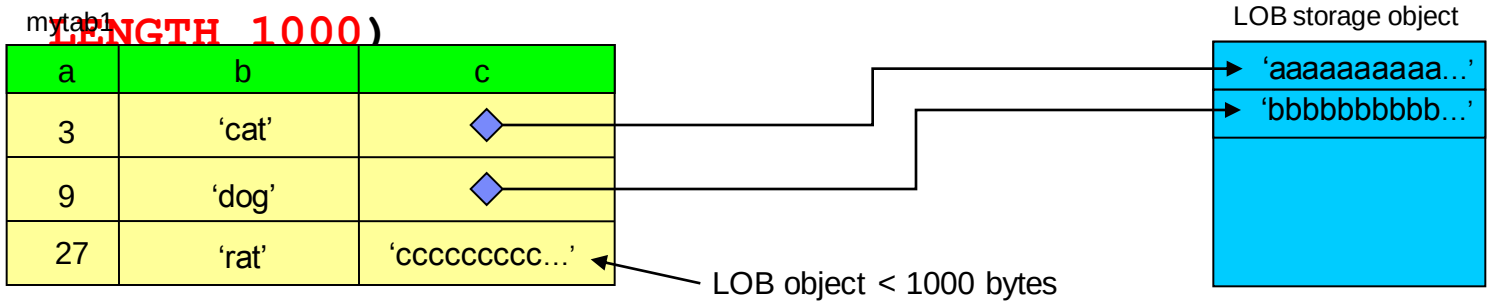
- 按多个维度将数据分块存放
- 用于提升负载查询的性能（BI/OLAP 负载）
- 维度字段：
  - 不要使用唯一字段作为维度，否则表会非常大
  - 要确保有尽量多的记录充满每个数据块

物化视图（MQTs）

- 用于提升诸如GROUP BY、GROUPING、RANK或者ROLLUP OLAP函数的查询性能
- 对用户和应用透明
- 支持实时刷新和延时刷新

DB2 – XML和LOB 内联

- XML 和LOBs 存放在独立的外部存储空间
  - 和页大小有关，一个可供内联对的LOB数据的最大尺寸是32 669 bytes
- 描述符（Descriptors）存放在基表的页面中，用来指向相关联的XML/LOB数据
- 例如:
  - `CREATE TABLE mytab1 (a int, b char(5), c clob INLINE LENGTH 1000)`



为什么使用XML和LOB内联?

性能

- 内联的XML 和 LOB可以缓存到缓冲池中
- XML和LOB数据可以被压缩，以减少I/O开销

存储

- 总的存储空间减少了（尽管基表的空间增加了）

索引设计

- 避免太多索引，
  - OLTP，创建3~5个
  - OLAP/DSS，创建5个左右
- 用组合索引，可以让多个查询受益
  - 将查询中引用最多的列放在定义的前面
- 基数较大的列很适合用来作索引
- 避免添加与已有索引相似的索引
- 如果表是只读的，并且包含很多行，那么可以尝试定义一个索引，通过 INCLUDE 子句使该索引包含查询中引用的所有列
- 要提高对父表执行删除和更新操作，在外键上创建关系索引
- 对于只读表上的索引，PCTFREE为0
- 删除没用的索引
  - 检查 SYSCAT.INDEXES.LASTUSED字段

聚集索引

- 使用带CLUSTER选项的CREATE INDEX语句创建
- 聚集索引使得表中数据按照结果集的需要而有序存放
  - 通常1个表只能建1个聚集索引
- 通常建立在 integer、char(10)或唯一性columns字段上
- 对于聚集索引，PCTFREE应设置大些，例如15~35，以保证聚集索引不会被分成太多的碎片。

如何选择：索引、表分区和全访问?

问题:

在数据库设计中，很多开发人员在使用索引、表分区和MQT时缺乏原则性，导致最终系统上线后麻烦很多。

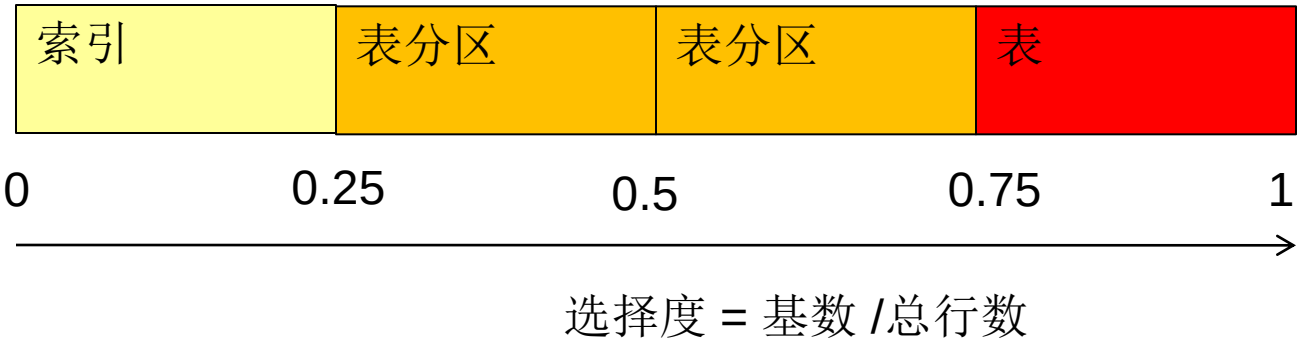
解决办法:

根据查询的选择度（ Selectivity ）来定：

区间（0，0.25），强选择性，考虑使用索引；

区间（0.25，0.75），中选择性，使用分区；

区间（0.75，1），若选择性，使用表访问。



炼数成金逆向收费式网络课程

- Dataguru（炼数成金）是专业数据分析网站，提供教育，媒体，内容，社区，出版，数据分析业务等服务。我们的课程采用新兴的互联网教育形式，独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围，重竞争压力的特点，同时又发挥互联网的威力打破时空限制，把天南地北志同道合的朋友组织在一起交流学习，使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本，直线下降至百元范围，造福大众。我们的目标是：低成本传播高价值知识，构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情，请看我们的培训网站 http://edu.dataguru.cn

Thanks

FAQ时间