重复数据删除(数据去重)

华宇

https://csyhua.github.io/

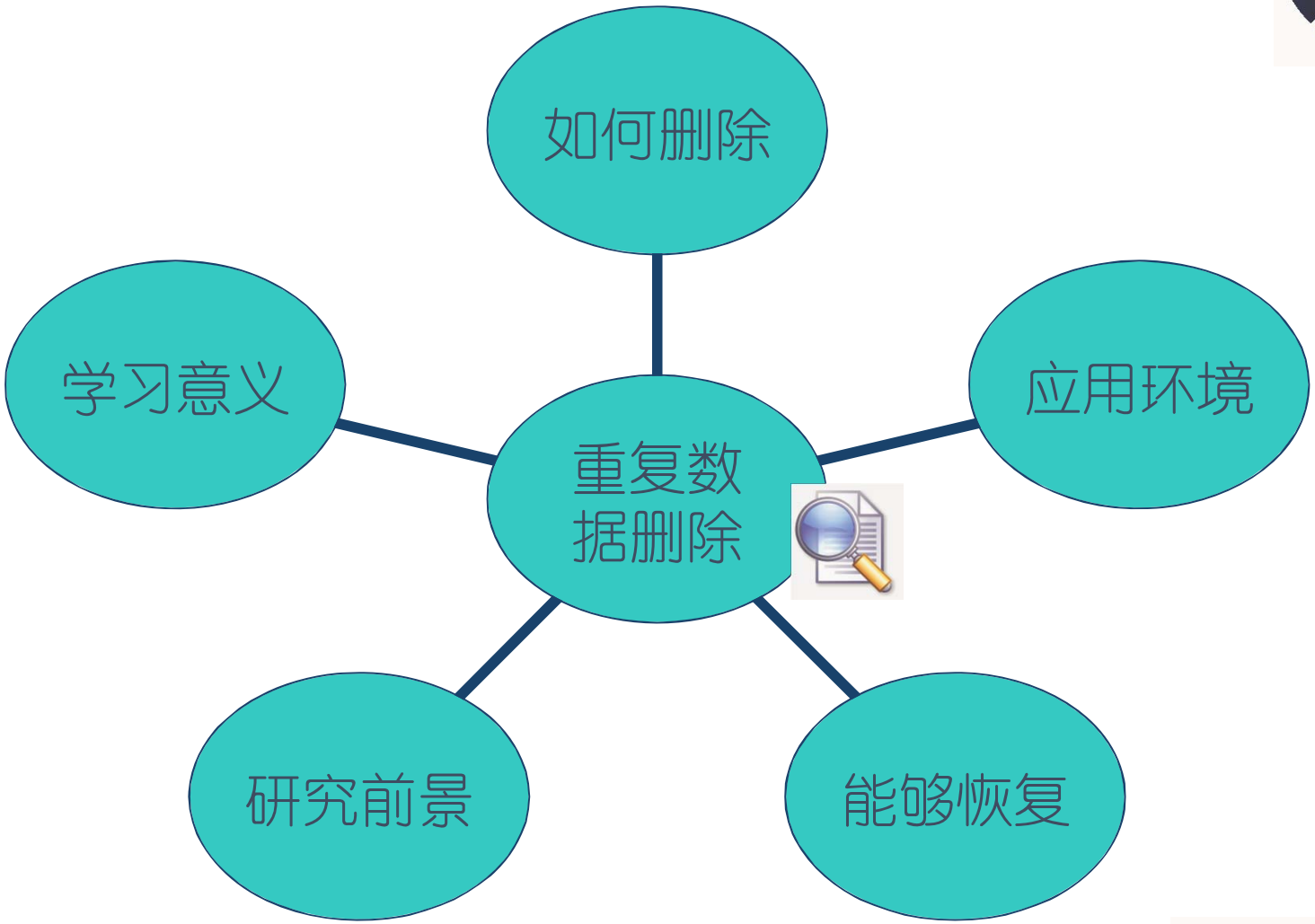
Data Storage & Application Lab

重复数据删除的过去

- 数据挖掘与知识开采领域。
- 论文查重，海量信息检索。
- 承前启后，重复数据删除。

DSAL Data Storage & Application Lab

你们最关心的问题



重复数据删除的优势

- 节省空间。
  - 消除重复数据。
  - 降低存储成本。
- 节省时间。
  - 减少存储传输。
  - 提高存储效率。

DSAL Data Storage & Application Lab

为什么重复数据删除

- 高效地节约存储空间，数据保存时间更长，或者备份更多。
- 减少网络中数据的传输量，也可以提高备份恢复的性能。
- 广域网环境，减少数据传输量的意义就更加明显，可以更容易地实现远程备份或容灾。
- 帮助用户节约时间和成本
  - 数据的恢复速度更快
  - 随着备份存储设备的减少，空间、电力、散热的成本消耗也在降低

DSAL Data Storage & Application Lab

重复数据删除流程

- 文件数据流分块。
- 数据块哈希指纹。
- 指纹查找。
- 数据存储。

Data Storage & Application Lab

重复数据删除应用

- 备份和归档系统。
- 主存储文件系统。
- 内存的缓存设计。
- 虚拟机存储优化。

Data Storage & Application Lab

重复数据删除的粒度

- 文件级。
- 数据块级.
  - 定长分块.
  - 变长分块.
- 粒度越小，重复数据删除所带来的元数据越多.

Data Storage & Application Lab

重复数据删除粒度再分析

- 各种粒度之间的差别包括：运算时间、准确度、重复数据的检测水平、索引的大小、可扩展性

File level
文件级的数据去重（或者叫做“单实例存储SIS”）通过检查文件的属性来确定重复文件。这种方法去重的效果不如其他粒度级别，但是技术比较简单，而且速度快。

Block Level
块级去重是将数据切分成大小相同的块。每个块都被赋予一个“指纹”，通过“指纹”与数据索引（指纹库）的比较判断是否为重复数据。如果块分割的越小，块数量相应就越多，索引也就越多。（产生较高的数据去重比率）。不过，我们还要评估一个重要的指标--就是I/O的压力，它与数据比较的频度成正比，加之数据块越小索引就越大，这可能导致备份性能的下降。

Byte Level
字节级去重过程是通过在新旧文件之间进行逐个字节的比较实现的。虽然这是唯一一种能够保证充分去除冗余的方法，但是对性能的影响却非常大。

重复数据删除范围

- 全局与局部。
- 全局的重复数据删除机制则关注多个存储节点之间的冗余数据。
- 局部的重复数据删除机制仅仅关注同一台机器，或同一个存储节点上的冗余数据，不关心多个存储节点之间存在的冗余数据；

重复数据删除时间

- 在线与离线。

- 在线机制是指在数据到达存储设备之前对相同的数据进行删除，存储设备上仅存储唯一的不重复的数据。

- 离线的实现机制是事先采用一个磁盘缓冲区，先将所有到达的数据缓暂存到一个磁盘缓冲区中，等所有的数据全部写完之后，在系统空闲的时刻，将磁盘缓冲区的数据重新读取出来再查找和删除其重复的数据。

DSAL Data Storage & Application Lab

重复数据删除位置

- 数据的传输和存储分为两端，一个为数据的发送方，即源端；另一个为数据的接收方和存储方，即目标端。
- 源端重复数据删除。
  - 源端实现的重复数据删除机制是指在数据开始传送之前，在源端将重复的数据进行删除，即重复的数据不需要进行传输和存储。
- 目标端重复数据删除。
  - 标端实现重复数据删除技术，是指在目标端的存储设备上删除重复的数据。在这种实现机制下，重复数据删除所带来的实现开销全部集中在目标端，源端不需要做任何的有关于重复数据删除的操作。

Data Storage & Application Lab

变长分块分块算法-1

- 基于内容的分块算法。
- **Rabin** 指纹分块算法。

Hash算法

- MD表示消息摘要(Message Digest，简记为MD)，MD5以512比特一块的方式处理输入的消息文本，每个块又划分为16个32比特的子块。算法的输出是由4个32比特的块组成，将它们级联成一个128比特的摘要值。

Hash碰撞问题

- 重复删除技术通常采用**MD-5 (a 128** 字节的散列) 或 **SHA-1 (a 160**字节的散列) 算法。
- 发生散列冲突的概率小于行星碰撞地球
  - 具有一百万小时MTBF的两个互为镜像的硬盘在一小时内发生故障的可能性是发生hash碰撞的10亿倍
  - 在95 EB（EB，exabyte数据量单位，1EB = 2^{16} Bytes ）数据中存在hash碰撞而导致数据块被误删除
- **hash**碰撞并不意味着数据会全部丢失。数据被错误识别的这个文件会被破坏。所有其它的数据会被正确地恢复。

DSAL Data Storage & Application Lab

SHA-1与MD5的比较

- 两种典型的哈希摘要算法。

	SHA－1	MD5
Hash值长度	160位	128位
分组处理长度	512位	512位
步数	80(4×20)	64(4×16)
最大消息长度	≤2^{64}位	不限
非线性函数	3（第2、4轮相同）	4

指纹索引表

- 索引表大小。
  - 8TB 数据, 20GB 指纹库.
  - 800TB 数据 ,2000GB 指纹库.

DSAL Data Storage & Application Lab

磁盘瓶颈问题

- 磁盘读写速度太慢.
  - 磁盘—典型的计算机系统瓶颈。
  - 频繁访问磁盘索引—不可接受。

- 内存速度？？
- 磁盘速度？？
- CPU缓存？？

重复数据删除难题

- 可扩展性
  - PB级别的备份归档系统.
  - PB级别—TB级的元数据.

- 吞吐率
  - 避免磁盘瓶颈。
  - 重删更好更快

Data Storage & Application Lab

DDFS（2）

- **DDFS潜在的问题。**
  - 内存开销。
  - 写吞吐率。
  - 恢复性能。

ChunkStash（3）

- 内存换速度？？
- **Cuckoo Hash VS Bloom Filter.**

Data Storage & Application Lab

重复数据删除流程总结

- 重删分块算法。
  - 定长&变长
  - 分块大小
- 哈希指纹算法。
  - SHA-1、MD5、
  - 写吞吐率。
  - 文件大小
  - 哈希算法
- 读性能。
  - 文件碎片

DSAL Data Storage & Application Lab

研究状态

- 分块算法，仍然继续。
- 哈希摘要，已经成熟。
- 指纹查找，依旧挑战。
- 数据读取，潜在问题。

DSAL Data Storage & Application Lab

挑战-可扩展性

- 可扩展性。
  - 著名的索引磁盘瓶颈问题。
  - 800TB数据，1TB的指纹。
  - 恢复性能。
  - 能耗问题。

- 解决方案？？

Data Storage & Application Lab

挑战-可靠性

- 可靠性
  - 数据块关联多个文件。
  - 可靠性副本与重删。

- 解决方案？？

总结

- 重复数据删除继续发展。
  - 二级存储系统。
  - 主存储系统。
  - 云存储、云计算。
  - 标准化、规范化。
  - 性价比的优势。

DSAL Data Storage & Application Lab