



FusionInsight HD
V100R002C60SPC200

术语

文档版本 02
发布日期 2017-07-25

版权所有 © 华为技术有限公司 2017。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址：深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址：<http://e.huawei.com>

前言

概述

本文档提供FusionInsight HD产品在整个使用阶段中涉及到的基本概念信息。

读者对象

本指南主要适用于以下读者：

- 规划工程师
- 安装工程师
- 运维工程师
- 软件开发工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	以本标志开始的文本表示有高度潜在危险，如果不能避免，会导致人员死亡或严重伤害。
 警告	以本标志开始的文本表示有中度或低度潜在危险，如果不能避免，可能导致人员轻微或中等伤害。
 注意	以本标志开始的文本表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
 窍门	以本标志开始的文本能帮助您解决某个问题或节省您的时间。
 说明	以本标志开始的文本是正文的附加信息，是对正文的强调和补充。

修订记录

修改记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

文档版本 02 (2017-07-25)

第二次发布。

文档版本 01 (2016-05-13)

第一次发布。

目 录

前 言..... ii

1 HBase..... 1

1.1 B.....2

1.2 D..... 2

1.3 F.....2

1.4 H..... 2

1.5 L.....2

1.6 M.....2

1.7 R.....2

1.8 S.....3

1.9 W..... 3

1.10 X..... 3

1.11 Y.....3

2 HDFS.....4

2.1 A..... 5

2.2 B.....5

2.3 C.....5

2.4 D..... 5

2.5 F.....5

2.6 G..... 5

2.7 H..... 6

2.8 J.....6

2.9 K..... 6

2.10 M.....6

2.11 N.....6

2.12 R.....6

2.13 S..... 7

2.14 T..... 7

2.15 W..... 7

2.16 Y..... 7

2.17 Z..... 7

3 Loader..... 8

3.1 D.....	9
3.2 J.....	9
3.3 N.....	9
3.4 S.....	9
4 MapReduce and YARN.....	10
4.1 A.....	11
4.2 C.....	11
4.3 G.....	11
4.4 H.....	11
4.5 N.....	11
4.6 R.....	11
4.7 Y.....	11
5 Solr.....	12
5.1 B.....	13
5.2 C.....	13
5.3 D.....	13
5.4 F.....	13
5.5 G.....	13
5.6 I.....	13
5.7 J.....	13
5.8 L.....	14
5.9 N.....	14
5.10 P.....	14
5.11 Q.....	14
5.12 S.....	14
5.13 T.....	15
5.14 W.....	15
5.15 X.....	15
5.16 Y.....	15
5.17 Z.....	15
6 Spark.....	17
6.1 D.....	18
6.2 F.....	18
6.3 G.....	18
6.4 H.....	18
6.5 J.....	18
6.6 Q.....	18
6.7 R.....	18
6.8 Y.....	18
6.9 Z.....	19
7 Kafka.....	20

7.1 B.....	21
7.2 F.....	21
7.3 K.....	21
7.4 S.....	21
7.5 T.....	21
7.6 X.....	21
8 Streaming.....	22
8.1 B.....	23
8.2 C.....	23
8.3 L.....	23
8.4 R.....	23
8.5 S.....	23
8.6 T.....	23
8.7 W.....	23
9 Redis.....	24
9.1 R.....	25
10 ZooKeeper.....	26
10.1 G.....	27
10.2 L.....	27
11 其他.....	28
11.1 C.....	29
11.2 D.....	29
11.3 F.....	29
11.4 G.....	29
11.5 H.....	30
11.6 J.....	30
11.7 K.....	31
11.8 M.....	31
11.9 P.....	31
11.10 Q.....	31
11.11 R.....	31
11.12 S.....	31
11.13 T.....	32
11.14 U.....	32
11.15 W.....	32
11.16 X.....	32
11.17 Y.....	32
11.18 Z.....	33

1 HBase

关于本章

[B](#)
[D](#)
[F](#)
[H](#)
[L](#)
[M](#)
[R](#)
[S](#)
[W](#)
[X](#)
[Y](#)

1.1 B

表 HBase的表是三个维度排序的映射。从行主键、列主键和时间戳映射为单元格的值。所有的数据存储在HBase的表单元格中。

1.2 D

单元格 HBase中通过行和列确定的为一个存储单元称为单元格。

1.3 F

分割 HBase中的分区大小是固定的，随着数据量增大，分区会逐渐分割成多个。
分区 类似传统数据库中的分区，每个RegionServer上可以有多个Region。

1.4 H

HFile HBase的文件格式。主键和值为字符串。

1.5 L

列 HBase表的一个维度。列名称的格式为“<family>:<label>”，<family>和<label>为任意字符组合。表由<family>的集合组成（<family>又称为列族）。HBase表中的每个列，都归属于某个列族。

列族 列族是预定义的列集合，且存储在HBase Schema中。如果需要在列族下创建一些列，首先需创建列族。列族将HBase中具有相同性质的数据进行重组，且没有类型的限制。同一列族的每行数据存储在同一个服务器中。每个列族像一个属性，如压缩包、时间戳、数据块缓存等等。

1.6 M

Major Compactions 将一个Store中的所有StoreFile合并成一个，这个过程会删除冗余数据。

MemStore MemStore是HBase存储的核心，当WAL中数据存储达到一定量时，加载到MemStore进行排序存储。

Minor Compactions 将多个小的StoreFile合并成一个StoreFile，这个过程不会删除冗余数据。

1.7 R

RegionServer RegionServer是HBase集群运行在每个工作节点上的服务。一方面维护Region的状态，提供对于Region的管理和服务；另一方面，上传Region的负载信息，参与Master的分布式协调管理。

1.8 S

时间戳	用于索引同一份数据的不同版本，时间戳的类型是 64位整型。时间戳可以由 HBase在数据写入时自动赋值或者由客户显式赋值。
Snappy	是一个压缩库。Snappy的目的是高速有效的压缩。
Store	是HBase存储的核心，一个Store拥有一个MemStore和多个StoreFile，一个Store对应一个分区中表的列族。
StoreFile	HBase数据文件的基本单元。MemStore中数据存储达到一定量时，会生成StoreFile进行存储。
索引	一种数据结构，提高了对数据库表中的数据检索效率。可以使用一个数据库表中的一列或多列，提供了快速随机查找和有效访问有序记录的基础。

1.9 W

WAL	参见 预写日志系统 (write ahead log)
-----	---

1.10 X

协处理器	HBase提供的在RegionServer执行的计算逻辑的接口。协处理器分两种类型，系统协处理器可以全局导入RegionServer上的所有数据表，表协处理器即是用户可以指定一张表使用协处理器。
行	HBase表的一个维度，包含行主键（由任意字符组合成）。表是按照行主键的字母进行排序。行主键用于表的检索。
行键	HBase中标的主键，表按照主键的字母升序排列。
序列化	将对象的状态信息转换为可以存储或传输的形式过程。

1.11 Y

预写日志系统	数据库中一种高效的日志算法。在相同的数据量下，采用预写日志系统（WAL）日志的数据库系统在事务提交时，磁盘写操作只有传统的回滚日志的一半左右，大大提高了数据库磁盘I/O操作的效率，从而提高了数据库的性能。HBase中每次增量加载的数据都是写在WAL中。
--------	--

2 HDFS

关于本章

A
B
C
D
F
G
H
J
K
M
N
R
S
T
W
Y
Z

2.1 A

ACL	参见 访问控制列表 (Access Control List)
安全模式	是HDFS的安全模式，用于管理和维护HDFS集群的安全。

2.2 B

Block Pool	Block Pool是隶属于单个Namespace的块的集合。DataNode存储来自集群中所有块池的块。每个块池都是独立管理的。这就允许一个Namespace为新块生成块ID，而不需要和其他Namespace合作。一个NameNode失效，不会影响DataNode为集群中其他NameNode提供服务。
备份数	HDFS中存储文件的副本数。
备NameNode	是HA框架中的一个或两个NameNode节点，当主NameNode出现故障时负责接管主NameNode的职责。

2.3 C

CAS	参见 中央认证服务 (Central Authentication Service)
Checkpoint node	与Secondary NameNode的作用和配置完全相同，只是启动不同。

2.4 D

单点登录	对于多个相关但又相互独立的软件系统的访问控制的属性。单点登录允许用户登录一次，就可访问所有的软件系统。
DataNode	是HDFS集群的工作节点。根据客户端或者是元数据节点的调度存储和检索数据，定期向元数据及客户端发送他们所存储的文件块的列表。
DN	参见 DataNode (DataNode)

2.5 F

访问控制列表	被授权访问某一资源的实体及其访问权限的列表。用于访问权限的控制。
FTP	参见 文件传输协议 (File Transfer Protocol)
FTPS	参见 基于SSL之上的文件传输协议 (File Transfer Protocol over SSL)
负载均衡	用于将集群中各节点数据量平衡化的工具。

2.6 G

高可用性	一个用于保证服务能够在所有时间保持可用的系统或者设计方案，避免出现错误，且不影响其性能。
------	--

2.7 H

HA	参见 高可用性 (High availability)
HDFS联邦	为了对NameService进行横向扩展，HDFS Federation使用多个独立的NameNode（或Namespace）。这些NameNode是联合的，但它们之间是独立的，且不需要相互合作。

2.8 J

基于SSL之上的文件传输协议	在安全套接层使用标准的FTP协议和指令的一种增强型FTP协议，为FTP协议和数据通道增加了SSL安全功能。
JN	参见 JournalNode (JournalNode)
JournalNode	HA集群下，用于存放日志文件的进程。

2.9 K

KDC	参见 密钥管理中心 (Key Distribution Center)
-----	---

2.10 M

密钥管理中心	是安全principal和key的存储中心，用于Kerberos维护认证的用户名和服务。
--------	--

2.11 N

NameNode	是HDFS集群的主节点。维护着文件树和文件数据的存储位置信息，并处理客户端的文件访问请求。
Namespace Volume	一种独立的（自给自足的）管理单元。一个Namespace和它的Block Pool，合称为“Namespace Volume”。当一个NameNode/Namespace被删除后，DataNode上的相关块池也会被删除。在集群升级时，每个命名空间卷将作为一个整体被升级。
NV	参见 Namespace Volume
NN	参见 NameNode (NameNode)

2.12 R

认证	用于确认一个人或者软件程序身份的功能。
----	---------------------

2.13 S

Secondary NameNode 帮助元数据节点管理元数据的工具。用于镜像备份，或日志与镜像的定期合并。

SSO 参见 [单点登录 \(Single Sign on\)](#)

2.14 T

同分布 将存在关联关系的数据或可能要进行关联操作的数据存储在相同的存储节点上。每个DataNode会存储该集合中所有文件的所有文件块的副本。

2.15 W

位置 用于定义用于写同分布文件中所有文件块的DataNode集合。

文件传输协议 一种TCP/IP协议。该协议使得文件可以通过网络从一台计算机传送到另一台计算机。在FTP传输中，两台计算机必须支持它们各自的FTP角色：一台必须是FTP客户端，另一台是FTP服务器。

文件块 是HDFS中存储的最小逻辑单元。每个HDFS文件由一个或多个文件块存储。所有的文件块存储在DataNode中。

文件块报告 是DataNode定期发送给主NameNode的报告。包含了文件块在DataNode中存储的元数据信息。

文件块副本 一个副本是存储在HDFS中的一些文件块拷贝件。同一个文件块存储多个拷贝件主要用于系统的可用性和容错。

2.16 Y

优选节点 客户端应用会优先选择用于写同分布文件的DataNode集合。

2.17 Z

中央认证服务 它是一种网页单点登录协议。其目的是用户只需提供一次凭据（如用户名和密码），即可访问多个应用程序。在Web应用程序没有获得用户的安全凭据（如密码）的情况下，它也允许Web应用程序对用户进行身份验证。CAS也指实现了这个协议的软件包。

主NameNode 是HA框架中用于接收或执行需求的一个或两个节点。

3 Loader

关于本章

D
J
N
S

3.1 D

导出（数据） 数据从内部数据源流向外部数据源，则为导出数据。

导入（数据） 数据从外部数据源流向内部数据源，则为导入数据。

3.2 J

Job 创建一个导入（导出）任务，则称为一个Job。可以对Job进行创建、启动、停止、复制、编辑和删除操作。每个Job都有一个唯一的Job ID和Job Name。当前Job没有运行完成时，再次提交该Job会失败。

3.3 N

内（外）部数据源 以Hadoop系统为参照，数据源在Hadoop系统内部，则为内部数据源，如HDFS、HBase；数据源在Hadoop系统外部，则为外部数据源，如SFTP服务器、关系型数据库、NFS服务器。

3.4 S

数据源 提供数据的地方，如SFTP服务器、关系型数据库（如Oracle，GaussDB等）、NFS服务器、HDFS、HBase表等。

4 MapReduce and YARN

关于本章

A
C
G
H
N
R
Y

4.1 A

ApplicationMaster 负责相应的调度和协调，结合从ResourceManager获得的资源和NodeManager协同工作来运行和监控任务。

4.2 C

Container 作为资源隔离，当前仅提供Java虚拟机CPU、内存的隔离。

4.3 G

归约 用来保证所有映射的键值对中的每一个共享相同的键组。

4.4 H

混洗 从Map任务输出的数据到Reduce任务的输入数据的过程称为Shuffle。

4.5 N

NodeManager 负责执行应用程序的容器，同时监控应用程序的资源使用情况（CPU，内存，硬盘，网络）并且向ResourceManager汇报。

4.6 R

ResourceManager 集群的资源管理器，基于应用程序对资源的需求进行调度。资源管理器提供一个调度策略的插件，它负责将集群资源分配给多个队列和应用程序。调度插件可以基于现有的能力调度和公平调度模型。

4.7 Y

映射 用来把一组键值对映射成一组新的键值对。

5 Solr

关于本章

- B
- C
- D
- F
- G
- I
- J
- L
- N
- P
- Q
- S
- T
- W
- X
- Y
- Z

5.1 B

布尔运算符 在关键字查询中可以使用AND、OR和NOT等运算符。

5.2 C

层面 对象集的一个方面或特定的一部分，可以用来资源分类。

查询分析器 查询分析器处理用户输入的条件。

词频 指在一个文件中一个单词出现的次数。

5.3 D

倒排索引 倒排索引源于实际应用中需要根据属性的值来查找记录。这种索引表中的每一项都包括一个属性值和具有该属性值的各记录的地址。由于不是由记录来确定属性值，而是由属性值来确定记录的位置，因而称为倒排索引(inverted index)。

5.4 F

分布式搜索 分布式搜索是指在查询时在多个分片上进行处理搜索。

分片 Collection的逻辑分片。每个Shard被化成一个或者多个Replica，通过选举确定哪个是Leader。

副本 Shard的一个拷贝。每个副本存在于Solr的一个Core中。

复制 指将主索引从一个服务器拷贝到另外一个服务器，或者更多的从属服务器和子服务器。

5.5 G

关联 用户对文档进行适当的相关搜索。

5.6 I

IDF 参见 [逆向文件频率 \(Inverse document frequency\)](#)

5.7 J

建议 指可以根据用户的输入，提供可能的查询条件的建议。

节点 一个Java虚拟机实例中运行的Solr。也称为Solr的服务器。

禁用词 一般情况下，禁用词对于用户的查询没有多大意义，但是它是自然语言查询的一部分。禁用词是非常小众的部分，主要包括介词和连词。

集群 在Solr中，集群被视作一个单元，由一系列被管理的Solr节点组成。集群可以包含Core、Collection、Shard和Replica。

5.8 L

乐观并发 也被称为“乐观锁”，这是一种方法，其允许更新文件中当前指数，同时保持锁定或版本控制。

Lucene Lucene是一套用于全文检索和搜寻的开源程式库，由Apache软件基金会支持和提供。Lucene提供了一个简单却强大的应用程序接口，能够做全文索引和搜寻。在Java开发环境里Lucene是一个成熟的免费开源工具。Solr的核心搜索技术便是Lucene。

5.9 N

逆向文件频率 TF-IDF是一种统计方法，用以评估一字词对于一个文件集或一个语料库中的其中一份文件的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。TF-IDF加权的各种形式常被搜索引擎应用，作为文件与用户查询之间相关程度的度量或评级。

5.10 P

拼写检查 拼写检查可以提供替代的搜索条件给用户，用来避免因为拼写错误导致搜索内容少或者没用任何结果。

5.11 Q

请求处理 逻辑和配置参数设定让Solr如何处理传入的“请求”，是否这些请求会返回搜索结果，索引文件，或者是否该处理其他的客户情形。

全体 指同时运行多个ZooKeeper的实例。

5.12 S

事务日志 事务日志确保更新无丢失，即使文档没有索引到磁盘。

Solr Schema (schema.xml) Solr的索引schema。Schema处于Solr主配置目录下的schema.xml文件中。

SolrCloud SolrCloud是Solr4.0版本以后基于Solr和ZooKeeper的分布式搜索方案。SolrCloud是Solr的基于ZooKeeper一种部署方式。Solr可以以多种方式部署，例如单机方式，多机Master-Slaver方式。

SolrConfig (solrconfig.xml) Solr的配置文件。该文件处于Solr主配置目录下。

搜索组件	Solr的查询涉及多个步骤，由多个Solr插件进行，插件被称为搜索组件。查询组件本身是最基本，最常见的。亮点组件或拼写检查组件也很常见。调试组件则在开发中非常流行。Solr提供了丰富的，默认的内置组件列表。开发人员可以指定其他组件来执行，修改它们的执行顺序，甚至开发自定义的搜索组件。在查询中，通过设置solr的默认配置文件中的参数或者其他参数，可以启用或禁用各种组件。
索引	在Solr中，一个或多个文件组合在一起的单个逻辑索引。索引必须有一个结构定义，可以跨越多个索引核。
索引核	也就是Solr Core，一个Solr中包含一个或者多个Solr Core，每个Solr Core可以独立提供索引和查询功能，每个Solr Core对应一个索引或者Collection的Shard，Solr Core的提出是为了增加管理灵活性和共用资源。
索引核重载	指在schema.xml，solrconfig.xml或者其他配置文件发生改变时，Solr会重新初始化。

5.13 T

提交	为了使文档的变化在索引中永久性的改变。在增加文档的情况下，将会先提交后进行搜索。
通配符	通配符是一种特殊语句，主要有星号(*)和问号(?)，用来模糊搜索文件。当查找文件夹时，可以使用它来代替一个或多个真正字符；当不知道真正字符或者懒得输入完整名字时，通常使用通配符代替一个或多个真正的字符。
同义词	同义词是指彼此意义相近的词语。在搜索引擎中实现，同义词可能是单词的缩写，或者输入条件和同义词并没有完全的相似性。

5.14 W

文档	文档，基于底层Lucene的搜索技术，类似于一个SQL行（但可以是任何东西，包括Word文档，网页等） - 在Solr中，是搜索和检索单位。文件可以添加，查询可以识别文件，搜索时，Solr返回文档。
----	---

5.15 X

巡察	在SolrCloud单个节点，负责涉及整个集群的处理措施。它跟踪现有节点和分片的状态，并将分片分配给节点。
----	---

5.16 Y

元数据	指描述数据的数据，对数据及信息资源的描述性信息。
-----	--------------------------

5.17 Z

召回	指搜索引擎获取所有可能的匹配用户的查询的能力。
----	-------------------------

主副本	赢得选举的Shard Replica。每个Shard有多个Replica，这些Replica需要选举来确定一个Leader。选举可以发生在任何时间，但是通常仅在某个Solr实例发生故障时才会触发。当索引documents时，SolrCloud会传递它们到此Shard对应的Leader，Leader再分发它们到全部Shard的Replica。
自动更新	提供只更新一个文件中的一个或者多个字段的方法，该方法无需重新索引整个文档。
字段	文档由一系列字段组成，这些字段都是简单的key-value对。每个字段都有一个类型，并且在schema中定义。Solr非常灵活，在文档中并不是所有的字段都有值。Solr还支持动态字段，即其值和名称未在schema中定义。
自然语言查询	用户的写说作为输入的搜索。

6 Spark

关于本章

D
F
G
H
J
Q
R
Y
Z

6.1 D

动作 是针对数据集的操作，例如返回写过、写入存储等。

6.2 F

分布式弹性数据集 分布式弹性数据集是一种抽象的分布式内存，允许开发者在分布式集群中进行容错计算。

6.3 G

工作 由Spark的Action（例如保存、采集）触发的并行计算，由多个任务组成。

工作节点 集群中用于运行Spark应用的节点。

归约 用来保证所有映射的键值对中的每一个共享相同的键组。

6.4 H

混洗 从Map任务输出的数据到Reduce任务的输入数据的过程称为Shuffle。

6.5 J

阶段 是Spark应用程序中由Shuffle隔开的任务集合。

集群管理 在集群中为外部提供集群资源管理的服务。例如Standalone manager、Mesos、YARN。

6.6 Q

驱动 是用于运行应用的main方法和创建SparkContext的程序。

6.7 R

RDD 参见 [分布式弹性数据集 \(Resilient Distributed Datasets\)](#)

任务 承载业务逻辑的运算单元，是Spark平台中可执行的最小工作单元。

6.8 Y

映射 用来把一组键值对映射成一组新的键值对。

应用 用户使用Java、Scala、Python设计语言定义的程序，包含驱动程序和执行者程序。用于在Spark中实现用户的业务计算逻辑。

6.9 Z

执行者 执行者是工作节点中执行Spark应用的进程。该进程运行任务并在内存或磁盘保存数据。每个应用都有其对应的执行者。

转换 是针对数据集的一种操作，将一个数据集转换成新的数据集。

7 Kafka

关于本章

B
F
K
S
T
X

7.1 B

Broker Kafka集群中的每一个节点服务器称为Broker。

7.2 F

分区 每一个Topic可以被分为多个Partition，每个Partition对应一个可持续追加的有序且不可变的log文件。

7.3 K

Kafka Kafka是一个分布式消息发布订阅系统。

7.4 S

生产者 将消息发往Kafka topic中的角色称为Producer。

7.5 T

Topic Kafka维护的同一类的消息称为一个Topic。

7.6 X

消费者 从Kafka topic中获取消息的角色称为Consumer。

8 Streaming

关于本章

B
C
L
R
S
T
W

8.1 B

Bolt 在一个Topology中接受数据然后执行处理的组件。Bolt可以执行过滤、函数操作、合并、写数据库等任何操作。

8.2 C

持续查询语言 CQL(Continuous Query Language)，持续查询语言，用于数据流上的查询语言。相对于标准的SQL，CQL加入了窗口的概念，使得数据可以一直保存在内存中，由此可以快速进行大量内存计算，CQL的输出结果为数据流在某一时刻的计算结果。

CQL 参见 [持续查询语言 \(Continuous Query Language\)](#)

8.3 L

流 流是一组无穷元素的序列，元素以分布式的机制被并发地产生及处理。

8.4 R

任务 在集群中，Spout或者Bolt作为任务被执行。每个任务对应一个执行线程，而流分组定义了元素如何从一组任务分发到另一组任务中。

8.5 S

Spout 在一个Topology中产生源数据流的组件。通常情况下Spout会从外部数据源中读取数据，然后转换为Topology内部的源数据。

8.6 T

拓扑 拓扑是一个计算流图。其中每个节点包含处理逻辑，而节点间的连线则表明了节点间的数据是如何流动的。

8.7 W

Worker 拓扑在1个或者多个worker进程中被执行。每个worker进程是一个物理JVM，并负责执行该拓扑中所有任务的一个子集。

9 Redis

关于本章

R

9.1 R

Redis 单机版客户端	用来访问Redis单机版服务端的客户端API。
Redis 单机版客户端（管道）	Redis单机版客户端通过管道的方式向Redis服务端一次发送多条命令，并一次取得返回结果，在其使用范围内能够较大提高单个客户端的性能。
Redis单机版服务端	服务端是单个Redis实例提供服务。
Redis服务端	提供服务的一方。
Redis集群版服务端	服务端由多个Redis主实例和若干从实例组成，共同提供服务。
Redis集群版客户端	用来访问Redis集群版服务端的客户端API。
Redis集群版客户端（管道）	同单机版客户端的管道模式类似，集群版客户端也可以使用管道模式一次向集群写入多条命令，提高客户端性能。
Redis客户端	发起读写命令，访问Redis服务的客户端。

10 ZooKeeper

关于本章

G
L

10.1 G

跟随者	负责处理读请求的模块，配合Leader一起进行写请求处理。也可作为Leader的储备，当Leader故障时从Follower当中选举出Leader，避免出现单点故障。
观察者	Observer不参与选举和写请求的投票，只负责处理读请求、并向Leader转发写请求，避免系统处理能力浪费。

10.2 L

领导者	作为ZooKeeper集群的领导者，由各Follower通过Zab协议选举产生。主要负责接受和协调所有写请求，并把写入的信息同步到Follower和Observer。
-----	---

11 其他

关于本章

C
D
F
G
H
J
K
M
P
Q
R
S
T
U
W
X
Y
Z

11.1 C

- 超文本传输安全协议** 运行在TLS（Transport Layer Security）或SSL（Secure Sockets Layer）协议上的HTTP协议。用于建立一个信息安全通道，以提供加密通讯及对网络服务器身份的鉴定。详细请参见RFC2818协议。
- 超文本传送协议** 客户端浏览器或其他程序与Web服务器之间的应用层通信协议。HTTP采用了请求/响应模型。客户端向服务器发送一个请求，请求头包含请求的方法、URL、协议版本以及包含请求修饰符、客户信息和内容类似于MIME的消息结构。服务器以一个状态行作为响应，响应的内容包括消息协议的版本，成功或者错误编码加上包含服务器信息、实体元信息以及可能的实体内容。详细请参见RFC2616协议。
- CLI** 参见 [命令行视图 \(command-line interface\)](#)

11.2 D

- DDL** 参见 [数据定义语言 \(data definition language\)](#)
- DML** 参见 [数据操作语言 \(data manipulation language\)](#)

11.3 F

- 分页** 一个支持分不同页面显示多个数据列表的特性。
- Flume** Flume是一个高可用的，高可靠的，分布式的海量日志采集、聚合和传输的系统，Flume支持在日志系统中定制各类数据发送方，用于收集数据；同时，Flume提供对数据进行简单处理，并写到各种数据接受方（可定制）的能力。
- FTP** 参见 [文件传输协议 \(File Transfer Protocol\)](#)
- FTPS** 参见 [基于SSL之上的文件传输协议 \(File Transfer Protocol over SSL\)](#)
- 服务** 服务对外表现为集群提供的组件业务能力，集群中的每个组件对应一个服务名，提供一种服务。

11.4 G

- 高可用性** 通常指采用主、备两个相同的模块以热备份的方式完成指定功能，以提高可靠性的方式。在主用模块故障时，备用模块会自动接替主用模块执行系统功能。

11.5 H

Hadoop	是一个分布式系统框架。用户可以在不了解分布式底层细节的情况下，开发分布式程序，充分利用了集群的高速运算和存储。Hadoop能够对大量数据以可靠的、高效、可伸缩的方式进行分布式处理。Hadoop是可靠的，因为它假设计算单元和存储会失败，因此维护多个工作数据副本，确保对失败节点重新分布处理；Hadoop是高效的，因为它以并行的方式工作，从而加快处理速度；Hadoop是可伸缩的，能够处理PB级数据。Hadoop主要由HDFS、MapReduce、Hbase和Hive组成。
Hive	Hive是基于Hadoop的一个数据仓库工具，可以将结构化的数据文件映射为一张数据库表，并提供简单的SQL查询功能，可以将SQL语句转换为MapReduce任务进行运行。
Hive查询语言	应用于Hive数据仓库的标准数据查询语言。
HQL	参见 Hive查询语言 (Hive Query Language)
HTTP	参见 超文本传送协议 (Hypertext Transfer Protocol)
HTTPS	参见 超文本传输安全协议 (Hypertext Transfer Protocol Secure)
Hue	Hue是一个能够与Apache Hadoop交互的Web应用程序，一个开源的Apache Hadoop UI，可用于数据分析。它支持文件和任务浏览，Hive、Spark、Hbase、Loader等Apache Hadoop生态圈应用程序。

11.6 J

Java管理扩展	是一个为应用程序、设备、系统等植入管理功能的框架。JMX可以跨越一系列异构操作系统平台和网络传输协议，灵活的开发无缝集成的系统、网络和服务管理应用。
JavaScript对象表示法	一种独立于语言的数据描述格式。它起源于JavaScript语言，由基本的键值对和数组构成。
Java数据库连接	JDBC是Java语言中用来规范客户端程序如何来访问数据库的应用程序接口，提供了诸如查询和更新数据库中数据的方法。JDBC也是Sun Microsystems的商标，它是面向关系型数据库的。
Java虚拟机	程序运行所在的环境。Java虚拟机为Java程序提供一个进行交互的软件计算机。因为Java虚拟机并不是一个真实的计算机，而是存在于软件之中，所以Java程序可以运行在任何物理的计算机平台上。
JDBC	参见 Java数据库连接 (Java database connectivity)
检查点	Checkpoint是一个内部事务，它定期将内存缓冲区的数据或edit日志复制到磁盘，并且将磁盘上的FsImage和edit日志合并以防止edit日志过大。在系统异常的情况下，系统恢复上次checkpoint时所保留的数据。
角色	角色是服务的组成要素，每个服务由一个或多个角色组成。服务通过角色安装到主机（即服务器）上，保证服务正常运行。
集群	将多个服务器集中起来使它们能够像一台服务器一样提供服务的计算机技术。采用集群通常是为了提高系统的稳定性、可靠性、数据处理能力或服务能力。例如，可以减少单点故障、共享存储资源、负荷分担或提高系统性能等。

基于SSL之上的文件传输协议 在安全套接层使用标准的FTP协议和指令的一种增强型FTP协议，为FTP协议和数据通道增加了SSL安全功能。

JMX 参见 [Java管理扩展 \(Java Management Extensions\)](#)

JSON 参见 [JavaScript对象表示法 \(JavaScript object notation\)](#)

JVM 参见 [Java虚拟机 \(Java virtual machine\)](#)

11.7 K

可扩展标记语言 W3C开发的规范。XML是SGML的修简版，是专门为Web文件设计的。它允许设计者创建自己的标签，定义、传输、有效化、以及解析应用软件之间和组织机构之间的数据。

11.8 M

命令行视图 应用程序和用户交互的一种方式，完全基于文本输入和输出。命令通过键盘或类似装置输入，由程序编译并执行。结果是以文本或图形的方式呈现在终端界面。

11.9 P

平衡器 一个Hadoop系统中的后台进程。该进程将数据块从过载的数据节点转移到空闲的数据节点。

11.10 Q

千万亿字节 等同于10的15次方的字节或者1000万亿字节的一个单位。前缀peta表示1000次幂。

11.11 R

RSA 参见 [RSA加密算法 \(Rivest-Shamir-Adleman\)](#)

RSA加密算法 一种非对称加密算法。在公钥加密标准和电子商业中RSA被广泛使用，是目前最有影响力的公钥加密算法。RSA算法基于一个简单的数论事实：将两个大素数相乘十分容易，但对其乘积进行因式分解却极其困难，因此可以将乘积公开作为加密密钥。RSA加密算法能够抵御到目前为止已知的所有密码攻击，已被ISO（International Organization for Standardization）推荐为公钥数据加密标准。

11.12 S

实例 当一个服务的角色安装到主机上，即形成一个实例。每个服务有各自对应的角色实例。

数据操作语言	HQL语言中一种负责对数据库对象进行操作的指令集，以INSERT、UPDATE、DELETE三种指令为核心，分别代表插入、更新与删除，是开发以数据为中心的应用程序必定会使用到的指令。
数据定义语言	HQL语言中一种负责数据结构定义与数据库对象定义的语言，由CREATE、ALTER与DROP三个语法所组成，最早由Codasyl（Conference on Data Systems Languages）数据模型开始发展，现已被纳入SQL指令并作为其中的一个子集。
数据节点	Hadoop分布式文件系统的工作进程，运行在工作节点上。从客户端或者NameNode接受到命令后，DataNode存储或读取数据块。同时，DataNode定期将存储的数据块清单报告给NameNode。
Snappy	是一个压缩库。Snappy的目的是高速有效的压缩。

11.13 T

Tomcat	是一个由Apache软件基金会、Sun、其他公司以及个人共同开发而成的免费的、开放源代码的Web应用服务器。Tomcat是一个小型的轻量级应用服务器，由于SUN的参与，总能支持最新的Servlet和JavaServer Page（JSP）规范，并且技术先进、性能稳定、扩展性好、运行时占用的系统资源小，因此在中小型系统和并发访问用户不多时被普遍应用。Tomcat本身内含了一个HTTP服务器，可被视为一个单独的Web服务器，并包含了一个配置管理工具，也可以通过编辑XML格式的配置文件来进行配置。Tomcat虽由Apache软件基金会开发，但是与Apache Web服务器却不相同，后者是使用C语言实现的HTTP服务器。
通用资源标识符	通用资源标识符是在注册的命名空间或地址中的通用集合名称中的一员。通用资源标志符用于定位Web上可用的每种资源，包括HTML文档、图像、视频片段、程序等。

11.14 U

URI	参见 通用资源标识符 (uniform resource identifier)
-----	--

11.15 W

文件传输协议	一种TCP/IP协议。该协议使得文件可以通过网络从一台计算机传送到另一台计算机。在FTP传输中，两台计算机必须支持它们各自的FTP角色：一台必须是FTP客户端，另一台是FTP服务器。
--------	---

11.16 X

XML	参见 可扩展标记语言 (Extensible Markup Language)
序列化	将对象的状态信息转换为可以存储或传输的形式过程。

11.17 Y

用户自定义函数	用户自定义实现的SQL函数。
---------	----------------

元数据

元数据（Metadata），又称中介数据、中继数据，为描述数据的数据（data about data），主要是描述数据属性（property）的信息，用来支持如指示存储位置、历史数据、资源查找、文件纪录等功能。

11.18 Z

ZooKeeper

是一个高可用性、开放源码的分布式应用程序协调服务。用于实现分布式应用程序的同步服务、配置维护和命名服务等。ZooKeeper能提供基于类似于文件系统的目录节点树方式的数据存储，但是它并不是用来专门存储数据的，它的主要作用是用来维护和监控所存储数据的状态变化，从而达到数据的集群管理。ZooKeeper可以帮助系统避免单点故障，从而建立可靠的应用程序。

作业

客户端提交的一次作业。包括数据存储、Map任务和Reduce任务的执行以及信息配置。Hadoop将作业分成两类，分别是Map任务和Reduce任务。