



# **《网络存储技术》 课程作业二**

## **SAN 技术概述**

付容天 学号 2020211616

班级 2020211310

计算机科学与技术系

计算机学院（国家示范性软件学院）

2022 年 10 月 15 日

# 目录

<b>1 引言</b>	<b>3</b>
<b>2 SAN 概要</b>	<b>3</b>
2.1 发展历程介绍 . . . . .	3
2.2 SAN 概念成分 . . . . .	4
2.2.1 Host Layer . . . . .	4
2.2.2 Fabric Layer . . . . .	4
2.2.3 Storage Layer . . . . .	5
2.3 SAN 与 DAS 的比较 . . . . .	5
2.4 SAN 和 NAS 的比较 . . . . .	6
<b>3 SAN 技术介绍</b>	<b>7</b>
3.1 SCSI . . . . .	7
3.1.1 SCSI 概要 . . . . .	7
3.1.2 SCSI 逻辑拓扑 . . . . .	7
3.1.3 SCSI 寻址过程 . . . . .	9
3.1.4 SCSI 总线概念 . . . . .	10
3.1.5 SCSI 的读操作和写操作 . . . . .	11
3.1.6 SCSI 类型 . . . . .	12
3.1.7 SCSI 接口 . . . . .	13
3.2 iSCSI . . . . .	14
3.2.1 iSCSI 优势功能介绍 . . . . .	15
3.2.2 iSCSI 工作原理 . . . . .	15
3.2.3 iSCSI 的实际实现 . . . . .	18
3.3 FC 与 FCP . . . . .	19
3.3.1 FC 与 FCP 概要 . . . . .	19
3.3.2 光纤通道发展历史 . . . . .	20

3.3.3	FC 网络拓扑 . . . . .	22
3.3.4	FC 帧结构 . . . . .	23
3.3.5	FC 流量控制 . . . . .	24
3.3.6	FC Fabric 介绍 . . . . .	24
3.3.7	FC Fabric 寻址 . . . . .	25
3.3.8	FC Fabric 登录 . . . . .	26
3.4	FC SAN . . . . .	27
3.4.1	FC SAN 概要 . . . . .	27
3.4.2	FC SAN 与 IP SAN 的比较 . . . . .	27
<b>4</b>	<b>结语</b>	<b>29</b>

# 1 引言

存储区域网络 (Storage Area Network, SAN) 是一种计算机网络, 它提供对整合的块级数据存储的访问。SAN 采用一种叫做“网状通道 (Fibre Channel, FC)”的技术, 这种技术通过 FC 交换机连接存储阵列和服务器主机, 从而建立专用于数据存储的区域网络。虽然各厂商的光纤交换技术并不完全相同, 但是经过产业界十多年的发展, SAN 技术已经相当成熟, 并在业界标准上达成了某种程度上的一致, 这极大推动了 SAN 技术的发展。

SAN 主要用于从服务器访问数据存储的设备, 例如磁盘阵列和磁带库, 从而使这些设备在概念上具备了“直连存储”的特性。注意, 本质上 SAN 技术仅提供块级访问, 但是构建在 SAN 之上的文件系统则可以提供文件级的访问, 这个文件系统被称为“共享文件系统”。当前企业级存储面临的两个最突出的问题是: 数据与应用系统紧密结合所产生的结构性限制、以及小型计算机接口 (SCSI) 标准带来的限制, 由于 SAN 技术拥有便于集成、能改善数据可用性和优秀的网络性能, 因此 SAN 被广泛认为是未来企业级存储问题的有效解决方案。

## 2 SAN 概要

### 2.1 发展历程介绍

SAN 技术的前身是集中式数据存储方案, 现在, 由于 SAN 往往作为一种计算机网络提供对整合的块数据的访问, 因此其可被称为服务器背后的网络。除了存储数据之外, SAN 还允许数据的自动等操作。

一般来讲, SAN 是硬件和软件的组合, 它源于以数据为中心的大型机架构, 其中网络中的客户端可以连接到多个服务器并存储不同类型的数据。随着数据量的不断增加, DAS 技术作为解决方案而出现, 该技术通过将磁盘阵列或其他成组存储设备连接到服务器上来增加容量, 但是很快发现 DAS 技术的致命缺陷是“单点故障问题”, 这意味着存储设备中一点失效会导致系统整体出现极大的问题。

尽管 DAS 作为第一个成熟的大规模存储架构得到了极为广泛的应用, 但是对于数据存储要求很高的地方仍然具备一定的局限性, 因此人们开发了网络附加存储 (NAS) 技术架构。在 NAS 技术中, 一个或多个专用文件服务器或专用存储设备在 LAN 中激活并使用, 提供文件级的服务。这种分布式的设计思路在一定程度上解决了 DAS 技术中的“单点故障问题”。但是, 随着信息时代的不断发展, 数据传输、尤其是数据备份的要求越来越高, 面对更多的数据和更高的要求, NAS 技术逐渐显得有些力不从心。

因此, 经过仔细设计的 SAN 技术正式问世, 它通过将专用存储网络连接到 LAN、以及专用高速和带宽网络传输 TB 级别的数据来提供更加强大和稳健的数据服务。在 SAN 中, 互相连接的存储设备使彼此的数据传输 (例如备份) 的实现隐藏在了服务器后面 (并且对于普通用户是透明的), 比如, 数据可以通过

TCP/IP 协议进行传输。SAN 配备了不同的支持协议，包括光纤通道 (FC)、SCSI 与 iSCSI、Infiniband 等。但是，SAN 通常有自己的网络和存储设备，必须单独购买、安装和配置。这使得 SAN 架构比 NAS 架构和 DAS 架构都更昂贵。

## 2.2 SAN 概念成分

SAN 通常有自己的网络设备，例如 SAN 交换机。为了访问 SAN，引入了 SAN 服务器的概念，这些服务器又连接到对应的 SAN 主机适配器。SAN 支持互连一系列数据存储设备，例如为 SAN 技术定制的磁盘阵列、JBODS 和磁带库等。

### 2.2.1 Host Layer

允许访问 SAN 系统及其相应存储设备的服务器被称为 Host Layer，这类服务器通常具有主机适配器，它们通常是一张连接到服务器主板上的插槽（通常是 PCI 插槽）的卡，这张卡与相应的固件和设备驱动程序一起运行，从而提供必不可少的访问服务。显然，服务器的操作系统可以通过主机适配器及硬件卡与 SAN 中的存储设备进行通信。

在光纤通道解决方案中，千兆接口转换器 (GigaBit Interface Converter, GBIC) 用于 SAN 内的交换机和存储设备，从而将数字位转换为光脉冲，然后将它们发送到光纤通道进行传输。在到达目的地时，GBIC 又可以将传入的光脉冲转换回数字位，从而完成信息解码的工作。

### 2.2.2 Fabric Layer

Fabric Layer 由 SAN 技术中提供网络服务的设备组成，包括 SAN 交换机、路由器、协议桥、网关设备和电缆等。SAN 网络设备在 SAN 内或在发送方（如服务器的 HBA 端口）与目标方（如存储设备的端口）之间移动数据。

最初构建 SAN 时，集线器是唯一具有光纤通道功能的设备，但随着光纤通道交换机被开发并不断完善，现在 SAN 技术已经将集线器淘汰了。与集线器相比，交换机可以提供一条专用链路来将其所有端口相互连接，因此交换机最大的优势就在于它们允许所有连接的设备同时进行通信，这是集线器所不能做到的。

为了保证系统的稳定性，SAN 的实际构建往往遵循冗余性原则。例如，单个 SAN 交换机可以有最少 8 个端口，最多可以有 32 个带有模块化扩展的端口；导向器级交换机则可以有多达 128 个端口。此外，SAN 交换机非阻塞地将服务器与存储设备连接起来，允许同时通过所有连接的线路传输数据。

支持交换技术的 SAN 实现中使用了光纤通道交换结构协议 FC-SW-6，在该协议下，SAN 中的每个设备在主机总线适配器 (HBA) 中都有一个硬编码的全球名称 (WWN) 地址。如果设备连接到 SAN，则它的 WWN 会在 SAN 交换机名称服务器中注册。SAN 光纤通道存储设备供应商还可以硬编码全球节点名称

(WWNN)。存储设备的端口通常有一个以 5 开头的 WWN，而服务器的总线适配器则以 10 或 21 开头。

### 2.2.3 Storage Layer

SAN 中的各种存储设备构成了它的 Storage Layer。在 SAN 中，磁盘阵列通过 RAID 技术进行连接，这使得许多硬盘看起来和执行起来就像一个大型存储设备。

各种存储数据的硬盘和磁带设备都有一个逻辑号 (LUN)，这个编号在 SAN 中是唯一的。SAN 中的每个节点、无论是服务器还是其他存储设备，都可以通过引用 LUN 来访问存储设备。并且，LUN 允许对 SAN 的存储容量进行分段并实施访问控制。例如，一个特定的服务器或一组服务器可以仅被授予对某些 LUN 指定的 SAN 存储层的特定部分的访问权。当存储设备接收到读取或写入数据的请求时，它将检查其访问列表以确定由其 LUN 标识的节点是否被允许访问同样由 LUN 标识的存储区域。

LUN 屏蔽是一种技术，通过该技术，主机总线适配器和服务器的 SAN 软件限制接受命令的 LUN。这样做时，服务器永远不应访问的 LUN 将被绝对屏蔽。另一种限制服务器访问特定 SAN 存储设备的方法是基于结构的访问控制或分区，由 SAN 网络设备和服务器强制执行。在分区下，服务器访问仅限于特定 SAN 区域中的存储设备。

## 2.3 SAN 与 DAS 的比较

DAS (Direct Attached Storage, 直连式存储) 技术是一种将数据直接存储在本地 (未接入网络) 的存储设备上的存储方式，其通常由机械硬盘、固态硬盘、光盘等存储单元构成。一个典型的 DAS 设备通常通过 HBA (Host Bus Adapter, 主机总线适配器) 直接连接到计算机上，通常作为已有存储设备的拓展，这决定了 DAS 技术不具有易于在不同主机之间直接进行数据共享的特点，因为在不同主机之间没有类似于 hub、switch、router 等网络设备进行连接。

和 DAS 相比，SAN 具有的优点包括：

- 通过网络连接，便于进行不同主机之间的数据共享，尤其是数据的备份，其可以隐藏在服务器后透明地完成，极大方便了用户的使用
- 当网络情况稳定时，SAN 提供的服务往往具有更好的性能，因为 SAN 设备可以针对文件服务进行精确调整

而 SAN 相比 DAS 的缺点包括：

- SAN 提供的文件服务不如 DAS 稳定，SAN 提供的服务往往和当前网络的速度和拥塞程度有极大的关系，而 DAS 服务的质量则仅取决于本地计算机的硬件配置，其服务质量是可以在一定程度上得到保证的

- DAS 可以提供更加具有特色的存储服务，往往支持对硬件和低级软件的定制；而 SAN 受限于网络连接要求，其硬件基础往往是不支持定制的
- 块级存储固有缺陷决定了文件访问时具有的局限性，所以 SAN 往往需要搭配“共享文件系统”进行使用

## 2.4 SAN 和 NAS 的比较

NAS (Network Attached Storage, 网络附加存储) 技术是一种将分布、独立的数据整合为大型、集中化管理的数据中心、从而便于不同主机和应用对服务器进行访问的文件级技术。简单来讲，NAS 就是一种“将文件数据存储在网上供主机进行存取和处理”的技术。NAS 可以被定义为一种特殊的专用数据存储服务器，其最重要的功能就是跨平台文件共享功能。NAS 通常在一个 LAN 上占有自己的节点，无需其他应用服务器的干预，并允许用户从该网络上进行数据的存取、更新、删除等操作。在这种配置中，NAS 集中管理和处理所在局域网上的所有数据，有效降低数据管理和共享的成本。

NAS 通常占用所在 LAN 上的一个专用节点，从而对其他其他主机提供数据服务。从此角度来看，不妨认为 NAS 服务器是一种专用的数据服务器，它可以授权网络用户和异质客户端从集中位置存储和检索数据。显然，NAS 具有灵活和易于横向扩展的特点。NAS 的专用目的决定了其硬件和软件基础也通常是专用的，其硬件、软件和配置共同构成其颇具特色的文件服务。NAS 的实现基础通常包含一个或多个通常排列成逻辑存储器、冗余存储器或 RAID 存储驱动器的网络设备。通常来讲，用于 NAS 的硬盘驱动器在功能上与其他驱动器相似，但在固件、振动耐受性或功耗上有不同之处，以使其更适合在 RAID 阵列中使用。

NAS 支持的协议包括 Andrew File System、Apple Filing Protocol、Server Message Block、File Transfer Protocol、SSH File Transfer Protocol、Hypertext Transfer Protocol、Network File System 等协议，这些协议往往在不同领域具有各自的优势，在不同的使用场景中使用了网络附加存储 (NAS) 的概念。

和 NAS 相比，SAN 具有如下优点：

- SAN 架构的速度往往高于 NAS 架构，这是因为 NAS 架构大量使用以太网和 TCP/IP 协议进行内存传输，这样做不但增加了大量的 CPU 指令周期，而且使用了低速传输介质，但 SAN 中大部分操作都由适配卡上的硬件完成，CPU 开销的增加有限
- 由于目前 NAS 的根本瓶颈是底层链路的速度，因此当底层链路速度较低或不稳定时，SAN 架构所能提供服务的质量显著优于 NAS 架构

而 SAN 相比 NAS 的缺点包括：

- 造价较高，在 SAN 中必须使用专有协议例如 Fibre Channel、iSCSI 和 Infiniband 等，因此 SAN 通常有自己的网络和存储设备（尽管 iSCSI 可以运

行在现有设备上,但这会导致 SAN 性能下降),必须单独购买、安装和配置,使得 SAN 技术的安装和使用比较昂贵。但在 NAS 架构中,数据通过成熟的 TCP/IP 协议进行传输,这可以利用已有的硬件基础

- NAS 提供统一的文件系统,方便文件的存储和共享,而 SAN 仅提供基于块的存储,这将文件系统问题遗留给了用户端,虽然 SAN 往往提供匹配的共享文件系统,但这种方案所能提供的服务及其稳定性仍与 NAS 有一定的差距

需要注意的是,尽管存在差异,但 SAN 和 NAS 并不相互排斥,并且可以组合为 SAN-NAS 混合体,从而提供来自同一系统的文件级协议(NAS)和块级协议(SAN)。

## 3 SAN 技术介绍

### 3.1 SCSI

#### 3.1.1 SCSI 概要

Small Computer System Interface (SCSI) 是一组用在计算机和外围设备之间进行物理连接和传输数据的标准,该标准定义了命令、协议、电气以及光学和逻辑接口。理论上,SCSI 可以用于几乎任何设备的接口上,这是因为 SCSI 协议本质上和传输介质无关,SCSI 可以在多种介质上实现(甚至是虚拟介质)。最初的并行 SCSI 常用于硬盘驱动器和磁带驱动器,但它也可以广泛连接其他设备,包括扫描仪和 CD 驱动器。

最早的 SCSI 标准 X3.131-1986,一般简称为 SCSI-1,是由美国国家标准协会(American National Standards Institute, ANSI)的 X3T9 技术委员会于 1986 年发布。SCSI-2 于 1990 年 8 月发布为 X3.T9.2/86-109,在 1994 年进行了进一步修订,随后采用了多种接口。随着发展的不断推进,SCSI 性能得到了持续不断的改进,其对不断增加的数据存储容量的支持也越来越强大。

#### 3.1.2 SCSI 逻辑拓扑

SCSI 协议在设计上是颇为巧妙的,通过将存储系统进行巧妙地“分层设计”,从而提高了系统的稳定性和提供服务的质量。



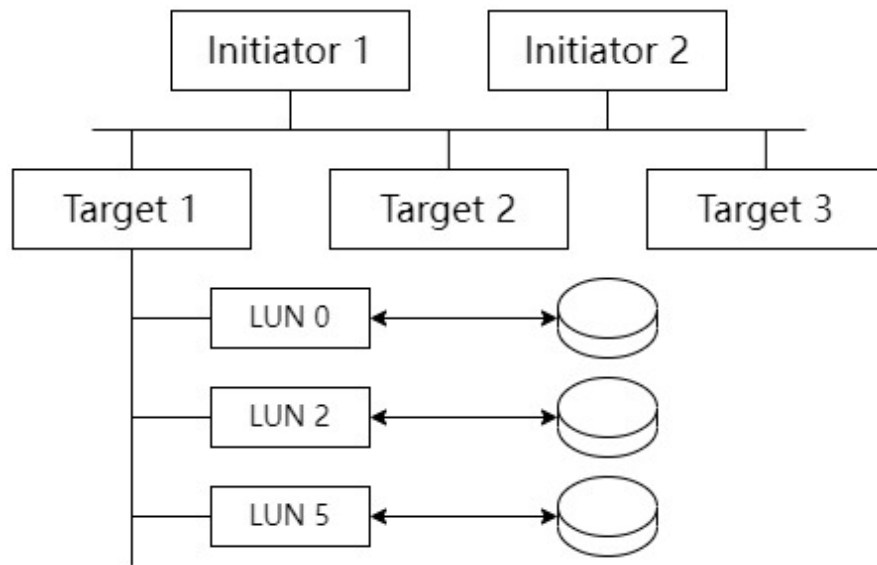


图 1: SCSI 逻辑拓扑概念图

上图展示了 SCSI 技术的逻辑拓扑图，其中：

- 逻辑单元 (LUN): LUN 是 SCSI 目标器中所描述的名字空间资源，一个目标器可以包括多个 LUN，而且每个 LUN 的属性可以有所区别，比如 LUN#0 可以是磁盘，LUN#1 可以是其他设备
- 启动器 (Initiator): SCSI 是一个 C/S 架构，其中客户端为 Initiator，负责向 SCSI 目标器发送请求指令，一般主机系统都充当了启动器的角色
- 目标器 (Target): 处理 SCSI 指令的服务端称为目标器，它接收来自主机的指令并解析处理，比如磁盘阵列的角色就是目标器。SCSI 的 Initiator 与 Target 共同构成了一个典型的 C/S 模型，每个指令都是“请求/应答”这样的模型来实现

对于 Initiator 而言，从 SCSI 的体系结构来说，一共有架构层（中间层）、设备层、传输层三个层级，因此无论 Initiator 上运行的操作系统是什么，SCSI 都分为三个层次，它们之间的相互关系如下图所示：

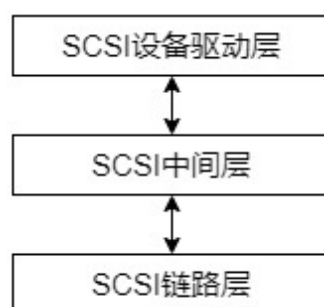


图 2: SCSI 的 Initiator 层次概念图

例如，Windows 下的 Initiator 分为三个逻辑层次，其中 ScsiPort 负责实现 SCSI 的基本框架性处理流程，比如设备发现，名字空间扫描等；Linux 下的启动器架构则将 SCSI 的 Initiator 分为三个逻辑层次，其中 scsi\_mod 中间层复杂处理 SCSI 设备无关和适配器无关的流程处理，比如一些异常，名字空间维护等。HBA 驱动提供 SCSI 指令的打包解包传输等链路实现细节，设备驱动实现特定的 SCSI 设备驱动，比如著名的 sd（SCSI 磁盘）驱动、st（SCSI 磁带）驱动、sr（SCSI 光盘设备）驱动等；AIX 下的 Initiator 架构同样分为三层，即：SCSI 设备驱动层、SCSI 中间层和 SCSI 适配驱动层。

对于 Target 而言，往往也参考 SCSI 体系结构将 Target 分为三个层次，分别是链路端口层、中间层和目标设备层，它们之间的相互关系如下图所示。其中最重要的是中间层，在中间层中将以 SAM/SPC 为规范，对 LUN 命名空间，链路端口，目标设备，任务，任务集，会话等进行管理维护。端口层的驱动都以注册的形式动态载入，设备层的驱动也是动态载入。

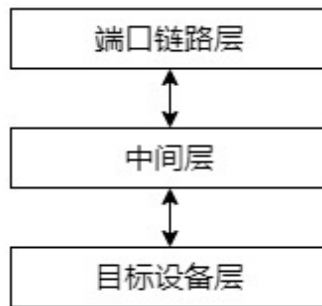


图 3: SCSI 的 Target 层次概念图

### 3.1.3 SCSI 寻址过程

为了对连接在 SCSI 总线上的设备进行寻址，SCSI 协议引入了 SCSI 设备 ID 和逻辑单元号 LUN (Logical Unit Number)。在 SCSI 总线上的每个设备都必须有一个唯一的设备 ID，当然服务器中的主机总线适配器也拥有自己的设备 ID，固定为 7。每条总线，包括总线适配器，最多允许有 8 个或者 16 个设备 ID。设备 ID 一方面用以寻址，另一个作用是标识该设备在总线使用上的优先级。此外，在同一条总线上连接的不同的设备的设备 ID 必须不同，否则就会引起寻址和优先级的冲突。

每一个存储设备可能包括若干个子设备，如虚拟磁盘、磁带驱动器等。因此 SCSI 协议引入了逻辑单元号 LUN ID，以便于对存储设备中的子设备进行寻址。

传统的 SCSI 控制器连接单条总线，相应的只具有一个总线号。企业级的一个服务器则可能配置了多个 SCSI 控制器，从而就可能有多条 SCSI 总线。在引入存储网络之后，每个 FC HBA (Host Bus Adapter) 或 iSCSI (Internet SCSI) 网卡也都各连接着一条总线，因此必须对每一条总线分配一个总线号，在他们之间依靠不同的总线号加以区分。我们可以使用一个三元描述标识一个 SCSI 目标：总线号/目标设备 ID/逻辑单元号 LUN ID，它们之间的关系如下图所示。

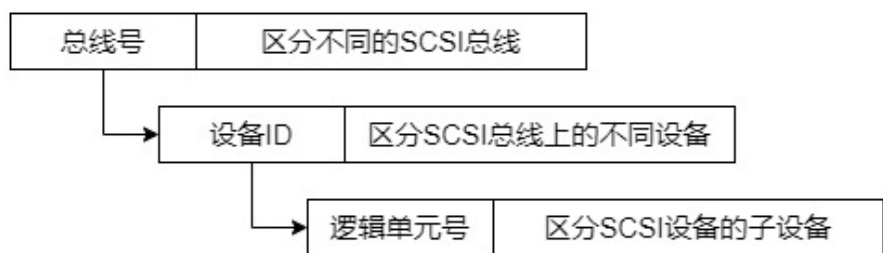


图 4: SCSI 寻址概念图

### 3.1.4 SCSI 总线概念

在 SCSI 工作时，控制器首先向总线处理器发出请求使用总线的信号。该请求被接受之后，控制器高速缓存就开始执行发送操作。在这个过程中，控制器占用了总线，总线上所连接的其它设备都不能使用总线。当然，由于总线具备中断功能，所以总线处理器可以随时中断这一传输过程并将总线控制权交给其它设备，以便执行更高优先级的操作。

SCSI 控制器相当于一个小型 CPU，有自己的命令集和缓存。SCSI 是一种特殊的总线结构，可以对计算机中的多个设备进行动态分工操作，对于系统同时要求的多个任务可以灵活机动的适当分配，动态完成。SCSI 总线接入计算机总线的概念图如下所示。

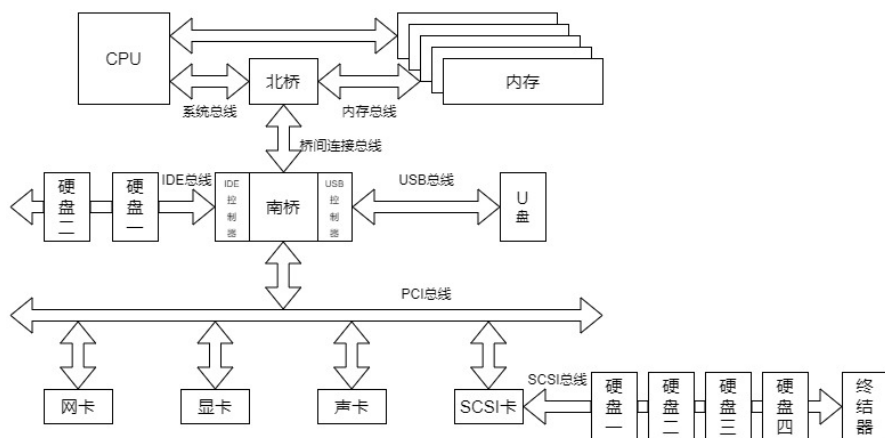


图 5: SCSI 总线概念图

注意，普通台式机主板一般不集成 SCSI 控制器，如果想接入 SCSI 磁盘，则必须增加 SCSI 卡。SCSI 卡一端接入主机的 PCI 总线，另一端用一个 SCSI 控制器接入 SCSI 总线。卡上有自己的 CPU（频率很低，一般为 RISC 架构），通过执行 ROM 中的代码来控制整个 SCSI 卡的工作。经过这样的架构，SCSI 卡将 SCSI 总线上的所有设备经过 PCI 总线传递给内存中运行着的 SCSI 卡的驱动程序，这样操作系统便会知道 SCSI 总线上的所有设备了。如果这块卡有不只一个 SCSI 控制器，则每个控制器都可以单独掌管一条 SCSI 总线，这就是多通道 SCSI 卡。通道越多，一张卡可接入的 SCSI 设备就越多。

注意，在 SCSI 链的最后一个 SCSI 设备要用终结器（中间设备是不需要终结器的，一旦中间设备使用了终结器，那么 SCSI 卡就无法找到以后的 SCSI 设备了）。如果最后一个设备没用终结器，SCSI 是无法正常工作的。终结器是由电阻组成的，位于 SCSI 总线的末端，用来减小相互影响的信号，维持 SCSI 链上的电压恒定。

### 3.1.5 SCSI 的读操作和写操作

下面以 SCSI 读操作和写操作为例介绍 SCSI 的工作流程。

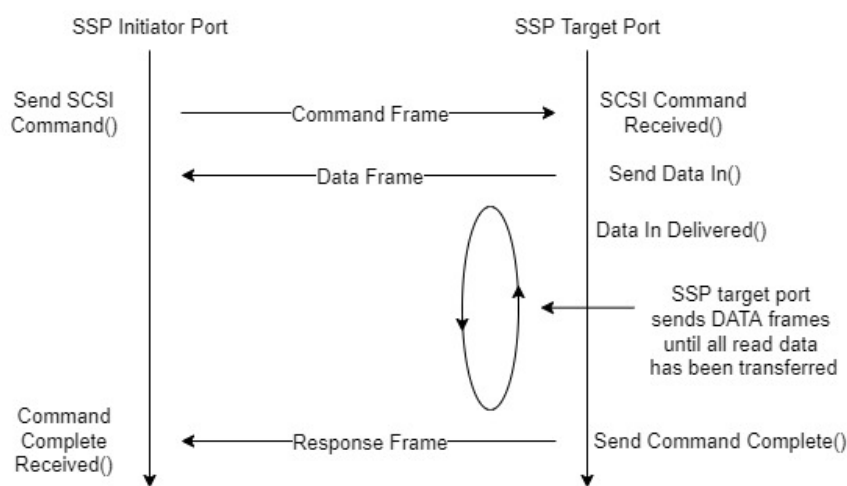


图 6: SCSI 读操作概念图

对于 SCSI 读操作而言，发起方在获得总线仲裁之后，会发送一个 SCSI Command 读命令帧。接收端接收后，立即将该命令中给出的 LUN 以及 LBA 地址段的所有扇区的数据读出，传送给发起端。所有数据传输结束后，目标端发送一个 RESPONSE 帧来表示这条 SCSI 命令执行完毕。SCSI 协议语言就是利用这种两端节点之间相互传送一些控制帧，来达到保障数据成功传输的目的。

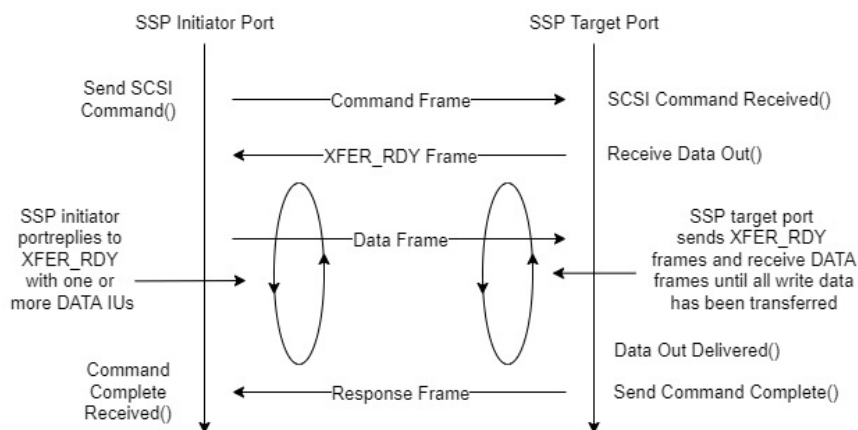


图 7: SCSI 写操作概念图

而对于写操作，发起方在获得总线仲裁之后，会发送一个 SCSI Command 写命令帧，其中包含对应的 LUN 号以及 LBA 地址段。接收端接收后，就知道下一步对方就要传输数据了。接收方做好准备后，向发起方发送一个 XFER\_RDY 帧，表示已经做好接收准备，可以随时发送数据。发起方收到 XFER\_RDY 帧之后，会立即发送数据。每发送一帧数据，接收方就回送一个 XFER\_RDY 帧，表示上一帧成功收到并且无错误，可以立即发送下一帧，直到数据发送结束。接收方发送一个 RESPONSE 帧来表示这条 SCSI 命令执行完毕。

### 3.1.6 SCSI 类型

在发展历程中，SCSI 逐渐发展出了下面的类型：

- SCSI-1：这是最原始的版本，异步传输的频率为 3MB/S，同步传输的频率为 5MB/s。虽然该版本几乎被淘汰了，但还会使用在一些扫描仪和内部 ZIP 驱动器中，采用的是 25 针接口。也就是说，若是将 SCSI-1 设备联接到常见的 SCSI 卡时，必须要有一个内部的 25 针对 50 针的接口电缆；若使用外部设备，就不能采用内部接口中的任何一个（即此时的内部接口均不可以使用）
- SCSI-2：早期的 SCSI-2 也称为 FastSCSI，该版本的 SCSI 通过提高同步传输的频率使据传输速率从原有的 5MB/s 提高为 10MB/s，并支持 8 位并行数据传输，最多可连接 7 个外设。后来出现的 Wide SCSI，则最多支持 16 位并行数据传输，数据传输率也进一步提高到了 20MB/s，最多可连接 16 个外设。此版本的 SCSI 使用一个 50 针的接口，目前主要用于扫描仪、CD-ROM 驱动器及老式硬盘中
- SCSI-3：1995 年，更为高速的 SCSI-3 正式面世，称为 UltraSCSI，数据传输率达到了 20MB/s。若使用 16 位传输的 Wide 模式，数据传输率更可以提高至 40MB/s。此版本的 SCSI 使用一个 68 针的接口，主要应用在硬盘上。SCSI-3 的典型特点是总线频率大大提高，信号的干扰也大大降低，稳定性得到了极大的增强。SCSI-3 有很多型号，比如：
  - Ultra (fast-20) 传输频率 20MHz，数据频宽 8 位，传输率 20MBps
  - Ultra wide 传输频率 20MHz，数据频宽 16 位，传输率 40MBps
  - Ultra 2 传输频率 80MHz，数据频宽 16 位，传输率 80MBps
  - Ultra 160 传输频率 80MHz，数据频宽 16 位，传输率 160MBps
  - Ultra 320 传输频率 80MHz，数据频宽 16 位，传输率 320MBps
  - Ultra 640 的传输频率 160MHz，数据频宽 16 位，传输率 640MBps

SCSI 总线技术拥有诸多优点，可以总结如下：

- SCSI 可支持多个设备，SCSI-2 (FastSCSI) 最多可接 7 个 SCSI 设备，Wide SCSI-2 以上版本的 SCSI 则可接 15 个 SCSI 设备。也就是说，所有的设备

只需占用一个 IRQ，同时 SCSI 还支持相当广的设备，如 CD-ROM、DVD、CDR、硬盘、磁带机、扫描仪等

- SCSI 还允许在对一个设备传输数据的同时，另一个设备对其进行数据查找，这就可以在多任务操作系统（如 Linux 和 WindowsNT 等）中获得更高的性能
- SCSI 占用 CPU 极低，确实多任务系统中占有明显的优势。由于 SCSI 卡本身带有 CPU，可处理一切 SCSI 设备的事务，在工作时主机 CPU 只要向 SCSI 卡发出工作指令，SCSI 卡就会自己进行工作，工作结束后返回工作结果给 CPU，在整个过程中，CPU 均可以进行自身工作
- SCSI 设备还具有智能化，SCSI 卡自己可对 CPU 指令进行排队，这样就提高了工作效率。在多任务时硬盘会在当前磁头位置，将邻近的任务先完成，再逐一进行处理
- 最快的 SCSI 总线有 160MB/s 的带宽，这要求使用一个 64 位的 66MHz 的 PCI 插槽，因此在 PCI-X 总线标准中所能达到的最大速度为 80MB/s，若配合 10000rpm 或 15000rpm 转速的专用硬盘使用将带来明显的性能提升

### 3.1.7 SCSI 接口

下图（来自 Wikipedia）展示了常见的 SCSI 接口及相关信息：


Interface	Alternative names	Specification body / document	Width (bits)	Clock <sup>[a]</sup>	Line code	Maximum		
						Throughput	Length <sup>[b]</sup>	Devices <sup>[c]</sup>
Ultra-320 SCSI	Ultra-4; Fast-160	SPI-5 (INCITS 367-2003)	16	80 MHz DDR	none	320 MB/s (2560 Mbit/s)	12 m	16
SSA	Serial Storage Architecture	T10 / INCITS 309-1997	serial	200 Mbit/s	8b10b	20 MB/s <sup>[d][e][f]</sup> (160 Mbit/s)	25 m	96
SSA 40		T10 / INCITS 309-1997	serial	400 Mbit/s		40 MB/s <sup>[d][e][f]</sup> (320 Mbit/s)	25 m	96
Fibre Channel 1Gbit	1GFC	T11 / X3T11/94-175v0 FC-PH Draft, Revision 4.3	serial	1.0625 Gbit/s	8b10b	98.4 MB/s <sup>[e][f]</sup> (850 Mbit/s)	500 m / 10 km <sup>[g]</sup>	127 (FC-AL) 2 <sup>24</sup> (FC-SW)
Fibre Channel 2Gbit	2GFC	T11 / X3T11/96-402v0 FC-PH-2, Rev 7.4	serial	2.125 Gbit/s		197 MB/s <sup>[e][f]</sup> (1,700 Mbit/s)	500 m / 10 km <sup>[g]</sup>	127/2 <sup>24</sup>
Fibre Channel 4Gbit	4GFC	T11 / INCITS Project 2118-D / Rev 6.10	serial	4.25 Gbit/s		394 MB/s <sup>[e][f]</sup> (3,400 Mbit/s)	500 m / 10 km <sup>[g]</sup>	127/2 <sup>24</sup>
Fibre Channel 8Gbit	8GFC	T11 / INCITS Project 2118-D / Rev 6.10	serial	8.5 Gbit/s		788 MB/s <sup>[e][f]</sup> (6,800 Mbit/s)	500 m / 10 km <sup>[g]</sup>	127/2 <sup>24</sup>
Fibre Channel 16Gbit	16GFC	T11 / INCITS Project 2118-D / Rev 6.10	serial	14.025 Gbit/s	64b66b	1,575 MB/s <sup>[e][f]</sup> (13,600 Mbit/s)	500 m / 10 km <sup>[g]</sup>	127/2 <sup>24</sup>
SAS 1.1	Serial attached SCSI	T10 / INCITS 417-2006 <sup>e</sup>	serial	3 Gbit/s	8b10b	300 MB/s <sup>[e][f]</sup> (2,400 Mbit/s)	6 m	16,256 <sup>[h]</sup>
SAS 2.1		T10 / INCITS 478-2011 <sup>e</sup>	serial	6 Gbit/s		600 MB/s <sup>[e][f]</sup> (4,800 Mbit/s)	6 m	16,256 <sup>[h]</sup>
SAS 3.0		T10 / INCITS 519 <sup>e</sup> 	serial	12 Gbit/s		1,200 MB/s <sup>[e][f]</sup> (9,600 Mbit/s)	6 m	16,256 <sup>[h]</sup>
SAS 4.0		T10 / INCITS 534 <sup>e</sup>  (draft)	serial	22.5 Gbit/s		128b150b	2,400 MB/s <sup>[e][f]</sup> (19,200 Mbit/s)	tbd
IEEE 1394-2008	Firewire S3200, iLink, Serial Bus Protocol (SBP)	IEEE Std. 1394-2008 <sup>e</sup>	serial	3.145728 Gbit/s	8b10b	315 MB/s (2,517 Mbit/s)	4.5 m	63
SCSI Express	SCSI over PCIe (SOP)	T10 / INCITS 489 <sup>e</sup>	serial	8 GT/s (PCIe 3.0)	128b130b	985 MB/s <sup>[e][f][i]</sup> (7,877 Mbit/s)	short, backplane only	2 <sup>58</sup>
USB Attached SCSI 2	UAS-2	T10 / INCITS 520 <sup>e</sup>	serial	10 Gbit/s (USB 3.1)	128b132b	~1,200 MB/s <sup>[e][f]</sup> (~9,500 Mbit/s)	3 m <sup>[j]</sup>	127
ATAPI over Parallel ATA	ATA Packet Interface	T13 / NCITS 317-1998 	16	33 MHz DDR	none	133 MB/s <sup>[k]</sup> (1,064 Mbit/s)	457 mm (18 inches)	2
ATAPI over Serial ATA			serial	6 Gbit/s	8b10b	600 MB/s <sup>[l]</sup> (4,800 Mbit/s)	1 m	1 (15 with port multiplier)
iSCSI	Internet Small Computer System Interface, SCSI over IP	IETF / RFC 7143	mostly serial	implementation- and network-dependent		1,187 MB/s <sup>[m]</sup> or 1,239 MB/s <sup>[n]</sup>	implementation- and network-dependent	2 <sup>128</sup> (IPv6)
SRP	SCSI RDMA Protocol (SCSI over InfiniBand and similar)	T10 / INCITS 365-2002 <sup>e</sup>	implementation- and network-dependent					

图 8: SCSI 接口信息

## 3.2 iSCSI

iSCSI (Internet Small Computer System Interface, Internet 小型计算机系统接口) 是一种基于 Internet 协议和的 SCSI-3 协议的存储网络标准, 用于链接数据存储设施, iSCSI 由 IBM 和 Cisco 于 1998 年首创, 并于 2003 年 2 月 11 日成为正式的标准。iSCSI 也是 IP-SAN 中最重要的、最先获批实施的协议标准, 除了 iSCSI 外, 还有 iFCP、FCIP 等标准。

iSCSI 通过在 TCP/IP 网络上传输 SCSI 命令来提供对存储设备的块级访问。iSCSI 有助于通过 Intranet 传输数据并管理远距离存储。它可用于通过 LAN、WAN 或 Internet 来传输数据, 并且可以实现与位置无关的数据存储和检索。同时, iSCSI 协议允许客户端 (称为 Initiator, 启动器) 向远程服务器上的存储设备 (称为 Target, 目标) 发送 SCSI 命令。

作为一种 SAN 协议, iSCSI 协议允许存储整合到存储阵列中, 同时使客户端 (例如数据库和 Web 服务器) 就像访问本地连接数据存储一样来访问网络存储。

iSCSI 技术有以下技术优势:

- iSCSI 的基础是传统的以太网和 Internet, 同时能大大减少总体拥有成本
- IP 网络的带宽发展相当迅速, 1Gbps 以太网早已大量占据市场, 10Gbps 以太网也已整装待发
- 在技术实施方面, iSCSI 以稳健、有效的 IP 及以太网架构为骨干, 使鲁棒性大大增加
- 简单的管理和布署, 不需要投入培训, 就可以轻松拥有专业的 iSCSI 人才
- iSCSI 是基于 IP 协议的技术标准, 它实现了 SCSI 和 TCP/IP 协议的连接, 只需要不多的投资, 就可以方便、快捷地对信息和数据进行交互式传输及管理
- 完全解决数据远程复制及灾难恢复的难题。安全性方面, iSCSI 已内建支持 IPSEC 的机制, 并且在芯片层面执行有关指令, 确保安全性

尤其是与传统的 SCSI 技术相比, iSCSI 技术有以下的三个革命性的变化:

- 把原来只用于本机的 SCSI 协议透过 TCP/IP 网络发送, 使连接距离可作无限的地域延伸
- 连接的服务器数量无限 (原来的 SCSI-3 的上限是 15)
- 由于是服务器架构, 因此也可以实现在线扩容甚至动态部署

### 3.2.1 iSCSI 优势功能介绍

iSCSI 技术允许两台主机进行协商，然后使用 IP 网络交换 SCSI 命令。这样，iSCSI 就可以采用流行的高性能本地存储总线并在广泛的网络上对其进行仿真，从而实现 SAN 技术。与某些其他 SAN 协议不同，iSCSI 不需要专用电缆，它可以在现有的 IP 基础设施上运行。因此，iSCSI 通常被视为 FCP 版本的 SAN 的低成本替代方案，但是，如果不在专用网络或子网（如 LAN 和 VLAN）上运行，那么 iSCSI SAN 部署的性能可能会因为固定带宽的竞争而严重下降。

尽管 iSCSI 可以与任意类型的 SCSI 设备通信，但它几乎总是被使用来允许服务器（例如数据库服务器）访问存储阵列上的磁盘卷。iSCSI SAN 通常具有以下两个目标之一（这两个目标也是 iSCSI 最具优势的功能）：

- **Storage Consolidation:** iSCSI 可以在数据中心内将分散的存储资源从其网络周围的服务器转移到中央位置，这可以提高存储分配的效率，因为存储本身不再与特定服务器绑定。在 SAN 环境中，可以为服务器分配一个新的磁盘卷，这仅需要对系统软件进行适当的修改，而无需对硬件进行任何更改
- **Disaster Recovery:** iSCSI 可以将存储资源从一个数据中心镜像到远程数据中心，在长时间中断的情况下，该数据中心可以作为备用数据总线，这对于提供鲁棒性的数据总线服务尤其重要。而且，iSCSI SAN 允许通过极少的配置更改来使用 WAN 迁移整个磁盘阵列，通过这种思路，存储变成了“可路由的”，就像普通的网络通信一样

### 3.2.2 iSCSI 工作原理

iSCSI 在物理实现上仍然基于传统的 SCSI 模式，即 Initiator-Target 模式（两者之间的 SCSI 线缆可以为普通电缆或光缆），如下图所示。

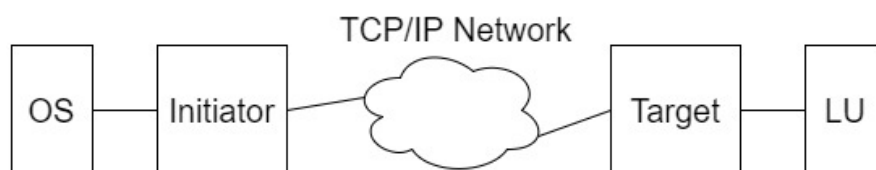


图 9: iSCSI 架构概念图

iSCSI 通过 TCP 面向连接的协议来保护数据块的可靠交付。由于 iSCSI 基于 IP 协议栈，因此可以在标准以太网设备上通过路由或交换机来传输，其协议头的概念图如下所示。





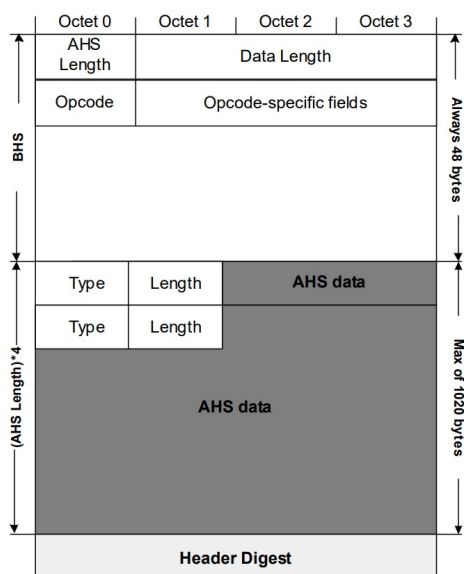


图 13: iSCSI 头

其中每个字段的分析如下：

- **AHS Length**: 以 4 字节为单位给出 AHS Data 部分的长度，且最大允许 1020 字节
- **Data Length**: 以字节为单位给出数据部分的长度，单个 iSCSI PDU 最多允许 16 个字节的数据
- **Opcode**: 在 RFC 7143 中有详细定义，可以总结如下

Opcode ▾	Originator ▾	Operation Name ▾	Reference ▾
0x00	Initiator	NOP-Out	<a href="#">[RFC7143]</a>
0x01	Initiator	SCSI Command	<a href="#">[RFC7143]</a>
0x02	Initiator	SCSI Task Management Function Request	<a href="#">[RFC7143]</a>
0x03	Initiator	Login Request	<a href="#">[RFC7143]</a>
0x04	Initiator	Text Request	<a href="#">[RFC7143]</a>
0x05	Initiator	SCSI Data-Out	<a href="#">[RFC7143]</a>
0x06	Initiator	Logout Request	<a href="#">[RFC7143]</a>
0x10	Initiator	SNACK Request	<a href="#">[RFC7143]</a>
0x1c-0x1e	Initiator	Vendor-specific codes	<a href="#">[RFC7143]</a>
0x20	Target	NOP-In	<a href="#">[RFC7143]</a>
0x21	Target	SCSI Response	<a href="#">[RFC7143]</a>
0x22	Target	SCSI Task Management Function Response	<a href="#">[RFC7143]</a>
0x23	Target	Login Response	<a href="#">[RFC7143]</a>
0x24	Target	Text Response	<a href="#">[RFC7143]</a>
0x25	Target	SCSI Data-In	<a href="#">[RFC7143]</a>
0x26	Target	Logout Response	<a href="#">[RFC7143]</a>
0x31	Target	Ready To Transfer (R2T)	<a href="#">[RFC7143]</a>
0x32	Target	Asynchronous Message	<a href="#">[RFC7143]</a>
0x3c-0x3e	Target	Vendor-specific codes	<a href="#">[RFC7143]</a>
0x3f	Target	Reject	<a href="#">[RFC7143]</a>

图 14: iSCSI Opcode 字段

- **Opcode-Specific Fields:** 这个 3 字节字段根据 Opcode 的类型具有不同的含义。例如，iSCSI 登录操作码 (Opcode=0x03) 使用此区域来存储 iSCSI 版本号和登录控制标志
- **Type (AHS Type):** 是 AHS 部分的第一个字段，该字段标识了如何对 AHS Data 字段进行解析，ANS Type 字段可以进一步分解为：
  - bit 0 (MSB): 含义是 Drop Bit，如果该位被设置，那么如果接收方不能理解 AHS Type 时将把其对应的 PDU 丢弃
  - bit 1: 被发送方设置为 0，被接收方忽略
  - bit 2-7: 真正的 AHS Type 字段，该字段中包含的值指示如何解释 AHS Data 字段中的位。该字段可以取 0 到 63 之间的任何值。0 到 62 之间的值保留给 iSCSI 分配。而值 63 表示 AHS Data 字段包含 iSCSI 未定义的信息
- **AHS Length:** 是 AHS 部分的第二个字段，该字段以 4 字节为单位给出了分配给完整 AHS 的单元数量（包括 AHS Type 和 AHS Length 字段本身的单元）。最小 AHS 的 AHS Length 为 1，最大值为 255 个字或 1020 个字节
- **AHS Data:** 是 AHS 部分的第三个字段，长度在 2 到 1018 字节之间，包含的信息取决于 AHS Type 字段中包含的值，比如，当 AHS Type 字段为 0000101 时，所代表的含义为 FCP\_CRN，这时 AHS Length 为 1，AHS Data 字段的内容为 0x008c

### 3.2.3 iSCSI 的实际实现

在 iSCSI 配置中，iSCSI 主机或服务器将请求发送到节点。主机包含一个或多个连接到 IP 网络的启动器，以发出请求，并接收来自 iSCSI 目标的响应。

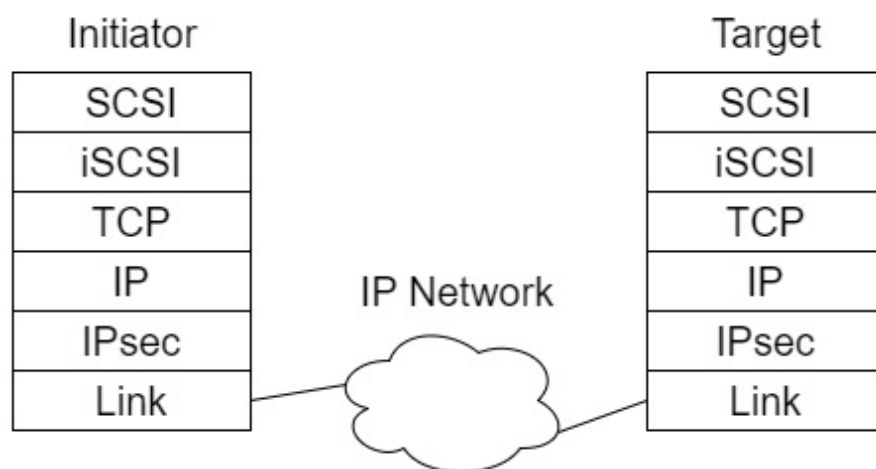


图 15: iSCSI 实现过程概念图

每个启动器和目标都指定了一个唯一的 iSCSI 名称，如 iSCSI 限定名 (IQN) 或扩展的唯一标识 (EUI)。IQN 是 223 字节的 ASCII 名称。EUI 是 64 位标识。iSCSI 名称代表全球唯一命名方案，该方案用于标识各启动器或目标，其方式与使用全球节点名 (WWNN) 来标识光纤通道光纤网中设备的方式相同。

iSCSI Target 是响应 iSCSI 命令的设备。iSCSI 设备可以是诸如存储设备的结束节点、或者可以是诸如 IP 与光纤通道设备之间的网桥的中间设备。每个 iSCSI Target 由唯一的 iSCSI 名称标识。如果要通过 IP 网络传输 SCSI 命令，iSCSI 驱动程序必须安装到 iSCSI Initiator 和 Target 中。驱动程序用于通过主机或目标硬件中的网络接口控制器 (NIC) 或 iSCSI HBA 来发送 iSCSI 命令和响应。注意，为实现最佳性能，需要使用传输速度为 1000 Mbps 的千兆以太网适配器在 iSCSI Initiator 和 iSCSI Target 间进行连接。

iSCSI 会话建立在一个 Initiator 与一个 Target 之间，一个会话允许多个 TCP 连接，并且支持跨连接的错误恢复。大多数通信还是建立在 SCSI 基础之上的，例如，使用 R2T 进行流量控制。iSCSI 添加于 SCSI 之上的有：立即和主动的数据传输以避免往返通信；连接建立阶段添加登录环节，这是基于文本的参数协商。建立一个 iSCSI 会话，包括如下阶段：

- 命名阶段：确定需要访问的存储，以及 Initiator，与 FC 不同，命名与位置无关
- 发现阶段：找到需要访问的存储
- 登录阶段：建立于存储的连接，读写之前首先进行参数协商，按照 TCP 连接登录

建立了 iSCSI 会话之后，便可以开始使用 iSCSI 服务。

## 3.3 FC 与 FCP

### 3.3.1 FC 与 FCP 概要

Fibre Channel (FC) 是一种高速数据传输协议，可提供有序、无损的原始块数据传输，主要用于将计算机数据存储连接到商业数据中心的存储区域网络 (SAN) 中的服务器。

FC 形成了一种交换结构，因为网络中的交换机作为一个大交换机统一运行。FC 通常在数据中心内部和之间的光纤电缆上运行，但也可以在铜缆上运行。支持的数据速率包括 1、2、4、8、16、32、64 和 128 Gbps，这是由于连续几代技术的改进而产生的。

FC 有多种上层协议，包括两种用于块存储的协议：Fibre Channel Protocol (FCP) 是一种通过光纤通道网络传输 SCSI 命令的协议；Fibre Connection (FICON) 是一种通过光纤通道传输 IBM 大型计算机使用的 ESCON 命令的协议。

下面来重点介绍 FCP 的相关内容。

Fibre Channel Protocol (FCP) 是利用底层光纤通道连接的 SCSI 接口协议。光纤通道标准定义了一种高速数据传输机制，可用于连接工作站、大型机、超级计算机、存储设备和显示器。FCP 解决了对大量信息的快速传输的需求，并且可以减轻系统制造商支持各种渠道和网络的负担，因为它为网络、存储和数据传输提供了一种标准。光纤通道的一些特性包括：

- 性能从 266 Mbps 秒到 16 Gbps
- 支持光纤和铜线介质，传输距离最长可达 10 公里
- 小型连接器（最常见的是 sfp+）
- 对距离不敏感的高带宽利用率
- 支持从小型系统到超级计算机的多种成本/性能级别
- 能够承载多个现有接口命令集，包括 Internet 协议 (IP)、SCSI、IPI、HIPPI-FP 和音频/视频等协议

自从 1988 年出现以来，FCP 已经发展成为一项非常复杂、高速的网络技术。它最初并不是研究来作为一种存储网络技术的。最早版本的 FCP 是一种为了包括 IP 数据网在内的多种目的而推出的高速骨干网技术，它是作为惠普、Sun 和 IBM 等公司组成的 R&D 实验室中的一项研究项目开始出现的，当时的 FCP 开发者认为这项技术有一天会取代 100BaseT 以太网和 FDDI 网络，他们将 FCP 作为一种高速骨干网络技术，而将存储作为不重要的应用。

### 3.3.2 光纤通道发展历史

光纤通道在国际信息技术标准委员会 (INCITS) 的 T11 技术委员会中进行了标准化，该委员会是 ANSI 认可的标准委员会。光纤通道始于 1988 年，并于 1994 年获得 ANSI 标准批准，主要优势是合并多个物理层的实施，包括 SCSI、HIPPI 和 ESCON。

光纤通道被设计为串行接口，从而克服 SCSI 和 HIPPI 物理层并行信号铜线接口的限制。此类接口面临的挑战之一是保持所有数据信号线（SCSI 为 8、16 和 32 条，HIPPI 为 50 条）之间的信号时序一致性，以便接收器可以确定所有电信号值何时稳定有效。随着数据信号频率的增加，这一挑战在大规模制造的技术中变得越来越困难，部分技术补偿是不断减少支持的连接铜平行电缆长度。光纤通道采用领先的多模光纤开发克服了 ESCON 协议的速度限制的技术。通过使用大量 SCSI 磁盘驱动器并利用大型机技术，光纤通道实现了规模经济效应，并且开始广泛而经济地进行部署。

当标准被批准时，低速版本的光纤通道已经不再使用。光纤通道是第一个达到千兆位速度的串行存储传输，并且自 1996 年以来，光纤通道的速度每隔几年

Name	Line-rate(gigabaud)	Line-coding	MBps	Availability
1GFC	1.0625	8b10b	100	1997
2GFC	2.125	8b10b	200	2001
4GFC	4.25	8b10b	400	2004
8GFC	8.5	8b10b	800	2005
10GFC	10.51875	64b66b	1200	2008
16GFC	14.025	64b66b	1600	2011
32GFC	28.05	256b257b	3200	2016
64GFC	28.9	256b257b	6400	2019
128GFC	28.05×4	256b257b	12800	2016
128GFC	57.8	256b257b	12800	2022
133 Mbps	0.1328125	8b10b	12.5	1993
256GFC	28.9×4	256b257b	25600	2019
266 Mbps	0.265625	8b10b	25	1994
533 Mbps	0.53125	8b10b	50	?

就会翻一番。光纤通道从一开始就得到了积极的发展，在各种底层传输介质上都有了许多速度改进。下表显示了本机光纤通道速度的进展：

除了现代物理层之外，光纤通道还增加了对任意数量的“上层”协议的支持，包括 ATM、IP (IPFC) 和 FICON，其中 SCSI (FCP) 是主要用途。需要注意的是，光纤通道并不遵循 OSI 模型分层，而是常分为五层：

- FC-4 (协议映射层)：可以通过光纤通道执行的应用程序接口，将 NVMe、SCSI、IP、FICON 等上层协议封装到信息单元中并交付给 FC-2。当前的 FC-4 包括 FCP-4、FC-SB-5 和 FC-NVMe
- FC-3 (公共服务层)：一个最终可以实现加密或 RAID 冗余算法等功能的薄层，可以实现条带化、寻线组和多播等高级功能所需的通用服务，遵循多端口连接原则
- FC-2 (信令协议层)：由光纤通道成帧和 FC-FS-5 标准定义，由低级光纤通道网络协议组成，完成帧、序列和交换的传输（包括协议信息单元的传输），遵循端口到端口连接原则
- FC-1 (传输协议层)：实现信号的线路编码，即对于物理媒体上传输的数据和带外物理链路控制信息的编码和解码
- FC-0 (物理层)，包括电缆、连接器等设备

### 3.3.3 FC 网络拓扑

FC Arbitrated Loop (FC AL) 是一种光纤通道拓扑，其中设备以环形拓扑中的单向环路方式连接。它允许连接许多服务器和计算机存储设备，而无需使用较为昂贵的光纤通道交换机。现在，随着光纤通道交换机的成本不断下降，FC AL 的应用范围不断缩小，但目前在存储系统中仍然很常见。

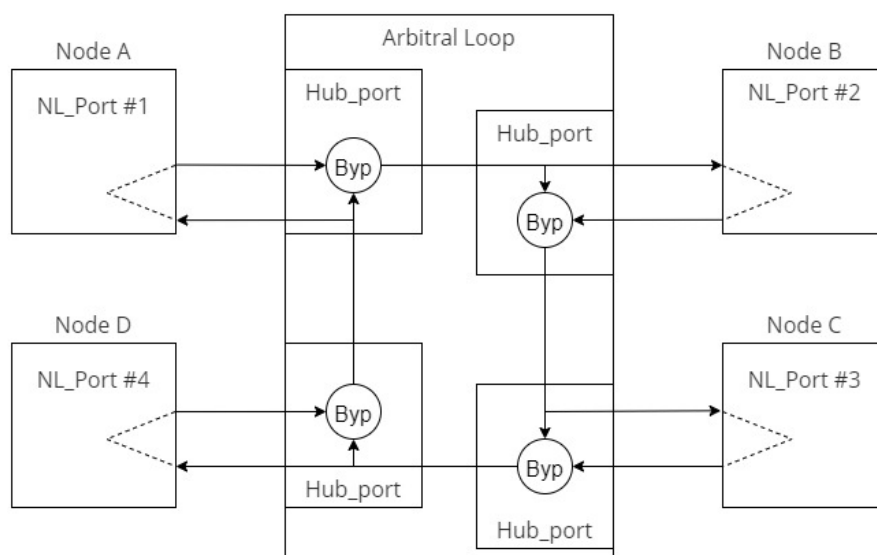


图 16: 仲裁环概念图

FC AL 拓扑类似于以太网共享总线拓扑，但是连接方式不是总线，而是一条仲裁环路。每个 FC AL 设备首尾相接构成了一个环路。一个环路能接入的最多节点是 128 个，实际上是用了一个字节的寻址容量，但是只用到了这个字节经过 810b 编码之后奇偶平衡的值，也就是 256 个值中的 134 个值来寻址，这些被筛选出来的地址中又被广播地址、专用地址等占用了，最后只剩下 127 个实际可用的节点地址。

FC AL 可以以环形方式或使用集线器进行物理连接。如果链中的一个设备发生故障，物理环将停止工作。另一方面，集线器在保持逻辑环的同时，允许在电缆级别上采用星形拓扑。集线器上的每个接收端口都被简单地传递到下一个活动传输端口，而绕过任何非活动或故障端口。

因此，光纤通道集线器具有另一项功能：提供旁路电路，以防止在一个设备发生故障或被移除时环路中断。如果设备从环路中移除，集线器的旁路电路会检测到信号缺失并立即开始将传入数据直接路由到环路的下一个端口，从而完全绕过丢失的设备。这至少为循环提供了一定程度的鲁棒性，即循环中的一个设备发生故障不会导致整个循环无法运行。

综上所述，FC AL 具有如下的特点：

- 是一种串行架构，可用作 SCSI 网络中的传输层，最多可支持 127 个设备，环路可以通过其端口之一连接到光纤通道结构

- 环路上的带宽在所有端口之间共享
- 在环路上一次只能通信两个端口，一个端口赢得仲裁，并且可以在半双工或全双工模式下打开另一个端口
- 能够仲裁环路通信的光纤通道端口有节点环路端口（NL\_port）和结构环路端口（FL\_port），统称为 L\_ports。端口可以通过集线器相互连接，电缆从集线器延伸到端口，但集线器上的物理连接器不是协议的端口，集线器不包含端口
- 没有结构端口（而只有 NL\_ports）的仲裁环路是私有环路
- 通过 FL\_port 连接到结构的仲裁环路是公共环路
- NL\_Port 必须提供结构登录（FLOGI）和名称注册设施，以通过结构启动与其他节点的通信而成为发起者（Initiator）

### 3.3.4 FC 帧结构

FC 网络中数据传输的基本单位是 FC 帧，在 FC-2 中对帧的格式给出了统一的规定：一个 FC 帧是由 SOF、帧内容以及 EOF 3 部分组成，而帧内容又可以分为帧头、数据字段及 CRC 共三个部分。FC 帧格式如下图所示。

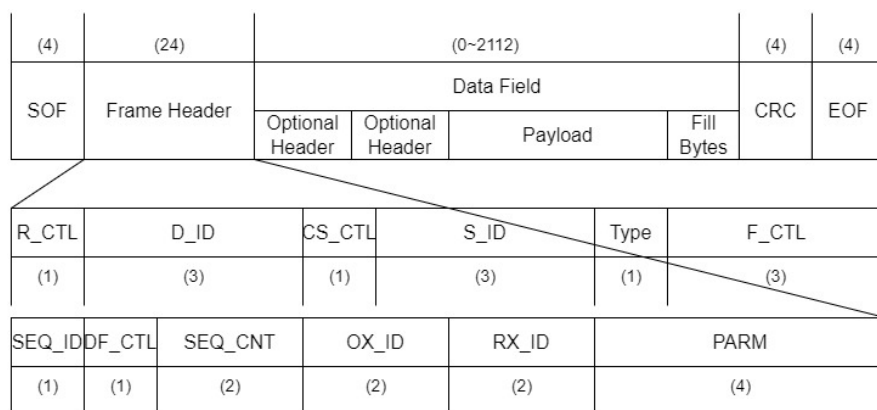


图 17: FC 协议头字段

并且，各个字段的含义如下：

- SOF 和 EOF：这 2 个有序集用于标识帧的开始和结束，且 SOF 和 EOF 在不同的帧、不同的使用环境中的具体数值是不一样的
- 帧头：在帧格式中，帧头是一个 24 B 的字段，按照字边界进行传输。用于控制链路操作、设备协议的传输以及帧丢失或乱序检测，帧头中每个字段的含义如下：
  - R\_CTL：说明帧的类型，1 字节



- D\_ID: 帧的目标地址, 3 字节
  - CS\_CTL: 帧的分类控制信息, 1 字节
  - S\_ID: 帧的源地址, 3 字节
  - Type: 帧内的协议类型, 1 字节
  - F\_CTL: 帧控制字段, 3 字节
  - SEQ\_ID、DF\_CTL、SEQ\_CNT: 用于帧的顺序控制, 共 4 字节
  - OX\_ID: 发起者交换机的 ID, 2 字节
  - RX\_ID: 响应者交换机的 ID, 2 字节
  - PARM: 参数字段, 4 字节
- CRC 字段: 该字段包含 4 B 的 CRC 校验码, 用于验证 FC 帧中的帧头和数据字段的完整性, 但是 SOF 和 EOF 没有包含在 CRC 校验中

### 3.3.5 FC 流量控制

在链路层上, FC 定义了两种流控策略: 一种为端到端的流控, 另一种为缓存到缓存的流控。端到端流控比缓存到缓存流控要上层和高级。在一条链路的两端, 首先面对链路的一个部件就是缓存。接收电路将一帧成功接收后, 就放入了缓存中。如果由于上位程序处理缓慢而造成缓存已经充满, FC 协议还有机制来通知发送方减缓发送。如果链路的一端是 FC 终端设备, 另一端是 FC 交换机, 则二者之间的缓存到缓存的流量控制只能控制这个 FC 终端到 FC 交换机之间的流量。而通信的最终目标是网络上的另一个 FC 终端, 这之间可能经历了多个 FC 交换机和多条链路。而如果数据流在另外一个 FC 终端之上发生拥塞, 则这个 FC 终端就必须通知发起端降低发送频率, 这就是“端到端”的流量控制。

### 3.3.6 FC Fabric 介绍

Fabric 网络架构是近些年十分流行的数据中心网络架构, 其中最具有代表性的就是 Facebook 在 2014 年公开的其数据中心架构。Fabric 网络架构主要包含客户端节点、CA 节点、Peer 节点、Orderer 节点等部分, 其中, 各个节点功能简述如下:

- CA 节点功能: Fabric 网络中的成员提供基于数字证书的身份信息, 可选, 可以用第三方生成的证书
- 客户端节点的功能: 与 Fabric 区块链交互, 实现对区块链的操作。常见 cli 容器及 SDK 客户端。客户端代表最终用户的实体。它必须连接到 peer 与区块链进行通信。客户端可以连接到其选择的任何 peer。客户端创建交易
- Peer 节点功能: Fabric 每个组织都包含一个或者多个 peer 节点, 每个节点可以担任多种角色, 常见的有:

- Endorser Peer (背书节点) 功能: 对客户端发送交易提案时进行签名背书, 充当背书节点的角色。背书 (Endorsement) 是指特定 Peer 节点执行交易并向生成交易提案 (proposal) 的客户端应用程序返回 YES/NO 响应的过程。背书节点是动态的角色在链码实例化的时设置背书策略 (Endorsement policy), 指定哪些节点对交易背书才有效。只有在背书时是背书节点, 其他时刻是普通节点
- Leader Peer (主节点) 功能: 主要负责与 orderer 排序节点通信, 获取区块及在本组织进行同步。主节点的产生可以动态选举或者指定
- Committer Peer (记账节点) 功能: 对区块及区块交易进行验证, 验证通过后将区块写入账本中
- Anchor Peer (锚节点) 功能: 主要负责与其他组织的锚节点进行通信
- Orderer 节点功能: 排序服务节点接收包含背书签名的交易, 对未打包的交易进行排序生成区块, 广播给 Peer 节点。排序服务提供的是原子广播, 保证同一个链上的节点为接收到相同的消息, 并且有相同的逻辑顺序

并且, 在 Fabric 中, 引入了通道的概念。一般情况下, 一条区块链网路的子链是按照“1 个通道 + 1 个账本 + N 个成员”的基本组成。通道是两个或多个特定网络成员之间的通信的私有“子网”, 用于进行需要数据保密的交易。在 Fabric 中, 建立一个通道相当于建立了一个子链。创建通道是为了限制信息传播的范围, 是和某一个账本关联的。每个交易都是和唯一的通道关联的。这会明确地定义哪些实体 (组织及其成员) 会关注这个交易。

### 3.3.7 FC Fabric 寻址

鉴于上述 Fabric 网络架构的特点, Fabric 网络架构中的寻址就是一个特别需要注意的问题, 现在讨论该问题如下。

和以太网端口 MAC 地址类似, FC 网络中的每个设备自身都有一个 WWNN (World Wide Node Name), 不管这个设备上有多少个 FC 端口, 设备始终拥有一个固定的 WWNN 来代表它自身。然后, FC 设备的每个端口都有一个 WWPN (World Wide Port Name) 地址, 也就是说这个地址在世界范围内是唯一的, 世界上没有两个接口地址是相同的。

FC Fabric 拓扑在寻址和编址方面与以太网又有所不同。具体体现在以太网交换设备上的端口不需要有 MAC 地址, 而 FC 交换机上的端口都有自己的 WWPN 地址。这是因为 FC 交换机要做的工作比以太网交换机多, 许多 FC 逻辑都被集成在 FC 交换机上, 而以太网的逻辑相对就简单了许多, 因为上层逻辑都被交给诸如 TCP/IP 这样的上层协议实现了。然而 FC 的 Fabric 网中, FC 交换机担当了很重要的角色, 它需要处理到 FC 协议的最上层。每个 FC 终端设备除了和最终通信的目标有交互之外, 还需要和 FC 交换机打好交道。

WWNN 每个 FC 设备都被赋予一个 WWNN, 这个 WWNN 一般被写入设备的 ROM 中不能改变, 但是在某些条件下也可以通过运行在设备上的程序动态的

改变。注意，WWPN 和三个 IDWWPN 地址的长度是 64 位，比以太网的 MAC 地址还要长出 16 位。然而，如果 8B 长度的地址用于高效路由的话，无疑效率太低。所以 FC 协议决定在 WWPN 之上再映射一层寻址机制，就是像 MAC 和 IP 的映射一样，给每个连接到 FC 网络中的接口分配一个 Fabric ID，用这个 ID 而不是 WWPN 来嵌入链路帧中做路由。这个 ID 长 24 位，高 8 位被定义成 Domain 区分符、中 8 位被定义为 Area 区分符、低 8 位定义为 PORT 区分符。

也就是说，WWPN 被映射到 Fabric ID，一个 Fabric ID 所有 24b 又被分成 Domain ID、Area ID、Port ID 这三个亚寻址单元，它们各自的含义如下：

- **Domain ID**：用来区分一个由众多交换机组成的大的 FC 网络中每个 FC 交换机本身。一个交换机上所有接口的 Fabric ID 都具有相同的高 8 位，即 Domain ID。Domain ID 同时也用来区分这个交换机本身，一个 Fabric 中的所有交换机拥有不同的 Domain ID。一个多交换机组成的 Fabric 中，Domain ID 是自动被主交换机分配给各个交换机的。根据 WWNN 号和一系列的选举帧的传送，WWNN 最小者获胜成为主交换机，然后这个主交换机向所有其他交换机分配 Domain ID，这个过程其实就是一系列的特殊帧的传送、解析和判断
- **Area ID**：用来区分同一台交换机上的不同端口组，比如 1、2、3、4 端口属于 Area 1，5、6、7、8 端口属于 Area 2 等。其实 Area ID 这一层亚寻址单元意义不是很大。我们知道，每个 FC 接口都会对应一块用来管理它的芯片，然而每个这样的芯片却可以管理多个 FC 端口。所以如果一片芯片可以管理 1、2、3、4 号 FC 端口，那么这个芯片就可以属于一个 Area，这也是 Area 的物理解释。同样，在主机端的 FC 适配卡上，一般也都是用一块芯片来管理多个 FC 接口的
- **Port ID**：用来区分一个同 Area 中的不同 Port

经过这样的三段式寻址体系，我们可以区分一个大 Fabric 中的每个交换机、交换机中的每个端口组及每个端口组中的端口，这就是 FC Fabric 的寻址过程。

### 3.3.8 FC Fabric 登录

下面以 FC 设备登录到 Fabric 网络为例介绍 FC Fabric 的应用过程。

首先，发起者向地址为 0xFFFFFE 的注册服务器发送 FLOGI 请求，注册服务器向发起者发送回端口地址。发送者获取许可并接收到端口地址之后，向名称服务器发送端口注册请求，是否允许注册则由名称服务器决定，如果发送者得到注册许可，那么将同时收到由名称服务器发来的可访问设备列表。

在这一过程中，有两套编址体系（Fabric 和 WWPN）起作用，那么就必须有地址映射法则来处理这个问题，FC 协议中地址映射步骤如下：

- 当一个接口连接到 FC 网络中时，如果是 Fabric 架构，那么这个接口会发起

一个登录注册到 Fabric 网络的动作，也就是向目的 Fabric ID 地址 FFFFFFFE 发送一个登录帧，称为 FLOGIN

- 交换机收到地址为 FFFFFFFE 的帧之后，会动态地给这个接口分配一个 24b 的 Fabric ID，并记录这个接口对应的 WWPN，做好映射
- 此后这个接口发出的帧中不会携带其 WWPN，而是携带其被分配的 ID 作为源地址

这样，就完成了 FC 设备登录到 Fabric 网络的过程。

## 3.4 FC SAN

### 3.4.1 FC SAN 概要

FC SAN 是由网线、网络适配器和集线器或交换机等组件组成的 SAN，其中不同类型的 FC 架构有：

- 点到点架构 (Point-to-point)：在该架构中，两个节点直接相互连接。此配置为节点之间的数据传输提供专用连接。但是，点对点架构限制了连接性和可扩展性
- 光纤通道仲裁环路架构 (Fibre channel arbitrated loop)：在该架构中，设备连接到共享环路。每个设备都与其他设备竞争执行 I/O 操作。环路上的设备必须通过仲裁机制获得对环路的控制，在任何给定时间，只有一个设备可以在环路上执行 I/O 操作。由于每个设备必须等待轮到它处理 I/O 请求，所以 FC AL 环境中的整体性能较低。此外，添加或移除设备会导致环路重新初始化，这可能会导致环路流量暂时暂停。作为环路架构，FC AL 可以在没有任何互连设备的情况下通过电缆将一个设备直接连接到环中的另外两个设备来实现。FC AL 实现也可以使用 FC 集线器，仲裁环路通过该集线器以星形拓扑物理连接
- 光纤通道交换架构 (Fibre channel switched fabric)：该架构通常涉及单个 FC 交换机或 FC 交换机网络来互连节点，所有节点在同一个逻辑空间相互通信。在该架构下，任意两台交换机之间的链路称为交换机间链路，这种链路使交换机能够连接在一起以形成单个更大的结构

在此之前，我们已经对 FC SAN 中的部分概念进行了十分细致的讨论，此处不再重复。

### 3.4.2 FC SAN 与 IP SAN 的比较

下表列出了 FC-SAN 技术和 IP-SAN 技术的比较：

	IP SAN	FC SAN	说明
网络速度	1Gb、10Gb	1Gb、2Gb、4Gb、8Gb	
网络架构	使用现有 IP 网络	单独建设光纤网络和 HBA 卡	SAN 技术中的 iSCSI 协议可以使用现有 IP 网络，但这会导致 SAN 性能受限
传输距离	理论上没有距离限制	受到光纤传输距离的限制	
技术成熟度	IP SAN 是在最近 2 年开始为业务所推行，并为用户所认识的新技术	FC SAN 是从九十年代末即开始发展的存储网络技术，其发展已经历至少三代：1Gb、2Gb 以及目前的 4Gb 乃至 8Gb。其应用已将近十年，是非常成熟和可靠的技术，其采用和认可的广泛程度可以说是遍布几乎各类机构，大中小型企业的数据中心里	在技术成熟度上，FC SAN 比 IP SAN 要高得多
协议效率	IP SAN 实质上就是将 SCSI 指令封装在 IP 包中，利用 IP 技术进行包的传输，利用的是 IP 技术的广泛性和普及性。但是 IP 封装有一个显著的弱点：就是 IP 封装的开销大，效率低——即任何一个 IP 包中要附加的包头和包尾，以及检验码过多因此其总体的效率不高	比较 FC SAN，将 SCSI 指令在 FC 包中进行封装，包头和包尾以及校验码所占比例非常低，因此其效率非常高	从效率上看，FC SAN 明显高于 IP SAN，因此 FC SAN 更加适合于对效率敏感的应用，例如对性能要求很高的数据库应用，而 IP SAN 则主要应用到对性能和效率要求不高的环境中，例如 OA，文档处理，多媒体环境等
性能	IP SAN 协议中的封装效率不高，因此 IP SAN 对环境的硬件速度要求更好，才能获得与 FC 差不多的性能，可惜的是目前 IP SAN 最好环境还只是在千兆网中，因此其性能目前还无法与 FC SAN 相比	FC SAN 协议本身效率高，同时目前 FC SAN 已经开始普遍部署 4Gbps 的环境，所以说 FC SAN 要比目前 IP SAN 性能块很多	乐观地讲，10Gb 即万兆网中 IP SAN 的性能可能会有显著改善，并能初步满足相关应用的要求，与目前 FC SAN 中的性能可以一比

	IP SAN	FC SAN	说明
<b>稳定性和安全性</b>	较低，容易丢包、截取	较高	IP SAN 是建立在普通 IP 网上，FC SAN 是建立在 FC 网络中。FC 网络的抗干扰性要强；同时 FC 网络的封闭性要高一些，不想 IP 网络非常开放，因此 FC SAN 协议上要相对安全和稳定
<b>成本</b>	与 FC SAN 相比，购买与维护成本都较低，有更高的投资收益比例	购买（光纤交换机、HBA 卡、光纤磁盘阵列等）、维护（培训人员、系统设置与监测等）成本高	IP SAN 的成本低主要体现在：设备价位低和运行维护费用低
<b>容灾能力</b>	本身可以实现本地和异地容灾，且成本低	容灾的硬件、软件成本高	
<b>兼容性</b>	目前 IP SAN 主要完成的是 Windows，Linux 等较低端的服务器的兼容性测试；厂商支持度：服务器方面，主要是 PC Server 厂商和低端 Unix 服务器明确支持，部分高端服务器还不支持兼容性；存储方面：虽然大多数存储都能支持 IP SAN，但是在用户环境中应用的主要还是中低端存储	FC SAN 兼容性测试已非常充分，遍布所有高端、中端、以及低端的服务器均能支持，厂商支持度：不管服务器还是存储方面，几乎所有的服务器（不论档次）和独立存储系统都完全支持 FC	在兼容性和厂商支持度上看，同样建议对于重要和关键业务系统，目前还是要采用 FC 要更稳定和可靠

从上表可以看出，在非关键环境中、或在建设成本非常有限的条件下，可以考虑采用 IP SAN；而对于企业或者机构核心业务、关键业务中，其稳定性、性能、对技术的成熟度要求高，应当采用 FC SAN，这样企业和机构的主要业务系统将更有保障。

## 4 结语

SAN 技术是一种连接网络从而提供数据存储功能的技术，其主要特点包括：块级存储、以数据为中心、将存储设备与服务器分离、集中管理数据等。SAN 技术有效地提高了数据服务的速度、提高了性能、增加了系统的鲁棒性，这使其实

际性能表现远高于使用其他的一些存储方式。

在本文中，我简要梳理了 SAN 技术的一些常见内容，感觉收获颇丰，同时又惊叹于计算机工业界实践成果之富，这值得我们持续不断地学习与探索！