雪花结构:一种新型数据中心网络结构

刘晓茜[1] 杨寿保[1] 郭良敏[1),2)] 王淑玲[1] 宋 浒[1]

[1)](中国科学技术大学计算机科学与技术学院 合肥 230026)

[2)](安徽师范大学计算机科学与技术系 安徽 芜湖 241003)

摘 要 该文分析了传统数据中心的不足以及新型数据中心具备的新特点,借鉴已有数据中心结构,依据著名的科赫曲线(Koch curve),提出了新型数据中心网络结构——雪花结构.该结构充分考虑了数据中心的可扩展性,在保证交换机与服务器较低数量比例(0.125~0.333)的前提下,可以在较短的平均路径内实现节点间路由机制,具有较小的网络开销.

关键词 数据中心;网络结构;协议;路由

中图法分类号 TP393 **DOI 号**: 10.3724/SP.J.1016.2011.00076

Snowflake: A New-Type Network Structure of Data Center

LIU Xiao-Qian[1)] YANG Shou-Bao[1)] GUO Liang-Min[1),2)] WANG Shu-Ling[1)] SONG Hu[1)]

[1)](*School of Computer Science and Technology, University of Science and Technology of China, Hefei* 230026)

[2)](*Department of Computer Science and Technology, Anhui Normal University, Wuhu, Anhui* 241003)

Abstract The paper analyses the deficiency of traditional data center and new characteristics of current data center. Basing on the famous Koch curve, it proposes a new-type network structure of data center, called Snowflake. It takes scalability into full consideration. On the premise of low proportion of switches and servers, it can efficiently achieve routing mechanisms in shorter mean path lengths with less network cost.

Keywords data center; network structure; protocol; routing

1 引 言

过去两年里随着碳排放和电费的不断激增,使得高能效数据中心的研究已经发展成为热点.数据中心是提高能源效率最困难的商业建筑,因为安装在数据中心的计算机需要大量的电力并且排放出大量的热量,而数据中心消耗的一半电力都用于支持IT设备的电力和制冷空调等冷却基础设施.

美国联邦环境保护署(Environmental Protection Agency,EPA)于2009年8月提交了一份关于数据中心的报告.这份报告称,数据中心的能源消耗从2000年~2006年增长了一倍,预计到2011年再增长一倍.此外,全球权威机构Gartner调查也显示,IT行业每年的二氧化碳排放量约为3500万吨,占全球总排放量的2%,数据中心成碳排放大户.企业每年在用电成本上的花费已经大于当年硬件设备投资额.面对如此严峻的形势,节能已经成为建设绿色数据中心的重点.从机房用电分配看,服务器设备占电能总能耗的52%,而制冷系统和电源系统各占

收稿日期:2010-04-22;最终修改稿收到日期:2010-09-02.本课题得到国家自然科学基金(60673172)、国家"八六三"高技术研究发展计划项目基金(2006AA01A110)、中国科学技术大学研究生创新基金(KD0901110)资助.**刘晓茜**,女,1983年生,博士研究生,研究方向为云计算、网格计算.E-mail:xql@mail.ustc.edu.cn.**杨寿保**,男,1947年生,教授,博士生导师,研究领域为云计算、网格计算、无线网络.**郭良敏**,女,1980年生,博士研究生,研究方向为云计算、对等计算.**王淑玲**,女,1988年生,博士研究生,研究方向为分布式搜索、云存储.**宋 浒**,男,1986年生,博士研究生,研究方向为云计算、高性能计算.

38%和 9%，照明系统仅占 1%. 因此从服务器的角度出发，在保证数据中心海量服务器的前提下尽可能降低交换机个数，提升服务器与交换机的数量比例，不仅可以有效降低能耗达到减排目的，同时还可以降低交换机的成本开销.

本文借鉴已有的数据中心网络结构，依据著名科赫曲线[1]，提出新型数据中心结构——雪花结构(Snowflake，Snow). 该结构充分考虑了数据中心的可扩展性，在保证交换机与服务器较低数量比例的前提下，可以在较短时间内实现节点间路由机制，具有较小的网络开销.

本文第 2 节介绍相关工作；第 3 节是全文主体，详细介绍基于雪花结构的数据中心网络构建方法及其属性内容；第 4 节阐述该结构中节点间的路由协议和算法；第 5 节补充说明其它方面的内容，例如雪花结构不完全时的情况等；第 6 节是实验模拟，将雪花结构与 DCell 结构作比较；最后，总结全文.

2　相关工作

“数据中心”是上世纪 IT 界的一大发明，标志着 IT 应用的规范化和组织化. 传统数据中心结构[2]如图 1 所示.

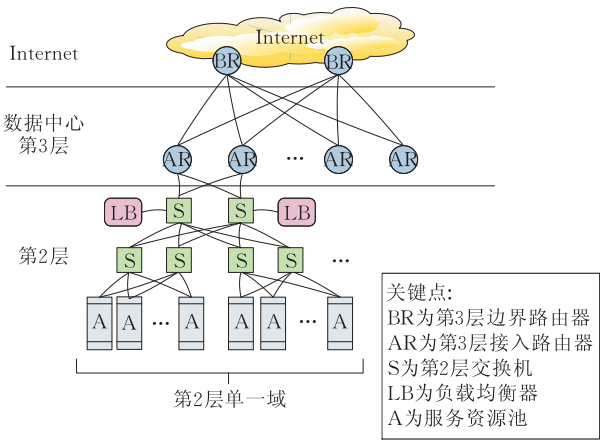

图 1　传统数据中心网络结构

Internet 中的请求经过第 3 层的边界路由器(Border Router，BR)和接入路由器(Access Router，AR)被转发到基于虚拟 IP 地址(Virtual IP，VIP)的第 2 层. 处于第 2 层顶端的交换机连接了两个负载均衡器，若其中一个均衡器失效，另一个仍然可以维持工作. VIP 就配置在这两个负载均衡器中. 此外，每个负载均衡器配置了一组目的 IP(Destination IP，DIP)，因此一组 DIP 与一个 VIP 相对应. DIP 是机架上物理服务器的私有内部地址. 这组 VIP 就定义了服务器资源池，用以解决发送给 VIP 的请求，而负载均衡器则负责在资源池中的 DIP 间传播请求.

然而随着科技的不断进步，传统数据中心的局限性也逐步显现出来，它存在以下不足之处[3]：

(1) 可扩展性差. 服务器数量限制在 4000 台左右，若建立百万规模的数据中心则会导致高额的成本和低效的网络性能.

(2) 静态的网络分配[2]. 为支持数据中心的内部流量，独立的应用通常被映射到特定的物理设备. 这种方式基于 VLAN 及其生成算法来满足服务器与特定应用映射. 虽然直接映射服务与相关物理设备可以提供一定的安全性和独立性，但是在执行管理和安全性维护时容易造成 VLAN 策略超载. 另外 VLAN 生成算法和大服务器池会将流量限制在树的顶端，易造成链路流量超载.

(3) 资源分片[2]. 主流的负载均衡技术通常要求同一个 VIP 资源池中的所有 DIP 在相同的两层中，因此应用需要更多服务器时，不能利用其它两层中的服务器，导致资源分片和低利用率. 虽然全 NAT 负载均衡允许利用其它两层中的服务器，但基于 IP 的应用通常要求服务器知道客户端的 IP，而全 NAT 无法支持.

(4) 人力成本高. 当服务需要在服务器间重分配时，传统数据中心网络的地址空间分片会导致巨大的人工配置成本，且人工操作出错的概率很高.

(5) 硬件成本高. 传统数据中心网络使用专用的交换机，位于上层的交换机成本较高. 此外，负载均衡器扩容时需要成对更新，成本较高.

因此，针对不断出现的新变化和新需求，新型数据中心必须满足以下特点[3-4]：

(1) 模块化的标准基础设施. 在新一代数据中心中，为使 IT 基础设施简化和具有适应性与可扩展性，需要对服务器、存储设备、网络等基本组成按标准进行模块化配置设计，以使这种配置更易于针对数据中心的服务需求量身打造. 基于标准的模块化系统能够简化数据中心的环境，加强对成本的控制，进而实现使用一套可扩展、灵活的 IT 系统和服务来构建更具适应性的基础设施环境，从而提高数据中心工作效率，降低复杂性和风险.

(2) 虚拟化的资源和环境. 在新一代数据中心中，广泛采用虚拟化技术将物理资源集中在一起形成一个共享虚拟资源池，从而更加灵活和低成本地使用资源. 通过服务器虚拟化、存储虚拟化、数据中

心虚拟化等解决方案,不仅可以降低服务器数量,还可以优化资源利用率.虚拟化是新一代数据中心中使用最为广泛的技术,也是与传统数据中心的最大差异.

(3)良好的扩展性.数据中心的可扩展性包括3个方面:①物理结构必须是可扩展的.理想的结构必须支持十万甚至百万台服务器的低成本扩展,每个节点的链路数不宜过多或者不依赖于高端交换机;②物理结构必须支持增量扩展.当增加新的服务器时,不会影响已有服务器的运行;③通信协议设计必须是可扩展的,例如路由协议.

(4)良好的容错性.在当前的数据中心中,故障是非常普遍的.硬件、软件和能源等因素可能引起各种各样的服务器、链路、交换机和机架故障.当网络规模足够大时,单独的服务器和链路的故障甚至比异常发生的频率更高,因此新型数据中心必须具备足够的物理冗余和良好的容错性.

(5)良好的服务器间通信性能.部署在数据中心的许多应用在服务器间的流量远大于与外部客户交互的流量,如网页检索、分布式文件系统、科学计算等.因此良好的服务器间通信性能是保障服务QoS的基础.

(6)位置无关的地址结构.服务需要采用与物理位置无关的地址结构来解决数据中心对服务器地址的限制问题.这样数据中心的任意服务器都可以成为任意资源池的一部分,既保证了服务的可扩展性又可以提高资源利用率,简化管理配置.

(7)节省能源和空间.传统数据中心设计追求的是性能,而新一代数据中心在当今能源紧缺与能源成本迅猛增涨的情况下必然需要综合考虑能源效率问题,提高数据中心空间利用率,解决传统数据中心的过量制冷和空间不足的问题.

目前有代表性的数据中心网络结构有Fat-Tree[5]、DCell[6]、BCube[7]和VL2[8]等.我们将详细介绍这几种结构并分析其各自的特点[3].

Fat-Tree结构将服务器分为k个子群,每个子群包含两层端口数为k的$k/2$个交换机,下层每个交换机的$k/2$个端口连接到$k/2$台主机,其余$k/2$个端口分别与每个聚和层交换机连接;核心层需要$(k/2)2$个端口数为k的核心交换机,最多能支持$k3/4$台主机.Fat-Tree抛弃了传统数据中心采用专用交换机的模式,转而采用商业以太网交换机,较大提高了性价比.它能为包含上万台服务器的数据中心提供高聚合带宽,不需要对主机网络接口、操作系统进行修改便可构建,且与以太网、TCP/IP等通信协议兼容良好.Fat-Tree各层的链路数相等,使得所有服务器产生的最大流量和核心层的最大吞吐量相等,不存在网络瓶颈.而且它采用两张路由表进行两级路由,并采用一定的链路错误检测机制来实现容错路由.Fat-Tree结构解除了树形结构上层链路对吞吐量的限制,并能为内部节点间通信提供多条并行链路.但是Fat-Tree的扩展性受限于核心交换机端口数量,目前比较常用的是48端口10GB核心交换机,在三层树结构中能够支持27648台主机.长远来讲,规模在十万以内的数据中心是无法满足应用需求的,因此Fat-Tree存在扩展性不足的缺点.Fat-Tree的另一个缺点是容错性差,具体表现为处理交换机故障能力不足及路由协议容错性不强.MSRA研究表明[9]:Fat-Tree对低层交换机故障非常敏感,严重影响系统性能.因为Fat-Tree仍然是树结构,本质上具有树结构的缺陷.

DCell是一种递归定义的网络结构,使用位于第$i-1$层的DCell构建第i层的DCell.当节点度增加时DCell的规模接近以2的指数次方扩展.通常$DCell_0$内包含常数服务器,一般为3～8台,并通过微型交换机互连.DCell容错性较好,没有单点故障并且能够在严重的链路和节点故障的情况下利用其分布式协议实现接近最短路径的路由.DCell还能为各种各样的服务提供比传统树结构更高的网络容量.另外DCell可以增量扩展并且在不完全结构的情况下表现出上述性能.虽然$DCell_0$很小,但DCell能支持的服务器数量是惊人的,例如当$DCell_0$包含6台主机时,$DCell_3$可以支持326万台服务器.由于DCell连接方式接近完全图,并且DCell路由协议(DCell Fault-tolerant Routing,DFR)利用链路状态和贪心算法来实现容错路由,所以DCell可以在服务器、链路或交换机严重故障的情况下,实现性能较好的路由.然而,DCell也有不足之处.首先,其完全图的连接方式可能带来巨大花费,而且实际链路规模庞大,连接和维护困难.其次,DCell中流量在不同层次分布不均匀,level0承担了过多流量,严重影响吞吐量.最后,由于DCell使用服务器执行路由,增大了网络延迟,而且其路由协议也不适于在链路故障时发现最短路径,网络延迟较大.

BCube是DCell的模块化版本[10-12],它的连接方式是在$BCube_0$以外的层次中采用微型交换机实现连接.它在每层单元的数量上与DCell不同,如果

DCell 在第 k 层有 n 个 $DCell_{k-1}$，那么第 $k+1$ 层则有 $n+1$ 个 $DCell_k$；而 BCube 在任一级都具有相同的单元数 n，易得出 $BCube_k$ 拥有 $nk+1$ 台服务器（n 为 $BCube_0$ 中的服务器数）. 交换机作为连接媒介使 BCube 具有很多冗余路径，这可以保证容错路由并方便模块化连接而且路由速度比 DCell 快. 另外 BCube 应用 BSR（BCube Source Routing）选取路径，采用路径自适应协议，能够很好地开拓网络中的最短并行路径，并实现可靠数据传输. BCube 能够高效无带宽限制地执行 One to One、One to All、One to Several、All to All 类型的通信，很好地支持 GFS，MapReduce 类应用. BCube 的连接方式和递归结构使得数据中心可以模块化建设，实现性好. BCube 在结构不完整的情况下表现出比 DCell 具有更好的性能. 实验表明[9]给定 2048 台服务器，在 Fat-Tree、DCell 和 BCube 结构都不完整的情况下，BCube 的网络吞吐量和容错性能是最好的，这是因为 BCube 对不完整结构采用完整结构的交换机连接策略. BCube 的不足体现在可扩展性上，BCube 在 $k=3$、$n=8$ 时（k 为 BCube 层次，n 为 $BCube_0$ 中服务器数），仅支持 4096 台服务器，与 DCell 差距较大.

VL2 是一种可扩展的灵活的数据中心网络结构，它能够支持超大规模的数据中心，为服务器间提供均衡的高带宽通信性能，服务间的性能隔离及以太网第 2 层语义. 第 2 层语义是指将第 2 层所有的域虚拟化为统一的域，在这个层面上所有主机都位于同一个域中. VL2 在结构上改变的是第 3 层交换机的连接方式，采用特殊协议实现虚拟第 2 层；而其它结构在物理连接方面的改变是整体. 另外，地址表示和路由协议在 VL2 中更为重要，直接关系到虚拟第 2 层的实现. VL2 中采用 VLB（Valiant Load Balancing）进行路由，为均衡各路径流量，VL2 将各中间交换机设为相同 IP，采用随机的方式选择一个中间交换机实现路由.

3　新型数据中心网络结构——雪花结构

这一部分详细介绍雪花结构的构建方法及其特有的一些属性. 之所以称为雪花结构，是因为这种结构依据科赫曲线，形似科赫雪花.

3.1　雪花结构及其构建方法

雪花结构包含两个组成部分：服务器和微型交换机. 前面已经介绍了 DCell 和 BCube 结构，二者均采用了递归的方法来定义各自结构. 在雪花结构中，我们同样采用递归定义的方法，在（$n-1$）级雪花结构的基础上添加若干个 0 级雪花结构，构成 n 级雪花结构. 这里的“若干个”我们在后面会详细介绍. 这样定义的好处是，首先，我们用添加某种固定结构的方式尽可能地将结构模块化处理，有利于结构的模块化连接；其次，当 n 级结构没有扩展完全时，继续扩展（$n+1$）级结构也较容易. 此时，若发现 n 级结构没有扩展完全，可以在不改变已有（$n+1$）级结构的情况下，继续补充不完全的 n 级结构，有利于结构的扩展. 下面我们来详细介绍雪花结构的构建方法.

0 级雪花结构（$Snow_0$）由一个微型交换机和若干个服务器组成. 借鉴已有数据中心结构，设置服务器的个数最少为 3 个，不超过 8 个[6-7]. 如图 2（a）所示，我们以 3 个服务器为例（$k=3$），将 3 个服务器连接到一个 n 端口的微型交换机上. 如图 2（b）所示，我们调整 3 个服务器的理论位置，为 3 个服务器的两两直接互联添加了 3 条虚线. 这 3 条虚线并非实际的服务器互联，而是为了方便说明构造下一级雪花结构，我们称之为虚连接，即不存在的网络连接，只是为了方便说明结构的构建. 在 $Snow_0$ 基础上，我们断开 3 条虚连接（如图 2（b））；每断开一条虚连接，即要添加一个 0 级雪花结构，将该新添 0 级结构中的微型交换机分别与虚连接的两端节点相连. 相对于虚连接，这样的连接称为实连接，即断开虚连接重新构建的连接. 这里的实连接是实际存在的连接，是新添加结构中的交换机与原有结构中服务器的连接. 这样得到一个 1 级（$Snow_1$）雪花结构，如图 3 所示. 由于是在断开的虚连接处添加 0 级雪花，此时虚连接的状态已经变为实连接（1 条虚连接转化为 2 条实连接），链接状态从无到有，因此在此处不再添加虚连接. 需要特别说明的是，实连接不只是实际存在的连接，更体现了由虚连接到实连接，这种连接从无到有的变化状态，因此在 $Snow_0$ 中服务器与交换机相连的 3 处不看作是实连接，它虽然是实际存在的连接，但是并没有状态变化的体现. 仔细观察可以

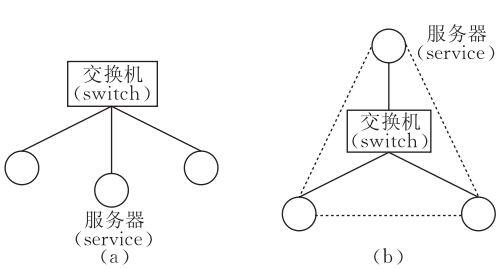

图 2　$k=3$ 时的 $Snow_0$

发现，断开虚连接添加的 0 级结构与 $Snow_0$ 是有区别的，它缺少了一条虚连接，并不是真正意义上的 $Snow_0$ 结构. 为了区分二者，不至于混淆读者，这种后来断开虚连接不断添加的缺少一条虚连接的 0 级结构，我们称其为 Cell，以区分 $Snow_0$.

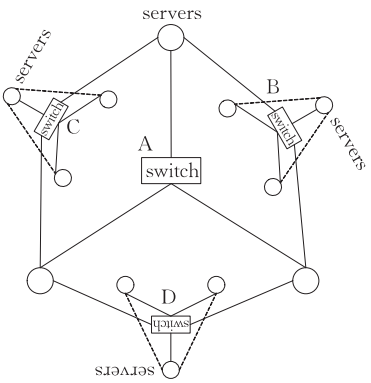

图 3 $k=3$ 时的 $Snow_1$

$Snow_1$ 包含 6 条实连接和 6 条虚连接. 在之后的每一级雪花结构中，总是断开上一级中包含的所有实连接和虚连接，添加 Cell，构成新的高级结构. 图 4 显示了 k 为 3 时的 2 级($Snow_2$)雪花结构.

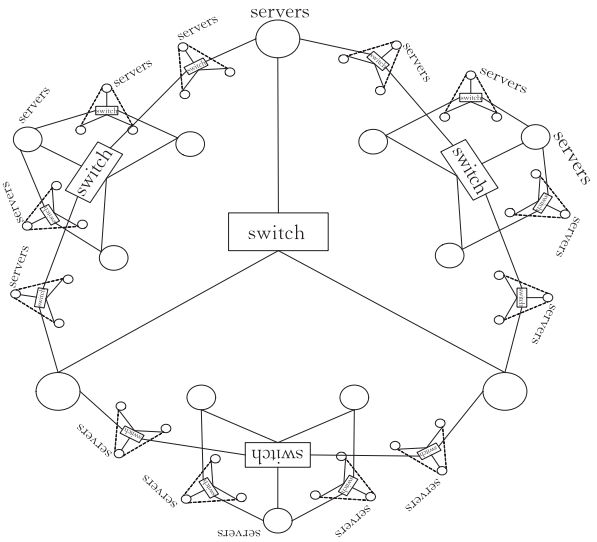

图 4 $k=3$ 时的 $Snow_2$

构建雪花结构的伪代码如算法 1 所示.

算法 1. 构建雪花结构.

```
1. Construct Snow(k,n)
      //k 是 Snow0 中包含的服务器个数,构建 Snown
2. If (n==0) //构建的是 Snow0
3.   For (int i=0; i<k; i++)
4.     Connect ith server to switch;
5.     标记两两相邻的服务器; //虚连接标记
6.     将标记的服务器成对放入队列;
     //记录虚连接,将被虚连接的节点对加入队列
7.     记录队列长度 queue_len;
     //队列长度即下一级要添加 Cell 的个数
8.   Return;
9. Else while (n!=0)
10.  {while (queue_len!=0)
11.    {从队列中取出节点对,添加 Cell, Cell 中的交
       换机分别连接原节点对中的两个节点,形成
       实连接;
12.    将添加 Cell 产生的 2 条新的虚连接和 2 条新
       的实连接节点对关系加入队列;
13.    记录新的队列长 queue_new_len;
14.    queue_len--; //添加一次 Cell 队列长减一
15.    }
16.   n--; //当队列长为 0 时,表明新的一级雪花
            结构构建完成
17.  queue_len=queue_new_len;
18.  }
```

3.2 雪花结构的属性

在这一节，我们介绍并证明雪花结构的特有属性.

定理 1. 假设第 0 级雪花结构中包含 k 个服务器(Cell 也包含 k 个服务器)，则第 n 级雪花结构包含 SV_n 个服务器，SW_n 个微型交换机. 其中 $k\in\{3,4,5,6,7,8\}$，$n\geq 0$. SV_n 和 SW_n 分别满足下列条件：

$$SV_n = SW_n \times k \tag{1}$$

$$SV_n = k\times(k+1)^n \tag{2}$$

证明. $Snow_0$ 包含 k 个服务器，因此 $Snow_0$ 中，$SV_n : SW_n = k : 1$.

Cell 包含 k 个服务器，因此 Cell 中，$SV_n : SW_n = k : 1$.

每一级雪花结构都是在前一级的基础上不断添加 Cell，服务器与交换机的比例没有改变，因此 $SV_n : SW_n = k : 1$ 是显而易见的. 变换形式即得到式(1)结论.

下面，我们用数学归纳法来证明式(2).

当 $n=0$ 时，$SV_0 = k$，$k\times(k+1)^n = k$，此时，$SV_n = k\times(k+1)^n$.

当 $n=1$ 时，$SV_1 = k+k\times k = k(k+1)$，$k\times(k+1)^n = k(k+1)$，$SV_n = k\times(k+1)^n$.

假设，当 $n=m$ 时，等式成立，即 $SV_m = k\times(k+1)^m$.

则当 $n=m+1$ 时，$SV_{m+1} = SV_m + k\times$(断开 $Snow_m$ 中虚连接和实连接添加的 Cell 个数).

这里，断开 $Snow_m$ 中虚连接和实连接添加的

Cell 个数＝$Snow_m$ 中虚连接和实连接个数；

$Snow_m$ 中虚连接和实连接个数＝$(k+1)\times[SV_m-SV_{m-1}]/k=k\times(k+1)^m$；

因此 $SV_{m+1}=k\times(k+1)^{m+1}$.

所以，$SV_n=k\times(k+1)^n$，等式(2)成立.　证毕.

从式(1)和(2)可以得到，当 $k=3$、$n=10$ 时，雪花结构可以达到 314 万个服务器. 该结构在最初阶段扩展较慢，但是一旦形成即迅速增长. 当 $k=6$、$n=4$ 时，雪花结构可以达到 14406 个服务器、2401 个交换机，交换机与服务器的比例为 0.167. 在 BCube 中，当 $k=6$、$n=4$ 时，只能达到 7776 个服务器，却同时包含交换机 2850 个. 不仅服务器数量减半，扩展性较弱，而且交换机与服务器的比例达到了 0.367，是雪花结构的 2 倍多. 当 $k=4$、$n=4$ 时，BCube 的交换机与服务器比例甚至达到 2.328，交换机的数量是服务器的 2 倍多. 考虑到数据中心这种大规模结构的制冷开销以及交换机成本，这种交换机个数的较高比例甚至超过服务器同比例增长，显然是不利的. 此外，雪花结构在保证交换机的低数量方面也是很有优势的，当 $k=6$ 时，始终保持交换机与服务器的比例为 0.167，最坏情况 $k=3$ 时，也可以达到 0.333 的低比例.

3.3　雪花结构的性能分析

在这一节，我们从网络交换容量、瓶颈链路以及延时 3 个方面来分析雪花结构的性能.

3.3.1　网络交换容量

我们假设交换机与交换机之间的带宽为 A Mbps，交换机与服务器之间的带宽为 B Mbps，一般情况下 A 大于等于 B.

当任意 2 个服务器之间需要发送数据时，这 2 个服务器之间的带宽受限于二者之间路径上的最低带宽值.

雪花结构的一大特点就是用较少的交换机连接较多的服务器，网络扩展性好. 同时，节点之间的带宽也较大. 综合以上的特点，该结构适合于节点之间有频繁通信，且需要较大节点规模的应用. 例如在该结构上运行大规模数据挖掘等.

3.3.2　瓶颈链路

我们将瓶颈分为 2 种：节点瓶颈与路径瓶颈.

节点瓶颈是指，某个节点(服务器或者交换机)由于数据流量过大成为瓶颈. 越是层级低的节点越容易成为瓶颈节点. 因此我们在第 5 节补充问题研究里面将低层级中的一些服务器改进为交换机，其中的交换机也可以换成高吞吐量如万兆交换机，以减轻数据传输的压力.

路径瓶颈是指，2 个子网之间有且只有一条路径，当该路径断开时，这 2 个子网即失去连接，这样的路径我们称为瓶颈路径. 在定理 5 中我们已经证明，Snow_n 中任意 2 个服务器包含并行路径，至少 2 条不超过 2^{2n} 条. 因此理论上雪花结构中不存在路径瓶颈的问题.

3.3.3　延　时

RTT(Round-Trip Time)表示从发送端发送数据开始，到发送端收到来自接收端的确认(接收端收到数据后便立即发送确认)，总共经历的时延. 当 2 点之间的 RTT 值较大时说明这 2 点之间的延时较大；反之，则说明 2 点之间延时较小. 从 RTT 的定义可以看出，RTT 与 2 点之间的最短路径成正比. 当 2 点之间距离较长时，RTT 较大；反之，则 RTT 较小. 我们在定理 4 及推论 4 中已经得出最长最短路径为 $(2n+1)$ 跳. 因此，延时为 $O(n)$.

4　雪花结构中协议和路由策略

前面已经说到，当 $k=3$，$n=10$ 时，雪花结构可以达到 314 万个服务器，由于该结构中的服务器个数随着 n 值的增加呈指数次方不断增长，它的目标是扩展连接十万百万量级的服务器，因此基于雪花结构的数据中心不适合使用全局的链路路由机制，易造成带宽瓶颈和单点失效的 OSPF[13] 协议自然不适合这种结构.

在这一节，我们首先介绍无失效情况下节点路由情况；接着提出基于雪花结构的路由协议；最后，详细阐述了依据路由协议的节点路由情况.

4.1　无失效路由

在这一小节，首先分析雪花结构中的无失效路由情况. 假设源节点为 *src*，目的节点为 *des*.

定理 2.　$Snow_0$ 中任意 2 个服务器之间仅 1 跳.

如图 2(a)和(b)所示，$Snow_0$ 中任意两个服务器间通过交换机即可达，1 跳显然.

定理 3.　$k=3$ 时，$Snow_1$ 中任意 2 个服务器之间不超过 2 跳. 当 $k\in\{4,5,6,7,8\}$ 时，任意 2 个服务器之间不超过 3 跳.

$k=3$ 时，如图 3 所示，$Snow_1$ 中节点路由可以分为以下 5 种情况：

$Snow_1$ 中服务器→$Snow_0$ 中服务器：1 跳；

$Snow_0$ 中服务器→$Snow_1$ 中服务器：1 跳；

$Snow_1$ 中服务器→$Snow_0$ 中服务器→$Snow_1$ 中服务器:2 跳;

$Snow_1$ 中服务器→$Snow_0$ 中服务器→$Snow_0$ 中服务器:2 跳;

$Snow_0$ 中服务器→$Snow_0$ 中服务器→$Snow_1$ 中服务器:2 跳.

不超过 2 跳的结论是很容易得到的.

当 $k\in\{4,5,6,7,8\}$时,同理可得到 3 跳结论.

定理 4. $Snow_n$ 中任意 2 个服务器之间最长最短路径为$(2n+1)$跳.其中最短路径是指 2 个服务器之间路由经过的最少服务器个数.

假设源节点 *src* 位于 $Snow_j$ 中,目的节点 *des* 位于 $Snow_k$ 中.j 和 k 都小于等于 n.

此时,考虑最长最短路径情况.源节点 *src* 从 $Snow_j$ 逐级向下经由 $Snow_{j-1}$,$Snow_{j-2}$…路由到 $Snow_0$ 中的服务器,最多需要 j 跳.$Snow_0$ 中的服务器经由 $Snow_1$,$Snow_2$…路由到 $Snow_k$ 中的服务器最多需要 k 跳.而 $Snow_0$ 中的服务器彼此之间最多 1 跳.因此 $Snow_j$ 中的源节点 *src* 路由到 $Snow_k$ 中的目的节点 *des* 最长最短路径为$j+k+1$ 跳.当 j 和 k 都取最大值 n 时,*src* 与 *des* 的最长最短路径达到最大值 $2n+1$ 跳.

由上述分析我们还可以得到,当 j 和 k 不同时取到 n 或者 j 和 k 不相等时,*src* 与 *des* 的最长最短路径必然小于等于 $2n+1$ 跳.因此,我们得到推论 4.

推论 4. $Snow_j$ 与 $Snow_k$(j 不等于 k,且 j 与 k 均小于等于 n)中任意 2 个服务器之间最长最短路径上限为 $2n+1$ 跳.

定理 5. $Snow_n$ 中,任意 2 个服务器包含并行路径,至少 2 条,不超过 2^{2n} 条.并行路径是指,2 个服务器之间同时存在的路径,这些路径彼此之间独立无依赖关系.

我们通过在前一级雪花结构的基础上断开所有的实连接和虚连接,添加 Cell 来构建下一级雪花结构,每一个添加的 Cell 都是通过交换机的左右两条连接才加入新结构中,因此,任意 2 个服务器之间至少包含 2 条并行路径.

当 $Snow_j$ 中源节点 *src* 路由到 $Snow_k$ 中的目的节点 *des* 时,*src* 从 $Snow_j$ 开始经由$Snow_{j-1}$ $Snow_{j-2}$…路由到 $Snow_0$,每一次从 $Snow_p$ 到 $Snow_q$ 总是有 2 条并行路径,因此至多包含 2^j 条并行路径.同理,$Snow_0$ 到 $Snow_k$ 也至多包含 2^k 条路径.因此,从 $Snow_j$ 的源节点 *src* 路由到 $Snow_k$ 的目的节点 *des* 至多有 2^{j+k} 条并行路径.当 *src* 与 *des* 都处于 $Snow_n$ 时,两节点至多包含 2^{2n} 条并行路径.

4.2 协 议

4.2.1 节点唯一标识方法

我们对每一个服务器和交换机进行标记,标记结构为〈级别,度数〉.级别表明该服务器处于雪花结构的这个级别中,是在该级别被添加进雪花结构的.然而由于在每个级别中并非只添加了一个节点,仅仅级别这一度量还不足以唯一标识雪花结构中的节点位置,因此,我们引入度数这个度量,结合级别唯一标识雪花结构中的每一个服务器和交换机.下面详细解释节点标识的方法.

如图 5 所示,标记 $Snow_0$ 中的 3 个服务器分别为 $A\langle 0,0/360\rangle$,$B\langle 0,120\rangle$,$C\langle 0,240\rangle$.其中第 1 位表示级别,即这 3 个服务器均处于 $Snow_0$ 中.第 2 位表示度数,这里的度数表明的是相对位置,均是以 $Snow_0$ 中的交换机为基础参照物得到的相对位置.因此 $Snow_0$ 中的交换机被标记为$\langle 0,0\rangle$,服务器 A 在交换机的正上方,相对度数为 0°,依照顺时针方向,服务器 B 为 120°,服务器 C 为 240°.这样 $Snow_0$ 中的 3 个服务器即被唯一标识.为了区分服务器与交换机,对交换机添加 w 标记,即 $Snow_0$ 中的交换机标记为 $S\langle 0,0,w\rangle$.需要说明的是,对于特殊位置的服务器 A,同时标记其度数为 360°,方便构造高级雪花结构的标识.

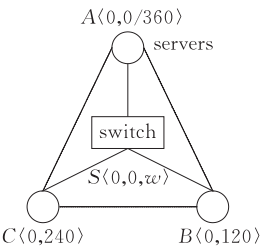

图 5 $Snow_0$ 中的节点标识

我们在 3.1 节已经阐述了如何构建 $Snow_1$.这里,同样利用产生 $Snow_1$ 的 3 个 $Snow_0$ 服务器来产生 $Snow_1$ 的标识.如图 6 所示,$Snow_1$ 中的交换机 D 所处 Cell 添加在服务器 A、B 之间并与其直接相连,利用 A、B 来产生 D 的标识,记 D 的标识为$\langle 1,60,w\rangle$,即 $Snow_0$ 构建产生 $Snow_1$,级别为 1,度数为 A、B 和的一半.用这一位置同时标记交换机 D 的正上方(相对于 D 的位置)服务器 E 为$\langle 1,60\rangle$.前面说到服务器 A 的度数同时标记为 360°,当 A、C 用以标识交换机 M 时,采用 A 的 360°构造 M 为$\langle 1,300,w\rangle$.即 A 与 180°内的节点构造新标识时采用 0°,与 180°~360°内的节点构造新标识时采用 360°.

这样做的好处是可以形成一个规范化的节点唯一标识方法.

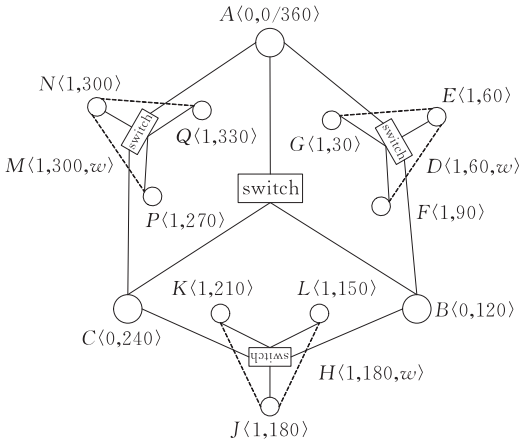

图 6　$Snow_1$ 中的节点标识

4.2.2　请求数据包的结构定义

定义源节点 src 发送的请求数据包中，包含 src 的标识信息，数据包 ID 以及目的节点 des 的标识信息. 这样一个请求数据包可以通过其自身 ID 与 src 的标识来唯一标记. 当转发数据包的路由节点发现已经转发过某一数据包时，即放弃该数据包不再转发以降低网络流量. 当目标节点 des 收到请求数据包后，也可以根据包中携带的 src 的标识信息获得请求节点的相关信息. 具体的路由机制在 4.3 节详细阐述.

4.3　路由策略

我们定义源节点 $src\langle j,a\rangle$ 和目的节点 $des\langle k,b\rangle$. 当 src 发送请求数据包时，包内携带的信息包含 $(ID,\langle j,a\rangle,\langle k,b\rangle)$.

如图 7 所示，在该雪花结构中，虚线表示中间省略的若干级别. 左节点标记为 $src\langle j,a\rangle$，右节点标记为 $des\langle k,b\rangle$. 当 src 要发送数据包给 des 时，首先查看 des 的度数 b 的所属范围. 由于 b 属于 0～120°范围则通过服务器 A 或者 C 路由到 B，再经由中间若干交换器不断地接近 des，最后从服务器 G 到达 des. 根据度数 b 可以确定路由的方向，根据级别 k 可以确定路由的大概路径长度.

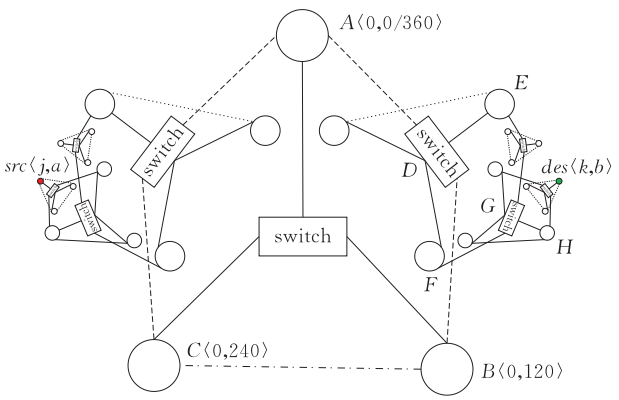

图 7　src 路由 des 示意图

以图 8 为例，src 需要发送请求包给 des 时，首先查看 des 的度数 b，这里为 45°. src 第 1 步经由交换机 $M\langle 1,300,w\rangle$ 可以路由到 N 或者 C，由于 N 更接近服务器 $A\langle 0,0/360\rangle$，A 同时为 0°较 C 更靠近 des，因此 src 选择 M—N—A 的路径. A 可达 $H\langle 2,30,w\rangle$，虽然与 des 同层级，但是度数较低，继续路由达交换机 $D\langle 1,60,w\rangle$. D 可达 $K\langle 2,90,w\rangle$，虽然 K 与 des 同级但是度数较大，进一步发现 D 连接的两个服务器 $E\langle 1,60\rangle$ 和 $G\langle 1,30\rangle$ 均属于 $Snow_1$，E 和 G 可构建 $Snow_2$ 级，且 45°属于 30°～60°之间，因此，D 经由 E 或者 G 到交换机 $F\langle 2,45,w\rangle$，最后达 des.

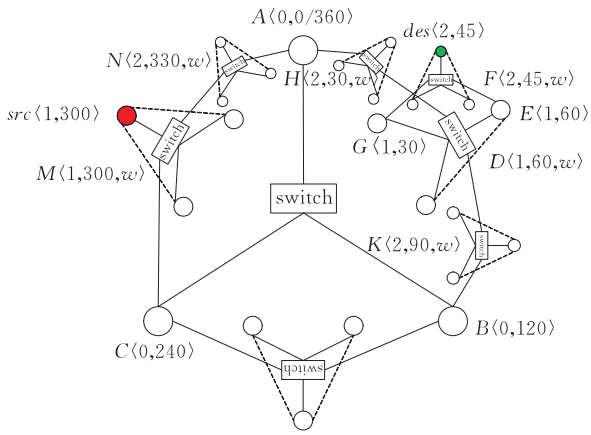

图 8　src 路由 des 实例

5　补充问题研究

当雪花级别不断扩大，结构中的服务器数目不断增多，达到百万级别时，路由的消息数量剧增，此时 $Snow_1$ 中的 12 个服务器显然会负载较重的信息转发任务，因此我们对上述提出的雪花结构略微改进，将 $Snow_1$ 中的 12 台服务器替换为交换机，如图 9 所示，以缓解低级雪花结构的负载强度，提高消息转发速度.

我们在 $Snow_{n-1}$ 的基础上添加若干个 Cell 形成 $Snow_n$. 此时，若新添加的 Cell 中的交换机发生故障时，会导致该交换机连接的 3 台服务器完全脱离雪花结构. 当这种情况出现较为严重时，$Snow_n$ 即退化为 $Snow_{n-1}$ 级结构，可用服务器数量显著受到影响. 而这种影响源于交换机并非服务器本身造成. 为了避

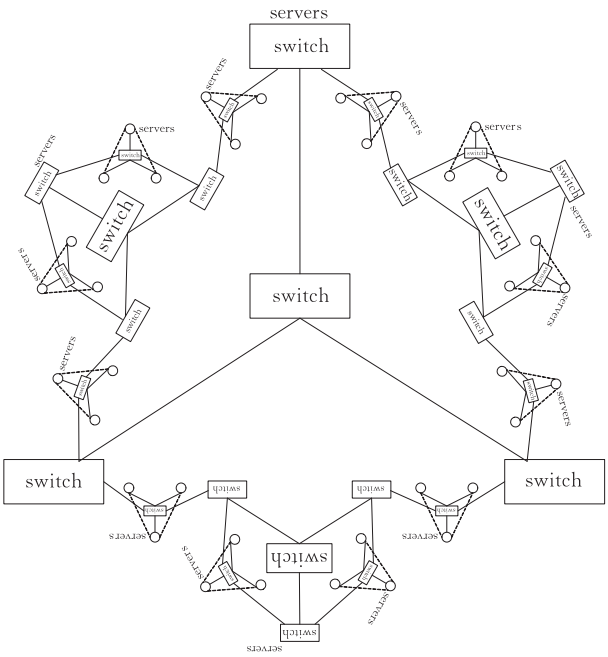

图 9 改进 $Snow_1$ 后的雪花结构

免这种情况的出现，我们补充规定，在新添加的 Cell 中，3 台服务器轮流每 20 s 发送一个查询交换机是否存活的消息，这样每个服务器平均 1 min 发送一次消息. 当 3 台服务器不能连通时表明交换机出现故障，服务器已经脱离雪花结构，这时需要检查该交换机，确保其稳定工作.

6 实验模拟

在这一节，我们采用模拟的方法来评估雪花结构的性能. 模拟程序采用 Java 编写，开发工具为 Eclipse 3.5.0，在一台联想 X200 笔记本上运行.

场景 1. 无节点失效时，不同层级节点总数及平均最短路径长度.

首先来看无节点失效情况. 表 1 显示了当 $k=3$ 时，$Snow_0$ 中的服务器到其余节点的平均最短路径情况. 由于未考虑节点失效情况，因此不可达节点数均为 0. 由平均最短路径长度可以发现，在 $Snow_4$ 的范围内，平均路径可以保持在 3 跳内，与 4.1 节中的定理 4 相符，但是实验结果优于定理 4 的理论推导结果.

表 1 无节点失效情况下，各个节点到 $Snow_0$ 中 3 个服务器的平均最短路径情况

雪花级别	服务器总数	交换机总数	可达节点数	不可达节点数	最短路径总和	平均最短路径长度
$Snow_2$	48	16	47	0	80	1.702
$Snow_3$	192	64	191	0	416	2.178
$Snow_4$	768	256	767	0	2048	2.670

场景 2. 有节点失效时，平均最短路径长度与路径失效率.

这里的节点失效是指服务器故障.

当 $k=3$ 时，我们详细设置了节点失效比例，测试了 $Snow_4$ 中，随着节点失效比例的增加，平均最短路径长度变化情况.

从图 10 可以看出，当节点失效保持在 0.05～0.30 时，平均路径长度波动不大，基本处于 2.6～2.7 之间.

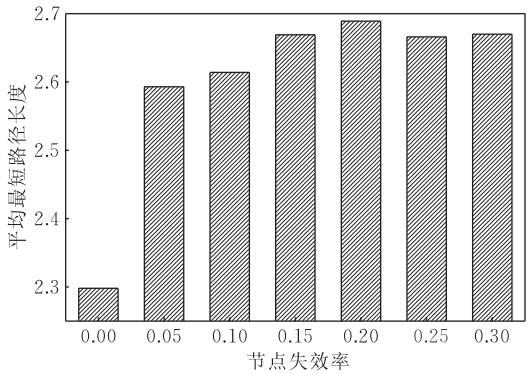

图 10 随着节点失效率的增加 $Snow_4$ 平均最短路径长度变化情况

当 $k=3$ 时，我们测试在了 $Snow_4$ 中随着节点失效率的增加，路径失效率的情况，并与 DCell[6] 中的实验结果做了对比. 图 11 所示为 $Snow_4$ 与 $DCell_4$ 的路径失效对比图. 从图中可以看出，Snow 的路径失效率围绕 DCell 上下波动，差别不大.

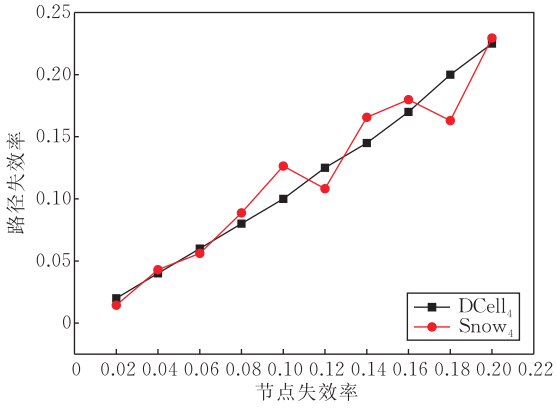

图 11 随着节点失效率的增加 $Snow_4$ 与 $DCell_4$ 的路径失效率比较

场景 3. 有链路失效时，平均最短路径长度与路径失效率.

这里的链路失效是指交换机与服务器之间的链接断开，或者交换机故障，并非服务器本身故障.

从图 12 可以看出，随着链路失效率的增加，平均最短路径长度变化幅度不大. 链路失效率在 0.2 以内时，平均最短路径长度始终在 2.6～2.9 之间.

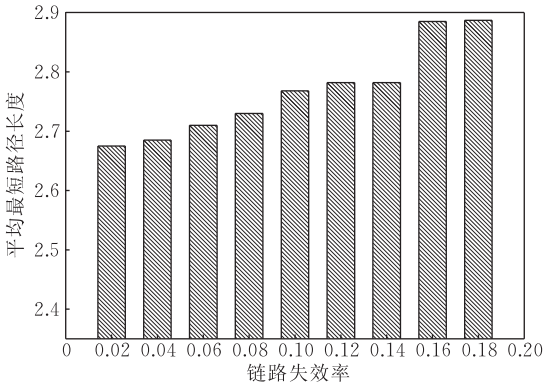

图 12　随着链路失效率的增加 $Snow_4$ 平均最短路径长度变化情况

当 $k=3$ 时，我们测试在了 $Snow_4$ 中随着链路失效率的增加，路径失效率的变化情况，同样与 DCell[6] 中的实验结果做了对比. 图 13 所示为 $Snow_4$ 与 $DCell_4$ 的路径失效对比图. 从图中可以看出，Snow 的路径失效率比较稳定，始终保持 0.001～0.004 之间.

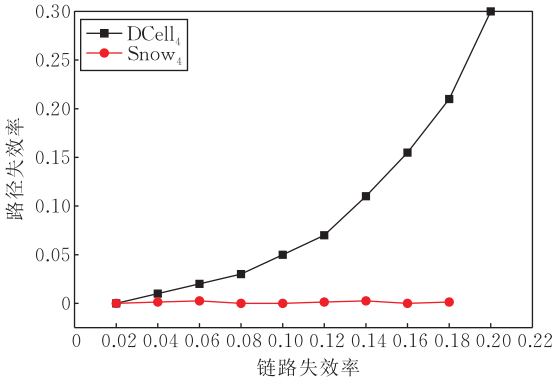

图 13　随着链路失效率的增加 $Snow_4$ 与 $DCell_4$ 的路径失效率比较

最后，我们综合比较了 DCell、BCube 以及 Snow，如表 2 所示.

表 2　DCell、BCube 和 Snow 的比较

网络结构	DCell	BCube	Snow
交换机与服务器的比例	高	高	低(优点)
节点失效与路径失效的关系	较好	较好	与 DCell 效果相当
链路失效与路径失效的关系	较好	较好	较 DCell 失效率更低更稳定
网络吞吐量	由于网络扩展性较好，因此有较高的网络吞吐量	与 DCell 相比较差	低级别的服务器和交换机可能会成为系统瓶颈(缺点)

7　结束语

本文分析了传统数据中心的不足以及新型数据中心具备的新特点，借鉴已有数据中心结构，依据著名科赫曲线，提出了新型数据中心网络结构——雪花结构. 该结构充分考虑了数据中心的可扩展性，在保证交换机与服务器较低数量比例的前提下，可以在较短的平均路径内实现节点路由机制，具有较小的网络开销.

参　考　文　献

[1] http://zh.wikipedia.org/zh-cn/%E7%A7%91%E8%B5%AB%E6%9B%B2%E7%BA%BF

[2] Albert Greenber, Parantap Lahiri, David A Maltz, Parveen Patel, Sudipta Sengupta. Towards a next generation data architecture: Scalability and commoditization//Proceedings of the ACM Workshop on Programmable Routers for Extensible Services of Tomorrow. Seattle, WA, USA, 2008: 57-62

[3] Niu Xianlong, Chen Huaping, Zhou Wenyu, Wu Bin, Liu Xiaoqian. A survey on the new-type network structure of data center//Proceedings of the China National Grid Annual. Beijing, 2010: 35-39(in Chinese)
(牛宪龙，陈华平，周文煜，武斌，刘晓茜. 新型数据中心网络结构研究进展综述. 中国国家网络年会论文集，北京，2010：35-39)

[4] Zhu Weixiong, Wang De'an, Cai Jianhua. The theory and practice of new data center. The People's Posts and Telecommunications Press, 2009(in Chinese)
(朱伟雄，王德安，蔡建华. 新一代数据中心建设理论与实践. 北京：人民邮电出版社，2009)

[5] Leiserson C E. Fat-trees: Universal neworks for hardware-efficient supercomputing. IEEE Transactions on Computers, 1985, C34(10): 892-901

[6] Guo Chuan-Xiong. DCell: A scalable and fault-tolerant network structure for data centers//Proceedings of the SIGCOMM 2008. Seattle, WA, USA, 2008: 75-86

[7] Guo Chuan-Xiong. BCube: A high performance, server-centric network architecture for modular data center//Proceedings of the SIGCOMM 2009. Barcelona, Spain, 2009: 63-74

[8] Albert Greenberg. VL2: A scalable and flexible data center network//Proceedings of the SIGCOMM 2008. Seattle, WA, USA, 2008: 51-62

[9] Greenberg A, Hamilton J, Maltz D A, Patel P. Cost of cloud. ACM SIGCOMM Computer Communication Review, 2009, 39(1):

[10] Naous J, Gibb G, Bolouki S, McKeown N. NetFPGA: Reusable router architecture for experimental research//Proceedings of the PRESTO'08, 2008

[11] Hamilton J. Cooperative expandable micro slice servers (CEMS)//Proceedings of the 4th CIDR. Asilomar, CA, USA, 2009

[12] Lockwood J, McKeown N, Watson G, Gibb G, Hartke P, Naous J, Raghuraman R, Luo J. NetFPGA—An open platform for gigabit-rate network switching and routing//Pro-

ceedings of the IEEE International Conference on Microelectronic Systems Education. San Diego, CA, USA, 2007

[13] Moy J. OSPF Version 2. RFC 2328, April 1998

LIU Xiao-Qian, born in 1983, Ph.D. candidate. Her research interests include cloud and grid computing.

YANG Shou-Bao, born in 1947, professor, Ph. D. supervisor. His research interests include cloud and grid computing, wireless network.

GUO Liang-Min, born in 1980, Ph. D. candidate. Her research interests include cloud and P2P computing.

WANG Shu-Ling, born in 1988, Ph. D. candidate. Her research interests include wide area service discovery and cloud storage.

SONG Hu, born in 1986, Ph. D. candidate. His research interests include cloud computing and high performance calculation.

Background

This paper is sponsored by the National Natural Science Foundation of China (No. 60673172), National High Technology Research and Development Program (863 Program) of China (No. 2006AA01A110) and USTC Innovation Foundation "Campus Cloud and the Typical Application" (No. KD0901110).

This work focuses on Data Center Network Structure which is a hot topic in recent years. From 2008 up to now, there were several famous data center structures, such as Fat-Tree, DCell, BCull and VL2. Everyone has its own characteristics. Fat-Tree follows the merits of tree structure, but still has the problem of single node failure. The scalability of DCell is very good, but the network flow of $DCell_0$ and maintaining cost are a little large. The mean path length of BCull is short, while, its scalability is bad and uses many switches.

This paper refers to Koch Curve, and proposes a new-type data center structure, called Snow Structure for the first time. It takes scalability into full consideration, keeps servers in exponential growth of n, and obviously is better than BCull. In addition, it reaches the destination node in $O(n)$ hop, and the numbers of switches reduces clearly.