

# 基于存储联盟的双层动态副本创建策略—SADDRES

孙海燕,王晓东,周斌,贾焰,王怀民,邹鹏

(国防科技大学计算机学院,湖南长沙 410073)

**摘要:** 针对面向企业的数据网格应用,给出了平民数据网格 DGHB 的概念,并提出了针对 DGHB 网格的一种基于存储联盟的双层动态副本创建策略—SADDRES. 通过在数据网格模拟工具 OptorSim 中的模拟,分析、比较了 SADDRES 策略与基于缓存的副本创建策略、基于经济模型的副本创建策略等多种策略的性能. 实验结果表明,SADDRES 策略在面向企业的数据网格应用中具有更好的系统性能.

**关键词:** 企业数据网格;存储联盟;复制;副本创建策略

**中图分类号:** TP393 **文献标识码:** A **文章编号:** 0372-2112 (2005) 05-1222-05

## The Storage Alliance Based Double-Layer Dynamic Replica Creation Strategy-SADDRES

SUN Hai-yan, WANG Xiao-dong, ZHOU Bin, JIA Yan, WANG Huai-min, ZOU Peng

(School of Computer Science, National University of Defense Technology, Changsha, Hunan 410073, China)

**Abstract:** Data Replication technologies have been widely applied in data grid systems to improve the performance of grid applications, and replica creation strategy is one of the key technologies of Data Replication. A new concept Data Grid of Humble Birth (DGHB) is proposed for enterprise-oriented data grid applications. And a replica creation strategy - Storage Alliance based Double-layer Dynamic Replica Creation Strategy (SADDRES) is also proposed. The simulation results on OptorSim prove that the SADDRES is more efficient than other replica creation strategies for enterprise-oriented data grid applications.

**Key words:** enterprise-oriented data grid; storage alliance; replication; replica creation strategy

### 1 引言

数据网格<sup>[1]</sup> (Data Grid) 是当前网格领域的研究热点,其目标是使地理上广泛分布的用户能够共享数据资源,以完成对大规模海量数据集的复杂分析和处理. 数据网格系统广泛采用数据复制技术以提高系统性能. 数据复制技术通过在数据网格系统内提供多个数据副本以缩短数据的访问延迟,实现系统容错及负载均衡等. 由于数据网格系统分布于广域网上并具有一定的动态性且数据量、用户量都很大,因此一般由数据网格系统在运行时刻动态创建数据副本<sup>[2]</sup>.

目前针对副本创建策略<sup>[3,4]</sup>的研究主要是在以欧洲数据网格为代表的网格系统环境中展开的,主要考虑数据网格的层次式拓扑结构、源数据存储于顶层节点、系统以具有固定高带宽的网络环境为基础、单个节点具有较强的存储能力等特点. 而很多面向企业应用的数据网格并不具有上述特点,该类数据网格通常以 Internet 为基础、具有非层次的 P2P 拓扑结构、源数据在网格节点间均匀分布、单个节点的存储能力较弱. 与欧洲数据网格相比,由于受到企业投资的资金限制,企

业数据网格的投入较少,基本上以既有的机器和网络资源为基础,故本文称这类数据网格为平民数据网格—Data Grid of Humble Birth (DGHB).

由于网络环境的差异,目前已有的副本创建策略不能适应 DGHB 网格的特点,在很大程度上影响了面向企业数据网格应用系统中的数据访问效率. 本文分析了数据网格系统中的数据复制性能并总结了当前副本创建策略的研究现状,针对 DGHB 网格环境,提出了一种基于存储联盟的双层动态副本创建策略—SADDRES. 通过在数据网格模拟工具 OptorSim 中的模拟,分析、比较了 SADDRES 策略与基于缓存的副本创建策略、基于经济模型的副本创建策略等多种策略的性能. 实验结果表明,SADDRES 策略在面向企业的数据网格应用中具有更好的系统性能.

### 2 数据网格系统的复制性能分析

数据网格系统中,当客户对某数据项发出访问请求时,复制系统从该数据项的多个数据副本中为客户选择响应时间最短的副本. 由于数据网格系统中操作粒度都很大,故数据网格

系统中客户访问数据时通常先将数据从源节点拷贝到本地节点,再在本地访问该数据<sup>[5]</sup>.某个数据项  $LFN$  对特定请求  $Req$  的响应时间  $Responsetime(Req, LFN)$  可用下面公式表示:

$$Responsetime(Req, LFN) = \min_{0 < i < N} Responsetime(Req, i) \quad (i \text{ 为整数})$$

其中  $N$  为当前数据网络系统中数据项  $LFN$  的副本数,  $Responsetime(Req, i)$  为副本  $i$  对  $Req$  的响应时间,而  $Responsetime(Req, i)$  可表示为:

$$Responsetime(Req, i) = \text{filetransfer.time}(Req, i) + \text{fileaccess.time}(Req, i)$$

其中  $\text{filetransfer.time}(Req, i)$  为将副本  $i$  从副本存储节点传输到请求节点的传输时间,  $\text{fileaccess.time}(Req, i)$  为访问副本  $i$  所花费的时间,包括传输前的等待时间、访问存储设备的时间以及传输到本地节点后访问存储设备所花费的时间等<sup>[6]</sup>.  $\text{filetransfer.time}(Req, i) = \text{size}(LFN) / \text{transfer.rate}$ ,  $\text{size}(LFN)$  为数据项  $LFN$  的大小,  $\text{transfer.rate}$  为当前网络数据传输率.因此,副本存储节点和请求节点间的网络数据传输率以及数据项的大小直接影响到单个副本的响应时间.而影响  $\text{fileaccess.time}(Req, i)$  的因素包括副本存储节点的负载、存储终端效率以及请求节点的存取效率等.广域网环境下,一般有  $\text{filetransfer.time}(Req, i) \gg \text{fileaccess.time}(Req, i)$ .

另一方面,客户的请求模式也直接影响复制系统的性能.客户对数据的请求可能具有多种访问局部性,包括时间局部性、地理局部性和空间局部性等<sup>[3]</sup>.一般地,客户访问数据的局部性越大,通过复制得到的好处越多.在一般的分布存储系统中,客户访问模式都有一定程度的局部性.影响复制性能的因素还包括单个节点的存储能力以及网格系统的拓扑结构.若节点的存储能力太弱,可能会引起数据的频繁复制和删除.而网格的拓扑结构直接影响到副本的创建和选择.

### 3 副本创建策略研究现状

在副本创建策略方面,目前比较有名的研究团体包括阿岗国家实验室、Ian Foster 领导的芝加哥大学分布系统实验室以及欧洲数据网格项目组.由于欧洲数据网格项目在数据网格领域有着广泛的影响,上述几个小组的研究工作主要以欧洲数据网格为研究背景.欧洲数据网格具有以下特点<sup>[3]</sup>:

♡拓扑结构:欧洲数据网格为 3 至 4 层的层次结构,最顶层只有一个节点,为系统的中央节点,其他层存在多个节点,客户节点分布在系统的最底层;

♡源数据分布:所有源数据存储于最顶层的中央节点;

♡网络:基于固定带宽的网络,网络带宽为 622Mbps;

♡节点存储能力:节点的存储能力强.

针对此类数据网格,比较典型的副本创建策略包括:最佳客户副本创建策略、瀑布副本创建策略、基于缓存的副本创建策略、缓存加瀑布副本创建策略、快速传播副本创建策略、基于经济模型的副本创建策略.文献[3]对上述策略进行了相应的模拟,结果表明在给定的测试环境下,上述副本创建策略在大部分情况下都能够缩小访问延迟并节省网络带宽.但瀑布副本创建策略、缓存加瀑布副本创建策略以及快速传播副本

创建策略是层次式拓扑结构,源数据存储于顶层节点的数据网格所独有的副本创建策略,无法应用于 DGHB 网格.而最佳客户副本创建策略、基于缓存的副本创建策略和基于经济模型的副本创建策略虽然可应用于 DGHB,但其并没有考虑 DGHB 网格的拓扑结构、数据分布、网络带宽以及节点存储能力等特点,故其性能较差.

## 4 基于存储联盟的双层动态副本创建策略-SADDRES

跟欧洲数据网格相比, DGHB 数据网格具有以下特点:

♡拓扑结构:具有非层次的 P2P 拓扑结构;

♡源数据分布:源数据均匀存储于不同的网格节点;

♡网络:通常基于 Internet,网络带宽得不到保证;

♡节点存储能力:由于通常基于既有的机器,因此单个节点的存储能力远远小于欧洲数据网格的单个节点存储能力.

为了有效提高 DGHB 网格的性能,本文提出了一种基于存储联盟的双层动态副本创建策略-SADDRES (Storage Alliance based Double-layer Dynamic Replica crEation Strategy),充分考虑了 DGHB 的特点,在 DGHB 数据网格环境下具有较高的复制效率.

### 4.1 存储联盟

DGHB 网格中,单个节点的存储能力有限,很容易导致数据的频繁复制与删除,在很大程度上影响到复制的效率.另一方面,Internet 上的数据传输率分布极为不均匀,地理上较近的节点间网络数据传输率一般高于地理上较远的节点间网络数据传输率.

针对 DGHB 的上述特点,本文提出了“存储联盟”的概念.集合  $SA$  为网格系统  $GS$  节点集合的子集,且  $SA \neq \emptyset$ ,若  $SA$  满足以下条件:

♡任意节点  $n_i, n_j \in SA$ ,若  $n_i, n_j$  都有  $\text{transfer.rate}(n_i, n_j) > R_v$  ( $R_v$  为网格系统预先设定的网络数据传输率阈值);

♡若网格节点  $n_i \in GS$ ,对于任意的节点  $n_j \in SA$  都有  $n_i, n_j$  间的网络数据传输率  $\text{transfer.rate}(n_i, n_j) > R_v$ ,则  $n_i \in SA$ ; 则称  $SA$  为网格系统  $GS$  的一个存储联盟.

一个 DGHB 网格系统可以划分为多个存储联盟,存储联盟可由网格系统管理员事先设定,也可以在运行时刻根据网络状态动态形成.在划分存储联盟时,应选择适当的网络数据传输率阈值,而阈值的取值决定了存储联盟的大小.属于同一个存储联盟的网格节点间的网络数据传输率一般大于属于不同存储联盟的网格节点间网络数据传输率.设存储联盟  $SA_1$  中任一节点  $n$ ,发出对数据项  $LFN$  的访问请求  $Req, R_1$  和  $R_2$  为  $LFN$  在网格系统中的两个副本,其中  $R_1$  存储于节点  $x, x \in SA_1, R_2$  存储于节点  $y, y \in SA_1$ ,在节点  $x$  和  $y$  负载以及存储设备延迟一样时,在一般的情况下有  $Responsetime(Req, R_1) < Responsetime(Req, R_2)$ .

### 4.2 SADDRES 副本创建策略

SADDRES 副本创建策略以存储联盟概念为基础,提出了基于存储联盟的双层动态副本创建策略.其中第一层 SABC (Storage Alliance Based Cache) 算法主要考虑存储联盟之间的副

本复制. 第二层 ROSA (Replica Optimization in Storage Alliance) 算法主要考虑存储联盟内部数据的合理分布.

**4.2.1 SABC 算法描述** SABC 算法的基本思想类似于基于缓存的副本创建策略, 不同的是 SABC 算法将网格系统的每个存储联盟看作一个整体. 如果某个节点所请求的数据项在节点所属的存储联盟内存在副本, 且请求节点上不存在空闲空间则不作任何复制. 如果存储联盟内不存在请求数据项的副本, 则在该存储联盟内创建该数据项的副本. SABC 算法包括 SABC 主算法和 MakeSpace 算法, 其中 MakeSpace 算法在存储联盟没有足够的空闲空间时删除存储联盟内的某些数据以便创建新的数据项副本. 由于 MakeSpace 算法首先删除在存储联盟内存在多个副本的数据项副本, 因此系统运行稳定后, 每个数据项在一个存储联盟内部一般只有一个副本. SABC 算法和 MakeSpace 算法分别如图 1、2 所示.

```

算法执行前提: SAA 内节点 n 发出访问数据项 LFN 的请求
if (节点 n 上存在 LFN 的副本)
    exit
if 节点 n 上的空闲空间 Size(LFN) // Size(LFN) 为数据项的大小
{
    在节点 n 上创建 LFN 的副本;
    exit
}
if (SAA 中存在 LFN 的副本)
    exit;
K = MakeSpace(SAA, size(LFN)) // 在存储联盟 SAA 内腾出足够的空间
if (K == null)
{
    抛出无法缓冲异常 // 该存储联盟找不到足够的存储空间
}
    在节点 K 上创建 LFN 的副本;
  
```

图 1 SABC 算法

```

MakeSpace 算法:
Input: SA, size: // SA 存储联盟标记, size 需要的空闲空间大小
Output: 网络节点; // 该网络节点具有 size 的空闲空间
if (SA 中存在节点 X, X 上的空闲空间 size)
{
    return X;
}
// 如果存储联盟内部不存在节点具有足够的空闲空间, 则需要删除的某些数据项集合 File R 为在存储联盟 SA 内存在多个副本的数据项集合
if (File R != null)
{
    选择 File R 中大小 size 的集合, 设为 FileList;
    if (FileList != null)
    {
        选择 FileList 所有数据项中被访问次数最少的副本设为 R;
        设 R 存储在节点 D 上;
        删除节点 D 上的副本 R;
        return D;
    }
}
  
```

```

选择 File R 中最大的数据项, 设为 LFND;
选择 LFND 在 SAA 中被访问次数最少的副本, 设其存储在节点 K 上;
设 File K 为 File R 中在节点 K 上存在副本的集合, 并按照节点 K 上副本的访问次数成降序排列;
for (i = 0; i < File K.Length; i++)
{
    删除 File K[i] 在节点 K 上的副本;
    if (节点 K 上的空闲空间 size)
        return K;
};
do
{
    删除节点 K 上访问次数最少的数据;
} until (节点 K 上的空闲空间 size)
return K;
}
根据存储联盟 SAA 的访问记录, 选择被访问次数最少的副本, 设该副本存储在节点 E 上;
do
    删除节点 E 上访问次数最少的副本;
} until (节点 E 上的空闲空间 size)
Return E;
  
```

图 2 MakeSpace 算法

**4.2.2 ROSA 算法描述** 在同一个存储联盟内部, 可能因为数据分布的不合理导致数据在存储联盟内部频繁拷贝, 而存储联盟内过多的数据拷贝同样会在一定程度上影响整个数据网络系统的响应时间.

跟整个数据网络系统相比, 一个存储联盟内的节点数目有限, 一般为几个到几十个, 因此在一定的时间段内可将存储联盟视为一个封闭系统. 设存储联盟 SA 存在 N 个节点, 共存储了 L 个数据项. 对于数据项  $i (0 \leq i < L)$ , 若该数据项有副本存储在节点 k 上, 则  $\text{location}(k, i) = 1$ , 否则  $\text{location}(k, i) = 0$ . 由 SABC 算法可知在系统运行稳定后, 每个数据项在一个存储联盟内部一般只有一个副本. 假设在一定时间段内节点 k ( $0 \leq k < N$ ) 对副本 i 的请求数为  $\text{Request}(k, i)$ , 当前存储联盟 SA 内部网络数据传输率为  $\text{SABandwith}$ , 则在该段时间内, 存储联盟内部用于数据传输的时间为:

$$\text{Transfertime}_{SA} = \sum_{i=0}^{L-1} \sum_{k=0}^{N-1} \frac{\text{Request}(k, i) * (1 - \text{location}(k, i))^*}{\text{size}(i) / \text{SABandwith}}$$

在存储联盟内部, 副本优化的目标是通过副本在存储联盟内的合理分布使得  $\text{Transfertime}_{SA}$  最小. ROSA 算法的核心思想是根据一段时间内存储联盟的客户访问记录, 合理分布对响应时间影响较大的少量数据项. 在 ROSA 算法中, 只关注在一定时间段内被访问次数超过特定阈值的数据项, 而阈值的具体值与网络的拥塞状况以及用户的访问模式等有很大关系, 可在运行时刻动态调整. ROSA 算法如图 3 所示.

```

算法执行前提: 在一定的时间段 T 内定期执行该算法
选择存储联盟内所有副本 R, 满足 RequestR > Requestvalue, 并将数据
  
```

项按照  $Request_R$  成降序排为序列  $List$ ; //  $Request_R$  为副本  $R$  在一定时间段内被存储联盟内部节点访问的次数,  $Request_{val}$  为访问次数阈值

```

for (i = 0; i < List.length; i++)
{
    if (Best(List[i]) == Node(List[i])) // 其中 Node(List[i]) 为数据项 List[i] 在存储联盟内的副本存储节点, Best(List[i]) 为该时间段内存储联盟内访问数据项 List[i] 次数最多的节点
        continue; // 副本正好存储在最佳节点上, 无需任何调整
    if (节点 Best(List[i]) 上的空闲空间 < size(List[i]))
    {
        在节点 Best(List[i]) 上创建 List[i] 的副本;
        删除 Node(List[i]) 上 List[i] 的副本;
        continue;
    }
    do
    {
        选择 Best(List[i]) 上该节点最少访问的副本设为 Least
         $K = MakeSpace(SA_i, size(Least))$ ;
        IF ( $K' = Null$ )
        {
            在节点  $K$  上创建 Least 的副本;
            删除节点 Best(List[i]) 上 Least 的副本;
        }
    } until (节点 Best(List[i]) 上的空闲空间 < size(List[i]));
    在节点 Best(List[i]) 上创建 List[i] 的副本;
    删除 Node(List[i]) 上 List[i] 的副本;
}

```

图 3 ROSA 算法

## 5 性能模拟

SADDRES 策略充分考虑了 DGHB 网络的特点. 在 DGHB 网络应用中, SADDRES 策略跟其他副本创建策略相比具有以下优势:

♥ SADDRES 策略提出了存储联盟的概念, 充分考虑了 Internet 网络的特点以及客户访问的时间局部性、地理局部性;

♥ SADDRES 策略在单个节点存储能力较弱时, 能够在很大程度上避免数据的频繁替换;

♥ 在同一个存储联盟内部, SADDRES 策略能够在一定程度上实现数据的合理分布.

本文在欧洲数据网格小组开发的数据网格模拟工具 OptorSim 上模拟实现了 SADDRES 策略. OptorSim 模拟工具中已经实现了多个副本创建策略, 包括基于缓存的副本创建策略以及基于经济模型的副本创建策略等. 我们在 OptorSim 工具上模拟实现 SADDRES 策略时, 为了实现上的简单同时根据本实验的具体情况将 SADDRES 策略中 ROSA 算法的  $Request_{val}$  值设置为 50.

### 5.1 OptorSim 模拟工具的配置

在 OptorSim 工具中, 用户通过配置文件描述网格系统的拓扑结构、节点的存储能力、网络带宽、作业以及作业的调度方式等<sup>[7]</sup>. 当作业按既定的方式分配到不同的网格节点后, 每

个网格节点按照配置好的方式调度作业运行. 每个作业需要访问一系列文件, 在访问文件的过程中执行相应的副本创建策略. OptorSim 的主要输出参数为网格系统所有节点的工作时间之和, 而总工作时间能够直接反映网格系统的性能.

我们的模拟环境

采用如图 4 所示的网格拓扑结构. 网格系统包括 26 个网格节点, 网格节点基于全连接的 P2P 拓扑结构. 根据 SADDRES 策略的需要, 事先将网格节点按照网络数据传输率分布划分为 6 个存储联盟, 分别为 SA0, SA1, SA2, SA3, SA4, SA5. 网格系统的每个文件大小为

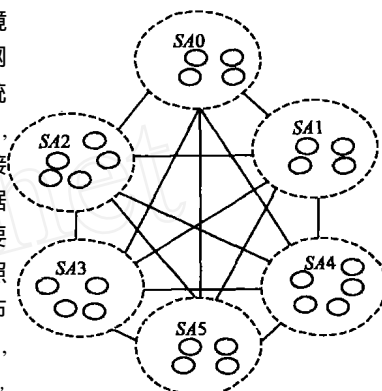


图 4 模拟网络的拓扑结构

1 G, 源数据文件均匀分配到不同的网格节点上.

### 5.2 模拟结果及分析

基于上述模拟环境, 我们模拟、比较并分析了作业数量、存储联盟内部数据传输率以及网格节点的存储能力等因素对采用 SADDRES 策略、基于经济模型的副本创建策略以及基于缓存的副本创建策略的数据网格系统的性能影响.

5.2.1 作业数量对不同策略的影响 在存储联盟内部节点间的网络数据传输率为 100Mbps, 存储联盟间的数据传输率为 10Mbps, 单个节点的存储能力为 100G 时, 数据网格系统分别采用 SADDRES 策略、基于缓存的副本创建策略以及基于经济模型的副本创建策略时完成作业的总工作时间与作业数量的关系如图 5 示.

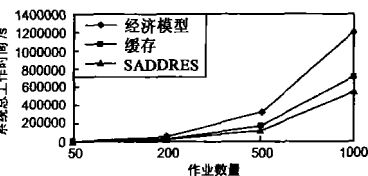


图 5 作业数量与不同策略的系统总工作时间关系图

由图 5 可以看出, 随着作业数量的增大, 采用三种不同策略时数据网格系统的总工作时间都快速增加. 在相同的工作量时, 采用 SADDRES 策略时网格系统的总工作时间最短, 采用基于缓存的副本创建策略时次之, 采用基于经济模型的副本创建策略时网格系统的总工作时间最长. 同时, 随着作业量的增加, 采用 SADDRES 策略时的总工作时间的增长速度明显低于采用其他两种策略时系统总工作时间的增长速度, 故 SADDRES 具有更好的可扩展性.

5.2.2 存储联盟内部数据传输率对不同策略的影响 在作业量为 200, 存储联盟间数据传输率为 10Mbps, 单个节点的存储能力为 100G 时, 网格系统采用 SADDRES 策略、基于缓存的副本创建策略以及基于经济模型的副本创建策略时系统总工作时间与存储联盟内部数据传输率的关系如图 6 所示.

由图 6 可以看出, 随着存储联盟内部网络数据传输率的增大, 采用三种不同副本创建策略时系统的总工作时间都有

不同程度的降低.但采用 SADDRES 策略时系统的总工作时间降低速度明显高于采用其他两种策略时系统总工作时间的降低速度.

### 5.2.3 网格节点存储能力对不同策略的影响

在作业量为 200,存储联盟间网络数据传输率为 10Mbps,存储联盟内数据传输率为 100Mbps 时,采用三种不同策略时系统的总工作时间与节点存储能力的关系如图 7 所示.

由图 7 可以看出,随着节点存储能力的增强,采用三种不同策略时系统的总工作时间都有不同程度的降低.但采用 SADDRES 策略时的总工作时间降低速度低于采用其他两种策略时系统的总工作时间的降低速度,甚至在节点存储能力为 1000G 时,采用 SADDRES 策略时系统的总工作时间略高于采用缓存策略时的总工作时间.因此,SADDRES 策略更适用于单个节点存储能力较弱的的数据网格系统.

上述模拟实验表明,SADDRES 策略比基于缓存的副本创建策略和基于经济模型的副本创建策略具有更好的可扩展性,在基于 Internet 网络、具有 P2P 拓扑结构、源数据均匀分布于不同网格节点且单个网格节点存储能力较弱的数据网格环境下具有更好的系统性能.

## 6 结束语

随着 Internet 环境下企业级应用的不断拓展,面向企业的数据网格应用正成为业界关注的焦点.本文通过分析面向企业数据网格应用环境的特点,提出了平民网格 DGHB 的新概念;基于对数据网格系统中数据复制性能的分析和对目前副

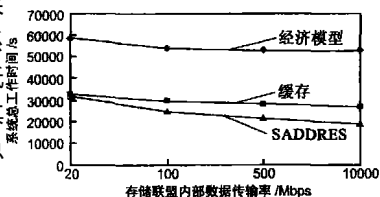


图 6 存储联盟内数据传输率与不同策略的总工作时间关系图

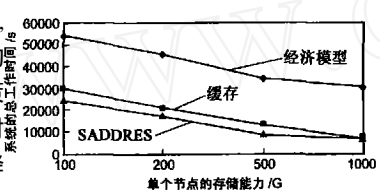


图 7 单个节点的存储能力与不同策略的系统总工作时间图

本创建策略研究现状的总结,提出了一种适合面向企业数据网格应用的、基于存储联盟的双层动态副本创建策略 SADDRES;通过在欧洲数据网格研究小组开发的数据网格模拟工具 OptorSim 上的性能评测实验表明,与针对欧洲数据网格环境研究的基于缓存的副本创建策略和基于经济模型的副本创建策略相比,SADDRES 策略更加适应面向企业的网络应用环境.

### 参考文献:

- [1] Ian Foster, Carl Kesselman. The Grid: Blueprint for a New Computing Infrastructure[M]. Morgan Kaufmann Publishers, July 1998.
- [2] Heinz Stockinger. Database Replication in World-wide Distributed Data Grids[D]. University of Vienna, 2001.
- [3] Kavitha Ranganathan, Ian Foster. Design and Evaluation of Dynamic Replication Strategies for a High Performance Data Grid[A]. International Conference on Computing in High Energy and Nuclear Physics Proceedings[C]. 2001. 106 - 118.
- [4] M Carman, F Zini, L Serafini, K Stockinger. Towards an economy-based optimisation of file access and replication on a data grid [A]. 2nd IEEE ACM International Symposium on Cluster Computing and the Grid (CCGRID '02) [C]. Berlin: IEEE CS Press, 2002. 240 - 345.
- [5] Allcock B, et al. Secure, Efficient Data Transport and Replica Management for High Performance Data-Intensive Computing[A]. IEEE Mass Storage Conference Proceedings[C]. San Diego, California, 2001.
- [6] Kurt Stockinger, et al. Access cost estimation for unified grid storage systems[A]. Fourth International workshop on Grid Computing Proceedings[C]. Phoenix, Arizona, 2003.
- [7] William H, et al. OptorSim: A grid simulator for studying dynamic data replication strategies [J]. International Journal of High Performance Computing Applications, 2003, 17(4): 403 - 416.

### 作者简介:

孙海燕 女,1976 年出生于山东省莱阳市,博士研究生,主要研究领域为分布计算、网格技术等. E-mail: shyfirst@hotmail.com.

王晓东 男,1973 年出生于湖南省长沙市,博士,副研究员,主要研究领域为协同工作、移动计算等.