移目的和正常元数据请求被赋予一定的优先级。并发控制算法考虑元数据请求的语义，比较请求之间的优先级，保证目录元数据迁移的有效性。由于并发情况较多，在描述目录的并发控制算法基础上，本章仅采用 Petri Net 描述和验证了目录迁移请求与删除目录子文件请求的并发控制的活跃性，其他情况同样可以进行验证。

总之，BWMMS 通过比较请求之间的优先级，结合元数据请求语义，根据优先级判断请求能否被抢断，从整体上提高了系统的请求并行度，增强了系统的处理效率。

第七章 元数据服务扩展能力评估

结合元数据存储服务和元数据请求服务关键问题的解决，BWMMS 得到原型实现，并运用到 BWFS 中。本章将介绍原型系统实现中的关键问题，然后评估系统对传统应用和新兴应用的扩展支持能力。结果表明，尽管原型系统在实现上存在一些不足，BWMMS 采用的动态元数据服务机制和策略，既能够很好支持传统应用，也能够很好支持新兴应用，为 BWMMS 未来进一步发展提供了佐证。

7.1 系统原型实现

在系统服务器的物理构成上，原型系统包括若干应用服务器（AS），若干元数据服务器（MS），1 个绑定服务器（BS）和若干个存储设备（SN）。服务器的软件模块及其交互流程如图 7.1 所示。

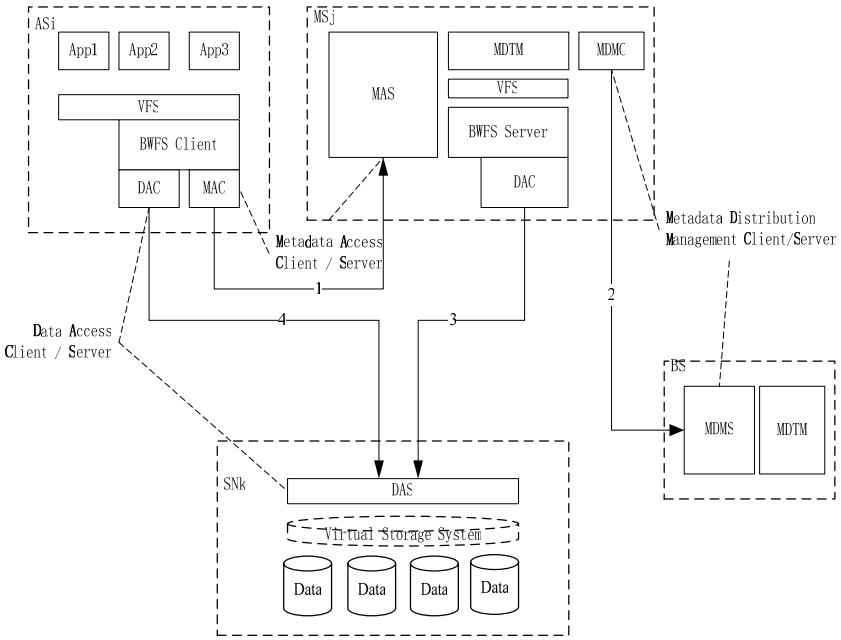

图 7.1 BWMMS 原型系统结构

AS 和 MS 使用数据访问客户端模块（Data Access Client, DAC）分别从存储设备读取文件的数据和元数据，数据访问服务器模块（Data Access Server, DAS）是存储设备上负责处理数据和元数据请求的服务。AS 通过元数据访问客户端模块（Metadata Access Client, MAC）与 MS 上的元数据访问服务器模块（ Metadata Access Server, MAS）交互，获取文件元数据信息。元数据分布信息管理客户端(MDMC)和元数据分布信息管理服务器(MDMS)是 MS 和 BS 上用来完成元数据分布信息管理的软件模块。

对于涉及多个元数据的元数据请求，其处理过程如下：

1. 查询文件的元数据请求发送给父目录的宿主 MS。如果被查询文件当前分布在其他 MS，用户从被查询文件的宿主 MS 读取该文件的元数据信息。
2. 创建文件的元数据请求发送到父目录的宿主 MS。该元数据服务器按照文件创建语义完成请求的处理后，再向 BS 查询新创建文件的宿主 MS。根据 BS 返回结果，如果本 MS 不是新创建文件的宿主 MS，则需要将创建操作涉及到的文件（父目录和新创建文件）写回到 SN，并返回请求结果给 AS。
3. 删除文件的元数据请求发送到父目录的宿主 MS。根据需要，将请求涉及的元数据通过元数据迁移协议集中到该服务器。单个 MS 按照请求要求的语义完成请求处理。
4. 移动文件的元数据请求发送到文件移动请求的源父目录的宿主 MS，由它负责集中文件移动请求需要的元数据，并按照文件移动语义完成请求处理。为避免多个文件移动请求导致文件系统目录树结构的不一致，系统在同一时间只处理一个文件移动请求。所以，文件移动请求的处理，首先需要从 BS 申请文件移动请求处理权限，其他的文件移动请求必须在本请求完成后才能进行。

为避免复杂的系统交互过程，对于动态迁移协议，系统实现没有支持“选择完成元数据请求处理的服务器”，元数据请求由收到该请求的服务器负责处理。原型系统也没有实现“被迁移的目录重新迁到源 MS”的协议。因为 BWMMS 根据元数据活跃性管理元数据请求的分布，被迁移的元数据是否有必要迁到原来的 MS，需要在今后的工作中，根据实际应用的数据进行进一步的分析。

BWFS 的客户端模块是内核模块，没有服务进程数量的限制。MS 的 MDS 以内核线程方式存在，可以指定内核线程的数量，默认值为 8 个。为了实现的简单和减少进程调度的开销，BS 实现为基于 RPC 通信的单个服务进程结构的后台程序。元数据分布决策算法的服务器负载以各个 MS 当前分布的活跃元数据数目为参数。

在后续章节中，我们将通过实验，评估系统在传统的创建/删除文件、用户访问共享目录测试文件集和用户访问私有目录测试文件集等情况的聚合 I/O 性能的扩展情况，也将评估对新兴的生物计算应用的支持能力。所有测试均在同一测试平台进行。测试使用的服务器是 Intel® Xeon® 3.4GHz,3.0GB 的内存，运行 RedHat® Linux® 8.0。服务器通过千兆以太网络互联。文件系统的构成包括 1 个存储设备，若干元数据服务器和 1 个绑定服务器。存储设备采用 120GB 的 SATA 硬盘提供数据和元数据的存储。

7.2 元数据服务扩展能力评价指标

根据 1.1 节定义的元数据服务扩展能力，本部分元数据服务扩展能力的评估针对两种情况的系统进行：1）服务器数目固定，变化客户端的元数据请求负载，获得聚合元数据吞吐率；2）客户端负载固定，变化服务器数目，获得聚合元数据吞吐率；

假定每个客户端的元数据请求吞吐率为$Throughput(i)$，则系统聚合元数据吞吐率为：$\sum_{i=1}^{n} Throughput(i)$。为衡量系统的扩展能力，系统规模变化带来的加速比计算公式为：$\sum_{i=1}^{n} Throughput(i) / \sum_{i=1}^{1} Throughput(i)$，即 n 个评测对象系统规模的聚合元数据吞吐率与一个评测对象的元数据吞吐率的比值。

系统扩展能力的评估将不仅针对传统的 Web Service、E-Mail 等，也将针对逐渐占据重要地位的生物信息计算等新兴应用。

第 7.3 节通过能够模拟传统应用访问特征的 benchmark，测试系统在不同数据共享模式下的扩展能力。测试结果表明，BWMMS 能够提供具有较好扩展能力的元数据服务。同时，它还表明，由于原型系统实现尚存的不足，跨服务器请求比例对系统扩展能力的影响还较为明显，需要优化系统的实现。

第 7.4 节评估系统对新兴的生物信息计算应用的支持能力。评估采用生物信息计算常用的 BLAST（Basic Local Alignment Search Tool）完成。BLAST 首先将库文件初始化，然后在库文件中进行查找，明确输入序列的相似性。评估结果表明，随着服务器数目增加，元数据服务时间减少速度在 20%左右，为 BLAST 提供具有较好扩展能力的分布式文件系统元数据服务。

7.3 对传统应用的扩展支持评估

为了充分评估系统对传统应用扩展的支持能力，本节将针对共享和私有两种访问模式，评测元数据服务的扩展能力。同时，还将通过频繁的文件创建/删除，明确系统可能存在的不足。

7.3.1 共享使用模式的扩展能力评估

只读共享是用户间常见的数据共享方式。本节通过固定客户端负载，增加服务器数目，评估服务器数目增加对用户访问速度、BS 负载的影响，验证元数据请求分布策略对服务器规模的扩展支持。其期望值是，服务器数目的增加，能够将元数据请求分布到多个服务器，并获得聚合元数据吞吐率的提升。

测试首先在共享目录下生成测试文件集，然后将系统重新启动，消除系统元数据缓存和元数据分布信息缓存的影响。

7 个客户端同时使用“find . -exec stat {} \;”命令访问共享目录下的文件。测试所用文件集是 Redhat® Linux® 8.0 的 Linux 内核代码 Linux-2.4.18-14，其目录和文件总数为 7,246 个。元数据服务器数目分别是 1、2、4 和 6 个。测试使用 Linux®的 time 工具收集

“find . -exec stat {} \;”所用时间。系统聚合吞吐率如表 7.1 所示。

表 7.1 共享模式聚合吞吐率随服务器变化

服务器数目	1	2	4	6
聚合吞吐率(ops)	178.44	179.73	180.05	180.63
吞吐率加速比	1	1.0072	1.0090	1.0123

从表 7.1 的聚合吞吐率随服务器数目的变化、聚合吞吐率加速比看，元数据服务的扩展能力不太明显。表 7.2 的服务器负载和图 7.2 的 BS 元数据分布决策进程的 CPU 占用率表明，由于测试环境规模的限制，客户端的元数据请求没有给系统提供足够的压力，导致聚合元数据吞吐率没有表现出随服务器数目的显著提升。

从表 7.2 的各个服务器的负载情况来看，由于 BWMMS 的元数据分布策略有效地将元数据请求分布到多个元数据服务器，各个服务器能够并行高效地处理元数据请求，元数据服务器的聚合服务时间呈现显著降低的趋势。

表 7.2 共享模式的服务器负载

服务器数目	1	2	4	6
MS 聚合服务时间(s)	0.61	0.60	0.35	0.32
分布信息缓存命中率(%)	97.67	96.85	96.56	96.70

在共享访问模式中，元数据分布决策将由多个客户端的访问驱动。当多个用户请求访问同一个元数据时，某个客户的请求驱动元数据分布决策，其他请求可以利用该请求获得的分布信息，完成元数据请求的处理。所以，元数据分布信息缓存发挥出极大的作用，如表 7.2 所示，元数据缓存命中率在 96.5%以上，大大缓解 BS 的压力。从图 7.2 的绑定服务器的元数据分布决策进程的 CPU 占用率来看，在元数据分布信息缓存的作用下，元数据服务器数目的增加不会显著提升后端绑定服务器的压力。

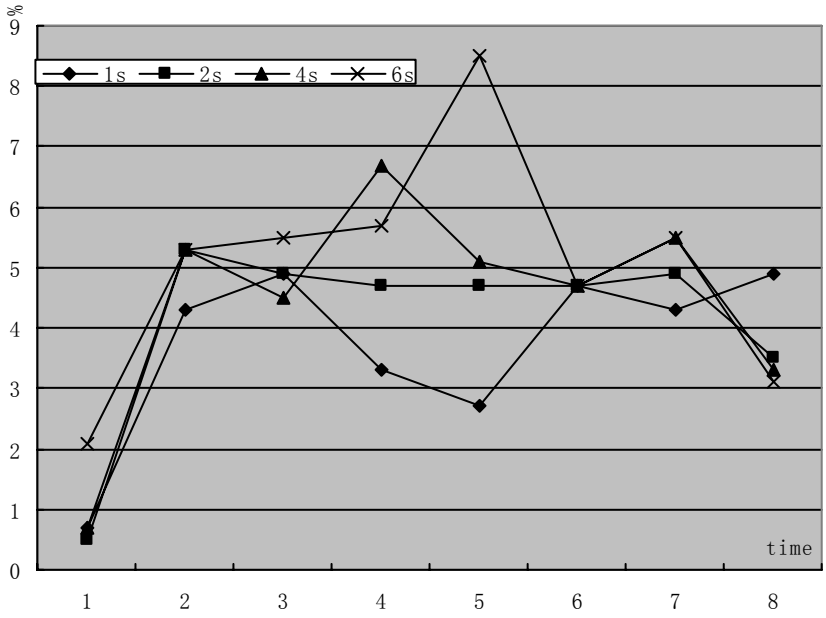

图 7.2 共享模式 BS 的 CPU 占用率