文件系统大部分的文件和数据不会被经常重复地访问，活跃文件和数据占整个系统的比例较少。据统计结果，从文件数目角度统计，活跃文件的比例大致为 4%-20%。从活跃的数据比例看，活跃数据的比例大致为 10%-30%[Ramakrishnan1992][Williams2005]。

3. 单个用户的元数据请求表现出局部性、突发性和动态变化的特征。

在一定时间内，单个用户的文件访问往往集中在某些目录下[Floyd1989][Agrawal2007]，不会发散到整个文件系统。单个用户的文件访问表现出随时间变化的突发性和动态性[Baker1991][Roselli2000][Vogels1999][Ellard2003]。

4. 不同的元数据请求的统计频率不同。

元数据请求存在不同的统计频率[Bozman1991][Vogels1999][Hsu2001][Ellard2004]，针对文件的元数据请求和文件查找的请求所占比例超过 95%，而需要更改多个元数据的文件创建、删除和移动等元数据请求所占比例不足 5%。

5. 元数据间在访问上存在关联性。

根据元数据访问协议，查找文件、创建文件、删除文件和移动文件等元数据请求都需要同时访问多个元数据。为保证系统元数据请求的聚合吞吐率，元数据服务器的负载平衡应该以保证访问统计频率较高的元数据请求的处理效率为前提。

所以，元数据请求分布策略应该综合考虑用户元数据访问的局部性、动态性和访问相关性，结合元数据访问协议，根据“最频繁的请求最快完成”原则，主要关注针对单个文件的元数据请求、查找文件的元数据请求的处理效率，并为其他元数据请求的正确处理提供必要的支持。

## 3.3 相关研究概况

现有的分布式文件系统元数据请求分布管理主要可以分为两大类：第一类是目录子树分区法，如 LOCUS[Popek1986]、AFS[Morris1986]、CODA[Satyanarayanan1990]、Sprite[Ousterhout1988]等。它依据文件系统名字空间结构，将文件系统目录树划分成目录子树，单个元数据服务器处理一个目录子树的所有元数据请求，文件的创建、删除和移动等元数据请求只能在同一目录子树的范围内进行。当各个目录子树的元数据请求差异很大时，元数据服务器负载将很难平衡。为尽可能地平衡服务器的负载，解决元数据服务随文件系统目录子树深度扩展的问题，动态目录子树分区法[Weil2004]动态收集元数据服务器的负载。当元数据服务器负载超过预定值时，将原来由该元数据服务器管理的目录子树，迁移到其他元数据服务器管理。由于以粒度非常大的目录子树为请求分布单位，其迁移过程对系统的影响较大。

第二类是哈希法。哈希法以文件名等静态的文件系统信息为哈希函数的参数，根据哈希函数计算结果决定负责元数据请求处理的元数据服务器。这类系统包括 Vesta[Corbett1996]、InterMezzo[Braam1999]、RAMA[Miller1997]、Lustre[Braam2002]和

Dynamic Hashing[Li2006]等。仅仅通过哈希函数的计算，不需要根据文件系统目录树遍历，哈希法就能够定位负责元数据请求处理的服务器，可以提高文件查找等请求的处理效率。但是，哈希法存在明显不足，包括：（1）静态设计的哈希函数很难支持系统的元数据服务器规模的动态扩展；（2)哈希函数的参数选择对元数据服务器负载的影响很大，设计不好的哈希函数可能将元数据请求集中到部分元数据服务器；（3）哈希法较难对文件访问的逻辑性提供支持。统计结果表明，单个用户在一定时间内的访问经常集中在几个目录。但是，哈希法可能将父目录和子文件分布到不同的服务器，加大分布式元数据请求的可能性；（4）当哈希参数发生变化，需要重新分布元数据请求时，需要大量的元数据请求分布信息更新操作和元数据迁移操作来适应新的元数据分布结果。

总之，目录子树分区法和哈希法主要以文件系统的静态信息为参数，完成元数据请求分布决策。它们有自己的优点，但无法很好地支持用户元数据请求的动态变化。

## 3.4 BWMMS 元数据请求分布管理

根据文件系统元数据的部分活跃和单个应用元数据访问的局部性特征，BWMMS 仅管理应用当前访问的活跃元数据的请求分布。同时，BWMMS 区分对待目录和常规文件的请求分布，在考虑相关性的同时，结合服务器的负载情况，动态决定元数据请求的分布。结合元数据请求的访问频率，保证频率高的请求尽可能快地完成。

本节首先定义后续论述中需要的相关概念，然后讨论请求分布管理的机制、协议和分布决策算法。

### 3.4.1 相关概念定义

元数据请求分布以文件索引节点为最小单位进行。当负责请求处理的元数据服务器确定后，与文件索引节点紧密相关的元数据也由该服务器负责读写。

在定义元数据请求分布管理的重要概念前，首先需要明确与管理相关的对象。

- 文件系统信息集合。集合 B 表示文件系统当前所有信息的集合，即

$$B = \{b \mid b\text{是文件系统的信息}\},$$

- 文件数据信息集合。集合 D 是普通文件的数据，则

$$D = \{d \mid d \in B, d\text{是普通文件的数据}\},$$

- 元数据信息集合。集合 M 是文件系统的元数据集合，则

$$M = \{m \in B \wedge m \notin D\},$$

集合 B，集合 D 和集合 M 间的存在的关系为：

$$M \cup D = B, M \cap D = \varnothing。$$

- 工作元数据服务器集合。集合 S 为系统当前所有提供服务的元数据服务器，即

$$S = \{s \mid s\text{是当前提供元数据服务的元数据服务器}\}\text{。}$$

根据以上集合的定义，我们将定义活跃元数据、元数据请求分布映射、元数据宿主和单一映射等概念。

定义 3.1 活跃元数据。活跃元数据（Active Metadata, AM）是文件系统当前被访问的元数据集合。任何当前被访问到的文件索引节点和其关联的元数据信息、该索引节点文件名路径前缀上的所有目录索引节点和其关联的元数据都是活跃元数据。

$$AM = \{am \mid am \in M \ \wedge \ am\text{正在被访问}\}$$

定义 3.2 元数据请求分布映射。为活跃元数据指定负责元数据请求处理的元数据服务器的过程，称为“元数据分布映射”。

$$F = \{f \mid f =< am, s >,\ am \in AM,\ s \in S\}$$

请求分布映射是活跃元数据和元数据服务器间存在的一种动态关系，根据元数据的活跃性动态建立。元数据第一次被访问而变成活跃时，将完成请求分布映射关系的建立。在元数据不活跃时，请求分布映射关系将被解除。

定义 3.3 元数据宿主。当前负责元数据请求处理的元数据服务器称为“元数据的宿主”，表示为 HOST(am)。

$$s \ = \ f(am),\ am \in AM, s \in S$$

定义 3.4 活跃元数据的单一映射。在同一时间，任意活跃元数据有且仅有一个元数据宿主。其结果是活跃元数据在元数据集群中仅有一份拷贝，可以简化元数据更新的同步问题。

$$\forall s1, s2 \in S,\ am \in AM,\ s1 \ = \ f(am),\ s2 \ = \ f(am) \ \Rightarrow s1 \ = \ s2$$

定义 3.5 服务器负载。服务器负载（WorkLoad, WL）是通过多个因素以一定的权值比例构成的，综合反映服务器当前的压力。

$$WL = \sum_{i=1}^{n} W_i P_i,\ \text{其中} \sum_{i=1}^{n} W_i = 1$$

### 3.4.2 分布管理机制

元数据请求分布管理机制主要包括分布式决策和集中式决策两大类[Braam2002][Anderson2000-1][Brandt2003]，如图 3.1 所示。

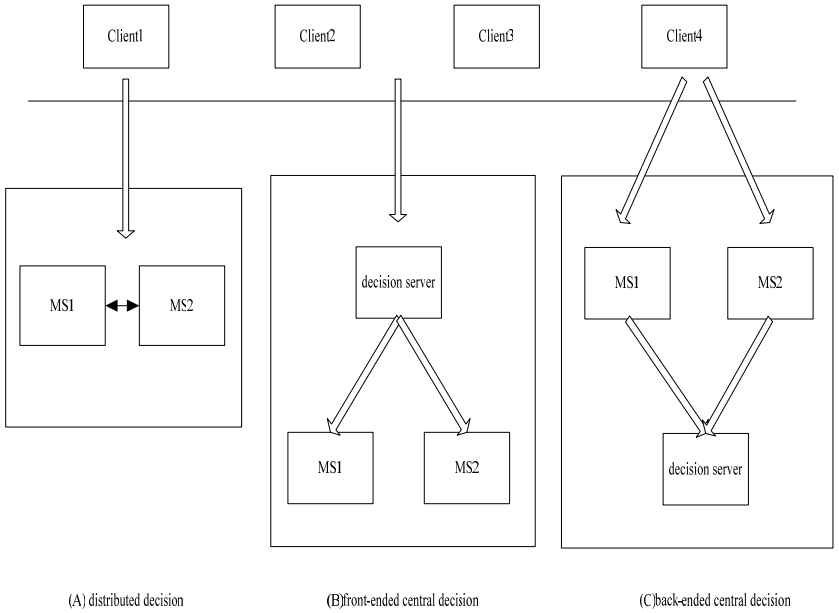


图 3.1 三种元数据请求分布管理机制

分布式决策系统的各个元数据服务器具有很大程度的自治权限。在服务器负载没有超过预先设定值时，各个服务器独立管理元数据请求的分布。当服务器负载超过设定值后，它需要与系统所有的服务器通信，协同完成元数据请求的分布决策。从结构上讲，分布式决策系统不存在单一服务器限制系统扩展的可能。但是，其采用的某些集中控制机制同样可能限制系统的扩展能力。比如，请求的分布决策要求服务器维护大量的决策信息，并公布分布决策结果。并且，由于缺乏统一的调度，服务器间的负载很难平衡。

与分布式决策不同的是，集中式决策采用专门的服务器，集中决策整个系统的元数据请求分布。集中式决策系统的各个元数据服务器为决策服务器提供必要的决策信息，由决策服务器集中决定元数据请求的分布。集中决策可以减轻各个服务器的分布决策开销，并能够适应服务器负载的变化，以统一简洁的方式完成请求的分布决策。各个服务器根据需要，向决策服务器查询请求分布现状，每次分布决策结果不需要广播。

按照决策服务器所处的位置，集中式决策系统可以分为“前端集中决策”和“后端集中决策”。由于元数据访问的局部性特征，后端集中决策将通过在元数据服务器缓存元数据分布结果，将必需决策服务器完成的处理变成可选，大幅度降低决策服务器的负载，提高系统的扩展能力。

BWMMS 基于“集中决策，分布处理”的原则，采用后端集中决策机制完成文件系统元数据请求的分布决策，系统的集中决策点位于元数据请求处理路径的末端。元数据服务器彼此间不需要为完成元数据请求分布决策通信，他们仅需要与集中决策服务器交互。在大多数的情况下，元数据服务器可以根据自己缓存的元数据分布信息，完成请求的处理。在必要的时候，它与集中决策服务器交互，要求决策元数据请求的分布，并根

据分布结果填充元数据分布信息缓存。通过多级的元数据分布信息缓存，BWMMS 形成层次化的元数据分布信息管理机制。其结构如图 3.2 所示。

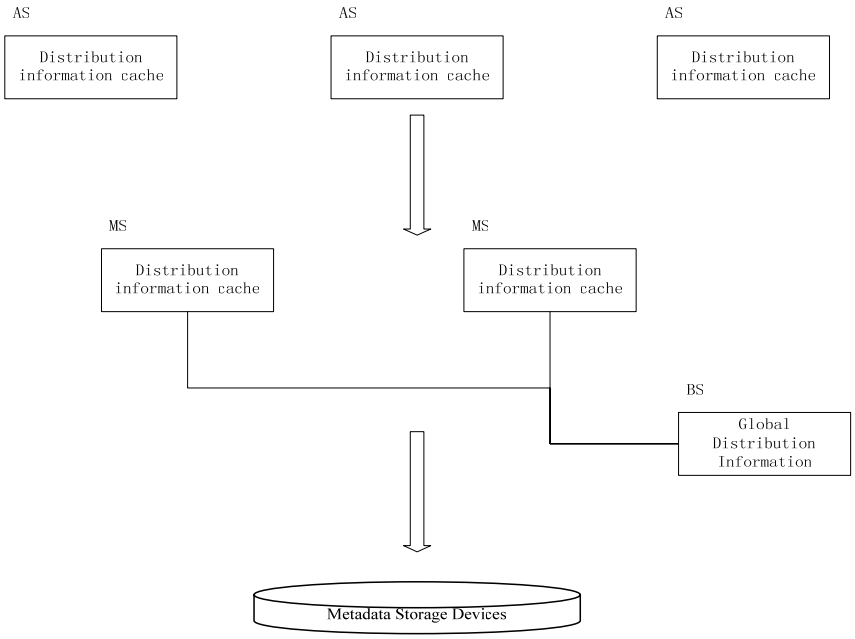


图 3.2 BWMMS 层次化元数据请求分布管理机制

图 3.2 描述的 BWMMS 层次化元数据请求分布管理机制由绑定服务器（Binding Server, BS），元数据服务器（Metadata Server, MS）和应用服务器(Application Server, AS)三个部分构成。BS 是文件系统元数据请求分布的集中决策点，主要完成活跃元数据与元数据服务器映射关系的管理工作。MS 接收 AS 的元数据请求，根据自己缓存的元数据分布信息处理请求。在必要的时候 MS 请求 BS 决策，并根据决策结果刷新元数据分布信息缓存。AS 是运行各种服务的服务器，它根据元数据请求访问结果缓存元数据的分布信息。

### 3.4.3 分布管理协议

元数据请求分布管理协议包括 AS 与 MS、MS 与 BS 之间的请求分布管理协议。AS 根据元数据请求的处理结果填充、或刷新其元数据分布信息缓存，其协议相对简单。MS 与 BS 间的元数据请求分布管理协议主要需要完成元数据请求分布映射的建立、查询和解除等工作。如图 3.3 所示。

活跃元数据的单一映射策略，决定活跃元数据的请求分布映射仅能由一个元数据服务器完成。当元数据分布在请求建立映射的服务器上时，这部分工作由同一个服务器完成。对应到图 3.3 的左半部分，其过程为：

根据 AS1 访问，MS1 通过 MAP 协议，请求 BS 为第一次访问变成活跃的元数据进行请求分布决策。其协议格式是(索引节点号，具有访问相关性的索引节点，元数据类型，映射模式)。BS 通过元数据请求分布算法，决定该元数据的宿主。并将结果返回给 MS1。