

网站架构设计

架构演变第一步：物理分离 webserver 和数据库

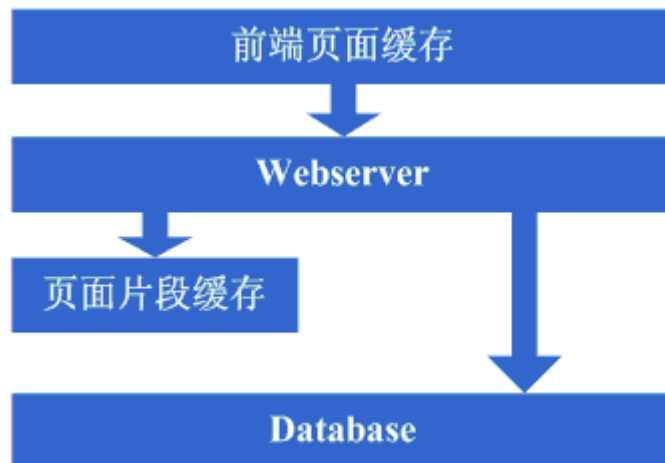


架构演变第二步：增加页面缓存



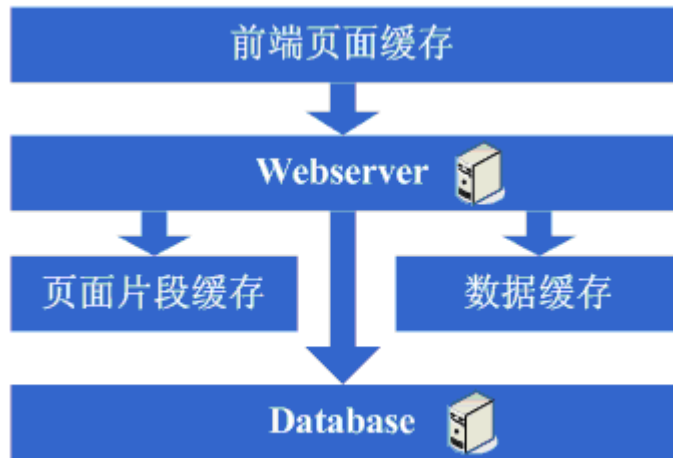
涉及知识点：前端页面缓存技术，例如 squid，如想用好的话还得深入掌握下 squid 的实现方式以及缓存的失效算法等。

架构演变第三步：增加页面片段缓存



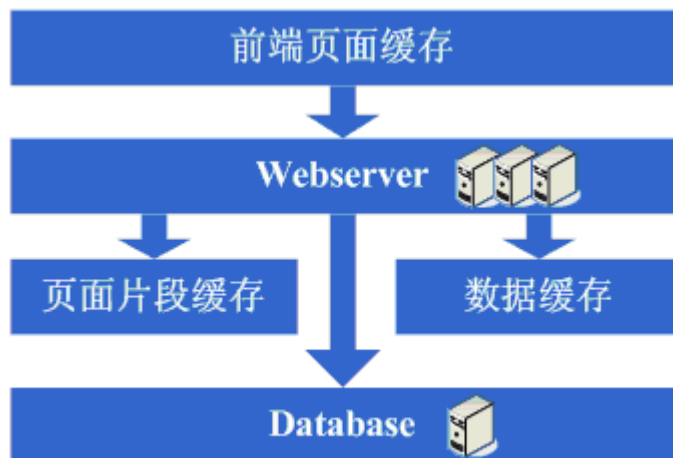
涉及知识点：页面片段缓存技术，例如 ESI 等，想用好的话同样需要掌握 ESI 的实现方式等。

架构演变第四步：数据缓存



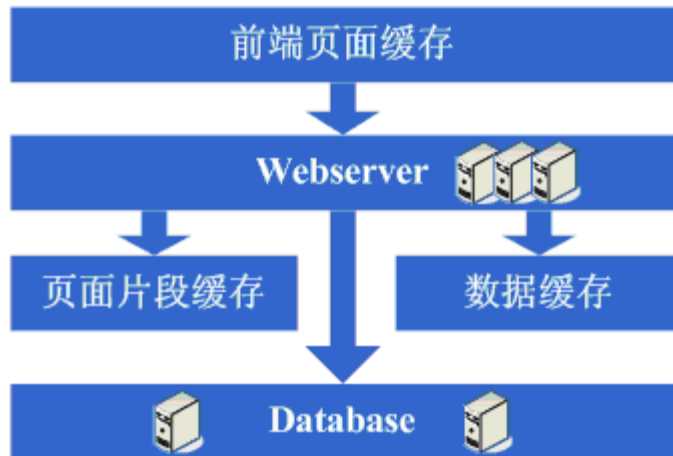
涉及知识点：缓存技术，包括像 Map 数据结构、缓存算法、所选用的框架本身的实现机制等。

架构演变第五步： 增加 webserver



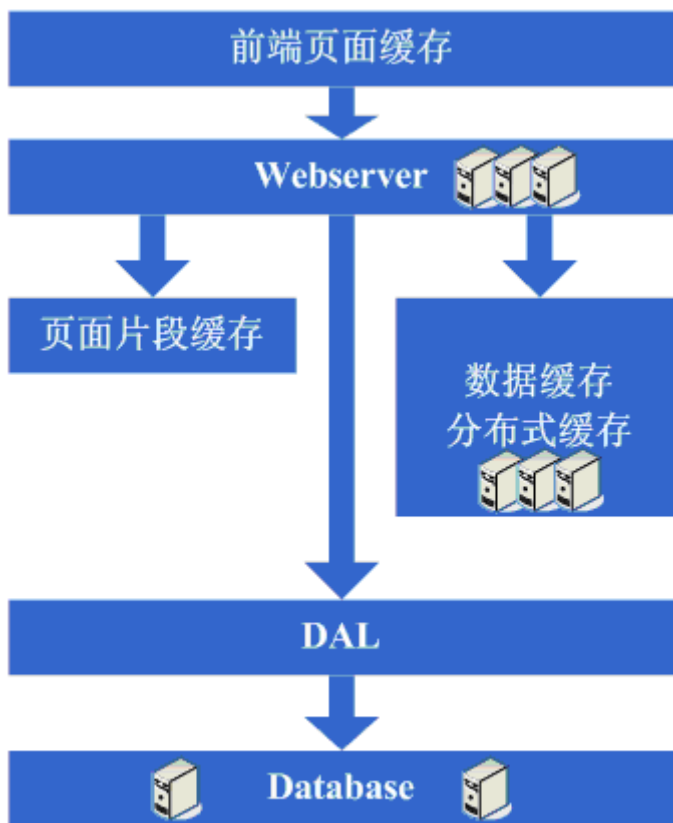
涉及的知识点：负载均衡技术（包括但不限于硬件负载均衡、软件负载均衡、负载均衡算法、linux 转发协议、所选用的技术的实现细节等）、主备技术（包括但不限于 ARP 欺骗、linux heart-beat 等）、状态信息或缓存同步技术（包括但不限于 Cookie 技术、UDP 协议、状态信息广播、所选用的缓存同步技术的实现细节等）、共享文件技术（包括但不限于 NFS 等）、存储技术（包括但不限于存储设备等）。

架构演变第六步：分库



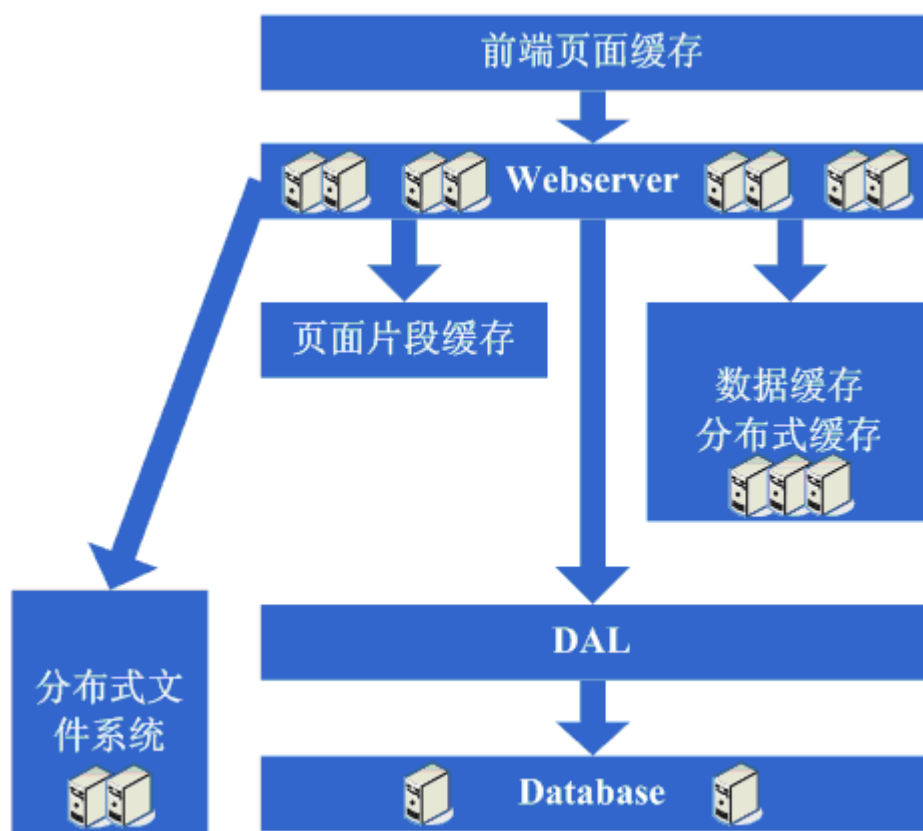
涉及的知识点：这一步更多的是需要从业务上做合理的划分，以实现分库，具体技术细节上没有其他的要求；但同时随着数据量的增大和分库的进行，在数据库的设计、调优以及维护上需要做的更好，因此对这些方面的技术还是提出了很高的要求的。

架构演变第七步：分表、DAL 和分布式缓存



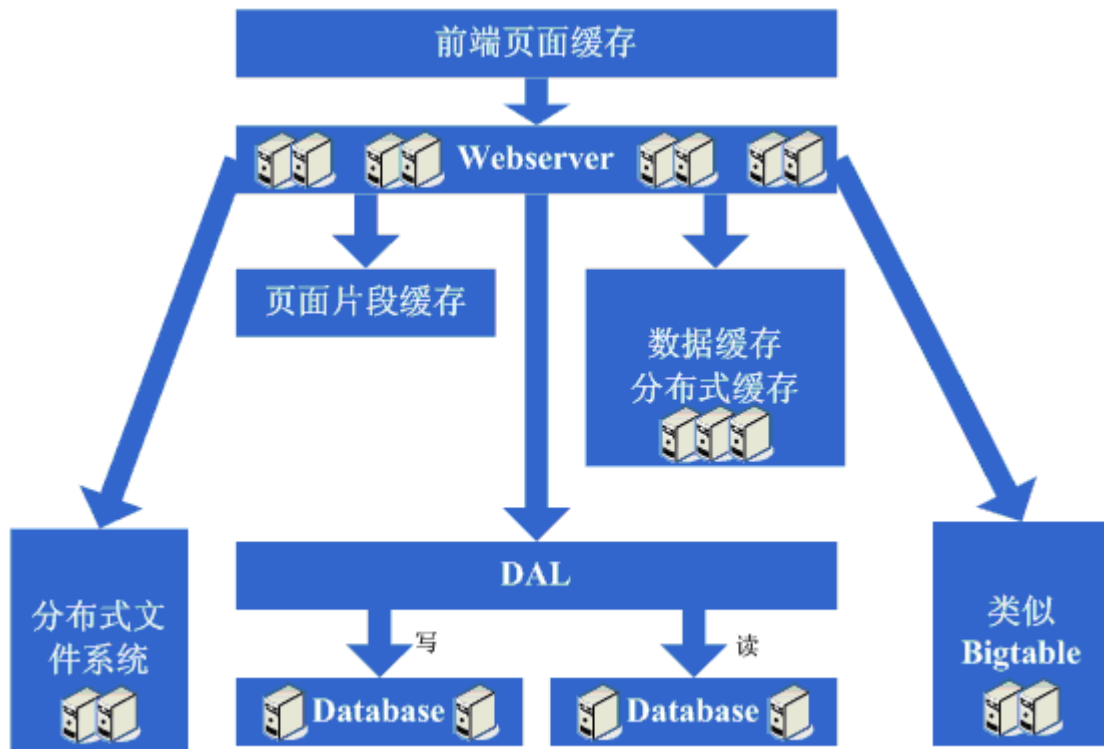
涉及的知识点:分表更多的同样是业务上的划分，技术上涉及到的会有动态 hash 算法、consistent hash 算法等；DAL 涉及到比较多的复杂技术，例如数据库连接的管理（超时、异常）、数据库操作的控制（超时、异常）、分库分表规则的封装等。

架构演变第八步：增加更多的 webserver



涉及的知识点：随着机器数的不断增长、数据量的不断增长和对系统可用性的要求越来越高，这个时候要求对所采用的技术都要有更为深入的理解，并需要根据网站的需求来做更加定制性质的产品。

架构演变第九步：数据读写分离和廉价存储方案



涉及的知识点：数据读写分离要求对数据库的复制、standby 等策略有深入的掌握和理解，同时会要求具备自行实现的技术；廉价存储方案要求对 OS 的文件存储有深入的掌握和理解，同时要求对采用的语言在文件这块的实现有深入的掌握。

架构演变第十步：进入大型分布式应用时代和廉价服务器群梦想时代



涉及知识点：对通信、远程调用、消息机制等有深入的理解和掌握，要求的都是从理论、硬件级、操作系统级以及所采用的语言的实现都有清楚的理解。运维这块涉及的知识体系也非常的多，多数情况下需要掌握分布式并行计算、报表、监控技术以及规则策略等等。

PS：数据库集群、数据挖掘、搜索、提升硬件配置、网络环境、改造操作系统、CDN 镜、安全、运维、运营、服务、存储。