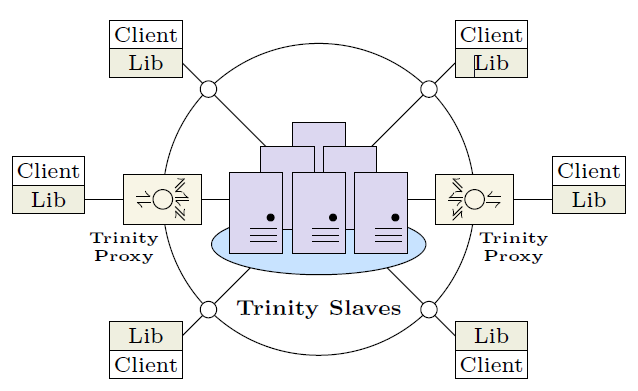
**基于分布式缓存的图处理系统：trinity**

**Trinity体系结构**

****

* Slave节点：存储一部分图数据，执行图计算任务。图计算任务包括向其他各类节点收发消息。
* Proxy节点：系统中的可选节点，不包含数据，只处理消息。作为client和slave节点的中间层，也用作消息聚集节点，可汇总来自多个slave的消息。
* Client节点：用户接口层，通过API和slave以及proxy节点通讯。

**Trinity的分布式缓存**

数据空间被划分为2^p个内存块（trunk），分布于m（ 2^p >m）个节点上，通常一个节点容纳多个trunk。

分解成多个trunk的原因：

1）分解成多个trunk有利于并发；

2）维持一个大型的哈希表将导致哈希冲突的概率增加。

**Trinity的图计算模式**

Trinity采用restrictive模型：每个vertex节点只和一个固定的邻居节点集合通讯。