

数据的逻辑结构和存储结构（物理结构）详解

《数据结构有哪些》一节讲到，数据的存储方式可分为[线性表](#)、[树](#)和[图](#)三种存储结构，而每种存储结构又可细分为顺序存储结构和链式存储结构。数据存储方式如此之多，针对不同类型的数据选择合适的存储方式是至关重要的。

那么，到底如何选择呢？数据存储结构的选择取决于两方面，即数据的[逻辑结构](#)和[存储结构](#)（又称[物理结构](#)）。

逻辑结构

数据的逻辑结构，简单地理解，就是指的数据之间的逻辑关系。

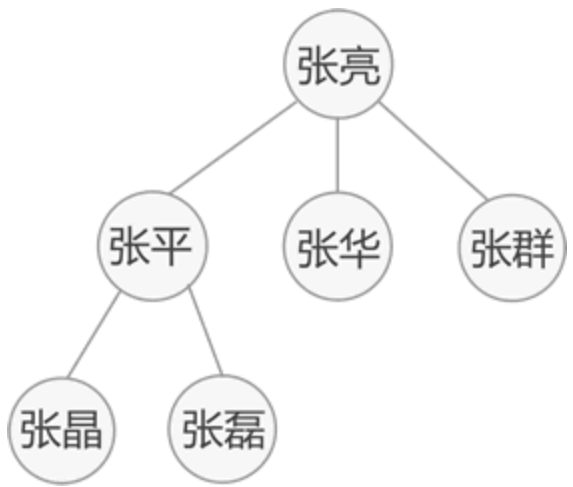


图 1 家庭成员关系图

例如，图 1 显示是一张家庭的成员关系图，从图中可以看到，张平、张华和张群是兄弟，他们的父亲是张亮，其中张平有两个儿子，分别是张晶和张磊。

以上所说，父子、兄弟等这些关系都指的是数据间的逻辑关系，假设我们要存储这样一张家庭成员关系图，不仅要存储张平、张华等数据，还要存储它们之间的关系，两者缺一不可。

一组数据成功存储到计算机的衡量标准是要能将其完整的复原。例如图 1 所示的成员关系图，如果所存储的数据能将此成员关系图彻底复原，则说明数据存储成功。

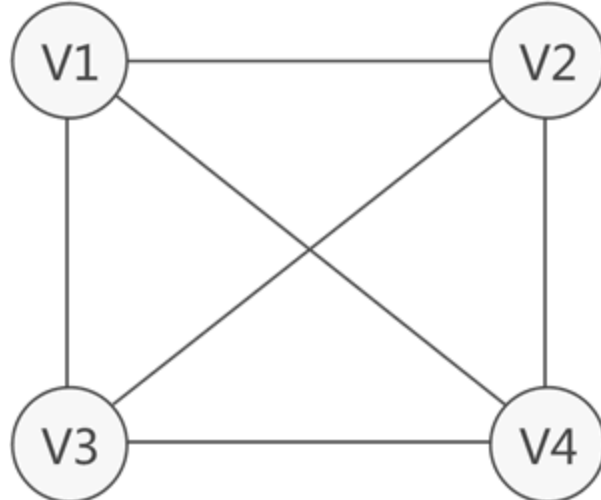


图 2 “多对多” 关系示意图

数据之间的逻辑关系可细分为三类，“一对一”、“一对多”和“多对多”：

- “**一对一**”：类似集合 $\{1,2,3,\dots,n\}$ 这类的数据，每个数据的左侧有且仅有一个数据与其相邻（除 1 外）；同样，每个数据的右侧也只有一个数据与其相邻（除 n 外），所有的数据都是如此，就说数据之间是“一对一”的逻辑关系；
- “**一对多**”：图 1 中的数据就属于“一对多”，因为对于张平来说，有且仅有一个父亲（张亮），但是有 2（多）个孩子；
- “**多对多**”：拿图 2 来说，从 V1 可以到达 V2、V3、V4，同样，从 V2、V3、V4 也可以到达 V1，对于 V1、V2、V3 和 V4 来说，它们之间就是“多对多”的关系；

通过学习数据结构，我们可以学到 3 种存储结构分别存储这 3 类逻辑关系的数据，换句话说：

1. **线性表**用于存储具有“一对一”逻辑关系的数据；
2. **树结构**用于存储具有“一对多”关系的数据；
3. **图结构**用于存储具有“多对多”关系的数据；

由此，我们可以通过分析数据之间的逻辑关系来决定使用哪种存储结构，但具体使用顺序存储还是链式存储，还要通过数据的物理结构来决定。

存储结构（物理结构）

数据的存储结构，也就是物理结构，指的是数据在物理存储空间上选择集中存放还是分散存放。假设要存储大小为 10G 的数据，则集中存放就如图 3a) 所示，分散存放就如图 3b) 所示。



图 3 数据的物理存储方式

如果选择集中存储，就使用顺序存储结构；反之，就使用链式存储。至于如何选择，主要取决于存储设备的状态以及数据的用途。

我们知道，集中存储（底层实现使用的是[数组](#)）需要使用一大块连续的物理空间，假设要存储大小为 1G 的数据，若存储设备上没有整块大小超过 1G 的空间，就无法使用顺序存储，此时就要选择链式存储，因为链式存储是随机存储数据，占用的都是存储设备中比较小的存储空间，因此有一定几率可以存储成功。

并且，数据的用途不同，选择的存储结构也不同。将数据进行集中存储有利于后期对数据进行遍历操作，而分散存储更有利于后期增加或删除数据。因此，如果后期需要对数据进行大量的检索（遍历），就选择集中存储；反之，若后期需要对数据做进一步更新（增加或删除），则选择分散存储。

至于为什么，我们会在详解两种存储结构时告知大家。