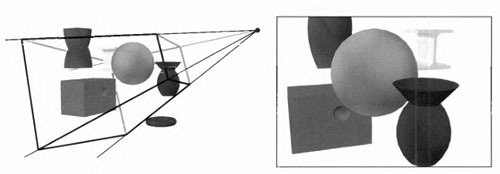
# 第2章 图形绘制管线

**“A chain is no stronger than its weakest link.”（链子的强度取决于它的最薄弱环节）**

本章的内容是实时绘制的核心，也就是后面将要讲述的绘制管线（或者简称管线）。绘制管线的主要功能就是在给定虚拟相机、三维物体、光源、照明模式，以及纹理等诸多条件下，如何生成或者绘制一幅二维图像。因此，不难看出，对于实时绘制来说，绘制管线是基础。图2.1描述的就是一个使用管线的过程，物体在图像中的位置与形状由它们的几何形状、相机位置以及环境特性等因素决定。而视觉外观则受材料属性、光源、纹理，以及光照模型的影响。



**图2.1 在左图中，相机放置在金字塔的顶端（四条线段的交汇点），只绘制视体内的图元。对于透视投影的绘制方式（如本图所示）来说，视体实际上就是一个锥体，即以矩形为底面的被裁金字塔。右图显示相机所能“看到”的东西，可以注意到在右边一幅图像中，并没有绘制左圈中最下面的那个红色圆环，这是因为它在视体之外。此外，左图中蓝色的扭曲棱柱被视锥体剪切掉了顶部。**

本章将集中讨论和解释绘制管线所包含的几个不同的工作阶段，重点介绍其中的功能而不是如何具体实现这些功能，其目的就是要深入理解图形绘制的具体功能，关于实现手段和方法将会在随后的章节中详细介绍。我们可以用例子来解释这个目的，例如，绘制直线时我们关心的是直线的特性（如顶点数据格式、颜色、模式类型等），它具体是通过Bresenham直线绘制算法还是通过对称双步算法实现的，不是很重要。通常，大部分绘制管线阶段都是通过不可编程的硬件来实现的，要进行优化和性能改善会非常困难。关于基本的绘制和填充算法，在Rogers所编写的书中有详细介绍。对于我们来说，就是决定什么时候，以及如何使用这些给定的方法。

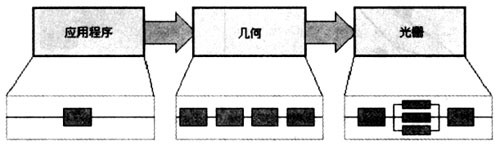
# 2.1 体系结构

在现实世界中，管线概念的表现方式非常多，如输油管道、工厂的装配线、滑雪缆车等，对于图形绘制来说，也同样采用了管线的概念。

一条管线通常包括多个阶段，无论其中的个别管线速度有多快，管线的整体快慢程度是由管线中最慢的那个阶段决定的。例如，在输油管道中，只有等到石油已经从第1阶段流到第2阶段以后，石油才能从第2阶段流入第3阶段。

理论上，如果要把一个非管线结构分成n个管线阶段，就应该给予一个大小为n的加速因子，例如，当一个滑雪缆车只有一个座位时，效率不会很高，但是如果增加座位，就可以提高效率，在相同的时间内可以把更多的滑雪者运送到山顶上。如果管线各阶段以平行的方式进行工作，那么必须等到其中最慢一个阶段结束工作之后，整个管线阶段才算结束。例如，在汽车装配线上，如果驱动车轮的装配阶段需要3分钟，而其他阶段均需要两分钟，那么制造一辆汽车可以达到的最快速度是3分钟，在装配驱动车轮的过程中，其他阶段必须要闲置1分钟。对于这类特殊的管线，驱动车轮的装配阶段是一个瓶颈，因为它决定整个装配过程的速度。

在实时计算机图形学中，也存在这种管线结构（如图2.2所示）。在概念上可以将其粗略分为3个阶段：应用程序、几何以及光栅。这种结构是绘制管线的核心机制，在实时计算机图形学应用中也广为使用，它同时也是随后的章节中讨论的重点。在绘制管线中，每个阶段自身也可能就是一条管线，这就是意味着它可以包含几个子阶段。这里，首先给出概念阶段（应用程序、几何，以及光栅）、功能阶段，以及管线阶段之间的区别。功能阶段规定了这个阶段需要执行的任务，并没有限制该任务在管线中的执行方式。从另一方面来说，一个管线阶段应该与其他管线阶段同时执行，但是为了满足高性能的要求，也可以对它们进行并行化处理。例如，几何阶段可以分成5个功能阶段，但是从图形系统的实现上，却将其划分为一个管线阶段。一个具体的实现方法可能会将两个功能阶段合并为一个管线阶段，同时也可能将另外一个比较耗时的功能阶段分成几个管线阶段，甚至可以让它们互相并行执行。



**图2.2 绘制管线的基本结构包括3个阶段：应用程序、几何，以及光栅。每个阶段本身也可能是一条管线，如图中的几何阶段所示。此外，还可以对有的阶段进行全部或者部分并行化处理，如图中的光栅化阶段。应用程序阶段虽然是一个单独的过程，但是依然可以对之进行管线化或者并行化处理**

最慢的管线阶段决定绘制速度，即图像更新速度，这种速度一般用fps来表示，也就是每秒绘制的图像数量，或者用Hz来表示。由于涉及到管线，因此不能将想绘制的所有数据通过整个管线的时间进行简单相加，这就是管线结构的特点，它允许对某些阶段进行并行处理。如果能够确定其中的瓶颈位置，也就是管线中的最慢阶段，同时知道所有数据通过该阶段需要花费的时间，就可以计算出绘制速度。例如，假设瓶颈阶段需要20ms来执行，那么绘制速度是1/0.020＝50Hz。但是前提条件是输出设备能够以这种速度不断更新；否则，实际的数据输出速度就会更慢。在其他一些管线应用场合中，绘制速度往往可以用吞吐量这个术语来替代。

**示例：绘制速度**。假设输出设备的最高更新频率为60 Hz，而且找到了绘制管线的瓶颈位置，并且执行该瓶颈阶段需要的时间是62.5ms。这样，绘制速度可以按下列步骤计算。首先，忽略输出设备，则可以提供的最高绘制速度为1/0.0625＝16Hz。其次，将这个值调整为输出设备的频率：60Hz意味着绘制速度可以是60Hz、60/2＝30Hz、60/3＝20Hz、60/4＝15Hz、60/5＝12Hz等。这就说明期望的绘制速度可以是15Hz，因为这是小于16Hz时输出设备可以达到的最快速度。

如同字面意思所表达的那样，应用程序阶段由应用程序来驱动，因此可以通过软件在通用处理器CPU中实现，这些CPU通常都是多核的，可以并行地处理多个线程，这让CPU可以高效地运行应用程序阶段的大量任务。在CPU上执行的传统任务包括碰撞检测、加速算法、动画，物理模拟等。第二阶段是几何阶段，它处理变换、投影、光照等处理。此阶段主要是计算绘制的内容、如何绘制，以及在什么地方绘制。几何阶段主要是在图形处理单元（GPU）中执行的，GPU包含了很多可编程和固定功能的内核。最后，光栅阶段是利用前面阶段产生的数据进行图像绘制，光栅阶段完全在GPU中进行处理。所有这些阶段和它们内部的管线将在2.2～2.4节中详细讨论，有关GPU处理的细节将在第3章中介绍。