

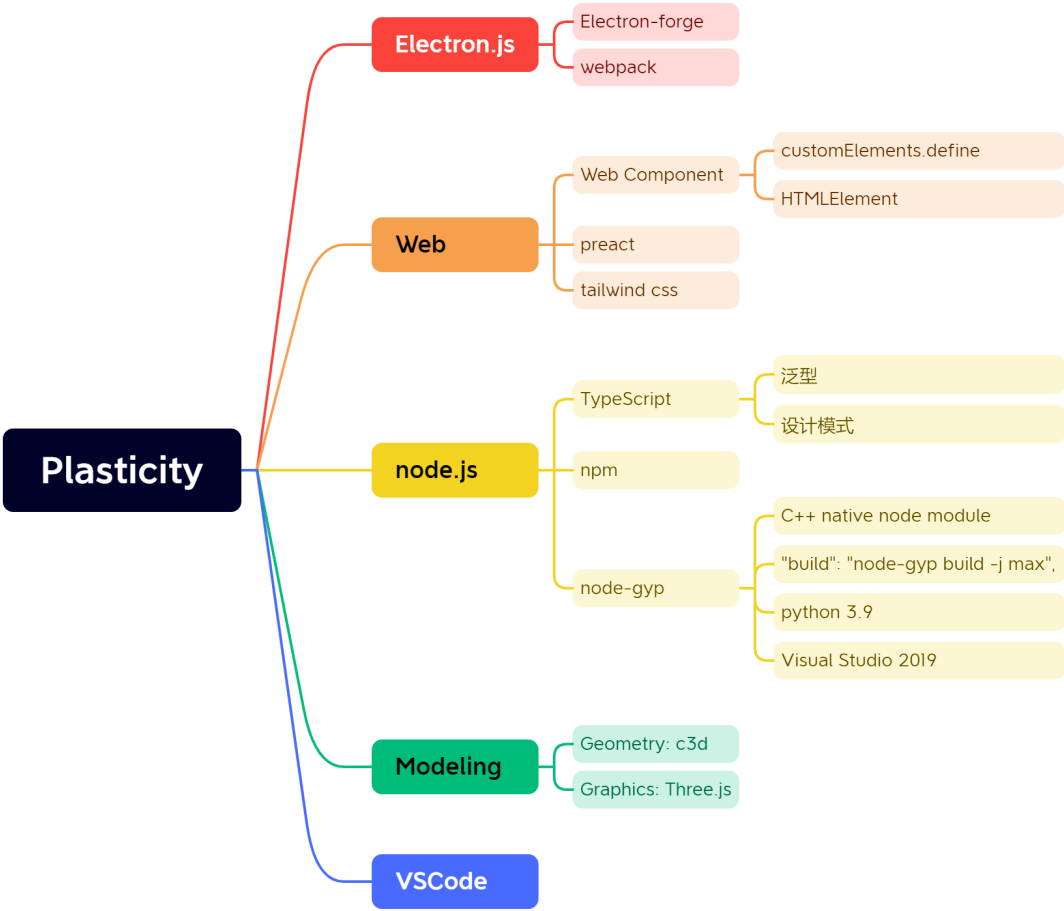
2022年中的源代码版本，最后为0.6.30。这是支持c3d的最后版本。这个版本的代码不能使用完整的实体造型功能，只能作为代码调试，学习交互建模开发。

学习资源

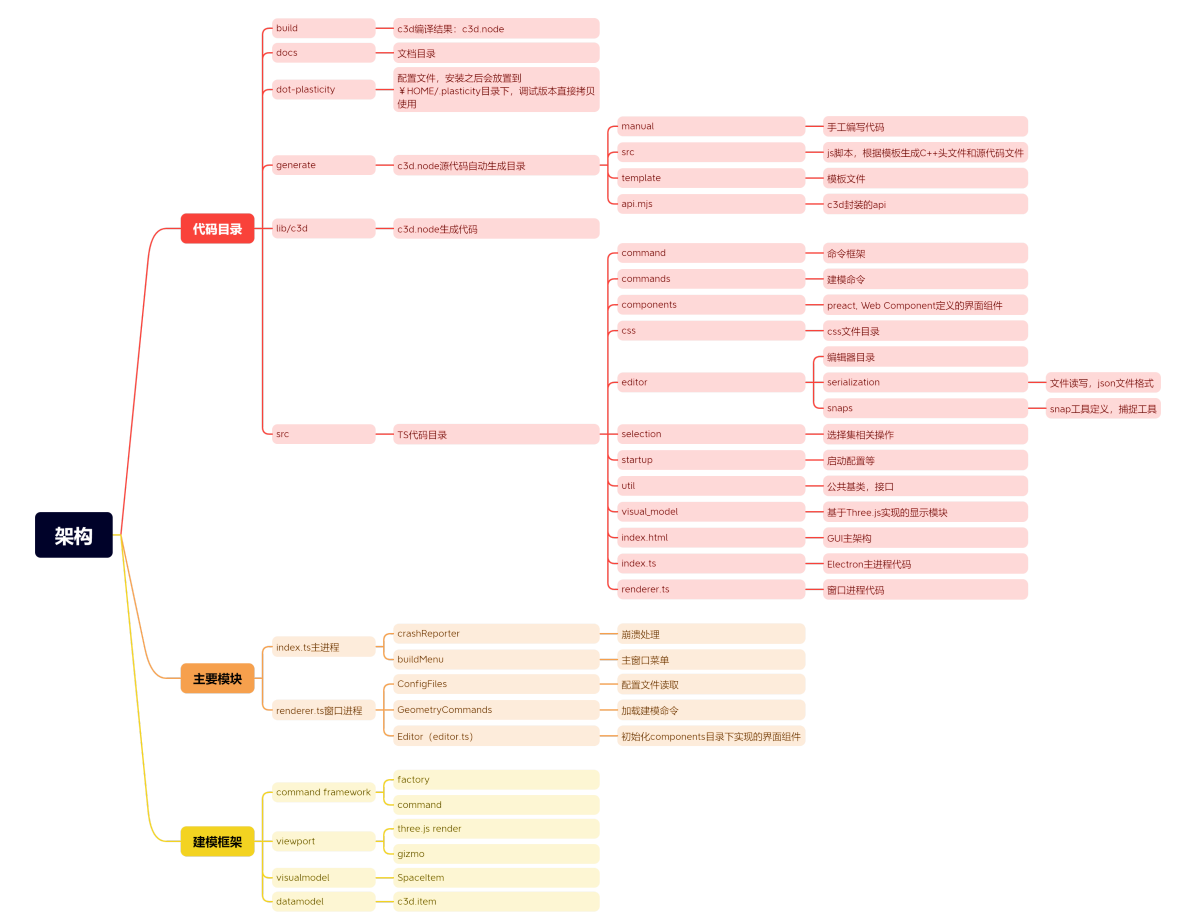
开源版本0.6.30，相应的学习资源：

- [【中文字幕】13 分钟介绍 Plasticity 0.3哔哩哔哩bilibili](#)
- [PLASTICITY基础建模教学01 乐高积木 新手教程保姆级教学哔哩哔哩bilibili](#)
- [【Plasticity】轻量级颜值高的3D建模利器&案例演示哔哩哔哩bilibili](#)

技术栈

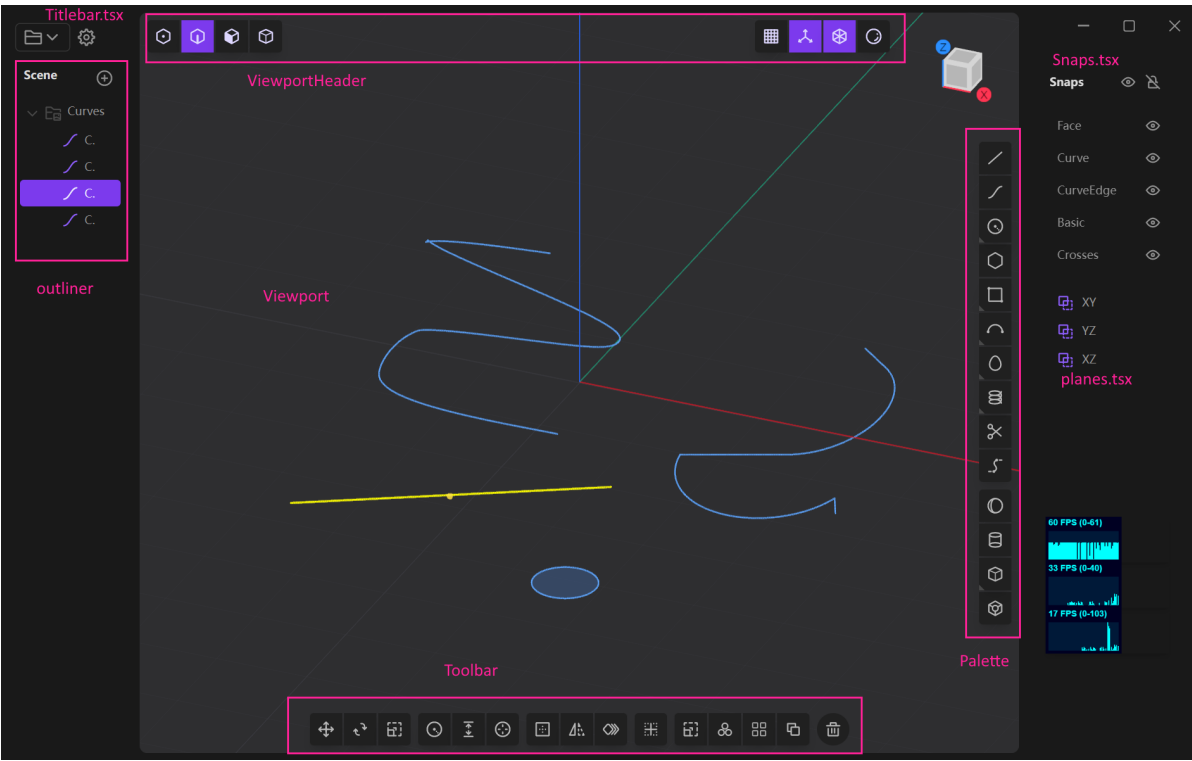


架构

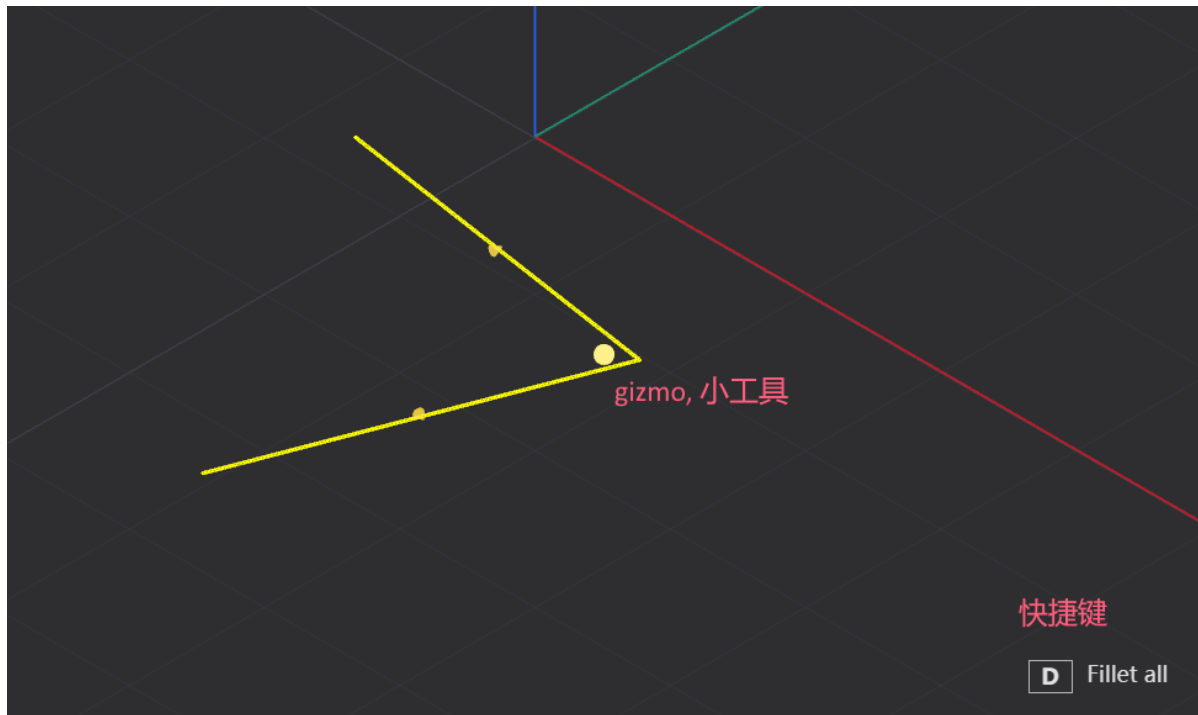


Presented with xmind

界面



Modeling



plasticity中，几何建模功能通过以下两种方式启动交互过程：

- 命令。右侧的palette中定义的基本命令，选中几何后toolbar显示的相关命令。可以通过鼠标停留在按钮上显示的tooltip文字搜索相应的命令名字。例如三点圆弧命令："Three-point arc"，对应的command是ThreePointArcCommand (ThreePointArcCommand.ts) ,execute函数是命令执行流程。注意，命令中有await调用的同步函数，会等待用户输入点或者选择几何元素，例如这里是PointPicker，只有取消命令或者返回点才从函数返回。这个命令非常典型，可以设置断点学习这里需要用到的各种建模需要实现的相应类和接口
- gizmo。小工具实际上也是通过命令系统启动。例如上图给出的是一条折线。选中后，显示两种gizmo，大圆点是fillet工具，对应的命令是ModifyContourCommand (ModifyContourGizmo.ts)。ModifyContourCommand命令是如何启动的呢？
 - editor的数据成员commandForSelection: SelectionCommandManager。这个类负责维护一个映射表，参考他的成员函数commandFor。对应选择的几何体，按照类型返回相应的修改命令。这个图是ModifyContourCommand
 - 如果没有当前命令，根据当前选中几何体执行命令，也就是上一步的逻辑，参考CommandExecutor.dequeue()。可以通过在ModifyContourGizmo.ts中设置断点查看相应的调用堆栈和调用逻辑。

ThreePointArcCommand

