|  |  |  |  |
| --- | --- | --- | --- |
|  | 场景与数据 | 物理模块 | 渲染模块 |
| 第一阶段：  设计  功能测试  任务分解  demo | 1：定义接口，可使用保证引用渲染模块或物理模块的原生数据结构  2：urdf场景数据支持 | 熟悉第三方引擎基本的需求接口使用方法，包括刚体碰撞设置,关节驱动方式。 | 熟悉图形库基本的渲染方法，此阶段不关心渲染质量，打通基本的场景渲染工作流，支撑物理位姿的精准显示即可。 |
| 第二阶段：  封装  层次化  严谨化 | 1：支持引擎层与应用层的逻辑交互(c++与python交互)。 | 解耦模型物理数据结构  统一场景数据结构  抽象/封装物理引擎接口 | 解耦模型的数据结构  统一场景数据结构  抽象/封装渲染引擎接口 |
| 第三阶段  扩展开发 | 1：urdf场景格式完善  2：usd场景格式支持  3：其他模型格式支持  3：指令集抽象 | 高级物理特性支持  比如如用网格模型代替凸包集 | 物理渲染完善与优化  性能优化 |
| 第四阶段 | 场景应用开发 | 物理仿真性能优化 | 硬件光追  可微渲染等等  其他引擎替代参考  定制自研替代参考 |
| 第四阶段  集成环境 | 对外接口开放  工具界面开发 | 1：性能优化  2：最新特性支持 | 1：性能优化  2：最新特性支持 |

仿真引擎开发