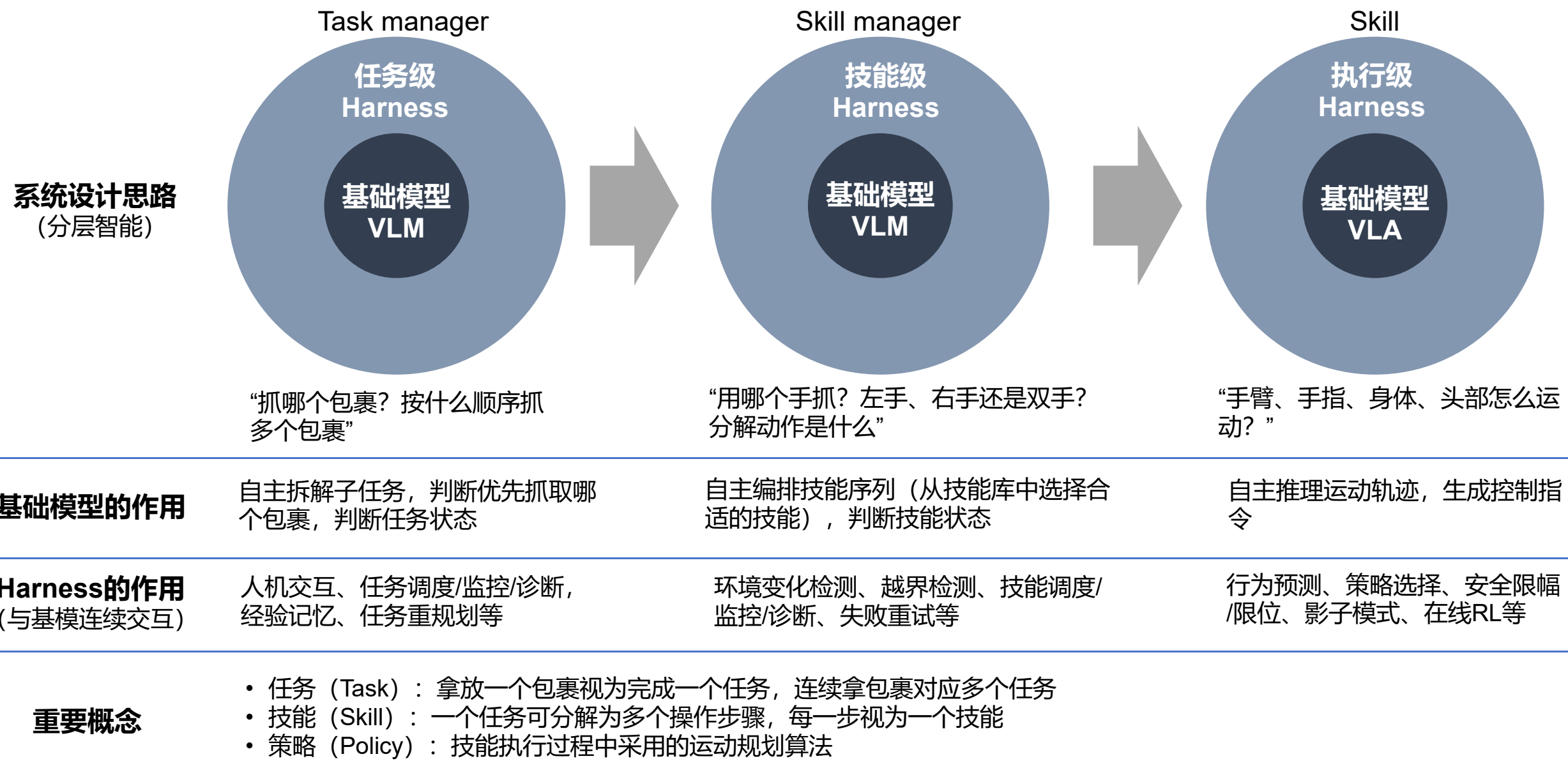
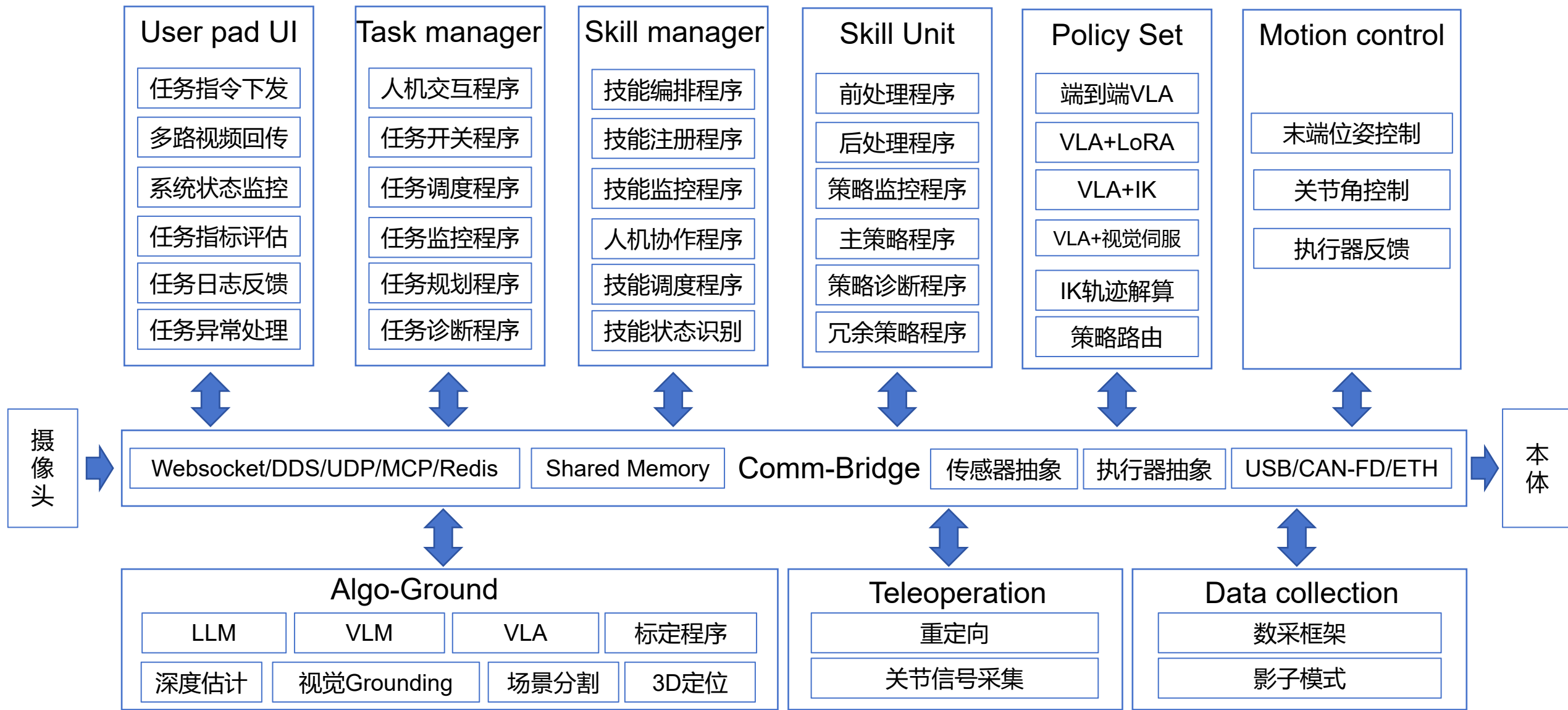


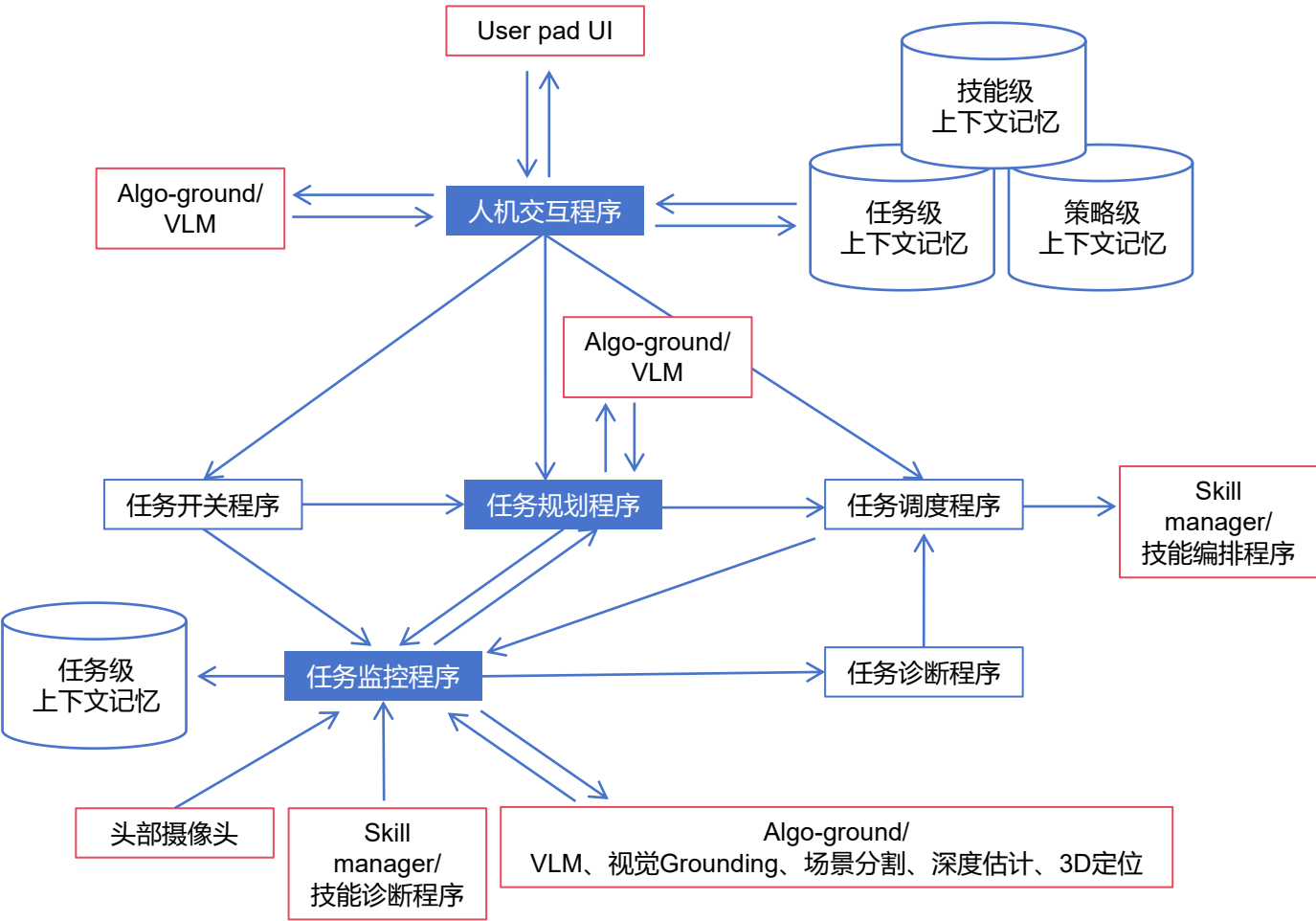
# 总体思路



# 功能逻辑架构

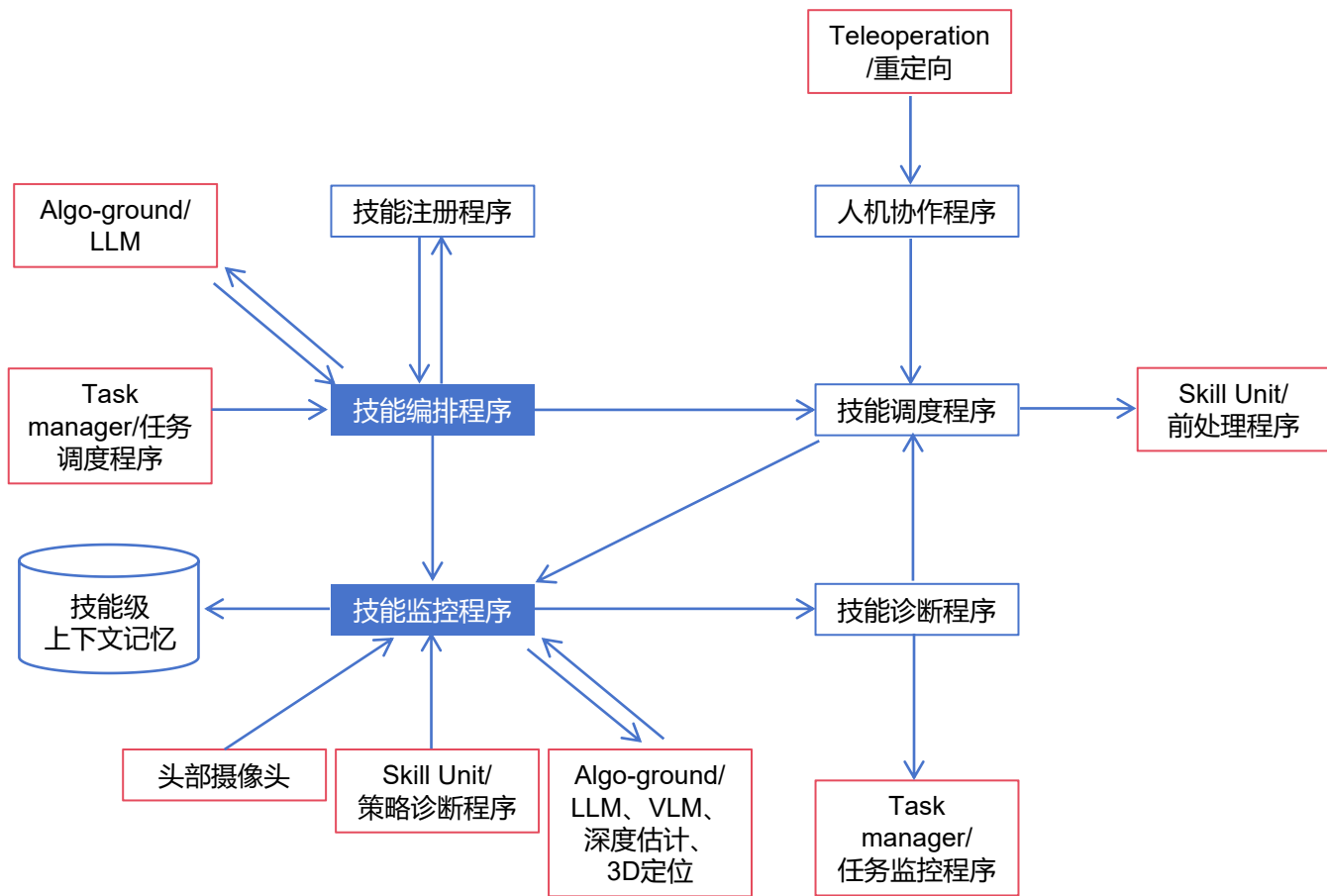


# Task manager



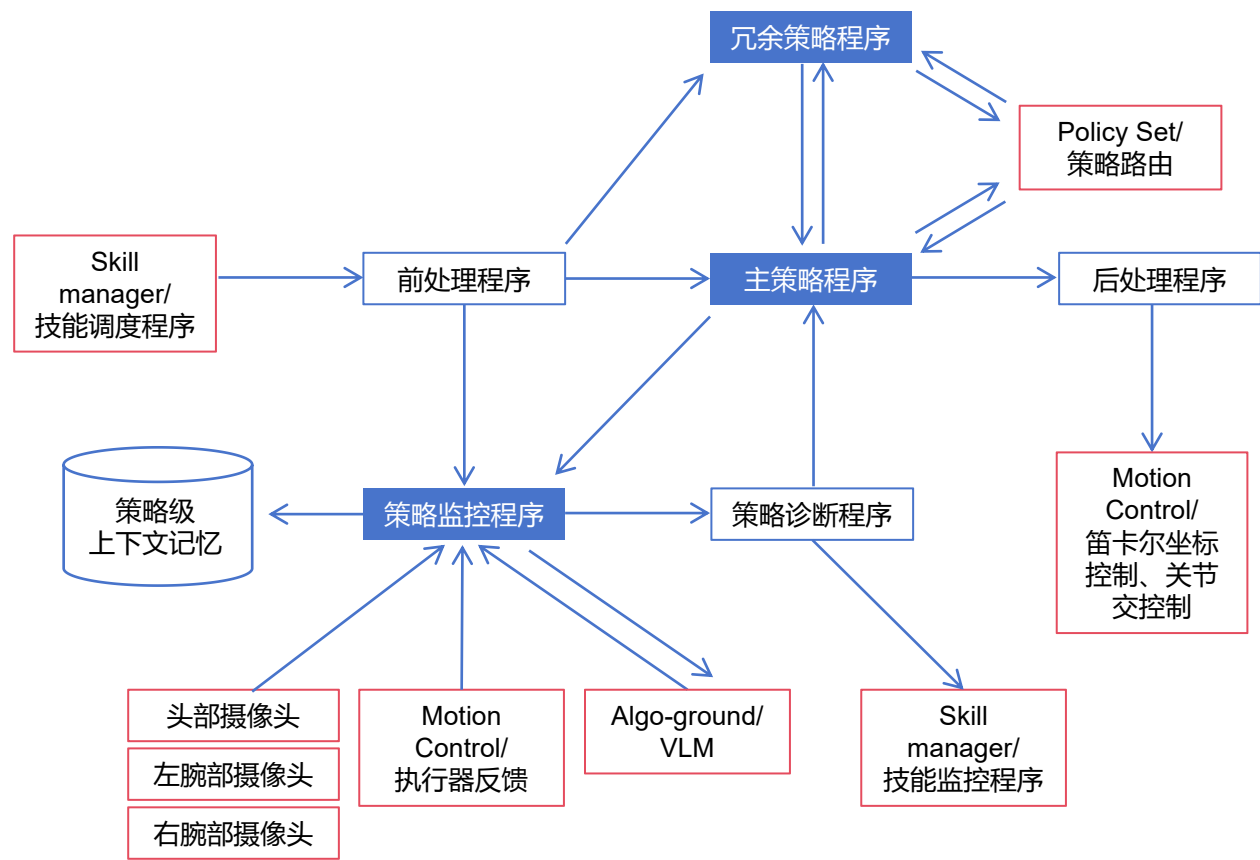
| 功能模块            | 功能点                                                                                                              |
|-----------------|------------------------------------------------------------------------------------------------------------------|
| 人机交互程序          | 接收并理解多模态任务指令（指令结构化、任务开启/终止/暂停/复位等语言指令）、在线反馈任务状态数据（成功/失败/进度、当前子任务名称、包裹类型/尺寸/位姿）、模式切换指令（遥控/自主）、打印日志、异常信息反馈、上下文回传   |
| 任务开关程序<br>★ (蒲) | 执行任务启动流程（系统初始化/参数加载/模型加载）、执行任务结束流程（持久化/生成任务报告）                                                                   |
| 任务规划程序<br>★ (蒲) | 自主分解任务队序（包裹优先级顺序，队列长度=3）、接收人工指定任务、任务可行性校验、任务队列动态管理（出队、入队、重排、插队、剔除）                                               |
| 任务调度程序<br>★ (蒲) | 按任务队列顺序发送子任务指令，发送系统复位指令、响应任务中断（任务故障码激活）、模式切换                                                                     |
| 任务监控程序<br>(尹)   | 根据整体任务目标和当前子任务指令，识别包裹类型、观测包裹尺寸、选择包裹、识别包裹位姿/所在区域、确定包裹放置位置、★<br>判断任务进度、<br>校验任务跳转条件、判断任务是否结束、判断任务成功/失败、★<br>任务统计分析 |
| 任务诊断程序          | 故障识别（任务规划失败、软件故障）、置位/取消故障码（写入任务级上下文）                                                                             |
| 任务级上下文<br>(蒲)   | 任务提示词、任务日志、任务调试数据、任务关键帧数据、★<br>标定数据、任务约束、★<br>质量要求、安全要求、故障报警数据、任务统计数据、上下文压缩                                      |

# Skill manager



| 功能模块            | 功能点                                                                                 |
|-----------------|-------------------------------------------------------------------------------------|
| 技能注册程序<br>★ (李) | 技能库索引、技能添加/更新/删除、结构化技能描述、技能参数配置、技能版本管理                                              |
| 技能编排程序<br>★ (李) | 根据子任务指令和技能库，组装结构化技能原语、自主生成技能序列，在技能库中匹配技能API、技能可行性校验、技能序列动态管理（出队、入队、重排、插队、剔除）        |
| 技能监控程序<br>(朱)   | 越界检测、★<br>碰撞检测、<br>判断技能执行进度、<br>校验技能跳转条件、★<br>判断技能成功/失败、★<br>判断技能执行是否结束、★<br>技能统计分析 |
| 人机协作程序<br>★ (李) | 通过遥操设备信号判断人工干预意图，触发人工接管                                                             |
| 技能调度程序<br>★ (李) | 行为树、响应技能中断（技能故障码激活）                                                                 |
| 技能诊断程序          | 故障识别（技能编排失败、软件故障）★<br>置位/取消故障码（写入技能级上下文）                                            |
| 技能级上下文记忆        | 技能日志、技能调试数据、技能关键帧数据、★<br>标定数据、技能约束、★<br>操作要求、安全要求、故障报警数据、<br>技能统计数据、上下文压缩           |

# Skill Unit



| 功能模块               | 功能点                                                  |
|--------------------|------------------------------------------------------|
| 前处理程序<br>★ (谢、蒲、李) | 前置条件校验、参数提取、图像预处理                                    |
| 主策略程序<br>★ (谢、蒲、李) | 推理程序、调用主策略算法、切换冗余策略算法                                |
| 冗余策略程序<br>★ (朱)    | 推理程序、调用冗余策略算法                                        |
| 后处理程序<br>★ (朱)     | 轨迹内插平滑、接口适配、IK运动学解算、安全限位                             |
| 策略监控程序<br>★ (朱)    | 失败概率/失败类型预测 (力、位、视觉混合判断)                             |
| 策略诊断程序<br>(朱)      | 故障识别 (动作异常、软件故障、本体故障、传感器故障)、★<br>置位/取消故障码 (写入策略级上下文) |
| 策略级上下文<br>(谢、蒲、李)  | 策略关键帧数据、标定数据、策略约束、★<br>故障报警数据、策略统计数据、上下文压缩           |

# 结构化技能

System 2 (VLM + Harness):

输入: "把杯子放到桌上"

推理: → 拆解为 [技能: place, 目标: cup, 位置: table\_surface]

输出: 结构化技能调用 + 约束条件

```
{
  "skill": "place",
  "target": "cup",
  "destination": "table_surface",
  "constraints": {
    "orientation": "upright",
    "max_force": 5N,
    "avoid": ["near_edge"]
  }
}
```

System 1 (VLA + Harness):

输入: 上述结构化指令 + 当前视觉

推理: → VLA 将技能参数映射为动作轨迹

输出: 关节扭矩序列

| 类别  | 编号 | 原子技能名称     | 结构化定义 |
|-----|----|------------|-------|
| 感知类 | P1 | 视觉定位 (尹)   | {XXX} |
|     | P2 | 视觉检测 (尹)   |       |
| 运动类 | M1 | 头部运动 (朱)   |       |
|     | M2 | 腰部运动 (朱)   |       |
| 抓取类 | G1 | 左手抓取 (李)   |       |
|     | G2 | 右手抓取 (谢)   |       |
| 操作类 | O1 | 拨移包裹 (谢)   |       |
|     | O2 | 左手旋转包裹 (李) |       |
|     | O3 | 右手旋转包裹 (谢) |       |
| 协作类 | C1 | 左右手传递 (蒲)  |       |
|     | C2 | 双手搬移 (蒲)   |       |