

流程编排功能 (基础能力)

1. 介质说明

下载路径: <http://14.18.100.250:8989/release/orchestration/0-basic/>

- 环境要求: 环境要求.txt
- 云工程: gpfdc_orchestration
- 工作空间备份包: 流程编排_完整工作空间.zip

2. 部署步骤

- 按 **环境要求** 将相关内容升级到对应版本
- 导入 **云工程** 并发布插件
- 创建新的工作空间
- 进入工作空间
- 切换工作空间菜单 -> 导入 -> 完整导入 **工作空间备份包**

3. 使用说明

- 界面使用
 - 访问所在平台 /orch 应用
 - 智能体菜单切换可用智能体进行使用
 - 节点运行完毕后点击绿色图标可查看对应数据
 - 节点运行过程点击运行图标可查看大模型输出内容
 - 后台数据中可见对应智能体所有会话
 - 允许通过点击返回并继续会话
- 设计示例
 - 见 **测试智能体** 设计
- 生效按钮
 - 智能体首次部署时需点击
 - 智能体设计节点数量或关系发生变更时需点击
- 智能体设计调整除需生效情况外其余皆实时生效
- 智能体处理逻辑建议分开独立工程进行开发
- 迁移说明
 - 导出
 - 独立工程发布到云仓库
 - 相关的LLM配置通过界面选中导出
 - 智能体设计通过界面选中导出
 - 导入
 - 导入云工程

- 通过界面导入LLM配置
- 通过界面导入智能体设计并生效
- 界面嵌入

```
Map<String, String> params = new HashMap<>();
// params.put(对应智能体参数名, 参数值);

String instanceUuid = null;
String agentCode = "智能体编码";
String initword = "初始化文本";

OrchestrationChatPanel<OrchestrationChatParam> orchestrationChatPanel = new
OrchestrationChatPanel<>();
OrchestrationChatParam param = new OrchestrationChatParam();
param
    .setAgentParams(params)
    .setAgentCode(agentCode)
    .setCurrentInstanceUuid(instanceUuid) // 若存在实例, 则使用该实例回显
    .setInitword(initword) // 若当前实例UUID为空, 则尝试发送该文本创建新实例
    // 默认聊天的信息, 不给也无所谓
    .setGptServiceConfig(new GptServiceConfigDto())
    .setPrompt("");
orchestrationChatPanel.setWidgetParam(param);
SinglePanelDto panelDto = (SinglePanelDto)
orchestrationChatPanel.getWidget(panelContext);
```

- 更新说明
 - **工作空间备份包**: 上传时间无变更时无需重复导入
 - **云工程**: 保持最新版本

测试智能体

1. 介质说明

下载路径: <http://14.18.100.250:8989/release/orchestration/appTest>

- 云工程: `gpf_dc_orch2AppTest`
- 智能体设计: `数据(智能体设计).zip`
- LLM配置: `数据(LLM配置).zip`

2. 部署步骤

- 见 **流程编排功能 -> 迁移说明 -> 导入**

3. 使用说明

- 界面使用
 - 见 **流程编排功能 -> 使用说明 -> 界面使用**
- 界面嵌入
 - 见 **流程编排功能 -> 使用说明 -> 界面嵌入**
- api 调用
 - 常量定义: `orch2AppTest.consts.AppTestAgentConst`
 - 输入格式: `流程UUID#应用编码`
 - 参数说明
 - 自定义路径: `AppTestAgentConst.PARAM_KEY_CUSTOM_PATH`
 - DTO类: `orch2AppTest.dto.RoutePathDefinedDto`
 - JSON数据列表, 示例

```
[
  {
    "path": [
      {
        "node": "节点1",
        "action": "提交"
      },
      {
        "node": "节点2",
        "action": "提交"
      },
      {
        "node": "节点3",
        "action": "通过"
      },
      {
        "node": "节点4",
        "action": "通过"
      },
      {
        "node": "节点5",
        "action": "提交"
      },
      {
        "node": "结束"
      }
    ]
  }
]
```

- 自定义数据: `AppTestAgentConst.PARAM_KEY_CUSTOM_DATA`
 - DTO类: 嵌套Map结构, 以字段中文名作为key
 - JSON数据列表, 示例

```
[
  {
    "名称": "项目报告",
    "关联模型字段A": "关联模型A-2",
    "关联模型字段B": "关联模型B-3",
    "时间字段": 1737601555280,
    "整数字段": 123456,
    "小数字段": 99.99,
    "嵌套模型字段A": {
      "名称": "子项目",
      "嵌套模型字段B": {
        "名称": "详细说明"
      }
    },
    "嵌套模型字段B": {
      "名称": "备注信息"
    }
  }
]
```

- 接口代码:

- 接口类: `cell.orchestration.service.IOrchestrationRuntimeService`

```
/**
 * 创建对话实例
 *
 * @param
 * @param dao      dao对象
 * @param agentCode 智能体编码
 * @param initword  初始语句
 * @param code      标识, 无唯一限制
 * @param paramMap  参数表
 * @return
 * @throws Exception
 */
PDCForm createInstance(IDao dao, String agentCode, String initword,
String code, Map<String, String> paramMap) throws Exception;
```

- 更新说明

- **云工程**: 保持最新版本
- **智能体设计**: 上传时间无变更时无需重复导入, 导入后需手动生效