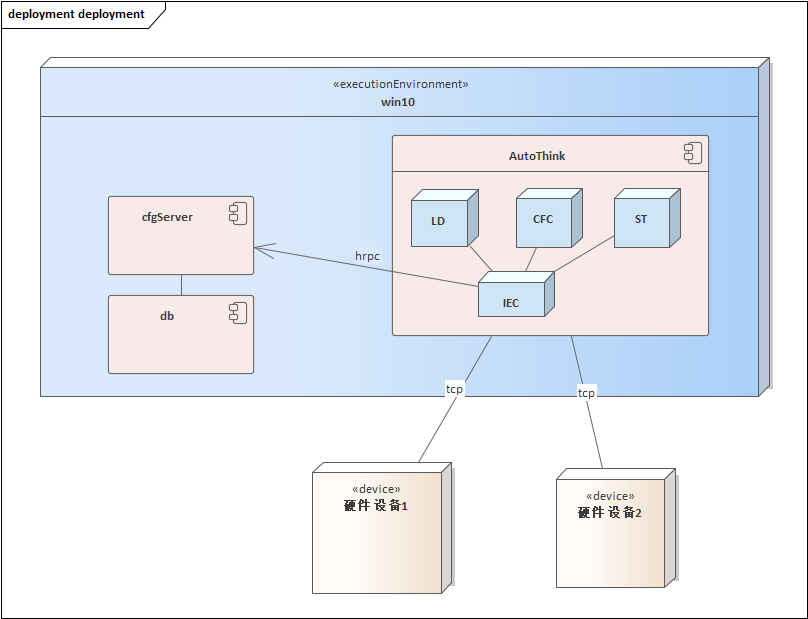
LD模块设计说明

# 1需求

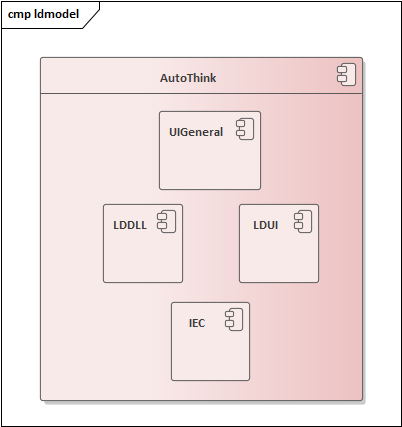
1. 满足LD编辑器各种界面操作功能
2. 满足LD编辑器导入导出支持IEC-10标准
3. 满足LD编辑后语法错误校验功能
4. 满足LD编辑后转C功能

# 2业务处理

## 2.1模块部署



## 2.2模块架构



AT中需要增加LDDLL数据模块以及LDUI模块

### 2.2.1 IEC模块职责

1、负责和数据库交互，进行数据存储和加载等。

2、负责IEC平台公共处理，创建不同语言POU，编译预编译等。

### 2.2.2 LDDLL模块职责

1、负责维护组态逻辑数据。

2、负责序列化反序列化数据。

3、负责转C。

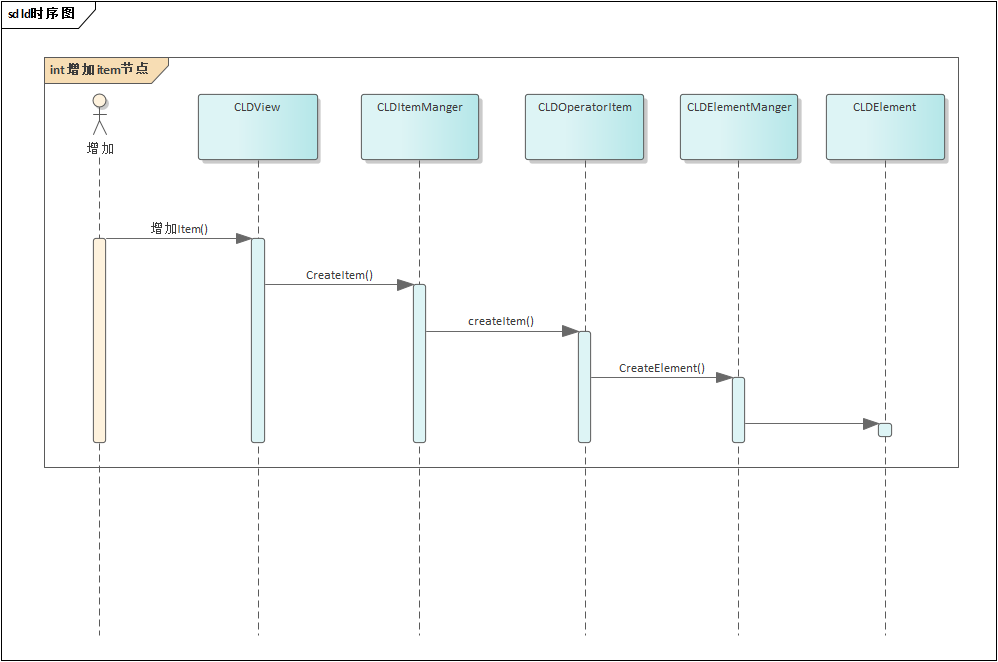
### 2.2.3 LDUI模块职责

1、负责和框架交互

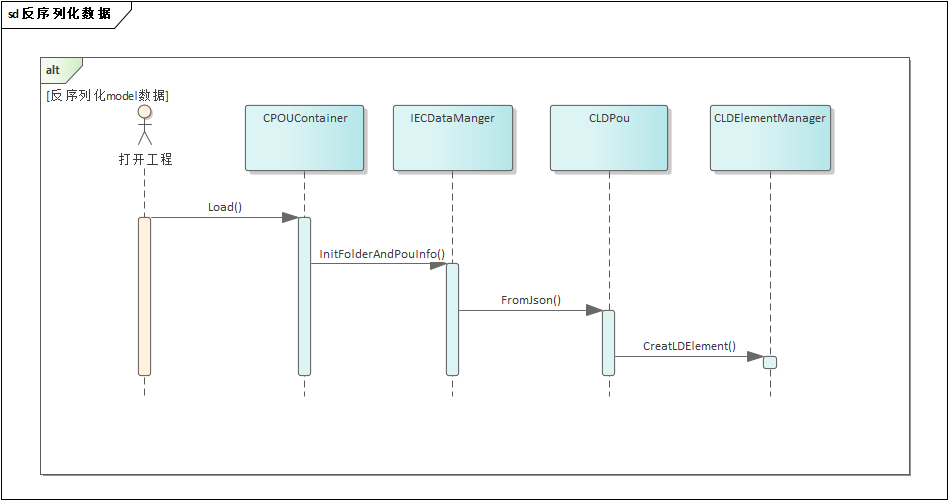
2、负责界面组态功能

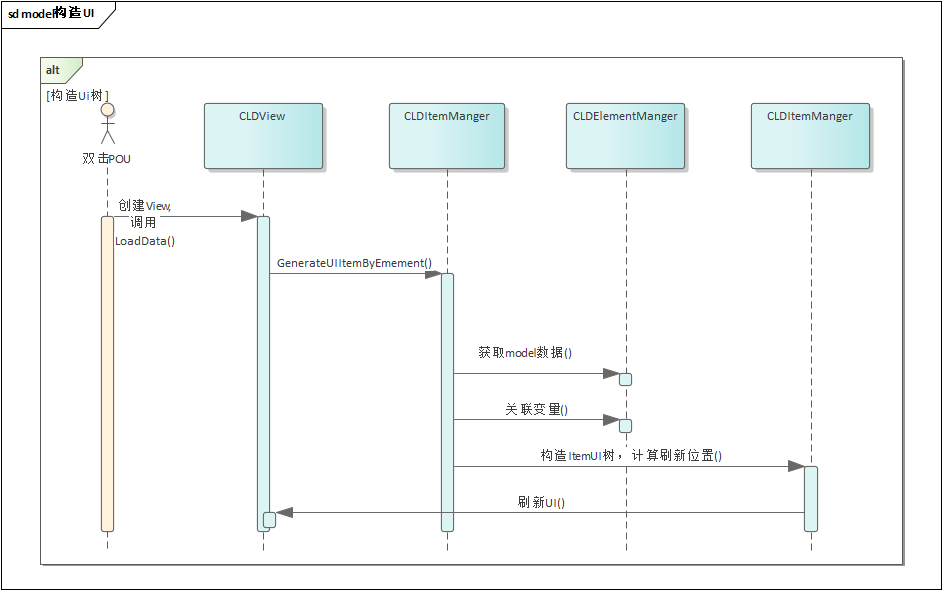
## 2.3主要业务流程

### 2.3.1、增加Item流程



### 2.3.2、反序列化生成逻辑树以及UI树流程

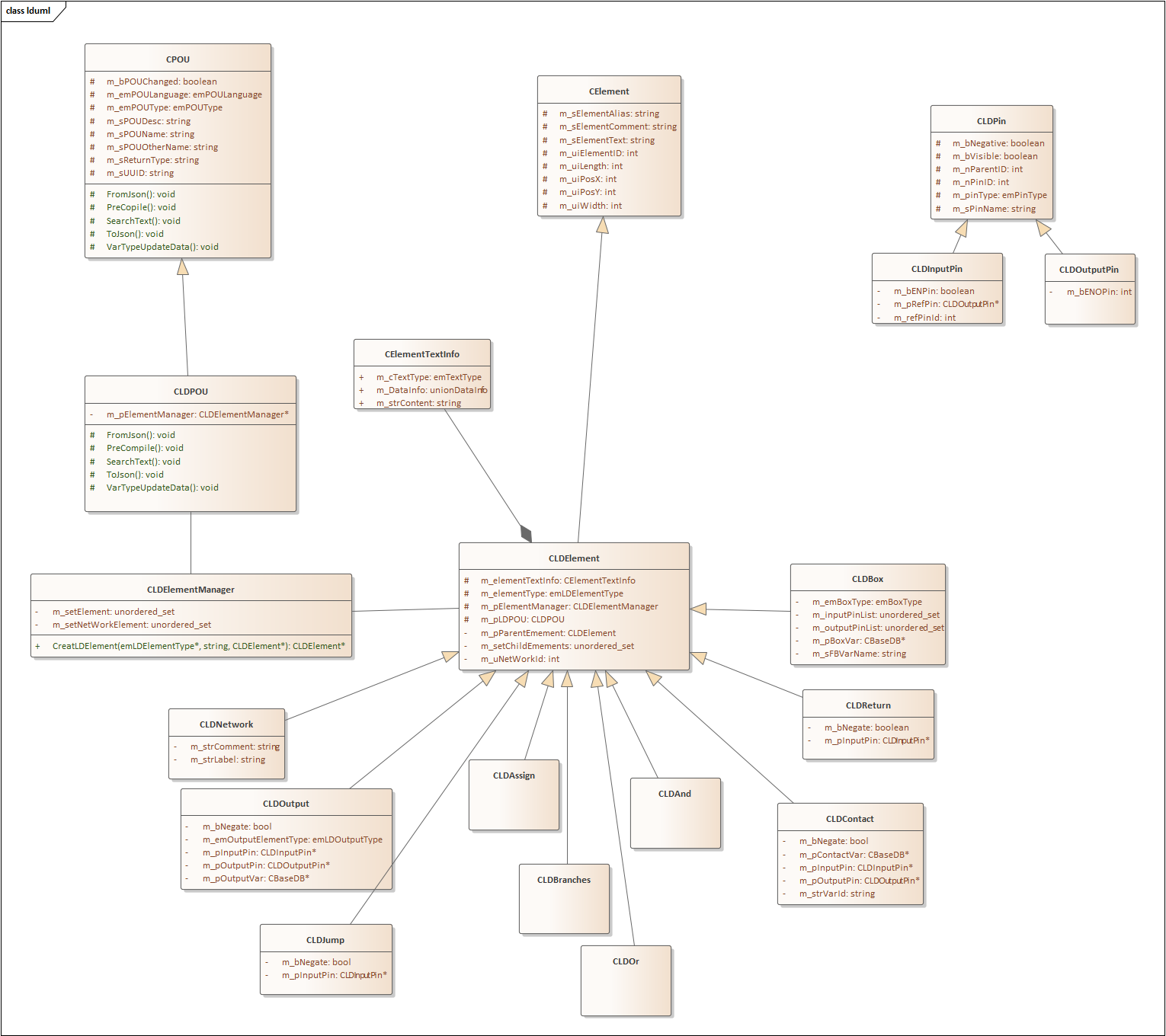




# 3类图实现

## 3.1 LD编辑器UML类图

## 3.2 LD Mode UML类图



# 4数据格式

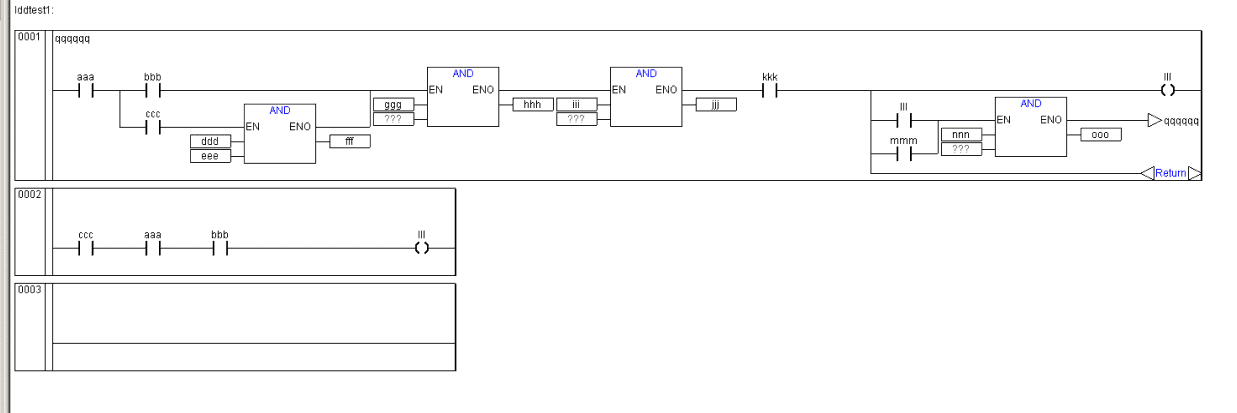
## 4.1节点：

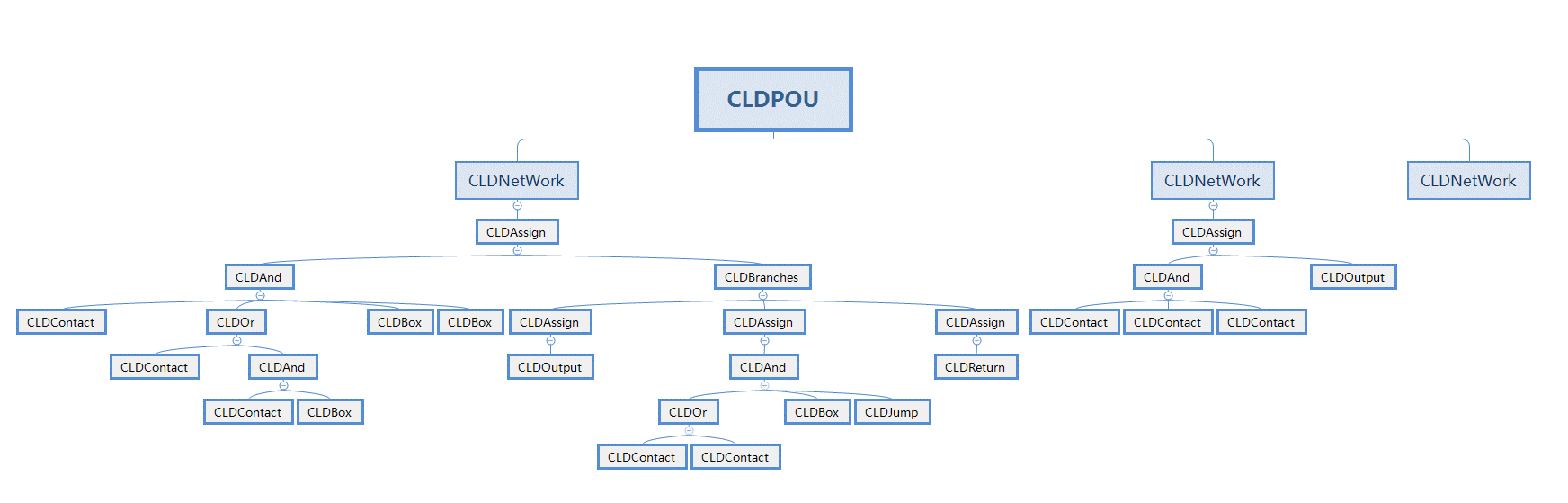
1. Comment注释
2. Label 标签---标准没有
3. Block 块
4. Jump 跳转
5. Return 返回
6. LeftPowerRail 左电源轨线---目前没有支持
7. RightPowerRail 右电源轨线---目前没有支持
8. Coil 线圈
9. Contact 触点
10. CompareContact 比较触点---目前没有支持，>,>=,=,<,<=,!=
11. Rung 网络节点《横档》
12. --Connector 添加了输入引脚的标签---需要讨论是否支持
13. --Continuation 添加了输出引脚的标签---需要讨论是否支持
14. ---ActionBlocks 复杂类型“动作块”表示“动作块”或“串联动作块”---需要讨论是否支持

## 4.2M7 json、M6xml及标准数据存储格式，附件。



## 4.3LD编辑器图形元素存储结构





# 5数据序列化反序列化方案

## 5.1方案一

参照目前M6存储数据格式，序列化反序列化数据。

优点：数据结构贴近业务，容易实现。

缺点：和IEC-10数据标准不一致，需要导入导出工具做中间层业务适配。

## 5.2方案二（采用）

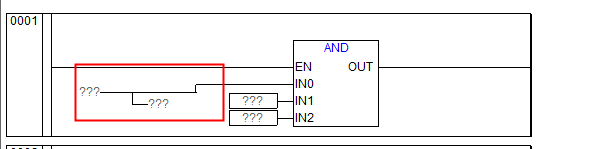
参照IEC-10标准扁平化数据格式，序列化反序列化数据。

优点：数据格式对标，不需要中间层做业务适配，只需扁平化解出节点。

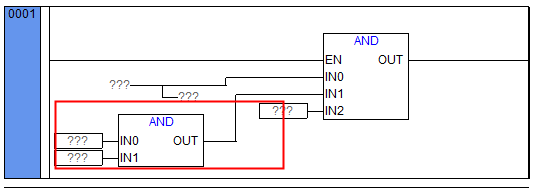
缺点：需要根据数据结构，反推逻辑树，需要开发此功能，相当于中间适配层下沉。

# 功能支持点

## 6.1M6支持赋值元件



## 6.2M6支持左边块连接



## 6.3支持直接连线

