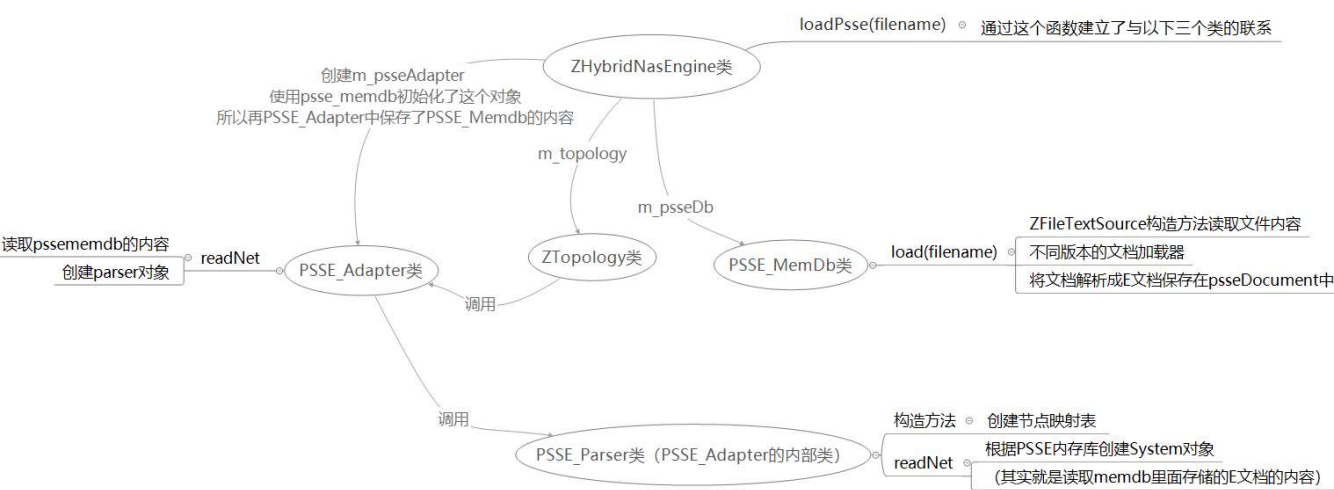


PSS/E文件阅读笔记

系统中读写PSSE文件的总流程



.raw文件的解析

psse_memdb中的parseRaw函数，分为十八部分数据（算例，母线，负荷，固定补偿器，发电机，非变压器支路，变压器，区域，双端直流输电线路，vsc直流传输线，变压器阻抗校正，多端直流线路，多段线路组，区域间交换，拥有者，FACTS设备，可投切并联补偿器）进行解析。

最根本的数据解析方法（函数readTable实现）是 **顺序从文件中读取行，通过数据长度来分割不同类型的数据 (textSource.readLine(m_buffer, sizeof(m_buffer)))**

算例 是处理文件的前三行数据，读取文件的基本信息。

非变压器支路，变压器，双端直流输电线路，vsc直流传输线 根据数据的特殊性做了特殊的数据筛选处理。

其余的十三种数据均直接调用readTable实现。

raw文件的数据可以直接在PSSE32软件中阅读到。

.seq文件的解析

psse_memdb中的parseSeq函数，分为八部分数据（修改码，发电机正序参数，发电机负序参数，发电机零序参数，负荷负序参数，负荷零序参数，线路零序参数，线路零序互阻抗）进行解析。

最根本的数据解析方法（函数readTable实现）是 **顺序从文件中读取行，通过数据长度来分割不同类型的数据 (textSource.readLine(m_buffer, sizeof(m_buffer)))**

修改码 是处理文件的第一行数据，读取文件的基本信息。

其余的七种数据均直接调用readTable实现。

感觉seq的文件结构可能比raw要清晰得多？主要是数据种类少，并且没有筛选数据的步骤

适配到ZSystem

函数	system对象
system.setBaseMva(baseMva);	设置基准电压
system.addNode(pNode); system.setArea(pNode,areaName,areaName); system.setZone(pNode,zoneName,zoneName); system.setOwner(pNode,ownerName,ownerName);	添加节点信息 设定地区、分区、所有者
system.addBusbar(pBusbar);	添加母线
system.addGroundBranch(nodeId,g,b,0.0,0.0);	读取负荷表和固定补偿器表都是添加接地支路
system.addGen(pGen);	添加发电机
system.addBranch(pBranch);	读取Line线路表、变压器表 添加支路信息
system.addCap(pCap);	添加可投切并联补偿器