原子服务分类：

1. 首先分为两大类：更新类和非更新类
2. 非更新类分为：
   1. 查询类
   2. 校验类
   3. 计算类
3. 更新类分为：
   1. 锁定更新类，update和delete（原则上不允许使用delete）操作为锁定更新类，多个客户端可能同时发起对一个数据的更新
   2. 非锁定更新类，insert为非锁定更新类

应用层拦截业务层对服务的调用，进行如下处理：

1. 对调用顺序纠错，校验类及校验类所需的查询和计算类尽量前置
2. 对于需要先后依赖的，分成同步组。(1、2在编译时刻就做完)
3. 多个同步组之间并发执行，并发组执行都返回后，再执行update类服务
4. 校验类服务返回false的，直接记录校验错误信息，退出scenario。
5. 如果没有事务block，所有update并发执行
6. 如果有事务block，执行并发事务处理，嵌套的小事务全部展开一并处理
7. 事务处理器内部决定采用哪种事务模式
8. Scenario只有两种返回，成功和失败，由于事务处理器在类Saga模式下可能会出现调用服务不响应的问题，此类都定义为失败。
9. 所有分布式事务，都不能保证强一致性，除分区容错性外，要么选择一致性牺牲可用性(CP)；要么选择可用性牺牲一致性(AP)。事务处理器应同时提供两种选择

执行顺序和分组原则：

1. 缺省顺序为查询、计算、校验，最后更新。能自动判别依赖的前提是：
   1. 只能单向依赖
   2. 同级不能依赖
2. 除更新外，前三类可能并行运算，有依赖关系的组成一个小组，多个小组之间无依赖关系可以并行运算。
3. 分组原则为，如果计算A的参数与查询B的参数有交集，则B依赖A；同理如果校验A和计算B有交集，则A依赖B。计算可能依赖查询和计算、校验可能依赖查询、计算，校验不能依赖于校验。依赖组可以嵌套
4. 支持人工设置依赖组

校验规则：校验规则为返回true继续执行，否则中断；不进行复杂的逻辑组合