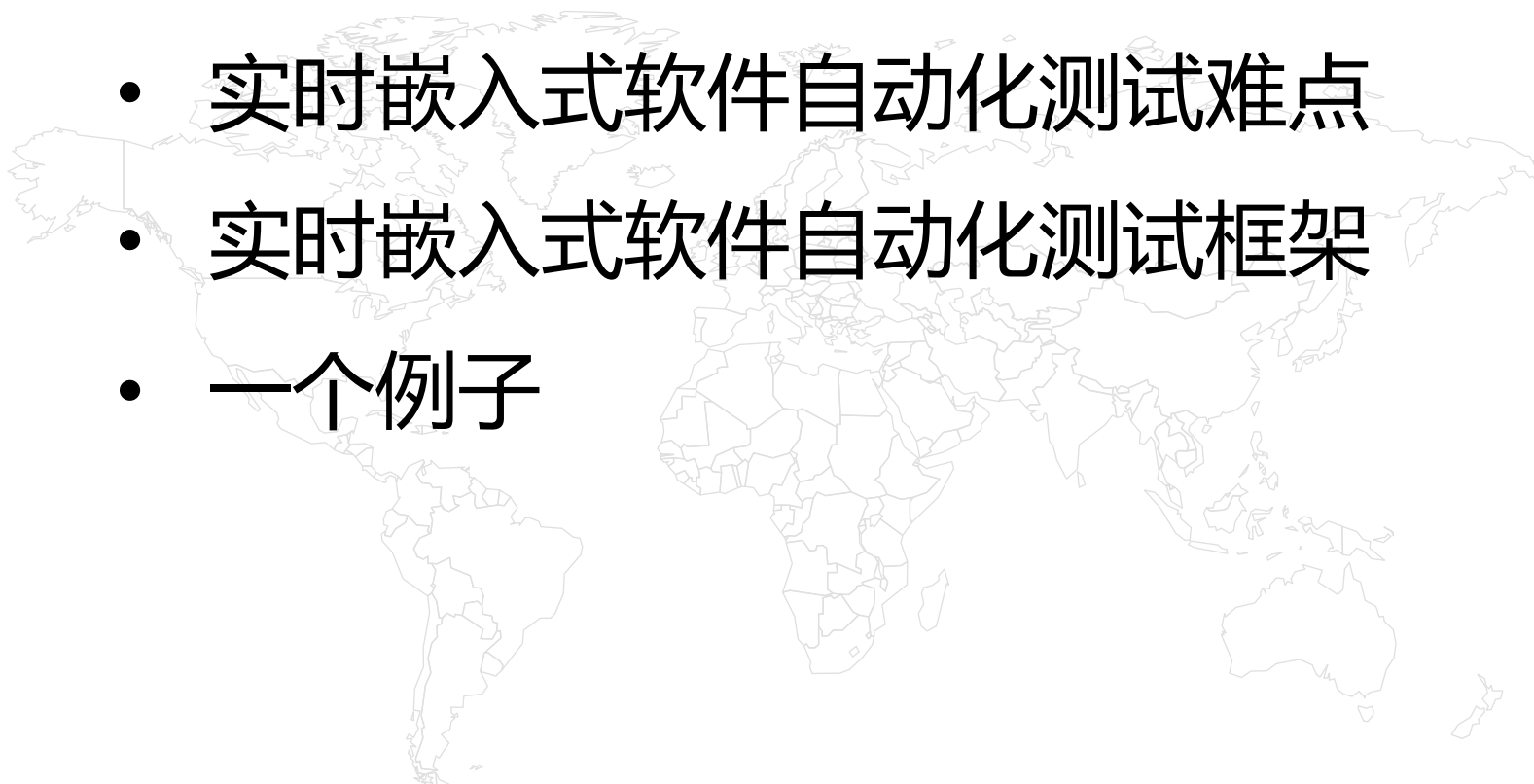


# 实时嵌入式软件自动化测试实践



# • 目录

- 
- 实时嵌入式软件自动化测试难点
  - 实时嵌入式软件自动化测试框架
  - 一个例子

# ● 实时嵌入式软件自动化测试难点

① 测试场景复杂，从仿真系统的角度看,实时嵌入式软件仿真测试平台实际上是一种面向实时嵌入式软件测试的半实物仿真系统。因此，在测试场景中需仿真大量设备，并能通过测试脚本，精确控制这些仿真设备的行为。

# ● 实时嵌入式软件自动化测试难点

② 测试激励的实时性与时序性，当采用黑盒测试的方式（激励-反馈机制）对城轨嵌入式软件自动化测试时，首先，被测系统需要实时获取和处理外部激励数据，测试平台也需要实时获取和分析被测对象的反馈数据；其次，测试平台需确保对于相同的测试用例，每一次进行测试执行的过程中，其产生的测试激励数据在时序关系上是完全一致的。

# ● 实时嵌入式软件自动化测试难点

③ 测试结果处理困难，一方面，需要在被测对象中合理的嵌入测试代理模块捕获被测对象的测试结果；另一方面，测试平台需在线或离线分析这些测试结果数据。

# ● 实时嵌入式软件测试自动化测试框架

测试框架应该解决的以下几方面的问题：

- ① 自动化测试框架应能提供基于业务描述的脚本，使得测试人员在编写测试用例时，专注业务需求而不必关心具体的测试驱动细节；
- ② 自动化测试框架提供了测试用例管理功能，使得测试用例在整个测试生命周期中可以复用；
- ③ 自动化测试框架提供了测试结果分析功能，在复杂场景的测试用例中，该功能可以显著的提高测试效率。

# ● 实时嵌入式软件测试自动化测试框架

测试框架符合以下的设计原则：

① 测试框架的集成应基于统一开放的标准，具有良好的通用性、松耦合性、开放性和可扩展性，确保框架中的子模块的实现不局限与特定的开发语言和技术，并且当子模块进行修改或重构时，整个框架保持稳定；

# ● 实时嵌入式软件测试自动化测试框架

测试框架符合以下的设计原则：

② 测试数据的管理基于统一的数据格式，子模块能透明的提交和获取测试数据进行处理；



# ● 实时嵌入式软件测试自动化测试框架

测试框架符合以下的设计原则：

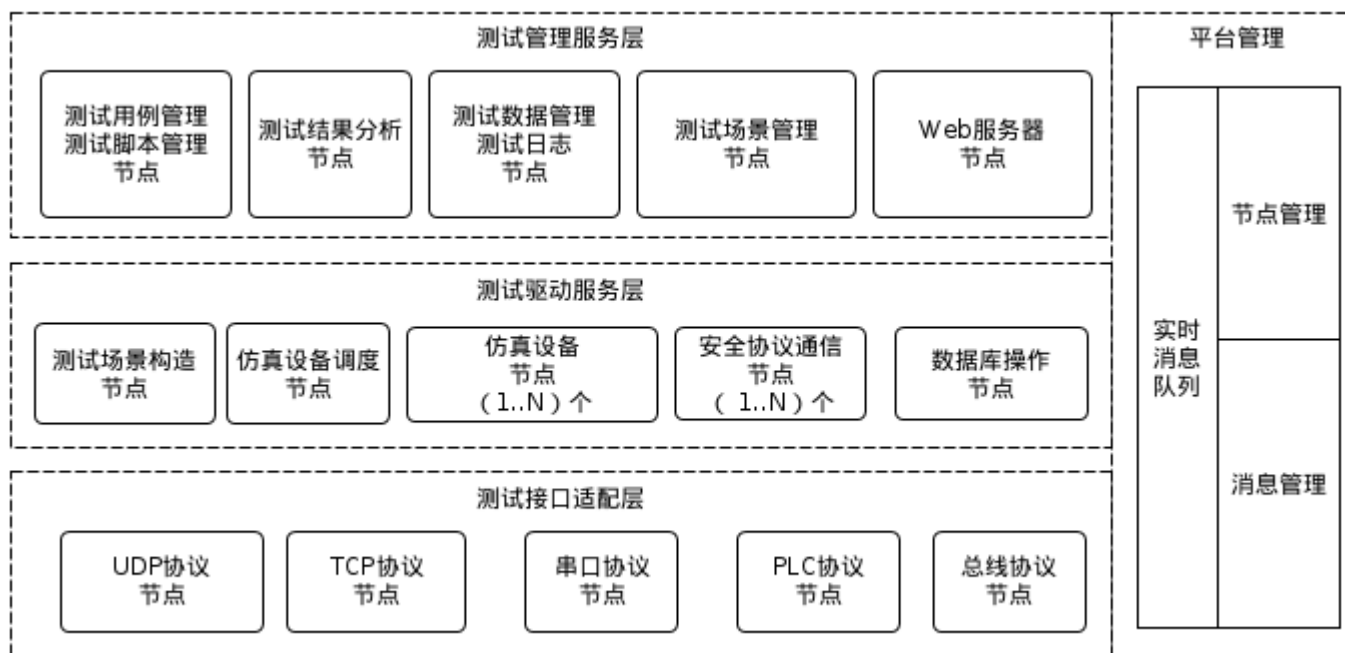
③ 实时性，为了确保对被测系统激励的实时性，测试框架在架构上应确保消息在平台内部能实时的处理和传递；

# ● 实时嵌入式软件测试自动化测试框架

测试框架符合以下的设计原则：

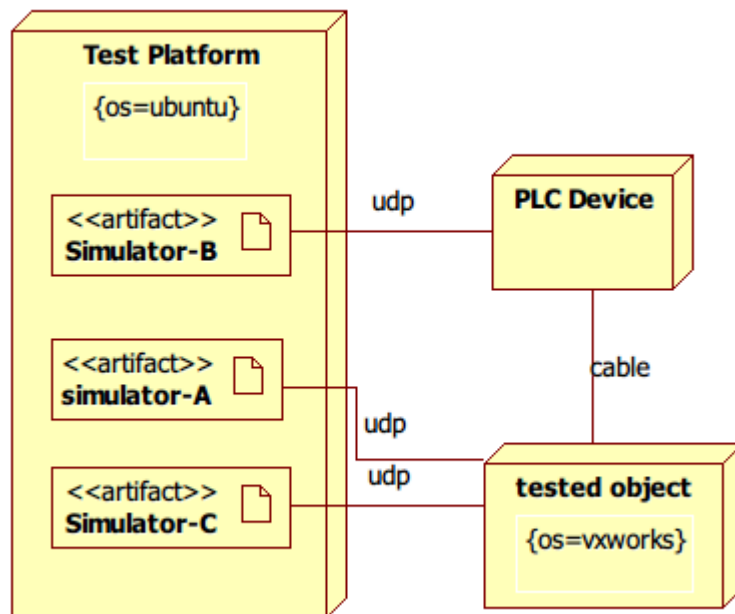
④ 大容量和高性能，为了满足城轨软件大容量数据测试的要求，测试框架应采取分布式的系统架构，在提高仿真设备数量时，并不影响测试平台性能；

# 实时嵌入式软件测试自动化测试框架



# • 一个例子

- ① 设备A，基于UDP消息的测试激励设备
- ② 设备B，基于IO的测试激励设备
- ③ 设备C，基于UDP的日志获取和分析设备



# ● 一个例子

## ① 设备A的组件脚本

|      |     |           |                                           |             |            |  |
|------|-----|-----------|-------------------------------------------|-------------|------------|--|
| Item | udp | UDPServer | /setting/simulator/simulator_networks.xml | startServer | stopServer |  |
|------|-----|-----------|-------------------------------------------|-------------|------------|--|

## ② 设备A的用例脚本

```
<Time Loophour="11" Description="">
  <Set Origin="Variant" Name="Varint_1" Value="22"/>
</Time>
```

# • 一个例子

## ③ 设备C的用例脚本

```
<Time Loophour="12" Description="">  
  <Set Origin="Variant" Name="Trace_Var" Value="Varint_1"/>  
</Time>
```

```
_case.setKeyName(r'Varint_1:\w*')  
_case.setEqualRule(13, Varint_1='22')
```

谢谢观看

