# 系统的应用模式

软件测试工具主要用于电子信息系统软件装备试验和软件测评任务，其应用模式如图 7‑1所示，主要包括软件装备试验和软件测评两种主要应用模式。软件装备试验根据被测对象的规模又可分为单软件装备试验、单系统软件装备试验和分布式软件装备试验，其中分布式系统软件装备试验最为复杂，试验以研制总要求为依据，按照规定的使用环境开展试验。软件测评主要分为黑盒测试和白盒测试，黑盒测试包括功能测试、性能测试、接口测试等，白盒测试包括静态分析、代码审查和逻辑测试，测试的主要依据是研制总要求和软件需求规格说明，测试的粒度较软件装备试验更小一些。



图 7‑1软件测试工具应用模式分析

不管是软件装备试验还是软件测评，开展工作的难点包括以下几个方面：一是测试环境开始部署比较复杂、繁琐；二是测试设计和测试执行工作量较大；三是需要大量的测试工具支撑；四是测试用例复用困难。软件测试工具的研制要支持软件装备试验和软件测评两种模式，在测试环境开设部署、测试设计和测试执行自动化、测试工具支持和测试复用等方面提供有效的解决方案。基于此，主要以分布式信息系统软件装备试验应用模式和电子信息装备软件测评应用模式作为典型应用模式进行分析。

## 分布式信息系统软件装备试验应用模式

分布式信息系统软件装备试验是软件装备试验最为复杂的一种应用场景，具有节点多，组网复杂，应用模式灵活等特点，主要以研制总要求为试验依据，软件测评的黑盒测试与软件装备试验的测试模式类似，可作为参考。

分布式信息系统软件装备试验典型试验场景如图 7‑2示，试验中需要部署多个或多级指挥部/指挥所，每个指挥部/所按需设置若干机构/部位，每个机构/部位分别部署所需的席位，每个席位按业务需要部署不同的被测软件，各个席位所需的服务由服务器集群按应用服务和数据库服务部署，试验需要的模拟数据，如空情、海情信息由实际陪测设备产生或者由各类模拟器产生。上下级信息系统或同级信息系统通过网络互联，按需发送、接收、处理所需信息。试验中测试人员根据研制总要求设计和执行测试项，按步骤执行试验任务，记录试验结果。

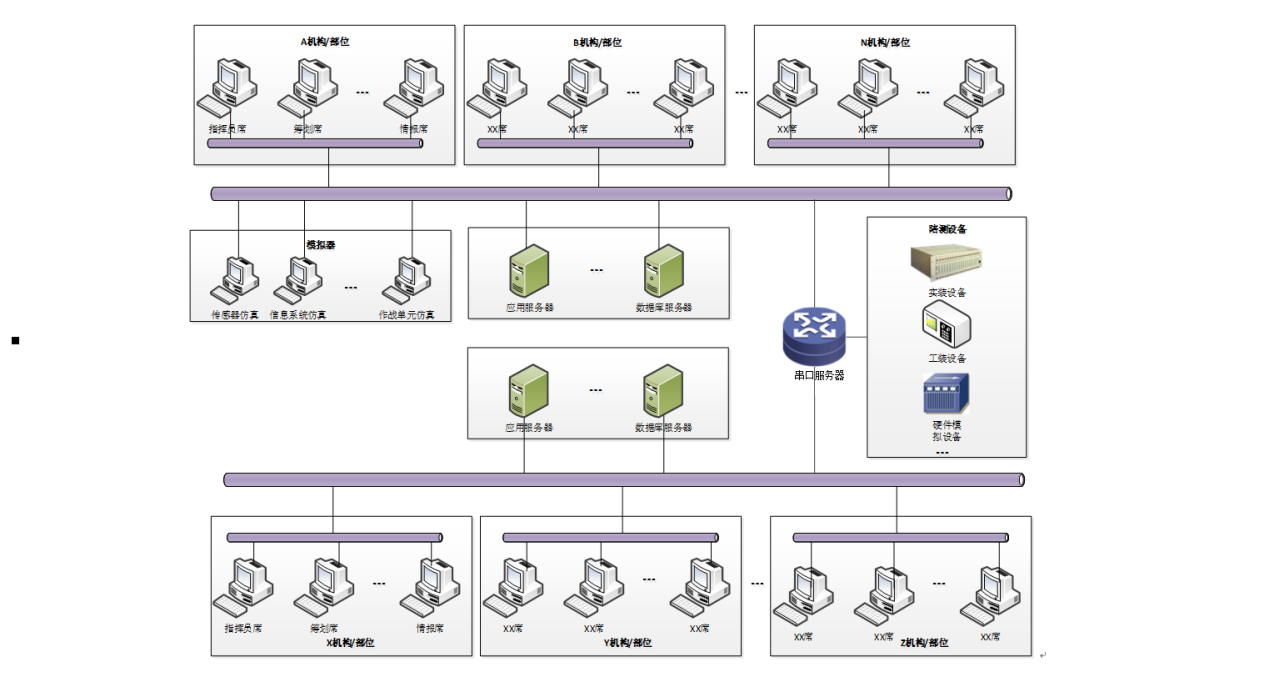


图 7‑2 分布式信息系统软件装备试验场景

为提供分布式信息系统软件装备试验所需的测试环境，软件测试云平台提供基于需求的精确计算资源、软件可定义的网络拓扑结构、模式可重用的软件测试环境，其试验环境构建如图 7‑3所示。



图 7‑3 分布式信息系统软件装备试验应用模式

软件测试云平台通过云计算技术形成计算、存储、网络等虚拟资源池，根据软件装备研制总要求和软件实际运行环境，设计试验环境规划，包括指挥部/所结构关系、服务器和席位设置，并申请计算、存储等运行资源，按需在各个节点部署信息系统软件、云化软件测试工具等；根据装备软件运行模式和组网需求申请网络资源，依据信息系统软件组网模式软件定义网络拓扑结构；实现仿真试验系统的云化部署，作为试验配试资源，提供传感器、信息系统节点仿真等，可通过外部网络接入测试云平台内部，形成外部真实装备、云化仿真试验系统、云化测试工具与被测信息系统软件一体集成的综合测试环境。试验过程和结果由数据采集与分析分系统集中存储，并可基于大数据技术进行测试数据的分析与评估。

使用软件测试工具开展分布式信息系统软件装备试验的流程如图 7‑4所示，主要包括启动试验任务、试验测试任务规划、测试环境规划、测试需求分析与测试设计、测试环境开设部署、测试任务分配与调度、测试任务执行、测试记录、测试监视与控制、测试数据分析、测试总结与评价、清理测试环境等。

其中，测试环境规划的主要任务包括规划指挥机构，机构与机构之间的连接关系；规划机构间网络连接关系；规划指挥机构信息系统构成，包括内部部门服务器、席位等；设计服务器参数，包括计算需求、存储需求、操作系统、基础软件、被测软件、网络参数等；规划席位端机，规划计算需求、存储需求、操作系统、基础软件、网络参数等；规划资源节点配置，包括传感器、信息系统、作战单元等各类模拟器等。

环境开设部署，以虚拟资源的方式按照测试环境规划和已有资源情况，开设虚拟服务器、虚拟客户端、虚拟网络，并部署所需的操作系统、数据库管理软件、基础的软件段和数据段，以及被测软件。



图 7‑4 分布式信息系统软件装备试验流程

测试需求分析与测试设计，提供测试需求和测试设计工具，按需求设计试验测试项、编写测试脚本和测试用例。

测试试验执行，提供测试调度能力，以人工与自动化相结合的方法执行试验和测试，对于自动化测试，提供云化的功能测试、性能测试和接口测试工具支持。

测试数据采集与分析记录与测试总结评估，试验测试中设计的测试项、测试用例和测试脚本自动记录到相应的数据库中，测试执行过程中产生的结果，按类别记录到相应的数据库和数据文件中，提供数据分析与评估方法。

测试环境撤收，试验结束后，整理保存数据，回收分配计算资源、存储资源和网络资源，恢复初始设置。

## 电子信息装备软件测评应用模式

在电子信息装备软件测评应用模式下，软件云测试系统主要完成软件测试工具的云化应用，针对测试工具的可云化程度，主要分为三种应用方式：多用户并行使用方式、多用户分时使用方式、单用户登陆使用方式，如图 7‑5。

软件云测试系统共包括Etest、UFT、LoadRunner、Parasoft、LDRA Testbed、TCS、KlocWork、X-DBCheck、DT10、Questa Prime等10种测试工具，其中：

完全云化测试工具包括：X-DBCheck，可以实现多用户并行使用；

竞争管理浮动License云化测试工具包括： UFT、LoadRunner、Parasoft、LDRA Testbed、TCS、KlocWork、DT10 、Etest，可以实现多用户分时使用；

登陆管理不支持云化测试工具包括：Questa Prime，可以实现单用户登陆使用。



图 7‑5 电子信息装备软件测评应用模式

### 完全云化测试工具使用

完全云化测试工具包括：X-DBCheck，可以实现多用户并行使用。

在完全云化测试工具X-DBCheck的使用中，用户通过多线程方式实现多用户并发使用，当一个用户对某数据进行数据规范检查时，给该数据表加入悲观锁的方式，实现对该数据的检查保护。

**（1）多线程并发机制实现多用户并发使用**

当一个用户申请X-DBCheck测试工具使用的时候，系统会为用户分配一个X-DBCheck测试工具线程，多用户申请使用的时候，会为每个用户分配一个X-DBCheck测试工具线程，实现多用户多线程并发使用。

以下为多线程工作示意图如图 7‑6。

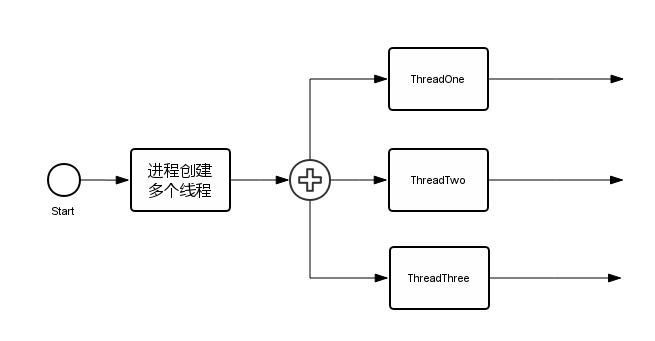
[](http://abner-nimengbo.cn/images/%E5%A4%9A%E7%BA%BF%E7%A8%8B.png)

图 7‑6 多线程工作示意图

**（2）悲观锁机制实现对数据库检查的同步与互斥**

当多用户同时使用X-DBCheck对数据库数据进行数据规范检查时，使用悲观锁机制，当一个用户使用X-DBCheck对数据库某个数据表进行规范检查时，会对该数据表加入悲观锁，它指的是对数据被外界（包括本系统当前的其他[事务](https://baike.baidu.com/item/%E4%BA%8B%E5%8A%A1)，以及来自外部系统的[事务处理](https://baike.baidu.com/item/%E4%BA%8B%E5%8A%A1%E5%A4%84%E7%90%86/217482)）修改持保守态度，因此，在整个数据处理过程中，将数据处于锁定状态。

悲观锁具有强烈的独占和排他特性。它指的是对数据被外界（包括本系统当前的其他[事务](https://baike.baidu.com/item/%E4%BA%8B%E5%8A%A1)，以及来自外部系统的[事务处理](https://baike.baidu.com/item/%E4%BA%8B%E5%8A%A1%E5%A4%84%E7%90%86/217482)）修改持保守态度，因此，在整个数据处理过程中，将数据处于锁定状态。悲观锁的实现依靠数据库提供的锁机制（也只有数据库层提供的锁机制才能真正保证数据访问的[排他性](https://baike.baidu.com/item/%E6%8E%92%E4%BB%96%E6%80%A7/2137580)，否则，即使在本系统中实现了加锁机制，也无法保证外部系统不会修改数据）。

乐观锁（ Optimistic Locking ） 相对[悲观锁](https://baike.baidu.com/item/%E6%82%B2%E8%A7%82%E9%94%81)而言，乐观锁机制采取了更加宽松的加锁机制。乐观锁，大多是基于数据版本（ Version ）记录机制实现。即为数据增加一个版本标识，在基于数据库表的版本解决方案中，一般是通过为数据库表增加一个 “version” 字段来实现。读取出数据时，将此版本号一同读出，之后更新时，对此版本号加一。此时，将提交数据的版本数据与数据库表对应记录的当前版本信息进行比对，如果提交的数据版本号大于数据库表当前版本号，则予以更新，否则认为是过期数据。

悲观锁与乐观锁使用原理图如图 7‑7。

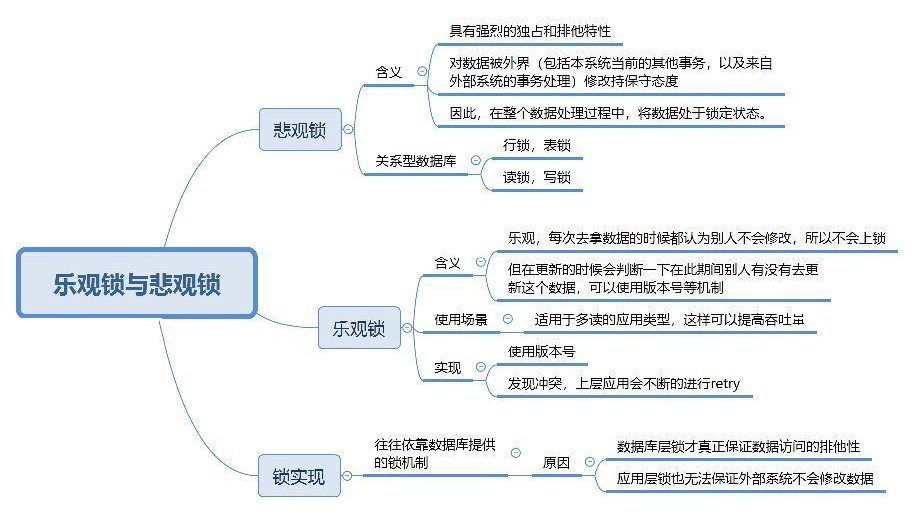


图 7‑7 悲观锁与乐观锁原理图

X-DBCheck测试工具数据规范检查流程图如图 7‑8。



图 7‑8 完全云化测试工具使用流程

完全云化测试工具X-DBCheck使用流程：

1.用户执行数据规范检查测试用例，申请X-DBCheck测试工具使用权限；

2.系统为用户分配X-DBCheck测试工具线程；

3.系统通过调度模块获取用例执行脚本；

4.检查数据库被检查数据是否被锁，如果被锁说明有其它用户正在对该数据进行检查，需要排队等待，如果没有被锁，则可以对该数据进行规范检查，同时对该数据加入悲观锁进行保护；

5.检查结果进行分析后存入到数据库；

6.测试结束，取消悲观锁，释放X-DBCheck执行线程。

### 竞争管理浮动License云化测试工具使用

竞争管理浮动License云化测试工具包括： UFT、LoadRunner、Parasoft、LDRA Testbed、TCS、KlocWork、DT10 、Etest，可以实现多用户分时使用。

在竞争管理浮动License云化模式下，用户通过竞争机制分时使用License,解决方案具体功能包括：

许可证集中监控报告（LicReport）：提供实时监控各个测试工具许可证使用情况的功能，包括测试工具名称、测试工具授权方式、安装方式、云化集成方式、License数量，使用License数量等。



图 7‑ 测试工具使用状态监视图

许可证动态释放(LicRecycle）：闲置状态识别；动态释放/激活；动态回收许可证，允许用户保存当前作业数据；智能回收占而不用的许可证；

许可证调度（LicDispatch）：设置时间内为用户预留具体功能模块及数量；许可证可分组分类设置；手动释放许可证；服务器许可证没使用完时客户端能自动续上；

许可证授信控制（LicControl）：仅仅是受管理员允许的用户才可以使用许可证，保证许可证不被非授权用户使用；控制同一用户同时仅能够打开一个功能模块；控制用户使用的许可证数量不超过管理员允许的数量；针对商务约定了许可证数量但技术上不控制的事后审计计费的模式，设定许可数量，防止产生昂贵的非预售费用。

Vlic：一项许可证高效的调度技术，将浮点许可证再次浮动使用，改变目前许可证一旦被某个用户占有则一直占有的现状，可以实现许可证在超过被应许范围一定比例的情况下使用。



图 7‑10 竞争管理浮动License云化测试工具使用流程

竞争管理浮动License云化测试工具使用流程如图 7‑10。

竞争管理浮动License云化测试工具使用流程：

1.用户执行测试用例，申请测试工具License使用权限；

2.系统检查改测试工具License的使用状态，如License没有空余，则排队等待，如果License空余，则分配License，系统通过云适配器获取测试工具使用权限；

3.系统通过调度模块获取测试用例执行脚本；

4.测试工具执行测试，执行结束后，测试结果存储到数据库；

5.释放License，测试工具License使用状态更新。

### 单用户登陆测试工具使用

登陆管理不支持云化测试工具包括：Questa Prime，可以实现单用户登陆使用。

在单用户登录使用模式系，用户通过自旋锁的方式实现对Questa Prime的使用。自旋锁是为实现保护[共享资源](https://baike.baidu.com/item/%E5%85%B1%E4%BA%AB%E8%B5%84%E6%BA%90)而提出一种锁机制，在任何时刻最多只能有一个用户获得锁。如果Questa Prime测试工具已经被用户占用，系统会自动给Questa Prime测试工具加入自旋锁，当其它用户想调用Questa Prime测试工具时，调用者就一直循环在那里看是否该自旋锁的保持者已经释放了锁，直到该自旋锁释放。

登陆管理不支持云化测试工具Questa Prime使用流程如图 7‑11。



图 7‑ 单用户登录测试工具使用流程

单用户登录测试工具使用流程：

1.用户执行测试用例，申请测试工具Questa Prime使用权限；

2.系统检查改测试工具Questa Prime的使用状态，如Questa Prime被锁定，则排队等待，如果未被锁定，则用户获取Questa Prime使用权限，同时给Questa Prime加入自旋锁进行保护；

3.系统通过调度模块获取测试用例执行脚本；

4.测试工具执行测试，执行结束后，测试结果存储到数据库；

5.释放自旋锁，测试工具锁定状态更新。

### 电子信息装备软件测评过程应用

电子信息装备软件测评环境图如下：



图 7‑12 电子信息装备软件测评环境图

电子信息装备软件测评测试需求分析、测试策划、测试设计与实现、测试执行各个阶段使用模式设计如下。

#### 测试需求分析

系统接收到被测件后，首先听过测试过程管理服务，对被测件进行配置管理，存储到测试资产库入库。

辅助设计服务模块，接收被测件文档就，对被测文档进行分析，提取出测试项需求、测试环境需求、测试工具需求、测试数据需求等，测试项上传测试过程管理服务模块进行测试项管理，测试环境需求、测试工具需求、测试数据需求存储到测试资产库。

测试需求分析数据流图如下：



图 7‑13 测试需求分析数据流图

#### 测试策划

测试策划主要工作是生成测评大纲和测试环境搭建，测试过程管理服务模块接收测试资产库存储的测试需求项、测试环境需求、测试数据需求、测试工具需求和人工完善项，生成软件测评大纲。

生成软件测评大纲数据流图如下：



图 7‑14 生成测评大纲数据流图

用户通过测试环境设计模块，设计网络拓扑结构，在网络拓扑结构图中选中虚拟机，可以添加虚拟机的硬件配置，包括CPU、内存、存储等配置，和软件配置，包括操作系统、支撑软件、测试工具等配置，测试设计模块根据用户设计内容生成测试环境规划设计方案。

测试环境定制服务模块根据测试环境规划设计方案，能自动生成用户需要的测试环境。

测试环境搭建数据流图如下：



图 7‑15 测试环境搭建数据流图

测试环境自动生成界面图如下：

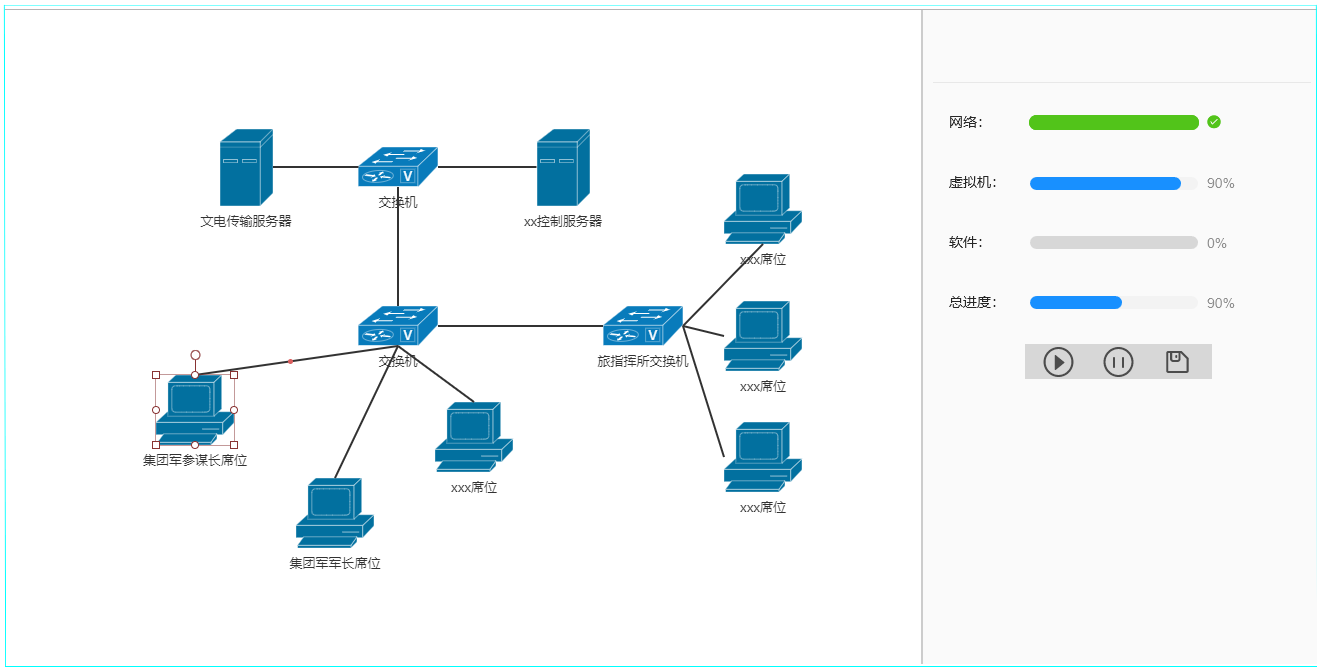


图 7‑16 测试环境自动生成界面

#### 测试设计与实现

此阶段工作主要是测试用例设计和测试用例中脚本的录制。

**1.测试用例设计**

测试管理系统自主开发，完全云化，用户可通过虚拟机客户端登录浏览器在线设计测试用例，测试用例输入数据可通过TCS工具辅助设计，设计好的测试自动存入测试管理系统。

**2.测试用例脚本录制**

测试用例设计完成后，黑盒测试中，通过云平台调用UFT、LoadRunner等工具录制测试用例执行脚本，录制完成后，脚本和测试工具信息存入测试管理系统。

对于白盒测试用例，在测试用例中，加入对应的软件源代码，同时加入对应的测试工具信息。

黑盒测试用例设计数据图如下：



图 7‑17 黑盒测试用例设计数据流图

白盒测试用例数据流图如下：



图 7‑18 白盒测试用例设计数据流图

#### 测试执行阶段

测试执行阶段，当执行黑盒测试和白盒测试时，系统执行测试用例脚本时，根据测试工具信息和脚本后缀名自动判断测试工具，调取对应的测试工具执行脚本，测试执行结果自动存入测试管理系统。

（1）完全云化测试工具执行

当系统执行数据标准规范检查测试用例时，系统检查X-DBCheck的使用状态，当X-DBCheck使用状态为不被占用可以使用时，系统通过云适配器调用X-DBCheck，对测试用例中规定的数据进行检查，同时对检查的数据加入悲观锁进行保护，检查结果经过分析后，释放悲观锁，检查结果和分析结果自动存入到数据库。

X-DBCheck测试执行数据流图如下：



图 7‑19X-DBCheck测试工具执行数据流图

（2）竞争管理浮动License云化测试工具执行

当系统调用ETest、UFT、LoadRunner、Parasoft、LDRA Testbed、KlockWork、DT10时，系统会根据先检查该测试工具的License使用状态，当该测试工具License处于空余状态时，系统通过云适配器调用该测试工具，系统通过调度服务测试用例中的执行脚本或源代码，进行规定的测试，测试执行结果经过测试数据采集分析后，测试执行结果和分析结果自动存入到数据库。

以黑盒测试工具UFT为例，测试执行数据流图如下：



图 7‑20 UFT测试工具执行数据流图

以白盒测试工具TestBed为例，测试执行数据流图如下：



图 7‑21 TestBed测试工具执行数据流图

（3）单用户登录测试工具使用执行

当系统调用Questa Prime执行FPGA程序测试时，系统会根据先该的锁定状态，当Questa Prime处于未被锁定时，系统单机调用该测试工具，根据测试用例内容进行规定的测试，测试执行结果经过分析后，测试执行结果和分析结果自动存入到数据库。

Questa Prime执行测试数据流图如下：



图 7‑22 Questa Prime测试工具执行数据流图