**系统测试管理规程**

**XZSKC-30**

**编写： 张珂 日期： 2022/2/8**

**审核： 朱绍攀 日期： 2022/2/17**

**批准： 朱绍攀 日期： 2022/2/17**

**徐州市勘察测绘研究院有限公司**

**文档修订记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 版本号 | \*变化  状态 | 简要说明 | 日期 | 变更人 | 批准日期 | 批准人 |
| V1.00 | C | 初始创建 | 2022/2/8 | 张珂 | 2022/2/17 | 朱绍攀 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

\*变化状态：C = 创立，A = 增加，M = 修改，D = 删除

目录

[1 简介 1](#_Toc73302044)

[1.1 目的 1](#_Toc73302045)

[1.2 适用范围与预期读者 1](#_Toc73302046)

[1.3 术语表 1](#_Toc73302047)

[1.4 参考资料 2](#_Toc73302048)

[2 过程总体描述 2](#_Toc73302049)

[2.1 过程概述 2](#_Toc73302050)

[2.2 过程流程图 3](#_Toc73302051)

[3 过程元素描述 3](#_Toc73302052)

[3.1 制定软件测试计划 3](#_Toc73302053)

[3.2 单元测试 6](#_Toc73302054)

[3.3 系统测试 7](#_Toc73302055)

[3.4 验收测试 9](#_Toc73302056)

[4 测试内容 10](#_Toc73302057)

[4.1 数据和数据库完整性测试 10](#_Toc73302058)

[4.2 功能测试 11](#_Toc73302059)

[4.3 业务周期测试 12](#_Toc73302060)

[4.4 用户界面测试 13](#_Toc73302061)

[4.5 性能评测 13](#_Toc73302062)

[4.6 负载测试 15](#_Toc73302063)

[4.7 强度测试 16](#_Toc73302064)

[4.8 容量测试 17](#_Toc73302065)

[4.9 安全性和访问控制测试 18](#_Toc73302066)

[4.10 故障转移和恢复测试 19](#_Toc73302067)

[4.11 配置测试 22](#_Toc73302068)

[4.12 安装测试 23](#_Toc73302069)

[5 活动裁剪 24](#_Toc73302070)

# 简介

## 目的

本文的目的是规范测试工作，为软件测试工作提供详细的指引。以发现错误为目的，提高组织内部软件测试的管理水平，确保组织中开发产品的质量。

软件测试确保最终交给用户的产品的功能符合用户的需求，把尽可能多的问题在产品交给用户之前发现并改正。具体地讲，测试一般要达到下列目标：

* 确保产品完成了它所承诺或公布的功能，并且所有用户可以访问到的功能都有明确的书面说明。
* 确保产品满足性能和效率的要求。
* 确保产品是健壮的和适应用户环境的。

## 适用范围与预期读者

适用于本公司的所有软件项目。

预期读者：1、部门领导和主管；2、项目组；3、质量保证人员。

## 术语表

* **驱动模块（Driver）**：在单元测试中，协调输入和输出的测试程序；
* **桩模块（Stub）：**在单元测试中，模拟被调用单元的测试程序；
* **测试计划：**包含项目范围内的测试目的和测试目标的有关信息，确定了实施和执行测试时使用的策略，确定了测试所需资源。
* **测试用例：**为特定目标开发的测试输入执行条件和预期结果的集合。

## 参考资料

# 过程总体描述

## 过程概述

软件测试流程贯穿了项目的整个开发和测试生命周期，与整个软件开发过程基本上是并行进行并相互协调的。

## 过程流程图



# 过程元素描述

## 制定软件测试计划



### 目的

根据批准的需求规格说明书和相关设计文档，确定项目测试阶段的目标和策略，确保测试工作有序、有效进行。

### 角色和职责

**项目经理：**与测试负责人一起批准测试计划

**测试人员：**负责编写测试的计划和测试用例

### 进入准则

* 客户需求已明确。

### 输入

* 《软件需求规格说明书》
* 《详细设计说明书》
* 《项目计划》

### 工作任务

1. 确定系统的测试需求，如功能需求、性能需求、安全性要求、可使用性需求等；
2. 测试人员与项目经理协商，逐步确定测试项目的测试范围、测试粒度（覆盖标准）以及测试方案、测试阶段的出入口准则；
3. 根据《项目计划》以及项目的复杂度初步估计测试项目工作量，制定测试计划的进度安排。逐步细化测试计划及测试规模估计；测试进度安排中要留有合理的测试缺陷、用例管理时间；
4. 形成系统测试计划并提交项目经理审核。批准人为项目经理。同时测试人员可发起测试计划的评审；审核批准通过则放入配置库；
5. 测试计划审核通过后，可以开始编写测试用例；
6. 用例设计：
   * + - 测试人员参与需求和设计的评审，正确理解系统需求并确认需求的可测性，获取测试项目需求；
       - 根据批准的测试项目需求（在测试计划中有测试需求的详细描述）、测试目标的逻辑实现和约束、测试工具及其测试环境等限制条件，设计测试用例。
       - 测试人员根据系统的软件需求，编写系统测试用例；
       - 测试人员发起组织相关人员进行测试用例评审，从而提高测试用例的质量；系统测试用例审核人可以是测试人员、项目经理，批准人为项目经理。
7. 用例管理：
   * + - 测试人员负责进行测试用例的实施、跟踪及用例统计分析工作、改进测试用例等管理活动；
       - 当软件需求或设计变更引起测试需求变更时，将变更测试用例文档；
       - 测试人员实时或定期根据缺陷数据、状态和测试用例执行情况进行分析，以确定是否需要对目前测试的模块设计新的测试用例，对不稳定的模块，测试人员负责与项目经理讨论确定测试范围、粒度和执行方案等，并新增测试用例；
       - 新增测试用例批准后由测试人员执行；
8. 当项目开发计划或测试需求发生变更时，测试计划应考虑是否需要变更。

### 退出准则

* 《系统测试计划》已批准。

### 输出

* 《系统测试计划》

## 单元测试



### 目的

使用测试用例及相应编码准则等，验证程序代码单元及其函数、接口已按照预设的方式（系统设计）调用执行，并产生合乎期待的结果。

### 角色和职责

**项目经理：**参与单元测试用例的编写

**开发人员：**编写单元测试用例，进行单元测试

### 进入准则

* 被测代码已完成

### 输入

* 《详细设计说明书》

### 工作任务

1. 在项目设计阶段，项目经理组织开发人员进行单元测试中测试用例的编写；
2. 开发人员在符合规定测试环境条件下，使用指定测试及管理工具，编码规范和单元测试用例，从配置库中提取标识代码模块实施测试活动；

* 静态测试：根据开发计划和测试计划安排，由项目经理指定人员依编码规范对单元模块代码进行同行评审，及时发现、记录并修订代码中存在的语法规范或逻辑错误；
* 动态测试（包括动态分析）：根据项目计划和测试计划安排，开发人员设计单元测试用例，编写驱动模块和桩模块，执行单元测试用例；

1. 执行单元测试，并在《单元测试用例》中记录是否通过，直到所有单元测试通过为止。

### 退出准则

* 单元测试用例全部测试通过。

### 输出

* 《单元测试用例》
* 《测试用例》

## 系统测试



### 目的

执行系统测试用例，验证已各通过各阶段测试的功能模块已具有满足需求规格说明所规定的功能、质量和性能等方面特性。

### 角色和职责

**测试人员：**执行测试、缺陷提交、跟踪验证、回归关闭缺陷；编写测试总结报告

### 进入准则

* 单元测试完成
* 系统集成完成，编译无误。

### 输入

* 被测系统
* 系统测试用例

### 工作任务

1. 单元测试完成后，可以进行系统测试；
2. 测试人员搭建测试环境，依据《产品集成计划》和《产品集成规程》从项目组配置库中提取集成单元并组装进行测试活动。脚本须及时放入配置库。对于由测试脚本产生的自动化测试用例，应该在测试用例文档自动测试脚本一栏标明配置库存放路径；
3. 测试实施全过程中，始终存在测试计划变更和测试用例变更以及缺陷管理过程。
4. 测试人员定期对系统测试质量及效果、进度情况进行评估，确定测试覆盖完整性，检验测试结果是否达到测试出口准则，见测试计划；
5. 当测试实施过程中，对于测试出的问题有争议时，由测试人员充分参考项目经理的意见，进行最后仲裁；若对于仲裁的结果仍然有争议时，应继续上报上级经理，直至问题得到解决；
6. 系统测试结束后，测试人员负责汇总、分析测试结果，形成《测试报告》提交评审。

### 退出准则

* 测试用例100%覆盖产品需求，测试用例100%执行，严重级别的缺陷全部修复，一般缺陷修复在80%以上。

### 输出

* 《测试报告》
* 《缺陷记录》

## 验收测试



### 目的

验收测试是一个以用户为主的测试，一般是在软件系统测试结束以及配置审计之后开始的，应由用户、测试人员、软件开发人员参与，验证软件系统的功能和性能及其他特性是否与用户的要求一致。

### 角色和职责

**项目经理：**组织实施验收测试，编写验收报告

**测试人员：**配合项目经理进行验收计划、用例的编写，配合客户进行验收测试

**配置管理人员：**发布最终产品

### 进入准则

* 软件系统测试结束以及配置审计之后。

### 输入

* 被测系统

### 工作任务

1、项目经理和客户代表一起确定产品验收小组并制定验收计划，体现在《项目计划》中项目验收部分以及《验收计划》。

2、建立试运行环境，根据验收计划安排验收人员进行验收。

3、验收小组依据测试用例，参考用户文档、需求规格说明书、详细设计说明书、产品集成计划，从配置库中集成被测试系统的目标代码或集成单元，执行测试用例。并将测试时发现的问题填写到《缺陷记录》中

4、由项目经理负责评估、分析问题原因，并指派人员进行必要的修订，直到所有发现问题得到解决；

5、由配置管理人员负责将经测试验收确认的最终版本进行封版，并发布基线。最后从发布基线集成最终发布产品纳入产品库，并交付给客户。

6、验收测试完成后，形成验收报告，由客户方签字确认。

### 退出准则

* 客户签字确认。

### 输出

* 《验收计划》
* 《验收报告》
* 《验收确认单》

# BUG定义

### 4.1 Bug、缺陷种类定义

测试BUG分类：轻微、一般、严重、致命

|  |  |
| --- | --- |
| 轻微 | 界面问题、提示信息、建议性、易用性、统一性（以开发技术规范中定义但是系统中未实现为标准） |
| 一般 | 性能问题、安全问题、校验问题、乱码 |
| 严重 | 功能问题、业务逻辑问题、数据控制、保存失败等，影响流程或功能实现的问题 |
| 致命 | 系统崩溃或电路板烧毁 |

### 4.2 测试用例覆盖率

测试覆盖率 = 测试通过需求数目/需求总数目 ×100％

### 4.3 测试需求覆盖度

需求覆盖率是指经过测试的需求/功能和需求规格说明书中所有需求功能的比值，通常情况下要达到100％的目标。

### 4.4 Bug密度

权重系数即换算标准：1个非常严重的BUG=8个普通的BUG；一个严重BUG=4个普通BUG； 一个轻微BUG=0.5个普通BUG。

BUG密度 = bug加权数量/被测功能点数\*100%

### 4.5缺陷（BUG）密度计算

|  |  |  |
| --- | --- | --- |
| **计算标准** | **计算方法** | **示例** |
| 代码行数 | 缺陷(BUG)个数/被测代码行数(千行) | 代码数为75千行，BUG数为9 缺陷密度= 9/75kloc =0.12bug/kloc |
| 功能点个数 | 缺陷(BUG)个数/被测功能点个数 | 总功能点个数为90，BUG数为9 缺陷密度= 9/90 =0.1 bug/功能点 |

# 测试内容

## 数据和数据库完整性测试

|  |  |
| --- | --- |
| 测试目标： | [确保数据库访问方法和进程正常运行，数据不会遭到损坏。] |
| 技术： | [调用各个数据库访问方法和进程，并在其中填充有效的和无效的数据（或对数据的请求）。  检查数据库，确保数据已按预期的方式填充，并且所有的数据库事件都已正常发生；或者检查所返回的数据，确保为正当的理由检索到了正确的数据] |
| 完成标准： | [所有的数据库访问方法和进程都按照设计的方式运行，数据没有遭到损坏。] |
| 需考虑的特殊事项： | [测试可能需要 DBMS 开发环境或驱动程序在数据库中直接输入或修改数据。  进程应该以手工方式调用。  应使用小型或最小的数据库（记录的数量有限）来使所有无法接受的事件具有更大的可视度。] |

## 功能测试

编写说明：对测试对象的功能测试应侧重于所有可直接追踪到业务功能和业务规则的测试需求。这种测试的目标是核实数据的接受、处理和查询是否正确，以及业务规则的实施是否恰当。此类测试基于黑盒技术，该技术通过图形用户界面 (GUI) 与应用程序进行交互，并对交互的输出或结果进行分析，以此来核实应用程序及其内部进程。以下为各种应用程序列出了推荐使用的测试概要：

|  |  |
| --- | --- |
| 测试目标： | [确保测试对象的功能正常，其中包括导航、数据输入、处理和检索等功能。] |
| 技术： | [利用**有效的和无效的**数据来执行各个功能点，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。] |
| 完成标准： | [所计划的测试已全部执行。  所发现的缺陷已全部解决。] |
| 需考虑的特殊事项： | [确定或说明那些将对功能测试的实施和执行造成影响的事项或因素（内部的或外部的）] |

## 业务周期测试

编写说明：业务周期测试应模拟在一段时间内对项目执行的活动。应先确定一个时间段（例如一年），然后执行将在该时间段（一年内）发生的事务和活动。这种测试包括所有的日、周和月周期，以及所有与日期相关的事件（如备忘录）。

|  |  |
| --- | --- |
| 测试目标 | [确保测试对象及背景的进程都按照所要求的业务模型和时间表正确运行。] |
| 技术： | [通过执行以下活动，测试将模拟若干个业务周期：  将修改或改进对测试对象进行的功能测试，以增加每项功能的执行次数，从而在指定的时间段内模拟若干个不同的用户。  将使用有效的和无效的数据或时间段来执行所有与时间或数据相关的功能。  将在适当的时间执行或启用所有周期性出现的功能。  在测试中还将使用有效的和无效的数据，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。 |
| 完成标准： | [所计划的测试已全部执行。  所发现的缺陷已全部解决。} |
| 需考虑的特殊事项： | [系统日期和事件可能需要特殊的支持活动  需要通过业务模型来确定相应的测试需求和测试过程。] |

## 用户界面测试

编写说明：用户界面 (UI) 测试用于核实用户与软件之间的交互。UI 测试的目标是确保用户界面会通过测试对象的功能来为用户提供相应的访问或浏览功能。另外，UI 测试还可确保 UI 中的对象按照预期的方式运行，并符合公司或行业的标准。

|  |  |
| --- | --- |
| 测试目标： | [核实以下内容：  通过测试对象进行的浏览可正确反映功能和需求，这种浏览包括窗口与窗口之间、字段与字段之间的浏览，以及各种访问方法（Tab 健、鼠标移动、和快捷键）的使用，窗口的对象和特征（例如，菜单、大小、位置、状态和中心）都符合标准。] |
| 技术： | [为每个窗口创建或修改测试，以核实各个应用程序窗口和对象都可正确地进行浏览，并处于正常的对象状态。] |
| 完成标准： | [成功地核实出各个窗口都与基准版本保持一致，或符合可接受标准] |
| 需考虑的特殊事项： | [并不是所有定制或第三方对象的特征都可访问。] |

## 性能评测

编写说明：性能评测是一种性能测试，它对响应时间、事务处理速率和其他与时间相关的需求进行评测和评估。性能评测的目标是核实性能需求是否都已满足。

|  |  |
| --- | --- |
| 测试目标： | [核实所指定的事务或业务功能在以下情况下的性能 行为：  正常的预期工作量；预期的最繁重工作量] |
| 技术： | [使用为功能或业务周期测试制定的测试过程。  通过修改数据文件来增加事务数量，或通过修改脚本来增加每项事务的迭代数量。  脚本应该在一台计算机上运行（最好是以单个用户、单个事务为基准），并在多个客户机（虚拟的或实际的客户机，请参见下面的“需要考虑的特殊事项”）上重复。] |
| 完成标准： | [单个事务或单个用户：在每个事务所预期或要求的时间范围内成功地完成测试脚本，没有发生任何故障。]  [多个事务或多个用户：在可接受的时间范围内成功地完成测试脚本，没有发生任何故障。] |
| 需考虑的特殊事项： | [综合的性能测试还包括在服务器上添加后台工作量。  可采用多种方法来执行此操作，其中包括：  直接将“事务强行分配到”服务器上，这通常以“结构化查询语言”(SQL) 调用的形式来实现。  通过创建“虚拟的”用户负载来模拟许多个（通常为数百个）客户机。此负载可通过“远程终端仿真”工具来实现。此技术还可用于在网络中加载“流量”。  使用多台实际客户机（每台客户机都运行测试脚本）在系统上添加负载。  性能测试应该在专用的计算机上或在专用的机时内执行，以便实现完全的控制和精确的评测。  性能测试所用的数据库应该是实际大小或相同缩放比例的数据库。] |

## 负载测试

编写说明：负载测试是一种性能测试。在这种测试中，将使测试对象承担不同的工作量，以评测和评估测试对象在不同工作量条件下的性能行为，以及持续正常运行的能力。负载测试的目标是确定并确保系统在超出最大预期工作量的情况下仍能正常运行。此外，负载测试还要评估性能特征，例如，响应时间、事务处理速率和其他与时间相关的方面。

|  |  |
| --- | --- |
| 测试目标： | [核实所指定的事务或商业理由在不同的工作量条件下的性能行为时间。] |
| 技术： | [使用为功能或业务周期测试制定的测试。  通过修改数据文件来增加事务数量，或通过修改测试来增加每项事务发生的次数。] |
| 完成标准： | [多个事务或多个用户：在可接受的时间范围内成功地完成测试，没有发生任何故障。] |
| 需考虑的特殊事项： | [负载测试应该在专用的计算机上或在专用的机时内执行，以便实现完全的控制和精确的评测。  负载测试所用的数据库应该是实际大小或相同缩放比例的数据库。] |

## 强度测试

编写说明：强度测试是一种性能测试，实施和执行此类测试的目的是找出因资源不足或资源争用而导致的错误。如果内存或磁盘空间不足，测试对象就可能会表现出一些在正常条件下并不明显的缺陷。而其他缺陷则可能由于争用共享资源（如数据库锁或网络带宽）而造成的。强度测试还可用于确定测试对象能够处理的最大工作量。

|  |  |
| --- | --- |
| 测试目标： | [核实测试对象能够在以下强度条件下正常运行，不会出现任何错误：  服务器上几乎没有或根本没有可用的内存（RAM 和 DASD）  连接或模拟了最大实际（实际允许）数量的客户机  多个用户对相同的数据或账户执行相同的事务  最繁重的事务量或最差的事务组合（请参见上面的“性能测试”）。  注：强度测试的目标可表述为确定和记录那些使系统无法继续正常运行的的情况或条件。  客户机的强度测试在“配置测试”的第 3.1.11 节中进行了说明。] |
| 技术： | [使用为性能评测或负载测试制定的测试。  要对有限的资源进行测试，就应该在一台计算机上运行测试，而且应该减少或限制服务器上的 RAM 和 DASD。  对于其他强度测试，应该使用多台客户机来运行相同的测试或互补的测试，以产生最繁重的事务量或最差的事务组合。 |
| 完成标准： | [所计划的测试已全部执行，并且在达到或超出指定的系统限制时没有出现任何软件故障，或者导致系统出现故障的条件并不在指定的条件范围之内。] |
| 需考虑的特殊事项： | [如果要增加网络工作强度，可能会需要使用网络工具来给网络加载消息或信息包。应该暂时减少用于系统的 DASD，以限制数据库可用空间的增长。使多个客户机对相同的记录或数据账户同时进行的访问达到同步。] |

## 容量测试

编写说明：容量测试使测试对象处理大量的数据，以确定是否达到了将使软件发生故障的极限。容量测试还将确定测试对象在给定时间内能够持续处理的最大负载或工作量。例如，如果测试对象正在为生成一份报表而处理一组数据库记录，那么容量测试就会使用一个大型的测试数据库，检验该软件是否正常运行并生成了正确的报表。

|  |  |
| --- | --- |
| 测试目标： | [核实测试对象在以下高容量条件下能否正常运行：  连接或模拟了最大（实际或实际允许）数量的客户机，所有客户机在长时间内执行相同的、且情况（性能）最坏的业务功能。  已达到最大的数据库大小（实际的或按比例缩放的），而且同时执行了多个查询或报表事务。] |
| 技术： | [使用为性能评测或负载测试制定的测试。  应该使用多台客户机来运行相同的测试或互补的测试，以便在长时间内产生最繁重的事务量或最差的事务组合（请参见上面的“强度测试”）。  创建最大的数据库大小（实际的、按比例缩放的、或填充了代表性数据的数据库），并使用多台客户机在长时间内同时运行查询和报表事务。] |
| 完成标准： | [所计划的测试已全部执行，而且在达到或超出指定的系统限制时没有出现任何软件故障。] |
| 需考虑的特殊事项： | [对于上述的高容量条件，哪个时间段是可以接受的时间？] |

## 安全性和访问控制测试

编写说明：安全性和访问控制测试侧重于安全性的两个关键方面：

应用程序级别的安全性，包括对数据或业务功能的访问。

系统级别的安全性，包括对系统的登录或远程访问。

应用程序级别的安全性可确保：在预期的安全性情况下，只能访问特定的功能，或者只能访问有限的数据。例如，可能会允许所有人输入数据，创建新账户，但只有管理员才能删除这些数据或账户。如果具有数据级别的安全性，测试就可确保用户一能够看到所有客户消息（包括财务数据），而用户二只能看见同一客户的统计数据。

系统级别的安全性可确保只有具备系统访问权限的用户才能访问应用程序，而且只能通过相应的网关来访问。

|  |  |
| --- | --- |
| 测试目标： | 应用程序级别的安全性：[核实主角只能访问其所属用户类型已被授权访问的那些功能或数据。]  系统级别的安全性：[核实只有具备系统和应用程序访问权限的主角才能访问系统和应用程序。] |
| 技术： | 应用程序级别的安全性：[确定并列出各用户类型及其被授权访问的功能或数据。]  [为各用户类型创建测试，并通过创建各用户类型所特有的事务来核实其权限。]  修改用户类型并为相同的用户重新运行测试。对于每种用户类型，确保正确地提供或拒绝了这些附加的功能或数据。  系统级别的访问：[请参见以下的“需考虑的特殊事项”] |
| 完成标准： | [各种已知的主角类型都可访问相应的功能或数据，而且所有事务都按照预期的方式运行，并在先前的应用程序功能测试中运行了所有的事务。] |
| 需考虑的特殊事项： | [必须与相应的网络或系统管理员一起对系统访问权进行检查和讨论。由于此测试可能是网络管理或系统管理的职能，可能会不需要执行此测试。] |

## 故障转移和恢复测试

编写说明：故障转移和恢复测试可确保测试对象能成功完成故障转移，并能从导致意外数据损失或数据完整性破坏的各种硬件、软件或网络故障中恢复。

故障转移测试可确保：对于必须持续运行的系统，一旦发生故障，备用系统就将不失时机地“顶替”发生故障的系统，以避免丢失任何数据或事务。

恢复测试是一种对抗性的测试过程。在这种测试中，将把应用程序或系统置于极端的条件下（或者是模拟的极端条件下），以产生故障（例如I/O故障或无效的数据库指针和关健字）。然后调用恢复进程并监测和检查应用程序和系统，核实应用程序或系统和数据已得到了正确的恢复。

|  |  |
| --- | --- |
| 测试目标： | [确保恢复进程（手工或自动）将数据库、应用程序和系统正确地恢复到了预期的已知状态。测试中将包括以下各种情况：  客户机断电  服务器断电  通过网络服务器产生的通信中断  DASD 和/或 DASD 控制器被中断、断电或与 DASD 和/或DASD 控制器的通信中断  周期未完成（数据过滤进程被中断，数据同步进程被中断）。  数据库指针或关键字无效  数据库中的数据元素无效或遭到破坏] |
| 技术： | [应该使用为功能和业务周期测试创建的测试来创建一系列的事务。一旦达到预期的测试起点，就应该分别执行或模拟以下操作：  客户机断电：关闭 PC 机的电源。  服务器断电：模拟或启动服务器的断电过程。  通过网络服务器产生的中断：模拟或启动网络的通信中断（实际断开通信线路的连接或关闭网络服务器或路由器的电源）。  DASD 和DASD 控制器被中断、断电或与 DASD 和 DASD 控制器的通信中断：模拟与一个或多个 DASD 控制器或设备的通信，或实际取消这种通信。  一旦实现了上述情况（或模拟情况），就应该执行其他事务。而且一旦达到第二个测试点状态，就应调用恢复过程。  在测试不完整的周期时，所使用的技术与上述技术相同，只不过应异常终止或提前终止数据库进程本身。  对以下情况的测试需要达到一个已知的数据库状态。当破坏若干个数据库字段、指针和关键字时，应该以手工方式在数据库中（通过数据库工具）直接进行。其他事务应该通过使用“应用程序功能测试”和“业务周期测试”中的测试来执行，并且应执行完整的周期。] |
| 完成标准： | [在所有上述情况中，应用程序、数据库和系统应该在恢复过程完成时立即返回到一个已知的预期状态。此状态包括仅限于已知损坏的字段、指针或关键字范围内的数据损坏，以及表明进程或事务因中断而未被完成的报表。] |
| 需考虑的特殊事项： | [恢复测试会给其他操作带来许多的麻烦。断开缆线连接的方法（模拟断电或通信中断）可能并不可取或不可行。所以，可能会需要采用其他方法，例如诊断性软件工具。  需要系统（或计算机操作）、数据库和网络组中的资源。  这些测试应该在工作时间之外或在一台独立的计算机上运行。] |

## 配置测试

编写说明：配置测试核实测试对象在不同的软件和硬件配置中的运行情况。在大多数生产环境中，客户机工作站、网络连接和数据库服务器的具体硬件规格会有所不同。客户机工作站可能会安装不同的软件例如，应用程序、驱动程序等而且在任何时候，都可能运行许多不同的软件组合，从而占用不同的资源。

|  |  |
| --- | --- |
| 测试目标： | [核实测试对象可在所需的硬件和软件配置中正常运行。] |
| 技术： | [使用功能测试脚本。  在测试过程中或在测试开始之前，打开各种与非测试对象相关的软件（例如 Microsoft 应用程序：Excel 和 Word），然后将其关闭。  执行所选的事务，以模拟主角与测试对象软件和非测试对象软件之间的交互。  重复上述步骤，尽量减少客户机工作站上的常规可用内存。] |
| 完成标准： | [对于测试对象软件和非测试对象软件的各种组合，所有事务都成功完成，没有出现任何故障。] |
| 需考虑的特殊事项： | [需要、可以使用并可以通过桌面访问哪种非测试对象软件？  通常使用的是哪些应用程序？  应用程序正在运行什么数据？例如，在 Excel 中打开的大型电子表格，或是在 Word 中打开的 100 页文档。  作为此测试的一部分，应将整个系统、Netware、网络服务器、数据库等都记录下来。] |

## 安装测试

编写说明：安装测试有两个目的。第一个目的是确保该软件在正常情况和异常情况的不同条件下例如，进行首次安装、升级、完整的或自定义的安装都能进行安装。异常情况包括磁盘空间不足、缺少目录创建权限等。第二个目的是核实软件在安装后可立即正常运行。这通常是指运行大量为功能测试制定的测试。

|  |  |
| --- | --- |
| 测试目标： | 核实在以下情况下，测试对象可正确地安装到各种所需的硬件配置中：  首次安装：以前从未安装过该软件的新计算机  更新：以前安装过相同版本的该软件的计算机  更新：以前安装过该软件的较早版本的计算机 |
| 技术： | [手工开发脚本或开发自动脚本，以验证目标计算机的状况（首次安装 – 该软件从未安装过；该软件安装过相同或较早的版本）。  启动或执行安装。  使用预先确定的功能测试脚本子集来运行事务。] |
| 完成标准： | 该软件事务成功执行，没有出现任何故障。 |
| 需考虑的特殊事项： | [应该选择哪些事务才能准确地测试出应用程序已经成功安装，而且没有遗漏主要的软件构件？] |

# 活动裁剪

在进行项目管理活动裁剪时，项目经理可以参考下表进行裁剪，流程活动按大、中、小型项目进行对照，对应打“√”的，说明该活动不应被删除，而其它活动项目经理可以进行裁剪。

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **编号** | **活动名称** | **大型项目** | | | **中型项目** | | | **小型项目** | | |
| **一级版本** | **二级版本** | **三级版本** | **一级版本** | **二级版本** | **三级版本** | **一级版本** | **二级版本** | **三级版本** |
| *4.1* | 数据和数据库完整性测试 | *√* | *√* |  | *√* | *√* | *√* | *√* | *√* |  |
| *4.2* | 功能测试 | *√* | *√* | *√* | *√* | *√* | *√* | *√* | *√* | *√* |
| *4.3* | 业务周期测试 | *√* | *√* | *√* | *√* | *√* | *√* | *√* | *√* | *√* |
| *4.4* | 用户界面测试 | *√* | *√* | *√* | *√* | *√* | *√* | *√* | *√* | *√* |
| *4.5* | 性能测试 | *√* | *√* |  | *√* |  |  | *√* |  |  |
| *4.6* | 负载测试 | *√* | *√* |  | *√* |  |  |  |  |  |
| *4.7* | 强度测试 | *√* | *√* |  | *√* |  |  |  |  |  |
| *4.8* | 容量测试 | *√* | *√* |  | *√* |  |  |  |  |  |
| *4.9* | 安全性和访问控制测试 | *√* |  |  | *√* |  |  | *√* |  |  |
| *4.10* | 故障转移和恢复测试 | *√* |  |  | *√* |  |  |  |  |  |
| *4.11* | 配置测试 | *√* |  |  | *√* |  |  | *√* |  |  |
| *4.12* | 安装测试 | *√* | *√* |  | *√* | *√* |  | *√* |  |  |

注：一级版本：V1.0.0、2.0.0……

二级版本：V1.1.0、1.2.0……

三级版本：V1.1.1、1.1.2……

**修订状况记录表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **版本号** | **页码** | **修改条款** | **审核** | **批准** | **批准日期** | **实施日期** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |