



中华人民共和国国家标准

GB/T 29832.3—2013

系统与软件可靠性 第3部分：测试方法

Reliability of system and software—
Part 3: Testing method

2013-11-12 发布

2014-02-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会 发布

目 次

前言 I

引言 II

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 一般测试方法与过程 1

 4.1 测试方法 1

 4.2 测试过程 1

5 成熟性 2

 5.1 失效度 2

 5.2 故障度 3

 5.3 测试度 4

 5.4 有效度 5

6 容错性 6

 6.1 正常运行度 6

 6.2 抵御误操作率 7

7 易恢复性 8

 7.1 重启成功度 8

 7.2 修复成功度 9

附录 A（资料性附录） 可靠性测试过程的说明 11

参考文献 15

前 言

GB/T 29832《系统与软件可靠性》分为如下三部分：

- 第 1 部分：指标体系；
- 第 2 部分：度量方法；
- 第 3 部分：测试方法。

本部分为 GB/T 29832 的第 3 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

本部分由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本部分起草单位：上海计算机软件技术开发中心、珠海南方软件产品检测中心、中国电子技术标准化研究院、深圳市中安测标准技术有限公司、北京邮电大学、上海浦东软件平台有限公司、上海宝信软件股份有限公司、上海鲁齐信息科技有限公司、河南电力试验研究院。

本部分主要起草人：丁志刚、阎梦天、宗宇伟、蔡立志、侯建华、张毅、张旻旻、袁玉宇、李家宏、张露莹、胡劲华、董文生、张彦琼、郭龙祥、郭庆、许彦淳、牛霜霞。

引 言

GB/T 29832 的本部分对可靠性度量的测试过程进行了描述,给出了具体执行步骤或相应的方法,便于用户或者评测人员进行操作。

本部分所列的方法并非一个完备集,不包括尚在探索中的新方法。进行可靠性测试的人员可以从本部分中选择合适的单一方法或者将多个方法进行组合,用来测量被测系统的可靠性质量情况;也可以修改或使用本部分未包括的其他方法。

本部分适用于各种具有可靠性需求的计算机软件产品及相关系统,但并非每种方法适用于各种计算机软件产品及相关系统,并且本部分假设被测软件及相关系统已经过一段时间的运行、已进行过功能性测试。

本标准预期的主要使用者包括:

- a) 需方(从供方获得或采购系统、软件产品或软件服务的个体或组织);
- b) 评价者(实施评价的个体或组织);
- c) 供方(按所签合同向需方提供系统、软件产品或软件服务的个体或组织),其在合格性测试中确认软件质量时使用。

GB/T 29832.1《系统与软件可靠性 第1部分:指标体系》给出了可靠性指标体系,GB/T 29832.2《系统与软件可靠性 第2部分:度量方法》给出了如何获得可靠性指标测量值的度量方法。本部分旨在与 GB/T 29832.1 和 GB/T 29832.2 联合使用。

系统与软件可靠性

第3部分：测试方法

1 范围

GB/T 29832 的本部分规定了如何获得可靠性指标测量值的测试方法。

本部分适用于具有可靠性要求的系统与软件的可靠性测试和评价。

注：本部分中所指的系统主要是软件系统。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本(包括所有的修改单)适用于本文件。

GB/T 11457 软件工程术语

GB/T 29832.1 系统与软件可靠性 第1部分：指标体系

GB/T 29832.2 系统与软件可靠性 第2部分：度量方法

3 术语和定义

GB/T 11457 和 GB/T 29832.1 中界定的术语和定义适用于本文件。

4 一般测试方法与过程

4.1 测试方法

测试方法包括如下：

专家评审法——一种主观的测评方法。评审时，应根据被评审对象和评审目的设计评审项目表，列出打分栏目、分值、权重和打分规则。可由 N 个专家组成一个评审组，专家根据自身的经验与认知进行判断打分。然后，依据专家的权重和统计规则进行分值汇总计算，其计算得出的值作为评审的结果。打分规则可以是分等级的，也可以是一个取值区间，或选择一个其他的合适规则；

技术测试法——一种客观的测评方法。技术测试时，可依据被测对象和测试目的选择适用的自动化测试工具进行测试，也可由人工进行手工测试。技术测试获得的结果通常是一种量化的测量结果；

数学计算法——一种利用数学模型进行计算的测评方式。测评时，应根据被测对象和测试目的选择适用的数学模型，然后进行计算，最终得出作为测评结果的数值；

用户调查法——一种面向特定用户群的问卷征询方法。用户调查时，应根据调查的目的和特定的用户群设计调查表，让被调查对象填写并反馈。调查表的回收数应达到一定的数量，并不低于发出数的适当比例。然后，对回收的调查表进行汇总计算，其计算得出的值作为用户调查的结果。

4.2 测试过程

可靠性的一般测试过程可包括可靠性测试需求分析、功能剖面构造、测试用例设计、测试场景设计、测试环境和工具的准备、可靠性测试实施、数据收集和分析、故障跟踪等环节，如图1所示。主要过程具体说明参见附录A。

GB/T 29832.3—2013

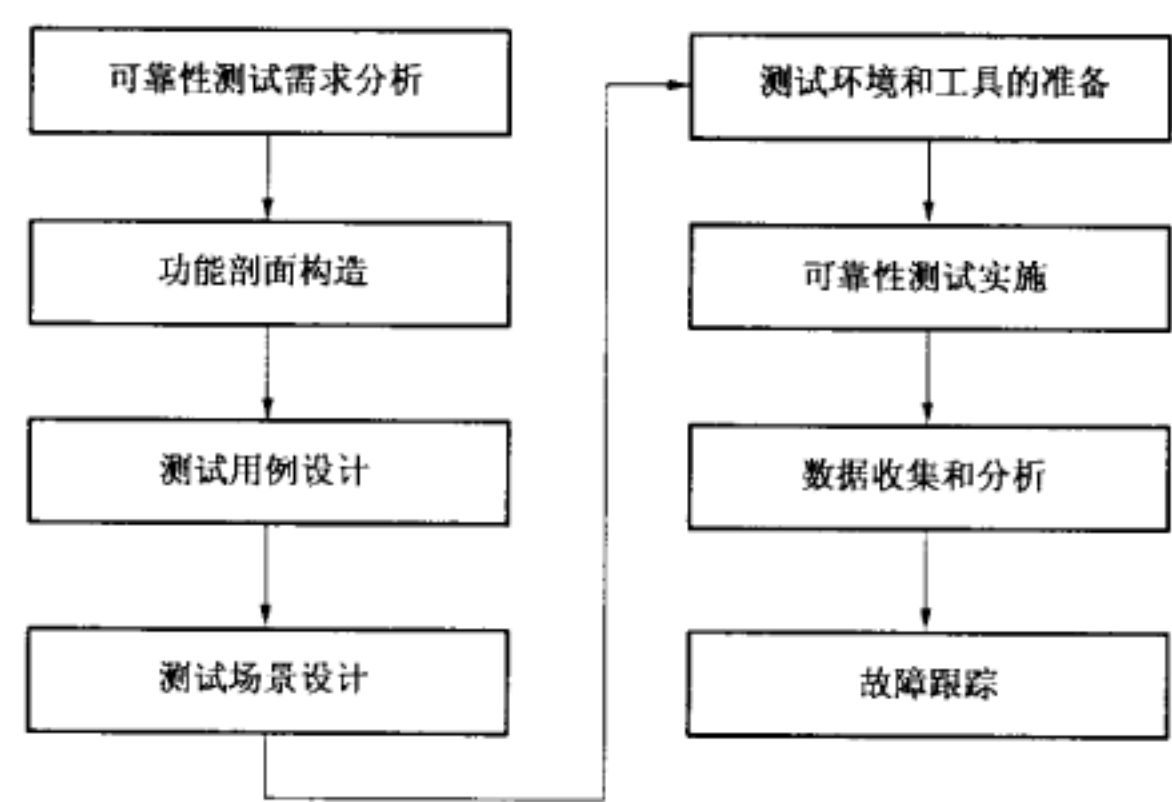


图 1 可靠性测试流程

5 成熟性

5.1 失效度

5.1.1 失效密度

失效密度的测试要求如下：

——前置条件

被测软件已经过一段时间的运行；

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等；

——测量类型

计数/计数；

——测试方法

技术测试法；

——实施要点：

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等，针对每个功能编制对应的测试用例；
- 2) 规定试验周期时长，执行所有测试用例，收集和分析测试结果；
- 3) 依据测试结果，确定检测到的失效数，并记为 A ；
- 4) 统计实际执行的测试用例总数，并记为 B ；
- 5) 根据 GB/T 29832.2 中的对应公式 $X=A/B$ ，计算出度量结果 X 。

注：本度量取决于测试用例是否充分，测试用例的设计宜包括适当的测试用例类型，即正常的、例外的和异常的测试用例。

5.1.2 失效解决率

失效解决率的测试要求如下：

——前置条件

被测软件已经过一段时间的运行，并已进行过功能性和失效密度测试；

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等；

——测量类型

计数/计数；

——测试方法

技术测试法；

——实施要点：

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等，针对每个失效编制对应的测试用例，执行所有测试用例，收集和分析测试结果；
- 2) 依据问题报告和测试结果，确定有多少失效在同样的条件下再未出现，并记为 A ；
- 3) 将失效密度测试中实际检测到的失效数记为 B ；
- 4) 根据 GB/T 29832.2 中的对应公式 $X=A/B$ ，计算出度量结果 X 。

注 1：在使用这一度量时，建议监视其趋势。

注 2：可以用可靠性增长估计模型来预测潜在的失效总数。

5.2 故障度

5.2.1 故障密度

故障密度的测试要求如下：

——前置条件

被测软件已经过一段时间的运行；

——测试输入

被测软件的需求文档、设计文档、操作手册、源代码、运行报告、问题报告等；

——测量类型

计数/计数；

——测试方法

技术测试法；

——实施要点：

- 1) 依据需求文档、设计文档、操作手册、源代码、运行报告、问题报告等，针对每个功能编制对应的测试用例；
- 2) 规定试验周期时长，执行所有测试用例，收集和分析测试结果；
- 3) 依据测试结果，确定检测到的故障数，并记为 A ；
- 4) 选定一种规模度量的方法，如源代码行数或功能点数，将得到的产品规模记为 B ；
- 5) 根据 GB/T 29832.2 中的对应公式 $X=A/B$ ，计算出度量结果 X 。

注：若选用这一度量分析其趋势或横向对比时，需注意保持规模度量的一致性。

5.2.2 潜在故障率

潜在故障率的测试要求如下：

——前置条件

被测软件已经过一段时间的运行，并已进行过功能性和故障密度测试；

——测试输入

被测软件的需求文档、设计文档、操作手册、源代码等；

——测量类型

计数/计数；

——测试方法

GB/T 29832.3—2013

数学计算法；

——实施要点：

- 1) 收集和分析故障密度测试结果；
- 2) 选定一种可靠性增长估计模型，预测潜在的故障总数，并记为 A_1 ；
- 3) 将故障密度测试中实际检测到的故障数记为 A_2 ；
- 4) 选定一种规模度量的方法，如源代码行数或功能点数，将得到的产品规模记为 B ；
- 5) 根据 GB/T 29832.2 中的对应公式 $X=ABS(A_1-A_2)/B$ ，计算出度量结果 X 。

注 1：建议使用多种可靠性增长估计模型，选择其中最合适的一种。

注 2：对潜在的故障数预测一个上限和一个下限数可能会有帮助。

注 3：若选用这一度量分析其趋势或横向对比时，需注意保持规模度量的一致性。

5.2.3 故障排除率

故障排除率的测试要求如下：

——前置条件

被测软件已经过一段时间的运行，并已进行过功能性和故障密度测试；

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等；

——测量类型

计数/计数；

——测试方法

技术测试法；

——实施要点：

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等，针对每个故障编制对应的测试用例，执行所有测试用例，收集和分析测试结果；
- 2) 依据测试结果和问题报告，确定已排除故障的数目，并记为 A_1 ；
- 3) 将故障密度测试中实际检测到的故障数记为 A_2 ；
- 4) 选定一种可靠性增长估计模式，预测潜在的故障总数，并记为 A_3 ；
- 5) 根据 GB/T 29832.2 中的对应公式 $X= A_1/A_2$ ，计算出度量结果 X ；根据 GB/T 29832.2 中的对应公式 $Y=A_1/A_3$ ，计算出度量结果 Y 。

注：建议使用多种可靠性增长估计模型，选择其中最合适的一种。

5.3 测试度

5.3.1 测试覆盖率

测试覆盖率的测试要求如下：

——前置条件

被测软件已经过一段时间的运行，并已进行过功能性测试；

——测试输入

被测软件的需求文档、设计文档、操作手册等；

——测量类型

计数/计数；

——测试方法

技术测试法；

——实施要点：

- 1) 依据需求文档、设计文档、操作手册等,针对覆盖需求编制对应的测试用例,执行所有测试用例,收集和分析测试结果;
- 2) 确定在测试期间实际执行的测试用例数,并记为 A ;
- 3) 统计按覆盖要求计划执行的测试用例数,并记为 B ;
- 4) 根据 GB/T 29832.2 中的对应公式 $X=A/B$,计算出度量结果 X 。

5.3.2 测试通过率

测试通过率的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告等;

——测量类型

计数/计数;

——测试方法

技术测试法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告等,针对覆盖需求编制对应的测试用例,执行所有测试用例,收集和分析测试结果;
- 2) 确定在测试或运行中通过的测试用例数,并记为 A ;
- 3) 统计按覆盖要求计划执行的测试用例数,并记为 B ;
- 4) 根据 GB/T 29832.2 中的对应公式 $X=A/B$,计算出度量结果 X 。

注:建议确保执行下列类型的测试并顺利地通过它们:

——用户操作的场景测试;

——高峰期的强度测试;

——过载的数据输入测试。

5.4 有效度

5.4.1 平均失效间隔时间

平均失效间隔时间的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等;

——测量类型

时间/计数;

——测试方法

技术测试法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等,针对每个功能编制对应的测试用例;
- 2) 确定运行时间(记为 T_1),在 T_1 周期内执行所有测试用例,收集和分析测试结果;
- 3) 依据测试结果和运行报告,累计相继发生失效之间的时间间隔,并记为 T_2 。

GB/T 29832.3—2013

- 4) 依据测试结果,统计实际检测到的失效总数,并记数为 A ;
- 5) 根据 GB/T 29832.2 中的对应公式 $X = T_1/A$,计算出度量结果 X ;根据 GB/T 29832.2 中的对应公式 $Y = T_2/A$,计算出度量结果 Y 。

5.4.2 有效服务时间率

有效服务时间率的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等;

——测量类型

时间/时间;

——测试方法

数学计算法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等,收集和分析有关数据;
- 2) 确定无失效的服务时间,并记为 A ;
- 3) 累计总的服务时间,并记为 B ;
- 4) 根据 GB/T 29832.2 中的对应公式 $X = A/B$,计算出度量结果 X 。

5.4.3 累计有效服务时间

累计有效服务时间的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等;

——测量类型

时间;

——测试方法

数学计算法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等,收集和分析有关数据;
- 2) 累计所有已记录的无失效的服务时间,并记为 T ;
- 3) 根据 GB/T 29832.2 中的对应公式 $X = T$,得到度量结果 X 。

6 容错性

6.1 正常运行度

6.1.1 避免宕机率

避免宕机率的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性和失效密度测试;

- 测试输入
被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等；
- 测量类型
计数/计数；
- 测试方法
数学计算法；
- 实施要点：
 - 1) 收集和分析失效密度测试结果；
 - 2) 依据运行报告、问题报告，并分析软件系统的运行历史日志，确定导致宕机发生的失效数，并记为 A ；
 - 3) 将失效密度测试中实际检测到的失效数记为 B ；
 - 4) 根据 GB/T 29832.2 中的对应公式 $X=1-A/B$ ，计算出度量结果 X 。

注：宕机意味着在系统重新启动之前，用户所有的任务均已停止，或对系统失去控制，系统只能被迫停机。

6.1.2 避免失效率

避免失效率的测试要求如下：

- 前置条件
被测软件已经过一段时间的运行，并已进行过功能性测试；
- 测试输入
被测软件的需求文档、设计文档、操作手册等；
- 测量类型
计数/计数；
- 测试方法
技术测试法；
- 实施要点：
 - 1) 依据需求文档、设计文档、操作手册等，针对每种故障模式编制对应的测试用例，执行所有测试用例，收集和分析测试结果；
 - 2) 执行对应故障模式的测试用例时，统计未发生关键的和严重的失效的测试用例数，并记为 A ；
 - 3) 统计执行的故障模式的测试用例总数，并记为 B ；
 - 4) 根据 GB/T 29832.2 中的对应公式 $X=A/B$ ，计算出度量结果 X 。

注 1：建议将避免失效的水平按对故障影响的缓急程度进行分类，具体参见附录 A。

注 2：故障模式举例：

- 数据超范围；
- 死锁。

6.2 抵御误操作率

抵御误操作率的测试要求如下：

- 前置条件
被测软件已经过一段时间的运行，并已进行过功能性测试；
- 测试输入
被测软件的需求文档、设计文档、操作手册等；
- 测量类型

- 计数/计数；
 - 测试方法
 - 技术测试法；
 - 实施要点：
 - 1) 依据需求文档、设计文档、操作手册等，针对每种误操作模式编制对应的测试用例，执行所有测试用例，收集和分析测试结果；
 - 2) 执行误操作模式的测试用例时，统计未发生关键的和严重的失效的测试用例数，并记为 A ；
 - 3) 统计执行的误操作模式的测试用例总数，并记为 B ；
 - 4) 根据 GB/T 29832.2 中的对应公式 $X=A/B$ ，计算出度量结果 X 。
- 注 1：除软件系统失效外，还会有数据损害。
- 注 2：备份的建立可确保必要的数据在部分操作数据丢失的事件发生后能很快地恢复。
- 注 3：误操作的模式：
- 错误的数据类型作参数；
 - 错误的输入数据序列；
 - 错误的操作序列。

7 易恢复性

7.1 重启成功度

7.1.1 平均宕机时间

- 平均宕机时间的测试要求如下：
- 前置条件
 - 被测软件已经过一段时间的运行，并已进行过功能性测试；
 - 测试输入
 - 被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等；
 - 测量类型
 - 时间/计数；
 - 测试方法
 - 技术测试法、数学计算法；
 - 实施要点：
 - 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等，针对每个失效和故障编制对应的测试用例；
 - 2) 确定特定的试验周期，执行所有的测试用例，收集和分析测试结果；
 - 3) 依据测试结果和运行报告、问题报告，累计在特定的试验周期内每次从宕机起到软件可正常使用所花费的时间，并记为 T ；
 - 4) 统计特定的试验周期内所观察(或记录)到的宕机次数，并记为 N ；
 - 5) 根据 GB/T 29832.2 中的对应公式 $X=T/N$ ，计算出度量结果 X ；
 - 6) 宜测量最坏的情况及宕机时间的分布情况。

7.1.2 平均恢复时间

- 平均恢复时间的测试要求如下：
- 前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等;

——测量类型

时间/计数;

——测试方法

技术测试法、数学计算法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等,针对每个失效和故障编制对应的测试用例;
- 2) 确定特定的试验周期,执行所有的测试用例,收集和分析测试结果;
- 3) 依据测试结果和运行报告、问题报告,确定每次从失效起到完全恢复所花费的时间,并分别记为 T_1, \dots, T_n ;
- 4) 统计特定的试验周期内软件系统进入恢复的总次数,并记为 N ;
- 5) 根据 GB/T 29832.2 中的对应公式 $X = (T_1 + \dots + T_n) / N$,计算出度量结果 X 。

注 1: 建议测试多个测试用例中最坏情况下的最大恢复时间或恢复时间的分布情况。

注 2: 建议区分恢复工作的难度,如恢复被毁坏的数据库比恢复被破坏的事务难得多。

7.2 修复成功度

7.2.1 易修复性

易修复性的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等;

——测量类型

计数/计数;

——测试方法

技术测试法、数学计算法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等,针对每个失效和故障编制对应的测试用例;
- 2) 确定特定的试验周期,执行所有的测试用例,收集和分析测试结果;
- 3) 依据测试结果和运行报告、问题报告,确定成功完成恢复的测试用例数,并记为 A ;
- 4) 统计执行的恢复测试用例总数,并记为 B ;
- 5) 根据 GB/T 29832.2 中的对应公式 $X = A / B$,计算出度量结果 X 。

注: 修复需求举例:

——数据库检查点;

——事务检查点;

——重做功能;

——还原功能。

7.2.2 修复有效性

修复有效性的测试要求如下:

——前置条件

被测软件已经过一段时间的运行,并已进行过功能性测试;

——测试输入

被测软件的需求文档、设计文档、操作手册、运行报告、问题报告等;

——测量类型

计数/计数;

——测试方法

技术测试法、数学计算法;

——实施要点:

- 1) 依据需求文档、设计文档、操作手册、运行报告、问题报告等,针对每个失效和故障编制对应的测试用例;
- 2) 确定特定的试验周期,执行所有的测试用例,收集和分析测试结果;
- 3) 依据测试结果和运行报告、问题报告,确定满足目标恢复时间成功完成恢复的测试用例数,并记为 A ;
- 4) 统计执行的恢复测试用例总数,并记为 B ;
- 5) 根据 GB/T 29832.2 中的对应公式 $X=A/B$,计算出度量结果 X 。

附录 A
(资料性附录)
可靠性测试过程的说明

A.1 可靠性需求分析

A.1.1 软件失效的等级

将失效的水平依对软件影响的缓急程度进行分类,如表 A.1 所示。

表 A.1 失效等级表

失效等级	描述	说明
等级 1:关键性失效	整个系统中止或严重毁坏数据库	例如,宕机、蓝屏
等级 2:严重性失效	重要功能无法正常运行,并且没有替代的运行方式(与工作有关)	例如,程序错误
等级 3:普通失效	绝大部分功能仍然可用,次要功能受到限制或要采用替代方式(与工作有关)	例如,打印功能错误
等级 4:轻微失效	少数功能在有限的操作中受到限制	例如,界面不规范
等级 5:可忽略的失效	影响未波及到最终用户	

A.1.2 确定可靠性需求

软件可靠性需求分析是研究用户要求得到的软件可靠性的定义过程。具体地说就是了解和确认用户对软件的可靠性需求,可通过表 A.2 来描述软件可靠性需求与软件可靠性测试目标之间的对应关系。

表 A.2 用户需求与测试目标的关系

测试目标	用户需求
测量软件产品的容错性	软件能对何种失效实现容错,能抵御什么样的误操作和错误数据
测量软件产品的易恢复性	软件出现何种失效时能否实现自动恢复
估计软件产品稳定运行能力、平均失效恢复时间和平均失效间隔时间	实际运行时,软件能确保多长时间无故障运行,或软件的无故障操作次数

在可靠性需求分析的基础上,确定可靠性测试的目标:

- 有效地发现程序中影响软件可靠性的缺陷,从而实现可靠性增长;
- 验证软件可靠性满足一定的要求。通过对软件可靠性测试中观察到的失效情况进行分析,验证软件可靠性的定量要求;
- 估计软件可靠性水平。通过对软件可靠性测试中观察到的失效数据进行分析,可以评估当前软件可靠性的水平,预测未来可能达到的水平,从而为开发管理提供决策依据。

根据软件需求确定可靠性测试指标,确定可靠性测试易恢复性指标,如:

- 民用在线系统平均修复时间(MTTR)小于 2 min;

——一般系统平均修复时间小于 7 min。

对于可靠性要求比较高的系统,一般认为多用户操作时系统平均半年出现一次失效是可以接受的。

A.1.3 选择可靠性模型

应尽早确定可靠性模型。

可靠性模型选择应考虑模型的假设条件,这些条件应符合软件的实际情况,或与假设条件冲突的因素在软件系统中是可以忽略的。应对每一条假设条件都进行分析。

一个可靠性模型需要通过实践不断改善才能提高预测能力。在满足其他条件的情况下,应选择比较成熟、应用范围广的模型作为分析模型。

模型的输出值应满足软件可靠性需求。

模型需要的数据在软件系统中应易于收集,同时应考虑测试前期需要投入的成本。

数据的输入也能通过测试工具获得。

如果软件可靠性测试过程中没有发生失效,除非已经对软件进行完全的测试,而这是几乎是不可能的,不应根据模型认为不会发生失效,而应根据测试用例的数量,在一定的置信度水平上,估计可靠性等级。

A.2 功能剖面构造

软件功能剖面是关于如何使用软件系统的一种定量描述。任何一个软件系统在使用过程中,各个功能部件的使用概率是不同的。软件功能剖面详细地刻画了软件系统实际的使用情况。功能剖面的定义对软件可靠性测试的测试用例的选择有着重要的影响。

采用基于功能剖面的方法选取测试用例,避免测试工作量分配不均匀的问题。

在进行可靠性测试时,首先要确定软件的功能剖面。可以通过分析同类软件的实际使用情况,或通过对重点用户的调查的信息分配到软件的功能路径上,即确定每一功能模块在给定系统中的使用概率,由此来确定测试用例数的选择方案,进行测试用例的选择。

根据不同功能模块的使用频率来均匀的分配测试工作,可以在软件可靠性测试中更真实地反映软件系统在实际使用中的情况,使软件中的每个功能都得到了充分的测试。

A.3 测试用例设计

在实际应用中,一般是对测试软件系统的关键功能模块或者复杂度高的模块设计较多的测试用例。

设计可靠性测试用例之前应先对软件系统构造功能剖面并对各功能模块的复杂度进行度量,然后针对复杂度高的功能模块设计较多的测试用例。

在实施可靠性测试时,需要运行系统相关的业务,这就需要一些数据支持才可运行业务。有时候需要准备大量数据,例如飞机订票系统查询订票信息,就需要准备大量的订票记录。又比如测试并发查询业务,就要求对应的数据库和表中有相当的数据量,并且数据的种类应能覆盖每种业务。

有时候还需要考虑特殊系统需要的测试数据,例如测试杀毒软件的扫描速度时,硬盘上布置的不同类型文件的比例要尽量接近真实环境,这样测试出来的数据才有意义。

测试数据的来源可以有几种方式:购买、工具生成、系统捕获、手工构造、随机产生等。不同的方式有不同的特性。方式的选择应该根据具体情况而定。

A.4 测试场景设计

测试场景设计可根据如下步骤进行：

- 根据相关文档和访谈对用户可能在运行软件时可能出现的失效模式进行总结并分类，确定关键的失效模式；
- 确定软件在实际应用中的业务流程，根据软件需求方提供的实际业务情况确定被测软件在应用中的业务流程；
- 根据历史数据估算同时运行软件的用户数量；
- 根据用户执行活动的进入时间和退出时间设计用户进入和退出软件系统时间；
- 设计测试执行周期，确定可靠性测试执行周期和测试时间。

A.5 测试环境和工具的准备

测试环境和工具的准备主要包括：

- 根据被测试的软件系统准备相应的软硬件及网络环境；
- 根据选定使用的测试工具准备相应的软硬件及网络环境；
- 根据设计的特定测试场景准备相应的软硬件及网络环境；
- 准备测试数据。

A.6 可靠性测试实施

对测试场景的部署可通过测试工具本身提供的功能来实现，这需要熟悉测试工具的人员来完成。一般由测试实施人员完成，并由测试设计人员确认其部署是否符合预期的设计。

准备好测试用例和部署好测试环境以及场景以后，就可以运行测试场景执行测试了。在测试工具的协助下，对测试执行的过程和结果进行记录。

在测试场景执行期间，需要完成以下工作：

- 记录在测试场景设计中定义的持续时间；
- 收集生成的错误、警告和通知信息。

测试数据记录可以依靠测试工具完成。通过测试工具的监控模块，可以获取并记录需要关注的各时间及次数记录值。

当测试工具本身不提供对所需关注的时间及次数进行监控的功能时，可以利用一些操作系统的工具，自行编制部分脚本解决这个问题。一般的方法是用脚本调用操作系统提供的工具，在脚本实现中将各时间及次数记录值分析出来，并按照一定格式记录在本地文件中。

A.7 数据收集和分析

应尽早根据确定的可靠性模式确定需要收集的可靠性数据，并定义可靠性数据规范和记录方法，如时间、失效、失效等级等。

制定可靠性数据收集计划，指定收集人员，收集人员可以来自开发方，也可以来自软件使用者。软件完成交付后，可根据可靠性需求，继续收集可靠性数据。

可以利用数据库来完成可靠性数据的存储。

可靠性数据的分析主要包括失效分析和可靠性分析：

——失效分析

失效分析是根据运行结果判断软件是否失效,以及失效的程度、后果、原因等;

——可靠性分析

可靠性分析主要是指根据失效数据,估计软件的可靠性水平,预计可能达到的水平,评价产品是否已经达到要求的可靠性水平,为管理决策提供依据。

如果软件的运行结果与需求不一致,则称软件发生失效。通过失效分析,找到并纠正引起失效的程序中的故障,从而实现软件可靠性的增长。

A.8 故障跟踪

故障跟踪一般应包括打开、分配、测试、重新打开以及关闭等状态。

打开:如果测试人员已能描述和隔离问题,则可以公开报告,将其作为已知的故障来公布。

分配:将故障分配给适当的开发经理,开发经理又依次把故障分配给开发人员,开发人员进行修改。

测试:一旦开发小组修复了问题,就要进入测试状态。对修复的故障要在测试组织中进行验证测试和回归测试。

重新打开:如果对修复验证测试失败,则重新打开故障报告;如果修复通过验证测试,但没有通过回归测试,则打开一个新的故障报告。

关闭:如果修复通过验证测试和回归测试,则关闭故障报告。

故障跟踪宜使用测试工具。

参 考 文 献

- [1] GB/T 5271.1—2000 信息技术 词汇 第1部分:基本术语(eqv ISO/IEC 2382-1:1993)
 - [2] GB/T 5271.14 2008 信息技术 词汇 第14部分:可靠性、可维护性和可用性(ISO/IEC 2382-14:1997, IDT)
 - [3] GB/T 5271.20—1994 数据处理词汇 20部分:系统开发(eqv ISO/IEC 2382-20:1990)
 - [4] GB/T 16260.2 2006 软件工程 产品质量 第2部分:外部度量(ISO/IEC TR 9126-2: 2003, IDT)
-

中 华 人 民 共 和 国
国 家 标 准
系统与软件可靠性
第 3 部分：测试方法
GB/T 29832.3—2013

*

中国标准出版社出版发行
北京市朝阳区和平里西街甲 2 号(100013)
北京市西城区三里河北街 16 号(100045)
网址 www.spc.net.cn
总编室:(010)64275323 发行中心:(010)51780235
读者服务部:(010)68523946
中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销

*

开本 880×1230 1/16 印张 1.25 字数 32 千字
2014 年 1 月第一版 2014 年 1 月第一次印刷

*

书号: 155066·1-47957 定价 21.00 元

如有印装差错 由本社发行中心调换
版权专有 侵权必究
举报电话:(010)68510107



GB/T 29832.3-2013