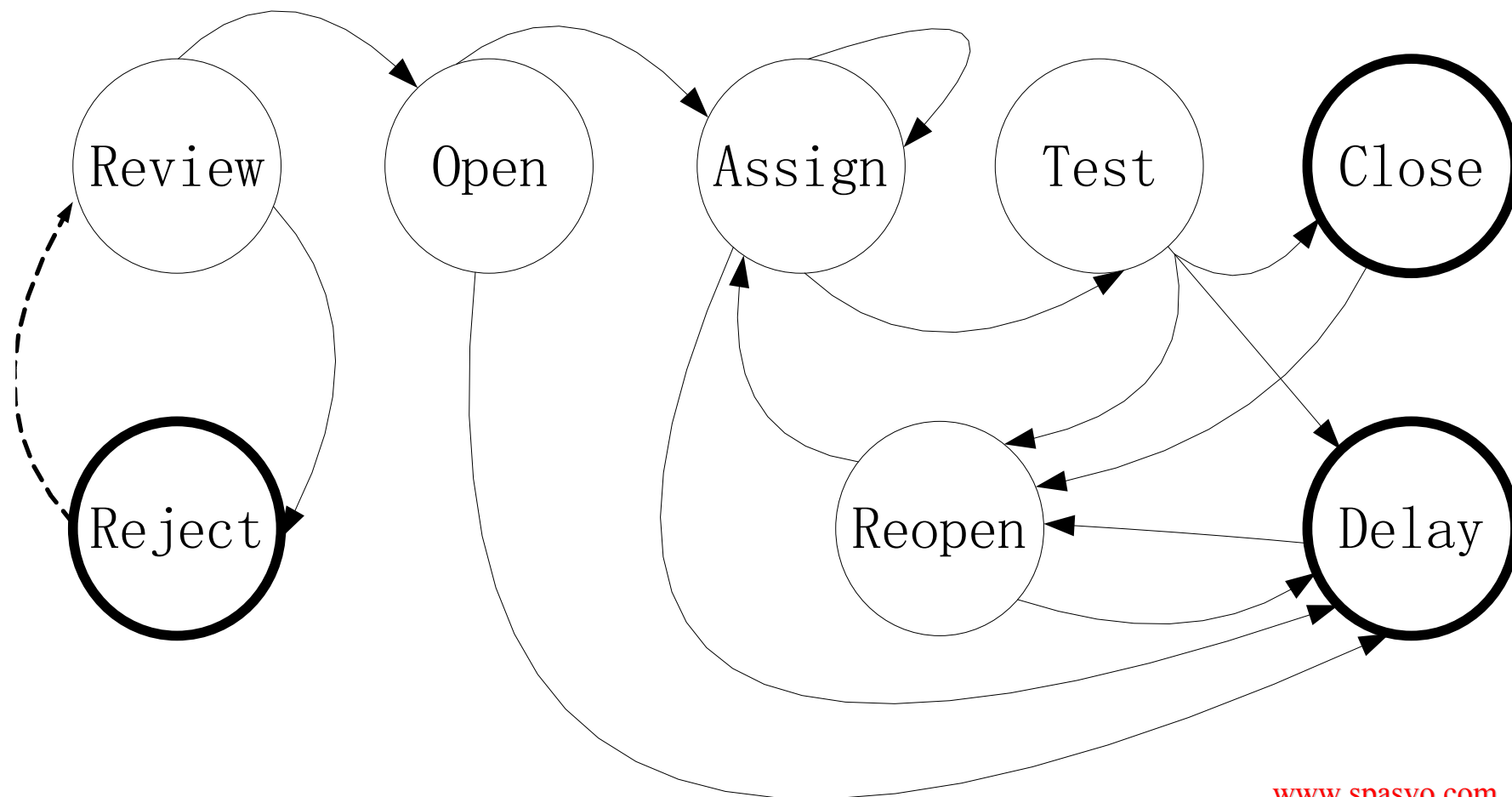
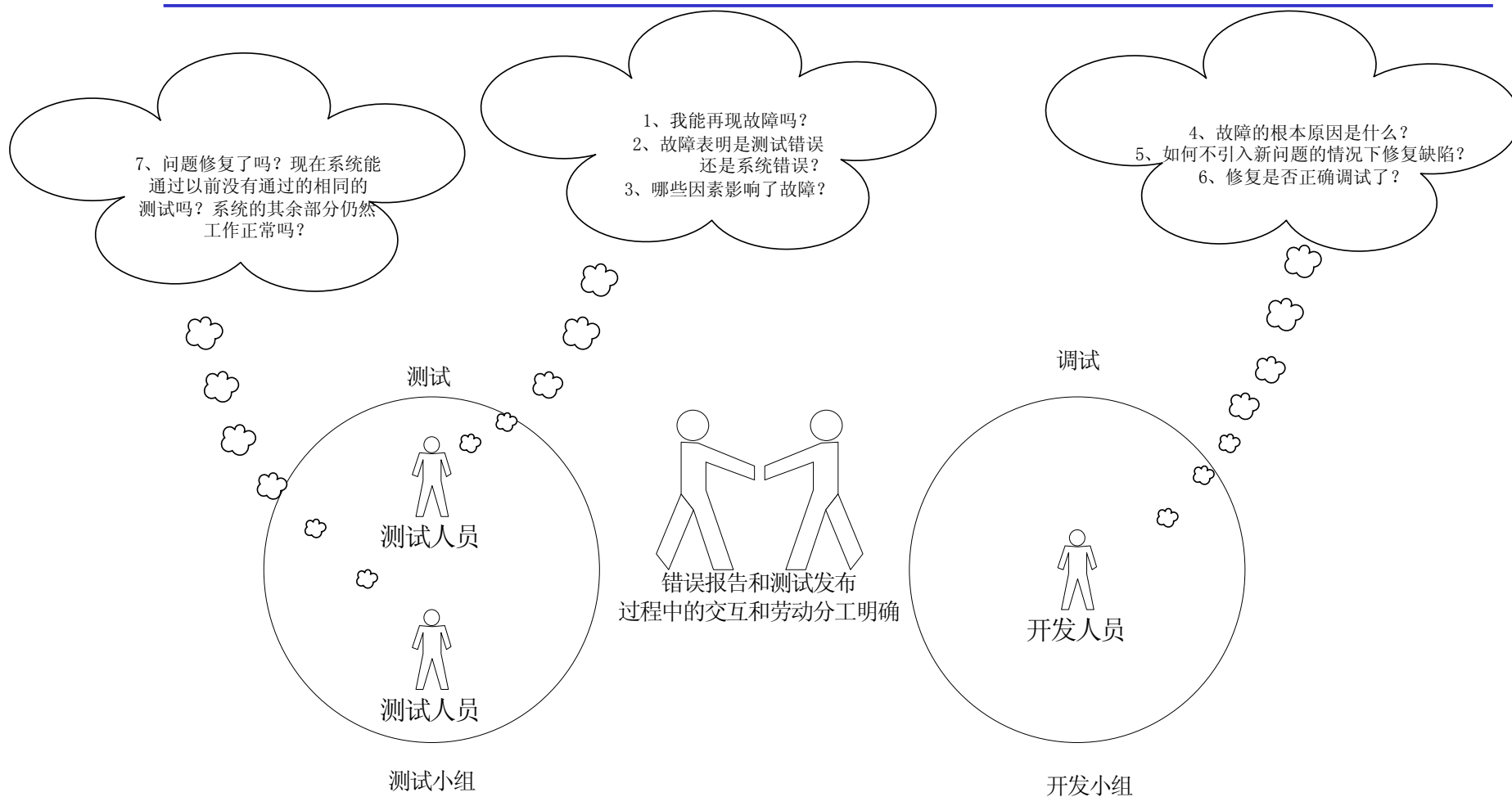


缺陷追踪与管理

缺陷报告生命周期或 workflow



错误生命周期中的问题、角色和转移



如何收集缺陷

- 缺陷既指程序中存在的错误，例如语法错误、拼写错误或者是一个不正确的程序语句，缺陷也指可能出现在设计中，甚至在需求、规格说明或其他的文档中的种种错误。
- 为了对缺陷进行管理，首先应对缺陷进行分类，通过对缺陷进行分类，可以迅速找出哪一类缺陷的问题最大，然后集中精力预防和排除这一类缺陷。
- 这正是缺陷管理的关键，一旦这几类缺陷得到控制，再进一步找到新的容易引起问题的几类缺陷上。

缺陷的优先级

代码	解决优先级	描述
1	立即解决 Resolve Immediately	缺陷必须被立即解决
2	正常排队 Normal Queue	缺陷需要正常排队等待修复 或列入软件发布清单
3	不紧急 Not Urgent	缺陷可以在有时间允许时被纠正

缺陷的状态

代码	缺陷状态	描述
1	已提交 Submitted	已提交的缺陷
2	已打开 Opened	已确认的缺陷，等待修复
3	已拒绝 Rejected	拒绝提交的缺陷，不需要修复或不是缺陷
4	已解决 Resolved	缺陷已被修复
5	已关闭 Closed	被修复的缺陷被确认，将其关闭

缺陷的严重程度

代码	缺陷严重等级	描述
1	致命缺陷 Critical	不能执行正常工作功能或重要功能 危及生命或财产
2	严重缺陷 Major	严重影响系统要求或基本功能的实现 且没有办法更正
3	普通缺陷 Minor	影响系统要求或基本功能的实现 但有办法更正
4	轻微缺陷 Cosmetic	是操作者遇到麻烦或不方便，但不 影响执行功能
5	其他缺陷 Other	其它错误，例如文字错误、 布局不合理等

缺陷的类型

按错误的影响和后果分类

- 其他错误：只对系统输出有一些非实质性影响。如，输出的数据格式不合要求等。
- 轻微错误：对系统的运行有局部影响。如输出的某些数据有错误或出现冗余。
- 普通错误：系统的行为因错误的干扰而出现明显不合情理的现象。比如开出了0.00元的支票，系统的输出完全不可信赖。
- 严重错误：系统运行不可跟踪，一时不能掌握其规律，时好时坏。
- 非常严重的错误：系统运行中突然停机，其原因不明，无法软启动。
- 致命错误：系统运行导致环境破坏，或是造成事故，引起生命、财产的损失。

按错误的性质和范围分类

- 功能错误
 - 规格说明错误：规格说明可能不完全，有二义性或自身矛盾。
 - 功能错误：程序实现的功能与用户要求的不一致。这常常是由于规格说明中包含错误的功能、多余的功能或遗漏的功能所致。
 - 测试错误：软件测试的设计与实施发生错误。软件测试自身也可能发生错误。
 - 测试标准引起的错误：对软件测试的标准要选择适当，若测试标准太复杂，则导致测试过程出错的可能就大。

- 系统错误

- 外部接口错误: 外部接口指如终端、打印机、通信线路等系统与外部环境通信的手段。所有外部接口之间,人与机器之间的通信都使用形式的或非形式的专门协议。如果协议有错,或太复杂,难以理解,致使在使用中出错。此外还包括对输入/输出格式错误理解,对输入数据不合理的容错等等。
- 内部接口错误: 内部接口指程序之间的联系。它所发生的错误与程序内实现的细节有关。例如,设计协议错、输入/输出格式错、数据保护不可靠、子程序访问错等。
- 硬件结构错误: 这类错误在于不能正确地理解硬件如何工作。例如,忽视或错误地理解分页机构、地址生成、通道容量、I/O指令、中断处理、设备初始化和启动等而导致的出错。
- 操作系统错误: 这类错误主要是由于不了解操作系统的工作机制而导致出错。。当然,操作系统本身也有错误,但是一般用户很难发现这种错误。
- 软件结构错误: 由于软件结构不合理或不清晰而引起的错误。这种错误通常与系统的负载有关,而且往往在系统满载时才出现。这是最难发现的一类错误。例如,错误地设置局部参数或全局参数;错误地假定寄存器与存储器单元初始化了;错误地假定不会发生中断而导致不能封锁或开中断;错误地假定程序可以绕过数据的内部锁而导致不能关闭或打开内部锁;错误地假定被调用子程序常驻内存或非常驻内存等等,都将导致软件出错。
- 控制与顺序错误: 这类错误包括:忽视了时间因素而破坏了事件的顺序;猜测事件出现在指定的序列中;等待一个不可能发生的条件;漏掉先决条件;规定错误的优先级或程序状态;漏掉处理步骤;存在不正确的处理步骤或多余的处理步骤等。
- 资源管理错误: 这类错误是由于不正确地使用资源而产生的。例如,使用未经获准的资源;使用后未释放资源;资源死锁;把资源链接在错误的队列中等等。

- 加工错误

- 算术与操作错误：指在算术运算、函数求值和一般操作过程中发生的错误。包括：数据类型转换错；除法溢出；错误地使用关系比较符；用整数与浮点数做比较等。
- 初始化错误：典型的错误有：忘记初始化工作区，忘记初始化寄存器和数据区；错误地对循环控制变量赋初值；用不正确的格式，数据或类型进行初始化等等。
- 控制和次序错误：这类错误与系统级同名错误类似，但它是局部错误。包括：遗漏路径；不可达到的代码；不符合语法的循环嵌套；循环返回和终止的条件不正确；漏掉处理步骤或处理步骤有错等。
- 静态逻辑错误：这类错误主要包括：不正确地使用CASE语句；在表达式中使用不正确的否定（例如用“>”代替“<”的否定）；对情况不适当地分解与组合；混淆“或”与“异或”等。

- 数据错误

- **动态数据错误:** 动态数据是在程序执行过程中暂时存在的数据。各种不同类型的动态数据在程序执行期间将共享一个共同的存储区域，若程序启动时对这个区域未初始化，就会导致数据出错。由于动态数据被破坏的位置可能与出错的位置在距离上相差很远，因此要发现这类错误比较困难。
- **静态数据错误:** 静态数据在内容和格式上都是固定的。它们直接或间接地出现在程序或数据库中。由编译程序或其它专门程序对它们做预处理。这是在程序执行前防止静态错误的好办法，但预处理也会出错。
- **数据内容错误:** 数据内容是指存储于存储单元或数据结构中的位串、字符串或数字。数据内容本身没有特定的含义，除非通过硬件或软件给予解释。数据内容错误就是由于内容被破坏或被错误地解释而造成的错误。
- **数据结构错误:** 数据结构是指数据元素的大小和组织形式。在同一存储区域中可以定义不同的数据结构。数据结构错误主要包括结构说明错误及把一个数据结构误当做另一类数据结构使用的错误。这是更危险的错误。
- **数据属性错误:** 数据属性是指数据内容的含义或语义。例如，整数、字符串、子程序等等。数据属性错误主要包括：对数据属性不正确地解释，比如错把整数当实数，允许不同类型数据混合运算而导致的错误等。

- 代码错误
 - 语法错误
 - 打字错误
 - 对语句或指令不正确理解所产生的错误

按软件生存期阶段分类

- 问题定义（需求分析）错误
 - 它们是在软件定义阶段，分析员研究用户的要求后所编写的文档中出现的错误。换句话说，这类错误是由于问题定义不满足用户的要求而导致的错误。

-
- 规格说明错误
 - 这类错误是指规格说明与问题定义不一致所产生的错误。它们又可以细分成：
 - 不一致性错误：规格说明中功能说明与问题定义发生矛盾。
 - 冗余性错误：规格说明中某些功能说明与问题定义相比是多余的。
 - 不完整性错误：规格说明中缺少某些必要的功能说明。
 - 不可行错误：规格说明中有些功能要求是不可行的。
 - 不可测试错误：有些功能的测试要求是不现实的。

- 设计错误

- 这是在设计阶段产生的错误，它使系统的设计与需求规格说明中的功能说明不相符。它们又可以细分为：
- 设计不完全错误：某些功能没有被设计，或设计得不完全。
- 算法错误：算法选择不合适。主要表现为算法的基本功能不满足功能要求、算法不可行或者算法的效率不符合要求。
- 模块接口错误：模块结构不合理；模块与外部数据库的界面不一致，模块之间的界面不一致。
- 控制逻辑错误：控制流程与规格说明不一致；控制结构不合理。
- 数据结构错误：数据设计不合理；与算法不匹配；数据结构不满足规格说明要求。

-
- 编码错误
 - 编码过程中的错误是多种多样的，大体可归为以下几种：数据说明错、数据使用错、计算错、比较错、控制流错、界面错、输入/输出错，及其它的错误。

缺陷报告

好的缺陷报告

- 概要
- 再现步骤
- 隔离

概要

- 使用一两句话来描述错误，给客户或系统用户留下深刻印象
- 告诉经理、开发人员以及其他读者为什么他们应该关心该问题
- 简洁、切中要害

- 好与坏比较

- “屏幕分辨率有问题”
- “设定屏幕分辨率为800*1024，屏幕不能刷新”

再现步骤

- 准确的描述
- 步骤顺序
- 简明但完全，不含糊且准确
- 环境配置信息
- 发现几率，例如“重复以上步骤四次，三次观察到错误的发生”

隔离

- 确认错误是一个真正的问题
- 识别影响错误表现的因素而收集的结果和信息，即改变了操作行为，测试的结果是什么样子的，例如：不能用激光打印机打印，能用喷墨打印机打印吗？

良好的错误报告

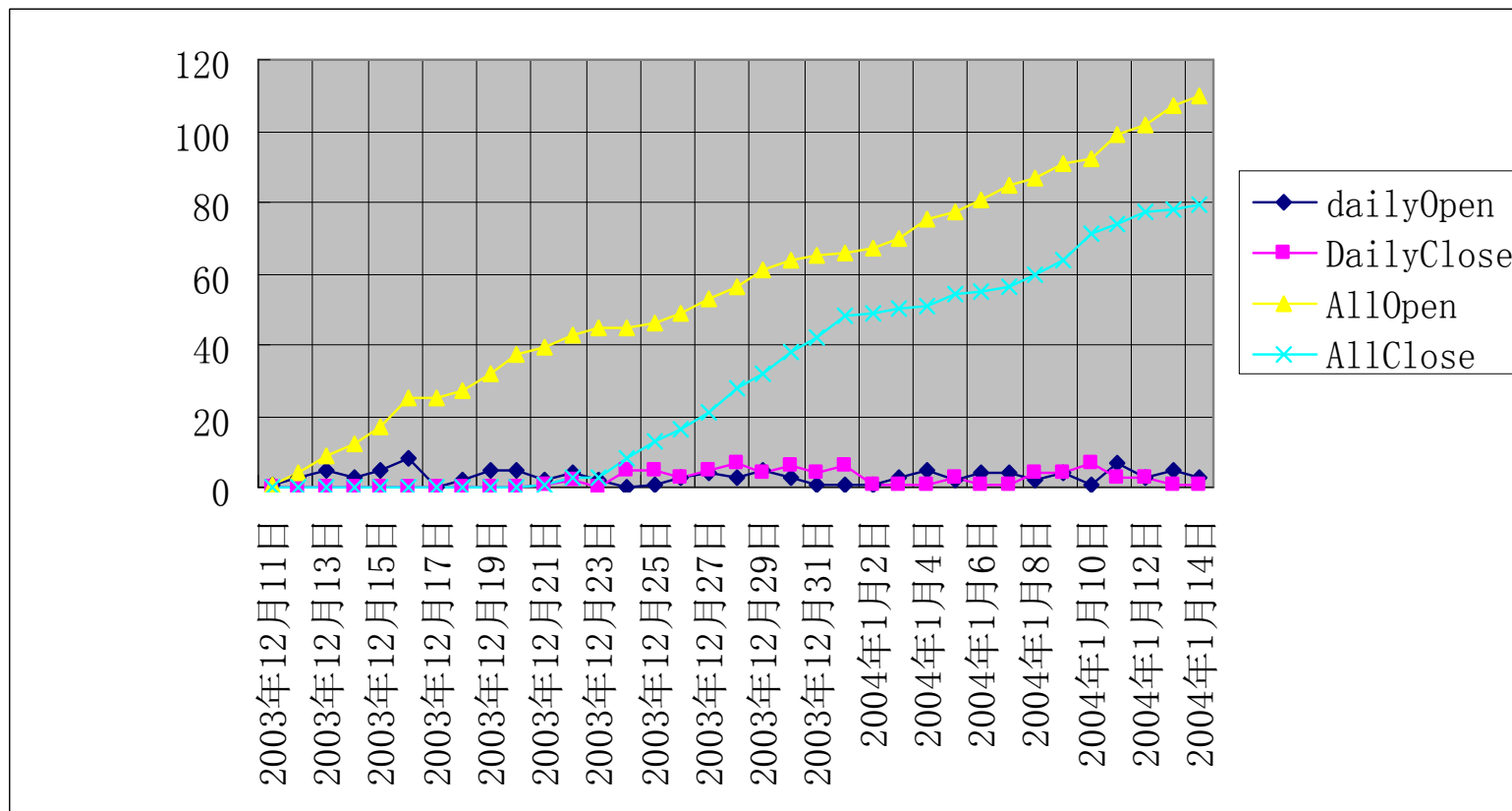
- 摘要
 - Arial、Wingdings和Symbol字体会破坏新文件
- 再现步骤
 - 1、启动SpeedyWriter编辑器，然后创建新文件
 - 2、输入四行文本，重复输入“The quick fox jumps over the lazy brown dog.”
 - 3、选中所有四行文本，然后选择字体下拉菜单，并选择Arial。
 - 4、所有文本被转换成控制字符、数字和其他明显的随机二进制数据
 - 5、重复三次，结果都一样
- 隔离
 - 新建版本1.1.018；同样的测试用例在1.1.007到1.1.017上都通过。对Wingdings和Symbol字体重复相同的步骤
 - 粗略估计是格式问题，保存文件，关闭SpeedyWriter并重新打开文件，但是数据仍然被破坏
 - 对现存文件，错误不发生
 - 只在Windows98下发生，而不出现在Solaris、Mac或其他版本的Windows系统

含糊不完整的错误报告

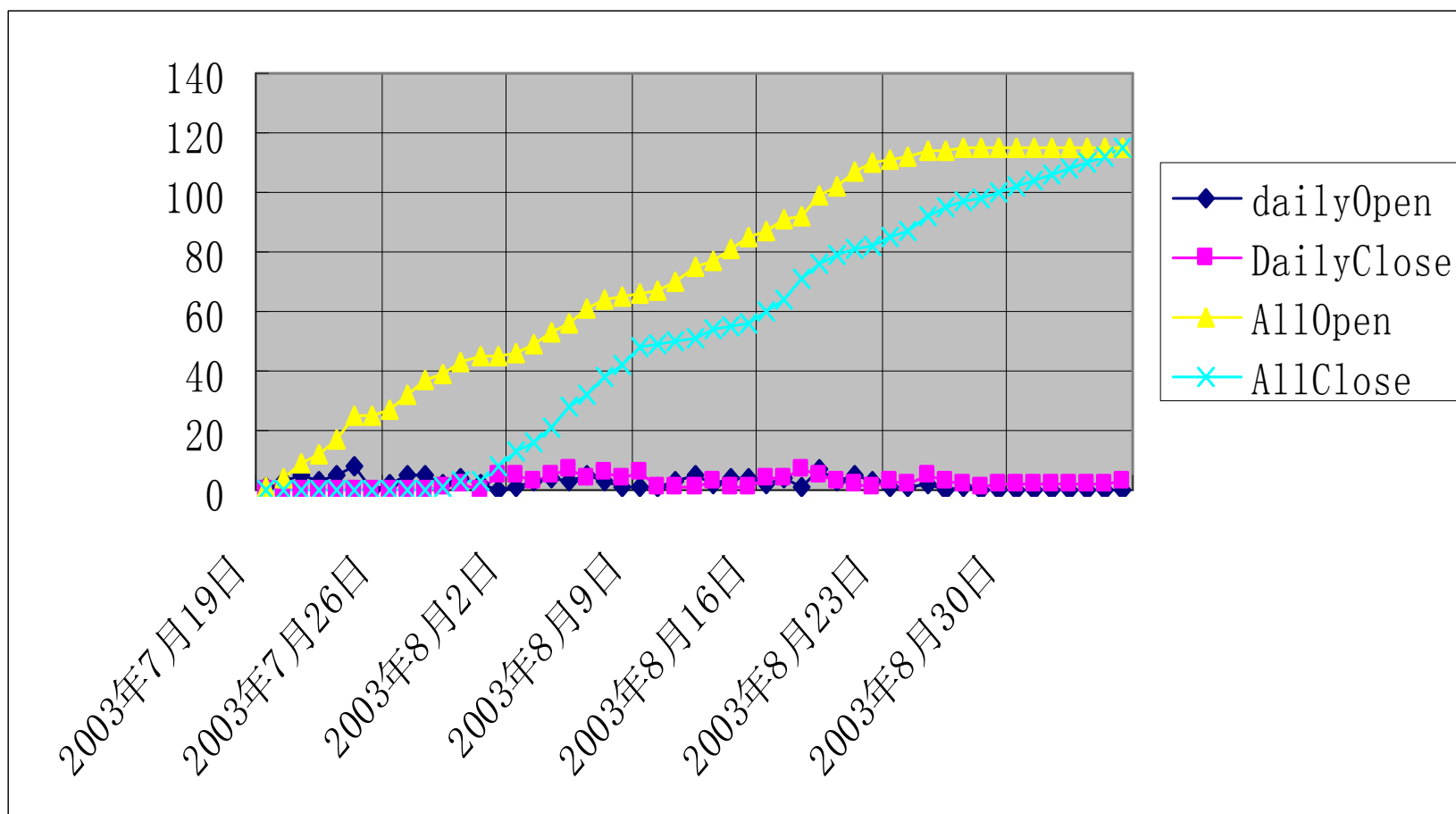
- 概要
 - SpeedyWriter处理Arial字体时有问题
- 再现步骤
 - 1、打开SpeedyWriter
 - 2、输入一些文本
 - 3、选择Arial
 - 4、文本被破坏
- 隔离
 - N/A

缺陷趋势

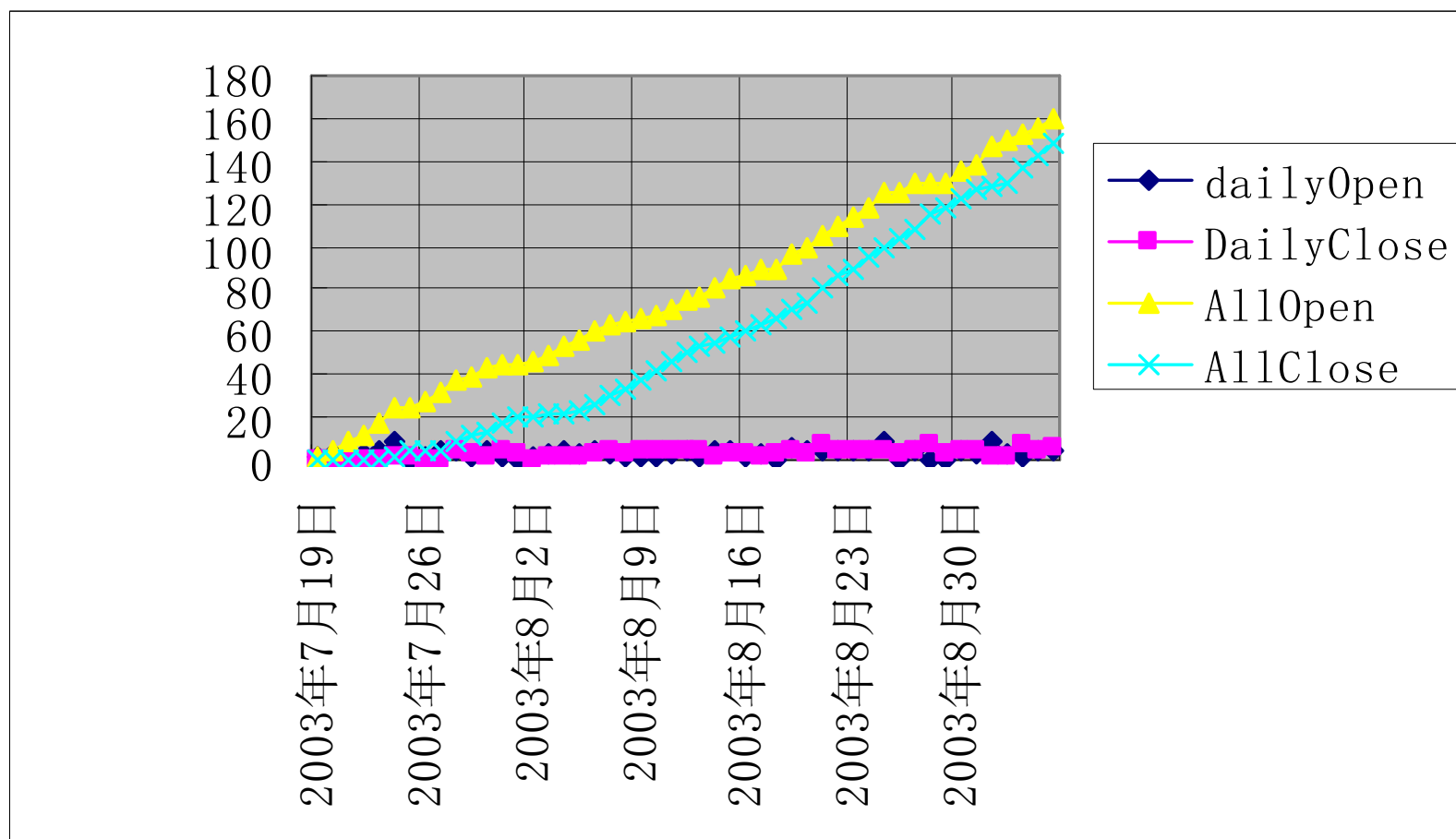
Open&Close图形



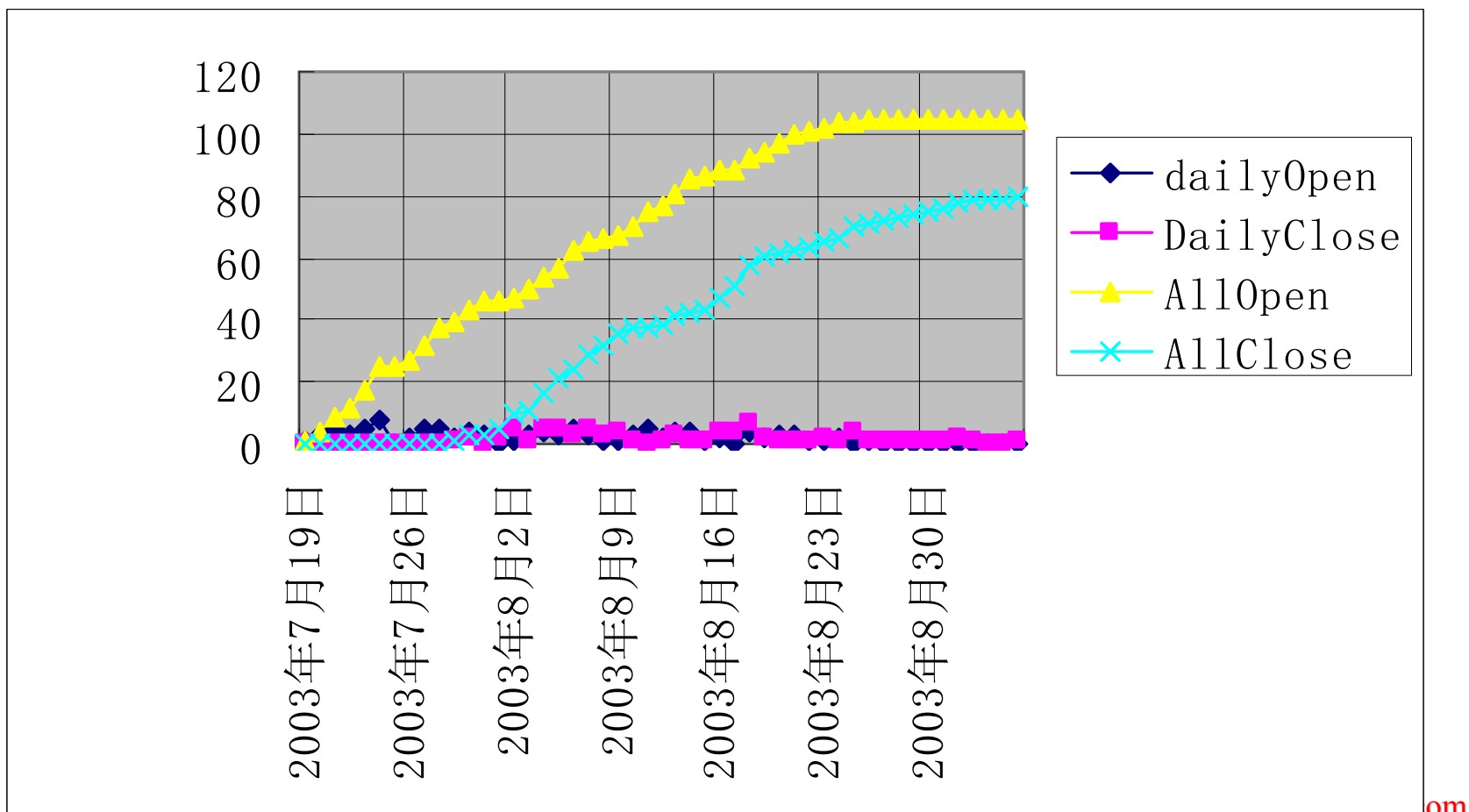
理想Open&Close图形



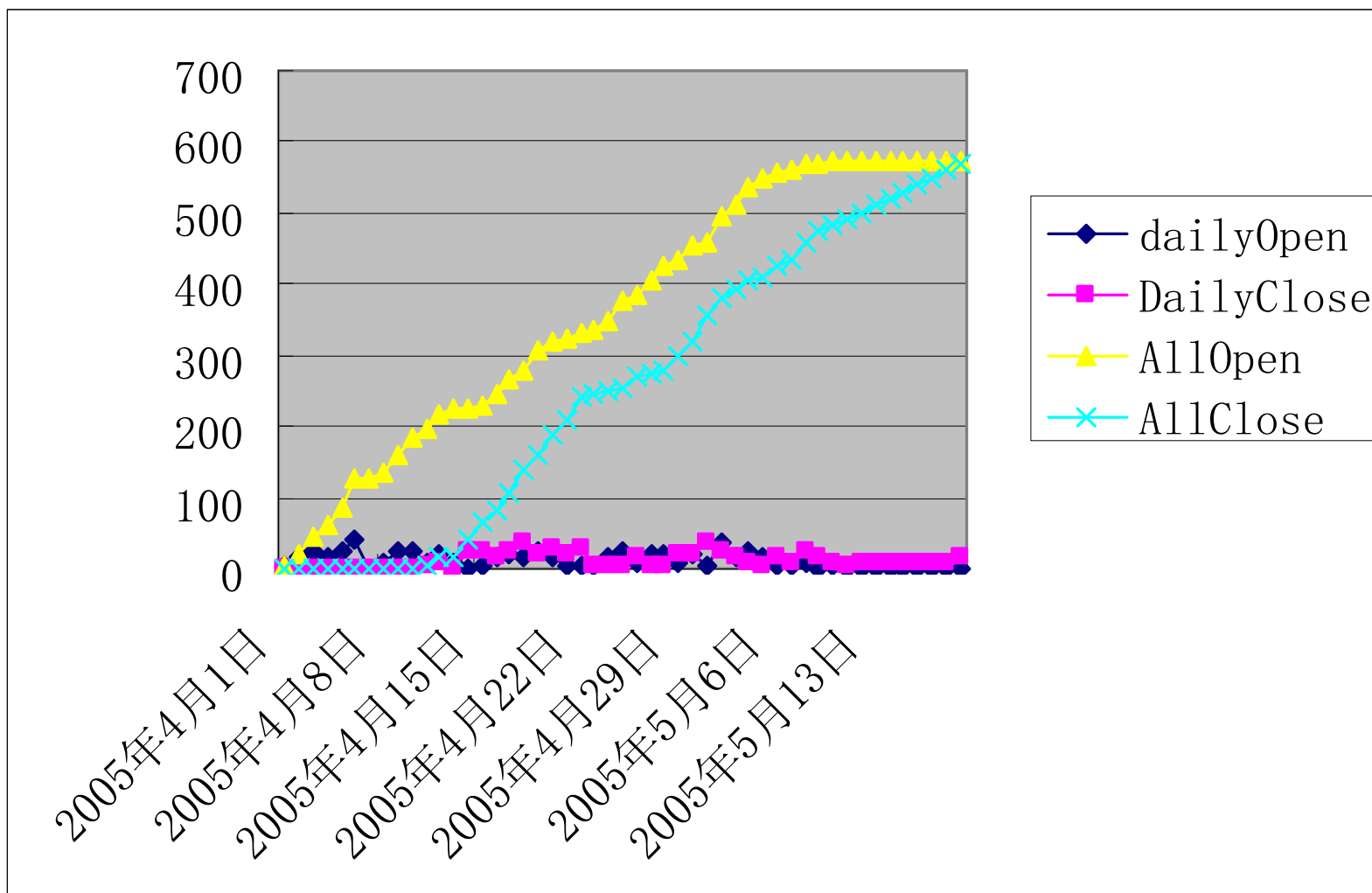
无休止的Open&Close图形



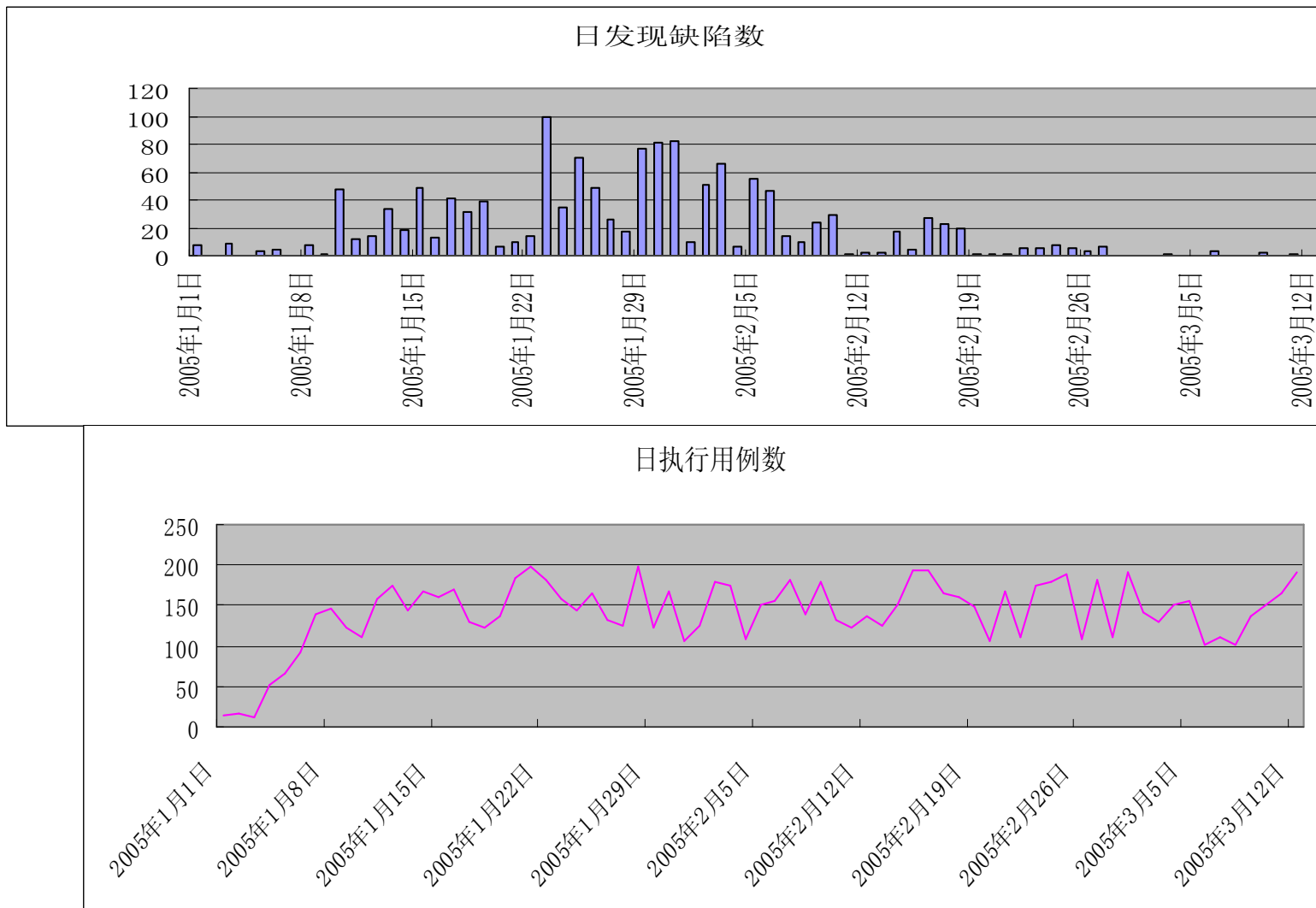
缺陷遗漏图形



全部解决图形

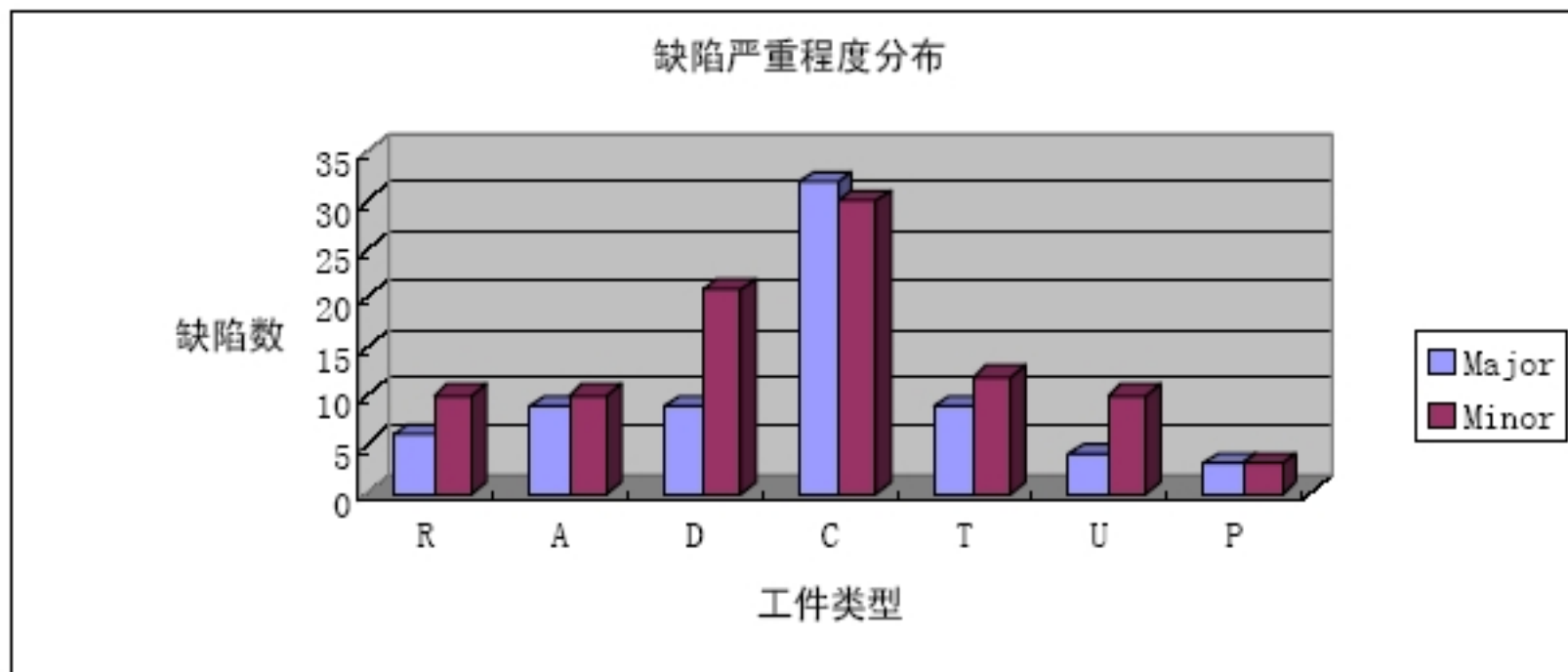


缺陷发现趋势图形

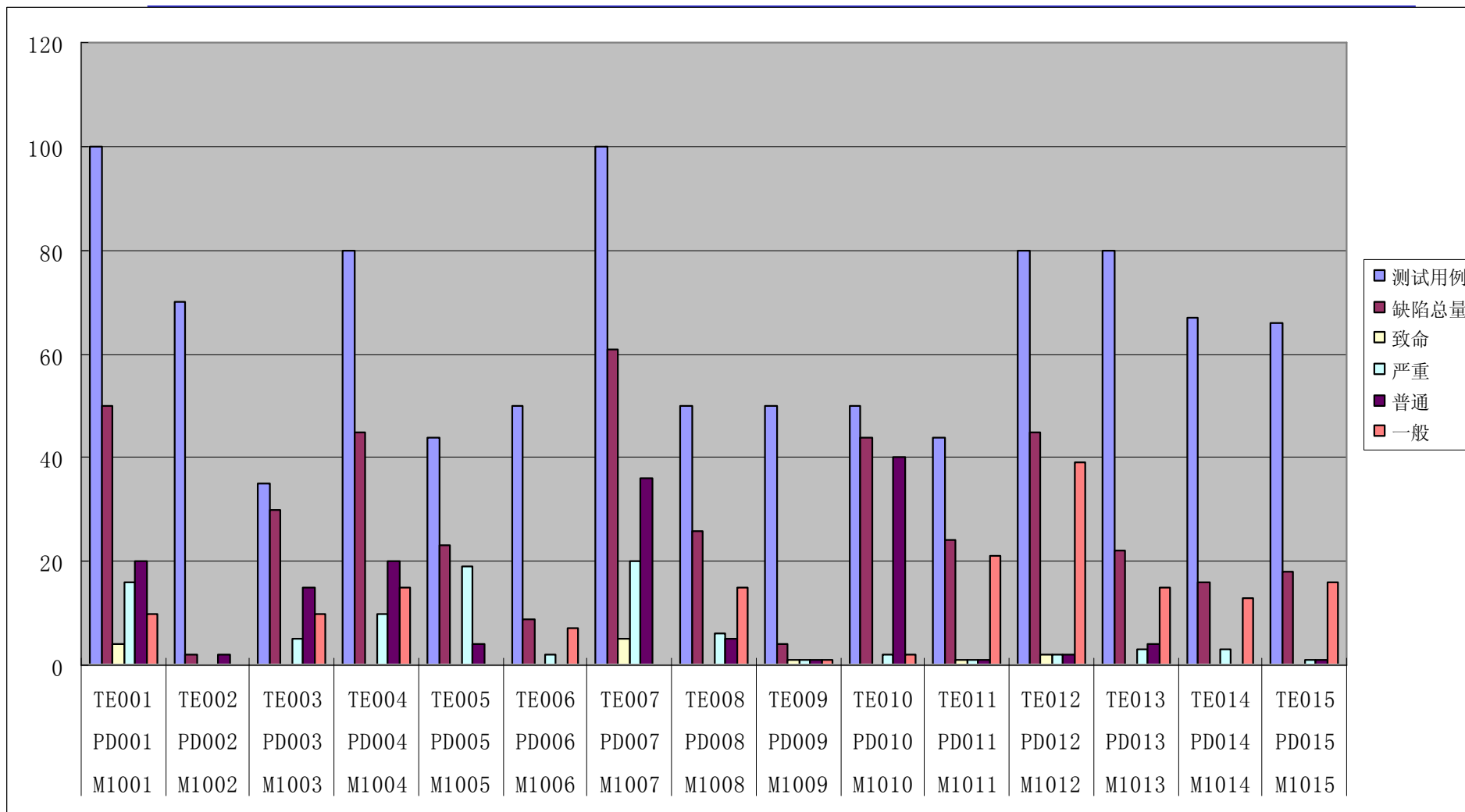


缺陷统计

缺陷严重程度分布统计



模块-开发人员-测试员-缺陷统计



模块缺陷率统计

模块代号	缺陷数	N(KLOC)	缺陷数/KLOC
M 1001	16	5	3.2
M 1002	19	12	1.6
M 1003	30	7	4.2
M 1004	62	20	3.1
M 1005	21	1	21
M 1006	14	3	4.7
M 1007	6	2	3

5年以下的工程师：缺陷数/KLOC > 12
12年以下的工程师：缺陷数/KLOC < 6

项目缺陷率统计

