

SilkTest International Tutorial

长沙高新开发区思诚信息技术服务有限公司

Contents

- ◆ 第一章 SilkTest 简介
- ◆ 第二章 预备工作
- ◆ 第三章 测试计划
- ◆ 第四章 应用捕获(Capturing Your Application)
- ◆ 第五章 创建测试用例(Creating Testcases)
- ◆ 第六章 运行测试用例(Running Testcases)
- ◆ 第七章 测试结果

第一章 SilkTest 简介

- 一、自动化测试
- 二、SilkTest基本概念
- 三、SilkTest特点
- 四、SilkTest测试主要流程
- 五、SilkTest中主要文件类型
- 六、SilkTest主要窗口
- 七、使用SilkTest进行Web测试

第一章 SilkTest 简介

一、自动化测试

■ 人工测试

- ◆ 费时费力
- ◆ 测试一致性难以保证
- ◆ 项目时间限制，难以进行充分测试

■ 自动化测试的意义

- ◆ 回归测试的测试覆盖程度高
- ◆ 降低测试时间，加快产品面市周期
- ◆ 改善人工测试效率
- ◆ 改善测试重用性
- ◆ 具有详细的测试记录

第一章 SilkTest 简介

一、自动化测试

- 测试最常用的方法是采用：分层次的测试方法(The Layered Approach)
 - ◆ Operability Tests (可操作性测试:each object的: state、size、caption、contents)
 - ◆ Functionality Tests (功能测试:several objects组合在一起提供的特性)
 - ◆ System Tests (系统测试:被测软件在其他软、硬件环境下的表现)
- 其他类型的测试
 - ◆ Regression Tests (回归测试:existing tests for new version of program)
 - ◆ Error Tests (错误处理测试: system response for error condition)
 - ◆ Stress Tests (压力测试:measure system response under repetitive or large amounts of data)
 - ◆ White-Box Tests (白盒测试:internal structure of software)
 - ◆ Black-Box Tests (黑盒测试:view software from end-user perspective and is unaware of the underlying code)

第一章 SilkTest 简介

二、SilkTest基本概念

■ SilkTest如何工作

- ◆ 在SilkTest看来，应用由许多GUI对象组成，用户通过键盘/鼠标与这些对象交互
- ◆ 使用SilkTest时，SilkTest通过面向对象的概念（类/属性/方法），识别这些GUI对象
- ◆ 测试时，SilkTest模拟用户的操作，与这些GUI对象交互，并记录测试结果

第一章 SilkTest 简介

二、SilkTest基本概念

■ SilkTest组成

- ◆ SilkTest Host (Host Machine)
 - 用于develop/edit/compile/run/debug 测试脚本和测试计划
- ◆ SilkTest Agent (Remote Machine)
 - interact with GUI of AUT (Application Under Test 被测应用)
 - translate script into GUI commands
 - driving and monitoring application
 - run locally or remote



第一章 SilkTest 简介

二、SilkTest基本概念

- SilkTest如何记录用户行为
 - ◆ SilkTest repository
 - 用于记录应用中各种GUI对象的描述
 - ◆ SilkTest自动识别GUI对象，自动记录用户操作过程中对各个GUI对象的操作，将这些操作转换成相应的方法，写入脚本程序中。

第一章 SilkTest 简介

二、SilkTest基本概念

■ SilkTest如何记录用户行为

- ◆ 映射举例 (SilkTest Tutorial.pdf P12)

What you do	What SilkTest records
You pick a menu item	Pick
You check a check box	Check
You uncheck a check box	Uncheck
You write text in a text field	SetText
You scroll a scrollbar to the maximum position	ScrollToMax
You select an item from a list box	Select
You close a dialog box	Close
You set the main window active	SetActive
You select a radio button from a group	Select

第一章 SilkTest 简介

二、SilkTest基本概念

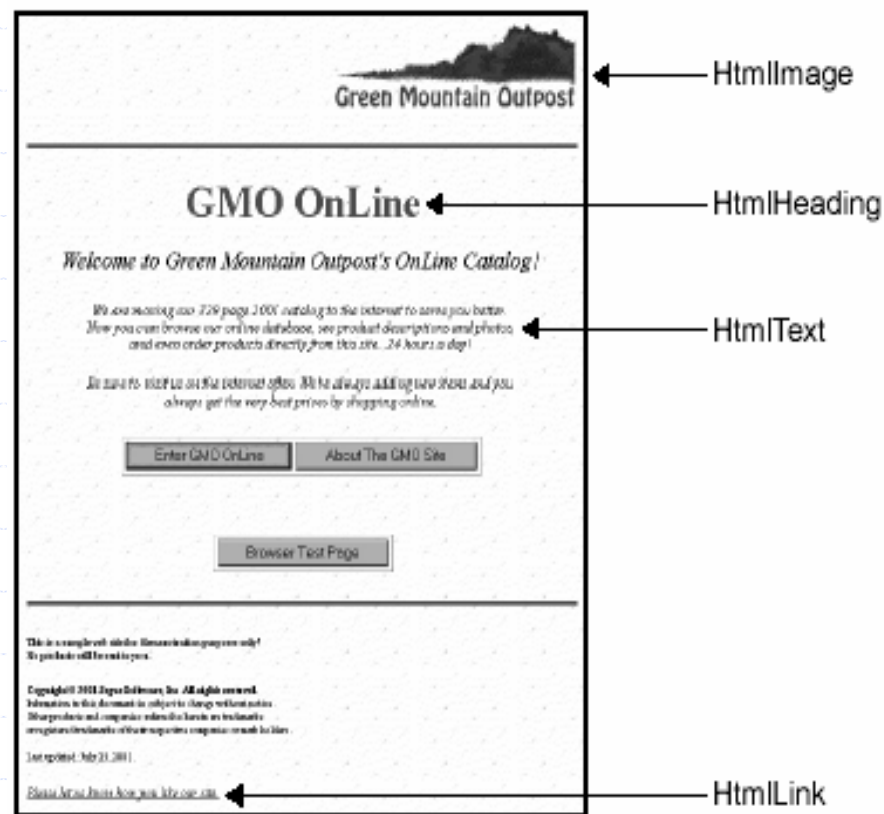
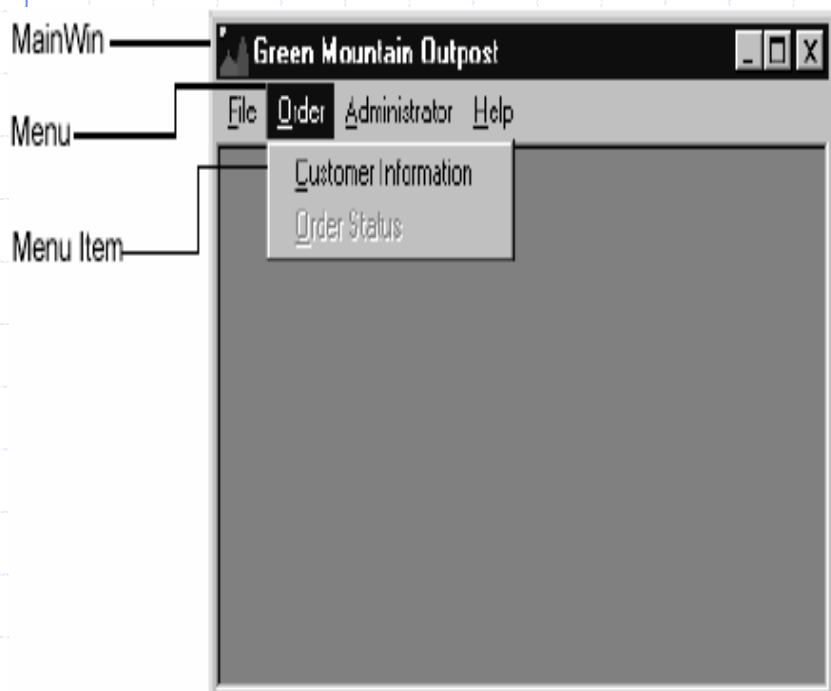
- 4Test 脚本语言
 - 面向对象的4GL脚本语言
 - 大量面向对象的类库
 - Statements/operators/data type
 - 举例 (SilkTest Tutorial.pdf P13-16)

第一章 SilkTest 简介

二、SilkTest基本概念

■ 4Test 脚本语言

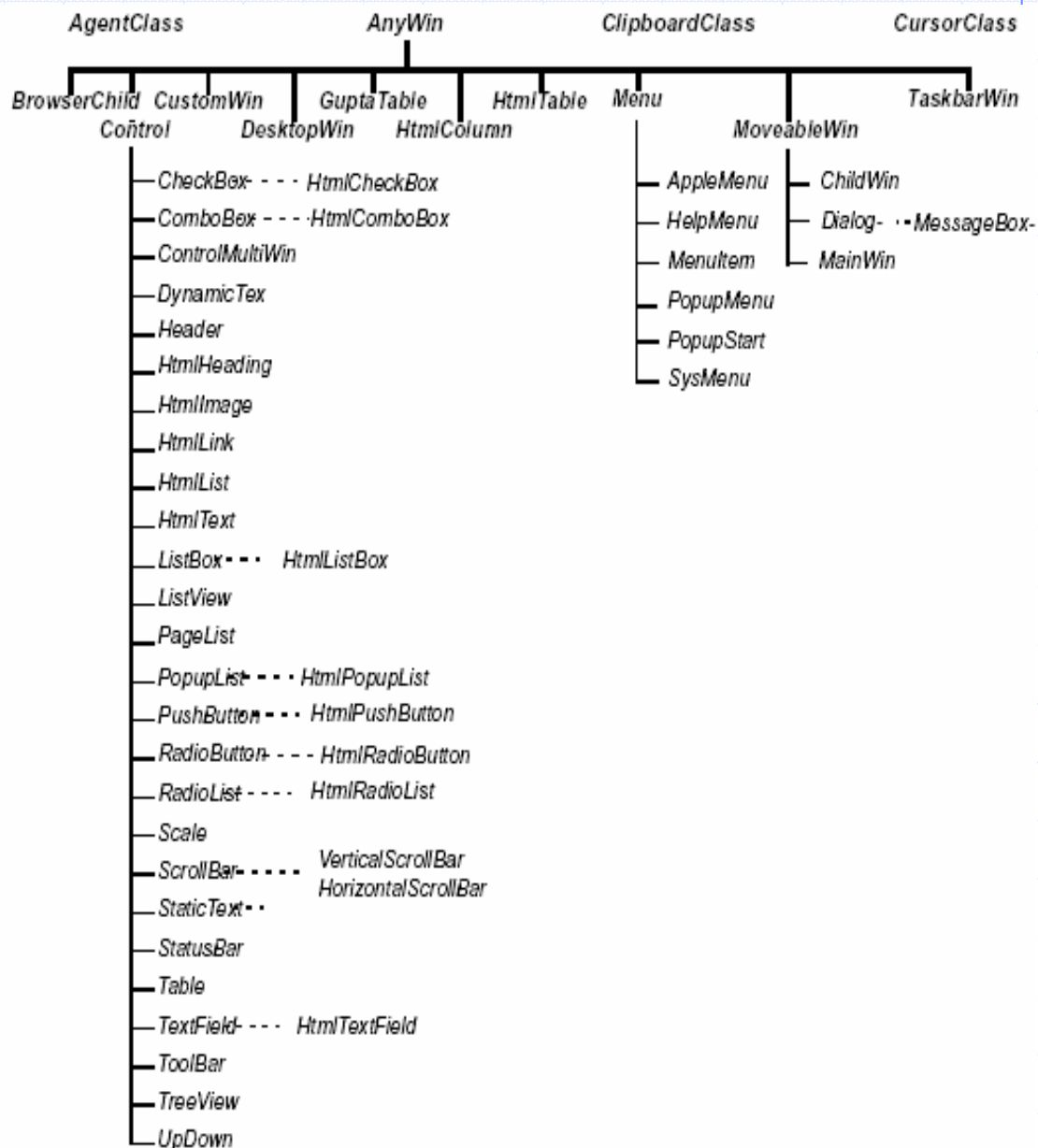
◆ Classes 类



第一章 SilkTest 简介

二、SilkTest 基本概念

- 4Test 脚本语言
 - ◆ Classes 类



第一章 SilkTest 简介

二、SilkTest基本概念

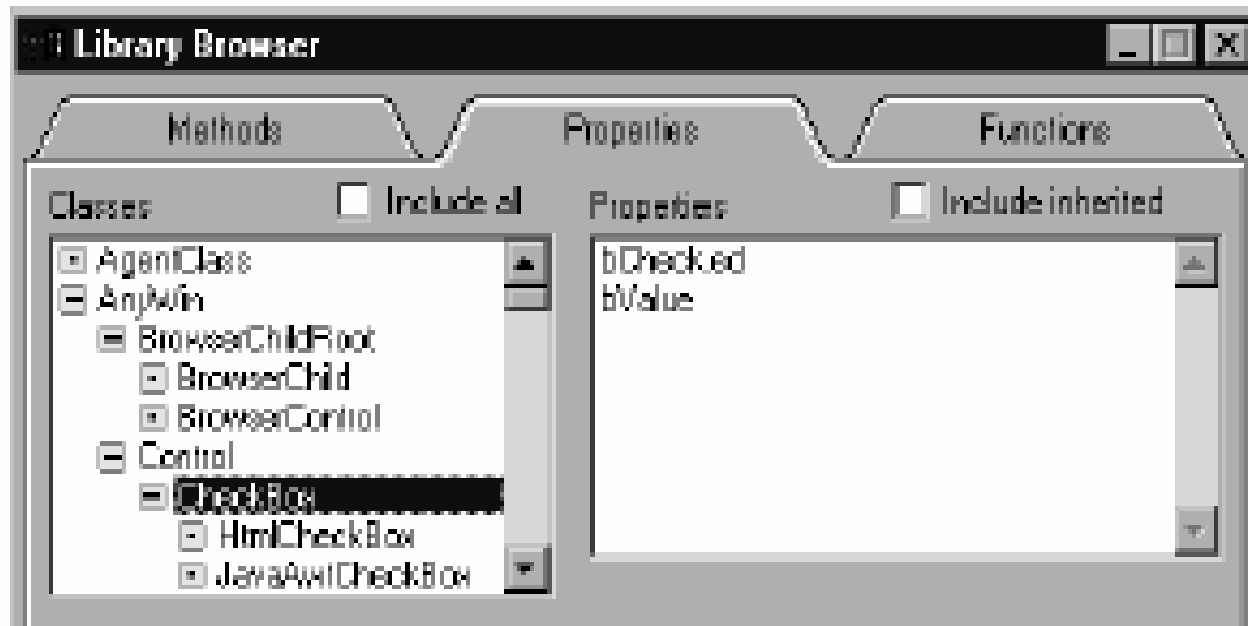
- 4Test 脚本语言
 - ◆ Objects



第一章 SilkTest 简介

二、SilkTest 基本概念

- 4Test 脚本语言
 - ◆ Properties



第一章 SilkTest 简介

二、SilkTest基本概念

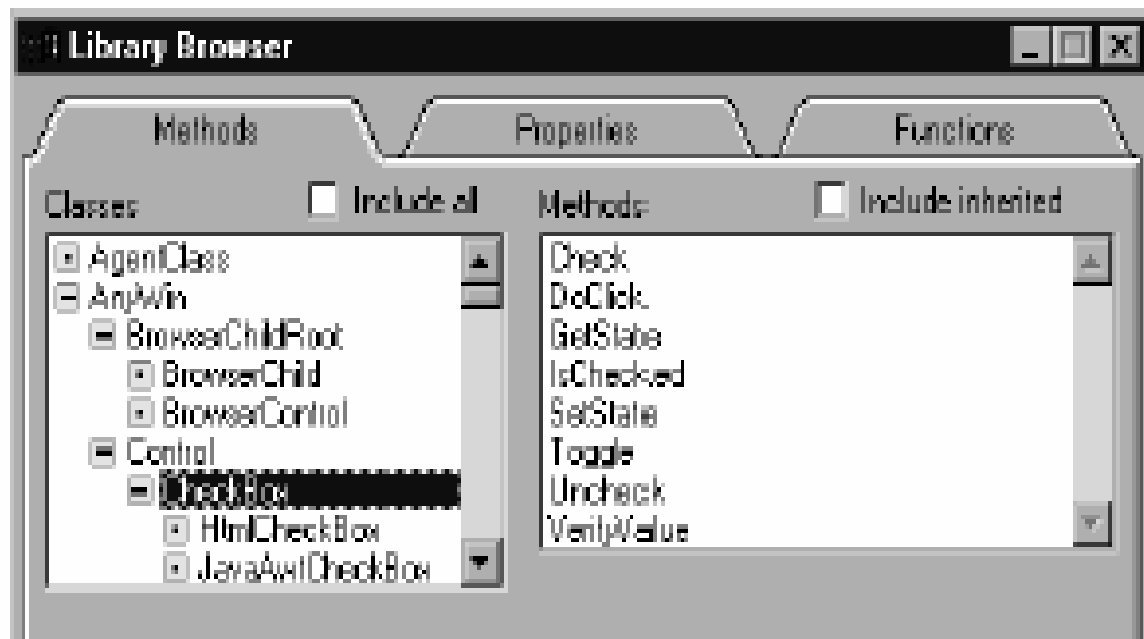
- 4Test 脚本语言
 - ◆ Properties

Characteristic	Property Name
Whether a check box is checked	State
Whether an object is enabled	Enabled
The text of the selected item in a radio list	SelText
The index of the selected item in a radio list	SelIndex
The caption of a window	Caption
The text in the text field	Text
The highlighted text in a text field	SelText
The size and position of a window	Rect

第一章 SilkTest 简介

二、SilkTest基本概念

- 4Test 脚本语言
 - ◆ Methods



第一章 SilkTest 简介

二、SilkTest基本概念

- 4Test 脚本语言
 - ◆ Methods

User Action	Method Name
You click on a push button	Click ()
You pick a menu item	Pick ()
You check a check box	Check ()
You clear a check box	UnCheck ()
You type text into a text field	SetText ()
You read the value of a text field	GetText ()
You choose an item from a list box	Select ()
You count the number of items in a list box	GetItemCount ()
You close a dialog box	Close ()
You set the MainWin active	SetActive ()
You choose an item from a radio list	Select ()

第一章 SilkTest 简介

二、SilkTest 基本概念

■ 内置的恢复系统(Recovery System)

- ◆ 当某一测试用例失败，导致应用崩溃时，Recovery System能自动重新启动应用，使应用回复到测试的初始状态，继续进行下一个测试用例测试。
- ◆ 用于保证实现真正的无人干预的自动化测试

第一章 SilkTest 简介

三、SilkTest特点

- SilkTest提供了针对C/S应用、Web应用的完整的自动化测试方案。可以：
 - ◆ Functional, operability and regression testing
 - ◆ 测试计划管理 testplan editor
 - ◆ 数据库访问校验 DBTester
- SilkTest一般特点
 - ◆ 容易录制
 - 通过最终用户操作一遍，录制生成4Test脚本
 - 属性的校验通过简单的点选方式，录制时自动产生相应的语句
 - ◆ 真正的对象识别
 - ◆ 内置的Recovery System保证实现无人干预的自动化测试
 - ◆ 集成测试计划管理与维护功能
 - ◆ 数据库测试：通过DBTester可以在脚本中使用SQL直接访问数据库，辅助进行校验
 - ◆ 完全分布式测试：可以针对C/S应用进行测试

第一章 SilkTest 简介

三、SilkTest特点

- Web应用测试特点
 - ◆ 独立于各种浏览器和平台
 - ◆ 独立于各种Web技术
 - ◆ Multiple testing options using browser extensions
 - DOM(Document Model Object)方式访问Web页面的各种对象
 - SilkTest自有的VO(Virtual Object)方式访问Web页面的各种对象

第一章 SilkTest 简介

四、SilkTest测试主要流程



Phase	Action	Description
I	Plan	Determine the testing strategy and define specific test requirements.
II	Capture	Classify the GUI objects in your application and build a framework for running your tests.
III	Create	Create automated, reusable tests. Use recording and/or programming to build test scripts written in Segue's 4Test language.
IV	Run	Select specific tests and execute them against the AUT.
V	Report	Analyze test results and generate defect reports.
VI	Track	Track defects in the AUT and perform regression testing.

第一章 SilkTest 简介

四、SilkTest测试主要流程

■ Phase I Plan 计划阶段

- ◆ 确定测试策略、测试需求
- ◆ 使用test plan editor进行计划管理
 - test plan editor采用分级的树状结构，包含有如下信息：
 - **Description:** 每一个/组测试的描述信息
 - **Statements:** 将描述信息与测试用例关联的语句
 - **Attributes:** 针对每一个/组测试的描述信息定义属性，用于测试运行和生成报表
- ◆ 实际上，对于大型、复杂的应用来说，将所有可能的路径进行全部测试，在时间或金钱上往往是得不偿失的。因此，必须能够识别应用中的至关重要的部分，并对这些部分保证进行充分的测试，达到很高的测试覆盖。这往往要借助于：
 - 客户的需求
 - 设计文档
 - 功能描述
 - 市场需求
 - 产品的相关文档
- ◆ 建议：在创建测试用例之前，制定完整的测试策略和测试计划

第一章 SilkTest 简介

四、SilkTest测试主要流程

- Phase II Capture 捕获应用主要信息阶段
 - ◆ 创建被测应用(AUT)的测试框架(Test Frame)
 - ◆ Test Frame是一个central repository, 含有被测应用的GUI数据结构信息, 这些信息用于支持测试脚本的运行, 包括:
 - Web应用的URL; C/S应用的command line信息
 - Constants and Variables
 - AUT中, 所有GUI对象的declarations
- Phase III Create 创建测试用例阶段
 - ◆ 通过录制生成/编写4Test测试脚本, 创建测试用例(TestCase)
 - ◆ 一个测试用例应完成如下:
 - 驱动被测应用从确定的初始状态开始进行测试
 - 确认应用实际运行后的状态与期待状态是否吻合
 - 将应用返回到确定的初始状态

第一章 SilkTest 简介

四、SilkTest测试主要流程

- Phase IV Run 运行阶段
 - ◆ 运行测试用例
 - 单独运行一个脚本
 - 一组脚本(a suit)
 - 从testplan中挑选, 通过testplan中的attributes
 - 整个testplan
- Phase V Report 生成测试报告
 - ◆ 分析测试结果, 产生缺陷报告
 - ◆ SilkTest自动将测试结果生成到result文件中
 - ◆ 从result文件中, 可以产生pass/failed 报告
- Phase VI Track 缺陷跟踪
 - ◆ 识别AUT中的缺陷, 使用缺陷追踪系统进行缺陷处理跟踪
SilkRadar
 - ◆ 进行回归测试 Regression Test(用原有的测试用例对新版程序进行测试)

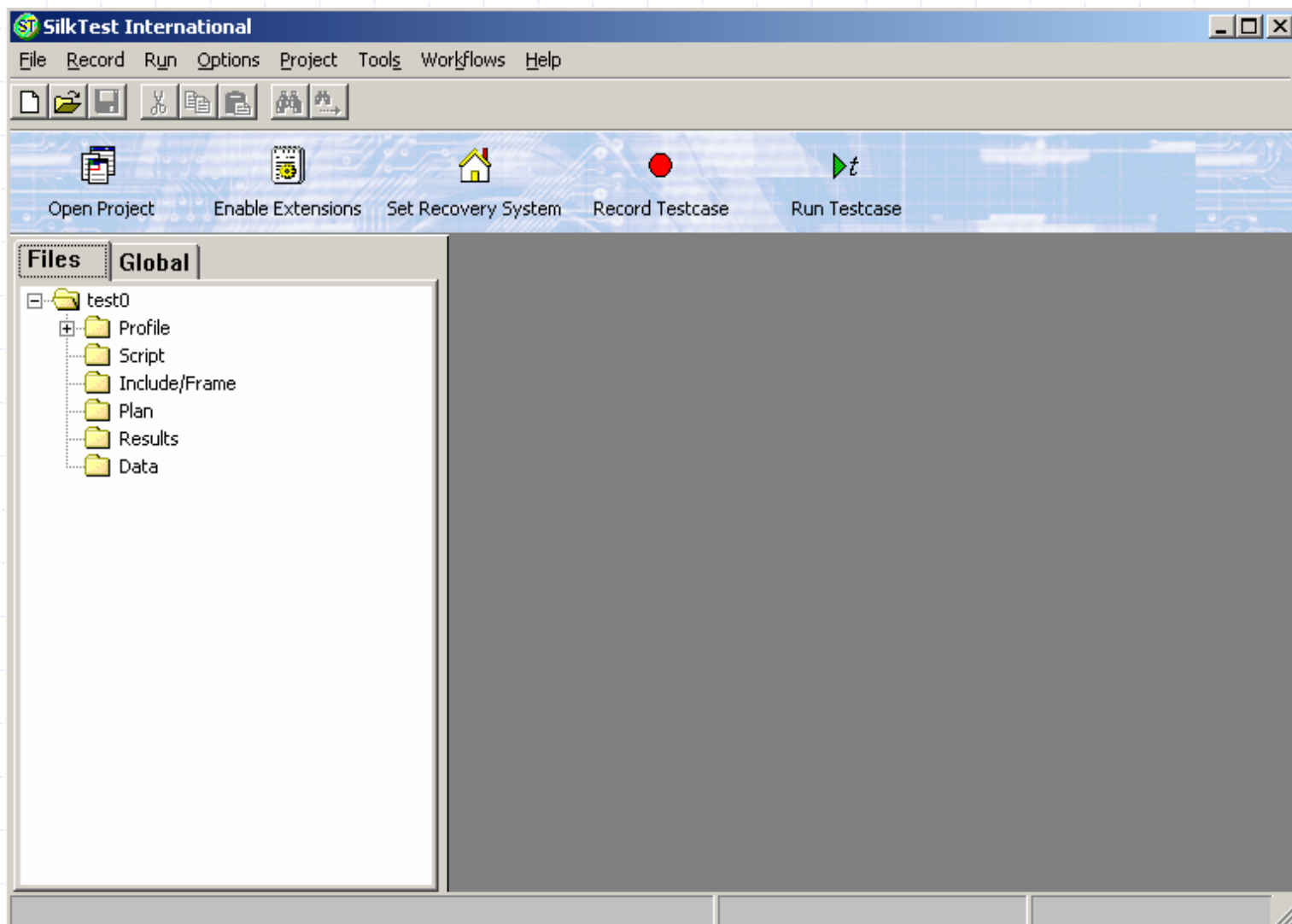
第一章 SilkTest 简介

五、SilkTest中主要文件类型 (P20)

文件类型	说明	后缀
Test Plan	测试计划说明	.pln
Test Frame	AUT的的测试框架信息，包括：main window declarations, URL/command line, 其他信息, 用于支持录制与运行	.inc
4Test Script	通过4Test脚本语言实现的测试用例程序	.t
4Test Include File	Include文件，包含各种Windows declarations, constants, variables, classes, user-defined functions用于支持录制与运行	.inc
Suite	顺序执行的一组脚本程序	.s
Text File	文本文件	.txt
Result File	存放测试结果	.res
Initialization File	初始化文件	.ini

第一章 SilkTest 简介

六、SilkTest主要窗口 (P22-23)



第一章 SilkTest 简介

六、SilkTest主要窗口 (P22-23)

- SilkTest将主要操作流程用操作向导条进行固化，简捷易用



第一章 SilkTest 简介

七、使用SilkTest进行Web测试

■ Web应用测试的挑战

- ◆ 运行于各种不同平台的不同浏览器上
- ◆ 内容总是不断变化的，每周/每天/每刻
- ◆ 响应时间非常不一致
- ◆ 使用了各种Web技术

■ Browser Extensions

- ◆ Extension是一个文件，当AUT中含有非标准控件时，通过使用extension，SilkTest可以测试这些非标准控件
- ◆ 使用之前，必须enable host and target
- ◆ 针对Web应用，SilkTest提供两种extension
 - DOM(Document Object Model)方式访问Web页面的各种对象，W3C标准
 - SilkTest自有的VO(Virtual Object)方式访问Web页面的各种对象

第一章 SilkTest 简介

七、使用SilkTest进行Web测试

■ Browser Extension

- ◆ 当enable extension时，SilkTest会：
 - 将extension的include文件加入到Use Files中，确保extension的class可用
 - 确保extension的class说明被Library Browser可用。（通过将extension的帮助文件 .ht加到 Library Browser使用的文件中）
 - 将extension使用的property sets与SilkTest默认的property sets合在一起。

第二章 准备工作

- 一、安装SilkTest
- 二、测试Web应用之前的Setup
- 三、熟悉Web应用
- 四、测试C/S应用之前的Setup
- 五、熟悉C/S应用

第二章 准备工作

一、安装SilkTest

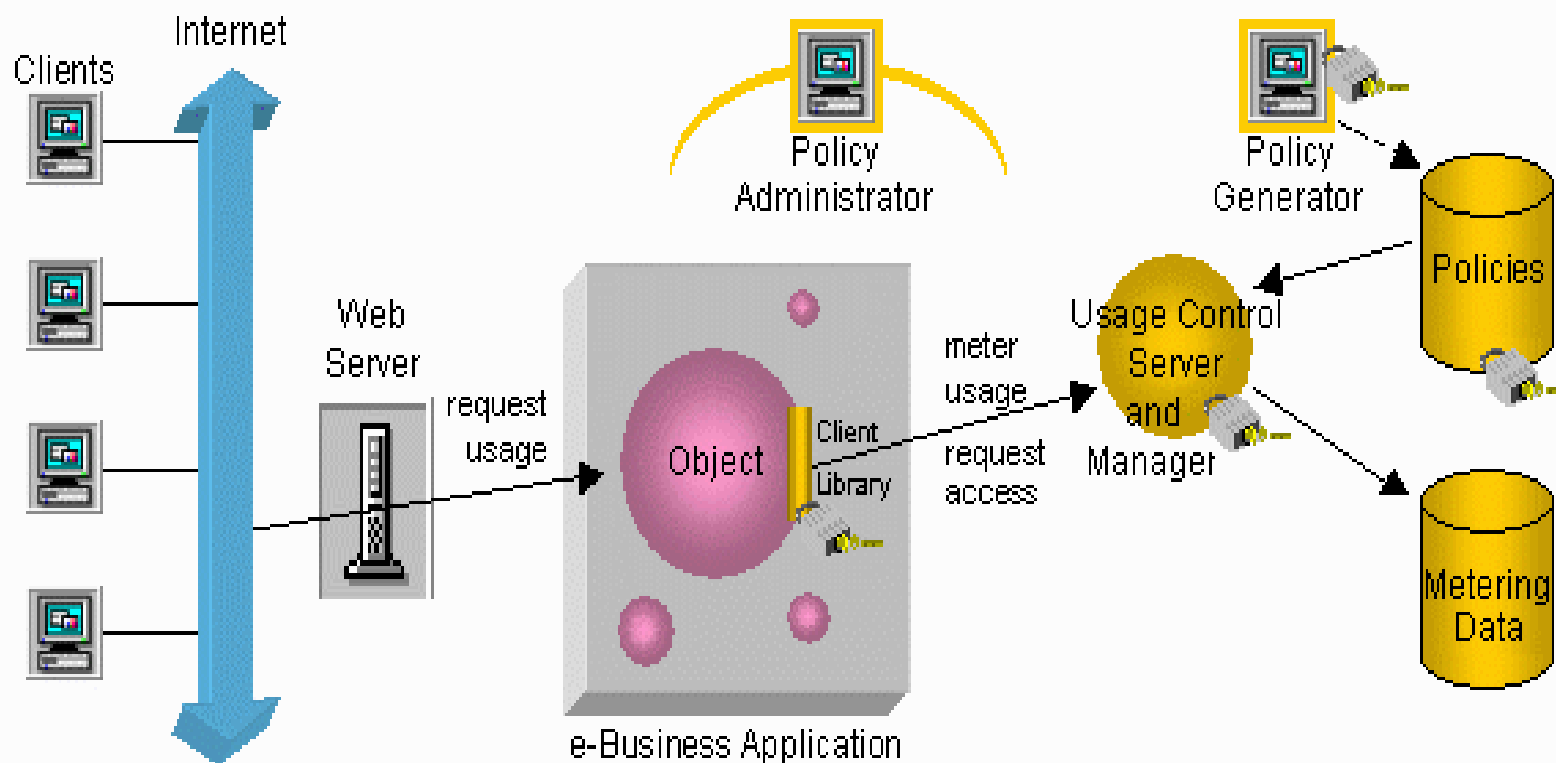
■ 安装SilkMeter

- ◆ SilkMeter 是Silk产品系列的License管理器
- ◆ 采用Check-In/Check-Out机制
- ◆ Local Host(Local Server)/Host ID(Remote Server)
- ◆ Pentium 166
- ◆ OS:
 - Remote Server: Windows98 / Windows Me / Windows 2000 /Windows NT4.0+SP5,6
 - Local Server: Windows 2000 /Windows NT4.0+SP5,6
- ◆ Memory: 32M(98、Me、NT4.0) / 64M(2000)
- ◆ HD: 30M
- ◆ SIILK_CONFIG_PATH 环境变量

第二章 准备工作

一、安装SilkTest

■ 安装SilkMeter



第二章 准备工作

一、安装SilkTest

- 安装SilkTest International
 - ◆ 默认 JDK 1.3
 - ◆ 确定 License Server
 - ◆ 安装 ...

第二章 准备工作

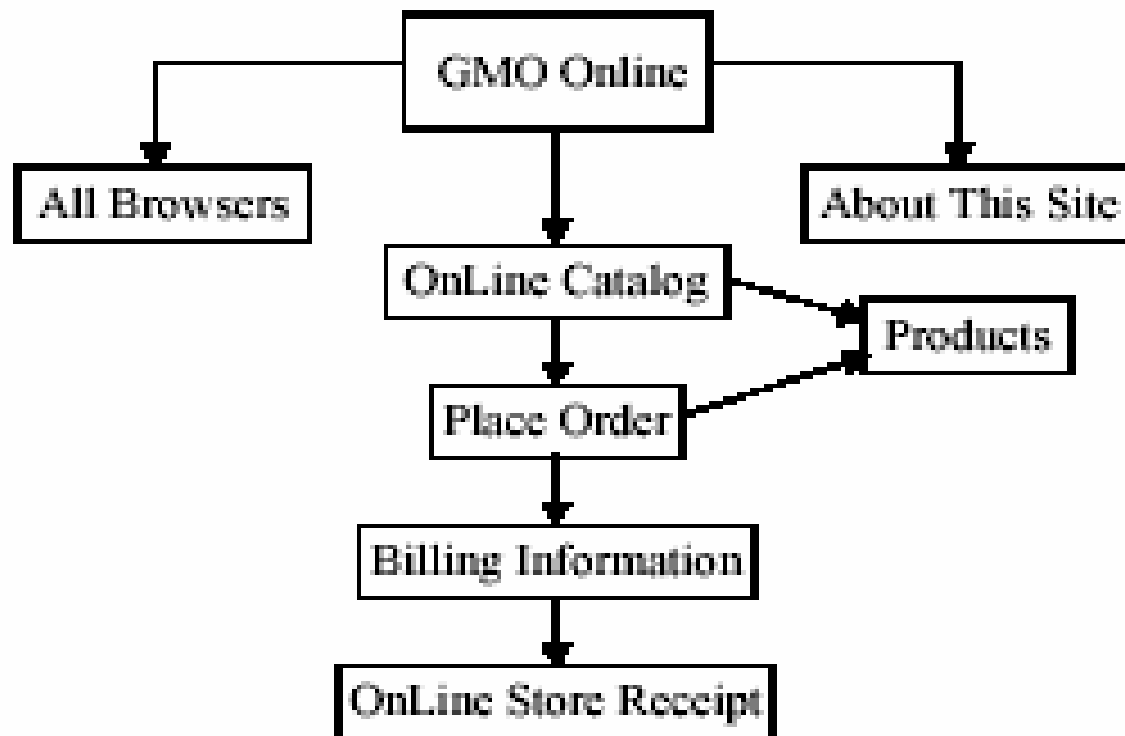
二、测试Web应用之前的Setup

- Enable support for the browser extensions
 - ◆ On a target machine Agent端的设置 enable extension 为IE DOM (P29)
 - ◆ On a host machine Host端的设置 enable extension 为IE DOM (P30)
- Setting Up the Runtime Environment for the Web Application
 - ◆ Specifying the Use Files 指定include 文件 P31-32
 - ◆ Specifying the default browser
- Configuring the Browser P33-35
- Setting Agent Options P35

第二章 准备工作

三、熟悉Web应用

- <http://gmo.seguate.com>



第二章 准备工作

三、熟悉Web应用

- 通过SilkTest识别页面目标 P₃₈₋₄₁
- Tag与 Multi-Tagging P₄₁

第二章 准备工作

四、测试C/S应用之前的Setup

- Installing GMO 应用
P42-43
- Setting Up the Runtime Environment for the C/S Application
 - ◆ Specifying the Use Files
指定include 文件 P44-45
 - ◆ Remove the default browser P46

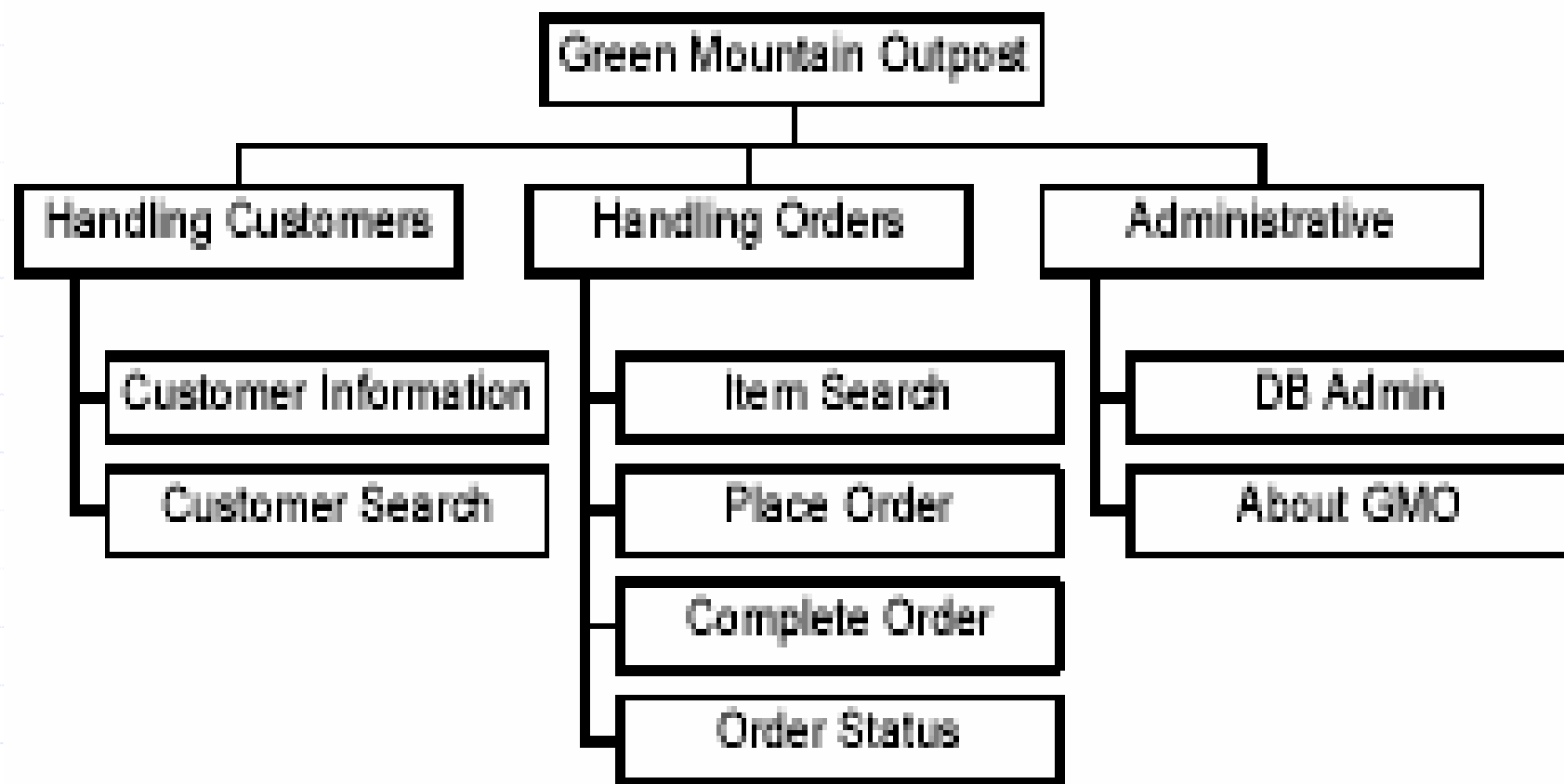
The screenshot shows the 'Runtime Options' dialog box for a program named '4Test'. The dialog is organized into several sections:

- Agent Name:** A dropdown menu set to '(local)'. **Network:** A dropdown menu set to '(disabled)'.
- Arguments:** An empty text input field.
- Use Path:** A text input field containing 'D:\Segue\gmo\Exercise'.
- Use Files:** A text input field containing 'D:\Segue\gmo\Exercise\gmoc.inc'.
- Objfile Path:** An empty text input field.
- Gui Targets:** An empty text input field.
- Default browser:** A dropdown menu set to '(none)'.
- Options:** Two checkboxes: 'Save object files during compilation' (unchecked) and 'Enable Y2K date transformation rules' (checked).
- Buttons:** 'Compiler constants...' and 'Y2K Rules...'.
- Results:** A section containing:
 - Directory/File:** An empty text input field.
 - History size:** A spin box set to '5'.
 - Find Error stops at warnings:** Checked.
 - Write to disk after each line:** Checked.
 - Show overall summary:** Checked.
 - Log elapsed time, thread, and machine for each output line:** Unchecked.
- Execution:** A group box containing:
 - Minimize while running:** Unchecked.
 - Show detailed status window:** Checked.
 - Save status window position:** Unchecked.
- Debugging:** A group box containing:
 - Print agent calls:** Unchecked.
 - Print tags with agent calls:** Unchecked.

On the right side of the dialog, there are three buttons: 'OK', 'Cancel', and '<- Add...'.

第二章 准备工作

五、熟悉C/S应用



第二章 准备工作

五、熟悉C/S应用

- 在SilkTest中，对一个屏幕对象的标识同时采用了多种方式，包括：caption、prior text、index、window ID、location五个方面。默认同时采用caption和window ID。因此，当屏幕对象中上述五个方面的小部分发生改动，其余大部分仍然匹配时，SilkTest仍能够顺利识别屏幕对象，保证了较高的可维护性。

- 屏幕对象的动态标识(Dynamitc)

SilkTest不使用test frame时，对屏幕对象采用动态标识法，有两种方式：

- ◆ Syntax: Class(“tag”)

如：MainWin(“Customer Information”)

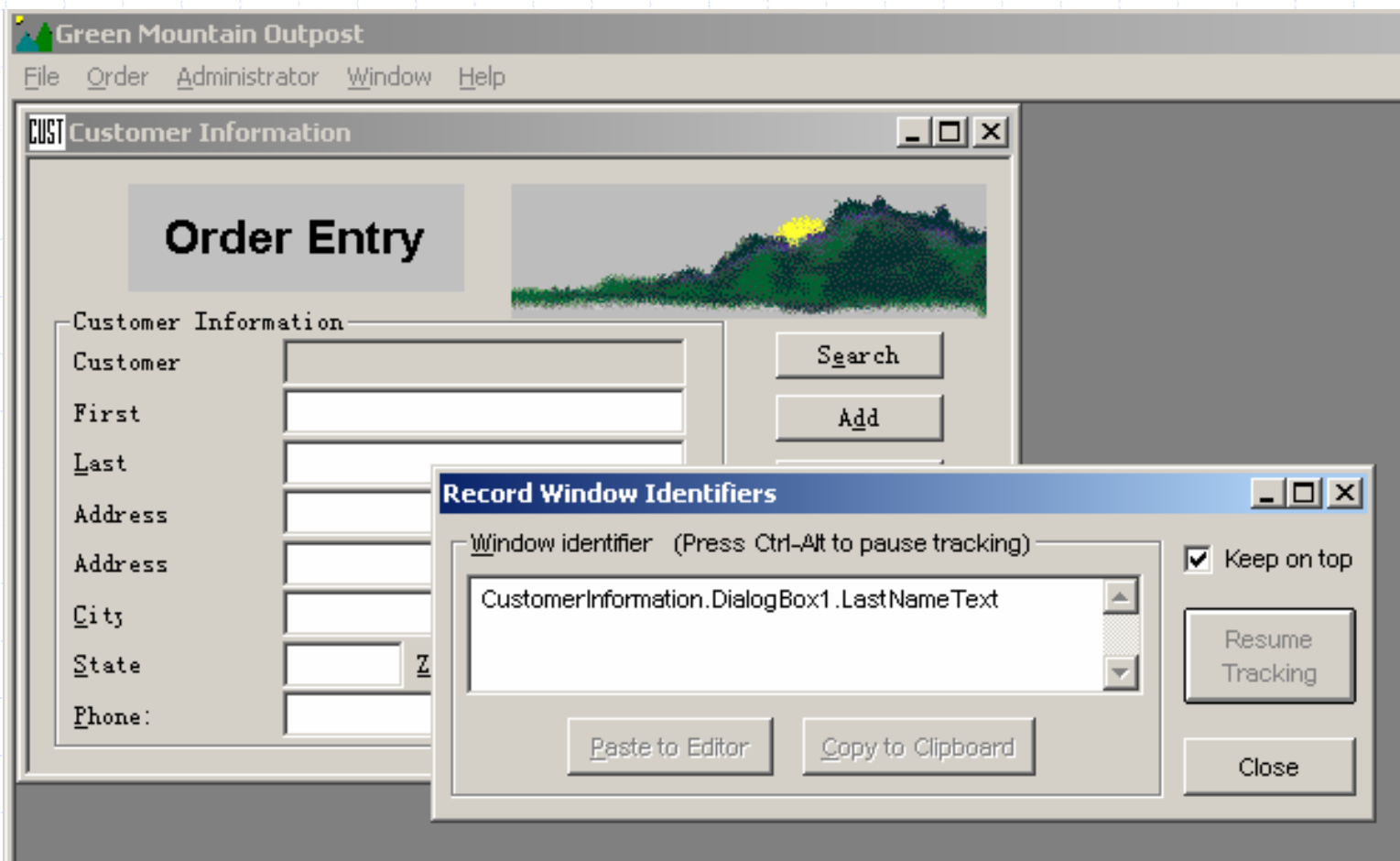
- ◆ Multi-tagging Syntax: Class(“caption|index|window ID”)

如：PushButtom(“Add|\$1010”)

第二章 准备工作

五、熟悉C/S应用

■ 使用test frame文件后的屏幕标识



第三章 测试计划

- 一、SilkTest Testplan 是什么?
- 二、了解Testplan Editor
- 三、创建Testplan
- 四、定义Attribute

第三章 测试计划

一、SilkTest Testplan 是什么？

- SilkTest testplan是一个测试计划大纲，是软件测试过程的框架，用于对软件测试需求进行组织与管理。
- 使用testplan的益处：
 - ◆ 采用自顶向下的方式定义测试需求，并细化。
 - ◆ 针对单独的或一组测试，可以指定属性(Attribute)；通过Attribute将各个测试分组，分组运行测试或生成报表。便于对各个测试进行组织与管理。
 - ◆ 采用树状结构描述测试需求，减少冗余。
 - ◆ Testplan是各种测试需求的详细目录
 - ◆ 通过做标记的方式，可以有选择地指定运行某些测试，生成报表。
 - ◆ 针对测试评估者而言，可以对测试的规划过程进行回顾，并评估测试计划的完整程度和逻辑的正确性

第三章 测试计划

一、SilkTest Testplan 是什么？

- SilkTest testplan包括两部分，outline和statements
 - ◆ Outline: 测试需求的格式化描述 (树状结构)
 - ◆ Statements: 将测试需求与测试脚本、测试用例相关联的语句，实现测试需求
- 通过testplan，我们可以得到如下信息：
 - ◆ 应用和测试方法的结构信息
 - ◆ 与测试相关的风险
 - ◆ 测试的相关人
 - ◆ 测试的先决条件
 - ◆ 软件的相关性
 - ◆ 软件的评估与修订历史
- 决定测试什么：测试计划中最重要步骤是确定应用的哪些特性需要测试，而哪些根本不需要测试。这些决定可以依赖于不同的信息来源，主要包括：产品需求文档和产品功能描述以及用户的反馈意见、产品维护人员的意见。

第三章 测试计划

一、SilkTest Testplan 是什么？

- 针对一个应用进行测试时，并非全部都需要自动化测试，需要根据以下判断是否需要自动化测试：
 - ◆ 需要不断重复进行
 - ◆ 相对比较容易进行自动化
 - ◆ 手工方式难以再现
 - ◆ 应用发布前，如果没有进行充分测试，风险性很大

第三章 测试计划

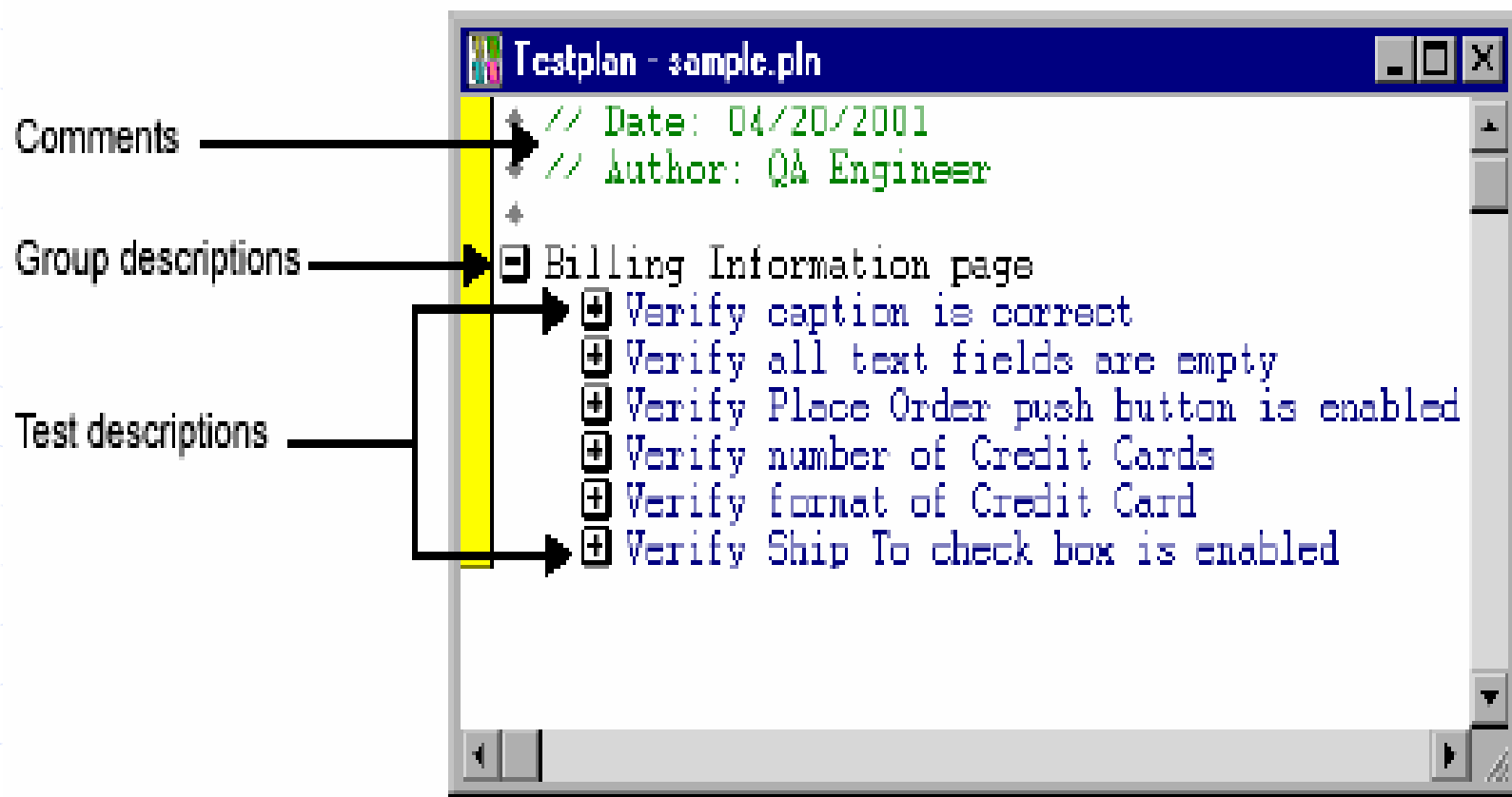
二、了解Testplan Editor

- SilkTest提供的编写测试计划的编辑器
- 采用缩进方式表示树状结构
- 不同颜色的字体表示不同含义

颜色	代表的元素	含义
Green	Comment	以//开头，注释
Black	Group description	成组测试的描述，树状结构的上层
Blue	Test description	单个测试的描述，树状结构的最下层
Dark Red	Testplan statements	链接语句，链接测试用例、数据，inc文件等
Magenta 紫红	Open subplan file marker	打开子plan

第三章 测试计划

二、了解Testplan Editor



第三章 测试计划

三、练习：创建Testplan P62-67



```
Testplan - d:\segue\gmo\exercise\gmoc_ch.pln (UTF-8)
♦ *****
♦ //GMO 测试计划举例
♦ //作者：吴凯
♦ //日期：2003/08/15
♦ //本测试计划包括两部分：针对一个窗口的GUI测试和功能测试
♦ //*****
♦
♦ + header:
♦
♦ - Customer Information窗口测试
♦   - GUI 测试
♦     ♦ 检查窗口标题(Caption)是否正确
♦     ♦ 检查输入焦点(Input focus)是否在 First Name域
♦     ♦ 检查所有文本框初始化为空
♦     ♦ 检查 Verify 按钮可用(enable)
♦     ♦ 检查 Add 按钮可用(enable)
♦     ♦ 检查 Update 按钮可用(enable)
♦
♦   ♦ //.....
♦
♦   ♦ //以下测试完功能测试
♦   ♦ //第一个测试用例为PriceCheck，用于检查应用中，当完成定单操作后，价格总数的
♦   ♦ //计算是否正确。这个测试用例中，订购了三个背包，然后校验总数是否为$591.13
♦   ♦
```


第三章 测试计划

四、定义Attribute

- Attribute是指测试计划中用到的针对某个点或项目的属性，并可关联到某一个test description或group description。
- 通过使用attribute，可以将测试分类、分组管理。
 - ◆ 通过分组，可以成组运行测试。
 - ◆ 成组生成报表
- Default attributes

Attribute	含义
Category	测试用例的类型
Component	被测模块的名称
Developer	测试用例的开发人

第三章 测试计划

四、定义Attribute

- Default attributes存储在ini文件中，默认为testplan.ini
- 可以自定义attributes及其value
- 练习 P69-72
 - ◆ Define values for default attributes



第三章 测试计划

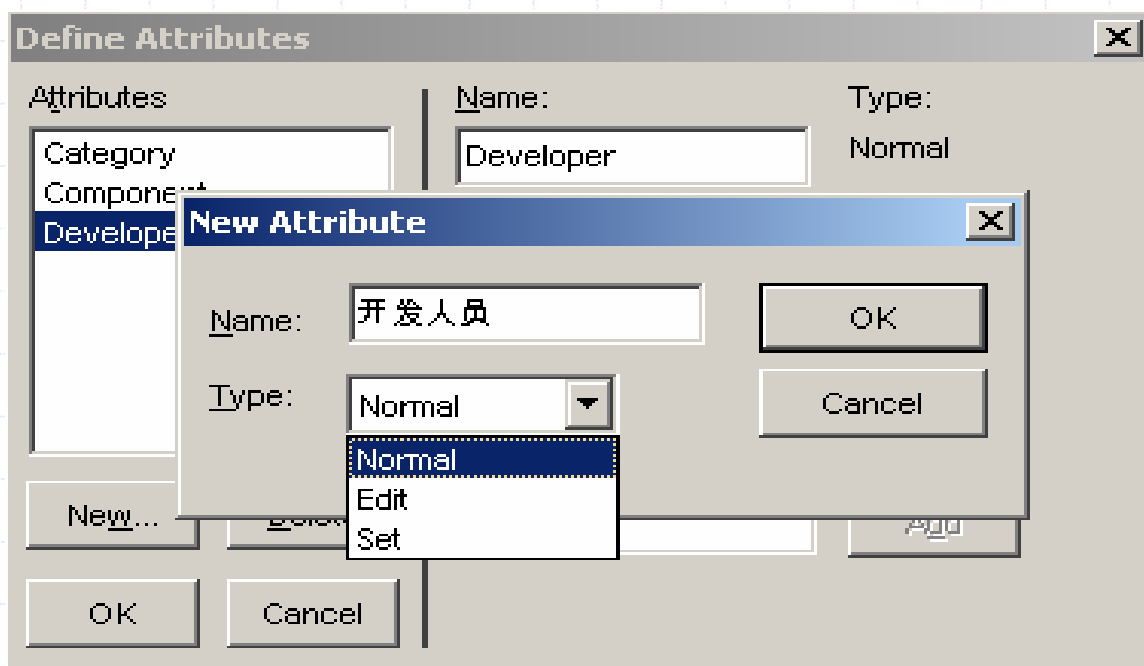
四、定义Attribute

■ 练习 P69-72

- ◆ Define new attributes and their values

■ Attribute type

Type	说明
Normal	针对该attribute,每一个testplan语句只有一个attribute值,attribute值需要预先定义
Edit	针对该attribute,每一个testplan语句只有一个attribute值,但可临时编辑, attribute值不能预先定义
Set	每一个testplan语句可有多个attribute值

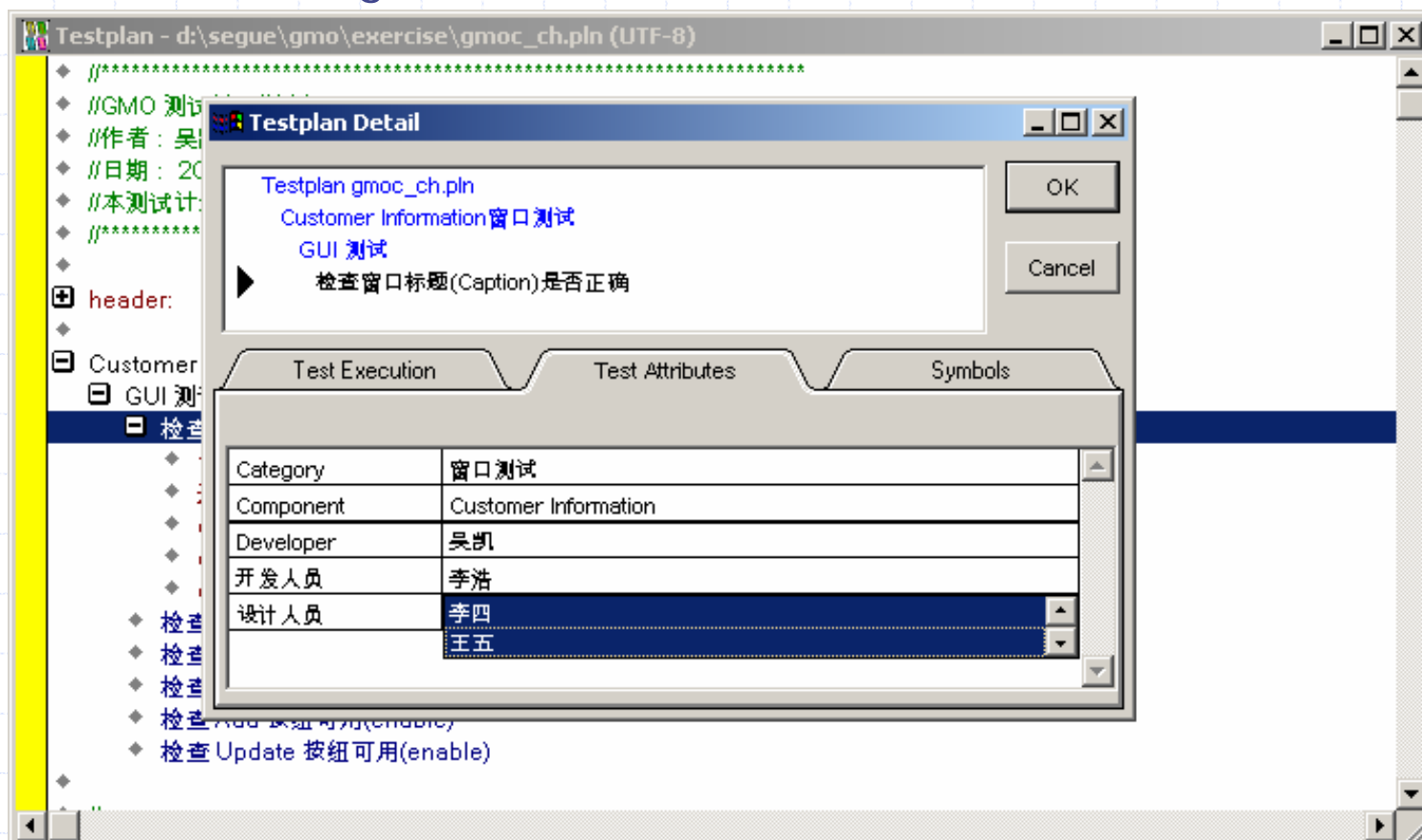


第三章 测试计划

四、定义Attribute

■ 练习 P69-72

◆ Associating Attributes: test detail



第四章 应用信息捕获(Capturing Your Application)

- 一、Test Frame的概念
- 二、创建Test Frame
- 三、The Window Declaration
- 四、先于Main Window的window
- 五、测试 DefaultBaseState
- 六、Tags and Identifiers

第四章 应用信息捕获(Capturing Your Application)

一、Test Frame的概念

- 在开始录制测试用例之前，SilkTest需要知道一些AUT的信息。通过capture，可以获得这些信息，存放在Test Frame中。
- Test Frame是一个include(.inc)文件，相当于一个repository，存放AUT的主要信息：
 - ◆ Window declarations
 - ◆ Constants and variables declarations
- Test Frame中的信息用于支持SilkTest进行录制和运行测试用例，包括两部分信息：应用主窗口（Main Window）的信息和每一个窗口的信息，因此，Test frame的建立包括两部分。
- Test Frame的建立包括两阶段
 - ◆ 第一阶段：记录应用主窗口(main window)的window declaration，由File→New→Test Frame完成（main window 即为Default Base State）
 - ◆ 第二阶段：记录应用中每个window的window declaration，由Record→ Window Declaration完成

第四章 应用信息捕获(Capturing Your Application)

一、Test Frame的概念

■ Web应用的Test Frame的信息

wMainWindow

comments

window BrowserChild

sLocation

sUserName

dPassword

constant BrowserSize

constant bDefaultFont

Object declarations

```
4Test Include - gmoow.inc
+
+ const wMainWindow = GMO
+
+ // wMainWindow points to the main window of a Web application.
+ // Note that its identifier(GMO) is the same as the home page
+ // of the GMO application, so the recovery system will restore
+ // the application to this page if necessary for testcases
+ // using DefaultBaseState.
+
+ // Home page
+ window BrowserChild GMO
+ * tag "Welcome to Green Mountain Outpost"
+
+ // The URL of this page
+ * const sLocation = "http://gmo.segue.com"
+
+ // The login user name
+ * const sUserName = ?
+
+ // The login password
+ * const sPassword = ?
+
+ // The size of the browser window
+ * const POINT BrowserSize = {600, 400}
+
+ // Sets the browser font settings to the default
+ * const bDefaultFont = TRUE
+
+
+ @ HtmlMeta ContentType
+ @ HtmlMeta GENERATOR
+ @ HtmlImage Logo
+ @ HtmlHeading GMOOnline
+ @ HtmlText WelcomeText
+ @ HtmlText WeAreMovingOur325Page200
+ @ HtmlText BeSureToVisitUsOnTheInt1
+ @ HtmlTable BorderlessTable1
+ @ HtmlTable BorderlessTable2
+ @ HtmlText ThisIsASampleWebSiteFor
+ @ HtmlText InformationInThisDocumentI
+ @ HtmlText Copyright©2001SegueSoftwar
+ @ HtmlLink PleaseLetUsKnowHowYouLik
```

第四章 应用信息捕获(Capturing Your Application)

一、Test Frame的概念

■ Web应用的Test Frame的信息

A constant named wMainWindow	This constant holds the identifier value of your application's home page, that is, the page that was displayed in the browser when you created the test frame. The recovery system uses wMainWindow to restore the browser to that page when a test fails. This is known as Base State. Typically the Base State navigates to the first page of the application.
A window of class BrowserChild	This window has the same identifier as the value of wMainWindow; this is the window that SilkTest loads in order to restore the application to Base State.
sLocation	The URL for the homepage; used by the recovery system to load the page.
sUserName and dPassword	Two commented constants that specify the user name and password to access the application if required.
BrowserSize and bDefaultFont	Two commented constants that specify the size of the browser window and the default font to use for displaying text.
Objects Declarations	Declarations of all objects on the page, such as HtmlHeadings, HtmlText, HtmlLinks, HtmlText, HtmlPushButtons and so on.

第四章 应用信息捕获(Capturing Your Application)

一、Test Frame的概念

■ C/S应用Test Frame的信息

wMainWindow

comments

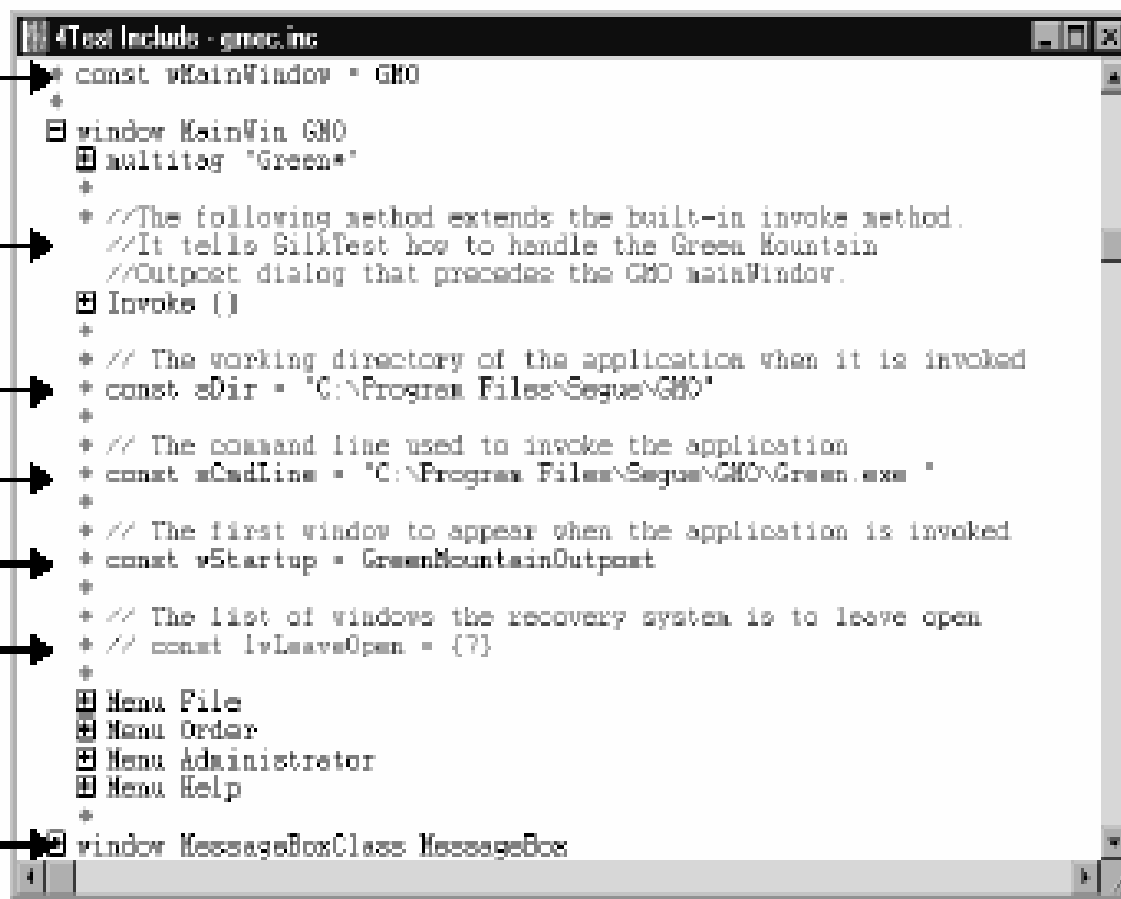
sDir

sCmdLine

wStartup

lvLeaveOpen

MessageBoxClass



```
4Test Include - green.inc
* const wMainWindow = GHO
*
* window MainWin GHO
*   multitag "Green*"
*
* //The following method extends the built-in invoke method.
* //It tells SilkTest how to handle the Green Mountain
* //Outpost dialog that precedes the GHO mainWindow.
* Invoke ()
*
* // The working directory of the application when it is invoked
* const sDir = "C:\Program Files\Segue\GHO"
*
* // The command line used to invoke the application
* const sCmdLine = "C:\Program Files\Segue\GHO\Green.exe "
*
* // The first window to appear when the application is invoked
* const wStartup = GreenMountainOutpost
*
* // The list of windows the recovery system is to leave open
* // const lvLeaveOpen = {}
*
* Menu File
* Menu Order
* Menu Administrator
* Menu Help
*
* window MessageBoxClass MessageBox
```


第四章 应用信息捕获(Capturing Your Application)

一、Test Frame的概念

■ C/S应用Test Frame的信息

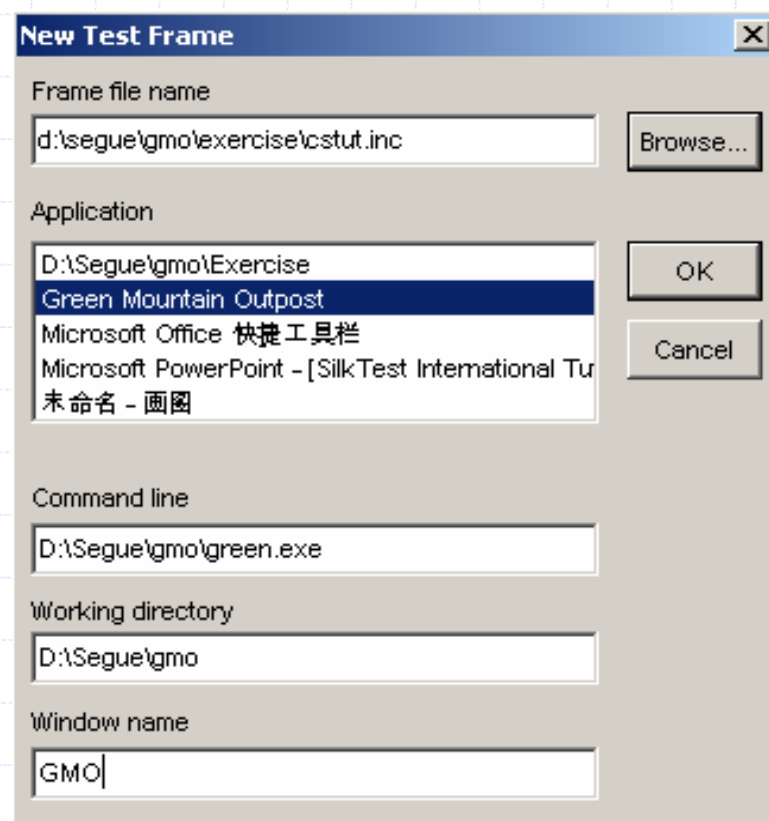
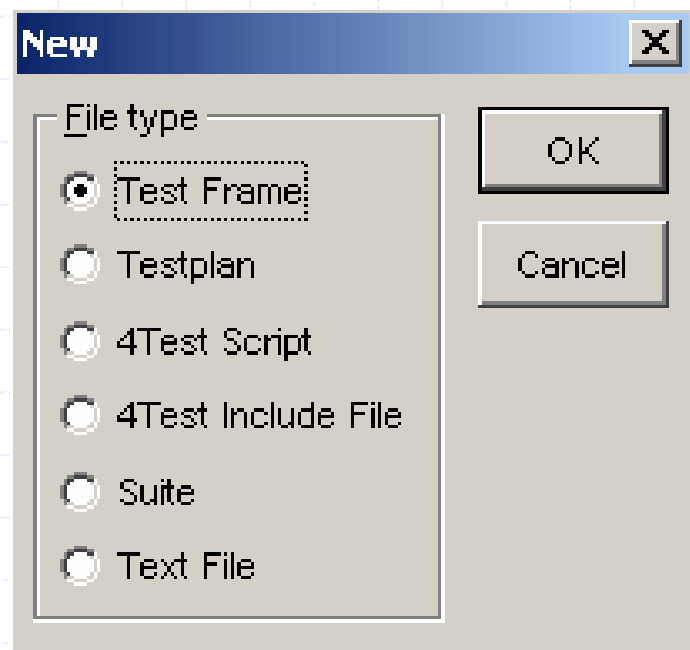
wMainWindow	创建Test Frame时应用的主窗口 main window的标识 Recovery System恢复应用 Base State
sDir	AUT的working directory
sCmdLine	启动AUT的命令行
wStartup	应用启动时的第一个窗口的标识，由Recovery System控制，默认comment掉
lwLeaveOpen	包含一个或多个window的identifier，表明测试用例完成后应保留的窗口名称，默认comment掉
MessageBoxCla	常用对话框的声明

SS

第四章 应用信息捕获(Capturing Your Application)

二、练习：创建Test Frame P81

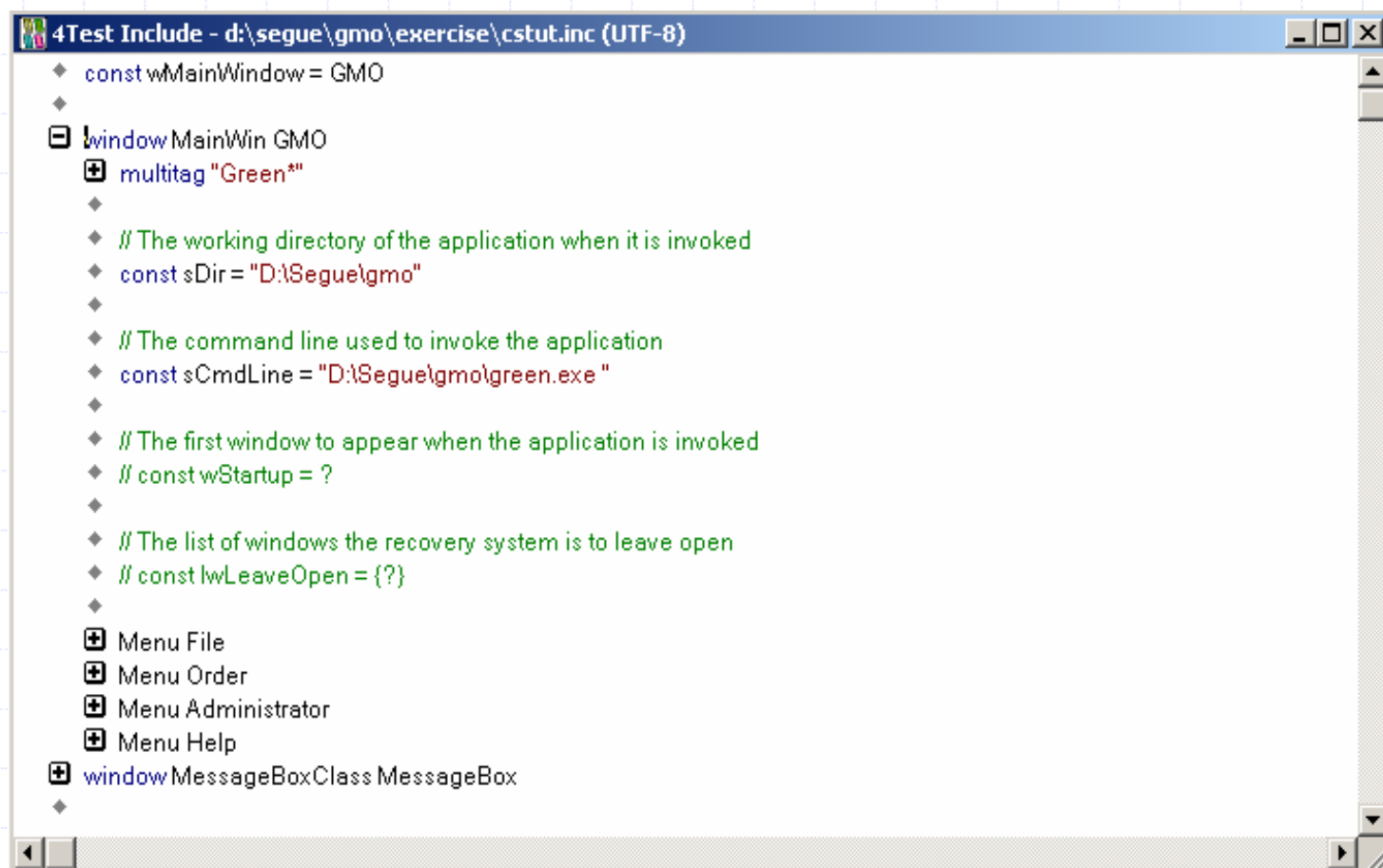
- 第一阶段：创建main window的declaration
 - ◆ File → New → Test Frame



第四章 应用信息捕获(Capturing Your Application)

二、练习：创建Test Frame P81

- 第一阶段：创建main window的declaration
 - ◆ 生成的test frame inc文件自动加入到Use Files中



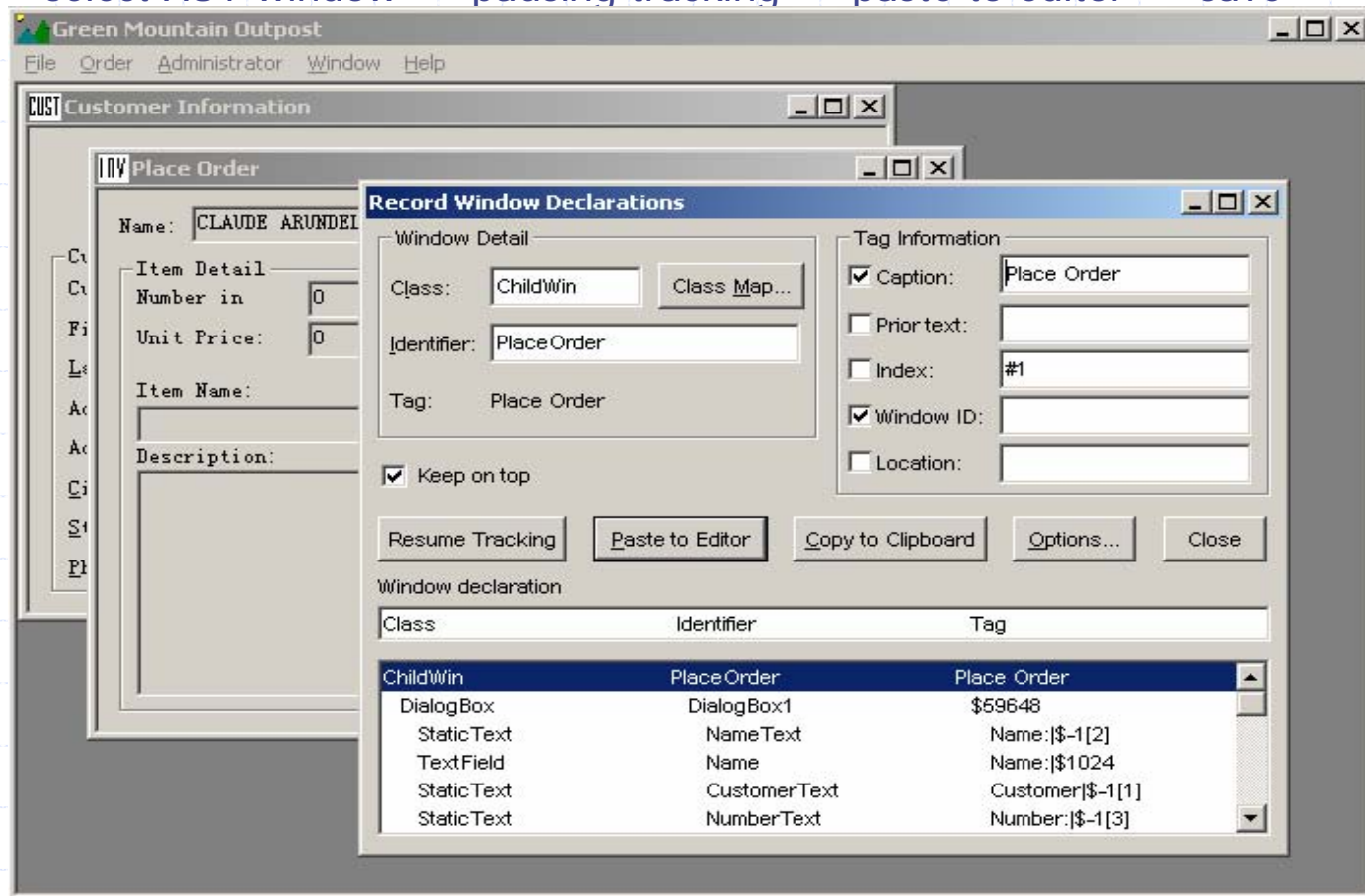
```
4Test Include - d:\segue\gmo\exercise\cstut.inc (UTF-8)
◆ const wMainWindow = GMO
◆
window MainWin GMO
  + multitag "Green*"
  ◆
  ◆ // The working directory of the application when it is invoked
  ◆ const sDir = "D:\Segue\gmo"
  ◆
  ◆ // The command line used to invoke the application
  ◆ const sCmdLine = "D:\Segue\gmo\green.exe "
  ◆
  ◆ // The first window to appear when the application is invoked
  ◆ // const wStartup = ?
  ◆
  ◆ // The list of windows the recovery system is to leave open
  ◆ // const lwLeaveOpen = {}
  ◆
  + Menu File
  + Menu Order
  + Menu Administrator
  + Menu Help
  + window MessageBoxClass MessageBox
  ◆
```

第四章 应用信息捕获(Capturing Your Application)

三、练习：The Window Declaration P82-83

■ 创建应用中其他window的declaration

- ◆ In SilkTest open test frame -> open AUT -> open Record / Window Declaration -> select AUT window -> pausing tracking -> paste to editor -> save

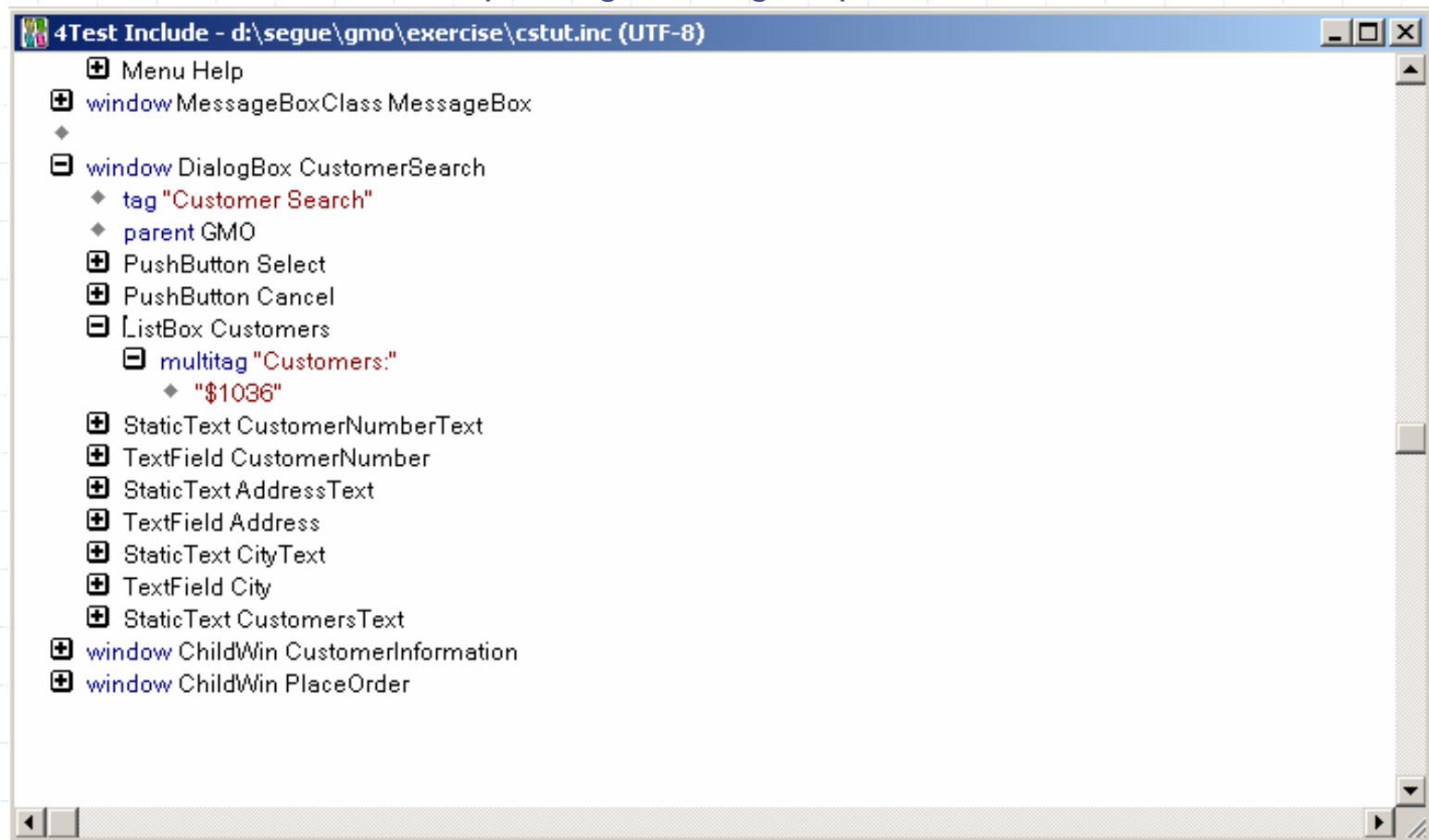


第四章 应用信息捕获(Capturing Your Application)

三、练习：The Window Declaration P82-83

■ 创建应用中其他window的declaration

- ◆ In SilkTest open test frame -> open AUT -> open Record / Window Declaration -> select AUT window -> pausing tracking -> paste to editor -> save



第四章 应用信息捕获(Capturing Your Application)

四、先于Main Window的window

- 很多应用并不是一开始就进入主窗口，往往先经过一个窗口再进入，如：**login**或某些提示的对话框等。这样，就可能导致自动测试时无法达到所要求的DefaultBaseState，也就无法实现测试用例的测试。
- 可以引入Invoke method解决。
 - ◆ 录制第一个窗口，并加入到testframe中。
 - ◆ 将wStartup constant填写完整，指向该第一个窗口。
 - ◆ 在main window的declaration中，定义Invoke method，完成所有到达第一个窗口的行为。

第四章 应用信息捕获(Capturing Your Application)

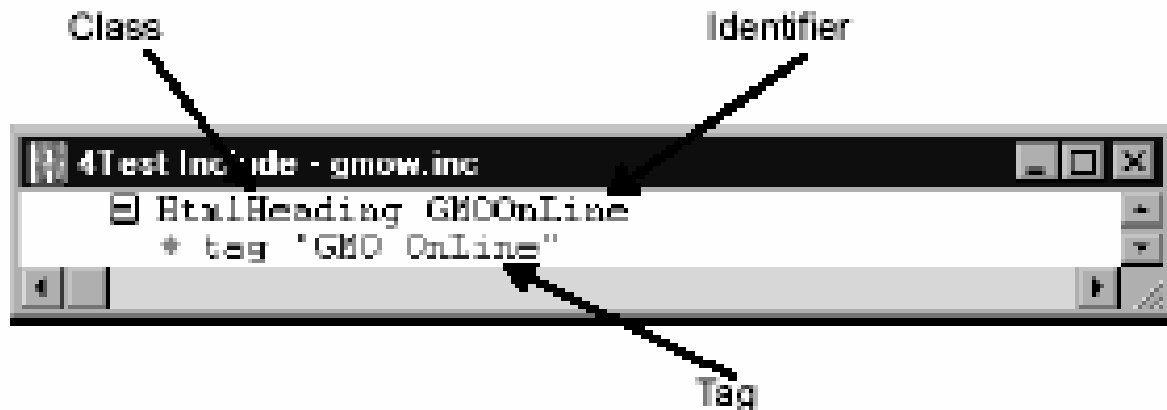
五、测试 Application State P88-89

- ◆ 事实上，每个测试用例都可以从不同的应用状态开始。这些状态称为 Base State，把应用的主窗口称为 Default Base State。
- ◆ Default Base State 可以通过 new frame 方式完成，其他 Base State 则通过 record → Application state 完成。
- ◆ 可以通过 Run → Application State 去检测 Default Base State 和各个 Base State 是否正确。

第四章 应用信息捕获(Capturing Your Application)

六、Tag 与 identifier

- Capture过程中，每个对象都有一个class、tag和identifier
- Class和tag由SilkTest产生、识别、内部使用，不允许修改；Identifier可以修改，用于脚本中对屏幕对象的标识。
- 事实上，SilkTest往往采用多个来Tag标识屏幕对象，称为multi-tag，包括caption,prior text,index,Window ID,Location等。这样，当一个屏幕对象的若干少部分因素发生变化时，仍有可能识别该对象，降低脚本的维护量。
- Identifier可以修改，目的：增加脚本的可读性 P90
- 修改时，可以在record window declaration时或直接修改frame文件



第五章 创建测试用例(Creating Testcases)

- 一、创建测试用例指南
- 二、录制一个测试用例
- 三、将脚本、测试用例与测试计划连接
- 四、从测试计划中录制
- 五、测试计划与测试脚本、测试用例的连接方式
- 六、检查4Test 语句的结构
- 七、了解Recovery System如何工作

第五章 创建测试用例(Creating Testcases)

一、创建测试用例指南

当创建测试用例时，应明确

- 一个测试用例完成的三项基本任务
 - ◆ 驱动被测应用进行测试
 - ◆ 确认应用实际运行后的状态与期待状态是否吻合
 - ◆ 将应用返回到初始状态
- 每一个测试用例都是相互独立的，不能有某一个脚本/测试用例依赖于前一个脚本/测试用例的成功完成。
- 测试用例的目的应单一，便于应用问题的诊断
- 每个测试用例开始于一个基本状态(Base State)。

第五章 创建测试用例(Creating Testcases)

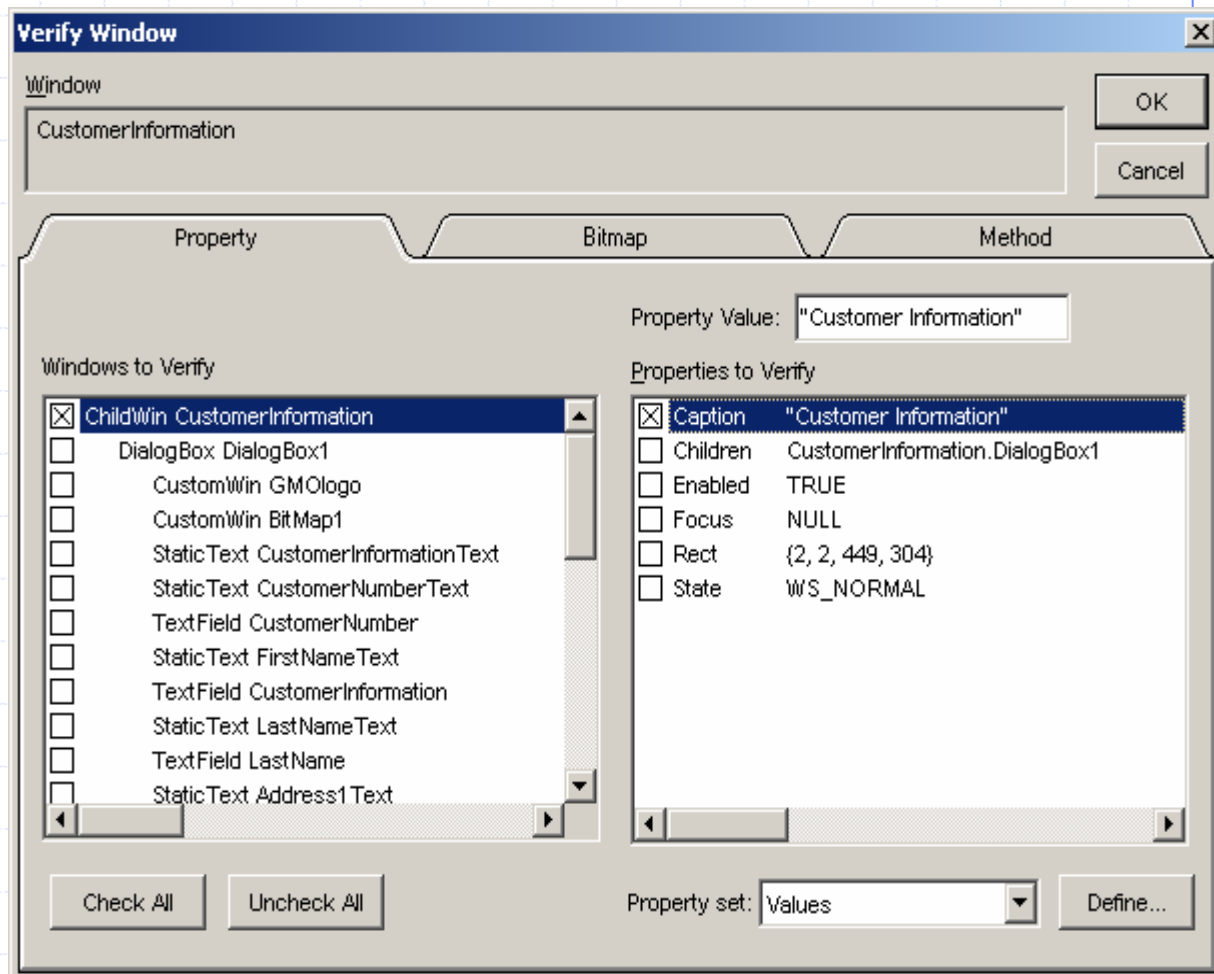
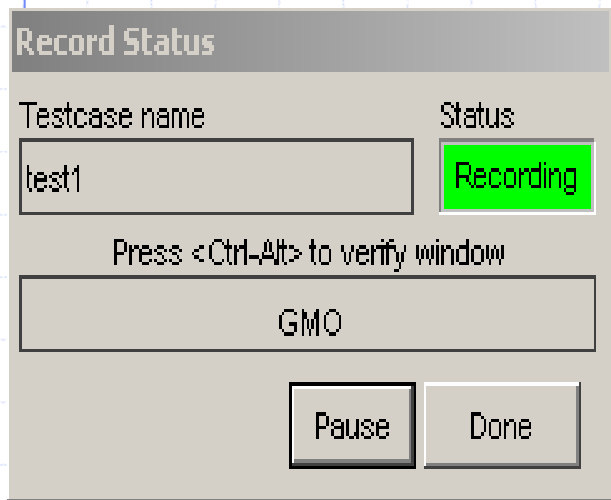
二、录制一个测试用例P94-100

测试用例的创建通过录制的方式辅助生成测试脚本，包括三个阶段：

- 录制前的设置
 - ◆ 确保使用正确的frame或include文件，frame文件在Use Files中。
- 录制从应用开始到选定的Base State之间的操作
 - ◆ 录制驱动应用从主窗口到被测窗口的过程
 - ◆ 该过程也可通过Record→Application State生成的各个Base State完成
- 录制有关校验的部分
 - ◆ Ctrl+Alt 打开Verify Window
 - ◆ 选择需要校验的部分
 - ◆ 粘贴到脚本文件中，生成脚本
- 录制过程中注意：鼠标移动时慢一些，确保recorder能够捕获到。

第五章 创建测试用例(Creating Testcases)

二、录制一个测试用例P94-100



第五章 创建测试用例(Creating Testcases)

三、将脚本、测试用例与测试计划连接

- 在测试计划中加入Statements，可以将测试计划与测试脚本、测试用例关联起来。P101

四、从测试计划中录制 P102-107

第五章 创建测试用例(Creating Testcases)

三、将脚本、测试用例与测试计划连接

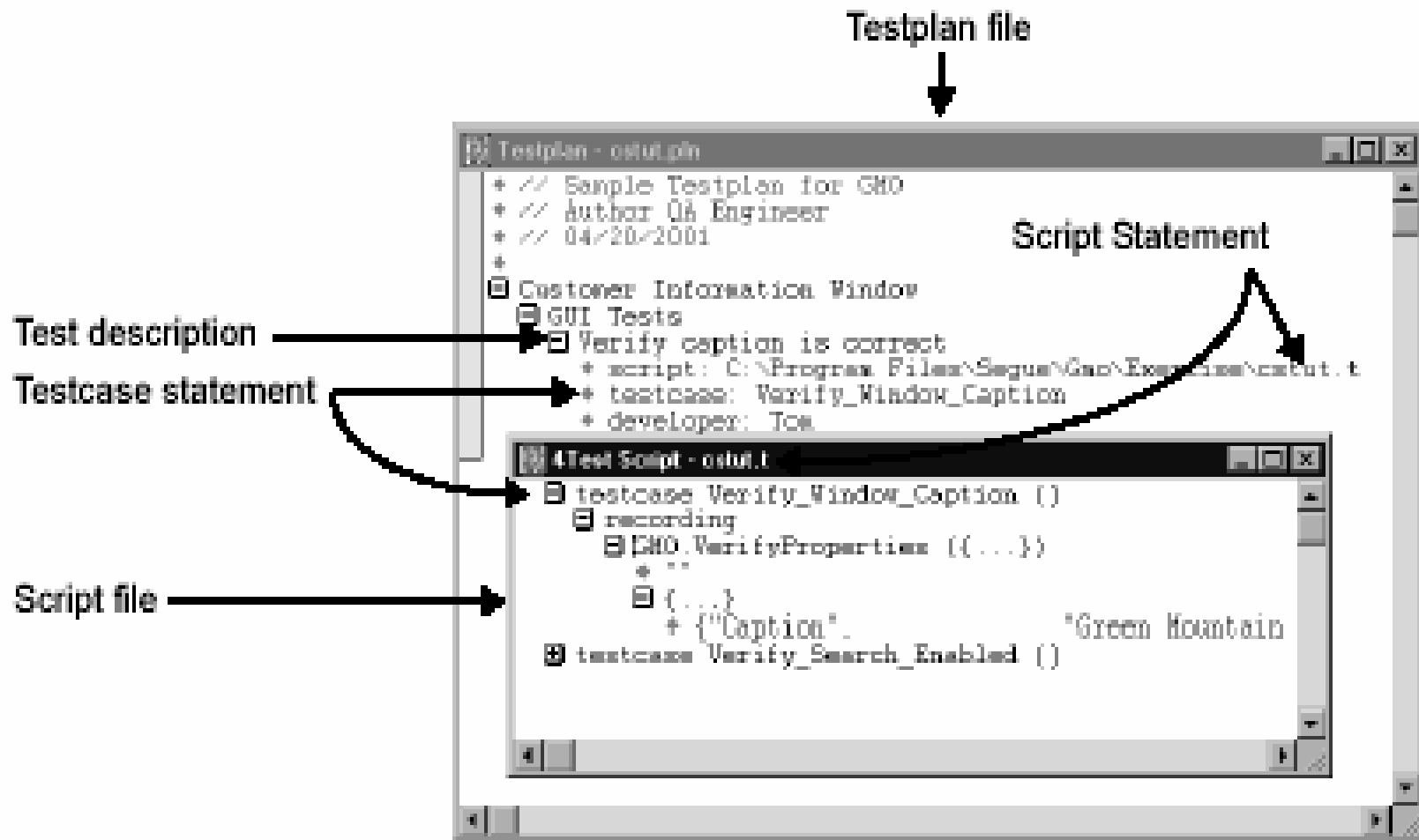
- 在测试计划中加入Statements，可以将测试计划与测试脚本、测试用例关联起来。 P101
 - ◆ 打开测试计划文件
 - ◆ 选择某一个description
 - ◆ 直接写入script 和 testcase

四、从测试计划中录制 P102-107

- ◆ 打开测试计划文件
- ◆ 选择某一个description
- ◆ Record→testcase
- ◆ SilkTest自动插入 script 和 testcase

第五章 创建测试用例(Creating Testcases)

五、测试计划与测试脚本、测试用例的连接方式



第五章 创建测试用例(Creating Testcases)

六、检查4Test 语句的结构

This Line of 4Test script language	Tells SilkTest
<code>testcase Verify_Window_Caption ()</code>	That the testcase is named <code>Verify_Window_Caption</code>
<code>GMO.SetActive ()</code>	To make the GMO application active
<code>GMO.Order.CustomerInformation.Pick ()</code>	To select the Customer Information menu item under the Order menu in the main window
<code>CustomerInformation.VerifyProperties ({...})</code>	To verify selected properties of the Customer Information window
<code>" " {...} {"Caption", "Customer Information"}</code>	To verify the caption in the Customer Information window
<code>CustomerInformation.SetActive ()</code>	To make the Customer Information window active
<code>CustomerInformation.DialogBox1.Close.Click ()</code>	To close the Customer Information window

第五章 创建测试用例(Creating Testcases)

六、了解Recovery System如何工作

- Recovery System是SilkTest独有的特性，保证每一个测试用例的开始和结束都在所需要的状态。这个状态称为Base State。
- Base State是应用的一个稳定的状态，是测试用例开始执行前所处的状态，同时也是测试用例结束后必须返回到的状态。
 - ◆ Base State可以保证测试的完整性，保证一个测试用例的失败不会影响到下一个测试用例运行。
 - ◆ 不同的testcase可以有不同的base state.
 - ◆ 各个base state可以通过record→Application State完成，也可以通过录制中从应用开始到base state的全部动作实现。

第五章 创建测试用例(Creating Testcases)

六、了解Recovery System如何工作

- Recovery System在三个阶段保证应用处于Base State
 - ◆ 测试用例运行之前
 - ◆ 运行过程中
出现错误时，
 - 停止测试用例的执行
 - 将控制传递给Recovery System，由Recovery System将应用恢复到Base State，同时详细记录错误情况
 - 运行下一个测试用例
 - ◆ 成功运行之后
- SilkTest为每个应用提供了一个Default Base State，可以确保录制或运行时：
 - ◆ 应用正在运行
 - ◆ 应用不是最小化
 - ◆ 应用是当前激活应用
 - ◆ 只有应用的主窗口打开

第六章 运行测试用例(Running Tstcases)

- 一、从测试计划中运行测试用例的选择
- 二、从测试计划中单独运行一个测试用例
- 三、运行测试计划中全部测试用例
- 四、运行测试计划中标记的测试用例

P113-119

第七章 测试结果

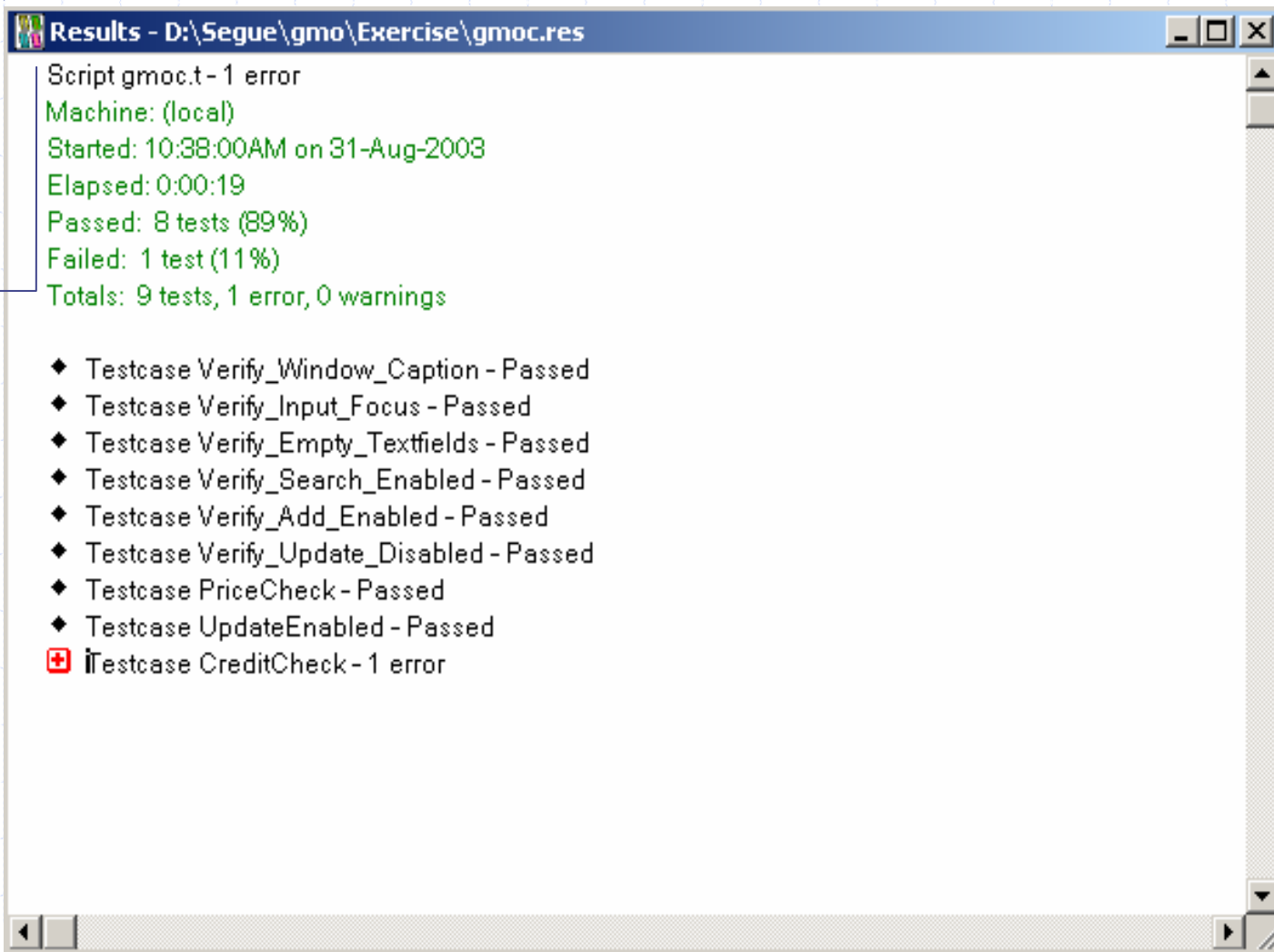
- 一、测试结果文件
- 二、测试用例中的错误
- 三、使用Difference Viewer
- 四、产生错误报告

第七章 测试结果

一、测试结果文件

■ .res文件

Result summary



```
Results - D:\Segue\gmo\Exercise\gmoc.res
Script gmoc.t - 1 error
Machine: (local)
Started: 10:38:00AM on 31-Aug-2003
Elapsed: 0:00:19
Passed: 8 tests (89%)
Failed: 1 test (11%)
Totals: 9 tests, 1 error, 0 warnings

◆ Testcase Verify_Window_Caption - Passed
◆ Testcase Verify_Input_Focus - Passed
◆ Testcase Verify_Empty_Textfields - Passed
◆ Testcase Verify_Search_Enabled - Passed
◆ Testcase Verify_Add_Enabled - Passed
◆ Testcase Verify_Update_Disabled - Passed
◆ Testcase PriceCheck - Passed
◆ Testcase UpdateEnabled - Passed
❌ Testcase CreditCheck - 1 error
```

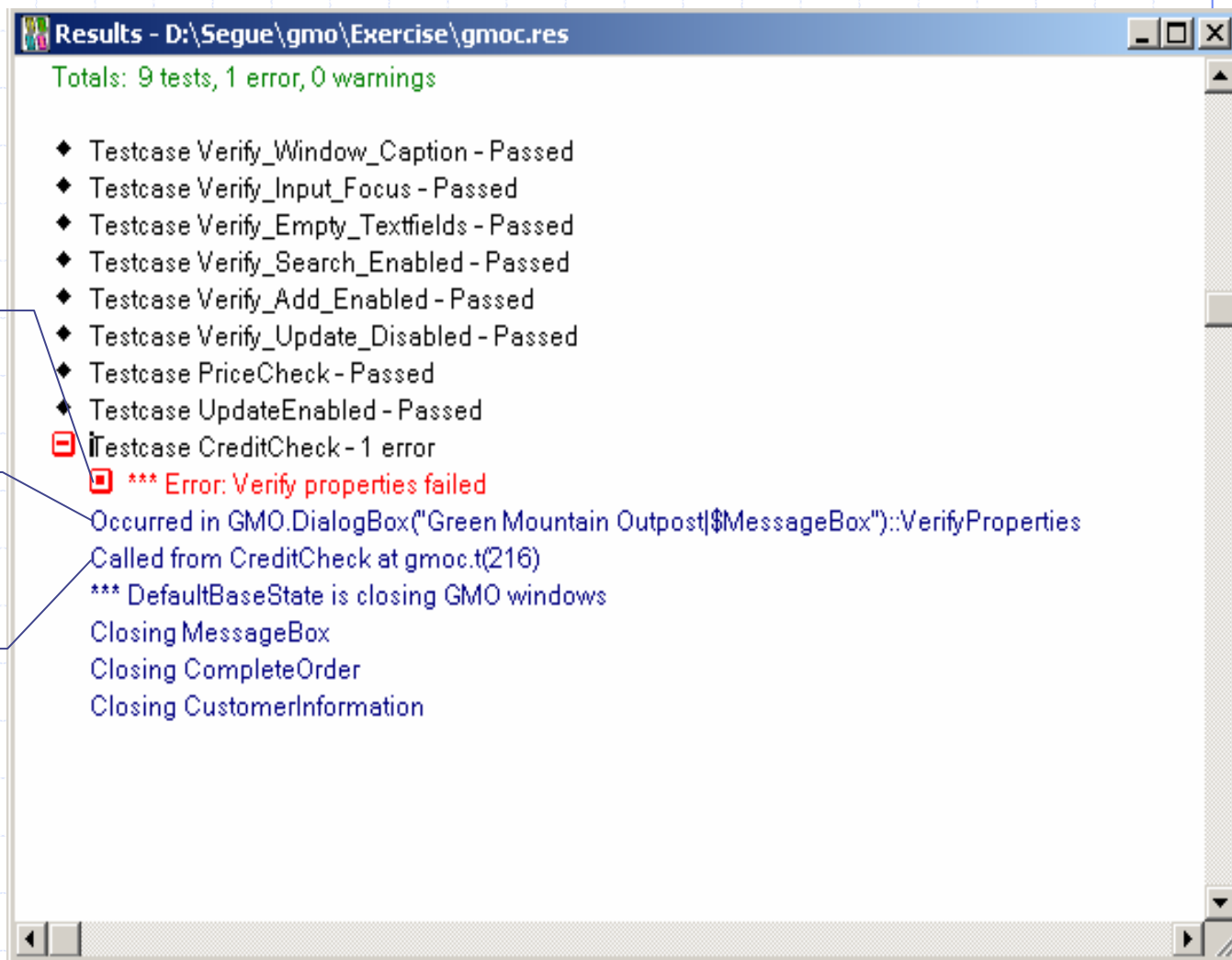
第七章 测试结果

二、测试用例中的错误

What error in AUT

Detect By...

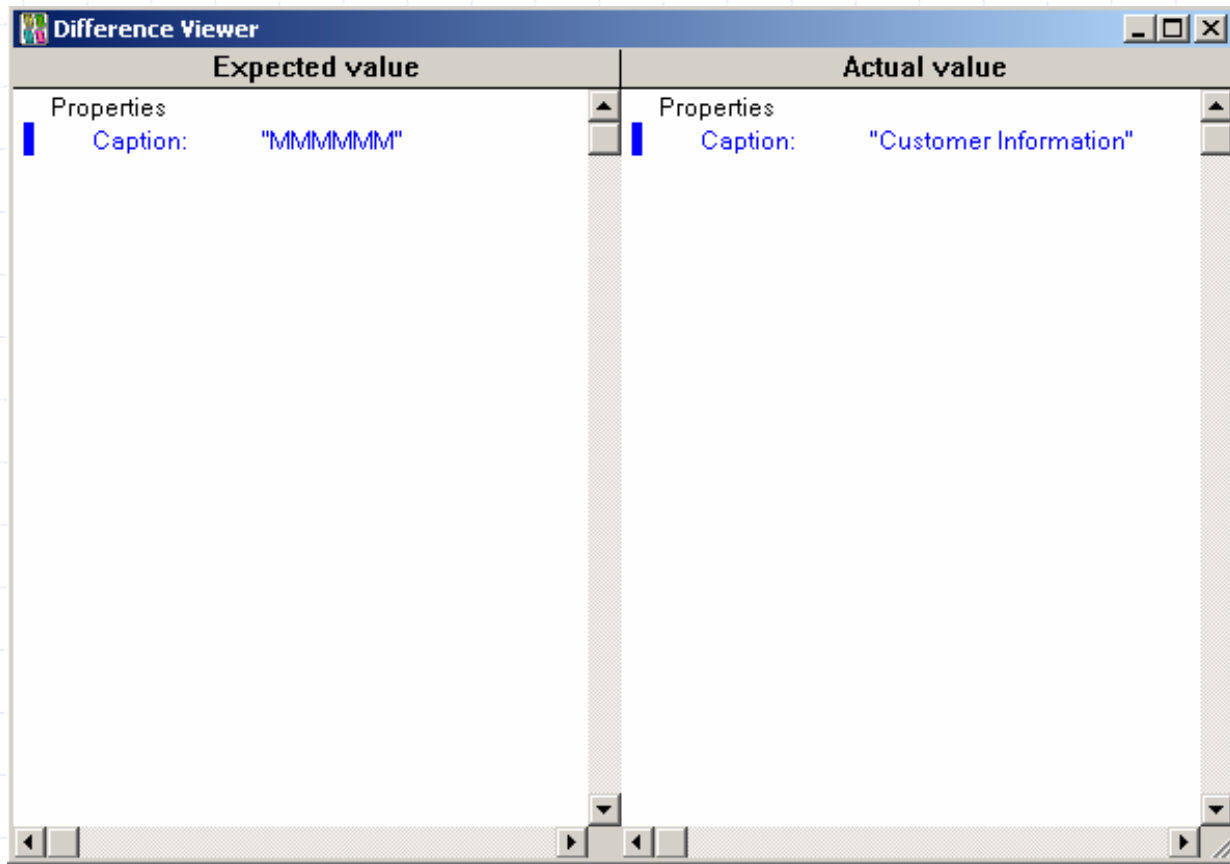
Where in script



第七章 测试结果

三、使用Difference Viewer

- SilkTest使用Difference Viewer，将测试出问题的脚本中的期望值与测试到的实际值并列显示，便于确定错误。
- 在Difference Viwer中，同样可以修改脚本中的expect value



第七章 测试结果

三、产生错误报告

- SilkTest可以根据Attribute产生各种report

