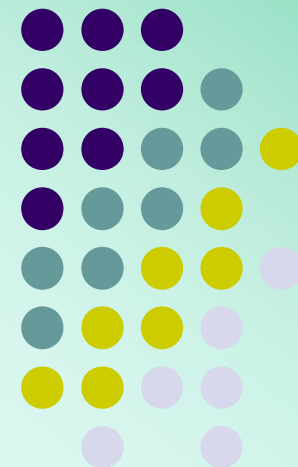


# 自动化功能测试技术

---



# 目 录



自动化功能测试框架



QuickTest Pro主要功能



QuickTest Pro测试流程



QuickTest Pro上机实践



# 1: 自动化功能测试背景

- 软件测试的目的是为了发现尽可能多的缺陷，不是为了说明软件中没有缺陷
- 由于软件系统的功能增加越来越快，测试难度越来越大，手工测试已经跟不上这种发展趋势。自动化功能测试方法应运而生
- 自动化功能测试特别适用于回归测试，有效排除人为因素影响，保证测试标准一致，测试结果可重复，并且提高测试效率



# 1: 自动化功能测试意义

- 自动化测试可以将测试人员从重复枯燥的手工操作中解放出来，让测试人员有更多的精力关注在对被测系统的分析上。测试人员将更加重视测试用例的设计
- 自动化测试在现阶段不能取代手工测试
- 凡是重复枯燥的工作，都可以也应该被自动化测试取代
- 回归测试是使用自动化测试的一个比较适宜的阶段



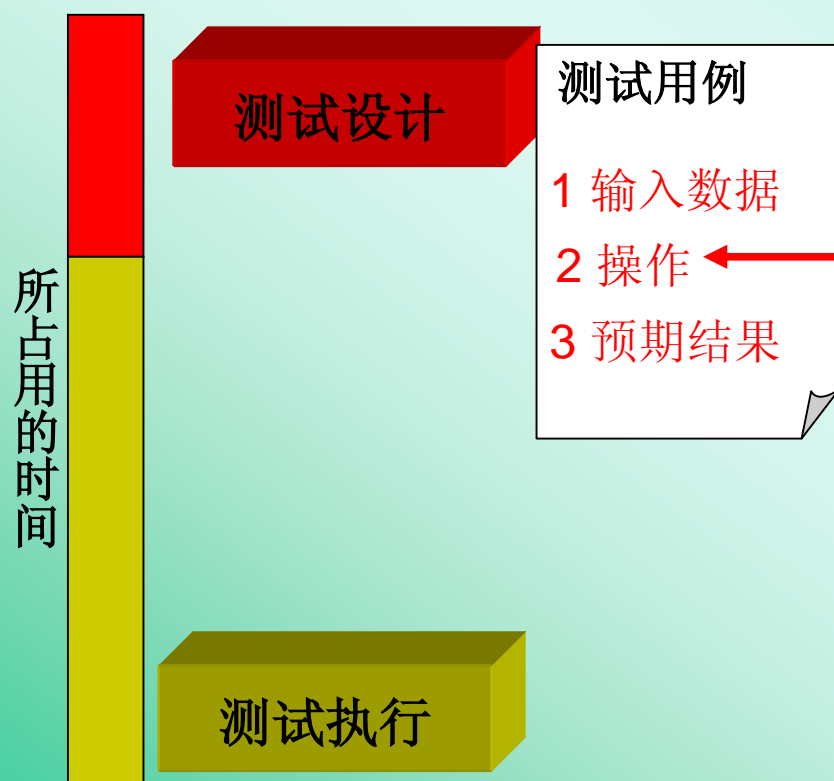
# 1: 自动化功能测试前提

- | 自动化测试需要在覆盖率和实现成本上找到平衡点。覆盖率越高，实现和维护成本越大。覆盖率低，自动化测试没有实现的价值。针对产品，应该在自动化测试中实现实施测试的关键流程。
- | 自动化测试需要测试人员有一定的软件设计和开发能力，同时也要对被测系统的业务有一定的认识。
- | 自动化测试需要有完善的测试流程和测试策略做保证。
- | 自动化测试的脚本的维护成本比较高。

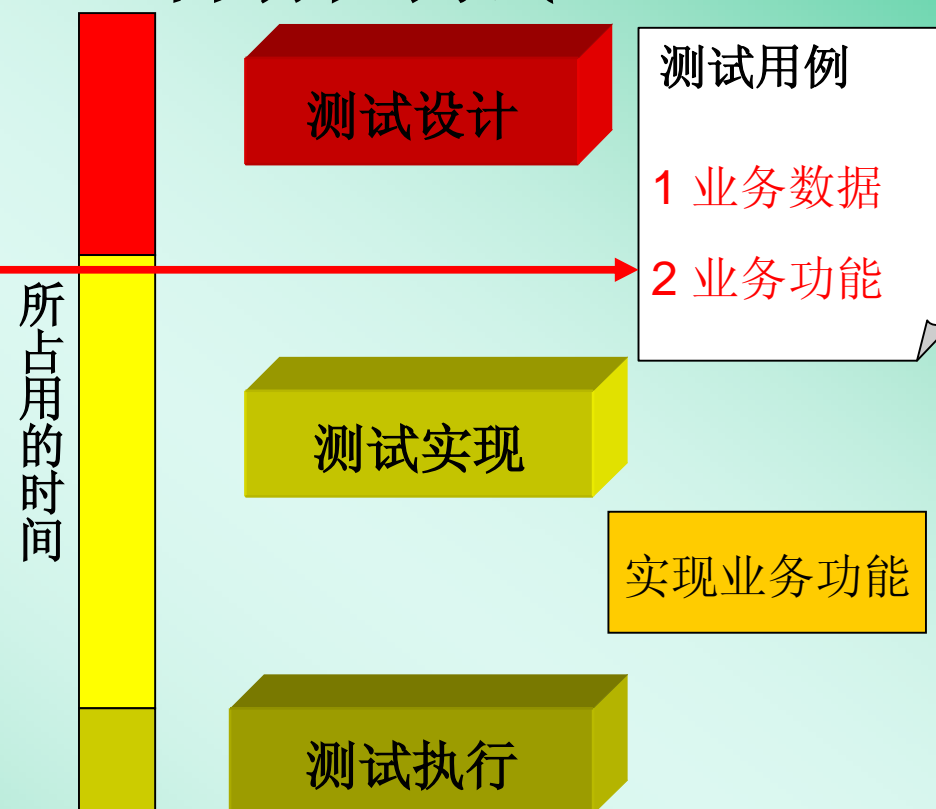


# 1: 自动化功能测试优势

## I 手工测试



## I 自动化测试





# 1: 自动化功能测试优势

## I 手工测试

- ⑧ Time-consuming and tedious(费时并单调)
- ⑧ Impossible to test each feature thoroughly (不能彻底测试)

## I 自动化测试

- I Fast (快速)
- I Reliable (可靠)
- I Repeatable (可重复)
- I Programmable(可程序化)
- I Comprehensive (全面)
- I Reusable(可重用)

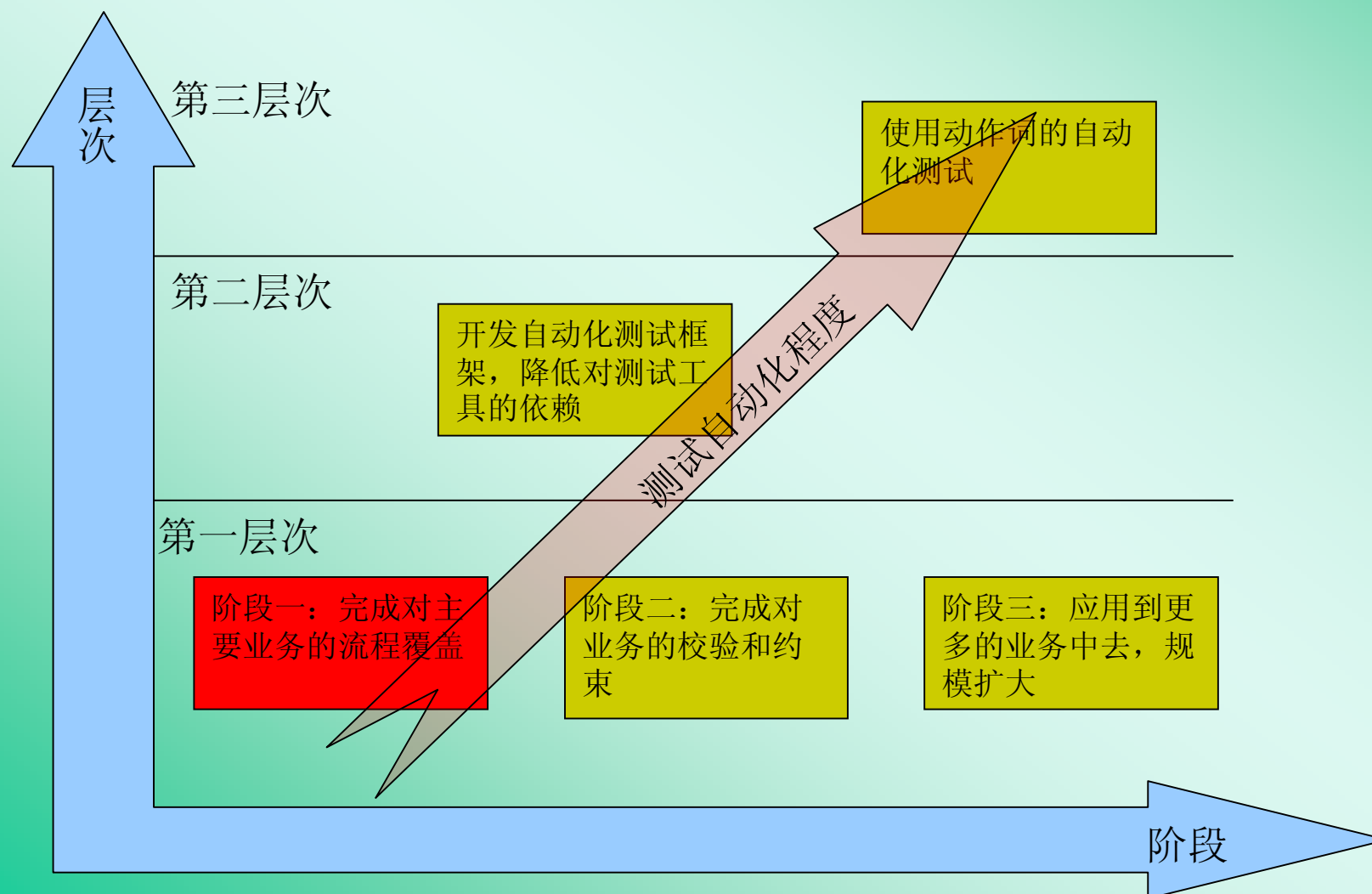


# 1:自动化功能测试原理

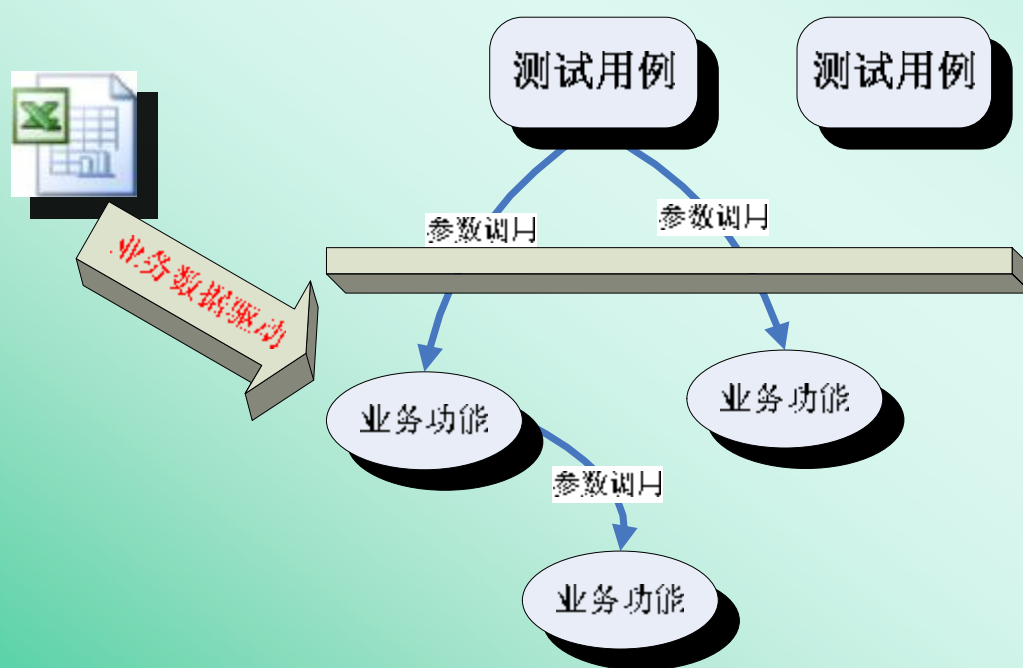
- | 录制+ 回放
- | 录制:执行具体的业务操作,通过工具形成脚本
- | 完善脚本
  - | 添加验证点
  - | 添加数据驱动测试
  - | 设计各种流程
- | 回放:还原被测试环境,回放脚本,查看日志
- | 工具适用的前提:
  - | 能够识别应用的各种对象



# 1: 自动化功能测试实施规划



# 1:自动化功能测试设计



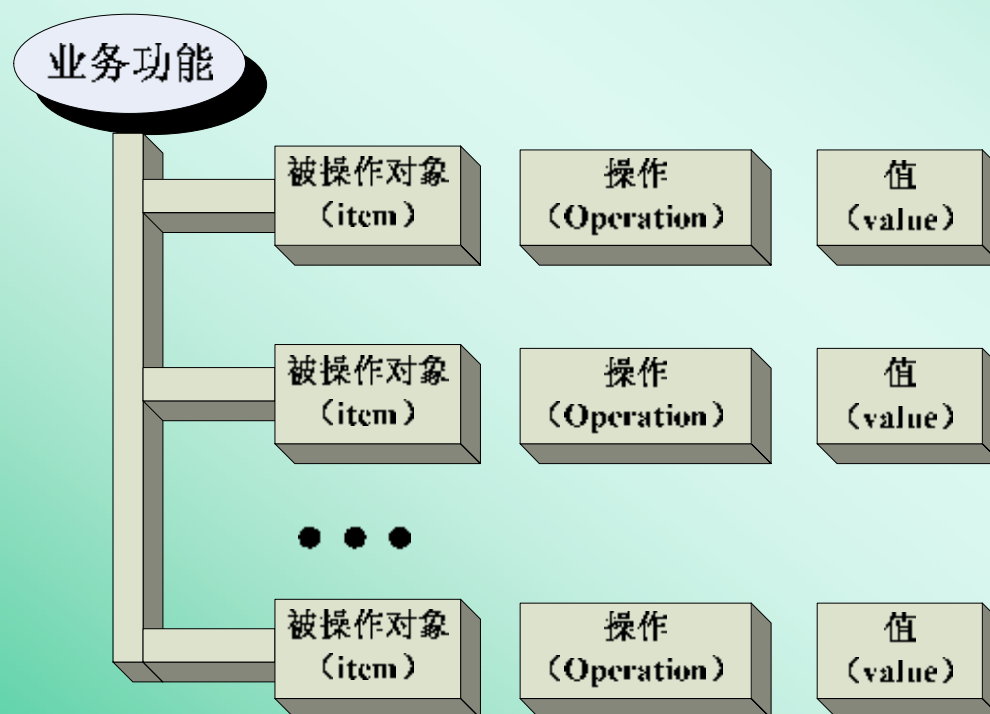
## 测试用例

业务数据驱动业务功能，完成一个完整的测试用例。

## 模块化设计

基于功能分解的思想，把业务对象分解成最基本的业务动作，并建立可独立执行的脚本。

# 1: 自动化功能测试实现



## 关键字驱动

关键字包括三类：被操作对象 (Item)、操作 (Operation) 和值 (value)，且面向对象形式可将其表现为 Item.Operation(Value)



# 1: 自动化功能测试难点

1. 设计——功能分解
2. 实现——对象识别



# 1: 自动化功能测试功能分解

- 画出业务流程图
- 根据业务流程分解业务功能
- 可以被复用的功能也要分解出来
- 按照路径覆盖的思想，组织测试用例

# 1:自动化功能测试对象识别(静态)



序号	Web对象	推荐识别属性	方法
1	Browser	creationTime	
2	Page	index	
3	Frame	name	
4	WebTable	name	
5	WebEdit	name	Set
6	WebButton	text	Click
7	WebList	name	Select
8	Link	text	Click
9	Image	name	
10	WebCheckBox	name	
11	WebRadioGroup	name	
12	WebElement	innerText/html tag	
13	Dialog	text	
14	WinButton	text	

# 1 : 自动化功能测试对象识别(静态)



## I 示例:

1. `Browser("creationTime:=0").Page("index:=0").WebEdit("name:=amount").Set "1000"`

在第一个被创建的IE窗口，查找name属性为amount的输入框，输入数据1000

2. `Browser("creationTime:=1").Page("index:=0").WebButton("text:=保存").Click`

在第二个被创建的IE窗口，查找按钮，该按钮上显示的文字是“保存”，点击该按钮

# 1 : 自动化功能测试对象识别(动态)



## ■ 一个好助手——Object Spy



- 通过Object Spy可以查看到所需要识别的对象的  
所有属性值，挑选其中可以唯一识别该对象的一个  
属性或多个属性，对该对象进行描述。



# 1 : 自动化功能测试对象识别(动态)



- I **GetToProperty**:Returns the value of the specified property from the test object description.  
**GetTOProperties**:Returns the collection of properties and values used to identify the object.  
**GetROProperty**:Returns the current value of the test object property from the object in the application.
- I 举个例子:  
假设在库中有一个对象"窗口A",用于识别该对象的属性有2个,  
一个属性是"text",在库中记录的值是"QQQQ".  
另一个属性是"name",记录的值是"MM"  
在实际运行脚本时属性"text"的值是"PPPP"而不是"QQQQ"  
那么:  
Window("窗口A").GetToProperty("text")返回的是:"QQQQ"  
Window("窗口A").GetRoProperty("text")返回的是:"PPPP"  
Window("窗口A").GetToProperties("text")返回的是用于识别"窗口A"的两个属性和值的集合



# 1 : 自动化功能测试工具

- | WinRunner（原Mercury，现HP）
- | QuickTest Professional（原Mercury，现 HP）
- | Rational Robot（原Rational，IBM）
- | Rational Functional Tester（原Rational，IBM）
- | SilkTest（原Segue，现Borland）
- | .....



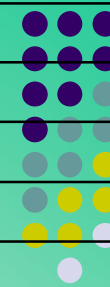
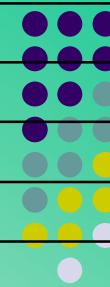
## 2: QuickTest Pro 基本原理

- 对象库Object Repository是QTP的核心和灵魂，在录制的时候，QTP把操作过的所有对象都记录下来，保存在对象库中，记录的形式是一个逻辑名加上若干识别属性。
- 在执行脚本的某个语句时，QTP分析该脚本语句要对哪个对象操作，根据该语句中的对象逻辑名，在对象库中查找对这个对象的详细记载，然后在运行的真实软件中按图索骥，找到需要操作的对象，把语句规定的操作施加在该对象上。施加的主要动作就是把操作的相关消息事件放入到该对象窗口的消息循环队列中。



## 2: QuickTest Pro 插件

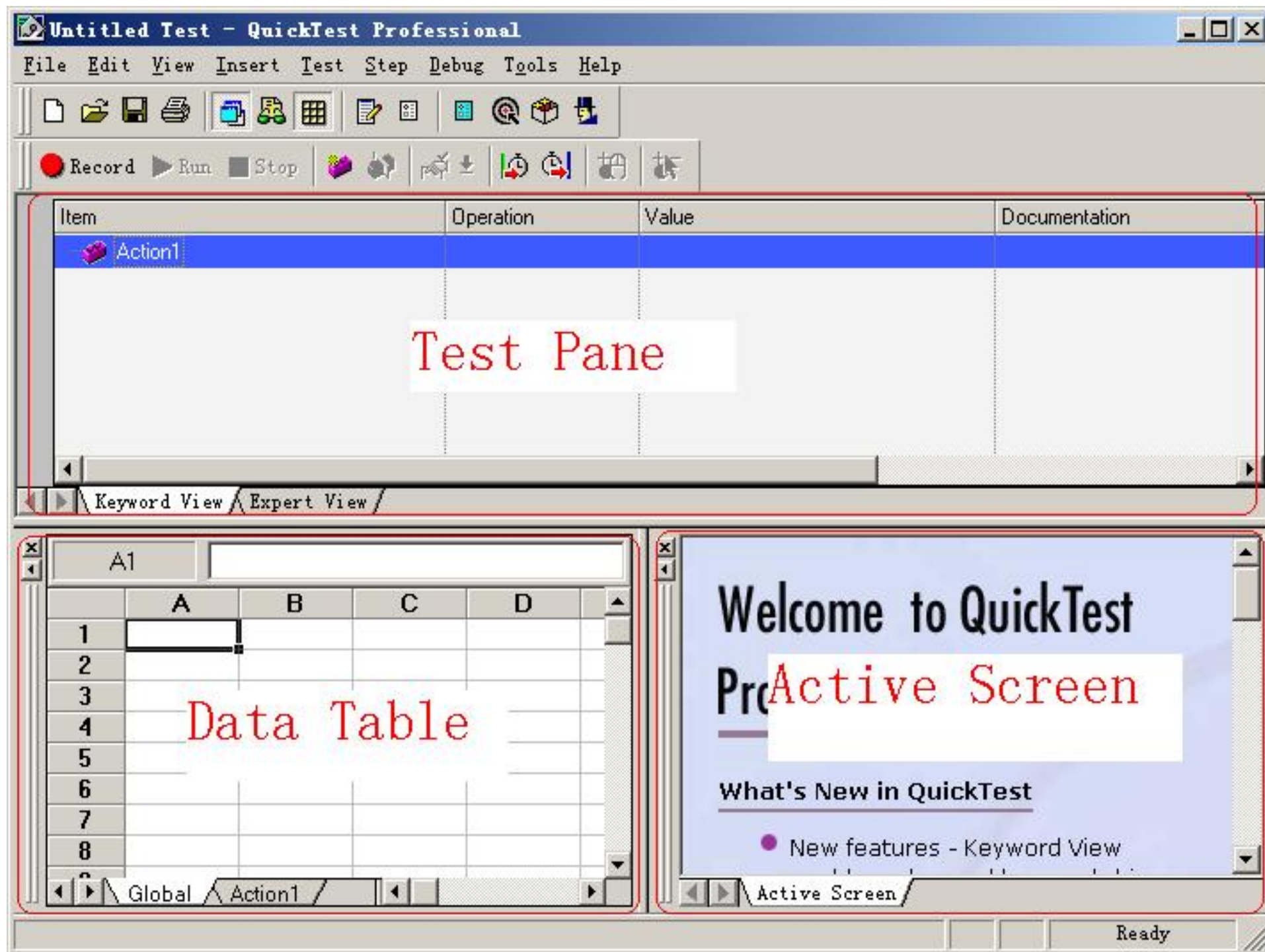
- | .NET
- | JAVA
- | Web services
- | Visualage
- | Peoplesoft
- | Stingray
- | SAP
- | Oracle
- | TE(Terminal Emulator )
- | Siebel
- | Delphi

项目	开发环境	
Web-Related Environments	IE, Netscape, AOL	
	JDK, Java Foundation Classes, AWT	
	Symantec Visual Café	
	ActiveX Controls	
ERP/CRM	Oracle: Jinitiator, 11i, NCA	
	SAP	
	Siebel 7.x	
	PeopleSoft 8.x	
Custom Client Server	Windows	
	C++/C	
	Visual Basic	
Operating Systems	Windows 98, 2000, NT, ME, XP	
Legacy	3270, 5250 Emulators	
	VT100	
.Net	WinForms	
	WebForms	
	.Net controls	
Web Services	XML, HTTP	
	WSDL, SOAP	
	J2EE, .Net	
Multimedia	Real Audio/Video	
	Flash	

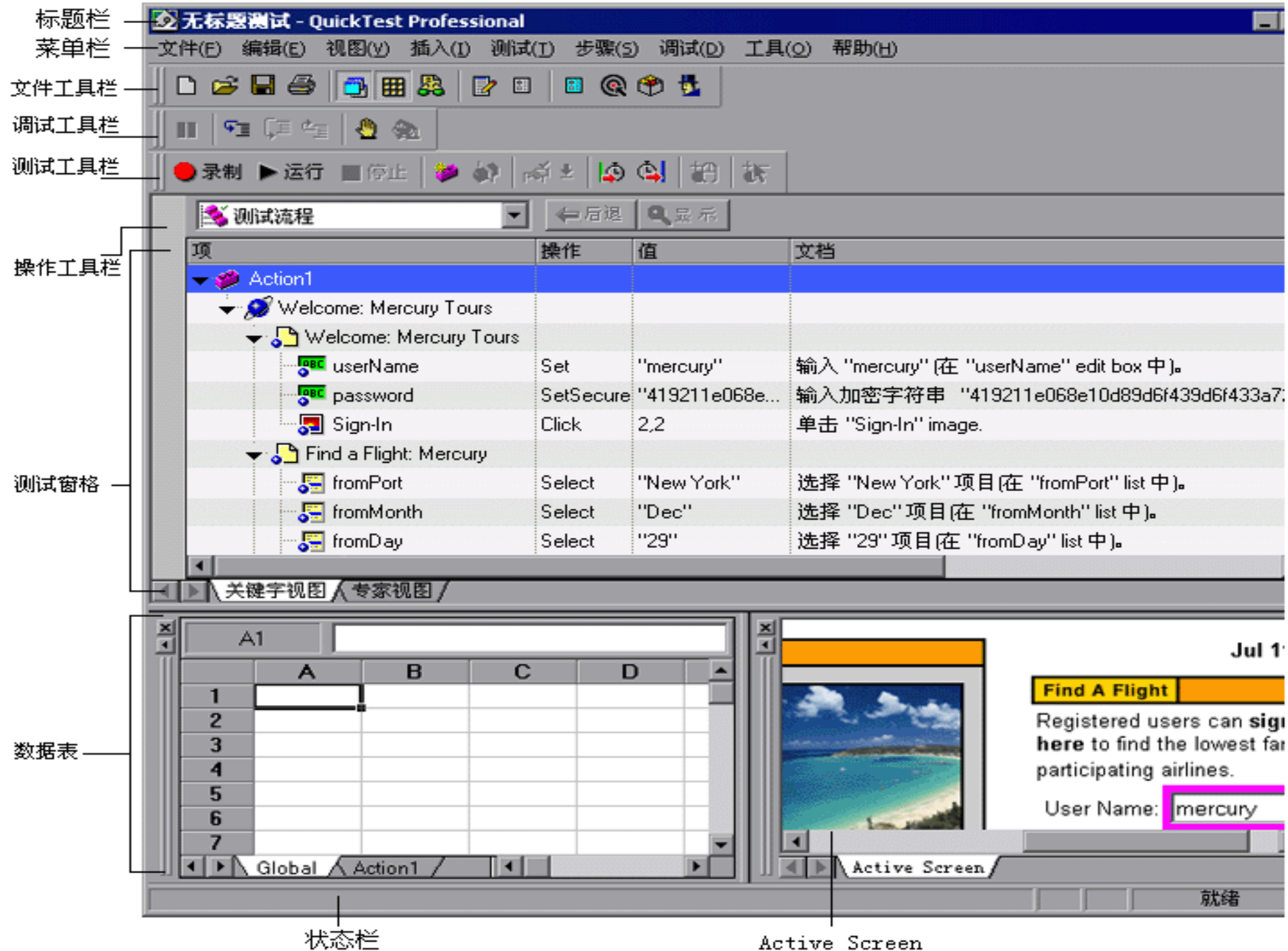


## 2.: QuickTest Pro 界面介绍

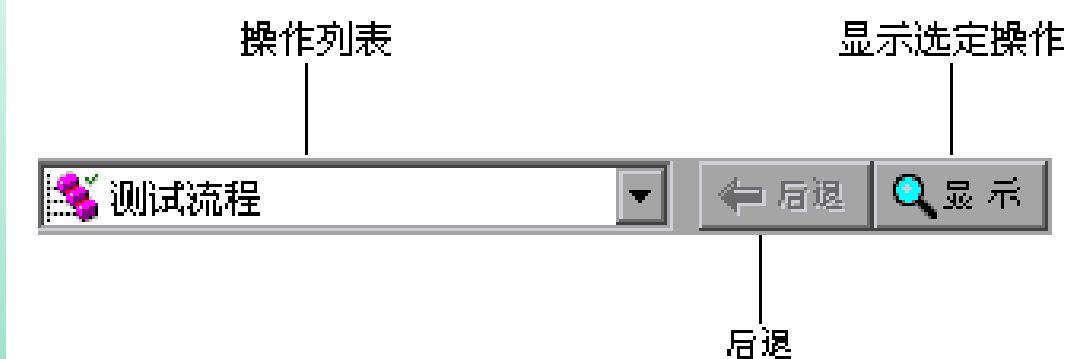
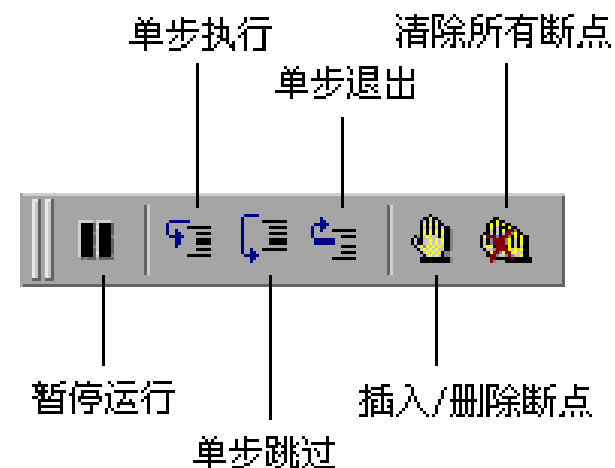
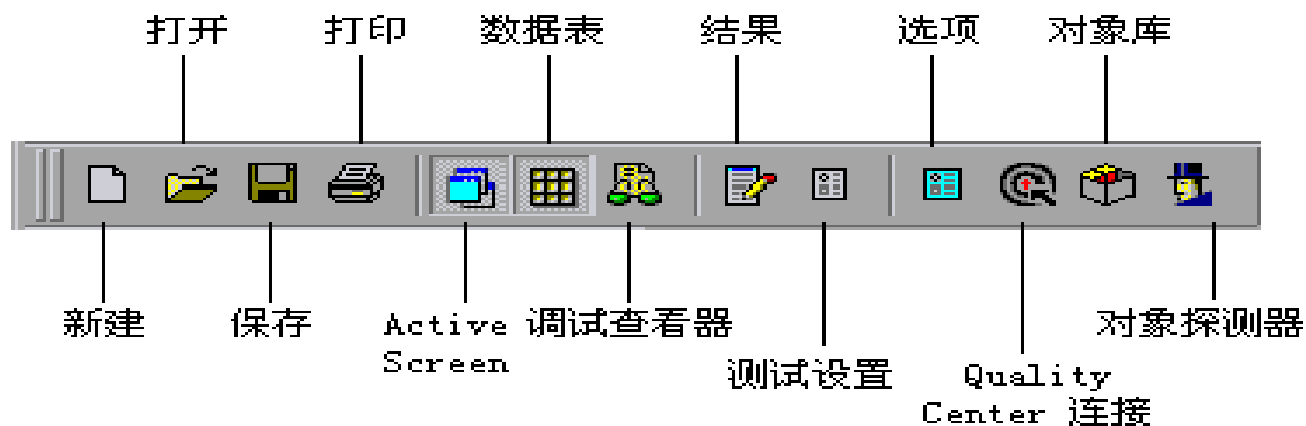
- 1 **Test Pane:** 包括Keyword View 和Expert View两个视图。图中显示的是关键词视图，录制生成的脚本可以在这里很直观的看到，可以在此视图完成参数化的工作。Expert View可以在在此视图中直接修改生成的脚本，适合对VB脚本和QTP的函数比较熟悉的测试人员使用
- 1 **Data Table:** 就是一个Excel，用于提供自动化测试脚本所需的输入数据或者校验数据。指向测试脚本文件目录下的Default.xls文件。可以直接在Excel中编辑数据
- 1 **Active Screen:** 录制脚本时生成，记录下Web页面，可以在此完成大量的修改脚本工作，如添加检查点等













## 3: QTP的测试流程

- 1、设计测试数据
- 2、录制测试脚本
- 3、修改并调试测试脚本
- 4、执行测试脚本
- 5、分析测试结果
- 6、报告发现的缺陷

### 3 : QTP的测试流程



- | 分析业务，设计测试用例，决定要校验哪些功能点
- | 设计测试用例相关的测试数据
- | 决定哪些用例手工执行，哪些用例使用**QTP**执行
- | 合理安排录制脚本的顺序，使生成的脚本复用



## 3: QTP的测试流程

例：现在要测试某个系统登录页面，详细的测试数据如下：

步骤	原始数据	输入	执行动作	预期结果
1	用户名: <a href="#">test@test.com</a> 密码为test	用户名: 空 密码: 空	提交	弹出对话框, 提示“all required fields must be filled in”
2		用户名: 1 密码: 空	提交	弹出对话框“all required fields must be filled in”
3		用户名: 空 密码: 1	提交	弹出对话框“all required fields must be filled in”
4		用户名: <a href="#">test@test.com</a> 密码: 1	提交	提示“Password does not match for : <a href="#">test@test.com</a> ”
5		用户名: test 密码: test	提交	提示“Could not find account for : test”
6		用户名: <a href="#">test@test.com</a> 密码: test	提交	登录成功



### 3: QTP的测试流程

当在Web页面或者其它的应用程序的GUI上执行操作时，QTP会记录下你的操作步骤，并产生测试脚本，在Keyword View中以表格的方式显示测试脚本的操作步骤（steps）。根据测试数据，执行后生成的脚本如下：

### 3: QTP的测试流程



QTP Test Flow Table:

Object	Action	Value	Description
Plants by WebSphere			
Plants by WebSphere			
banner			
LOGIN	Click		Click the "LOGIN" link.
work			
userid	Set	"test@test.com"	Enter "test@test.com" in the "userid" edit box.
passwd	SetSecure	"488830fc591b7894a2f23bed8d7c"	Enter the encrypted string "488830fc591b7894a2f23bed8d7c" in the "passwd" edit box.
Sign in	Click	24,13	Click the "Sign in" image.
Plants by WebSphere	Sync		Wait for the Web page to synchronize before continuing the run.

Keyword View / Expert View

Test Data Table:

A1					
A	B	C	D	E	F
1					
2					
3					
4					
5					
6					
7					
8					
9					

Web Page Screenshot:

PLANTS BY WEBSHERE

Flowers Fruits & Vegetables Trees Accessories HOME : SHOPPING CART : LOG

E-mail address: test@test.com

Password:

Sign in



### 3: QTP的测试流程

- | 添加对象和步骤
- | 参数化
- | 增加条件控制
- | 建立检查点
- | 建立输出值
- | 分割脚本





## 3: QTP的测试流程

### I 添加对象和步骤

如果在我们录制的脚本中，缺少用户名和密码的输入框这两个对象。  
下面增加用户名和密码的输入框：

Ø选中“Plants By WebSphere”这个Page对象,点击insert new step

Ø如果你所要添加的控件QTP已经识别过类似的控件，你可以使用这个类似的控件，并且通过修改属性来添加这个新的步骤，如果QTP没有识别过，选择step generator，弹出对话框

Ø点击图中的  标按钮，弹出对话框，点击其中的按钮  鼠标会从箭头变为手型指针,在你要测试页面上选中你要添加步骤所需要的控件—用户名和密码的输入框





## 3: QTP的测试流程




### I 添加对象和步骤

- Ø 添加用户名和密码输入框中的值（**value**），“ ”是指在输入框中输入空。点击**ok**，所选对象的属性将被保存到对话框中
- Ø 最后，使用**Test—Update Run**方式运行脚本。**Update Run**方式运行脚本，可以在运行的同时更新**Active Screen**，**checkPoint**以及**Test Object**。特别是更新**Active Screen**的功能，手工添加的步骤使用普通方式运行脚本，不会生成**Active Screen**，这样就无法添加检查点
- Ø 另外，也可以在对象库（**Object Reponsitory**）中添加对象



## 3: QTP的测试流程

### I 参数化

- I 根据测试数据，可以发现，执行的所有操作，只是修改登陆名和密码，所以我们可以把这两个输入值参数化，使得脚本执行时，自动取DataTable的值来运行。调出参数化窗口的方法有两个：
  - Ø 在KeyWord View中点击想要被参数化的值，如图中的“test1”，再点击出现的图标，或者使用快捷键“Ctrl+F11”；
  - Ø 在  userid  这个输入框对象上，右键，弹出对话框，选择“Object properties”



## 3: QTP的测试流程

### I 参数化

- I 参数化时，可以选择使用Global Sheet或者Local Sheet，当把参数化的值放到Global Sheet中，就成为全局变量，放到Local Sheet中后，就是局部变量，只在本Action中可以调用。一般建议使用Local Sheet。
  - Ø Global全局参数：它控制整个Action的运行次数,global里有几行参数,则运行几次脚本。
  - Ø Current Action(Local Sheet): 对应于单一的Action的参数,对于一个Action没有实际的意义,而几个Action共同在一个脚本中使用时,就由他来控制单一Action的循环次数。



## 3: QTP的测试流程

### I 增加条件控制

**If** 条件1正确 **Then**

执行语句1

**Elseif** 条件2正确 **Then**

执行语句2

**Else**

执行语句3

**End If**

可以添加**Elseif**子句以扩充 **If...Then...Else** 语句的功能，可以控制基于多种可能的程序流程



## 3: QTP的测试流程

### I 建立检查点


验证在测试运行时应用程序是否可以显示预期的信息

检查点类型	描述	用法示例
标准检查点	检查对象的属性值。	检查是否选中某单选按钮。
图像检查点	检查图像的属性值。	检查图像源文件是否正确。
表检查点	检查表中的信息。	检查表单元格中的值是否正确。
页面检查点	检查网页的特性。	检查加载网页所需的时间，或者检查网页是否包含中断链接。
文本/ 文本区域检查点	检查文本字符串是否显示在网页或应用程序窗口中的适当位置。	检查预期的文本字符串是否显示在网页或对话框上的预期位置。
位图检查点	将网页或应用程序的某个区域捕获为位图后对其进行检查。	检查网页或网页的任何部分是否能按预期显示。
数据库检查点	检查应用程序或网站所访问的数据库内容	检查数据库查询中的值是否正确。
可访问性检查点	对网站区域进行识别，以检查是否符合 508 部分。	检查网页上的图像是否包含 ALT 属性（该属性是 W3C Web 内容可访问性规则所要求的）。
XML 检查点	检查 XML 文档的数据内容。	<b>注意：</b> XML 文件检查点用于检查特定的 XML 文件；XML 应用程序检查点用于检查网页中的 XML 文档。



## 3: QTP的测试流程

### I 建立检查点

- I 弹出对话框的内容是我们需要校验的重要信息，可以在弹出对话框的文字上添加标准检查点：
  - ∅ 在Active Screen中显示的弹出对话框的文字上右键，选择“Insert Standard checkpoint”，在弹出框中选择“Static: 请输入密码”，然后点击“OK”按钮
  - ∅ 选中“请输入密码”的字样，然后在Configure value的标签中选中Parameter的单选框，点击  图标按钮
  - ∅ 在name的输入框中输入你给弹出框内容起的名字，做一个标识，如“校验信息”，这对应到DataTable中的列名。同样，建议使用Local Sheet



## 3: QTP的测试流程

### I 建立输出值

与建立检查点类似，可以在Active Screen中右键“Insert Output Value”，操作方式相同。可以把弹出框的信息输出到DataTable中去。



## 3: QTP的测试流程

### I 分割脚本

- I 可以把在页面输入和对弹出框的校验分成两个Action。
  - ∅ 在Keyword View 中，选中弹出对话框的对象，点击“Step”菜单，选择“Split Action”
  - ∅ 分别给页面输入和弹出框两个Action命名。点击“OK”按钮
- I 这样，原来的一个Action就分成了两个Action。任何一个Action都可以被其它的Action调用。
- I 在测试登录的这个例子中，不需要分割Action。





### 3: QTP的测试流程

- I 在DataTable中输入预先设计好的测试数据，然后执行编辑好的测试脚本：
- I Browser("Plants by WebSphere").Page("Plants by WebSphere").Frame("banner").Link("LOGIN").Click
- I Browser("Plants by WebSphere").Page("Plants by WebSphere").Frame("work").WebElement("Password:").Check  
CheckPoint("Password:")
- I Browser("Plants by WebSphere").Page("Plants by WebSphere").Frame("work").WebElement("Password:").Output  
CheckPoint("Password:\_2")
- I Browser("Plants by WebSphere").Page("Plants by WebSphere").Frame("work").WebEdit("userid").Set  
"test@test.com"



### 3: QTP的测试流程

- | `Browser("Plants by WebSphere").Page("Plants by WebSphere").Frame("work").WebEdit("passwd").Set Secure "49a351520f66c3d526bf2effb1e3"`
- | `Browser("Plants by WebSphere").Page("Plants by WebSphere").Frame("work").Image("Sign in").Click 32,12`
- | `Browser("Plants by WebSphere").Page("Plants by WebSphere").Sync`
- | `Browser("Plants by WebSphere").Close`
- | 在Keyword View 中选中Action，右键，点击“Action Call Properties”，弹出对话框，显示执行本Action的次数。选择“Run on all rows”，执行所有的数据。



### 3: QTP的测试流程

执行完测试数据后，或者意外中断，测试执行结束，就会生成测试报告。测试报告也是一个树形结构，需要注意标记为“X”的报告项。如果有标记为“X”的报告项，可能是执行脚本出错，或者是检查点校验没有通过，可能是一个缺陷。

para [TempResults] - Test Results

File View Tools Help

Test para Summary

Run-Time Data Table

para Iteration 1 (Row 1)

Action1 Summary

Plants by WebSphere

Plants by WebSphere

Checkpoint "Plants by WebSphere"

banner

work

userid.Set

userid

Checkpoint "userid\_2"

passwd.SetSecure

Sign in.Click

banner

work\_2

work\_3

work\_4

Checkpoint "work\_4"

Output "work\_5"

Checkout Now.Click

work\_5

work\_6

Checkout Now.Click

Plants by WebSphere.Sync

Plants by WebSphere.Close

para Iteration 2 (Row 2)

Action1 Summary

Plants by WebSphere

Plants by WebSphere

Checkpoint "Plants by WebSphere"

banner

work

banner

work\_2

work\_3

work\_4

work\_5

work\_6

Plants by WebSphere.Sync

Plants by WebSphere.Close

para Iteration 3 (Row 3)

Standard Checkpoint "userid\_2": Passed

Date and Time: 2008-10-27 - 13:56:41

Details

userid\_2 Results

Property Name	Property Value
html tag	INPUT
innertext	
name	userid
type	text
value	test@test.com

PLANTS BY WEBSHERE

FlowersFruits & VegetablesTreesAccessories

HOME : SHOPPING CART : LOGIN : HELP

Home

Login or Register

If you are a returning customer and previously set up an account, please enter your e-mail address and password below.

E-mail address: test@test.com

Password:

Sign in

Result Details

Screen Recorder

For Help, press F1

Ready



### 3: QTP的测试流程

如果QTP和缺陷管理工具如Test Director以及Quality Center连接起来，就可以把QTP中执行测试用例中发生的错误以及发生错误时的截图发到缺陷管理工具中去，生成缺陷报告。

## 4: QTP上机实践

参考《QTP上机实习》





# 自动化功能测试应用分析

- | 有图形用户界面的应用系统
- | 需求变化不大的核心流程
- | 需求变化不大的主要功能点
- | 覆盖率要求高的主要功能点
- | 适合于回归次数多的测试策略
- | 自动化测试实现案例复用技术

# 问题与讨论

---

