

QTP 学习与实践经验总结

前言:

QTP 全名 Quick Test Professional,是 MI 公司出品的一款测试工具.它的优点是使用方便,功能强大并且可以通过与测试管理工具的互连达到自动化功能测试的目的. QTP 是专门针对 B/S 模式的测试工具.它具有:识别能力强,回放精确等优点,是进行 B/S 模式下功能测试的首选工具.下面是本人学习 QTP 的经验总结,它包括了在学习阶段和使用阶段所遇到的问题.当然我这里所说的还很不全面,有很多欠缺,希望大家通过上面的联系方式与我联系,大家共同学习.

1.1	TD 与 QTP 的连接.....	5
1.2	QTP 自动提交错误到 TD 中.....	13
1.3	本机创建,修改与提交.....	16
1.4	QTP fail 自动截图.....	20
1.5	上网助手等插件对 QTP 的影响.....	23
1.6	防火墙对 TD 与 QTP 的连接的影响(瑞星 2005 下).....	24
1.7	如何定时运行 QTP.....	26
2	QTP 的录制.....	28
2.1	QTP 的录制方法.....	29
2.2	QTP 录制应先作录制计划.....	28
2.3	关于下拉菜单的更改.....	29
2.4	输入法对于回放的影响.....	32
2.5	如何修改系统时间.....	32
3	QTP 的自定义操作.....	33
3.1	手工添加步骤.....	33
3.2	手工识别控件.....	44
4	多个 Action 一起执行.....	47
4.1	为什么会用到多个 Action 调用.....	47
4.2	对被调用 Action 的设定.....	48
4.3	调用 Action.....	51
4.4	调用 Action 的模式.....	58
4.5	copy Action 与 call existing Action 的区别.....	58
5	对于 QTP 一些功能键的介绍.....	59
5.1	关于录制设置的区分 web event recording configuration.....	59
5.2	关于 QTP 录制 mouseover 得问题.....	63
5.3	关于 start transAction 的问题.....	63
5.4	虚拟对象的使用.....	67
5.5	对于 Action screen 捕捉不精确的解决办法.....	71
5.6	QTP 录制方式的分类.....	74
6	对于内部控件属性的一些设定.....	错误!未定义书签。
6.1	关于 regular expressions.....	74
7	关于 datatable.....	79
7.1	global 与 current 的区别.....	79
7.2	import excel 的问题.....	80
7.3	关于 editing the data table.....	82
8	关于检查点.....	83
8.1	checkpoint 的返回值.....	83
8.2	QTP 检查点的设定.....	86
8.3	关于文字检查点得参数化.....	86
8.4	设立可重用检查点.....	92
8.5	QTP 检查点添加有问题,造成无法识别控件,导致脚本无法运行的问题.....	94
8.6	标准检查点.....	94
9	关于错误报告的问题.....	97
9.1	关于 report.ReportEvent.....	97

9.2	关于 reporter.fiter.....	101
10	关于在 expert 中操作的问题.....	104
10.1	关于 getRoproperty ,getToproperty, getToproperty	104
10.2	编码时在出现 expect end of statement 的问题.....	108
11	关于 C/S 模式的录制.....	112
11.1	QTP 是否可以 cs 方式下的应用系统.....	112

1 TD 与 QTP 的连接

1.1 TD 与 QTP 的连接

A 问题描述：使用 TD 和 QTP 互联可以实现如下功能：

- A1 测试版本管理
- A2 远程控制实现自动化测试以及回归测试

B 处理流程：

- B1 安装 TD 和验证
- B2 安装 QTP 和验证
- B3 安装插件
- B4 QTP 参数设置
- B5 验证 TD 和 QTP 互联成功

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具，他采用 B/S 模式，使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具，主要用于 B/S 模式的系统测试

D 实例

D1 安装 TD，详细安装见 TD 说明书

检验 TD 是否安装成功，启动 TD Server 服务器后，在与 Server 相连的其他计算机上，打开 IE 浏览器，在地址栏输入 http://TD Server 的 ip 地址/TDbIn/start_a.htm 如果能够浏览到 TD 界面,说明安装成功，界面见下图

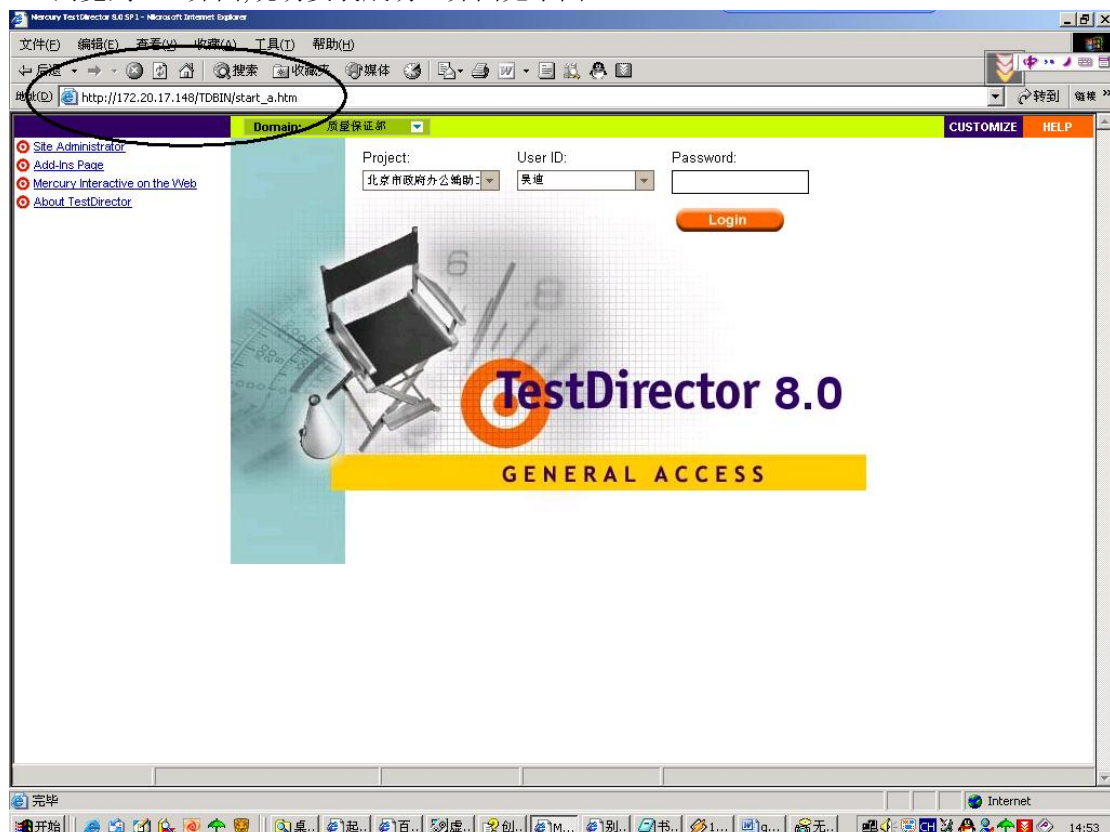


图 1.1.D1.1

D2 安装 QTP，详细安装见 QTP 使用说明书

验证 QTP 安装成功，进入 windows 操作系统，点击，开始->程序->Quick Test Professional-> Quick Test Professional，打开 QTP 使用界面，如下图

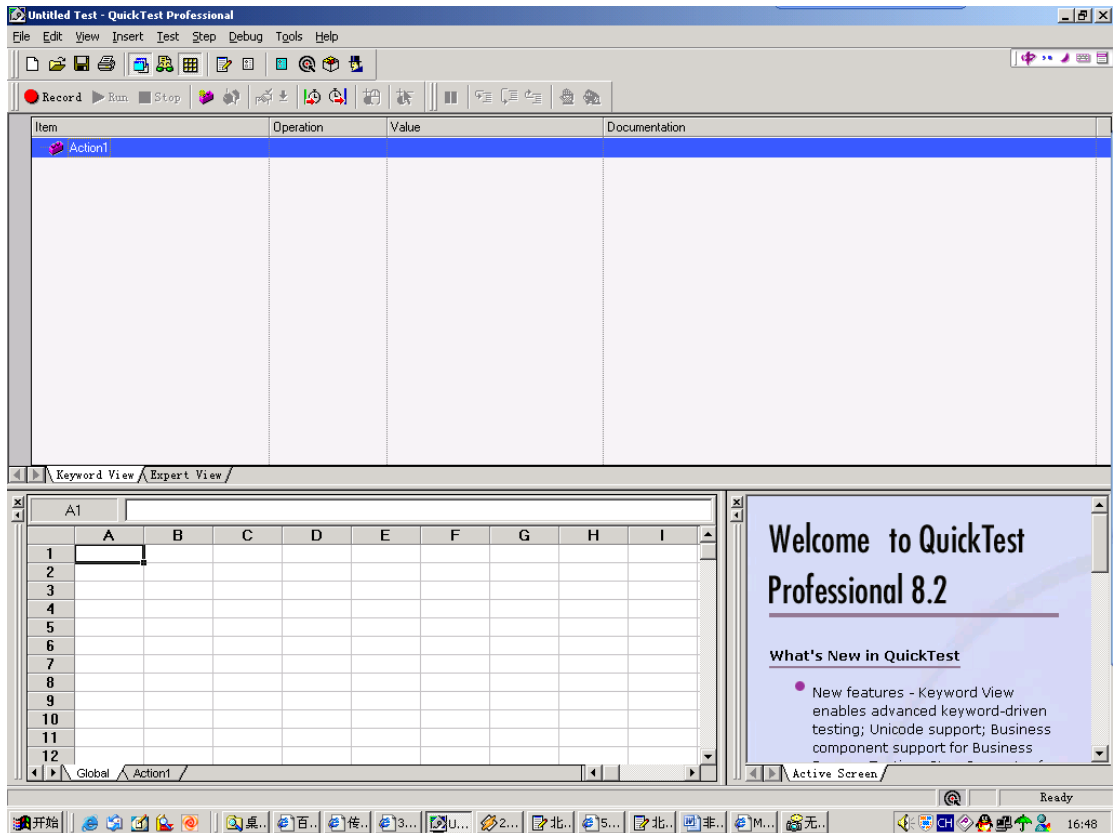


图 1.1.D2.2

进行简单的录制，保证脚本录制成功，并且可以回放，与你的预操作达到的结果一致，证明 QTP 安装成功

D3 安装插件，进入 TD 首页，点击见面上的 TD Add In 如下图

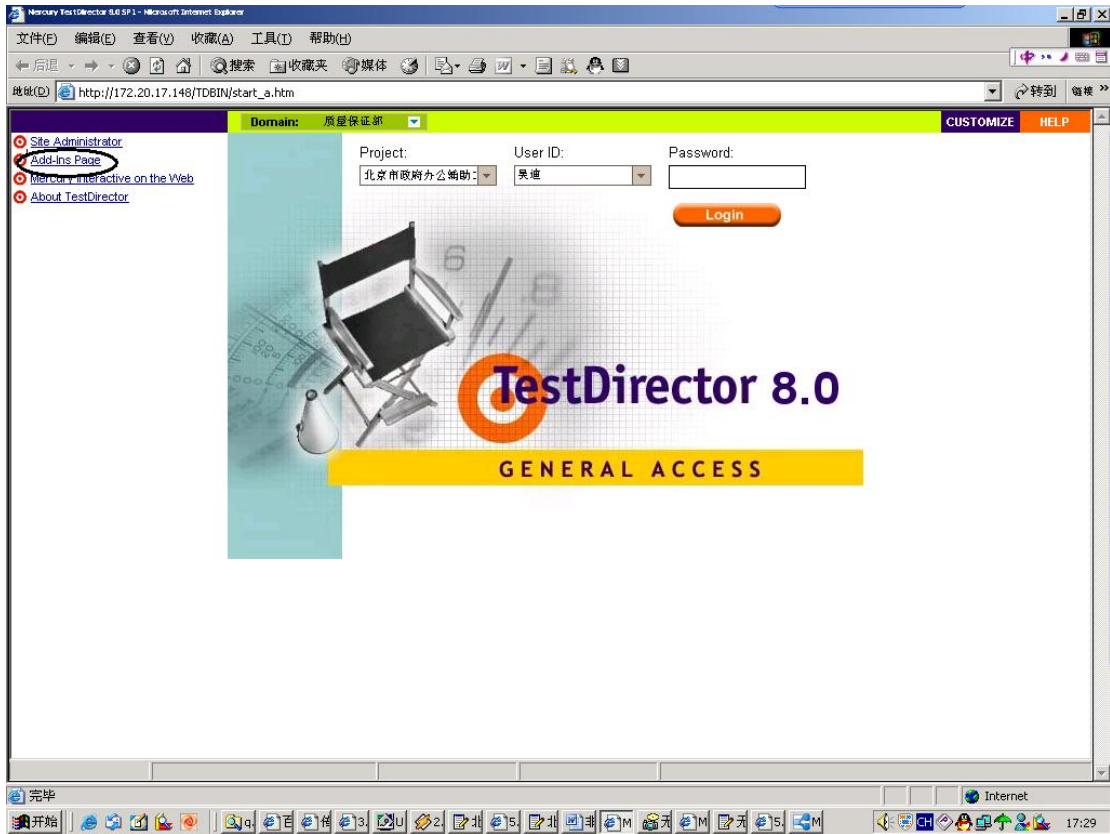


图 1.1.D3.3

进入 TD Add In 界面点击 More TestDirector Add Ins 进入 More TestDirector Add Ins 界面，如下图

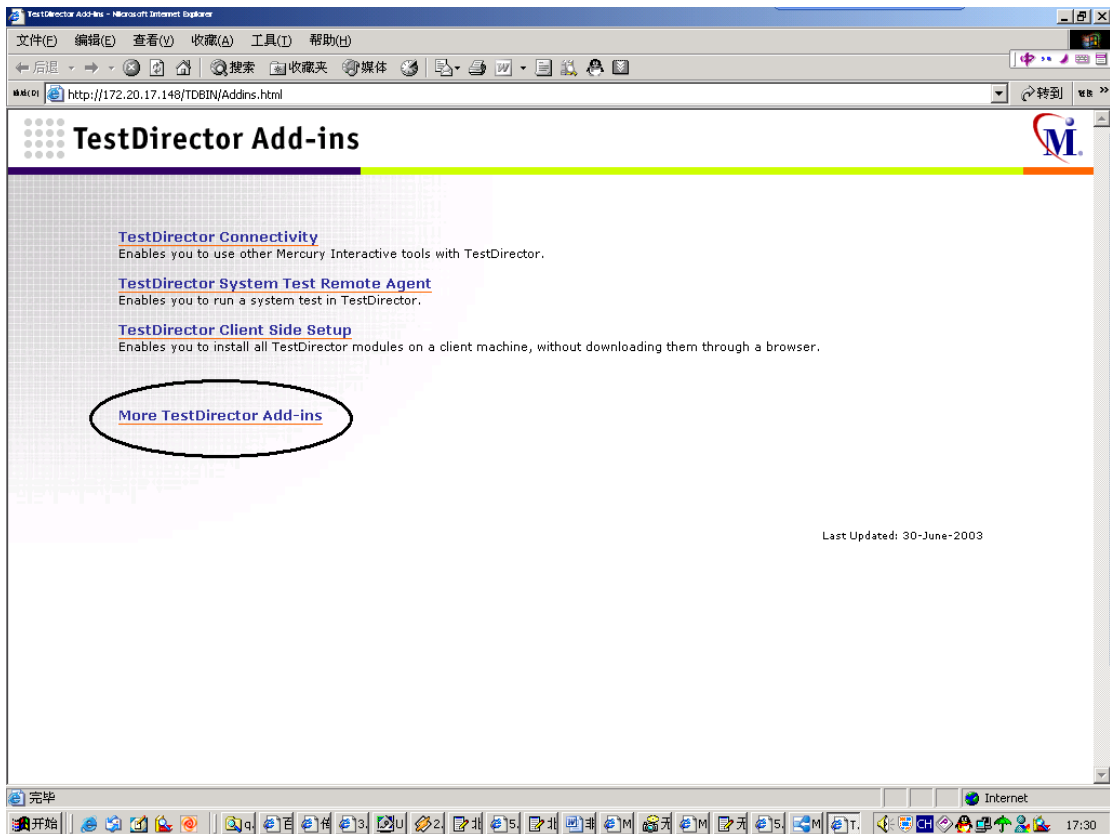


图 1.1.D3.4

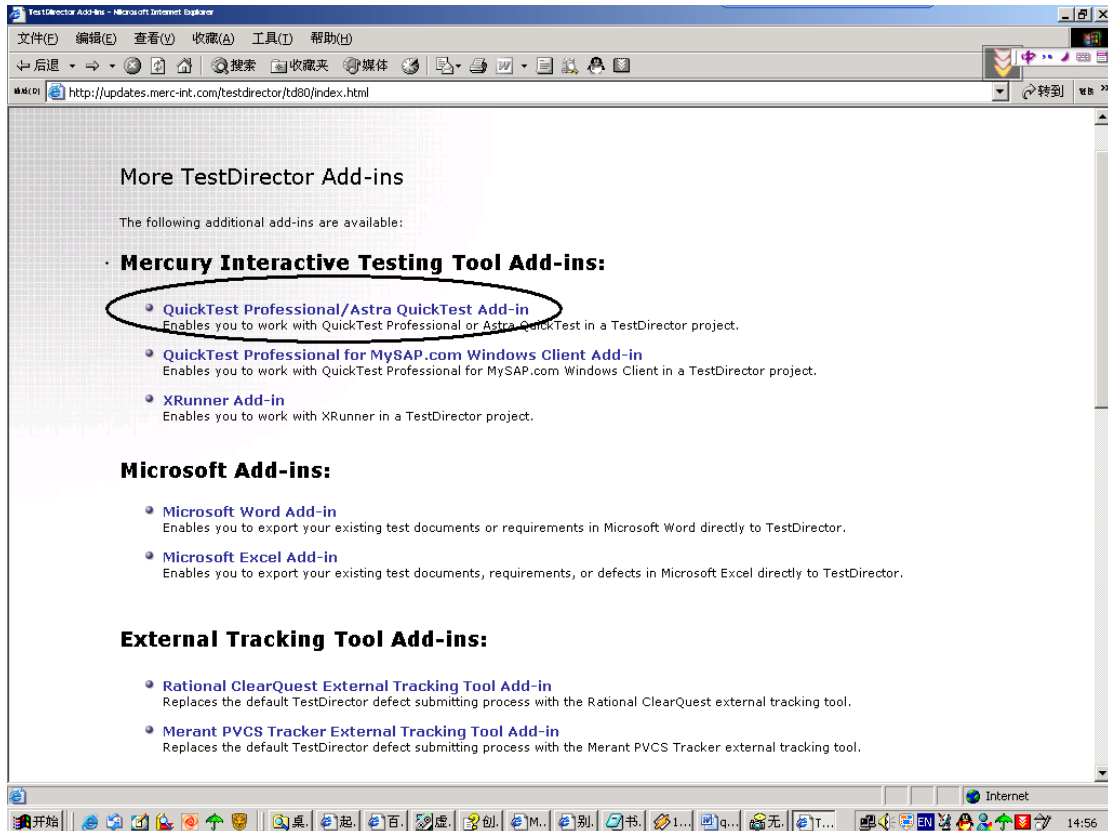


图 1.1.D3.5

下载名为 TDplugInsSetup 的插件，大约 35 兆

并且在 Server 与 QTP 客户机上安装这个插件，安装成功后，会要求你重启电脑，重启后会看到，开始一程序中有个新安装的程序“Quick Test Professional add in for Quality Center”，登入 TD 进入 test plan 检测是否可以在 test plan 中添加 QTP 类型的测试用例如下图：

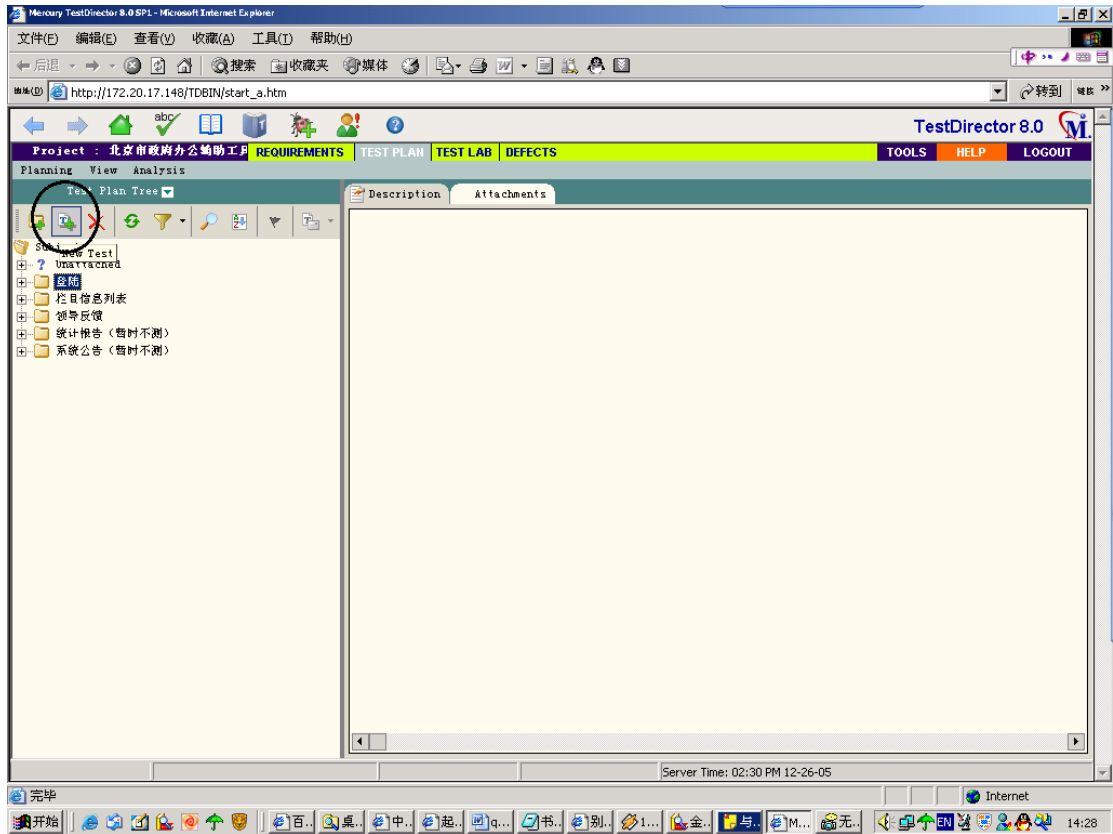


图 1.1.D3.6

点击后,出现如下图对话框,并且可以建立 QTP 的测试用例

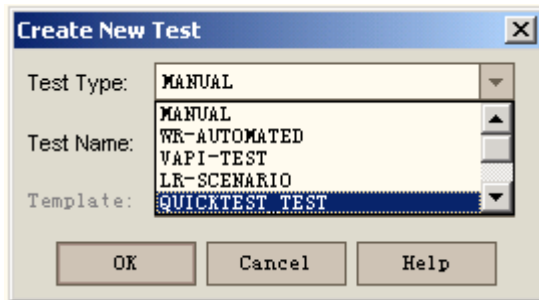


图 1.1.D3.7

检查没有问题之后,还需要在安装 QTP 的机子上做一些相应的设定
D4 对于 QTP 的一些设定

进入 QTP 主界面,点击文件栏的 Tools-Options,进入 Options 对话框

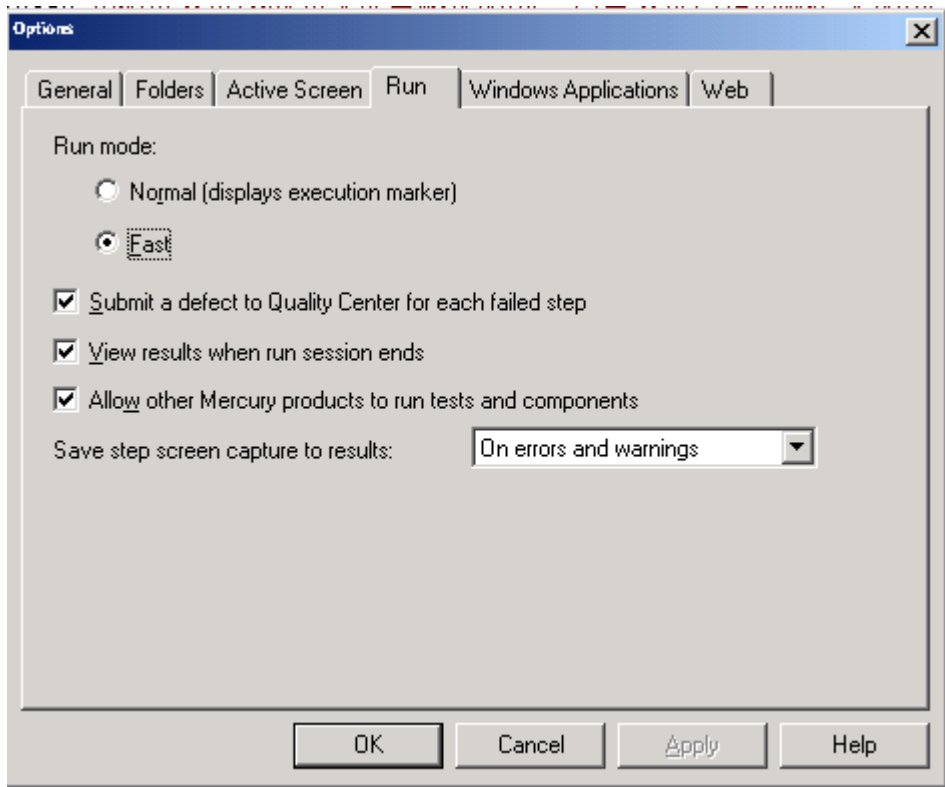


图 1.1.D4.8

在 Run Tab 中选择 Allow other Mercury products to run test and components,以及 Submit a defect to Qualiyt Center for each failed step 选中后,点击 ok 保存这个设置
 在 QTP 主界面中点击 Tool—Quality Center Connection

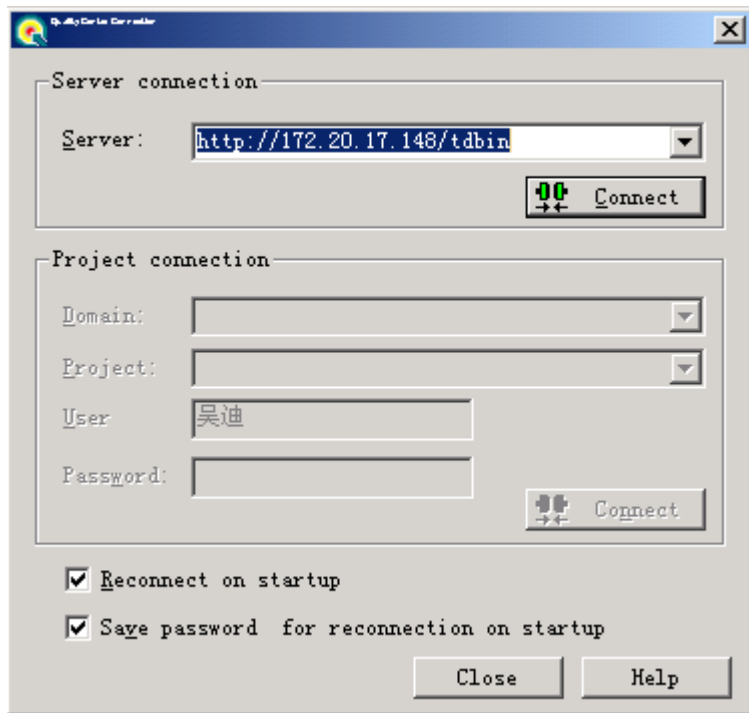


图 1.1.D4.9

在 Server 中输入 http:// TDSeriver ip 地址/TDbIn(本例中使用 htt.....),点击 connect 按钮,QTP 连接到服务器上,如果服务器连接有问题,则会提示连接失败

在 project connection 中包含一下几项:

Domain: TD 服务器站点管理中建立项目的域, 如下图

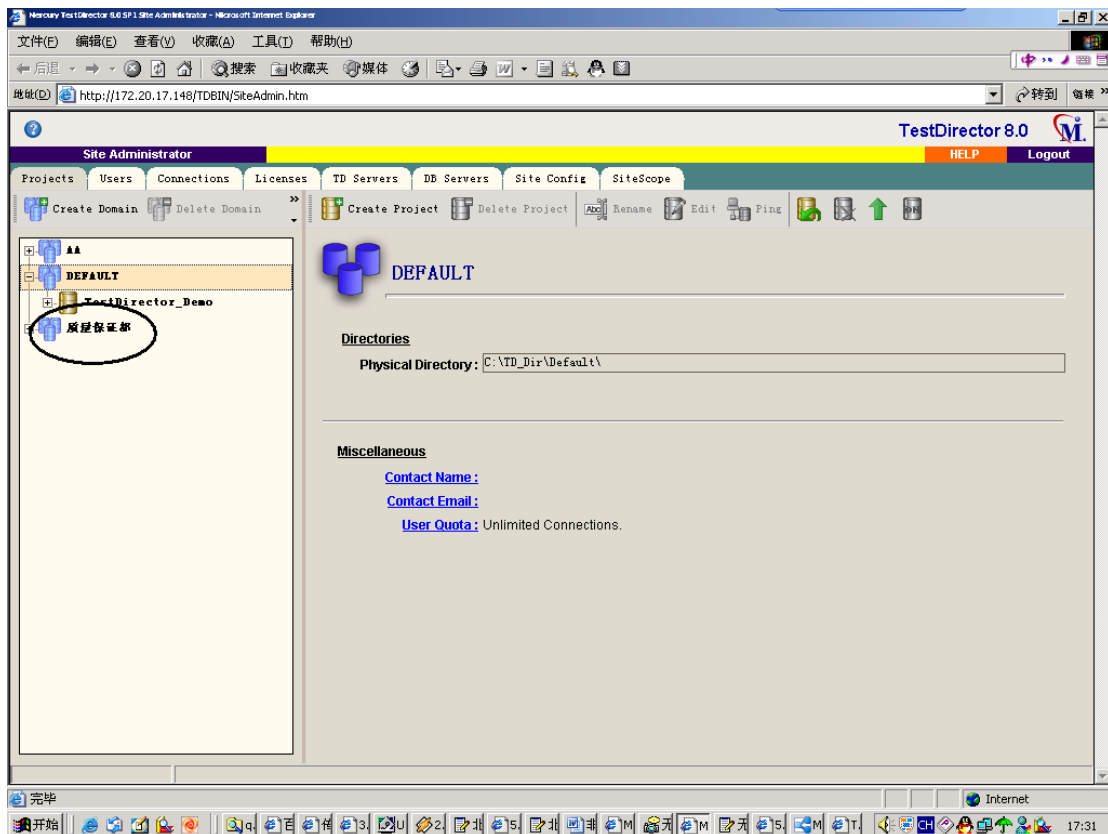


图 1.1.D4.10

Project: 域中所建立的项目

User: 登陆项目所用的用户名

Password: 相应用户名使用的密码

Reconnect on startup 在下次启动 QTP 时自动登陆到你的项目中

Save password for reconnection on startup 为下次登陆保存密码

完成了上述操作就 TD 和 QTP 就连接起来, 可以从 TD 调用 QTP 脚本了
连接成功后会在 QTP 右下角出现一个标志, 如下图:

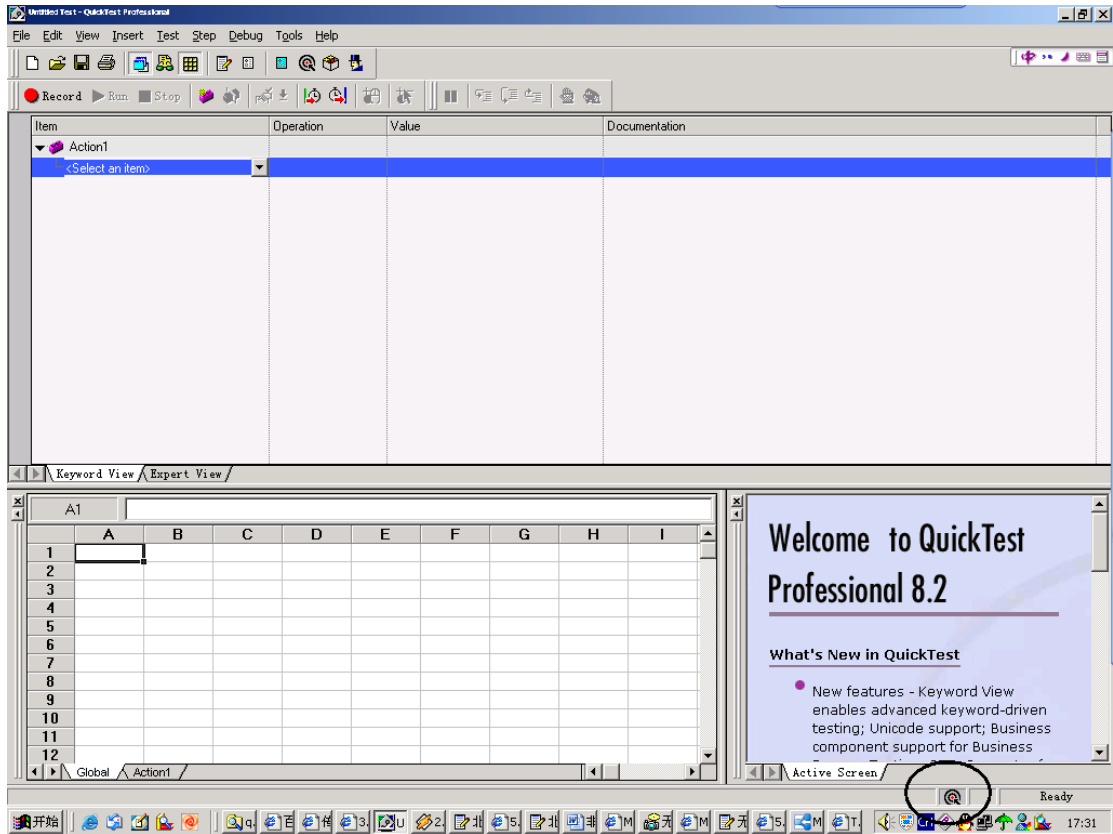


图 1.1.D4.11

出现这个标志说明连接成功

E 个人感受:

TD 是测试管理工具,单独使用它,只能管理手工测试的用例,无法做到回归测试

QTP 是自动化功能测试工具,单独使用它,虽然能完成回归测试,但是无法进行测试版本管理

通过 TD 与 QTP 的联合使用,有以下几个优点

1. 可以通过 TD 管理 QTP 的测试脚本
2. 通过 TD 命令在其他计算机执行在本机录制的 QTP 脚本,以节省时间,让自己可以继续录制其他用例的脚本

1.2 QTP 自动提交错误到 TD 中

A 问题描述:

QTP 和 TD 相连后, QTP 通过在脚本中设立的检查点, 来检查运行结果是否与预期结果一致, 并将错误的结果值返回到 TD 中, 以便开发人员查看

B 处理流程:

B1.在 QTP 中的参数设置

B2.在 TD 中执行 QTP 脚本, 并且自动获得 QTP 提交的错误报告

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 QTP 与 TD 连接之后能够通过设定的检查点,自动向 TD 报送错误,并且将错误保存在 TD 的 defect 中

在 QTP 主界面, 选择 tools-options 进入 option 对话框

在 run tab 中选择 submit a defect to Quality Center for each failed step, 并且保存这个设置, 如下图

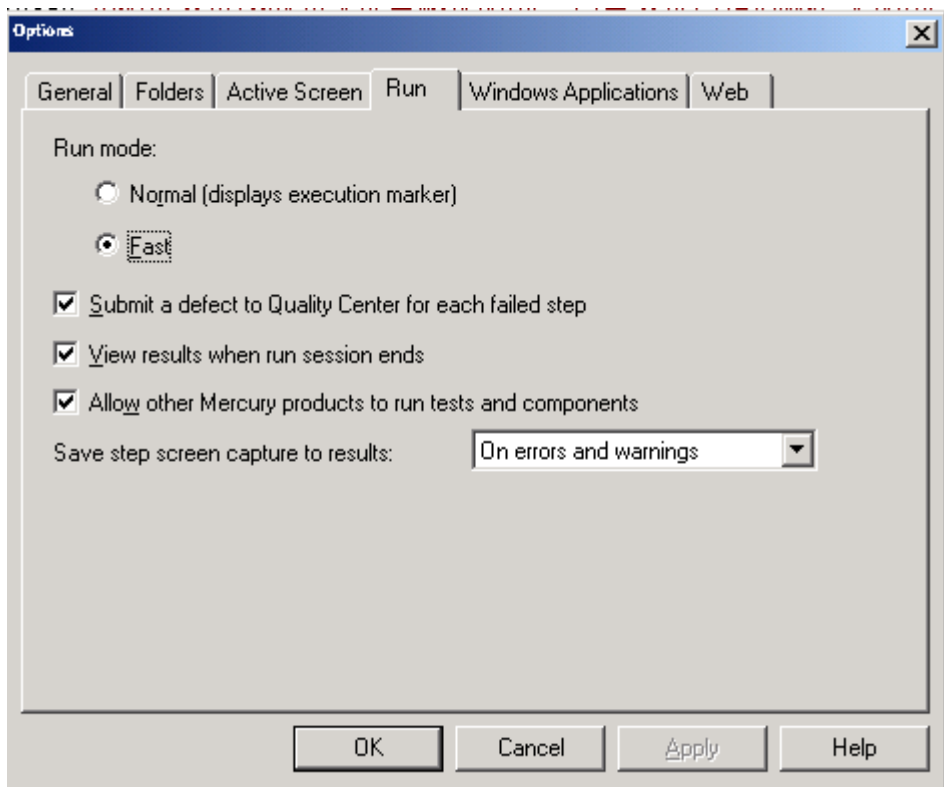


图 1.2.D1.1

在 windows 系统主界面, 点击开始—程序—Quick Test Professional—test result viewer, 打开 test result viewer 如下图

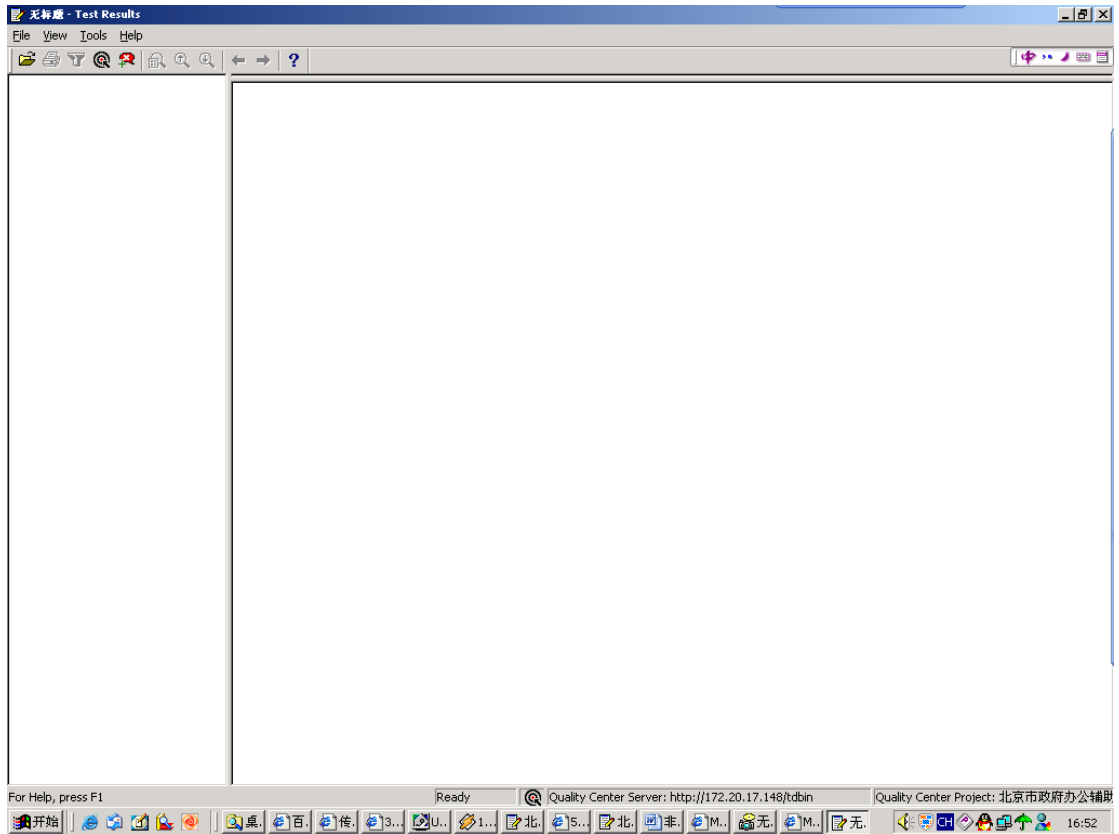


图 1.2.D1.2

点击 tool—quality center connection, 弹出连接对话框如下图

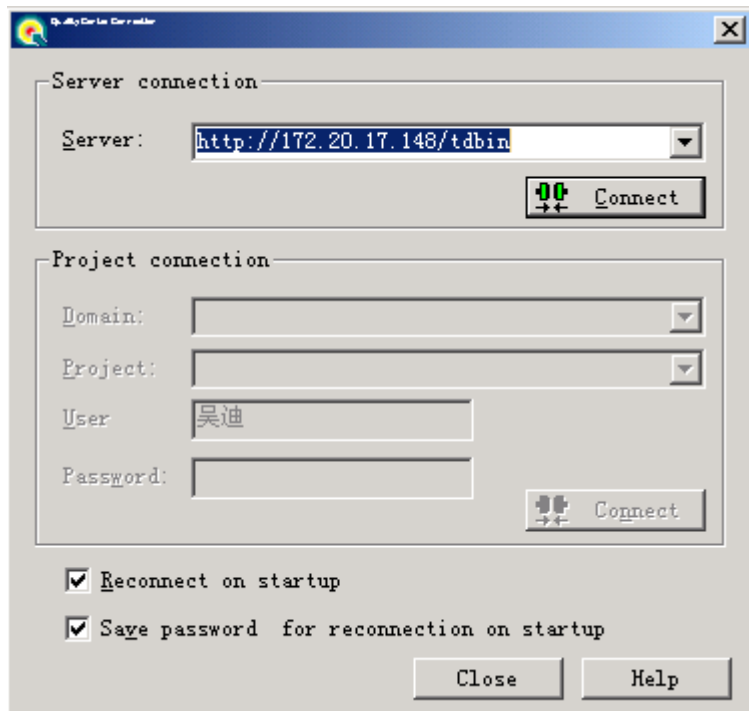


图 1.2.D1.3

Server: 与 QTP 输入的服务器地址一致

Domain: 与 QTP 输入的服务器地址一致

User: 与 QTP 输入的服务器地址一致

Password: 与 QTP 输入的服务器地址一致

Reconnect on startup 在下次启动 QTP 时自动登陆到你的项目中

Save password for reconnection on startup 为下次登陆保存密码

D2 在 TD 中运行 QTP 脚本，检查点检查到的错误信息将被保存在 TD 的 defect 中。详细见 TD 使用手册

E 个人感受:无

1.3 本机利用 QTP 创建,修改与提交脚本

A 问题描述:

当 QTP, test result viewer 与 TD 连接成功后,就可以通过 TD 调用 QTP 来录制脚本,录制脚本完成后,脚本将保存到你的 TD 服务器中,当你修改脚本时,需要通过 TD 调用出你所需要修改的脚本,修改完成后,脚本继续保存在 TD 服务器中

B 处理流程:

B1 在 TD 中建立 QTP 测试用例

B2 调用 QTP

B3 录制脚本

B4 保存脚本

B5 在 TD 中查看脚本

B6 修改脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 在 TD 中建立 QTP 测试用例,点击 new test 并且选择 QTP 测试用例,如下图

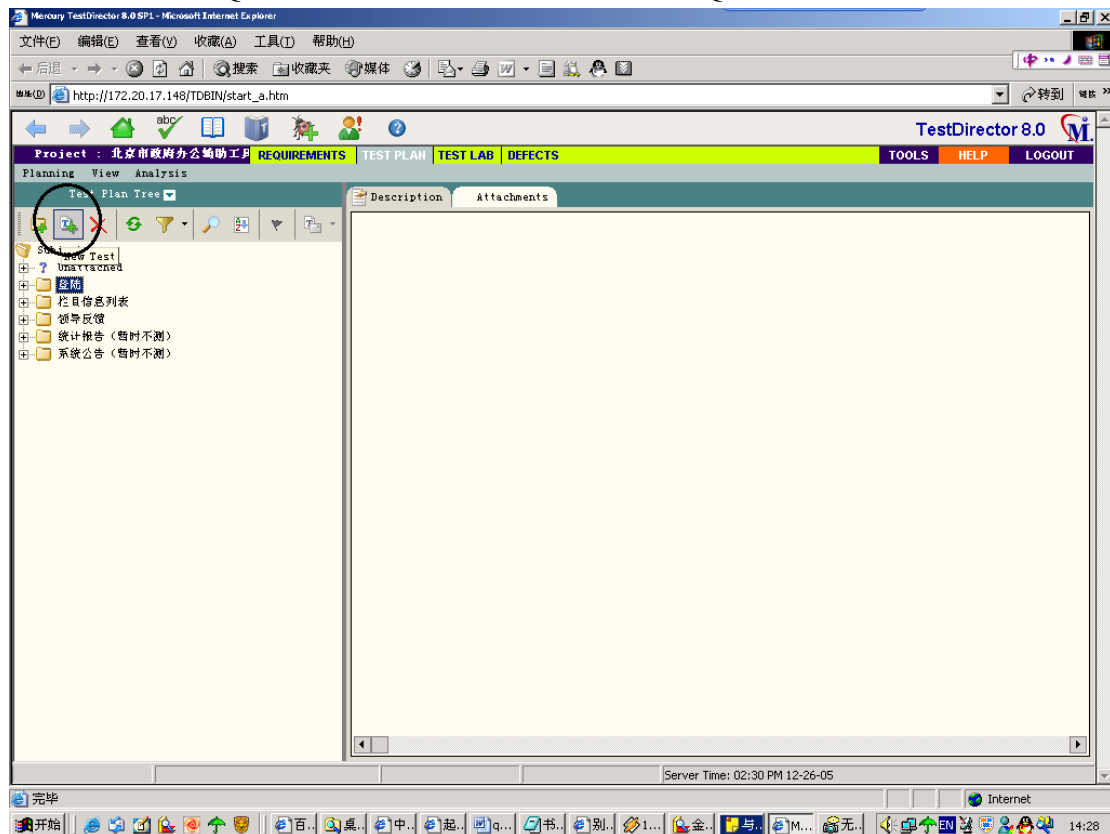


图 1.3.D1.1

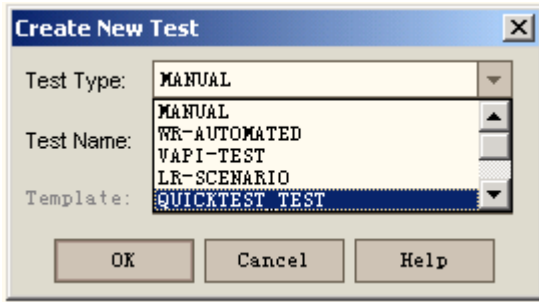


图 1.3.D1.2

D2 调用 QTP, 开启 QTP

脚本建立成功后, 点击 test scrip, 并且点击其中的 lanch 键如下图

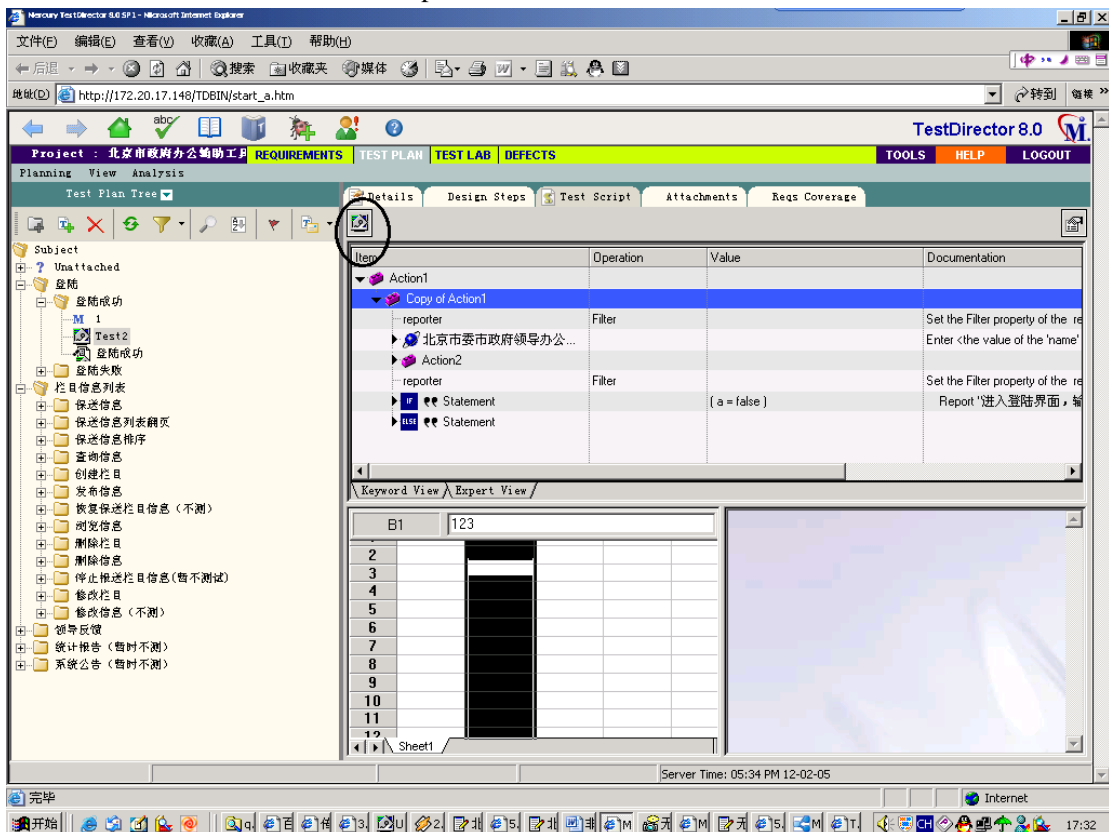


图 1.3.D2.3

D3 录制脚本, 详细见 QTP 说明书

D4 保存脚本, 当你的脚本录制完成后, 点击 save, 出现如下对话框

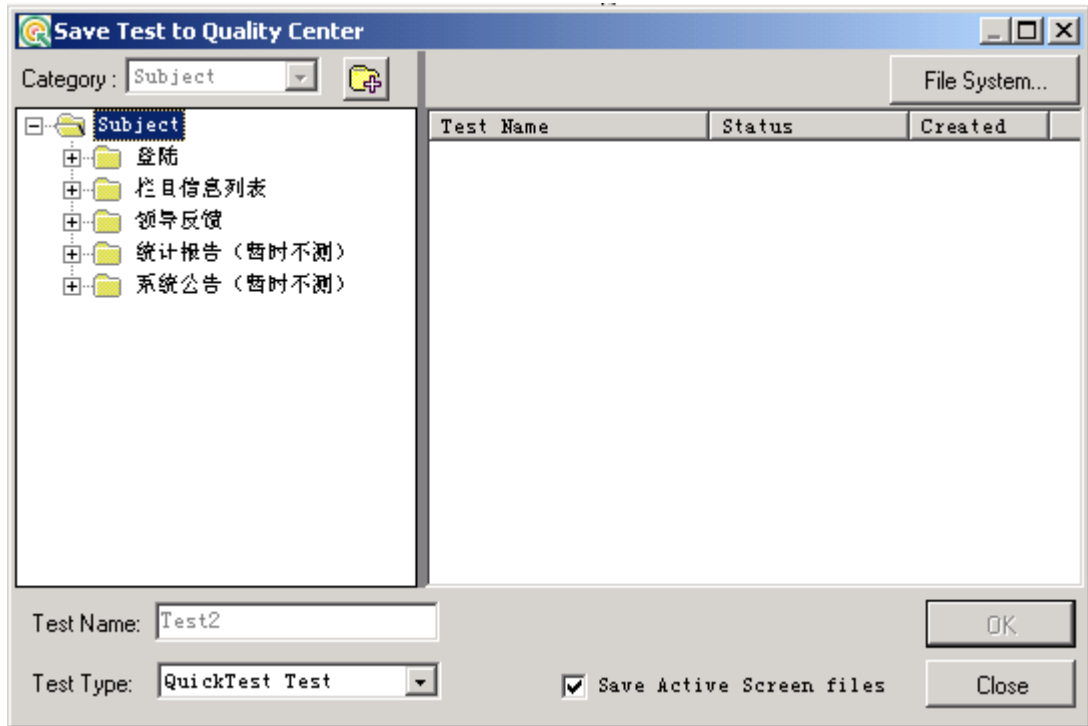


图 1.3.D4.4

左边的文件树是你的 test plan,在 test name 中输入测试名称,点击保存,你的脚本将保存到 test plan 下,你也可以点击右上角的 file system...将脚本保存到本机上

Save active screen files,选中这个之后回将你 QTP 中保存的 active screen 随着脚本一起保存到 TD 服务器中

D5 在 TD 中查看脚本,在 TD 的 test plan 中查看脚本,如下图



图 1.3.D5.5

选中一个测试脚本,点击右边的 test scrip,能够看到详细的步骤,如下图

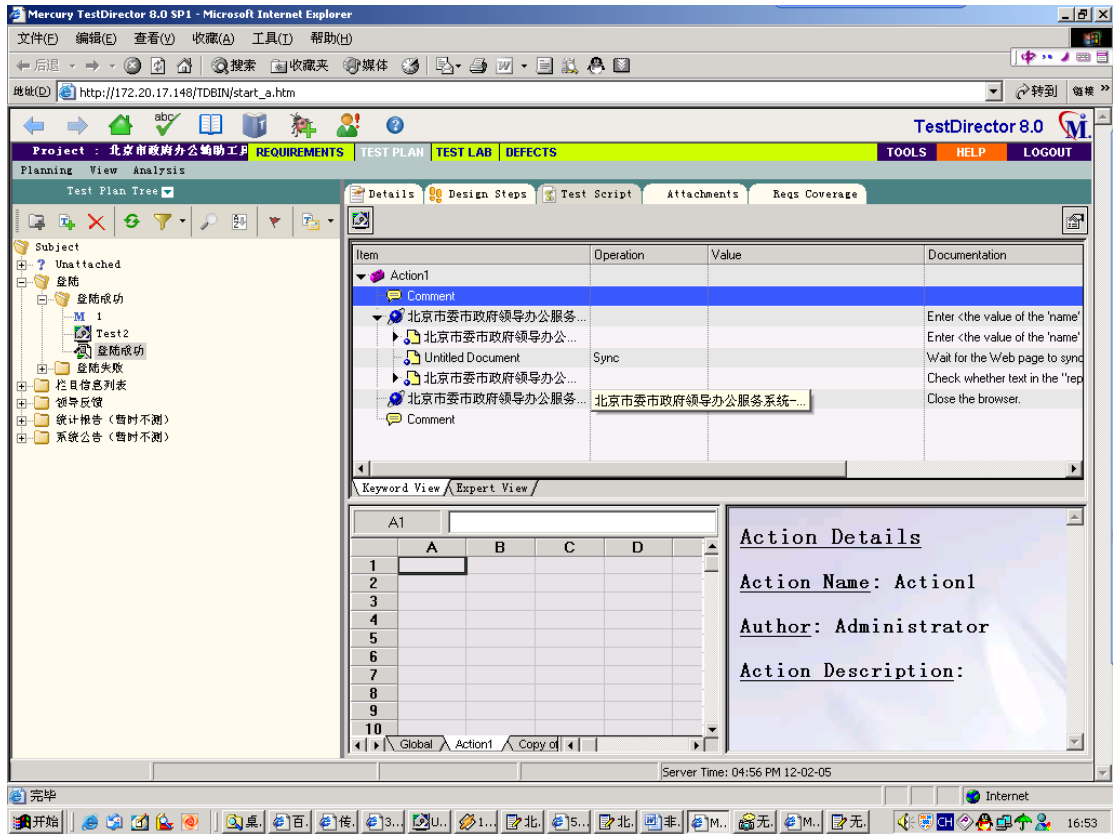


图 1.3.D5.6

D6 修改脚本，如上操作打开 QTP，打开后，修改你的脚本，修改完成后，点击 save 新脚本自动保存到 TD 服务器中

E 个人感受:无

1.4 QTP fail 自动截图功能的使用

A 问题描述:

在测试过程中,常常出现随机性错误,很难重现,但并不意味着错误不存在,这时候与开发人员交流起来就会有问题,因为开发人员不能准确的知道你所要展现的错误,自动截图功能,能在检查点失败的时候,截取当时的页面,通过这些图片,能让开发人员更加直观的了解错误信息,改正错误

B 处理流程:

B1 设置 QTP

B2 在 TD 中运行 QTP 脚本

B3 查看截图

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 设置 QTP, 进入 QTP 操作界面,点击 tools-option 中的 run tab 如下图

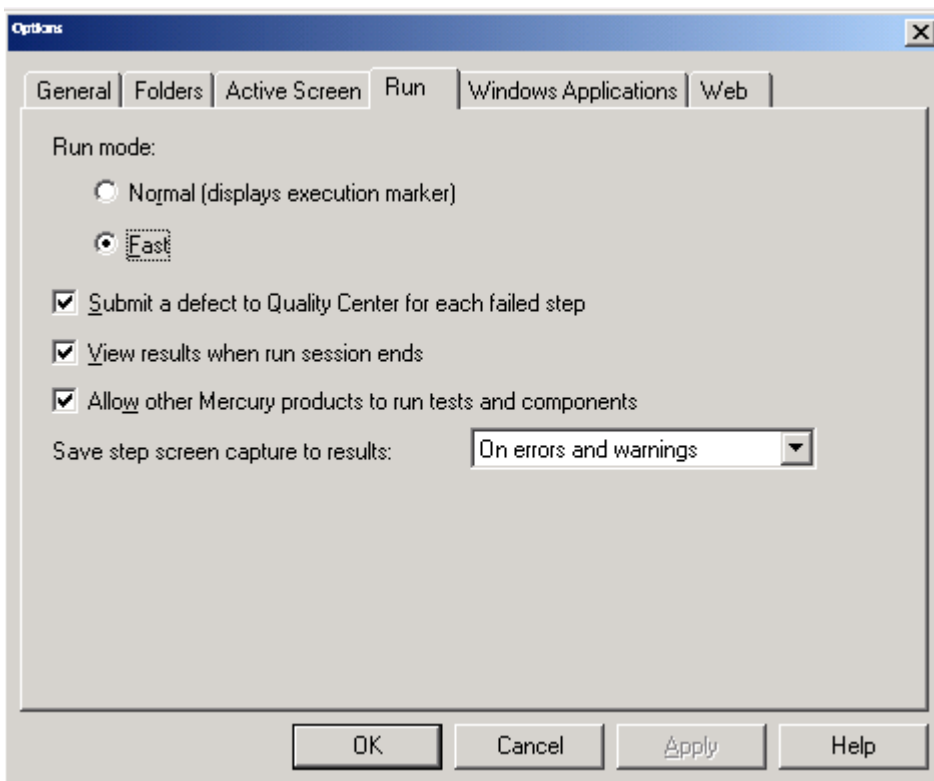


图 1.4.D1.1

其中 save step screen capture to results 是控制向 QTP 提交图片的选项,其中包括 Always 保存每步的界面

On errors 在发生 errors 的时候,当检查点不通过的时候保存图片

On errors and warnings 在发生 errors 和 warinings 时,保存图片

Never 不保存图片

D2 在 TD 中运行脚本,(详见 1.3)

D3 查看截图,所截的图,并不能以附件的形式出现在 TD 的 defect 中,而是出现在 test tab 中,双击的你的自动化测试

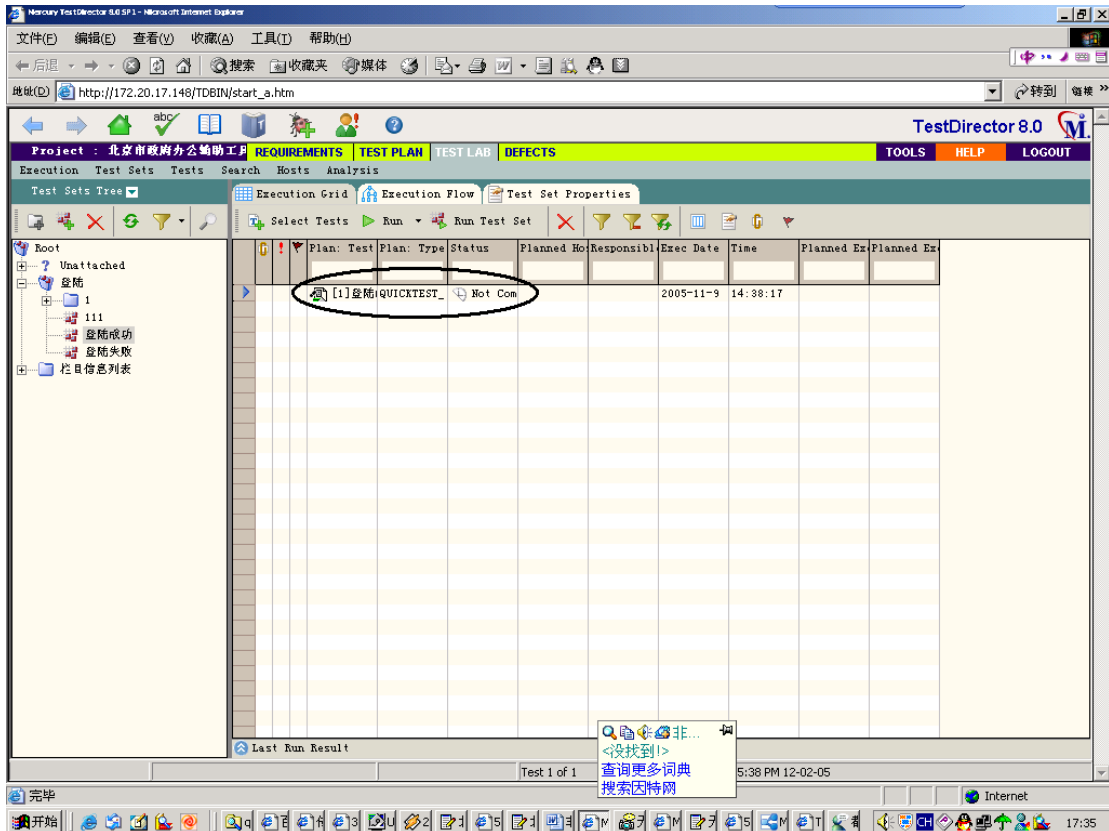


图 1.4.D3.2

出现如下图界面,选中一条 fail,通过点击其中的 launch 键来调出 QTP 自带的错误报告

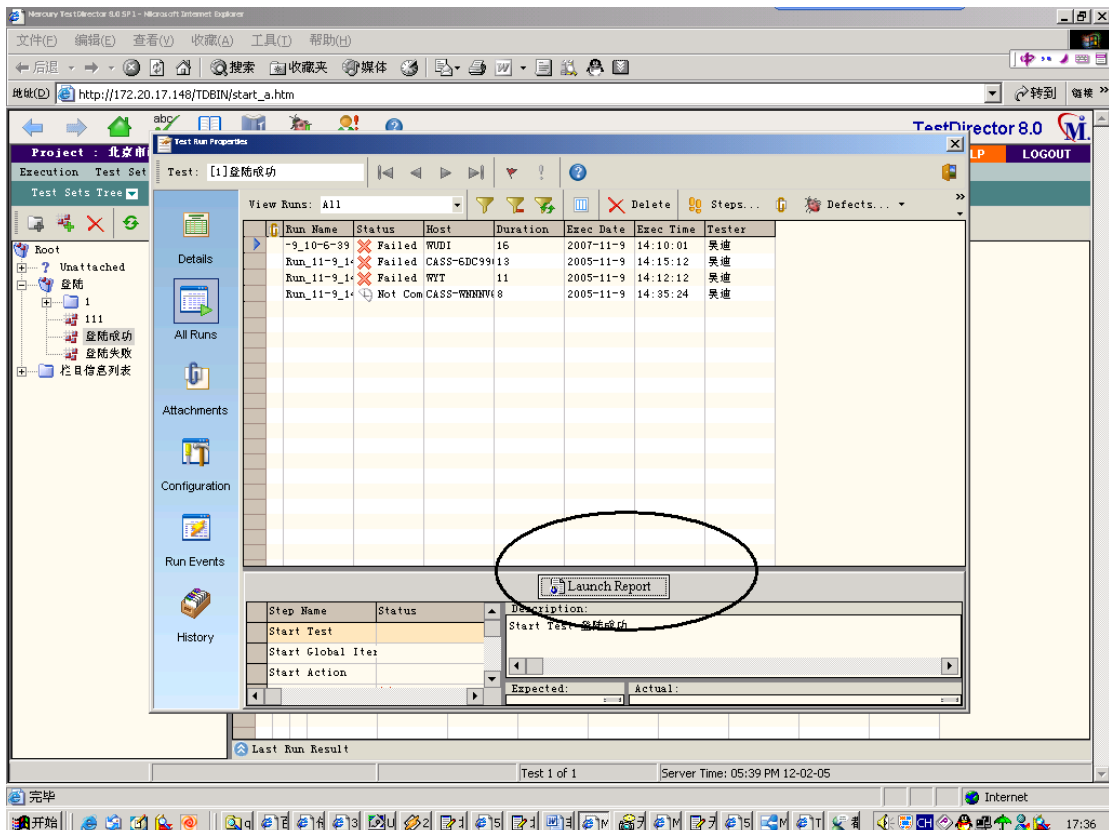


图 1.4.D3.3

通过这个来查看截图

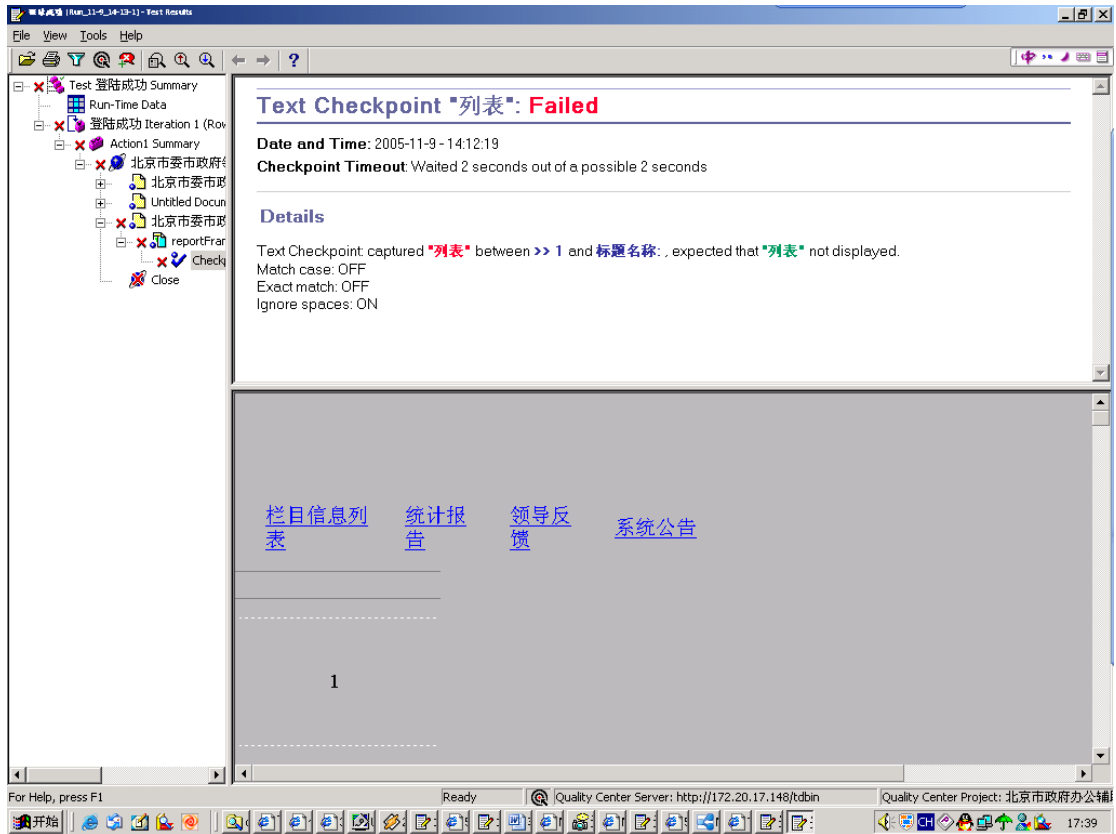


图 1.4.D3.4

E 个人感受:

这里的图不能以附件的形式保存到 defect 中,需要知道错误是哪个计划中的哪个用例执行之后的结果.

1.5 上网助手等插件对 QTP 的影响

A 问题描述:

在运行 QTP 时,由于 QTP 对浏览器的反映速度比较高,一些影响浏览器浏览速度的插件将导致 QTP 无法正常运行

B 处理流程:

B1 删除插件

B2 运行 QTP

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 检查你的浏览器上有什么非默认的插件(建议使用浏览器的最初版,不要安装任何插件),删除这些插件

D2 运行 QTP(详见 QTP 使用手册)

E 个人感受:插件对 QTP 的执行影响比较大,很多不明错误都是不明插件引起的

1.6 防火墙对 TD 与 QTP 的连接的影响(瑞星 2005 下)

A 问题描述:

TD 与 QTP 的连接是基于网络的,需要通过防火墙设置允许 TD 调用 QTP 的插件,否则将无法正常使用

B 处理流程:

B1 防火墙自动识别程序

B2 手工识别

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 防火墙自动识别程序,当 TD 调用 QTP 时,会自动启动一个调用程序,通常情况下,防火墙会自动识别出来,并且问用户是否允许通过,选择允许通过

D2 手工识别,如果防火墙没有自动识别调用程序就需要手工识别这个程序,手工需要添加的程序如下图



图 1.6.D2.1

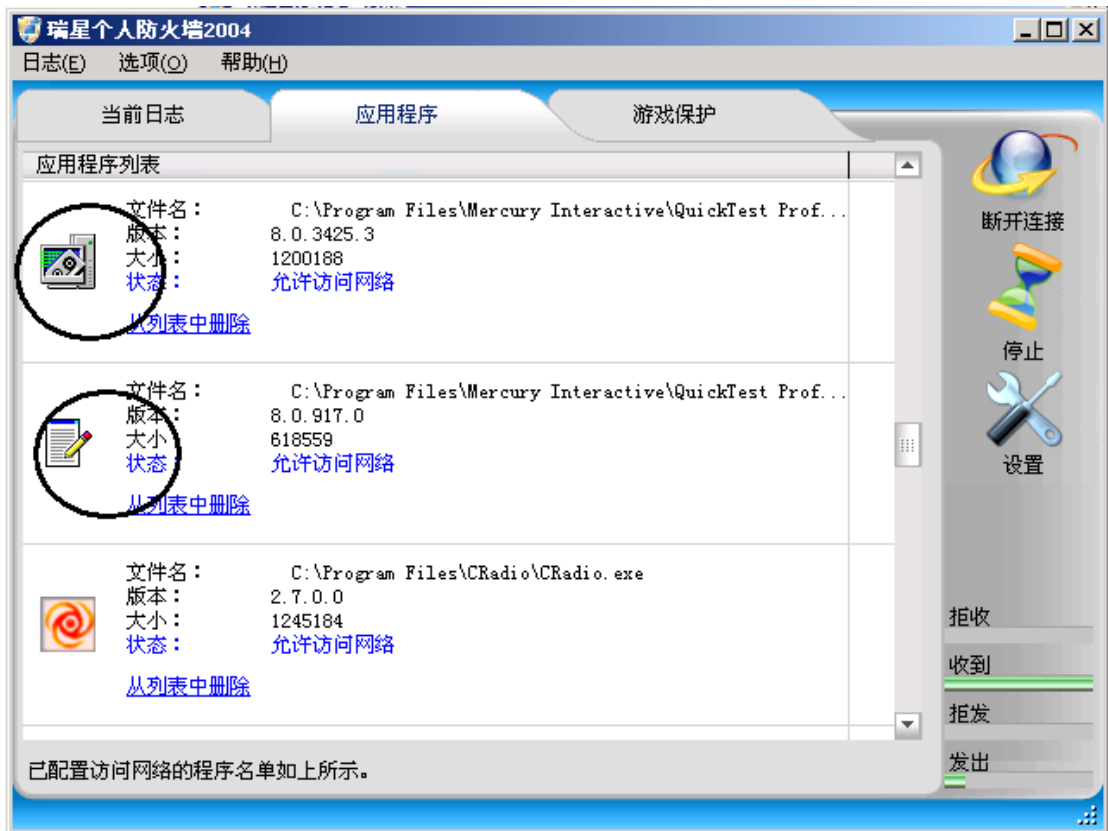


图 1.6.D2.2

E 个人感受:

需要防火墙通过验证,不然无法正确执行

1.7 如何使用 QTP 进行定时测试

A 问题描述:

当使用 TD 连接 QTP 后,就可以通过 TD 来对 QTP 的运行进行管理,控制 QTP 何时运行

B 处理流程:

B1 在 TD 中设定时间

B2 QTP 定时运行

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 在 TD 中设定时间,在 test lab 中添加一个控件,如下图

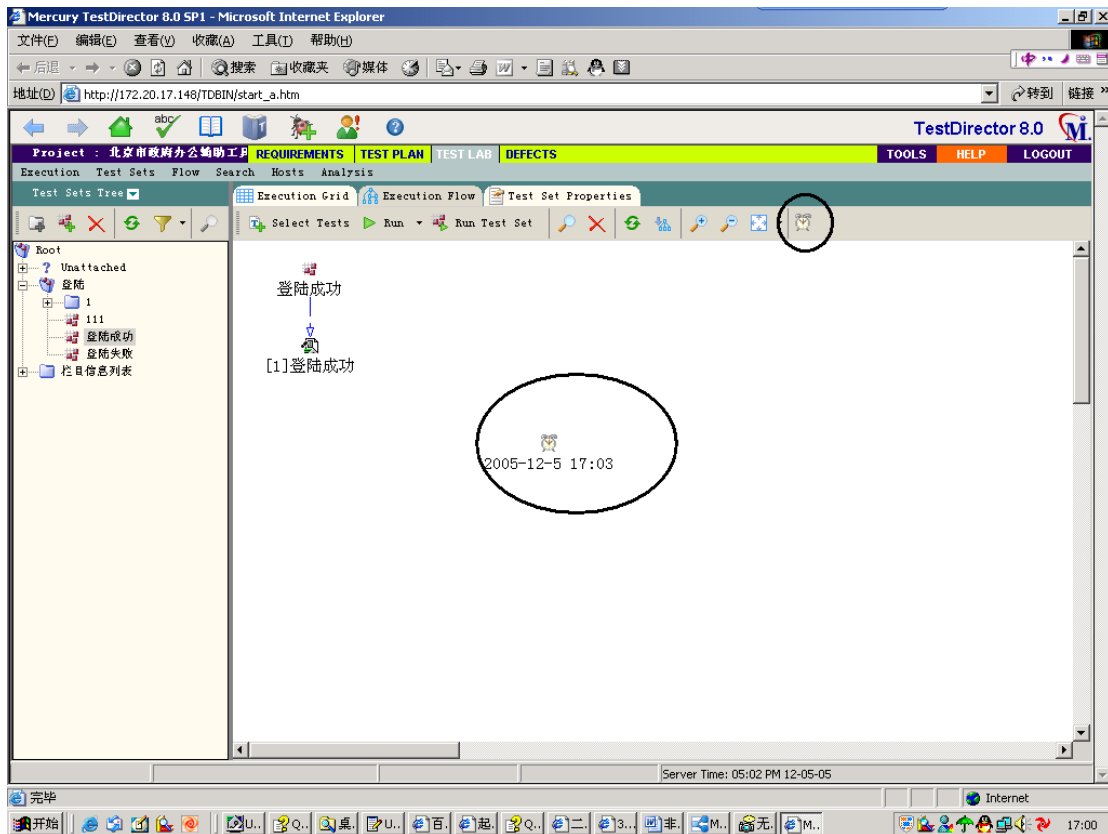


图 1.6.D1.1

选中这个新的控件,并且点击他的中心,拖动鼠标,让他发出的连线与你的测试相连如下图

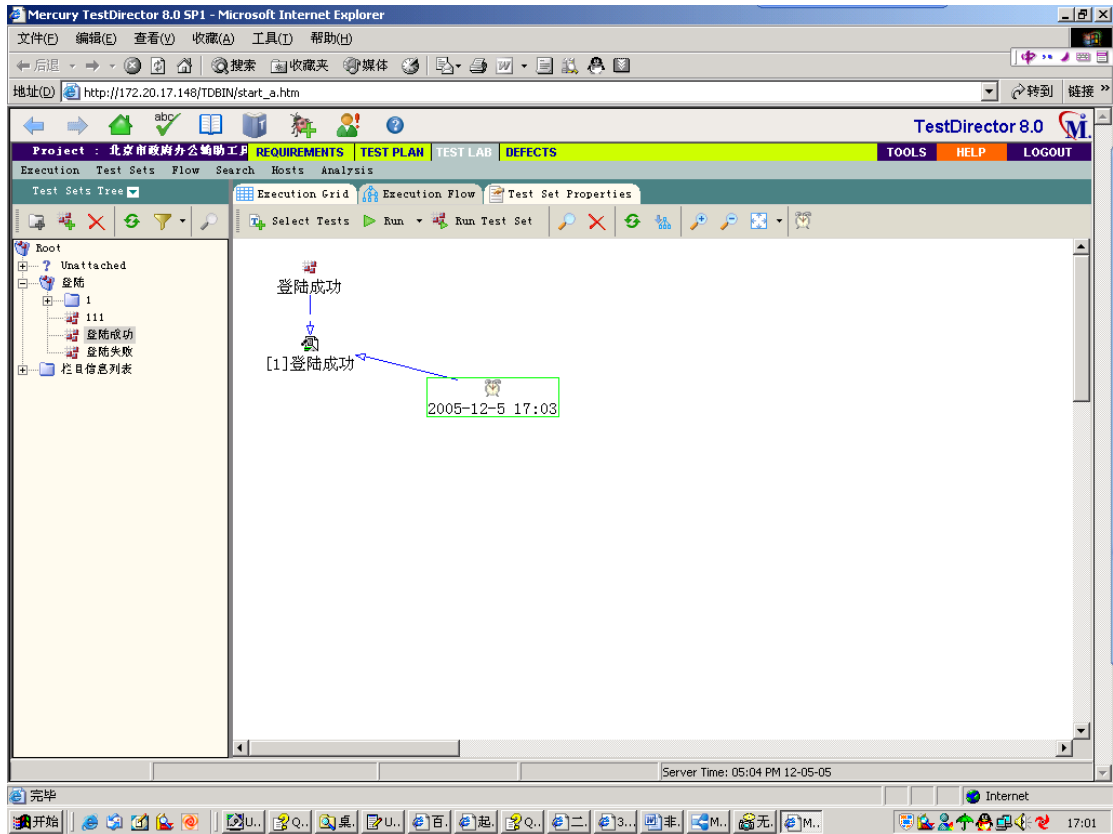


图 1.6.D1.2

你可以通过单击右键选择 time dependency 来设定时间

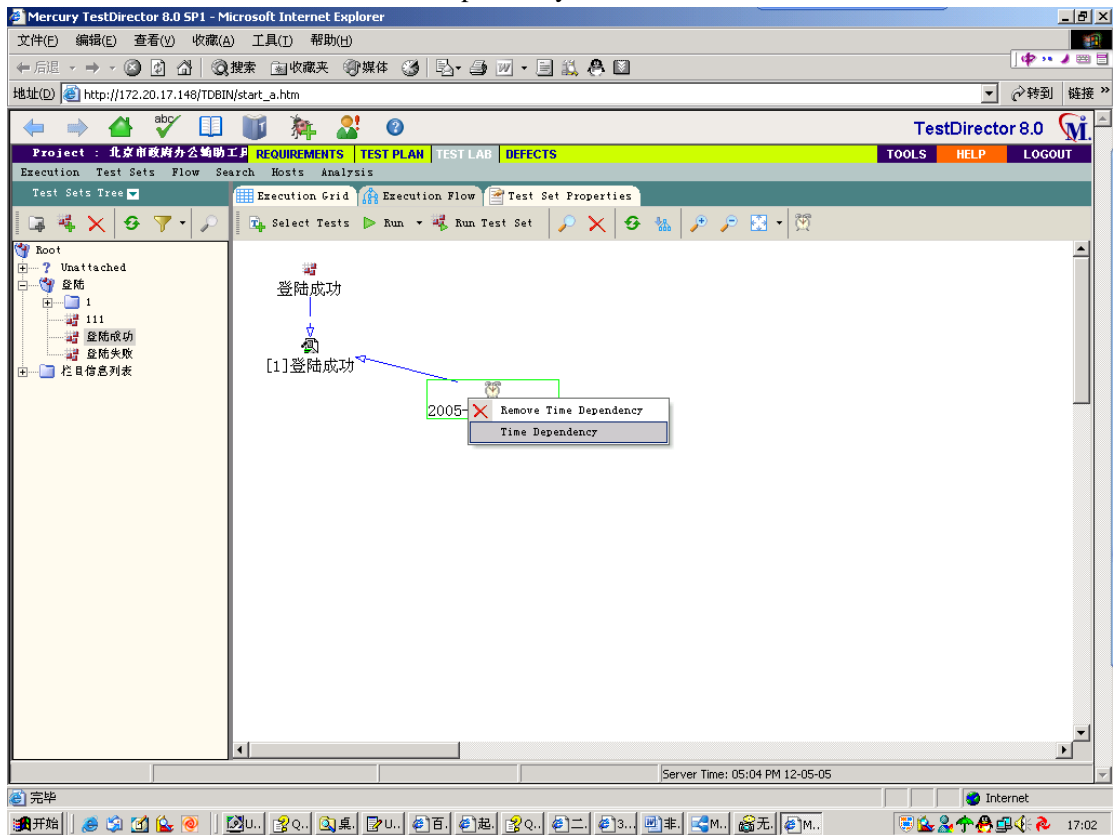


图 1.6.D1.3

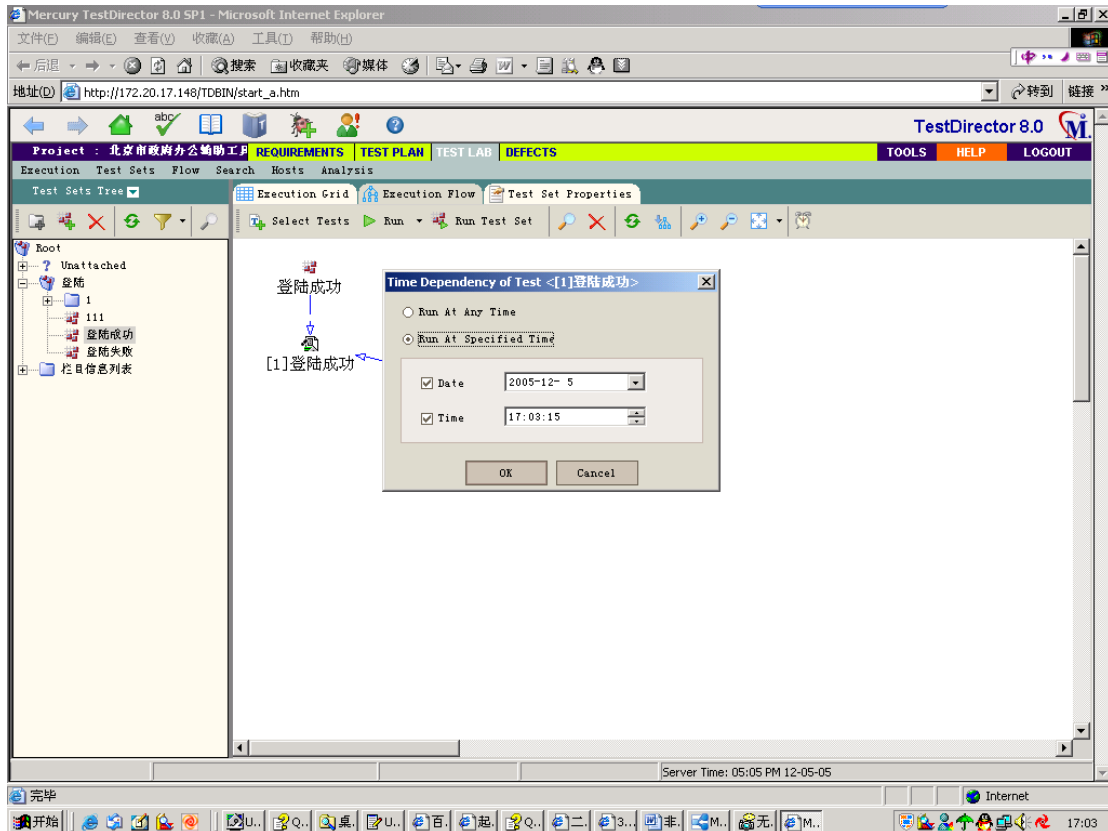


图 1.6.D1.4

D2 QTP 定时运行当设定完毕后,点击 run 键,TD 会弹出运行界面,点击 run QTP 将等待,直到你设定的时间后才开始执行测试脚本

E 个人感受:无

2 QTP 的录制

2.1 QTP 录制应先作录制计划

A 问题描述

在录制脚本时,通常会遇到一些特殊的情况,比如在编辑脚本时无法添加文字检查点等问题

B 解决办法

在录制脚本之前作录制计划,不一定要详细的计划,只是在头脑中要清晰你要录制什么流程,设立什么检查点,在哪设立检查点.避免不作计划先录制脚本,然后通过编辑的方法来完善脚本,有的时候由于录制级别设置的问题,有的界面可能无法被保存下来,那么你就无法在其上使用任何检查点,边录制边添加检查点的方法就可以避免这个问题.

C 个人感受: 无

2.2 QTP 的录制方法

A 问题描述

在学会使用 QTP 之后需要掌握其正确的录制方法才能达到事半功倍的效果,往往方法不对会导致运行结果与预期不一致,现以录制登陆为例

B 解决办法

以登陆为例,登陆一般分为两个脚本录制,分别是合法登陆,非法登陆

合法登陆:使用正确得用户名,密码,登陆系统,检查点检查登陆到系统中

非法登陆:使用错误的用户茗,密码,登陆系统,检查点检查因没有登陆进系统,而弹出的提示信息

C 个人感受: 无

2.3 关于下拉选择数据输入框的更改

A 问题描述:

在 QTP 录制脚本的过程中经常会遇到下拉选择数据输入框的问题和录制后会遇到需要通过参数化来使脚本的重用性提高的问题,你可以通过修改下拉选择数据输入框的属性的值的方法完成参数化

B 处理流程:

B1 录制脚本

B2 参数化下拉菜单

B3 运行脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server

相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具，主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,录制一个关于下拉选择数据输入框的脚本如下图

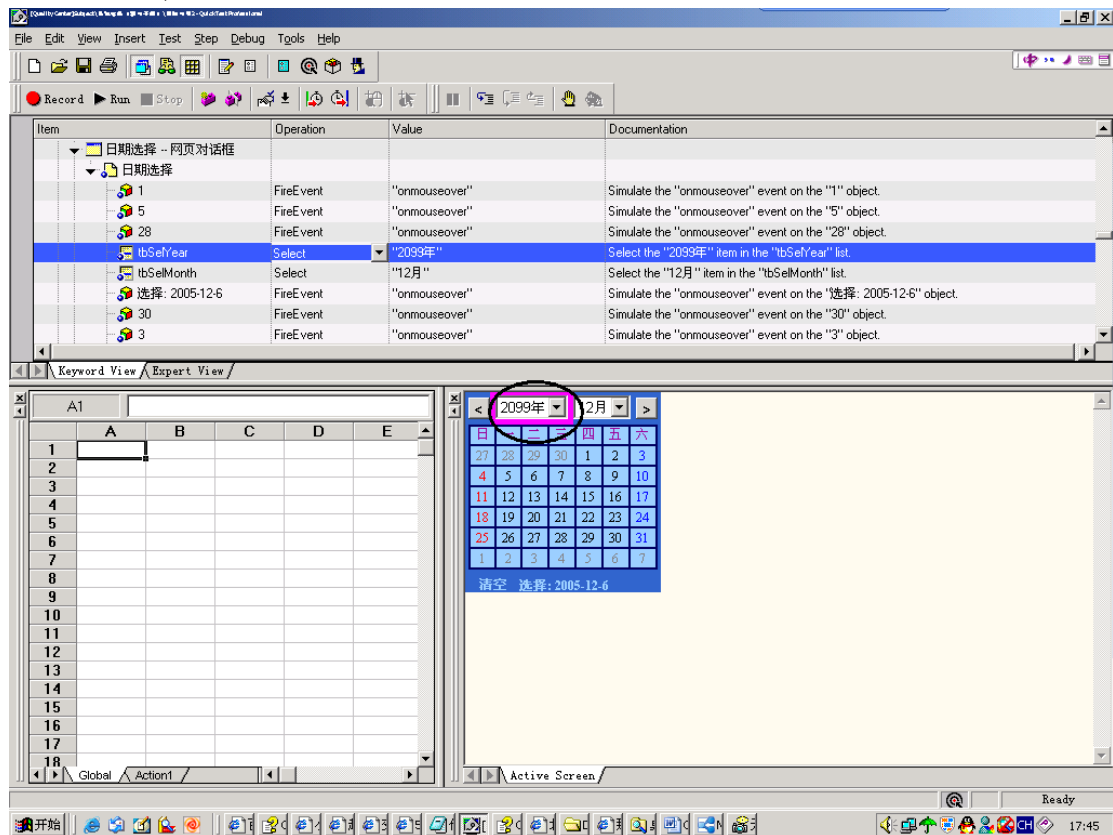


图 2.3.D1.1

D2 参数化下拉选择数据输入框,展开你的脚本,找到有下拉菜单的那个步骤,直接在他的值上作更改或者参数化即可

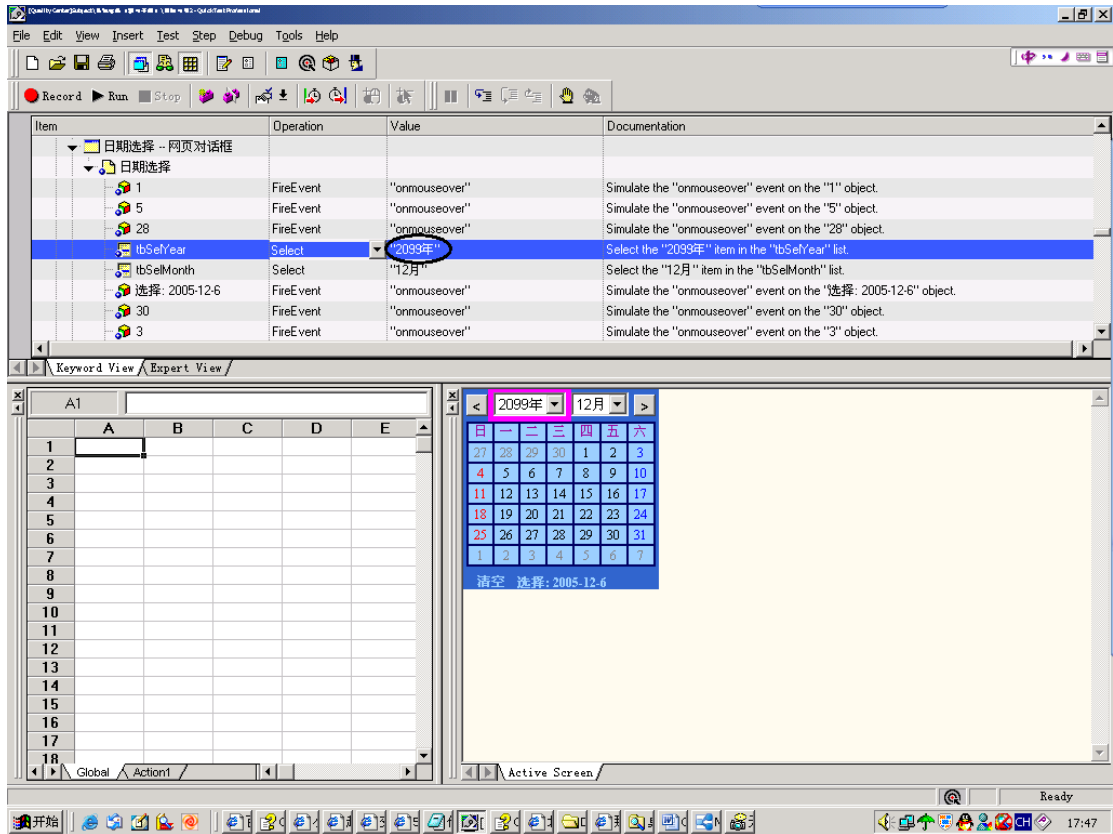


图 2.3.D2.2

根据你更改的数值,QTP 会在下次回放时自动找到这个数值

关于多级下拉选择数据输入框的问题,使用同样的方法 QTP 会自动识别二级菜单你所输入的内容

D3 运行脚本

E 个人感受:无

2.4 中文输入法对于回放的影响

A 问题描述

由于录制和回放时输入法设定不一致,使得回放时,QTP 不能识别某些输入法的特定字符,导致脚本无法正常运行

B 解决办法

保证录制脚本的环境与回放时的环境相同,避免产生类似的问题,这个问题可以扩展到其他方面,很多情况下,都是由于回放环境与录制脚本环境不一致导致脚本无法正常运行,输入法只是一个例子

C 个人感受: 无

2.5 如何修改系统时间

A 问题描述:

使用 QTP 连续修改系统时间

B 解决办法:

通过编码实现

建立一个bat文件, 输入

```
time %1
```

QTP中:

```
SystemUtil.Run "c:\settime.bat","15:30:00"
```

"15:30:00": 为需要设置的系统时间

关于录制这部分,因为每个人遇到的问题不一样,所以没法写的很多,只能把大体的录制思想和主要的录制技巧写一下,如果有更多问题,还需要继续交流

C 个人感受:无

3 QTP 的自定义操作

3.1 手工添加步骤的方法

A 问题描述:

在录制脚本的过程中经常会遇到 QTP 没有按照要求录制其中一个步骤的情况,如果出现这种情况就重新录制,会浪费很多时间,我们可以通过手工添加步骤的方式,来添加步骤

B 处理流程:

B1 录制脚本

B2 添加缺少的步骤

B3 运行脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,以登陆系统为例,在登陆系统时需要输入用户名,密码,有时 QTP 会遗失其中的一个步骤,导致无法正常回放,如下图缺少密码输入的步骤

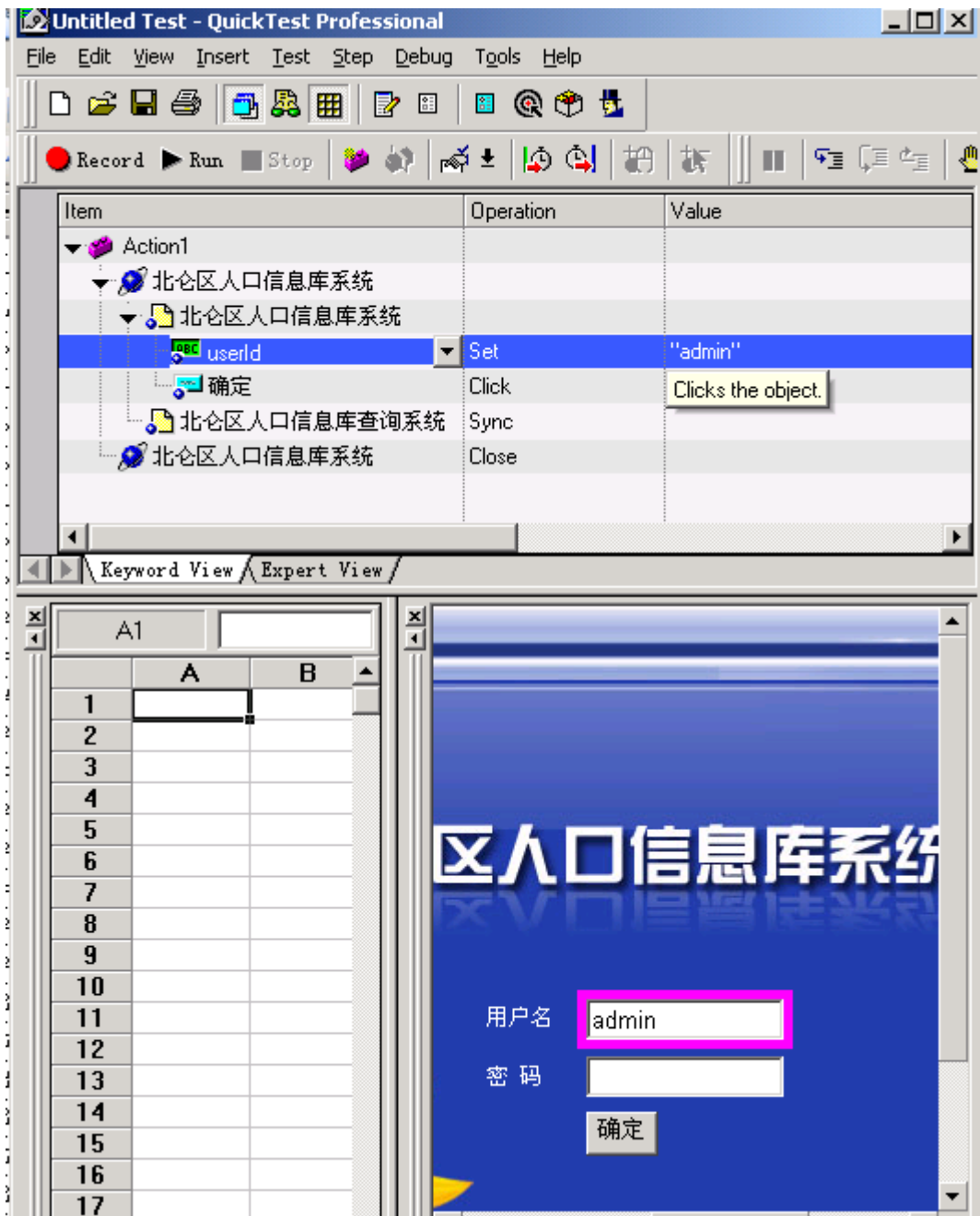


图 3.1.D1.1

D2 添加缺少的步骤,选中输入用户名这个步骤,点击 insert-newstep

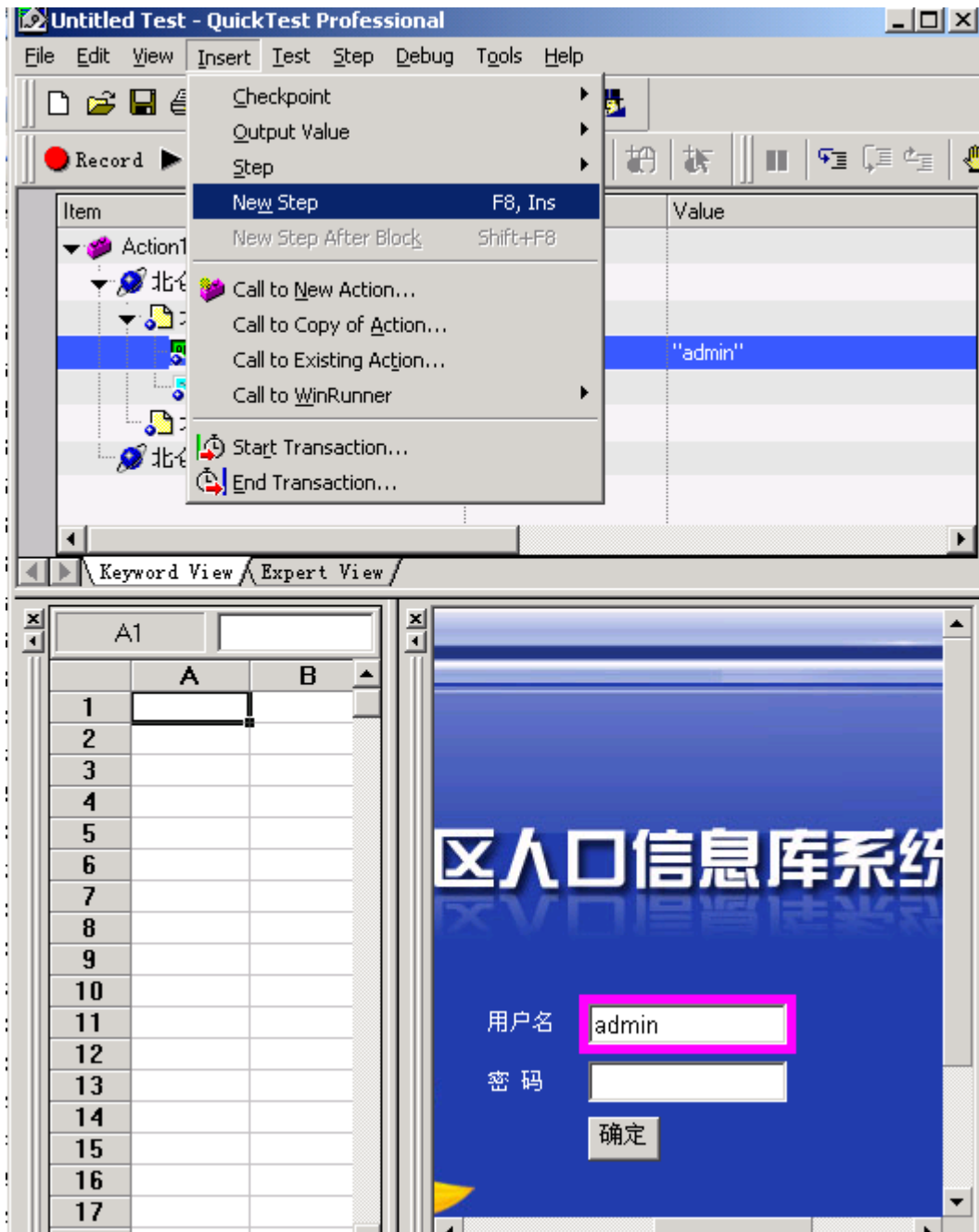


图 3.1.D2.2

出现如下界面

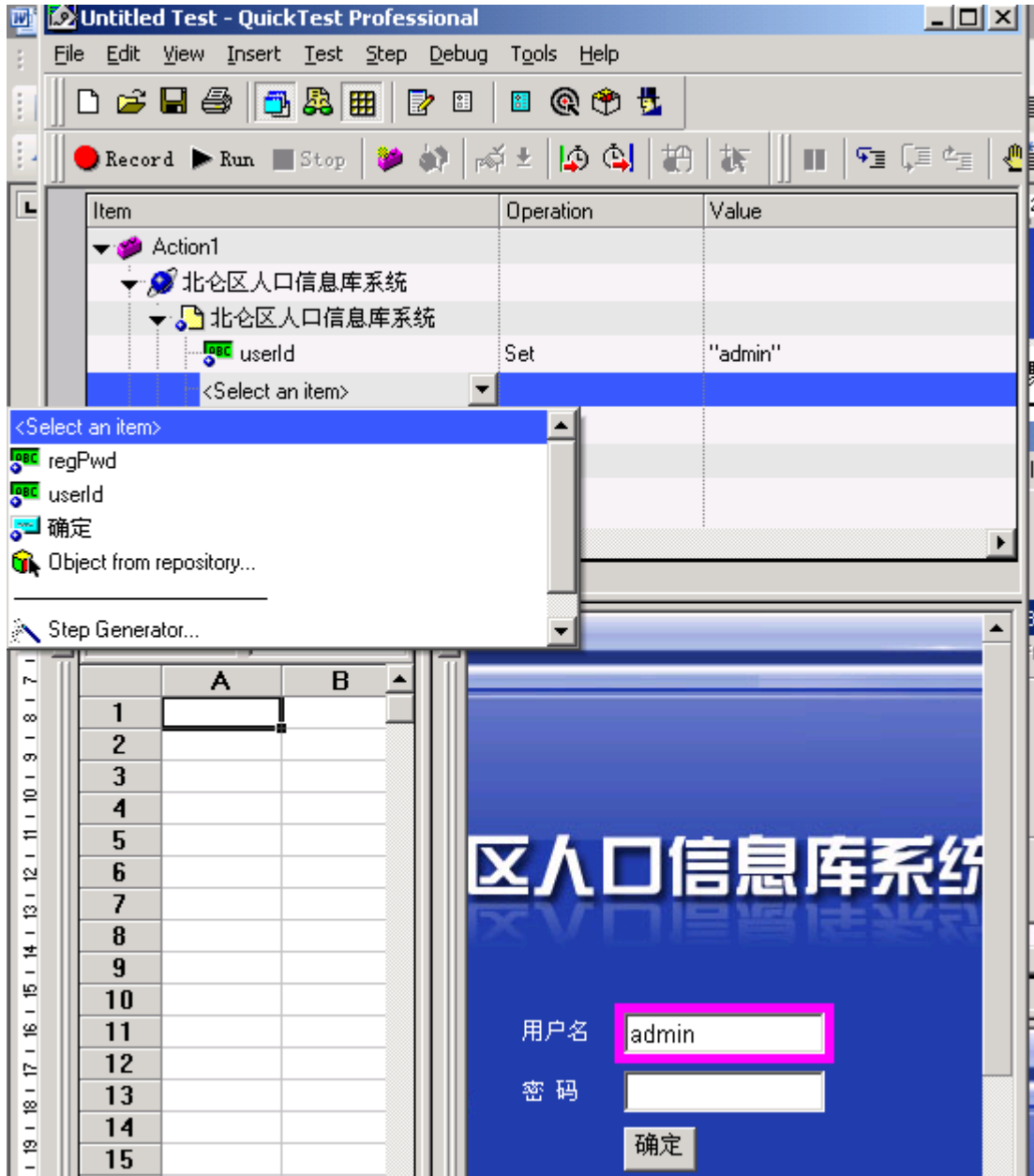


图 3.1.D2.3

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,弹出如下对话框

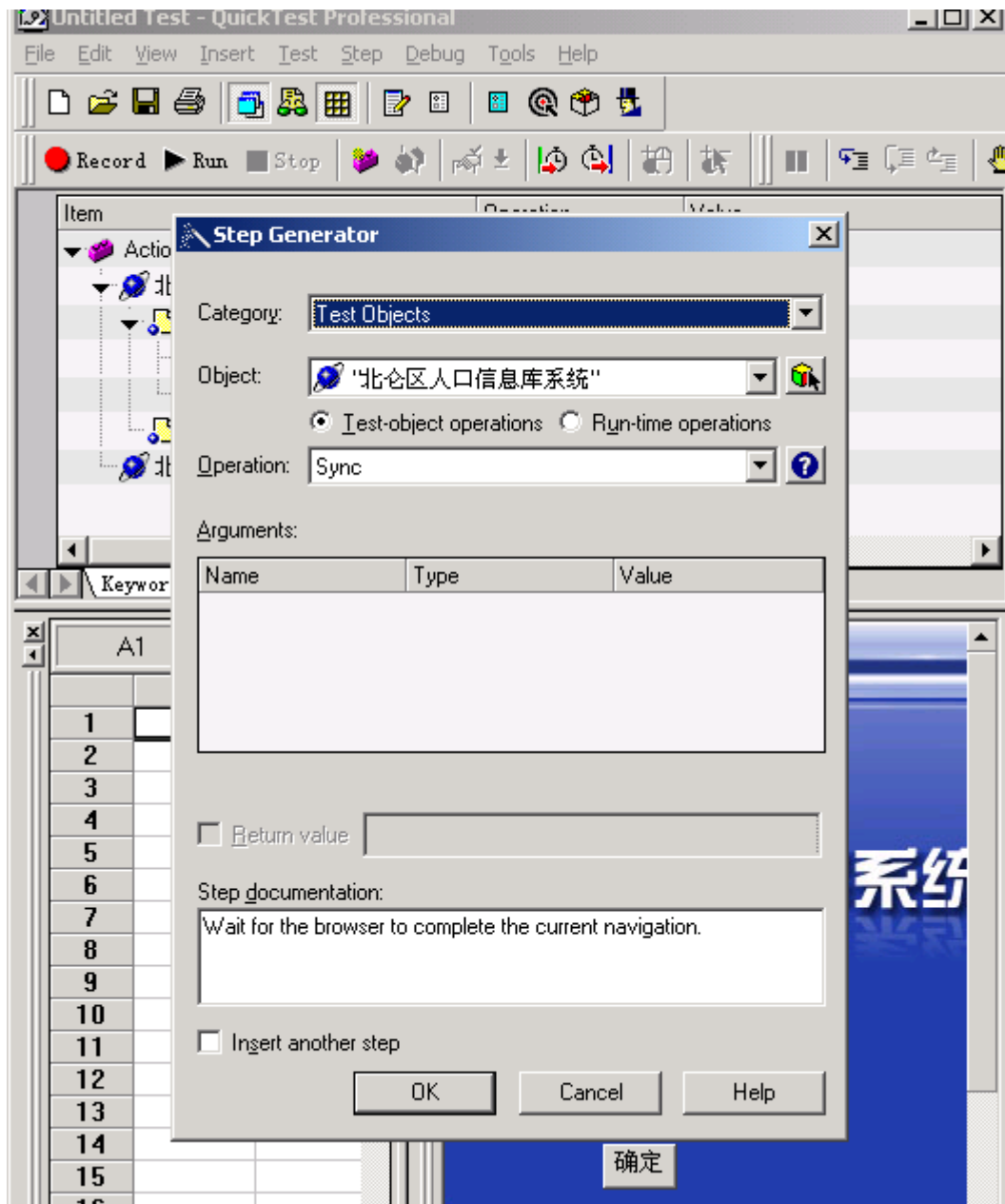


图 3.1.D2.4

点击如下图所示的按钮

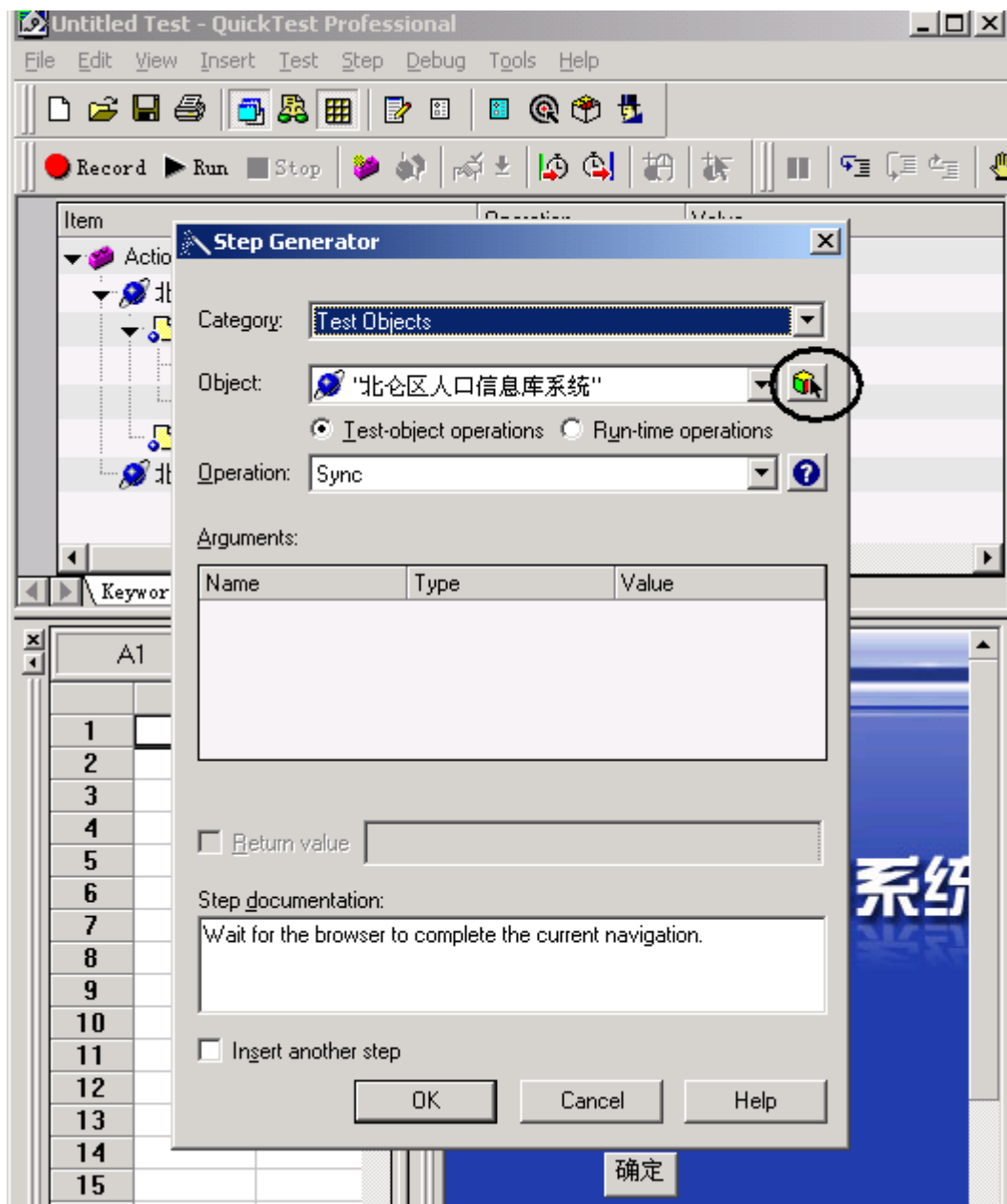


图 3.1.D2.5

弹出如下对话框

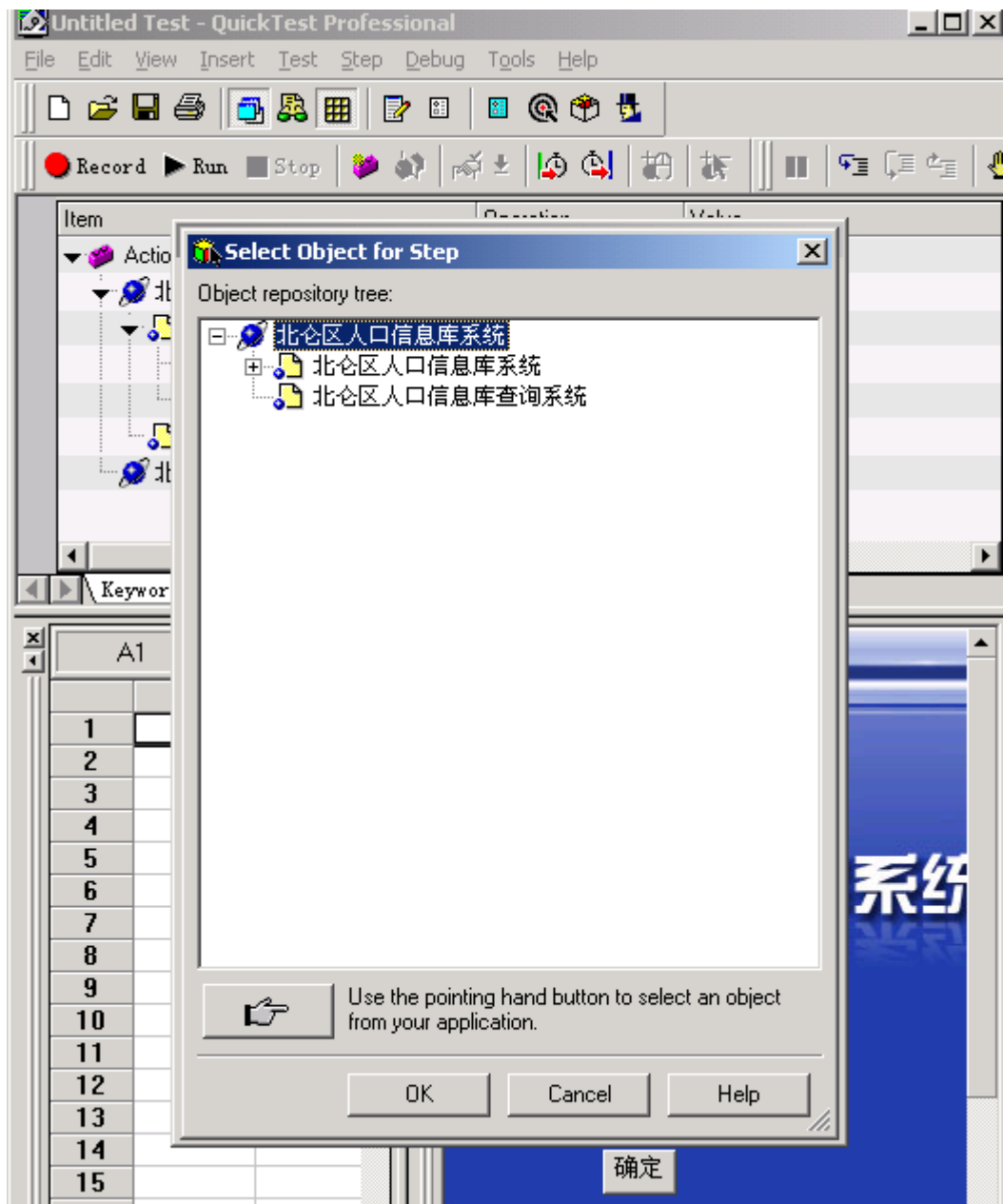


图 3.1.D2.5

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,



图 3.1.D2.6

弹出如下图对话框

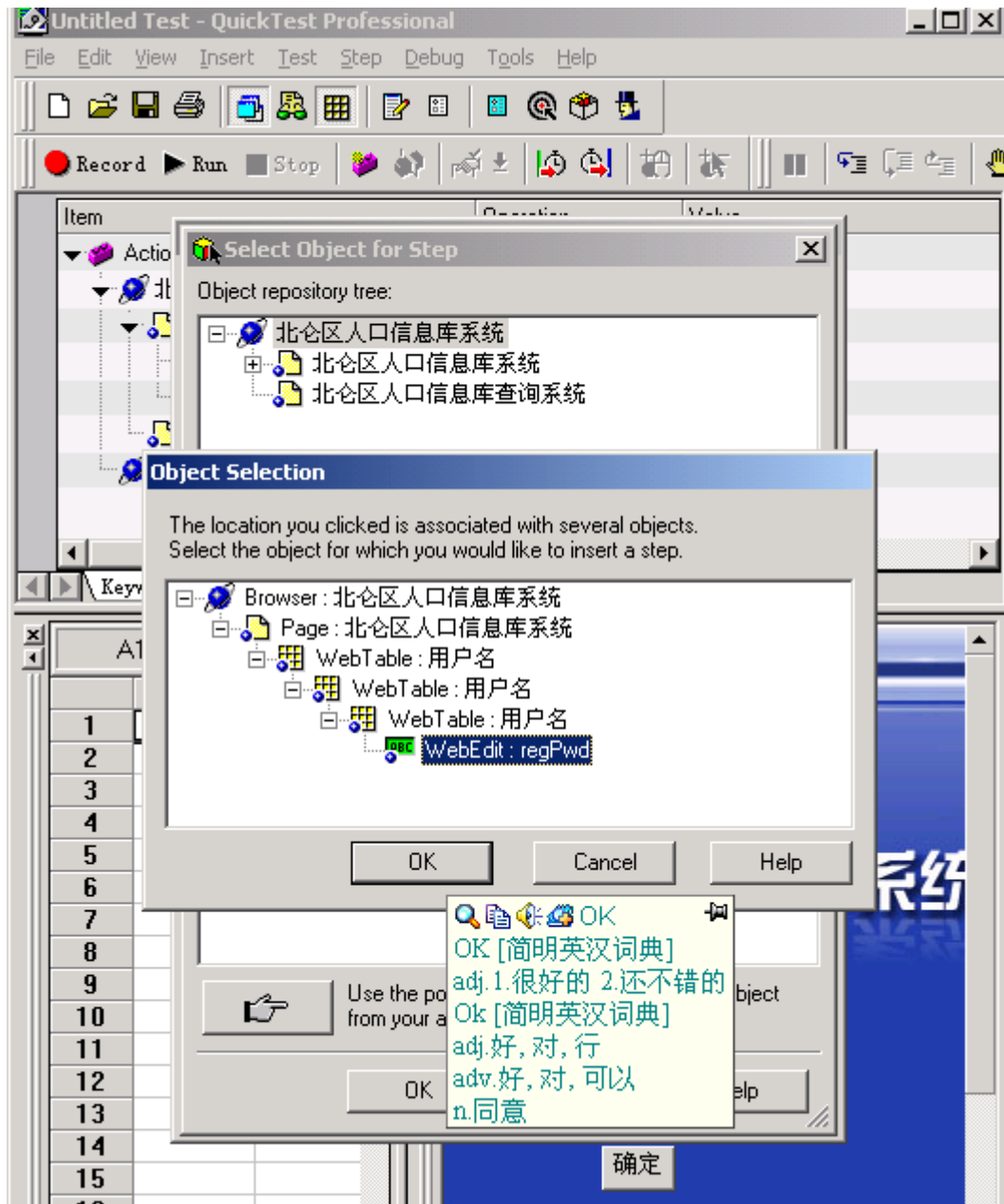


图 3.1.D2.7

点击 ok,所选对象的属性将被保存到如下对话框中

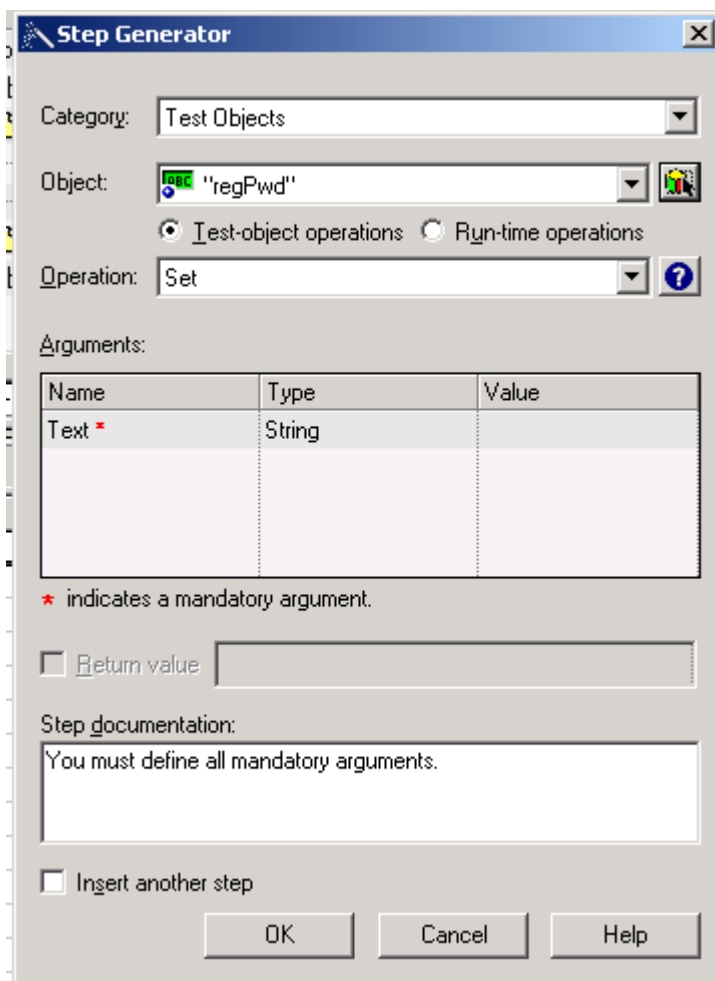


图 3.1.D2.8

对于需要输入值的控件,需要在如下图位置输入一个值

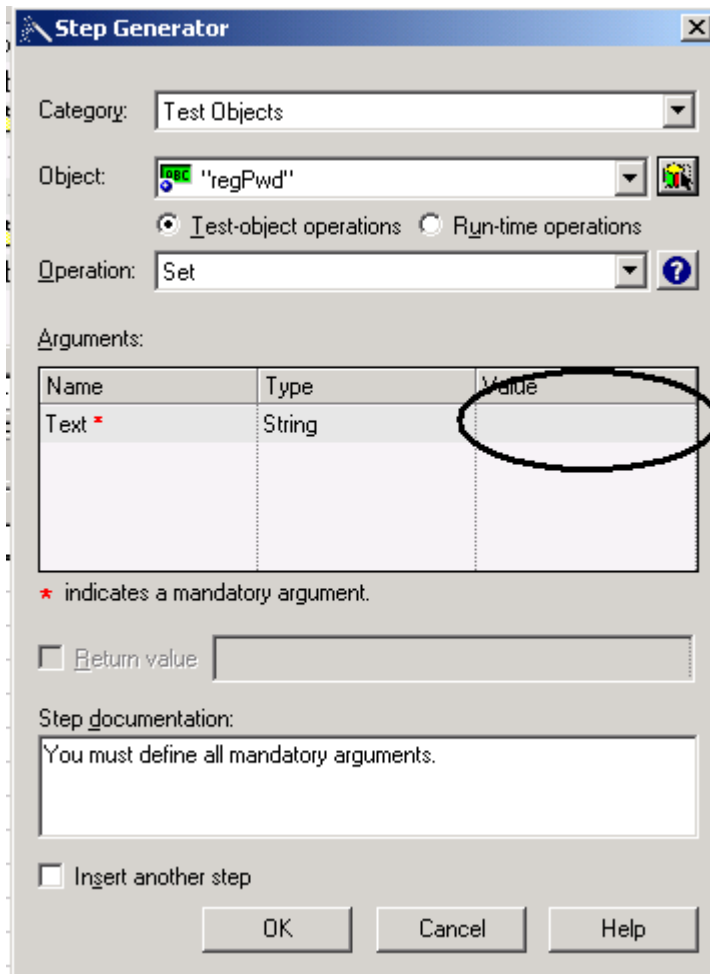


图 3.1.D2.9

点击 ok,新的步骤被添加

D3 运行脚本,完成脚本后可以继续运行脚本了

E 个人感受:无

3.2 手工识别控件的方法

A 问题描述:

在 QTP 的使用过程中,很多时候需要知道一些非录制流程上控件的属性,这时就可以使用手工识别控件,来获取其他控件的属性

B 处理流程:

B1 录制完成脚本

B2 编译脚本时,需要其他的控件的属性

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 录制完成脚本(详见 QTP 使用手册)

D2 在编译时,需要其他控件的属性,这时在 QTP 主界面点击 object spy 如下图

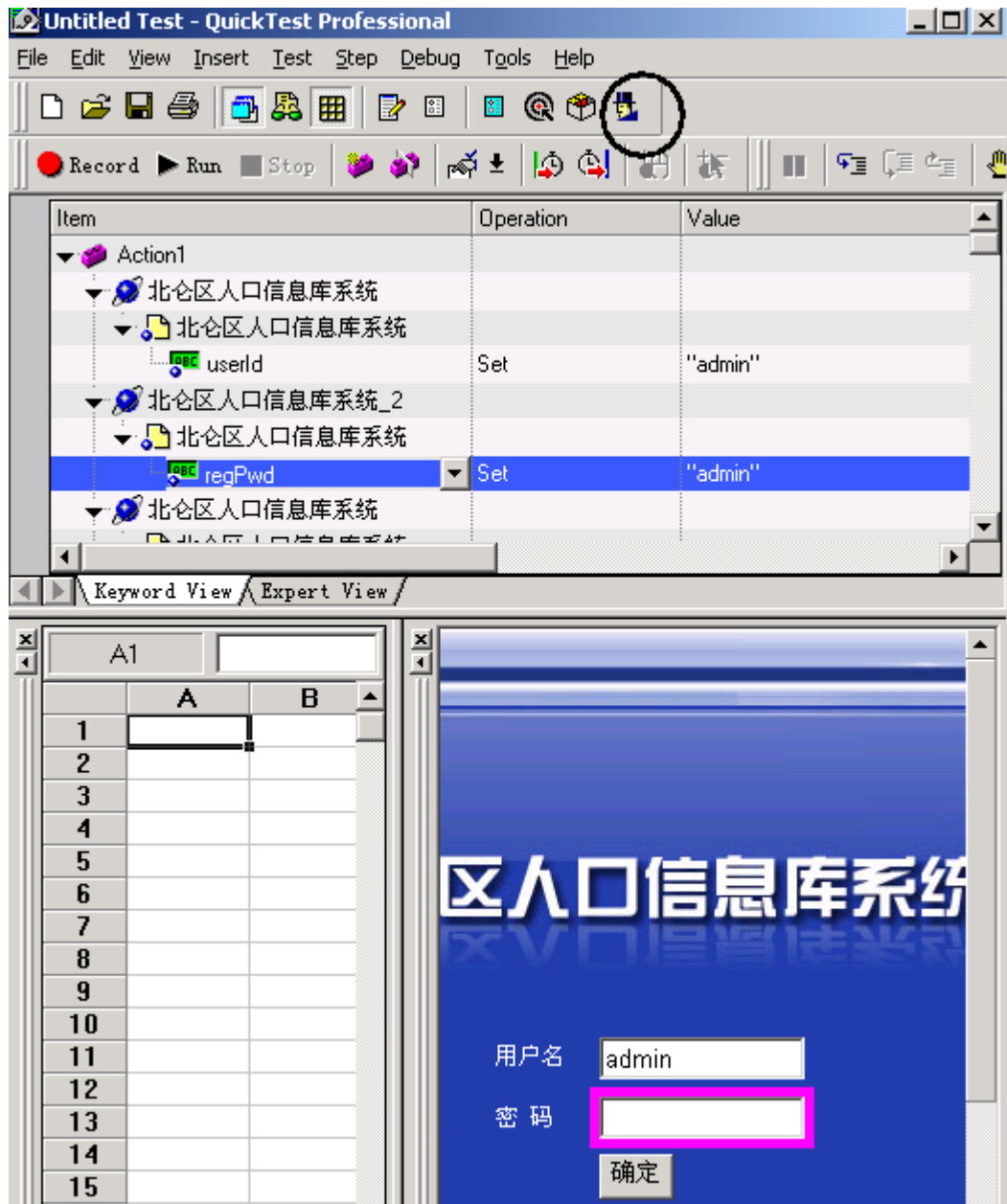


图 3.2.D2.1

弹出如下对话框

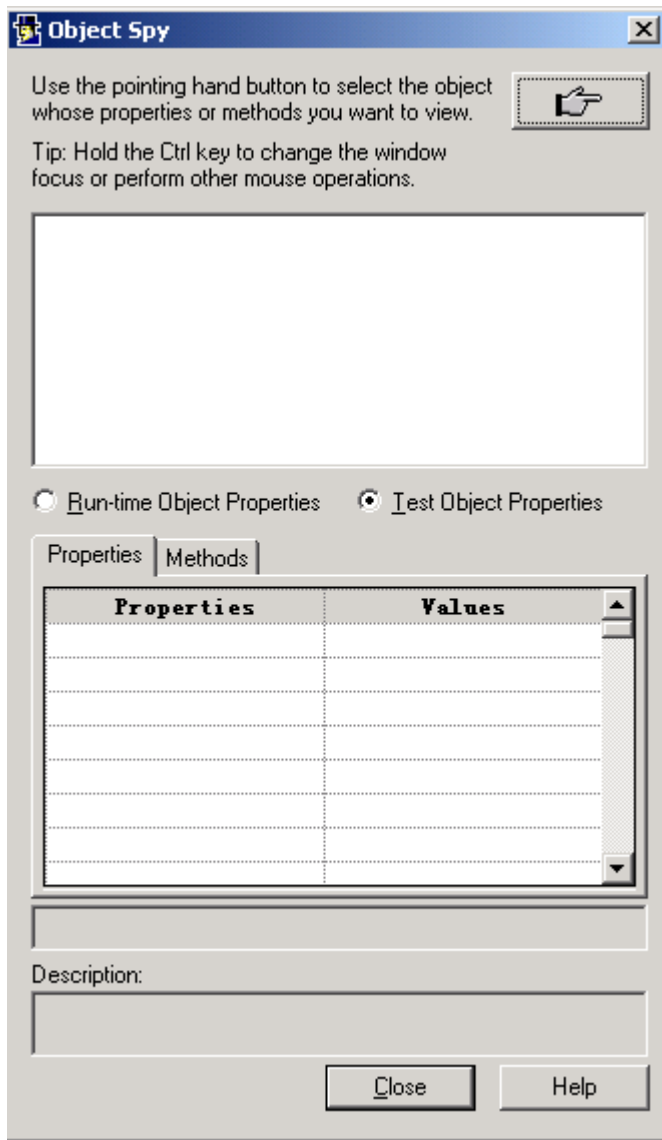



图 3.2.D2.2

点击  鼠标从箭头变成手型指针,在你的测试页选中一个你想要了解属性的控件,出现如下图

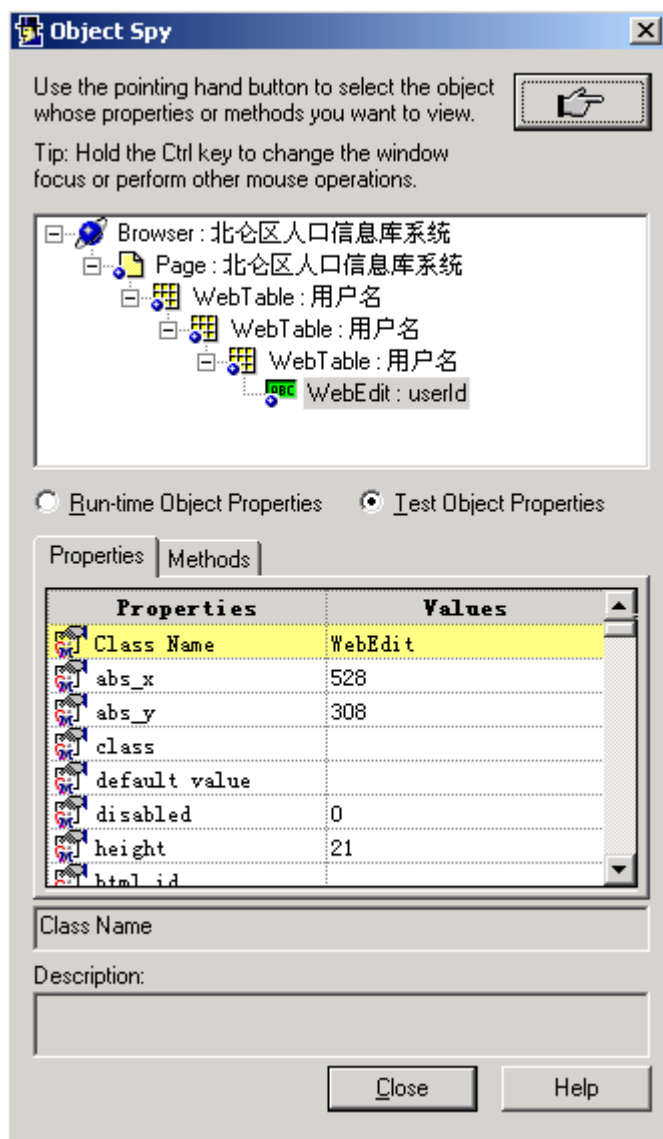


图 3.2.D2.3

他显示了控件的位置,以及其中的各种属性,通过这个你可以与你原有的控件属性进行对比

E 个人感受:无

4 多 Action 的问题

4.1 为什么会使用多个 Action

A 问题描述:

在录制脚本的时候通常会遇到这种情况,一个步骤需要反复执行,但是其他步骤不需要跟着执行,就比如添加信息,一般分为登陆,添加,退出,这三步,只有添加这个步骤是需要重复执行的,登陆和退出不需要重复执行

B 解决办法:

这个时候就需要用到多个 Action 调用的方式

C 个人感受: 无

4.2 Action 调用的设置

A 问题描述:

在决定使用多个 Action 调用之后需要对整个脚本有一个划分,解决这个问题,在录制脚本之前,先要对脚本进行划分,还是以添加信息为例,划分成三步,登陆,添加,退出,其中添加是主要部分而添加和退出是次要部分,分别对主要部分和次要部分作相应的设置,就能达到你的要求

B 处理流程:

B1 分别将登陆,退出录制成为单独的 Action 并且单独保存下来

B2 对这两个 Action 作设置

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 以登陆为例,录制好登陆脚本如下图

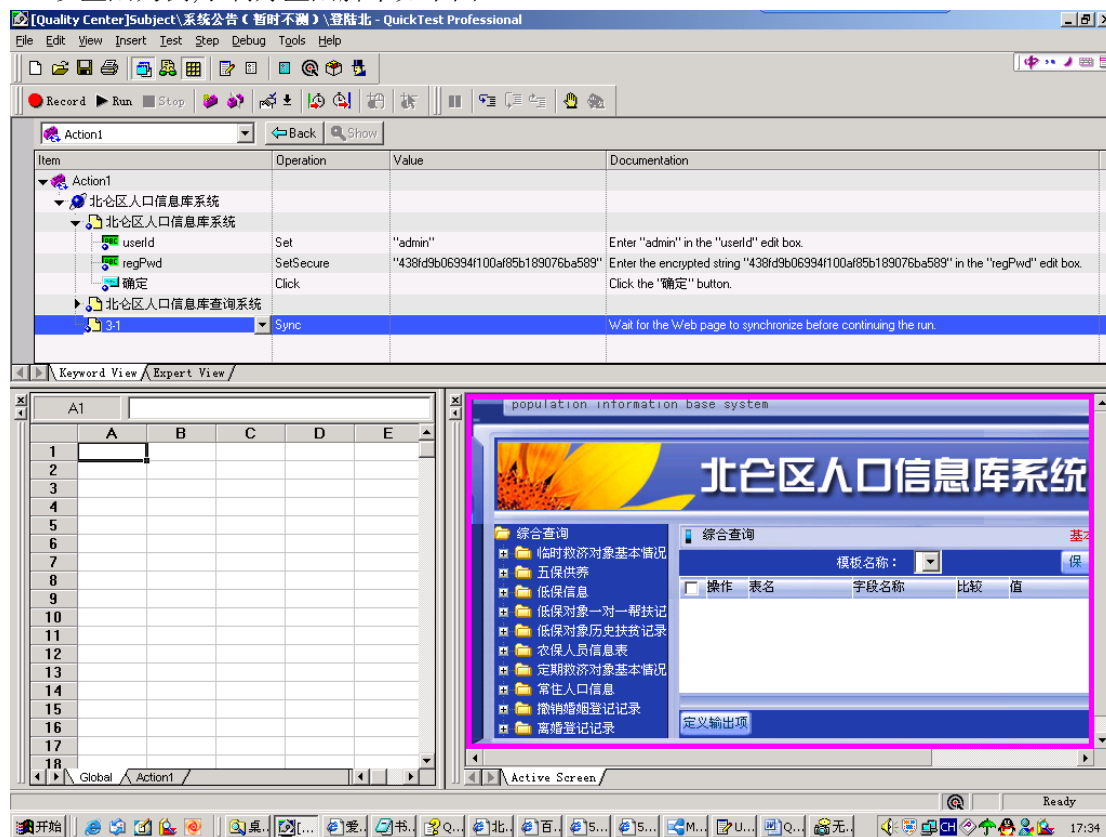


图 4.2.D1.1

注意: 很多人录制的时候习惯性的关闭录制完成后的浏览器,这里不能让 QTP 录制下关闭浏览器这个步骤,所以当登陆进系统后,先停止录制,再关闭浏览器

D2 对两个 Action 作设置,还是以登陆为例,选中脚本上的 Action

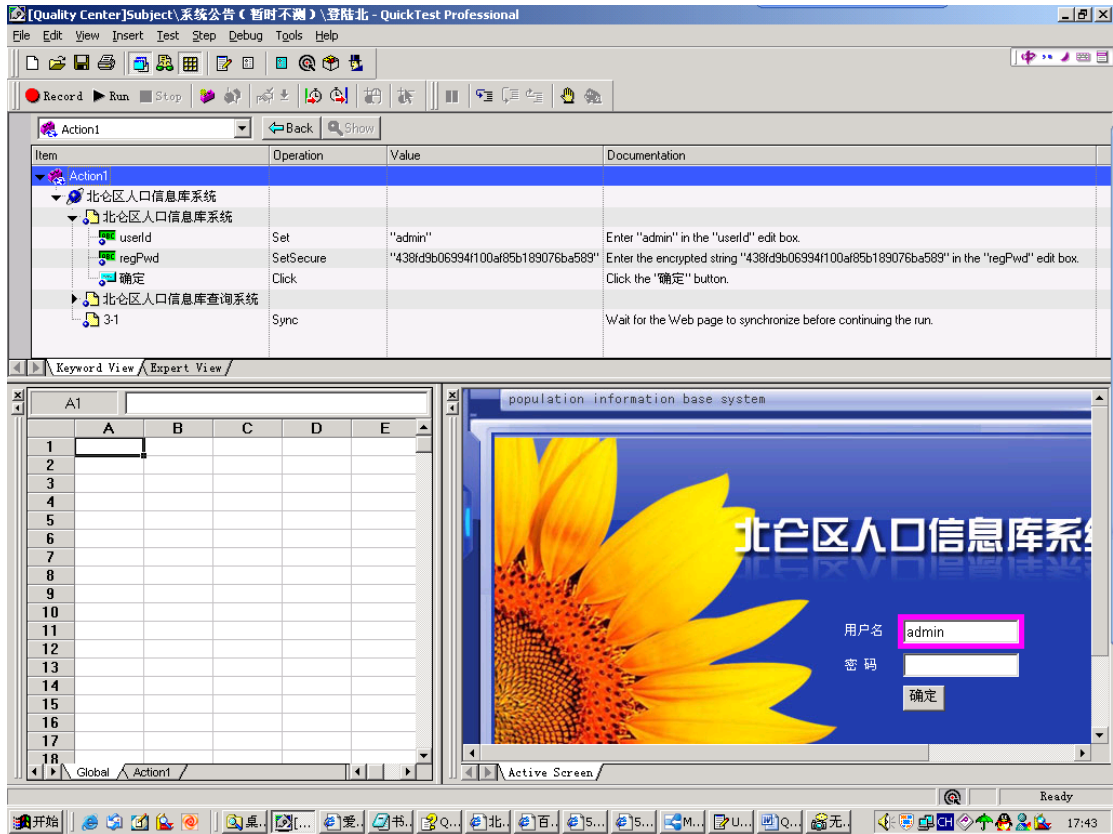


图 4.2.D2.2

点击右键,选择 Action properties 出现如下对话框

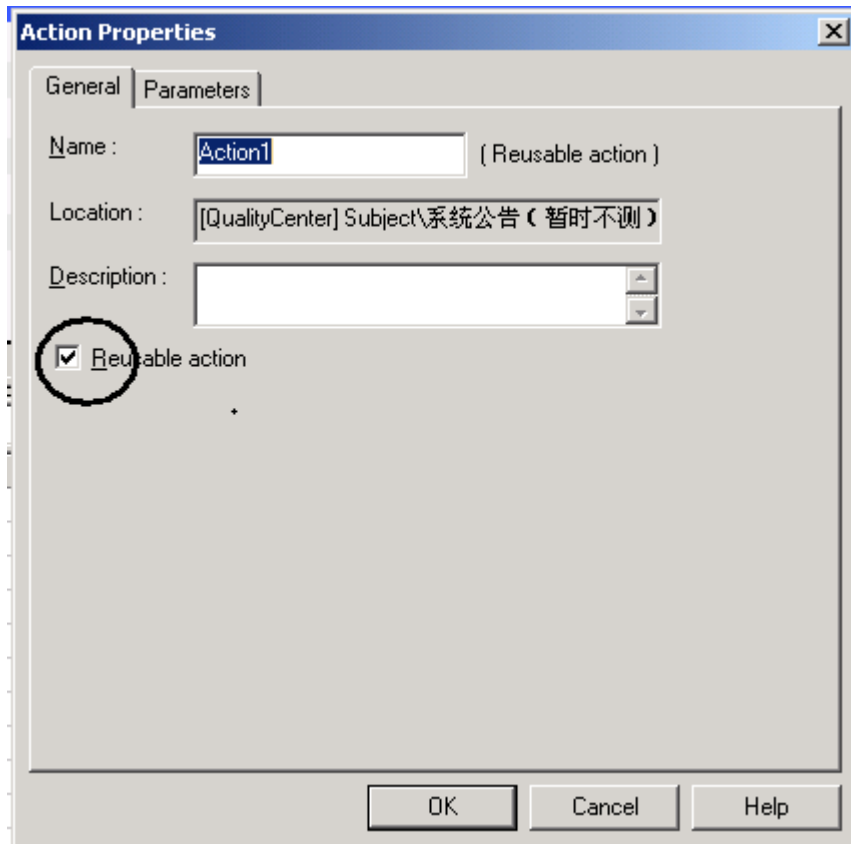


图 4.2.D2.3

将 reusable Action 选中,点击 ok,主界面变为下图所示

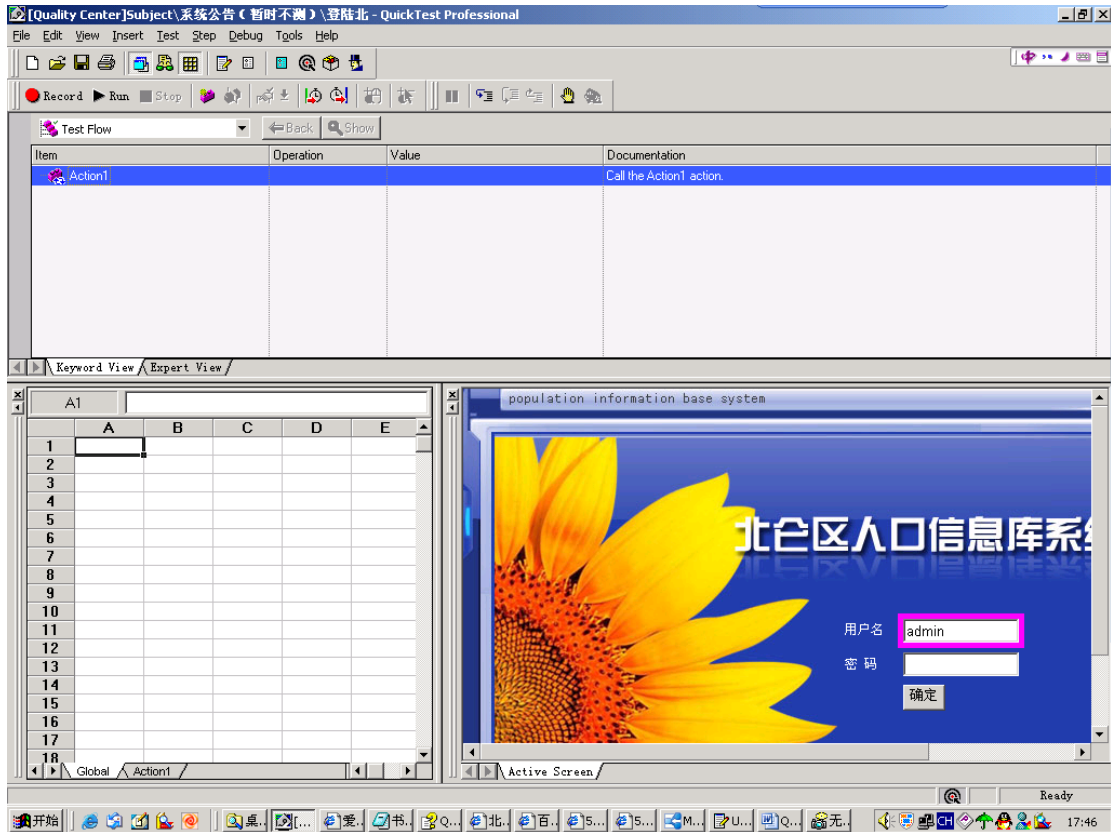
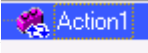


图 4.2.D2.4

Action 图标变为 

将退出也按上述步骤设定

E 个人感受:无

4.3 调用 Action

A 问题描述:

在设定完登陆和退出之后,还需要在整个脚本的主要部分(添加)上作一些设置

B 处理流程:

B1 调用已有脚本

B2 设定 record and run setting

B3 调整 Action 的位置

B4 录制脚本

B5 对添加脚本进行参数化

B6 对于添加 Action 属性的设定

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 调用脚本,在 QTP 主页面选择 insert ---call to existing Action

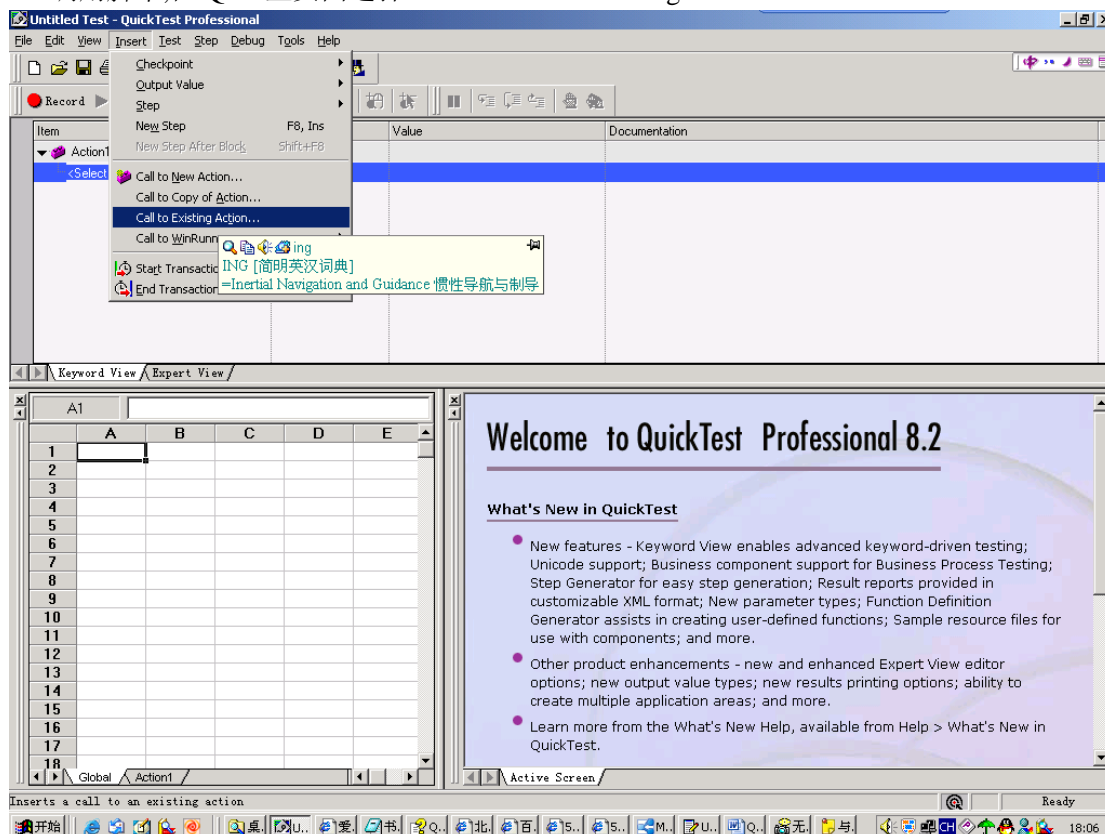


图 4.3.D1.1

弹出如下对话框

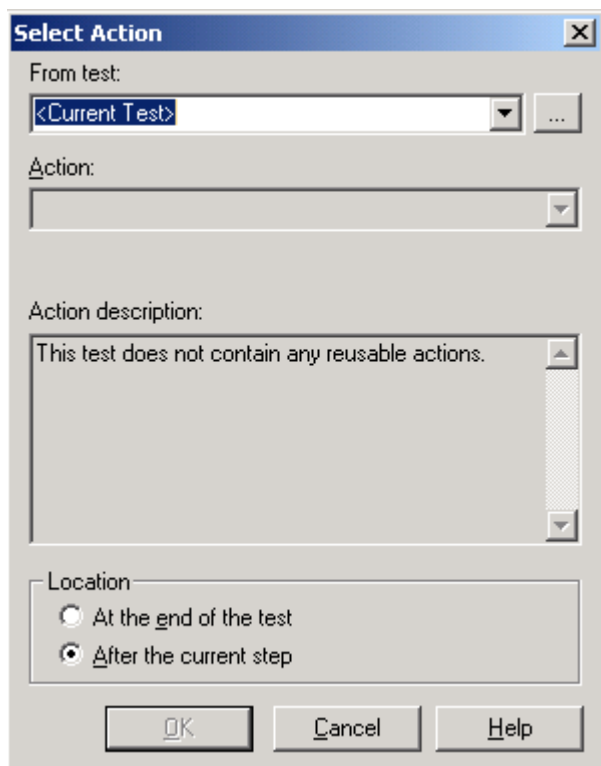


图 4.3.D1.2

在 from test 中,点击... 会弹出文件夹选项(如果 QTP 已经连接 TD 等测试中心,那么会弹出测试中心的保存目录,本例是连接 TD 的)如下图

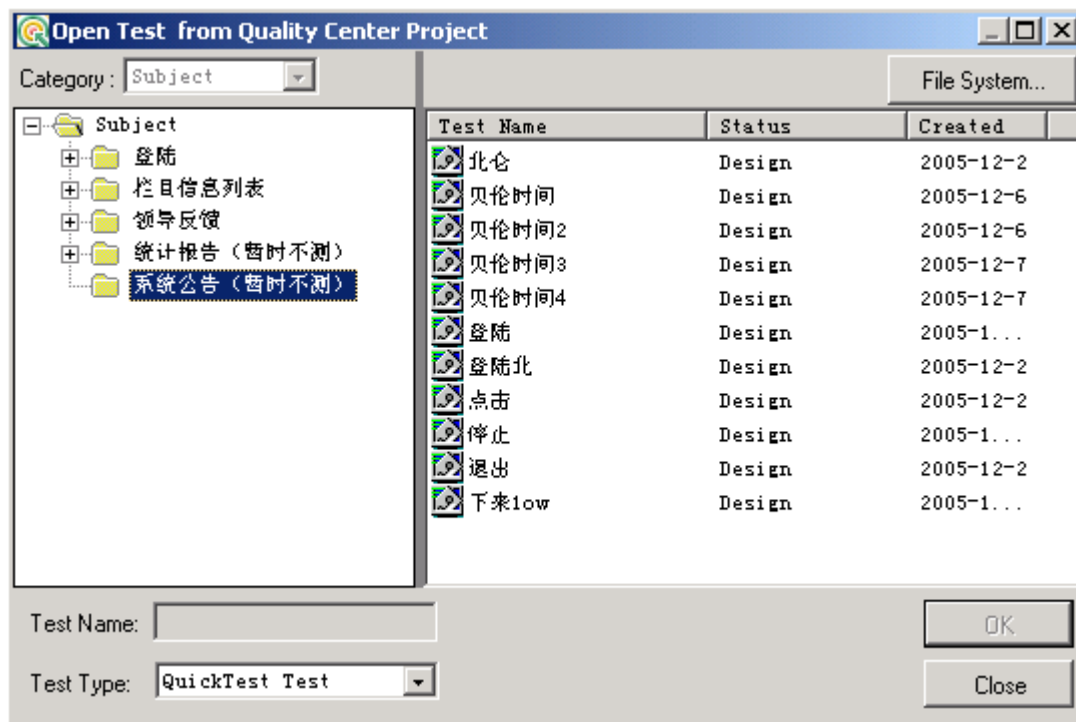


图 4.3.D1.3

选择你刚刚录制并且设置完成的登陆,点击 ok,弹出如下对话框

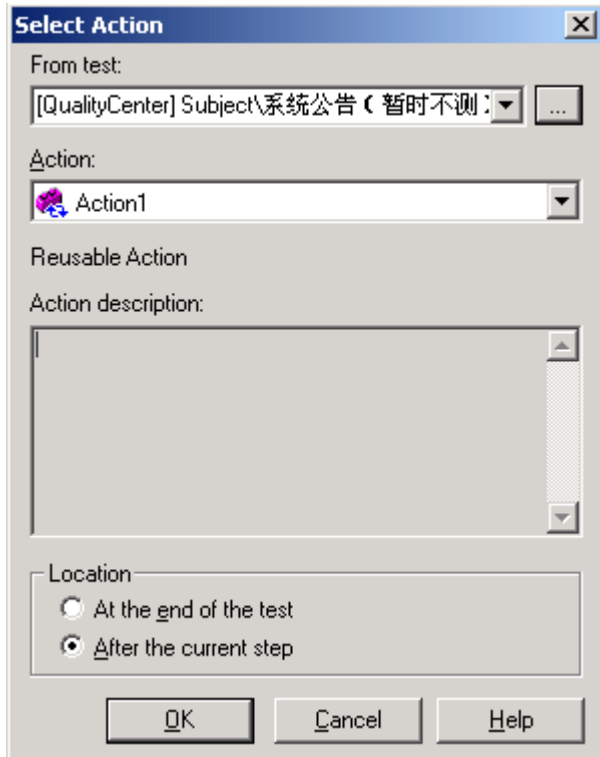


图 4.3.D1.4

选择 at the end of the test 点击 ok

登陆 Action 被调用到当前的 Action 中,执行如上操作,将退出也导入到当前的 Action 中
D2 设定 record and run setting,在 QTP 主界面,选择 test---record and run setting

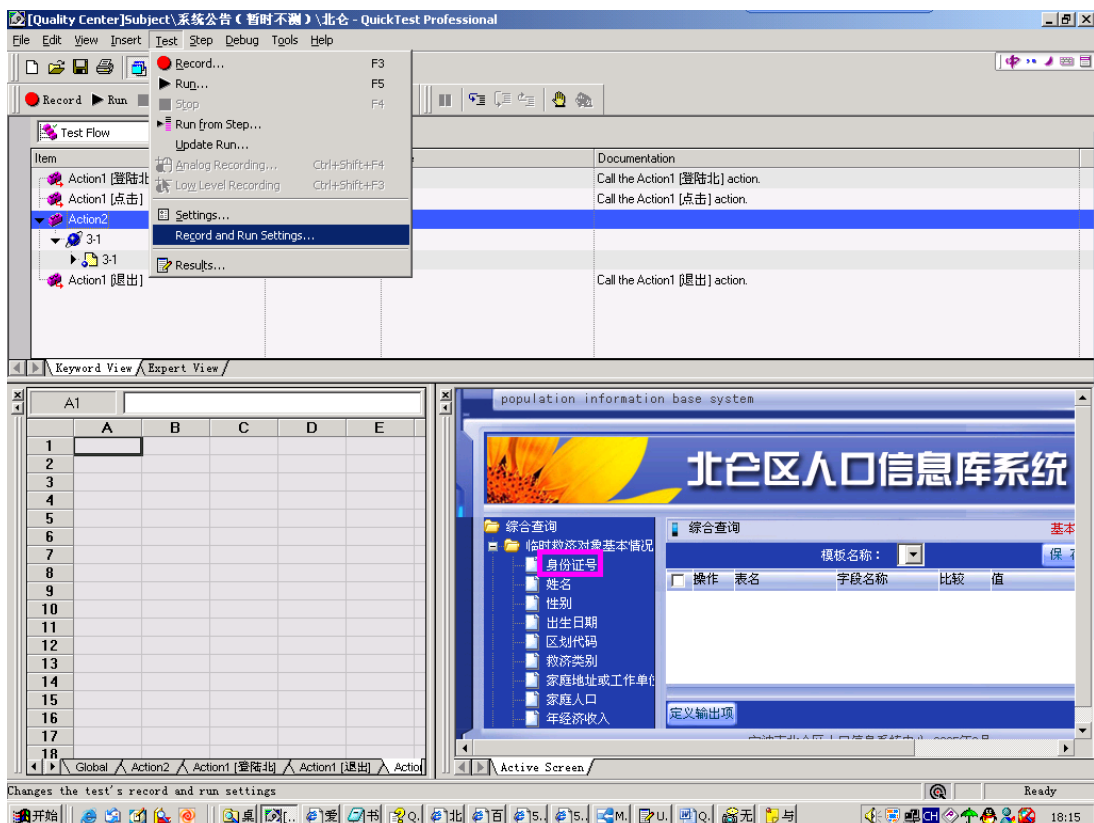


图 4.3.D2.5

弹出如下对话框

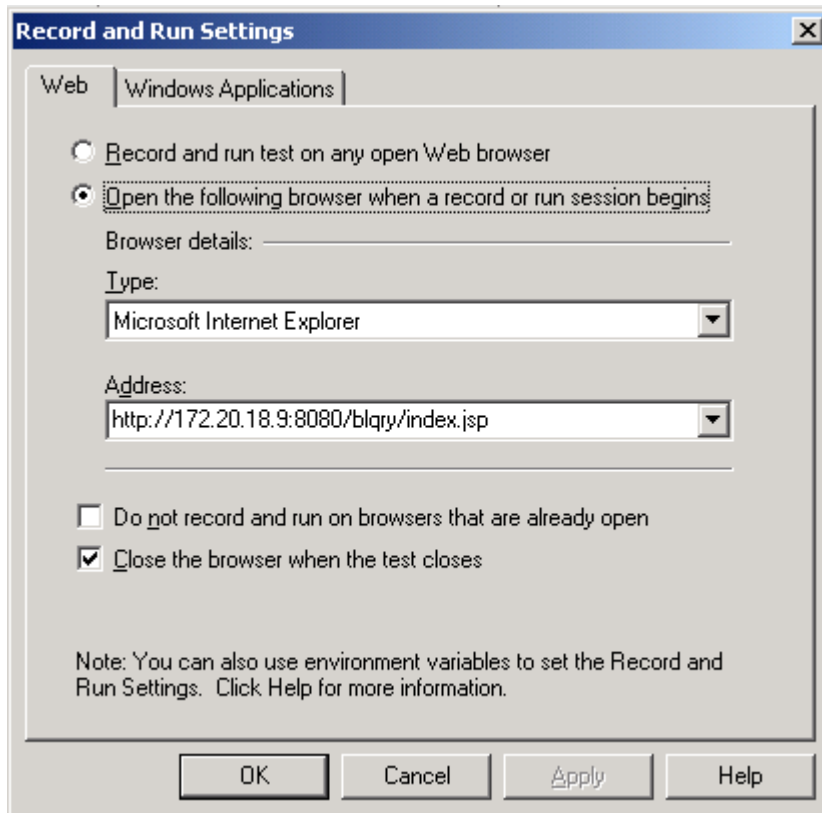


图 4.3.D2.6

选择 record and run test on any open web browser

D3 调整 Action 的位置,当完成上述操作,你会看到你的界面如下图所示

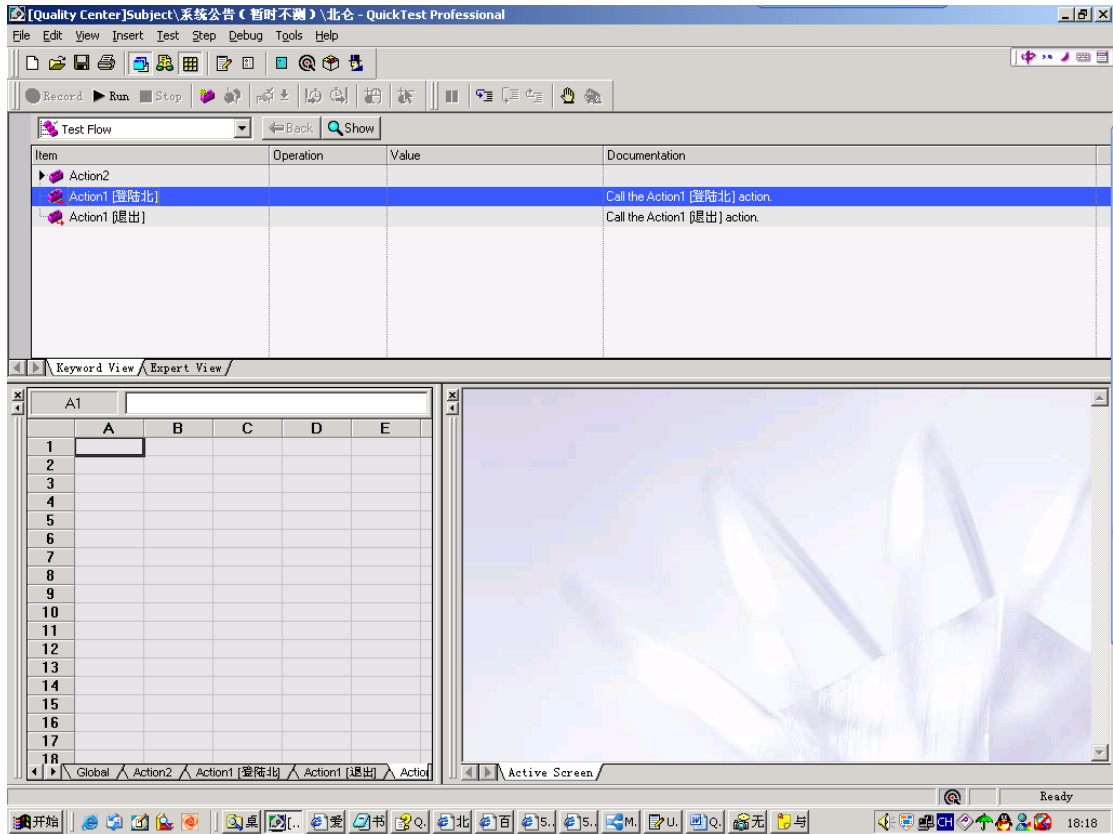


图 4.3.D3.5

三个 Action 顺序排列.但是你需要将登陆 Action 放在第一步,添加 Action 放在第二步,退出 Action 放在第三步,你可以通过用鼠标选中其中的一个 Action 通过拖动鼠标的方式来调整 Action 的位置

D4 录制脚本,手动打开 IE 浏览器,手动登陆到你登陆 Action 执行完毕后的那个界面,然后在 QTP 中点击 record,开始录制脚本,注意,不要录制最后关闭这个步骤

D5 对添加脚本进行参数化,选择你要参数化的步骤,并且参数化你的步骤,但是在选择 global 或者 Action 时,选择 Action,如下图

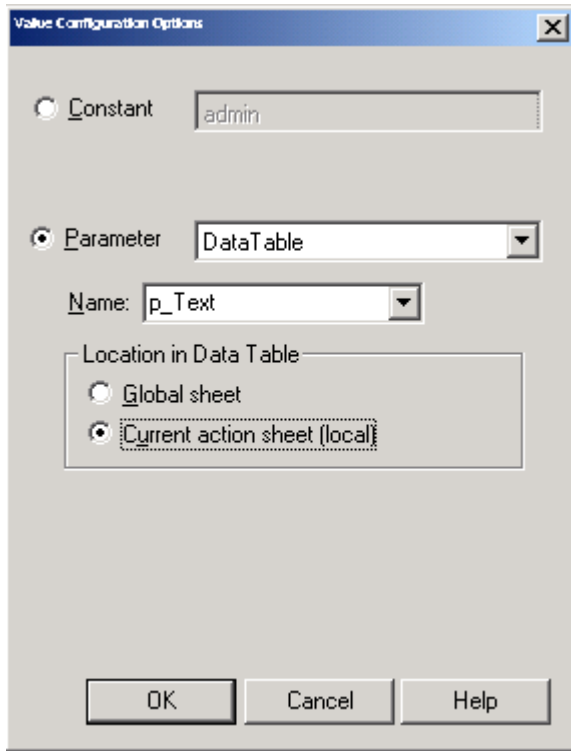


图 4.3.D5.6

D6 对于添加 Action 属性的设定,选中你的添加 Action,点击右键,并且选择 Action call properties 如下图

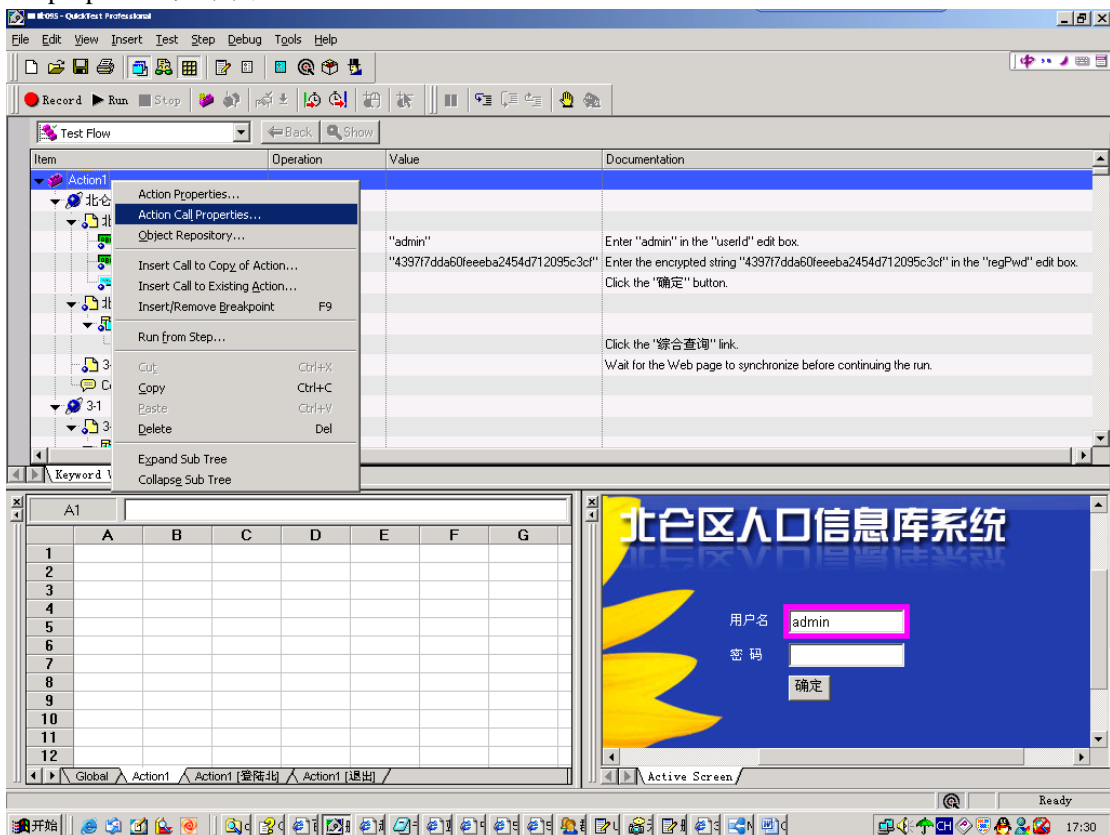


图 4.3.D6.7

在弹出的对话框中选择 run on all row,如下图

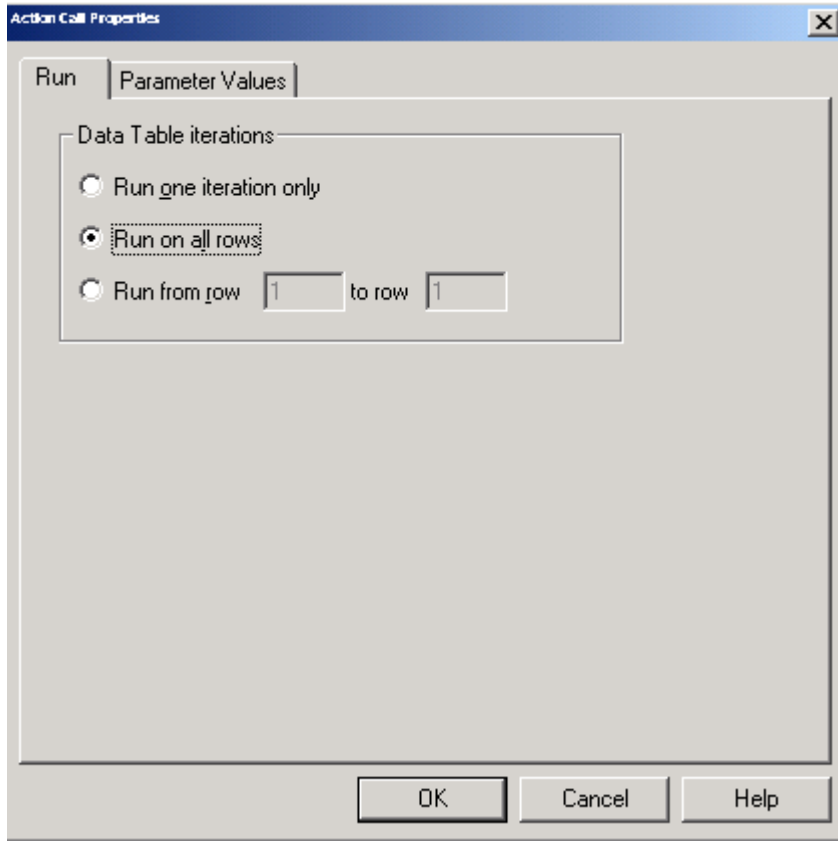


图 4.3.D6.8

E 个人感受:无

4.4 调用 Action 的模式

A 问题描述与解答

要达到 4.2 所要达到的目标,其中的几个 call Action 必须依照 4.2 的属性来设置,否则将无法按要求运行

B 个人感受:无

4.5 copy Action 与 call existing Action 的区别

A 问题描述

在 QTP 中可以使用两种调用其他 Action 的方法分别是 call to copy of Action 与 call existing Action,他们之间的区别如下

B 问题解答

两种方式都是将以被保存的 Action 调用过来,但是 copy Action 是将原本 Action 中的所有属性,及其参数值调用过来,并且可以在其上随意的更改,而且这些更改并不影响源 Action,而 call existing Action 虽然也是将源 Action 的所有属性及其参数值都调用过来,但是你不能更改其中的任何步骤和参数值,如果一定要更改,需要重新打开源 Action,在源 Action 上进行更改,所作的改动将自动体现在调用的 Action 上

C 个人感受: 无

5 QTP 系统参数设置对录制的影响

5.1 关于录制设置的区分 web event recording configuration

A 问题描述:

在录制过程中,默认的录制模式很多情况下是不能满足录制要求的,需要进行特殊的设定,你可以在 web event recording configuration 中完成这些设定

B 处理流程:

B1 启动 QTP

B2 进入到 web event recording configuration 对话框

B3 web event recording configuration 的设置

B4 保存退出,录制脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 启动 QTP,如下图

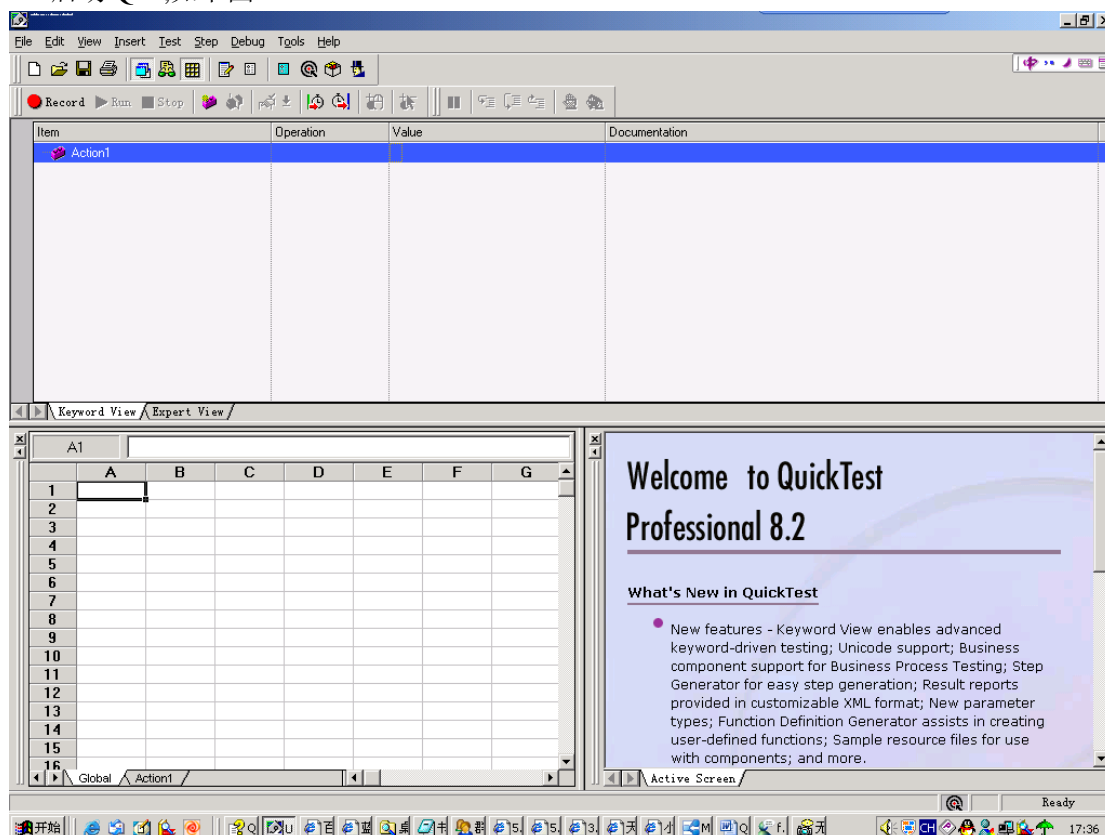


图 5.1.D1.1

D2 进入 web event recording configuration 对话框,点击 tools---web event recording

configuration 如下图

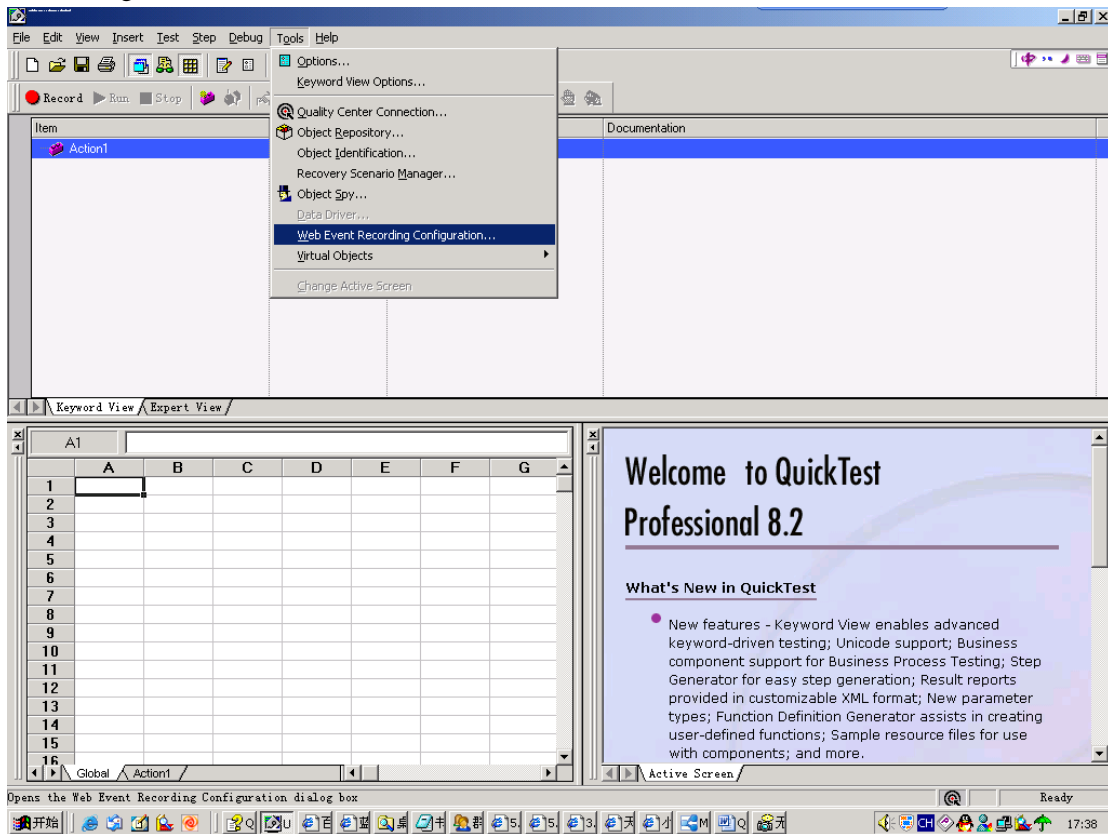


图 5.1.D2.2

D3 web event recording configuration 的设置,弹出 web event recording configuration 对话框, 如下图

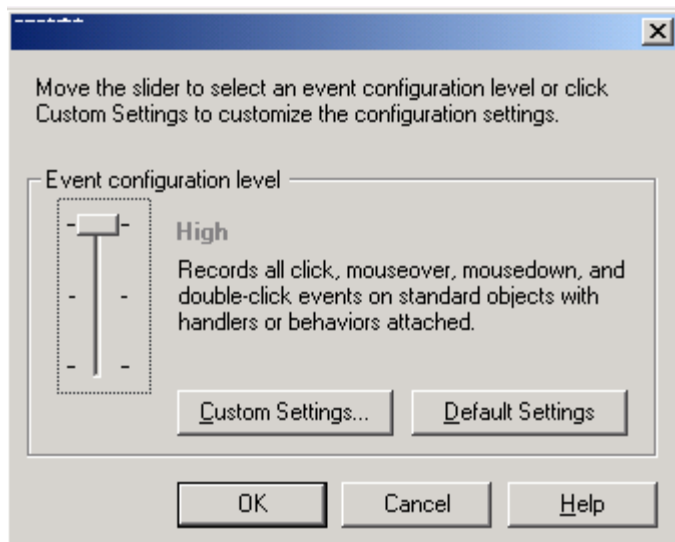


图 5.1.D3.3

这里有三个录制级别可以设置分别为最低,中等,高级
如果你认为还是不能满足你的录制需要可以点击 custom settings..按钮弹出如下对话框

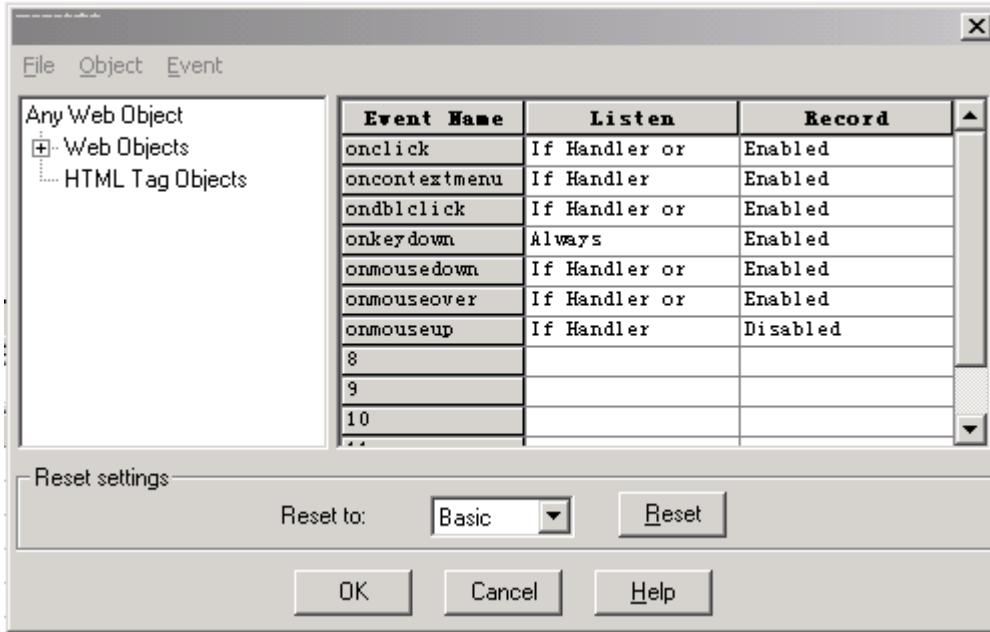


图 5.1.D3.4

左边是你的对象列表树,如下图



图 5.1.D3.5

他包含了 web 中 QTP 所能识别的所有控件,你可以通过 add object 添加 object 控件

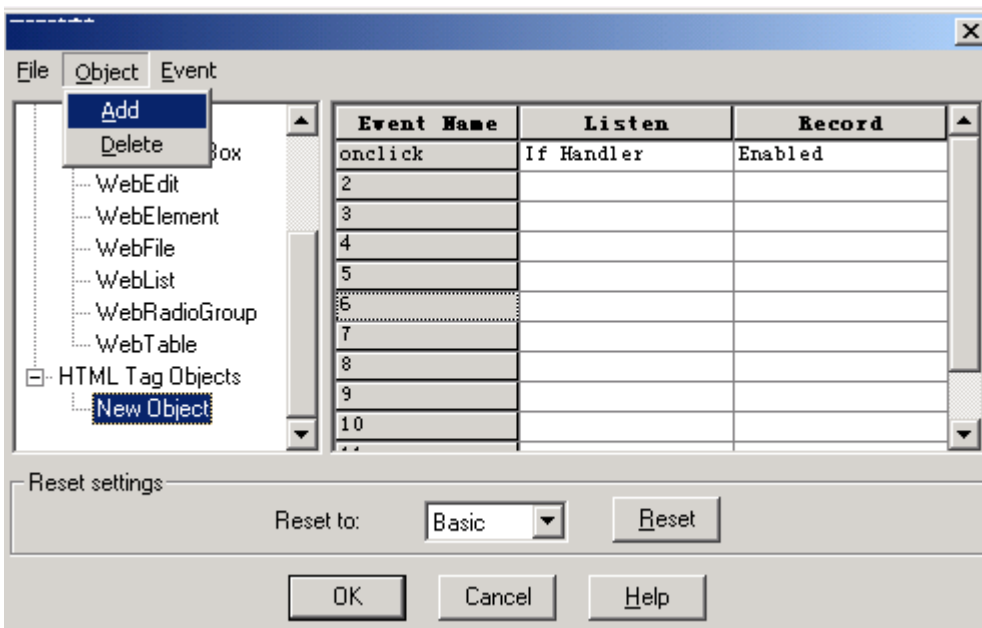


图 5.1.D3.6

右边是跟控件相对应的实例,如下图

Event Name	Listen	Record
onclick	If Handler or	Enabled
oncontextmenu	If Handler	Enabled
ondblclick	If Handler or	Enabled
onkeydown	Always	Enabled
onmousedown	If Handler or	Enabled
onmouseover	If Handler or	Enabled
onmouseup	If Handler	Disabled
8		
9		
10		

图 5.1.D3.7

他包含了相对应的控件的鼠标录制属性,你可以通过 add event 来添加 event 控件

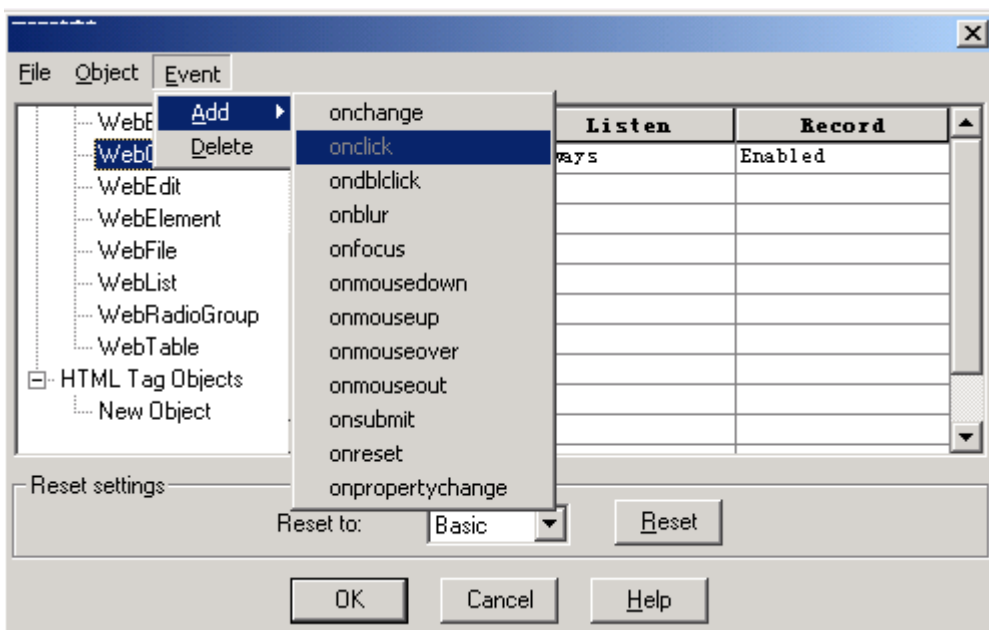


图 5.1.D3.8

D4 保存退出,录制脚本,点击 ok 保存设置,这时可以按照你的要求录制脚本
E 个人感受:无

5.2 关于 QTP 录制 mouseover 得问题

A 问题与解答

修改了 Web Event Recording Configuration 中的以下设置:

将 Any Web Object、Web Object-->Image、Web Object -->WebArea 这三项上的 Onmouseover 都改成 Always Enable 就可以了

B 个人感受: 无

5.3 关于 start transAction 的问题

A 问题描述:

有时需要查看 QTP 执行脚本的执行时间,以及使用 loadrunner 调用 QTP,也需要加一个特殊的功能键 start transAction,因为 loadrunner 只能调用 start transAction 和 end trasAction 之间的代码段

B 处理流程:

B1 录制脚本

B2 添加 start transAction

B3 运行脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,进入 QTP 页面,录制你的脚本,录制完成后点击 stop

D2 添加 start transAction 如下图在你需要查看时间的脚本上端添加 start transAction,如下图, 点击 start transAction 按钮

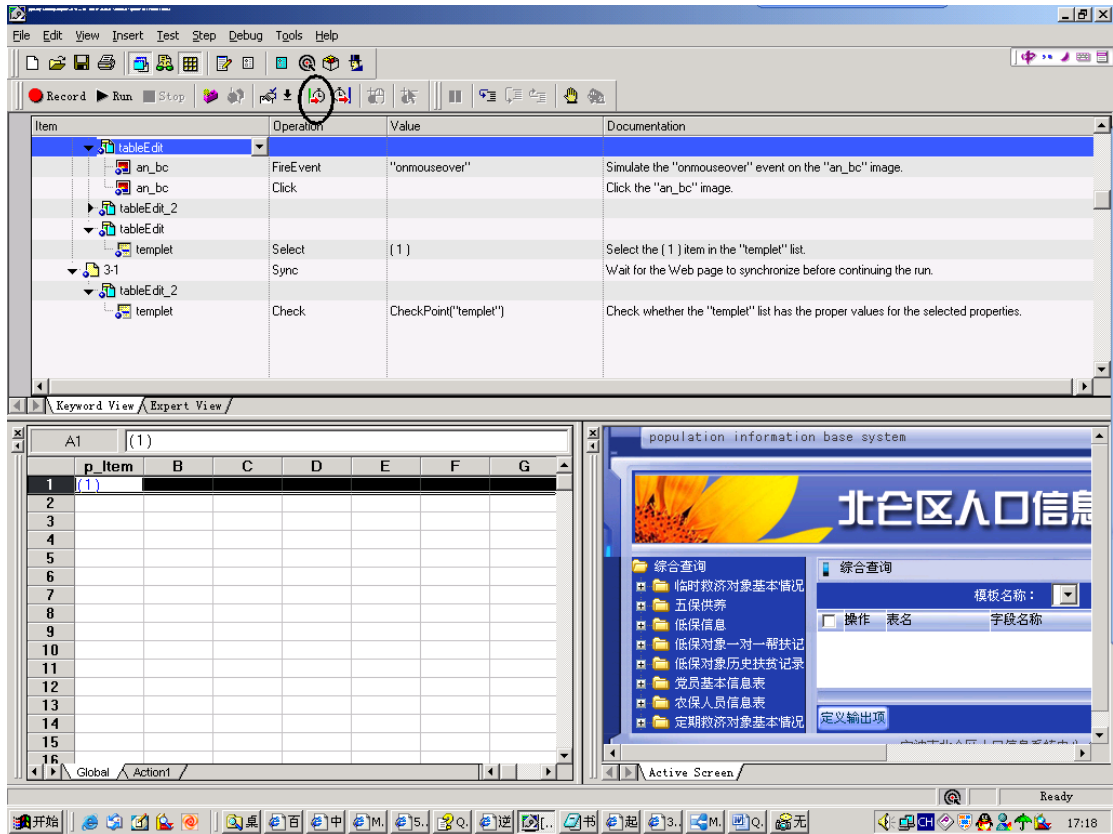


图 5.3.D2.1

弹出如下图对话框

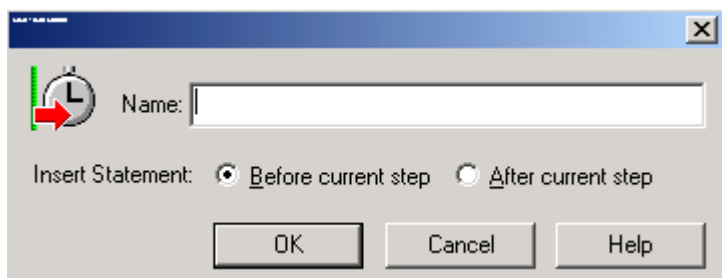


图 5.3.D2.2

输入名称,并且选择是在你所选步骤之前加还是之后,设定好后,点击 ok 添加一个步骤,如下图

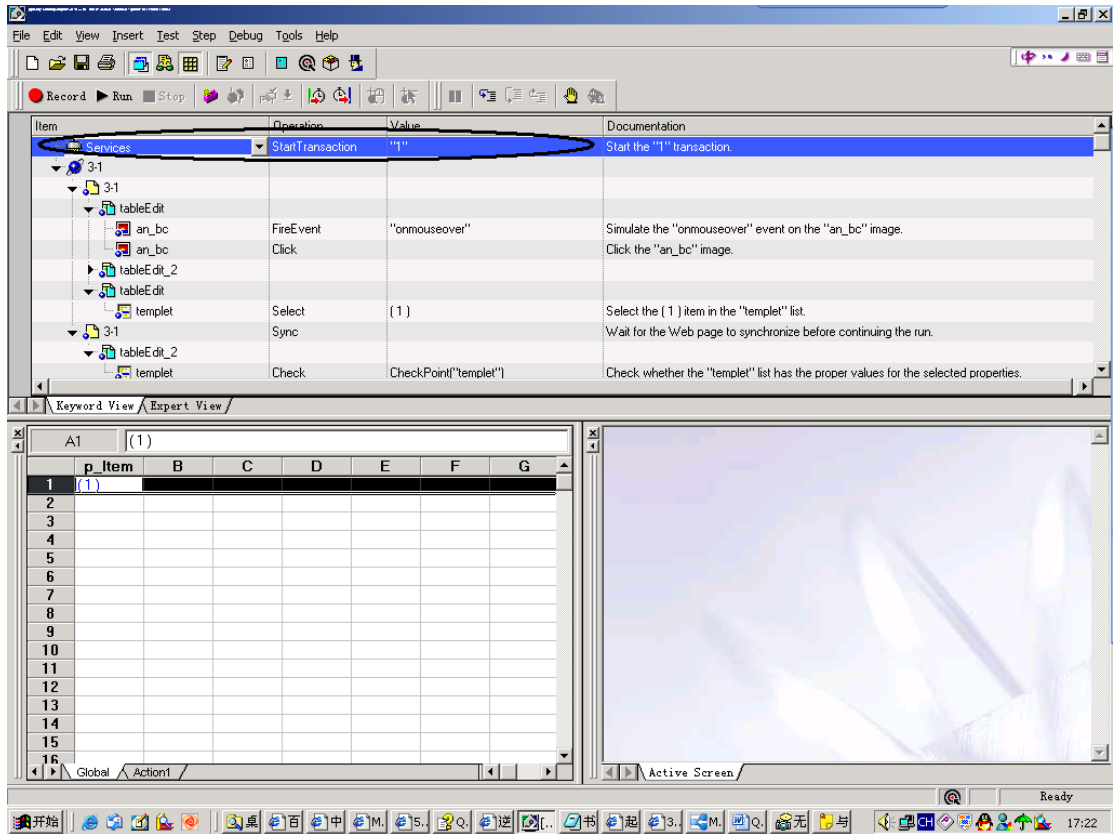


图 5.3.D2.3

再选择你要中止的步骤,点击 end stransAction,如下图

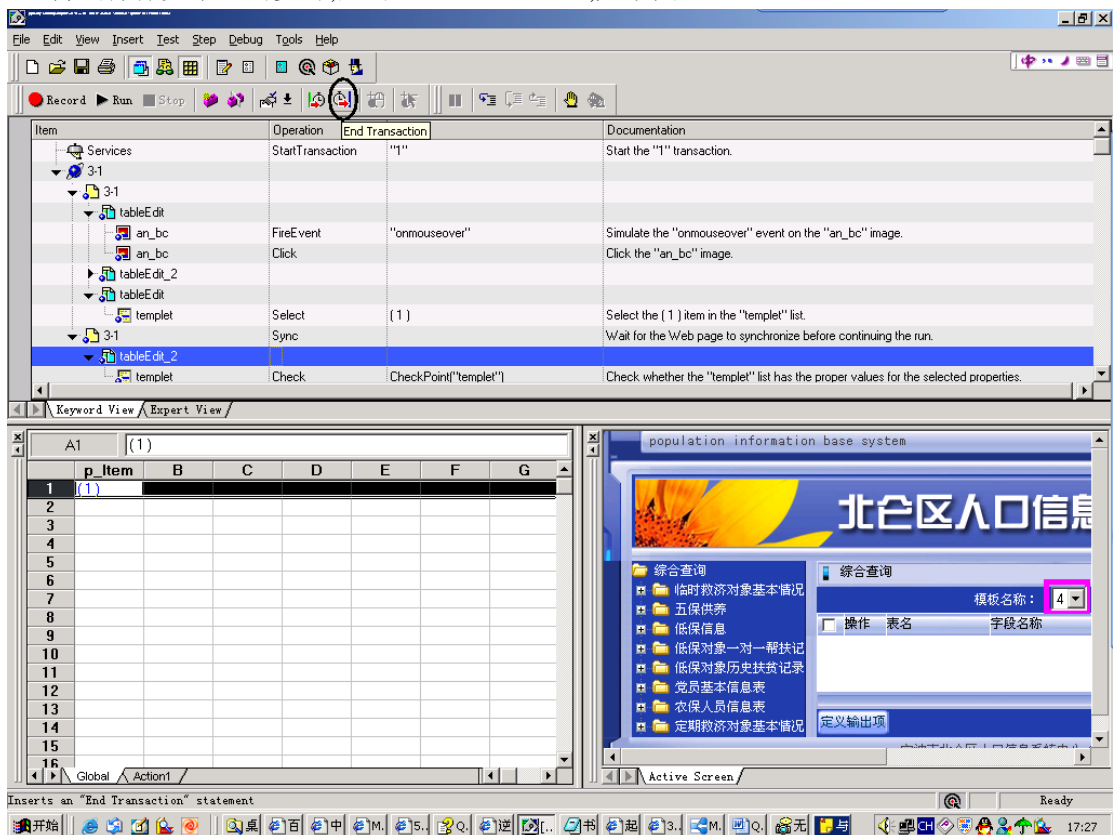


图 5.3.D2.4

弹出如下对话框

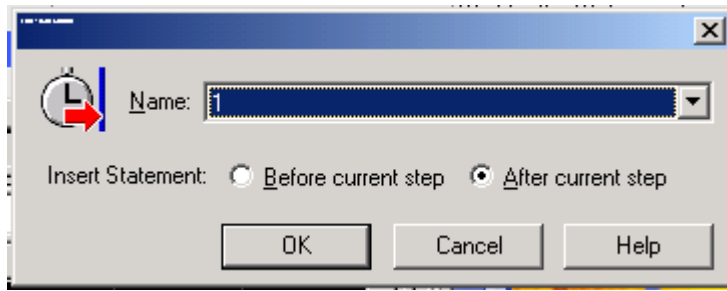


图 5.3.D2.5

选择你添加的 start stransAction

点击 ok

添加成功

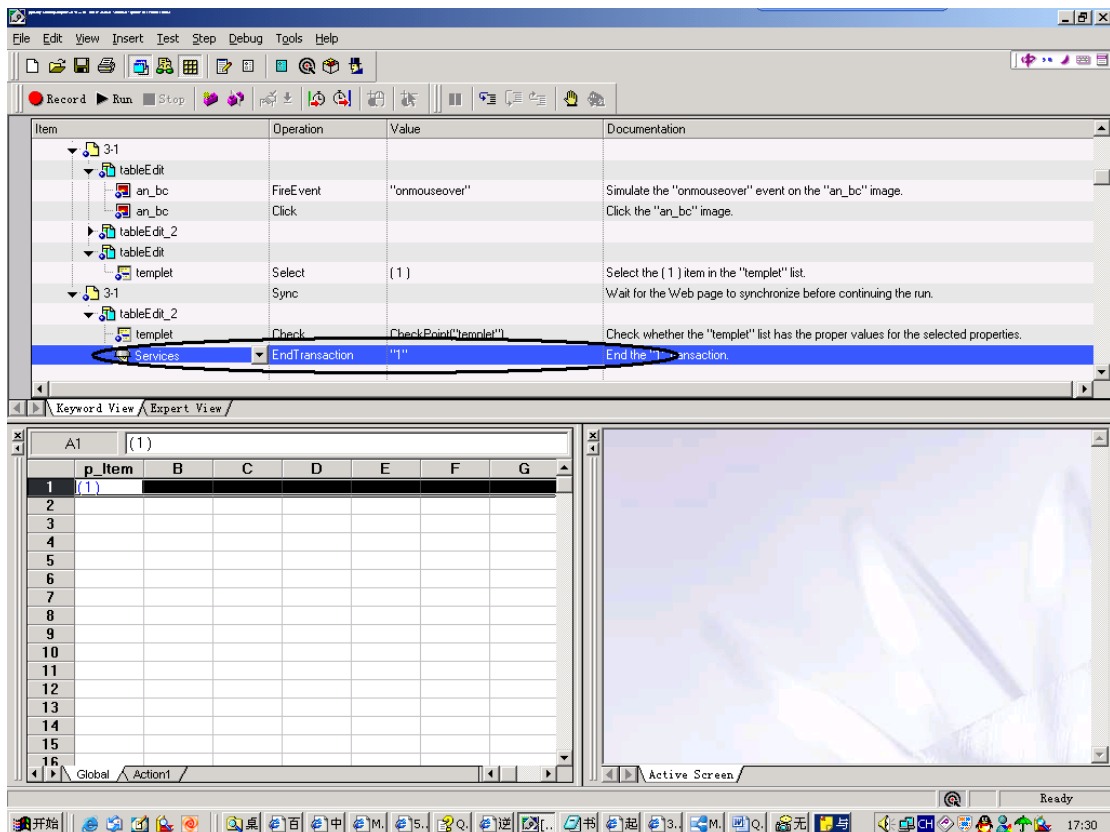


图 5.3.D2.6

添加成功

D3 运行脚本,运行后就能在 result 中查看时间信息

E 个人感受:无

5.4 虚拟对象的使用

A 问题描述:

在实际测试过程中,很多时候网页上的控件并不能都被 QTP 识别,例如网页上一个 flash 动画,实际上是一个链接,但是 QTP 无法识别出 flash 这个控件,这时使用 QTP 中的虚拟对象来解决这个问题

B 处理流程:

B1 使用虚拟对象,将 flash 设为一个对象

B2 录制脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 使用虚拟对象,将 flash 设为一个对象,点击 tools---virtual objects---new objects,弹出如下对话框



图 5.4.D1.1

点击 next,进入下一个对话框

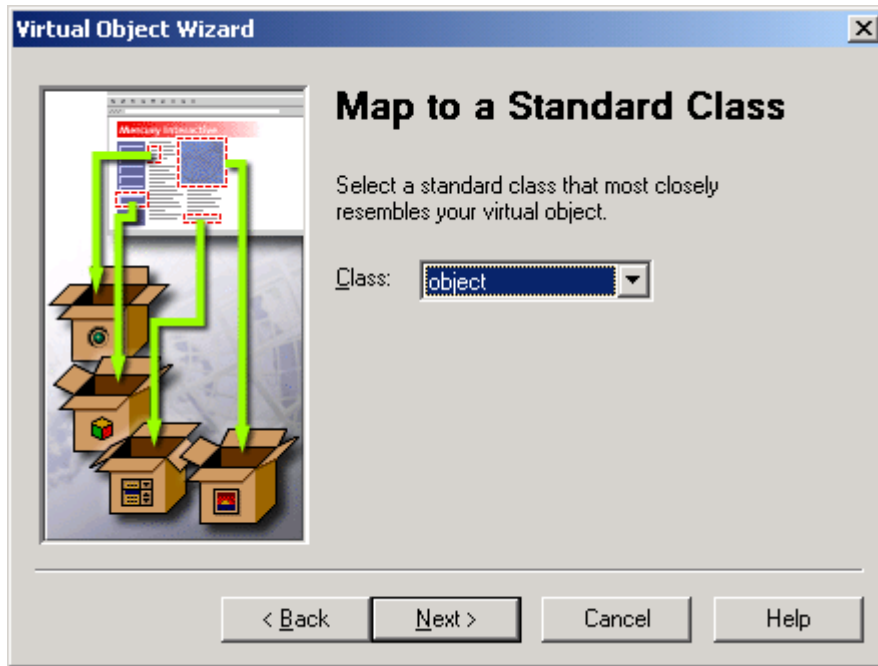


图 5.4.D1.2

在 class 里选择你想将 flash 设为那个 QTP 能够识别的对象,点击 next

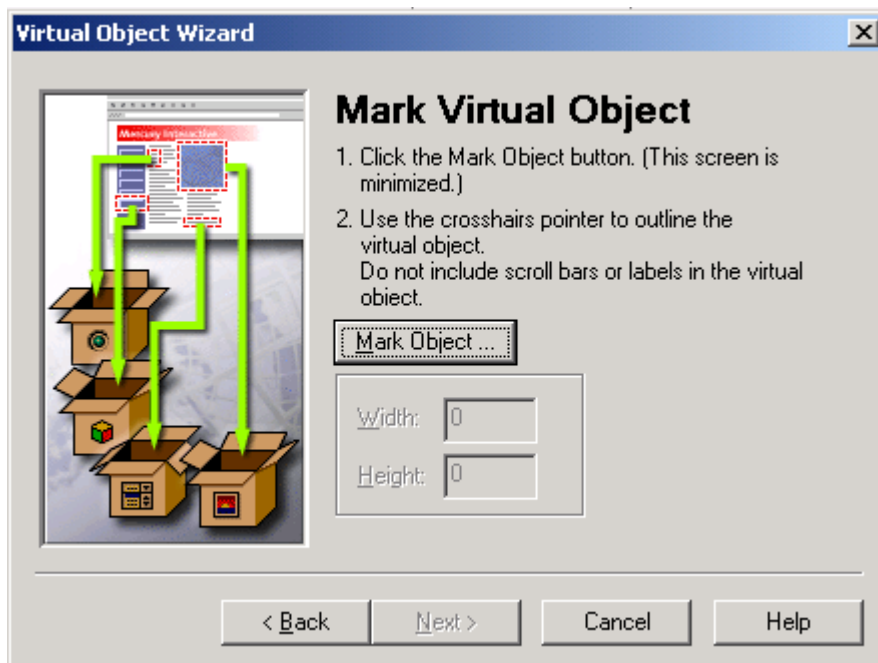


图 5.4.D1.3

打开你的页面,页面中包含你要的虚拟的对象,并且点击上面对话框中的 mark object 按钮,页面会最小化,鼠标会编程"+",你可以使用"+"来限定虚拟对象的范围选好后,会在 width height 中显示虚拟对象的高和宽,并且点击 next,弹出如下对话框

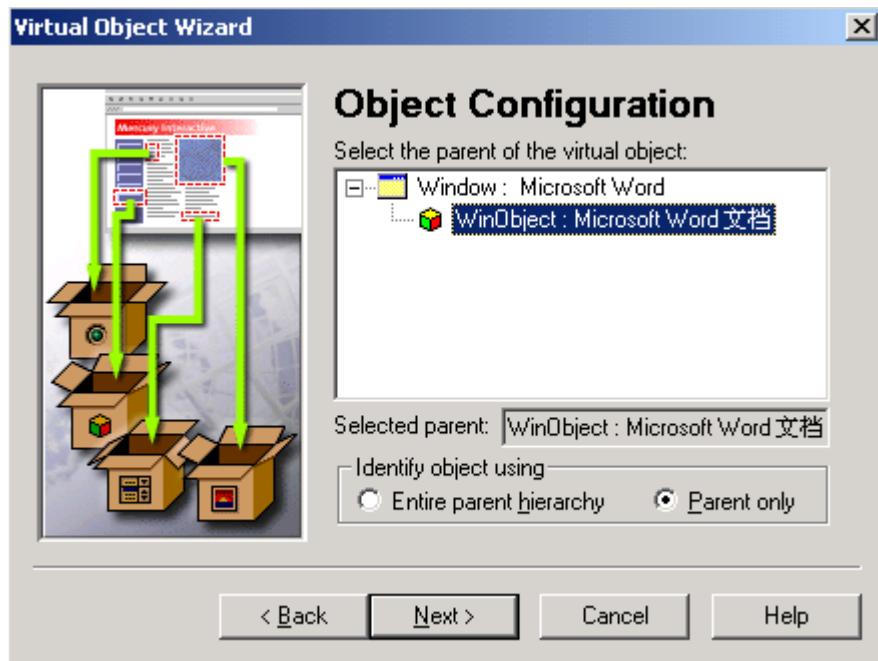


图 5.4.D1.4

选择 `entire parent hierarchy` 或者 `parent only` 点击 `next`



图 5.4.D1.5

设定虚拟对象的名称,以及收藏的名字,点击 `finish`

虚拟对象添加完成

点击 `tools---virtual object--- virtual object manager` 如下图

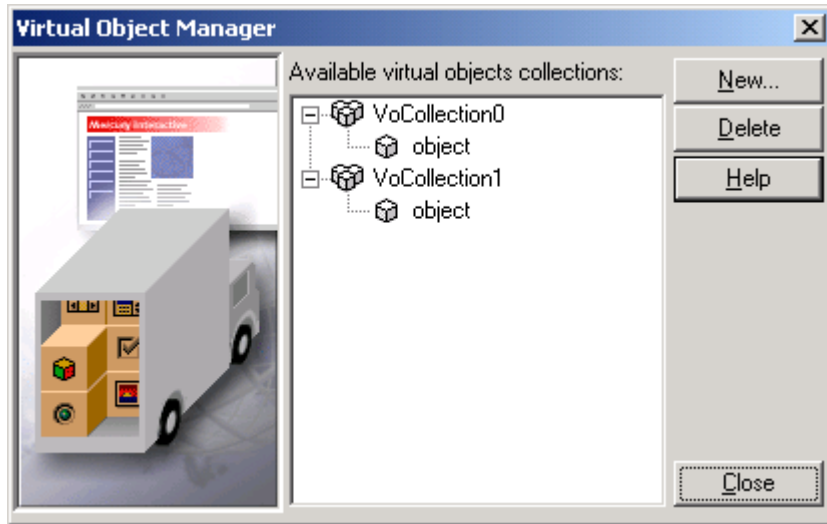


图 5.4.D1.6

刚加的 object 在 Vocollection1 中,你可以在其中对这个虚拟对象,删除或者添加新的虚拟对象

D2 录制脚本,录制脚本,在录制过程中会自动将 flash 控件识别成 object 控件

E 个人感受:无

5.5 对于 Action screen 捕捉不精确的解决办法

A 问题描述:

QTP 不能 100%精确的捕捉到 Action screen,但是对于 Action screen 捕捉不精确的问题,可以通过 QTP Action screen 的设置来逐步精确(建议在录制时设检查点,避免捕捉不精确的问题)

B 处理流程:

B1 在 QTP 中设置捕捉级别

B2 录制脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 在 QTP 中设置捕捉级别,点击 tools—options 如下图

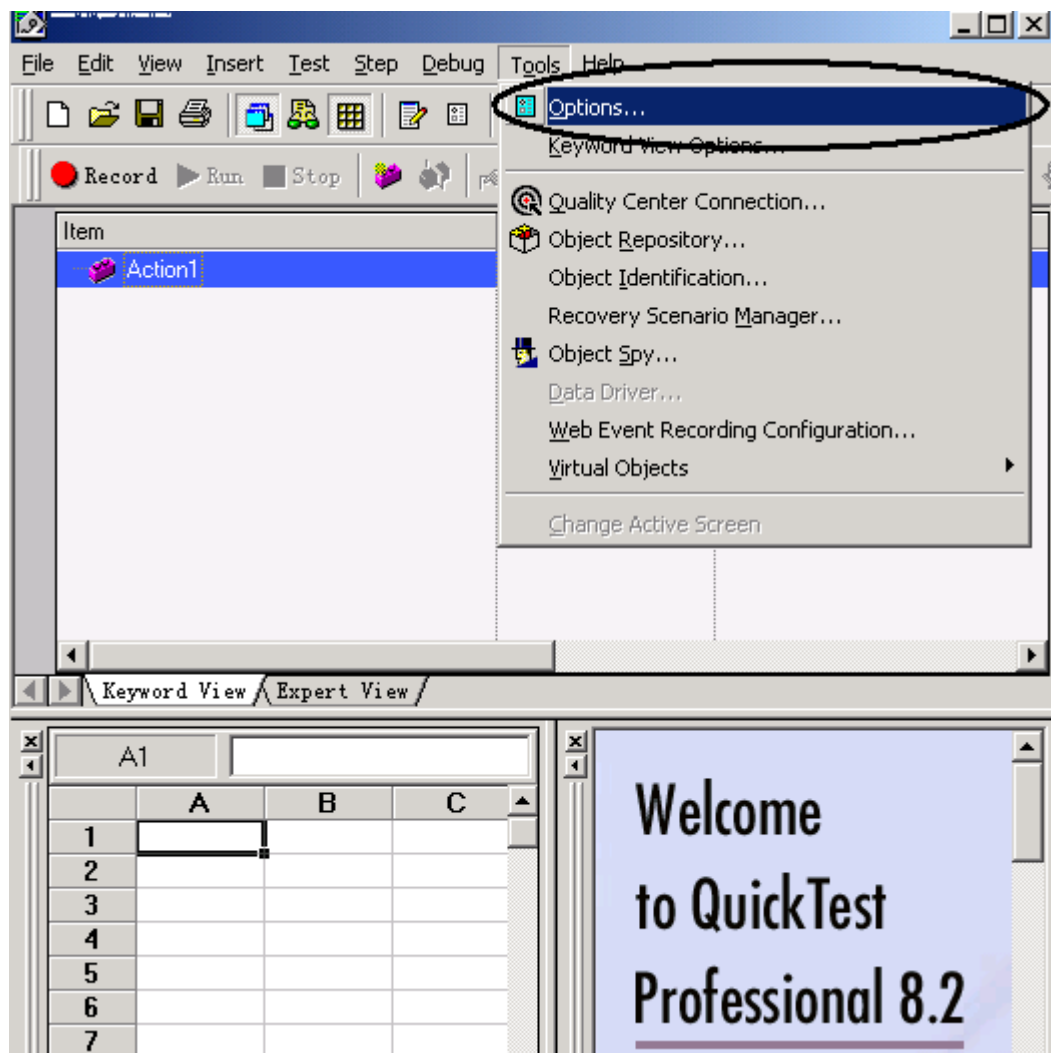


图 5.5.D1.1

弹出如下对话框

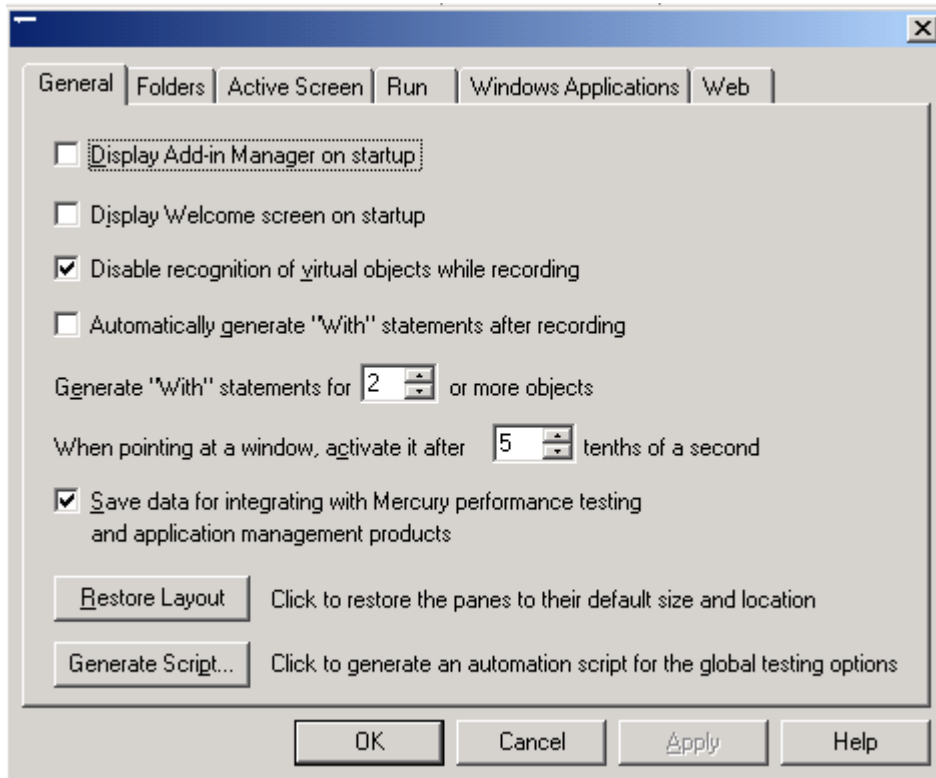


图 5.5.D1.2

选择 active screen tab

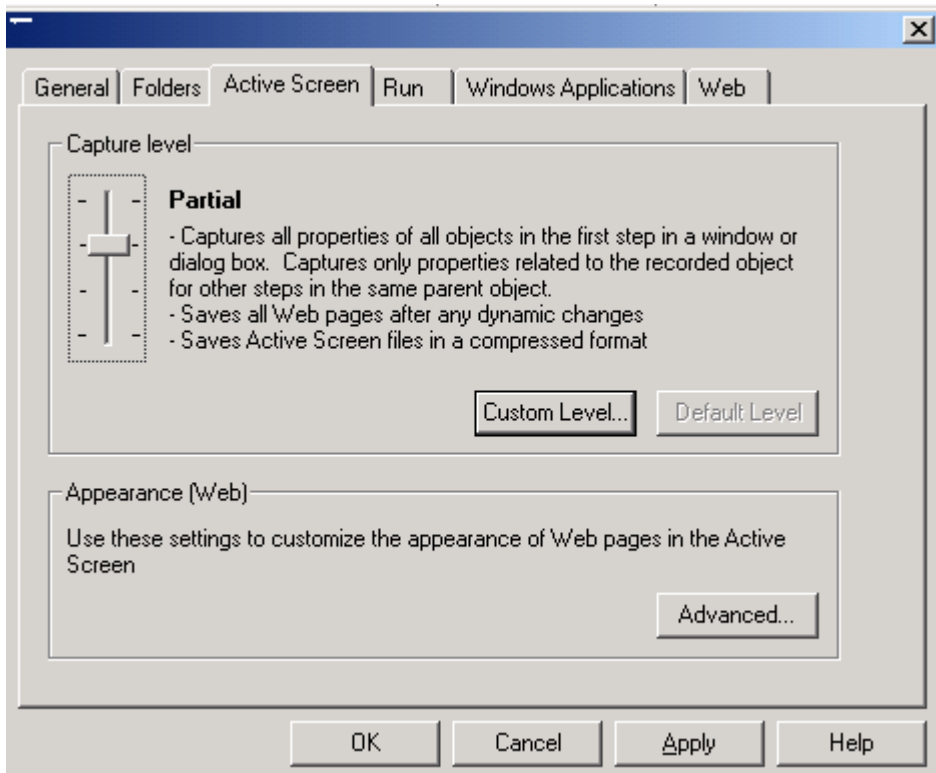


图 5.5.D1.3

分为 4 个级别 none,minimum,partial,complete

None: 不保存 active screen

Minimum: 最小化保存 active screen

Partial: 中级别保存 active screen

Complete: 最高级别保存 active screen

可以点击 custom level 来自定义,弹出如下对话框

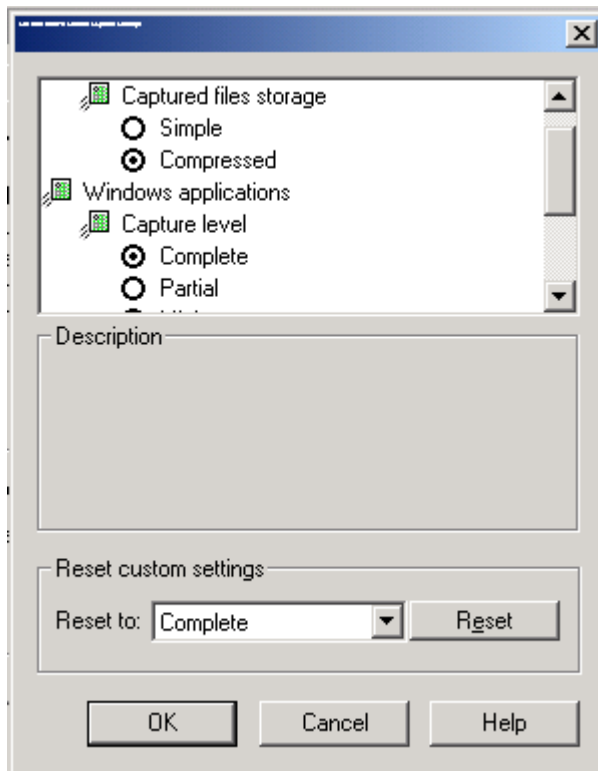


图 5.5.D1.4

可以根据你的需要自定义保存 active screen,选择好之后,点击 ok,设定被保存
D2 录制脚本,设定好之后,就可以录制脚本了,QTP 会根据你的设定保存 active screen
E 个人感受:无

5.6 QTP 录制方式的分类

A 问题描述:

QTP 录制有 3 中基本的录制方式,根据不同情况使用不同的录制方式

B 问题与解答: QTP 分为:基本录制,low-level record,analog record

B1 基本录制:

QTP 的默认录制模式,可以通过上面的 activescreen 来进行设置

B2 Low-level record:

低级别录制,录制级别高于基本录制,并且能够编辑它录制下来的脚本

B3 Analog record:

高级别录制,能够录制鼠标的动作,但是录制下来的脚本无法进行编辑和修改,如果页面中控件位置发生变化,则无法继续使用这个模式录制的脚本,当选用 analog record 模式之后会弹出如下对话框

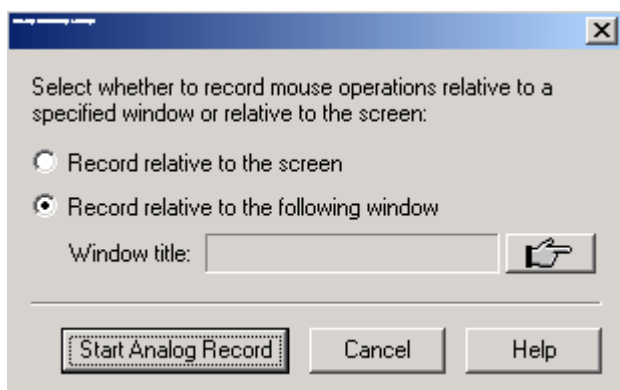


图 5.6.B3.1

选择 record relative to the screen 录制当前页,并且回放的时候也是在当前页回放,如果当前页不是录制时的当前页,那么将会出现错误

选择 record relative to the following window 可以指定回放时的界面,点击  按钮

指定界面,设定好后,点击  按钮

C 个人感受:无

6 关于 regular expressions

A 问题描述:

在 QTP 中可以是使用 regular expressions 来完善你的参数化

B 处理流程:

B1 参数化你的系统

B2 将参数化设定为 regular expressions

B3 在表格中使用 regular expressions

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具，他采用 B/S 模式，使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具，主要用于 B/S 模式的系统测试

D 实例:

D1 参数化你的系统,以参数 1980-2000 年中间相隔 20 年为例,需要比较 1980-1990 以及 1990-2000 之间的值,如果使用一般的参数化,那么需要设定 20 个参数 1980,1981,....2000,使用 regular expressions 则不需要这么麻烦

D2 在参数化时选为 regular expressions,选定你要参数化的步骤,参数化它在你参数化的过程中会遇到这如下对话框

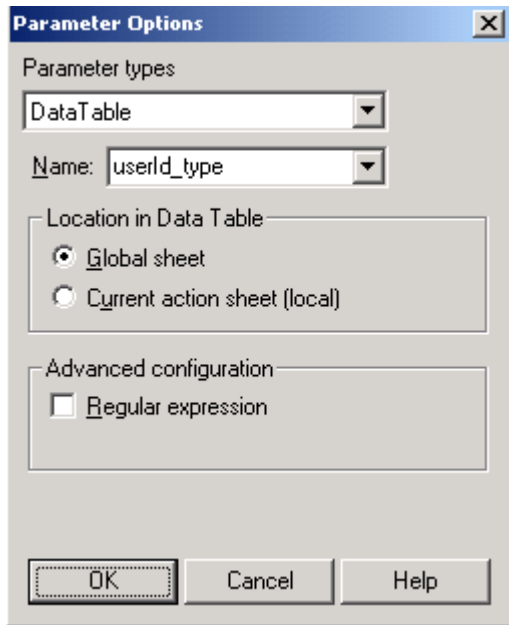


图 6.D2.1

选定 regular expression,点击 ok

在 QTP 的 table 中会出现你参数化的一列

	A1	text		
		userId_type	B	C
1		text		
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				

图 6.D2.2

D3 在表格中使用 regular expressions 以 "[","]" 为例,可以在 table 中输入 198[0-9],199[0-9],2000 如下图

A3		2000	
	userId_type	B	C
1	198[0-9]		
2	199[0-9]		
3	2000		
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			

图 6.D3.3

这样只用三行参数,就可以满足 20 行参数达到的要求

以上只是 regular expression 的一个例子, regular expression 还有类似”[]”的方便符号,详细可查阅 QTP user help,通过如上方法就能将这些符号应用到你的测试中

下面是对 regular expression 的一个简单介绍

如果我们问那些 UNIX 系统的爱好者他们最喜欢什么, 答案除了稳定的系统和可以远程启动之外, 十有八九的人会提到正则表达式; 如果我们再问他们最头痛的是什么, 可能除了复杂的进程控制和安装过程之外, 还会是正则表达式。那么正则表达式到底是什么? 如何才能真正的掌握正则表达式并正确的加以灵活运用? 本文将就此展开介绍, 希望能够对那些渴望了解和掌握正则表达式的读者有所助益。

入门简介

简单的说, 正则表达式是一种可以用于模式匹配和替换的强有力的工具。我们可以在几乎所有的基于 UNIX 系统的工具中找到正则表达式的身影, 例如, vi 编辑器, Perl 或 PHP 脚本语言, 以及 awk 或 sed shell 程序等。此外, 象 JavaScript 这种客户端的脚本语言也提供了对正则表达式的支持。由此可见, 正则表达式已经超出了某种语言或某个系统的局限, 成为人们广为接受的概念和功能。

正则表达式可以让用户通过使用一系列的特殊字符构建匹配模式, 然后把匹配模式与数据文件、程序输入以及 WEB 页面的表单输入等目标对象进行比较, 根据比较对象中是否包含匹配模式, 执行相应的程序

举例来说, 正则表达式的一个最为普遍的应用就是用于验证用户在线输入的邮件地址的格式是否正确。如果通过正则表达式验证用户邮件地址的格式正确, 用户所填写的表单信息将会被正常处理; 反之, 如果用户输入的邮件地址与正则表达的模式不匹配, 将会弹出提示信息, 要求用户重新输入正确的邮件地址。由此可见正则表达式在 WEB 应用的逻辑判断中具有举足轻重的作用。

\ 基本语法

在对正则表达式的功能和作用有了初步的了解之后, 我们就来具体看一下正则表达式的语法格式。

正则表达式的形式一般如下:

/love/

其中位于“/”定界符之间的部分就是将要在目标对象中进行匹配的模式。用户只要把希望查找匹配对象的模式内容放入“/”定界符之间即可。为了能够使用户更加灵活的定制模式内容，正则表达式提供了专门的“元字符”。所谓元字符就是指那些在正则表达式中具有特殊意义的专用字符，可以用来规定其前导字符（即位于元字符前面的字符）在目标对象中的出现模式。

较为常用的元字符包括：“+”，“*”，以及“?”。其中，“+”元字符规定其前导字符必须在目标对象中连续出现一次或多次，“*”元字符规定其前导字符必须在目标对象中出现零次或连续多次，而“?”元字符规定其前导对象必须在目标对象中连续出现零次或一次。

下面，就让我们来看一下正则表达式元字符的具体应用。

`/fo+/`

因为上述正则表达式中包含“+”元字符，表示可以与目标对象中的“fool”，“fo”，或者“football”等在字母 f 后面连续出现一个或多个字母 o 的字符串相匹配。

`/eg*/`

因为上述正则表达式中包含“*”元字符，表示可以与目标对象中的“easy”，“ego”，或者“egg”等在字母 e 后面连续出现零个或多个字母 g 的字符串相匹配。

`/Wil?/`

因为上述正则表达式中包含“?”元字符，表示可以与目标对象中的“Win”，或者“Wilson”，等在字母 i 后面连续出现零个或一个字母 l 的字符串相匹配。

除了元字符之外，用户还可以精确指定模式在匹配对象中出现的频率。例如，

`/jim{2,6}/`

上述正则表达式规定字符 m 可以在匹配对象中连续出现 2-6 次，因此，上述正则表达式可以同 jimmy 或 jimmmmy 等字符串相匹配。

在对如何使用正则表达式有了初步了解之后，我们来看一下其它几个重要的元字符的使用方式。

`\s`: 用于匹配单个空格符，包括 tab 键和换行符；

`\S`: 用于匹配除单个空格符之外的所有字符；

`\d`: 用于匹配从 0 到 9 的数字；

`\w`: 用于匹配字母，数字或下划线字符；

`\W`: 用于匹配所有与 \w 不匹配的字符；

`:` 用于匹配除换行符之外的所有字符。

（说明：我们可以把 \s 和 \S 以及 \w 和 \W 看作互为逆运算）

下面，我们就通过实例看一下如何在正则表达式中使用上述元字符。

`^s+ /`

上述正则表达式可以用于匹配目标对象中的一个或多个空格字符。

`^d000 /`

如果我们手中有一份复杂的财务报表，那么我们可以通过上述正则表达式轻而易举的查找到所有总额达千元的款项。

除了我们以上所介绍的元字符之外，正则表达式中还具有另外一种较为独特的专用字符，即定位符。定位符用于规定匹配模式在目标对象中的出现位置。

较为常用的定位符包括：“^”，“\$”，“\b”以及“\B”。其中，“^”定位符规定匹配模式必须出现在目标字符串的开头，“\$”定位符规定匹配模式必须出现在目标对象的结尾，\b 定位符规定匹配模式必须出现在目标字符串的开头或结尾的两个边界之一，而“\B”定位符则规定匹配对象必须位于目标字符串的开头和结尾两个边界之内，即匹配对象既

不能作为目标字符串的开头，也不能作为目标字符串的结尾。同样，我们也可以把“^”和“\$”以及“\b”和“\B”看作是互为逆运算的两组定位符。举例来说：

```
/^hell/
```

因为上述正则表达式中包含“^”定位符，所以可以与目标对象中以“hell”，“hello”或“hellhound”开头的字符串相匹配。

```
/ar$/
```

因为上述正则表达式中包含“\$”定位符，所以可以与目标对象中以“car”，“bar”或“ar”结尾的字符串相匹配。

```
^bbom/
```

因为上述正则表达式模式以“\b”定位符开头，所以可以与目标对象中以“bomb”，或“bom”开头的字符串相匹配。

```
/man\b/
```

因为上述正则表达式模式以“\b”定位符结尾，所以可以与目标对象中以“human”，“woman”或“man”结尾的字符串相匹配。

为了能够方便用户更加灵活的设定匹配模式，正则表达式允许使用者在匹配模式中指定某一个范围而不局限于具体的字符。例如：

```
[A-Z]/
```

上述正则表达式将会与从 A 到 Z 范围内任何一个大写字母相匹配。

```
[a-z]/
```

上述正则表达式将会与从 a 到 z 范围内任何一个字母相匹配。

```
[0-9]/
```

上述正则表达式将会与从 0 到 9 范围内任何一个数字相匹配。

```
/([a-z][A-Z][0-9]+)/
```

上述正则表达式将会与任何由字母和数字组成的字符串，如“aB0”等相匹配。这里需要提醒用户注意的一点就是可以在正则表达式中使用“()”把字符串组合在一起。“()”符号包含的内容必须同时出现在目标对象中。因此，上述正则表达式将无法与诸如“abc”等的字符串匹配，因为“abc”中的最后一个字符为字母而非数字。

如果我们希望在正则表达式中实现类似编程逻辑中的“或”运算，在多个不同的模式中任选一个进行匹配的话，可以使用管道符“|”。例如：

```
/to|too|2/
```

上述正则表达式将会与目标对象中的“to”，“too”，或“2”相匹配。

正则表达式中还有一个较为常用的运算符，即否定符“[]”。与我们前文所介绍的定位符“^”不同，否定符“[]”规定目标对象中不能存在模式中所规定的字符串。例如：

```
/[^A-C]/
```

上述字符串将会与目标对象中除 A，B，和 C 之外的任何字符相匹配。一般来说，当“^”出现在“[]”内时就被视做否定运算符；而当“^”位于“[]”之外，或没有“[]”时，则应当被视做定位符。

最后，当用户需要在正则表达式的模式中加入元字符，并查找其匹配对象时，可以使用转义符“\”。例如：

```
/Th\*/
```

上述正则表达式将会与目标对象中的“Th*”而非“The”等相匹配。

E 个人感受:无

7 关于 datatable

7.1 global 与 current 的区别

A 问题描述:

在我们参数化的过程中总会遇到让我们选择 global 或者 current Action 的情况,下面对这两种模式作相应的介绍

B 问题解答:

Global 全局参数:

他控制整个 Action 的运行次数,global 里有几行参数,则运行几次脚本

Current Action:

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

在前面介绍的多个 Action 共同运行时,实际我们已经介绍了他们的使用方法

以登陆系统,添加信息,再退出为例,登陆系统的用户名,密码是 global 参数,他控制这个脚本的循环次数,为了达到一次登陆多次添加的目的,将登陆名密码设为一行,而在添加 Action 中将其中添加的那些信息参数化后,将这些参数设为 current Action,这控制当运行到这个 Action 时,一共运行几次这个 Action

当设定好后,实际运行一遍脚本,但是,多次运行其中的添加 Action
C 个人感受:无

7.2 import excel 的问题

A 问题描述:

在使用 QTP 参数化时,我们可以使用在以前项目中已经设定好的参数,将这些参数保存为 excel 格式,并在使用时将他们导入到 QTP 的 table 中

B 处理流程:

- B1 录制脚本
- B2 设置参数
- B3 导入参数

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,录制你要的脚本

D2 设置参数,无需特殊设置

D3 导入数据,在 table 中点击右键选择 file-import from file 如下图

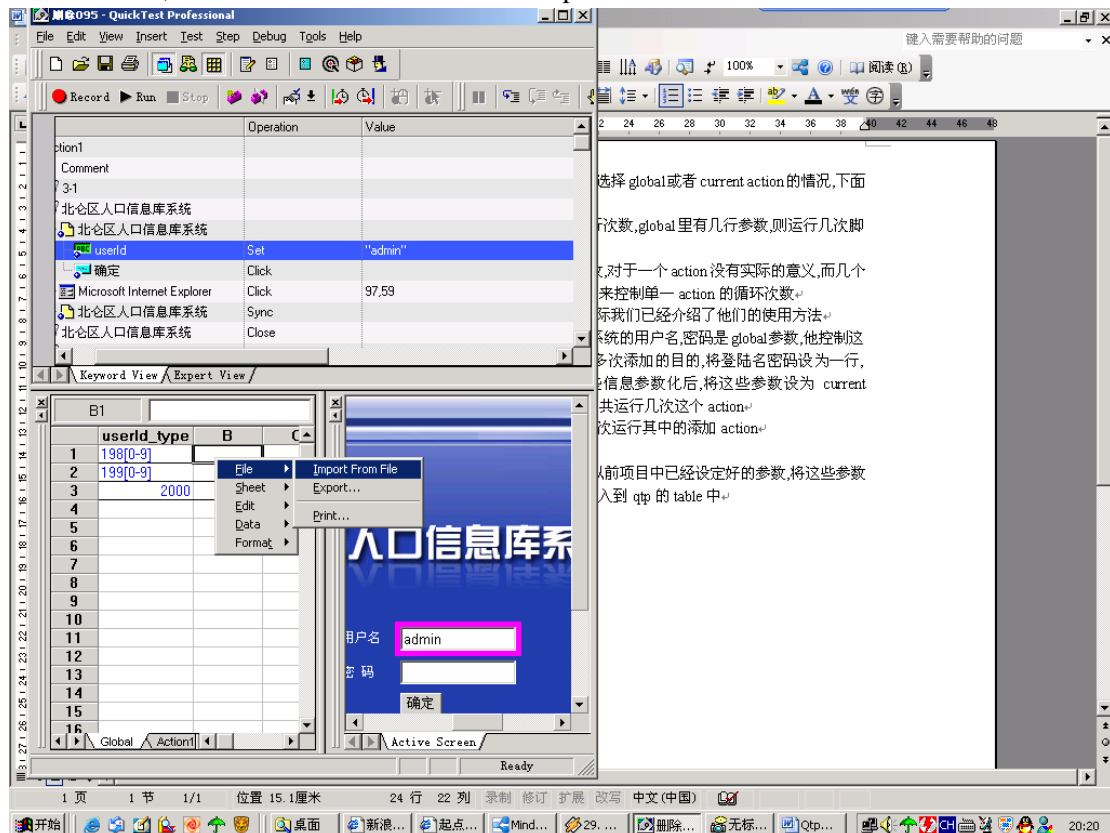


图 7.2.D3.1

QTP 会给你一个提示信息,表示你现在表格中的名称 B 将被改成导入 excel 表中的名称



图 7.2.D3.2

点击确定,进入如下对话框,选择你的 excel 表格

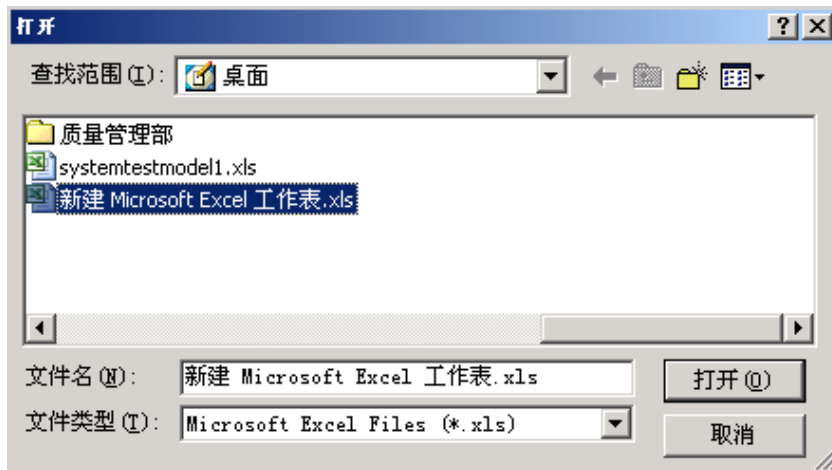


图 7.2.D3.3

打开,QTP 如下图

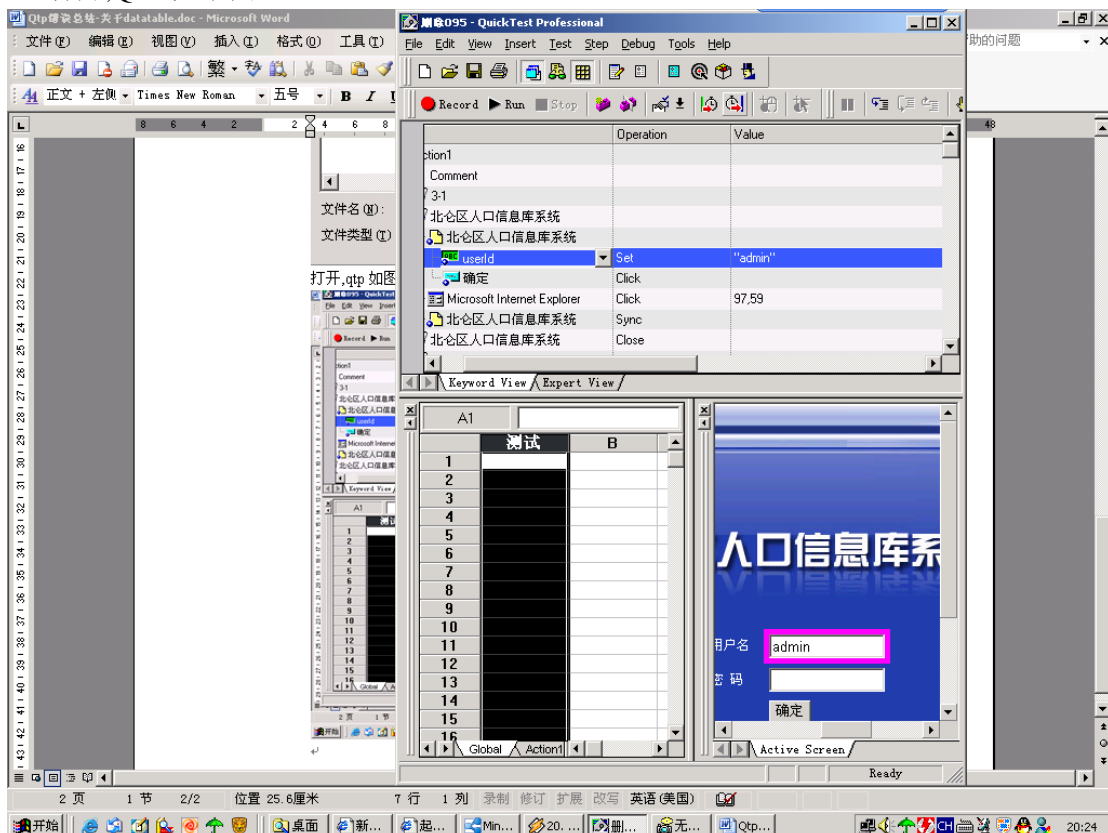


图 7.2.D3.4

原先的 a 列被改为测试列

提示: 使用这个方法将使 excel 中的所有信息都导入到 QTP 中,如果 a 的位置已经有数据,数据将被 excel 中的数据覆盖

E 个人感受:无

7.3 关于 editing the data table

A 问题描述:

当参数化完成后,很多情况下需要对参数进行修改

B 处理流程:

B1 录制脚本

B2 参数化脚本

B3 在 table 中修改参数

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,查看 QTP user help

D2 参数化脚本,查看 QTP user help

D3 在 table 中修改参数,在 table 中点击右键,选择 edit,如下图

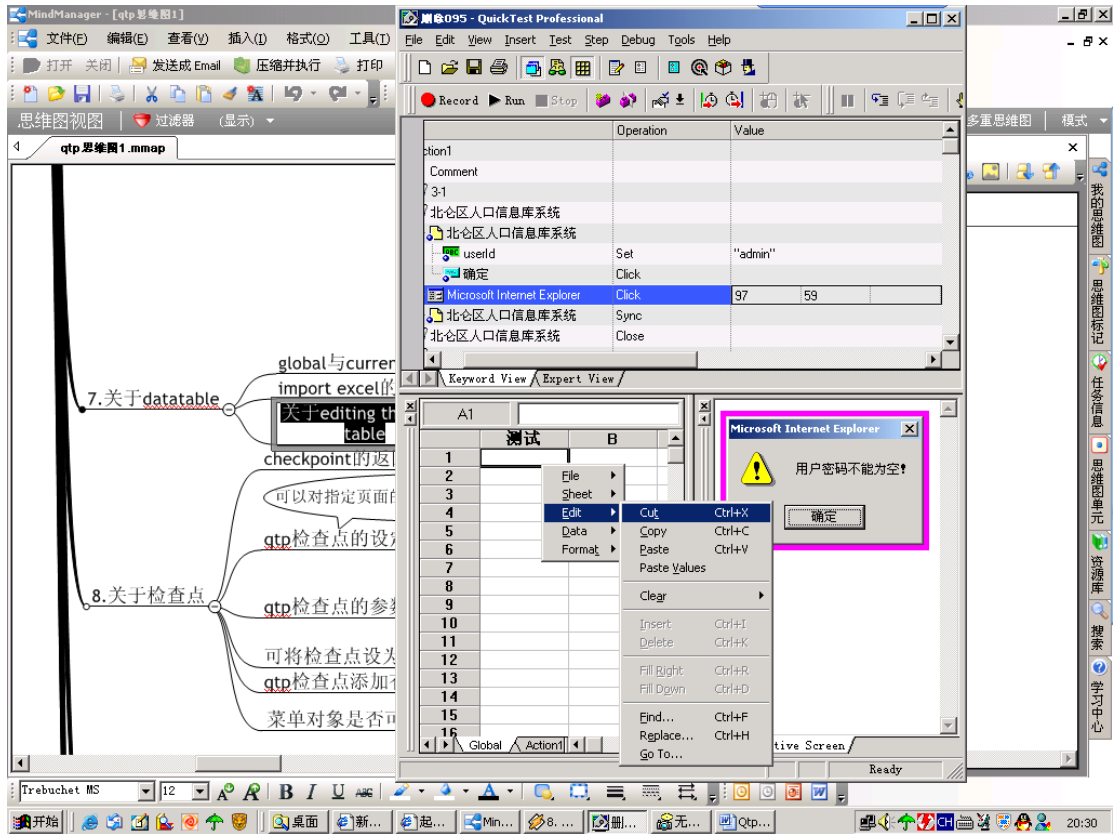


图 7.3.D3.1

提示:在这里你可以对参数进行删除和修改,如果想把这个参数行彻底删除,则选中该行,点击右键,并且选择 delete

E 个人感受:无

8 关于检查点

8.1 checkpoint 的返回值

A 问题描述:

在一些特定情况下,脚本需要知道检查点是否通过,这时候就需要通过检验检查点的返回值来判断检查点是否通过

B 处理流程:

B1 录制脚本,设定检查点

B2 在 expert view 中设定检查点

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,设定检查点,如下图

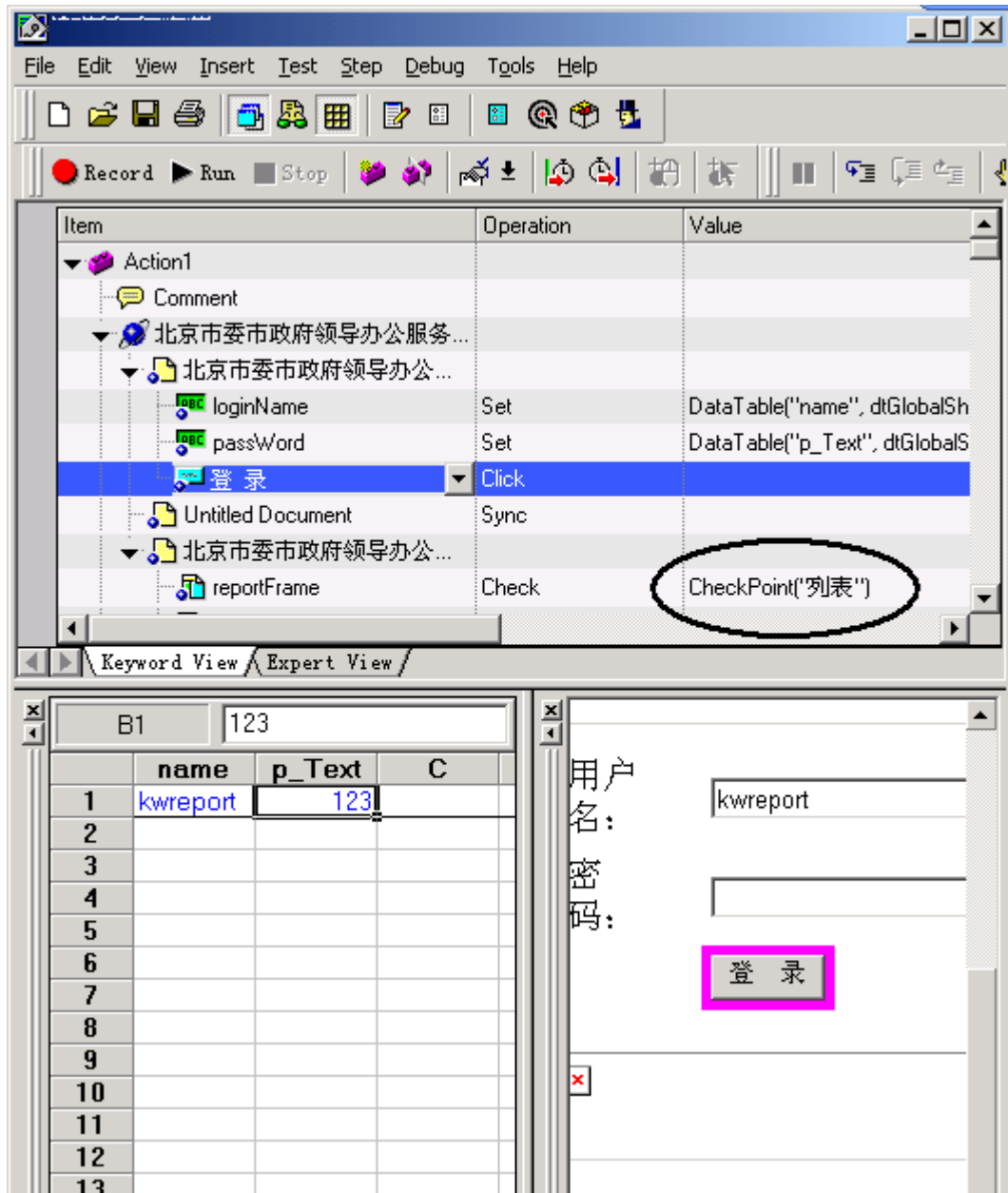


图 8.1.D1.1

D2 在 expert vIEW 中设定检查点,点击 expert vIEW,进入 expert vIEW 看到如下图所示代码

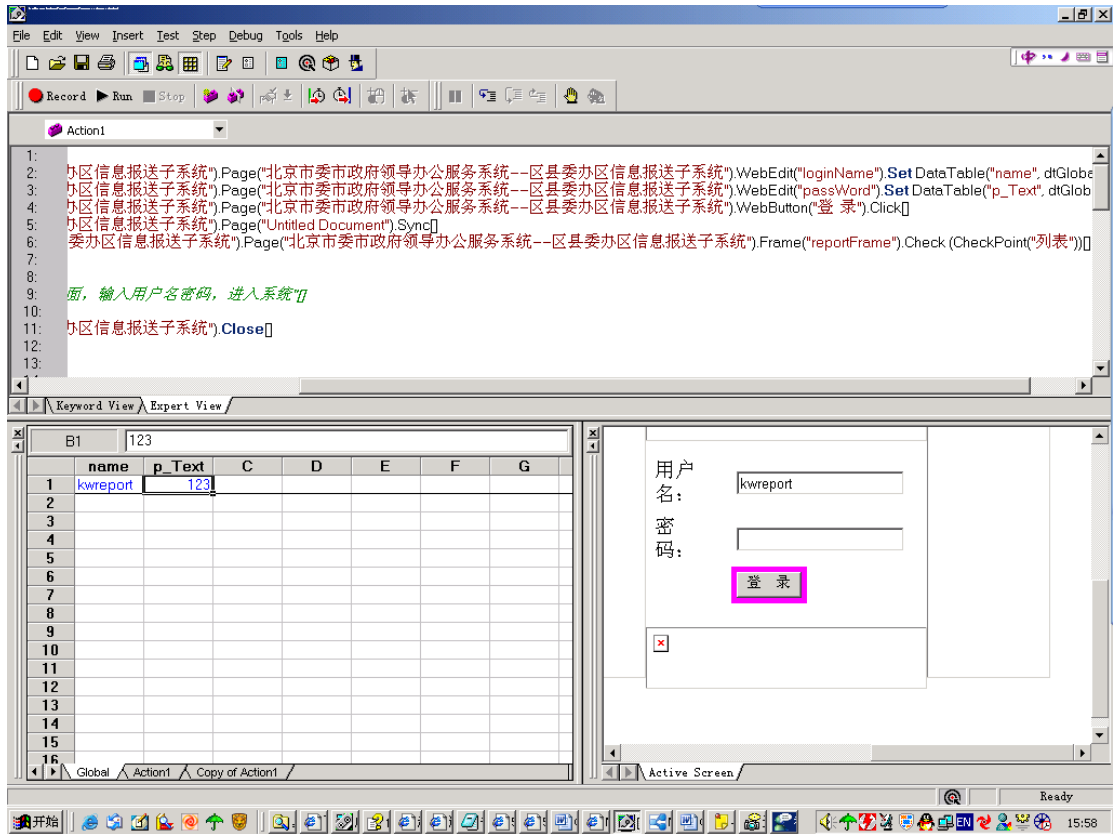


图 8.1.D2.2

代码是 check checkpoint(),你需要在 check 后面加一个(),使代码变为 check .(checkpoint()) 并且在整行的前面加上一个变量(变量无需提前定义)如

a = Browser("").Page("").Frame("reportFrame").Check (CheckPoint("列表"))

这样检查点的返回值就可以被 a 记录下来,那么就可以通过判断 a 的值来确定检查点是否通过

E 个人感受:无

8.2 QTP 检查点的设定

A 问题与解答:

QTP 通过检查点可以检查流程是否正确,但如果在一个脚本中同事检查正确或者不正确,就会遇到问题,这是就需要分开录制脚本

B 实例:

以一个系统得登陆操作为例,就需要录制两个脚本,分别是登陆成功,登陆失败,不同得检查点,不同的参数化

提示:在检查登陆时,登陆成功,可以检查登陆成功后得一些特殊得图像,或者文字,登陆失败则要检查错误提示

C 个人感受:无

8.3 关于文字检查点得参数化

A 问题描述:

在建立文字检查点后,参数化检查点,但是随着每次循环环境得不通,检查点得可能无法正确检查到文字,这种情况是因为检查点两边得限制条件发生变化的原因造成得解决这个问题,就需要对文字检查点的限制条件也进行参数化

B 处理流程:

B1 录制脚本,设定文字检查点

B2 参数化文字检查点

B3 参数化文字检查点两端的限定条件 test before 和 test after

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具, 主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,设定文字检查点详见 QTP 使用手册

D2 参数化文字检查点,对于文字检查点,可以通过参数化这个方法,通过一个检查点,检查多个情况

当添加文字检查点后,点击这个检查点会弹出如下对话框

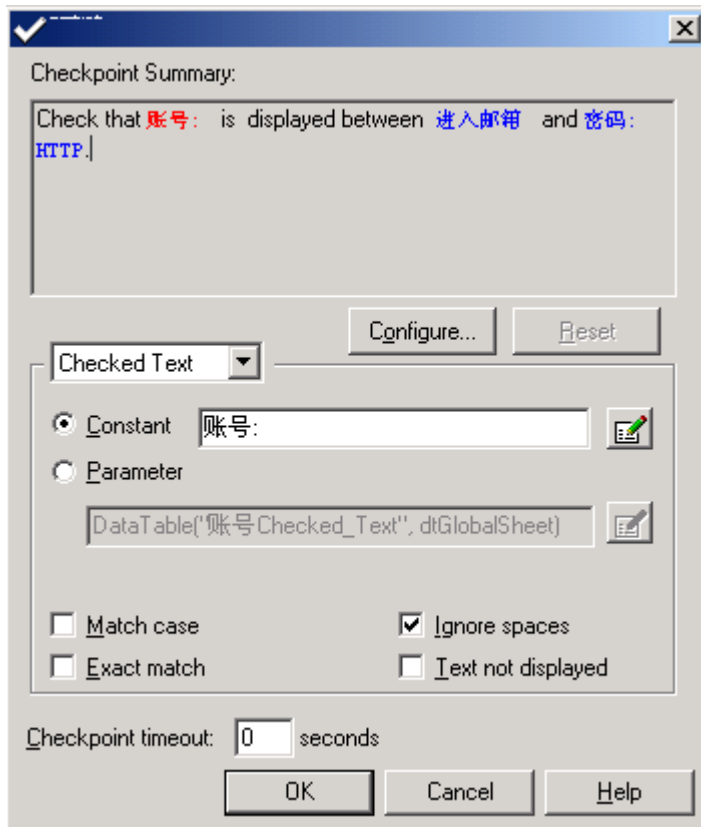


图 8.3.D2.1

在这里可以对文字检查点作一些设定,点击 configure 按钮弹出如下对话框

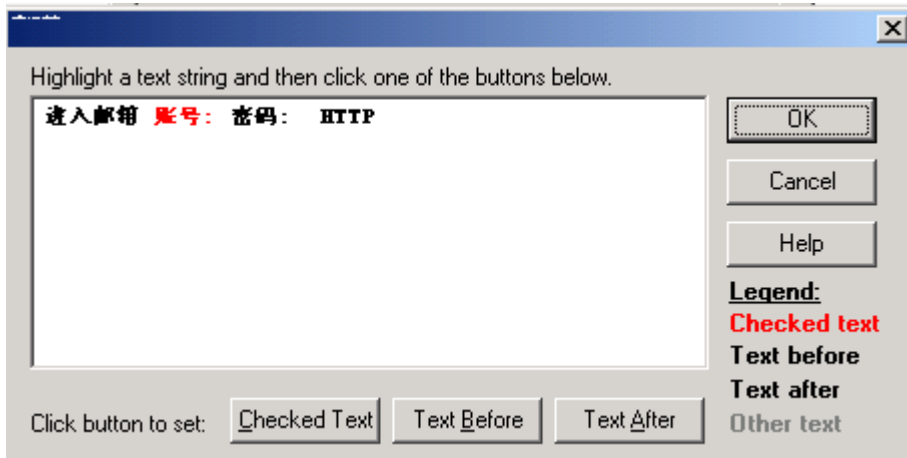


图 8.3.D2.2

在这里你可以对检查点作简单得设定,选中字体,然后点击 checked text 这些文字被设定为被检查对象,点击 text before,这些文字被设定为限定条件 text before ,text after 同 before 一样

提示: 当选定中间的文字为 text before 或者 text after 时他们前面或者后面的文字就会消失,且不能重现

如我选定邮箱为 text before

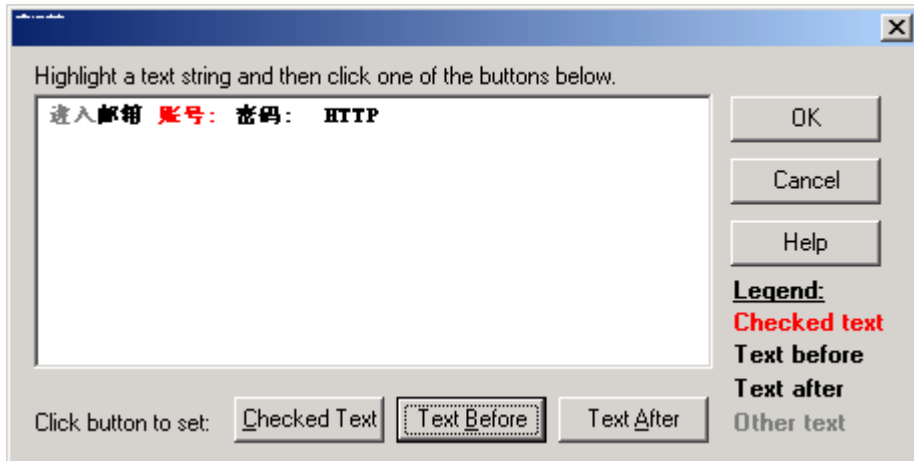


图 8.3.D2.3

进入两个字变为灰体,点击 ok

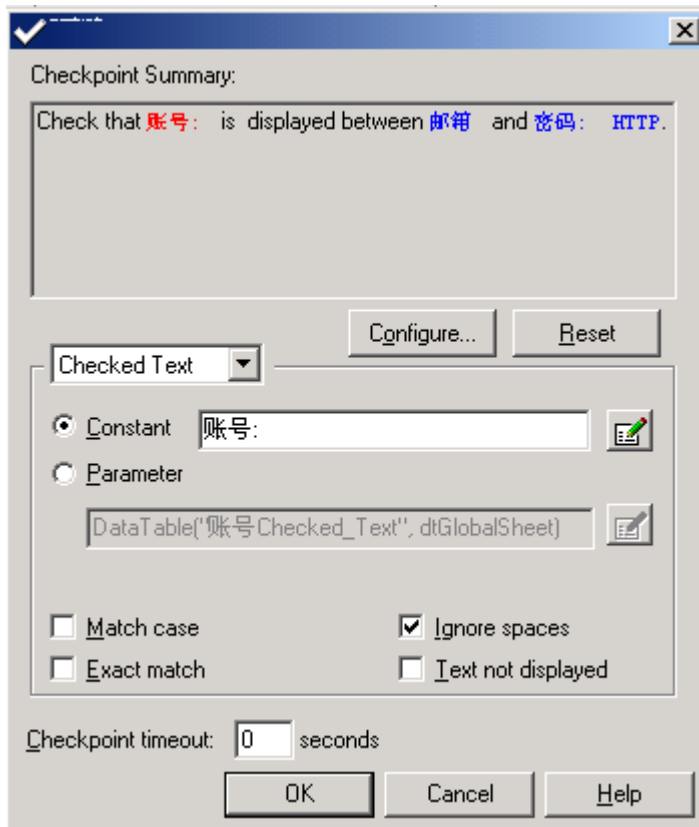


图 8.3.D2.4

再次点击 configure,如下图

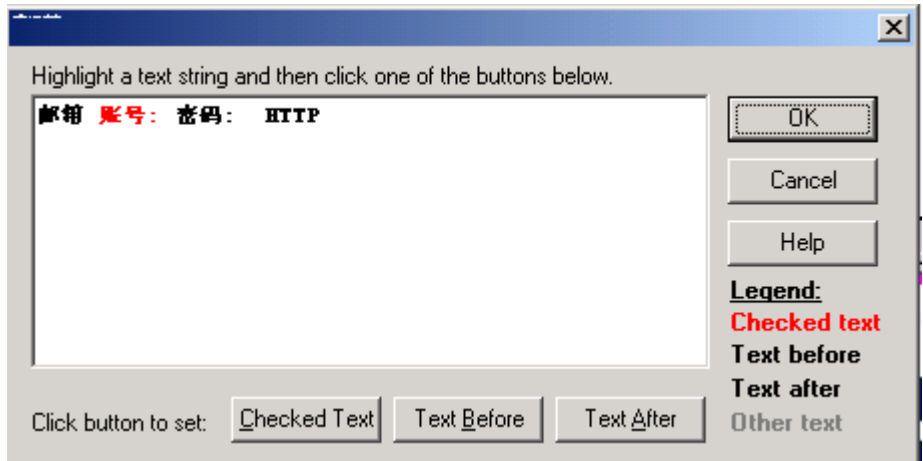


图 8.3.D2.5

进入两字不再出现
在对话框的下半部分中

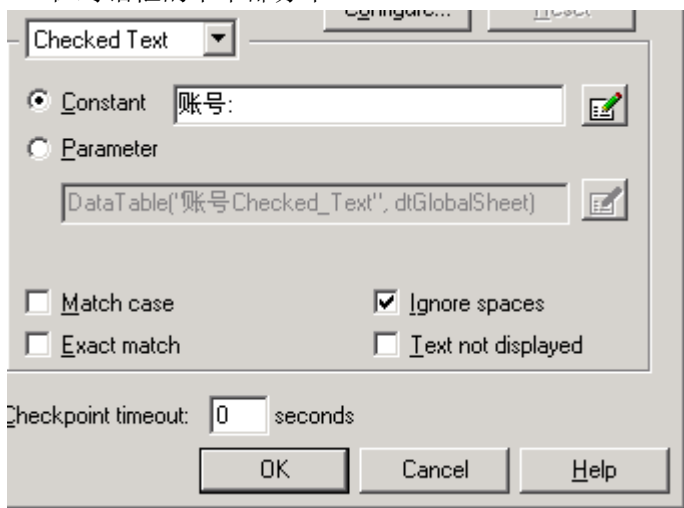


图 8.3.D2.6

你可以对 checked text 作详细的设定以及参数化,详见 QTP user help
D3 参数化文字检查点两端的限定条件 test before 和 test after,当你设定好 checked text 后,
还需要对 text before 和 text after 作一些设定,你可以通过选择如下图的下来菜单

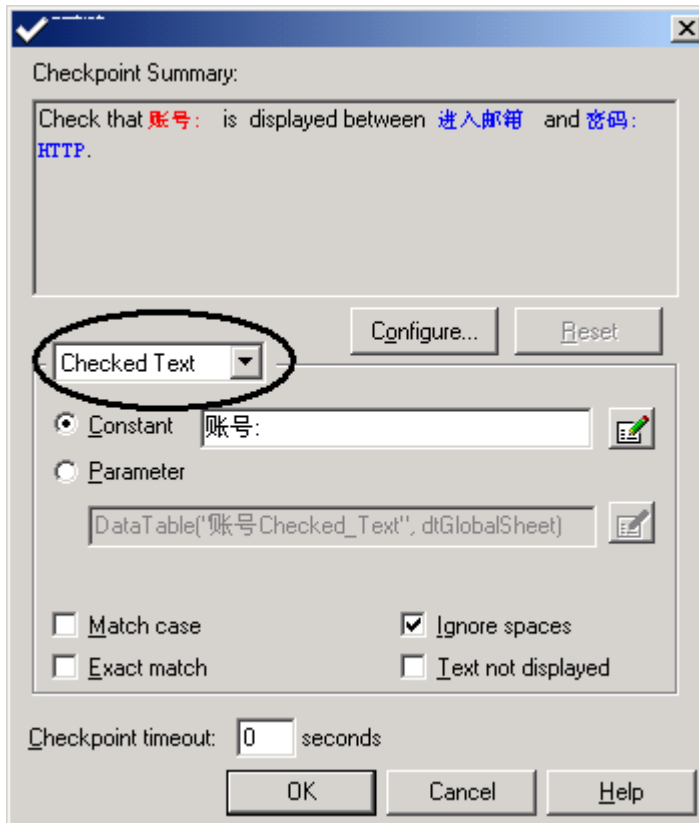


图 8.3.D3.7

来进入 text before 或者 text after 的对话框

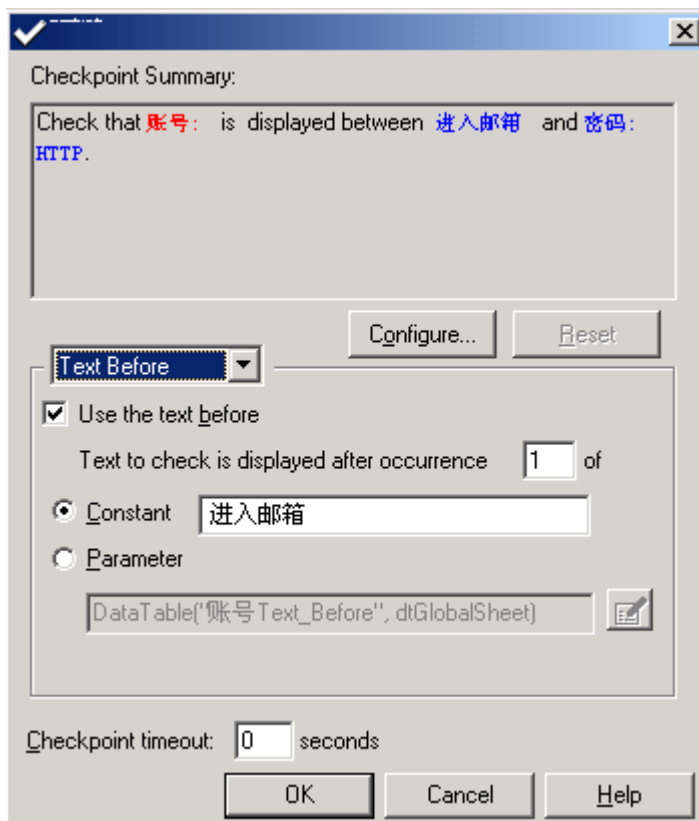


图 8.3.D3.8

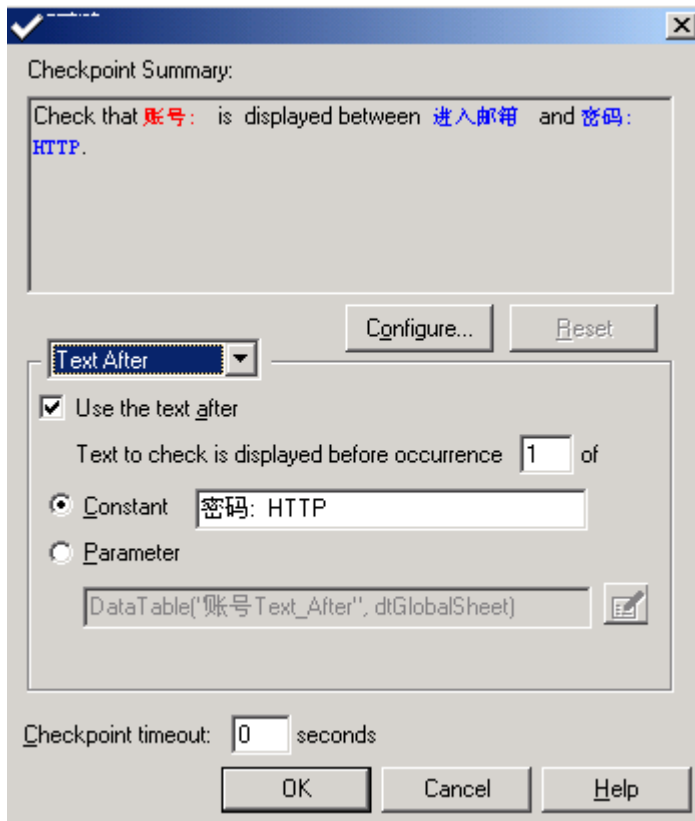


图 8.3.D3.9

当你不想使用 text before, text after 时可以将 Use the text after 单选框, 勾没, 这样 QTP 就会在整个页面中寻找你的 check text

Text to check is displayed before occurrence of 限定 checked text 出现的次数

E 个人感受: 无

8.4 设立可重用检查点

A 问题描述:

每个脚本都需要设定检查点,但有些脚本的检查点可能相同,这时重复设定检查点就会使工作没有实际意义,这时可以设立可重用检查点来解决这个问题

B 处理流程:

- B1 录制脚本,不设立检查点
- B2 录制可重用检查点
- B3 调用可重用检查点

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,不设立检查点

D2 录制可重用检查点,将 QTP record and run setting 设为录制当前页,开始录制,不录制步骤,直接在录制过程中添加检查点,将这个只有检查点的 Action 设为可重用 Action 如下图

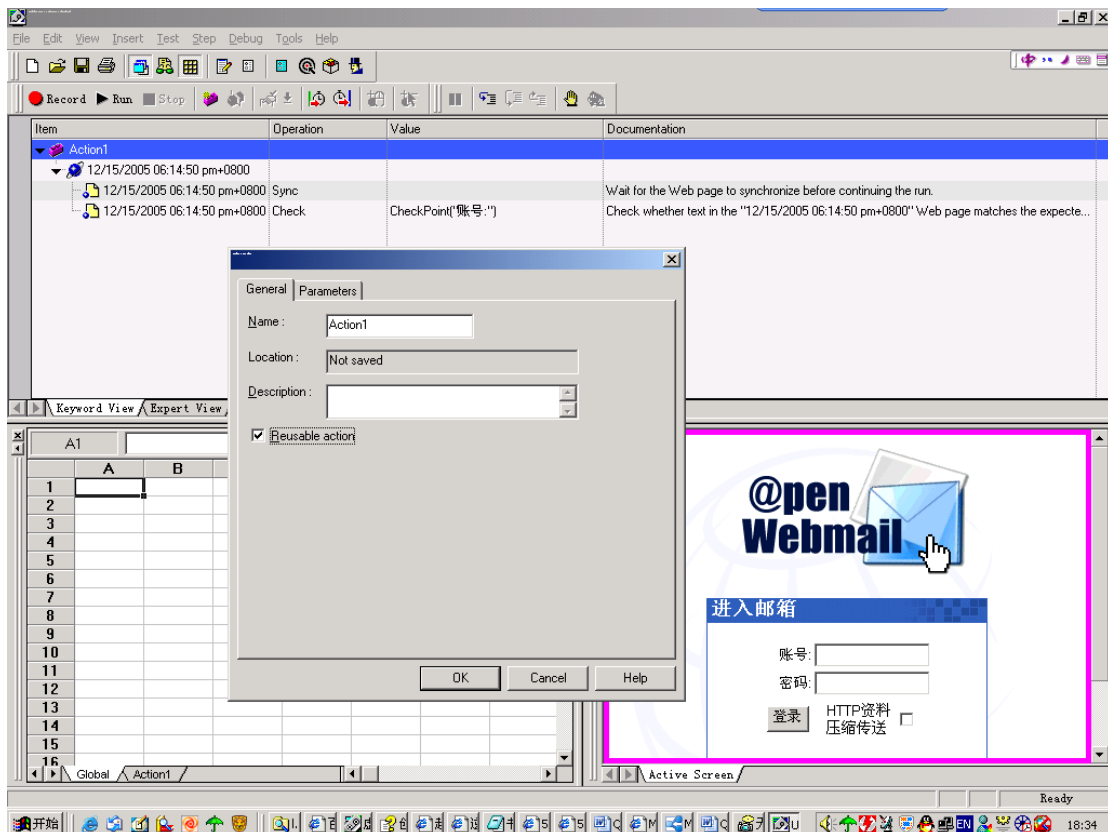


图 8.4.D2.1

D3 调用可重用检查点,在第一步录制好的脚本中调用这个可重用检查点,首先选中需要添加检查点的步骤,然后选择 insert—call existing Action 但是选择 after current step

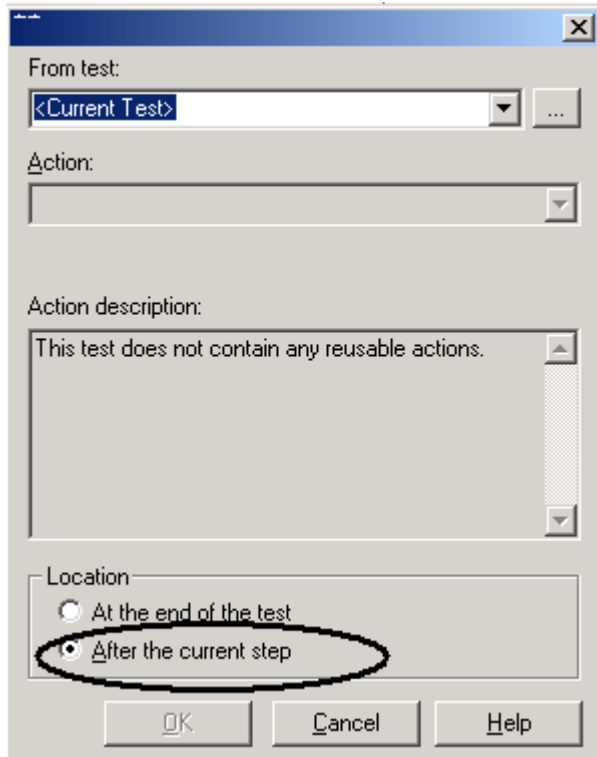


图 8.4.D3.2

检查点添加到步骤之后,通过这种发放这个检查点可以被多次使用
E 个人感受:无

8.5 QTP 检查点添加有问题,造成无法识别控件,导致脚本无法运行的问题

A 问题与解答:

可以在录制时添加文字检查点,虽然这个检查点在运行时还是会出错,但是我们可以通过找个检查点对于控件作不做任何操作的记录,添加检查点之后,这个控件上的信息将被记录下来,可以按照常规方法添加文字检查点,再将录制时添加的检查点删除即可

B 个人感受: 无

8.6 标准检查点

A 问题与解答:

个人认为标准检查点是 QTP 最方便实用的检查点,检查点捕捉控件的所有可检查的属性,并且选择其中认为应该检查的加以检查,你可以通过手工设定,增加或者减少检查的属性如下图

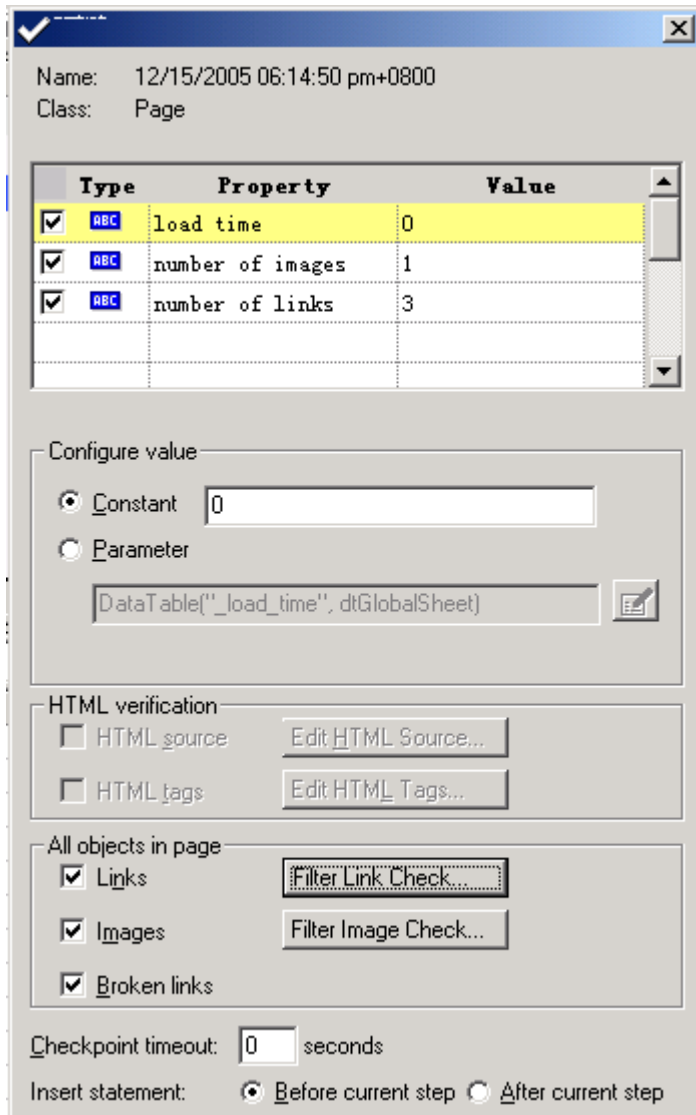


图 8.6.A.1

可以通过点击 和 按钮设定检查的具体内容

点击 , 弹出如下对话框

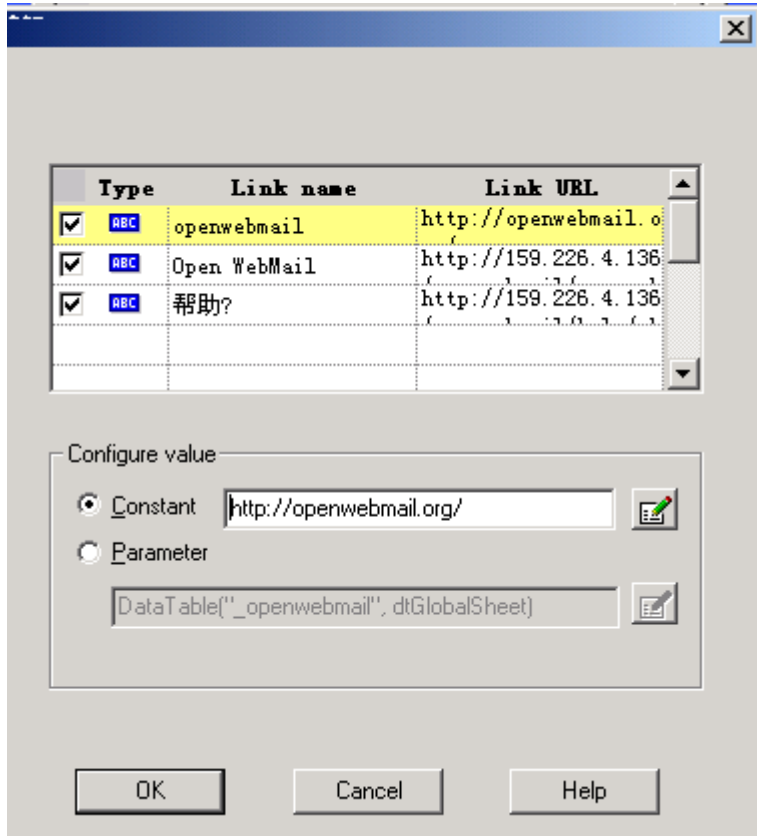


图 8.6.A.2

可以对 link 的具体属性作修改

点击 **Filter Image Check...** 弹出如下对话框

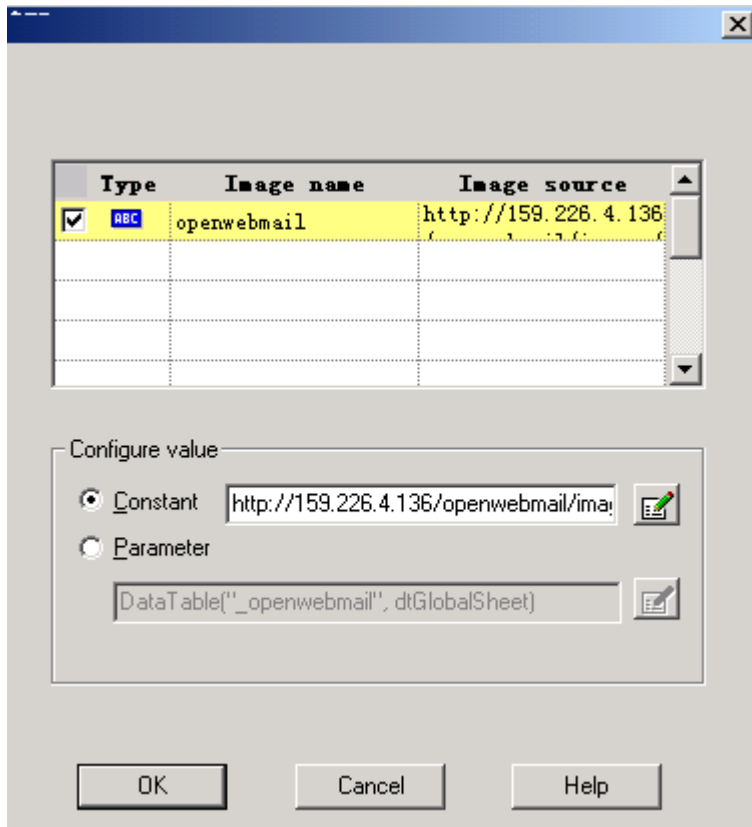


图 8.6.A.3

也可以对 imag 作修改
可以通过

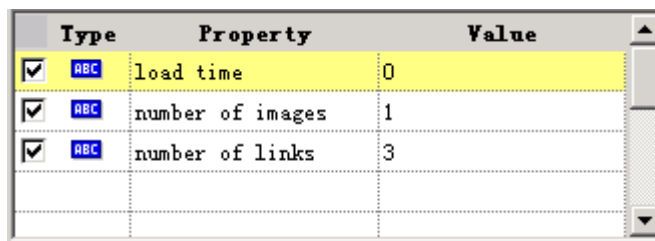


图 8.6.A.4

来选择其中要检查的控件属性

- B 个人感受:标准检查点是非常常用的检查点,一般的问题,标准检查点都能解决,建议多使用标准检查点

9 关于错误报告的问题

9.1 关于 report.ReportEvent

A 问题描述:

当设定好检查点后,QTP 能够自动向 TD 中添加错误报告,但是有的时候 QTP 自动提交的错误报告,并不能满足你的要求,这时就需要使用一些自定义的错误报告

B 处理流程:

B1 录制脚本

B2 添加自定义错误

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具，他采用 B/S 模式，使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具，主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,详细查看 QTP user help 使用说明

D2 添加自定义错误,在 QTP 主界面选择 insert---step---reporter

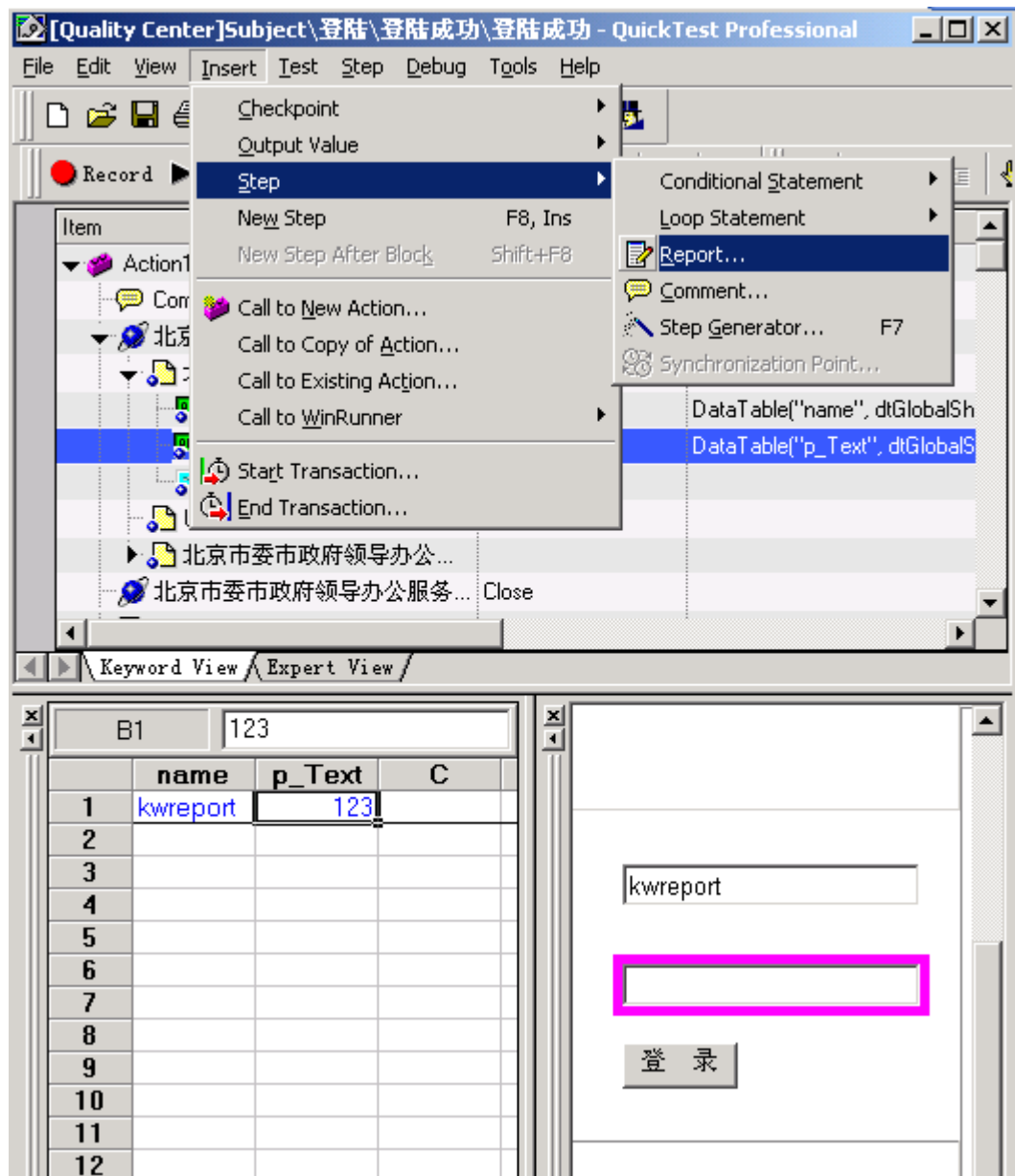


图 9.1.D2.1

弹出如下对话框

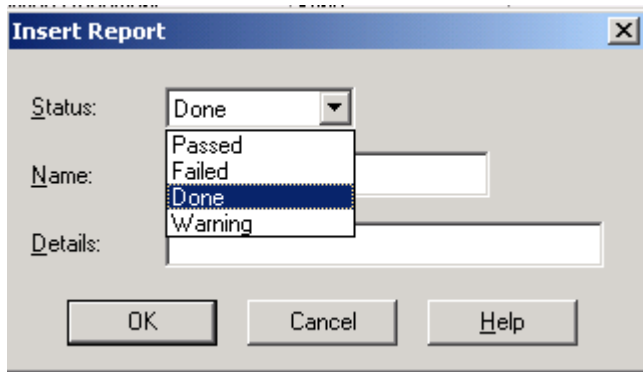


图 9.1.D2.2

页面中包含三个内容: status name details

Status: 报告的身份,本问题介绍的报告,不单可以用在错误报告上,也可以用在其他的报告上,比如 passed,done warning,failed

提示: 如果你选择了 failed 那么即使脚本正常通过,检查点也正确,报告也显示为 failed 并且自动提交到 TD 中,使用条件语句能够达到判断是否向 TD 中添加错误的目的

Name: 报告的名称

Details: 报告的细节,你可以在这里写清脚本或者错误的详细描述

设定完成后,点击 ok

添加的 reporter 会显示在你选择的脚本,或者检查点后,如下图

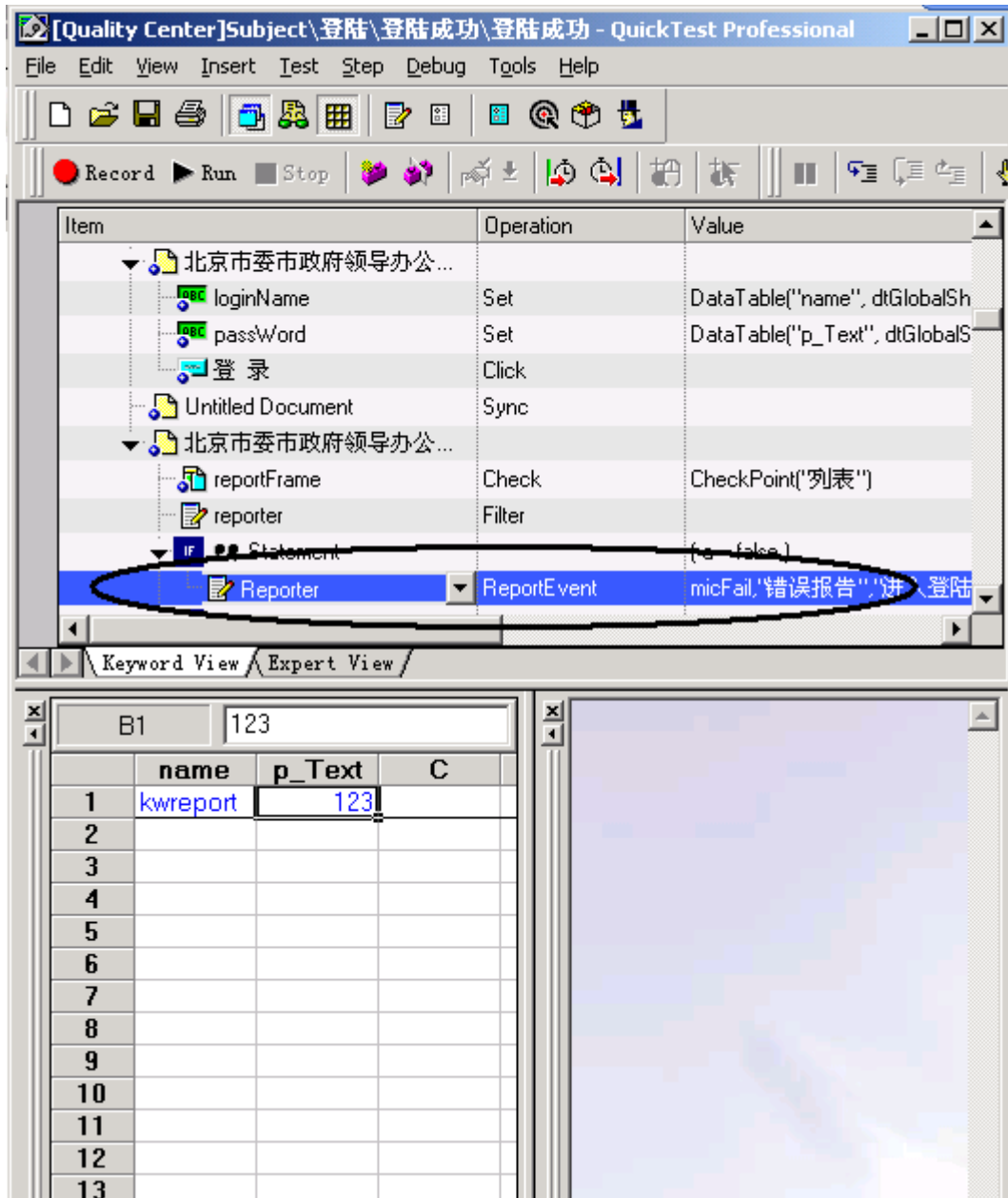


图 9.1.D2.3

你也可以在 expert view 中通过代码添加错误报告

使用 report.ReportEvent, 格式为 report.ReportEvent status, "name", "details"

Status 的设定分为四种:

- 0 或者 micpass
- 1 或者 micfail
- 2 或者 micdone
- 3 或者 micwarning

例子如下图:

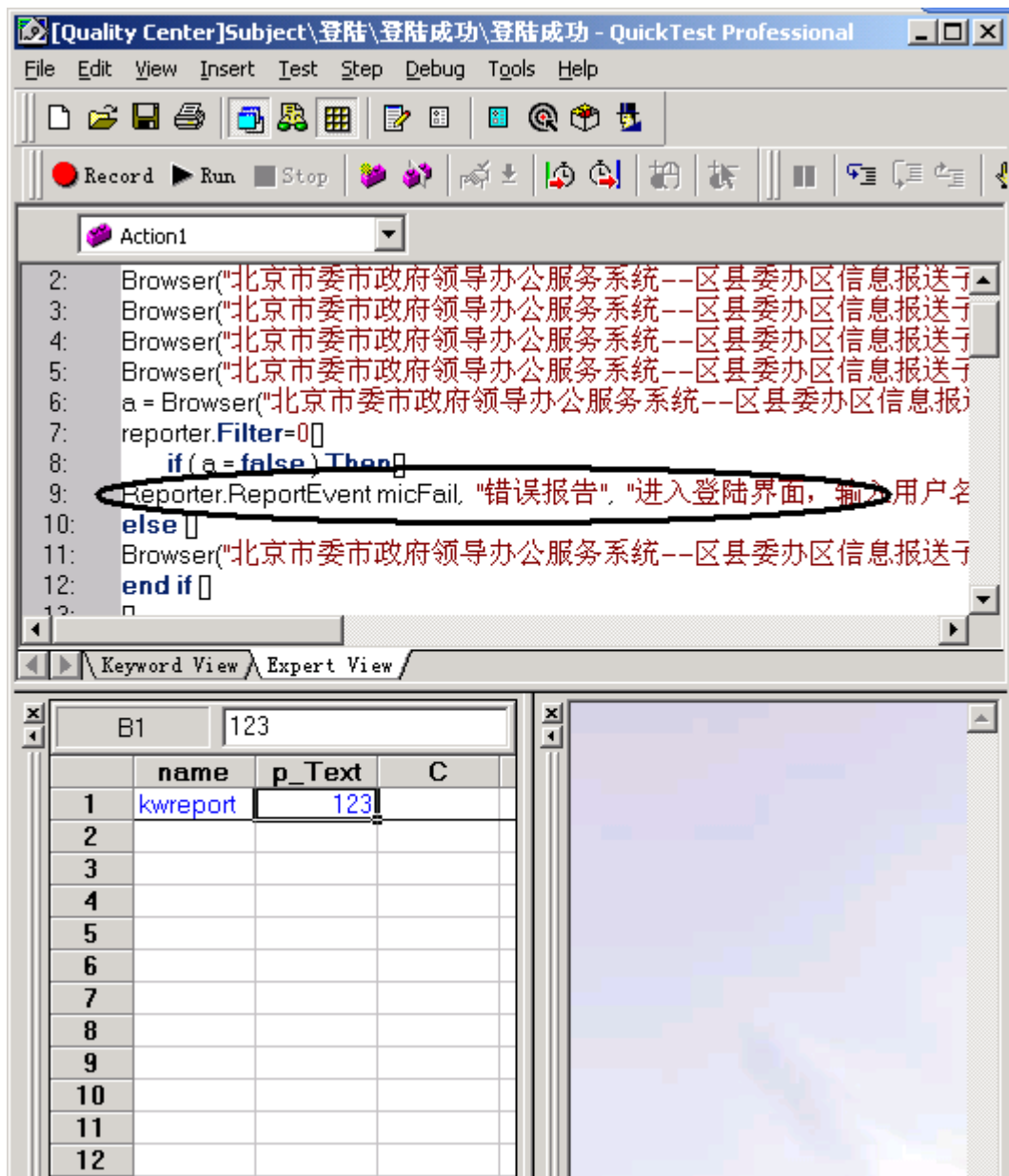


图 9.1.D2.4

也可以写成 reporter.reportevent 1, “错误报告.””进入登陆界面”

E 个人感受:无

9.2 关于 reporter.fiter

A 问题描述:

在你查看 QTP 的 reporter 时当有参数化时,reporter 右边的选择树会很长,即使看起来很不方便,也不利于查找错误

B 处理流程:

B1 录制脚本

B2 在 expert vIEW 中设定

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具, 他采用 B/S 模式, 使得用户可以在任何与 Server

相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具，主要用于 B/S 模式的系统测试

D 实例:

D1 录制脚本,详细见 QTP 使用手册

D2 在 expert vIEW 中的设定,当录制完成脚本,并且设置好检查点后,进入 expert vIEW,并且在其中输入 `reproter.filter=...`如下图

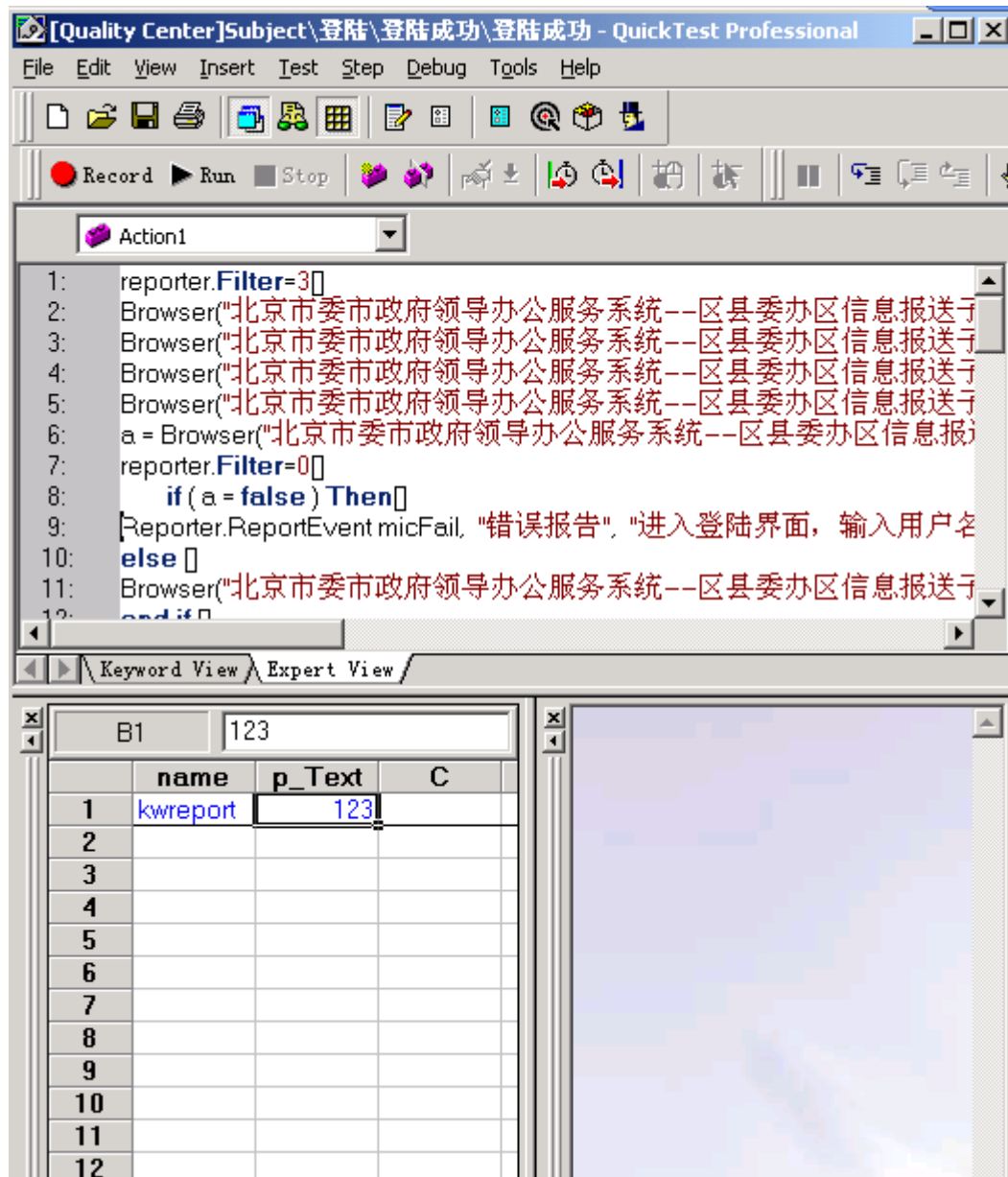


图 9.2.D2.1

Reporter.filter 分为四个属性:

0 或者 `renableall` 所有的报告都显示在 test result 中

1 或者 `enableerrorandwarning` 错误和警告信息显示在 test result 中

2 或者 `renableerroronly` 只把错误显示在 test result 中

3 或者 `rdisableall` 不在 test result 中显示任何信息

你可以根据需要设定你要显示的错误

如本问题例子

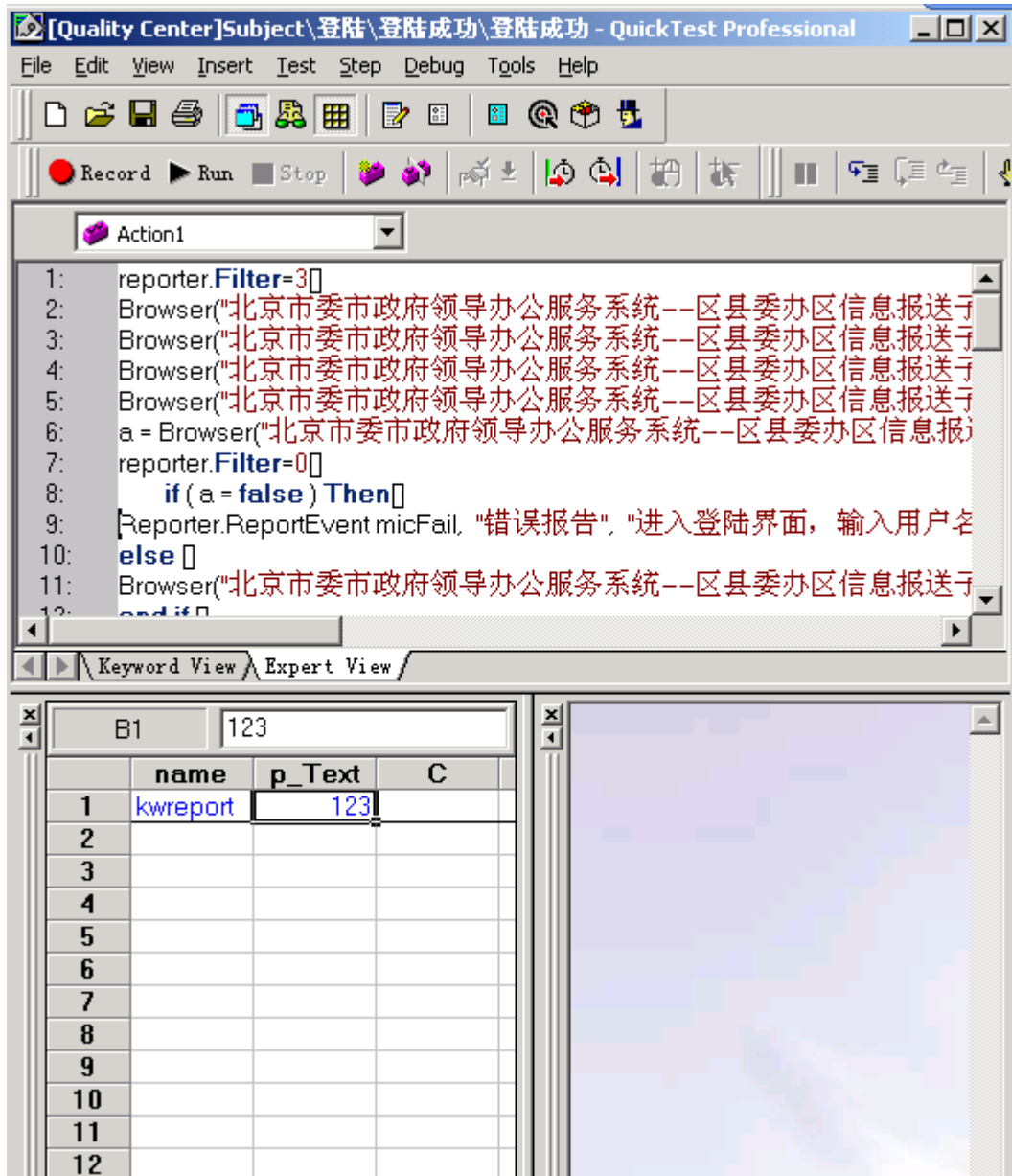


图 9.2.D2.2

在脚本的开头,设定不显示任何信息,包括检查点的错误信息

在检查点的下一个步骤加一个条件语句,如果检查点不通过,则显示自定义的错误信息

在条件语句之前,加上 reporter.filter 让信息显示出来

提示:如果 reporter.filter=3 也就是不显示任何信息,那么即使检查点检查出错误信息也不自动提交到 TD 中

E 个人感受:无

10 关于在 expert 中操作的问题

10.1 关于 getRoproperty ,getTopropertyse, getToproperty

A 问题描述

在 expert vIEW 中编写测试代码的时候需要得到一些控件的属性值,通过 getRoproperty, getTopropertyse, getToproperty 这三个函数,可以得到你想要的属性值,但是他们互相有区别,功能不同,以下对其作一些介绍

B 解答

B1 getRoproperty

返回控件的一个当前值,所谓当前值就是 QTP 捕获的控件的所有属性值,不一定是 QTP 识别控件的基本值,如下图,在 object propertIEs 中捕获的值都是可以由 getRoproperty 单独提取出来的

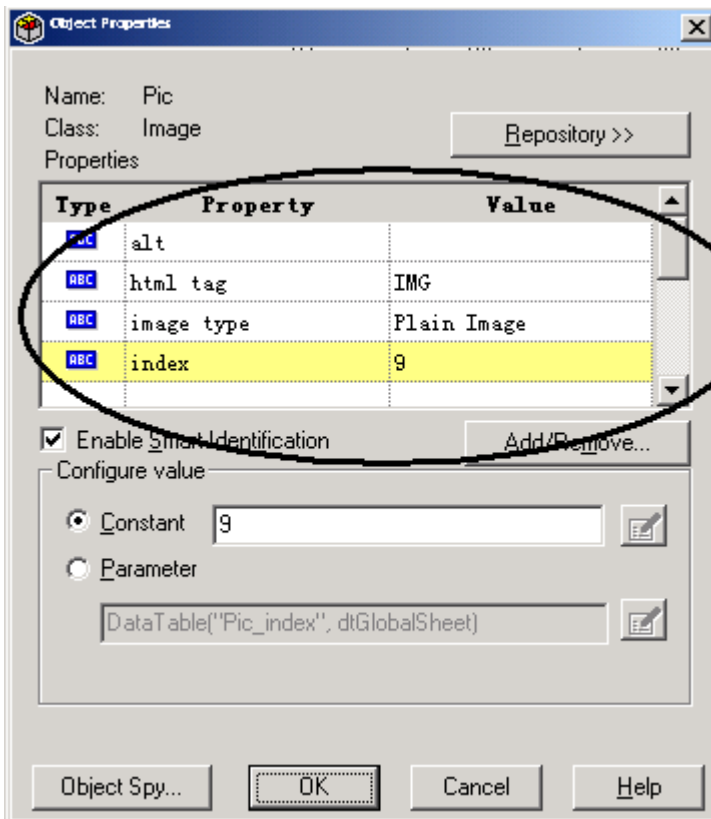


图 10.1.B.1

你也可以通过 add/remove 来增加新的属性

书写格式: object.GetROProperty (Property)

其中 property 为属性的名称

例:

下面的例子取出 radio group 的项目数,保存在 numoflterms 中

```
NumOfItems = Browser("Mercury Tours").Page("Find Flights").WebRadioGroup("seat pref").GetROProperty("Items Count")
```

NumOfItems 内容为 3

B2 getTopropertyse

返回控件所有被捕获的属性值,如下图

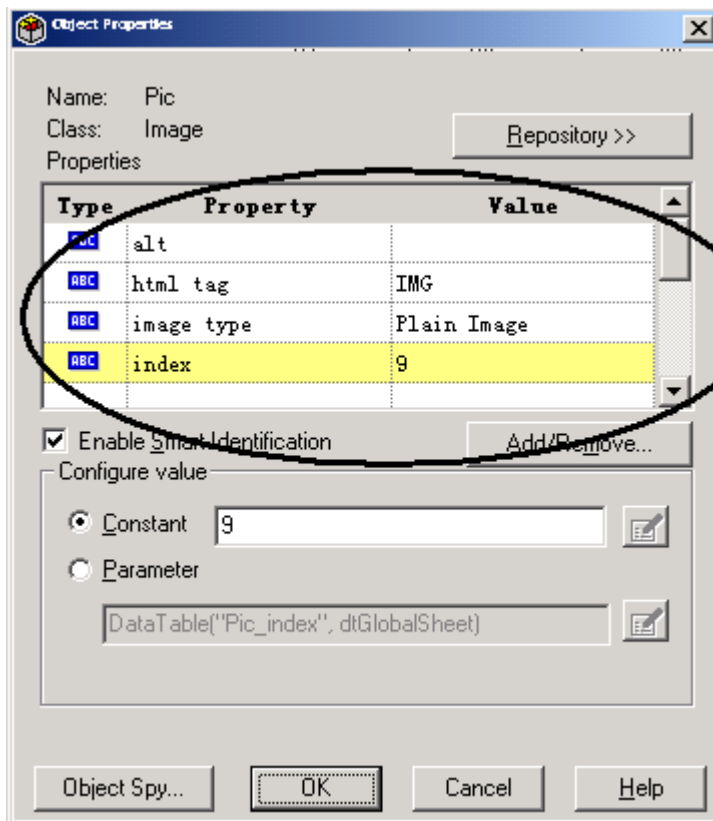


图 10.1.B.2

你也可以通过 add/remove 来添加新的属性

书写格式: object.GetTOProperties

例子:

以下的例子是使用 GetTOPropertyIEs 来将 web table 中的所有属性值提取出来并且将值付给 talbledesc

```
Set TableDesc = Browser("Mercury Tours").Page("Method of Payment").WebTable("FirstName").GetTOPPropertyEs
```

B3 getToproperty

返回 object description 中的当前值,如下图

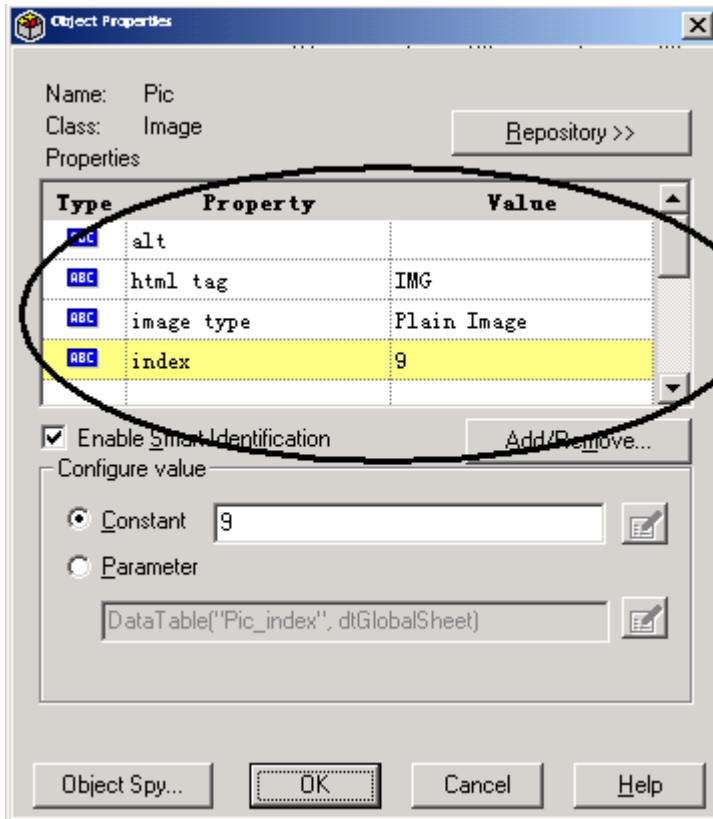


图 10.1.B.3

提示:如果你使用 `getToproperty` 返回了一个在 `object description` 不存在的属性值,那么 QTP 将在报告中以 `warning` 形式提示出来

书写格式: `object.GetTOProperty (Property)`

其中 `preperty` 为控件的属性名称

例子:

下面的例子使用 `GetTOProperty` 来返回 `WebTable` 的 `HtmlTag` 属性在 `Object`

`Repository` 中的的值 `Object`

```
TableTag = Browser("Mercury Tours").Page("Method of Payment").WebTable("FirstName").GetTOProperty("HtmlTag")
```

B4 `getToproperty` 看起来与 `getRoproperty` 类似,但是他们有实质的区别

`GetToproperty` 返回的是 object description 中的值,如下图

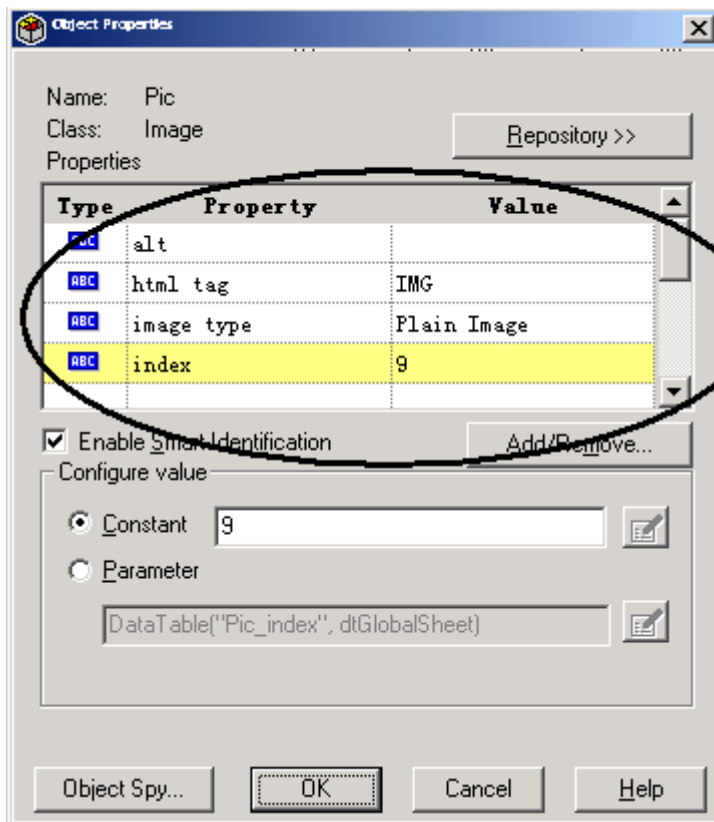
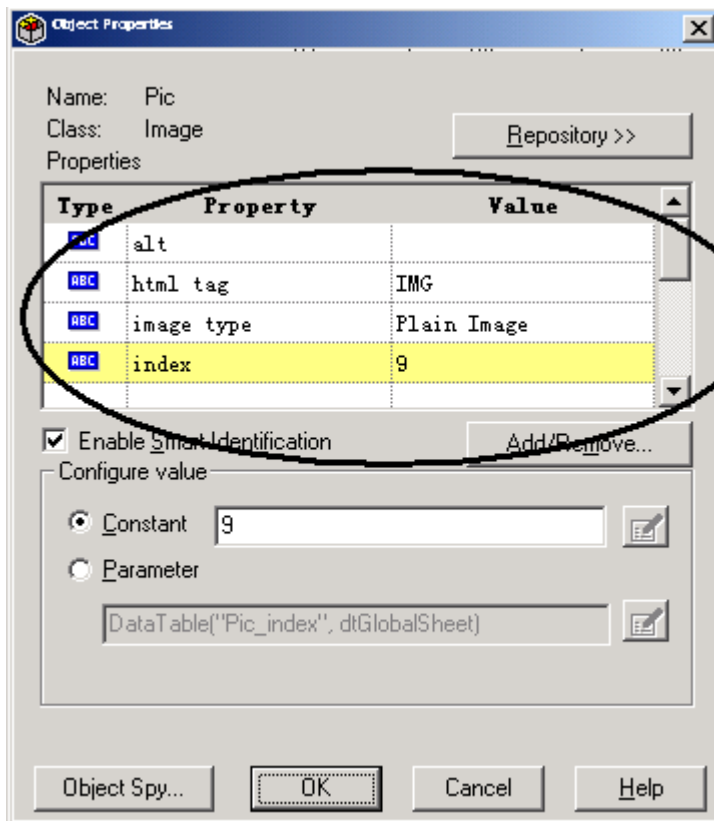


图 10.1.B.4

如其中的 `index` 这项,他的值在录制的过程中是 9,那么使用 `getToproperty` 返回的值就是 9

`GetRoproperty`:返回的是当前的值,如下图



其中的 index 值为 9 但是当 QTP 运行脚本时,由于脚本可能因为控件的属性发生改变,或者你对 index 这个值做了参数化,发现 index 值不为 9 了,变为了 10,使用 getRoproperty 返回的值就为 10

E 个人感受:无

10.2 编码时在出现 expect end of statement 的问题

A 问题描述:

当在 expert vIEW 中运行编写脚本后,可以通过点击 keyword vIEW 来检验编写是否有语法错误,但是 QTP 会给出错误提示,但是错误提示一般不会准确,并且在遇到他不能识别的错误时只能给出一种错误提示 expect end of statement

B 处理流程:

B1 编写代码

B2 点击 keyword vIEW

B3 弹出错误提示

B4 修改错误

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具，他采用 B/S 模式，使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具，主要用于 B/S 模式的系统测试

D 实例:

D1 编写代码,还是以登陆为例,如下图

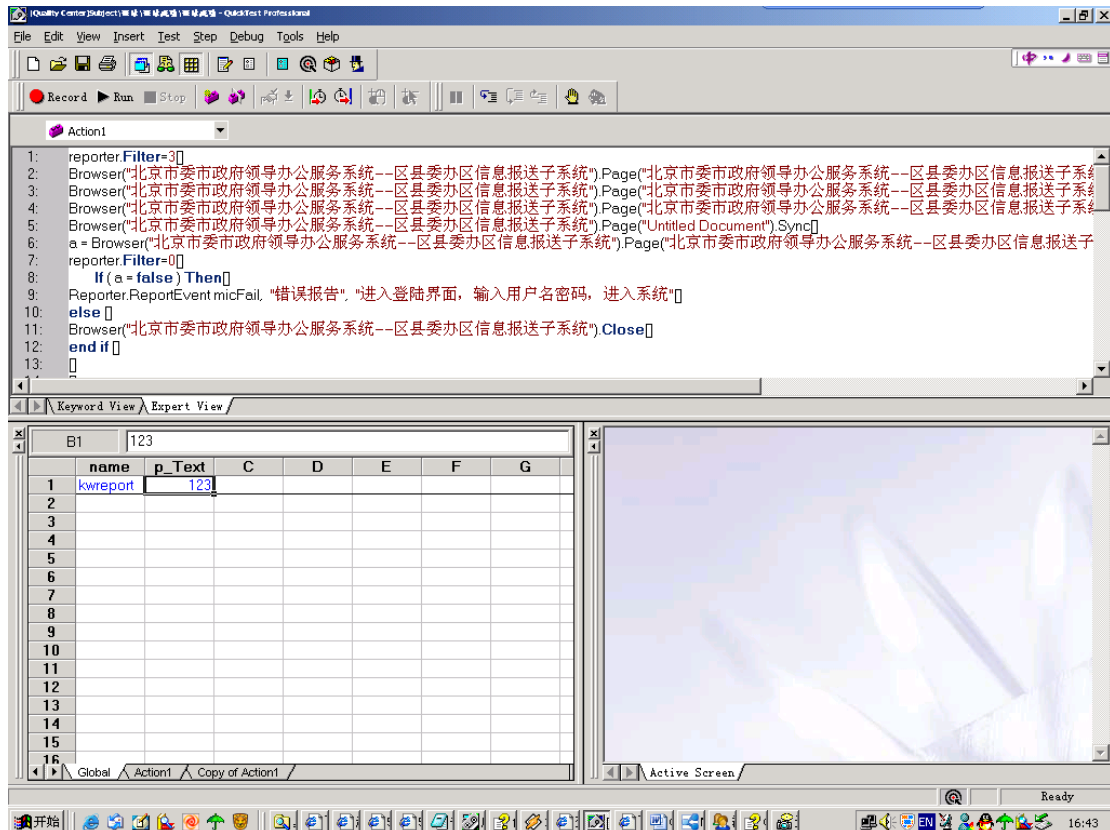


图 10.2.D1.1

D2 点击 keyword vIEW

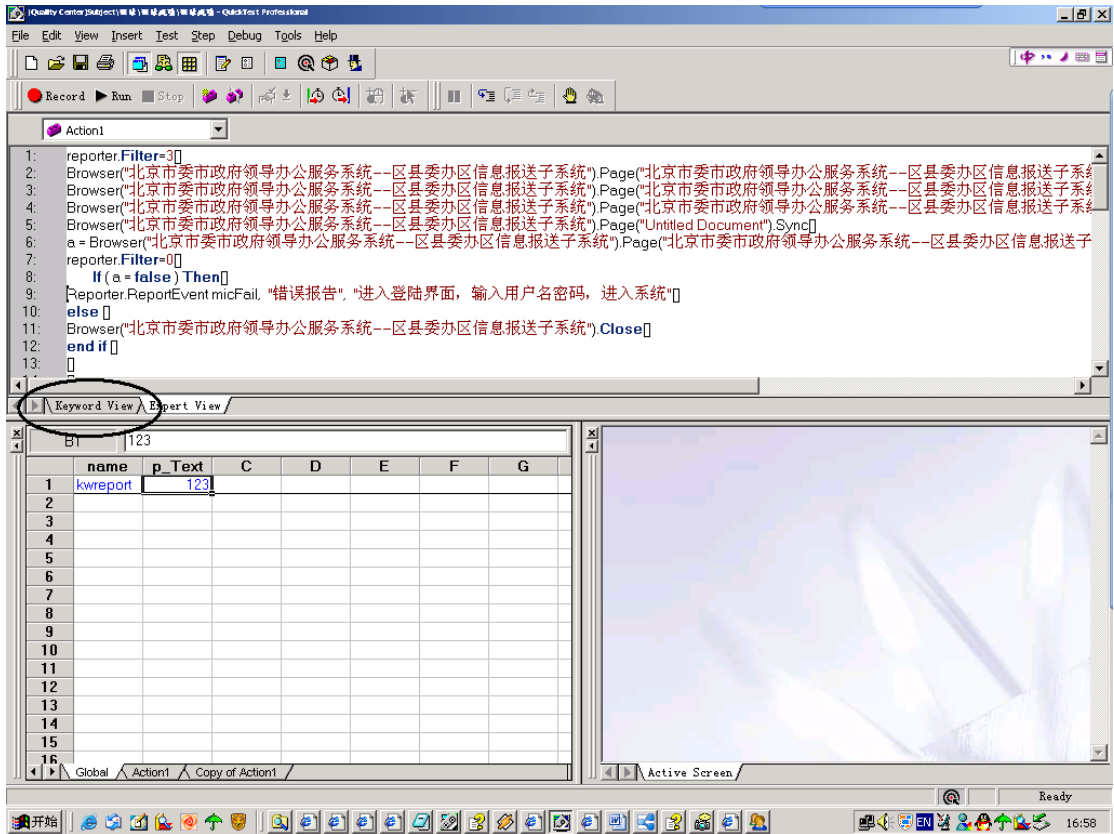


图 10.2.D2.2

D3 弹出错误提示,如果这段代码是 QTP 能够识别的错误,那么他会弹出对话框,提示哪个语法出现错误,不过一般 QTP 不能识别大多数错误,一般的对话框为

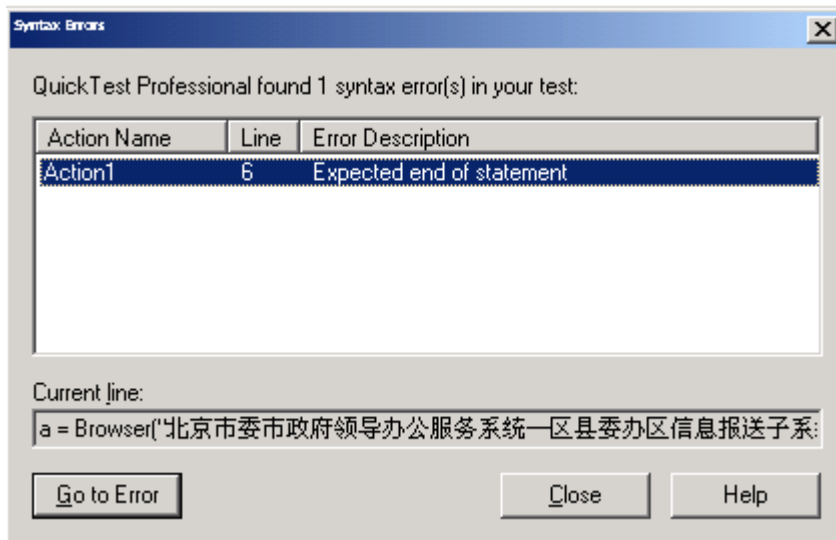


图 10.2.D3.3

这个对话框就是错误提示框,他显示了错误的行数,以及错误所在 Action,并且显示了错误行,在 error description 中显示了错误类型,当 QTP 无法识别错误类型时则出现 expected end of statement

D4 修改错误,点击  按钮,QTP 会自动将光标移动到出错的位置,如下图

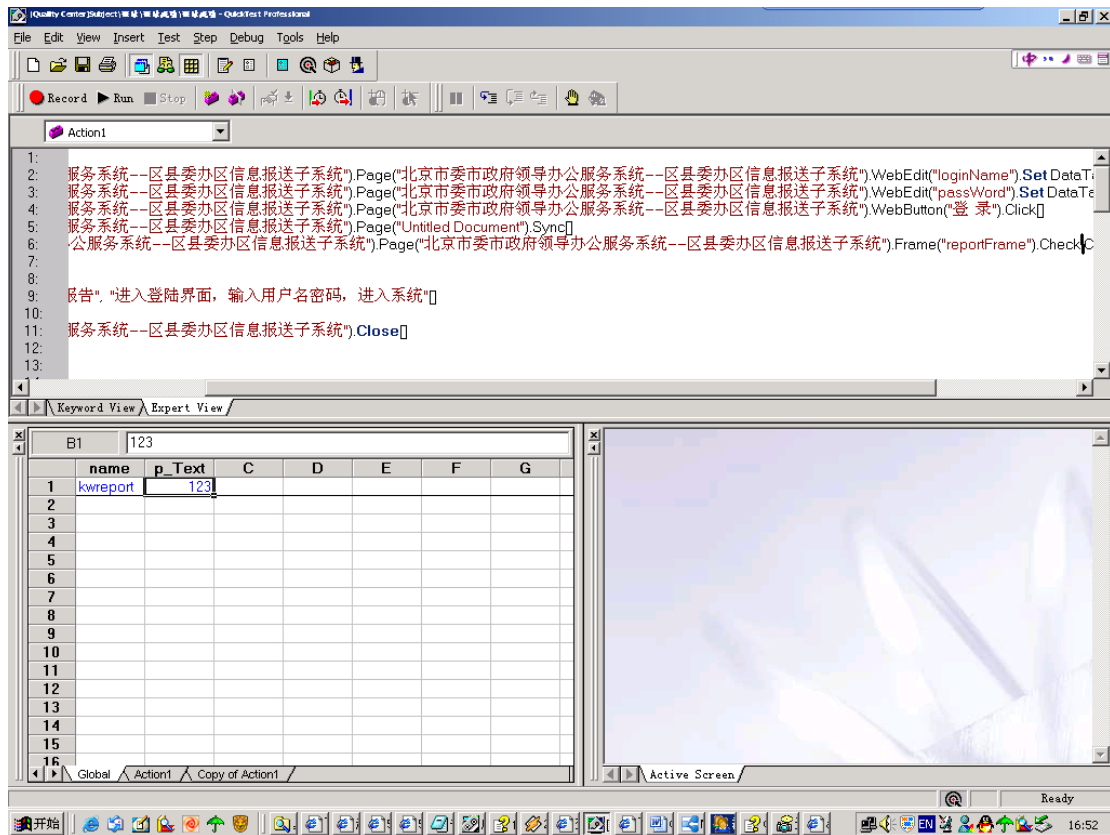


图 10.2.D4.4

在光标位置查找你的错误,并且修改,修改完成后,重复上 4 步,如果没有错误提示,则说明脚本没有语法错误,但是需要注意的是,逻辑错误 QTP 无法检验出来,需要一定的开发经验

E 个人感受:无

11 关于 cs 模式的录制

11.1 QTP 是否可以 cs 方式下的应用系统

A 问题描述:

通过以上的讲解,大家都知道 QTP 是一种测试 B/S 模式很好的工具,QTP 不但能够测试 B/S 模式,也可以测试 cs 模式

B 处理流程:

B1 进行 test and recording setting 的设置

B2 录制脚本

B3 回放脚本

C 术语说明

C1 TD 是 MI 公司的一款测试管理工具,他采用 B/S 模式,使得用户可以在任何与 Server 相连的计算机上实现测试用例的管理

C2 QTP 是 MI 公司推出的功能测试工具,主要用于 B/S 模式的系统测试

D 实例:

D1 进行 test and recording setting 的设置,打开一个新的 Action,如下图

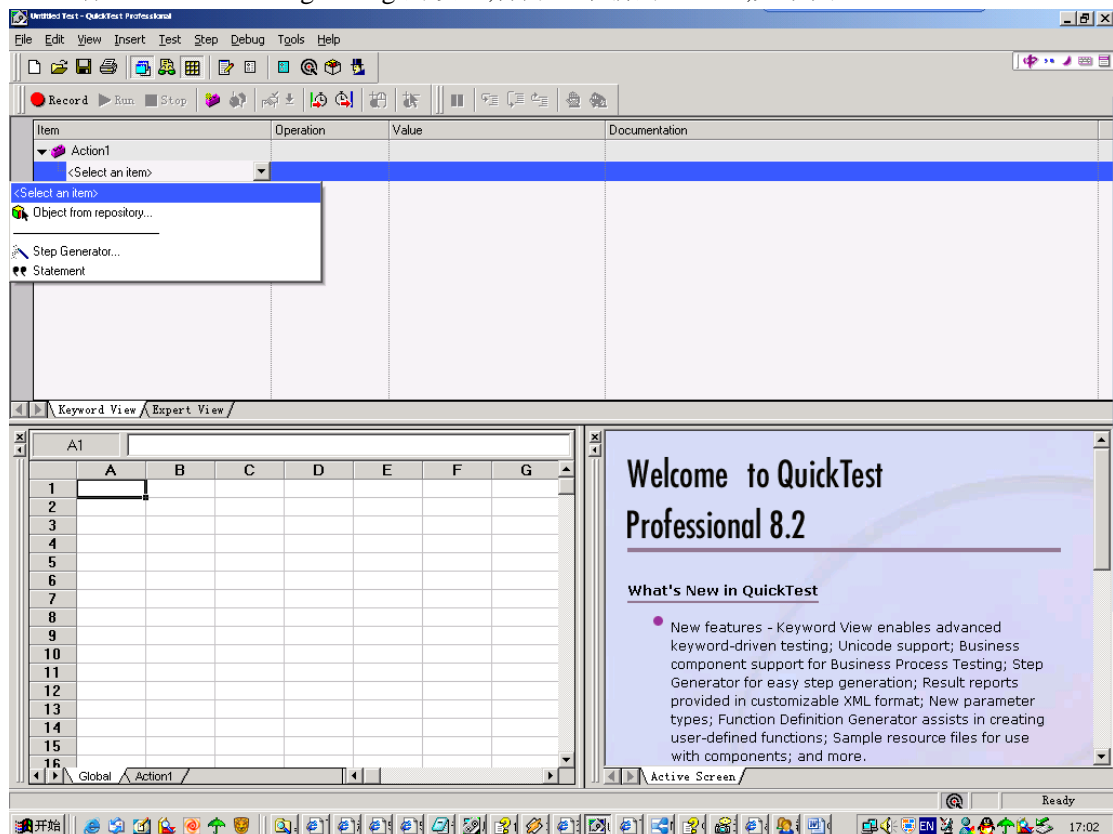
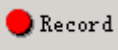


图 11.1.D1.1

点击 record 按钮 ,弹出如下对话框

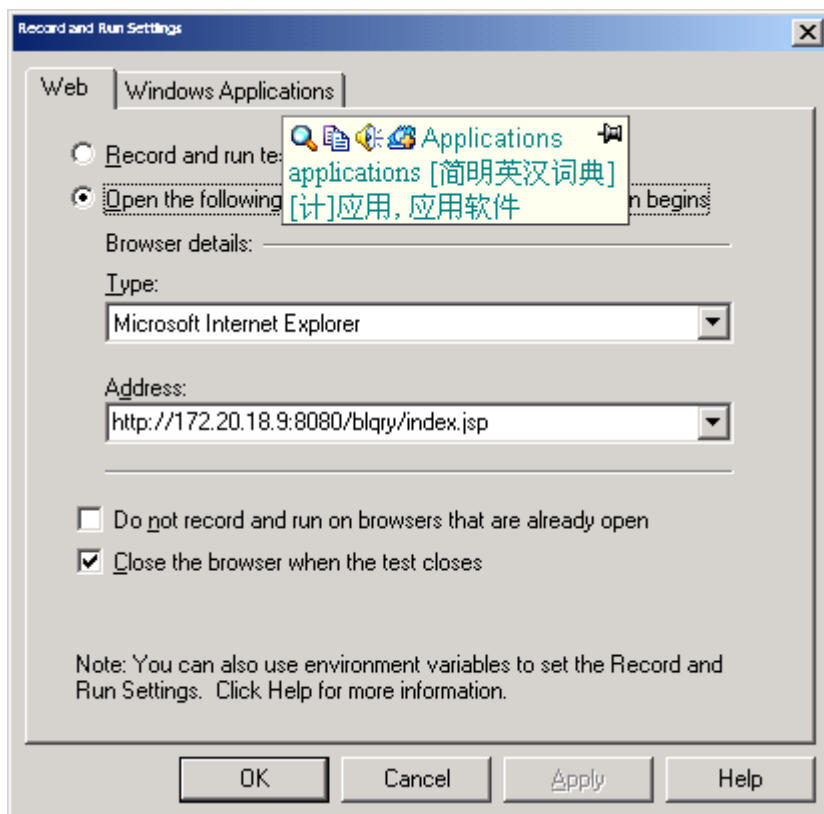


图 11.1.D1.2

选择 Windows Applications 如下图

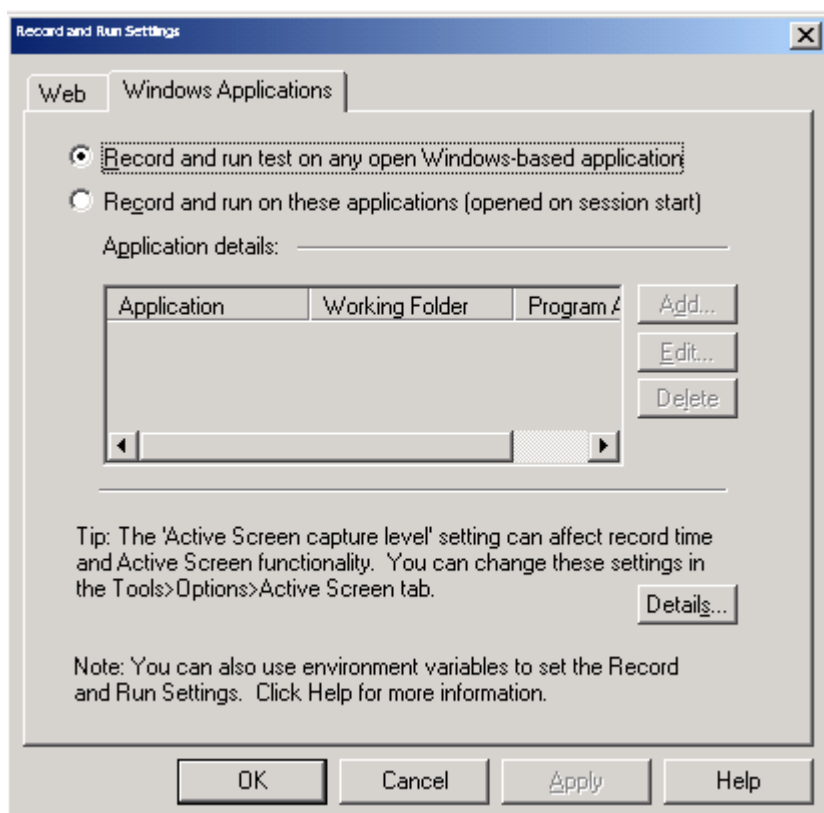


图 11.1.D1.3

这里如同 Web 中一样,可以设定是录制当前以打开的程序,还是录制指定路径的程序

Record and run test on any open windows-based application 为录制当前打开的程序
Record and run test on these application (opened on session start)录制指定路径的程序
选择 Record and run test on these application (opened on session start),如下图

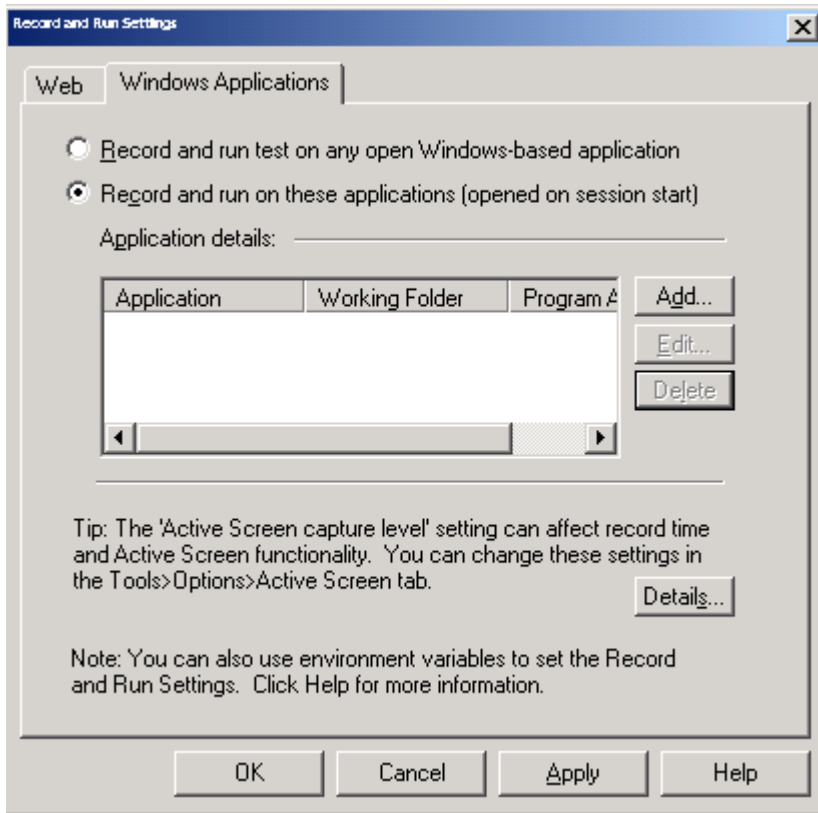


图 11.1.D1.4

选择 add  出现如下对话框

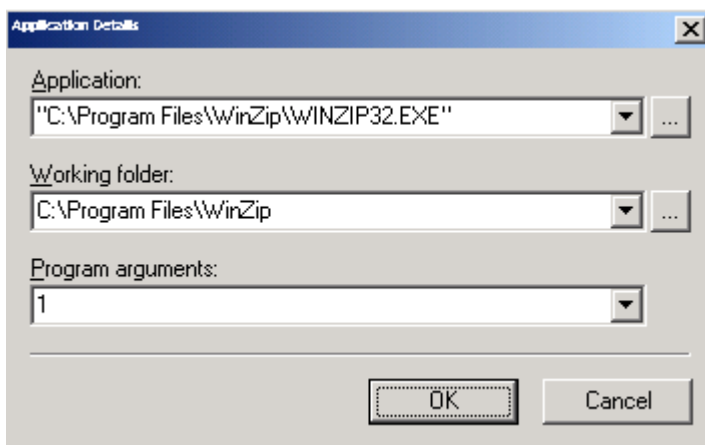



图 11.1.D1.5

点击 application 中的  按钮,选择一个程序,本例子选择 winzip

Working folder 为运行文件夹

Program arguments 为描述

点击 ok 开始录制

提示: 在设置完成 application 后应该确保 web 中的选项如下图所示

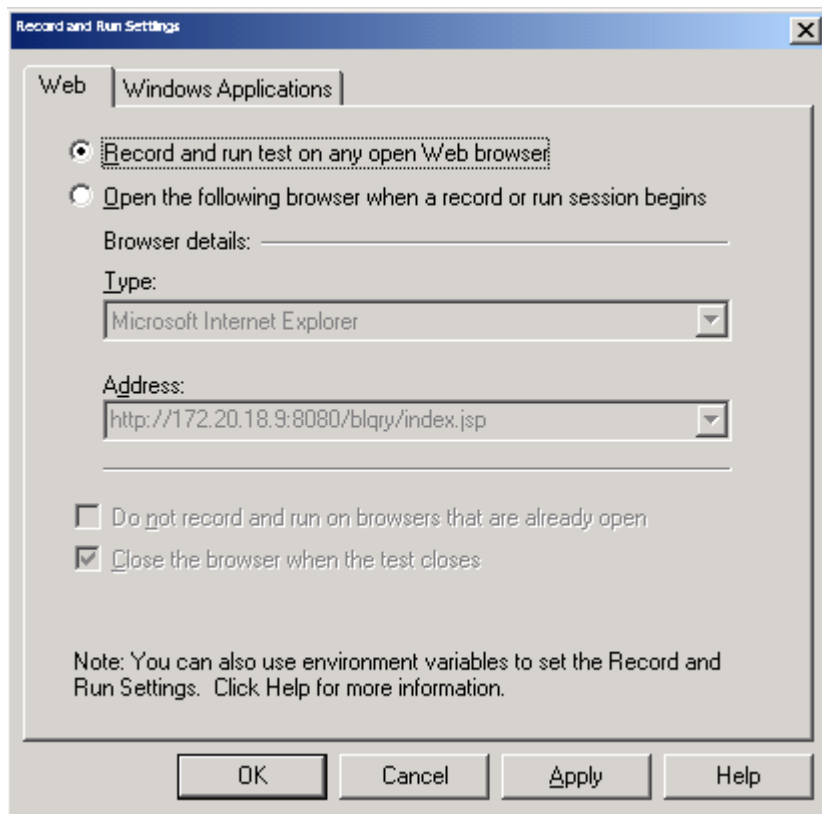


图 11.1.D1.6

D2 录制脚本

录制完成如下图所示显示脚本,并且可以回放

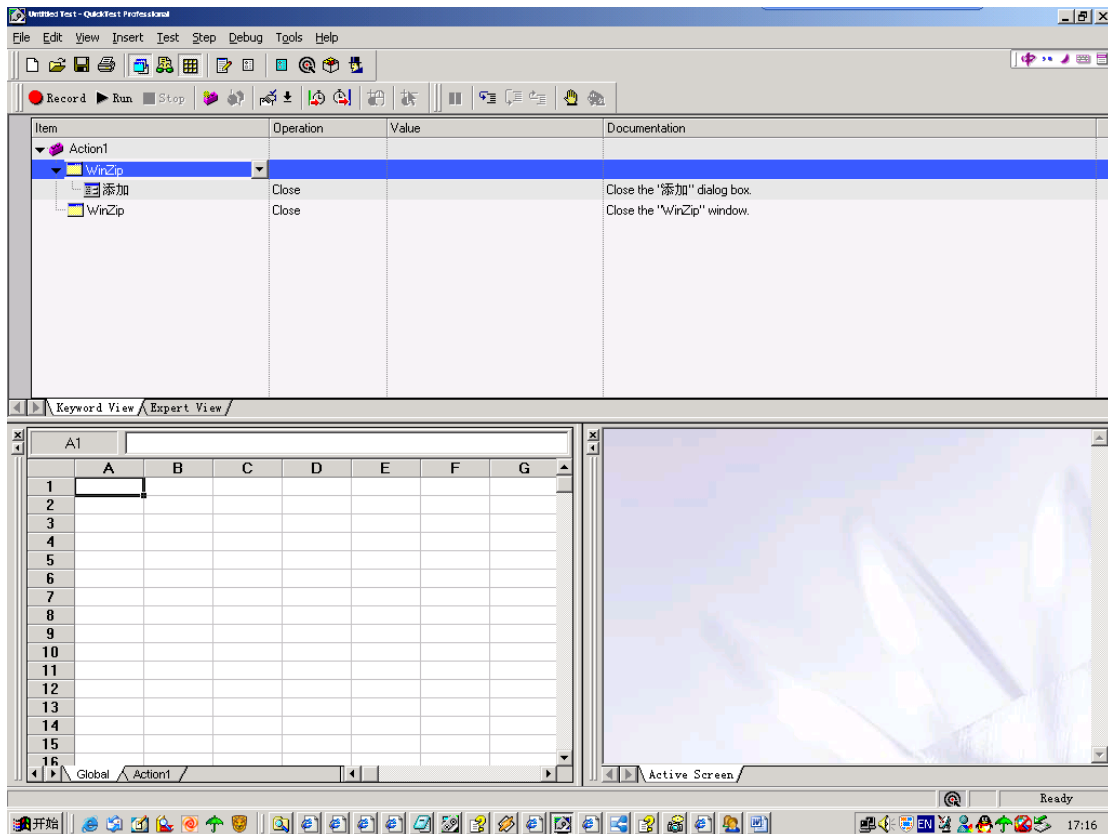


图 11.1.D2.7

D3 回放脚本

但是这里的 active screen 里面没有页面,因为页面 active screen 是为 web 而设

提示: 由于没有 active screen,这就需要如果要添加检查点,则要一般录制一边添加

E 个人感受:无