Testing a servlet using TTCN-3

By Bernard Stepien

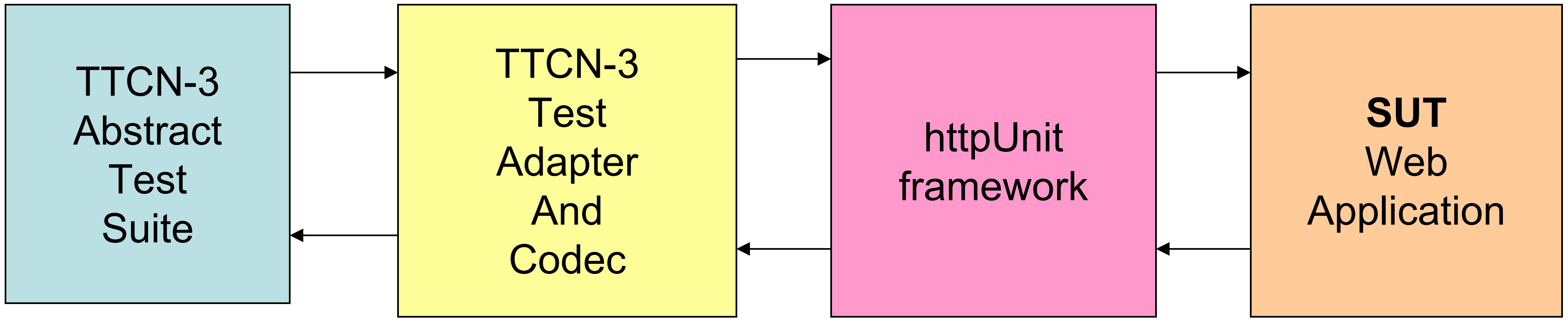
University of Ottawa

bernard@site.uottawa.ca

Topics covered

- Invoking a web page
- Modeling web responses
- Simulating link clicking
- Modeling the shopping cart
- Simulating form submission
- Integrating TTCN-3 and httpUnit

Separation of concerns



Invoking a web page

In the TTCN-3 Abstract Test Specification:

web_port.**send**("**http://localhost:8080/widgets/servlet**");

In the TestAdapter:

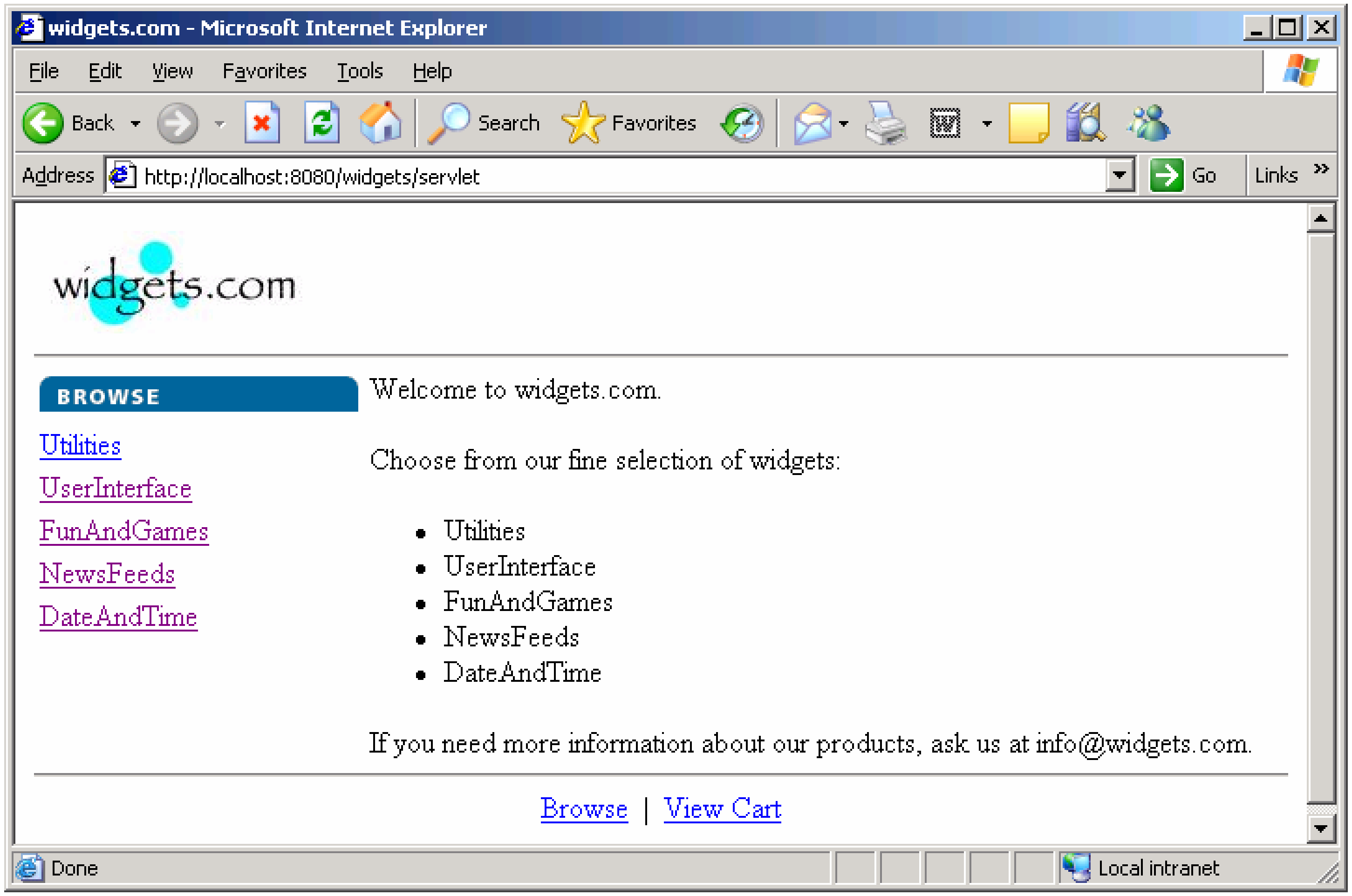
static WebConversation wc;

In the constructor of the test adapter:

wc = new WebConversation();

```
public TriStatus triSend(…, TriMessage sendMessage) {

        String theMsg = getStringMsg( sendMessage);

        WebRequest request = new GetMethodWebRequest( theMsg );

        response = wc.getResponse( request );
        …
}
```

The Home Page



Welcome to widgets.com.

Choose from our fine selection of widgets:

- Utilities
- UserInterface
- FunAndGames
- NewsFeeds
- DateAndTime

If you need more information about our products, ask us at info@widgets.com.

Modeling a web response data type

TTCN-3 types

```
type record WebResponseType {

        integer statusCode,
        charstring title,
        charstring content,
        linkList links optional,
        formSetType forms optional,
        TableSetType tables optional

}
```

httpUnit methods

WebResponse class

getResponseCode()
getTitle()
getText()
getLinks()
getForms()
getTables()

And many other elements of information depending on the test purpose

Modeling web links

```
type record linkType {
            charstring text,
            charstring link
}
```

```
<a href=/widgets/servlet/browse?action=widgets&category=Utilities>Utilities</a>
```

Links enforcement approach

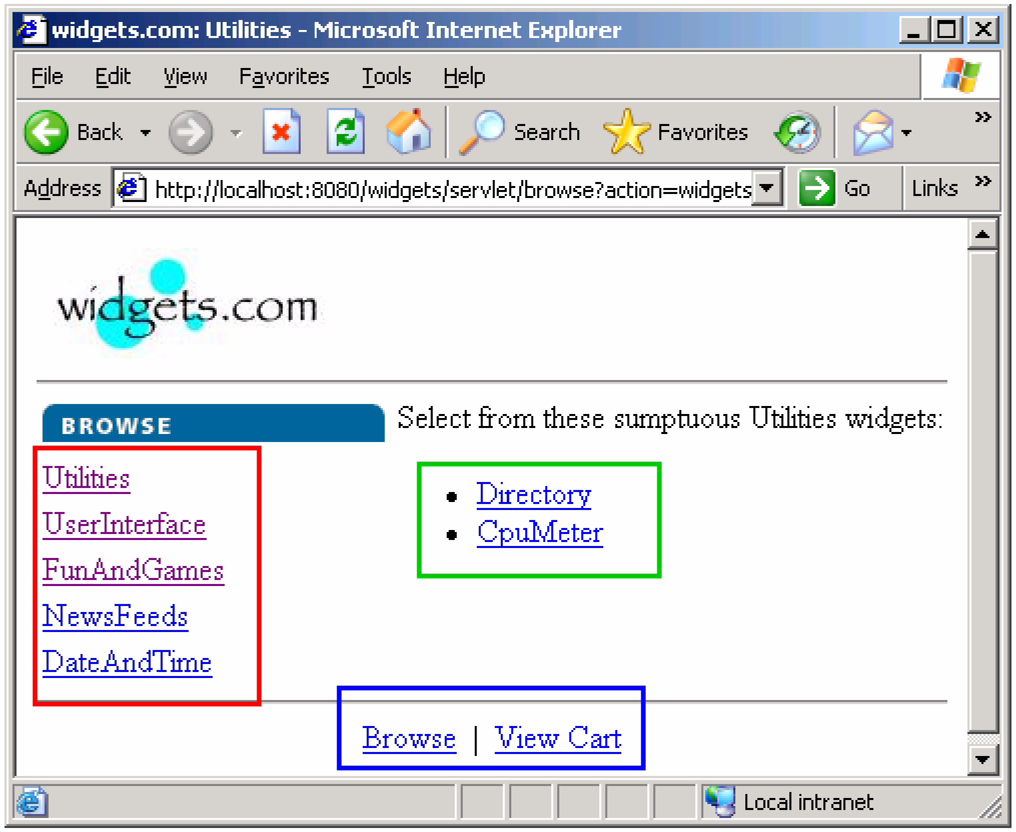
```
template linkType utilitiesTemplate := {
            text := "Utilities",
            link := "/widgets/servlet/browse?action=widgets&category=Utilities“
}
```

Links discovery approach

```
template linkType utilitiesTemplate := {
            text := "Utilities",
            link := “?“
}
```

```
type set of linkType linkList;
```

Structuring templates for links


widgets.com: Utilities - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/widgets/servlet/browse?action=widgets
Go Links
widgets.com
BROWSE
Local intranet

Select from these sumptuous Utilities widgets:

Utilities
UserInterface
FunAndGames
NewsFeeds
DateAndTime

- Directory
- CpuMeter

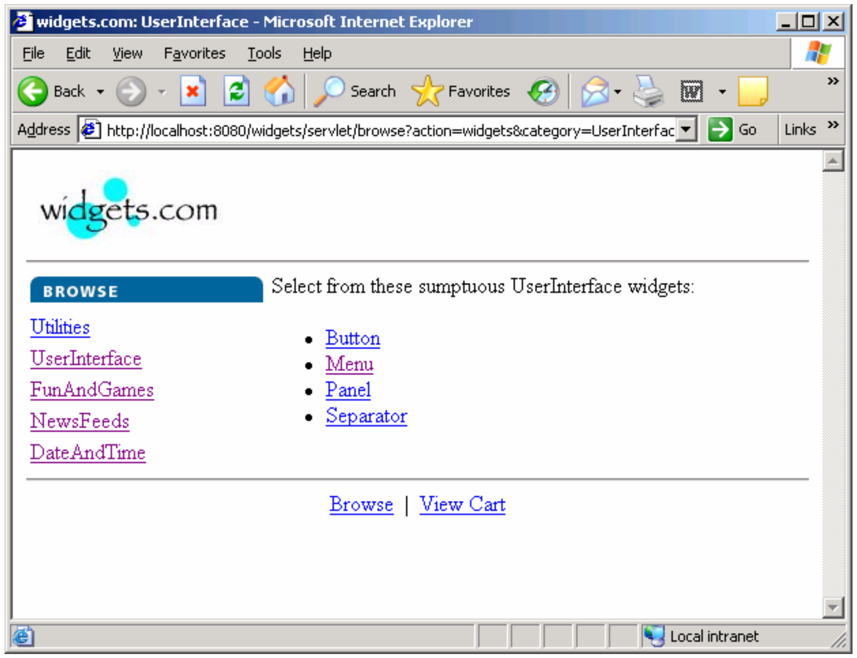
Browse | View Cart

- Browse pages are composed of three groups of links:
  - Categories
  - Items in a category
  - Home page/view cart
- Categories and home page/view cart are constant from one page to another
- Impossible to obtain clearly separated subsets of links from httpUnit
- Solution uses set concatenation

Structuring templates for links

```
template linkList theCategoriesLinks := {
    {text := "Utilities", link := "/widgets/servlet/browse?action=widgets&category=Utilities"},
    {text := "UserInterface", link := "/widgets/servlet/browse?action=widgets&category=UserInterface"},
    {text := "FunAndGames", link := "/widgets/servlet/browse?action=widgets&category=FunAndGames"},
    {text := "NewsFeeds", link := "/widgets/servlet/browse?action=widgets&category=NewsFeeds"},
    {text := "DateAndTime", link := "/widgets/servlet/browse?action=widgets&category=DateAndTime"}
}

template linkList theDisplayLinks := {
    {text := "Browse", link := "/widgets/servlet/browse"},
    {text := "View Cart", link := "/widgets/servlet/browse?action=view-cart"}
}

template linkList theUtilitiesItems := {
    {text := "Directory", link := "/widgets/servlet/browse?action=details&category=Utilities&widget=Directory"},
    {text := "CpuMeter", link := "/widgets/servlet/browse?action=details&category=Utilities&widget=CpuMeter"}
}

template linkList theHomePageLinks := theDisplayLinks & theCategoriesLinks;

template linkList theUtilitiesLinks := theHomePageLinks & theUtilitiesItems;
```

Modeling a web response templates


widgets.com: UserInterface - Microsoft Internet Explorer

Address http://localhost:8080/widgets/servlet/browse?action=widgets&category=UserInterfac

widgets.com

BROWSE

Select from these sumptuous UserInterface widgets:

- Utilities
- UserInterface
- FunAndGames
- NewsFeeds
- DateAndTime

- Button
- Menu
- Panel
- Separator

Browse | View Cart

Local intranet

```
template WebPageType UserInterfaceLinksTemplate := {
            statusCode := 200,
            title := "widgets.com: UserInterface",
            content := pattern "*Select from these sumptuous UserInterface widgets:*",
            links := theUserInterfaceLinks,
            forms := ?,
            tables := ?
}
```

Modeling web response behavior

```
function BrowsePagesBehavior(WebPageType thePageTemplate) runs on PTCType {

        alt {
                [] web_port.receive(thePageTemplate) -> value theBrowsePageResult {
                }
                [] ErrorBrowseBehavior()
                }
        }
}
```

Test case invocation

```
Webport.send("/widgets/servlet/browse?action=widgets&category=UserInterface");

BrowsePagesBehavior(UserInterfaceLinksTemplate );
…
```

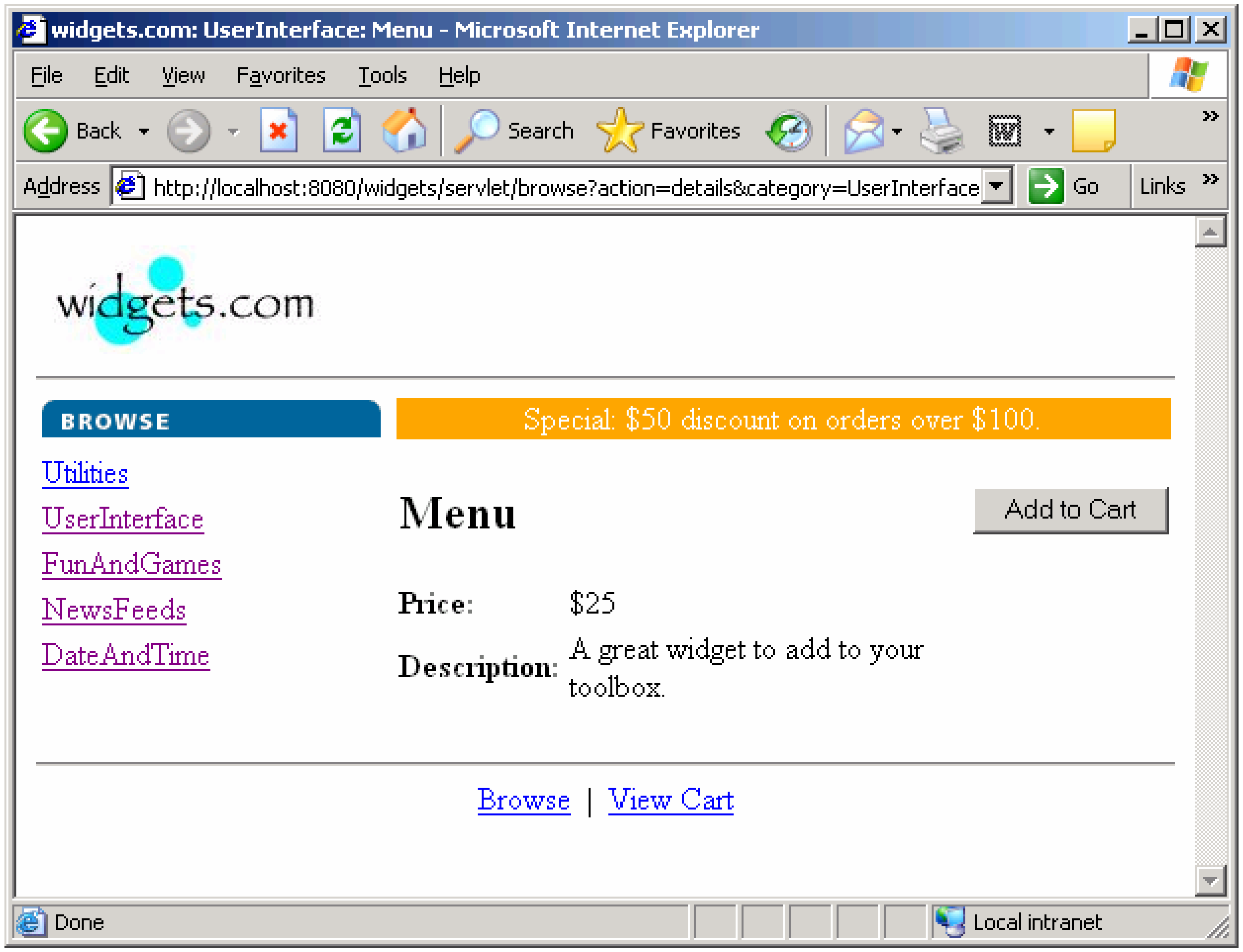
Simulating link clicking

- After each response, save the collected links from the WebResponse template into a variable
- Invoke a function that searches the list of links by name and returns a URL
- Make a TTCN-3 send using the found URL

Link clicking solution

```
Testcase BaseCaseTest() {
            clickOnLink("Utilities");
            UtilitiesBehavior();
}


  function clickOnLink(charstring theLinkName) runs on MTCType {
              var charstring thePressedLink;

              thePressedLink := getHref(theLinkName, theBrowsePageResult.links);

              log("cliked link: " & theLinkName & " --- " & thePressedLink);

              web_port.send(theBase & thePressedLink);
  }
```

After clicking the Menu link


widgets.com: UserInterface: Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/widgets/servlet/browse?action=details&category=UserInterface
Go Links
widgets.com
BROWSE
Utilities
UserInterface
FunAndGames
NewsFeeds
DateAndTime
Add to Cart
Browse | View Cart
Done
Local intranet

Special: $50 discount on orders over $100.

Menu

Price:	$25
Description:	A great widget to add to your toolbox.

- This page has a form with a button to add this item to the shopping cart

The Add-to-cart form HTML

```
<form name=details action=/widgets/servlet/browse?action=add-to-cart method=post>
        <input type=hidden name=category value=UserInterface>
        <input type=hidden name=widget value=Menu>
        <table width=100%>
                <tr>
                        <td bgcolor=orange align=center> <font color=white>Special: $50 dis
                </tr>
        </table>
        <br>
        <table width=100%
                <tr>
                        <td colspan=2 valign=top> <h2>Menu</h2> </td>
                        <td valign=top> <input type=submit value='Add to Cart'> </td>
                </tr>
                <tr>
                        <td> <b>Price:</b> </td>
                        <td> <span id=price>$25</span> </td>
                </tr>
                <tr>
                        <td> <b>Description:</b> </td>
                        <td> A great widget to add to your toolbox. </td>
                </tr>
        </table>
</form>
```

Modeling web forms in TTCN-3

```
type record browseFormType {
        charstring name,
        charstring formAction,
        charstring kindMethod,
        formElementSetType elements
}

type record formElementType {
        //charstring elementType,
        charstring name,
        charstring elementValue
}

type set of formElementType formElementSetType;

type set of browseFormType formSetType;
```

The Add-to-cart form template

```
template browseFormType MenuForm := {
        name := "details",
        formAction := "/widgets/servlet/browse?action=add-to-cart",
        kindMethod := "post",
        elements := {
                {name := "category", elementValue := "UserInterface" },
                {name := "widget", elementValue := "Menu" },
                {name := "action", elementValue := "add-to-cart" }
        }
}

template WebPageType MenuTemplate := {
        statusCode := 200,
        title := "widgets.com: UserInterface: Menu",
        content := pattern "*Menu*25*A great widget to add to your toolbox*",
        links := theItemPageLinks,
        forms := { MenuForm },
        tables := ?
}
```

Executing a form submit

Dependency on **httpUnit** approach to form submission

```
WebForms theForms =response.getForms();
```

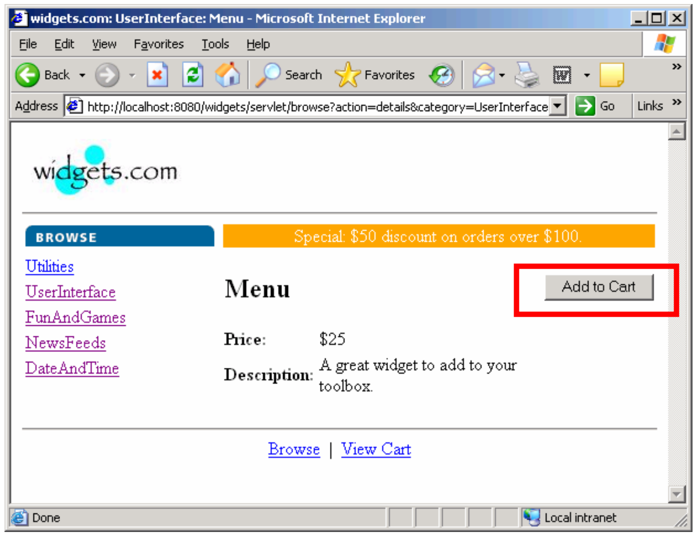
1. Get the correct form by name or using a submit button name
2. Invoke the submit button of the appropriate form

```
WebResponse newResponse = theForms[i].submit(“Add to Cart”);
```

```
type record formSubmitType {
            charstring formName,
            charstring buttonName,
            ParameterValuesSetType parameterValues
}
```

```
template formSubmitType addToCartTemplate := {
            formName := "details",
            buttonName := "Add to Cart",
            parameterValues := {}
}
```

```
web_port.send(addToCartTemplate);
```


widgets.com: UserInterface: Menu - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/widgets/servlet/browse?action=details&category=UserInterface
Go Links
widgets.com
BROWSE
Utilities
UserInterface
FunAndGames
NewsFeeds
DateAndTime
Add to Cart
Browse | View Cart
Done
Local intranet

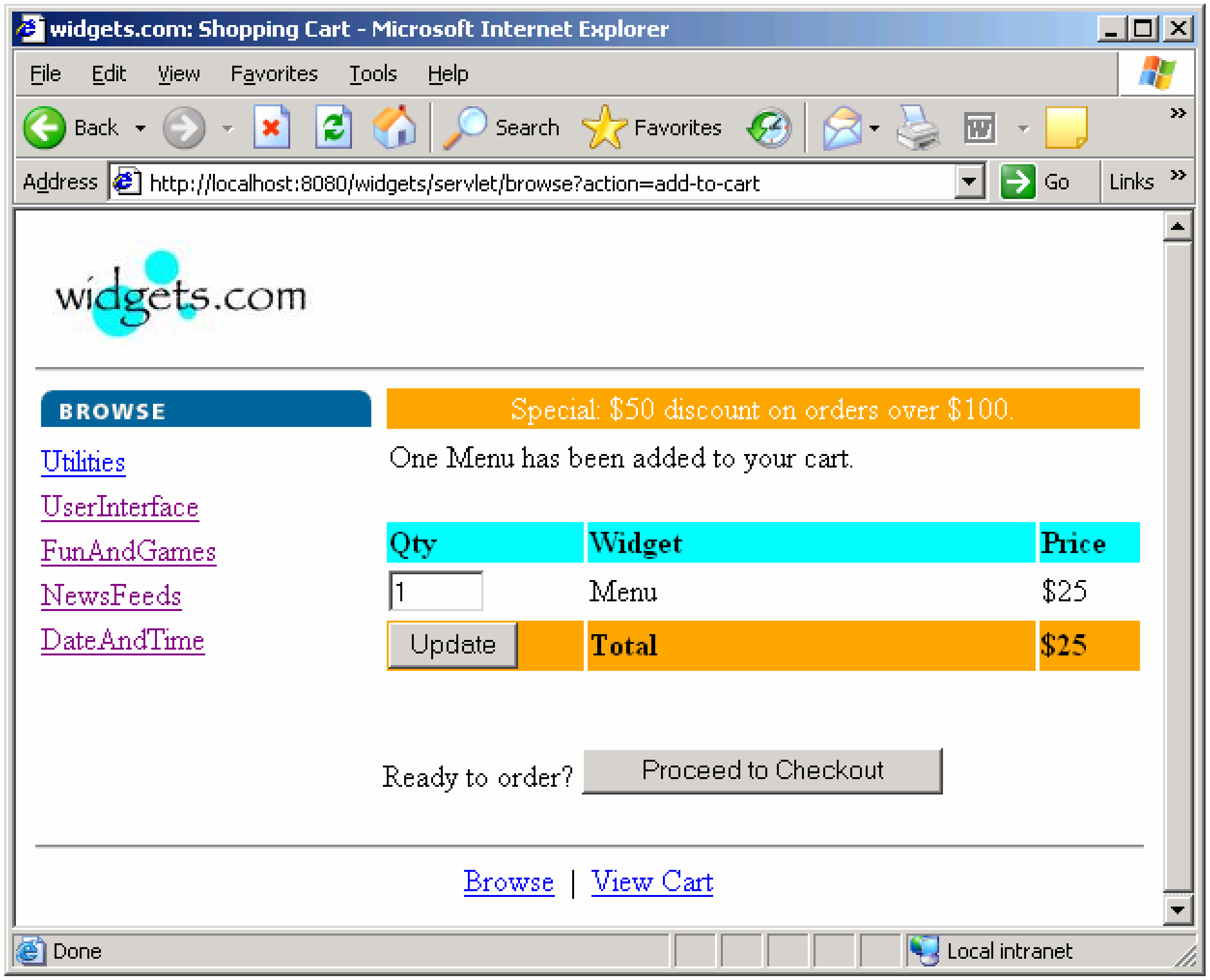
Special: $50 discount on orders over $100.

Menu

Price: $25

Description: A great widget to add to your toolbox.

After adding an item to the cart


widgets.com: Shopping Cart - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/widgets/servlet/browse?action=add-to-cart
Go Links
widgets.com
BROWSE
Utilities
UserInterface
FunAndGames
NewsFeeds
DateAndTime
Update
Proceed to Checkout
Browse | View Cart
Done
Local intranet

Special: $50 discount on orders over $100.

One Menu has been added to your cart.

Qty	Widget	Price
1	Menu	$25
Update	**Total**	**$25**

Ready to order? Proceed to Checkout

Display the content of the shopping cart along with the calculation of the total amount

Testing the content of the shopping cart

- After adding a number of items we need to check that the shopping cart content displayed on the page is correct. This includes:
  - Correct quantities
  - Correct product names
  - Correct unit prices
  - Correct total to date
- Somehow we need to keep track in the ATS of what the content of the shopping cart should be
- This would save tedious manual calculations of TTCN-3 template values.

Modeling the shopping cart data types

```
type record shoppingCartEntryType {
        integer quantity,
        charstring name,
        float price
}

type set of shoppingCartEntryType shoppingCartSetType;


type component MTCType {
        var shoppingCartSetType theCurrentShoppingCart := {};
        …
        port web web_port;
}


addToShoppingCart(1, "Menu", 25.00);
```

Modeling the shopping cart utilities

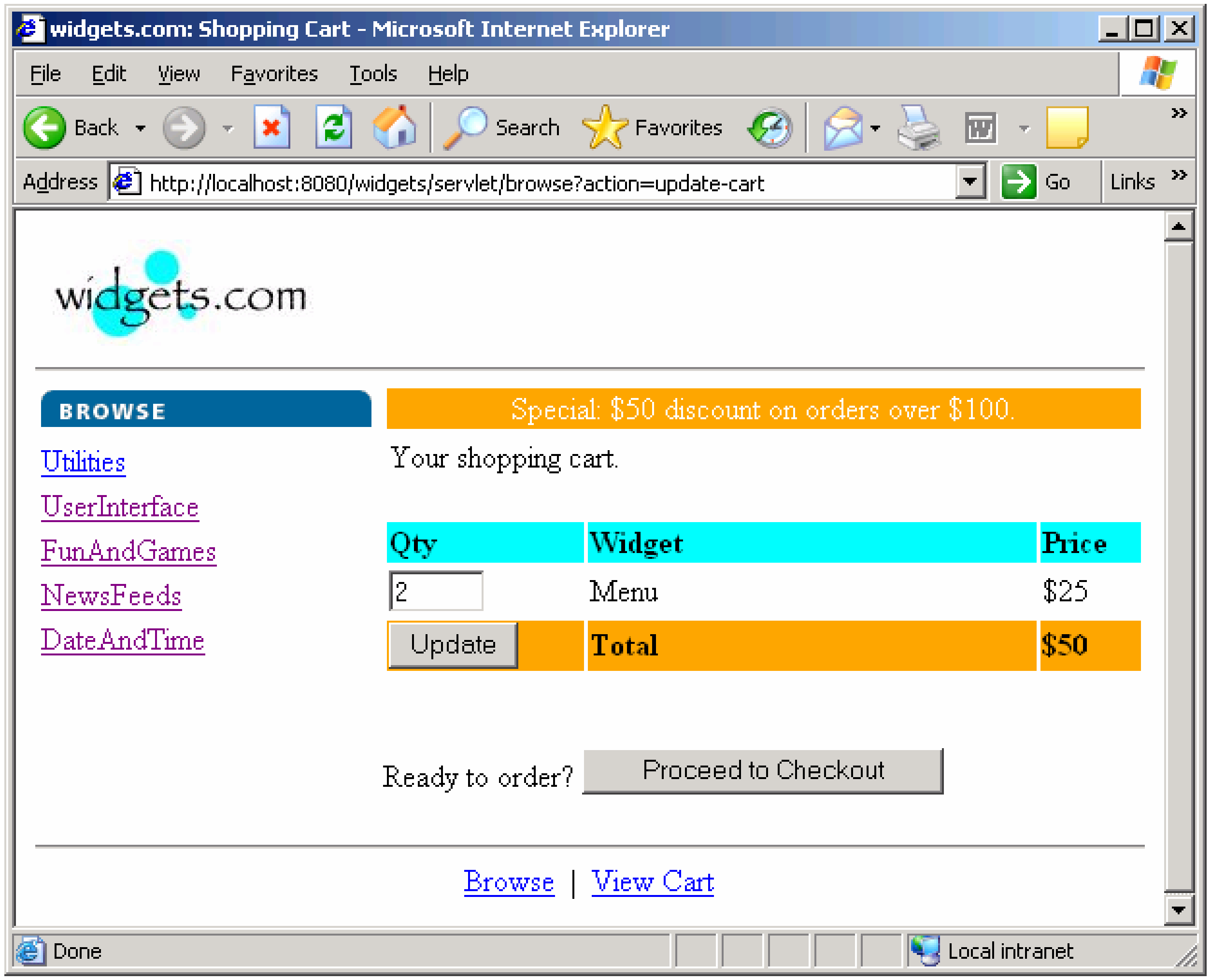
```
function addToShoppingCart(integer theQuantity, charstring theItem, float thePrice)
                                                                runs on MTCType {
        var integer nbItems := sizeof(theCurrentShoppingCart);
        var integer i;

        for(i:= 0; i < nbItems; i := i+1) {
                if(theCurrentShoppingCart[i].name == theItem) {

                        theCurrentShoppingCart[i].quantity :=
                                theCurrentShoppingCart[i].quantity + theQuantity;
                        return
                }
        }

        theCurrentShoppingCart[nbItems] := { theQuantity, theItem,
                                            (thePrice * int2float(theQuantity)) };
}
```

Simulating form values updates

- To change the quantity purchased we use the update form template
- We use a parametric template to pass the set of parameter values

```
template formSubmitType UpdateTemplate(ParameterValuesSetType
                                                        theParameterValues) := {
          formName := "cart",
          buttonName := "Update",
          parameterValues := theParameterValues
}

web_port.send(UpdateTemplate({ {parmName := "quantity.Menu",
                                                        parmValue := "2"} }));
```

After pressing the update button



widgets.com: Shopping Cart - Microsoft Internet Explorer

http://localhost:8080/widgets/servlet/browse?action=update-cart

widgets.com

Utilities
UserInterface
FunAndGames
NewsFeeds
DateAndTime

Special: $50 discount on orders over $100.

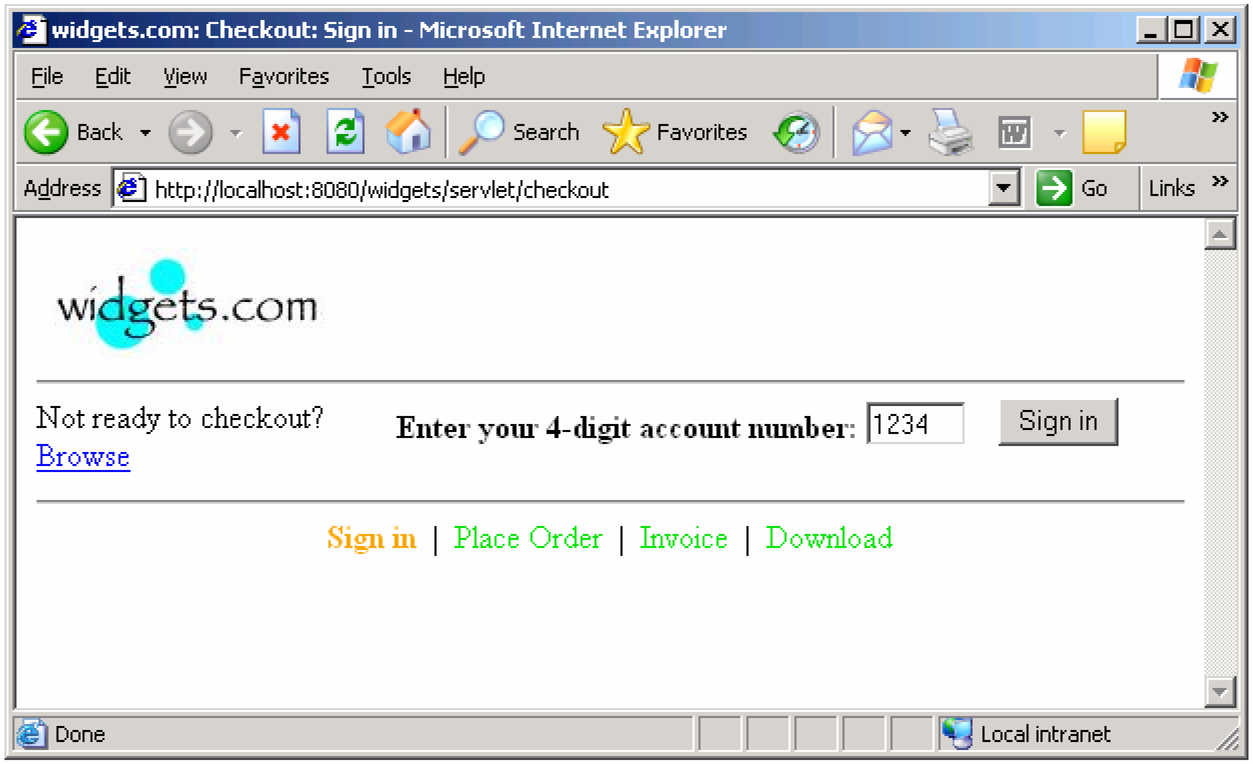
Your shopping cart.

Qty	Widget	Price
2	Menu	$25
Update	**Total**	**$50**

Ready to order? Proceed to Checkout

Browse | View Cart

After pressing the proceed to checkout button


widgets.com: Checkout: Sign in - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/widgets/servlet/checkout
Go Links
widgets.com

Not ready to checkout?
Browse

Enter your 4-digit account number: 1234 Sign in

Sign in | Place Order | Invoice | Download

Modeling the sign in form templates

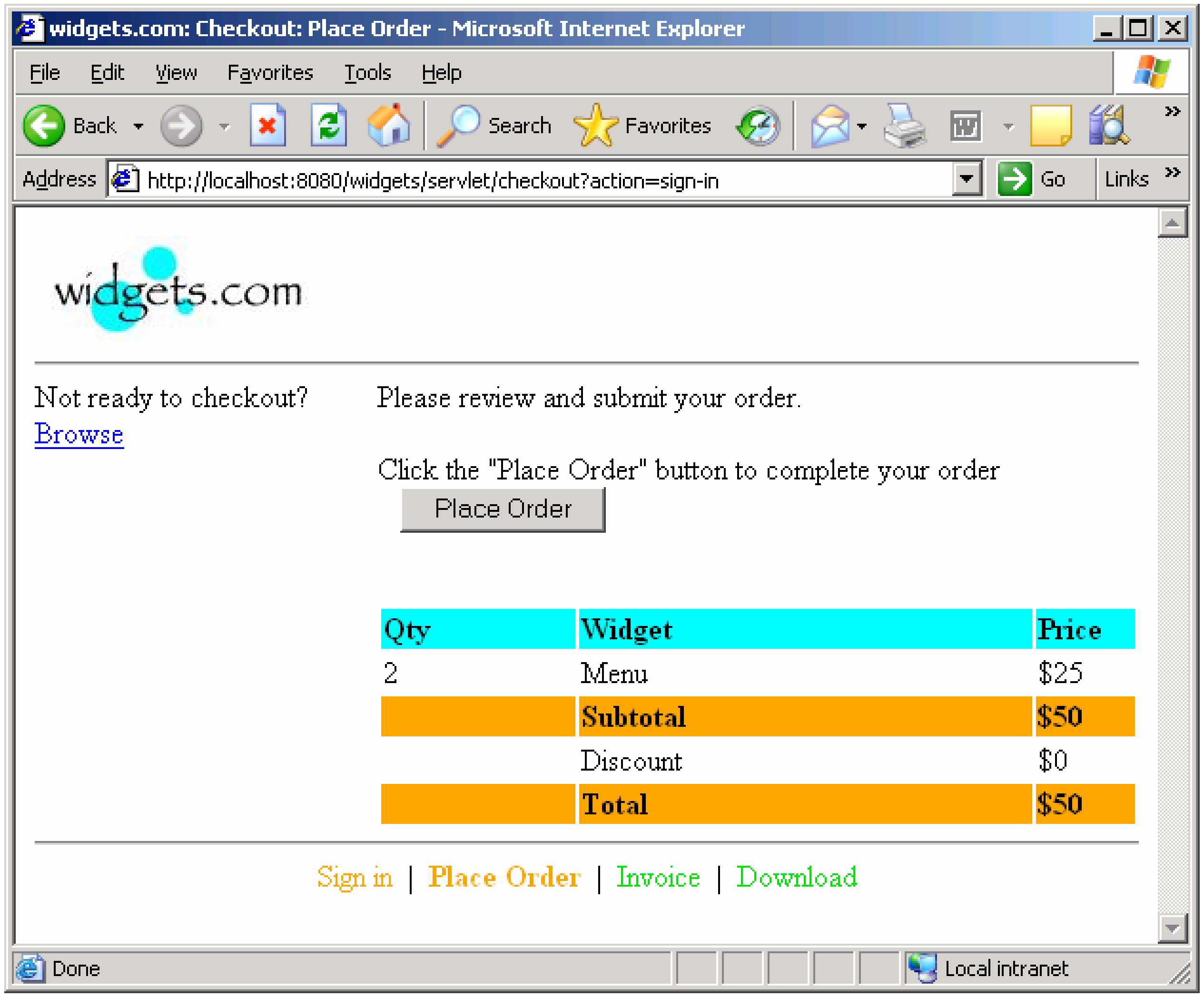
In **HTML**:

```
<form name=sign-in action=/widgets/servlet/checkout?action=sign-in method=post>
        <b>Enter your 4-digit account number:</b>
        <input type=text name=account maxlength=4 size=4>
        &nbsp;&nbsp;
        <input type=submit value='Sign in'>
</form>
```

In **TTCN-3**:

```
template browseFormType SignInForm := {
        name := "sign-in",
        formAction := "/widgets/servlet/checkout?action=sign-in",
        kindMethod := "post",
        elements := {
                {name := "account", elementValue := "" },
                {name := "action", elementValue := "sign-in" }
         }
}
```

After pressing the sign-in button


widgets.com: Checkout: Place Order - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/widgets/servlet/checkout?action=sign-in
Go Links
widgets.com

Not ready to checkout?
Browse

Please review and submit your order.

Click the "Place Order" button to complete your order

Place Order

Qty	Widget	Price
2	Menu	$25
	Subtotal	**$50**
	Discount	$0
	Total	**$50**

Sign in | Place Order | Invoice | Download

- This is a screen where the table is of interest for testing purposes

Handling tables in TTCN-3

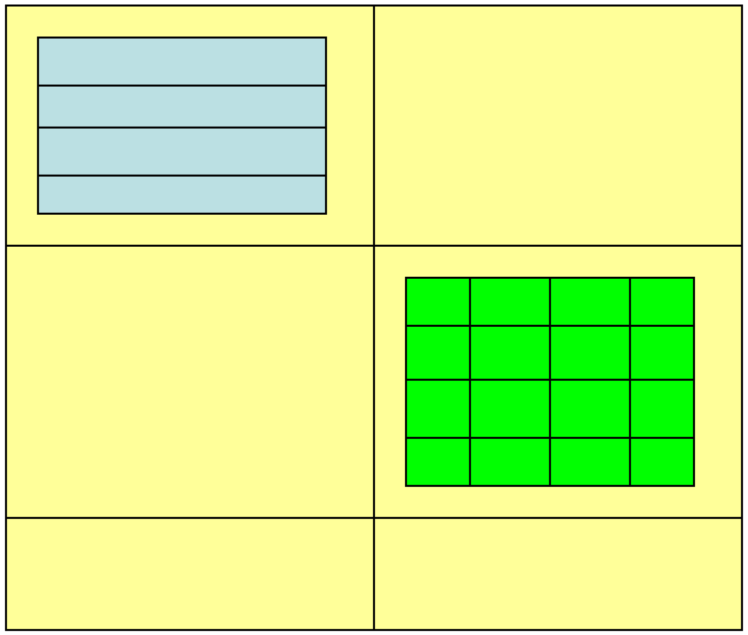
- Tables are used for formatting purposes.
- Tables are frequently nested.
- Tables contain information that can be obtained by other means, i.e. links.
- Tables are not always strict matrices. The span feature produces missing cells.
- The tester is usually not interested in nesting unless the purpose of the test is formatting

Table testing solution

- De-nest tables by collecting them in a flat set
- Make a template with the table of interest and use the superset feature.
- This will mean that the set of tables found in a page shall contain at least the member of interest

De-nesting tables

Nested tables (HTML)



De-nested tables in TTCN-3

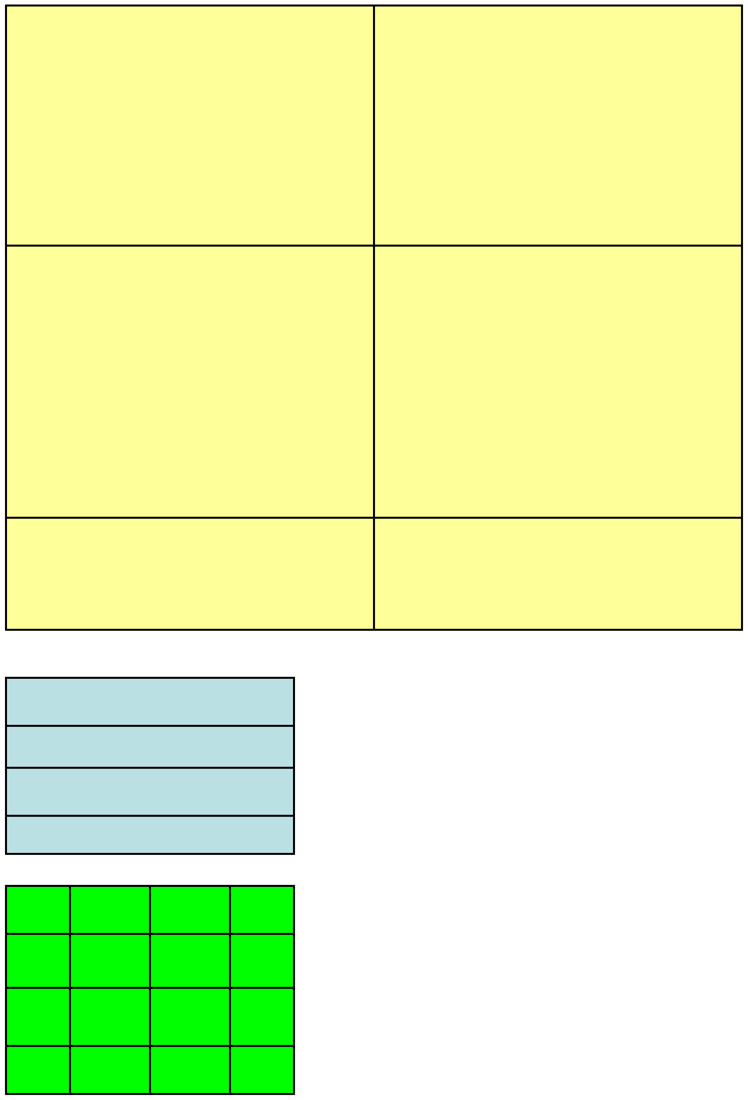
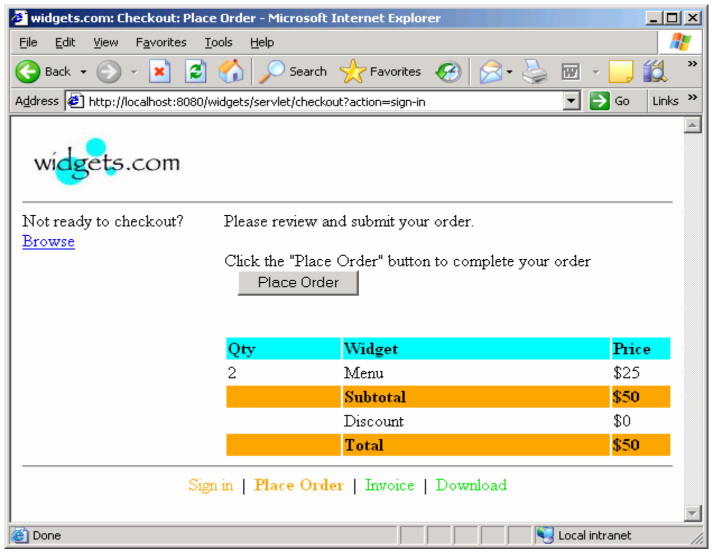


Table testing solution data types

```
type set of charstring RowCellSetType;

type record tableRowType {
        RowCellSetType cells
}

type set of tableRowType tableRowSetType;

type record TableType {
        tableRowSetType rows
}

type set of TableType TableSetType;
```

Table testing solution templates



widgets.com: Checkout: Place Order - Microsoft Internet Explorer

http://localhost:8080/widgets/servlet/checkout?action=sign-in

widgets.com

Not ready to checkout?
Browse

Please review and submit your order.

Click the "Place Order" button to complete your order

Qty	Widget	Price
2	Menu	$25
	Subtotal	**$50**
	Discount	$0
	Total	**$50**

Sign in | Place Order | Invoice | Download

- The summary table containing the shopping cart items is the only table of interest
- This table will be generated from the current shopping cart content

```
template WebPageType PlaceOrderTemplate(TableType theSummaryTable) := {
            statusCode := 200,
            title := “widgets.com Checkout:Place Order,
            content := “*Please review and submit your order*”,
            links := theCheckoutLinks,
            forms := { PlaceOrderForm },
            tables := superset (  theSummaryTable  )
}
```

Template table content generation

```
function BuildShoppingCartTable() runs on MTCType return TableType {
        …
        for(i:= 0; i < nbItems; i := i+1) {
                aRow := { cells := {
                                        int2str(theCurrentShoppingCart[i].quantity),
                                        theCurrentShoppingCart[i].name,
                                        "$" & int2str(float2int(theCurrentShoppingCart[i].price))
                                 }
                             };
                subtotal := subtotal + (theCurrentShoppingCart[i].quantity * float2int(theCurrentShoppingCart[i].price));

                theTableRows[i+1] := aRow;
        }

        aRow := { cells := {"", "Subtotal", "$" & int2str(subtotal) } };

        theTableRows[i+1] := aRow; i := i + 1;

        if(subtotal > 100) { discount := 50}

        aRow := { cells := { "", "Discount", "$" & int2str(discount) } };

        theTableRows[i+1] := aRow; i := i + 1;
        grandtotal := subtotal - discount;
        aRow := { cells := { "", "Total", "$" & int2str(grandtotal)          } };
        theTableRows[i+1] := aRow;

        theTable := { rows := theTableRows };
        return theTable;
}
```

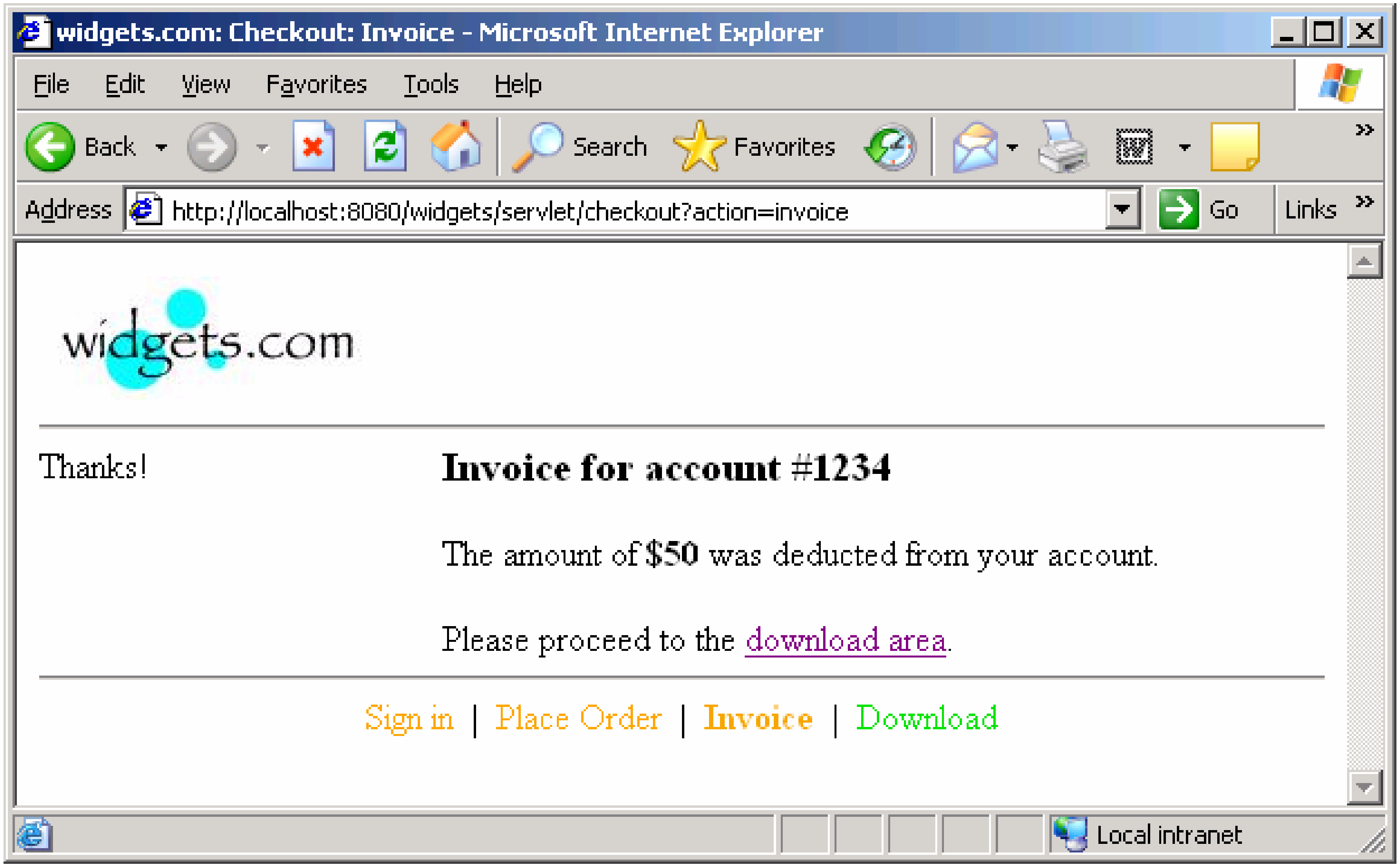
Advantages of the parametric template

- This screen is displayed every time a user has added a product to the shopping cart
- Too tedious to compute manually
- Enables long test sequences
- Enables a great variety of test scenarios

Table testing solution behavior

```
function PlaceOrderBehavior() runs on MTCType {

   alt {
         [] web_port.receive(PlaceOrderTemplate(BuildShoppingCartTable()))
                                                   -> value thePlaceOrderPage {
                     log(“shopping cart is correct”)
            }
         [] web_port.receive(IncorrectAccountTemplate) {
                     log("incorrect sign in, resubmit correct account #");
                               }
         [] ErrorBrowseBehavior() {
                     log(“unexpected message received”)
            }
}
```

After pressing the place order button



widgets.com: Checkout: Invoice - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address
Go
Links
Local intranet

http://localhost:8080/widgets/servlet/checkout?action=invoice

widgets.com

Thanks!

Invoice for account #1234

The amount of $50 was deducted from your account.

Please proceed to the download area.

Sign in | Place Order | **Invoice** | Download

A test case example

```
testcase BaseCaseTest() runs on MTCType system SystemType {
        var charstring thePressedLink;
        var ParameterValuesSetType theCurrentUpdateElements;

        map(mtc:web_port, system:system_web_port);

        web_port.send("http://localhost:8080/widgets/servlet");
        HomePageBehavior();

        clickOnLink("Browse");
        HomePageBehavior();

        clickOnLink("Utilities");
        UtilitiesBehavior();

        clickOnLink("UserInterface");
        UserInterfaceBehavior();

        clickOnLink("Menu");
        ItemDetailsBehavior(MenuTemplate);

        addToShoppingCart(1, "Menu", 25.00);
        web_port.send(addToCartTemplate);
        ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));

        modifyQuantityOrdered("Menu", 2);
        web_port.send(UpdateTemplate({ {parmName := "quantity.Menu",  parmValue := "2"} }));
```

Test case example (cont.)

```
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));
log("after updating Menu to 2 pieces");

clickOnLink("UserInterface");
UserInterfaceBehavior();

log("purchasing a Separator");

clickOnLink("Separator");
ItemDetailsBehavior(SeparatorTemplate);
addToShoppingCart(1, "Separator", 10.00);
web_port.send(addToCartTemplate);
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));

clickOnLink("FunAndGames");
FunAndGamesBehavior();

clickOnLink("UserInterface");
UserInterfaceBehavior();

clickOnLink("Menu");
ItemDetailsBehavior(MenuTemplate);
addToShoppingCart(1, "Menu", 25.00);
web_port.send(addToCartTemplate);
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));
```

Test case example (cont.)

```
clickOnLink("NewsFeeds");
NewsFeedsBehavior();

clickOnLink("UserInterface");
UserInterfaceBehavior();

log("purchasing a Separator");

clickOnLink("Separator");
ItemDetailsBehavior(SeparatorTemplate);
addToShoppingCart(1, "Separator", 10.00);
web_port.send(addToCartTemplate);
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));

clickOnLink("UserInterface");
UserInterfaceBehavior();

log("purchasing a Button");
clickOnLink("Button");
ItemDetailsBehavior(ButtonTemplate);
addToShoppingCart(1, "Button", 10.00);
web_port.send(addToCartTemplate);
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));

clickOnLink("DateAndTime");
DateAndTimeBehavior();
```

Test case example (cont.)

```
clickOnLink("UserInterface");
UserInterfaceBehavior();

log("purchasing a Panel");
clickOnLink("Panel");
ItemDetailsBehavior(PanelTemplate);
addToShoppingCart(1, "Panel", 15.00);

web_port.send(addToCartTemplate);
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));

clickOnLink("UserInterface");
UserInterfaceBehavior();

log("purchasing a Button");
clickOnLink("Button");
ItemDetailsBehavior(ButtonTemplate);
addToShoppingCart(1, "Button", 10.00);
web_port.send(addToCartTemplate);
ViewCartBehavior(currentCartViewTemplate(UpdateForm(BuildShoppingCartForm())));

web_port.send(checkoutTemplate);
SignInBehavior();

// first send the incorrect account # 15
web_port.send(SignInIncorrectSubmitTemplate);
PlaceOrderBehavior();
```

Test case example (cont.)

```
        // send a correct account # 1234
        web_port.send(SignInSubmitTemplate);
        PlaceOrderBehavior();

        web_port.send(confirmTemplate);
        InvoiceBehavior();

        ShowShoppingCart();

        setverdict(pass);
}
```

Integrating TTCN-3 and httpUnit

- Takes place in the test adapter/codec layer
- Primary activity is to transform TTCN-3 abstract data into concrete data
- Then use the concrete data when invoking httpUnit methods
- httpUnit has an influence on design of TTCN-3 data types, templates and behavior

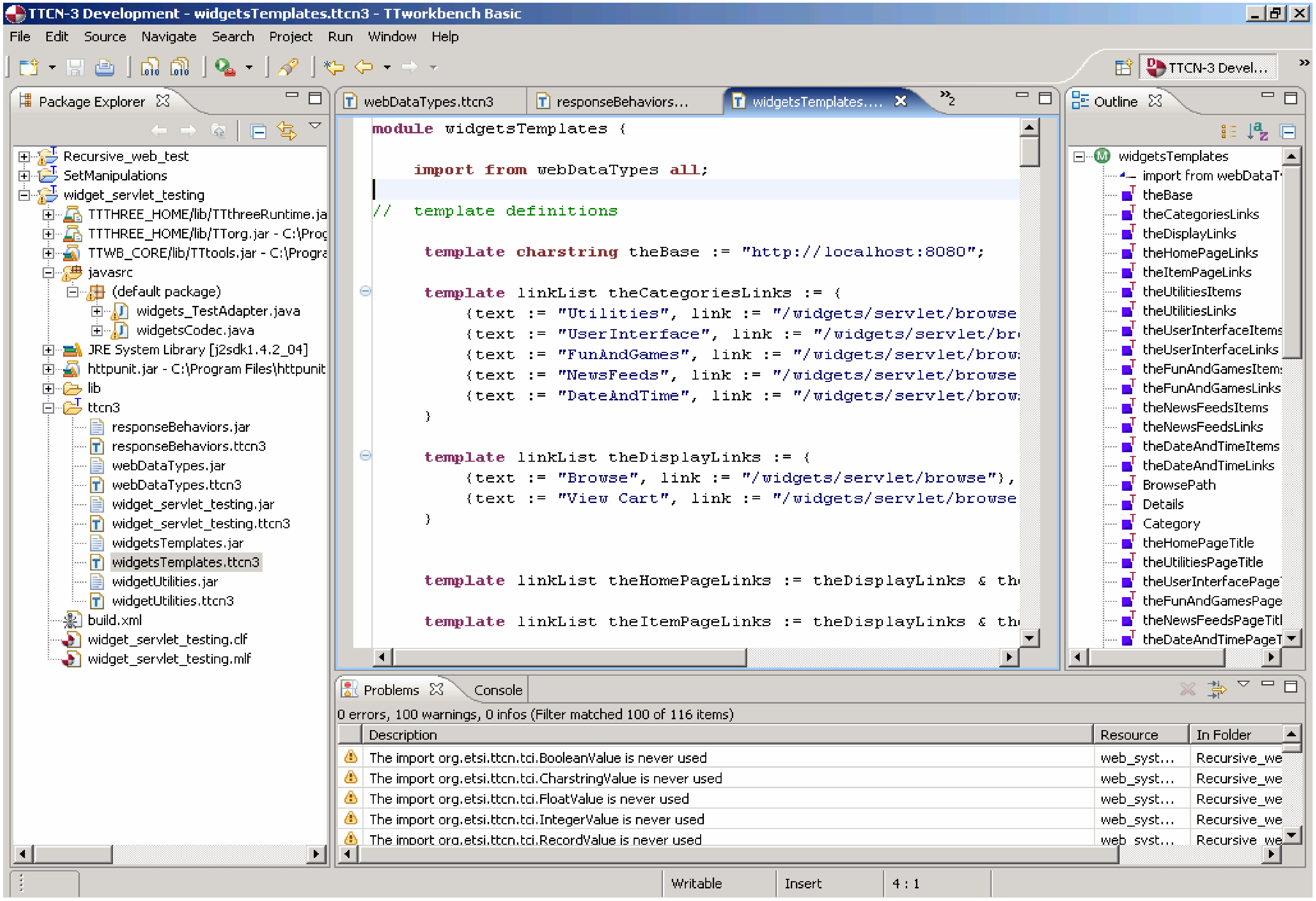
Extracting data using httpUnit in the codec

```
int theStatus = theAdapterInstance.response.getResponseCode();
theStatusValue.setInt(theStatus);

String theTitle = theAdapterInstance.response.getTitle();
theTitleValue.setString(theTitle);

String theContent = theAdapterInstance.response.getText();
theContentValue.setString(theContent);

theLinksValue = ExtractLinks(theWebPageValue);
theFormsValue =  ExtractForms(theWebPageValue);
theTablesValue = ExtractTables(theWebPageValue);
```

Form submit using httpUnit

```
SubmitButton[] theSubmitButtons = theCurrentForms[iform].getSubmitButtons();

SubmitButton theSubmitButton = null;

for(int i=0; i < theSubmitButtons.length; i++) {
        theSubmitButton = theSubmitButtons[i];

        if(theSubmitButton.getValue().equals(theFormButton)) {
                response = theCurrentForms[iform].submit(theSubmitButton);

                break;
        }
}
```

TTCN-3 module editor

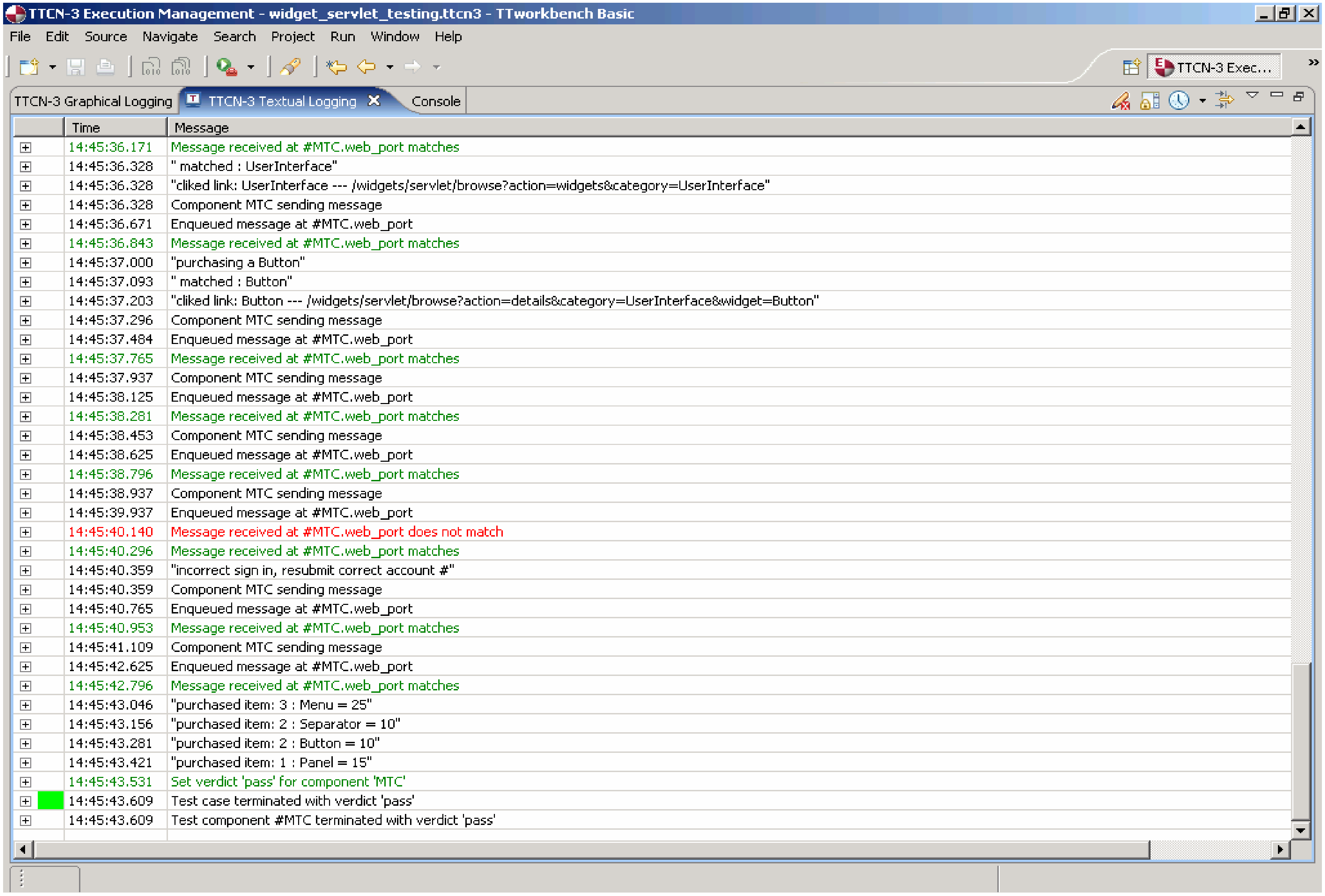


```
module widgetsTemplates {

    import from webDataTypes all;

//   template definitions

     template charstring theBase := "http://localhost:8080";

     template linkList theCategoriesLinks := {
         {text := "Utilities", link := "/widgets/servlet/browse
         {text := "UserInterface", link := "/widgets/servlet/br
         {text := "FunAndGames", link := "/widgets/servlet/brow
         {text := "NewsFeeds", link := "/widgets/servlet/browse
         {text := "DateAndTime", link := "/widgets/servlet/brow
     }

     template linkList theDisplayLinks := {
         {text := "Browse", link := "/widgets/servlet/browse"},
         {text := "View Cart", link := "/widgets/servlet/browse
     }


     template linkList theHomePageLinks := theDisplayLinks & th

     template linkList theItemPageLinks := theDisplayLinks & th
```

0 errors, 100 warnings, 0 infos (Filter matched 100 of 116 items)

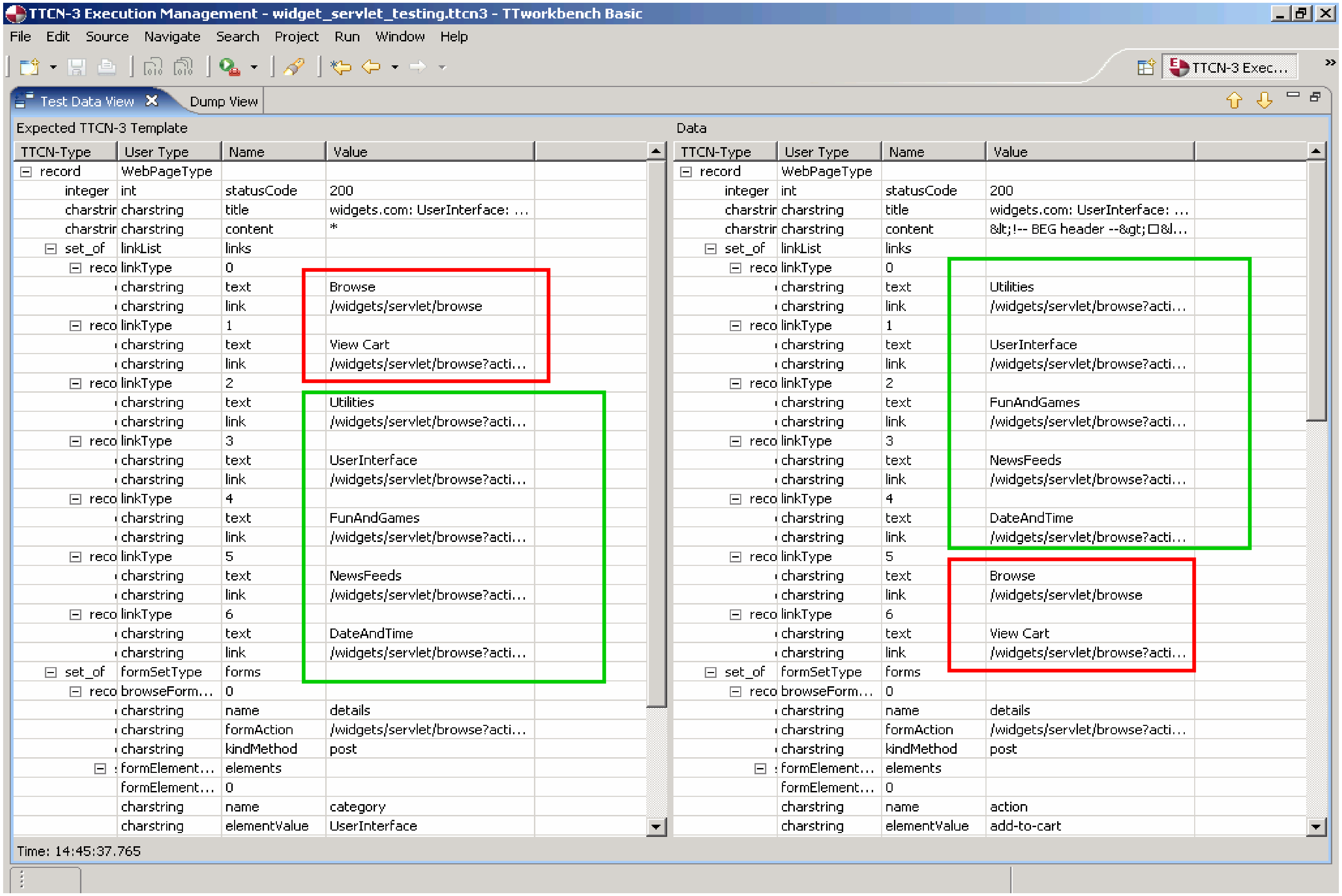
	Description	Resource	In Folder
⚠	The import org.etsi.ttcn.tci.BooleanValue is never used	web_syst...	Recursive_we
⚠	The import org.etsi.ttcn.tci.CharstringValue is never used	web_syst...	Recursive_we
⚠	The import org.etsi.ttcn.tci.FloatValue is never used	web_syst...	Recursive_we
⚠	The import org.etsi.ttcn.tci.IntegerValue is never used	web_syst...	Recursive_we
⚠	The import org.etsi.ttcn.tci.RecordValue is never used	web_syst...	Recursive_we

Test execution text view


TTCN-3 Execution Management - widget_servlet_testing.ttcn3 - TTworkbench Basic
File Edit Source Navigate Search Project Run Window Help
TTCN-3 Exec...
TTCN-3 Graphical Logging
TTCN-3 Textual Logging
Console

	Time	Message
	14:45:36.171	Message received at #MTC.web_port matches
	14:45:36.328	" matched : UserInterface"
	14:45:36.328	"cliked link: UserInterface --- /widgets/servlet/browse?action=widgets&category=UserInterface"
	14:45:36.328	Component MTC sending message
	14:45:36.671	Enqueued message at #MTC.web_port
	14:45:36.843	Message received at #MTC.web_port matches
	14:45:37.000	"purchasing a Button"
	14:45:37.093	" matched : Button"
	14:45:37.203	"cliked link: Button --- /widgets/servlet/browse?action=details&category=UserInterface&widget=Button"
	14:45:37.296	Component MTC sending message
	14:45:37.484	Enqueued message at #MTC.web_port
	14:45:37.765	Message received at #MTC.web_port matches
	14:45:37.937	Component MTC sending message
	14:45:38.125	Enqueued message at #MTC.web_port
	14:45:38.281	Message received at #MTC.web_port matches
	14:45:38.453	Component MTC sending message
	14:45:38.625	Enqueued message at #MTC.web_port
	14:45:38.796	Message received at #MTC.web_port matches
	14:45:38.937	Component MTC sending message
	14:45:39.937	Enqueued message at #MTC.web_port
	14:45:40.140	Message received at #MTC.web_port does not match
	14:45:40.296	Message received at #MTC.web_port matches
	14:45:40.359	"incorrect sign in, resubmit correct account #"
	14:45:40.359	Component MTC sending message
	14:45:40.765	Enqueued message at #MTC.web_port
	14:45:40.953	Message received at #MTC.web_port matches
	14:45:41.109	Component MTC sending message
	14:45:42.625	Enqueued message at #MTC.web_port
	14:45:42.796	Message received at #MTC.web_port matches
	14:45:43.046	"purchased item: 3 : Menu = 25"
	14:45:43.156	"purchased item: 2 : Separator = 10"
	14:45:43.281	"purchased item: 2 : Button = 10"
	14:45:43.421	"purchased item: 1 : Panel = 15"
	14:45:43.531	Set verdict 'pass' for component 'MTC'
🟩	14:45:43.609	Test case terminated with verdict 'pass'
	14:45:43.609	Test component #MTC terminated with verdict 'pass'

Analyzing the match of templates



TTCN-3 Execution Management - widget_servlet_testing.ttcn3 - TTworkbench Basic

File Edit Source Navigate Search Project Run Window Help

Test Data View | Dump View

Expected TTCN-3 Template

TTCN-Type	User Type	Name	Value
record	WebPageType		
integer	int	statusCode	200
charstrir	charstring	title	widgets.com: UserInterface: ...
charstrir	charstring	content	*
set_of	linkList	links	
reco	linkType	0	
	charstring	text	Browse
	charstring	link	/widgets/servlet/browse
reco	linkType	1	
	charstring	text	View Cart
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	2	
	charstring	text	Utilities
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	3	
	charstring	text	UserInterface
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	4	
	charstring	text	FunAndGames
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	5	
	charstring	text	NewsFeeds
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	6	
	charstring	text	DateAndTime
	charstring	link	/widgets/servlet/browse?acti...
set_of	formSetType	forms	
reco	browseForm...	0	
	charstring	name	details
	charstring	formAction	/widgets/servlet/browse?acti...
	charstring	kindMethod	post
:	formElement...	elements	
	formElement...	0	
	charstring	name	category
	charstring	elementValue	UserInterface

Data

TTCN-Type	User Type	Name	Value
record	WebPageType		
integer	int	statusCode	200
charstrir	charstring	title	widgets.com: UserInterface: ...
charstrir	charstring	content	&lt;!-- BEG header --&gt;□&l...
set_of	linkList	links	
reco	linkType	0	
	charstring	text	Utilities
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	1	
	charstring	text	UserInterface
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	2	
	charstring	text	FunAndGames
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	3	
	charstring	text	NewsFeeds
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	4	
	charstring	text	DateAndTime
	charstring	link	/widgets/servlet/browse?acti...
reco	linkType	5	
	charstring	text	Browse
	charstring	link	/widgets/servlet/browse
reco	linkType	6	
	charstring	text	View Cart
	charstring	link	/widgets/servlet/browse?acti...
set_of	formSetType	forms	
reco	browseForm...	0	
	charstring	name	details
	charstring	formAction	/widgets/servlet/browse?acti...
	charstring	kindMethod	post
:	formElement...	elements	
	formElement...	0	
	charstring	name	action
	charstring	elementValue	add-to-cart

Time: 14:45:37.765

conclusions

- TTCN-3 is very adequate for servlet testing
- Adapter/codec can be re-used for unlimited number of different web application testing
- Both ATS and adapter/codec can be seen as a TTCN-3 framework that would allow the rapid development of a wide variety of web or e-commerce applications
- TTCN-3 is very flexible and allows the development of all kinds of testing features.