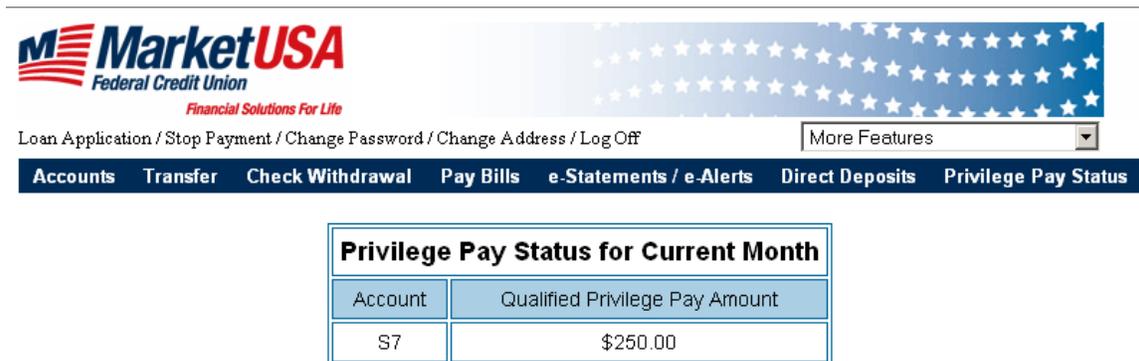


Displaying Privilege Pay Status In Internet Banking

Utilizing Ultradata's Member Exchange interface we can display any piece of data from the database. This document explains how to display a member's Privilege Pay available amount.

Example of screen in Internet Banking



The screenshot shows the MarketUSA Federal Credit Union internet banking interface. At the top left is the logo for MarketUSA Federal Credit Union with the tagline "Financial Solutions For Life". Below the logo is a navigation menu with links: Loan Application / Stop Payment / Change Password / Change Address / Log Off, and a dropdown menu for More Features. A dark blue navigation bar contains the following menu items: Accounts, Transfer, Check Withdrawal, Pay Bills, e-Statements / e-Alerts, Direct Deposits, and Privilege Pay Status. Below the navigation bar is a table titled "Privilege Pay Status for Current Month".

Privilege Pay Status for Current Month	
Account	Qualified Privilege Pay Amount
S7	\$250.00

Tasks Overview:

- Add records to UE.STDDICT that point to the Privilege Pay dictionaries we want
- Create a Member Exchange Data Object with the STDDICT items added
- Create a Member Exchange Vendor using the Data Object created
- Place asp.net page on Internet Banking server
- Allow your Internet Banking server to process asp.net pages
- Update your Internet Banking menu for this feature

Adding Items to the UE.STDDICT file:

We need to add privilege pay data. There are already dictionaries in the PPAY.CLIENT file that give us what we want. We need the qualified amount, manual amount, opt out code, and the accounts in the PPAY.CLIENT file. Below are the records that need to be added. You can use the AE tool to edit these records and copy and paste the information.

Executing AE

File >Top of "CuPpayAccount" in "UE.STDDICT", 4 lines, 26 characters.

```
001: PK
002: PPAY.CLIENT
003: UD_ACCOUNT
004:
Bottom.
```

Top of "CuPpayLimit" in "UE.STDDICT", 4 lines, 24 characters.

```
001: PK
002: PPAY.CLIENT
003: UD_LIMIT
004:
Bottom.
```

Top of "CuPpayOptout" in "UE.STDDICT", 4 lines, 23 characters.

```
001: PK
002: PPAY.CLIENT
003: OPT_OUT
004:
Bottom.
```

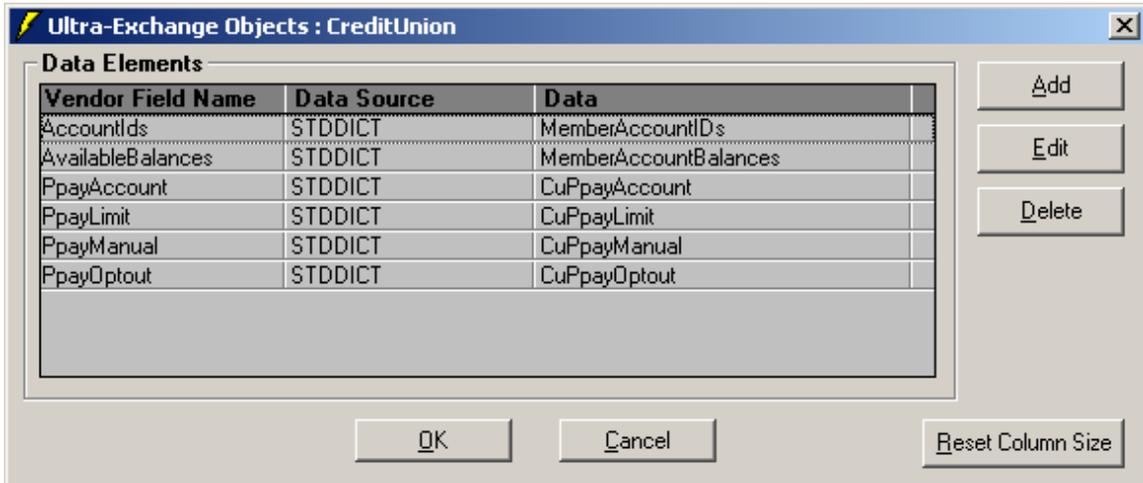
Top of "CuPpayManual" in "UE.STDDICT", 4 lines, 34 characters.

```
001: PK
002: PPAY.CLIENT
003: UD_ALTERNATE_LIMIT
004:
Bottom.
```

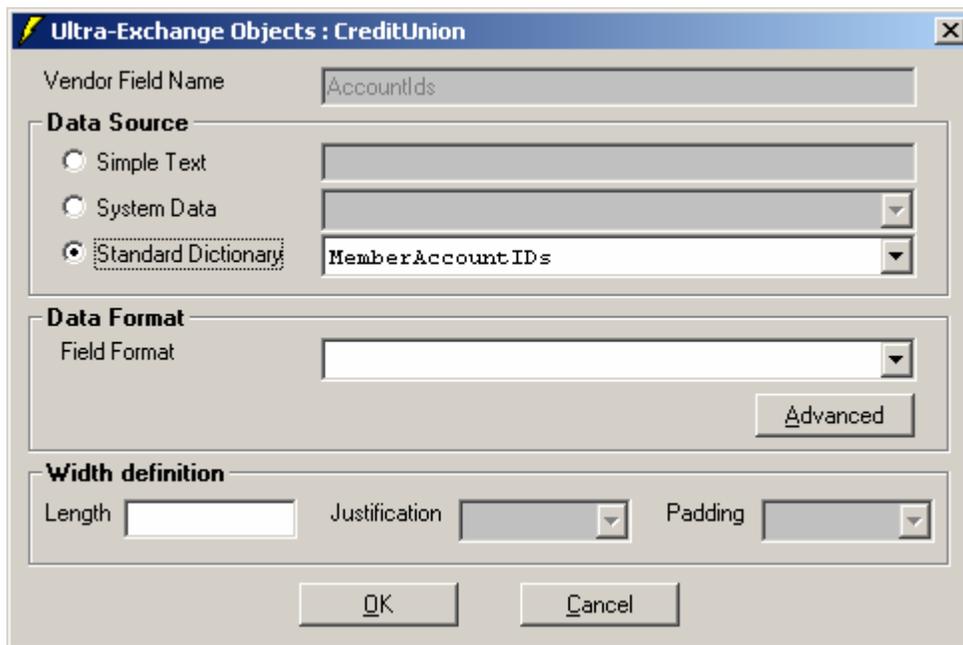
Creating the Member Exchange Data Object:

The dictionaries we added are now available in the UE Objects Tool STDDICT dropdown

Here are screenshots for the setup of the data object.



Each individual data source



Ultra-Exchange Objects : CreditUnion

Vendor Field Name: AvailableBalances

Data Source

- Simple Text
- System Data
- Standard Dictionary: MemberAccountBalances

Data Format

Field Format: Decimal (nn.nn) - MD2

Advanced

Width definition

Length: [] Justification: [] Padding: []

OK Cancel

Note that the data format does not have commas. The asp code parses the data objects by commas so a comma here will mess it up.

Ultra-Exchange Objects : CreditUnion

Vendor Field Name: PpayAccount

Data Source

- Simple Text
- System Data
- Standard Dictionary: CuPpayAccount

Data Format

Field Format: []

Advanced

Width definition

Length: [] Justification: [] Padding: []

OK Cancel

The screenshot shows a dialog box titled "Ultra-Exchange Objects : CreditUnion". It contains the following fields and options:

- Vendor Field Name: PpayLimit
- Data Source:
 - Simple Text
 - System Data
 - Standard Dictionary
- Field Format: Decimal (nn.nn) - MD2
- Advanced button
- Width definition:
 - Length: []
 - Justification: []
 - Padding: []
- Buttons: OK, Cancel

Note that the data format does not have commas. The asp code parses the data objects by commas so a comma here will mess it up.

The screenshot shows a dialog box titled "Ultra-Exchange Objects : CreditUnion". It contains the following fields and options:

- Vendor Field Name: PpayManual
- Data Source:
 - Simple Text
 - System Data
 - Standard Dictionary
- Field Format: Decimal (nn.nn) - MD2
- Advanced button
- Width definition:
 - Length: []
 - Justification: []
 - Padding: []
- Buttons: OK, Cancel

Note that the data format does not have commas. The asp code parses the data objects by commas so a comma here will mess it up.

Ultra-Exchange Objects : CreditUnion [X]

Vendor Field Name

Data Source

Simple Text

System Data

Standard Dictionary

Data Format

Field Format

Width definition

Length Justification Padding

Creating the Member Exchange Vendor:

The screenshot shows a dialog box titled "Ultra-Exchange Vendors" with a close button (X) in the top right corner. The dialog has four tabs: "Exchange Information" (selected), "Fee Information", "Application Controls", and "Member Classification". The "Exchange Information" tab contains the following fields:

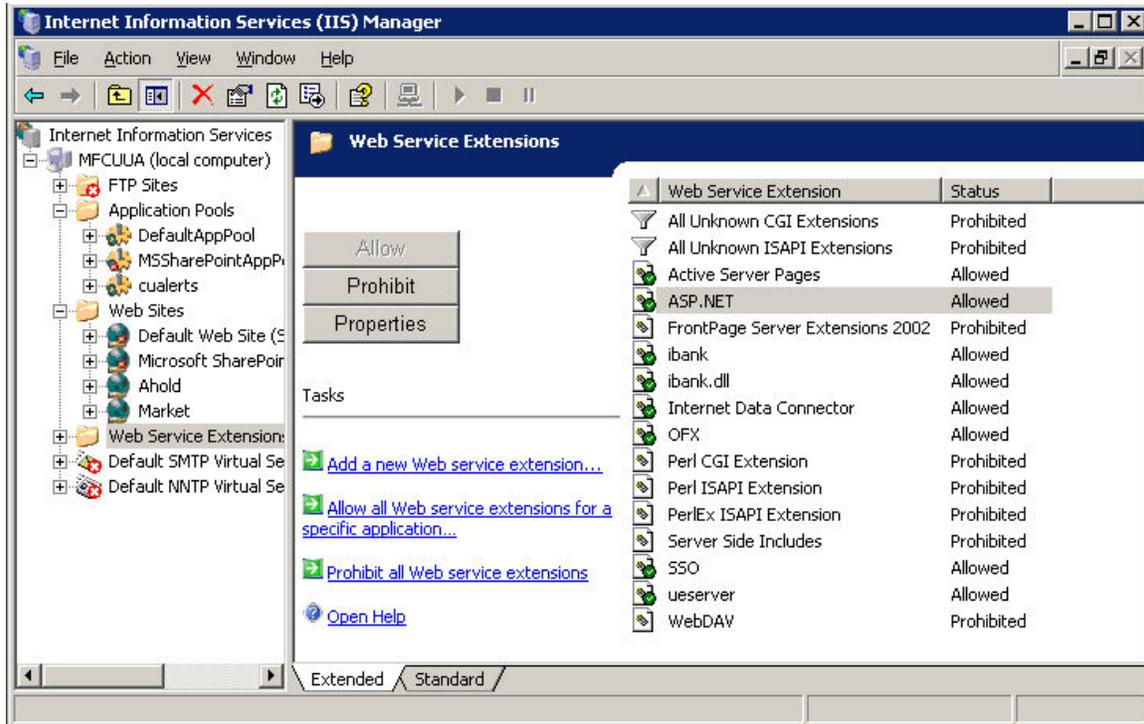
Vendor Name	CreditUnion		
Vendor Site Description	Credit Union use		
Vendor Site URL	https://hb.marketusafcu.com/UA2002/ppay.aspx		
Client Certificate for Vendor Site URL			
Vendor Call Back IP			
Financial Institution ID	01		
Password			
Data Object	CreditUnion		
Disclosure File Name			
Timeout Length (In Minutes)	10	Use	Partial Browser Screen
Exchange Method	GET		
Additional Encryption	None	Set Accounts	

At the bottom of the dialog are "OK" and "Cancel" buttons.

Change the "Vendor Site URL" to match your website and where you choose to put the page. Assign the Data Object to what you named it when you created it above.

Configure IIS to allow asp.net:

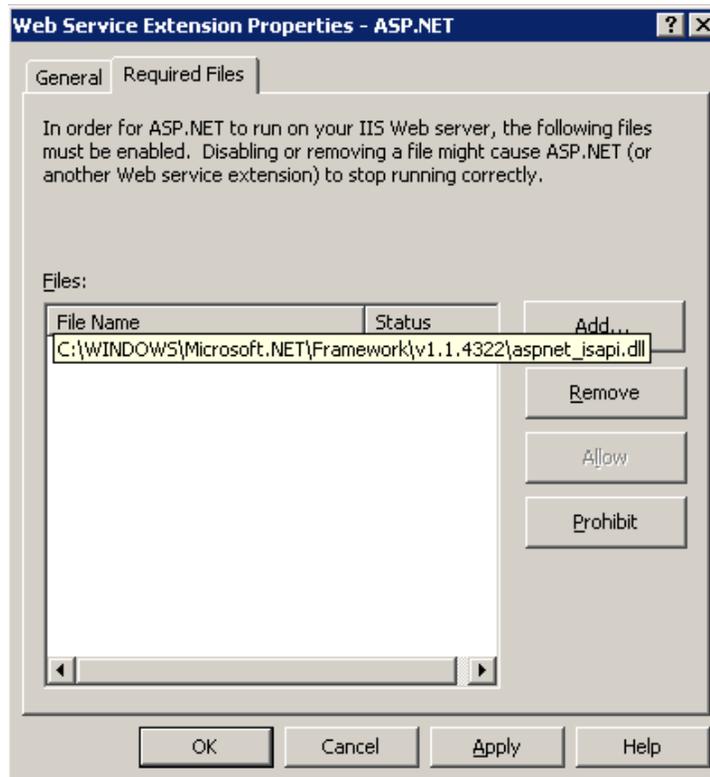
In the web service extensions make sure ASP.NET is allowed.



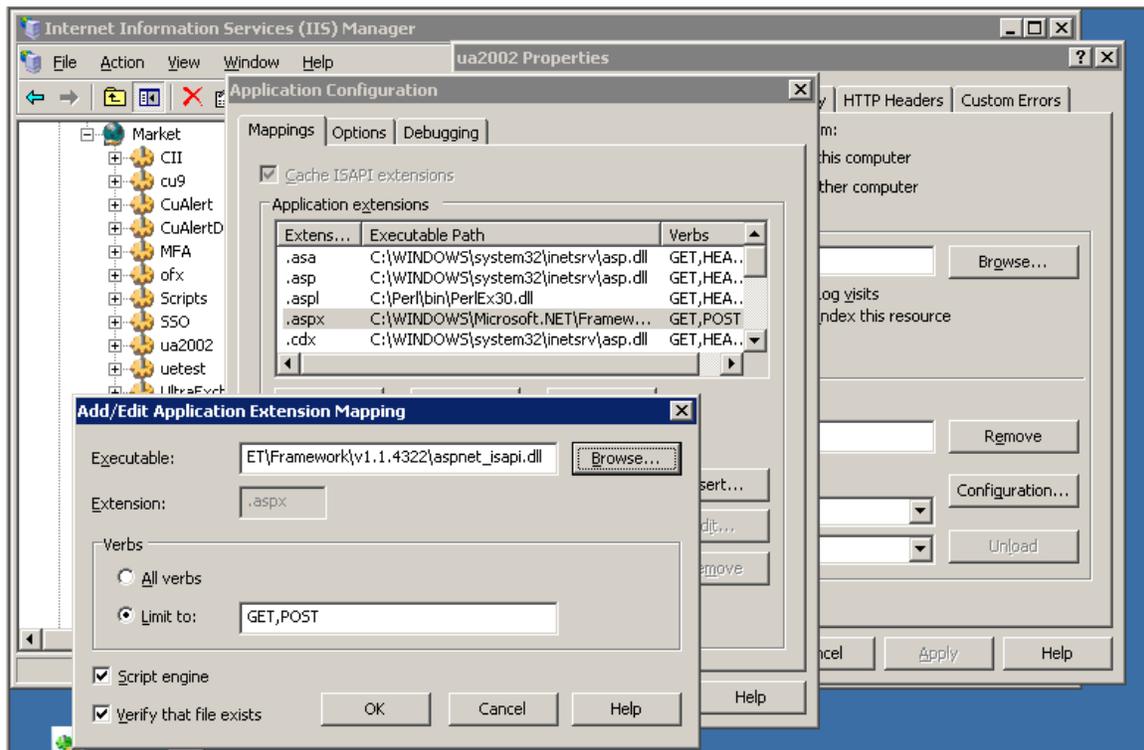
If the extension is not already there then select “Add a new Web service extension” and make sure it points to the correct program to handle it.

C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\aspnet_isapi.dll

Your version of asp.net may vary so use the newest version.



Configure the virtual directory that you will be placing the asp.net page in to handle the aspx extension. Make sure the executable is the same that you allowed for the web service extension.



Note: In this screenshot the Verbs are limited to GET and POST. The current version of this process uses GET but will be updated to use POST. Follow your security practices in how you limit the Verbs.

Your IIS server should now be configured to serve asp.net pages.

Add the asp.net page to your Internet Banking server:

Download the ppay.zip file. Extract the ppay.aspx and cuinfo.css files to your server. Save the ppay.aspx file in the same location you specified in the Member Exchange Vendor setup. Save the css file to ua2002/styles. If you wish to put it in a different place be sure to update the link within ppay.aspx. Update the color scheme as necessary.

www.ghrconcepts.com/FreeDownloads/ppay.zip

Add the option to view Privilege Pay status to your Internet Banking menu:

This is located in the frmenu.htm file.

Depending on how your menu is setup you can either have the link use javascript to submit the form or as a form button. The form is similar to the regular Internet Banking form but with an added input type of vendor. This is where you specify the vendor name setup in the Member Exchange Vendors tool.

The example below uses javascript to submit the form.

```
<form target="main" action="/scripts/ibank.dll" method="post" name="weform">
  <input type="hidden" name="homepath" value="%HOMEPATH%">
  <input type="hidden" name="Func" value="WELaunch">
  <input type="hidden" name="Vendor" value="">
</form>
```

```
<a href="#" onfocus="this.blur()" onclick="document.weform.Vendor.value =
'CreditUnion';document.weform.submit()">Privilege Pay Status</a>
```

If you are using an unmodified version of the frmenu and each menu button has it's own form then this is an example.

```
<form target=main action="/scripts/ibank.dll" method=post>
  <input type=hidden name=homepath value=%HOMEPATH%>
  <input type="HIDDEN" Name=Func Value="WELaunch">
  <input type="hidden" name="Vendor" value="CreditUnion">
  <input type="IMAGE" border=0 src="/%HOMEPATH%/images/Yourimage.gif">
```

</form>

Implementation Complete:

This concludes how to display the Privilege Pay status in Internet Banking. You can use these methods to display any other pieces of data you wish from the Ultradata server.

Courtesy,
GHR Concepts LLC
www.ghrconcepts.com