

Creating Inbox Items

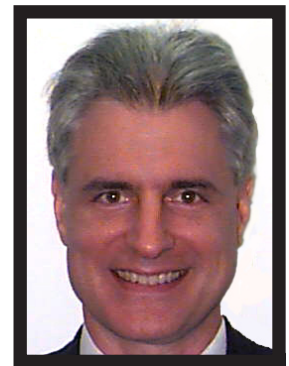
An Oracle “How To” Paper

Presented By:

Jeremy Blankenship



William Indest



Create Inbox Item

*All generalizations are false,
including this one.*

Mark Twain

Our generalization: We wanted to add items (Cases and Tips) to the Inbox independent of either the approval process or the Task Based User Interface. We created a button that when clicked creates a new Inbox item from the selected record.

Step 1: Create the Inbox Type

1. Login to any Siebel client as SADMIN.
2. Go to the Administration-Inbox Screen > All Inbox Types View.
3. Look to see if a suitable Inbox Type already exists.
4. Enter the drilldown information in the Views tab.
5. Advanced Inbox Users (and wannabes): It is possible that a suitable Inbox Type exists and you don't need to create a new Inbox Type. Inbox Types are arranged by Business Object and by Category (Approvals, Events, etc). Inbox Types have different properties such as the ability to transfer the Inbox items and the ability to include timeout information.
6. Get a permanent marker and write the name of your newly created Inbox Type on your hand.
7. Here is what we used: (note the view for the drilldown is shown.)

Administration - Inbox | Administration - Search | Home | Calendar

All Inbox Types | All Inbox Items | All Submitted Items | All Completed Items

Inbox Types | Menu | New | Delete | Query

Name	Business Object	Category	Action Type
Case	HLS Case	APPROVALS	UINBOX_STATUS_TYPE
> Case Creation	HLS Case	EVENT	UINBOX_STATUS_TYPE
Tip	PUB Tip	APPROVALS	UINBOX_STATUS_TYPE

More Info | Translations | Views | Actions

Menu | New | Delete | Query

View	Sequence #	SmartScript
> HLS Case List View	1	

In the Siebel application

*If you die in an elevator, be sure
to push the Up button.*

Sam Levenson

In Siebel Tools

Step 2: Adding the button and code to the Applet

1. Create a new record under Controls for your applet. Add CreateInboxItem to the MethodInvocation property; HTML Type is MiniButton.
2. Add Create Inbox Item to the Caption – String Override property.
3. Invoke Edit Web Layout for your applet.
4. Drag your Minibutton Control and drop it in an empty slot.
5. Save your work.
6. Open the CreateInboxItem **PreCanInvoke.txt** file; cut and paste to the PreCanInvokeMethod of your applet.
7. Open the **CaseCreateInboxItem.txt** file; cut and paste to the Pre-InvokeMethod of your applet.
8. You should alter sInboxName, sInboxTypeName and sOwner as appropriate for your own use. Look at their values in red in the code.

sInboxName	Displayed in the Inbox's Name column. In our code example we use values from the current Business Component and put them in the field.
sInboxTypeName	The value of the Inbox Type. Use what you wrote on your hand in Step 1.
sOwner	The ROW_ID of the Inbox Item's owner. Position ROW_ID values cannot be used. Can either be a value that is part of the current record such as Created By or Owner Id or it can be hardcoded.

9. Compile and check.
10. Drink. Repeat.

*Eggheads unite! You have
nothing to lose but your yolks.*

Adlai Stevenson

Ah, code...

eScript code snippet 1

```
function WebApplet_PreCanInvokeMethod (MethodName, &CanInvoke)
{
    if (MethodName == "CreateInboxItem")
    {
        CanInvoke = "TRUE";
        return (CancelOperation);
    }
    return (ContinueOperation);
}
```

eScript code snippet 2

```
function WebApplet_PreInvokeMethod (MethodName)
{
    if (MethodName == "CreateInboxItem")
    {
        var sInboxName =
        "Case Creation: Case Id"+this.BusComp().GetFieldValue("Id");
        var sInboxTypeName = "Case Creation";
        var sOwner =
        this.BusComp().GetFieldValue("Created By");

        var bsUniversalInbox = TheApplication().
        GetService("Universal Inbox");

        var psInputs = TheApplication().NewPropSet();
        var psOutputs = TheApplication().NewPropSet();

        with (psInputs)
        {
            SetProperty("InboxName",sInboxName);
            SetProperty("InboxTypeName",sInboxTypeName);
            SetProperty("ObjectId",this.BusComp().
            GetFieldValue("Id"));
            SetProperty("OwnerPartyId",sOwner);
            SetProperty("InboxPartyId",sOwner);
        }

        bsUniversalInbox.InvokeMethod("CreateInboxEx",psInputs,psOutputs);

        bsUniversalInbox = null;
        return (CancelOperation);
    }
    return (ContinueOperation);
}
```



Creating Inbox Items

Authors: Jeremy Blankenship and William Indest

Managers: James Perry and Todd Kyle

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2007, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

End