How to expand Default Script functions for browser tools in custom.js.

Ross Glover - 2022-06-22 - Comments (0) - Temenos FAQs

temenos

The Custom.js file resides in the Transact BrowserWeb.war file and it contains a number of EFS Script functions that are called when adding web browser tools like PDF Preview and eSignatureScan.

These instructions are only relevant to the older Transact installations.
Whilst the scripts are supplied by Formpipe during installation, they are often updated and modified during this process, so the versions below may not always match exactly what is implemented. Despite this, the principles that are discussed can be applied to fit any version of the script.

The filenames and paths mentioned are only relevant for ModelBank on JBOSS. The custom.js file containing these scripts may be in a different .war file or in a different path.

Below is an example of the EFS Signature Capture button being added to the browser. It is linked to a function in the custom.js file called:
`efs_version_launchPdfRequest_signature()`

The function is run on the ENQuiry screen in Transact by editing the Enquiry and calling the function on a tool.

Below are two functions that your Formpipe Project manager will have added into custom.js if these specific Transact interfaces were installed during the project. Whilst they are very similar,`efs_version_launchPDFRequest_signature()` contains less code and includes the ability to record the version number, compared to `efs_version_launchPDFRequest()`.

You can customise these if you want to alter the functionality.

For example, if you use `efs_version_launchPDFRequest_signature()` you will notice two differences in the output, as compared to the output from `efs_version_launchPDFRequest()`:

- You may find that certain tags containing data are missing from the XML file that is generated and sent to Lasernet when hitting the “SIGN DOC” button
- You will find that the tag containing the version number is also missing from the XML file

The launchPDFRequest code

```
function efs_version_launchPdfRequest() {
    // store current form values

    try {


        var appreqForm = FragmentUtil.getForm(currentForm_GLOBAL);
        var previousStylesheet = appreqForm.clientStyleSheet.value;
        var previousTarget = appreqForm.target;
        var previousWindowName = appreqForm.windowName.value;

        // create popup and update form values
        appreqForm.windowName.value =
FragmentUtil.getWindowOrFragmentName();
        appreqForm.clientStyleSheet.value =
"/transforms/custom/itransform-version.xsl";
        appreqForm.target = createResultWindow("NEW", 900, 600);
        if (appreqForm.RecordRead.value == 1) {
            // we're editing the record, so run validate
            appreqForm.ofsOperation.value = _VALIDATE_;
            appreqForm.ofsFunction.value = "I";

            // update transaction id for security validation
            var transactionId = getFormFieldValue(currentForm_GLOBAL,
"transactionId");
            transactionId = validateTransId(transactionId);
            setFormFieldValue(currentForm_GLOBAL, "transactionId",
transactionId);
```

```
            checkChangedFields();
        } else {
            // we're viewing the record, so run build
            appreqForm.ofsOperation.value = _BUILD_;
            appreqForm.ofsFunction.value = "S";
        }
        FragmentUtil.submitForm(appreqForm);

        // reset form values
        appreqForm.clientStyleSheet.value = previousStylesheet;
        appreqForm.target = previousTarget;
        appreqForm.windowName.value = previousWindowName;


    } catch (ex) {
        alert(ex.message);
    }

}
```

The lunchPDFRequest_signature code

```
function efs_version_launchPdfRequest_signature() {

    //Get the form
    var appreqForm = FragmentUtil.getForm(currentForm_GLOBAL);

    //Add/update the correlationId and oneWay flag
    var correlationId = _efs_createCorrelationId();
    _efs_addOrUpdateInputWithValue(appreqForm, _efs_correlationId,
correlationId);
    _efs_addOrUpdateInputWithValue(appreqForm, _efs_noResponse,
"true");

    //update the style sheet
    appreqForm.clientStyleSheet.value =
"/transforms/custom/itransform-version.xsl";

    //set the global transaction id for some reason
    transactionId = appreqForm.transactionId.value;
```

```
    //grab the version and update
    var savedVersion = appreqForm.version.value;
    appreqForm.version.value = "";

    //grab the routine args and update
    var savedRoutineArgs = appreqForm.routineArgs.value;
    appreqForm.routineArgs.value = "1";

    //grab the target and update to a background iframe
    _efs_addHiddenIFrame(_efs_backgroundWorker_response);
    var savedTarget = appreqForm.target;
    appreqForm.target = _efs_backgroundWorker_response;

    //grab the window name and update
    var savedWindowName = appreqForm.windowName.value;
    appreqForm.windowName.value = window.name;

    //some other form changes
    appreqForm.routineName.value = "";
    appreqForm.ofsFunction.value = "I";
    appreqForm.ofsOperation.value = _BUILD_;
    appreqForm.requestType.value = _OFS__APPLICATION_;

    //submit the form and revert saved changes
    FragmentUtil.submitForm(appreqForm);
    appreqForm.target = savedTarget;
    appreqForm.windowName.value = savedWindowName;
    appreqForm.version.value = savedVersion;
    appreqForm.routineArgs.value = savedRoutineArgs;
    appreqForm.clientStyleSheet.value = "";

    //start the client
    _efs_addHiddenIFrame(_efs_backgroundWorker_client);
    //var appUrl = "pdmclient://Signature
Application/search=eSignature/parameters={correlationId:"+
correlationId + "}/engine=tablet";
    var appUrl = "pdmclient://Signature
Application/search=eSignature/parameters={correlationId:" +
correlationId + "}/engine=wacom";
    window.open(appUrl, _efs_backgroundWorker_client);
}
```

While they are similar, the biggest difference with the ESignature function is that some of the tags and data, as well as the version number, is not sent in the XML. To add this functionality into the ESignature interface, you need to replace the existing code:

```
    appreqForm.ofsFunction.value = "I";
    appreqForm.ofsOperation.value = _BUILD_;
```

with the following code from `efs_version_launchPDFRequest()`

```
        if (appreqForm.RecordRead.value == 1) {
            // we're editing the record, so run validate
            appreqForm.ofsOperation.value = _VALIDATE_;
            appreqForm.ofsFunction.value = "I";

            // update transaction id for security validation
            var transactionId = getFormFieldValue(currentForm_GLOBAL,
"transactionId");
            transactionId = validateTransId(transactionId);
            setFormFieldValue(currentForm_GLOBAL, "transactionId",
transactionId);
            checkChangedFields();
        } else {
            // we're viewing the record, so run build
            appreqForm.ofsOperation.value = _BUILD_;
            appreqForm.ofsFunction.value = "S";
        }
```

It can be seen from line of code in `efs_version_launchPDFRequest_signature()` that version is being set to blank: `appreqForm.version.value = "";`.

This means that it will not appear in the XML. However, if the above line is commented out, the function will set the version number and it will be sent in the XML.

The function below shows the result of making the above edits to the standard `efs_version_launchPDFRequest_signature()` function in custom.js. Whilst some functions will differ slightly in appearance, the outcome should be identical if the edits are carried out carefully.

```
function efs_version_launchPdfRequest_signature() {
```

```
    //Get the form
    try {
        document.getElementById("version").value =
document.getElementById("version").value + ".SIGN";
        var appreqForm = FragmentUtil.getForm(currentForm_GLOBAL);
        //Add/update the correlationId and oneWay flag
        var correlationId = _efs_createCorrelationId();
        _efs_addOrUpdateInputWithValue(appreqForm,
_efs_correlationId, correlationId);
        _efs_addOrUpdateInputWithValue(appreqForm, _efs_noResponse,
"true");
        //update the style sheet
        appreqForm.clientStyleSheet.value =
"/transforms/custom/itransform-version.xsl";
        //set the global transaction id for some reason
        transactionId = appreqForm.transactionId.value;
        //grab the version and update
        var savedVersion = appreqForm.version.value;
        //appreqForm.version.value = "";
        //grab the routine args and update
        var savedRoutineArgs = appreqForm.routineArgs.value;
        appreqForm.routineArgs.value = "1";
        //grab the target and update to a background iframe
        _efs_addHiddenIFrame(_efs_backgroundWorker_response);
        var savedTarget = appreqForm.target;
        appreqForm.target = _efs_backgroundWorker_response;
        //grab the window name and update
        var savedWindowName = appreqForm.windowName.value;
        appreqForm.windowName.value = window.name;
        appreqForm.routineName.value = "";
        if (appreqForm.RecordRead.value == 1) {
            // we're editing the record, so run validate
            appreqForm.ofsOperation.value = _VALIDATE_;
            appreqForm.ofsFunction.value = "I";
            // update transaction id for security validation
            var transactionId = getFormFieldValue(currentForm_GLOBAL,
"transactionId");
            transactionId = validateTransId(transactionId);
            setFormFieldValue(currentForm_GLOBAL, "transactionId",
transactionId);
            checkChangedFields();
```

```
        } else {
            // we're viewing the record, so run build
            appreqForm.ofsOperation.value = _BUILD_;
            appreqForm.ofsFunction.value = "S";
        }
        //some other form changes
        //appreqForm.ofsFunction.value="S";
        //appreqForm.ofsOperation.value=_BUILD_;
        appreqForm.requestType.value = _OFS__APPLICATION_;
        //submit the form and revert saved changes
        FragmentUtil.submitForm(appreqForm);
        appreqForm.target = savedTarget;
        appreqForm.windowName.value = savedWindowName;
        appreqForm.version.value = savedVersion;
        appreqForm.routineArgs.value = savedRoutineArgs;
        appreqForm.clientStyleSheet.value = "";
        //start the client
        _efs_addHiddenIFrame(_efs_backgroundWorker_client);
        //var appUrl = "pdmclient://Signature
Application/search=eSignature/parameters={correlationId:"+
correlationId + "}/engine=tablet";
        var appUrl = "pdmclient://Signature
Application/search=eSignature/parameters={correlationId:" +
correlationId + "}/engine=wacom";
        window.open(appUrl, _efs_backgroundWorker_client);
    } catch (ex) {
        alert(ex.message);
    }
}
```