Base Language - Feature #2013

implement the SELF system handle

02/19/2013 05:15 PM - Greg Shah

Status:	Closed	**Start date:**	02/19/2013
Priority:	Normal	**Due date:**	07/05/2013
Assignee:		**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:	Runtime Support for Server Features		
billable:	No	**version:**	
vendor_id:	GCD		

Description

Subtasks:

Feature # 2014: conversion support for SELF system handle	**Closed**
Feature # 2015: runtime support for SELF system handle	**Closed**

History

#1 - 02/19/2013 05:19 PM - Greg Shah

- Estimated time deleted (24.00)

The SELF system handle returns the handle to the resource currently being processed in a trigger or other callback processing.

For milestone 7, this is only used in a SAX callback (an arbitrary procedure in the app), it must access the corresponding SAX resource handle.

These attributes are in use:

PRIVATE-DATA
LOCATOR-LINE-NUMBER
LOCATOR-COLUMN-NUMBER

Of course, it's use for other resources should not be disabled by adding this capability.

#2 - 02/21/2013 05:58 AM - Constantin Asofiei

How I see this working: we have a SelfManager class which manages a stack of "self" handles. Each event responsible of creating a new SELF handle must call SelfManager.pushSelf(<handle>) before calling the trigger/procedure/etc and will call SelfManager.popSelf() after calling the trigger/procedure/etc. KW_SELF will be converted to SelfManager.self() and will act as a real handle. Any i.e. ch = self:private-data calls will be converted to:

```
ch.assign(handle.unwrap(SelfManager.self()).getPrivateData());
```

#3 - 02/21/2013 06:23 AM - Constantin Asofiei

- File ca_upd20130221c.zip added

- Status changed from New to WIP

Attached update with proposed solution (built on top of revision 10179). With this approach, all attribute/method access convert naturally and we have little impact on the generated code.

#4 - 02/21/2013 10:20 AM - Greg Shah

I like it.

#5 - 02/21/2013 12:10 PM - Constantin Asofiei

I'm leaving a conversion to run over night, hopefully all SELF-handle usage will convert nicely.

#6 - 02/22/2013 03:18 AM - Constantin Asofiei

- File ca_upd20130222a.zip added

Fixed a conversion error, I'm running conversion again.

#7 - 02/22/2013 03:20 AM - Constantin Asofiei

- File ca_upd20130222c.zip added

~~Added testcases.~~ they are for session handle, I've deleted it from here.

#8 - 02/22/2013 03:22 AM - Constantin Asofiei

- File ca_upd20130222d.zip added

Added testcases (previous were for session handle)

#9 - 02/22/2013 03:22 AM - Constantin Asofiei

- File deleted (ca_upd20130222c.zip)

#10 - 02/22/2013 06:39 AM - Constantin Asofiei

- File ca_upd20130222e.zip added

Conversion has finished, there was a APPLY ... TO SELF problem, fixed in attached update. All other changes look good. I'm running conversion again.

#11 - 02/22/2013 12:14 PM - Constantin Asofiei

Conversion regression testing has passed - compilation was OK, all changes are expected.

#12 - 02/22/2013 12:34 PM - Greg Shah

Code Feedback:

1. refName is a dead variable and can be removed from methods_attributes.rules.

2. How do we emit the frame/widget getter for static widget (not handles) attribute/method dereferencing? It would be useful to explain this in a comment in methods_attributes.rules.

Otherwise, it looks good.

Am I correct in understanding that all that remains on the runtime side is to review all event types (other than UI triggers which are already implemented) and implement the handle push/pop properly. That should not take long at all, as far as I can guess.

Be ready to merge up. We will go into testing later today.

#13 - 02/22/2013 01:33 PM - Constantin Asofiei

- File ca_upd20130222h.zip added

> 1. refName is a dead variable and can be removed from methods_attributes.rules.

OK, removed

> 2. How do we emit the frame/widget getter for static widget (not handles) attribute/method dereferencing? It would be useful to explain this in a comment in methods_attributes.rules.

something like this:

```
form i with frame f1.
i:row in frame f1 = 1.
```

converts to this:

```
f1Frame.widgeti().setRow(new integer(1));
```

I've added notes to methods_attributes.rules

> Am I correct in understanding that all that remains on the runtime side is to review all event types (other than UI triggers which are already implemented) and implement the handle push/pop properly. That should not take long at all, as far as I can guess.

Correct.

Attached update is merged with 10187.

#14 - 02/22/2013 01:46 PM - Greg Shah

The code looks good.

```
      This rule-set covers only the cases when an attribute/method is accessed using a handle ref
      (returned by a method, a variable, system handle, etc). All cases of widget attributes
      and methods accessed using the widget (with or without a frame qualifier) are managed by
      the frame-related rules, as annotations/frame_scoping, covert/ui_statements and
      convert/frame_generator.
```

I think it might be more correct to say that the rule-set only handles the resource reference for handle types. In the cases where static frame/widget/buffer resources are accessed using the COLON operator, the resource reference is emitted elsewhere but the attribute or method part is still emitted by methods_attributes.rules, right?

#15 - 02/22/2013 02:17 PM - Constantin Asofiei

- File ca_upd20130222h.zip added

> I think it might be more correct to say that the rule-set only handles the resource reference for handle types. In the cases where static frame/widget/buffer resources are accessed using the COLON operator, the resource reference is emitted elsewhere but the attribute or method part is still emitted by methods_attributes.rules, right?

Yup, you are correct, I was misguided by the frame-related options which are emitted in various places (and the setters/getters look similar to the attributes).

Attached update has the same name as previous one.

#16 - 02/22/2013 04:27 PM - Greg Shah

This is the latest version merged up to bzr 10188. It will go into testing next.

#17 - 02/22/2013 04:27 PM - Greg Shah

- File ca_upd20130222j.zip added

#18 - 02/22/2013 04:30 PM - Greg Shah

- File deleted (ca_upd20130222j.zip)

#19 - 02/22/2013 04:31 PM - Greg Shah

- File ca_upd20130222j.zip added

Here is the right version.

#20 - 02/22/2013 10:14 PM - Greg Shah

Passed conversion testing. Checked into bzr as 10189.

#21 - 02/23/2013 01:30 PM - Constantin Asofiei

- File ca_upd20130223h.zip added

Fixed initialization of context-local data in SelfManager.

#22 - 02/23/2013 05:43 PM - Greg Shah

The change looks fine. This is being conversion tested now.

#23 - 02/24/2013 06:16 AM - Constantin Asofiei

Committed to bzr revision 10194, passed runtime regression testing.

#24 - 04/12/2013 05:10 AM - Constantin Asofiei

About the runtime implementation of SELF handles: I don't want to do this in a separate task; instead, each case which can generate a new SELF handle should be implemented when that feature is implemented (and the implementation is as simple as adding push/pop calls using SelfManager APIs).

Transcript from the docs:

```
• You can reference the SELF handle only within a user-interface trigger or the
event procedure for an ActiveX control or asynchronous remote request.
• In user-interface triggers, SELF is not automatically the widget that has input
focus. To give input focus to the widget referenced by SELF, you must apply the
ENTRY event to SELF within the trigger block. Note that you must do this for fill-in
widgets whose AUTO-ZAP attribute you want to set, as in this fragment
• In the event procedure of an asynchronous remote request or in the context of a
procedure called directly or indirectly by this event procedure, SELF returns the
associated asynchronous request handle.
• In the event procedure of an ActiveX control, SELF returns the control-frame
handle and the COM-SELF system handle returns the control-frame
COM-HANDLE value.
• If referenced within a READ-RESPONSE event procedure, then SELF is the
socket handle associated with the connection that received the message. If
referenced within the CONNECT event procedure, then SELF is the server socket
handle.
```

We should add notes to each task associated with each above case and close #2015 and #2013 (note that UI triggers are implemented properly).

#25 - 09/27/2013 05:22 AM - Constantin Asofiei

NullWidget usage should be removed and SELF handle (for widgets) tested carefully - see notes 21/22 in #1612.

#26 - 10/20/2013 01:47 PM - Greg Shah

What is left to do on this task? I think your recent work with async events/sockets probably filled most of the gaps. We aren't going to do anything with Active-X controls in this task, so I wonder what is actually open here. Perhaps just the NullWidget stuff is open?

#27 - 10/20/2013 02:06 PM - Constantin Asofiei

> Perhaps just the NullWidget stuff is open?

Yes, only the NullWidget usage remains to be removed.

#28 - 01/20/2014 09:18 AM - Constantin Asofiei

- File ca_upd20140120a.zip added

This removes NullWidget and makes both SELF and FOCUS act as real handles. Manual testing looks OK, waiting for the automated testing to finish.

#29 - 01/20/2014 09:52 AM - Greg Shah

Code Review 0120a

I am fine with the changes. Commit and distribute them when they pass testing.

#30 - 01/22/2014 09:21 AM - Constantin Asofiei

- File ca_upd20140121d.zip added

Passed conversion and runtime testing, committed to bzr rev 10445.

#31 - 01/22/2014 09:50 AM - Greg Shah

- Status changed from WIP to Closed

#32 - 11/16/2016 11:42 AM - Greg Shah

- Target version changed from Milestone 7 to Runtime Support for Server Features

Files

ca_upd20130221c.zip	68.7 KB	02/21/2013	Constantin Asofiei
ca_upd20130222a.zip	69.6 KB	02/22/2013	Constantin Asofiei
ca_upd20130222d.zip	394 Bytes	02/22/2013	Constantin Asofiei
ca_upd20130222e.zip	88.3 KB	02/22/2013	Constantin Asofiei
ca_upd20130222h.zip	91 KB	02/22/2013	Constantin Asofiei
ca_upd20130222h.zip	90.9 KB	02/22/2013	Constantin Asofiei
ca_upd20130222j.zip	91.1 KB	02/22/2013	Greg Shah
ca_upd20130223h.zip	1.64 KB	02/23/2013	Constantin Asofiei

ca_upd20140120a.zip	109 KB	01/20/2014	Constantin Asofiei
ca_upd20140121d.zip	109 KB	01/22/2014	Constantin Asofiei