

Base Language - Feature #4373

finish core OO 4GL support

10/30/2019 09:49 AM - Greg Shah

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
billable:	No	vendor_id:	GCD
Description			
Related issues:			
Related to Base Language - Feature #4374: parameter validation for Progress.L...		New	
Related to Base Language - Feature #3751: implement support for OO 4GL and st...		Closed	
Related to Base Language - Feature #4349: add support for 4GL ENUM		Closed	
Related to Base Language - Feature #4347: add runtime support for STOP-AFTER ...		WIP	
Related to Base Language - Bug #4348: investigate and fix extent issues in me...		New	
Related to Base Language - Feature #4353: finish issues in progress.lang clas...		Rejected	
Related to Base Language - Feature #4352: finish progress.lang.apperror and p...		Rejected	
Related to Base Language - Feature #4350: method overload when they differ by...		WIP	
Related to Base Language - Feature #4351: instance vs static dual behavior fo...		New	
Related to Base Language - Feature #4354: add GET-CLASS() function support		Closed	
Related to Base Language - Bug #4355: exception member of the CATCH block can...		Closed	
Related to Base Language - Bug #4563: assigning a dynamic extent var to a OO ...		New	
Related to Base Language - Bug #4602: fixes for OO 4GL and structured error h...		New	
Related to Base Language - Feature #4629: implement fully compatible 4GL coll...		New	
Related to Base Language - Feature #4658: OO serialization support		WIP	
Related to Base Language - Bug #5011: unimplemented OO 4GL quirks and bugs		New	
Related to Base Language - Feature #5113: implement c'tor overloading for cas...		Rejected	
Related to Base Language - Feature #5146: record additional legacy class and ...		New	
Related to Base Language - Bug #5651: conditions raised from a finally block		WIP	
Related to Base Language - Bug #5691: deviation in error generation for inval...		New	
Related to Base Language - Feature #5729: implement polymorphic reflection su...		New	
Related to Base Language - Feature #5745: New access modes: package-private, ...		New	
Related to Base Language - Bug #5761: replace ErrorManager.displayError with ...		New	
Related to Base Language - Feature #6161: add support for DYNAMIC-PROPERTY() ...		New	
Related to Base Language - Feature #6292: nested catch blocks		New	
Related to Base Language - Feature #6300: class definitions with USE-WIDGET-POOL		Closed	
Related to Base Language - Feature #6410: implement additional built-in OO cl...		WIP	
Related to Base Language - Feature #6490: DYNAMIC-INVOKE which returns an extent		New	

History

#1 - 10/30/2019 09:53 AM - Greg Shah

This task is to aggregate the work to finish the OO 4GL implementation. The majority of the work for OO 4GL was already done in [#3751](#). This is just the list of the last remaining work.

- [#4349](#) enum support
- [#4347](#) block option STOP-AFTER (needs runtime)
- [#4348](#) extent method return for ObjectOps.invoke (and LegacyClass versions). We may have issues with extent return type (but I'm not sure).
- [#4353](#) Progress.Lang.Class:hasWidgetPool()
- [#4374](#) parameter validation for Progress.Lang.Class.invoke and new
- [#4353](#) progress.lang.ParameterList:setParameter - argument validation

- [#4352](#) apperror vs syserror - these weren't tested comprehensively, the support is pretty good, but when combined with ROUTINE-LEVEL or BLOCK-LEVEL statements, we may have missing issues (as I recall, we need to convert ERROR condition to SysError class).
- [#4350](#) method overload when they differ by a temp-table, dataset, buffer or object or extent or parameter modes - this is both conversion and runtime (to i.e. find the correct API in case of a dynamic call).
- [#5761](#) replace ErrorManager.displayError with recordOrThrowError and fix recordOrShowError to morph into a recordOrThrowError in certain cases
- [#5113](#) implement c'tor overloading for cases where type erasure is an issue
- [#4351](#) the dual resource behavior (static and instance); we have define stream support, but we need to add support for the other resources which behave like this (frames, query, etc)
- [#4354](#) GET-CLASS built-in function
- [#4355](#) catch block variable definitions
- [#4363](#) assigning a dynamic extent var to a OO method call as rvalue added
- [#4602](#) fixes for OO 4GL and structured error handling added
- [#4658](#) OO serialization support
- [#5146](#) record additional legacy class and method data in annotations
- [#5651](#) conditions in a finally block
- [#5691](#) deviation in error generation for invalid method/property chaining
- [#5729](#) implement polymorphic reflection support for properties and variables

Deferred Work (not to be completed now)

- [#4629](#) implement fully compatible 4GL collections (backed by protected temp-tables)
- [#5011](#) unimplemented OO 4GL quirks and bugs
- [#5745](#) New access modes: package-private, package-protected
- [#6161](#) add support for DYNAMIC-PROPERTY built-in function and DYNAMIC-PROPERTY language statement
- [#6292](#) nested catch blocks
- [#6300](#) class definitions with USE-WIDGET-POOL
- [#6410](#) more built-in classes/interfaces
- [#6490](#) DYNAMIC-INVOKE which returns an extent

#2 - 10/30/2019 09:54 AM - Greg Shah

- Related to Feature [#4374](#): parameter validation for Progress.Lang.Class:invoke and new added

#3 - 10/30/2019 09:55 AM - Greg Shah

- Related to Feature [#3751](#): implement support for OO 4GL and structured error handling added

#4 - 10/30/2019 09:55 AM - Greg Shah

- Related to Feature [#4349](#): add support for 4GL ENUM added

#5 - 10/30/2019 09:55 AM - Greg Shah

- Related to Feature [#4347](#): add runtime support for STOP-AFTER block option added

#6 - 10/30/2019 09:56 AM - Greg Shah

- Related to Bug #4348: investigate and fix extent issues in method/function return type added

#7 - 10/30/2019 09:56 AM - Greg Shah

- Related to Feature #4353: finish issues in progress.lang classes related to 4GL reflection added

#8 - 10/30/2019 09:56 AM - Greg Shah

- Related to Feature #4352: finish progress.lang.apperror and progress.lang.syserror added

#9 - 10/30/2019 09:56 AM - Greg Shah

- Related to Feature #4350: method overload when they differ by a temp-table, dataset, buffer or object or extent or parameter modes added

#10 - 10/30/2019 09:56 AM - Greg Shah

- Related to Feature #4351: instance vs static dual behavior for class-defined resources added

#11 - 10/30/2019 09:56 AM - Greg Shah

- Related to Feature #4354: add GET-CLASS() function support added

#12 - 10/30/2019 09:56 AM - Greg Shah

- Related to Bug #4355: exception member of the CATCH block can be referenced after the block catching it added

#13 - 03/03/2020 08:14 AM - Constantin Asofiei

- Related to Bug #4563: assigning a dynamic extent var to a OO method call as rvalue added

#14 - 03/30/2020 02:32 PM - Constantin Asofiei

- Related to Bug #4602: fixes for OO 4GL and structured error handling added

#15 - 04/27/2020 10:05 AM - Greg Shah

- Related to Feature #4629: implement fully compatible 4GL collections (backed by protected temp-tables) added

#16 - 05/29/2020 08:25 AM - Greg Shah

- Related to Feature #4658: OO serialization support added

#17 - 11/20/2020 09:19 AM - Greg Shah

- Related to Bug #5011: unimplemented OO 4GL quirks and bugs added

#18 - 01/27/2021 04:43 PM - Greg Shah

- Related to Feature #5113: implement c'tor overloading for cases where type erasure is an issue added

#19 - 02/12/2021 10:41 AM - Greg Shah

- Related to Feature #5146: record additional legacy class and method data in annotations added

#20 - 09/28/2021 04:07 PM - Greg Shah

- Related to Bug #5651: conditions raised from a finally block added

#21 - 09/29/2021 09:14 AM - Greg Shah

- Related to Bug #5691: deviation in error generation for invalid method/property chaining added

#22 - 10/14/2021 11:03 AM - Greg Shah

- Related to Feature #5729: implement polymorphic reflection support for properties and variables added

#23 - 10/21/2021 01:29 PM - Greg Shah

- Related to Feature #5745: New access modes: package-private, package-protected added

#24 - 10/21/2021 01:29 PM - Greg Shah

- Related to Bug #5761: replace `ErrorManager.displayError` with `recordOrThrowError` and fix `recordOrShowError` to morph into a `recordOrThrowError` in certain cases. added

#25 - 03/11/2022 12:50 PM - Greg Shah

- Related to Feature #6161: add support for `DYNAMIC-PROPERTY()` built-in function and `DYNAMIC-PROPERTY` language statement added

#26 - 04/25/2022 12:55 PM - Greg Shah

- Related to Feature #6292: nested catch blocks added

#27 - 04/26/2022 01:46 PM - Greg Shah

- Related to Feature #6300: class definitions with `USE-WIDGET-POOL` added

#28 - 05/25/2022 03:51 PM - Greg Shah

- Related to Feature #6410: implement additional built-in OO classes/interfaces added

#29 - 06/05/2022 02:57 PM - Greg Shah

- Related to Feature #6490: `DYNAMIC-INVOKE` which returns an extent added