



Gedae 5.4 Release Notes

January 2008

Address: Gedae, Inc.
1247 N Church St, STE 5
Moorestown, NJ 08057
Telephone: (856) 231-4458
FAX: (856) 231-1403
Internet: www.gedae.com

1 Retired Feature

1 Retired Feature

Encoded Queue to Parameter Connection

The ability to connect encoded queues to parameters has been removed from Gedae. This feature is superseded by the segmented parameter capability introduced in Gedae 5.3.2.

2 New Features

Trace Table Enhancements

Four new features have been added to enhance the user's ability to trace and debug an application's execution. These features are added as new sections to the Gedae Trace Table Users Manual. The new sections describe:

- Trace Event Filtering – reduce number of events collected and displayed
- Getting Trace Information from Non-Responsive Processors
- Unmapped Trace Memory – collect large trace buffers on the Cell Processor or other memory limited processors
- User Settable Data Trace Probes – collect data streams in circular trace buffers that can be dumped to host at the user's request.
- Saving trace table as file of comma delineated event descriptions.

The manual is available at:

<http://www.gedae.com/Manuals/Gedae%20Trace%20Table%20Users%20Manual.pdf>

Efficient Scheduling of Families

Scheduling of static thread execution order is now done more efficiently when there are many identical threads. The new scheduling capability identifies equivalent threads and runs the scheduling algorithm for only one of them. The other equivalent threads get a copy of the first thread's schedule. The goal of this work is to avoid doing redundant work.

It is unclear at this time if this new capability will save time or actually cost time as the algorithm to determine schedule equivalences is fairly involved. This new feature can be turned off using the `-no_sgeq` argument. That is starting Gedae up as

```
gedae ... -no_sgeq ...
```

will turn off the identification of equivalent schedules.

Improved Compilation Statistics

The Utilities->Print Timers menu option accessed from the top-level Gedae graph now prints a more complete profile of the different functions required to do graph compilation. Please send this information to gedae@gedae.com for graphs that take a long time to compile. This information will help us at Gedae Inc improve the performance of the compilation algorithms.

Code Generation of Static Schedule Threads

Static schedule threads are executed interpretively by the Gedae static schedule execution function. A new feature to bypass this interpretive execution has been developed where the static schedule threads are code generated. The code generation reduces the overhead between primitive function calls. Currently this code generation is restricted to subschedules where the performance improvement will be most noticeable.

The code generated static schedules are not used when the trace option is enabled. This is due to the fact that trace collection calls are not code-generated.

Functional Interface to Launch Package

A user can now convert a Gedae graph into a simple function call that can be linked with any external program. This capability allows users to replace functions in their current code with a call to a Gedae created launch package that runs on a high performance processor. Replacing functions with Gedae generated functions provides immediate acceleration of a user's application with minimal impact to their current code. A detailed description of this capability can be found in the: Function Interface to Launch Package Programmers Manual. The manual describes:

- How to create an application graph that implements a particular function
- The function generated for a graph
- How to code-generate the function from the graph
- Code-generation products
- Using the generated functions:
- Products of the code-generation
- Linking the final application

The manual is available at:

<http://www.gedae.com/Manuals/Gedae%20Functional%20Interface%20to%20Launch%20Package%20Programmers%20Manual.pdf>

BSP Enhancements

Dumping Memory from Non-Responsive Processors

The Gedae board support package has been enhanced with features that support the both the user assisted and automatic dumping of trace information from non-responsive partitions. These enhancements have been documented in Appendix A of the Gedae Board Support Package Development Kit

Double Conversion Routines

Routines have been added to support conversion of doubles from the host to the target format and from the target to the host format. These routines are described in the Gedae Board Support Package Development Kit.

Inplace Pointer Streams

An inplace pointer output stream can be used to save both execution time and memory by directly setting the output to a runtime selectable pointer into one or more input streams. The inplace pointer allows zero copy operations like selecting a subvector out of a vector, multiplexing multiple matrices to a single output and dynamically selecting which output of a branch to send data. Examples of how to create and use inplace pointer streams are described in the section describing the forward function in the Gedae Primitive Function Reference Manual available at:

<http://www.gedae.com/Manuals/Gedae%20Primitive%20Function%20Reference%20Manual.pdf>

Flow Graph Editor Enhancements

The Parameter Table (View->Parameter Table...) and the Restore and Manage Dialog (File->Restore and Manage...) have been updated as described in the partial version of the Flow Graph Editor Reference Manual available at

<http://www.gedae.com/Manuals/Gedae Flow Graph Editor Reference Manual.pdf>

The Parameter Table enhancements have been added to support hierarchical parameter settings. Parameter files for subgraphs can be loaded from the Parameter Table, and the parameter file hierarchy is saved and restored. In addition, the “default” parameter file is now automatically loaded during flow graph instantiation.

The document also describes changes to the Add Data dialog to support the creation of double precision real and complex values.

3 Bugs Fixed

SCR	Title	Problem Description
002349	FGU does not work with long pathnames	When moving a FGU tar from Solaris to Windows, the Windows system cannot reproduce long filenames. This seems to be a problem with the Windows tar executable.
002694	FGED archive restore goes to wrong directory on Windows	The restore and manage dialog was integrated into the Flow Graph Editor in version 5.4, replicating the capability of the Flow Graph Utility. However, calling restore from the Editor restored the files under <user>/<os>/FGlibraries, which provides proper restoration on Linux and Solaris but not Windows. This bug has been resolved in 5.5.
002701	Pressing delete key in fast entry area causes FGED items to also be deleted	Pressing the delete key in the fast entry area causes the selected items in the flow graph editor to be deleted. If the focus is on the fast entry area, the delete key should only apply to the text in the fast entry area, not to the flow graph editor.
002702	Box with natural granularity of zero not detected	Gedae does not detect primitives that have a natural granularity of zero. This failure to detect a zero granularity caused an arithmetic error during the allocation phase of graph compilation. The problem is fixed. Gedae now displays a message dialog showing the first box found with a natural granularity of zero. Also all of the boxes with natural granularity of zero are printed to the terminal.
002703	Undefined target symbols when different box classes have same simple box name	
002707	Successive fast entry searches cause search dialog to improperly expand/collapse	If first search from the fast entry area brings up the search dialog, or the dialog is otherwise collapse to hide the data entry area, then successive fast entry area searches may cause the dialog to get into an unusable state. The dialog may show the entry area but not show the top of the list of matches, hiding the matches at the top of the list as well as the Expand/Collapse button. The dialog must be canceled or destroyed in order to return to proper functioning.
002709	Memalign no longer supported by Cell SDK	The Cell SDK 3.0 General Release introduced the removal of support for memalign. Gedae's memory allocation has changed to SDK supported functions.
002719	Autodocumentation does not allow Windows to render before screenshot	The autodocumentation feature creates an HTML document to describe a flow graph by incorporating screenshots of Gedae windows. The utility for taking a screenshot would sometimes not allow the window to render before taking the screenshot, creating images with blank spots. This problem was noticed on Windows but may have affected other builds as well. The autodocumentation utility now waits for the Window to render before taking each screenshot.
002721	Persistent memory in a subschedule memory is erroneously shared	Memory that is persistent (as a result of using delay or overlap statements in a function box) must not be shared. However, if the persistent memory is in a subschedule, then the memory is erroneously shared with other subschedules in the same parent schedule. This problem has been fixed.
002726	Double endian conversion on Cell	Using double parameters on the Cell with an x86 development host did not use an endian conversion when transferring the data to the SPEs from the development host.

002733	Improve speed of graph archiving	The archiving speed for large graphs was slow because files were added to the archive incrementally. The archive creation has been changed to do many files at once.
002736	Gsim unresponsive to X Windows events	Gsim is unresponsive to X Windows events. This is caused because the monitor thread is invoked only after a certain amount of time has passed. But the time is being measured in terms of Gsim time - not real time. To solve the Gsim main loop was modified to call a polling function on a real time periodic basis. The polling function that is registered handles X Windows events.
002741	The primitive vx_halfK is incorrect	The primitive embeddable/vector/complex/vx_halfK produces incorrect results due to an improperly formed call to an E library function.
002763	Invalid connections during successive add-boxes can lead to a segfault	If two v_constant boxes are added one after the other, Gedae cannot automatically make connections between them. To compensate, Gedae removes the 2nd one, deselects the data output, and adds the 2nd one again. This behavior led to a segfault because some code still referred to the removed instance of the box and not the re-inserted one.
002765	Exclusive segmented family driving non-merge box input doesn't work	The following situation does not work: Graph A is driven by a segmented output Graph A contains an exclusive branch that drives a family of subgraphs B One of the subgraph B outputs drives a box C with a static input. In the above case the box C is not given the segment boundaries produced by A.
002766	embStrdup not defined in Solaris/Mercury launch packages	
002771	Control->Detach can get called on unattached application	For some BSPs Gedae can attach to an already running application using the -attach command line argument. Applications that are attached to in this way can be detached and allowed to continue running by selecting the Control->Detach menu. However, if Control->Detach is called on an application that is not attached Gedae will segfault. This problem has been fixed by sensitizing Control->Detach only when the application is attached. In addition Gedae will now exit immediately if an attached application is detached using Control->Detach or if the application is terminated using Control->Terminate. In either case the user is notified that Gedae will exit and is asked if Gedae should continue the operation. An additional change was made that an attached application that is stopped using Control->Stop can be continued with Control->Continue but a new run cannot be started using Control->Run.
002776	Family pointer stream output connected to family forwarding pointer stream input segfaults	
002777	Group settings are easily lost after graph has been edited.	If a box is deleted or replaced at any level of a graph then old group settings files may not restore correctly.
002778	Gsim segfaults on graph termination during closing of ports	

002780	Development environment timeout during subschedule causes incorrect dataflow behavior	
--------	---------------------------------------------------------------------------------------------------	--

4 Known Bugs

4 Known Bugs

SCR	Title	Problem Description
002012	Running two VxWorks processes on the same processor.	This problem occurs when trying to run two separate Gedae generated VxWorks executables on the same processor; however, the entry point for each executable has the same name, VxWorks_main, making this impossible.
002015	Multiple exclusive sources with some of the sources not used by every mode.	If there are multiple exclusive sources to a family of modes and some of the sources are not used by every mode, then Gedae crashes during development time scheduling. For example, if two exclusive branches drive three downstream modes and one of the branches has one of its outputs unused by the third mode, then this causes a segfault during scheduling. The workaround is to add dummy inputs to the modes to allow all the sources to be used by every mode.
002019	Graphs with host to target control ports fail on Linux and Solaris	The problem is that the host is not performing mailbox services while it is waiting to establish a control port to the target processors. Unix processors require the host to perform these services in order to make connections.
002020	Check in all licenses on exit.	Gedae relies on FLEXlm to find checked out licenses that are no longer in use and check them back in. This works on most systems, but a BSP user reports that it does not work on their system. We should explicitly check in all licenses on Gedae exit.
002022	Inplace box scheduling problems.	Copy boxes occasionally need to be added to a graph by the user to avoid scheduling problems associated with primitive outputs marked as being inplace with an input.
002024	Connecting Graph Parameter to User Define Type Segfaults.	Gedae erroneously allows standard C parameter types to be connected to user defined parameter types. This type of connection causes Gedae to segfault.
002047	Allow Changing Length of Parameter String at Runtime.	Changing the value of a string parameter to a stream may cause Gedae to stop executing. Gedae stops executing if the length of the string is changed.
002054	Modifying a running graph segfaults.	Modifying a Gedae graph that is running can cause a segfault. This problem has been reported several times but has not been duplicated by the Gedae support group. Most edits are disabled during graph execution.
002062	Outlaw segmented static schedules controlled by nondet inputs with multiple boxes.	Segmented static schedules controlled exclusively by nondet inputs and that contain more than one box should be outlawed. These graphs are currently considered problematic and can produce unexpected results.
002064	Outlaw pointer streams followed by delays or overlap	Pointer streams followed by delay boxes or boxes with input overlap parameters do not work and should be outlawed.
002078	Gedae can go to sleep if processes are polling.	A Gedae process can go to sleep if a process is polling. The sleeping is only seen on the NT BSP, which is currently the only BSP that implements the sleep

		capability. Gedae should only sleep when all schedules are in the paused state.
002104	Handling of null segments in distributed graphs	If a segmenter controls a segmented subgraph that is distributed and if the segmenter produces null segments, then parts of the distributed graph that are not directly controlled by queues will not see the end-of-segment.
002158	Trace Table send/recv webs do not work from attached launch package	When the Gedae Development Environment attaches to a launch package, the Trace Table send/recv webs do not work. No send/recv webs are displayed.
002221	Search on type does not follow route boxes.	If you search on the type of the input to a scope1, it does not recognize the type because there is a route box between the input and the next primitive.
002244	Using f6 to disconnect a constant can cause a segfault.	Disconnecting a constant source from a constant destination using the f6 button causes a segfault. This occurs because a stub is now connected to the constant destination and the evaluation of the constant fails.
002245	Gedae does not prompt the user to save a graph.	If a graph has been modified using the f6 cursor and that was the only modification, then Gedae does not prompt the user to save the graph when exiting Gedae.
002261	embTerminateError called from Reset does not stop the primitive.	If embTerminateNormal or embTerminateError are called from a Reset method of a graph that has more than one static schedule, then the static schedule containing the primitive is not terminated, as it should be. The schedule Apply methods will be called even though one of the schedule s Reset failed.
002267	Nonfamily output connected to family input causes graph to crash.	If a nonfamily stream source is connected to a family stream input, then Gedae segfaults when the user runs the graph. This type of graph should be detected as an error, and Gedae should not be allowed to run.
002282	Limitations on the use of typedef primitives	Typedef primitives will not work if connected to variable vectors or matrices. Typedef primitives will not work if connected to route boxes.
002312	The Gedae dy4av2 BSP function e_zvrcip can kill process	Calling the Gedae Dy4av2 BSP function e_zvrcip can cause the process to exit with an arithmetic exception. The problem is that even if the argument to e_zvrcip has only its real or imaginary part set to zero (but not both), then a divide by zero error still occurs.
002360	Trigger boxes with Reset method but no input parameters don't get fired	A trigger box that has a Reset method but no other inputs does not get fired when the user selects Control->Run.
002361	Eval boxes with no inputs are not included in the launch package	This feature can be confusing as when the Eval box calls getcwd. In this case the launch package will use the wd from which it was created - not the one it is run from.
002378	Create Subgraph does not support multidimensional families	Create Subgraph cannot handle boxes and data that use multidimensional families.
002399	Gedae built-in function names should be ignored if used in nonfunction context.	Gedae built-in functions like "time" should be ignored if not used as a function. That is time(x) should be converted by the Gedae parser but x.time (or even x.time(x)) should not.
002400	Family of array parameters to a trigger box does not codegen	An input parameter like float [N]in[Max] does not codegen correctly for trigger boxes. The data values are not correct. If only one family member is set, then the box still

	correctly.	executes; however, the array dimensions are invalid.
002406	Primitive with EndOfSegment but no Apply does not work.	If a primitive has an EndOfSegment method but no Apply method, the primitive is not included in the running application and the EndOfSegment method is never called.
002419	FGTable edit box does not recognize termination	Gedae does not accept some FGTable entry while the graph is running. If you are trying to edit an FGTable entry, and then terminate the graph so that you can change the value, the FGTables do not automatically recognize the graph has terminated. The user must unselect the entry area then reselect it to enter the data.
002420	Microphones sporadically stop collecting data on AFG for Linux	AFG would sometimes stop getting data from the microphones. Users were successfully running the graph for several tries, but at the next execution, it would stop working. AFG recognizes the microphone but just does not produce any data.
002423	Dy4Av2 Installation Requirement	We needed to remove the files rsh.exe and cygwin1.dll from %gedae%\nt\bin because delivering these files and placing %gedae%\nt\bin in the users PATH variable caused many users to have conflicting versions of cygwin1.dll in their path. It is now necessary to copy cygwin1.dll and rsh.exe from the cygwin bin directory into %gedae%\nt\bin. For example, these files can be copied as: copy c:\cygwin\bin\cygwin1.dll %gedae%\nt\bin copy c:\cygwin\bin\rsh.exe %gedae%\nt\bin
002431	Local dcomplex variable is treated like a complex variable.	If you create a dcomplex local Local: { dcomplex temp[N]; } and then look at the Memory Map, the Memory Info dialog states Type: complex ... Elem Size: 8 ... The correct Elem Size is 16. Running with a local dcomplex causes a segfault because this buffer is not big enough.
002432	Connection from float on canvas to double input parameter does not cast correctly.	If there is a double parameter input to a primitive and a float on the canvas, the user can connect the float to the double, however the cast is not performed correctly. The value inside the primitive is 0. Either cast floats to doubles correctly or disallow the connection.
002447	Entering Launch Package Directory in launch info dialog does not cause recompile.	Entering a directory name in the launch package creation dialog does not cause a group that is in the compiled state to move down to the allocated state. As a result, hitting the Make button on the launch package dialog has no effect.
002450	Loading group settings when the target host is set can fail.	If group settings are already loaded in which the Target Host is set to something other than default and if a new group setting is loaded in which a partition is mapped to a target logical number that appears in the Target Host configuration file but not the default configuration file then Gedae complains that "Logical processor 100 is not in embedded configuration file" and fails to set the "Run on Target" toggle on.
002451	Undefined symbols of form I0023_recv in target executables.	The error message: "Warning: lookupDirectorySerialNumber: directory name %s not in database and database is locked\n" can occur when compiling a distributed graph mapped to a target processor. This error may cause the undefined symbols of the form I0023_recv when linking the target executables.
002467	Symbol "hpererror" multiply defined in Exceed and	Including both <X11/X.h> and <stdlib.h> introduces a compiler warning for some mixtures of versions of Exceed

	Visual C++	XDK and Visual C++. The problem was noticed when building with .NET 2003, and an evaluation version the eval version of XDK 2007. This problem is not noticed with the same Visual C++ and the full release version of XDK 2007. A workaround is to add a line like #define hperror hperror2 in front of the include for the 2nd of these 2 files.
002472	The build in size function does not work for dynamic inputs	If a primitive input is declared as dynamic stream in; then the build in function size does not work correctly. The value of size(in) will be zero instead of the number of data elements in the queue. The primitive should use granularity instead. If the declaration is dynamic stream in[N][M](D), then the Apply method should replace size(in) with granularity*N*M*D. This problem should either be detected or fixed.
002473	Gsim exits if group is set to run on target model and there is no target model.	Should not allow graph to run in such a situation.
002478	Memory Map Dialog "View->Changes" does not work for target command programs.	In the situation where the development environment is talking to a target processor through a target command program, selecting the View->Changes option on the Memory Map Dialog for the target causes Gedae to segfault.
002479	If a compile is not necessary do not say complete and successful - say up to date.	If a compile is not necessary do not say complete and successful - say up to date.
002498	Partition not set correctly when adding new box to partitioned subgraph	When adding a box to a subgraph that has its partition set in the Partition Table, the new box should have its partition set to the subgraph partition. This does not happen. A workaround is to save the group settings (if needed) before adding the box and then reload the group settings after the box is added.
002499	Changing connection does not mark graph as dirty	Linux PPC
002502	Multiple dyndeqs are added when fanning out from dynamic output to multiple static dests.	
002503	Variable vector dyndeq's are inefficient	
002507	Hitting return in blank Launch Package Dialog directory field.	If a return is hit in the Launch Package Dialog's directory field and the field is blank it does not "unset" the directory value as it should, but rather sets the directory value to an empty string.
002521	Don't allow setting Firing Table Granularity if Gran Mult > 1	When using the Firing Table, if the Gran Mult field on a schedule is > 1, then the user should not be allowed to set the Granularity field of any primitive in the schedule. Additionally, setting the Gran Mult should clear all the schedule Granularities.
002549	Memory leak when increasing the queue size of a shared queue.	When increasing the queue size of a shared queue. Gedae does not free the original memory. This results in a memory leak. If the queue size is increased a second time,

		it would be possible to free the memory allocated the first time. Gedae does not keep track of this.
002550	FGU of hierarchical typedef boxes	FGU does not transfer hierarchical typedef boxes correctly. The typedef used to define the input of the box is set to the old directory rather than the new.
002551	Graph Stalls	A rare condition can cause a graph to stall (or segfault) when the controlled static schedule is partitioned to two processors in the following form: A->B->A. The problem scenario is that the schedule is partitioned into three parts, with the first and last parts mapped to the same processor. Usually Gedae puts the parts mapped to processor A in the same static schedule; however, to allow efficient pipelining, Gedae splits the two parts mapped to processor A into two different static schedules. They are numbered n.1 and n.2 (for example 2.1 and 2.2). To see if any schedules have been broken into two parts, the user can pop up the Schedule Info Dialog and see if any of the schedule names contain a decimal point. The decimal point in the schedule name does not necessarily indicate a problem. The problem only occurs when the data source driving the processing is faster than the graph, causing the control message queue to back up and overflow. The condition is rare because the problem only happens when the graph is not keeping up with the input data rates.
002553	Embedded build can require a makeGEDAE CLEAN	If an application is repartitioned, then the target executables do not get relinked. The problem is that all the .o files are older than the targets, and the fact that there is a new link line does not force the target library and target executables to rebuild.
002554	Unterminated comments	Unterminated comments cause the Gedae parser to segfault.
002555	Parser problem	The Gedae parser does not handle an odd number of quotes well.
002556	Arrays of strings not allowed	Gedae currently allows string array graph parameters to be declared as: const string X[] = { hello , world } or string X[i] = [i]Y where Y is a family of strings. In either case, the values so declared are not correctly set, and therefore, should be considered illegal.
002557	Function appFree memory leak	A command program running on VxWorks does not free all the resources allocated (memory, sockets, etc). The appFree function must release everything allocated. Gedae should automatically generate a call to appFree for the standard exec-host command program.
002558	External Code does not recompile	Make is not called after a successful run, so changes to code listed in the Personal_Emb_Obj_List do not get recompiled. To force the recompile, it is currently necessary to change something from the Gedae GUI. For example, saving a primitive or toggling the Group "Run on Embedded" toggle off and on will force a recompile.
002559	Large Graphs fail to display on flattened graph.	If a graph is too large, then it cannot be displayed on the flattened graph. This occurs when the flattened width or height exceeds the allowable pixmap width or height of 32768.
002560	Primitive cannot	If a primitive Input, Output or Local section is modified at

	recompile	runtime, then Gedae segfaults when the primitive is recompiled, and the graph is rerun. Currently, the user must exit Gedae after a primitive Input, Output or Local section has been modified.
002561	DSA with fan-out does not work for some BSPs.	If a box output fans-out to several boxes mapped to several different processors, then the DSA communication mechanism does not work correctly for Mercury and Sky BSPs.
002562	FFT primitives only work with power of 2 sized vectors.	The FFT boxes do not support non-power-of-2 lengths; however, the comments make no mention of this fact. If these boxes only support a power-of-2, then it would be useful to have a separate set of boxes that support a non-power-of-2.
002563	Constants propagated through typedef boxes	Constants propagated through typedef boxes cannot be used for instantiation.
002564	Stream box with push in hostless launch package	If a stream box contains a call to push and it is made part of a hostless launch package, then the launch package will fail to compile, as the code for the push is not included in the standalone library.
002565	Transferring doubles between host and target with different endian (Dy4av2)	Currently, the Gedae BSP does not support providing byte swapping of doubles when transferred between the host and target processors.
002583	Math.h symbol floor undefined in Windows build with VC6.0	When using the function e_vfloor on a Windows system linking with the Visual C++ 6.0 libraries, it has been reported that the math.h symbol floor is undefined. Newer versions of Visual C++ do not report this error.
002589	Saving a primitive to a directory that does not exist	When saving a primitive to a directory that does not exist, Gedae should create the directory.
002603	Box names are not updated correctly in an FGTable	The box names in FGtables sometimes do not match up with the ones on the canvas. If a subgraph is saved under a new name using Save As, then the tables such as Hierarchy and Fire Table do not update. Closing and reopening also does not update the name - the name stays stale, set to the old name. Exiting and reentering Gedae is the only way to see the new name.
002606	Error occurs when partition mapped to host and launch package is set to run with no host	If a partition is mapped to the host and the launch package is set to run with the option "Create schedules in Target Executables (No Host)", then Gedae does not flag this as an error until the user tries to build the application. The error message states: "Cannot create a target command program for the host BSP because the BSP does not have the createPortName function set". Instead Gedae should pop up an error message stating that the launch package is set to the no-host option but indicate that some partition is mapped to the host. In addition, it should pop up an error message when the launch package is already set to no-host and when you try to map a partition to the host.
002612	Linux BSP does not set nbsize of socket transfers correctly	Setting the nbsize of Linux (eredhat) stream transfer methods over 250000 does not work. For example, if the nbsize is set to 500000 and the sender tries to send 500000 bytes, then the sender will block until the receiver executes. Because the Gedae static scheduler counts on

		the sender being able to send the nbsize amount without blocking, the failure to send the nbsize amount of data can cause the application to deadlock.
002623	Dynamic data flow params: implement iterate constraint	The new iterate data flow parameter will not work if it is being controlled by a segmented parameter.
002696	Problem with subscheduling.	If a connected subschedule has a data path that exits and then reenters the subschedule, then the subschedule cannot fire. This situation should be detected and reported to the user. Currently the problem is reported as an error in setting dynamic dependencies - even if there are not dynamic queues in the graph.
002699	Trace data is not portable from LinuxPPC to Windows	Trace events saved on a PS3 (LinuxPPC) cannot be viewed on a Windows installation.
002710	FGTable entries that override equations don't always apply	If a family is first partitioned by equation, and then one member of the family is partitioned to a fixed value, then the fixed partition value does not show up in the Mapping Table. Similar behavior occurs for other FGTables as well.
002722	Subschedule feeding overlap input does not work	
002723	Dynamic dependency algorithm does not work with subschedules	
002737	Overlapping exclusion sets do not work	Suppose two exclusive outputs each have a family member that controls different sets of subgraphs. If these sets overlap but one set does not contain the other, then the exclusive set creation algorithm will not work correctly.
002742	Common recv port inplace with common send port does not work	
002743	Pointer forwarding does not work correctly with DSA or common	
002768	Replacing a second box in flowgraph while attached causes segfault	A segfault was observed when - a graph was executed - the graph was terminated - one box in the graph was replaced - a second box was replaced
002769	Exclusive subgraphs that output results to static input primitive do not work.	An exclusive branch that goes to a set of exclusive subgraphs typically ends in a merge box controlled by the same control input that controlled the exclusive branch. However, if one of the exclusive subgraph outputs goes to a static box input, then the static box input does not correctly propagate higher level segment information.
002782	Adding probe point to unmapped memory stream does not work	
002783	Adding probe point to primitive without Apply method does not work	
002785	Improve reliability and testing of inplace iteration	Inplace iteration sometimes doesn't work. Sometimes subscheduling has an affect on whether it works or not, and/or the results of it not working.