



Gedae 5.5 Release Notes

June 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 Deprecated Features

1 Deprecated Features

Start Method State Variables

Prior to Gedae 5.5 a stream primitives Start method could examine the value's of Local state variables. For example a Start method might check if a pointer was previously allocated and free it as:

```
Input: {  
    int N;  
}
```

```
Local: {  
    float *ptr;  
}
```

```
Start: {  
    if (ptr) {  
        free(ptr);  
    }  
    ptr = calloc(N,sizeof(float);  
}
```

This behavior is changed. The value of ptr may be set by the Start method but the prior value may not be examined. Instead the pointer should be freed in the primitives Terminate method. For example

```
Terminate: {  
    if (ptr) {  
        free(ptr);  
    }  
}
```

2 Features

As of Gedae 5.5 the Features section is organized by the manual in which the feature is described. New features will always be included in one of the online manuals.

[Gedae Primitive Function Reference Manual](#)

embGlobalStop

The embGlobalStop function works like the embBreak function except that instead of stopping the partition its called from it stops all partitions.

part

The part function allows a primitive to partition its output into nearly equal sized pieces that completely cover the output. The function is an improvement over dividing a size parameter (such as the row dimension of a matrix) as R/N which only covers the output if R is divisible by N .

[Gedae Debugging User's Manual](#)

The Gedae Debugging User's Manual is a new document. The document currently only describes new features. The document will be expanded in the future to describe existing debugging features in Gedae. The following new features are described.

Probes

A probe is a circular buffer that can be assigned to record the data in any stream prior to graph compilation. Probes are described including how specify probes, dump data from probes and probes to previously collected probes.

Break Points

The break point feature has been enhanced since Gedae 5.4. Documentation of breakpoints in general including the enhanced features is now provided including a description of how to use break points and probes to create automated test scripts to test graph execution.

[Gedae Compiler User's Manual.](#)

The Gedae Compiler User's Manual is a new document. The document contains sections for each of the dialogs that allow the user to control how the compiler implements a Gedae graph. Currently most of these sections are marked TBD.

Multibuffering

The new Multibuffering feature for implanting overlapped processing and I/O when moving data to and from unmapped memory is described. The documentation describes how to request multibuffered transfers and describes how these transfers are implemented.

Primitive Programmers Guide

The Primitive Programmer's Guide has been enhanced to describe a number of new features:

Tiled Matrices

The tiled matrix token type allows a primitive to specify that a matrix (or any N dimensions object) is embedded in a larger matrix. Tiled streams are typically used in conjunction with pointers to provide a zero copy specification of partitioning and concatenation of matrices into subtiles to be processed.

Iterate

A new static data flow parameter – iterate – is provided that allows a primitive to fire a fixed number of times on the same token. The new feature, while generally useful, was added to make it possible to subschedule the processing of matrix tiles.

Named Parameter Notation

The new iterate parameter is now the fourth static data flow parameter provided by Gedae. A new position independent named parameter list for specifying data flow parameters is described. The old position dependent parameter list is still available and is also described.

Pointer Streams

Pointer streams are an old feature in Gedae but are newly described in this document.

Optional Inplace Streams

Optional inplace streams allow a primitive to specify that an output may be inplace with one of several inputs but need not be. Optional inplace streams give the Gedae graph compiler more flexibility in creating an efficient implementation of a graph.

Inplace Output Pointer Streams

An inplace output pointer stream provides a zero copy method of setting an output token to a pointer within an input token. This feature is generally useful but is often coupled with tiled streams

Inplace Input Pointer Streams

An inplace input pointer stream provides a zero copy method of setting a primitive input token to a pointer within an output token. This feature is again often coupled with tiled streams.

Inplace Tiled Pointer Streams

Inplace Tiled Pointer Streams combine a number of features. Examples of their use are given.

Setting of Tiled Stream Dimensions

The rules for how tiled stream tiled and normal dimensions are set based on how they are connected in the larger graph are described.

Unmapped Memory

Memory that a processor can directly dereference is called **mapped** memory. For example if `float *out` is a mapped memory pointer then the expression `*out` is valid and will return the value of `out`. Memory that can only be accessed by copying it using a functional interface is called **unmapped** memory.

Some processors provide large areas of unmapped memory. For example the Cell/B.E. allows each SPU to access only 256kbytes of mapped memory but allows to many gigabytes of unmapped memory. In Gedae some BSPs declare different unmapped memory types and Gedae provides generic BSP functions for moving data between unmapped and mapped memory.

Streams can be marked unmapped using the `unmapped` keyword. This keyword is described along with example primitives.

Persistent Memory

Declaring an output stream persistent reserves the memory so that the data written to it in one execution of a schedule remains valid for successive executions. Only the primitive that declares the memory area persistent and any destination primitives (including inplace destinations that modify the memory area) can have access to it. Streams can be marked persistent using the persistent keyword. This keyword is described.

Data Analysis and Display Reference Manual

A set of new displays has been added to Gedae for improved data analysis capabilities. The following test graphs are available to illustrate the new displays:

gui/TestConstPlot	gui/TestMatrixPlot
gui/TestScalarSpectralPlot	gui/TestVectorIntBarPlot
gui/TestEnergyPlot	gui/TestScalarComplexLinePlot
gui/TestScalarXYPlot	gui/TestVectorIntPlot
gui/TestHistoPlot	gui/TestScalarIntPlot
gui/TestVectorBarPlot	gui/TestVectorIntXYPlot
gui/TestMatrixImagePlot	gui/TestScalarIntSpectralPlot
gui/TestVectorComplexLinePlot	gui/TestVectorPlot
gui/TestMatrixIntImagePlot	gui/TestScalarIntXYPlot
gui/TestVectorConstPlot	gui/TestVectorXYPlot
gui/TestMatrixIntPlot	gui/TestScalarPlot
gui/TestVectorHistoPlot	

Installation Guide for the Cell

The Cell install process has been updated to include new test graphs that utilize the tiling language features introduced in this release. These test graphs are available in the FGLibraries/boxes/examples/tiles directory and are further explained in the Cell installation guide.

Installation Guide for Intel SSE3

A new version of the Windows and Linux x86 Intel BSPs have been added that use the Intel software development tools to create an implementation optimized for Intel processors that use have SSE3 capability (Streaming SIMD Extensions), such as the Core 2 Duo. This BSP requires the Intel compiler and Math Kernel Library (MKL).

3 Bugs Fixed

SCR	Title	Problem Description
002830	Filtering out of duplicate probes does not work	A probe file can specify the same probe in multiple ways. The algorithm for filtering out duplicate probes does not work correctly and can put the same probe into the graph multiple times.
002845	Subschedule beginning with a non-pointer inplace iterator input	If a subschedule begins with a non-pointer inplace iterator input then the iteration value is not set correctly and as a result the streams that are inplace with the input have their pointers advanced faster than they should. The pointers into parent memory should stay fixed for every iteration firings of the primitives but instead advance every firing.
002849	Renaming during restore fails when boxes did not exist before	When files are renamed during an archive restoration, it depends on previous versions of the file existing in the library in order to alter the data structures and export the new code.
002850	Pointer Streams: Avoid adding unnecessary release primitives	
002852	FG Editor incorrectly warns about not saving group settings files for subgraphs	When exiting a graph where multiple group settings have been made, the user sometimes is told Status: 0 of xx applications files saved But is not given the opportunity to save all "xx" group settings files. The hierarchy that was added to parameter files was incorrectly being used in group settings files as well. The user should not have to save subgraph group settings.
002854	Archive creation fails in old Windows's with smaller line limits	The archive creation utility does not work on Windows 2000 (and older) because the command line length limit of Windows (2047) is less than the limit built into Gedae (8191). To fix this, Gedae will simply use the old Windows command line length limit.
002855	Pointer Streams: Release is incorrectly scheduled in subschedule	The release box associated with a set_ptr can be scheduled inside a subschedule if there is an inplace pointer chain into the subschedule.
002881	Graph stalls when sending double from target to host	When doubles are sent to the host the recv function always returns 0 even if data was sent from the target.
002883	Segmented Parameters: Unconstrained segparam connected to non-dataflow param fails during preprocessing	Unconstrained segparam connected to non-dataflow param fails during preprocessing. Problem is detected when evaluating static data flow expressions. Only dataflow parameters should require a constraint.
002885	internal_sd_forward, internal_ds_forward, and internal_dd_forward are undefined on Windows	The forward function cannot be used with dynamic streams due to undefined symbols from the linker when running on the Windows host.
002886	vm_flip_ud and vmx_flip_ud don't set the output matrix sizes correctly	

002889	Gen.exe segfaults during some flow graph diffs (or archive management)	During a flow graph diff (which also happens during archive management), gen.exe is called multiple times to perform the difference. Some comparisons resulted in segfaults, which may have prevented finding some differences.
002890	Creating a new primitive using add Box when typedef output selected segfaults	When a user attempts to add a box that does not exist the user is prompted to either create a primitive, a flowgraph or cancel. If the user selects to create a primitive and at the same time a typedef flowgraph output is selected gedae segfaults.
002895	Changing Mapping Table entry does not note that the group settings file needs to be saved	
002896	Once trace is turned on it can't be turned off on the target processors.	If the user enables trace and then disables it while a target application is running the target application continues to collect trace information - slowing down the processor.
002911	Parameters saved by Save methods do not follow hierarchy	When saving hierarchical parameters, boxes with Save methods are saved in each level of the hierarchy, polluting parameter files with redundant settings.
002920	Trace Table Time sync doesn't work on powermp	A problem exists, observed on the powermp BSP, where the type synchronization done during trace table collection is incorrect and inconsistent from run to run.
002921	Processor parameter argument list keywords cannot contain as a prefixed another keywords	If a processor parameter keyword contains as a prefix another keyword then the simpler parameter will return the value of the more complicated parameter. For example on the Cell Processor BSP the processor both -lph0 and -lph are processor keywords. The problem is that if the user sets the processor parameter to "-lph0 1000000" this behaves exactly as though the user set the processor parameter to "-lph0 1000000 -lph 1000000". This error has been fixed. As a consequence however group settings that have worked previously may no longer work if the processor parameter contains extra characters at the end that are not part of the actual keyword.
002925	Boxes with input pointers cannot have non-pointer inputs	Boxes with both pointer inputs and non pointer inputs do not schedule correctly. After scheduling the pointer box is moved before the boxes whose outputs it sets. But any boxes driving non-pointer inputs are not also moved. Such primitives are now declared to be illegal and are reported as an error when gedae attempts to read them.

002930	E Library Consistency, Length 0, and other corrections	<p>The following fixes were made to the E Library</p> <ul style="list-style-type: none"> e_cvrampr - always writes at least one complex output value even when the length parameter "n" is zero e_ddotpr - gives a non-zero result when length parameter "n" is zero e_dotpr - gives a non-zero result when length parameter "n" is zero e_dvabs - calls the integer function "int abs(int)" when it should call "double fabs(double)" e_dvmaxmg - uses post-increment as a parameter to the MAX macro, resulting in a double increment. e_dvminmg - uses post-increment as a parameter to the MIN macro, resulting in a double increment. e_dvsadd - header file defines e_dvaddK instead of e_dvsadd e_imeasqv - performs the square of an integer then accumulates the squares in a float. For large input values, the square may overflow causing incorrect results. It is probably better to convert inputs values to float before squaring to prevent overflow. e_ivlog2 - enters an infinite loop if the input vector includes one or more negative values. e_imaxmgv, e_iminmgv, e_minmgv - to be consistent with e_maxmgv, these functions now return an index of -1 if the length is 0. e_vpolrect, e_dvpolrect - When the x output is in-place, calculation of x overwrites an input, causing incorrect values for y e_vrectpol, e_dvrectpol - When the r output is in-place, calculation of r overwrites an input, causing incorrect values for theta
002942	Killing a graph doesn't clear its application files	
002946	Firing table priorities should override prioritization due to down stream sink box	<p>Primitive priorities in the firing table don't always make boxes with a higher priority fire first. This occurs if a lower priority box is upstream from a sink box (a box with no outputs or that has a higher priority than all its outputs). While boxes upstream from sink boxes should be fired before those that don't this should not take precedence over user set priorities. This problem was fixed.</p> <p>Warning: As a result of this change in the static scheduler existing applications that set priorities in the firing table may produce static schedules where the firing occurs in a different order than in previous versions of Gedae. While the schedules will still produce valid results it may be that the order of firing does not execute as efficiently as previously observed. It may be necessary for the user to set the priorities of the sink boxes higher to duplicate the previous firing order.</p>
002947	Bug in evaluation phase when arguments to function contains subexpressions.	<p>A primitive output with an expression containing a function with arguments that are parenthesized fails to evaluate. For example stream float [f:F]out[part(f,F,(N+1))]; fails to evaluate because the third argument to part is an expression with parenthesis.</p>
002951	Segfault during attach of development environment to running application	<p>It is possible for Gedae to segfault when attaching to a running application that contains internal boxes.</p>
002954	Gedae may segfault if group settings are set on a flowgraph and a box inside the flowgraph is deleted	<p>If a graph has group settings assigned to a flow graph (rather than a primitive) then if a primitive within the flow graph is deleted it is possible that gedae will segfault. This could occur, for example, if a flowgraph was assigned to run on partition 0 in the group setting file.</p>

002955	"Replace All" in Archive Browser doesn't work on Linux	
002962	Mapping equations not applied to new partitions	If a partition is mapped by equation if a new partition is added that should be mapped by the equation the equation is not applied. For example if partitions 0, 1, 2, 3 exist and are mapped by the equation 100+\$1 to logical processors 100, 101, 102, 103 then if a new partition 4 is added it should be immediately mapped to logical processor 104. Instead it is mapped to "host". This problem has been fixed.
002964	Cannot set string parameters from parameters table	
002965	File Selection Dialogs do not resize correctly	File dialogs like the Add Box dialog and File Open dialog do not resize correctly. When the dialog size is increased the Directories and Files list should increase in size.
002967	Graphs with unconnected host inputs are allowed to run	Graphs with unconnected host inputs (such as a stream/sink/scope2 with its b input unconnected) are allowed to schedule and run. This error should be detected during preprocessing during the check for unconnected inputs.
002971	Don't read in new box class description when doing a save from the primitive editor.	If a user creates a new box description in the primitive editor and saves an intermediate version of that box then Gedae reads that description. It does not reread the description on subsequent saves so any mistakes in the input or output interface definitions are not corrected until the user exits and reenters the Gedae development environment. This problem has been fixed. However, if a user adds the box to the Gedae canvas or if the user compile the primitive before adding it to the canvas Gedae will read the box description. Change to the interace that occur after this will not take affect until the user exits and renters Gedae.
002974	Opening templates from primitive editor does not work on Windows	When a user attempts to open templates from the primiitve editor on windows a choose dialgo listing all the same template is displayed.
002976	Improve documentation of Link_Func	The link_func file in \$GEDAE/<os>/<target>/bin can be customized by placing another link_func file in \$GEDAEHOME/<os>/embedded/<target>. However, this was undocumented. Documentation is now included in the "Gedae Functional Interface to Launch Package Programmers Manual".

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 lthe 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 correctly.	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 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, 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 config 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 Visual C++	Including both <X11/X.h> and <stdlib.h> introduces a compiler warning for some mixtures of versions of Exceed 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 hpererror hpererror2 in front of the include for the 2nd of these 2 files.
002472	The built 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 dev env talking to target command program.	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
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 recompile	If a primitive Input, Output or Local section is modified at 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.
002583	Math.h symbol floorf 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 floorf 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 matchup 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	

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	
002816	Partitions do not respond to stop command when break point occurs in attached command program	
002821	Stream primitive output parameters cannot be sent to primitive inputs in a different group.	If you send primitive output parameters to a primitive in a different group and the destination group is processed first then the parameters for that group are not set and gEDA fails to complete scheduling.
002823	Inplace dynamic segmented exclusive output	Inplace dynamic segmented exclusive outputs do not work. Pointer forwarding should go down only the exclusive branch on which data is forwarded.
002834	Tiled Streams: Segparams should be able to control tile dimensions	
002861	Tiled Streams: Allow boxes having a mix of input streams that are pointers and non-pointers	Currently boxes that have both pointers and nonpointer stream inputs are outlawed by the primitive parser (see 2925). Adding this feature back means that we must insure when pointer box is moved forward in schedule all boxes that drive the nonpointer input are also moved forward. We must still outlaw graphs that contain a circular dependency. A box with an input pointer that has another stream input that is not an input pointer and is connected to the box that sources the input pointer cannot be scheduled correctly. The problem is the input pointer box must be scheduled before the source box (so the input pointer can set the source output pointer) but the source box must be scheduled before the input pointer box (so the non-input pointer stream will have the tokens needed to fire). This circular dependency must be detected and reported as an error in the graph.
002868	Changes to subgraph parameters register subgraph parameter files as dirty	If a graph G has a subgraph A that has a primitive P and A's input K is connected to P's input K, then setting K on A I/Os causes A's parameter file to be dirty. The desired behavior is to ignore hierarchy when setting parameters through the FG Editor and only apply hierarchy through the Parameter Table.
002878	Cannot save bookmark of	

	untitled graph persists after titling	
002879	Function interface to LP must use synonyms or else names don't resolve	
002888	Tiled Streams: Unmapped target to host boundaries do not get conversion box added	An unmapped source driving an mt_rpart connected directly to an m_disp box running on the host is allowed. No conversion box is added and the transfer table shows the transfer type as "undefined"
002891	Inplace box after common should cause send box to be treated like its inplace.	An inplace box after a common data transfer receive currently has a copy box added to avoid modifying the data on the source. Instead we should fire any boxes on the source that use this data before the common data send.
002903	Fannout to typedef box inputs fails to schedule	A box that fans out to the inputs of a typedef box causes the gedae compiler to be confused about the connectedness of downstream boxes
002959	Changing constant values does not reinstantiate graph	
002960	Loading constants after a graph has already been instantiate with another constants file does not reinstantiate the graph	
002961	Group schedule memtype settings are easily lost	
002969	Input pointer stream cannot be directly connected to an output pointer stream	An input pointer stream currently cannot be directly connected to an output pointer stream but Gedae does not detect this during compilation and can segfault as a result when the graph is run. One resolution would be to automatically always insert a copy primitive between the input pointer and the output pointer. Another possibility is to detect cases where the input and output can be legitimately inplace and to provide a means of knowing that the input and output pointer assignment algorithms are identical. In such a case only one of the primitives needs to be scheduled and the other can be deleted.
002972	force recompile if box changes between runs.	
002973	Save-as on subgraph parameter table does not remove parent graph parameter overrides	When a subgraphs parameter table is saved using the Save-As function the parameter overrides saved at the parent graph level should be removed.
002975	Breakpoints loaded for a graph are used with a newly loaded graph	If a breakpoint file is loaded for one application graph and a second application graph is loaded using File->Open then Gedae attempts to apply the old breakpoints to the new file causing a segfault.