

OPNET 常见错误

编撰者 北京邮电大学 吕召彪

lvzhib_2001@bj1860.net

仅学术交流，勿商业用途；如需转载，请勿肢解

初学 OPNET，个人觉得要注重三个方面：

- 1，OPNET 的网络仿真机制，包括三层建模、离散事件驱动仿真机制、通信机制、有限状态机等，主要是看帮助文档中的原理部分；
- 2，OPNET 的核心函数，多看帮助文档中各核心函数的介绍以及示例很有帮助；
- 3，掌握 ODB 的调试，个人感觉 OPNET 的调试不难，关键是要善于从 OPNET 的仿真原理入手来找可能的错误所在，我的经验就是按着数据走的流程来分析。

另外，有时间的话，可以找一个 OPNET 提供的标准模块代码来看看，有利于理清通信的流程和掌握写代码的一些常用技巧。

说明：

- 1，下面材料主要来自自己平时遇到错误以及网上论坛的帖子，本人无意侵犯他人版权。
- 2，由于是平时的笔记，因此其中不乏错误和错字，也显随意潦草，大家见笑！

1, unresolved externals symbol:

错误消息如下:

```
err log——
<<< Program Abort >>>
* Time:      21:15:50 D??úè? áù?? 02
* Product:   modeler
* Program:   op_runsim (Version 10.0.A PL1 Build 2291)
* System:    Windows NT 5.0 Build 2195
* Package:   Simulation Kernel
* Function:  sim_load_repos_load
* Error:     Error encountered rebuilding repository -- unable to proceed

* Function call stack: (builds down)
```

	Call	Block	
	Count	Line#	Function

0)	1	1073741928	0x00004c00 [name not available]
1)	1	1879048703	0x00004c00 [name not available]
2)	1	-805306207	0x0000c400 [name not available]
3)	1	284	m3_main
4)	1	857	sim_main
5)	1	975	sim_init_phase2
6)	1	278	sim_load_models
7)	1	1269	sim_load_repos_load

编译是通过的, 仿真时——

```
Creating library E:\LuZhaobiao\1\?????? à í \my papers\OBS í ??? • ???? & D í
\delay_test-delay_test1.dev32.i0.nt.lib and object E:\LuZhaobiao\1\?????? à í \my
papers\OBSí??? • ????&Dí\delay_test-delay_test1.dev32.i0.nt.exp
dpt_propdel.dev32.i0.ps.obj : error LNK2001: unresolved external symbol _link_delay
E:\LuZhaobiao\1\?????? à í \my papers\OBS í ??? • ???? & D í
\delay_test-delay_test1.dev32.i0.nt.dll : fatal error LNK1120: 1 unresolved
externals
```

网友们分析的各种原因:

1), 由函数调用栈可见, 在主函数运行前就已报错, 这和运行仿真时反馈的错误一致 (建立 repository 在开始仿真之前进行), 所以无法使用 ODB 进行调试。从编译时的提示信息来看, 应该是没有找到模型中引用的 link_delay。解决的方法就是在 FILE/Declare external files... 将 link-delay 文件包括进来。『采用这个方法解决了以上问题』
但是可以利用 ODB 下编译发现是编译哪个模型或外部文件时报错, 确定错误范围。

2), 对于 unresolved external symbol, 应该表示引用的 external symbol 没有找到定义, 你查查 link 中的 model attribute 和你的 Process 中有没有用到提示中的 symbol, 需要 extern 声明一下。

另外, 声明了太多的本模型下没有用到的外部文件, 也会导致出现这样的错误, 解决方法就是去掉不用文件的声明。

3), 自己在其它编辑器上写的 h 和 c 代码可以在你要用到那个 c 中的函数的 Process 中选择 File-->Declar External Files... 中选中你自己写的 ex.c 吧, 这样让 OPNET 知道外部的文件, 编译好象意味着能找到 c 中的函数, 只是找到了一个声明而已。

4), 是 link model 的延时参数没有设置正确仔细对照 tutorial 看看

【 在 landmass (江山) 的大作中提到: 】

: run 时出现错误信息:

: Creating library C:\op_models\pksw\pksw_net-Scel.i0.nt.lib and object

: C:\op_models\pksw\pksw_net-Scel.i0.nt.exp

: dpt_propdel.i0.ps.o : error LNK2001: unresolved external symbol

: _link_delay

: C:\op_models\pksw\pksw_net-Scel.i0.nt.so : fatal error LNK1120: 1

: unresolved externals

: 怎么办?

5), 是版本不符的问题?nbsp;在 configure simulation 时, 点击左下角的 Declare External Files... 然后选择包含 link_delay

【 在 haobow (拣漏成功) 的大作中提到: 】

: 编译成功, 运行仿真时出现以下错误:

: unresolved external symbol _link_delay

: fatal error LNK1120: 1 unresolved externals

: 以上错误信息是 vc 返回的。感觉像是没有联接 opnet 关于链路的标准库

: \opnet8.0c\...\

: models\std\links。但我在系统变量中关于 opnet 的目录都设了的, 包括 lib, bin...

: 去年五月份在学校里用得好好的, 大半年不用就忘了怎么设得了。请大虾支招。

unresolved externals symbol:

如何发现未声明的外部文件:

去掉开始的_, 然后前面几个单词就是为声明的外部文件名前面字符。如

_oms_data_def_entry_access

则是没有声明 oms_data_def 文件。

2, unreferenced local variables——定义了本地变量，但是没有引用过。一般属于 **warning**。

3，无法写入文件

```
<<< Recoverable Error >>>
```

```
write to file failed
```

```
File Path: E:\LuZhaobiao\1\????? à í \my papers\OBS í ??? • ??? £ D í  
\delay_test-delay_test1_data.of
```

```
T (51.1843), EV (19594437), MOD (top.subnet_0.node_0.p_24)10:24 2004-6-6
```

```
write to file failed
```

```
File Path: E:\LuZhaobiao\1\????? à í \my papers\OBS í ??? • ??? £ D í  
\delay_test-delay_test1.ov
```

```
T (1000), EV (393027170), MOD (NONE)
```

原因分析：

1) 一般最好用英文的目录和文件名保存文件，不要用中文的目录和文件名。否则，opnet 可能无法在指定的中文文件夹下建立文件。

4，最后没有通信，仿真中止。

```
Simulation terminated without final communication.
```

```
Check DES log (select 'Open DES Log' in 'DES' menu) for possible error messages.
```

1) 重新仿真一次后就没有该提示了。原因分析，似乎是计算机反应迟钝。

5，output stream index (0) is out-of-range.

```
<<< Recoverable Error >>>
```

```
* Time: 19:51:40 D??úúú àù?? 05
```

```

* Product:    modeler
* Program:    op_runsim (Version 10.0.A PL1 Build 2291)
* System:     Windows NT 5.0 Build 2195
* Package:    process (_packet_capsule) at module (top.subnet_0.node_0.q_0)
* Function:    sim_strm_send
* Error:      Output stream index (0) is out-of-range.
               T (3234.39), EV (4231), MOD (top.subnet_0.node_0.q_0), KP
(op_pk_send)

```

* Function call stack: (builds down)

Call	Block		
Count	Line#	Function	

0)	1	1073741928	0x00004c00 [name not available]
1)	1	1879048703	0x00004c00 [name not available]
2)	1	-805306207	0x0000c400 [name not available]
3)	1	284	m3_main
4)	1	935	sim_main
5)	1	2364	sim_ev_loop
6)	2115	387	sim_strm_insert
7)	4232	522	sim_obj_qps_intrpt
8)	2116	21	_packet_capsule [arrival enter execs]
9)	2256	6918	op_pk_send (pkptr, outstrm_index)
10)	2256	470	sim_strm_send

原因分析:

1) 输出流索引超出范围, 可是 0 是输出流索引的起始编号, 不可能超出的。结果发现, 这是一个队列模块, 而该队列模块只有输入流, 没有输出流连接。解决方法, 添加一个处理器模块 (sink), 用包流连接队列和处理器。

6, 指定了未存在的对象

可恢复错误。

<<< Recoverable Error >>>

```

* Time:      10:41:40 D??ùè? áù?? 06
* Product:    modeler
* Program:    modeler (Version 10.0.A PL1 Build 2291)
* System:     Windows NT 5.0 Build 2195
* Package:     External Tool Support Programmable Editor Interface / Topology
Subpackage
* Function:    Ets_Topo_Child_Get_By_Name

```

* Error: invalid ets object handle type

* Function call stack: (builds down)

	Call	Block	
	Count	Line#	Function

0)	1	108	0x00004c00 [name not available]
1)	1	518	0x78cd7e0a [name not available]
2)	1	174	0x6400c506 [name not available]
3)	1	137	vos_event_loop
4)	1041	61	vos_event_client_vevent_handler
5)	131	436	vmmi_ets_action_notify
6)	3	5429	pj_command_scenario_statistics_view
7)	3	6975	pj_tool_results_view
8)	3	6726	an_chooser_dbox_open
9)	33	944	Mmi_Ctl_Notebook_Current_Page_Set
10)	34	8760	vmmi_ctl_notebook_set_current_page
11)	8273	151	vmmi_ctl_event_dispatch
12)	5	5355	an_chooser_dbox_page_change_handle
13)	5	5527	an_chooser_dbox_ov_refresh
14)	11	6324	an_chooser_tree_view_objects_remove
15)	12	6324	an_chooser_tree_view_objects_remove
16)	13	6324	an_chooser_tree_view_objects_remove
17)	14	6307	an_chooser_tree_view_objects_remove
18)	4	6873	pj_tool_results_view_not_in_view_filter
19)	4	10895	Ets_Obj_From_Hname_Get
20)	12	1486	Ets_Topo_Child_Get_By_Name

原因分析:

1) 网络域中，一条链路中指定的 transmitter 和 receiver 在其连接的节点中并不存在。在 DES log 中会指出链路指定了不存在的对象。

7，在统计量收集模型中指定收集不存在的统计量。

告警

<<< Warning >>>

- * Time: 10:42:42 D??úè? àù?? 06
- * Product: modeler
- * Program: modeler (Version 10.0.A PL1 Build 2291)

```

* System:      Windows NT 5.0 Build 2195
* Package:     Network Statistic Browser
* Function:    nt_global_collect_node_create_with_handlers
* Error:       Some of the statistics marked for collection in the selected report
may not
                be collectable in this scenario.  Only collectable statistics are
displayed.

```

```

* Function call stack: (builds down)

```

	Call	Block	
	Count	Line#	Function
0)	1	108	0x00004c00 [name not available]
1)	1	518	0x78cd7e0a [name not available]
2)	1	174	0x6400c506 [name not available]
3)	1	137	vos_event_loop
4)	1070	61	vos_event_client_vevent_handler
5)	134	436	vmmi_ets_action_notify
6)	1	5822	pj_command_choose_results
7)	1	7792	pj_tool_results_choose
8)	1	3462	nt_global_collect_browser_create
9)	1	3438	nt_global_collect_node_create
10)	1	3405	nt_global_collect_node_create_with_handlers

原因分析:

- 1) 在仿真中使用的统计量收集模型中，指定了很多不存在的对象的统计量。
- 2) 收集一个统计量，首先在进程中进行收集定义，其次在进程模型中进行统计量声明，最后在节点模型的 **INTERFACE/NODE STATISTICS** 中进行提升。（如果不是节点统计量, 则不需要在节点统计量中声明）

8，有时无法使用 **choose。。statistics** 进行统计量收集定义

原因分析：可能是由于没有删除 recovered scenario。导致无法选择被收集的对象以及统计量。

情形二：使用 `choose statistics(advances)` 指定统计量收集是，却提示 `no global statistic are available` 或者指定节点统计量是提示无法打开节点模型。

原因分析：从提示信息中会发现提示无法打开的节点模型和我们实际使用的节点模型不一样，原因是这个场景是通过复制得到，而以前场景使用的节点模型被删除，重新赋予了新的节点

模型，可是探针模型中并没有更新。操作：Objects/Set Network Model。

9，无效内存访问

这类错误调试十分麻烦，错误也很隐蔽。

一般是由于读内存越界所致。例如，子队列为空，但是没有做判别就使用 op_subq_pk_remove 读取包，由于子队列为空，所以将读不到任何数据。

10，指针问题

包指针引用了一个没有归属的包。

Program Abort reported by process (burst_create) at module (top.subnet_0.node_2.assembly_1) at T (4.0298), EV (6197), MOD (top.subnet_0.node_2.assembly_1), KP (op_sar_segbuf_pk_insert)
Packet pointer references unowned packet (2507)

11，仿真编译问题 1:

经验:

- 很多出错信息都包含在
C:\Documents and Settings\lvzhbl\op_admin\tmp\cc_err_2020。
- 很多问题都是相互关联的，一个错误可能导致好几个报错。

Simulation Kernel: development, sequential

<<< Recoverable Error >>>

Unable to read result collection file (OPS_assembly_comparsion-hurst_parameter)

注释:

<<< Recoverable Error >>>

Unable to read packet model: pk_format

The specified file does not exist.

注释:

<<< Recoverable Error >>>

Process model (Sup_FRP_spawn_pkfor) compilation failed
due to error on source line (26)

Errors given in file (C:\Documents and Settings\lvzhbl\op_admin\tmp\cc_err_2020)

注释: 这是由于找不到头文件。

<<< Recoverable Error >>>

Object repository construction failed

Process model (E:\workoffice\opnet\process model\Sup_FRP_spawn_pkfor.pr.c)
compilation failed, see error messages in (C:\Documents and
Settings\lvzhl\op_admin\tmp\cc_err_2020)

注释：这个报错是由于上一个错误引起。上一个是由于进程编译无法通过才报错；这一个是由于建立 repository 找不到文件而报错。

<<< Program Abort >>>

Error encountered rebuilding repository -- unable to proceed

解决方法：

将需要的头文件和进程文件放到一个文件夹中。

此时，出现

```
Creating library E:\workoffice\opnet\assembly
comparsion\OPS_assembly_comparsion-hurst_parameter.dev32.i0.nt.lib and object
E:\workoffice\opnet\assembly
comparsion\OPS_assembly_comparsion-hurst_parameter.dev32.i0.nt.exp
oms_bgutil.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_data_def_entry_insert
oms_bgutil.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_data_def_entry_access
oms_bgutil.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_basetraf_support_tracer_pk_dealloc_proc
oms_bgutil.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_basetraf_segm_info_mem_alloc
oms_bgutil.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_basetraf_segm_info_dealloc
oms_bgutil.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_basetraf_rr_info_write
oms_pipeline.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_basetraf_set_bgutil_for_demands
oms_pipeline.dev32.i0.ex.obj : error LNK2001: unresolved external symbol
_oms_pr_process_discover
E:\workoffice\opnet\assembly
comparsion\OPS_assembly_comparsion-hurst_parameter.dev32.i0.nt.dll : fatal error
LNK1120: 8 unresolved externals
```

unresolved externals symbol:

如何发现为声明的外部文件：

去掉开始的_，然后前面几个单词就是为声明的外部文件名前面字符。如

_oms_data_def_entry_access

则是没有声明 oms_data_def 文件。

12，无法读取结果收集文件：或者无法读取分析配置文件

<<< Recoverable Error >>>

Unable to read result collection file (OPS_assembly_comparsion-hurst_parameter)

原因分析：

这是由于没有定义要收集的统计量，也就是没有定义探针模型文件。探针模型文件默认的文件名是工程名-场景名。

这个是运行仿真时出现的错误提示。

13，无法读取包格式模型：

<<< Recoverable Error >>>

Unable to read packet model: pk_format

The specified file does not exist.

原因分析：

指定的包格式没有定义。

一种情况是在进程模型的代码中生成指定的包格式时，指定了未定义的包格式。

二种情况是生成指定包格式的进程模型中声明包格式时，指定了未定义的包格式。

14，进程编译错误信息

OPS_edge_pkalloc.pr.tmp.c

E:/workoffice/opnet/process model/OPS_edge_pkalloc.pr.c(177) : warning C4550: expression evaluates to a function which is missing an argument list

原因分析：

由提示信息可知，错误发生的原因是

1) 表达式出错

2) 函数缺少参数列表

所以错误应该定位在含有函数的表达式中，并且是函数缺少了形式参数列表。
从源代码中 177line 分析发现是，状态转移语句存在问题，原因是状态转移图中的转移程序函数丢失了括号。

15，读不到节点属性值

语句没有问题。

最后发现问题出现在节点属性设置上。

- 1) 在节点域上，Interface/model attribute 的设置中，属性值的数据类型（设置为 integer）与进程中用来保存属性值的变量的类型（设置为 double）不一致。

16，编译的告警 bug

在 diagnostic block 中输入代码，编译报错，提示很多变量 unreferenced。后来发现这些没有引用的变量都是 local temporary variables。查了自己的代码，的确在 diagnostic block 中没有引用任何临时变量。但我觉得奇怪，不引用怎么会告警，不知道是不是我代码哪里有问题。怎么能不让 OPNET 告警。

答案：这个是 OPNET 的 bug，不用理会。

后来将 diagnostic block 中的代码删去，编译时仍然如此报错。后来发现是环境参数设置不恰当，将

comp_flags_common 重新设置成

-ID:\Progra~1\OPNET\10.0.A\models\std\include，也就去掉标志/W3 就不再报错。

17，FAQ for OPNET from cnl.gmu.edu

1.Binding Problem

Set these variables in your [env db file](#) in your op_models directory:

comp_prog: "/opt/bin/gcc"

bind_libs: "-L/usr/ucb/lib -L/usr/lib -lucb -lresolv -lsocket -lnsl -lelf"

2.Simulation Terminated Abnormally.

- a. Don't start opnet from the xterm with black background (site)

- b. Check for logic (process model)
- c. See [OPNET Debugger \(ODB\)](#)

////****

OPNET Debugger (ODB)

One of the features in OPNET to help you trace your errors is the Debugger. This tool is also useful when the simulation program gets stuck at run time or when the simulation is taking too long. You run the debugger from a Unix C Shell.

When running a simulation, if you encounter an error, an error banner will be displayed with information concerning that error. In that banner you will see two important fields, event number (displayed ev(#)) and time (displayed T(#)). These are the event number and time at which the error occurred.

Now, run the simulation using -debug. Once in ODB, there are two commands you can use to go to that error. The preferred method is to go to the error via the event number. The command evstop will do this.

```
example odb> evstop #
```

The other method is to use tstop and go to the time the error occurred.

```
example odb> tstop time
```

Once at the error, type "full" to get into full trace mode and type next to continue. This should show you where the error occurred.

```
example odb> full
```

```
example odb> next
```

****//

3. Printing Problem

Change your print_prog variable in [env_db file](#) as print_prog: "lpr -h -Prm18a" where "rm18a" is a printer in ST2-lab. (choose your own)

4. Program Abort "no true transitions from state"

- a. First check state is it FORCED or UNFORCED? (eg. idle is UNFORCED)
- b. Check the logic (transition conditions) add "default" or other conditions as needed.
- c. Enable begin simulation attribute of the node. (eg. queue)

5. Quota Problem

- a. OPNET(tm) creates backup of your models and the printjobs (ps files) in ~/op_admin/tmp remove these if you don't need them. (use m3_clrtmp)
- b. Remove the files with extensions ".sim" and ".ah" from your ~/op_models directory after you are done with the assignment. (simulation & animation)

These FAQ material consists of e-mails and replies to them:

Empty Plot

Question 1

Questions:

I have successfully compiled my new acb_fifo process and built a new node and network with it. My network has Archived and Binded successfully.

I am having problems in using the "probe editor" for determining blocking statistics.

When I create a statistic probe and set the submodule attribute to NONE oustat (0) is available for selection in the statistic attribute, but there is no information available for plotting after the simulation is run.

When I set the submodule attribute to subqueue (0), oustat (0) is NOT available in the statistic attribute, and again no information is available for plotting. When I state that no information is available for plotting I mean that a graph window comes up but there is no data there. Why is there no information in the plot?

What should the submodule and subqueue attributes be set to?

I never used the op_q_stat() function, is it needed?

I found no information on op_q_stat() in the Simulation Kernel, Basic Process Tutorial, or Process Domain Definitions Manuals - is there some where else to look?

Answer 1

ANSWERS:

Every queue process model has a set of predefined variables that are automatically updated by the process model itself. The names and descriptions of these variables are shown under "Queue Statistic Symbolic Constants" in the Symbolic Constants section of the Simulation Kernel Manual. The current value of any of these can be retrieved with the "op_q_stat" procedure, which is documented in the Simulation Kernel Manual in the chapter on queueing procedures.

The current value of ,e.g., OPC_QSTAT_OVERFLOW can be retrieved and used to calculate percentage of blocked packets. [The result can be written to an output statistic that has been defined in the process model.](#) See pages Bpt-12 to Bpt-14 of the Basic Process Tutorial for an example of how to define and write to an output statistic. [Once you have properly defined the new outstat, its value can be monitored via a statistic probe by selecting the appropriate OUTSTAT of the queue module.](#) Note that since you have defined the statistic in the queue module (not the subqueue), you would set the "submodel attribute" to NONE in the probe definition.

Outstat Variable

Question 2

I am assuming that there is a variable already defined in OPNET that keeps track of the blocked arrivals. I have looked in the symbolic Constants and did not find it. Since we are suppose to write the proportion of blocked arrivals (blocked / total arrivals) , where do I find that variable (packets blocked ?). Thank you.

Answer 1

ANSWERS:

Every queue process model has a set of predefined variables that are automatically updated by the process model itself. The names and descriptions of these variables are shown under "Queue Statistic Symbolic Constants" in the Symbolic Constants section of the Simulation Kernel Manual. The current value of any of these can be retrieved with the "op_q_stat" procedure, which is documented in the Simulation Kernel Manual in the chapter on queueing procedures.

The current value of ,e.g., OPC_QSTAT_OVERFLOW can be retrieved and used to calculate percentage of blocked packets. The result can be written to an output statistic that has been defined in the process model. See pages Bpt-12 to Bpt-14 of the Basic Process Tutorial for an example of how to define and write to an output statistic. Once you have properly defined the new outstat, its value can be monitored via a statistic probe by selecting the appropriate OUTSTAT of the queue module. Note that since you have defined the statistic in the queue module (not the subqueue), you would set the "submodel attribute" to NONE in the probe definition.

op_q_stat

Question 3 not left value

Question

I have added the following lines in the acb_fifo processor

```
arival++;      /*this is in the beginning of the routine*/  
OUTSTAT +=op_q_stat (OPC_QSTAT_OVERFLOWS); /*this line and the next are*/  
                                          /* at the end of the routine*/  
  
op_stat_local_write(OUTSTAT, (OUTSTAT/arival));
```

I get the error message "invalid lvalue in line which has the op_q_stat function". I don't know what that means?? Am I doing something wrong??

Answer 3

Answer

The 'OUTSTAT' cannot be used on the left side of the equation. You need to define some name for OUTSTAT 0 (see the Basic Process Tutorial for an example), then define a state variable to use on the left side of the equation and write the state variable to the OUTSTAT you defined. You may be able to just put the name of the defined OUTSTAT on the left side, I'm not sure. But with the code shown above you haven't told the function which OUTSTAT it should be using.

note: 'lvalue' = variable on the left side of an assignment statement

Scalar Statistics

Question 4

I am having problems trying to get the scalar statistics to plot out.

I have run the simulation ten times and the values for "Mean End-to-End Delay", "Throughput", and "Return Probability" seem to be constant. Since OPNET only allows a scalar value to be plotted with another scalar, I get straight lines in my plot. Do you have any suggestions

Answer 4

Did you vary the seed value each time? If not, the numbers would be exactly the same on each run (it's a pseudo-random generator.) You can take a look at the actual scalar values by clicking the right mouse button on your scalar plot and selecting one of the options on the pulldown menu, I think it's "show values" but you can just try the various options to see.

pk_sink Process Compilation

Question 5

Q:

1. I am getting several errors when compiling the pk_sink process. Most of

the errors are parse errors which I am not sure what they really mean.

2. Do we need any code in the INIT state of pk_sink?

Answer 5

We do not need any code (ENTER execs or otherwise) in the INIT state of pk_sink. The only initialization needed is for

"total_pkts_rcvd", and this is done in its Header definition. (在 HB 中定义变量和常数时可以初始化)

If you are getting "parse" errors they are probably due to misspellings or syntax problems somewhere in your code. The error messages will lead you to the exact lines of code that have a problem - please examine each such line carefully, verifying syntax. Remember that syntax for kernel procedures is shown in the KP manuals. Also make sure that each line of code in the EXECs ends with a semicolon, and remember to watch upper case versus lower case characters - they are NOT the same in C.

What to do when OPNET crashes?

Answer

If Opnet crashes or hangs up while you are editing a model, you may find that the next time you try to edit the model Opnet will not let you modify it. This is because the model files retain file locking information which must be reset if the editing process is aborted.

There are two workarounds for this. The simplest is to go ahead and open the model as "read-only", then save it under a different name. You could then delete the old model file and 'mv' the new name back to the previous name. The other way is to reset the edit lock on the offending file. Cd to your op_models directory and type:

```
m3_manfile -m model -header_print
```

substituting the name of your model for 'model.' This will show you whether the model file has an edit lock. If it does, you can release the lock by typing:

```
m3_manfile -m model -edit_unlock
```

You will need to exit and re-open the model after releasing the edit lock in order to re-save it under the original name.

This info came from the 'External Interfaces' chapter of the Opnet online manual.

Why can't I access my previous Opnet Models?

[Answer](#)

Due to some system changes, some people may not be able to access their Opnet models - e.g., if you go to the Node Editor and try to pull up a model that you had defined previously it may not show up on the list. Don't worry, your models are probably still there but Opnet is looking in the wrong place.

You may be able to correct this by running the 'op_newuser' script again (I haven't tried this), otherwise you need to edit your 'op_admin/env_db' file. On the line that says 'mod_dirs: /home//op_models,' change it to 'mod_dirs: /student//op_models,' so it will point to your private model directory.

18, objid out of range

已知两个 node 之间只有一个双向的链路

但是下面这条语句返回的 l_id 总是 out of range 负值

```
l_id=op_topo_connect(s_id,d_id,OPC_OBJTYPE_LKDUP,0);
```

为什么?

解释: 这条语句的语法没问题, 可能是参数取的不对, 就是 s_id 或 d_id 取的不是所要求的 ID 值。

19, 显示 comp_msvc 编译命令的告警信息:

```
OPS_assembly_hybrid_1.pr.tmp.c
comp_msvc: issued the command: CL /c /nologo /MD /QIfdiv- /Zl
/DHOST_PC_INTEL_WIN32 /D__STDC__ -DVOSD_FIN_TRACING
-ID:\Progra~1\OPNET\10.0.A\models\std\include -DOPD_SUPPORT_PRIORITIES
-DOPD_ALLOW_ODB /Zi /Od /DHOST_PC_INTEL_WIN32 /D_X86_ "C:\Documents and
Settings\lvzhh1\op_admin\tmp\OPS_assembly_hybrid_1.pr.tmp.c"
/Fo"E:\workoffice\opnet\process
model\OPS_assembly_hybrid_1.dev32.i0.pr.obj"
/ID:\PROGRA~1\OPNET\10.0.A\sys\include /I"E:\workoffice\opnet\process
model"
```

解释: 这是 preference 的设置问题, 关闭 debug_mk 开关即可编译时不再显示这些编译信息。

20，如果统计量事先没有进行注册，那么编译时是不会提示错误的。

21，OPNET 10 中 add model directory 有问题，不能立刻反应变化。

解决：

如果 add model directory，那么需要进行 refresh model directory 操作。但是，如果该文件夹下没有文件，那么在 open 窗口中不会显示出该文件夹。

OPEN 窗口中的列表具有过滤功能，包括对文件夹的过滤。

另外如果对文件夹进行了删改，那么必须要在 preference 中进行修改，否则不能显示出所作的删改。

22，Repository missing process model (dynproc_root)

Simulation Sequence 窗口输出信息如下：

Beginning simulation of dynproc_new-root_process at 10:11:59 星期日 三月 14

Simulation Kernel: development, sequential

<<< Warning >>>

Repository missing process model (dynproc_root)

<<< Program Abort >>>

The set of models necessary for running the simulation is incomplete.

Check that all the 'repositories' attributes are complete.

Err log 中信息:

<<< Warning >>>

```
* Time:      10:13:08 ĐÇÆÚÈÕ ÈýÔÂ 14
* Product:   modeler
* Program:   op_runsim (Version 10.0.A PL1 Build 2291)
* System:    Windows NT 5.0 Build 2195
* Package:   Simulation Kernel
* Function:   sim_load_repos_prmatch
* Error:     Repository missing process model (dynproc_root)
```

* Function call stack: (builds down)

	Call	Block	
	Count	Line#	Function
0)	1	1073741928	0x00004c00 [name not available]
1)	1	1879048703	0x00004c00 [name not available]
2)	1	-805306207	0x0000c400 [name not available]
3)	1	284	m3_main
4)	1	857	sim_main
5)	1	975	sim_init_phase2
6)	1	230	sim_load_models
7)	1	1307	sim_load_repos_load
8)	1	1723	sim_load_repos_prmatch

<<< Program Abort >>>

```
* Time:      10:13:32 ĐÇÆÚÈÕ ÈýÔÂ 14
* Product:   modeler
* Program:   op_runsim (Version 10.0.A PL1 Build 2291)
* System:    Windows NT 5.0 Build 2195
* Package:   Simulation Kernel
* Function:   sim_load_repos_load
* Error:     The set of models necessary for running the simulation is
incomplete.
```

Check that all the 'repositories' attributes are complete.

- * Function call stack: (builds down)

	Call	Block	
	Count	Line#	Function
0)	1	1073741928	0x00004c00 [name not available]
1)	1	1879048703	0x00004c00 [name not available]
2)	1	-805306207	0x0000c400 [name not available]
3)	1	284	m3_main
4)	1	857	sim_main
5)	1	975	sim_init_phase2
6)	1	230	sim_load_models
7)	1	1380	sim_load_repos_load

注：此时 repositories 参数设置如下：

```
Name: repositories    Value: stdmod      Source:
env.file    Group:Development
```

默认的 repositories 参数设置如下:

Name: repositories Value: empty Source: default Group:Development

Make sure that the repositories preference is set correctly before running the simulation.

解决方法:

这个问题的出现是因为 `repositories` 参数设置不正确，导致仿真时无法在 `repositories` 指定的找到需要的进程模型 `process model`。改为默认设置。

23, Unable to open file (Permission denied)

错误信息如下:

```
Beginning simulation of dynproc_new-child_process at 11:47:10 ĐÇÆÚÒ» ÈýÔÂ
15
```

```
Simulation Kernel: development, sequential
-----
```

```
<<< Recoverable Error >>>
```

```
Unable to open file(Permission denied)      /*这个错误的出现是由于该日志
文件当时被设置为只读，无法访问。问题是此时进行的是另外一个文件夹中的实
验，怎么日志会写到此文件中呢？*/
```

```
File Path: E:\LuZhaobiao\OPNET\myworking\王文博书实验备份
\Lab_4\dynproc_new-child_process.nt.log
-----
```

```
<<< Recoverable Error >>>
```

```
Unable to open log file (dynproc_new-child_process) for simulation logging.
Log file will not be created for this simulation.
-----
```

```
Module (2), (top.Node.dp)
```

```
From procedure: dynproc_table_display (prefix, table)
```

```
Packets handled: (0) 1769 (1) 1127 (2) 774
-----
```

```
Module (2), (top.Node.dp)
```

```
From procedure: dynproc_child_status ()
```

```
Packets handled: (0) 1769 (1) 0 (2) 0
-----
```

```
Module (2), (top.Node.dp)
```

```
From procedure: dynproc_child_status ()
```

```
Packets handled: (0) 0 (1) 1127 (2) 0
-----
```

```
Module (2), (top.Node.dp)
```

```
From procedure: dynproc_child_status ()
```

```
Packets handled: (0) 0 (1) 0 (2) 774
-----
```

```
Simulation Completed - Collating Results.
```

```
Events: Total (7,345), Average Speed (20,346 events/sec.)
```

```
Time: Elapsed (0.36 sec.), Simulated (1 hr. 0 min. 0 sec.)
-----
```

```
<<< Recoverable Error >>>
```

```
Unable to open file
(Permission denied)
```

File Path: E:\LuZhaobiao\OPNET\myworking\王文博书实验备份
\Lab_4\dynproc_new-child_process.ov

<<< Warning >>>

Unable to open file: dynproc_new-child_process.ov

Used secondary file instead:

E:\LuZhaobiao\OPNET\myworking\OpnetBookLab\Lab_2\dynproc_new-child_process_1704.ov

解决方法:

以上问题的出现是由于其中的 dynproc_child 进程模型是先前编译过的，可能其中将自身与以前的位置关联。只要在当前文件夹下重新编译一次就可以解决问题。分析的信息来源见蓝色部分。

解决方法：这个问题的出现很可能是由于中文目录名所致。在 OPNET 仿真中，最好全部使用英文名称。

24, op_stat_reg("Assembly Packet Size", OPC_STAT_INDEX_NONE, OPC_STAT_GLOBAL) 用错：在进程模型的 local statistics 中声明，而在注册时注册为 GLOBAL statistics。

Simulation Kernel: development, sequential

<<< Recoverable Error >>>

Unable to register global stat (Assembly Packet Size). Please check spelling and include a group name if stat is defined with a group name.

T (0), EV (2), MOD (top.OPS_edge_router.transmission_queue), KP (op_stat_reg)

<<< Recoverable Error >>>

Attempting use of unregistered statistic handle.

T (5.00005), EV (27), MOD (top.OPS_edge_router.transmission_queue), KP (op_stat_write)

25.1,仿真运行完之后,在观察结果 **view results** 时发现没有结果显示.

问题之一:在进程模型中编写了统计量收集代码(统计量注册/统计量写入),在进程模型中进行了统计量声明(本地统计量,全局统计量),在探针模型中(choose individual statistics)定义了统计量收集.

问题:在 view results/RESULTS(view statistics)发现没有结果.view results(advanced)发现"Analysis Configuration model () could not be read.The specified file does not exist."

解决措施:

这个问题的产生不是因为使用 choose individual statistica 或 choose statistics(advanced)的区别,这两种探针收集定义是一致的,没有区别.

这个信息是提示没有定义分析配置模型,因为事先没有定义,当然读取模型时会报错.装载生成的输出矢量文件后保存,这个文件就生成了.

可是 view results 中仍然没有结果显示:

这里的问题可能发生在节点统计量的定义和理解上。在探针模型 DES/ choose individual statistica 或 choose statistics(advanced)中指定的是节点统计量,那么是不是应该先在节点模型中声明节点统计量呢,否则这样定义统计量的收集可能不会得到结果。

如果不声明为节点统计量,也可以通过每个节点的右键菜单选择 choose individual DES statistics,然后定义模块统计量。

25.2 仿真运行完后,在 **view statistics** 中看到统计量仅显示横坐标,纵坐标没有刻度,而且没有任何结果输出(图形中空白,没有蓝线)

分析原因:说明仿真中根本没有对统计量进行收集,那么必须检查收集统计量的几个步骤,如果是矢量,则:声明统计量、注册统计量、写统计量,另外则重要的一个方面就是统计量的计算过程。

解决方法:后来发现统计量的计算过程有问题,计算中涉及到一个 SV,这个 SV 的值来自一个模型属性,可是模型属性定义为 double,而 SV 定义为 int,在读模型属性进行 SV 赋值时没有实际赋值,可是也不报错。

25.3 仿真运行完后，在 **view statistics** 中看到统计量仅显示横坐标，纵坐标没有刻度，但输出结果（图形中显示一条纵坐标刻度为 0 的水平蓝线）

分析原因：此时可以查看结果统计数值，看最大值和最小值，检查是否存在无限大值。
解决方法：后来发现存在无限大，这说明由于存在无限大，所以纵坐标刻度无法全部显示，故显示从 0 开始的没有刻度直线。这种原因往往是进行了 0 除操作。

26，启动 OPNET 时出现问题，无法启动。

首先，可以从 err_log 中得到错误信息提示。

如下：

```
<<< Warning >>>
* Time:      09:06:34 星期四 九月 02
* Product:   Generic Product
* Program:   modeler (Version 10.0.A PL1 Build 2291)
* System:    Windows NT 5.0 Build 2195
* Package:   Vos (Virtual Operating System) / Env (Environment Variable)
* Function:  vos_env_file_parse
* Error:     parse error in env. file: expected colon but found word
              (discontinuing parsing of this file)
              File Path: C:\Documents and Settings\lvzhl\op_admin\env_db10.0,
Line: 3
```

* Function call stack: (builds down)

		Call	Block
		Count	Line# Function

0)	1	3169	m3_main
1)	1	857	Op_Std_Open
2)	1	2023	Vos_Env_Std_Sources
3)	4	1592	Vos_Env_Pair_List_File_Read
4)	4	2501	vos_env_file_parse

<<< Program Abort >>>

```
* Time:      09:06:37 星期四 九月 02
* Product:   Generic Product
* Program:   modeler (Version 10.0.A PL1 Build 2291)
* System:    Windows NT 5.0 Build 2195
* Package:   Mmi (Man-Machine Interface)
* Function:  mmi_tool_config_file_parse
* Error:     Unable to load editor configuration file (modeler)
              because the file could not be found.
```

* Function call stack: (builds down)

				Call	Block
				Count	Line# Function
<hr/>					
0)	1	108	0x00004c00	[name not available]	
1)	1	518	0x78cd7e0a	[name not available]	
2)	1	174	0x6400c506	[name not available]	
3)	1	3753	m3_main		
4)	1	3616	Mmi_Tool_Environment_Create_Core		
5)	1	3458	mmi_tool_config_file_parse		

后来发现是配置文件 env_db10.0 出现问题，被改动了。

第二行应该是

```
# Created automatically on 16:27:51 星期三 九月 01 圖?罾 g 圖?814.$814.$& for program
modeler.
```

结果从中断开了，所以找不到结尾符号“.”。

27，内存泄漏问题：

主要就是因为分配了包没有及时销毁不在使用的包。

现象：

第一个，可以从 memory usage 曲线可以看出来，曲线会很快的不停增长。

第二个，可以从 memory stats 中查看包分配的内存情况，从而可以发现是何种格式的包耗用内存。

解决方法：

后来发现问题出现中转发包的模块中，进行转发工作的状态设置成了非强制状态，结果包转发后不能立刻回到等待状态，从而每次转发了一个包后，还必须通过一次流中断来使状态回到等待状态，显然，这个过程中有一半的包没有转发出去。

另外，仿真过程中，内存在不停增长，并不完全是因为包使用后被销毁造成的。例如，包源

的生成速率远远大于节点发射机的发射速度，由此，在发射机模块中就会积累大量的包无法及时发射出去，这时内存就会消耗很厉害，但是不会造成崩溃。

内存泄漏的原因之二：

链表中的元素从链表中删除后，没有及时释放内存；

内存泄漏的原因之三：

仿真事件数目急剧增加，导致仿真事件链表无限增大。比如，在分组组装中，如果时间门限设置得过去，那么单位时间内需要调度的事件非常多，这样会消耗大量内存。

28，如何生成统计仿真结果的配置文件。

opnet 中的 pb.m 文件是二进制文件，比较复杂，我需要统计的值不多，但是不知道该怎么写这种文件，或者有什么其他方法可以得到需要的结果。

解决方法：

pb.m 文件是系统生成的，就是你在 OPNET 里面选择统计数据时 OPNET 自己生成，仿真时调用这个配置文件。???? 还没有解决。

29，opnet 仿真结果读取的问题

opnet 的仿真结果应该在.os 和.ov 文件中吧，有什么办法可以转化成文本格式呢？

的确用 ema 就可以操作.ov 文件，ema 的说明文档中就有函数操作 ov 文件里面的；至于.os 文件我没试过，不知到用什么可以解决。

找到了转换.os 的命令，可以转成.gdf 格式，用 ultraedit 看跟文本一样，不过这个命令暂时记不起来了，好像是**_cvos，**可能是 op 吧

external interface->external model access->output vector 里面有.ov 文件格式的说明以及里面数据的存放格式，结合前面的 ema 的函数就可以把.ov 文件里面的数据读出来

30，操作有时无法生效

问题：有时在 OPNET 中对配置进行了更改，可是发现重新打开窗口后所做的修改并没有生效。

解决方法：

每次进行了操作后，最好进行保存，OPNET 似乎一般不会自动进行保存。所以进程的 save 操作是十分有必要的。

31，无法找到头文件

问题：

运行 opnet 自带的 wlan 的模块，出错说一个头文件找不到，可是看了一下，那个头文件在 opnet 的安装目录里，就是\models\std\include 里，这个目录在 mod_dirs 里已经包括进去了。

解决方法之一：在 cmd 里执行 path, 看一下有没有 PATH=C:\Program Files\OPNET\10.0.A\sys\pc_intel_win32\bin; 如果没有，在环境变量里加进去。

解决方法之二：可能是外部的头文件（也就是不在\models\std\include），看是否在进程模型中是否声明了外部文件。

解决方法之三：还是不行的话，就将该文件拷贝的进程模型文件的同一文件夹下。

31，速率类统计量的定义

速率类统计量的定义如下：

```
op_stat_write(A, 1);  
op_stat_write(A, 0);
```

然后再结合统计量的收集模式就可以计算出速率类统计量的结果。
在定义探针模型时，各个探针还有个收集模式的选项，包括

all values, \sum , \sum /t 等等类型。

32，Tda 索引超出包范围。

<<<< Program Abort >>>>

Tda index (15) is out-of-range for packet (0)

T (10), EV (17), MOD (top.node_4.xmt), KP (op_td_get_int)

33, 编译错误

错误提示:

'rsvp_path_to_uni_msg_map' : 'struct SimT_Packet *(struct SimT_Packet *)' differs in levels of indirection from 'struct SimT_Packet *(struct SimT_Packet *)'

解决方法:

发现 HB 中这个函数被声明了两次, 而且每次声明不同。

错误提示:

'=' : 'struct LSP_TUNNEL_IPv4_FILTER_SPEC *' differs in levels of indirection from 'int '

解决方法:

发现 FB 中相应位置的语句是 `event_info.lsp_tunnel_ipv4_filter_spec_ptr = OPC_FALSE;`
赋值类型不对, 应该是赋值 `OPC_NIL`。

错误提示:

E:/workoffice/opnet/opnet debug/guilin/sour_UNI_C.pr.c(500) : error C2143: syntax error : missing ')' before 'type'

E:/workoffice/opnet/opnet debug/guilin/sour_UNI_C.pr.c(500) : error C2059: syntax error : ')'

E:/workoffice/opnet/opnet debug/guilin/sour_UNI_C.pr.c(501) : error C2143: syntax error : missing ')' before 'type'

E:/workoffice/opnet/opnet debug/guilin/sour_UNI_C.pr.c(501) : error C2059: syntax error : ')'

E:/workoffice/opnet/opnet debug/guilin/sour_UNI_C.pr.c(539) : error C2059: syntax error : '('

E:/workoffice/opnet/opnet debug/guilin/sour_UNI_C.pr.c(545) : error C2059: syntax error : '('

解决方法:

第一个错误 500~501 行, 是由于下面两个函数

```
uni_path_fields_ptr =uni_msg_path_fields_mem_alloc(void);
```

```
common_header_ptr =uni_mem_common_header_alloc(void);
```

其中的参数是 void 类型, 那么不该使用 void, 而是置空。

后面两个错误是

`message_id_ptr->flags_epoch.msg_id_epoch= flags_epoch.msg_id_epoch;`使用不对。

Message_id_ptr 是一个指向 message_id 结构体的指针, 其中有 flags_epoch 组成元素, 而 flags_epoch 又在进程模型中定义为了状态变量。所以这句在 OPNET 中前后两个 flags_epoch

都显示为黄色，表示状态变量。

```
typedef struct {
    Object_header*      msg_id_header;
    struct bit_field {
        u_int32_t  msg_id_flags :8    ;
        u_int32_t  msg_id_epoch :24   ;
    } flags_epoch;                  //use bit field to present the 24 bit field
    u_int32_t      msg_id;
}MESSAGE_ID;
```

解决方法：将 MESSAGE_ID 中的 flags_epoch 该成其他名字。

错误提示：

burst_schedule_edge_lvzhhb.pr.tmp.c

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(264) : warning C4029:
declared formal parameter list different from definition

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(376) : warning C4029:
declared formal parameter list different from definition

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(382) : warning C4020:
'channel_schedule_one_channel' : too many actual parameters

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(532) : warning C4029:
declared formal parameter list different from definition

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1223) : warning C4020:
'channel_schedule_onebyone' : too many actual parameters

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1360) : warning C4020:
'write_back_strm_index' : too many actual parameters

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1584) : error C2143: syntax
error : missing ';' before 'type'

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1594): error C2065:
'_op_fstack_local_info_ptr' : undeclared identifier

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1594) : error C2223: left of
'->_op_last_line_passed' must point to struct/union

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1677) : error C2223: left of
'->_op_last_line_passed' must point to struct/union

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1694) : error C2223: left of
'->_op_last_line_passed' must point to struct/union

E:/workoffice/opnet/process model/burst_schedule_edge_lvzhhb.pr.c(1712) : error C2223: left of
'->_op_last_line_passed' must point to struct/union

错误分析：

declared formal parameter list different from definition 与 too many actual parameters 是同一个错

误，情形是在 **FB** 中定义了函数，但是定义的形式参数的个数与在 **HB** 中声明的函数的形式参数个数不一致，声明时少声明了形式参数，所以在实际使用时会报告太多的实际参数。

syntax error : missing ';' before 'type'

错误分析：这种问题往往是（）不搭配，后来在另外一个模块的入代码中发现少了一个）。

错误提示：

uni_msg_support.ex.tmp.c

E:/workoffice/opnet/opnet debug/guilin/uni_msg_support.ex.c(514) : warning C4715:
'uni_path_msg_size_get' : not all control paths return a value

E:/workoffice/opnet/opnet debug/guilin/uni_msg_support.ex.c(546) : warning C4715:
'uni_resv_msg_size_get' : not all control paths return a value

解决方法：

错误提示不是所有的控制路径都返回了数值。

后检查 514,发现是一个函数的结束},这个函数是必须返回一个数值的,可是 switch 的 default 路径下没有返回任何数值。修改代码，保证函数末尾是一个 **FRET** 命令。

下一个错误同样如此。

错误提示：

Line Block Error

15 schedule_fdl illegal indirection

29 schedule_fdl illegal indirection

以上两个错误是类型不匹配

3 insert_compl '=': left operand must be l-value

这个错误是提示赋值号左边应该是一个值，而不是一个表达式。

错误提示：

Line Block Error

96 encap illegal case

12 encap illegal case

10 wait illegal case

18 wait syntax error : '}'

解决方法：

发现 96 是那个状态中代码的最后一行。

结果发现原来在那个状态的代码中多添了一个}, 可是提示错误却老是最后的末尾。

警告提示:

OPS_buffer_fdl.pr.tmp.c

E:/workoffice/opnet/process model/OPS_buffer_fdl.pr.c(621) : warning C4761: integral size mismatch in argument; conversion supplied

E:/workoffice/opnet/process model/OPS_buffer_fdl.pr.c(653) : warning C4761: integral size mismatch in argument; conversion supplied

解决方法:

问题不大, 这是因为

```
/* obtain the fdl NO. triggering self-interrupt */
```

```
temp = op_intrpt_code()-LEAVE_FDL;//subtracting 100 is to avoid confliction with other self-interrupt
```

```
/* extract packet at head of FDL[num_fdl]; this */
```

```
/* is the packet just leaving this FDL */
```

```
pkptr = op_subq_pk_remove (temp, OPC_QPOS_HEAD);
```

编译时系统将 op_intrpt_code()值从 0 开始验证, 明显, op_subq_pk_remove 中 temp 的取值会越界, 但是只要实际中 temp 保证不小于 0 就可, 故给出的是警告信息而不是错误信息。仿真中如果值越界, 那么就会提示 invalid memory access 错误。

34, 无法识别的源文件类型:

运行仿真提示的错误:

Beginning simulation at 21:14:31 ?????? 16

<<< Recoverable Error >>>

External code file (uni_msg_suport) compilation failed

Source code was unavailable to the compiler

<<< Recoverable Error >>>

External code (uni_msg_support) compilation failed.

Compile it individually for actual errors.

Moving on to next external code file.

<<< Program Abort >>>

Error encountered recompiling models -- unable to proceed

错误分析:

错误信息中明显提示外部代码文件 uni_msg_suport 不可获得。

经检查发现, sour_UNI_C 中声明了外部代码文件 uni_msg_support 和 uni_msg_suport。

而文件夹中的确有 uni_msg_support 的一套文件 (包括.c, .ex.c, 和.i0.ex.o, 注意这里缺少对

目标文件。), 但没有 uni_msg_suport 的相关文件。

Uni_mem_support 有一套文件 (.c, .ex.c, i0.ex.o 和 dev32.i0.ex.obj)。

在 declare external files 中将 uni_msg_suport 取消, 然后刷新文件夹和保存各个模型, 再打开 declare external files 发现, 只有 uni_mem_support 出现了 (因为文件夹下没有 uni_msg_support 的目标文件)。

打开外部代码编辑器, 重新编译 uni_msg_support (编译时产生错误 uni_msg_support.ex.tmp.c
E:/workoffice/opnet/opnet debug/guilin/uni_msg_support.ex.c(514) : warning C4715:
'uni_path_msg_size_get': not all control paths return a value
E:/workoffice/opnet/opnet debug/guilin/uni_msg_support.ex.c(546) : warning C4715:
'uni_resv_msg_size_get': not all control paths return a value), 再查看 declare external files 发现
列出了 uni_msg_support 文件, 文件夹下也有了 uni_msg_support.dev32.i0.ex.obj 文件。

35, 绑定程序出错 (一是没有声明外部文件 (包括系统提供和用户自定义的), 二是重复定义) 的运行仿真提示:

<<< Recoverable Error >>>

Object repository construction failed

due to errors encountered by the binder program (bind_so_msvc)

<<< Program Abort >>>

Error encountered rebuilding repository -- unable to proceed

错误提示:

提示绑定程序遇到错误, 这包括几种错误:

1 就是没有声明外部文件, 一般提示

sour_UNI_C.dev32.i0.pr.obj : error LNK2001: unresolved external symbol
_call_setup_req_timer_expires_sender_notify

解决方法: 在 declare external files 中声明没有包括的文件。

2 是重复定义, 一般提示:

uni_ipal.dev32.i0.pr.obj : warning LNK4006: _dest_uni_c_node_addr already defined in
simple_source_UNI_C.dev32.i0.pr.obj; second definition ignored

解决方法:

(在 tmp 文件夹下的 bind_err_number 会记录哪些文件没有绑定。)

36, Subqueue selection flag (2) is unrecognized

Recoverable Error reported by process (OPS_buffer_fdl) at module

(top.OPS_FDL_BUFFER.FDL)

at T (5.00003), EV (23), MOD (top.OPS_FDL_BUFFER.FDL), KP (op_subq_pk_insert)

Subqueue selection flag (2) is unrecognized

解决方法:

后来调试发现, 队列模块的子队列中的 subqueue 属性的 rows 设定为 1, 也就是表明子队列只有一个, 所以向子队列 2 中插入包当然不对。

队列模块的子队列属性中 rows 可以设定子队列的数目。

37, ex_exec_simset_launch Error: program abort – Invalid

Memory Access

错误分析:

读取仿真属性设置错误, 这包括属性没有提升到仿真属性中, 可在仿真属性中设定了该值; 也包括设置了仿真属性值, 可后来又在网络模型属性中设置了该值。

38, 获得子队列数目的方法:

```
op_ima_obj_attr_get (my_id, "subqueue", &subq_objid);
```

```
subq_nums = op_topo_child_count(subq_objid, OPC_OBJTYPE_SUBQ);
```

39, 编译错误, 由于采用 C 编译器编译, 可进程中使用了 C++ 代码

Hurst_estimation_online.pr.tmp.c

E:\workoffice\opnet\process model\gamma.h(5) : error C2061: syntax error : identifier 'namespace'

E:\workoffice\opnet\process model\gamma.h(5) : error C2059: syntax error : ';'

C:\Program Files\Microsoft Visual Studio\VC98\include\eh.h(32) : fatal error C1189: #error : "eh.h is only for C++!"

解决方法: 删除 Header block 中包括的 C++ 头文件即可。

40, KP 要求队列上下文

<<<< Program Abort >>>>

KP requires queue context

T (1.0001), EV (112), MOD (top.OPS MAN.node_0.ass_q2), KP (op_subq_empty)

<<< Program Abort >>>

Unhandled C++ exception in process model (OPS_ring_switch)
T (1.0001), EV (123), MOD (top.OPS MAN.node_0.buffer/switch)

分析：在处理器模块中进行了队列资源的操作，如 subqueue。

41，未处理异常

<<< Program Abort >>>

* Time: 11:55:21 星期四 三月 17
* Product: modeler
* Program: op_runsim (Version 10.0.A PL1 Build 2291)
* System: Windows NT 5.0 Build 2195
* Package: process (OPS_ring_switch) at module (top.OPS MAN.node_0.buffer/switch)
* Function: OPS_ring_switch [End_Sim enter execs]
* Error: Unhandled C++ exception in process model (OPS_ring_switch)
T (60), EV (260988608), MOD (top.OPS MAN.node_0.buffer/switch)

* Function call stack: (builds down)
[NOTE: Optimized libraries used; call stack is incomplete]
[For detailed call stack, set 'kernel_type' to development]
[and recompile models with 'comp_trace_info' set to TRUE.]

	Call	Block	
	Count	Line#	Function

0)	39237880	1073741928	0x00004c00 [name not available]
1)	39237880	1879048703	0x00004c00 [name not available]
2)	39237880	-805306207	0x0000c400 [name not available]
3)	39237880	78	OPS_ring_switch [End_Sim enter execs]

<<< Program Abort >>>

* Time: 11:55:34 星期四 三月 17
* Product: modeler
* Program: op_runsim (Version 10.0.A PL1 Build 2291)
* System: Windows NT 5.0 Build 2195
* Package: op_runsim
* Function: sim_runsim_exec
* Error:

* Function call stack: (builds down)

	Call	Block	
	Count	Line#	Function

0)	1	513	m3_main
1)	1	320	sim_runsim_exec

分析：从调用过程来看，问题出现在仿真结束中断引起的状态转移时，在状态的入口代码出现问题。

ODB 调试后发现，问题语句是 ofstream 对象的 logfile 命令。这应该属于 C++编程的问题。

42，包使用了未定义的结构：

'optical_pkptr' uses undefined struct 'SimT_Packet'

分析：后检查发现，如此定义包指针不妥（Packet* head_pkptr,optical_pkptr;），正确的定义方法该是分开定义：

Packet* head_pkptr;

Packet* optical_pkptr;

43，如何自定义一个结构数据类型：

c++语言中定义结构变量是 struct 可以省略，c 不可以省。所以在 opnet 的状态变量中定义结构变量时，如果采取 c 编译，那么需要加上 struct。

Beginning simulation of OPS_pktassemble_realpksize-Hurst_Hybrid at 21:20:52
05

44，几种仿真异常

事件数很少！

Simulation Kernel: optimized, sequential

Simulation Completed - Collating Results.

Events: Total (12), Average Speed (0 events/sec.)

Time: Elapsed (0.00 sec.), Simulated (30 sec.)

由于设置的自相似源的 H 很小（0.55），那么业务源突发性很低，在 30 秒的仿真时间内产生的包数很少，那么事件数相应的很少了。

没有最后通信而仿真终止！

Beginning simulation of OPS_pktassemble_realpksize-Hurst_Hybrid at 03:02:26

Simulation Kernel: optimized, sequential

Simulation terminated without final communication.

Check DES log (select 'Open DES Log' in 'DES' menu) for possible error messages.

一种原因是系统繁忙，仿真资源不够，这是正常的，仍然可以进行仿真；另一种原因就是模型本身有问题，导致仿真到某步时无法继续进行。

对象库构造失败

Beginning simulation of OPS_pktassemble_realpksize-Hurst_Length at 03:29:09

Simulation Kernel: optimized, sequential

<<< Recoverable Error >>>

Object repository construction failed

due to errors encountered by the binder program (bind_so_msvc)

这一般是进程模型有问题，比如外部变量没有声明等。

仿真结果显示是 0

判断结果是否的确为 0。

有时结果个别数据非常大，比如无穷大（如果存在结果无穷大时，纵坐标会没有任何数据显示，如没有任何刻度标示），而其它结果又非常小，那么显示时，其它所有数据看起来都是 0。这时可以将结果倒出到文本文件中，看数据是否的确为 0。

45, 仿真无法执行 Line is longer than maximum size (512 characters).

<<< Recoverable Error >>>

* Time: 19:04:44 ĐÇÆÚÒ» ËÄÔÂ 11

* Product: modeler

* Program: modeler (Version 10.0.A PL1 Build 2291)

```

* System:   Windows NT 5.0 Build 2195
* Package:  Vos (Virtual Operating System)
* Function: Vos_File_Read_Line_Lim
* Error:    Line is longer than maximum size (512 characters).
* Function call stack: (builds down)

```

Call Block			
Count	Line#	Function	

0)	1	108	0x00004c00 [name not available]
1)	1	518	0x78cd7e0a [name not available]
2)	1	174	0x6900d606 [name not available]
3)	1	137	vos_event_loop
4)	2290	61	vos_event_client_vevent_handler
5)	75	436	vmmi_ets_action_notify
6)	16	1424	mmi_sysop_error_log_view
7)	18	1377	mmi_sysop_log_view
8)	110314	1887	Vos_File_Read_Line_Lim

原因分析: ?

46，统计量无法收集

分析：场景如下，进行丢包率的收集。结果打开 **view results** 时发现统计量查看禁用。后对进程分析，发现可能是没有进入丢包率计算的代码，所以 **op_stat_write** 写丢包率的语句没有执行，所以整个仿真过程中就从没有写过该统计量。

47，分支语句中的统计量问题

```

if (wvl_idle)
{
    wavelength[wvl_idle_id].busyuntil = current_time + svc_time;

    op_pk_nfd_get(pkptr,"dest_address", &dest_address);
    pk_cnt[dest_address] = pk_cnt[dest_address] + 1;
    pk_size_bit[dest_address] = pk_size_bit[dest_address] + pk_len;

    op_pk_send_delayed (pkptr, TO_OUT_STRM, svc_time); //

```

```

        pk_out_cnt ++;
        pk_out_size = pk_out_size + pk_len;

        /* burst/data rate statistics */
        op_stat_write(burst_rate_lsh, 1);
        op_stat_write(burst_rate_lsh, 0);

        op_stat_write(data_rate_lsh, pk_len);
        op_stat_write(data_rate_lsh, 0);

    }
else
    {
        op_pk_destroy(pkptr);

        pk_drop_cnt ++;
        pk_drop_size = pk_drop_size + pk_len;
    }

```

分析：AB 的位置是不妥当的，比如，假设丢包率很低，那么由于 B 在分支内，有可能很长时间都不会进入 B 的位置进行丢包率的更新计算，那么在写入下一个统计值的时候可能是很长时间之后的事件，这就无法实时的反应统计量的变化情况，会导致计算出来的统计量忽高忽低。所以 AB 都该知道 if 语句分支的外面。

48, undefined; assuming extern returning int

提示信息： /workoffice/opnet/OBS-LVZHB/obs_edge_oxc_00.pr.c(1857) : warning C4013:

'oxc_link_check' undefined; assuming extern returning int

原因分析: oxc_link_check 函数出问题。

初步检查发现, 函数的定义和函数的声明都没有问题, 保持一致。查警告信息提示的地方, 发现是调用函数的语句, 但参数和声明定义的一致, 也没有问题。后来采用 find 搜索 oxc_link_check 字段, 发现在 oxc_link_check 函数定义中存在问题, 其 fin 语句是 FIN (oxc_link_check (link_objid, wavelength, status));明显不对。这是其一错误, 但这不是导致警告消息的错误原因。

后来继续分析提示信息, assuming extern returning int。假设外部函数, 返回 int 类型。也就是说调用函数时系统认为调用了一个没有定义的函数, 可是比较发现, 这个函数明明定义, 而且函数名无错, 那么猜想是不是调用函数时其中的函数名中夹有不可显示的其它字符, 于是把函数定义处的函数名拷贝过来代替原先函数调用处的函数名, 再编译发现无告警信息, 怪哉!!!!

49, warning C4090: 'function' : different 'const' qualifiers

提示信息:

obs_edge_oxc_00.pr.tmp.c

E:\workoffice\opnet\OBS-LVZHB\obs_edge_oxc_00.pr.c(987) : warning C4090: 'function' : different 'const' qualifiers

E:\workoffice\opnet\OBS-LVZHB\obs_edge_oxc_00.pr.c(987) : warning C4028: formal parameter 1 different from declaration

E:\workoffice\opnet\OBS-LVZHB\obs_edge_oxc_00.pr.c(987) : warning C4090: 'function' : different 'const' qualifiers

E:\workoffice\opnet\OBS-LVZHB\obs_edge_oxc_00.pr.c(987) : warning C4028: formal parameter 2 different from declaration

E:\workoffice\opnet\OBS-LVZHB\obs_edge_oxc_00.pr.c(987) : warning C4024: 'prim_prg_list_sort' : different types for formal and actual parameter 2

错误分析:

经过不停分析和网上查找, 将错误定位在形式参数的声明中, 而且是与 const 有关。

后分析 ip 经常中的相关用法, 发现 op_prg_list_sort (list_ptr, oxc_rte_list_sort);函数中的比较程序的形式参数应该是 const 限定符, 也就是在 op_prg_list_sort 中对比较元素是声明为 const, 所以 oxc_rte_list_sort 中也必须声明为 const。

50, 多个版本之间的问题:

错误提示信息: 10.5 下运行

Creating library E:\workoffice\opnet\OBS-LVZHB\obs-trail_01.dev32.i0.nt.lib and object E:\workoffice\opnet\OBS-LVZHB\obs-trail_01.dev32.i0.nt.exp

oms_string_support.dev32.i0.ex.obj : error LNK2001: unresolved external symbol

_This_File_Was_Compiled_With_OPNET_Release_10_5

.....

obs_edge_control_source_gchild.dev32.i0.pr.obj : error LNK2001: unresolved external symbol
_This_File_Was_Compiled_With_OPNET_Release_10_5

解决办法:

根据错误提示信息, 怀疑是多个 opnet 版本发生了揉杂, 系统安装了 8.1, 10.0, 10.5, 而 8.1 和 10.0 运行多时没法发生问题, 那么 8 和 10 版本之间应该发生冲突的可行性较小, 10.0 和 10.5 相近, 很可能存在冲突。因此卸载 10.0 重新在 10.5 下运行仿真, 错误提示信息消失。

51, 包格式定义

如果在定义包格式时数值类型设定的不恰当, 则有可能出现错误。如, qos 域设定为 double 或 float 类型, 即使赋值时写成 op_pk_nfd_set(optical_pkptr, "QoS", 0), 在 qos 域中的值仍然为浮点型, 可能是计算机中最小的正数, 而不是整型的 0。

52, 取消调度

要取笑特定某个调度事件发生中断, 使用 op_intrpt_disable (Prohibits the specified class of interrupts from being delivered to the invoking process, either for an indefinite period, or just for the next occurrence of such an interrupt. 可见似乎是指定的一类调度不要传给进程, 而不是特定的一次或一个) 也许不行。此时可以使用 op_ev_next/op_ev_next_local, op_ev_valid, op_ev_current, op_ev_type, op_ev_code, op_ev_cancel 等结合使用找出需要取笑调度的事件, 然后获得事件句柄, 予以取消。

```
/* Obtain the currently executing event (it must be at this module, */
/* since this module woke up) and the one after it. */
this_event = op_ev_current ();
next_event = op_ev_next_local (this_event);
/* Loop through all of the events scheduled for this module */
/* and cancel any that are retransmission timers. */
while (op_ev_valid (next_event))
{
    if ((op_ev_type (next_event) == OPC_INTRPT_SELF) &&
        (op_ev_code (next_event) == RETRANS_TIMER))
    {
        retrans_timer = next_event;
        next_event = op_ev_next_local (retrans_timer);
        op_ev_cancel (retrans_timer);
    }
    else
        /* Obtain the next event; if there are no more events, op_ev_next_local () */
```



```
/* will return an invalid event handle and the loop will terminate. */
next_event = op_ev_next_local (next_event);
}
```

53, 全局统计量注册不上, ODB 调试下显示 stat <invalid>。

网络论坛:

1, 全局变量

因为 opnet 是基于 c 代码的, 实际上 process 的变量就相当与一个 c 的子文件, 在一个文件中定义的变量应用于另一个文件的方法就是:

```
int a ;( A 文件 HB)
extern int a; (B 文件 HB)
```

疑问:

如果一个进程模型被多个处理器或队列模块使用, 那么其实就是该进程模型的多个实例。如果进程模型中定义了一个全局变量, 而在初始状态中进行了初始化, 试问这个全局变量是否存在多次定义, 和多处初始化。

同理, 如果是全局统计变量, 那么是否会存在全局统计变量的多次定义和多次注册。

2, 统计量结果显示 :

normal 就是你写什么就显示什么出来, 对于你的统计来说, 刚开始的时候

total_num_packet/op_sim_time() 肯定很小, 然后慢慢的趋于平稳。

而对于 bucket 来说, 是选择 sum/time 的模式, 那么它就是在一段时间内将你写入的数加起来再

平均, 就是 sum/time 嘛。

那比如说你发包的速率是 100pks/s 的话, 那么对于 normal 来说肯定是一条上升的曲线, 因为你

不是从仿真一开始就发包的, 开始几乎是 0, 然后慢慢趋于平稳, 到 100pks/s。而对于 bucket, 则它应该是很快就稳定在 100pks/s 了, 因为它是在 bucket 这段时间内做的统计, 而不管前面你

还没有发包的情况。

3，本地统计量和全局统计量的区别

local 统计量是相对于 node 级别的，每个 node 维护一份，如 node A 和 node B 写入的 local 统计量是写入到两个不同的统计量中。

而 global 统计量是全局的，一旦注册好，所有的节点写这个统计量时会写入到一个统计量中，如 node A 和 node B 各写一次某个 global 统计量后，该 global 统计量中会有两个值。

4，从节点名获得节点对象 id：

`op_id_from_name()`。

5，包的合并

传输后还要拆分成原来的包。sar 是先分段后重组回原来的包，我要做的是先合并（合并过程不破坏原有的每一个包的包结构），后拆分出原来的包。

解决方案：

可以把你的数据包的域设定为 Packet 类型，这样就可以把你的一个包封装在另一个包的某个域中，可以在包中设置多个这样的域。

如果你要封装的包的个数不确定，则可以把这些包都存储在一个 list 中，再把这个 list 的指针放在你要传输的包中。

6，网络负载的概念

7，定义外部函数

我想定义一些比较通用的函数，在各个进程中都可以使用，应该怎样做呢？

[用外部文件](#)

包含这些函数的文件可以为.c 文件呢还是必须为.ex.c 文件呢？必须定义一个关联的头文件为这些函数申明原型吗？如果需要头文件，是在 HB 中加入#include 语句吗？

[应该是 ex.c，用头文件，在 HB 中加入#include](#)

我看到进程模型编辑器 file 菜单下有一个子菜单 declare external files，可是怎么也看不到我编辑的外部文件（.ex.c 文件），也没有地方可加。

[你没编译外部文件吧](#)

7 , op_pk_nfd_access 和 op_pk_nfd_get 的区别 :

get 后那个 field 的内容就没了。

Access 后还可以再 get。

get 改变了 packet 那个 field 的所有权，access 没有。access 的执行速度应该更有效

8 , 根进程和子进程的唤醒

子进程和根进程的最大差别是初始状态是何时执行的。根进程是根据 begsim intrpt 来进行的，而子进程的 begsim intrpt 是无效的，可以按照 disabled 来理解。

其余的就没什么不一样了。阻塞后都是要通过中断来触发。只是一般来说，子进程都是由父进程来触发的。如果你没做什么特别处理，那么给某一模块的中断都是由根进程来响应的，然后由根进程来判断该触发哪个子进程来处理该中断。

上面讲的是一般情况。子进程也可以通过注册某一类型的中断来直接响应这类型的中断，而无须先由根进程判断后再触发。具体 KP 看看 doc 上 intrpt

子进程运行到非强制状态后，仿真核心会返回到父进程。

子进程的激活方法包括：父进程使用 op_pro_invoke 函数激活子进程，子进程通过自中断激活自己；子进程直接接收数据包的流中断激活自己等。

9 , 工程保存问题

1、如果你的 Project 是新建的，那么它将会保存在你 mod_dir 属性的第一个目录中，如果是已经存在的，那么你的打开或写入操作将会继续在这个目录中进行。

2、不同目录中如果有相同名称的节点或进程模型，那么 OPNET 将会将 mod_dir 中靠前的目录中的模型视为有效。

其它问题杂记：

1 ,我在退出 OPNET 后 ,再次进入 OPNET 编辑先前的 project 时 , 往往出现一个*.bk*的工程 ,而且

在原来的 project 中多出几个名为 UNRECOVERED*的 scenario, 看工作目录多出几个带~或# 或 bak 的文件。好像是 OPNET 自动备份的。感觉十分不爽。

#这种情况应该非正常退出发生的现象,在重新打开项目时会提醒有文件怀疑是该项目的文件,可以点击恢复,然后进入 scenarios->manage scenarios,删除 recovered 就行了,或者 edit->preferences 中编辑 back interval 属性。

#等文件

有时 recovered 文件被删除后,重新打开工程时仍又出现,此时可以再次删除,然后退出 OPNET,重新启动打开就 OK 了。

2 , 链路、信道、发射机和接收机的输出流和输入流、包流之间的关系。

节点内的一个接收机可以对应节点间多个链路吗?似乎是不能的,接收机/发射机对应链路,接收机/发射机的输入流和输出流(端口)对应信道。

这几者间的层次关系:

链路(link)——两个节点间的物理连接

信道(channel)——链路可以进一步分为多个信道

3 , 统计量包括定义和声明以及收集定义三部分。

1. 如果统计量要在探针模型中出现,则统计量必须在进程的 interface 部分进行声明。
2. local 统计量是相对于 node 级别的,每个 node 维护一份,如 node A 和 node B 写入的 local 统计量是写入到两个不同的统计量中。global 统计量是全局的,一旦注册好,所有的节点写这个统计量时会写入到同一个统计量中,如 node A 和 node B 各写一次某个 global 统计量后,该 global 统计量中会有两个值。
3. 对于节点统计量,则必须进一步在节点域的 interface/node statistics 中声明,以便可以作为节点统计量进行收集。

4. 在统计量定义的进程中，必须对该统计量进行收集定义（编程）。系统也会自己内建很多统计量收集的定义。
5. 探针文件既可以收集标量统计量，也可以收集矢量统计量。
6. 本地统计量和全局统计量既可以是标量统计量，也可以是全局统计量。
7. 如果不需要使用探针模型指定统计量收集工作，则无需在进程模型的本地统计量或全局统计量进行声明。只需要在代码中定义好统计量和计算和收集工作即可。

4，OPNET 和 VC 联合调试设置：

OPNET 和 VC 调试的各自特点：

- 1) ODB 调试功能强大，但主要侧重于逻辑上的调试，可以显示函数的调用情况、状态的转移情况、代码的返回值等，但无法显示一些赋值、比较等代码，无法显示临时变量、状态变量等（除非使用诊断代码显示输出）。
- 2) VC 调试侧重于局部精细地跟踪调试，可以查看变量的变化，或进入函数查看细节。VC 的功能更加强大，可观察变量、单步跟踪、跟踪子程序等。

VC 和 OPNET 联合调试的步骤：

1) 设定环境变量

第一部分是系统环境变量的设置，包括：

在 VC 安装过程中，要注册环境变量

我的电脑/属性/高级/环境变量中设置：

include

Lib

MSDevDir

Path

环境变量中包括两部分：Visual C++环境变量设置和 OPNET 环境变量设置。

如果 Visual C++环境变量设置不正确，则 OPNET 编译时无法利用 VC 编译器，OPNET 的编译是利用 VC 的编译器。如果 OPNET 环境变量设置不对，则无法实现 VC 和 OPNET 的联合调试。

第二部分是 OPNET 环境变量的设置，包括：

EDIT/PREFERENCES：编译器/连接器设置

在 OPNET 编译 C++代码，为了能够正确编译和连接代码，必须正确设置以下参数：

【C 编译器】设置：comp_prog: comp_msvc（指定使用 microsoft VC）

【C++编译器】设置：comp_prog_cpp:comp_msvc (op_mkmsim)或 comp_prog_cpp:comp_msvc (op_runsim)这是设置编译器程序（指定使用 microsoft vc）

【连接器设置】：bind_static_prog:bind_msvc (op_mkmsim)【静态连接器】或 bind_shobj_prog:bind_so_msvc (op_runsim)【动态连接器】这是设置连接器程序（指定使用 microsoft vc）。

注：从上设置是否也发现：op_mkshim 是静态连接，而 op_runsim 是动态连接。

Binder 程序：ECI 翻译器，将对象代码文件连接起来生成一个可执行的程序。通过指定 bind_shobj_prog/bind_static_prog 环境属性可以间接的激活 binder 程序。

EDIT/PREFERENCES：连接器标志设置

1. Bind_shobj_flags(动 态 链 接) : /LIBPATH:D:\Program Files\OPNET\10.0.A\sys\pc_intel_win32\lib /DEBUG

定义动态连接标志，指定在生成共享对象（shared object 动态连接）时传给 binder 程序的标志，这里指定库函数路径；

2. Bind_static_flags (静 态 链 接) : /LIBPATH:D:\Program Files\OPNET\10.0.A\sys\pc_intel_win32\lib /DEBUG

指定在生成静态可执行文件时传给 binder 程序的标志。

/DEBUG 目的：指定在将所有的目标文件(*.obj)连接成为一个静态连接库/动态连接库(*.dll)文件时加入调试信息 (/DEBUG)。

解释：连接的时候需要保留调试信息，所以在 bind_shobj_flags 后面要加上 /DEBUG。

3. Bind_shobj_libs: 指定传给 binder 程序的可选库。

4. Bind_static_libs: 指定传给 binder 程序的可选库。

总结：

动态连接/静态连接分别包括三个参数设置：

bind_shobj_prog/bind_static_prog

bind_shobj_flags/bind_static_flags

bind_shobj_libs/bind_static_libs

EDIT/PREFERENCES：编译器标志设置（编译器标志的设置要与使用的编译器匹配）

Comp_flags_c++_specific: 指定 C++编译器使用的可选编译器标志。这个属性值是针对特定编译器的，必须采用所用编译器可接受的格式。

Comp_flags_c_specific: 指定 C 编译器使用的可选编译器标志。这个属性值是针对特定编译器的，必须采用所用编译器可接受的格式。

Comp_flags_common: 指定所有编译使用的可选编译器标志。这个属性值是针对特定编译器，必须采用所有编译器可接受的格式。注意，这个属性值是指用于全部编译。

Comp_flags_devel: 指定在调试模型下编译代码时使用的可选编译器标志。

/Zi/Od: 表示在编译时产生调试信息，并在调试时关闭编译器的优化功能。

Comp_flags_mt: 指定在平行模式下编译代码时使用的可选编译器标志。

Comp_flags_optim: 指定在优化模式下编译代码时使用的可选编译器标志。

Comp_trace_info: 修正编译后的源代码以包括额外的跟踪信息。

其它可能有用属性:

Debug_mk: 指定 OPNET 程序在标准输出设备中显示它送往操作系统的编译和连接命令, 包括标志 (flags)。如果开启了这个开关, 那么编译后会报警, 其实并不是存在问题, 而只是提醒输出了编译命令信息。

Kernel_type: 指定 op_runsim 仿真时, 连接何种内核。包括四种内核。

Diag_enable: 指定是否在 err_log 文件中检测、显示、记录诊断错误。

诊断错误是指程序中不正确但可以恢复的问题。

Rec_err_suppress: 指定可恢复错误和告警被抑制输出。但仍检测并记录在 err_log 文件中。

解释: 大家都知道, 要利用 VC 来进行调试, 就必须在编译和连接的时候生成调试信息。

编译的时候产生调试信息的参数是 /Z7 或 /Zi, (注意: /Z8 并不是合法的参数)。

调试时还需要关闭编译器的优化功能, 所以还要加上 /Od。

目前还没找到任何资料说上面的两个参数有顺序问题, 我自己试验的结果是没有影响。

VC 参数说明:

/Od

Disable (Debug) Turns off all optimizations in the program and speeds compilation. This option simplifies debugging because it suppresses code movement.

/Z7

C 7.0- Compatible Produces an .OBJ file and an .EXE file containing line numbers and full symbolic debugging information for use with the debugger. The symbolic debugging information includes the names and types of variables, as well as functions and line numbers.

/Zi

Program Database Produces a program database (PDB) that contains type information and symbolic debugging information for use with the debugger. The symbolic debugging information includes the names and types of variables, as well as functions and line numbers.

2) 设定 OPNET 参数

配置仿真属性: configure simulation

设置使用 OPNET debugger

设置强行编译

3) 首次联合调试选择 OPNET 强制编译 (force compile)

4) 绑定 (attach) OPNET 仿真进程;

1. 首先要运行 OPNET simulation, 从而启动 op_runsim 程序, 进入 ODB 界面。
打开 VC, 选择 build/Start debug/attach to process, 从中选择 op_runsim_dev.exe 程序。如果这里没有该进程, 可以使用任务管理器打开进程调试。
2. 然后打开文件, File->Open 到模型目录下的相应进程文件打开。
3. 设置好了断点后, 要在 odb 里头用 ODB 命令运行仿真, 没有运行仿真, VC 中 Debug 菜单里面的子菜单都是灰色。

5) 观察变量。

6) 问题:

把 op_runsim_dev.exe 打开调试, 这时看到的 VC 里面都是一些汇编语言:

原因分析: VC 没有打补丁, 去找个 VC 的 SP5 打上吧

OPNET 8 (和 9) 的设置:

Comp_flags (10.0 中改为 Comp_flags_c_specific)

/W3 -IC:\PROGRA~1\OPNET\8.1.A\models\std\include /Zi /Od

Comp_flags_cpp (10.0 中改为 Comp_flags_c++_specific)

/W3 -IC:\PROGRA~1\OPNET\8.1.A\models\std\include /Zi /Od

注意, 在 10.0 中新增一个属性参数设置: comp_flags_devel, 即用于设置/Zi/Od。

(另外其它的参数还有:

comp_globals_export	FALSE
comp_prog	comp_msvc
comp_prog_cpp	comp_msvc (这两者是指采用何种编译器进行编译)
comp_trace_info	TRUE
)	
bind_shobj_flags	/LIBPATH:C:\PROGRA~1\OPNET\8.1.A\sys\pc_intel_win32\lib
bind_shobj_libs	null
bind_shobj_prog	bind_so_msvc
bind_static_flags	/LIBPATH:C:\PROGRA~1\OPNET\8.1.A\sys\pc_intel_win32\lib
bind_static_libs	null
bind_static_prog	bind_msvc (这是指链接程序)

5 , handle_exception FALSE 程序异常可以由 VC 来调试。

经过 1/2/3/4/5 的设置之后, 就可以进行 ODB 和 VC 的联合调试了, 在 ODB 中调试, 如果出现异常提示对话框, 点取消按钮就可以打开 VC 进行调试。

6 , OPNET 自带的自相似业务模型的应用 :

必须与 IP 或 MAC 捆绑使用, 否则报错。

7 , 一些小概念 :

OPNET 的进程模型采用 FSM, 状态是必须互斥的, 系统在任何时刻只能处于一个状态上。当仿真开始时, 由仿真核心激活所有启用了仿真开始中断的进程。其它的进程则由仿真核心根据中断激活或者由其它进程采用 op_pro_invoke 激活。一个进程再次激活后则从上次阻塞时的状态继续运行。

初始状态是指一个进程在首次被激活时所进入的状态。一般进行进程的初始化工作。

主流中断就是除了仿真初始中断以外的中断。

包如何在节点的输入(对应接收机。输入流)和节点的输出(对应发射机。输出流)间进行交换, 细节由编程人员自己确定, 编程时要根据建立模型的包流关系自己确定流的端口。一个链路对应一个发射机和接收机, 一个信道对应收发机的流(即端口), 每个输入流和输出流在节点内对应一条 packet stream。每个流都有相应的数值索引, 本地有效。

种子的概念: 仿真中要使用随机数, 而产生随机数需要输入初始值, 这就是种子。如果采用同一种子, 那么经过多次仿真后, 产生的随机数会重复出现, 因为一个种子根据算法对应产生一个随机数序列。

地图的设置中最重要的是单位距离。因为这才是可能影响传输的因素。

队列中是不是至少必须有一个子队列????????

在 OPNET 8 中, 外部文件 link_delay 是默认被包含进头文件的, 而在以后版本中, 需要在 DECLARED EXTENAL FILES 进行声明。

状态变量的使用：临时变量作用域在于进程的一次激活。本地统计量在该进程模块的所有进程中。全局统计量在一个网络模型中。有待确认？

一次同时运行多个场景的仿真，Manage Scenarios 中设置收集。

8，获得包流索引

首先 op_topo_assoc 获得相连对象的 ID，然后 op_topo_connect 获得连接器的对象 ID，然后 op_ima_obj_attr_get 获得源流和目的流的索引。