

S60

AnalyzeTool User's Guide

Copyright © 2008, 2009 Nokia Corporation and/or its subsidiaries.

This component and the accompanying materials are made available under the terms of "Eclipse Public License v1.0" which accompanies this distribution and is available at the URL "<http://www.eclipse.org/legal/epl-v10.html>".

Initial Contributors:

Nokia Corporation - initial contribution.

This Nokia Product includes Open Source Software. Certain software included in this Nokia Product is licensed and distributed under licenses containing obligation or permission to provide the source code of such software with the binary / executable form delivery of the said software. The source code is delivered to you in accordance with the referred license terms and conditions in the media attached to this document. The exact license terms, as well as the required copyright and other notices, permissions and acknowledgements are reproduced in and delivered to you as part of the referred source code.

Change history:

| Version | Date | Status | Comments |
|---------|------------|----------|---|
| 13.0 | 20.11.2009 | Approved | Licence changed to EPL. |
| 12.0 | 14.10.2009 | Approved | Sections 1.1, 2, 3.1, 5.3 and 5.5 updated. Section 4, 5.3 and appendix B removed. |
| 11.0 | 12.10.2009 | Approved | Sections 3.1, 4.1, 4.3.1, 4.3.2 and 4.5 updated. |
| 10.0 | 16.04.2009 | Approved | Changes to command line parameters (for information on usage, see Section 4.1): -m parameters changed Emulator support now integrated to other chapters. Used terms unified throughout the document. |
| 9.0 | 16.02.2009 | Approved | Section 5 Emulator support BETA added. |
| 8.0 | 19.12.2008 | Approved | Section 4.1 Compiling source codes updated. Appendix B: Replaced previous license with gSOAP Public License. Editorial corrections. |
| 7.0 | 22.10.2008 | Approved | Sections, 1.1, 1.3, 2, 2.1, 3.1, 3.2, 4.1, 4.2, 4.3.2, 4.4, 4.4.1, 4.4.2, 4.5, 4.6, 4.7, 5, 5.1, 5.2, 5.3, and 5.4 updated. Added a new Chapter 6 Technical clarification. Added a new Appendix B. |
| 6.0 | 18.9.2008 | Approved | Sections, 2.1, 3.1, 4.1, 4.3, and 4.4 updated. |
| 5.0 | 29.8.2008 | Approved | Table in Section 5.5 updated. |
| 4.0 | 15.8.2008 | Approved | Step 2 updated in Section 3.2, and one cell added to the table in Section 5.5. |
| 3.0 | 03.07.2008 | Approved | Added Section 2.1.4 Basic workflow of AnalyzeTool. Section 2 Introduction updated. |
| 2.0 | 12.6.2008 | Approved | Section 4.3.2 Getting the analysis data file updated: The command for deleting data files changed from <i>-d</i> to <i>-delete</i> |
| 1.0 | 02.06.2008 | Approved | |

Table of contents:

| | | |
|---|--------------------------------------|-----------|
| 1. | Document control | 4 |
| 1.1 | References | 4 |
| 1.2 | Documentation conventions | 4 |
| 1.3 | Abbreviations and definitions | 4 |
| 2. | Introduction | 5 |
| 2.1 | Basic workflow of AnalyzeTool | 5 |
| 3. | Installation | 6 |
| 3.1 | Preconditions | 6 |
| 3.2 | Installing the tool | 6 |
| 3.3 | Installing other components | 6 |
| 4. | Data analysis | 7 |
| 4.1 | Analyzing the results | 7 |
| 4.2 | Reading the results | 7 |
| 4.2.1 | Combining results | 10 |
| 4.3 | Data file contents | 11 |
| 4.4 | Testing servers set to critical..... | 11 |
| 4.5 | Troubleshooting AnalyzeTool | 12 |
| 5. | Technical clarification | 13 |
| 5.1 | AnalyzeTool architecture..... | 13 |
| 5.2 | Software to be tested | 13 |
| Appendix A: The Apache Software License, Version 2.0 | | 15 |
| Appendix B: GNU General Public License..... | | 19 |
| Appendix C: Other notices | | 28 |

1. Document control

1.1 References

| | |
|----------------------|---|
| [1] <i>DebugView</i> | http://technet.microsoft.com/en-us/sysinternals/bb896647.aspx |
|----------------------|---|

1.2 Documentation conventions

Italic font is used to emphasis files and directories. Code is written in the `Courier new` font. Argument choices are shown in <> brackets, that is, <choice>.

1.3 Abbreviations and definitions

| | |
|----------|--|
| SIS | Symbian Installation System |
| Test run | A test session that is defined by the process start-end (lifetime) |

2. Introduction

AnalyzeTool is a dynamic test tool that is used for testing Symbian software for memory and resource leaks and pinpointing the leaks from the source code. AnalyzeTool can analyze leaks only in target hardware, also platform simulator is supported.

This user guide explains the essentials of using the tool. Documentation for Carbide.c++ Extension can be found from the Carbide help menu.

AnalyzeTool supports automated testing without need for any user interaction. AnalyzeTool works as a background service in target hardware, so no user input is necessary during the test run. Analysis does not interfere with program run so it can be executed among any other testing activities.

Currently AnalyzeTool supports memory analysis on Symbian OS based devices and emulator by storing all the reserved memory addresses never released. Memory addresses with call stack information are used for identifying the code line of the possible memory issue. AnalyzeTool can also spot unclosed resource handles at process exit. In most of the cases, it can pinpoint the exact code line where the memory issue was introduced.

Note: S60 3.1, S60 3.2, or S60 5.0 phone with R&D certificates is required for the use of AnalyzeTool. A trace connection is required for collecting real time data from target hardware to PC if data is collected in external data gathering mode. AnalyzeTool requires that the process (an executable) to be tested is compiled with AnalyzeTool. In case of a DLL component, the process that uses the DLL needs to be compiled with AnalyzeTool.

The source codes of the application to be tested are needed for compilation but also for pinpointing the leaks after testing. Analysis is currently limited to the user side software instead of kernel software.

2.1 Basic workflow of AnalyzeTool



1. Compile the software to be tested with AnalyzeTool.
2. Install the software and the Symbian components of AnalyzeTool to the target hardware.
3. Collect data to PC or target hardware while tests are run.
4. After the test is run, analyse to spot memory issues.

3. Installation

3.1 Preconditions

Note the following preconditions for using the AnalyzeTool:

- The tool has been tested to work with S60 3.1, S60 3.2 and S60 5.0 & PF52.50 platforms.
- AnalyzeTool requires currently ALL -TCB capabilities which can be granted with the R&D certificates.
- The software project to be tested has to have an MMP file.
- The test process's MMP with target type of EXE needs to be compiled with AnalyzeTool.
- For external data gathering: Trace connection is needed.

3.2 Installing the tool

PDK Tools Overlay installer should be used to install AnalyzeTool to the used environment. After that AnalyzeTool should be build to all used platforms and both build types (udeb/urel) in that environment.

AnalyzeTool is found from `sf\pdt\dynaanaactrlandcptr\memana\analyzetestclient` folder. Building should be done in group folder.

Note: Before building check `symbian_version.hrh` file that it defines correct version for your environment.

3.3 Installing other components

AnalyzeTool uses `dbghelp.dll` as a 3rd party component which is not included in the installation.

`Dbghelp.dll` is used to pinpoint leaks for emulator builds. `Dbghelp.dll` is a Windows component, but the version which is included with Windows is not as good as the latest available. You can download and install the latest Debugging Tools from Microsoft at <http://www.microsoft.com/whdc/devtools/debugging/>. This includes the `dbghelp.dll` file. After installation, you need to either copy the `.dll` to same folder as used for `atool.exe`, or replace the default `dbghelp.dll` from the Windows folder.

4. Data analysis

This chapter describes how to read AnalyzeTool results. Even if the code line can show exactly which allocation was not freed, understanding the leaks is the most complex task of the analysing process and usually requires both code level and design understanding.

4.1 Analyzing the results

When results show leak information, every leak contains one or several call stack items that are shown as below. The topmost item represents the topmost item from call stack when allocation was done.

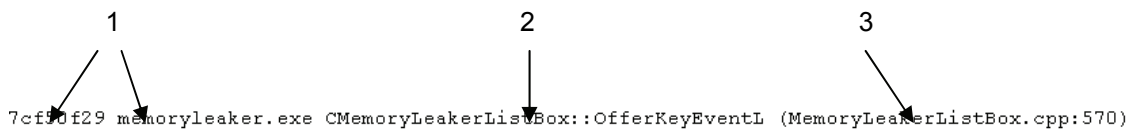


Figure 1 One row from the analysis output

1. A call stack item always contains the memory address. The memory address is for where the code is and not the address of the reserved memory. A module name is displayed if the memory address is in the module’s memory address space, otherwise “???” is displayed.
2. The function name is displayed if the function is identified, otherwise “???” is displayed. The class name is also displayed if the function belongs to a class.
3. A CPP file name is displayed if the source code is available. If the address can be matched to an exact code line, the line number is shown. The code line number is “???” if the memory address is within function memory area but does not point to any specific code line.

The following shows an example output:

```
Memory leak 1 (20 bytes) (0x618c48) Wed Mar 18 14:14:41.839 2009 eikcoct1.dll
794381ab atoolmemoryleakerdll.dll CMemoryLeakerDll::NewLC (memoryleakerdll.cpp:32)
794381cb atoolmemoryleakerdll.dll CMemoryLeakerDll::NewL (memoryleakerdll.cpp:41)
78b5ce57 memoryleaker.exe CMemoryLeakerListBox::ConstructL (memoryleakerlistbox.cpp:503)
78b5ce95 memoryleaker.exe CMemoryLeakerListBox::NewLC (memoryleakerlistbox.cpp:473)
78b5ceab memoryleaker.exe CMemoryLeakerListBox::NewL (memoryleakerlistbox.cpp:448)
78b5c2a9 memoryleaker.exe CMemoryLeakerListBoxView::DoActivateL (memoryleakerlistboxview.cpp:165)
78b5c27f memoryleaker.exe CMemoryLeakerListBoxView::DoActivateL (memoryleakerlistboxview.cpp:153)
78b5daad memoryleaker.exe E32Main (memoryleakerapplication.cpp:66)
```

4.2 Reading the results

The next step is to go through the AnalyzeTool reports. This can be done by checking one memory leak result at a time. Note that leaks may be related to the same memory management issue which is shown as several leaks in the AnalyzeTool results.

Example 1

Below is one line of analysis output that is pinpointed to a correct .cpp line:

7cf50f29 memoryleaker.exe CMemoryLeakerListBox::OfferKeyEventL (MemoryLeakerListBox.cpp:570)

Figure 2 One call stack item from the analysis output

```
554 | if ( aKeyEvent.iScanCode == 167 && aType == EEventKeyDown )
555 | {
556 |     switch( iListBox->CurrentItemIndex() )
557 |     {
558 |     case 0:
559 |     {
560 |         iMemoryLeakerDll->LeakLoop( 1 );
561 |         break;
562 |     }
563 |     case 1:
564 |     {
565 |         iMemoryLeakerDll->LeakLoop( 10 );
566 |         break;
567 |     }
568 |     case 2:
569 |     {
570 |         CEikDialog* dummyDialog = new( ELeave ) CEikDialog();
571 |         break;
572 |     }
573 |     case 3:
574 |     {
575 |         iMemoryLeakerDll->LeakTwoFunctions();
576 |         break;

```

Memory allocation without reallocation

Figure 3 Content of a CPP file

Example 2

In the following example, AnalyzeTool locates a leak in a DLL. The EXE creates an object from the DLL but does not destroy it. The correction to this memory leak must be made in the EXE. In Figure 4 below, analysis is made in DLL group folder and addresses for the EXE are not analysed. If you want to display source lines from the EXE, analysis must be made from the EXE group folder (output is shown in Figure 5 below).

```

Memory leak 1 Tue Jan 06 14:08:27.678 2008 atoolmemoryleakerdll.dll
7d1041bb atoolmemoryleakerdll.dll CMemoryLeakerDll::NewLC (MemoryLeakerDll.cpp:22)
7d1041db atoolmemoryleakerdll.dll CMemoryLeakerDll::NewL (MemoryLeakerDll.cpp:31)
7cf28de1 memoryleaker.exe
7cf28e0f memoryleaker.exe
7cf28e25 memoryleaker.exe
7cf282bf memoryleaker.exe
7cf2827b memoryleaker.exe
7cf29de8 memoryleaker.exe
7cf28000 memoryleaker.exe
7cf28000 memoryleaker.exe
7cf28000 memoryleaker.exe
7cf28000 memoryleaker.exe
7cf29351 memoryleaker.exe
7cf293b5 memoryleaker.exe
7cf2ad44 memoryleaker.exe
7cf29351 memoryleaker.exe
7cf296a0 memoryleaker.exe
7cf28000 memoryleaker.exe
7cf29571 memoryleaker.exe
7cf28000 memoryleaker.exe
7cf28000 memoryleaker.exe

20 EXPORT_C CMemoryLeakerDll* CMemoryLeakerDll::NewLC()
21 {
22     CMemoryLeakerDll* self = new (ELeave) CMemoryLeakerDll;
23     CleanupStack::PushL(self);
24     self->ConstructL();
25     return self;
26 }
    
```

Figure 4 Analysis output and code line in DLL

From the analysis run on the EXE part of the program, you can find the code line where the object was created, and you can see from this data that the program did not delete the object as it should have.

```
Memory leak 1 Tue Jan 08 14:08:27.678 2008 atoolmemoryleakerdll.dll
7d1041bb atoolmemoryleakerdll.dll
7d1041db atoolmemoryleakerdll.dll
7cf28de1 memoryleaker.exe CMemoryLeakerListBox::ConstructL (MemoryLeakerListBox.cpp:478)
7cf28e0f memoryleaker.exe CMemoryLeakerListBox::NewLC (MemoryLeakerListBox.cpp:448)
7cf28e25 memoryleaker.exe CMemoryLeakerListBox::NewL (MemoryLeakerListBox.cpp:423)
7cf282bf memoryleaker.exe CMemoryLeakerListBoxView::DoActivateL (MemoryLeakerListBoxView.cpp:156)
7cf2827b memoryleaker.exe CMemoryLeakerListBoxView::SetupStatusPanel (MemoryLeakerListBoxView.cpp:144)
7cf29de8 memoryleaker.exe ???
7cf28000 memoryleaker.exe _E32Startup
7cf28000 memoryleaker.exe _E32Startup
7cf28000 memoryleaker.exe _E32Startup
7cf29351 memoryleaker.exe ???
7cf293b5 memoryleaker.exe E32Main (MemoryLeakerApplication.cpp:57)
7cf2ad44 memoryleaker.exe ???
7cf29351 memoryleaker.exe ???
7cf296a0 memoryleaker.exe CallThrdProcEntry
7cf28000 memoryleaker.exe _E32Startup
7cf29571 memoryleaker.exe RunThread
7cf28000 memoryleaker.exe _E32Startup
7cf28000 memoryleaker.exe _E32Startup

458 void CMemoryLeakerListBox::ConstructL(
459     const TRect& aRect,
460     const CCoeControl* aParent,
461     MEikCommandObserver* aCommandObserver )
462 {
463     if ( aParent == NULL )
464     {
465         CreateWindowL();
466     }
467     else
468     {
469         SetContainerWindowL( *aParent );
470     }
471     iFocusControl = NULL;
472     iCommandObserver = aCommandObserver;
473     InitializeControlsL();
474     SetRect( aRect );
475     ActivateL();
476     // [[[ begin generated region: do not modify [Post-ActivateL initializations]
477     // ]]] end generated region [Post-ActivateL initializations]
478     iMemoryLeakerDll = CMemoryLeakerDll::NewL();
479 }
```

Figure 5 Analysis output and code line in EXE

4.2.1 Combining results

If results consist of several different modules (exe, DLL and static libraries) it is often best to combine these to one project. As an example here's an actual use case example.

The application Landmarks (*landmarks.exe*) was tested for memory leaks. The project was build using `ls60\app\location\LandmarksUi\group\bld.inf`. When Landmarks application was opened and one new manually entered landmark was created the application was exited. AnalyzeTool results show 2 or more memory leaks. With this kind of test only *landmarks.exe* code lines are pinpointed and other components shown only as module names. Callstack contained modules *ServiceHandler.dll* and *mnaiwprovider.dll* so new *bld.inf* was made to include those also under same project. The new *bld.inf* looked like this:

```
#include "../app/location/LandmarksUi/group/bld.inf"
#include "../mw/classicui/applicationinterworkingfw/ServiceHandler/group/bld.inf"
#include "../mw/location/services/locationmapnavfw/aiwprovider/group/bld.inf"
```

After building this project with AnalyzeTool and running analysis again the result pinpoint call stack items to those two mentioned DLL source codes also.

4.3 Data file contents

The data file in internal data gathering mode is saved in ascii file format. It is not meant to be read as is but used as an input file for analysis. However, you can open this data file in a file editor and read it to help solve problematic AnalyzeTool test cases.

Data file contains the following information:

- **DATA_FILE_VERSION:** identifies the used AnalyzeTool data file version
- **Process start:** process name, process id and start time.
- **DLL load:** identifies the DLL that is loaded from the current module memory address area.
- **DLL unload:** identifies the DLL that is unloaded from the current module memory address area.
- **MEM_LEAK:** each MEM_LEAK row contains the following information: leak time, memory allocation size, and call stack memory addresses.
- **HANDLE_LEAK:** contains the number of unclosed handles.
- **Process end:** process id and end time.

See Figure 6 below for an example of the data file contents.

```
DATA_FILE_VERSION: 5
PROCESS_START: MemoryLeaker.exe[edf5a8b2]0001 142 443279789e819
DLL_LOAD: AToolMemoryLeakerDll.dll 7cf24000 7cf248a4
DLL_LOAD: MemoryLeaker.exe 7cf50000 7cf536e8
DLL_LOAD: 10281fc6.dll(000a0000)[10281fc6] 7cfa4000 7d02955c
DLL_LOAD: AToolMemoryLeakerDll2.dll(000a0000)[04463b81] 7d2ac000 7d2ac1dc
DLL_UNLOAD: AToolMemoryLeakerDll2.dll(000a0000)[04463b81] 7d2ac000 7d2ac1dc
MEM_LEAK: 4432797a21687 14 7cf241bb 7cf241db 7cf50de1 7cf50e0f 7cf50e25 7cf502bf 7cf5027b 7cf51de8 7cf50000 7cf50000
MEM_LEAK: 4432797a21a6f 22 7cf241a3 7cf241d1 7cf241db 7cf50de1 7cf50e0f 7cf50e25 7cf502bf 7cf5027b 7cf51de8 7cf50000
MEM_LEAK: 4432797b0f625 22 7cf2423d 7cf24588 7cf50f13 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b97570 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b97958 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b97d40 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b98128 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b98510 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b988f8 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b98ce0 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b990c8 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b994b0 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797b99898 22 7cf2423d 7cf24588 7cf50f1f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351
MEM_LEAK: 4432797c0fbb5 9c 7cf50f29 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351 7cf516a0 7cf50000
MEM_LEAK: 4432797c0ff9d 10 7cf50f2f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351 7cf516a0 7cf50000
MEM_LEAK: 4432797c10385 3c 7cf50f2f 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351 7cf516a0 7cf50000
MEM_LEAK: 4432797c86d78 22 7cf24285 7cf246dc 7cf242a7 7cf245d0 7cf50f3d 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5
MEM_LEAK: 4432797c87160 22 7cf24263 7cf2472c 7cf242ad 7cf245d0 7cf50f3d 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5
MEM_LEAK: 4432797d0323d 22 7cf24628 7cf240c1 7cf24500 7cf24628 7cf242ed 7cf24628 7cf50f47 7cf50000 7cf50000 7cf50000
MEM_LEAK: 4432797e97392 13 7d2ac07b 7d2ac0a9 7d2ac0ad 7d2ac0b3 7cf50add 7cf52b00 7cf52a4c 7cf50565 7cf50609 7cf507bf
MEM_LEAK: 4432797f0091e 18 7cf50981 7cf52860 7cf50495 7cf50529 7cf50635 7cf50617 7cf51698 7cf50617 7cf51571 7cf50000
MEM_LEAK: 4432797f99286 e 7cf508c1 7cf52668 7cf503c7 7cf50459 7cf506d1 7cf506b3 7cf51698 7cf506b3 7cf51571 7cf50000
MEM_LEAK: 443279801ca87 e 7cf51079 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351 7cf516a0 7cf50000
MEM_LEAK: 44327980c3297 e 7cf51081 7cf50000 7cf50000 7cf50000 7cf51351 7cf513b5 7cf52d44 7cf51351 7cf516a0 7cf50000
PROCESS_END: 142 4432798241556
```

Figure 6 Example of data file

4.4 Testing servers set to critical

AnalyzeTool overwrites functions `TInt User::ProcessCritical(TCritical aCritical)` and `TInt User::SetProcessCritical(TCritical aCritical)` when compiled with AnalyzeTool because setting process or thread to critical needs to be disabled. This is done to ensure that the process or thread can terminate without restarting the phone. Proper termination of processes is needed to allow AnalyzeTool to end analysis. If the process or thread terminates unexpectedly, the tool cannot get the information on which reservations are not freed at the closure of the process.

4.5 Troubleshooting AnalyzeTool

| Symptom: | How to correct it: |
|---|--|
| Running tests do not create any data. | Check that you have installed the correct version of AnalyzeTool or the software to be tested to the phone. |
| Build error: " make[1]: *** No rule to make target <code>`lepoc32\release\ARMV5\udeb\AToolStaticLib.lib'</code> " | Your build environment is missing the necessary files for building with AnalyzeTool. Build AnalyzeTool from sources in <code>...\analyzetoollclient\group\</code> folder to all used platforms and both build types (udeb/urel). |

5. Technical clarification

This chapter explains in technical details how AnalyzeTool works to collect data which is used for analysing memory leak problems. The purpose of this chapter is to give details to interested parties, but reading this is not necessary for using AnalyzeTool successfully. However, reading this chapter is recommended to people who are using AnalyzeTool in very complex SW projects.

5.1 AnalyzeTool architecture

The figure below shows how AnalyzeTool works between a software process to be tested and a Symbian memory allocator. The memory allocator is a component which is automatically used in a process to allocate and free memory during the process life cycle. When a project is built with AnalyzeTool, the software operation is altered so that it will use the AnalyzeTool version of memory allocator. AnalyzeTool therefore hooks all memory operations. The process under test does not see any change in memory allocation since AnalyzeTool passes allocations and frees memory to the Symbian allocator, and sends responses from the Symbian allocator back to the software to be tested. When in use, AnalyzeTool memory allocator sends the information of the memory operations to a so called storage server which collects the data to be analysed.

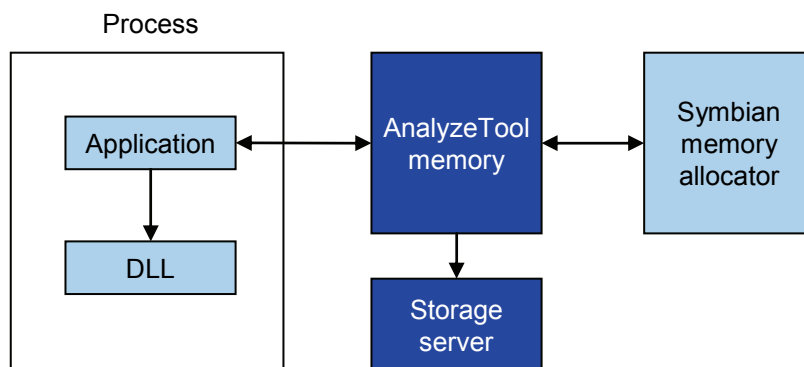


Figure 7 AnalyzeTool architecture

In the internal data gathering mode, the storage server stores the collected information and performs an analysis when the test process is closed, and in the end creates a data file of the test results.

In the external data gathering mode, the storage server passes all the information it gets from the AnalyzeTool memory allocator to the trace connection. This data is later analysed for memory leaks with PC-side software.

5.2 Software to be tested

When the software to be tested has been built with AnalyzeTool, it basically does two things.

Firstly, AnalyzeTool (*atool.exe*) alters the build so that instead of using the Symbian memory allocator it will use the AnalyzeTool memory allocator for the software to be tested. This change does not alter the source code of the software to be tested, but adds the AnalyzeTool static libraries to the project's MMP if its target type is an executable. The changes AnalyzeTool makes to the MMP file for build process are reverted when the build is complete.

Secondly, AnalyzeTool generates *.lst* and MAP files for the software to be tested into the *atool_temp* directory which is located in the directory where the *bld.inf* or project's MMP file is located. These files are needed for locating the memory allocation related to the reported memory leak. Without these files, AnalyzeTool is only able to tell the name of the module (for example *myexample.dll*) where the memory allocation occurred. If they are available, AnalyzeTool can in

most cases tell the exact code line of the memory allocation and call stack items leading to that allocation.

For AnalyzeTool to be able to catch memory operations, it is necessary to build the project which creates the process, meaning that the target type is an executable. Other types of software components, such as DLLs, are only necessary to build if the exact code line of a memory allocation issue needs to be located. DLLs can be built even after the AnalyzeTool results are received from a test run. Note that if the project under testing is leaking memory because of a defect in a system DLL, AnalyzeTool can locate the memory allocation also to that DLL file.

Appendix A: The Apache Software License, Version 2.0

This product includes software developed by The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation, <http://www.ibm.com>

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Appendix B: GNU General Public License

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program – to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how

the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
```

This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<http://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <<http://www.gnu.org/philosophy/why-not-lgpl.html>>.

Appendix C: Other notices

addr2line notice

Copyright (c) 1987 Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.