Chapter
9

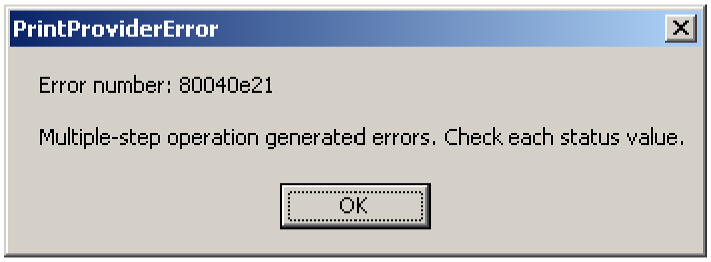
Chapter 9: Troubleshooting

The following chapter is a compendium of error and bug fixes that have been encountered during the creation of Sacman. They are referenced here to aid the programmer who may wish to further develop this project.

Print Provider Errors

Problem

If you encounter the following type of error,



```
PrintProviderError
Error number: 80040e21
Multiple-step operation generated errors. Check each status value.
```

Figure 4

the problem may be that the custom record-set you are using (usually a class with the name of the form RadoRs<TableName>) is not binding properly with the associated database table.

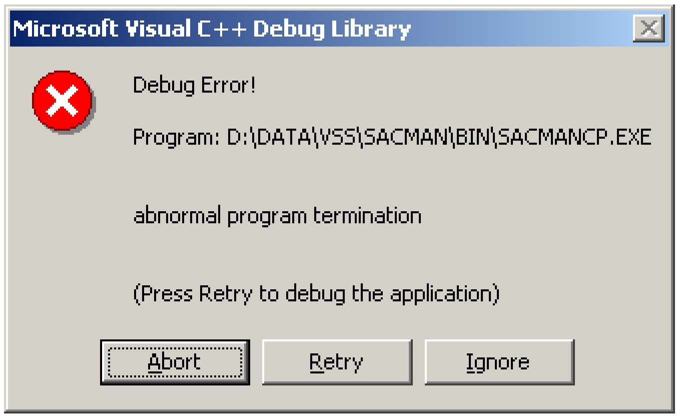
Solution

The solution usually involves changing the fields in the database table or changing the class declaration of the custom record-set class. Either way, the code and the database table must match.

Startup Crashes

Problem

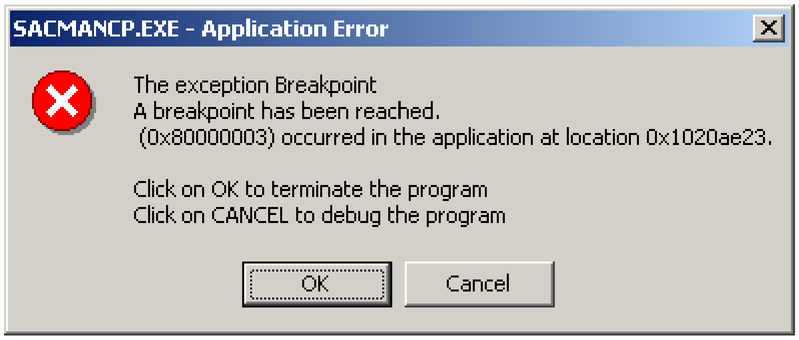
As regards the MSIDD1 computer. I removed all of the previous contents of the d:\data\vss\sacman directory and then "got the latest version" from the Source Safe. Everything compiled and was able to run the program. However, when I tried to startup, I got the following error:



```
Debug Error!

Program: D:\DATA\VSS\SACMAN\BIN\SACMANCP.EXE


abnormal program termination


(Press Retry to debug the application)
```

Start Up Crash Error Message 1

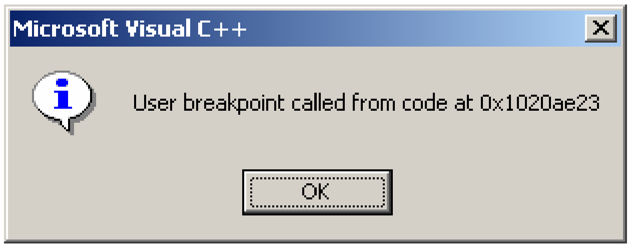
Clicking on Retry lead to the following error message:



```
The exception Breakpoint
A breakpoint has been reached.
 (0x80000003) occurred in the application at location 0x1020ae23.

Click on OK to terminate the program
Click on CANCEL to debug the program
```

Start Up Crash Error Message 2

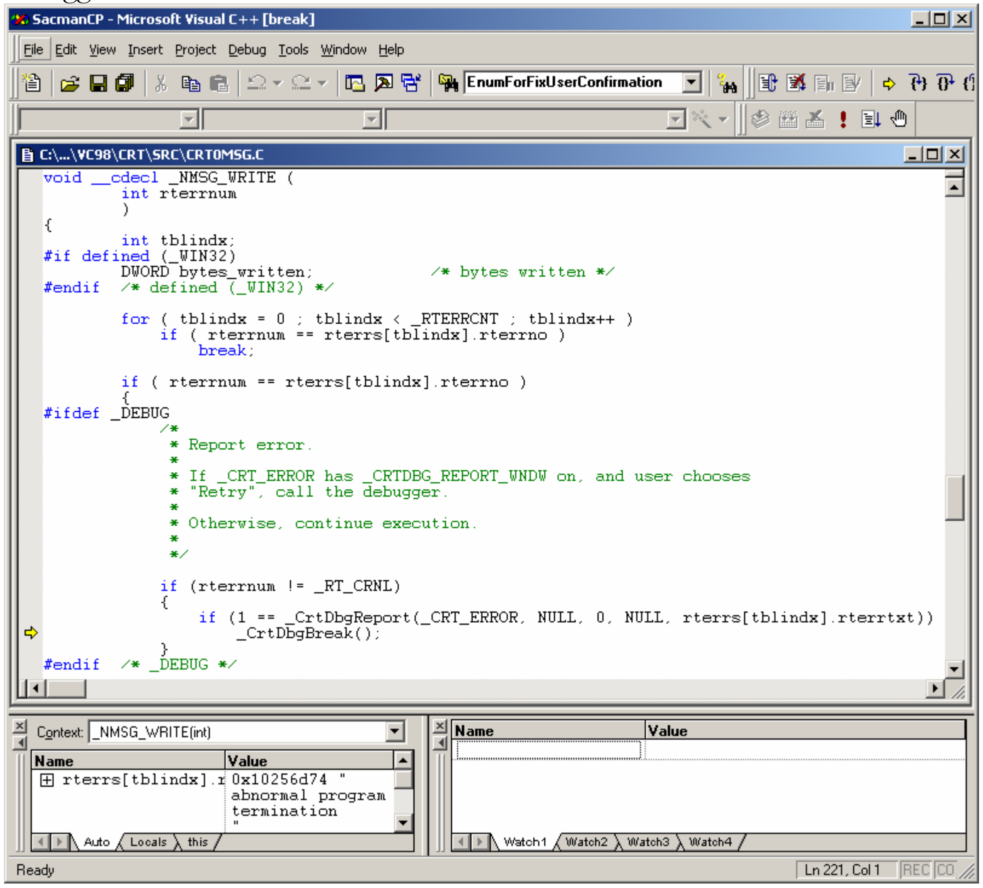
and then pressing Cancel resulted in the following error message:



User breakpoint called from code at 0x1020ae23

Start Up Crash Error Message 3

whereupon pressing the OK button I was shown the following in the debugger:


SacmanCP - Microsoft Visual C++ [break]
File Edit View Insert Project Debug Tools Window Help
EnumForFixUserConfirmation
C:\...\VC98\CRT\SRC\CRT0MSG.C
Auto Locals this
Watch1 Watch2 Watch3 Watch4
Ready
Ln 221, Col 1
REC COL

```
void __cdecl _NMSG_WRITE (
        int rterrnum
        )
{
        int tblindx;
#if defined (_WIN32)
        DWORD bytes_written;            /* bytes written */
#endif  /* defined (_WIN32) */

        for ( tblindx = 0 ; tblindx < _RTERRCNT ; tblindx++ )
            if ( rterrnum == rterrs[tblindx].rterrno )
                break;

        if ( rterrnum == rterrs[tblindx].rterrno )
        {
#ifdef _DEBUG
            /*
             * Report error.
             *
             * If _CRT_ERROR has _CRTDBG_REPORT_WNDW on, and user chooses
             * "Retry", call the debugger.
             *
             * Otherwise, continue execution.
             *
             */

            if (rterrnum != _RT_CRNL)
            {
                if (1 == _CrtDbgReport(_CRT_ERROR, NULL, 0, NULL, rterrs[tblindx].rterrtxt))
                    _CrtDbgBreak();
            }
#endif  /* _DEBUG */
```

Context: _NMSG_WRITE(int)

Name	Value
⊞ rterrs[tblindx].r	0x10256d74 " abnormal program termination "

Name	Value

Start Up Crash Error Message 4

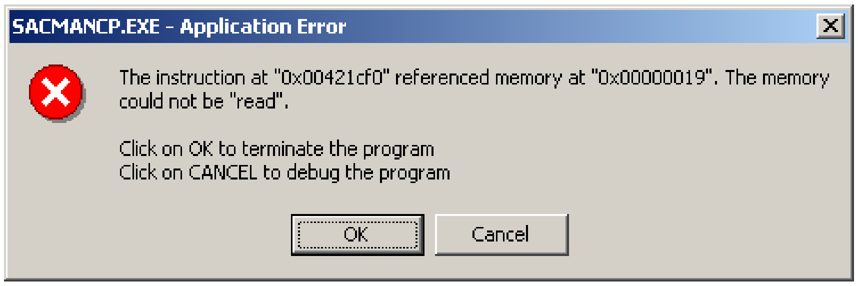
Solution

As it turns out, this bug seems to be the result of not having the proper System DSN setup in the ODBC Administrator. Now that we have multiple databases, we must be set each up properly. This ultimately should be done by an InstallShield application.

Tree/ Start up Bug

Problem:

Previously, when trying to "start up" the SacmanCP program, the following error -message would appear.



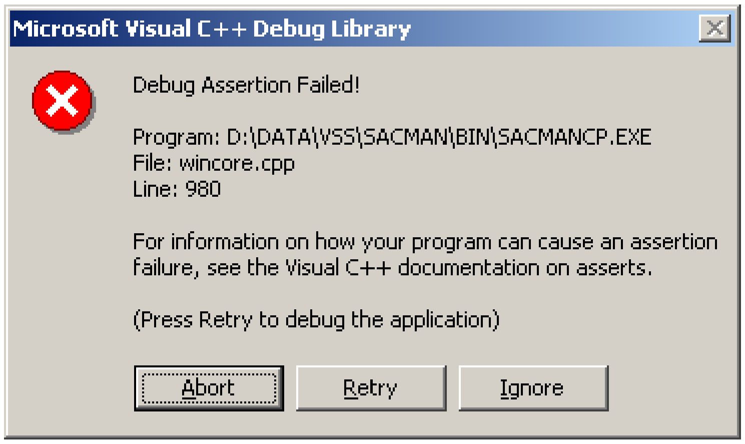
SACMANCP.EXE - Application Error

The instruction at "0x00421cf0" referenced memory at "0x00000019". The memory could not be "read".

Click on OK to terminate the program
Click on CANCEL to debug the program

Tree/Start Up Bug Error Message 1

Pressing OK would then lead to the following information.



Microsoft Visual C++ Debug Library

Debug Assertion Failed!

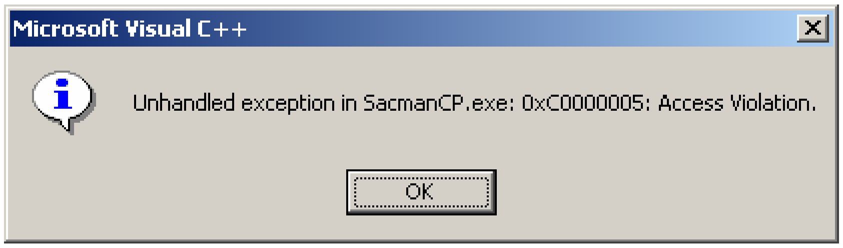
Program: D:\DATA\VSS\SACMAN\BIN\SACMANCP.EXE
File: wincore.cpp
Line: 980

For information on how your program can cause an assertion failure, see the Visual C++ documentation on asserts.

(Press Retry to debug the application)

Tree/Start Up Bug Error Message 2

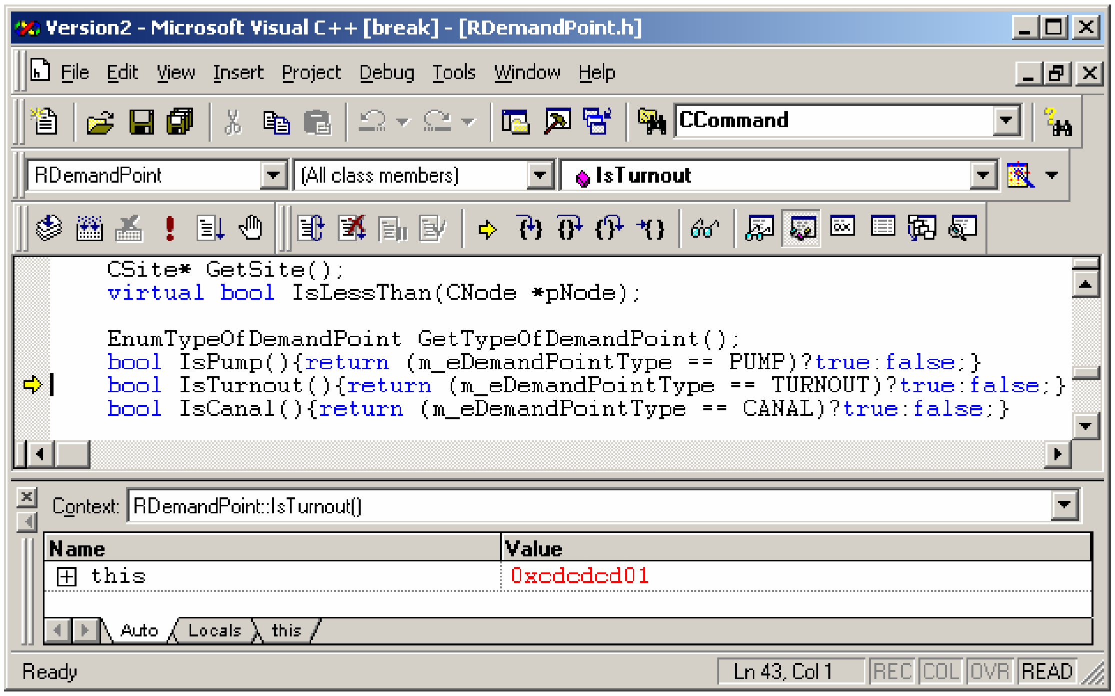
When debugging the program (i.e., running the program via the IDE debugger), the following message appeared after attempting to "start up" the SacmanCP program.



```
Unhandled exception in SacmanCP.exe: 0xC0000005: Access Violation.
```

Tree/Start Up Bug Error Message 3

After clicking the OK button, the debugger then identified the following line of code as the problem (note the line of code referred to by the yellow arrow in the following screen capture).


Version2 - Microsoft Visual C++ [break] - [RDemandPoint.h]
File Edit View Insert Project Debug Tools Window Help
CCommand
RDemandPoint
(All class members)
IsTurnout
Context: RDemandPoint::IsTurnout()
Auto
Locals
this
Ready
Ln 43, Col 1
REC COL OVR READ

```
    CSite* GetSite();
    virtual bool IsLessThan(CNode *pNode);

    EnumTypeOfDemandPoint GetTypeOfDemandPoint();
    bool IsPump(){return (m_eDemandPointType == PUMP)?true:false;}
    bool IsTurnout(){return (m_eDemandPointType == TURNOUT)?true:false;}
    bool IsCanal(){return (m_eDemandPointType == CANAL)?true:false;}
```

Name	Value
⊞ this	0xcdcdcd01

Tree/Start Up Bug Error Message 4

At this point, it seemed apparent that the run-time error was a result of using an invalid pointer (specifically, an invalid pointer to an RDemandPoint object). This conclusion was supported by the evident simplicity of the IsTurnout function – there would seem to be no way for this function to fail so long as the data member m_eDemandPointType was properly allocated.

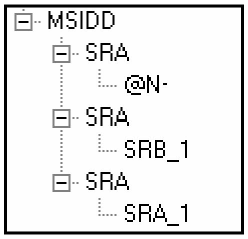
The peculiar nature of this problem was that the data structures appeared to be correctly built and correctly represented by the SacmanCP dialog's tree. But, after destroying the data structures and redisplaying the tree, the program crashed. So was it a problem with the tree or with the data structures? Using the Sherlock Holmes method (after ruling out the impossible, whatever is left – however improbable – must be the truth) of commenting out code and re-running until it worked, coupled with some experimentation (specifically, the apparently offending function was called 3 times in succession in order to more rapidly provoke the crash) led to some odd program behavior that proved to be very indicative of the problem's source. Specifically, when the following code (in the CSacmanCPDlg::OnInitDialog function) was executed:

```
this->m_canMan.CreateNewSystem();
this->m_canMan.CreateNewSystem();
this->m_canMan.CreateNewSystem();
this->UpdateDisplayTree();
```

and the function call was commented out (in the CSacmanCPDlg::AddCanals function):

```
// this->AddSupplyAndDemandPoints(hSubItemParent,pSite);
```

then the following tree resulted:



```
⊟ MSIDD
  ⊟ SRA
     └ @N-
  ⊟ SRA
     └ SRB_1
  ⊟ SRA
     └ SRA_1
```

Sacman CP Tree Figure 1.

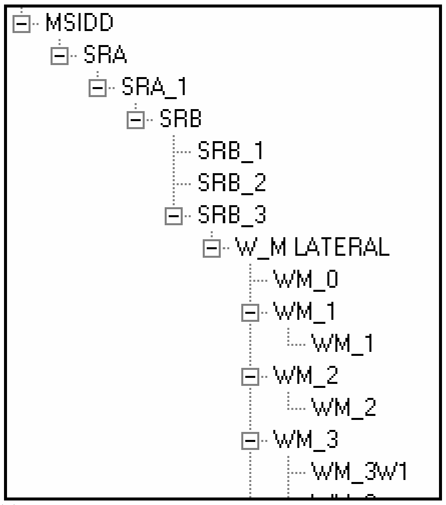
Solution

The oddness of this behavior at first seemed to point at a failure in the logic used for generating the tree. But after further consideration, it became apparent that the error was not in the traversal of the tree (after all, so long as the data structures were created only once, the tree did properly display the data structures as expected). Instead, the problem seemed to be in the destruction of the data structures. Looking further at the tree above, it can be seen that only the last sub-tree correctly shows the relationship that the SRA canal has with its child SRA_1 site. The second sub-tree is incorrectly associating SRB_1 with SRA, while the first sub-tree suggests that there was a corruption of the string used to represent the child site of SRA. Furthermore, why was there three occurrences of the SRA canal in the MSIDD system's list of canals? What was causing this redundancy? The fact that there were 3 calls made to the CreateNewSystem() seemed quite related to the occurrence of 3 SRA canals. In fact, it proved to be the spark that ignited an important question. Were we properly disposing of the list of canals during the destruction phase of the CreateNewSystem function? The answer was, NO!

By adding a call to destroy the list of canals (see code below and note the comment //~bvs~20020719: bug fix!),

```
int CSystem::Destroy()
{
        // Declare and initialize local variables.
        int destroyed = 0;

        // Destroy each of the system's lists and indicate to the
        // caller the success of the function by returning a one
        // to the caller.
        destroyed += this->m_listOfSegments.Destroy();
        destroyed += this->m_listOfSites.Destroy();
        destroyed += this->m_listOfCanals.Destroy(); //~bvs~20020719: bug
fix!

        //~bvs~20020402
        // Destroy whatever is in the Fix Connection member.
        destroyed += this->m_fixConnection.Destroy();

        return destroyed;

        //~maa~20011029
}
```

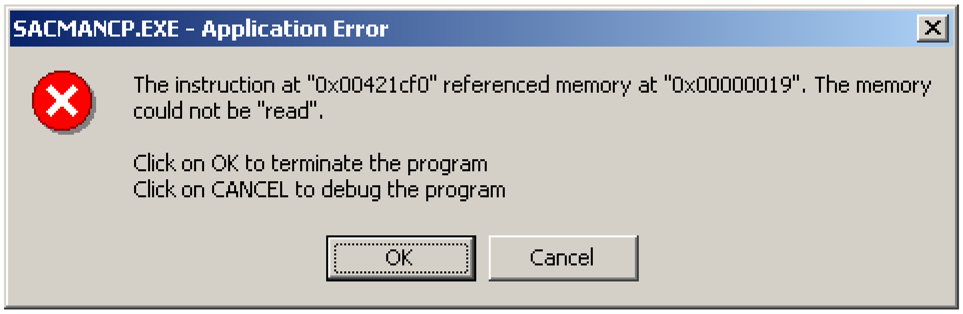
and restoring the OnInitDialog and AddCanals functions, the program worked perfectly, and reflected the following tree (regardless of how many times the data structures were built and destroyed):



Sacman CP Tree Figure 2.

Summary

When encountering an error of the form,



SACMANCP.EXE - Application Error

The instruction at "0x00421cf0" referenced memory at "0x00000019". The memory could not be "read".

Click on OK to terminate the program
Click on CANCEL to debug the program

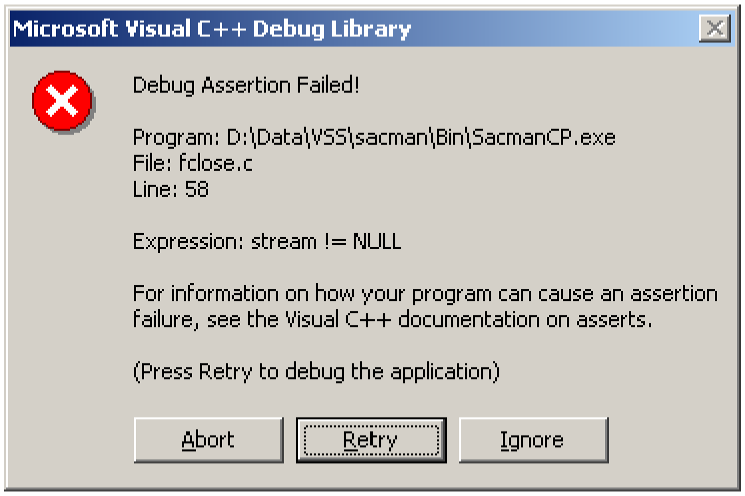
Tree/Start Up Bug Error Message 5

it's probably caused by an invalid pointer (either an un-initialized pointer, or pointer that points to an object that has been destroyed). In the case of this bug, the problem occurred when trying to access a CSite object that was correctly destroyed from an RCanal object that was incorrectly not destroyed.

Sacman Start up Bug

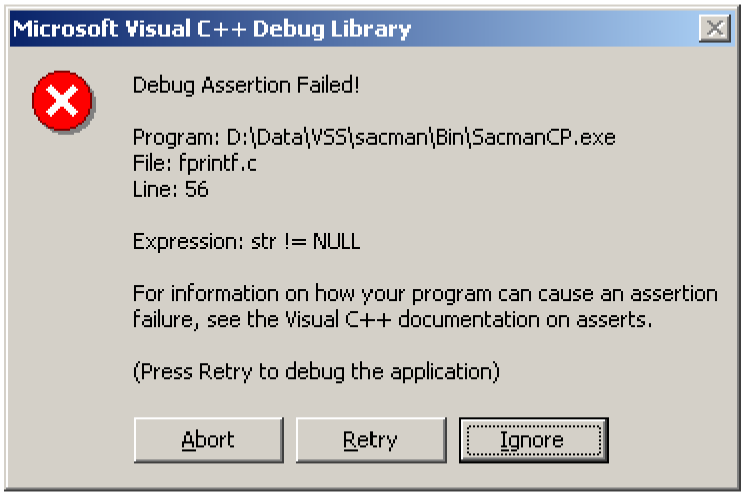
Problem

After startup, if I tried to change the dialog's selected tree item, the program would sometimes crash.



```
Debug Assertion Failed!

Program: D:\Data\VSS\sacman\Bin\SacmanCP.exe
File: fclose.c
Line: 58

Expression: stream != NULL

For information on how your program can cause an assertion
failure, see the Visual C++ documentation on asserts.

(Press Retry to debug the application)
```

Start Up Crash Error Message 5



```
Debug Assertion Failed!

Program: D:\Data\VSS\sacman\Bin\SacmanCP.exe
File: fprintf.c
Line: 56

Expression: str != NULL

For information on how your program can cause an assertion
failure, see the Visual C++ documentation on asserts.

(Press Retry to debug the application)
```

Start Up Crash Error Message 6

Depending on the circumstances, either of the prior two messages would appear.

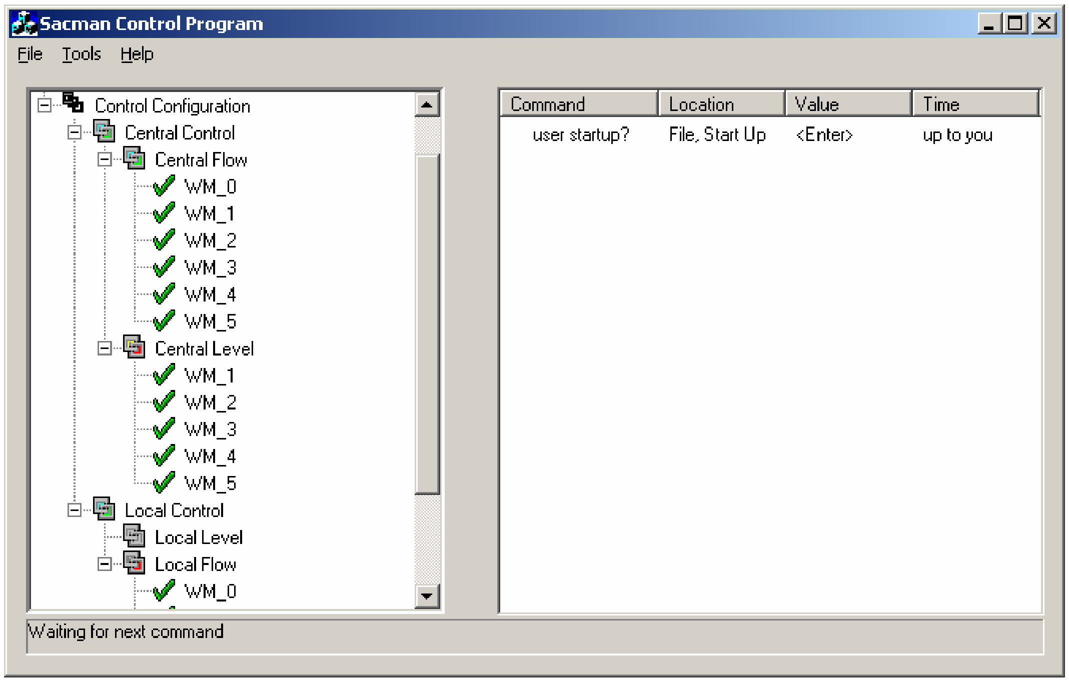
Noticing that these assertions make reference "fprintf" and "fclose," I started to suspect that the error was result of trying to write to a file. Furthermore, since the error occurred when there was a potential for multithreaded activity, it made sense that the problem might be due to competitive use of CLogNode (which writes diagnostic information to a file). Sure enough, I discovered that the CSacmanCPDlg::UpdateDisplayList() function did instantiate a CLogNode object. This meant that as I made changes to the selected tree item, I would trigger writing to the log file. Simultaneously, the CCanalManager::ImplementSchedule routine was also leading to the use of CLogNode objects. Therefore, the solution to this problem was to avoid the competition.

My method for fixing the problem was to introduce a CCriticalSection member to the CLog class. This member would be used to "Lock" the log file whenever it was opened and "Unlock" it whenever it was closed. The SacmanCP program now seems to work correctly when the user interacts with the tree during and after startup.

Bugs in Sacman CP Tree

Problem

According to the following picture, only WM_0 through WM_5 are under central control.

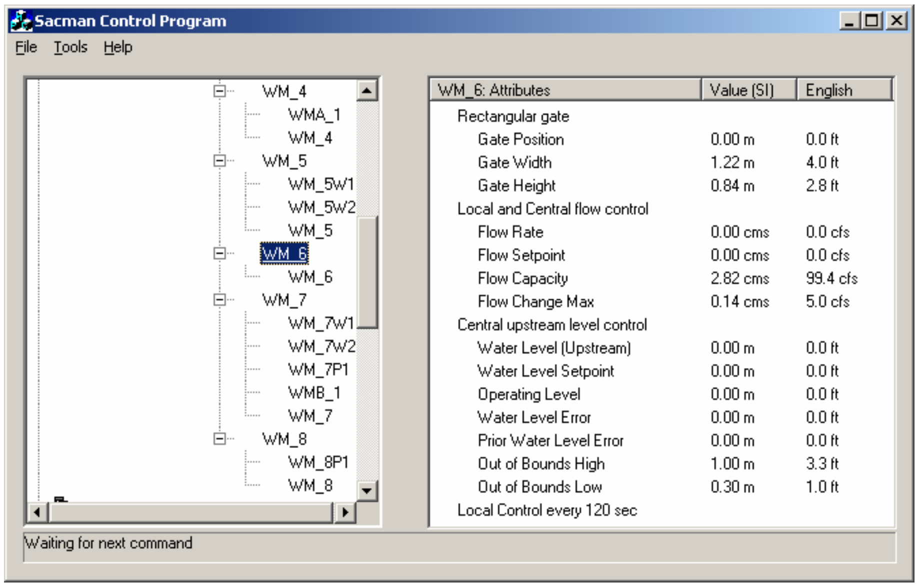


Command	Location	Value	Time
user startup?	File, Start Up	<Enter>	up to you

Waiting for next command

Sacman Cp Tree Error Message 1

However, based on the information shown for the sites in the physical representation,

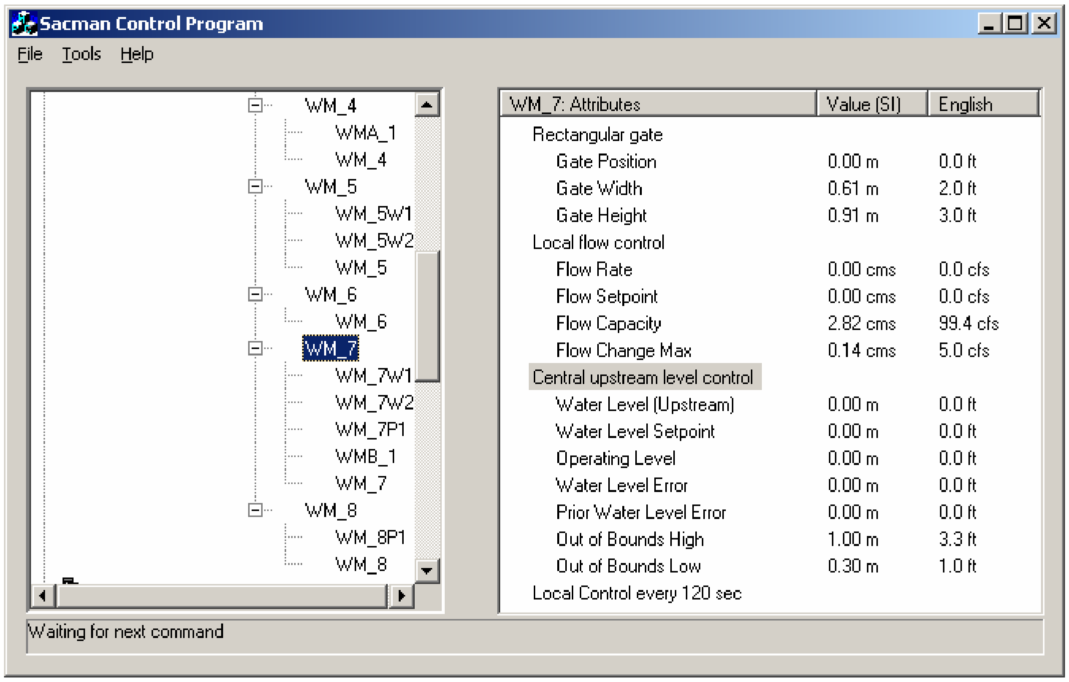


WM_6: Attributes	Value (SI)	English
Rectangular gate		
Gate Position	0.00 m	0.0 ft
Gate Width	1.22 m	4.0 ft
Gate Height	0.84 m	2.8 ft
Local and Central flow control		
Flow Rate	0.00 cms	0.0 cfs
Flow Setpoint	0.00 cms	0.0 cfs
Flow Capacity	2.82 cms	99.4 cfs
Flow Change Max	0.14 cms	5.0 cfs
Central upstream level control		
Water Level (Upstream)	0.00 m	0.0 ft
Water Level Setpoint	0.00 m	0.0 ft
Operating Level	0.00 m	0.0 ft
Water Level Error	0.00 m	0.0 ft
Prior Water Level Error	0.00 m	0.0 ft
Out of Bounds High	1.00 m	3.3 ft
Out of Bounds Low	0.30 m	1.0 ft
Local Control every 120 sec		

Waiting for next command

Sacman Cp Tree Error Message 2

WM_6 should also appear under both the Central flow and Central level control.

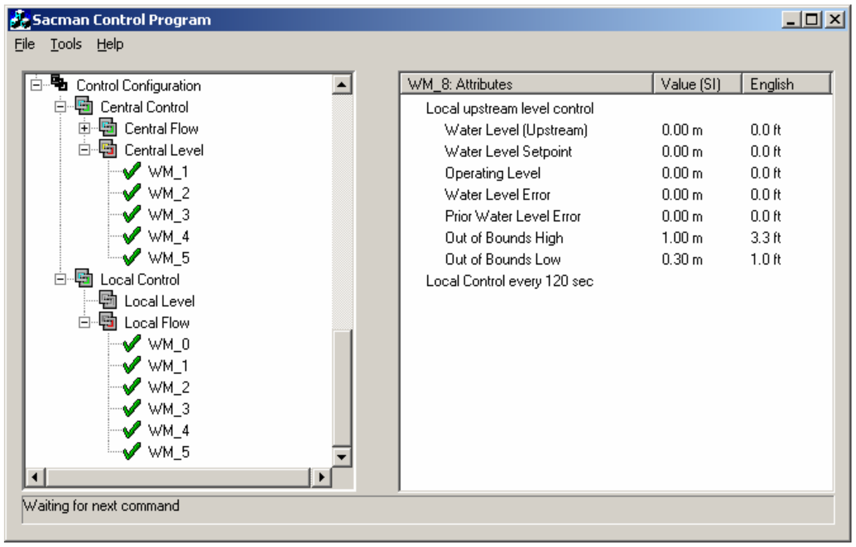


WM_7: Attributes	Value (SI)	English
Rectangular gate		
Gate Position	0.00 m	0.0 ft
Gate Width	0.61 m	2.0 ft
Gate Height	0.91 m	3.0 ft
Local flow control		
Flow Rate	0.00 cms	0.0 cfs
Flow Setpoint	0.00 cms	0.0 cfs
Flow Capacity	2.82 cms	99.4 cfs
Flow Change Max	0.14 cms	5.0 cfs
Central upstream level control		
Water Level (Upstream)	0.00 m	0.0 ft
Water Level Setpoint	0.00 m	0.0 ft
Operating Level	0.00 m	0.0 ft
Water Level Error	0.00 m	0.0 ft
Prior Water Level Error	0.00 m	0.0 ft
Out of Bounds High	1.00 m	3.3 ft
Out of Bounds Low	0.30 m	1.0 ft
Local Control every 120 sec		

Waiting for next command

Sacman Cp Tree Error Message 3

Also, according to the information for WM_7, It should be showing up under central level control and local flow control. But as shown below, it is in neither.

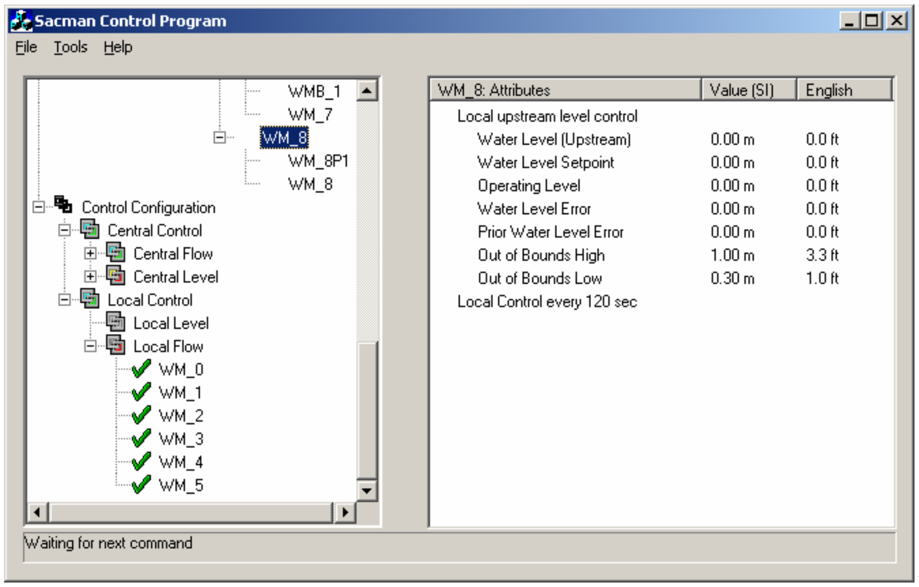


WM_8: Attributes	Value (SI)	English
Local upstream level control		
Water Level (Upstream)	0.00 m	0.0 ft
Water Level Setpoint	0.00 m	0.0 ft
Operating Level	0.00 m	0.0 ft
Water Level Error	0.00 m	0.0 ft
Prior Water Level Error	0.00 m	0.0 ft
Out of Bounds High	1.00 m	3.3 ft
Out of Bounds Low	0.30 m	1.0 ft
Local Control every 120 sec		

Waiting for next command

Sacman Cp Tree Error Message 4

Further still, WM_8 has been selected for local level control. However there are no "local level" control sites showing up in the Control Configuration tree.



WM_8: Attributes	Value (SI)	English
Local upstream level control		
Water Level (Upstream)	0.00 m	0.0 ft
Water Level Setpoint	0.00 m	0.0 ft
Operating Level	0.00 m	0.0 ft
Water Level Error	0.00 m	0.0 ft
Prior Water Level Error	0.00 m	0.0 ft
Out of Bounds High	1.00 m	3.3 ft
Out of Bounds Low	0.30 m	1.0 ft
Local Control every 120 sec		

Waiting for next command

Sacman Cp Tree Error Message 5

Possible solution:

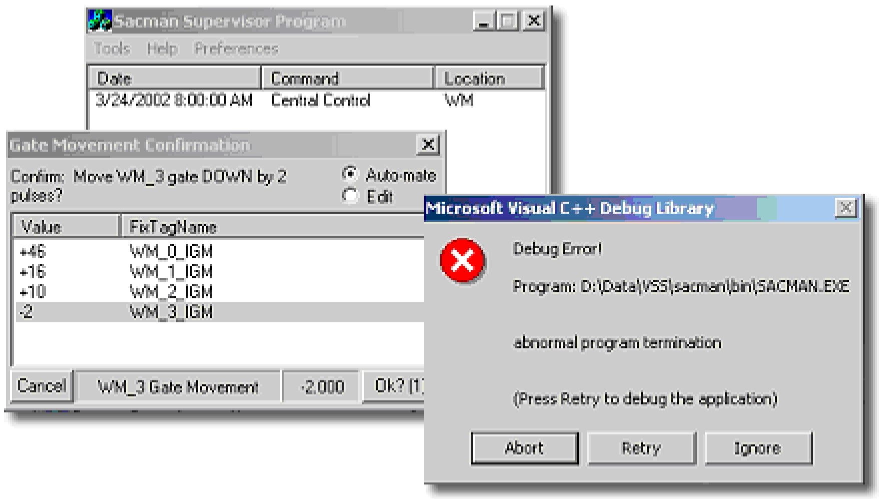
Instead of trying to go through the Control Segment and its associated Control Vector (of its State Space Feedback Configuration), perhaps it would work best to go through the system's list of sites. By going through all sites and checking their configuration, it might be easier to account for all cases.

Crash Testing

The following errors occurred when running the program unsupervised over an extended period.

Problem

Saturday March 23, 2002: 9:17 pm Ran Sacman Overnight Sunday March 24, 2002 Witnessed the following:


Sacman Supervisor Program
Tools Help Preferences
Gate Movement Confirmation
Auto-mate
Edit
Cancel
Ok? (1
Microsoft Visual C++ Debug Library
Abort
Retry
Ignore

Date	Command	Location
3/24/2002 8:00:00 AM	Central Control	WM

Confirm: Move WM_3 gate DOWN by 2 pulses?

Value	FixTagName
+46	WM_0_IGM
+16	WM_1_IGM
+10	WM_2_IGM
-2	WM_3_IGM

WM_3 Gate Movement -2.000

Debug Error!

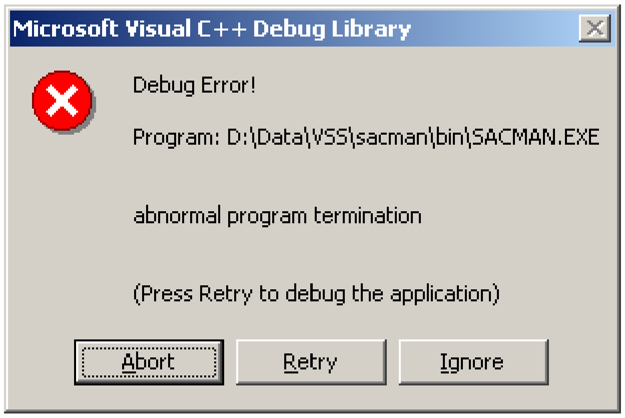
Program: D:\Data\VSS\sacman\bin\SACMAN.EXE

abnormal program termination

(Press Retry to debug the application)

Crash Test Error Message 1

After choosing retry from the error window I observed the following screen.



Debug Error!

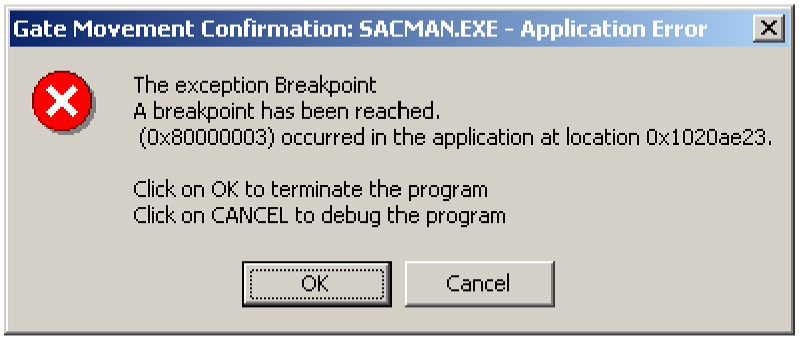
Program: D:\Data\VSS\sacman\bin\SACMAN.EXE

abnormal program termination

(Press Retry to debug the application)

Crash Test Error Message 2

Then the following appeared:



Gate Movement Confirmation: SACMAN.EXE - Application Error

The exception Breakpoint
A breakpoint has been reached.
(0x80000003) occurred in the application at location 0x1020ae23.

Click on OK to terminate the program
Click on CANCEL to debug the program

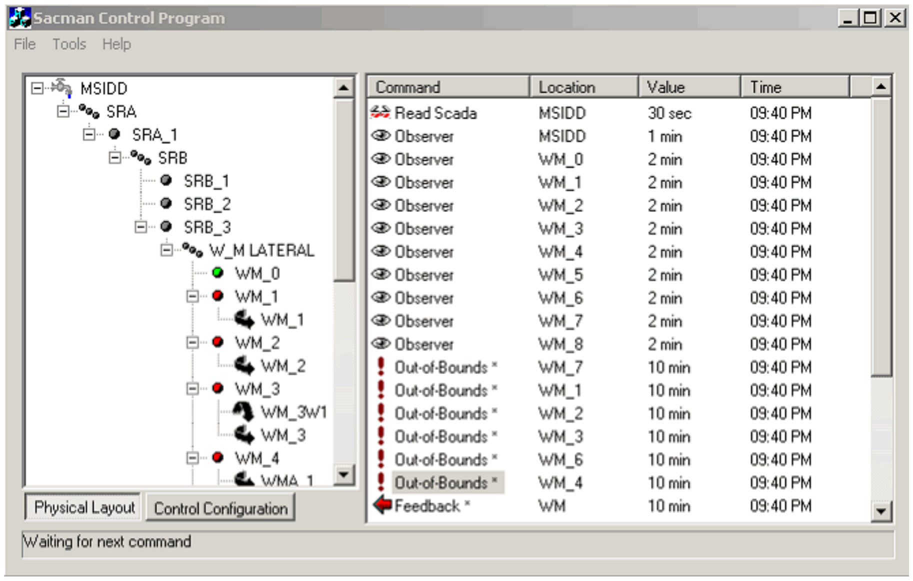
Crash Test Error Message 3

Pressed Cancel. Then the Control Program terminated. The Supervisor remained active.

Problem

10/17 11:13 AM Sacman crashed while running overnight.

The following are screen snapshots taken of SacmanCP after running it overnight.

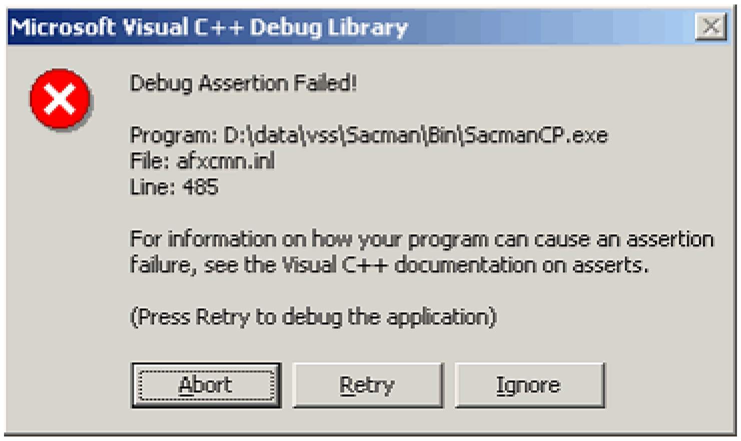


```
MSIDD
  SRA
    SRA_1
      SRB
        SRB_1
        SRB_2
        SRB_3
          W_M LATERAL
            WM_0
            WM_1
              WM_1
            WM_2
              WM_2
            WM_3
              WM_3W1
              WM_3
            WM_4
              WMA_1
```

Command	Location	Value	Time
Read Scada	MSIDD	30 sec	09:40 PM
Observer	MSIDD	1 min	09:40 PM
Observer	WM_0	2 min	09:40 PM
Observer	WM_1	2 min	09:40 PM
Observer	WM_2	2 min	09:40 PM
Observer	WM_3	2 min	09:40 PM
Observer	WM_4	2 min	09:40 PM
Observer	WM_5	2 min	09:40 PM
Observer	WM_6	2 min	09:40 PM
Observer	WM_7	2 min	09:40 PM
Observer	WM_8	2 min	09:40 PM
Out-of-Bounds *	WM_7	10 min	09:40 PM
Out-of-Bounds *	WM_1	10 min	09:40 PM
Out-of-Bounds *	WM_2	10 min	09:40 PM
Out-of-Bounds *	WM_3	10 min	09:40 PM
Out-of-Bounds *	WM_6	10 min	09:40 PM
Out-of-Bounds *	WM_4	10 min	09:40 PM
Feedback *	WM	10 min	09:40 PM

Waiting for next command

Crash Test Error Message 4



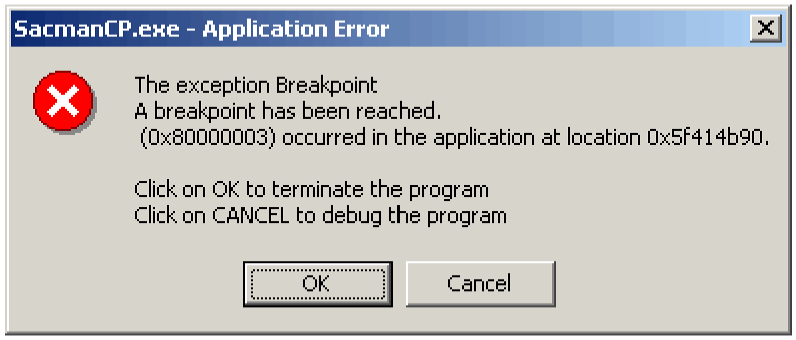
Debug Assertion Failed!

Program: D:\data\vss\Sacman\Bin\SacmanCP.exe
File: afxcmn.inl
Line: 485

For information on how your program can cause an assertion failure, see the Visual C++ documentation on asserts.

(Press Retry to debug the application)

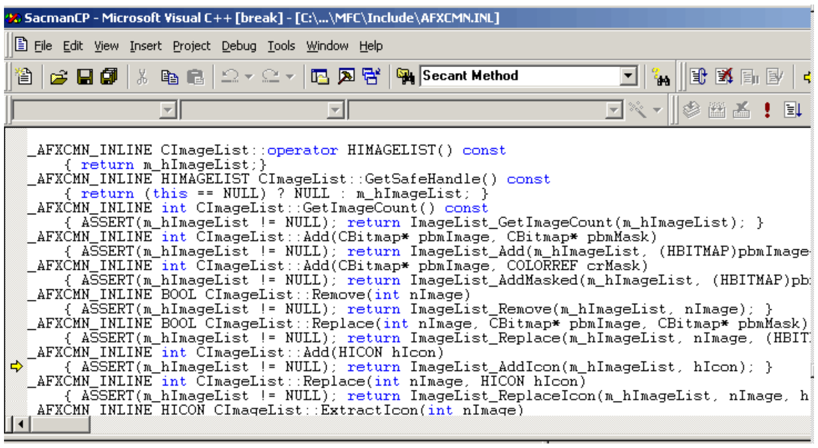
Crash Test Error Message 5



The exception Breakpoint
A breakpoint has been reached.
(0x80000003) occurred in the application at location 0x5f414b90.

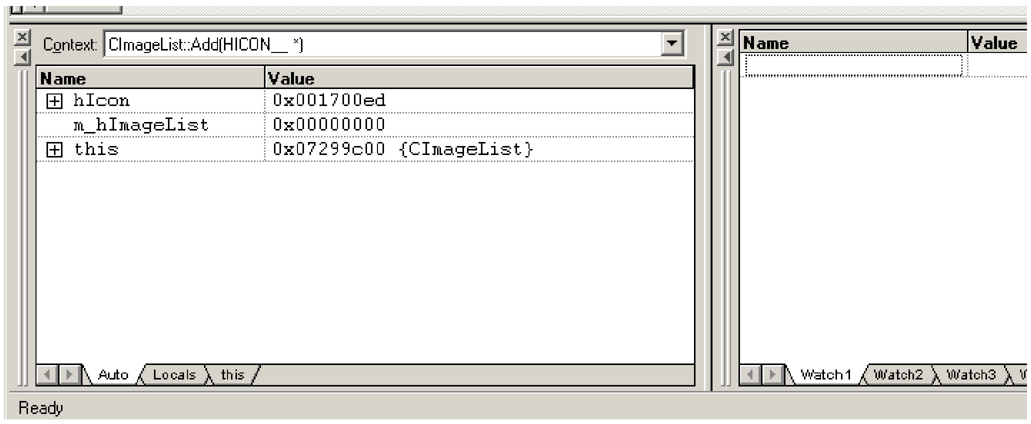
Click on OK to terminate the program
Click on CANCEL to debug the program

Crash Test Error Message 6


SacmanCP - Microsoft Visual C++ [break] - [C:\...\MFC\Include\AFXCMN.INL]
File Edit View Insert Project Debug Tools Window Help
Secant Method

```
_AFXCMN_INLINE CImageList::operator HIMAGELIST() const
    { return m_hImageList;}
_AFXCMN_INLINE HIMAGELIST CImageList::GetSafeHandle() const
    { return (this == NULL) ? NULL : m_hImageList; }
_AFXCMN_INLINE int CImageList::GetImageCount() const
    { ASSERT(m_hImageList != NULL); return ImageList_GetImageCount(m_hImageList); }
_AFXCMN_INLINE int CImageList::Add(CBitmap* pbmImage, CBitmap* pbmMask)
    { ASSERT(m_hImageList != NULL); return ImageList_Add(m_hImageList, (HBITMAP)pbmImage
_AFXCMN_INLINE int CImageList::Add(CBitmap* pbmImage, COLORREF crMask)
    { ASSERT(m_hImageList != NULL); return ImageList_AddMasked(m_hImageList, (HBITMAP)pb
_AFXCMN_INLINE BOOL CImageList::Remove(int nImage)
    { ASSERT(m_hImageList != NULL); return ImageList_Remove(m_hImageList, nImage); }
_AFXCMN_INLINE BOOL CImageList::Replace(int nImage, CBitmap* pbmImage, CBitmap* pbmMask)
    { ASSERT(m_hImageList != NULL); return ImageList_Replace(m_hImageList, nImage, (HBIT
_AFXCMN_INLINE int CImageList::Add(HICON hIcon)
    { ASSERT(m_hImageList != NULL); return ImageList_AddIcon(m_hImageList, hIcon); }
_AFXCMN_INLINE int CImageList::Replace(int nImage, HICON hIcon)
    { ASSERT(m_hImageList != NULL); return ImageList_ReplaceIcon(m_hImageList, nImage, h
_AFXCMN_INLINE HICON CImageList::ExtractIcon(int nImage)
```

Crash Test Error Message 7



```
Context: CImageList::Add(HICON__ *)
Name            Value
⊞ hIcon         0x001700ed
  m_hImageList  0x00000000
⊞ this          0x07299c00 {CImageList}
```

Crash Test Error Message 7 contd.

Potential Solution

It seems that the problem occurred during initialization of a dialog -- specifically, while trying to add an icon to an image list. I tried increasing the number of images allocated in the list from 10 to 20 and ran the program again. It has not crashed since, but has not been run for more than a day. Very long runs may still cause a crash.