

Function Fault Injector

A tool to test your exception handlers



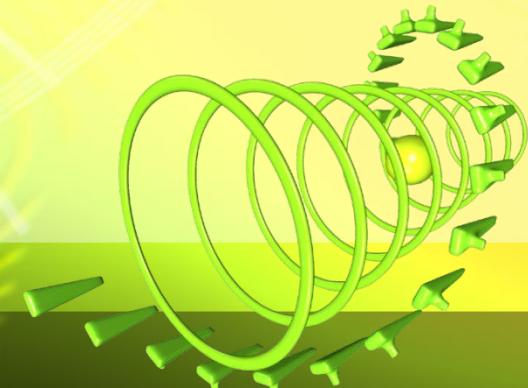
8 Weeks Summer Training
At **Microsoft** IDC Hyderabad.

Atishay Jain
10783017
3-T5

Index



- **About Fault Injection.**
- **About C# and Visual Studio IDE as well as WinCE and the Platform Builder.**
- **Application Verifier and the FFI & their working.**
- **The FFI features.**
- **DFD**
- **Working, Coding & Testing.**
- **Results & Scope of Improvement.**



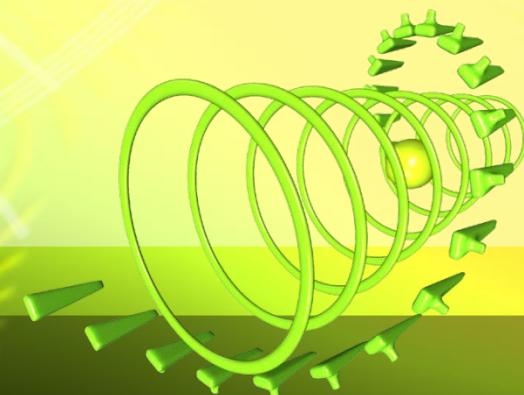


About Fault Injection

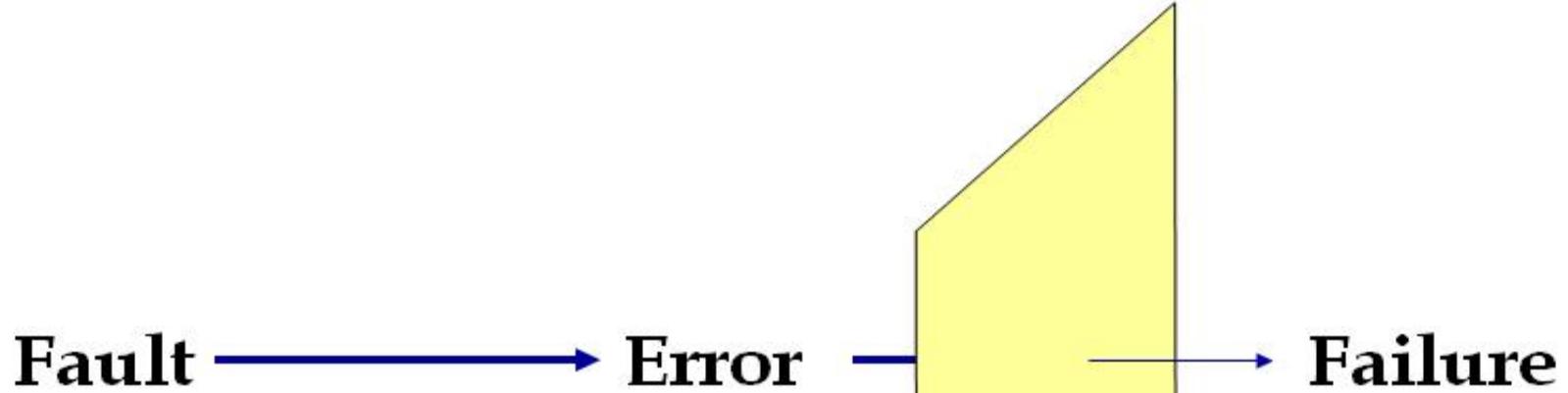


Definition

“Deliberate insertion of upsets (faults or errors) in computer systems to evaluate its behavior in the presence of faults or validate specific fault tolerance mechanisms in computers.”



What is being tested?



Fault tolerant mechanisms





Development Environment

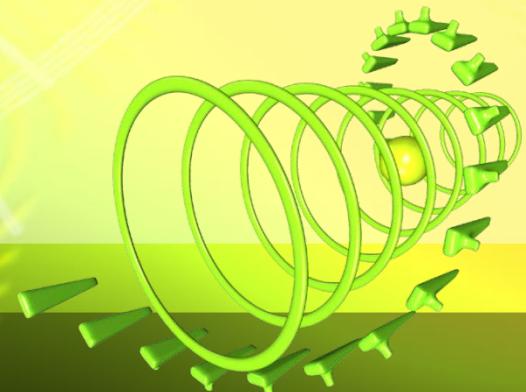




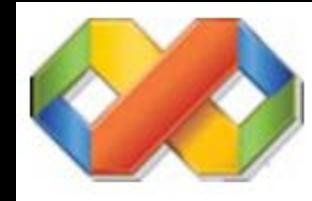
C# the language...



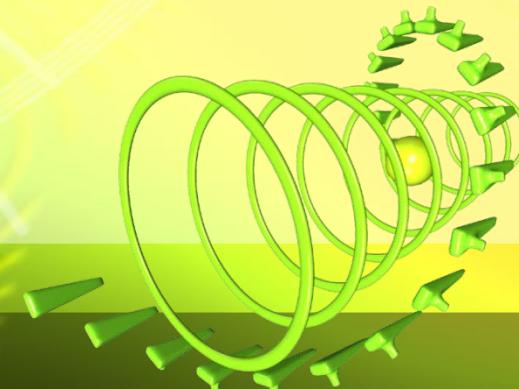
- Object Oriented.
- Windows Friendly.
- Powerful.
- Integrated into the Visual Studio Suite.
- Popular.
- The language of the reusable components.



Visual Studio



- User Friendly Development IDE.
- Intellisense.
- Debugging features like breakpoint and walk into code.
- Support for direct connection into Windows CE devices.
- Code Generation.





FFI

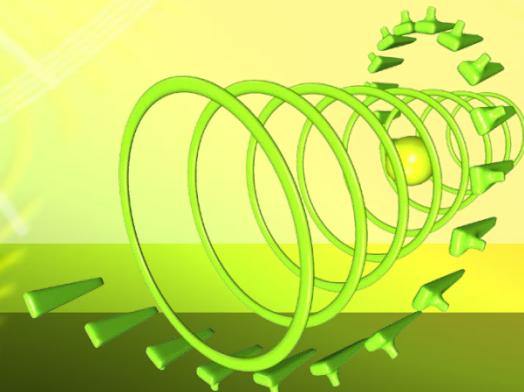
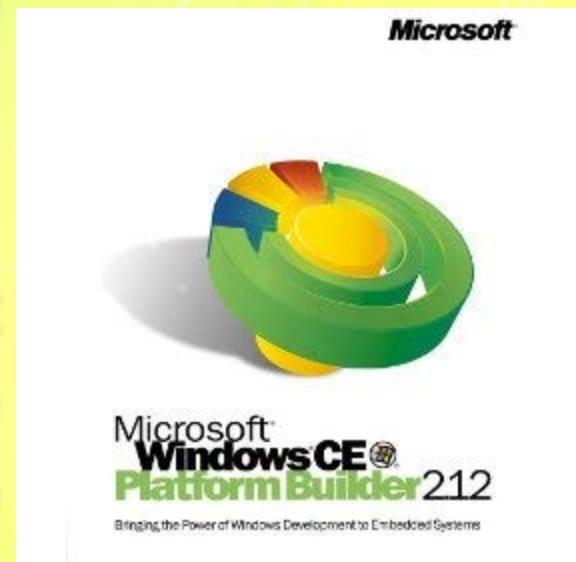
- Target OS.
- Required for Application testing.
- Directly testing of code via the use of the mobile emulator available with the Visual Studio.
- Windows programming on reduced ARMv4 architecture.



Platform Builder



- Makes OS for windows mobile device with embedded software.
- Contained the code to be tested.
- The target software development environment as well as the environment to connect to the actual device.





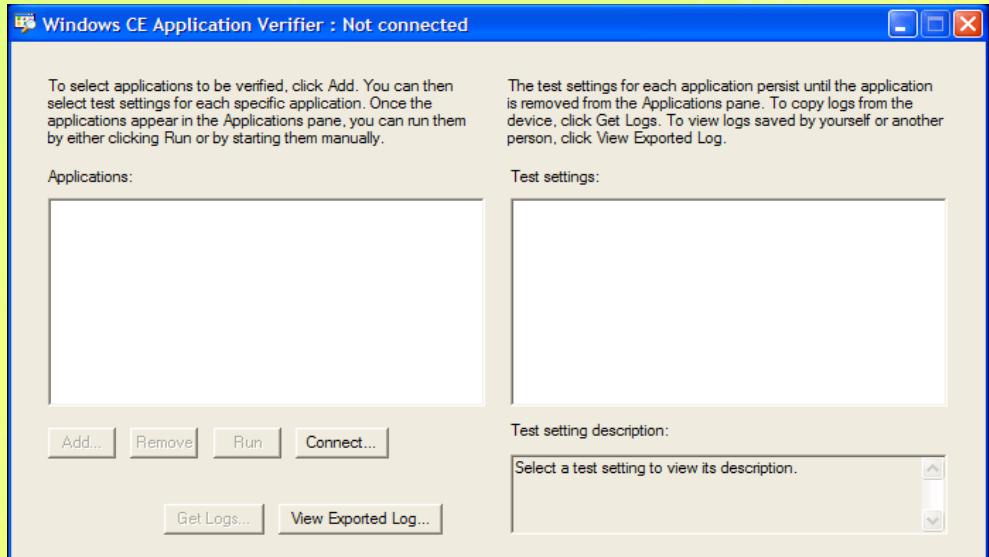
Application Verifier & FFI



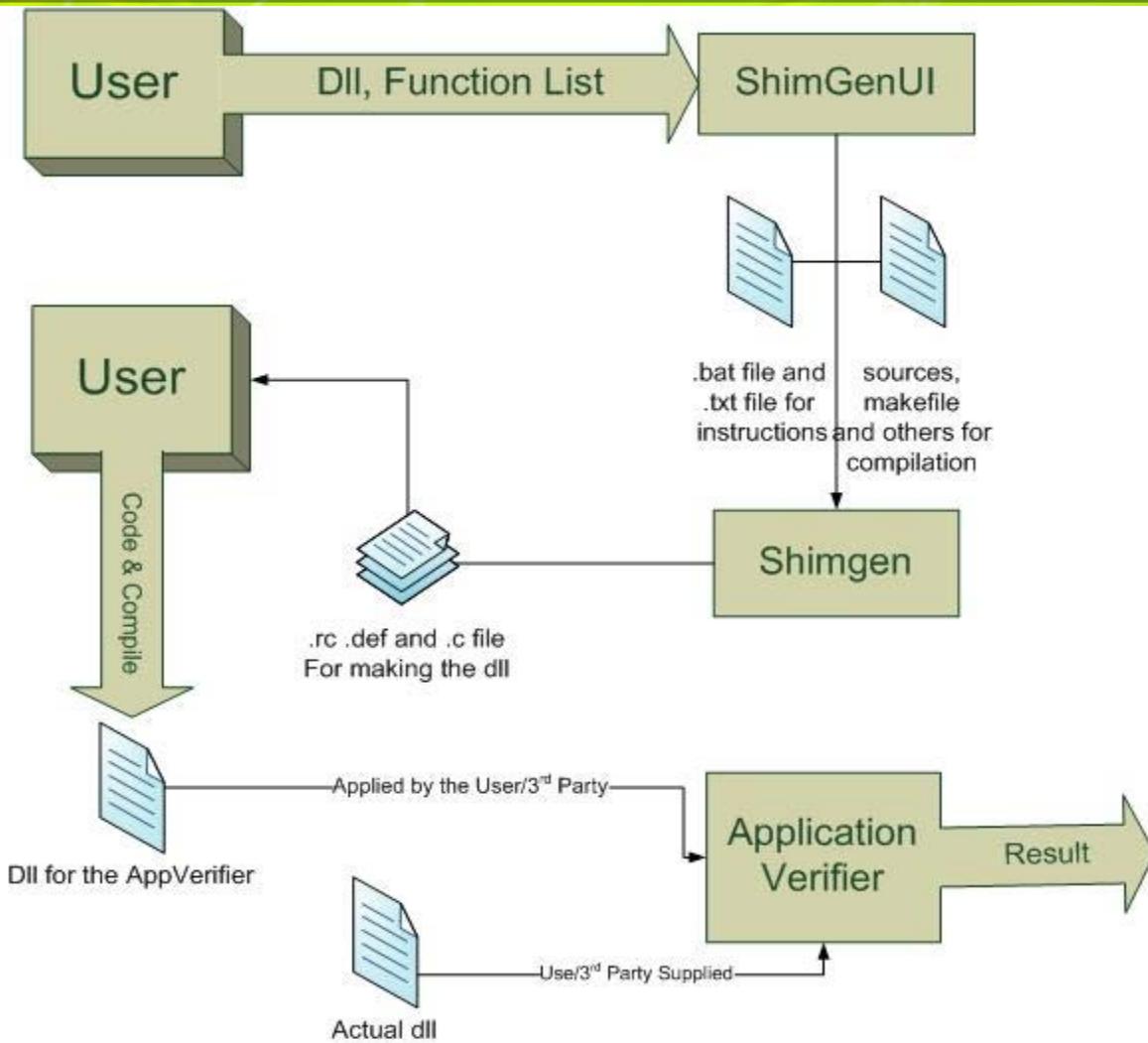
Application Verifier



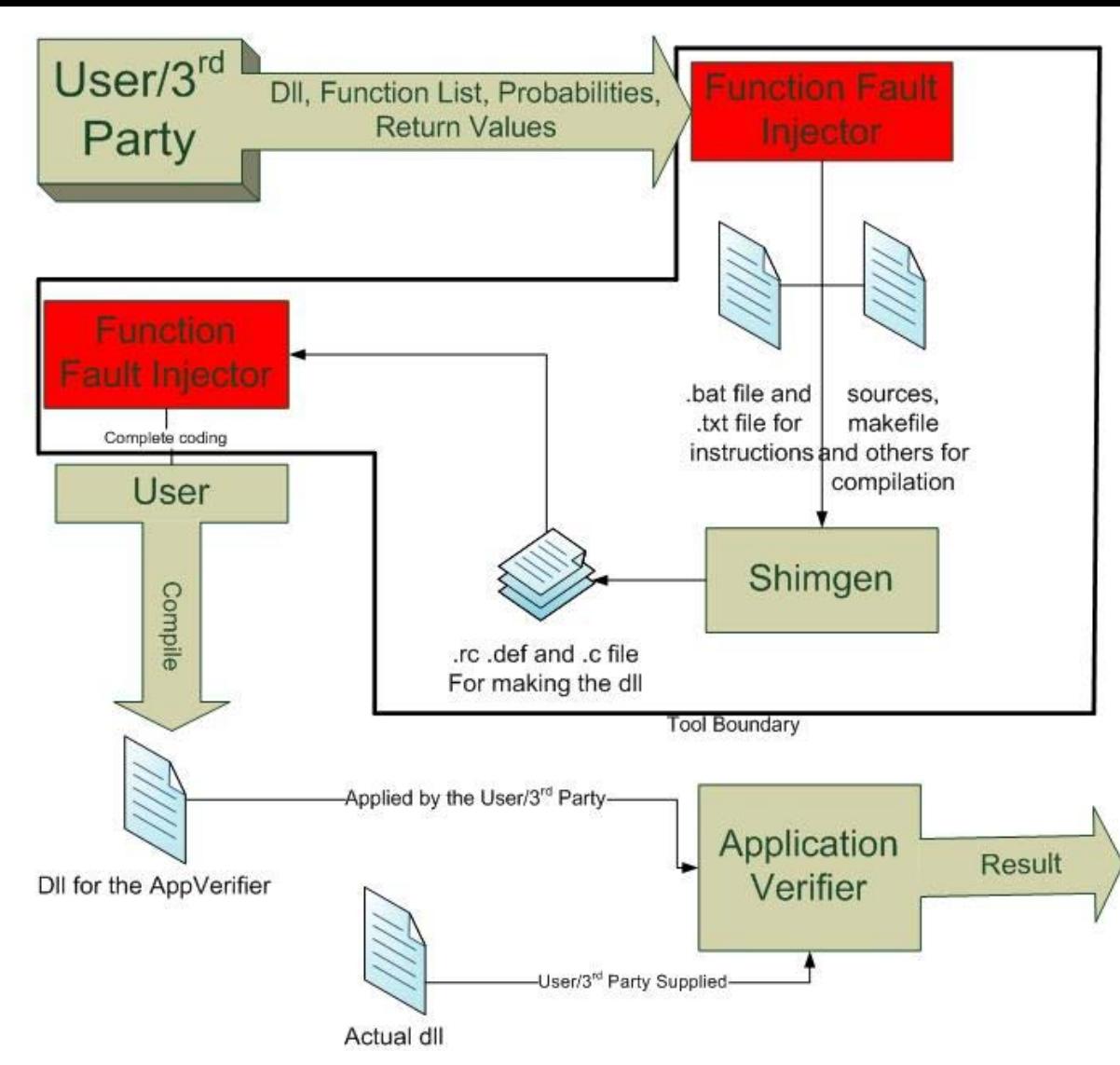
- **Freeware.**
- **Has both x86 as well as Arm versions.**
- **A lot of help available online.**
- **Can easily alter the code and disrupt calls.**
- **Already used as fault injector for many type of faults.**



Working-AppVerifier

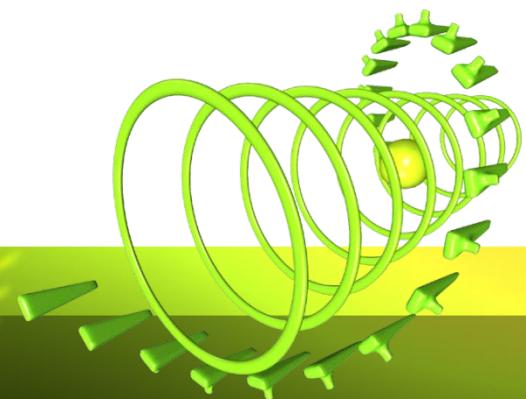


Working-FFI





Features



Features



- 1. Intercept the program flow (via the application verifier) and pass the original function to the shimmed dll where the following can be applied:**
 - a. Give alternate Return Value.**
 - b. Fill in a stub function, to replace the original one.**
 - c. Change the passed parameters to the original function return the result produced henceforth.**
 - d. Wait for the original function to complete and modify the output or the return value after the function has ended its flow.**
- 2. Have probabilities associated with each type of return and also with the original function.**
- 3. Modify the probabilities dynamically, through the windows registry.**
- 4. No injection into the original dll.**



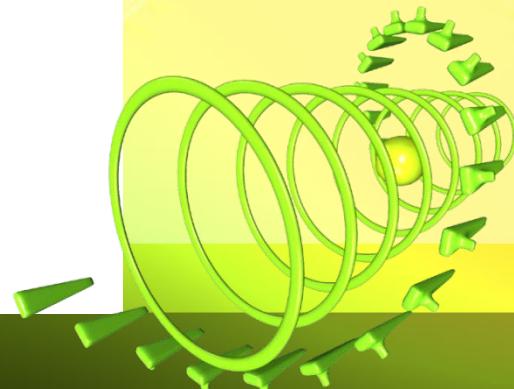
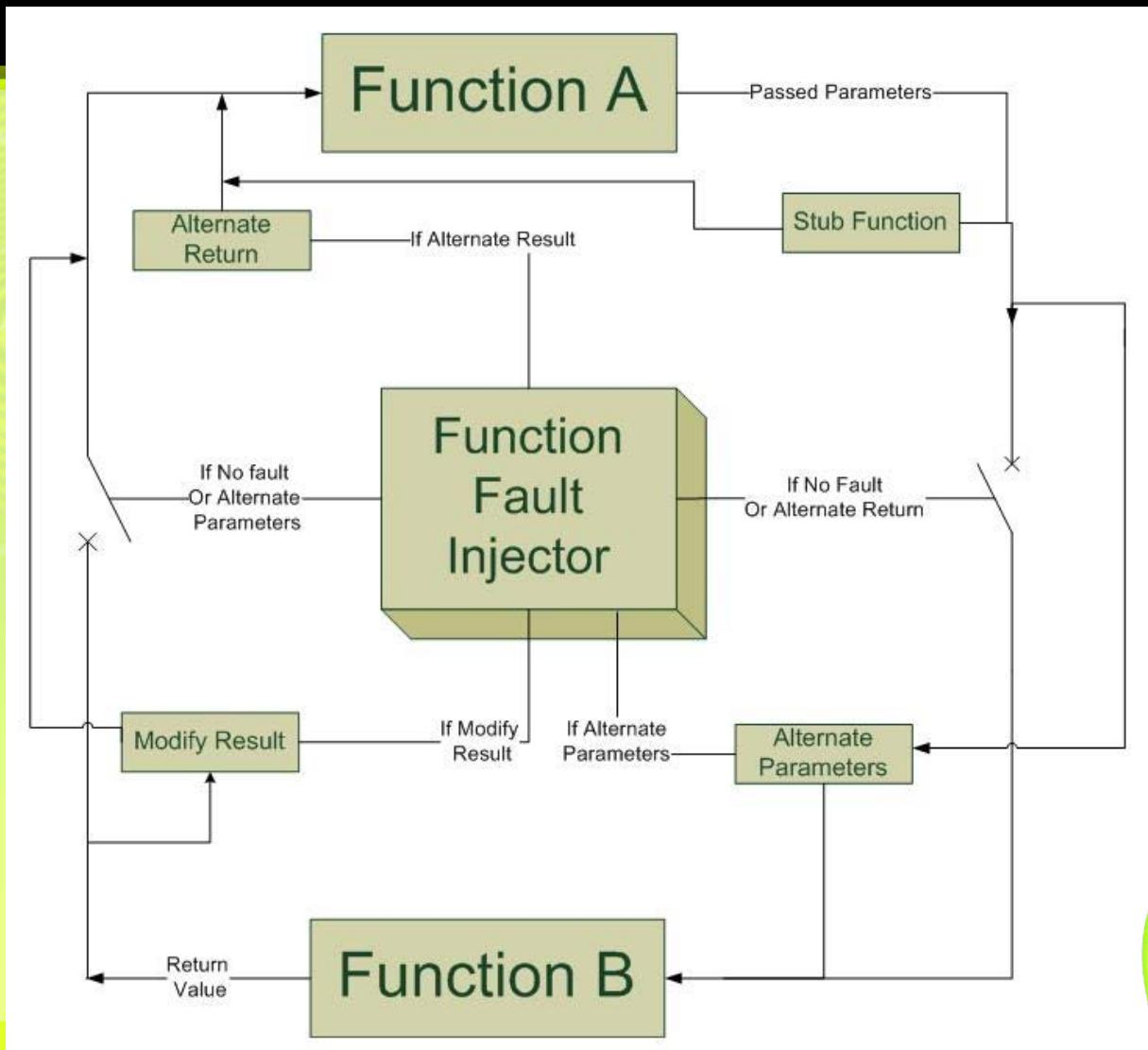
Features



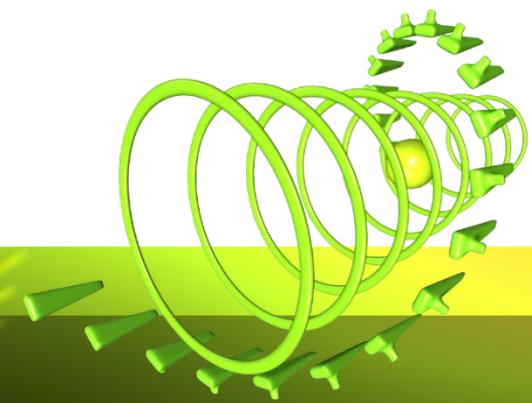
5. Can remove the shim from application via just one command.
6. Save and modify the project any time and apply the modified shim by just replacing the old one.
7. Full flexibility to write any C code and include custom headers.
8. Can also be used for API testing through the modify passed values or parameters option.
9. System dlls can be shimmed to produce other types of faults by say restricting the memory available.
10. A lot of time is saved as the tool automatically generates the code and the support files which in normal shimming process takes much more time.
11. Free from many leaks and flaws present in the original **shimgen** and hence a good alternative to produce new shims for other purposes.
12. Leaves the C files uncompiled for any modification as the user wants.



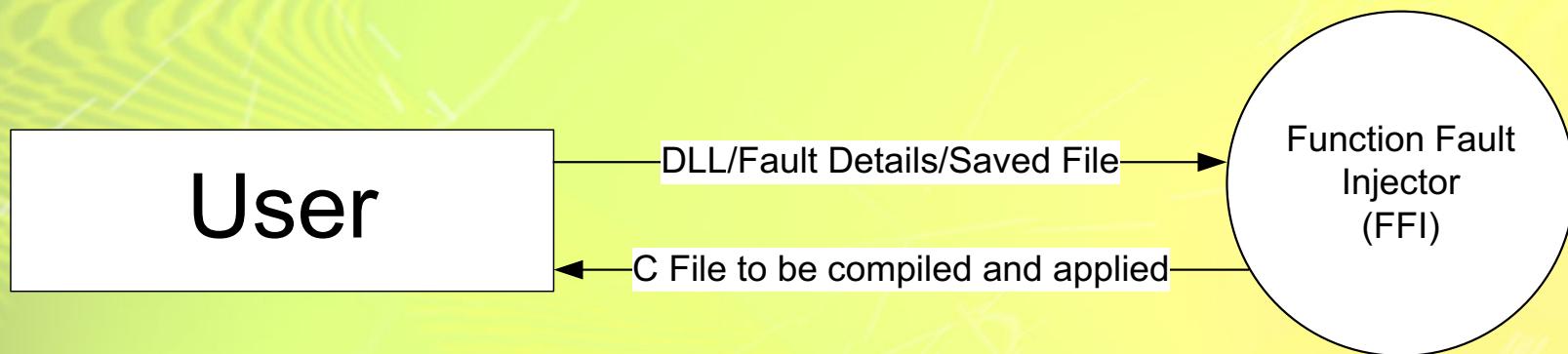
Features



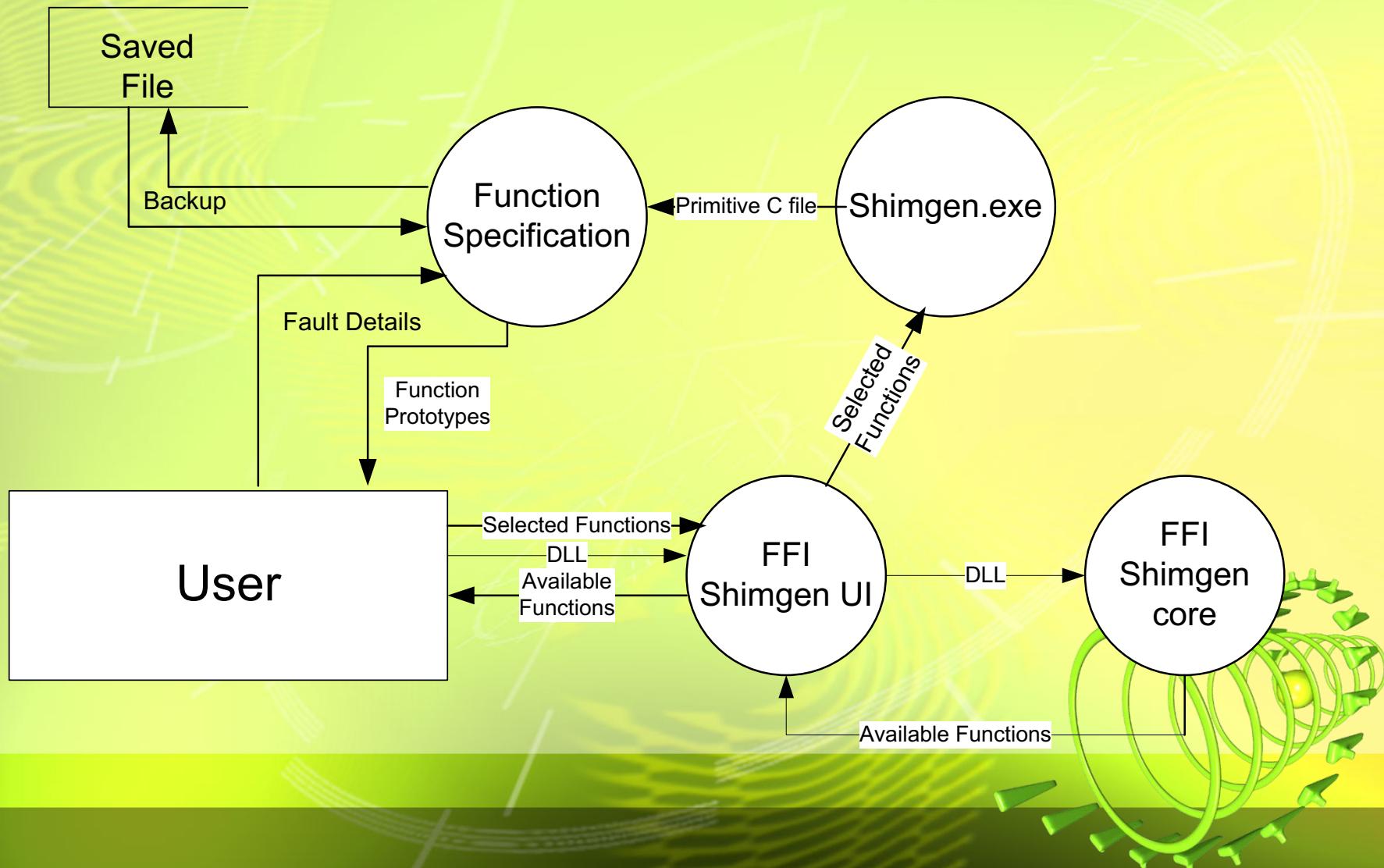
DFDs



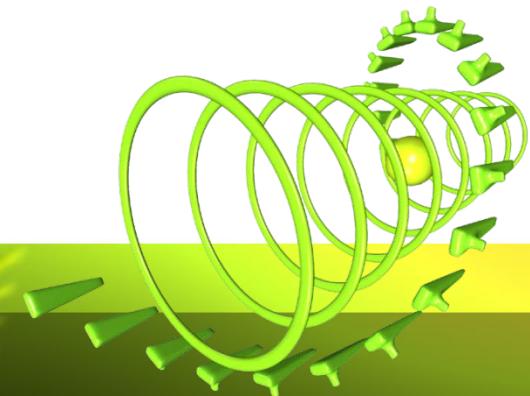
Context Diagram



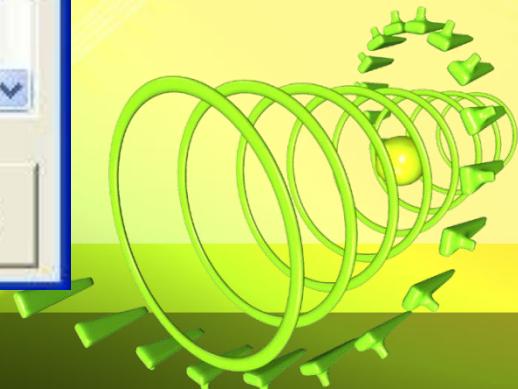
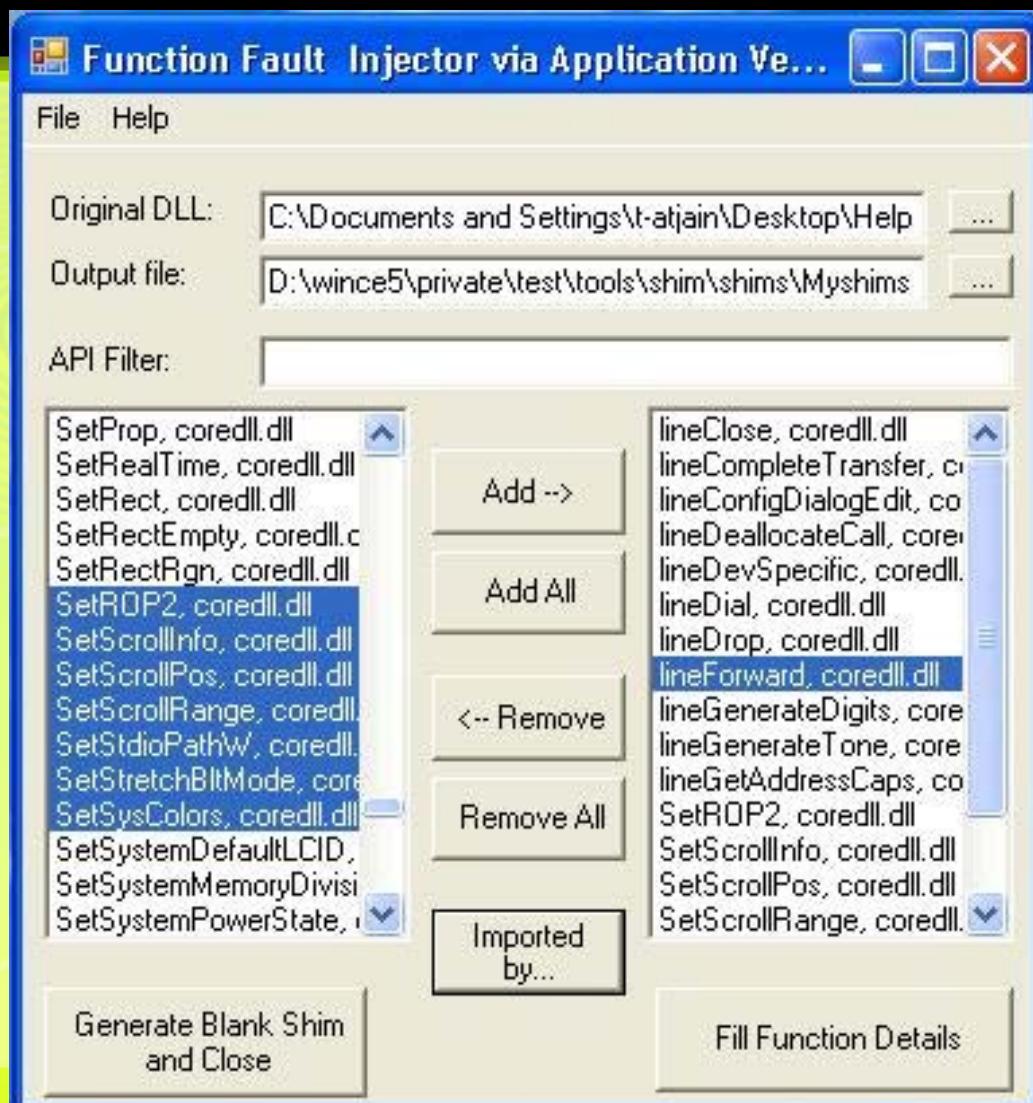
Level 1 DFD



Working, Coding & Testing



Function Selection UI



API Filter



API Filter: get

- _get_invalid_parameter
- _getstdfilex, coredll.dll
- _getws, coredll.dll
- _getws_s, coredll.dll
- fgetc, coredll.dll
- fgetpos, coredll.dll
- fgets, coredll.dll
- fgetwc, coredll.dll
- fgetws, coredll.dll
- getchar, coredll.dll
- GetForegroundKeyboard
- GetKeyboardTarget, cor
- gets, coredll.dll
- gets_s, coredll.dll
- getwchar, coredll.dll

Add -->

Add All

<- Remove

Remove All

Imported by...

- lineClose, coredll.dll
- lineCompleteTransfer, c
- lineConfigDialogEdit, co
- lineDeallocateCall, core
- lineDevSpecific, coredll.
- lineDial, coredll.dll
- lineDrop, coredll.dll
- lineForward, coredll.dll
- lineGenerateDigits, core
- lineGenerateTone, core
- lineGetAddressCaps, co
- SetROP2, coredll.dll
- SetScrollInfo, coredll.dll
- SetScrollPos, coredll.dll
- SetScrollRange, coredll.

API Filter: get\$

- GetForegroundKeyboard
- GetKeyboardTarget, cor
- SetKeyboardTarget, cor
- SHGetShortcutTarget, c

Add -->

Add All

<- Remove

Remove All

Imported by...

- lineClose, coredll.dll
- lineCompleteTransfer, c
- lineConfigDialogEdit, co
- lineDeallocateCall, core
- lineDevSpecific, coredll.
- lineDial, coredll.dll
- lineDrop, coredll.dll
- lineForward, coredll.dll
- lineGenerateDigits, core
- lineGenerateTone, core
- lineGetAddressCaps, co
- SetROP2, coredll.dll
- SetScrollInfo, coredll.dll
- SetScrollPos, coredll.dll
- SetScrollRange, coredll.

API Filter: ^get

- getchar, coredll.dll
- gets, coredll.dll
- gets_s, coredll.dll
- getwchar, coredll.dll

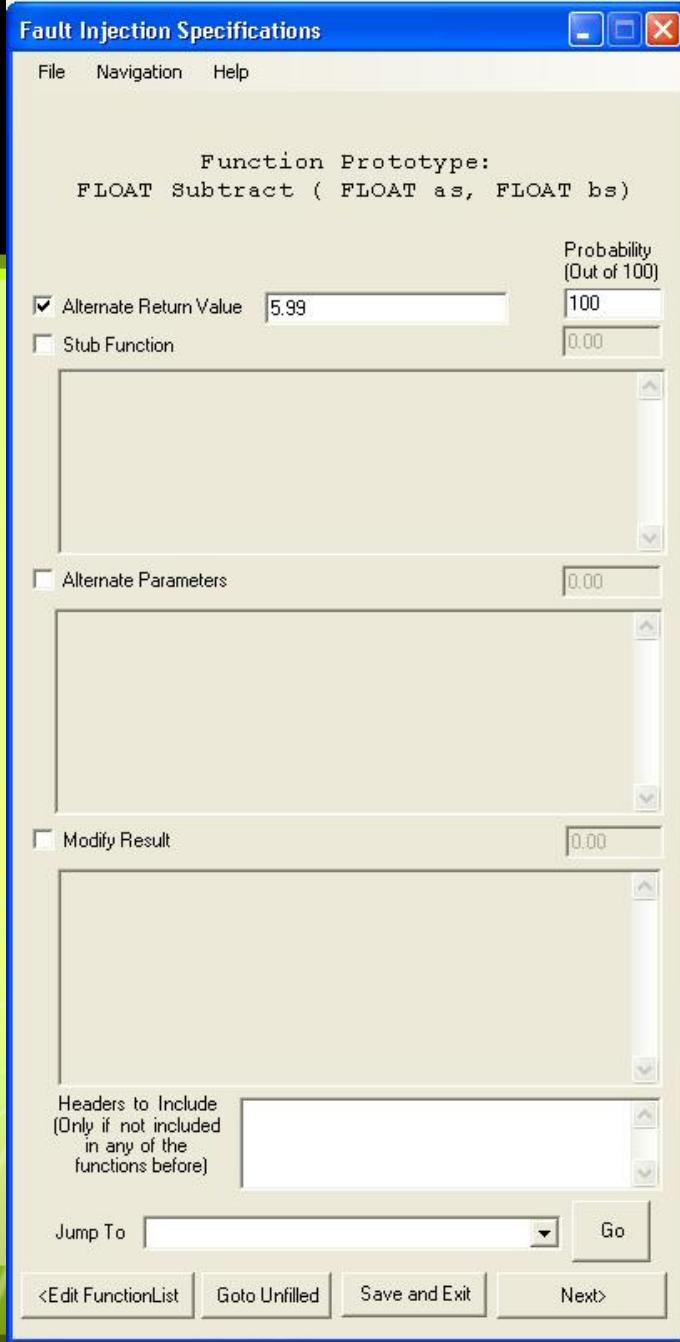
Add -->

Add All

- lineClose, coredll.dll
- lineCompleteTransfer, c
- lineConfigDialogEdit, co
- lineDeallocateCall, core
- lineDevSpecific, coredll.
- lineDial, coredll.dll
- lineDrop, coredll.dll
- lineForward, coredll.dll



Filling Fault Details



Generated Files



```
add.c - Notepad
File Edit Format View Help
//FFIBFv1.0
#include <windows.h>
#ifndef UNDER_CE
#include <tchar.h>
#endif
//FFI Generated additions to the shim
//Adding in custom headers
#include<stdlib.h> // For the probability generation function's rand
//Adding user requested headers - may contain errors if user typed an incorrect c header or format.
// From Function : Add
#include<math.h>

//this is the Random Number generator function which generates the random number to match the probability
int RandomNumber()
{
    return ((int) (10000*(rand()/(RAND_MAX + 1.0)))) ;
}

//Now adding the prototypes of all the shimmed functions
// From Function : Add
int Add ( int a, int b);

// NOTE: Fill in the following stub functions. This code is a normal user dll, so
// you can call any code you'd like. To pass the call on to the 'original' function,
// simply call the original api; the application verifier engine will 'protect'
// shim dll's from having their imports redirected.

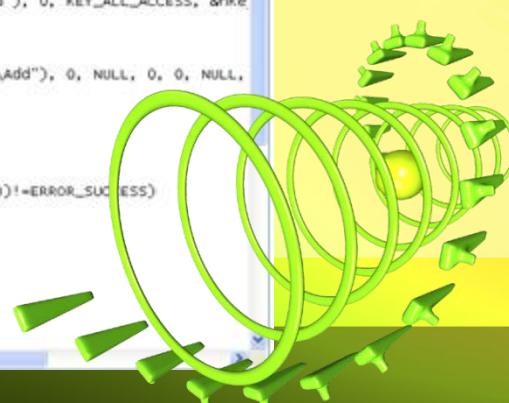
INT __stdcall APIHook_Add(
    INT a,
    INT b)
{
    INT result;
    int ThisTimeProbability = RandomNumber();
    DWORD dwValue;
    DWORD dwType;
    DWORD dwCount = sizeof(DWORD);
    HKEY hkey;
    DWORD dwDisposition;
    int InputtedProbability = 10000;
    int ProbabilityofAlternateParameters=0;
    int ProbabilityofAlternateReturn=0;
    int ProbabilityofModifyResult=0;
    int ProbabilityofStub=0;

    //Do not Modify. Checking values from the registry
    if (RegOpenKeyEx(HKEY_LOCAL_MACHINE, TEXT("Software\\Microsoft\\Application Verifier\\Function Fault Injector\\Add.dll\\Add"), 0, KEY_ALL_ACCESS, &hkey)
    {
        // The registry does not contain an entry for the function -> it is first run. Creating Keys
        RegCreateKeyEx( HKEY_LOCAL_MACHINE, TEXT("Software\\Microsoft\\Application Verifier\\Function Fault Injector\\Add.dll\\Add"), 0, NULL, 0, 0, NULL,
        if (dwDisposition != REG_CREATED_NEW_KEY && dwDisposition != REG_OPENED_EXISTING_KEY)
            printf("\nError creating the desired subkey (permissions?).\n");

        //Writing the values to the probabilities in the registry
        InputtedProbability=3000;
        if (RegSetValueEx(hkey, TEXT("ProbabilityofAlternateReturn"), 0, REG_DWORD,(const BYTE*)&InputtedProbability, sizeof(int))!=ERROR_SUCCESS)
            printf("\nthe value of the key was not set\n");
    }

    //Reading values
    dwValue = (DWORD)0;
    RegQueryValueEx ( hkey, (LPTSTR)TEXT("ProbabilityofAlternateReturn"), NULL, &dwType, (LPBYTE)&dwValue, &dwCount );
    ProbabilityofAlternateReturn=(int) dwValue;

    ProbabilityofAlternateReturn = ProbabilityofAlternateReturn + ProbabilityofStub;
}
```



Compilation



```
BUILD: [00:0000000153:PROGC ] Saving D:\wince5\private\test\tools\Build.dat.
BUILD: [00:0000000155:PROGC ] Done.
BUILD: [00:0000000156:PROGC ]
BUILD: [00:0000000157:PROGC ] Midl
BUILD: [00:0000000158:PROGC ] Message
BUILD: [00:0000000159:PROGC ] Precomp Header
BUILD: [00:0000000160:PROGC ] Resource
BUILD: [00:0000000161:PROGC ] MASM
BUILD: [00:0000000162:PROGC ] SHASM
BUILD: [00:0000000163:PROGC ] ARMASM
BUILD: [00:0000000164:PROGC ] MIPSASM
BUILD: [00:0000000165:PROGC ] C++
BUILD: [00:0000000166:PROGC ] C
BUILD: [00:0000000167:PROGC ] Static Libraries
BUILD: [00:0000000168:PROGC ] Exe's
BUILD: [00:0000000169:PROGC ] Dll's
BUILD: [00:0000000170:PROGC ] Preprocess deffile
BUILD: [00:0000000171:PROGC ] Resx
BUILD: [00:0000000172:PROGC ] CSharp Compile
BUILD: [00:0000000173:PROGC ] Other
BUILD: [00:0000000174:PROGC ]
BUILD: [00:0000000175:PROGC ] Total 14 0 0
```

Execution & Testing



```
Windows CE>appverif -m calldemo.exe -s shim_heap.dll
```

Verifier loader: SUCCESS

```
Windows CE>appverif -m calldemo.exe -s shim_calldemo.dll -opt
```

Verifier loader: SUCCESS

```
Windows CE>s calldemo.exe
```

```
Windows CE>_
```

Windows CE Remote Registry Editor

Registry Edit View Connection Help

My Computer

Default Device

HKEY_CLASSES_ROOT

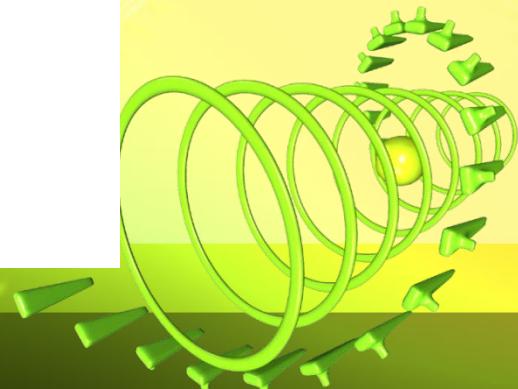
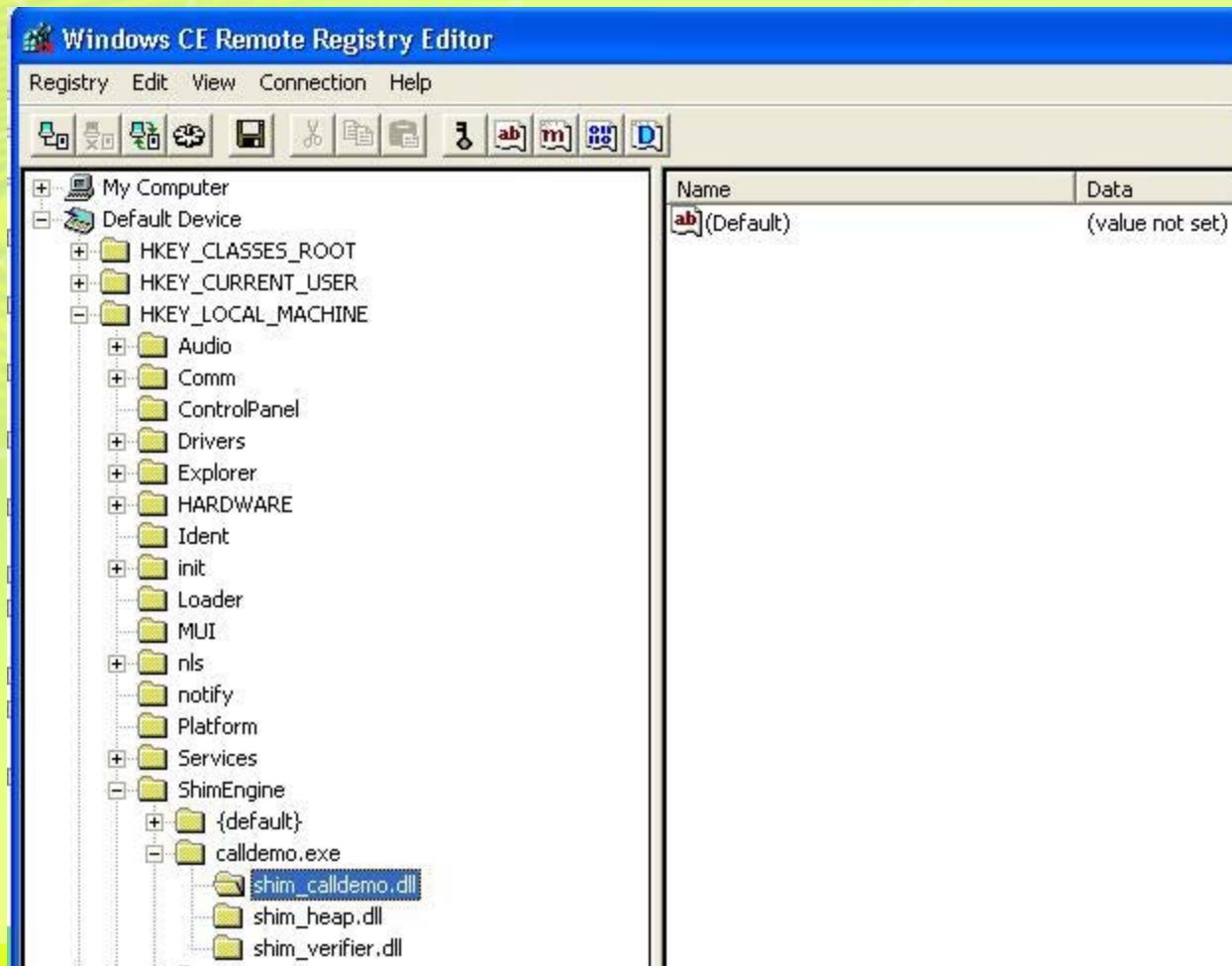
HKEY_CURRENT_USER

HKEY_LOCAL_MACHINE

- Audio
- Comm
- ControlPanel
- Drivers
- Explorer
- HARDWARE
- Ident
- init
- Loader
- MUI
- nls
- notify
- Platform
- Services
- ShimEngine
- Snd
- SOFTWARE
- Apps
- Microsoft
- .NETCompactFramework
- Application Verifier
- Function Fault Injector
- DemoDll.dll
- Add
- IsNull
- Subtract

Name	Data
ab (Default)	(value not set)
ProbabilityOfStub	2000
ProbabilityOfModifyResult	2000
ProbabilityOfAlternateReturn	2000
ProbabilityOfAlternateParameters	2000

Execution & Testing



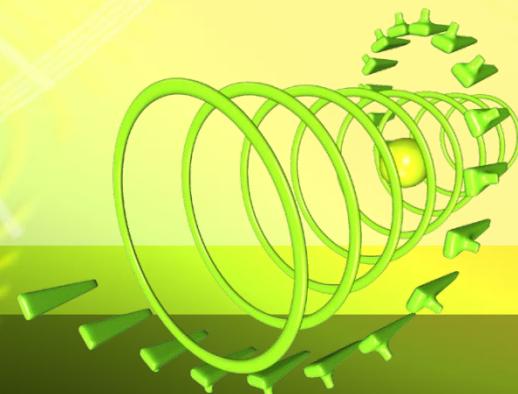
Results



Results and Future scope



- Accepted and Presently under use.
- Embedded as a part of the WinCe 6 testing tools.
- Future scope includes independent compilation of the WinCE code and hence possibly a release as a component of the next version of the Application Verifier.





**Thank
You**