



# Debugging.TV

Frame 0x13

Presenter: Dmitry Vostokov

MEMORY DUMP ANALYSIS SERVICES

[DumpAnalysis.com](http://DumpAnalysis.com)

Including Crash and Hang Analysis Audit, Training and Seminars

Sponsors

**OPENTASK**

Iterative and Incremental Publishing

# Topics

- Trace Analysis Patterns
- Optimized Message Pattern
- Return Value Optimization

# Trace Output

```
$ ~/Documents/MacOSX-Debugging/RV0/Build/Products/Debug/RV0
```

```
A 1: created.
```

```
A 1: bar called.
```

```
A 2: created.
```

```
A 2: bar called.
```

```
A 2: destructed.
```

```
A 1: destructed.
```

```
/* Expected
```

```
A 1: created.
```

```
A 1: bar called.
```

```
A 2: created.
```

```
A 3: copied.
```

```
A 3: bar called.
```

```
A 3: destructed.
```

```
A 2: destructed.
```

```
A 1: destructed.*/*
```

# Source Code

```
class A
{
public:

    int id;

    A() { id = n++; cout << "A " << id << ": created.\n"; }

    A(const A& rhs)
    {
        id = rhs.id;
        cout << "A " << id << ": copied.\n";
    }

    ~A() { cout << "A " << id << ": destructed.\n"; }

    void bar() { cout << "A " << id << ": bar called.\n"; }

    static int n;
};

int A::n = 1;

A foo()
{
    return A();
}

int main()
{
    A a;                // A 1: created.

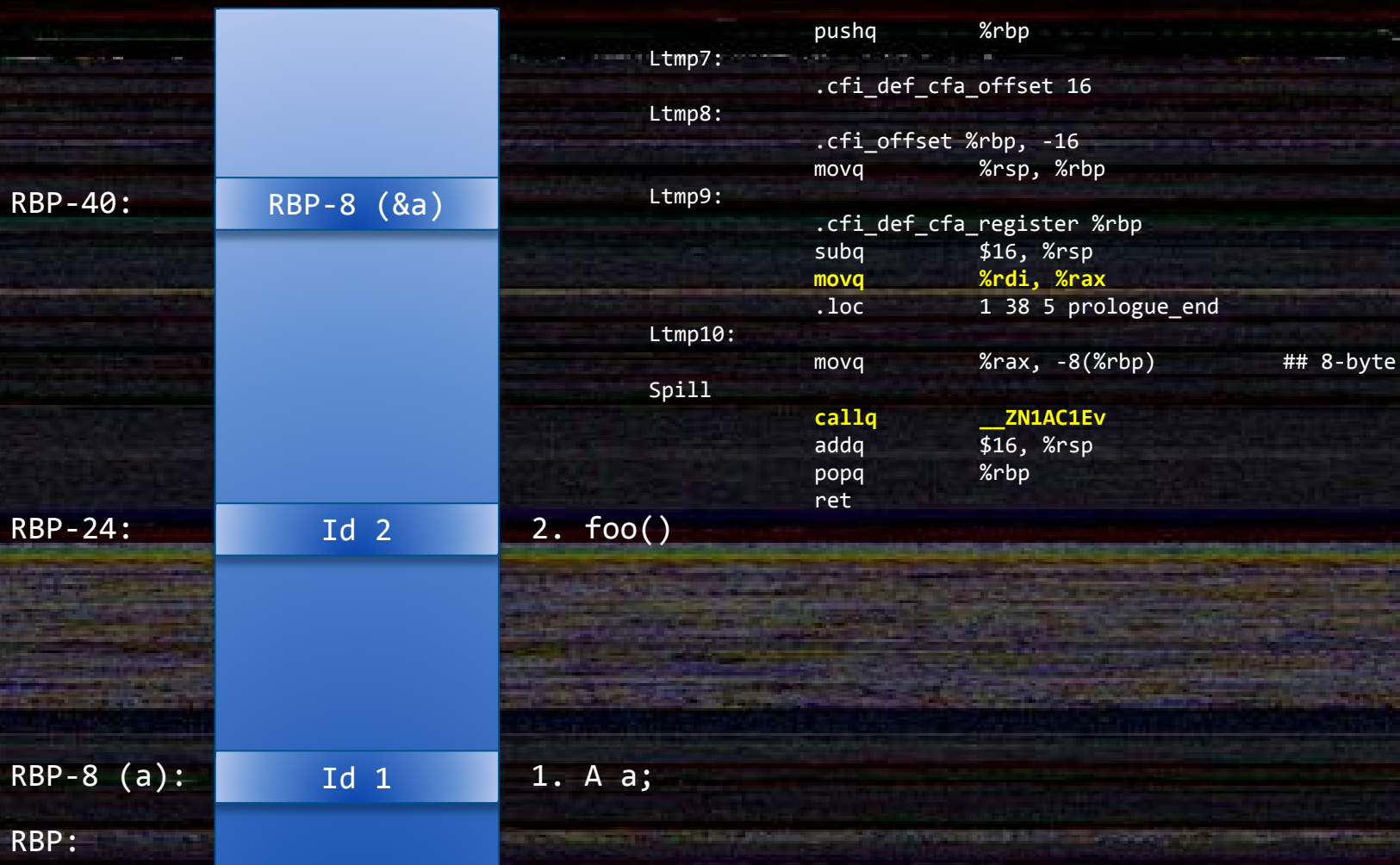
    a.bar();           // A 1: bar called.
    a = foo();         // A 2: created.
    a.bar();           // A 2: bar called.

    return 0;          // A 2: destructed. A 1: destructed.
}
```

# Assembly Extract

```
    leaq        -8(%rbp), %rax
    .loc       1 43 8 prologue_end      ## /Users/DumpAnalysis/Documents/MacOSX-Debugging/RVO/RVO/main.cpp:43:8
Ltmp32:
    movq       %rax, %rdi
    movq       %rax, -40(%rbp)         ## 8-byte Spill
    callq     __ZN1AC1Ev
    .loc       1 44 5                  ## /Users/DumpAnalysis/Documents/MacOSX-Debugging/RVO/RVO/main.cpp:44:5
Ltmp19:
    movq       -40(%rbp), %rdi         ## 8-byte Reload
    callq     __ZN1A3barEv
Ltmp20:
    jmp        LBB3_1
LBB3_1:
Ltmp33:
    ##DEBUG_VALUE: a2 <- [%rbp+$-24]+$0
    .loc       1 46 12                ## /Users/DumpAnalysis/Documents/MacOSX-Debugging/RVO/RVO/main.cpp:46:12
Ltmp21:
    leaq       -24(%rbp), %rdi
    callq     __Z3foov
Ltmp22:
    jmp        LBB3_2
LBB3_2:
    .loc       1 48 5                  ## /Users/DumpAnalysis/Documents/MacOSX-Debugging/RVO/RVO/main.cpp:48:5
Ltmp24:
    leaq       -24(%rbp), %rdi
    callq     __ZN1A3barEv
Ltmp25:
    jmp        LBB3_3
LBB3_3:
    leaq       -24(%rbp), %rdi
    .loc       1 50 5                  ## /Users/DumpAnalysis/Documents/MacOSX-Debugging/RVO/RVO/main.cpp:50:5
    movl      $0, -4(%rbp)
    movl      $1, -28(%rbp)
    .loc       1 51 1                  ## /Users/DumpAnalysis/Documents/MacOSX-Debugging/RVO/RVO/main.cpp:51:1
    callq     __ZN1AD1Ev
    leaq      -8(%rbp), %rdi
    callq     __ZN1AD1Ev
```

# RVO Internals





# !Ad Hardcore Software Support Training

October 12-15, 2012	<u><a href="#">Accelerated Windows Software Trace Analysis</a></u>
November 16-26, 2012	<u><a href="#">Accelerated Windows Memory Dump Analysis</a></u>
November 2-5, 2012	<u><a href="#">Accelerated Mac OS X Core Dump Analysis</a></u>
December 7-10, 2012	<u><a href="#">Accelerated Windows Malware Analysis</a></u>
December 17, 2012	<u><a href="#">Philosophy of Software Diagnostics</a></u> (FREE)

## Debugging<sup>3</sup>

Coming soon

Debugging.TV