the gamedesigninitiative at cornell university

Lecture 17

Debugging Strategies

There are Two Main Strategies

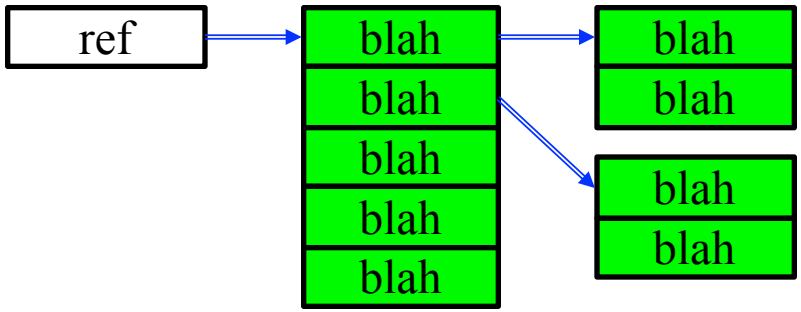
- **Confirmation**
  - Confirm everything you believe to be true
  - Find the thing that is not actually true
  - In worse case, have to look at every line of code
- **Binary Search**
  - Identify where the code is working properly
  - Identify where the code is not working properly
  - Limit confirmation to the space in between

There are Two Main Strategies

- **Confirmation**
  - Confirm everything you believe to be true
  - Find the thing that is not actually true
  - In worse case, have to look at
- Everything else is a fancy tool to do this
  - Identify where the code is working properly
  - Identify where the code is not working properly
  - Limit confirmation to the space in between

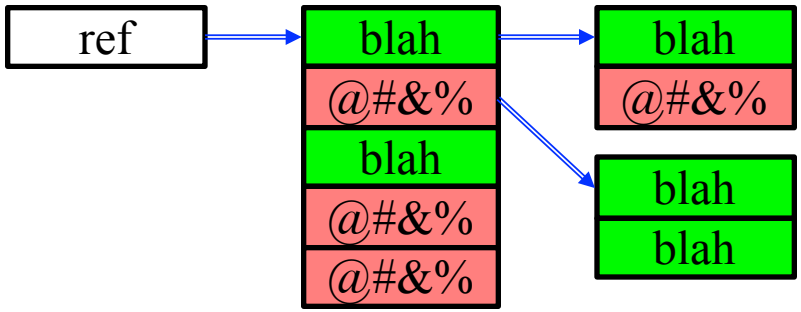
The Challenge of Finding Errors

- *Access errors* are the hardest
  - Refer to object in memory
  - Object is deleted somehow
  - Refer to attribute of object
  - May/may not cause crash
- Remember the 1110 rule
  - Error found != error cause
  - Cause is somewhere before
- Must work up the *call stack*
  - Part of the **binary search**



The Challenge of Finding Errors

- *Access errors* are the hardest
  - Refer to object in memory
  - Object is deleted somehow
  - Refer to attribute of object
  - May/may not cause crash
- Remember the 1110 rule
  - Error found != error cause
  - Cause is somewhere before
- Must work up the *call stack*
  - Part of the **binary search**



- "Deletion" is not immediate
  - Marks it for deletion
  - Will be deleted later
- Can still access object
  - Data corrupted as recycled

Primitive Confirmation Tools

- **Assertions** (CCAssert)
  - Check that your assumption is true
  - Crash the code if it is not
  - **Goal**: Make error closer to the crash
- **Logging** (CCLog)
  - Print out a variable value to check it
  - Alternatively print out a trace of program flow
  - **Goal**: View the internal program state

Primitive Confirmation Tools

- `CCAssert(test, statement)`
  - Test is any boolean statement
  - Statement must be char* or wchar*
  - **Ex**: `CCAssert(myptr != nullptr, "myptr is null")`
- `CCLog(statement, v1, v2, v3...)`
  - Uses same syntax as printf()
  - Need to use char* to display string names
  - **Ex**: `CCLog("Node is %s", node->getName().c_str())`

Problems with Logging

- **Verbose**
  - Code with print every animation frame
  - Way too much information to sort through
  - Most game designers will log to a file
- **Distortionary**
  - Logging and other I/O is a blocking operation
  - Will change the thread behavior of your app
  - Can cause errors to appear/disappear

Advanced Tools

- **Breakpoints**
  - Stop the execution of the code
  - Can continue running from that point
  - Can continue one step at a time
- **Watches**
  - Look at the value of an individual variable
  - Can drill down into object attributes
  - But only works when variable is in scope

Advanced Tools

- **Memory Dumps**
  - Look at a raw memory location
  - Does not require a variable to be in scope
  - Good way to look at heap for corruption
- **Thread Monitors**
  - Stack traces for all running threads
  - All threads are frozen by a breakpoint
  - Allows you to compare state across threads

XCode Tools

RocketBugs-desktop 〉 My Mac Running RocketBugs-mobile : RocketBugs-desktop 240

RocketBugs 〉 Classes 〉 RDGameController.cpp 〉 No Selection

```
 * not handle collisions, as those are managed by the parent class WorldController.
 * This method is called after input is read, but before collisions are resolved.
 * The very last thing that it should do is apply forces to the appropriate objects.
 *
 * @param  delta    Number of seconds since last animation frame
 */
void GameController::update(float dt) {
    _input.update(dt);

    // Process the toggled key commands
    if (_input.didDebug()) { setDebug(!isDebug()); }
    if (_input.didReset()) { reset(); }
    if (_input.didExit())  {
        CCLOG("Shutting down");
        _rootnode->shutdown();
    }

    // Apply the force to the rocket
    _rocket->setFX(_input.getHorizontal() * _rocket->getThrust());          Thread 1: breakpoint 4.1
    _rocket->setFY(_input.getVertical() * _rocket->getThrust());
```

```
MAIN, _rocket->getFY() > 1);
LEFT, _rocket->getFX() > 1);
RIGHT, _rocket->getFX() < -1);
```

```
burner

the rocket model because of our philosophy
eight.  Animation includes sounds and other
s in the model
```

Thread 1: com...read (serial)
- 0 GameController::update(float)
- 1 RocketRoot::update(float)
- 2 void cocos2d::Scheduler::sch...
- 3 decltype(std::__1::forward<co...
- 4 void std::__1::__invoke_void_re...
- 5 std::__1::__function::__func<v...
- 6 std::__1::function<void (float)...
- 7 cocos2d::Scheduler::update(f...
- 8 cocos2d::Director::drawScen...
- 9 cocos2d::DisplayLinkDirector...
- 10 cocos2d::Application::run()
- 11 main
- 12 start
- 13 start

Thread 2 Queue:
Thread 9
com.apple.audio
Thread 14
Thread 15
Thread 16
Thread 17
Thread 18
com.apple.NSEv
Thread 20
Thread 21
Thread 22
Thread 24
Thread 25
Thread 26 Queue: NS...I I/O (serial)
- 0 __open
- 14 start_wqthread

Enqueued from com.apple.main-th...
- 0 _dispatch_barrier_async_f
- 19 glfwPollEvents
- 20 cocos2d::GLViewImpl::pollE...
- 21 cocos2d::Director::drawSce...
- 22 cocos2d::DisplayLinkDirecto...
- 23 cocos2d::Application::run()

(RocketModel *) 0x101042520
- **cocos2d::BoxObstacle**
- _force (cocos2d::Vec2)
- _shipTexture = (std::__1::string) "rocket"
- _mainTexture = (std::__1::string) "flames"
- _mainBurner = (cocos2d::AnimationNode *) 0x101936a00
- _mainSound = (std::__1::string) "burn"
- _mainCycle = (bool) true
- _leftTexture = (std::__1::string) "flames-left"
- _leftBurner = (cocos2d::AnimationNode *) 0x101937600
- _leftSound = (std::__1::string) "sounds/sideburner-right.mp3"
- _leftCycle = (bool) true
- _rghtTexture = (std::__1::string) "flames-right"
- _rghtBurner = (cocos2d::AnimationNode *) 0x101937000
- _rghtCycle = (bool) true
- _rghtSound = (std::__1::string) "sounds/sideburner-left.mp3"

ocketBugs-mobile 〉 Thread 1 〉 0 GameController::update(float)

39240
- _input (InputController)
- _rootnode = (RocketRoot *) 0x103875a00
- _worldnode = (cocos2d::Node *) 0x101028ea0
- _debugnode = (cocos2d::Node *) 0x10103c7e0
- _winnode = (cocos2d::Label *) 0x10191d200
- _world = (cocos2d::WorldController *) 0x101040ee0
- _scale (cocos2d::Vec2)
- _goalDoor = (cocos2d::BoxObstacle *) 0x10103dfe0
- _rocket = (RocketModel *) 0x101042520
- _active = (bool) true
- _complete = (bool) false
- _debug = (bool) false

```
    gl.vendor: NVIDIA Corporation
    gl.supports_PVRTC: false
    gl.renderer: NVIDIA GeForce GT 750M OpenGL Engine
    cocos2d.x.compiled_with_profiler: false
    gl.max_texture_size: 16384
    gl.supports_ETC1: false
    gl.supports_BGRA8888: false
    cocos2d.x.build_type: DEBUG
    gl.supports_discard_framebuffer: false
    gl.supports_NPOT: true
    gl.supports_S3TC: true
    gl.version: 2.1 NVIDIA-10.8.14 310.42.15f01
    gl.supports_ATITC: false
    gl.max_texture_units: 16
    cocos2d.x.compiled_with_gl_state_cache: true
}

Scale is 1024.000, 576.000
(lldb)
```

No Selection

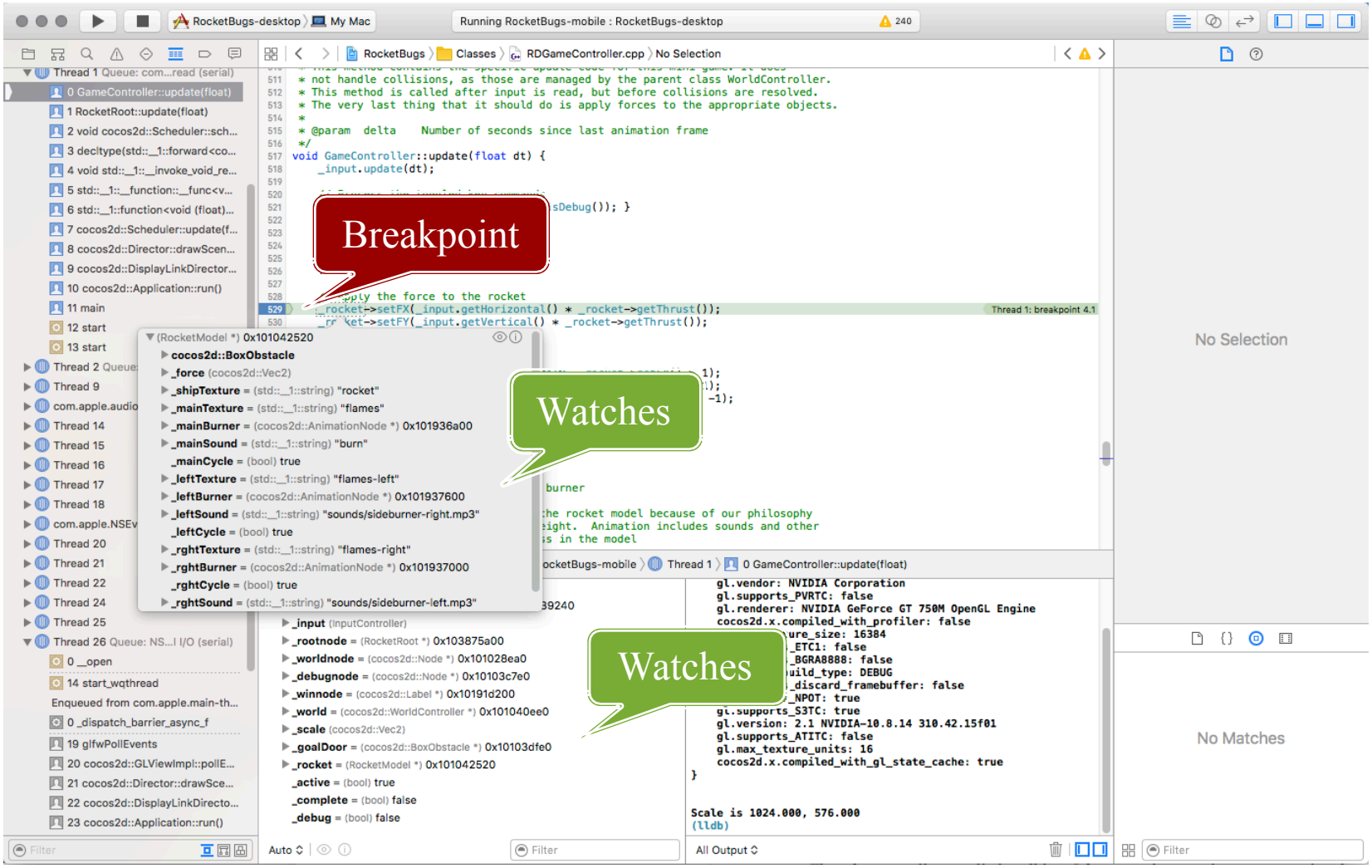
No Matches

Auto Filter All Output Filter

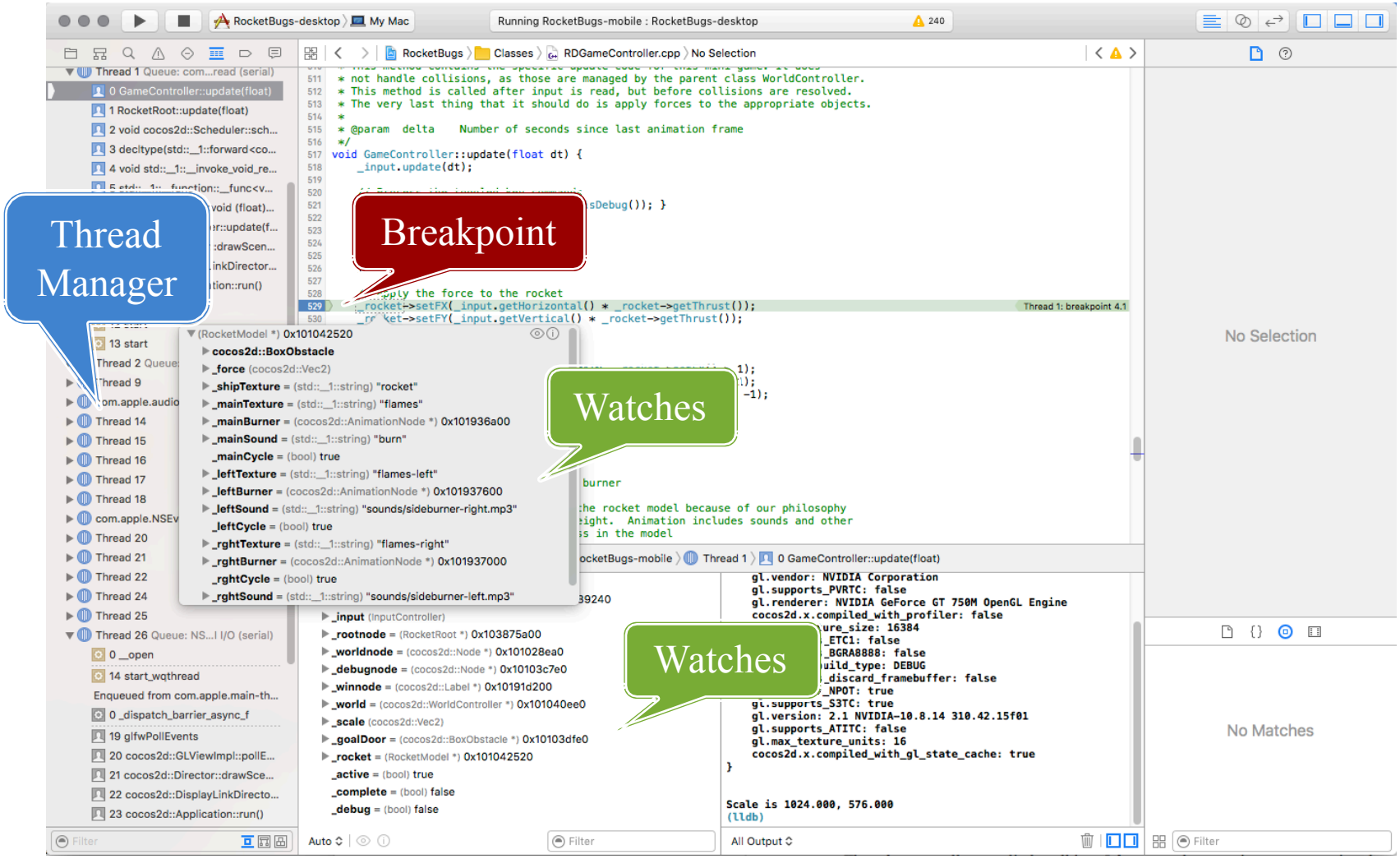
XCode Tools

Breakpoint

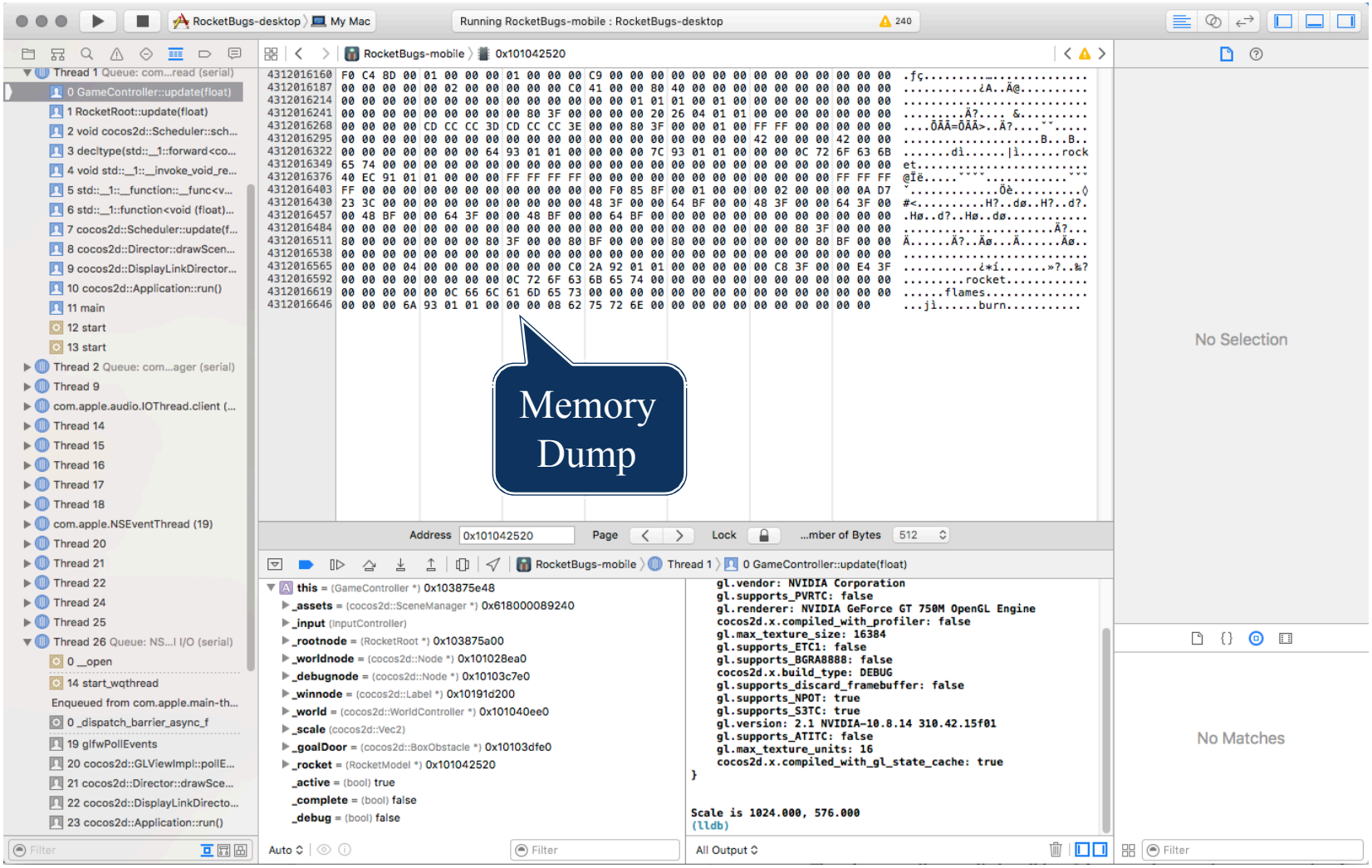
XCode Tools



XCode Tools



XCode Tools


RocketBugs-desktop
My Mac
Running RocketBugs-mobile : RocketBugs-desktop
RocketBugs-mobile
0x101042520
Memory Dump
Address
0x101042520
Page
Lock
...mber of Bytes
512
RocketBugs-mobile
Thread 1
0 GameController::update(float)
this = (GameController *) 0x103875e48
_assets = (cocos2d::SceneManager *) 0x618000089240
_input (InputController)
_rootnode = (RocketRoot *) 0x103875a00
_worldnode = (cocos2d::Node *) 0x101028ea0
_debugnode = (cocos2d::Node *) 0x10103c7e0
_winnode = (cocos2d::Label *) 0x10191d200
_world = (cocos2d::WorldController *) 0x101040ee0
_scale (cocos2d::Vec2)
_goalDoor = (cocos2d::BoxObstacle *) 0x10103dfe0
_rocket = (RocketModel *) 0x101042520
_active = (bool) true
_complete = (bool) false
_debug = (bool) false
gl.vendor: NVIDIA Corporation
gl.supports_PVRTC: false
gl.renderer: NVIDIA GeForce GT 750M OpenGL Engine
cocos2d.x.compiled_with_profiler: false
gl.max_texture_size: 16384
gl.supports_ETC1: false
gl.supports_BGRA8888: false
cocos2d.x.build_type: DEBUG
gl.supports_discard_framebuffer: false
gl.supports_NPOT: true
gl.supports_S3TC: true
gl.version: 2.1 NVIDIA-10.8.14 310.42.15f01
gl.supports_ATITC: false
gl.max_texture_units: 16
cocos2d.x.compiled_with_gl_state_cache: true
Scale is 1024.000, 576.000
(lldb)
No Selection
No Matches

Visual Studio Tools

JSONDemo (Debugging) - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Architecture Test Analyze Window Help

Process: [0x37E4] JSONDemo.exe Thread: [0x3FA8] ucrtbased.dll thread Stack Frame: LevelModel::loadCrate

Memory 1

Address: 0x02BF4B20

```
0x02BF4B20  b4 5a 2a 00 01 00 00 00 65 00 00 00 00 00 00 00 00 00 00 00 50 2b ec 05 f8 8f e8 05 cd cd cd cd cd cd cd cd cd cd cd cd cd cd 54 00 00 00 76 00 00 00  ´Z*.....e..........P+ì.ø.è.ÍÍÍÍÍÍÍÍÍÍÍÍT...v...
0x02BF4B50  00 00 00 00 00 00 00 00 00 00 00 00 00 00 42 00 00 90 41 00 00 00 00 cd cc 9c c0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 d0 c9 f5 05  ..............B...A....ÍÌœÀ................ÐÉõ.
0x02BF4B80  80 eb f7 05 d8 89 a0 02 70 2a ec 05 00 00 00 00 00 00 00 00 00 00 f0 26 ec 05 f8 08 ec 05 04 09 ec 05 04 09 ec 05 fd fd fd fd 00 00 00 00  €ë÷.Ø. .p*ì.............ð&ì.ø.ì...ì...ì.ýýýý....
0x02BF4BB0  ef fc 9c 73 00 0c 00 80 43 00 3a 00 2f 00 55 00 73 00 65 00 72 00 73 00 2f 00 57 00 61 00 6c 00 6b 00 65 00 72 00 20 00 57 00 68 00 69 00 74 00  ïüœs...€C.:./.U.s.e.r.s./.W.a.l.k.e.r. .W.h.i.t.
0x02BF4BE0  65 00 2f 00 44 00 65 00 73 00 6b 00 74 00 6f 00 70 00 2f 00 42 00 75 00 67 00 67 00 79 00 2f 00 4a 00 53 00 4f 00 4e 00 42 00 75 00 67 00 73 00  e./.D.e.s.k.t.o.p./.B.u.g.g.y./.J.S.O.N.B.u.g.s.
0x02BF4C10  2f 00 70 00 72 00 6f 00 6a 00 2e 00 77 00 69 00 6e 00 33 00 32 00 2f 00 44 00 65 00 62 00 75 00 67 00 2e 00 77 00 69 00 6e 00 33 00 32 00 2f 00  /.p.r.o.j...w.i.n.3.2./.D.e.b.u.g...w.i.n.3.2./.
0x02BF4C40  6a 00 73 00 6f 00 6e 00 73 00 2f 00 6c 00 65 00 76 00 65 00 6c 00 2e 00 6a 00 73 00 6f 00 6e 00 00 00 2f 00 2e 00 00 00 14 fc ab 73 00 0d 00 80  j.s.o.n.s./.l.e.v.e.l...j.s.o.n.../......ü«s...€
```

JSLevelModel.cpp

JSONDemo | LevelModel | loadCrate(JSONReader & reader)

```
543     Vec2 crateSize = reader.getVec2(SIZE_FIELD);
544
545     // Get the object, which is automatically retained
546     CrateModel* crate = CrateModel::create(cratePos,(Size)crateSize);
547
548     // Using the key makes too many sounds
549     // crate->setName(reader.getKey());
550     crate->setName(reader.getString(TEXTURE_FIELD));
551     CCLOG("Name is %s",crate->getName().c_str());
```

133 %

Diagnostic Tools

Select Tools | Zoom In | Zoom Out | Reset View

Diagnostics session: 3 seconds (3.158 s selected)

Events

Process Memory (MB) — Snapshot, Private Bytes

Events | Memory Usage | CPU Usage

Search Events

Event	Time	Duration	Thread
Stopped at Exception	3.15s	3,158ms	[16296]

Locals

Name	Value	Type
this	0x02bf4b20 {_root=0x00000000 <NULL> _bounds={origin={x=0.0000000	LevelMo
btype	"dynamic"	std::basic
crate	0x05f56ee8 {_crateTexture=<Error reading characters of string.> _debugC	CrateMo
cocos2d::BoxObstacle	{_shape={m_centroid={x=-1.99839716e+18 y=-1.99839716e+18 } m_vertic	cocos2d:
_crateTexture	<Error reading characters of string.>	std::basic
_debugColor	{r=0xdd 'Ý' g=0xdd 'Ý' b=0xdd 'Ý' }	cocos2d:
_debugOpacity	0xdd 'Ý'	unsigned
cratePos	{x=14.5000000 y=14.2500000 }	cocos2d:
crateSize	{x=2.00000000 y=2.00000000 }	cocos2d:

Autos | Locals | Watch 1

Call Stack

Name	Lang
JSONDemo.exe!LevelModel::loadCrate(cocos2d::JSONReader & reader) Line 557	C++
JSONDemo.exe!LevelModel::load() Line 290	C++
libcocos2d.dll!cocos2d::GenericBaseLoader::Coordinator::allocate(cocos2d::Asset * asset) Line 143	C++
libcocos2d.dll!cocos2d::GenericBaseLoader::Coordinator::loadAsync::__l2::<lambda>() Line 125	C++
[External Code]	
libcocos2d.dll!cocos2d::ThreadPool::threadFunc() Line 107	C++
[External Code]	
[Frames below may be incorrect and/or missing, no symbols loaded for ucrtbased.dll]	

Call Stack | Breakpoints | Exception Settings | Command Window | Immediate Window | Output | Error List

Visual Studio Tools

JSONDemo (Debugging) - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Architecture Test Analyze Window Help

Process: [0x37E4] JSONDemo.exe Thread: [0x3FA8] ucrtbased.dll thread Stack Frame: LevelModel::loadCrate

Memory 1

Address: 0x02BF4B20

JSLevelModel.cpp

```
543     Vec2 crateSize = reader.getVec2(SIZE_FIELD);
544
        ject, which is automatically retained
        rate = CrateModel::create(cratePos,(Size)crateSize);

        key makes too many sounds
        // crate->setName(reader.getKey());
550     crate->setName(reader.getString(TEXTURE_FIELD));
551     CCLOG("Name is %s",crate->getName().c_str());
```

Breakpoint

Diagnostic Tools

Diagnostics session: 3 seconds (3.158 s selected)

Events

Process Memory (MB)

Event	Time	Duration	Thread
Stopped at Exception	3.15s	3,158ms	[16296]

Locals

Name	Value	Type
this	0x02bf4b20 {_root=0x00000000 <NULL> _bounds={origin={x=0.00000000	LevelMo
btype	"dynamic"	std::basic
crate	0x05f56ee8 {_crateTexture=<Error reading characters of string.> _debugC	CrateMo
cocos2d::BoxObstacle	{_shape={m_centroid={x=-1.99839716e+18 y=-1.99839716e+18 } m_vertic	cocos2d:
_crateTexture	<Error reading characters of string.>	std::basic
_debugColor	{r=0xdd 'Ý' g=0xdd 'Ý' b=0xdd 'Ý' }	cocos2d:
_debugOpacity	0xdd 'Ý'	unsigned
cratePos	{x=14.5000000 y=14.2500000 }	cocos2d:
crateSize	{x=2.00000000 y=2.00000000 }	cocos2d:

Call Stack

Name	Lang
JSONDemo.exe!LevelModel::loadCrate(cocos2d::JSONReader & reader) Line 557	C++
JSONDemo.exe!LevelModel::load() Line 290	C++
libcocos2d.dll!cocos2d::GenericBaseLoader::Coordinator::allocate(cocos2d::Asset * asset) Line 143	C++
libcocos2d.dll!cocos2d::GenericBaseLoader::Coordinator::loadAsync::__l2::<lambda>() Line 125	C++
[External Code]	
libcocos2d.dll!cocos2d::ThreadPool::threadFunc() Line 107	C++
[External Code]	
[Frames below may be incorrect and/or missing, no symbols loaded for ucrtbased.dll]	

Visual Studio Tools

JSONDemo (Debugging) - Microsoft Visual Studio

Memory 1

Address: 0x02BF4B20

JSLevelModel.cpp

```
543     Vec2 crateSize = reader.getVec2(SIZE_FIELD);
544
        ject, which is automatically retained
        rate = CrateModel::create(cratePos,(Size)crateSize);

        key makes too many sounds
        // crate->setName(reader.getKey());
550     crate->setName(reader.getString(TEXTURE_FIELD));
551     CCLOG("Name is %s",crate->getName().c_str());
```

Breakpoint

Diagnostic Tools

Diagnostics session: 3 seconds (3.158 s selected)

Events

Process Memory (MB)

Event	Time	Duration	Thread
Stopped at Exception	3.15s	3,158ms	[16296]

Locals

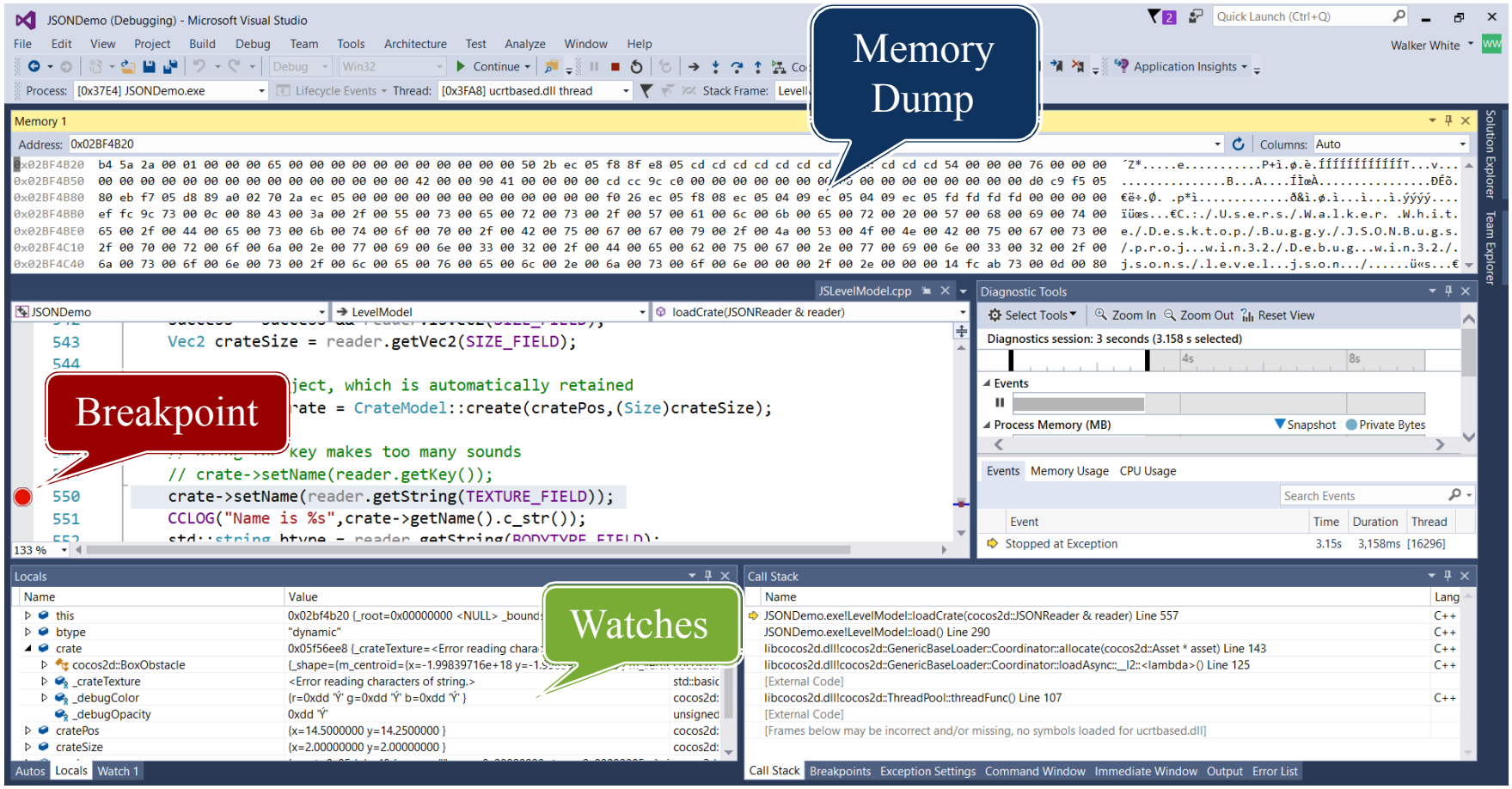
Name	Value
this	0x02bf4b20 {_root=0x00000000 <NULL> _bound
btype	"dynamic"
crate	0x05f56ee8 {_crateTexture=<Error reading chara
cocos2d::BoxObstacle	{_shape={m_centroid={x=-1.99839716e+18 y=-1.9
_crateTexture	<Error reading characters of string.>
_debugColor	{r=0xdd 'Ý' g=0xdd 'Ý' b=0xdd 'Ý' }
_debugOpacity	0xdd 'Ý'
cratePos	{x=14.5000000 y=14.2500000 }
crateSize	{x=2.00000000 y=2.00000000 }

Watches

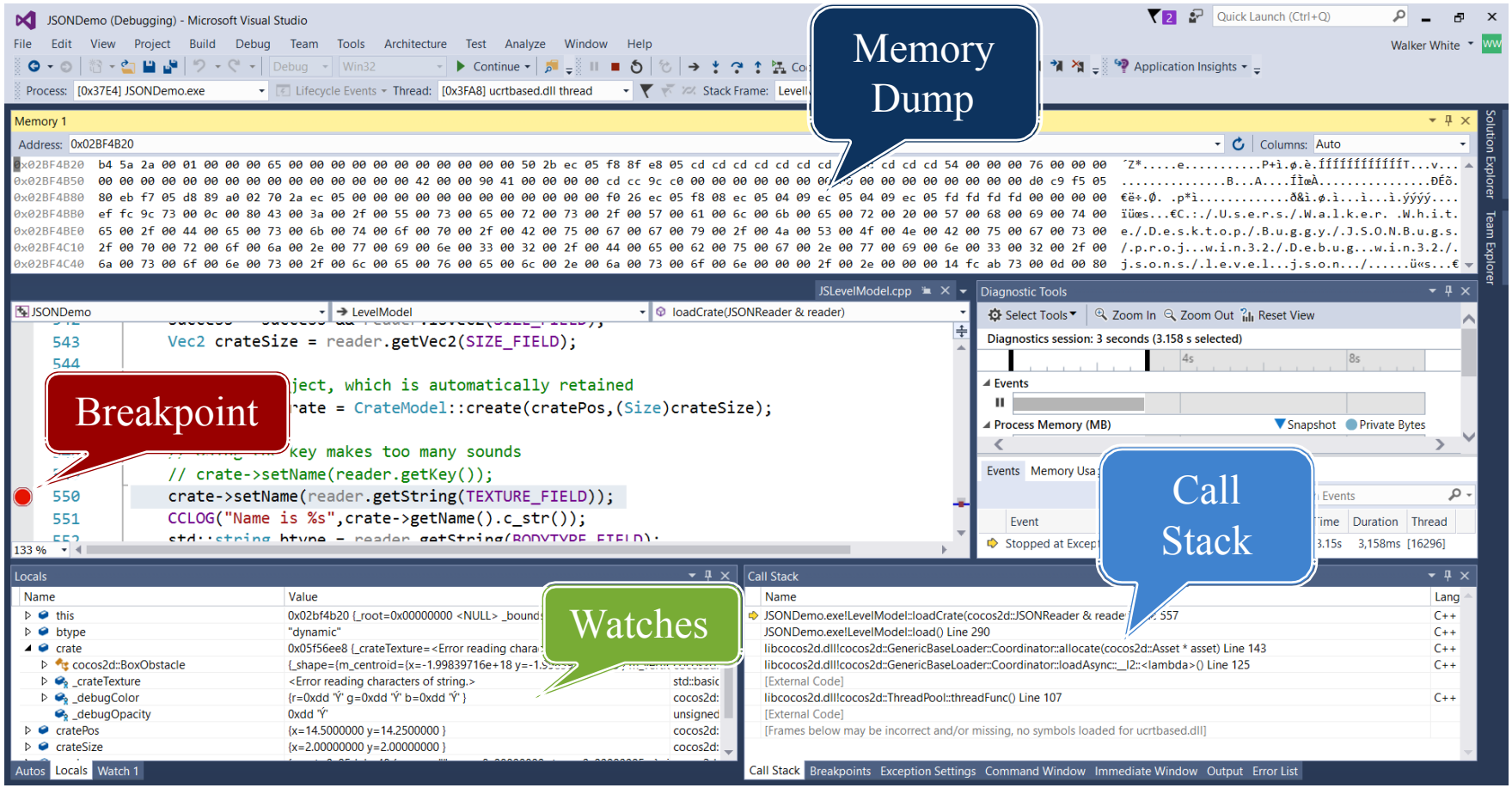
Call Stack

```
JSONDemo.exe!LevelModel::loadCrate(cocos2d::JSONReader & reader) Line 557
JSONDemo.exe!LevelModel::load() Line 290
libcocos2d.dll!cocos2d::GenericBaseLoader::Coordinator::allocate(cocos2d::Asset * asset) Line 143
libcocos2d.dll!cocos2d::GenericBaseLoader::Coordinator::loadAsync::__l2::<lambda>() Line 125
[External Code]
libcocos2d.dll!cocos2d::ThreadPool::threadFunc() Line 107
[External Code]
[Frames below may be incorrect and/or missing, no symbols loaded for ucrtbased.dll]
```

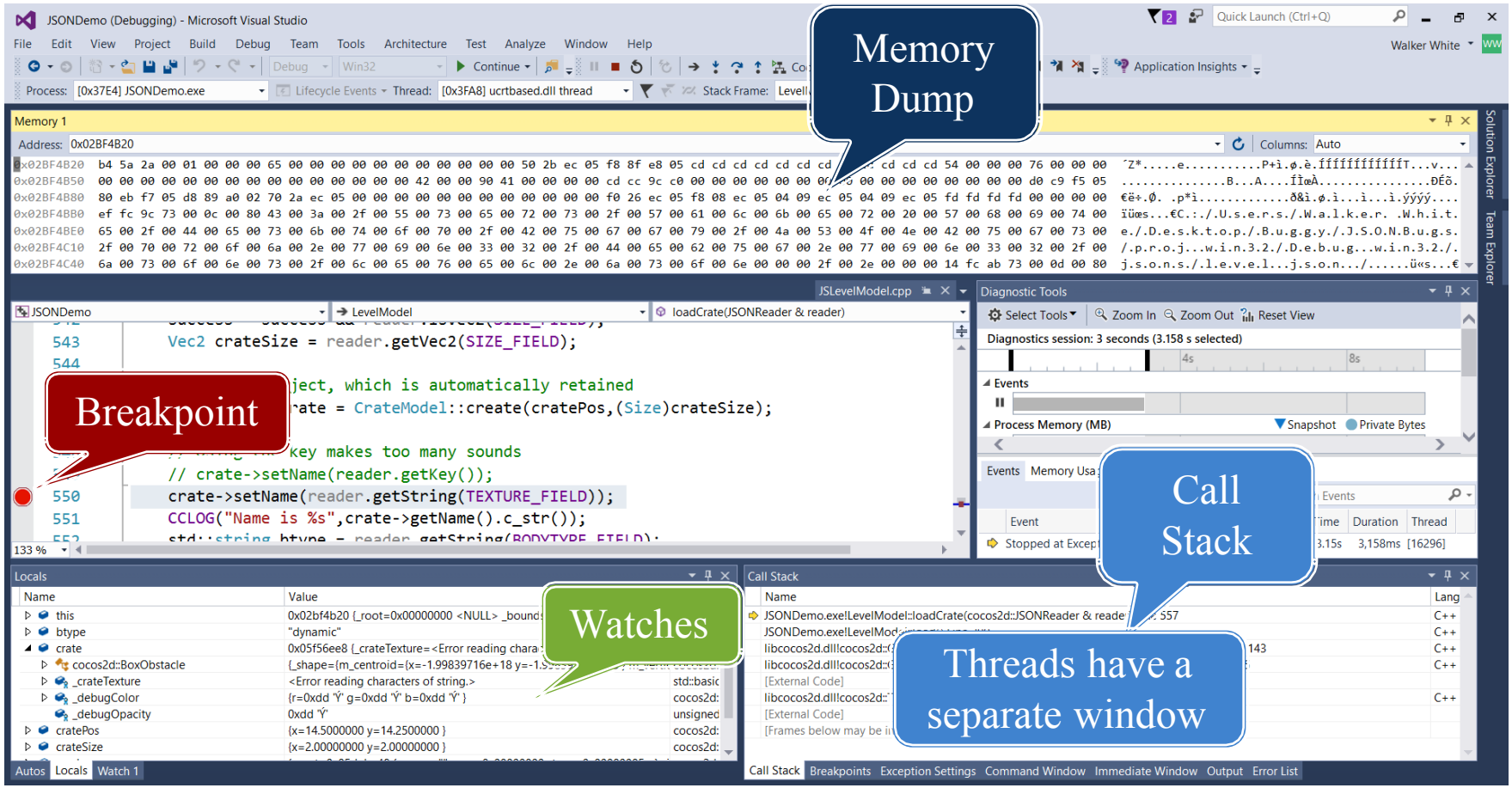
Visual Studio Tools



Visual Studio Tools



Visual Studio Tools



Breakpoint Strategies

- **Break early**
  - Break before the error, to check everything is okay
  - Step forward and watch how the code changes
- **Break infrequently**
  - If you always break, cannot initialize or animate anything
  - Design special conditionals for your breakpoint
- **Break on deletion**
  - Put breakpoints inside of all your destructors
  - Allows you to track accidental deletion

Problems with Code Stepping

- Code stepping is not "thread safe"
  - Will never leave your current thread
  - Have to choose "continue" instead of "step"
- Makes it very difficult to find thread errors
  - May miss when a variable changes state
  - We cannot find Win32 `AudioEngine` bug
- **Solution**: Rely heavily on assertions
  - Assert every variable shared across threads
  - Assert them everywhere they may change

Case Study: JSON Loading

- Problem was a thread *race condition*
  - Appeared on Windows, but not OS X
  - Because of particular Windows thread schedule
  - But technically unsafe on all platforms
- Found by putting **breakpoints in destructors**
  - Models getting deleted immediately after creation
  - Watched the reference counts to find problem
  - Autorelease pool was deleting before `retain()`

Case Study: b2BlockAllocator

- *Memory address* problem in Box2D engine
  - Problem was because we put Box2D in a DLL
  - Required stepping through the allocation process
  - Required **memory dumps** to view the heap
- Problem with the *static global variables*
  - DLLs have a distinct global space
  - BlockAllocator was initialized inside of the DLL
  - When it was used outside the DLL, not initialized