Automated Testing & Instant Replays in Retro City Rampage

Brian Provinciano | @BriProv
Vblank Entertainment

GAME DEVELOPERS CONFERENCE®
MOSCONE CENTER · SAN FRANCISCO, CA
MARCH 2-6, 2015 · EXPO: MARCH 4-6, 2015

Recipe

Ingredients

1 part **Recorded Player Input**

1 part **Deterministic Engine**

Directions

1. **Sit back and watch use cases**
2. **Sit back and watch the how to!**
3. **Ask questions**
4. **Download the source code**

PART I:
THE MANY USES OF RECORDED PLAYER INPUT

Use case #1
Automated QA with Pre-recorded Input

- Gives you piece of mind for every compile.
- Gives you access to QA 24/7.
- Allows hands-on QA be step #2.
  Let your automated playthroughs be the first line of defense.

PS4 PS3 PSVITA Wii NINTENDO 3DS XBOX 360


SAMSUNG

USE CASE #2

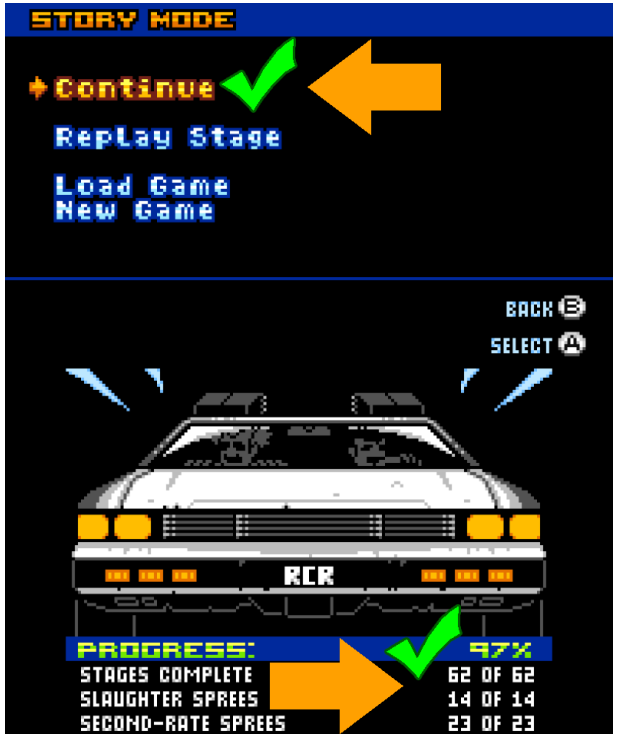
BUILT-IN AUTOMATED PLAYTHROUGHS

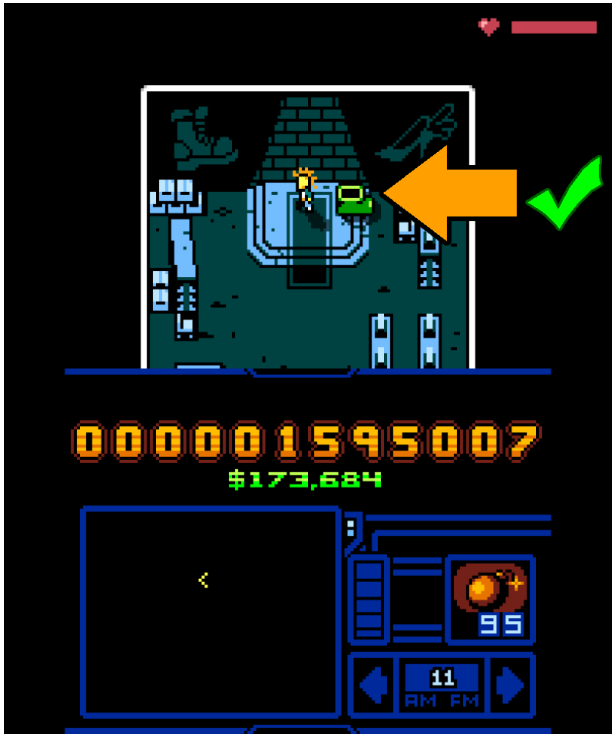
PS VITA:

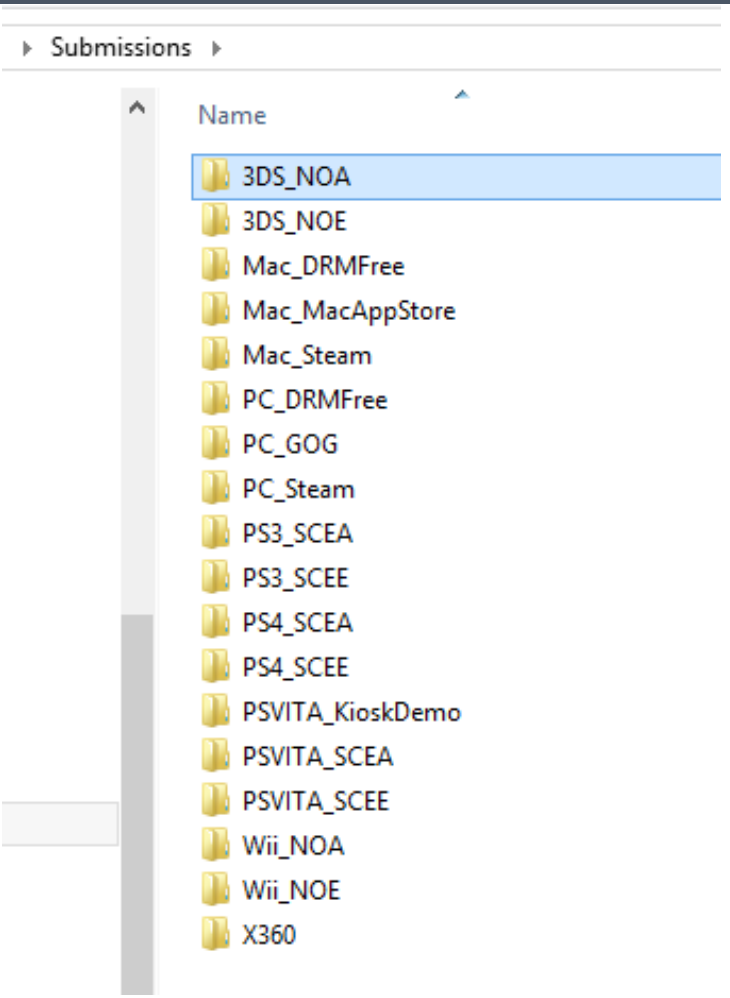
O O O R R R LEFT LEFT LEFT R O O

NINTENDO 3DS:

B B B R R R LEFT LEFT LEFT R B B







Use Case #3

Multiple SKUs, Certification & Re-Submissions

- Late 2014, single-handedly/simultaneously:
  - 15 SKUs, 9 console cert submissions
- Slashes the required redundant testing
  - Separate regions/Separate builds
  - Re-submissions

USE CASE #4
CATCHING GAMEPLAY BUGS

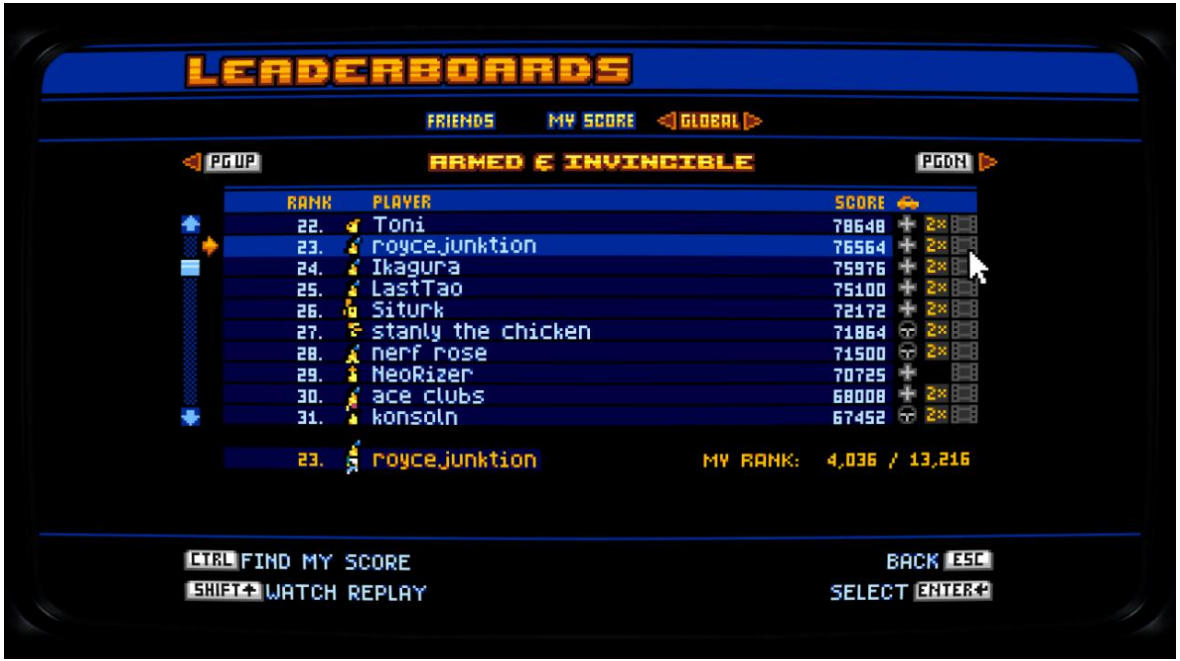
1. _autorec_.rec 10/8/2014 12:42 PM 1 KB

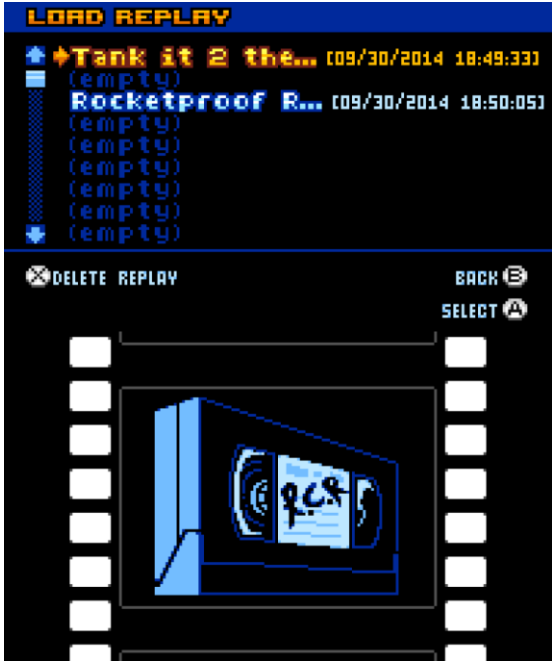
2. Debugger to launch: Local Windows Debugger

Command	$(TargetPath)
Command Arguments	-autoplayback
Working Directory	$(ProjectDir)_data
Attach	No
Debugger Type	Auto
Environment	
Merge Environment	Yes

3.
```
F32 angle = PointsToAngle(spr->x, spr->y, x, y);
if(spr->index==25 && game.framecounter>=91625)
{
    DEBUG_PRINTF("\n%08d: (%4d, %4d, %4d, %4d) -> %-3.3f\n", game.framecounter, spr->x, spr->y, x, y, angle);
}
```

USE CASE #5
USER REPLAYS



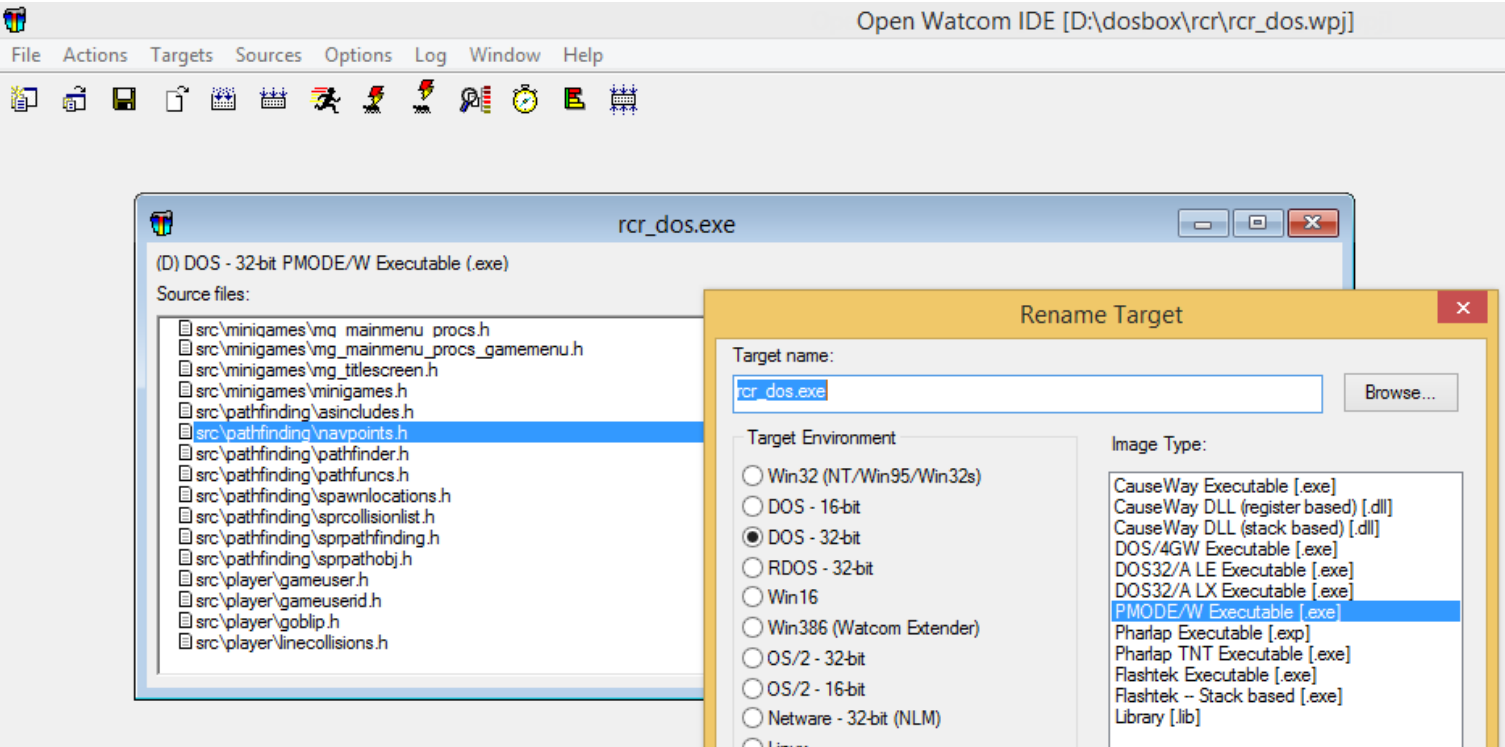


USE CASE #6

CONTROLLING NPCS DURING CUTSCENES

Use Case #7
Classic Arcade Attract Modes

USE CASE #8
TRIMMING MEMORY



Other Example Uses of Recorded Player Input

- Speeding up trailer/video creation
- Asynchronous multiplayer
- Improving remote QA/playtesting
- Generating analytics/heat maps from playtesters/PAX/E3 attendees

PART II:
RECORDING INPUT

```cpp
/*****************************************************************************/
// SIMPLE INPUT RECORD/PLAYBACK
// (c) 2015 Brian Provinciano
//
// You are free to use this code for your own purposes, no strings attached.
//
// This is a very basic sample to record and playback button input.
// It's most useful when activated on startup, deactivated on shutdown for
// global button recording/playback.
//
// For details on more advanced implementations, see my GDC 2015 session:
// -> Automated Testing and Instant Replays in Retro City Rampage
// The slides and full video will be available on the GDC Vault at a later date.
/*****************************************************************************/


/*****************************************************************************/
// wrap it so it can be conditionally compiled in.
// for example, set INPUTREPLAY_CAN_RECORD to 1 to play the game and record the input, set it to 0 when done
// INPUTREPLAY_CAN_RECORD takes priority over INPUTREPLAY_CAN_PLAYBACK

#define INPUTREPLAY_CAN_PLAYBACK    1
#define INPUTREPLAY_CAN_RECORD      1

#define INPUTREPLAY_INCLUDED        (INPUTREPLAY_CAN_PLAYBACK || INPUTREPLAY_CAN_RECORD)
/*****************************************************************************/

#if INPUTREPLAY_INCLUDED

#define INPUT_BUTTONS_TOTAL 32      // up to 32
#define MAX_REC_LEN         0x8000  // the buffer size for storing RLE compressed button input (x each button)
```

Recorded Input

- **Buttons** *(pressed / not pressed)*
- **Values** *(ie. analog sticks)*
- **Events** *(ie. mouse click, touch drag)*
- **Debug Information** *(optional)*
- **Time Deltas** *(if your game isn't fixed-frame)*

EXAMPLE: RECORDED INPUT IN RCR

- **Header** – standard header data & current state
  - control options, player outfit, camera zoom, etc.
- **Button streams**
  - 1 bit per frame for pressed/released
- **Analog/stick streams**
- **Event streams**
  - Mouse/Touch press/release/drag
- **Debug Information**
  - Checksums – Verbose mode for dev (checksum per-frame), light mode for shipping (per second). Game displays error if desync occurs.

Simple Compression For RCR

- RLE Button Streams
  - A separate stream per button = better compression
- zlib Entire File
  - Super easy/fast.

PAIR EVERYTHING FOR CONSISTENCY

```
#ifdef RECORD_PLAYER
    if(g_playerAutoRec.IsInitialized())
    {
        g_playerAutoRec.Update();
    }
    else
#endif
    {
        joypad.Update(systeminput.buttons, systeminput.ldirection, syste
    }
```

Part III:
Building a Deterministic Game

INITIALIZATION

- Obviously, everything must be properly initialized.

- Avoid static/global initialization.
  - Pay attention to static class constructors

```
static BOOL g_bSomethingInitializedGlobally = FALSE;
```

rand()

Implement and seed your own

```
class CarAudio
{
public:
    void Init();
    void Deinit();
    void Update();

    unsigned int m_randSndIdx;
    unsigned int m_honkDelay;
    RandomGenerator m_carAudioRand;
```

TIP: Use separate instances

For example, by using a separate Rand() for gameplay vs effects,
previously recorded replays are unlikely to break if effects are added/changed.

CALLBACKS/SYSTEM EVENTS

- For example: End of song callback
  - Don’t use the system/driver callback
  - Implement your own based on your ‘game ticks’

MIDDLEWARE

- Not the end of the world
- Investigate!

FLOATING POINT

- A key cause of non-portable replays
  - Single console – easier
  - PCs – more problems *(ie. AMD vs Intel, etc.)*
    - Manually specify precision if possible
    - Be careful of drivers
- Don’t focus on portability unless you need it

SAFER

MANUALLY USING LOW PRECISION LUTS

```
#define BCos(angleRad)      g_cossinTable[RAD_TO_TABLEIDX_SIN(angleRad)]
#define BSin(angleRad)      g_cossinTable[RAD_TO_TABLEIDX_COS(angleRad)]

const F32 g_cossinTable[COSSIN_TABLE_SIZE] =
{
    0.000000f,  // (0.0000 deg, 0.0000 rad)
    0.001534f,  // (0.0879 deg, 0.0015 rad)
    0.003068f,  // (0.1758 deg, 0.0031 rad)
    0.004602f,  // (0.2637 deg, 0.0046 rad)
    0.006136f,  // (0.3516 deg, 0.0061 rad)
    0.007670f,  // (0.4395 deg, 0.0077 rad)
    0.009204f,  // (0.5273 deg, 0.0092 rad)
    0.010738f,  // (0.6152 deg, 0.0107 rad)
    0.012272f,  // (0.7031 deg, 0.0123 rad)
    0.013805f,  // (0.7910 deg, 0.0138 rad)
    0.015339f,  // (0.8789 deg, 0.0153 rad)
    0.016873f,  // (0.9668 deg, 0.0169 rad)
    0.018407f,  // (1.0547 deg, 0.0184 rad)
    0.019940f,  // (1.1426 deg, 0.0199 rad)
```

LESS SAFE

CHAINED MULTI-FRAME CALCULATIONS

```
// run each frame until at destination
F32 maxratiosq = SQUARED(255);
F32 negratiosq = (maxratiosq-(distsq/maxratiosq))*0.85f;
F32 ratiosq = maxratiosq-negratiosq;
if(distsq<SQUARED(2))
{
    prop->x = player.spr->x;
}
else
{
    prop->x = F32_TO_U16((prop->x*negratiosq + player.spr->x*ratiosq) / maxratiosq);
}
```

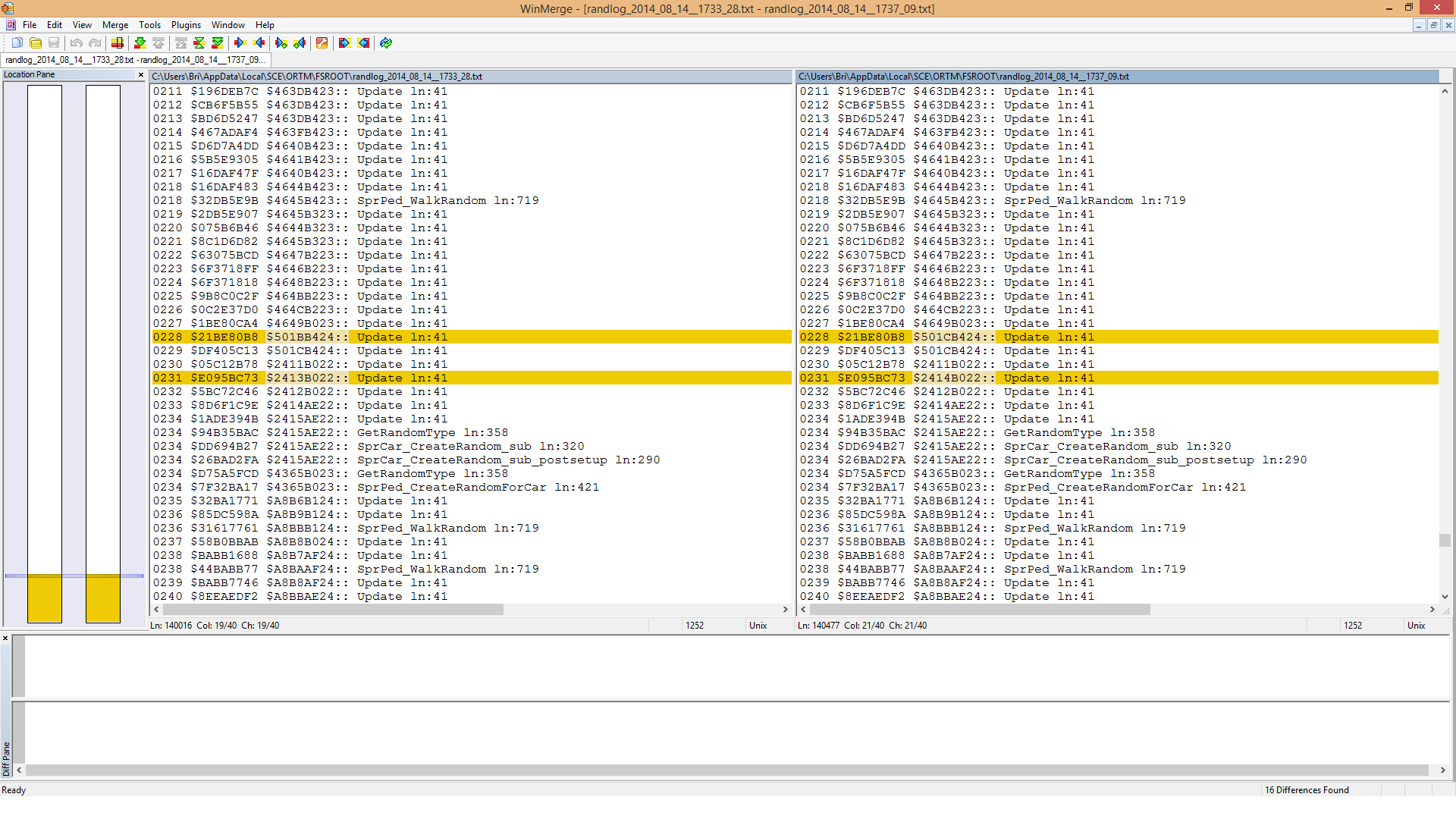
SANDBOXING

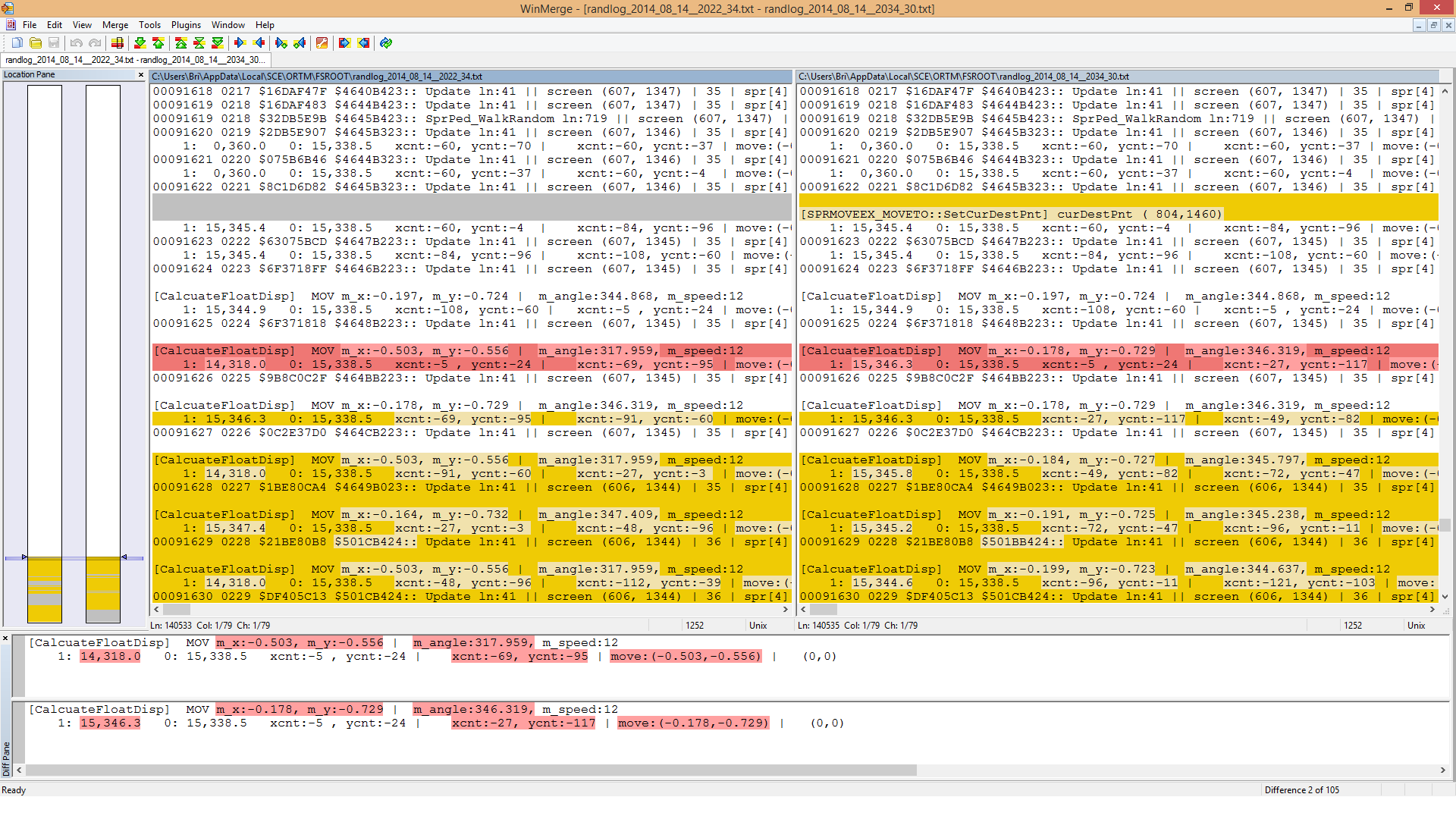
- No longer 100% code coverage, but 99% is still great!
- Leaderboards, Achievements, System Utilities, Online
  - Use fake leaderboads, fake achievements, fake online.
- Save Games *(file vs. in-memory)*
  - Ability to switch between *file-io* & *in-memory* save data protects user's save data in shipping build.

PART IV:
DEBUGGING IT ALL

Tracking Down Bugs

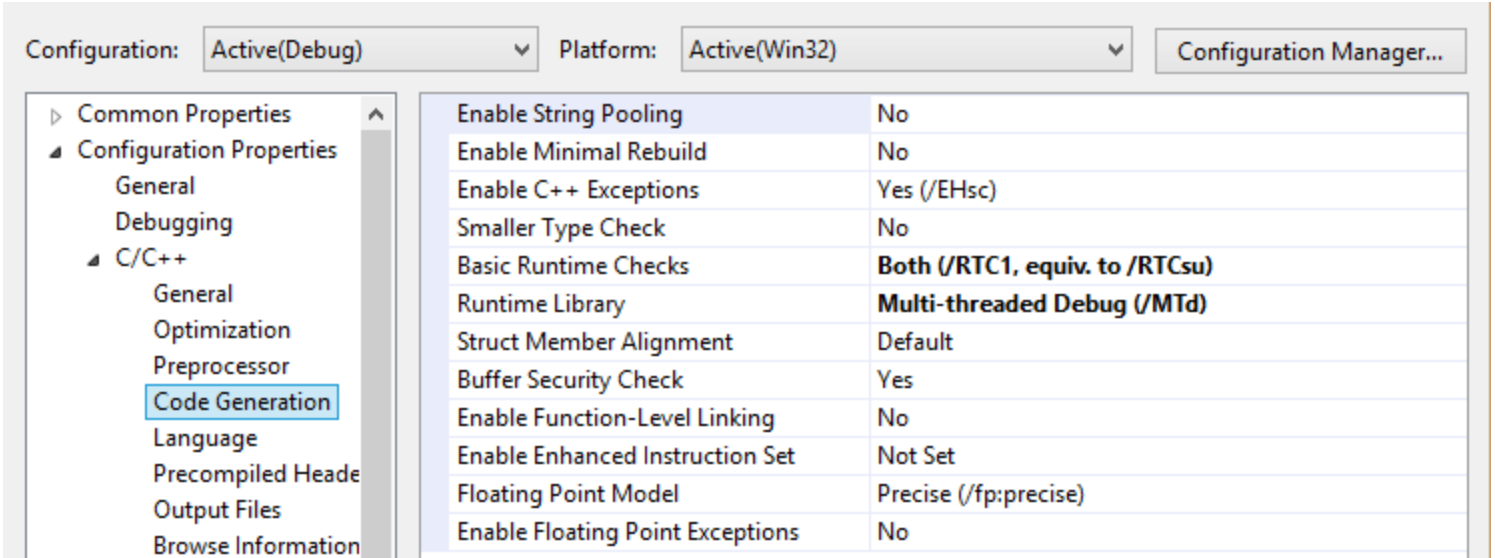
- Generate a log
  - Both, while recording, while playing back
- Grab a diff tool *(ie. WinMerge)*
- Comparing logs helps you track down issues when…
  - things playback incorrectly
  - things playback correctly on one system but not on another, or in Debug but not Release, for example


WinMerge - [randlog_2014_08_14__1733_28.txt - randlog_2014_08_14__1737_09.txt]
File Edit View Merge Tools Plugins Window Help
randlog_2014_08_14__1733_28.txt - randlog_2014_08_14__1737_09...
Location Pane
C:\Users\Bri\AppData\Local\SCE\ORTM\FSROOT\randlog_2014_08_14__1733_28.txt
0211 $196DEB7C $463DB423:: Update ln:41
0212 $CB6F5B55 $463DB423:: Update ln:41
0213 $BD6D5247 $463DB423:: Update ln:41
0214 $467ADAF4 $463FB423:: Update ln:41
0215 $D6D7A4DD $4640B423:: Update ln:41
0216 $5B5E9305 $4641B423:: Update ln:41
0217 $16DAF47F $4640B423:: Update ln:41
0218 $16DAF483 $4644B423:: Update ln:41
0218 $32DB5E9B $4645B423:: SprPed_WalkRandom ln:719
0219 $2DB5E907 $4645B323:: Update ln:41
0220 $075B6B46 $4644B323:: Update ln:41
0221 $8C1D6D82 $4645B323:: Update ln:41
0222 $63075BCD $4647B223:: Update ln:41
0223 $6F3718FF $4646B223:: Update ln:41
0224 $6F371818 $4648B223:: Update ln:41
0225 $9B8C0C2F $464BB223:: Update ln:41
0226 $0C2E37D0 $464CB223:: Update ln:41
0227 $1BE80CA4 $4649B023:: Update ln:41
0228 $21BE80B8 $501BB424:: Update ln:41
0229 $DF405C13 $501CB424:: Update ln:41
0230 $05C12B78 $2411B022:: Update ln:41
0231 $E095BC73 $2413B022:: Update ln:41
0232 $5BC72C46 $2412B022:: Update ln:41
0233 $8D6F1C9E $2414AE22:: Update ln:41
0234 $1ADE394B $2415AE22:: Update ln:41
0234 $94B35BAC $2415AE22:: GetRandomType ln:358
0234 $DD694B27 $2415AE22:: SprCar_CreateRandom_sub ln:320
0234 $26BAD2FA $2415AE22:: SprCar_CreateRandom_sub_postsetup ln:290
0234 $D75A5FCD $4365B023:: GetRandomType ln:358
0234 $7F32BA17 $4365B023:: SprPed_CreateRandomForCar ln:421
0235 $32BA1771 $A8B6B124:: Update ln:41
0236 $85DC598A $A8B9B124:: Update ln:41
0236 $31617761 $A8BBB124:: SprPed_WalkRandom ln:719
0237 $58B0BBAB $A8B8B024:: Update ln:41
0238 $BABB1688 $A8B7AF24:: Update ln:41
0238 $44BABB77 $A8BAAF24:: SprPed_WalkRandom ln:719
0239 $BABB7746 $A8B8AF24:: Update ln:41
0240 $8EEAEDF2 $A8BBAE24:: Update ln:41
Ln: 140016 Col: 19/40 Ch: 19/40
1252
Unix
C:\Users\Bri\AppData\Local\SCE\ORTM\FSROOT\randlog_2014_08_14__1737_09.txt
0211 $196DEB7C $463DB423:: Update ln:41
0212 $CB6F5B55 $463DB423:: Update ln:41
0213 $BD6D5247 $463DB423:: Update ln:41
0214 $467ADAF4 $463FB423:: Update ln:41
0215 $D6D7A4DD $4640B423:: Update ln:41
0216 $5B5E9305 $4641B423:: Update ln:41
0217 $16DAF47F $4640B423:: Update ln:41
0218 $16DAF483 $4644B423:: Update ln:41
0218 $32DB5E9B $4645B423:: SprPed_WalkRandom ln:719
0219 $2DB5E907 $4645B323:: Update ln:41
0220 $075B6B46 $4644B323:: Update ln:41
0221 $8C1D6D82 $4645B323:: Update ln:41
0222 $63075BCD $4647B223:: Update ln:41
0223 $6F3718FF $4646B223:: Update ln:41
0224 $6F371818 $4648B223:: Update ln:41
0225 $9B8C0C2F $464BB223:: Update ln:41
0226 $0C2E37D0 $464CB223:: Update ln:41
0227 $1BE80CA4 $4649B023:: Update ln:41
0228 $21BE80B8 $501CB424:: Update ln:41
0229 $DF405C13 $501CB424:: Update ln:41
0230 $05C12B78 $2411B022:: Update ln:41
0231 $E095BC73 $2414B022:: Update ln:41
0232 $5BC72C46 $2412B022:: Update ln:41
0233 $8D6F1C9E $2414AE22:: Update ln:41
0234 $1ADE394B $2415AE22:: Update ln:41
0234 $94B35BAC $2415AE22:: GetRandomType ln:358
0234 $DD694B27 $2415AE22:: SprCar_CreateRandom_sub ln:320
0234 $26BAD2FA $2415AE22:: SprCar_CreateRandom_sub_postsetup ln:290
0234 $D75A5FCD $4365B023:: GetRandomType ln:358
0234 $7F32BA17 $4365B023:: SprPed_CreateRandomForCar ln:421
0235 $32BA1771 $A8B6B124:: Update ln:41
0236 $85DC598A $A8B9B124:: Update ln:41
0236 $31617761 $A8BBB124:: SprPed_WalkRandom ln:719
0237 $58B0BBAB $A8B8B024:: Update ln:41
0238 $BABB1688 $A8B7AF24:: Update ln:41
0238 $44BABB77 $A8BAAF24:: SprPed_WalkRandom ln:719
0239 $BABB7746 $A8B8AF24:: Update ln:41
0240 $8EEAEDF2 $A8BBAE24:: Update ln:41
Ln: 140477 Col: 21/40 Ch: 21/40
1252
Unix
Diff Pane
Ready
16 Differences Found


WinMerge - [randlog_2014_08_14__2022_34.txt - randlog_2014_08_14__2034_30.txt]
File Edit View Merge Tools Plugins Window Help
randlog_2014_08_14__2022_34.txt - randlog_2014_08_14__2034_30...
Location Pane
C:\Users\Bri\AppData\Local\SCE\ORTM\FSROOT\randlog_2014_08_14__2022_34.txt
C:\Users\Bri\AppData\Local\SCE\ORTM\FSROOT\randlog_2014_08_14__2034_30.txt
00091618 0217 $16DAF47F $4640B423:: Update ln:41 || screen (607, 1347) | 35 | spr[4]
00091619 0218 $16DAF483 $4644B423:: Update ln:41 || screen (607, 1347) | 35 | spr[4]
00091619 0218 $32DB5E9B $4645B423:: SprPed_WalkRandom ln:719 || screen (607, 1347) |
00091620 0219 $2DB5E907 $4645B323:: Update ln:41 || screen (607, 1346) | 35 | spr[4]
1: 0,360.0 0: 15,338.5 xcnt:-60, ycnt:-70 | xcnt:-60, ycnt:-37 | move:(-
00091621 0220 $075B6B46 $4644B323:: Update ln:41 || screen (607, 1346) | 35 | spr[4]
1: 0,360.0 0: 15,338.5 xcnt:-60, ycnt:-37 | xcnt:-60, ycnt:-4 | move:(-
00091622 0221 $8C1D6D82 $4645B323:: Update ln:41 || screen (607, 1346) | 35 | spr[4]
[SPRMOVEEX_MOVETO::SetCurDestPnt] curDestPnt (804,1460)
1: 15,345.4 0: 15,338.5 xcnt:-60, ycnt:-4 | xcnt:-84, ycnt:-96 | move:(-
00091623 0222 $63075BCD $4647B223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
1: 15,345.4 0: 15,338.5 xcnt:-84, ycnt:-96 | xcnt:-108, ycnt:-60 | move:(-
00091624 0223 $6F3718FF $4646B223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.197, m_y:-0.724 | m_angle:344.868, m_speed:12
1: 15,344.9 0: 15,338.5 xcnt:-108, ycnt:-60 | xcnt:-5 , ycnt:-24 | move:(-
00091625 0224 $6F371818 $4648B223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.503, m_y:-0.556 | m_angle:317.959, m_speed:12
1: 14,318.0 0: 15,338.5 xcnt:-5 , ycnt:-24 | xcnt:-69, ycnt:-95 | move:(-
00091626 0225 $9B8C0C2F $464BB223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.178, m_y:-0.729 | m_angle:346.319, m_speed:12
1: 15,346.3 0: 15,338.5 xcnt:-69, ycnt:-95 | xcnt:-91, ycnt:-60 | move:(-
00091627 0226 $0C2E37D0 $464CB223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.503, m_y:-0.556 | m_angle:317.959, m_speed:12
1: 14,318.0 0: 15,338.5 xcnt:-91, ycnt:-60 | xcnt:-27, ycnt:-3 | move:(-
00091628 0227 $1BE80CA4 $4649B023:: Update ln:41 || screen (606, 1344) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.164, m_y:-0.732 | m_angle:347.409, m_speed:12
1: 15,347.4 0: 15,338.5 xcnt:-27, ycnt:-3 | xcnt:-48, ycnt:-96 | move:(-
00091629 0228 $21BE80B8 $501CB424:: Update ln:41 || screen (606, 1344) | 36 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.503, m_y:-0.556 | m_angle:317.959, m_speed:12
1: 14,318.0 0: 15,338.5 xcnt:-48, ycnt:-96 | xcnt:-112, ycnt:-39 | move:(-
00091630 0229 $DF405C13 $501CB424:: Update ln:41 || screen (606, 1344) | 36 | spr[4]
00091618 0217 $16DAF47F $4640B423:: Update ln:41 || screen (607, 1347) | 35 | spr[4]
00091619 0218 $16DAF483 $4644B423:: Update ln:41 || screen (607, 1347) | 35 | spr[4]
00091619 0218 $32DB5E9B $4645B423:: SprPed_WalkRandom ln:719 || screen (607, 1347) |
00091620 0219 $2DB5E907 $4645B323:: Update ln:41 || screen (607, 1346) | 35 | spr[4]
1: 0,360.0 0: 15,338.5 xcnt:-60, ycnt:-70 | xcnt:-60, ycnt:-37 | move:(-
00091621 0220 $075B6B46 $4644B323:: Update ln:41 || screen (607, 1346) | 35 | spr[4]
1: 0,360.0 0: 15,338.5 xcnt:-60, ycnt:-37 | xcnt:-60, ycnt:-4 | move:(-
00091622 0221 $8C1D6D82 $4645B323:: Update ln:41 || screen (607, 1346) | 35 | spr[4]
[SPRMOVEEX_MOVETO::SetCurDestPnt] curDestPnt (804,1460)
1: 15,345.4 0: 15,338.5 xcnt:-60, ycnt:-4 | xcnt:-84, ycnt:-96 | move:(-
00091623 0222 $63075BCD $4647B223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
1: 15,345.4 0: 15,338.5 xcnt:-84, ycnt:-96 | xcnt:-108, ycnt:-60 | move:(-
00091624 0223 $6F3718FF $4646B223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.197, m_y:-0.724 | m_angle:344.868, m_speed:12
1: 15,344.9 0: 15,338.5 xcnt:-108, ycnt:-60 | xcnt:-5 , ycnt:-24 | move:(-
00091625 0224 $6F371818 $4648B223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.178, m_y:-0.729 | m_angle:346.319, m_speed:12
1: 15,346.3 0: 15,338.5 xcnt:-5 , ycnt:-24 | xcnt:-27, ycnt:-117 | move:(-
00091626 0225 $9B8C0C2F $464BB223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.178, m_y:-0.729 | m_angle:346.319, m_speed:12
1: 15,346.3 0: 15,338.5 xcnt:-27, ycnt:-117 | xcnt:-49, ycnt:-82 | move:(-
00091627 0226 $0C2E37D0 $464CB223:: Update ln:41 || screen (607, 1345) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.184, m_y:-0.727 | m_angle:345.797, m_speed:12
1: 15,345.8 0: 15,338.5 xcnt:-49, ycnt:-82 | xcnt:-72, ycnt:-47 | move:(-
00091628 0227 $1BE80CA4 $4649B023:: Update ln:41 || screen (606, 1344) | 35 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.191, m_y:-0.725 | m_angle:345.238, m_speed:12
1: 15,345.2 0: 15,338.5 xcnt:-72, ycnt:-47 | xcnt:-96, ycnt:-11 | move:(-
00091629 0228 $21BE80B8 $501BB424:: Update ln:41 || screen (606, 1344) | 36 | spr[4]
[CalcuateFloatDisp] MOV m_x:-0.199, m_y:-0.723 | m_angle:344.637, m_speed:12
1: 15,344.6 0: 15,338.5 xcnt:-96, ycnt:-11 | xcnt:-121, ycnt:-103 | move:
00091630 0229 $DF405C13 $501CB424:: Update ln:41 || screen (606, 1344) | 36 | spr[4]
Ln: 140533 Col: 1/79 Ch: 1/79 1252 Unix
Ln: 140535 Col: 1/79 Ch: 1/79 1252 Unix
[CalcuateFloatDisp] MOV m_x:-0.503, m_y:-0.556 | m_angle:317.959, m_speed:12
1: 14,318.0 0: 15,338.5 xcnt:-5 , ycnt:-24 | xcnt:-69, ycnt:-95 | move:(-0.503,-0.556) | (0,0)
[CalcuateFloatDisp] MOV m_x:-0.178, m_y:-0.729 | m_angle:346.319, m_speed:12
1: 15,346.3 0: 15,338.5 xcnt:-5 , ycnt:-24 | xcnt:-27, ycnt:-117 | move:(-0.178,-0.729) | (0,0)
Diff Pane
Ready
Difference 2 of 105

BULLETPROOFING

- Keep Rendering and the Game Loop Separate
- Run Static Analyzers, Sanitizers, etc.
- Create Two Project Configs, Record/Replay Between Each:
  - 32/64-bit, Debug/Release(with debugging information), Different Struct Alignment, Floating Point Model, etc.



PART V:
BUILDING BLOCKS

BATCH FILE GLOBAL AUTOMATED PLAYTHROUGHS

```
@ECHO OFF
REM clear save games so the recordings can work cleanly
@ECHO ON

del rcr.cfg /a
del *.rsv /a
del *.dat /a

retrocityrampage.exe -gamespeed 180 -playback pc_pt_1.rec
IF NOT ERRORLEVEL 0 GOTO ERROREND

retrocityrampage.exe -gamespeed 180 -playback pc_pt_2.rec
IF NOT ERRORLEVEL 0 GOTO ERROREND

retrocityrampage.exe -gamespeed 180 -playback pc_pt_3.rec
IF NOT ERRORLEVEL 0 GOTO ERROREND

retrocityrampage.exe -gamespeed 180 -playback pc_pt_4.rec
IF NOT ERRORLEVEL 0 GOTO ERROREND

retrocityrampage.exe -gamespeed 180 -playback pc_pt_5.rec
IF NOT ERRORLEVEL 0 GOTO ERROREND

retrocityrampage.exe -gamespeed 180 -playback pc_pt_6.rec
IF NOT ERRORLEVEL 0 GOTO ERROREND
pause
```

Self-contained Automated Playthroughs

- Engine must properly deinit, re-init
  - WIN-WIN – results in a more robust codebase.
- Use of in-memory save game recommended
  - Doesn't affect user's real save data.
- For RCR, loops from replay_1.rec … until file not found, then returns control to user.

Level Replays

- **All about state**

- **Simple solution:**
  1. Reset Level State *(always)*
  2. **IF Recording ->** Start Recording
     ELSE IF Playing Back -> Start Playing Back

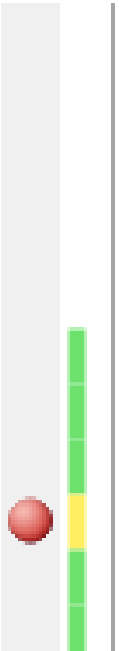
Entity Control

- **All about state**

- **Simple solution:**
  1. Reset Entity State *(always)*
  2. **IF Recording ->** Start Recording
     ELSE IF Playing Back -> Start Playing Back

BACKWARDS COMPATIBILITY

After shipping, you must maintain old code



```
m_recLength = m_hdr.lengthInFrames;
m_recLengthCnt = m_recLength;
m_recLengthFrameNum = 0;

m_bLoadedRecording = TRUE;

if(m_hdr.rectype!=eRecType_ObjControl)
{
    SetRadiusBugEnabled(m_hdr.version <= 8);
}
```

PART VI: SUMMARY!

SUMMARY

- Try even if your game isn't fully deterministic.
  - You have the code already: http://bripro.com/gdc/SimpleInputRec.cpp
- Start simple, build up.

Automated Testing & Instant Replays in Retro City Rampage

Brian Provinciano | @BriProv
Vblank Entertainment

SAMPLE: http://bripro.com/gdc/SimpleInputRec.cpp

GAME DEVELOPERS CONFERENCE®
MOSCONE CENTER · SAN FRANCISCO, CA
MARCH 2-6, 2015 · EXPO: MARCH 4-6, 2015