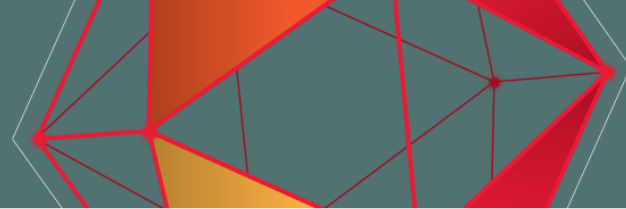




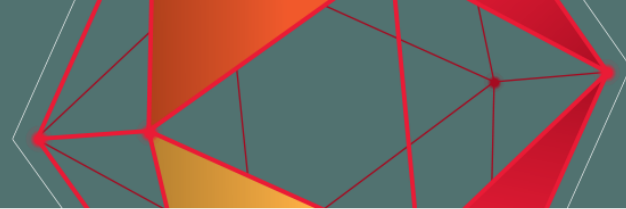
Tools to Reduce Open Bug Count at Media Molecule

Amy Phillips
High Priestess of Tools Code (Tools
Programmer) at Media Molecule
amy@mediamolecule.com



- Less open bugs -> more stable build -> fun bit of making games!
- Connecting off the shelf tools together – advantages of bespoke but less cost
- Dreams
 - UGC – kinda like LBP but so much more!
 - Sculpting, stamping, characters, logic, animation, music, audio,...
 - Media Molecule has ~50 employees (Daniel Kidney and I wrote this stuff – 1.4 progs! Now 2.4 with Alex Parker)
 - Bang for buck
- Studio environment means tools must be frictionless or they don't get used – these are in regular use

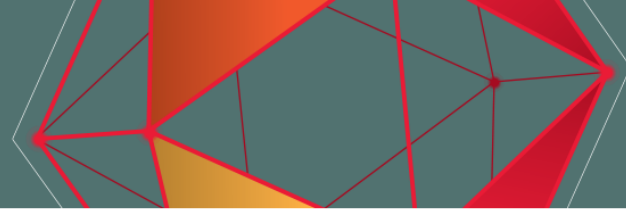




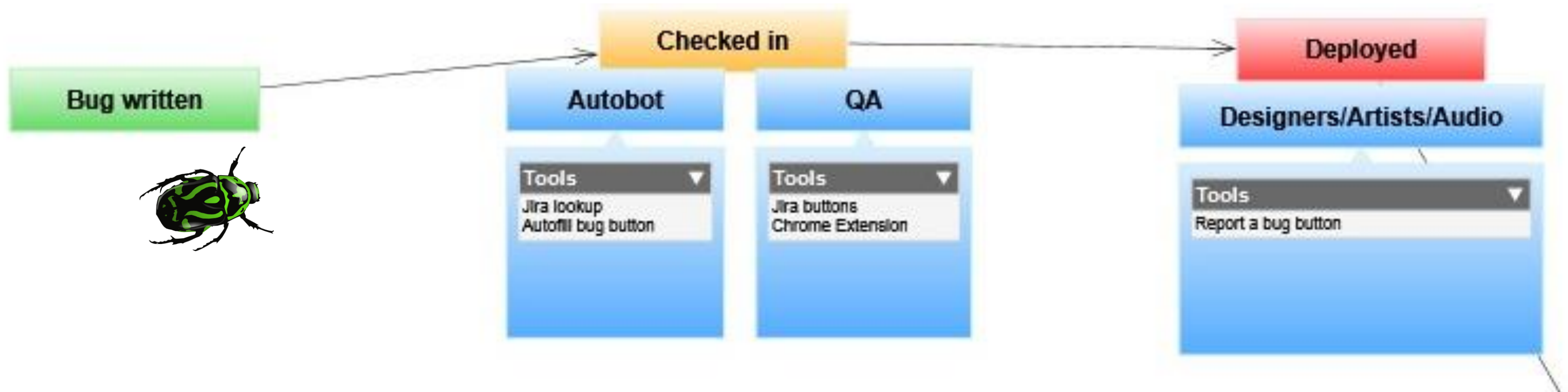
Dreams video

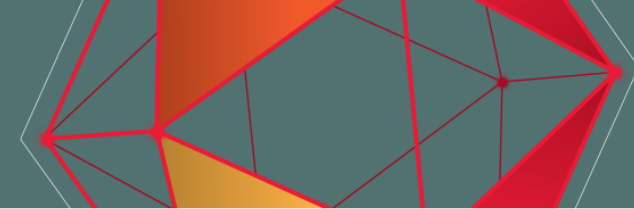
See <https://youtu.be/c-i0PcfboGc> for the Dreams Trailer shown here



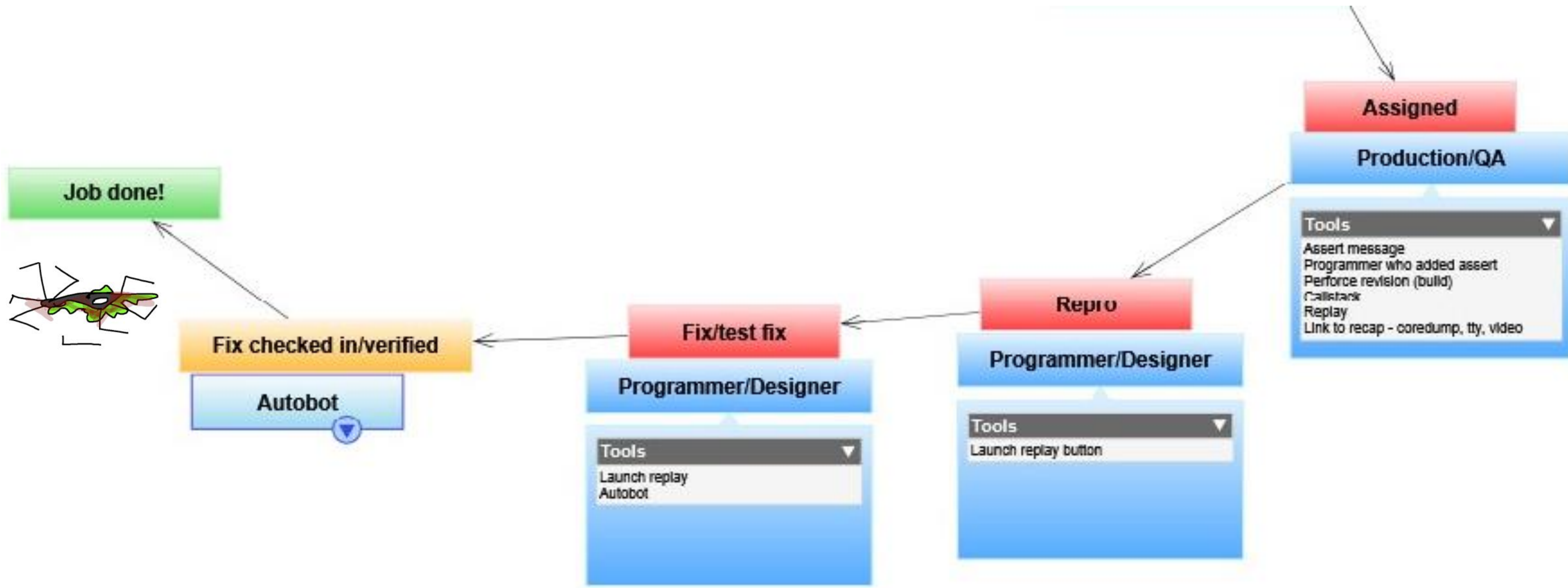


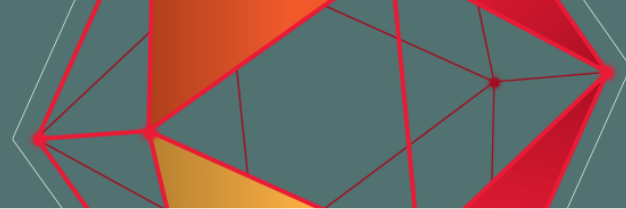
Life Cycle of a Bug (1)





Life Cycle of a Bug (2)





Detecting bugs – Autobot - Functionality

- Runs after every checkin
 - Checks for asserts, hangs, crashes, out of syncs, nans, logic errors
 - More in depth tests are run overnight
- Results are emailed
 - What has changed – did I break everything? (In email subject)
 - Assert message, callstack, link to existing bug for this issue (colour coded)
 - Previous results – when did it break?





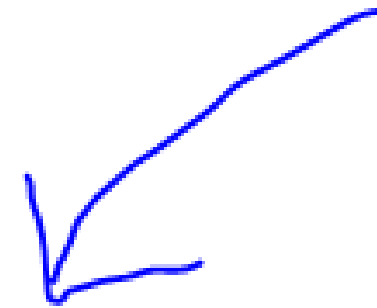
Mon 05/03/2018 19:22

autobot@mediamolecule.com

[r85486-r85488] AutoBasicSmoke ASSERT[+1]

To Mark Zarb-Adami; Paolo Carabaich; BumbleBee

You forwarded this message on 05/03/2018 21:24.



Found 1 new assert messages since last test causing an extra 88 tests to fail. Doh! See [Assert Messages](#) section for more info.

Summary Report 2018-03-05T19:21:52.540086

SuiteRun:

[AutoBasicSmoke 03-05 17:48:16](#)

Revision:

[r85488](#)

Last Revision:

[r85485](#)



Assert Messages

NEW

C:\dev\unicorn_test\dd\src\gfx\driver\gfx.cpp(521): ASSERT ok
out of GpuLong mem!
<http://p4swarm.medmol.local/files/mmdepot/code/trunk/games/dd/src/gfx/driver/gfx.cpp#521>

DebugLogStacktrace()
C:/dev/unicorn_test/dd/src/core/core.cpp:379
AssertLogV(char const*, ...)
C:/dev/unicorn_test/dd/src/debug/debuglog.cpp:238
MemWrapper::Alloc(EMemType, GfxHeapName, unsigned int, unsigned int)
C:/dev/unicorn_test/dd/src/gfx/driver/gfx.cpp:521
GfxTexture::CreateFromSpec(sce::Gnm::TextureSpec const&, sce::Gnm::ResourceMemoryType)
C:/dev/unicorn_test/dd/src/gfx/driver/gfx.cpp:1250
GfxTexture::CreateRW2D(unsigned int, unsigned int, unsigned int, unsigned int, EGfxFormat, sce::Gnm::ArrayMode)
C:/dev/unicorn_test/dd/src/gfx/driver/gfx.cpp:0
PrimRenderGlass(Environment const&, RenderTarget&, DSRenderTarget&)
C:/dev/unicorn_test/dd/src/ui/prim_glass.cpp:1507
SillyRender(Game const*, Trans)
C:/dev/unicorn_test/dd/src/game/renderer.cpp:802
RenderFrame(Game const*, RenderFeedbackSeq&)
C:/dev/unicorn_test/dd/src/game/renderer.cpp:1191
main
C:/dev/unicorn_test/dd/src/game/main.cpp:1482
_start
??:0

PlanName	2018-03-05 18:45:04.157270	2018-03-05 18:53:26.560505	2018-03-05 19:01:52.769121	2018-03-05 19:10:56.713510	2018-03-05 19:21:52.540086
0245_L_DS4_PS_Basicsmoke_Gadgets_Joints-2018-01-24_16-19-54 -r \optimusprime\ReplayStorage\3fb030b11066afc1-038d19baab92a534 -c -s \optimusprime\ReplayScreenshots\debug\amy_test_loc	success	success	success	success	failed same assert
0285_G_DS4_AP_Basicsmoke_Timeline_PlayPause-2017-11-27_15-04-08 -r \optimusprime\ReplayStorage\59d82c7ba0b409650-7dec84717057cebe -c -s \optimusprime\ReplayScreenshots\debug\amy_test_loc	success	success	success	success	failed same assert

r85482:

r85482:

r85483:
mmalexparker
[\[r85483,r85482\]](#)

r85485:
mmalexparker
[\[r85483\]](#)

r85488:
mmmarkz
[\[r85487,r85486\]](#)
mmpaolo
[\[r85488\]](#)



Detecting bugs - Autobot – more functionality

- Results also available/searchable on web UI
 - Bug report button for QA prefills
 - Assert message, programmer who added assert
 - Perforce revision – uniquely identifies a build
 - Callstack
 - Replay is copied and attached to bug
 - Link to recap session – coredump, tty, video
 - Description
 - TODO – autofill which revision it broke in, and who checked that in!
 - Prefilling FTW
 - removes copy-paste errors
 - Ensures consistent/complete set of information
 - Saves QA time
 - Launch replay button
 - Check a replay is testing what you think it is testing – screenshot, json gamestate





<input type="checkbox"/>	VALID	ASSERT	SUCCESS	JSON	REPORT	NANCHECK	FLGD	BUILD	PLANNAME	BUTTONS
<input type="checkbox"/>	✓	✗	✗	✗	✗	✓	u	r80001	RGT_A_text_gadget_outputs_test	<div>Launch</div> <div>JIRA</div>
<input type="checkbox"/>	✓	✗	✗	✗	✗	✓	u	r80001	RGT_B_D-Bug_health_death_checkpoint-2017-10-18_12-39-51	<div>Launch</div> <div>Bug</div>

FILTER

By TestTime

- Any date
- Today
- Past 7 days
- This month
- This year

By Valid

- All
- Yes
- No

By Assertyness

- All
- No Assert
- Assert Failed

By Success

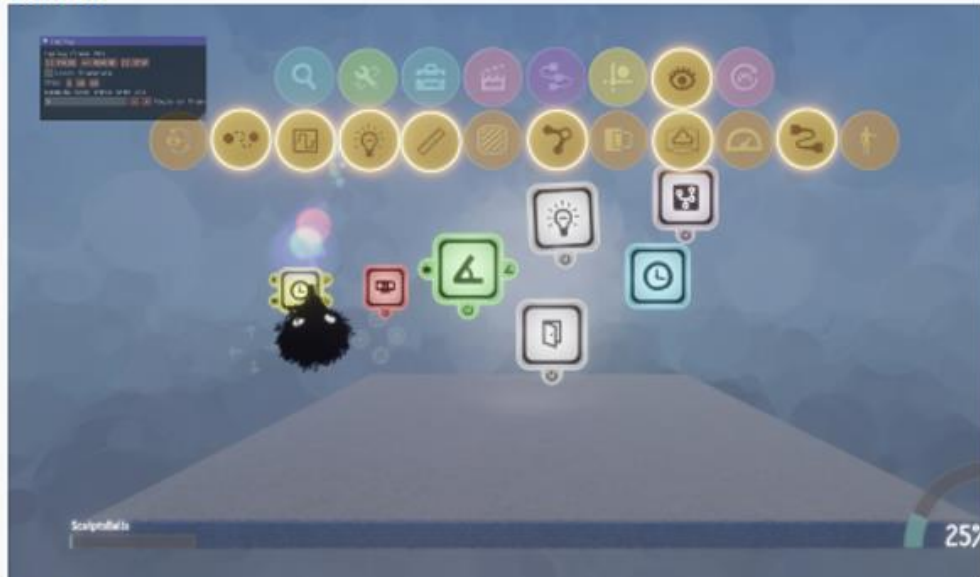

- All
- Yes
- No

By JSON Comparison

- All
- Same
- JSON different



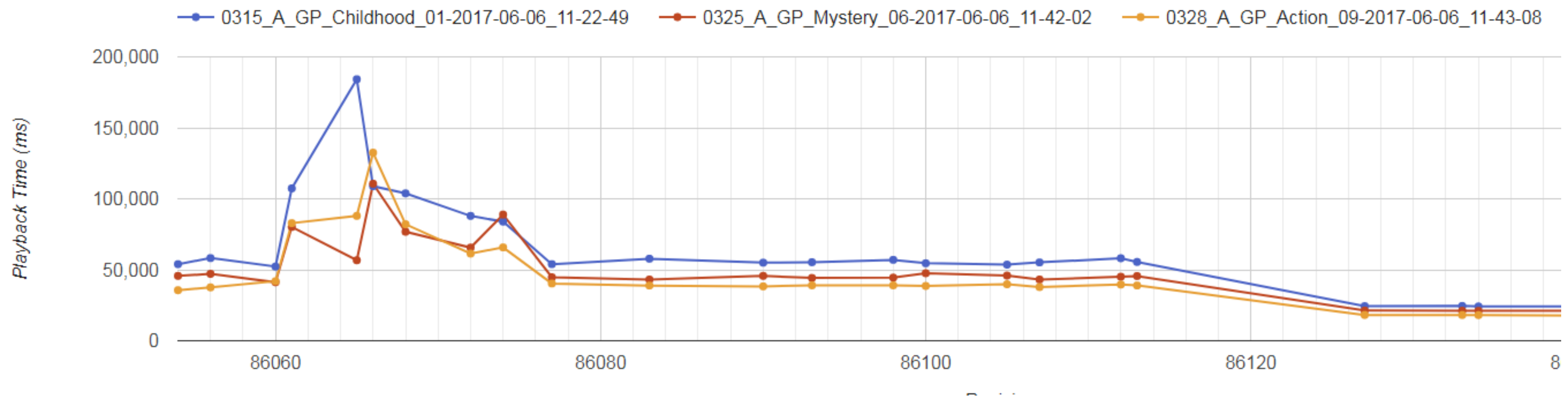


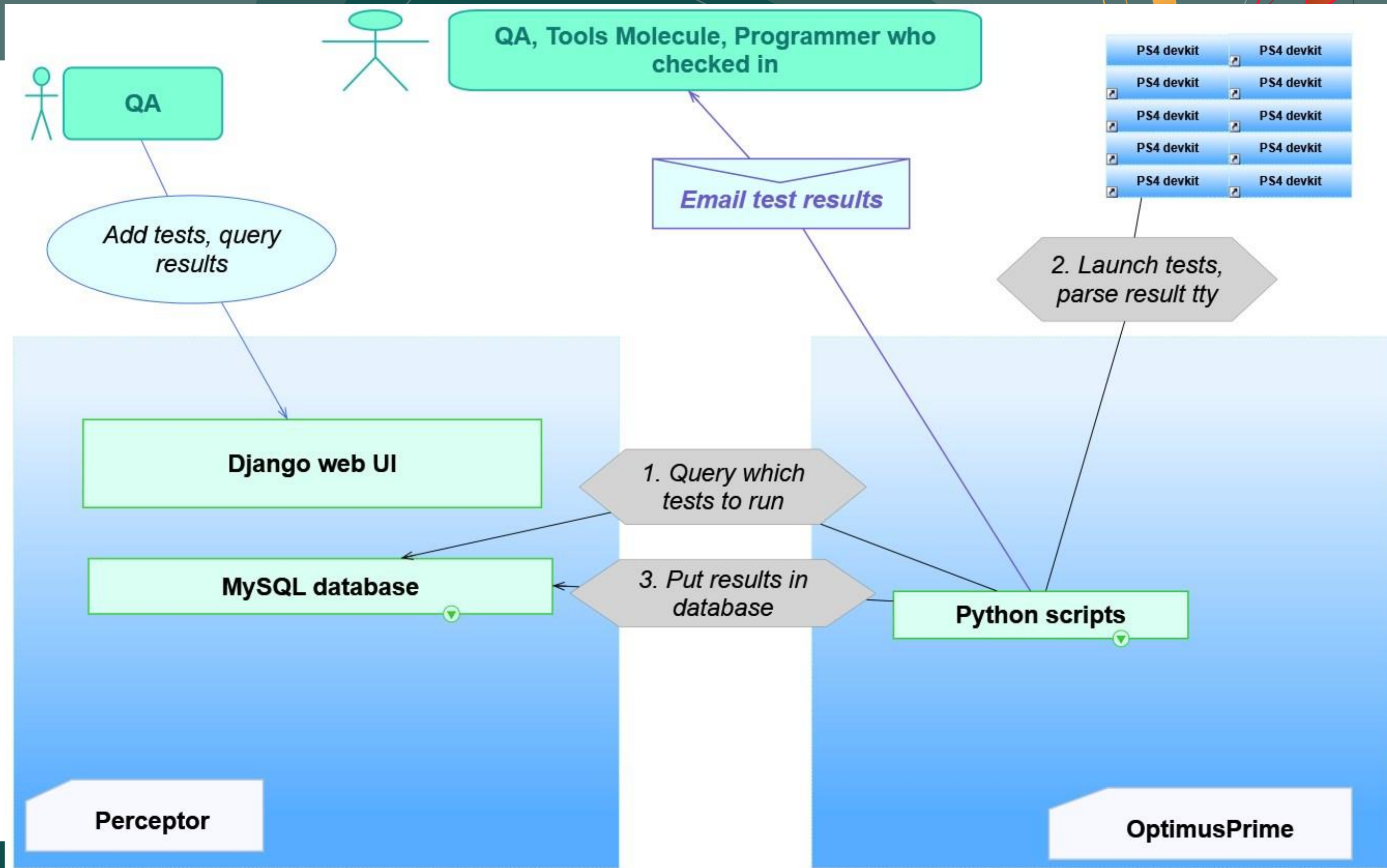
<input type="checkbox"/>	VALID	ASSERT	SUCCESS	JSON	REPORT	NANCHECK	FLGD	BUILD	PLANNAME	BUTTONS	LAST SCREENSHOT
<input type="checkbox"/>	✓	✓	✓	✓	✓	✓	u	r80114	0205_E_DS4_AP_Basicsmoke_Filters_Gadgets-2017-10-23_12-20-09	Launch	<p>control:</p>  <p>this run:</p> 

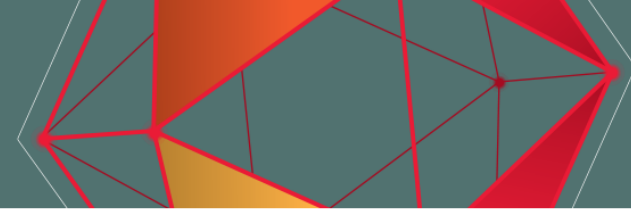




Performance Summary



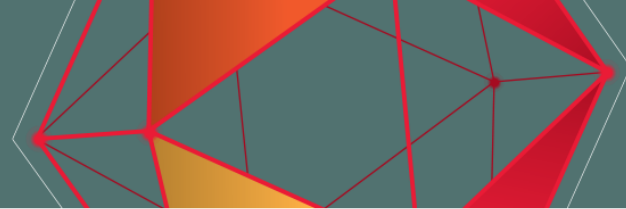




Detecting bugs – Autobot Tech

- Django with MySQL database - perceptor
 - Django admin pages as UI to display/alter database data
- Database stores
 - Which replays are run for each test suite – smoke, sculpt, gadgets,...
 - Results for test runs
- Python scripts on windows – optimusprime
- 27 devkits
- Custom PS4Launcher exe – TMAPI
 - Timeout for hangs – used to halt entire system
 - Upload coredumps
 - Bonus features – can change devkit settings – language, Japanese buttons, PSPro





Detecting bugs – Replay System

- Game update deterministic
- Replay consists of initial gamestate and inputs (includes audio import, text entry etc, not just controller input). If you change level we store gamestate you loaded.
- Serialise revision
 - Incremented more than you might expect
 - Branching joy – not!
- Repro bugs





Detecting Bugs - QA

- Chrome extension
 - Javascript, Rest APIs, tabs, cookies, notifications
- Jira buttons
 - Attach screenshot – one click
 - Start/stop replay
 - Fill in details – perforce revision, game location

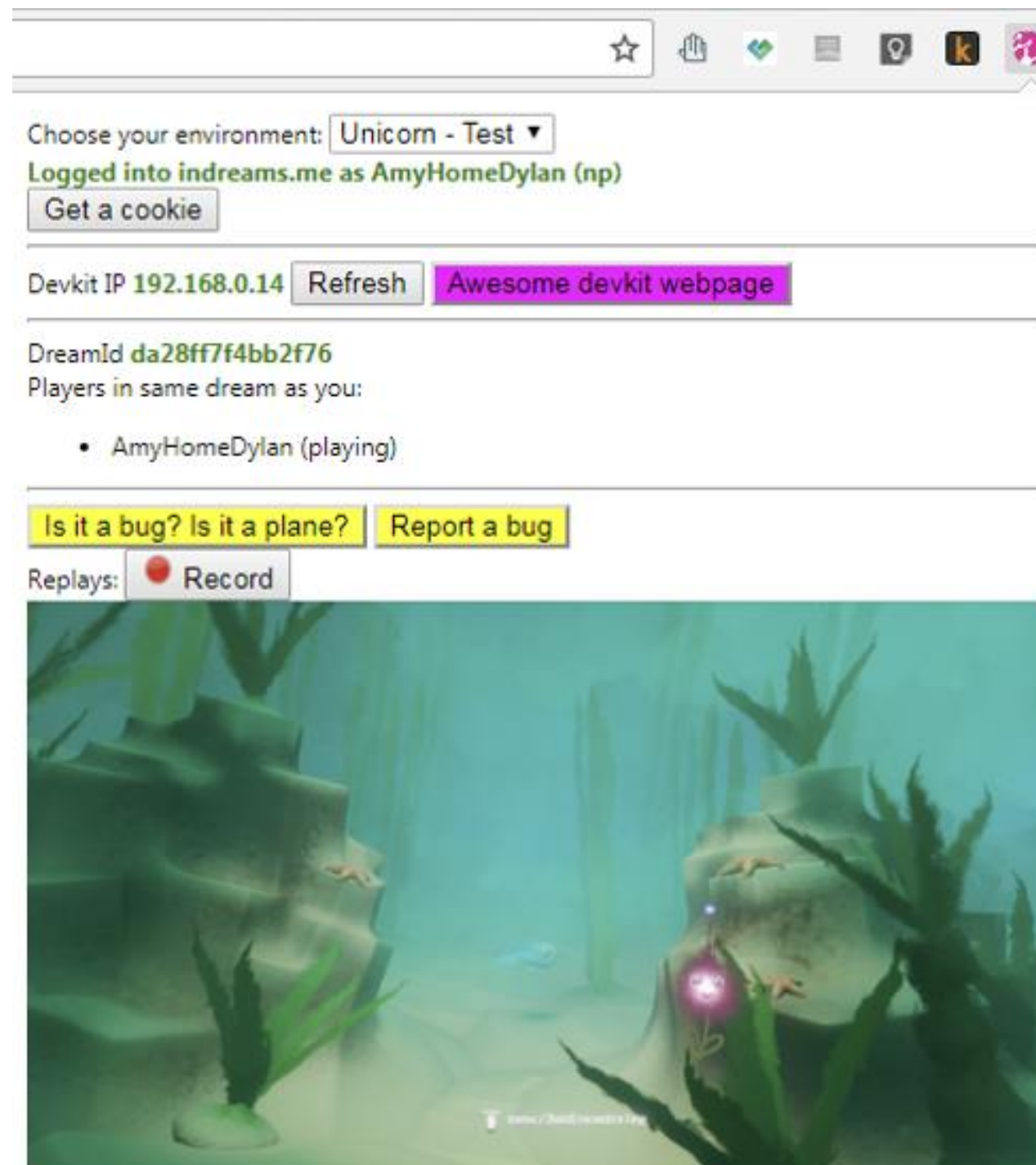
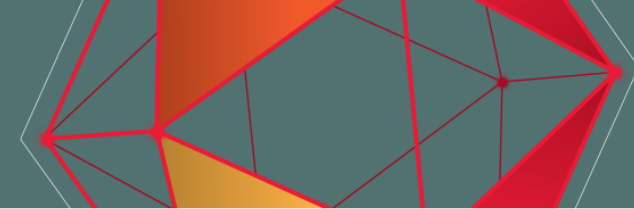




Detecting bugs – designers/artists/audio bods

- Dashboard
 - Dependencies – Target Manager, Remote Viewer Server, ...
 - List builds
 - latest green build – update low risk
 - Launch build on devkit (will update SDK if necessary)
- Fatal asserts
- Chrome extension
 - Report bug button – autofills, add screenshot, replay buttons
 - Uses Jira CreateIssueDetails url – params in url







Create Issue - Media Mo x

Not secure | https://jira.mmqd.ship.scea.com/jira/secure/CreateIssueDetails!init.jspa?pid=10001&issuetype=10100&labels=Dev_Is...

Apps 804 Gmail Facebook JellyTots Childcare Vouchers SET Card Game All Sony products : Sc Issues - Cooking Sche canitalkabout.me Free Hosting EU

JIRA Dashboards Projects Issues Agile TestRail Create Search

Create Issue

summary: You must specify a summary of the issue.

Project **Dreams**

Issue Type **QA Func**

ENG JPN

Summary*

You must specify a summary of the issue.

Build

Begin typing to find and create labels or press down to select a suggested label.

The build that the issue was found on

Class **None**

The severity category of the issue

Game Locations

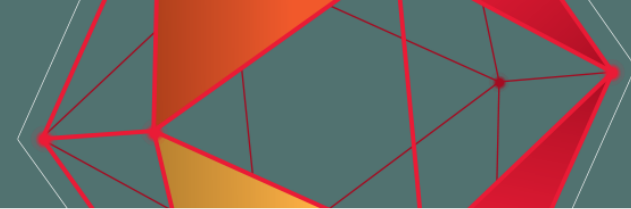
```
{"dreamId":"da682ff9865d238","versionId":"v0999e4eb68b21c","title":"Default hub simple","uploaded_by_onlineid":"mm_todd","build_id":{"timestamp":1510266354,"id":0,"serialise_revision":925,"perforce_revision":79846,"asserts":1,"debug":0,"platform":0}}
```

Dreams Game Location, used to paste in data from a google chrome extension

Func Type **None**

For Functional issues, this is the bug type





Routing Bugs

- Autofilled information
 - Programmer who added assert
 - OOS gotcha
 - Perforce auth joy – security level 3
 - Assert message
 - Callstack
 - Perforce revision
 - Replay
 - Screenshot
 - Sometimes video, but replays mostly replace video, todo!





Fixing bugs – repro it

- Chrome Extension – there are two more Jira buttons

- Launch replay button – single click repro
 - Gotcha content
- Launch game location
 - Better for content
 - Todo – iterate content repro tools





See <https://youtu.be/12XNX7bp01U> for a video of the Jira Launch Replay Button of Joy

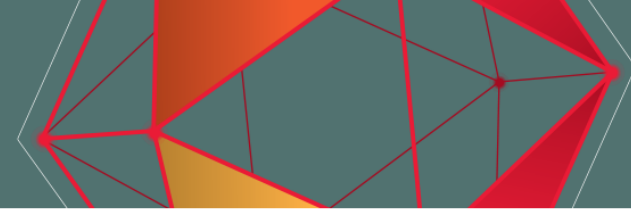




Testing your Fix

- Replay – just run the replay again and check it completes
- Autobot
 - Script to build local code, upload it to autobot and run smoke test
 - Emails you results
 - Takes about 20 mins
 - Used for scary changes
- Script to mark bugs fixed based on checkin comment [!345]

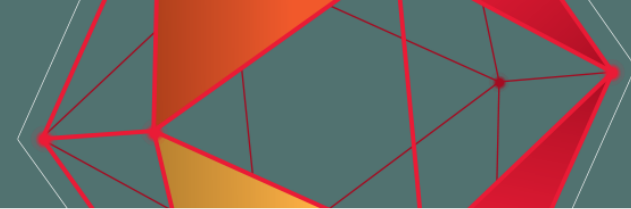




Verifying a Fix

- Chrome extension Jira buttons
 - Launch replay
- Think about adding a replay to autobot if this was not an autobot bug
 - Autobot replay consideration – keep it short, finish screenshot
- Todo
 - Track asserts/crashes that have been fixed when autobot completes a run
 - Put them in results email
 - Tag the bug in Jira so progs don't waste time trying to repro





Summary

- Tools

- Autobot automated test system (Django, mysql, python)
- Jira bug tracking (rest api – love it!)
- Recap (rest api is improving rapidly)
- Deterministic replays
- Our game server can forward on a message to the devkit of a particular user and route the reply back
- All linked together with a Chrome extension which talks to all of them

- To minimise the impact of bugs you want to find them early, collate all the necessary information to repro, route them to the right person, make it simple to repro and fix them and verify the fix.

- Lots of little tweaks all add up





Questions???

I'm on email at amy@mediamolecule.com,

or twitter [@amy_phillips](https://twitter.com/amy_phillips)

