



Running Live: Many Games for a Long Time

ChangKyu Song

Head of Live Infra Technology, Nexon

Who am I?

ChangKyu Song

Head of Live Infra Technology Department, Nexon

- 1999, developed HanStar, a Korean Localization Utility for Starcraft
- 1999, Gabriel Knight 3 Korean Localization
- 2001, Worms World Party Korean Localization
- 2002, Crazy Arcade BnB (Arcade) Programmer
- 2002-2003, Dizzy Pang (Puzzle Arcade) Lead Programmer
- 2004-2006, Big Shot (2D Shooter) Lead Programmer
- 2006-2010, Bubble Fighter (Third-person Shooter) Lead Programmer
- 2010-2011, Mabinogi 2 (3D MMORPG) Programmer
- 2011-2014, Dungeon & Fighter (2D Action MORPG) Technical Director
- 2014-, Head of Live Infra Technology Department, Nexon



About Nexon

- A Korean Publisher
Running many games for a long time



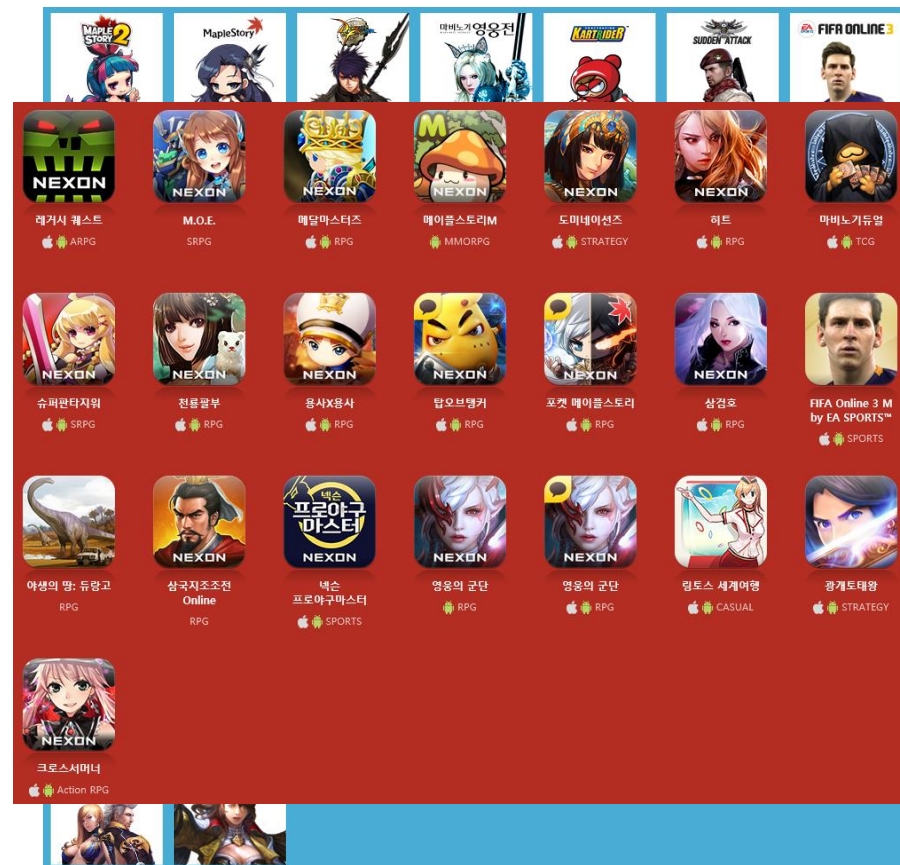
About Nexon

- A Korean Publisher
Running many games for a long time
- >30 PC titles
 - Dungeon & Fighter, Maple Story, Bubble Fighter, ..



About Nexon

- A Korean Publisher
Running many games for a long time
- >30 PC titles
 - Dungeon & Fighter, Maple Story, Bubble Fighter, ..
- >20 mobile games
 - DomiNations, HIT, Legacy Quest ..



About Nexon

- 2015's full year revenue:

\$1.67 billion

(190.3 billion yen)

Last week **Nexon** announced its financial results for the fourth quarter and fiscal year ended December 31, 2015 with financial growth across the board. The full year revenue for Nexon came to 190.3 billion yen (USD 1.67 billion), a 10 per cent jump on 2014. Meanwhile, operating income grew by over a third (37 per cent) to reach 62.3 billion yen (USD 0.51 billion). Nexon's Clash of Clans-esque strategy title **DomiNations** was among the new releases helping to drive revenue, as the game's popularity surged following its release in April.

Q4 alone delivered a seven per cent rise versus the previous year, with 'higher than expected' Dungeon Fighter and HIT sales boosting revenue to 45.8 billion yen (USD 0.37 billion).

"Nexon's solid performance throughout 2015 was driven by our relentless focus on the quality of our live and new games and the expansion of world class partnerships. Our full-year revenues, operating income and net income all grew year-over-year, reflecting the positive response from players of our high-quality titles," said president and CEO Owen



Nexon CEO Mahoney

Table of Contents

- Fighting against Legacy - Running game for a Long Time
 - Focusing on hotspot by change analysis
 - Establishing Integration Process during live stage
- Mitigating Common Technical Risk of Many Games
- Conclusion



Fighting against Legacy – Running game for a Long Time

Finding hotspot in source code by change analysis

Working on Dungeon & Fighter

- Joined and worked as Global Development Lead and Technical Director in 2011
- One of most successful game in Nexon
(*Neople is a subsidiary company of Nexon)
- Launched On Aug 2005 (10 years old)
- More than 3 million concurrent users in China
- More than \$1.0 billion revenue in 2015
- Still one of most top-grossing MMOs in 2016



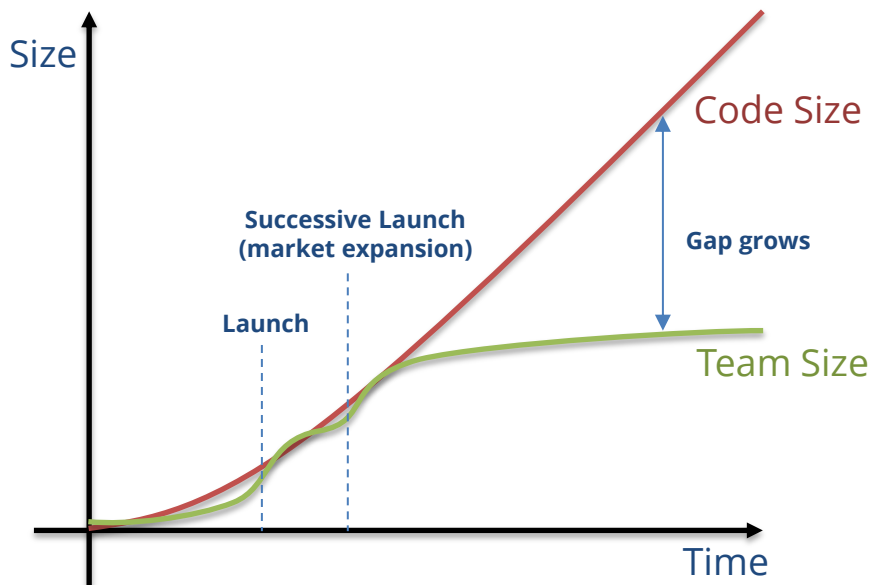
CNBC HOME U.S. ▼ NEWS MARKETS INVESTING TECH SMALL BIZ

Digital gaming sales hit record \$61 billion in 2015

Digital PC

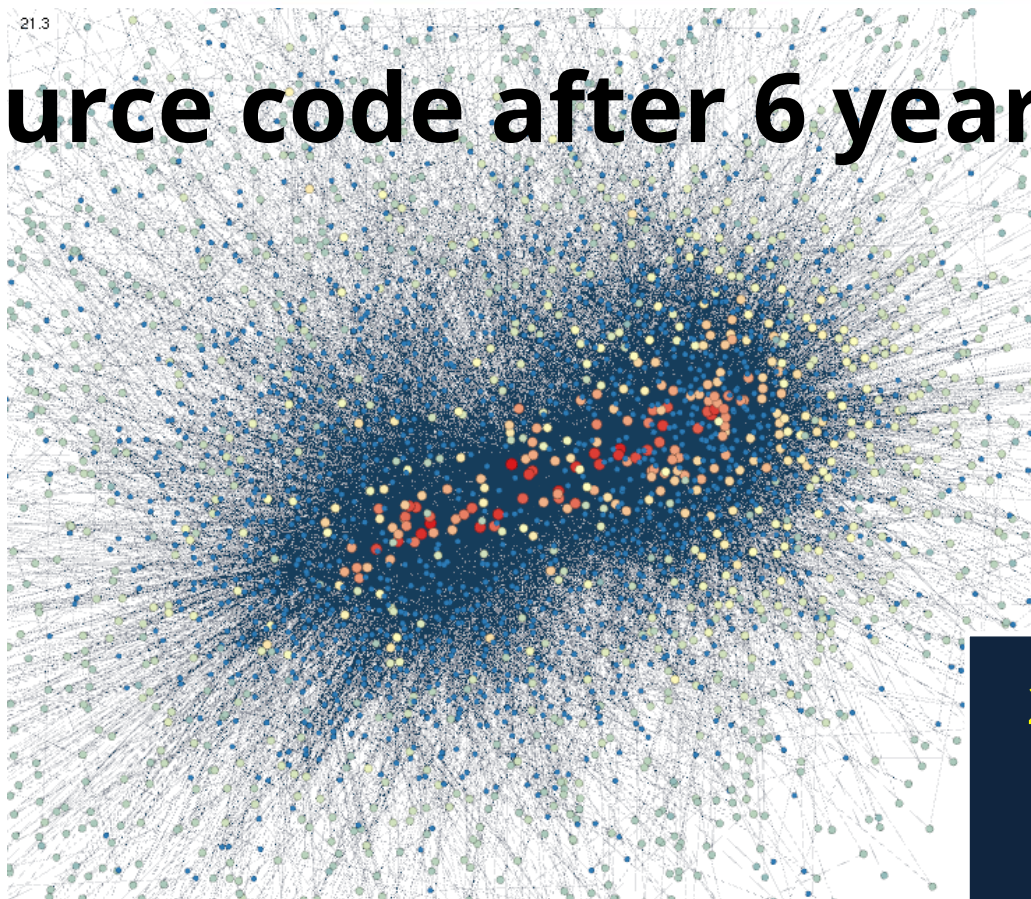
1. "League of Legends" (Tencent/Riot Games, \$1,628)
2. "Crossfire" (SmileGate, \$1,110)
3. "Dungeon Fighter Online" (Neople, \$1,052)
4. "World of Warcraft" (Activision, \$814)

When running live service for a long time, Code size grows, team doesn't scale



21.3

Source code after 6 years



← A single source file

2282 .cpp files
2935 .h files
5855 classes

Game grew too big

- As in 2014 (After 9 years of service):
 - >2,000 skills
 - >4,500 quests
 - >10,000 C++ source files
 - >50,000 equipment items
 - >200,000 animations
 - >5,000,000 lines of code
 - >10,000,000 images
- The biggest and fastest-paced project ever experienced

Needed to pay off Technical Debt

- Dev team was suffering from technical debt
 - Adding very small content was very costly
- Needed to pay off technical debt



Finding the realistic, efficient way

- >10,000 source code files
- Most code smelled bad
- Rather than understanding gigantic size of code and improving hand by hand
- Decided to take advantage of automation

Doing some Automation

- Introduced Build Server, Data Validation Test and Static Analysis
- Build Server, Data Validation helped a lot
 - Improving notification made it much better
- Static Analysis didn't help much
 - Fixed thousands of warnings over weeks with many people
 - Valuable but high cost, little gain
- Need to focus on important thing/area
 - Added basic things like crash reporter,
 - "What is important?"
 - Where is the important area? – each developer said different area



Finding what's important for Software quality

- Separating frequently-changing area and less-frequently changing area is important

Example case	Frequently-changing area	Less frequently changing area
Software Engineering	Interface	Implementation
STL	Algorithms	Data Structure
Template/Generic	Logic	Data Type
Game Engine / Framework / Library	Library Code	Logic Code
Data-driven Development	Data	Code

Finding hotspot: Principle of locality

- If particular source location is changed, it is likely to change again in the near future

Types of locality [\[edit\]](#)

There are several different types of locality of reference:

Temporal locality

If at one point in time a particular memory location is referenced, then it is likely that the same location will be referenced again in the near future. There is a temporal proximity between the adjacent references to the same memory location. In this case it is common to make efforts to store a copy of the referenced data in special memory storage, which can be accessed faster. Temporal locality is a special case of the spatial locality, namely when the prospective location is identical to the present location.

Spatial locality

If a particular memory location is referenced at a particular time, then it is likely that nearby memory locations will be referenced in the near future. In this case it is common to attempt to guess the size and shape of the area around the current reference for which it is worthwhile to prepare faster access.

Analyzing change of code over time

- Not just doing static analysis which treats code as static data,
- But also tried to treat code as changing data and analyze **code changes** in VCS over time

Static analysis approach:



Static data

A gray rounded rectangle containing the text 'Static data'.

Expanded approach:



Past data

A gray rounded rectangle containing the text 'Past data'.



changes

A yellow arrow pointing right with the word 'changes' in red text.



Current data

A gray rounded rectangle containing the text 'Current data'.



changes

A yellow arrow pointing right with the word 'changes' in red text.



Future data

A gray rounded rectangle containing the text 'Future data'.

First step: Simple Analysis

- How many files have changed and how often has each changed?

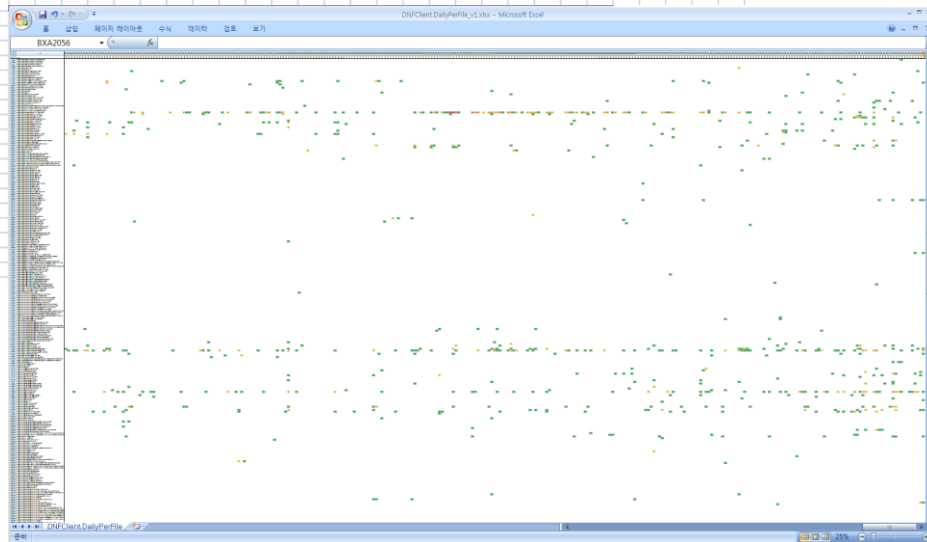
Wrote 300 lines of python script to analyze change of source code

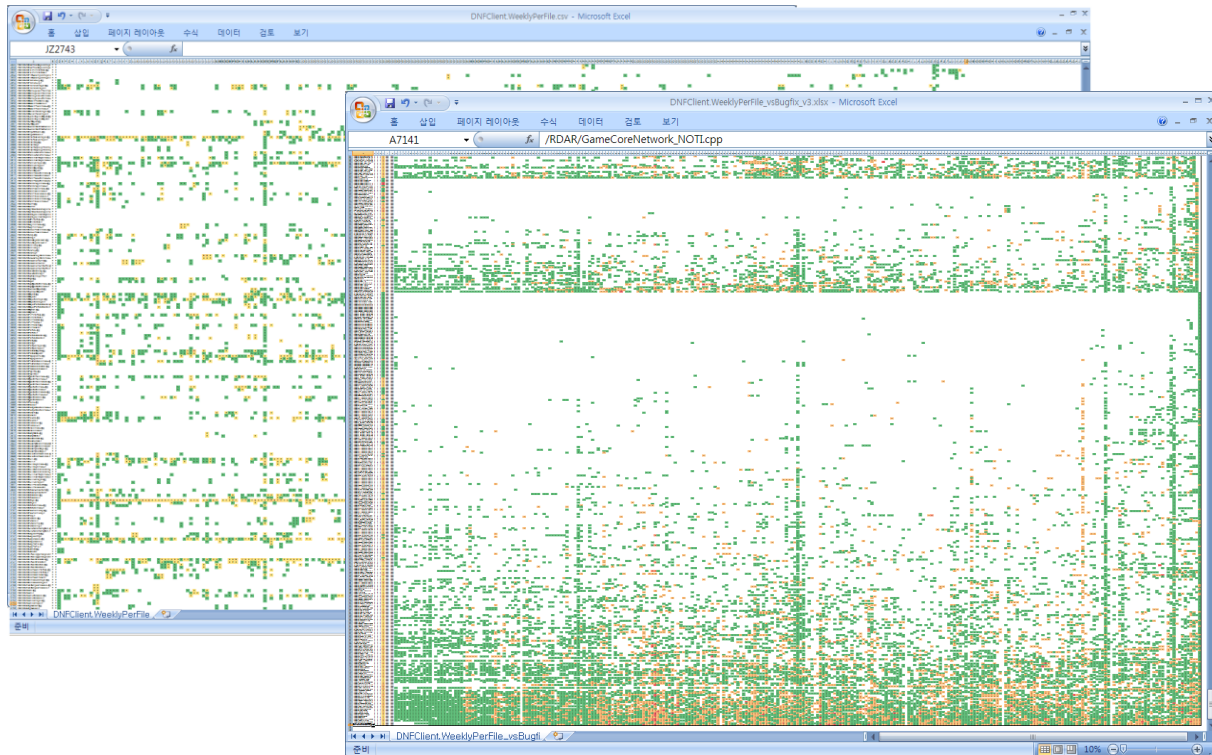
```
262 for i in xrange(len(msgs)):
263     msg = msgs[i]
264     print "processing %d / %d" % (i+1, len(msgs))
265     logmsg = msg.message.decode("utf-8")
266
267     logFlags = set()
268     if hasKeywords(logmsg, ModifyKeywords):
269         logFlags.add("modify")
270     if hasKeywords(logmsg, BugfixKeywords, BugfixExcludeKeywords):
271         logFlags.add("bugfix")
272     if hasKeywords(logmsg, RevertKeywords):
273         logFlags.add("revert")
274
275     # 500개 이상의 변경은 무시한다
276     if len(msg.changed_paths) > 500:
277         continue
278
279     for chg in msg.changed_paths:
280         if chg.action != "M":
281             continue
282
283         st = time.localtime(msg["date"])
284         unitTime = time.mktime((
285             st.tm_year,
286             st.tm_mon,
287             st.tm_mday - st.tm_wday,
288             0, 0, 0, 0, 0))
289
290         fullpath = chg.path.decode("utf-8")
291         path = fullpath
292         if path.startswith(repopath):
293             path = path[len(repopath):]
294
295         if not path.endswith(".cpp"):
296             continue
297
298         url = repobase + fullpath
299         url = url.replace(" ", "%20")
300
301         try:
302             procOne(url, msg.revision.number, unitTime, logFlags)
303         except Exception as e:
304             fe = open("exception.log", "at")
305             fe.write("%d:%s\n" % (msg.revision.number, fullpath.encode("cp949")))
306             fe.write("%s\n" % e.message)
307             fe.close()
308
309     print "writing.."
310     f = open("FuncHistory.pkl", "wb")
311     cPickle.dump((tbl, colMin, colMax), f, 2)
312     f.close()
```

[illegible]

Nothing interesting

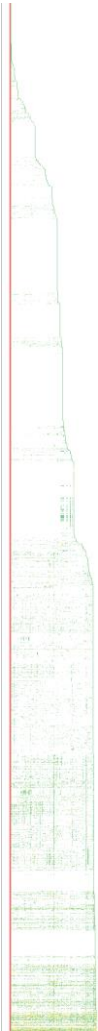
First result





More readable result
Zoomed out (group by week)
& Sorted by recent change, frequency of change

Bird's-eye view

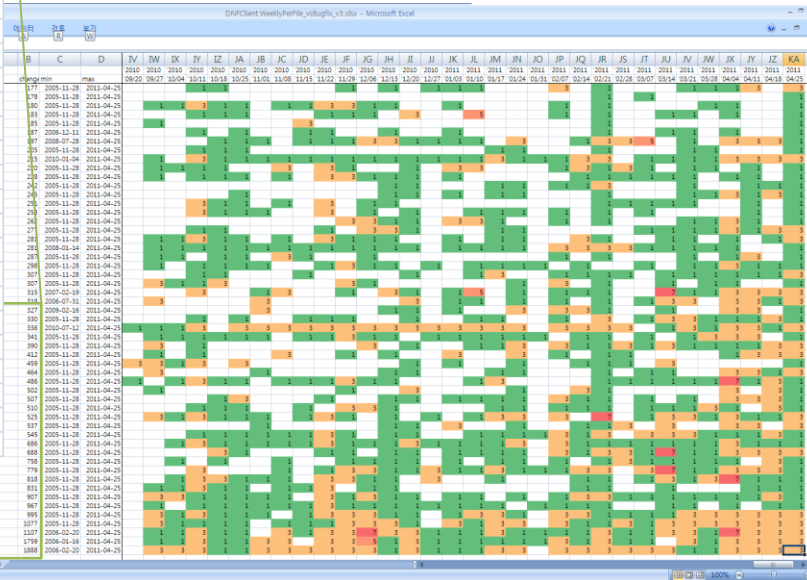


Source files that are changing often
= Hotspots

Highly efficient when improved

/113	/RDAR/Character/CNPriest.cpp	318
/114	/RDAR/Character/CNThief.cpp	327
/115	/RDAR/IRDAActiveObject.h	330
/116	/RDAR/Character/CNATFighter.cpp	336
/117	/RDAR/IRDPopupWindow.cpp	341
/118	/RDAR/IRDCharacter.h	390
/119	/RDAR/CNRDVirtualCharacter.cpp	412
/120	/RDAR/Character/CNSwordman.cpp	459
/121	/RDAR/Character/CNGunner.cpp	464
/122	/RDAR/GameCoreInit.cpp	486
/123	/RDAR/Character/CNFighter.cpp	502
/124	/RDAR/PopupWindow/CNRDItemInfoWindow.cpp	507
/125	/RDAR/IRDCollisionObject.cpp	510
/126	/RDAR/CNRDUdpModule.cpp	525
/127	/RDAR/Character/CNMage.cpp	537
/128	/RDAR/IRDAActiveObject.cpp	545
/129	/RDAR/CNRDInterfaceManager.cpp	686
/130	/RDAR/CNRDEquipment.cpp	688
/131	/RDAR/CNRDStage.cpp	758
/132	/RDAR/IRDMonster.cpp	779
/133	/RDAR/GameCoreSub.cpp	818
/134	/RDAR/CNRDChatting.cpp	831
/135	/RDAR/RDCommon.cpp	907
/136	/RDAR/RDAR.vcproj	967
/137	/RDAR/CNRDBaseInterface.cpp	995
/138	/RDAR/IRDCharacter.cpp	1077
/139	/RDAR/GameCoreNetwork_CMD.cpp	1107
/140	/NeopleMain/dnfstr	1759
/141	/RDAR/GameCoreNetwork_NOTI.cpp	1888

/141	/RDAR/GameCoreNetwork_NOTI.cpp	1888
/142	/RDAR/CNRDStage.cpp	758
/143	/RDAR/CNRDInterfaceManager.cpp	686
/144	/RDAR/Character/CNMage.cpp	537
/145	/RDAR/IRDAActiveObject.cpp	545
/146	/RDAR/CNRDUdpModule.cpp	525
/147	/RDAR/IRDCollisionObject.cpp	510
/148	/RDAR/PopupWindow/CNRDItemInfoWindow.cpp	507
/149	/RDAR/Character/CNFighter.cpp	502
/150	/RDAR/GameCoreInit.cpp	486
/151	/RDAR/Character/CNGunner.cpp	464
/152	/RDAR/Character/CNSwordman.cpp	459
/153	/RDAR/CNRDVirtualCharacter.cpp	412
/154	/RDAR/IRDCharacter.h	390
/155	/RDAR/IRDAActiveObject.h	330
/156	/RDAR/Character/CNThief.cpp	327
/157	/RDAR/Character/CNPriest.cpp	318



File	change min	max
/WTL80/include/atlsresce.h	1 2009-03-16	2009-03-16
/WTL80/include/atlsrcl.h	1 2009-03-16	2009-03-16
/WTL80/include/atlsplit.h	1 2009-03-16	2009-03-16
/WTL80/include/atltheme.h	1 2009-03-16	2009-03-16
/WTL80/include/atluser.h	1 2009-03-16	2009-03-16
/WTL80/include/atlwince.h	1 2009-03-16	2009-03-16
/WTL80/include/atlwinx.h	1 2009-03-16	2009-03-16
/WTL80/readme.htm	1 2009-03-16	2009-03-16
/Tools/DNFSpline	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline.sln	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/DNFSpline.cpp	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/DNFSpline.h	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/DNFSpline.vcproj	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/DNFSpline_2005.rc	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/ReadMe.txt	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/res	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/res/DNFSpline.ico	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/res/DNFSpline.manifest	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/res/DNFSpline.rc2	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/res/DNFSplineDoc.ico	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/res/Toolbar.bmp	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/stdafx.cpp	1 2009-03-23	2009-03-23
/Tools/DNFSpline/DNFSpline/stdafx.h	1 2009-03-23	2009-03-23
/DNFInGameTools/lib	1 2009-05-11	2009-05-11
/DNFInGameTools/lib/ticpp.lib	1 2009-05-11	2009-05-11
/DNFInGameTools/lib/ticppd.lib	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/TiCPP.vcproj	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/TinyXML++ncb	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/TinyXML++sln	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/TinyXML++suo	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/build_instructions.txt	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/changes.txt	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/dox	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/premake.exe	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/premake.lua	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/premake4.lua	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/readme.txt	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/ticpp.cpp	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/ticpp.lua	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/ticpp4.lua	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/ticpprch	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/tinystr.cpp	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/tinystr.h	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/tinyxml.cpp	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/tinyxml.h	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/tinyxmlerror.cpp	1 2009-05-11	2009-05-11
/DNFInGameTools/ticpp/tinyxmlparser.cpp	1 2009-05-11	2009-05-11
/Tools/EquipmentImagePack	1 2009-06-22	2009-06-22
/Tools/EquipmentImagePack/EquipmentImagePack.cpp	1 2009-06-22	2009-06-22
/Tools/EquipmentImagePack/EquipmentImagePack.ncb	1 2009-06-22	2009-06-22
/Tools/EquipmentImagePack/EquipmentImagePack.sln	1 2009-06-22	2009-06-22
/Tools/EquipmentImagePack/EquipmentImagePack.suo	1 2009-06-22	2009-06-22

March 14-18, 2016 · Expo: March 16-18, 2016 #GDC16

For source files that change rarely we can consider separating them from the main project and reduce build time

WTL (Windows Template Library)
DNFSpline:
Tinyxml
EquipmentImagePack

Each name looks a lot like library code



```
11>Compiling...
11>profiler.cpp
11>Compiling...
11>TEA.cpp
11>Compiling...
11>sha256.cpp
11>Compiling...
11>SHA.cpp
11>Compiling...
11>Rijndael.cpp
11>Compiling...
11>Method.cpp
11>Compiling...
11>CNRDUIEnchantTool.cpp
11>Compiling...
11>SlangFilterSub.cpp
11>SlangFilter.cpp
11>Compiling...
11>RDChampionConst.cpp
11>Compiling resources...
11>Microsoft (R) Windows (R) Resource Compi
11>Copyright (C) Microsoft Corporation. All
11>Linking...
11>Build Time 32:54
11>RDAR - 0 error(s), 73 warning(s)
===== Rebuild All: 11 succeeded, 0 fai
```

Before

```
11>wave.c
11>CWebPage.c
11>Compiling...
11>profiler.cpp
11>Compiling...
11>TEA.cpp
11>Compiling...
11>sha256.cpp
11>Compiling...
11>SlangFilterSub.cpp
11>SlangFilter.cpp
11>Compiling...
11>RDChampionConst.cpp
11>Compiling resources...
11>Microsoft (R) Windows (R) Resource Compi
11>Copyright (C) Microsoft Corporation. All
11>Linking...
11>Build Time 15:37
11>RDAR - 0 error(s), 73 warning(s)
===== Rebuild All: 11 succeeded, 0 fai
```

After

53% decreased
(x2.1 improvment)

Applying unity build technique for the half of less-frequently-changing code made compiling twice as fast.

Not good if files with big impact
change frequently

Considering about Change Impact

- Change Impact =
Change Frequency * Impact (=Reverse-dependencies including itself)

	A	B	C	D	E	F	G	H	I	J	K	L
1	File	Impact Size	Impact Count	02/28	03/07	03/14	03/21	03/28	04/04	04/11	04/18	04/25
38	cnscriptloader.h	443	2	0	0	0	220	0	0	0	0	223
39	cnwarroom.h	508	2	0	253	0	255	0	0	0	0	0
40	cngamesocket.h	582	2	0	0	0	285	0	0	0	0	297
41	cnrditeminfowindow.h	585	7	0	162	0	82	0	0	0	255	86
42	cnrdudpmodule.h	596	2	0	0	596	0	0	0	0	0	0
43	udpcore.h	618	2	0	0	307	311	0	0	0	0	0
44	icreature.h	800	4	0	0	0	0	0	0	0	598	202
45	cnrdquestmodule.h	875	4	0	0	0	0	0	436	218	0	221
46	cnrdvirtualcharacter.h	925	6	0	0	0	0	0	154	462	309	0
47	cnrdmovementunit.h	1145	3	380	0	0	0	0	0	765	0	0
48	cnrdiautomationcontrolloader.h	1207	8	125	0	258	266	136	139	0	0	283
49	cnwarroommodule.h	1218	5	0	0	0	244	244	243	243	244	0
50	cnrdiautomationssystem.h	1248	7	0	0	0	192	266	270	0	0	580
51	udpcommondefine.h	1265	4	0	0	309	632	0	0	324	0	0
52	cnrdchatting.h	1310	5	0	0	0	0	257	521	0	0	532
53	cnusermanager.h	1691	3	0	0	0	557	1134	0	0	0	0
54	cnrditemmanager.h	2003	3	656	0	0	663	0	0	0	0	684
55	cnrdstage.h	2182	2	0	0	0	1089	1093	0	0	0	0
56	cnrdobjectmanager.h	2243	3	0	0	0	742	0	0	0	749	752
57	cnrdinterfacemanager.h	3135	6	0	0	516	1040	1051	528	0	0	0
58	cnrdbaseinterface.h	3223	18	171	344	0	176	712	905	181	734	0
59	gamecore.h	3953	11	344	693	694	1059	0	0	722	0	441
60	cnaimonster.h	3979	4	0	0	979	0	0	2000	1000	0	0
61	irdskillstate.h	4017	8	0	0	0	1979	2023	0	15	0	0
62	cnrdableobject.h	4949	5	0	0	1965	0	1987	997	0	0	0
63	cnrdskillatemanager.h	4982	8	0	0	0	3953	1016	13	0	0	0
64	cnrdskill.h	5030	5	0	0	989	1003	1007	1010	0	1021	0
65	irdaicharacter.h	5208	12	0	0	856	0	870	435	870	0	2177
66	digginglog.h	5305	2	0	0	0	0	0	5305	0	0	0
67	imouse.h	5322	2	0	0	0	0	0	5322	0	0	0
68	image.h	5463	2	0	0	2714	0	2749	0	0	0	0
69	cnrdpassiveobject.h	6188	3	0	0	0	0	2048	0	0	0	4140
70	cnrdequipment.h	7125	23	0	0	0	890	600	1800	1244	2264	327
71	ipvpmodule.h	7597	7	0	1068	0	1077	1086	3267	1099	0	0
72	icontrolveth.h	7919	4	0	0	0	1966	1966	1983	0	0	2004
73	irditem.h	8459	9	0	0	0	2289	769	2154	1077	1080	1090
74	cnuser.h	10660	9	0	1166	3516	1183	1192	0	0	3603	0
75	graphicsystem.h	10708	4	0	0	0	0	0	0	2676	8032	0
76	cnrdactionscrip.h	11156	5	0	2207	2213	2228	0	2253	2255	0	0
77	securityclientm.h	11287	5	0	2235	2235	2250	0	2272	0	0	2295
78	irdcollisionobject.h	13170	6	0	0	0	4346	4381	0	0	0	4443
79	icontrol.h	15626	8	3842	0	1947	3894	1964	1979	0	0	2000
80	cnrdobject.h	15647	7	0	0	2209	2216	4467	0	2247	2251	2257
81	irdpopupwindow.h	17690	19	0	902	0	3677	1855	3726	3750	941	2839
82	cnrdanimation.h	20185	9	0	4446	0	4472	9003	2264	0	0	0
83	rdcommon.h	28924	11	0	2599	5219	10498	0	5296	2652	0	2660
84	irdcharacter.h	29870	30	972	974	3906	2965	1987	7956	1992	4040	5078
85	irdactiveobject.h	31002	18	0	1691	5089	1705	6887	1731	3466	5205	5228
86	irdmonster.h	33218	21	0	1552	7794	3140	1579	6369	6374	3192	3218

Analyzing change impact along the time

- Average impact of irdmonster.h: 3178
 - Change of irdmonster.h caused **3178** cpp files to be required to recompile for a week in average
 - Huge side effect along the time
- Began to work on major bottlenecks
 - Popup window
 - CNRDAnimation

Focusing on hotspot was quite efficient

- Large gain with low cost
 - When focusing on hotspot by changes and change impacts over time
- Build time cut in half with very small effort
- Effective even without understanding the whole system or source code



Fighting against Legacy – Running game for a Long Time

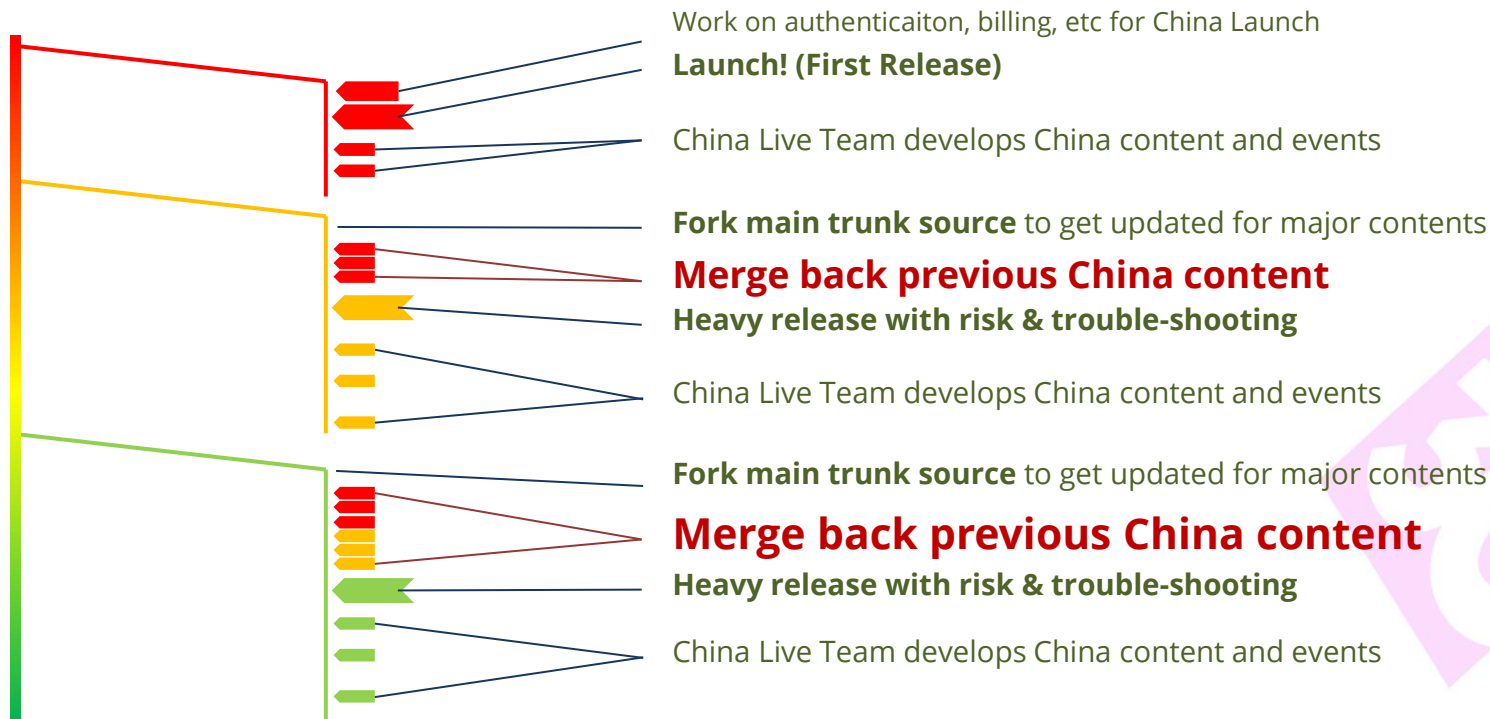
Establishing Integration Process while in live

Another important problem:

Branches were not being integrated

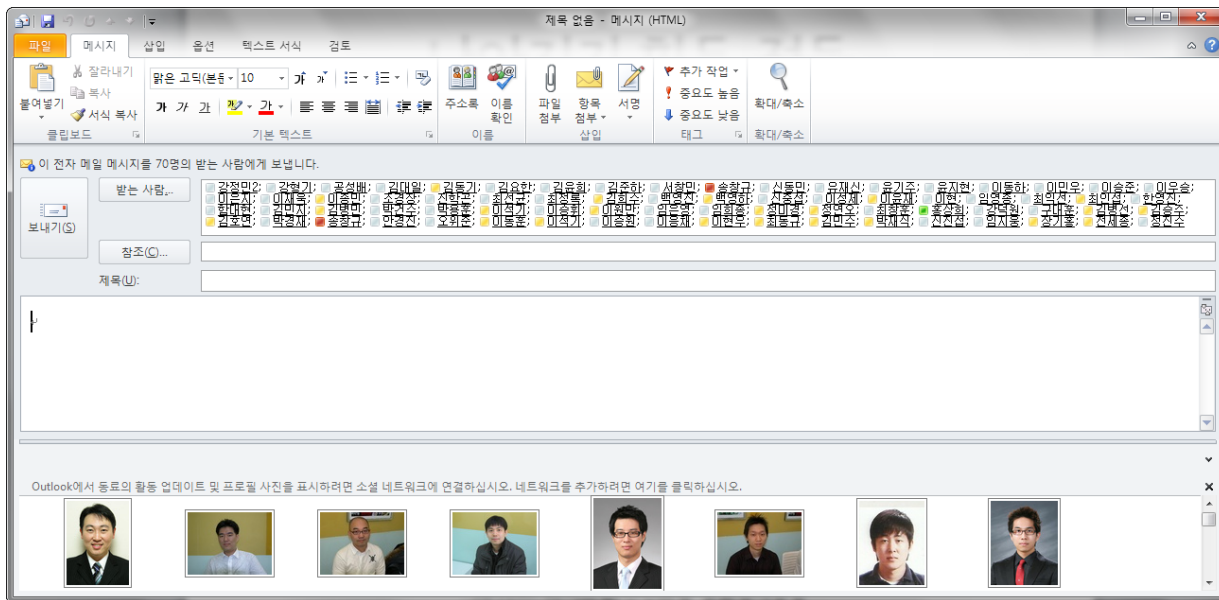
- Was a small company when launching
- Dev team and the industry didn't have much experience of launching service overseas 10 years ago
- Made separate branches for service regions and separate teams worked on them

Overseas live dev team suffered from merge hell



Gone too far, too late

- Cost and time to merge along the main branch for a major update constantly increased to 4 months to work
 - “I feel like launching a new game every 6 months”
- Missed golden time to integrate
- Decided to establish an integrate process
 - unsustainable process, highly likely not to make the major update within 2~3 years



7 Development Teams on a Project



http://dnfsvn/svn/dnf - Log Messages - TortoiseSVN

Filter by Messages, Paths, Authors, Revisions, Bug-IDs, Date, Date Range

From: 2013-04-16 To: 2013-04-23

Revision	Actions	Author	Date	Message
102705		crickerdiper21	2013년 4월 16일 화요일 오후 6:35:08	[생성시 ai 생성 안되던 부분 수정 - 몬스터 저장 부분 추가 수정 - [targeting bonus], [warlike] e
102704		sleepitank	2013년 4월 16일 화요일 오후 6:34:47	오려주게 걸기 한정 로직 수정
102703		gong	2013년 4월 16일 화요일 오후 6:34:23	ent-NOT1분기[GameCoreNetwork_NOT1.cpp -> GameCoreNetwork_NOT1_Town.cpp 분리
102702		crickerdiper21	2013년 4월 16일 화요일 오후 6:34:15	tox 그리주게 음선 버튼 추가 - F1, Ctrl 누를시 moBox 그리주게 추가
102701		kdd372418	2013년 4월 16일 화요일 오후 6:30:22	102054, 102057, 102418, 102444, 102600 from trunk: [jwp도너먼트개최권] 진출사 und
102700		crickerdiper21	2013년 4월 16일 화요일 오후 6:29:59	키그 추가
102699		dong1036	2013년 4월 16일 화요일 오후 6:28:58	14) IGA 링크 on/off 관련 형틀 추가.
102698		aquasoul16	2013년 4월 16일 화요일 오후 6:28:06	음 프럼이 미리 그리기 부분 넘체크 슈퍼마대 슬로우 objectManager 관련 null 체크 Merged rev
102697		genicat	2013년 4월 16일 화요일 오후 6:27:24	강비/마법석 밑봉시 내구도 확인 관련 메시지 #문 수정 _C_BUG_ENDURANCE_SUPPORT_MAGIC
102696		genicat	2013년 4월 16일 화요일 오후 6:26:05	강비/마법석 칼레리도 박스 가능하도록 수정 _C_BUG_ENDURANCE_SUPPORT_MAGIC_STONE_J
102695		coke3711	2013년 4월 16일 화요일 오후 6:21:21	- 공격정보와 레벨 정보 링크
102694		gong	2013년 4월 16일 화요일 오후 6:21:19	ent-NOT1분기[GameCoreNetwork_NOT1.cpp -> GameCoreNetwork_NOT1_Dungeon.cpp 분리
102693		durum04	2013년 4월 16일 화요일 오후 6:18:41	전 추가 UNDO 가능 추가
102692		sssin	2013년 4월 16일 화요일 오후 6:17:56	드 수정
102691		senicova	2013년 4월 16일 화요일 오후 6:17:43	그 출력 개선; 로그에서 한글 출력 삭제
102690		patori	2013년 4월 16일 화요일 오후 6:17:39	키 릴리즈 발드 오류 수정
102689		hikary	2013년 4월 16일 화요일 오후 6:17:05	102686 from trunk: [배그][DNF-12394]남격가 년가드 플타입 감소하지 않는 배그 실설 이동 ...
102688		AAAAAGood	2013년 4월 16일 화요일 오후 6:16:45	버릴시 오버백스 일단 클라그 내림
102687		gong	2013년 4월 16일 화요일 오후 6:15:42	ent-CHD 분리[CHDPACKET_START_GAME 누락 처리, GamecoreNetwork_CHD_h.cpp 연결코드만
102686		hikary	2013년 4월 16일 화요일 오후 6:13:26	남격가 년가드 플타입 감소하지 않는 배그 실설 이동
102685		pulse	2013년 4월 16일 화요일 오후 6:12:45	CMD_ONEDAY_ITEM_SHOP_IPN_ 일일상점 리스트 갱신 지점 추가.
102684		aquasoul16	2013년 4월 16일 화요일 오후 6:10:51	파이마대 슬로우 obje-Manager 관련 null 체크
102683		aquasoul16	2013년 4월 16일 화요일 오후 6:09:55	음 프럼이 미리 그리기 부분 넘체크
102682		victory99-001	2013년 4월 16일 화요일 오후 6:07:55	
102681		patori	2013년 4월 16일 화요일 오후 6:07:41	키 릴리즈 발드 오류 수정
102680		pulse	2013년 4월 16일 화요일 오후 6:07:28	CMD_ONEDAY_ITEM_SHOP_IPN_ 일일상점 리스트 갱신 지점 추가.
102679		oogrs	2013년 4월 16일 화요일 오후 6:06:41	가 아닐 때에만 업조 전송
102678		gong	2013년 4월 16일 화요일 오후 6:05:31	ent-CHD 분리[GameCoreNetwork_CHD.cpp -> GameCoreNetwork_CHD_Town.cpp 초기버전 커밋
102677		ey0912	2013년 4월 16일 화요일 오후 6:05:14	102676 from trunk: [마법부여개편] 다른유저가 상점을 사용할 때 패킷을 보낼 때 주인의 uid를
102676		ey0912	2013년 4월 16일 화요일 오후 6:03:47	문유저가 상점을 사용할 때 패킷을 보낼 때 주인의 uid를 보내도록 수정
102675		oogrs	2013년 4월 16일 화요일 오후 6:03:38	가 아닐 때에만 업조 전송
102674		goobobsa	2013년 4월 16일 화요일 오후 6:02:28	헬파티 난이도가 바뀌는 문제 수정
102673		chikino	2013년 4월 16일 화요일 오후 6:02:24	브렉트 파괴 출력 버그 수정
102672		goobobsa	2013년 4월 16일 화요일 오후 6:01:18	헬파티 난이도가 바뀌는 문제 수정

[리팩토링머지][C]ent-NOT1분기[GameCoreNetwork_NOT1.cpp -> GameCoreNetwork_NOT1_Town.cpp 분리

Path

Path	Action	Copy from path
/branches/cn/tasks/TimeGate_130109_for_merge_base	Modified	
/branches/cn/tasks/TimeGate_130109_for_merge_base/DNFClient	Modified	
/branches/cn/tasks/TimeGate_130109_for_merge_base/DNFClient/RDAR	Modified	
/branches/cn/tasks/TimeGate_130109_for_merge_base/DNFClient/RDAR/GameCoreNetwork_NOT1.cpp	Modified	

Showing 900 revision(s), from revision 102669 to revision 103568 - 1 revision(s) selected, showing 5 changed paths

☒ Show only affected paths

☒ Stop on copy/rename

☐ Include merged revisions

Show All Next 100 Refresh

Statistics Help OK

Growing superfast

- >100,000 revisions
- New source comitted every 1~2 min
- Beginning of 2011: **5,000** files
- End of 2012: **10,000** files
- End of 2013: **11,000** files

Growing superfast:
**Source commit
visualization
in a single day**

*gource visualization

Tuesday, www.fraps.com 10:31:39

hyunnam

19 official releases every month

- Korea: 4 releases / month
- Japan: 2 releases / month
- China: 2~4 releases / month
- U.S: 2 releases / month

2011년 10월

PATCH 검색(Ctrl+E)

	월요일	화요일	수요일	목요일	금요일	토요일	일요일
10월 3일	4일	5일	6일	7일	8일	9일	
10-3 - 10-8			일본 패치				
10일	11일	12일	13일	14일	15일	16일	
10-10 - 10-15	한국 패치 패치	미국 패치 한국 실습 패치	한국 실습 패치	한국 실습 패치	한국 패치 패치		
17일	18일	19일	20일	21일	22일	23일	
10-17 - 10-22	중국 패치 한국 패치 패치	일본 패치 한국 실습 패치	한국 실습 패치	한국 패치 패치			
24일	25일	26일	27일	28일	29일	30일	
10-24 - 10-29	한국 패치 패치		미국 패치 일본 패치 중국 패치				
31일	11월 1일	2일	3일	4일	5일	6일	
10-31 - 11-5	한국 실습						

Replacing wheels of a car while driving

Establishing integration process while live operation

Must not stop update releases with marketing plans

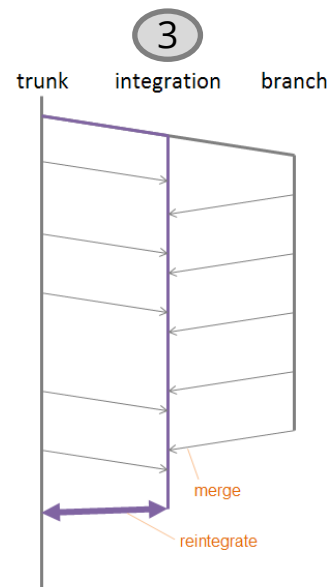
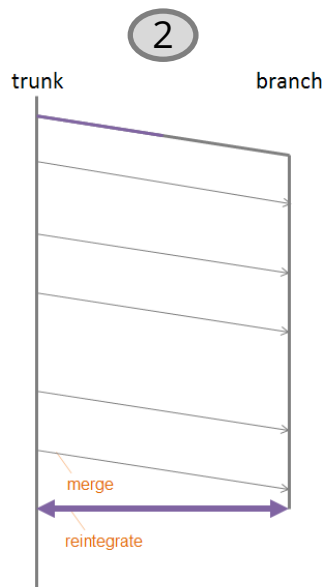
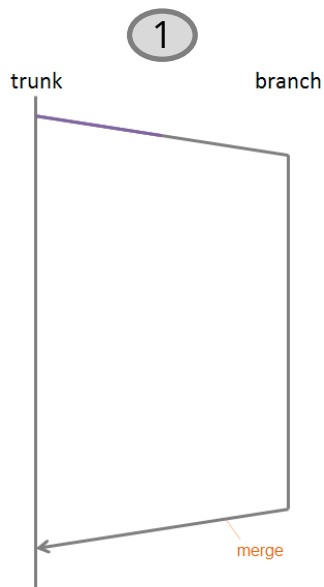


Long Story, Short Takeaways

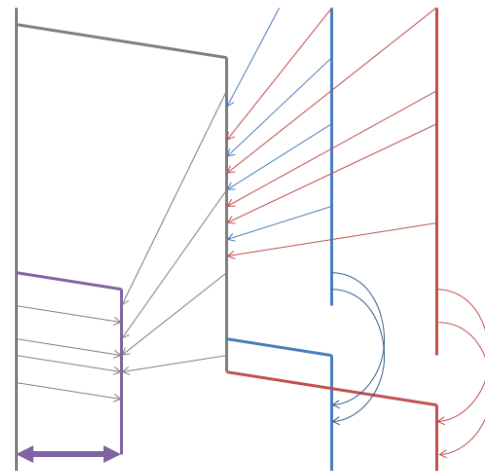
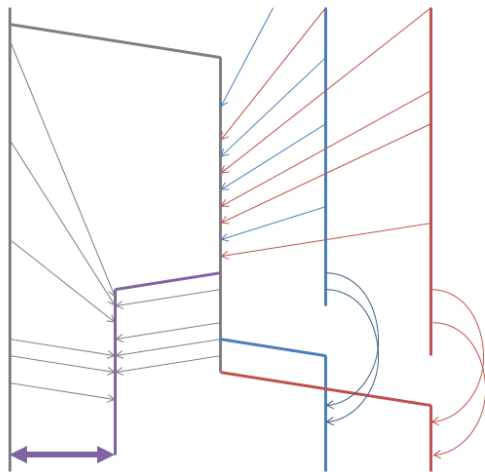
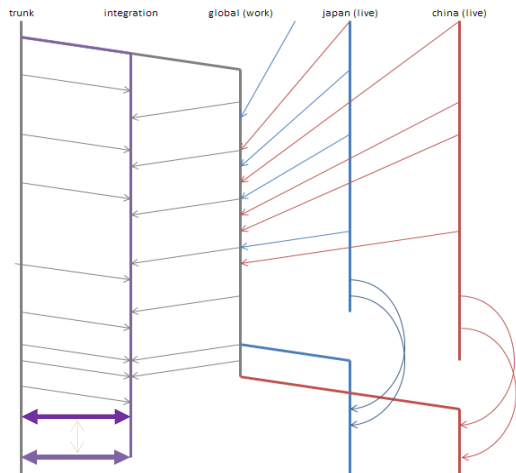
- Took 2~3 years to establish integrating pipeline
 - as a every 7 team's merge/test/integrating process
- DRY - Duplication is Evil
 - Do not underestimate the force of darkness and the evil
- To make large size of change / refactoring
 - Can't do big thing by self or with small number of people
 - Establish Roadmap in big picture, Share Vision
 - Make people talk and communicate about the long term need, and goal
 - Cooperate with influential people and leaders to follow the vision and goal
 - Persuade the opponent side
 - Plan to Minimize Risk as much as possible
 - Establish Roles and Responsibilities
 - especially for parts that have overlapped responsibilities between different organizations
 - Earn small-wins, Leverage them



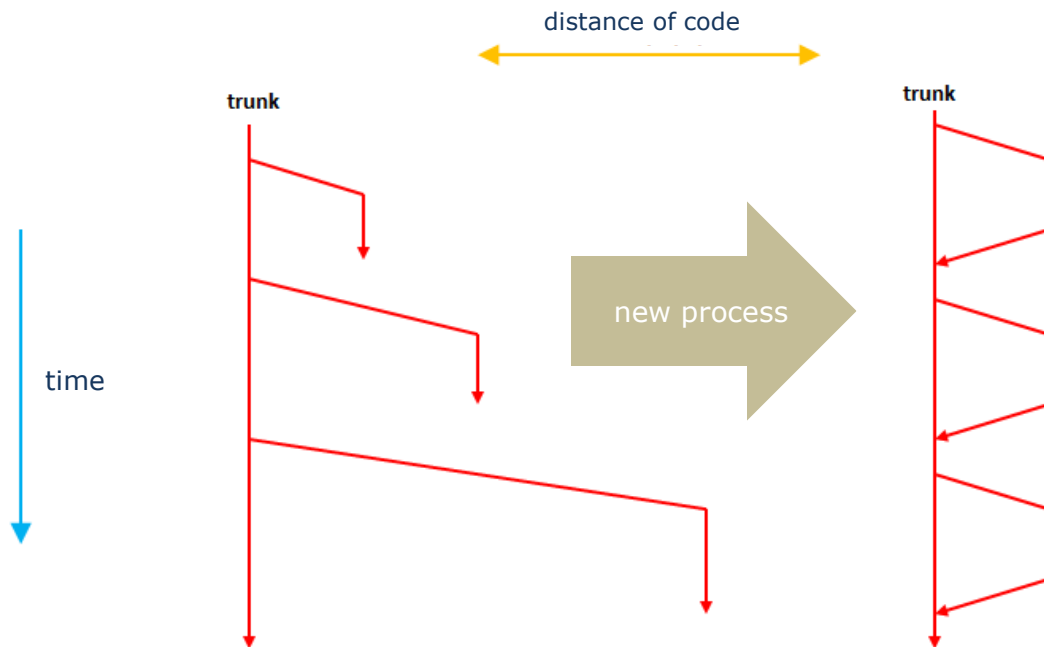
Short Tips : 3 ways to merge in SVN



Short tips: with integration branch, you can choose timing and speed to merge, integrate



Established Code Integration Process





Mitigating Technical Risk of Many Games for a long time (in reusable, efficient way)

What Nexon is doing more
For Maintaining Success for the Long Term

Nexon's Aged Live Games:

Games that Nexon is Running for a Long Time

Title	Release date	Age (as of Mar 2016)
Maple Story	Apr 2003	13 yrs
Mabinogi	Jun 2004	12 yrs
Atlantica	Feb 2008	7 yrs
Shaiya	Jun 2006	10 yrs
Vindictus	Jan 2010	5 yrs
Counter Strike Online	Dec 2007	9 yrs
Kart Rider	Jun 2004	12 yrs
Crazy Arcade BnB	Oct 2001	15 yrs
Nexus: The Kingdom of the Winds	Apr 1996	20 yrs
Dungeon and Fighter	Aug 2005	11 yrs

20 years!

>60 services
in different countries

Most games suffered from similar problems

- Most live teams were individually solving similar problems
 - Some of the solutions were not working properly
- In 2014, Made '*Live Infra Technology Team*'
 - Support 14 titles (60+ live game services)
 - Create efficient solutions to solve common technical problems
 - Focus to help live game services, while persuing Reusable/Scalability
 - Help future titles/services launches



Important problems in running live service

- For Game as a Service, mitigating risk is crucial to maintain its success
- Kind of common risks in game services:
 - Technical Risks (=Stability Issues)
 - Crash often
 - Memory error
 - Too laggy to play
 - Preventing client hacking (especially in PvP)
 - Server problems (cannot log in, unstable server)
 - Other risks
 - Loss of Virtual Properties (items, achievements, etc)
 - Negative press

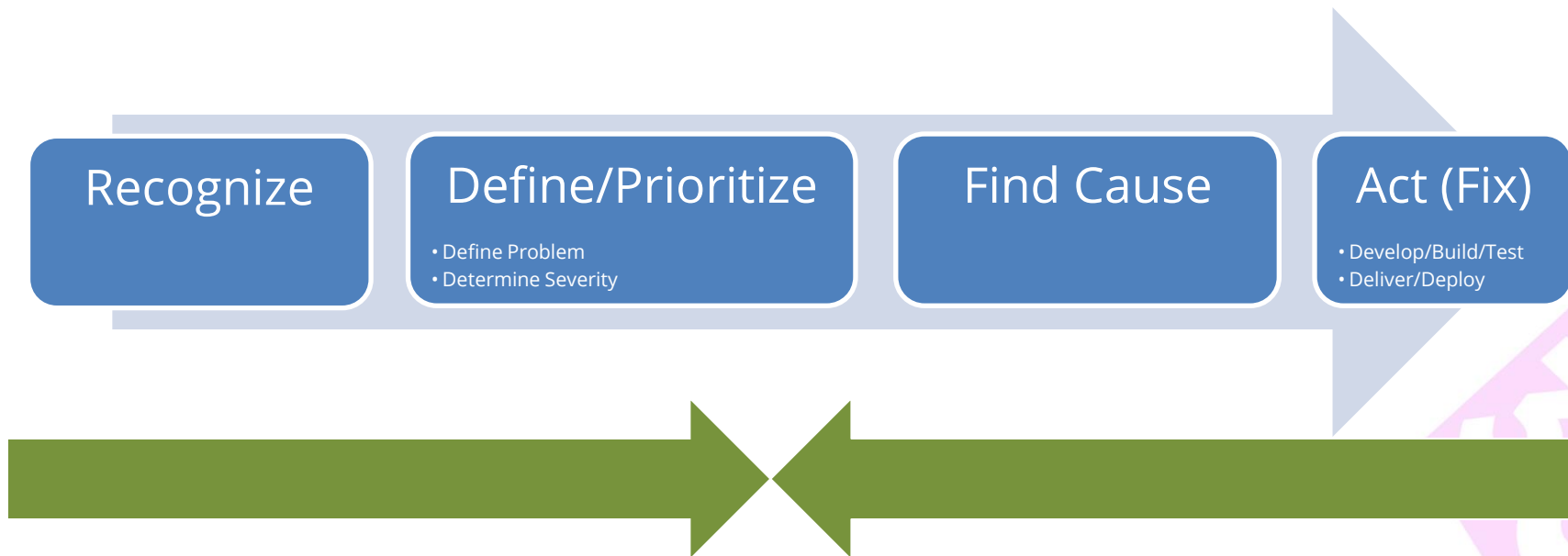


Stability Problems are important

- Stability Problems are big in online games because:
 - Users will leave the game when stability problem last for days
 - Users usually don't get a choice to upgrade or rollback
 - Stability unresolved = Dead Game→ Ensuring minimum stability baseline is very important
- Major Stability Issues
 - Crashes
 - Memory Problems
 - Lag Issues
 - Unstable Server
 - Hacking



How should we deal with the Risks



The lesser time & cost, The better

Major Stability Issues

Recognize

Prioritize

Find Cause

Fix

- Crashes
 - Memory Problems
 - Lag Issues
 - Unstable Server
 - Hacking
- (not this talk's scope)

Many of old PC games used manual tools

Recognize

Prioritize

Find Cause

Fix

- Collecting and Analyzing data was costful
 - Had to collect dump across servers
 - Required a lot of manual work and time to collect and analyze
 - Publisher, QA team requested crash reports if issued but not for every releases

	A	B	C	D	E
1	0181FAA4	443240	const NiTArray<class NiDX9AdapterDesc *,class NiTMallocInterface<class NiDX9AdapterDesc *	const NiTArray<class NSBD3D10TextureStage *,class NiTMallocInterface<class NSBD3D10Textu	const NiTPrimitiveArray<class NiTextureEffect const *
2	01821748	116488	const NiSPStream::vftable'		
3	01819280	114522	const NiBSplineCompTransfymEvaluator::vftable'		
4	0181F568	106653	const NiDX9LockableDataStream<class StaticLockPolicy<class NiDX9DataStream> >::vftable'(for 'NiObject')		
5	0181F540	106652	const NiDX9LockableDataStream<class StaticLockPolicy<class NiDX9DataStream> >::vftable'(for 'NiSPDataStream')		
6	01817750	95540	const NiTransformEvaluator::vftable'		
7	01837B30	92314	const NiTSPFixedInput<class NiMatrix3x4>::vftable'	const NiTSPStream<class NiTransform **>::vftable'	const NiTSPFixedInput<struct NiPSSimulatorFinalKerne
8	01761A04	77452	const cltPItemInfo_Client_SolNPC::vftable'		
9	01821DCC	60451	const NiTPointerList<class NiPointer<class NiRoom> >::vftable'	const NiTPointerList<class NiPointer<class NiDX92DBufferData> >::vftable'	const NiTPointerList<class NiPointer<class NSBShader
10	01818298	55509	const NiTransformData::vftable'		
11	01821CF0	55078	const NiTListBase<class NiTPointerAllocator<unsigned int>,class NiDynamicEffect *>::vftable'	const NiTListBase<class NiTPointerAllocator<unsigned int>,class NSBD3D10StateGroup::NSBD3	const NiTListBase<class NiTPointerAllocator<unsigned
12	018264E4	54653	const NiSemanticAdapterTable::vftable'		
13	018264E0	54647	const NiTArray<class NiSemanticAdapterTable::SemanticMapping,class NiTNewInterface<class I	const NiTObjectArray<class NiSemanticAdapterTable::SemanticMapping>::vftable'	
14	01834810	47497	const NiMesh::vftable'		
15	018335E4	37680	const NiTObjectArray<class NiPointer<class NiShaderDesc> >::vftable'	const NiTObjectArray<class NiPointer<class NiControllerSequence> >::vftable'	const NiTArray<class NiPointer<class NSBUserDefined
16	01823098	21206	const NiMaterialProperty::vftable'		
17	01822A7C	17024	const NiTexturingProperty::Map::vftable'		
18	01821724	16259	const NiSPTask::vftable'		
19	018216E0	16256	const NiSPTaskImpl::vftable'		
20	01822AB0	14496	const NiTexturingProperty::vftable'		
21	0175B334	12933	const NiRefObject::vftable'		
22	01833F20	12852	const NiStringExtraData::vftable'		
23	01822088	11831	const NiSourceTexture::vftable'		
24	01860DF4	11600	const NiTPointerList<class NiInputDevice::ControlDesc *>::vftable'	const NiTPointerList<class NiDynamicEffect *>::vftable'	const NiTPointerList<struct IUnknown *>::vftable'
25	01818068	11098	const NiEvaluatorSPData::vftable'		

Crash Reporting as a Service

Recognize

Prioritize

Find Cause

Fix

- For game as a services, Crash Reporting also should be a service
- everybody should be able to crash status
not only developers, but also QA team
- How to make it **more efficient and effective**
to live game services?



*In mobile games, App Stores usually have developers' console for crashes

Determining Severity Properly

Recognize

Prioritize

Find Cause

Fix

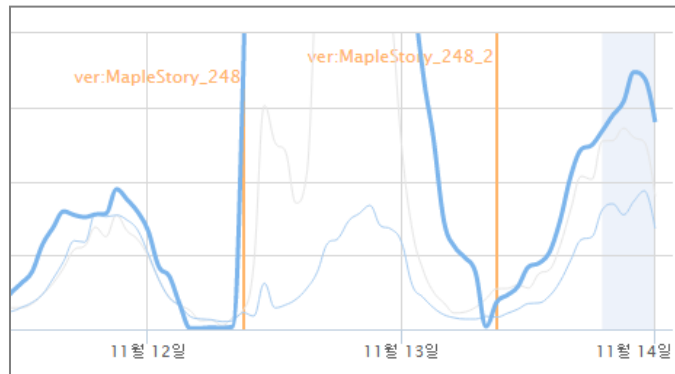
- There're many noises to determine severity properly
 - Out of memory crashes, C Runtime Errors weren't caught in general
 - Many crashes were being occurred after pressing Exit Game button - priorities different
 - Crash count fluctuated with DAU

Crash Report	
13th. NEW	EXCEPTION_ACCESS_VIOLATION(0xc0000005) Canvas+0x799 MapleStory!CBitmapStrNumber::~~CBitmapStrNumber+0x9 ntdll!RtlpInsertUCRBlock+0x4
61 (0.6%)	
12th.	EXCEPTION_ACCESS_VIOLATION(0xc0000005) MapleStory!CUIToolTip::~~CUIToolTip+0x5a8 [uitooltip.cpp @ 860] MapleStory!CIconButton::~~CIconButton+0x10d [ctrlbutton.cpp @ 58] MapleStory!CIconButton::~~CIconButton+0x10d [ctrlbutton.cpp @ 58]
~5 60 (0.6%)	
15th.	EXCEPTION_ACCESS_VIOLATION(0xc0000005) MapleStory!ZMapCUnsignedInt_com_ptr_t_com_IIID<IWzSoundState,&GUID_d9730ba4_23f5_4c2d_95d4_0_> MapleStory!CSoundMan::StopSE+0x54 [soundman.cpp @ 163] MapleStory!stop_skill_sound+0x12 [util.cpp @ 829]
~1 53 (0.5%)	
16th.	EXCEPTION_ACCESS_VIOLATION(0xc0000005) MapleStory!ZRecyclableAvBuffer<ZRefCountedDummy<com_ptr_t_com_IIID<IWzCanvas,&GUID_7600dc6c_9_> MapleStory!ZList_com_ptr_t_com_IIID<IWzCanvas,&GUID_7600dc6c_9328_4bff_9624_5b0f5c01179e> > MapleStory!ZList_com_ptr_t_com_IIID<IWzCanvas,&GUID_7600dc6c_9328_4bff_9624_5b0f5c01179e> > >
~4 51 (0.5%)	
17th.	EXCEPTION_ACCESS_VIOLATION(0xc0000005)

Monitoring Crashes

Is it still unstable?

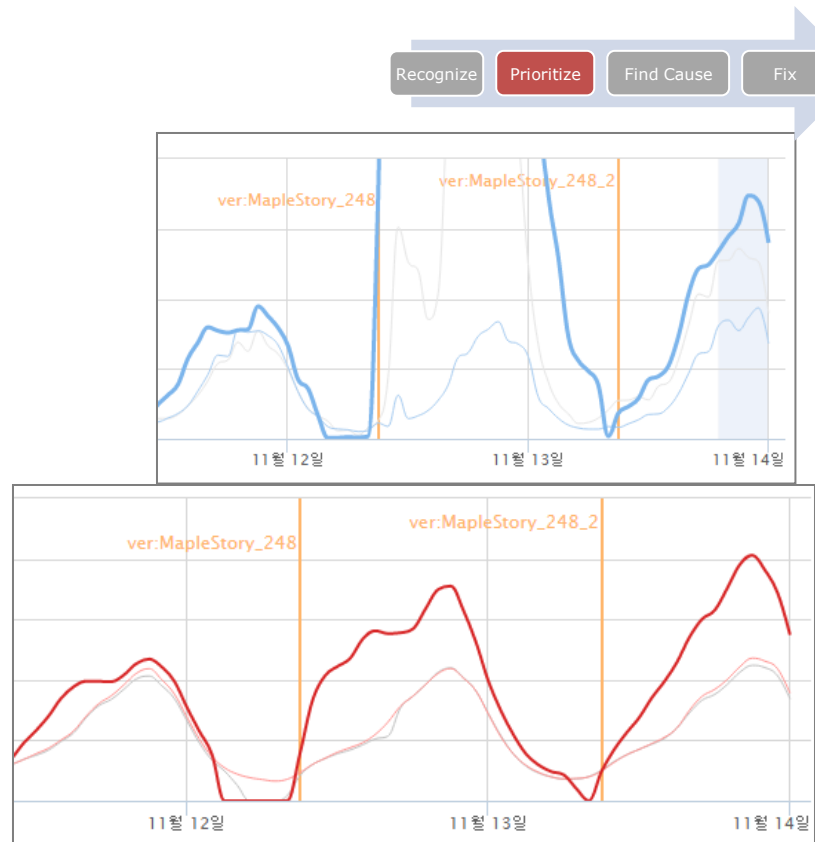
- After 12 Nov update, # of crashes increased a lot
- Released 13 Nov fix patch
- Still high on 14 Nov
- Is it still unstable?



Monitoring Crashes

Is it still unstable?

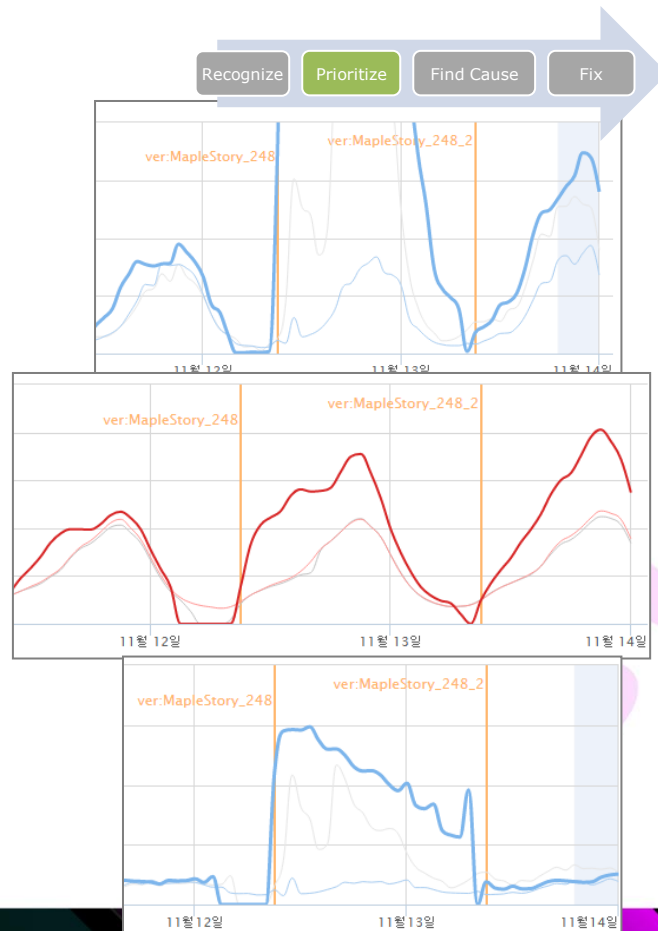
- Active Users increased a lot after 11/13 patch



Monitoring Crashes

Is it still unstable?

- Tried to reduce fluctuation noise regardless of active user volume
 - Crashes per session – better. still noisy
 - Crashes per hour – represents very well
- Both session and time are related but per unit time indicated stability better



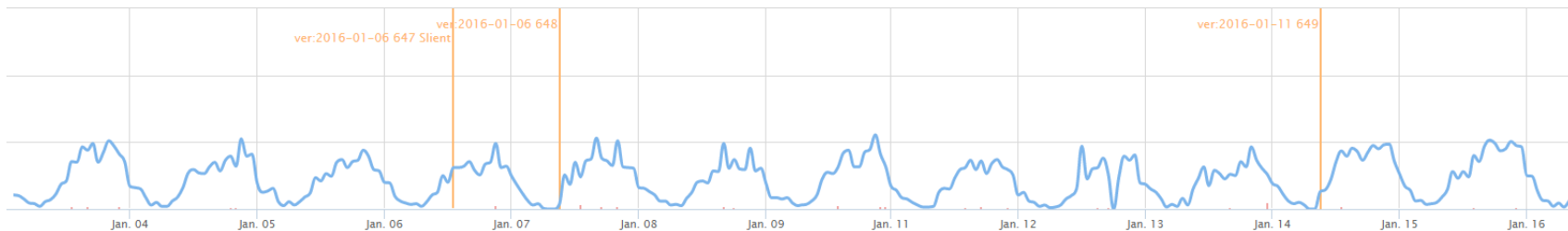
Avoiding Boiling Frog Syndrome

Recognize

Prioritize

Find Cause

Fix



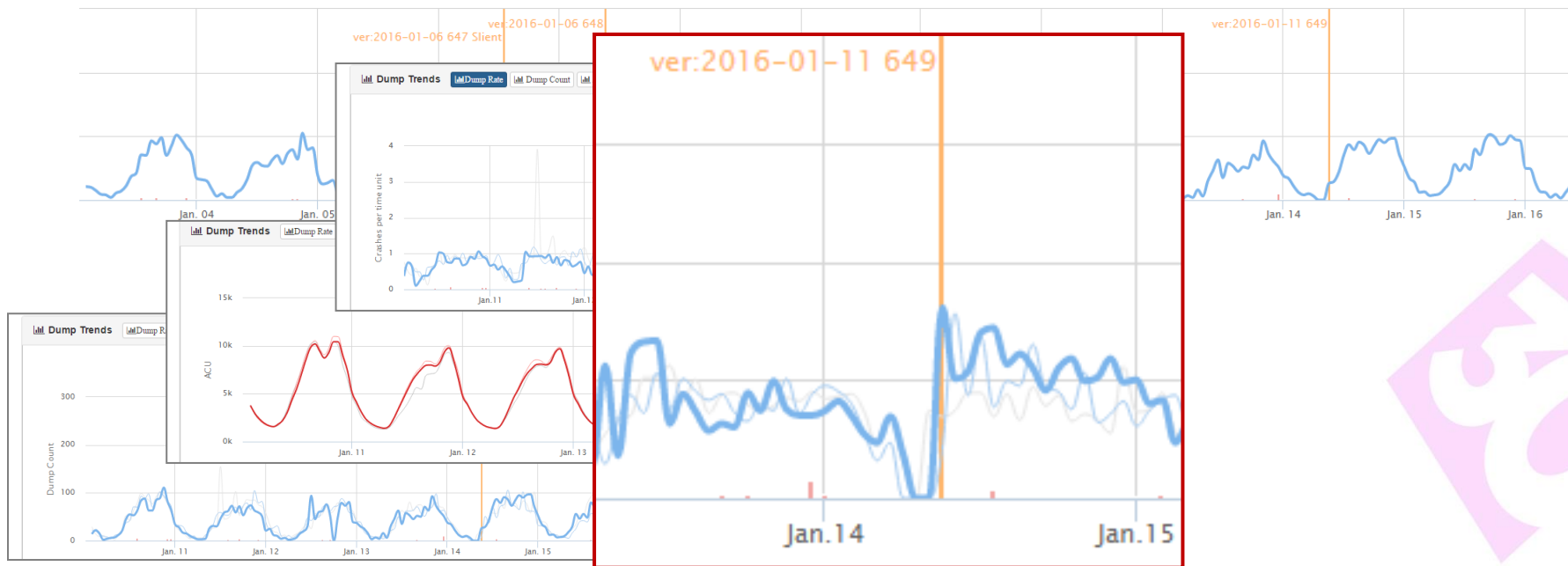
Avoiding Boiling Frog Syndrome

Recognize

Prioritize

Find Cause

Fix



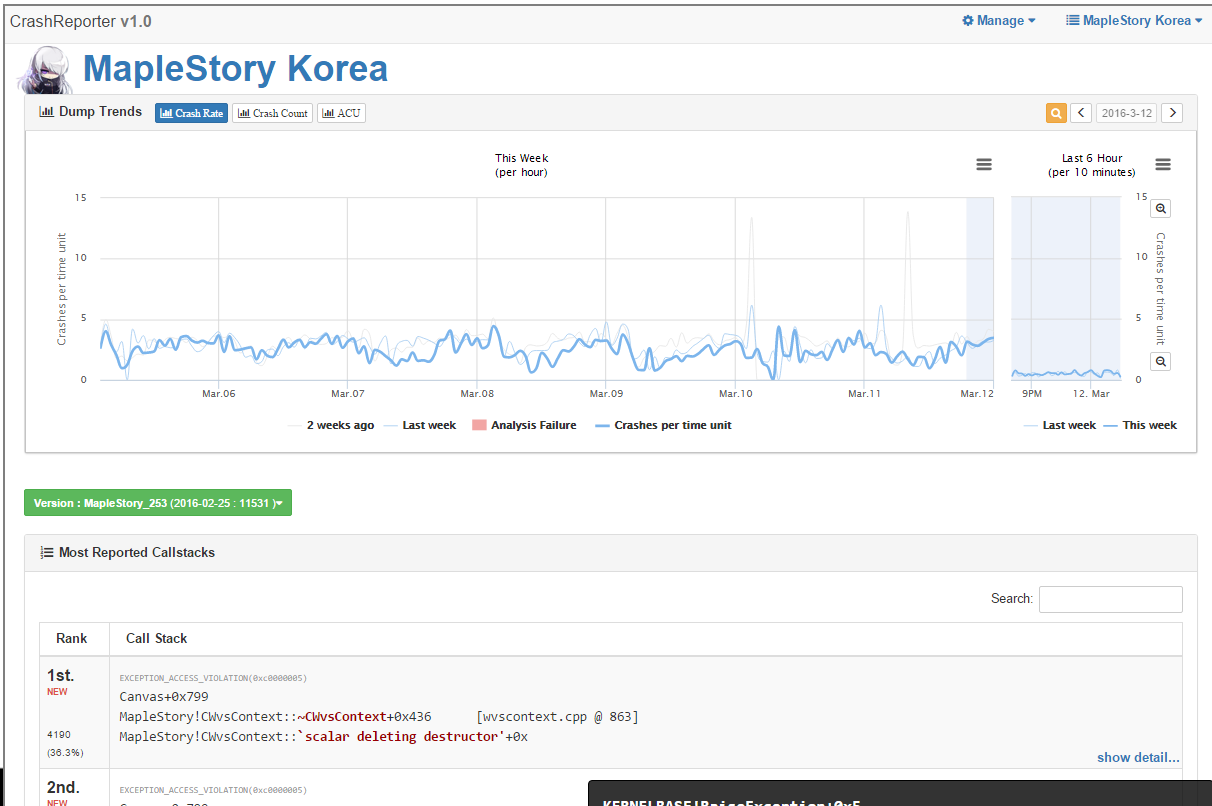
Made a Crash Report as a Service

Recognize

Prioritize

Find Cause

Fix



For many game services

Recognize

Prioritize

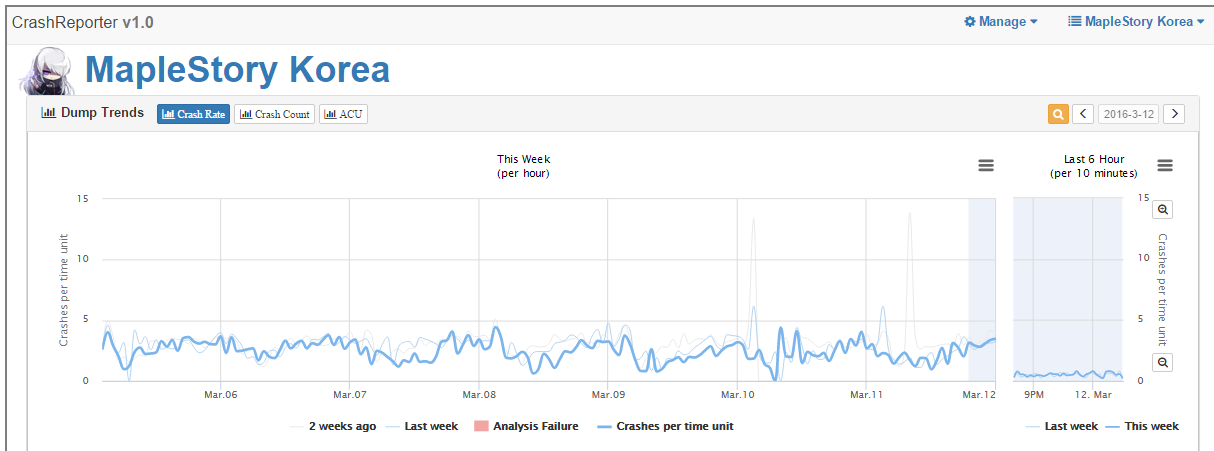
Find Cause

Fix

- 14 projects
- 54 crash dashboards
- Not only client
- But also
 - Game Servers
 - Dedicated Servers
 - Middlewares



Made a Crash Report as a Service



Version : MapleStory_253 (2016-02-25 : 11531)▼

☰ Most Reported Callstacks

Search:

Rank	Call Stack
1st <small>NEW</small>	EXCEPTION_ACCESS_VIOLATION(0xc0000005) Canvas+0x7f99 MapleStory!CWwsContext::~CWwsContext+0x436 [wvscontext.cpp @ 863]
4190 (30.3%)	MapleStory!CWwsContext::~scalar deleting destructor'+0x

[show detail...](#)

[show detail...](#)

Details

Recognize

Prioritize

Find Cause

Fix



MapleStory Korea

Version: MapleStory_253_2 (2016-03-03: 10473) +

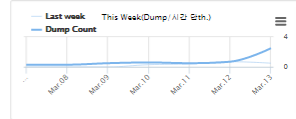
Most Reported CallStacks

Search:

Rank	Call Stack
1st.	UNKNOWN_EXCEPTION[0x00000000] (0x00000000) 682 (3.3%) 1. KERNELBASE!RaiseException+0x5 MapleStory!_CxxThrowException+0x48 [throw.cpp @ 161] MapleStory!_com_raise_error+0x38 [comraise.cpp @ 18]
2nd.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 421 (2.0%) 2. Gr2D_DX9+0xe93 PCOM+0x162b PCOM+0x162b
2rd.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 368 (3.8%) 3. Canvas+0x799 ntdll!RtlpInsertUCRBlock+0x4 ntdll!RtlpInsertUCRBlock+0x4
2th.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 312 (3.2%) 4. Gr2D_DX9+0xe91 PCOM+0x162b PCOM+0x162b
5th.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 281 (2.7%) 5. Gr2D_DX9+0xe91 PCOM+0x162b PCOM+0x162b
6th.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 220 (2.1%) 6. MapleStory!CWndMan::s_Update+0x4a [wndman.cpp @ 782] MapleStory!_dyn_tls_init_callback <PERF> (MapleStory+0x1e38786 MapleStory!_dyn_tls_init_callback <PERF> (MapleStory+0x1e410e9)
7th.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 181 (1.7%) 7. lgdumd32+0x4f5 lgdumd32+0x51c
8th.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 121 (1.2%) 8. MapleStory!ZRecyclableAvBuffer+ZMap<long,ZRef<CItemInfo::EQUIPITEM>,long>::PAIR,16,ZMap<long,ZR... MapleStory!ZMap<long,ZRef<CItemInfo::EQUIPITEM>,long>::Insert+0x6a [_zcoll.inl @ 1856] MapleStory!CItemInfo::RegisterEquipItemInfo+0x3c3 [iteminfo.cpp @ 2158]
9th.	UNKNOWN_EXCEPTION[0x00000000] (0x00000000) 106 (1.0%) 9. KERNELBASE+0xb3fc MapleStory!_CxxThrowException+0x48 [throw.cpp @ 161] MapleStory!_com_raise_error+0x38 [comraise.cpp @ 18]
10th.	EXCEPTION_ACCESS_VIOLATION[0x00000000] 89 (0.8%) 10. 0x148e5 0x10844

Details

Search:

66 Rank
8th. (1.16%)Exception Code
EXCEPTION_ACCESS_VIOLATION(0xc0000005)First Occurrence
Version: MapleStory_248
Date: 2015-11-12 09:44:15

Call Stack

```

MapleStory!ZRecyclableAvBuffer+ZMap<long,ZRef<CItemInfo::EQUIPITEM>,long>::PAIR,16,ZMap<long,ZRef<CItemInfo::EQUIPITEM>,long>::Insert+0x6a [_zcoll.inl @ 1856]
MapleStory!CItemInfo::RegisterEquipItemInfo+0x3c3 [iteminfo.cpp @ 2158]
MapleStory!CItemInfo::GetEquipItem+0x11d [iteminfo.cpp @ 1871]
MapleStory!CItemInfo::GetEquippedEmotion+0xa [iteminfo.cpp @ 13470]
MapleStory!CAvatar::NotifyAvatarModified+0x723 [avatar.cpp @ 443]
MapleStory!CAvatar::Init+0xaf [avatar.cpp @ 282]
MapleStory!CAvatarMegaphone::OnCreate+0x327 [avatarmegaphone.cpp @ 63]
MapleStory!CWnd::CreateWnd+0x59e [wnd.cpp @ 69]
MapleStory!CAvatarMegaphone::CAvatarMegaphone+0x17d [avatarmegaphone.cpp @ 48]
MapleStory!CWndContext::OnSetAvatarMegaphone+0x22e [wndcontext2.cpp @ 11895]
MapleStory!CWndContext::OnPacket+0x94e [wndcontext.cpp @ 5639]
MapleStory!_dyn_tls_init_callback <PERF> (MapleStory+0x1e4b47e
0xa68c827

```

Dump List

Search:

Crash Time	Download	Account	Character	IP	Memory Usage	Call Stack
2016-03-13 00:38:22				tsdiekgm72	58.127.108.183	1225 MB
2016-03-13 00:16:19				c***tm1***	118.39.115.91	1270 MB
2016-03-13 00:12:23				h**55***	118.130.100.177	1002 MB
2016-03-13 00:08:01				-	121.176.174.248	1015 MB
2016-03-13 00:03:32				bantell	211.202.108.23	833 MB
2016-03-12 22:59:03				l**jm961128***	122.32.105.104	1283 MB
2016-03-12				-	219.248.94.248	1114 MB

Details



MapleStory Korea

Version : MapleStory_253.2 (2016-03-03 : 10473) +

Recognize

Prioritize

Find Cause

Fix

Details

Rank
8th. (1.16%)

Exception Code
EXCEPTION_ACCESS_VIOLATION(0xc0000005)

First Occurrence
Version:: MapleStory_248
Date: 2015-11-12 09:44:15

Call Stack

```
MapleStory!ZRecyclableAvBuffer<ZMap<long, ZRef<CItemInfo::EQ
MapleStory!ZMap<long, ZRef<CItemInfo::EQ
MapleStory!CItemInfo::RegisterEquipItem
MapleStory!CItemInfo::GetEquipItem+0x11
MapleStory!CItemInfo::GetEquippedEmotion
MapleStory!CAvatar::NotifyAvatarModified
MapleStory!CAvatar::Init+0xaf [avatar
MapleStory!CAvatarMegaphone::OnCreate+0
MapleStory!CWnd::CreateWnd+0x58e[wnd.cp
MapleStory!CAvatarMegaphone::CAvatarMeg
MapleStory!CWndContext::OnSetAvatarMega
MapleStory!CWndContext::OnPacket+0x94e
MapleStory!__dyn_tls_init_callback <PER
```

search Q



Details

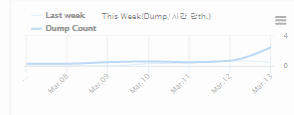
Rank
8th. (1.16%)

Exception Code
EXCEPTION_ACCESS_VIOLATION(0xc0000005)

First Occurrence
Version: MapleStory_248
Date: 2015-11-12 09:44:15

Call Stack

```
MapleStory!ZRecyclableAvBuffer<ZMap<long, ZRef<CItemInfo::EQ
MapleStory!ZMap<long, ZRef<CItemInfo::EQ
MapleStory!CItemInfo::RegisterEquipItem
MapleStory!CItemInfo::GetEquipItem+0x11
MapleStory!CItemInfo::GetEquippedEmotion
MapleStory!CAvatar::NotifyAvatarModified
MapleStory!CAvatar::Init+0xaf [avatar
MapleStory!CAvatarMegaphone::OnCreate+0
MapleStory!CWnd::CreateWnd+0x58e[wnd.cp
MapleStory!CAvatarMegaphone::CAvatarMeg
MapleStory!CWndContext::OnSetAvatarMega
MapleStory!CWndContext::OnPacket+0x94e
MapleStory!__dyn_tls_init_callback <PER
```



Dump List

Search:

Crash Time	Download	Account	Character	IP	Meomry Usage	Call Stack
2016-03-13 00:38:22		tlsldekjm72	장인공만일	58.127.168.183	1225 MB	
2016-03-13 00:16:19		c**tm1*@**	진규	118.39.115.91	1279 MB	
2016-03-13 00:12:23		h**55*@**	황학칠새	118.130.169.177	1002 MB	
2016-03-13 00:08:01		-	Rit쓰	121.176.174.248	1015 MB	
2016-03-13 00:03:32		bantell	80아라	211.202.168.23	633 MB	
2016-03-12 22:59:03		l**jm961126*@**	살인카이저	122.32.185.104	1283 MB	
2016-03-12 22:25:20		-	xxekzmfkdlxx	219.248.94.248	1114 MB	

Search:

Account	Character	IP	Meomry Usage	Call Stack
tlsldekjm72	장인공만일	58.127.168.183	1225 MB	
c**tm1*@**	진규	118.39.115.91	1279 MB	
h**55*@**	황학칠새	118.130.169.177	1002 MB	
Rit쓰	Rit쓰	121.176.174.248	1015 MB	
80아라	80아라	211.202.168.23	633 MB	
살인카이저	살인카이저	122.32.185.104	1283 MB	
xxekzmfkdlxx	xxekzmfkdlxx	219.248.94.248	1114 MB	

Monitoring Crashes

Adding Notification and Alarms

Recognize

Prioritize

Find Cause

Fix

Automatically detects patch
and sends notification mail
after 2 hours

Some kind of dumps
are sent immediately
(Server dumps, from internal test)



2016-01-01 (금) 오전 4:18

_인프라기술팀 [infratech]

[크래시리포트] 패치 후 크래시 현황보고(카스온라인/한국)

받는 사람: @_CSO실 클라이언트직군 [cso-client], @_인프라기술팀 [infratech]

* 본 메일은 시스템을 통해 자동으로 발송되는 메일입니다.



카스온라인/한국 프로덕트의 (VER_2015_12_17_03/Client) 버전이 패치되었습니다.
버전의 처음 발생시간(2016-01-01 02:17:01)으로 부터 2시간이 지나 덤프 추세 현황 공유드립니다.



Monitoring Crashes

Adding Notification and Alarms

Recognize

Prioritize

Find Cause

Fix

If crash rate is increased
a lot more than usual,
it immediately sends mail
within 2 hours



2015-12-16 (수) 오후 10:07

_인프라기술팀 [infratech]

[크래시리포트] 패치 직후 크래시량 증가현상 알림(컴벳앰즈/유럽)

받는 사람: _인프라기술팀 [infratech]

① 중요도가 높은인 메시지를 보냅니다.

* 본 메일은 시스템을 통해 자동으로 발송되는 메일입니다.

크래시리포트
페이지로 이동

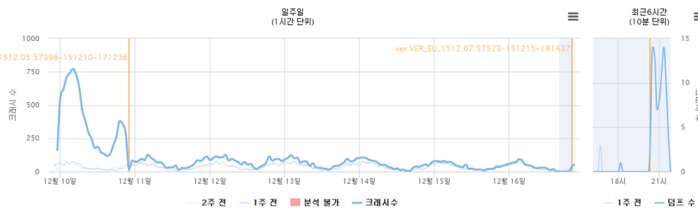
컴벳앰즈/유럽 프로젝트의 (VER_EU_1512.07.57523-151215-181437/Client)버전에 패치되었습니다.
버전 릴리즈 후 1시간 40분이 지난 현재 **다수(90개)의 크래시가 발생 하고있습니다.** (패치 후 2시간 동안 평균 일보계수:8개)

• **크래시 수가 기존 패치 직후 평균보다 21% 많습니다. 개발팀에서는 패치된 버전의 크래시 증가현상에 대해 확인해주시기 바랍니다.**
측정된 값은 단순 크래시 개수로, 통일이 되어나오는 등의 현상에 의해서도 크래시 개수가 달라질 수 있으므로 참고 바랍니다.



컴벳앰즈 유럽

내 팀의 그래프 (1주일 / 2시간 / 1시간)



최근 섹션: VER_EU_1512.07.57523-151215-181437 (2015)

팀 디프 리스트

Search:

순위	콜스택
1위	cplushook+0x5d2
+26	cplushook+0x1a14

Monitoring Crashes:

Sometimes Call Stack is not enough..

Recognize

Prioritize

Find Cause

Fix

- When broken state caused crash later
 - Heap Corruption
 - Dangling Pointer
- If it is not reproduced, it is sometimes very hard to find cause

Monitoring Crashes:

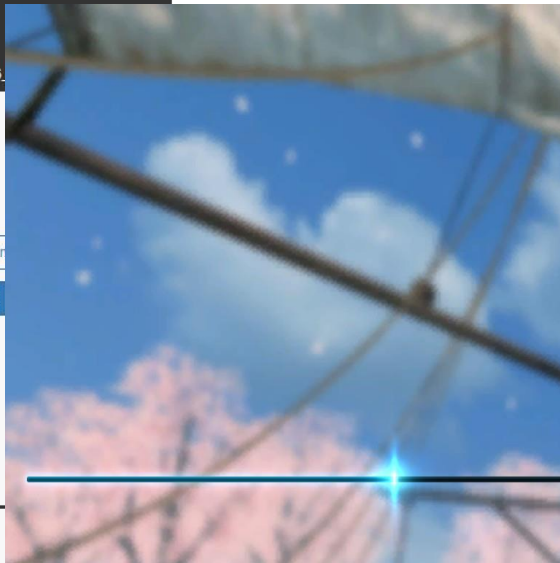
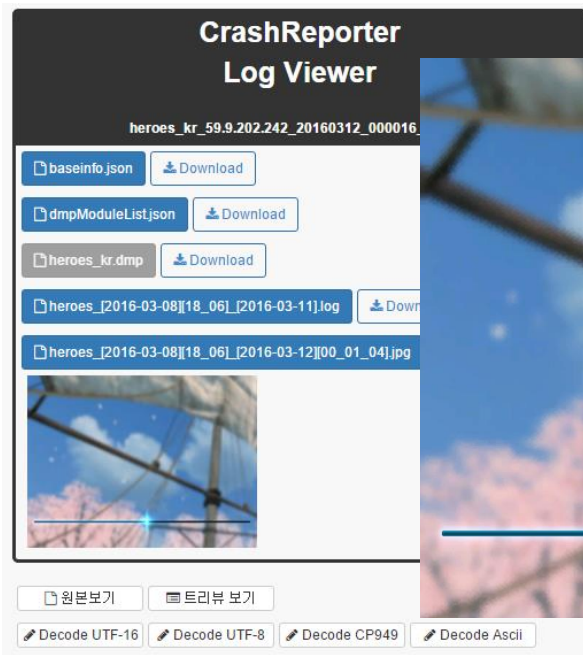
Provided more detailed informations

Recognize

Prioritize

Find Cause

Fix



```
baseinfo.json
BaseCharacterType : "Arisha"
CHAR_NAME : "IKrewella"
CID : 635429979636306600
CPU_INFO : "Intel(R) Celeron(R) CPU G1850 @ :
CRASH_BEHAVIOR_BITFLAG : 13
CR_VERSION : "V1.2.0.1"
DUMP_TYPE_BITFLAG : 3
ENCODING : "UNICODE"
Level : 80
MEMORY_PEAK_USAGE : "1403 MB"
MEMORY_PHYSICAL : "3714 MB"
MEMORY_USAGE : "727 MB"
OS_BUILDNUMBER : "9200"
OS_HAS_CHINAIME : false
OS_LOCALE_SYSTEM : "한국어(대한민국)"
OS_LOCALE_USER : "한국어(대한민국)"
OS_PRODUCTTYPE : "1"
OS_SERVICEPACK_MAJOR : "0"
OS_SERVICEPACK_MINOR : "0"
OS_SUITEMASK : "0x00000300"
OS_VERSION_MAJOR : "6"
OS_VERSION_MINOR : "2"
OS_VERSION_NAME(log_viewer전용) : "Window
OS_WOW64 : false
PLAY_TIME : 26079
UTC_CRASH : "2016-03-11 15:01:04"
```

*Type of data to collect varies for different game and countries under game policy and user agreements.

Screenshots give clues

Recognize

Prioritize

Find Cause

Fix



*Type of data to collect varies for different game and countries under game policy and user agreements.

Monitoring Crashes:

Don't require to ask player to send client log

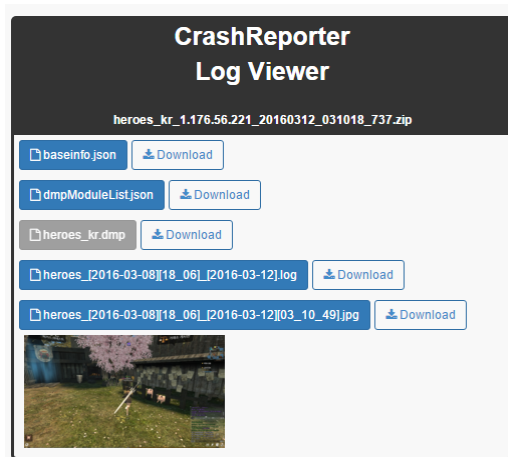
Recognize

Prioritize

Find Cause

Fix

```
L 03/12/2016 - 00:41:10: #####
L 03/12/2016 - 00:41:10: Log file started (version "6331")
L 03/12/2016 - 00:41:10: #####
L 03/12/2016 - 00:41:10:
L 03/12/2016 - 00:41:11: Shader_Connect
L 03/12/2016 - 00:41:11: ConnectStudioRender
L 03/12/2016 - 00:41:11: IDataCacheFactory
L 03/12/2016 - 00:41:11: IPhysicsFactory
L 03/12/2016 - 00:41:11: IMDLCacheFactory
L 03/12/2016 - 00:41:11: IMatSystemSurfaceFactory
L 03/12/2016 - 00:41:11: IAviFactory
L 03/12/2016 - 00:41:11: IHammerFactory
L 03/12/2016 - 00:41:11: ConnectMDLCacheNotify
L 03/12/2016 - 00:41:12: EngineInit
L 03/12/2016 - 00:41:12: RunListenServer
L 03/12/2016 - 00:41:12: TraceInit : COM_InitFilesystem( pModName )
L 03/12/2016 - 00:41:13: TraceInit : MapReslistGenerator_Init()
L 03/12/2016 - 00:41:13: TraceInit : DevShotGenerator_Init()
L 03/12/2016 - 00:41:13: materials->ModInit
L 03/12/2016 - 00:41:13: Host_ReadPreStartupConfiguration
L 03/12/2016 - 00:41:14: NMCOGame::_nmman.Init()
L 03/12/2016 - 00:41:19: NMCOGame::_nmman.SetLocaleAndRegion()
L 03/12/2016 - 00:41:20: NMCOGame::_nmman.Initialize()
L 03/12/2016 - 00:41:24: NMCOGame::NMGameLogManager.Initialize()
L 03/12/2016 - 00:41:24: nmcoGame->Init()
L 03/12/2016 - 00:41:24: endPointNetwork->Init()
L 03/12/2016 - 00:41:24: mmoEndPointNetwork->Init()
L 03/12/2016 - 00:41:24: AppSystem::RegisterHandler<ServerCmdMessage>
L 03/12/2016 - 00:41:25: AppSystem::RegisterHandler<ClientCmdHandler>
L 03/12/2016 - 00:41:25: AppSystem::RegisterHandler<SyncFeatureMatrixMessageHandler>
L 03/12/2016 - 00:41:25: AppSystem::RegisterHandler<IGSSecurityMessageHandler>
```



*Type of data to collect varies for different game and countries under game policy and user agreements.

Jupyter connected

- Can do any other analysis just-in-time:
(powered by python & Jupyter)
 - How much of RAM do crashed users have
 - Check whether crashes is occurring more to specific users



Monitoring Crashes

Helped Live Service & Live dev team!

Recognize

Prioritize

Find Cause

Fix

Helped live dev team to concentrate more
on developing game
without being distracted with live issues

Major Stability Issues

- Crashes
- **Memory Problems**
- Lag Issues



Memory Problem:

Inevitable when running game for a long time

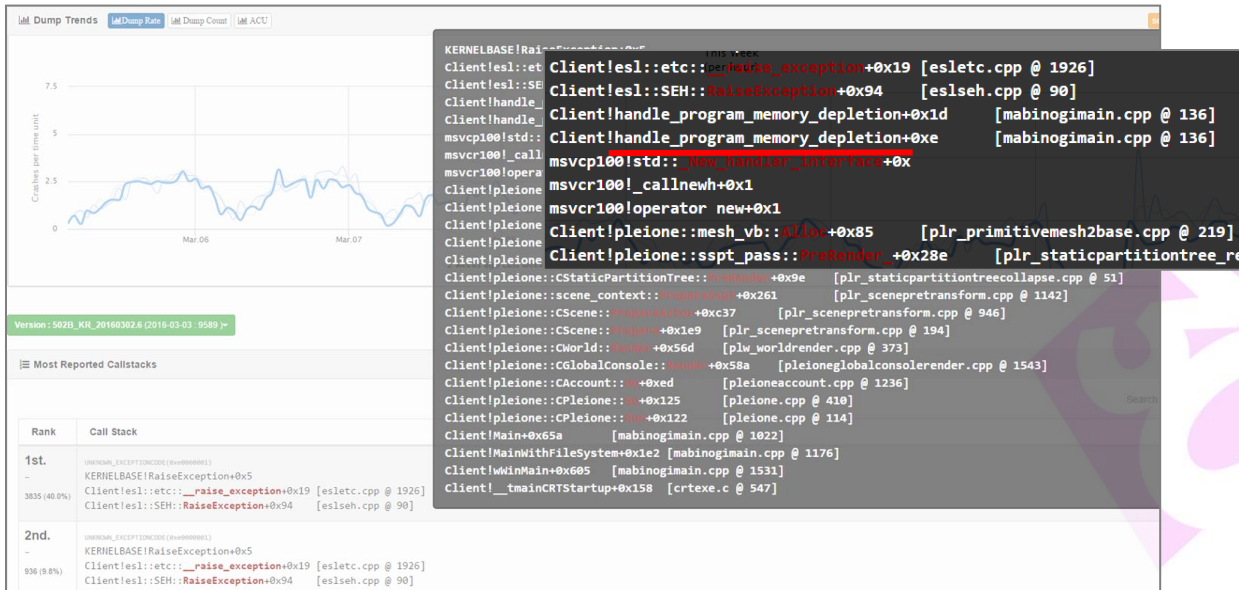
- As contents accumulated, most games suffered out of memory problem even with plenty of RAMs
 - 32 bit process memory address space limitation: 2GB
 - 2GB is not enough for many module images and resources including UI images, textures, animations and sounds
- /LARGEADDRESSAWARE helped, but an one-time opportunity
(allows process to use 3GB of user memory address)
- Especially with many of high-res textures, or many displayed characters:
 - 2D animation with many frames
 - MMORPG with various character costumes
 - Lots of background/UI/Illustration images

Memory Problem:

Many of Crashes are because of Out of Memory

- Most game projects operating more than 5 years had plenty of out-of-memory problems

Most Reported Callstacks	
Rank	Call Stack
1st (14.2%)	UNKNOWN_EXCEPTIONCODE (0xa0000002) KERNELBASE!RaiseException+0x5 engine!Sys_Error+0xc0 [sys_d11.cpp @ 401] engine!MemAllocFailHandler+0x32 [sys_d112.cpp @ 2038]
2nd (8.7%)	UNKNOWN_EXCEPTIONCODE (0xa0000002) kernel32!RaiseException+0x5 engine!Sys_Error+0xc0 [sys_d11.cpp @ 401] engine!MemAllocFailHandler+0x32 [sys_d112.cpp @ 2038]
3rd (3.2%)	UNKNOWN_EXCEPTIONCODE (0xa0000002) KERNELBASE!RaiseException+0x5 engine!Sys_Error+0xc0 [sys_d11.cpp @ 401] engine!MemAllocFailHandler+0x32 [sys_d112.cpp @ 2038]
4th (1.2%)	UNKNOWN_EXCEPTIONCODE (0xa0000002) engine!Sys_Error+0xc0 [sys_d11.cpp @ 401] engine!Sys_SpinFunc+0x18f [sys_d11.cpp @ 590]
5th (1.2%)	UNKNOWN_EXCEPTIONCODE (0xa0000002) KERNELBASE!RaiseException+0x5 engine!Sys_Error+0xc0 [sys_d11.cpp @ 401] engine!Sys_SpinFunc+0x18f [sys_d11.cpp @ 590]
6th (1.2%)	UNKNOWN_EXCEPTIONCODE (0xa0000002) KERNELBASE!RaiseException+0x5 engine!Sys_Error+0xc0 [sys_d11.cpp @ 401] engine!Sys_SpinFunc+0x18f [sys_d11.cpp @ 590]



Recognize

Prioritize

Find Cause

Fix

Memory Problem:

Tracking Memory wasn't easy for PCs

- Especially for many old games didn't have good engine with resource/memory tracking functionality.
- Reports based on Task Manager used to have noises
 - XP Task Manager displayed "working set" which doesn't count paged-out memories
- Size gap was big between "Private Bytes" and tracked memory size, Needed to narrow the gap
 - Using '`_CrtSetAllocHook`' and Overriding new didn't have full coverage (e.g. DirectX, GUI middlewares, Sound library)
 - Performance Counter did not represent application level memory usage, but showed memory page usage
 - did not count: free & allocate again, pooling
 - Couldn't distinguish between graphics, sound or cache usages

Memory Problem:

Limitations Using existing Memory Profiler

- Tried Memory Validator, GlowCode, Visual Leak Detector
- Need to 'play' game, without running too slowly
- Client Security Solution denied access to client process, including Visual Studio
 - Need to track real-server-situation with many of active users
 - Many problems were not reproduced with test doll characters
e.g. realm vs. realm battle, village with many people doing many things
- Should handle large amount of allocation data to track out-of-memory situation
 - Most used a lot of memory to keep result
 - Used to crash because of out of memory while profiling
- Want to compare certain two points of time in detail
 - To check leaks between certain two point of time (ex. between after second battle and third battle)
 - Or to find cause of the drastic increase
- Want to track memory-pooled objects

Recognize

Prioritize

Find Cause

Fix

Memory Problem: Made own Memory Profiler for DNF

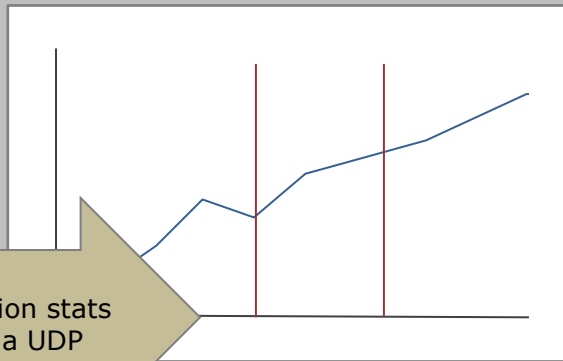
Game Client (32bit Native Application)

in-process
Memory Profiler Module

- hooks HeapAlloc API
- callback from Memory Pool Manager

sends
memory allocation stats
periodically via UDP

Profile Result Viewer (64bit .NET application)



```
bool LoadAnimation(..) {  
    ...  
    If (LoadImage(imageName) == false) {  
        HandleError();  
        return false;  
    }  
    ...  
}  
  
LoadAnimation(..)  
LoadImage(..)  
CreateTextureFromMemory(..)  
..  
HeapAlloc(..)
```

Rank	Stack	Total Bytes Allocated	Total Count of allocation	Avg. Allocation Size
1	LoadAnimation():12	xxx,xxx,xxx	x,xxx	x,xxx,xxx
2	LoadBackground():23	xxx,xxx,xxx	x,xxx	x,xxx,xxx
3	..	xx,xxx,xxx	xxx	x,xxx,xxx
..

Memory Problem:

Made own Memory Profiler for DNF

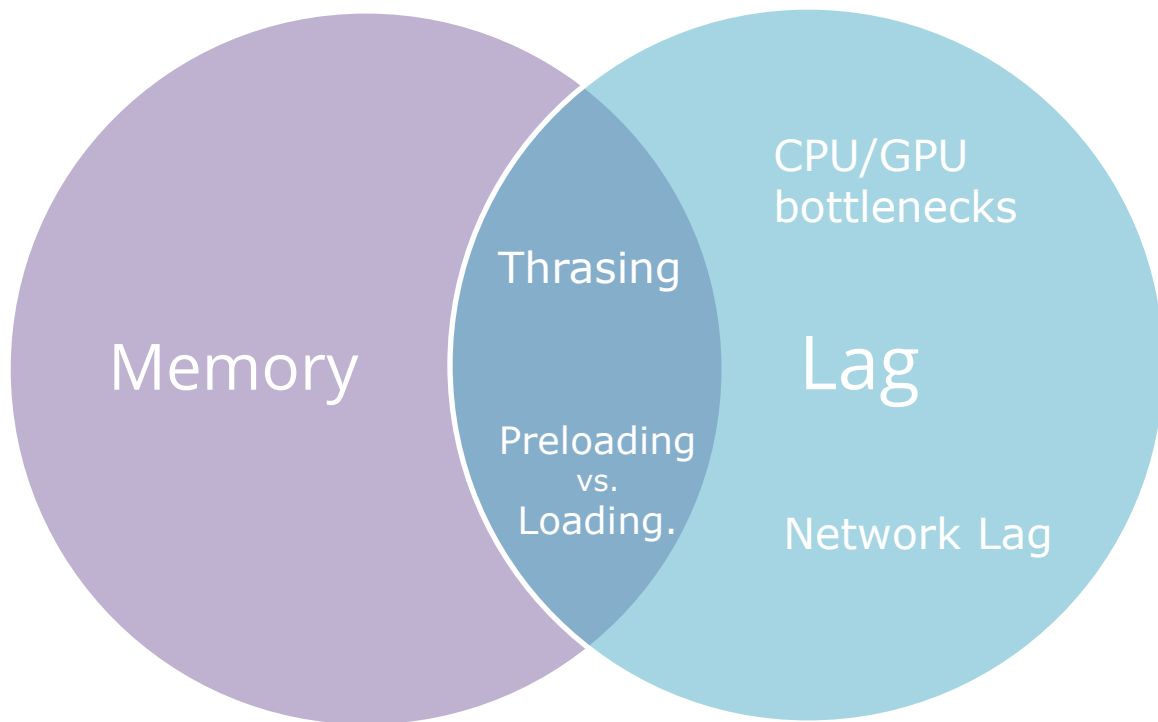
- Project-independent but in-process memory profiler
 - Just one initialize call hooks win32 HeapAlloc
 - Pooled object was monitored via one line of simple callback
 - Sends allocation stats to aggregator/analyzer server (64-bit .NET application) via UDP periodically (fast / no insufficient memory)
- Runtime switches to trade off performance and detail (fast)
- Usable in real environment with client security solution
- Can track external library memory usage (DirectX textures, UI middlewares, sound library, ..)
- Can diff memory blocks and its call stack between marked point
- Very useful to track untracked memory gap between performance counter and in-game memory tracking, and various leaks as well
- Plan: Making a generic memory profiler for live games as a service ready-to-use at any time to developers.
 - Process-independent solution – reusable in many live games in many cases.

Major Stability Issues

- Crashes
- Memory Problems
 - **Memory-Lag overlapped Problem**
- Lag Issues



Memory, Lag - Overlapped Problem



Memory vs. Lag trade-offs



Shorter Stage Loading Time	Longer Stage Loading Time
Less Memory	More Memory Causes Out of memory even with plenty of RAM (insufficient 32bit Address Space) Causes Thrashing with little RAM
More in-game Loading Lag (Can't use background loading sometimes)	Less in-game Loading Lag
	Need Preload Resources, Need to find their reverse-dependency easily.

Recognize

Prioritize

Find Cause

Fix

Preload manager for DNF

- Challenge: Hard to determine likely-used resources
 - Didn't have game engine or resource manager with unified resource references
 - Many of resource reference was not in declarative form
- Suggested to build preload list based on actual resource usage log
 - method used in Bubble Fighter
- AI/Skill-related resources were especially hard for DNF
 - Some were not bound to single animation
 - DNF had variety of AI/skills (> 2000 skills) which were not represented in declarative form

Building Preload List automatically

- Logged every player's event in Test Server for couple of weeks
 - 77,500,000 image file loadings
 - 1,400,000 skill events

```

    @Override
    public void start() {
        // Start the main loop
        while (true) {
            // Get the current time
            long currentTime = System.currentTimeMillis();

            // Check for new messages
            if (currentTime - lastMessageTime > 1000) {
                lastMessageTime = currentTime;
                // Process the message
                processMessage();
            }

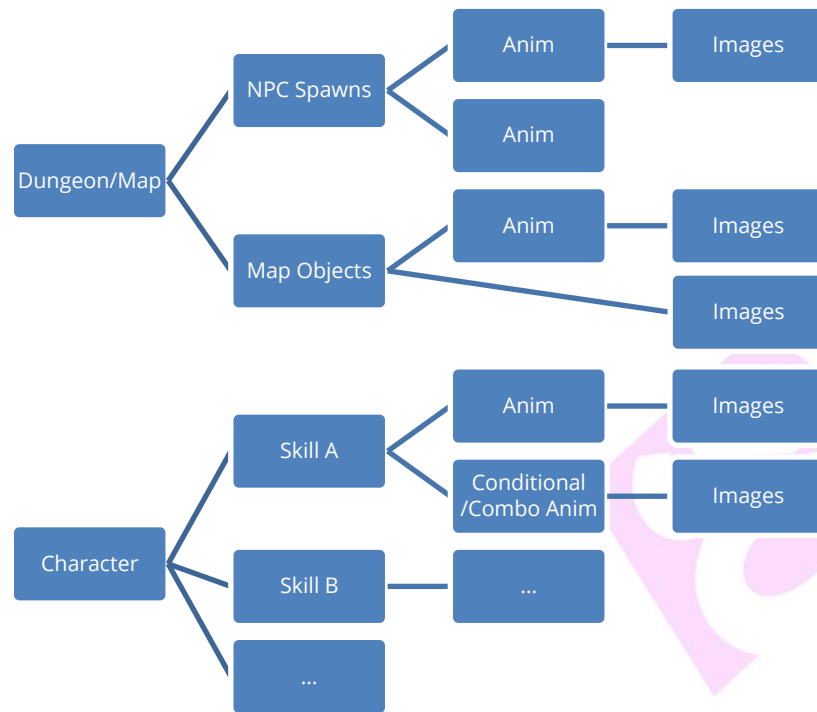
            // Sleep for a short time
            try {
                Thread.sleep(100);
            } catch (InterruptedException e) {
                // Ignore the exception
            }
        }
    }

    private void processMessage() {
        // Process the message
        // ...
    }
}

```

Building Preload List automatically

- Performed Cluster Analysis with variance of time, and associated with likely-related Skill or Map ID.



Building Preload List automatically

- A lot of noise, couldn't cover 100% but very effective
 - Preloading skill resources make players feel far more lag-less than others because it's related to control response time in battle
- Problems / Limitations
 - Lots of noise in skill timing, because of low-
 - Players with low RAM experienced thrashing
 - Added automatically decided option to choose preloading
- Plan: Generic preload manager module with dashboard service
 - Another General Approach for costly and hard problems
 - A Burden for live dev teams to develop full system as a service and maintain
 - Helpful to many live projects having memory-lag issues

Major Stability Issues

- Crashes
- Memory Problems
- **Lag/Slowdown Issues**



Recognize

Prioritize

Find Cause

Fix

Lag/Slowdown issues: Hard to recognize and define problem

- Lag/Slowdown problem is hard to define a issue because it occurs differently in different environment
- Development Team and QA Team had a good machine with SSD
 - couldn't realize about most of loading / lag issues
 - got a test machine with 1GB RAM / XP
- Hard to realize how much it is happening to every/certain kind of players
- Still many XP users
 - especially in other countries like China, Vietnam, ..
- Not easy to test all the contents with every kind of machine for weekly releases

Limitations of Conventional FPS monitoring

- Normally monitors average FPS
 - Collects FPS a single user session / game session
 - Monitors FPS for all or specific contents
- Can monitor overall performance degradation for all or specific contents
- Hard to monitor FPS spikes / Frame Lag

Monitoring Lag/slowdown better

- FPS monitoring to collect FPS distribution per time slot
- Along with various information
 - Map, Character Type
 - Recent Skills
 - System environments (Graphic option / Screen resolutions)
- Can monitor overall performance including laggy experiences
- Still hard to reproduce, find cause and optimize

Limitations in Finding Cause using existing Performance Profiler

- Hard to use conventional profiler in game
 - Client security solution denies whole access from outer process, including Visual Studio
- Many games had developer-mode profiler
 - Many Had to define profile entries
 - or had limitations in breaking down
- Hard to reproduce performance degradation
 - Get an old system to reproduce
 - Have to perform several experiments because of side effects like hard disk cache

Recognize

Prioritize

Find Cause

Fix

JYP – Just Yet-another Profiler (actually not)

- FPS monitoring with just-in-time Sampling profiler
 - embedded sampled Call Stack info with FPS monitoring
- Can find performance hotspot when needed
 - When frame rate drops
 - When a player experiences lag
- Not to be confused with JYP, famous K-POP producer (쥘)

JYP ARTISTS



J.Y.Park



Wonder Girls

Recognize

Prioritize

Find Cause

Fix

JYP – Challenges

- FPO (Frame pointer omission)
 - Can not perform stack-walking properly with PDB
 - Requires PDB to analyze properly
 - Stack-walking by PDB is very slow
- Security
 - PDB must be not accessible from client / publisher
- Performance
 - Must not slow down gameplay

Recognize

Prioritize

Find Cause

Fix

JYP Prototype & PoC

- FPS distribution with various tag filters



Recognize

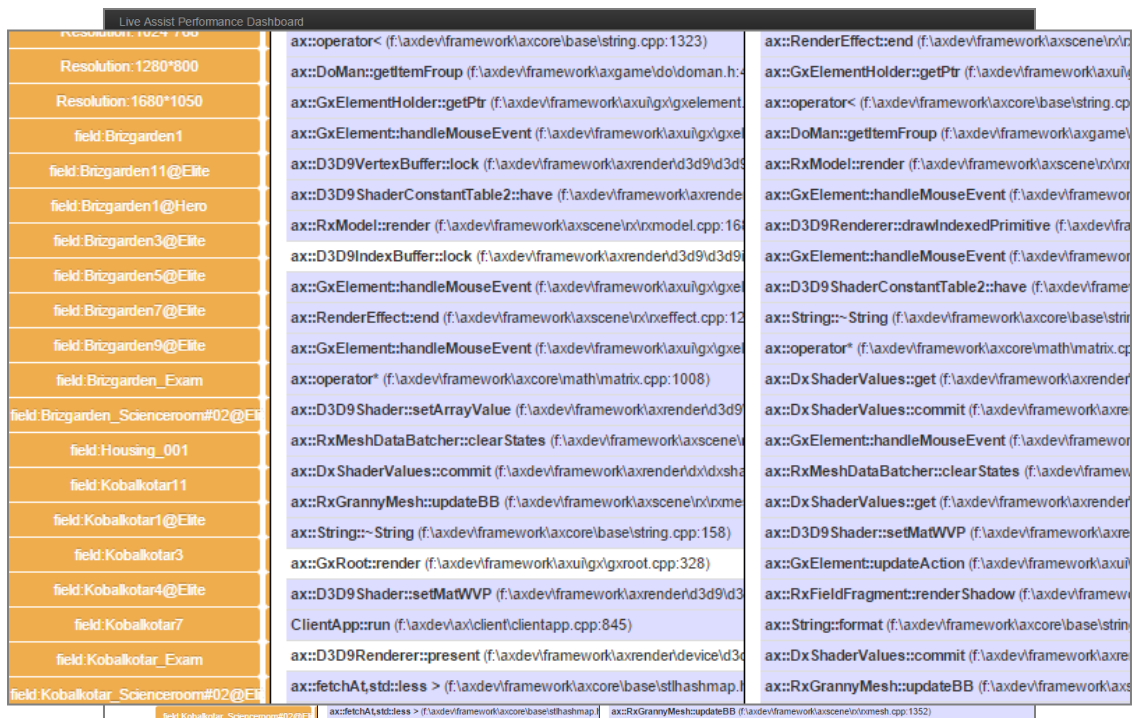
Prioritize

Find Cause

Fix

JYP Prototype & PoC

- FPS distribution with various tag filters
- When clicked histogram, can compare call stack hot spots



Recognize

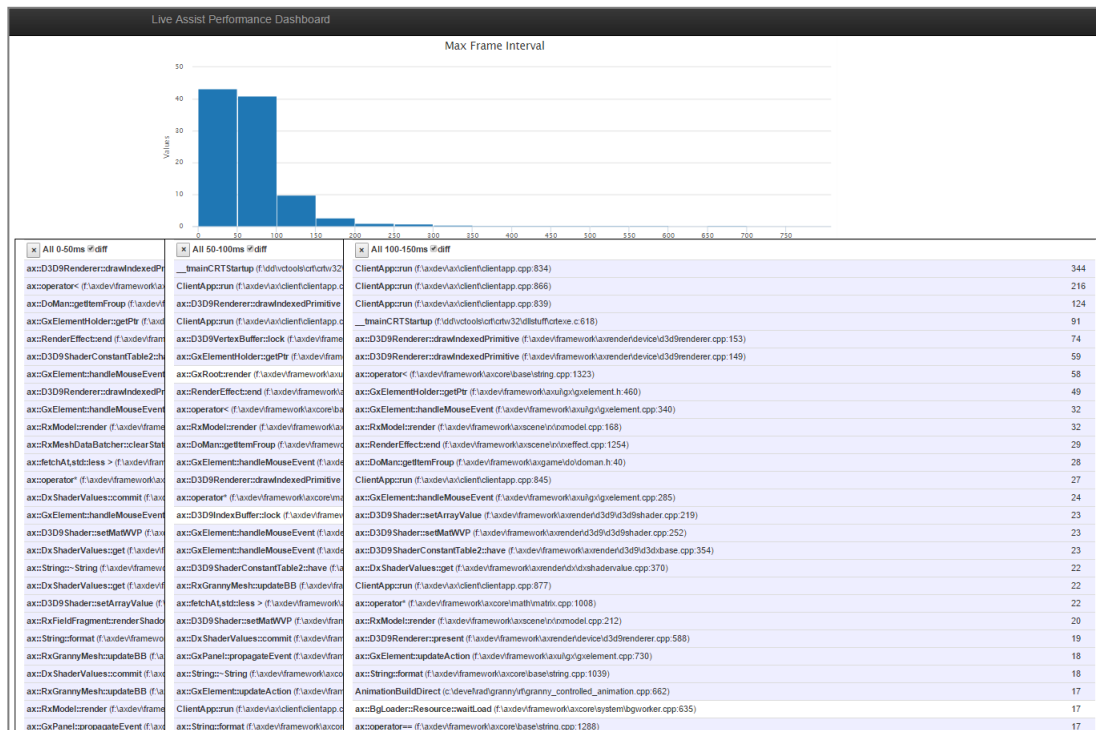
Prioritize

Find Cause

Fix

JYP Prototype & PoC

- Can grouped by max frame interval as well
- Represents laggy experiences well and can break down into call stacks and find causes



Recognize

Prioritize

Find Cause

Fix

JYP Prototype & PoC

- Can grouped by max frame interval as well
- Represents laggy experiences well and can break down into call stacks and find causes

ax::D3D9VertexBuffer::lock (f:\axdev\fram	ax::D3D9VertexBuffer::lock (f:\axdev\fram	ax::D3
ax::GxElementHolder::getPtr (f:\axdev\fram	ax::GxElementHolder::getPtr (f:\axdev\fram	ax::D3
ax::GxRoot::render (f:\axdev\framework\axu	ax::GxRoot::render (f:\axdev\framework\axu	ax::D3
ax::RenderEffect::end (f:\axdev\framework\k	ax::RenderEffect::end (f:\axdev\framework\k	ax::Gx
ax::operator< (f:\axdev\framework\axcore\ba	ax::operator< (f:\axdev\framework\axcore\ba	ax::Gx
ax::RxModel::render (f:\axdev\framework\ax	ax::RxModel::render (f:\axdev\framework\ax	ax::Rx
ax::DoMan::getItemFroup (f:\axdev\framewo	ax::DoMan::getItemFroup (f:\axdev\framewo	ax::Re
ax::GxElement::handleMouseEvent (f:\axde	ax::GxElement::handleMouseEvent (f:\axde	ax::DoMan::getItemFroup (f:\axdev\framework\axgame\do\domain.h:40)
ax::D3D9Renderer::drawIndexedPrimitive	ax::D3D9Renderer::drawIndexedPrimitive	ClientApp::run (f:\axdev\ax\client\clientapp.cpp:845)
ax::operator< (f:\axdev\framework\axcore\ba	ax::operator< (f:\axdev\framework\axcore\ba	ax::GxElement::handleMouseEvent (f:\axdev\framework\axui\gx\gxelement.cpp:285)
ax::D3D9IndexBuffer::lock (f:\axdev\framev	ax::D3D9IndexBuffer::lock (f:\axdev\framev	ax::D3D9 Shader::setArrayValue (f:\axdev\framework\axrender\d3d9\d3d9shader.cpp:219)
ax::GxElement::handleMouseEvent (f:\axde	ax::GxElement::handleMouseEvent (f:\axde	ax::D3D9 Shader::setMatWVP (f:\axdev\framework\axrender\d3d9\d3d9shader.cpp:252)
ax::GxElement::handleMouseEvent (f:\axde	ax::GxElement::handleMouseEvent (f:\axde	ax::D3D9 ShaderConstantTable2::have (f:\axdev\framework\axrender\d3d9\d3d9shader.cpp:354)
ax::D3D9 ShaderConstantTable2::have (f:\a	ax::D3D9 ShaderConstantTable2::have (f:\a	ax::Dx ShaderValues::get (f:\axdev\framework\axrender\d3d9\d3d9shader.cpp:370)
ax::RxGrannyMesh::updateBB (f:\axdev\fra	ax::RxGrannyMesh::updateBB (f:\axdev\fra	ClientApp::run (f:\axdev\ax\client\clientapp.cpp:877)
ax::fetchAt, std::less > (f:\axdev\framework\k	ax::fetchAt, std::less > (f:\axdev\framework\k	ax::operator* (f:\axdev\framework\axcore\math\matrix.cpp:1008)
ax::D3D9 Shader::setMatWVP (f:\axdev\fram	ax::D3D9 Shader::setMatWVP (f:\axdev\fram	ax::RxModel::render (f:\axdev\framework\axscene\n\rxmodel.cpp:212)
ax::Dx ShaderValues::commit (f:\axdev\fram	ax::Dx ShaderValues::commit (f:\axdev\fram	ax::D3D9Renderer::present (f:\axdev\framework\axrender\device\d3d9renderer.cpp:588)
ax::GxPanel::propagateEvent (f:\axdev\fram	ax::GxPanel::propagateEvent (f:\axdev\fram	ax::GxElement::updateAction (f:\axdev\framework\axui\gx\gxelement.cpp:730)
ax::String::~String (f:\axdev\framework\axco	ax::String::~String (f:\axdev\framework\axco	ax::String::format (f:\axdev\framework\axcore\base\string.cpp:1039)
ax::GxElement::updateAction (f:\axdev\fram	ax::GxElement::updateAction (f:\axdev\fram	AnimationBuildDirect (f:\axdev\framework\axscene\animation\animation.cpp:663)
ClientApp::run (f:\axdev\ax\client\clientapp.	ClientApp::run (f:\axdev\ax\client\clientapp.	ax::BgLoader::Resource::waitLoad (f:\axdev\framework\axcore\system\bgworker.cpp:635)
ax::String::format (f:\axdev\framework\axco	ax::String::format (f:\axdev\framework\axco	ax::operator== (f:\axdev\framework\axcore\base\string.cpp:1268)

Call stack hotspot in lag with 50~100ms:

GxRoot::Render (GUI)
IndexBuffer::lock

Call stack hotspot in lag with 100~ms:

BgLoader::Resource::waitLoad

Doing more to prevent risks

- Major Stability Issues
 - Crashes
 - Memory Problems
 - Lag/Slowdown Issues
- **Doing more**



Recognize

Prioritize

Find Cause

Fix

Doing more in Runtime, On-demand

- Often, need to put debug-log to find causes of not-reproducible problems
- Usually takes 1~2 weeks to develop, test, and deploy for average projects running in multiple service regions
- Isn't there a way to track state easily on-demand?

Recognize

Prioritize

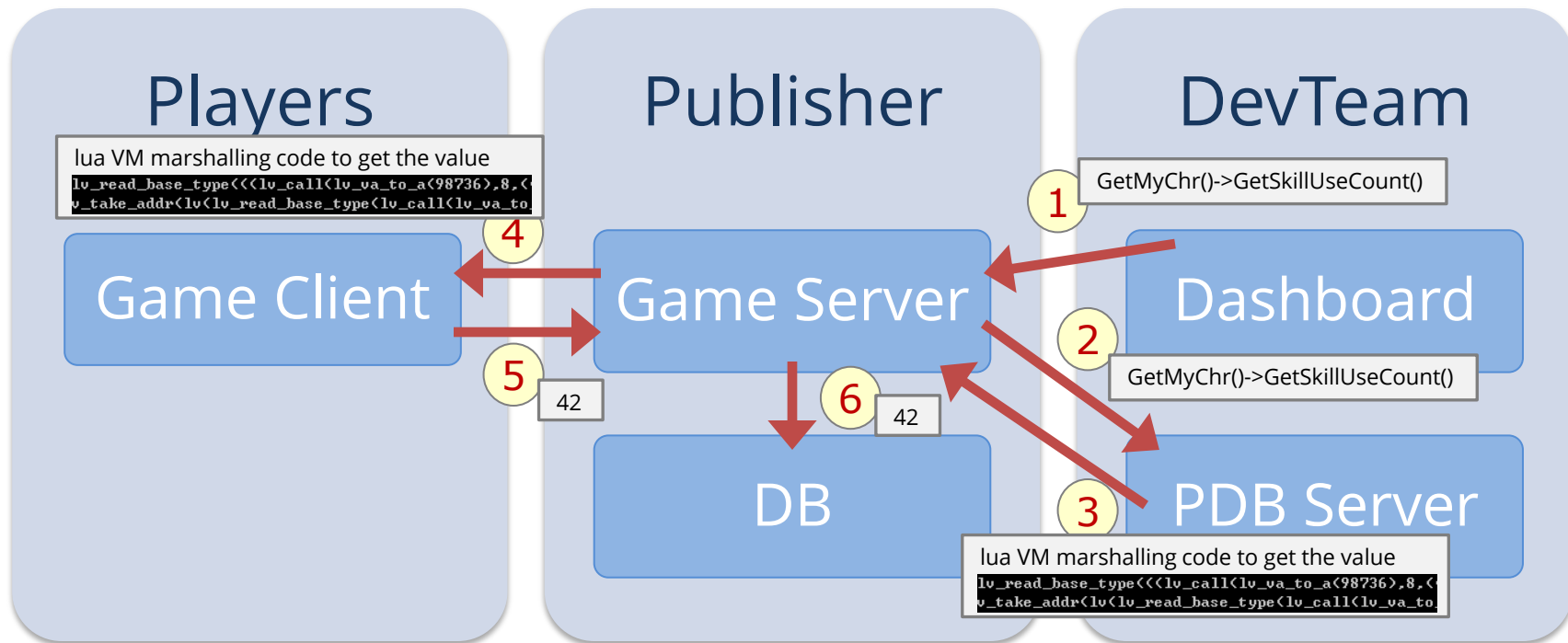
Find Cause

Fix

Live Watch technology

- Telemetry on-demand
 - Not only for debugging
 - But also for Data Analysis, Hacking Prevention, Service Operation
 - Can retrieve variety of information that are not planned, prepared
- Doesn't need to rebuild or redeploy
 - which takes days ~ weeks to release
- Doesn't even need to add telemetry code
 - prevents log bloat

Live Watch technology



Recognize

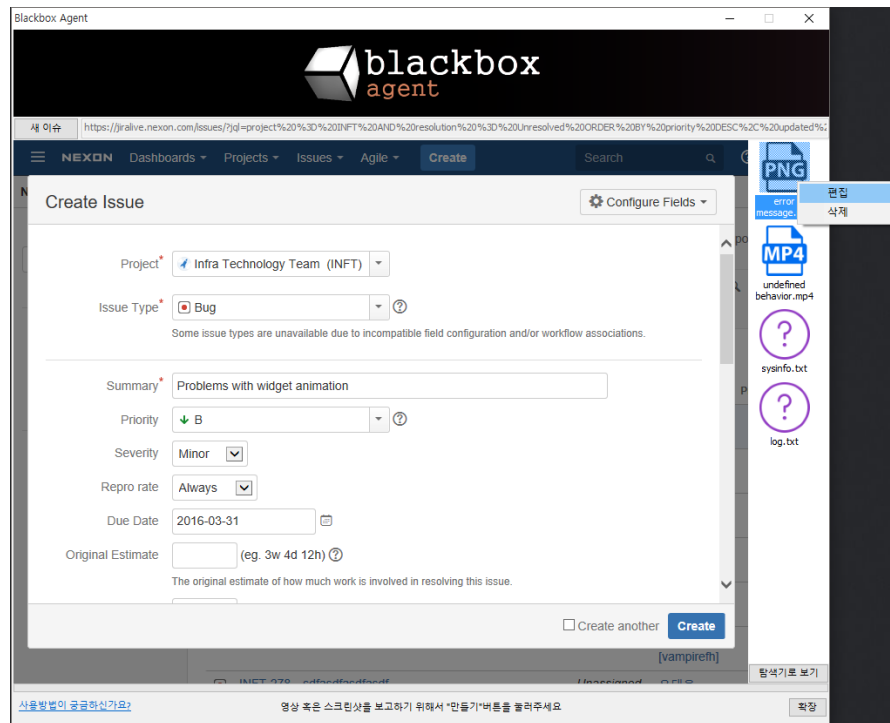
Define

Find Cause

Fix

“Blackbox” – a dashcam for a game

- Records gameplay video constantly
 - only activated for developers / QA team
- Ctrl+F12 directly pops up JIRA create-issue-window with recent gameplay video attached
- Equipped with functionalities
 - auto-login
 - screenshots / edit to annotate
 - enables video preview with re-encoding (JIRA doesn't support, had to download)



Recognize

Prioritize

Find Cause

Fix

"UserVoice" – hearing from user's voice

User Voice - 도메이션즈

작성 날짜 조회

제목 작성 날짜 조회

음.. 도메이션즈가 왜 전화오면 당기는지 알아냄 [6] 03-12 (토) 153
 갤럭시s7/s7엣지 홍보하려고

박격포 포탄에 복엽기가 당겨져나가는 영상 [10] 03-12 (토) 235
 1분 10초쯤에 나옵니다. 그동안 비행선 위주로 쓰다가 복엽기 사용하고 있는데
 오늘 처음 경험했네요. c c 신기했어오 ㅋㅋ

무슨 공격받으면하면 당기나 [1] 03-12 (토) 34
 월드워 공격물어가면당기는데 이거 뭐냐 안고치날?

너무당긴다 [0] 03-12 (토) 0
 너무 많이 당기네요 그것도 전투 시작하고 나서만 맨날 병력 날라가고 완전 짜
 흥나는게임

왜 로딩 다되기도전에 당기나—— [0] 03-12 (토) 0
 빨리 어영거움해봐

도메이션즈가 만들어가지네요 [0] 03-12 (토) 0
 게임아이를 누르면 로딩창에서 당기네요ㅠㅠ

전쟁하러가면 당겨서 기회도 놓치고 정말 심하다 [0] 03-12 (토) 0
 월드워 전쟁 들어가면 너무 많이 당기고 기회도 잃어버리고 정말 심하다 심해

종운대 [0] 03-12 (토) 0

한번 당기는게 어려웠나보군요
 by 오자서(pow****) | 2016-03-10 21:22 | 조회 125 | 추천 0
<http://cafe.naver.com/nexdominations/194422> 존소프산

날들 당길땐 그렇게 괜찮다가 이제서야 한번 당기고나니 상습적으로 당기네요
 백그라운드 업 정리같은건 기본이고 전정기치 미배치된 것도 있는데
 월드워때나 악탈시 종종 극심한 버벅거림의 핵이 생기는데 주로 이때 당긴 현상이 보입니다
 월 더 누르면 당길까봐 버벅 거리면 얼얼불난하는데 오늘 또 깔끔하게 당겨주시니
 말할능의 게임 계속 해야할거 외면이군요

댓글 [11]	날짜 ↑
전 마일올리고그릴... 테블릿은 플리팝인데 안당기고... 마일더움업엇을 안한건 지? 실존하능건지 ㅋㅋ 기본적인버그도 안고치는발로써선 놀고있는거같은데 ㅋ 안드아신가요?	03-10 21:25
아이폰 6입니다.	03-10 21:27
아이폰이 특히 심한가보네요 ㅜㅜ 상상보이는 분들이 대부분 아이폰이네요	03-10 21:27
개인적인 생각이지만 도미 게임들이 메모리 최적화를 발판해서 메모리가 폭주 하는듯한 느낌이 들어요. 그중에서도 메모리 사용량이 제일 많은게 바로 워해리니그리니	03-10 21:28

Recognize


Prioritize

Find Cause

Fix


"UserVoice" – hearing from user's voice

2016-03-11 (금) 오전 8:56

 _유저보이스 [uservice]
[키워드알림] 2016-03-11 (금) 유저보이스 일간 키워드 알림 (도미네이션즈)

받는 사람 田 _DominationsGM [gm-domi]

이 메시지가 표시되는 방식에 문제가 있으면 여기를 클릭하여 웹 브라우저에서 메시지를 확인하십시오.

 **실시간 모니터링/긴급모니터링** - 24시간 동안 22개 글 (구성키워드: 결제, 결제, 공지, 구매, 구입, 환불, 계정, 로그인, 실행, 안돼, 안되, 연동, 초기화, 튕겼, 튕김, 튕기, error, 문제, 버그, 어부져, 어부징, 에러, 오류, 불법, 악용, 일꾼핵, 제보, 치트, 핵)

제목 / 내용	작성 시간
강증버그를 쓰자 (강증 버그를 고쳐달라) 많은 온라인 게임들을 해보면 항상 누군가 의도하든 의도치 않든 버그를발견한다그리고 ... 더 보기	2016-03-11 04:18
평조없이 자리비움하면 1분도 안되서 공격당할수있다 평조없이 자리 비우는순간 1분도 안되서 공격당하는 사태가 있조예시를 보여주조8시간 3... 더 보기	2016-03-11 02:38
팅김 현상 계속 발생 해결 방법좀 부탁합니다. 메모리정리 재부팅 해왔는데 계속 발생되네요..... 더 보기	2016-03-11 00:19
독바로 운영하라 넥슨 자국민들에게는 버그나 오류가 넘치는 게임을 가져다 먹이면서 미국이나 다른 자본에는 ... 더 보기	2016-03-10 23:11
♥♡♥Lazencia save us / 즐거운걸전,부담없는월드워,고전~산업 군주님 환영해요♥♡... 신생길드 > Lazencia Save Us <(고 신해철의 노래 제목이예요 ^^@)♥ 주 2회 적당... 더 보기	2016-03-10 22:22

[나머지 17개 글 더 보기](#)

2016-02-01 (월) 오전 8:56

 _유저보이스_개발 [uservice_dev]
[트렌딩키워드] 2016-01-31 (일) 유저보이스 일간 트렌딩 키워드 (도미네이션즈)

받는 사람 田 _유저보이스_모니터링 [uservice_monitoring]

이 메시지가 표시되는 방식에 문제가 있으면 여기를 클릭하여 웹 브라우저에서 메시지를 확인하십시오.

도미네이션즈 일일 급상승 키워드

#	2016-01-29(금)	2016-01-30(토)	2016-01-31(일)
1	리플레이 NEW: 8위	산업 40위→1위 ▲39	닌자 NEW: 1위
2	인종 NEW: 10위	홀치기 128위→4위 ▲124	공략 NEW: 6위
3	재부팅 NEW: 18위	프랑스 82위→2위 ▲80	고의 NEW: 10위
4	숲 NEW: 16위	계몽 12위→3위 ▲9	사거리 NEW: 14위
5	인구수 NEW: 19위	벡지 191위→6위 ▲175	사보타주 NEW: 21위
6	나이 208위→6위 ▲197	중국 322위→9위 ▲313	기쁨 NEW: 24위
7	영여 NEW: 22위	도서관 NEW: 23위	영상 NEW: 28위
8	무적함대 NEW: 23위	전차 NEW: 25위	1별 NEW: 29위
9	태풍강림 NEW: 24위	궁수 299위→13위 ▲286	그리스 67위→3위 ▲64
10	급식중 NEW: 28위	프랑 NEW: 28위	검치기 NEW: 31위

[더 자세히 보기](#)

도미네이션즈 일일 주요 키워드

#	2016-01-29(금)	2016-01-30(토)	2016-01-31(일)
1	알렉스	산업 ▲39	닌자 NEW
2	알렉스님	프랑스 ▲60	중전차 ▲24
3	월드	계몽 ▲49	그리스 ▲64
4	글 ▲1	홀치기 ▲124	영국 ▲56
5	공카 ▲3	석유 ▲31	훈련소 ▲39

Recognize

Prioritize

Find Cause

Fix

Anomaly Monitoring

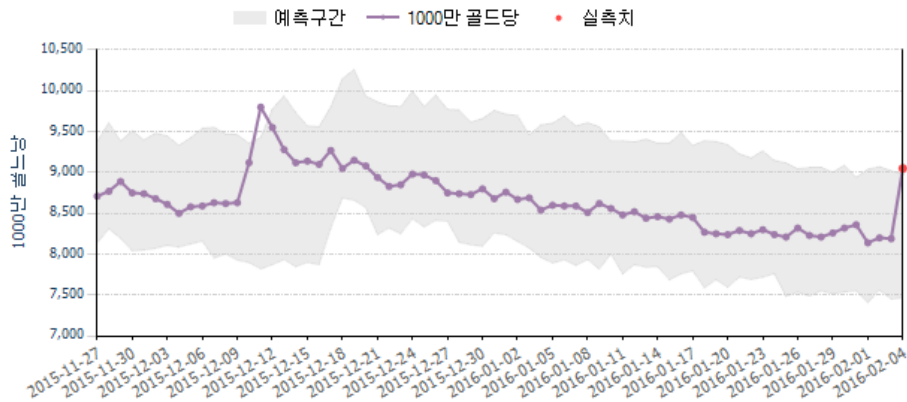
NCS 이상지표

2016년 02월 04일 목요일 일일 이상지표

1. 머뎁의전설 : 문의사항 (전체) - 문의인원수 [웹에서 보기](#)

2016년 02월 04일 목요일 일일 이상지표

1. 던전앤파이터 : 서버 (안톤) - 1000만 골드당 [웹에서 보기](#)



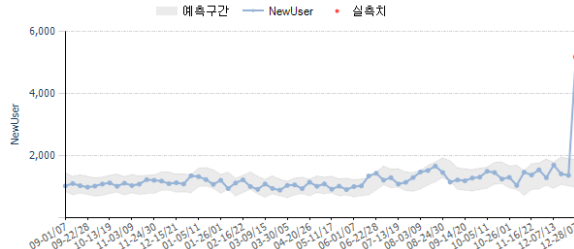
2016-01-07 (목) 오전 10:14

RE: [지표이상탐지리포트] 2015년 12월 28일 - 03일 주간지표

받는 사람 이남미 [mory]; 여민규 [mk3056]

참조 _라이브지표모니터링 [livedata_monitor]; DW팀 [dw]; _데이터분석팀 [da]

23. 아스가르드 : 국가 (한국) - NewUser [웹에서 보기](#)



게임	구분	구분값	이상지표	실측치	예측 구간(H)	예측
아스가르드	국가	한국	NewUser	5,189	1,862	예측

24. 던전앤파이터 : 성별 (남자) - UU [웹에서 보기](#)

※ 해당 지표는 동일커데고리 내 다른 항목과 추세 차이가 있습니다.
데이터 해석 시 [웹에서 보기](#) 링크를 통해 다른 항목과 비교하시기 바랍니다.

예측 구간(H)

30

예측 구간(L)

-11

Recognize

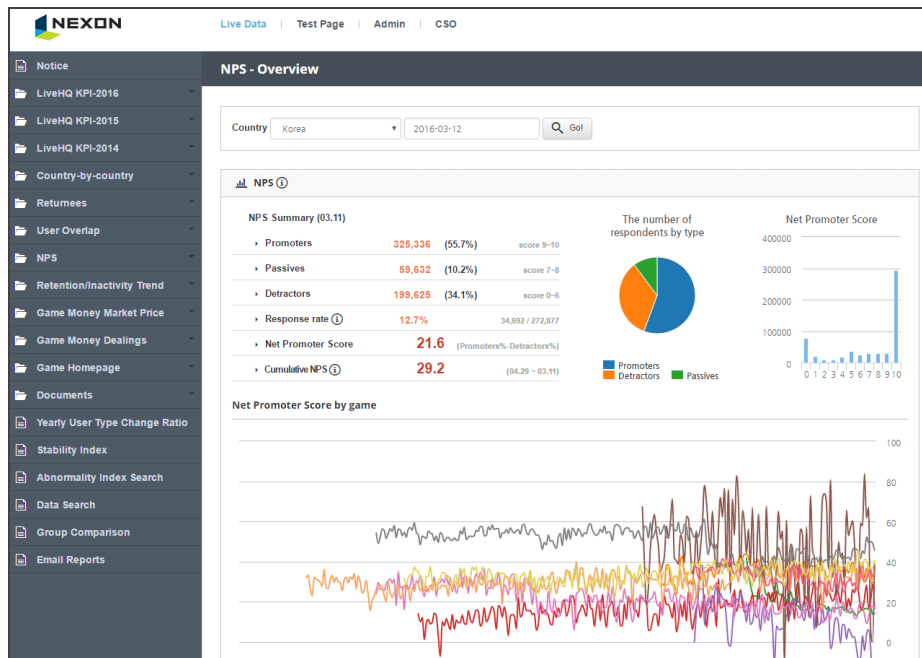
Prioritize

Find Cause

Fix

Live Data Portal – Data Analysis On-demand

- Variety of reports from live data as a service including:
 - Anomaly Reports
 - User Overlap Reports
 - Group Comparison



Bottomline

- In live services, mitigating and preventing risks are very important
 - Prevent service not to stay unstable
 - Making problem **visible** is important
In many cases, problems are not easily visible
 - If you make a solution or tool,
Make a solution **easy-to-use, on-demand** for everyone, and
Make a solution more **reusable** as you find duplicated costs
 - e.g. Crash Analysis, Data Analysis anytime for anyone
 - Minimizing time spent to recognize prevent risk is crucial
Whatever it takes time,
we can try to break it down to reduce time spent



Thank You

innover@nexon.co.kr

https://twitter.com/_CKSong