

BENJAMIN **FORREST**

Case Study Booklet • 2011



Excellence in Project Management

Hi. I'm the small print. Most people don't read me, but I can tell that you're a stickler for details.

I like that. We should get along marvelously.

Normally this page is left intentionally blank, but I thought, "What the heck; why not write something here to make this booklet look really official!?" So here we are.

Document created December 2011 using Adobe InDesign, Photoshop, and a bunch of other tools too numerous to name. Hey why are you still reading this? The good stuff starts on the next page!

It's so nice to see you. Come on in and let me share with you a few of my thoughts about some of the more interesting projects I've managed over the last ten years or so ■■



4-7

Sure, everyone thinks that testing video games might be easy - but in reality it's one of the most challenging and taxing mental tasks there is - With this in mind, how do you go about training someone to do it?



8-11

Taking DevSuite on a world wide tour sounds like fun (which it was), but it was also a huge endeavour that required lots of planning.



12

Contact info •



Case Study

Game**Test**Analyst/

Building a dynamic & engaging technical training program from the ground up.

Who do you call when you want to realign procedures, methodology, terminology, and a bit of organizational culture? Oh, and you want to do it quickly and at low cost. Sounds like a tall order, right? Impossible?

Here's how I pulled it off.

Project Background

It was late 2003 and at that time First Party QA (FPQA) in the US was split across two different campuses and though there were training materials, there was no standardization across groups and many procedures were done differently depending on which team performed them. That's where my team and I came in. After meet-

I learned a lot about managing projects on this one - some of it the hard way - in the end though I'm still pretty proud of having managed the creation of something still in use today, nearly 10 years later.



ing with stakeholders at both campuses, I recommended an overhaul of current training materials and drew up a project proposal that would be broken down into phases over a period of months with the help of resources from several teams

across both campuses.

The Process

It bears being said that this was not a wholly new idea - management had hired training personnel in the past to attempt a solution, but without success. Some of what went well came about as the result of following my gut, but I also had the advantage of knowing what *wouldn't* work.

Previous attempts didn't involve much input from the teams involved in this significant change and the previous consultant had burned bridges with the managers and line supervisors over a period of months.



A slide from the GTA presentation. Not your average presentation; then again, it's not your average audience, either. ▲

Alternate GTA Participant manual cover, made to look like a PS3 game cover to carry course gaming theme through the materials. One of several iterations we've used over the years. ➡

"[Ben] brings that rare combination of technical proficiency, clear communication skills & strategic insight. I have repeatedly seen him influence his executive management team, negotiate complex budgets, champion new programs and advocate for his direct reports. He is the kind of technical leader organizations dream about."

— Mathew Landes, on working with me during the GTA project.



GAME TEST ANALYST

PLAYSTATION 3



GAME TEST ANALYST



ILLUMINATING THE DARKNESS
OF UNTESTED CODE



Game**Test**Analyst/Continued...



...it was the planning, execution, monitoring, communication, and transition from project close to repeatable process that won the day.

The key to our eventual success with this project wasn't just the idea - it was the planning, execution, monitoring, communication, and transition from project close to repeatable process that won the day. This being one of my first big projects - certainly the first involving multiple stakeholders, development resources other than myself, and all of it being managed across locations - I count myself a little lucky that things went as well as they did!

The project was really three projects rolled into one and each came with its own set of challenges. The first thing that needed to be done was to get the key players to sit down and agree to make some decisions. This consisted of a series of regularly scheduled meetings between the four key managers of the two test campuses wherein I moderated and helped the management team negotiate a list of identified deltas between procedures - each decision was directly linked to an instruction development task in the next phase.

I'm pretty sure that the only reason the management team agreed to meet with me was my promise to do three things: keep the meeting "tasks" short (we typically worked in 15-30 minute sessions), communicate our progress regularly, and keep after them to meet with me otherwise, like that spare fridge in your garage that somehow gets filled up even though you were able to fit everything in when you only had a single fridge, they'd find a way to need more time.

Keep in mind that this wasn't just a matter of picking a winner between existing documentation or procedures - in some cases this process also identified major areas for improvement in test processes that neither campus had recognized previously and everyone agreed that the materials needed to be updated and "made pretty".

From here, I was provided additional resources, both subject matter experts to consult with from the teams, but also

development resources in the form of a contract instructional designer and authorization to use some of my trainers as writers/developers on the project. Needless to say, I learned to rely on calendars and lists a lot until I finally learned how to bend MS Project to my will later in the project.

Lessons Learned

Like any project, there are going to be challenges and hopefully lessons to be learned from those challenges. My first lesson was:

Don't be afraid to tell someone things are taking longer than we thought.

Like that Band-Aid you're afraid to remove as a kid, this lesson only really sets in when things get behind (as any project may do) and you fail to let everyone know early on. I had seen this first hand in projects run by others, but it was only after some PM mentorship that I learned to look for areas of risk early and be on the

watch for any signs that a task or deadline might slip before it actually does. A little risk mitigation can really save a lot of fan cleaning later on.

I also learned very quickly that letting the person who is actually doing the task should be the one estimating how long said task will take. I got into trouble early on here, especially when suggesting time lines to junior developers on my team who weren't sure how long a given task would take. Just because I can do x task in y days doesn't mean they will, especially when they're just starting out. I also learned some valuable lessons about contract resources and ensuring that their queue stays full of stuff that makes the most sense - use your expensive contractors to tackle tasks that junior developers would struggle with, especially if you're running out of slack in the schedule. It's cool to build in opportunities for junior devs to develop and grow, but if you need to make up time call in the heavies (and make sure you plan ahead for that contin-

gency - don't blow the consulting \$\$\$ up front!)

Once the initial decisions had been made and everyone agreed on our direction, the real work began. Having the development tasks hammered out early really helped, but anyone who has attempted a change management initiative like this knows that getting management's buy in was only half of the work.



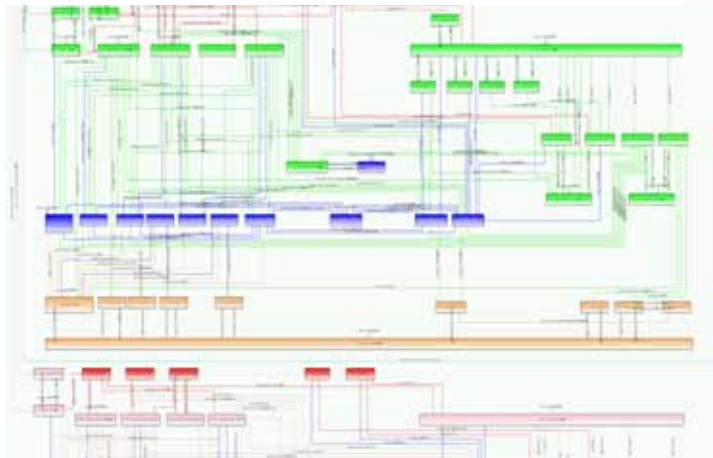
I quickly began to understand that communicating changes out to the test teams themselves and getting their input and feedback was the only way to make this project successful. Of course back then, I had no idea that what I was doing was called change management - I just knew that I didn't want to see this effort still born like others before it and analyzed previous failed attempts in an effort to avoid those mistakes.

GTA Today

The Game Test Analyst program is still being delivered to new Global First Party Quality Assurance personnel today; however, it should be said that the current form is the result of an annual reboot that follows a process very similar to the path forged by the project originally undertaken in 2003-2004. Over the years, it has continued to evolve and change to maintain relevancy and stay in pace with the demands of the teams who rely on the student graduated from the program.●

Implementing **DevSuite** /

or “How We Stopped Worrying
& Learned to Love the Workflow”



▲ What the?? I don't even? How is it possible to both love & understand such a thing as this?

Ok, I'll tell you: lots of work. Love hurts, man.

○ ne database to rule them all...
Sounds pretty good, right? A

single application that would unite all
of World Wide Studios seemed like a tall
order, but we were just fearless (okay,
maybe foolish) enough to take on such
a challenge.

Ever try to herd cats or executives towards
a single goal? No?

Project Background

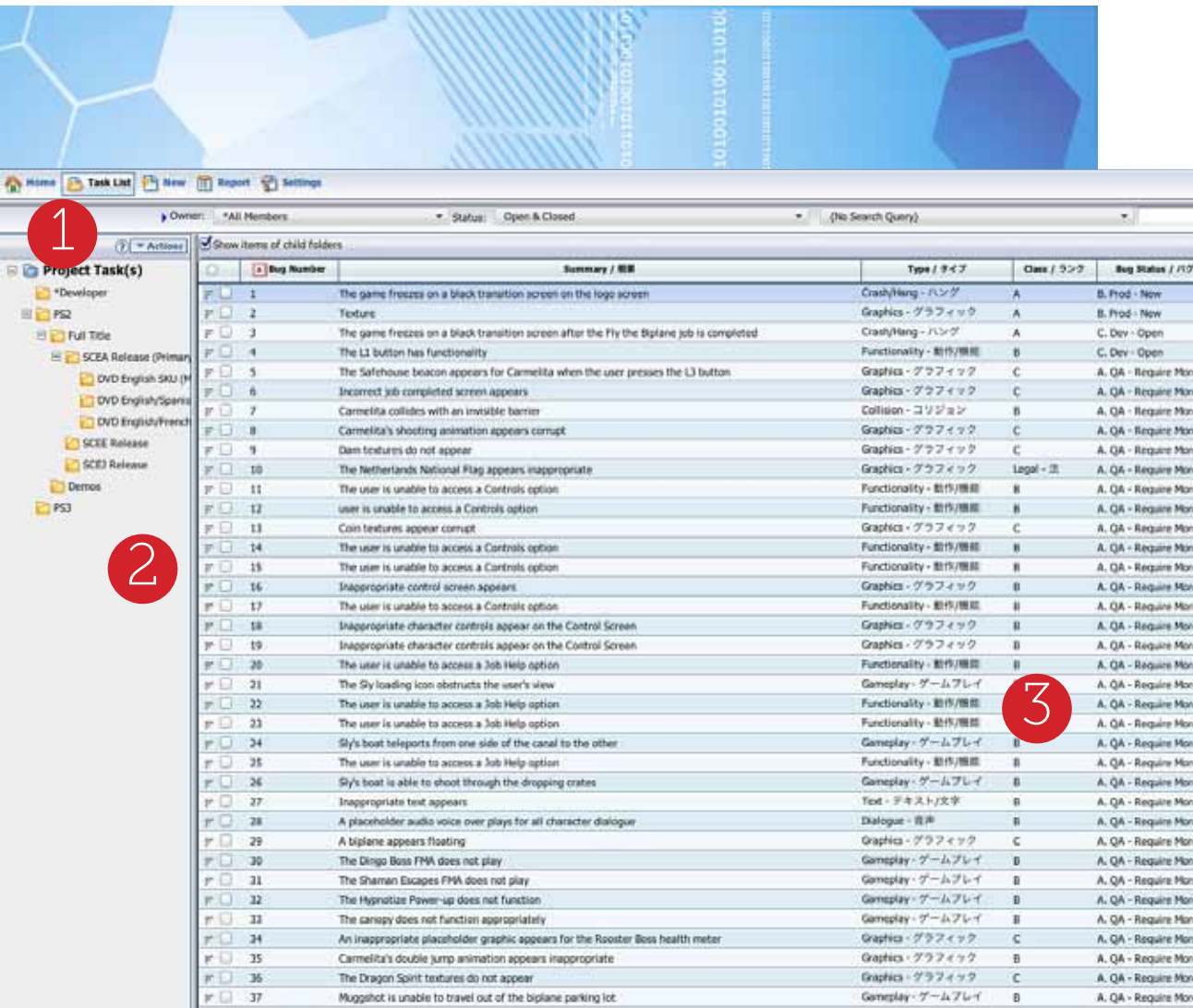
In order to make this project work, not only did we have to do the proper planning, research, and preparation (of which there was plenty); we had to log some serious mileage. I'm not talking a few trips here and there - I'm talking about visiting every major studio in World Wide Studios to get their buy-in.

We had to figure out a way to get nearly two dozen studio executives from different corners of the globe to agree on what would be, at that time, one of the largest software purchases in Sony Computer Entertainment history.

Not only that, we were given only 18 months to do it in.

The Process

This project was so incredibly remarkable to me for several reasons, not the least of which was how incredibly smooth it went, given the stakes and circumstances. I can only attribute this to enormous amount of planning and thought that went into each phase of the project.



1 First Rule of DevSuite
Is You Don't Get Lost Using DevSuite. The Second Rule of DevSuite is, well you get the picture. Our first responsibility to users was to ensure that whatever database we chose had a human experience that was easy to pick up and reasonably intuitive. Searching, filtering, and custom queries therefore are key and it's something that DevSuite did really quite well.

2 Dude, where's my...?
Did you happen to notice how much this looks like another familiar application that you use all the time? Yeah we did too. Comparing our scary new database app to a friendly e-mail app made the transition easier on everyone. It also helped people grasp the essential notion that bugs *move* through the system, like e-mail between users. The only big difference was in DevSuite you couldn't just delete an e-mail you didn't like.

3 Big in 日本!
Customization and language support are king when it comes to asking for support from a global organization like WWS. Polyglot apps FTW!

↑ DevTrack may not love you back, but that's ok. In fact, I'm pretty sure it doesn't love you. It told me it's incapable of love. "It's not you, it's me" is what it said I seem to recall. But that's cool, right? Not everyone has to love you. I mean, I'm sure they're plenty of people who don't love you out there. Don't let it get you down.

What's that you were saying? This caption text is creepy? Oh, right, we were talking about DevTrack. Yeah. so this is a screenshot of the interface...

Implementing **DevSuite** / Continued...



Unfortunately, I can't take credit for Project Managing the entire effort - that honor was well earned by Jim Wallace, who oversaw the project as a whole and who was assisted by Matt Harper, Tim Vanlaw, and many others, from whom I learned quite a bit during the span of this project; however, I am quite proud of both my involvement in phases run by other PMs, as well as the phase I managed personally.

As with other complex initiatives in the past, this one was split into roughly three phases each run by different teams and overseen by Jim: RFP & Evaluation, Configuration, and Training & Implementation. Though I was involved in each phase in

some capacity, it was the final Training & Implementation phase that I was charged personally with managing. I oversaw the design, development, and scheduling of training for each and every user in WWS - from sending trainers onsite to studios, to having them deliver training via webinar and pre-recorded screencasts. Additionally, though it wasn't part of my formal mandate, I took on the added responsibility of ensuring that my team engaged in constant change management activities to ensure the project's eventual success; this meant "leaking" juicy tidbits about the new system to users on the test floor and creating marketing materials preparing users for the changes to come.

Lessons Learned

This might go without saying but, no matter how much you like the vendor, at the end of the day those guys are only as good as their reputation. Quite a bit of risk management was done up front in this project, but one thing I learned during and after the initial purchase: if promised something magical and/or shiny and new - get it in writing. A vendor may tell you that the new dual language feature is going to be ready for next month's release and have the best of intentions, it doesn't mean he can really deliver. Even if you get it in writing, it's probably not a great idea to promise that new feature to your clients if you're not 100% certain said feature will be in your hands on time.

Failure of a vendor to deliver something on time wasn't ever a good shield against not being able to execute one's own tasks.

Users down the line aren't going to be sympathetic to a team's inability to provide a deliverable on time due to a slipped deadline from a vendor. In other words, don't offer what you don't yet possess. It's much more effective to be honest and provide a service that is reliable and available than one that holds promise but remains vaporware.

Lastly, when presenting to hundreds of employees, make sure you not only double and triple check before you make that presentation, go to ludicrous check if you have to - you only get one shot to make a good first impression. I saved myself tons of heartache and embarrassment by simply double checking on things prior to go time. •



HEY! THIS **IS** THE DROID YOU'RE LOOKING FOR!

feel free to contact me via phone, interwebs, or
carrier pigeon:

Benjamin Forrest P: 1 (619) 379-4175
5151 Palm Street E: ben@thepixelpub.com
San Diego, CA
92105

<http://www.thepixelpub.com>

