Precision Castparts Corp. Report

Sam Quinn

MECOP Internship #2

Summer/Fall 2015

Intern:
Sam Quinn
Computer Science

Mentors:
Steve Jessie
sjessie@pccstructurals.com
(503)777-8811
Dave Rajkumar
drajkumar@pccstructurals.com
(503)777-8816

November 24, 2015

Contents

1 Introduction

For my second MECOP internship I was fortunate to be placed at Precision Castpart Corp. This was my second time being immersed into a professional atmosphere in the field that I am studying and I enjoyed it equally as the first. I am computer science student with a focus in security and even thought that Precision Castparts Corp. was more of an old school non-tech company, I was greatly mistaken. Precision Castparts Corp. had a lot of advanced technology and very intricate security, which was great for me to get hands on with. It was good for me to learn the basics of how a large corporation like Precision Castpart Corp. handles security. I now feel more well rounded and have at least some experience relating to my security focus.

2 Precision Castparts Corp.

2.1 What does Precision Castpart Corp. do?

When I was first placed at Precision Castpart Corp. I had not heard much about them. I definitely was not a ware that they were one of Oregon's two fortune 500 companies. Precision Castparts Corporation specializes in the casting of metal aerospace parts. They are the world leader in investment casting, forged components, and airfoil castings. The company was started over 60 years ago in Portland, OR by Joseph Cox, the owner of Oregon Saw Chain. The initial intention was to cast a special cutter for his chain saw products. The company began focusing their time on developing the casting operations and looking for more outside work. This portion of the company became so successful that it branched out and became a separate business called Precision Castparts Co. The company began pushing to create larger castings and, by 1962, was able to pour a casting weighing 1,000 pounds. PCC soared above all of the competition, and later signed contracts with General Electric, Boeing, Pratt & Whitney, and several other aerospace companies. The company began developing their processes with titanium, nickel, and stainless steel alloys. PCC soon became the world leader of manufacturing large titanium castings as well. The company currently has several campuses throughout Portland, and has recently been acquired by Warren Buffett's Berkshire Hathaway.

2.2 MIS

During my internship at Precision Castparts Corp I was exposed to many different departments. The casting process has not changed much over the 60 years Precision Castparts Corp has been around, but what has changed a lot has been the technology that they have been using to refine the casting process and this is where the MIS department comes in. All of the technology is managed by the MIS (Management Information System) department. For a large portion of my internship I was stationed at the helpdesk. I liked the atmosphere of the helpdesk and the opportunity to get hands on experience with technology all around the plant. I would help individual users with minor problems like email to large issues that were halting production like robotic arms.

2.3 Mentor/Supervisors

My mentors were the managers of the MIS department Steve Jessie and Dave Rajkumar. Both Steve and Dave were responsible for keeping all of Precision Castparts Corp technology up to date and secure.

There management encompassed the LAN Admins, Programmers, Helpdesk, and the Telecom employees. Even though I was often at my desk in the helpdesk which was separate from the MIS department we kept a close relationship with the other programmers, LAN admins, and techs. I worked with a lot of individuals during my time at Precision Castpart Corp. Often times when a problem would arise we would all have a role in getting the issue resolved.

3 Projects

3.1 Tinstall

Tinstall was one of the first projects I worked on. Tinstall's purpose is to create a more uniform, organized, and easy to maintain directory for internally created tools and utilities. Tinstall reads from a configuration file to get instructions on which files to install and where. Each file must be CVS revision controlled and up to date or else Tinstall will error out. Tinstall must verify that the user has permissions to install to the requested directory. Tinstall will create necessary folders and only overwrite a file if it has been updated. If Tinstall determines that everything checks out, it will install all of the files to the correct directories and create desired symbolic links. In the case of an installation failing, Tinstall will automatically restore the contents of the install directory to the state before Tinstall attempted the install. After a successful install, Tinstall creates a "bill of materials" file and updates the log with the user who executed the installed.

This was the first time I have been introduced to the programing language Tcl. Since Eric Martinson had begun to create this tool before me, there were some files in place that I studied to become familiar with Tcl. It was really nice to have some skeleton code in place before I started, and to view how a professional laid out the base of the program. Mimicking the coding style helped me create a well designed tool. I ended up liking Tickle a lot, even though many people seem to dislike it.

One of the biggest takeaways from this project was the importance of error checking. In my past projects that I completed for school and personal use, I often never implemented error logs or error checking. Studying the code before beginning my work on this project, I noticed that there were comments and error checks everywhere. Since I mimicked Eric's coding style, I too used an abundance of comments and checks. I soon came to the realization of how much easier it is to find and take care of bugs when you know where they come from.

Things Learned:

- Tcl
- Unix Permissions
- Semaphores
- Following design guidelines
- Fail-Safe programing techniques

3.2 DFM FILL WRAP QA

The majority of my time spent at Mentor Graphics was helping Sam with testing and debugging the new DFM Fill Wrap feature. There was an existing DFM Fill Wrap in place when I started but with how

much smaller the IC layouts are getting it was causing problems. Developers decided to make a new Wrap function that can handle smaller layouts like 10nm designs. It was very interesting to be apart of this new product and to understand the need for it.

Eugene explained to me that the reason the old Wrap Fill will no longer work is that designs now require precision cuts that are smaller than the light wave used for optical cutting. This problem can be solved passing the light through a viscous chemical that concentrates the light into multiple parallel straight lines. This is the root of the problem with generating fill shapes. The old Fill Wrap would place dummy fill shapes in a bi-directional manner, since now the cuts are only created in parallel straight lines there cannot be bi-directional fill shapes only uni-directional.

Sam was really good about including me in on all of the Wrap related emails and always invited me to the meetings. I created many test cases and felt like I was a real addition to the team. Also the fact that we are now creating microchips that need precision smaller than the width of a light wave fascinates me, and I am very fortunate to have worked on a bleeding edge product.

Things Learned:

- How IC are cut using light and chemicals
- CVS Repositories
- Creating test cases
- Working with a QA team
- How to use Calibre software

3.3 PERC QA

With about a month left in my internship, I was asked to help with PERC QA. PERC is a part of the LVS tool, which compares a finished layout with the schematic to ensure that the physical implementation of the circuit matches its logical definition thus the name Layout Vs. Schematic. Specifically PERC is Programmable Electronic Rule Checking that verifies layouts with advanced electrostatic discharge checks.

I was called on board to help out with PERC to help gather information about around 400 tests that fail when ran in forced parallel mode. The PERC test suite was built much differently than that of the DFM tests I was used to. The PERC test utilized an intricate and intelligent harness that would analyze each test case and figure out how to run it automatically. It was fairly difficult to get caught up on the PERC QA logic because of how different it was from DFM. I will be creating a dictionary of all the PERC built-ins that will be correlated to existing tests that are failing. Special analysis and repair efforts can be stream lined to built-ins with low coverage or failing patterns.

Things Learned:

- More Tcl
- Using a harness to run test cases
- How to creating a dictionary for debugging purposes
- How to analyze a core file after a core dump

4 Minor Projects

4.1 TWiki Upgrade

During my down time while working with Sam debugging the Fill Wrap feature I had the opportunity to gain knowledge in website hosting. I was asked to investigate what needed to be done to upgrade DFM QA's internal Wiki site. The DFM QA Wiki site was created with TWiki and open source wiki interface and hosted locally at Mentor Graphics. Before diving head first into this project, I had no prior experience working with Apache, CGI scripts, or really any web tools for that matter, but I was very excited to learn.

I spent about a week creating an exact replica of the DFM QA TWiki site on my local machine with the updated version. I ran into a lot of problems with permissions and how Apache can own files that even root can't access. With help from Sam and ICV BIG, I eventually got a working copy running on my computer. Since the TWiki was accessed by all of the DFM QA team, there was too much information for me to verify everything copied correctly during the upgrade. The way Sam and I tested everything was correct, was we did a live switch and allowed all of the team to poke around the new version to check that all of their paths and normal day links continued to work. After we got our good results back from the team, we copied my local copy to the permanent server location.

Things Learned:

- Apache configuration
- Unix Permissions
- CGI scripts
- General web hosting

4.2 Test Suite Search Tool

After working with Dave in my Tinstall project, Dave approached me with a problem that he thought I could help with. Dave's problem was that when he finds a bug within the test suite he often times did a series of greps and pipes to narrow down his search. Since there wasn't really any built-in UNIX or inside tool that could search for multiple terms, I decided to help him with this.

Dave gave me the freedom to use whatever language I thought best, since I was writing this program from scratch. After enjoying learning TCL so much in the Tinstall program, I decided to learn a new language and went with Perl for the code and Tk for the GUI. Dave thought Perl and Tk were a good choice and after explaining to him that I have never programmed in Perl before he showed me our company library, which was a great resource in later projects.

Things Learned:

- Tk
- Perl
- Making a GUI
- Grep to a higher degree

5 Conclusion

5.1 What I learned

During my 6 month internship I learned a comparable amount of degree related knowledge than I would have at school. The main difference was that the things I learned while Working at Mentor Graphics no class could teach. One of my most exciting things I learned during my internship is how to be efficient. I consider myself 100% more comfortable in a UNIX environment. I believe that I will be a much better programmer and will be able to produce a higher quality of work from the small tips and tricks that were taught to me during my time at Mentor Graphics. The real world work environment seems far less scary to me now that I have had a slight taste of it. It was very neat to see how my background knowledge from school got integrated into the work place. I would often times recall information from previous classes that would help me solve problems I faced during my internship.

5.2 How the internship benefited me

This internship has greatly benefited me in many ways. One of the greatest benefits I will takeaway from this internship is knowing that I can work in the industry. It was great to see that I was working on real projects that full time employees were working on and helped solve unsolved problems. I liked working on nonstandard problems in contrast to how at school I felt like I was finishing a homework assignment that I know every other CS student has solved before me.

5.3 How I benefited Mentor Graphics

I really didn't know how important my contributions were to Mentor Graphics until I was talking to one of the employees about what I was working on with the new Wrap function and he said "You are working on probably the most important update in the upcoming release". After I heard this I truly felt like Mentor Graphics valued my work and that I was actually making a difference. I created more than 30 test cases, upgraded a site most people on the QA team access daily, and helped find which functions that need to be tested.

5.4 Presentations

After finishing Mentor Graphic's internal tool, Tinstall, I was asked to present to a majority of the QA employees explaining my tool. This presentation took place via telecommunication with people viewing from Mentor Graphics locations all over the world.