2 Jun, 2011

Robert Trzeciak

Program Manager

Northrop Grumman ICS

2829 Guardian Lane

Virginia Beach, VA 23452

757-631-2334 work

**Progress Report – 1 May 2011 – 31 May 2011**

Contract Number: HSHQDC-06-D-00022

Order Number: HSCG23-07-J-TED150

Task Order – Performance Work Statement (PWS) 1.12

Attachments: (1) SAROPS subcontractor financial reports

1. **SAROPS Systems Component Services Tasks**
   1. **Activities completed:**
      1. **Wrote up April's Monthly and attended TWGs**
      2. **Created a new drop on May 2nd. Initially, the 64-bit drop wasn’t asked for, but the team started testing it soon enough and now it is included as part of the standard install. I messed up some “GUIDs” when I did this, but Jim and company seemed to recover. Sorry about that.**
      3. **Answered some Great Circle Calculation questions from Jim. Gave him GC intersection code that he paraphrased and found useful.**
      4. **Found and seemed to have knocked out the Computational Geometry problem that surfaced when a point hit the area of interest exactly. This took a couple of days to run down and I rewrote two major routines to better handle the strange case.**
      5. **Chased down an apparent bug in LOB scenarios. It appeared as if some of my particles were going outside of the 3-sd range. The cause of this was simply that the flat-earth rendering by SAROPS is inaccurate. I wrote code to enhance my gui, which has far less distortion than SAROPS'. The 3-sd bearing boundary now shows up and particles are clearly inside of it. In other words, there was no error, but this required verification.**
      6. **Chased down an apparent bug in the track spacing output. We have a requirement for at least 2 search legs that I had been ignoring. The initial placement algorithm (used during the "GetInitialOnly" phase) as well as the iterative solves had to be worked on quite a bit. Now I forbid a move if it reduces the number of search legs to 1. Fortunately, the moves are defined as what they do to the patterns (not to the box) so this was only about a day's worth of work. I did it by defining a pattern that has fewer than 2 search legs as having a track spacing violation equal to the entire path length. That strongly encourages the algorithm to relieve itself of that penalty. Still, this is tricky since increasing the number of legs (from 1 to 2) *shortens* the search leg length, and the normal goal is to allocate more of the path to the search legs. I wrote this up in a long email to Jim. Now we increase the number of legs if need be and then forbid it from going back to 1. Preliminary testing indicates that it works.**
      7. **Discussed this with Jack since the constraint about 1 leg doesn't seem necessary. The team is now aware of the issue and we'll probably allow a single search leg in some form or another in the near future.**
      8. **Logging reared its ugly head again. In a case that was submitted to me for other reasons, there was an SRU that had 0 POS. I couldn't figure out why; the pos table in my gui had reasonable pos values for every other SRU. So I isolated this one SRU. The problem turned out to be that it ended before the particles ceased to be of type "originating craft" and it had no lateral range curve for originating craft. But the problem was that this made that case extremely fast to compute and the logging mechanism, under those circumstances, was creating an enormous log file. I cut down on the logging.**
      9. **Put together a new drop on May 9th including 64-bit, but funding did not permit me to test it. I used some initial swags for the 64-bit parameters. No problems in either distribution were reported until May 15th. Initially, it was thought that I had changed the planner xml, but when I investigated that possibility, it turned out to be false. Then I noticed that the case had bad environmental ranges, but I suspected that to be a red herring and plowed on. With Jim's help, I found the source of the error in my code and cleaned it up. I had changed the result of the GetStatus to make the 64-bit distribution more useful, but that turned out to be incompatible with Beta19. No spec for GetStatus existed and I was unaware of how it was being used. Now that I do know, I put a flag in that, for the re-release of Beta19, is set to "useOldGetStatus." After discussing with Judy and Jim, I used the new GetStatus for Beta20.**
      10. **Documented the parameter settings for the 64-bit SimWebServer; number of simultaneous cases that can be run, etc..**
      11. **Found a case (Forward/Bravo) where a crash (from bad input data) caused SimWebServer to not exit smoothly. Caught this bad data and logged it, exiting this case, and then worked to have SWS exit smoothly on crashes. I'm trying to make SWS more robust even with bad data. With multiple cases running, it's important that a single bad case does not cause the SimWebServer problems; it should recognize the bad case, announce it and then go right on to processing the next case.**
      12. **First try at keeping the length at most 3 times the width for a cs case. I already had been using the long edge for the search. This was in beta 19. In beta 20, I try to immediately correct an input (non-frozen) cs box has a length more than 3 times the width. I do this by rotating the pattern 90 degrees. Thereafter, I prevent that from happening during the minor moves.**
      13. **The team is having trouble with SimWebServer. Not too surprising, but I'm doing what I can to help. I answer questions, set it up so that the number of engines is configurable, went over how to do that, as well as setting the number of particles lower for testing multiple cases on a small 32-bit box.**
      14. **There seems to be a problem with a single web server and 2 planner problems running simultaneously. I've been chasing this one for a while. Rob sees the problem, Judy and I do not, but Rob's environment is perhaps more complex. I did notice that Rob's xml is referring to particle file names as UCN's instead of particle file names, but that turned out to be irrelevant.**
      15. **One problem that could cause this was the following, which I fixed: When an error in the data comes in, and is noticed by Engine-1 at the same time that Engine-0 is writing out the shape files, there was a crash. What happened was that I had to set System.err to null to keep from writing an err file that the rest of SAROPS was reading. That was happening in Engine-1. Then Engine-0 was using System.err, which should cause a null pointer exception, which it was, and this exception indeed was being caught. But in the handling of this exception, I had written code that again used System.err and this crash was not caught. Because of how we do logging now, I don't have to set System.err to null while writing the shape files, and I check for System.err being null anyway when I process the original exception.**
      16. **Related to that is that the logging mechanism is a bit changed. Logging goes to the case folder (plan.err.txt, or plan.out.txt or sim.err.txt or sim.out.txt). But the case folder is not formally specified. I had always used the folder containing the particle file. This is also the folder where I'm supposed to put the results file. But if the xml is "healthy" and the particle file is not, I want to log that problem as well; and I have no place to put it. Hence, I use a "generic" out.txt/err.txt pair. But if that is not available, then I use the "System.out/System.err" mechanism. This is redirected, by my installer, to <appDir>/LogFiles/SimWebServer.out.txt/err.txt for SimWebServer and similarly for Sim.exe. Unfortunately, these cannot be easily "cleaned out" by me so I overwrite them for every run.**
      17. **I finally found the problem I think. Rob has not reported bad behavior since my second beta 20 install. It was a needle in a haystack type of problem; at one point, I was taking max(a,b) and should have been taking min(a,b). Hence, I was tieing up all of the threads in the worker pool and there was deadlock.**
      18. **SaropsTrk 854 discussion: Worked on improving and correcting the POS report in SAROPS. The complaint is that SAROPS’ POS reports are slow. The POS reports within Planner’s internal gui are, by comparison, instantaneous so the poor performance of SAROPS POS reports is apparently not necessary.  
          But, the POS table in Planner’s internal gui displays the POS with respect to the selected particles, and it uses cached computations from the planner itself, which is downsampling to speed up the evaluations. The "Eval runs" have no cached calculations and are restricted to using all of the particles. I suggested at the IPR that we use downsampling to get Eval runs closer in performance to Planner’s internal gui's performance. I had a bug in the new code that “downsamples all” instead of simply “downsampling selected.” There was one place where it was simply not grabbing *any* particles.** **This was causing NaN’s in the output file. I found and fixed this bug.**
   2. **Travel completed:**
      1. **None**
   3. **Upcoming activities scheduled:**
      1. **Respond to bugs for code freeze. I hope I get them soon enough.**
   4. **Travel planned:**
      1. **None planned**
   5. **Concerns or recommendations:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Activity Worked** | **Hours Worked** | **Hourly Cost** | **Total Cost** |
| Kratzke | Coding/Doc/Travel | 79 | 230 | 18170 |
| Stone | Doc | 0 | 223 | 0 |
| Vergamini | Meeting/Consulting | 0 | 223 | 0 |
| L White (Tech Writer) |  | 0 |  | 0 |
|  |  |  |  |  |
| **Totals** |  | 79 |  | 18170 |
|  |  |  |  |  |