



(U) Letters to the Editor: Still More on Tool Development

FROM: the editor
Unknown
Run Date: 02/03/2006

(U) Some additional thoughts on the "tool development" discussion...

Comment:

(U) Speaking from an analyst/programmer point of view, there is no way that you can please all the people all the time. Everybody has a different job and has different ways to do that job. Anybody ever heard of the old phrase "If it isn't broke, don't fix it"? How many people repaint their houses or cars every couple years because they don't like the way things look? Isn't that what is going on here?

(U//FOUO) Aren't the majority of the tools being produced today presentation tools? The more complex the tool, the more encapsulated the data and the greater chance for data to never be seen or to contain errors. Give the analysts a set of tools to view and retrieve their data and provide them with a separate means to process it. Sure, we have more data to look at nowadays, and tools like Mainway enable analysts to do jobs they could never do before. But, what tools are available for the analyst to actually analyze potentially reportable data they are seeing on the screen? Or should they just assume that everything is correct and all numbers have been normalized correctly? How many analysts think they could accurately assess/analyze an event with just one screen full of data? If that could be done, then why don't we just have a "submit" button?

(U) **Instead of investing in different presentation assets, we should be investing in our analyst assets.** Technology will never be able to replace the analyst, but that seems to be the path we are going down.

-- Anonymous

Comment:

(C) In his Letter to the Editor "Getting Buy-In for Tool Development" ██████████ noted that there was "great resistance" to the introduction of the NUCLEON and CREST tools. What he did not say, however, was that that resistance was prompted by the fact that the new voice tools, at their introduction, did not work as advertised. Linguists found that they were much slower than the previous tools and were plagued by crashes and the inability to do some tasks that were common with the previous tools. It took a significant amount of time and effort by our linguists and tech directors to get the worst of the problems rectified.

(U) The problems were pretty much typical of new tools developed for this agency. Invariably, the tools are tested on small systems isolated from the stresses of overloaded networks and large numbers of users. This is to be expected -- you don't want to test

 **SERIES:**
(U) Roadblocks to Change

1. [Study Points Out 'Roadblocks to Change'](#)
2. [Letters to the Editor : About 'Roadblocks to Change'](#)
3. [Letter to the Editor : More on Tool Development](#)
4. [Letter to the Editor : A Tool Developer's Perspective on 'Roadblocks to Change'](#)
5. [Letter to the Editor : Getting Buy-In for Tool Development](#)
6. [Letters to the Editor : Still More on Tool Development](#)

your new software on a system where the hardware is going to mask software errors. The problem is that all too many of these tools are developed without consideration for the difficulties involved in scaling up from the test system to the actual user population. As a result, when the tool is introduced, its operation resembles molasses in January in Juneau. **If developers took more notice of scaling issues, their tools would face less resistance from the user population.**

-- [REDACTED]

Comment:

(U//FOUO) So far, all the comments on this topic have been very good. This is an important issue which is greatly in need of attention. My comments come from the perspective of providing tools to the end user (specifically) signal analyst (SA), and as having spent many years doing survey/collection work as a "collection engineer." It is my belief that the resistance to change is linked and exacerbated by the institutions of tool development, systems deployment, budget and all the other aspects of life in a large organization. There is no one problem or solution. Below are some of the issues which I believe affect the adoption of new tools and technology. Please note that the order of these is random and does not imply any particular ranking.

(U//FOUO) It is true that many analysts are "institutionally" resistant to change. However, as a provider of tools, I believe a greater issue is lack of experience -- and in my experience this occurs most often with military SAs. Now please, before everyone gets upset, it is not my intention to imply that military analysts are any less capable. However, they do face some unique and specific challenges to becoming truly proficient.

(C) The SA tech schools generally teach very specific tools and techniques to new SAs before sending them to their first assignments. What I see in the field, when deploying systems which may have new tools, is that SOME SAs know the tool and procedure, and may not always completely understand the theory of what they are doing. A specific example would be the use of XAVG and WVT, or XQAM and MERLIN for measuring signal parameters or demodulation. Both sets of software do essentially the same task, and it is usually a simple matter of looking where the appropriate settings are made. I cannot tell how often I have had to try to explain the theory and issues with frequency translation and filtering as applied to analog-to-digital conversion to SAs with years in the field.

(U//FOUO) For the military, this problem is exacerbated by the way their assignments are handled. Just as they are becoming proficient, they are PCSed and may need to learn a whole new target set, and a whole new tool set. I have even seen cases where a very smart SA was being PCSed to become a truck driver! Also, I have yet to see a field site with the luxury of enough analysts to do their jobs, AND provide in-depth training.

(U//FOUO) We also have problems on the developer side, which I see all too often. Developers of software and systems are often not users of their own product. An engineer may understand a certain protocol, but may not understand how the information from that protocol gets used further down the processing/analysis chain. Consequently, a very important feature may get buried within their tool, or not even get implemented. Adding to this, SAs for one target may need different tool features than SAs for a different

target. This makes it very difficult to make a "one-tool-fits-all" software tool.

(U//FOUO) The platform issue further complicates things (PC, DEC, SUN, etc.) with different tool suites for incompatible platforms being used by different offices, sites or agencies. Standardization to a common platform is a great goal, however, you can't force the implementation of a hardware platform before the software tools are ready. This only exacerbates the resistance to change. As the tools merge to a common platform, analysis systems will become more centralized, and the CCBs (configuration control boards) will have to become more flexible about allowing the use of additional tools on their systems - as long as the tool isn't incompatible and creates problems, and is properly documented and supported. This is the classic problem of management implementing a good idea without enough knowledge to think through all the possible consequences (e.g. the early problems with Beanstalk eating up valuable CPU time on mission processors).

(U//FOUO) Another of the many platform issues is the language used to write the new tools. The current platform of choice seems to be Java, which has many nice features. However, Java applications do not run well remotely (especially over SSH/SSL), and more and more we are being asked to remote capabilities. Java tools need to be written using a client/server approach, and Web Start. Web Start works quite well, but most systems are not set up with web servers or Web Start built in. Making this issue worse is the woeful state of our desktop web browsers. We shouldn't have to put in remedy tickets to get a browser and Java version which is compatible with Web Start.

(U//FOUO) As the project office responsible for providing the tools to the SA, our job is made more difficult by the uncentralized process of tool development. True, there are tool-depots, and there is a real need for all the utilities which specific analysts write for specific functions. However, the greatest need is for a comprehensive suite of those tools which does not require expert knowledge.

(U//FOUO) A final institutional problem exists which makes all this that much more difficult. As a whole, we are exceptionally creative at working around the problems and shortcomings we face each day. I believe that in many ways we are our own worst enemies because we find it easier to go out and create our own local solution instead of trying to get the bigger problem fixed. We treat the myriad symptoms which not only leaves precious little time for treating the disease, but in many ways makes it more difficult for to find and implement an appropriate solution.

(U//FOUO) I believe that all of these issues contribute to the real and perceived resistance to change. This is made all the more difficult by re-orgs, moving funding, etc. (What happens when a tool development team gets reorganized across fund organizational fund lines???) **This is a large set of problems, with no simple "one fix" solution.**

-- Charles Knutson

(U//FOUO) *Editor's note: At this point, we'll close out this series. Thanks to all for sharing their opinions.*

"(U//FOUO) SIDtoday articles may not be republished or reposted outside NSANet without the consent of S0121 ([DL sid comms](#))."

DYNAMIC PAGE -- HIGHEST POSSIBLE CLASSIFICATION IS
TOP SECRET // SI / TK // REL TO USA AUS CAN GBR NZL
DERIVED FROM: NSA/CSSM 1-52, DATED 08 JAN 2007 DECLASSIFY ON: 20320108