NETWARS Enhancements via Third Party:
Case Study, TNAPS+ to NETWARS Translator

Robert Richards
Stottler Henke Associates, Inc.
951 Mariners Island Blvd., Suite 360
San Mateo, CA 94404 USA
E-mail: Richards@StottlerHenke.com

Dan Sparrell
OPNET Technologies, Inc.
7255 Woodmont Avenue
Bethesda, MD 20814 USA
E-mail: dsparrell@opnet.com

Kristen Parton
Stottler Henke Associates, Inc.
951 Mariners Island Blvd., Suite 360
San Mateo, CA 94404 USA
E-mail: Parton@StottlerHenke.com

Abstract

Stottler Henke, in conjunction with OPNET and DISA, has integrated a TNAPS+ to NETWARS translator into NETWARS 2005-1. The translator, developed by Stottler Henke, was initially designed as a standalone product. The product was modified to integrate seamlessly with NETWARS. This paper emphasizes the integration process, i.e. how the translator was packaged to be as simple as possible to incorporate into NETWARS. This is the first case of a third party extension being incorporated into NETWARS, for which a formal process does not exist. The lessons learned will help potential future third party enhancements more easily integrate with NETWARS.

Introduction

The Tactical Network Analysis and Planning System Plus (TNAPS+) is a legacy PC-based tool utilized by the Air Force and other military branches for tactical communications planning and control (https://esc-digd.hanscom.af.mil/Tnaps). TNAPS+ assists the planner in building an exercise or operation database that consists of state-of-the-art commercial equipment and TRI-TAC equipment. TNAPS+ produces a series of output records describing the resulting networks and equipment configurations. The system controller can use this database, with program support, to monitor, manage, and reconfigure in-place communications. It is a detailed planning and engineering tool built around a database management system (DBMS). TNAPS+ supports tactical communications planning and control at two levels: the network level and the nodal/equipment level. Most of TNAPS+ is a Windows product with a small portion running under DOS but operating on the same database. An example of TNAPS+ Windows user interface is shown in **Figure 1**. The functionality that TNAPS+ provides is a subset of the extensive capabilities present in NETWARS. Whereas NETWARS is constantly being upgraded with new features and enhancements, TNAPS+ is a legacy program that is no longer maintained.

Presently, network design utilizing TNAPS+ is a requirement for many Air Force deployments. However, the entire DoD is moving towards utilizing NETWARS. A barrier to switching to NETWARS occurs because a user may be required to model their network in TNAPS+ and will not want to duplicate everything in NETWARS. NETWARS does provide extensive functionality beyond what TNAPS+ provides so there is incentive to model in NETWARS if the initial effort can be minimized.

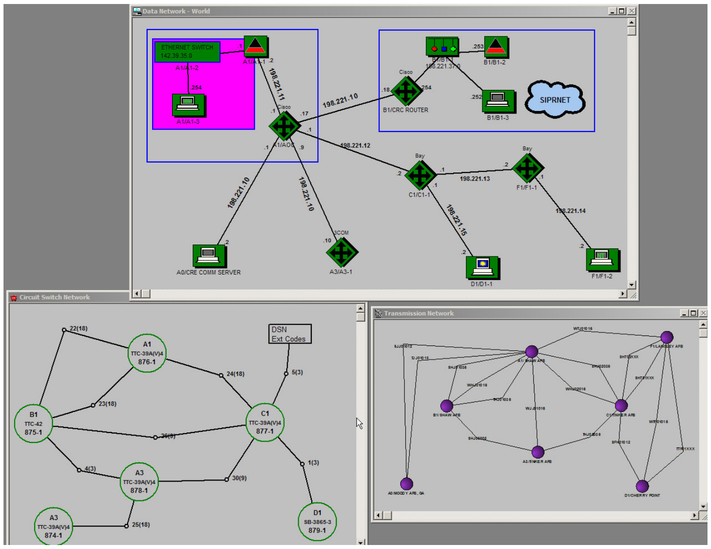


Figure 1 TNAPS+ network modeling.

To minimize the work required by TNAPS+ users, Stottler Henke has developed a TNAPS+ to NETWARS translator. With the assistance of DISA and OPNET, this translator has been incorporated into NETWARS 2005-1.

The translator allows the TNAPS+ user to benefit from much of the additional functionality provided by NETWARS without having to rebuild the network. This also makes training much simpler, because the Air Force presently trains users to use TNAPS+ with limited NETWARS training. Once modeling information is read in from TNAPS+, much less knowledge needs to be learned about NETWARS to be productive.
As mentioned, NETWARS has many capabilities beyond TNAPS+. An example is analysis based on location. TNAPS+ does not store geographical location information with entities; one can record textually where something is, but this has no meaning to TNAPS+. Once information is imported into NETWARS, location information can be added to allow the wireless transport analysis to be performed, etc. Most other types of analysis are not available in TNAPS+ either.

This paper focuses on some of the lessons learned per the incorporation of a third-party (Stottler Henke) developed capability (TNAPS+ translator) into NETWARS. This is the first attempt at a third party extension being incorporated into NETWARS, for which a formal process does not exist.

XML

This integration benefited greatly from the availability of NETWAR's XML schema. Stottler Henke had previously worked with OPNET products via the ODK (OPNET Development Toolkit). The availabilty of XML greatly simplified the NETWARS interface process.

Without XML, the translator would have had to make ODK calls for all the operations to rebuild the TNAPS+ network in NETWARS. With XML, the task was to understand the particular NETWARS XML schema and then output the TNAPS+ network into this XML schema. NETWARS already supports reading in XML; therefore, the translation can be performed without having to directly interface with NETWARS. That is, one can translate multiple TNAPS+ files, with each saved to its own file. These translated networks can then be opened in NETWARS at any time.

Although using XML was advantageous to the integration, it also presented some roadblocks. Using NETWARS XML is non-trivial for two main reasons:

1) The XML schema is still evolving
2) The XML import function is still evolving.

Verifying that the XML was created correctly and that the XML was being imported correctly proved a challenge, often because of subtle errors. This error identification required close coordination between Stottler Henke and OPNET/DISA. The XML import function had not been used extensively with XML files created by applications other than NETWARS, especially with files containing entire NETWARS scenarios, which might contain equipment connected in previously unconsidered ways. Because of this, the error messages given by the XML parser were very opaque and were not always useful for locating the underlying problems. Since the XML files were on the order of thousands of lines long, finding the problems by hand was often unrealistic. However, OPNET developers worked closely with Stottler Henke engineers to track down problems. Developers also took feedback from this experience into account and improved the error messages in the next NETWARS release (for instance, adding the line number of the parsing problem in the XML file).

One particular area that required enhanced XML support was in the area of multiplexors, due to their many different configuration parameters. Stottler Henke worked with OPNET's model experts to resolve all these issues. By starting with simpler TNAPS+ files and gradually adding more complicated network configurations, the team was able to work together to rapidly fix XML related issues.

Even with all the challenges encountered, it was obvious that the XML route was much more efficient and supportable than the ODK alternative. For third-party companies, having less dependence on ODK or a NETWARS codebase is advantageous because they can develop standalone applications that can be reused for other projects. Conversely, third-party companies that already have applications that could be utilized in NETWARS do not need to spend a lot of resources integrating the entire application with ODK or the NETWARS codebase.

NETWARS Versions and Directories

The integration actually first occurred with NETWARS 2004-2. This has provided lessons learned with regard to supporting different versions of NETWARS. Keeping track of previous versions is important for NETWARS, and thus also important for the translator. For example, since the XML schema is evolving, the XML output from the translator is different between NETWARS 2004-2 and NETWARS 2005-1. Since the XML schemas are not completely backwards compatible, it is necessary to update the XML output for each translator release.

In order to maintain compatibility between future versions of NETWARS and the TNAPS+ to NETWARS XML translator, Stottler Henke modified the design of the translator to allow plug-ins to be developed for each version of NETWARS. The translator is responsible for the core functionality of reading and parsing TNAPS+ files, transforming the data to match the NETWARS model, and intelligently filling in missing data (i.e., data required by NETWARS that was not present in TNAPS+). The plug-in implements an interface to the translator, and is responsible for formatting the NETWARS data into the proper XML schema.

Coordinating where the translator is installed in the release image was important. In order to install the translator properly for different versions of NETWARS, it was essential to be able to look up the version number dynamically.

Source Code Versioning

The translator code base was evolving repidly in conjunction with the rapidly changing NETWARS code base. Besides being affected by the evolving XML support in NETWARS, most of the rest of the code could be built to be independent of NETWARS code. This allowed Stottler Henke to utilize a revision control system (CVS) without needing to have access to the NETWARS code base (which was not available).

Compiler Support

Most of the integration was language independent since Stottler Henke created a standalone application that performed the translation. However, the standalone application required a small DLL that NETWARS communicated with, which started the standalone application.

The original Stottler Henke DLL did not work correctly with NETWARS because OPNET was utilizing an earlier version of the C++ compiler. However, the source code was compatible, so OPNET sucessfully compiled the DLL.

Another small problem encountered was that the translator required a few DLLs from the newer compiler (used by Stottler Henke). Although the addition of DLLs seems trivial to fix, it was quite difficult to actually find the cause of the compiler error. Once these DLLs were included in the distribution, the translator worked.

Meetings via the Internet

Communication between the geographically distant OPNET (District of Columbia area) and Stottler Henke (California) involved phone calls, email and Web based meetings. The Web-based meetings allowed for debugging that proved difficult via other means. The complete development process occurred virtually, there were never any face-to-face meetings between OPNET/DISA and Stottler Henke.

NETWARS Look and Feel

For user ease, we designed the external functionality of the translator to be as similar as possible to other NETWARS functions. For example, the translator is located in the same menu structure as other translators. The ability to integrate directly into the NETWARS menus and toolbars via text configuration files (ETS files) was a significant asset in integrating Stottler Henke's plug-in seamlessly with NETWARS. The subsections below show the translation process to detail how it has been integrated to look like any other functionality in NETWARS.

Running the Converter

Specifically, to run the converter :

1.

In the Scenario Builder, select

Scenarios->Import->TNAPS to XML…

See **Figure 2** below.

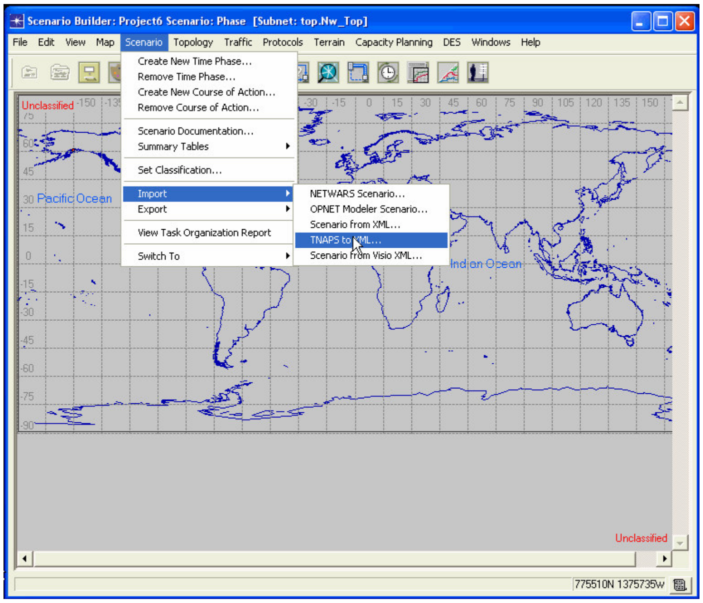


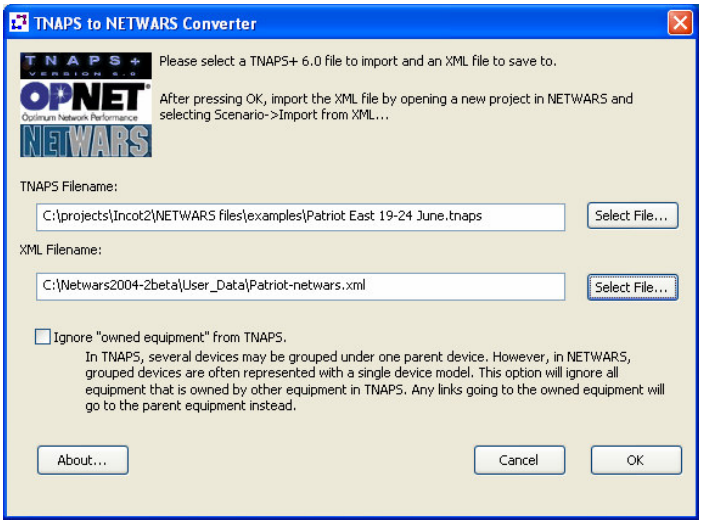
Figure 2. TNAPS to XML Menu in NETWARS that invokes the Stottler Henke translator application.

2.

The converter program will open in a separate window (see **Figure 3** below) and will not block the NETWARS program.

Select a TNAPS+ 6.0 file to open and an XML file to save to.

If you create a new XML file, then include the extension (.xml). Then click OK.



Please select a TNAPS+ 6.0 file to import and an XML file to save to.

After pressing OK, import the XML file by opening a new project in NETWARS and selecting Scenario->Import from XML…

TNAPS Filename: C:\projects\Incot2\NETWARS files\examples\Patriot East 19-24 June.tnaps

XML Filename: C:\Netwars2004-2beta\User_Data\Patriot-netwars.xml

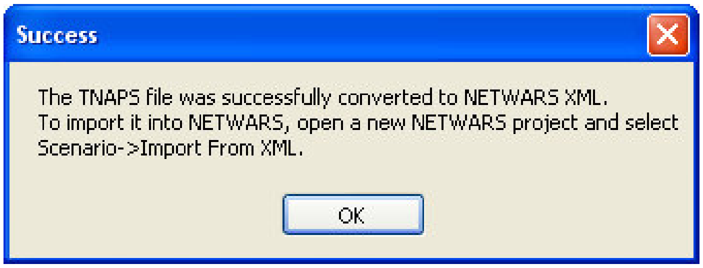
☐ Ignore "owned equipment" from TNAPS.

In TNAPS, several devices may be grouped under one parent device. However, in NETWARS, grouped devices are often represented with a single device model. This option will ignore all equipment that is owned by other equipment in TNAPS. Any links going to the owned equipment will go to the parent equipment instead.

Figure 3, Stottler Henke TNAPS to XML Converter, started from within NETWARS.

3.

The TNAPS+ file will be converted to the XML file and a "Success" or error message box will appear as shown in **Figure 4** below. When you press OK, the dialog will disappear. Nothing is imported into NETWARS.



The TNAPS file was successfully converted to NETWARS XML.
To import it into NETWARS, open a new NETWARS project and select Scenario->Import From XML.

Figure 4. Successful Conversion.

4.

Switch back to the Scenario Builder in NETWARS, and select New Project. Select

Scenario->Import->Scenario from XML…

See **Figure 5** below showing the menu choice.

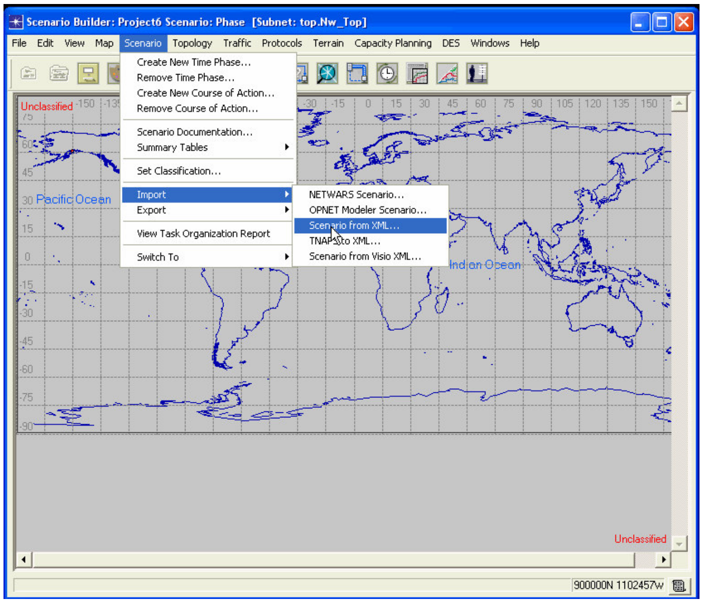
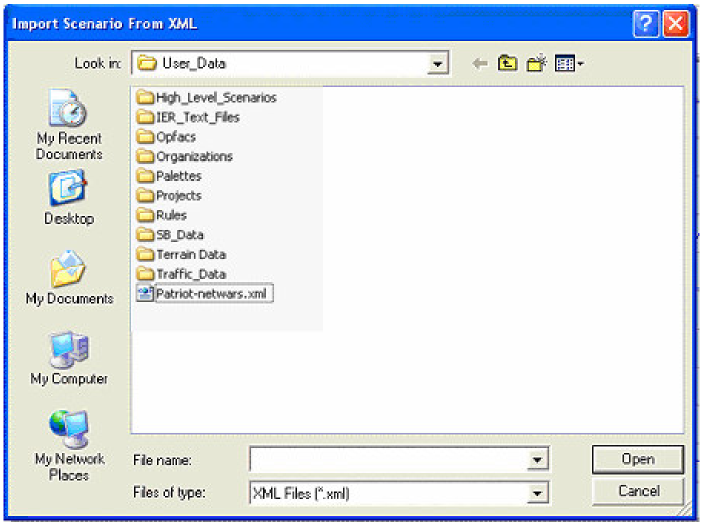


Figure 5. Import Scenario from XML Menu in NETWARS.

Then select the XML file created by the Converter, see **Figure 6**.



Look in: User_Data

File name:

Files of type: XML Files (*.xml)

Figure 6. Import Scenario from XML.

5.
Using the TNAPS+ file included with this converter, "Patriot East 19-24 June.tnaps", the network appears as in **Figure 7** below.

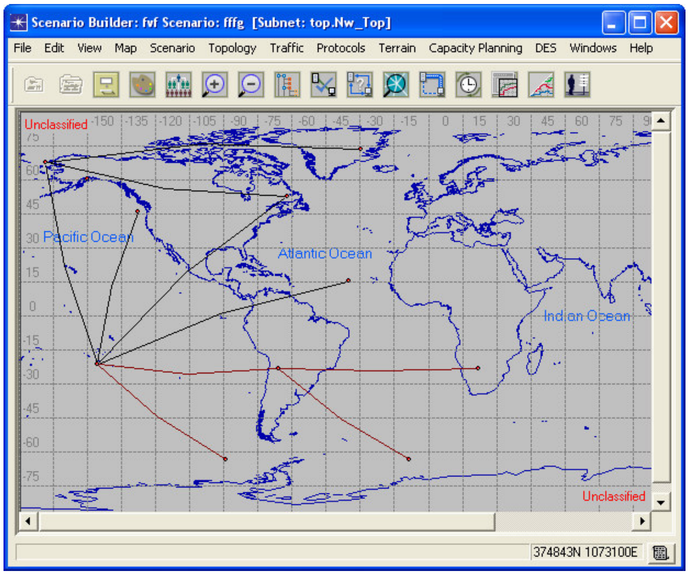

Scenario Builder: fvf Scenario: fffg [Subnet: top.Nw_Top]
File Edit View Map Scenario Topology Traffic Protocols Terrain Capacity Planning DES Windows Help
Unclassified
Pacific Ocean
Atlantic Ocean
Indian Ocean
374843N 1073100E

Figure 7. Imported Scenario.

The network includes all the original equipment and interconnections, which is exemplified in **Figure 8** below showing the results of drilling down on one of the top nodes

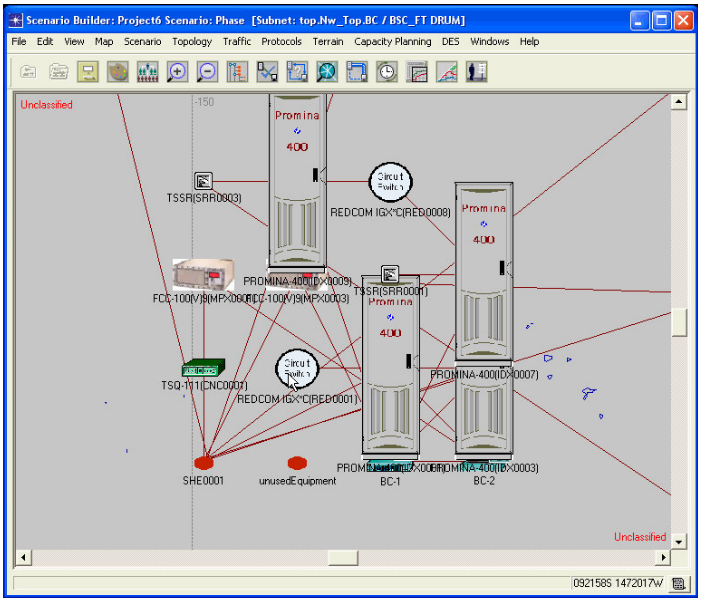


Figure 8. Details of Imported Scenario.

The Translator retains the hierarchical nature of the original network, as shown in **Figure 9** below.

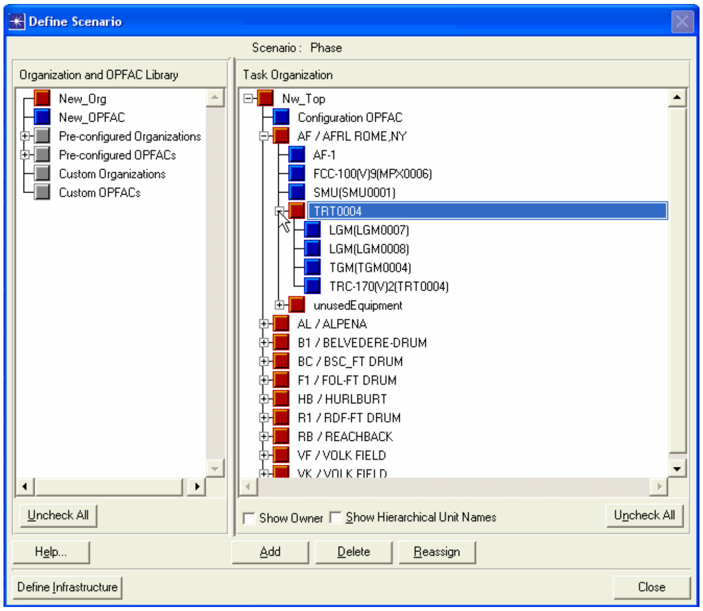


Figure 9. Hierarchy of Imported Network.

Examples Included with the Converter

A subdirectory called **examples** is created in the user's Netwars directory under:

%OPNET_DIR%\11.0.A\netwars\tnaps\

The subdirectory contains two sample TNAPS files called:

- **Patriot East 19-24 June.tnaps** – Example TNAPS file for testing.
- **Small.tnaps** – A smaller example TNAPS file for testing.

Summary

By working together, OPNET/DISA and Stottler Henke have provided a third-party enhancement to NETWARS. This enhancement is allowing legacy TNAPS+ network models to be re-used and analyzed in NETWARS.

Since this was the first case of a third-party enhancement, the experience provided some lessons learned that might prove helpful to other entities that attempt third-party enhancements. To summarize:

- If XML can be used to communicate with NETWARS, use it instead of building and maintaining another type of interface.
- Write code and other aspects of the third-party enhancement with NETWARS version information declared explicitly and plan for version dependencies.
- Try to create enhancement as encapsulated as possible so it can be called via a single command in a way that does not block other NETWARS functionality.
- Plan for iterative development, with an easy mechanism for quick web meetings.

Finally, even though it is not a lesson learned regarding development, the goal of any enhancement should be seamless integration in look and feel to give the user a consistent interface experience.