# New Solutions from Atraxis TPF Development

*by Peter Broad and Martin Burgess*

For many TPF developers, it must surely seem that many of the benefits and advances in the wider IT world have largely passed them by. So 'PROFs' may now have given way to 'Office' and the dumb-terminal may now be emulated on a slightly less dumb PC, but the basic tools of the trade have seem to have changed little in the last decade or so. The TSO/OS390 and VM environments still dominate most of the primary development tasks. This too is in the process of changing. The catalyst for this in many TPF shops is the adoption of the IBM VisualAge product and new source control and management tools. As the main development platform moves on then so must the various tools supporting this environment. In the TPF Development Services Group at Atraxis, we have developed, and are continuing to develop many such new solutions to help ensure a productive and stable future for TPF development. As well as traditional TPF, TSO, CMS and related skills, we have embraced C++, Java, XML, Unix, NT and JSP technologies to arrive at the best solution for our clients both internal and external to the company. Here are just a few examples of recent projects.

**Regression Testing**

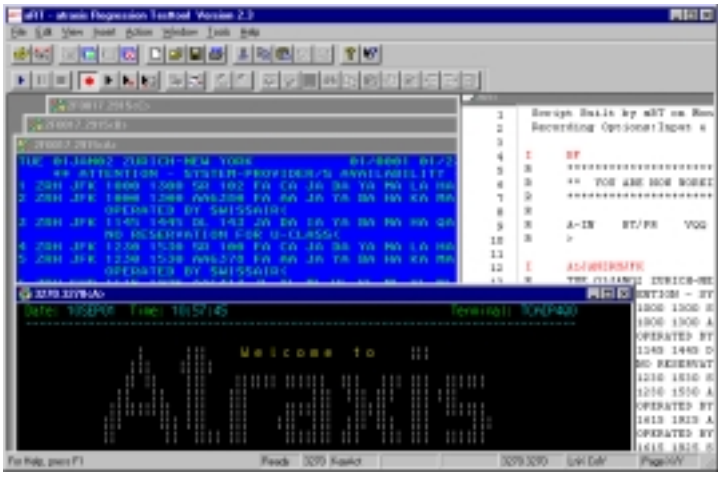
One of the most difficult, time-consuming but nevertheless critical tasks in development is testing. The decades of legacy code on many TPF systems makes this especially challenging. aRT, the Atraxis Regression Test Tool was developed with two main goals in mind; to ensure testing was consistent and repeatable and to make the whole process as simple as possible. aRT currently has over 200 users at Atraxis, Datalex and AMS. Everyone from developers, operators and system testers to marketing groups use aRT.

At its simplest, it functions as a PC based 2915 or 3270 terminal emulator, with the ability to record a dialogue between the terminal and the host system. The operation is extremely straightforward. The basic functions act exactly like a cassette recorder. This dialogue may then be stored as a script and later edited and 're-played' into the system. The new output can then be compared automatically to the original and any variation is shown in a report.

With the long and often cryptic entries used by the TPF system, this is a useful aid for development, maintenance and unit testing. The recorded scripts may then be added to a centrally stored database and built into a comprehensive library for package, product or system-wide regression testing. The database manages such things as version control and organization e.g. the storage of scripts by product, group, program and sub-program.

The execution of a script may also be scheduled to run automatically as a Windows NT system service. For example, at a specific time of night during a utility run or to run test scripts at a time when the system expects least traffic.

aRT has it's own easy-to-use scripting language. This can be used to loop and make entries multiple times, stop if the system sends a specified response and ignore certain parts of a response, such as a date or time, which are expected to change. It can also copy a portion of the host response and use it within a follow entry. For example, through a series of entries a record (such as an airline Passenger Name Record) can be created and the reference retained for later retrieval. aRT is also able to dynamically insert data retrieved from columns from within Microsoft Excel as a convenient data source during repetitive entries.



*aRT application showing a script, connection to a 3270 session and connection to 3 screens of a single 2915 session.*

**TPF Database Maintenance**

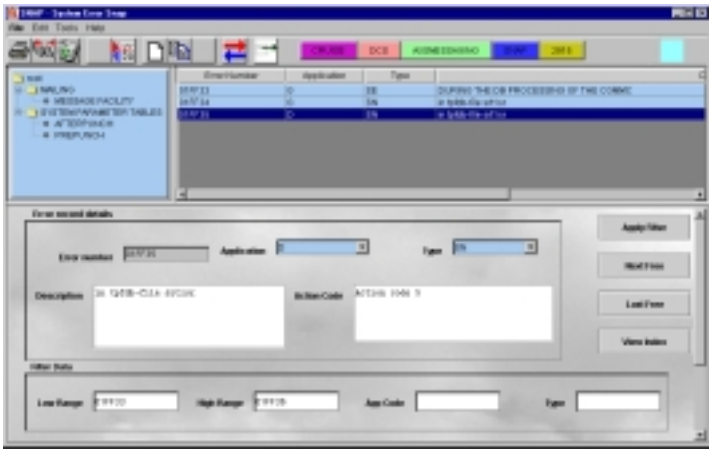
"Adept" is a Java GUI application which allows TPF and other proprietary databases to be viewed and modified in a convenient and user-friendly manner. It is modular in design and

can be extended to access virtually any source of data to which it can gain connectivity, such as a database through ODBC, XML data or even simple text files. Within Atraxis, it is mainly used with a general purpose TPF host interface that communicates using normalized data keyed with a transaction identifier. It can also be used to simply emulate existing, if somewhat convoluted TPF entries.

The application is comprised of three main panels;

The navigation panel (top, left-hand corner) is used to select the database to work with. A tree structure is used to allow tables to be grouped or to show relationships between the various data structures. Selecting a node on the tree makes the application active. Multiple connections may be active simultaneously.To the right is the data panel. This shows the data in a simple, tabular form. From here, an individual record may be selected.

When selected, the data is copied into the record detail panel below and may be edited, deleted or a new record added. The layout of this editing panel is configured using the WYSIWYG Editor in Visual Basic. The standard windows components such as a textbox or a checkbox may be used. Adept fully manages security, access control and permission levels. The next version Adept will include an enhancement to automatically generate templates from the labels from the macro source files, allowing dynamic access to all TPF files.



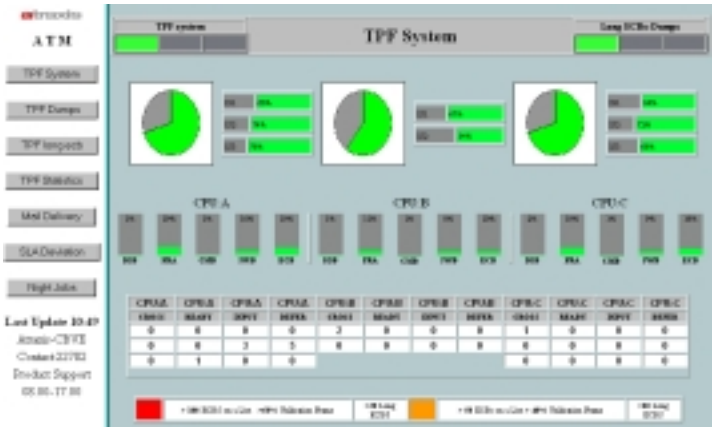
*Adept application connected to a TPF database*

**Statistics and System Monitoring**

TPF and supporting systems produce considerable quantities of data relating to their use and performance. Many groups require such data. In the 'front-line', the obvious examples are the operators and other technicians. However management, strategic planning and marketing groups also have their own requirements. The problems are related to accessing this information, storing and analyzing trends over a period of time, sorting what is relevant and presenting it in an a convenient manner. Not everybody needing system related information is able to check and note down what appears on the prime CRAS terminal and make endless series of "Z" entries!

At Atraxis we have two intranet based utilities. These produce both periodic reports on static pages and dynamic, 'real-time' reports accessed via a web browser. ATM, the Atraxis TPF Monitor displays information on system usage in tabular and graphical form. This is updated automatically in the browser every few seconds. The main page shows processor usage for each I-Stream across the complex, core usage, queue status and information on ECBs which have been executing for a long period of time. It includes a database, which logs more information about dumps and long executing ECBs. These can then be formatted into a report spanning a week, two week, the month or two months for long term analysis of application problems. The reports can be mailed automatically to designated persons. The final components of ATM display details of (currently) 20 utilities that run overnight on the system, such as the purge utility for the Flight Master Record and a monitor that records any deviation from a service level agreement for the Schedule maintenance.

The static utility uses simple HTML from any editor. To this it adds it's own set of tags to query the database, perform calculations with data and draw graphs and pie charts. No special knowledge of anything other than HTML is needed to configure the output for any system or application. The data can be gathered from any system that can output a text file using XML markup. As a Java application, it is also fully portable and can talk to almost any database.




*ATM application showing live system status data.*

**Product Monitoring**
"Atraxis Monitor" keeps track of the usage of our other tools such as aRT and Adept.

It is a server based application and receives information from the tools every time they are used. It records who is using what application, and version/revision numbers. This is used to generate a report, which is emailed to the responsible person periodically. The report shows a list of users and identifies those who are using old versions or those who are using an application with an expired license. This improves greatly the quality of support given to users of our applications, ensures that licensing fees are collected and service level agreements are maintained.

**Conclusion**
These are just a few examples of the tools that will replace and enhance those used currently under TSO or CMS. The primary advantage is one of communication. The considerable amount of information generated by TPF systems only has worth if it can be used effectively. Within a TPF based environment, new technologies and improved connectivity can greatly enhance the ability of programmers, technicians, planners and marketing groups to fulfil their roles. These people are now beginning to enjoy the same benefits that more mainstream users gained in the 1990s with the development of client GUIs.

Peter Broad and Martin Burgess.
Atraxis 10th September, 2001.
tpfcc@atraxis.com




## ACP•TPF *Today* ADVERTISERS



**Be sure to visit us online at:**

**www.tpftoday.com**

**Your first stop on the Internet!**