



THE NEW MEDIA SERVER

By Mike Fahl, Dataton AB

Abbreviations for dummies:

- CRT** Cathode Ray Tube
- IC** Integrated Circuit
- DMX 512** Digital MultipleXing lighting control protocol
- MIDI** Musical Instrument Digital Interface
- RS** Recommended Standard (eg: RS232, RS422)
- HTTP** HyperText Transfer Protocol
- FTP** File Transfer Protocol
- ISDN** Integrated Service Digital Network
- HTML** HyperText Markup Language
- TCP** Transmission Control Protocol
- IP** Internet Protocol

Undoubtedly, the advent of the Internet and intranets has set the stage for the next phase in computing and communications technology. Today, computers are no longer islands. Instead, they cooperate through networks to provide new levels of functionality. In this article, Mike Fahl, Dataton AB, shows you how you can use the client/server capabilities built into the latest version of Dataton TRAX to build media control solutions that are tailored to the specific needs of your customers.

IN THE BEGINNING, that is, about 20 years ago or so, there were no personal computers.

At that time, kids still had hobbies instead of Nintendos. I remember designing and building a two projector dissolve control unit for a friend, using entirely analog components, such as transistors, and capacitors. There was simply no other way in those days.

Although personal computers were yet to be invented, there were mainframes and their little cousins, known as minicomputers. My first encounter with a mainframe was a set of huge, refrigerator-style boxes in a closet, known collectively as the PDP-11. This monster served multiple clients using a network of CRT and teletype terminals. It ran a, for that time, very powerful, multi-tasking, multi-user operating system which went by the name RSX-11/M. Incidentally, that operating system's command line interface became the bedrock of CP/M, which—in turn—set the stage for MS-DOS, which is still at the core of Windows 95.

The single most important factor that enabled that computer evolution was the development of the integrated circuit, IC. Before that, computers had to be painstakingly built using discrete components such as relays, valves or transistors. With the arrival of the IC, thousands—and eventually millions—of transistors could be combined onto a single chip.

This has resulted in a dramatic reduction in size and cost, while performance levels have soared. For example, my first computer—a little, blue, shoebox-shaped thing—ran at 1MHz and had a total memory capacity of 256 bytes (yeah, bytes, not kilobytes or megabytes). Yet, it cost about US\$ 1 000 in 1977. That same amount of money will today buy me a 200 MHz machine with 32 megabytes of memory (not counting the 1.4 gigabyte hard disk). And then I haven't even started taken the effects of inflation into account.

The network is the computer

The fact that the computers themselves have become so cheap has allowed us to start thinking of them more as large ICs. In the same way that ICs were combined to build computers back in the 70s, computers are now being combined to form networks. Now, instead of collecting all data and doing all processing in one computer, multiple computers are networked to perform the job in a more flexible, powerful and safe manner. The computer that performs a service is known as a server, and the computer that uses that service is known as a client.

In a sense, this is a rerun of the old PDP-11 story, where the refrigerator in the closet was the server, and the dumb CRT terminals were the clients. However, today, the clients are about as powerful as the servers, allowing the clients to act as servers as well, if desired. This makes it possible to build not only the hierarchical networks of the 70s but truly distributed systems, where individual computers sometimes act as servers and sometimes as clients to yet other computers. The power of this arrangement became apparent in the unparalleled success of the largest network of them all; the Internet.

The true beauty of the Internet is that although it is a mishmash of computers and networks, it all behaves as one, cohesive, gigantic, supercomputer. When you're browsing the Net, you click a link on a page and, without even knowing it, end up in a totally different computer on the other side of the planet. When you want to find something, you go to Lycos, Yahoo, or any of the other search engines, and it finds the stuff you're looking for in a matter of seconds, regardless of where on the Internet it happens to be. I have to admit it: I sometimes find it easier to locate things on the Internet than on my own computer!

Gotta keep it simple

Those were the kinds of observations we had in mind when we set out to design TRAX version 3.5. We wanted to provide a single, uniform, network-savvy, scripting front end to our media control system. In the same way that the Internet has managed to hide the complexities of the various network standards and protocols under a unified interface, we wanted to provide seamless access to the powers of our system to whomever needed it.

In a sense, the situation in the show control business is similar to what's going on behind the scenes on the Internet, but with different acronyms. We have DMX-512, MIDI and RS422 where they have HTTP, FTP and ISDN. What it all boils down to is providing a uniform, yet powerful, front end to access and control all the stuff out there.

That kind of uniform front end has been the primary raison d'être for TRAX since we started development almost 10 years ago. For instance, in TRAX you use the same Locate cue to locate a frame on a laserdisc, a slide in a slide projector or a scene in a lighting console. Likewise, on the hardware side, you use the same SMARTPAX to control virtually any kind of presentation device from a basic PC running PowerPoint in a conference room up to a million-dollar star projector in a planetarium.

The act of reducing complexity while retaining power and flexibility is a rather tricky one. Particularly on the user interface side, where everybody seems to have their own ideas about how things ought to be done. Thus, in our pursuit of making things simple, we have sometimes been accused of being too "closed" on the user interface end. While we have supported almost any protocol and interface standard available on the device control end of our system, we have not allowed anyone to take control over the core of the system: TRAX itself.

Unleashing the power

When we started making computerized control equipment for audiovisual equipment, things were rather simple. One size fit all, so to speak. Not so anymore. Over the past couple of years, it has become increasingly apparent that this has become a highly diversified industry. Customers approach us from all angles, wanting to use our system. Often, they show up with a partial solution, and they only need to tie into our system for controlling a few this-or-thats. In that case, telling them "Sure, you can use our system; all you have to do is throw out all your existing software and reprogram it in TRAX" really isn't such a great idea. Yet that's pretty much what we've been doing up until now in the name of "keeping things simple."

In a nutshell, the dilemma we faced was this: How can we allow those people, with all their various computers, operating systems and applications, to take control over our system, without complicating the system so much that nobody ends up being able to use it?

Eventually, we came up with the idea of providing a back-door into TRAX, using a unified scripting language. Using this language, anybody would be able to access all the capabilities of our system, regardless of where they come from and which system they were using. In a sense, this is similar to the idea behind HTML which is the predominant protocol on the Internet. By providing a single, platform-independent, scripting language, with an open specification that's easy to understand and implement, we have finally unleashed the power of TRAX to anyone who wants to tap it.

According to script

The idea behind scripting is simple. You write a sentence that tells TRAX what to do and send it off to TRAX:

Set the run mode of timeline "Product Introduction" to play

The above line does exactly what it says; it plays the timeline named "Product Introduction". Likewise, a slight modification to the above sentence allows you to read back the status of the timeline:

Get the run mode of timeline "Product Introduction"

That statement returns "play" if the timeline is still playing.

Notice that the language used is very readable. It's a far cry from the bits and bytes that run back and forth through our network to actually accomplish what it says. ►

Not only can you control TRAX itself in this way, but you can also directly manipulate all aspects of the devices controlled through the SMARTPAX control units linked to TRAX. For instance, the sentence

Set the value of status "Level" of device "DMX 12" to 55

sets the output level of the specified DMX channel to 55 percent. Notice how the "Set" command is being used to talk to an individual device's level value as well as an entire timeline, as shown in the first example above.

You can also fire off complete cues using the "Perform cue" command, to control devices in more powerful and dynamic ways, such as cross-fading one lighting scene into another using a Fade cue.

As shown by those simple examples, you can access all the important functions of TRAX itself, as well as the devices controlled through it, using only the three commands "Set", "Get" and "Perform". As a matter of fact, the complete vocabulary of the TRAX scripting language lists only eight commands. Less really is more in this case.

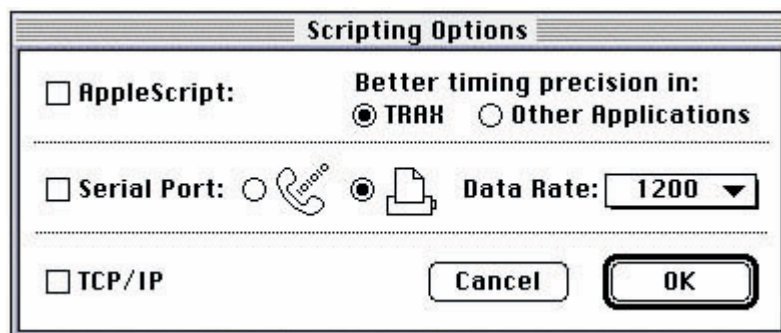
Scripting ports

If the above sounds interesting to you, I guess the question that's on your mind now is "How do I get these scripting commands into TRAX?" The answer to that question is threefold:

TCP/IP, serial port, AppleScript

As mentioned earlier; by providing these scripting capabilities, we wanted to make it possible for anybody to tap the potential of TRAX, regardless of the originating computer platform. Thus we have tried to provide as many and as flexible "inputs" to TRAX as possible through which you can send your scripting commands.

Although you may mentally think of these as three connectors, only one of them actually is a physical connector; the serial port. In this case, you simply specify that



one of the serial ports on the back of the TRAX computer is the scripting input, and send your commands through that serial port. Any replies come back the same way, as simple character strings. This scripting port is compatible with virtually any computer system or micro-processor based control system out there. It's reasonably fast and easy to understand. The disadvantage is that it is point-to-point—it only allows you to connect a single computer or other device to TRAX.

TCP/IP network scripting

The TCP/IP scripting port, on the other hand, is network based—allowing you to connect virtually any number of client computers to TRAX at the same time. Furthermore, it is also possible to talk to several TRAX computers at the same time from one client machine.

TCP/IP is a major standard in computer networking as it's the core protocol of the Internet. When I said it's not a "physical connector" on the computer, I was only partially correct. The truth is that it can use virtually any kind of connection, ranging from a simple serial port (using the PPP standard, popular for dial-up Internet connections) up to ISDN, Ethernet and Token Ring network ports. Using TCP/IP with a valid IP number (the unique "phone numbers" on the Internet), you could even put your TRAX computer up as a server on the Internet, directly accessible from anywhere in the world!

The speed you get through the TCP/IP scripting port varies widely depending on the kind of physical connection being used. In a typical set-up, using a dedicated Ethernet network, you'll get 10 to 100 times the throughput you can count on using the serial scripting port.

AppleScript

Finally, AppleScript is the standard scripting port provided by many MacOS compatible application programs such as Excel, FileMaker Pro, HyperCard and others. This can be used to control TRAX and its devices from such scriptable programs, running on the same computer or on another one connected via a network. Running both applications on the same machine works great for "off-line" applications, such as building a cue list from a database, or copying the script of a show into note cues along a timeline. However, the performance when running both TRAX and the client program on the same computer is usually not satisfactory for applications requiring good timing precision. In this case, running AppleScript from a separate computer via a network offers better performance, since most of the scripting management is then done on the client computer.

While AppleScript doesn't shine as a speed demon, it does excel as a user-friendly script development platform. As AppleScript was from the outset developed by Apple as an end-user scripting solution, there are many excellent tools available to create, test and debug scripts. Scripts, like any kind of computer programming, can become complex, and errors can creep in. In this case, the AppleScript programming environment and documentation, comes in handy. There's also a large number of books available that teach AppleScript from a very basic level. As the TRAX scripting language remains the same, you can deploy all the knowledge you gained from AppleScript through any of the three scripting ports.

A case study: Macromedia Director

Macromedia Director is a well-known program for creating computer animations and presentations. It is commonly used to build interactive kiosks. It excels at choreographing images on the computer's monitor, sound, CD-ROM and other internal computer media

capabilities. Director comes with its own scripting language, called Lingo, which is used to build interactive presentations. While Lingo gives you a great deal of control over what happens in the computer itself, it is rather limited when it comes to controlling things outside the computer.

By linking Director to TRAX, you get a solution that gives you full access to all the capabilities of a TRAX system from within Director. This link exists in the form of a Director plug-in software module, called an Xtra. This Xtra acts as a bridge between Lingo—the scripting language built into Director—and TRAX. In essence, it enhances Lingo with all the scripting capabilities of TRAX, as described earlier in this article.

As the Director Xtra communicates with TRAX through its TCP/IP scripting port, you can have multiple Director clients talking to one TRAX server at the same time. Likewise, one Director client could talk to multiple TRAX servers, if desired. This communication can co-exist with other kinds of network traffic on an existing network, allowing you to use, for example, already existing Ethernet network wiring.

Future possibilities

As we continue to develop additional scripting interfaces to other software packages and operating systems, you will be able to tap the power of TRAX in a wide variety of applications. Owing to the use of open protocols and a standardized scripting language, writing interfaces for specialized applications is relatively straightforward. Indeed, in the case of the Serial scripting port, all you need to do is hook up a cable between TRAX and the client, and send the commands through a serial port.

As always when we've brought out new inventions, such as this scripting capability, what impresses me most aren't the technical capabilities as such. Instead, it is all the incredible and ingenious applications where you, our dear users, use them. I have no doubt that one year from now, there will be a multitude of scripting applications which I could never have dreamed of—ranging from staid database applications to dazzling graphical front ends.

As a matter of fact, a few days ago someone told me they have already started developing a Netscape plug-in, which will allow you to control your TRAX system using a Web browser from anywhere on the planet. Considering that, as of writing, TRAX 3.5 has not yet been released, this sure holds a lot of promise for the future.

See you there! **Circle 22**



TRAX true multimedia software from Dataton runs on the MacOS platform. Due to the extremely versatile and powerful performance of TRAX, it has become the de facto industrial standard for show control and interactive audiovisual events.

NEW FEATURES IN TRAX 3.5

Although the scripting capabilities are among the most intriguing new features of TRAX 3.5, there are plenty more...

Use your computer monitor as an operator panel

If you have programmed TOUCHLINK using earlier versions of TRAX 3, you will know that it's the easiest way known to mankind to make a touch panel control anything from a conference room, to a rig full of AV equipment. Now you can access that same flexibility and ease of use from your own computer monitor.

Simply choose the "Computer Monitor" type inside the Panel device, zoom it up to cover the entire screen (including the menu bar), and use your mouse or a touch overlay as your operator interface. A password feature allows you to protect the system from unauthorized access, providing control only through your custom-made panel screens.

Play any audio CD using the internal CD-ROM

Simply insert a regular audio CD into the CD-ROM drive, add an Audio Disc device, and choose "Internal CD-ROM" on the type pop-up menu in the device's configuration dialog box. This gives you all the features you're used to when controlling an external CD player through a SMARTPAX, including precise synchronization and tracking during show editing.

Play sound files off the hard disk

For even more flexibility, add sound clips, narration or entire soundtracks straight from your hard disk. Most computers these days come with a hard disk in the gigabyte range, giving you plenty of room for sound even at full 44kHz, 16 bit stereo quality.

Use any sound editing software to edit your soundtracks, then play them back from inside TRAX in exactly the same way as you can control a CD for regular audio playback. Depending on which computer model you have, you can play several sound files at the same time. You can even do cross-fades between hard disk files as they play, as well as the audio CD.

Improve performance up to six times

TRAX 3.5 runs fully native on your PowerPC-based computer, giving you six times the speed on processor intensive assignments such as status calculations when moving along complex timelines, or the scanning of a long task list. This results in even better timing precision, tighter synchronization, faster screen-redraws and improved editing speed.

Needless to say, TRAX 3.5 can still run on 68k Macintosh models capable of operating System 7. The same TRAX application file is native to both the 68k and the PowerPC platforms, making it easy to move a show from one machine to another without having to worry about processor compatibility. And TRAX retains its remarkably low appetite for computer memory space!

Add the power of variables to tasks and cues

Have you ever felt there had to be a better way to manage the functions performed by multiple, almost identical, tasks? TRAX 3.5 sports an all new Expression window, allowing you to build arbitrarily complex (or simple) mathematical formulas, whose results can then be used in the task list as well as inside cues.

This allows you, for example, to fade multiple lighting channels using a single fader on a TOUCHLINK or a MIDI input. Or it can be used to create numeric key-pads, pass parameters to timelines, create generic cues whose behavior can be changed dynamically, etc.

More news

With the revised Locate cue you can move multiple steps forward or backwards—useful if you're doing stills off laserdiscs or similar. For even more flexibility, use a named expression from the Expression window, mentioned above, in place of the number, frame or time to be located.

A new task-triggering condition allows you to take action not only when inputs change from off to on, but also when gradual changes occur, such as a slider controlling the volume level in a room.

In the new Fader cue—your built-in lighting or mixing console—you can now specify relative levels, making it gradually increase or decrease by a specified amount from a preset level. Furthermore, you can scale the overall "look" of a complete lighting scene to tweak it, or link it to a named expression from the Expression window to control it fully interactively.



Wow, sounds interesting, if only I could get a TRAX demo in my computer within 15 minutes!

It's possible Jim! Just visit Dataton's Web site: www.dataton.com and download a free TRAX demo from the Product Info section!

