

# TAKING THE PLUNGE WITH CREATIVE USER INTERFACES

*While the device control capabilities of the system certainly are important, a creative and powerful user interface is crucial for a successful system design. After all, what good is all the high-tech stuff in the rack if the user interface isn't well adapted to the client's needs? In this article, Dataton's software expert Mike Fahl steers you from the shallow to the deep end of user interface options—from a simple, hard-wired push-button to the latest on networking, scripting and custom-designed software front ends.*

by Mike Fahl, Dataton AB

**W**HAT IS A USER INTERFACE? When I say “user interface” most computer-oriented people probably think of windows and menus. However, before diving into the deep end, I'd like you to take a step back and focus on the actual user for a while. Before you set out to design a user interface, you have to know some key things about the expected user(s) of the system:

- Is the system to be used by employees or casual visitors?
- If used by employees, what kind of training can be assumed?
- If used by visitors, what's their anticipated knowledge, background and age?
- What's the primary reason for the user wanting to interact with the system?

If there are too many or too complex answers to the above questions, it's most likely that you will need to design several separate user interfaces to your system. This will increase the complexity, and could potentially make the general user interface even more confusing.

Oddly enough, the last point of the four above is often overlooked, even though it may seem to be the most obvious. Last summer, I visited a museum that had a very sophisticated, touch panel based, control system newly installed. The panel had lots of buttons on several pages. However, it turned out that the primary use was to control the motorized curtains. In order to accomplish this, the attendant had to go into a locked closet (because the touch panel was too expensive to have it out in the open), navigate through a set of menu screens, and eventually push the Curtain button. The poor guy didn't seem to have a clue of what all the other buttons did – and if he accidentally touched the wrong button, he had to switch the whole system off in order to get it back to square one. In this case, a simple, hard-wired push-button somewhere out in the main room would have made a lot more sense, with the touch panel left for more sophisticated commands.

## Systems interfaces

Another factor that's becoming increasingly important in our technology driven world is interaction with other computers and systems. Often, the control system needs to be integrated with databases, the Internet or other external information feeds. Such external information can be directly integrated as part of the presentation, or interact with the flow or other aspects of the presentation in more subtle ways.

Sometimes, the user interface and the external information source may need to be tightly integrated into one piece of software. Perhaps the primary user interface is a database application used to register visitors. All that's needed in terms of 'show control user interface' is

a simple button to start the presentation. When clicked, the average demographics of the registered visitors determines the contents and flow of the presentation. In this case, the ability to integrate the control system user interface right into the primary user interface (ie, the database application) saves the user from interacting with two separate pieces of technology. As far as the user is concerned, her job is to enter the visitor's information into the database and then just “click that button in the corner of the screen” to start the presentation. I'll talk more about this kind of software integration later in this article.

## The invisible user interface

To enlarge on the pool metaphor, you don't always need to make a splash when it comes to user interfaces. In some cases, the best user interface is no user interface! Imagine a presentation environment that automatically does what it's supposed to do, when it's supposed to do it, without even being told. If you can figure out or anticipate what's expected, and just get on with it without any further ado, you have created the ultimate user experience. This is particularly the case if you've set out to inform or entertain. Having to wade through menu screens or punch in numbers can turn away even the most enthusiastic visitor.

Rather than forcing the visitor to humor the system, try to make the system respond automatically to the visitor. In its simplest form, a motion sensor strategically positioned can work wonders—making the system come to life with sound, light or visuals as the user approaches. More sophisticated systems may use visitor tags to track visitors throughout the environment, presenting information specifically tailored to each individual. When entering the presentation environment, the user is registered with some basic information (first name, sex, approximate age, etc). A transponder ID number is tied to this information through a database. Sensors throughout the environment then interact through this database and custom written software to control the presentation (more on that later).



If implemented properly, this can significantly enhance the visitor's experience. What's more, it'll happen without the user even noticing what's going on—just making him feel a little bit “special.” I recently visited the REX in Paris, an old, refurbished cinema now also housing what my friends at Dataton France call a “walk ride.” Here they had installed just such a presentation control system, using the technology in exactly the right, subtle ways. Suddenly, as I

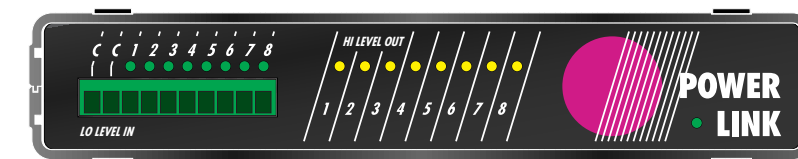
*The REX cinema capitalizes on a novel user interface in its “walk ride.”*

walked across what looked like an old stage, my name was called out, followed by applause. At another point, I was filmed without even noticing. I then appeared live in a short movie presented in a miniature-theater at the end of the ride (with my name in the credits!). Finally, when I left the cinema, I was given the chance to buy a videotape to take home, featuring *moi* as the hero. That's what I call a creative user interface—and I didn't have to push a single button.

## Going low-tech

Let's dip our toes in the user interface pool by looking at the most basic interface: a regular, mechanical push-button. The control system behind the scenes may be state-of-the-art, but that doesn't necessarily mean the user interface has to be. When installing a high-tech control system such as ours, it's easy to always think in terms of high-tech user interfaces. In many people's minds, a touch panel is the ultimate user experience. After all, it can present as many buttons and flashy graphics as you want. Some even allow you to run live video. What more can you possibly ask for?

The answer: simplicity and familiarity. Particularly when dealing with adult, casual visitors. (Kids don't seem to have problems with any user interface—usually it's the user interface that has trouble keeping up with the kids!) Most casual visitors just aren't that comfortable interacting with something that looks like a TV set or a computer screen. Besides, nothing usually happens when touching a TV screen anyway, so why risking making a fool of yourself?

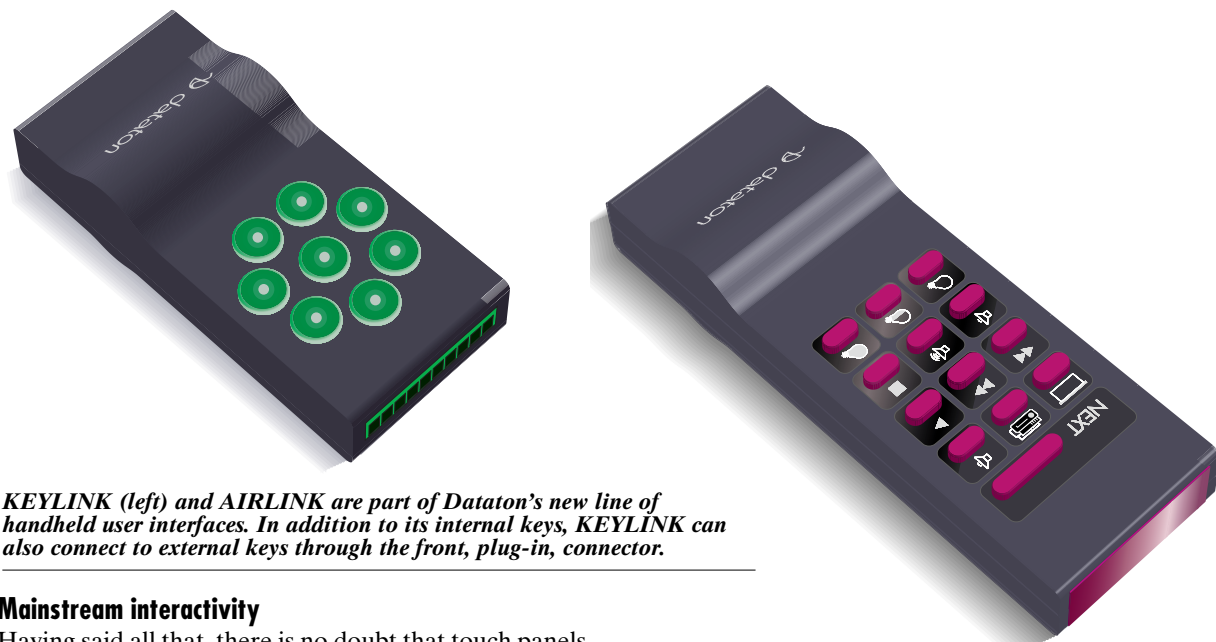


In this case, a good old, clunky push-button with a large sign next to it saying “Push this button to start the presentation” probably works much better. You can connect any button to our control system through a DIGITAL SMARTLINK or the newly introduced POWERLINK. Buttons are programmed through the Task window in TRAX, allowing you to combine them with other system status information, time of day, etc.

Another reason to avoid touchscreens could be ergonomics. If you're controlling the presentation environment as part of giving a lecture or guiding a group of people, having to look down on a touch panel to find a button can be distracting. A mechanical button can be easily found by your finger, without losing eye contact with the audience. The new KEYLINK provides eight push-buttons, complete with illumination for positive feedback, wired through a port in the control system. You can cascade up to four such keypads for greater capacity, or to distribute inputs over a larger area. AIRLINK is similar, but provides wireless remote control. All buttons are programmed through the Task window in TRAX, as mentioned earlier.

*Dataton's new POWERLINK allows the user to connect eight simple push buttons from the front panel, as well as switching high power loads from relays accessible at the back.*

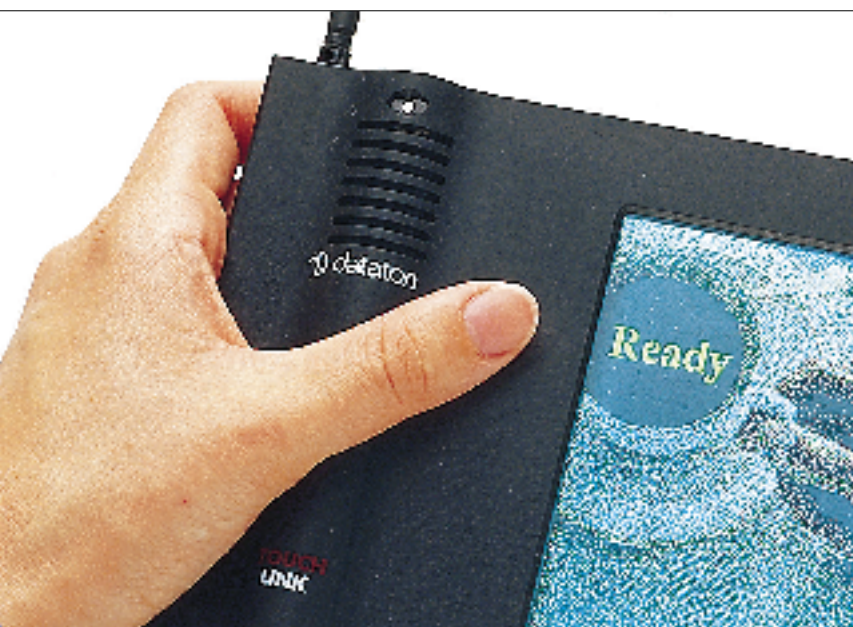




**KEYLINK (left) and AIRLINK are part of Dataton's new line of handheld user interfaces. In addition to its internal keys, KEYLINK can also connect to external keys through the front, plug-in, connector.**

### Mainstream interactivity

Having said all that, there is no doubt that touch panels and monitors have their proper place as user interfaces. They're often the most effective solution when dealing with trained employees or visitors with a high-tech background. A touch panel can provide a virtually unlimited number of buttons and other controls. Those can be logically grouped into pages that can be accessed from other pages or brought up automatically. User feedback is provided by text fields and other indicators. Graphics and textual instructions can further guide the user.



**Dataton's TOUCHLINK sports color graphics, a built-in speaker for sound samples, and a motion detector.**

Perhaps the most significant advantage of a touch panel is that it is very easy to change. Since the panel is configured through the panel editor built into TRAX, any function can be easily changed without rewiring or drilling holes for new buttons.

Alternatively, you can use a regular computer monitor connected directly to the TRAX computer. This monitor provides the same capabilities as TOUCHLINK, and is programmed in the same way using the panel design tools in TRAX. The monitor gives you a larger screen area, allowing you to put more functions onto each page. Some computers also provide video-in-a-window display capabilities, allowing live video to be seamlessly integrated into the user interface. While using the monitor as a user interface, the rest of TRAX can be password protected. You can use the computer's mouse to interact with TRAX, or substitute a touchscreen overlay, if desired.

Due to the flexibility and ease of programming of touch panels, they're often considered as the mainstream user interface for interactive applications and basic presentation room control functions. However, as discussed earlier in this article, there are also many instances where a simpler approach gives a far better user experience. And, there are also cases when casually floating along with the touchscreen mainstream crowd just doesn't cut it. So strap on your snorkel and get ready to dive into the deep end.

### Beyond the touch panel

As you know, Dataton is in the business of impressing people—not just making a screen go up and down. In order to build an impressive presentation system, it's often necessary to link up with other information systems, computers and software programs. Although a touch panel provides the ultimate in user interface flexibility, the buck stops right there—with the user putting his bare finger on the panel.

What if the user's requirements can't be met by just a touch panel? What if it is necessary to integrate the presentation system with other systems or a computer software? Many presentation system designs call for close

integration with other computer platforms or software standards, often dictated by externally imposed requirements that may have little or nothing to do with the presentation system. But, nevertheless, the systems must be able to coexist, share information, and to some extent control each other.

In order to meet such requirements, we've developed 'scripting gateways' into our system. Just like our system always has been very flexible at the back-end—allowing you to create your own device drivers and hardware interfaces—these scripting gateways open up the front-end of the system, allowing you to build your own fully customized solutions on top of our system. The following are some examples and case studies of what you can do with these scripting gateways, and how. But first, get a bird's eye view of what they are and how they compare to TRAX in the box at the bottom of this page.

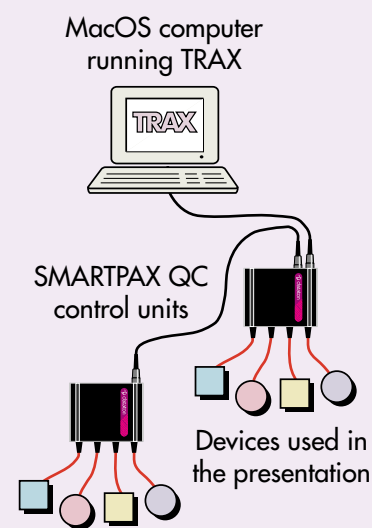
### Scripting showcase

To give you an idea of what can be accomplished by scripting, I'll take you through some applications and case studies. They represent a good cross-section of the capabilities at your disposal. However, designing a user interface based on TRAXSCRIPT or SMARTSCRIPT requires a bit more work on your part than putting together a touch panel interface in TRAX. In addition to understanding TRAX with its concept of devices and cues, you also need to understand the host application or development platform to be used in order to create your script-based user interface.

You can choose from a wide variety of applications running on MacOS and Windows 95/NT, as well as other computers capable of using the TCP/IP protocol (the standard Internet protocol).

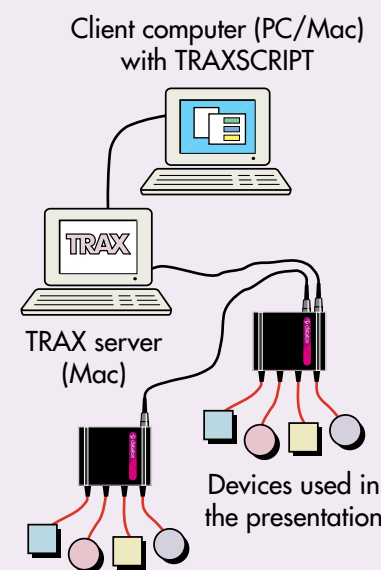
### TRAX

Dataton TRAX traditionally tops the chain of command in our system. Running on a MacOS computer, it provides all the configuration, design and programming capabilities needed by most users. It can also double as an operator panel by using the computer's monitor as a custom-designed user interface, while TRAX performs all its control functions in the background protected by a password. Furthermore, TRAX can play soundtracks from the hard disk or the CD-ROM drive built into many computer models, resulting in a very cost-effective solution.



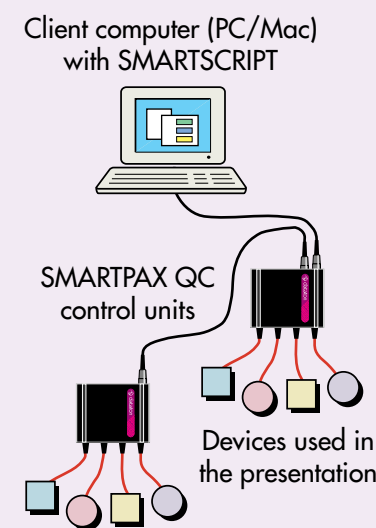
### TRAXSCRIPT

TRAXSCRIPT allows you to build fully customized solutions on top of TRAX, using virtually any kind of computer or software platform. It gives you access to all the capabilities of TRAX and the devices being controlled. You can start and stop timelines, fire cues, directly manipulate devices, etc. Multiple TRAXSCRIPT clients can talk to one TRAX server, or you can have multiple TRAX servers in the system. The connection between the TRAXSCRIPT client and TRAX can be a serial port, an Ethernet network or even the Internet.



### SMARTSCRIPT

SMARTSCRIPT provides a direct connection from a Windows 95/NT or MacOS computer to SMARTPAX, eliminating the need for a MacOS computer running TRAX in the final installation. SMARTSCRIPT gives you all the control capabilities provided by SMARTPAX and its device drivers, but doesn't provide the higher level functions normally handled by TRAX, such as timelines or precision synchronization. SMARTSCRIPT is suitable when you primarily need direct, interactive device control, or when the timing of events is controlled by other means (eg, from the Score window in Director, or directly by a VR simulation program as part of other actions).



## Macromedia Director

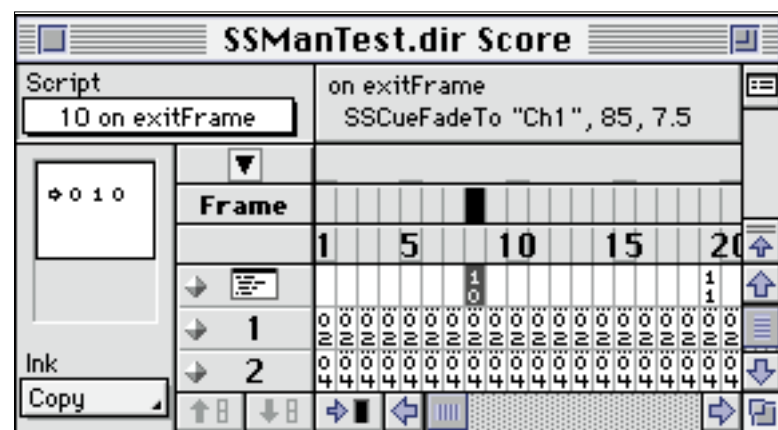
Macromedia Director is one of the most popular programs for building interactive computer multimedia kiosks, games and corporate presentations. It provides a timeline not unlike the one in TRAX, as well as a sophisticated scripting language of its own called Lingo. By using Director together with TRAXSCRIPT or SMARTSCRIPT, you can create preprogrammed or interactive presentations involving not just the computer's screen, but the entire presentation environment.

The TRAXSCRIPT as well as SMARTSCRIPT implementations for Director essentially enhance Director's built-in scripting language with new commands, allowing it to talk to TRAX and SMARTPAX. These capabilities are provided as "Xtra" plug-in modules, available for both MacOS and Windows platforms.

The illustration below shows the simplest possible use of SMARTSCRIPT directly from within Director's Score window. The script channel of the current frame contains a single statement:

**SSCueFadeTo "Ch1", 85, 7.5**

This tells SMARTSCRIPT to send a cue to fade the lighting channel named "Ch1" to 85 percent over 7.5 seconds.



The SMARTSCRIPT software package makes it possible to create user interfaces like this from within Director's Score window.

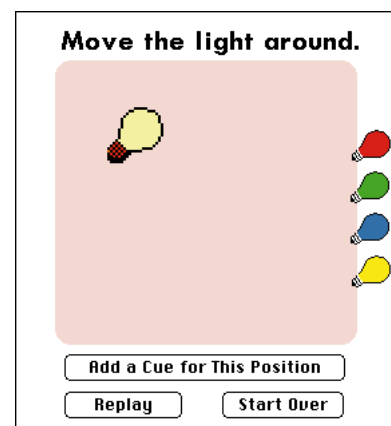
When using SMARTSCRIPT, you simply connect the Mac or PC running Director straight to the SMARTPAX using a cable from the computer's serial port, as shown in the SMARTSCRIPT illustration in the box on the previous page. SMARTPAX connects directly to the lighting dimmers and any other devices you wish to use with our smartlink cables.

Since the cues needed to control the external devices sit right in the Score window, they're performed by Director exactly at the right frame in the score, maintaining good timing precision with other Director-controlled events, such as sound or animations.

The next example uses TRAXSCRIPT to control a motorized lighting fixture. However, not only does it allow you to directly manipulate the light interactively, it also allows you to lay down cues along a timeline in

TRAX, and replay those cues by clicking a button. Thus, it acts as an extension of TRAX itself, providing additional capabilities not available in TRAX (in this case, programming of an X/Y positioning device).

Scripting in TRAX extends its native capabilities, as in this example with x/y lighting control

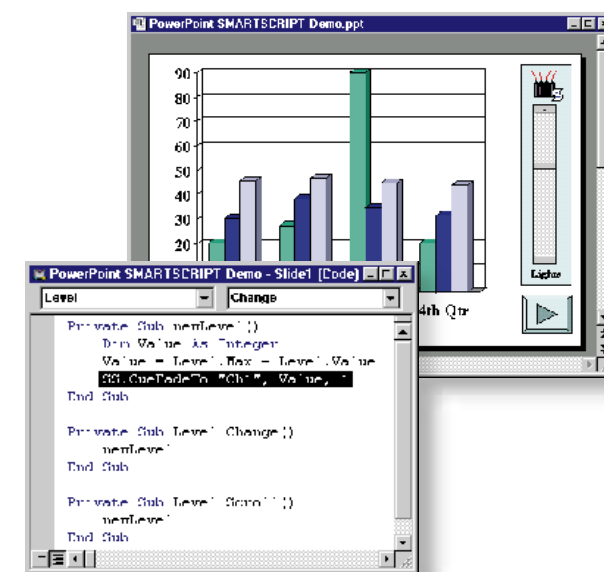


## Microsoft PowerPoint

PowerPoint is the de facto standard program for computer based speaker support presentations. While PowerPoint is usually associated with bar charts and bullets, it actually has quite a lot of multimedia capabilities, as well as its own built-in scripting language, called Visual Basic for Applications (VBA), allowing it to be used for building interactive presentations as well as the more traditional computer slide shows.

TRAXSCRIPT and SMARTSCRIPT is provided in the form of an ActiveX plug in module. This integrates seamlessly with the VBA scripting language, adding external device and multimedia control capabilities to your PowerPoint presentations.

The illustration (right) shows how a slider for controlling the room lights has been added to a PowerPoint slide. The highlighted line in the programming for the slider shows how the light value is taken directly from the slider named Level and used in the CueFadeTo SMARTSCRIPT command.



Technology to watch

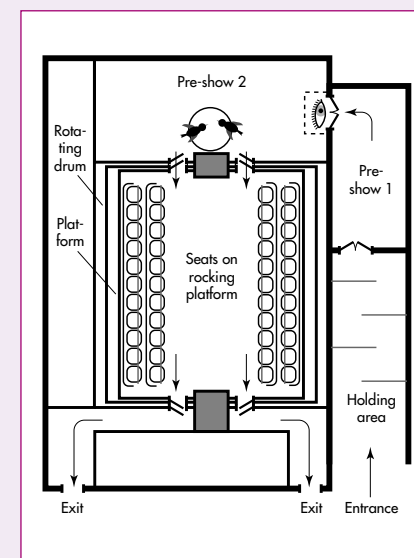
Controlling houselights from within a PowerPoint presentation is a snap if you get the hang of SMARTSCRIPT.



You can read more about the Valhalla installation at Tivoli in Copenhagen on page 30.

## Copenhagen Tivoli

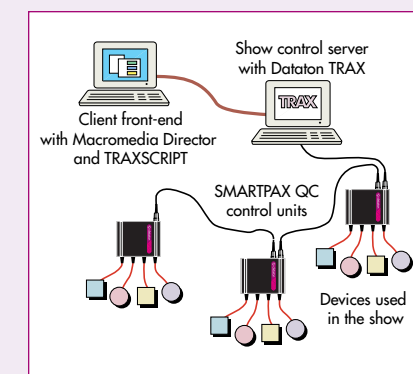
The new Valhalla attraction at the Copenhagen Tivoli amusement park is a combined animatronics show/ride. Housed in a large, box-shaped, wooden building, it consists of a holding area, two preshow areas and the main attraction. The holding area uses sound and live actors to create an atmosphere of ancient times. The two preshow areas add moving lights as well as animatronics in the form of Thor's eye and his two ravens Hugin and Munin.



The main attraction is a platform with rows of seats along two sides. The ride section commences with the platform slowly rocking back and forth, accompanied by lighting, narration and sound effects. The movement of the platform increases to a point where it feels as if it is just about to tip over. And then it does indeed tip over! Or, at least, it feels like it. But as my bag was still sitting on the floor, more or less, I realized that this was only an illusion.

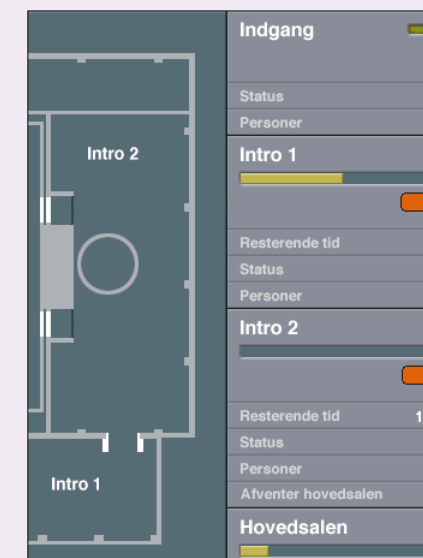
It turns out that the platform sits inside a giant drum. However, there's no sign of this when boarding the platform—the drum shell simply looks like the inside of the walls. As the rocking movement of the platform increases, the drum begins to rock in the opposite direction. This gives the illusion of an ever increasing movement, until the drum eventually spins full circle, making it feel as if the platform turned around. However, as it turns out, the platform turns only about 30°—that's enough to fool your senses when combined with the cleverly controlled movement of the drum.

From a control system point of view, the design was relatively straightforward. The equipment being controlled includes the audio sources (multiple CD players and Akai DR4 hard disk players), audio volume levels (analog control through VCAs), motorized doors, lighting and interfacing to the animatronics and the PLC controlling the drum and platform movement.



From a safety point of view, this gigantic washing machine could of course be extremely dangerous if started at the wrong moment. Thus, a separate, industrial PLC system was utilized to control the drum and platform movement. This interacted with the rest of the Dataton show control system through a DIGITAL SMARTLINK, which also provided other auxiliary functions and interfaced to the animatronics.

The interesting part of this installation—from a user interface viewpoint—was the client's requirements of interactive control, feedback, provisions for visitor statistics, and similar system management functions. The solution, developed by Dataton wizard, Mikkelsen Christensen, from MouseHouse A/S in Copenhagen, was based on Macromedia Director and TRAXSCRIPT. This allowed the user interface and other system management functions to be developed independently of the show control system, while taking full advantage of Macromedia Director for creating an attractive user interface (see partial screen image below).



This arrangement further allowed the computer running the Director-based user interface to be located in the operator room. The show control computer running Dataton TRAX was located in the machine room in the basement of the building—out of reach from unauthorized access. Furthermore, through the addition of a backup TOUCHLINK in the operator room, the system was set up in such a way that it could continue to run even in the case of a breakdown of the Director user interface.

email contact: [info@dataton.dk](mailto:info@dataton.dk)

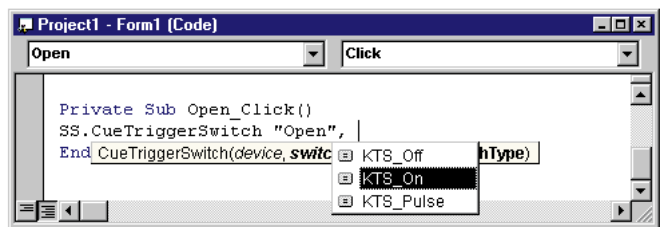


## Microsoft Excel

Just as you can integrate SMARTSCRIPT and TRAXSCRIPT in PowerPoint, you can also use them in Excel. This uses the same ActiveX plug in modules, and the same VBA scripting language as most other Microsoft applications. The example below shows how some buttons have been added to an Excel spreadsheet, allowing the curtains to be controlled.

SMARTSCRIPT  
used to add  
curtain control  
from within an  
Excel spread  
sheet.

When programming SMARTSCRIPT using any Microsoft application, small windows pop up next to the code you're entering, showing the applicable choices at each step. In the example below you can see a pop-up menu that appears while entering the SMARTSCRIPT command, allowing you to choose one of the things you can do with the switch device named "Open".



Online help while programming SMARTSCRIPT with Microsoft applications.

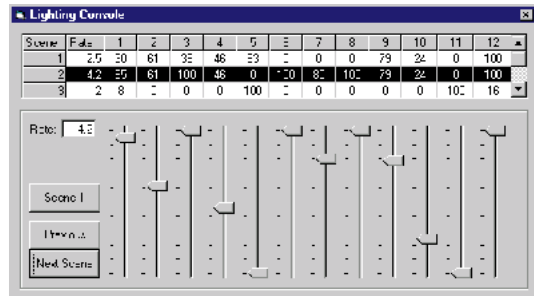
## FileMaker Pro database

This example uses TRAXSCRIPT to build a customized application that integrates a department store's FileMaker Pro database with TRAX. The database contains a list of the songs on all records stored in a large CD jukebox. After the user has made a database selection of songs to be played in-store that day, TRAXSCRIPT steps in, telling TRAX to create a timeline and add the cues required to play the selected songs. Finally, it plays the timeline, causing the CD player to play the selected songs.

Playing a selection of songs  
from a FileMaker Pro  
database is easy—with a little  
help from TRAX's scripting  
capabilities.

## Microsoft Visual Basic

The example below shows a simple lighting control program, developed using SMARTSCRIPT together with Visual Basic for Windows. Although there's a fair amount of Visual Basic programming required to make the buttons, sliders and the scene list work, only a single line of SMARTSCRIPT was needed to program this virtual lighting console.



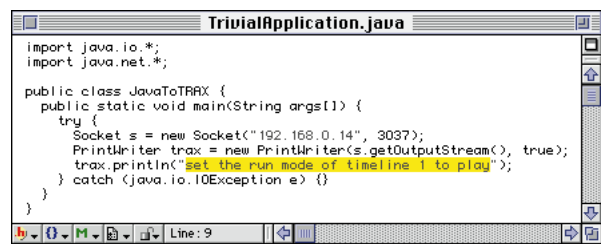
A lighting console created with Visual Basic controls real lighting via Dataton SMARTPAX units.

This example, and many more, are included—complete with all source code and full documentation—in the "SMARTSCRIPT for Windows" package, available in the "Free Software" section at [www.dataton.com](http://www.dataton.com).

## Java

Sun's Java programming language is becoming increasingly popular for building platform-independent user interfaces. Although often associated with the Internet, Java can also be used to create stand-alone applications, that have nothing to do with the Internet.

Java provides direct access to TCP/IP based computer networks. The same TCP/IP networking standard is used by TRAXSCRIPT. Hence, it becomes straightforward to write TRAXSCRIPT programs in Java. This is the complete source code of a Java-based TRAXSCRIPT program that tells TRAX to start playing a timeline:



Java used for commanding TRAX via its scripting port.

## Web browser

These days, it sometimes seems that what you can't do through a web browser isn't worth doing. People shop, listen to music, read news, etc, using the Internet. So let's finish our diving session by looking at how you can control your multimedia system using a web browser. Top next page is a slight variation of the Java program in the previous example, but here in the form of a Java applet, which is a small Java program that can run inside a web browser, across the Internet.



A Java applet running  
from an Internet  
browser controls TRAX  
over the Internet!

Is it really useful to be able to control your presentation room over the Internet? I have no idea (but, then again, the same thing could probably be said about many other things on the 'Net). But at least you know that it can be done, which gives you something to brag about at the next party :-).

email contact: [info@dataton.se](mailto:info@dataton.se)

## Software availability

All software required to implement a TRAXSCRIPT based solution is included with Dataton TRAX, our MacOS based production software. TRAX is available free of charge under the "Free Software" section at [www.dataton.com](http://www.dataton.com). The TRAX 3.6 User's Guide Addendum has all details on TRAXSCRIPT. The complete SMARTSCRIPT software package and documentation is also available free of charge at the same web site.

## Stop press

The latest addition to the growing family of scripting front-ends for the Dataton system is SMARTSCRIPT running on the Silicon Graphics platform. This allows environmental control to be integrated into simulations created using SGI computers. Such systems are commonly used in virtual reality centers. By opening up the Dataton control system to the SGI platform, external devices can now be seamlessly controlled as part of the simulated experience. The SGI version of SMARTSCRIPT will be made available shortly.

## BEHIND THE SCENES

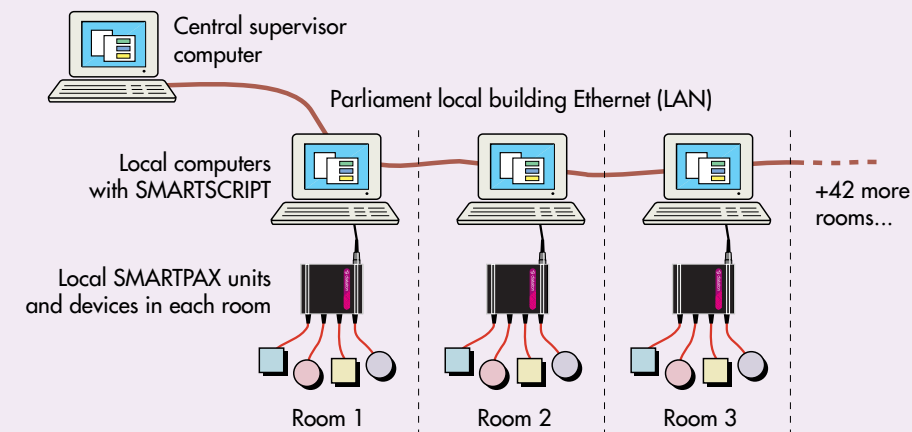


You can read more about  
this unique installation on  
page 24. The picture to  
the left shows the  
building in the final  
stages of construction.

## The European parliament

The European Parliament, Strasbourg, France, was one of the first major projects to utilize SMARTSCRIPT. Here, SMARTSCRIPT acts as a bridge between the computer control system—specifically developed for the Parliament—and the standard Dataton hardware used to control the presentation devices in each room.

Each of the 45 conference rooms has a set of local AV gear, usually controlled by the computer and/or a TOUCHLINK touch panel in that room. Local AV equipment includes video sources and projection, sound, lighting, etc.



A huge multi-room installation involving 45 rooms with distributed SMARTPAX units, PCs running SMARTSCRIPT, and networked with a central computer over a LAN.

In addition to the local computer that controls each room, a centrally located computer supervises the system through the building's Ethernet network. This supervisor can override local control functions in each room, and can also link control functions between rooms, allowing one room to temporarily use resources from another room, if desired.

Due to the open nature of SMARTSCRIPT, integration with the custom designed software by Dataton France's David Rouchet was straightforward. This combination allowed the customer's requirement in terms of computer hardware and software to be met, while utilizing off-the-shelf Dataton products for all device control functions.

email contact: [info@dataton.fr](mailto:info@dataton.fr)