Figure 1: The Clarity LX family.

LSC Clarity LX

By Rob Halliday

Is it just me who has noticed that Australians tend to do things differently? I think perhaps it's a facet of distance; certainly in the olden days that distance meant it was often easier for them to puzzle problems out on their own, then engineer their own solutions, than to go to the trouble and expense of shipping in products from the rest of the world. Even in this "always connected" world, I think some part of that still holds true.

Case in point: LSC, the Melbourne-based lighting manufacturer of lighting products from dimmers to data distribution to a range of consoles, and OpenClear, creators of lighting software. Five years or so ago, LSC bucked the trend of big, expensive consoles by launching a software-based lighting package called Clarity, backed up by a series of USB-connected wings for those who wanted more immediate, hands-on control than a QWERTY-keyboard or mouse could provide. In the time since, a number of the players in the "big" console world, notably MA and ETC, have moved in this direction, making wing/software versions of their leading consoles and touting their versatility and tourability. LSC has tacked the opposite way: The Clarity software and Clarity wings are still available (the software now called Desktop Clarity), but the company has also taken the software and created a range of great, big, real lighting consoles to wrap around it: the Clarity LX family.

In a world that already has MA, ETC, Hog, Avo, and others, this feels like an awfully brave thing to do. But, given the expense of designing and building real hardware, it's a decision LSC couldn't have made lightly. Certainly the Clarity LX has a physical resemblance to its rivals—but maybe that Australian "think different" mentality will bring differences in both software and hardware to make them stand out from the crowd.

Figure 2: Control area.

Hardware

Clarity LX is available in three sizes: the big LX900, medium-sized LX600, and baby LX300 (Figure 1). Common to all is the right-hand "control" area of the console, featuring, on the lower section, a backlit numeric keypad plus two separate, smaller function keypads; four rotary encoders that can also be push-clicked with a series of buttons above and below them; a level wheel and then what might be termed a "soft keypad" (which LSC calls the "command center"—a set of buttons arranged around a small LCD display (Figure 2). The display provides the button labels and so the ability to both adjust those labels according to con-

Figure 3: Encoder wheel touch screen.

text, and (through three "shift" keys beneath the display) to provide quick access to more functions. Above, on the upper section, is a 10.4" 800 x 600 touch screen which LSC called the "encoder wheel touch screen," surrounded by a series of regularly spaced buttons (Figure 3).

There's some different thinking on show here already, particularly in the balance of real keys to multi-function soft keys. Real keys around the encoders (these buttons are round, clearly signaling their relationship to those encoders) give immediate access to functions such as prev/next, fan, spread, home, knock-out, and fine, the last being either a temporary or permanent override depending on whether pressed and held or tapped to lock-on; all include indicator lights to show when they're in operation. Conversely, some seemingly critical functions—record, for one—are on command center keys. The logic seems to be that when you're directly manipulating the lights you want to stay "heads up," looking at the stage, rather than having to look down to find functions on a touch screen, whereas when you're recording, you'll already be looking down to check where to record.

This actually works pretty well, particularly since the text labeling those softkeys is big enough to read in a hurry, and sometimes provide useful extra information (what "undo" is actually about to undo, or which cue list you're about to record to). The functions on these keys are mostly customizable, but record can't be moved so your muscle memory can learn its position. Two other noteworthy things here: The first is the really high-quality feel of the encoders and wheel; they have a softness to their surfaces that makes them comfortable to the touch, the encoders' height makes gripping to turn easier than with other consoles (Figure 4), and all also offer a precise smoothness without any sense of wobble or the slop that sometimes makes wheels, in particular, change value slightly when you lift your hand off them. I do miss the

Figure 4: Encoders.

color-changing trackball of the Clarity wing, though! The second is the keys themselves, which are quite flat, but, because of their size and spacing—fractionally but noticeably bigger than on other consoles—are easy to work on quickly and accurately, and also quiet in operation.

The big differences between consoles happen to the left of and above this area. LX600 and 900 have a pivoting top panel housing four more sets of ten softkeys—LSC calls these "action buttons"—around small (120 x 140 pixel) color touch screens, then either one (LX600) or two (LX900) 17" 1280 x 1024 touch screens (Figure 5). The LX300 is clearly the baby brother here, its top panel at a fixed angle and housing 15 sets of playback buttons

Figure 5: LX600 touch screen.


Figure 6: Fader color coding.

above small playback touch screens. On the lower panel, the bigger consoles have motorized faders with full-height LED indicator strips that change color to show what mode each fader is in (15 on the LX600, 30 on the LX900), while the LX300 has 15 non-motorized faders (Figure 6). The faders can be paged, and you can split each bank into two separately paged sections or lock individual faders out of paging. If you need more physical faders, you can add the existing Clarity USB fader wings. To the extreme left, the bigger consoles have master go/back buttons plus a grand master; they also have a keyboard (complete, unusually, with numeric keypad and trackpad) hidden away in a drawer.

Common interfaces across the consoles are 4 five-pin XLR DMX ports that also support RDM, MIDI in/out/thru connectors, stereo audio in/out connectors (3.5mm jack on the LX300, balanced XLR3 on the bigger ones; as we'll see, these have more uses than on many other lighting consoles), and twin gigabit Ethernet connectors capable of supplying either Art-Net or sACN. LX900 adds a further

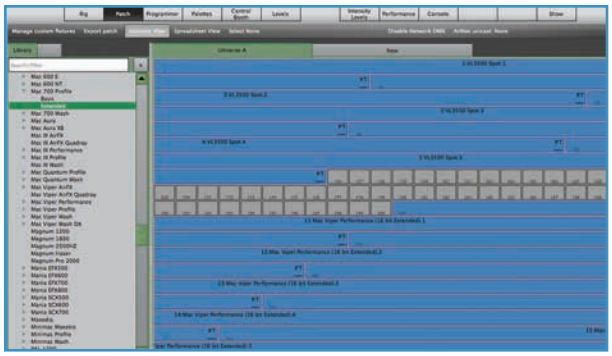

Figure 7: Patch.

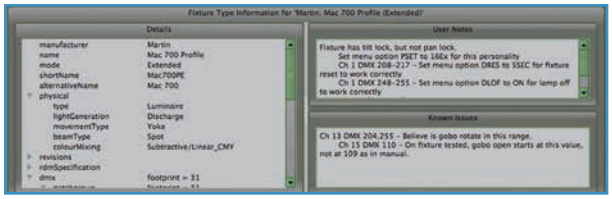

Figure 8: Fixture information.

Figure 9: Universal controller.

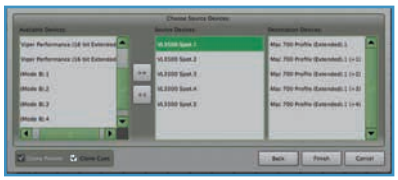

Figure 10: Clone.

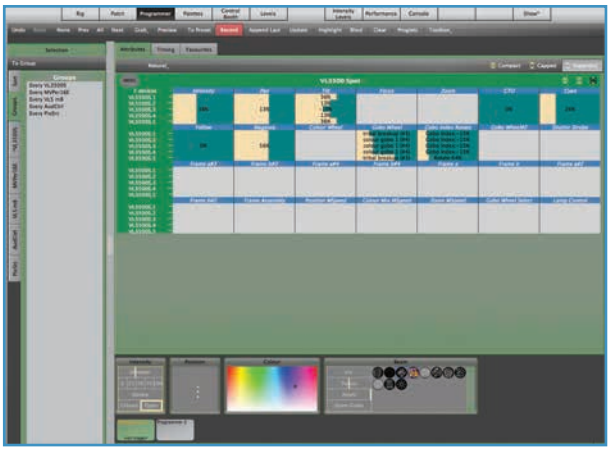

Figure 11: Programmer.

Figure 12: Two programmers.

two real DMX outputs and two DMX inputs. LX300 has a single DVI-D out connector for an external monitor, while the bigger consoles each offer dual DVI-D outputs. USB ports are scattered all over the place, including inside the keyboard drawer.

The standard DMX output capacity varies by console: 4,096 slots on the LX300, 8,192 on the LX600, 64,000 on the LX900. The processor is the same on all three consoles, though, so all can be upgraded to 64,000 slots. All now use fast, rugged SSD storage.

While budget will lead some to the LX300, and the very biggest live shows might demand the extra faders and capacity of the LX900, I suspect that for most people the LX600 will provide the best balance of features and price. Reflecting this, it was an LX600 that we had for testing (running software version 2.3.5)—and certainly it felt like a good "office" to be working in. Though the different-sized screens lend a slight wonkiness to its appearance, the smaller control screen means you're not lifting your hands so high to get to the top of it. It does feel like the programming area is quite a long way to the right—the master go is a bit of a stretch—but the real buttons with their little displays are great. And, a silly, but important, detail: a comfortable armrest!

Inside, it's a PC, of course, running Linux, though this is all thoroughly hidden from you. Useful is the ability to maintain multiple versions of the Clarity software on the console, so that if something goes awry with a new version, it's quick to switch back to using an earlier one.

Software

Of course, good hardware is really nothing without good software to bring it to life. Here, I'd imagine the challenge for both LSC and for the software's author, Nick Denville (one of the early Hog 2 software team before going on to create the remarkable time line-based Jands Vista; the "OpenClear" you see referenced in Clarity is Nick's company) was taking software originally designed to be operated largely using a cursor and adapting it to suit both touch screens and the "real" hardware of a lighting console.

The hardware incorporates details to help with this. As well as the buttons around the control screen, which give a physical connection to on-screen buttons, there are two more keys, at top right and left, which act as mouse right-click keys; holding either of these, then touching the screen gives access to the right-click functions in the software-only Clarity.

That works surprisingly well. Where the transition does fail slightly are the moments when it feels like the software is getting between you and the lighting, usually when an open dialogue box prevents you from doing

something else. It doesn't happen often but, when it does, that and the screen targets that sometimes feel a bit small for fat fingers, rather than a small cursor, are reminders of the console's origins.

The advantage of that history should be in the years of real user feedback the software has already had: We first looked at it in this magazine almost six years ago (see *LSA* February 2010), and the fundamentals haven't changed much since then, so it's worth taking a look at that review, too: http://plasa.me/duc6r. (To access this, you may be asked to fill out a short form.)

Starting a new show takes you first to the patch, its ease of use a good lead-in to Clarity: Pick a fixture type and drag it into the patch (Figure 7). The fixture library is from Carallon, as on many other consoles, but Clarity shows a lot of useful information (including DMX mapping, notes, and known issues) that other consoles don't (Figure 8). There is a separate tool included to make your own fixtures should you need to.

Two things—"clone from" in patch, then a display called "universal controller"—provide an insight to the way Clarity works. Universal controller tries to present you with a control view that is independent of the way a light actually works—so you always adjust a parameter called "index" and another called "rotate" regardless of how a particular light actually maps those functions internally (Figure 9). What Clarity ultimately stores into cues, though, are the DMX values required by the light. That's fine until you have to change to a different fixture type on tour; the old DMX values will make no sense to the new fixture. So Clarity's clone function will do an intelligent translation, figuring out what the old DMX values actually did and using the appropriate DMX values to achieve that in the new fixture throughout the show (Figure 10). The same process is used if you copy information from one fixture and paste it to another (through the real copy and paste keys!). Doing this translation once, rather than constantly, means that the processor isn't working to do this when it's also under load running a busy show, no doubt helping its ability to deal with 64,000 parameters without requiring expensive external processors.

Driving the console, there are two fundamentals to remember. One: You edit fixtures through a programmer (Figure 11)—with the twist that there are actually two independent programmers you can switch freely between (Figure 12), with the useful ability to release intensity first—i.e. lights fade out neatly before moving back to their previous position, and with a further twist of settings that make the programmer behave in a slightly different way preferred by those working in television, where the programmer levels are still mastered by a cue's playback fader. Two: This is at heart a tracking console, trying to store only "changed value" instructions into cues, and so allowing unchanged levels to track through cues. It does have options for modifying this behavior, for example by allowing cues to play back out of sequence affecting only the channels they have stored values for, but the language LSC uses for this mode (cue only) is so problematic that the manual has to issue stern warnings about being aware of the difference between a cue only record and a cue only cue.

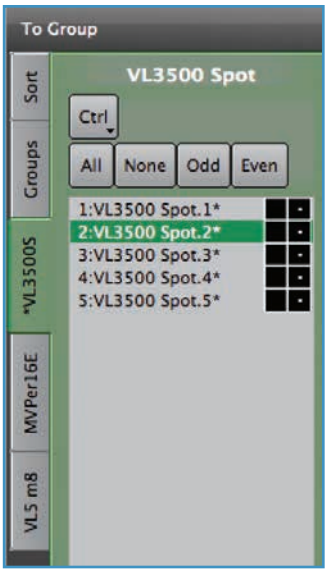

Figure 13: Fixture group lists.

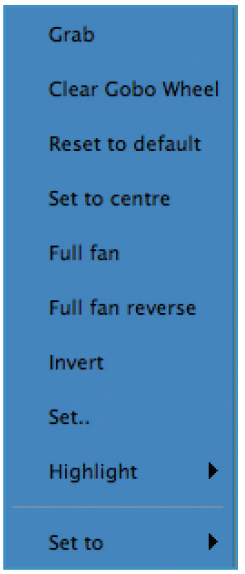

Figure 14: Parameter right click.

Figure 15: Group name in command line.

Figure 16: Icons.

The command line has really nice touches, such as displaying a group's name when you type its number (Figure 15). If you're graphically inclined, you'll want the "rig" view, a magic sheet-type layout with icons for both lights and groups that can be arranged as you like. Multiple layouts are available; the icons pack in a lot of information, giving a quick overview of what a light is doing (Figure 16), and you can import images as a background to the rig (Figure 17).

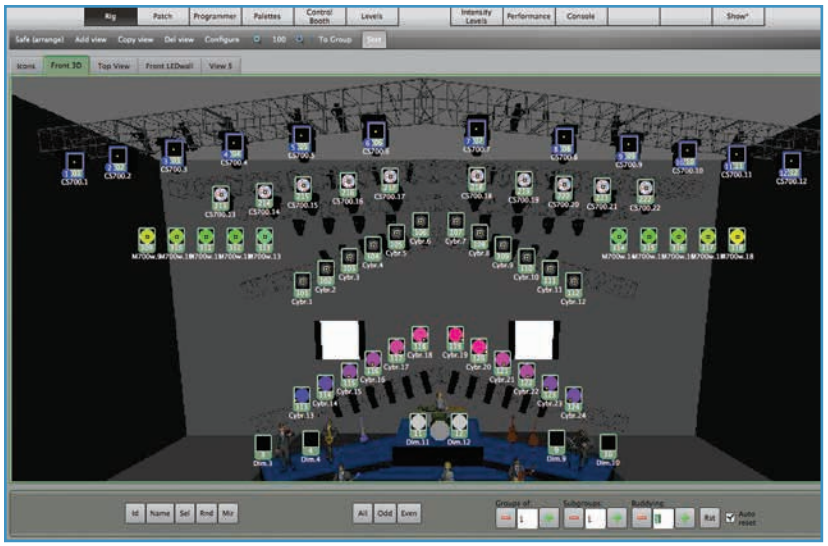

Figure 17: Rig view.

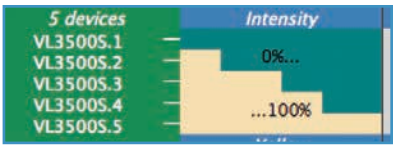

Figure 18: Levels within a group.

Figure 19: Group levels.

More detailed information is available in a variety of forms; the most interesting part of the Clarity interface was, and remains, how it evolves some of the pioneering work Vista did in terms of presenting information about groups of lights rather than just multiple sets of lights. Set a group of lights to 100% and you'll just see "100%" in the middle of the list of ten lights. Set the group at a range of levels from 10%—70% and you'll see "10...70" plus little bar graphs showing which light is at which level (Figure 18).

If you don't care about the detail, you can compress the group down to just one line with the level range showing (Figure 19). It's clever even if it can end up sometimes being confusing (if chan

1 is at FL and chan 10 is at 0, you'll still see "0...100" rather than the "100...0" that might be more helpful, for example); there is also a more traditional channel level screen, if you want to go back to that.

Those bar graphs provide a very intuitive way of editing times and delays, particularly when trying to range them across sets of channels: Switch to the timing view, then stretch or compress a block of channels to change the overall time, or skew it to offset delays (Figure 20). You can right-click and type precise times if you need to, but it's incredibly easy both to set and, later, see the pattern of a fade on-screen; accessing fade curves is equally easy, if you need to. Or you can just set an overall cue time, including, now, a separate down time and delay.

Where things can get a little confusing is in the similar-looking, but slightly different, ways the console can present information—for example, in the "universal" view (where you see the same attributes for every light—"gobo rotate" and "gobo index," for example) versus the "programmer" view, where you see the actual attributes in the light. Generally, I found myself wishing that Clarity did a better job of filtering out information not relevant at any moment: In universal view, seeing "gobo" on a VL5 (even if it's crossed out) just makes me pause to wonder why it's there—whether I've mis-selected something, though LSC argues that this makes the display more consistent across fixture types.

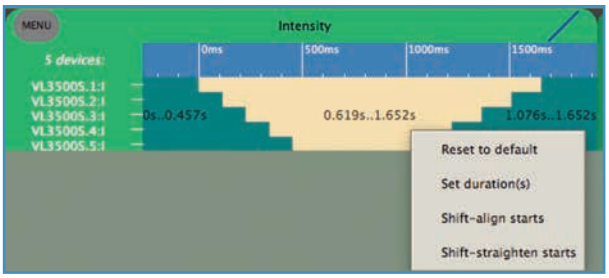

Figure 20: Timing.

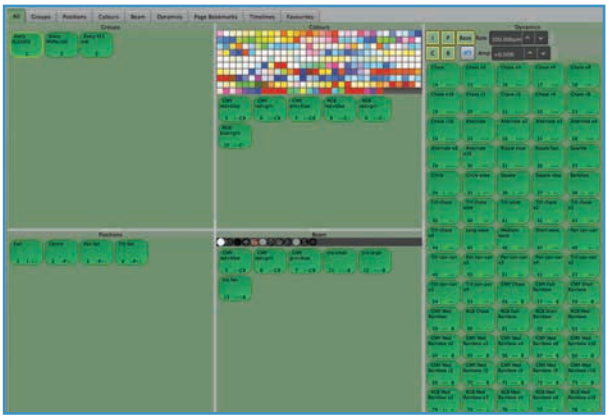
Figure 21: Palettes.

Figure 22: Dynamics.

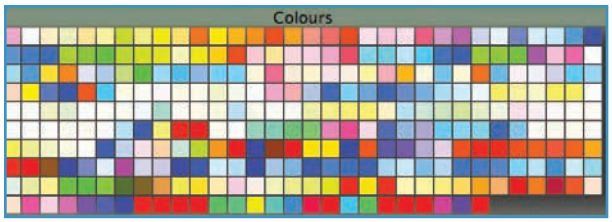

Figure 23: Freeset color library.

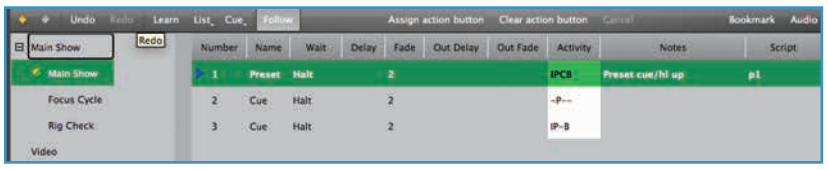

Figure 26: Cue list folders.

Figure 27: Metronome.

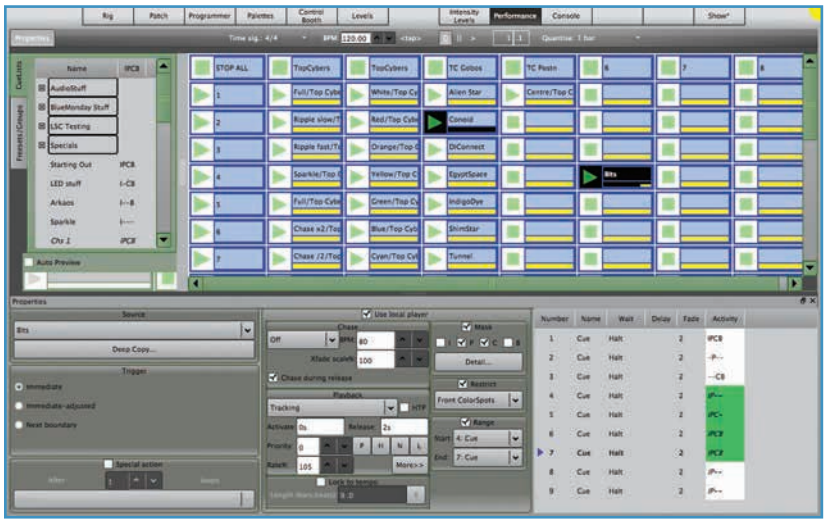

Figure 28: Performance window.

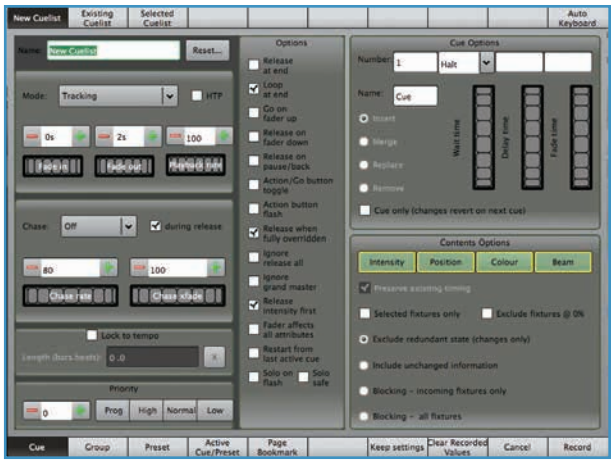

Figure 24: Recording new cue list.

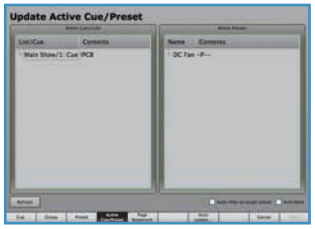

Figure 25: Update.

As usual, parameter data can be stored into referenced palettes, Clarity dividing these into position, color, beam, and dynamic categories (Figure 21), the last for storing combinations of modulator waveforms (Figure 22). You can store a mixture of parameter types into a palette; Clarity will split them out into separate palettes in each category with the same name. Clarity then has a separate palette type called a "freeset," intended for storing how you want something to look rather than how that look is achieved for a particular light. Clarity then uses its fixture intelligence to translate that look to data for a particular light. An example of this is that Clarity's built-in Lee, Gam, and Rosco color libraries are actually presented as freesets (Figure 23). Freesets can include ranges of data that can be applied across differently sized sets of lights, to create rainbows or fans, but freesets are not then referenced from cues. Presets and freesets are amongst the many things that can be assigned to the action buttons for direct access; a "favorites" tab also

lets you gather commonly used palettes together.

Look made, “record” will store it into a cue list on a playback, with options for naming it, applying timing, deciding what actually gets recorded including intensity/position/color/beam filtering (another area that could do with better filtering: Position shouldn’t be shown if there’s no position data to be filtered), and with the nice touch that when you first record into a new cue list, the settings for that cue list are also shown so they can be adjusted there and then (Figure 24). I did find myself constantly wanting to just complete a record by pressing the console’s enter key rather than having to use the “record” softkey. Change things later and there’s an update command, though one button actually leads to two possible versions of update and the whole update command feels like it would require too much consideration in that heat-of-the-tech, next-cue-approaching moment where you just want to update and go (Figure 25). A nice touch are the extra fields available on cues with friendly real-world names, “notes” and “script;” other nice touches are the ability to organize cue lists into folders (Figure 26), and a drag holding area—a place to temporarily dump cues as you’re scrolling around a cue list to find a new place for them.

Real-world names and functions can be found elsewhere in the console, particularly in the area of chases, where you can tie chase timing to a “metronome” and define a time signature, so that you can adjust the timing of running chases altogether as the band speed up or slow down (Figure 27). Clarity also has two other fantastic tools for dealing with busked shows. The first, the “performance” window, is a grid of playbacks into which you can dump one or many cue lists, then override the cue lists’ behavior (chase, rate, limited range of cues, parameter filtering) on a playback-by-playback basis without messing up the original cue list (Figure 28). The second is the

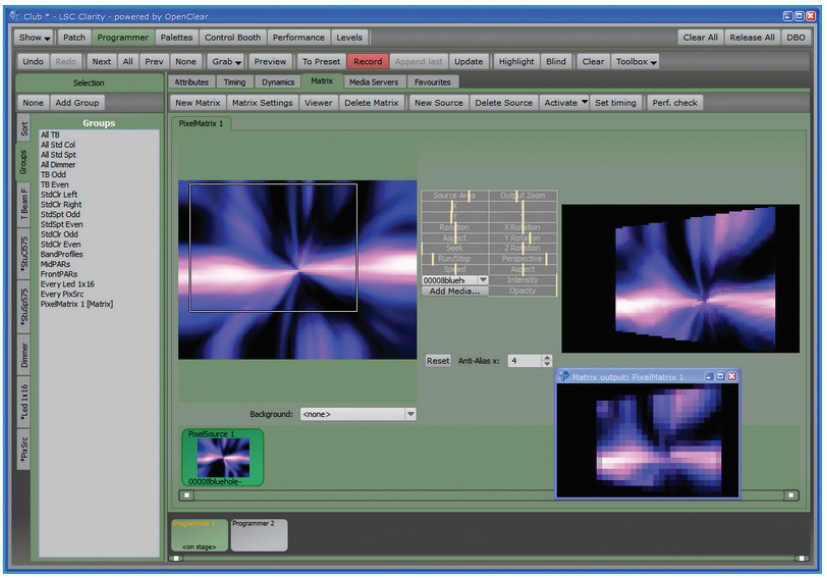

Figure 29: Matrix-800.

Figure 30: Media thumbnails.

ability to build up a long command line with multiple selections, actions, and complex timing, then trigger it all in one go—you could create a fly-out live using this alone. Beyond lights, Clarity offers powerful tools for dealing with arrays of lights using what is in effect a built-in mini-media server that can run still images or movie files across grids of pixels via its matrix control; each image setup you create is stored as a PixelSource, with each PixelSource appearing in the programmer group list as another fixture that can be faded in or out or have its parameters altered cue-by-cue, just like any other light (Figure 29). A range of transitions are also available to help you move from look to look.

Dealing with external media servers is also well-handled, Clarity not shying away from supporting the open CITP/MSEX protocol as well as the Hippotizer HMAP protocol plus HTTP

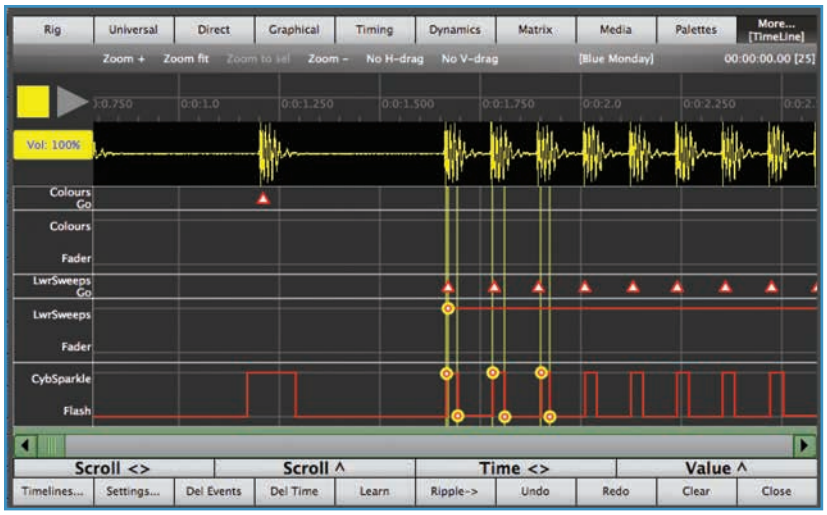

Figure 31: Time line.

Figure 32: Audio playback.

for Catalyst and Mbox, and using them to pull media thumbnails from the media server for easy file selection on the console. These remain cached for use even if the media server is then disconnected. This is a much, much more elegant way of working than manually defining presets for all of your media (Figure 30).

And talking of media, Clarity supports another media type: sound. Just as it can import images and video for use in its matrix system, it can import audio files in most common formats. One use for this is that the audio can be played and its waveform displayed on screen when working in the time code/time line editor, letting you precisely align events to moments in the audio (Figure 31). More fun is that Clarity can play back audio files on cue; one can either be treated as an audio fixture with volume and play/pause/stop controls, or a cue can trigger a clip directly (Figure 32). The first approach means Clarity can happily provide audio as well as lighting playback to productions (its balanced XLR outputs even making it sound engineer-friendly). For the second, LSC has a great story: an LD recording his spot calling and having the right calls played back by the relevant cues. The bigger consoles include speakers and headphone sockets, so you can hear the audio even without being connected to a sound system.

Media files added for video or sound can be collected together with the show file for transfer from console to console; Clarity calls this "exporting" a show rather than just saving it. The console also provides quite comprehensive tools for merging data from one show file into another, including the ability to merge data into palettes that have matching names; this is something I've longed for on other consoles. In part, though, Clarity needs this kind of show-merge power to make up for the fact that it doesn't have the networked-together, multiple users on multiple consoles collaborating on a common show file that its main rivals all have now. Sure, not everyone uses that functionality—but having it turns out to be unexpectedly useful on all kinds of occasions, and I suspect it will be missed on Clarity; however good the merge is, you still need to find time to do it, then check it. Clarity can network to another console or to the software version to use it as a tracking backup; it can also listen to the OSC show control protocol on the network, giving you other external control options.

Like any high-end console these days, Clarity's functionality goes wide and deep, and while the principles are often familiar from other consoles, the names and, particularly, button locations aren't. To help you figure things out, the user manual is accessible within the console. Taking advantage of Clarity's audio and video playback, there are also onboard training videos, hosted by a pop-up help version of LSC's Richie Mickan (Figure 33).

Summary

Clever and fun though Clarity LX is, there's an obvious question hanging in the air: Why buy one of these rather than one of the "standards," which in the UK are, arguably, the grandMA2 for big live shows and events, and ETC's Eos for big theatrical shows (though each often crosses into the other's territory). To be clear: Clarity is an immensely capable console more than happy to play in this league; for the most part, each can do what the others do; where one has some

unique feature, the others parry with another, and all now offer the software-plus-wings approach to supplement the big consoles.

But . . . on a purely practical side, with Clarity you do get a lot of console for your money. No-one pays list price, but it provides a useful yardstick: an 8192 output LX600 is about £22,000 ($32,000) in the UK, the grandMA2 Light about £5,000 ($7,250) more for 4,096 parameters. Both can ultimately drive more than 60,000 parameters, but doing this on the MA2 requires the external NPU processors.

At the top end, an LX900, with 64,000 outputs enabled, is £31,000 ($45,000), an MA2 Full Size about £7,000 ($10,150) more and an Eos Ti about the same as the LSC but with a much lower default output count.

Of course, console choice rarely (never?) comes down to money alone: It's about preference, capability and, of course, knowing how to work it—not just the programmer, but others who have to deal with the console on a show. The rival consoles have the advantage of masses of people with that knowledge; those armies are perhaps the biggest challenge facing LSC in getting more people to embrace the Clarity LX consoles.

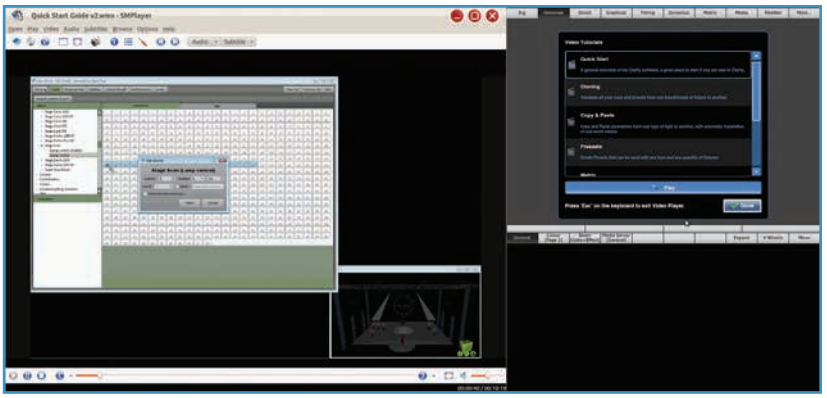

Figure 33: Pop-up Richie.

But people who use it, clearly like it—it's gained a lot of show credits, particularly in television, where they seem to like the "channel controller" alternative version of the programmer. The hardware feels solid, comfortable. Plus, as ever, the software is free to download and try (with new configuration options to make it emulate the layout of the various consoles, including on-screen representations of their control surfaces)—and if you don't try new things, you might never know what you're missing from that Aussie "do things different" mindset.