KIM BJØRN

PUSH TURN MOVE

INTERFACE DESIGN IN ELECTRONIC MUSIC

EDITED BY
MIKE METLAY & PAUL NAGLE

CONTENTS

INTERVIEWS

FRAMEWORK

Electronic instruments emerge in the mysterious space between innovators and musicians, driven by the ongoing march of technology and the visions of those at its leading edge. The designs are based on core principles, evolving technological possibilities, cultural trends, traditions, desires and imagination. The following model sets out the framework for this book and for exploring, understanding and studying the world of electronic instruments and their many forms.

USER

Electronic instruments seldom make sound without a human initiating an action or a process somewhere. As users, we tap on pads, press buttons or tweak knobs to generate, route or modify the sound. While doing this, we are engaged in a focused or explorative workflow, whether playing on stage or in the studio, in a collaborative situation or alone, and we react to feedback from our actions - whether visual or audible. We do, see, hear - and do again.

This human-machine interaction can be quantifiably successful, annoyingly stressful, or qualitatively joyful. It depends on our experience, efforts and results with a given device. User experiences are thus also dependent on a range of factors such as situation, skill level, and understanding.

SOUND

Electronic music starts with sound. Listening to sound is important when working with it, but visualization of sound adds a new dimension and supports further understanding. This can aid the generation, routing or modification of the sound while interacting with an electronic instrument.

Visualizing sound on an instrument, in a software interface or in an app, can be accomplished in countless ways. However, there are some commonly-used methods to help the user perform tasks across a variety of products and/or interfaces. These can include everything from reaching out for the sustain knob on a synthesizer to filtering out certain frequencies in a mix, adjusting the velocity or scale of notes on a sequencer, or beat-matching songs in a live DJ performance.

CONTROL

Sound is controlled by either being set or generated, routed/patched somewhere, or modified. Whether it's a full track played, mixed and effected by a DJ, or a waveform being modulated, filtered and routed in a modular synthesis system, the three main functions of controlling sound are: Generation, Routing, and Modifying.

To control sound we need to interact with it. Traditional techniques like adding vibrato to a string on an electric guitar or playing with a soft attack are commonly applied to synthesis, which employs knobs, faders, modulation wheels, buttons and many other types of control. This section examines the many control elements that directly connect the user to the sounds being made.

LAYOUT

The controls available on an electronic music device are laid out according to certain design principles that have evolved over many decades. These principles guide the user with respect to the function and relevance of the controls.

Whether on a hardware synthesizer, DJ mixer, Eurorack module or a music app, these principles derive from the way we perceive elements and objects in our environment. We ask: What is nearby? What should be grouped together? What is important?

Applying these ideas requires a mix of gestalt rules, ergonomic considerations, design traditions, and a wide variety of interface factors: visual and tactile feedback, legibility, familiarity, color, consistency, grouping, and more.

CONCEPT

Historically, electronic instruments have inspired a diverse spread of interface concepts. This chapter explores how these are realized in different software and hardware platforms.

Here we look beyond traditional categories such as drum machines, synthesizers, sequencers, etc. Instead, we'll consider the ideas that drive contemporary electronic music making, drawing attention to commonly-shared principles and interesting approaches.

TIME

Our experience as either maker or user is widely colored by the time we live in and therefore the technology that is currently available – and already established. How we use instruments and devices changes over time, as does technology, production methods, availability, aesthetics and style. The swipe on an iPad is today as natural an interaction for most musicians as sliding a fader or pressing a key. To put things into perspective, we look at some significant devices that defined or challenged previous and contemporary ways of interacting with electronic music – or managed to do both at once.

INSTRUMENT CATEGORIES

Over time, recognizable categories of electronic instruments and related devices have emerged but now, more than ever, the distinctions between them have become blurred.

Today, for example, a drum machine can be anything from the classic hardware device to an iOS app, a part of a software DAW, a small handheld device or a modular Eurorack system, and may also be capable of sampling, synthesizing, and sequencing other gear.

As such, it does not always make sense to put an electronic device in one of these categories. But in the context of interface design, usability and workflow, it's useful to have a brief look at the core duties of devices within them. The devices here are not visually nor technically descriptive of their category but merely serve as examples.

Workstation

A workstation usually contains a combination of several categories. As an example, it could be a synthesizer, sequencer, recorder, drum machine, and mixer. A self-contained workstation can either be a dedicated piece of hardware like the one shown above, or a unified software environment. By its nature, a workstation is capable of performing many duties, from musical sketch pad to final production platform.

Sampler

A sampler is a dedicated audio recorder designed to capture and play back audio in alternative ways. Mostly, the purpose is to arrange or rearrange source material to construct new music or to create a new instrument based on sounds or loops of individual notes. Detailed control of elements such as pitch, looping, shaping and maybe even sequencing and granular processing may be involved.

Synthesizer

The essence of a synthesizer is to produce sound on the basis of one or more synthesis methods. Typical tasks for analog subtractive synthesis would be choosing and tuning oscillator waveforms, filtering harmonics and deciding amplitude and filter envelopes, and finally choosing modulation sources and destinations to add movement to the sound. In addition to this, many synthesizers feature built-in sequencers, effects, and external inputs. Synthesizers come in many shapes and sizes and range from large modular systems to pocket-sized, battery-powered noise machines. They can be software, hardware, digital or analog, and have a keyboard or some far less familiar controller.

Effect

Effects may exist as independent hardware or software devices, or may appear as part of mixers, synthesizers, etc. Their task is to process an audio input and send the processed audio to another device. When using effects, the user should be able to conveniently route audio to the effect, adjust settings and parameters, and output the result. Mixing the effect with the original audio source and modulating effects parameters are also common functions. The distinction between synthesis and effects is often blurred, but for convenience we think of an effect as processing the final sound output (e.g. reverb, delay) while synthesis typically occurs 'per voice' (e.g. ring modulation, filtering).

Drum machine

The core duty for a drum machine is to provide a beat. This is accomplished with an internal sequencer, and usually some sort of musical division is apparent in the form of steps or pads. Typical tasks on a drum machine are the selecting and perhaps tweaking of sounds, the placement of triggers within musical measures, recording, playing and editing patterns of notes. Examples of drum machines range from software and small handheld devices to larger desktop units with sampling options or within a workstation, where drum sounds can be played on the keyboard and sequenced like any other musical note event. Effects, filters, modulation and the like also frequently appear on drum machines.

Sequencer
Sequences are the core basis of music – a linear process with a start and an end. When sequencing, we input notes or events, either manually or by activating steps on a grid. A dedicated, standalone sequencer rarely includes sound sources; its most important role is to keep your other gear busy! Sequencers are sometimes built into synthesizers and are an inevitable component of drum machines, but can also be found on sampling and effects units.

Mixer
The basic function of a mixer is to allow the user to route several audio sources to appropriate destinations in the audio chain, to make a balanced and coherent mix of them. In this process, it's important to not only be able to adjust individual channels of audio, but also add effects, combine them into groups, and process them separately or collectively.

Acoustic inspired
Electronic instruments looking and acting like their acoustic counterparts or mimicking their visual characteristics or playing surfaces. From the electric guitar, MIDI wind controllers, electronic mallets, and violins to electronic drum kits and much more. Though there are very different interfaces in this category, they have one essential thing in common, and that is to provide a familiar playing style on a somewhat known instrument surface, possibly with expanded features.

Recorder
The recording of audio in some form or another – whether analog or digital. The core task here is to record something and be able to play it back, whether whole performances for publishing, field recordings or careful recording of a piano for reproduction in a software sampler. When recording we want to control transport and levels – but also handle the recording with actions such as saving, loading, deleting and the like. We may also wish to loop pieces of audio, process them, or slice them into smaller units for use in samplers.

Stage piano/keyboard
Stage pianos are relatively simple keyboards with quick access to the common piano, organ and maybe synth sounds that are considered usable in a variety of musical settings. The primary objective with the interface design of an instrument in this category is to give clear and fast access to the most-used sounds and provide the expected controls over them. Front panels on this kind of instrument might be simple and free from visual 'noise' to allow fast orientation in a performance situation.

Controller
The controller category is diverse and includes everything from mixer-like interfaces to MIDI keyboards, grid controllers, drum controllers, etc. Members of this category don't generate sound themselves and are usually highly configurable to cope with a range of hardware or software. The essential duties of a controller are (besides controlling, obviously) to provide detailed configuration options and offer flexibility. A controller may be either very generic or specifically targeted towards a single duty (for example DJing).

INTERFACE TYPES

Electronic instruments have more faces than a diamond, and more shapes than a jellyfish. They gaze at us from iPad apps, self-built devices, massive modular systems and ware both hard and soft. Within the same instrument category there can be huge differences, yet across categories there are often surprising similarities.

An interface can include visual elements like displays, text, colors, and graphics; physical elements like knobs, buttons, and faders; or be as abstract as computer code or a motion sensing system.

Throughout this book, we will explore the many technological and visual approaches to interfacing with electronic instruments, and though there are many different concepts, we can start by grouping interfaces into these four overall types.

Touch-based

Touch interfaces are based on the user touching a flat surface. This surface can be more or less sensitive to factors such as the number of touch-points (fingers), the amount of pressure or the movements performed on the surface.

Touch-based interfaces are dependent on touchstrips, capacitive touch sensing surfaces or multitouch screens. Screens may be built into products or may represent the whole product, as in tablets for example. Such interfaces have many forms.

This is the ROLI Lightpad Block, which is a touch-based interface that's able to detect multiple touch nuances on its flexible but firm surface.

Tangible

Tangible interfaces are 3-dimensional and feature physical control objects like knobs, faders, buttons, keys and the like. One of the advantages of a physical interface is that the user does not have to look at the control being used, but can simply feel their way across, say, a set of synthesizer keys. It is traditionally the most widespread method of interacting with electronic instruments.

The Sequential Circuits Pro-One vintage synthesizer is a typical example of a highly tangible interface. It relies on the user grabbing controls, turning knobs, pushing buttons, and pressing the keys.

Graphical and text based

The graphical user interface is found in software, apps, on screens, and in virtual reality applications. It consists of visual representations of parameters and objects. Though the interaction with a graphical interface is done by using input devices like a mouse, computer keyboard, multitouch, or tangible controls, it is termed graphical because it only exists as a visual representation.

Text, graphics, colors, and menu systems are an essential part of the graphical interface – as shown here in the digital audio workstation (DAW) Logic Pro X from Apple. Visual representations sometimes resemble real-life objects, but they may also be very abstract in their graphical presentation.

Gestural and sensor based

Gestural interfaces process elements of a physical performance in order to translate the motion into musical or control data. For this, an array of sensor types exist, either as wearables on the performer's hand or body or via external tracking such as cameras, infrared sensors or components that register the electrical current of the human body.

One of the best known gestural interfaces is the theremin, which creates sound according to proximity with two antennae, one resulting in changes of pitch, the other of volume. Moog's Theremini is a modern interpretation.

USER

Today, almost everyone has access to electronic instruments, whether they be artists, DJs and composers or students, hobbyists and casual dabblers.

The term 'user' might seem cold and impersonal, but the intention is anything but. If the user is not placed at the center of development, well, let's just say that there are plenty of failures scattered through the history of electronic instruments. Of course, it's still possible to have a success without considering the humans who will play the instrument - but why begin with obstacles to overcome?

In this chapter we take a close look at the user and the interaction process. We'll consider the user experience from practical real-world perspectives, noting factors that are important during interface development or iteration.

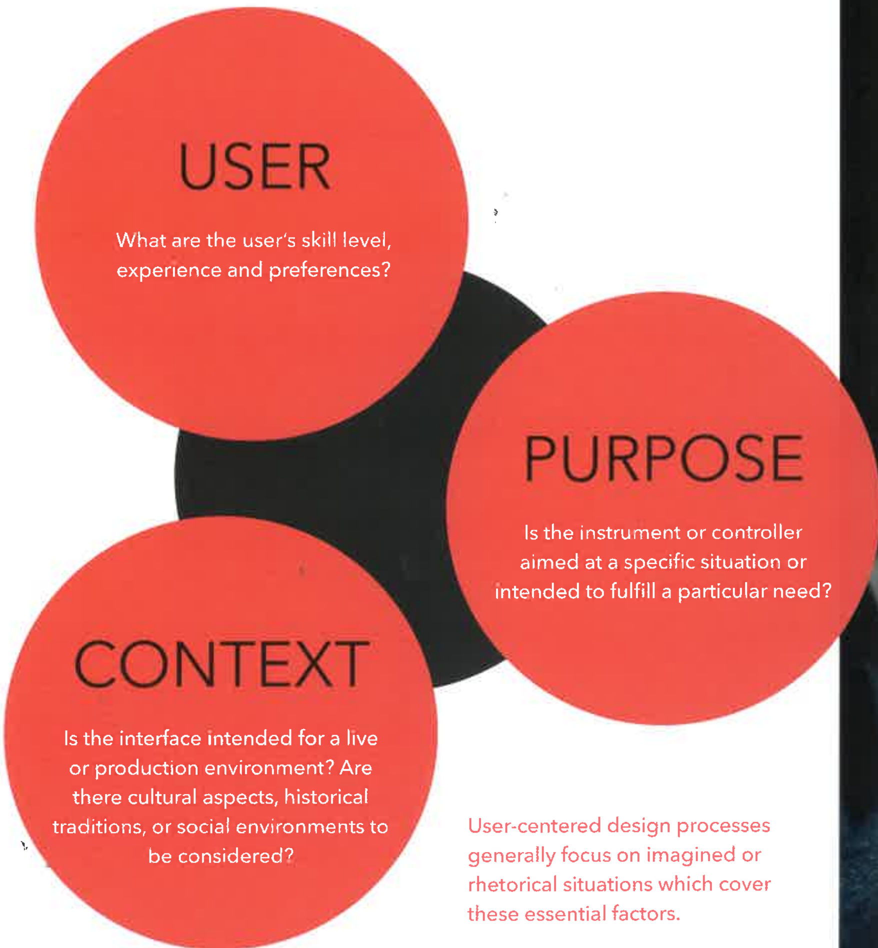
" Interaction is a very hot topic these days. Design is seen as not just something static, but something that becomes part of the observer. There is a communication going back and forth. Some of the communication we understand, some of it, frankly, we don't understand. For instance, why do you feel at one with your instrument? Why do you feel that your instrument is an extension of your hands?

Bob Moog "Moog – a documentary film by Hans Fjellestad", 2004

USER-CENTRIC DESIGN

From the earliest times, electronic music has blurred the lines between composer and technician, artist and creator. Often, an artist will become an instrument maker in order to see a new idea realized. Or an engineer will seek to actively collaborate with the producers, sound designers, DJs and musicians who will rely on the finished product(s). While it's true that many technological advances have been brought about by curious engineers or visionary creators working alone, a symbiotic relationship feeds both sides and cross-pollination drives innovation and, ultimately, superior results.

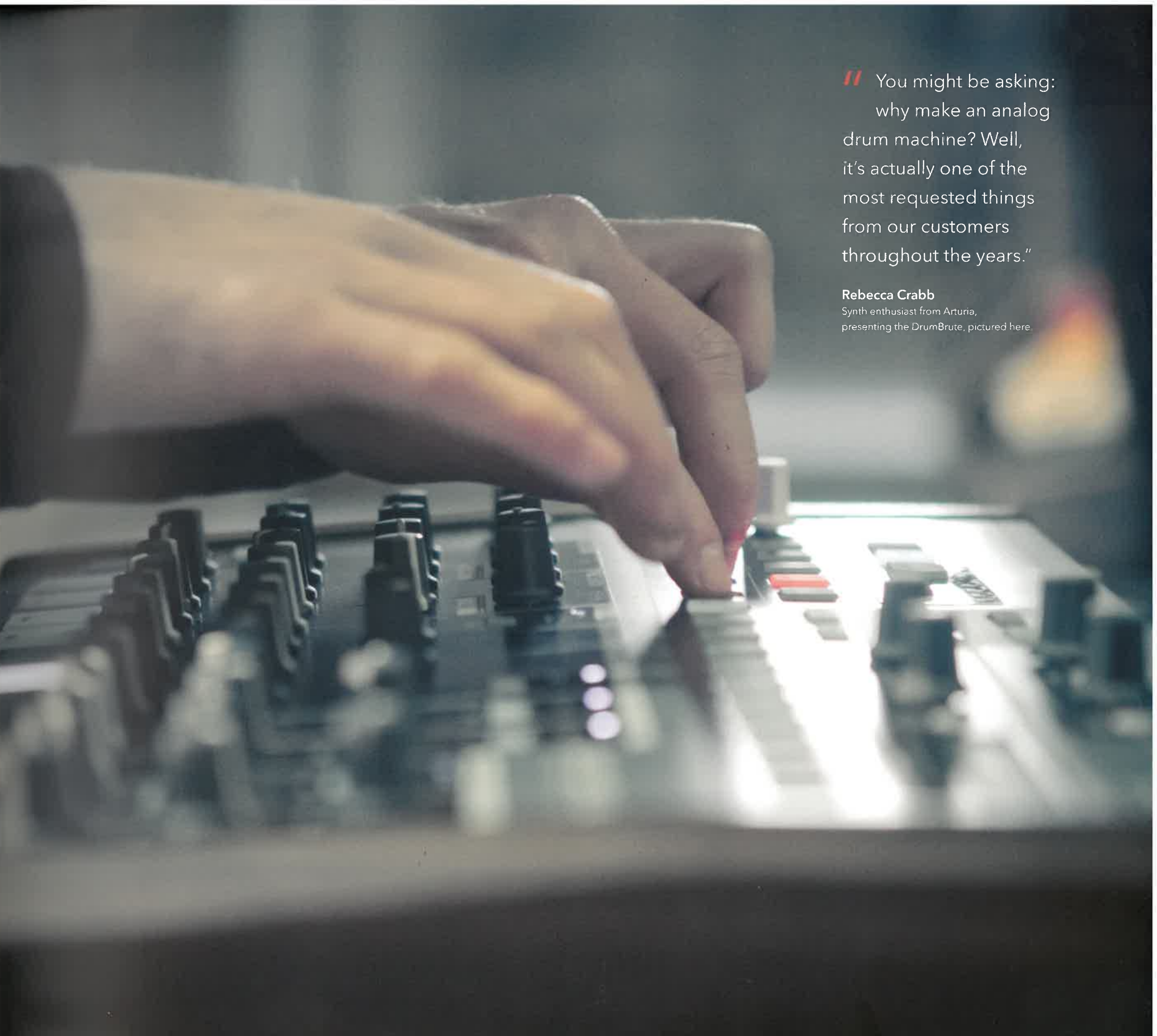
Of course this is nothing new; Bob Moog asked for – and responded to – his customers' feedback in the 1960s. These days, music technology companies use crowdsourcing, crowdfunding and early user engagement to ensure their products are perfectly on target. As such, there are multiple ways to keep the user firmly in the picture. This chapter offers a basic user-centric framework for interface design in electronic music.



User-centered design processes generally focus on imagined or rhetorical situations which cover these essential factors.

> // Artist feedback drove *all* my development work … The point is that I don't design stuff for myself. I'm a toolmaker. I design things that other people want to use.

Bob Moog · Interviewed in *Salon* by Frank Houston.



"You might be asking: why make an analog drum machine? Well, it's actually one of the most requested things from our customers throughout the years."

Rebecca Crabb
Synth enthusiast from Arturia, presenting the DrumBrute, pictured here.

CONTEXT

There are as many different contexts as there are artists and genres. Nevertheless, the designer of an instrument or controller must try to anticipate probable circumstances of use and fine-tune accordingly. Sometimes this is referred to as a 'musts vs. likes' exercise. Otherwise, the interface will either be generic and indifferent, or will lack elements deemed essential. We can identify four high-level contexts for an interface; several (or all) may be relevant to a single instrument.

Solo

Working alone, the artist is forced to become familiar with a range of different hardware and software. Typically, this involves learning many – sometimes conflicting – interfaces and modes of operation. In a performance situation, where there are are extra demands on concentration and heightened levels of stress, instruments with consistent or commonly-known functionalities and workflow are vital to success.

Collaborative

When working with one or more collaborators – say, in a band – any one artist can focus on fewer devices and give them more attention, but that attention must also be shared with the rest of the band for greater interaction.

Instruments designed with collaboration in mind often delineate and separate various music making tasks in a clear way. (See also p. 306).

DJ and producer Chris Liebing performing in a hectic club environment with Native Instruments controllers providing important feedback via color and light.

> As far as the customer is concerned, the interface is the product.
>
> **Jef Raskin** · Human Computer Expert working with the early Apple Macintosh

Studio

A studio setting typically offers more space and time than a live situation does. This allows for larger, more expensive, and/or more delicate machinery than one would take on the road, and more time to work with interfaces that are not optimal for live use (such as the mouse and computer keyboard).

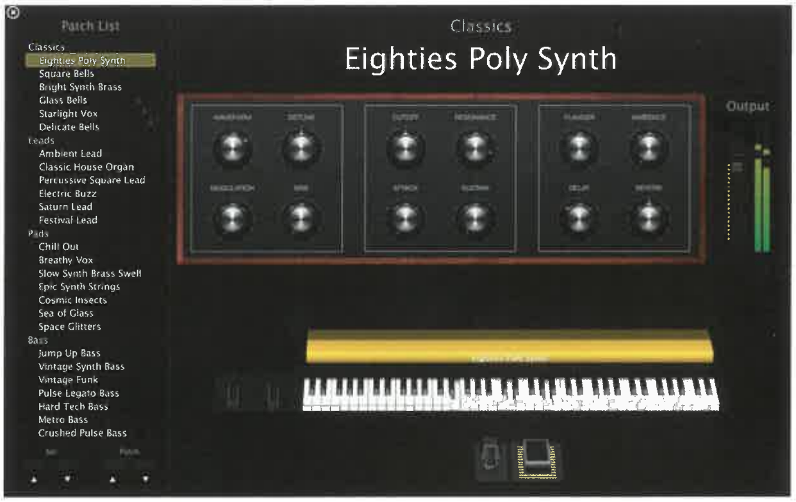
A large modular synth is more suited to life in a studio than out on the road, but that's not to say that modulars and other such gear aren't usable in a live context; anything is possible with practice! However, larger and more complex instruments are simply more comfortable to explore in the context of a relaxed studio setting.

Live

In a live performance, a well thought-out user interface is essential and the instrument must remain usable in a variety of lighting conditions. As an example, the screen of an iPad or keyboard workstation can be practically invisible in bright sunlight, whereas on a dark stage, buttons with no backlight or knobs without indicator lines can be frustrating to work with. It's also important that the device is rugged enough for the challenges of travel and potentially rough handling.

The interaction with the controller or instrument should be simple and direct – button combinations that require multiple presses or both hands are often less desirable in these situations. Shortcuts, custom programming, fast recall of presets and the like, are common techniques for an optimized workflow in the heat of live performance.

Ableton Live was the first Digital Audio Workstation (DAW) to bridge the gap between studio work and live performance. Being able to change between Session and Arrangement view provides a convenient and fast transformation from edited tracks to loops for live jamming and performance of the tracks. (See p. 50).



Apple's MainStage software is created and optimized for live performance. A select few and clear controls are available and configurable along with mapping and organization of the virtual instruments on the computer.

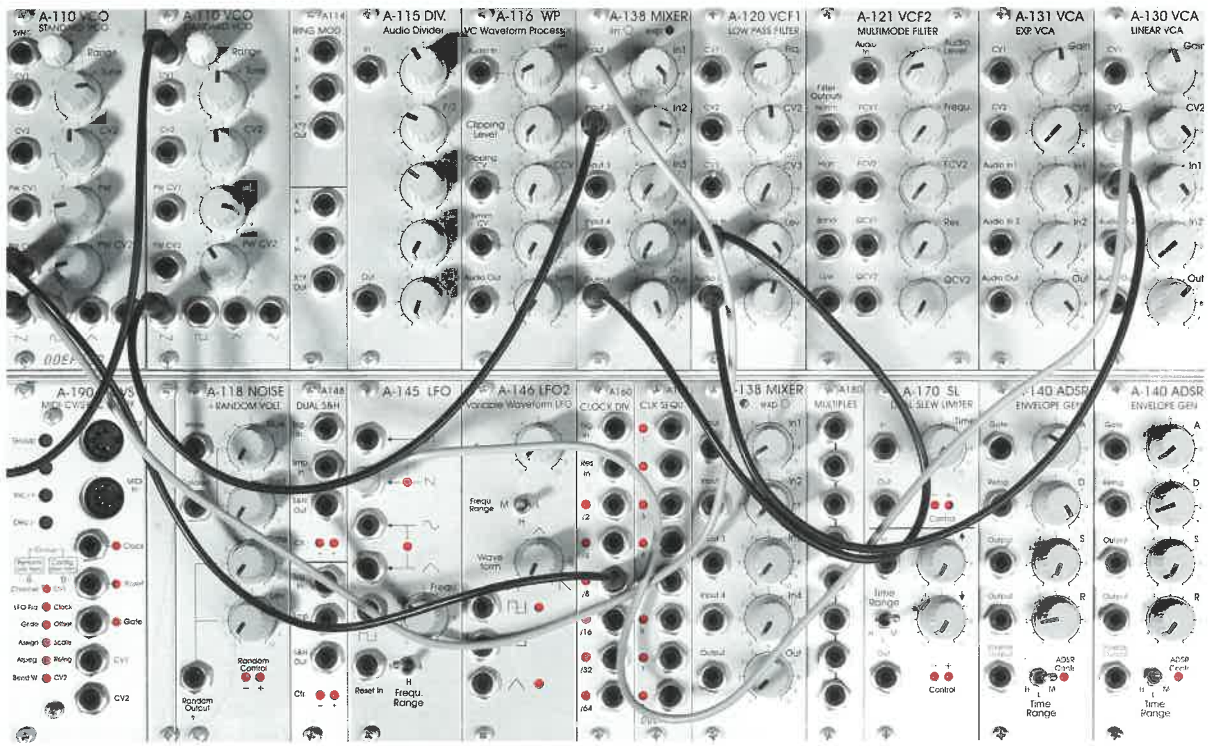
WORKFLOW & PREFERENCE

At the heart of many decisions is the balancing act between a product's desired functionality and ease of use. Interfaces have long exercised influence over workflow – but it's also true that users are forever discovering tricks and techniques not imagined by the designers.

Fixed/Targeted
It's perfectly valid to approach electronic instruments in a goal-oriented manner – i.e. with something specific already in mind. This might be a sound or a particular effect. In such cases, the interface needs to present the relevant options clearly. Otherwise, this places greater demands on the user, who might need to call upon experience or intuition to deduct the path to the necessary parameters.

Random/Exploratory
More often than not, random discoveries are made when attempting to fulfill an original intention. The interface may support random exploration, whether or not it was intended to do so. All artists, producers and the like have found new, surprising and often great-sounding results when exploring an electronic instrument. Indeed, it may have been this treasure trove of 'happy accidents' that led some to view the synthesizer with suspicion. Some interfaces even have buttons or built-in functions that encourage this random experience.

If an interface is powerful and flexible, it should help the user not only to reach expected goals but also to chance across the unexpected.



A modular system can provide the basis for a lot of experimentation and happy accidents, but may also provide a better understanding for a learner about the relationships between the building blocks of sound and control. On the other hand, some users might prefer a much simpler interface and fewer options.

This M-Audio CODE25 controller offers a variety of mappable control elements: wheels, pads, faders, knobs, buttons, and an XY grid. Each of these elements must be mapped deliberately, so random exploration must take place at a later stage in the sound design process.

> Power users don't decide very much based on aesthetics. Look and feel creates appeal, not loyalty.
>
> **Joel Marsh** · Author of 'The Composite Persuasion'

A user will prefer one instrument over another for reasons other than functionality or sound quality alone – these can be highly personal anyway. Preference is based on a combination of aesthetic, cultural, financial and emotional reasons. The same product can be vital to one user and unnecessary, even counterproductive, to another.

A user-centric design approach calls for attention to the users' preferences. These may include the following:

Cost

Reducing costs often results in wider availability. This can lead to devices with fewer buttons (and therefore more button combinations), no screens, cheaper materials, or even DIY assembly. However, a low-cost interface need not compromise usability.

(Build) Quality

To the user, knobs and faders with a bit of resistance can 'feel good' and thereby imply quality, as can a keyboard with weighted keys. In software interfaces, attention to graphic detail, legibility, and usability can convey a sense of quality.

Aesthetics

The visual appearance can be bold or balanced, vintage-looking or modern. This may attract some users while repelling others – regardless of sound or functionality.

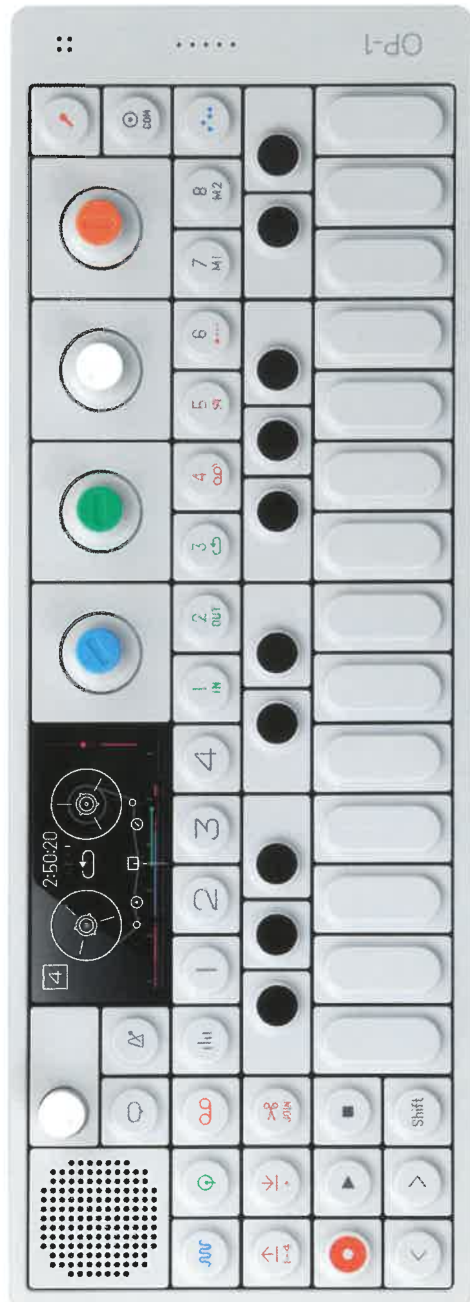
Functionality

If the instrument lacks features desirable to the user's purpose, high aesthetics and low price are of little relevance.

The Teenage Engineering OP-1 is small and may look like a toy, but it is heavy, suggesting durability and good build quality. It's not inexpensive but its high-resolution screen, logical interface and usability have won many friends.

The Korg volca sample provides sample manipulation and sequencing in a small and affordable package.

The Multimoog from 1978 was the big brother to the more famous Micromoog. It was meant to be a less expensive alternative to the iconic Minimoog. The Multimoog, however, was one of the first synthesizers to feature aftertouch. Today it holds a certain vintage and nostalgic value, not to mention its fascinating aesthetic appearance.



KNOWLEDGE & SKILL LEVEL

Ideally, a beginner-friendly interface should make it easy to understand what everything does and how it all connects together. The novice who obtains good results from clearly-indicated actions is encouraged to explore more deeply. The more advanced user also benefits from the quick and easy results of such an interface design. In general, beginners perform best with less control, while experts perform best with more control.

Novice

For an interface to be enjoyable for a beginner, obtaining immediate results is desirable but not essential. If possible, the interface should aid comprehension and make connections, states, feedback etc. easy to grasp. If, when starting to learn a new interface, the connections between action and result are clear, this helps foster deeper feelings of satisfaction and involvement. Beginners benefit from structured interactions, minimal choices, constraints and access to help.

Experienced

Experienced users typically wish to dive deeper, customizing and optimizing their interaction with the instrument. They benefit from less structure, and from direct access to functions with minimal use of constraints. This can involve anything from deleting all the preset sounds in order to create new ones from scratch, to programming an environment such as Max for Live, creating specific MIDI mappings or exploring parameters not immediately viewable. Some software and hardware implements 'advanced' tabs or 'expert modes' for just such a user. Even so, advanced features shouldn't be hard to locate, access or manage.

The Roland Space Echo effect had the manual printed inside the lid for easy access during operation. The interface is rather self-explanatory, though.

The Trope iPhone app, created by Brian Eno and Peter Shilvers, generates ambient music in response to colored shapes drawn on the screen. No prior musical knowledge is required, thanks to a clear and intuitive interface.

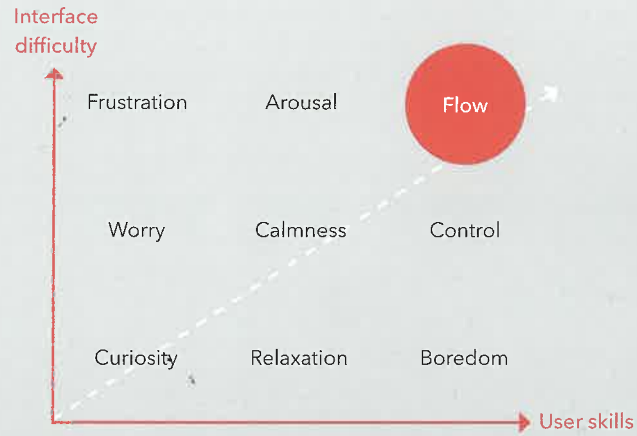
The original microKORG from 2002 is renowned for containing the same synthesis engine and features as the larger MS2000. The user can select sounds by genre using the large rotary switch, while five knobs on the upper right give immediate control over the most-used parameters. Experienced players can dig deeper using the editing matrix, which grants access to all parameters.

> Good design makes a product understandable. **Dieter Rams** · German industrial designer

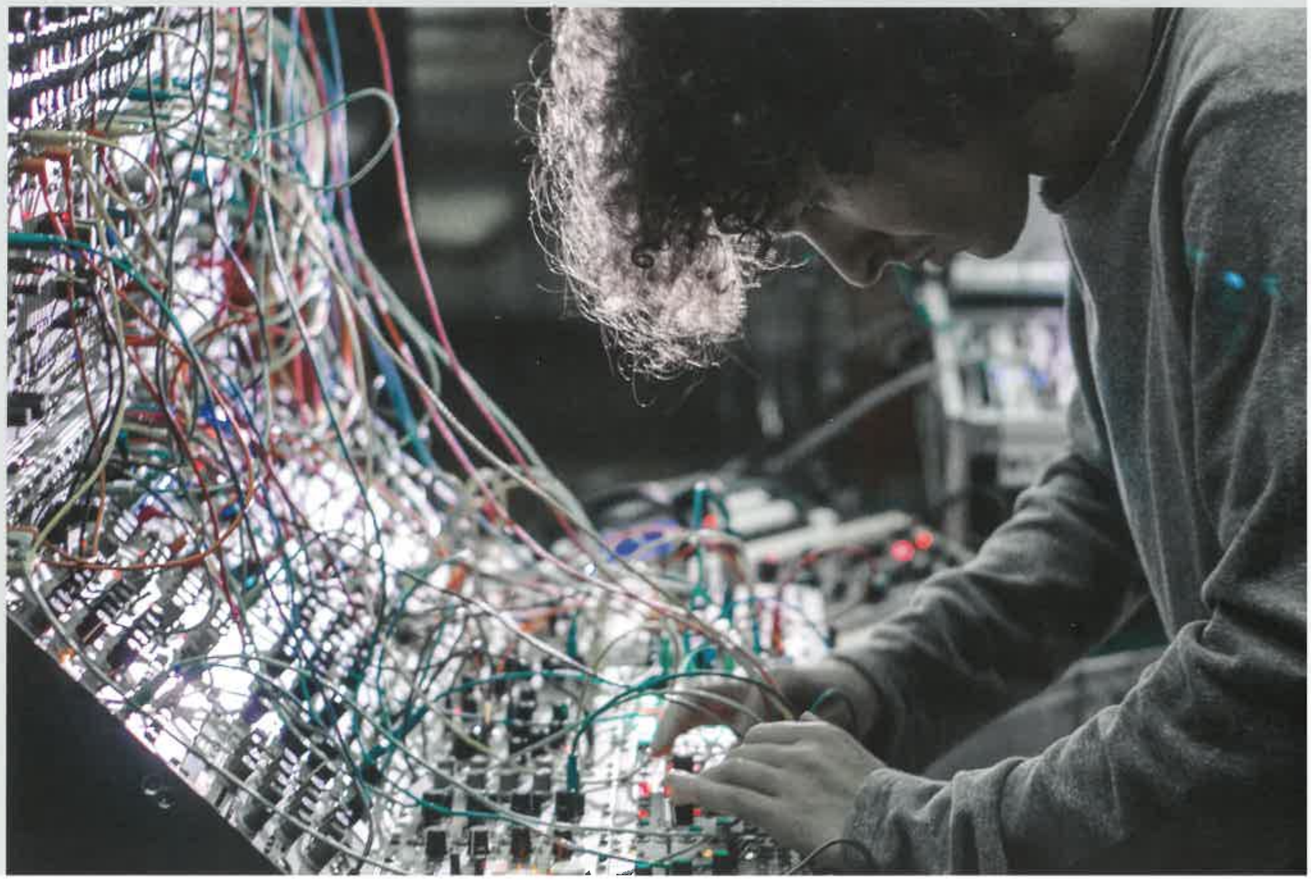
Flow is a term artists and musicians instinctively understand; it refers to a state where they are working near to their skill level. In other words, the interface is offering no interruptions to the process, and is supporting and/or challenging the user in a positive way.

Flow walks a fine line, rewarding the user after a certain amount of challenge. If an instrument isn't challenging at all – push a button, get a sound – boredom damages the flow. If it's too challenging in the wrong ways – set up fifteen submenus before anything can happen – worry and frustration kill the flow.

Studies show that the level of difficulty of an interface should match – or, ideally, slightly exceed – the skill level of the user[2]. If the dificulty is too far beyond the user's abilities, this is known as a steep learning curve. Manuals and tutorials seek to enhance user skills, but the interface should play its part by being self-explanatory as well, or at least by offering some rewards. The experience of flow turns an interface that is usable into one that is enyoyable.



The Dutch modular artist, Colin Benders, has been known for his immersive live sets performed on a complex Eurorack system. For a live performance to succeed with an interface so complex, an equally high level of skill is required. It is common for skilled artists to constantly seek new and more advanced possibilities to explore and express their art.



INTERACTION

What goes on between artist and instrument, producer and controller, or DJ and decks, is a complicated affair involving senses, brain processes, computing, electrical routing, physical movement and many other things. Let's have a closer look.

We input information to a system through the user interface using our hands, body, feet or voice. Even facial expressions, eye movements or brainwaves contain information that can be processed by computers and translated into data used in the system.

The system then processes the input and returns feedback to the user through the interface – whether it is as simple as a knob position, joystick movement, a lit button or text information on a screen. This feedback loop is the essence of every interaction.

User
Besides our brain using short- and long-term memory and coordinating the nervous system and muscular actions, it also interprets sensory information from our ears, eyes, hands, feet and even the whole body. We listen to the result, watch what is on a screen, observe how a knob is positioned or if a key is pressed. We not only input information but also receive feedback. Based on the feedback, new actions often follow; this creates a feedback loop.

Brain decides action for body to perform.

Hands, feet, body perform actions based upon decisicions in the brain (or upon stored, unconscious reactions).

Other parts of the human body can be used for input, e.g. using voice input or eye tracking.

Eyes receive visual feedback.

Ears receive audible feedback.

Hands, feet, body receive tangible feedback.

Interface
Depending on the type of hardware, the interface might provide tangible controls such as physical knobs, faders, keys, switches or the like. Touchscreens, by contrast, may offer many different input types such as text, gestures, button-presses, etc., through a simple interface. The interface receives input based on the user's actions and in turn presents information in the form of feedback.

Tangible input from pushing, turning, applying pressure or moving a control object (knob, fader, mouse, button, joystick, wheel, keys etc.)

Touch input from gestures on a touchscreen.

Gestural input from body movements.

Data input from mouse, computer keyboard etc.

Sensory input from meters, light, sound etc.

Visual feedback by graphical user interface, screens, LEDs, position of tangible controls.

Audible feedback by system or master output.

Tangible feedback by position of controls like buttons, knobs, faders, switches.

System
The underlying system, whether analog, digital, software or hardware, interprets the state of the interface and translates it into actions, computations, voltage or the like. Likewise, the result should be communicated back to the user via the state of the interface – i.e. reporting what has happened. Whether tangible, audible or visible, the interface must communicate the state of the system.

Feedback loop
The feedback loop is the result of the human-machine interaction. The human could opt to accept the input from a computer, an electrical sensor or, for instance, from a camera.

If the feedback is immediate (less than 0.1 second) the user has a feeling of directly creating the reaction. If the feedback loop lasts for more than one second, the user will notice that it is actually a system creating the feedback.[1] Immediate feedback is desirable for a functional, reliable or even usable or pleasurable experience.

FEEDBACK

> People ignore design that ignores people.
>
> **Frank Chimero** Author and designer

As we need feedback from our actions, it should be very clear, consistent and tactile if possible. If users receive no feedback or unclear feedback, they can't be certain of what the system is doing.

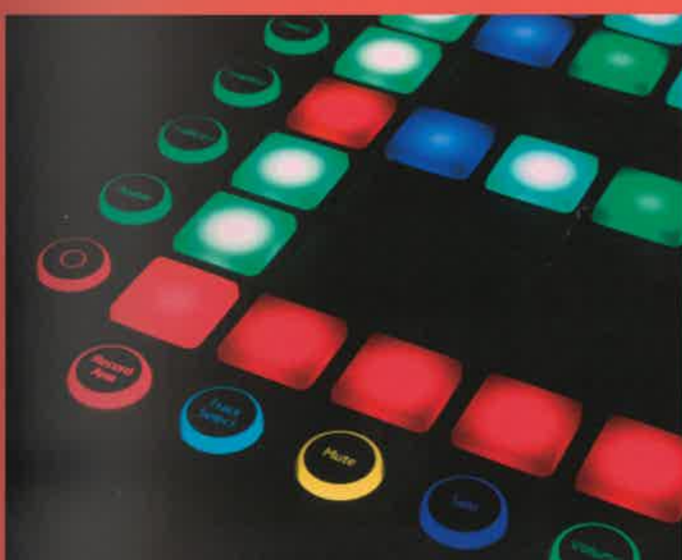
Feedback can be communicated in many ways, often by controls being in one of several different states. Some examples of feedback:

- Buttons pressed up or down
- Lit or unlit button/light
- Selected item marked with a color
- Blinking cursor waiting for input

The Tempi Eurorack module from Make Noise has six colored buttons which provide feedback to the user. Each button blinks at the tempo corresponding to that specific channel (1-6) and the color indicates which state the channel is in – e.g. muted.

The Native Instruments Maschine software makes use of strong color contrasts to provide feedback for selections and values.

The Launchpad Pro controller from Novation makes use of colors on the grid buttons to identify the clips, matching the colors to those of the software. These lights, together with the labels and colors of adjacent buttons, offer very clear feedback.



On hardware, like the Minimoog Model D, the position of knobs, faders, and switches provides immediate feedback.

HUMAN FACTOR LOADS

The behavior of humans towards a product, how they react psychologically and physically, is known as the human factor. We only have a certain capacity for processing information, retaining memories, making decisions and performing tasks, all at the same time.

The user experience is a result of the number of demands in three distinct areas of interaction. The mental or cognitive demands, which are about how much thinking, deciding, calculating or memorizing the user has to do. The visual demands, which are about how much visual sensory information the user is exposed to. The physical demands, which are about the amount, intensity, and detail of the physical activity required. The less the demand on the user's capacity in these three areas, the more user-friendly or enjoyable the experience.

The human factor load is dependent on the experience and skill level of the user. As an example, playing the piano with both hands is a heavy physical and cognitive load for some people but not for others. A lot of controls on a synthesizer can put a huge demand on the visual system, forcing the user to put demands on the cognitive system trying to remember their locations until muscle memory is established.

> "Limitations are not necessarily a bad thing in terms of interface, because if you spend less time thinking, you spend more time doing actual music.
>
> **Iftah Gabbai**, Skinnerbox

Visual

When an interface offers many visual indicators or physical controls, it increases the demand on the user's visual system. This is a dilemma. On the one hand, an abundance of visible controls make it easier to know the status of the whole system. On the other hand, too many visual elements to process make finding your way around complicated and time-consuming – especially if they are not well organized. In the studio this is less of a problem but on stage it can be crucial.

Visuals, however, can ease the load on other areas. For example, it is not necessary to remember what the controls are for if icons or text labels are visible.

Mental/Cognitive

If users have to struggle to understand what is going on, remember what some controls do, or where they can find a menu item, this puts a load on the brain. The mental load can exist in many forms but is usually high when device usability is low (due to ergonomic or other concerns).

Examples of high mental/cognitive load are: menu-surfing, seeking feedback or visible cues, or remembering button combinations in order to perform a task. In live performance situations, it's important that the mental load on the user is as low as possible. Practice (and the attendant training in muscle memory) lowers the mental/cognitive load.

Physical

Physical demands on the user generally involve the hands and/or feet. As stated above, some people find playing piano with two hands a heavy physical and mental load (albeit one that lessens with practice).

Physical demands can include stretching hands to reach button combinations or having to use both hands to navigate menus. Controls that are too small or fiddly, screens that are hard to reach, small buttons or controls that are hard to hit with a mouse click, or interfaces that require scrolling to navigate – all can create unnecessary physical demands.

Easier access

Some synthesizers from the digital era (the 1980s-1990s in particular) required many button presses to edit parameters and thus placed a heavy physical load on the user. By utilizing small screens, these instruments were limited in the feedback they could provide.

The PG-800 programmer from Roland was developed to give easy access to the editing features on the Roland Super JX-10 and MKS-70. The faders provide visual feedback and ease of physical operation, which minimizes the mental and physical load for the user.

It is not uncommon to see software editors for instruments with limited interfaces, easing the operation for the user. They are usually welcomed by those who otherwise were frustrated due to the heavy load on the mental/cognitive and physical systems.

Minimizing and optimizing load

In this example from the Arturia iMini iPad app, the user enters this modal view for easier editing after tapping on a knob. The background is dimmed, which minimizes the visual load and supports focus, and the text instructions minimize the mental load; we don't have to wonder what is possible, but can read a helpful text. In this modal view, the user can slide the finger anywhere on the screen to adjust the value, which makes the physical interaction much easier and not as sensitive to mistakes as in the normal view. In this way, the physical load has been lowered as well.

Load balancing and distribution

On the Ableton Push 2 controller, the load is distributed among the three types: Mental load is low because text, colors, and the screen give clear visual information. The visual load is minimized by placing the control elements in a grid and by keeping all buttons the same black color as the background. Finally, the size and placement of pads, the feel of the raised buttons, and the lowered contour of the touch strip make physical navigation easy and very tactile.

TASK

If we start by focusing on the specific tasks a user is expecting to carry out with an instrument, there are 3 quantitative goals to be fulfilled for a successful operation. At a minimum, it should work (functional). It's better if it works all the time (reliable). Better still, it should engender confidence and a sense of connection in the user (usable), and thus becomes a working part of the user's world. When tasks can be performed in a functional, reliable and usable manner, this creates the basis for a positive qualitative experience.

When evaluating how well an interface performs, it might be beneficial to align each goal with the '7 Stages Of Action' laid out in Donald A. Norman's book 'The Design Of Everyday Things'.

The 7 stages of action for the user are:
1. Forming a goal
2. Forming the intention
3. Specifying an action
4. Executing the action
5. Perceiving the state of the world
6. Interpreting the state of the world
7. Evaluating the outcome

An interface has to be designed to support and foresee these 7 stages with the result being functional, reliable and usable. Examples: What is the user's intention in each situation? How does the user know which actions are possible? How does color help identify status? What controls are natural for the execution of each action?

OBJECTIVE/QUANTITATIVE INTERFACE GOALS

Functional
When the unit is switched on or the software app is opened, we expect everything to work. If the app crashes or the unit is unresponsive, the experience cannot be worse. If we succeed in opening a piece of software, we next expect everything in there to be functional: buttons do what they are expected to, sound comes out, etc.

Upgrading of operating systems and platforms can require the user to update the software frequently to stay operational/functional. Hardware synthesizers, mixers, DJ controllers, etc., often contain hardware parts that are designed to be replaced when worn out (e.g. the heavily-used crossfader on a DJ mixer) or can sometimes be upgraded via firmware updates.

Reliable
Reliability is not only expected from software systems but also refers to durability and build quality in hardware. One of the worst-case scenarios during a live performance is that gear is not reliable and breaks down, freezes, etc. On the other hand, a lot of vintage analog gear could be termed unreliable when oscillators go out of tune or flaky circuits result in unintended 'grit' – often mistakenly attributed to a 'warm analog sound'. But in general, there is no other feature artists and producers value more than reliability. Some manufacturers, though, are better than others in providing upgrades and thus ensuring greater reliability, either when issues arise or software environments change.

Usable
It is simply not enough that a feature or control exists and works. For the user to have patience with the product, it must be easy to use on a consistent basis. Usability is obtainable via a fast and clear feedback loop, and an interface designed and evaluated according to these nine principles. These are also explained on the coming pages:
- Ergonomics
- Tolerance
- Consistency
- Clarity
- Discoverability
- Structure
- Simplicity
- Affordance
- Control

FOCUS:

INTERACTION + FEEDBACK + HUMAN FACTOR LOADS + WORKFLOW + KNOWLEDGE & SKILLS

EXPERIENCE

> // Designing a product is designing a relationship.
>
> **Steve Rogers** UX Director, Google

Experience derives from the execution of tasks followed by the evaluation of results. The experience we have using an instrument or controller is based on our perception, interpretation, and evaluation of the tasks performed.

The first step in a qualitative experience is that of enjoyment. We enjoy processes that work out as expected and are easy to figure out. Furthermore, we begin to form attachments to instruments that produce pleasant surprises and spark creativity and self-expression.

If all prior states are fulfilled to a certain degree, the ultimate experience of trust and loyalty is obtained – this is when artists form deep relationships with their instruments, and maybe even attribute human traits like soul, character, or mood to them.

The user experience evolves over time and is closely related to the topics previously discussed, such as solo vs. collaborative use, live vs. studio work, and the balance of skill and difficulty that determines flow.

SUBJECTIVE / QUALITIVE INTERFACE GOALS

Enjoyable
When the objective goals have been met to a certain degree, users are more likely to enjoy playing an instrument. The pleasure we want the user to achieve by interacting with the instrument and the community built up around it can be explored through these four areas:
Physical – When the user is hearing, seeing and feeling the instrument.
Ideological – The user's values, preferences, and beliefs in relation to the brand and instrument and whether it's cool, new or coveted, etc.
Psychological – The user being in a flow state, discovering new patterns, learning, being satisfied with the cost-benefit balance, etc.
Social – The common interests and activities of the user, their social status, sharing knowledge about the instrument, etc.

Meaningful & Creative
One of the most important qualitative goals for interface design is to make the interface meaningful – a foundation where the user can express creativity unhindered. Meaningful processes and creative results are the basis for further exploration and for emotional engagement.

A musical instrument can be designed in a creative and unusual way to inspire interaction, or it can be designed in a conventional and unobtrusive way to aid the user in achieving their goal. The most successful designs are those combining these two approaches by building on, expanding or even challenging conventional interfaces.

Trust & Loyalty
If the previous objective and subjective goals have been reached, the ultimate goal for an instrument maker is that users perceive their instrument as a loyal companion.

An instrument or a controller sparks a trusting relationship if it is reliable, has a user-friendly interface, creates joy and meaning, and lets us engage in creative processes and reach personal goals.

This happens over time and can therefore contain an element of nostalgia, since older instruments carry more of an air of history about them, and are more likely to be thought of in a familiar way by the user.

FOCUS:

CONTEXT + WORKFLOW + PREFERENCE + KNOWLEDGE & SKILLS

USABLE

Usability refers to the extent that users can employ a product fit for a specific purpose and to which degree they are able to reach their designated goals efficiently and satisfactorily.

Considering these nine principles of usability[3] when designing interfaces (whether for hardware or software) has a significant impact on user satisfaction.

When laying out the controls and interface elements, decisions about design principles such as contrast, gestalt laws, composition, color, etc. should support these usability goals (see p. 124).

Ergonomics

Easy to use physically. Does not challenge or exceed motor skills or sensory abilities.

Examples

Button combinations should be obtainable with no more than two fingers/hands. Controls are appropriately scaled and with sufficient space for fingers to push, turn or move them without error.

Tolerance

Reasonable interpretations of entries, clicks and button presses. Minor mistakes are ignored.

Examples

A drop-down menu should take into account a certain degree of inaccuracy, so it does not disappear if the user moves the mouse, say, just one pixel outside the designated area.

Consistency

Consistent, recognizable presentation and interaction. Matches user expectations.

Examples

Highlighted buttons are shown in the same color. The same action performs the same function. For example, double- or long-pressing a button in an interface always opens extra menus or advanced functionalities.

Clarity

Comprehensible and precise communication. Clear context and distinction between elements.

Examples

Naming functionalities according to conventions, for instance, 'LFO' or 'ADSR'. Visualizing parameters like waveforms or placing legible text labels (legends) close to controls.

Discoverability

Elements are visible, not hidden. Designed and placed to be easily found.

Examples

Designing a 'one-knob-per-function' interface with a 1:1 mapping of control elements to functions. Hiding the minimum of functionality under drop-down menus or shift operations.

Structure

Categorization and grouping of elements according to users' needs and mental models.

Examples

Designing the interface according to the signal flow or workflow. For example, VCO>VCF>VCA or a patch browsing process where the user first selects a genre, then mood, and lastly sound.

Simplicity

As simple as possible. Not overwhelming. No unnecessary steps.

Examples

Giving the user immediate access to often-used controls, like the filter cutoff on a synthesizer. Providing shortcuts to parameters and not overloading the interface with extraneous features.

Affordance

Usage is self-evident. Controls suggest how they are used.

Examples

It may not be obvious that a touch strip is configured to play notes or that a knob also has a push action. A fader implies by its visual appearance that a parameter value can be changed continuously.

Control

The user is in control, not the system. Appropriate pace. Actions can be reverted.

Examples

Being able to undo drastic actions like deleting a bank of sounds. The system should keep dialogs on screen long enough for the user to react. Using nondestructive audio editing or being able to cancel operations like freezing a track.

> Design is not just what it looks like and feels like. Design is how it works.
>
> **Steve Jobs**

The drum machine iOS app DM1 from Fingerlab has nine big and easily-struck (affordance) pads for finger drumming on a simple screen (simplicity). This makes it easier for the user to tap away without much worry (ergonomics).

Tabs provide access to different sections of the instrument according to use scenario (structure), and the recording button is the traditional red (consistency and affordance).

The Novation Bass Station 2 has clear and precise descriptions of controls on the front panel (clarity). In the Oscillator section, the sub oscillator (Sub Osc) is tucked in as part of the bigger group (structure) but is still its own entity. The switches clearly show that they control 'either-or' parameters (affordance).

Searching and selecting sounds in the software instrument Omnisphere 2 from Spectrasonics is a guided experience allowing the user to filter down or search across genres and banks (discoverability, structure, and control).

ENJOYABLE

When analyzing the elements of an enjoyable interactive experience with an electronic instrument, we can benefit from the framework outlined below[4].

Good interface design should take these four areas into account and seek to balance them according to the intended purpose of the product.

Ideological
Ideological pleasure is related to values and beliefs. It is about what one thinks is right and wrong, good and bad. In the world of sound synthesis, discussions about analog vs. digital are not uncommon.

Interface-wise, it often boils down to a user's preference for particular control elements, such as whether a fader or knob is best suited to adjustment of the filter cutoff or the stages of an ADSR envelope.

The designer of an instrument can attempt to satisfy preconceptions, e.g. by adding wood panels to a synthesizer to give the impression of being vintage or analog.

Psychological
The brain rewards itself when it recognizes patterns or learns new skills, resulting in psychological pleasure. This is a powerful motivational factor. Scientific research reveals that the brain enjoys surprises – a fact that is very applicable in an interface context.

Initiating positive surprises by, for example, facilitating the creation of randomly-generated yet useful sounds, can greatly increase the perceived value of a device or piece of software.

Psychological enjoyment can also come from a feeling that the instrument was a good price or a worthwhile investment.

Physical
Physical enjoyment comes from the stimulation of our senses, which can be:

Touch: Do we like the way a knob feels when we turn it, the resistance of pushing a fader, the satisfying 'click' of pushing a button or the feel of an instrument when we slide our hand across the surface or press down the key?

Sight: Does the interface look appealing due to its choice of colors, illustrations, typography, etc.?

Sound: Arguably a vital parameter: do we like the sound we get from the instrument, whether hardware or software?

Social
Social pleasure comes from social interaction created when people meet friends and new acquaintances.

Social enjoyment also comes from belonging to groups and strengthening or improving one's social position. This may involve use of customized controllers or possession of rare or expensive instruments.

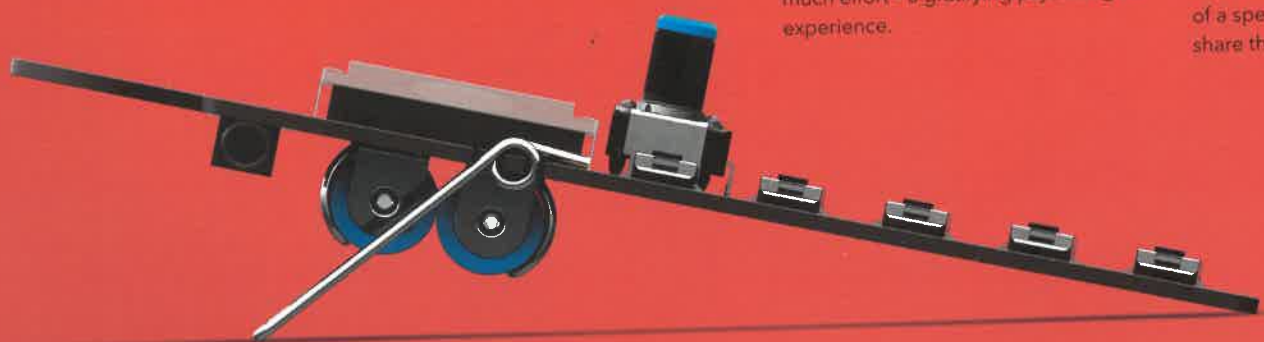
Pleasure may also derive from interaction with others with similar interests, either online or at meetings such as 'modular meets' or EMS Synthi lovers' get-togethers.

Sharing experiences or knowledge when collaborating, teaching or performing also creates levels of social pleasure.

> // Design must seduce, shape, and more importantly, evoke an emotional response.
>
> April Greiman

In the Teenage Engineering Pocket Operators, users find joy in a small portable device with a simple interface and quirky sounds. Pushing effect buttons provides instant audio effects without much effort – a gratifying psychological experience.

The instruments have a pleasing aesthetic resemblance to retro gaming devices, and the exposure of their inner electronics creates a unique physical and visual experience. Socially the devices create attention and the inclusion of a speaker means bystanders can share the music too.



// Like many things in life I do not fully understand it. For me in some ways that's a joy, because essentially you can input and output in any way you wish. Sometimes it works and sometimes it doesn't. To me there's a real joy in the unexpected. **Scanner** (Robin Rimbaud) about his use of modular synthesis · 'Scanning Through The Waves', horizontalpitch.com

Audio Damage's Axon drum synthesizer plug-in has an unusual way of sequencing the voices, its hexagonal structure providing pleasant surprises to the user.

The tactile and visual aesthetics of shapes and materials, like the red aluminium and glowing red rubber wheels of this Moog Little Phatty, create an important enjoyable user experience.

The immediacy of the Korg monotribe and monotron analog synthesizers provides a pleasing visual

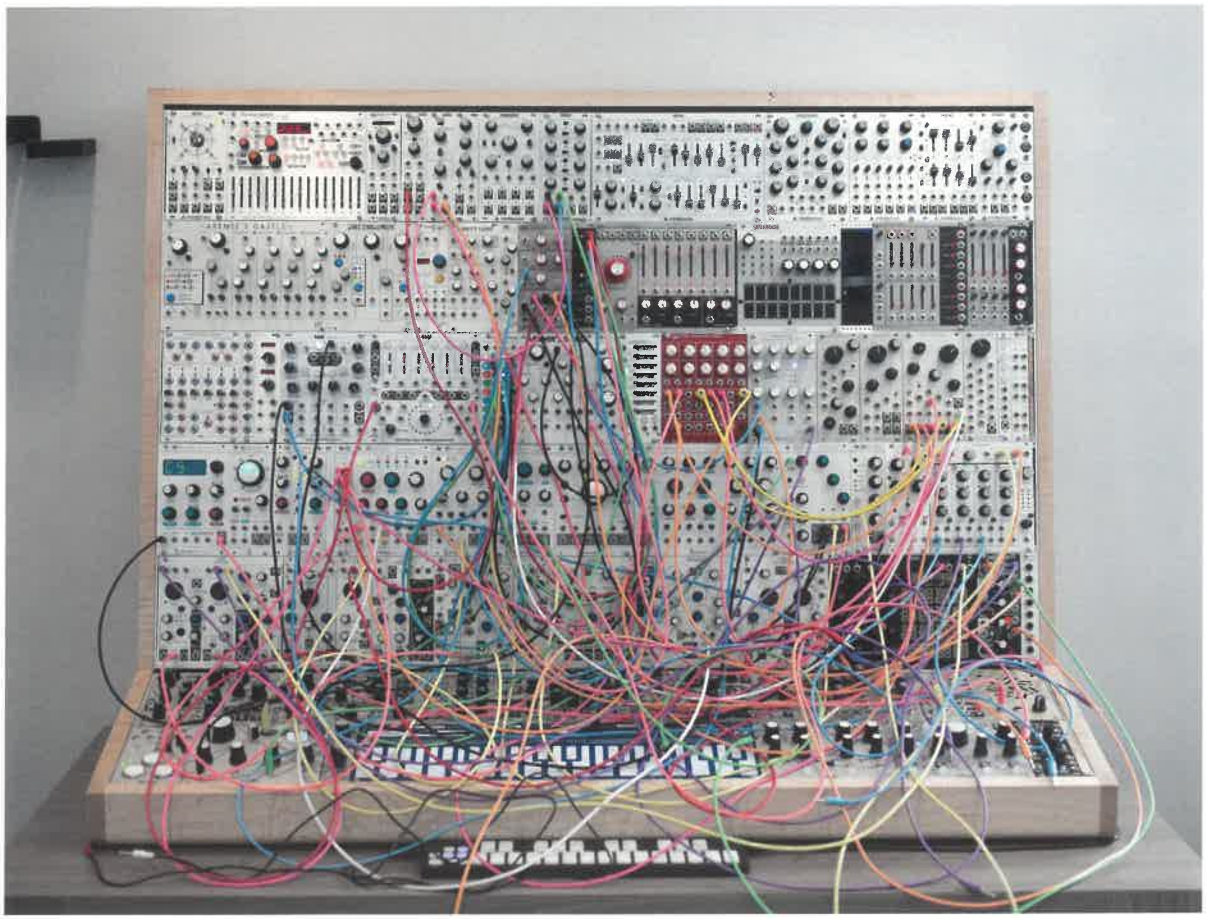
MEANINGFUL & CREATIVE

When we enjoy playing an instrument the sole act of doing so becomes meaningful. Meaning can also be part of the results or the process of using the instrument.

When users discover new ground, expanding their artistic and creative capabilities, it heightens their qualitative experience. It can happen for a beginner at the piano or with sophisticated software for the experienced user – or vice versa. The experience of meaning and creative flow is highly individual and differs vastly from person to person. Two different artists can have very different experiences with the same instrument – it is a matter of, among other things, experience, skill level, personality, expectations, preferences, and context. Research shows (not surprisingly) that people feel most spontaneously alive when they participate in activities they have freely chosen and are keenly interested in. An interface can support this sense of meaning, and spark creativity by not forcing the user into actions that may seem irrelevant to the current task or goal.

User-centric interface design asks the questions: What is meaningful to the user in this context? How do we instill creative possibilities for different skill levels? What are the opportunities with this interface: are they too few, too many, or just right? Should the design be bold and creative, or should it 'step into the background' to motivate creativity and meaning?

Although designers seek to answer these questions within the design process, users will always find new ways of using intruments that makers never thought of. This creates new meaning and opportunities for everyone.



Some users may find a large modular system, like the one pictured here, meaningful to their workflow and creative goals.

The Thingamagoops (opposite page) react to light; that particular feature might be highly meaningful and spark creative interest with some users, while others may find it uninteresting. The two-edged sword for manufacturers is catering to the mass market on one side, potentially causing an instrument to be generic and boring, while on the other side, a unique approach to an interface risks misunderstanding and commercial failure.

> I have always believed that you should use instruments that feel like an extension of your personality.
>
> **JunkieXL** (Tom Holkenborg)

Though Roland marketed the TB-303 toward guitarists and keyboard players as a bass accompaniment machine, it found its way into (and became the defining sound of) genres such as Acid, Techno and Trance. Creative users delighted in removing and inserting the batteries, which resulted in random patterns. Building on this, Roland implemented random pattern-generation functions in their successors, the AIRA TB-3 and the TB-03 (see also p. 198).

TRUST & LOYALTY

To many artists, there's no distinction between electronic and acoustic instruments when it comes to forming deep relationships.

Whether software or hardware, instruments are a tool for self-expression and for achieving certain results. If this can be done without major functional issues, a sense of loyalty to and from the instrument is likely to become apparent.

It is a natural tendency of human psychology to attribute human traits, emotions, and intentions to things.[5] The phenomenon is called 'anthropomorphism' and usually occurs if all the prior states of the user experience have been fulfilled to a satisfactory level. Creative and meaningful results don't necessarily provide a basis for trust or loyalty, though. That only happens if performance and results are consistent.

Other possible factors that increase the likelihood of anthropomorphism, trust, and loyalty are:

- Nostalgia in sound, look or features
- Humor in look, text or graphics
- Customizing done by the user
- Uniqueness or rarity
- Size ('small' = 'cute')
- Organic materials
- Branding/storytelling

> // I know the TR-909 so well that it's really considered a good friend. I'm aware of a lot of other hardware, but the TR-909 has helped me so many times that I can always depend on it.
>
> **Jeff Mills** in 'R is for Roland'

The iconic Roland TR-909 drum machine from 1983 found successful use in genres like Techno and House music. The 16-step sequencer is especially useful as it can chain patterns into full songs.

By giving the Chronoblob Eurorack delay module an unusual color scheme and a unique name, Alright Devices aimed to make it a keepsake for modular synth users. The illustrations and small figures build a sense of anthropomorphism by use of branding and storytelling.

Besides a possible enjoyable aesthetic experience visually and sonically, the prospect of emotional attachment to a module like this is greater (for some) than to a more generic-looking module.

The EMS Synthi A from Electronic Music Systems has, since its debut in 1971, been a favorite of many pioneering electronic music artists. The Synthi A was small for its time. It is essentially a portable version of the VCS3, and was Jean-Michel Jarre's first synthesizer, described by him as 'an old friend'.

> // I would say that the Synthi A could be called *my favorite* because he was always present and never let me down. Note that I call him *he* and not *it*.
>
> Klaus Schulze