Handcrafted Ale

for beer and computer

Eric Honour

Technical Requirements/Instrumentation

Performing *Handcrafted Ale* requires the following:

1. One bottle or can of high-quality beer. Preferably, use homebrew, brewed by yourself or a friend. If that is not available, use something produced by a craft brewer. Do not perform the work with "macroswill" (e.g. Budweiser, Miller, Coors, Heineken, or other, similar brews). The size of the bottle or can is up to the performer, but only one bottle or can should be used for this version of this work.
2. A glass for the beer. Sounds produced by the glass form a major part of the soundscape for this work, so it is important to choose a glass that makes sounds you like. Ideally, you should be able to make a variety of sounds with the glass—including the ability to elicit a clear pitch, both when empty and filled with beer—via striking it, rubbing the rim, or other means. Generally, a glass with thin walls works best.
3. A bottle opener (if using a bottle).
4. Three microphones total. The work requires two contact microphones. It is relatively easy and inexpensive to build contact microphones, if you do not already have them available. Contact the composer for instructions or search online. Ideally, the piece uses a third, lavaliere microphone. A wireless, omnidirectional lavaliere mic works best. However, it is certainly possible to perform the work with a wired lavaliere and/or stand-mounted mic, and other pickup patterns will also work.
5. An Intel Mac computer running Mac OS 10.6 or later, with Max (or the freely available Max 6 Runtime) and the free SoundMagic Spectral suite of plug-ins installed. The work requires Max 6.1.3 or later and SoundMagic Spectral 6 or later. Max is available from Cycling '74 (www.cycling74.com). The SoundMagic Spectral plug-ins are available from Michael Norris (www.michaelnorris.info/software.html). At present, it is not possible to perform this work on Windows computers, sorry.
6. The *Handcrafted Ale* software ("HandcraftedAleDistro.mxf"), which should be included in the archive containing this document. If you need to download it again, it is available at www.erichonour.com. This file is a Max 6 collective. You must open it with either Max 6 or Max 6 Runtime.
7. An audio interface providing stereo output (note: the work can also be performed in quad or octophonic versions; contact me for details) and appropriate inputs and mic preamps to accommodate the microphones you are using. Please see the setup instructions below: the software expects the microphones to appear on specific inputs.
8. An iPad running the TouchOSC app or a MIDI controller capable of providing simultaneous, independent, tactile control over at least 28 continuous controller numbers and generating on/off messages for 8 notes. A Keith McMillen QuNeo is strongly recommended, although it is possible to make the piece work with other controllers. For QuNeo users, a QuNeo preset is included with the software download package. The controller mappings are detailed below. For iPad/TouchOSC users, instructions are provided below.
9. For the optional 4- or 8-channel versions of the work, users may incorporate an iPhone running TouchOSC and outputting accelerometer data to control panning and spatial dispersion of the granular outputs. This feature is only available in the 4-/8-channel version of the work: it is not available in the standard stereo version of the work. It generally works best to attach the iPhone to the bottom of the

glass, although other arrangements are possible and acceptable. Note that at present, the patch *will not* work with two different iDevices running TouchOSC. Thus, it is not currently possible to use an iPad to control the work while using the accelerometer data from an iPhone to control panning.

Setup Instructions

1. If you are using a lavaliere microphone (recommended), affix it to the bottle, as close to the opening of the bottle as possible. This mic is primarily used to capture the sounds of the bottle/can being opened, beer being poured from the bottle/can into the glass, and breath/lip sounds generated near the opening of the bottle/can. An omnidirectional mic is recommended, to avoid proximity effect and provide some capture of other sounds in the performance space. If you are using a stand-mounted mic, position it so that you can obtain as much of the sounds described above as possible. This microphone should be connected to input 1 of the audio interface.
2. Affix one contact microphone to the beer bottle/can you are using. One or two thin strips of gaffer's tape works well for this. Experiment with locations during rehearsals, but a spot halfway up the side of the bottle/can usually works well. Connect this microphone to input 2 of the audio interface.
3. Affix one contact microphone to the glass. Careful experimentation during rehearsal is needed to determine the best location and mounting method for this microphone. It is important to make sure that the microphone mount does not interfere with the ability to obtain pitches and other sounds from the glass. Placing the mic in the wrong spot, or using too much tape, can easily mute the glass or make it sound too dull. Connect this microphone to input 3 of the audio interface.
4. Position your equipment so that you can easily work the MIDI controller/iPad with one hand while interacting with the bottle and/or glass with the other. Place the computer screen in a location where you can see it easily while performing, as it provides details about the current settings for the various processes occurring in the software.
5. Adjust the gain of your mic preamps so that the levels on each microphone peak no higher than -6 dBfs.
6. If you are using an iPad running the *TouchOSC* app to control this work, you must set up that app before moving on. Please see the **iPad & TouchOSC Setup** section at the end of this document for instructions. Otherwise, move directly to step 7 below.
7. Launch the *Handcrafted Ale* software. At the right-hand side of the window, you will find a "Setup" section, on a yellow background. The first step in Setup is to open the Audio Status window and configure it properly for your hardware. Everyone's hardware is different, but for most users, you will at least need to select your audio interface from the menus at the top of the window. Note that this software is CPU intensive. If you find that it overtaxes your computer as you rehearse the work, try increasing the size of the IO buffer and signal vector.
8. Next, load the plug-ins. For each of the plug-ins, click the button and navigate to the spot on your computer where they are installed. If you followed Michael Norris's instructions, you should have the plug-ins installed in this path:

    ~/Library/Audio/Plugins/Components

The three plug-ins are MNSpectralDronemaker, MNSpectralPartialGlide, and MNSpectralShimmer.

9. After loading the plug-ins, click the reinitialize button. Then turn DSP on. Ensure that levels from the microphones are showing on the input meters and that controls on the QuNeo (or other controller) are connecting properly to the parameters in the software.
10. When testing is complete, reinitialize the software, using the "Reinitialize" button again. This will reset all the controls, clear the buffers on the effects processors, and turn DSP off.
11. When you are ready to begin the performance, turn DSP back on and raise the output levels to full, then begin.

Details of Computer Processing

This work uses the live inputs from three microphones and real-time computer processing to create the electroacoustic output. The actual output comprises eight separate audio streams. One of these is mono and the other seven are stereo. Four of the audio streams are derived from the microphones on the bottle. The other four are derived from the microphone on the glass.

The performer is responsible for mixing these output streams to create the actual, stereo output of the work heard by the audience.* The performer is also responsible for manipulating various parameters of the processors to influence the content of the various audio streams, and for generating the audio initially, via the bottle/can and glass.

The following audio streams are derived from the bottle:

- Dry: the unaffected, dry input from the lavaliere mic. This is the only bottle-related stream using a single mic; the others are all fed a mix of the lavaliere and contact mic inputs.
- Drone: output from the SoundMagic DroneMaker plug-in
- Munge: output from a granular synthesis processor
- Delay: output from a stereo, single-tap delay, with feedback. Note: this processor is fed from the output of the granular processor for the bottle, NOT from the dry input.

These streams are derived from the glass:

- Shimmer: output from the SoundMagic Spectral Shimmer plug-in
- Drone: output from the SoundMagic Partial Glide plug-in
- Munge: output from a granular synthesis processor
- Delay: output from a stereo, single-tap delay, with feedback. Note: this processor is fed from the output of the granular processor for the glass, NOT from the dry input.

---

* Note: in the 4- and 8-channel versions of this work, the performer is still responsible for mixing these output streams together, and is also responsible for controlling panning/spatial dispersion.

† For the QuNeo pads, "X" means that the parameter is mapped to continuous control over the X (horizontal) dimension of the pad. "Y" means that is mapped to the Y (vertical) dimension of the pad.

Performing the Work

In essence, a performance of this work is entirely built around the performer drinking a beer—using a glass—and manipulating the various sounds that result via the computer. In observing the various performance requirements in this document, this essential view of the work should inform all your decisions. The audience should clearly see that you open, pour, and drink a bottle or can of good beer, and that the actions and rituals involved form the basic structure of the work.

The duration of the work is variable, depending on the size of the bottle/can the performer selects and the speed at which he or she drinks. That said, performances of this work should last between 6 and 10 minutes.

Volume levels within the hall are at the performer's discretion, but should be neither deafening nor quiet. The audience should be able to hear some of the natural sounds of the performance, as well as the processed sounds. These natural sounds should be picked up by the lavaliere mic, and are available within the software on the "Dry Mic Output," in case they need to be reinforced in the hall. However, the overall balance should be somewhat in favor of the processed sounds.

Dynamics within the work are at the performer's discretion, except that the piece should begin quietly, the performer should explore a reasonably wide dynamic range over the course of the work, and the performer should achieve several high and low points during the work, with regard to dynamics and intensity.

There is no graphic score for this work. Aside from those listed in this document, there are no specific requirements for the performer. You should spend plenty of time in rehearsal exploring the various processing methods available in the software, and then improvise freely with those to create the work in performance. It is expected that performers will make use of the majority of the processing options available in any performance of the work, but you are not required to use all of them.

Similarly, the performer should experiment extensively ahead of time with the sounds available from the beer, bottle/can, glass, acts of drinking (e.g. opening the bottle, smelling the beer, sipping, pouring, swirling), and other performer actions (e.g. striking the glass with fingers/fingernails, rattling the bottle opener against the bottle, tilting the glass when it is partially full to obtain *glissandi*, blowing into/across the bottle/can). Again, performers are not required to use all possible sounds in any performance, but you are encouraged to plan some of the sounds you intend to use ahead of time.

A performance should begin with the bottle/can, glass, and performer in clear view of the audience. Other equipment may or may not be in view. The bottle/can should be closed, so that the act of opening it is part of the sound world of the piece (note: opening the bottle/can does not have to be the first sound the audience hears).

The inspiration for this work came from playing with a glass of beer and noting that the pitch of the glass changed markedly, depending on the amount of beer it held. The pitch of the glass is highest when it is empty. It gets lower and lower as more beer is added, although most of the difference occurs after the first quarter of the glass is filled, for most glasses. This phenomenon is at the heart of this work, and is one of the things that provides an overall form and structure for the piece:

You must clearly elicit the pitch of the glass before you pour beer into it. Make extended use of this pitch—via playing the glass and/or feeding this pitch into the computer for processing—to cement it in the ears of the audience, before pouring beer into the glass.

The process of performing the work must involve you eventually pouring some or all of the beer into the glass. This could occur in one or more steps. You must elicit another pitch from the glass when it is as full as you intend it to be, to set the lower pitch boundary for the work. You should elicit a number of other pitches from the glass, when it is full of different amounts of beer.

The process of completing the work must involve you eventually drinking all of the beer that you have poured into the glass. (It is not strictly required that you drink all the beer from the bottle/can.) When the glass is empty again, you must elicit the empty pitch once more, to close the work.

Aside from the details given above and in the rest of this document, the details of performance are up to the performer.

Suggestions and Cautions

- Note: be careful with your pour. Beer foam dulls the sounds of the glass quite a bit. A haphazard pour may fill the glass with foam, which could take quite a while to subside, with deleterious effects on both the performance and your enjoyment of the beer.
- Beers with very thick beer foam may present an additional challenge, in that the foam may lace the inside of the glass, even when it subsides, which will again dull the sound of the glass. It is best to experiment with the specific brew you intend to use ahead of time, so that you know its characteristics.
- It is very easy to get carried away with the various sounds you can make via the bottle/can, glass, opener, and software, and this can easily lead to very extended durations for the work. Make sure to keep an eye on the clock throughout the work. It is generally best to strive for several distinct sections, using sounds from before the beer is opened, after opening but before it is poured, after pouring but before drinking, during drinking, and after drinking. The first and last of these are usually shorter than the others, but it works well for the middle sections to be roughly balanced in duration. This is not a requirement, but only a suggestion.
- It is also very easy to get carried away and knock over the bottle/can or glass. This is particularly a problem when the performer uses one hand on the MIDI controller/iPad and the other to make sound. Be careful!
- Large glasses, with thin walls, usually provide excellent sounds for this work. However, if using a very large glass, make sure to use a large enough amount of beer to be able to fill it at least half full, to obtain a good range of pitches. For very large glasses, you may need to use an oversize bottle or can of beer.

Controller Mappings

This table shows the various parameters the performer can control in the work and how they are mapped in MIDI. Note: if you use a QuNeo, you can use the "HandcraftedAle.quneopreset" file in the distribution folder to quickly and easily set up your QuNeo for performance. The fourth column below provides details on which QuNeo pads are mapped to each parameter in that preset. See the QuNeo diagram below the table for the pad names. If you will use an iPad running TouchOSC to control the work, please the **iPad & TouchOSC Setup** section below for instructions.

Parameter	Available Range (or Steps)	MIDI CC/Note Pitch	QuNeo Pad
DSP On/Off	0, 1	Note 11	Play
Reinitialize	N/A	Note 9	Diamond
Dry Mic Output Level (dBfs)	-inf – 0.	CC #1	HSlider 1
Main Output Level (dBfs)	-inf – 0.	CC #5	LSlider
Glass Drone Interpolation Length (seconds)	1., 2., 3., 4., 6., 8.	Notes 7 & 8	L/R 1
Glass Drone FFT Size (samples)	512, 1024, 2048, 4096, 8192	Notes 5 & 6	L/R 2
Glass Drone Level (dBfs)	-inf – 0.	CC #2	HSlider 2
Glass Munger Delay Length (ms)	100. – 15000.	CC #11	Pad 13 (X)[†]
Glass Munger Recording On/Off[‡]	0, 1	Note 3	L/R 3 (Left)
Glass Munger Grain Separation (ms)	5. – 150.	CC #12	Pad 14 X
Glass Munger Grain Size (ms)	5. – 300.	CC #13	Pad 15 X
Glass Munger Pitch (decimal multiplier, relative to 1. = unison)[§]	0.5 – 2.	CC #14	Pad 16 X
Glass Munger Pitch Variation (± around the current pitch value)	0. – 0.2[**]	CC #15	Pad 16 Y
Glass Munger Level (dBfs)	-inf – 0.	CC #3	HSlider 3

[†] For the QuNeo pads, "X" means that the parameter is mapped to continuous control over the X (horizontal) dimension of the pad. "Y" means that is mapped to the Y (vertical) dimension of the pad. Pressure data from the QuNeo pads is not used for this piece.

[‡] Each of the mungers features an input buffer of 15,000 ms. If recording is turned on, the input signal is immediately fed out of the buffer for processing and the buffer contains the most recent 15,000 ms that have been input (updated continuously). If recording is turned off, the current status of the buffer is frozen. Processing will continue to occur, but on whatever 15,000 ms are in the frozen buffer, with no updating. If recording is turned back on, the buffer will begin to fill again. Note that the buffer is NOT cleared when recording is turned back on, but instead is overwritten in real time. Thus, previously frozen material can persist in the buffer until it is overwritten.

[§] The pitch value is slewed, to smooth out the instantaneous control data and avoid large, sudden leaps in pitch. All pitch changes drift back to unison over 10 seconds. The performer may send more pitch changes while this drift is occurring and the system will respond as usual. After the last change the performer sends, however, the system will begin to drift back to unison. It is not possible to turn this behavior off.

[**] All of the MIDI CC #s output a range of 0 – 127 by default. In many cases, this range is curved in various ways within this work. Most of these are invisible to the user and simply adapt the "feel" of the controller. In the case of pitch variation, however, it is important to understand that inputs of 0 – 64 will result in a pitch variation of 0. This is intended to make it very easy to return to 0 and hard to depart it (thus protecting the performer from unintended pitch variations). Input values of 64 – 96 produce a range of 0. – 0.05 for pitch variation. Input values of 96 – 127 produce a range of 0.05 – 0.2.

Glass Delay Level (relative to total Munger + Delay output)[††]	0. – 0.5	CC #21	Pad 9 X
Glass Delay Feedback Level (%)	0. – 80.	CC #22	Pad 10 X
Glass Delay Time L (ms) (controls both L and R)	100. – 1000.	CC #23	Pad 11 X
Glass Delay Time R (ms) (controls only R)	100. – 1000.	CC #24	Pad 11 X
Bottle Shimmer FFT Size (samples)	512, 1024, 2048, 4096, 8192, 16384	Notes 12 & 13	U/D 1
Bottle Shimmer Feedback Level (%)	0. – 99.	CC #7	VSlider 1
Bottle Shimmer Output Level (dBfs)	-inf – 0.	CC #8	VSlider 2
Bottle Glide FFT Size (samples)	512, 1024, 2048, 4096, 8192, 16384	Notes 14 & 15	U/D 2
Bottle Glide Rate (Pitch Shift in cents per Second)	-5 – -600	CC #9	VSlider 3
Bottle Glide Output Level (dBfs)	-inf – 0	CC #10	VSlider 4
Bottle Munger Delay Length (ms)	100. – 15000.	CC #16	Pad 5 (X)
Bottle Munger Recording On/Off	0, 1	Note 1	L/R 4 (Left)
Bottle Munger Grain Separation (ms)	5. – 150.	CC #17	Pad 6 X
Bottle Munger Grain Size (ms)	5. – 300.	CC #18	Pad 7 X
Bottle Munger Pitch (decimal multiplier, relative to 1. = unison)	0.5 – 2.	CC #19	Pad 8 X
Bottle Munger Pitch Variation (± around the current pitch value)	0. – 0.2	CC #20	Pad 8 Y
Bottle Munger Level (dBfs)	-inf – 0.	CC #4	HSlider 4
Bottle Delay Level (relative to total Munger + Delay output)	0. – 0.5	CC #25	Pad 1 X
Bottle Delay Feedback Level (%)	0. – 80.	CC #26	Pad 2 X
Bottle Delay Time L (ms) (controls both L and R)	100. – 1000.	CC #27	Pad 3 X
Bottle Delay Time R (ms) (controls only R)	100. – 1000.	CC #28	Pad 4 X

iPad & TouchOSC Setup

Rob Hexler's *TouchOSC* app (available on Apple's App Store) provides an excellent and inexpensive means for controlling works like this one. If you have access to an iPad, you can purchase the TouchOSC app ($4.99 at the time of this writing) and use that to control *Handcrafted Ale*. You can also freely create your own control layouts for anything that responds to MIDI and/or OSC. The app works on the iPhone as well, but note that the layout for controlling *Handcrafted Ale* will not work on the iPhone. The same goes for Android devices—TouchOSC exists for these, but the layout included with this work is intended for iPad use. I have not tested it on Android

[††] The munger output is fed through the delay. The actual output heard is the sum of munger + delay. The delay level parameter controls the balance between the munger and delay signals. As the delay is increased, the level of the munger signal is decreased. When the delay level = 0., only the munger signal is heard. When the delay level = 0.5, the two signals are evenly balanced.

devices, but I do not think it will work properly. Also remember that *TouchOSC* control, at present, is only available for the stereo version of the work—the 4-/8-channel version does incorporate *TouchOSC*, but uses it control spatial dispersion/panning.

Once you have the *TouchOSC* app installed on the iPad, setting things up is straightforward. Load the TouchOSC layout for *Handcrafted Ale* onto your iPad. The layout is included in the distribution ZIP archive of this work ("HCAleLayout.touchosc"). Follow the instructions available on the TouchOSC website (hexler.net) for loading the layout. You can accomplish this either via iTunes sync or by syncing layouts using the free TouchOSC Editor app, available on the TouchOSC website. You only need to load the layout once—it is unnecessary to load it every time you perform the work (you do need to make sure you have that layout selected in the app every time).

For each performance, then, these are the steps involved in setting up the iPad and Max. With practice, negotiating these steps takes less than 60 seconds.

1. Make sure the TouchOSC app is not currently running on your iPad. In recent iOS versions, double-click the Home button to bring up all the apps that are currently running. If TouchOSC appears in the list, throw it upwards to quit the app. Quitting the app ensures that it will launch with up-to-date IP address information.
2. Create an ad hoc WiFi network on your computer: go to the WiFi icon in the top menu bar, and select "Create Network…" from the pull-down menu. Name it however you like, and apply a password if you like. Once the network is active, find the IP address of the computer.
    a. To find the IP address of your computer, go to the Apple menu: System Preferences..., then click Network. In the top center of the window, look for the "Status" area. In this area, it should tell you the IP address of the computer.
3. On the iPad, go to Settings: WiFi, and join the network you created on your computer. If you receive a warning that this network is not connected to the Internet, click "Join Anyway."
    a. Note that when your iPad goes to sleep, it will most likely abandon this connection and return to an Internet-capable WiFi connection. When you wake the device up (i.e. to perform the work), you will need to check again that the IP addresses are set properly before you begin performing.
4. Now launch the TouchOSC app. If you currently see controls in the app, touch the white dot in the top right corner to access the settings.
    a. **Important:** each time you create an ad hoc network on your computer, it creates a random IP address. For that reason, you should expect to go through these steps every time you perform the work.
    b. In the TouchOSC app, under "Connections," click OSC.
        i. Make sure OSC is enabled
        ii. Set the Host IP address to match the IP address of your computer.
        iii. Port (outgoing) should be set to 8000 and Port (incoming) should be set to 9000.
        iv. Take note of your local IP address; you will need to enter it in the *Handcrafted Ale* patch for it to operate properly.

v. In the top left corner, click “TouchOSC” to return to the main menu.
vi. In the “Layout” section, verify that “HCAleLayout” is selected; if not, click there and select it.
vii. Then click “Done” at the top right of the main menu to enter the HCAleLayout.

5. On your computer, launch the *HandcraftedAleDistro.mxf* file, either in Max Runtime or in Max.
    a. In addition to the other setup instructions listed earlier in this document, you must set up the IP address of your iPad. In the top right corner of the *HandcraftedAleDistro* window on your computer, look for an object named “p HCAleTouchOSCStuff.” Double-click that object to open the set up window.
    b. In the set up window, type the IP address of your iPad into the corresponding field (delete the other contents first). Then click the button above the field. Note: double-check that you have entered the IP address correctly—if any numbers are incorrect or out of order, the patch will not work properly!
    c. Close the HCAleTouchOSCStuff window. In the main window, in the Setup Area at the right-hand side, click the “reinitialize” button. If TouchOSC is properly set up, you should see all the controls on the iPad jump to their initial values. You are now ready to start the piece.