

Reverb Foundry Unveils Tai Chi



Software company Reverb Foundry announces their new flagship reverb, TAI CHI. Combining state of the art DSP with a unique and creative set of audio crafting tools and techniques, Tai Chi redefines the sound design standards of reverb technology while integrating unprecedented control over fidelity and dynamics.

At the heart of Tai Chi is a Feedback Delay Network engine that takes the art to an entirely new dimension creating an incredibly thick, rich and luscious chorus unlike any other reverb on the market.

What really sets Tai Chi apart from the pack are a highly imaginative and useful set of onboard sound design options that are fully integrated into the reverb itself. These include dynamics processors, multi-band filters and fidelity tools. The onboard compressor and ducker can be set to control the late reverb while the preserving the integrity of critical early reflections. Up to four bands of multi-band reverb decay time allow surgical sonic sculpturing above and beyond the usual reverb contouring controls, unlocking the ability to take the spectral shape of the decay into all sorts of interesting places never before heard in a contemporary

Reverb Foundry Unveils Tai Chi

Montag, 22. November 2021 16:54

reverb design. The bit-crushing and density controls deliver the grit and dirt that gave vintage reverbs their character and flavor, while providing the option to apply signal degradation to different elements such as the reflections or the tail, or apply the master EQ's low / high roll-off and shelving filters for the perfect finish.

Tai Chi Lite boasts the same lusciously chorused sound engine as Tai Chi, but in exchange for a more affordable sticker price drops the surround, dynamics, multi-band and some of the fidelity tools.

Tai Chi supports channel counts up to 7.1.2. Tai Chi Lite is a stereo reverb processor.

Tai Chi is available now for \$199 in AU, VST 2, VST3 and AAX formats on Mac and PC. Tai Chi Lite is also available now for \$99.

www.reverbfoundry.com