

# OscillaScore: A Modular Platform for Graphic Notation in Networked Music Performance

Dr Rob Canning

Independent Researcher  
rc@kiben.net

## ABSTRACT

This paper introduces *OscillaScore* (<https://robcanning.github.io/oscilla>), a browser-based framework for authoring and performing cue-driven graphic scores using SVG, WebSockets, and OSC. Developed as an open-source project, *OscillaScore* enables synchronised, media-rich performance environments in which animated notation and time-based cues drive ensemble coordination across networked devices. It supports a hybrid approach to score representation, accommodating both fixed-form and open-form works, while allowing composers to embed interactive logic directly into visual elements via a declarative mini-notation system.

*OscillaScore*'s architectural principles, cue syntax, and animation logic—based on XML ID-embedded transformations—are described through the lens of a foundational case study: the 2025 premiere of *Six Inches to the Mile*, a structured improvisation commissioned by the Stuttgart-based ensemble Pony Says. Developed and rehearsed remotely using *OscillaScore*, the work served as both the compositional impetus and foundational case study for the system. The score's design included animated graphical elements, embedded performance annotations, and indeterminate behaviors implemented via lightweight browser-side logic.

Positioned at the intersection of composition, performance, and interactive digital notation, *OscillaScore* supports both fixed and improvised practices through a modular, browser-native platform for cue-driven graphic scores. By embedding animation and cue behaviors directly into visual elements, it enables composers to shape dynamic performance environments without relying on proprietary tools or external scripting. The system's accessibility and adaptability make it suitable for a wide range of contemporary music practices, from solo improvisation to networked ensemble work.

## 1. INTRODUCTION

Digital score systems have expanded the possibilities for musical notation, enabling dynamic visual elements, real-time interaction, and distributed performance. These tools are particularly valuable in contexts where fixed notation

is insufficient—whether due to improvisatory structures, multimedia integration, or the need for synchronised but flexible ensemble coordination. However, many systems remain either overly complex or tightly coupled to specific software environments, limiting accessibility and adaptability.

Recent systems such as INScore [9], the Decibel ScorePlayer [10], and John, the Semi-Conductor [11] have each offered distinctive models for programmable or collaborative scoring environments. *OscillaScore* builds on this lineage while responding to a need for lightweight deployment, browser-native accessibility, and modular animation and cue logic. It also engages with broader aesthetic considerations around screen-based notation and performer interaction as explored by Hope and Vickery [12].

*OscillaScore* is a lightweight, browser-based framework for creating and performing cue-driven animated scores using open web technologies. Developed by the author to support both fixed and open-form composition, the system enables composers to embed animation logic, cue triggers, and interpretive annotations directly within SVG files. Playback is coordinated via a central server and synchronised across clients, with optional Open Sound Control (OSC) integration for interfacing with external media systems.

This paper introduces *OscillaScore* through a technical and artistic account of its development and first use in the composition *1:10,560 (Six Inches to the Mile)*. The discussion situates the system within a improvisational practice—balancing structure with interpretive freedom—and reflects on the interplay between software development, score design, and performance. A concluding section explores the potential of *OscillaScore* as a community-driven platform for graphic notation and collaborative experimentation.

## 2. SYSTEM OVERVIEW

*OscillaScore* is a browser-based framework for real-time graphic notation, designed to support structured and open-form compositions in ensemble contexts. It enables composers to author scores as animated SVG documents, embedding cue logic and playback behavior directly



example, scaling sequences can be used to suggest phrasing intensity or entrance timing, while path-following animations can indicate the trajectory of a glissando or dynamic swell. In one composition, a rotating arrow was used to randomly point toward different musical fragments arranged in a circle, signaling which material a performer should interpret next. These mappings and their interpretive implications are explored further in Section 3.

Beyond single-parameter effects, animations in OscillaScore can be layered and nested to create composite gestures. For example, a single object might simultaneously rotate, scale, and follow a path, creating complex temporal trajectories or emergent visual rhythms. These combinations allow composers to shape the interpretive space of a score over time, supporting forms of notation that are not merely illustrative but structurally generative. While animations currently follow fixed timing or randomized behavior, future extensions may explore performer-interactive or data-driven animation control, including integration with live coding environments for real-time manipulation.

## 2.2 Synchronisation and Playback

OscillaScore is designed to support distributed performance, where multiple performers interact with the same score via networked devices. The playback environment consists of a **Node.js server** (handling timing, playback state, and OSC output), **browser-based clients** (typically tablets or laptops, displaying and animating the SVG score), and a **local or remote file server** hosting the SVG and media assets. A detailed architectural overview is provided in Appendix A.

The score scrolls horizontally at a steady rate unless paused or interrupted by a cue. Manual rehearsal navigation is supported via **jump2rehearsal** cues, allowing performers to skip to specific points in the timeline. The system is designed to handle reconnects gracefully and maintain synchronised timing across devices. Screen resolution differences are managed with responsive layout techniques, although currently uniform device dimensions (e.g. iPads) are preferred in performance contexts.

The player interface includes user-facing controls consistent with common media software conventions. These include play, pause, rewind, jump-to-start, fullscreen toggle, and navigation via rehearsal marks, along with toggle buttons for annotation visibility, OSC output, and WebSocket connectivity. A local audio mute button is provided for rehearsal flexibility. An additional “sync pause” toggle allows users to temporarily suppress network updates and desynchronise their playback view — useful for personal score preview or navigation without affecting others. When re-engaged, the client automatically resynchronises to the current global position via a server message. The system is fully

distributed, with no designated master client, allowing any connected device to operate independently if needed.

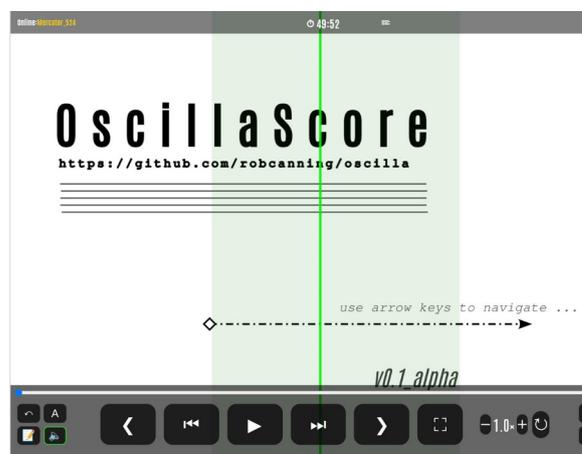


Figure 2: Default OscillaScore help file showing the playback interface, including transport controls and the animated playhead. This built-in guide introduces key concepts such as cue triggering, timeline navigation, and annotation visibility.

In networked performances, a single client can be designated as the audio output device, disabling audio playback on all others to avoid redundancy and conserve system resources. This client has the ability to mute or stop audio cues as needed during rehearsal or live performance.

A real-time stopwatch is displayed prominently (and can be toggled fullscreen), and connected clients are listed for ensemble coordination. While designed for complex animated scores, the system can also operate in a minimal “light mode” — functioning as a synchronised stopwatch with occasional cue displays. This makes OscillaScore a flexible tool for a wide range of musicians, including improvisers working with time-based forms. While iPads or other high-end GPU-accelerated tablets are preferred for complex scores due to their reliable browser performance and rendering capabilities, OscillaScore also runs effectively on low-cost Android tablets and recycled laptops for simpler use cases. This flexibility supports accessible deployment in educational, community, and non-institutional contexts.

## 3. CASE STUDY: 1:10,560 (SIX INCHES TO THE MILE)

*1:10,560 (Six Inches to the Mile)* is a networked digital score for electric guitar, synthesiser, and electronic drums. Commissioned by the Stuttgart-based trio Pony Says with support from the Arts Council of Ireland / An Chomhairle Ealaíon, the work premiered at the 2025 Music Current Festival in Dublin, presented in

cooperation with Dublin Sound Lab (<https://musiccurrent.ie/2025/pony-says-concert.html>).

The composition explores the metaphor of mapping as both a creative and critical framework. Drawing inspiration from historical Irish Ordnance Survey maps, the score blends cartographic visual language with contemporary digital performance practices. Thematically, the work reflects on how maps structure our perception of space, movement, memory, and identity—both in the physical world and in the navigational language of the score itself.

The score was authored entirely in SVG using Inkscape and scripted media elements, with additional LilyPond-generated notational fragments converted to vector paths. Screenshots of online map details—particularly historical Ordnance Survey imagery—were easily imported and converted to vector paths in Inkscape, allowing the composer to incorporate real cartographic material directly into the graphic score. This fluid workflow enabled rapid integration of visual elements from diverse sources, tightly aligning the score's appearance with its conceptual themes. It was presented on synchronised iPads using the OscillaScore framework, with a local server managing playback state, cue triggering, and animation coordination. Performers engaged with a time-scrolling graphic score featuring animated symbols, path-following indicators, and dynamic visual cues. Object-to-path animations were used to represent glissandi, dynamics, and gestural flows, while rotating and scaling objects suggested energy, directionality, or interpretive density.

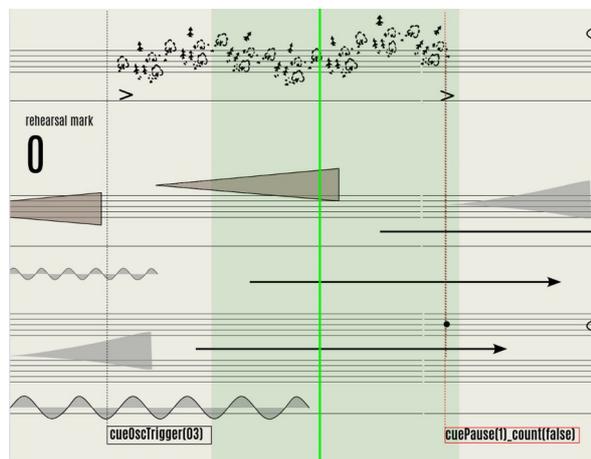


Figure 3: Excerpt from *Six Inches to the Mile* showing elements of the OscillaScore system in use. Visible are `cueOscTrigger(03)`, used to trigger parameter changes in optional live electronics, and `cuePause(1)_count(false)`, which introduces a short pause without displaying a countdown overlay. The rehearsal mark “O” forms part of the automated navigation system, while animated arrow symbols indicate interpretive direction and timing—though their motion is not visible in this still image.

Structurally, the work is a fixed-form composition with embedded pockets of controlled indeterminacy. Some sequences were rendered stochastically using client-side randomization logic, resulting in non-synchronous fragment playback between devices. Cue-based triggers such as **cuePause(180)** invited improvisation within defined zones, while visual countdowns provided temporal clarity. During rehearsal, performers could tap to bypass paused states using the embedded rehearsal mark logic, supporting fluid navigation without needing to manually scrub through the score. Annotation layers were toggleable and served as interpretive guides, offering suggested mappings between visual gestures and musical response.

The rehearsal and development process took place over several months, with performers accessing the evolving score via a remote server as features were added and refined. This iterative model allowed the composer to respond quickly to feedback, resolving bugs and implementing new behaviors in real time. The final performance used a local network setup to ensure reliability, with each client fully synchronised but capable of operating independently in case of failure. Notably, the technical workings of the score were never a topic of concern among the performers themselves—an indication that the infrastructure operated smoothly and unobtrusively throughout the rehearsal and performance process.

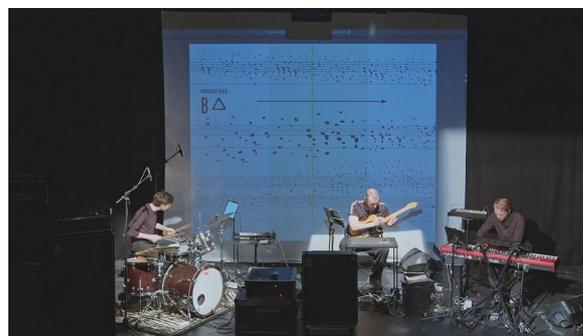


Figure 4: PonySays trio performing Rob Canning's composition *1:10,560 (6 inches to the Mile)*, 2025 — intermedia score for electric guitar, synthesiser, and drums — at Dublin Sound Lab's Music Current Festival, Project Arts Centre, Dublin. The musicians performed using iPads synchronised over a local network with Oscilla, while the projector was connected as a fourth client displaying the score to the audience.

#### 4. DISCUSSION AND REFLECTION

The development and use of OscillaScore in *1:10,560* raised several questions and insights beyond the technical implementation. At its heart, the system supports a improvisational ethos—providing structured, animated guidance while allowing performers freedom to shape timing and detail. The score's combination of constraint and openness created space for interpretation without undermining the work's formal identity. These

compositional values were further reflected in the dual role of composer as system developer, where moments of musical inspiration often emerged from technical problem-solving, and vice versa.

#### 4.1 Comprovisation and Constraint

The compositional approach in *1:10,560* reflects a improvisational sensibility: the use of structured materials that are open to interpretation, variation, and real-time responsiveness. While the overall form and cue sequence were fixed, performers were given agency in how they responded to visual gestures, managed timing within pauses, and interpreted animations such as scaling, rotation, and motion.

Rather than functioning as a prescriptive timeline or playback script, the score acted as a guide—an interactive surface of suggestion. Performers were encouraged to interpret notational gestures in ways that fit their instrument and idiom. This flexible framework enabled divergent responses while maintaining a shared temporal and structural outline across the ensemble.

In this setting, trust between composer and performer became a foundational element. Because the system handled cue triggering, synchronisation, and transitions automatically, the composer did not need to micromanage detail. Instead, their role shifted toward shaping an environment in which musical decisions could emerge organically from the score's behaviors. This shift in authorship underscores a broader philosophy of graphic score practice: that control can be distributed, and structure does not preclude freedom.

#### 4.2 Composer–Developer Dual Role

The creation of *1:10,560* highlighted both the possibilities and tensions that arise when the composer is also the software developer. The dual role offers an unusual degree of control over both medium and message: musical ideas can lead directly to the invention of technical features, and conversely, technical discoveries can inspire new compositional directions. In this project, specific animation strategies, cue behaviors, and interaction patterns were often developed in response to emerging musical needs—such as the desire to pause, loop, or randomly reorder fragments—and those features now remain available for future works.

At times, however, the separation between composing and coding became blurred or even conflicted. A musical problem might be deferred in favour of solving a technical one, or vice versa. One role could serve as a refuge from the demands of the other, leading to productive cycles but also occasional imbalance. This push and pull shaped the piece's development: the software did not merely support the score, it *became part of the compositional terrain*.

This experience echoes a wider trend in experimental music and digital art, where composers and artists increasingly create their own tools. Yet it also underscores the need for usable, adaptable systems that do not require deep technical expertise to modify. While the development of OscillaScore remains ongoing, the author recognises the importance of building a broader community of users and contributors who can shape the tool in new directions—without starting from scratch.

#### 4.3 Workshops, Accessibility, and Community Potential

Beyond its use in a commissioned work, OscillaScore has shown promise as a lightweight, adaptable framework for broader artistic and educational contexts. This was tested in a workshop presented by the composer at the Contemporary Music Centre of Ireland during the Music Current Festival 2025. In this session, ten composers each ran a separate OscillaScore instance on shared infrastructure—a single budget laptop and consumer-grade router—demonstrating the system's ability to support multiple users in parallel with minimal technical overhead. Server processes were isolated by port, and setup scripts allowed rapid deployment.

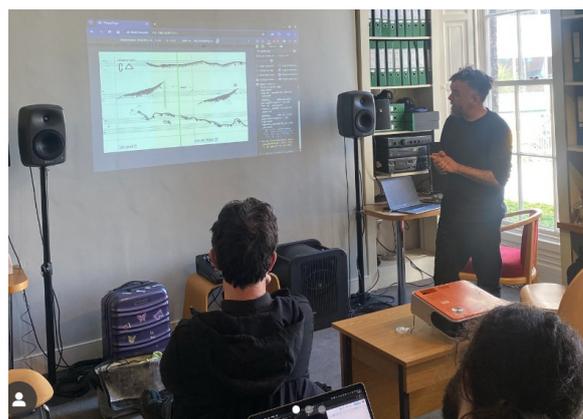


Figure 5: The author demonstrating OscillaScore to participating composers during a workshop at the Contemporary Music Centre of Ireland, Music Current Festival 2025. Each participant ran an individual instance of the score system on shared local infrastructure.

Participants responded enthusiastically, with many expressing interest in using the system for their own compositions. Several noted that OscillaScore's architecture—rooted in standard web technologies and modular JavaScript—made it accessible to those with even modest coding experience. The system's clear structure and well-commented codebase encouraged composers to imagine extensions or adaptations aligned with their personal artistic needs.

These responses suggest a strong potential for OscillaScore not just as a performance tool, but as a collaborative, open-source platform. For such a

community to emerge, continued development of documentation, onboarding tools, and example scores will be essential. A future goal is to support a contributor model in which composers and developers alike can share templates, animation modules, cue handlers, and performance strategies. This would strengthen OscillaScore's position not just as a singular system, but as part of a wider ecology of experimental notation and distributed performance practices.

## 5. CONCLUSION

OscillaScore offers a flexible, browser-based environment for animated graphic scores, supporting a range of compositional approaches from fixed-form structures to open, improvisational frameworks. Developed in parallel with the composition of *1:10,560 (Six Inches to the Mile)*, the system demonstrates how lightweight, modular web technologies can serve both the compositional process and the practical needs of ensemble rehearsal and performance.

By embedding animation, cue logic, and annotation directly into the SVG score format, OscillaScore enables composers to design interactive, time-based experiences without requiring performers to install complex software or learn new systems. The smooth integration of visual design and playback logic, combined with network synchronisation and OSC support, makes it a versatile tool for contemporary practice.

The author's experience—as both composer and developer—underscored the potential of this approach, while also revealing the need for community engagement, documentation, and extensibility. Early responses from performers and fellow composers suggest that OscillaScore is already enabling new modes of experimentation. As the project continues, its open-source foundation invites others to adapt, extend, and contribute to a growing ecosystem of graphic score creation and performance. Future development will include support for live-codable animations and OSC-reactive behaviors, enabling real-time score manipulation and expanded performative interaction.

OscillaScore is free and open-source, with a pre-release version currently available on GitHub. A stable public release is forthcoming, marking the next milestone in its development.

## Acknowledgments

The author gratefully acknowledges the Arts Council of Ireland / An Chomhairle Ealaíon for funding the commission of *1:10,560 (Six Inches to the Mile)*, and the Stuttgart-based trio Pony Says for commissioning and premiering the work. Thanks also to Fergal Dowling and Dublin Sound Lab for organising the Music Current Festival, and to the Contemporary Music Centre of Ireland for supporting the development and dissemination

of this project. Technical thanks go to the Ljudmila Art and Science Laboratory (Ljubljana) for server infrastructure during development—building on ideas first explored during an artist residency there in 2010—with special thanks to Luka Frelj for his support. Appreciation is also extended to the *kompot.si* community for hosting the open-source repository.

## 6. REFERENCES

1. R. Canning, “Interactive parallax scrolling score interface for composed networked improvisation,” in *Proc. Int. Conf. New Interfaces for Musical Expression (NIME)*, 2014, pp. 144–146.
2. R. Canning, “Real-time web technologies in the networked performance environment,” in *Proc. Int. Computer Music Conf. (ICMC)*, Ann Arbor, MI, 2012.
3. R. Canning, “The M2 Compositions: A technical overview of a modular workflow towards the creation of a video score,” in *Proc. Int. Computer Music Conf. (ICMC)*, Huddersfield, UK, 2011.
4. R. Canning, *OscillaScore: Browser-based animated graphic score system*, Open source repository, 2025. [Online]. Available: <https://github.com/robanning/oscilla>
5. R. Canning, *Parallaxis: A networked scrolling score system for composed improvisation*, Performed at the Linux Audio Conference (LAC), Graz, Austria, May 10, 2013. [Online]. Available: <https://archive.org/details/ParallaxisForFourInstrumentsAndElectronics>
6. R. Canning, *Parallaxis Score System*, Open source repository, 2013. [Online]. Available: <https://git.kompot.si/rob/parallaxis>
7. R. Canning, *Desire Lines for Four Violins*, digital score composition and performance. [Online]. Available: <https://archive.org/details/DesireLinesForFourViolins>
8. R. Canning, *Pathways for Operation Integratus*, digital score composition and performance. [Online]. Available: <https://archive.org/details/PathwaysForOperationIntegratus>

9. D. Fober, Y. Orlarey, and S. Letz, "INScore: An Environment for the Design of Live Music Scores," in *Proc. Linux Audio Conf. (LAC)*, 2012.
10. L. Hope, A. Vickery, and C. Furneaux, "The Decibel ScorePlayer—A Digital Tool for Reading Graphic Notation," in *Proc. Int. Computer Music Conf. (ICMC)*, 2015.
11. V. Goudard, "John, the Semi-Conductor: A Tool for Comprovisation," *arXiv preprint*, 2018. [Online]. Available: <https://arxiv.org/abs/1811.06858>
12. C. Hope and L. Vickery, "Screen Scores: New Media Music Manuscripts," in *Proc. Int. Computer Music Conf. (ICMC)*, Huddersfield, UK, 2011.

## 6. APPENDICES

### Appendix A – System Architecture Diagram

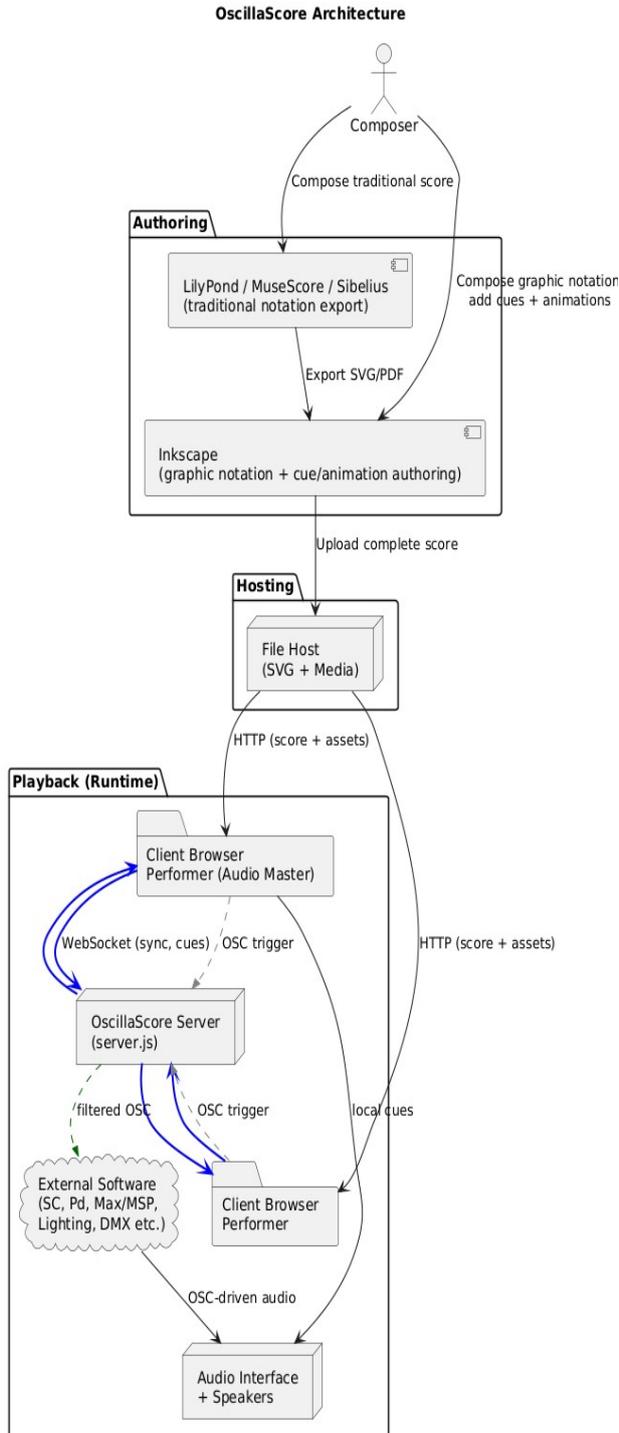


Figure 6 This diagram illustrates the full authoring-to-performance pipeline of the OscillaScore system. Composers create scores in Inkscape, embedding cue and animation logic directly into SVG element IDs. Traditional notation exported from software like LilyPond, MuseScore, or Sibelius may optionally be imported into the SVG layout to coexist with graphical material. The completed score and any associated media are uploaded to a file host, which serves them to performer clients via standard HTTP. During playback, browser-based clients run OscillaScore and synchronize with a central Node.js server via WebSockets, sharing timing, cue triggers, and state updates in real time. OSC messages are centrally routed through the server to external systems tenor such as SuperCollider, Pure Data, lighting rigs, video playback engines, or electromechanical devices. A designated audio master client is responsible for triggering audio playback—either using native browser technologies for simple fixed media, or via OSC to external audio software for more complex setups. This modular, synchronized architecture supports flexible, distributed performance with minimal technical overhead.