Holographic Encoding and Boundary–Bulk Analogs

The Holographic Principle: Information Encoded on Boundaries

The *holographic principle* posits that all the information contained in a volume of space can be encoded on the lower-dimensional boundary that encloses it [1]. This idea arose from black hole physics: Jacob Bekenstein observed that a black hole's entropy (information content) is proportional to the surface area of its event horizon, not its volume [2]. In general, there is an upper bound on the information (entropy) in any region, scaling with the area of its boundary – suggesting that the boundary's degrees of freedom suffice to describe the entire bulk region. In Leonard Susskind's words, our 3D world may be "an image of reality coded on a distant two-dimensional surface" [3].

At heart, the holographic principle treats **boundaries as informational interfaces**. Any interaction between two physical systems can be viewed as an exchange of information across a common boundary surface [4]. In this view, a boundary (whether a black hole horizon or any separating surface) behaves like a channel that *records* and transmits data about the bulk. The principle implies a kind of data redundancy: a complete description of bulk phenomena can be obtained by data on the boundary, much like a hologram stores a 3D image on a 2D film. This principle, though originating in quantum gravity, is very general – it hints that maximal information efficiency is achieved when data is stored on lower-dimensional boundaries rather than in volumetric form.

AdS/CFT Correspondence – A Concrete Bulk–Boundary Duality

The AdS/CFT correspondence is the prime example of the holographic principle in action. First conjectured by Juan Maldacena in 1997, it is an **exact duality** between two theories: a gravity theory in a *(d+1)*-dimensional *anti–de Sitter* (AdS) spacetime (the "bulk") and a *d*-dimensional *conformal field theory* (CFT) defined on that spacetime's boundary [5]. In simple terms, everything that happens in the AdS bulk (a universe with a negatively curved, hyperbolic geometry) is encoded by, and equivalent to, phenomena in the lower-dimensional boundary quantum field theory [6]. The bulk and boundary descriptions are **physically identical**, even though one is gravitational and the other has no gravity.

This correspondence provides a concrete dictionary between bulk and boundary entities. For example, a particle moving in the AdS bulk corresponds to some excitation or operator in the boundary CFT. Every bulk field $\phi(x)$ has a counterpart CFT operator $\mathcal{O}(x_{\text{boundary}})$ such that the bulk field's behavior is mirrored in correlation functions of the boundary operator [7]. Notably, the AdS/CFT duality preserves locality in a coded way: bulk locality (things happening in a certain region of the bulk) translates into specific *non*-local patterns on the boundary. A small change deep in the bulk corresponds to a subtle, distributed change in the boundary state.

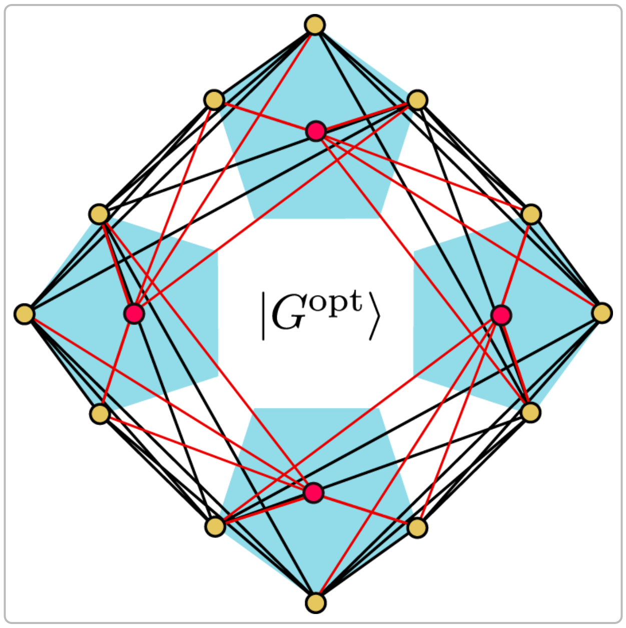
One remarkable aspect of AdS/CFT is how **geometry emerges from quantum entanglement** on the boundary. The famous Ryu–Takayanagi formula, for instance, equates the entanglement entropy of a region of the CFT to the area of a minimal surface in the bulk AdS geometry [8]. In other words, the structure of entanglement between boundary degrees of freedom literally defines the shape of spacetime in the bulk. This insight strengthened the notion that the boundary theory isn't just an approximate description but encodes *all* bulk properties – distances, curvature, even black holes – in its quantum state. The boundary "image" can thus be used to reconstruct the bulk "reality."

From the encoding perspective, AdS/CFT acts as a mapping (often called the **holographic dictionary**) that **encodes bulk information into boundary data**. Every state of the bulk gravitational system corresponds to a state in the boundary field theory; the mapping is one-to-one and invertible (an isomorphism of theories). However, decoding the bulk from the boundary can be highly non-trivial – much like deciphering a complex code. This is where the connection to error-correcting codes arises: the boundary representation turns out to be redundant and robust, as we discuss next.

Holographic Quantum Error-Correcting Codes (HaPPY Code)

A surprising discovery in the 2010s was that AdS/CFT's bulk–boundary mapping has the structure of a **quantum error-correcting code** [9] [10]. In essence, the boundary encodes the bulk in such a way that if part of the boundary information is lost or corrupted, the essential bulk information can still be recovered from the remaining part. This idea was pioneered by Almheiri, Dong, and Harlow, who pointed out that the AdS/CFT correspondence protects bulk "logical" information from localized boundary erasures, much like a quantum code protects logical qubits from errors on physical qubits [11]. Bulk operators were shown to have **multiple equivalent representations on different regions of the boundary**, suggesting a redundancy analogous to quantum secret-sharing [12]. A bulk excitation isn't localized to one specific set of boundary bits; instead, it's smeared out in a way that several distinct boundary regions can independently reconstruct it. This redundancy is precisely what one expects in an error-correcting code.

To make this concrete, researchers constructed toy models of holography using *tensor networks* that explicitly realize this bulk–boundary encoding. Foremost among these is the **HaPPY code** (named after its authors: Harlow, Pastawski, Preskill, and Yoshida). The HaPPY code uses a network of interconnected tensors arranged on a hyperbolic tiling (often pentagons/hexagons) to map a set of *bulk qubits* to a larger set of *boundary qubits*. Each tensor in the network is a specially chosen **"perfect tensor"** that is maximally entangling – it acts as a small error-correcting code on its own, with the property that any subset of its outputs contains no more information than any subset of its inputs (maximally mixed) [13]. By wiring many such perfect tensors in a fractal-like pattern, one obtains an encoding map (an isometry) from the bulk Hilbert space to the boundary Hilbert space [14]. In this construction, **bulk degrees of freedom correspond to logical qubits**, and **boundary degrees of freedom correspond to physical qubits** of a code [15]. The entire tensor network functions as an encoder for a quantum error-correcting code that has a geometric, holographic structure.



A small instance of a holographic code (the hyperbolic pentagon "HaPPY" code) represented as a network of qubits [16]*. Here the central bulk qubits (red nodes, in the interior) are encoded into a larger number of boundary qubits (white nodes on the outer circle) through entangled tensors. The code is an isometry from bulk to boundary: any given bulk qubit's state is "smeared out" over many boundary qubits. This redundancy means even if some boundary qubits are lost (erased), the bulk state can still be recovered from the remainder – a hallmark of quantum error correction.*

The HaPPY code and similar **holographic codes** capture key features of AdS/CFT in an exactly solvable way. Not only do they demonstrate how **bulk operators can be represented on multiple, different boundary regions** (mimicking the AdS phenomenon that the same bulk information is accessible from various subregions of the boundary) [10], but they also obey the Ryu–Takayanagi entropy relation exactly for the regions in these toy models [17]. In HaPPY, a logical (bulk) operator acting on the code space can be reconstructed on *either* of two different boundary segments, say segment A or segment B, even if those segments are far apart – much like how a bulk point in AdS can lie in the intersection of the entanglement wedges of two separate boundary regions. This non-uniqueness of the boundary representation is precisely what one expects if the bulk operator is a *logical qubit*: it can be recovered from multiple different sets of physical qubits [18]. In code language, the bulk operator is *protected* against erasure of any one of those boundary sets, since it can be recovered from an alternate set.

Crucially, these holographic codes are **robust to loss of boundary data**. In the HaPPY code, one can remove (erase) an arbitrary subset of boundary qubits up to a certain size and still perfectly reconstruct the original bulk state from the remaining qubits [19] [20]. This is exactly the property of a quantum error-correcting code: the logical information (bulk) is distributed with redundancy across the physical qubits (boundary). The code's distance (how many qubits can be erased before information is lost) is related to how "deep" in the bulk a given logical qubit is – qubits near the center are the most protected (requiring a large portion of the boundary to be erased to lose them), whereas qubits near the boundary are less protected [21]. This mirrors the intuition in AdS/CFT that information deep in the bulk is encoded globally and holographically across the whole boundary, while information very close to the boundary is more localized.

In summary, the holographic principle in AdS/CFT can be understood as a **coding theory**: the boundary CFT serves as a *container* of encoded data, storing the bulk physics in a redundant way. Foundational works like Pastawski *et al.* (2015) and subsequent research have cemented this view by building explicit codes and demonstrating their error-correcting features [13] [22]. This not only provides a satisfying explanation for why AdS/CFT is so robust (small perturbations on parts of the boundary don't spoil the bulk), but also offers new toy models to explore quantum gravity.

Reconstructing Bulk Information from the Boundary

A central question in any holographic encoding is **"bulk reconstruction"**: given some subset of boundary data, how much of the bulk can we recover? In AdS/CFT, this question led to the concept of the **entanglement wedge** and associated reconstruction theorems. The entanglement wedge of a given boundary region A is the region of the bulk geometry that is causally and entanglement-wise tied to A (more formally, it's bounded by the boundary region A and the minimal extremal surface homologous to A in the bulk). The *Entanglement Wedge Hypothesis* states that *everything* inside this wedge can be reconstructed from the degrees of freedom in A alone [23]. In other words, if a bulk operator lies within the entanglement wedge of A, then there is an equivalent boundary operator supported only on region A that produces the same physics [23]. This is a precise formulation of bulk–boundary encoding: boundary region A contains a complete encoding of its wedge of the bulk.

In 2016, Dong, Harlow, and Wall proved a key result: **any bulk operator can indeed be reconstructed on a boundary region A if the operator lies in the entanglement wedge of A** [24]. This went beyond earlier results that limited reconstruction to the smaller *causal wedge*. The proof used quantum information tools (specifically properties of quantum relative entropy) combined with the idea of AdS/CFT as a quantum error-correcting code [25]. Essentially, the code properties guarantee that for the allowed "code subspace" of bulk states (usually low-energy states near a semiclassical geometry), there exists a boundary operator on A that acts identically to the desired bulk operator, as long as the bulk operator is not too deep outside A's reach. This result provides a firm theoretical underpinning for the notion of subregion duality: *subregion A of the boundary ⇔ bulk entanglement wedge $E_W(A)$*. It means the boundary really does store bulk info in a *locally decodable* way – you don't always need the entire boundary, just a large enough piece to cover the bulk region of interest.

From the quantum coding perspective, one can think of each boundary region A (above a certain size) as containing a *redundant backup* of the bulk information in its wedge. The holographic code acts somewhat like a *quantum secret-sharing scheme*: the "secret" (bulk state) can be recovered from certain authorized sets of qubits (large boundary regions), but not from smaller, unauthorized sets. Indeed, one can characterize the AdS/CFT code by a threshold: any boundary region larger than a certain fraction of the whole can reconstruct a given bulk qubit, whereas smaller regions cannot – analogous to an erasure threshold in coding. For HaPPY-like codes, it's found that roughly half the boundary is the threshold: the code can tolerate up to nearly 50% of the boundary being erased and still recover central bulk information [21]. This aligns with the fact that in AdS, two halves of the boundary each have entanglement wedges that together cover the whole bulk; only when you remove more than half the boundary do you start losing parts of the bulk (like the "island" in the center that no single side can see).

Another illuminating aspect is **multiple representations**: the same bulk operator Φ can be reconstructed on different boundary regions A, B, etc., provided Φ lies in each of their entanglement wedges (for example, a bulk point near the center lies in many wedges). In the code picture, this is reflected in the non-unique logical operator representation: there are distinct boundary operators O_A, O_B, … acting on regions A, B, … that all behave as Φ on the code subspace [7] [18]. These operators O_A and O_B differ on high-energy (non-code) states but agree on the low-energy states that correspond to a smooth bulk. Almheiri *et al.* noted that this non-uniqueness is precisely what you expect if Φ is a logical operator in an error-correcting code – you can implement the same logical operation by acting on different subsets of physical qubits [12]. If one subset (boundary region) is erased (error), you can use an alternate subset to perform the operation (recovery). This correspondence has now been borne out in many models and is a cornerstone of our modern understanding of holography: *bulk locality is an emergent, redundant property supported by quantum error correction on the boundary*.

In practical terms, to *reconstruct* the bulk from the boundary, one needs to apply the correct decoding map. For full AdS/CFT, this "decoder" is not yet known in a simple closed-form – it corresponds to solving complex equations or doing integrals like the HKLL prescription (an integral kernel that smears a boundary field to produce a bulk field operator) [26]. But conceptually, the existence of the decoder is guaranteed by the code structure. In tensor network models like HaPPY, one can explicitly *push* a bulk operator outward through the network tensor by tensor (a procedure illustrated by tensor pushing algorithms) to obtain an operator on the boundary [27] [28]. Different choices of how to push through the network yield different boundary regions holding the operator (blue vs. green regions in the model correspond to two reconstructions of two different bulk points) [29] [30]. All reconstructions agree on the code subspace, showing the code's consistency [31].

The upshot is that *any* information in the bulk can, in principle, be accessed or recovered from an appropriate portion of the boundary – bulk information is not locality encoded at the boundary, but rather in a highly nonlocal, overlapping way. This deepens our analogy: it's as if the bulk is a message and the boundary is a codeword – and even when you only have part of the codeword, you might still decode the message. The larger the part you have, the more of the message (bulk) you can decode. This insight is powerful not just for physics (where it helps resolve puzzles like the black hole information paradox by suggesting how interior info might be redundantly stored in Hawking radiation), but also inspires thinking in other domains where one might want robust encodings of a complex "bulk" into a boundary interface.

Parallels for User Interface Design: "UI as a Holographic Boundary"

Beyond physics, the *conceptual logic* of holographic encoding provides a rich analogy for designing interfaces and system architectures. A **user interface (UI)** in software or product design can be thought of as a kind of "boundary" that encodes the state and capabilities of a complex underlying system (the "bulk"). The idea is that the UI presents all the relevant information and controls from a higher-dimensional internal system in a lower-dimensional form (e.g. a 2D screen) that users can interact with. Just as in holography the boundary **hides the complexity** of the bulk while still capturing its essential data, a well-designed UI exposes the functionality of a system while abstracting away the implementation details [32]. The UI is effectively the top layer of a layered architecture – in computing terms, *"the top-level virtual machine is the user interface"*, allowing limited manipulations and observations of the underlying system while concealing the deep complexity [32]. This encapsulation is exactly analogous to how the physics inside a region is encoded on its boundary: the boundary (UI) is an interface that *both reveals and restricts* access to the bulk.

Complete encoding: One lesson from the holographic principle is that an interface can in principle be *complete* – containing enough information to fully reconstruct the internal state. For UIs, this suggests an ideal of *transparency*: all the important state of the system should be perceivable from the interface, and any action that can affect the system should be achievable through the interface. In other words, the UI should serve as a faithful encoding of the “bulk” (the application logic, database, etc.). If the UI omits some critical system information or control, then from the user’s perspective that part of the “bulk” is hidden behind the boundary (analogous to losing information beyond a horizon). The holographic analogy encourages designers to think of the UI as *not just a window into the system, but as a hologram of the system* – every significant element inside has a representation on the boundary.

Redundancy and error correction: Holographic encoding teaches us about robustness via redundancy. In AdS/CFT, the bulk info is redundantly stored across the boundary; likewise, a resilient UI might incorporate redundancy so that the system’s functionality isn’t jeopardized by the failure of one element. For example, critical information could be shown in multiple places or forms (text, icon, color) such that even if one channel is missed or broken, the user still gets the message (analogous to reconstructing bulk data from a different boundary region). Similarly, multiple pathways to perform an important action could be provided (e.g. a keyboard shortcut, a menu item, and a button) – if one interface element is unavailable, another still allows the operation. This is akin to the boundary’s ability to reconstruct the bulk from different subsets: **no single point of failure** should spoil the user’s ability to interact with core functionality. The concept of *graceful degradation* in UI (where the interface still works under reduced conditions) resonates with holographic error correction, where the code gracefully handles erasures. Just as the best quantum codes recover information from just over half the qubits [33], a well-thought-out UI could allow users to accomplish goals even if parts of the interface are partially broken or removed.

Modular subregions and entanglement wedges: In a complex application, different parts of the UI often correspond to different subsystems (for instance, a dashboard might have panels for analytics, settings, user profile, etc.). We can think of each panel or section of the UI as analogous to a *boundary subregion* that gives access to a particular “entanglement wedge” of the system’s functionality. For example, the settings page (boundary region A) might let the user manipulate the configuration subsystem (its bulk wedge), while the analytics panel (boundary region B) surfaces the data processing subsystem. Ideally, these UI regions overlap enough that there isn’t a hard cut – important shared information might appear in both. This overlap is like entanglement: the regions are not independent but share context, ensuring the user can cross-reference and get a coherent picture. If one UI section is missing, the user might still retrieve some of that section’s underlying state via another section (if the design is redundant), analogous to how overlapping entanglement wedges allow reconstruction of bulk points from either region.

Layered, multi-scale representation: Holography also draws attention to **multi-scale encoding** – high-level, coarse information vs. fine-grained details. In AdS, the radial direction can be viewed as a renormalization scale: moving inward corresponds to more coarse-grained (low-energy) description, moving outward adds detail [34]. Many UIs employ a similar concept: an overview screen gives broad aggregates (coarse info) and the user can drill down into detailed views (fine info). This is reminiscent of a tensor network like MERA (Multiscale Entanglement Renormalization Ansatz), which inspired the original holographic code idea [34]. Designing a UI with a clear hierarchy of information – a top layer for summary and deeper layers for detail – ensures that the user isn’t overwhelmed (just as a holographic code organizes information by scale to handle it efficiently). One can imagine a UI that, like a perfect tensor network, cleanly splits into parts where each part presents information independent of the others up to a certain point, but together they form a complete picture.

Hiding implementation, preserving capability: Perhaps the strongest parallel is the notion of *hiding complexity while preserving capability*. In computing systems, as one adds layers of abstraction, the high-level interface hides the lower-level workings (e.g. high-level APIs hide assembly code). The user interface is the ultimate layer that hides nearly everything about internal implementation, presenting only what's necessary for the user's task [32]. Yet, from the user's perspective, the UI *is* the system – all they can know or do is through that boundary. This aligns with the holographic principle's almost philosophical implication: **reality (bulk) as perceived by an observer is entirely encoded in the information available on the boundary (interface) between observer and system** [4]. For a UI designer, this means the user's reality of the product *is* the interface. All features and feedback must be encoded at this boundary. If the encoding is done well (like a good holographic code), the user can fully control and observe the system's behavior through the UI without needing to peek "behind the curtain." If done poorly, the user will sense that some aspects are "mysterious" or not represented – akin to missing regions of the bulk that cannot be recovered from the boundary.

In summary, thinking of a UI as a **holographic boundary** encourages a mindset where *every important piece of the system has a representation on the interface*, and the interface is designed with redundancy and structure to be robust and complete. It should be possible (at least conceptually) to *reconstruct the internal state* from what is presented on the UI – much as a physicist at the boundary can, in principle, decipher the entire bulk. This doesn't mean exposing raw complexity; rather, it means encoding it in a clever way (perhaps through visualizations, summaries, alerts, and interactions) such that nothing critical is lost. The holographic analogy also inspires resilience: just as spacetime has an "intrinsic robustness" attributed to error correction [35] [36], interfaces can be made fault-tolerant and user-error-tolerant by incorporating overlapping cues and multiple ways to achieve outcomes. Ultimately, holographic encoding offers a rich metaphor and set of principles for UI/UX architects: **design the boundary (interface) to fully capture the bulk (system) in a user-accessible form, ensure redundant safe-guards for critical information, and structure the presentation in layers so that users can navigate complexity as needed without being overwhelmed.** In doing so, one creates a product interface that, much like a hologram, gives users a window into the whole system from a constrained dimension, with all the necessary information shining through.

Sources: The concept of holographic encoding originates from quantum gravity and string theory research, notably 't Hooft and Susskind's proposals [37] and the AdS/CFT correspondence by Maldacena [5]. The connection to quantum error correction was pointed out by Almheiri *et al.* [9] and further developed by Pastawski *et al.* with the HaPPY code model [13] [22]. Entanglement wedge reconstruction was formalized by Dong *et al.* [24]. The parallels to UI design and system interfaces are not from a single source but emerge from applying these concepts (e.g. Fields *et al.* on interfaces as boundaries [32]) in a broader context to inspire robust, transparent design. The discussion above synthesizes ideas across these works to bridge physics and product design.

---

[1] [2] [3] [37] Holographic principle - Wikipedia
https://en.wikipedia.org/wiki/Holographic_principle

[4] [32] arxiv.org
https://arxiv.org/pdf/2210.16021

[5] [6] [7] [8] [10] [12] [13] [14] [15] [17] [18] [19] [20] [21] [22] [26] [27] [28] [29] [30] [31] [34] [1503.06237] Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence
https://ar5iv.labs.arxiv.org/html/1503.06237

[9] [11] [1411.7041] Bulk Locality and Quantum Error Correction in AdS/CFT
https://arxiv.org/abs/1411.7041

[16] Fig. 1: A holographic graph code. | npj Quantum Information
https://www.nature.com/articles/s41534-024-00822-z/figures/1?error=cookies_not_supported&code=7dcb4df6-6913-482f-afad-8a9a890dd248

[23] Entanglement Wedge Reconstruction
https://www.emergentmind.com/topics/entanglement-wedge-reconstruction

[24] [25] [1601.05416] Reconstruction of Bulk Operators within the Entanglement Wedge in Gauge-Gravity Duality
https://arxiv.org/abs/1601.05416

[33] [35] [36] How Space and Time Could Be a Quantum Error-Correcting Code | Quanta Magazine
https://www.quantamagazine.org/how-space-and-time-could-be-a-quantum-error-correcting-code-20190103/