Qwen

Executable Contracts: A Framework for Machine-Enforced Safety in Autonomous Neurotechnology

Hardware Integrity and Data Normalization via Manifest-Centric Constraints

The foundational layer of the Cyconetics framework is built upon the principle of treating safety and governance not as soft guidelines or documentation, but as hard, machine-enforced boundaries [86]. This approach mandates that all subsequent capabilities, including autonomy and complex data processing, must operate within a verifiable and technically constrained environment. The primary mechanism for establishing this foundational integrity is the enhanced Device Capability Manifest (DCM). The DCM serves as the definitive, executable contract that defines the absolute limits of a piece of hardware, ensuring that any device deemed unsafe at the physical level is rejected before it can interact with the broader software ecosystem. This strategy moves beyond simple device identification to create a detailed safety envelope rooted in established medical and engineering standards. The research goal explicitly calls for embedding "explicit electrical limits, session ceilings, jurisdiction tags, privacy levels, and K/S/R risk bands directly into DCMs to enforce safety at device and XR-Grid binding time" . This transforms the DCM from a passive descriptor into an active gatekeeper.

The integration of explicit electrical and physiological limits into the DCM is paramount for ensuring physical safety. Clinical EEG practice provides a clear precedent, demonstrating that safe operation depends on strict adherence to parameters like leakage currents, grounding patterns, and skin integrity . These concrete requirements can be encoded directly into the manifest's schema. For instance, a DCM could include fields for maximum permissible touch current, isolation voltage ratings, and compliance with standards such as IEC 60601-1, which governs the basic safety and essential performance of medical electrical equipment [9] [27] [28]. By making these specifications first-class citizens within the manifest, the system can perform automated validation checks during the device binding process to XR-Grid zones. A device whose manifest specifies a leakage current exceeding clinical norms would fail validation and be barred from operation, preventing potential harm before any data streams or interactions can begin. Similarly,

session-related constraints—such as maximum session length, permissible stimulation protocols, and cooling periods—are critical for preventing user fatigue, discomfort, or adverse physiological responses. These session ceilings can be defined as flags or numerical limits within the DCM itself, enforced by the runtime environment to prevent unbounded or inappropriate use .

Beyond physical safety, the DCM must codify jurisdictional and privacy constraints, creating a globally aware yet locally adaptable safety framework. Jurisdiction tags, such as "US-CA" or "US-AZ," allow the system to apply region-specific legal and ethical rules . An XR-Grid zone configured for California might have stricter data privacy laws than one in Arizona, and this policy can be programmed into the zone's profile. The DCM validation process would then cross-reference the device's declared jurisdictions with the target zone's requirements. For example, a rule could state that a device tagged for "US-CA" cannot be bound to a "US-AZ" zone unless explicitly permitted by a higher-level policy . Privacy levels are another critical dimension. A DCM might include a field like `PrivacyLevel::High`, which triggers a cascade of restrictions across the entire system . As demonstrated in the provided Rust code, a policy engine could enforce that a high-privacy DCM cannot contain any High-risk HCI export rules, effectively preventing sensitive neural data from being exposed to potentially insecure or high-risk applications . This creates a multi-layered defense-in-depth, where privacy settings are not just a setting to be configured by a user but a fundamental property of the device that is respected by every component of the stack.

To support these constraints, the Cyconetics architecture mandates the early and universal normalization of all incoming BCI data streams. This architectural decision is crucial for enabling device-agnostic processing and applying consistent governance logic across a heterogeneous landscape of sensors . Instead of dealing with vendor-specific formats and proprietary data structures, all BCI hardware—from consumer-grade wearables using EEG to advanced clinical systems—is treated as a normalized, timestamped multichannel stream of data plus discrete events [64] . This abstraction layer, potentially leveraging existing frameworks like BrainFlow or Lab Streaming Layer (LSL), converts raw sensor readings into a standard Rust struct . This standard struct would contain essential metadata such as timestamps, per-channel vectors of signal data, quality flags indicating issues like bad contact or heavy artifacts, and other relevant metadata . By normalizing data early, the system ensures that all downstream tools, whether they are for analysis, HCI interaction, or AI training, receive a consistent and predictable input format. This prevents vendor-specific complexities from becoming sources of unreliability, security vulnerabilities, or interpretation errors, thereby strengthening the overall integrity of the system. The normalized data product becomes the common language for all components, from the device driver to the highest-level AI agent.

Finally, the most innovative aspect of this manifest-centric approach is the direct integration of a quantitative risk assessment model, based on K/S/R (Knowledge/Useful Knowledge, Social Impact, Risk-of-Harm) scores, into the DCM and associated site policies . This operationalizes the qualitative risk management functions found in mature governance frameworks like the NIST AI Risk Management Framework (RMF) and ISO/IEC 42001 [60] [106]. Rather than leaving risk assessment as a manual, ad-hoc process, the Cyconetics model embeds it as a structured attribute of the device itself. A DCM would carry K/S/R fields, with each score representing a quantified measure of its positive utility, social benefit, and potential for harm . These scores are not arbitrary; they are derived from a rigorous evaluation process informed by clinical norms, regulatory guidance, and technical analysis [120]. Site profiles and XR-grid zoning policies can then be configured to accept only devices and drivers that fall within a specified K/S/R threshold . For example, a low-risk EEG grid in an AZ lab might be set to accept only devices with an R-score below a certain value, while a more advanced CA research lab might permit slightly higher-risk devices, provided they meet additional criteria . This transforms abstract principles of risk mitigation into a concrete, automated, and scalable enforcement mechanism. It aligns perfectly with the Cyconetics philosophy of locking down safety and governance as hard boundary conditions before expanding capabilities, providing a powerful structural lever for managing the inherent risks of neurotechnology . The table below summarizes the key constraints embedded within the enhanced DCM.

Constraint Category	Specific Parameter Examples	Enforcement Mechanism
Electrical Limits	Maximum leakage current, isolation voltage ratings, grounding pattern compliance (e.g., IEC 60601-1) [9] [28].	Automated validation against a known-safe baseline during device binding to an XR-Grid zone.
Session Ceilings	Maximum session duration, minimum inter-session interval, stimulation flag enforcement .	Runtime enforcement by the driver or controller, preventing unbounded or unsafe usage patterns.
Jurisdiction Tags	Country/state identifiers (e.g., US-CA, US-AZ) .	Cross-referencing with XR-Grid zone policies to ensure legal and regulatory compliance before binding.
Privacy Levels	Predefined levels (e.g., High, Standard) that trigger cascading restrictions .	Policy engine enforces rules, such as prohibiting high-risk data exports from high-privacy devices.
K/S/R Risk Bands	Quantitative scores for Knowledge, Social Impact, and Risk-of-Harm .	Zone-based filtering; XR-zones can be configured to accept only devices meeting a minimum K/S/R threshold.
Stimulation Flags	Indication of whether the device can deliver electrical or magnetic stimulation .	Mandatory human approval or different gating rules for devices capable of closed-loop stimulation.

By building this comprehensive, multi-faceted constraint model directly into the DCM, the Cyconetics framework establishes an unbreachable foundation of hardware integrity and data consistency. Every subsequent layer of the system, from data processing to AI-driven autonomy, operates knowing that its inputs are guaranteed to be within

predefined safety envelopes. This manifest-centric approach is the cornerstone of the entire research framework, enabling all future capabilities while rigorously containing their potential for harm.

Policy-Governed Interaction with HCI Export Profiles

Once a robust foundation of hardware integrity and data normalization is established, the next critical layer of the Cyconetics framework focuses on governing the flow of information between the brain-computer interface (BCI) and the human-computer interface (HCI). This is achieved through the introduction of "HCI export profiles," a pivotal concept that acts as a highly constrained, auditable, and validated interface . The core purpose of an HCI export profile is to define, with machine-checkable precision, exactly which BCI-derived states may be consumed by higher-level HCI or Extended Reality (XR) tools, under what specific conditions of jurisdiction, XR-zone, and risk, and with what safety and privacy constraints . This mechanism directly addresses the profound concerns surrounding data privacy and the potential for "black-box mind control" by making the flow of sensitive neural information transparent, controllable, and consent-oriented . Instead of allowing HCI tools to pull arbitrary signals from the BCI stream—a dangerous open-door policy—the system adopts a gated-access model where every piece of information is authorized by a formal rule within the DCM's corresponding export profile .

An HCI export profile is designed as a first-class, validated artifact that is cryptographically and semantically tied to the underlying Device Capability Manifest (DCM) it describes . This tight coupling is a fundamental security feature. In the proposed Rust implementation, the `HciExportProfile` struct contains a `device_manifest_id` field, which is checked during validation against the loaded DCM's ID . If there is a mismatch, the profile is immediately rejected, ensuring that HCI behavior is always anchored to the verified capabilities and constraints of the physical hardware . This prevents scenarios where a malicious or misconfigured HCI tool attempts to interpret data from a device it was never authorized to access, or where a device's capabilities are misrepresented.

The structure of an HCI export profile is meticulously defined to provide granular control over the exported data. At its core, the profile consists of a set of `HciExportRule` objects, each defining a single allowed mapping from a BCI-derived state to an HCI/XR output channel . Each rule is rich with metadata that collectively forms the policy

governing its use. The `kind` field, an enum (`CoarseCognitiveState`, `DiscreteIntent`, etc.), categorizes the type of mental state being exported, preventing ambiguity about what the data represents . The `label` field provides a human-readable name for the state, aiding in debugging and audit trails . Crucially, each rule is bound to specific `jurisdictions` and `xr_zones`, ensuring that the data flow is geographically and spatially restricted according to pre-defined legal and operational policies . For example, a rule exporting a "workload_level" state might be valid only within the "Phoenix CA-mode research lab grid" and only for the "US-CA" jurisdiction .

Furthermore, each export rule is assigned a `risk_level` (Low, Medium, or High) and a `rate_limit` (maximum updates per second) . The risk level is a critical parameter used by downstream policy engines to make decisions about how the data can be used. The risk level directly informs policy decisions, such as whether a state can be used for closed-loop actuation or if it requires explicit, per-session user consent . The rate limit prevents information overload and potential misuse by capping the frequency of data emission, a simple but effective throttling mechanism . One of the most important constraints is the `no_closed_loop_use` boolean flag, which explicitly forbids any rule marked with it from being used to drive actuators or create closed-loop feedback systems . This is a non-negotiable safety guardrail, particularly for states that are coarse approximations of a user's internal state, preventing an AI from making consequential decisions based on potentially unreliable interpretations of brain activity.

The practical implementation of these profiles is demonstrated through a complete Rust module, `hci_profile.rs`, which defines the necessary structs (`XrZoneRef`, `HciStateKind`, `HciRateLimit`, `HciRiskLevel`, `HciExportChannel`, `HciExportRule`, and `HciExportProfile`) and, critically, a `validate_against` method . This method performs a series of structural and logical checks to ensure the integrity of the profile relative to the DCM and its own internal rules . For example, it verifies that the profile's `device_manifest_id` matches the DCM's ID and that at least one export rule is defined. More importantly, it implements business logic, such as the policy that a high-privacy DCM cannot have any high-risk export rules . This validation logic is the engine of the entire system; it is what makes the manifest an "executable contract." When a controller loads a DCM and an associated HCI export profile, it must call `profile.validate_against(&dcm)` and only proceed if the call returns `Ok(())`. This step is a mandatory gate in the driver creation pipeline, ensuring that no invalid or unsafe configuration can ever reach the hardware .

The following JSON example illustrates a concrete application of an HCI export profile for a BrainFlow-backed OpenBCI-class device operating in a US-CA lab XR-zone . This profile is designed to be safe and compliant, exporting only coarse-grained workload and

engagement states, both of which require explicit user consent and are rate-limited to prevent excessive polling. The workload state is designated as "medium" risk, while the engagement state is "low" risk. The workload state is also flagged as `no_closed_loop_use`, reflecting a conservative approach to acting on cognitive load estimates . The entire profile is cryptographically signed and stored in a sovereign registry alongside the DCM, forming a complete, verifiable, and self-contained unit of capability description .


```
{
  "id": "1c0f0b43-58e0-4a6f-8a4f-3a0b2b847901",
  "name": "OpenBCI-CA-Lab-Coarse-HCI",
  "version": "0.1.0",
  "device_manifest_id": "4f3f44a2-7c9b-4b8a-9a84-4473ed1fbb10",
  "created_at": "2026-01-21T10:50:00Z",
  "rules": [
    {
      "id": "a0c5c4d9-3e8a-4b6f-9b58-19b30f89b201",
      "kind": "coarse_cognitive_state",
      "label": "workload_level",
      "jurisdictions": ["US-CA"],
      "xr_zones": [
        {
          "zone_id": "XR-ZONE-CA-LAB-1",
          "description": "Phoenix CA-mode research lab grid"
        }
      ],
      "risk_level": "medium",
      "rate_limit": { "max_hz": 1.0 },
      "export": {
        "topic": "bci.hci.workload",
        "anonymized": true,
        "notes": "3-level workload, 10 s rolling window"
      },
      "requires_explicit_consent": true,
      "no_closed_loop_use": true
    },
    {
      "id": "b7504cdc-9b75-4c86-9cb5-d6f5d9953a10",
      "kind": "engagement_flag",
      "label": "engagement_state",
```

```
      "jurisdictions": ["US-CA"],
      "xr_zones": [
        {
          "zone_id": "XR-ZONE-CA-LAB-1",
          "description": "Phoenix CA-mode research lab grid"
        }
      ],
      "risk_level": "low",
      "rate_limit": { "max_hz": 2.0 },
      "export": {
        "topic": "bci.hci.engagement",
        "anonymized": true,
        "notes": "binary engaged/disengaged, hysteresis smoothing"
      },
      "requires_explicit_consent": true,
      "no_closed_loop_use": false
    }
  ]
}
```

This JSON representation, when deserialized into the corresponding Rust structs, becomes subject to the full power of the `validate_against` method, ensuring that the intended safe behavior is preserved and enforced by the machine . By treating HCI export profiles as first-class, validated artifacts, the Cyconetics framework provides a powerful grammar for specifying a safe and auditable interaction surface between the user's brain and the computational world. It precisely answers the question of "what information can flow where and under what rules?", transforming a complex problem of neuro-interface governance into a structured, manageable, and machine-verifiable set of constraints.

Sovereign Crate Development and Artifact Distribution

With a solid foundation in manifest-driven hardware constraints and policy-governed HCI interaction, the third pillar of the Cyconetics framework addresses the secure and sovereign development and distribution of the software that runs on this constrained platform. This involves a deliberate two-stage strategy for crate development and the establishment of a resilient, trust-minimized distribution backbone. The strategy prioritizes stability and security by starting with sovereign, internally-versioned crates before deliberately opening them up to controlled external contributions. This phased

approach ensures that the foundational constraints of the system are stable and aligned with strict K/S/R targets before introducing external variables, thereby minimizing early-stage chaos and governance drift . The ultimate goal is to create a self-sustaining ecosystem where all software artifacts—from core libraries to drivers and protocols—are governed by the same principles of verifiability and accountability as the hardware they run on.

The initial phase of crate development is strictly sovereign. New, focused Rust crates such as `cyconetics-bci-core` (for DCMs, device abstractions, and HCI profiles) and `cyconetics-bci-policy` (for site profiles and XR-grid zoning rules) are authored and versioned internally by the core stakeholder group . This sovereign development path mirrors a "secure Create" pipeline, where the emphasis is on establishing a rock-solid baseline of functionality and governance . During this phase, all development occurs within a controlled environment, likely using a sovereign Git repository backed by decentralized storage networks like ALN/bostrom to ensure persistence and immutability . This initial stage allows the team to stabilize the core contracts—the schemas for DCMs, the logic for policy validation, and the definitions for XR-grid bindings—without the pressure of external demands or the risk of dilution for the sake of convenience . The artifacts produced in this phase are not intended for public consumption; their sole purpose is to form the trusted root of the entire cybernetic system.

Once this sovereign baseline is considered stable and production-ready, the framework transitions to the second phase: controlled openness. This is achieved by exposing carefully bounded extension points within the crates, such as traits for new driver implementations, schemas for new protocol graphs, or definitions for new risk rules . These extension points are not arbitrary hooks for free-form code; they are themselves governed by the manifest and policy system. External contributors, including autonomous AI agents, can propose changes or extensions that adhere to these predefined contracts . However, these proposals do not land directly in the main codebase. Instead, they enter a formal, DID-signed review pipeline . This pipeline is the heart of the governance model. It ensures that any modification to the sovereign codebase must pass through a series of checks: automated static analysis, synthetic-board tests, and, most importantly, human or multi-party approval [46] . Only after a proposal has been reviewed and approved by stakeholders holding specific Policy DIDs is the resulting artifact signed with a Release DID key and published to the sovereign registry . This process guarantees that authorship is always attributable to valid, privileged stakeholder positions and that no unvetted change can compromise the integrity of the system . It effectively separates the creative drafting phase from the authoritative publishing phase, turning the AI chat into a powerful assistant rather than a deployment authority .

Central to this entire model is the sovereign distribution backbone, a minimal but robust infrastructure designed to guarantee safe self-upgrade paths for AI agents and other system components. This backbone comprises three key elements: DID verification, local caching, and immutable registries . The use of Decentralized Identifiers (DIDs) is fundamental to establishing trust in a distributed system [34]. Every artifact—be it a Cargo crate, a DCM, a driver binary, or a protocol definition—is cryptographically signed by its creator's DID [35] [37]. A `did:self` registry-less method could even be used for simpler cases, though a more robust system would involve a verifiable data registry [36] [38]. When a node in the XR-Grid needs to load a new artifact, it first verifies the signature against a trusted root of DIDs maintained locally. This cryptographic verification replaces trust in a central server or repository with trust in a decentralized, mathematically provable chain of custody [89]. This mechanism is a direct defense against supply chain attacks, where an attacker compromises a build pipeline or repository to inject malicious code, a growing concern highlighted by incidents targeting CI/CD systems like GitHub Actions [46] [57].

The registry itself is envisioned as being potentially backed by decentralized persistent identifier (DPI) infrastructure like IPFS or bostrom/ALN, which offers benefits like immutability, availability, and resistance to censorship [19] [33]. Immutability is a critical property; once an artifact is published and its hash is recorded, it cannot be changed [33]. This ensures that all nodes can deterministically verify that they are loading the exact, approved version of a component. Redundant mirrors of this registry would provide automatic failover, enhancing availability . To address the reality of network partitions and temporary outages, the backbone incorporates local caching . Nodes cache recently loaded artifacts, allowing them to continue operating safely and reliably even if the remote registries become temporarily unavailable. This read-only mode on failure is a key resilience pattern, ensuring that the system's core functionality persists without degradation . The combination of DID-signing, decentralized storage, and local caching creates a distribution system that is not only secure but also robust and autonomous, providing the necessary substrate for AI agents to learn, adapt, and upgrade themselves safely within the confines of the Cyconetics framework.

Governing Autonomous AI Through Executable Contracts

The Cyconetics framework is designed not to suppress the power of artificial intelligence but to harness it within a tightly controlled, safety-first environment. The guiding

principle for AI interaction is to treat AI agents, particularly AI-chats, as "template fillers" rather than "free coders" . Their role is confined to operating within specific, manifest-governed "vertices" where they can assist in generating code or configurations that adhere to strict schemas and policies . This approach fundamentally shifts the relationship with AI from one of delegation to one of assisted drafting, where the AI proposes changes but the final authority rests with the human-controlled validation and signing gates. This strategy is a practical embodiment of "Compliance-as-code," where governance rules are not just documented but are deeply integrated into the very fabric of the development toolchain, making non-compliant actions difficult or impossible to execute [86].

The framework identifies several distinct vertices where AI assistance can be safely and effectively applied. The first and most critical vertex is DCM and HCI profile template completion . An AI agent, given a high-level natural-language goal like "create an export profile for a research-grade EEG device that monitors coarse workload levels in our California lab zone," can be prompted to populate the JSON or Rust struct literals for an `HciExportProfile` . The AI's output is not executed directly. Instead, it is fed into the `validate_against` function of the DCM. The validation logic, written in Rust, performs a battery of checks: it confirms the DCM ID match, verifies that the jurisdiction and zone references are valid, ensures that the risk level is appropriate for the device's privacy classification, and confirms that all required fields are present . Only if the AI-generated manifest passes all these machine-enforced checks is it considered valid. This turns the AI into a sophisticated autocompleter for a highly structured domain-specific language, preventing syntax errors and simple logical inconsistencies.

A second vertex involves per-zone crate configuration . For different XR-Grid zones (e.g., a high-security lab vs. a public demonstration area), different subsets of functionality may be desired. Within this vertex, an AI can be tasked with generating small "profile crates" or Rust feature flags that select a subset of the validated export rules defined in a master HCI profile . The AI's task is not to invent new rules or modify existing ones, but to compose existing, validated rulesets into a new configuration. All selections made by the AI must be proven to be a subset of the rules available in the parent profile, a check that can be performed by the build system or a dedicated validator crate. This allows for flexibility and specialization without compromising the integrity of the foundational ruleset.

A third vertex is tool-call binding . Once a safe and validated export rule exists—for example, one that defines a topic named `"bci.hci.workload"`—the AI can suggest mappings from this topic to specific entry points in other tools or to UI widgets in the XR environment . The AI might propose that messages arriving on the `bci.hci.workload`

topic should update a particular gauge in the VR interface. However, the AI is explicitly forbidden from changing the underlying constraints, such as the sampling rate, session limits, or, most importantly, the risk level of the rule itself . Any attempt to alter these core properties would require the generation of an entirely new manifest version, which would then have to go through the full validation and review cycle. This separation of concerns is vital: the AI can help connect the dots between a validated data source and a specific application, but it cannot rewrite the rules of the data source itself.

This entire model relies on a clear distinction between the types of operations that are permissible. The table below outlines the distinctions in the proposed workflow.

Operation Type	Permissible AI Action	Gatekeeper / Validator	Rationale
Schema Population	Filling in `HciExportProfile` JSON/ Rust literals from natural language prompts.	`validate_against` function in `hci_profile.rs`.	Prevents structural and logical errors in the manifest before it is ever signed or used.
Configuration Composition	Generating feature flags or small "profile crates" that select subsets of validated rules.	Build system / Schema validator.	Allows for flexible, zone-specific configurations derived from a stable, sovereign base.
Tool Binding	Suggesting mappings from an export topic (e.g., `bci.hci.workload`) to a tool entry point or UI widget.	Human reviewer / Policy engine.	Connects validated data to applications without altering the data's fundamental properties or risk rating.
Code Generation	Generating raw FFI code, unsafe blocks, or modifying low-level driver logic.	**Prohibited**. Hand-written and heavily audited.	Low-level code has direct hardware access and presents the highest risk of introducing catastrophic bugs or security vulnerabilities.
Policy Modification	Changing a DCM's session ceiling, risk band, or privacy level.	**Prohibited**. Requires a new manifest version and a full DID-signed review process.	Core safety and governance parameters must undergo a rigorous, auditable, and consensus-based approval process.

By restricting AI to these specific, well-defined vertices, the framework leverages its strengths in pattern recognition and text generation while completely avoiding its weaknesses in reasoning, security, and reliability. The AI becomes a powerful co-pilot for developers and researchers, accelerating the creation of compliant artifacts without ever stepping outside the bounds of the executable contracts that protect the system and its users. This approach ensures that all code, whether written by a human or suggested by an AI, ultimately conforms to the same stringent safety and governance standards, maintaining a consistent and trustworthy environment for autonomous operation.

Incident-Driven Feedback Loops for Continuous Improvement

The pinnacle of the Cyconetics framework is its capacity for autonomous learning and adaptation, enabled by a sophisticated incident-driven feedback loop. This mechanism moves the system beyond a static set of rules to a dynamic, self-improving entity that can evolve its own safety and governance policies based on real-world experience. The core idea is that autonomous AI agents, operating within the XR-Grid, are equipped not only to execute tasks but also to monitor for anomalies, diagnose issues, and propose targeted refinements to the underlying manifests and policies . This creates a continuous "Plan-Do-Check-Act" cycle, a cornerstone of modern quality management systems like ISO 42001, but implemented at a technical, automated level [60] [86]. The sovereign distribution backbone, featuring DID-signed registries and local caches, provides the essential, trustworthy channel through which these learned improvements can be deployed, ensuring that fixes and policy updates are propagated securely and reliably across the network .

The process begins when an AI agent detects an incident. This could be a wide range of events, from a sudden spike in signal artifacts indicating poor electrode contact, to a violation of a session ceiling, to an unexpected behavioral pattern from a BCI-derived state that contradicts its expected label. The agent's first action is to log the incident with full context: the device ID, the DCM ID, the XR-zone, the jurisdiction, and the relevant policy rules that were in effect . This creates a detailed, auditable trace that is invaluable for later analysis . With the incident logged, the agent can then analyze the cause. It might compare the observed data against the device's electrical limits, query the K/S/R database for similar past incidents, or check the validity of the HCI export profile currently in use.

Following diagnosis, the agent's most critical function is to propose a corrective action. This proposal is not a vague suggestion but a concrete, executable change to an artifact within the system. For example, if the incident was caused by a session running too long, the agent might propose a new version of the DCM with a lower `session_max_minutes` value. If the issue was a misclassified risk level, it might propose a revised `HciExportRule` with a different `risk_level` tag. These proposals are not modifications to live, running code but are instead new versions of the immutable artifacts that govern the system's behavior. They are packaged as new DCMs, revised HCI profiles, or updated site policies, each with a new version string and signed by the agent's operational DID .

The proposed change then enters the DID-signed review pipeline, which serves as the "Check" and "Act" phases of the cycle . The proposal is routed to the appropriate reviewers, who might be human experts or a quorum of stakeholder DIDs. The reviewers examine the evidence provided by the AI agent, assess the proposed change, and decide whether to approve or reject it. This human-in-the-loop oversight is a critical safety net, preventing the propagation of flawed or malicious AI-driven changes [13]. If the change is approved, a release DID signs the new artifact version. This signature is the final authorization, cryptographically binding the change to a trusted stakeholder identity . The newly signed artifact is then pushed to the sovereign registry.

The final step is deployment. Because the system relies on a local artifact cache, nodes within the XR-Grid can autonomously check the registry for updates. Upon detecting a new, signed version of a manifest or policy that applies to their local configuration, they can download and activate it. This triggers a safe self-upgrade path, where the node's behavior is refined based on the lessons learned from the incident. The old, problematic behavior is replaced by the new, corrected behavior without requiring manual intervention. This entire loop—from detection to deployment—is automated, allowing the collective intelligence of the AI agents to continuously improve the safety and efficiency of the entire system. This incident-driven refinement is the engine of the "quantum-learning" behavior described in the research goal, enabling the system to adapt to unforeseen circumstances and incorporate new knowledge in a secure and verifiable manner . It transforms the static contracts of DCMs and policies into a living, evolving document that grows smarter and safer over time.

Synthesis and Future Research Directions

The Cyconetics research framework presents a comprehensive and coherent architecture for developing autonomous, AI-compatible BCI/HCI systems grounded in machine-enforced safety and governance. Its central thesis is unequivocal: true autonomy cannot be responsibly pursued until a verifiable and technically enforceable boundary of safety is firmly established. This principle is operationalized through a multi-layered, policy-driven system that treats manifests, policies, and artifacts as "executable contracts" that all system components, including AI agents, must obey [86]. The architecture is not merely a collection of disparate technologies but a deeply integrated cybernetic loop, designed to manage risk at every stage, from hardware specification to autonomous adaptation.

The framework's synthesis can be understood as a hierarchical, constraint-based system. At its base lies **Layer 1: Hardware Integrity**, secured by enhanced Device Capability Manifests (DCMs) that encode explicit electrical limits, session ceilings, and jurisdictional tags, drawing on established medical device standards like IEC 60601-1 [9] [28]. This is complemented by a universal data normalization layer that translates all BCI streams into a device-agnostic, timestamped format, enabling consistent processing and governance . Building upon this foundation is **Layer 2: Policy-Governed Interaction**, where HCI export profiles act as gated interfaces, precisely controlling the flow of BCI-derived states to higher-level applications based on risk, jurisdiction, and explicit user consent . This layer is fortified by a **Layer 3: Sovereign Software Lifecycle**, managed through internally-developed Rust crates and a resilient distribution backbone of DID-signed, immutable registries. This ensures that all software components are cryptographically verifiable, auditable, and protected against supply chain attacks [46]. Finally, the framework enables **Layer 4: Autonomous Learning**, where AI agents operate within these rigid constraints, using incident-driven feedback loops to propose and deploy refined manifests and policies, thus closing the loop on a continuous improvement cycle .

Despite its robustness, the successful implementation of this framework hinges on addressing several critical areas of uncertainty and requires further research. The most significant challenge is the **implementation complexity of the validation logic**. The entire system's safety depends on the correctness of the Rust code responsible for validating manifests and policies, such as the `validate_against` method . Given the combinatorial explosion of possible interactions between DCMs, XR-zones, jurisdictions, and AI-generated protocols, ensuring this logic is exhaustive and bug-free will be immensely challenging. The application of formal methods, model checking, or symbolic execution techniques, as discussed in computer-aided verification conferences like CAV, may be necessary to provide mathematical assurance of the validator's correctness [92].

Second, the **governance of the review pipeline** remains a critical social and organizational question. While the framework mentions "DID-signed review pipelines," the operational mechanics are not fully specified . Defining the roles, responsibilities, and quorum requirements for reviewers (e.g., a simple majority vs. a supermajority of Policy DIDs) is essential to prevent bottlenecks, bias, or capture of the governance process. Developing a transparent and equitable governance model that can handle disputes and evolving stakeholder needs is as important as the underlying technology.

Third, the **performance overhead** introduced by multiple layers of validation, serialization/deserialization, and cryptographic verification must be carefully evaluated. For real-time BCI/HCI applications where millisecond latencies matter, the cumulative delay from these checks could be prohibitive. Optimizing the performance of the Rust-

based validators and exploring efficient serialization formats will be crucial for practical deployment. The use of WebAssembly (Wasm) plugins has been explored for similar extensibility scenarios, though some note its limitations [14] [43] [45].

Fourth, the **definition of "high-risk" states** requires deep interdisciplinary work. While the framework proposes K/S/R scores, the criteria for assigning these scores to emergent states derived by AI models (e.g., a novel intent class) are not yet defined. This necessitates collaboration between neuroscientists, psychologists, ethicists, and legal scholars to establish a principled basis for risk assessment that is both scientifically sound and ethically defensible.

Finally, the **long-term scalability and sustainability** of the sovereign infrastructure pose a significant operational challenge. Running and maintaining a redundant, globally distributed network of self-hosted, DID-signed artifact registries is a substantial undertaking. Investigating the economic models, automation strategies, and community governance structures needed to ensure the long-term viability of this infrastructure, especially for smaller labs or citizen-led initiatives, is an essential area for future research.

In conclusion, the Cyconetics framework provides a visionary and actionable blueprint for building the next generation of neurotechnology. By anchoring all capabilities in a foundation of machine-enforced, executable contracts, it offers a compelling path toward achieving powerful autonomy without sacrificing safety or user sovereignty. The framework successfully translates high-level ethical goals into a concrete, implementable architecture grounded in Rust, DIDs, and policy-driven manifests. The remaining challenges, while significant, represent opportunities for further innovation in software engineering, governance, and human-computer interaction.

Reference

1. (PDF) Neural interface technology for human-computer ... https://www.researchgate.net/publication/376680794_Neural_interface_technology_for_human-computer_interaction
2. Complex, Intelligent and Software Intensive Systems https://link.springer.com/content/pdf/10.1007/978-3-031-96096-3.pdf

3. Extended Reality: Lucio Tommaso de Paolis Pasquale ... https://www.scribd.com/document/679594904/978-3-031-43404-4
4. Intelligent Technologies for Interactive Entertainment https://link.springer.com/content/pdf/10.1007/978-3-030-76426-5.pdf
5. Using $crate in Rust's procedural macros? https://stackoverflow.com/questions/44950574/using-crate-in-rusts-procedural-macros
6. Integral Security Pillars for Medical Devices https://www.mdpi.com/2076-3417/15/12/6634
7. IEEE/UL Standard for Clinical Internet of Things (IoT) Data ... https://ieeexplore.ieee.org/iel8/10697444/10697445/10697446.pdf
8. Guidelines for Registration of Medical Devices Doc. No. https://rwandafda.gov.rw/monitoring-tool/documents-management/uploads/1/Guidelines/1767694148_Guidelines%20for%20Registration%20of%20Medical%20Devices_December%202025.pdf
9. Yadin B. David https://www.fda.gov/media/160315/download
10. Medical Device Safety Management Using Cybersecurity ... https://www.researchgate.net/publication/342250668_Medical_Device_Safety_Management_Using_Cybersecurity_Risk_Analysis
11. Sonosite ST User Guide https://www.sonosite.com/support/userdocs/Sonosite_ST_UG_ENG_P31244-04B_e.pdf
12. EU AI Act vs ISO 42001 vs NIST AI RMF https://www.linkedin.com/posts/runa-dalal-cyber-strategy-ai-risk_comparing-ai-governance-requirements-activity-7381601865446195200-T15a
13. AI Act | Shaping Europe's digital future - European Union https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai
14. 使用Rust为网格代理开发Wasm插件 - 阿里云文档 https://help.aliyun.com/zh/asm/sidecar/developing-a-wasm-plugin-for-grid-agents-using-rust
15. 为Rust 程序提供插件功能：WebAssembly 组件模型 https://juejin.cn/post/7438442651399307279
16. How Augment Code solved the challenge of AI for big ... https://www.linkedin.com/posts/scottdietzen_garry-tan-garrytan-on-x-activity-7339012057440112641-wnWJ
17. 虚幻引擎5.0版本说明 https://dev.epicgames.com/documentation/zh-cn/unreal-engine/unreal-engine-5.0-release-notes?application_version=5.0
18. Model Context Protocol servers https://gitee.com/mirrors/Model-Context-Protocol
19. (PDF) Decentralized Persistent Identifiers: a basic model ... https://www.researchgate.net/publication/

331007493_Decentralized_Persistent_Identifiers_a_basic_model_for_immutable_handlers

20. Microsoft Marketplace general listing and offer policies https://learn.microsoft.com/en-us/legal/marketplace/certification-policies
21. Issue when replacing a crates dependency with a local ... https://stackoverflow.com/questions/32791086/issue-when-replacing-a-crates-dependency-with-a-local-version
22. Software Architecture Validation and Verification https://www.qt.io/quality-assurance/blog/software-architecture-verification-and-validation
23. A Functional Architecture for the Verification, Validation ... https://dl.acm.org/doi/10.1145/3507485.3507486
24. Emerging Trends in Software Architecture from the ... https://arxiv.org/html/2507.14554v2
25. A Layered Software Architecture for the Development of ... https://www.mdpi.com/2076-3417/15/7/3664
26. WiseCFD V2023: A software framework with open ... https://hkxb.buaa.edu.cn/EN/10.7527/S1000-6893.2024.30440
27. Recognized Consensus Standards: Medical Devices - FDA https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfStandards/detail.cfm?standard__identification_no=45394
28. Recognized Consensus Standards: Medical Devices - FDA https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfStandards/results.cfm?start_search=1&productcode=&category=&type=&title=&organization=&referencenumber=60601®ulationnumber=&effectivedatefrom=&effectivedateto=&pagenum=25&sortcolumn=pdd
29. •Basic Safety and Essential Performance of Medical ... https://www.fda.gov/media/142385/download
30. Iec 60601-2-57 - 2023 | PDF | Technology & Engineering https://www.scribd.com/document/665553540/IEC-60601-2-57-2023
31. Recognized Consensus Standards: Medical Devices - FDA https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfStandards/results.cfm?start_search=1&productcode=&category=&type=&title=&organization=&referencenumber=60601%2D2%2D®ulationnumber=&effectivedatefrom=&effectivedateto=&pagenum=50&sortcolumn=pd
32. 9.5 Release Notes | Red Hat Enterprise Linux | 9 https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html-single/9.5_release_notes/index
33. Charlotte: Reformulating Blockchains into a Web of ... https://dl.acm.org/doi/full/10.1145/3607534
34. Decentralized Identifiers (DIDs) v1.0 https://www.w3.org/TR/did-core/

35. Verifiable Credentials Data Model v2.0 https://www.w3.org/TR/vc-data-model-2.0/
36. did:self A registry-less DID method https://arxiv.org/html/2504.20767v1
37. Verifiable Credentials Data Model v1.1 https://www.w3.org/TR/vc-data-model-1.1/
38. Self-verifiable content using decentralized identifiers https://www.sciencedirect.com/science/article/abs/pii/S138912862300244X
39. Procedural Macros - The Rust Reference https://rustwiki.org/en/reference/procedural-macros.html
40. Rust Macros System https://dev.to/godofgeeks/rust-macros-system-1661
41. Procedural macros under the hood: Part II https://blog.jetbrains.com/rust/2022/07/07/procedural-macros-under-the-hood-part-ii/
42. Writing and Optimizing Custom Derives with Rust's ... https://technorely.com/insights/writing-and-optimizing-custom-derives-with-rusts-proc-macro-for-code-generation
43. 使用Wasm插件扩展ASM监控指标的维度信息 - 阿里云文档 https://help.aliyun.com/zh/asm/sidecar/use-the-wasm-plug-in-to-extend-the-dimension-information-of-asm-monitoring-metrics
44. Rust + WebAssembly：为大型语言模型生态构建基础设施 https://zhuanlan.zhihu.com/p/667210058
45. Short answer: No. | Dan Lorenc https://www.linkedin.com/posts/danlorenc_can-wasm-replace-containers-activity-7262457541043843072-DVCG
46. GitHub Actions supply chain attack spotlights CI/CD risks https://www.techtarget.com/searchitoperations/news/366621078/GitHub-Actions-supply-chain-attack-spotlights-CI-CD-risks
47. Building Scalable AI Agents: A Deep Dive into Decoupled ... https://caseywest.com/building-scalable-ai-agents-a-deep-dive-into-decoupled-tools-with-adk-mcp-and-cloud-run
48. First Week learning rust https://dev.to/enyelsequeira/first-week-learning-rust-4j7i
49. MemTrust: A Zero-Trust Architecture for Unified AI Memory ... https://arxiv.org/html/2601.07004v1
50. 医疗美容使用的非激光光源设备IEC 60601-2-57测试 https://www.ctnt-cert.com/xwgg/19798.html
51. Managing security evidence in safety-critical organizations https://www.sciencedirect.com/science/article/pii/S0164121224001274
52. AI Incident Response Framework for Safety-Critical Systems https://www.linkedin.com/posts/suwoongsik_ai-activity-7413068724259348480-hMP6

53. 9.2 Release Notes | Red Hat Enterprise Linux | 9 https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html-single/9.2_release_notes/index
54. Planet Mozilla https://planet.mozilla.org/rss10.xml
55. Model-Based Reasoning in Science and Technology https://www.researchgate.net/profile/Jeffrey-White-12/publication/237996791_Understanding_and_Augmenting_Human_Morality_An_Introduction_to_the_ACTWith_Model_of_Conscience/links/644b9fcc809a535021363c81/Understanding-and-Augmenting-Human-Morality-An-Introduction-to-the-ACTWith-Model-of-Conscience.pdf
56. Fundamentals and Practical Implications of Agentic AI https://arxiv.org/html/2505.19443v1
57. How to Avoid CI/CD Supply Chain Incidents with GitHub ... https://www.linkedin.com/posts/vipulgupta2048_if-your-organization-uses-github-actions-activity-7366807774292299776-gd91
58. CORTEX: Composite Overlay for Risk Tiering and ... https://arxiv.org/html/2508.19281v1
59. IISO 42001 vs NIST AI RMF: How to Choose the Right ... https://www.hicomply.com/blog/iso-42001-vs-nist-ai-rmf
60. EU AI Act NIST AI RMF and ISO 42001 Compared https://www.softwareseni.com/eu-ai-act-nist-ai-rmf-and-iso-42001-compared-which-framework-to-implement-first/
61. Using PL/Rust to write PostgreSQL functions in the Rust ... https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/PostgreSQL.Concepts.General.Using.PL_Rust.html
62. Release 7.13.0 Keylime Developers https://keylime.readthedocs.io/_/downloads/en/latest/pdf/
63. Guarding Our Vital Systems: A Metric for Critical Infrastructure ... https://pmc.ncbi.nlm.nih.gov/articles/PMC12349531/
64. Electroencephalography (EEG) for psychological hazards ... https://www.sciencedirect.com/science/article/pii/S0926580525003863
65. Wearables in ADHD: Monitoring and Intervention—Where Are ... https://pmc.ncbi.nlm.nih.gov/articles/PMC12468562/
66. Amazon Bedrock AgentCore - Developer Guide https://docs.aws.amazon.com/pdfs/bedrock-agentcore/latest/devguide/bedrock-agentcore-dg.pdf
67. A Novel Zero-Trust Identity Framework for Agentic AI https://arxiv.org/html/2505.19301v1
68. Arxiv今日论文| 2025-11-03 http://lonepatient.top/2025/11/03/arxiv_papers_2025-11-03

69. LoCoBench-Agent: An Interactive Benchmark for LLM ... https://arxiv.org/html/2511.13998v1

70. P505272-3c838d61-08f9-41ac-a199- ... https://documents1.worldbank.org/curated/en/099122125121523841/txt/P505272-3c838d61-08f9-41ac-a199-59990baf9b6b.txt

71. (PDF) Nova Academia: A Blueprint and Prototype Platform ... https://www.researchgate.net/publication/396515920_Nova_Academia_A_Blueprint_and_Prototype_Platform_for_a_Federated_Transparent_and_Democratic_Open-Science_Infrastructure

72. Full Report | PDF | Creative Commons License https://www.scribd.com/document/976326420/Full-Report

73. Experts for Nodejs-Mobile-React-Native Plugins Readme https://www.linknovate.com/search/?query=nodejs-mobile-react-native%20plugins%20readme

74. 9.1 Release Notes | Red Hat Enterprise Linux | 9 https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html-single/9.1_release_notes/index

75. Index of all Modules — Ansible Community Documentation https://docs.ansible.com/projects/ansible/latest/collections/index_module.html

76. Fundamentals and Practical Implications of Agentic AI https://arxiv.org/pdf/2505.19443

77. ICT in Motion: The Next Wave of AI Integration (2025) https://www.cisco.com/content/dam/cisco-cdc/site/m/ai-workforce-consortium/documents/2025-ai-workforce-consortium-full-report.pdf

78. näkeva ai : post # 1 https://www.linkedin.com/pulse/n%C3%A4keva-ai-post-1-imran-bashir-meduc

79. glove.6B.100d.txt-vocab.txt https://worksheets.codalab.org/rest/bundles/0xadf98bb30a99476ab56ebff3e462d4fa/contents/blob/glove.6B.100d.txt-vocab.txt

80. Enhancing driver attention and road safety through EEG ... https://www.sciencedirect.com/science/article/pii/S1568494624010949

81. Transforming Healthcare: Intelligent Wearable Sensors ... https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.202500412

82. Proceedings of the Annual Meeting of the Cognitive ... https://escholarship.org/uc/cognitivesciencesociety

83. Planet Mozilla https://planet.mozilla.org/

84. Protecting Brain Privacy in the Age of Neurotechnology https://www.researchgate.net/publication/384971350_Protecting_Brain_Privacy_in_the_Age_of_Neurotechnology_Policy_Responses_and_Remaining_Challenges

85. cmnt_vocab.txt https://www.cs.cmu.edu/~ark/blog-data/data/blog_data_v1_0/dk/hbc_data/data/cmnt_vocab.txt
86. EU AI Act vs NIST AI RMF vs ISO 42001 vs US AI Legislation https://www.linkedin.com/posts/hoshedarriskmanagement_ciso-iso42001-nist-activity-7411207041391714305-VMEd
87. Verifiable Credentials Working Group F2F, 2nd day https://www.w3.org/2017/vc/WG/Meetings/Minutes/2022-09-16-vcwg
88. Selective Disclosure Approaches in Self-Sovereign Identity https://ieeexplore.ieee.org/iel8/6287639/11323511/11316648.pdf
89. un-transparency-protocol.pdf https://untp.unece.org/un-transparency-protocol.pdf
90. Arxiv今日论文| 2025-12-11 http://lonepatient.top/2025/12/11/arxiv_papers_2025-12-11
91. 333333 23135851162 the 13151942776 of 12997637966 ftp://ftp.cs.princeton.edu/pub/cs226/autocomplete/words-333333.txt
92. UC Berkeley https://escholarship.org/content/qt5cs11779/qt5cs11779.pdf
93. ETHICALLY ALIGNED DESIGN A Vision for Prioritizing ... https://www.researchgate.net/publication/378975517_ETHICALLY_ALIGNED_DESIGN_A_Vision_for_Prioritizing_Human_Wellbeing_with_Artificial_Intelligence_and_Autonomous_Systems
94. EU AI Act vs NIST AI RMF: A Comparison https://www.linkedin.com/posts/janachander_eu-euai-nist-activity-7373302098932834304-FhkB
95. ERCIM News 139 https://hal.science/hal-04769799v1/file/ERCIM_NEWS_139.pdf
96. Introducing BMSSP: A New Algorithm for Shortest Path ... https://www.linkedin.com/posts/reuvencohen_for-more-than-forty-years-dijkstras-activity-7367271384869236737-G0s8
97. Nucleus Research - LCAP Technology Value Matrix 2025 https://www.oracle.com/a/ocom/docs/database/nucleus-rsearch-lcap-technology-value-matrix-2025.pdf
98. Ethics and Innovation https://www.wipo.int/edocs/pubdocs/en/wipo_pub_1072.pdf
99. Chapter 4. New features | 9.2 Release Notes https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/9/html/9.2_release_notes/new-features
100. 8.9 Release Notes | Red Hat Enterprise Linux | 8 https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/8/html-single/8.9_release_notes/index
101. Diagnostics 15 02359 | PDF | Attention Deficit Hyperactivity ... https://www.scribd.com/document/976298043/Diagnostics-15-02359
102. Low Price Neural Sensors AI Vision Robot https://www.alibaba.com/showroom/neural-sensors.html

103. (PDF) Standardized Threat Taxonomy for AI Security ... https://www.researchgate.net/publication/397906127_Standardized_Threat_Taxonomy_for_AI_Security_Governance_and_Regulatory_Compliance
104. 90-Days Roadmap to Implementing NIST AI RMF in Your ... https://www.linkedin.com/pulse/90-days-roadmap-implementing-nist-ai-rmf-your-rakesh-kumar-mehoc
105. ISO/IEC 42001 Clause 8 Operation – Guidance & Best Practices https://cyberzoni.com/standards/iso-42001/clause-8/
106. ISO/IEC 42001:2023(en), Information technology https://www.iso.org/obp/ui/en/#!iso:std:81230:en
107. AI Lifecycle Risk Management - ISO - IEC 42001 https://www.scribd.com/document/935271242/AI-Lifecycle-Risk-Management-ISO-IEC-42001-2023-for-AI-Governance-AWS-Security-Blog
108. ISO/IEC 42001:2023 for AI governance | AWS Security Blog https://aws.amazon.com/blogs/security/ai-lifecycle-risk-management-iso-iec-420012023-for-ai-governance/
109. AI Act Service Desk - Annex III - European Union https://ai-act-service-desk.ec.europa.eu/en/ai-act/annex-3
110. EU database for high-risk AI systems listed in Annex III https://ai-act-service-desk.ec.europa.eu/en/ai-act/article-71
111. ISO/IEC 42001:2023 - AI management systems https://www.iso.org/standard/42001
112. ISO/IEC 42001:2023(en), Information technology https://www.iso.org/obp/ui/es/#!iso:std:81230:en
113. Regulation (EU) 2024/1689 of the European Parliament and ... https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=OJ:L_202401689
114. Regulation - EU - 2024/1689 - EN - EUR-Lex - European Union https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng
115. ISO 42001 Explained | Full List Of Clauses And Controls https://cyberzoni.com/standards/iso-42001/
116. NIST AI Risk Management Framework for Security and ... https://www.linkedin.com/posts/delight-muromba-64b791172_artificial-intelligence-risk-management-framework-activity-7411467133362167808-bBex
117. The Academic Guide to AI Act Compliance https://hal.science/hal-05365570v1/file/The%20Academic%20Guide%20to%20AI%20Act%20Compliance%20%20-%202025%20-%20Ed.%20MHODAC%20%26%20CP.pdf
118. Subject Roles in the EU AI Act: Mapping and Regulatory ... https://arxiv.org/html/2510.13591v1

119. The NIST AI Risk Management Framework and Third-Party ... https://mitratech.com/resource-hub/blog/nist-ai-risk-management-framework-rmf/

120. A survey of artificial intelligence risk assessment ... https://www.trilateralresearch.com/wp-content/uploads/2022/01/A-survey-of-AI-Risk-Assessment-Methodologies-full-report.pdf

121. Actionable Guidance for High-Consequence AI Risk ... https://arxiv.org/pdf/2206.08966

122. Regulation - EU - 2024/1689 - EN - EUR-Lex - European Union https://eur-lex.europa.eu/legal-content/EN-DE/ALL/?uri=CELEX:32024R1689&from=EN

123. Regulation - EU - 2024/1689 - EN - EUR-Lex - European Union https://eur-lex.europa.eu/legal-content/EN-FR/TXT/?from=EN&uri=CELEX%3A32024R1689

124. Search results - EUR-Lex https://eur-lex.europa.eu/search.html?DB_MENTIONING=32024R1689&SUBDOM_INIT=ALL_ALL&DTS_SUBDOM=ALL_ALL&DTS_DOM=ALL&lang=en&type=advanced&qid=1747965913861&page=2

125. Regulation - EU - 2024/1689 - EN - EUR-Lex - European Union https://eur-lex.europa.eu/legal-content/EN-PL/TXT/?uri=CELEX:32024R1689

126. Rules for trustworthy artificial intelligence in the EU - EUR-Lex https://eur-lex.europa.eu/EN/legal-content/summary/rules-for-trustworthy-artificial-intelligence-in-the-eu.html

127. L_202402847EN.000101.fmx.xml - EUR-Lex https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=OJ:L_202402847

128. EUROPEAN COMMISSION Brussels, 19.11.2025 ... - EUR-Lex https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:52025PC0836

129. EUR-Lex https://eur-lex.europa.eu/