CHAI
COALITION FOR HEALTH AI

Best Practice Guide

Agentic AI: Healthcare-specific protocol language considerations

TABLE OF CONTENTS

Agentic AI: Healthcare-Specific Protocol Language Considerations

For Developers & Implementers

What Does This Guide Include?

Use case specific Best Practice Guides (BPGs) provide high-level, industry- and consensus-defined insights and recommendations for the application of responsible AI principles to a specific use case. This guide focuses on agentic AI in healthcare: autonomous and semi-autonomous AI agents that use tools, maintain memory and state, and may coordinate as multi-agent systems to carry out healthcare tasks. Agentic AI is evolving rapidly; this BPG reflects the current state of the technology and industry as of June 2026.

This BPG is structured somewhat differently from earlier guides in the series. Rather than discrete best practice statements, the recommendations here take the form of protocol language considerations: healthcare-specific capabilities ("healthcare extensions") that build on top of existing sector-agnostic agent protocols (e.g., A2A, MCP) and interoperability standards (e.g., FHIR, HL7 v2). These considerations describe what an agent protocol should additionally express, constrain, or document to meet healthcare needs around responsible AI. Where possible, considerations are framed around the problem to be solved and the characteristics a sound solution should have, rather than prescribing a single implementation, since specific technical approaches may change quickly while the underlying principles remain durable.

This BPG begins with General Health Agent Considerations that apply broadly across healthcare agents and then presents considerations specific to use case scenarios the work group examined. Within each section, best practice considerations are organized by persona (Developer or Implementer), and each is tagged with the responsible AI principle area(s) it most advances.

Note

While this BPG focuses on best practices for the design, deployment, and governance of agentic AI systems, organizations should recognize that agentic architectures represent only one component of the broader AI implementation landscape. Many healthcare use cases are more appropriately addressed through deterministic tools, services, or "skills" that perform well-defined functions without autonomous planning or decision-making. The decision to implement a capability as an agent, a skill, or a hybrid orchestration of both should be informed by factors such as workflow complexity, required autonomy, risk tolerance, auditability, and validation requirements. In many cases, the most effective and trustworthy solutions combine agentic orchestration with validated skills that execute specific tasks in a predictable and governable manner.

Use Case Description

The aim of this work group is to define healthcare-specific capabilities that extend existing protocol language (open-agent, general-purpose, or proprietary) to support safe and interoperable healthcare AI agents. Sector-agnostic protocols provide the baseline technical infrastructure for agent communication, coordination, and tool interaction, but they do not encode healthcare-specific meaning, constraints, or accountability. The work group develops consensus-defined protocol language considerations that translate healthcare requirements into standardized, domain-specific capabilities, documented through a model/agent card framework that may capture compliance posture (e.g., HIPAA-aligned logging), operational constraints (e.g., human-in-the-loop requirements), and decision-making boundaries (e.g., no autonomous clinical decisions).

For each representative use case scenario, the work group defined a high-level workflow, decomposed it into the discrete actions an agent should perform, mapped those actions to existing protocol language where applicable, identified gaps where general-purpose protocols are insufficient for healthcare, and drafted illustrative healthcare-specific protocol language to address them. Representative gaps included the lack of a standard for patient identity confidence, undefined boundaries for an agent's administrative authority, the absence of purpose-of-use context for data access, and medical-record attribution for AI-assisted updates.

Use Case Scenarios

The Agentic AI work group examined five unique, high-priority use case scenarios: General Health Agent Considerations, Clinical Decision Support, Voice-Enabled Patient Scheduling, Contract Search, and Pre-Visit Prep & Summarization. Additional scenarios will be incorporated as the work group continues. A short description of each scenario is included below:

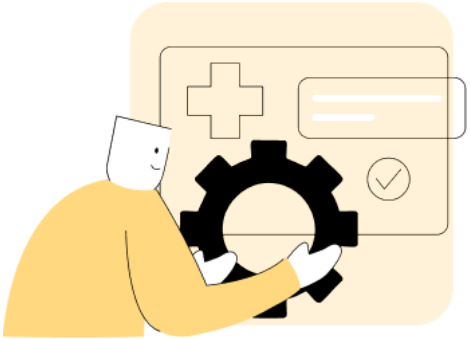
- **General Health Agent**: Cross-cutting practices that apply to agentic AI across healthcare workflows regardless of the specific application. This category covers agent scoping and architecture, read/write access guardrails and human verification, provenance and audit logging, multi-source data reconciliation, evaluation and benchmarking frameworks, escalation and failure-mode handling, and ongoing monitoring.

- **Clinical Decision Support**: AI agents that assist clinicians by assembling patient context, generating evidence-informed recommendations, and supporting decisions across the encounter lifecycle. This category emphasizes standardized, interoperable transfer of patient context between AI components, stateful orchestration with the EHR, transparent and explainable clinical reasoning, and low-burden mechanisms for capturing clinician feedback.
- **Voice-enabled Patient Scheduling:** Conversational voice agents that handle inbound and outbound patient scheduling by interpreting natural-language needs, mapping them to the correct visit types, and booking against operational rules. This category focuses on reliable caller identity verification, equitable speech recognition across accents and dialects, clear escalation triggers for sensitive or clinically urgent calls, and ongoing monitoring of accuracy and call outcomes.
- **Contract Search:** AI agents that retrieve and reason over contract and policy documents to answer user queries. This category stresses configurable guardrails tailored to the deployment context, safeguards to ensure only current and authoritative document versions are used, and user education on the difference between semantic and literal keyword retrieval.
- **Pre-visit Prep & Summarization:** AI agents that discover, retrieve, and synthesize patient information from across systems ahead of a visit. This category addresses cross-system data discovery, semantic reconciliation of conflicting data with proper temporal anchoring, interval summaries with fine-grained and verifiable citations, and tailoring summary content to the clinician's role and use case.

The above use case scenarios were prioritized by the work group for v1.0 of the Best Practice Guide and are not exhaustive of the full landscape. Additional use case scenarios, such as revenue cycle and administrative workflows (e.g., prior authorization, claims follow-up), referrals and care coordination, and pharmacy access (e.g., benefit verification), will be considered in future iterations of this Best Practice Guide.

Who Is This Guide For?

Developers: individuals involved in the software development process, including requirements gathering, design, coding, testing, and maintenance of software applications (derived from IEEE, 12207:2017)



Implementers: individual(s) responsible for the procurement, deployment, and/or overall realization of a system or component in accordance with a specified design (derived from IEEE 829 and IEEE 730)

How Is This Guide Organized?

Healthcare-specific protocol language considerations are grouped by the specific use case scenarios examined. Within each use case scenario, considerations are organized by persona. Each consideration is tagged with one or more responsible AI principle areas.

Listening In: A Summary of Challenges & Insights

Across work group sessions, members worked through concrete healthcare use case scenarios contributed by developers and implementers. A consistent picture emerged of where today's agent protocols and interoperability standards fall short for healthcare, and where the group found pragmatic footing. The following challenges and insights informed the best practice considerations that follow.

Challenge/Insight #1: Agent protocols are not currently built for healthcare, and healthcare standards are not built for agents.

Members repeatedly observed that sector-agnostic protocols such as MCP and A2A are not optimized for healthcare, while FHIR and HL7 are not optimized for AI. FHIR can tell you an oxygen value exists without telling you whether it is clinically appropriate to use. Much of the group's work is building this bridge: how to handle PHI inside agent protocols, how to trace AI-based decisions, and how agents should hand-off information to one another.

Challenge/Insight #2: There is no standard way for agents to pass, preserve, or validate patient context across a multi-agent handoff.

As scenarios decomposed into specialized agents (patient-context, evidence-discovery, validation), members noted there is no healthcare standard for what context must be retained between them, and material reasoning can fail to propagate downstream. Retrieval responses also arrive in inconsistent shapes depending on whether data came from FHIR, a CDA document, an HIE transaction, or an MCP server, pushing teams toward a normalized “patient memory” abstraction.

Challenge/Insight #3: Evaluation lacks a gold standard and the tools meant to scale evaluation do not reflect real users.

“LLM-as-judge” was acknowledged as valuable for scale but also has shortcomings that need to be overcome: the judge can inherit the same biases, prompts, and limitations as the agent it evaluates, so evaluation is best performed "out-of-band" (e.g., by an independent evaluator such as a different model, agent, or deterministic checker that does not share the agent's prompts or training). Manual checks are themselves error-prone, and routine clinical actions may not represent objective ground truth. Human review remains essential but does not scale since clinician evaluator time is scarce and expensive.

Challenge/Insight #4: Autonomy boundaries and decision thresholds are often difficult to observe, and human oversight depends on reviewers having the knowledge and confidence to

evaluate agent outputs.

As agents become more autonomous, decisions may be harder to trace and verify through standard logs, increasing the need for clear guardrails, accountability, and auditability. Participants also noted that growing autonomy can widen AI literacy gaps, making effective human oversight more challenging.

Challenge/Insight #5: Transparency, traceability, and post-deployment monitoring remain areas of active development.

Organizations often focus governance efforts on intake and approval processes, while monitoring AI systems in production requires distinct capabilities that are still maturing. Standard logs do not typically capture agentic reasoning. Participants noted that transparency (disclosing AI involvement), traceability (understanding how outputs were generated), and ongoing observability are related but separate needs, each requiring dedicated approaches and supporting infrastructure.

Challenge/Insight #6: Low-code/no-code agent builders are outpacing governance, and the underlying technology often cannot deliver the promised capability.

One health system found roughly three times as many agents built in approved low-code/no-code tools as the use cases that had passed formal governance. Separately, broadly capable agents repeatedly broke against data realities: clinician scheduling preferences and decision-tree logic often live in email, spreadsheets, and front-desk knowledge rather than any interoperable API.

Challenge/Insight #7: Voice agents face healthcare-specific safety, identity, and bias concerns that general protocols don't currently anticipate.

Members shared the sentiment that "every phone call is an edge case." Identity verification is fragile when the caller is not in the system, is a caregiver rather than the patient, or when caller ID is spoofed. Automatic speech recognition is markedly worse for non-Western names, hyphenated names, accents, and non-English languages. Real-time output filtering adds latency that degrades conversation, so some harmful utterances can only be caught after the fact. It is important to note that these challenges are framed around human callers. This Agentic AI Work Group also flagged emerging agent-to-agent (AI-to-AI) voice interactions (e.g., where the caller is itself an automated system) as a distinct future consideration that may require detection and audit patterns not yet addressed.

Challenge/Insight #8: Curated safety triggers and escalation rules are commonly used to manage risk, but what constitutes a "red flag" is highly context dependent.

Organizations often define specific triggers that prompt human review or intervention. Because the same signal may represent routine care in one setting and a potential safety concern in another, these triggers must be tailored to the clinical context. Most organizations favor a conservative approach, escalating uncertain cases to a human reviewer to prioritize patient safety.

Challenge/Insight #9: Evaluation results are only meaningful when the evaluation methodology is clearly defined and reported.

Organizations should distinguish who conducted the evaluation, what was evaluated, when it occurred, and how performance was measured. Simulations, structured testing, and real-world deployment each

provide different insights and should not be treated as interchangeable. Transparent reporting of evaluation methods helps others understand the applicability and limitations of the results.

General Health Agent Considerations

Best Practice Considerations for Implementers

Define and Display the Agent's Operational Scope

Clearly define and display the agent's operational scope (e.g., patient-specific vs. workspace-wide) to the end user. For Developers, include intended context in metadata about the agent (a model/agent card or similar documentation) to help Implementers decide which agents to onboard for specific clinical tasks. Agent behavior and required tools change drastically depending on whether the scope is a single patient, a cohort, or general policy research; without clear scope definitions, users may misapply an agent's output.

Example

A Prior Authorization agent looks entirely different when assisting with a specific patient's template versus helping a user generally understand payer policies. A user guide or visual filter within the interface helps clinicians know exactly what data the agent can see and act upon.

Evaluate Whether a Multi-Agent Orchestration Pattern Is Appropriate

Evaluate whether a multi-agent orchestration pattern fits the specific healthcare use case, particularly when workflows involve distinct tasks with different data-access needs, computational loads, or privacy sensitivities. Decomposing a complex goal into specialized agents (often coordinated by an orchestration agent) may improve efficiency, reduce token usage, limit unnecessary exposure of patient data, and enable hybrid approaches combining rule-based, classical ML, knowledge-graph, and generative methods. Weigh efficiency and data-minimization gains against the added complexity multi-agent setups introduce, and stay current as model capabilities evolve.

Define a Standardized Evaluation Governance Framework

Define a standardized evaluation framework that specifies roles and responsibilities, the types of interactions evaluated, the stage of system maturity, evaluator perspectives, and measurable performance criteria (e.g., safety, latency, reliability). Implementers should establish internal governance processes for approval and monitoring, while Developers should document and communicate the intended user persona(s), stage of development (e.g., pilot, limited release, production), evaluated interaction types, and known performance characteristics to support institutional review.

Example

One member organization standardized evaluation by defining the "5 W's": who evaluates (institutional reviewers vs. vendor testing teams), what interaction type is assessed, when in the lifecycle the evaluation occurs, which user persona is represented, and which measurable criteria (e.g., latency, safety, reliability) apply.

Structure Early-Stage Evaluation as a Human-in-the-Loop Assessment

Structure early-stage evaluation as a human-in-the-loop assessment whose goal is to calibrate the threshold for human intervention. Consider starting with high oversight (a clinician reviews all agent actions) and reduce oversight only as the agent meets pre-defined, measurable exit criteria informed by the workflow's risk profile and applicable medical device regulatory standards. Conduct the evaluation with a representative stakeholder group over a time-bound window, verify that the agent respects autonomy guardrails (e.g., requiring user confirmation for actions that should not be autonomous), test behavior in the presence of other autonomous agents where applicable, and document failure modes so the agent and its guardrails can be adjusted. Automated methods should augment, not replace, human review.

Derive Safety Metrics from Human-Supervision Analogues and Agent-Specific Failures

Define safety metrics for agentic workflows through guided sessions that ask clinicians and other stakeholders (nursing, IT, billing, operations) how they would supervise a staff member performing the agent's autonomous tasks. The same sessions should also ask, "what could this agent get wrong that a human almost certainly wouldn't?" to surface agent-specific failures—such as silently proceeding when uncertain rather than pausing to escalate, or acting on implausible content. Pay particular attention to scenarios where agentic action speed exceeds human reaction time, and translate the outputs into clearly defined, agreed-upon quantitative metrics.

Example

An Implementer may ask "How would you know a staff member was doing a sufficient job?" to translate qualitative supervision into quantitative benchmarks, paired with "What could this agent get wrong that a staff member almost certainly wouldn't?" to surface agent-specific risks.

Solicit failure modes from each stakeholder within their own domain of expertise (e.g., clinical staff for clinical failure modes and technical staff (e.g., cloud or integration engineers) for infrastructure failure modes such as an API failure, since no single role can anticipate every type of failure.

Separate Workflow Integration from ROI, and Anchor ROI to the Bottleneck

Perform an end-to-end map of the workflow to identify the primary operational bottleneck, then define ROI metrics anchored to that bottleneck rather than to the efficiency of the agent's isolated task. An agent may integrate smoothly into a workflow yet deliver negligible ROI if it does not address the primary bottleneck in the patient journey, so workflow integration and ROI must be evaluated independently and in the right order. As part of this mapping, assess whether the underlying workflow

should be redesigned or retired before introducing an agent. Automating a low-value or avoidable workflow adds cost and "agent clutter" without improving outcomes.

Example

Evidence on ambient scribes illustrates this distinction: documentation time savings may be smaller than the time required for an additional patient visit, so a solution can improve a measurable sub-task without moving the bottleneck that actually constrains throughput.

Adopt a Tiered, Deal-Breaker-First Evaluation Framework

Adopt a tiered evaluation framework that begins with a focused, time-bound deal-breaker review—starting with the failure mode that is both most likely and easiest to evaluate (e.g., UI/usability, workflow fit, or safety risks)—before moving to a longer trial period (e.g., 60-90 days) for efficiency. When safety is the appropriate first gate, conduct a dedicated safety review (e.g., 2-4 weeks) specifically checking for failure modes, edge cases, and risk of harm. If a solution fails on usability or workflow fit first, a formal safety evaluation may not be needed.

Example

One health system ran a phase 1 for ambient scribes in which 20 evaluators reviewed 200 notes specifically for harm that might occur if the AI's output were left unaltered by a clinician.

Codify Non-Negotiable Transparency and Safety Requirements Before Vendor Evaluation

Define and codify non-negotiable transparency and safety requirements before entering any collaborative or tiered evaluation arrangement with a vendor. At minimum, all Implementers—regardless of size or maturity—should obtain documented evidence of the vendor's internal quality processes, known failure modes, and training-data characteristics relevant to the intended patient population. Above this floor, the depth and form of vendor evaluation may be scaled to organizational capacity and AI maturity.

Examples

1. *Large health systems with mature AI evaluation capacity may request performance dashboards evaluated against local patient demographics (e.g., social determinants of health) and consider publishing those evaluations as community benchmarks that less-resourced systems can align to. Some vendors offer feedback APIs that can then translate into dashboards and metrics about the satisfaction of the user base.*
2. *Medium health systems that lack capacity for independent analysis may define their risk tolerance around population-level gaps not covered by community assessments and align to evaluations published by comparable organizations rather than accepting raw data tranches for independent analysis.*
3. *Small health systems may find the most viable path through co-development partnerships with early-stage vendors (e.g., jointly designing dashboards and evaluation criteria in exchange for access to an underserved patient population that the vendor otherwise couldn't reach).*

It is important to note that performance dashboards may be beneficial to both large and small health systems. A practical pattern is for larger systems to participate in an initial partnership and metric-definition work and then share the resulting dashboards with smaller systems that may lack capacity to build their own.

Best Practice Considerations for Developers

Ensure Observability While Protecting Intellectual Property

Implement mechanisms to ensure observability—such as logging key decision steps, system interactions, and tool dependencies—to enable retrospective review. These mechanisms may be maintained internally to protect IP while still enabling appropriate transparency to Implementers through summarized explanations or structured reasoning artifacts that do not expose sensitive details. Complex AI workflows can otherwise obscure how a specific conclusion was reached.

Note

Gather the agent's reasoning trace or decision metadata for each tool call for internal audit, while selectively surfacing explanation summaries or traceable outputs to health-system partners in a way that balances observability with IP protection.

Best Practice Considerations for Developers & Implementers

Use Out-of-Band Evaluation to Avoid Circular "LLM-as-Judge"

Use out-of-band evaluation frameworks where the judge is meaningfully different from the agent being evaluated (e.g., a model purpose-built and tested for evaluation, an independent agent, or a deterministic checker). "LLM-as-judge" can become circular if the judge inherits the same biases, prompts, or limitations as the agent. Incorporate objective metrics such as tool-usage checks to verify whether an agent correctly invoked required tools, and complement them with automatic adversarial challenges.

Note

Objective checks should not replace human review. Out-of-band expert evaluators remain essential for capturing clinical nuance. Monitoring tool calls can introduce determinism into otherwise non-deterministic outputs: if an agent answers a question about a patient's age without ever invoking the "patient age" tool, that is an objective failure a human or judge-agent can flag.

Request Human Attestation Before Committing Agent Output to the EHR

User interfaces should request an authenticated human attestation or co-signature from an authorized clinician in any situation where agent-generated output is presented as human-authored or where there is an expectation that a human performed or reviewed the work, before that output is committed to the primary EHR database. Where AI-generated content is added to the record without a human in (or on) the loop, it should instead be clearly labeled as AI-generated rather than requiring a co-signature for every entry. The system should clearly distinguish the specific data points the agent has added or modified—distinguishing them from pre-existing record content—so reviewers can more quickly verify changes without re-reading the full record. This guards against safety risks when clinicians cannot easily distinguish agent-modified data from existing content.

Continuously Monitor Performance Using Local Data

Establish a structured feedback and monitoring process that continuously evaluates performance using local organizational data wherever feasible, to assess real-world performance across specific patient populations and care settings. Incorporate findings into system updates or governance decisions to maintain alignment with intended clinical or operational objectives. Where direct clinical SME involvement is resource-constrained, adopt a tiered approach—such as periodic chart audits or cross-functional review committees—to sustain oversight in a scalable manner.

Example

One member organization implemented a closed-loop feedback system that uses continuous calibration to tune the AI's architecture and response logic based on performance data, evaluated against site-specific datasets rather than stock or benchmark libraries alone.

Scale Evaluation with Automation Validated Against Human Review

Incorporate scalable evaluation mechanisms—such as automated review systems—while routinely validating their outputs against human assessment to ensure reliability and calibration over time, using pragmatic approaches that account for limited clinical SME bandwidth. Human review remains essential but is insufficient alone for large-scale or continuous deployments.

Example

One example is using LLM-as-a-judge or similar automated adjudication for scalable, real-time evaluation, supplemented by periodic targeted human audits for high-risk scenarios. Developers and Implementers should also collaborate on tooling that enables streamlined, opportunistic human review (e.g., embedded feedback buttons, lightweight rating workflows) within existing clinical systems.

Define and Document Decision-Authority Thresholds

Clearly define and document decision-authority thresholds, specifying when the system may act autonomously and when human review or escalation is required; adherence to these thresholds should be periodically audited. Autonomous agents often operate with "invisible" decision boundaries, such as fuzzy-match thresholds, which can lead to incorrect actions.

Note

Explicitly define which confidence levels allow autonomous action and which require escalation to a human, and document those boundaries so they can be reviewed and audited.

Catalog Failure Modes and Establish Fallback Protocols

Catalog all known failure modes and establish defined fallback protocols—such as service degradation or human handoff—to ensure safe system behavior under stress conditions. Resource permitting, validate these mechanisms through structured test scenarios that simulate identified failure conditions. Technical failures such as API timeouts or unexpected inputs can otherwise cause agents to produce hallucinated or incorrect decisions.

Note

Service degradation means intentionally reducing system functionality in a controlled way when something goes wrong, rather than allowing the system to fail unpredictably or produce unsafe outputs.

Evaluate Subjective Experience Dimensions Alongside Technical Accuracy

Consider evaluating subjective performance dimensions (e.g., professionalism, demeanor, clarity, and emotional appropriateness) alongside technical accuracy to support a positive and trustworthy end-user experience. The technical quality of an output (e.g., text-to-speech tone) directly affects non-functional but critical dimensions like patient empathy and trust. Organizations should operationalize dimensions appropriate to their use case and population, drawing on national or industry definitions where available, and may reflect organizational mission and values within system prompts. Take care to avoid over-personification of the AI.

Benchmark Against the Highest-Quality Available Reference Data

Establish transparent benchmarking processes that measure system performance against the highest-quality available reference data, while documenting methodology, data limitations, and relevant demographic context. Avoid relying solely on routine human actions or historical clinician decisions as validation targets without additional expert review, since manual checks are themselves error-prone and routine actions may not represent objective ground truth.

Make Agent Interactions Traceable to Encounter Elements

Healthcare-specific protocols should ensure that all AI agent interactions are traceable to the corresponding elements of the clinical encounter within the EHR. Each agent action, recommendation, or decision point should be attributable to a specific encounter context—including the patient, clinician, and phase of care—to support documentation integrity, audit tracing, and accountability.

> **Note**
> *Existing audit and logging frameworks were not originally designed to capture the granular, real-time interactions generated by AI agents in clinical workflows, and may require extension to meet documentation and regulatory requirements.*

Apply Role-Based Access Controls to Agent Memory

Apply role-based access controls to agent memory with the same rigor as data and tool access. Memory scopes (e.g., session, user, administrative) must be clearly delineated and isolated to prevent cross-contamination between patients, users, or contexts. Memory behavior should be fully documented—what is stored, at which scope, for how long, and who can read or modify it—so auditors, administrators, and clinicians can understand and verify memory boundaries. Without proper controls, memory becomes a significant risk for unauthorized data exposure.

Combine Real-Time and Longitudinal Monitoring

A monitoring framework should combine real-time logging and trace capture with ongoing longitudinal performance evaluation, calibrated to the agent's risk and scope of action. Before deployment, define the agent's baseline behavior and what "performing as intended" looks like, including the metrics and thresholds that constitute acceptable performance and the deviation patterns that should trigger review. Specify clear ownership for each layer—who interprets the data, who acts on issues, and the escalation path when thresholds are breached.

Set a Universal Governance Baseline for All Builders, Including Low-/No-Code

Establish a universal minimum governance baseline that applies to all AI builders regardless of technical expertise, and calibrate additional rigor to the use case's risk and scope of action rather than to the builder's background. Patient-facing criteria should apply equally whether an agent was built in code, low-code, or no-code. Define explicit criteria any agent must meet before production; run low-code/no-code platforms within organizationally constrained (and, where possible, sandboxed) environments with system-level security and compliance guardrails enforced on every agent. Higher-risk capabilities can be gated by required training or certification, while lower-risk capabilities may remain more accessible.

Normalize Ontologies Before Agents Reason Over Data

Implement ontology normalization as a preprocessing step before agents reason over retrieved data. Map lab identifiers to standard LOINC codes and apply equivalent normalization to other clinical concepts (e.g., medications to RxNorm, problems to SNOMED CT or ICD) where applicable. The same clinical concept frequently arrives in different representations across sources; without normalization, semantic retrieval accuracy drops even when the underlying data is correct. Normalization should occur upstream of agent reasoning rather than relying on the agent to infer equivalence at inference time.

Build Patient Data Protections into the Protocol

Data safety controls should travel with patient data through the full agent chain, which includes every tool call, agent-to-agent handoff, and MCP resource access. Include immutable, auditable purpose-of-use metadata on every data request (e.g., why the data is requested, for which patient, by which agent, on behalf of which human, and for what clinical or operational purpose). Enforce data minimization through field-level scoping, where the requesting agent specifies the elements it needs and the serving system filters accordingly. Propagate a consent-scope object through every handoff and request downstream agents to validate their intended use against the consent-scope before processing. Request every MCP server in the chain to declare its data handling posture (e.g., retention, secondary use, jurisdiction, encryption, deletion) in machine-readable metadata.

Example

For example, an agent generating a medication summary should not receive the patient's full psychiatric history if it is not relevant to the task. Under field-level scoping the EHR filters those elements out, even though the record contains them. Likewise, where a patient consents to agent-assisted clinical care but opts out of agent-driven population analytics, the consent-scope object carries that limit through the chain, and a downstream analytics agent must validate against it and decline the handoff.

Clinical Decision Support Considerations

Best Practice Considerations for Developers & Implementers

Pass and Validate Standardized Patient Context Between Agents

Healthcare-specific protocols should require AI agents to pass standardized key elements of patient context (e.g., patient identity, relevant dates, and clinical details) using established interoperability standards (e.g., FHIR, HL7 v2). Information passed between agents should be validated against canonical sources of truth to prevent propagation of misinformation; significant mismatches or errors should trigger a resolution process before proceeding, while minor discrepancies should be flagged rather than halt the workflow. The protocol should maintain a granular audit trail documenting each agent's inputs, logic, and outputs, and should account for differing PHI access permissions across agents.

Implement a Stateful Orchestration Layer Across the Encounter Lifecycle

Healthcare-specific protocols should implement a stateful orchestration layer that persists interaction and clinical context across the encounter lifecycle (e.g., pre-visit chart review and documentation, the encounter itself, post-visit research and documentation) so agents can maintain awareness of what has been discussed, addressed, or acted upon. Where applicable, align with existing FHIR-based mechanisms for real-time, bidirectional communication (e.g., FHIRcast, SMART Web Messaging) rather than building proprietary alternatives. The orchestration layer should provide transparency into context-storage parameters and maintain audit and provenance records of state changes to support clinical accountability.

Capture the Chain of Clinical Reasoning

Healthcare-specific standards should capture the chain of clinical reasoning—from the patient context and query generation, to the evidence and guideline resources consulted and their dates, to any associated clinical reasoning and confidence levels. This transparency is critical for building safe AI systems and for explainability.

Note

Because AI outputs can feel like a "black box," tracing decisions back to specific medical literature or model versions is essential for Developer assessment and clinical safety. Related work includes the HL7 AI Transparency on FHIR Implementation Guide.

Capture Low-Burden User Feedback on Recommendations

Healthcare-specific protocols should allow users to indicate broadly why they made a decision (e.g., distinguishing a different but valid clinical decision from an erroneous model recommendation, alongside

reasons such as timing, “does not apply,” or competing demands) and should capture, where feasible, the patient factors influencing the decision so bias can be detected. Existing specifications such as CDS Hooks may serve as a foundation.

Note

Feedback mechanisms should minimize clinician burden. Prioritize passive signals (e.g., edits to ambient documentation output) over interruptive prompts, and avoid pop-ups for routine disagreements. Treat collected feedback as a learning opportunity (“learning from variance”), and clarify how feedback affects accountability for outcomes on the record.

Additionally, specify who is responsible for monitoring this feedback and acting on it (e.g., the implementing organization's governance committee, the vendor, or both), and clarify whether/how vendors receive post-deployment input so recommendations can be improved over time.

Voice-enabled Patient Scheduling Considerations

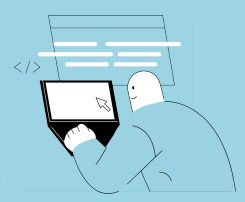
Best Practice Considerations for Developers

Provide a Baseline Library of Validated Escalation Triggers

Provide a baseline library of validated escalation triggers (e.g., red flags derived from patient natural language) covering both clinical urgency and administrative or social-determinant signals, with the two types clearly distinguished so they route to appropriate resources. Treat this baseline as a starting point rather than an exhaustive list: health systems should be able to extend it with population-specific triggers (e.g., oncology centers adding cancer-specific symptoms), and the baseline should be transparent to the deploying health system so it can validate applicability to its patient population. Align with existing specialty or professional-society guidelines where applicable.

Note

Examples of red flags include clinical urgency signals (“I've been having chest pain since this morning”), social-determinant signals (“I don't have a way to get there”), and confusion or distress signals (“I don't understand what you're asking,” repeated).



Best Practice Considerations for Developers & Implementers

Automate Scheduling Decisions Only on Reliable, Machine-Consumable Rules

Agents should automate scheduling decisions only when the governing rules and constraints (e.g., clinician scheduling constraints, visit eligibility criteria, operational preferences) are maintained in a reliable, reviewable, and machine-consumable form, with defined ownership and a clear update process. When these conditions are not met, automation should be deferred or scoped to advisory output rather than executing on stale or ambiguous inputs. This pattern extends to other preference-driven clinician workflows (e.g., documentation conventions, pre-visit summaries).

Example

One member organization is developing a clinician preference model for documentation and pre-visit summaries, recognizing that typical conversational memory does not reliably encode such preferences or meet clinicians' consistency expectations.

Rely on Authoritative, Version-Controlled Scheduling Configuration

Agents should rely on authoritative, version-controlled sources of scheduling configuration (e.g., clinician-department-visit type mappings, eligibility logic, and related rules). Where existing FHIR or HL7 v2 profiles cover relevant data, align with those profiles rather than building proprietary alternatives. For configuration data not yet covered by accepted standards, machine-readable formats such as structured text or configuration files are acceptable provided they are versioned, auditable, and treated as authoritative.

Map Patient Needs to Visit Types with Auditable Triage Logic

Implement an auditable approach for mapping patient needs, symptoms, or requests to appropriate visit types using clinical context and triage logic. The mapping must include explicit confidence thresholds, exception handling, and escalation to human review when patient needs cannot be confidently classified, and should incorporate safety-net prompts for patients who lack the vocabulary to describe their concerns. Inference should not rely on sensitive biometric profiling and should comply with applicable transparency and data-protection rules.

Use Layered, Multi-Factor Soft Verification for Inbound Calls

Voice agents handling inbound healthcare calls should implement a layered, multi-factor soft verification approach that accommodates real-world variability in caller responses, with graceful retry logic before escalating to a live agent. Because the agent cannot pre-confirm who answered, relying on a single

data point produces unacceptably high failure rates, particularly across culturally and linguistically diverse populations.

Example

For example, use two-factor soft verification (e.g., date of birth plus the last four digits of a phone number or a zip code) and allow graceful retry before escalating to a live agent.

Detect and Accommodate Proxy Callers Early

Voice agents should introduce an early branching prompt near the start of every inbound call to detect and accommodate proxy callers (e.g., a parent, spouse, or caregiver). When a caregiver or guardian is identified, the verification and data-collection flow should dynamically adjust to capture the proxy's identity alongside the patient's identifying information, maintaining both security and continuity. Failing to anticipate this scenario early can lead to dead-ends and poor caller experience.

Example

For example, an early prompt of "Are you calling on behalf of someone else?" can capture the proxy relationship and then adjust verification to caregiver identity plus patient date of birth.

Treat Easily Spoofed or Copied Signals (e.g., Caller ID) as a Non-Factor for PHI Access

Voice agents should treat any signal that is easily spoofed or copied (e.g., caller ID) as a non-factor in any PHI-gated interaction. Verification should rely exclusively on knowledge-based authentication (information only the patient or authorized party would know) regardless of the number displayed. Inbound calls can originate from spoofed numbers, making phone-number matching an insufficient and legally untenable basis for granting access to PHI. Anomalous patterns should be logged and flagged for fraud review rather than silently passed through.

Use Phonetic Matching for Names Across Languages

Voice agents should apply phonetic similarity scoring rather than exact-match logic when verifying patient names, and should offer a spell-confirm prompt rather than asking patients to simply repeat themselves. Confidence thresholds should be calibrated to trigger human review rather than outright rejection. Automatic speech recognition is disproportionately less accurate for names from non-native-English-speaking regions, and hard-match rejection logic compounds this inequity.

Example

For example, use phonetic similarity scoring, prompt the agent to spell-confirm rather than repeat, avoid hard-match rejection, and use a fuzzy threshold with a human-review trigger at low confidence.

Support Multiple Accent and Dialect Profiles

Voice agents should support multiple regional and demographic accent profiles, with the ability to offer a language or dialect preference option at the start of a call. Extraction accuracy should be tracked and audited by demographic segment, and confidence thresholds adjusted per profile to maintain consistent performance across caller populations. Uniform thresholds otherwise produce a system that performs well for some groups and poorly for others, introducing inequity at the point of scheduling.

Note

Beyond accent and dialect, voice agents should accommodate patients with hearing or speech limitations or other barriers to voice interaction. Voice agents may offer alternative channels (text, web, or human callback) and ensure escalation paths do not assume fluent spoken interaction.

Maintain Distinct Inbound and Outbound Architectures

Voice agent platforms should maintain distinct architectural pathways for inbound and outbound call flows. Inbound pathways require broader intent-detection coverage, more graceful and fault-tolerant failure modes, and lower confidence thresholds before escalating to a live agent, reflecting the higher ambiguity of unsolicited, unstructured inbound interactions. Repurposing outbound architecture for inbound use has a high likelihood of producing brittle systems that escalate too quickly, misclassify intent, and generally fail under real-world conditions.

Enforce Immediate Escalation on Sensitive Disclosures

Voice agent deployments should maintain a clinical red-flag phrase library, developed and regularly reviewed by qualified clinical staff. Any detected trigger phrase must initiate an unconditional, immediate transfer to a live agent with no further AI-generated dialogue following the trigger. This escalation pathway should be treated as non-negotiable and exempt from optimization or exception logic. Patients may disclose clinically sensitive information (e.g., mental-health crises, domestic violence, or suicidal ideation) during what begins as a routine scheduling call, and AI agents have no safe path for managing such disclosures autonomously.

Note

For voice-to-voice models that do not expose an intermediate text layer, deployments should implement comparable mechanisms to recognize patient distress, emergencies, and other escalation triggers directly from the audio stream.

Monitor Across Multiple Operational Domains

Voice agent deployments should be governed by a continuous monitoring framework that tracks performance across multiple operational domains—including, but not limited to, escalation rate, scheduling completion rate, identity-verification failure rate, clinical red-flag detection accuracy, and demographic equity in outcomes. Monitoring thresholds should be defined at deployment and reviewed on a scheduled basis, with automated alerting when any tracked metric crosses a defined threshold. Monitoring limited to a single dimension (e.g., transcription confidence) will miss degradation occurring elsewhere.

Implement Multi-Tiered Clinical Urgency Detection

Voice agents handling scheduling calls should implement a multi-tiered clinical urgency detection framework, developed in partnership with qualified clinical staff, that distinguishes between organizationally defined response tiers. The symptom and phrase library used to trigger each tier should be clinically validated, regularly maintained, and scoped to the specific care context of the deployment (e.g., dental, primary care, behavioral health). The agent should never attempt to independently assess clinical severity beyond routing to the appropriate tier. Failing to differentiate urgency levels risks both over-escalation that burdens live-agent capacity and under-escalation of time-sensitive but non-emergent conditions.

Contract Search Considerations

Best Practice Considerations for Developers

Design Configurable, Role- and Purpose-Aware Guardrails

Architect agent configurations as modular, role- and purpose-aware policies that can be independently reviewed, versioned, and audited. Each business unit (e.g., Security, Compliance, Legal, PMO, Privacy) should receive its own system prompt, search scope, escalation rules, and data-isolation boundary, with least-privilege defaults and documented justification for any expansion of scope. All guardrail configurations should be traceable to a named accountable owner and subject to periodic review. Guardrail configurability cannot be retrofitted: a single generic configuration cannot uphold the distinct privacy, safety, and regulatory obligations that apply to different roles.

Implement Document Currency Controls as an Accuracy Safeguard

Build freshness warnings (e.g., a 30-day threshold), must-acknowledge gates, and upload-date metadata into the agent workflow, and surface document age and last-verified status alongside any agent response that relies on a document. Where feasible, integrate with authoritative systems of record so the agent reasons over canonical, version-controlled content rather than user-uploaded copies, and document the provenance of every source consulted. Stale data can lead to decisions based on outdated obligations, superseded terms, or retired policies.

Explain, in Plain Language, How the Agent Retrieves and Reasons

Provide transparent, accessible explanations of how the agent works and its known limitations, including onboarding materials, example prompts, and in-app query guidance that explain in plain language how results are retrieved (e.g., semantic reasoning vs. exact keyword match) and that users should verify results against source documents. This consideration highlights the risk of a mismatch between a user's mental model and the system's actual operation. For example, users unaware that an agent performs

semantic rather than keyword-literal retrieval may over- or under-trust results because the system's behavior does not match their mental model.

Note

Reminding users to verify results against source documents is not sufficient on its own and can become a liability-shifting disclaimer. The agentic solution should actively support validation, including helping the user understand not only what was retrieved but also what types of information may have been excluded.

Pre-visit Prep & Summarization Considerations

Best Practice Considerations for Developers

Enable Privacy-Preserving Cross-System Discovery and Retrieval

Recognizing that no comprehensive standard exists today, enable privacy-preserving cross-system discovery and retrieval through both forward-looking protocols and near-term operational controls: (1) define a patient-scoped source-discovery protocol that returns available endpoints, data types, and freshness metadata for a given patient, scoped to the minimum data necessary for the stated clinical purpose and enforced at the query level (e.g., via purpose-of-use parameters and query filters); (2) extend cross-network exchange frameworks (e.g., TEFCA) and authorization standards (e.g., FHIR Auth) to support delegated, purpose-bound agent access on behalf of a clinician; and (3) standardize a minimum healthcare response profile across transports while preserving source metadata and provenance so downstream users can identify the canonical record.

Note

Absent fully standardized discovery and retrieval protocols, developers should document which data sources the agent queries, what data types are available from each, and any known gaps, and surface this to the end user so clinicians can assess the completeness of the data informing the agent's output.

This documentation is especially important where data minimization or minimum necessary access is combined with summarization, since narrowing what the agent can see can introduce clinically significant omissions. Users need a reliable way to check their own mental model of what was searched, included, and excluded.

Enable Semantic Reconciliation and Temporal Anchoring

Enable semantic reconciliation and temporal anchoring across conflicting clinical data from multiple sources and formats (FHIR, CDA, PDF, unstructured notes). This may include: (1) defining a reconciliation output format that distinguishes confirmed, conflicting, and inferred facts, each carrying provenance and confidence metadata; (2) standardizing temporal anchoring—how to express a reference point and the interval window over which changes are computed; and (3) requiring reconciliation logic to be transparent and auditable to the reviewing clinician.

Generate Interval Summaries with Fine-Grained Citation

Enable interval summaries with fine-grained, viewable citation. This may include: defining a discrete, data-centric representation for interval/delta summaries with metadata for anchor point, time window, and source scope; treating AI-generated interval summaries as a fact-reconciliation layer that separates canonical facts from their display; standardizing data element citation that links each finding to specific source data (FHIR resource, CDA section, document page) and lets the user view the referenced data; displaying confidence or validation indicators alongside summary elements; and documenting the logic by which the summary was generated, including which sources were considered, how conflicts were resolved, and what was excluded. This should complement the HL7 AI Transparency on FHIR IG, recognizing that transparency and traceability are distinct needs.

Note

Citation through footnotes or highlighted spans may help but is easy to skip and may become impractical as the number of citations grows. Organizations may consider an intermediate claim- or fact-viewer that presents each claim with its supporting data clearly enough to reduce the validation burden for end users.

Example

One member organization ties summary elements to source documents using hyperlinks to original note content and highlights the specifically summarized text, tailoring each summary to a use case such as palliative care, advanced illness care, or pre-operative assessment.

Prioritize Content by Role and Use Case

Enable role- and use-case-based content prioritization. This may include: defining a summary profile that maps clinician role, care setting, and visit reason to content-prioritization rules (which sections to foreground, compress, or de-emphasize); treating prioritization as emphasis, not omission, with safeguards that prevent role- or specialty-based filtering from concealing clinically relevant data and a baseline universal view always available; standardizing how clinician context is communicated to the summarizing agent as structured input; documenting the prioritization logic; and establishing default profiles through clinician-led review, validating them against real workflows before release, and monitoring downstream outcomes. Each profile should have a defined review cadence and a named clinical owner.

Appendix 1: Consensus Method

Best practice statements are collected from work group presentations and discussions. To ensure alignment across stakeholders, CHAI uses a three-phase consensus process for Best Practice Statements (BPS) generated through work group activities:

Phase 1: Initial Consensus Check

- **Purpose**: Gauge initial agreement on each draft BPS.
- **Voting Options**:
- *Include / Include Contextually / Exclude / Abstain*
- **Decision Rules**:
  - If ⩾ 2/3 vote "Include" → **Consensus achieved** (no further action).
  - If < 2/3 "Include", but ⩾ 2/3 combined "Include" + "Include Contextually" → **Flagged for Phase 2**.
  - If ⩾ 25% vote "Exclude" → **Flagged for Phase 3**.
  - If ⩾ 2/3 vote "Exclude" → **Automatically excluded**

Phase 2: Revote with Revisions

- **Purpose**: Re-evaluate BPS that did not reach consensus in Phase 1 (but had < 25% "Exclude").
- **Format**: Original and revised BPS shown side-by-side (based on Phase 1 feedback).
- **Voting Options**: *Include / Exclude / Abstain*, with an optional comment field.
- **Outcome**: Results used to determine final inclusion or exclusion.

Phase 3: Live Discussion and Vote

- **Purpose**: Address BPS with ⩾ 25% "Exclude" in Phase 1 (strong disagreement).
- **Steps**:
  - Facilitated group discussion of flagged BPS.
  - Live revote during the meeting + optional 1-week offline voting.
- **Voting Options**: *Include / Exclude / Abstain*
- **Outcome**: Final decision made based on discussion and revote results.

Appendix 2: Thank You and Attribution

We want to start by thanking every individual who showed interest, participated, listened, and came along with us in our work. CHAI is at its core, a convener, and a member-driven non-profit. We are so grateful to be on this journey with you towards responsible AI in health for all. Your experiences, your feedback, your contributions, all make us who we are and help bring us to where we need to be.

CHAI extends its sincere appreciation to all participants for their time, expertise, and thoughtful feedback throughout the development process. CHAI would like to provide special recognition to the individuals below who requested attribution by name for their contributions to these efforts:

- Ashish Patel, MHS, CareSet
- Pawan Jindal, MBBS, MS, Darena Health
- Anand Chowdhury, MD, MMCi, Duke University School of Medicine
- Katherine Eisenberg, MD, PhD, Dyna AI
- Benjamin Hollis, EBSCO, Inc.
- Daniel D. Johnson, BSN, RN, NI-BC, Essentia Health
- Muhammad Xhemali. PharmD, RPh, Family Health Center of Worcester
- Ashley M Hopkins, Flinders University
- Sonali Tamhankar, PhD, Fred Hutchinson Cancer Center
- Cheryl Campbell, Healthcare Performance Group Inc.
- Joe Derenzo, PMP, Healthcare Performance Group Inc.
- Michael Lardieri, LCSW, iBPM, LLC
- Rémy Ibarcq, Individual Contributor
- Dorcas Yao, MD, Individual Contributor
- Michael Choma, MD, PhD, Infinitus Systems, Inc.
- Tapan Shah, PhD, Innovaccer
- Bita Behrouzi, MD, MaineHealth Maine Medical Center; Tufts University School of Medicine
- Jeremy Attermann, MSW, National Council for Mental Wellbeing
- Scott Ross, Opala
- Shiba Kuanar, University of Minnesota
- Seneca Perri Moore, PhD, RN, University of Utah, College of Nursing
- Keri Hall, MD, MSc, Wolters-Kluwer - UpToDate

Appendix 3: Document Process & Governance

AI Transparency Statement: AI was used in the initial drafting, thematic analysis, and summarization to aid in the development of this document. AI was also used to summarize workgroup discussions and presentations into draft best practice statements. All best practice statements and initial drafts were reviewed and heavily edited by workgroup leads, members, and the CHAI program management team.

Document Maintenance Plan:

- BPG will be hosted on CHAI website.
- JIRA/Microsoft Issue Form will be available for ongoing feedback and new suggested content
- CHAI program management (PM) will monitor feedback submissions via internal tracker
- For any submission that is deemed “major”, CHAI PM will flag and raise with work group leads.
- CHAI PM and ad-hoc WG Leads will communicate via email with the flagged feedback submission. CHAI PM will review all feedback and flag "major" feedback.
- For flagged feedback, 1-3 options will be considered after work group leads review:
  - No action needed
  - Straightforward feedback - can be resolved via email correspondence and CHAI PM reflects changes onwebsite
  - Complex feedback - brief 30min meeting with WG Leads to discuss and resolve. CHAI PM reflects changes on website. This meeting can be used for 1:M complex feedback submissions. Formal decisions will require simple majority vote by WG Leads + CHAI PM