THE SERVICE HANDSHAKE

An open interaction standard for service interactions where one or more parties are represented by AI agents.

v1.1

Author: Maria McCann, Founder, Neos Wave

Version: 1.1 — Updated March 2026

Status: Open Interaction Standard — Published March 2026

DOI: 10.5281/zenodo.19046746

Contact: neoswave.com

Foreword

Customer Service has always had one designer. The company built the systems, flows, data and rules. The customer entered the company's environment on the company's terms. Every channel, journey, metric assumes this. Until now. Now customers have AI that acts on their behalf. Two designers. Two agendas. One interaction. And almost nobody has designed for this reality.

Without each party designing their own service terms, AI agents can be inadvertently virtuous or malicious to the other party. Not because anyone intended harm. Not because the technology failed. Because no one designed for the interaction in the first place.

An agent that tries to help a vulnerable person navigate a housing repair may escalate a complaint in ways the landlord's system cannot receive. An agent managing a return on behalf of a consumer may commit a brand to a resolution it has no authority to honour. An agent acting on behalf of a business may expose data its customer never authorised. In each case, the agent was doing exactly what it was built to do. The problem was the absence of a declared handshake between parties: an agreed set of terms that each side brings to the interaction before it begins.

This standard applies wherever agents meet in service: between consumers and brands, between businesses, and across the layered relationships of platforms, suppliers, and the people they both serve.

Service is now happening in four modes. Organisations have designed carefully for Modes 1 and 2 (human to human, human to brand AI). Modes 3 and 4 (consumer AI contacting brand systems, and brand AI meeting consumer AI) are already occurring, with no equivalent design thinking applied to either. The infrastructure exists. The interactions are happening. The design standard does not.

I have been testing this reality for the past year. I have deployed consumer AI agents into live service environments, without brand-side declarations to receive them, because none existed. I have built triadic AI systems where the architect, the beneficiary, and the organisation being contacted are all different parties with different authority levels. What those tests revealed is not a technology problem. It is a design problem. Liquid service, the fluid combination of AI, people, knowledge, and process, requires each party to define their own terms before the interaction begins. Without that, even the best-designed agent is operating without a brief.

This document provides that brief. It is an open interaction standard that makes AI-involved service interactions designable, auditable, and safe, regardless of which agents, platforms, or identity layers are involved. The question is the same whether the agent contacting your service today is narrow in scope or approaching something more general: what is it authorised to do, and what happens when things go wrong?

This standard sits above the protocol stack being built by Google, Anthropic, OpenAI, and Mastercard. Those layers answer: who is this agent, how does it communicate, and what can it buy? This standard answers the question they have deliberately left open: what is each party trying to achieve, what are they authorised to do, and what happens when things go wrong? The standard is transport-agnostic: it operates above protocols such as Anthropic MCP, Google Agent-to-Agent, and IBM ACP, consuming their outputs and governing the interaction layer none of them address.

Humans are not an escape hatch in this standard. They are designed participants: first-class parties with full context inheritance, explicit authority levels, and always-available pathways. That is not a philosophical position. It is the architectural lesson of every AI governance failure to date.

We are making this standard open to anyone to use because the pace of change means no single organisation has the complete picture. We are sharing what we see, so the industry can benefit from it, build on it, and improve it. For the good of everyone involved.

Maria McCann

Founder, Neos Wave | March 2026

1. Why This Standard Exists

1.1 The Proof That Demand Is Real

OpenClaw: Ecosystem Signal (2026)

OpenClaw reached 247,000 GitHub stars in weeks and was acquired by OpenAI, its founder joining OpenAI's team and its open-source agent becoming foundation model infrastructure. It is the clearest proof that consumer-side agentic AI is not an emerging trend. It is already mainstream.

It is also the clearest proof of what happens without structured declarations. Users reported agents deleting entire inboxes, sending hundreds of unsolicited messages, and initiating disputes with third parties, all from misinterpreted instructions with no fallback rules. Security researchers found malicious skills in the ClawHub marketplace performing data exfiltration without user awareness. A security audit identified 512 vulnerabilities; Shodan scans found approximately 1,000 publicly accessible instances with no authentication.

OpenClaw had a personality file, an agent file, and a tools file. It had no declaration of what it was authorised to commit to on behalf of its human. That is the gap this standard fills. OpenClaw is not an outlier. It is the first signal of what the consumer agent ecosystem looks like at scale. Every major platform is building towards this. The standard exists because that ecosystem is arriving faster than service design has responded.

Source: OpenClaw GitHub (Jan 2026); security audit reports (Jan–Feb 2026); Wired, "The AI Agent That Ate Your Inbox" (Feb 2026); Reuters, "OpenClaw founder Steinberger joins OpenAI, open-source bot becomes foundation model infrastructure" (Feb 2026)

Case 2: Air Canada, Brand AI Commits Without Authority (2024)

In November 2022, a passenger booked last-minute flights to attend his grandmother's funeral, having been told by Air Canada's chatbot that he could submit for a discounted bereavement fare within 90 days of travel. When he submitted his claim, Air Canada refused; the actual policy required the request to be made before travel. The airline's representative admitted the chatbot had provided "misleading words."

Air Canada attempted to argue the chatbot was "a separate legal entity responsible for its own actions." The British Columbia Civil Resolution Tribunal rejected this emphatically, finding Air Canada liable for negligent misrepresentation and ordering compensation. The ruling established that organisations are legally responsible for commitments made by their AI systems, regardless of whether those commitments were authorised.

The AI had no declaration of what it was authorised to commit to on behalf of the brand. A Service Handshake Declaration would have defined the brand AI's authority level explicitly, preventing both the misleading commitment and the liability that followed.

Source: Moffatt v. Air Canada, 2024 BCCRT 149 (British Columbia Civil Resolution Tribunal, February 2024); CBC News, February 2024

Case 3: AI Shopping Agents, Consumer Agents Contacting Brands Without Disclosure (2025)

By 2025, AI shopping agents from OpenAI, Google, and others were executing purchases, initiating returns, and contacting retailer customer service teams autonomously, without identifying themselves as AI or declaring the scope of their authority. Major e-commerce platforms began issuing cease-and-desist letters to AI agent providers, citing terms of service violations, fraud risk, and the impossibility of distinguishing authorised consumer agents from malicious bots.

The legal question of who is liable when an agent acts beyond its authorisation, and how a brand can verify what a consumer agent is permitted to do, remained entirely unresolved. Visa, Mastercard, and Google each launched separate frameworks to address agent identity and payment authorisation. None addressed what the agent was attempting to achieve, what constraints it was operating under, or what should happen when things went wrong.

Every one of these interactions is a Mode 3 contact occurring without a declaration standard. The Service Handshake provides the missing layer: what each party is trying to achieve, what they are authorised to do, and what happens when the interaction fails.

Sources: Mastercard Agent Pay announcement (April 2025); Davis Wright Tremaine, "The Next AI Frontier: From Prompts to Purchases" (2025); Medium, "Why AI Shopping Agents Are Facing Legal Pushback" (November 2025); Morgan Stanley Research, agentic commerce projections (2025)

These are not edge cases or predictions. They are documented failures across both sides of the interaction: consumer AI acting without declared authority, and brand AI committing without declared limits. Consumer agents are already contacting service organisations on behalf of real people. Brand AI is already making commitments on behalf of organisations. The question is not whether this is happening. The question is whether your organisation is designed for it.

1.2 The Four Modes of Modern Service

Service now happens in four interaction modes. Organisations have designed carefully for Modes 1 and 2. Modes 3 and 4 are already occurring without equivalent design thinking.

Mode	Participants	Current State	Standard Applies
Mode 1	Human — Human	Fully designed. High cost, high empathy.	Optional. Documents existing SOPs.
Mode 2	Human — Brand AI	Heavily invested. Rapidly commoditising.	Brand declaration required.
Mode 3	Customer AI — Human	Near-zero readiness. Already happening.	Both declarations required.
Mode 4	Customer AI — Brand AI	Zero readiness. No standard exists.	Both declarations required. Reject without.

Modes 3 and 4 must reject or severely limit interaction if declarations are missing or incompatible. This is not a preference. It is the minimum safety condition for autonomous agent interaction.

1.3 The Design Gap

The market is building every layer around this standard. Identity and verification layers answer: who is this agent? Transport protocols answer: how do agents communicate? Commerce protocols answer: what can agents buy? None of them answer the question this standard addresses:

The Unanswered Question

What is each party trying to achieve, what are they authorised to do, what data can they use to achieve their goals, and what happens when things go wrong?

This standard occupies the interaction design and intent layer: the layer that no major protocol has yet claimed.

1.3a Application Contexts

The four interaction modes apply consistently across three commercial contexts. The declaration mechanic does not change. What changes is the nature of the relationship between parties, the commercial weight of commitments made, and the complexity of the authority structure involved.

Business to Consumer (B2C)

In B2C, the Service Handshake operates as a cold interaction between parties with no prior relationship. A consumer or their AI agent arrives at a brand's service environment and declares their terms. The brand's system either receives that declaration and responds within it, or escalates to a human pathway. Neither party has pre-negotiated what the interaction will look like. Sovereignty is asserted at the point of contact. This is the minimum viable case the standard was designed for.

Business to Business (B2B)

In B2B, the Service Handshake has a structural advantage that B2C does not: there is usually a contract. An onboarding process. An existing commercial relationship. This means the declaration does not have to be negotiated in real time. It can be pre-agreed as part of the commercial relationship itself. The bilateral Service Handshake Declaration becomes a treaty: established at contract stage, machine-readable, and enforceable at the interaction layer. Every subsequent agent interaction between the two parties operates within pre-declared terms that both sides have already agreed.

This changes the authority level question significantly. In B2B, an agent is not just acting on behalf of a person. It is acting on behalf of an organisation, potentially committing commercial and contractual terms at scale. The authority level field in the declaration carries legal weight. Fallback rules are not just operational: they are contractual. The bilateral treaty established at onboarding is what makes autonomous B2B agent interaction safe.

Business to Business to Consumer (B2B2C)

B2B2C is the most structurally complex application of the standard, and the most underserved. It combines both of the above and introduces a third layer that neither alone addresses.

In B2B2C, three sets of terms are in play simultaneously. First, a pre-agreed bilateral treaty between the two businesses, established at contract stage, governing what each party's agents can commit to, access, and escalate within the commercial relationship. Second, a live unilateral declaration from the consumer or their agent, asserted cold at the point of interaction, independent of the B2B relationship behind it. Third, a resolution hierarchy: the declared order of precedence when the pre-agreed treaty and the live consumer declaration conflict. Which layer takes precedence? What is the fallback when no resolution within the treaty is possible? Who is informed, and how?

The resolution hierarchy is the element that makes B2B2C designable. Without it, a conflict between treaty terms and consumer declaration has no governed outcome. The agent defaults to one party's interests, usually the one it was built by, without transparency to the other.

Consider a housing association deploying an AI agent to manage tenant service contacts on behalf of a local authority. The housing association and local authority have pre-agreed their bilateral declaration at contract stage. A tenant contacts the service with a damp and mould complaint. Their needs may sit outside what the housing association's treaty with the local authority permits it to resolve autonomously. The resolution hierarchy determines what happens next: what the agent can offer, what it must escalate, and to whom. Without it, the agent is either inadvertently virtuous, offering a resolution it has no authority to deliver, or inadvertently unhelpful, escalating unnecessarily because it cannot distinguish what it can and cannot commit to.

The Service Handshake provides the framework for all three contexts from a single standard. The schema does not change. The declaration elements do not change. What changes is how parties establish their terms: cold and unilateral in B2C, pre-agreed and bilateral in B2B, and layered with a resolution hierarchy in B2B2C.

2. Where This Standard Sits

This standard is protocol-agnostic. It sits above the emerging agent infrastructure stack and is designed to be compatible with whichever transport, identity, or commerce layers an organisation uses.

Layer	Who Owns It	Relationship to This Standard
Security & Governance	Cloud Security Alliance ATF, OWASP	Trust enforcement layer that declarations feed into
Identity & Verification	OpenAI signatures, KYA, OAuth/JWT	Referenced in declaration via identity_tokens field
Transport & Communication	Google A2A, Anthropic MCP, IBM ACP	Agnostic — declarations travel over any transport
Commerce & Transaction	Mastercard Verifiable Intent, Stripe ACP, Google UCP	Complementary — covers service interactions they don't
Agent Platforms	OpenClaw, Operator, Apple Siri, enterprise copilots	These agents carry declarations into interactions
Interaction Design & Intent	This Standard	The layer this standard claims

This positioning makes the standard an enabler of the protocol ecosystem, not a competitor. A declaration made under this standard can travel over A2A, be authenticated via OpenAI signatures, and feed into ATF governance — without modifying any of those layers.

The Critical Distinction from Commerce Protocols

Commerce protocols such as Google UCP treat human involvement as a failure state — a 'requires_escalation' condition triggered when automation cannot complete a transaction.

This standard treats humans as first-class designed participants with full context inheritance, explicit authority levels, and always-available pathways.

Service interactions — unlike purchases — involve emotional load, vulnerability, legal liability, and triadic authority structures (carer, beneficiary, advocate). They cannot safely treat human involvement as an error condition.

This architectural distinction is not replicable by extending a commerce protocol. It requires a different design philosophy from the ground up.

3. The Declaration Framework

Before any service interaction where at least one party is represented by an AI agent, each party makes a structured declaration. This declaration — the Service Handshake Declaration — specifies six elements: four primary and two modifiers.

Element	What It Declares
1. Goals	What this party is trying to achieve, in priority order. Includes primary goal, secondary goals, and explicit non-goals.

Element	What It Declares
2. Options & Constraints	What this party can and cannot do. Allowed actions, hard constraints, soft constraints, and jurisdictional limits.
3. Data Permissions	What data can be shared, for what purposes, with what retention policy and trust thresholds.
4. Fallback Rules	What happens when confidence is low, goals conflict, or harm risk is detected. Triggers, actions, automation scope, and responsibility assignment.
5. Cost Parameters	What the interaction is allowed to cost in time, tokens, or effort. Escalation rules tied to cost limits.
6. Authority Level	Who the agent is acting for, who benefits, what the agent can commit to, and evidence of delegation.

3.1 Goals

What this party is trying to achieve in this interaction, in order of priority. The non_goals field is critically important — it provides explicit instruction to prevent scope creep, manipulation, and mission drift. In healthcare and social care contexts, non_goals such as 'do not discuss billing during a clinical appointment' carry direct safety implications.

```
Example — Consumer Agent Goals Declaration

primary_goal: "Resolve missing delivery by replacement or refund"
secondary_goals: ["Minimise interaction time", "Maintain account
relationship"]
non_goals: ["Do not accept less than full refund or equivalent
replacement",
            "Do not discuss unrelated products", "Do not authorise
account changes"]
success_metrics: { delivery_resolved: true, time_under_minutes: 10 }
```

3.2 Options and Constraints

What this party can and cannot do to reach their goals. Hard constraints are non-negotiable. Soft constraints are preferences and trade-offs. Jurisdictional constraints handle legal and regulatory limits.

3.3 Data Permissions

What data each party allows to be shared, for which purposes, with what retention rules, and at what trust threshold the agent can act autonomously. The trust_levels field distinguishes between the agent's confidence in its own actions (self_trust) and what it will accept from counterparty agents without additional verification (counterparty_trust).

3.4 Fallback Rules

The most operationally critical element of the declaration. Fallback rules define what happens when the interaction cannot proceed safely within the declared parameters.

Three categories of fallback trigger must be specified:

- Confidence thresholds — at what point the agent must stop acting autonomously
- Conflict patterns — what happens when goals cannot be reconciled
- Harm indicators — mandatory routing when vulnerability signals are detected

On Vulnerability Detection

The harm indicator fallback — 'if user appears vulnerable, route to specialist team' — is the most consequential line in any declaration.

In healthcare and social care contexts, this is the difference between a useful tool and a dangerous one. What constitutes a vulnerability signal must be explicitly defined, not left to agent inference.

Organisations deploying this standard in healthcare, social care, housing, or financial services contexts are strongly advised to develop their vulnerability detection criteria in collaboration with safeguarding specialists and relevant regulatory bodies.

This version of the standard identifies the requirement. A dedicated vulnerability declaration module is in development for v1.2 of The Service Handshake.

The maximum_automation_scope field defines the outer boundary of autonomous action within fallback rules: what the agent may initiate but not complete without human approval. Examples: 'AI may book appointments but not cancel treatment.' 'AI may negotiate bills but not switch providers.'

3.5 Cost Parameters

What the interaction is allowed to cost in time, tokens, or effort. Cost parameters set limits on resource consumption — duration, loops, and effort — and trigger escalation when those limits are approached.

3.6 Authority Level

Who the agent is acting for, who benefits from the outcome, what the agent can commit the principal to, and evidence of delegation. This element enables triadic interactions — where the architect, the beneficiary, and the counterparty organisation are all different parties — to be correctly modelled.

Example — Triadic Authority Declaration

A family member configures an AI agent to manage an elderly parent's utility contracts.

`principal_id`: family_member (the architect — initiates, configures, owns the agent)

`beneficiary_id`: elderly_parent (the beneficiary — receives the outcome)

`authority_scope`: manage_billing_queries, accept_service_changes

`delegation_proof`: power_of_attorney

`fallback_trigger`: if_contract_cancellation_proposed → escalate_to_architect_for_approval

The organisation receiving this interaction needs a framework to receive and validate this declaration. Without it, they have no way to correctly assess the authority of the agent contacting them, the interests of the party who will be affected, or what escalation pathway is appropriate.

4. Machine-Readable Schema

The declaration is designed to be both human-readable (for service design, legal, and operations teams) and machine-readable (for agents and orchestration systems). The minimum serialisation is JSON.

Key schema fields at the top level:

- protocol_version — e.g. 'SH-1.1'
- principal_id — identifier for the party the agent represents
- agent_id — identifier for the agent itself
- beneficiary_id — identifier for the party who benefits (may differ from principal)
- identity_tokens — verification tokens from Apple, OpenAI, Google or equivalent
- valid_from / valid_to — declaration validity window
- issuer — who authored and approved this declaration

The six declaration elements (goals, options_constraints, data_permissions, fallback_rules, cost_parameters, authority_level) are nested objects within the schema. The condensed top-level structure is shown below. The full field definitions, validation rules, and worked examples are available in the companion Technical Specification. The canonical machine-validatable JSON Schema is published separately as sh-1.1.schema.json at neoswave.com/service-handshake/sh-1.1.schema.json. Implementations SHOULD validate declarations against the canonical schema before compatibility evaluation.

```
// Service Handshake Declaration — SH-1.1 top-level structure
{
  // ── Identity fields
  "protocol_version": "SH-1.1",
  "principal_id": "string",
  "agent_id": "string",
  "beneficiary_id": "string",    // may differ from principal_id in
triadic interactions
  "identity_tokens": [],          // tokens from Apple, OpenAI, Google,
KYA
  "issuer": "string",
  "valid_from": "ISO-8601",
  "valid_to": "ISO-8601",

  // ── Six declaration elements
  "goals": { "primary_goal": "...", "non_goals": [...],
"success_metrics": {...} },
  "options_constraints": { "allowed_actions": [...], "hard_constraints":
{...} },
```

```
  "data_permissions": { "data_sources_allowed": [...], "trust_levels": {
"self_trust": 0.85 } },
  "fallback_rules": { "fallback_triggers": {...}, "fallback_actions":
[...],
                      "responsibility_assignment": "role" },
  "cost_parameters": { "max_duration_minutes": 10, "max_loops": 3 },
  "authority_level": { "authority_scope": [...], "delegation_proof":
"..." }
}
```

Identity tokens from Apple, Google, OpenAI and emerging standards such as KYA (Know Your Agent) and AgentFacts live in the identity_tokens field. These are validated by lower-level authentication systems. This standard consumes only their status — valid/invalid, scope — and focuses on the declaration layer that follows identity confirmation.

On Storage and Implementation

Declarations can be stored as custom fields or objects in CRM and ticketing systems (Zendesk, Salesforce), held in orchestration platforms (LangChain, Make.com, custom Node.js services), or maintained as external config files referenced by interaction ID.

This standard does not prescribe a specific technology stack. A declaration that travels over Google A2A, authenticated via OpenAI cryptographic signatures, and stored in Zendesk custom fields is fully compliant.

Each declaration instance should include protocol_version, issuer, and valid_from/valid_to. Changes to organisational declaration templates should follow a simple governance process (service design + legal + risk) with change logs retained for audit.

5. Worked Examples

5.1 Mode 3 — Retail Delivery Resolution

A consumer AI agent contacts a major UK retailer's contact centre on behalf of a customer whose delivery has not arrived. The agent passes identity verification, describes the issue, and negotiates resolution with a human agent — who, in documented tests, did not know they were speaking to a machine.

This scenario has been tested in production. Full transcripts exist. The consumer agent completed the interaction successfully in all three test calls. None of the human agents identified the contact as AI-generated.

Declaration Element	Consumer Agent Declares	Brand Declares
Goals	Resolve missing delivery. Accept replacement if within 48 hours. Non-goal: do not accept less than full refund or equivalent replacement.	Resolve to customer satisfaction. Success: issue closed, customer satisfied.

Declaration Element	Consumer Agent Declares	Brand Declares
Options & Constraints	Accept refund or replacement only. Hard constraint: replacement within 48 hours.	Offer refund up to £100, replacement, or escalation. Hard constraint: max_refund = £100.
Data Permissions	Order number, delivery address, account email. Problem resolution only. No marketing.	Access order and account history. Problem resolution only.
Fallback Rules	If unresolved after 3 loops: request human escalation.	If confidence < 0.7: escalate to human. Responsibility: Customer Care Team Lead.
Cost Parameters	Maximum 10 minutes.	Resolve within standard handle time.
Authority Level	Account holder verified. Can accept refund or replacement. Cannot change account details.	Brand agent. Can commit to standard policy resolutions.

5.2 Mode 3 — Triadic Interaction (The Dadbot Scenario)

A family member configures an AI assistant — built on enterprise-grade, GDPR-compliant infrastructure — to manage their elderly parent's utility contracts. The agent contacts a utility provider to negotiate a broadband contract renewal on the parent's behalf.

This interaction introduces three distinct parties: the architect (who configures and deploys the agent), the beneficiary (who receives the outcome), and the organisation being contacted. Without a declaration standard, the organisation has no framework to assess the authority of the agent, the interests of the affected party, or what escalation pathway is appropriate.

The authority_level declaration resolves this: it explicitly names the principal, the beneficiary, the scope of what the agent can commit to, and the delegation evidence. The fallback rule — escalate to architect for approval if contract cancellation is proposed — ensures that decisions beyond the declared scope return to a human.

5.3 Mode 4 — Brand AI Meets Consumer AI

A brand deploys an AI agent to handle inbound service contacts. A consumer deploys an AI agent to manage their service interactions with that brand. When the two agents interact, the outcome is determined by whether they can reach alignment on goals, constraints, and fallback rules.

Without Declaration	With Declaration
Agents loop without clear exit criteria.	Agents negotiate within declared parameters.
Escalation happens inconsistently.	Escalation triggers are explicit and auditable.
No audit trail of who decided what.	Disputes can be reviewed against original declarations.

Without Declaration	With Declaration
Consumer agent may be manipulated by brand agent into scope outside its mandate.	Non-goals field prevents scope creep by either party.
No defined responsibility when interaction fails.	Responsibility assignment routes to named role.

6. The DualCX Exposure Map

The Exposure Map is the diagnostic tool that organisations use to assess their current readiness across all four interaction modes. It maps four modes against three declaration dimensions to identify design gaps.

	Goals Alignment	Data Protocol	Fallback Design
Mode 1: Human — Human	Designed (SOPs, scripts)	Designed (GDPR policy)	Designed (escalation paths)
Mode 2: Human — Brand AI	Designed (chatbot intent)	Designed (data policy)	Partially designed
Mode 3: Customer AI — Human	GAP: No inbound AI declaration	GAP: No consent/ permission framework	GAP: Human receives no context
Mode 4: Customer AI — Brand AI	GAP: No shared contract exists	GAP: No bilateral data protocol	GAP: No audit trail

Most organisations score green across Modes 1 and 2 and red across Modes 3 and 4. The Modes 3 and 4 gaps represent the design work this standard enables.

A free interactive Exposure Map diagnostic — which assesses your organisation across all twelve cells and generates a personalised readiness report — is available at neoswave.com.

7. Regulatory Context

This standard is not a compliance product. It is a design standard that makes compliance a natural by-product. Organisations that adopt it will find that the EU AI Act's documentation and audit requirements, human-in-the-loop obligations, and transparency provisions are addressed as a consequence of good interaction design rather than as a separate compliance exercise.

Regulation	Relevance to This Standard
EU AI Act (transparency obligations August 2026; full enforcement August 2027)	Transparency obligations (AI disclosure), human-in-the-loop requirements, and audit trail requirements map directly to declaration elements.

Regulation	Relevance to This Standard
EU AI Act — High Risk Classification	AI systems influencing customer decisions require conformity assessments and documentation. Declaration records provide this foundation.
UK Data (Use and Access) Act 2025	In force February 2026. Raises PECR fines to £17.5M or 4% of global turnover. Data permissions declarations directly address this.
UK CMA Powers	Direct consumer-protection enforcement up to 10% of global annual turnover. Fallback and authority declarations reduce autonomous action risk.
GDPR (ongoing)	Data permissions element provides the per-interaction consent and purpose limitation framework GDPR requires for AI-processed contacts.

A note on the field evidence referenced in this document: the live service tests were conducted without brand-side declarations specifically to establish that the design gap exists. Under The Service Handshake, consumer agent contacts carry explicit declarations — the standard addresses transparency obligations rather than creating a conflict with them. Undeclared contact is the problem this standard solves.

The EU AI Act's transparency obligations come into force in August 2026, with full enforcement following in August 2027. Organisations that adopt this standard before those deadlines will have auditable, documented pre-interaction frameworks ready for regulatory review.

8. Evidence Base and Maturity

This standard is grounded in real experiments, not theoretical frameworks. Evidence maturity is declared transparently:

Mode	Evidence Maturity
Mode 1: Human — Human	Production. Extensive operational evidence across multiple sectors.
Mode 2: Human — Brand AI	Production. Well-documented across contact centre deployments.
Mode 3: Customer AI — Human	Field evidence. Tested in live service environments with documented interaction records. All organisations involved have been contacted and advised of the research. Full methodology, sample scope, and findings will be published in the State of Dual CX report, Q3 2026.
Mode 4: Customer AI — Brand AI	In development. Declaration schema implemented and validated. Full Mode 4 interaction testing in progress.

The standard will be updated as pilot evidence matures. A State of Dual CX report — based on Exposure Map assessments across 85 organisations — will be published in Q3 2026. Current benchmark across ecommerce, retail, healthcare, and SaaS: 100% score green for Mode 1 (Human to Human). 52% for Mode 2 (Human to Brand AI). 0% for Modes 3 and 4.

9. Adopting This Standard

9.1 A Practical Starting Point

This standard is open to anyone to use. If you are a CX or operations leader, a service designer, a technology team building agentic systems, or an organisation navigating the shift to Mode 3 and 4 readiness — this is for you. You do not need permission to adopt it. Start here.

1. Pick one use case — missing delivery, billing query, appointment booking.
2. Draft a brand-side declaration for that use case in JSON.
3. Store it in your contact platform as custom fields or an org profile.
4. Train agents or configure AI to follow the declaration rules.
5. Send AI-generated test interactions into the declaration-defined queue.
6. Measure: repeat contacts, time to resolution, escalation quality, agent cognitive load.
7. Iterate the schema based on what breaks or what is missing.

9.2 Benefits by Stakeholder

Stakeholder	Immediate Benefit	Strategic Benefit
CX & Operations Leaders	Design framework for Mode 3/4 contacts arriving now	Auditable AI interaction records for regulatory readiness
Service Designers	Vocabulary and schema for multi-party interaction design	Foundation for AI-era service architecture
Legal & Risk Teams	Pre-interaction documentation of constraints and authority	Compliance foundation for EU AI Act and UK regulations
Technology Teams	Machine-readable schema compatible with existing platforms	Protocol-agnostic layer above A2A, MCP, and identity stacks
BPOs & Outsourcers	Framework for safely increasing AI handling share	Differentiated capability as consumer agent contacts scale
Consumers & Their Agents	Explicit fallback and escalation pathways	Structured framework for agents to operate within

10. About This Document

This is a living standard. To contribute to its evolution: test in pilot environments, gather feedback from service designers, legal and risk teams, and developers, and submit observations to neoswave.com. Revised versions will be published with clear change logs.

About the Author

Maria McCann is the co-founder of Neos Wave and has spent over twenty years designing and operating customer experience at scale — including senior roles at Spotify and ASOS. She has been designing and testing AI-involved service interactions since 2022, including consumer-side agent deployments, triadic AI systems, and live contact centre tests.

Neos Wave builds AI-powered service transformation for organisations navigating the shift from Mode 1-2 to Mode 3-4 readiness. neoswave.com

Version 1.0 published March 2026. Version 1.1 updated March 2026. Published as The Service Handshake v1.1.

DOI **10.5281/zenodo.19046746**