RETAIL PAYMENTS INFRASTRUCTURE BOARD CONSULTATION

Personal response on programmable and agentic payments

Q16, Q17 and its sub-question a, Q18, and the Equality Act question, in the consultation on the Design of the Future Retail Payments Infrastructure

Respondent	King Yew Choo, London, United Kingdom
Capacity	Personal capacity
Contact	k.choo@trueprimary.com
Affiliation	Founder, True Primary
Written record	https://kingyewchoo.co.uk
Date	2 September 2026
Confidentiality	None. I am content for this response to be published in full and attributed to me.
Structure	Numbered paragraphs 1 to 54, with an unnumbered summary above and a closing below.

Summary of what I ask for

I ask for five things in the core, listed here and argued below.

1. **A mandate reference** carried unaltered end to end, so the institution enforcing a delegated authority and a later reviewer identify the same mandate (paragraphs 18 and 44).
2. **An agent-initiated marker**, distinct from the agent's identity, attested by the authorised participant that submits the instruction (paragraphs 22 and 44).
3. **A client-supplied request key**, scoped by the submitting participant and carried into the core with that participant's identifier, with that participant recording the result against the key and answering a repeat with the original result (paragraphs 5 and 11).
4. **A uniqueness constraint on that key**, applied where inter-participant obligations are determined, so one key and one canonical instruction produce at most one clearing outcome (paragraphs 9 and 14).
5. **A record binding authority, payment and outcome**, required and retained at the infrastructure boundary, holding references and not content, with bounded access and retention (paragraphs 24 and 25).

The first three are fields the core carries. On the third, the replay behaviour is enforced at the submitting participant's interface, not in the core. The last two are properties the core enforces. None of them requires the core to run agent logic.

> **Central position.** The core should carry three fields whose meaning it never reads, enforce one uniqueness property it can actually guarantee, and require one record at its boundary. That record is the largest of the five and the most privacy-sensitive; paragraph 25 sets the limits it needs. It should not administer mandates, authenticate an agent, judge a purpose or allocate liability. Those belong to institutions, schemes and regulation. I set out below what each of those is, and what the core cannot do however it is designed.

How the pieces fit

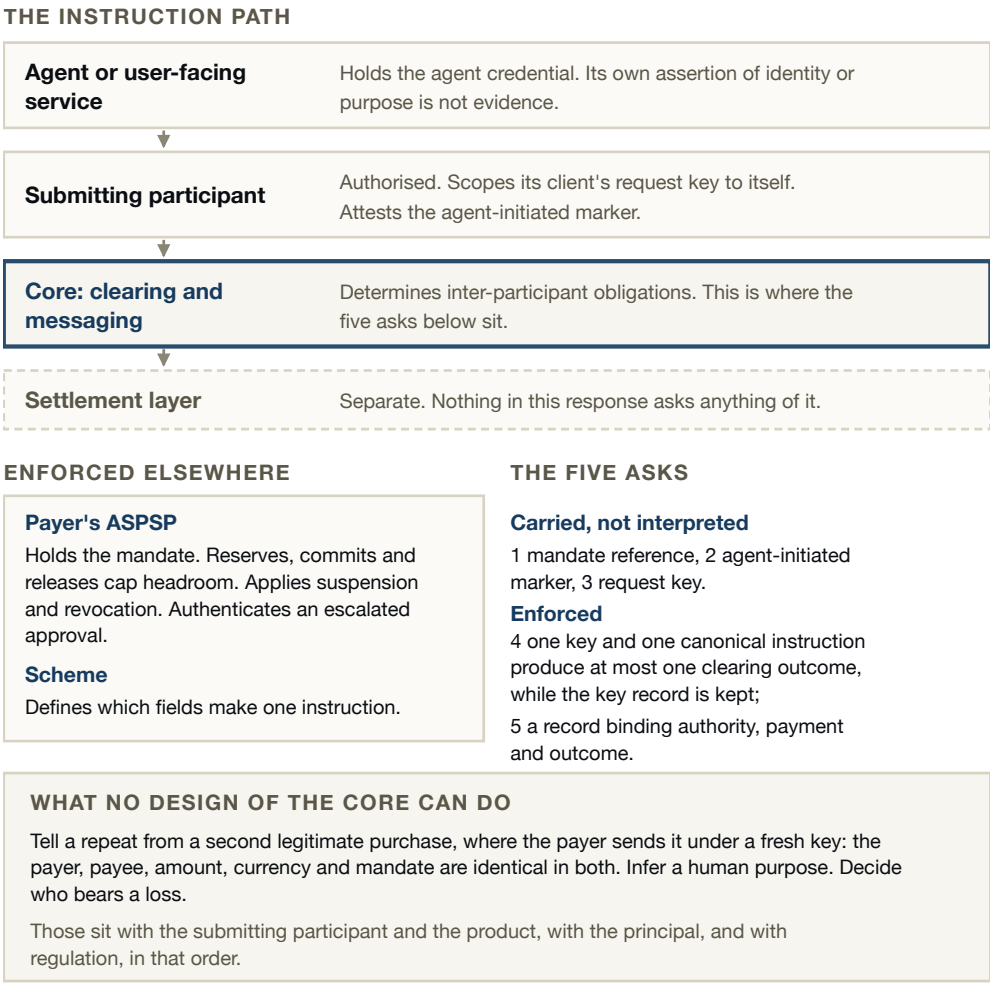

Figure 1. Where each ask sits. The instruction path runs from the agent or user-facing service to the submitting participant, then to the core, with settlement in a separate layer. The core carries three fields and enforces two properties; everything else is enforced elsewhere.

Introduction

1. I, King Yew Choo, respond in a personal capacity. I am the author of a working paper that specifies and formally models authorisation, payment and audit controls for payments that software agents request on a principal's behalf. I am also the founder of True Primary. This response addresses Q17 on the safe provision of agentic payments, and the sub-question the consultation labels a, on barriers to their development or adoption. It applies the same analysis to Q16 on programmable payments and Q18 on consumer protection and fraud. Beyond the Equality Act question at paragraph 54, I answer no other question. The argument below reaches two of the others without answering them. Paragraph 10 argues that one validation belongs inside a core function the consultation already names, and paragraph 24 that one record belongs within the core rather than in a separately provided service. That is the choice Q7 and Q8 put between them, the baseline services delivered within the core infrastructure and the payment-enabling or value-add services delivered separately. On every other question I have nothing grounded to add. I use "agent" in the consultation's sense, a system taking autonomous action on a user's behalf. Regulation 2(1) of the Payment Services Regulations 2017 gives the word a separate defined meaning.[1] There it is a person who acts on behalf of an authorised payment institution, a small payment institution or a registered account information service provider in the provision of payment services. I flag that sense where I rely on it, at paragraph 44. I call a transaction agent-initiated to name whose software triggered the request. That is separate from the regulated role of initiating a payment order.

2. Declaration of interest. True Primary builds software through which agents request purchases of expert work, and has a commercial interest in the adoption of agentic payments. The two central proposals rest on supervised UK payment practice: the request key on the Open Banking idempotency requirement at paragraph 6, and the mandate structure on the variable recurring payment consent at paragraph 40. I would resist any design of the next-generation infrastructure that named or privileged a particular agent-payment protocol, including any in which I have a commercial interest. The paper cited below argues a design position for True Primary's own product surface. Asks 3 to 5 are also properties that paper models. Ask 3's replay behaviour is already required in supervised UK practice, at paragraph 6, except that it returns a current status rather than the recorded result, so the rest can be tested against that practice rather than against my paper. Carriage of the key into the core is not, and rests on the arguments at paragraphs 9 and 11. Ask 4 extends that requirement to the core, and ask 5 has no supervised precedent I could find, as paragraph 39 records, so both rest on the arguments at paragraphs 9 and 24.

3. No part of this response is confidential, and I am content for it to be published in full, attributed to me. The paper: King Yew Choo, "An MPP-First, Settlement-Flexible Architecture for Self-Serve Agentic Expert-Access: Protocol Analysis, Formal Model, and Business-Model Transition", working paper, SSRN Abstract 6928639, June 2026.[2] MPP there is the Machine Payments Protocol, which is not mine.

Within its model the paper proves a gate-before-fulfilment safety invariant, idempotency under a request key, absence of double payment, and end-to-end auditability. The arguments in this response stand without it. I am also the author of an individual Internet-Draft on gating agent-initiated tasks on payment or entitlement.[3] It is not adopted by any working group, and this response does not rely on it either.

Question 16: functional requirements for programmable payments at scale

4. I agree with the boundary the consultation draws for programmability. The core should be compatible with programmable patterns without embedding complex logic within it. I answer at the level of properties guaranteed at that boundary. The consultation's enabling patterns are an open list and I endorse them: conditional hold and release of funds, enriched messaging and data fields to allow automation, and clear handling of failure or timeout scenarios, with design choices remaining compatible with settlement finality and legal certainty. Two properties follow. The consultation names failure and timeout handling in that list, and end-to-end traceability and audit trails in its consumer-protection section. Paragraphs 5 to 11 set out the first where instructions arrive; paragraph 12 sets out the second. Q16 also asks about the wider ecosystem. Four of the requirements below fall there rather than on the core: the scheme-defined canonical form at paragraph 5, the scheme duplicate window at paragraph 11, the fallback during an outage at paragraph 13, and separate performance budgets at paragraph 15.

5. **A request key, and a repeat that returns the original result.** A client whose request times out cannot tell whether the instruction succeeded, so retries are routine. The first requirement is idempotency of every payment-initiating instruction. A request key supplied by the client of the participant that submits the instruction, and scoped by that participant, should be recorded with the result for a scheme-stated period. The client here is whoever instructs that participant, which for an agentic payment is the agent's own provider. What the participant carries into the core must be unique within that participant, because that is the scope the core can check. A repeat under that key then returns the original result. The consultation defines certainty of fate as a reliable confirmation to both payer and payee whether a payment has succeeded or failed. A request key is what delivers the payer's half when the sender's request times out and the sender cannot tell whether the instruction succeeded. The same key presented over a different canonical form of the request is refused, and creates or alters nothing. Which fields enter that canonical form must be defined by the scheme rather than by each participant, because a key whose binding each party computes differently is not one key.

6. **What Open Banking already specifies.** It has the replay half of this.[4] Its `x-idempotency-key` is mandatory for POST requests to idempotent resource endpoints, and may be at most forty characters, a longer one being refused. On a repeat the account servicing payment service provider (ASPSP) creates no new resource, and responds with the existing resource's current status, or one at least as current as the ASPSP's own online channels offer, and a 201 Created code. That status is the one current when the repeat arrives, not the result recorded when the first request was answered. On a changed body under the same key it requires the client not to change the body, requires the ASPSP not to modify the end resource, and permits it to treat the change as a fraudulent action. The refusal of the same key over a different canonical form, at paragraph 5, is therefore my addition rather than a restatement.

7. **Why the identifiers already in the message do not do this.** ISO 20022 messages already carry five transaction identifiers. EndToEndIdentification is assigned by the initiating party, is mandatory in a customer credit transfer, and passes unchanged along the whole chain. The standard states its use as reconciliation and linking tasks relating to the transaction, and gives it no uniqueness scope.[5] It is not reliably unique in practice either: the SEPA usage rule requires the literal `NOTPROVIDED` where the customer gave no reference, so all traffic with no customer reference carries one identical value in that field.[6] InstructionIdentification is optional and explicitly point to point, meaningful only between one instructing and one instructed party. TransactionIdentification is assigned by the first instructing agent and carries the only uniqueness obligation of the five, and that obligation binds the sender, not the receiver.[5] The UETR is a version 4 universally unique identifier, optional in the standard itself and made mandatory by scheme usage guidelines, including CHAPS and TARGET. ClearingSystemReference is optional and is assigned by a clearing system to identify the instruction unambiguously, so it is the system's own reference rather than anything a sender could supply, and it carries no receiver obligation.

8. The gap is a receiver obligation, and it is absent by design. Across the full message definition report for payments clearing and settlement, the words "duplicate" and "idempotent" do not appear once in any of its element definitions.[5] ISO 20022's own best-practice guidance says as much, warning that anyone reusing these identifiers for linking must assume they are not already used for other purposes in the payment system, such as duplicate control.[7] Duplicate control is therefore a property of the system, not of the identifier, which is exactly the choice I am asking the next-generation infrastructure to make explicit.

9. **What the constraint can and cannot guarantee.** The key guarantee is application-level, and it holds only at the interface enforcing it. Two limits follow, and both matter more than the guarantee itself. First, a duplicate sent under a fresh key meets no key check at all. Second, and this is the harder limit, the core cannot detect that duplicate by inspecting the payment. Two legitimate purchases can carry the same payer, payee, amount, currency and mandate reference. A rule that treated that tuple as evidence of a retry would refuse a payment the payer

intended. Detecting a semantic duplicate under a fresh key is therefore a risk control for the submitting institution and the product, not a property the core can supply. What the core can guarantee is narrower and worth asking for: **one request key and one canonical instruction should produce at most one clearing outcome**. That property holds only where the check and the recording of that check's result are one indivisible step, and only while that record is kept, for the bounded window paragraph 14 requires. Where they are separate steps, two concurrent copies of one instruction can each pass the check before either is recorded, for the same reason the payment authority at paragraph 23 prevents a second payment only where its consumption is atomic. How that is achieved is an implementation choice. The consultation's glossary defines the core as clearing and messaging that determines inter-participant obligations before settlement, and its conceptual architecture places settlement in a separate layer.[8] I am not asking for a guarantee in that settlement layer. The constraint belongs where the obligation is first determined.

10. **Which layer the constraint belongs in.** The consultation says the core would focus on a minimum set of essential functions. The first is clearing and settlement co-ordination, which it describes as determining payment obligations between direct participants, supporting clearance and irrevocability, and generating outputs for settlement in central bank money as the ultimate risk-free asset. The third is transaction processing and controls, described as validating transactions, managing batch and real-time processing, and supporting transaction lifecycle management.[8] The constraint I ask for is a validation, so it belongs in the third, applied to the instructions from which the first determines obligations. I am conscious that a minimum set is meant to stay minimal, and I am asking for one constraint inside a function the consultation has already named, not for a new one.

11. **Three further points keep the ask honest.** This constraint stops a second clearing outcome; it does not return the original result to a timed-out sender, which is the same limit I identify in scheme duplicate handling at paragraph 36. I accept it here because the request key already answers that sender. The core-level constraint is the backstop for the cases a participant's own interface cannot cover: an idempotency store that has failed or partitioned, a race across a participant's own gateways, and a participant that does not enforce the key at all. And what the core cannot refuse, it can still report. Telling the submitting participant that an instruction matches a recent one on payer, payee, amount, currency and mandate, inside a scheme window, leaves the decision with the party able to judge it. That report is not one of the five asks, and it is a larger ask than the exact-key check costed at paragraph 14, because a match on five attributes needs an index of its own, and it sits with the wider-ecosystem requirements at paragraph 4.

12. **An integrity-protected record, and what a cleared instruction means.** A programmable flow executes automatically after set-up, so user trust depends on what the payer can check afterwards. The second requirement is a record produced with the transaction. It binds the parameters the payer authorised, the payment that resulted and its final execution status, and it is checkable against the

institution's own authority, instruction and outcome records. The core carries the authorising reference unaltered and makes the clearing leg's record retrievable. Binding that leg to the authority the payer gave sits with the institutions holding that authority. Legal certainty where execution is automated rests on the same record. A cleared instruction means the core has determined the inter-participant obligation. It does not mean settlement has completed, and a response that presented it as completed settlement would mislead the payer.

13. **And when the automation does not hold.** Both properties must hold when the automation does not. An instruction whose conditional logic cannot be evaluated should fail closed. The ordinary authenticated instruction should remain available as the fallback, so an outage in the programmable layer does not remove the payer's ability to pay. That fallback deserves one caution, because it creates the very duplicate paragraph 9 says the core cannot catch: it carries a fresh key and no mandate reference. It should therefore reference the instruction it replaces, and the replaced instruction should be cancelled, or marked at the party that would execute it, before the fallback is sent. That ordering stops a second execution only where that cancellation or mark is durable at the party that would execute the replaced instruction, and mutually exclusive with executing it, before the fallback goes. An instruction marked only at the sender's side, or cancelled while the fallback is already in flight, leaves both live and both can execute. The fail-closed outcome belongs in the execution status the record carries.

14. **The cost of this at scale.** The consultation notes that conditional and event-driven payments can increase demands on performance, resilience and availability, and that throughput choices will shape which programmable patterns can be supported reliably at scale.[8] I agree, and the properties above are not free. A uniqueness check adds a write and a lookup to the critical path of every payment-initiating instruction. Its key should cover the submitting participant's identifier and the request key that participant carried, held for a bounded window, for which Open Banking's 24 hours is a defensible starting point subject to the floor below.[4] Because that key is composite on the submitting participant's identifier, the store partitions by participant, and the indivisible step at paragraph 9 is local to one partition rather than global. The window should then be set from the retry behaviour a scheme actually observes. It must outlast the longest period an instruction can stay live without a final result, or a repeat under a key whose record has already expired meets no check at all. That cost is bounded and known. The loss it prevents, a second clearing outcome on one instruction, is neither, and I have no incidence figure for the residual the core-level check catches beyond a participant's own. Conditional processing adds durable state, deadlines and contention on top of that.

15. **Three consequences worth designing for.** Conditional and ordinary workloads need separate performance budgets, so that a surge in one cannot exhaust the other. Admission control should reject an instruction before it acquires any lifecycle, and that rejection must be plainly distinguishable from a refusal after acceptance. Where the programmable layer exceeds its latency budget, the core

should shed conditional work and keep the ordinary authenticated path available, rather than degrade both. I do not propose numerical targets. Those should follow the expected workload and the Bank's own approach to impact tolerances, not a figure invented in a consultation response.

Question 17: capabilities for the safe provision of agentic payments

16. An agentic payment is safe to the extent the ecosystem can bound what a delegated agent may do, verify its authority when cost is committed, and leave a checkable record. Five capabilities carry that weight. They address the challenges the consultation lists, among others: how users understand and control what agents are permitted to do, who is accountable when transactions go wrong, how agent-initiated transactions are identified, and how agents are authenticated. The consultation also raises consent, performance and trust, and five capabilities are not the whole of a safe deployment. None of them requires the core to run agent logic. They are a mandate the system can check (paragraph 18), a gate before cost is committed (paragraph 20), and idempotent execution with an attested agent-initiated marker (paragraph 22). The other two are a checkable record binding authority, payment and outcome (paragraph 24), and an escalation condition a valid mandate cannot bypass (paragraph 27). Paragraphs 19, 21, 23, 25, 26 and 28 carry them further; none of those adds a sixth capability. These five are the capabilities Q17 asks about. The five items I prioritise for the core itself are the ones in the summary above. Three of these five capabilities reach the core: the mandate at paragraph 18 as a carried reference, the marker and the request key at paragraph 22 as carried fields, and the record at paragraph 24. The gate at paragraph 20 and the escalation at paragraph 27 sit with institutions and products. The remaining ask, the uniqueness constraint, comes from paragraph 9 rather than from this list.

17. Accountability is easier to test by role than by layer. One firm may hold several roles, but each duty still needs an owner.

Role	Duty under these capabilities
Agent or user-facing service	Present the authority and the transaction to the principal. Protect its credential. Hold the purchase reference. Its own assertion of identity or purpose is not evidence.
Submitting authorised participant	Authenticate its client. Check the mandate evidence it relies on. Scope its client's request key to itself, record the result against it, and answer a repeat with the original result. Attest the agent-initiated marker it supplies.
Payer's ASPSP	Verify authority and funds. Reserve, commit and release cap headroom. Apply suspension and revocation. Authenticate an escalated approval.

Role	Duty under these capabilities
Core operator	Validate message and lifecycle rules. Enforce uniqueness on one key. Carry the mandate reference, the marker and the request key unaltered. Require and retain the record.
Scheme and standards bodies	Define which fields make an instruction the same instruction, the retention window, access rules and conformance tests.
Regulators	Determine authorisation, liability, reimbursement and redress. No marker in a message settles any of these.

18. **A mandate the system can check.** A human principal should be able to delegate spending authority to a named agent under an explicit, machine-readable mandate. It needs six elements:

    - a **spending cap**, per transaction and cumulative over a stated period;
    - an **authorised scope**, what the agent may do and with whom;
    - an **expiry** at which authority lapses;
    - a **withdrawal of authority** the principal can exercise at any time;
    - the **identity of the agent** whose requests it authorises; and
    - an **escalation condition**, the threshold at which a transaction returns to a human.

    The cumulative cap holds only if the amount is reserved before the request proceeds. Without reservation, two concurrent requests each pass against the same headroom, and the cap fails in the case it exists for. Reservation alone is not enough either. The party holding the cap must also commit the reservation when the payment clears, release it when the instruction fails or lapses, and adjust it on a partial execution, a fee, a currency conversion, a return or a refund. Each of those consumes or restores headroom, and a scheme that leaves them undefined will find its cap drifting.

19. **Who can enforce which element.** The principal's ASPSP can hold the mandate and enforce the elements a payment instruction lets it test: the caps, the permitted payees, the expiry and the revocation. It already enforces variable recurring payment consent parameters in that way (paragraph 40). Two are not testable from an instruction, because the purchase itself sits outside it: the half of the authorised scope that says what the agent may do, and the escalation threshold. Those belong to the party that accepts the agent's request, and the mandate should name the enforcer of each element, because one party does not hold them all. The core's part is one field: a reference in the instruction to the authority it was given under.

20. **A gate before cost is committed.** Every step that commits cost or cannot be freely reversed should pass one authorisation decision. That decision validates, together, the agent's control of a credential bound to its identity in the mandate, the request's fit with the authorised scope and caps, the mandate's unrevoked and unexpired status, and an unconsumed payment authority of the kind described at paragraph 22. It may be taken in as many components as an implementation needs, on three conditions. No route may reach a cost-committing step with only part of the check applied. The components must resolve into one decision that every enforcing point treats as authoritative, composed in the same order wherever that decision is formed. And that decision must state how long it remains good for, after which it is retaken, because a mandate unrevoked when the gate ran may be revoked by the time the transfer is created. Scheme rules should say whether a revocation arriving in that interval stops the instruction or only the next one.

21. **One element needs its enforcement point named.** Where an agent instructs an authorised provider rather than the payer's own institution, that institution never sees the agent and cannot test its credential. The credential check therefore belongs to the party that accepts the agent's request, which should attest in the instruction that it verified control of a credential bound to the agent named in the mandate. Where the settlement method can record a transfer, a card authorisation or a capture outside that decision, money may move before the gate completes. Fulfilment should still wait for it, with the financial result reconciled afterwards. A check after settlement cannot prevent the transfer, and redress then depends on a return the agent's counterparty may be unable to make. The decision belongs where cost is committed, usually at the principal's ASPSP or the service being bought. The core should not require a cost-committing step to be recorded before those checks can complete.

22. **Idempotent execution, and an agent-initiated marker that is attested.** Retries are routine for the software beneath an agent, so the infrastructure should support the request key of paragraph 5 on every payment-initiating instruction. Separately, the agent identifier says whose requests the mandate authorises. Marking a transaction as agent-initiated needs its own indicator, because a receiver that infers the initiation method from an identifier is guessing. That marker should be attested by the authorised participant submitting the instruction, not asserted by the agent, and it describes how initiation occurred rather than transferring any regulatory duty to software. The gate at paragraph 20 also needs a payment authority that is valid for one economic purchase and cannot be replayed. I state it that way rather than as a single-use token, because a card authorisation and its later capture are two cost-committing steps and one token cannot be unused at both.

23. **How that authority has to behave.** The authority should be issued for the purchase, tested at authorisation, and tested again at each capture against its recorded consumption and its remaining uncaptured amount. It prevents a second payment only where consumption is atomic with the payment it authorises; an authority checked and then spent in two steps can be spent twice. Whether that authority

sits in the core or with the payer's ASPSP is an implementation choice, and it is not among the items I prioritise for the core at paragraph 49.

24. **A checkable record binding authority, payment and outcome.** Every agentic payment should yield a record binding the authorising mandate, the payment that resulted and, by reference only, the service it paid for. A refusal or a hold at the gate should yield the same record, the events a disputing principal most needs evidence of. A hold can reach it through the execution status at paragraph 12. A refusal never becomes a cleared instruction and cannot reach a record kept at the boundary, so the refusing party should retain it under scheme rules, linked by the same mandate reference. For refusals I accept that the record then sits with a party to the dispute, and the independence argument below does not reach them. Keeping refusals with the refusing party also keeps instructions that never became payments out of the infrastructure-held record, consistent with the limits at paragraph 25. Evidence a disputing party can rely on is stronger for being held where the party whose conduct is in question can neither delete it nor amend it. A record required and retained at the infrastructure boundary achieves that, and it is the argument for placing the requirement in the core. I accept that this is the only one of my five that adds to a core function rather than constraining one. It is worth being precise about what this buys. The binding is produced by a participant, so retention at the boundary protects the record's survival and integrity, not its truthfulness at the moment it was made. Scheme rules should therefore require that participant's binding to be signed and independently checkable. The consultation puts the alternative placement directly. It asks how far certain services should be (a) delivered within the core infrastructure or (b) provided separately by ecosystem participants, and its illustrative list includes identity management, covering the management of delegated authorities, and dispute management.[8] A mandated dispute-management service would satisfy the independence argument above if it were compulsory, held the record outside the participants whose conduct is in question, and carried the same access and retention limits. I prefer the boundary for coverage: every cleared instruction crosses it, so the record exists without a separate adoption decision product by product. The record settles what evidence a dispute can draw on. Who bears the loss remains a question for scheme rules and regulation, as paragraph 26 sets out.

25. **What that record must not become.** This is the strongest objection to the previous paragraph, and it deserves an answer. A mandate reference carried end to end through clearing is a linkable identifier across a payer's agentic activity, and a retained authority-and-outcome store is personal data. Three limits belong in the design from the start. The record should hold references and identifiers, not transaction content, purchase descriptions or any behavioural profile; the mandate's own terms stay with the institution enforcing them. Retrieval should be limited to the parties to a dispute and the relevant supervisor, on a stated trigger, with queries logged and the identifier space protected against enumeration. Retention should be tied to the dispute window and bounded there, and the request-key record that prevents a replay needs a shorter life than the evidential record.

26. **And the law it has to answer to.** Pseudonymising the reference does not put it outside data protection law: the Information Commissioner's Office treats pseudonymised data as personal data in the hands of anyone holding the additional information needed to reverse it.[9] Two principles bear directly on the design, and one of them no longer reads as it did. Data minimisation, at Article 5(1)(c) of the UK GDPR, is unamended. Purpose limitation, at Article 5(1)(b), was rewritten with effect from 5 February 2026 by section 71 of the Data (Use and Access) Act 2025, which also moved the compatibility test for further processing into a new Article 8A.[10] The same Act also replaced Article 22 with Articles 22A to 22D on automated decision-making, by section 80, in force from the same date. A refusal or a hold taken at the gate with no person present is a candidate significant decision under Article 22A, and the escalation condition at paragraph 27 is where a data protection analysis would look for meaningful human involvement. A record specified now should be specified against the current text. A design that adopts this record without those limits builds an infrastructure-level registry of who delegated what to which agent, which is not what I am asking for. Standardising the record's content, access and retention makes audit and dispute evidence portable. Allocating liability, and allocating the cost of holding the record, remain work for scheme rules and regulation.

27. **An escalation condition a valid mandate cannot bypass.** Some requests should never execute automatically. The gate should apply a conditional hold when the escalation condition is met. It is released only on an authenticated, transaction-specific approval, given by the principal or by a delegate the principal nominated in the mandate for that purpose, and it lapses unapproved if none arrives inside a stated period. Setting the threshold, deciding which requests qualify and reaching the principal belong to the product or access layer. What I ask of the core is narrow and consistent with paragraph 49: its design should not preclude an instruction being held unexecuted pending a release, and the release or the lapse should appear in the execution status it already carries (paragraph 12).

28. **The race at the deadline.** The harder problem belongs to scheme rules. Lapse and release must be mutually exclusive, one clock must be authoritative for the deadline, and whichever outcome is durably recorded first must stand, with the other answered by the recorded outcome and creating no second event. A principal whose approval arrives too late must be told it did not take effect. Authenticating the approval sits with the payer's ASPSP, which already authenticates the principal and would need an equivalent enrolment before a nominated delegate could approve. Approval is an added condition and cannot cure a failed authority check. An expired mandate, one the principal has withdrawn, or a request over its cap or outside its scope stays refused, whoever approves it.

The sub-question following Q17: barriers to development or adoption

29. Five barriers stand between these capabilities and their adoption at scale. They are set out at paragraphs 30, 33, 34, 38 and 39. Paragraphs 31, 32, 35, 36 and 37 develop those five rather than adding to them. I set out the five, then an existing supervised UK structure that offers a partial adoption path.

30. **Agent mandate schemas exist, and three gaps remain.** Machine-readable delegated mandates for software agents are now published. Google announced the Agent Payments Protocol on 16 September 2025, published version 0.2.0 on 28 April 2026 under the Apache 2.0 licence, and donated the protocol to the FIDO Alliance on the same date.[11] Its two mandate objects, the Checkout Mandate and the Payment Mandate, each defined in an open and a closed stage, carry four of the six elements at paragraph 18:

    - the **agent** whose requests they authorise, bound by its public key in a `cnf` confirmation claim the specification requires while a mandate is open;
    - **caps**, as a budget and an amount range;
    - **scope**, as allowed payees, merchants, payment instruments and payment initiation service providers; and
    - **expiry**, as an `exp` claim and an execution date bounded by a not-before and a not-after.

    The card networks have moved in parallel. Mastercard announced Agent Pay on 29 April 2025 and Agent Pay for Machines on 10 June 2026, and Visa announced Intelligent Commerce on 30 April 2025, each setting spending limits and conditions on a tokenised agent credential.[12]

31. Three gaps remain at the point where the next-generation infrastructure would have to rely on this work. Two are in this paragraph and the third is at paragraph 32. Version 0.2.0's specification documents define no mandate revocation status and no escalation condition. The release does ship a checkout status value `requires_escalation` and a sample credential-revocation tool.[11] Neither is a property of a mandate that a third party could rely on, which is what a payer, a payee and the institutions between them would need.

32. **And no one is yet required to follow the protocol.** It is a vendor-authored specification in donation to a standards body, and as at 2 September 2026 the FIDO Alliance has published no agentic payments specification, only working groups. The Alliance's own account of that work states that trust frameworks for agent-driven interactions do not yet exist.[13] Nothing here is yet a ratified standard that a payer, a payee and the institutions between them are obliged to interpret identically. It is also agnostic to the payment instrument used and places the commerce protocol outside its scope. Beyond naming which providers may facilitate a payment, it maps onto no account-to-account scheme message, finality rule or rulebook obligation. The supervised mandates I examined each carry

revocation and each sit inside one product or scheme: the direct debit instruction, the continuous payment authority on cards, and the Open Banking variable recurring payment consent. None is interpreted outside its own scheme. What the core determines is whether an instruction can carry a reference to a mandate of either kind, and whether the conditions that mandate expresses can be validated at the boundary at all.

33. **Safety guarantees coupled to individual rails.** On the public evidence, agent-payment capability is being built rail by rail. The announcements I can cite are on the card networks, and the most recent of them qualifies that reading: Mastercard's Agent Pay for Machines of 10 June 2026 is announced as supporting reliable, guaranteed multi-rail settlement across cards, accounts and stablecoins.[12] That is one network's own commitment across payment types, and it binds no other scheme's rules. The Agent Payments Protocol sits above the rails and settles nothing itself, so the guarantees a payment actually receives remain the rail's own. Card rails, account-to-account rails and that protocol layer share a common minimum: whether the agent's request is within authority, whether the instruction is fresh and whether it has already executed. Beyond that, credential models, revocability, finality, recourse and the allocation of legal responsibility all differ. Answering the common minimum rail by rail fragments the guarantees. The mandate check can be specified once and applied on any rail, with the payment authorisation still bound to the settlement instruction and the scheme rules governing it. The core requirement here is a negative one. Its instruction formats and validation rules should not bind agent authorisation to a single settlement method, a constraint that payments across different forms of money would need in any case.

34. **Inconsistent idempotent retry semantics.** Open Banking's replay behaviour at paragraph 6 is what paragraph 22 asks for, except that it returns a current status rather than the recorded result, so my point here is about coverage and consistency. The Open Banking specifications state that idempotency is used sparingly within them, and that outside the designated endpoints the institution must ignore the key. The idempotent treatment binds only where the same key was received from the same provider in the preceding 24 hours.[4] Beyond that window the ASPSP is not required to recognise the repeat, so the same instruction can be treated as a new one. As at 2 September 2026 the general pattern remains unstandardised. The IETF httpapi working group adopted "The Idempotency-Key HTTP Header Field", and revision -07 of `draft-ietf-httpapi-idempotency-key-header` expired on 18 April 2026 without becoming an RFC.[14]

35. **Duplicate handling between banks is a scheme decision, not a standard one.** ISO 20022 supplies constructs for identifying a duplicate, and its prescribed handling is narrow and conditional. Its Business Application Header Message Usage Guide, version 3.0 of 3 November 2021, is distributed in the current Business Application Header message set alongside `head.001.001.04`.[15] Section

2.10 addresses the sender who suspects the receiver never got the message. Where that sender resends the same message with the possible-duplicate indicator set and the original identified in the Related element, and the receiver did hold the original, the guide directs that the resent message must be ignored. That is one narrow case, and schemes answer the general one differently.

36. **What the schemes actually do with one.** Payments Canada's Real-Time Rail ISO 20022 companion leaves the copy duplicate and possible duplicate indicators optional, with no impact on processing and content that is not validated.[16] Where a duplicate is caught and answered under the SEPA Instant Credit Transfer scheme, the outcome is a reject carrying reason code AM05, Duplication.[17] That guidance covers SCT Inst alone, where AM05 admits a reject and nothing further. Under SEPA Credit Transfer the same code admits a reject or a return, and under SEPA Direct Debit a reject, a return or a reversal, each set out in that scheme's own guidance.[18] Ignoring the message, rejecting it, returning it and reversing it all stop or unwind the second payment. None of them returns the original payment's result to the sender, which is what a timed-out client needs to resolve its retry. Duplicate handling at that layer is a scheme decision, so a retry meeting no scheme control can settle again.

37. **What I could and could not establish on the UK rails.** The like-for-like mapping the CHAPS migration was designed around placed AM05 Duplication in the return-code table, so a duplicate could be returned after settlement.[19] The mapping does not say whether one could instead be suppressed before it. I have not verified the position in the current CHAPS schemas. Faster Payments is further away still. Its published principles specify it in ISO 8583 throughout and carry none of the five identifiers at paragraph 7 as ISO 20022 elements. Pay.UK separately publishes an ISO 8583 to ISO 20022 mapping whose landing page describes the central infrastructure as one that uses ISO 20022 messages, and I record that tension rather than resolve it. The nearest analogue Faster Payments can carry is the 31-character end-to-end reference in tag 62, and the Open Banking Read/Write API records that the Faster Payments Scheme can access only thirty-one characters of the `EndToEndIdentification` field that API defines. Pay.UK's published principles treat the provision of that reference as competitive rather than mandated, and address retries only as insufficient-funds re-attempts.[20] They do not address retry after an ambiguous result at all. I could not establish from public documentation whether the Faster Payments scheme rules define duplicate handling at message level, because the message specifications sit behind a login. I record that as a limit on what I checked rather than as a finding about the rules.

38. **Unclear liability where no human is present at the moment of payment.** The Payment Services Regulations 2017 and scheme rules already govern authorisation, unauthorised transactions and disputed payments, and they apply to a delegated-mandate payment as to any other. Where a variable recurring payment runs under an exemption from Strong Customer Authentication (paragraph 41), the parameter test determines authorisation (paragraphs 40 and 42). The open question sits before it. It is whether a mandate obtained by

deception, or one whose agent was manipulated or whose credential was compromised, is valid at all, given that the principal authorised the mandate and no person authorised the individual payment. In my assessment that question slows adoption more than the technical gaps above. I expect institutions to be reluctant to carry delegated-agent traffic at scale while loss allocation for a misused mandate is unclear. One live UK regime already allocates loss of a neighbouring kind on these rails, and it sharpens the question here rather than answering it. Since 7 October 2024 a reimbursement requirement has applied to authorised push payment fraud on Faster Payments and on CHAPS, with a maximum of £85,000 per claim as at 2 September 2026.[21] The Bank of England maintains the CHAPS reimbursement rules as that system's operator. The requirement reaches a payment the consumer was deceived into authorising, and payment initiation service transactions are in scope on Faster Payments. What it does not settle is the case here. Where the principal authorised a mandate months earlier and no person authorised the individual order, it is unsettled whether the authorisation the consumer was deceived into granting is the mandate or each order made under it. That answer decides whether the claim sits inside the requirement or falls back to the Payment Services Regulations. The capabilities at Q17 record what was authorised, what was checked and what settled. The allocation itself is work for regulators and scheme rules.

39. **No common evidence profile across platforms.** Payment messages and scheme rules already carry dispute evidence. I have found no common way to link the mandate, the agent that acted, the payment identifiers and the final status of the payment. Without that linkage each platform rebuilds dispute handling and audit for itself. The evidence does not travel between the parties to a disputed agentic payment, and the protections discussed at Q18 rest on whatever a given platform chose to record.

40. **An adoption path built on a structure the UK already supervises.** A UK Open Banking variable recurring payment consent is a delegated authority an account holder grants to an authorised payment initiation service provider. Under it, payments are initiated with the account holder absent where the ASPSP applies an exemption from Strong Customer Authentication. Where authentication is applied to each payment instead, as at paragraph 41, the payer is in session for each one. The consent carries a maximum individual payment amount, and one or more periodic spending limits, each taking one of six period types: day, week, fortnight, month, half-year and year.[22] The ASPSP must ensure a payment instruction adheres to those control parameters. That parameter test applies to each instruction, and a breach carries error `U014`.[23] Open Banking Limited's knowledge base states that `ValidToDateTime` may be left blank, making validity indefinite, so the lapsing expiry at paragraph 18 is an addition rather than a restatement.[24] The Open Banking Standard does not say how concurrent instructions are ordered against one periodic limit, or whether headroom is reserved before a request proceeds, so the reservation paragraph 18 requires is a second addition rather than a restatement.[23]

41. **How authentication works inside it.** Strong Customer Authentication applies to each individual payment, or once when the authority is established, where the ASPSP applies the trusted beneficiaries exemption at Article 13 of the technical standards on strong customer authentication.[25] Article 13(1) requires Strong Customer Authentication when the payer creates or amends the list of trusted payees, and Article 13(2) permits the ASPSP to exempt later payments to a payee on that list. Article 2 requires transaction monitoring in either case. Article 13 is not the only exemption in play. Open Banking Limited's own variable recurring payment proposition names transaction risk analysis and low value as alternatives an ASPSP may apply instead, and permits a transfer-to-self exemption where a sweeping payment goes to an account held at the same ASPSP.[26] Open Banking Limited's Customer Experience Guidelines describe a further route, delegated Strong Customer Authentication, applied to each individual payment by the payment initiation service provider or another party to whom the ASPSP has delegated it.[26] It is optional and rests on a contract between the two, so the ASPSP remains the delegating party.

42. **What follows if a payment falls outside those parameters.** The point is worth stating carefully, because "parameters" is the Open Banking Standard's term and not the statute's. Regulation 67(1) of the Payment Services Regulations 2017 provides that a transaction is authorised only if the payer consented either to that transaction or, at limb (b), to the execution of a series of payment transactions of which it forms part. Where the payments run under an exemption from Strong Customer Authentication, the payer's consent is given once, at set-up, to the series and its parameters, so I take limb (b) to be the one in play. Where the consent is bounded, a payment outside those bounds is not within the series consented to, and so on that reading is not authorised. The payer may withdraw that consent at any time under regulation 67(4), subject to regulation 83(3) to (5), with prospective effect on future transactions. Open Banking Limited reads both of its routes as giving explicit consent for each payment, the exemption route through the consent parameters and the delegated route through authentication at each payment, and I do not settle which limb the delegated route falls under.[26] Within Part 7 the Regulations use revocation for revocation of a payment order under regulation 83, and withdrawal for consent under regulation 67. The word carries a different sense at regulations 149 and 157 and in Schedule 9, where it means the revocation of an earlier instrument.

43. Two developments bound how far that path currently reaches. Sweeping remains the only use case for which the CMA mandates variable recurring payments under its Retail Banking Market Investigation Order 2017. That rests on the definition the CMA agreed on 26 July 2021 and clarified on 14 March 2022, under which the transaction must be between two accounts belonging to the same person or legal entity, and the destination must serve the Order's objectives.[27] The CMA narrowed it again on 3 December 2025, restricting mandated sweeping access for credit repayment to destination credit accounts having characteristics similar to an overdraft, that is a revolving facility with no fixed repayment schedule.[28] Sweeping to other destinations is untouched. Separately, a commercial scheme launched on

2 June 2026, when UK Payments Initiative Ltd announced its variable recurring payments scheme.[29] The FCA and the Payment Systems Regulator record its first phase as covering regulated financial services, regulated utilities, and local and central government.[30] The FCA said on the same date that, subject to legislation expected to give it new powers, it will consult on a long-term regulatory framework by the end of 2026.[31]

44. Providing payment initiation services requires authorisation; registration alone does not extend to it. Regulation 138(1) of those Regulations restricts the provision of a payment service to a listed set of persons, several of them registered rather than authorised, and 138(2) makes a contravention a criminal offence. What bars a registered small payment institution is regulation 14(4), which makes the absence of payment initiation services a condition of that registration. The technical-support exclusion at Schedule 1 Part 2 paragraph 2(j) expressly excludes payment initiation services. The Regulations impose their duties on persons, and the duties here fall on the authorised provider. Where the activity is a payment service, initiation must remain attributable to an authorised provider, whether it acts directly or through an agent included on the register under regulation 34, in the regulation 2(1) sense. What generalises is the consent. The agent selects within the mandate and instructs an authorised provider, which initiates the payment and carries the regulatory duties. The mandate names the agent; the consent of record stays with the provider. The requirement this path places on the next-generation infrastructure is narrow: carry the mandate reference and an agent-initiated marker distinct from the agent identity, and leave the parameter check with the ASPSP.

Question 18: consumer protection and fraud capabilities to prioritise

45. **Where the mandate reference and the request key would sit.** A design team will ask, and the answer is not obvious, so I set out what I have been able to establish. For the mandate reference, `MndtId` is the closest existing element. It is optional, one to thirty-five characters, and it is not confined to direct debits: the same tag appears in the customer credit transfer and the interbank credit transfer under a distinct registered component.[32] The component it sits in is shaped for a creditor-to-debtor mandate: its siblings are the date of signature, the date of verification, the first and final payment dates, the frequency and the setup reason.[32] A payer-to-agent delegation has neither role, so reusing it would rest on scheme convention and not on the standard's own semantics. The request key needs no new element. Paragraph 7 records that `InstrId` is optional and point to point between one instructing and one instructed party, which is the scope the constraint at paragraph 9 needs. `InstrId` is also one to thirty-five characters. The value carried is the participant's scoped key rather than its client's, which may be longer: the Open Banking requirement at paragraph 6 permits a key of up to forty characters, and a version 4 universally unique identifier is thirty-six. A scheme should therefore state how the participant derives the carried value, because a derivation each participant chooses differently is not one key. `ClrSysRef` is assigned by the

clearing system itself, not carried in from the sender, so it cannot hold a client-supplied key. What is missing from `InstrId` is the receiver obligation of paragraph 8, and a scheme can add that by usage rule, which is how SEPA already adds a sender-side uniqueness obligation to the transaction identification.[6] Of the three fields the core carries, only the marker needs a registration of its own.

46. **The initiation marker has no home at all.** The nearest element, `InitnSrc`, names the application or software used to initiate a payment. It sits only in the group header of the customer-to-bank message and is absent from the interbank messages, so it dies at that boundary, and it names the product rather than saying whether a person authorised the payment.[32] Nothing in the purpose, local-instrument or category-purpose code sets expresses agency; the few initiation-related codes there mark a channel and are scheme-specific. No code in the External Code Sets release of 28 August 2026 expresses agency, across its 366 purpose, 115 local-instrument and 49 category-purpose codes.[33] `SupplementaryData` can carry anything, but it is a weak home for a permanent attribute: eighteen extensions are registered in total, thirteen of them securities corporate actions, and it needs approval, re-registration against each message version, and prior agreement by every party involved.[34] The consultation names this same gap when it asks how agent-initiated transactions are identified. My answer is that the marker would need a registered element of its own, and that registration sits with ISO 20022 and the scheme.

47. **Two of my five asks are already on the Bank's own list.** Under consumer protection, fraud and wider financial crime, the consultation sets out four design considerations with twelve illustrative examples.[8] It draws a line there that I have kept to throughout. The infrastructure "is not intended to establish a new consumer protection or reimbursement regime, nor to alter existing liability arrangements", and a clear separation should hold between capabilities the infrastructure may enable and the scheme rules and regulation that determine how protections apply in practice. Two of the twelve examples are things I ask for, in the Bank's own words, a third names the execution status my record depends on, and a fourth names the fallback I set out at paragraph 13.

48. **The four, in the Bank's words.** The fourth consideration lists "Clear attribution of the initiation method (for example, user initiated versus not), supporting appropriate fraud controls and consumer understanding". That is the agent-initiated marker at paragraph 22. The third lists "End to end traceability and audit trails across interoperable payments, enabling quicker investigation and recovery", and "Standardised status messages and timestamps, reducing ambiguity about execution, failure or delay and supporting faster resolution of errors and disputes". Those are the record at paragraph 24 and the execution status at paragraph 12. The fourth consideration also lists "Fallback and recovery mechanisms where automation or conditional logic fails, helping to prevent small errors from escalating into consumer harm". That is the fail-closed fallback at paragraph 13. What I add to them is the mandate reference that makes the attribution checkable, and the request key that makes the status recoverable.

49. **The parity standard, and what I would prioritise.** The fourth consideration sets a test I would adopt. It lists "Limits, controls and monitoring that apply to programmable or agentic payments as robustly as to user-initiated payments". Everything at Q17 is one reading of what that parity requires once the payer is a delegated agent. I address Q18 only as it concerns agentic payments. I would prioritise the five items in the summary above, and the ordering is a judgement rather than a measurement. I do not rank the first three against one another: they are one group, and the constraint and the record follow it in that order. The first three are fields the core carries, and they cost least because they require the core to interpret nothing. The uniqueness constraint follows because it is the one property that only the core can hold. The record comes last of the five because it is the most expensive and the most privacy-sensitive, and paragraph 25 sets the limits it needs. The conditional hold at paragraph 27 is deliberately not in this set, because it is a product-layer control, and what I ask of the core is only that its design not preclude one.

50. **Bounded authority and protection by default.** A mandate's cumulative cap and expiry bound the direct payment loss from a compromised agent before any monitoring acts, provided cumulative spend is reserved atomically across concurrent instructions. Where that reservation holds, a stolen mandate admits no more than its cap in one cap period, and nothing after its expiry. Where it does not, two concurrent instructions each pass against the same headroom and the bound fails in the case it exists for. Whether that reservation holds on the adoption path at paragraph 40 is not something the Open Banking Standard states. Even where the reservation holds, the bound is narrower than it sounds. It does not cover a duplicate that escapes the key check, nor a refund that restores headroom (paragraph 51), nor non-monetary harm. Nor does a cap bound who is actually paid: a request inside the mandate's scope can still reach a payee that is not the party the principal intended, or one the payer's institution may not lawfully pay. Those risks sit with the payee-verification and financial-crime controls in the first consideration, which apply to an agentic payment as they do to any other. I do not add to them.

51. **The refund loop.** Where scheme rules let a refund restore headroom, a compromised agent paying a payee it controls can pay, refund and pay again. The cap then bounds each cycle rather than the period: gross outflow becomes the cap multiplied by the cycles completed, and the principal's protection rests on every refund proving good. Scheme rules should state whether a refund restores headroom, because both answers cost something: restoring it reopens that loop, and not restoring it penalises a consumer whose legitimate purchase was refunded. Nor does the cap replace the two controls a mandate needs once compromise is suspected: immediate suspension, and withdrawal of the authority, which for a consent given to a payment initiation service provider is already exercisable on the terms at paragraph 42. Each should reach every point enforcing the mandate before any further instruction executes. Suspension and withdrawal both require someone to notice and act, while authority lapses at the expiry with nobody acting.

52. **Human approval, replay resistance and checkable evidence.** An escalation condition routes a qualifying request to the principal before execution, catching a large, authorised-looking purchase the principal never intended. Being within the mandate's cap and scope does not bypass it. Operationally this resembles authorised push payment fraud, with the deception aimed at the agent. Whether such a payment is authorised for reimbursement purposes is one of the unresolved questions at paragraph 38, and the escalation is worth its place whichever way that is resolved, because it acts ahead of any allocation of loss. On the technical side, the request key returns the original result on a repeat, and the payment authority of paragraph 22 cannot be spent twice where its consumption is atomic. The checkable record then gives a defrauded consumer and their payment service provider a trail across the institution's authority, payment and outcome records. None of these controls determines liability or asks the infrastructure to adjudicate disputes.

53. **A minimum baseline across platforms.** Where agentic payments are offered to consumers, scheme and product rules should set the protections above as a baseline: authority bounded by cap, scope and expiry, promptly suspendable and withdrawable; an escalation condition; duplicate-safe initiation; and a checkable record. Without a common baseline, a consumer's protection depends on which agent platform they chose.

The Equality Act question: impact on persons who share protected characteristics

54. **The Equality Act observations.** The consultation asks respondents to indicate whether any proposal is likely to impact persons who share protected characteristics. I make three observations, and all three sit inside the questions I answer. My first observation concerns delegation as an accessibility route. Delegating routine payment to an agent could improve access for disabled and older consumers in particular, and it is a benefit to test with affected users rather than to assume. My second observation concerns the approval and renewal routes. The same design can exclude them where an escalation condition requires a prompt response through a single digital channel, or presses a user into approving without time to consider. That engages age and disability. The approval route should therefore admit a nominated delegate, a longer response window and a non-agent alternative, and those routes should be tested with affected users.[35] The lapsing expiry at paragraph 18 keeps protecting a user who cannot monitor a delegation continuously, because authority ends with nobody acting (paragraph 51). It cuts the other way where re-establishing that authority is itself burdensome, so any renewal route needs the same accommodations as the approval route above. My third observation concerns an ask of my own. The agent-initiated marker at paragraph 22 records how a payment was initiated, and the consultation pairs that attribution with "supporting appropriate fraud controls" (paragraph 48). Where a payment service provider treats the marker alone as a reason to decline or to score a payment more harshly, the cost falls first on the consumers for whom

delegation is the accessible route to paying. Scheme rules should therefore state that the marker is an input to fraud controls and never a sufficient reason to decline, and should require agent-initiated decline rates to be monitored. The attestation records how initiation occurred, and it carries no information about the principal's own risk. These three observations concern the agentic-payment direction the consultation sets out and the controls I propose for it. I make no observation on the consultation's other proposals.

Closing

These capabilities are the machine-checkable form of controls that established payment practice already applies. Expressing them where instructions arrive would let agentic payments rest on guarantees common to every participant. I have tried throughout to be clear about what the core cannot do: it cannot tell a repeat from a second legitimate purchase sent under a fresh key, it cannot infer a human purpose, and it cannot decide who bears a loss. I would be pleased to provide further detail on any of these points.

References

1. The Payment Services Regulations 2017, S.I. 2017/752, as amended. The provisions relied on in this response are regulation 2(1) at paragraph 1, regulations 67(1), 67(4) and 83(3) to (5) and the word-sense point at regulations 149 and 157 and Schedule 9 at paragraph 42, and regulations 14(4), 34, 138(1), 138(2) and Schedule 1 Part 2 paragraph 2(j) at paragraph 44. https://www.legislation.gov.uk/uksi/2017/752/contents Accessed 1 September 2026.
2. King Yew Choo, "An MPP-First, Settlement-Flexible Architecture for Self-Serve Agentic Expert-Access: Protocol Analysis, Formal Model, and Business-Model Transition", SSRN working paper, Abstract 6928639, posted 24 June 2026. https://ssrn.com/abstract=6928639 and https://doi.org/10.2139/ssrn.6928639 Verified 31 August 2026.
3. King Yew Choo, "A Server-Side Model for Gating Agent-Initiated Human-Sourced Asynchronous HTTP Tasks on Payment or Entitlement", Internet-Draft draft-king-yew-choo-agentic-payments-00, submitted 23 July 2026, expires 24 January 2027. An individual submission, not adopted by any IETF working group. https://datatracker.ietf.org/doc/draft-king-yew-choo-agentic-payments/ Verified 23 August 2026.
4. Open Banking Limited, Open Banking Read-Write API Profile, Read/Write API version 4.0, sections Request Headers and Idempotency. The `x-idempotency-key` header is mandatory for POST requests to idempotent resource endpoints and must not be sent on others. The key "must be at most 40 characters in size", and where a longer one is provided the ASPSP "must reject the request with a status code is 400 (Bad Request)", quoted as written. Where a request repeats a key received from the same third-party provider in the preceding 24 hours, the ASPSP must not create a new resource, and must respond with "the current status of the resource (or a status which is at least as current as what is available on existing online channels)" and HTTP 201 Created. Version 4.0.1, released 1 April 2026, is the current version of the Standard; the provisions cited are identical in both. https://openbankinguk.github.io/read-write-api-site3/v4.0/profiles/read-write-data-api-profile.html Accessed 20 August 2026.
5. ISO 20022 Message Definition Report Part 2, Payments Clearing and Settlement, Maintenance 2025 to 2026, version 1, March 2026, approved by the Payments SEG on 12 January 2026: InstructionIdentification at 10.1.9.2.1, EndToEndIdentification at 10.1.9.2.2, TransactionIdentification at 10.1.9.2.3 and UETR at 10.1.9.2.4, four of the five elements of the PaymentIdentification component in the report's Message Components section; the fifth, ClearingSystemReference, sits at 10.1.9.2.5 and is treated at paragraph 7 of this response (the numbering differs by edition; the December 2020 document, issued for evaluation by the Payments SEG rather than as an approved edition, carries the four at 10.1.10.2.1 to

10.1.10.2.4). A case-insensitive search of that edition returns zero occurrences of "duplicat" and zero of "idempot" across its element definitions, and the same search returns the same result on the December 2020 edition. The four definitions pinned above are character-identical in both. Each definition above is paraphrased rather than quoted. I checked each paraphrase against the ISO Definition reproduced at table indices 2.2 to 2.6 of the pacs.008 message-element table in the European Payments Council's SEPA Instant Credit Transfer Inter-PSP Implementation Guidelines 2025 v1.0. That guide gives InstructionIdentification's XML tag as `InstrId`, its type as `Max35Text` and its ISO Length as 1 to 35. The catalogue page below is a landing page: the report is reached by searching it for the Payments Clearing and Settlement message set and downloading the complete set; inside it the report file is `ISO20022_MDRPart2_PaymentsClearingandSettlement_2025_2026_v1.pdf` as distributed. https://www.iso20022.org/iso-20022-message-definitions Accessed 20 August 2026.

6. European Payments Council, SEPA Instant Credit Transfer Inter-PSP Implementation Guidelines 2025 v1.0 (EPC122-16), table index 2.3 within the pacs.008 message-element table, the SEPA usage rule on End To End Identification: "A customer reference that must be passed on in the end-to-end chain. In the event that no reference was given, `NOTPROVIDED` must be used." The transaction identification at table index 2.4 carries the usage rule "Mandatory. Must contain a reference that is meaningful to the Originator's PSP and is unique over time." https://www.europeanpaymentscouncil.eu/sites/default/files/kb/file/2024-11/EPC122-16%20SCT%20Inst%20Inter-PSP%20IG%202025%20V1.0.pdf Accessed 20 August 2026.
7. ISO 20022 Real Time Payments Group, Best Practices: Linking ISO 20022 messages to Customer Credit Transfer (pacs.008), December 2021: reusing these identifiers to link messages "would have to assume that these are not used for other purposes in the payment system, such as for duplicate control". https://www.iso20022.org/sites/default/files/2021-12/RTPG_BP_Linking_ISO20022_pacs008_December2021.pdf Accessed 20 August 2026.
8. Bank of England, RPIB consultation on the Design of the Future Retail Payments Infrastructure, published 25 June 2026, open until 11 September 2026. https://www.bankofengland.co.uk/paper/2026/cp/rpib-consultation Accessed 20 August 2026.
9. Information Commissioner's Office, guidance on pseudonymisation: "Pseudonymised data is personal data in the hands of someone who holds the additional information." The page carries a notice that it is under review following the Data (Use and Access) Act. https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/data-sharing/anonymisation/pseudonymisation/ Accessed 20 August 2026.
10. Data (Use and Access) Act 2025 (c. 18), section 71, fully in force 5 February 2026 (S.I. 2026/82, reg. 2(d)), having been in force from Royal Assent only so far as it confers power to make regulations (section 142(2)(h)): it rewrote the purpose-limitation principle at Article 5(1)(b) of the UK GDPR, omitted the former Article 6(4) and inserted a new Article 8A carrying the compatibility test for further processing. Article 5(1)(c), data minimisation, is unamended. https://www.legislation.gov.uk/ukpga/2025/18/section/71 Section 80 of the same Act, fully in force 5 February 2026 (S.I. 2026/82, reg. 2(j)), substituted new Articles 22A to 22D for Article 22 of the UK GDPR: Article 22A(1)(a) provides that a decision is based solely on automated processing "if there is no meaningful human involvement in the taking of the decision", and Article 22C requires safeguards for a significant decision taken solely by automated processing, including measures that "enable the data subject to obtain human intervention on the part of the controller". https://www.legislation.gov.uk/ukpga/2025/18/section/80 Both accessed 31 August 2026.
11. Agent Payments Protocol (AP2), version 0.2.0, published 28 April 2026 under the Apache 2.0 licence; announced by Google on 16 September 2025, https://cloud.google.com/blog/products/ai-machine-learning/announcing-agents-to-payments-ap2-protocol , and donated to the FIDO Alliance on 28 April 2026, https://blog.google/products-and-platforms/platforms/google-pay/agent-payments-protocol-fido-alliance/ Repository: https://github.com/google-agentic-commerce/AP2 All three: Verified 31 August 2026.
12. Mastercard, "Mastercard unveils Agent Pay, pioneering agentic payments technology to power commerce in the age of AI", 29 April 2025, https://www.mastercard.com/global/en/news-and-trends/press/2025/april/mastercard-unveils-agent-pay-pioneering-agentic-payments-technology-to-power-commerce-in-the-age-of-ai.html and "Mastercard launches Agent Pay for Machines to unlock super-fast, always-on payments", 10 June 2026, https://www.mastercard.com/global/en/news-and-trends/press/2026/june/mastercard-launches-agent-pay-for-machines.html ; Visa introduced Visa Intelligent Commerce in "Find and buy with AI: Visa unveils new era of commerce",

https://corporate.visa.com/en/sites/visa-perspectives/newsroom/product-drop-intelligent-commerce.html which carries the spending-limit and tokenised-credential wording. That page states no publication date. A companion Visa release recorded here on 24 August 2026 has since become unreachable, its own URL returning 404 and the press-release listing no longer carrying it, so I cite only the page that survives. Accessed 24 and 31 August 2026, re-checked 2 September 2026.

13. FIDO Alliance, "Standards for Trusted Agentic Interactions": "Trust frameworks for agent-driven interactions do not yet exist". The same page records the work as sitting in two technical working groups, the Agentic Authentication Technical Working Group and the Payments Technical Working Group, the second of which "is developing specifications for agent-initiated commerce including secure delegation, verifiable authorization and trusted transaction execution", specifications that "will draw from initial contributions from Google (AP2) and Mastercard (Verifiable Intent)". https://fidoalliance.org/fido-alliance-agentic-ai/ The Alliance's published user-authentication specifications remain three sets: "FIDO Universal Second Factor (FIDO U2F), FIDO Universal Authentication Framework (FIDO UAF) and the Client to Authenticator Protocols (CTAP)", and the same page lists no agentic or payments specification. https://fidoalliance.org/specifications/ Both accessed 31 August 2026.

14. IETF, "The Idempotency-Key HTTP Header Field", `draft-ietf-httpapi-idempotency-key-header-07`, posted 15 October 2025 and expired 18 April 2026 without becoming an RFC. The datatracker gives the document type as "Expired Internet-Draft (httpapi WG)", the state as "Expired & archived", versions 00 to 07 with no later revision, and no intended RFC status. https://datatracker.ietf.org/doc/draft-ietf-httpapi-idempotency-key-header/ Accessed 31 August 2026.

15. ISO 20022 Business Application Header Message Usage Guide, version 3.0, 3 November 2021, section 2.10: where the receiver did hold the original message identified in Related, the resent message "MUST BE IGNORED". Distributed in the Business Application Header message set alongside `head.001.001.04`, which is the current identifier. Verified 17 August 2026.

16. Payments Canada, Real-Time Rail ISO 20022 messaging companion, version 1.4, 30 May 2025, section 3.2.2, "Optional Elements". https://www.payments.ca/systems-services/payment-systems/real-time-rail-payment-system Verified 17 August 2026.

17. European Payments Council, Guidance on Reason Codes for SCT Inst R-transactions, EPC059-18 version 7.0, 5 October 2025, row AM05: Duplication, R-transaction type "Reject" and nothing further. That document applies to the SCT Inst rulebook alone; the SCT and SDD dispositions are at reference 18. https://www.europeanpaymentscouncil.eu/sites/default/files/kb/file/2025-10/EPC059-18%20v7.0%20Guidance%20on%20Reason%20Codes%20for%20SCT%20Inst%20R-transactions.pdf Verified 17 August 2026.

18. European Payments Council, Guidance on Reason Codes for SCT R-transactions, EPC135-18 version 6.0, 28 November 2024, row AM05 (Reject, Return), https://www.europeanpaymentscouncil.eu/document-library/guidance-documents/guidance-reason-codes-sepa-credit-transfer-r-transactions ; and Guidance on Reason Codes for SDD R-transactions, EPC173-14 version 8.0, 28 November 2024, row AM05 (Reject, Return, Reversal), https://www.europeanpaymentscouncil.eu/document-library/guidance-documents/guidance-reason-codes-sepa-direct-debit-r-transactions Accessed 20 August 2026.

19. Bank of England, CHAPS ISO 20022 migration, draft like-for-like schemas issued for industry review, August 2019, Annex 1, Mapping Table 4 (Return Codes), page 25: field 72 code AM05 maps to the GPI return code DUPL and to the ISO reason code AM05, "Duplication". Cited as the mapping the migration was designed around; the current CHAPS schemas supersede it. https://www.bankofengland.co.uk/-/media/boe/files/payments/rtgs-renewal-programme/iso-20022/chaps-iso-20022-migration-draft-like-for-like-schemas-review.pdf Accessed 20 August 2026 and 1 September 2026. A case-insensitive search of all 60 pages returns "AM05" twice, both in that one table row, and zero occurrences of "duplicate", "suppress" and "after settlement", so the draft maps the code without stating how a duplicate is handled.

20. Pay.UK, Faster Payments System Principles, information guide, January 2026 version 11. The Faster Payments System operates on ISO 8583: the guide specifies it in ISO 8583 tags throughout, at tags 35, 42, 62, 120 and 121, and "20022" occurs zero times across its 35 pages. Pay.UK separately publishes an "ISO 8583 standard mapping to and from ISO 20022 messages" as a registration-gated reference library, whose landing page describes the central infrastructure as one "which uses ISO 20022 messages" even though the guide itself is specified wholly in ISO 8583, https://www.wearepay.uk/what-we-do/payment-systems/faster-payment-system/faster-payment-system-iso20022-standards-library/ The provision of the 31-

character end-to-end reference within tag 62 is described as competitive. So is the re-try process for single immediate payments, where "How this is executed is competitive"; for standing order and forward dated payments the sending participant "will re-try ... later in the day (as required by the Financial Conduct Authority (FCA))". Every re-try the guide describes is a response to a lack of funds. https://www.wearepay.uk/wp-content/uploads/2026/02/Pay.UK-Faster-Payments-System-Principles-v-11-Jan-2026.pdf Accessed 20 and 25 August 2026. On the thirty-one characters: Open Banking Limited's Read/Write API data model annotates its `EndToEndIdentification` entry "OB: The Faster Payments Scheme can only access 31 characters for the EndToEndIdentification field", on Domestic Payment Consents, https://openbankinguk.github.io/read-write-api-site3/v4.0.1/resources-and-data-models/pisp/domestic-payment-consents.html , and on Domestic VRPs, https://openbankinguk.github.io/read-write-api-site3/v4.0.1/resources-and-data-models/vrp/domestic-vrps.html , both read at version 4.0.1 on 1 September 2026.

21. Payment Systems Regulator, "APP scams reimbursement requirement: consolidated policy statement", PS25/5, May 2025. The requirement came into effect on 7 October 2024, covering "any reimbursable Faster Payments or CHAPS APP scam payments that took place on or after" that date (paragraph 3.7). It applies to "Faster Payments and CHAPS payments sent and received by PSPs in the UK across the Faster Payments system, including payment initiation service (PIS) transactions" (paragraph 3.2), quoted whole because the PIS clause sits inside that sentence's Faster Payments limb. On the maximum: "We have set the maximum level for Faster Payments at £85,000. The Bank has set the maximum level for CHAPS at £85,000" (chapter 3, Table 2, policy 7). Once the sending provider has reimbursed its consumer it "can require 50% of the amount from receiving PSP(s)", and "cannot request this contribution unless and until they have reimbursed the consumer" (paragraph 3.21). "The Bank of England, as the Payment System Operator for CHAPS, maintains the CHAPS reimbursement rules" (paragraph 1.4). Scope turns on an authorisation the consumer "has been deceived into granting" (paragraph 2.5). PS25/5 is guidance: its paragraphs 1.3 and 1.4 record that the definitive requirements sit in the regulator's legal instruments and, for CHAPS, in the CHAPS reimbursement rules the Bank maintains, both of which prevail over it. I checked the regulator's published policy statements, consultations and APP-scams publications on 31 August 2026 for a later change to either maximum and found none. I did not open the legal instruments or the CHAPS reimbursement rules themselves, and I record that as a limit on what I checked. https://www.psr.org.uk/media/rhelv4op/ps25-5-app-scams-reimbursement-consolidated-policy-statement-may-2025.pdf Accessed 31 August 2026.

22. Open Banking Limited, Domestic VRP Consents, Read/Write API version 4.0. `MaximumIndividualAmount` is mandatory within the control-parameters block, and `PeriodType` enumerates six values: Day, Week, Fortnight, Month, Half-year and Year. The provision is unchanged in version 4.0.1. https://openbankinguk.github.io/read-write-api-site3/v4.0/resources-and-data-models/vrp/domestic-vrp-consents.html Accessed 20 August 2026.

23. Open Banking Limited, Domestic VRPs, Read/Write API version 4.0: "The ASPSP must ensure that the payment instruction adheres to the limitations set by the corresponding VRP consent's ControlParameters", and "When a payment would breach a limitation set by one or more ControlParameters, the ASPSP must return an error with code `U014` and pass in the control parameter field that caused the error in the `Field` field of the error message." A case-insensitive full-text search of that page returns zero occurrences of "reserv", "concurren", "atomic", "simultaneous" and "race", so the page states the duty without stating how it is enforced across concurrent instructions. The provisions are unchanged in version 4.0.1. https://openbankinguk.github.io/read-write-api-site3/v4.0/resources-and-data-models/vrp/domestic-vrps.html Accessed 20 August 2026. The same five stems return zero occurrences across the Domestic VRP Consents page, the Read/Write API Profile, the Variable Recurring Payments proposition, the delegated-SCA Customer Experience Guidelines page and the Domestic Payment Consents page, and two unrelated occurrences: "PATCH operations are atomic" on the consents page, about amending a consent, and "V05 and V06 has been reserved for 'attended' payments" in the knowledge base. Open Banking Limited's knowledge base, at reference 24, was asked whether payments with a rejected status are excluded when the periodic limit amount is calculated, and answered that ASPSPs and payment initiation service providers "should exclude those with RJCT or BLCK status". It does not say how two instructions arriving together are ordered against one limit. Accessed 1 September 2026.

24. Open Banking Limited, Variable Recurring Payments knowledge base: "ValidToDateTime can be left blank which means the validity of the consent is indefinite." https://openbankinguk.github.io/knowledge-base-pub/standards/general/vrp.html Accessed 20 August 2026.

25. The technical standards on strong customer authentication and common and secure methods of communication, being the assimilated Commission Delegated Regulation (EU) 2018/389 as amended, which the Financial Conduct Authority maintains in the technical standards section of its Handbook. Articles 2 and 13, reached in the Handbook at Technical Standards, Payment Services, 2021, Strong Customer Authentication and Common and Secure Methods of Communication, Chapter 1 Article 2 and Chapter 3 Article 13. https://handbook.fca.org.uk/technical-standards The assimilated text is also published at https://www.legislation.gov.uk/eur/2018/389/contents Both accessed 31 August 2026.

26. Open Banking Limited, Customer Experience Guidelines version 4.0.1, 18 March 2026, "VRP Payments with delegated SCA": payments "that do not rely on the application of an SCA exemption by the ASPSP, but rather the application of delegated SCA to each individual VRP Payment". The same page continues, "This provides explicit consent for each payment instruction, dynamically linking the amount and a payee", and says of that journey that "the customer is required to be in session for each VRP payment(s)". https://standards.openbanking.org.uk/customer-experience-guidelines/payment-initiation-services/vrp-payments-with-delegated-sca/latest/ Who performs it, and the basis, are set out in Open Banking Limited, Variable Recurring Payments proposition P26 version 1.2, 29 September 2021: at section 6.1 the PISP will "Carry out SCA of the PSU either themselves or via another party to whom the ASPSP has delegated the responsibility for SCA", which that section labels "customer in session" against "unattended payments" for the exemption route; section 6.5 describes bilateral contractual access, under "A bilateral agreement between each PISP and ASPSP"; and section 8 states that the VRP standards requirements are "optional" for implementation by ASPSPs and/or TPPs for non-sweeping use cases. Section 4.2.1 states that VRP Payments under an SCA exemption "must rely on the application of an available exemption", and that "the ASPSP will likely rely on the Article 13 exemption set out in the SCA-RTS (by setting up the payee as a trusted beneficiary) or another available exemption (e.g., Transaction Risk Analysis or low value)", and continues that the customer "can be treated as having given explicit consent for each VRP Payment under a VRP Consent" where the payee is fixed and the frequency and value are bounded; section 4.2 says explicit consent for payment instructions "can be achieved in two ways", that route and the delegated route at section 4.2.2. Section 9 requires for sweeping "the application of the Trusted Beneficiary SCA exemption by the ASPSP for each Sweeping Payment. OR the 'Transfer to Self' SCA exemption by the ASPSP for each sweeping payment within the same ASPSP." Section 9 adds, of the trusted-beneficiary limb, that the ASPSP "may choose not to apply the exemption in exceptional circumstances with an objective approach in line with the proportionality requirements of the PSRs". In the technical standards at reference 25 those are Article 18, transaction risk analysis, Article 16, low-value transactions, and Article 15, credit transfers between accounts held by the same natural or legal person; read in the assimilated text at https://www.legislation.gov.uk/eur/2018/389/contents on 31 August 2026. https://standards.openbanking.org.uk/get-started/propositions/p26/ The two Open Banking Limited pages were accessed 24 August 2026 and re-read on 1 September 2026.

27. Competition and Markets Authority, sweeping definition agreed 26 July 2021 and clarified by letter of 14 March 2022, under the Retail Banking Market Investigation Order 2017. The 26 July 2021 date is stated in the CMA's Sweeping Clarification, Summary of Responses. https://assets.publishing.service.gov.uk/media/622ef8f8d3bf7f5a89aecea5/Sweeping_Clarification_-_Summary_of_Responses_.pdf and https://assets.publishing.service.gov.uk/media/622ef71fd3bf7f5a86be8fa4/Sweeping_clarification_letter_to_be_sent_14_March_2022__.pdf Accessed 20 and 25 August 2026.

28. Competition and Markets Authority, letter to Open Banking Limited, 3 December 2025: sweeping to repay credit may be mandated only where the destination has "similar characteristics to an overdraft", that is a revolving credit facility with no fixed repayment schedule. https://assets.publishing.service.gov.uk/media/692ebf09a245b0985f0343bf/CMA_response_to_OBL.pdf Accessed 20 August 2026.

29. UK Payments Initiative Ltd, "UK banks and fintechs join forces to launch new payment scheme", 2 June 2026. https://www.ukpaymentsinitiative.co.uk/uk_banks_and_fintechs_join_forces_to_launch_new_payments_scheme/ Accessed 20 August 2026.

30. Financial Conduct Authority and Payment Systems Regulator, prioritisation statement on commercial variable recurring payments (the UKPI scheme, Phase 1/Wave 1), footnote 3: "I.e. regulated financial services, regulated utilities, local and central government." https://www.fca.org.uk/publication/corporate/fca-psr-prioritisation-commerical-variable-recurring-payments-ukpi.pdf Accessed 20 August 2026.

31. Financial Conduct Authority, statement on the launch of the UK Payments Initiative scheme, 2 June 2026: "subject to legislation expected to give us new powers", the FCA "will consult on a long-term regulatory framework by the end of 2026". https://www.fca.org.uk/news/statements/open-banking-launch-uk-payments-initiative-scheme Accessed 20 August 2026.

32. ISO 20022 message schemas, opened 17 August 2026: `MndtRltdInf` appears in pain.008.001.12 and pacs.003.001.12 as MandateRelatedInformation16, and in pain.001.001.13, pacs.008.001.14 and pain.013.001.12 as CreditTransferMandateData1; `MndtId` is an optional Max35Text in both, and in CreditTransferMandateData1 its siblings are `Tp`, `DtOfSgntr`, `DtOfVrfctn`, `ElctrncSgntr`, `FrstPmtDt`, `FnlPmtDt`, `Frqcy` and `Rsn`. The published schemas carry no element documentation, so they establish the component and not the registered definition. `InitnSrc` (PaymentInitiationSource1) appears in the group header of pain.001.001.13 only and in no pacs message. `InstrId` is an optional `Max35Text` in PaymentIdentification13, the payment-identification type of pacs.008.001.14, and the same declaration holds in pacs.008.001.08 through .14 and in pacs.003, pain.001, pain.008 and pain.013; `Max35Text` restricts a string to a minimum length of 1 and a maximum of 35. Schemas read at https://www.iso20022.org/sites/default/files/documents/messages/pacs/schemas/pacs.008.001.14.xsd on 31 August 2026; the catalogue page, https://www.iso20022.org/iso-20022-message-definitions , was not reachable from the command line that day and was read in a browser.

33. ISO 20022 External Code Sets, publication of 28 August 2026, quarter 2Q2026, distributed as `2Q2026_externalcodesets_v2.xlsx`. Downloaded from the registration authority's permalink https://www.iso20022.org/sites/default/files/media/file/ExternalCodeSets_XLSX.zip and read 31 August 2026. The Intro&History worksheet gives that publication date and records that the sets are maintained quarterly. The AllCodeSets worksheet carries 3,305 codes across 136 sets, of which ExternalPurpose1Code holds 366, ExternalLocalInstrument1Code 115 and ExternalCategoryPurpose1Code 49. A case-insensitive search of every code value, code name and code definition in those three sets returns zero occurrences of "agentic", "autonom", "delegat" and "robot"; the three hits for "agent" are the securities-lending sense (BKFE, LREV, LSFL) and the two for "machine" are an unattended vending machine (IPU2, IPUW). The release is identified from inside the file itself rather than from a catalogue listing.

34. ISO 20022, guidance on SupplementaryData and the Catalogue of registered ISO 20022 message extensions, version 7, May 2024: eighteen registered extensions, thirteen of them DTCC securities corporate actions, counted from the Current Catalogue worksheet of `ISO20022_SupplementaryData_May2024_v1.xlsx`. A SupplementaryData extension requires SEG approval, registration in the ISO 20022 Data Dictionary, re-qualification against each new message version, and prior agreement by all parties involved. https://www.iso20022.org/catalogue-messages/additional-content-messages/supplementary-data Accessed 24 August 2026.

35. Financial Conduct Authority, Guidance for firms on the fair treatment of vulnerable customers, Finalised Guidance FG21/1, February 2021. https://www.fca.org.uk/publications/finalised-guidance/guidance-firms-fair-treatment-vulnerable-customers Accessed 20 August 2026.