

AIIM Contribution

Artificial Intelligence Identity Management Community Group

Tom Jones editor 2025-10-30

Context

The purpose of this contribution is to create a defined set of terms and use cases where Artificial Intelligence workstreams can be defined as to their goals and limitations. One goal is to make it clear that the number of combinations of humans and agents is very large and any solution for a Digital Public Infrastructure needs to be inclusive of all of the participants.

Taxonomy

People are humans as recognized by the Universal Declaration of Human Rights that expects to have agency of their personally identifiable information.

Entities are securely named objects

Users are legal entities, like: people, corporations and governments.

Enterprises are legal entities that are not people.

User Agents today are typically the Internet browsers or web sites.

Digital Agent = a persistent identifiable workload with goals, environmental awareness and independent ability to act on prompts or changes in the environment.

Agent = a workload that is able to sign statements on its own behalf. With an SDA it can also sign on the user's behalf. (need to pick one)

Holders = subjects that control devices, like smartphones that contain verified claims.

Subject = any entity that comes with a collection of identifiers, attributes, behaviors and grants.

Claim = any assertion by or about a Subject. If there is some corroboration of the claim, it is called a Validated or Verified Claim.

Subject Delegation of Authority (SDA) is a certificate issued to the Holder or other Agent on the device.

Policy = formal statement that defines the rules, behaviors, or constraints governing how agents operate within a given context. Policies are used to enforce security, manage resources, delegate authority, and ensure predictable behavior across digital environments.

Delegated Agents have at least one SDA to represent a user in ways defined by a policy.

Autonomous Agents do not require an SDA to function. They do require a secure ID. Delegation = A transfer of privileges controlled by one entity to another with policy constraints about where, when and why they can be exercised.

Workload = 1 “A running instance of software executing for a specific purpose.” (WIMSE) that is an application identified in a secure manner.

Workload = 2 An independent computational unit that can autonomously receive and process invocations, and can generate invocations of other workloads.

Holder Client Component = Agent workload operating on a device under holder management

MCP = Model Context Protocol, which is an open protocol that standardizes how applications provide context to large language models (LLMs)”

Session = a concept binding a user, or secure user device, to an AI agent context or workload instance (this allows a session to continue if the user is not active.)

Context = persistent information about an agent’s interaction with users at least for the duration of a session.

Relationship = a persistent shared identification between an agent and a human or another agent. There is no a priori requirement that any identity information is shared to create a relationship, only the ability to be sure that messages are secured by that relationship.

Prompt = a query or command addressed to an Agent which might be Autonomous.

PII = a subject’s personally identifiable information. Trust Domain = The authority that identifies the domain within which the identifier is scoped. (WIMSE)

Protocol = a set of formal rules or procedures (e.g., diplomatic protocol).

Registry = number: the official filing number of a document in an office.

Technical protocol in computing = a system of rules for communication (e.g., HTTP, TCP/IP).

Data Controller / Orchestrator / Transparency / Policy

AI Agents are like politicians, they have a task they are given to complete and often decide that they need more control (more privilege) than they have. Then they ask for more control and once they have it are not likely to let that control elapse as they might need it again to complete some unknown future task.

- Who is policy owner and who is enforcer/executor/auditor
- What is the duration of any delegation of authority
- Policy can happen on both sides. An individual user is able to confirm that their policies are enforced - not just the service provider. Policy can happen on both sides. A potential solution is some sort of Policy Audit Point (PAP) that reports on the actual application of policy.
- What is the default policy that a user can impact? Acts on the user's behalf to determine which agent gets to see user data/context.
- Can users control policy? (like settings on a smartphone)
- Can an AI be sued for liable? [Google AI Falsely Says YouTuber Visited Israel, Forcing Him to Deal With Backlash](#)
-

Digital Persona

A digital persona is a context-specific representation of you online—your attributes, behavior, and preferences as they appear in different platforms or roles.

You might have a work persona (professional tone, task-focused), a home persona (casual, expressive), and even a civic persona (engaged in public discourse or governance).

Alternate terms used in other contexts: Contextual Identity, Role Avatar, Digital Twin, Agentic Facet. This should not be confused with the term Persona as used in User Experience Research.

User cases

The digital world is becoming so complex that it will soon become impossible for anyone other than technical experts to even navigate without an AI agent. And we are not talking about one agent that acts as a digital twin but dozens engaged every day of our lives. Humans seem to want a lot of choice, but be careful what you wish for. The trade off may be loss of the ability to navigate without help, every step of the way. In such a

rapidly evolving digital environment, we need to be careful not to leave out those people who do not have access to the same digital tools that allow full use of the digital public infrastructures that exist across the globe. Use cases are needed to investigate the large swaths of the digital landscape that the user may need to navigate: at home, at work, in the car or learning environments that used to be called school.

1 User at workstation conversation with an Intelligent Agent

- Looking at user choice in learning, this case focuses on learning at home.
- Current computer/user interface augment with AI prompts. (today's savvy user case)
- It is possible that this use case could be combined with #3, developer use.
- At any sufficiently powerful and connected computing/display device.
- Where privacy is not the primary concern.
- One-off tasks to be complete (task oriented)

In this use case the user is in full control of a highly functional device that can run a local version of an AI agent that has full access to the user's local context, for example open web pages and similar resources. The device will maintain local context that can be deleted by the user on demand or via some user-selected time-out period. The goal is to help the user with either a short-term problem or long term project with rich prompts from the user which grow as the project unfolds.

2 User at home with a TV + cable/network box Use case

- Privacy is a primary concern.
- The user doesn't really expect AI interaction/conversation.
- Common user problems - it is exceedingly difficult to find the right place to even ask a question as the current AI agents generally do not talk to each other. When there is more than one path to a given solution, it does not seem that there is any possible way for the user to find the least expensive solution.

The TV device comes with some o/s, like google, apple or the device manufacturer.

Each app is separately loaded and comes with its own separate list of subjects, each with their own list of authorized subscriptions. The following case describes content Subscriptions on Google TV + Xfinity Cable box with voice remote.

- 1) Xfinity remote control with microphone allows voice input.
 - a) Any cable channel that is part of subscription plan
 - b) One advantage is that it is not listening except when the button is pressed.
 - c) On Demand events like movies including user saved events
 - d) Access to streaming apps like Netflix, Hulu, Prime Video

- e) Voice remote control for content search and playback
- f) Pay-per-view events like sports and much more
- 2) Google TV Subscriptions
 - a) Google Play Subscriptions:
 - i) YouTube Premium
 - ii) Google TV rentals and purchases
 - iii) App-based subscriptions (Disney+, Max, Paramount+, etc.)
 - b) Third-Party Streaming Apps:
 - i) Installed directly on Google TV or cast from mobile
 - ii) Includes Netflix, Peacock, Apple TV+, Sling, etc.
- 3) Identifiers and Access Mechanisms for Authentication & Identity
 - a) There will likely be more than one user on this device and that user could many a dozen devices, several of which are in the same room as the TV.
 - b) Some TVs and the old XBOX Kinect determine the ID of the person in the field of view of the camera and automatically log them into the program guide they last used.
 - c) Xfinity ID: Used to log into Xfinity Stream and manage subscriptions
 - d) Google Account: Required for Google TV personalization, app downloads, and purchases
 - e) Device Identifiers:
 - i) Chromecast device ID
 - ii) Google TV serial number
 - iii) MAC address for network access
 - f) Voice Profile Used to personalize voice commands
 - g) Note that with subscription proliferation this could be 40 different user IDs.

Other Smart Devices at home

Many devices that might be in a home or office that contain user IDs. List of devices in your home that likely have user ID accounts, either for personalization, access control, or cloud syncing:

1. Smart TVs (Samsung, LG, Sony, etc.)
2. Google TV / Android TV / Apple TV
3. Amazon Fire TV, Roku and other streaming devices
4. Network gating devices like the cable company or satellite access
5. Game consoles (Xbox, PlayStation) used for streaming
6. Smartphones (Android → Google account, iPhone → Apple ID)
7. Tablets (iPad, Android tablets)
8. Laptops and desktops (Windows → Microsoft account, macOS → Apple ID)

9. Smart speakers (Amazon Echo → Alexa account, Google Nest → Google account)
10. Home hub - which could start as a smart speaker and grow to information server
11. Home routers and mesh networks (Netgear, Eero, Google WiFi)
12. Smart thermostats (Nest, Ecobee)
13. Smart lights and plugs (Philips Hue, Kasa, TP-Link)
14. Security cameras and doorbells (Ring, Arlo, Wyze)
15. Bird or wireless cameras that can report on the type of animal present
16. Smart appliances (fridges, washers with app control)
17. Game consoles (Xbox Live, PlayStation Network, Nintendo)
18. VR headsets (Meta Quest → Meta account, HTC Vive → Steam account)
19. ChatGPT, Siri, Alexa, Google Assistant—all tied to user accounts
20. AI companion apps or productivity tools (e.g., OneDrive, Copilot on Windows)
21. Cloud accessible (SAAS like GitHub)
22. Access devices like door locks including garage door with remote control
23. External devices like cars, bikes and boats
24. Fitness trackers and smartwatches (Fitbit, Apple Watch, Garmin)
25. Health monitoring from Urinalysis to exercise bicycle with video displays.
26. Printers with cloud services (HP Smart, Epson Connect, CVS print shop)

Trust Domains within the Home

- 1) Family access to home and some mobile equipment
- 2) Individual personas for each member of the household, like work, home
- 3) Guests - think air b&b
- 4) These guest personas can come and go - works like a PC today
- 5) Family access to SAAS and streaming services like Amazon & Disney

3 - Asynchronous Coding Workflow Use Case

Developers want to prompt a background coding agent to complete an end-to-end development workflow (see Cursor Background Agents as an example). To complete this task, the coding AI agent needs to both write code AND access developer tooling.

For example, to create successful preview deployments of web apps, a coding workflow may need to set settings in external auth vendors (e.g., set redirect URLs) or control deployment configurations (e.g., Vercel, Cloudflare, or AWS configurations). It may wish to pull from or write to ticketing systems (e.g., Linear) or access external configuration information (e.g., Stripe Configurations). In some cases, CLI tools can provide access to these configuration options, but background agents may not have knowledge of the tools (not in pretraining data or context window), may become confused about what CLI

tools to use, or may not be pre-authorized to use the tool. MCP can allow for access to external configuration tooling, support OAuth at task creation time, and provide structured context on tool use in the agent's context window. The MCP servers are clearly remote protected resources.

Further, the background agent may need access to specific documentation. Including this in an MCP server allows the developer to explicitly place the documentation as an accessible tool into the context window, reducing the risk of an agent finding unrelated or out-of-date documentation via web search. This resource doesn't explicitly need protection, and it can be run remotely or locally via stdio as a read-only tool or resource.

Finally, the coding agent may want access to local tools that aid in the coding workflow. For example, environment permitting, a front-end coding task may want access to a Playwright server via stdio, which allows the agent to orchestrate browser usage dynamically to open localhost pages, screenshot the UI, and pass that visual context back to the coding agents. This runs locally and is explicitly not an OAuth resource server.

These are each different types of MCP servers, working together to control and structure the context window of the AI agent (context engineering) and complete the specific background software task as asynchronously and successfully as possible.

Control of Subject Private data by an Agent Use Case

Subject data is maintained within a Holder Client Component and not shared with MCP agents except as specifically and individually authorized by the data's subject.

Data Subject is not the Holder of the Device Use Case

For cases like picking up medications from a pharmacist it is necessary to enter the creds of the subject of the prescription, which might just be their birthdate.

--

How to establish a Session

A W3C Device Bound Session Credential (DBSC) aims to reduce account hijacking caused by cookie theft. It does so by introducing a protocol and browser infrastructure to maintain and prove possession of a cryptographic key.

[Device Bound Session Credentials explainer](#)

How to pass context from one AI to another

As the user's AI context moves from a local AI agent to more specialized agents in the web there needs to be some collection of data that moves but does not unnecessarily

move fully identifiable information about the user. As one possibility, consider a JSON object which could be called a Connectoid that builds a (typically) short term connection that only consumes high cost resources as absolutely necessary.

The first step is creation of some sort of identifier that until the task is completed at which time the entire context is destroyed everywhere.

JSON from the human

Identifier

Policy (including fiduciary)

Context

Proofing

MIT NANDA project

Governance

The question is whether AI governance is applied at design time or at execution time, or perhaps both. Most of the discussion about governance is how to design fail safe systems. It is perhaps time to turn the focus back to execution time where the user intention, the various applicable policies and legal requirements are met before any data is released.

The policy enforcement point (PEP) is the place in the finished product where the decision is made to release the output, or not. This is the Governance Model of a recent series from Apple.

Appendices

Can an AI be ethical?

- Is that like asking whether a machine can be ethical?
- Clearly a machine can create a legal obligation.
- Many reports of rogue AI relate to decisions made that turn out disastrous
- “If we use, to achieve our purposes, a mechanical agency with whose operation we cannot efficiently interfere once we have started it... then we had better be quite sure that the purpose put into the machine is the purpose which we really desire and not merely a colorful imitation of it.” from Brian Christian’s “The Alignment Problem”

What Is the Alignment Problem?

- At its core, the alignment problem asks:
- How do we ensure that AI systems pursue goals that reflect human values, intentions, and safety—especially as they become more autonomous?
- It's not just about avoiding catastrophic outcomes. It's about embedding agency, consent, and accountability into systems that increasingly act on our behalf.

The core of the problem is Alignment which is fundamentally ethical because it involves:

- Value selection: Whose values are encoded? Whose are excluded?
- Moral pluralism: Can one system reflect diverse, even conflicting, human norms?
- Consent and autonomy: Are users aware of what the system is doing? Can they opt out?
- Responsibility: Who is accountable when things go wrong?

Data Flow Diagrams and Threat Model

- This threat model has a data flow diagram showing how different AI agents can interact with a user at a browser window.

<https://github.com/TomCJones/threat-modeling/blob/main/models/AI%20in%20a%20Scripted%20User%20Agent.pdf>

- This paper shows a mental model of the use case where one user is prompting one AI during one interaction, or session. An interesting diagram on Page 8 https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/ANSSI-BSI-joint-releases/LLM-based_Systems_Zero_Trust.html?trk=feed-detail_comments-list_comment-textb

- Beautiful Venn Diagram of Human / AI responsibilities

<https://workshop.vennfactory.com/p/why-ai-cant-be-your-actual-therapist>

