

From the Connected Device to the Agent-based System: An Empirical Analysis of the Transformation of Design Work through a Case Study of a Next-generation Internet Box

Type: Case Study

Received: October 08, 2026

Published: October 30, 2026

Citation:

Robin Vivian. "From the Connected Device to the Agent-based System: An Empirical Analysis of the Transformation of Design Work through a Case Study of a Next-generation Internet Box". PriMera Scientific Engineering 9.5 (2026): 14-29.

Copyright:

© 2026 Robin Vivian. This is an open-access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Robin Vivian*

Perseus Laboratory, University of Lorraine, Metz, France

***Corresponding Author:** Robin Vivian, Perseus Laboratory, University of Lorraine, Metz, France.

Abstract

The historical model of user-centred design (UCD) is based on the creation of deterministic graphical and physical interfaces optimised for human perception. The integration of autonomous agents based on *Large Language Models* (LLMs) and equipped with action capabilities (*tool use*) is revolutionising this paradigm: the interface no longer addresses humans directly, but becomes an intermediate semantic layer orchestrated by and for the machine. This article analyses this epistemological shift through a comprehensive industrial case study: the design of a next-generation internet box combining a hardware unit, an embedded screen, a companion mobile app and home network/home automation services. We demonstrate how the designer's workflow is transforming, shifting from the design of static screens to the engineering of ontologies, the definition of semantic action capabilities (*tools*) and the configuration of *agency* sliders. Finally, we propose a framework of mixed evaluation metrics (AIX) that enables the computational performance of agents to be assessed whilst preserving the intelligibility and decision-making autonomy of human users.

Keywords: Human-Computer Interaction (HCI); Agent-Centred Design (ACD); Artificial Intelligence Experience (AIX); IoT & Smart Home; Semantic Affordance; Ephemeral Interfaces; Autonomy Governance; Governance of Autonomy

Introduction

Context: The Hegemony of User-Centred Design (UCD)

For over three decades, the design of software systems and connected consumer products has been governed by an unchallenged theoretical and methodological paradigm: User-Centred Design (UCD) or Human-Centred Design (Norman, 1988; Shneiderman, 2022). Emerging in the 1980s in response to an IT sector driven solely by technical feasibility, this conceptual framework established the integration of principles from cognitive psychology and ergonomics at the heart of development cycles (ISO 9241-210). The fundamental axiom of the UCD model stipulates that the technical system must be optimised exclusively to match the capabilities, limitations and mental models of the human operator. In this traditional framework, the computing artefact — be it a mobile application, an operating system or the interface of a piece of hardware — is viewed as a purely reactive and deterministic tool.

The graphical user interface (GUI) acts as a rigid visual translator between human intention and machine execution. The assessment of a product's quality is based on standardised anthropocentric metrics, such as visual affordance, reduced cognitive load, effectiveness, efficiency and user satisfaction (Nielsen, 1993). The UX designer has thus established themselves as the architect of this mediation, modelling screen hierarchies, *user journeys* and stable graphical components designed to guide the human eye and gesture.

This dominance of UCD has taken the form of the institutionalisation of rigid design methodologies, centred on the three-part process of observation, prototyping and evaluation. Development teams have gradually structured their software architectures according to strict display constraints, whereby each system state had to correspond to a screen that had been previously designed and approved. In the field of telecoms equipment and consumer connected devices, this approach led to an increasing complexity of graphical interface : for every new network or home automation feature, an additional sub-menu or form was added to mobile administration applications. The user thus found themselves at the centre of a direct command chain, forced to learn the syntax and navigation structure imposed by the product designer in order to achieve their goals.

The agentic paradigm shift and the crisis of the deterministic interface

This anthropocentric paradigm is now facing a major epistemological shift brought about by the emergence of generative artificial intelligence and the deployment of autonomous agent architectures based on *large language models* (LLMs). Equipped with the ability to perceive, plan and execute actions via *tool use and function calling*, these new systems no longer merely respond linearly to an explicit command. They possess computational agency that enables them to break down a macroscopic objective, interact with third-party APIs and generate content or code in a stochastic manner. This stochasticity — that is, the probabilistic and non-deterministic nature of the agent's behaviour — undermines the model of the static graphical user interface (Yang, Steinfeld, Rosé, & Zimmerman, 2020; Dove, Halskov, Forlizzi, & Zimmerman, 2017). Faced with an identical human instruction, an autonomous agent may follow distinct algorithmic trajectories from one session to the next. Consequently, traditional methods of scripting and prototyping prove structurally incapable of modelling systems where the interface is no longer a structure pre-conceived by a human designer, but a dynamic semantic flow generated in real time by and for machines.

The emergence of these agent-based architectures also challenges the foundations of Norman's (1988) theory of action. In the classical model, the user bridges the 'execution gap' by translating their intention into physical actions on the interface, and then bridges the 'evaluation gap' by visually interpreting the system's change of state. With an autonomous agent, these two gaps are largely delegated to the algorithmic entity: it is the agent that interprets the overall intention formulated in natural language, plans the sequence of actions to be carried out, and verifies that the result has been achieved with the technical subsystems. The result is a gradual phasing out of the traditional user interface in favour of semantic exchanges invisible to the human eye, reducing direct manipulation to a simple validation or guidance of intent.

Research problem and hypothesis

This gradual shift away from placing humans at the centre of direct execution loops reveals a fundamental contradiction within the discipline of interaction design. If human users no longer directly manipulate a system's buttons, menus and settings, but delegate these tasks to a synthetic intermediary, maintaining human-centred design as the sole methodological compass becomes a technical anachronism (Vosgiens, Olry & Vivian, 2026). This observation leads us to formulate the following research question: *How does the transition from user-centred design (UCD) to agent-centred design (ACD) and the artificial intelligence experience (AIX) concretely transform the value chain, design artefacts, evaluation metrics and the role of the designer in the development of a complex hardware and software system?* Our research hypothesis posits that the designer's role does not disappear, but undergoes an ontological transformation: it is shifting from the role of designer of static graphical interfaces for the human eye to that of semantic architect, devising data models intelligible to machines (*schemas*), capabilities for action (*tools*) and ethical safeguards for regulating autonomy (*agency sliders*).

This transformation requires us to reconsider the very notion of affordance. Classical visual affordance, defined by Gibson (1979) and adapted to software design by Norman (1988), was based on visual cues prompting action (a domed button suggesting a press, a

scroll bar suggesting movement). Within the framework of agent-centred design, affordance becomes *semantic*: it relates to a language model's ability to understand the scope of a function's action based on its textual description. The designer no longer shapes the user's visual perception, but rather the agent's linguistic perception.

Subject of study and structure of the article

In order to ground this theoretical discussion in the industrial reality of product design, this article draws on an empirical and comparative case study: the design of a next-generation internet box ('N-Gen'). This consumer device is an ideal subject of study as it combines three complementary dimensions of interaction: the physical artefact (the box itself and its on-board display), the companion mobile app, used for control and diagnostics, and the ecosystem of on-board services (Wi-Fi network management, security, home automation). The remainder of this article is structured as follows: Section 2 provides a critical review of the limitations of the traditional UX model in the face of agent-based AI and sets out the conceptual framework of AIX and Agent-Centred Design (ACD). Section 3 details the methodology and compares the design workflows for the internet box using the classical UX approach versus the agent-based approach. Section 4 analyses the results across the four key dimensions of the transformation of the designer's profession (functional, hardware, software and governance). Finally, Section 5 discusses the risks of professional alienation (semantic Taylorisation), formulates new mixed evaluation metrics and concludes with the future prospects for symbiotic coexistence between biological and synthetic intelligences.

State of the art and theoretical framework

The limitations of traditional UX in the face of stochastic systems

The inability of traditional HCI methodological frameworks to model systems incorporating machine learning components has been the subject of pioneering work in computer science and design (Yang et al., 2020; Dove et al., 2017). Yang et al. (2020) demonstrated that the inherently evolving, adaptive and non-deterministic nature of probabilistic models renders the majority of conventional usability heuristics invalid. The *behavioural drift* inherent in learning algorithms prevents designers from anticipating, scripting or prototyping all possible states of a system when it goes live. To address this structural instability, several academic and industrial consortia have attempted to formalise empirical tuning guidelines. The work of Amershi et al. (2019), conducted under the auspices of Microsoft, has thus defined 18 guidelines for human-AI interaction, aimed at increasing predictability and mitigating error through principles such as clarifying the system's capabilities or contextually calibrating services.

Nevertheless, as Dove et al. (2017) rightly observe, these recommendations suffer from a major conceptual limitation: they treat artificial intelligence as a mere malleable design material, that is to say, a computational resource subordinate to a controlled, static graphical interface. These approaches fail to theorise the contemporary paradigm shift in which AI is breaking free from the graphical grid to act as an autonomous agent. The fundamental opacity of deep learning architectures (the 'black box' effect in the context of deep learning) combined with the absence of predictable interface states, undermines the effectiveness of traditional evaluation heuristics (Nielsen, 1993), leaving the human user at a loss when faced with machine behaviour that they can neither anticipate nor visually audit.

Furthermore, the prototyping methodology underlying traditional UX relies heavily on the ability to faithfully simulate the final experience using interactive mock-ups (*wireframes* or high-fidelity prototypes). However, when a system incorporates an autonomous agent capable of invoking tools in a non-linear manner, the proliferation of possible state combinations makes exhaustive prototyping physically impossible. Designers are faced with a combinatorial explosion: every natural-language interaction opens up a decision tree with potentially infinite ramifications. The traditional graphical prototype thus becomes a misleading construct that reveals only an arbitrary fraction of the trajectories the agent might follow under real-world conditions of use.

The redistribution of agency: from direct manipulation to semantic supervision (U-AIX)

A second major strand of the HCI literature focuses on the reconfiguration of roles within hybrid systems, an intermediate configuration that we term the human-AI experience (U-AIX). At the heart of this field of research lies an intense scientific debate concerning

the preservation of the human operator's sovereignty. Shneiderman (2020, 2022) rigorously defends the *Human-Centred AI* (HCAI) model, arguing that future architectures must simultaneously maximise both the level of human control and the level of machine automation. According to this perspective, the total autonomy of agents must be rejected to prevent the loss of legal accountability, the decline in cognitive skills and the disempowerment of human agency, whilst ensuring that the user remains in the validation loop (*human-in-the-loop*).

Conversely, researchers such as Heer (2019) and Subramonyam et al. (2022) are exploring approaches based on cognitive symbiosis and co-creativity. In these environments, the agent enjoys extensive freedom of planning and autonomy of execution, which structurally shifts human activity towards high-level orchestration tasks, the framing of intentions and semantic supervision. This shift alters the very nature of the ergonomic workload. As highlighted by Zamfirescu-Pereira, Wong, Hartmann and Yang (2023), the transition from direct manipulation (clicks, gestures, navigating tree structures) to control based on natural language intentions (*prompt engineering*) introduces unprecedented cognitive barriers for non-expert users. The difficulty of stabilising a mental model when dealing with a stochastic partner leads to methodological frustrations and breaches of trust. Managing this shared agency and defining real-time interruption protocols thus constitute one of the most complex scientific challenges in contemporary HCI.

This redefinition of task allocation raises the question of the granularity of control granted to the user. Whilst full automation frees up time and reduces immediate cognitive effort, it risks causing a phenomenon of attentional disengagement, placing the human in a situation of ineffective passive supervision. In the event of a sudden failure of the agent, the user, deprived of context, experiences considerable difficulty in re-entering the control loop and correcting the system's trajectory. Conversely, imposing systematic human validation for every algorithmic micro-decision undermines the very purpose of the autonomous agent by reintroducing an unbearable cognitive load. The U-AIX framework therefore requires the search for a dynamic equilibrium point where the degree of autonomy granted to the machine varies in real time according to the measured level of confidence, the risk of the action and the criticality of the domain of intervention.

The emergence of the ACD (Agent-Centred Design) paradigm and machine-to-machine interfaces

The modelling of digital environments in which software agents constitute the primary users or consumers of an ecosystem is rooted in the historical critique of industrial automation. As early as 1983, Lisanne Bainbridge formalised in her seminal work, **Ironies of Automation**, the paradox that the more a system is automated, the further the human operator is removed from direct control, becoming unable to intervene effectively in the event of a critical failure. This observation takes on new significance in the era of autonomous multi-agent networks, where the distance between humans and execution is no longer merely physical but conceptual. The work of Inie, Falk and Tanimoto (2023) shows that creative professionals' relationship with generative AI is forcing a redefinition of design practice, as traditional graphical interfaces give way to new forms of collaboration with generative systems.

It is at this intersection that the paradigm of Agent-Centred Design (ACD) and the Artificial Intelligence Experience (AIX) takes shape. ACD brings about a shift in perspective: it posits that the autonomous agent, as it becomes the primary actor within digital environments, must be positioned as the new subject of the design experience (Vosgiens, Olry & Vivian, 2026). The quality of an artefact is no longer measured by the regularity of a pixel grid or the elegance of a graphic style guide, but by the clarity of its machine documentation (OpenAPI schemas, JSON-LD ontologies), the density of information per computational unit, and its ability to eliminate communicational noise between algorithmic entities. Table 1 summarises this paradigm shift from human-centred design to agent-centred design.

This shift leads us to reconsider the nature of interaction errors. In the UCD model, an error is attributed either to a design flaw in the interface (poor visibility of a control, ambiguous labelling) or to a discrepancy in the user's cognitive representation. In the ACD model, interaction errors mainly occur in the form of *tool* hallucinations, malformed function calls or semantic misunderstandings relating to the preconditions for executing an API. The agent-centred designer must therefore develop diagnostic methods specifically geared towards correcting linguistic and structural ambiguities within the system specifications that they provide to language models.

<i>Design dimension</i>	<i>Human-Centred Design (HCI / UX)</i>	<i>Agent-Centred Design (AIX)</i>
Primary object of interaction	Graphical User Interface (GUI), screens, menus, buttons, fixed navigation trees.	Application Programming Interface (API), semantic structures, system prompts, tokens.
Key prototyping activities	Visual mock-ups in Figma/Axure, wire-framing, interactive user flows.	Ontological modelling (JSON/OpenAPI), definition of <i>tools/functions</i> , prompt engineering.
Validation methodology	Empirical usability testing, field observation, card sorting, heuristic evaluations.	Agent-based evaluations, simulations using synthetic personas, token efficiency audits.
Performance metrics	Task completion time, error rate, System Usability Scale (SUS), cognitive load.	Inference latency, token cost per transaction, semantic <i>drift</i> rate, determinism ratio.

Table 1: Paradigm shift from Human-Centred Design to Agent-Centred Design (adapted from Vosgiens, Olry, & Vivian, 2026).

Industrial case study: The design of the ‘N-Gen’ Internet box

Description of the hardware and software system

To demonstrate the operational viability of the ACD/AIX framework, we analyse the design of a new-generation consumer internet box, known as ‘N-Gen’. In the traditional industrial model, this device is designed as a fragmented system comprising a physical unit fitted with light-emitting diodes (LEDs) indicating the network status, a complex technical administration web interface (accessible via a local IP address), a companion mobile app for remote Wi-Fi management, and a graphical menu displayed on the television for accessing multimedia services. Although the economic aspect is often decisive, it is not a factor taken into account in our example.

In the ‘N-Gen’ version, re-architected around the agent-based paradigm, the system is unified under the control of an autonomous, embedded software orchestrator. This local agent interacts with the home’s physical and digital environment through three integrated layers of interaction:

1. **The physical hardware device:** A sleek enclosure incorporating a low-power dynamic display (high-resolution e-Ink/OLED screen) capable of displaying natural language text and simplified explanatory diagrams in place of traditional indicator lights;
2. **The companion mobile app:** A dynamic interface without a fixed menu tree, but featuring a contextual and adaptive ‘intent stream’ that generates temporary graphical views on the fly only when required;
3. **The ecosystem of services and automation:** An engine for managing the local network, security filtering and home automation, capable of calling software tools (*function calling*) and communicating with third-party devices.

This hardware and software architecture is based on a strict division of computational responsibilities. In order to guarantee the sovereignty of personal data and ensure a minimum level of operational continuity in the event of an internet outage, the embedded agent utilises a hybrid structure: a modest-sized SLM (*Small Language Model*) runs directly on-premises on the dedicated chip within the N-Gen device to process routine queries and monitor the line status, whilst complex planning operations requiring large volumes of data rely on a larger remote LLM. This technical hybridisation itself constitutes a major design constraint, requiring the designer to define the thresholds for switching between secure local processing and extended remote processing.

Comparative analysis of the step-by-step design workflow

The transformation in working methods is evident throughout the product’s production chain.

To grasp the operational scope of this transformation, it is necessary to analyse how the design stages have been profoundly restructured. Figure 1 illustrates this fundamental shift from a linear, deterministic production chain to a semantic and adaptive architecture. In the traditional model (UX/Human-Centred), the workflow progresses sequentially, from user research to the design of mock-ups and fixed menus, culminating in a static graphical interface. Conversely, the transformed workflow (Agent-Centred Design / AIX) shifts the focus of creative effort to the early stages of the project. The designer no longer designs screens as the end point, but instead models the ontology of the system, designs interface contracts (*schemas*) and defines the boundaries of autonomy (*agency sliders*). The

graphical user interface thus ceases to be a permanent structure: it becomes an ephemeral artefact dynamically generated on the fly by the agent according to the user’s contextual needs.

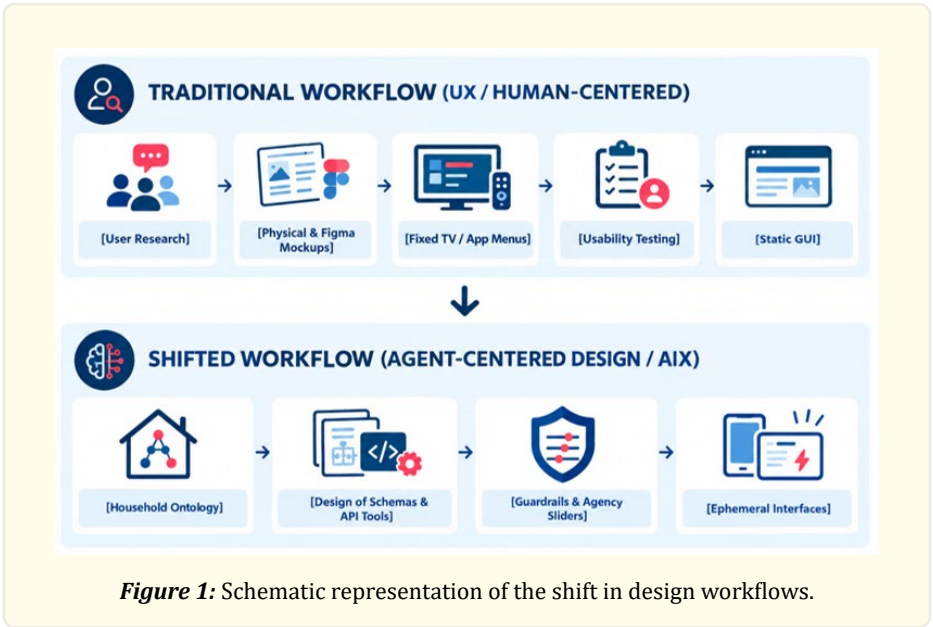


Figure 1: Schematic representation of the shift in design workflows.

Table 2 provides a detailed mapping of the operational differences between the traditional UX workflow and the adapted ACD/AIX workflow. It outlines this transition phase by phase, comparing traditional UX practices with the emerging ACD/AIX methodologies for the ‘N-Gen’ internet box. Each key stage of the project undergoes a complete reorientation of its deliverables and objectives. Whereas the scoping phase traditionally relied on demographic *personas* and rigid user journeys, it now focuses on the ontological modelling of household entities and behavioural patterns. The prototyping phase, historically dominated by the assembly of visual components in Figma, is evolving into an exercise in prompt engineering and the framing of agent-based behaviours. Finally, evaluation no longer seeks merely to measure the visual usability of a menu amongst a sample of human users, but extends to automated audits simulating complex interactions to test the system’s semantic robustness and computational efficiency.

Project phase	Traditional UX Workflow (Human-Centred)	ACD / AIX Muté Workflow (Agent-centred)
1. Scoping & Research	Conducting semi-structured interviews, home observations and creating demographic <i>personas</i> . Mapping user journeys in the form of <i>User Journey Maps</i> detailing every click.	Modelling the domain as a semantic ontology of the household (knowledge graph of devices, family members, bandwidth priorities and privacy rules).
2. Specification & Architecture	Defining the mobile app’s navigation structure and TV menus using flowcharts (<i>sitemaps</i>). Drafting functional specifications detailing the action of each button.	Formalisation of OpenAPI/JSON-LD <i>schemas</i> and drafting of semantic descriptions of the functions (<i>tools</i>) that the agent will be able to invoke to control the set-top box and the network.
3. Prototyping & Design	Design of static and then interactive visual mock-ups in Figma. Creation of a rigid graphical <i>design system</i> (buttons, forms, typography, icons).	Drafting of <i>system</i> prompts, programming of security <i>guardrails</i> and definition of rendering templates for ephemeral interfaces generated on the fly.

4. Hardware Integration	Selection of status LEDs and design of screen-printed markings on the plastic housing. Definition of LED flashing codes (e.g. red flashing = no fibre synchronisation).	Design of semantic explainability templates for the housing's dynamic display. Modelling of the agent's thought processes (<i>Chain-of-Thought UI</i>) in natural language.
5. Evaluation & Acceptance Testing	Conducting usability tests in an ergonomics laboratory with human users. Calculation of the SUS (System Usability Scale) score and counting of handling errors.	Automated behavioural audits through the injection of <i>synthetic personas</i> . evaluation of resistance to hallucinations, measurement of token efficiency and audits of response times.

Table 2: Detailed comparative analysis of the design workflow for an internet box.

The reorientation of activities during the scoping and research phase strikingly illustrates this transformation. Whereas the traditional UX designer set about constructing static user archetypes (*personas*) based on socio-demographic criteria that were often reductive, the agent-centred designer develops a dynamic knowledge graph representing the household ecosystem. This ontology not only catalogues physical entities (computers, telephones, smart light bulbs, televisions) and occupant profiles, but also specifies their functional relationships, their theoretical activity ranges and their relative priorities. This living semantic structure constitutes the model of the world to which the agent refers in order to interpret ambiguous requests made by household members.

Similarly, during the specification phase, the traditional sitemap is replaced by a tool *registry*. Each functionality of the network or home automation system is expressed in the form of a machine-readable interface contract, accompanied by explanatory annotations drafted with almost lexicographical rigour. The design work then consists of ensuring that two distinct functionalities do not present any cross-semantic ambiguities that might cause the agent to hesitate in decision-making or lead to the invocation of an inappropriate tool.

Results: The four dimensions of the transformation of the designer's profession

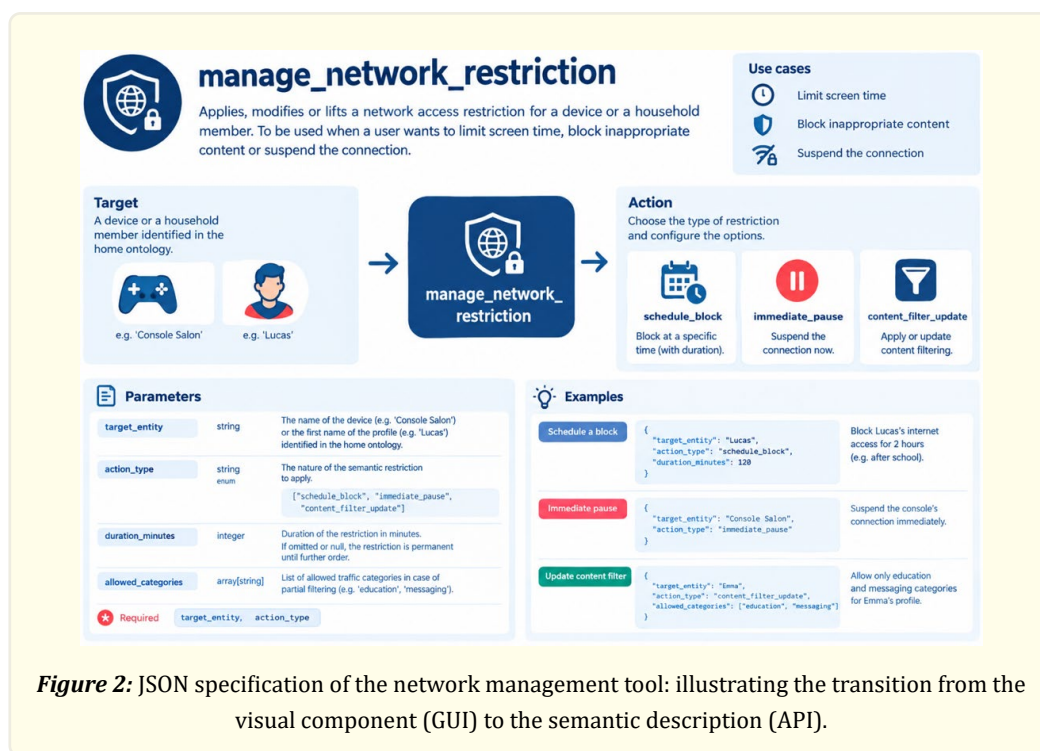
The functional aspect: From UI components to semantic API specifications

In the traditional design of an internet box, implementing a complex feature such as parental controls or security filtering requires the UX designer to create a multitude of graphical screens. They must design a form for selecting the child's profile, a grid for selecting access time slots, a tick box list of prohibited website categories and a summary dashboard of associated devices. The abundance of visual elements places a heavy cognitive burden on users, often leading non-tech-savvy individuals to abandon the platform due to its administrative complexity.

Within the framework of agent-centred design (ACD), the designer's role shifts from creating visual interfaces to engineering intentions and specifying semantic action capabilities (*tool definitions*). The designer no longer designs the 15 screens of the filtering form; instead, they draft the formal description in natural language and the JSON structure of the function that the software agent will execute when interpreting an explicit or contextual request from the user. The JSON code example below illustrates the artefact produced by the designer for this functionality:

```
{
  "type": "function",
  "function": {
    "name": "manage_network_restriction",
    "description": "Applies, modifies or removes a restriction on access to the local network or the internet for a device or a member of the household. To be used when a user wishes to limit screen time, block inappropriate content or suspend the connection.",
```

```
"parameters": {  
  "type": "object",  
  "properties": {  
    "target_entity": {  
      "type": "string",  
      "description": "The name of the device (e.g. 'Living Room Console') or the first name of the profile (e.g. 'Lucas') as identified in the household ontology."  
    },  
    "action_type": {  
      "type": "string",  
      "enum": ["schedule_block", "immediate_pause", "content_filter_update"],  
      "description": "The nature of the semantic restriction to be applied."  
    },  
    "duration_minutes": {  
      "type": "integer",  
      "description": "Duration of the restriction in minutes. If omitted or set to zero, the restriction is permanent until further notice."  
    },  
    "allowed_categories": {  
      "type": "array",  
      "items": { "type": "string" },  
      "description": "List of permitted traffic categories in the event of partial filtering (e.g. 'education', 'messaging')."  
    }  
  },  
  "required": ["target_entity", "action_type"]  
}
```



This shift in artefact is fundamental: the clarity, precision and removal of ambiguity in the textual description of the 'description' field constitute the new semantic affordance. If the description written by the designer is imprecise, the agent will make errors when invoking the tool or will select an inappropriate one. The designer's task is therefore to ensure that the machine interprets the system's capabilities unambiguously.

To illustrate this requirement for semantic engineering in practical terms, let us consider the case of complex requests made by human users in everyday language that is rich in implicit meaning, such as: *'Turn off the children's Wi-Fi whilst they are revising for their exams'*. In a traditional system, such a request would require navigating through settings, checking which devices correspond to the children's phones or tablets, estimating the duration of their revision, and manually applying a block. With the semantic specification defined above, the agent is able to analyse the intention, consult the household ontology to identify the devices associated with the 'children' profile, select the action 'immediate_pause' or 'schedule_block', and execute the 'manage_network_restriction' function by passing the correct JSON arguments. If any ambiguity remains (for example, if one of the children is using their laptop to access educational resources needed for revision), the agent can invoke a subroutine to request targeted clarification from the user before applying the rule.

The hardware aspect: From physical affordance to embedded symbiotic explainability

The physical dimension of the 'N-Gen' internet box is undergoing an equally radical transformation. The history of consumer network equipment has been characterised by the use of LED indicators, whose visual language remains incomprehensible to the majority of users. A diode flashing rapidly in orange or turning a steady red forces the user to consult a paper manual or decipher an abstract error code. This approach stems from a design philosophy in which the device passively displays its internal state without providing a causal explanation.

The introduction of an autonomous software agent changes the role of the hardware: the physical casing becomes the machine's means of expression and explanation. Equipped with a high-visibility dynamic display, the casing no longer shows static status icons,

but translates the agent's reasoning in real time through patterns of symbiotic explicability (*Chain-of-Thought UI*). Industrial designers and interaction designers no longer design the form of screen-printed graphics on plastic, but define the narrative and visual structure through which the agent accounts for its actions.

When a service disruption occurs (for example, a drop in data rate or a fibre-optic disconnection), the on-board agent does not display a technical error code. It generates a textual and graphical summary structured according to the template designed by the designer: *[Incident report] → [Root cause analysis carried out by the agent] → [Autonomous action taken] → [Simple instruction for the user]*. The device's screen therefore displays: 'Disruption detected on the main fibre line. The agent has switched priority traffic (remote working) to the backup 5G network. No action required'. The physical affordance no longer lies in a button to press, but in the provision of immediate transparency that maintains user confidence and avoids unnecessary manipulation of the equipment (figure 3).

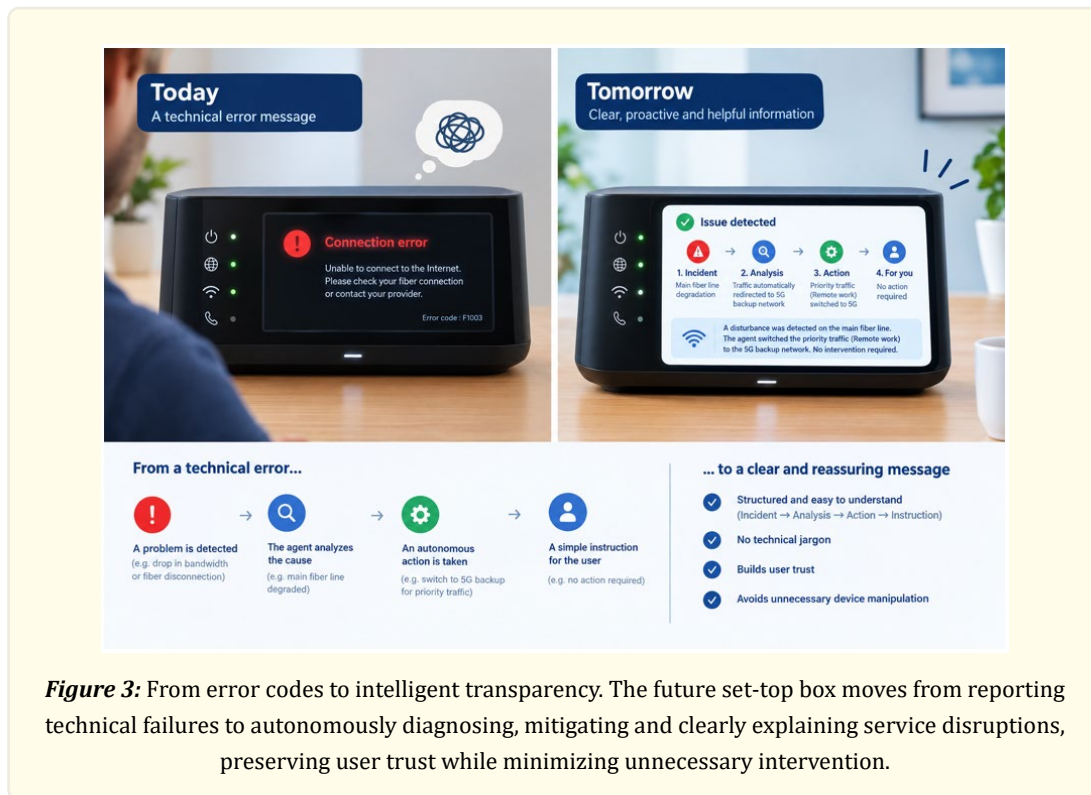


Figure 3: From error codes to intelligent transparency. The future set-top box moves from reporting technical failures to autonomously diagnosing, mitigating and clearly explaining service disruptions, preserving user trust while minimizing unnecessary intervention.

This built-in explainability resolves one of the major pitfalls identified by Bainbridge (1983) in the study of automated systems: causal opacity. Under normal circumstances, a highly automated system operates transparently, but when an unforeseen failure occurs, the human operator finds themselves thrust unprepared into a technical situation about which they know nothing. By requiring the user to articulate their technical line of reasoning (*Chain-of-Thought*) on the physical screen, the designer reintroduces a shared thread of awareness between the equipment and the household's occupants. The physical screen no longer serves merely as a monitor of past events, but as a channel for reassurance and active education on the functioning of the home network.

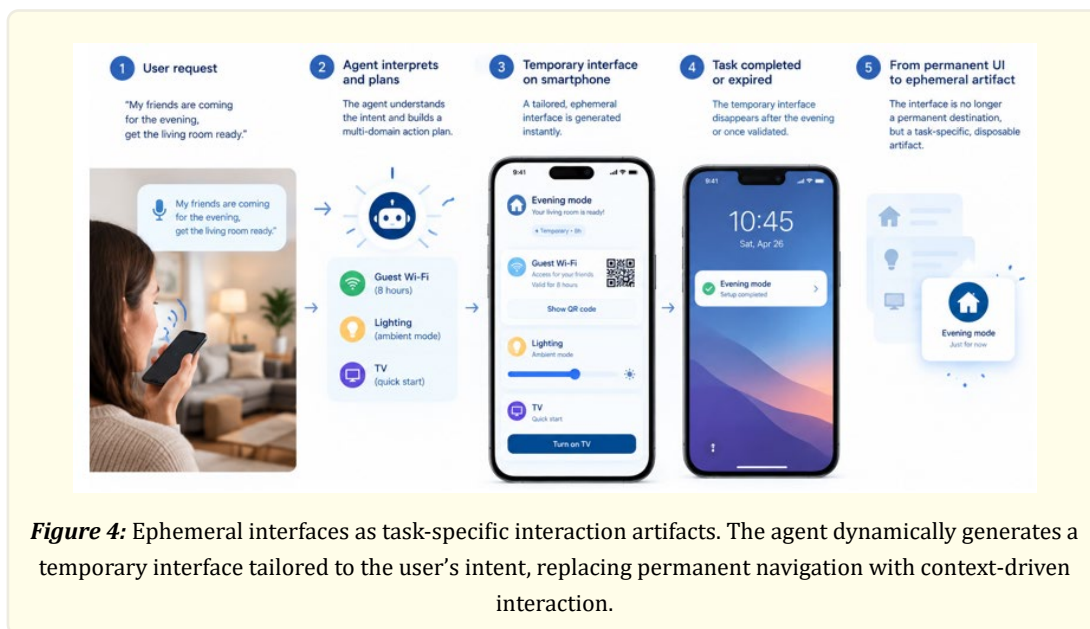
Mobile software control: The generation of ephemeral interfaces (*On-the-fly UI*)

In the traditional UX paradigm, the mobile app for managing the internet router is a cumbersome structure, comprising dozens of sub-menus organised hierarchically. To carry out a one-off action — such as granting Wi-Fi access to guests spending the evening over — the user must navigate to the *Settings* section, then *Guest Network*, toggle a virtual switch, configure a network name (SSID), set a password and finally display a QR code (except for devices running OS X). This structural rigidity forces the user to conform to the

classification logic chosen by the app's designers.

Under the ACD/AIX paradigm, the mobile app moves away from this fixed tree structure to become a dynamic canvas of ephemeral interfaces (*On-the-fly UI*). The designer no longer creates rigid navigation paths, but rather a system of modular graphical components that the agent can assemble, display and then dismantle depending on the user's expressed intention or the detected context.

When the user utters a natural language phrase such as: '*My friends are coming over for the evening, get the living room ready*', the agent interprets the overall intention, formulates a multi-domain action plan and instantly generates a temporary interface map on the smartphone screen. This map concisely brings together a QR code for guest Wi-Fi access set to last for 8 hours, a slider to control the smart lighting ambience, and a quick command to switch on the television. Once the evening is over or the '' instruction has been confirmed, this specific interface disappears. The graphical interface ceases to be a permanent spatial destination and becomes a disposable interactive artefact, tailor-made to resolve an immediate task.



This approach, based on ephemeral interfaces, profoundly transforms the role of the company's *Design System*. In the traditional model, the *Design System* provides a library of stable visual components (buttons, text fields, cards, navigation bars) intended to be assembled manually by interface designers when preparing mock-ups in Figma. In the agent-based model, this same *Design System* is enriched with semantic metadata and fed directly into the application's rendering engine. It is the agent itself that dynamically selects the most appropriate visual components to build the temporary interface best suited to the user's expressed intent, whilst adhering to the accessibility constraints and graphic consistency rules defined in advance by the design team.

System governance: Agency Sliders

The crux of the transformation in design work lies in modelling the boundaries of delegated autonomy. When a system has the capacity to act independently within the home network (for example, cutting off access, reallocating bandwidth, purchasing an additional data allowance or blocking a device deemed malicious), the risk of rejection or a sense of loss of control on the part of the user is at its highest (Bainbridge, 1983; Shneiderman, 2022).

The designer's role is no longer to mask technical complexity behind graphical animations, but to design the agent's governance framework. This task is carried out by modelling '*Agency Sliders*', which enable the agent's actions to be categorised according to three

levels of criticality and the need for human intervention:

1. **Level 1: Full autonomy (silent execution).** Applies to technical operations with a high level of certainty and low risk of adverse impact on the user. Example: the dynamic optimisation of Wi-Fi channels in the event of interference with neighbouring networks, or the seamless switch to a backup mobile connection during a minor fibre-optic outage. The agent carries out the task without interrupting the user and records the action in a passive audit log.
2. **Level 2: Supervised autonomy (Notification and right of veto).** This applies to moderately critical actions that alter the occupants' experience without involving any major financial or security implications. Example: the detection of a new, unidentified device attempting to connect to the local network. The agent temporarily isolates the device as a precaution, sends a brief notification to the administrator's smartphone and allows them to lift the restriction with a single tap.
3. **Level 3: Delegation subject to prior approval (mandatory human intervention).** Applies to irreversible decisions, actions involving a financial cost, or significant changes to privacy policies. Example: automatic subscription to a paid 5G data pass when a data allowance is exceeded, or the transmission of comprehensive network diagnostics to a third-party technician. The agent prepares the action, drafts a summary of the reasons and consequences, and suspends its execution until explicit consent is obtained from the user.

The design of these three levels of autonomy requires meticulous risk analysis and contextual modelling. The same technical action may, in fact, shift from one level of autonomy to another depending on the household's operational context. For example, temporarily disconnecting a device may fall under Level 1 if it is a faulty smart bulb generating background interference, under Level 2 if it is a games console being used in the evening, and under Level 3 if it is a parent's laptop identified as being in the middle of a remote working session. The designer's role is precisely to formalise these context matrices in order to avoid any conflicts of use and to ensure that the agent never exceeds its implicit mandate.

Discussion and ethical critique

The evolution of skills: From visual empathy to semantic Taylorisation

The transition from UX to AIX represents not merely a change in tools, but an ontological metamorphosis of the designer's profession (Vosgiens, Olry, & Vivian, 2026). Historically, the added value and professional legitimacy of the UX designer rested on their ability to demonstrate empathy towards human beings, to translate psychological and physiological needs into visual forms, and to act as the user's advocate in the face of engineering constraints.

By becoming a practitioner of Agent-Centred Design (ACD), the designer sees their field of activity shift towards ontological engineering, the formalisation of data schemas, the optimisation of context windows and the fine-tuning of system prompts. This shift carries the risk of a form of semantic Taylorisation of the profession. Stripped of their creative, visual and cultural dimensions, designers risk being relegated to the role of metadata administrators, tasked with formatting the human world to make it digestible and processable by the latent layers of language models. Design research must urgently challenge this loss of aesthetic signature and ensure that alignment with the machine does not come at the cost of impoverishing the sensory qualities of interfaces.

This shift also raises questions about the initial and continuing training of design professionals. Higher education programmes in HCI and interaction design, traditionally focused on visual ergonomics, theories of perception, graphic prototyping and qualitative user evaluation, must now incorporate skills drawn from computational linguistics, formal logic and software engineering. If this technical shift is not managed properly, the discipline of design risks losing its status as a humanist counterbalance within technology organisations, becoming merely a sub-branch of prompt engineering and data structuring for language models.

The closed-loop trap and the drift towards synthetic personas

One of the most worrying methodological developments linked to the development of AIX lies in the temptation to replace real human users with 'Synthetic Personae'. In the name of cost-efficiency and faster development cycles, AI agents are now trained to

simulate the behaviours, biases, frustrations and cognitive responses of target user groups during testing phases.

This practice traps the design process in a potentially sterile computational loop. Designers now spend their time designing AI systems which they then have evaluated by other AI systems. This closed-loop process eliminates otherness, resistance from the field, behavioural unpredictability and the empirical reality of the lived world. There is a major risk that systems will emerge which are perfectly optimised to pass tests conducted by synthetic agents, yet are completely disconnected from the needs, cultural variations and emotional subtleties of biological human beings.

Furthermore, the widespread use of Synthetic Personas tends to invisibly amplify the historical biases contained in the datasets used to pre-train language models. When a synthetic agent simulates the behaviour of an older user or someone with limited technical expertise when interacting with an internet set-top box interface, it is merely expressing a statistical caricature of technical incompetence shaped by its training data. By validating design choices on the basis of these biased simulations, designers risk reinforcing usage stereotypes and closing off interfaces to the real diversity of everyday practices, creating a self-fulfilling prophecy in which the machine validates its own assumptions about humans.

Proposals for mixed evaluation metrics (AIX Metrics)

Given the obsolescence of traditional metrics for measuring ergonomic quality (such as the SUS System Usability Scale score or on-screen typing time), it is becoming essential to formalise new evaluation metrics suited to hybrid ecosystems. These metrics must be capable of capturing both the agent's computational performance and the preservation of human cognitive integrity. We propose a framework structured around four AIX evaluation indicators:

1. ***The Intent Alignment Rate (IAR)***: Measures the proportion of times when the action plan generated by the agent and the functions invoked correspond exactly to the human user's implicit objective, without requiring clarification or retroactive correction.
2. ***Transparency Latency (TL)***: Calculates the time taken by the agent to display a comprehensible explanation of its action on the on-board screen or mobile application in the event of an error or network incident.
3. ***Computational efficiency and frugality (Token-to-Task Ratio - TTR)***: Assesses the amount of computational resources (tokens consumed, inference requests, computing power required) needed to resolve a user request. A good AIX design must maximise information density whilst minimising the system's carbon footprint and inference cost.
4. ***Supervision Overhead Rate (SOR)***: Measures the decision-making burden imposed on the human user by assessing the frequency of arbitration requests generated by the agent. A well-designed system must minimise unjustified interruptions whilst ensuring an optimal level of security.

The operationalisation of these four AIX metrics requires the implementation of mixed-methods experimental protocols within HCI laboratories. The evaluation of an agent-based internet box such as N-Gen can no longer be limited to observing whether a user manages to find a button in a menu in under 30 seconds. It must measure, over extended test periods (several weeks under real-life domestic conditions), the evolution of the trust placed in the agent, the relevance of the decisions required of the resident, and the overall energy balance of the inference requests generated by the household. It is only through this dual assessment—both computational and psycho-social—that the true quality of an agent-centred experience can be scientifically established.

The Visual Instability Paradox: Appropriation Crisis, Loss of Skill Transferability, and Operational Pathways to Resolution

The dynamic generation of ephemeral interfaces on the fly (*On-the-fly UI*) introduces a major ergonomic and cognitive vulnerability that HCI literature cannot underestimate: the breakdown of procedural automation, spatial memory, and mental model continuity. Human learning, appropriation, and mastery of a digital artifact historically rely on repetition, predictability, and the stability of visuospatial cues (Norman, 1988; Nielsen, 1993). By continuously altering the layout, form, and hierarchy of graphical components based on instantaneous context or the stochastic interpretation of the agent, the system deprives the user of two fundamental pillars of embodied cognition:

1. **Procedural Appropriation and Sensorimotor Memory:** Users can no longer consolidate automated action routines (muscle memory, reflex visual scanning, optimized saccadic eye movements). Every interaction re-imposes a costly phase of visual exploration and decoded analysis, converting a routine secondary task into a problem requiring constant focused attention.
2. **Transferability and Cognitive Schemas:** The inability to build upon acquired operational knowledge to transpose it from one screen or session to another breaks the continuity of mental models. Users find themselves plunged into a state of structural interface amnesia, where expertise gained during past interactions becomes obsolete.

This extreme adaptation paradox risks turning purportedly “seamless” and “intelligent” interfaces into sources of cognitive friction, disorientation, and decision fatigue. Faced with a protean visual environment that shifts beneath their touch, users develop a sense of learned helplessness and a loss of the locus of control, leading either to system rejection or passive submission (Bainbridge, 1983; Shneiderman, 2022).

To overcome this bottleneck without forfeiting the adaptive flexibility of the agent, Agent-Centred Design (ACD) must move beyond the myth of free visual generation (*ex nihilo*) to formalize a rigid framework of Interface Variability Governance. This framework relies on the architectural implementation of four Semantic-Spatial Consistency Anchors:

- **Invariant Spatial Grid Topologies:** The designer defines an immutable spatial matrix dividing the screen into fixed functional zones: a *Status and Reasoning Zone* (agent explicability), a *Critical Control Zone* (irreversible commands), and an *Adaptive Context Zone* (ephemeral components). While the agent retains the freedom to adapt components within its designated zone, the overall topology remains predictable, preserving the user’s visual map.
- **Fixed Metaphorical and Semantic Grammar:** The agent is constrained to obey a strict correspondence system between visual form and criticality level. For instance, the color, elevation level, geometric shape, or haptics of a virtual component invariably signal the degree of action reversibility or the required autonomy level (Levels 1, 2, or 3 of the *Agency Sliders*). The user decodes the intent of an ephemeral component without prior learning.
- **Adaptive Inertia and Temporal Regularization:** The interface must not mutate with every micro-variation of context. Designers program temporal inertia rules that enforce a minimum visual stability over defined time windows (e.g., maintaining the same layout throughout a single usage session), preventing flickering or untimely rearrangements.
- **Anchoring Declaration and Elasticity Control (UI Elasticity & Pinning Control):** The system incorporates a feedback mechanism allowing humans to extract an agent-generated ephemeral component and “pin” it permanently into their workspace. This operation compels the agent to perpetuate this graphical view as a deterministic menu, transforming a one-off machine intuition into a stabilized, reusable usage routine.

By theorizing these Consistency Anchors, Agent-Centred Design does not aim to stifle agent intelligence, but to engineer the invariant constraint ecosystem within which algorithmic generativity can manifest without destroying human appropriation.

Conclusion

The prospective and empirical analysis conducted throughout this paper demonstrates that the advent of autonomous agent architectures and Artificial Intelligence Experience (AIX) represents far more than a mere tool update or an incremental technical enhancement within Human-Computer Interaction (HCI). It constitutes a major epistemological rupture that shakes the theoretical foundations established over four decades by the User-Centred Design (UCD) model. Emancipating from the status of reactive, deterministic tools to become entities endowed with stochastic action capabilities, machines cease to be passive receivers of graphical interfaces drawn exclusively for human visual perception. They now assert themselves as the primary subjects, decisional intermediaries, and chief consumers of digital structures (Vosgiens, Olry, & Vivian, 2026).

Through the in-depth industrial case study of the “N-Gen” next-generation Internet router, this paper has operationalized the Agent-Centred Design (ACD) framework and demonstrated the concrete metamorphosis across all product dimensions:

- **Functionally**, storyboarding fixed trees and drawing complex visual forms yield to the rigorous specification of API semantic capabilities (*tools*), system prompts, and household ontology models;
- **Hardware-wise**, the passive exposure of technical states via LED indicator lights yields to dynamic reasoning explicability patterns (*Chain-of-Thought UI*), re-establishing a shared thread of awareness on the embedded display;
- **Mobile software-wise**, the rigidity of hierarchical menus dissolves into the on-the-fly generation of ephemeral, disposable interfaces tailored to instantaneous intent resolution;
- **Governance-wise**, traditional visual ergonomics yield to the formal modeling of risk matrices and autonomy regulation sliders (*agency sliders*).

However, the contributions of this research extend beyond observing this tool mutation by illuminating a fundamental ergonomic and cognitive pitfall previously underestimated by advocates of extreme algorithmic personalization: **the cognitive cost of interface instability**. By generating protean graphical components that mutate according to real-time context or agent stochasticity, the system destroys spatial repetition and procedural predictability—the very pillars essential for sensorimotor memory and mental model consolidation. Deprived of automated routines and the ability to transfer skills across screens, users face a structural interface amnesia that induces decision fatigue, disorientation, and a diminished sense of control.

To resolve this central contradiction between agentic flexibility and human appropriation, this paper has laid the theoretical foundations for **Interface Variability Governance** through the concept of **Semantic-Spatial Consistency Anchors**. We have demonstrated that the role of the agent-centred designer is not to grant the machine total freedom of visual generation *ex nihilo*, but to engineer an architecture of invariant constraints: invariant surface topologies, fixed metaphorical grammars, temporal inertia rules, and user-controlled elasticity (*pinning control*) allowing humans to perpetuate emerging routines into deterministic menus.

This reveals a profound redefinition of professional design responsibilities. Tomorrow's designer will no longer be an executor of static mockups or a drawer of screen layouts. They must navigate between the threat of semantic Taylorization—which would reduce them to metadata administrators for language models—and the risk of methodological alienation caused by confinement within closed simulation loops via Synthetic Personas. By asserting themselves as architects of consistency rules, mediators of explicability, and ethical guardians of autonomy delegation thresholds, designers reclaim their core humanist mission. The ultimate ambition of AIX and Agent-Centred Design is not to construct fully autonomous systems that exclude humans from the command loop, but to civilize machine autonomy to ensure a symbiotic, predictable, and emancipatory coexistence between biological and synthetic intelligence.

References

1. Amershi S., et al. "Guidelines for human-AI interaction". In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (2019): 1-13.
2. Bainbridge L. "Ironies of automation". *Automatica* 19.6 (1983): 775-779.
3. Chiou L-Y., et al. "Designing with AI: An exploration of co-ideation with image generators". In: Proceedings of the 2023 ACM Designing Interactive Systems Conference (2023): 1410-1424.
4. Dove G., et al. "UX design innovation: Challenges for working with machine learning as a design material". In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (2017): 278-288.
5. Gibson JJ. "The ecological approach to visual perception". Houghton Mifflin, Boston (1979).
6. Heer J. "Agency plus automation: Designing artificial intelligence into interactive systems". *Proceedings of the National Academy of Sciences* 116.6 (2019): 1844-1850.
7. Inie N, Falk J and Tanimoto S. "Designing participatory AI: Creative professionals' worries and expectations about generative AI". In: Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (2023): 1-8.
8. Nielsen J. "Usability engineering". Academic Press, San Diego (1993).
9. Norman DA. "The psychology of everyday things". Basic Books, New York (1988).
10. Shneiderman B. "Human-centered artificial intelligence: Reliable, safe & trustworthy". *International Journal of Human-Computer*

Interaction 36.6 (2020): 495-504.

11. Shneiderman B. "Human-centered AI". Oxford University Press, Oxford (2022).
12. Subramonyam H., et al. "Solving separation-of-concerns problems in collaborative design of human-AI systems through leaky abstractions". In: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (2022): 1-16.
13. Vosgiens M, Olry A and Vivian R. "When AI becomes the user: Redefining the boundaries of human-machine interaction in the age of AIX". Manuscript submitted for publication (2026).
14. Yang Q., et al. "Re-examining whether, why, and how human-AI interaction is uniquely difficult to design". In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (2020): 1-13.
15. Zamfirescu-Pereira JD., et al. "Why Johnny can't prompt: How non-AI experts try (and fail) to design LLM prompts". In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (2023): 1-21.