

DriveAdapt-DUI: Prototyping a Distributed User Interface for a Smart Car

Alaa Eddine Anis Sahraoui

LouRIM, Université catholique de Louvain
Louvain-la-Neuve, Belgium
alaa.sahraoui@uclouvain.be

ABSTRACT

Adaptive vehicle interfaces reduce driver workload by re-configuring displays as context changes, but they usually do so without informing the occupants or allowing them to intervene. This paper presents DRIVEADAPT-DUI, an early prototype that applies the Extra-User Interface (Extra-UI) concept to a distributed, multi-device cockpit. An Extra-UI is layered over the target car interface so that occupants can inspect, accept, reject, and redistribute adaptations at runtime. Rules are expressed in the CEAV notation and positioned in the ISATINE lifecycle, and a safety-aware distribution policy restricts the moving driver to three low-cost actions while delegating richer control to the passenger and to post-trip review.

AUTHOR KEYWORDS

Distributed user interface; Extra-user interface; adaptive interface; automotive HCI; user control; transparency.

DOI: 10.37789/icusi.2026.4

INTRODUCTION

A contemporary vehicle integrates an instrument cluster, an infotainment console, a head-up display, passenger screens, and ambient lighting into a single interactive space [6]. An interface that hides irrelevant detail and relocates secondary controls to a less critical display can reduce driver workload, but the same mechanism can confuse or distract the occupant if the change is neither visible nor reversible.

This trade-off is well established in the literature. Adaptivity reduces cognitive load and improves contextual fit, but when it is applied without explanation the costs appear elsewhere: users lose their sense of control, can no longer predict system behaviour, and withdraw trust [1]. In a vehicle the consequences are more severe, since a driver whose navigation view is reduced by an invisible rule may not notice what was removed, and discovering this while driving is unsafe.

The Extra-User Interface, or Extra-UI [5, 14], addresses this problem. It is a meta-control layer that runs alongside the target application and exposes adaptation as something the user can act upon: observe it, read its rationale, accept or reject it, and relocate it. It has been applied to data-visualization dashboards [14], but not to a vehicle. The cockpit is a suitable setting, since it is already a Distributed User Interface (DUI) [10, 16], whose parts are spread across several displays used by more than one person.

This paper presents DRIVEADAPT-DUI, a prototype that instantiates the Extra-UI concept in a distributed cockpit. Its central design decision is to distribute the Extra-UI itself across five locations rather than to place a single adaptation panel within the driver's reach. Distribution serves a functional purpose here: it is the mechanism that enforces the

safety constraint, because while the vehicle is in motion the driver is offered only a minimal control and anything richer is delegated to a passenger or deferred to post-trip review. The contributions are a vehicular instantiation of the Extra-UI concept, a role-based and safety-aware distribution policy, and a modular prototype architecture connecting context sensing to adaptation rules and to a distribution mechanism.

RELATED WORK

Distributed and Meta-User Interfaces

A Distributed User Interface is an application user interface whose parts can be spread across displays, platforms, and users, along four dimensions of distribution: users, platforms, displays, and environments [16]. The operational vocabulary for building one is provided by distribution primitives, the elementary operations that display, hide, move, copy, or recombine components at runtime [9], together with the distribution graph, a state chart whose states are whole-system distribution states [10]. Coutaz introduced the meta-user interface as a layer controlling the behaviour of an ambient space [4], and Demeure et al. refined this into a family of supplementary interfaces [5], within which the Extra-UI is defined as an interface *for* an existing interface: it adds observability, configurability, and control over that interface's adaptation rather than replacing it. Prior work organises the concept into a design space of eight dimensions and a set of generic services, validated on ambient and data-visualization prototypes [14].

Adaptive Automotive Interfaces

Adapting in-vehicle interfaces to reduce driver distraction is framed by the NHTSA visual-manual guidelines [12]. Most deployed systems are system-controlled: the rule is applied silently, without notice and without consent. Graefe et al. [7] argue that automotive adaptivity should be understandable, transparent, and controllable, and Taylor et al. [15] report that transparent in-vehicle agents obtain higher behavioural trust, which indicates that explanation influences driver behaviour and not merely opinion. Calvary and Sahraoui [3] note that deployed systems implement only the middle stages of the adaptation lifecycle.

The Passenger as an Interaction Resource

A parallel line of research establishes the front-seat passenger as an active participant. Perterer et al. [13] observed driver-passenger pairs ethnographically and documented how passengers routinely assume navigation and secondary-task duties. Meschtscherjakov et al. [11] argue that this collaboration improves both driving experience and safety, since delegated tasks leave the driver's visual and manual resources available for the primary task, and Berger et al. [2] found explicit,

Table 1: Distributed Extra-UI components in DRIVEADAPT-DUI

Component	Role	Allowed interaction
Driver micro	Minimal awareness and quick control	Accept, undo, postpone
Central panel	Inspect adaptations and distribution state	Detailed control when stopped
Passenger	Manage delegated non-critical tasks	Modify route, media, suggestions
Post-trip app	Review history, configure future rules	Rule editing, personalization
Ambient	Low-effort adaptation awareness	Awareness only, no input

system-supported delegation preferable to informal arrangement.

These strands leave a gap. Adaptive automotive work determines *what* to adapt but rarely gives the user authority over the process; DUI research distributes content but not its governance; and Extra-UI research provides the governance concept but has not been placed in a vehicle, where several occupants are present and where safety imposes a ceiling on driver interaction.

DRIVEADAPT-DUI DESIGN

Design Problem and Distributed Components

The central design constraint is the conflict between usefulness and distraction. Simplifying the driver's view in dense traffic is beneficial, but requiring the driver to approve or configure that change while the vehicle is moving introduces exactly the secondary-task load that regulation seeks to suppress, since the NHTSA guidelines limit visual demand for an in-vehicle task to two seconds of eyes-off-road time per glance [12].

DRIVEADAPT-DUI addresses this by distributing the Extra-UI itself. Rather than placing a complete adaptation panel on the centre console within the driver's reach, the prototype allocates adaptation services to five locations, each chosen so that the interaction cost it permits matches the risk carried by the occupant who uses it (Table 1). The driver micro Extra-UI is restricted to three single-tap actions because any richer control would exceed the glance budget above. The central panel offers detailed inspection but unlocks it only when the vehicle is stopped, which removes the conflict rather than managing it. The passenger Extra-UI carries full control over delegated non-critical tasks, which is defensible precisely because the passenger bears no driving load [2, 11, 13]. The post-trip application holds rule editing, the operations with the highest cost and the lowest urgency, and the ambient channel requires no input at all, which makes it the only surface available to the driver under every condition.

Why CEAV and ISATINE Are Both Required

DRIVEADAPT-DUI draws on two existing models, and they are not interchangeable. CEAV [3] is a rule notation: Condition, Event, Action, Value. It extends the conventional event-condition-action rule with an explicit *value*, the intended benefit of the change. That field is what the prototype needs, because it is the only element of a rule that carries a rationale expressible to the user, and a rationale is precisely what an Extra-UI must display. ISATINE [8] is a process model: it decomposes an adaptation episode into seven stages and therefore identifies the points at which an Extra-UI can attach. It



Figure 1: DRIVEADAPT-DUI in dense traffic. The driver cluster shows a simplified view and a three-action micro Extra-UI; route and media detail are delegated to the passenger screen.

also functions as a diagnostic, since deployed vehicle systems implement application and transition while omitting initiative, interpretation, and evaluation, and those three omissions are what DRIVEADAPT-DUI supplies. One model is per-rule and static, the other per-episode and temporal. CEAV alone would yield rationales with no defined moment at which to present them; ISATINE alone would yield attachment points with nothing specific to present. The rule behind the dense-traffic scenario below illustrates the pairing. Its *CONDITION* is that the vehicle is moving, traffic complexity is high, and a passenger screen is available; its *EVENT* is entry into a dense-traffic zone; its *ACTION* simplifies navigation, suppresses non-critical alerts, and moves media controls to the passenger Extra-UI; and its *VALUE*, to reduce the driver's visual-manual demand while keeping secondary functions reachable, supplies the text that the micro Extra-UI displays at the interpretation stage.

USE-CASE SCENARIO

Figure 1 shows the situation the prototype targets. Traffic is dense, the driver's display has just been simplified, and a small panel offers three choices while route and media detail are relocated to the passenger screen.

During ordinary driving the centre display carries full navigation, media, and comfort controls. When the Context Manager determines that traffic complexity is high, from speed, inter-vehicle distance, and navigation alerts, the rule fires and the driver receives a short message: *"Interface simplified. Reason: dense traffic ahead. Media and route details moved to the passenger screen."* The cluster is reduced to speed, lane guidance, and the next instruction, and the micro Extra-UI presents Accept, Undo, and Explain Later. Media and route detail appear on the passenger Extra-UI, where they can be operated without cost to the driver. After the drive the occupant opens the post-trip application to review the event and decide whether to retain, edit, or disable the rule.

The scenario illustrates the separation the design maintains. The *target UI* handles driving, navigation, comfort, and information, while the *Extra-UI* is the meta-control surface for inspecting, approving, undoing, and redistributing the target UI's adaptations. Keeping the two apart confines the prototype to adaptation governance rather than a redesign of the vehicle's own systems.

PROTOTYPE ARCHITECTURE

Figure 2 shows two structural views of the prototype. Panel (a) is the module architecture, organised as a pipeline rather than

as peer services: the Context Manager reads vehicle, traffic, and driver state with the user profile and device availability; the Adaptation Engine selects candidates from the CEAV rule base and ranks them by expected value; the Distribution Manager decides which components appear on which surface; and the Extra-UI Orchestrator keeps the five instances consistent through a shared event log. Two modules sit across that pipeline rather than in it. The Safety Gatekeeper intercepts every proposed placement and rejects any interaction exceeding the complexity permitted by the driver's current state, which is where the policy of Table 1 is enforced rather than merely declared. The Logger records events and user responses, which is what makes post-trip review possible, since review is replay of that log.

Panel (b) is the information distribution map, showing which function classes are permitted on which display: critical alerts, speed, and navigation summary remain with the driver; media, route detail, and search move to the passenger; and non-critical notifications are withheld while the vehicle is moving. The two panels state the same policy at two levels, as module responsibility and as permitted content per surface.

Distribution behaviour is expressed as a state chart whose states record the joint distribution state of every node and whose transitions carry a distribution script [10]. The prototype implements four distribution primitives [9], exposed in code as `migrate`, `clone`, `suppress`, and `restore`. The dense-traffic transition uses all four: media and route widgets migrate to the passenger surface, the cluster media panel is suppressed, a read-only clone mirrors the simplified view for shared awareness, and an Undo restores the previous state. Because each primitive is explicit and reversible, any adaptation can be replayed in the post-trip application exactly as it occurred. The prototype runs as synchronized web clients, one per Extra-UI surface, with a React frontend, a Node.js backend, WebSocket synchronization, and a document store for rules, preferences, and logs.

DISCUSSION

There is a genuine tension between maximising an occupant's authority over adaptation and minimising what the driver must do. DRIVEADAPT-DUI resolves it by making the distribution policy editable: in post-trip review a driver can adjust which adaptations are accepted without confirmation, which raises or lowers the threshold at which the micro Extra-UI appears at all. The degree of system initiative is therefore a user setting rather than a fixed property of the vehicle.

Delegating control to the passenger is the decision most open to challenge, since it appears to move a burden rather than remove it. The collaboration literature suggests otherwise. Passengers already assume navigation and secondary tasks informally [13], this division of labour is associated with improved driving experience and safety [11], and explicit system support is preferred to ad hoc arrangement [2]. DRIVEADAPT-DUI therefore formalises an existing practice rather than imposing a new one, which is also why the policy degrades to awareness-only rather than to a driver-facing panel when no passenger is present.

The event log that makes post-trip review possible also retains a detailed trace of context and behaviour, so the transparency the Extra-UI provides has to extend to that data: occupants can inspect what is recorded, and the trace remains on the vehicle or the occupant's own device by default.

LIMITATIONS AND FUTURE WORK

This is a design and prototype contribution and no empirical results are reported. The Context Manager currently reads scripted and emulator-sourced context rather than a full driving simulator, so tighter simulator coupling and calibration of the safety thresholds remain to be completed. The static role assignment assumes a two-occupant cockpit, and the work centres on a single scenario, so other events may stress the design differently.

The planned evaluation is a within-subjects driving-simulator study comparing a silent adaptive baseline, a centralized Extra-UI with all controls on the centre console, and the distributed DRIVEADAPT-DUI policy, measuring distraction through glance behaviour, workload, perceived control, transparency, and trust. The centralized condition is the informative comparison, since it also offers transparency and control, so the contrast tests whether *where* the control resides matters and not only whether it exists. Adaptation batching is a further open question, since a vehicle raising a separate prompt for every event would overload the driver as the event rate grows.

CONCLUSION

This paper presented DRIVEADAPT-DUI, a prototype that treats the cockpit as a multi-device distributed user interface and layers an Extra-UI governance surface over it, so that occupants can observe, explain, accept, reject, and configure the adaptations applied to the primary vehicle interface, with a safety-aware policy allocating that control across five surfaces according to interaction cost and driving state. The wider argument is that the conflict between adaptive usefulness and driver safety need not be resolved by withdrawing control from the user: distributing control by role can serve both requirements at once.

GENERATIVE AI DISCLOSURE

The author used a generative AI assistant for language editing and for cross-checking bibliographic metadata. All concepts, the design, and the architecture are the author's own, and the author takes full responsibility for the content.

ACKNOWLEDGMENTS

The author thanks LouRIM at the Université catholique de Louvain for supporting this research.

REFERENCES

- [1] Victoria Bellotti and Keith Edwards. 2001. Intelligibility and Accountability: Human Considerations in Context-Aware Systems. *Human-Computer Interaction* 16, 2-4 (2001), 193–212. doi:10.1207/S15327051HCI16234_05
- [2] Melanie Berger, Debargha Dey, Aditya Dandekar, Bahareh Barati, Regina Bernhaupt, and Bastian Pfleging. 2022. Together in the Car: A Comparison of Five Concepts to Support Driver-Passenger Collaboration. In *Proc. AutomotiveUI '22*. ACM, 183–194. doi:10.1145/3543174.3544940
- [3] Gaëlle Calvary and Alaa Eddine Anis Sahraoui. 2025. Contextual Adaptation of User Interfaces. In *Handbook of Human-Computer Interaction*, Jean Vanderdonck, Philippe Palanque, and Marco Winckler (Eds.). Springer, Cham, 1–47. doi:10.1007/978-3-319-27648-9_17-1
- [4] Joëlle Coutaz. 2007. Meta-User Interfaces for Ambient Spaces. In *Proc. TAMODIA (Lecture Notes in Computer Science, Vol. 4385)*. Springer, 1–15. doi:10.1007/978-3-540-70816-2_1
- [5] Alexandre Demeure, Grzegorz Lehmann, Mathieu Petit, and Gaëlle Calvary. 2011. Enhancing Interaction with Supplementary Supportive UIs: Meta-UIs, Mega-UIs, Extra-UIs, Supra-UIs. ... In *Proc. EICS*. ACM, 333–334. doi:10.1145/1996461.1996552
- [6] Fei Gao, Xiaojun Ge, Jinyu Li, Yuze Fan, Yun Li, and Rui Zhao. 2024. Intelligent Cockpits for Connected Vehicles: Taxonomy, Architecture, Interaction Technologies, and Future Directions. *Sensors* 24, 16 (2024), 5172. doi:10.3390/s24165172

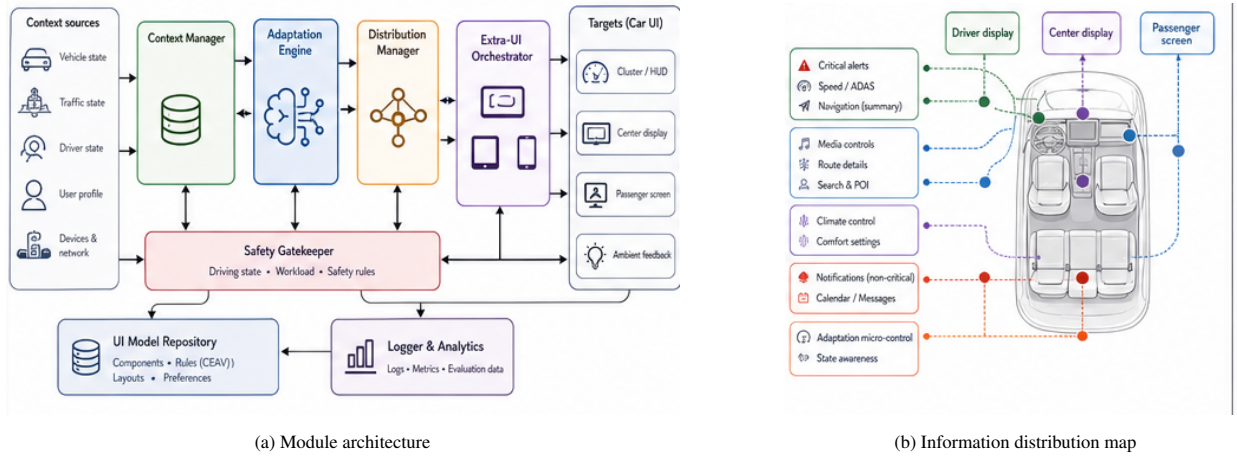


Figure 2: Structure of the DRIVEADAPT-DUI prototype. (a) The seven modules, with the Safety Gatekeeper filtering every placement proposed by the Distribution Manager and the Logger recording the event stream that post-trip review replays. (b) The function classes permitted on each display under the role-based policy.

- [7] Julia Graefe, Doreen Engelhardt, and Klaus Bengler. 2021. What Does Well-Designed Adaptivity Mean for Drivers? A Research Approach to Develop Recommendations for Adaptive In-Vehicle User Interfaces That Are Understandable, Transparent and Controllable. In *Adjunct Proc. AutomotiveUI*. ACM, 43–46. doi:10.1145/3473682.3480261
- [8] V. López-Jaquero, J. Vanderdonckt, F. Simarro, and P. González. 2007. Towards an Extended Model of User Interface Adaptation: The ISATINE Framework. In *Engineering Interactive Systems (Lecture Notes in Computer Science, Vol. 4940)*. Springer, 374–392.
- [9] Jérémie Melchior, Jean Vanderdonckt, and Peter Van Roy. 2011. Distribution Primitives for Distributed User Interfaces. In *Distributed User Interfaces*, José A. Gallud, Ricardo Tesoriero, and Victor M. R. Penichet (Eds.). Springer, London, 23–31. doi:10.1007/978-1-4471-2271-5_3
- [10] Jérémie Melchior, Jean Vanderdonckt, and Peter Van Roy. 2011. Modelling and Developing Distributed User Interfaces Based on Distribution Graph. In *Proc. EICS*. ACM, 1–10.
- [11] Alexander Meschtscherjakov, Nicole Perterer, Sandra Trösterer, Alina Krischkowsky, and Manfred Tscheligi. 2017. The Neglected Passenger: How Collaboration in the Car Fosters Driving Experience and Safety. In *Automotive User Interfaces*, Gerrit Meixner and Christian Müller (Eds.). Springer, Cham, 187–213. doi:10.1007/978-3-319-49448-7_7
- [12] National Highway Traffic Safety Administration. 2013. *Visual-Manual NHTSA Driver Distraction Guidelines for In-Vehicle Electronic Devices*. Technical Report. U.S. Department of Transportation. Federal Register 78(81):24818–24890.
- [13] Nicole Perterer, Petra Sundström, Alexander Meschtscherjakov, David Wilfinger, and Manfred Tscheligi. 2013. Come Drive with Me: An Ethnographic Study of Driver-Passenger Pairs to Inform Future In-Car Assistance. In *Proc. CSCW '13*. ACM, 1539–1548. doi:10.1145/2441776.2441952
- [14] Alaa Eddine Anis Sahraoui. 2025. Supporting User Control of User Interface Adaptation with an Extra-UI. In *Companion Proc. EICS '25*. ACM, Trier, Germany.
- [15] Skye Taylor, Manhwa Wang, and Myounghoon Jeon. 2023. Reliable and Transparent In-Vehicle Agents Lead to Higher Behavioral Trust in Conditionally Automated Driving Systems. *Frontiers in Psychology* 14 (2023), 1121622. doi:10.3389/fpsyg.2023.1121622
- [16] Jean Vanderdonckt. 2010. Distributed User Interfaces: How to Distribute User Interface Elements Across Users, Platforms, and Environments. In *Proc. Interacción*. Universidad Politécnica de Valencia.