Reimagining Databox with User-Facing Agents

Josh Millar, Amir Al Sadi, Hamed Haddadi
Imperial College London

Ryan Gibb, Anil Madhavapeddy
University of Cambridge

1 Challenges in Managing Personal Data

The mining and profiling of users' behaviors and relationships is the basis on which most online platforms and services operate. Yet centralized control by large platforms often leaves individuals with limited visibility and recourse over their data usage. Moreover, such platforms typically have only partial views of individuals' data footprints, which can encourage overly aggressive data collection strategies, and result in inaccuracies and bias in the data they hold. This is only becoming more of a concern as personal data is increasingly used to train LLMs and other foundation models, often without transparency, consent, or deletion guarantees [2, 4, 6].

Consider a typical smart home ecosystem, composed of heterogeneous IoT devices (lights, thermostats, security cameras, voice assistants) from different manufacturers, each connecting to their own online service. With such fragmented infrastructure, and often opaque privacy policies, *how do we provide data subjects with meaningful, enforceable control over their digital footprint*?

2 The Databox Architecture

Databox is a hybrid personal data infrastructure proposed to challenge the prevailing centralized data model [1]. The common-case Databox setup combines physical device(s) augmented by cloud-hosted services that collate, curate, and mediate access to our personal data. By adding a physical layer, Databox provides affordances unavailable to a pure cloud-hosted solution (*e.g.*, proximity-based access control), alongside improved resilience and latency.

Fig. 1 outlines the core Databox architecture [3]. Databox follows a micro-services framework: components exist in separate containers communicating via explicit APIs, helping to promote portability between physical and cloud hosting. Dedicated *drivers* interface with data sources (*e.g.*, smart meters, APIs, physical devices) and write to versioned append-only *stores* enforcing local access policies. Having a distinct *store* for each data source provides granular control over access permissions, alongside improved security guarantees. Databox *apps* represent third-party software, are isolated/sandboxed by default, and must explicitly declare their input/output requirements. The *manager* acts as the control plane, responsible for maintaining app/driver containers, managing access permissions, logging data flows for audit, and routing communication between *apps*, *stores*, and external parties.

3 User-Facing Agentic Layer

Various social challenges have limited the wider adoption of the Databox, such as inadequate regulation compelling data processors to provide users access to their personal data. The main technical hurdle, however, lies in reducing the complexity of configuring the Databox and managing its data flows. To address this, we propose an *agentic interface layer* atop the core architecture.

Fig. 1 illustrates how this layer integrates with the Databox framework, . By augmenting the Databox's interface with a user-facing agent, we hugely improve its usability – the user (i.e., the data subject) can govern flow between available data sources, stores, and processors, mediated by verifiable, auditable agents. For example,

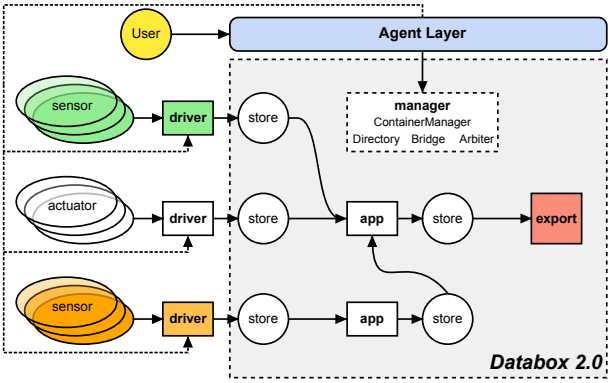

Figure 1: The Databox architecture, with agentic layer [3]

in our smart home scenario, a *driver agent* might discover a new network/IoT device, and then write the necessary code to ingest its data [5]. Meanwhile, a *policy agent* learns household routines and authors context-aware data sharing rules: "Your voice assistant searches reveal sensitive health queries – should I anonymize certain categories before sharing with the service provider?".

With external interaction mediated via inter-Databox protocols, agents assist users in curating purpose-scoped, *shared* containers within their personal data stores. Moreover, the Databox should filter and review data processing requests to ensure compatibility with the user's privacy preferences, and a *recommendation agent* could provide tailored recommendations to non-technical data owners about their risks. For example, when a smart home platform requests access to power consumption data, the agent might propose sharing aggregated daily totals rather than full usage data that could reveal personal habits. Over time, the agentic layer can monitor data logs for dynamic policy generation based on changing user preferences, activity patterns, and external context.

In this talk, we outline how agentic abstractions can contribute towards a more practical, user-tailored personal data architecture.

References

[1] Hamed Haddadi, Heidi Howard, Amir Chaudhry, Jon Crowcroft, Anil Madhavapeddy, and Richard Mortier. 2015. Personal Data: Thinking Inside the Box. arXiv:1501.04737 [cs.CY] https://arxiv.org/abs/1501.04737

[2] Shayne Longpre, Robert Mahari, Naana Obeng-Marnu, William Brannon, Tobin South, Katy Gero, Sandy Pentland, and Jad Kabbara. 2024. Data Authenticity, Consent, & Provenance for AI are all broken: what will it take to fix them? arXiv:2404.12691 [cs.AI] https://arxiv.org/abs/2404.12691

[3] Richard Mortier, Jianxin Zhao, Jon Crowcroft, Liang Wang, Qi Li, Hamed Haddadi, Yousef Amar, Andy Crabtree, James Colley, Tom Lodge, Tosh Brown, Derek McAuley, and Chris Greenhalgh. 2016. Personal Data Management with the Databox: What's Inside the Box?. In *Proceedings of the 2016 ACM Workshop on Cloud-Assisted Networking* (Irvine, California, USA) *(CAN '16)*. Association for Computing Machinery, New York, NY, USA, 49–54. https://doi.org/10.1145/3010079.3010082

[4] Henrik Nolte, Michèle Finck, and Kristof Meding. 2025. Machine Learners Should Acknowledge the Legal Implications of Large Language Models as Personal Data. arXiv:2503.01630 [cs.LG] https://arxiv.org/abs/2503.01630

[5] Leming Shen, Qiang Yang, Xinyu Huang, Zijing Ma, and Yuanqing Zheng. 2025. GPIoT: Tailoring Small Language Models for IoT Program Synthesis and Development. arXiv:2503.00686 [cs.SE] https://arxiv.org/abs/2503.00686

[6] Biwei Yan, Kun Li, Minghui Xu, Yueyan Dong, Yue Zhang, Zhaochun Ren, and Xiuzhen Cheng. 2024. On Protecting the Data Privacy of Large Language Models (LLMs): A Survey. arXiv:2403.05156 [cs.CR] https://arxiv.org/abs/2403.05156