Abstract—This vision paper considers the challenge of providing users with control over how information about them is used, in an ecosystem of assistants that help people achieve personal goals based on lots of data about their activities.

I. PERSONAL GOAL SYSTEMS: THE CONTEXT

Emerging pervasive and ubiquitous computing technologies are creating new ways to help people achieve a diversity of their most important long-term goals, and new businesses are arising to exploit these technologies. A user can now capture large amounts of relevant personal information with modest effort: a sensor on a wrist band can measure heart rate, an application might track from keystrokes how often one is sitting at the computer, and another might ask you after every meal to indicate how many serves of vegetables you ate. The recently announced Apple Watch and Microsoft Band show that major vendors are focused on this trend, with forthcoming products that will collect diverse measurements and communicate them to applications. Products such as Fitbit Tracker (a step counter), or Withings (internet-enabled bathroom scales), already have many users. This information can be the foundation for personal goal assistants, software applications that guide people towards reaching a target they have set for themselves by making better decisions based on personal data [1].

These goal assistant applications are replacing the traditional “self-help” books by software. This is part of a larger trend sometimes called quantified self or personal informatics, in which people collect data about themselves and their activities [2], [3]. Common goals are concerned with health, wealth and education, but a wider variety of domains are possible, such as time-management or human relationships.

Goal assistants can combine information, sometimes about different measurements for a given user; for example, a goal of improved physical condition might combine information about you from your bathroom scales, your heart rate monitor, and your on-line meal planner. Assistants can also combine measurements across many users to create cohort models, provided users agree to share some of the data about themselves. A user can make sense of their own data and its implications when they compare it with data of others with similar characteristics (education, age, etc); it is known that people’s behaviour is heavily influenced by normative models and social comparison [4]. For example, competitive attitudes can help someone to their goal of walking more, if they see how their step count compares to that of a similar population. Sophisticated models can be built within assistants, using a wide diversity of techniques from logic, statistics or machine learning; these models can provide a way to estimate some un-measured feature of the user, or to describe a cohort of users, or to represent relationships between features. Furthermore, one model may be based on information in multiple other models, creating deep and broad chains of influence or dependency. This is illustrated in Figure 1.

Storage of the logical data we describe above is actually spread among many different data stores. Typically, a sensor device is provided by a vendor that seeks to control (and probably monetise) the data on many users, that gets collected by instances of the device type. Web-based interfaces are usually available for users to see what is being reported, and also mobile app are provided that access the data through a service-based API. For example, data from a variety of Fitbit devices worn by many customers will be collected, and combined into cohort models, by the Fitbit Inc vendor company, and users see reports on their progress and how it
compares to cohorts, accessed in diverse formats. Similarly, data collected by an application will be combined, over many users, in a store held by the vendor of the app. These different data sets can be considered as silos, but they also will often exchange information; for example, vendors of complementary goal assistants or devices will form alliances to build more sophisticated applications from combined information. Thus, the different datasets can be considered as forming a dataspace [5], [6], where data is spread among platforms, and these platforms co-exist, with data integration taking place "pay-as-you-go", rather than in a schema designed before data is loaded.

A further complexity in the distributed management of this information arises from the widespread expectation that a person has some rights in the information about themselves. International data privacy legislation reflects users’ wishes for access to and control over their own data [7], [8], [9], [10], [11], [12]. One way that people seek a measure of control is to form a personal model, where they bring together information about themselves, within a software platform for which they are an authority. The Personis platform [13], [14], [15] embodies this architectural approach.

Thus, personal goal assistants are forming into an ecosystem where synergy occurs between assistants for specific goals, together with applications managing data for individual users. These components are administrated by different companies, agencies, or humans that both compete and cooperate, under varying legal constraints on information usage and disclosure. This environment is the context for this vision paper.

II. THE CHALLENGE: USERS CONTROL OF DATA FLOW

While personal assistant software is spreading, a major barrier to adoption lies ahead: users currently have little control of how and where data about them is used; they often feel unsure that they understand what can be done with their data.

An over-riding concern in personalized systems (such as personal assistants) is “scrutability” [16], [17], so the user can understand how the system arrived at the information the user sees. For example, with data relevant to educational goals, the Asilomar convention says “sharing practices must be made transparent to learners” (http://asilomar-highered.info/asilomar-convention-20140612.pdf). People need to know how data about them is used, and how to prevent uses they find inappropriate. One aspect of this is privacy, where the user wants their information hidden from certain other users. A significant minority of people have high levels of concern about privacy [18], [8]. There is particular concern about data from unobtrusive pervasive sensors [19], [20], [21] and mobile phones [22]. People appear to have become increasingly concerned about their privacy being compromised in personalised systems [23], [24]. Community attitudes to aggregation of data across personal goal systems has been explored [25], revealing a tension between risks and potential benefits. There are two broad approaches for privacy-friendly systems [26]. “Privacy-by-architecture” has a focus on anonymisation and client-side processing. This lacks the flexibility needed for people to be able to have different policies for the management of different personal data. The alternative, “privacy-by-policy” has a focus on supporting notice and choice principles, which calls for human-in-the loop systems. Recent research attempts to enforce privacy and usage policies, within a single organisation’s data collection [27], [28].

Another aspect of user control is the need for some data to be retracted that is, forgotten or removed from the system [29]. The reason for retraction could be that the original data was erroneous, but users can also desire to remove traces of correct data (for example, if they are embarrassed by some facts, or simply because enough time has passed) [30]. Ways of licensing for creative works suggests further types of control over the use of data: people may wish particular data to be unavailable for commercial purposes but shared by non-commercial programs, or they may only allow their data to enter models to which the user themselves has access, or they may be willing to allow access but require reporting of the uses back to the source.

Unless people can feel reassured that they know how data about them is used, and that they can prevent uses they find inappropriate, there will be more and more user disquiet about potential misuse, similar to the concerns that have been raised in the past about Google and Facebook, but intensified as the data involved comes closer to intensely personal matters of health, education and money. Providing this assurance to users requires innovations in human-computer interaction (to help users understand what uses have occurred, and express their desired restrictions on fair use). This contrasts with traditional data management research, that has concentrated on technology concerns such as query performance, or on mathematical foundations such as query expressivity. However, some leading scholars have identified the usability of data management systems as a core challenge. For example, in a keynote at SIGMOD’07 [31], H. Jagadish said: We assert that the usability of a database is as important as its capability.

Of course, as well as an intuitive and compelling interface for users to interact with information about information flow and usage, we also need platform support, so that collection of information about the flow, and control over the flow, can be achieved efficiently and without extraneous limitations on what (nearly) autonomous applications can do. Mechanisms in the dispersed data management infrastructure must track the effects of a piece of data. We here assume that legal or social imperatives will cause parties to adhere to any commitments they make; that is, a site will not lie about where data has been used. As well as direct copying, or relational computation (such as a join), we need to trace the flow of information in more subtle ways such as from a data item to its influence of the parameter values of a sophisticated machine learning model, and then indirectly on the predictions made by that model. Tracking needs to work across disparate and autonomous system components; similarly, restrictions on usage need to be conveyed to the various components through some agreed pattern of message exchange.

The combination of user interface, subsystem internals, and distributed protocol design, makes our vision a challenge.

A strawman design (similar to one for fine-grained provenance in databases [32]) would record, on every item, and every coefficient in every model, what other pieces of information were used to compute it; then (by taking the transitive closure of this"influences" relationship) a user could be told where their data has been used, and they could choose to allow or
disallow each kind of potential use. This is of course infeasible in our setting; three key flaws are (i) because the inference chains are so broad and deep, any item will influence so many others, that a user would suffer cognitive overload in trying to understand or control what is happening with their data; (ii) the performance degradation from tracking so many dependencies would be prohibitive; and (iii) it fails to separate a case where one item is used in an important way that matters to users, from situations such as a data item that is one among thousands that contributes to an average value (so the user can consider this "influence" as minor or neglectable). Our vision in contrast is for efficient tracking of the important information flows, and provide users with a manageable view of the flows.

As initial ideas that could help achieve this, we suggest basing a manageable user interface around a simple model [33] of human memory, where some events are considered in detail, others only in summarised form, and others are forgotten. By defining these different classes of data items (classified based on their source, their age, or their values), detailed item-by-item usage decisions and tracking may be limited only to a small amount of the most recent information; managing the flows for the other classes can be done in an aggregated way. We also propose adapting the distributed computing technique of summary meta-data called vector clocks [34] to replace long lists of detailed influence tracking (called causal dependency); furthermore, in systems with wide and deep branching, the important influences can be approximated in practice when users provide appropriate guidance as discussed in [35].

III. RELATED WORK

a) Database provenance: The importance of provenance or lineage information in databases was the focus of the TRIO system [36]. Much focus has been on not merely tracking impact, but on allowing forward and/or backward adjustments; for example, seeing which inputs might need checking, given a clearly inaccurate output value. An important distinction is made between coarse-grained provenance, which concentrates on connections between one data set and another, and fine-grained, which tracks from individual values (or records) in a dataset to particular values in the output. Theory behind provenance management was developed [32], [37]. Provenance within workflows (especially for manipulating scientific data) has been extensively explored [38], [39]. A recent focus is privacy for provenance information, especially structural privacy, that is, hiding some cases of the fact that information has flowed from one data set to another [40], [41]. This field is surveyed in [42].

b) Database access control: A theme in database research concerns access control and privacy [43]. Current practice falls into the tradition of discretionary control, following the principles set in System R [44], with permission on tables (and sometimes columns), based on the action being executed (SELECT, INSERT, etc). Research has studied how the data owner can control exposure or corruption of data, or miscalculation of results, when data is outsourced, that is, stored in (and operated on by) a separate entity (such as a cloud service provider) [45], [46]. Protecting user’s data in a dbms is a focus of the the “Hippocratic” dbms [47]. A foundation for distinguishing important exposure of information from acceptable use in aggregates is “differential privacy” [48].

c) Particular platforms: Adding tags to data items can track how information flows through a program, a database, a web application or an operating system [49], [50], [51], [52]. Tracking and visualising large volumes of provenance as information flows among cloud-hosted applications has been studied [53], [54], [55].

d) Usable security: An exciting subfield connects HCI and security; seminal work looks empirically at how users respond to attacks such as phishing [56], or to defenses like web security labels [21], privacy warnings [57] or password policies [58]. This poses serious usability challenges, as was found with P3P in relation to web privacy policies [21]. Recent work provides mechanisms for allowing users to express access-control intentions within their existing interactions [59].

REFERENCES
