Privacy Languages: Are we there yet to enable user controls?

Jun Zhao
Dept. of Computer Science
University of Oxford
Oxford, United Kingdom
jun.zhao@cs.ox.ac.uk

Reuben Binns
Dept. of Computer Science
University of Oxford
Oxford, United Kingdom
reuben.binns@cs.ox.ac.uk

Max Van Kleek
Dept. of Computer Science
University of Oxford
Oxford, United Kingdom
max.van.kleek@cs.ox.ac.uk

Nigel Shadbolt
Dept. of Computer Science
University of Oxford
Oxford, United Kingdom
nigel.shadbolt@cs.ox.ac.uk

ABSTRACT

Privacy protection is one of the most prominent concerns for web users. Despite numerous efforts, users remain powerless in controlling how their personal information should be used and by whom, and find limited options to actually opt-out of dominating service providers, who often process users information with limited transparency or respect for their privacy preferences. Privacy languages are designed to express the privacy-related preferences of users and the practices of organisations, in order to establish a privacy-preserved data handling protocol. However, in practice there has been limited adoption of these languages, by either users or data controllers. This survey paper attempts to understand the strengths and limitations of existing policy languages, focusing on their capacity of enabling users to express their privacy preferences. Our preliminary results show a lack of focus on normal web users, in both language design and their tooling design. This systematic survey lays the ground work for future privacy protection designs that aim to be centred around web users for empowering their control of data privacy.

Keywords

Personal data; Privacy; Survey; Social Machines

1. INTRODUCTION

Today most Internet and mobile device users are enjoying the convenience of being able to quickly navigate to a restaurant nearby, or find a web page most suitable to their unspoken preferences, without fully recognising the risk of their data privacy. Given the rise of social networking on the Web and ubiquitous sensor tracking, the range of personal information that is accessible to service providers are growing beyond a normal Web user's full comprehension [4]. Even the data that are explicitly shared by the users, with their consent and full awareness, are being reused beyond their original intention. Users still find unexpected advertisements in their mailboxes which may indicate that their privacy preferences have not been fully respected [15] or that their data have been made accessible to third-parties [36]

Users are leaving digital footprints everywhere on the Internet, when they shop, exercise, or travel. Service providers justify their access to these personal information by the provision of personalised services, which are often welcomed by normal web users, for higher quality of services [40]. However, users are losing their battle to control who should have access to their supposedly invisible digital footprints. They have limited means to stop their daily exercise routines from being put together with their shopping habits, which can then be used to produce a personalised picture of their lifestyle and shared with insurance companies or potential employers that the users never intended for.

These paradigmatic examples of privacy concerns increasingly arise in the context of human-machine collaboration systems or so-called 'social machines', such as Wikipedia and citizen science projects [42]. Such systems invariably depend on user participation and exchange of personal data in various forms. Ensuring some degree of coherence with user preferences regarding their personal data is therefore key for the long-term success of social machines.

A natural response to this problem might be to reach for state-of-the-art access control systems. However, despite their success in Web-based systems, they have huge limitations in the open Web, where information is copied and aggregated without a centralised control [44]. Furthermore, a lot of the time, personal information are being exploited not due to lack of access control, but rather a weakness in ensuring a contractual agreement with the service providers, in terms of how an individual's data should be respectfully accessed and reused, within users' consent.

Over the years, researchers have sought practical solutions to this lack of users' control through various privacy enhancement technologies or establishment of accountability [44]. Instead of enforcing restrictive access controls, the

WWW'16 Companion, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4144-8/16/04.
http://dx.doi.org/10.1145/2872518.2890590 .

goal of these approaches has been raising users' awareness of privacy issues (such as building privacy-aware search engines [16] or generating privacy icons [25]), or helping them trace who should be accountable in case something goes wrong [44]. Although both approaches have achieved some success in enabling users to gain control, the actual mechanism for users to express their wishes has not been the focus of existing studies.

A *privacy preference* language is a first-step answer to this call for user controls. Privacy policies are widely used by service providers, to express privacy policies of the organisations so that they can be more amendable to legal enforcements. A privacy preference language provides the expressions for the users to express their privacy preference over their own personal data. For example, objection to data processing for professional purposes, or requesting data deletion after a period of time. In this way, users gain an ability to declare how they expect their personal data to be used. For the *users*, this clear declaration on the terms of use could assist personal data to be processed more responsibly. And for *data controllers*, they could respect this data-terms-of-use, and only use and process personal data according to the purposes and terms declared by the users.

In the past two decades a number of privacy languages have been proposed by scholars [31, 29, 13, 28]. In this survey we focus on analysing how suitable these existing languages are for enabling users to express their expectations of their personal data, i.e how their data should be used, by whom, for what purposes, etc. The primary goal of our survey is to identify gaps in existing policy languages for enabling users to gain controls over the privacy of their personal data. To our knowledge this is the first survey that focuses on analysing how well existing policy languages can enable users to express their privacy preferences. We present our preliminary results and discuss our design suggestions for future languages.

2. METHODOLOGY

We identified existing languages from several existing survey studies [31, 29, 13, 28]. The languages included in the survey should meet the following criteria: 1) the language should focus on enabling either an organisation or individual to express their privacy policies or preferences, rather than only an access control mechanism; and 2) the languages can express policies in a machine-readable format. Access control policies are related to data privacy issues, but they are more likely to be defined and implemented by the service providers, based on their internal business logic or security requirements, rather than reflecting users' needs for controlling their data usage. However, we do not exclude languages that support access control as part of their features.

18 languages (see Appendix) were identified from existing literature mentioned above. All languages can represent policies in a machine-readable format. We eliminated those languages that were designed for expressing access control only or for expressing an organisation's internal policies, which are too fine-grained for expressing web privacy [31]. We ended up with 10 candidate languages for our review. Background information about each candidate language can be found in the appendix. We focus on examining the following features of each language:

- *Purpose of the language,* what the language has been designed or developed for (for example, capturing privacy commitment of an organisation rather than enabling users to express their preferences). This attribute has been used in a previous survey [31] and it is key to helping an adopter choose a policy language.
- *Tooling support*: what privacy tools have been implemented in which the language can be used by users to express their privacy preference, to validate their expression of preferences, or more generally to gain more control on their data usage. We focus on user-facing privacy tools, rather than applications that demonstrate a proof-of-concept.
- *Interoperability*: The production and collection of personal information and digital footprints online is largely decentralised, and can be anywhere, at any time. Exchange of privacy preferences and policies can take place in a Peer-to-Peer context (e.g. a user declaring his/her privacy preference when sharing information with a potential recruiter), a Business-to-Consumer context (e.g. an organisation processing a user's personal data), or a Business-to-Business context. Interoperability between various languages is therefore essential to avoid creating privacy silos.

3. PRELIMINARY RESULTS

3.1 Expression of Privacy Preferences

Our first observation (see Table 1) is that we see a stronger emphasis on supporting the preference expression of data controllers, i.e what information they would collect from an individual and what they would do with it. Those languages that do support users' expressions, they can be either too simplistic (such as APPEL) or too complicated (such as PPL) to be used by normal users to express a common preference, such as "do not use my data for recruitment purposes".

Preference languages (such as APPEL or XPref) are built as an extension to an existing policy language (such as P3P). It is a good practice to keep the preference language compatible with the policy language used by the organisations, so that they can be used as a communication and negotiation tool between the data owner and data consumer, to establish a privacy-preserving data handling protocol [41].

Future policy languages should bear this compatibility in mind and consider the privacy specification requirements from both the organisations and individuals. The preference vocabulary should be simple and easy-to-adopt by the users, the policy vocabulary should be sufficiently complete and extensible for organisations to express their data practices, and the two perspectives should be inter-changeable. Achieving the right balance is a challenging task.

Since the initial review of 2007 we have seen a more balanced development in supporting the expression of users' preference in privacy languages. However, these languages have not been accompanied by thorough considerations of user-facing tool developments and this may partly explain the lack of adoption to date.

3.2 Tooling support

Our second observation is that existing policy language efforts have largely not focused on designing an easy-to-use

Table 1: The purpose of the languages.

Policy language	First published	Desing purposes
P3P [19]	1997	Privacy policy language
APPEL (P3P) [18]	1997	Privacy preference language
Rei [32]	2002	Privacy policy language
XPref [3]	2003	Privacy preference language
AIR [27]	2008	Accountability policy language
PPL [43]	2009	Privacy policy and preference language
SecPAL4P [12]	2009	Privacy preference language
Jeeves [45]	2012	Privacy policy language
A-PPL [8]	2013	Accountability policy language
P2U [26]	2014	Privacy policy language

user-facing tools as part of their approaches (see Table 3). Furthermore, very limited systematic usability studies have been performed to understand what are needed to enable users to adopt these technologies [20].

Independent of policy language development we have seen some other endeavours to improve this situation through the creation of intuitive and usable privacy summaries for the users. For example privacy icons created by PrivacyBird [22], KnowPrivacy [24], Mozilla [1], and others produce an intuitive and simplified summary of a service provider's privacy policy, so that users can quickly choose between alternative service providers based on their privacy implications. However, although there have been some positive feedback from the users through their user studies, these approaches have received limited uptakes in general, due to the following reasons [20]:

- limited adoption of standardised privacy policies by organisations or data controllers remain a barrier to making these icons as meaningful or useful as they could have been;
- understanding the privacy dimensions that are most meaningful and relevant to users' interest remains a huge challenge in building these user-facing tools; and finally
- bogus policy declarations, even by leading service providers, seriously reduce the effectiveness of those programmes processing these policies, and hence their adoption by the users [33].

3.3 Interoperability

Although all the reviewed languages were designed as a web privacy language, i.e. assuming an open web as the default platform, they are hardly interoperable with each other. This is reflected at both the language representation level and at the actual tool implementation level. While the representations of reviewed languages are predominately in XML or RDF format, there are also languages grounded upon logic (such as SecPAL4P) or programming languages (such as Jeeves). The majority of the implementations are stand-alone prototypes, as a proof-of-concept, with the exception of PrivacyBird (grounded upon P3P) which is a plug-in for several leading web browsers, and Jeeves, which are available as platform-independent libraries for developers to build privacy-preserving applications.

[1]https://wiki.mozilla.org/Privacy_Icons

P3P has been a partial exception amongst them as it was published as a W3C recommendation in 2002. However, it has arguably failed, given the closure of the W3C P3P Working Group [2] and analysis of its actual uptakes by organisations [21, 10]. Another more recent effort in facilitating an interoperable approach towards privacy enhancement is the W3C Tracking Protection Working Group [3]. As stated in their charter, the Working Group was set out to "improve user privacy and user control by defining mechanisms for expressing user preferences around Web tracking and for blocking or allowing Web tracking elements" [1]. By web tracking, the group largely refer to "the collection of data regarding a particular user's activity across multiple distinct contexts and the retention, use, or sharing of data derived from that activity outside the context in which it occurred" [23], although the exact definition of tracking proved a point of contention during the development of the standard. The working group have published their last call in August 2015, which has received very mixed feedback from various industrial players, ranging from leading authorities to smaller-sized organisations. The main criticisms include its lack of mechanisms of enforcement and its possibility of introducing unfair business competitiveness. Without a global technical agreement, any new privacy-enhancement proposals must look harder at how to work closely with existing approaches, particularly existing web technologies and architectural designs, and think harder about how to conceive a design most in line with users' incentives and existing business processes.

4. DESIGN IMPLICATIONS

Today users are taking the centre stage in privacy research. There have been numerous studies on understanding users' attitude of online privacy [17, 34], in order to provide a solution that is most fitted for users' needs. However, our review has clearly revealed gaps in existing policy language research, in terms of providing languages and tools that can empower users to gain control of their personal data.

Understand your user group.

Earlier efforts like APPEL, XPref, or PPL have considered extensively users' requirements for expressing privacy preference at the time of design, and provided vocabularies to cover key aspects including purposes, obligations, and

[2]https://www.w3.org/P3P/; Accessed in January 2016

[3]https://www.w3.org/2011/tracking-protection/; Accessed in January 2016

data retentions. However, these languages ended up as being either too complicated or being tied up with specific policy language (like APPEL). Following-up studies [20] have shown that it is impractical to expect extensive inputs from users for setting up their preferences, creating an avoidable barrier for adoptions. Literature on users' privacy attitudes could provide valuable inputs to future language designs, to finesse the balance between language expressiveness and the practicality of gathering the required information.

Design for machine interpretation and prediction.

Some of the sophisticated structures in existing languages may not be fit for direct usage by data owners, but more appropriate for representing the privacy preferences of a privacy persona [37], that could be inferred or computed by computer programmes [39], based on their past privacy behaviours. This approach of predicting personalised privacy preference provides a promising alternative to existing approaches, where a heavy investment or buying-in from users is a prerequisite. However, language for this type of applications may have a higher requirement for logical completeness and verifiability, as well as interoperability, which may take a higher priority in their designs than those use-facing languages.

Design for decentralised use and consumption.

An implicit but crucial indication from our survey is the importance of considering the decentralised nature of the web. Most of the existing languages that we review have this assumption that data controllers are the central platforms for processing and reusing personal data. However, in a decentralised setting, the dissemination of personal information is becoming more ubiquitous. The emergency of personal data clouds and other tools potentially provide individuals more power to negotiate with other entities on an equal footing. This possible change of power relationship is crucial for designing future privacy protection, and calls for stronger support for enabling user controls, in order to achieve the critical mass that is core to privacy protection.

5. CONCLUSIONS AND FUTURE WORK

In this survey paper we evaluate ten policy languages developed over the last two decade, to investigate their support for expressing users' privacy preferences, accompanying tool development, and interoperability with each other. This is the first survey that examine these languages from the perspective of their fitness for enabling end-user controls.

Users' privacy preferences are known to be diverse and complex. Our results show that existing languages are either too complicated for normal web users or simplistic to cope with the diverse requirements. Their tool developments have largely failed to deliver a user-friendly interface or an seamless integration with existing business processes to promote uptakes by the users. Efforts like privacy icons show promising results. However, the lack of machine-processable privacy policies restricts their usefulness.

Another observation from our survey shows that existing developments have a stronger focus on the data controllers, as shown by the elaborate policy languages as well as focus on usable policy notifications (such as the icons). These approaches indeed help with leveraging the existing privacy issues. However, the control required by data owners has the danger to be treated as by-products, leaving users being continuously overpowered by whatever privacy controls committed by the data controllers [21, 10, 20]. User controls must take a more central role in privacy research, given that today users are presented with more opportunities of managing their own data. Users must be equipped with a stronger self-control mechanism in order to retain their fundamental rights to their personal information.

Current research on understanding users' privacy attitudes [17, 34] provides valuable inputs for us to understand the type of data that users most care about. Understanding privacy preference modelling is core to our future work of establishing a more transparent and accountable digital space for the users, so that users can gain more control by tracking how things have gone wrong from what they have preferred.

6. ACKNOWLEDGEMENT

This work is supported under SOCIAM: The Theory and Practice of Social Machines. The SOCIAM Project is funded by the UK Engineering and Physical Sciences Research Council (EPSRC) under grant number EP/J017728/2 and comprises the Universities of Oxford, Southampton, and Edinburgh.

7. REFERENCES

[1] Tracking protection working group charter. Technical report, Access on 16 January 2016.

[2] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Hippocratic databases. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 143–154. VLDB Endowment, 2002.

[3] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. An xpath-based preference language for p3p. In *Proceedings of the 12th international conference on World Wide Web*, pages 629–639. ACM, 2003.

[4] J. Angwin. Online tracking ramps up – popularity of user-tailored advertising fuels data gathering on browsing habits. *Wall Street Journal*, 2012.

[5] C. A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati. A privacy-aware access control system. *Journal of Computer Security*, 16(4):369–397, 2008.

[6] P. Ashley, S. Hada, G. Karjoth, C. Powers, and M. Schunter. Enterprise privacy authorization language (epal 1.2). *Submission to W3C*, 2003. Accessed December 2015.

[7] P. Ashley, S. Hada, G. Karjoth, and M. Schunter. E-p3p privacy policies and privacy authorization. In *Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, pages 103–109. ACM, 2002.

[8] M. Azraoui, K. Elkhiyaoui, M. Önen, K. Bernsmed, A. S. De Oliveira, and J. Sendor. A-ppl: An accountability policy language. In *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance*, pages 319–326. Springer, 2015.

[9] A. Barth, J. C. Mitchell, and J. Rosenstein. Conflict and combination in privacy policy languages. In *Proceedings of the 2004 ACM workshop on Privacy in the electronic society*, pages 45–46. ACM, 2004.

[10] P. Beatty, I. Reay, S. Dick, and J. Miller. P3p adoption on e-commerce web sites: a survey and analysis. *Internet Computing, IEEE*, 11(2):65–71, 2007.

[11] M. Y. Becker, C. Fournet, and A. D. Gordon. Design and semantics of a decentralized authorization language. In *20th IEEE Computer Security Foundations Symposium (CSF)*, pages 3–15, 2007.

[12] M. Y. Becker, A. Malkis, and L. Bussard. A framework for privacy preferences and data-handling policies. Technical report, Microsoft Research Cambridge Technical Report, MSR-TR-2009-128, 2009.

[13] F. Bélanger and R. E. Crossler. Privacy in the digital age: a review of information privacy research in information systems. *MIS quarterly*, 35(4):1017–1042, 2011.

[14] K. Bohrer and B. Holland. Customer profile exchange (cpexchange) specification. 2000. Accessed December 2015.

[15] M. Borghi, F. Ferretti, and S. Karapapa. Online data processing consent under eu law: a theoretical framework and empirical evidence from the uk. *International Journal of Law and Information Technology*, 21(2):109–153, 2013.

[16] S. Byers, L. F. Cranor, D. Kormann, and P. McDaniel. Searching for privacy: Design and implementation of a p3p-enabled search engine. In *Privacy Enhancing Technologies*, pages 314–328. Springer, 2005.

[17] F. Chanchary and S. Chiasson. User perceptions of sharing, advertising, and tracking. In *Proceedings of the Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 53–67, Ottawa, July 2015. USENIX Association.

[18] L. Cranor, M. Langheinrich, and M. Marchiori. A p3p preference exchange language 1.0 (appel 1.0): W3c working draft 15 april 2002. *World Wide Web Consortium (W3C), URL: http://www. w3. org/TR/P3P-preferences*, 2002. Accessed December 2015.

[19] L. Cranor, M. Langheinrich, M. Marchiori, M. Presler-Marshall, and J. Reagle. The platform for privacy preferences 1.0 (p3p1. 0) specification. *W3C recommendation 16 April 2002*, 2002. Accessed December 2015.

[20] L. F. Cranor. Necessary but not sufficient: Standardized mechanisms for privacy notice and choice. *J. on Telecomm. & High Tech. L.*, 10:273, 2012.

[21] L. F. Cranor, M. Arjula, and P. Guduru. Use of a p3p user agent by early adopters. In *Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, pages 1–10. ACM, 2002.

[22] L. F. Cranor, P. Guduru, and M. Arjula. User interfaces for privacy agents. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 13(2):135–178, 2006.

[23] N. Doty, H. West, J. Brookman, S. Harvey, and E. Newland. Tracking compliance and scope w3c last call working draft 14 july 2015. Technical report. Accessed in January 2016.

[24] J. Gomez, T. Pinnick, and A. Soltani. Knowprivacy. Technical report, 2009.

[25] L.-E. Holtz, H. Zwingelberg, and M. Hansen. Privacy policy icons. In *Privacy and Identity Management for Life*, pages 279–285. Springer, 2011.

[26] J. Iyilade and J. Vassileva. P2u: A privacy policy specification language for secondary data sharing and usage. In *IEEE Security and Privacy Workshops (SPW)*, pages 18–22. IEEE, 2014.

[27] L. Kagal, C. Hanson, and D. Weitzner. Using dependency tracking to provide explanations for policy management. In *9th IEEE International Workshop on Policies for Distributed Systems and Networks. POLICY 2008*, pages 54–61. IEEE, 2008.

[28] S. Kasem-Madani and M. Meier. Security and privacy policy languages: A survey, categorization and gap identification. *arXiv preprint arXiv:1512.00201*, 2015.

[29] J. Kolter. *User-Centric Privacy ? A Usable and Provider-Independent Privacy Infrastructure (Chap 4)*. PhD thesis, University of Regensburg, 2009. https://www.ics.uci.edu/ kobsa/phds/kolter.pdf.

[30] U. König. Primelife checkout. In *Privacy and Identity Management for Life - PrimeLife International Summer School, Helsingborg, Sweden, August 2-6, 2010*, pages 325–337. Springer, 2011.

[31] P. Kumaraguru, L. Cranor, J. Lobo, and S. Calo. A survey of privacy policy languages. In *Workshop on Usable IT Security Management (USM 07): Proceedings of the 3rd Symposium on Usable Privacy and Security*. ACM, 2007.

[32] K. Lalana. Rei: A policy language for the me-centric project. *TechReport, HP Labs*, 2002. Accessed December 2015.

[33] P. G. Leon, L. F. Cranor, A. M. McDonald, and R. McGuire. Token attempt: the misrepresentation of website privacy policies through the misuse of p3p compact policy tokens. In *Proceedings of the 9th annual ACM workshop on Privacy in the electronic society*, pages 93–104. ACM, 2010.

[34] P. G. Leon, B. Ur, Y. Wang, M. Sleeper, R. Balebako, R. Shay, L. Bauer, M. Christodorescu, and L. F. Cranor. What matters to users?: factors that affect users' willingness to share information with online advertisers. In *Proceedings of the Ninth Symposium on Usable Privacy and Security (SOUPS 2013)*, page 7. ACM, 2013.

[35] A. Matheus and J. Herrmann. Geospatial extensible access control markup language (geoxacml). *Open Geospatial Consortium Inc. OGC*, 2008.

[36] J. R. Mayer and J. C. Mitchell. Third-party web tracking: Policy and technology. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 413–427. IEEE, 2012.

[37] S. Preibusch. Managing diversity in privacy preferences: How to construct a privacy typology. In *Workshop on Privacy Personas and Segmentation, co-located at the 10th Symposium On Usable Privacy and Security (SOUPS)*, 2014.

[38] E. Rissanen. extensible access control markup language (xacml) version 2.0. *Oasis*, 2013. Accessed December 2015.

[39] J. L. B. L. N. Sadeh and J. I. Hong. Modeling users? mobile app privacy preferences: Restoring usability in

a sea of permission settings. In *Symposium on Usable Privacy and Security (SOUPS)*, 2014.

[40] S. Senecal and J. Nantel. The influence of online product recommendations on consumers? online choices. *Journal of retailing*, 80(2):159–169, 2004.

[41] O. Seneviratne and L. Kagal. Httpa: Accountable http. In *IAB/w3C Internet Privacy Workshop*, 2010.

[42] N. Shadbolt, M. Van Kleek, and R. Binns. The rise of social machines. *Consumer Electronics Magazine, IEEE*, 5(1), 2016.

[43] S. Trabelsi, J. Sendor, and S. Reinicke. Ppl: Primelife privacy policy engine. In *IEEE International Symposium on Policies for Distributed Systems and Networks (POLICY)*, pages 184–185. IEEE, 2011.

[44] D. J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigenbaum, J. Hendler, and G. J. Sussman. Information accountability. *Communications of the ACM*, 51(6):82–87, 2008.

[45] J. Yang, K. Yessenov, and A. Solar-Lezama. A language for automatically enforcing privacy policies. In *ACM SIGPLAN Notices*, volume 47, pages 85–96. ACM, 2012.

APPENDIX

A. THE LANGUAGES

P3P.

P3P (Platform for Privacy Preferences) [19] is a XML-based language that enables a data owner to assess whether the privacy practices declared by a service provider comply with his/her privacy preferences. The privacy policy of the service provided can be defined in the P3P language, specifying the purpose of their data collection, consequences of data release, and data retention policy. To allow more automated privacy matching according to individuals' needs, users can explicitly specify their privacy preferences through APPEL [18], which is interoperable with P3P. A privacy agent can compare the privacy preferences of a user with the P3P policies of a service provider in order to achieve personalised privacy protection. P3P became a W3C recommendation in April 2002.

XPref.

XPref is a proposal aimed to fix some of issues of APPEL, for example, being unable to express what is acceptable, limited ability to express combination of preferences and lack of robustness [3]. However, both XPref and APPEL have been criticised for their expressiveness, for unable to express negotiations or attribute-based conditions [5].

PPL.

The PrimeLife Policy Language (PPL) [43] is a privacy language developed by the PrimeLife project[4]. It is designed as an extension to XACML, and therefore it can support authorisation based on compatible privacy preferences between a data owner and a data controller. It can express privacy preferences of a data owner or an organisation, in terms of purposes, obligations and authorisations. It is designed for a service-oriented architecture and therefore its design also considered downstream privacy control through a mechanism called "sticky policies", which can represent the obligations that a data controller has agreed to adhere to for a resource.

[4]http://www.primelife.eu

Rei.

Rei [32] is generic policy language that is grounded upon deontic concepts. The language is aimed to be application and domain-independent.It allows for the definition of actions, constraints, obligations, delegation and policy types. A policy type can be instantiated, and a policy and an action can be bound to a certain subject. Meta-policies contain priority and precedence information for policy interpretation and policy conflict resolving.

SecPAL4P.

SecPAL4P [12] is a language that extends SecPAL, which is mainly an access control language, for specifying the handling of personally identifiable information (PII). The language is designed for specifying both users' preferences on how their personally identifiable information should be treated by data collecting services, and services' policies on treating collected personal information. SecPAL4P was set out to fix the limited expressiveness and scope of existing language, in order to more sufficiently express policies represented in natural languages. It aims to be a generic policy language with sufficient abstraction to model across systems. Therefore it was not designed for direct user adoptions.

AIR.

AIR [27] (**A**ccountability **I**n **R**DF) is a generic rule-based policy language that are grounded upon Semantic Web technologies. Generally speaking an AIR policy can define the set of rules that are applicable to a set of variables within the scope of the policy. For a dataset and a set of AIR policies in Turtle format, the AIR reasoner can compute the compliance of data with respect to these AIR rules, and produce explanations for the compliance, also in Turtle format. The advantage of the AIR approach is that it is able to produce justifications for a policy decision in a dependency tree and it is fairly domain-neutral. There are no user-facing tools for creating AIR policies, but there is a justification UI showing proof trees.

Jeeves.

Jeeves [45] is a language which uses a programming model to make the system responsible for policy compliance. The approach separates non-policy related program functionality from a set of declarative policies explicitly associated with sensitive data. An advantage of this approach is that programmers can express policies explicitly rather than implicitly across the whole codebase. This approach does aim to encompass end-user requirements, and cites social network location privacy preferences as a motivating case study. However, the focus is again on the expressive capacity and functionality of language itself, and improving ease of implementation for programmers, rather than on user-facing tools for expressing simple preferences.

A-PPL.

A-PPL [8] is an Accountability Policy Language which builds on PPL, adding functionality for performing audits

Table 2: Policy languages considered for the review

Policy language	First published	Situation	Included
P3P [19]	1997	Web privacy policy language	Yes
APPEL (P3P) [18]	1997	Individual's privacy preference	Yes
CPExchange [14]	2000	Internal enterprise policy lanugage	No
PRML []	2001	Internal enterprise policy lanugage	No
E-P3P [7]	2002	Internal enterprise policy lanugage	No
Rei [32]	2002	Policy language	Yes
XACML [38]	2003	Access control language	No
EPAL [6]	2003	Internal enterprise policy lanugage	No
XPref [3]	2003	Individual's privacy preference	Yes
DPAL [9]	2004	Internal enterprise policy lanugage	No
GeoXACML [35]	2005	Access control language	No
SecPAL [11]	2006	An authorisation language	No
AIR [27]	2009	Accountability policy language	Yes
PPL [43]	2008	Privacy policy and preference language	Yes
SecPAL4P [12]	2009	Privacy preference language	Yes
Jeeves [45]	2012	Privacy policy language	Yes
A-PPL [8]	2013	Accountability policy language	Yes
P2U [26]	2014	Privacy policy language	Yes

to verify compliance with policies, regulations or user preferences. It is focused on providing accountability between cloud computing customers and providers regarding personal data, where the customer is a data controller with responsibility for end-user data, and needs to hold the cloud provider accountable for compliance with their privacy policy. It is therefore aimed primarily at a business-to-business context rather than at ensuring end-user control. Compliance with user preferences is cited as a potential application of A-PPL, but as with other languages surveyed, this is not reflected in the proof-of-concept nor supported by user-facing tools.

P2U.

P2U [26] ('Purpose to use') is a language which aims to deal with the problem of 'secondary' data sharing. The authors allege that previous languages are focused primarily on primary data collection, i.e. where the data collector is in a direct relationship with the data subject (such as a web user and a website). A significant benefit of P2U from the perspective of this survey is that it considers use of data by multiple entities. It builds in several elements lacking in previous languages, such as the ability to consent to or deny unanticipated uses of data, and a price-negotiation element to allow data to be monetised. Despite allowing for user negotiation, the language is still designed around the assumption that policies will be initially defined by data controllers rather than data subjects.

Table 3: Tooling support for key policy languages.

Policy language	Policy tool	Functions	Shortcomings
P3P+APPEL	PrivacyBird [22]	An Internet Explorer plug-in that can automatically detect how much a web site's P3P privacy policy fits into your privacy preferences and creates a coloured bird icon according to the level of compatibility	System-specific, platform-specific, P3P-specific
Rei	Rei policy engine [32]	A simple Java wrapper that enables users to query actions that can be performed or obligations of an agent, based on Rei policies expressed in Prolog or RDF	Hardly end-user usable
XPref	Part of Hippocratic DBMS [2]	A strawman design of database systems that provide privacy support including privacy creation, validation, privacy-enhanced data control, auditing and etc.	Designed for a closed world setting and without user-facing support
PPL	PPL Engine [43]	A prototype implementation that is able to process PPL policies, perform policy matching, access control and obligation enforcement	This is a middleware application as a Java class and service APIs, not for end users
	PrimeLife Checkout [30]	A web application prototype that enables users to control access to their personal data in order to avoid their personal information being used for credit check without users' consent. In addition to privacy matching, the application can also show how their personal data may be transfers by different parities involved in the shopping process (such as shipping company or the shop)	The privacy setting user interface is a matrix of checkboxes and can be too comprehensive for some users, and the system prerequisites the availability of machine-processable PPL policies from the shops. And it remains a very preliminary prototype.
AIR	AIR Justification UI [27]	A plug-in to Tabulator to show a justification of a set of policies against data as a dependency tree as well as a structured textual explanation (http://dig.csail.mit.edu/TAMI/2008/JustificationUI/howto.html)	Policies have to be hand-crafted and there are known usability issues with the proof trees