# The Coprelobri project: the logical approach to privacy

Guillaume Aucher[1], Catherine Barreau-Saliou[2], Guido Boella[3], Annie Blandin[4], Sébastien Gambs[1], Guillaume Piolle[5], Leendert van der Torre[6]

[1]Université de Rennes 1 - INRIA/IRISA, Campus universitaire de Beaulieu, 263 Avenue du Général Leclerc - Bât 12, 35042 Rennes Cedex, France – guillaume.aucher@irisa.fr, sebastien.gambs@irisa.fr
[2]Université de Rennes 1 - CEDRE, 9 rue Jean Macé, CS54203, 35042 Rennes Cedex, France – catherine.barreau-saliou@univ-rennes1.fr
[3] Dipartimento di Informatica and Centro di scienza cognitiva, Universita' degli Studi di Torino, Corso Svizzera, 185 I-10149 Torino Italy – guido@di.unito.it
[4]Télécom Bretagne, Technopôle de Rennes Atalante, 2 rue de la Châtaigneraie, CS 17607, 35576 Cesson-Sévigné Cedex - annie.blandin@telecom-bretagne.eu
[5]Supélec, CS 47601, Avenue de la Boulaie, 35576 Cesson-Sévigné Cedex, France – guillaume.piolle@supelec.fr
[6] University of Luxembourg, Faculté des Sciences, de la Technologie et de la Communication, 6, rue Richard Coudenhove-Kalergi, L-1359 Luxembourg – leon.vandertorre@uni.lu

**Abstract.** The aim of the Coprelobri project (standing for "Computers and privacy regulations: the logical bridge") is to provide tools to design, verify and enforce privacy policies, using logic as a means to reach this goal. The backbone of the project is the design of a logical language that can be used to represent and reason on privacy policies. In a first stage, this language, primarily based on epistemic and deontic logics, will be used to model current privacy regulations. Afterwards, theorem proving and model checking techniques will then be adapted to this language to build practical software tools for regulation makers, corporate lawyers, computer engineers and end users. Given the aims and expectations of this project, it is of paramount importance to take a truly multidisciplinay approach in which computer scientists and legal experts participate jointly to this research.

*Keywords* : Privacy policies, logic, legal computing.

## 1 Context and objectives

The role of data protection policies in the general quest for privacy in the information society now seems to be accepted as a significant one. However, their widespread use by system designers, service providers and end users still remains a challenge. Moreover, the fact that individual users have very few means to actually verify that service providers actually comply with the regulations when processing their personal data [8] can be seen as a consequence of this situation. We are currently in the phase of starting the Coprelobri multidisciplinary project with a new perspective and methodology on this issue.

From a lawyer's point of view, the compulsory nature of privacy-related obligations is made obvious by the intended sanctions and their sometimes significant severity. However, the efficiency of the protection of the individuals with respect to their personal data remains very weak and difficult to enforce in practice. Indeed, the traditional response of the legal science in terms of regulation has often proved to be insufficient: reinforcement of controls, sanctions, actions in reparation, class action procedures... Therefore in this setting, there seems to be a need for a new kind of response specifically tailored to the domain of privacy law [2].

In the computer science world, existing privacy policy frameworks such as P3P [15], SPARCLE [7] or XACML [10] have failed so far to impose themselves as practical solutions. This is due either to their ambiguity (both for users and software) or their lack of expressiveness (thus keeping significant part of privacy regulations outside their scope). Another significant shortcoming of these methods is the purely declarative nature of this kind of policy language. To summarize, they are unable to check that the expressed privacy policy is actually enforced by a service, or that the policy is compliant with a given normative system, such as a national legislation or a contract. In the recent years, research on logic-based privacy policy languages, which aim at building policies associated with automated reasoning capabilities, have begun to emerge [11,13,1]. However, even though these formalisms try to stick better to the semantics of usual privacy regulations, they are still too abstract. For this reason, the link with the actual implemented services and the understanding by users, lawyers and computer scientists remains problematic and sometimes too weak. In this context, the generalization of confidentiality and privacy policies through the deployment of dedicated software tools seems to be a relevant approach.

The observation of the current state of affairs has lead us to the fundamental following questions that form the core of the research project that we are launching:

> How can we bridge the gap between the way privacy regulations are expressed in the legislation and the way privacy policies are specified by computer scientists, so that:
> - We can check automatically that the privacy policies declared by a company or institution are compliant with respect to the privacy regulations existing in a given legislation, and
> - We can check automatically that the company or institution does enforce its declared privacy policy?

On the basis of this question, we propose to use logic as a bridge and a *lingua franca* between law and computer science, a strategy already exploited in the general field of legal computing [5,9]. Starting from an analysis of existing privacy regulations and needs expressed by law researchers, we will develop a logical language that will be able to express appropriately and formally any kind of privacy regulation. We will then build several tools based on this language, targeted to lawyers, computer engineers and end users, such as policy writing assistants and verification software. We believe the logical approach to be particularly promising, as many notions that are central to privacy regulations such as obligation, interdiction, knowledge and time have been the subject of extensive logico-philosophical analysis, and numerous logical formalisms dealing with these notions have been developped. These formalisms give rise in turn to automated reasoning tools which can be used for this project.

A key aspect of this research is the joint and entwined work between computer scientists and law researchers. Indeed, the starting point of our work will be the development of an efficient procedure for modelling legal text and only a lawyer versed in this particular field can properly validate that. Furthermore, some of the tools we plan to build are dedicated to corporate lawyers, so the specification must be designed together with them. In the next section, we present the logical basis on which we plan to build our pivotal language, then we detail the characteristics of the software that we are planning to develop.

## 2 Formal language

In a recent communication [1], a new logical language has been proposed to deal with privacy policies. Its main characteristic and originality is the addition of dynamics to a joint use of both deontic and epistemic logic [14,6], previously proposed in earlier logics [3,4]. Using deontic logic to express concepts such as obligations, interdictions and permissions is a rather common approach in logic-based security policies. The epistemic dimension (focusing on knowledge), although more unusual, is particularly adapted to privacy policies and information flow control in general. For instance, it allows to reason on a final epistemic statement such as “who knows what”. The combination of these two classes of modalities enables the design of rules telling which agent is allowed to know which information. For example, the privacy policy whereby it is not $P$ermitted for an agent $a$ to $K$now at the same time both pieces of information $p$ and $q$ can be expressed by the following formula: $\neg PK_a(p \land q)$. The logic of [1] then adds dynamics to this language by means of an operator $[send\ \phi]$ which reads as “information $\phi$ is sent to the agent”. This addition of dynamics allows for example to infer in the logic the formula $K_a p \to \neg P[send\ q]$: “if agent $a$ already $K$nows $p$, then it is not $P$ermitted (according to the privacy policy) to send him/her information $q$”.

The proposed language is still in an early form, currently it allows to reason on a single agent and with few modalities. During the first phase of the Coprelobri project, we plan to enrich the language in order to make it expressive enough to represent real-life privacy regulations. More precisely, the design of the language will be obtained through an iterative collaboration task between the computer scientists and the law researchers of the project, who will validate also the work at the end. Among the constraints stemming from the nature of the legal texts, we can stress the following ones:

- It must be possible to express every obligation of the regulation systems. This is a complex issue, in particular it is not uncommon that an explicit legal obligation must be broken down into several implicit obligations.
- The resulting logical framework must be able to adapt itself to the legal, jurisprudential and practical evolutions of the initial obligations. Adressing this inherent dynamism necessitates that computer scientists develop a genuine sensitivity to the legal discipline.

Consequently, the expected developments include the following:

- The combination with a temporal logic, so that it is possible to express the deontico-temporal notions commonly used in privacy policies, such as deadlines or delays [12], and to cope with the above dynamicity requirements.
- Introduction of multi-agent reasoning capabilities.
- Introduction of multiple norm source reasoning capabilities, for the language to be able to deal with inconsistent or conflicting regulations.

As a validation step for the expressiveness of the language, we plan to translate actual regulations in the resulting logical language. Targets will be chosen among HIPAA/COPPA U.S. acts. Building this logical language should roughly represent one year and a half of work, which is half the project's lifetime.

## 3 Software tools for modelling and enforcing privacy policies

Once the language has been developed, it will then serve as the basis for several software developments. More precisely, the decidability and complexity results, as well as the identified decision procedures, will help us to devise model-checking and theorem-proving software tools. This very formal stage, during which we plan to use and adapt existing techniques to fit our needs, will be used in the design of several end-user software.

The first type of software that we plan to develop is targeted to lawyers in charge of making privacy policies, regulations and law. It is a writing assistant, guiding the user in phrasing sentences that both have a clear meaning in the legal language and can be translated automatically into our pivotal formal language. Ideally, this assistant would be bound with a specialized clause directory in order to provide further assistant to the regulation writer. The main idea behind this tool is to be able to create regulations that act at the same time, as consistent legal texts and as computationally usable sets of clauses (which will be fed to other software tools). This kind of tool can easily be seen as an intrusion in the preserved domain of specialized lawyers. Therefore, the functional specification stage requires special care and should involve law researchers as well as field lawyers (in particular corporate lawyers). The output must be a product useful for both policy makers and enforcers (at organizational and computational levels).

The second type software is targeted to policy makers and service providers. It consists in a verification tool, whose task is to check that a given policy is compliant with a set of high-level regulations, or that the formal model of an application or service is compliant with a set of policies and regulations. The output of this tool, which will be based on theorem proving techniques, should be rich enough to enable the user to modify the policy or the model in order to make it compliant (or as compliant as it can get). This kind of software should be a key element of the practical implementation of "privacy by design" principle, which encourages organizations and companies to integrate the privacy issues as early as the design phase of a product.

Finally, another possible type of software is directed to technical staff attached to service providers. Based on model checking and dynamic analysis techniques, it would constitute a verification and monitoring tool, providing insight about the compliance of actual executions of a software system.

The development of these end-user tools will occur during the second phase of the project. We also plan a real-life experimentation of the tools, with the cooperation of the management team of an E-university platform.

## 4 Conclusion

To summarize, besides a logical analysis of privacy regulations, the Coprelobri project will also provide tools to design, verify and enforce privacy policies. During the first phase of the project, based on an analysis of existing privacy regulations and with the help of the law researchers, we will develop a logical language that can be used to express privacy regulations. In the second part of Coprelobri, and once the logical language is defined, we will use it to develop several types of software tools. The first type of software is intended to be used by law/policy makers to write out their regulations. A key functionality of this software is to produce at the same time two versions of the *same* privacy regulations: one version expressed in natural language (and immediately visible by the users of this software on its visual interface), and one version expressed in the logical language defined in the first part of Coprelobri. This second version expressed in the logical language will then be used and processed by computer scientists. In particular, it will enable the use of standard automated reasoning tools to verify the two kinds of compliance described in our research question, and to monitor and restore compliance. These other functionalities will be built into a second and a third type of software tool (as described in the above section), and will resort to theorem provers and model checkers developped in an earlier phase of Coprelobri. We will finally validate this software by considering as case study an e-education website such as the platform of a digital university. The expected results of the Coprelobri project encompasses the following points:

- A facilitation of regulation compliance for organizations and companies with respect to confidentiality and data protection policies;
- A reinforcement of the trust towards service providers and online services requesting personal data disclosure;
- A decrease in privacy policy management costs;
- A simplification of the survey activities of data protection authorities.

# References

1. G. Aucher, G. Boella, and L. Van der Torre. Privacy policies with modal logic: the dynamic turn. In G. Governatori and G. Sartor, editors, *Deontic Logic in Computer Science*, volume 6181 of *LNCS*, pages 196–213. Springer, 2010.
2. A. Blandin. *Les technologies de linformation au service des droits : opportunits, dfis, limites*, chapitre "Quelles solutions techniques pour rsoudre les problmes juridiques poss par la technique ?" Cahiers du CRID. Bruylant, 2010.
3. F. Cuppens. A logical analysis of authorized and prohibited information flows. In *IEEE Symposium on Research in Security and Privacy*, pages 100–109, Oakland, California, USA, 1993. IEEE Computer Society Press.
4. F. Cuppens. A logical formalization of secrecy. In *Computer Security Foundations Workshop VI*, pages 53–62, Franconia, USA, 1993. IEEE Computer Society Press.
5. J.-P. Delville. *Introduction la thorie formelle du droit.* Phd thesis, Universit de Rennes I, Rennes, France, 1992.
6. J. Hintikka. *Knowledge and Belief.* Cornell University Press, 1962.
7. G. Karjoth and M. Schunter. A privacy policy model for enterprises. In *15th IEEE Computer Security Foundations Workshop*. IEEE, IEEE Computer Society Press, 2002.
8. Lancelot-Miltgen, Caroline. *Dévoilement de soi et réponses du consommateur face une sollicitation de ses donnes personnelles : une application aux formulaires sur Internet.* Universit Paris-Dauphine, 2006, PhD thesis.
9. N. Love and M. R. Genesereth. Computational law. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law (ICAIL'05)*, pages 205–209, New York, NY, USA, 2005. ACM Press.
10. Organization for the Advancement of Structured Information Standards. Privacy policy profile of XACML v2.0. Technical report, OASIS, 2005.
11. G. Piolle. *Agents utilisateurs pour la protection des donnes personnelles : modlisation logique et outils informatiques.* PhD thesis, Universit Joseph Fourier - Grenoble I, Grenoble, France, 6 2009.
12. G. Piolle and Y. Demazeau. Obligations with deadlines and maintained interdictions in privacy regulation frameworks. In *Proceedings of the 8th IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'08)*, pages 162–168, Sidney, Australia, December 2008. IEEE Computer Society.
13. G. Piolle and Y. Demazeau. Déléguer la protection des données personnelles des agents cognitifs. *Revue d'Intelligence Artificielle*, 24(3/2010):357–390, June 2010.
14. G. H. von Wright. Deontic logic. *Mind*, 60:1–15, 1951.
15. World Wide Web Consortium. Platform for Privacy Preferences specification 1.1., 2006. `http://www.w3.org/P3P/`.