PROGRAM TRANSPARENCY FOR LEGAL MACHINES

Vytautas Čyras and Friedrich Lachmayer

Associate Professor, Vilnius University
Faculty of Mathematics and Informatics, Naugarduko 24, 03225 Vilnius, Lithuania
Vytautas.Cyras@mif.vu.lt; http://www.mif.vu.lt/~cyras/

Professor, University of Innsbruck
Innrain 47, 6020 Innsbruck, Austria
Friedrich.Lachmayer@uibk.ac.at; http://www.legalvisualization.com

Keywords: compliance, informational process, legal requirements, legal visualization, text culture and machine culture

Abstract: This paper attempts to define the transparency problem. The context is the changeover from a text culture to a machine culture in law. The paradigm change to electronic procedures reveals new contexts for justice. Note that equal access to e-procedures does not guarantee justice. The transparency of the law leads to the transparency of programs. We formulate two requirements for legal machines: 1) the architecture of the program must be made accessible; and 2) the program must provide legal protection. The implementation of these requirements is a subject for software engineering. A need therefore arises for the requirements to flow down to lower level specifications. In the end we define program transparency as a compliance problem.

1. Introduction

There are simple legal machines, such as traffic lights, barriers and vending machines, and complex ones, such as the electronic forms that are used in tax and finance. An example of the latter is FinanzOnline1 in Austria. A legal machine can be defined as a machine in a system whose actions have legal importance and legal consequences [Čyras & Lachmayer 2012]. Legal machines shift raw facts into institutional facts. The raw facts come from the Is world, whereas the institutional facts come from the Ought. Legal machines contribute to law enforcement, and their programs implement legal norms.

Suppose the following two elements:

1. A general norm which is formulated in a legal text such as a law or a by-law; and
2. An individual norm which is formulated in a court judgement or an administrative decision.

The following means of legal protection serve the different parties in the context of a text culture:

1. Transparency of law. Legal texts are made accessible to citizens. An example is the Austrian e-government application (portal) HELP,2 which states the applicable law for various situations and supports ex-ante analysis; and

---

1 FinanzOnline provides a one-click link to the Austrian tax administration; see https://finanzonline.bmf.gv.at/.

2 HELP.gv.at – a government agency help site on the Internet, which offers necessary information for living and working in Austria.
2. Ex-post legal protection. An example of this is RIS, which publishes cases and supports ex-post analysis.

These two types of information were not available in the beginning and were improved over the course of time, but they now come as a standard. However, this standard is missing in the beginning of the machine culture, and we will therefore consider how to compensate for this deficit. We see the problem of a lack of transparency and ex-ante analysis support in the machine culture that is still young. For example, a party cannot wait three years to resolve a conflict with a legal machine (Fig. 1).

Therefore we have a proposal that is formulated as two requirements for programs for legal machines:

Requirement 1: The architecture of the program for a legal machine must be made accessible to the user.

Requirement 2: The programs for a legal machine must provide trained, effective and rapid legal protection.

We aim to ensure that there is the same standard of transparency and legal protection in the machine culture as there is in the text culture.

![Diagram](image)

Fig. 1: Technical implementation of a changeover from a legal text to a program. The problem is to ensure transparency and legal protection at the same level in both (a) and (b)

As examples we see the following situations which serve as imaginary cases:

---

3 The Legal Information System of the Republic of Austria; http://www.ris.bka.gv.at/.
Example 1: The law provides for ten variations/possibilities but the program only contains nine.

Let us suppose this applies to expenditure categories in tax law. Suppose you want to declare expenditure in the tenth category in your tax declaration, but the program does not allow this option. What do you do?

Example 2: A vending machine is designed not to give money back. This causes a problem for customers who expect to receive change.

The latter situation can be observed in parking places which are equipped with unsophisticated ticket machines that give a very limited amount of money back (e.g. 50 cents).

The different variations/possibilities like the ten in Example 1 may be listed in different articles or laws. A legal norm is a result of the interpretation of legal texts, and need not be extracted from one article; its formulation may extend to several text passages.

2. The context of justice in the paradigm change to electronic procedures

Hans Kelsen (1967) pursued several objectives in his Pure Theory of Law:

- **First, a paradigm shift in legal theory.** The Pure Theory of Law became a new jurisprudential standard and in this respect shaped the twentieth century.
- **Second, the overcoming of the natural law.** Since the second edition of the Pure Theory of Law, it has not quite been possible to argue using the natural law, and thus this theory has outlived the natural law.
- **Third, to stop argue using the concept of justice;** but he did not succeed with this objective. Despite his long and learned essay on justice, the focus of his attack has survived and continues to be a common theme.

Of course, the talk today is more about injustice, but from a dialectical view this comes to the same thing. What is justice, and why has it remained a topical issue in the emerging machine culture? There are many approaches to defining justice. They come from different times and have different levels of importance. The easiest way seems to be to define justice as a meta-system for the law, which means that it is “above” the law as it is or, more generally, is in a relationship with the law. In this sense, justice is not a part of the law, thus not a sub-element of the law, but is different from it. However, external systems can be functionally connected to the main system, and so be like the meta-data of a document; the meta-data and the document are separated, and yet are in close conjunction.

With regard to content, justice can contain different elements:

- The meaning of legal material can be made “law” by justice. This is achieved from the law through autopoiesis. Alternatively, justice can be made into a legally external system.
- Traditionally, justice also includes the meta-norms of the law, i.e. how to shape the law and especially how to apply it. These aspects can be either just or unjust. As the examples of the past show, a correct general text can be perverted when it is enforced by the law. Also a general text may be formulated in an unfair way from the outset.

**The principle of equality before the law.** One of the fundamental principles of justice is equality. The difficulty with the principle of equality relates less to objectively justifiable differentiations than to the uneven application of the law in the context of friend-enemy schemes. Two legal statutes appear, one for friends and the other for enemies. The principle of equality is thus split in two. One principle of equality applies to friends, and the other applies to enemies. This is unjust and vaguely reminiscent of the ancient highwayman Procrustes, who had two beds and misused both.

A closer look reveals that the application of justice is less of a normative problem, because it is preceded by a cognitive problem concerning matters such as interpretation. A preliminary issue is
that what is found has to be interpreted. However, this is not a normative issue: the subsumption is interpretative, i.e. cognitive at its core. Depending on the legal concepts that are applied in the interpretation, the subsumption, which follows the interpretation, will be different.

If the image of a human being is taken as given, then human rights are applicable. But if the image of the legally honest citizen is denied by the dishonest, human rights are no longer applicable. The outlaws could be eliminated with impunity in the Middle Ages, and similarly today there are sanctioning machines which do the same job in a legal vacuum.

The context of justice does not only affect the large and the small politics that sometimes in their hubris put themselves forward as *legibus solutes*, i.e. legally free, and not within the scope of justice. It also affects legal machines.

The upcoming world of electronic administrative and judicial proceedings brings a new context for the application of justice. With electronic proceedings, the context is all about everyday justice for those citizens who are affected, and about the procedurally mediated law that they enjoy, individually and specifically. But even with IT-based everyday justice the cognitive problem is upstream of the normative problem.

**Subsumption.** The key to the application of the law and thus to access to justice is subsumption. The special feature of matters carried out with electronic forms (which we will refer to as e-proceedings or e-procedure) is that subsumption is accomplished by the parties and not by the court or the administrative authority. In traditional proceedings, the parties submit their allegations, these are confirmed or rejected in the course of the investigation, and finally the state of affairs (Ger. *Sachverhalt*) is established by the authority. The state of affairs serves as the basis for the subsumption. Thus there are a number of regulatory steps between the parties’ assertions and the subsumption.

In e-proceedings this is different. One party fills in the input fields itself. The entries in these fields, however, are not commonly known but require knowledge of jargon. Thus, the performance of subsumption is delegated to the parties. To make entries, the parties have to use their knowledge of the legal terms that are defined by the sources of law in a complicated way.

Back-office programs for e-proceedings are very quick and accurate at drawing legal conclusions from input data. However, the content of the conclusion depends to a large extent on the starting material that was completed in the parties’ front offices. A party who is not skillful in law may be overwhelmed by this. This is an opportunity for intermediaries, allowing them to continue to position themselves professionally. The complexity of the law is not reduced by e-proceedings, but, as in the Middle Ages, continues to require intermediaries who now must be expert in information technology.

All parties are equal in e-proceedings, and in this respect e-proceedings seem to be fair. However, in the actual performance of an e-procedure there may be a big difference between communicating with a human operator and communicating with a machine. A human operator can offer flexibility, and this cannot be offered by a machine. As an example, suppose a traveler sees a train already approaching and tries desperately to buy an appropriate ticket from the ticket machine. A human being would assist better than the machine.

The perfect rationality of the machine is preferred in situations where people’s emotions become a barrier. However, this has to be properly adapted in the new practice. The requirement is to adapt machines to people and not vice versa. Machine rationality is required to deal with people’s situational emotions.

The e-procedures are likely to prevail in the middle areas of all kinds of proceedings. More and more situations are standardized and can be handled abstractly. However, things are different in the
two peripheral areas. The hard cases will be handled as before with manual legal work. An example of non-compliant software design is provided in [Oberle et al. 2012]. This example demonstrates legal reasoning that leads to a violation of data privacy law. The running example in [Oberle et al. 2012] is a situation in which a user’s consent, given by clicking “yes” on his mobile phone, is not treated as an effective consent in the legal sense.

For trivial matters of life, machines will continue to catch on. But these tasks, although legally performed, will still not be perceived by people as law.

It is likely that a new type of situational personalization will occur, such as intelligent traffic lights. Similar animistic norm-setters have been recorded as long-lasting in people’s consciousness.

**Equality ≠ justice.** Equality is far from guaranteeing justice. An example is transactional taxes. These are all the same, but they have a greater impact on the income of the poor than on the income of the rich. Therefore this is perceived as unjust.

The knowledge of the law is similar. Nowadays the law is conveyed electronically in the same way for all people, whether by RIS or HELP. However there are big differences in individuals’ ability to deal with this knowledge. We also need to add the different technical skills and aptitudes in dealing with the new media.

**Ex-ante legal protection.** An open issue is certainly the lack of ex-ante legal protection against incorrect electronic forms in e-procedures. Consider again Example 1, in which the law provides for ten cases but only nine of these cases are mentioned in the e-forms. The question arises of how someone should behave if he wishes to enter information in the tenth case, when he cannot do this electronically. Any ex-post legal protection comes too late after the first instance of this and the expiration of the deadline, and it is dysfunctional, since it only involves the input stage.

The paradigm shift to e-procedures provides small- and medium-range domains with a new standardized legal culture, which could not be achieved before. This paradigm shift provides equality and is “fair”, apart from the concern over different starting conditions. However, as in the past, injustice will be a massive problem in the areas where it is so great that it cannot be consciously perceived. For nothing is as invisible as the things that may be overlooked. Here the paradigm shift to e-procedures will not change injustice.

### 3. Transparency in software engineering

Requirements 1 and 2 in Section 1 are formulated on a very high level. The next question is how they can be implemented in a legal machine. In principle, the architecture of the legal machine can be made available to the user. Designing legal machines is a subject for software engineering (SE). Transparency of computer programs is a concern of both SE and human-computer interaction.\(^4\) Here transparency is combined with the principles of information hiding\(^5\) and separation of

---

\(^4\) Transparency means that a distributed system hides its distributed nature from its users, appearing and functioning as a normal centralized system. In software engineering, it is also considered good practice to use different abstraction layers. There are many types of transparency, see http://en.wikipedia.org/wiki/Transparency_(human%E2%80%93computer_interaction).

\(^5\) In computer science, information hiding is the principle of the segregation of the design decisions in a computer program that are most likely to change. This protects other parts of the program from extensive modification if the design decision is changed. The protection involves providing a stable interface which protects the remainder of the program from the implementation (the details that are most likely to change), see http://en.wikipedia.org/wiki/Information_hiding.
On different abstraction levels, a software element is treated as a black box which can be viewed in terms of its input, its output and its transfer characteristics without any knowledge of its internal workings.

It is not an easy task to formulate and achieve transparency in SE. Different users may be interested in different architectural elements. Not every design solution can be made accessible to each user. This is for reasons of security and complexity, to mention just a few. Software requirements, including transparency requirements, are formulated in the early phases of the software development life cycle. Legal requirements are at a high level and are also a concern of requirements engineering. A flowdown of the requirements is needed to develop lower level specifications, which are formulated in the system specification or a similar document. There is a need to flow down the high level Requirements 1 and 2 in the Introduction above, too. After this the program is designed and implemented. Thus the resulting program can be made compliant with the initial legal requirements.

However, the story above is in practice rarely so simple. There may be a wide span between the legal requirements and the technical specifications. Failure to understand the law is one of the reasons why the program may be non-compliant [Silverman 2008, pp. 59–61]. On the one hand, legal texts constitute only a part of the whole legal system. The meaning of the law – the Ought realm – can scarcely be understood from a single legal text. Therefore it is difficult for a beginner to understand the spirit of law while reading a statute in isolation. For this reason only well-defined compliance problems can be implemented by ticking boxes in an audit document.

3.1. On implementing Requirement 2 – legal protection

Legal protection, which is mentioned in Requirement 2, is even more difficult to implement than the architectural transparency in Requirement 1. To make sure Requirement 2 flows down, the efforts of software engineers are not enough. Organisational means and invention may be needed. The following paragraph sets out an imaginary situation.

Consider an automatic barrier to a paying car park. Suppose you have used the car park, paid for your parking, receive a ticket and want to leave the car park, but the barrier does not lift up to let you out. Suppose that the cause is purely technical, such as a malfunction of the barrier’s motor. What action can you take and what means of legal protection do you have? Some car parks provide a 24 hours/7 days voice connection to a human operator. You can therefore explain the matter to the operator and he can arrange for the barrier to be lifted manually. However this is not always the case.

So far as legal protection is concerned, the architect of the car park should, under Requirement 2, supply instructions for how the user should behave in emergency cases.

To sum up: although legal machines are usually treated as black boxes, transparency can be advanced with a transparent flowdown of legal requirements.

4. Formulating program transparency as compliance

We now attempt to formulate program transparency as a problem of compliance with the law. Klaus Julisch (2008) suggests that academia should undergo a paradigm shift, from “selling” security when organizations seek to “buy” compliance to complementing current security research with additional research into security compliance:

---

6 In computer science, separation of concerns (SoC) is a design principle for separating a computer program into distinct sections, such that each section addresses a separate concern, see http://en.wikipedia.org/wiki/Separation_of_concerns.
“[A]s long as careers are terminated and people go to jail…for failures in compliance – rather than security – the commercial world will continue to pursue compliance rather than security as their primary goal.” [Julisch 2008, p. 71] 

Julisch defines security compliance as follows: “security compliance, in IT systems, is the state of conformance with externally imposed functional security requirements and of providing evidence (assurance) thereof” [Julisch 2008, p. 72]. He defines the security compliance problem as follows:

“Definition: Given an existing IT systems $S$ and an externally imposed set $R$ of security requirements. The Security Compliance Problem is to make system $S$ comply with the security requirements $R$ and to provide assurance that an independent auditor will accept as evidence of the compliance of system $S$ with requirement $R$.”

Following the definition above, we would like to formulate the Compliance Problem, as follows: (1) to make a legal machine’s program $S$ comply with requirements $R$ that relate to a law $L$; and (2) to provide assurance that an independent auditor will accept as evidence (Fig. 2).

**Fig. 2: The compliance problem**

We simply added a law $L$ to Julisch’s formulation. The semi-formal definitions above can only serve as a first iteration. In practice it will be difficult for solutions to the problem to result in a yes or a no. In practice, more elements are involved. Feedback loops would improve $S$, $R$ and $L$. The conceptualization of $L$ may involve different elements, depending on the abstraction level. $L$ may stand for a legal principle, a whole statute or a specific provision.

**Bonazzi-Hussami-Pigneur IT compliance framework.** There is no silver bullet to solve the compliance problem. This is also explained in the IT compliance framework, which is worth special attention; see [Bonazzi et al. 2009]. Two dimensions, Legal and IT, and two kinds of source for regulations with which a company must comply, External and Internal, are depicted by squares (Fig. 3). Different alignments are represented by arrows that point to the artefact that is defined.

Every concept in Fig. 3 denotes a broad field. Corporate non-compliance, corruption, etc. are just a few examples of violations. Non-compliance can be civil, criminal, or administrative, but also reputational or market-based. To analyze a company for compliance, one analyst would have to be aware of norms in various branches of the law.

5. Conclusions

We depart from the view that machines are tools, and instead take the view that legal machines are legal actors capable of triggering institutional facts. We have identified the transparency problem
and formulated two requirements for legal machines. However, implementing these requirements is not an easy task, even if it is possible for it to be achieved at all. There is no silver bullet to attack regulatory compliance requirements – no one-off, best-of-breed solution. Moreover, there are different contexts of justice, and e-procedures do not guarantee justice.

![Diagram](image)

**Fig. 3**: The regulation and IT alignment framework adapted from [Bonazzi et al. 2009]

### 6. Acknowledgement

V. Čyras has been supported by the project “Theoretical and engineering aspects of e-service technology development and application in high-performance computing platforms” (No. VP1-3.1-ŠMM-08-K-01-010) funded by the European Social Fund.

### 7. References


