

Certificar decisiones automatizadas en la administración pública

Joost J. Joosten

Universitat de Barcelona

Transparencia, inteligencia artificial y Administración Pública
Universida de Castilla~La Mancha (Via Teams)
19 de mayo del 2022



¿Deseable o necesario?



infracción imputada y sancionada en cuanto que no se han incumplido los tiempos de descanso semanales.

En segundo lugar considera que los hechos denunciados no están suficientemente probados a efectos de poderlos considerar constitutivos de la infracción sancionada. En este apartado señala que el tacógrafo del que se han obtenido datos tiene una programación o configuración de fábrica que adolece de errores y que hace que sus resultados no sean fiables ni ciertos. No se trata de una avería o de un mal funcionamiento sino de errores de fabricación, configuración y/o programación llamando la atención sobre la falta de homologación del tacógrafo y, especialmente, del software utilizado dentro del mismo. A lo anterior añade que no consta, y por lo tanto falta, la homologación del software utilizado por las autoridades para obtener y procesar los datos registrados en el tacógrafo.

Se acepta lo alegado por la parte demandante en lo que se refiere a la ausencia de prueba de cargo suficiente respecto al software utilizado por la autoridad correspondiente para obtener los datos registrados en el tacógrafo por lo que, sin necesidad de analizar el resto de la fundamentación invidia

N. Sentencia: 30/2019, CONTENCIOSO/ADMTVO corte N. 4 de Valladolid

Resultados dudosos



- Aplicación de la ley requiere de discrecionalidad

Resultados dudosos



- Aplicación de la ley requiere de discrecionalidad
- para ello, ambigüedad es deseable

Resultados dudosos



- Aplicación de la ley requiere de discrecionalidad
- para ello, ambigüedad es deseable
- cualquier proceso automatizado en el ámbito legal necesita des-ambigüedad

Resultados dudosos



- Aplicación de la ley requiere de discrecionalidad
- para ello, ambigüedad es deseable
- cualquier proceso automatizado en el ámbito legal necesita des-ambigüedad
- ¿El programador tiene que desambiguar?

Resultados dudosos



- Aplicación de la ley requiere de discrecionalidad
- para ello, ambigüedad es deseable
- cualquier proceso automatizado en el ámbito legal necesita des-ambigüedad
- ¿El programador tiene que desambiguar?
- Y, ¿si hay un error?

Resultados dudosos

III - Les formules de calculs DE L'APL ET DES AL

Secteur locatif ordinaire

Depuis la réforme intervenue le 1^{er} janvier 2001, le montant de l'aide est obtenu par application de la même formule en AL et en APL (cf. article D. 832-16 du CCH).

$$APL \text{ ou } AL = L + C - Pp$$

Logements-foyers

Le montant de l'APL foyer est obtenu par application de la formule (cf. article D. 832-24 du CCH) :

$$APL = K [E - E0]$$

Avec application de deux barèmes, APL 1 foyer et APL 2 foyer (cf. articles D. 832-25 et D. 832-26 du CCH)

Le montant de l'AL est obtenu selon la formule (cf. article D. 842-15 du CCH) :

$$AL = K [L + C - L0]$$

Accession

Le montant de l'APL et de l'AL est obtenu par application de la même formule (cf. articles D. 832-10 et D. 842-6) du CCH :

$$APL \text{ ou } AL = K [L + C - L0]$$

- Sistemas de pagos de bonificación del ejercito Francés: Louvois/SourceSolde

Resultados dudosos

III - Les formules de calculs DE L'APL ET DES AL

Secteur locatif ordinaire

Depuis la réforme intervenue le 1^{er} janvier 2001, le montant de l'aide est obtenu par application de la même formule en AL et en APL (cf. article D. 832-16 du CCH).

$$\text{APL ou AL} = \text{L} + \text{C} - \text{Pp}$$

Logements-foyers

Le montant de l'APL foyer est obtenu par application de la formule (cf. article D. 832-24 du CCH):

$$\text{APL} = \text{K} [\text{E} - \text{E0}]$$

Avec application de deux barèmes, APL 1 foyer et APL 2 foyer (cf. articles D. 832-25 et D. 832-26 du CCH)

Le montant de l'AL est obtenu selon la formule (cf. article D. 842-15 du CCH):

$$\text{AL} = \text{K} [\text{L} + \text{C} - \text{L0}]$$

Accession

Le montant de l'APL et de l'AL est obtenu par application de la même formule (cf. articles D. 832-10 et D. 842-6) du CCH:

$$\text{APL ou AL} = \text{K} [\text{L} + \text{C} - \text{L0}]$$

- Sistemas de pagos de bonificación del ejercito Francés: Louvois/SourceSolde
- en 2012: 465 M € pagos erróneos

Resultados dudosos

III - Les formules de calculs DE L'APL ET DES AL

Secteur locatif ordinaire

Depuis la réforme intervenue le 1^{er} janvier 2001, le montant de l'aide est obtenu par application de la même formule en AL et en APL (cf. article D. 832-16 du CCH).

$$\text{APL ou AL} = L + C - Pp$$

Logements-foyers

Le montant de l'APL foyer est obtenu par application de la formule (cf. article D. 832-24 du CCH):

$$\text{APL} = K [E - E0]$$

Avec application de deux barèmes, APL 1 foyer et APL 2 foyer (cf. articles D. 832-25 et D. 832-26 du CCH)

Le montant de l'AL est obtenu selon la formule (cf. article D. 842-15 du CCH):

$$\text{AL} = K [L + C - L0]$$

Accession

Le montant de l'APL et de l'AL est obtenu par application de la même formule (cf. articles D. 832-10 et D. 842-6) du CCH:

$$\text{APL ou AL} = K [L + C - L0]$$

- Sistemas de pagos de bonificación del ejercito Francés: Louvois/SourceSolde
- en 2012: 465 M € pagos erróneos
- dejó algunos soldados y sus familias sin sueldo durante meses

Mas cercano de casa: Civio vs Bosco

This article belongs to the debate » [The Rule of Law versus the Rule of the Algorithm](#)

02 April 2022

The Paradox of Efficiency: Frictions Between Law and Algorithms

On the 13th of January 2022, a Spanish Administrative court ruled in favour of algorithmic opacity. Fundación Civio, an independent foundation that monitors and accounts public authorities, [reported](#) that an algorithm used by the government was committing errors.¹⁾ BOSCO, the name of the application which contained the algorithm, was implemented by the Spanish public administration to more efficiently identify citizens eligible for grants to pay electricity bills. Meanwhile, [Civio designed a web app](#) to inform citizens whether they would be entitled for this grant.²⁾ Thousands of citizens used this application and some of them reported that, while Civio's web app suggested



Ana Valdivia

Dr Ana Valdivia is a Postdoctoral Researcher at King's College London (ERC Security Flows). She examines how algorithms impact on people's life from a technical, political, and legal perspective.



Javier de la Cueva

Javier de la Cueva is a lawyer, lecturer and researcher in topics related to open knowledge, ethics and the digital world.

Explore posts related to this:
[Algorithmic Efficiency](#), [Algorithmic Justice](#), [Rule of Law](#), [Rule of the Algorithm](#)

Programa Bosco: errores en el cálculo de bonificación social

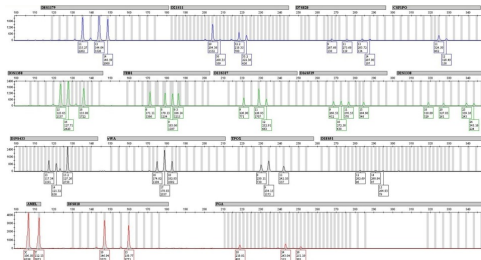
Lo mínimo: acceso al código fuente

En Francia es obligatorio publicar código fuente de software en la administración pública

Acceso al código fuente no resuelve todos los problemas.

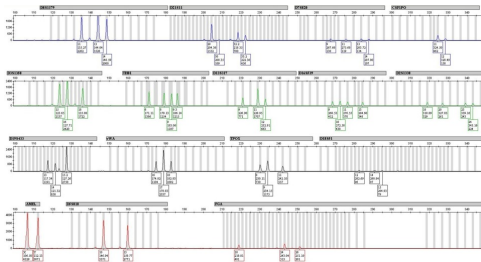
Evidencias dudosas

- En tres juicios en los EEUU se ha reclamado **abrir el código fuente al tribunal** del software propietario de secuenciación de ADN porque algunos resultados eran dudosos
 - STRmix



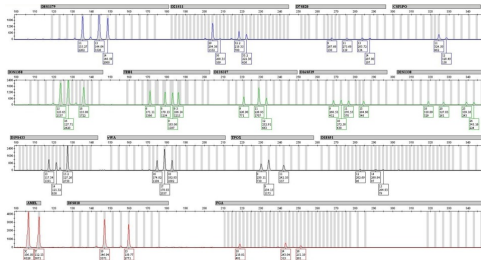
Evidencias dudosas

- En tres juicios en los EEUU se ha reclamado **abrir el código fuente al tribunal** del software propietario de secuenciación de ADN porque algunos resultados eran dudosos
 - STRmix
 - FST



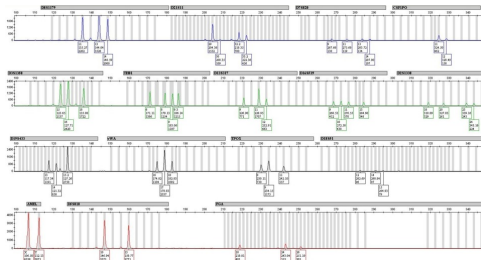
Evidencias dudosas

- En tres juicios en los EEUU se ha reclamado **abrir el código fuente al tribunal** del software propietario de secuenciación de ADN porque algunos resultados eran dudosos
 - STRmix
 - FST
 - TrueAllele (todavía propietario)



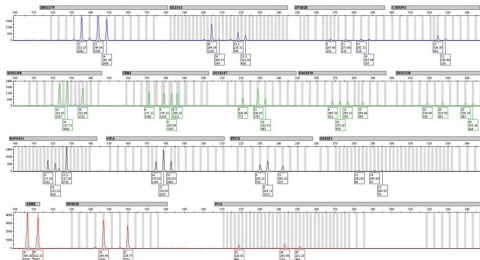
Evidencias dudosas

- En tres juicios en los EEUU se ha reclamado **abrir el código fuente al tribunal** del software propietario de secuenciación de ADN porque algunos resultados eran dudosos
 - STRmix
 - FST
 - TrueAllele (todavía propietario)
- En dos casos fue cedido



Evidencias dudosas

- En tres juicios en los EEUU se ha reclamado **abrir el código fuente al tribunal** del software propietario de secuenciación de ADN porque algunos resultados eran dudosos
 - STRmix
 - FST
 - TrueAllele (todavía propietario)
- En dos casos fue cedido
- Tener acceso al código fuente no lo soluciona todo



¿Qué es la homologación?



- ¿Es una cuestión de confianza?

¿Qué es la homologación?



- ¿Es una cuestión de confianza?
- Certificar \implies algo es cierto

¿Qué es la homologación?



- ¿Es una cuestión de confianza?
- Certificar \implies algo es cierto
- Verificar \implies algo es verídico

La imposibilidad de homologación sin restricciones



Alan Turing

- Teorema matemático:

La imposibilidad de homologación sin restricciones



Alan Turing

- Teorema matemático:
- homologación sin restricciones es imposible!

La imposibilidad de homologación sin restricciones

Llamemos a un programa P un *programa de homologación universal* cuando P coge dos entradas (inputs)”

- 1 otro programa Q en un lenguaje compatible con P y,
- 2 una especificación S en un lenguaje compatible con P que describe el comportamiento del programa Q ;

y, dado estas dos entradas Q and S , el programa P da como salida (output):

- “**YES**” si el programa Q hace lo que dice S i, da como salida
- “**NO**” si el programa Q hace algo distinto a lo que dice S .

Theorem

Es imposible que exista un programa P de homologación universal.

Software formalmente verificado

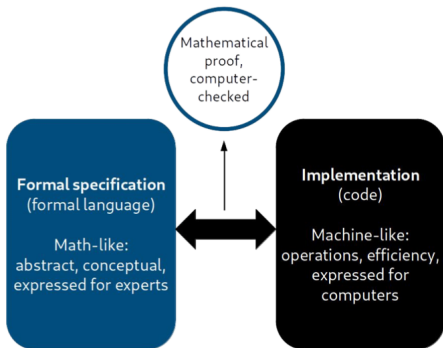
Componentes del software formalmente verificado

- Σ Especificación: descripción matemático no-ambiguo del comportamiento del software
- Π Implementación: el código, el software, el algoritmo que hace el trabajo
- Δ Prueba/demostración: una prueba matemática que Π concuerda con Σ

La especificación Σ está escrito en lenguaje formal (lógica, matemática), así que **¿como hacerlo accesible al público más genérico?**

¿Qué es la homologación?

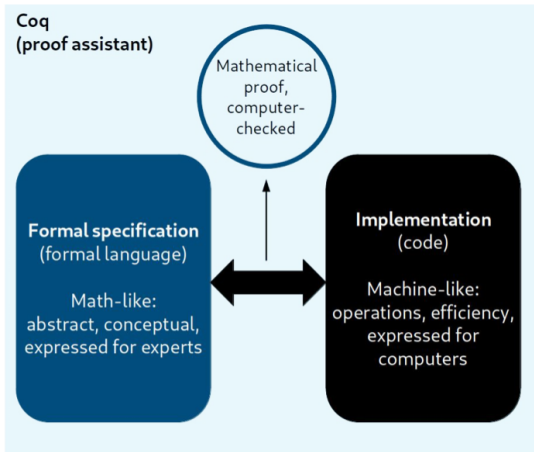
Formal verification



Transparencias FV: González Bedmar

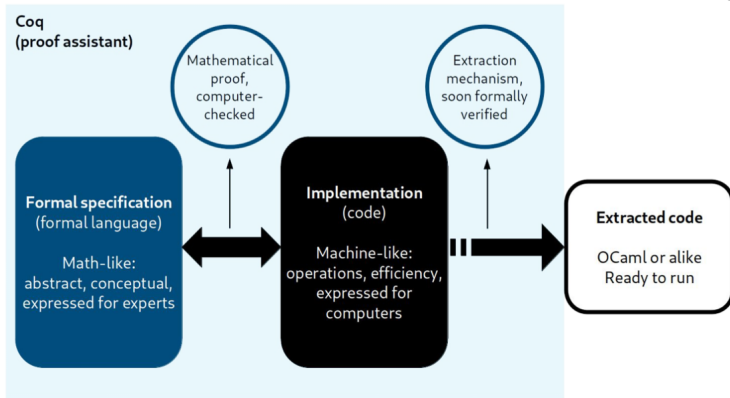
¿Qué es la homologación?

Formal verification



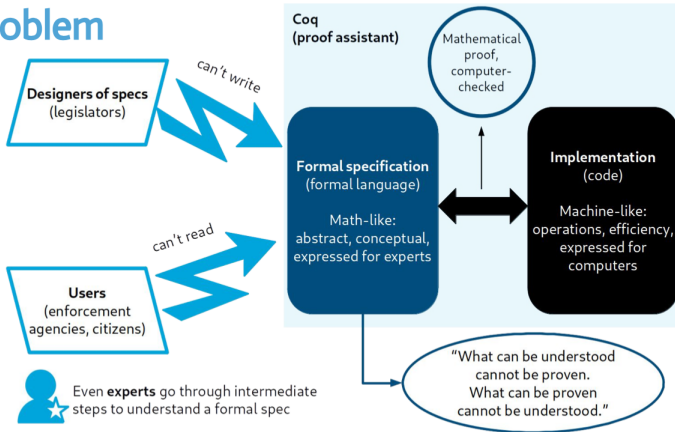
¿Qué es la homologación?

Formal verification

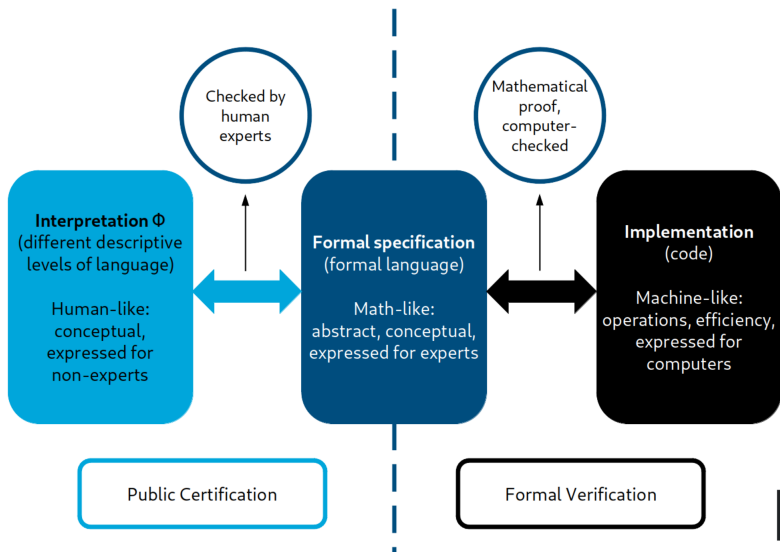


¿Qué es la homologación?

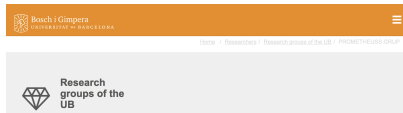
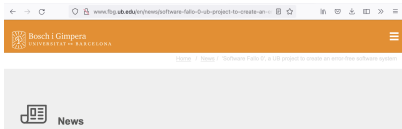
Problem



¿Qué es la homologación?



Siete años de investigación en Barcelona



'Software Fallo 0', a UB project to create an error-free software system

12 -02- 2019

All software contains bugs; even the software that controls the aeronautical or military industry has bugs in its final version. This situation is particularly troubling because of the increasing dependence on software of key processes such as computer voting mechanisms, medical technologies, and applications that decide whether or not a person complies with the law. A team from the University of Barcelona participates in a four-year project that promotes a new paradigm for the software industry: the development,

www.fbg.ub.edu/



PROMETHEUSS GRUP

Software unreliability and the legal system

Software malfunction can appear in one or several layers of the software development cycle, including: natural language specifications, technical specifications, formal specifications, coding, compilation, installation, and execution. The consequences of software malfunction in legal and administrative settings arguably imply the violation of legal principles, loss of valuable resources, attacks on civil rights (such as well-documented cases of automated racial discrimination), and degradation of legal systems. Also, in the future as well as in the present, it may aggravate the societal loss of confidence in technology and in government alike. Legally binding decisions taken based on data produced by software, or even decisions which are automated outright, very rarely acknowledge the existence of several crucial potential problems inherent to the nature

WHO WE ARE

Members

- Dr. Joost J. Joosten (mathematical logic, team leader)
- Ana Borges (mathematical logic)
- Joaquim Casals Buffardi (computer science)
- Mireia González Bedmar

Convenio Universidad de Barcelona (FBG), Formal Vindications S.L. & Guretruck S.L.

Hace tres semanas

Conference on Algorithmic Law Design and Implementation

APRIL 28-29, 2022
UNIVERSITAT DE BARCELONA

SCHEDULE, REGISTRATION and + INFO
<https://www.uh.edu/prooftheory/evant/lawdesign/>

KEYNOTE SPEAKERS

Formal Verification and Governance of Financial Algorithms with Inandra
Grant Olney Passmore
Inandra, USA

Hybrid intelligence for algorithmic law design
Bart Verheij
Bernoulli Institute of Mathematics, Computer Science and Artificial Intelligence - University of Groningen

INVITED SPEAKERS

Opening talk: Algorithmic Law Design and Implementation. From Greve to Creadie
Joost J. Joosten
Universitat de Barcelona, Spain

Legal Methods for Algorithmic Law
David Fernández-Duque
Ghent University, Belgium

Public Certification of Software and its necessity in Computable Laws. PV Time as the first application
Mirna González Bedmar
Formal Verification SL, Spain

Is coding the law legal? A French and European approach
Liane Hutterer
Université Paris 1 Panthéon-Sorbonne, France

Verifying well-behaved execution of legislative programs with the Catalan domain-specific language
Denis Merigoux
Inria, Project Team Prosecco, France

Drafting EU Legislation in the Era of AI and Digitalisation
Fernando Nubia Durango, Willy von Puymbroech
LEOS Project, European Commission

Monica Palmirani
Università di Bologna, Italy

Auditing IT-systems used for automated individual decision-making in public sector; experiment in The Netherlands
Marlies van Ech
Hooghiemstra & Partners and Radboud University, The Netherlands

Verified extraction to OCaml from Coq, in Coq
Yarnick Forster
Inria, Project Team Gallinette, France

Crafting a legislation ready for digital public administration
Julius Lyh-jensen, Christine Holmgreen Mejling & Mette Eigaard Rasmussen
Agency for Digitalisation, Ministry of Finance, Denmark

Model-Checking as an approach to algorithmic law and the case of Regulation 561
Moritz Müller
Universitat de Barcelona, Spain

Imagine lawyers are not your enemies: Legal challenges and digital rights
Susana de la Sierra
Universidad de Castilla-La Mancha, Spain

PROGRAM COMMITTEE

Joost J. Joosten
Inria, Project Team Prosecco, France

David Fernández-Duque
Ghent University, Belgium

Denis Merigoux
Inria, Project Team Prosecco, France

Fernando Nubia Durango
LEOS Project, European Commission

Monica Palmirani
Università di Bologna, Italy

ORGANIZING COMMITTEE

Joost J. Joosten
Inria, Project Team Prosecco, France

David Fernández-Duque
Ghent University, Belgium

Denis Merigoux
Inria, Project Team Prosecco, France

Moritz Müller
Universitat de Barcelona, Spain

SPONSORS

Universitat de Barcelona

Universitat de Girona

Universitat de Lleida

Universitat de València

Universitat de Zaragoza

Universitat de Sevilla

Universitat de Murcia

Universitat de Granada

Universitat de Córdoba

Universitat de Jaén

Universitat de Almería

Universitat de Cádiz

Universitat de Huelva

Universitat de Extremadura

Universitat de Castilla-La Mancha

Universitat de Castilla-La Mancha

SPONSORS

Bosch i Gimpera
UNIVERSITAT DE BARCELONA

J.J. Joosten (UB)

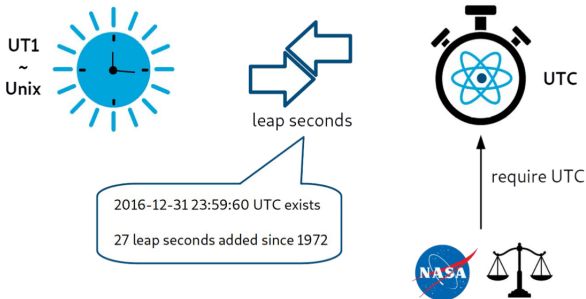
Tecnología al servicio del ciudadano

Amsterdam, 19 de mayo

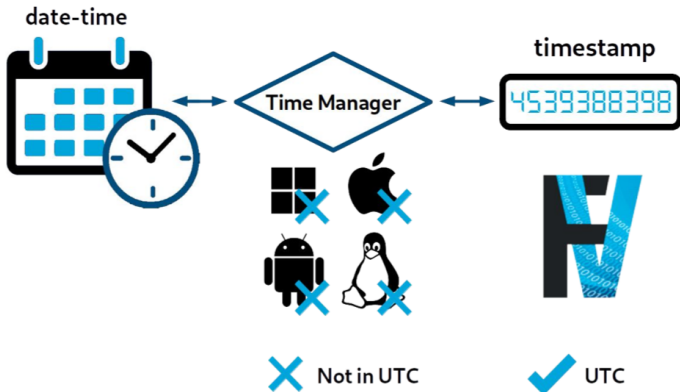
17 / 34

Librería del tiempo

Time measurement

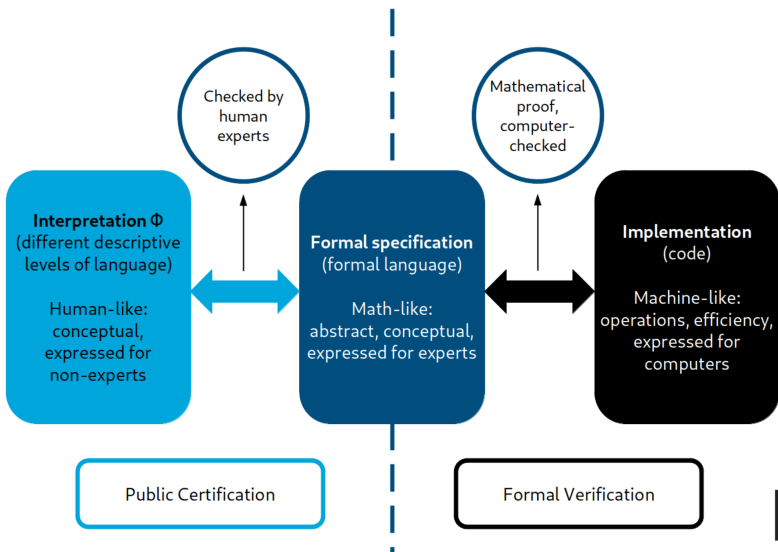


Time formats and managers



Más importante: formalmente verificado

¿Qué es la homologación?



Example in FV Time



Formal specification (in Coq): `utc_timestamp_plain`
`#|[pred t' | (epoch <= t' < t)%O|]`



Implementation
(code)



Formal specification in a descriptive level of language

Given a time t , returns the cardinality of the set of times that are equal or after the Unix epoch (1970-1-1 00:00:00) and before t .



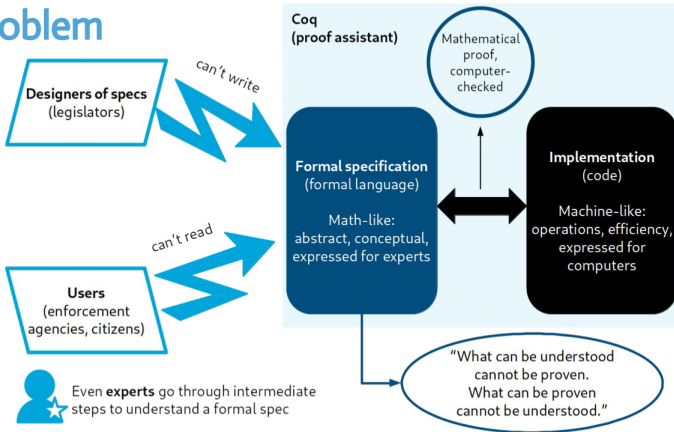
Intuitive specification

Given a time t , returns the number of seconds elapsed since the Unix epoch (1970-1-1 00:00:00).

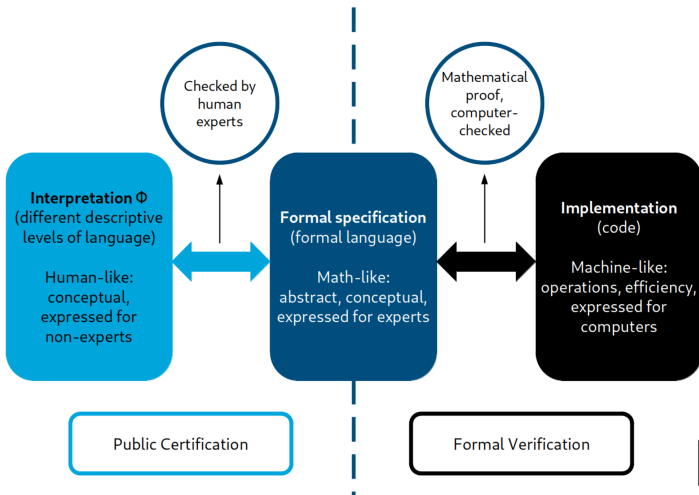
¡Actualmente unos mil veces más costoso que programación convencional!

Un problema central

Problem

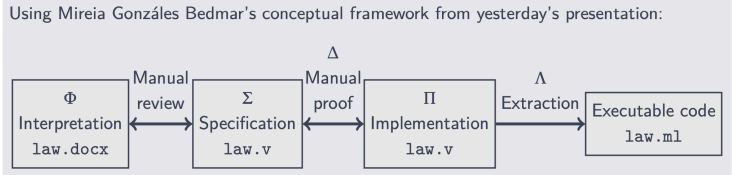


Certificación pública versus verificación formal



Catala: A Shortcut For Legal Expert System Certification

The Usual Way to Produce Verified Software



Catala's approach:



① Catala: A Language Reviewable by lawyers

US Tax Code, Section 132, (c)(1) Qualified employee discount

The term “qualified employee discount” means any employee discount with respect to qualified property or services to the extent such discount does not exceed—

(A) in the case of property, the gross profit percentage of the price at which the property is being offered by the employer to customers

```
scope QualifiedEmployeeDiscount :  
  definition qualified_employee_discount  
    under condition is_property consequence equals  
      if employee_discount >$ customer_price *$ gross_profit_percentage then  
        customer_price *$ gross_profit_percentage  
      else employee_discount
```

¿Puede el código ser la ley?

TENSION TABLE:

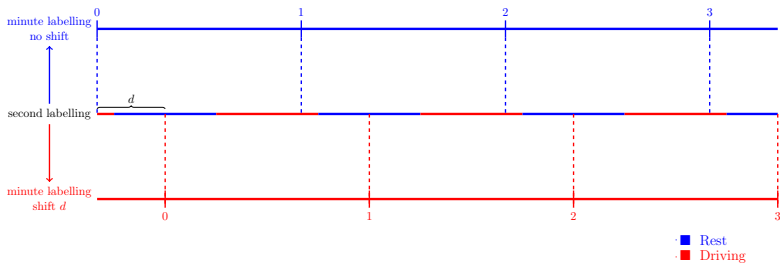
Computable laws:

Language, software paradigm and legal principles

	Specification Language	Programming paradigm	Legal Principles		
			Legal Certainty	Accountability	Contestability
More accurate and exact but less understandable for the general public	Natural Language	Not Formally Verified	Decisions will probably not be consistent with the established legal framework. The text will be accessible and comprehensible to the public and authorities.	Automated decision won't be reliable and explainability will be difficult: the software is not comprehensible to the public, challenging the principle of transparency.	Right to contest turns almost impossible since authorities can't explain software decisions, which will be unreliable.
	Technical Language	Not Formally Verified	Decisions will likely not be consistent with the established legal framework. The text will be less comprehensible to public and authorities.	Automated decision will be barely reliable and explainability will be difficult: the software is not comprehensible to the public, challenging the principle of transparency.	Right to contest turns almost impossible since authorities can't explain software decisions, which will be mostly unreliable.
	Formal Language	Not Formally Verified	Decisions will probably be consistent with the established legal framework. The text will only be accessible to experts.	Automated decision will be quite reliable and explainability will be difficult: the software is not comprehensible to the public, challenging the principle of transparency.	Right to contest turns almost impossible since authorities can't explain software decisions, yet they will probably be working according to the law
	Formal Language	Formally Verified	Decisions will be consistent with established legal framework. The text will only be accessible to experts	Automated decision will be reliable and explainability will be difficult, but it will be guaranteed that the software is the exact reproduction of its specification	Right to contest will be difficult since authorities can't explain software decisions, yet those are working according to the law

Legal certainty = Seguridad jurídica; Accountability = Redición de cuentas; Contestability = Impugnabilidad

Formalización conlleva otras ventajas



Hemos podido demostrar la no-invariancia a los *shifts*

Some regulations regarding weekly rest periods

Regulation (EC) No 561/2006

§8.6. En el transcurso de dos semanas consecutivas el conductor tendrá que tomar al menos:

- dos períodos de descanso semanal normal [de al menos 45 horas], o
- un período de descanso semanal normal y un período de descanso semanal reducido de al menos 24 horas; no obstante, la reducción se compensará con un descanso equivalente tomado en una sola vez antes de finalizar la tercera semana siguiente a la semana de que se trate.

...

Un pequeño análisis...

- Descanso semanal normal: ≥ 45 horas

Un pequeño análisis...

- Descanso semanal normal: ≥ 45 horas
- Descanso semanal reducido: ≥ 24 hours

Un pequeño análisis...

- Descanso semanal normal: ≥ 45 horas
- Descanso semanal reducido: ≥ 24 hours
- Cada semana tiene que haber un descanso semanal normal o un descanso semanal reducido

Un pequeño análisis...

- Descanso semanal normal: ≥ 45 horas
- Descanso semanal reducido: ≥ 24 hours
- Cada semana tiene que haber un descanso semanal normal o un descanso semanal reducido
- Cada otra semana tiene que tener un descanso semanal normal

Un pequeño análisis...

- Descanso semanal normal: ≥ 45 horas
- Descanso semanal reducido: ≥ 24 hours
- Cada semana tiene que haber un descanso semanal normal o un descanso semanal reducido
- Cada otra semana tiene que tener un descanso semanal normal
- Cada descanso semanal reducido tiene que ser compensado en los tres semanas siguientes

No-localidad de las compensaciones

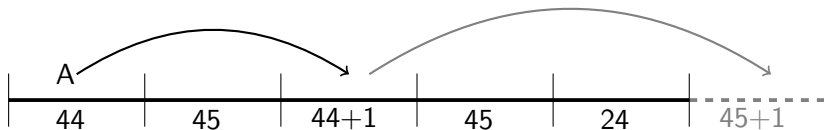


Ilegal

No-localidad de las compensaciones



Illegal

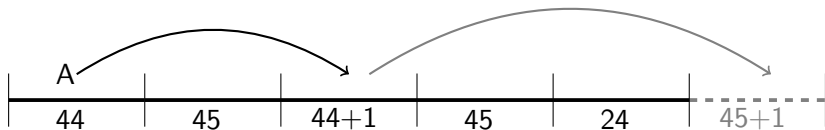


Legal

No-localidad de las compensaciones



Illegal



Legal



Se puede prolongar a libre albedrillo

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?
 - Shift-invariance del labelling, localisme de las condiciones de legalidad, etc.

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?
 - Shift-invariance del labelling, localisme de las condiciones de legalidad, etc.
 - Pero también: si x e y son contenedores de data similares (por ejemplo, repr. de personas legales) con la única diferencia que x tiene un pasaporte e y tiene dos pasaportes, ¿el comportamiento del programa será el mismo para ambos entradas?

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?
 - Shift-invariance del labelling, localisme de las condiciones de legalidad, etc.
 - Pero también: si x e y son contenedores de data similares (por ejemplo, repr. de personas legales) con la única diferencia que x tiene un pasaporte e y tiene dos pasaportes, ¿el comportamiento del programa será el mismo para ambos entradas?
 - ¿Los algoritmos implicados son computacionalmente factibles?

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?
 - Shift-invariance del labelling, localisme de las condiciones de legalidad, etc.
 - Pero también: si x e y son contenedores de data similares (por ejemplo, repr. de personas legales) con la única diferencia que x tiene un pasaporte e y tiene dos pasaportes, ¿el comportamiento del programa será el mismo para ambos entradas?
 - ¿Los algoritmos implicados son computacionalmente factibles?
- Implementar software de ERROR-CERO usando *proof assistants*.

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?
 - Shift-invariance del labelling, localisme de las condiciones de legalidad, etc.
 - Pero también: si x e y son contenedores de data similares (por ejemplo, repr. de personas legales) con la única diferencia que x tiene un pasaporte e y tiene dos pasaportes, ¿el comportamiento del programa será el mismo para ambos entradas?
 - ¿Los algoritmos implicados son computacionalmente factibles?
- Implementar software de ERROR-CERO usando *proof assistants*.
- Desarrollar modelos *general purpose* para leyes temporales cuantitativas.

Actividades de nuestro laboratorio

- Análisis logico-matemático de regulaciones cuantitativos
 - ¿Consistencia?
 - ¿Tiene un comportamiento deseable?
 - Shift-invariance del labelling, localisme de las condiciones de legalidad, etc.
 - Pero también: si x e y son contenedores de data similares (por ejemplo, repr. de personas legales) con la única diferencia que x tiene un pasaporte e y tiene dos pasaportes, ¿el comportamiento del programa será el mismo para ambos entradas?
 - ¿Los algoritmos implicados son computacionalmente factibles?
- Implementar software de ERROR-CERO usando *proof assistants*.
- Desarrollar modelos *general purpose* para leyes temporales cuantitativas.
- Dotar software formalmente verificado con un fragmento de diálogo que nos permite entrar en un diálogo rudimentario con el software sobre su comportamiento

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores
- Certificados de *zero-knowledge* para software propietario!

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores
- Certificados de *zero-knowledge* para software propietario!
- Desarrollar estándares para buena práctica de certificación pública

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores
- Certificados de *zero-knowledge* para software propietario!
- Desarrollar estándares para buena práctica de certificación pública
- Definir un fragmento formal del lenguaje natural (domain specific) para cerrar la brecha entre especificaciones formales y especificaciones técnicas (public certification)

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores
- Certificados de *zero-knowledge* para software propietario!
- Desarrollar estándares para buena práctica de certificación pública
- Definir un fragmento formal del lenguaje natural (domain specific) para cerrar la brecha entre especificaciones formales y especificaciones técnicas (public certification)
- Impartir cursos de técnicas de verificación/certificación (y consulting)

Actividades de nuestro laboratorio

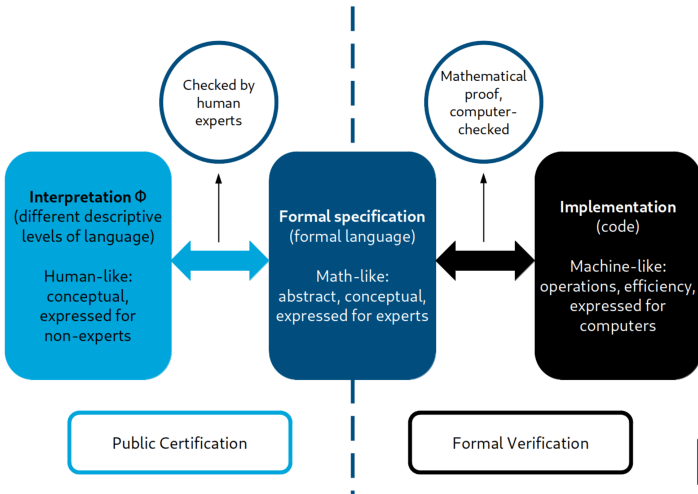
- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores
- Certificados de *zero-knowledge* para software propietario!
- Desarrollar estándares para buena práctica de certificación pública
- Definir un fragmento formal del lenguaje natural (domain specific) para cerrar la brecha entre especificaciones formales y especificaciones técnicas (public certification)
- Impartir cursos de técnicas de verificación/certificación (y consulting)
- Estudiar el impacto de las técnicas de verificación formal en la sociedad y los principios éticos/legales

Actividades de nuestro laboratorio

- Desarrollar certificados explicativos usando
 - ontologías y constructores de la granularidad adecuada para definir un lenguaje semi-formal
 - conversión de lambda-terms a equivalentes que solo usan estas ontologías y constructores
- Certificados de *zero-knowledge* para software propietario!
- Desarrollar estándares para buena práctica de certificación pública
- Definir un fragmento formal del lenguaje natural (domain specific) para cerrar la brecha entre especificaciones formales y especificaciones técnicas (public certification)
- Impartir cursos de técnicas de verificación/certificación (y consulting)
- Estudiar el impacto de las técnicas de verificación formal en la sociedad y los principios éticos/legales
- etc.

Mensaje para llevar a casa: sí se puede erradicar errores

algunas letras muy pequeñas



Muchas gracias

