FLAVOR: A FORMAL LANGUAGE FOR A POSTERIORI VERIFICATION OF LEGAL RULES

Romuald THION, Daniel LE MÉTAYER

UNIVERSITÉ LYON 1, LIRIS/INRIA GRENOBLE – RHÔNE-ALPES

IEEE International Symposium on Policies for Distributed Systems and Networks

Outline

FLAVOR: A FORMAL LANGUAGE FOR A POSTERIORI VERIFICATION OF LEGAL RULES

LICIT research team at INRIA

Legal Issues in Communication and Information Technologies

Computer science

(as seen by lawyers?)

Law

(as seen by scientists?)

Motivations

Examples of legal rules *(from the CS literature)*

- US Patriot Act [Giblin et al., 2005]
- Anti money-laundering [Liu et al., 2007]
- Health Insurance Portability and Accountability Act [Barth et al., 2006]
- Children' Online Privacy Protection Act [Barth et al., 2006]
- Gramm-Leach-Bliley Act [Barth et al., 2006]
- The Fair Credit Reporting Act [Johnson and Grandison, 2007]
- Airport regulations [Delahaye et al., 2006]
- U.S. Food and Drug Administration [Dinesh et al., 2008]

Motivations

Legal rules in IT systems

- Different sources (e.g., national, international, contracts. . .)
- Different objectives (e.g., business, privacy, security, crime. . .)
- Possibly very high stakes (e.g., financial losses, lawsuits, disrepute. . .)

How to manage and monitor legal rules in IT systems?

Toward a “compliance system”!

Contribution

A Formal Language for A posteriori Verification Of legal Rules

FLAVOR: key design choices

- Formal semantics
- Captures patterns of legal rules
- Oriented toward *a posteriori* verification
  - before: static analysis
  - while: monitoring
  - after: audit

Syntax

Excerpt of a business agreement

1. Within two weeks after receipt of the Software, *Customer* shall pay to *Supplier* the amount of twenty thousand Euros.
2. The payment of any additional service by *Customer* shall be due within four weeks after receipt of a valid invoice for the service.
3. In case of late payment, *Customer* shall pay, in addition to the due amount, a penalty of 5% of this amount.

Syntax

Characteristics of legal rules

- Conditional activation (*e.g., on receipt of an invoice*)
- Context (*e.g., invoice amount*)
- Deontic and temporal modalities (*e.g., must . . . within . . .*)
- Contrary to duty (*e.g., in case of a breach*)

FLAVOR is a domain specific language for legal rules which captures those constructors

Formal syntax

$$\mathcal{L} ::= \oplus\langle\rho,\delta\rangle \mid \ominus\langle\rho,\delta\rangle \mid \langle\rho,\delta\rangle \rightsquigarrow \phi \mid \langle\rho,\delta\rangle \dot{\rightsquigarrow} \phi \mid \psi \rhd \phi \mid \psi \wedge \phi$$

Informal semantics

ρ, δ	atomic properties (pattern matching on events)
$\oplus\langle\rho,\delta\rangle$	ought to do ρ before δ occurs
$\ominus\langle\rho,\delta\rangle$	ought not to do ρ until δ occurs
$\langle\rho,\delta\rangle \rightsquigarrow \phi$	for each ρ until δ, ϕ have to be satisfied
$\langle\rho,\delta\rangle \dot{\rightsquigarrow} \phi$	if ρ occurs before δ, then ϕ have to be satisfied
$\psi \rhd \phi$	if ψ is breached, then ϕ have to be satisfied

Semantics

Semantic function

$$[\![\psi]\!]_f : (E^\star \times \mathbb{N}) \to (\mathbb{B} \times \mathbb{N})_\perp$$

Given formula ψ and environment[a] f, produces a function $[\![\psi]\!]_f$

- from a trace ($\sigma \in E^\star$) and a point ($i \in \mathbb{N}$)
- tells whether the formula ψ, under environment f, is
  - satisfied at point j $(\mathbf{tt}, j)$
  - breached at point j $(\mathbf{ff}, j)$
  - pending ($\perp$)

[a] mapping from variables to values

Semantics

Obligation

$$[\![\oplus\langle\rho,\delta\rangle]\!]_f(\sigma,i) \begin{cases} (\mathbf{ff},i) & \text{if } \delta \text{ matches } \sigma(i) \\ (\mathbf{tt},i) & \text{if } \rho \text{ matches } \sigma(i) \\ [\![\oplus\langle\rho,\delta\rangle]\!]_f\ (\sigma,i+1) & \text{otherwise} \end{cases}$$

Prohibition

$$[\![\ominus\langle\rho,\delta\rangle]\!]_f(\sigma,i) \begin{cases} (\mathbf{tt},i) & \text{if } \delta \text{ matches } \sigma(i) \\ (\mathbf{ff},i) & \text{if } \rho \text{ matches } \sigma(i) \\ [\![\ominus\langle\rho,\delta\rangle]\!]_f\ (\sigma,i+1) & \text{otherwise} \end{cases}$$

Deadline takes precedence. $\oplus$ and $\ominus$ have dual behaviours.

Semantics

Conjunction

$$[\![\psi \wedge \phi]\!]_f(\sigma, i) = [\![\psi]\!]_f\ (\sigma, i) \sqcap [\![\phi]\!]_f\ (\sigma, i)$$

Both ψ and ϕ have to be satisfied.

Unique trigger

$$[\![\langle \rho, \delta \rangle \dot{\rightsquigarrow} \phi]\!]_f(\sigma, i) \begin{cases} (\mathbf{tt}, i) & \text{if } \delta \text{ matches } \sigma(i) \\ [\![\phi]\!]_{f'}\ (\sigma, i+1) & \text{if } \rho \text{ matches } \sigma(i) \\ [\![\langle \rho, \delta \rangle \dot{\rightsquigarrow} \phi]\!]_f\ (\sigma, i+1) & \text{otherwise} \end{cases}$$

If δ happens, the rule have reached its deadline. If ρ happens, then evaluates ϕ instanciated with environment updated.

Semantics

Multiple triggers

$$[\![\langle \rho, \delta \rangle \rightsquigarrow \phi]\!]_f(\sigma, i)$$

$$\begin{cases} (\mathbf{tt}, i) & \text{if } \delta \text{ matches } \sigma(i) \\ [\![\phi]\!]_{f'}(\sigma, i+1) \sqcap [\![\langle \rho, \delta \rangle \rightsquigarrow \phi]\!]_f(\sigma, i+1) & \text{if } \rho \text{ matches } \sigma(i) \\ [\![\langle \rho, \delta \rangle \rightsquigarrow \phi]\!]_f\ (\sigma, i+1) & \text{otherwise} \end{cases}$$

If ρ happens, then evaluates ϕ instanciated with environment updated and continues to evaluate the whole rule $\langle \rho, \delta \rangle \rightsquigarrow \phi$ (until some δ occurs).

Semantics

Contrary to duty

$$[\![\psi \succcurlyeq \phi]\!]_f(\sigma, i) \begin{cases} (\mathbf{tt}, j) & \text{if } [\![\psi]\!]_f\ (\sigma, i) = (\mathbf{tt}, j) \\ [\![\phi]\!]_f\ (\sigma, j) & \text{if } [\![\psi]\!]_f\ (\sigma, i) = (\mathbf{ff}, j) \\ \bot & \text{otherwise} \end{cases}$$

If ψ is satisfied, then the whole rule $\psi \succcurlyeq \phi$ is satisfied. If ψ is breached, then returns the result of the evaluation of ϕ.

Some properties

Impossible deadlines

If $\forall e \in E^\star$, e never matches δ, then:

- $\oplus\langle\rho, \delta\rangle$ is unbreachable
- $\ominus\langle\rho, \delta\rangle$ is unsatisfiable

Strength properties

ϕ is stronger than ψ ($\phi \succcurlyeq \psi$)

- $\phi \wedge \psi \succcurlyeq \phi$ and $\phi \wedge \psi \succcurlyeq \psi$
- $\langle\rho, \delta\rangle \rightsquigarrow \phi \succcurlyeq \langle\rho, \delta\rangle \dot{\rightsquigarrow} \phi$
- $\phi \succcurlyeq (\phi \succ \psi)$

Example analysis

Within two weeks after receipt of the Software, Customer shall pay to Supplier the amount of twenty thousand Euros. [. . .] In case of late payment, Customer shall pay, in addition to the due amount, a penalty of 5% of this amount

Formal expression in FLAVOR

1. Receipt of the software ($soft^{T_d}_{S \to C}$) triggers once ($\dot{\rightsquigarrow}$)
2. Customer must ($\oplus$) pay within two weeks ($T_a \geq T_d + 14$)
3. If customer does not pay in due time ($\gg$), then he is charged 5%

Formal expression in FLAVOR

$$\langle soft^{T_d}_{S \to C}, \mathbf{ff} \rangle \dot{\rightsquigarrow} \quad (1)$$

$$(\oplus \langle pay(20,000)_{C \to S}, x^{T_a} \wedge (T_a \geq T_d + 14) \rangle \gtrdot \quad (2)$$

$$\oplus \langle pay(21,000)_{C \to S}, \mathbf{ff} \rangle) \quad (3)$$

According to properties

- *pay 20.000 within 14d* $\gtrdot$ (*pay 20.000 within 14d pay* $\gtrdot$ *(eventually) pay 21.000*)
- Alternative obligation has no deadline . . .
- . . . so if the customer never pays, the rule will not be breached!

The rule is way too much permissive!

Related work

Modal logics

$$\mathcal{L} ::= p \in P \mid \phi \wedge \psi \mid \phi \vee \psi \mid \phi \Rightarrow \psi \mid \neg\psi \mid \Diamond\psi \mid \Box\psi$$

Paradoxes of deontic logic

- Material implication (Good Samaritan paradox)
- Disjunction introduction (Ross's paradox, free choice permission paradox)
- Contradictory statements (Sartre's dilemma, Chisholm's paradox)

Related work

Related language (Schneider *et al.*)

- Alternative Time Logic, Propositional Dynamic Logic, modal μ-calculus,
- With restrictions to prevent paradoxes
- Conflict detection and static analysis

Differences with FLAVOR

- Instantiation of rules
- Contrary to duty are not attached to atoms
- Uniformity of action/events

Conclusion

Essence of legal rules

- Atoms: testifiers for fulfillment or violation
- Instantiation of rules based on context
- Sanction/reparation connective

Technical aspects

- Denotational semantics
- Close to modal logics on finite linear trace
- Turned into and interpreter (written in Haskell)

Future work

Expressivity of FLAVOR

- A dual constructor $\phi \lessdot \psi$: *if* ϕ *satisfied, then* ψ
- Comparison with (real-time) temporal logics
- Is the language a closed subset of LTL formulae?
- What is the expressivity of the fragment?

Intuition

- FLAVOR captures some patterns of LTL [Dwyer et al., 1999]
- *holds weakly / holds strongly* semantics for LTL [Eisner et al., 2003]:
  - $\mathrm{rw}^+ : \mathcal{L} \to$ LTL, tells if satisfied
  - $\mathrm{rw}^- : \mathcal{L} \to$ LTL, tells if breached
  - $\mathrm{rw}^+(\phi) \vee \mathrm{rw}^-(\phi) = \mathbf{ff}$, pending

Thanks for your attention!

Questions?

Barth, A., Datta, A., Mitchell, J. C., and Nissenbaum, H. (2006).

Privacy and contextual integrity: Framework and applications.
In *SP '06: Proceedings of the 2006 IEEE Symposium on Security and Privacy*, pages 184–198, Washington, DC, USA. IEEE Computer Society.

Delahaye, D., Étienne, J.-F., and Donzeau-Gouge, V. V. (2006).

Reasoning about airport security regulations using the focal environment.
In *ISOLA '06: Proceedings of the Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (isola 2006)*, pages 45–52, Washington, DC, USA. IEEE Computer Society.

Dinesh, N., Joshi, A. K., Lee, I., and Sokolsky, O. (2008).

Checking traces for regulatory conformance.
In Leucker, M., editor, *RV*, volume 5289 of *Lecture Notes in Computer Science*, pages 86–103. Springer.

Dwyer, M. B., Avrunin, G. S., and Corbett, J. C. (1999).

Patterns in property specifications for finite-state verification.
In *ICSE '99: Proceedings of the 21st international conference on Software engineering*, pages 411–420, New York, NY, USA. ACM.

Eisner, C., Fisman, D., Havlicek, J., Lustig, Y., McIsaac, A., and Campenhout, D. V. (2003).

Reasoning with temporal logic on truncated paths.
In Jr., W. A. H. and Somenzi, F., editors, *CAV*, volume 2725 of *Lecture Notes in Computer Science*, pages 27–39. Springer.

Giblin, C., Liu, A. Y., and Samuel Müllerand Birgit Pfitzmann, X. Z. (2005).

Regulations expressed as logical models (REALM).
In IOS Press, A., editor, *Proceedings of the 18th Annual Conference on Legal Knowledge and Information Systems (JURIX 2005)*, pages 37–48.

Johnson, C. M. and Grandison, T. (2007).

Compliance with data protection laws using hippocratic database active enforcement and auditing.
IBM Systems Journal, 46(2):255–264.

Liu, Y., Müller, S., and Xu, K. (2007).

A static compliance-checking framework for business process models.
IBM Systems Journal, 46(2):335–362.