MASTER
Managing Assurance,
Security and Trust for sERvices

Inria
INVENTEURS DU MONDE NUMÉRIQUE

Better enforce than verify!
How to ensure compliance of business processes at runtime

Nataliia Bielova[1,3]

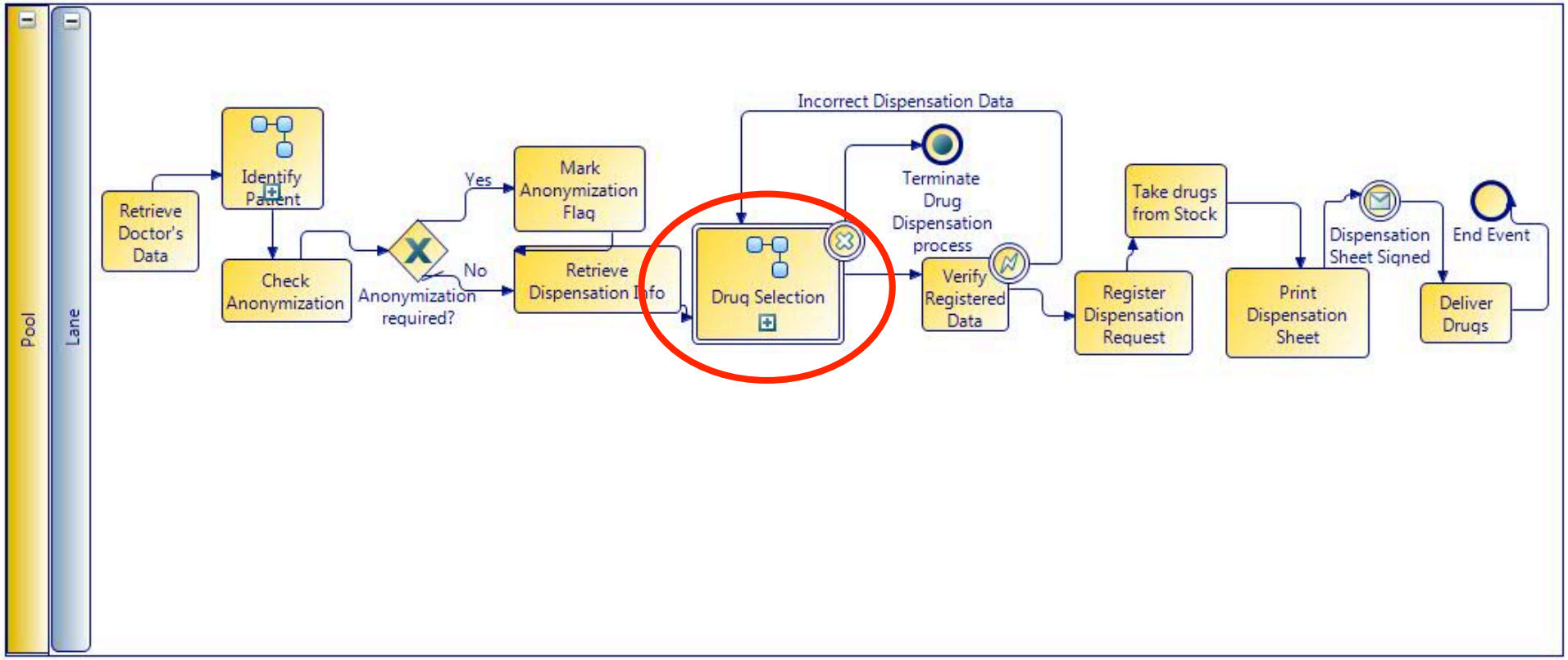
joint work with Fabio Massacci[1]

and reality checks by Andrea Micheletti[2]

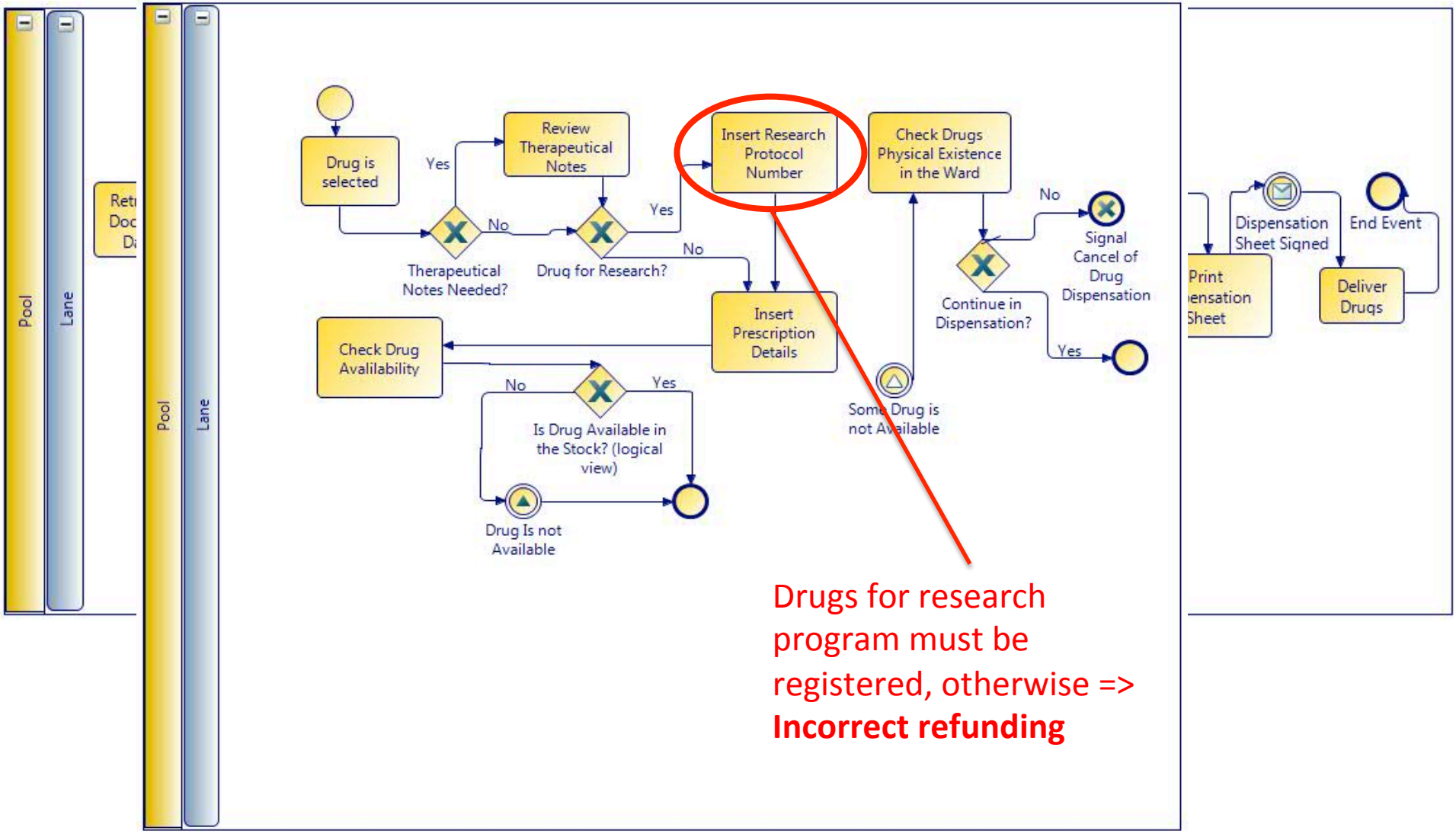
[1]University of Trento, Italy

[2]Hospital San Raffaele, Italy

BPMN: drug dispensation from Hospital



BPMN: drug dispensation from Hospital

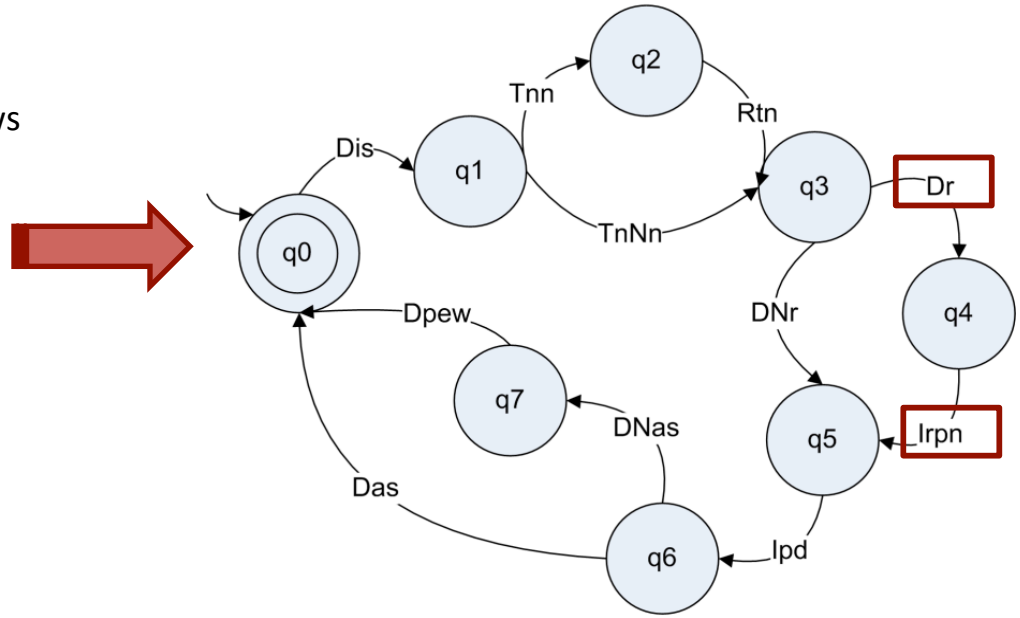


Hospital security policy

Security policy

- The doctor selects the drug.
- If the therapeutical notes needed ⇒ doctor reviews them.
- If patient is using prescribed drug for the research program purposes ⇒ doctor inserts the research protocol number.
- Doctor inserts all the prescription details.
- If drug is available in the stock ⇒ doctor takes it.
- If drug is available in the ward ⇒ doctor takes it.

Security policy as FSA



Executions in Drug Selection Process are iterations

Good iteration: starts in q0 and finishes in q0

Bad iteration: "Drug is for research but doctor "forgot" to insert research protocol number"

Runtime Enforcement Theory

- **Security policy P –** compliant executions
  - Only security properties – evaluated on a trace

- **Runtime enforcer –** controls the execution and ensures compliance with P

  - Security Automaton [Schneider TISSEC'00]
    - halts the execution when violation is detected
  - Edit Automaton [Bauer, Ligatti, Walker IJIS'05, TISSEC'09]
    - **modifies the execution** when violation is detected

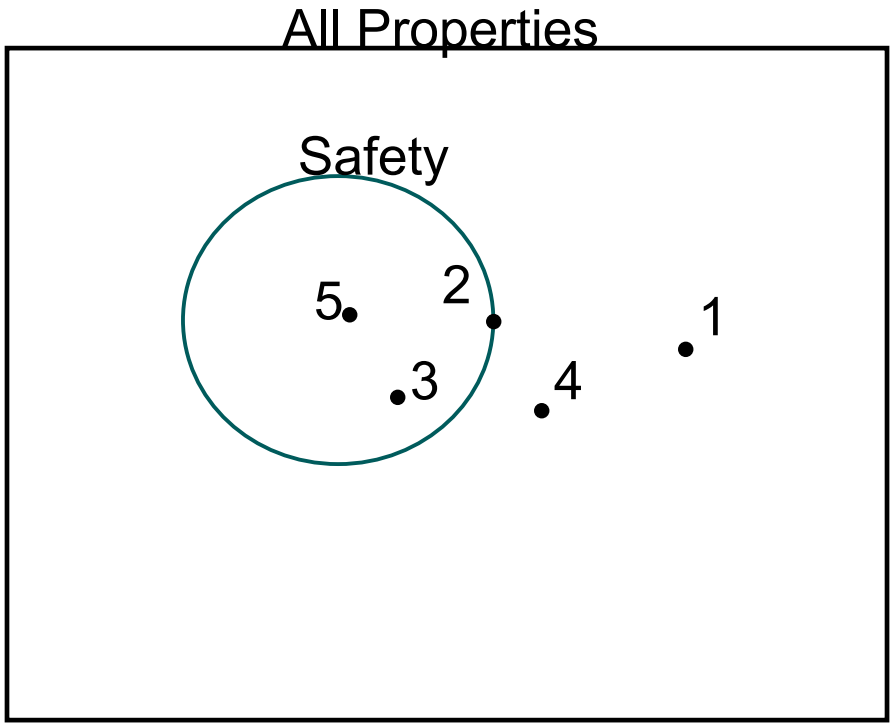
Use this theory in practice?

- We want to enforce Hospital security policy…
  - **Q1:** Is our policy **enforceable** by these mechanisms?
  - **Q2:** How **to construct** an enforcement mechanism for our policy?
  - **Q3:** What **formal guarantees** do we get?

Safety property

[Schneider, TISSEC'05]

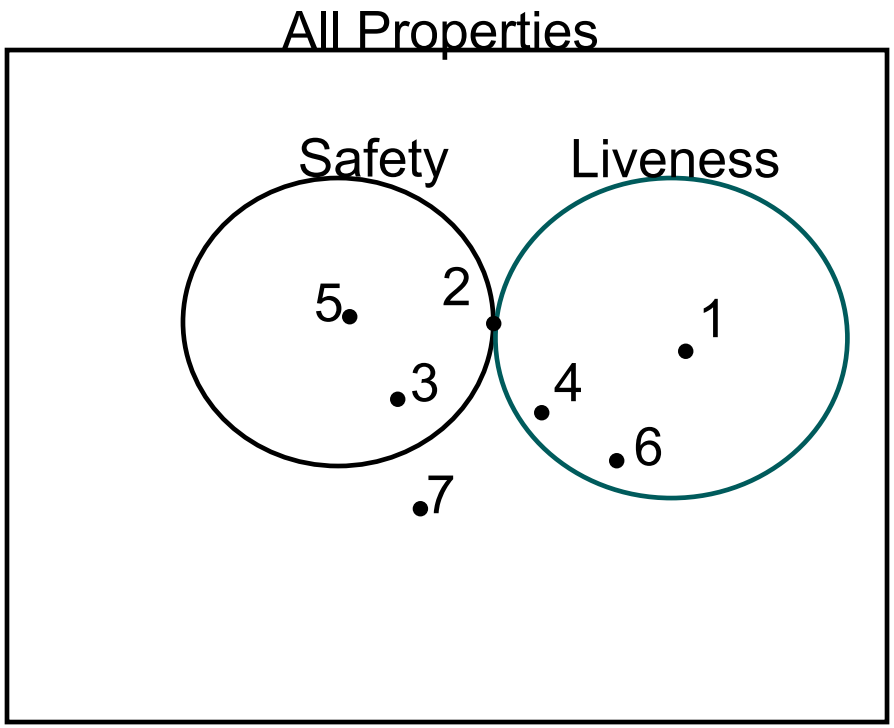
"Doctors are not allowed to make mistakes"



1 Nontermination
2 Trivial
3 Stack inspection
4 Eventually audits
5 All sequences with fixed length

Liveness property

"A doctor can always add the right actions to a finite execution"

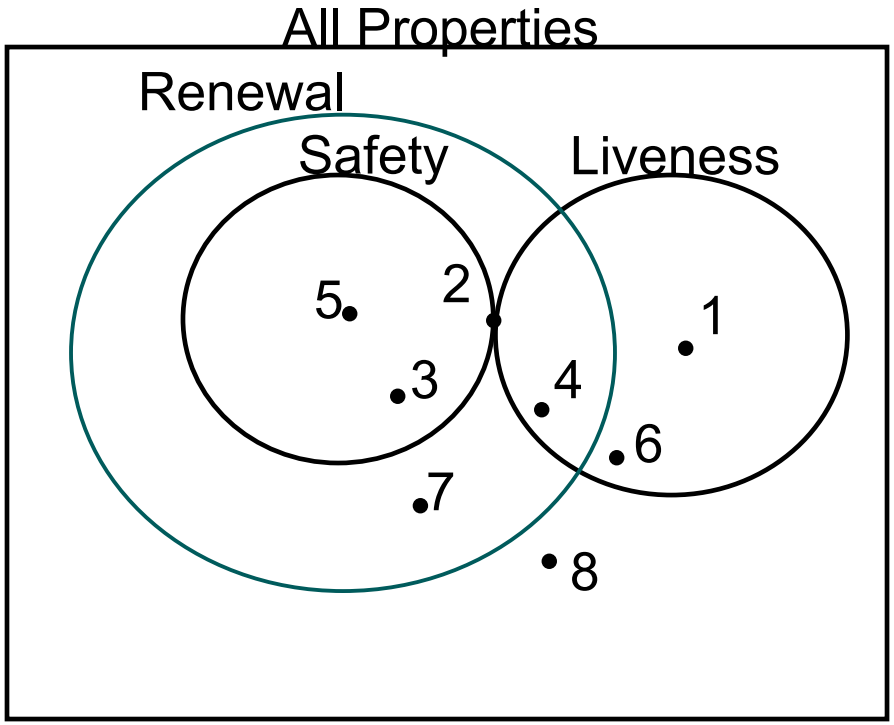


1 Nontermination
2 Trivial
3 Stack inspection
4 Eventually audits
5 All sequences with fixed length
6 Termination
7 Transaction property σ^{∞}

Renewal property

[Ligatti, Bauer, Walker IJIS'05, TISSEC'09]

"Good infinite executions could have had finitely many bad parts"

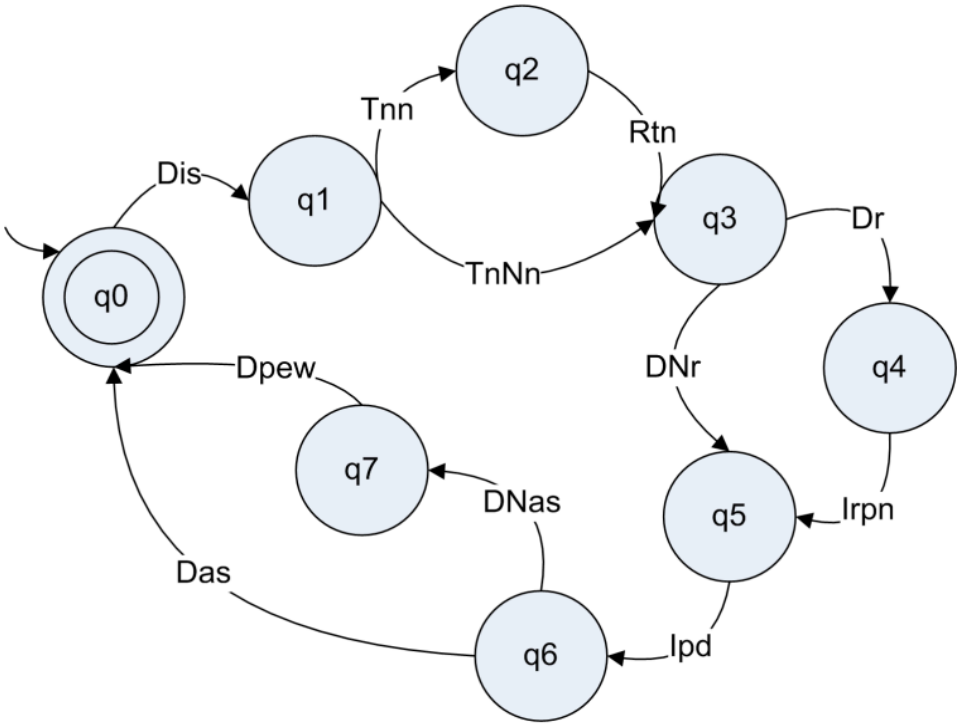


1 Nontermination
2 Trivial
3 Stack inspection
4 Eventually audits
5 All sequences with fixed length
6 Termination
7 Transaction property σ^{∞}
8 Termination + file access control

Hospital security policy

Good infinite execution: sequence of iterations

Finite number of bad prefixes: while interation is not finished

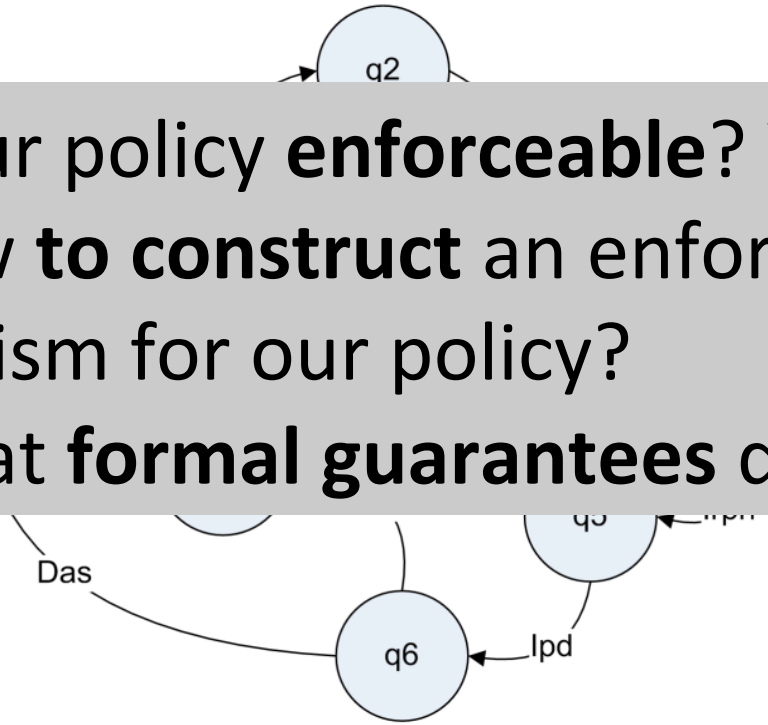


=> Hospital security policy is a (particular kind of) renewal property!

Hospital security policy

Good infinite execution: sequence of iterations

Finite number of bad prefixes: while interation is not finished



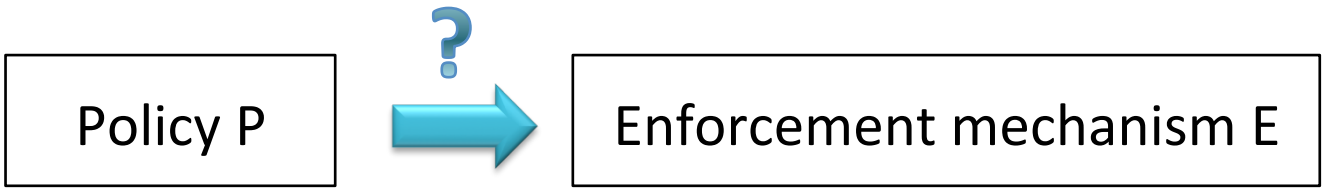
✓ **Q1:** Is our policy **enforceable**? YES

Q2: How **to construct** an enforcement mechanism for our policy?

Q3: What **formal guarantees** do we get?

=> Hospital security policy is a (particular kind of) renewal property!

How to construct an Enforcement mechanism?



- Requirements on Enforcement mechanism E:
  - **Soundness**: everything it outputs is secure

$$\forall \sigma \in \Sigma^{\infty} : P(E(\sigma))$$

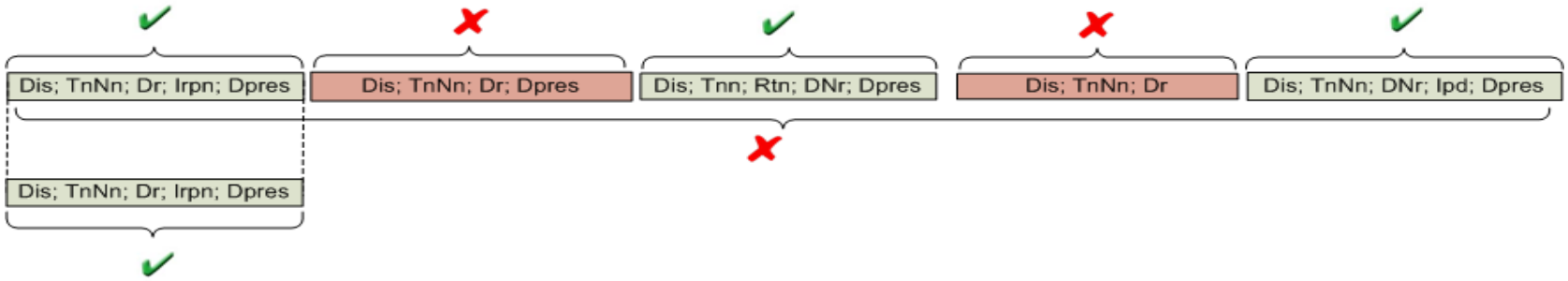
  - **Transparency**: all secure inputs are not modified

$$\forall \sigma \in \Sigma^{\infty} : P(\sigma) \Rightarrow E(\sigma) = \sigma$$

Construction 1: E outputs the longest valid prefix

Occasionally doctors make mistakes and forget to insert Research protocol number

- **Longest valid prefix**: "all the iterations before the doctor makes a mistake"



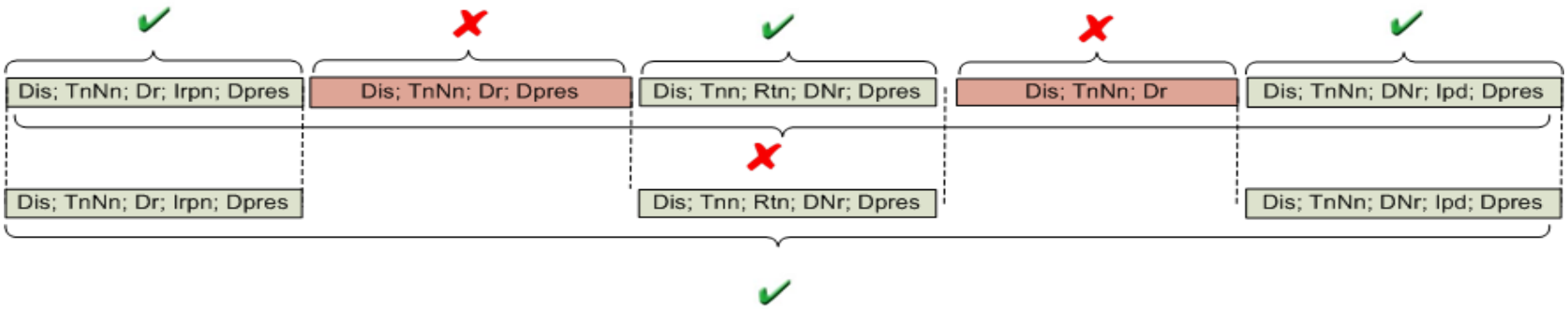
Formal guarantees:

- Soundness: $\forall \sigma \in \Sigma^{\infty} : P(E(\sigma))$
- Transparency: $\forall \sigma \in \Sigma^{\infty} : P(\sigma) \Rightarrow E(\sigma) = \sigma$

Construction 2: E suppresses invalid parts

Occasionally doctors make mistakes and forget to insert Research protocol number

- **Iterative suppression automaton** [Bielova, Massacci, Micheletti NordSec'09, JCS'11]



Formal guarantees:

- Soundness: $\forall \sigma \in \Sigma^{\infty} : P(E(\sigma))$
- Transparency: $\forall \sigma \in \Sigma^{\infty} : P(\sigma) \Rightarrow E(\sigma) = \sigma$

Construction 2: E suppresses invalid parts

Occasionally doctors make mistakes and forget to insert Research protocol number

- **Iterative suppression automaton** [Bielova, Massacci, Micheletti NordSec'09, JCS'11]

✓ **Q2:** How **to construct** an enforcement mechanism for our policy?
2 WAYS…

Q3: What **formal guarantees** do we get?
THE SAME

- Transparency: $\forall \sigma \in \Sigma^{\infty} : P(\sigma) \Rightarrow E(\sigma) = \sigma$

Are the enforcement mechanisms different?

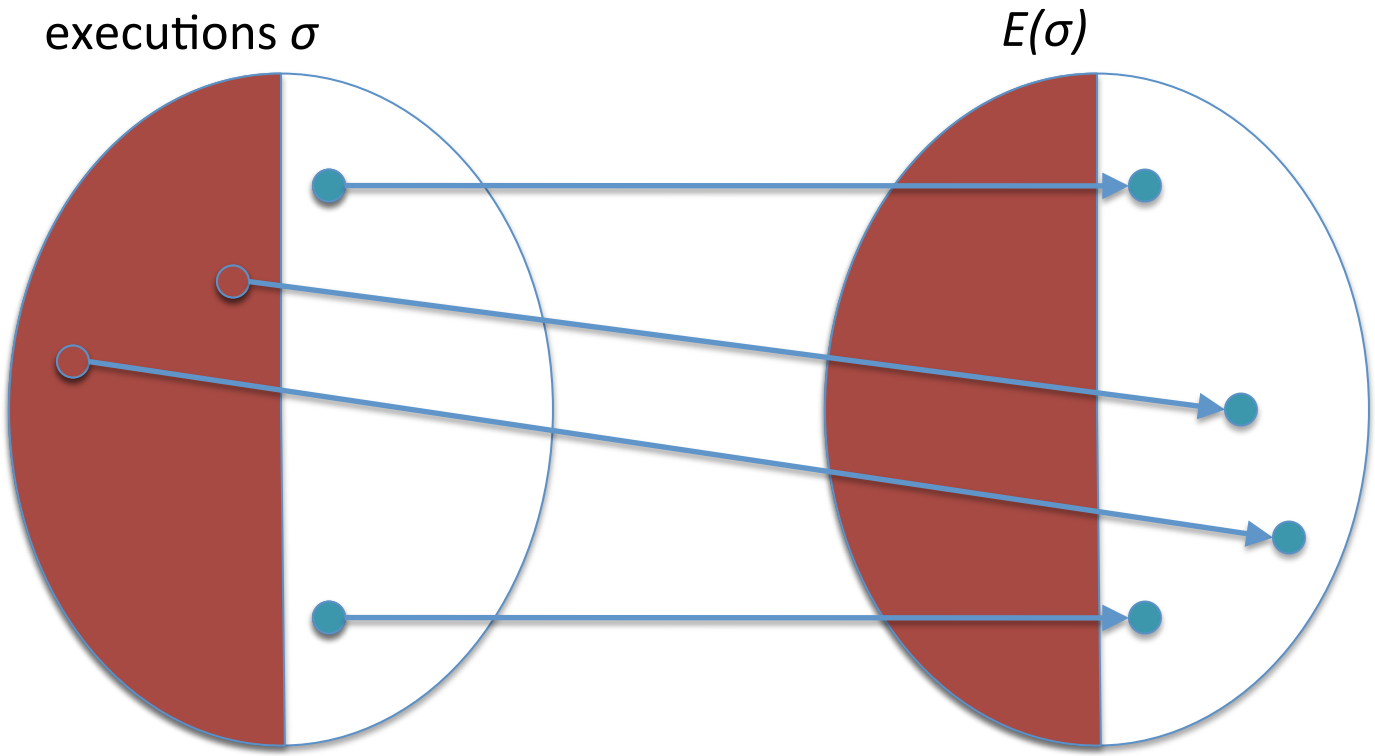
- Key idea: the mechanism is a trace transformer
  - It is evident the two mechanisms are different!
- Key requirements:
  - Soundness – both are sound
  - Transparency – both are transparent

- But they are different!
  - Hospital San Raffaele would not definitely pay the same money for both of them

- What distinguishes enforcement mechanisms is not what happens when traces are good (because nothing should happen) but *what precisely happens* when the sequence does not respect the policy

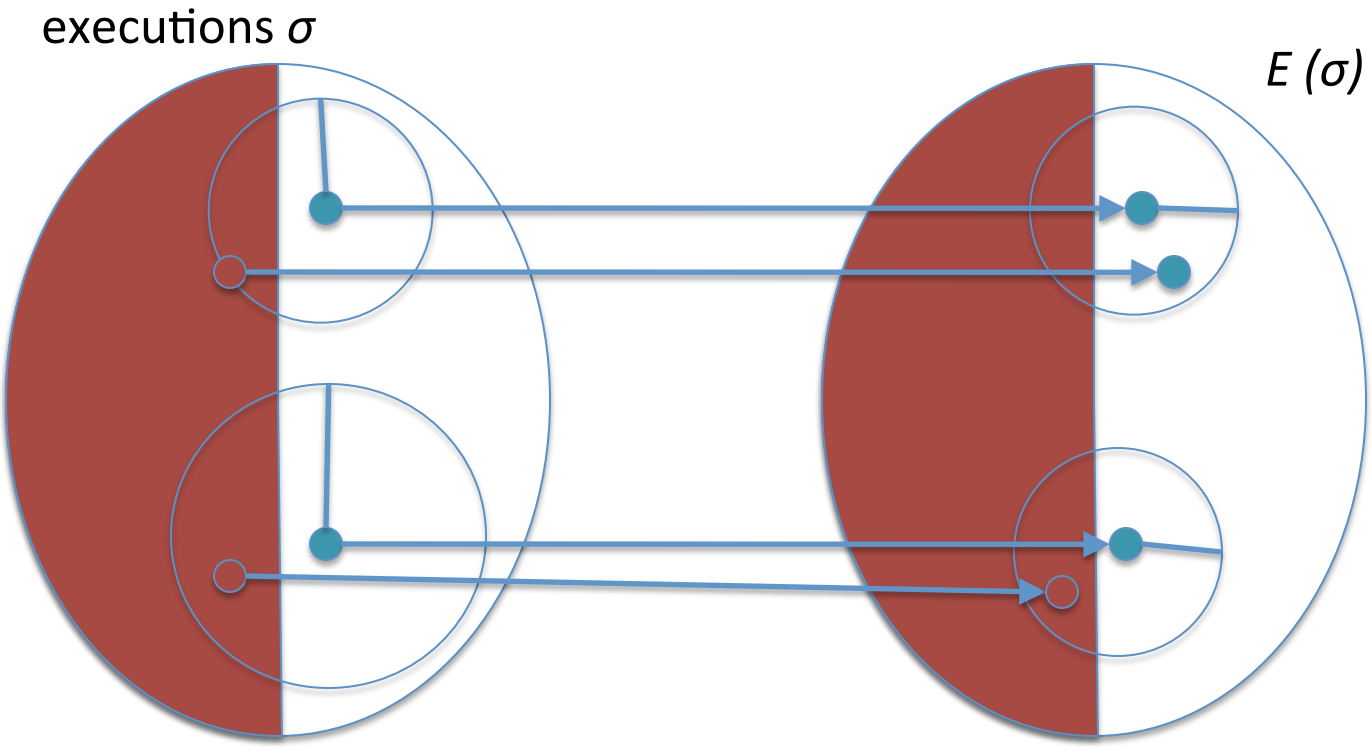
Soundness and transparency are not sufficient

- **Soundness**: for valid and invalid input
- **Transparency**: for valid input



Predictability [Bielova, Massacci ESSoS'11]

- **"No surprises":** for invalid input
  - input doesn't respect the policy => output is close to good input

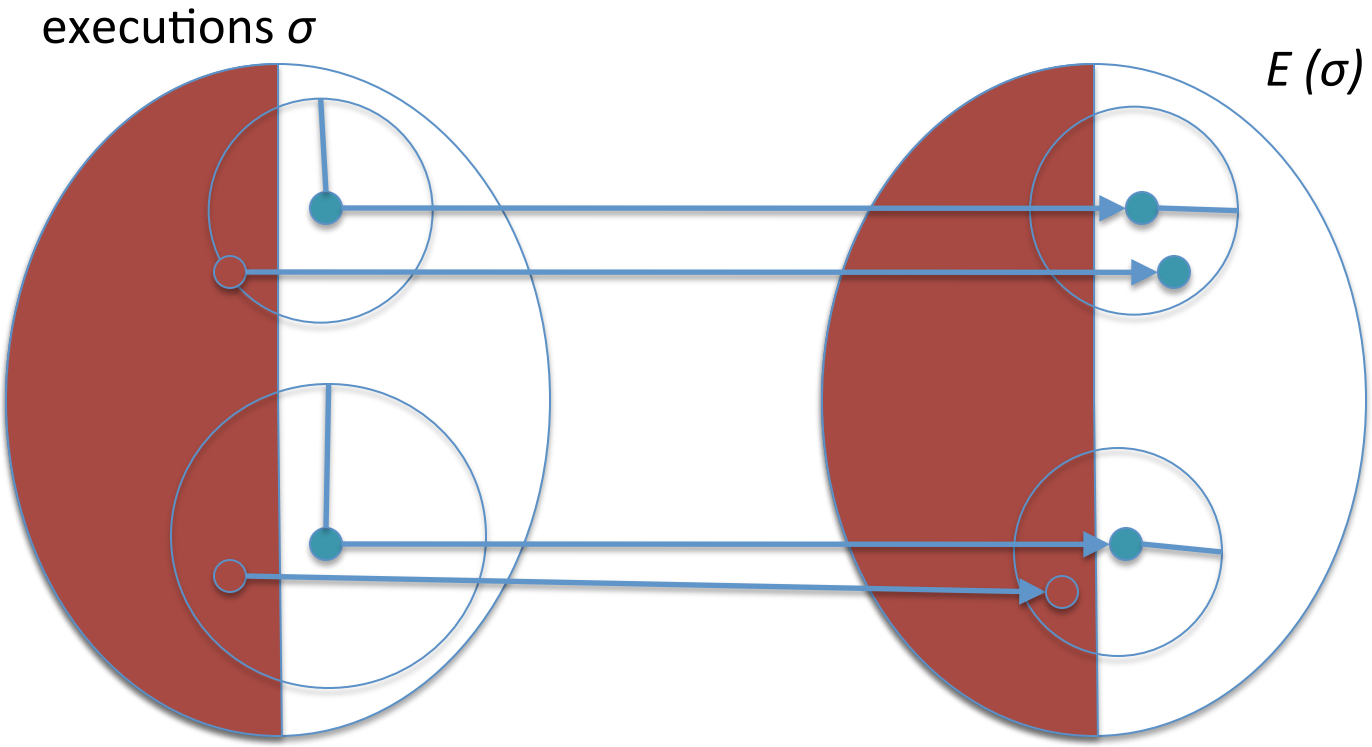


Predictability [Bielova, Massacci ESSoS'11]

- E is **predictable** within *k* if for every trace $\sigma_P \in P$

$$\forall \nu \geq k : \exists \delta > 0 : \forall \sigma \in \Sigma^* :$$

$$(d(\sigma, \sigma_P) \leq \delta \Rightarrow d'(E(\sigma), E(\sigma_P)) \leq \nu)$$



How to define the distance?

- Suppressing distance $d_S(\sigma, \sigma')$
  - **Reality check OK**: suppress some bad actions, bring back to the stable state
- Replacing distance $d_R(\sigma, \sigma')$
  - **Reality check OK**: correct small errors, don't change the protocol
- No way to transform σ into σ' $\rightarrow$ $d_S(\sigma, \sigma') = \infty$ / $d_R(\sigma, \sigma') = \infty$
- Levenshtein distance $d_L(\sigma, \sigma')$ (suppresion, replacement, insertion)
  - **Reality check NOT OK**: insertion of new actions is not acceptable, medical and legal consequences

Suppression distance for drug dispensation

- σ_P: Good
- σ : Dis; Good
- $d_S(\sigma, \sigma_P) = 1$

- E1 outputs the longest valid prefix
  - $E1(\sigma) = \bullet$
  - $\mathbf{d_S(E1(\sigma), E1(\sigma_P))} = d_S(\bullet, \text{Good}) = \mathbf{\infty}$

- E2 suppresses bad parts of execution
  - $E2(\sigma) = \text{Good}$
  - $\mathbf{d_S(E2(\sigma), E2(\sigma_P))} = d_S(\text{Good}, \text{Good}) = \mathbf{0}$

- **Result**: Suppression distance distinguishes E1 and E2

Enforcement of Error-tolerant policies

- Mechanism E_P enforces the policy P

- Mechanism $E_{Tolerate(P, k)}$ enforces the policy P and tolerates up to *k* errors

What kind of errors/deviations?

- Venial errors
  - Doctor forgot to Review therapeutical notes (no Rtn)
  - Close therapeutical notes window instead (Ctw)
  - Only a limited number of times, e.g., *k* times per day

- Amendable errors
  - Doctor did not Insert research protocol number (no Irpn, Cpw instead)
  - → the whole reimbursement process can go wrong!
  - Should be "corrected" by Inserting special number for Audit (InA)

Construction 3: E tolerates up to *k* errors
[Bielova, Massacci POLICY'11]

- Example
  - σ : Dis; Tnn; Ctw; Dr; Cpw; Dpres – 1 venial and 1 amendable
  - E(σ): Dis; Tnn; Ctw; Dr; InA; Dpres – 2 venial

Formal guarantees:

- **Transparency**: $\forall \sigma \in \Sigma^* : P(\sigma) \Rightarrow E(\sigma) = \sigma$
- **Predictability for *k*:** $\forall\, \nu \geq k : \exists \delta > 0 : \forall \sigma \in \Sigma^* :$

$$(d^{va}(\sigma, \sigma_P) \leq \delta \;\Rightarrow\; d^{v}(E(\sigma), E(\sigma_P)) \leq \nu)$$

- where
  - $d^{va}(\sigma, \sigma')$ – number venial and amendable errors
  - $d^{v}(\sigma, \sigma')$ - number of venial errors
  - amendable errors get transformed into venial errors or fixed

Conclusions

Q1: Is our policy **enforceable** by these mechanisms?

- Hospital policy can be enforced at runtime

Q2: How **to construct** an enforcement mechanism for our policy?

1. Longest valid prefix
2. Suppress bad iterations
3. Tolerate up to k errors

Q3: What **formal guarantees** do we get?

- Soundness and transparency are not sufficient!
  - what distinguishes enforcement mechanisms in reality is *what precisely happens* when the input does **not** respect the policy
- New notion: **Predictability**
  - your input doesn't respect the policy => your output is close to your input

New developments in the field…

- Extensions to the theory
  - Non-controllable actions [Basin, Juge, Klaedtke, Zalinescu POST'12, TISSEC'13]
  - Mandatory results [Ligatti, Reddy ESORICS'10]
  - Target aware [Mallios, Bauer, Kaynar, Ligatti STM'12]
  - Corrective Enforcement [Khoury, Tawbi FAST'10, TISSEC'12]

- Inexact enforcement: quantitative approach
  - Assigned cost to enforcement actions [Drábik, Martinelli, Morisset STM'12]
  - Probabilistic cost enforcement [Mallios, Bauer, Kaynar, Martinelli, Morisset STM'13]

You've got a paper ready in 26 days?

International Symposium on Engineering Secure Software and Systems

February 26-28, 2014

Munich, Germany

- Program co-chairs:
  - Jan Juerjens (Technical University Dortmund, DE)
  - Frank Piessens (KU Leuven, BE)
- Important Dates
  - Abstract submission: September 6, 2013
  - Paper submission: September 13, 2013
- https://distrinet.cs.kuleuven.be/events/essos/2014/

Publications

- [BM-IJIS'11] N. Bielova and F. Massacci. Do you really mean what you actually enforced? Edit Automata revisited. *IJIS'11.*
- [BM-JCS'12] N. Bielova and F. Massacci. Iterative Enforcement by Suppression: Towards Practical Enforcement Theories. *JCS'12.*
- [BM-FAST'08] N. Bielova and F. Massacci. Do you really mean what you actually enforced? Edit Automata revisited. *FAST'08.*
- [BMM-NordSec'09] N. Bielova, F. Massacci and A. Micheletti. Towards Practical Enforcement Theories. *NordSec'09.*
- [BM-ESSoS'11] N. Bielova and F. Massacci. Predictability of Enforcement. *ESSoS'11.*
- [BM-POLICY'11] N. Bielova and F. Massacci. Computer-Aided Generation of Enforcement Mechanisms for Error-Tolerant Policies. *POLICY'11.*