

Luchando contra Errores, Fallas y Chantadas para construir Software Confiable

Pedro R. D'Argenio

Grupo de Sistemas Confiables

Universidad Nacional de Córdoba – CONICET (AR)



Luchando contra Errores, Fallas y Chantadas para construir **Software** Confiable

Pedro R. D'Argenio

Grupo de Sistemas Confiables
Universidad Nacional de Córdoba – CONICET (AR)



Luchando contra Errores, Fallas y Chantadas para construir Sistemas Confiables

Pedro R. D'Argenio

Grupo de Sistemas Confiables
Universidad Nacional de Córdoba – CONICET (AR)



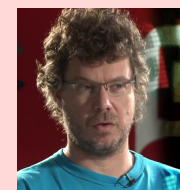
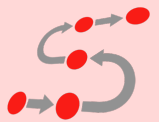
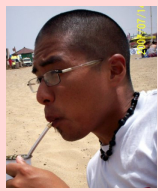
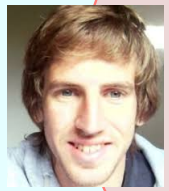




UNIVERSITY OF TRENTO



Fm Formal Methods & Tools



Inria



TECHNISCHE UNIVERSITÄT DRESDEN



RWTH AACHEN UNIVERSITY



imdea software



UNC

El software parece hacer maravillas...

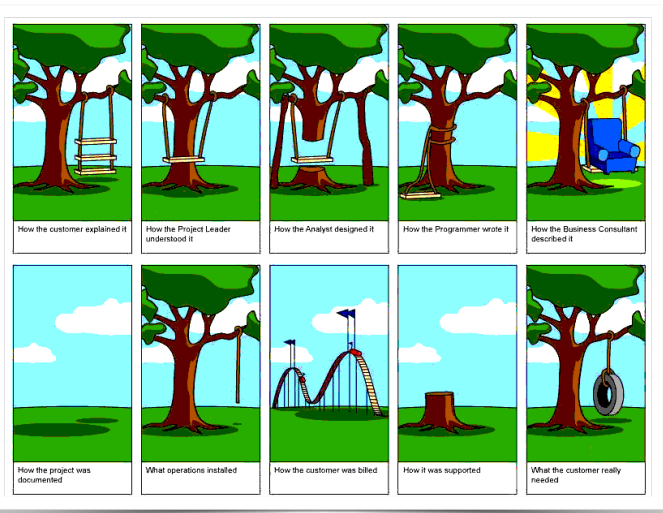


El software parece hacer maravillas...

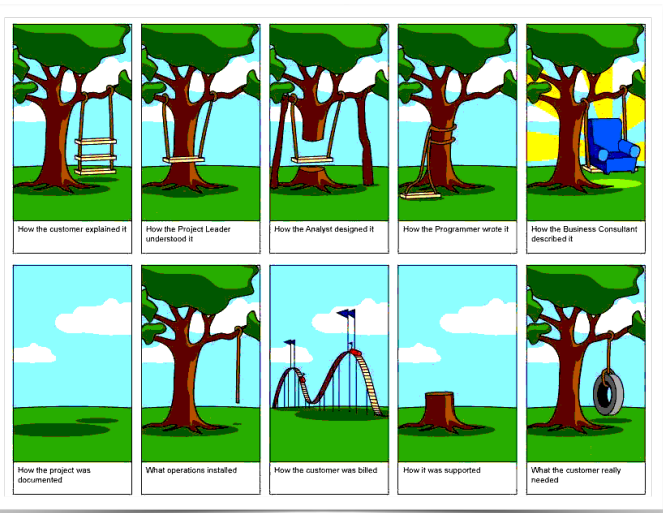
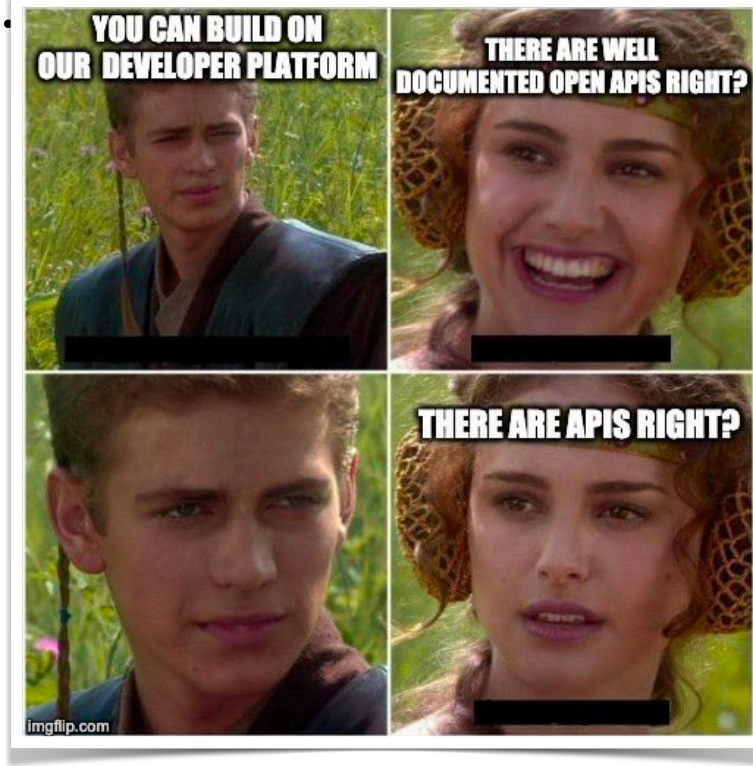


... pero debajo de esa cáscara de maravilla
sabemos muy bien que ...

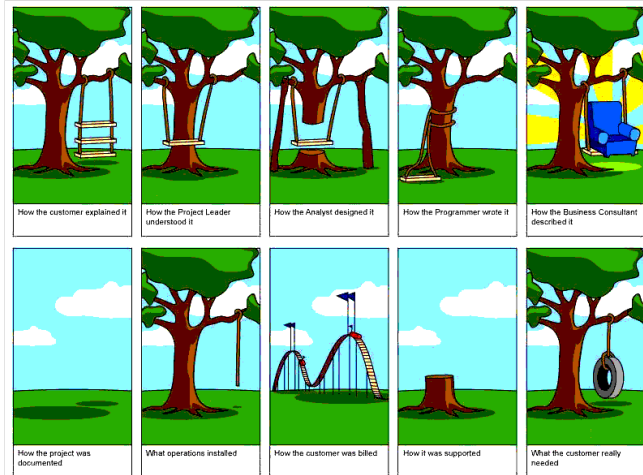
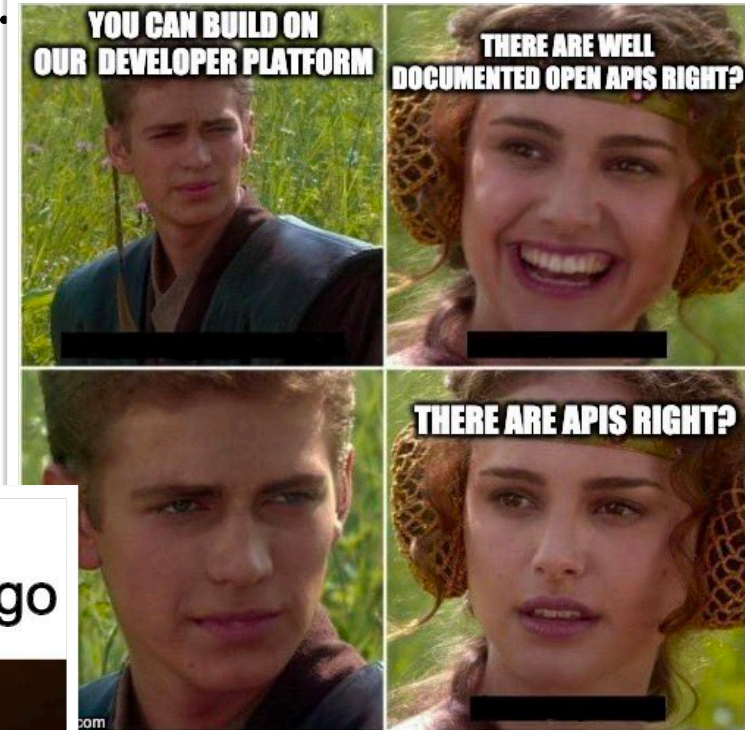
... pero debajo de esa cáscara de maravilla sabemos muy bien que ...



... pero debajo de esa cáscara de maravilla sabemos muy bien que .



... pero debajo de esa cáscara de maravilla sabemos muy bien que .



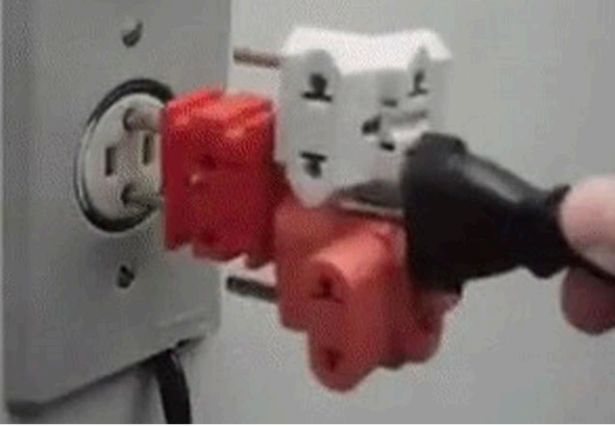
When you trying to look at the code you wrote a month ago

IT'S SOME KIND OF ELVISH

I CAN'T READ IT

... pero debajo de esa cáscara de maravilla sabemos muy bien que .

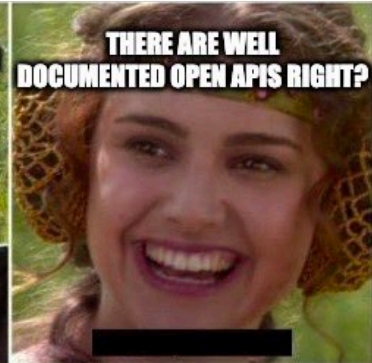
Modern software development



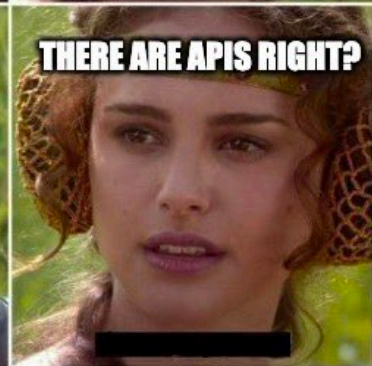
**YOU CAN BUILD ON
OUR DEVELOPER PLATFORM**



**THERE ARE WELL
DOCUMENTED OPEN APIS RIGHT?**

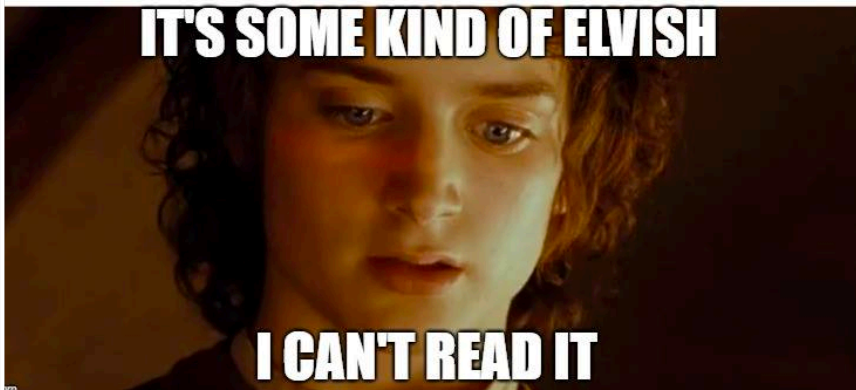


THERE ARE APIS RIGHT?

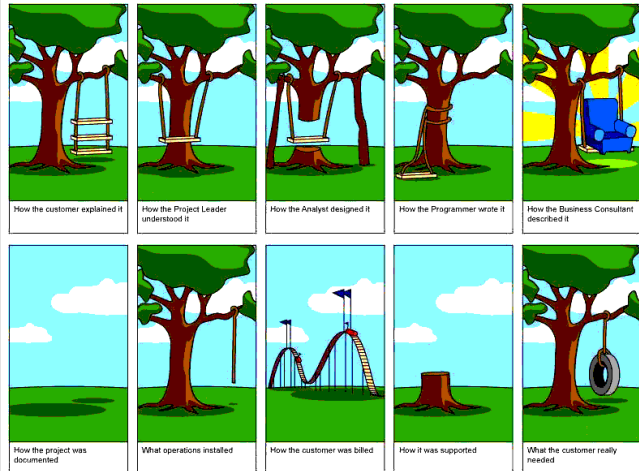


When you trying to look at
the code you wrote a month ago

IT'S SOME KIND OF ELVISH

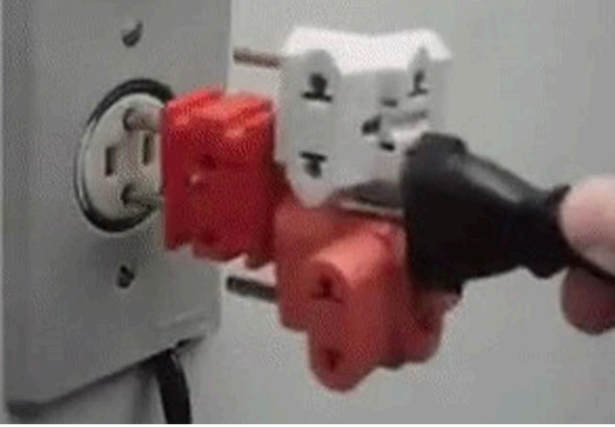


I CAN'T READ IT



... pero debajo de esa cáscara de maravilla sabemos muy bien que

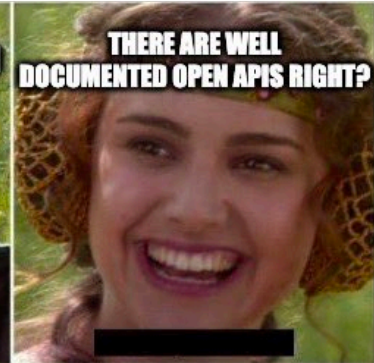
Modern software development



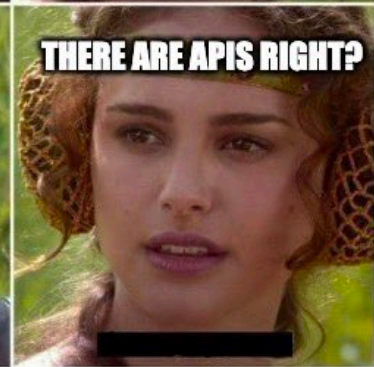
YOU CAN BUILD ON OUR DEVELOPER PLATFORM



THERE ARE WELL DOCUMENTED OPEN APIS RIGHT?

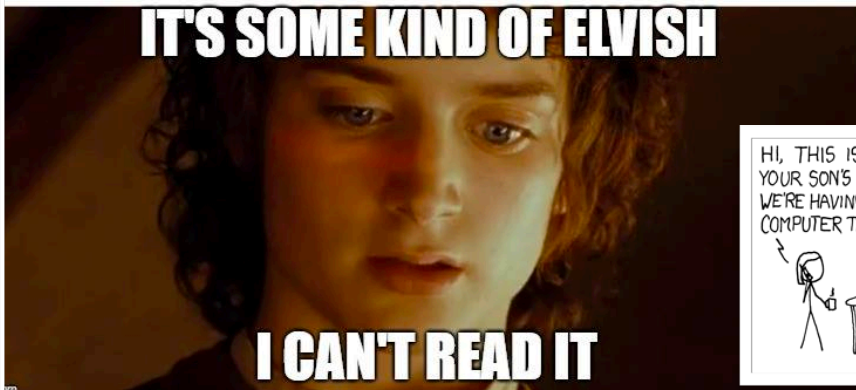


THERE ARE APIS RIGHT?

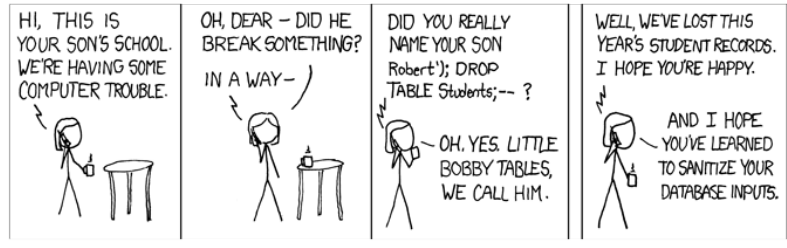
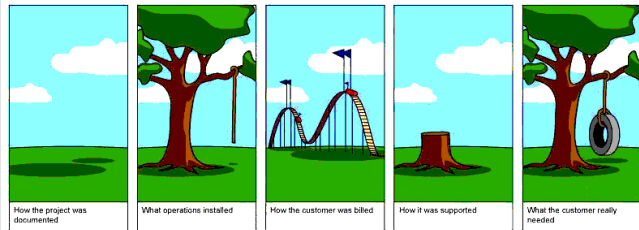
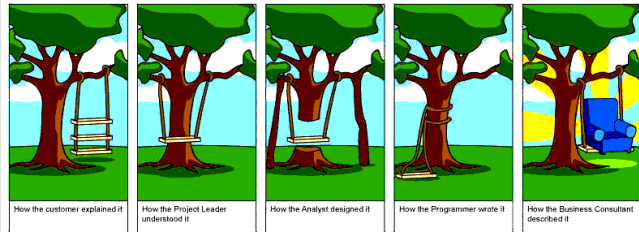


When you trying to look at the code you wrote a month ago

IT'S SOME KIND OF ELVISH



I CAN'T READ IT



... pero

maravilla

Modern software development



AGILE DEVELOPMENT

BECAUSE WHO NEEDS A PLAN

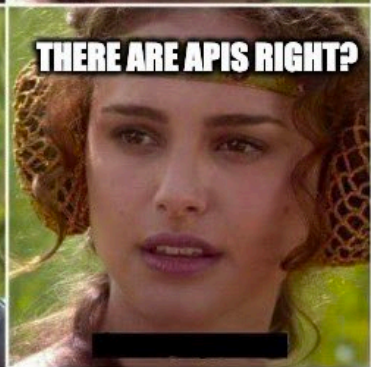
YOU CAN BUILD ON OUR DEVELOPER PLATFORM



THERE ARE WELL DOCUMENTED OPEN APIS RIGHT?

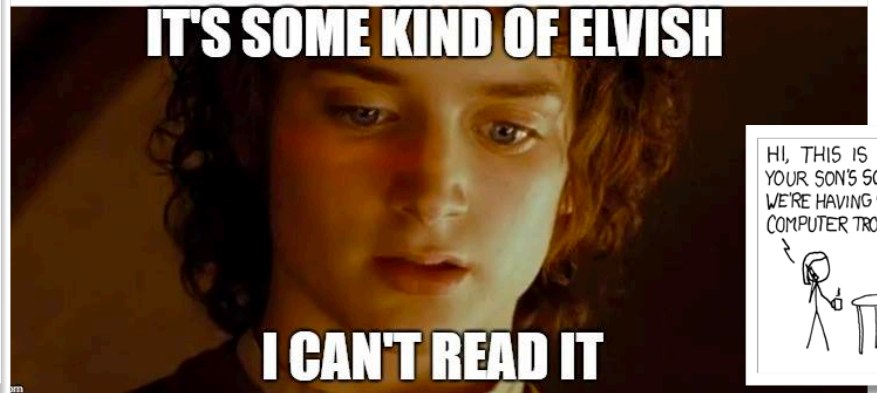


THERE ARE APIS RIGHT?

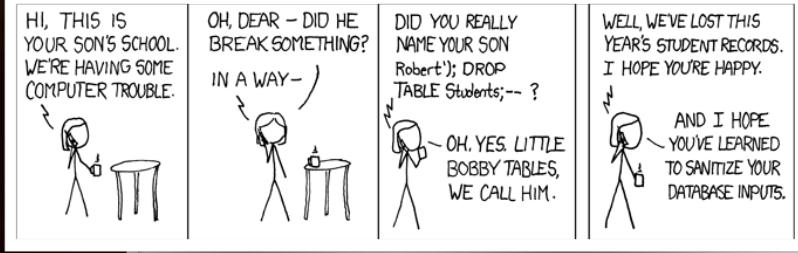
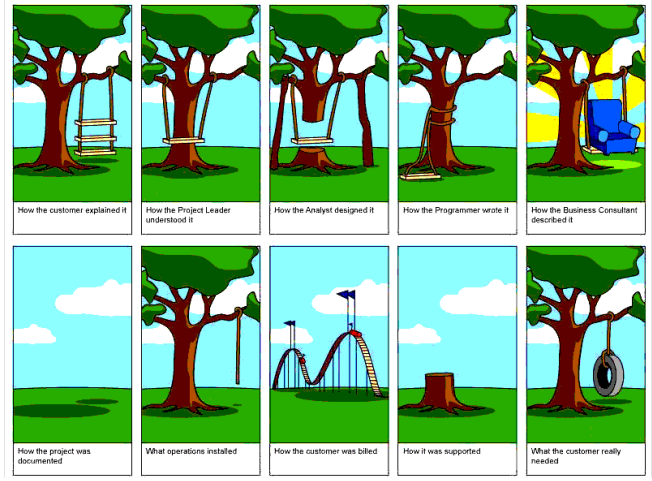


When you trying to look at the code you wrote a month ago

IT'S SOME KIND OF ELVISH



I CAN'T READ IT



... pero

AGILE DEVELOPMENT

maravilla

Modern software development

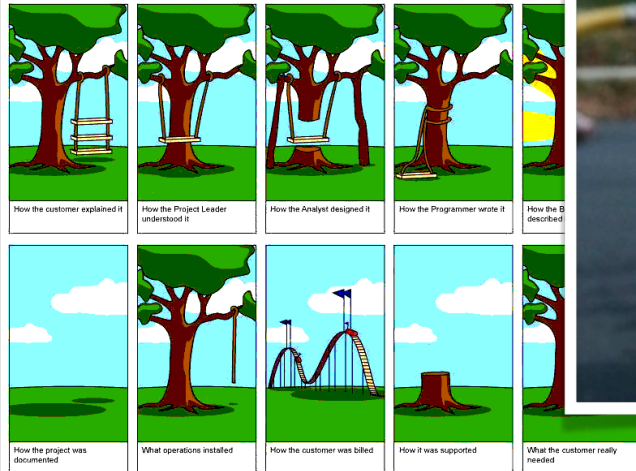
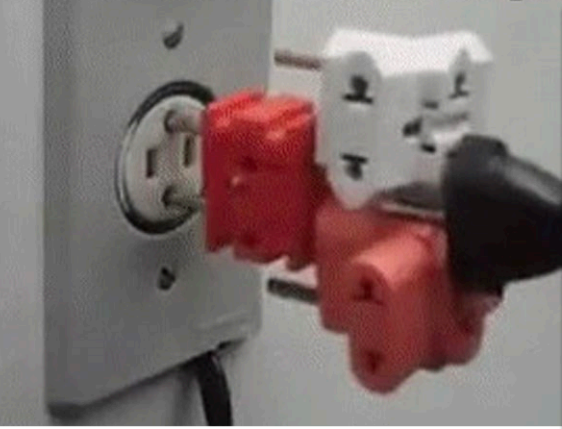
SOFTWARE DEVELOPMENT I



BIG FIRE

memegenerator.net

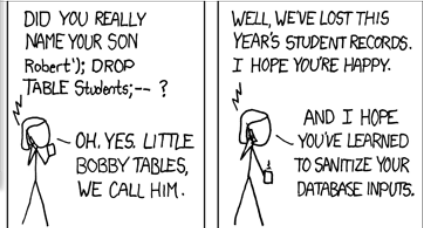
I CAN'T READ IT



YOU CAN BUILD ON OUR DEVELOPMENT PLATFORM

THERE ARE WELL DOCUMENTED OPEN APIS RIGHT?

THERE ARE APIS RIGHT?



¿Qué produce todo estos problemas?

- ❖ Errores en el desarrollo del software
- ❖ Fallas externas al software pero que son parte del sistema
(y son mal atendidas)
- ❖ Programación malintencionada

¿Qué produce todo estos problemas?

- ❖ Errores en el desarrollo del software

Bugs

- ❖ Fallas externas al software pero que son parte del sistema
(y son mal atendidas)
- ❖ Programación malintencionada

¿Qué produce todo estos problemas?

- ❖ Errores en el desarrollo del software
- ❖ Fallas externas al software pero que son parte del sistema
(y son mal atendidas)
- ❖ Programación malintencionada

MOCOS

¿Qué produce todo estos problemas?

- ❖ Errores en el desarrollo del software
- ❖ Fallas externas al software pero que son parte del sistema
(y son mal atendidas)
- ❖ Programación malintencionada

ERRORES

¿Qué produce todo estos problemas?

- ❖ Errores en el desarrollo del software
- ❖ Fallas externas al software pero que son parte del sistema
(y son mal atendidas)
- ❖ Programación malintencionada

ERRORES

FALLAS

¿Qué produce todo estos problemas?

- ❖ Errores en el desarrollo del software
- ❖ Fallas externas al software pero que son parte del sistema
(y son mal atendidas)
- ❖ Programación malintencionada

ERRORES

FALLAS

CHANTADAS

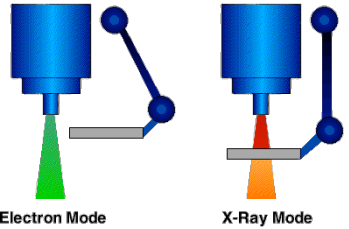
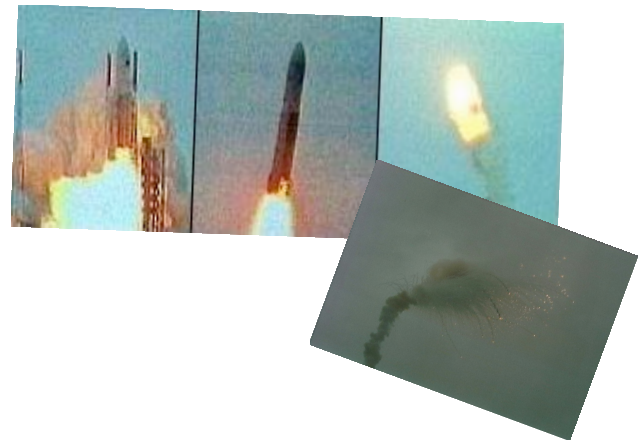
Errores

Errores famosos

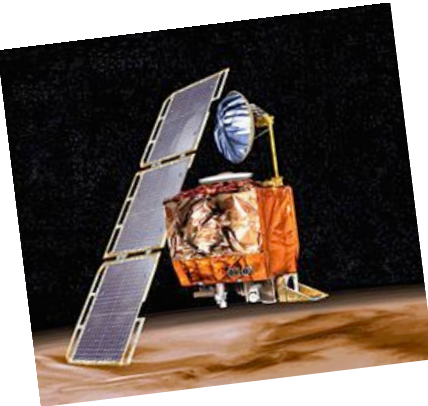


Pentium:
FDIV

Ariane 5:
64 bits fp
vs 16 bits int

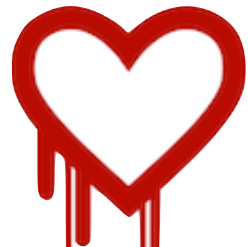
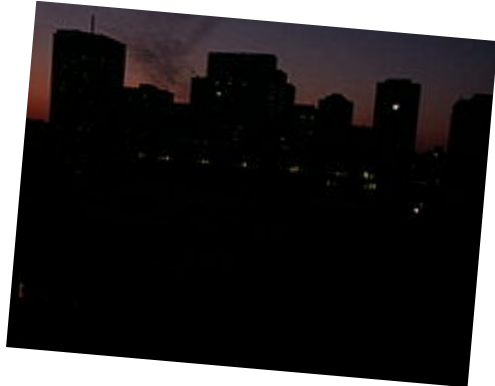


Therac-25:
Condición de
carrera



Mars Climate
Orbiter:
Métrico vs Imperial

Northeast blackout
en 2003:
Condición de carrera



Heartbleed:
Integridad/Confidencialidad

Más errores



911 blackout:
MAX value
reached

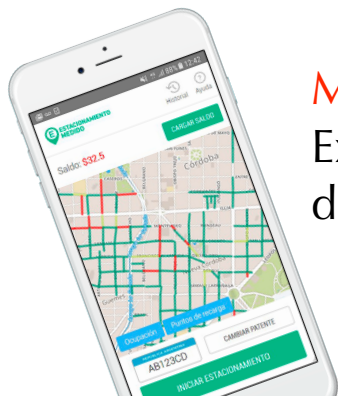
Nest Thermostat:
Drenado de
batería



Nissan airbag:
Sensado
incorrecto



Boeing 737 MAX 8:
Sensado incorrecto



Movypark:
Exposición de
datos personales

Tesla/Uber/Google
self-driving car:
aprendizaje con
limitaciones??



El problema de la corrección

Sistema \models Propiedad

El problema de la corrección

Sistema \models *Propiedad*

Usualmente una
abstracción que describe su
comportamiento

Describe lo que se espera
del sistema
(el criterio de corrección)

Model Checking

$\mathcal{M} \models \phi?$

```

int y1 = 0;
int y2 = 0;
short in_critical = 0;

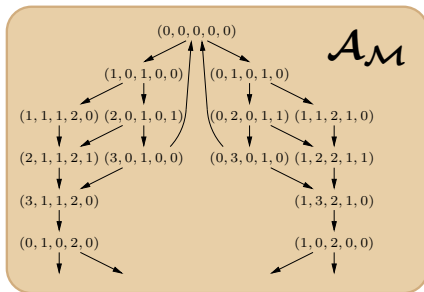
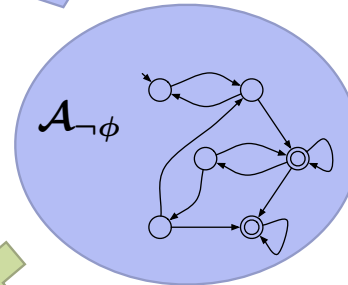
active proctype process_1() {
do
:: true ->
0:   y1 = y2+1;
1:   ((y2==0) || (y1<y2));
    in_critical++;
2:   in_critical--;
3:   y1 = 0;
od
}

active proctype process_2() {
do
:: true ->
0:   y2 = y1+1;
1:   ((y1==0) || (y2<y1));
    in_critical++;
2:   in_critical--;
3:   y2 = 0;
od
}
    
```

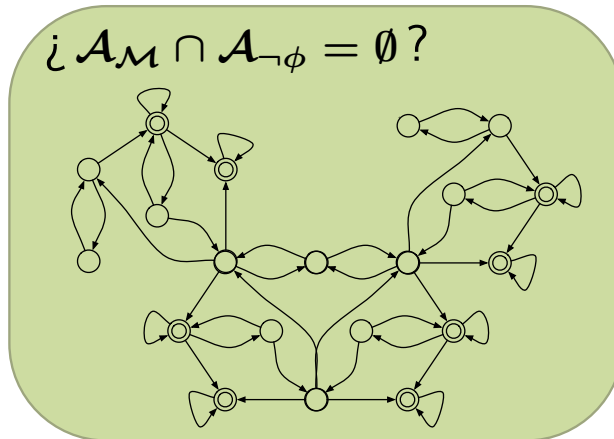
\mathcal{M}

$\phi : \square \diamond crit_1 \wedge \square \diamond crit_2$

El problema se reduce a análisis de grafos



$\mathcal{A}_M \cap \mathcal{A}_{\neg\phi} = \emptyset?$



Limitaciones del Model Checking

- ❖ Muchos algoritmos proponen (mejores) soluciones utilizando aleatoriedad.
- ❖ Leader Election Protocol en IEEE 1394 “Firewire”
- ❖ Binary Exponential Backoff en IEEE 802.3 “Ethernet”

Limitaciones del Model Checking

- ❖ Muchos algoritmos **proponen (mejores) soluciones** utilizando **aleatoriedad**.
 - ❖ Leader Election Protocol en IEEE 1394 “Firewire”
 - ❖ Binary Exponential Backoff en IEEE 802.3 “Ethernet”
- ❖ Muchas veces **no se puede establecer corrección** con una **lógica bivaluada**. Sin embargo la validez de la propiedad **puede cuantificarse probabilísticamente**.
 - ❖ Bounded Retransmission Protocol en Philips RC6
 - ❖ Binary Exponential Backoff en IEEE 802.3 “Ethernet”

Model Checking **Cuantitativo**

$?\mathcal{M} \models \phi?$

```

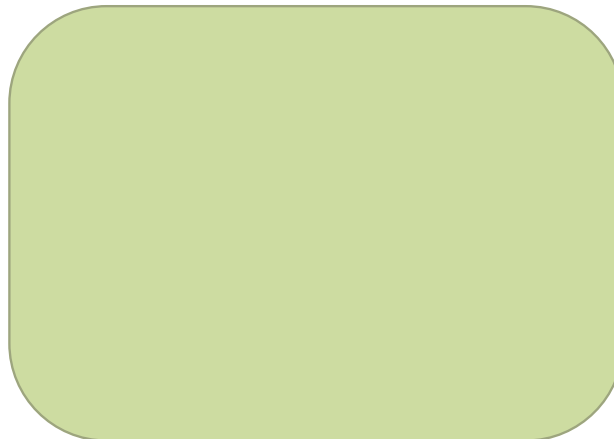
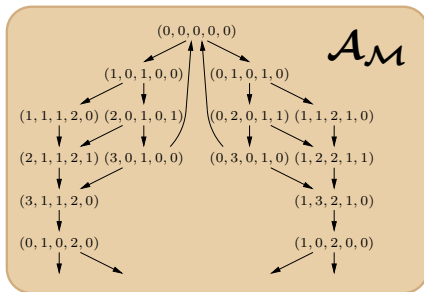
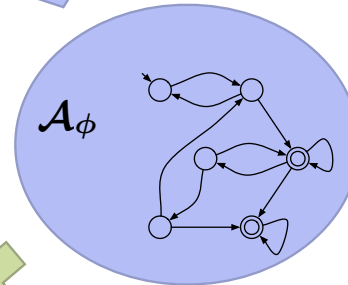
int y1 = 0;
int y2 = 0;
short in_critical = 0;

active proctype process_1() {
do
:: true ->
0:   y1 = y2+1;
1:   ((y2==0) || (y1<y2));
    in_critical++;
2:   in_critical--;
3:   y1 = 0;
od
}

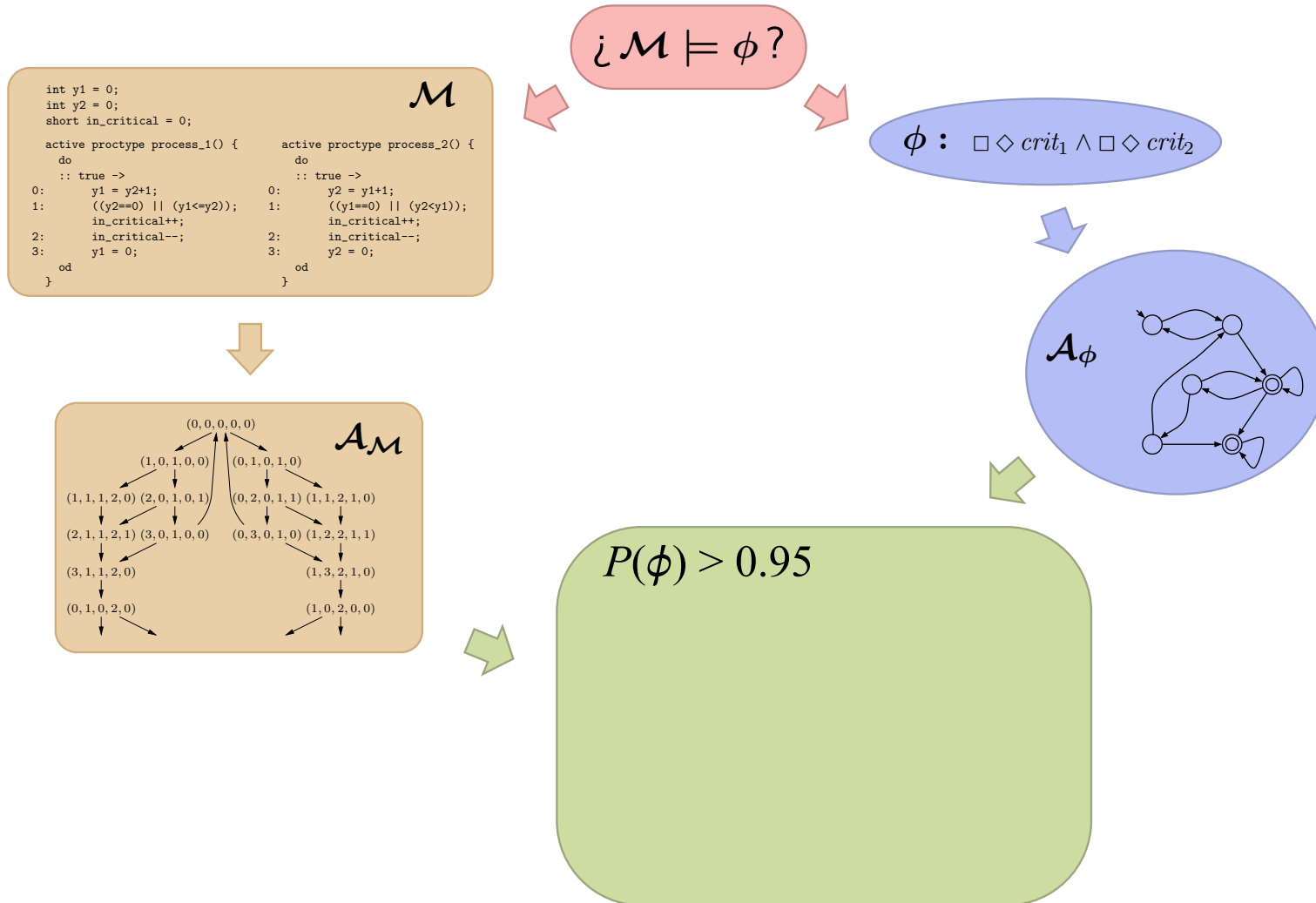
active proctype process_2() {
do
:: true ->
0:   y2 = y1+1;
1:   ((y1==0) || (y2<y1));
    in_critical++;
2:   in_critical--;
3:   y2 = 0;
od
}
    
```

\mathcal{M}

$\phi : \square \diamond crit_1 \wedge \square \diamond crit_2$



Model Checking **Cuantitativo**



Model Checking **Cuantitativo**

Incluye primitivas de aleatoriedad

```

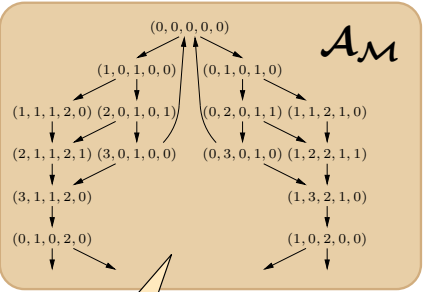
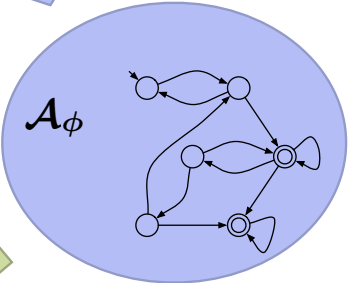
int y1 = 0;
int y2 = 0;
short in_critical = 0;

active proctype process_1() {
do
:: true ->
0:   y1 = y2+1;
1:   ((y2==0) || (y1<y2));
    in_critical++;
2:   in_critical--;
3:   y1 = 0;
od
}

active proctype process_2() {
do
:: true ->
0:   y2 = y1+1;
1:   ((y1==0) || (y2<y1));
    in_critical++;
2:   in_critical--;
3:   y2 = 0;
od
}
    
```

$¿\mathcal{M} \models \phi?$

$\phi : \square \diamond crit_1 \wedge \square \diamond crit_2$



Proceso de decisión de Markov

$P(\phi) > 0.95$

Model Checking **Cuantitativo**

Incluye primitivas de aleatoriedad

```

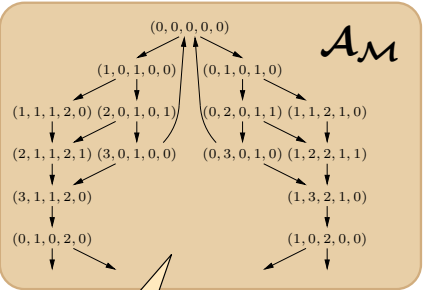
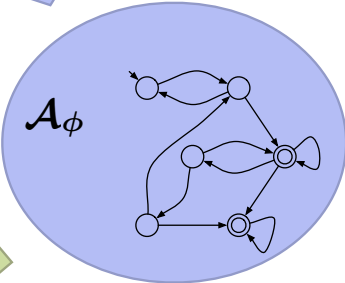
int y1 = 0;
int y2 = 0;
short in_critical = 0;

active proctype process_1() {
do
:: true ->
0:   y1 = y2+1;
1:   ((y2==0) || (y1<y2));
    in_critical++;
2:   in_critical--;
3:   y1 = 0;
od
}

active proctype process_2() {
do
:: true ->
0:   y2 = y1+1;
1:   ((y1==0) || (y2<y1));
    in_critical++;
2:   in_critical--;
3:   y2 = 0;
od
}
    
```

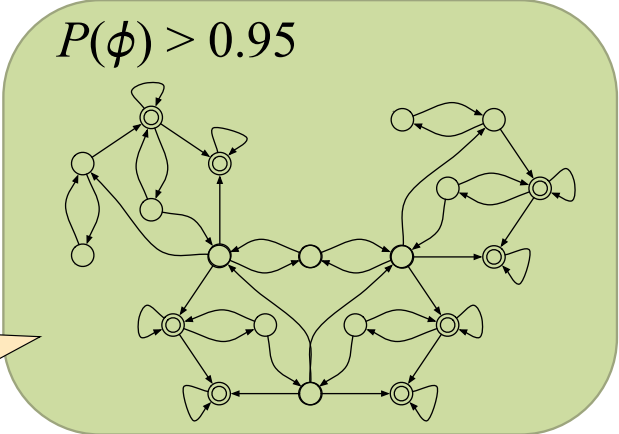
$¿\mathcal{M} \models \phi?$

$\phi : \square \diamond crit_1 \wedge \square \diamond crit_2$



Proceso de decisión de Markov

Proceso de decisión de Markov



Model Checking **Cuantitativo**

Incluye primitivas de aleatoriedad

```

M
int y1 = 0;
int y2 = 0;
short in_critical = 0;

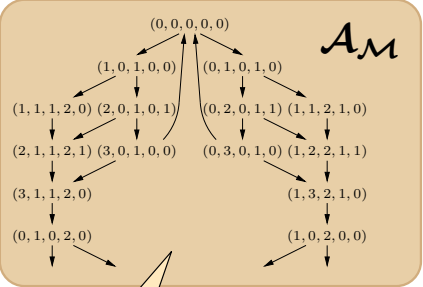
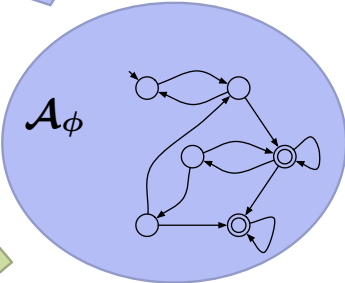
active proctype process_1() {
do
:: true ->
0:   y1 = y2+1;
1:   ((y2==0) || (y1<y2));
    in_critical++;
2:   in_critical--;
3:   y1 = 0;
od
}

active proctype process_2() {
do
:: true ->
0:   y2 = y1+1;
1:   ((y1==0) || (y2<y1));
    in_critical++;
2:   in_critical--;
3:   y2 = 0;
od
}
    
```

$\mathcal{M} \models \phi?$

$\phi : \square \diamond crit_1 \wedge \square \diamond crit_2$

Se reduce a un problema de optimización lineal



Proceso de decisión de Markov

$P(\phi) > 0.95$

$$x_s = \max_{a \in A} \sum_{t \in S} P_a(s, t) \cdot x_t \quad \text{if } s \in Pr^{>0}(B) \setminus B$$

$$x_s = 1 \quad \text{if } s \in B$$

$$x_s = 0 \quad \text{if } s \notin Pr^{>0}(B)$$

Model Checking Cuantitativo

$\mathcal{M} \models \phi?$

```

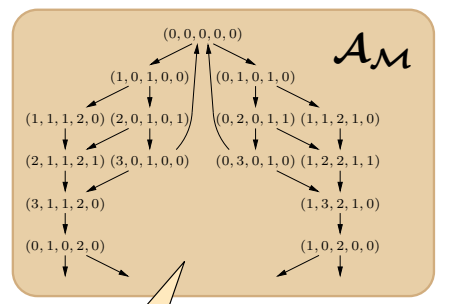
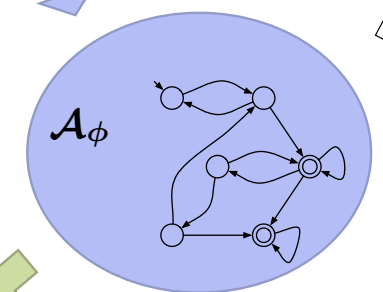
int y1 = 0;
int y2 = 0;
short in_critical = 0;

active proctype process_1() {
do
:: true ->
1:   y1 = y2+1;
2:   ((y2==0) || (y1<=y2));
3:   in_critical++;
od
y1 = 0;
}

active proctype process_2() {
do
:: true ->
0:   y2 = y1+1;
1:   ((y1==0) || (y2<y1));
2:   in_critical++;
3:   in_critical--;
od
y2 = 0;
}
    
```

$\phi : \square \diamond crit_1 \wedge \square \diamond crit_2$

Se reduce a un problema de optimización lineal



$P(\phi) > 0.95$

$$x_s = \max_{a \in A} \sum_{t \in S} P_a(s, t) \cdot x_t \quad \text{if } s \in Pr^{>0}(B) \setminus B$$

$$x_s = 1 \quad \text{if } s \in B$$

$$x_s = 0 \quad \text{if } s \notin Pr^{>0}(B) \cup B$$

Proceso de decisión de Markov

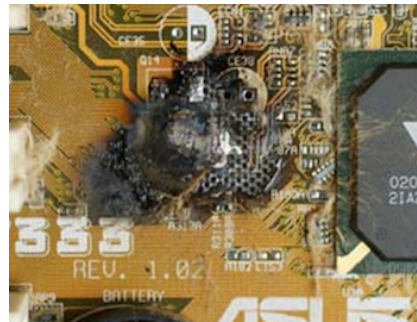
Incluye primitivas de aleatoriedad

Background references and citations:

- MODEST: A Compositional Modeling Formalism for Hard and Softly Timed Systems (Henrik Björnskov)
- Probabilistic Transition System Specification: Congruence and Full Abstraction of Bistimulation* (Fahmi)
- SOS rule formats for conxer and abstract probabilistic bisimulations* (Pedro R. D'Argenio, Daniel G. Dierker)
- Automating Bisimulation Equivalences and Metrics from Probabilistic SOS (Pedro R. D'Argenio, Daniel G. Dierker)
- Information and Computation (Cornelia Pasareanu)
- Reachability Analysis of Probabilistic Systems by Successive Refinements (Pedro R. D'Argenio, Bertand Jonsson, Hosang E. Lee, and Kim G. Larsen)
- Reduction and Refinement Strategies for Probabilistic Analysis (Pedro R. D'Argenio, Bertand Jonsson, Hosang E. Lee, and Kim G. Larsen)
- Partial Order Reduction on Concurrent Probabilistic Programs* (Pedro R. D'Argenio)
- Order Reduction for Probabilistic Schedulers: A Revision for Distributed Schedulers* (Sergio Giro, Pedro R. D'Argenio, and Luis María Ferrer Fioriti)
- Significant Diagnostic Counterexamples in Probabilistic Model Checking (Miguel E. Andrés, Pedro D'Argenio, and Peter van Rossum)
- Theoretical Computer Science (Elsevier)
- Significant Diagnostic Counterexamples in Probabilistic Model Checking (Miguel E. Andrés, Pedro D'Argenio, and Peter van Rossum)

Fallas

Fallas



Un sistema es **resiliente** si ...

... tiene la habilidad de **proveer y mantener un nivel de servicio aceptable** aún **bajo la presencia de fallas** y otros inconvenientes que puedan surgir y presentar un desafío al funcionamiento normal del sistema.

¿Cómo enfrentar las fallas?

¿Cómo enfrentar las fallas?

Failover

Redundancia
Redundancia
Redundancia
Redundancia
Redundancia
Redundancia

Votación

Timeouts y
reintentos

Detección y
corrección de errores

Eventos

Los eventos pueden cuantificarse probabilísticamente

Ejemplos:

- ❖ Probabilidad de pérdida de un mensaje
- ❖ Tiempo esperado de vida de una fuente de alimentación
- ❖ Tiempo esperado de reparación del disco rígido
- ❖ Tiempo esperado de transmisión tierra-satélite
- ❖ Probabilidad de alteración de un bit bajo radiación
- ❖ Tiempo esperado de refrescado de memoria

(Algunas) Técnicas de análisis

❖ Model checking cuantitativo

(Algunas) Técnicas de análisis

❖ Model checking cuantitativo

Ya lo vimos

(Algunas) Técnicas de análisis

❖ Model checking cuantitativo

Ya lo vimos

❖ Simulación por eventos discretos

(Algunas) Técnicas de análisis

❖ Model checking cuantitativo

Ya lo vimos

❖ Simulación por eventos discretos

En particular nos interesa la simulación de eventos raros

*es decir,
de muy baja probabilidad*

(Algunas) Técnicas de análisis

❖ Model checking cuantitativo

Ya lo vimos

❖ Simulación por eventos discretos

En particular nos interesa la simulación de eventos raros

*es decir,
de muy baja probabilidad*

❖ Model checking estadístico

(Algunas) Técnicas de análisis

- ❖ Model checking cuantitativo

Ya lo vimos

- ❖ Simulación por eventos discretos

En particular nos interesa la simulación de eventos raros

*es decir,
de muy baja probabilidad*

- ❖ Model checking estadístico

Es una variante específica de la simulación

(Algunas) Técnicas de análisis

- ❖ Model checking cuantitativo

Ya lo vimos

- ❖ Simulación por eventos discretos

En particular nos interesa la simulación de eventos raros

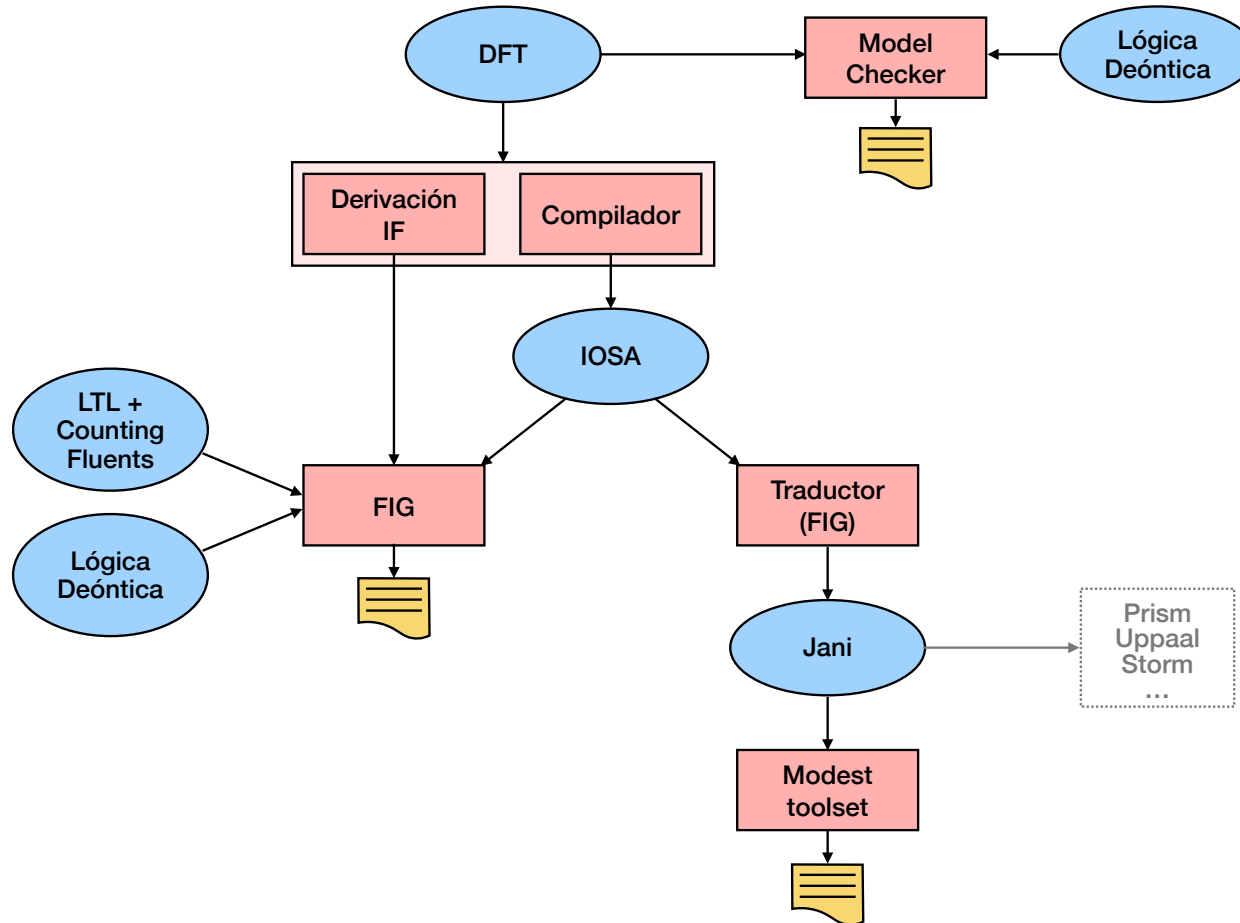
*es decir,
de muy baja probabilidad*

- ❖ Model checking estadístico

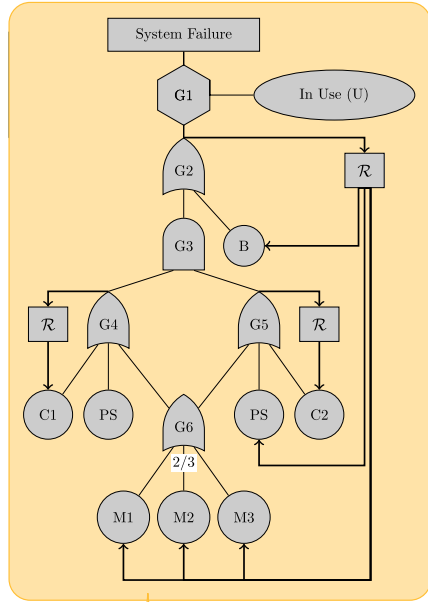
Es una variante específica de la simulación

+ no-determinismo

Proyectos RAFTSys y ARES



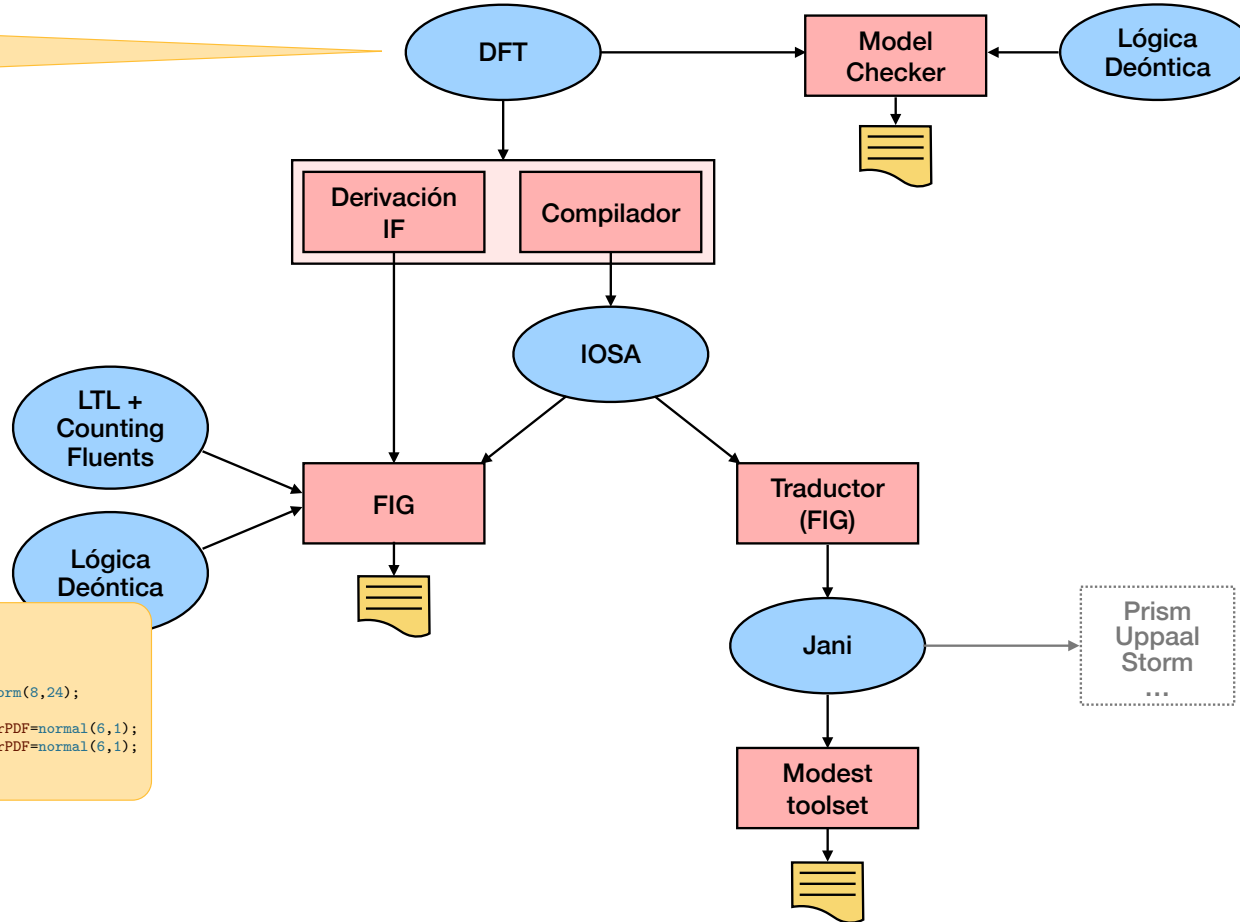
Proyectos RAFTSys y ARES



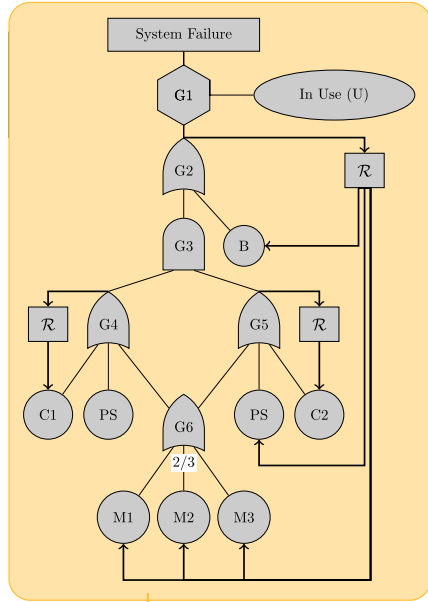
```

1 toplevel "G1";
2 "G1" and "G2" "B";
3 "G2" wsp "A" "S1" "S2";
4 "B" EXT_failPDF=rayleigh(6.0e-2) EXT_repairPDF=uniform(8,24);
5 "A" lambda=1.11e-3 EXT_repairPDF=normal(6,1);
6 "S1" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
7 "S2" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
8 "RB" repairbox_priority "B" "S2" "S1" "A";

```

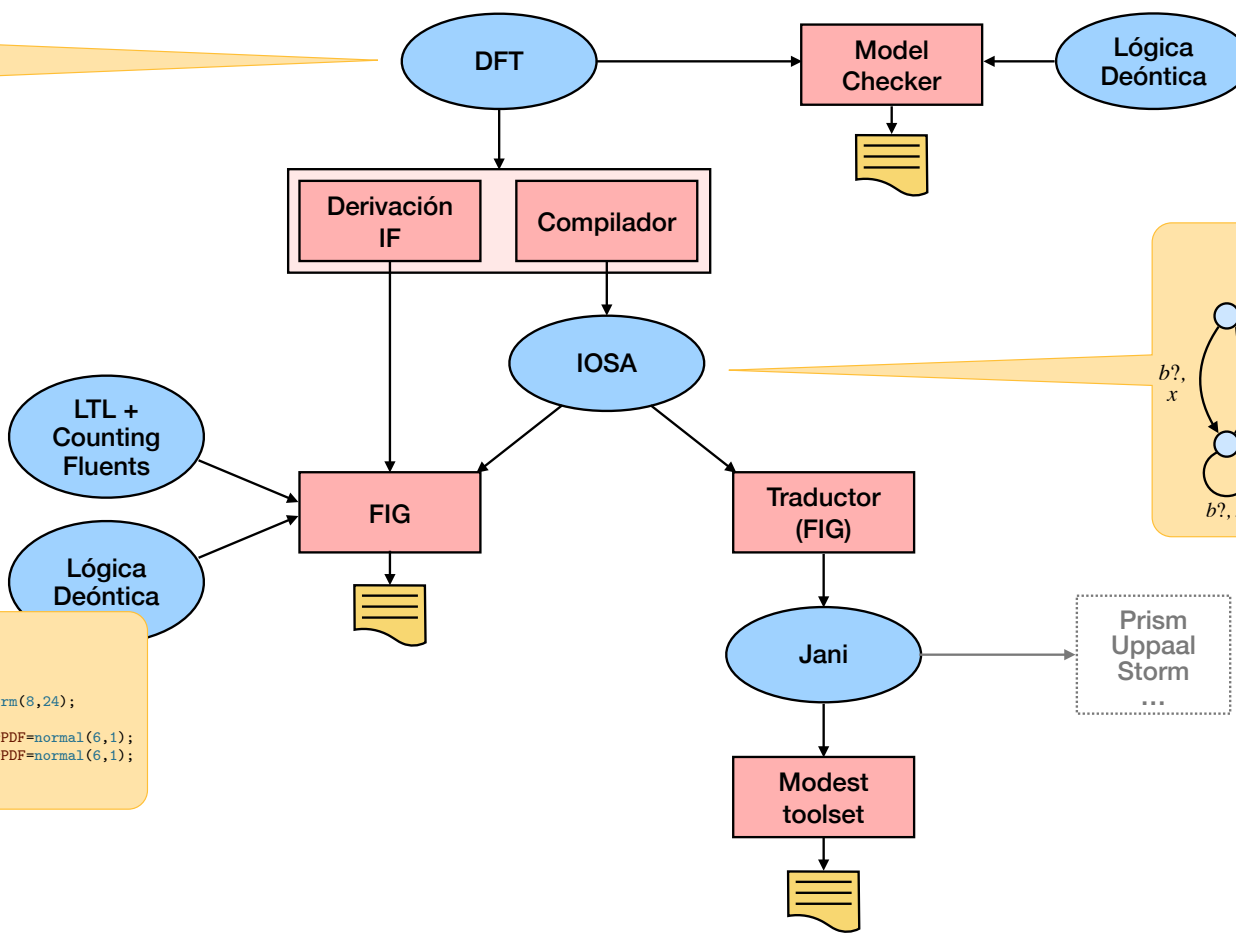


Proyectos RAFTSys y ARES



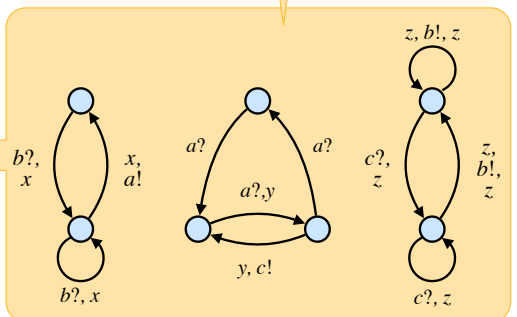
```

1 toplevel "G1";
2 "G1" and "G2" "B";
3 "G2" wsp "A" "S1" "S2";
4 "B" EXT_failPDF=rayleigh(6.0e-2) EXT_repairPDF=uniform(8,24);
5 "A" lambda=1.11e-3 EXT_repairPDF=normal(6,1);
6 "S1" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
7 "S2" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
8 "RB" repairbox_priority "B" "S2" "S1" "A";
  
```

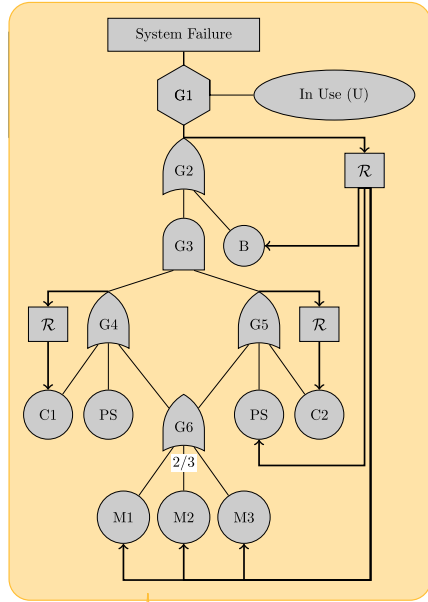


```

1 module BE
2 fc, rc : clock;
3 inform : [0..2] init 0;
4 broken : [0..2] init 0;
5
6 [ff!] broken=0 @ fc -> (inform'=1) & (broken'=1);
7 [r??] broken=1 -> (broken'=2) & (rc'=γ);
8 [up!] broken=2 @ rc -> (inform'=2) &
9 (broken'=0) & (fc'=μ);
10
11 [f!!] inform=1 -> (inform'=0);
12 [u!!] inform=2 -> (inform'=0);
13 endmodule
  
```



Proyectos RAFTSys y ARES

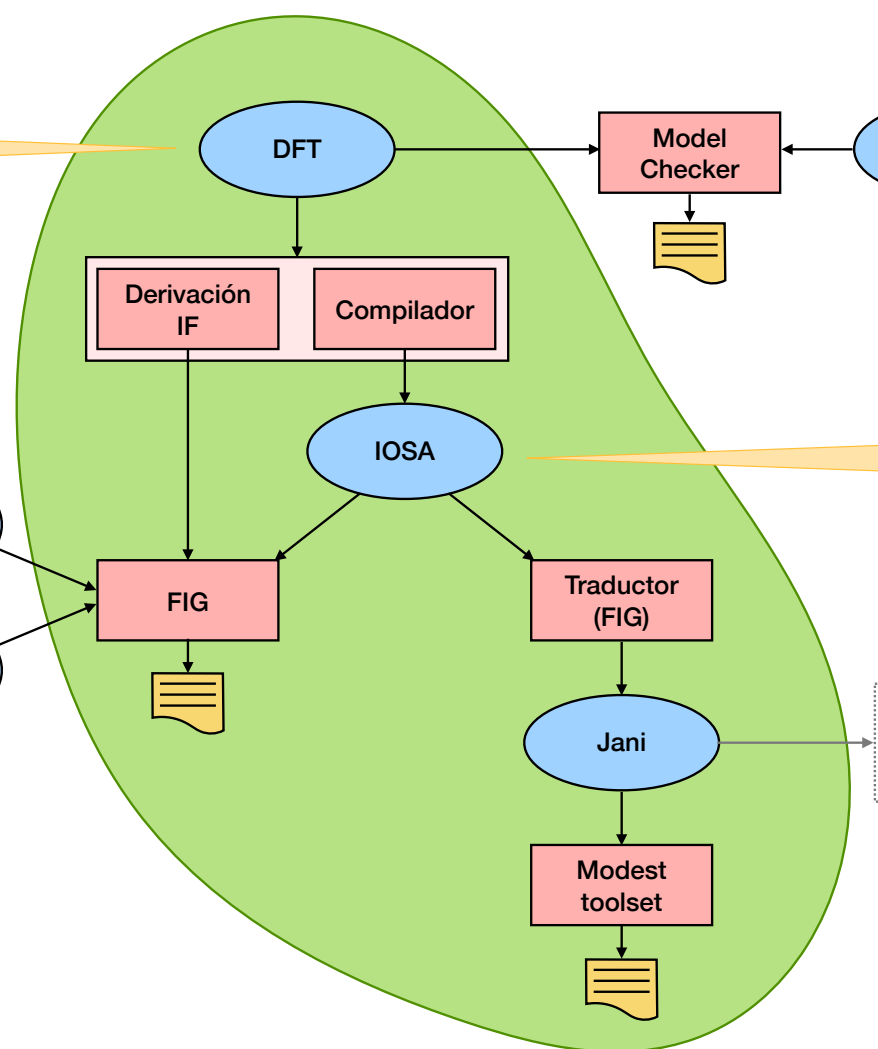


```

1 toplevel "G1";
2 "G1" and "G2" "B";
3 "G2" wsp "A" "S1" "S2";
4 "B" EXT_failPDF=rayleigh(6.0e-2) EXT_repairPDF=uniform(8,24);
5 "A" lambda=1.11e-3 EXT_repairPDF=normal(6,1);
6 "S1" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
7 "S2" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
8 "RB" repairbox_priority "B" "S2" "S1" "A";
  
```

LTL + Counting Fluents

Lógica Deóntica

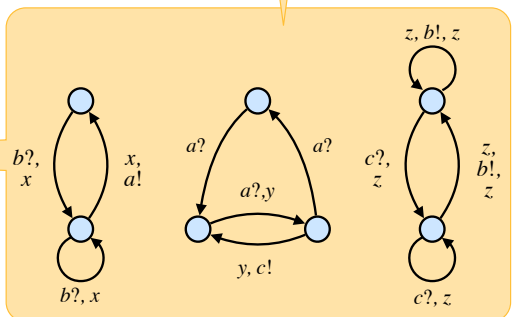


Model Checker

Lógica Deóntica

```

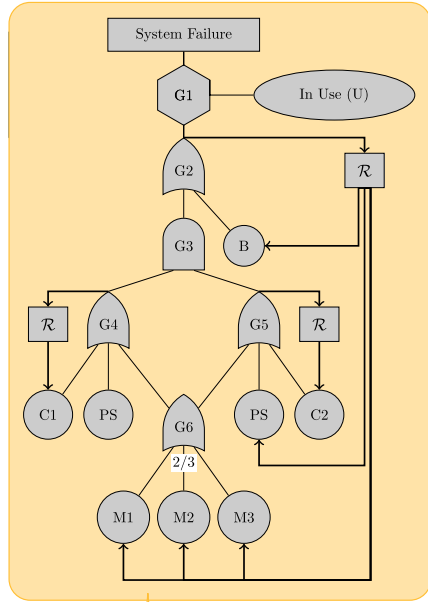
1 module BE
2 fc, rc : clock;
3 inform : [0..2] init 0;
4 broken : [0..2] init 0;
5
6 [fl!] broken=0 @ fc -> (inform'=1) & (broken'=1);
7 [r??] broken=1 -> (broken'=2) & (rc'=γ);
8 [up!] broken=2 @ rc -> (inform'=2) &
9 (broken'=0) & (fc'=μ);
10
11 [f!!] inform=1 -> (inform'=0);
12 [u!!] inform=2 -> (inform'=0);
13 endmodule
  
```



Prism Uppaal Storm ...

Proyectos RAFTS

Medición de tolerancia a fallas + Juegos estocásticos

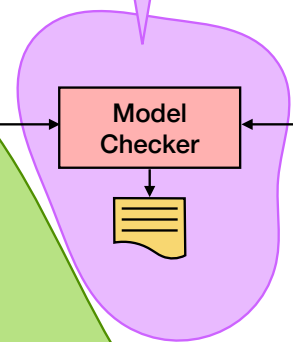
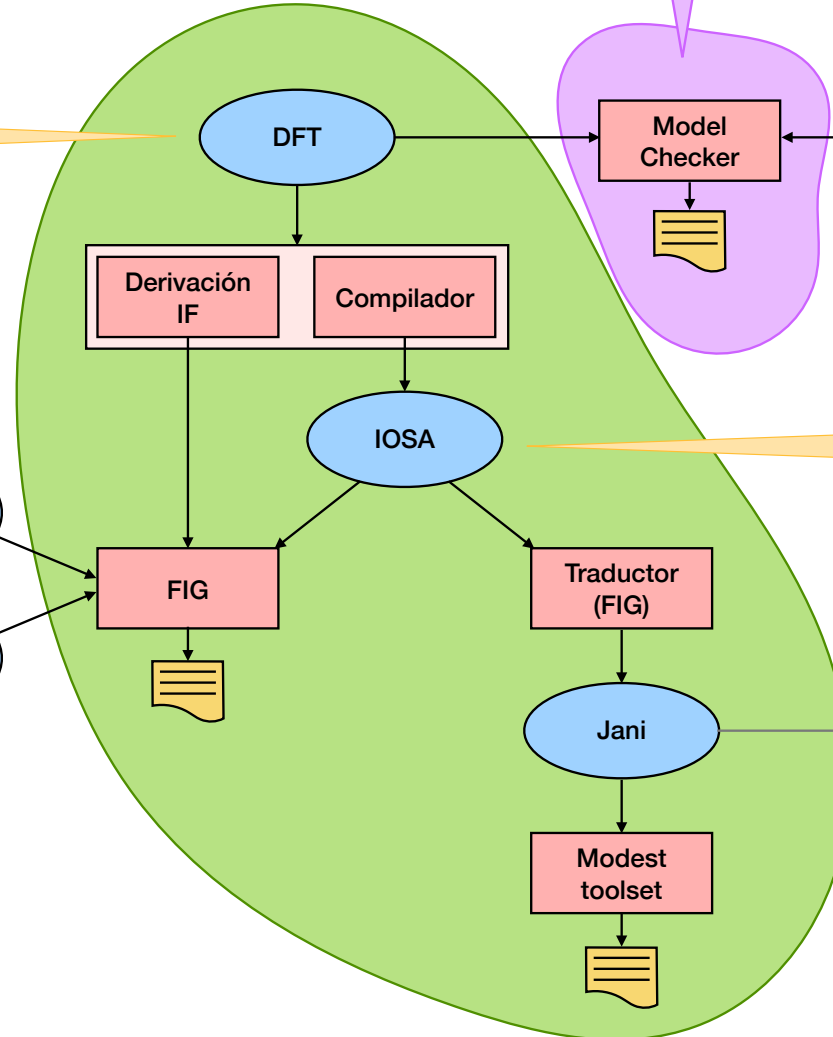


```

1 toplevel "G1";
2 "G1" and "G2" "B";
3 "G2" wsp "A" "S1" "S2";
4 "B" EXT_failPDF=rayleigh(6.0e-2) EXT_repairPDF=uniform(8,24);
5 "A" lambda=1.11e-3 EXT_repairPDF=normal(6,1);
6 "S1" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
7 "S2" lambda=0.2 EXT_dormPDF=erlang(9,0.3) EXT_repairPDF=normal(6,1);
8 "RB" repairbox_priority "B" "S2" "S1" "A";
  
```

LTL + Counting Fluents

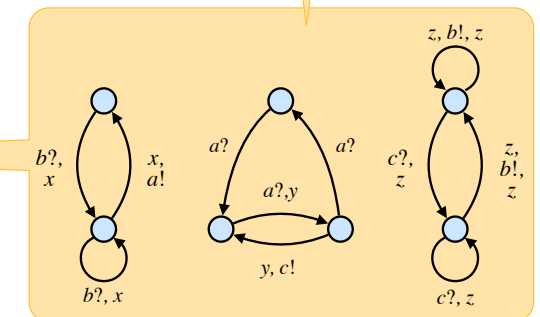
Lógica Deóntica



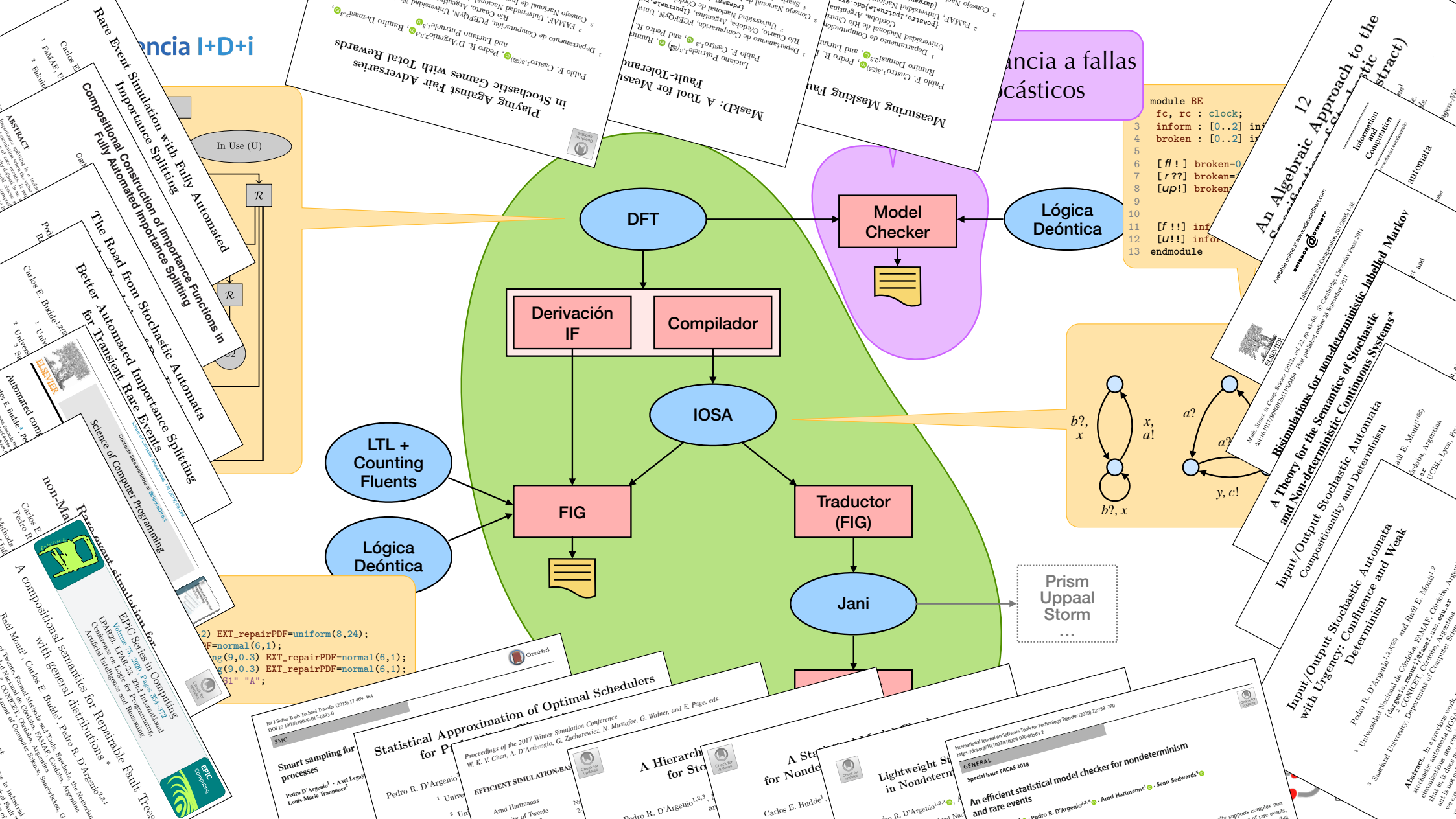
Lógica Deóntica

```

module BE
fc, rc : clock;
3 inform : [0..2] init 0;
4 broken : [0..2] init 0;
5
6 [f!] broken=0 @ fc -> (inform'=1) & (broken'=1);
7 [r??] broken=1 -> (broken'=2) & (rc'=γ);
8 [up!] broken=2 @ rc -> (inform'=2) &
    (broken'=0) & (fc'=μ);
9
10
11 [f!!] inform=1 -> (inform'=0);
12 [U!] inform=2 -> (inform'=0);
13 endmodule
  
```



Prism Uppaal Storm ...



Financia I+D+i

Financia a fallas
ocásticas

```

module BE
  fc, rc : clock;
  3 inform : [0..2] inf;
  4 broken : [0..2] inf;
  5
  6 [f!] broken=0;
  7 [r??] broken=
  8 [u!] broken=
  9
  10 [f!] inf;
  11 [r??] inf;
  12 [u!] inf;
  13 endmodule
  
```

Lógica Deóntica

Model Checker

DFT

Derivación IF

Compilador

IOSA

FIG

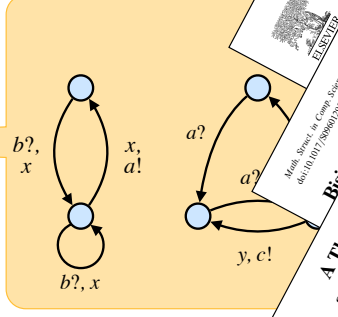
Traductor (FIG)

Jani

Prism Uppaal Storm

LTL + Counting Fluents

Lógica Deóntica



Rare Event Simulation with Fully Automated Importance Splitting

Better Automated Importance Splitting for Transient Rare Events

Automated Control Synthesis for Stochastic Systems

Reasonable Simulation for Repairable Fault Trees

Statistical Approximation of Optimal Schedulers

A Hierarchical Approach for Non-deterministic Systems

Playing Against Fair Adversaries in Stochastic Games with Total Rewards

MASKD: A Tool for Measuring Masking Fault-Tolerance

Measuring Masking Fault-Tolerance

An Algebraic Approach to the Verification of Stochastic Systems

A Theory for the Semantics of Stochastic and Non-deterministic Continuous Markov Automata

Input/Output Stochastic Automata with Urgency: Confidence and Weak Determinism

Input/Output Stochastic Automata with Urgency: Confidence and Weak Determinism

Lightweight Simulation in Nondeterminism

Smart sampling for processes

A Hierarchical Approach for Non-deterministic Systems

Lightweight Simulation in Nondeterminism

An efficient statistical model checker for nondeterminism and rare events

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract

Abstract



Redes de Satélites



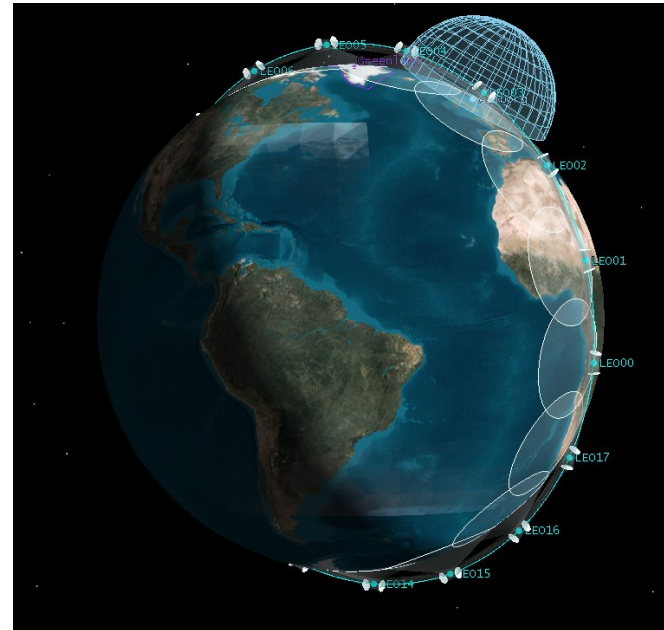
GomSpace (DK)



Redes de Satélites

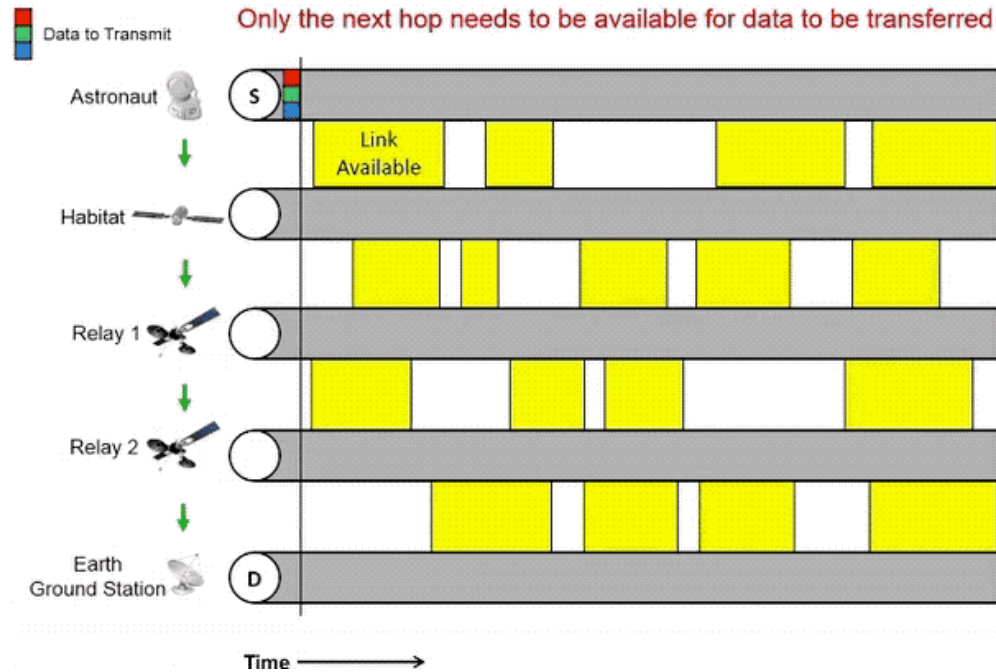


GomSpace (DK)



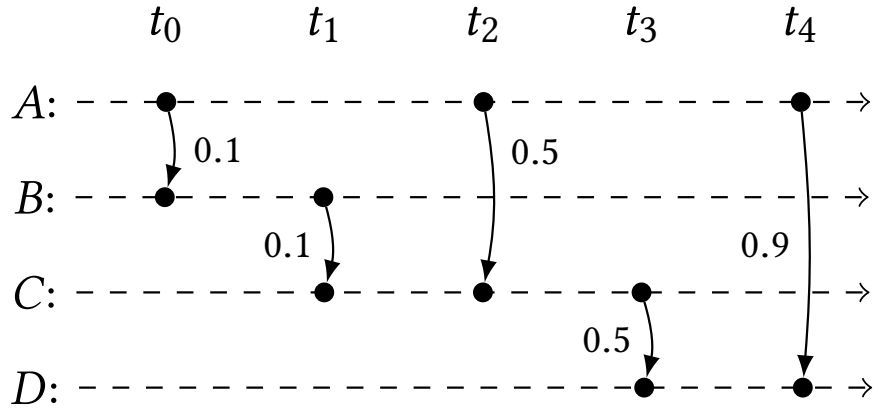
Redes Tolerantes a Demora

Sample Scenario Using DTN-Capable Nodes



Redes Tolerantes a Demora

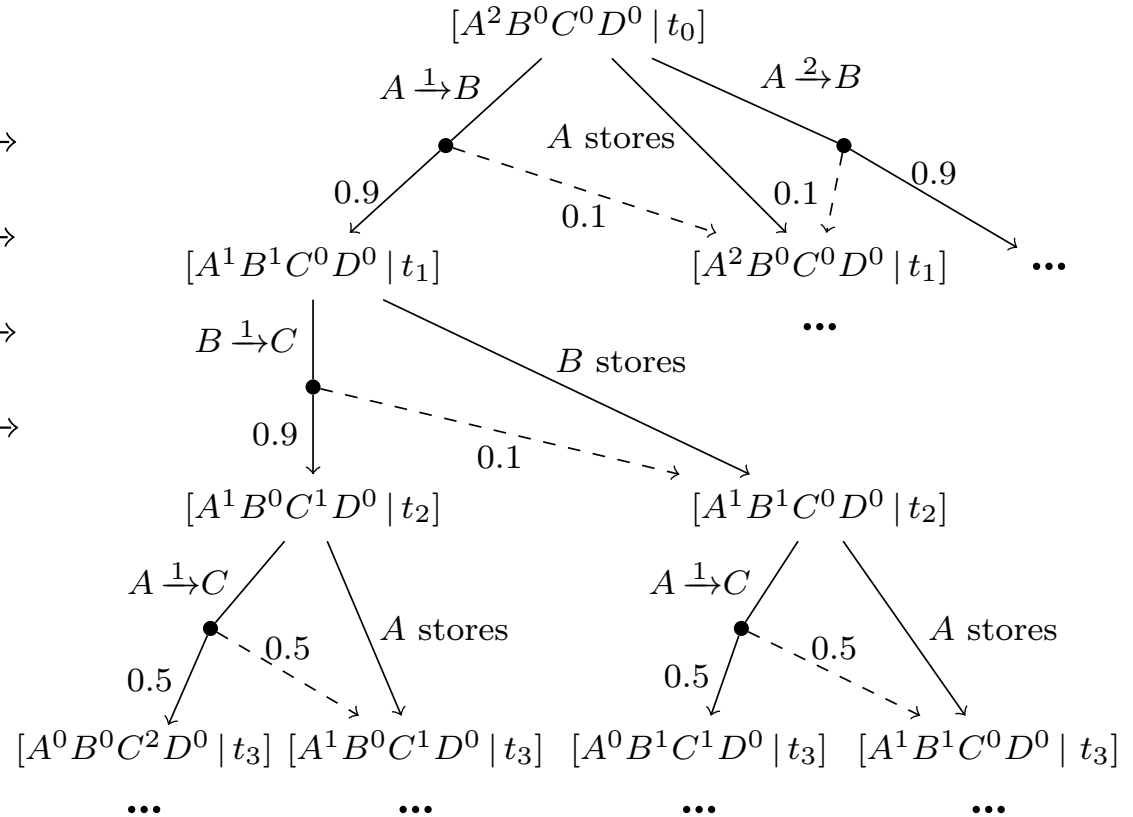
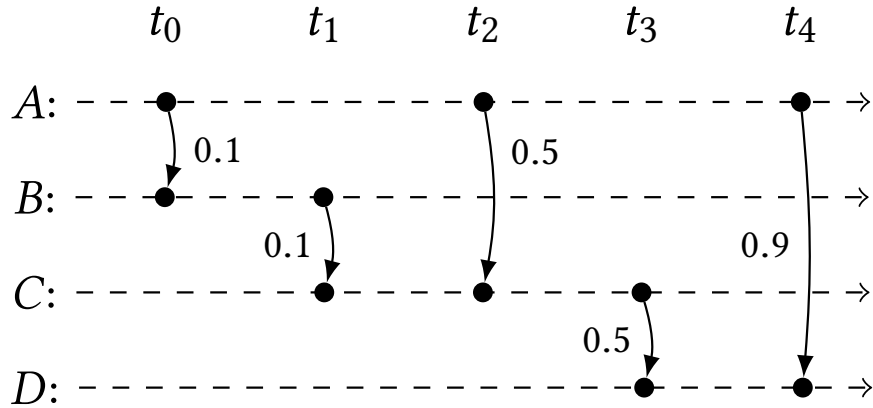
Ruteo basado en Model Checking Cuantitativo



Plan de contacto

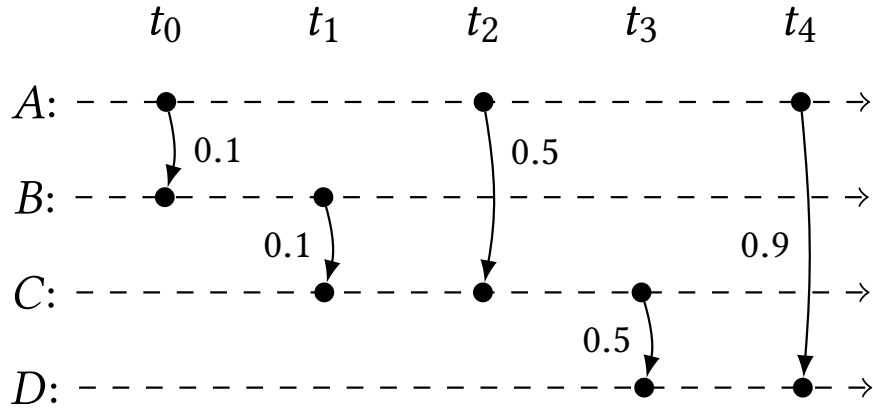
Redes Tolerantes a Demora

Ruteo basado en Model Checking Cuantitativo



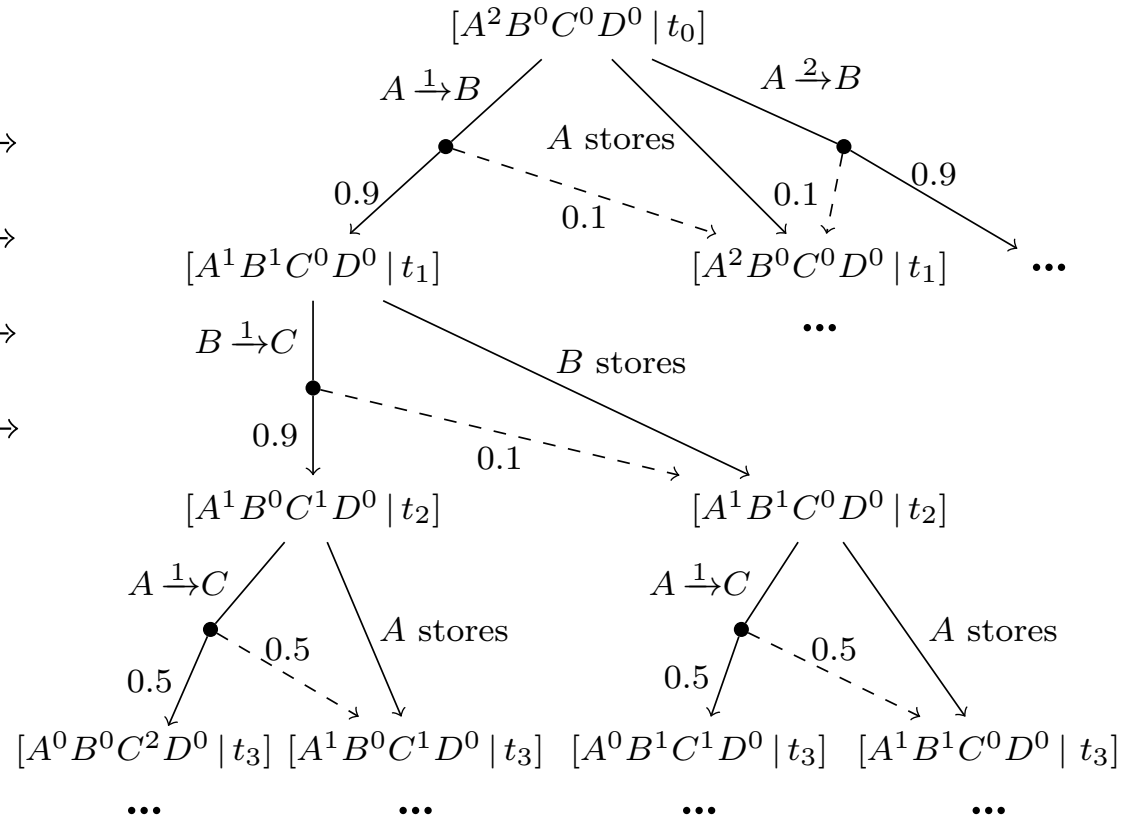
Redes Tolerantes a Demora

Ruteo basado en Model Checking Cuantitativo



Plan de contacto

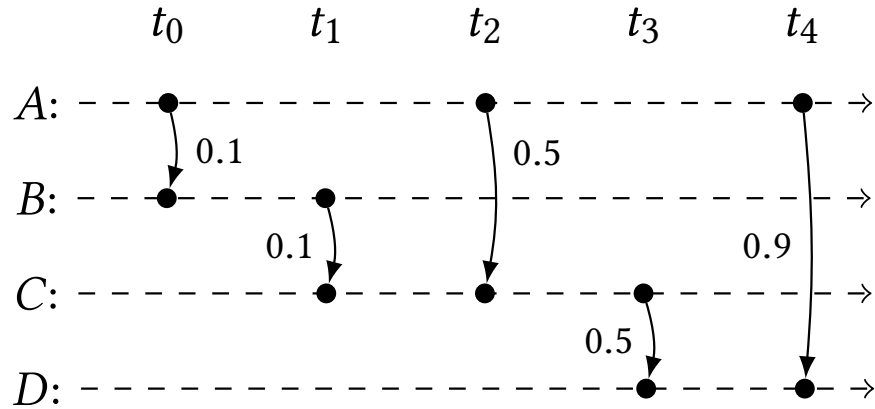
- ❖ RUCOP
- ❖ L-RUCOP
- ❖ CGR-UCOP



Proceso de decisión de Markov

Redes Tolerantes a Demora

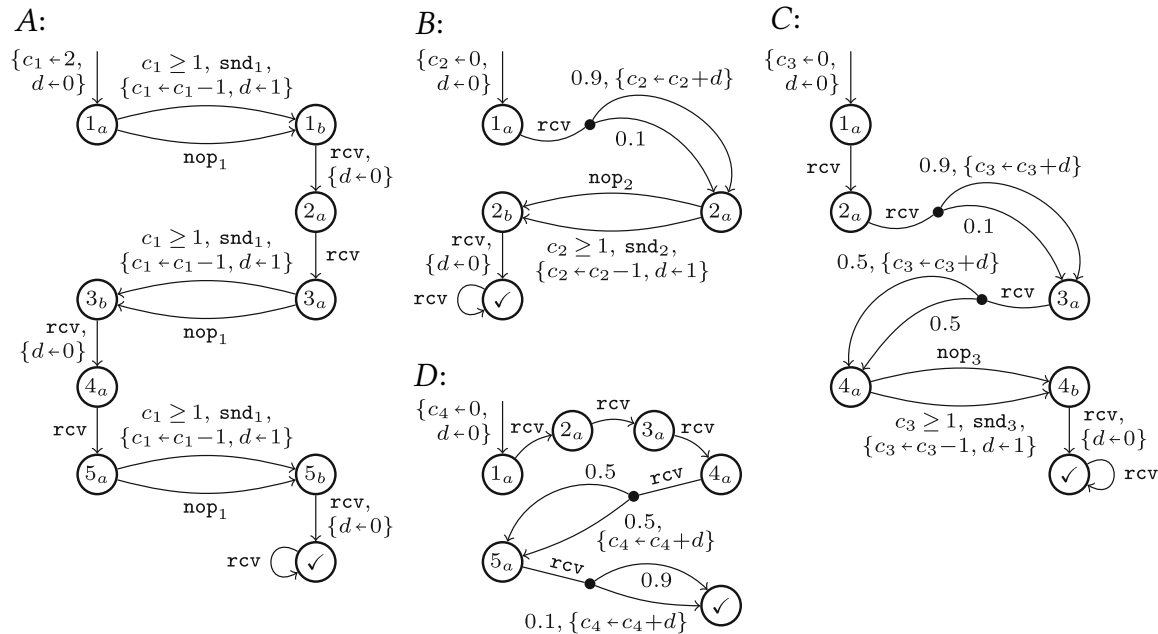
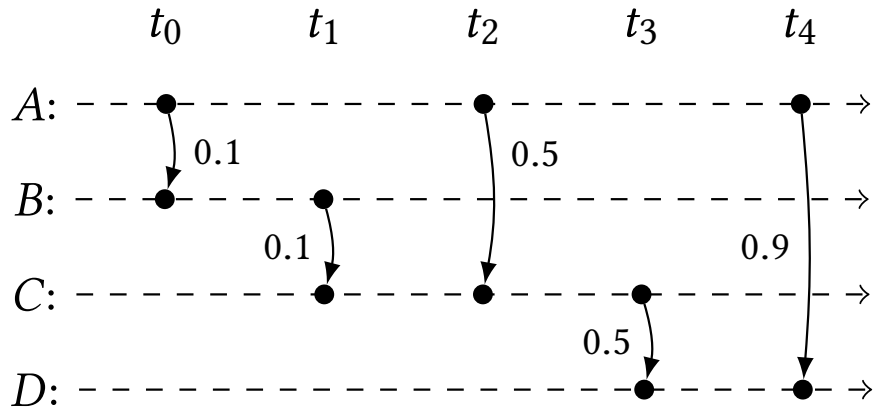
Ruteo basado en Model Checking Estadístico con LSS



Plan de contacto

Redes Tolerantes a Demora

Ruteo basado en Model Checking Estadístico con LSS

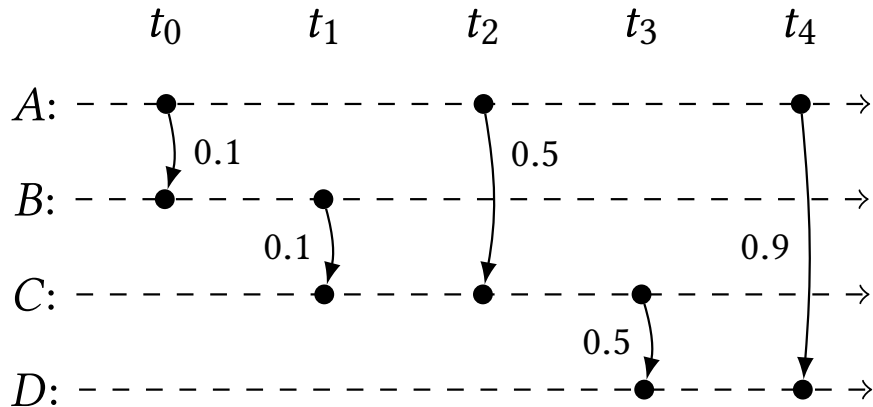


Red de procesos de decisión de Markov

Simulación por Eventos Discretos + LSS

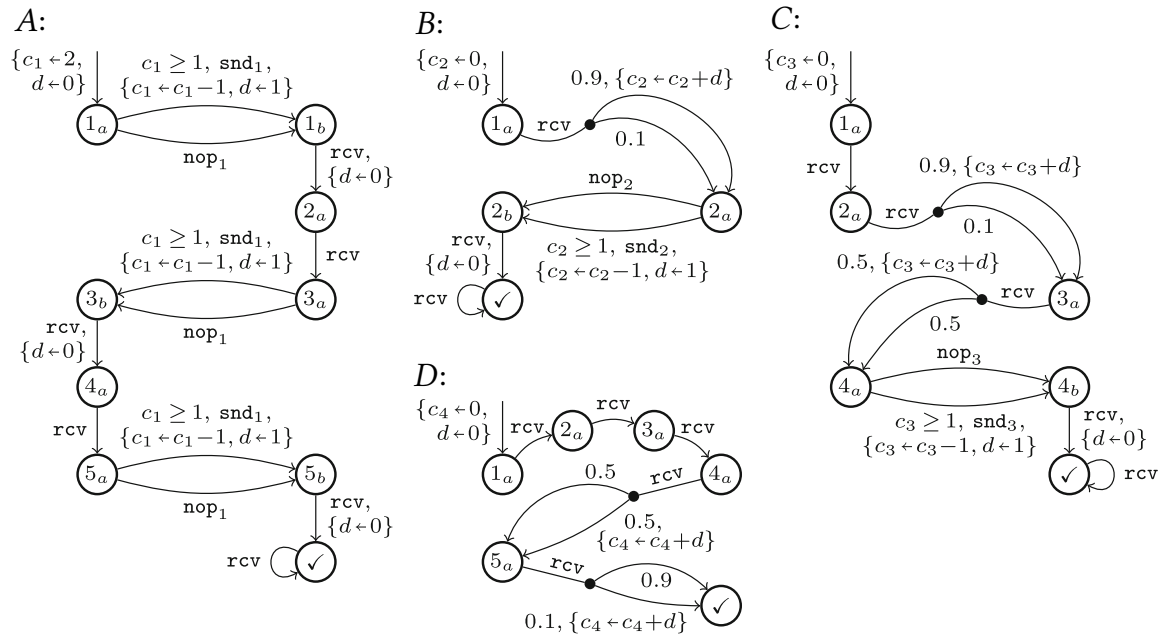
Redes Tolerantes a Demora

Ruteo basado en Model Checking Estadístico con LSS



Plan de contacto

- ❖ LSS
- ❖ L-LSS



Red de procesos de decisión de Markov
 Simulación por Eventos Discretos + LSS

Redes Tolerantes a Demora

(experimentos)

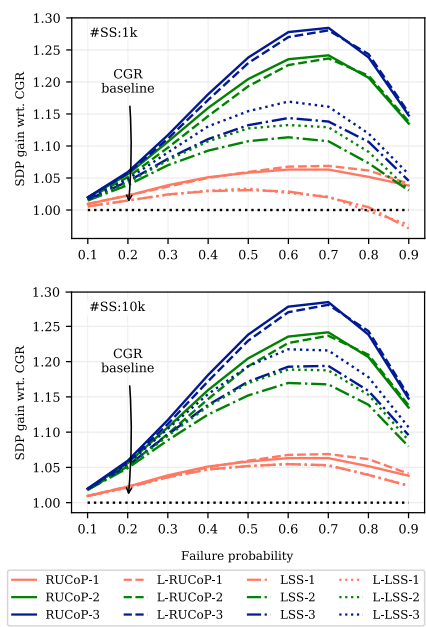


Figure 5: SDP gain over CGR in random networks.

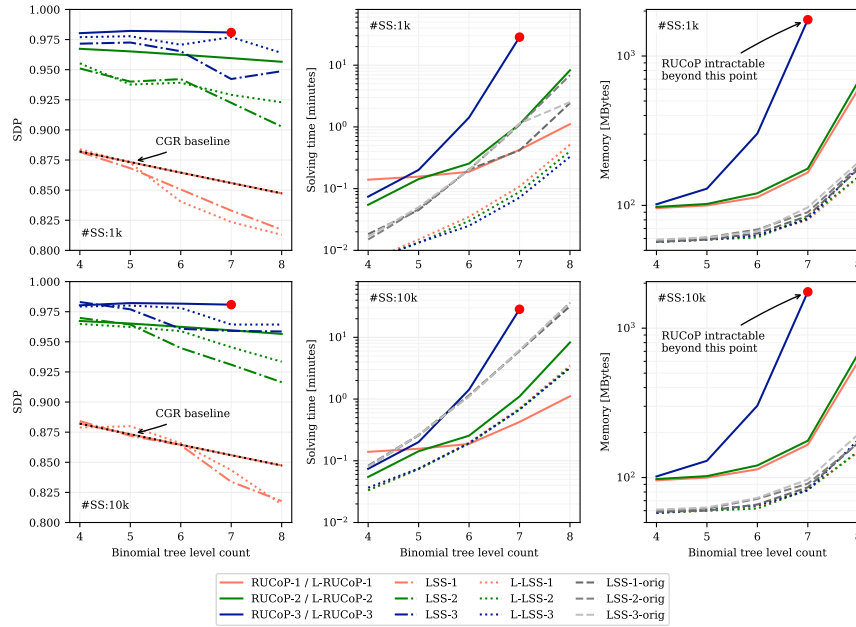


Figure 6: SDP, solving time, and memory for binomial networks with varying complexity (i.e., levels).

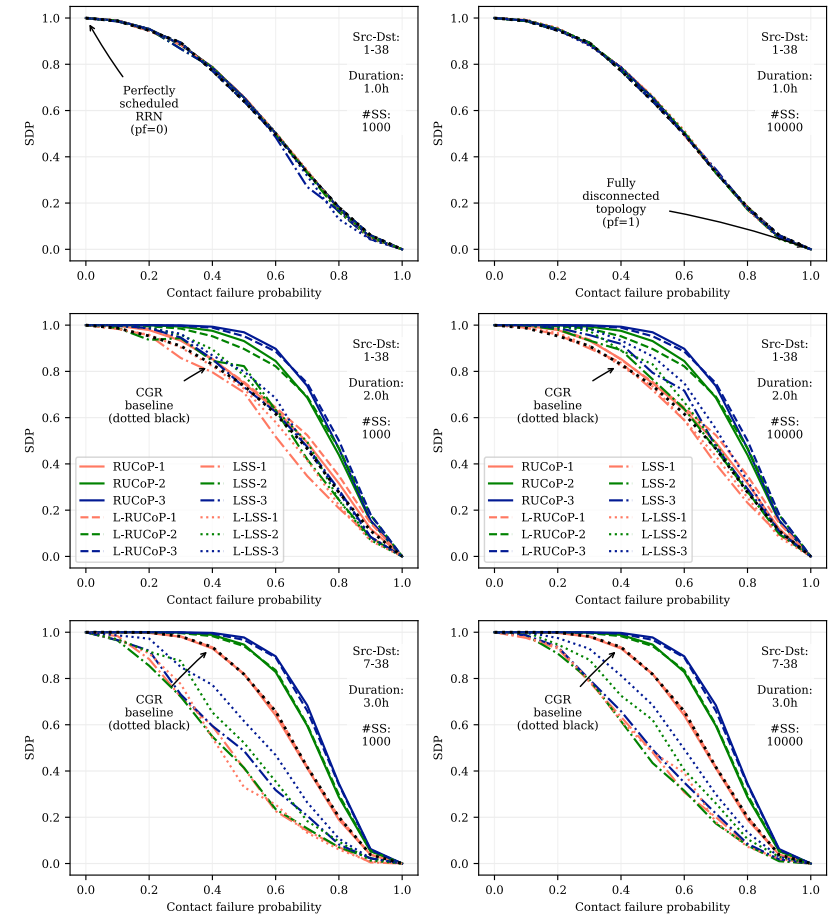


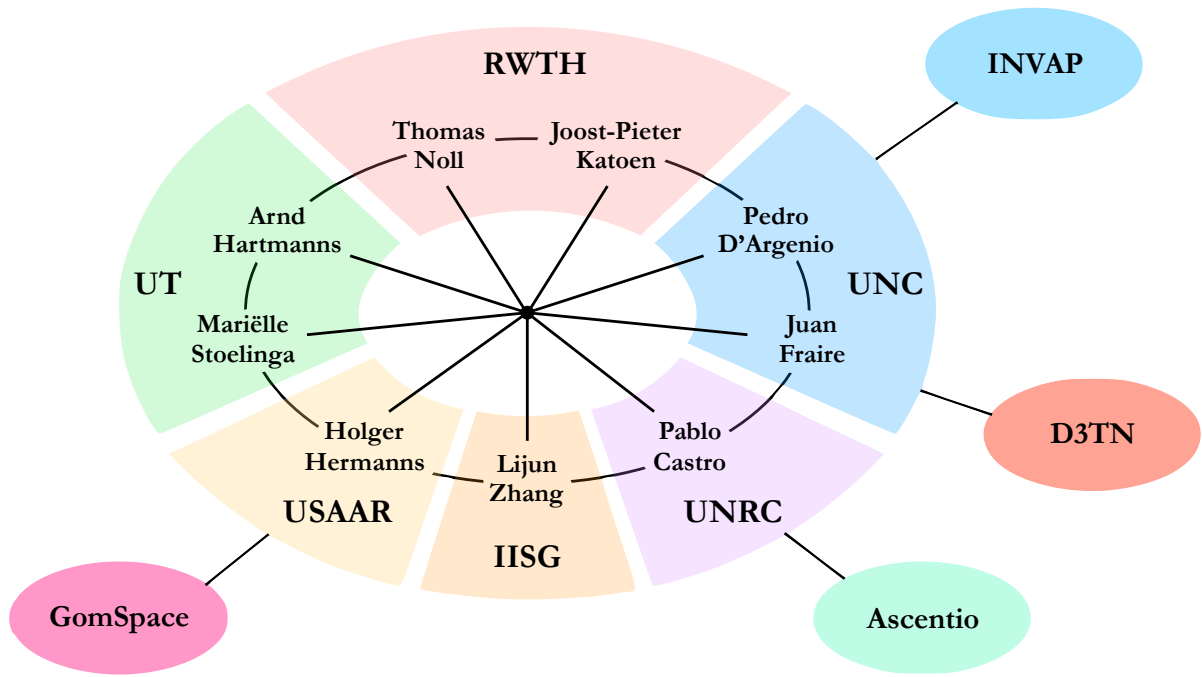
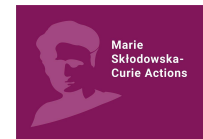
Figure 7: SDP for RRN for different source-target nodes, contact plan duration, and scheduler sampling.



MISSION



Horizon 2020
European Union funding
for Research & Innovation





Traditional space paradigm



Resilience & Reliability

New space paradigm



Cost



Traditional space paradigm



Resilience & Reliability

New space paradigm



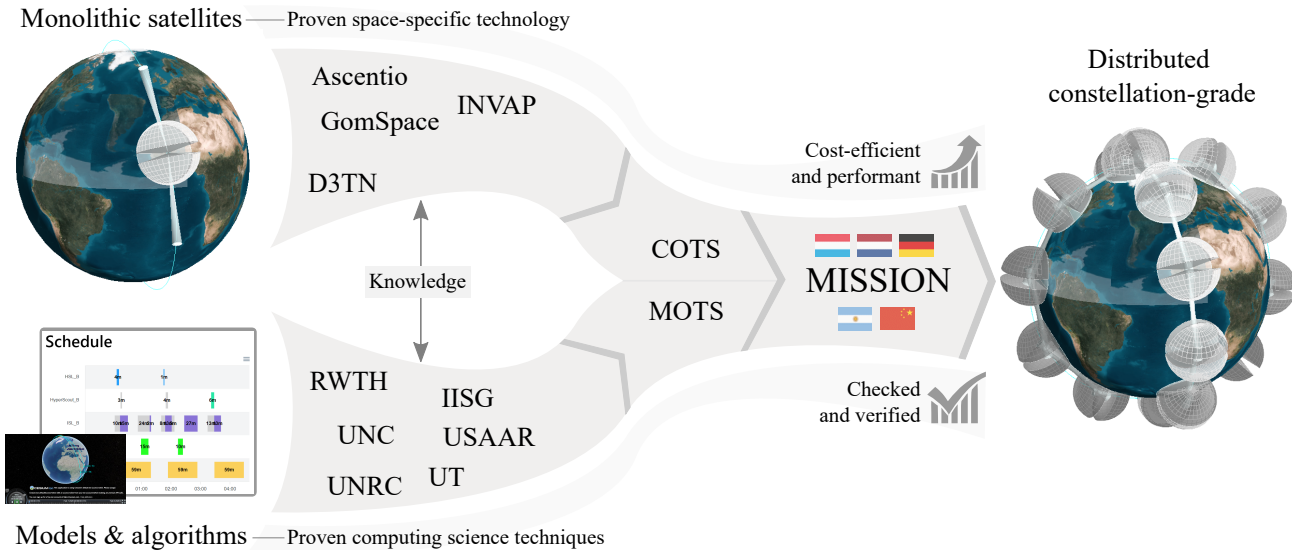
Cost



Resilience & Reliability

Links & Network

Energy & Resources



Chantadas

Chantadas abusivas



Estrategias de
“lock in”

Chantadas famosas



Chantadas famosas



International Business Times

Economy | Companies | Markets | Finance | Regulation

Business | Companies

VW scandal: Carmaker was warned by Bosch about test-rigging software in 2007

By Karthick Arvinth
Updated September 28, 2015 05:53 BST

Carthodas famosas

🏠 > Lifestyle > Cars > News
Diesel emissions scandal: Fiat under investigation



Economy | Companies | Markets | Finance | Regulation
International Business Times
Business | Companies

VW scandal: Carmaker was warned by Bosch about test-rigging software in 2007



By Karthick Arvinth
Updated September 28, 2015 05:53 BST

Car brands famosas

Home > Lifestyle > Cars > News
Diesel emissions scandal: Fiat under investigation



Germany orders Porsche recall over diesel emissions cheating

Economic May 18, 2018

Business | Companies
Markets | Finance | Regulation
National Business Times

VW scandal: Carmaker was warned by Bosch about test-rigging software in 2007



By Karthick Arvinth

Updated September 28, 2015 05:53 BST

Carthodas famosas

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

Nissan found guilty of using diesel emissions cheat device in South Korea
Nissan has denied any wrongdoing, but the South Korean government has ruled the Renault-

Germany orders Porsche recall over diesel emissions cheating

Economic Times | Markets | Finance | Regulation
Business | Companies

VW scandal: Carmaker was warned by Bosch about test-rigging software in 2007
By Karthick Arvinth
Updated September 28, 2015 05:53 BST

Car brands famosas

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

Nissan found guilty of using diesel emissions cheat device in South Korea
Nissan has denied any wrongdoing, but the South Korean government has ruled the Renault-

Germany orders Porsche recall over diesel emissions cheating

Business | Companies
VW scandal: Car about test devices
Daimler forced to recall Mercedes with defeat
By Karthik
Updated Se

© 11 June 2018
Diesel emissions scandal

Challenges for

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

Nissan found guilty of using diesel emissions cheat device in South Korea
Nissan has denied any wrongdoing, but the South Korean government has ruled the Renault-

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal
Share 1 Tweet G+ Share in Share

Germany orders Porsche recall over diesel emissions cheating

Economic Times | Markets | Finance | Regulation
Business | Companies
VW scandal: Car about test
Daimler forced to recall Mercedes with defeat devices
By Karthik Updated Sep 11, 2018
© 11 June 2018
Diesel emissions scandal

Challenges for

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

Nissan found guilty of using diesel emissions cheat device in South Korea
Nissan has denied any wrongdoing, but the South Korean government has ruled the Renault-

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal
Share 1 Tweet G+ Share in Share

Germany orders Porsche recall over diesel emissions cheating

Economic Times
May 18, 2018
Business | Companies
Markets | Finance | Regulation
VW scandal: Car companies recall Mercedes with defeat devices
Daimler forced to recall Mercedes with defeat devices
By Karthik
Updated Sep 11, 2018
© 11 June 2018
Diesel emissions scandal

Renault 'cheated on 25 years of pollution tests'

Challenges for

Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

Nissan found guilty of using diesel emissions cheat device in South Korea
Nissan has denied any wrongdoing, but the South Korean government has ruled the Renault best

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal

Germany orders Porsche recall over diesel emissions

Citroën may have breached emissions rules: report
A model tested by the European Commission recorded pollution levels more than seven times higher than labeled

Business | Companies
VW scandal: Car about test
Daimler forced to recall Mercedes with cereal

Renault 'cheated on 25 years of pollution tests'

Challenges for

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

Nissan found to cheat on emissions

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal

South Korean government has ruled the Renault best

LEADERSHIP > COMPANIES & EXECUTIVES

BMW Sued by U.S. Diesel Drivers Over Emissions-Test Cheating

March 27, 2018

Owners Porsche recall over diesel emissions

Citroën may have breached emissions rules: report

A model tested by the European Commission recorded pollution levels more than seven times higher than labeled

business | Companies
VW scandal: Car makers



By Karthik
Updated Sep

Daimler forced to recall Mercedes with diesel emission devices

© 11 June 2018

Diesel emissions scandal

Renault 'cheated on 25 years of pollution tests'

Challenges for

Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal

Nissan found cheat device

GM Accused of Cheating on Diesel Emissions

LEADERSHIP > COMPANIES & EXECUTIVES

BMW Sued by U.S. Diesel Drivers Over Emissions-Test Cheating

March 27, 2018

South Korean government has ruled the Renault best
Drivers Porsche recall over diesel emissions

business | Companies

Citroën may have breached emissions rules: report

A model tested by the European Commission recorded pollution levels more than seven times higher than labeled

VW scandal: Car about test



By Karthik
Updated Sep

Daimler forced to recall Mercedes with defeat devices

© 11 June 2018

Diesel emissions scandal

Renault 'cheated on 25 years of pollution tests'

Challenges for

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal
Share 1 Tweet G+ Share

Nissan found to cheat on emissions
Drivers Over Emissions-

GM Accused of Cheating on Diesel Emissions

Toyota truck unit shares plunge after it admits cheating on emissions data
Hino says it falsified diesel engine performance and fuel economy figures for some vehicles manufactured in Japan

LEADERSHIP > COMPANIES & EXECUTIVES
BMW Sued by U.S. for Test Cheating
March 27, 2018

Business | Companies
VW scandal: Car makers about test cheating

Citroën may have breached emissions rules: report
A model tested by the European Commission recorded pollution levels more than seven times higher than labeled
Recall Mercedes with cereal

By Karthik
Updated Sep 11, 2018
Daimler forced to recall devices
© 11 June 2018
Diesel emissions scandal

Renault 'cheated on 25 years of pollution tests'

Challenges for

Home > Lifestyle > Cars > News
Diesel emission scandal: Fiat under investigation

France
PSA Peugeot Citroën | French economy | auto industry
Peugeot suspected of fraud in diesel scandal
Share 1 Tweet G+ Share

Nissan found to cheat on emissions
Nissan drivers Over Emissions-

GM Accused of Cheating on Diesel Emissions

Toyota truck unit shares plunge after it admits cheating on emissions data

Hino says it falsified diesel engine performance and fuel economy figures for vehicles manufactured in Japan

LEADERSHIP > COMPANIES & EXECUTIVES
BMW Sued by U.S. for Test Cheating
March 27, 2018

has ruled the Renault best
Ford sued in U.S. court over claims of diesel emissions cheating; company denies assertion

business | Companies
Citroën may have breached emissions rules
A model tested by the European Commission was found to emit up to seven times higher than labeled.
Recall Mercedes

Eric D. Lawrence
Detroit Free Press
Published 5:14 p.m. ET Jan. 10, 2018 | Updated 1:38 p.m. ET Jan. 11, 2018

VW scandal: Car makers admit about test cheating
Daimler forced to recall Mercedes
By Karthik
Updated Sep 11, 2018

Renault 'cheated on 25 years of pollution tests'

© 11 June 2018
Diesel emissions scandal

Chantadas imperdonables



<https://www.mintpressnews.com/214505-2/214505/>

Software doping

❖ Es un problema **ético** y hasta **legal**.

❖ Un software está “**dopado**” si...

... el fabricante incluyó una **funcionalidad oculta** de manera tal que el comportamiento resultante **favorezca intencionalmente a una parte previamente designada**, en contra de los intereses de la sociedad o el licenciatarario del software

Software doping

❖ Es un problema **ético** y hasta **legal**.

Pero ...
propusimos una **solución**
técnica (formal)

❖ Un software está “**dopado**” si...

... el fabricante incluyó una **funcionalidad oculta** de manera tal que el comportamiento resultante **favorezca intencionalmente a una parte previamente designada**, en contra de los intereses de la sociedad o el licenciatarario del software

Software doping

- ❖ Es un problema **ético** y hasta **legal**.

Pero ...
propusimos una **solución**
técnica (formal)

- ❖ Un software está “**dopado**” si...

... el fabricante incluyó una **funcionalidad oculta** de manera tal que el comportamiento resultante **favorezca intencionalmente a una parte previamente designada**, en contra de los intereses de la sociedad o el licenciatarario del software

No es posible
formalizar

Software doping

definición formal

❖ Memoria: $\mu : \text{Variables} \rightarrow \text{Valores}$

❖ Un programa es un transformador de memoria: $(S, \mu) \Downarrow \mu'$

❖ Variables:

Entrada de interés: $i \in \text{Variables}$ Salida de interés: $o \in \text{Variables}$

❖ Software doping

S *no está dopado* si para todas μ_1, μ_2, μ'_1 y μ'_2 ,

$$\left. \begin{array}{l} \mu_1(i) \approx \mu_2(i) \\ (S, \mu_1) \Downarrow \mu'_1 \\ (S, \mu_2) \Downarrow \mu'_2 \end{array} \right\} \Rightarrow \mu'_1(o) \approx \mu'_2(o)$$

“se parece”

Software doping

definición formal

❖ Memoria: $\mu : \text{Variables} \rightarrow \text{Valores}$

❖ Un programa es un transformador de memoria: $(S, \mu) \Downarrow \mu'$

❖ Variables:

Entrada de interés: $i \in \text{Variables}$ Salida de interés: $o \in \text{Variables}$

❖ Software doping

no está dopado si para todas μ_1, μ_2, μ'_1 y μ'_2 ,

“se parece”

Facets of Software Doping

Gilles Barthel¹

² FaMAF, Universidad Nacional de Córdoba, Argentina

³ Saarland University, Germany

Abstract. This paper discusses several aspects of software doping.

Introduction

Is Your Software on Dope?
Formal Analysis of Surreptitiously “enhanced” Programs

Pedro R. D’Argenio^{1,2}, Gilles Barthel¹, and Bernd Finkbeiner²

¹ FaMAF, Universidad Nacional de Córdoba, Argentina
² Computer Science, Saarland University, Germany
³ IMDEA Software Institute, Spain

Abstract—We are confronted with a growing number of devices where device manufacturers equip their products with embedded software that includes functionalities that are not in the original software. Examples include customer lock-in strategies in printers and as a promotional case the diesel emissions scandal in the automotive industry. This software doping phenomenon is spreading more widely and presents a formal characterization challenge. In this paper, we present a formal characterization of software doping in the context of reactive programs, based on the idea of a doping test between the code and its environment.

2018 3rd Workshop on Monitoring and Testing of Cyber-Physical Systems
Cyber-Physical Doping Tests

Sebastian Biewer, Holger Hermanns
Saarland University – Computer Science
Saarland Informatics Campus
Saarbrücken, Germany



EPiC Series in Computing
Volume 57, 2018, Pages 1–17
LPAR-22, 22nd International Conference on Logic for Programming, Artificial Intelligence and Reasoning

Verification, Testing, and
of Automotive

Holger Hermanns
Pedro R. D’Argenio^{1,2,3}

¹ Saarland University, Saarland Informatics Campus, Germany
² FaMAF, Universidad Nacional de Córdoba, Argentina
³ CONICET, Universidad Nacional de Córdoba, Argentina

Doping Tests for Cyber-Physical Systems

Sebastian Biewer¹, Pedro D’Argenio^{1,2,3}

¹ Saarland University, Saarland Informatics Campus, Germany
² FaMAF, Universidad Nacional de Córdoba, Argentina
³ CONICET, Universidad Nacional de Córdoba, Argentina
⁴ Institute of Intelligent Systems, University of Stuttgart, Germany

Doping Tests for Cyber-physical Systems

SEBASTIAN BIEWER, Saarland University, Saarland Informatics Campus, Germany
PEDRO R. D’ARGENIO, Universidad Nacional de Córdoba, FaMAF, Argentina, CONICET, Argentina, and Saarland University, Saarland Informatics Campus, Germany
HOLGER HERMANNNS, Saarland University, Saarland Informatics Campus, Germany and Institute of Intelligent Software, China

Abstract—Running in embedded or cyber-physical systems is typically of proprietary nature, so users do not know what the systems they own are (incapable of doing. Most malfunctions of such systems are not classified as bugs, but some are, which means these cannot be classified as bugs. Examples have become public in the law, altogether polluting the software ecosystem. This paper presents a formal characterization of software doping in the context of reactive programs, based on the idea of a doping test between the code and its environment.

¿Qué conclusión sacan ustedes?

¿Qué conclusión sacan ustedes?

Las que saco yo:

- ❖ No está bueno echar moco
 - ➔ hacer lo posible por evitarlos y eliminarlos

- ❖ Las fallas inevitablemente ocurren
 - ➔ tratar efectivamente con ellas y de manera eficiente

- ❖ Las chantadas son una mala costumbre de la disciplina
 - ➔ No sólo contrarrestarlas ética y legalmente sino también técnicamente

¿Qué conclusión sacan ustedes?

Las que saco yo:

- ❖ No está bueno echar moco
 - ➔ hacer lo posible por evitarlos y eliminarlos
- ❖ Las fallas inevitablemente ocurren
 - ➔ tratar efectivamente con ellas y de manera eficiente
- ❖ Las chantadas son una mala costumbre de la disciplina
 - ➔ No sólo contrarrestarlas ética y legalmente sino también técnicamente

Las técnicas formales
(matemáticas) son **cruciales**
para todo esto

Objetivo del Grupo de Sistemas Confiables

Sistema \models Propiedad

Proveer lenguajes y lógicas para modelar sistemas y propiedades, y técnicas y herramientas que implementen la relación de satisfacción, todo dentro de un marco teórico matemático bien definido.

Luchando contra Errores, Fallas y Chantadas para construir Sistemas Confiables

Pedro R. D'Argenio

Grupo de Sistemas Confiables

Universidad Nacional de Córdoba – CONICET (AR)

<http://dsg.famaf.unc.edu.ar/>

<https://www.cs.famaf.unc.edu.ar/~dargenio/>

