

A Refinement Based Notion of Non-Interference for Interface Automata: Compositionality, Decidability and Synthesis (y otras verduras!)

Matias Lee

CONICET - FaMAF, Universidad Nacional de Córdoba
`{lee,dargenio}@famaf.unc.edu.ar`

I Jornadas de Doctorandos de Cs. de La Comp. de FaMAF
Diciembre 2010



Outline

- 1 El Director, El Doctorando y El Objetivo
 - El Malvado Jefe
 - El pobre Doctorando
 - El Objetivo
- 2 Nuestro último trabajo
 - Necesidad de Interfaces seguras
 - Nuestra Noción de Seguridad
- 3 Comentarios Finales
 - Más Resultados
 - Aplicabilidad



Dr. Pedro R. D'Argenio



Research interests: Formal methods for the modeling and analysis of reactive systems in general, including concurrent, distributed, embedded and (hard and soft) real-time systems. Techniques include, process algebra, automata, operational semantics, bisimulation, model checking, stochastic processes, temporal logics, and formal testing.

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



Lic. Matias Lee



WWW.PHDCOMICS.COM



I am a PhD Student.

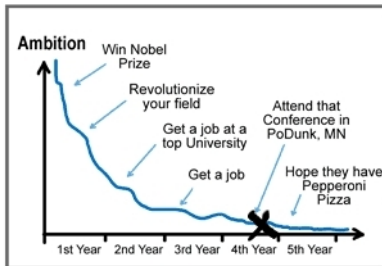
Research interests: Formal methods for the modeling and analysis of reactive systems in general, including concurrent, distributed, embedded and (hard and soft) real-time systems. Techniques include, process algebra, automata, operational semantics, bisimulation, model checking, stochastic processes, temporal logics, and formal testing.

<http://www.cs.famaf.unc.edu.ar/~lee/>



Por donde andamos...

YOUR LIFE AMBITION - What Happened??



X Yo estoy por acá



El Objetivo de MI doctorado:

El Objetivo de MI doctorado:

- Disernir que es relevante y que no lo es para una investigación.
- Aprender a definir objetivos claros y alcanzables para una investigación.
- Aprender a desarrollar un trabajo de investigación en forma correcta.
- Aprender a presentar los resultados de forma clara y concisa, donde realmente se puede apreciar el aporte del trabajo realizado... y si es de forma interesante mejor!
- Hacer todo esto de forma eficiente.



El Objetivo de MI doctorado:

i.e. formarme como investigador.



Outline

- 1 El Director, El Doctorando y El Objetivo
 - El Malvado Jefe
 - El pobre Doctorando
 - El Objetivo
- 2 Nuestro último trabajo
 - Necesidad de Interfaces seguras
 - Nuestra Noción de Seguridad
- 3 Comentarios Finales
 - Más Resultados
 - Aplicabilidad



Porqué las interfaces seguras son necesarias.

Hoy en día es común:

- Diseño de Software basado en componentes.
- Una *interfaz* es una descripción del comportamiento externo del componente.
- Algunas componentes/interfaces manipulan información confidencial.

Entonces es MUY, MUY, MUY, MUY importante definir interfaces seguras.



¿¿Qué es una Interfaz Segura??



¿¿Qué es una Interfaz Segura??

Nosotros definimos nuestra definición de seguridad (informalmente) cómo:

“Una interfaz es segura si la actividad confidencial no puede ser detectada.”



Ejemplo: Credit Request Process

Un banco quiere un nuevo servicio web para su proceso de aprobación de crédito online con los siguientes requerimientos:

- El proceso de aprobación puede ser computado localmente o por otro componente.
- El servicio puede ser configurado por un Administrador (*Usuario Alto*) para que realice sólo control local.
- Un cliente (*Usuario Bajo*) no debe poder detectar que el sistema está realizando sólo control local.

Esto es un requerimiento de seguridad! Un cliente no puede detectar cierta actividad confidencial.



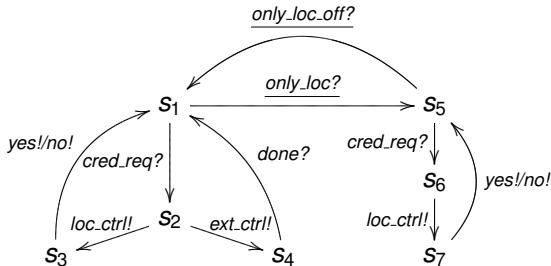
Ejemplo: Credit Request Process

... y una empresa de software presento el siguiente modelo
para una interfaz!!

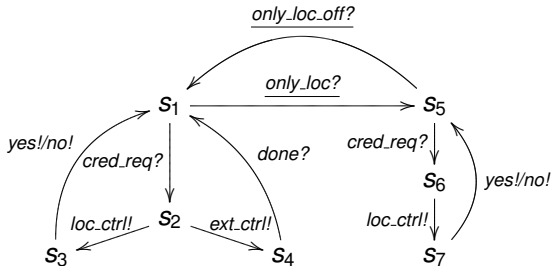
ta, taaaa, ta taannnnnn!!!



Ejemplo: Credit Request Process



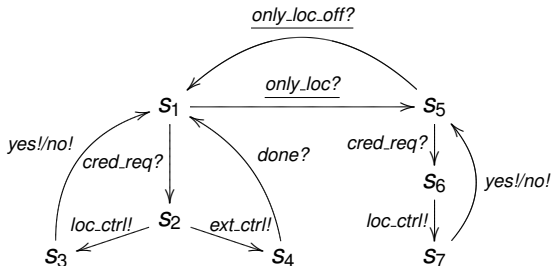
Ejemplo: Credit Request Process



- Esto es Interface Structure for Security (ISS). Tiene acciones de Salida!, de Entrada? y Ocultas;
- Una acción visible puede ser Confidencial o no.



Ejemplo: Credit Request Process

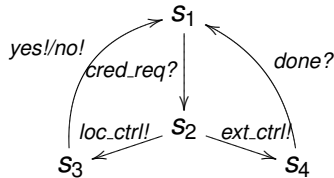


- Las acciones altas no son visibles por los usuarios bajos.
- Las acciones de entrada son detectables mientras que la de salidas no.

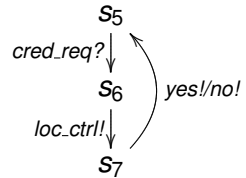


Ejemplo: Credit Request Process

El sistema sin Actividad Alta

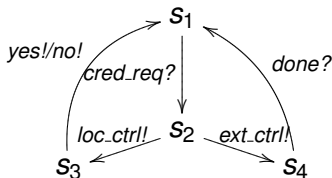


El sistema luego de la Actividad Alta

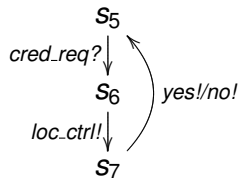


Ejemplo: Credit Request Process

El sistema sin Actividad Alta



El sistema luego de la Actividad Alta

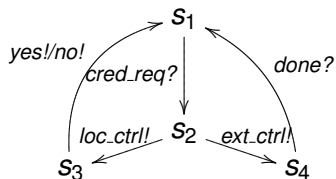


La actividad alta cambia los posibles comportamiento del sistema, lo cual podría ser detectable y esto ser un indicio de que el sistema no es seguro ...

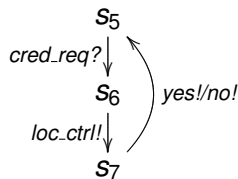


Ejemplo: Credit Request Process

El sistema sin Actividad Alta



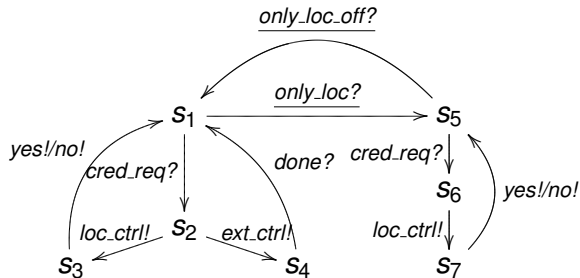
El sistema luego de la Actividad Alta



...por suerte no lo es :D



Ejemplo: Credit Request Process

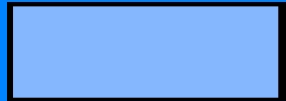


Ejemplo: Credit Request Process

Inputs Enables



Output



Made in Argentina

Chun Interfaces Corp.



Ejemplo: Credit Request Process

Inputs Enables

cred_req

Output

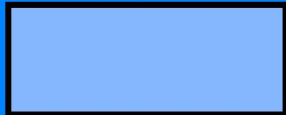
Made in Argentina

Chun Interfaces Corp.



Ejemplo: Credit Request Process

Inputs Enables



Output



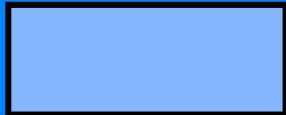
Made in Argentina

Chun Interfaces Corp.

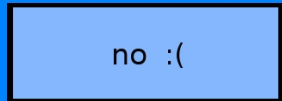


Ejemplo: Credit Request Process

Inputs Enables

A light blue rectangular box with a black border, representing an input field that is currently empty.

Output

A light blue rectangular box with a black border, containing the text "no :(" in a black, monospaced font.

Made in Argentina

Chun Interfaces Corp.

Ejemplo: Credit Request Process

Inputs Enables

cred_req

Output

Made in Argentina

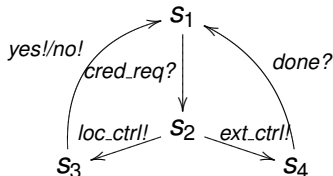
Chun Interfaces Corp.

Ejemplo: Credit Request Process

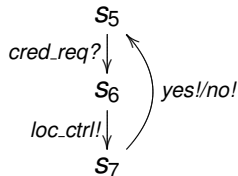
¿¿¿En cual de los sistemas se ejecuto el último ejemplo???

Espero que no lo puedan responder... :)

Sin actividad Confidencial



Con Actividad Confidencial

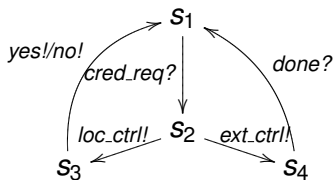


Ejemplo: Credit Request Process

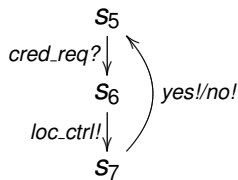
¿¿¿En cual de los sistemas se ejecuto el último ejemplo???

Espero que no lo puedan responder... :)

Sin actividad Confidencial



Con Actividad Confidencial



¡¡El sistema tiene que ser considerado seguro!!



Ejemplo: Credit Request Process

En nuestro último trabajo definimos formalmente las ideas
presentadas y más!!!

.... muuuuuuucho más!!

bueno... quizás no tanto... :)



Outline

- 1 El Director, El Doctorando y El Objetivo
 - El Malvado Jefe
 - El pobre Doctorando
 - El Objetivo
- 2 Nuestro último trabajo
 - Necesidad de Interfaces seguras
 - Nuestra Noción de Seguridad
- 3 Comentarios Finales
 - Más Resultados
 - Aplicabilidad



- El nuevo enfoque resuelve problemas con respecto a las acciones de entrada.
- Mostramos que la propiedad de seguridad presentada (SIR-SNNI / SIR-NNI) no es preservada por la composición pero damos condiciones suficientes para asegurarlas. **Compositionality**
- Presentamos dos algoritmos:
 - El primero determina si la propiedad es satisfecha. **Decidability**
 - Si no es satisfecha, el segundo detecta la existencia de un conjunto de acciones de entradas que al eliminarlas se obtiene una interfaz segura. **Synthesis**
- Y algunos otros resultados menores...



Aplicabilidad en el contexto de Web Service.

- Un IA puede construirse en función de una descripción de un servicio web en lenguaje OWL-S. S. Hashemian and F. Mavaddat, *A graph-based approach to web services composition*.
- Se han desarrollado técnicas para adaptar interfaces. M. Dumas, M. Spork, and K. Wang, *Adapt or Perish: Algebra and Visual Notation for Service Interface Adaptation*.
- Resultados en la composición pueden utilizarse para generar “contratos para garantizar composición segura”. K. Khan, J. Han, and Y. Zheng, *A framework for an active interface to characterise compositional security contracts of software components*.





¿¿Preguntas??
... no da el tiempo!
¿¿Alguién despierto??

