

# **Scalable Processing and mining of Complex Events for Security-analytics**



**Secur'IT kickoff – 8 sept 2015**

# Security Information and Event Management



- Existing tools for security monitoring
  - SIM/SEM
  - (Near) real-time event monitoring
  - Event correlation
  - Data and user monitoring
  - Behavioral profiling & Anomaly detection



SIEM Magic Quadrant - Gartner (July 2015)

# Observations: Current SIEMs



- Highly specialized systems
  - Mainly aimed towards network security
  - Integration with custom applications non-trivial and often not supported
  - Limited flexibility: predefined rules/rule templates
- Non-trivial to deploy
  - Limited support for custom rule definition and rule adaptation in a changing environment

# SIEM in a broader context



- Monitoring and analyzing system behavior is crucial for security operations in many other domains:
  - Fraud detection in financial transactions
  - Sensitive data protection (e.g., customer support centre)
  - Industrial control systems
  - ...

# Complex Event Processing (1/2)



- Setting: continuously arriving stream of events
- Goal: discern high-level events based on low-level events

# Complex Event Processing (1/2)



- Setting: continuously arriving stream of events
- Goal: discern high-level events based on low-level events

T  
I  
M  
E

```
connection from 192.168.10.4 dest_port=21
connection from 190.168.82.4 dest_port=80
connection from 192.168.10.4 dest_port=24
connection from 198.72.245.4 dest_port=80
connection from 190.168.82.4 dest_port=80
connection from 192.168.10.4 dest_port=45
```

# Complex Event Processing (1/2)



- Setting: continuously arriving stream of events
- Goal: discern high-level events based on low-level events

T  
I  
M  
E

↓

```
connection from 192.168.10.4 dest_port=21
connection from 190.168.82.4 dest_port=80
connection from 192.168.10.4 dest_port=24
connection from 198.72.245.4 dest_port=80
connection from 190.168.82.4 dest_port=80
connection from 192.168.10.4 dest_port=45
```

# Complex Event Processing (1/2)



- Setting: continuously arriving stream of events
- Goal: discern high-level events based on low-level events





# Complex Event Processing (2/2)



- CEP language: describe composite events

Upon

(Repeated 5 times:

$(\text{Login}(\text{User}, \text{Term1}); \text{Login}(\text{User}, \text{Term2}))_{[5 \text{ min}]} \& \text{Term1} \langle \rangle \text{Term2}_{[2 \text{ days}]}$

*Deduce: UserAccountHacked(User)*

- Many different languages and online CEP evaluation engines exist
  - The “big data” era has recently introduced new insights

# SPICES Goals



- Design of an **open**, reusable platform for **complex event mining, processing (CEP) and analytics**
  - More **generic** and flexible
  - Easy to deploy, integrate with other applications
- Supporting the complete system's life-cycle
  - CEP Rule **mining**
  - CEP Rule **execution**
  - CEP Rule **evolution**

# SPICES Technical Objectives



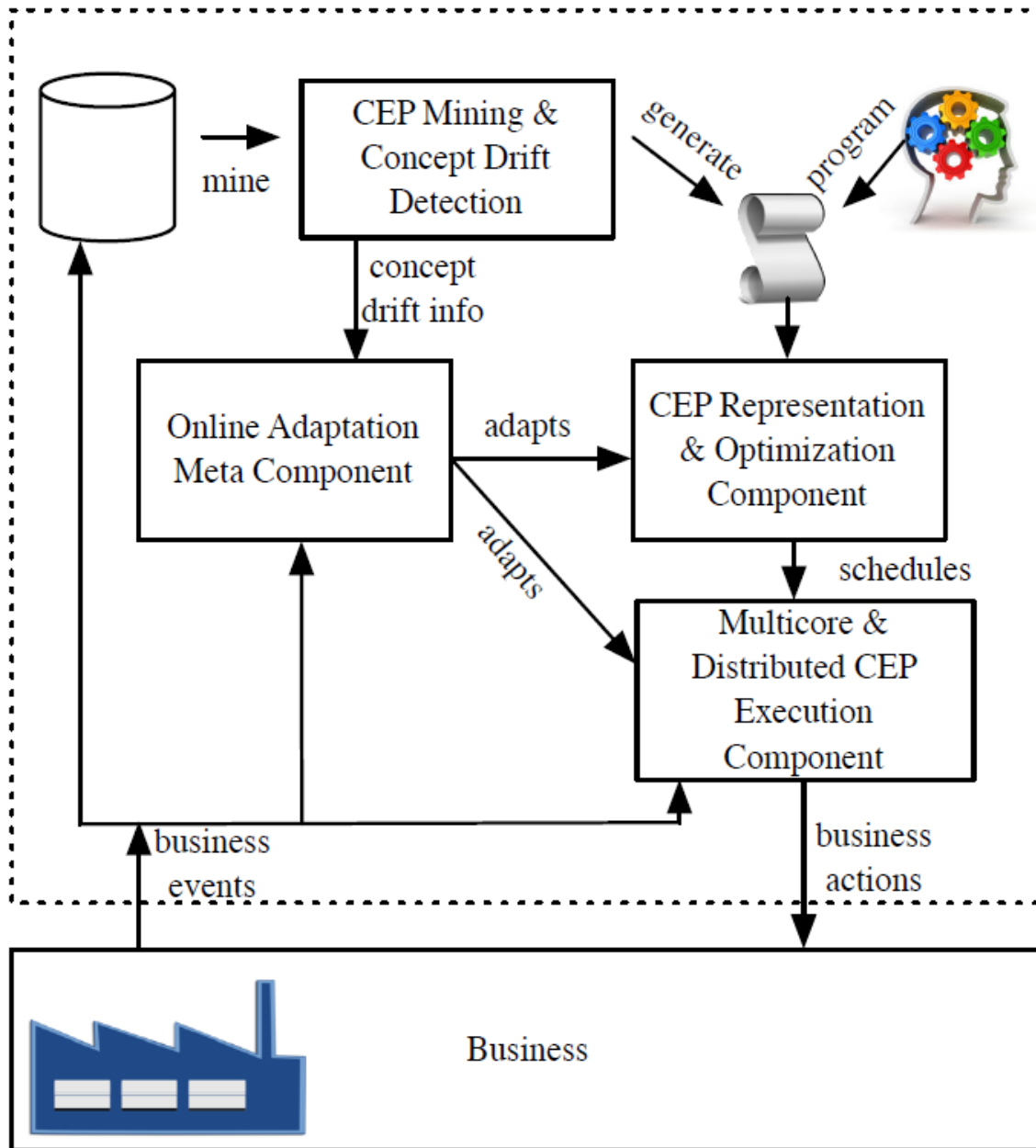
Design of an  
Integrated, high-  
level CEP  
specification  
language

Offline mining of  
CEP patterns based  
on historical,  
labeled data

Online  
“relearning” and  
adaptation of  
mined patterns

Scalable CEP  
execution  
(automatic optimization,  
multi-core, distributed)

Gracefully adapt to  
changes in rulesets  
and workloads



**Generic**  
framework

Based on  
new CEP  
language

Bus  
architecture

# SPICES Team (1/3)



ULB

**CODE/WIT/DM**



Toon Calders



Boris Cule

## Data Mining:

- Rule induction  
(pattern mining for sequence data)
- Concept drift detection and rule modification

# SPICES Team (2/3)



## SOFTWARE LANGUAGES LAB



Wolfgang  
De Meuter



Lode Hoste

### Software Engineering:

- Multicore computing
- Adaptation through Dynamic Meta-programming

# SPICES Team (3/3)



ULB

**CODE/WIT/QP**



Stijn Vansummeren



Martin Ugarte

## Database Query Processing:

- CEP Language design
- CEP rule optimization and scalable execution

# Our User Committee



- Community of interested parties from industry
- We aim for close contact with User Committee
  - Requirements analysis and driver scenarios
  - Technology audit: input from sponsors/market
  - Knowledge transfer: tutorials at half-yearly events
  - Proof of concept implementation
- Interested? Please join!



# SPICES - Summary



- Development of an open, reusable platform for security event monitoring
  - Centered around CEP
  - Declarative, expressive CEP language
  - Offline rule induction
  - Online drift detection and rule adaptation
  - Scalable and adaptive execution engine through query optimization and reactive meta-programming
- Bringing together necessary skills in Data Mining, Querying, and Software Engineering