SAFEGUARD EU-IST-2001-32685; Dec 2001 – May 2004

swisscom innovations

Electrical & Telecom Dependability & Security

Partners: ENEA, LIU, QMUL, AIA

Topic:

Alarm Reduction and Correlation in IDS

Authors:

Tobias Chyssler, Kalle Burbeck (LIU)

Stefan Burschka, Michael Semling, Tomas Lingval (INO)

Contents

- Safeguard Project
- Our Situation: Network operations
- Our Approach
- Normalisation, static and adaptive Filtering
- Aggregation
- Correlation
- Future
- Summary

Safeguard IST Project

Protection of Large Critical Infrastructures (LCCI)

- Automatic detection of and defence against known and unknown attacks, misconfigurations and failures
- **Support of administrators using IDS**
  - **Information Overload in IDS**
    - **Filtering, Aggregaton, Correlation**
    - **Enhancing time critical decision support & reactive capabilities**
- Network and component failures as well as their prevention
  - Selfhealing, graceful degradation

Situation: Network operations

Appropriate perimeter defence exists:

Attacks < 0.1%, Misconfigs = 70%, Failures = 30%

→ Loss of time and money

- Dynamic Environment Machines, services and responsibles change
- Huge number of network elements and machines
- Various and sometimes unkown network interconnections
- Fragmentation of knowledge
- Lacking control of policy compliance
- Too complex human decision chains

- Information overload, meaningless IDS Alarms
- Reaction time too high → breakdown

Approach

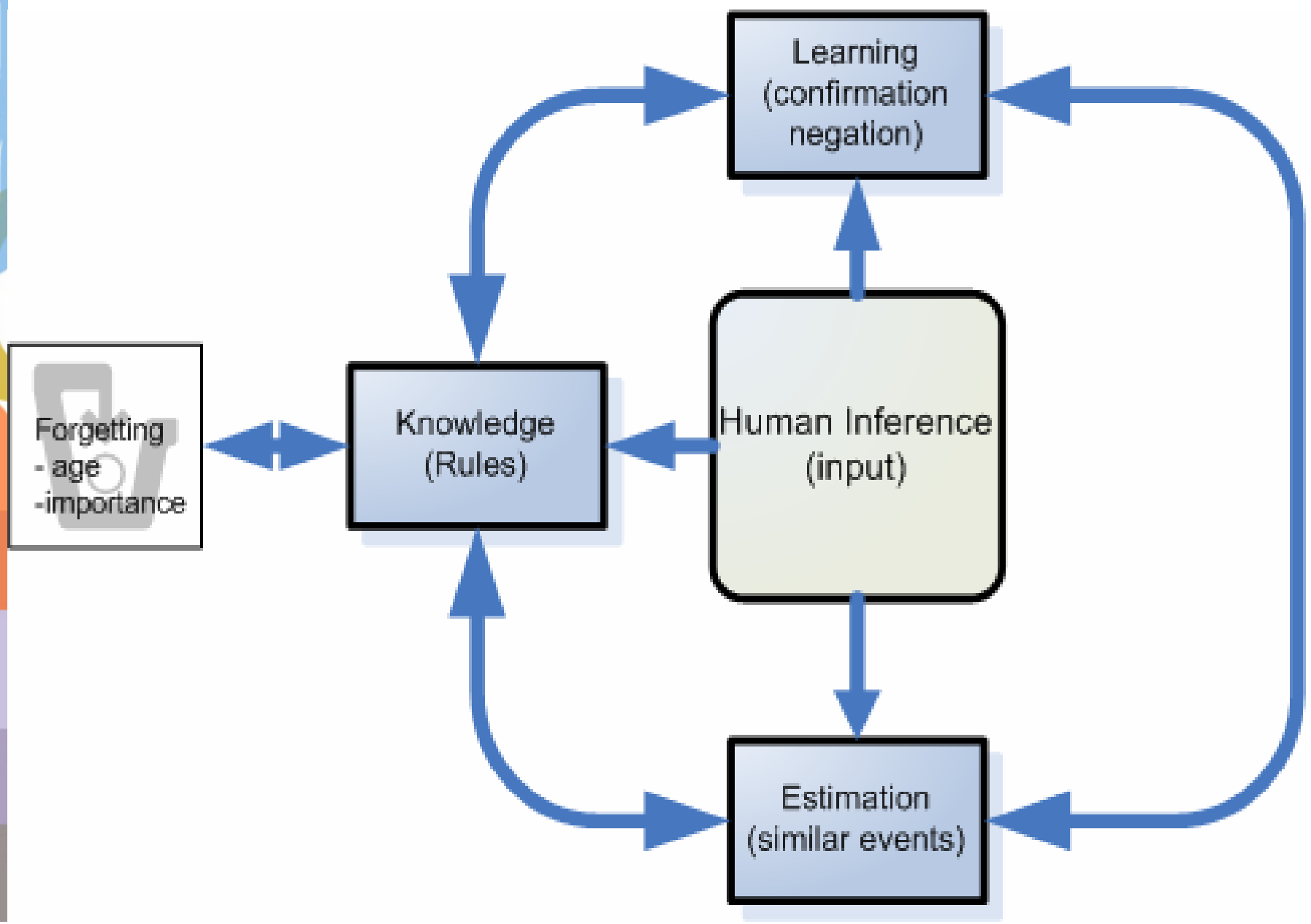
appropriate steps in critical situations

- Preprocessing of Alarms:
  - Filtering, feature extraction, aggregation

  Support for data analyist:
  - (Un-)Interesting alarms
  - Appropriate reactions
  - Information extraction of alarm data
  - Good features
  - Up-to-date topology and policy infomation

- Automatic response to confine damage
  - Auto immune reactions

Our Approach: Admin Task



Our Approach:

Human Inference Methods

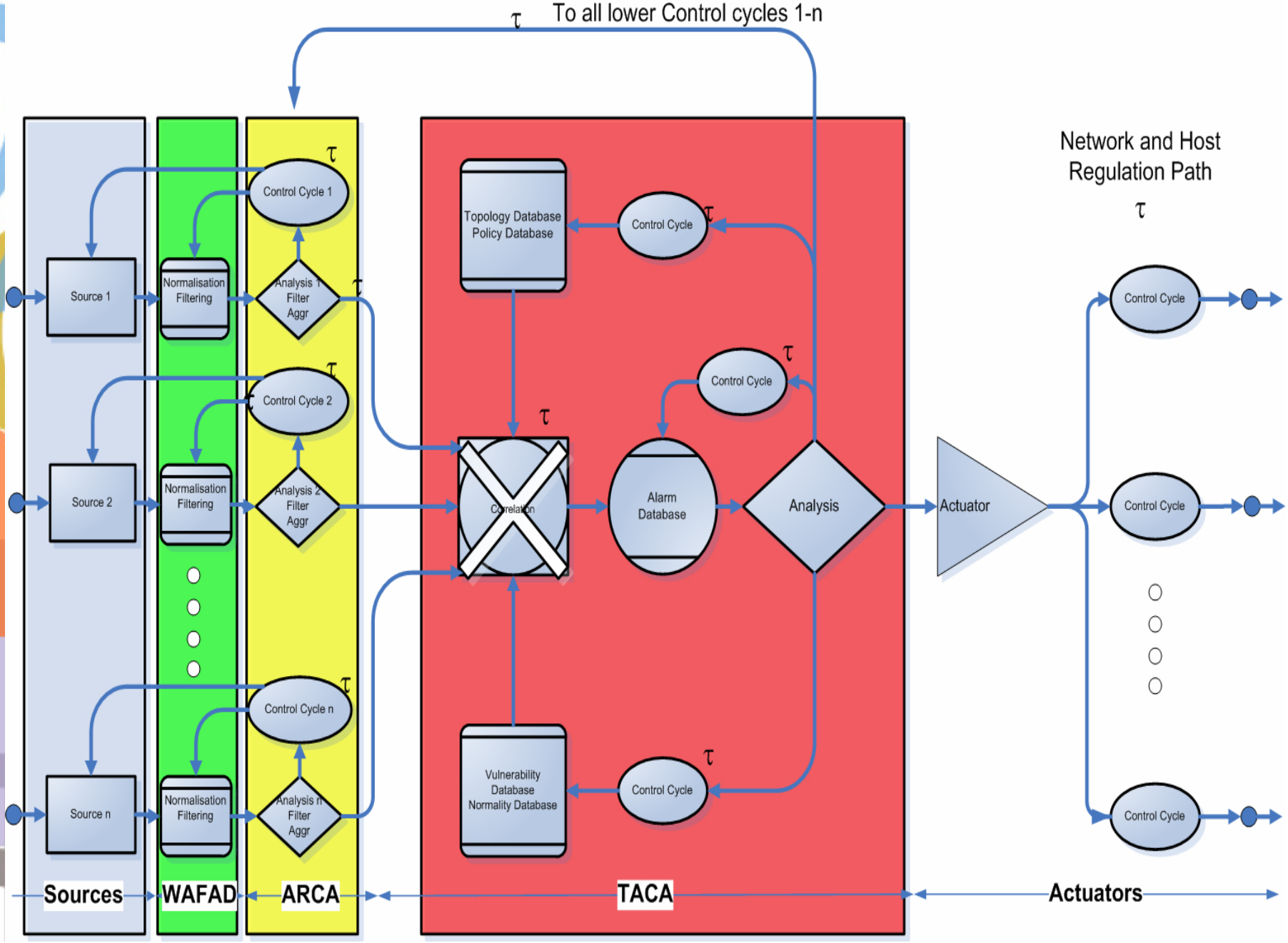
Classify and filter unknown alarms first	Faster for unknown Attacks FP elimination better
Filter known and suspicious alarms first	Faster for TP FP elimination slower

Basic Alarm features:

- Severity (Alert, critical, debug, error, warning, info, notice)
- Variety (hosts with a lot of Alarms)
- Number (hosts with many alarms)
- Uniqueness (Unusual Alarms)
- Frequency (Alerts / Minute)
- Payload (Strange Payload in normal Alarms)

- Vulnerability state of the network

Process Diagram



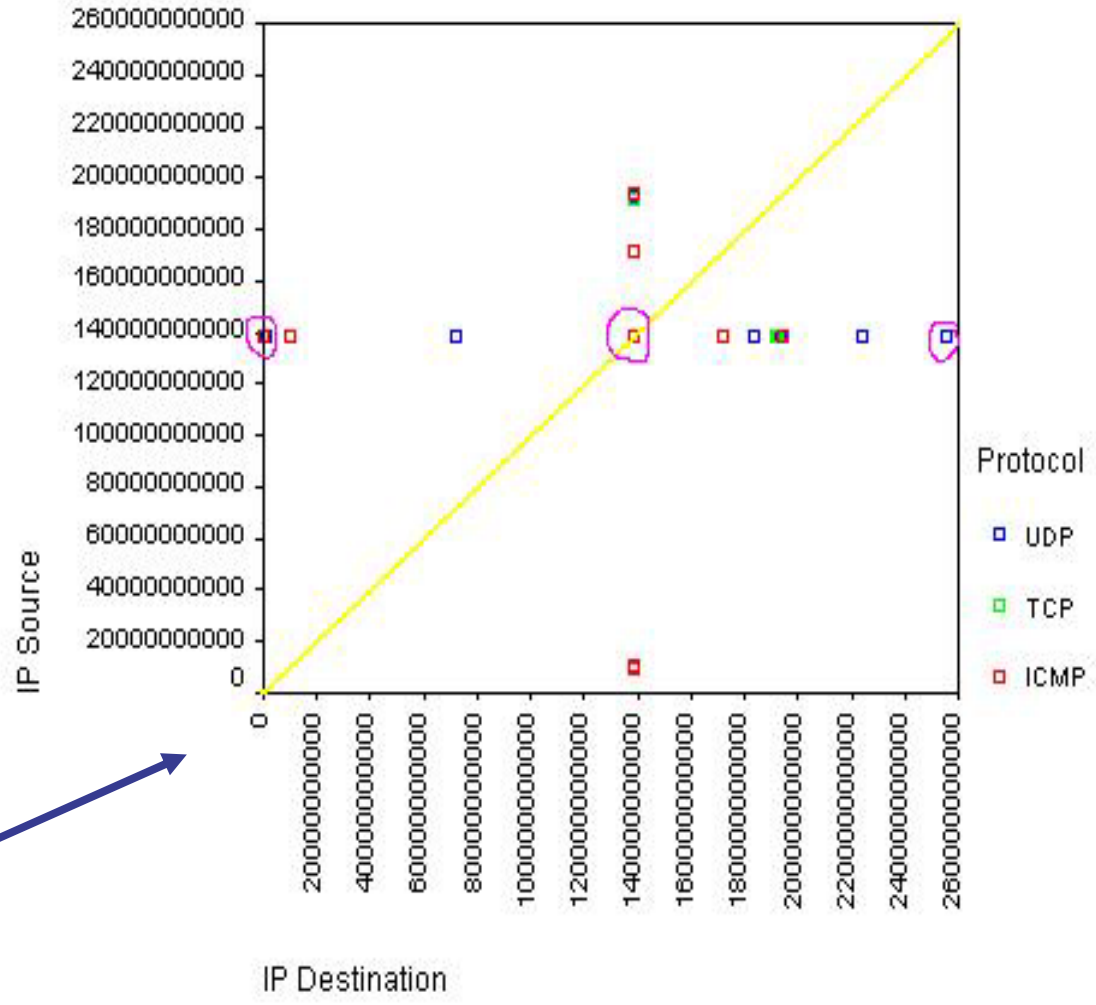
Existing Alarm Sources

(NTP synched)

- NIDS, HID, central logging
- Vulnerability scanner
- Host Health function
- Anomaly detection (Birch clustering)

Information Sources

- Static Information
  - Location, responsible
- Policies
- Vulnerability scans
- SNMP info
- Diversity statistics



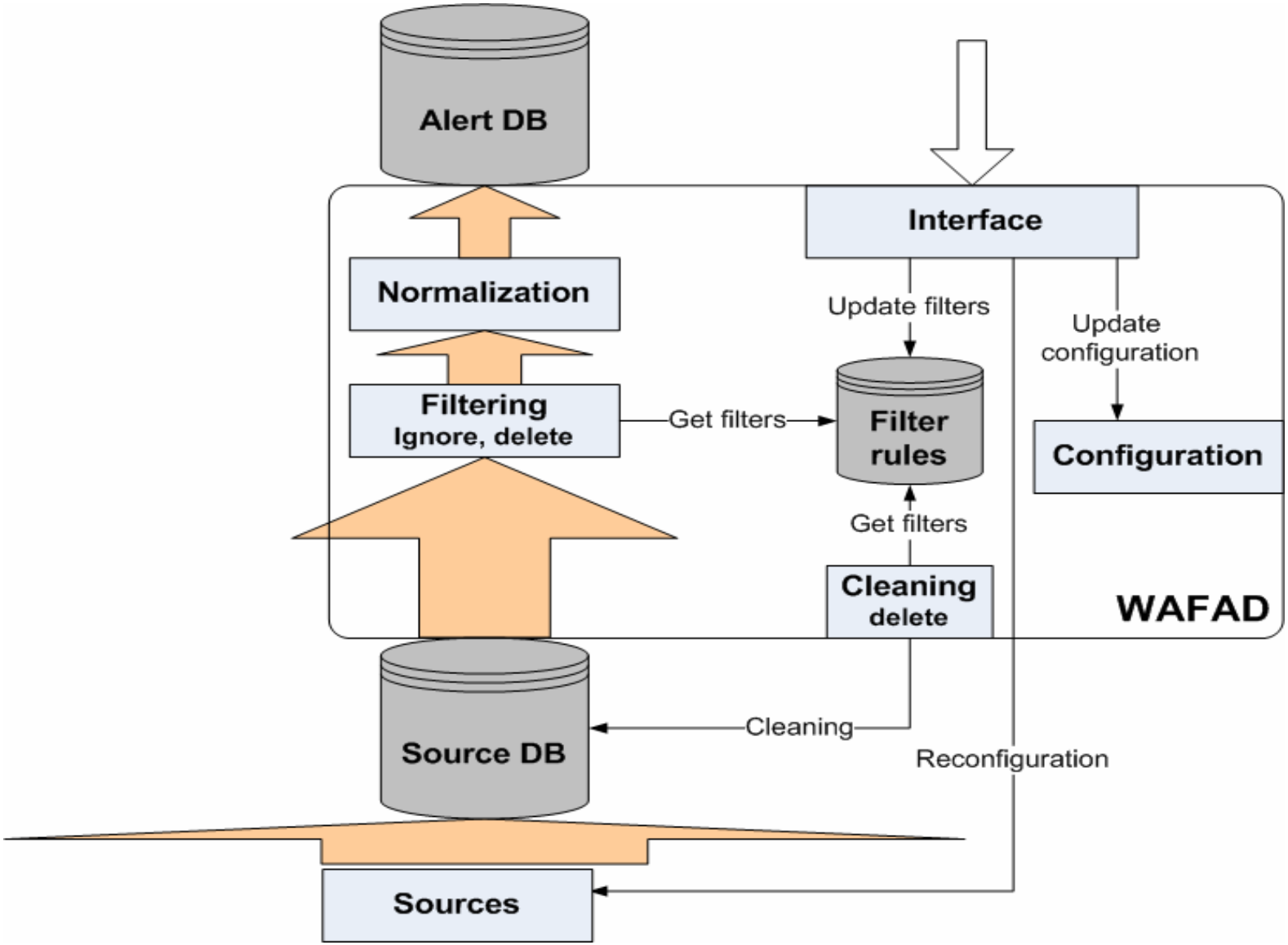
Alarm Reduction Chain

1. WAFAD:
   - Time synchrone normalization and data selection
   - Ignore, delete filters with lifetime

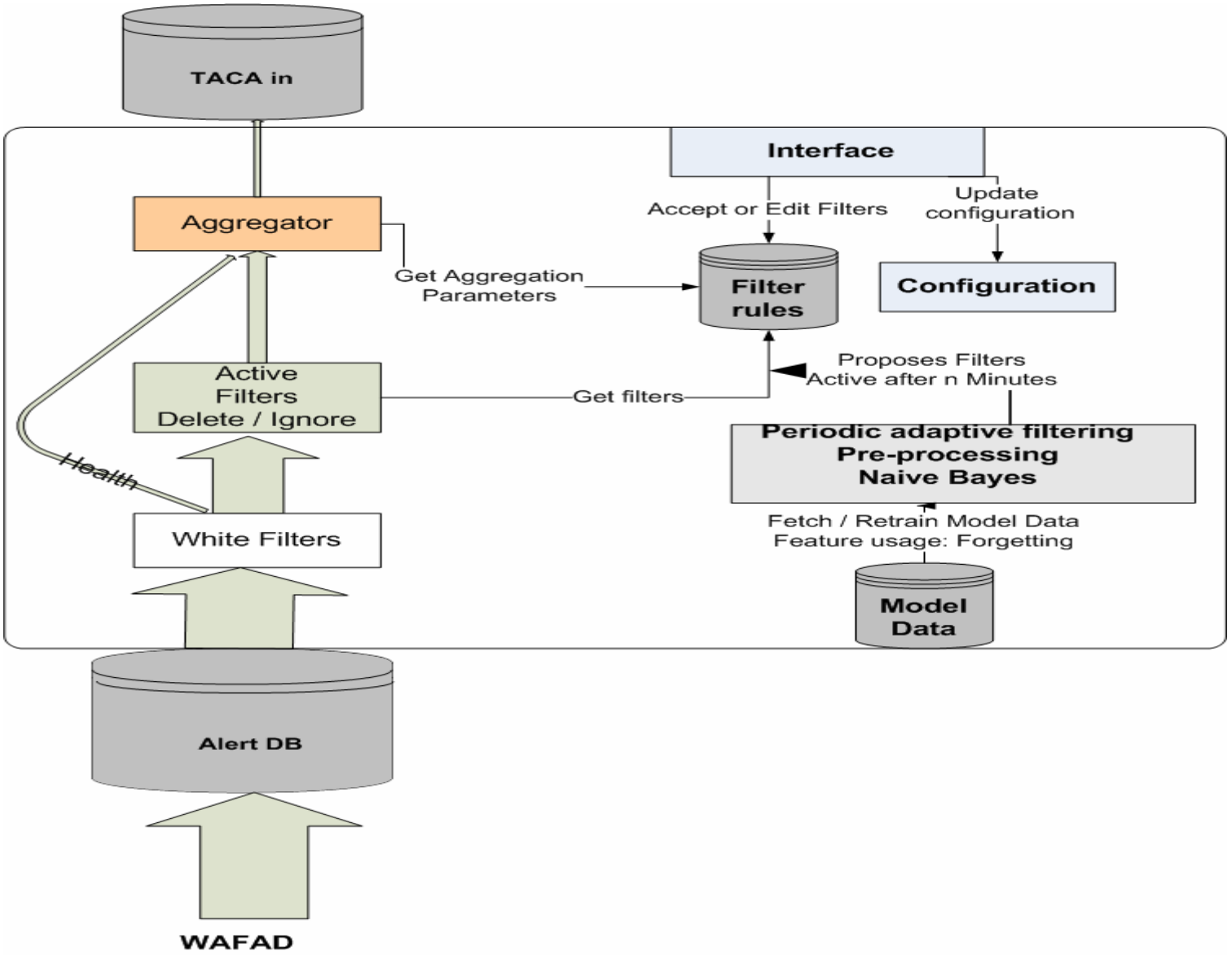
2. ARCA: (filtering)
   - White filters lists
   - Automatic filter proposal

3. ARCA: (Aggregation)
   - Aggregation of similar Alarms (FP, TP reduction)

Wrapper Agent for Alarm Databases (WAFAD)



Alarm Reduction & Correlation Agent (ARCA)

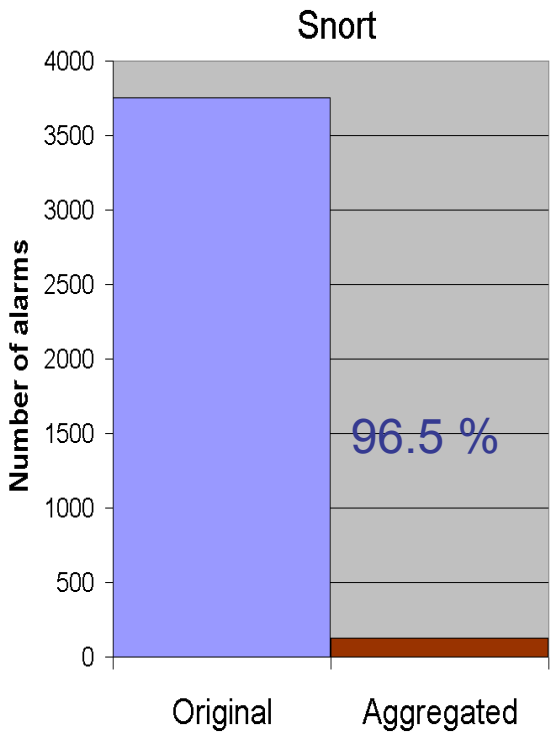


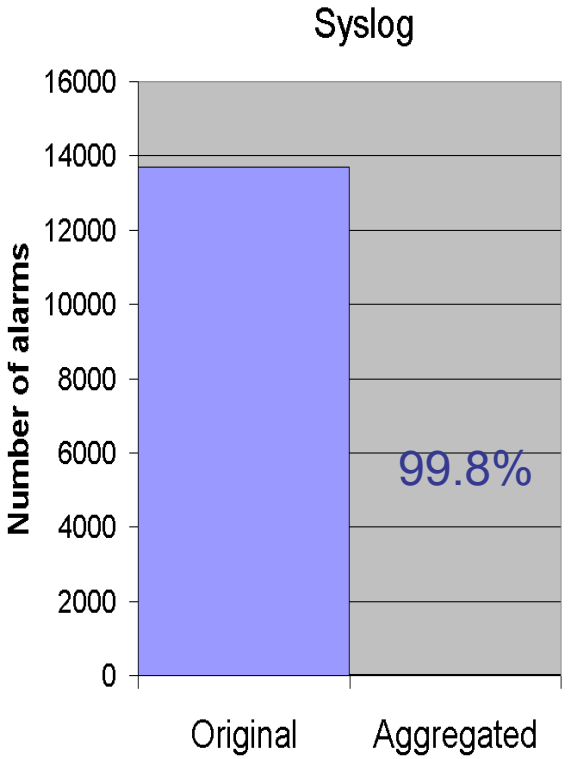
Results: Static filtering, Aggregation

Static filtering (WAFAD) in dynamic environment: 5 - 20% performance

Aggregation on different elements for different sources: IP, message, ...

Method: *Edit Distance for each word. Optimum: 70% similarity*





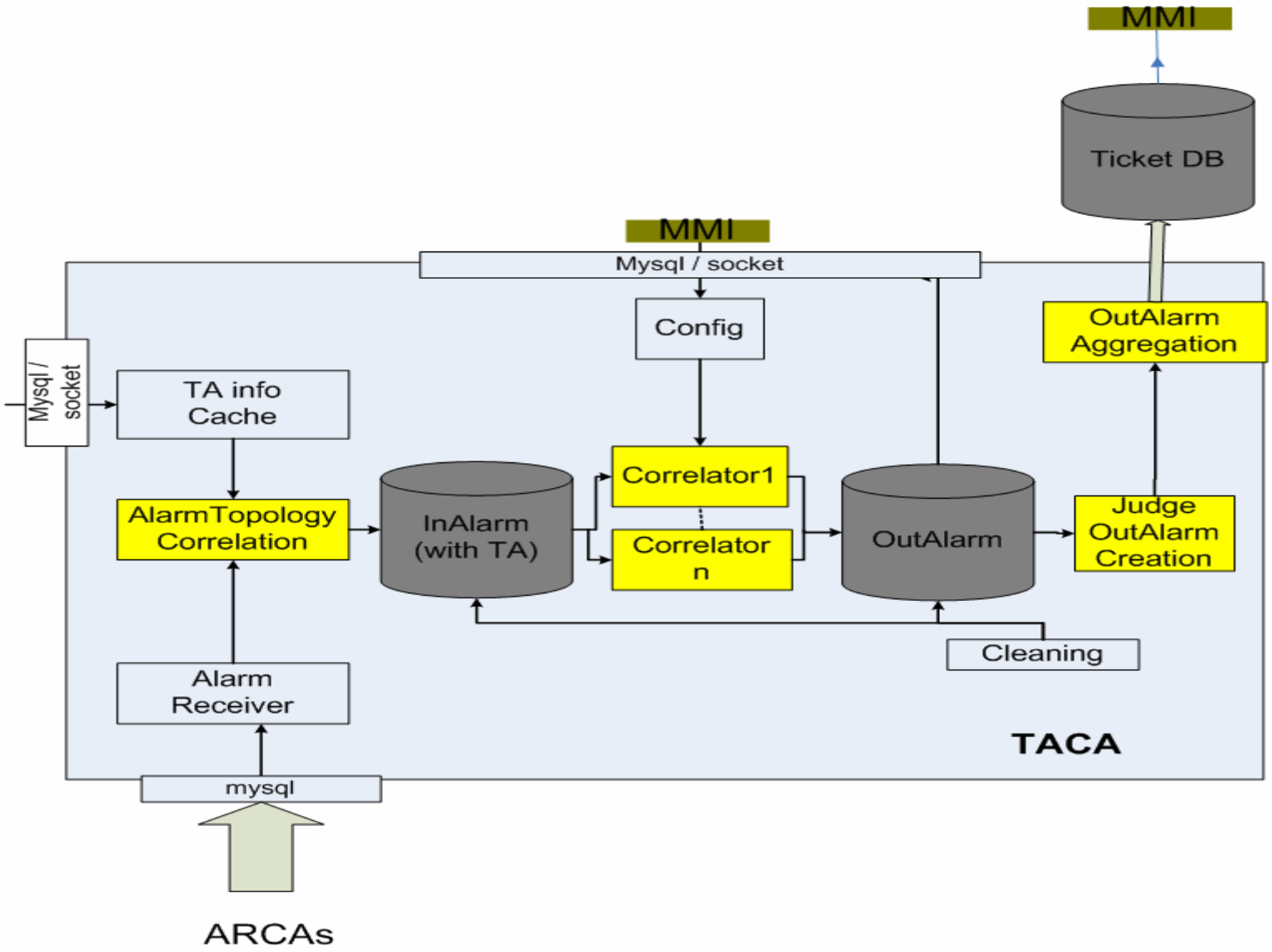
Sources — WAFAD — ARCA — TACA

Results: Naive Bayesian Network

(Dynamic Rule Proposal, supervised training)

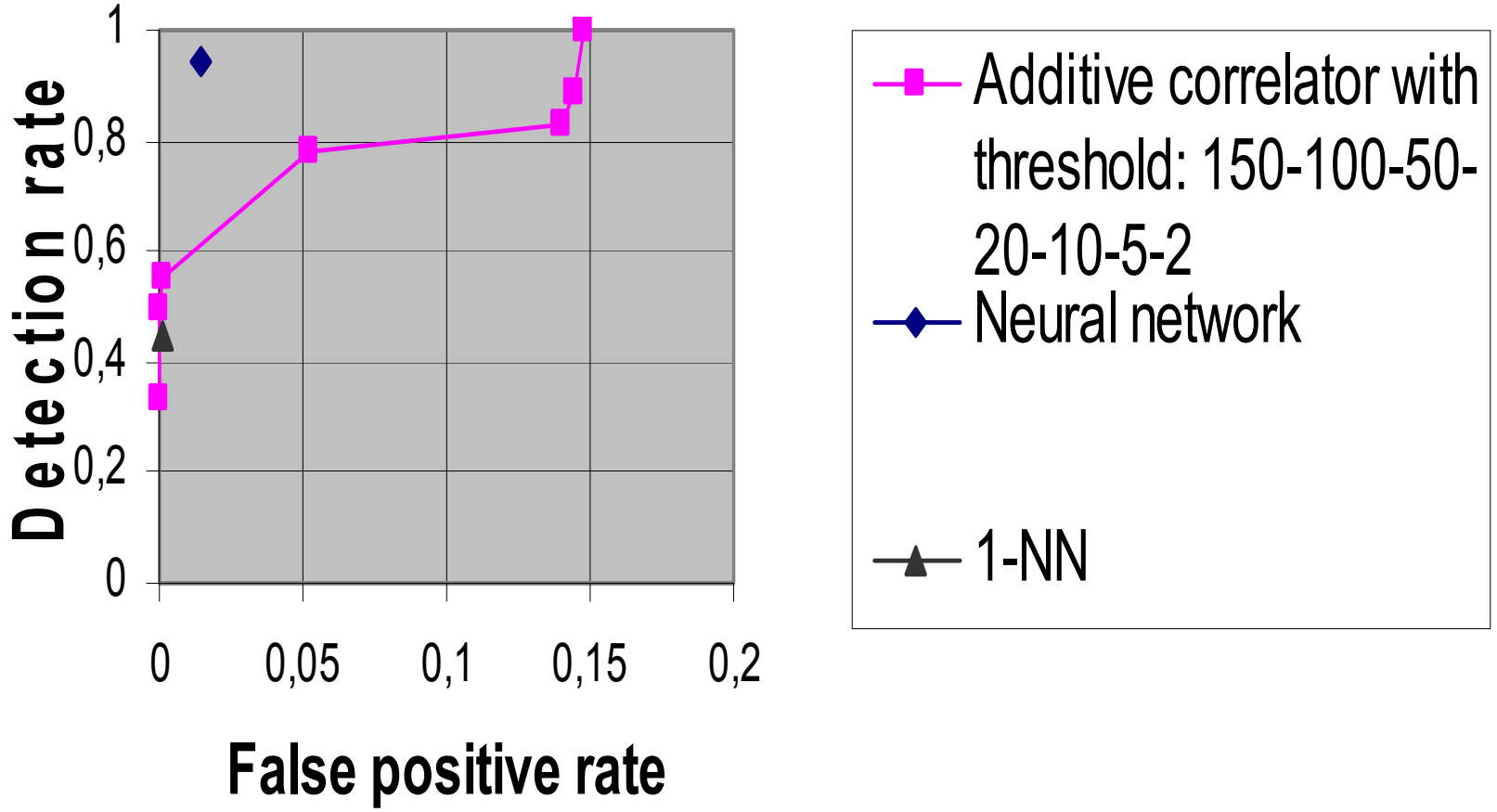
- Model trained on ca 60000 alarms (1/3 dataset)
- Performance enhancement with text classification tricks
- Classes: (Un-)Interesting, decision by human expert
- Cross validation against all other datasets
- Result: > 99% correct classification, but
- Rule proposals were accepted in 65%, edited 20%, discarded 15% by admin.

Topology and Correlation Agent (TACA)



Results: *AI Correlators (Un-)known attack detector*

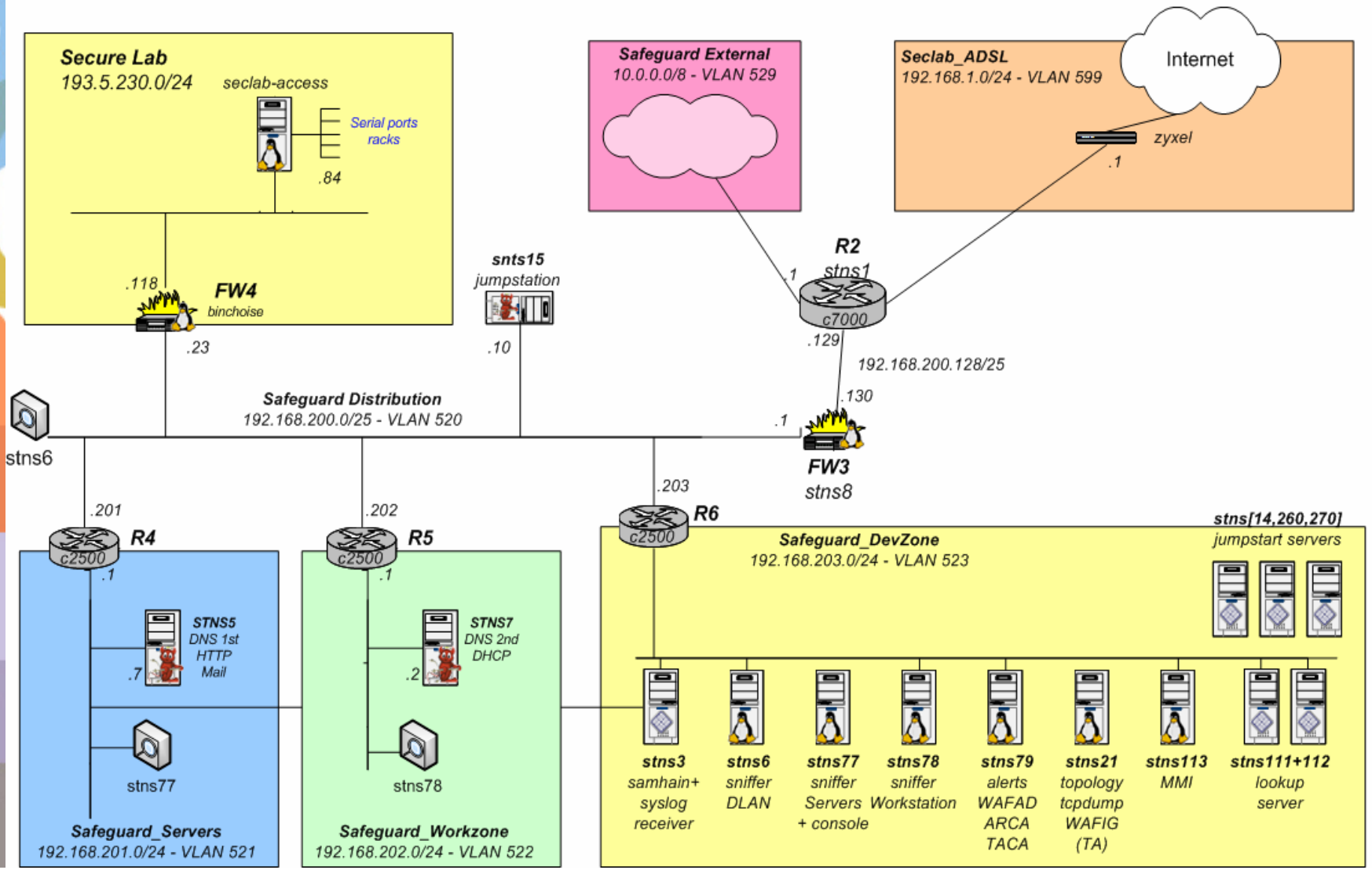
Method: Added Alarm Severities weighted by their frequency in a time window and over different sources



Sources — WAFAD — ARCA — TACA

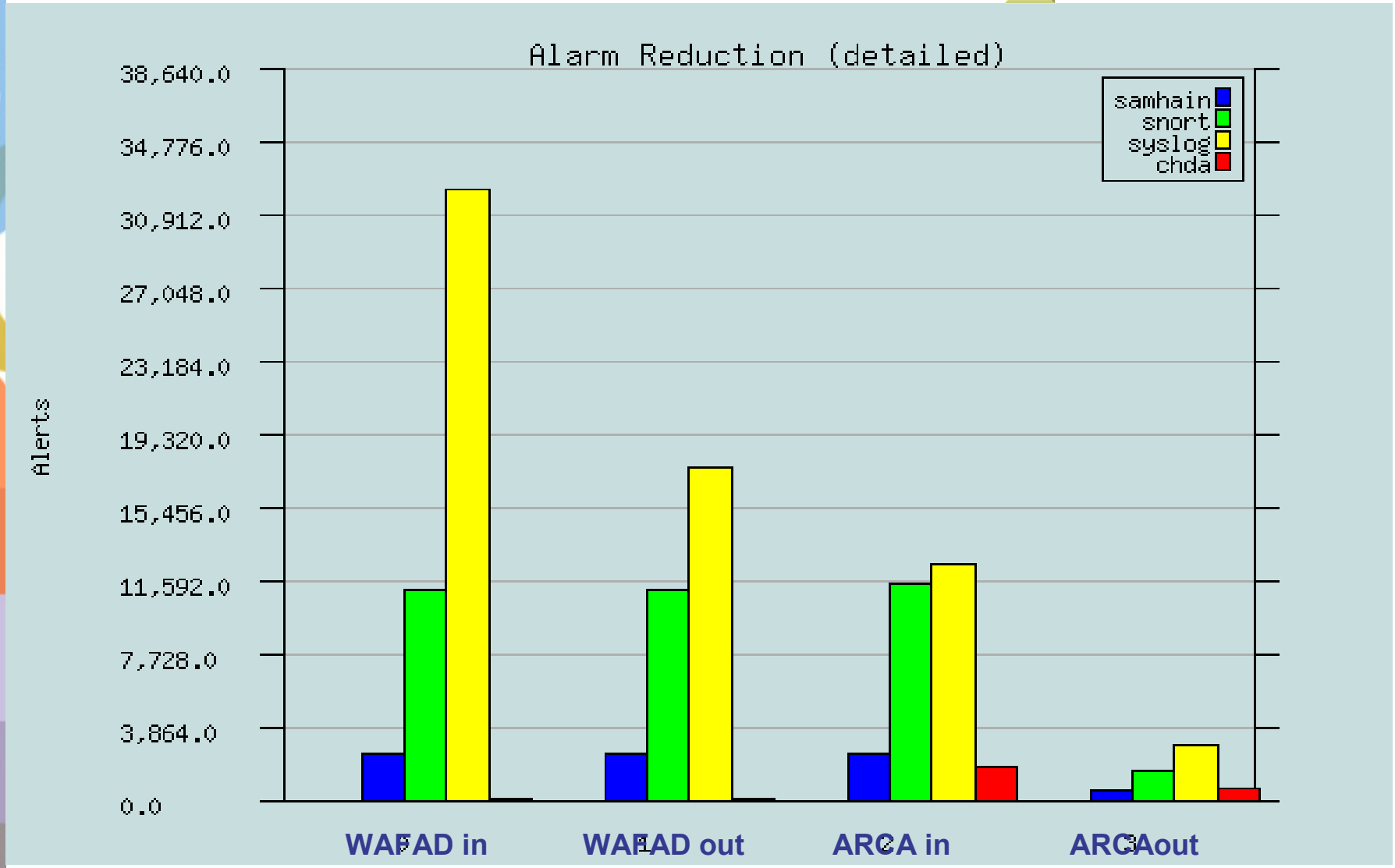
How to create mess?

Take 100 machines, several Admins, different HW, OS and SW



Alarm reduction in Testnet

Safeguard first time on line (AFAD,ARCA, HDA)
no DB cleaning ops, normal misconfig
no additional training of ARCA, no attacks



New Correlation Method:

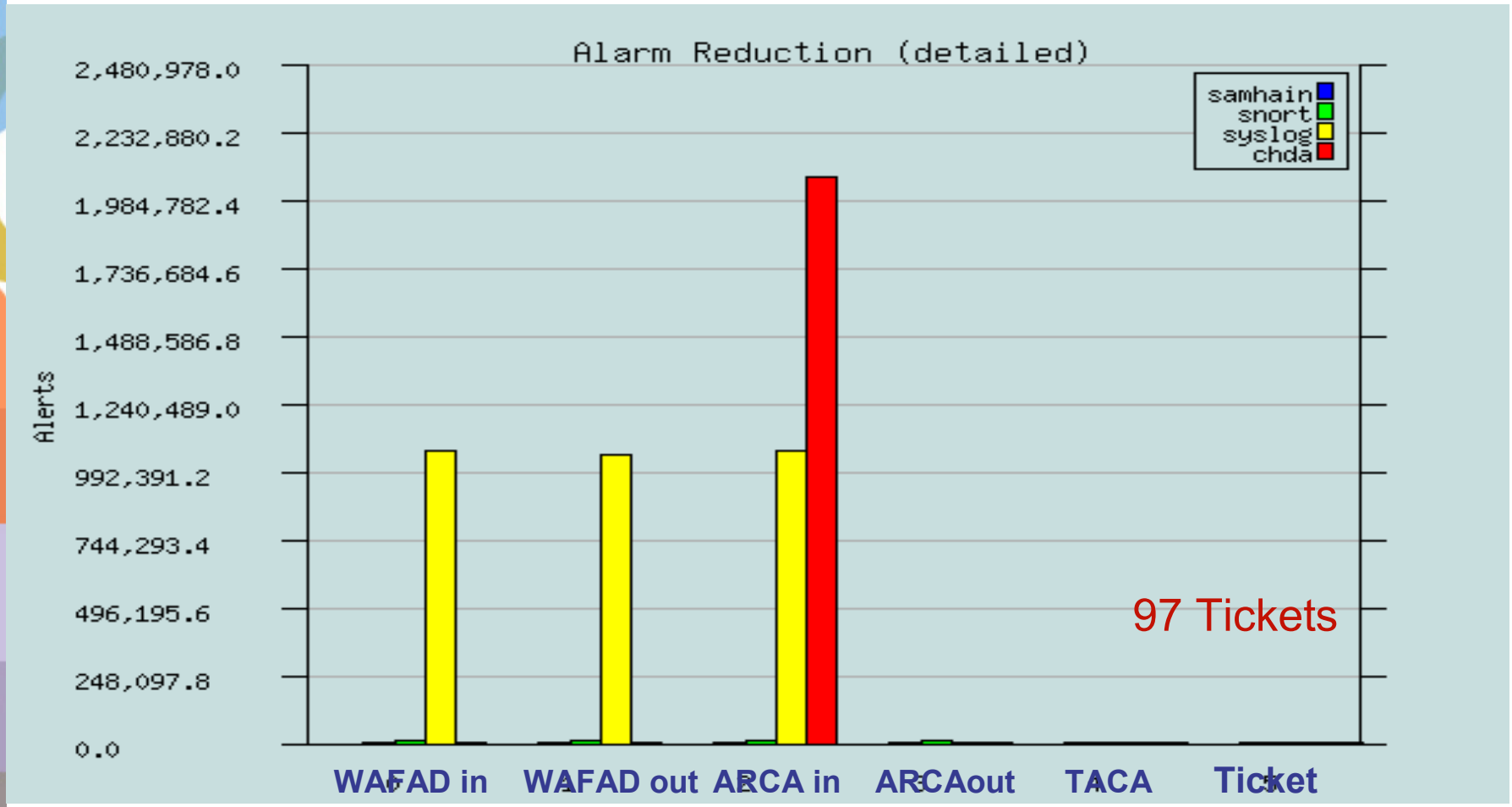
M-Brain-dump in our new test network

- Experiments showed insufficient alarm reduction
- Added topology and vulnerability info correlation
- Added anomaly detection (Birch Clustering)
- All this processed by M-Brain-Dump Correlator:
  - Implements actions and workflow of an Admin
  - Creation of alarms tickets.
  - Total alarm reduction 10^{3} - 10^{5}

Test net experiments:

Many misconfigured machines

DB Cleaning ops running, M-Brain-Dump Correlator



Lessons Learned

- Humans are already anomaly detectors
- KISS works best in real environments
- Distributed systems are a must for a certain network size
- Changes in the systems have to be slow
  (Adaption of algorithms)
- Algorithmic performance:
  Processing time ←→ Classification performance
- The human has to be the final instance in the decision chain (UNDO Button)

Future Research Topics

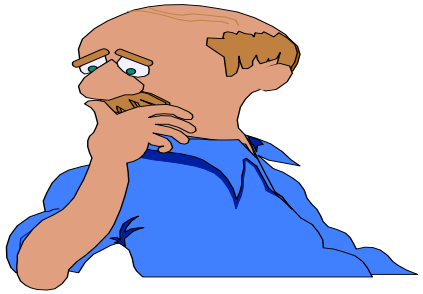
- **Prediction of system health → Graceful degradation**
  - Complex resource regulation
- **Anomaly detection in service content**
- **Handset and embedded systems security**
- **Automatic correlation tree learning, add in of ANN**
- **Improving our test network (+ 200 machines)**
  - Training & test data (annonym, available for everybody)

Summary

We showed:

- An Architecture which mimics the human analysis process
- Alarm reduction 10^{3} - 10^{5} depending on data
- AI - Correlators for (un-)known Attacks
- Real time system confines effects of misconfigurations, failures and attacks in order to guarantee the entire system‘s survival

Questions / Remarks?



Publications:

http://www.ist-safeguard.org

Test net mailing list:

group.safeguard-tnet-ino@swisscom.com

Contacts:

stefan.burschka@swisscom.com

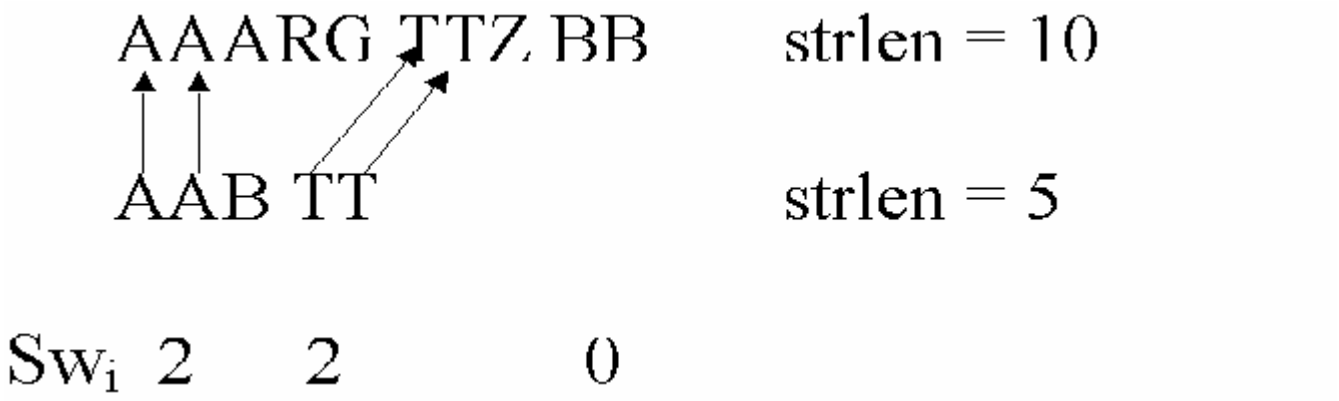
michael.semling@swisscom.com

thomas.dagonnier@swisscom.com

Alarm Aggregation

Which Alarms have similar content

- Aggregation over an arbitrary window size. Forward:
  - 1. Occurrence: The original alarm
  - At end of window one alarm + # of occurences



$$S_g = (\Sigma\ Sw_i) / \max(strlen) = 0.4$$

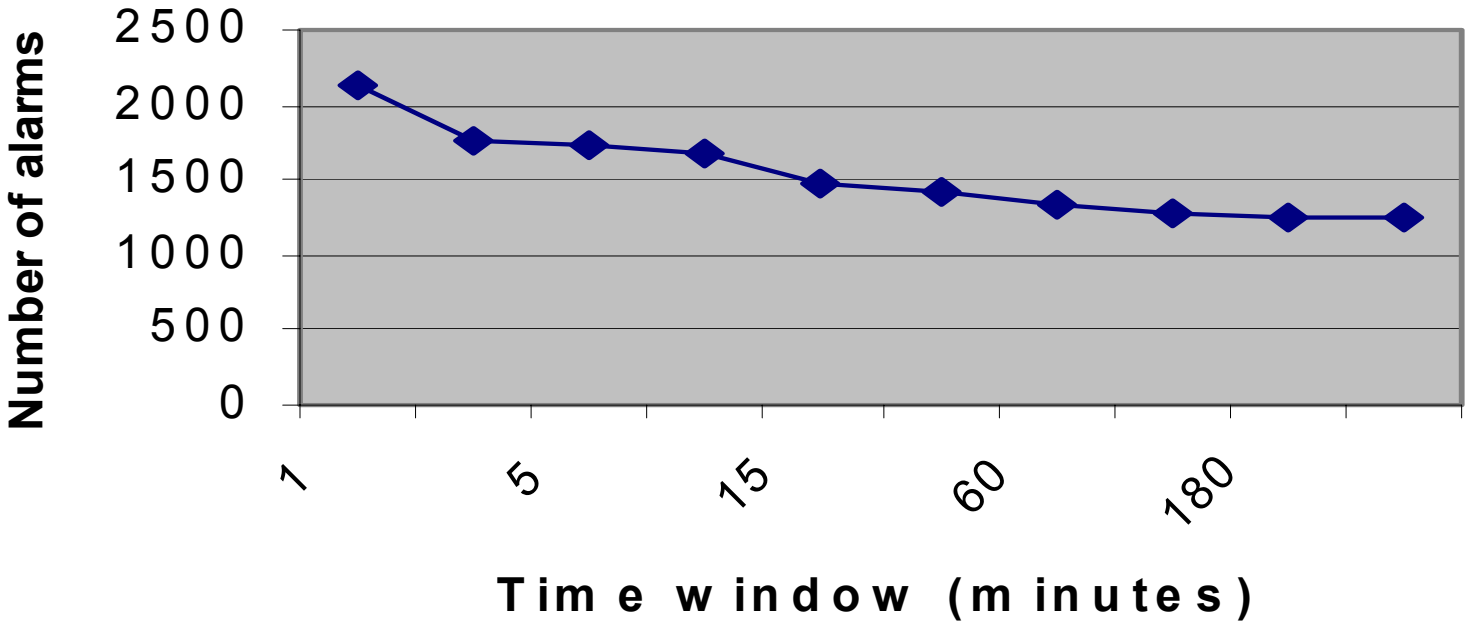
Threshold Sg = 0.65: Same messages recognized as identical

Threshold Sg = 0.7 : Alarm Reduction Snort 96.5 %. Syslog 99.8%

Alarm window size

Influence on Ticket Appearance

- Aggregation can swallow important alarms for a whole time window
- Sg and time window has to be tuned for optimal reactivity, severity dependent



Result: Frequency of messages as a filter?

- Test Data: Same as in Baysian experiment

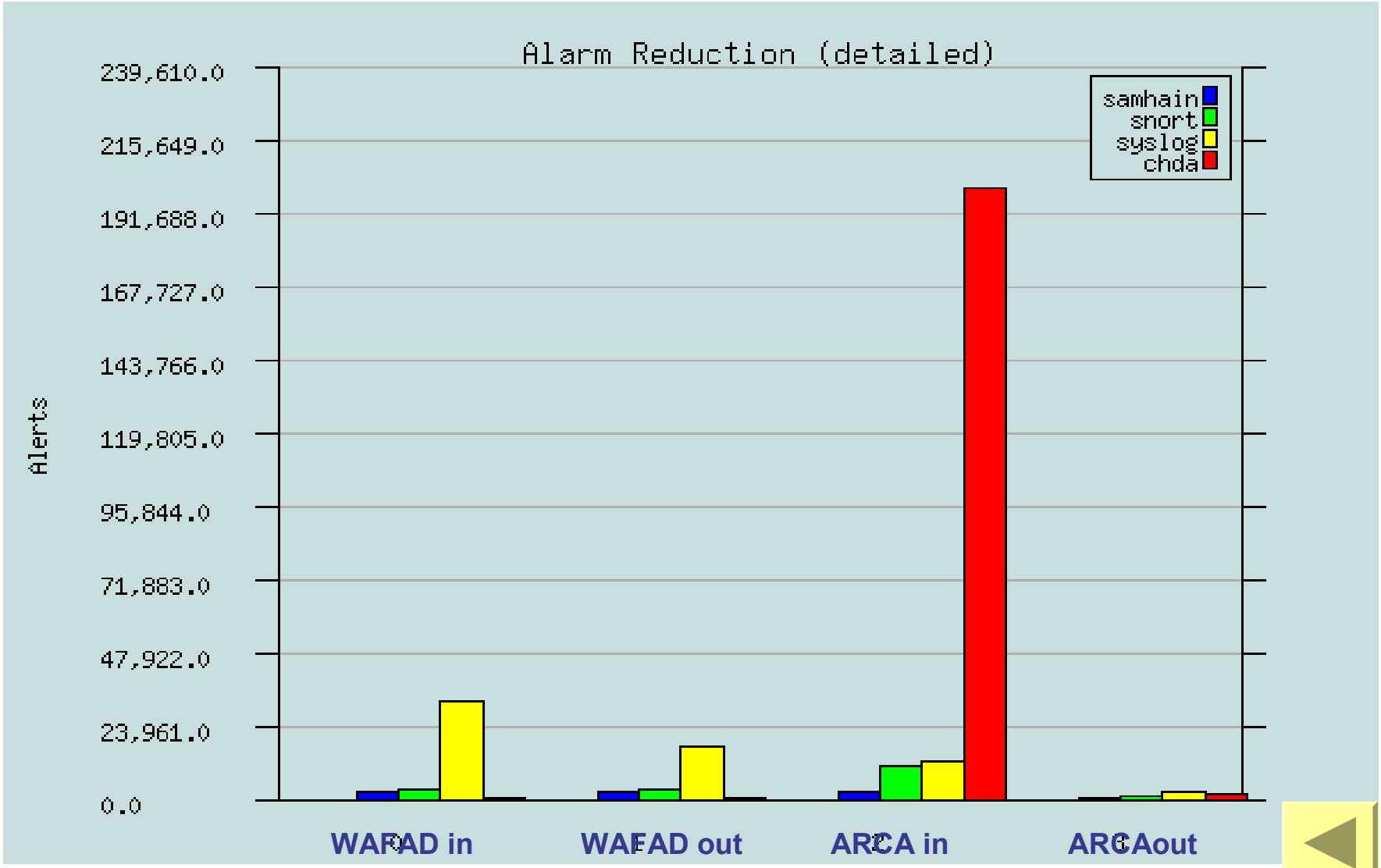
Frequency threshold	0,1	0,05	0,01	0,005	0,001	0,0005	0,0001
Messages removed rate	0,5889	0,8209	0,9558	0,9797	0,9961	0,9984	0,9995
Precision	0,6029	0,8348	0,9698	0,9832	0,9872	0,9862	0,9860
Interesting messages removed	0	0	0	26996	59170	67622	69934

Test Network

- Results so far in a small network, so lets see how is performs without retraining in a realy big messy one
- HW Architectures
  - SUN ULTRA 2,5, 10, Sparc 5,10,20
  - X86 Architecture
  - HP Risc PA
  - Embedded processors (Router, Switch)
- SW Architectures
  - Solaris (2.6 –2.10), all patch levels
  - HP-UX 10.x
  - Open BSD 3.x
  - Windows (95,98,NT,2000,XP,2003), all patch levels
  - Router, switches (IOS)

Alarm Reduction

Attack tools & DB cleaning ops running



Baysian filtering

Tricks to enhance Baysian performance:

- Char garbage filter and Len filter
- Noise decorrelator
- Prior shifter
- Multinominal Kernel: Word Frequencies =1
- Nonlinear function on posteriory probability

Results:

Data set	Correct classfications	Incorrect classfications	Precision
Best configuration for Syslog-adaptive-1	53682	40	0,99926
Best configuration for Syslog-adaptive-2	57942	136	0,99766
Best configuration for Syslog-adaptive-3	62631	85	0,99864

AI Correlation in time slots

	Snort	Samhain	Syslog	Added values
Timeslot n-1	Ping from 192.168.201.131. Severity 1			Snort: 1 Samhain: 0 Syslog: 0
Timeslot n	Portscan from 192.168.201.110. Severity 2 Buffer overflow attempt from 192.168.201.110 . Severity 7	/etc accessed. Severity 3	FTP-server error. Severity 5	Snort: 9 Samhain: 3 Syslog: 5