Komponenten für kooperative Intrusion Detection in dynamischen Koalitionsumgebungen

Marko Jahnke
FGAN/FKIE
Abt. Kommunikation
Neuenahrer Str. 20
D-53343 Wachtberg
jahnke@fgan.de

Unter Mitarbeit von
Sven Henkel,
Michael Bussmann
FGAN/FKIE

Jens Tölle
Universität Bonn

Frank Ausserlechner
FH Koblenz

06. Juli 2004

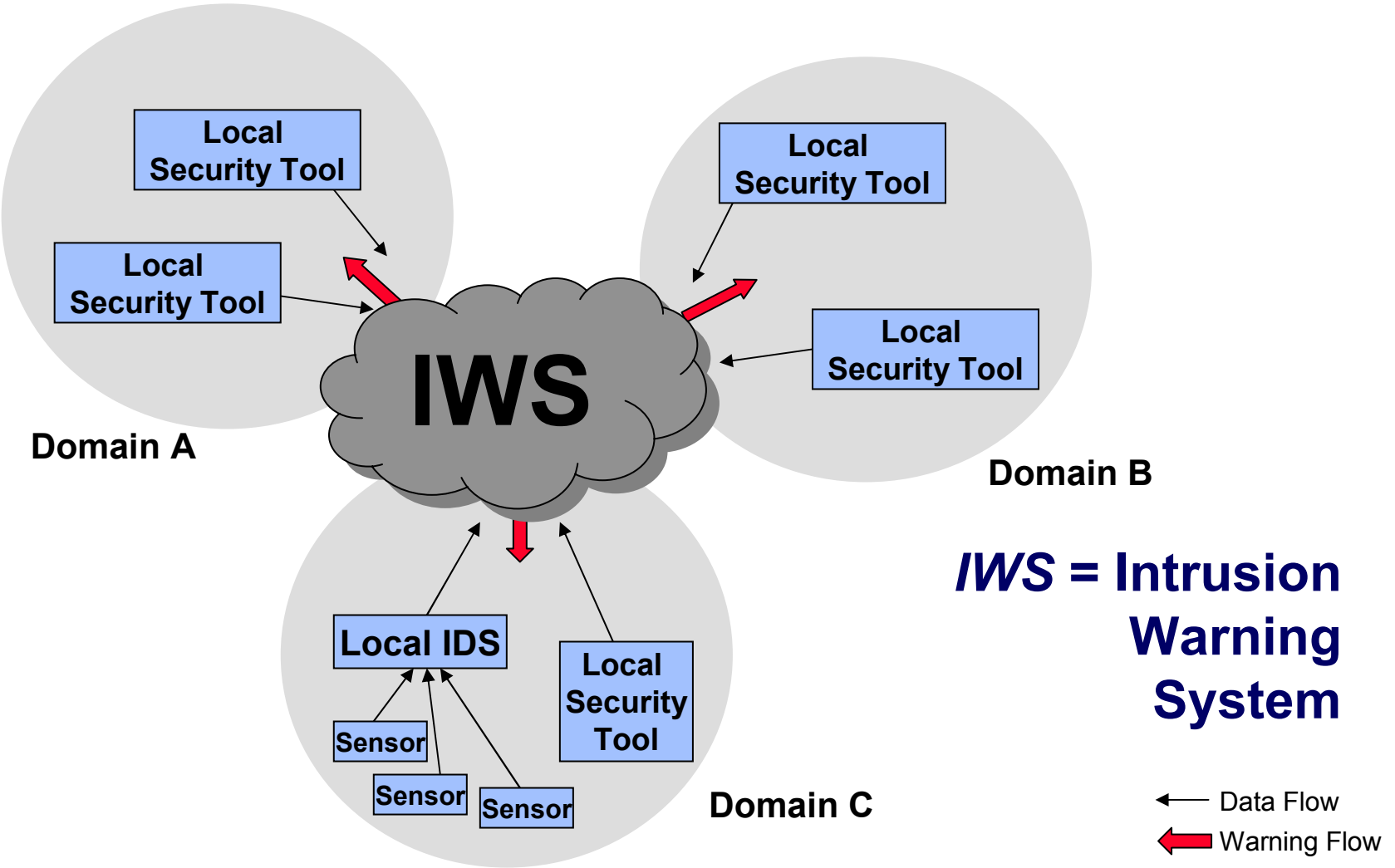
Übersicht

- Herausforderungen für IDS in Koalitionsumgebungen
- Architekturen für kooperative Intrusion Detection
- Vorverarbeitung von Ereignismeldungen
- Anomalieerkennung im Ereignismeldungs-Datenmodell
- *Implementation & bisherige Ergebnisse*

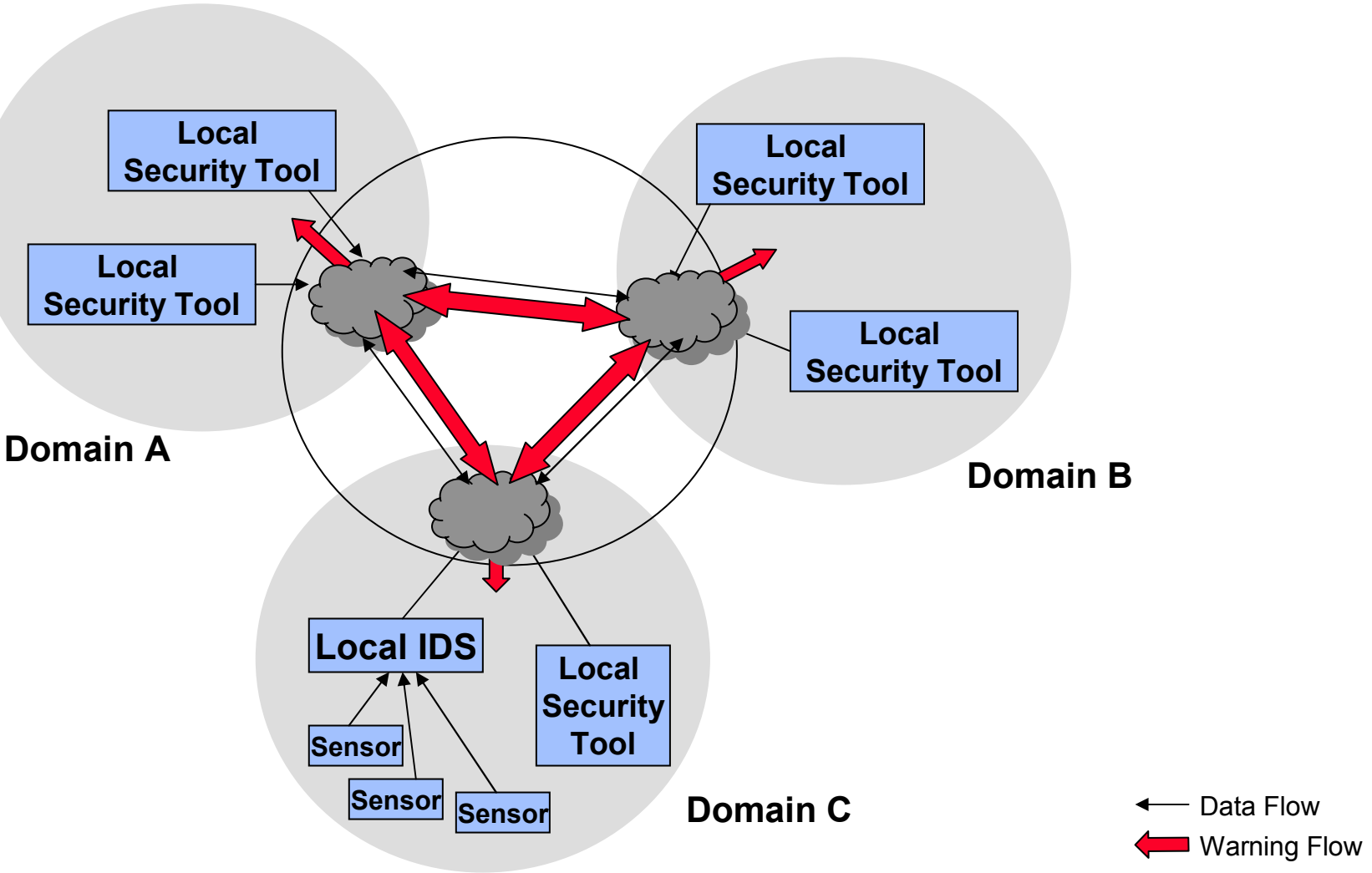
Dynamische Koalitionsumgebungen

- Veränderliche Menge kooperierender, gleichberechtigter Domänen
- Gemeinsame und konträre Ziele und Ansichten
- Beispiele für Koalitionsumgebungen
  - Allianzen von Wirtschaftsunternehmen
  - Kooperierende Strafverfolgungsbehörden
  - Militärische Netzwerke (NATO, SFOR, KFOR, ...)
- Beispiele für Kooperation in Koalitionsumgebungen
  - Logistik & konventionelle Infrastrukturen
  - Elektronische Infrastrukturen

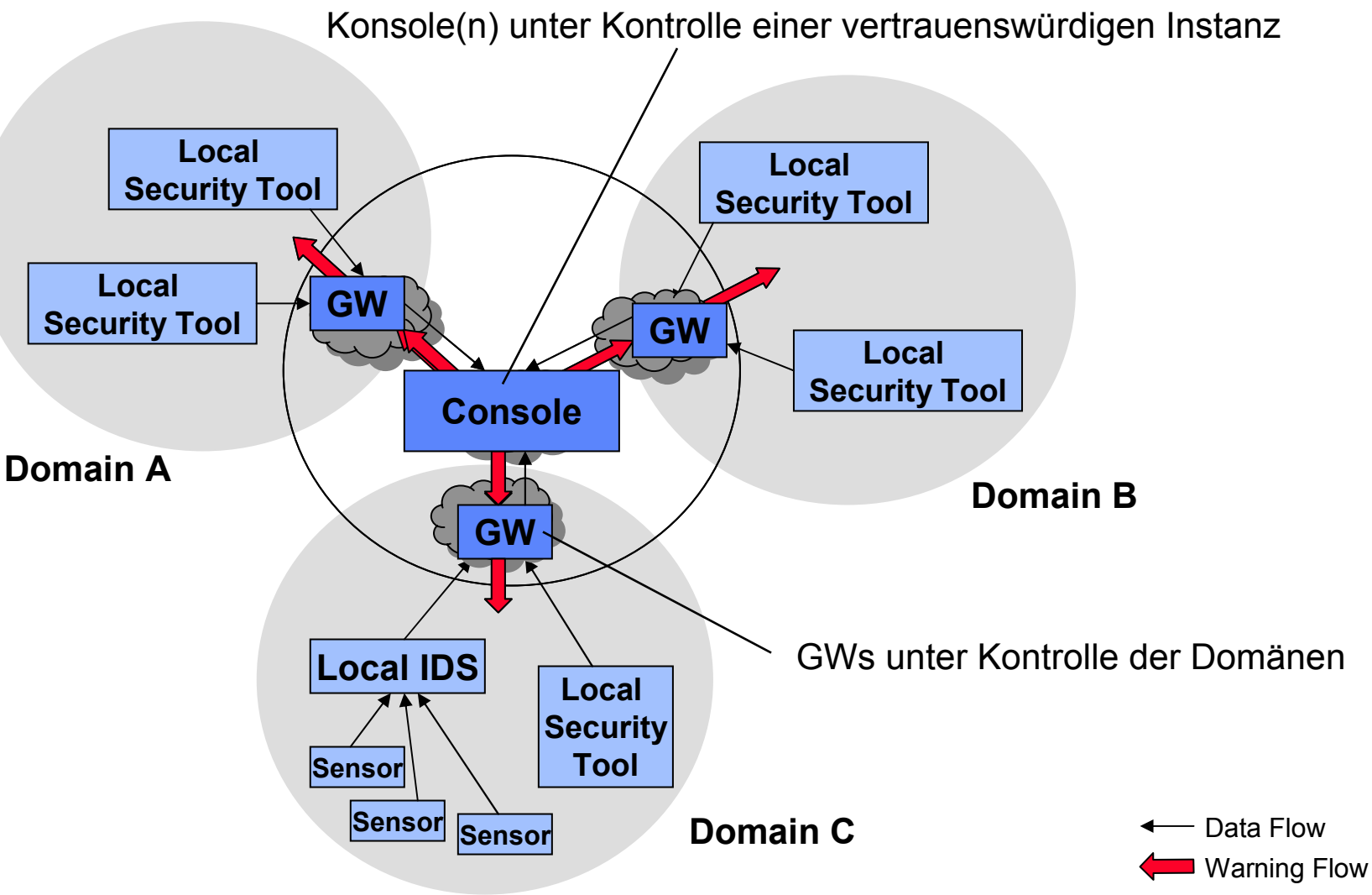
Kooperative IDS in Koalitionsumgebungen



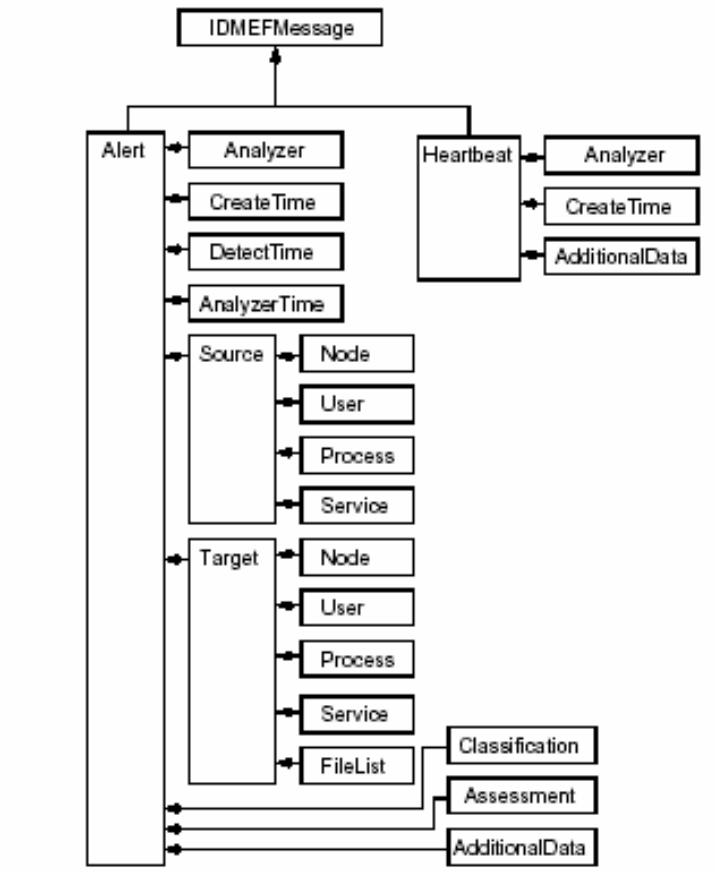
IWS-Architekturen (1): Rein verteilte Kooperation



IWS-Architekturen (2): Meta-IDS



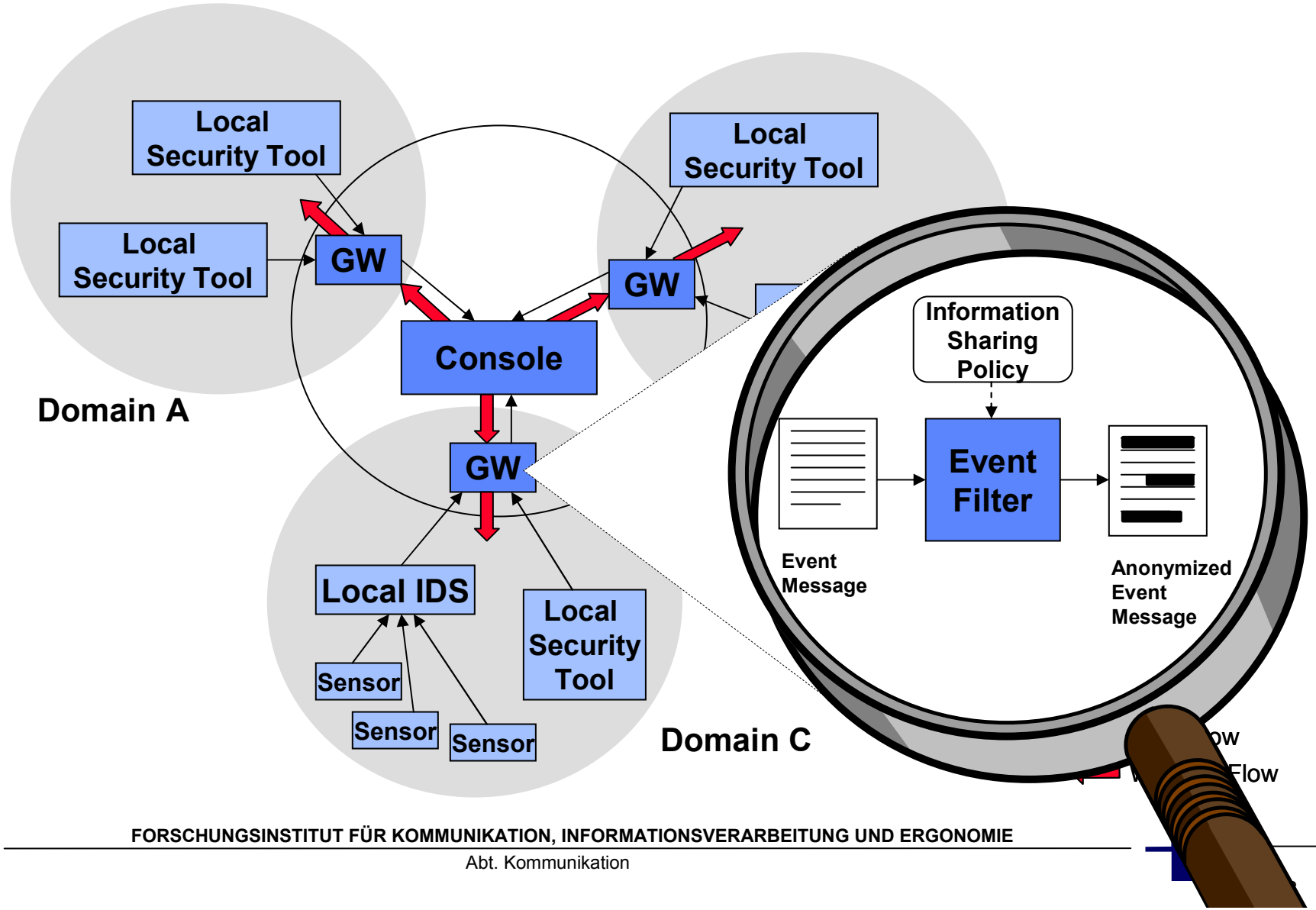
Interoperabilität: Datenmodell & -format



Intrusion Detection Message Exchange Format (IDMEF)

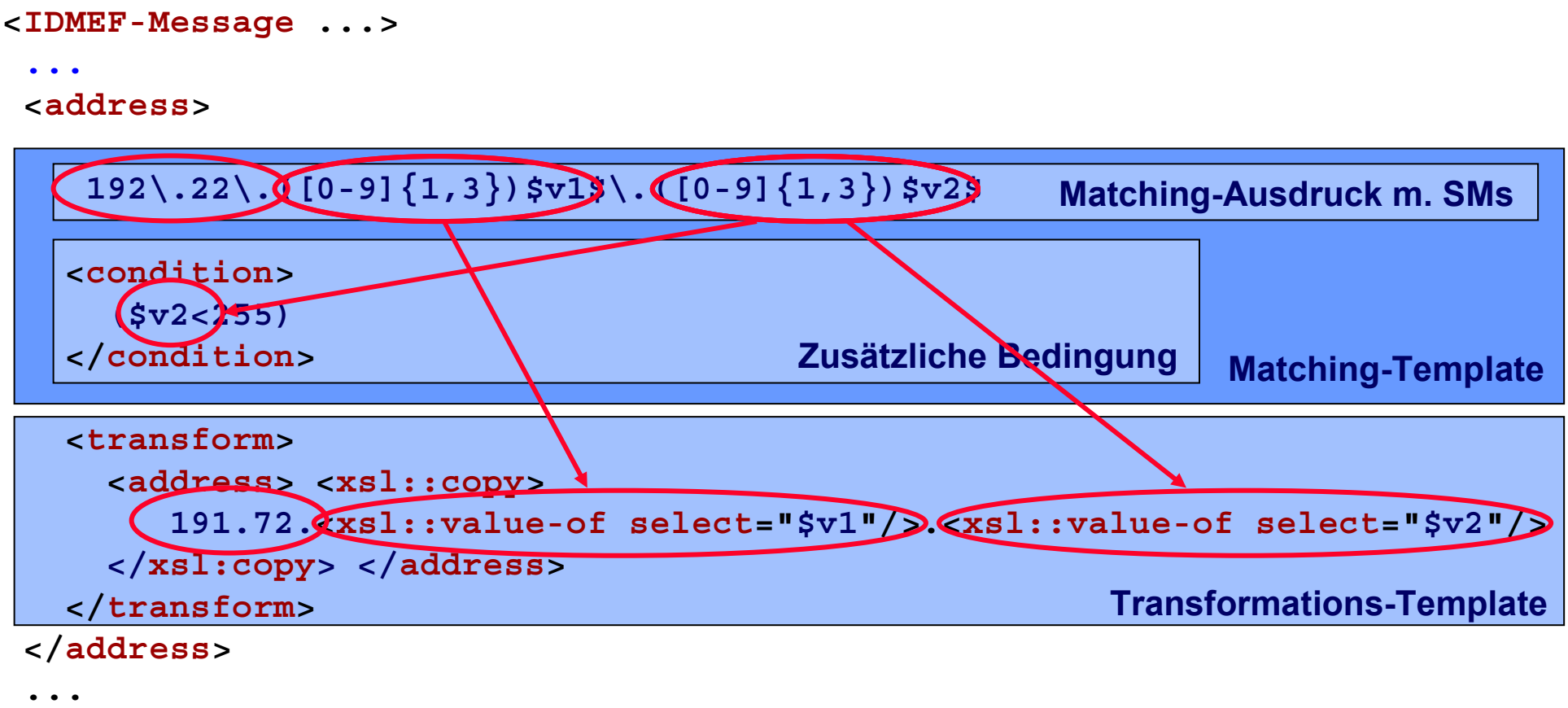
```
<IDMEF-Message version="1.0">
  <Alert ident="abc123456789">
    <Analyzer analyzerid="hq-dmz-analyzer01">
      <Node category="dns">
        <location>Headquarters DMZ Network</location>
        <name>analyzer01.example.com</name>
      </Node>
    </Analyzer>                                   Wer berichtet?
    <CreateTime ntpstamp="0xbc723b45.0xef449129">
      2000-03-09T10:01:25.93464-05:00
    </CreateTime>                                 Wann?
    <Source ident="a1b2c3d4">
      <Node ident="a1b2c3d4-001" category="dns">
        <name>badguy.example.net</name>
        <Address ident="a1b2c3d4-002" category="ipv4-net-mask">
          <address>192.0.2.50</address>
          <netmask>255.255.255.255</netmask>
        </Address>
      </Node>
    </Source>                                     Wer?
    <Target ident="d1c2b3a4">
      <Node ident="d1c2b3a4-001" category="dns">
        <Address category="ipv4-addr-hex">
          <address>0xde796f70</address>
        </Address>
      </Node>
    </Target>                                     Wo?
    <Classification origin="bugtraqid">
      <name>124</name>
      <url>http://www.securityfocus.com</url>
    </Classification>                             Was ist passiert?
  </Alert>
</IDMEF-Message>
```

Informationsbereinigung (1): Szenario

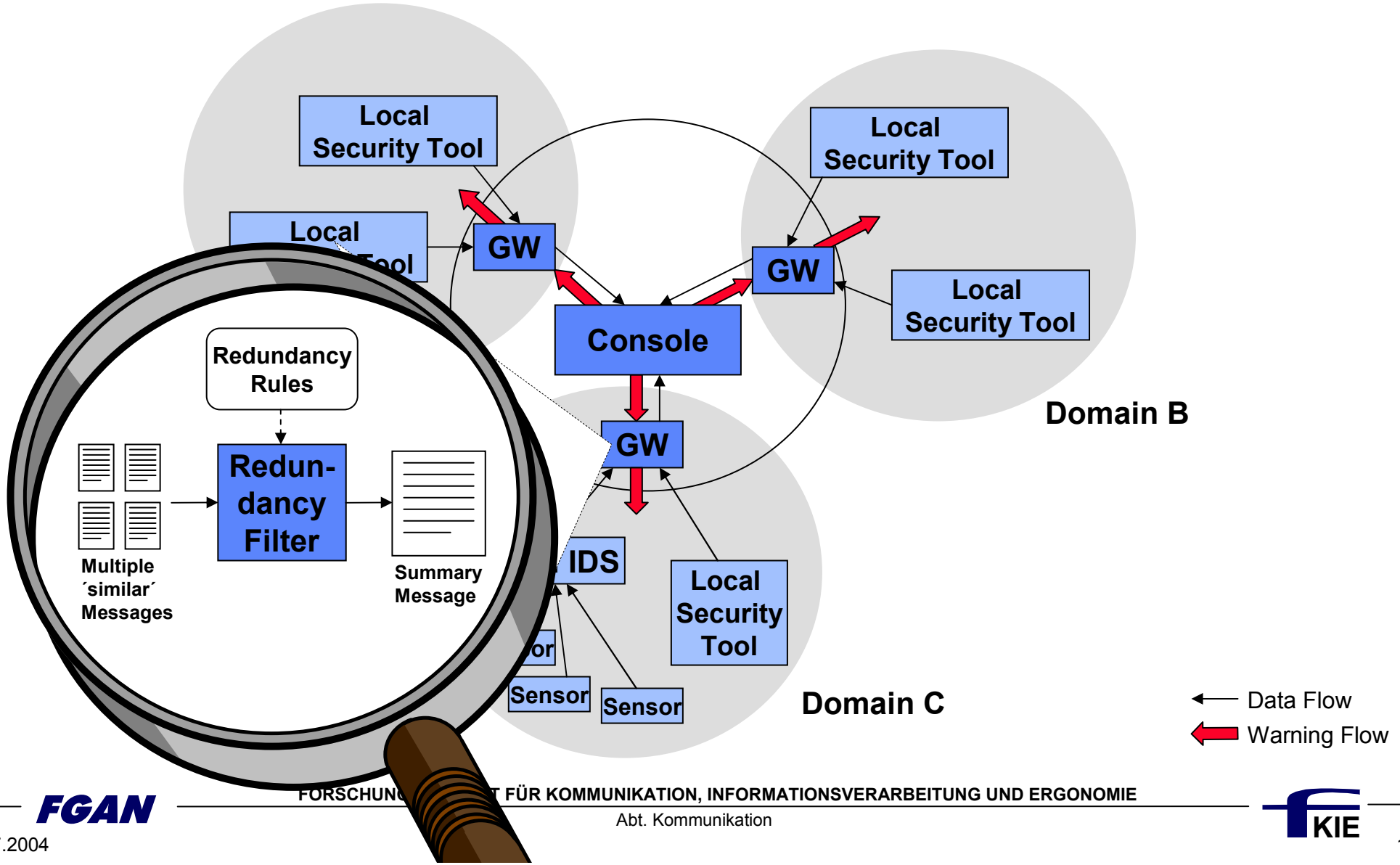


Informationsbereinigung (2): Implementierung

- Spezieller XSLT-Prozessor mit Erweiterungen für *Matching-sensitive* 1:1-Transformationen

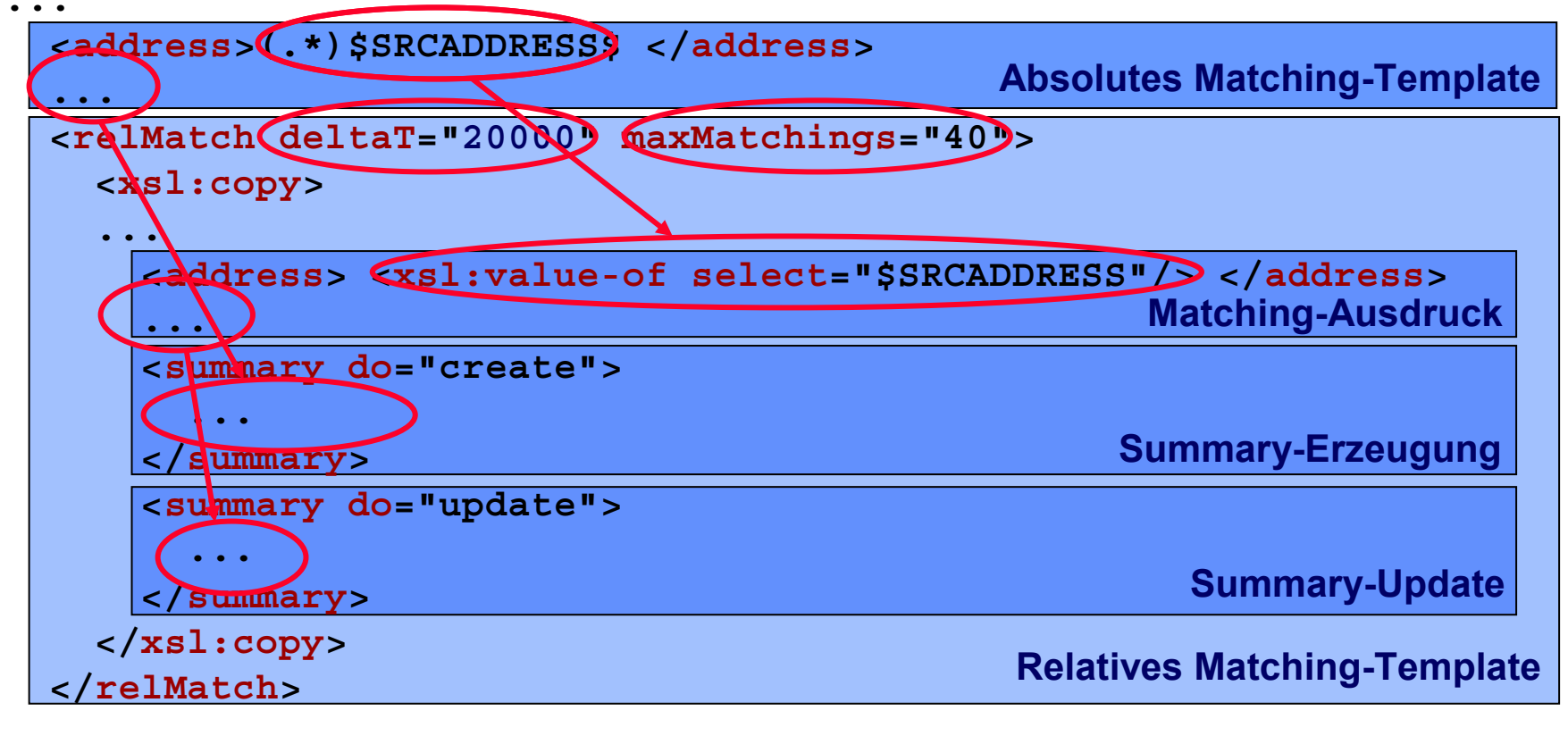


Redundanzfilterung (1): Szenario



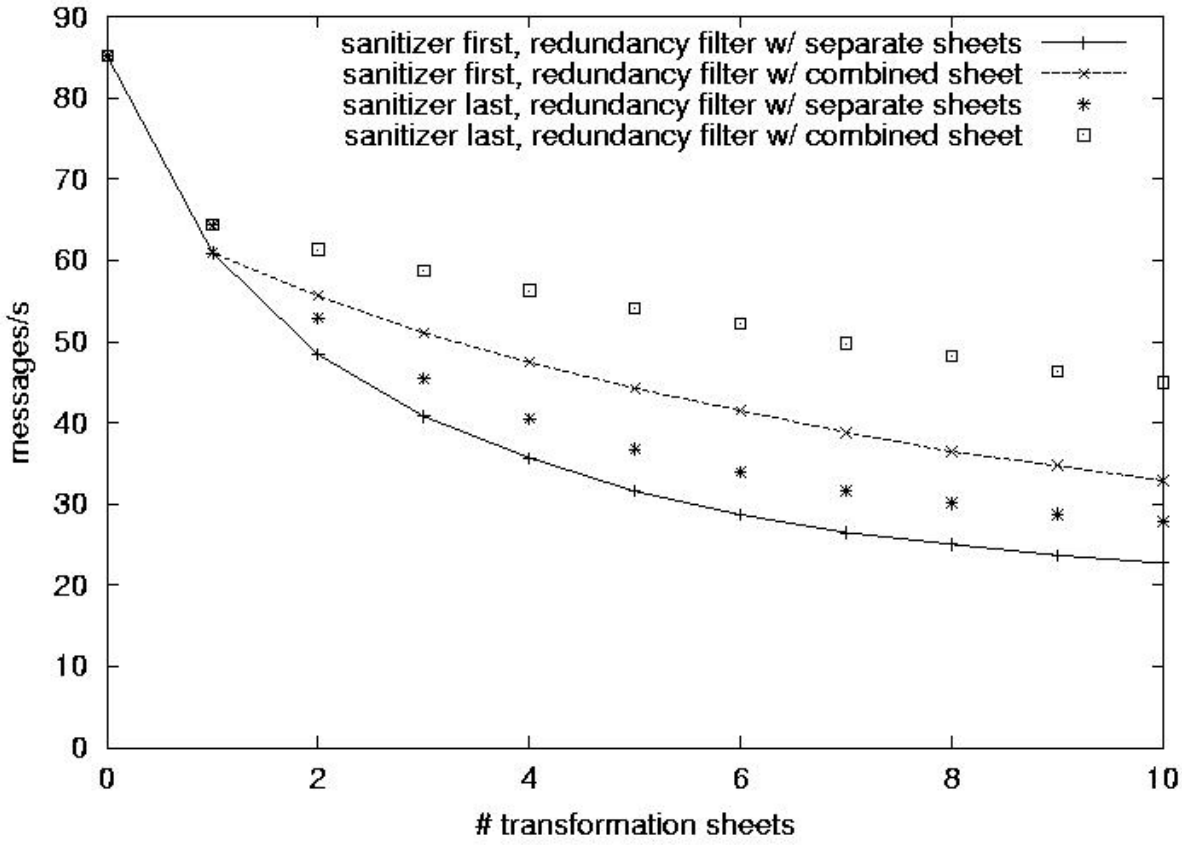
Redundanzfilterung (2): Implementierung

- Erweiterung des Informationsbereinigers um *relative Matchings* für Ähnlichkeitseigenschaft bei *n*:1 -Transformationen

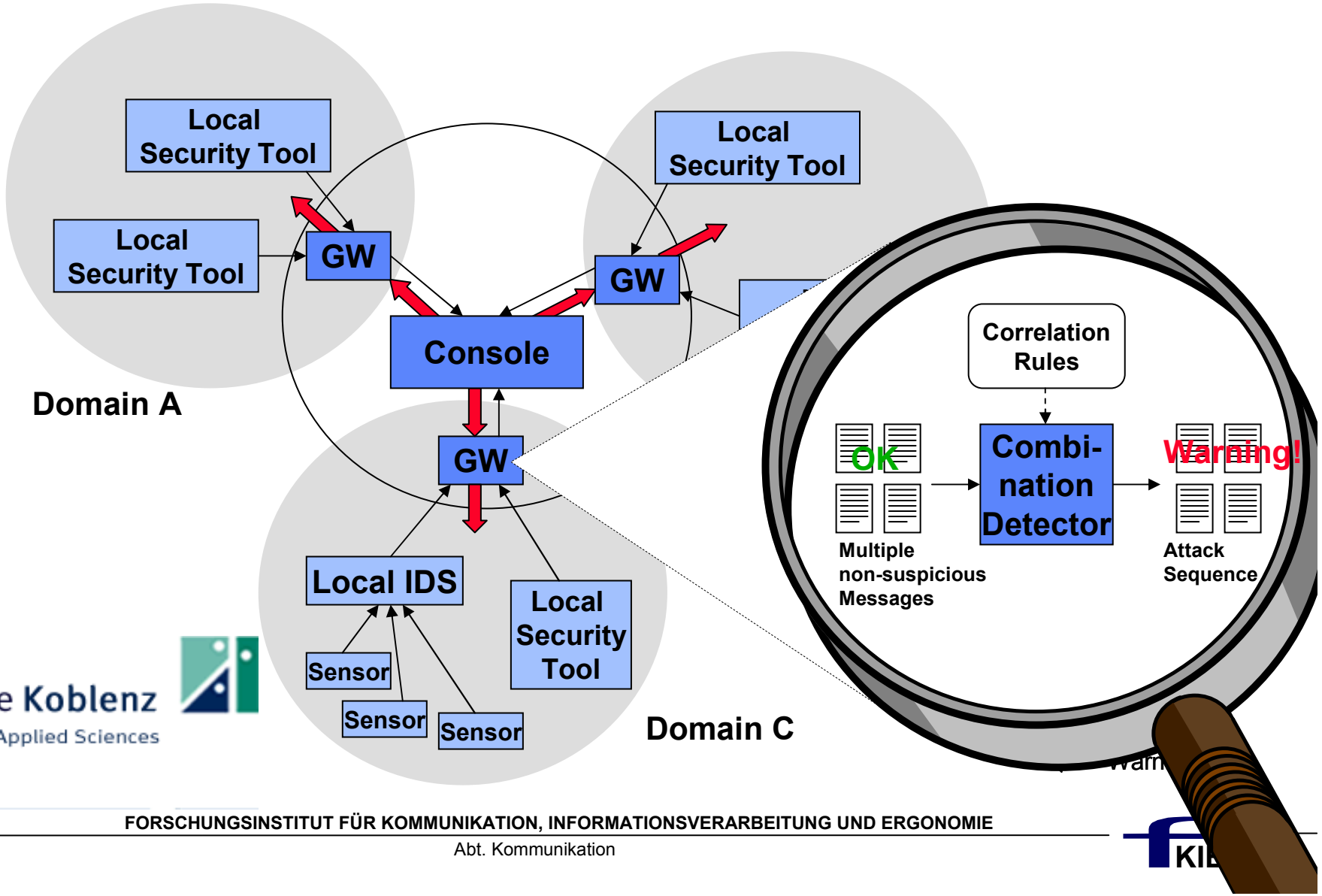


XSLT-Verarbeitung: Durchsatzmessungen

- Kombinierte Anwendung von Informationsbereiniger und Redundanzfilter im IWS-Gateway
- Hardware: PIII/1GHz/128MB
- $f(x) \approx 1 / (a + bx)$
- Anwendungs-reihenfolge ist entscheidend
- Separate vs. kombinierte Transformationen



Kombinationserkennung (1): Szenario

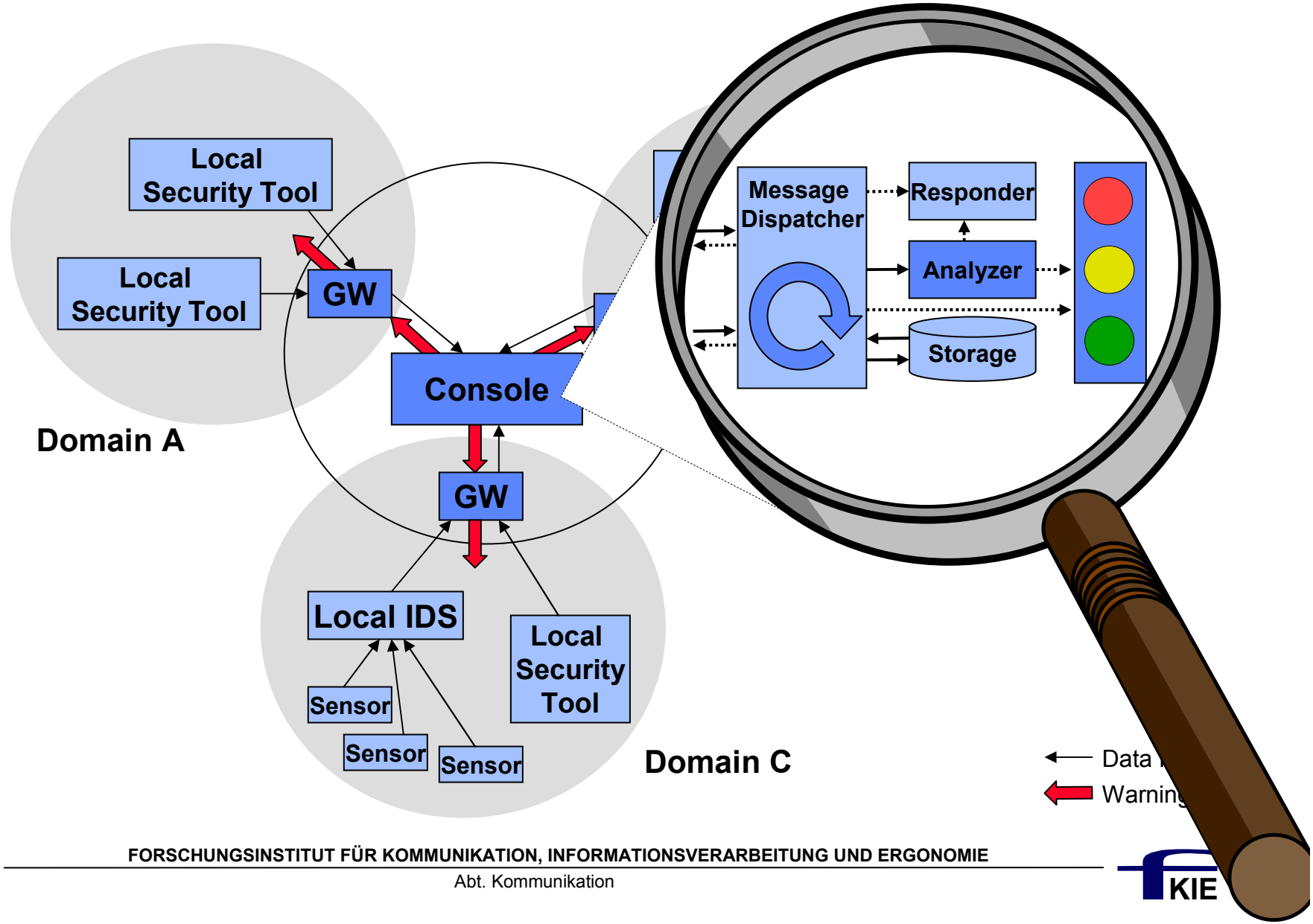


Fachhochschule Koblenz
University of Applied Sciences

Kombinationserkennung (2): Implementierung

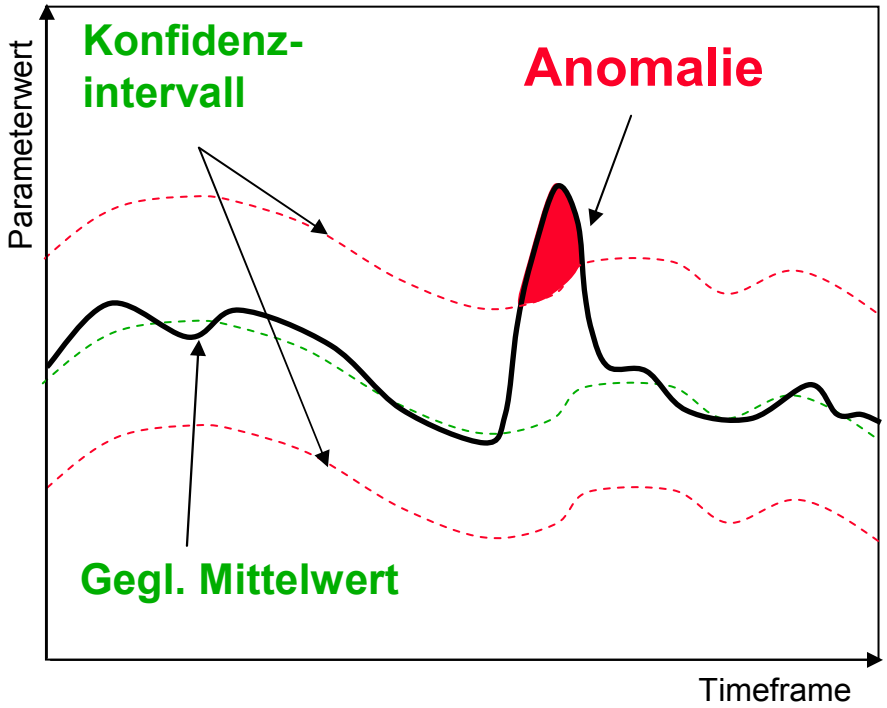
- *n*:(*n*+1) -Transformation in der Online-Variante
- Extremer Speicheraufwand für naive Implementation
- Erweiterung des Redundanzfilters um Just-In-Time-Erzeugung von Templates (*Matching-Stufen*)
- Verschachtelte Spezifikationen mit Transformationsvorschriften für die Kombinationsmeldung am Ende
- Integration und Evaluation dauern noch an

Anomalieerkennung (1): Szenario



Anomalieerkennung (2): Flussparameteranalyse

- Beispiele für Messparameter pro Δt:
  - # Meldungen
  - # Bytes / Meldung
  - # unterschiedlicher Klassifikationen (Signaturen)
  - # unterschiedlicher Quell-/Zieladressen
  - # unterschiedlicher Analyseknoten

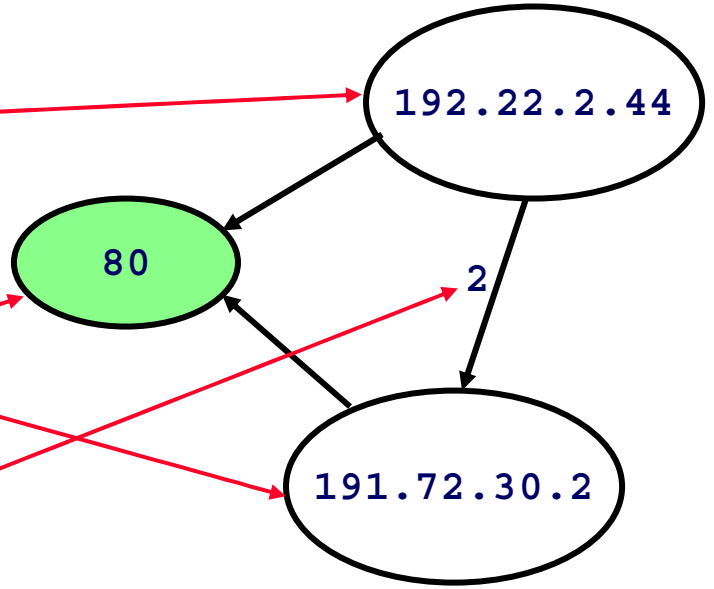


- Alarm, wenn Parameter Konfidenzintervall um Mittelwert verlässt
- **Nachteil**: Keine Betrachtung semantischer Information

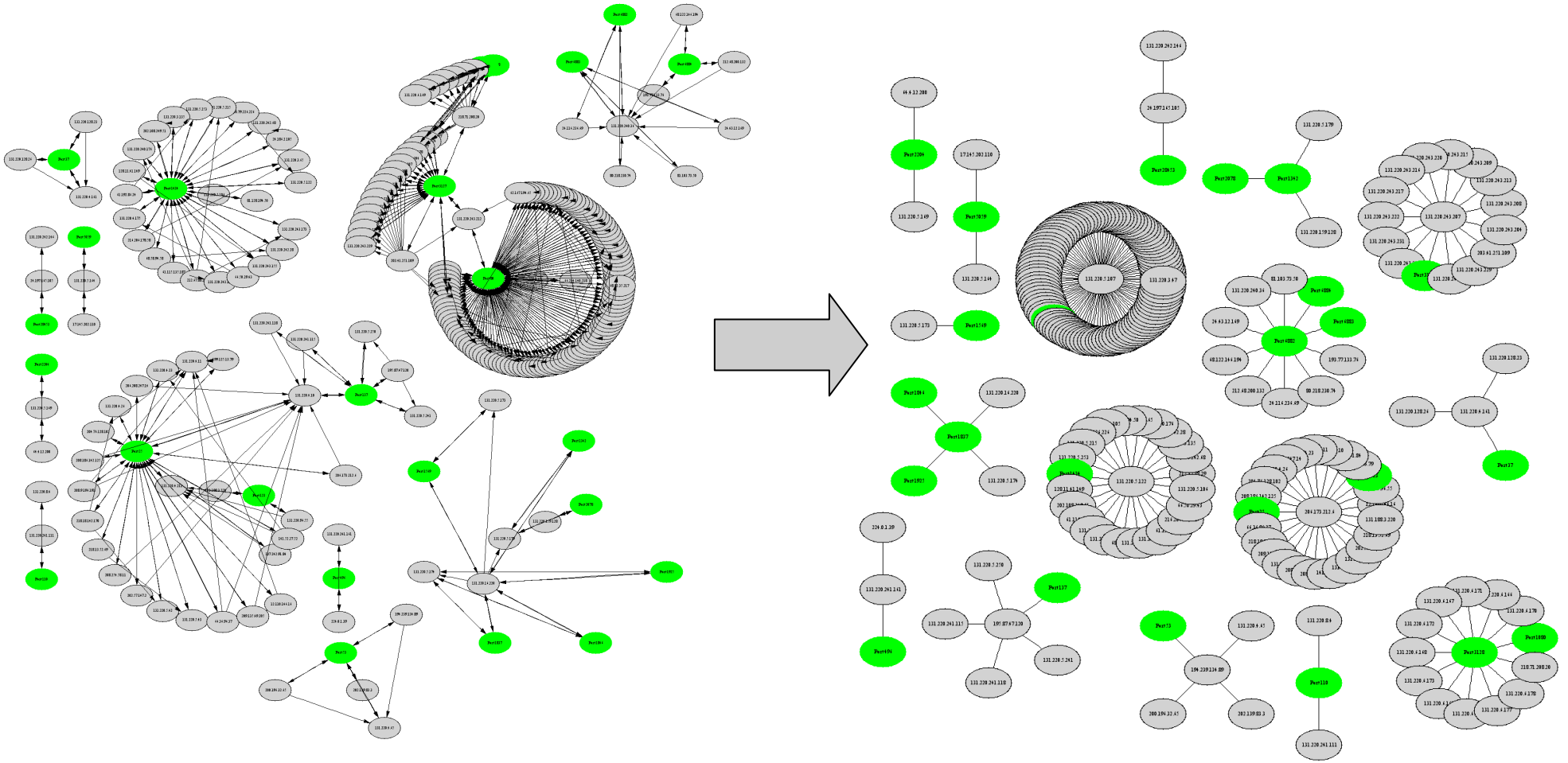
Anomalieerkennung (3): Quelle-Ziel-Graph

- Aufbau eines Quelle-Ziel-Graphen pro Timeframe

```
<IDMEF-Message ...
...
  <Source>
    <address> 192.22.2.44 </address>
     ...
  </Source>
  ...
  <Target>
    <address> 191.72.30.2 </address>
     ...
    <port> 80 </port>
  </Target>
  <severity> medium </severity>
  ...
</IDMEF-Message>
```



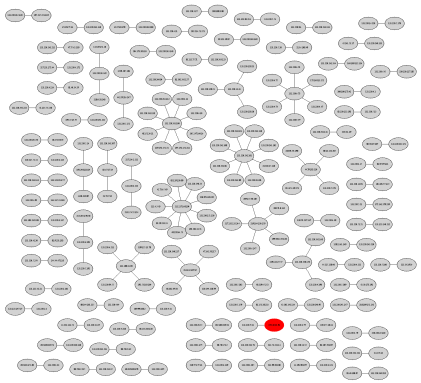
Anomalieerkennung (4): Graph-Clustering



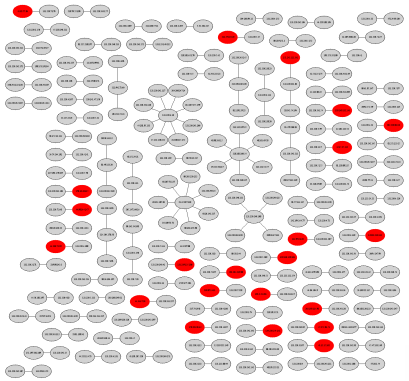
Quelle-Ziel-Graph mit Diensten/Ports

Cluster

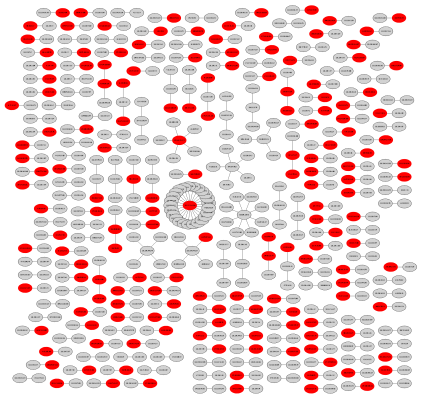
Anomalieerkennung (5): Clustering-Abweichung



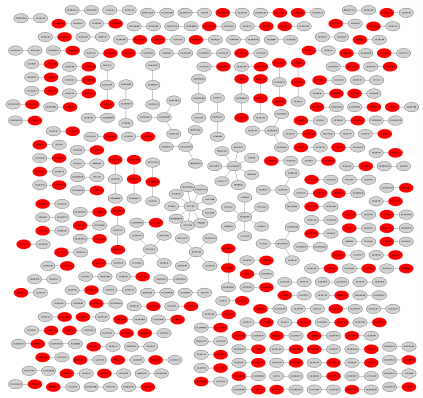
0:50



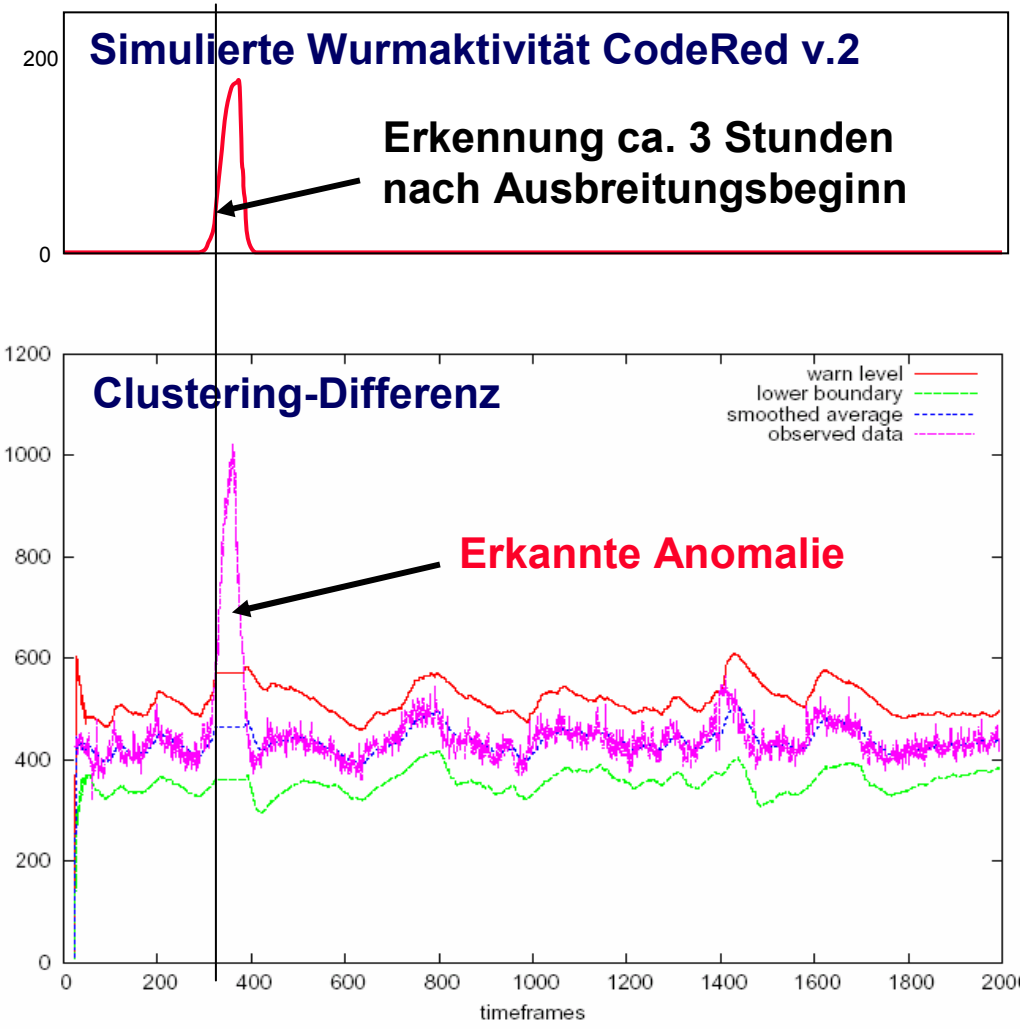
1:50



4:50



5:50



Zusammenfassung

- Meta-IDS als Architektur für IWS in Koalitionsumgebungen
- Vorverarbeitung von Ereignismeldungen durch XSLT
  - Informationsbereinigung
  - Redundanzfilterung
  - Online-Kombinationserkennung
- Anomalieerkennung im Ereignismeldungs-Datenmodell
  - Einfache Flussparameteranalyse
  - Clustering von Quelle-Ziel-Graphen
- XML/XSLT-basierte Ansätze leistungs- und ausbaufähig
- Anomalieerkennung erfolgreich in Simulation und Praxis

Ausblick

- Vervollständigung der Integration, Stabilisierung
- Multi-Domain-Demonstrator

- Extraktion von Informationen über Anomalien
- Auswirkungen der Informationsbereinigung auf die Anomalieerkennung
- Formalisierung & Simulationen
  - Grenzen der Verfahren
  - Skalierbarkeit

Kontakt

	Marko Jahnke	**Jens Tölle**
Post-anschrift:	FGAN/FKIE Abt. Kommunikation Neuenahrer Str. 20 D-53343 Wachtberg	Universität Bonn Institut für Informatik, Abt. IV Römerstr. 164 D-53117 Bonn
EMail:	jahnke@fgan.de	toelle@cs.bonn.edu

Kombinationserkennung (3): Implementierung

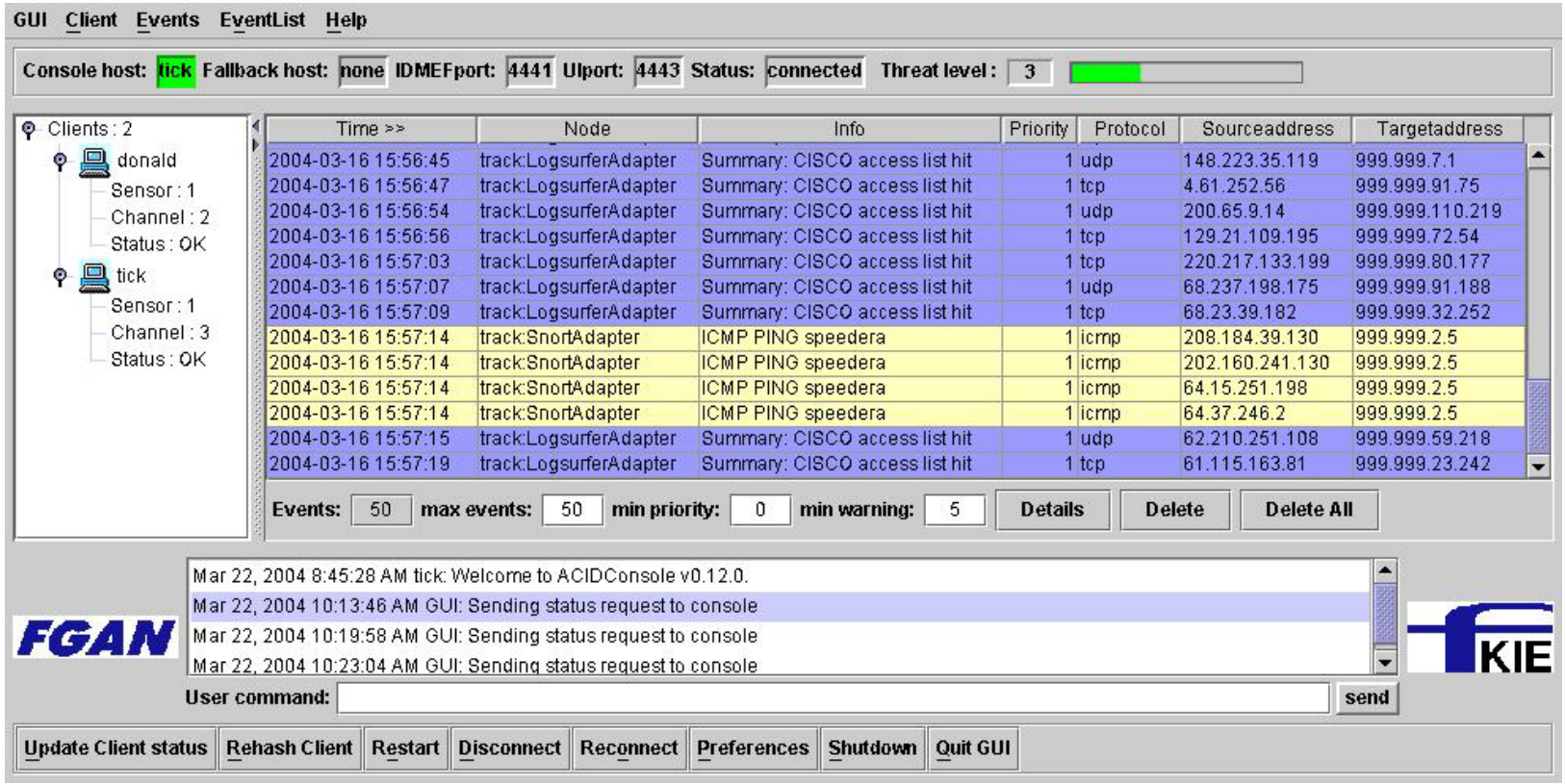
- *Just-In-Time*-Erzeugung von Templates durch Matching-Stufen

```
<IDMEF-Message ...>
  <corellation .../>
  ...                                          1. Initiales (absolutes) Matching-Template
  <relMatch deltaT="20000" threshold="40">
    ...
  </relMatch>                                  1. Relatives Matching-Template
  <transform finalMatching="false">
    <IDMEF-Message ...>
      <corellation .../>
      ...                                      2. Initiales Matching-Template
      <relMatch deltaT="20000" threshold="20">
        ...
      </relMatch>                              2. Relatives Matching-Template
      <transform finalMatching="true">
        ...                                    3. Matching-Stufe
      </transform>
    </IDMEF-Message>                           2. Matching-Stufe
  </transform>
</IDMEF-Message>                               1. Matching-Stufe
```

zurück

GUI Screenshots (1): IWS Console

- Console GUI visualization
  - Messages w/ anonymized addresses (prefix 999.999)
  - Summary messages from redundancy filter (blue rows)



GUI Screenshots (2): IWS Gateway

- Gateway GUI controls information sanitizing & filtering