

A Secure Event-Driven Framework for Service-Oriented Computing

Kees Leune
Tilburg University

Presentation Outline

- The PRONIR Project
- Service-oriented computing
- Security aspects
- The EFSOC Framework
- Summary and Conclusions

PRONIR

Profile-Based Retrieval of Networked Information Resources

- Joint project of Nijmegen University and Tilburg University
- Sponsored by NWO

Goal:

- Define a theory and implement prototype
- EFSOC: provides infrastructure
- VIMES: provide retrieval capabilities



Service-Oriented Computing

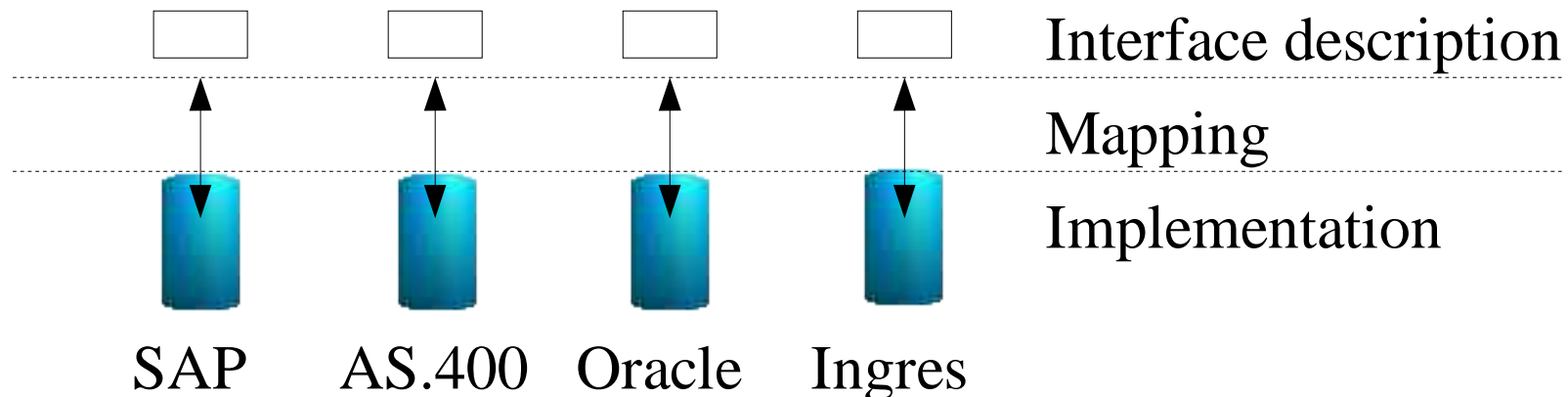
Service-oriented computing is a new paradigm for designing, building, testing and deploying loosely coupled, distributed software components.

- Loosely coupled
- Distributed software components

Distributed Software Components

Distributed software components require a separation between:

- Technology-neutral interface descriptions
- Technical mapping of interface to implementation infrastructure
- Availability of actual implementation

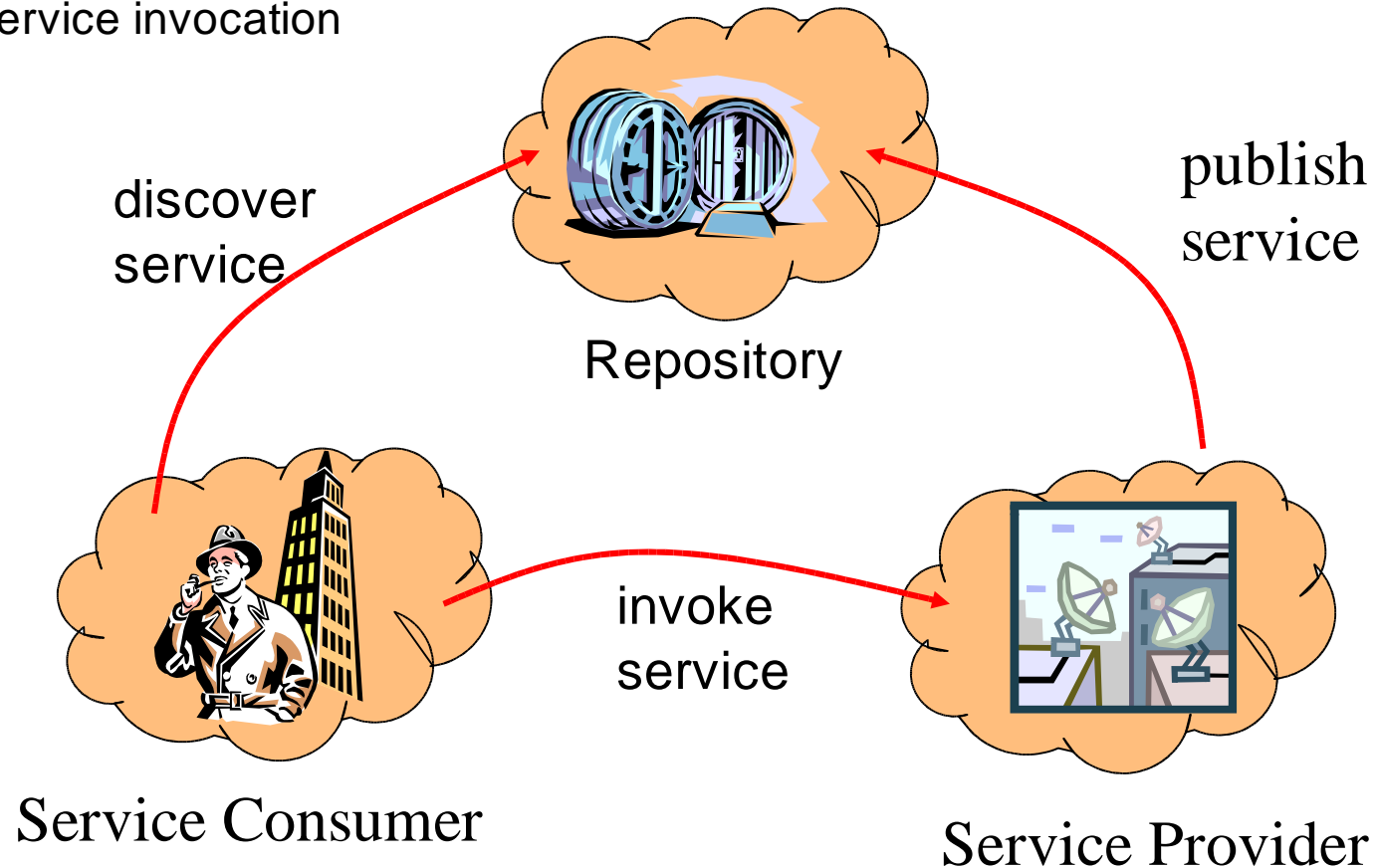


Loose couplings

- Rapidly changing (business) environments require the ability to quickly respond to new requirements and desires.
 - Shorter product life cycles
 - Need for more flexible integration of information systems
- Service Oriented Computing is based on two main concepts:
 - Separation between interface description and technology.
 - Ad-hoc selection and invocation of services based on functional requirements.

Web Services

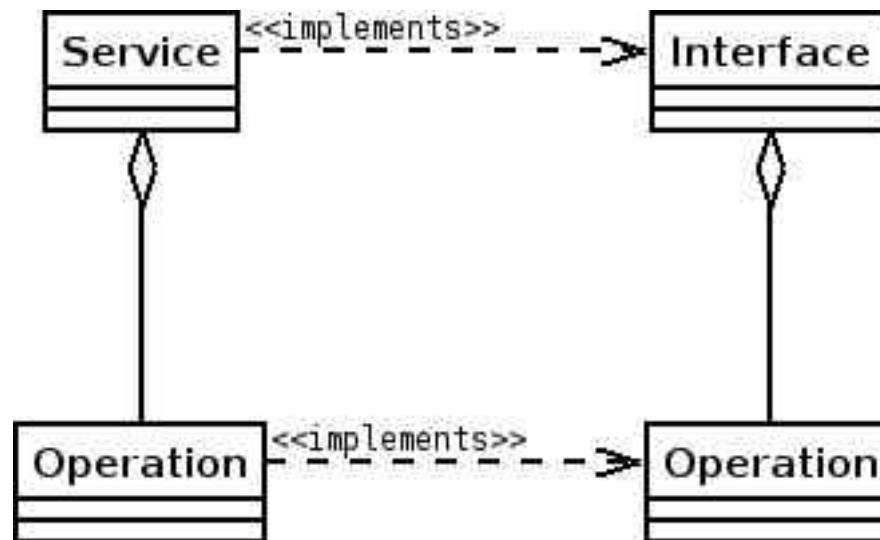
- Service description via XML documents
- Service discovery via UDDI repositories
- Ad-hoc service invocation



Web Services

Web service descriptions contain sections for

- Messages
- Interfaces
- Operations
- Bindings
- Services



Standards, Platforms and Vendors

- Service-Oriented Computing is a hot topic. Many rapidly changing standards and vendors have entered the arena
- Standards
 - WSDL: Service Description
 - SOAP: Invocation
 - WS-Policy: Policy Definition
 - WS-Security: Security Aspects
 - WS-Eventing: Event Support

Standards, Platforms and Vendors (2)

- Platforms
 - Microsoft .NET
 - IBM Websphere
- Vendors and institutes
 - IBM
 - Microsoft
 - BEA Systems
 - OASIS
 - etc.

Problems

- Too many standards to choose from.
- Standards are changing too fast.
- All vendors sell services, but nobody has an answer to what they really are.
- Very technology-oriented.
- Lots of talking, little doing.

Security Considerations

- 1) Service-Oriented Computing assumes rapidly changing processes using service providers selected on functional requirements.
 - 2) Services interact with business processes to deliver a product.
 - 3) Service providers change rapidly and may not be known in advance.
- Trust is a major issue!

Security aspects

- Authentication
 - Is someone who he claims to be?
- Authorization
 - Is person X allowed to do what he wants to do?
- Confidentiality
 - Can we be sure that only the recipient of a message understands it?

Security aspects (2)

- Integrity
 - Can we be sure that messages have not been changed between sender and receiver?
- Non-repudiation
 - Can receiver be sure that sender sent it, and vice versa?

EFSOC Framework

Basic assumptions:

- Loose coupling through events
- Security major consideration
- Business Processes/Services
- Role-Based access

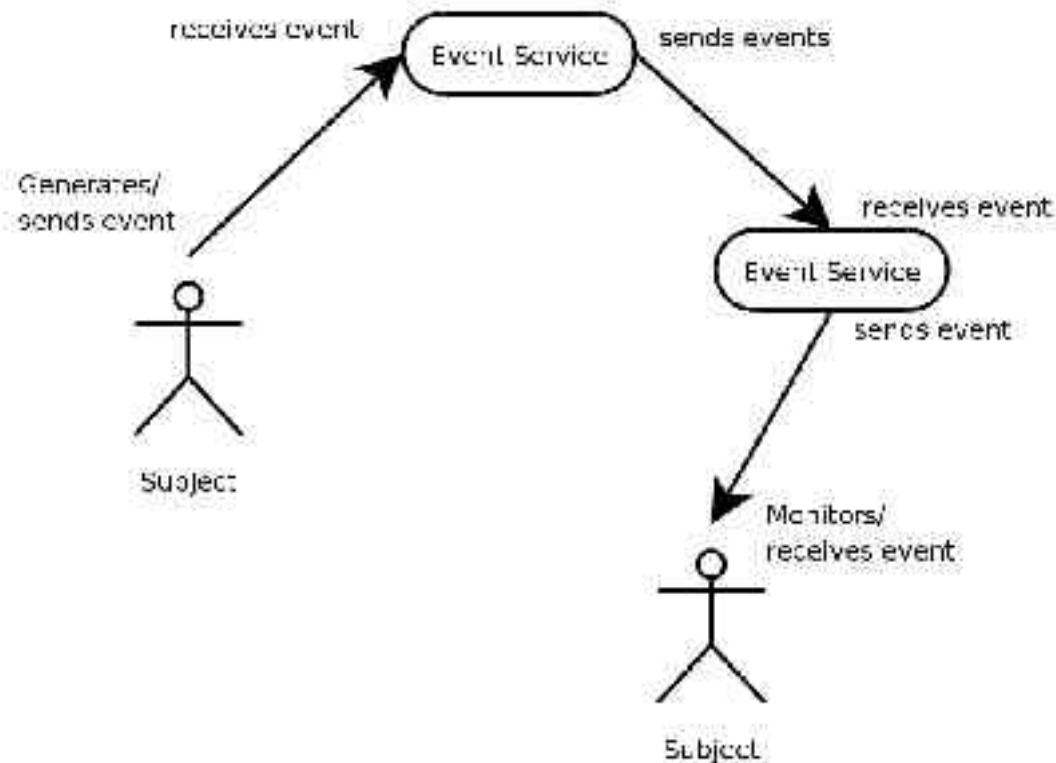
EFSOC overview

The EFSOC Framework is a secure framework for event-driven service-oriented computing, which may be divided in three tiers:



Event Tier

- Events represent “things” happening in organizations.
- Events and are sent and received by subjects.



Events

- Events have an envelope and a body
 - The event header contains meta-information used for security and routing
 - The event body contains additional information.
- All event body types must be defined before they are sent.
- A subject needs to subscribe to an event type before he receives it.

Event example

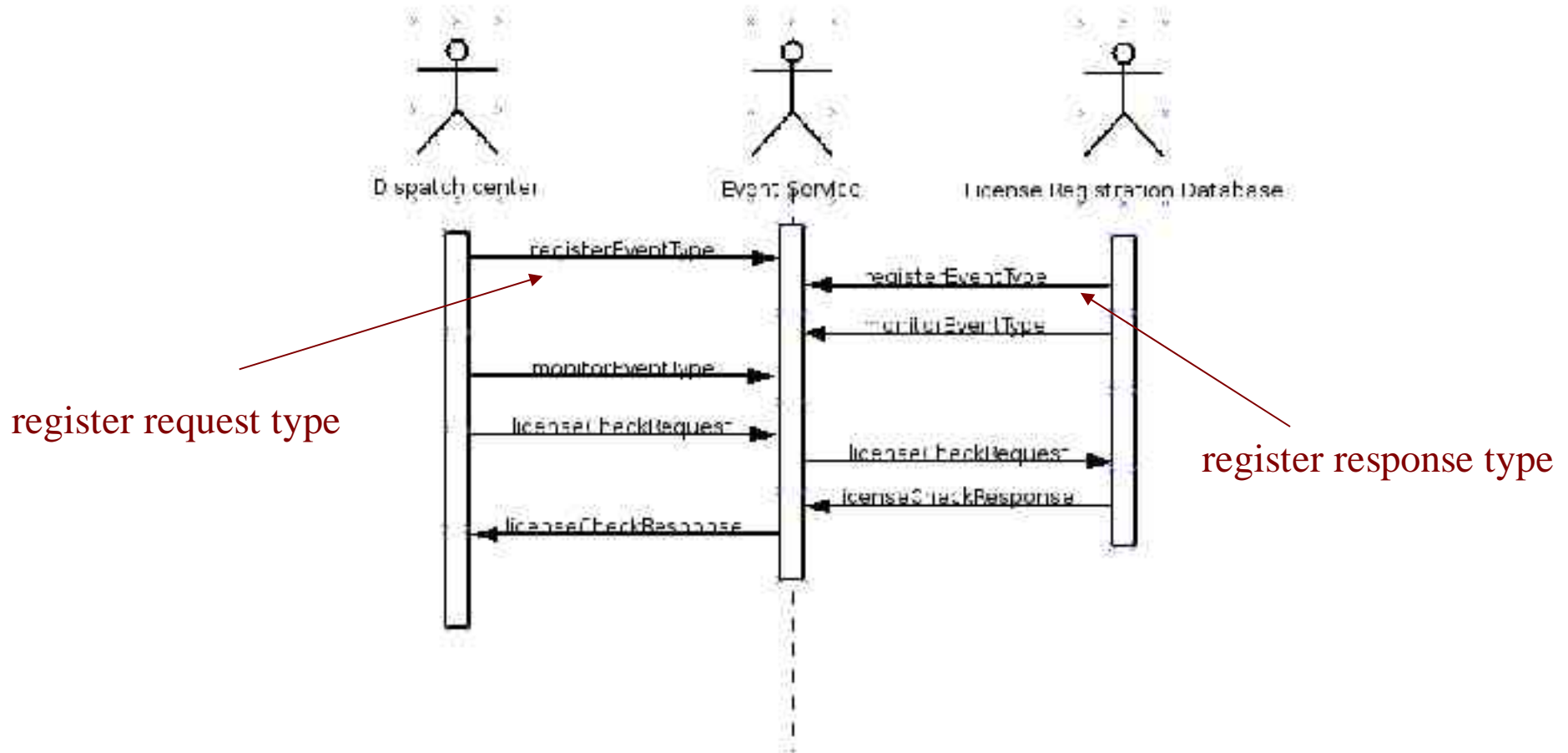
Define event body: license check request

- license plate number (required)
- brand of vehicle (optional)
- type of vehicle (optional)
- color of vehicle (optional)

Example:

- An event to request the registration of the licenseplate GL-DJ-16, a red Nissan Primera 2.0

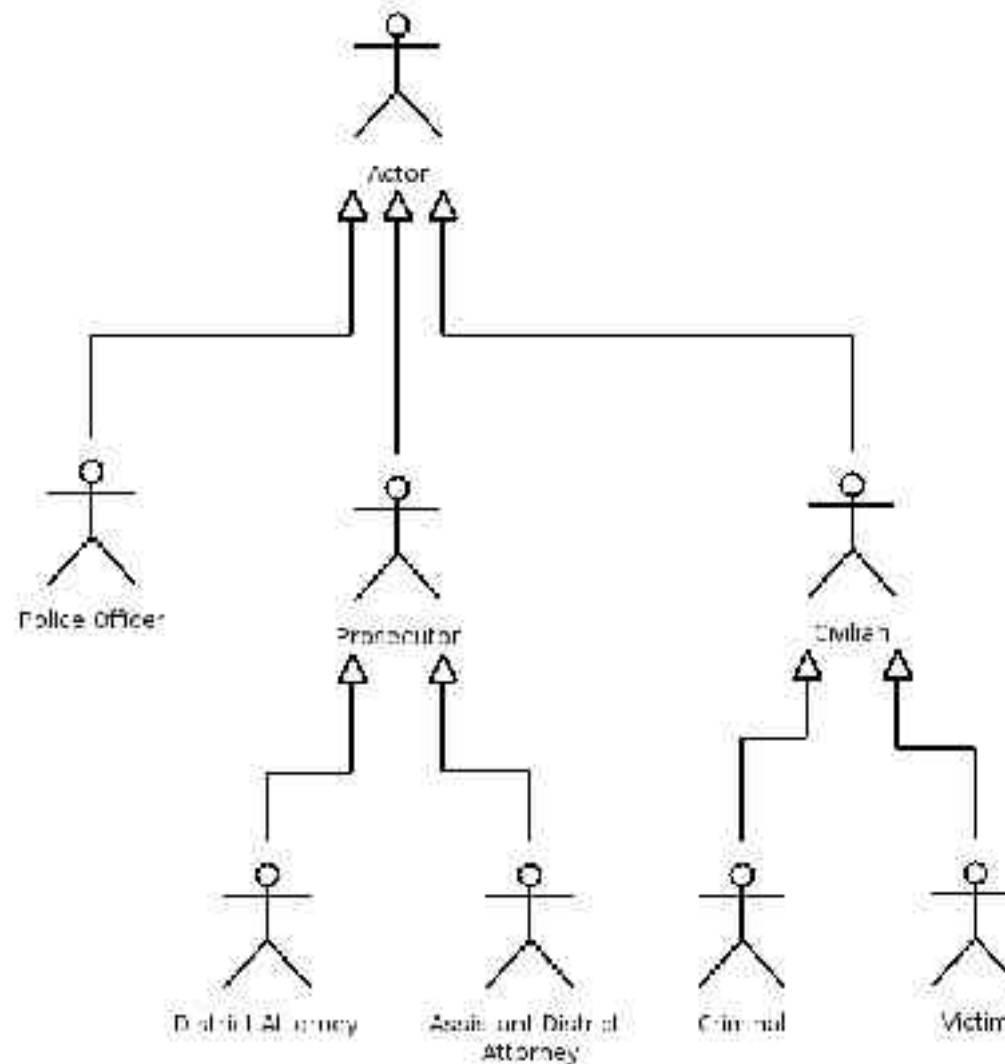
Event example (2)



Security Tier

- The EFSOC security model is based on Role-Based Access Control, i.e.
 - Subjects play roles
 - Roles are granted permissions
- Authentication:
 - unauthenticated: subject is unknown
 - partially authenticated: subject is known, but plays no roles
 - fully authenticated: subject is known and plays at least one role.

Role Example



Authorization

Associated with each Event Type is an Authorization Policy containing one or more rules.

– licenseCheckRequest:

- R1. All police officers may request license checks
- R2. Only dispatchers may receive license checks

– wireTapRequest:

- R1. Only District Attorneys may receive wireTapRequests
- R2. Only Police Officers may send wireTapeRequests

A typical authorization rule

- kees sends license check request

```
valid_sender(Subject, licenseCheckRequest) :-  
    is_fully_authenticated(Subject),  
    subject_has_role(Subject, Role),  
    role_may_generate(Role, licenseCheckRequest).
```

- dispatcher receives license check request

```
valid_recipient(Subject, licenseCheckRequest) :-  
    is_fully_authenticated(Subject),  
    subject_has_role(Subject, Role),  
    role_may_monitor(Role, licenseCheckRequest).
```

Active Security

- States of an request/response message
 - Request unsend
 - Request sent, but not yet received
 - Request sent and received
 - Response sent, but not yet received
 - Response sent and received
- Active Security requires that if authorizations change in any state, the

Confidentiality and Integrity

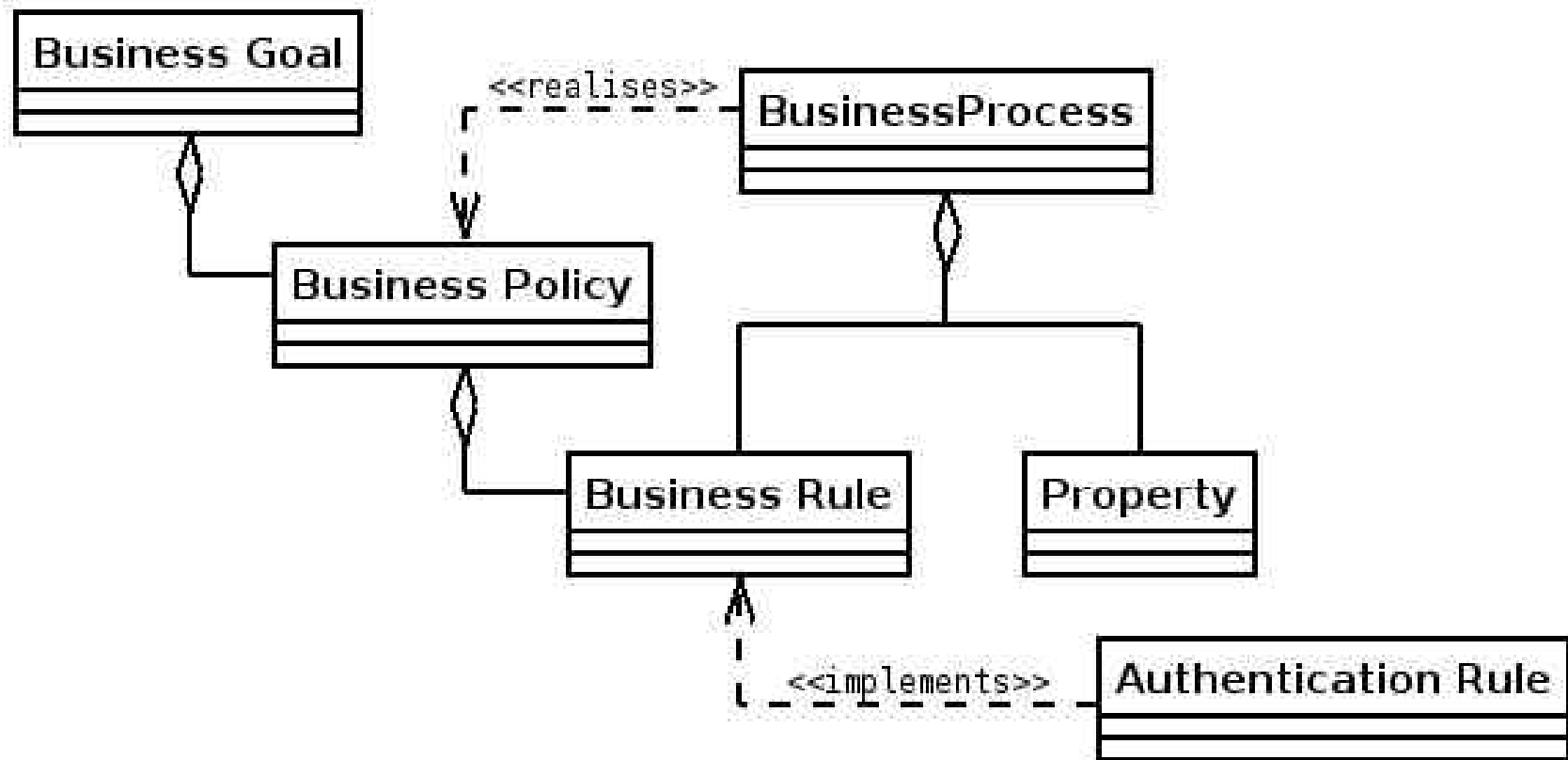
- All requests must be digitally signed and encrypted using adequate cyphers.
- Example:
 - Dispatcher generates licenseRequest, signs it with his key, and encrypts it for the Event Service
 - Event Service decrypts event, checks signature and determines it must be forwarded to registration database.
 - Event Service signs request, encrypts is for the registration database and delivers it.
 - Registration database decrypts and checks signature.

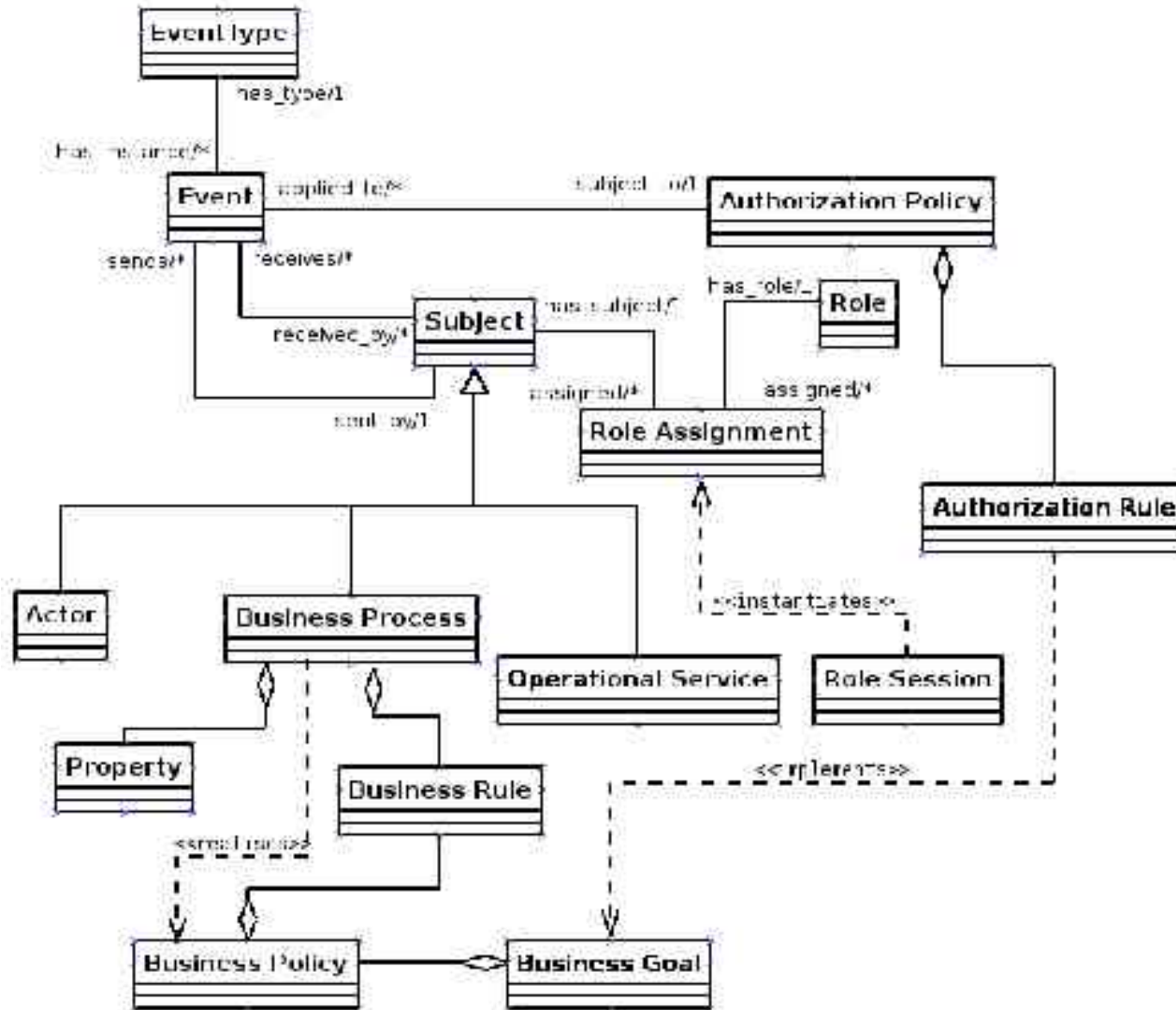
Business Process Tier

Manages business protocols and service specifications.

- (Operational) service descriptions are WSDL documents.
- A business process is a context in which a set of business rules are defined executed.
- Services are *external*, whereas business process are *internal*.

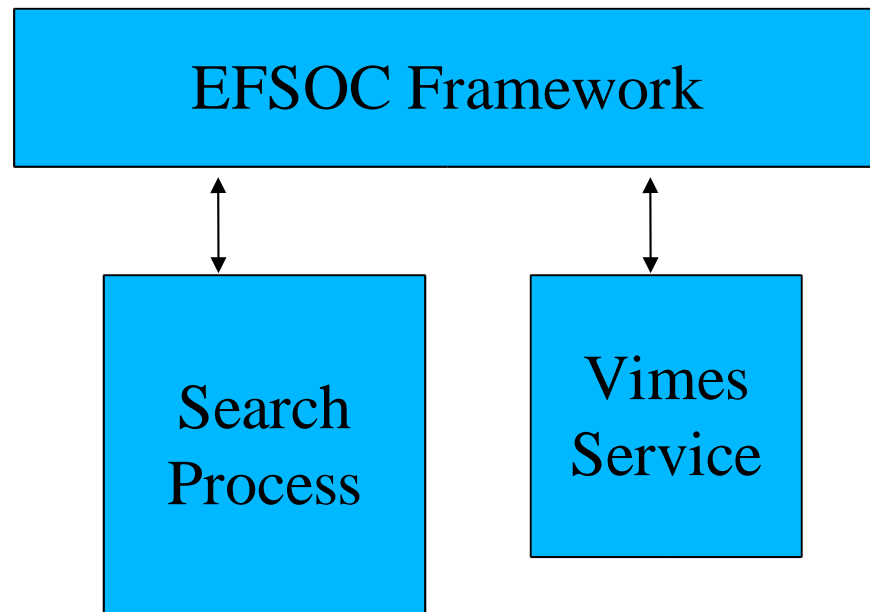
Business Processes





Using EFSOC for PRONIR

- PRONIR = EFSOC + VIMES
- EFSOC provides an INFRASTRUCTURE for VIMES



Questions ?

Contact me

email : kees@uvt.nl
phone : +31 13 466 2688
web : <http://www.leune.org>
address : Tilburg University
Infolab, Room B738
P.O. Box 90153
5000 LE Tilburg
NETHERLANDS