

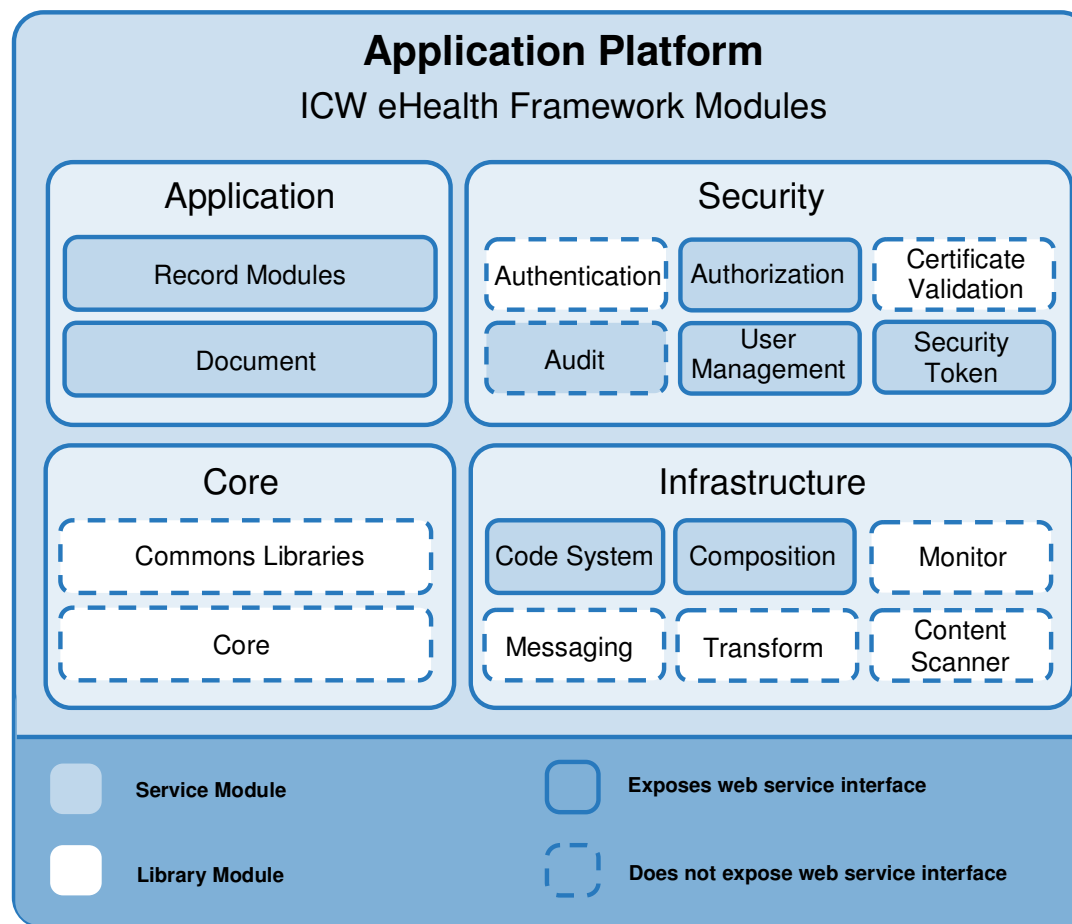
eHF – Application Platform

<Audience>

<Presenter> / <Date>

Agenda

- eHF Modules
 - Core
 - Security
 - Infrastructure
 - Application
- Examples
 - Add Observation
 - LifeSensor
- Outlook



Agenda

- eHF Modules

- Core

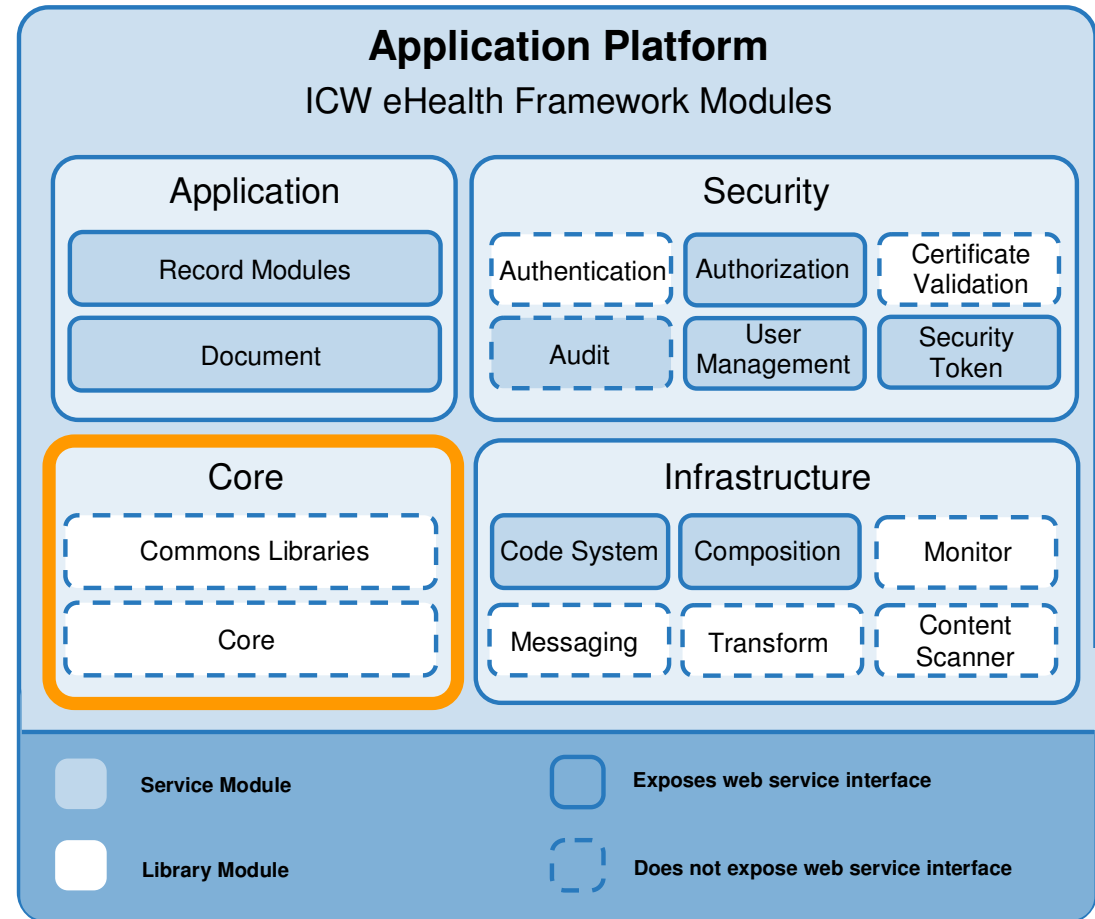
- Security

- Infrastructure

- Application

- Examples

- Add Observation



eHF Core Commons Library Module

- Is the fundament of the eHF – the central library module
- Includes definitions of general components and interfaces that are used, or enhanced, by the dependent modules
- Base classes included in the Commons module
(Exception class hierarchy, Base classes for domain objects and transfer objects, ...)
- Defines infrastructure for individual layers of the eHF architecture
(Persistence infrastructure, Generic services, Web services,...)
- Specializations have their own dedicated modules
(eHF Commons Code System, eHF Commons Expert Entry, eHF Commons Audit)

eHF Core

Commons Library Module - Components

Consist of a lot of individual parts:

- *Annotations*
- *AOP (Aspect Oriented Programming)*
- *Audit*
- *Axis Web Services*
- *CGLIB Support (Code Generation Library)*
- *Contract Relevance*
- *DAO Support (Database Access Object)*
- *Data Import*
- *Date and Time*
- *Domain Object*
- *DTO Support (Data Transfer Object)*
- *Exception Handling*
- *Event*
- *Hibernate*
- *Metadata*
- *Model Support*
- *Object Assembler*
- *Object Manager*
- *OCL (Object Constraint Language)*
- *Paging*
- *Principal*
- *Reflection*
- *Reporting*
- *Service Layer Support*
- *Servlet Filter*
- *Spring*
- *UUID*
- *Web Service Layer Support*

eHF Core

Commons Library Module - Date and Time

- Provides utilities for handling and converting date and time objects
- Java does not support partial dates (e.g. „March 2005“)
- ISO 8601 is a specification for date and time formats supporting partial dates
- Supports following formats
 - ISO 8601 dates
 - UTC (Coordinated Universal Time)
 - Canonical date & time
- Uses Joda library (Java implementation of ISO 8601 dates)

eHF Core

Commons Library Module - Metadata & OCL

- Metadata + *Validation*
 - Handling and validating domain object utilizing the object metadata
 - The metadata describes the properties of domain objects
- OCL (**O**bject **C**onstraint **L**anguage)
 - Provides interfaces and base classes for an EMF (Eclipse Modeling Framework) based framework to perform runtime OCL checks on the domain model
 - During an eHF module build, the eHF Generator creates EMF model classes and EMF framework classes
 - These classes base on the interfaces and classes of this OCL commons component and implement the constraints specified in the eHF Generator's source model

eHF Core

Commons Library Module - Persistence Infrastructure

- Generic Hibernate DAO implementation
- Object Manager
 - Cascades on object structure (e.g. saving dependent objects)
 - Validates objects before save or update
 - Fires events on object level (e.g. update event), replaces Hibernate interceptor and adds additional context information, currently mainly used for auditing
- Customizations for Hibernate framework
- Utility classes for dealing with Hibernate

eHF Core

Core Library Module – Introduction (1/2)

- Defines domain objects shared among several modules
- Prevents duplication of models and associated logic
- Enriches domain model with domain logic to support the usage of the central model elements
- Supports currently the following data categories
 - Personal Information (Person, Address, ...),
 - Terms (Date, Time, Time Interval, ...)
 - Codes (Code System, Code, ...)
 - Identifiers and Qualifiers (Reference, InstanceIdentifier...)

eHF Core

Core Library Module – Introduction (2/2)

- Considers each class a value object
- Does not provide a persistence mechanism of its own. So the actual persistence of the data is performed by the modules using the respective domain object of the Core Library module
- Plays a central role in the eHealth Framework as it provides important elementary classes on which a lot of other modules depend
- No web services provided. Besides the generated representation of the domain objects, no additional interfaces are defined

Agenda

- eHF Modules

- Core

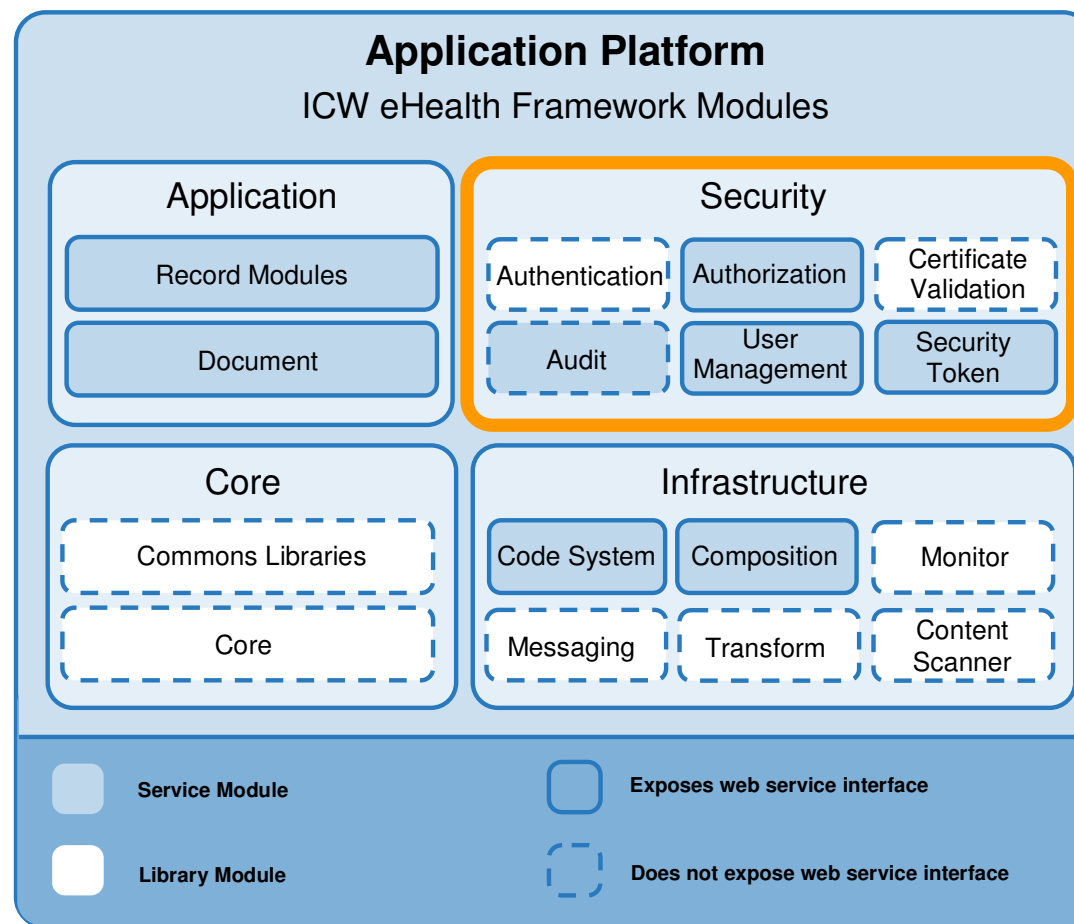
- Security

- Infrastructure

- Application

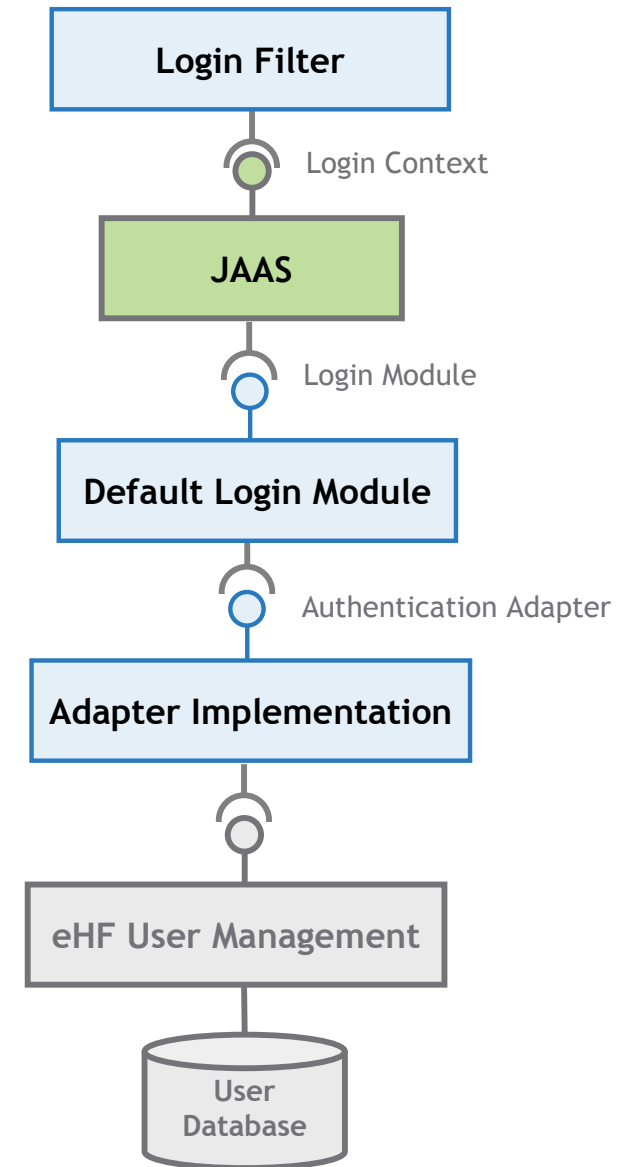
- Examples

- Add Observation



eHF Security Authentication Module

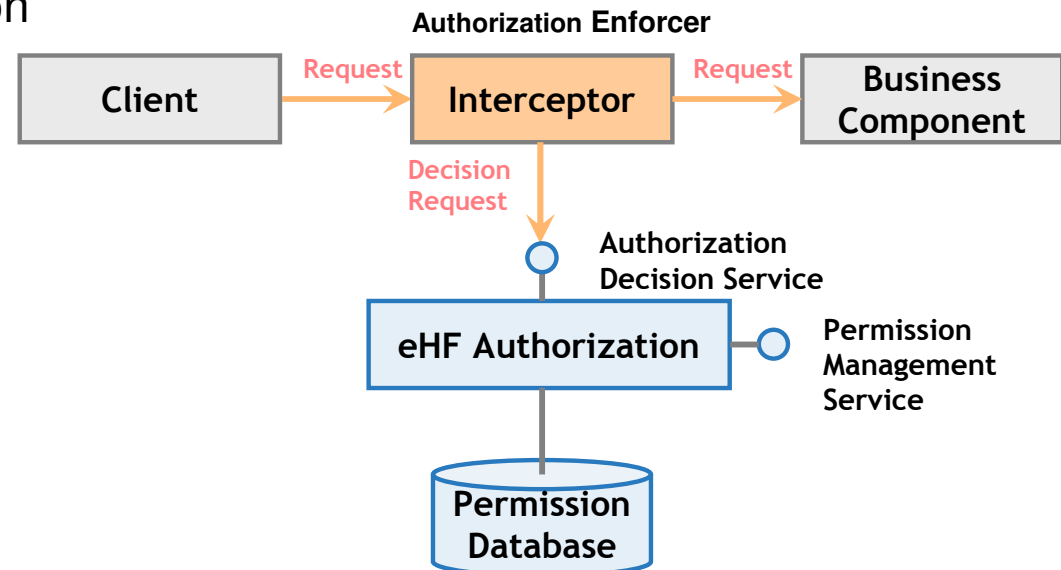
- Provides user identity verification
- Enforces authentication for web applications
- Utilizes Java Authentication and Authorization Service (JAAS)
- Extends Java Servlet 2.4 Specification (JSR 154)
- Supports multiple authentication schemes:
 - Form-based
 - HTTP Basic
 - X.509 Client Certificate
 - Secure PIN



eHF Security

Authorization Module

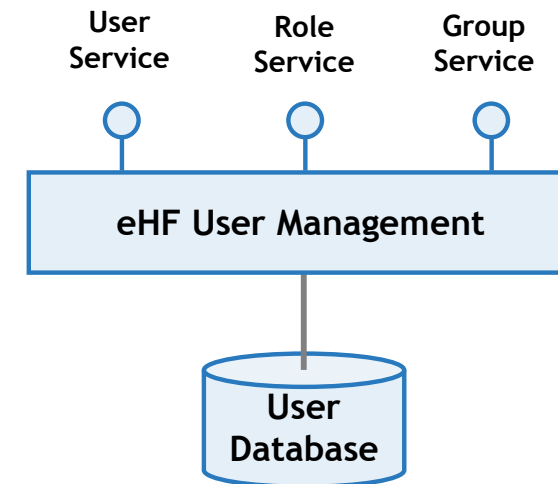
- Permission Management
 - Assignment of predefined permission collections (“Named Profiles”)
 - Creation and assignment of custom profiles
 - Enables hierarchical role based access control (RBAC)
- Business logic separated from security logic
- Authorization enforcement separated from authorization decision
- Security logic applied via AOP (interceptors)



eHF Security

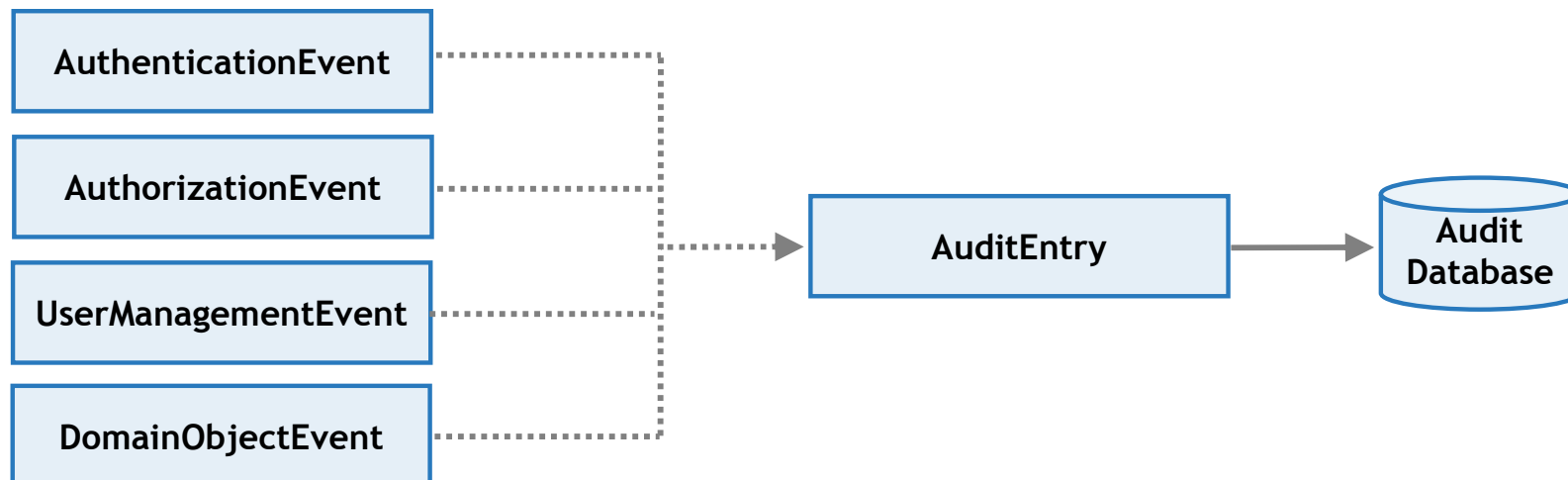
User Management Module

- User data management
 - Personal and contact data
 - Identifiers
 - Management of passwords according to configurable syntax and login policies
- Role Management
 - Role hierarchies
 - Enables hierarchical role based access control (RBAC)
- Group (Organization) Management
 - Group hierarchies



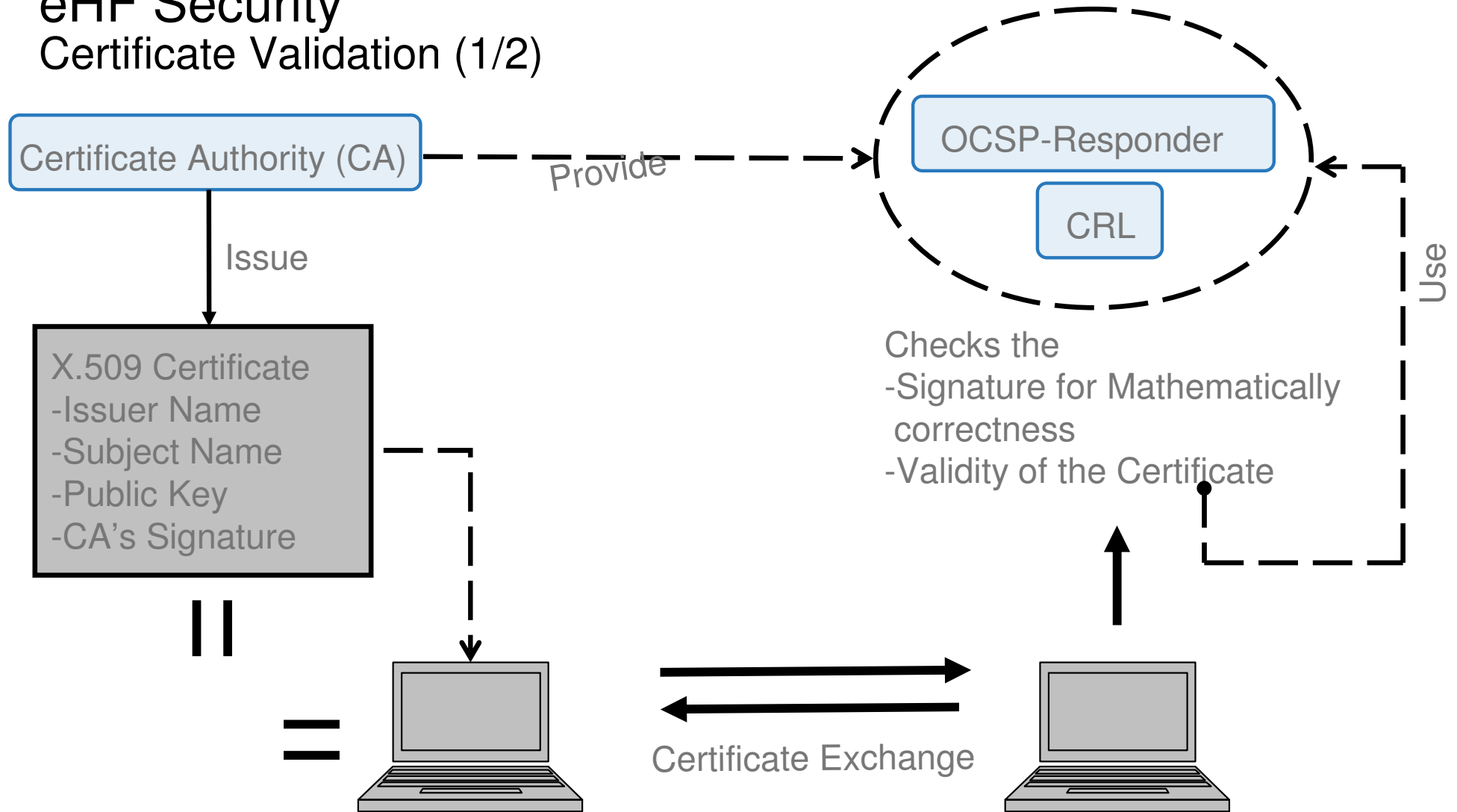
eHF Security Audit Module

- Provides storing of audit relevant information (e.g. a failed authentication)
- Can be consulted for security analysis
- Ships with adapters for different logging/persistence mechanisms
- Is based on the eHF Messaging infrastructure
- Uses eHF Transform to transform audit events



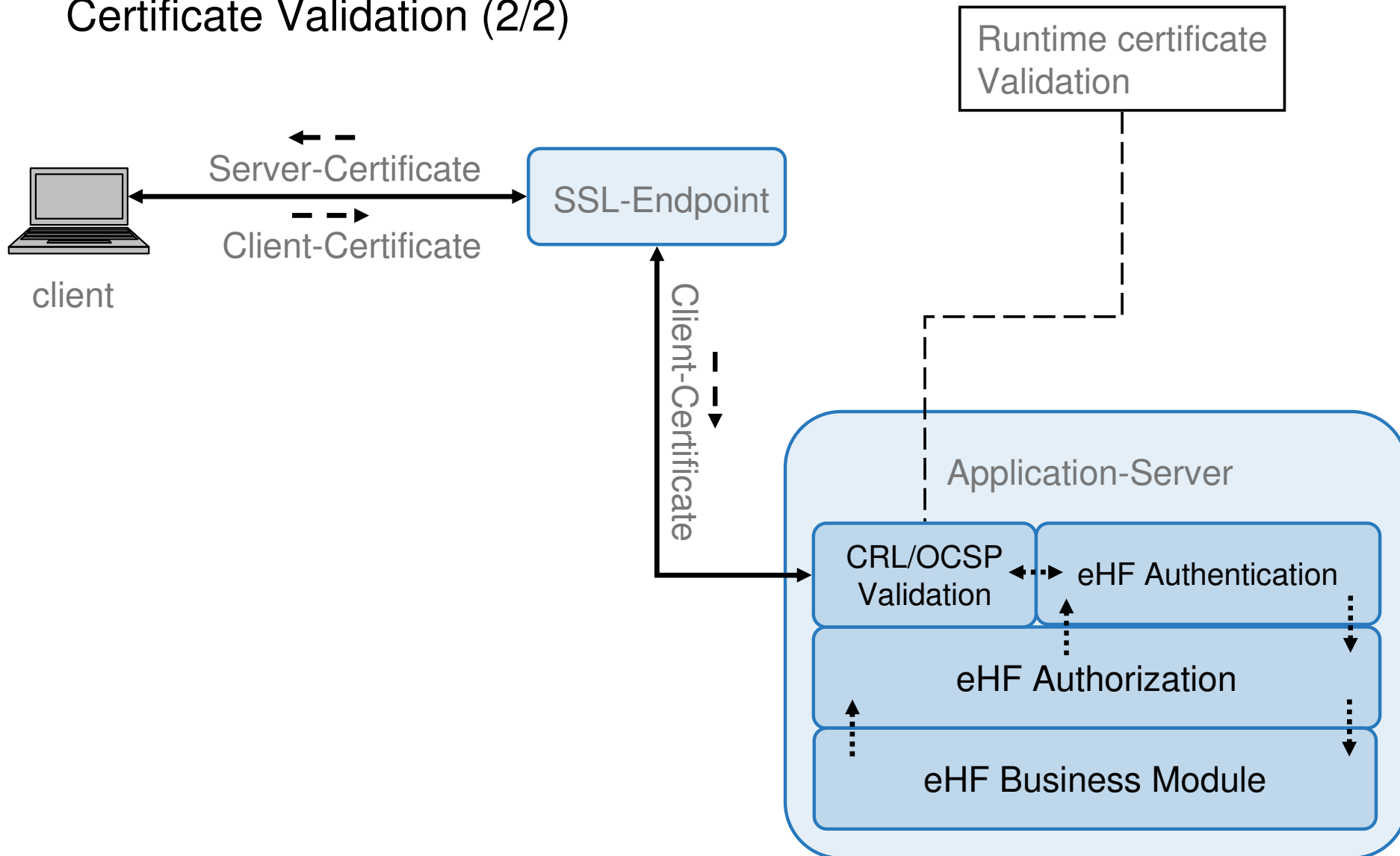
eHF Security

Certificate Validation (1/2)



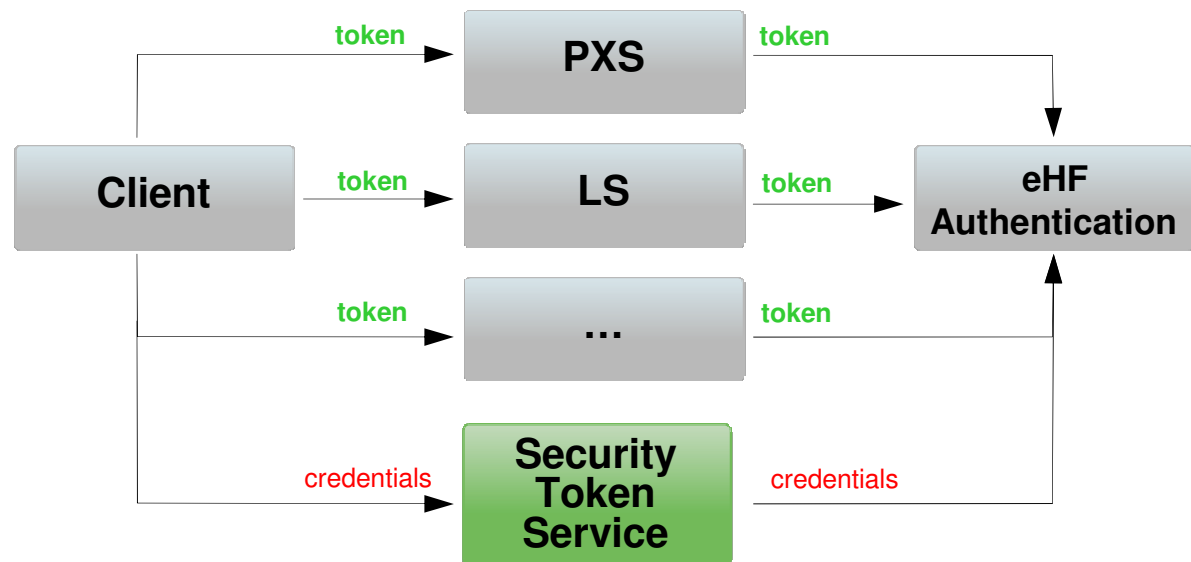
eHF Security

Certificate Validation (2/2)



eHF Security Token Service (1/2)

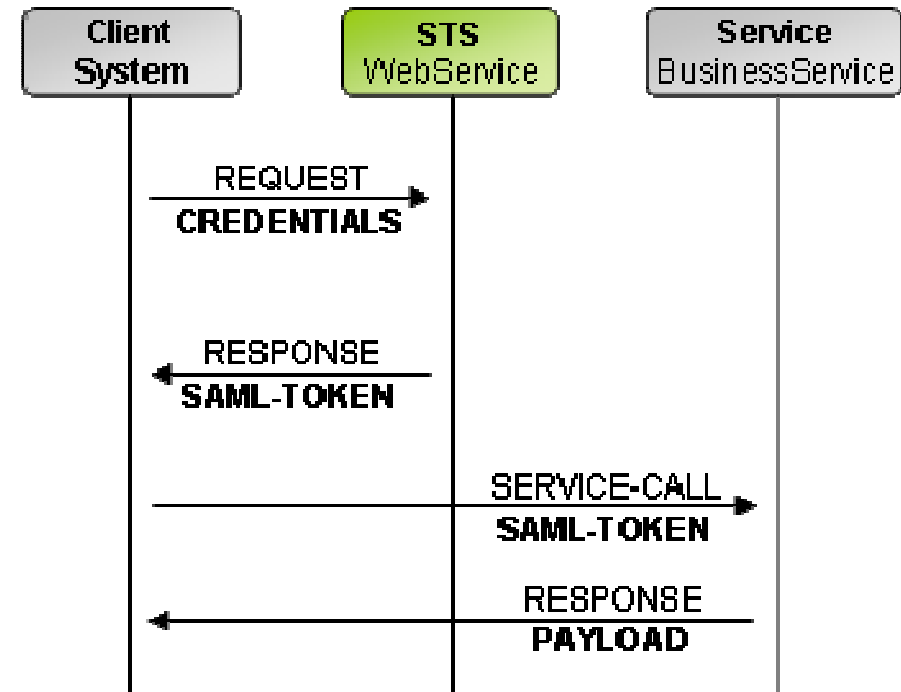
- Solution for single sign on.
- Basic idea of the new functionality is "Security as a Service"
- Trust in the secret is exchanged by trust in the token.
- The token asserts "**I am user x**"



eHF Security

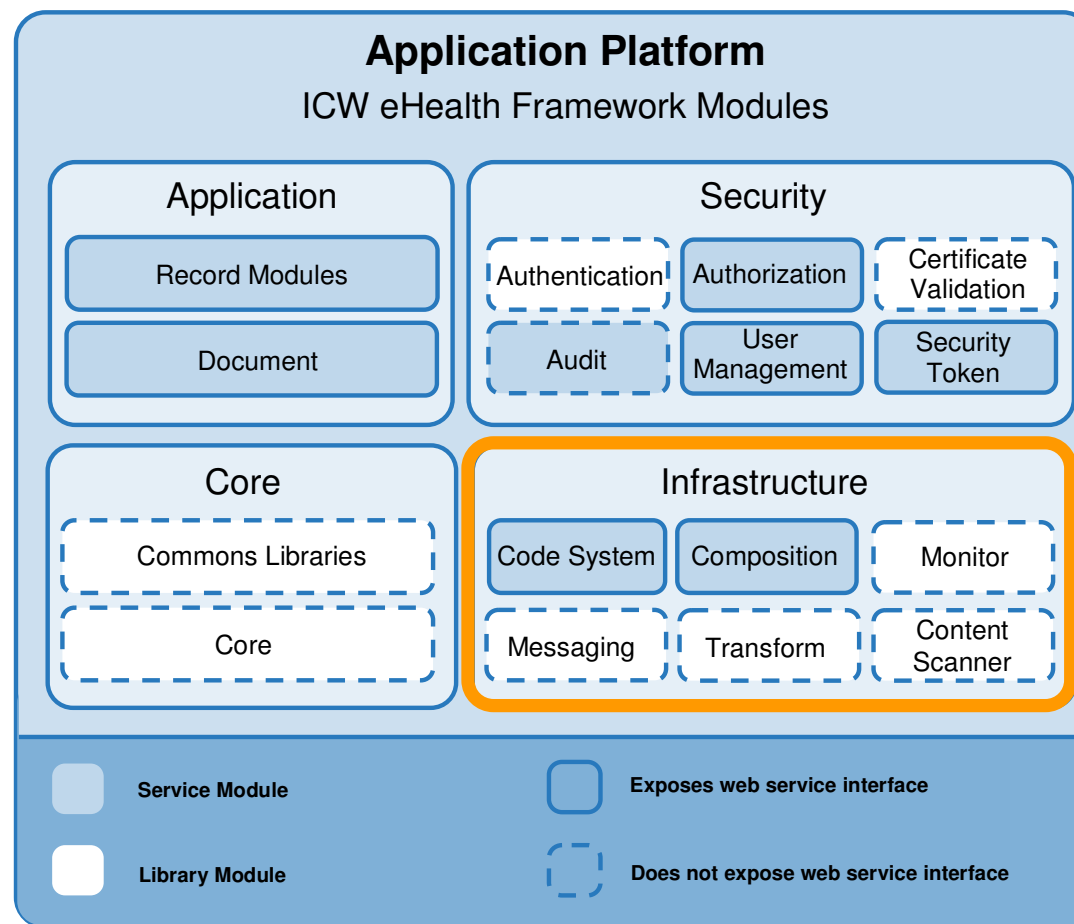
Token Service (2/2)

1. The client requests a token from the STS
2. The STS responds with a security token in the form of a SAML assertion (in case of successful authentication of the user)
3. The client invokes the business service, with the client's request the security token is also conveyed to the service (currently in the HTTP header)
4. The business service validates the token and executes its logic on behalf of the user identified with the security token.



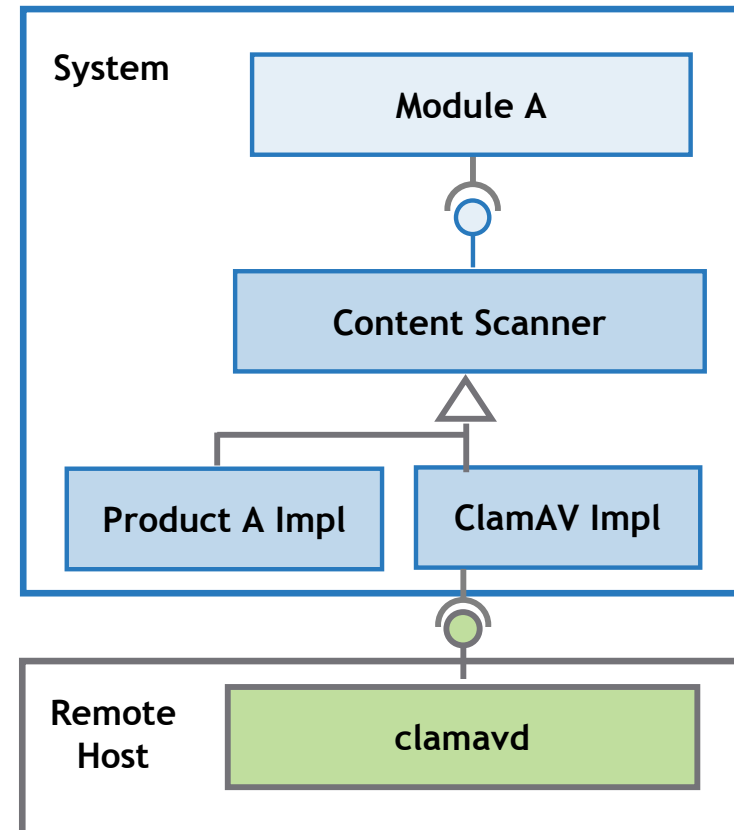
Agenda

- eHF Modules
 - Core
 - Security
 - Infrastructure
 - Application
- Examples
 - Add Observation



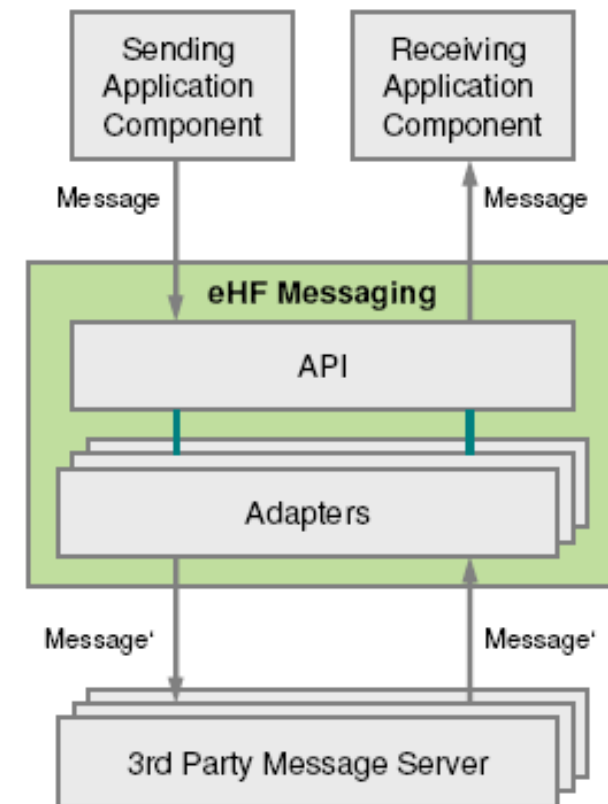
eHF-Infrastructure Content Scanner Module

- Java API to abstract from content scanner software (e.g. Antivirus software, malicious javascript content filters, ...).
- Comes with a default implementation using a remote ClamAV scanning host
- Implementation for Eicar test signatures available for testing the application
- Supports streaming → Low memory footprint

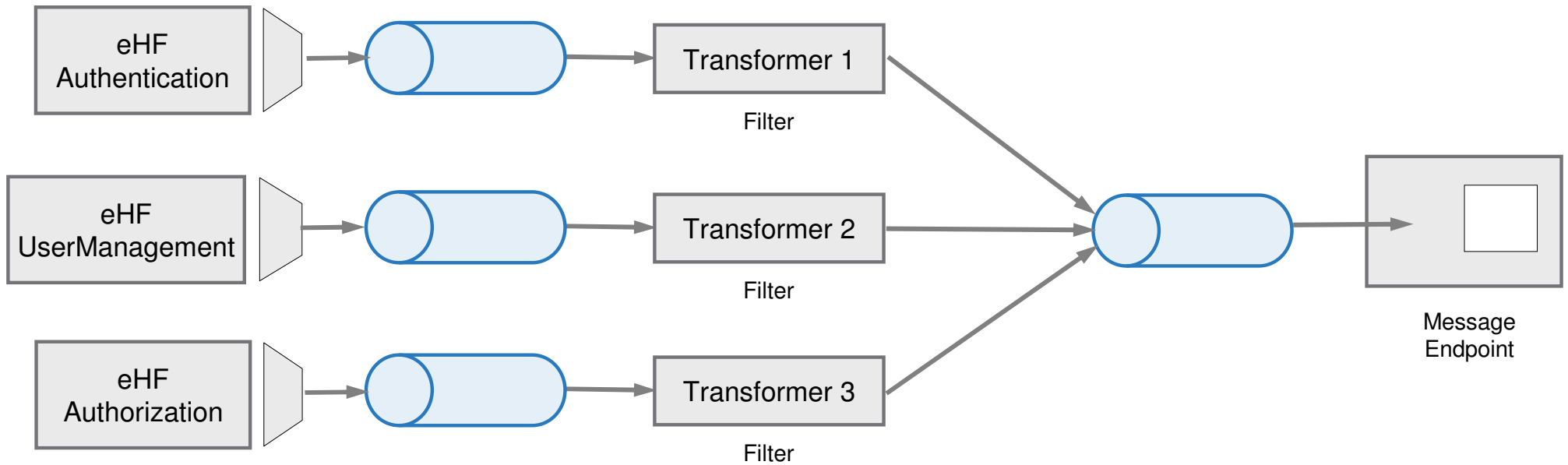


eHF-Infrastructure Messaging Module (1/2)

- Design principles
 - Additional layer to abstract from concrete messaging technologies.
 - Easily adaptation of new messaging systems.
 - Pipes and Filters architecture.
 - Enables Event Driven Architectures.
- Infrastructure Support
 - Thread-safe in-memory message queue (point2point) implementation.
 - Java Messaging Service (JMS) support through Spring.



eHF-Infrastructure Messaging Module (2/2)



eHF-Infrastructure

Codesystem & CS Repository Module

Codesystem

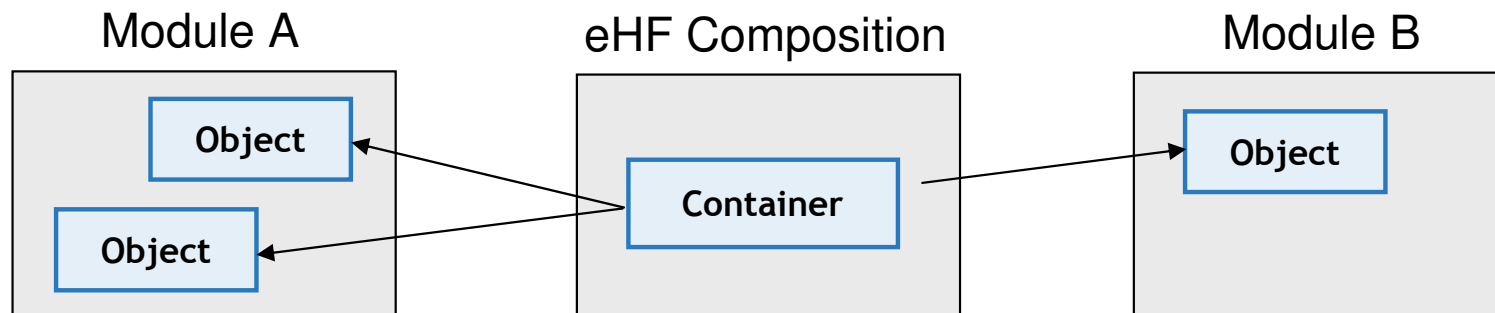
- Implements the technical concept of controlled vocabularies, dictionaries and classification systems.
- API to query and resolve codesystems
- Provides support for tasks like
 - Exchange of standardized information
 - Internationalization
 - Localization
 - Specification of types (gender, ...)

Codesystem Repository

- Contains coded information to describe concepts such as:
 - Administrative data (e.g. country, language, ...)
 - Medical terms (type of diagnosis, encounter, ...)
 - System and error messages
- Holds the actual coded data including translations in XML
- Bootstrapping of the data during deployment

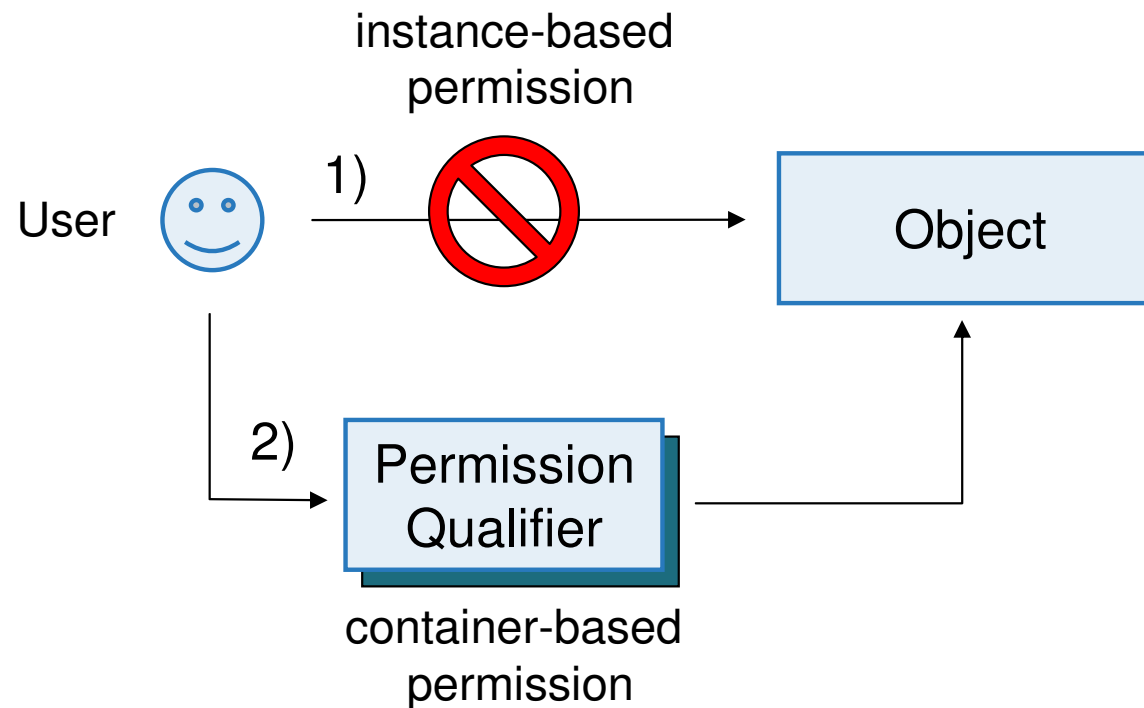
eHF-Infrastructure Composition Module (1/2)

- Creates inter-module links between domain objects
- Composes arbitrary medical data objects and documents in a container
- Provides functionality to manage containers, to put objects into a container and to retrieve objects from a container
- Enables fast sorting and filtering of objects in the container
- Provides an extended security concept: Container-based authorization



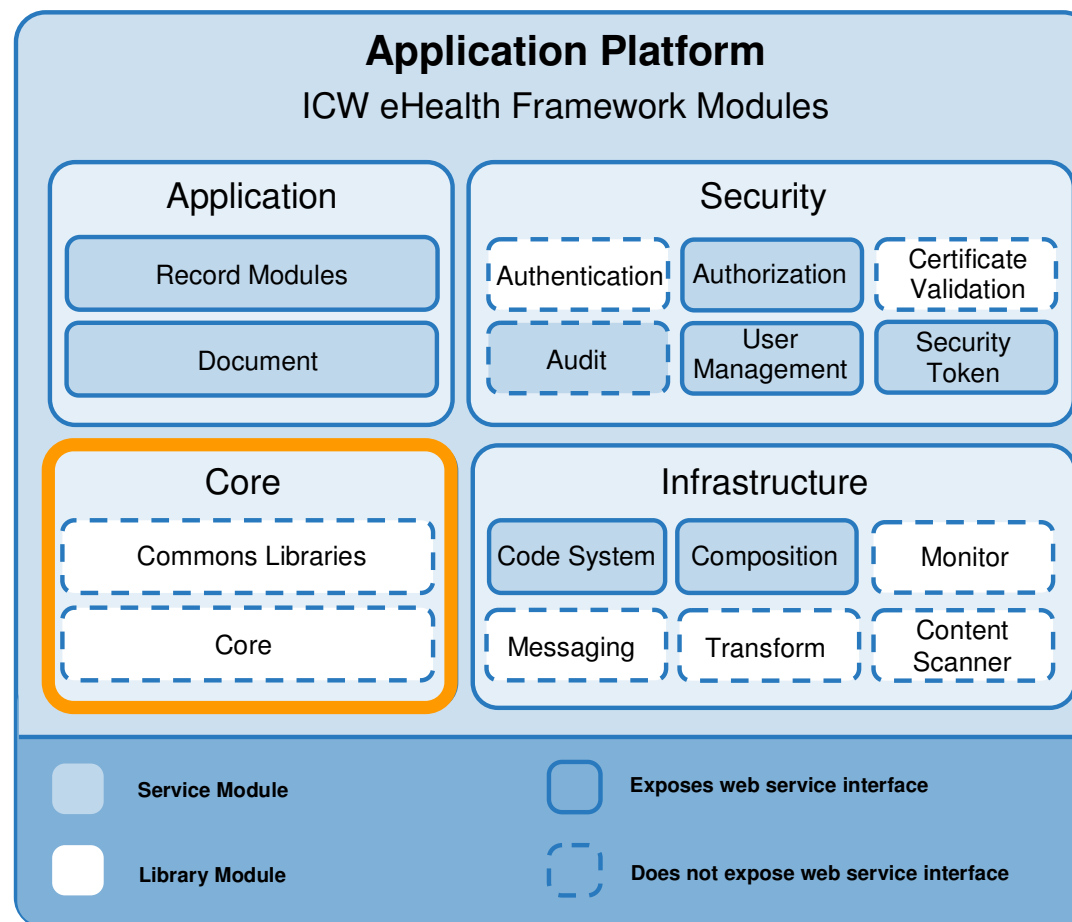
eHF-Infrastructure Composition Module (2/2)

- Container-based permissions may allow access to an Object, even when instance-based permissions would not allow access to this Object



Agenda

- eHF Modules
 - Core
 - Security
 - Infrastructure
 - Application
- Examples
 - Add Observation

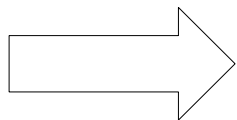


eHF-Application Document Module

- Archiving of arbitrary content
 - Binary content (Pictures, PDF, Word document, etc...)
 - Textual content (plain text, XML, style sheets, etc...)
- Uploading and downloading of documents.
- Can serve XML together with its corresponding XSL (e.g. emergency data set).
- Versioning of documents.
- Also serves as a storing mechanism for other modules, such as eHF Record.

eHF-Application Document Module Design Principles

- Separated saving of document metadata and document content in order to possibly customize storage mechanism in the future.
- Low memory profile
 - Use of streaming techniques
 - Document is never held in memory
 - Just metadata is loaded into memory (is faster, too!)



Enables handling of big documents

- Dedicated servlet for download (reduces load on application).
- Different set of document types possible for each localization.

eHF-Application Record Module

- Stores the central information of an electronic health record
- Using a structural coding based on HL7 v3 RIM general data structure.
- Contains business logic of the health care domain
 - CRUD
 - Linking between medical data
 - Pseudonymization
- Administrative data (personal and organizational data)
- Structured basic medical data (medical sensitive information): Encounter, Observation, Substance Administration , ...

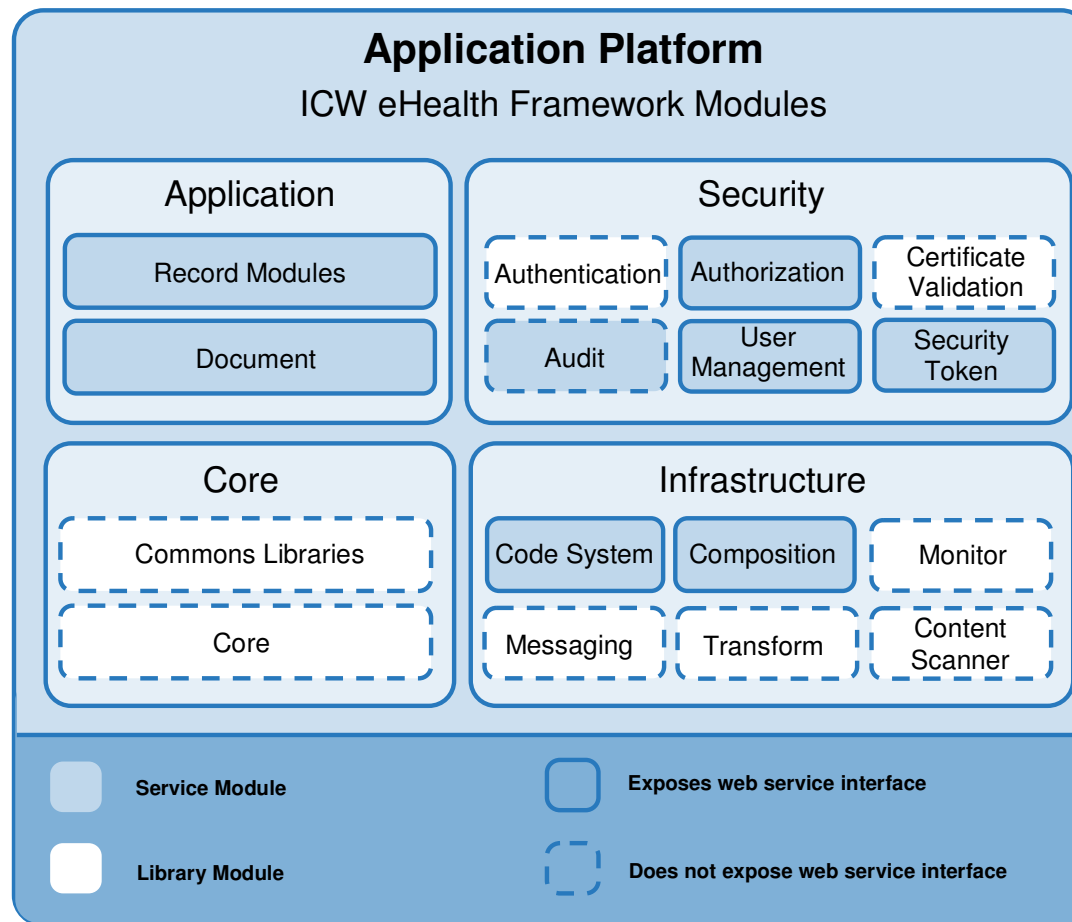
Agenda

- eHF Modules

- Core
- Security
- Infrastructure
- Application

- Examples

- Add Observation



Thank you for your attention!

<presenter>@icw.de