A Secure Next Generation Service Platform: Parlay

Gerald Lorang
T-Systems, ITC Security
Gerald.Lorang@t-systems.com

OMG DOC*sec Workshop*
2003.03.07

Overview of the Presentation

- Introduction
- Prerequisites
- PKI
- Architecture

- CCM mapping
- Component Architecture
- Security

Introduction 1

- **Service- Platforms**
  - Provide standardized means for
    - ï single sign-on
    - ï administration of services
    - ï Load balancing
    - ï Reliability of the systems
  - Security is enforced by the platform,
    - ï abstraction of security
    - ï Security is transparent for the services
    - ï Accountability is handled by the platform

Introduction 2 – Parlay basics

- **Parlay offers**
  - Service Management
  - Service Trading
    - ï Services are selected based on required properties
    - ï Access to services can be handled based on the properties of a person, e.g. age
  - User Management
  - Groups
  - Service Contract Management

Prerequisites for the Secure NGSP

- **CORBA Security Services**
  - Message protection
  - Access Control / Auditing
  - Authorisation Tokens (ATLAS) for delegation purposes (Single sign-on etc.)
- **Threading support**
- **Public Key Infrastructure**
  - Checking validity of certificates
- **Components**
  - Platform Objects have multiple interfaces
- **Optional: Transactions and NR**
  - Will be handled by the CCM container implementation

PKI 1

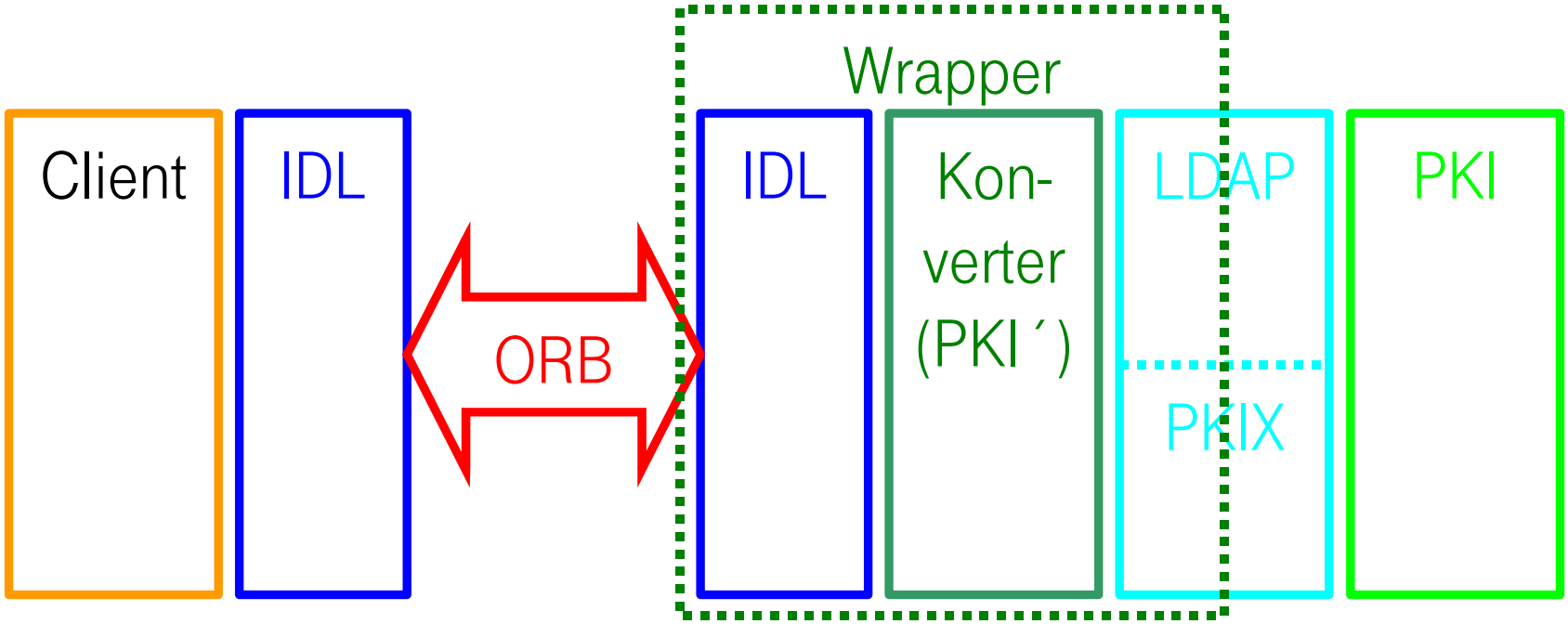
- **Public Key Infrastructure**
  - Administration of certificates
  - Generation of Certificates
  - Checking validity of certificates

- **DTSC- Group - University of Queensland, Australia**
  - Definition of a CORBA PKI binding
  - Adopted OMG Standard

PKI 2

- **CORBA binding:**
  - Provides an easy to use API for accessing any PKI supporting it from a CORBA application
  - Two possible solutions
    - ï PKI- Wrapper, works with basically any PKI
    - ï The PKI provides the CORBA interface directly

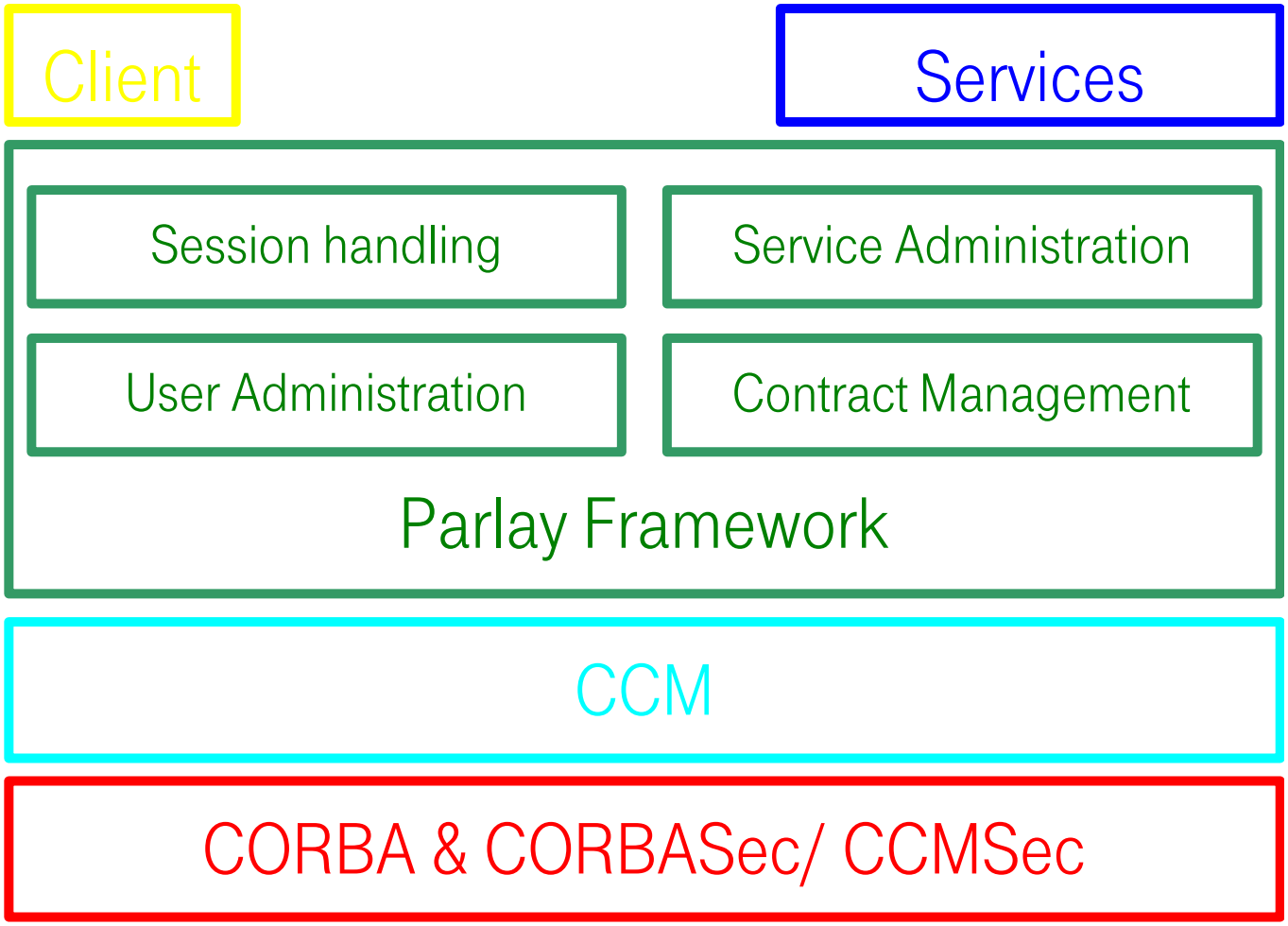
PKI- Wrapper working with any PKI



Architecture of the Platform

- **Framework**
  - Administration of Services
  - Session Handling
- **Services**
  - Basic Services
    - ï Authentication
    - ï Heartbeat (checks if a system is active, cf. ping)
    - ï Load Balancing (limited)
  - Customer Services
- **The overall archicture leads to a modular design**

Architecture of the Platform



Design and Implementation of the Platform

- **Framework**
  - Component model of the framework
  - Detailed threading model
  - Interfaces and interaction diagrams are specified in the Parlay standard (current Version 3.1)
  - First version without any security other than authentication, but with security in mind

- **Services**
  - Basic services - required to run the platform itself
  - Customer services, sample application

CCM- Mapping

- **Parts of the framework will be components, eg.**
  - Framework core
    - ï Service management
    - ï User Management
  - „Initial“ Component
    - ï Authentication
    - ï Opens Access Session
  - Access Session Component(s)
    - ï Sesion control interface
    - ï External Framework interfaces
  - Security will be integrated in all relevant components
- **Services will be components**

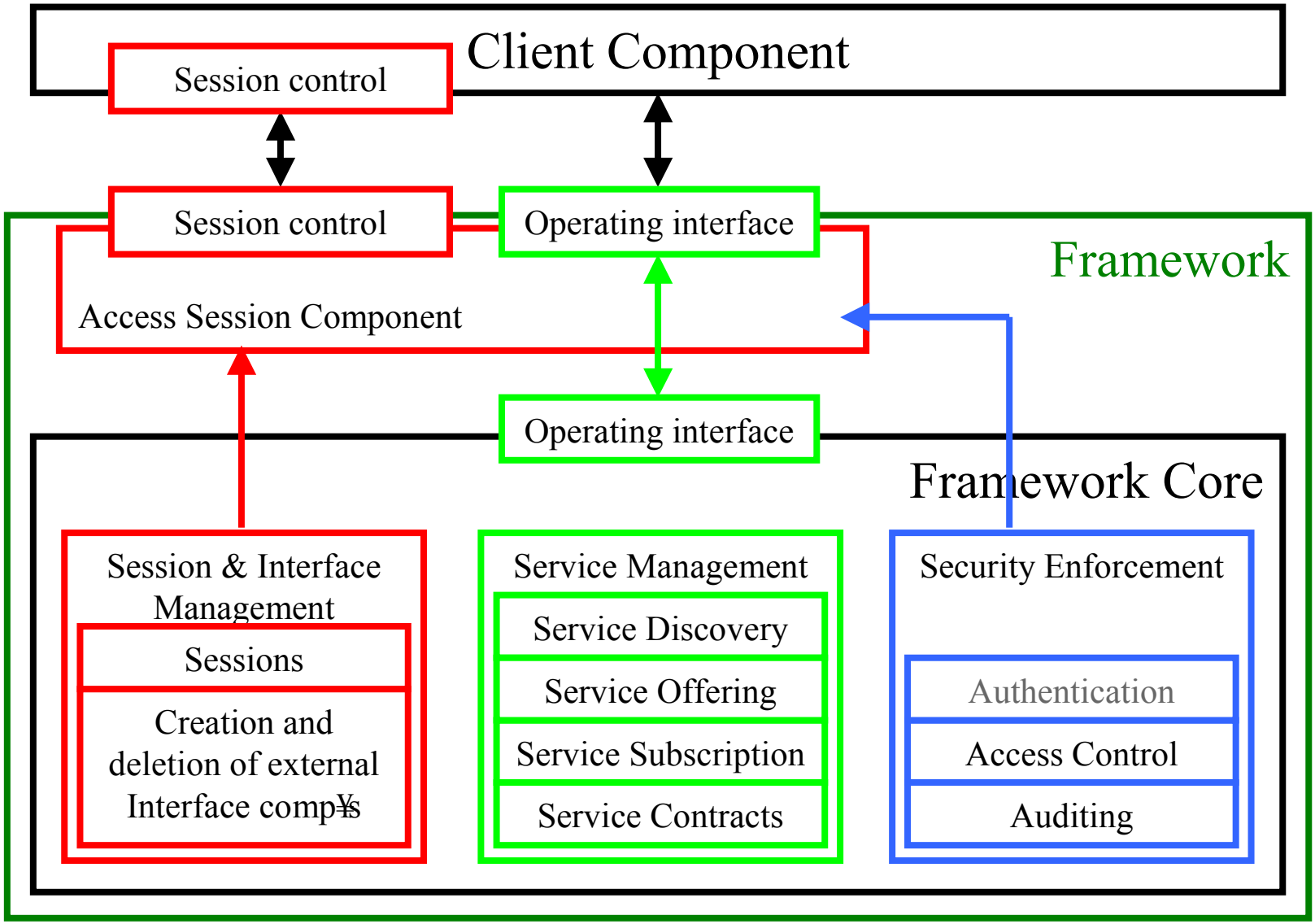
CCM – Mapping Architectural considerations (1)

- **Management Core - internal component without direct connections to the outside world**
  - Service Management
  - Service Contract Management
  - User Management (subscription etc.)
  - Lots of data are required across these functions making separate components infeasible

CCM – Mapping Architectural considerations (2)

- **Interfaces - components providing access to external systems**
  - Access- session: User-ID, setting User rights for all subsequent session upon availability of CORBASec 2.x
  - All interfaces requested will extend the session (server side only)
    - ï User ID can be obtained without using CORBASec repeatedly
    - ï Access Rights are set accordingly
    - ï session component
    - ï sessions will be silently closed when parent session is closed

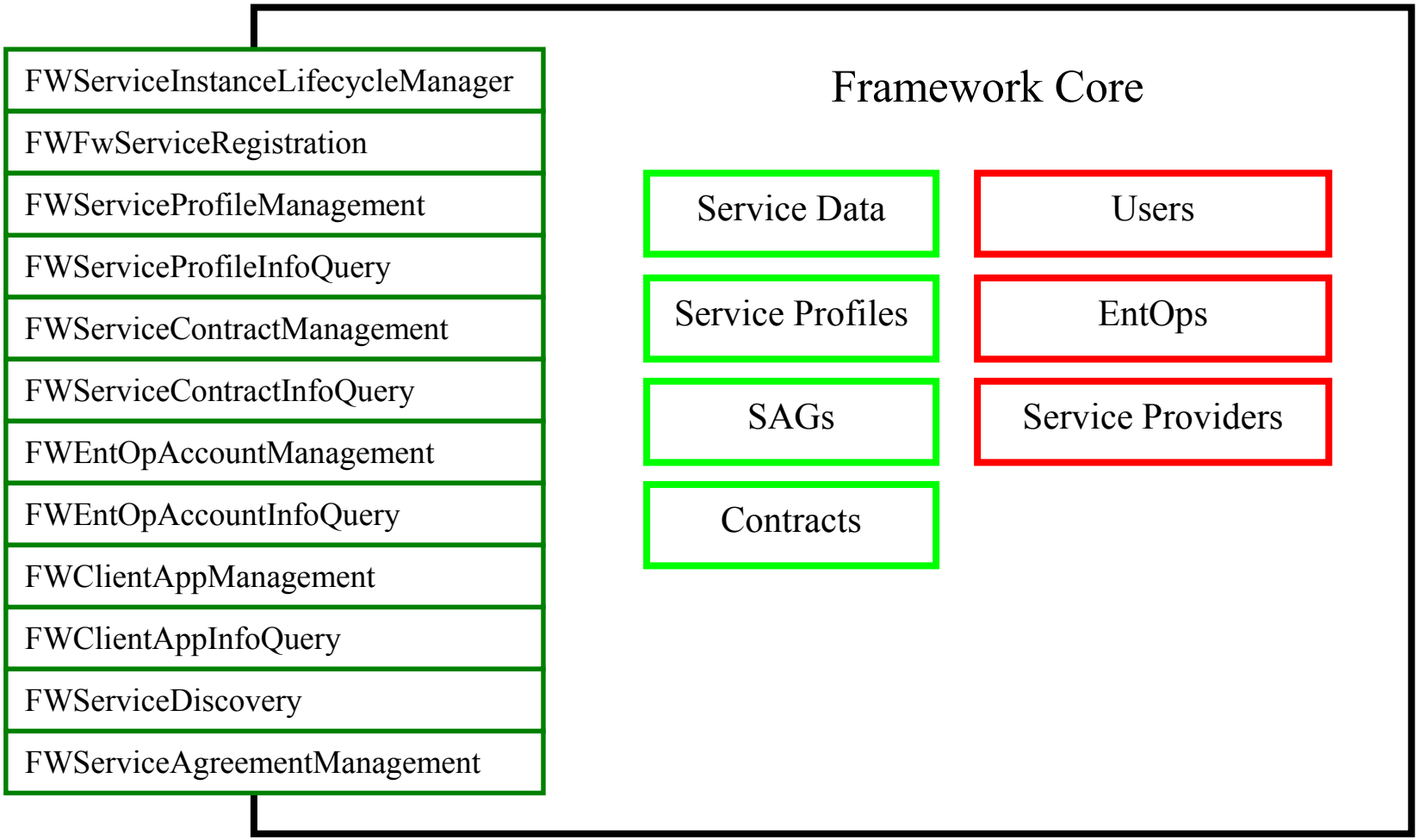
Component Arch. - Overview



Component Architecture Framework core (1)

- Provides the full set of interfaces described in the specification, but modified for the communication with the access session components (internal interfaces)
- Interacts only with the access session components which provide additional information
- Complete user, service, and subscription management

Component Architecture
Framework core (2)



Component Architecture
Initial Component

- **User Authentication**
  - Using the underlying security mechanism
  - Using a special security mechanism implemented for the Framework

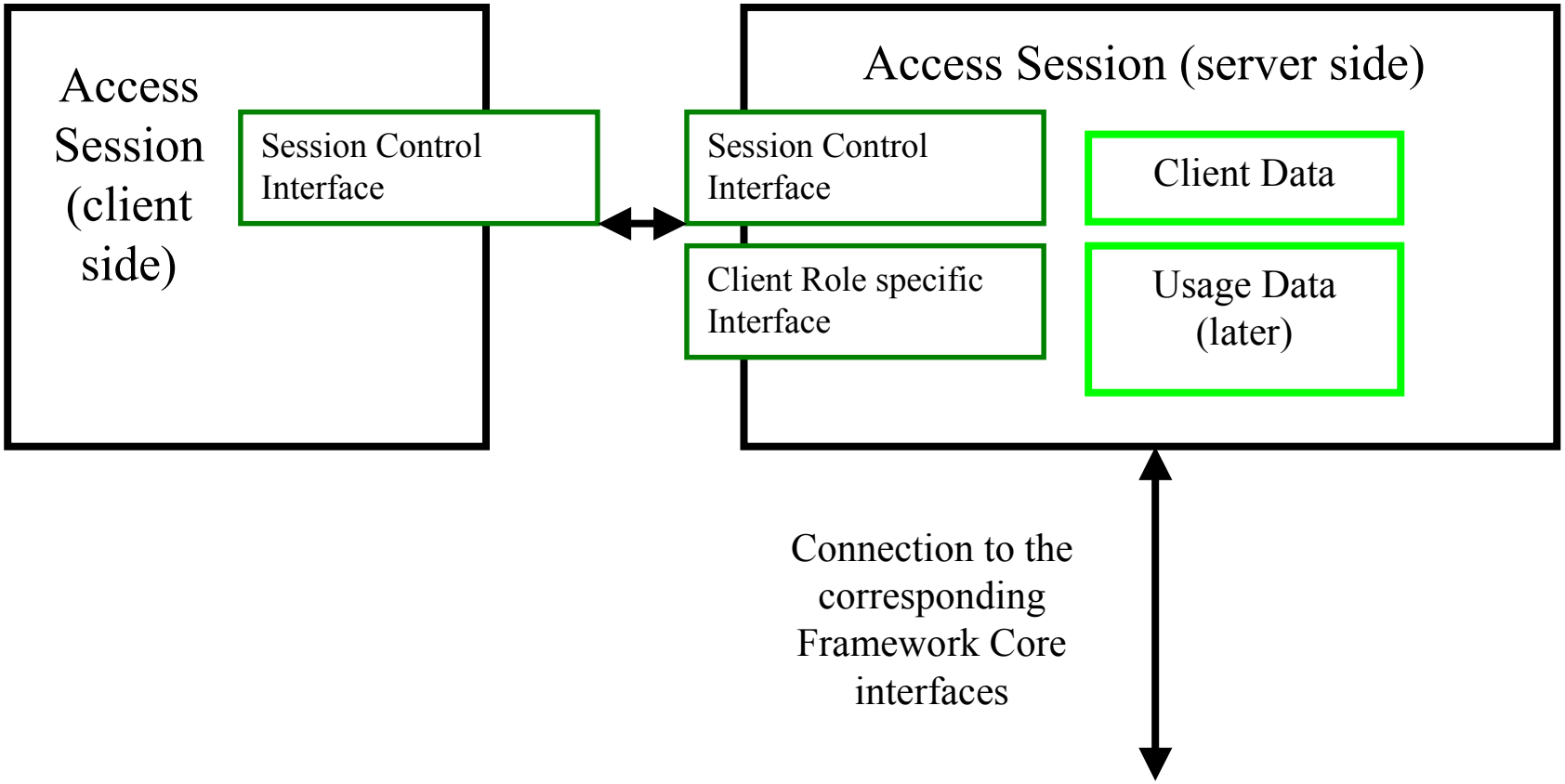
Component Architecture Access Session (1)

- All access to the framework´s functionality and administration functions via the access session (the creation of an extra service session proves unnecessary)
- Several different access session components
- Set of interfaces depending on the client´s Role
  - User
  - Service Provider
  - Enterprise Operator

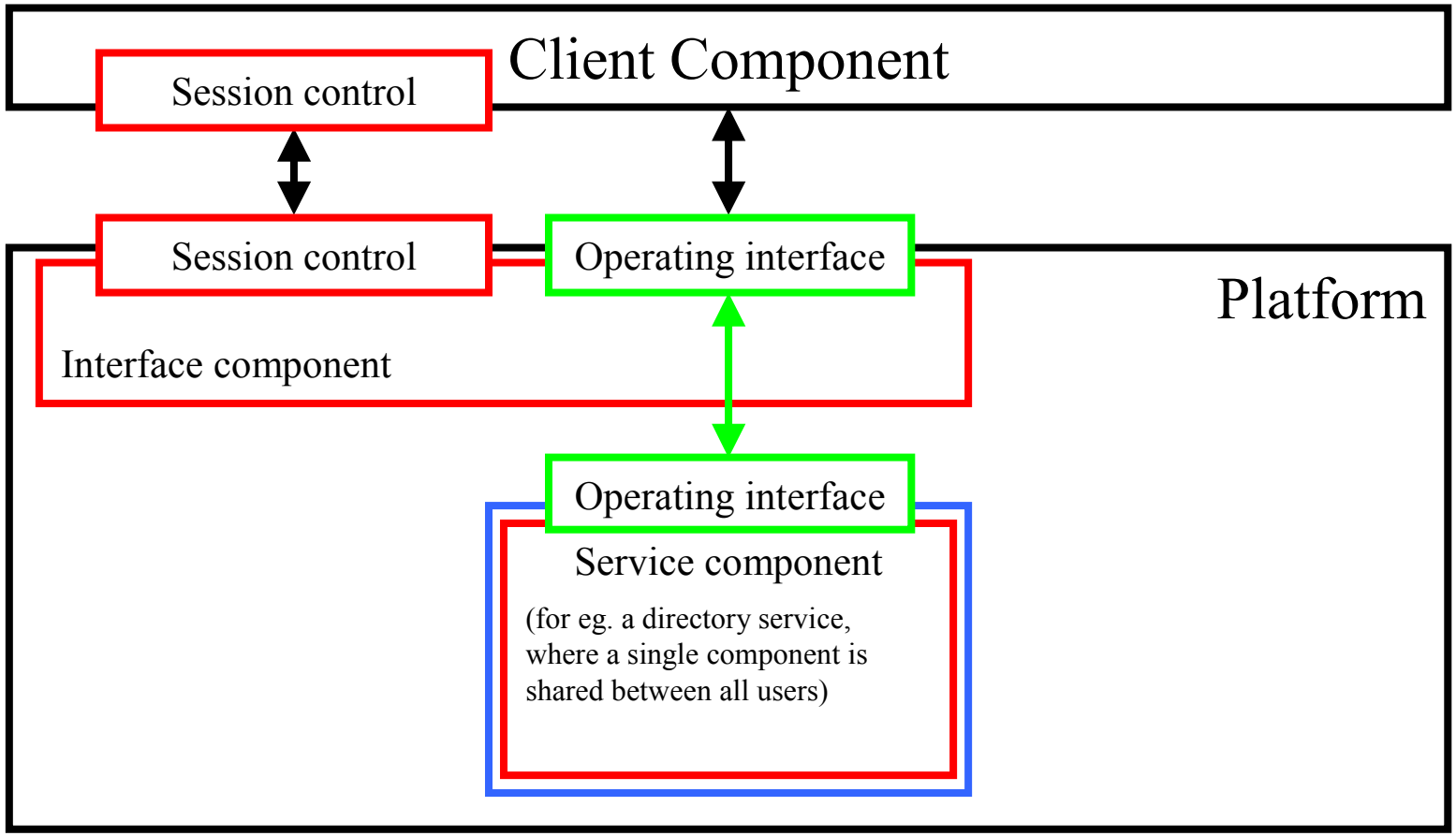
Component Architecture
Access Session (2)

- The access session object belongs to the user
- The framework receives user information and filters the information sent back to the client
- To be solved: reasonably easy changes of the role (currently a user would need to close the access session and re-authenticate with a different user ID)

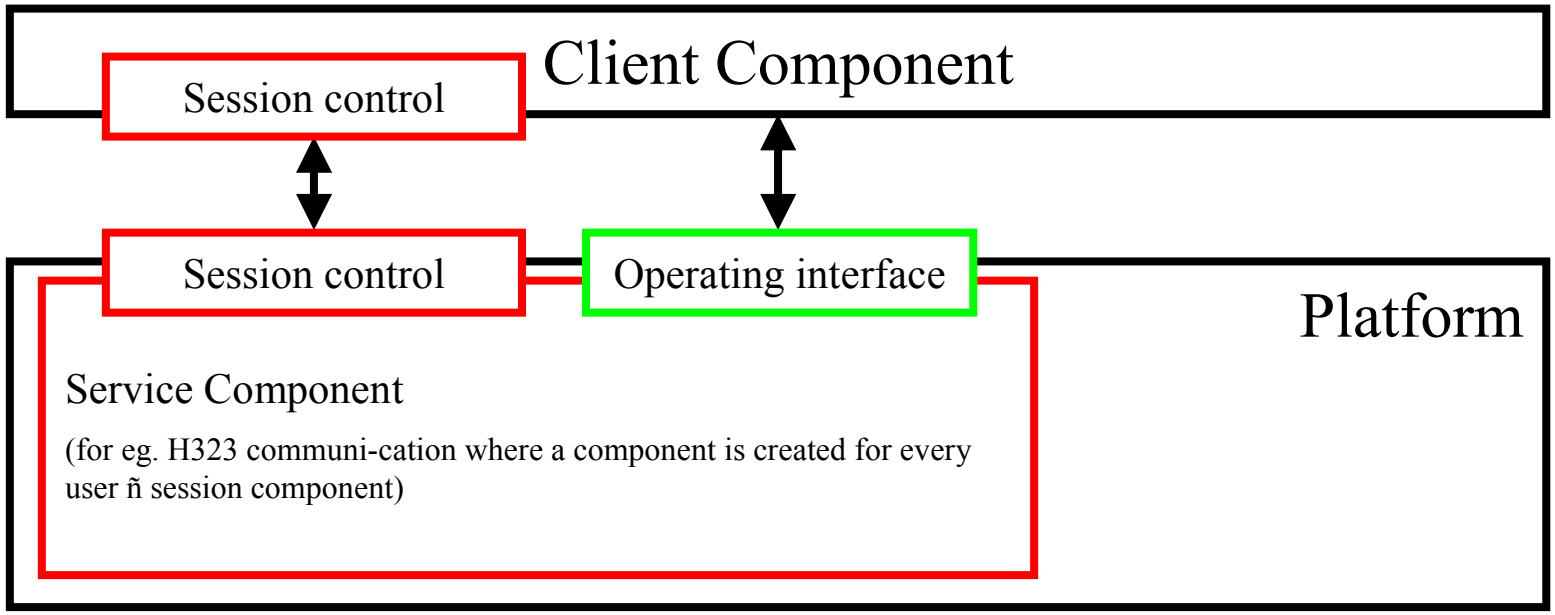
Component Architecture
Access Session (3)



Component Architecture
Non user-specific Services



Component Architecture
User-specific Services



Security 1 – General Requirements

- Secure transport
- Access control on any object carrying or transporting sensitive data
  - Session objects
  - Framework Core
  - Service Objects
- Access control for fault and integrity management
- Shutdown of abandoned sessions by the framework

Security 2 - Security functionality

- **Low level, provided by the CCM security services**
  - SSL
  - CSIv2
  - Access control/ Auditing / NR etc.
- **High Level, inside the platform**
  - Service usage or visibility can be restricted based on several properties of the person accessing the system, e.g. age
  - Service usage only permitted if a contract exists
- **We focus here on the low level security features**

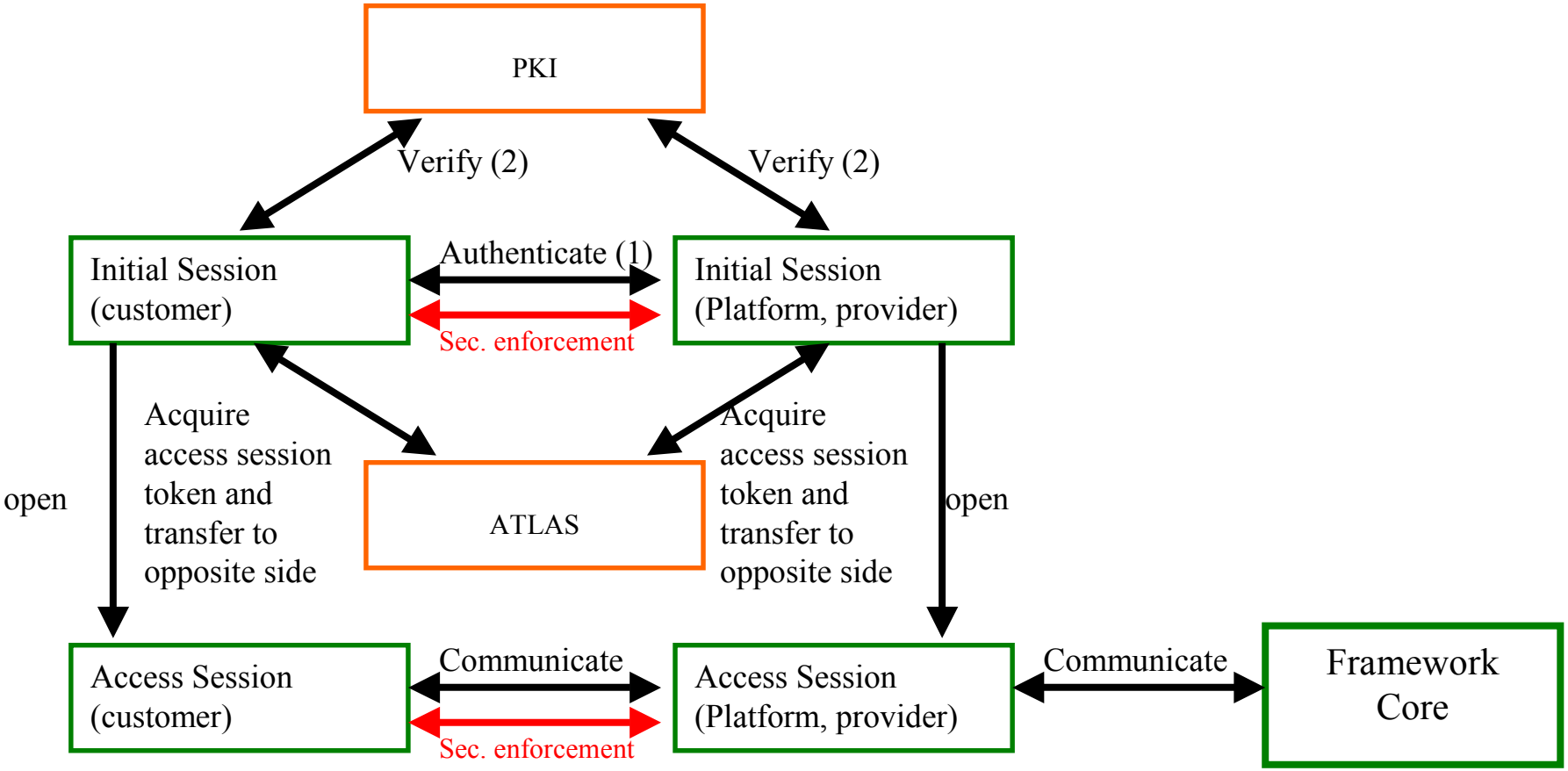
Security 3 – General

- **Based on a secure CCM implementation**
- **Certificate based authentication**
- **Access control**
  - Discrete Access Control for sessions
  - Role Based Access Control to Management functions, service subscription functionality
  - Special considerations for certain services, e.g. multiparty services like conferences
- **Accountability/ Auditing**
  - Handled by the session components

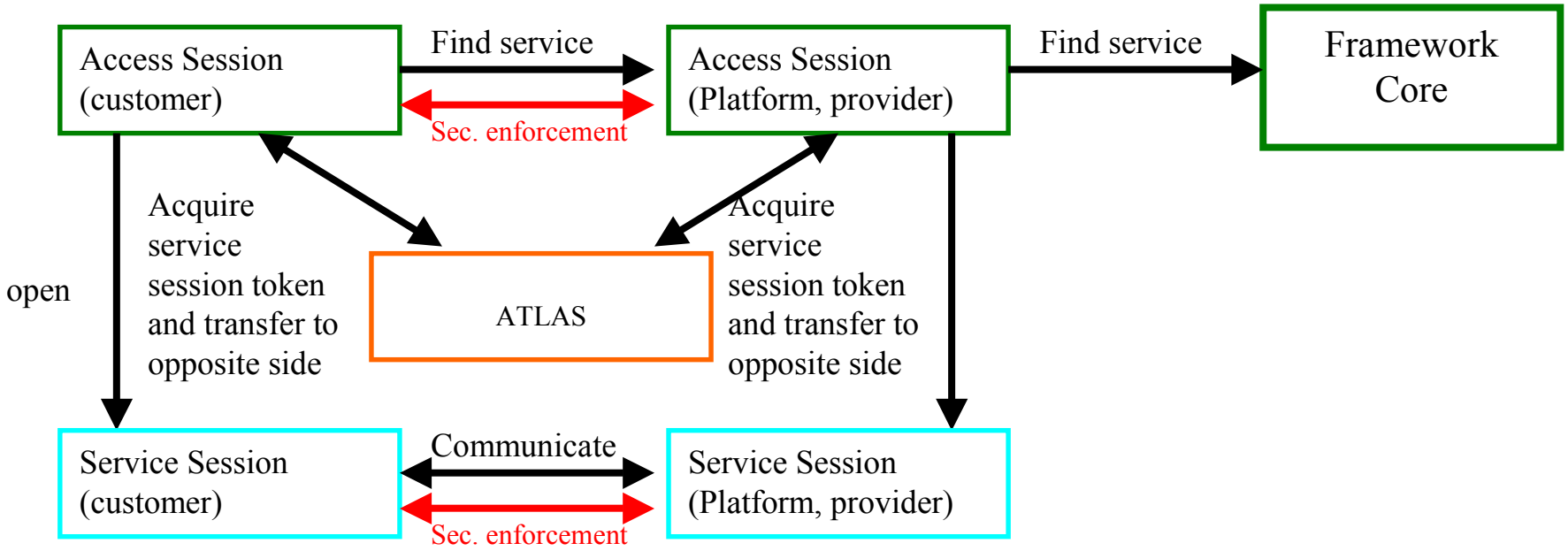
Security 4 – System Architecture

- **Interacting parts**
  - Platform
    - ï Framework Core
    - ï Initial session components
    - ï Access session components
    - ï Service Components
  - Outside
    - ï ATLAS
    - ï PKI

Security 5 – Access Session



Security 6 – Service Session



Security 7 – Trusted Parties and Trust Relationships

- **The PKI always acts as TTP (Trusted Third Party)**
- **Possible Arrangements of ATLAS**
  - ATLAS acting as TTP (as shown)
  - ATLAS may also be hosted by the platform provider, if the customer trusts the provider
    - ï The platform will create all access tokens
  - An ATLAS may reside on both sides, requiring a big client
- **Trust relationships**
  - The customer trusts the platform provider (to some extent)
  - The service provider trusts the platform provider
  - Everybody trusts the TTP(s)

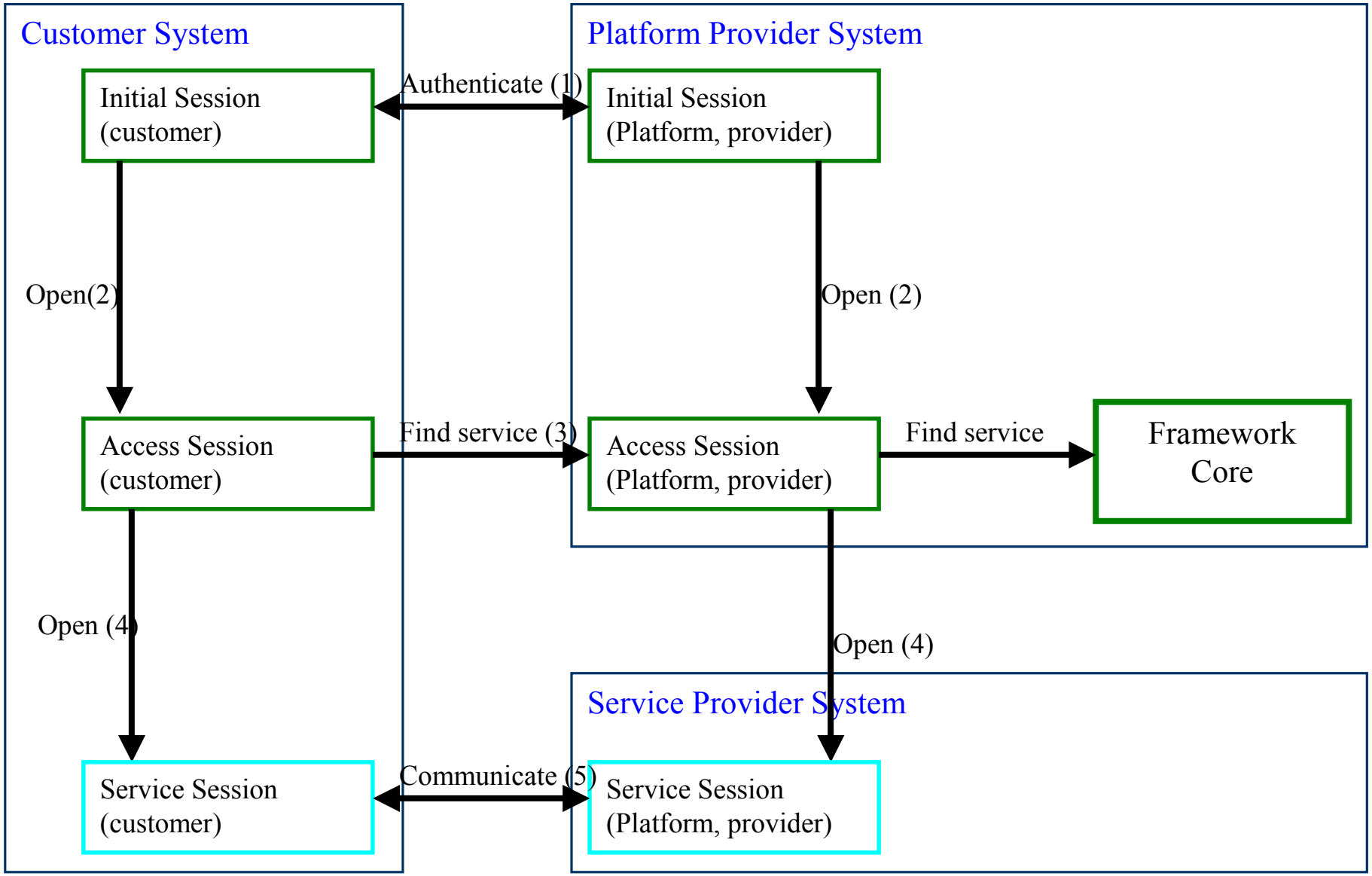
Security 8 – Trust Domains

- **Customer side**
  - Initial, Access, and Service Sessions
  - ATLAS (if located here)
- **Framework**
  - Initial and Access Session
  - Framework Core
  - ATLAS (if located here)
- **ATLAS if acting as TTP**
- **PKI**
- **Goal: Create and assign access tokens only within the same trust domain**

Security 9 – Distribution Scheme

- Currently, the framework core and the service session are on the same machine, connected via local interfaces
- Sessions may run on a diffferent machine than the framework core, requiring access control between elements of the framework
- For load balancing and fault tolerance purposes, it may be considered to have more than one framework core running
- Any remote access between framework core and access session objects requires
  - Access control between framework core and access session components
  - Solution: Delegation using CSIv2 features

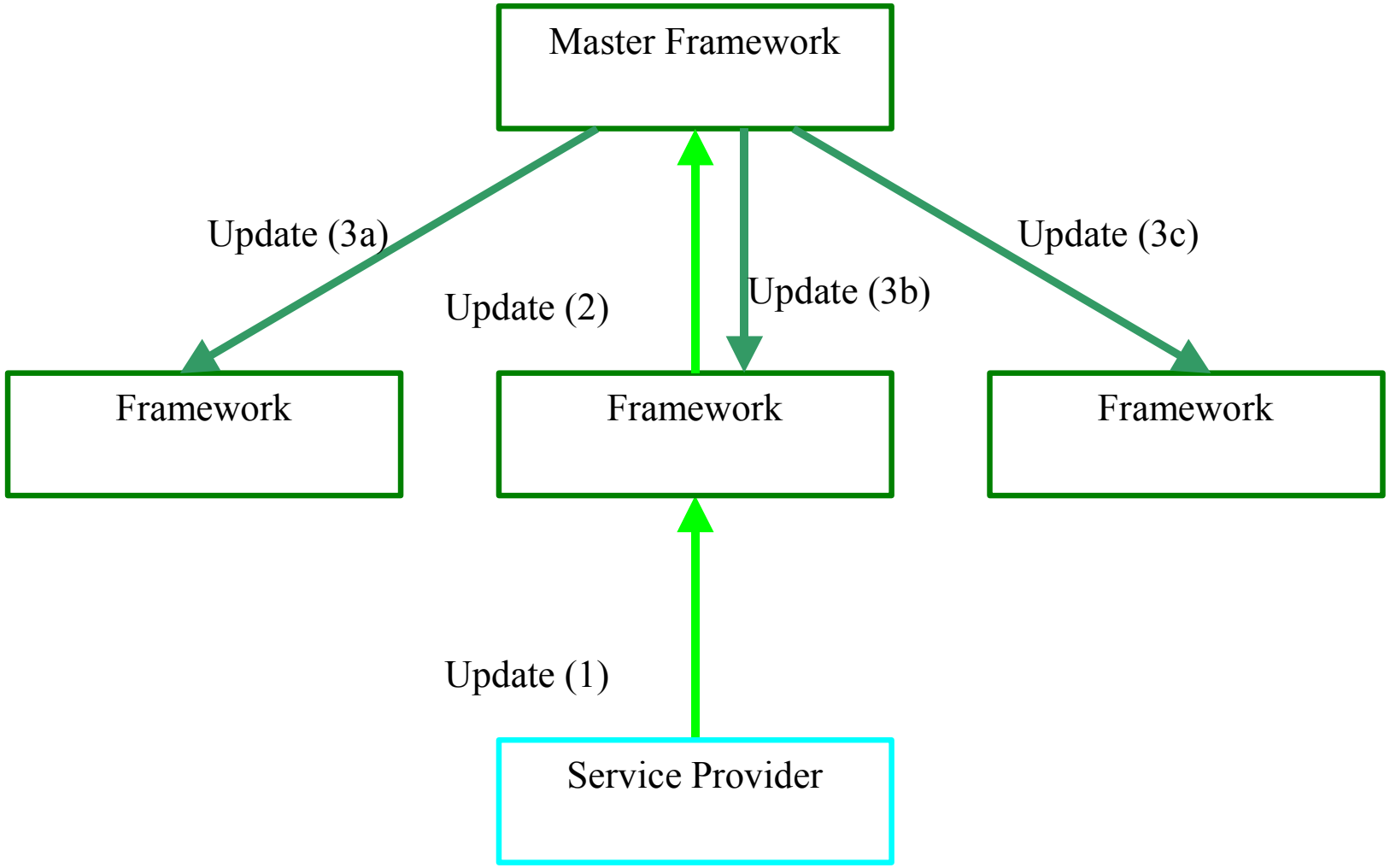
Security 10 – Distribution Scheme



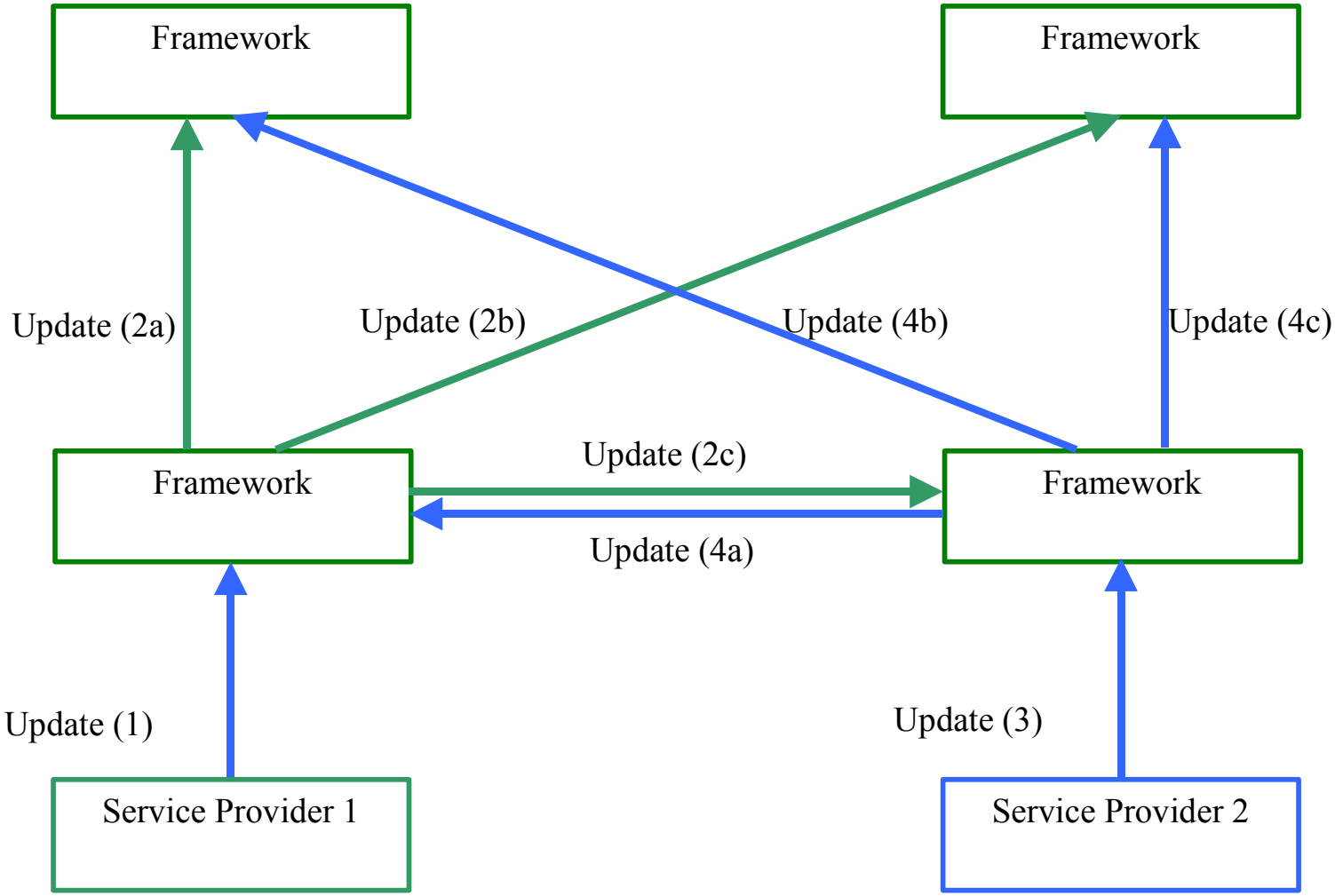
Load Balancing and Fault Tolerance (1) – Fed. Frameworks

- **Two Basic Layouts**
  - Tree
    - ï Relies on a single Master Framework
    - ï Only the Customers can use the Framework if the Master goes down
    - ï No Administration possible if the master is down
    - ï Simple update procedure
  - Star
    - ï The Platform is supposed to remain fully operational if on or more Frameworks go down
    - ï Difficult update procedure

Load Balancing and Fault Tolerance (2)– Tree



Load Balancing and Fault Tolerance (2)– Star



Conclusion

- **The current architecture**
  - solves security problems
  - can fulfil security requirements
  - maintains full compatibility with the Parlay standard
- **Implementation plan**
  - 2002 - done
    - ï Service- and User management
    - ï (Parlay-) Session components
    - ï Documentation
  - 2003
    - ï Services
    - ï Demo Application

Contacts and Partnerships

- Gerald.Lorang@t-systems.com
- This implementation is funded by the IST project COACH

- http://WWW.IST-COACH.ORG

Contacts and Partnerships 2

www.ist-coach.org

Contact Person:

Uwe G. Wilhelm
T-Systems
D-64307 Darmstadt
Germany

Uwe.Wilhelm@t-systems.com