Business Process Model and Notation for Forensic-Ready Software Systems

Lukas Daubner, Raimundas Matulevičius, Barbora Buhnova, Tomas Pitner

daubner@mail.muni.cz

Faculty of Informatics, Masaryk University, Brno

October 21, 2021

Why Forensic Readiness?

- Digital forensic investigation is:
  - Laborious
  - Costly
  - Time-consuming
  - Delicate

- Success is never assured
  - Data might be unavailable, corrupted, or tampered
  - Error in evidence handling jeopardies the process

- Data might me misleading

What is Forensic Readiness?

- Original definition
  - Maximizing the usefulness of incident evidence data
  - Minimizing the cost of forensics during an incident response

- Systematic preparation for forensic investigation

- Proactive measures
  - Opposed to actual investigation, which is reactive

What is Forensic Readiness?

- Approached as a set of general guidelines
  - Collection of evidence
  - Handling of evidence
  - Presentation of evidence
  - Staff training
  - Escalation policies

- Increases likelihood of successful investigation

Forensic Readiness in Software Engineering

- Prepare software system during its development
  - A.k.a. forensic-by-design
- Capable of:
  - Conducting digital forensic processes in a forensically sound way
  - **Producing forensically sound evidence**
- High-level non-functional requirement
- Measures for the failure of security measures

Forensic Readiness in Software Engineering

- It is true that software systems produce a lot of data
  - Logs
  - Documents
  - Database records
- But can we trust them?
- Are they complete?
- Will they help us during the investigation?

Forensic-Ready Software Systems

Requirements

- High-level non-functional requirement
- Further decomposed into:
  - Availability
  - Relevance
  - Minimality
  - Linkability
  - Completeness
  - Non-repudiation
  - Data provenance
  - Legal compliance
- Risk management to identify them

Forensic-Ready Software Systems

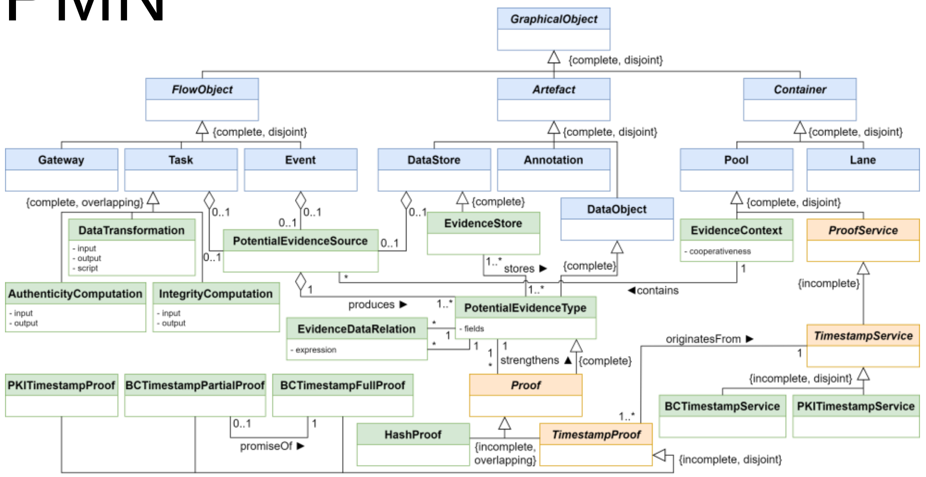
Modeling Challenge

- Provide assistance to the risk management decisions
- Represent the requirements in a concreate system
  - Model incident scenario and the relevant potential evidence
  - Model relationships between the potential evidence
  - Model lifecycle and properties of potential evidence
- Allow reasoning over the systems

BPMN for Forensic-Ready Software Systems

BPMN4FRSS

- Extension for BPMN 2.0
- Model risk management scenarios in BPMN
- Introduce the potential digital evidence
  - Point of origin
  - Handling
  - Storage
  - Relation to other pieces

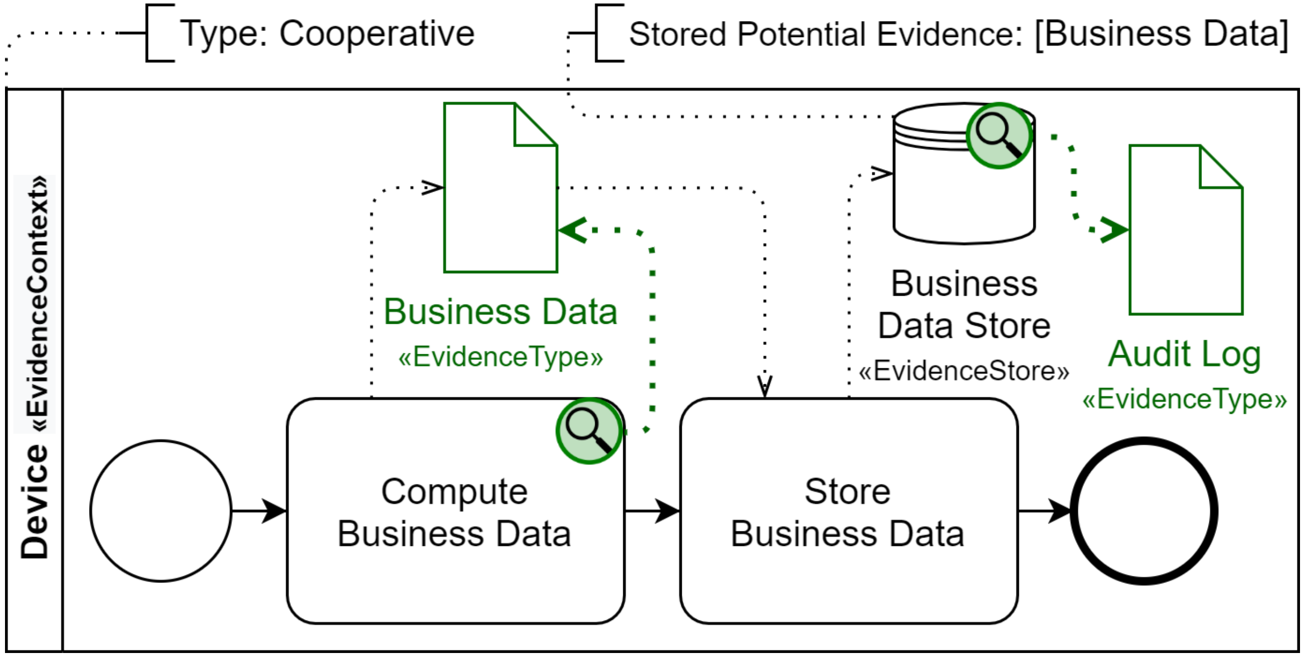


- Possible extensions for specific evidence-assuring mechanism

BPMN for Forensic-Ready Software Systems

Core concepts

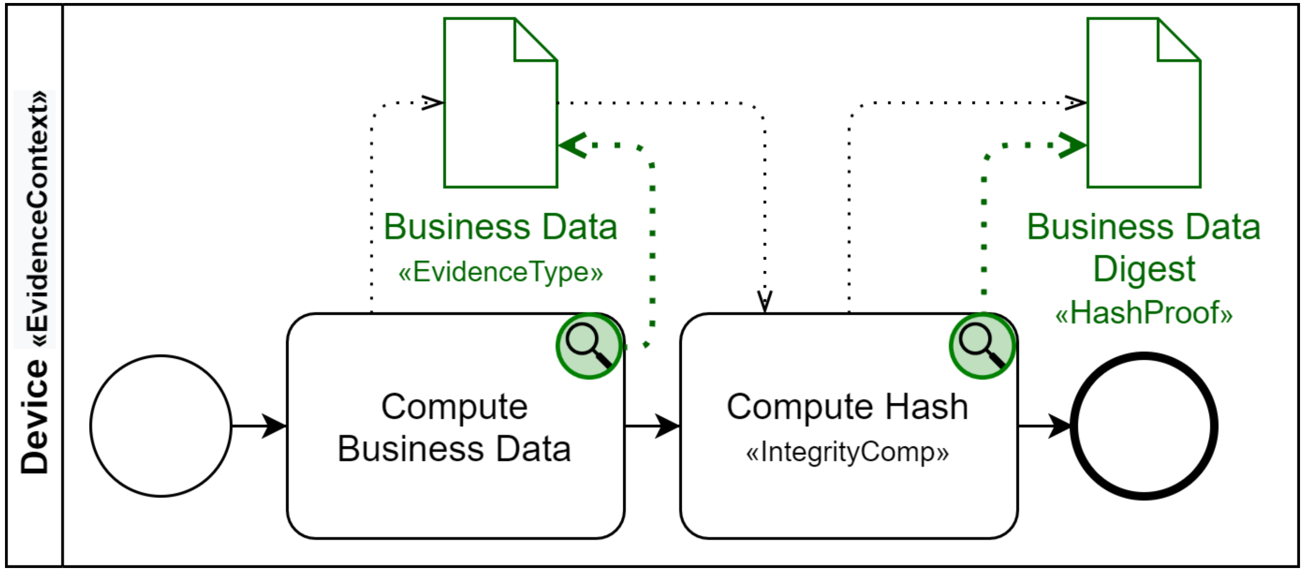
- Potential evidence
  - Where it originates?
  - Where is it stored?
  - In what context it is handled?



BPMN for Forensic-Ready Software Systems

Proof of potential digital evidence

- Strengthening the potential evidence
  - How is it obtained?
  - When is it created?
  - How it relates to the original?

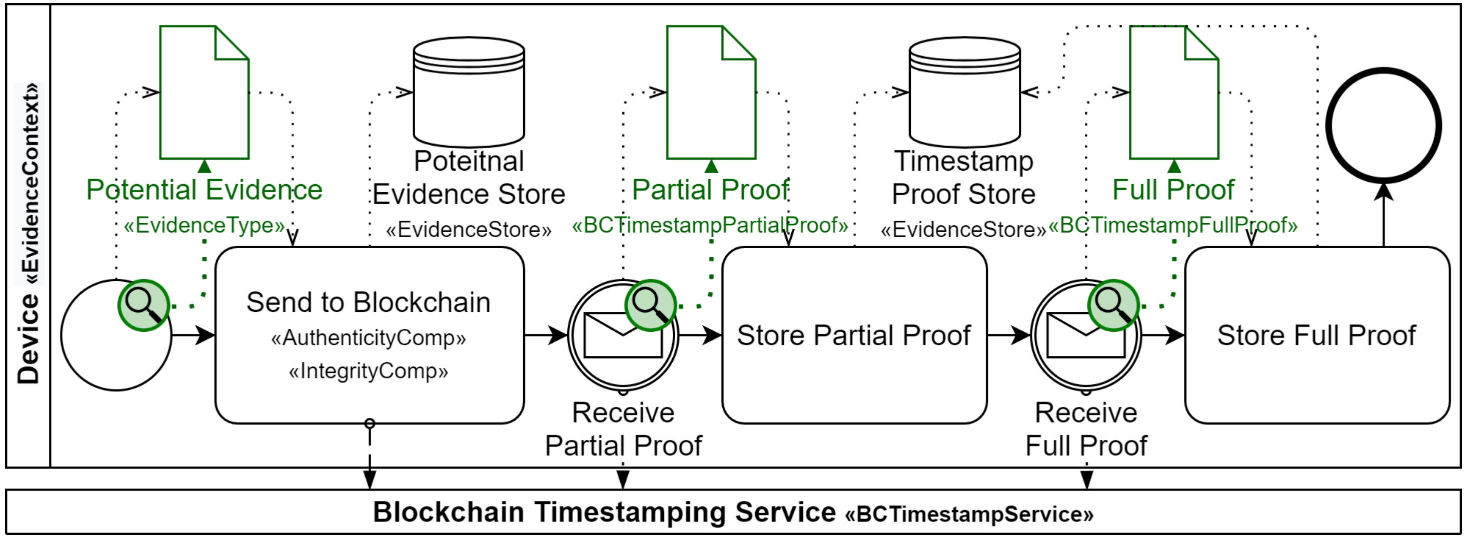


BPMN for Forensic-Ready Software Systems

Proof service

- Strengthening the potential evidence using an external service
  - Possibly 3rd party
  - What type of service?
  - How it creates the proof?

- Offloading the proof



BPMN for Forensic-Ready Software Systems

Scenario View & Evidence View

- But where are the relationships?

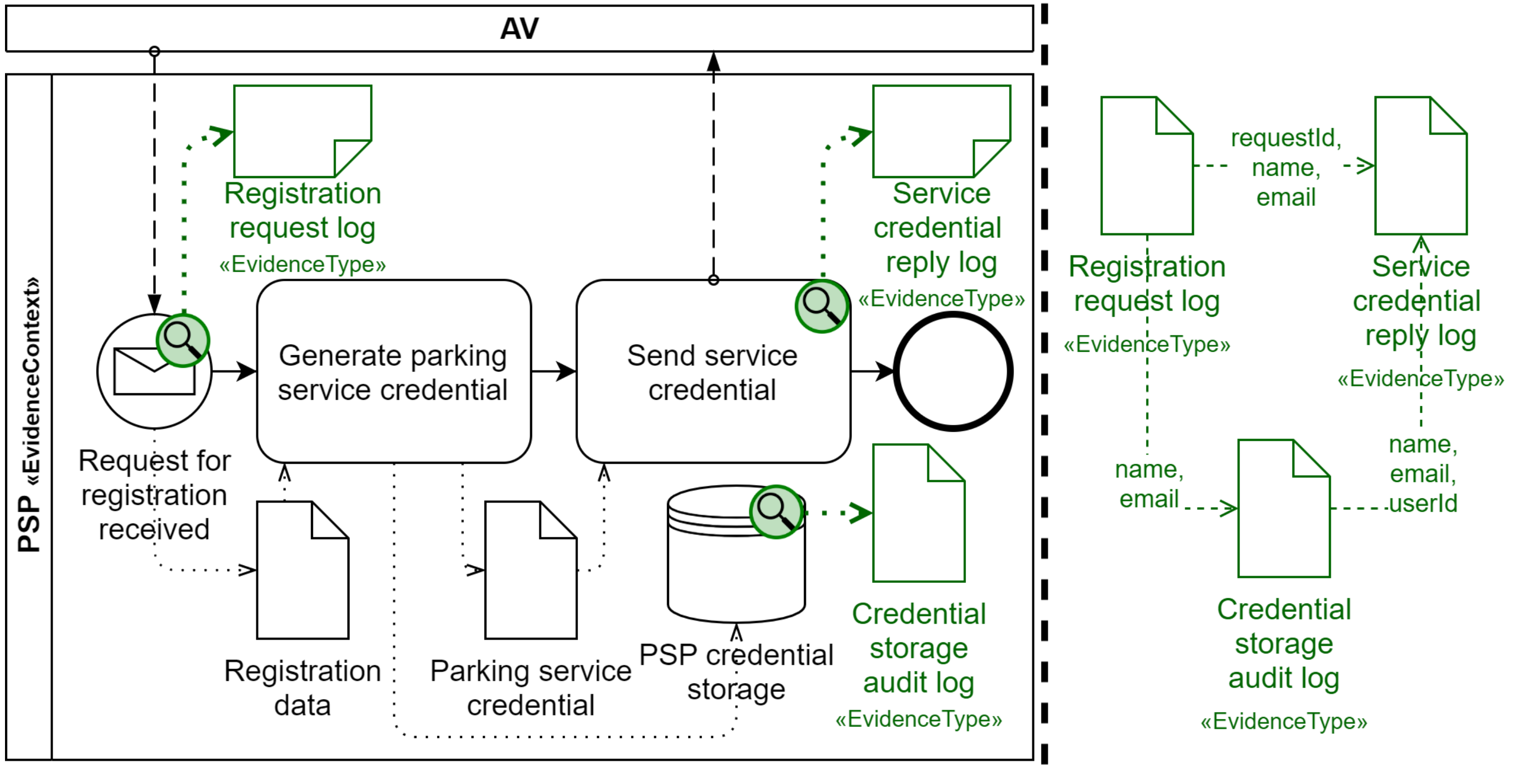
- Different needs

  =

  Different views

- One model,

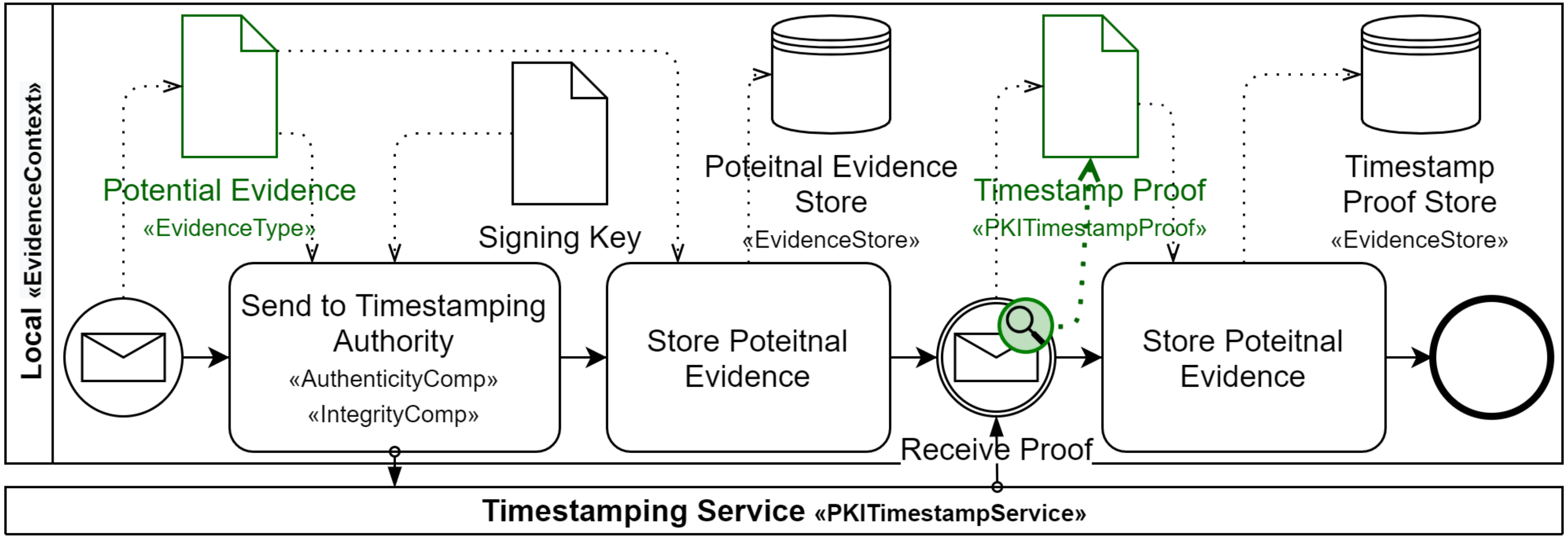
  two diagrams



BPMN for Forensic-Ready Software Systems

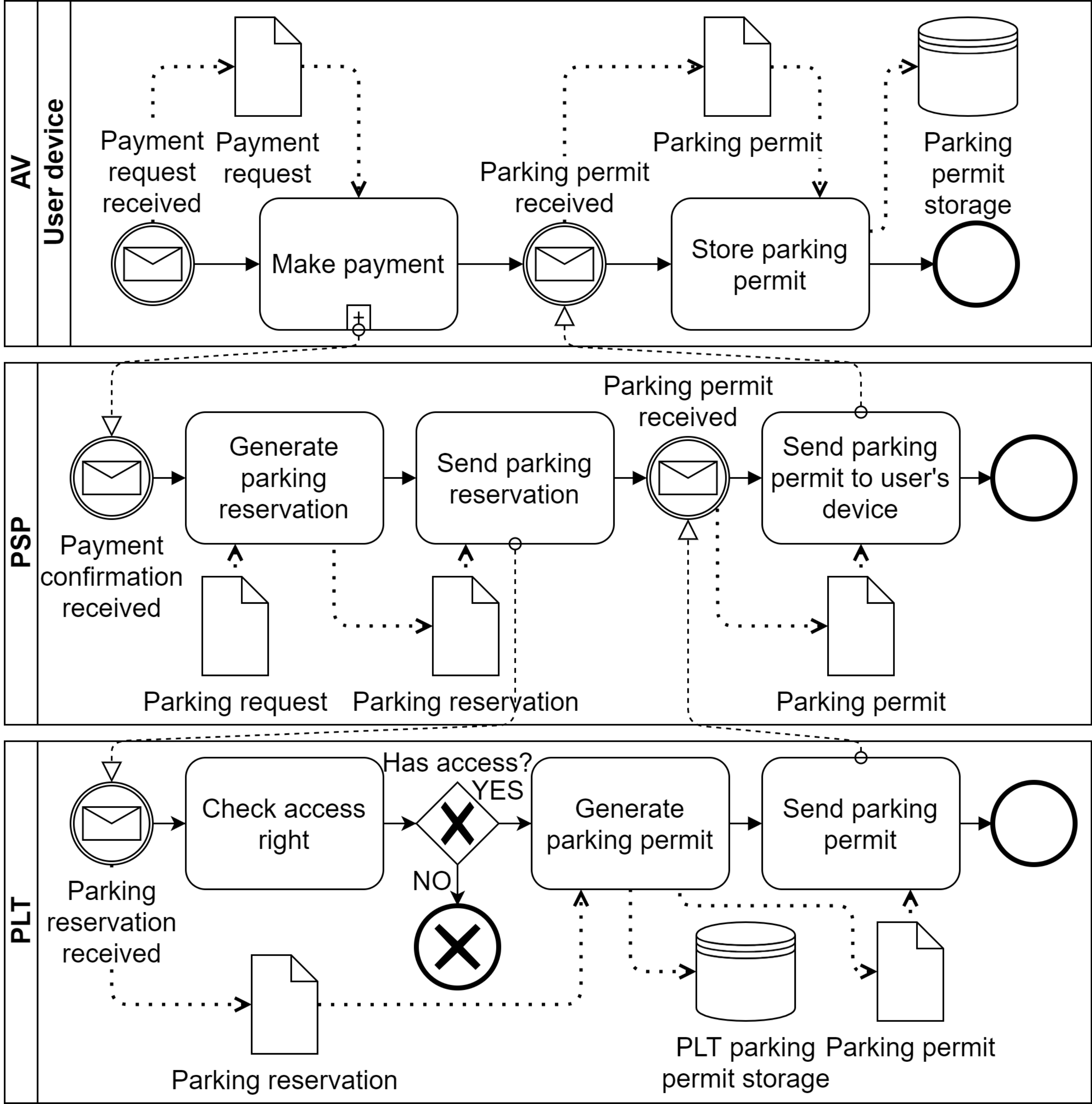
Lifecycle process

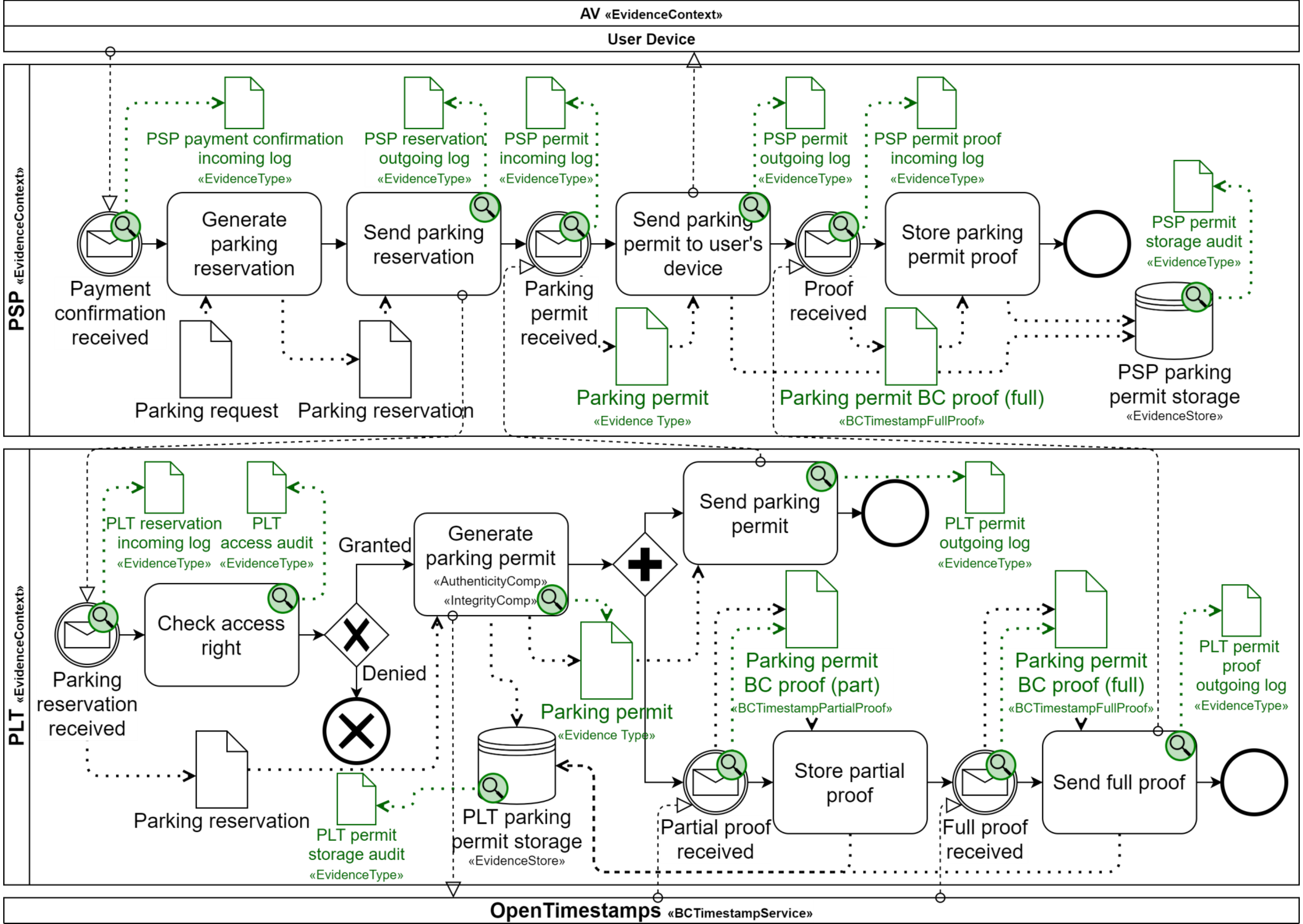
- Reusable model for evidence lifecycle
  - Reduce the clutter



Example scenario

- Autonomous parking
  - Payment
  - Generation of parking permit
- Risks:
  - Parking permit injection
  - Tampered access control
  - Parking permit repudiation
  - Zero-day attacks





AV «EvidenceContext»

User Device

PSP «EvidenceContext»

Payment confirmation received

Generate parking reservation

Send parking reservation

Parking permit received

Send parking permit to user's device

Proof received

Store parking permit proof

PSP payment confirmation incoming log «EvidenceType»

PSP reservation outgoing log «EvidenceType»

PSP permit incoming log «EvidenceType»

PSP permit outgoing log «EvidenceType»

PSP permit proof incoming log «EvidenceType»

PSP permit storage audit «EvidenceType»

Parking request

Parking reservation

Parking permit «Evidence Type»

Parking permit BC proof (full) «BCTimestampFullProof»

PSP parking permit storage «EvidenceStore»

PLT «EvidenceContext»

Parking reservation received

Check access right

Granted

Denied

Generate parking permit «AuthenticityComp» «IntegrityComp»

Send parking permit

PLT reservation incoming log «EvidenceType»

PLT access audit «EvidenceType»

PLT permit outgoing log «EvidenceType»

Parking reservation

PLT permit storage audit «EvidenceType»

PLT parking permit storage «EvidenceStore»

Parking permit «Evidence Type»

Parking permit BC proof (part) «BCTimestampPartialProof»

Parking permit BC proof (full) «BCTimestampFullProof»

PLT permit proof outgoing log «EvidenceType»

Partial proof received

Store partial proof

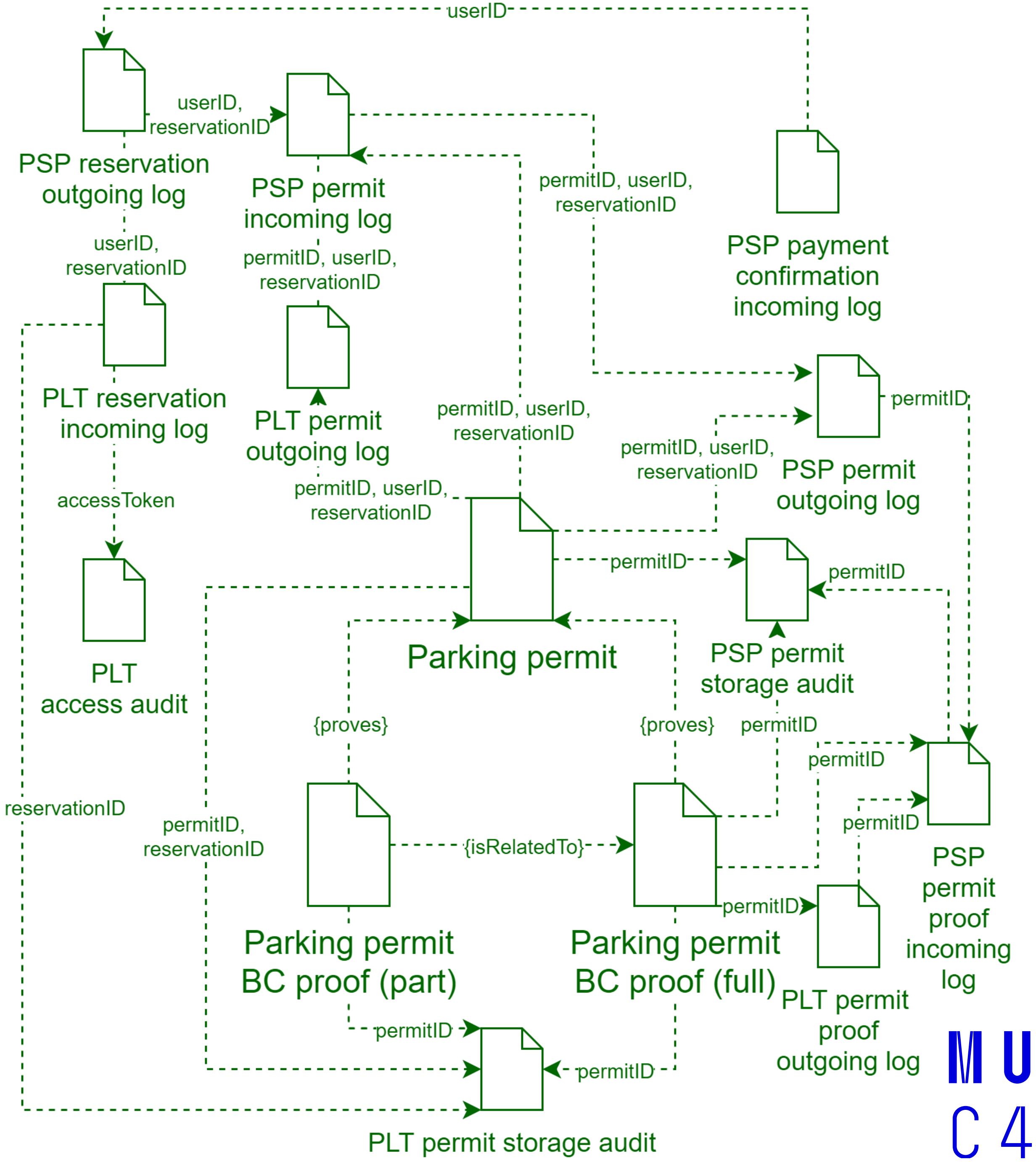
Full proof received

Send full proof

OpenTimestamps «BCTimestampService»

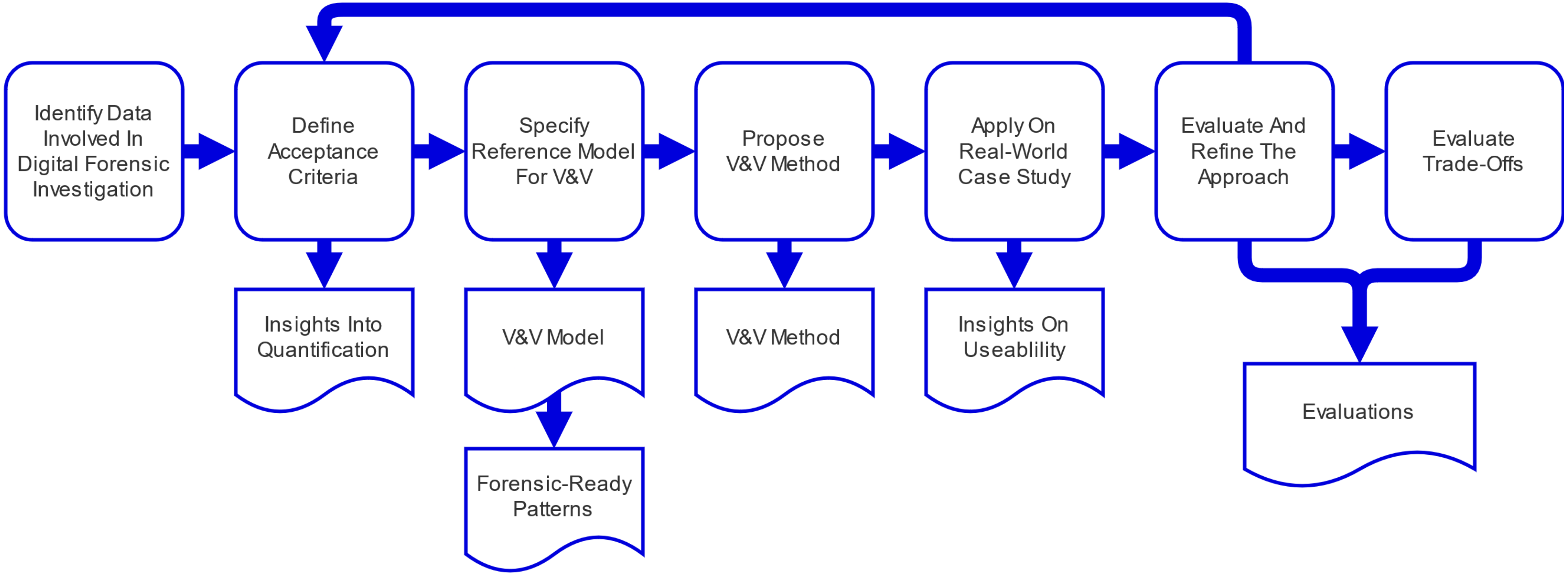
Example scenario

- Evidence View
- Relationships
  - Common data fields
  - Timing
  - Strengthening
- Nominal execution will contain everything
  - Attack should not


userID
PSP reservation outgoing log
userID, reservationID
PSP permit incoming log
permitID, userID, reservationID
PSP payment confirmation incoming log
userID, reservationID
permitID, userID, reservationID
PLT reservation incoming log
PLT permit outgoing log
permitID, userID, reservationID
permitID
accessToken
permitID, userID, reservationID
permitID, userID, reservationID
PSP permit outgoing log
PLT access audit
permitID
permitID
Parking permit
PSP permit storage audit
reservationID
permitID, reservationID
{proves}
{proves}
permitID
permitID
permitID
{isRelatedTo}
permitID
Parking permit BC proof (part)
Parking permit BC proof (full)
PSP permit proof incoming log
PLT permit proof outgoing log
permitID
permitID
PLT permit storage audit

Looking Forward

- Having model is just one step…


Identify Data Involved In Digital Forensic Investigation
Define Acceptance Criteria
Specify Reference Model For V&V
Propose V&V Method
Apply On Real-World Case Study
Evaluate And Refine The Approach
Evaluate Trade-Offs
Insights Into Quantification
V&V Model
V&V Method
Insights On Useablility
Evaluations
Forensic-Ready Patterns

Looking Forward

- Automated analysis based on the models
- Model validation
- Hint analysis
- Attack scenario analysis

- Evidence Generation Analysis
- Dispute Analysis

Conclusion

- Forensic readiness is an enhancement to security
  - Security risk management can be extended for this purpose
- BPMN for Forensic-Ready Software Systems
  - Representation of risk scenario and forensic-ready controls
- Scenario and Evidence view of the same model
- Foundation for validation and verification methods