Validating data integrity with blockchain

By Rosco Kalis & Adam Belloum

Contents

- Method overview
- Proof of concept audit trail
- Further use case: data provenance
- Limitations & Improvements

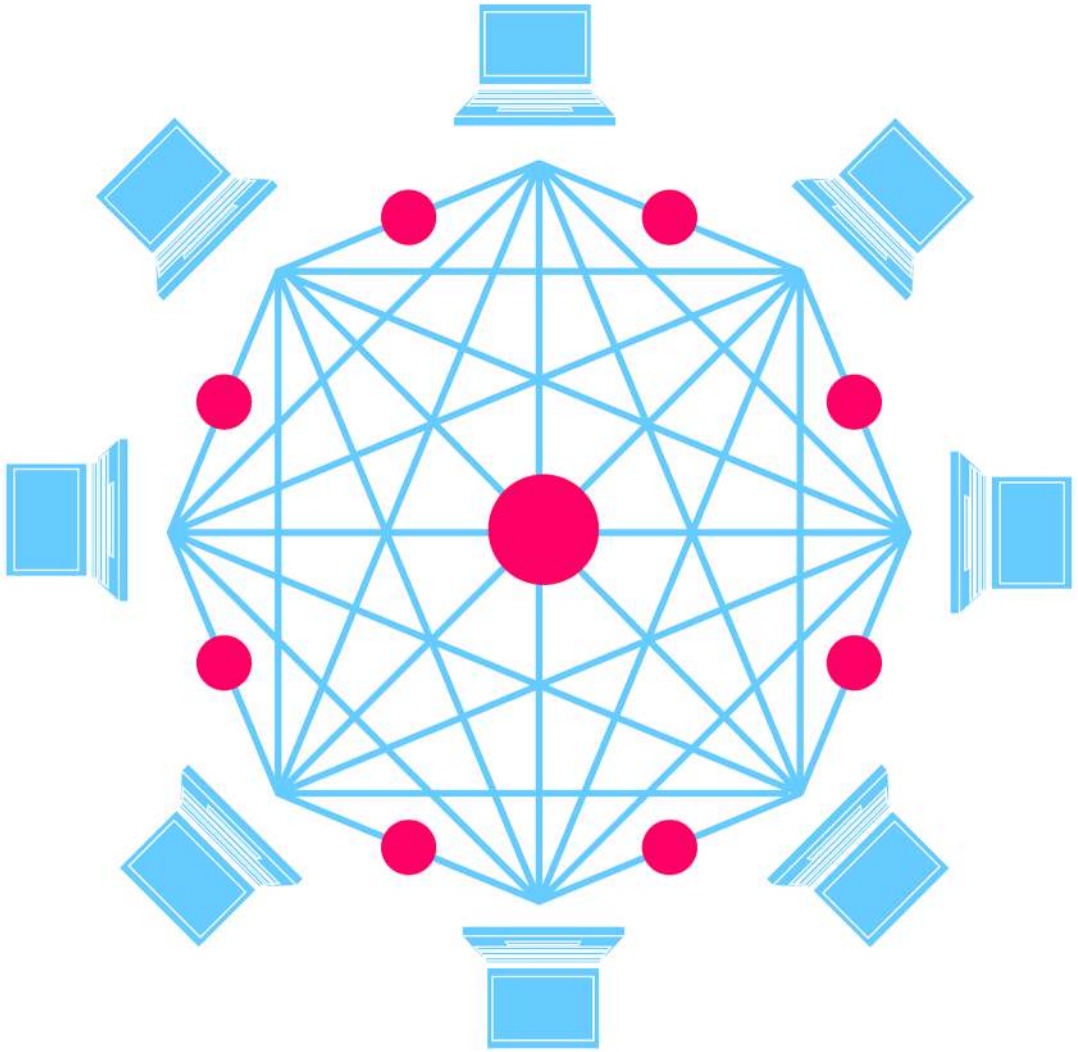
Method overview

Goal

- Validate the integrity of data
- Detect malicious data tampering

Smart contracts

- Code on the blockchain
- Data storage with persistent state



Smart contract storage

- Transaction Limits & Costs
  - Max ~11kB stored
  - €0.7 – €8.4 / kB stored
- Data confidentiality
  - Data encryption
  - Data hashing

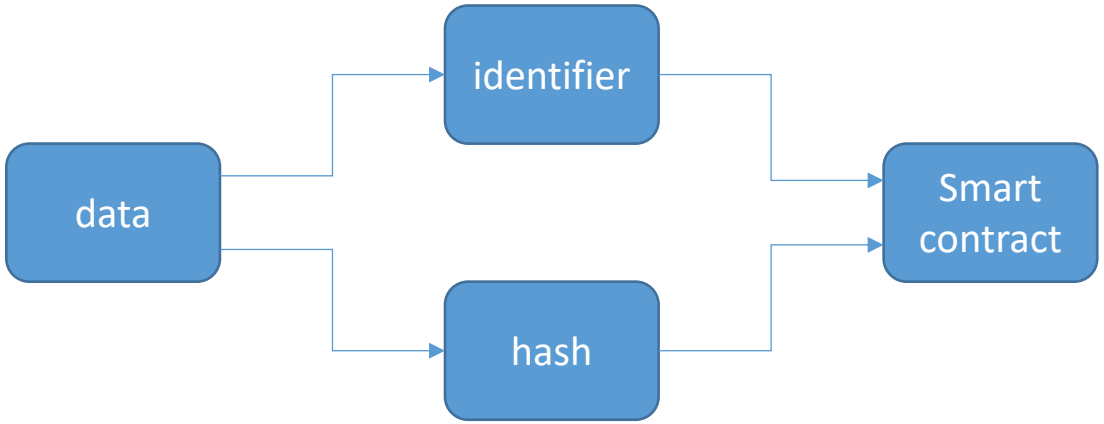
Appendix G. Fee Schedule

The fee schedule G is a tuple of 31 scalar values corresponding to the relative costs, in gas, of a number of abstract operations that a transaction may effect.

Name	Value	Description*
G_{zero}	0	Nothing paid for operations of the set W_{zero}.
G_{base}	2	Amount of gas to pay for operations of the set W_{base}.
$G_{verylow}$	3	Amount of gas to pay for operations of the set $W_{verylow}$.
G_{low}	5	Amount of gas to pay for operations of the set W_{low}.
G_{mid}	8	Amount of gas to pay for operations of the set W_{mid}.
G_{high}	10	Amount of gas to pay for operations of the set W_{high}.
$G_{extcode}$	700	Amount of gas to pay for operations of the set $W_{extcode}$.
$G_{balance}$	400	Amount of gas to pay for a BALANCE operation.
G_{sload}	200	Paid for a SLOAD operation.
$G_{jumpdest}$	1	Paid for a JUMPDEST operation.
G_{sset}	20000	Paid for an SSTORE operation when the storage value is set to non-zero from zero.
G_{sreset}	5000	Paid for an SSTORE operation when the storage value's zeroness remains unchanged or is set to zero.
R_{sclear}	15000	Refund given (added into refund counter) when the storage value is set to zero from non-zero.
$R_{suicide}$	24000	Refund given (added into refund counter) for suiciding an account.
$G_{suicide}$	5000	Amount of gas to pay for a SUICIDE operation.
G_{create}	32000	Paid for a CREATE operation.
$G_{codedeposit}$	200	Paid per byte for a CREATE operation to succeed in placing code into state.
G_{call}	700	Paid for a CALL operation.
$G_{callvalue}$	9000	Paid for a non-zero value transfer as part of the CALL operation.
$G_{callstipend}$	2300	A stipend for the called contract subtracted from $G_{callvalue}$ for a non-zero value transfer.
$G_{newaccount}$	25000	Paid for a CALL or SUICIDE operation which creates an account.
G_{exp}	10	Partial payment for an EXP operation.
$G_{expbyte}$	10	Partial payment when multiplied by $\lceil \log_{256}(exponent) \rceil$ for the EXP operation.
G_{memory}	3	Paid for every additional word when expanding memory.
$G_{txcreate}$	32000	Paid by all contract-creating transactions after the *Homestead transition*.
$G_{txdatazero}$	4	Paid for every zero byte of data or code for a transaction.
$G_{txdatanonzero}$	68	Paid for every non-zero byte of data or code for a transaction.
$G_{transaction}$	21000	Paid for every transaction.
G_{log}	375	Partial payment for a LOG operation.
$G_{logdata}$	8	Paid for each byte in a LOG operation's data.
$G_{logtopic}$	375	Paid for each topic of a LOG operation.
G_{sha3}	30	Paid for each SHA3 operation.
$G_{sha3word}$	6	Paid for each word (rounded up) for input data to a SHA3 operation.
G_{copy}	3	Partial payment for *COPY operations, multiplied by words copied, rounded up.
$G_{blockhash}$	20	Payment for BLOCKHASH operation.

Method

- Generate deterministic data identifier
- Generate data hash
- Store using a mapping in a smart contract



mapping(bytes32 => bytes32)

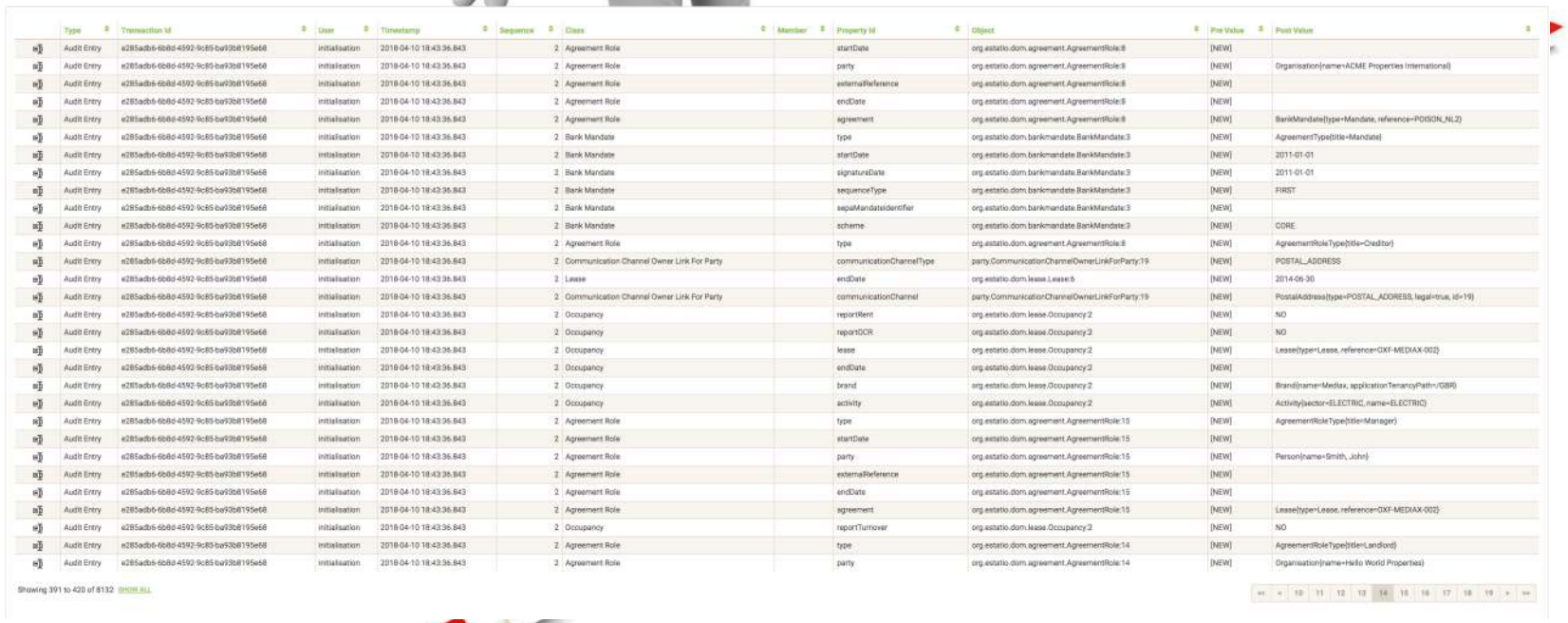
Proof of Concept

Audit trail

- Logging all application interactions
- A means to validate data integrity



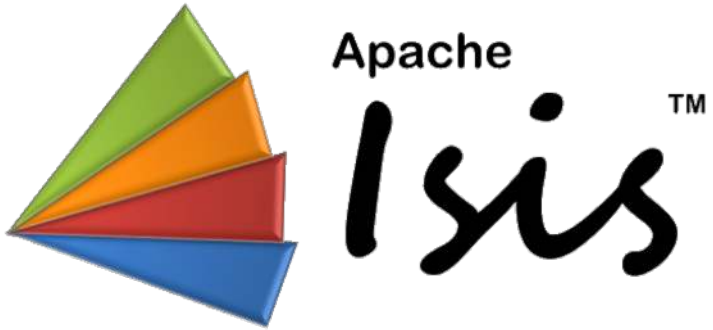
- But: Regular audit trails can still be tampered with
- This is where our method comes in



	Type	Transaction Id	User	Timestamp	Sequence	Class	Member	Property Id	Object	Pre Value	Post Value
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		startDate	org.estatio.dom.agreement.AgreementRole:8	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		party	org.estatio.dom.agreement.AgreementRole:8	[NEW]	Organisation{name=ACME Properties International}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		externalReference	org.estatio.dom.agreement.AgreementRole:8	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		endDate	org.estatio.dom.agreement.AgreementRole:8	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		agreement	org.estatio.dom.agreement.AgreementRole:8	[NEW]	BankMandate{type=Mandate, reference=POISON_NL2}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Bank Mandate		type	org.estatio.dom.bankmandate.BankMandate:3	[NEW]	AgreementType{title=Mandate}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Bank Mandate		startDate	org.estatio.dom.bankmandate.BankMandate:3	[NEW]	2011-01-01
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Bank Mandate		signatureDate	org.estatio.dom.bankmandate.BankMandate:3	[NEW]	2011-01-01
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Bank Mandate		sequenceType	org.estatio.dom.bankmandate.BankMandate:3	[NEW]	FIRST
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Bank Mandate		sepaMandateIdentifier	org.estatio.dom.bankmandate.BankMandate:3	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Bank Mandate		scheme	org.estatio.dom.bankmandate.BankMandate:3	[NEW]	CORE
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		type	org.estatio.dom.agreement.AgreementRole:8	[NEW]	AgreementRoleType{title=Creditor}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Communication Channel Owner Link For Party		communicationChannelType	party.CommunicationChannelOwnerLinkForParty:19	[NEW]	POSTAL_ADDRESS
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	1	Lease		endDate	org.estatio.dom.lease.Lease:6	[NEW]	2014-06-30
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Communication Channel Owner Link For Party		communicationChannel	party.CommunicationChannelOwnerLinkForParty:19	[NEW]	PostalAddress{type=POSTAL_ADDRESS, legal=true, id=19}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		reportRent	org.estatio.dom.lease.Occupancy:2	[NEW]	NO
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		reportOCR	org.estatio.dom.lease.Occupancy:2	[NEW]	NO
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		lease	org.estatio.dom.lease.Occupancy:2	[NEW]	Lease{type=Lease, reference=OXF-MEDIAX-002}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		endDate	org.estatio.dom.lease.Occupancy:2	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		brand	org.estatio.dom.lease.Occupancy:2	[NEW]	Brand{name=Mediax, applicationTenancyPath=/GBR}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		activity	org.estatio.dom.lease.Occupancy:2	[NEW]	Activity{sector=ELECTRIC, name=ELECTRIC}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		type	org.estatio.dom.agreement.AgreementRole:15	[NEW]	AgreementRoleType{title=Manager}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		startDate	org.estatio.dom.agreement.AgreementRole:15	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		party	org.estatio.dom.agreement.AgreementRole:15	[NEW]	Person{name=Smith, John}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		externalReference	org.estatio.dom.agreement.AgreementRole:15	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		endDate	org.estatio.dom.agreement.AgreementRole:15	[NEW]	
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		agreement	org.estatio.dom.agreement.AgreementRole:15	[NEW]	Lease{type=Lease, reference=OXF-MEDIAX-002}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Occupancy		reportTurnover	org.estatio.dom.lease.Occupancy:2	[NEW]	NO
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		type	org.estatio.dom.agreement.AgreementRole:14	[NEW]	AgreementRoleType{title=Landlord}
	Audit Entry	e285adb6-6b8d-4592-9c85-ba93b8195e68	initialisation	2018-04-10 18:43:36.843	2	Agreement Role		party	org.estatio.dom.agreement.AgreementRole:14	[NEW]	Organisation{name=Hello World Properties}

Showing 391 to 420 of 8132 SHOW ALL

« 10 11 12 13 14 15 16 17 18 19 »



- Framework for Rapid Domain Driven Development
- UI generated from domain model

```java
@DomainObject
public class SimpleObject implements Comparable<SimpleObject> {

    public static final int NAME_LENGTH = 40;

    public TranslatableString title() { return TranslatableString.tr("Object: {name}", "name", getName()); }

    public static class NameDomainEvent extends PropertyDomainEvent<SimpleObject,String> {}
    @javax.jdo.annotations.Column(
            allowsNull="false",
            length = NAME_LENGTH
    )
    @Property(
        domainEvent = NameDomainEvent.class
    )
    private String name;
    public String getName() { return name; }
    public void setName(final String name) { this.name = name; }

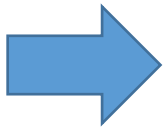
    public TranslatableString validateName(final String name) {
        return name != null && name.contains("!")? TranslatableString.tr("Exclamation mark is not allowed"): null;
    }

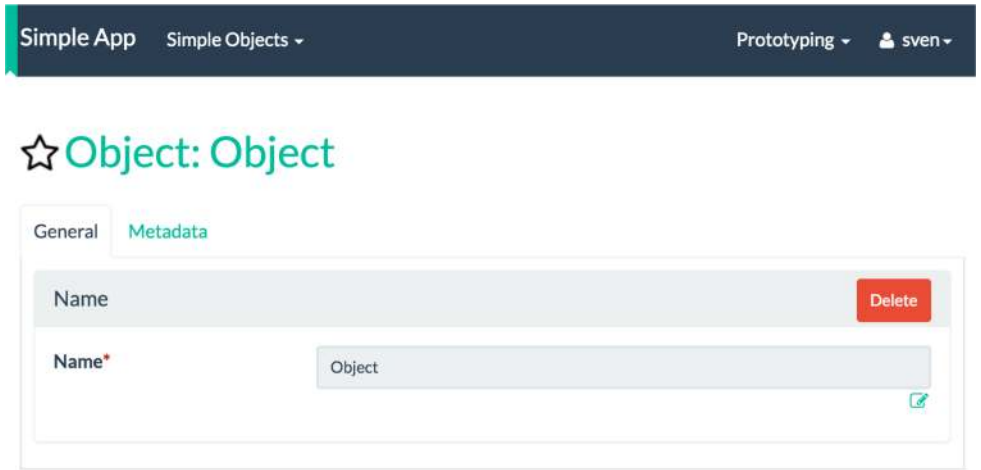
    public static class DeleteDomainEvent extends ActionDomainEvent<SimpleObject> {}
    @Action(
            domainEvent = DeleteDomainEvent.class,
            semantics = SemanticsOf.NON_IDEMPOTENT_ARE_YOU_SURE
    )
    public void delete() { repositoryService.remove(this); }



    @Override
    public int compareTo(final SimpleObject other) { return ObjectContracts.compare(this, other, "name"); }


    @javax.inject.Inject
    RepositoryService repositoryService;
}
```

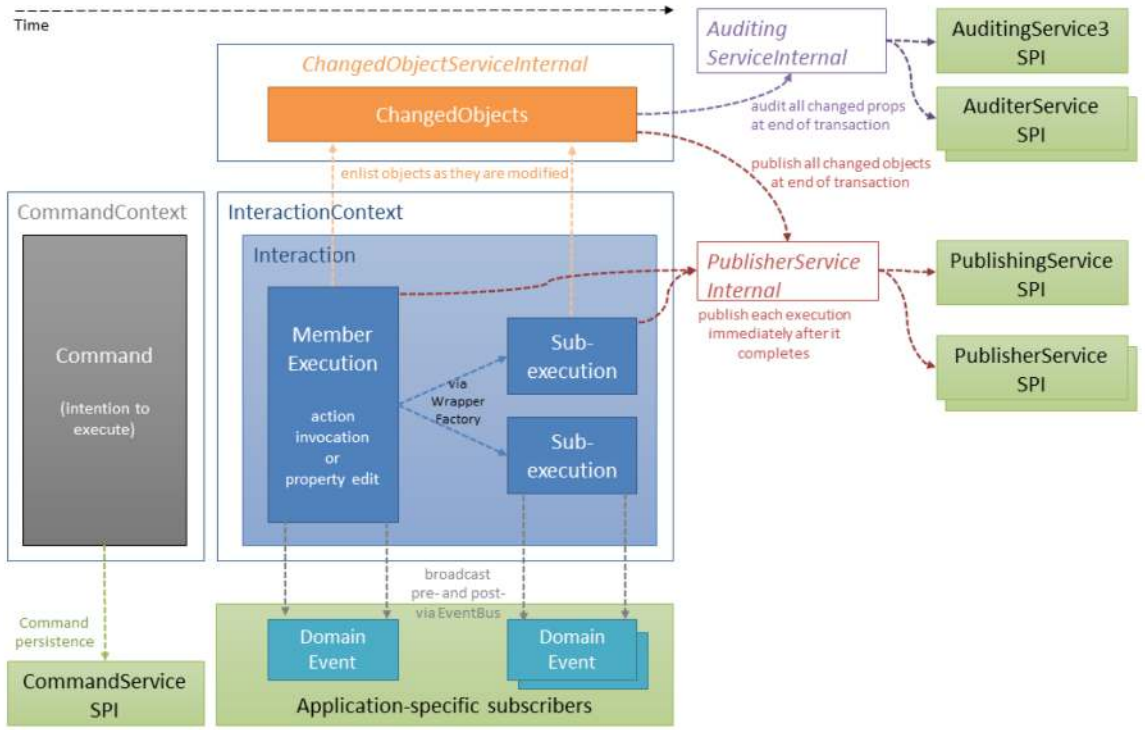




AuditerService

- Called once for every change
- Limit number of blockchain transactions
- Aggregate changes by transactionId + sequence
  - ThreadLocal AuditEntry
  - PublisherService

```
public interface AuditerService {
    boolean isEnabled();
    public void audit(
            UUID transactionId, int sequence,
            String targetClassName,
            Bookmark target,
            String memberIdentifier,
            String propertyName,
            String preValue, String postValue,
            String user, java.sql.Timestamp timestamp);
}
```



Smart contract code

- Mapping
  - Identifier to hash
- List of identifiers
- Audit method
- Validate method

```
pragma solidity 0.4.23;
contract AuditTrail {
    ...
    bytes28[] public auditedTransactions;
    mapping(bytes28 => bytes32) public dataHashes;
    ...
    function audit(bytes28 transactionIdentifier, bytes32 dataHash) external ownerOnly {
        require(dataHashes[transactionIdentifier] == 0, "A transaction can only be audited once");
        dataHashes[transactionIdentifier] = dataHash;
        auditedTransactions.push(transactionIdentifier);
    }

    function validate(bytes28 transactionIdentifier, bytes32 dataHash) external view returns(uint8) {
        return dataHashes[transactionIdentifier] == dataHash ? 0 : 1;
    }
}
```

Blockchain integration

- Web3j
  - Ethereum JSON-RPC
  - Smart contract wrappers
- Asynchronous transactions

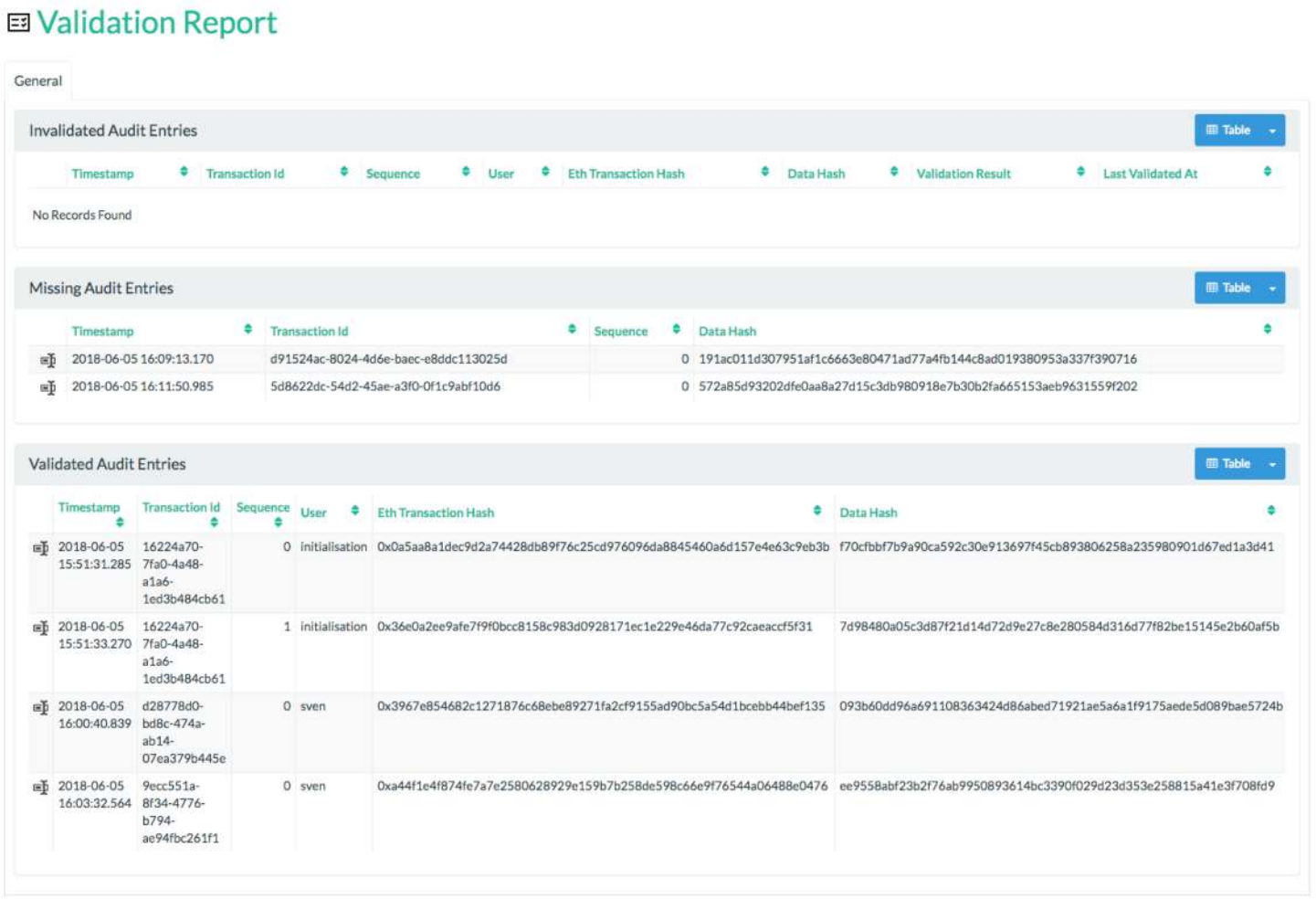
```
@Programmatic
public void commitAuditEntry() {
    AuditEntry auditEntry = currentAuditEntry.get();
    if (auditEntry == null) {
        return;
    }

    try {
        final Bookmark bookmark = bookmarkService2.bookmarkFor(auditEntry);
        web3Service.getAuditTrailContract()
                .audit(auditEntry.getIdentifier(), auditEntry.getHash())
                .sendAsync()
                .thenAccept(new TransactionReceiptConsumer(bookmark))
                .exceptionally(e -> {e.printStackTrace(); return null;})
                .thenRun(() -> System.out.println(currentAuditEntry.get().serialise()));
    } catch (Exception e) {
        e.printStackTrace();
    }
    currentAuditEntry.set(null);
}
```

```
@Override protected void doExecuteWithTransaction(final Object context) {
    TransactionReceipt receipt = (TransactionReceipt) context;
    AuditEntry auditEntry = bookmarkService2.lookup(bookmark, AuditEntry.class);
    auditEntry.setEthTransactionHash(receipt.getTransactionHash());
}
```

Audit trail validation

- Single Audit Entries
  - Smart contract validate-method
- Full audit trail
  - Validating individual Audit Entries
  - Iterating list of transaction identifiers



Validation Report

General

Invalidated Audit Entries — Table

Timestamp	Transaction Id	Sequence	User	Eth Transaction Hash	Data Hash	Validation Result	Last Validated At
No Records Found							

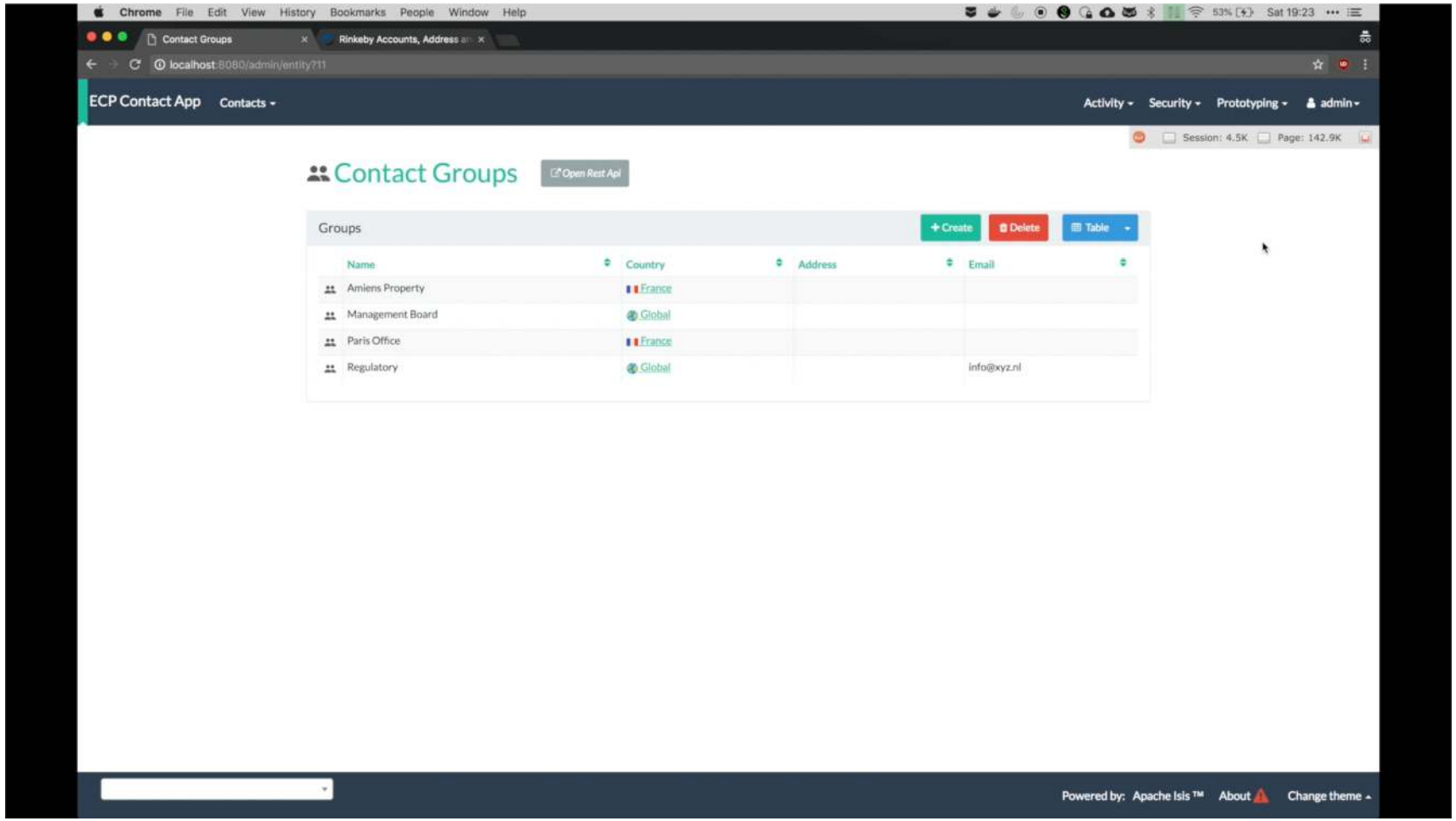
Missing Audit Entries — Table

Timestamp	Transaction Id	Sequence	Data Hash
2018-06-05 16:09:13.170	d91524ac-8024-4d6e-baec-e8ddc113025d	0	191ac011d307951af1c6663e80471ad77a4fb144c8ad019380953a337f390716
2018-06-05 16:11:50.985	5d8622dc-54d2-45ae-a3f0-0f1c9abf10d6	0	572a85d93202dfe0aa8a27d15c3db980918e7b30b2fa665153aeb9631559f202

Validated Audit Entries — Table

Timestamp	Transaction Id	Sequence	User	Eth Transaction Hash	Data Hash
2018-06-05 15:51:31.285	16224a70-7fa0-4a48-a1a6-1ed3b484cb61	0	initialisation	0x0a5aa8a1dec9d2a74428db89f76c25cd976096da8845460a6d157e4e63c9eb3b	f70cfbbf7b9a90ca592c30e913697f45cb893806258a235980901d67ed1a3d41
2018-06-05 15:51:33.270	16224a70-7fa0-4a48-a1a6-1ed3b484cb61	1	initialisation	0x36e0a2ee9afe7f9f0bcc8158c983d0928171ec1e229e46da77c92caeaccf5f31	7d98480a05c3d87f21d14d72d9e27c8e280584d316d77f82be15145e2b60af5b
2018-06-05 16:00:40.839	d28778d0-bd8c-474a-ab14-07ea379b445e	0	sven	0x3967e854682c1271876c68ebe89271fa2cf9155ad90bc5a54d1bcebb44bef135	093b60dd96a691108363424d86abed71921ae5a6a1f9175aede5d089bae5724b
2018-06-05 16:03:32.564	9ecc551a-8f34-4776-b794-ae94fbc261f1	0	sven	0xa44f1e4f874fe7a7e2580628929e159b7b258de598c66e9f76544a06488e0476	ee9558abf23b2f76ab9950893614bc3390f029d23d353e258815a41e3f708fd9

Proof of Concept demo



Further use case

Data provenance

- Scientific reproducibility
- Full research environment, input data, assumptions, etc.
- Guarding against fraudulent research

Validating data provenance integrity

- Publish provenance data at the same time as research
- Derive identifier from research title / DOI
- Submit identifier + data provenance hash
- Correct data provenance can always be verified

Limitations and Improvements

Limitations in the proof of concept

- Crashes / outages during the audit process
- Transaction ordering and failing transactions
- Malicious additions to the audit trail

Improvements to the method

- Permissions on smart contract
- Storing metadata in the contract
- Full data storage on IPFS or on-chain

```
struct MetaData {
    string user;
    uint256 timestamp;
    bytes32 hash;
}
```

Questions