Resource-Event-Agent-Unit

an ontology for sharing sensitive transaction data

Wim Laurier & Jesper Kiehn
Table of content

• Context
• Requirements
• Resource-Event-Agent-Unit
• Relationship to REA
  – Independent View
  – Dependent View
• Conclusion
Coopetition

CONTEXT
Coopetion

• Cooperation
  – Create & Add Value
  – Sharing Information & Data

• Competition
  – Divide (Share) Added Value
  – Hiding Information & Data

• How to reconcile these conflicting objectives?
REQUIREMENTS
Requirements

• Share collaborative data
  – Share Information on Exchanges

• Protect competitive advantage & privacy
  – Protect Business Process Information
  – Protect Cost/Accounting Information
  – Protect Product Composition
  – Protect Identity of natural persons involved
Solution:
Independent and Dependent View

Collaboration Space

ISO/IEC FDIS 15944-4: 2007(E)
A unified account of REA

RESOURCE-EVENT-AGENT-UNIT
REAU Model

- «Agent Group» -> Unit
- Unit -> For
- Unit -> OnBehalfOf
- Resource -> InFlow -> Unit -> For
- Event -> Duality
- Event -> Participate -> Agent
- Resource -> OutFlow
- Event -> Participate
REAU Exchange Pattern
INDEPENDENT VIEW
Independent View in REAU
Independent View in REA
Resource-Event-Agent

DEPENDENT VIEW
REA Model

Resource  →  Event  ←  Agent

StockFlow  →  Duality  ←  Participate
Buyer Perspective in REAU
Buyer Perspective in REA

- **Acquisition Clerk**: Agent
  - Participate
- **Insider**: Participate
- **Outside**: Participate
- **Sale**: Event
  - Purchase
- **Duality**: Event
  - For
  - From
  - Disbursement
  - Outside Participate
  - To
- **Payment**: Event
  - Disbursement
- **OutFlow**: Money
  - Resource

**Product**: Resource

**Money**: Resource
Seller Perspective in REAU

- **Product**: Resource
- **Sale**: Event
- **Payment**: Event
- **Money**: Resource
- **Supply Clerck**: Agent
- **Invoice Clerck**: Agent
- **OutFlow**: shipment
- **InFlow**: receipt
- **Participate**: from, to, on behalf of
Seller Perspective in REA

Diagram:
- **Product**: Resource
  - **OutFlow**: shipment
  - **Inside Participate**: on behalf of
- **Sale**: Event
  - **Outside Participate**: Duality
- **Payment**: Event
  - **Outside Participate**: Duality
  - **InFlow**: receipt
- **Money**: Resource
- **Supply Clerk**: Agent
  - **Inside Participate**: on behalf of
- **Invoice Clerk**: Agent
  - **Inside Participate**: on behalf of

**Key Terms**:
- **Buyer**: Unit
- **Seller**: Unit

**Links**:
- From **Product** to **Sale**
- From **Sale** to **Payment**
- From **Payment** to **Money**
- From **Supply Clerk** to **Seller**
- From **Invoice Clerk** to **Seller**
Conversion in REAU
Conversion in REA
Naming conventions

CONCLUSION
Conclusion

• REAU shows what is implicit or omitted in
  – REA’s Independent View
  – REA’s Dependent View
• Enables an automated transformation between both views.
• Allows for Protecting/hiding information of strategic importance with Dependent View
• While Sharing collaborative information with Independent View
Future Work

• Phase 1: Build prototype in P# (Prolog for .net)
• Phase 2: Add Commitments
• Phase 3: Do cross-company planning
• Phase 4: Build Prototypes in different Programming Language (e.g., C#, Java).
• Phase 5: Test Prototypes in real world setting