UNIVERSITY OF TRENTO - Italy

DETECTING INCONSISTENCIES IN SECURITY REQUIREMENTS

Elda Paja, Fabiano Dalpiaz, Paolo Giorgini

Re-Seminar

March 22nd 2012

Socio-Technical Systems (STS)

- An interplay of humans, organisations, and technical systems
  - Founded upon the notion of social reliance
- Complex systems
  - Defined in terms of interaction among actors
  - Each participant is autonomous
- Examples: smart homes, e-commerce sites, …

The Security Problem

- Not just technical (encryption, access control, …)
- Social aspects are a main concern
  - Decentralised setting: no controlling authority
  - Autonomy: security cannot be enforced

Security Requirements via Commitments

STS-ml

- Take a service-oriented stance
  - Relate security requirements to interaction between actors (service consumer and provider)
  - Allow actors to express constraints (security needs) over interactions
    - E.g.: in e-commerce buyer wants seller to use its credit card information strictly to conclude the payment and not to disclose them to other parties

- Specify security requirements in terms of social commitments
  - Social commitments represent the constraints the actors shall comply with while interacting
    - E.g.: seller commits not to disclose buyer's credit card details to other parties

The Inconsistency Problem

- Security specifications guide the design of a STS that satisfies the security requirements

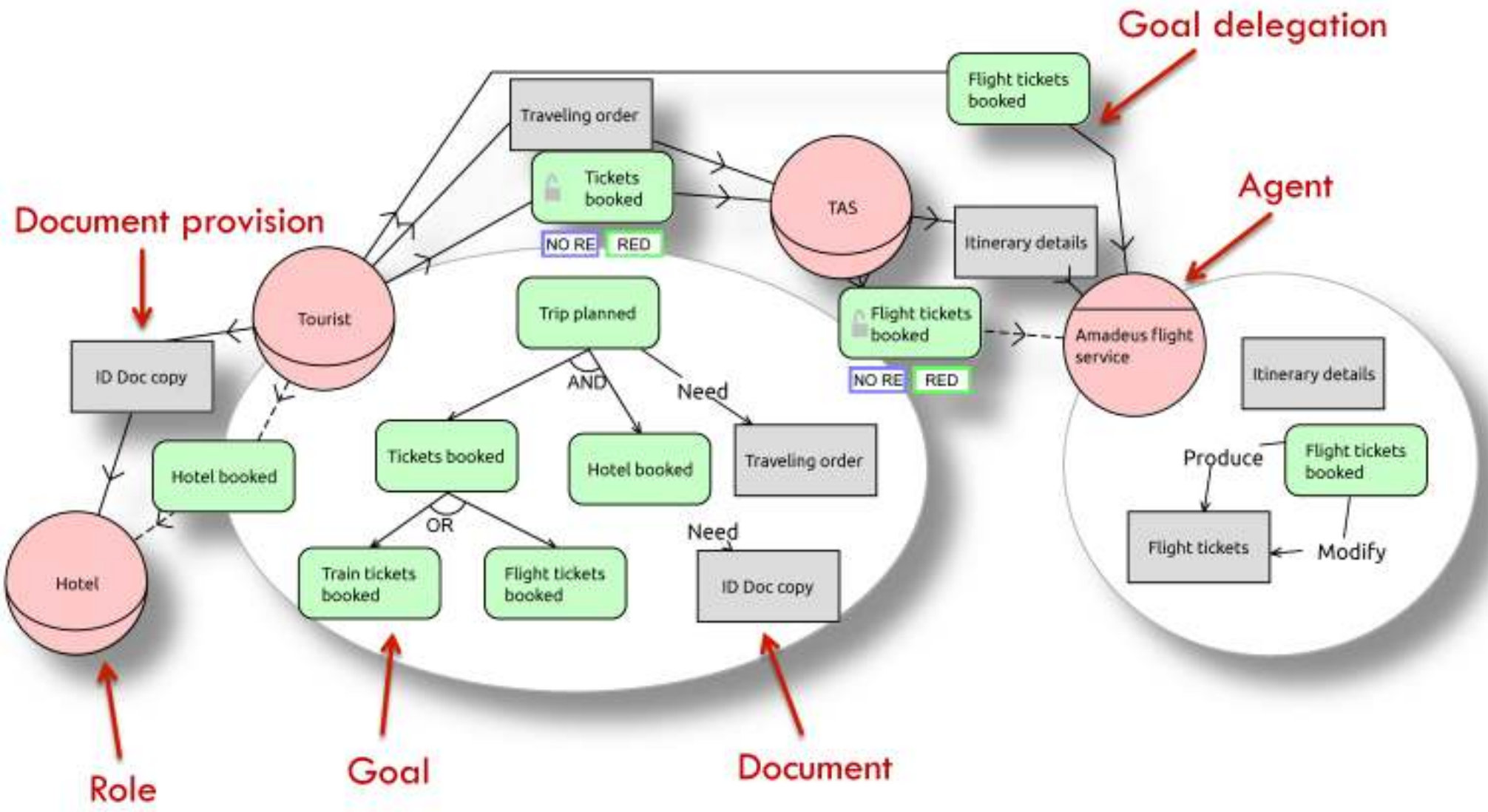
- Inconsistent security requirements have severe consequences
  - Implementation of a STS that will not satisfy at least one requirement
  - Violation of critical properties: confidentiality
    - Law infringement, monetary sanctions

- Key question: Is the specification consistent?

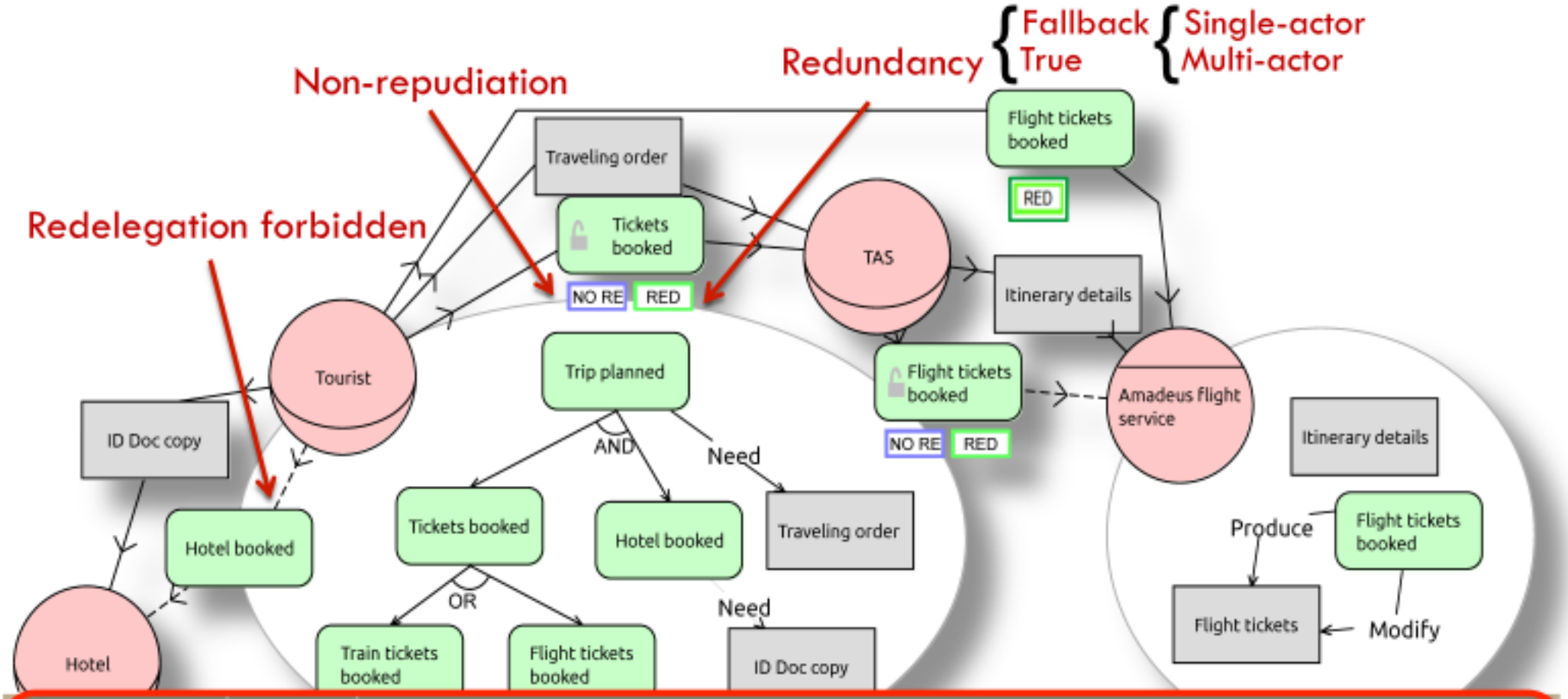
Formal Framework

- Focus on security requirements in a STS-ml specification
- A framework to detect inconsistencies
  - Inconsistencies not trivial to find
  - Scalability is an issue
- Formally Defined
  - Security needs supported by STS-ml
  - The derived security requirements (in terms of commitments)

STS-ml: Social View

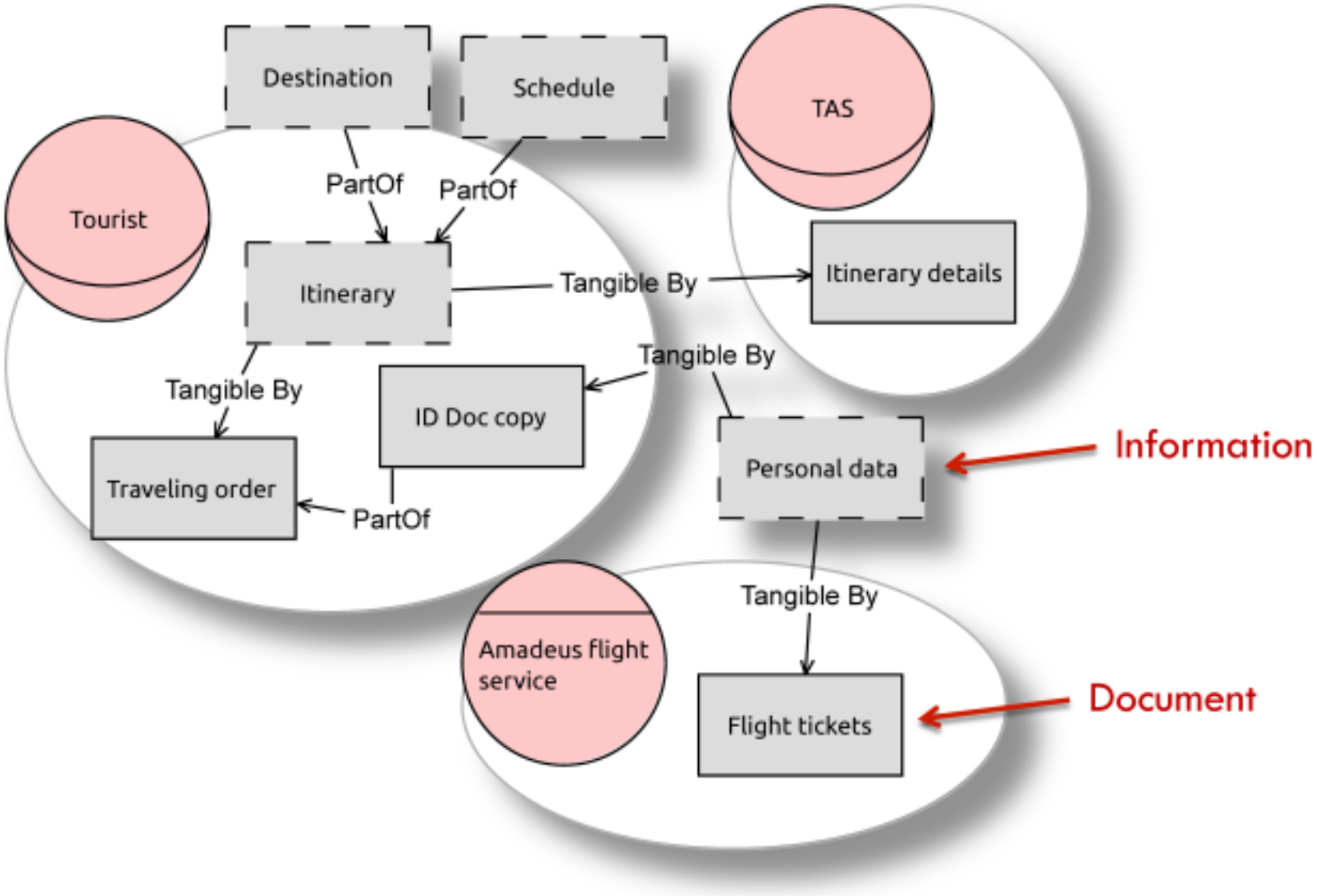


Social View: security needs

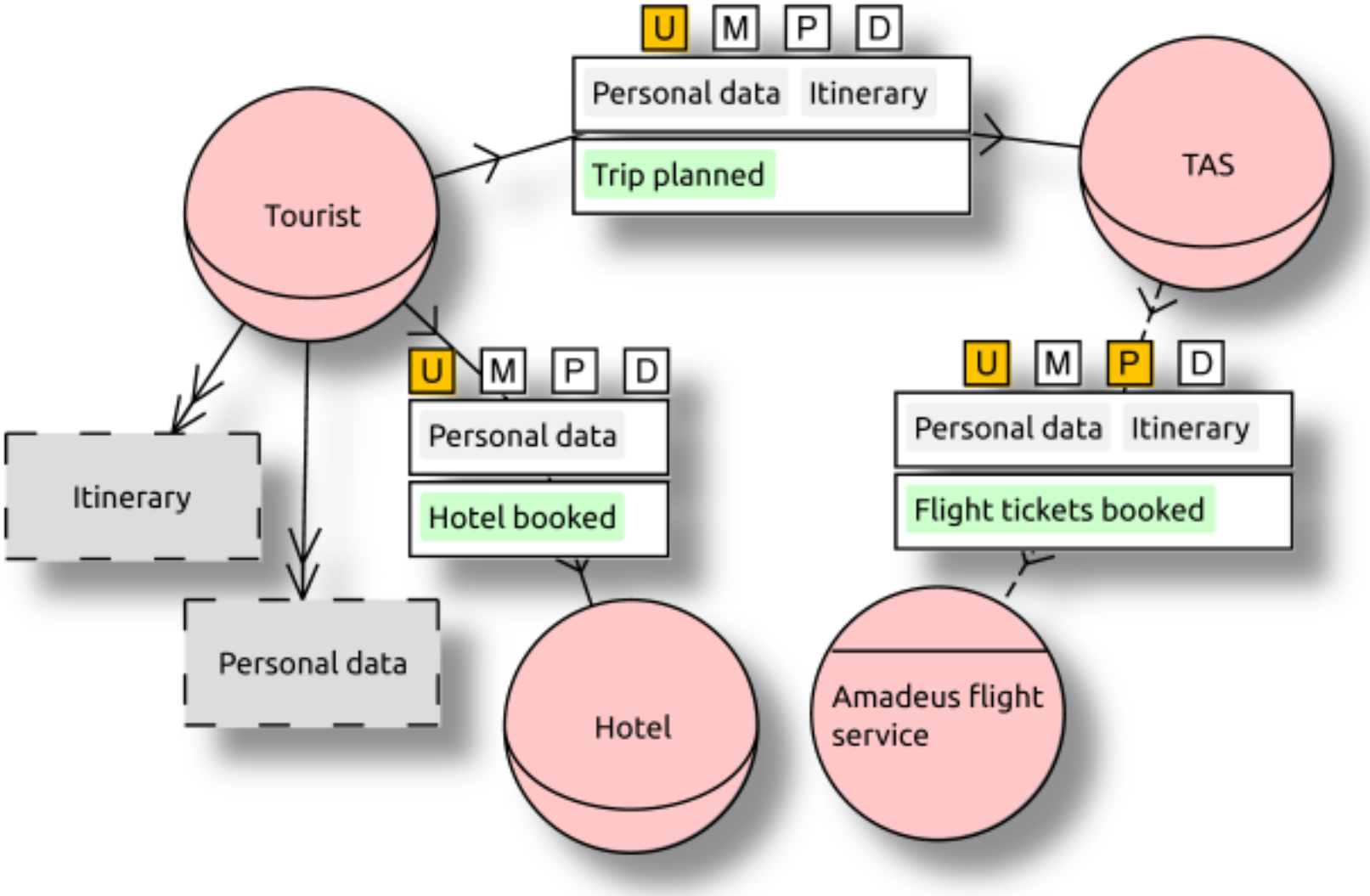


TAS	Tourist	d_1:=delegate(Tourist, TAS, Tickets booked), non repudiation(d_1)
Amadeus FS	Tourist	delegate(Tourist, Amadeus FS, Tickets booked), true_rm(Flight tickets booked)
Hotel	Tourist	delegate(Tourist, Hotel, Hotel booked), no-delegation(Hotel booked)

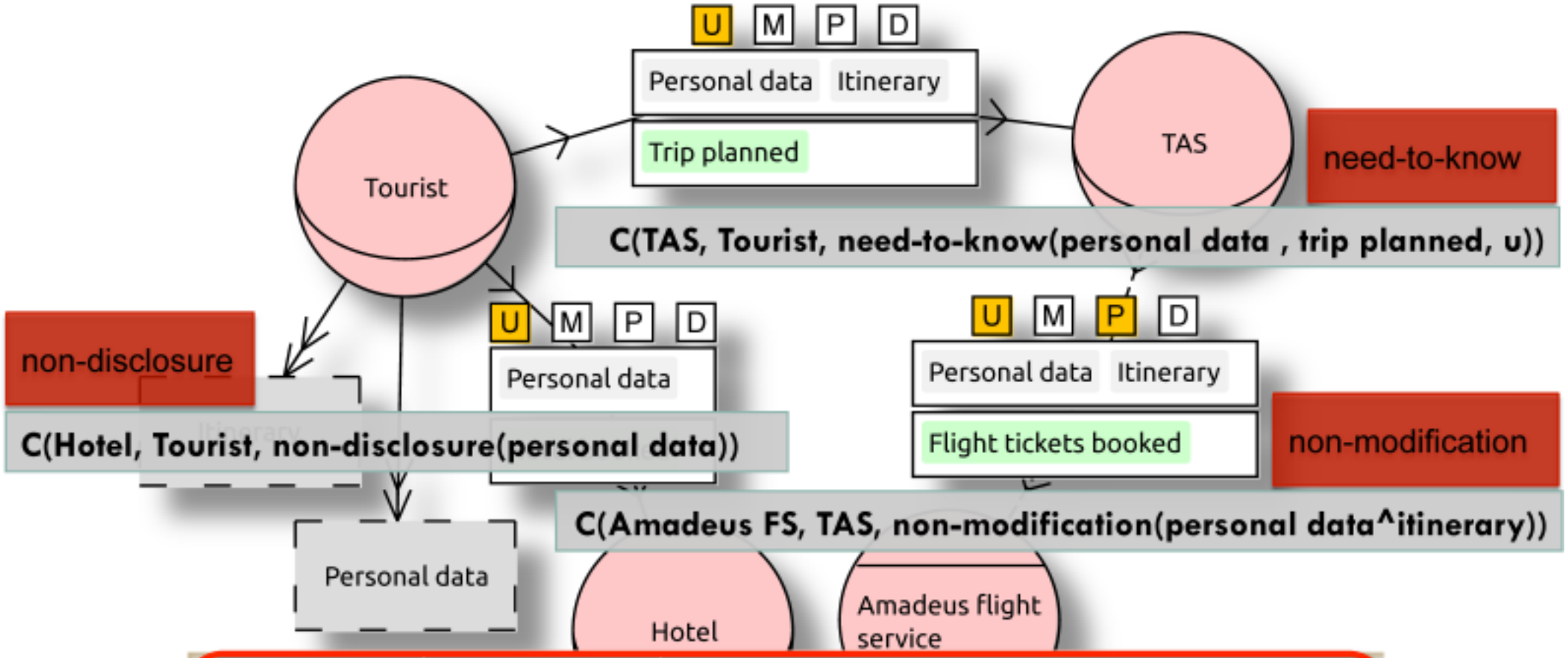
STS-ml: Information View



STS-ml: Authorisation View



STS-ml: Authorisation View



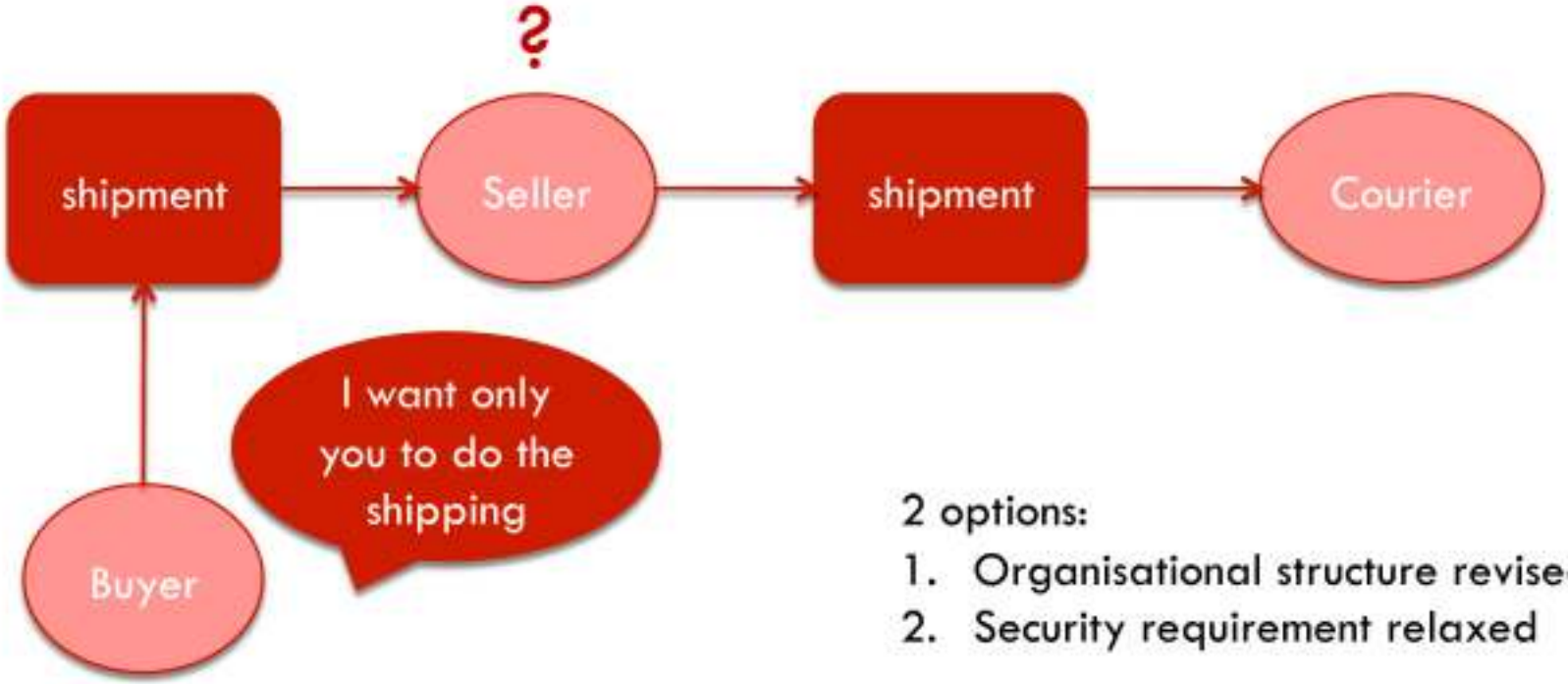
Hotel	Tourist	non-disclosure(personal data)
TAS	Tourist	need-to-know(personal data, trip planned, u)
Amadeus FS	TAS	non-modification(personal data^itinerary)

Security Specification

Debtor	Creditor	Security Requirement
TAS	Tourist	need-to-know(personal data , trip planned, u)
Hotel	Tourist	need-to-know(personal data, hotel booked, u)
Amadeus FS	TAS	need-to-know(personal data ^ itinerary, flight tickets booked, u ^ p)
TAS	Tourist	non-disclosure(personal data ^ itinerary)
Hotel	Tourist	non-disclosure(personal data)
Amadeus FS	TAS	non-disclosure(personal data ^ itinerary)
Hotel	Tourist	non-modification(personal data ^ itinerary)
TAS	Tourist	non-modification(personal data)
Amadeus FS	TAS	non-modification(personal data ^ itinerary)
TAS	Tourist	non-production(personal data ^ itinerary)
Hotel	Tourist	non-production(personal data)

Identifying Inconsistencies

- Two types of inconsistencies
  - Organizational requirements – Security requirements Inconsistencies
    - Security requirements cannot be satisfied in the modelled organisational structure

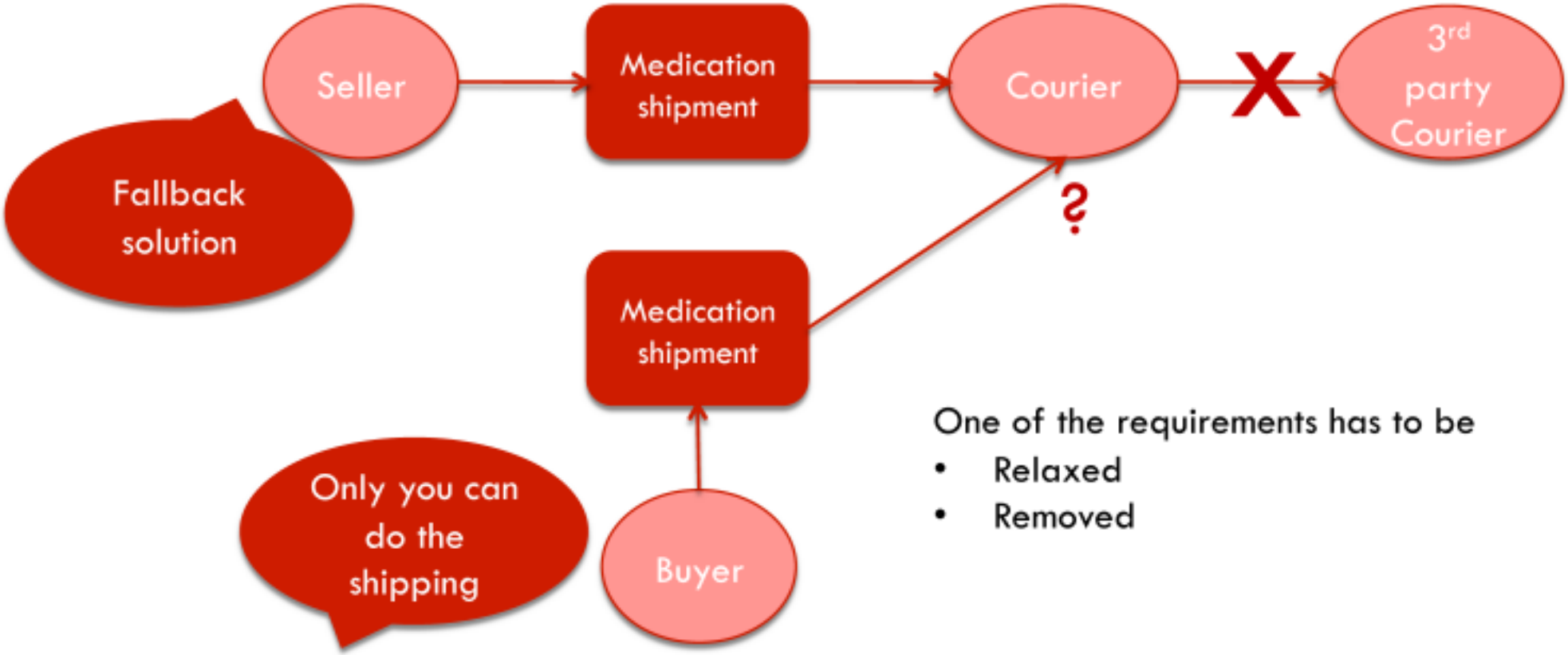


2 options:

1. Organisational structure revised
2. Security requirement relaxed

Identifying Inconsistencies

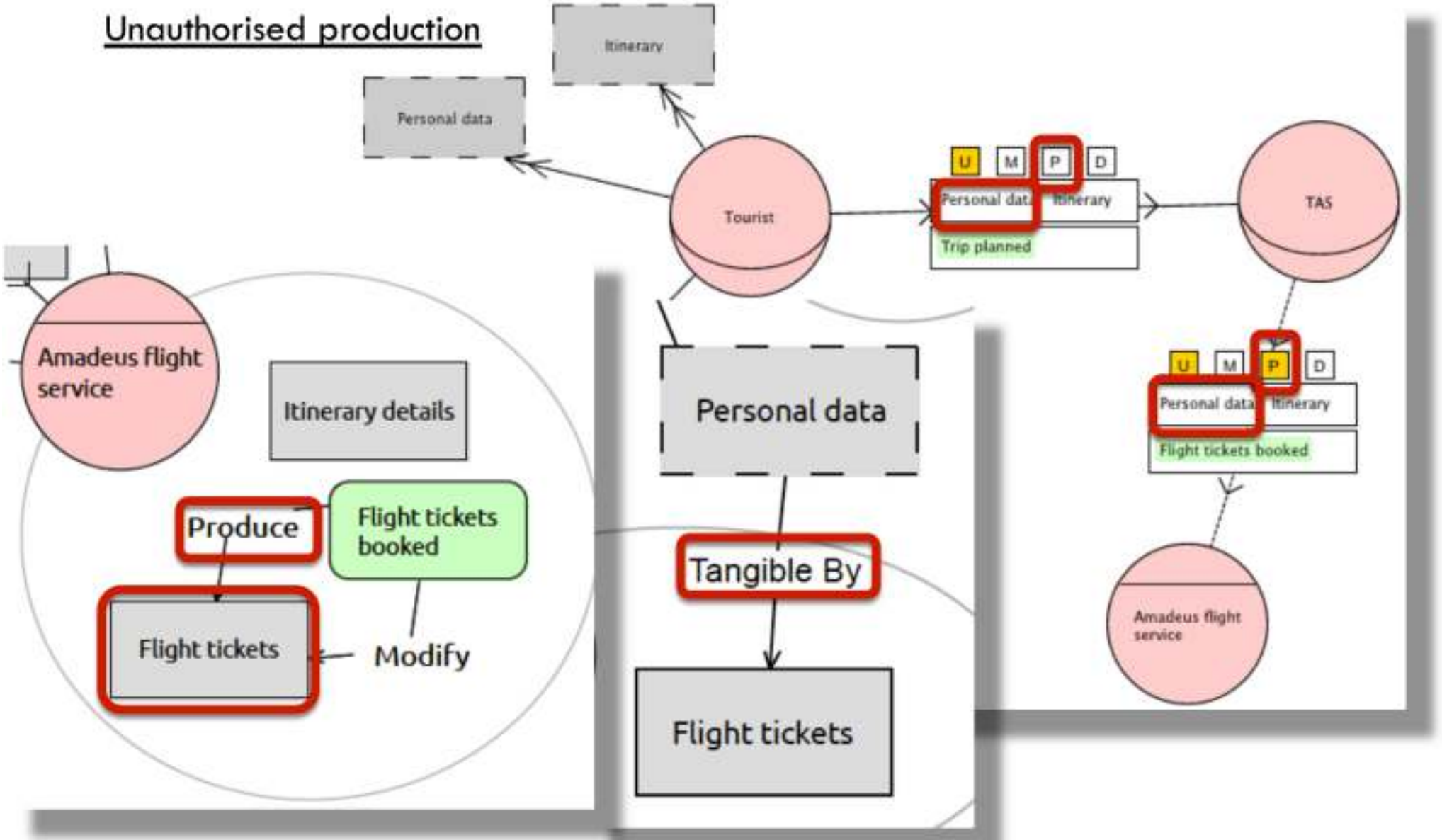
- Two types of inconsistencies
  - Security Requirements Inconsistencies
    - Two or more security requirements cannot be implemented by the same system



Organisational-Security Inconsistencies

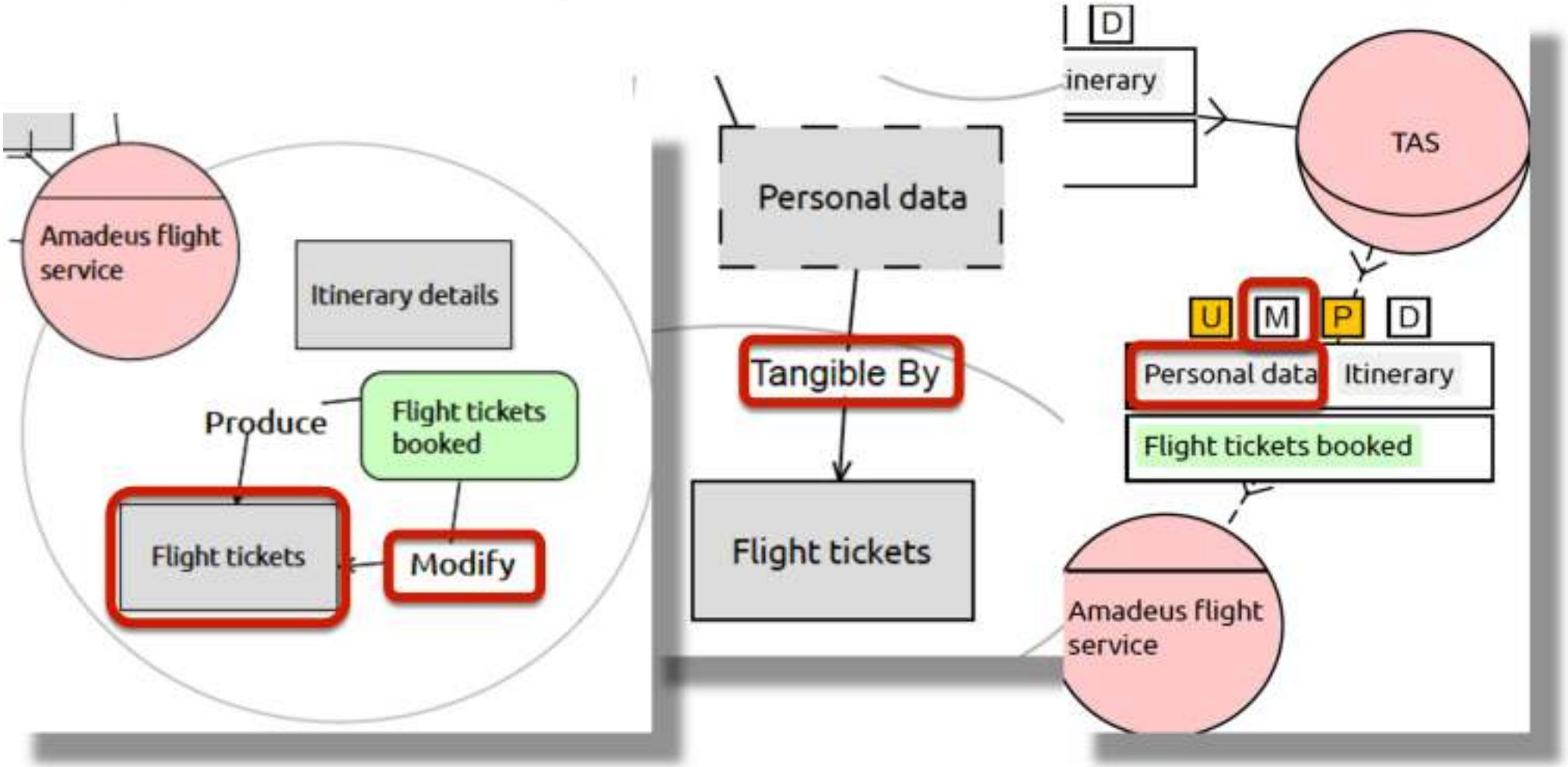
- Unauthorised delegation
  - Delegatee further delegates the goal even though no-delegation is specified
- Unauthorised utilisation
  - Information (or parts of it) is utilised for other purposes than authorised
- Unauthorised delegation of rights
  - Actor does not have the right itself and passes it to others
  - Actor has the rights, but not the right to transfer them to other actors, and still delegates
- Unauthorised Operations
  - Actor uses/modifies/produces/distributes some information without having the authorisation to do so

Example: unauthorised delegation of rights



Example: unauthorised operation

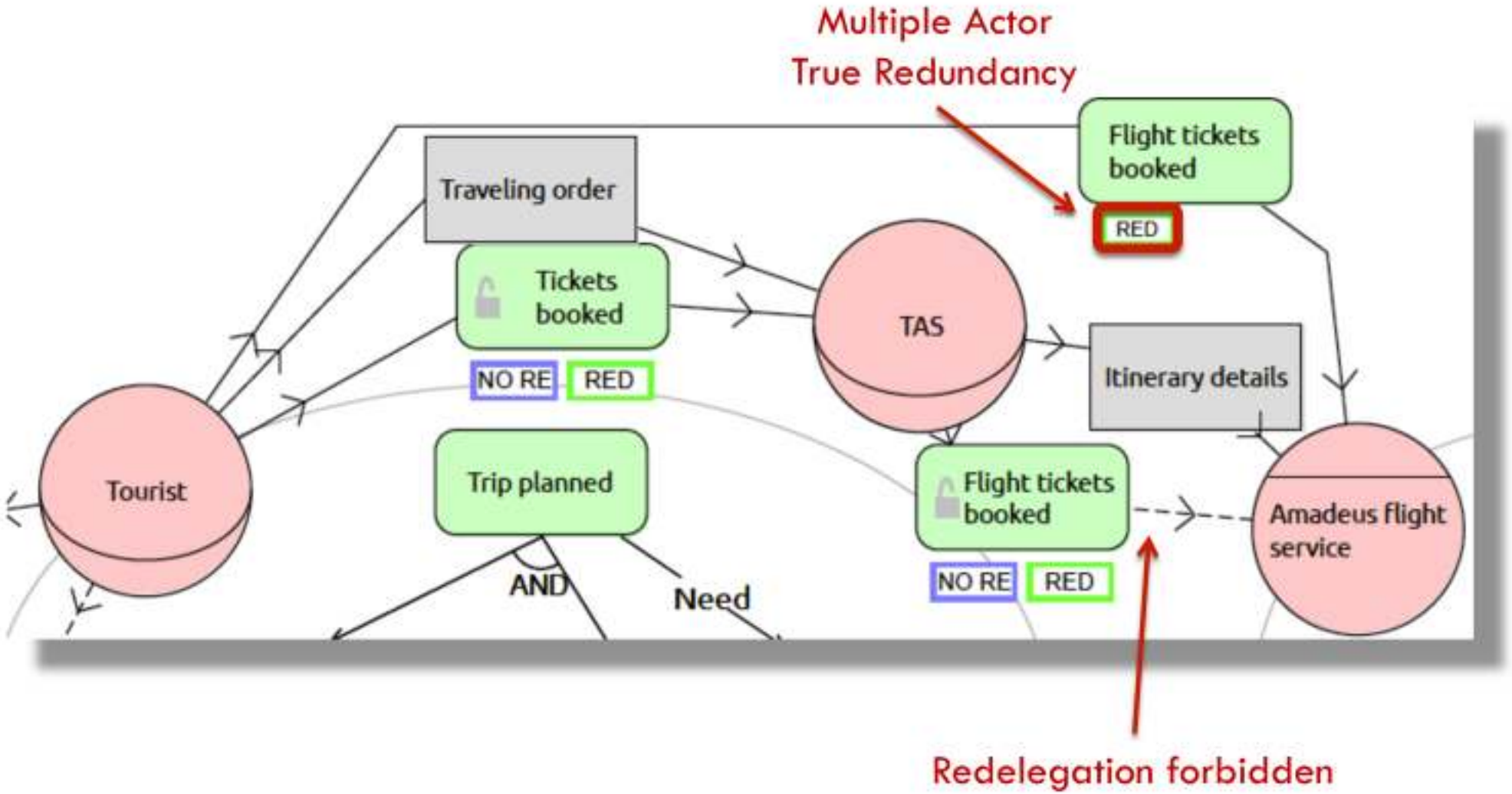
Unauthorised modification



Security Requirements Inconsistencies

- Conflicts over delegations
  - Multiple actor true redundancy and no-delegation
  - Single actor true redundancy and no-delegation result in single actor fallback redundancy
- Conflicts over authorisations
  - Actor receives contradicting authorisations from at least two different authorised actors
  - 5 types of conflicts (per operation + transferability)

Example: conflicts in delegations



Ongoing and Future Work

- Revise the formalisation
- Implement automated reasoning framework
- Evaluation
  - 3 different case studies
    - Air traffic management
    - E-Government
    - Telecommunication

The end

Thank you!

Questions?