ESWC 06, Budva, Montenegro

REWERSE© reasoning on the web

Semantic Web Policies

a discussion of requirements and research issues

P. Bonatti, C. Duma, N. Fuchs, W. Nejdl, D. Olmedilla, J. Peer, N. Shahmehri

REWERSE© WG I2 - policy language

Policies Are Everywhere
beyond security, privacy, and trust

- B2B contracts
  - e.g. quantity flexible contracts, late delivery penalties, etc.
- Negotiation
  - e.g. rules associated with auction mechanisms
- Workflow management
  - What to do under different sets of conditions
- Context aware computing
  - What service to invoke to access a particular contextual attribute
  - Context-sensitive preferences
- ...and more...

[by *Norman Sadeh, PSPW panel,* ISWC 2005]

Claims

Policies are interesting for the SW community because:

- Policies are semantic annotations
  - executable
- Policies are shared knowledge bases
- Policy-aware systems / web need

knowledge-based techniques

In this spirit we identify policy-related requirements and research directions

Outline

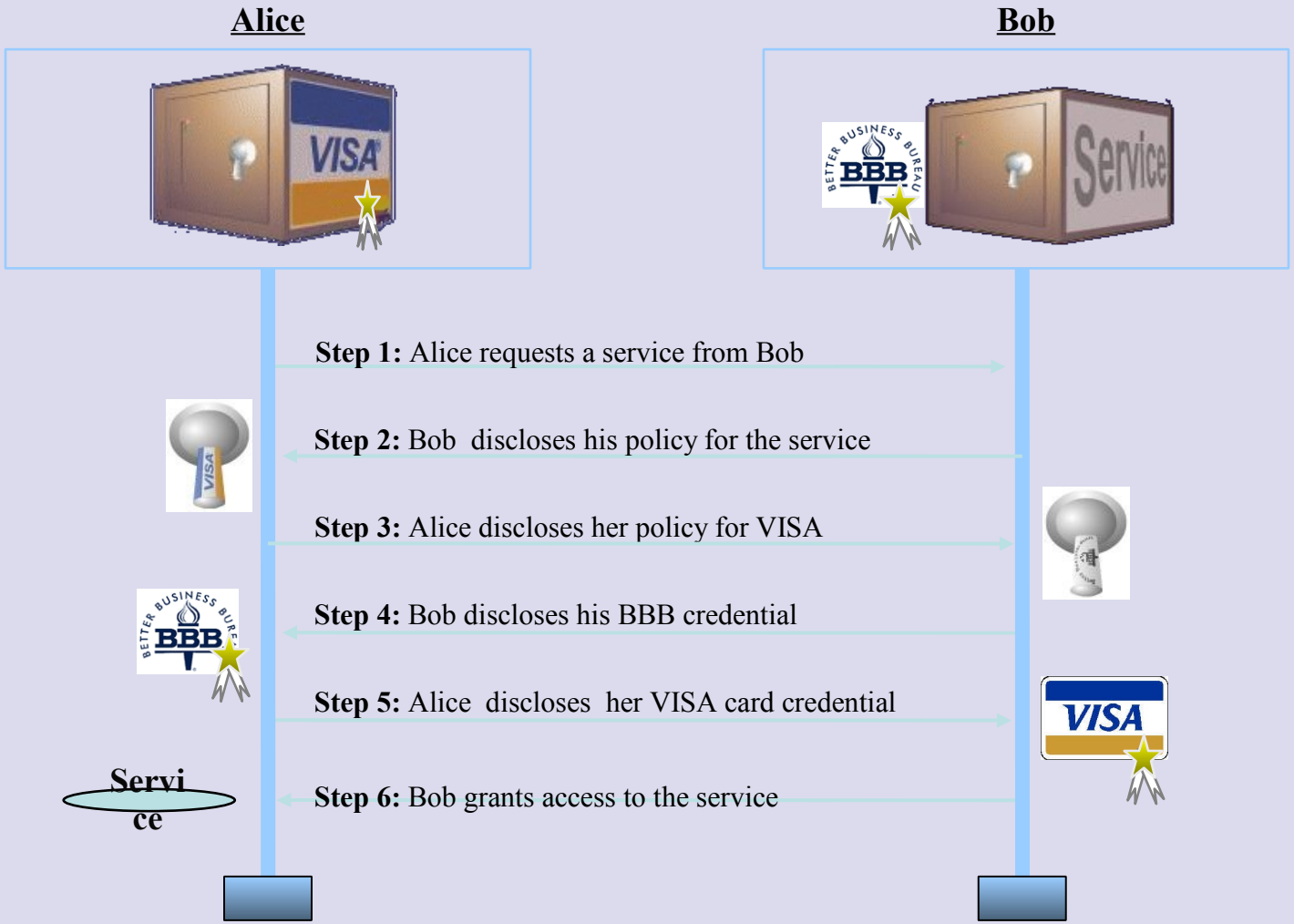
- Introduction / The Context
  - Standard approaches to trust management
- Requirements and open problems
  - Expressiveness
  - User awareness & control over policies
- What's new? The role of
  - knowledge-based techniques
  - and semantic web ideas

Everything started from:

- Computer security for open systems
  - Occasional users, unknown to the system
    - Traditional authentication is impossible or undesirable
    - Property-based access control
    - Digital credentials
- Privacy issues
  - Unknown servers
    - Limit disclosure of sensitive information
    - Raise the level of trust in the server
- Together security and privacy lead to **negotiations**

Negotiations

symmetric framework: credential are resources



[*Bonatti, Samarati.* **A Uniform Framework...** CCS 2000 and J. of Comp. Security 2002]

Formulating requests

- One by one?
  - Slow (more messages)
  - Bad w.r.t. privacy (unnecessary disclosures)
    - After submitting *n* credentials you realize you miss the next
- All alternatives at once?
  - Less messages (good!)
  - Combinatorial explosion: *one id and one credit card* →
    - Passport + VISA
    - Passport + Mastercard
    - ...
    - Student card + VISA
    - Student card + Mastercard
    - ...
- Send the policy!
  - As a compact representation of all alternatives

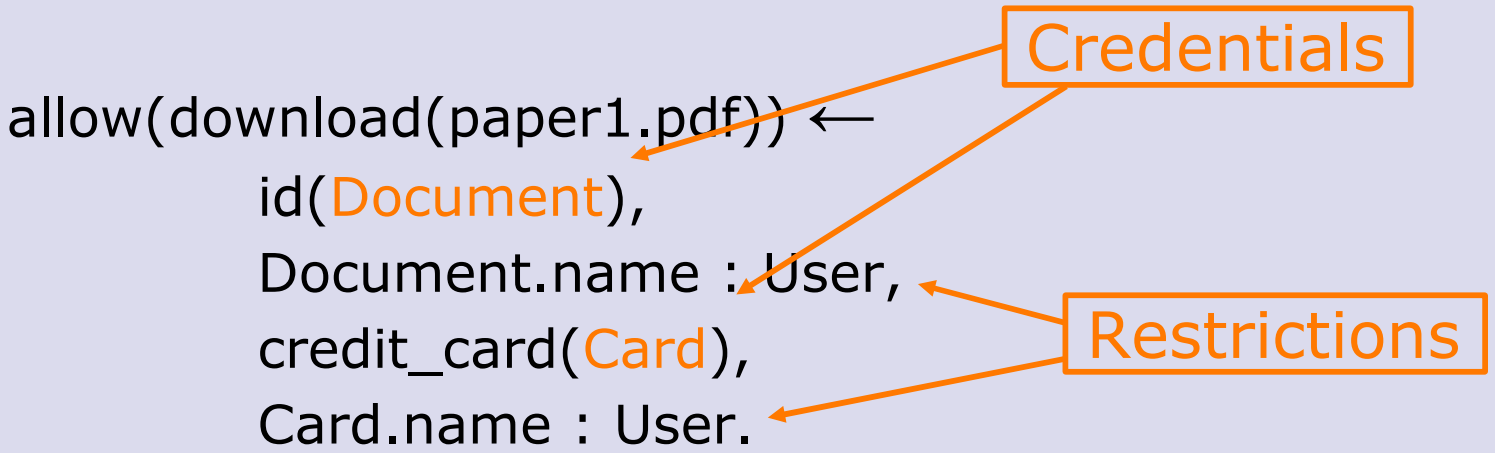
Expressiveness issues
how to represent the policy

- Many frameworks adopt **rules** (natural!)
- Arbitrary boolean combinations of credentials
- Restrictions on their attributes
- Possibly **recursive** conditions
  - Credential chains (~ transitive closure)

```
allow(download(paper1.pdf)) ←
        id(Document),
        Document.name : User,
        credit_card(Card),
        Card.name : User.
```

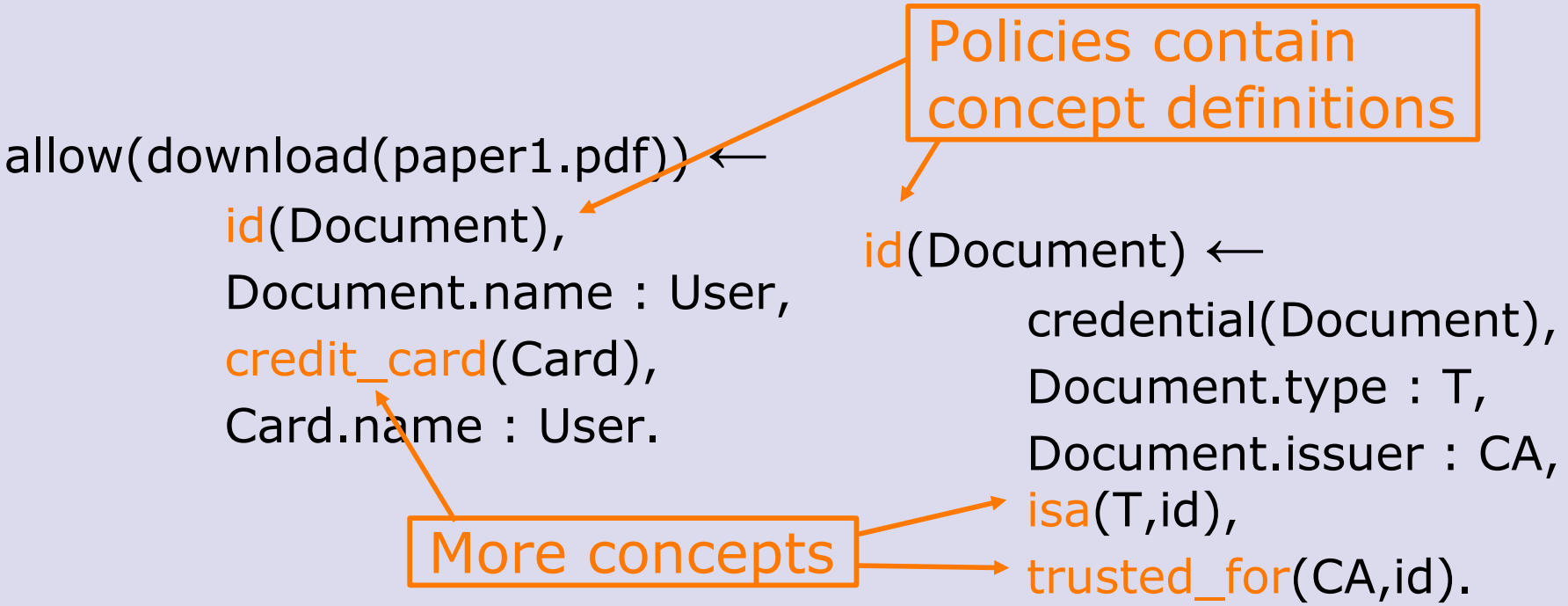
Expressiveness issues
how to represent the policy

- Many frameworks adopt **rules** (natural!)
- Arbitrary boolean combinations of credentials
- Restrictions on their attributes
- Possibly **recursive** conditions
  - Credential chains (~ transitive closure)



Expressiveness issues
how to represent the policy

- Policies define **concepts**



Therefore policies are

- Knowledge bases
- Containing simple ontologies
  - possibly rule-based
- Shared among peers (during negotiations)
- Enabling interoperability of heterogeneous peers
  - w.r.t. access control and information release
- Policies comprise both
  - Semantic **markup for decision making** and
  - The **ontology** for expressing the markup

Relevance to SW community

Regardless of whether

- Policies protect semantic data
- Policies refer to OWL ontologies

Minimal prerequisites for application: common understanding of

- Rule semantics
- Credential format (X.509 standard)
- No further semantic infrastructure needed
- Lightweight reasoning (Horn clauses)

Very close to short-term applications

Expressiveness issues

A broader notion of Policy

The term *policy* covers:

- Security/Privacy policies, Trust management
- Business rules
- Quality of Service directives
- Service-level agreements
  - *and more...*

They all make decisions based on similar pieces of information (evidence)

- user age, nationality, etc.
- customer profile,
- something that the user has (e.g. money)
- reputation
- contextual information

Policies are not (only) passive objects

Policies may specify

- Event logging
- Communications and notifications
- Workflow triggering
  - such as (partly) manual registration procedures

i.e. Policies may specify actions

- To be interleaved with the decision process

So policies are about

- *Decision support + behavior specifications*
  - **declarative** (despite the word "behavior")

Many Policies, One Framework

It is appealing to integrate all these aspects in one framework

- One common infrastructure
  - for **interoperability** and **decision making**
- Where policies can be harmonized & coordinated

Technical challenge

- Harmonize/integrate requirements
  - procedural (ECA) vs. declarative semantics
  - different derivation strategies
  - too complex for one representation language?

Strong, Soft, and Lightweight Evidence

How can individuals *prove* their eligibility?

- Strong evidence
  - e.g. **digital credentials** (id, credit cards, subscriptions)
- Soft evidence
  - e.g. **numerical reputation measures**
- Lightweight evidence
  - e.g. **"accept buttons"** (copyright/license agreements)

They should be integrated for balancing:

- trust level
- risk level
- computational costs
- usability (fetching credentials, personal assistants)

Strong, Soft, and Lightweight Evidence

How can individuals *prove* their eligibility?

- Strong evidence
  - e.g. **digital credentials**
- Soft evidence
  - e.g. **numerical reputation measures**
- Lightweight evidence
  - e.g. **"accept buttons"**

They should be integrated for balancing:

- trust level
- risk level
- computational costs
- usability (fetching credentials, personal assistants)

E.g. micropayments vs. buying plane tick

Strong, Soft, and Lightweight Evidence

Challenges

- Proper language (discrete + numerical), but
- Reputation models still in early stage
  - new models keep being introduced
  - vulnerabilities (e.g., to coalitions)
  - **parametric frameworks?** (current choice of REWERSE)
    - separate reputation module
    - integrated via generic constructs (cf. rule-based mediators)

Exploiting "external" systems

Decisions need data, information, and knowledge

- Each organization has its own
  - Already available through **legacy software and data**
  - A realistic solution *must* interoperate with them
- Third parties
  - Credit card sites for validity checking
  - Credential repositories
- Variety of web resources

Strong, Soft, and Lightweight Evidence

Challenges

- Interoperability on a larger scale
  - heterogeneous legacy software and third parties
  - more general credential formats
  - lightweight evidence can be based on any web contents
  - how to explain such requirements in a machine-understandable way?
  - **a standard semantic web issue – ontologies**
  - **still lightweight?...**

Strong, Soft, and Lightweight Evidence

Challenges

- Interoperability on a larger scale
  - heterogeneous legacy software and third parties
  - more general credential formats
  - lightweight evidence can be based on any web contents
  - how to explain such requirements in a machine-understandable way?
  - **a standard semantic web issue – ontologies**
  - **still lightweight?...**

E.g. point to a pictur on the conference pa to prove you attende ESWC'06

[J. Hendler]

Strong, Soft, and Lightweight Evidence

Challenges

- Interoperability on a larger scale
  - heterogeneous legacy software and third parties
  - more general credential formats
  - lightweight evidence can be based on any web contents
  - how to explain such requirements in a machine-understandable way?
  - **a standard semantic web issue – ontologies**
  - **still lightweight?...**

Expressive languages, ontology infrastructure

E.g. point to a pictur on the conference pa to prove you attende ESWC'06

[J. Hendler]

More requirements:

User awareness and control

Motivations

- Most violations caused by lack of awareness & control
  - Pre-defined policies: intrusion in **5 min.**
  - With personalized policy: secure for **2 weeks** (end of experiment)

  [Avantgarde. http://www.avantgarde.com/xxxxttln.pdf]
- No awareness on what the policy does or does *not* check
- Users are unable to personalize policies
- A social problem:
  - Everybody's machine is on the internet
  - Millions of computers can be exploited for attacks
  - *By taking advantage of the users' lack of technical competence*

Cooperative policy enforcement

Crucial for the success of a web service

- Never say (only) "*no*"!
- Encourage first-time users

You can't open this d
but you can ask Alice
permission

- Explain policy decisions
  - Advanced queries: Why / why not
- Guide users in acquiring missing permissions
  - Activate registration workflows
  - Provide instructions
  - Advanced queries: how-to, what-if

[*Bonatti, Olmedilla, Peer.* **Advanced Policy Explanations.** ECAI 2006]

A general list of challenges

- Explain policies and system decisions
  - Make rules & reasoning intelligible to the common user
  - A classical AI problem – perfectly in line with SW
- Encourage people to personalize their policies
  - Make it easy for users to write their own rules
- Use natural language?
  - "*Academic users can download the files in folder historical_data whenever their creation date precedes 1942*"
  - Suitably restricted to avoid ambiguities
  - Fortunately, users spontaneously formulate *rules*
- Use graphical language?

Explanation mechanism

Main challenge:

- Finding the right tradeoff between
  - Explanation quality (2nd generation explanation facilities)
    - Remove irrelevant information
    - User-friendly denotation of internal objects
    - User-oriented description of reasoning
  - Framework instantiation effort
    - The framework needs to be adapted to each application domain
    - Expensive in 2nd generation EF (ad hoc KB and engine)
    - Reduce the need for specialized staff

SEMANTIC WEB POLICIES

WHAT'S NEW?

What's new in SW scenarios?

- Security/Privacy/Trust community addressed
  - Open systems
  - Heterogeneous software interoperability
  - Deployment on the web
- No new requirements regarding
  - Public/private nature of policies
  - Stateful/stateless nature of negotiations
  - Unilateral/bilateral forms of negotiations

  (the idea of publishing policy rules is not new, either)

Within the realm of SW and KR&R
not in the main focus of the trust community

- Ontology-based interoperability
  - Pervasive lightweight evidence
- Regard policies as KBs
  - One knowledge – many uses
- Focus on intelligent interfaces
  - Explanations
  - Controlled NL front-ends
- Reasoning about policies
  - Select services based on their policies
  - Policy verification and validation

Within the realm of SW and KR&R
and not really tackled by security people

- Inference and record linkage problems
  - Explicitly released information may **entail** confidential information
  - Possibly using common knowledge and user knowledge
  - Possibly joining different (heterogeneous) data sources
  - Theoretical models exist (e.g. [Biskup et al.]), but
- Currently not checked
  - No machine-understandable model of available knowledge is implemented
- Ontologies and semantic markup
  - Enable automated inference-based attacks, but also
  - Enable automated inference checking (**efficient?**)

[*Biskup, Bonatti.* DKE 01, FoIKS 02, ESORICS 02, IJIS 04, AMAI 04, FoIKS 2006]

Conclusions

- Policies instantiate the SW vision
  - policies are semantic markup
  - and call for flexible and expressive representation languages
- Many interesting possible contributions from the SW community
  - powerful KR&R infrastructure
  - knowledge-based policy handling
  - Inference and record linkage control
- Without re-inventing the wheel

REWERSE
WG I2 - policy language

QUESTIONS?

[More on http://rewerse.net/I2/]