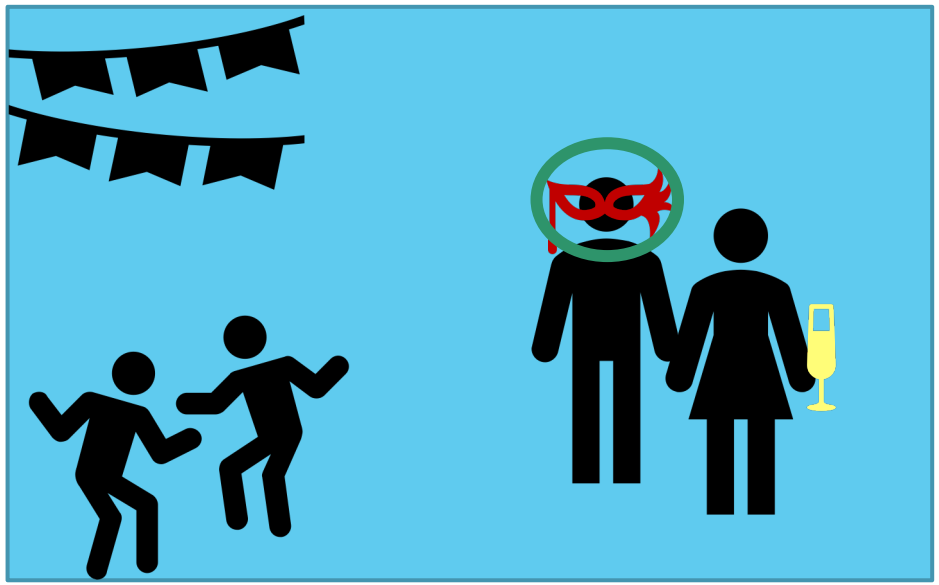
Data Sharing in Social Networks

Clara Bertolissi, **Alba Martinez Anton**, Nicola Zannone

SACMAT'23, June 7-9, Trento, Italy

Privacy of multi-user shared content



A photo posted by Alice of her party

What options has Bob to protect his privacy?

- Ask Alice to modify the visibility of her photo
- Delete the tag
- If the solution is not satisfying, report to the social network

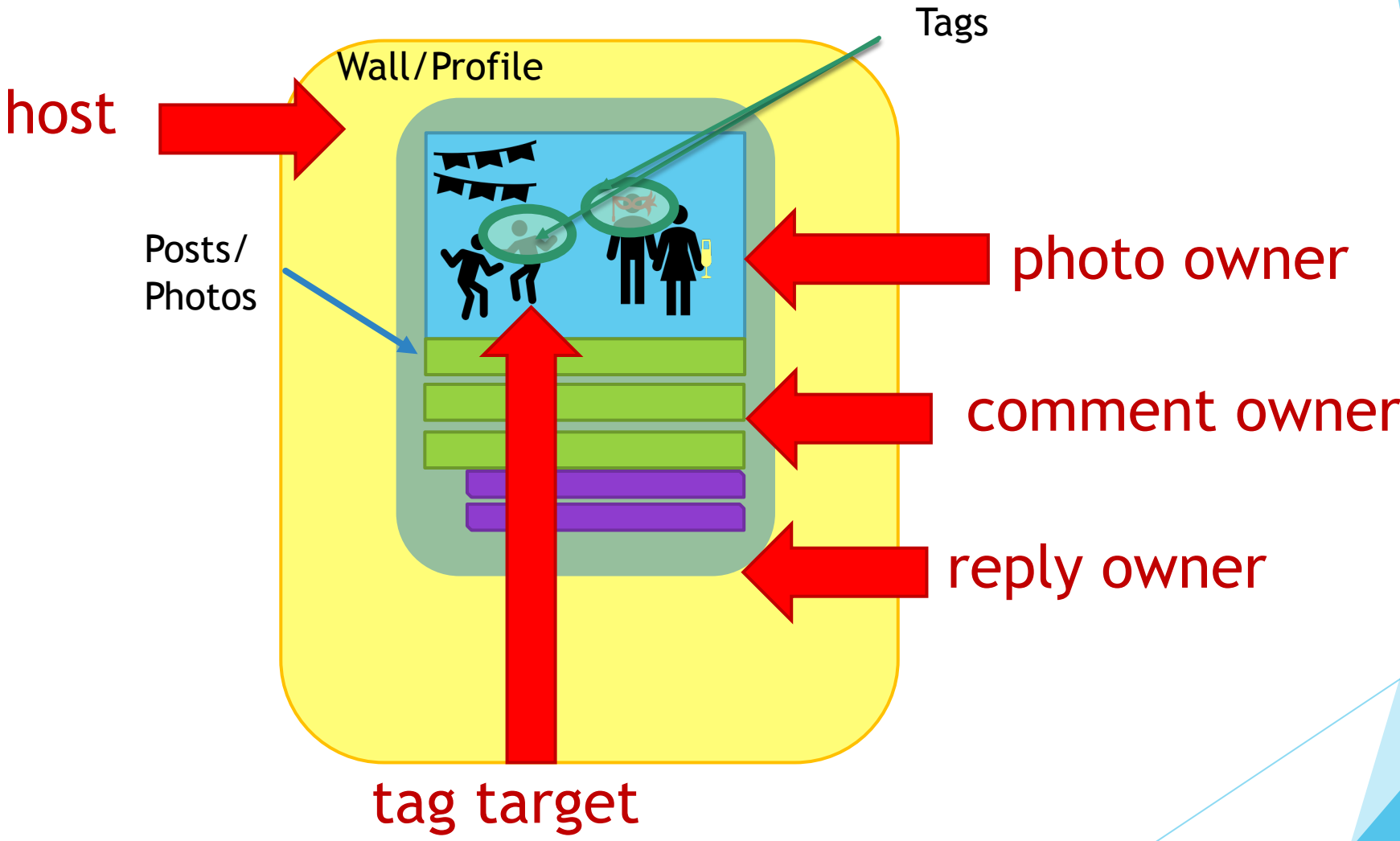
Social netwoks challenges

A piece of data has several co-owners.

The data can be “replicated”.

Some data cannot exist without other piece of data.

Idea : compound objects with multiple controllers



Objective and contributions

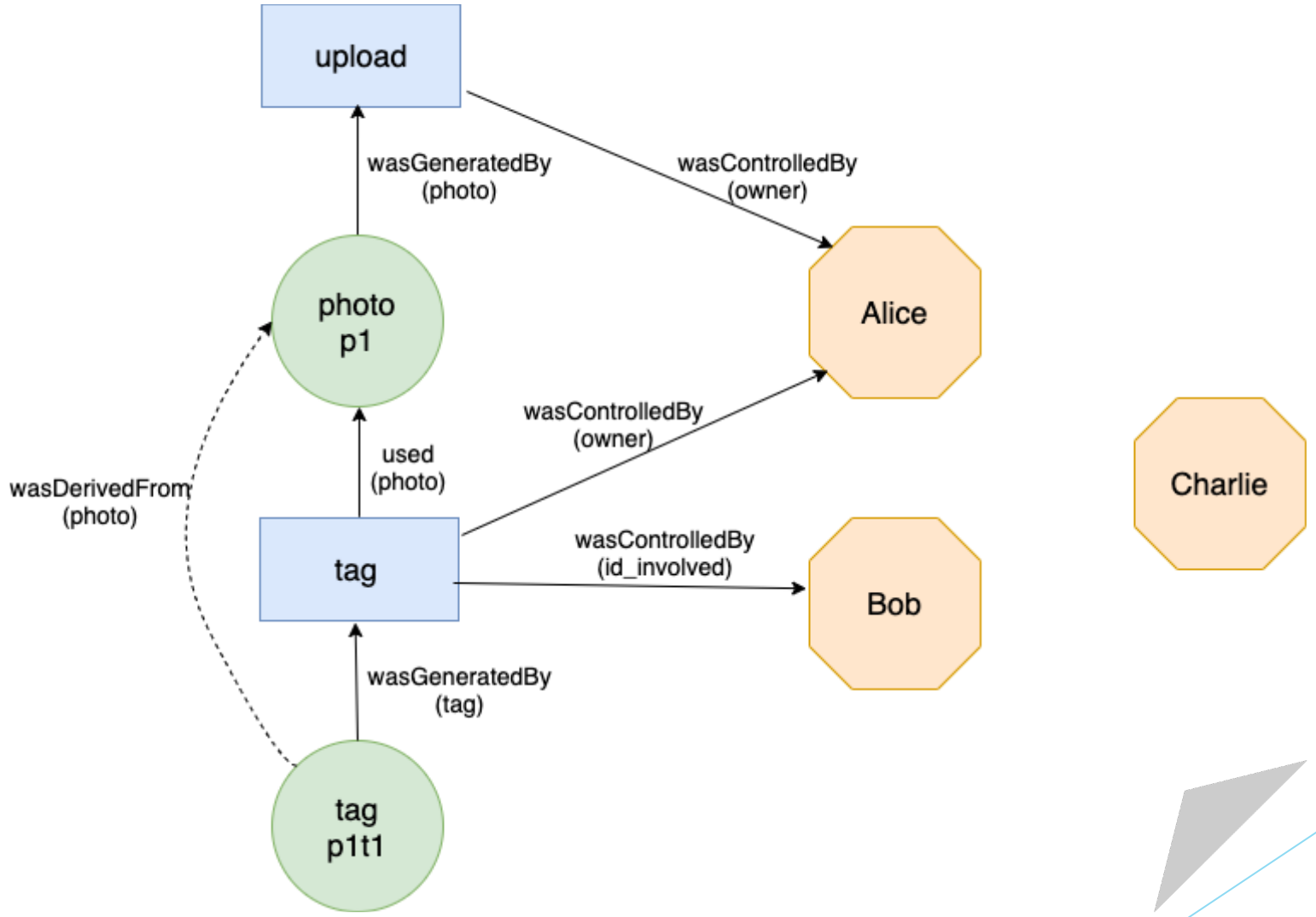
Objective:

Propose a fine-grained access control model for multi-user system

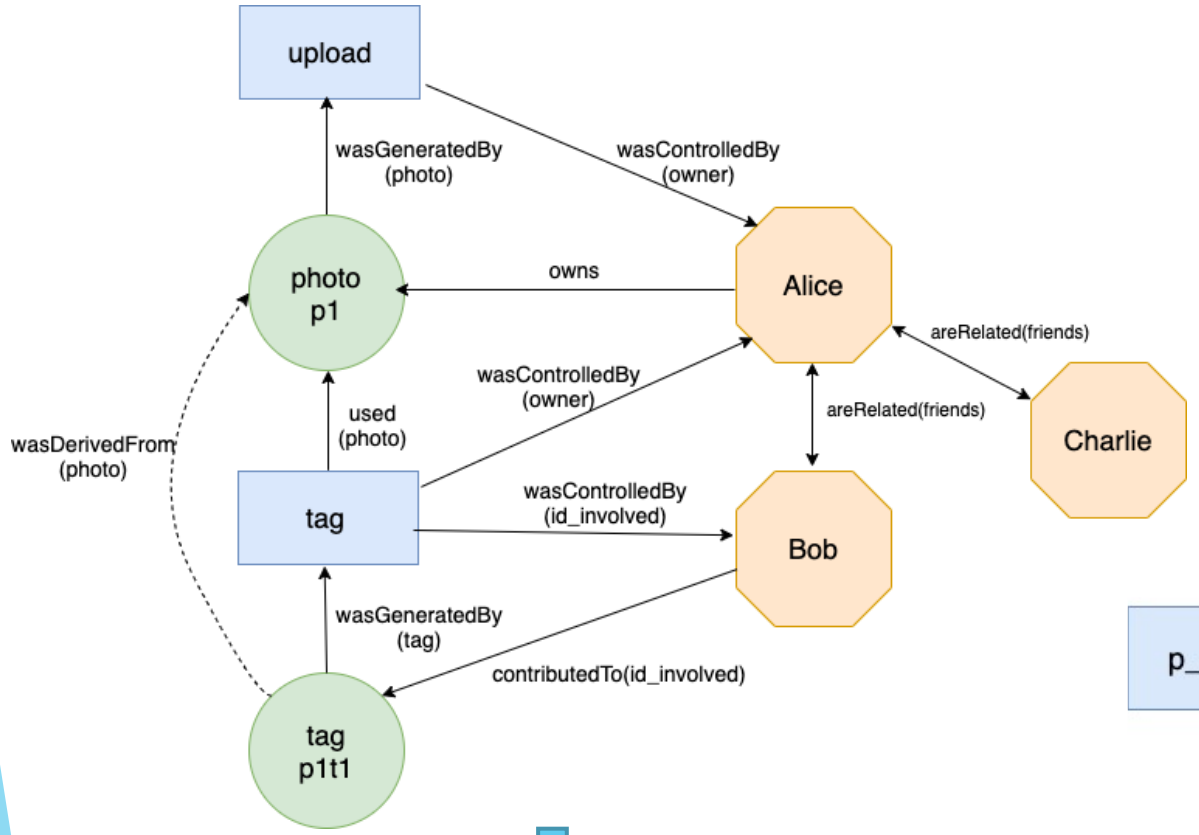
Contribution:

Extend attribute-based access control with provenance information

The Open provenance Model

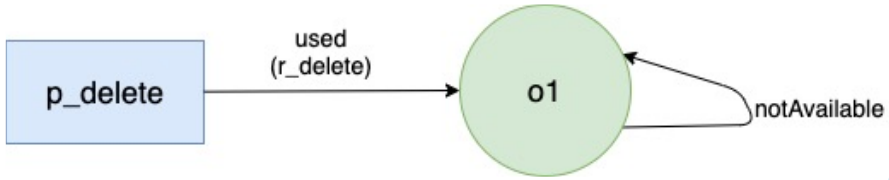


Extension for Social Networks



Evaluation graph

- New dependencies
- Relations between users
- Dynamic systems.



Policy definition: ABAC extended with path conditions

- User policies

  Give the user preferencies

$$pol_{photo,Alice} = (owner \vee host = Alice, type = photo),$$
$$(action = view \wedge areRelated^2(Alice, user_{id}, friend), 1) \vee$$
$$(action = \ldots, 1)$$

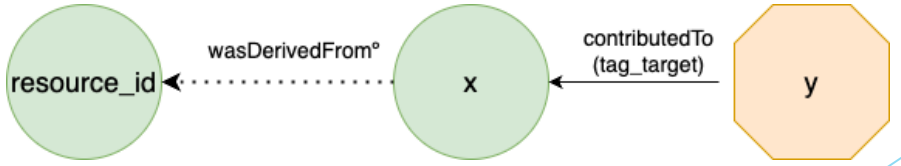
$$pol_{tag,Bob} = (resource_{target} = Bob, type = photo \vee tag),$$
$$(action = view \wedge areRelated(Bob, user_{id}, friend), 1)$$

- System policies

  Give the way that preferences have to be combined for evaluate an access

$$pol_{photo,SN} = (type = photo), (action = view, pol_{photo,host})$$

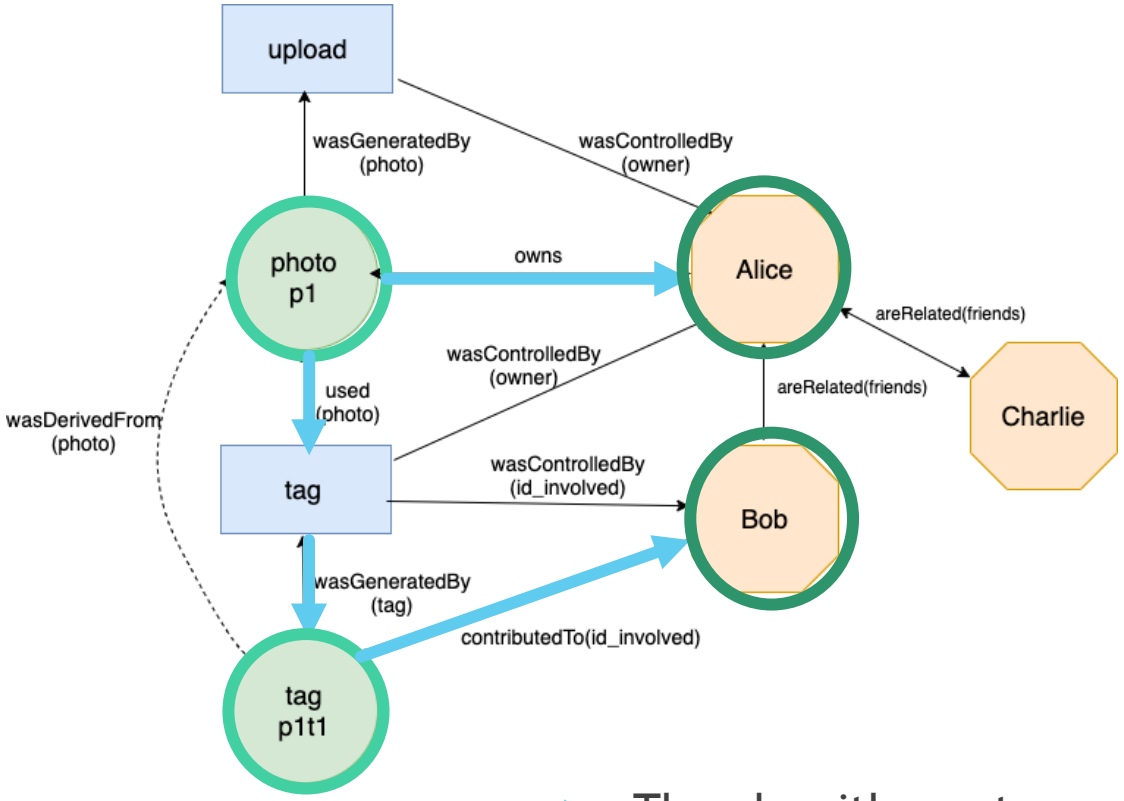
$$pol_{photo,SN} = (type = photo), (action = view,$$
$$dov(pol_{photo,owner}, pol_{photo,host}, dov(uri(x,y)$$
$$|wDf^{o}(x, resource_{id}) \wedge contributedTo(y, x, tag_{target}))$$



Policy evaluation

- Charlie wants to see Alice's photo

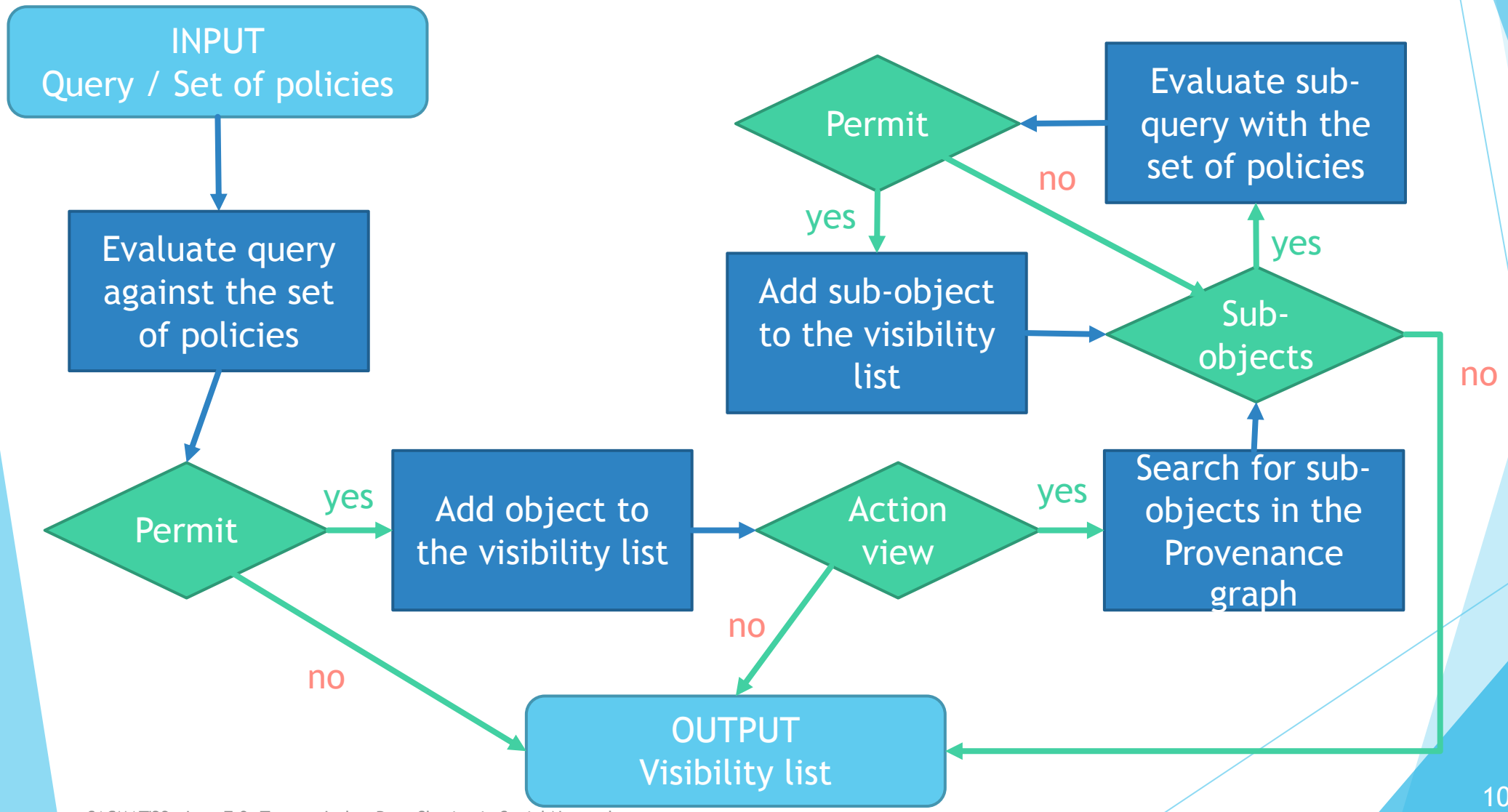
q = {(user_id,Charlie),(resource_id, p1), (action,view)}



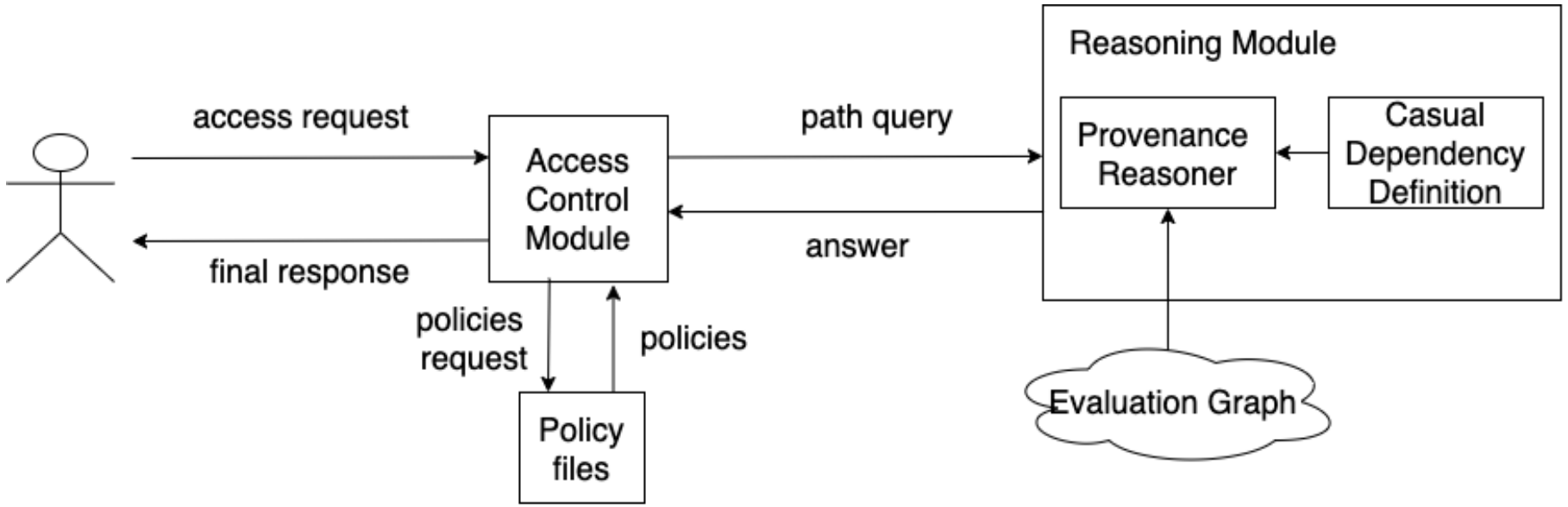
- Algorithm Steps:
  - Evaluation of the system policy
  - Search of the sub-objects in the provenance information and their controllers
  - Evaluation of the access to the sub-objects
  - Return the list of the available objects

- The algorithm returns: [[p1, permit], [p1t1, deny]]

Algorithm to evaluate requests



Prototype architecture

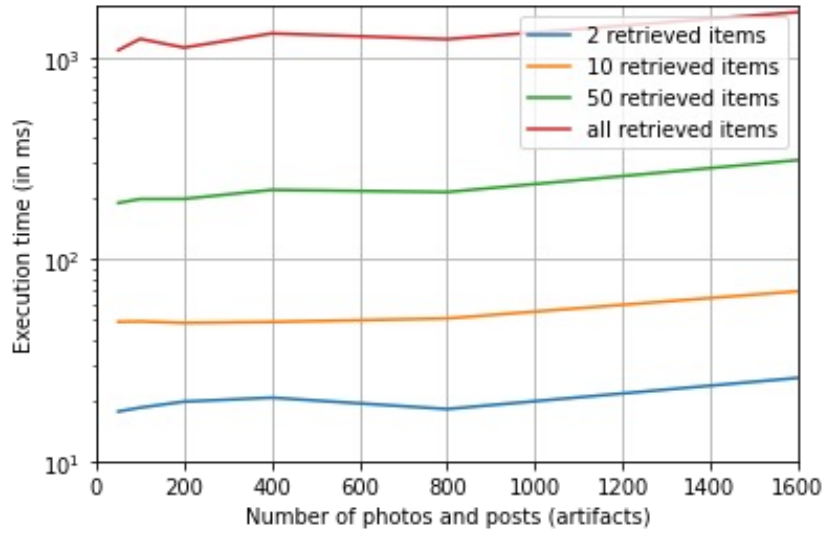


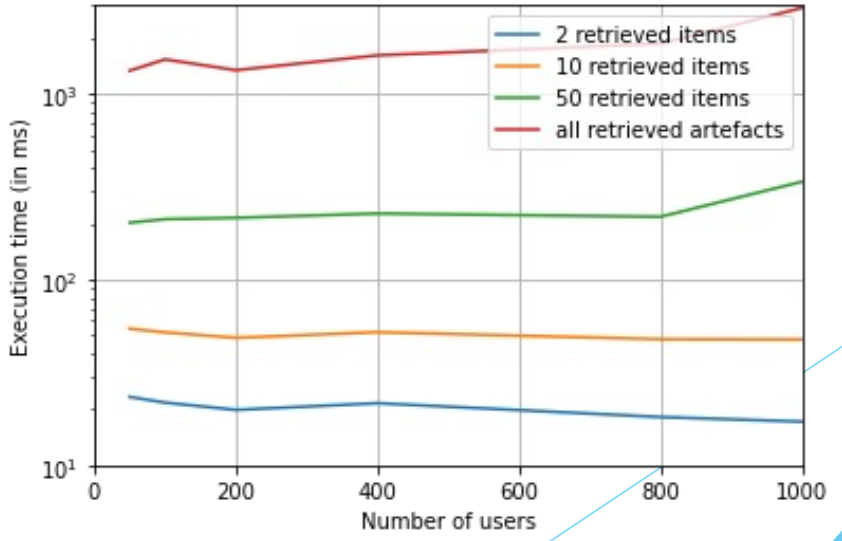
Prototype: Some Results

- Limited impact of the number of post and photos
- Important impact of the number of sub-objects

… but in real life …

- Number of sub-objects often limited, or displayed bit by bit





Conclusion & Future work

- ✓ Fine-grained access control for sharing data in OSN
- Integration into an XACML architecture
- Extension for transient relationship
- Conflict resolution and management