# CooPeD: Co-owned Personal Data Management

Lorena González-Manzano*, Ana I. González-Tablas*, José M. de Fuentes*, Arturo Ribagorda*

*Avda. de la Universidad, 30. Computer Science and Engineering Department. University Carlos III of Madrid, 28911 Leganés, Spain*

**Abstract**

With the spread of Web-Based Social Networks (WBSNs) managing access to data is a challenging matter. Providing personalized, fine-grained access control is essential to build trusted WBSNs. WBSNs data can be associated with owners and co-owners, namely users who upload the data and users who are linked to uploaded data respectively. Thus, a privacy-friendly WBSN must allow users the management of elements related to them. In this regard, CooPeD (Co-owned Personal Data management), a system that deals with co-ownership management of decomposable objects, is proposed. CooPeD is formed by a model and a mechanism. CooPeD is developed on the bases of $SoNeUCON_{ABC}$ usage control model. Particularly, an extension of $SoNeUCON_{ABC}$ is proposed to support co-ownership management by means of access control and administrative management. In CooPeD's mechanism objects, decomposed in parts, are attached to owners and co-owners who individually set their access control preferences. The evaluation of CooPeD consists of three parts. Firstly, a feasibility analysis for different architectures of CooPeD's model and mechanism, as well as of CooPeD's mechanism in Facebook is performed. Secondly, a prototype proves the feasibility of implementing CooPeD. Lastly, a survey study assesses the acceptance of CooPeD.

*Corresponding author

*Email addresses:* lgmanzan@inf.uc3m.es (Lorena González-Manzano ), aigonzal@inf.uc3m.es (Ana I. González-Tablas), jfuentes@inf.uc3m.es (José M. de Fuentes), arturo@inf.uc3m.es (Arturo Ribagorda)

*Keywords:* Co-ownership management, web-based social networks, access control policies, privacy, trust

## 1. Introduction

Web Based Social Networks (WBSNs) are an emerging social phenomenon. In these applications users upload a huge quantity of data, some of them personal data, which are in many cases out of control. The challenge is to control and carefully manage all WBSNs data, data either uploaded by ourselves or by other users with whom we have some kind of relationship. This issue is becoming even more challenging with the spread of everyday systems (e.g. cell phones, cameras, etc.) which actively share user-related data. Physical systems are more and more integrated in WBSNs. As an example, Facedeals' cameras recognize shoppers based on previously uploaded tagged Facebook pictures [1]. Thus, privacy preservation is of utmost relevance for building next generation trusted cyber-physical systems.

Data managed in WBSNs can be associated with multiple users, the owner and the co-owners. The former refers to the user who uploads the data and the latter refers to users who are associated with uploaded data. In relation to WBSNs, tagging is the most common technique to grant co-ownership [1]. Users become co-owners of objects in which they are tagged. Then, they may be exposed to unknown people and, unexpectedly, to unknown risks because they cannot manage access control on these objects.

Assorted techniques have been developed to combine owners and co-owners preferences (see Section 10). The voting scheme is the most common negotiation technique but disjoint preferences may compromise users privacy. Only K. Thomas *et al.* propose the intersection of all users preferences to completely preserve users privacy [2]. This approach is significantly restrictive because access

[1] Automatic photo tagging on Facebook while you shop, URL: http://www.neowin.net/news/automatic-photo-tagging-on-facebook-while-you-shop, last access December 2013

is denied unless all users reach a full consensus. Consequently, even being desirable the preservation of users privacy, a trade-off between privacy and users demands is an essential requirement. In other words, a system may become useless if it is too restrictive.

In view of the foregoing, this paper presents CooPeD (*Co-owned Personal Data Management*), a system that deals with co-ownership management of decomposable objects. It is formed by a model and a mechanism. CooPeD's model is based on the $SoNeUCON_{ABC}$ usage control model [3]. This usage control model provides expressive, fine-grained access control management for WBSNs. It allows managing privacy preferences regarding attributes of WBSN users, their data and the relationships between them. To address co-ownership, in this paper $SoNeUCON_{ABC}$ is extended by means of access control and administrative management.

In what concerns CooPeD's mechanism, it is focused on managing objects that are composed of parts, as it happens in [4]. Owners upload objects and manually or automatically assign parts to users to whom they are related, being these users referred to as co-owners. Each owner and co-owners individually manage their privacy preferences. Thus, instead of granting or denying access to an entire piece of data (which is the approach usually taken by current developments), in CooPeD an object is divided in parts and each of them can be accessed by different requesters.

The approach taken can be applied to any decomposable object. It must be noted that several of the most used WBSNs[2] (e.g. Facebook, Pinterest, Instagram or Google+) are particularly focused on images. Other WBSN data such as text comments are also well-known but the upload of photos remains being the most popular activity and Facebook is a key example in this regard[3].

[2] http://socialmediatoday.com/jonathan-bernstein/1894441/social-media-stats-facts-2013, last access March 2014

[3] http://www.jeffbullas.com/2013/09/20/12-awesome-social-media-facts-and-statistics-for-2013/, last access March 2014

Indeed, text comments are commonly based on images. Taking into account these facts, CooPeD is focused on image-based data (photos and videos without audio) without loss of generality.

This proposal has been evaluated in three ways. First, a feasibility analysis for different architectures of CooPeD's system (i.e. model and mechanism) is performed. Afterwards, although current WBSN do not apply the proposed model, the application of CooPeD's mechanism in a real-world WBSN (namely Facebook) is assessed. Second, a prototype has been built to prove the feasibility of implementing CooPeD. Third, a survey has been conducted to assess the usefulness and appealing of the proposal.

This document is structured as follows. Section 2 provides a background on the $SoNeUCON_{ABC}$ usage control model. Next, Section 3 presents an overview of the approach. In Section 4 CooPeD's model is described. Section 5 presents CooPeD's mechanism. The architectures in which CooPeD can be developed are described in Section 6. In Section 7 the feasibility of applying CooPeD system and of applying CooPeD's mechanism in Facebook is studied. Section 8 describes the developed prototype. In Section 9 the survey study is presented. Related work is introduced in Section 10. Lastly, conclusions and future work are presented in Section 11.

## 2. Background: $SoNeUCON_{ABC}$

$SoNeUCON_{ABC}$ usage control model has been developed to attain fine-grained access control management along the whole usage process in WBSNs and it is particularly focused on relationships management [3]. In this regard, $SoNeUCON_{ABC}$ is an expressive usage control model that allows the management of relationships, objects and subjects, mainly expressing a set of six features: distance [5], common-contacts [6], clique [6], multi-path [7], direction [6, 5] and flexible attributes [8]. Note that a brief introduction to $SoNeUCON_{ABC}$ model is provided herein, for more details see [3].

*2.1. Elements description*

Assuming that WBSNs are represented as graphs in which users are the nodes and relationship are the edges [9], this model manages the following sets of elements:

- *Subjects* ($S$) are the WBSN users. $ATT(S)$ denotes the set of subjects' attributes.
- *Objects* ($O$) are WBSN data, identified as photos, videos, wall messages and personal messages. $ATT(O)$ denotes the set of object attributes.
- *Relationships* ($RT$) represent the set of relations that exist between a pair of users of the WBSN. Given two users $v_i$ and $v_j$, such set is denoted as P($v_i$,$v_j$) which involve direct (i.e. one hop) and indirect relationships (i.e. multi-hop). $ATT(E)$ denotes the set of direct relationships' (edges) attributes. Note that indirect relationships are composed of direct ones.
- *Rights* ($R$) refer to the actions that can be performed over WBSN data such as reading, update or deletion.
- *Authorizations* ($A$) are the rules defined as functional predicates that have to be satisfied in order to grant a subject a right on an object. Throughout this paper these elements will be called policies.
- *Obligations* ($B$) refers to activities that have to be carried out by the subject S before or while the usage process.
- *Conditions* ($C$) correspond to a set of contextual features, such as network availability, etc.

Each object has a single administrator who is the user who uploads objects to the WBSN. Each administrator may specify a set of policies that will be applied to all his/ her administered objects. These policies will specify the rights to execute over objects, e.g. read or copy.

### 2.2. Access control policies description

Access control policies $\rho$ are denoted as $\rho(\rho_s; \rho_o; \rho_{rt}; r; \partial_b; \partial_c)$. $\rho_s$, $\rho_o$ and $\rho_{rt}$ are predicates defined, respectively, over the attributes of subjects, objects and relationships. In the case of relationships, it is considered the set P($v_a$,$v_s$) where $v_a$ is the administrator of the requested object and $v_s$ the subject that requests access to it. Moreover, $r$ denotes rights and $\partial_b$ and $\partial_c$ refer to sets of obligations and conditions.

An example of an access control policy would be the following:

*Read access to photos titled party is granted to friends if they are females under 30 years old or if they are females under 40 who have studied computer science or if they are females who have studied computer science and physics.*

$\rho = (((gender = female) \wedge ((age < 30) \vee ((age < 40) \wedge (studies = c.science)) \vee ((studies = c.science) \wedge (studies = physics)))); (title = party); ((((role = friend))), \emptyset, \emptyset); read; \emptyset; \emptyset)$

## 3. CooPeD's overview and scope

This Section presents an overview of the approach (Section 3.1) and its scope (Section 3.2).

### 3.1. CooPeD overview

CooPeD is a system that deals with the management of access control for co-owned data. It is focused on image-based objects which are composed by different parts and a background, such that $Object_j = \sum_i Object_j.Part_i + Object_j.Background$.

CooPeD is based on $SoNeUCON_{ABC}$ usage control model to manage access control. In $SoNeUCON_{ABC}$, each object is managed by the user who uploads it, that is, its owner (also referred to as administrator). Nonetheless, to address co-ownership, CooPeD extends this model so that each $Object_j.Part_i$ is managed by a single user who is related to it. In this way, such user becomes the co-owner of the whole object. Note that $Object_j.Background$ is a fixed part of each object

which is exclusively related and managed by its owner. Due to these matters, the extension affects access control and administrative management.

In general terms, once the owner and the co-owners have specified access control policies, access to parts and background is granted or denied accordingly. If access is granted some parts may be visible and some others may be hidden. An example is presented in Figure 1, where an object is composed of three parts in addition to the background and where a pair of co-owners and the owner establish access control policies. The owner creates a policy to grant access to users older than 18. By contrast, co-owner$_1$ grants access to users older than 24 and co-owner$_2$ creates two policies, one to grant access to friends and other to grant access to users older than 20. Due to these restrictions three different situations are distinguished: 1) Part$_1$ is the only one hidden, e.g. a user who is 23 years old gets access to Part$_2$, Part$_3$ and the background; 2) Part$_1$ and Part$_2$ are hidden, e.g. a user who is 19 years old gets access to Part$_1$ and the background; and 3) all parts, as well as the background, are hidden, e.g. a user who is 16 years old does not get access. Therefore, access control management is privacy-preserving, considering and respecting the privacy preferences of all users. Note that a simple technique to hide images parts consists of using opaque figures, while a more sophisticated one may be focused on replacing a part (a component) with another that prevents the identification of the replaced one.

It must be noted that CooPeD does not require negotiation or agreement among co-owners and their privacy preferences. Each object part is independently managed by a particular user and conflicts cannot exist.

### *3.2. CooPeD scope*

According to the EU Data Protection Directive (95/46/EC), personal data refers to "*any information relating to an identified or identifiable natural person ('Data Subject'); an identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to his physical, physiological, mental, economic, cultural*

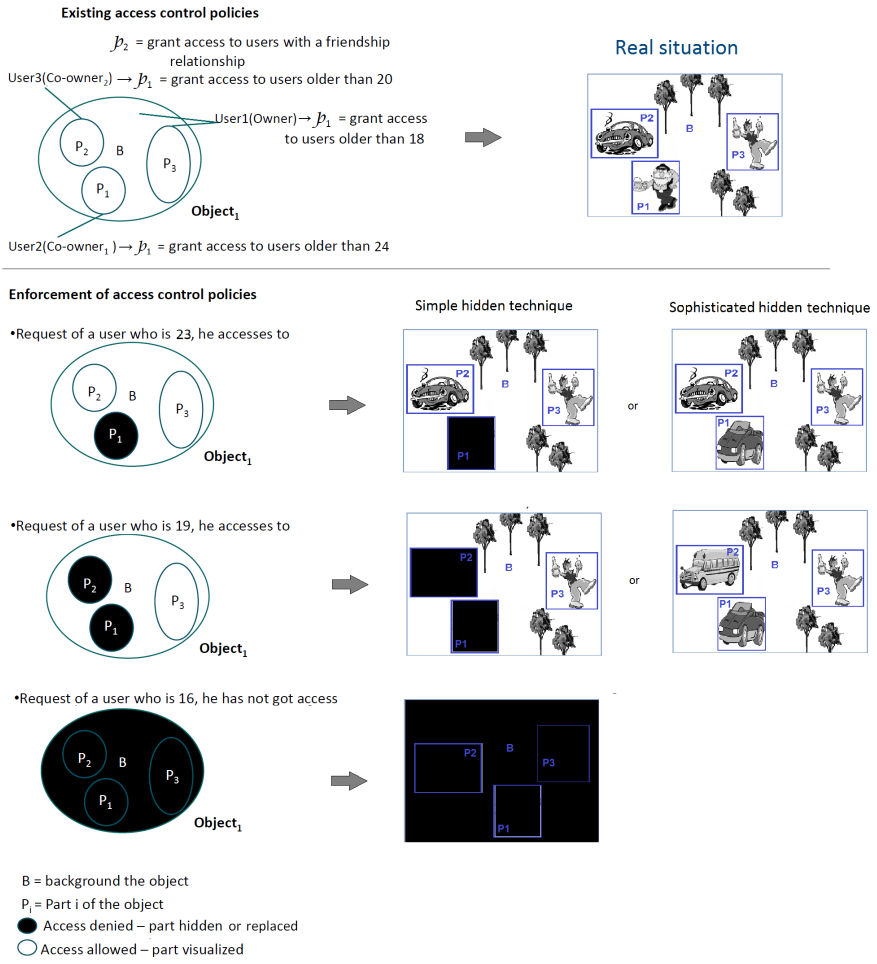



Figure 1: Co-ownership management of an object

*or social identity*"[4]. Then, parts of objects should correspond to elements that identify or facilitate the identification of a particular user.

In this proposal it is assumed that each object part belongs to a single user who manages it. Then, to achieve finer granularity the management of parts related to multiple users is left for future work. For instance, an image of a couple of users in front of their house opens up the discussion about who has to manage the part of the image related to the house.

Besides, CooPeD is based on image-based data, namely, photos composed of assorted elements such as users, vehicles, animals, etc. and videos without audio. H. Lipford *et al.* mentioned the appropriate use and possible application of graphical techniques to manage access control [10], being essential the analysis of recognition techniques to identify elements within photos. Similarly, videos without audio involve many photos per video sequence and thus, their management can be similarly performed but involving more computational costs. Nonetheless, other objects like documents, music or videos with audio can be also applied. In the case of documents, the appropriate sentences can be hidden; in relation to music, the right notes can be silenced; and in regard to videos with audio, the appropriate tracks can be omitted. In particular, documents decomposition would require the analysis of semantic processing techniques, being the study of recommender systems a key starting point [11]. Identifying how recommender systems work helps to recognize how sentences are associated with a given user. There are a lot of recommender systems, like content-based recommender systems focused on suggesting an item to a user in relation to items linked to this user in the past, or keyword-based systems focused on searching a particular keyword in a given text [12].

In respect to music or audio files, decomposition requires audio signal processing [13]. Specifically, the recognition of the voice of a given user and the identification of the same user throughout the audio is indispensable. However, speech recognition is hard due to the richness of languages, that is, natural

[4]http://ec.europa.eu/justice/index_hr.htm, last access December 2013

speech is continuous, with different pronunciations, large vocabulary, etc [13].

Moreover, concerning the decomposition of videos with audio, it consists of combining image and audio signal processing techniques.

All in all, given the difficulty in decomposing documents, music and videos with audio, and given the extensive management of image-based data in the WBSN field (recall Section 1), CooPeD manages photos and videos without audio and the management of other types of objects is left for future work.

## 4. CooPeD's model: $SoNeUCON_{ABC}$ extension

This Section presents the extension of $SoNeUCON_{ABC}$ in respect to both, access control (Section 4.1) and administrative (Section 4.2) management.

### *4.1. $SoNeUCON_{ABC}$ access control extension*

$SoNeUCON_{ABC}$ is extended in such a way that the entity Objects ($O$) includes an additional link to state that objects are composed of objects (Figure 2). Each object $o_i$ is decomposed in $n$ objects $o_i^j$, called components, such that $o_i = \sum_{j=1}^{n} o_i^j$. Each object $o_i$ can be represented as a tree structure where each component $o_i^j$ refers to a leaf and the whole object $o_i$ refers to the root of the tree. In this approach, for the sake of simplicity and without losing generality, objects are decomposed in objects and then, the depth of the tree is 1. However, decomposition could be recursively performed creating a tree of depth $h$ where each component $o_i^j$ at a depth $h$ can be, in turn, decomposed in other objects.

Concerning attributes, the existence if $o_i^j$ leads to the emergence of additional $ATT(O)$. Indeed, components $o_i^j$ own attributes like the type of the component. For instance, in a family photo components refer to family members and their type corresponds to *person*. In particular, an $o_i^j$ is distinguished from its parent object $o_i$ due to attached attributes. Nonetheless, each $o_i^j$ inherits attributes $att(o_i)$ from its parent object. As a final remark, analogously to $SoNeUCON_{ABC}$, $ATT(O)$ can be derived from the mapping $dAT : O \longrightarrow ATT(O)$.

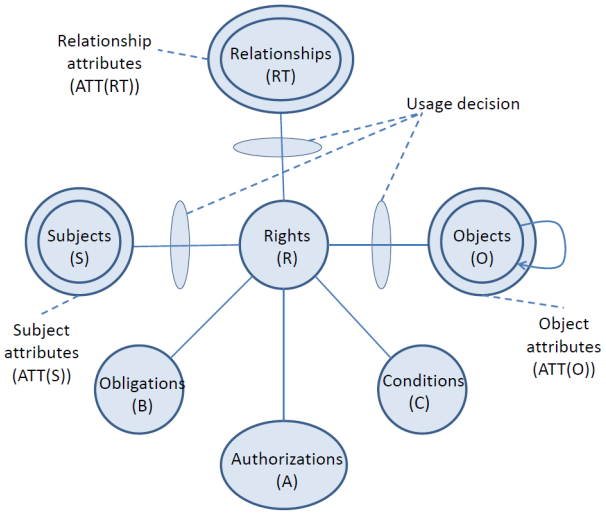



Figure 2: Extended $SoNeUCON_{ABC}$ model to support co-ownership management. Note the introduced recursive link on entity Objects (O)

It should be noticed that data WBSNs manage, specially, objects, relationships and users attributes, may be personal data. Thus, users have to be aware of this issue and trust the WBSNs in which data is left. Indeed, despite being out of the scope of this proposal, the application of mechanisms to protect data against WBSNs, e.g. cryptographic algorithms [14, 15], as well as mechanisms to evaluate access control policies avoiding disclosures of data, e.g. zero knowledge proofs [16] is a challenging matter to notice.

#### *4.1.1. Access control policies specification*

In terms of access control policies, their structure remains as $\rho(\rho_s; \rho_o; \rho_{rt}; r; \partial_b; \partial_c)$ (see Section 2). The main change is that $\rho_o$ can involve additional $ATT(O)$. Indeed, the use of these attributes helps to reach fine-grained access control policies particularly when co-ownership management takes place. For instance, the $att(o)$ $partType$ would help to determine the precise type of a component $o_i^j$ to which the policy applies. Given the variety of types of parts, if $partType$ takes the value $person$, it means that the $o_i^j$ to whom it is attached refers to the image of the owner/ co-owner himself. Similarly, if $partType$ takes value

*car*, it states that the $o_i^j$ to whom it is attached corresponds to the owner's/ co-owner's car.

Recalling the example proposed in Figure 1, it depicts an object $o_i$ which is decomposed in four parts $\{B, P_1, P_2, P_3\}$ where $B$ refers to the background. Based on Figure 1 proposed access control policies are pointed out below. All users specify *partType* to reach fine-granularity and guarantee that their established policies are enforced when $o_i^j$ come into play:

- User1(Owner:)

  $\rho_1 = ((age > 18);\ \emptyset;\ partType = Person;\ \emptyset; read;\ \emptyset;\ \emptyset)$

- User2(Co-owner$_1$):

  $\rho_1 = ((age > 24);\ \emptyset;\ partType = Person;\ \emptyset; read;\ \emptyset;\ \emptyset)$

- User3(Co-owner$_2$):

  $\rho_1 = (\emptyset;\ \emptyset;\ partType = car;\ ((((role = friend))), \emptyset,\ \emptyset);\ read;\ \emptyset;\ \emptyset)$

  $\rho_2 = ((age > 20);\ \emptyset;\ partType = car;\ \emptyset; read;\ \emptyset;\ \emptyset)$

### *4.2. SoNeUCON$_{ABC}$ administration extension*

WBSN users own use Rights (R) and Administrative Rights (AR). Use rights R refer to operations that can be performed with objects, e.g. read, move, copy, etc. Then, WBSN users who own R are able to request the right to access, copy, write, etc. objects. On the contrary, administrative rights AR refer to operations focused on managing objects, e.g. decomposing objects, defining access control policies, updating attributes, etc. In particular, owning AR involves the management of administrative objects ($AO$), the decomposition of objects, the delegation of R and AR and the revocation of R. Administrative objects ($AO$) are elements involved in the access control enforcement process. They refer to subjects $S$ (i.e. WBSN users), and their attributes $ATT(S)$ (e.g. age), objects $O$ (e.g. photos) and their attributes $ATT(O)$ (e.g. title), direct relationships $E$ (i.e. a direct contact) and their attributes $ATT(E)$ (e.g. role of friendship) and access control policies $ACP$ (e.g. grant access to cousins to

photos titled family). Depicted in Figure 3, owners grant use rights $R$ over objects $O$ regarding policies $ACP$ and execute administrative rights $AR$ over administrative objects $AO$.

In the following Sections administrative objects $AO$ (Section 4.2.1), the decomposition of objects (Section 4.2.2), the delegation of R and AR (Section 4.2.3) and the revocation of R (Section 4.2.4) are described.

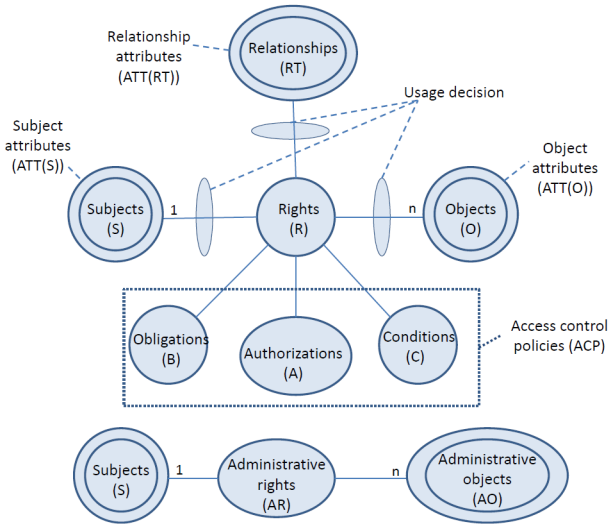



Figure 3: Administrative management in the extended version of $SoNeUCON_{ABC}$

Three issues should be noticed. Firstly, the background of an image is always managed by owners. Even though no other $o_i^j$ were attached to the owner, he would be in the possession of the background to delegate R over it. Secondly, in case a given $o_i^j$ could not be assigned to any user, e.g. the related user is not a WBSN user, it would be attached and managed by the owner as well. Lastly, it should be pointed out that administrative objects ($AO$) involve decomposable objects and then, for an object $o$ each $o_i^j$ is also considered an administrative object.

*4.2.1. Administrative objects management*

The management of administrative objects $AO$ is based on the creation, the modification and the deletion of owned elements (see Figure 4). Specifically, as mentioned above, $AO$ consists of $S$, $ATT(S)$, $O$, $ATT(O)$, $E$, $ATT(E)$ and $ACP$.

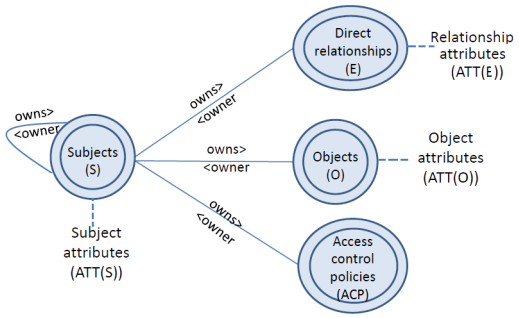



Figure 4: Administrative objects ($AO$) management

Concerning $S$, the loop point out in this entity (Figure 4) means that users can create an account in every WBSN in which they want to be enrolled and then, they become owners of each established access control policy and uploaded object. Likewise, WBSN users may cancel the account at any time. In terms of $ATT(S)$, the attributes of the user associated with an account can be established by its owner, retrieved from an identity provider where they may have been previously stored or obtained from personal devices, e.g. identity cards may store the value of attribute *birth data*.

In relation to $O$, owners upload objects, referred to as resources (eg. photos), to WBSNs. In centralized WBSNs like Facebook these objects are stored in data bases owned by the WBSNs themselves. By contrast, decentralized WBSNs like Diaspora allow the storage of objects in chosen hosts. Moreover, uploaded objects should be deleted whenever desired. Regarding $ATT(O)$, the attributes of the objects owned by a user can be defined by this user, e.g. attribute *title*, or obtained from objects metadata, e.g. *location*. However, if required, owners have to allow WBSNs to process objects metadata and automatically establish

attributes values.

On the other hand, regarding $E$, users can create, update or delete direct relationships (with one edge starting at themselves and ending at other user) and the attribute values of these relationships $ATT(E)$. Besides, $ATT(E)$ are considered identity data and then, they can be stored and subsequently retrieved, from identity providers.

Lastly, owners can create, update or delete access control policies $ACP$ which requires the specification of elements involved in them.

As a final remark, it is noticeable that, though $ATT(S)$, $ATT(O)$ and $ATT(E)$ are open sets of attributes, their use is bounded by WBSNs, because only those which are supported can be applied. It is the same case as with $\partial_b$ and $\partial_c$ involved in $ACP$, their use depends on available options.

#### *4.2.2. Decomposition management*

Each object $o_i$ is uploaded by a user, the owner, who initially owns R and AR over it. Then, if $o_i$ can be decomposed in $o_i^j$, the owner (or the WBSN on his behalf) decomposes it and assigns a co-owner to each $o_i^j$. As a result, co-owners own use rights R and administrative rights AR over their $o_i^j$.

#### *4.2.3. Delegation management*

Delegation focuses on giving permissions over an object to other users for a period of time or permanently. Delegating R can be compared with the establishment of access control control policies, if users who request a particular $r \in R$ over an object satisfy established policies, $r$ is granted.

On the contrary, the delegation of AR requires the definition of the following function:

- DELEGATE($v_k$,$v_j$,$o_i^j$,$\lambda$): It states that the user $v_k$ gives an specific AR $\lambda$ to a user $v_j$ over $o_i^j$. $\lambda$ refers to a partial delegation, to delegate some AR, or a complete delegation, to delegate all AR and change the owner of the delegated object. In case of complete delegation, $\lambda$ takes the value $*$.

This operation can be executed manually or automatically. The owner of an $o_i$ can manually identify $o_i^j$ and delegate administrative rights AR accordingly. By contrast, a WBSN can automatically detect users linked to $o_i^j$ and enforce the delegation on the owner's behalf. Moreover, both the WBSN and the owner are assumed to be trusted to execute this operation.

Note that, from a privacy point of view, the permanent delegation of AR is an essential requirement because it is assumed that each $o_i^j$ should be always managed by its owner/ co-owner. Thus, though the operation DELEGATE may consider temporal delegations in the future work, co-ownership management should preferably apply permanent delegations.

#### *4.2.4. Revocation management*

Revocation undoes the effect of delegation. In other words, it is the operation that undoes the granting of a right over an object to a user. Two revocation types are distinguished, weak and strong [17]. The former refers to the removal of granted rights over an object to users to whom rights were granted. By contrast, the latter, strong revocation, refers to recursively revoke permissions from users to whom rights were recursively granted. For instance, $\text{User}_A$ delegates the right to access an object $o_1^1$ to $\text{User}_B$ and $\text{User}_B$ delegates the right to access $o_1^1$ to $\text{User}_C$. In this scenario, if $\text{User}_A$ enforces weak revocation the right to access $o_1^1$ would be exclusively revoked from $\text{User}_B$. By contrast, in case $\text{User}_A$ enforces strong revocation, the right would be revoked from both users, $\text{User}_B$ and $\text{User}_C$. It should be noticed that, in this approach, given that each $o_i$ is decomposed in $o_i^j$, just weak revocation of use rights R is managed because recursive delegations are not applied and administrative rights AR are permanently delegated.

Specifically, revocation is the result of the modification, either of attributes or policies. For instance, assuming that objects entitled "work" are accessible to co-workers, this policy holds until the title of objects entitled "work" changes and then, access to co-workers is revoked when the title changes.

## 5. CooPeD's mechanism

CooPeD's mechanism focuses on the decomposition of objects in parts, components, the manual or automatic assignment of components to the owner and co-owners who individually manage their part/s and the enforcement of access control policies. More specifically, when $r \in R$ over $o_i \in O$ is requested and $o_i$ is composed of components $o_i^j$, co-ownership management starts. In general, noticing that developed functions are pointed out in brackets, access control enforcement starts identifying components $o_i^j$ of the requested object $o_i$ that belong to the owner [$FindObjects$] and policies attached to this user [$FindSubjectPolicies$]. Similarly, co-owners linked to the requested object $o_i$ are noticed [$FindCoOwners$] and objects parts $o_i^j$ attached to each of them are identified [$FindObjects$], as well as all of their access control policies [$FindSubjectPolicies$]. Subsequently, when components of the requested object and policies related to them are identified, the verification of access control policies is carried out analogously to $SoNeUCON_{ABC}$ [3]. Finally, objects are processed according to the owner and the co-owners access control policies, and the requested right $r$ is (or is not) granted [$ProcessObject$]. If the request matches the conditions of an access control policy of the owner and an access control policy of each co-owner, the requested right $r$ over the requested object $o_i$ is granted. By contrast, if the request does not match conditions of any policy of the owner and any policy of the co-owners, $r$ is not granted. On the other hand, if the request matches conditions of a policy of the owner or with the conditions of a policy of some co-owners, $o_i$ is processed and $r$ over appropriate $o_i^j$ is granted.

In the following, the main function ($CheckAccessCoOwner$) and the supporting ones, are described:

**CheckAccessCoOwner** The algorithm associated with this function is presented in Algorithm 1. Given a request $\{s, o, r\}$, where $s$ refers to the requester, $o$ to the requested object and $r$ to the requested right such that $o$ is decomposed in $o^j$ objects, access control enforcement consists of several tasks: 1) the owner

of $o$ is identified; 2) co-owners associated with $o$ are noticed; 3) the owner's policies are retrieved (*policiesOwner*); 4) $o^j$ attached to the owner are identified; 5)*policiesOwner* are evaluated per each $o^j$ attached to the owner, if one of them matches the identity of the related $o^j$ is stored (*listO*); 6) each co-owner's policies are retrieved (*policiesCoOwner[i]*); 7) $o^j$ attached to each co-owner are identified; 8)*policiesCoOwner[i]* are evaluated per each $o^j$ attached to each co-owner and whether they match the identity of the related $o^j$ is stored in *listO*; 9) the requested object $o$ is processed according to each $o^j$ in *listO*, thus grating or denying access to each part.

**Algorithm 1:** CheckAccessCoOwner function

```
Data: s, o, r
Result: ObjectProcessed
begin
    owner = GetAdmin(o);
    policiesOwner[] = FindSubjectPolicies(owner);
    oOfOwner[]=FindObjects(o, owner);
    for k ← 0 to oOfOwner do
        for j ← 0 to policiesOwner do
            if CheckAccess(s, oOfOwner[k], r, policiesOwner[j]) then
                listO.add(oOfOwner[k]);
    CoOwnersId[] = FindCoOwners(o);
    for i ← 0 to CoOwnersId do
        policiesCoOwner[i][] = FindSubjectPolicies(CoOwnersId[i]);
        oOfCoOwner[i][]=FindObjects(o, CoOwnersId[i]);
        for k ← 0 to oOfCoOwner[i] do
            for j ← 0 to policiesCoOwner[i] do
                if CheckAccess(s, oOfCoOwner[i][k], r,
                policiesCoOwner[i][j]) then
                listO.add(oOfCoOwner[i][k]) ;
    ProcessObject(o,listO);
```

**FindObjects** This function returns all $o^j$ of an $o$ attached to a given user.

**FindSubjectPolicies** This function returns policies defined by a particular user.

**GetAdmin** This function identifies the owner of a given object.

**FindCoOwners** This function returns the list of the identifiers of co-owners of a given object $o$.

**CheckAccess** This function is based on evaluating, in each request, if the subject ($s$), who is the requester, is allowed to execute the requested right ($r$) over the requested object ($o$). Five elements are verified. First, it is verified that attributes of $s$ match the set of subject attributes of the policy ($\rho_s$). Second, it is verified that attributes of $o$ match the set of object and objects attributes of the policy ($\rho_o$). Third, it is verified that attributes of the structure between the requester and the administrator match the set of the relationship attributes of the policy ($\rho_{rt}$). Fourth, the match between the right within the policy and $r$ is verified. Lastly, a pair of elements, conditions and obligations, are verified. For more details see [3].

**ProcessObject** This function processes the requested object ($o$) in such a way that requested right is denied over object parts ($listO$) whose related policies have not been satisfied. As a result, $o$ is returned appropriately processed.

## 6. Architectures for CooPeD

CooPeD consists of evaluating multiple policies of different users per request. Then, architectures to perform its deployment should focus on access control enforcement management. In particular, the retrieval and evaluation of policies and the later retrieval of requested data (if required) are the issues that directly affect the practical development of CooPeD's mechanism. In this regard elements involved in access control enforcement are the following ones:

- Reference Monitor (RM): it is the core part of access control management architectures. It consists of two elements, the **Policy Decision Point**

(PDP) and the **Policy Enforcement Point** (PEP). They are standardized with the X.812 access control framework (ITU-T, 1995) [18]. The PDP provides affirmative or negative responses in regard to the requested rights on a particular data according to defined policies. The PEP enforces decisions taken by the PDP.

- Data Management Module (DMM): it stores and allows the management of data. It interacts with the PEP to deliver requested data to it.
- Policy Management Module (PMM): it stores and allows the management of access control policies. It interacts with the PDP to deliver requested policies to it.

According to these elements, the Temporal Workload (TW) of requesting an object in CooPeD is calculated as the sum of the time that involves the retrieval of data, the retrieval of policies and the evaluation of policies. It is formally defined by the equation:

$TW = \sum_{i=1}^{N} \alpha_i + \sum_{i=1,j=1}^{N,M} \beta_{ij} + \sum_{i=1}^{N} \gamma_i$,
being $\alpha_i$ the time the PEP takes for retrieving parts of requested data (if policies satisfied) considering that parts are related to N users (the owner and co-owners); $\beta_{ij}$ the time the PDP takes for evaluating M policies of N users; and $\gamma_i$ the time the PDP takes for retrieving policies of all users N. Note that the focus on this proposal is to evaluate policy enforcement and thus, the time of sending a request to the server and receiving the response is neglected.

Three different architectures are distinguished, namely, centralized, partially decentralized and fully decentralized. Temporal workload is denoted by $TW_C$, $TW_P$ and $TW_F$ respectively. Furthermore, in partially and fully decentralized architectures the retrieval of data and policies, as well as the evaluation of policies can be sequential or parallel. Then, when the retrieval of data and policies and the evaluation of policies is sequential the temporal workload is denoted by $TW_{P_S}$ or $TW_{F_S}$; when the retrieval of data and policies is parallel the temporal

workload is denoted by $TW_{P_{RDP}}$ or $TW_{F_{RDP}}$; and when the retrieval of data and the evaluation of policies is parallel the temporal workload is denoted by $TW_{P_{RDEP}}$ or $TW_{F_{RDEP}}$. Note that the retrieval of policies and their evaluation cannot be paralleled because policies need to be first retrieved for the subsequent evaluation. A summary of proposed notation is depicted in Table 1.

Table 1: Temporal workload notation summary

| | Centralized | Partially decentralized | Fully decentralized |
|---|---|---|---|
| Sequential | $TW_C$ | $TW_{P_S}$ | $TW_{F_S}$ |
| Parallel retrieval of data and policies | | $TW_{P_{RDP}}$ | $TW_{F_{RDP}}$ |
| Parallel retrieval of data and evaluation of policies | | $TW_{P_{RDEP}}$ | $TW_{F_{RDEP}}$ |

The following Sections describe centralized (Section 6.1), partially decentralized (Section 6.2) and fully decentralized (Section 6.3) architectures, as well as examples of their application. Some other settings could have been devised but the described ones are representative.

### *6.1. Centralized architectures*

Centralized architectures refer to the one in which policies and data are stored in the same host and access control enforcement is also performed in such host, see Figure 5. In this architecture $\alpha$ and $\gamma$ are considered negligible because all elements are within the same host. Therefore, the policy enforcement process is accelerated and $TW_C = \sum_{i=1,j=1}^{N,M} \beta_{ij}$.

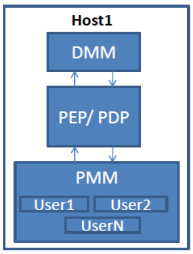



Figure 5: Centralized architecture

In the context of co-ownership management this architecture is the most used. Specifically, all studied approaches related to co-ownership management, except for [19], assume the use of a centralized architecture (see Section 10.1).

### *6.2. Partially decentralized architectures*

Partial decentralization, depicted in Figure 6, considers the decentralization of data and policies and the access control enforcement process. In particular, three hosts are at stake, each of them applied for the management of the DMM, the PMM and the RM respectively. In what concerns the TW, its calculation considers all possible elements but it depends on the parallel or sequential evaluation of policies $\beta$ and retrieval of data $\alpha$ and access control policies $\gamma$. Besides, due to efficiency reasons it is assumed that all policies and all objects parts are retrieved in one run per request. Then, $\alpha$ and $\gamma$ are not the result of a summation but a single value that refers to the time of retrieving all policies and object parts at once. Evaluating policies and retrieving data and policies sequentially $TW_{P_S} = \alpha + \sum_{i=1,j=1}^{N,M} \beta_{ij} + \gamma$. By contrast, applying parallelism $TW_{P_{RDP}} = \sum_{i=1,j=1}^{N,M} \beta_{ij} + max(\gamma, \alpha)$ and $TW_{P_{RDEP}} = max(\alpha, \sum_{i=1,j=1}^{N,M} \beta_{ij}) + \gamma$, where $max()$ means the maximum between all possible values. Note that in settings where the RM is located in the same host as the DMM or the PMM $\alpha$ or $\gamma$ are considered negligible. Besides, note that the access to data can be avoided when policies are not satisfied.

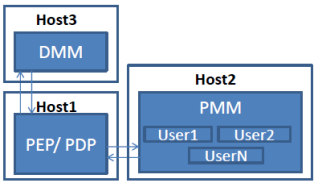



Figure 6: Partially decentralized architecture

Partially decentralized architectures are applied in many proposals in spite of not being focused on co-ownership management. For instance, [20] presents an architecture similar to the one described in this Section (Figure 6). Furthermore, other approaches propose partially decentralized architectures with some

variants regarding the one described herein. Assuming that policies (the PMM) are located in one host and the data (the DMM) and the RM in another one, M. Conti et al. propose the storage of fake data in WBSNs and the interchange of XML files (policies) between WBSN users to retrieve the real data [21].

### *6.3. Fully decentralized architectures*

Full decentralization consists of the independent management of each user's policies and data (object parts). Thus, a fully decentralized architecture involves as many hosts as users (the owner and co-owners) in addition to one in charge of the enforcement process, see Figure 7. Similar to partially decentralized architectures, sequential or parallel evaluation of policies and retrieval of data and policies can be performed. Assuming sequentiality $TW_{F_S} = \sum_{i=1}^{N} \alpha_i + \sum_{i=1,j=1}^{N,M} \beta_{ij} + \sum_{i=1}^{N} \gamma_i$. By contrast, working in parallel a significant decrease of the $TW$ is achieved, that is $TW_{F_{RDP}} = max(\sum_{i=1,j=1}^{N,M} \beta_{ij}) + max(\sum_{i=1}^{N} \gamma_i, \sum_{i=1}^{N} \alpha_i)$ and $TW_{F_{RDEP}} = max(\sum_{i=1}^{N} \alpha_i, \sum_{i=1,j=1}^{N,M} \beta_{ij}) + max(\sum_{i=1}^{N} \gamma_i)$. The use of parallelism is specially appropriate in decentralized scenarios because the sequential retrieval of data and policies and the later evaluation of policies penalizes efficiency to a large extent. Indeed, parallelism should be applied to take advantage of benefits that full decentralization provides.

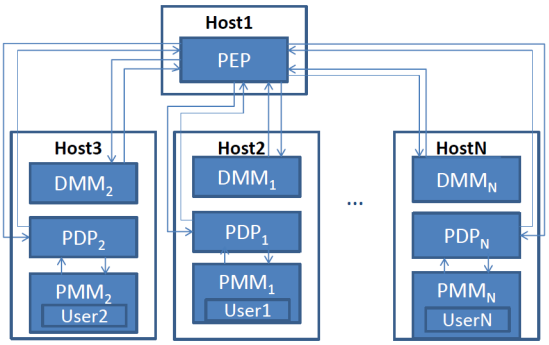



Figure 7: Fully decentralized architecture.

In sum, despite the variety of architectures, a fully decentralized one which applies parallelism may successfully contribute to the reduction of the access

control enforcement temporal workload. In what concerns the application of fully decentralized architectures, [19] is the only work which, from a theoretical point of view, makes use of them. It presents a collaborative access control model where policies are evaluated at each user host preventing a third party from knowing policies. Specifically, it is based on architectures where each user has a PDP, thereby facilitating the enforcement of parallel evaluation of policies. On the other hand, although not based on co-ownership, [20, 22] focus on data and policies decentralization and then, they are candidates for using this type of architecture. Furthermore [23] presents an architecture where policies and data are stored in each user's host and access control enforcement applies cryptographic procedures.

## 7. Feasibility analysis of CooPeD

CooPeD focuses on co-ownership and its feasibility is assessed through the identification of the number of supported co-owners. The main point is the study of policy enforcement, concluding the amount of policies that can be evaluated per object request. For the sake of simplicity and based on current WBSNs where each object is attached to a single access control policy, it is assumed that each owner/ co-owner establishes one policy per object. Then, supported co-owners are analogous to the amount of policies to evaluate which means that the calculation of temporal workload considers $\sum_{i=1}^{N} \beta_i$ instead of $\sum_{i=1,j=1}^{N,M} \beta_{ij}$.

Besides, successful WBSNs like Facebook or Flickr allow, per object, the specification of 50 and 75 tags respectively[5] [6] and the average number of tags that users establish per object[7] is 14. As a result, the TW of evaluating 1, 5, 14, 25, 50 and 75 policies is measured.

Under this perspective the scalability of the approach in every proposed architecture is analysed. Section 7.1 studies the temporal workload of pol-

[5] https://www.facebook.com/help/217258071632275–50, last access March 2014
[6] http://www.flickr.com/help/tags/, last access March 2014
[7] http://www.flickr.com/photos/mariannabolognesi/7073104431/, last access March 2014

icy enforcement for CooPeD system. Section 7.2 studies policy enforcement of CooPeD's mechanism in Facebook for being one of the most used WBSNs. Finally, Section 7.3 summarizes the policy enforcement analysis of CooPeD system and CooPeD's mechanism in Facebook.

### *7.1. CooPeD's system analysis*

The underlying model of CooPeD is $SoNeUCON_{ABC}$ extension and then, the feasibility of implementing $SoNeUCON_{ABC}$ for co-ownership management is analysed using the same proof of $SoNeUCON_{ABC}$ [3]. This proof of concept involves 4 random WBSN structures. Table 2 depicts the number of nodes ($\#v_i$), the number of relationships ($\#e_i$) and the mean number of relationships per node ($\overline{e_i/v_i}$) that each WBSN involves.

Table 2: WBSNs structure

| **WBSNs id** | $\#e_i$ | $\#v_i$ | $\overline{e_i/v_i}$ |
|---|---|---|---|
| **1** | 2,980,388 | 50,000 | 60 |
| **2** | 5,965,777 | 50,000 | 120 |
| **3** | 8,949,375 | 50,000 | 185 |
| **4** | 10,929,713 | 50,000 | 219 |

Concerning technical details, the proof of concept system is developed in Java 1.7, using a MySQL 5.2 database to store nodes and relationships. Moreover, experiments have been executed over a Pentium D 2.3 GHz with a Lion 10.8 operating system using 500 MB of RAM.

In the following Sections are presented, first, proposed access control policies to evaluate (Section 7.1.1) and second, a temporal workload analysis regarding proposed policies enforcement (Section 7.1.2).

#### *7.1.1. Proposed access control policies*

In $SoNeUCON_{ABC}$ access control policies manage the six features mentioned in Section 2. Based on these features the following policies are proposed:

P1 Access is granted to users who are friends of neighbours of his/ her relatives if the relationship between his/ her relatives and his/ her relatives' neighbours was established before 2,000.

P2 Access is granted to users who have three friends in common with the administrator of the requested object.

P3 Access is granted to users who belong to the clique in which two users and the administrator of the requested object are involved, having all of them a friendship relationship.

P4 Access is granted to users who are connected to the administrator by two different paths composed of unidirectional relationships oriented from the requester to the administrator. Moreover, relationships involved in all paths have to be highly trusted.

P5 Access is granted to users who are friends of the administrator of the requested object, also having a bidirectional relationship with him/ her.

P6 Access is granted to users who are friends of the administrator of the requested object.

P7 Access is granted to users if they are females under 30 years old or if they are females under 40 who have studied computer science or if they are females who have studied computer science and physics.

*7.1.2. Policy enforcement analysis*

For each created WBSN 7 random requesters and administrators are chosen and the enforcement of proposed policies is performed (*id*). Table 3 presents the number of relationships ($\#e_i$ *explo.*) and nodes explored ($\#v_i$ *explo.*) while evaluating proposed access control policies.

Despite the amount of possibilities to perform this study and on the bases of popular WBSNs like Facebook, in this proposal the analysis of policy enforcement is limited to WBSNs where users are associated with direct relationships and indirect relationships of length 2. Concerning the enforcement of policies in regard to WBSNs with direct relationships, their evaluation cannot involve the exploration of more than 190 nodes because this is the average number of contacts per user in a WBSN like Facebook [24]. Similarly, the evaluation of policies in WBSNs with indirect relationship of length 2 cannot exceed the exploration of 36,292 ($190+190^2+2$) nodes. Therefore, based on Table 3, the TW

Table 3: Nodes and relationships while policies evaluation.

| id | # ei explo. | # vi explo. | id | # ei explo. | # vi explo. |
|---|---|---|---|---|---|
| WBSN id = 1 | | | WBSN id = 2 | | |
| 1 | 209,041 | 418,082 | 8 | 1,958,163 | 3,916,326 |
| 2 | 3,544 | 7,088 | 9 | 13,557 | **27,114** |
| 3 | 63 | **126** | 10 | 139 | 278 |
| 4 | 54 | 108 | 11 | 126 | **252** |
| 5 | 57 | 114 | 12 | 119 | 238 |
| 6 | 65 | 130 | 13 | 115 | 230 |
| 7 | 1 | 2 | 14 | 1 | 2 |
| WBSN id = 3 | | | WBSN id = 4 | | |
| 15 | 6,163,496 | 12,326,996 | 22 | 11,115,845 | 22,231,690 |
| 16 | 29,771 | **49,542** | 23 | 445,839 | 91,678 |
| 17 | 201 | 402 | 24 | 244 | 488 |
| 18 | 187 | 374 | 25 | 230 | 460 |
| 19 | 174 | 348 | 26 | 216 | 432 |
| 20 | 174 | 348 | 27 | 216 | 432 |
| 21 | 37 | 74 | 28 | 33 | 66 |

of evaluating 1, 5, 14, 25, 50 and 75 policies of each different type (P1-P7) is performed in respect to the average between $id = 3$ and $id = 11$ because the amount of explored nodes ($\frac{126+252}{2} = 189$) is close to 190 and in regard to the average between $id = 9$ and $id = 16$ as explored nodes are close to the average ($\frac{27,114+49,542}{2} = 38,328$).

Based on aforementioned points, the analysis of CooPeD's system policy enforcement in centralized, partially decentralized and fully decentralized architectures is presented below.

*Policy enforcement in a centralized architecture.* Firstly, policy enforcement when all evaluated policies are of the same type is studied, thus $TW_C = \sum_{i=1}^{N} \beta_i = N \cdot \beta$. Considering that the tolerable waiting time of WBSN users for information retrieval is approximately 2,000 ms [25], it can be calculated from Table 4 that in a WBSN with direct relationships excluding P3 (which defines a clique), 34 policies of the same type (P1, P2, P4, P5, P6 or P7) can be evaluated per object without exceeding this limit, that is $TW_C$=58·34=1,972 ms for sets of policies P1, P2, P6 or P7 and $TW_C$=58.5·34=1,989 ms for sets of policies P4 and P5. On the contrary, when a WBSN with indirect relationships is at stake, 4 policies of the same type can be evaluated per object, except for

P3, without exceeding 2,000 ms, that is $TW_C$=493·4=1,972 ms for sets of policies P1, P4 or P5 and $TW_C$=492.5·4=1,970 ms for sets of policies P2, P6 or P7. Moreover, regarding P3, in respect to WBSNs with direct relationships a pair of policies can be evaluated without exceeding 2,000, that is $TW_C$=750.5·2=1,501 ms. Unfortunately, just the enforcement of a single policy P3 takes 2,898.5 ms

Table 4: Average $TW_C$ policy enforcement for co-ownership management. Analogous types of policies.

| **WBSNs with direct relationships** | | | | | | |
|---|---|---|---|---|---|---|
| id | 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
| P1/ P2/ P6/ P7- TW (ms) | | | | | | |
| id=3 | 28 | 140 | 392 | 700 | 1,400 | 2,100 |
| id=11 | 88 | 440 | 1,232 | 2,200 | 4,400 | 6,600 |
| Average | 58 | 290 | 812 | 1,450 | 2,900 | 4,350 |
| P3- TW (ms) | | | | | | |
| id=3 | 275 | 1,375 | 3,850 | 6,875 | 13,750 | 20,625 |
| id=11 | 1,226 | 6,130 | 17,164 | 30,650 | 61,300 | 91,950 |
| Average | 750.5 | 3,752.5 | 10,507 | 18,762.5 | 37,525 | 56,287.5 |
| P4/ P5- TW (ms) | | | | | | |
| id=3 | 28 | 140 | 392 | 700 | 1,400 | 2,100 |
| id=11 | 89 | 445 | 1,246 | 2,225 | 4,450 | 6,675 |
| Average | 58.5 | 292.5 | 819 | 1,462.5 | 2,925 | 4,387.5 |
| **WBSNs with indirect relationships (length 2)** | | | | | | |
| id | 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
| P1/ P4/ P5- TW (ms) | | | | | | |
| id=9 | 712 | 3,560 | 9,968 | 17,800 | 35,600 | 53,400 |
| id=16 | 274 | 1,370 | 3,836 | 6,850 | 13,700 | 20,550 |
| Average | 493 | 2,465 | 6,902 | 12,325 | 24,650 | 36,975 |
| P3-TW (ms) | | | | | | |
| id=9 | 2,498 | 12,490 | 34,972 | 62,450 | 124,900 | 187,350 |
| id=16 | 3,299 | 16,495 | 46,186 | 82,475 | 164,950 | 247,425 |
| Average | 2,898.5 | 14,492.5 | 40,579 | 72,462.5 | 144,925 | 217,387.5 |
| P2/ P6/ P7- TW (ms) | | | | | | |
| id=9 | 712 | 3,560 | 9,968 | 17,800 | 35,600 | 53,400 |
| id=16 | 273 | 1,365 | 3,822 | 6,825 | 13,650 | 20,475 |
| Average | 492.5 | 2,462.5 | 6,895 | 12,312.5 | 24,625 | 36,937.5 |

Secondly, the enforcement of policies of different types is studied, thereby

applying the equation $TW_C = \sum_{i=1}^{N} \beta_i$. Depicted in Table 5, it is noticed that the $TW_C$ of evaluating sets of different types of policies does not significantly differ from the evaluation of sets of policies of the same type. It is estimated that, on average, the enforcement of a policy P1-P7 distinct from P3 takes 58.17 ms for WBSNs with direct relationships and the enforcement of 34 policies of any type (no P3) without exceeding 2,000 ms is possible. For WBSNs with indirect relationships, the enforcement of a policy P1-P7 distinct from P3 takes 492.75 ms and the enforcement of 4 policies of any type is possible. On the other hand, when cliques management comes into play, that is, the evaluation of P3, the maximum waiting time for information retrieval is not exceeded for WBSNs with direct relationships applying a policy P3 and 21 policies of other types, as well as applying a pair of policies P3 and 8 policies of other types.

Table 5: Average $TW_C$ policy enforcement for co-ownership management. Different types of policies.

| **WBSNs with direct relationships** | | | | | |
|---|---|---|---|---|---|
| No matter type (P1-P7, no P3) - Avg. TW(ms) | | | | | |
| 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
| 58.17 | 290.83 | 814.33 | 1,454.17 | 2,908.33 | 4,362.5 |
| P3 + other types - Avg. TW(ms) | | | | | |
| 1 policy P3 + 21 of others | 2 policy P3 + 8 of others | 3 policy P3 | | | |
| 1,972 | 1,966.33 | 2,251.5 | | | |
| **WBSNs with indirect relationships (length 2)** | | | | | |
| No matter type (P1-P7, no P3) - Avg. TW(ms) | | | | | |
| 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
| 492.75 | 2,463.75 | 6,898.5 | 12,318.75 | 24,637.5 | 36,956.25 |
| P3 - TW(ms) | | | | | |
| 1 policy | | | | | |
| 2,898.5 | | | | | |

*Policy enforcement in partially decentralized architectures.* Based on centralized architectures, the policy enforcement temporal workload for partially decentralized architectures is estimated. Recalling that different values for the sequential or parallel retrieval of data ($\alpha$) and policies ($\gamma$) are applied, they are established by measuring the temporal workload to access to a friend's profile in

Facebook. The Firebug 1.12.8 browser extension[8] is applied for measuring this temporal workload which refers to the time the client (browser) is waiting for the server response. In this time period it is assumed that the server retrieves ($\gamma$) and evaluates ($\beta$) the right access control policy, as well as retrieves the requested data ($\alpha$). After 20 repetitions in which cache has been cleaned each time and separated by several minutes (to promote invalidation of potential intermediate caches) the average measured temporal workload is 170 ms. In this regard, assuming that Facebook applies a partially decentralized architecture it is estimated that $TW_{P_S} = \alpha + \beta + \gamma = 170$ ms in a sequential scenario and $TW_{P_{RDP}} = \beta + max(\gamma, \alpha) = 170$ ms and $TW_{P_{RDEP}} = max(\alpha, \beta) + \gamma = 170$ ms in case of parallelism. Note that in Facebook $\beta$ is not a summation but herein $\sum_{i=1}^{N} \beta_i$ is applied.

The following step is the concrete specification of values for $\alpha$, $\beta$ and $\gamma$. Given results from Table 5 it is established that $\beta$=58.17 ms for P1-P7 (no P3) and $\beta$=750.5 ms for P3 for WBSNs with direct relationship and $\beta$=492.75 ms for P1-P7 (no P3) and $\beta$=2898.5 ms for P3 for WBSNs with indirect relationships. $\beta$=58.17 ms is the only one that can be applied to calculate values for $\alpha$ and $\gamma$ because the rest of them exceed 170 ms. Then, assuming that $\alpha+\gamma < 170-58.17$, extreme and intermediate values for $\alpha$ and $\gamma$ are selected through empirical testing. Specifically $\alpha$=25 and $\gamma$=86.83; $\alpha$=111.83 and $\gamma$=0; and $\alpha$=0 and $\gamma$=111.83 are the chosen values. Tables 6 and 7 depict results of the analysis of policy enforcement in partially decentralized architectures.

Results show that the $TW_P$ of evaluating policies P1-P7 (no P3) in WBSNs with direct relationships does not exceed 2,000 ms for all established settings, allowing the evaluation of between 32 and 34 policies P1-P7 (excluding P3) per request (see Table 6). In fact, when $\alpha$=25 and $\gamma$=86.83, 32 policies are evaluated applying sequential retrieval of data and retrieval and evaluation of policies and 33-32 policies applying a parallel one; when $\alpha$=111.83 and $\gamma$=0, 32 policies applying sequential and parallel evaluation of policies and retrieval of

[8] https://addons.mozilla.org/es/firefox/addon/firebug/ , lass access May 2014

Table 6: $TW_P$ policy enforcement for co-ownership management. WBSNs with direct relationships.

| | | 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
|---|---|---|---|---|---|---|---|
| | | | | No matter type (P1-P7, no P3) | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=58.17 | $TW_{P_S}$ | 170 | 402,66 | 926,16 | 1566 | 3020,16 | 4474,33 |
| | $TW_{P_{RDP}}$ | 83,17 | 315,83 | 839,33 | 1479,17 | 2933,33 | 4387,5 |
| | $TW_{P_{RDEP}}$ | 145 | 377,66 | 901,16 | 1541 | 2995,16 | 4449,33 |
| | | | | P3 | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=750.5 | $TW_{P_S}$ | 862,33 | 3864,33 | 10618,83 | 18874,33 | 37636,83 | 56399,33 |
| | $TW_{P_{RDP}}$ | 775,5 | 3777,5 | 10532 | 18787,5 | 37550 | 56312,5 |
| | $TW_{P_{RDEP}}$ | 837,33 | 3839,33 | 10593,83 | 18849,33 | 37611,83 | 56374,33 |
| | | | | No matter type (P1-P7, no P3) | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=58.17 | $TW_{P_S}$ | 170 | 402,66 | 926,16 | 1566 | 3020,16 | 4474,33 |
| | $TW_{P_{RDP}}$ | 170 | 402,66 | 926,16 | 1566 | 3020,16 | 4474,33 |
| | $TW_{P_{RDEP}}$ | 111,83 | 290,83 | 814,33 | 1454,17 | 2908,33 | 4362,5 |
| | | | | P3 | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=750.5 | $TW_{P_S}$ | 862,33 | 3864,33 | 10618,83 | 18874,33 | 37636,83 | 56399,33 |
| | $TW_{P_{RDP}}$ | 862,33 | 3864,33 | 10618,83 | 18874,33 | 37636,83 | 56399,33 |
| | $TW_{P_{RDEP}}$ | 750,5 | 3752,5 | 10507 | 18762,5 | 37525 | 56287,5 |
| | | | | No matter type (P1-P7, no P3) | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=58.17 | $TW_{P_S}$ | 170 | 402,66 | 926,16 | 1566 | 3020,16 | 4474,33 |
| | $TW_{P_{RDP}}$ | 58,17 | 290,83 | 814,33 | 1454,17 | 2908,33 | 4362,5 |
| | $TW_{P_{RDEP}}$ | 170 | 402,66 | 926,16 | 1566 | 3020,16 | 4474,33 |
| | | | | P3 | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=750.5 | $TW_{P_S}$ | 862,33 | 3864,33 | 10618,83 | 18874,33 | 37636,83 | 56399,33 |
| | $TW_{P_{RDP}}$ | 750,5 | 3752,5 | 10507 | 18762,5 | 37525 | 56287,5 |
| | $TW_{P_{RDEP}}$ | 862,33 | 3864,33 | 10618,83 | 18874,33 | 37636,83 | 56399,33 |

Table 7: $TW_P$ policy enforcement for co-ownership management. WBSNs with indirect relationships (length 2).

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=492.75 | $TW_{P_S}$ | 604,58 | 2575,58 | 7010,33 | 12430,58 | 24749,33 | 37068,08 |
| | $TW_{P_{RDP}}$ | 517,75 | 2488,75 | 6923,5 | 12343,75 | 24662,5 | 36981,25 |
| | $TW_{P_{RDEP}}$ | 579,58 | 2550,58 | 6985,33 | 12405,58 | 24724,33 | 37043,08 |
| | | P3 | | | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=2898.5 | $TW_{P_S}$ | 3010,33 | 14604,33 | 40690,83 | 72574,33 | 145036,83 | 217499,33 |
| | $TW_{P_{RDP}}$ | 2923,5 | 14517,5 | 40604 | 72487,5 | 144950 | 217412,5 |
| | $TW_{P_{RDEP}}$ | 2985,33 | 14579,33 | 40665,83 | 72549,33 | 145011,83 | 217474,33 |
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=492.75 | $TW_{P_S}$ | 604,58 | 2575,58 | 7010,33 | 12430,58 | 24749,33 | 37068,08 |
| | $TW_{P_{RDP}}$ | 604,58 | 2575,58 | 7010,33 | 12430,58 | 24749,33 | 37068,08 |
| | $TW_{P_{RDEP}}$ | 492,75 | 2463,75 | 6898,5 | 12318,75 | 24637,5 | 36956,25 |
| | | P3 | | | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=2898.5 | $TW_{P_S}$ | 3010,33 | 14604,33 | 40690,83 | 72574,33 | 145036,83 | 217499,33 |
| | $TW_{P_{RDP}}$ | 3010,33 | 14604,33 | 40690,83 | 72574,33 | 145036,83 | 217499,33 |
| | $TW_{P_{RDEP}}$ | 2898,5 | 14492,5 | 40579 | 72462,5 | 144925 | 217387,5 |
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=492.75 | $TW_{P_S}$ | 604,58 | 2575,58 | 7010,33 | 12430,58 | 24749,33 | 37068,08 |
| | $TW_{P_{RDP}}$ | 492,75 | 2463,75 | 6898,5 | 12318,75 | 24637,5 | 36956,25 |
| | $TW_{P_{RDEP}}$ | 604,58 | 2575,58 | 7010,33 | 12430,58 | 24749,33 | 37068,08 |
| | | P3 | | | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=2898.5 | $TW_{P_S}$ | 3010,33 | 14604,33 | 40690,83 | 72574,33 | 145036,83 | 217499,33 |
| | $TW_{P_{RDP}}$ | 2898,5 | 14492,5 | 40579 | 72462,5 | 144925 | 217387,5 |
| | $TW_{P_{RDEP}}$ | 3010,33 | 14604,33 | 40690,83 | 72574,33 | 145036,83 | 217499,33 |

data and policies; and when $\alpha$=0 and $\gamma$=111.83, 32 policies again applying a sequential scenario and 34 applying the parallel one. By contrast, just 2 policies with cliques (P3) can be evaluated without exceeding 2,000 ms regardless of the established setting.

On the other hand, results are worse when indirect relationships are at stake (see Table 7). In particular, the maximum amount of policies P1-P7 (no P3) that can be evaluated is 3 either retrieving data and policies and evaluating policies sequentially or in parallel. Nonetheless, not a single policy P3 can be evaluated under the threshold of 2,000 ms.

Table 8: $TW_F$ policy enforcement for co-ownership management. WBSNs with direct relationships.

| | | 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
|---|---|---|---|---|---|---|---|
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=58.17 | $TW_{F_S}$ | 170 | 850 | 2380 | 4250 | 8500 | 12750 |
| | $TW_{F_{RDP}}$ | 145 | 145 | 145 | 145 | 145 | 145 |
| | $TW_{F_{RDEP}}$ | 145 | 145 | 145 | 145 | 145 | 145 |
| | | P3 | | | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=750.5 | $TW_{F_S}$ | 862,33 | 4311,65 | 12072,62 | 21558,25 | 43116,5 | 64674,75 |
| | $TW_{F_{RDP}}$ | 837,33 | 837,33 | 837,33 | 837,33 | 837,33 | 837,33 |
| | $TW_{F_{RDEP}}$ | 837,33 | 837,33 | 837,33 | 837,33 | 837,33 | 837,33 |
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=58.17 | $TW_{F_S}$ | 170 | 850 | 2380 | 4250 | 8500 | 12750 |
| | $TW_{F_{RDP}}$ | 170 | 170 | 170 | 170 | 170 | 170 |
| | $TW_{F_{RDEP}}$ | 111,83 | 111,83 | 111,83 | 111,83 | 111,83 | 111,83 |
| | | P3 | | | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=750.5 | $TW_{F_S}$ | 862,33 | 4311,65 | 12072,62 | 21558,25 | 43116,5 | 64674,75 |
| | $TW_{F_{RDP}}$ | 862,33 | 862,33 | 862,33 | 862,33 | 862,33 | 862,33 |
| | $TW_{F_{RDEP}}$ | 750,5 | 750,5 | 750,5 | 750,5 | 750,5 | 750,5 |
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=58.17 | $TW_{F_S}$ | 170 | 850 | 2380 | 4250 | 8500 | 12750 |
| | $TW_{F_{RDP}}$ | 58,17 | 58,17 | 58,17 | 58,17 | 58,17 | 58,17 |
| | $TW_{F_{RDEP}}$ | 111,83 | 111,83 | 111,83 | 111,83 | 111,83 | 111,83 |
| | | P3 | | | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=750.5 | $TW_{F_S}$ | 862,33 | 4311,65 | 12072,62 | 21558,25 | 43116,5 | 64674,75 |
| | $TW_{F_{RDP}}$ | 862,33 | 862,33 | 862,33 | 862,33 | 862,33 | 862,33 |
| | $TW_{F_{RDEP}}$ | 862,33 | 862,33 | 862,33 | 862,33 | 862,33 | 862,33 |

Table 9: $TW_F$ policy enforcement for co-ownership management. WBSNs with indirect relationships (length 2).

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=492.75 | $TW_{F_S}$ | 604,58 | 3022,9 | 8464,12 | 15114,5 | 30229 | 45343,5 |
| | $TW_{F_{RDP}}$ | 579,58 | 579,58 | 579,58 | 579,58 | 579,58 | 579,58 |
| | $TW_{F_{RDEP}}$ | 579,58 | 579,58 | 579,58 | 579,58 | 579,58 | 579,58 |
| | | P3 | | | | | |
| $\alpha$=25; $\gamma$=86.83; $\beta$=2898.5 | $TW_{F_S}$ | 3010,33 | 15051,65 | 42144,62 | 75258,25 | 150516,5 | 225774,75 |
| | $TW_{F_{RDP}}$ | 2985,33 | 2985,33 | 2985,33 | 2985,33 | 2985,33 | 2985,33 |
| | $TW_{F_{RDEP}}$ | 2985,33 | 2985,33 | 2985,33 | 2985,33 | 2985,33 | 2985,33 |
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=492.75 | $TW_{F_S}$ | 3010,33 | 15051,65 | 42144,62 | 75258,25 | 150516,5 | 225774,75 |
| | $TW_{F_{RDP}}$ | 604,58 | 604,58 | 604,58 | 604,58 | 604,58 | 604,58 |
| | $TW_{F_{RDEP}}$ | 492,75 | 492,75 | 492,75 | 492,75 | 492,75 | 492,75 |
| | | P3 | | | | | |
| $\alpha$=111.83; $\gamma$=0; $\beta$=2898.5 | $TW_{F_S}$ | 3010,33 | 15051,65 | 42144,62 | 75258,25 | 150516,5 | 225774,75 |
| | $TW_{F_{RDP}}$ | 3010,33 | 3010,33 | 3010,33 | 3010,33 | 3010,33 | 3010,33 |
| | $TW_{F_{RDEP}}$ | 2898,5 | 2898,5 | 2898,5 | 2898,5 | 2898,5 | 2898,5 |
| | | No matter type (P1-P7, no P3) | | | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=492.75 | $TW_{F_S}$ | 3010,33 | 15051,65 | 42144,62 | 75258,25 | 150516,5 | 225774,75 |
| | $TW_{F_{RDP}}$ | 492,75 | 492,75 | 492,75 | 492,75 | 492,75 | 492,75 |
| | $TW_{F_{RDEP}}$ | 492,75 | 492,75 | 492,75 | 492,75 | 492,75 | 492,75 |
| | | P3 | | | | | |
| $\alpha$=0; $\gamma$=111.83; $\beta$=2898.5 | $TW_{F_S}$ | 3010,33 | 15051,65 | 42144,62 | 75258,25 | 150516,5 | 225774,75 |
| | $TW_{F_{RDP}}$ | 2898,5 | 2898,5 | 2898,5 | 2898,5 | 2898,5 | 2898,5 |
| | $TW_{F_{RDEP}}$ | 3010,33 | 3010,33 | 3010,33 | 3010,33 | 3010,33 | 3010,33 |

*Policy enforcement in fully decentralized architectures.* This analysis is performed under the same conditions as in partially decentralized architectures. Results of the analysis are depicted in Tables 8 and 9.

Concerning direct relationships 32 policies P1-P7 (no P3) can be evaluated without exceeding 2,000 for all established values for $\alpha$ and $\gamma$ and just 2 policies P3 when performing a sequential evaluation of policies and a sequential retrieval of data an policies. By contrast, in case of parallelism, regardless of the established settings an unlimited number of policies (except for policies P3) can be evaluated.

On the contrary, the use of WBSNs with indirect relationships produces less successful results in the sequential scenario. Applying sequential retrieval of data and policies and sequential evaluation of policies, 3 policies P1-P7 (no P3) can be evaluated within the established threshold. However, analogous to WBSNs with direct relationship, unlimited number of policies can be evaluated applying parallelism. Finally, already pointed out in the remaining architectures, policies P3 cannot be evaluated in less than 2,000 ms.

### *7.2. CooPeD's mechanism in Facebook: policy enforcement analysis*

Current WBSNs, e.g. Facebook, apply centralized or partially decentralized architectures, even being unknown details of their internal management. Particularly, this Section analyses $TW$ of policy enforcement when applying CooPeD's mechanism in Facebook for both types of architectures. Results are depicted in Table 10.

Being 170 ms the temporal workload of accessing a friend's profile in Facebook (previously calculated), it is estimated that $TW_C = \beta = 170$ ms, $TW_{P_S} = \alpha + \beta + \gamma = 170$ ms and $TW_{P_{RDP}} = \beta + max(\gamma, \alpha) = 170$ ms and $TW_{P_{RDEP}} = max(\alpha, \beta) + \gamma = 170$ ms. In this regard values for $\alpha$, $\beta$ and $\gamma$ are established in respect to a regular case, $\alpha$=56.67, $\beta$=56.67 and $\gamma$=56.67, and a pair of extreme cases, namely, $\alpha$=84.5, $\beta$=1 and $\gamma$=84.5 and $\alpha$=1, $\beta$=168 and $\gamma$=1.

Again establishing 2,000 ms as the threshold value, in a centralized architecture 11 policies can be evaluated per request. Furthermore, in a partially

Table 10: TW policy enforcement of CooPeD's mechanism in Facebook

| | | 1 policy | 5 policies | 14 policies | 25 policies | 50 policies | 75 policies |
|---|---|---|---|---|---|---|---|
| Centralized architectures, $TW_C$ | | | | | | | |
| | | 170 | 850 | 2380 | 4250 | 8500 | 12750 |
| Partially decentralized, $TW_P$ | | | | | | | |
| $\alpha$=56.67; $\gamma$=56.67; $\beta$=56.67 | $TW_{P_S}$ | 170,00 | 396,67 | 906,67 | 1530,00 | 2946,67 | 4363,33 |
| | $TW_{P_{RDP}}$ | 113,33 | 340,00 | 850,00 | 1473,33 | 2890,00 | 4306,67 |
| | $TW_{P_{RDEP}}$ | 226,67 | 906,67 | 2436,67 | 4306,67 | 8556,67 | 12806,67 |
| $\alpha$=84.5; $\gamma$=84.5; $\beta$=1 | $TW_{P_S}$ | 170 | 174 | 183 | 194 | 219 | 244 |
| | $TW_{P_{RDP}}$ | 85,5 | 89,5 | 98,5 | 109,5 | 134,5 | 159,5 |
| | $TW_{P_{RDEP}}$ | 169 | 169 | 169 | 169 | 169 | 169 |
| $\alpha$=1; $\gamma$=1; $\beta$=168 | $TW_{P_S}$ | 170 | 842 | 2354 | 4202 | 8402 | 12602 |
| | $TW_{P_{RDP}}$ | 169 | 841 | 2353 | 4201 | 8401 | 12601 |
| | $TW_{P_{RDEP}}$ | 169 | 841 | 2353 | 4201 | 8401 | 12601 |

decentralized architecture when $\alpha$=84.5, $\beta$=84.5 and $\gamma$=84.5, 33 and 34 policies can be evaluated for the sequential and the parallel evaluation of policies and retrieval of data and policies respectively; when $\alpha$=84.5, $\beta$=1 and $\gamma$=84.5, 1,831 policies can be evaluated in a sequential scenario and 1,915 in a parallel one; and when $\alpha$=1, $\beta$=168 and $\gamma$=1, 11 policies can be evaluated either in a sequential or in a parallel scenario.

On the other hand, to study indirect relationships the $TW$ to access a the profile of a friend of a friend is measured. After 20 repetitions this $TW$ is 170.2 ms which is considered equivalent to the $TW$ when accessing to the profile of a friend, 170 ms. This result points out that Facebook applies similar techniques to evaluate direct or indirect relationships and due to the similarity between both $TW$, conclusions are considered equivalent for direct and indirect relationships.

### *7.3. Summary: policy enforcement analysis*

Concerning CooPeD's system, inherited from $SoNeUCON_{ABC}$, cliques management involves a great amount of time [3] and then, the evaluation of P3 (that considers a clique) exceeds 2,000 ms in the majority of cases. By contrast, in WBSNs with direct relationships 34 policies P1-P7 (no P3) can be evaluated in centralized architectures and between 32 and 34 policies P1-P7 (no P3) in

decentralized ones (except for fully decentralized which apply parallelism) either applying sequential or parallel evaluation of policies and retrieval of data and policies. Furthermore, fully decentralized architectures which parallel the retrieval of data and policies and the evaluation of policies, produce better results because an unlimited number of policies can be evaluated. Nonetheless, the evaluation of unlimited number of policies can be evaluated but this number may be bounded by the time to send/ received a request to/ from the server which is not considered in this study (recall Section 6). Therefore, leaving aside cliques management and being 14 the average number of co-owners per object in a WBSN, the application of CooPeD's system is feasible in WBSNs with direct relationships.

On the other hand, worse results are achieved applying CooPeD's system in WBSNs with indirect relationships (length 2). In all architectures and settings 2 or 4 policies P1-P7 (no P3) can be evaluated per request without exceeding 2,000 ms. Again, this result is enhanced when fully decentralized architectures are applied. In sum, CooPeD's system in WBSNs with indirect relationships supports 14 co-owners when establishing a higher threshold or when applying a fully decentralized architecture. Otherwise, indirect relationships management should be enhanced.

Conclusions drawn from studying policy enforcement in CooPeD's system highlight the relevance of decentralization together with parallelism. Fully decentralized architectures have the advantage of simplifying data management and storage, thus facilitating scalability. Each user manages and stores his data and policies. WBSNs are relieved from the burden of managing huge amount of users and data, as well as they are relieved from storage matters. Besides, apart from scalability issues, this type of architectures are particularly appropriate from a security point of view because it provides users with more control over their data.

Regarding CooPeD's mechanism in Facebook, 11 policies can be evaluated per request in centralized architectures. Then, this mechanism use is quite acceptable in these architectures. Similarly, it can be also satisfactorily applied

in partially decentralized architectures where a minimum of 12 policies can be evaluated and maximum of 1,915 when the retrieval of data and policies or the retrieval of data and the evaluation of policies is paralleled. Indeed, CooPeD's mechanism is specially appropriate when $\beta<123$ because this condition guarantees the possible evaluation of more than 14 policies per request, thus supporting more than 14 co-owners per request.

## 8. CooPeD prototype

A prototype to prove the feasibility of implementing CooPeD has been developed in C#, applying a MySQL database and Emuge CV 2.2.1 to facial recognition. It consists of a web application that allows co-ownership management of photos of people (photos of cars, animals, etc. are a matter of future work). It is expected that the prototype could be linked to a popular WBSN like Facebook in the future. However, given the limitations of the Facebook's API just the Facebook authentication process and photos stored in Facebook are applied in this prototype. Therefore, the use of Facebook simplifies users' authentication management and avoids the storage of photos in an additional data base.

In the following Sections the architecture and the functionality of the prototype are described (Section 8.1 and 8.2 respectively).

### *8.1. Architecture*

CooPeD architecture, depicted in Figure 8, consists of the following elements:

- *Data bases (DDBB)*: a pair of them is distinguished. *FB data DB*, refers to the Facebook database to authenticate users and manage photos. Additionally, a *Users data & relationships + Policies&objects parts DB* stores users attributes and relationships, as well as policies and the identities of owners, co-owners and the object parts assigned to each photo.

- *Management module*: it performs administrative operations. Based on Facebook, users log into the application (*DB authentication module*). Then,

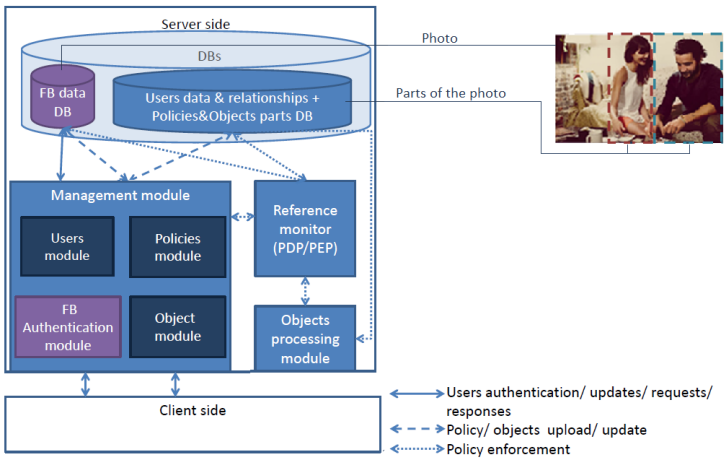



Figure 8: CooPeD prototype architecture

a set of tools allow users the definition of their data and the establishment of relationships (*Users module*), other sets of tools allows users the creation, the upload and the deletion of access control policies (*Policies module*) and another set allow the choice of objects (namely, photos), the recognition of objects parts and the link between parts and the appropriate users (*Objects module*).

- *Reference monitor*: it verifies access control policies and delivers (if required) the requested object to the *Management module* to be appropriately processed.

- *Object processing module*: it provides tools to hide objects parts of photos. After the *Reference monitor* informs about results of policy enforcement, *FB data DB* and *Policies&objects parts DB* provide the requested object and its parts respectively. Next, the object is processed and sent to the *Reference monitor*.

### *8.2. Functionality*

CooPeD offers four main functions which are the specification of users data together with the establishment of user relationships and the definition of access

control policies (depicted in Figure 9(a))), the tagging of users (depicted in Figure 9(b)) and the request of photos of direct contacts (depicted in Figure 9(c)). These functions are described in the following Sections.

*8.2.1. Account creation and relationships specification*

Enrolling in CooPeD requires authentication with Facebook. Users have to specify their age and college because this data is applied in access control policies. Afterwards, at any time, users can create relationships with other CooPeD users specifying the role of the relationship (e.g. friend, relative, etc.) and the level of trust (e.g. 1-low trusted, 10-high trusted). Note that instead of using Facebook contacts, the establishment of relationships is necessary because access control policies may involve the relationship attributes role and/ or trust. Contrary to WBSNs like Facebook, CooPeD bases on $SoNeUCON_{ABC}$ and then, access control policies consist of attributes management. This function is presented in Figure 9(a).

*8.2.2. Policies specification*

Based on Section 4.1.1, owners and co-owners establish access control policies to delegate R. Access control policies can be extremely assorted and they are established by all users of CooPeD. In this prototype, based on a simplified version of $SoNeCON_{ABC}$, access control policies do not consider conditions and obligations and just some types of rights and attributes are managed. Given a policy $\rho(\rho_s; \rho_o; \rho_{rt}; r; \emptyset; \emptyset)$: the right $r$ takes value *read*; $\rho_o$ involves attribute $Title$ or $\emptyset$ when any attribute is specified; $\rho_s$ involves attributes $Age$ and/or $Colleague$ or $\emptyset$; and $\rho_{rt}$ involves attributes $Role$ and/or $Trust$ or $\emptyset$. This function is presented in Figure 9(a).

For instance: a user may establish $\{Right = read,\ Title = party,\ Age > 20,\ Role = work\}$ to express that photos titled "party" can be accessed by users older than 20 with whom a worker relationship has been established.

*8.2.3. Tagging of users*

Once an owner selects a photo from those stored in Facebook, he specifies, if desired, a title, and a facial recognition system identifies the people involved in it. Then, each owner, from the set of CooPeD contacts, selects one for each tag. Moreover, tags can be modified if they are not properly located. Specifically, this function is compared with the decomposition of each object $o_i$ in $o_i^j$ objects, such that owners associate each $o_i^j$ with the appropriate user who becomes its co-owner. Note that tagging is a form of delegation where owners execute the delegation operation (DELEGATE($v_k$,$v_j$,$o$,$\lambda$)), being $v_k$ the owner, $v_j$ the chosen user, and $\lambda$ the value AR. This function is presented in Figure 9(b).

*8.2.4. Photos request*

WBSN users can request access to photos which, in CooPeD, are limited to photos of direct contacts. Per each photo request policies of tagged users are evaluated and the requested photo is processed accordingly. If some parts have to be hidden, the *Objects processing module* creates opaque, noise or pixelated rectangles from the top to the bottom of the photo passing through the rectangles that in the *uploading phase* identifies users faces. This function is presented in Figure 9(c).

## 9. Survey study

A survey is performed to analyse the relevance of co-ownership management and the usefulness of the proposal. The following sections present the applied methodology (Section 9.1) and results of the analysis (Section 9.2).

An important aspect is that the survey starts with a description of CooPeD, including snapshots of the prototype (see Appendix A). Consequently, even not using the prototype itself, users can have a clear idea of it and then, results can be compared with those achieved in a real user-centered evaluation.

### *9.1. Methodology*

First of all, the goal of the survey is determining the usefulness of CooPeD, highlighting the circumstances under which its use would be desirable.

According to this goal, a total of 9 questions were elicited (Q1-Q9), being all of them pointed out in Figure 10. Note that tagging is the only functionality related to co-ownership management in current WBSNs. This is the reason why all defined questions are mainly focused on the use of tags as a means of identifying co-owners.

Afterwards, the survey, which consists of a brief introduction to CooPeD and the proposed questions, was created in Google Drive[9].

In last place, a crawler was developed to send the survey URL worldwide. This program was run for 10 days. After three weeks since the crawler stopped, 206 people have completed the survey. This amount of people was considered significant and results were gathered.

### *9.2. Results of the study*

Figure 10 depicts results of the analysis in respect to each question individually. This study is divided in three different blocks: (I) the identification of WBSN users; (II) the study of the potential users profile; and (III) the analysis of the users expected satisfaction. Firstly, regarding Q1, 97.1% of respondents are WBSN users.

Secondly, the profile of potential users are analysed in Q2-Q5. From Q2 it is highlighted that the majority of respondents, 78.6%, grant access to their data to friends. Besides, in relation to Q3, 49% of the respondents affirm that they have less than 100 photos in their profile and 45.1 % point out that they are tagged in a reduced set of photos. Furthermore, concerning Q4, 45.1% of respondents have few photos in which they are tagged, 29.6% are tagged in most of the photos, 23.3% in about a half and 1.9% in all of them. However, as the

[9]http://www.google.com/drive/, last access December 2013

plot associated with Q5 depicts, 81.1% of respondents are worried about photos in which they appear but do not control.

Thirdly, the users expected satisfaction is studied regarding Q6-Q9. Results from Q6 point out that a 52.4% of respondents have photos that would not like to be entirely visualized by a person or a group of people. Moreover, concerning Q7, 81.6% of respondents agree with allowing that different users visualize the same photo differently. Specifically, based on the analysis of Q8, 63.6% of respondents affirm that they would use the system, 10.2% that would not, 23.8% that only for relevant photos and 2.4% in other cases. Furthermore, the plot related to Q9 shows that 79.1% of respondents affirm that their interest in using CooPeD would increase if sophisticated hidden techniques (e.g. replacement) were applied.

A deep analysis of the profile of respondents, who are WBSN users, is depicted in Table 11. In general, respondents choose "Friends" as privacy preferences (Q2) no matter the amount of photos they have in their profiles (Q3). In particular, 75.47% of respondents who have less than 100 photos in their profiles, 86.76% of respondents who have between 100 and 500 photos in their profiles and 75% of respondents who have more than 500 photos in their profiles, have established "Friends" as privacy preferences. Moreover, regardless of the chosen privacy preferences (Q2) or the amount photos users have in their profiles (Q3), respondents are worried about photos in which they are tagged (Q5). Indeed, all the respondents who have established privacy preferences as "Public", having less than 100 photos and having between 100 and 500 photos in their profiles, are worried about being tagged. Therefore, it can be concluded that co-ownership access control management is worth studying regardless of the kinds of users.

On the other hand, it should be recalled that CooPeD focuses on processing decomposable objects according to owners and co-owners privacy preferences granting access to the appropriate object parts. Thus, different users can visualize the same object in a different way. In this regard, Table 12 analyses the amount of respondents who, being WBSN users, are potential users of CooPeD.

Table 11: Analysis of users' profiles, relative percentages

| Q3 | | Q2 | | Q5 | |
|---|---|---|---|---|---|
| Estimate the number of photos you have on your profile | | What privacy preferences do you generally specify in social networks? | | Are you worried about what might happen with photos in which you are tagged even not being their owner? | |
| Answer | # responses (%) | Answer | # responses (%) | Answer | # responses (%) |
| Less than 100 | 106 (53.27%) | Public | 3 (2.83%) | Yes | 3 (100%) |
| | | | | No | 0 (0%) |
| | | Only me | 15 (14.15%) | Yes | 13 (86.67%) |
| | | | | No | 2 (13.33%) |
| | | Friends | 80 (75.47%) | Yes | 65 (81.25%) |
| | | | | No | 15 (18.75%) |
| | | Friends of friends | 2 (1.89%) | Yes | 1 (50%) |
| | | | | No | 1 (50%) |
| | | Groups | 6 (5.66%) | Yes | 6 (100%) |
| | | | | No | 0 (0%) |
| Between 100 and 500 | 69 (34.67%) | Public | 2 (2.90%) | Yes | 0 (0%) |
| | | | | No | 2 (100.0%) |
| | | Only me | 0 (0%) | Yes | 0 (0%) |
| | | | | No | 0 (0%) |
| | | Friends | 60 (86.76%) | Yes | 54 (90.00%) |
| | | | | No | 6 (10.00%) |
| | | Friends of friends | 5 (7.25%) | Yes | 1 (20.00%) |
| | | | | No | 4 (80.00%) |
| | | Groups | 2 (2.90%) | Yes | 1 (50.00%) |
| | | | | No | 1 (50.00%) |
| More than 500 | 24 (12.06%) | Public | 1 (4.17%) | Yes | 1 (100.0%) |
| | | | | No | 0 (0%) |
| | | Only me | 0 (0%) | Yes | 0 (0%) |
| | | | | No | 0 (0%) |
| | | Friends | 18 (75%) | Yes | 12 (66.67%) |
| | | | | No | 6 (33.33%) |
| | | Friends of friends | 1 (4.17%) | Yes | 0 (0%) |
| | | | | No | 1 (100.0%) |
| | | Groups | 4 (16.67%) | Yes | 3 (75.00%) |
| | | | | No | 1 (25.00%) |

85.40% of respondents who have photos which they want to show partially (Q6:yes), allow different visualizations of a photo by different users (Q7:yes). Likewise, from the set of respondents who do not have photos to hide partially (Q6:no), 78.35% of them also allow different users to visualize the same photo in a different way (Q7:yes).

Additionally, the analysis of users who would use the system (Q8) is essential to identify potential users. From the set of respondents who accept a different visualization of a photo (Q7:yes) and have photos to disclose partially (Q6:yes), 86.68% of them would use the system in any case and 18.18% for relevant photos. Furthermore, from the set of respondents that allow different visualizations of a photo (Q7:yes) and assuming that respondents who do not currently have photos to disclose partially (Q6:no) they may have in the future, 57.89% of them would use CooPeD in any case and 32.21% for relevant photos. Note that the remaining set of cases (namely, Q6:yes/no followed by Q7:no) are not relevant for the analysis because respondents involved in such sets do not allow a photo to be differently visualized by different users and it is essential to use CooPeD.

In the light of the foregoing results, potential users of CooPeD corresponds to respondents who having or not photos to partially disclose (Q6:yes or no), allow a photo to be differently visualized by different users (Q7:yes) and would use the system in any case (Q8:yes) or for relevant photos (Q8:only relevant). Thus, identified in Table 12 with symbol $*$, potential users correspond to 78.5% ($\frac{(71+16+44+26)\cdot 100}{200}$) of the set of respondents who are WBSN users and 76.2% ($\frac{(71+16+44+26)\cdot 100}{206}$) in respect to the total amount of respondents.

As a final remark, from Table 12 it is noticed that, in general, the interest of using the system would increase (Q9:yes) in case of applying hiding sophisticated techniques. More specifically, respondents who would use the system in any case or for relevant photos are the most interested in applying hiding sophisticated techniques. Besides, the interest of respondents who would not use the system would also increase applying such hiding techniques.

Table 12: Analysis of potential users, relative percentages

| Q6 | | Q7 | | Q8 | | Q9 | |
|---|---|---|---|---|---|---|---|
| Do you have photos you would not want someone (or group of people) to see them completely? | | As a photo owner, would you agree with the fact that different users visualize the same photo in a different way? | | Would you use the proposed system? | | Would your interest in using the system increase if the hiding technique was sophisticated? | |
| Answer | # responses (%) | Answer | # responses (%) | Answer | # responses (%) | Answer | # responses (%) |
| Yes | 103 (51.50%) | Yes | 88 (85.40%) | Yes | 71 (80.68%)* | Yes | 63 (88.73%) |
| | | | | | | No | 8 (11.27%) |
| | | | | No | 1 (1.14%) | Yes | 1 (100.0%) |
| | | | | | | No | 0 (0%) |
| | | | | Only relevant | 16 (18.18%)* | Yes | 14 (87.5%) |
| | | | | | | No | 2 (12.5%) |
| | | No | 15 (14.56%) | Yes | 9 (60.00%) | Yes | 7 (77.78%) |
| | | | | | | No | 2 (22.22%) |
| | | | | No | 3 (20.00%) | Yes | 2 (66.67%) |
| | | | | | | No | 1 (33.33%) |
| | | | | Only relevant | 3 (20.00%) | Yes | 2 (66.67%) |
| | | | | | | No | 1 (33.33%) |
| No | 97 (48.50%) | Yes | 76 (78.35%) | Yes | 44 (57.89%)* | Yes | 37 (84.09%) |
| | | | | | | No | 7 (15.91%) |
| | | | | No | 6 (7.89%) | Yes | 3 (50.0%) |
| | | | | | | No | 3 (50.0%) |
| | | | | Only relevant | 26 (34.21%)* | Yes | 21 (80.77%) |
| | | | | | | No | 5 (19.23%) |
| | | No | 21 (21.65%) | Yes | 7 (33.33%) | Yes | 4 (57.14%) |
| | | | | | | No | 3 (42.86%) |
| | | | | No | 10 (47.62%) | Yes | 2 (20.0%) |
| | | | | | | No | 8 (80.0%) |
| | | | | Only relevant | 4 (19.05%) | Yes | 1 (25.0%) |
| | | | | | | No | 3 (75.0%) |

*: potential users.

## 10. Related work

This Section analyses co-ownership management proposals (Section 10.1) and techniques and procedures to decompose image-based data applied in CooPeD (Section 10.2).

### *10.1. Co-ownership management*

This Section presents the analysis of 33 proposals, depicted in Table 13, related to co-ownership management in collaborative environments such as WBSNs. Firstly, the administration type is analysed: centralized ($C$), where a single entity decides who can have access to an object; or decentralized ($D$), where multiple users decide who can have access to objects. Secondly, the use of negotiation mechanisms is noticed. Finally, elements involved in access control management are identified.

Concerning centralized administration just 3 proposals fall in this category. In [26] and [27] a central authority is in charge of managing users and groups they belong to. Similarly, [28] proposes team management.

On the other hand, decentralized administration is enforced in 30 approaches in which assorted techniques are proposed. For instance, in [29, 30, 1] owners (called administrators) initiate the administration process by notifying updates to co-owners to, afterwards, become such co-owners involved in the administrative management process. By contrast, in [31, 30, 32, 33] co-owners have to discover co-owned data and request its management.

Collaborative environments such as WBSNs may produce conflicts of interests caused by disjointed preferences. This matter is managed by the development of negotiation mechanisms. The most common technique is based on voting schemes [34, 35, 36, 9, 37, 1]. Given a set of preferences, the number of votes that each of them receives is used to calculate which preferences apply. Similarly, H. Hu *et al.*'s proposal manages owners and co-owners preferences as sets [38]. In case a conflict of interest appears due to the existence of a subset, a superset, a partial set or a disjoint set of user preferences, measures of the

privacy risk and the sharing loss help to determine the preferences to apply. Nonetheless, all mentioned proposals have a common weakness. Owners and co-owners preferences may be contradictory and then, some users privacy may be dismissed.

Trying to reach a full consensus among owners and co-owners preferences, Q. Xiao *et al.* propose CAPE, a mechanism based on managing personal opinions and *peer effects* [39]. Users adjust their access control preferences regarding decisions of other users until a consensus is reached. However, when a user does not change his preferences even taking into account other users' decisions a full consensus is not achieved. Indeed, K. Thomas *et al.*'s approach is the only one that completely preserves users' privacy [2]. This solution calculates the intersection of all users preferences granting or denying access accordingly.

Last but not least, access control management elements are also studied. A significant percentage of approaches focus on roles management [45, 41, 47, 34, 48, 4, 28, 26, 50, 31, 33]. Users are assigned to roles with a set of permissions and they manage access according to roles they belong to. Similarly, some proposals focus on group management [29, 46, 40, 35, 38, 30, 1, 27, 39]. On the contrary, the management of trust and depth between users is an appealing issue in WBSNs [44, 51, 9, 37, 42, 43]. Users trust contacts with whom they are connected at different depths, being user relationships essential WBSN elements [8] manages. Finally, a total of 3 contributions leave opened the set of applied elements [52, 32, 19].

Summarizing, it is concluded that distributed administrative models suppose a challenge necessity in collaborative environments. Moreover, conflicts may occur and some users' preferences may be violated. Besides, access control management elements are quite limited, they specially focus on roles and other elements like object or subject attributes are neglected.

### *10.2. Techniques to decompose objects*

Elements managed in CooPeD are particularly related to photos and videos without audio. In this regard, multiple approaches are based on identifying

Table 13: Administrative features analysis

| **Proposals** | **Administration** | **Neg. Mechanisms** | **Management elem.** |
|---|---|---|---|
| [1] A.C. Squicciarini *et al.* | *D* | √ | Groups membership |
| [2] K. Thomas *et al.* | *D* | √ | General conditions |
| [4] F. Zhu *et al.* | *D* | | Roles |
| [8] Y. Cheng *et al.* | *D* | √ | User relationships |
| [9] B. Carminati *et al.* | *D* | √ | Users trust and depth |
| [26] W.K. Edwards | *C* | | Roles |
| [28] R.K. Thomas | *C* | | Roles |
| [27] H. Zhang *et al.* | *C* | | Groups membership |
| [29] A. Imine *et al.* | *D* | | Groups membership |
| [30] A. Besmer *et al.* | *D* | √* | Groups membership |
| [31] J. Jin *et al.* | *D* | | Roles |
| [32] R. Wishart *et al.* | *D* | √* | General conditions |
| [33] M.R. Thompson *et al.* | *D* | | Roles |
| [40] Y. Ren *et al.* | *D* | | Groups membership |
| [41] M. Prilla *et al.* | *D* | √* | Roles |
| [34] H.F. Wedde *et al.* | *D* | √ | Roles |
| [35] H. Hu *et al.* | *D* | √ | Groups membership |
| [36] V. Gligor *et al.* | *D* | √ | Users and object attributes |
| [37] Y. Sun *et al.* | *D* | √ | Users trust |
| [38] H. Hu *et al.* | *D* | √ | Groups membership |
| [42] A.C. Squicciarini *et al.* | *D* | √ | Users depth |
| [43] A.C. Squicciarini *et al.* | *D* | √ | Users depth |
| [39] Q. Xiao *et al.* | *D* | √ | Depth, groups membership |
| [44] B. Carminati *et al.* | *D* | | Users trust and depth |
| [45] R. S. Shandu *et al.* | *D* | | Roles |
| [46] A. Merlo *et al.* | *D* | | Groups membership |
| [47] K. Sikkel *et al.* | *D* | | Roles |
| [48] Z.Y. Zhang *et al.* | *D* | | Roles |
| [49] M. Lorch *et al.* | *D* | | Users attributes |
| [50] E. Cohen *et al.* | *D* | | Roles |
| [51] S. Braghin *et al.* | *D* | | Users trust and depth |
| [19] D. Lin *et al.* | *D* | √ | General conditions |
| [52] R. S. Shandu *et al.* | *D* | | General conditions |

*: exclusively mentioned (not managed)

different elements in photos and videos.

### *10.2.1. Photos*

An interesting set of techniques focus on people recognition. The most significant area of research is face recognition, being, *eigenfaces* the most remarkable proposal [53]. It is based on identifying the principal components of the face, called eigenvectors, which characterize the variation between faces. Then, assuming that each training face is represented as a collection of eigenvectors, new faces are compared with the stored ones. Furthermore, [54] is another well known approach. It is based on representing faces as rectangular graphs where each node is tagged with a set of coefficients called *jet* . Then, rectangular graphs are stored in respect to a set of training faces and they are compared with a given image. For the same purpose but with an additional advantage, Xu-Yang and Fang-lv present a technique to, apart from recognizing faces, identify the number of people in a particular image [55]. Similarly, O.K. Manyam *et al.* present a face recognition system exploding the dependence between face regions in images where there are multiple people [56]. On the other hand, C.N. Vasconcelos *et al.* propose an approach to identify people in non-controlled situations [57]. It is essentially focused on applying the Kohonen network to learn people's appearance.

Moreover, several contributions presents the analysis of images to detect vehicles, mainly applied to traffic applications [58, 59]. Likewise, other approaches propose techniques to identify animals like birds [60]. Indeed, there are general techniques to identify assorted elements [61] and depending on demands specific algorithms have to be developed accordingly.

### *10.2.2. Videos*

A set of approaches focus on identifying people from a sequence of images. One of them is based on motion-based recognition [62] and other focuses on supervised learning [63]. On the other hand, S.J. McKenna *et al.* present a color-based system for tracking people given a set of sequential frames [64].

Furthermore, different approaches identify vehicles from traffic videos [59] or animals from videos like documentaries [65, 66]. However, as in photos, elements recognition requires the development of specific algorithms.

## 11. Conclusions and future research issues

Web Based Social Networks (WBSN) are one of the most widespread online data sharing environments. For WBSNs to be trusted the preservation of users' privacy is critical. Fine-grained privacy systems must assist users in protecting their privacy no matter which user or physical system uploads the data. In this regard, this proposal presents CooPeD (*Co-owned Personal Data management*), a novel co-ownership management system, composed of a model and a mechanism. CooPeD's mechanism is based on managing decomposable objects according to owners and co-owners privacy preferences. CooPeD works over $SoNeUCON_{ABC}$ model, an expressive usage control model that has been extended to allow co-ownership management by defining access control and administrative management. This proposal has been evaluated in three different ways. Firstly, the feasibility of CooPeD's system, model and mechanism, has been assessed. In a WBSN with direct relationships between 32 and 34 co-owners are supported per object request without exceeding the tolerable waiting time of WBSN users for information retrieval applying any type of architecture. By contrast, in a WBSN with indirect relationships of length 2, between 3 and 4 co-owners are supported per object request applying proposed architectures. Nonetheless, fully decentralized architectures which apply parallelism provide challenging results as, in most cases, any amount of co-owners are supported. Concerning CooPeD's mechanism in Facebook, either for direct or indirect relationships, 11 co-owners are supported assuming the use of a centralized architecture and more than 14 assuming the use of a partially decentralized architecture in which the temporal workload of the evaluation of policies does not exceed 123 ms. Secondly, the development of a prototype has proven the possible implementation of CooPeD. Thirdly and lastly, a survey

study has tested the usefulness and acceptance of the proposal. Results of the survey show that 72.6% of respondents can be potential users of CooPeD.

Nonetheless, remarkable issues are left for future work. Particularly, the extension of $SoNeUCON_{ABC}$ could allow the modification of attributes of uploaded objects either by owners or by co-owners. Furthermore, the visualization of the same object in a different way for different people encourage the study of users satisfaction and users curiosity and suspiciousness. Unless using sophisticated techniques to hide object parts, questions such as who/which is under a hidden part? How can I get to know him/her/it? may arise. Likewise, a controversial issue is working on techniques to deal with parts that belong to multiple users. Moreover, concerning recognition techniques, they have to be powerful enough to satisfactorily identify assorted objects (documents, music, etc.), thereby achieving successful objects decompositions. Also, the search of concrete scenarios where CooPeD may contribute, e.g. to protect children privacy, is a relevant line of research. In this regard, the prototype should be improved to accurately detect users and other elements, as well as hide them with higher precision. Furthermore, the proposed extended model should be improved to make it suitable for WBSNs with indirect relationships are pointed out as future research lines. Finally, future work goes also towards the inclusion of techniques to prevent WBSNs from controlling stored data and to avoid the disclosure of data while evaluating policies.

**Acknowledgements**

Authors would like to thank the anonymous reviewers for their insightful comments which helped us to improve the paper. Moreover, we would like to thank Carlos Castaño for his initial performance test.

## Appendix A. Survey structure

This Section depicts the survey which was sent to users worldwide, see Figure A.11 and A.12. Although the one presented herein is in English, it was translated

to Spanish as well.

[1] A. Squicciarini, H. Xu, and X. Zhang, "CoPE: Enabling collaborative privacy management in online social networks," *Journal of the American Society for Information Science and Technology*, vol. 62, no. 3, pp. 521–534, 2011.

[2] K. Thomas, C. Grier, and D. M. Nicol, "unfriendly: multi-party privacy risks in social networks," in *Proceedings of the 10th international conference on Privacy enhancing technologies*, ser. PETS'10. Springer-Verlag, 2010, pp. 236–252.

[3] L. González−Manzano, A. I. González−Tablas, J. M. de Fuentes, and A. Ribagorda, "SoNeUCON$_{ABC}$, an expressive usage control model for Web-Based Social Networks," *Computers & Security, In Press*, 2014.

[4] F. Zhu and Q. Lv, "Aceac: A novel access control model for cooperative editing with workflow," in *Electronic Commerce and Security, 2008 International Symposium on*, 2008, pp. 1010–1014.

[5] B. Carminati, E. Ferrari, and A. Perego, "Rule−Based Access Control for Social Networks," in *Proc. OTM 2006 Workshops (On the Move to Meaningful Internet Systems)*, ser. LNCS, vol. 4278. Springer, 2006, pp. 1734–1744.

[6] P. W. Fong and I. Siahaan, "Relationship-based access control policies and their policy languages," in *Proceedings of the 16th ACM symposium on Access control models and technologies*, ser. SACMAT '11. ACM, 2011, pp. 51–60.

[7] B. Carminati and E. Ferrari, "Access control and privacy in web-based social networks," in *International Journal of Web Information Systems*, vol. 4, no. 4, 2008, pp. 395–415.

[8] Y. Cheng, J. Park, and R. Sandhu, "A User-to-User Relationship-Based Access Control Model for Online Social Networks," in *Data and Applications Security and Privacy XXVI*, ser. Lecture Notes in Computer Science, 2012, vol. 7371, pp. 8–24.

[9] B. Carminati and E. Ferrari, "Collaborative access control in on-line social networks," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2011 7th International Conference on*, 2011, pp. 231 –240.

[10] H. Lipford, G. Hull, C. Latulipe, A. Besmer, and J. Watson, "Visible flows: Contextual integrity and the design of privacy mechanisms on social network sites," in *CSE (4)*, 2009, pp. 985–989.

[11] R. Burke, "Hybrid recommender systems: Survey and experiments," *User modeling and user-adapted interaction*, vol. 12, no. 4, pp. 331–370, 2002.

[12] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*. Springer, 2011, pp. 73–105.

[13] B. Gold, N. Morgan, and D. Ellis, *Speech and audio signal processing: processing and perception of speech and music*. John Wiley & Sons, 2011.

[14] S. Jahid, S. Nilizadeh, P. Mittal, N. Borisov, and A. Kapadia, "Decent: A decentralized architecture for enforcing privacy in online social networks," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*. IEEE, 2012, pp. 326–332.

[15] W. Luo, Q. Xie, and U. Hengartner, "FaceCloak: An Architecture for User Privacy on Social Networking Sites," *2009 International Conference on Computational Science and Engineering*, pp. 26–33, 2009.

[16] G. Brassard, D. Chaum, and C. Crépeau, "Minimum disclosure proofs of knowledge," *Journal of Computer and System Sciences*, vol. 37, no. 2, pp. 156–189, 1988.

[17] R. Sandhu, V. Bhamidipati, and Q. Munawer, "The arbac97 model for role-based administration of roles," *ACM Transactions on Information and System Security (TISSEC)*, vol. 2, no. 1, pp. 105–135, 1999.

[18] ITU-T-team, "Itu-t recommendation x.812. data networks and open system communications security." http://owasptop10.googlecode.com/files/OWASP\%20Top\%2010\%20-\%202013.pdf,lastaccessApr.2014, 1995.

[19] D. Lin, P. Rao, E. Bertino, N. Li, and J. Lobo, "Policy decomposition for collaborative access control," in *Proceedings of the 13th ACM symposium on Access control models and technologies*, ser. SACMAT'08. ACM, 2008, pp. 103–112.

[20] L. González-Manzano, A. González-Tablas, J. de Fuentes, and A. Ribagorda, "U+f social network protocol: Achieving interoperability and reusability between web based social networks," in *Proceedings of the 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, ser. TRUSTCOM '12, 2012, pp. 1387–1392.

[21] M. Conti, A. Hasani, and B. Crispo, "Virtual private social networks," in *Proceedings of the first ACM conference on Data and application security and privacy*, ser. CODASPY '11. ACM, 2011, pp. 39–50.

[22] M. Machulak, E. Maler, D. Catalano, and A. van Moorsel, "User-managed access to web resources," in *Proceedings of the 6th ACM workshop on Digital identity management*, ser. DIM '10, 2010, pp. 35–44.

[23] K. Graffi, C. Groß, D. Stingl, D. Hartung, A. Kovacevic, and R. Steinmetz, "Lifesocial.kom: A secure and p2p-based solution for online social

networks," in *Proceedings of the IEEE Consumer Communications and Networking Conference*. IEEE Computer Society Press, Jan 2011.

[24] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, "The anatomy of the facebook social graph," *arXiv preprint arXiv:1111.4503*, 2011.

[25] F. F.-H. Nah, "A study on tolerable waiting time: how long are web users willing to wait?" *Behaviour & Information Technology*, vol. 23, no. 3, pp. 153–163, 2004.

[26] W. K. Edwards, "Policies and roles in collaborative applications," in *Proceedings of the 1996 ACM conference on Computer supported cooperative work*, ser. CSCW '96. ACM, 1996, pp. 11–20.

[27] H. Zhang, W. Wu, and Z. Li, "Open social based group access control framework for e-science data infrastructure," in *E-Science (e-Science), 2012 IEEE 8th International Conference on*. IEEE, 2012, pp. 1–8.

[28] R. K. Thomas, "Team-based access control (TMAC): a primitive for applying role-based access controls in collaborative environments," in *Proceedings of the second ACM workshop on Role-based access control*, ser. RBAC '97. ACM, 1997, pp. 13–19.

[29] A. Imine, A. Cherif, and M. Rusinowitch, "A flexible access control model for distributed collaborative editors," in *Proceedings of the 6th VLDB Workshop on Secure Data Management*, ser. SDM '09, 2009, pp. 89–106.

[30] A. Besmer and H. Richter Lipford, "Moving beyond untagging: photo privacy in a tagged world," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10. ACM, 2010, pp. 1563–1572.

[31] J. Jin and G.-J. Ahn, "Role-based access management for ad-hoc collaborative sharing," in *Proceedings of the eleventh ACM symposium on Access control models and technologies*, ser. SACMAT'06. ACM, 2006, pp. 200–209.

[32] R. Wishart, D. Corapi, S. Marinovic, and M. Sloman, "Collaborative privacy policy authoring in a social networking context," in *Policies for Distributed Systems and Networks (POLICY), 2010 IEEE International Symposium on*, 2010, pp. 1 –8.

[33] M. R. Thompson, A. Essiari, and S. Mudumbai, "Certificate-based authorization policy in a pki environment," *ACM Transactions on Information and System Security (TISSEC)*, vol. 6, no. 4, pp. 566–588, 2003.

[34] H. Wedde and M. Lischka, "Cooperative role-based administration," in *Proceedings of the eighth ACM symposium on Access control models and technologies*, ser. SACMAT'03. ACM, 2003, pp. 21–32.

[35] H. Hu, G.-J. Ahn, and J. Jorgensen, "Multiparty access control for online social networks: Model and mechanisms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 99, 2012.

[36] V. Gligor, H. Khurana, R. Koleva, V. Bharadwaj, and J. Baras, "On the negotiation of access control policies," in *Security Protocols*. Springer, 2002, pp. 188–201.

[37] Y. Sun, C. Zhang, J. Pang, B. Alcade, and S. Mauw, "A trust-augmented voting scheme for collaborative privacy management," in *Proceedings of the 6th international conference on Security and trust management*, ser. STM'10. Springer-Verlag, 2011, pp. 132–146.

[38] H. Hu, G.-J. Ahn, and J. Jorgensen, "Detecting and resolving privacy conflicts for collaborative data sharing in online social networks," in *Proceedings of the 27th Annual Computer Security Applications Conference*, ser. ACSAC '11. ACM, 2011, pp. 103–112.

[39] Q. Xiao and K.-L. Tan, "Peer-aware collaborative access control in social networks," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2012 8th International Conference on*. IEEE, 2012, pp. 30–39.

[40] Y. Ren, "Access control in a cooperative editing system," in *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, 2011, pp. 77 –80.

[41] M. Prilla and C. Ritterskamp, "Collaboration support by co-ownership of documents," in *Proceedings of the 2006 conference on Cooperative Systems Design: Seamless Integration of Artifacts and Conversations – Enhanced Concepts of Infrastructure for Communication*, 2006, pp. 255–269.

[42] A. Squicciarini, M. Shehab, and F. Paci, "Collective privacy management in social networks," in *Proceedings of the 18th international conference on World wide web*, ser. WWW '09, 2009, pp. 521–530.

[43] A. C. Squicciarini, M. Shehab, and J. Wede, "Privacy policies for shared content in social network sites," *The VLDB JournalâThe International Journal on Very Large Data Bases*, vol. 19, no. 6, pp. 777–796, 2010.

[44] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham, "Semantic web-based social network access control," *computers & security*, vol. 30, no. 2, pp. 108–115, 2011.

[45] R. S. Sandhu, K. Z. Bijon, X. Jin, and R. Krishnan, "RT-based administrative models for community cyber security information sharing," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2012 8th International Conference on*, 2011, pp. 473–478.

[46] A. Merlo and A. Armando, "Cooperative access control for the grid," in *Information Assurance and Security (IAS), 2010 Sixth International Conference on*, 2010, pp. 228 –233.

[47] K. Sikkel, "A group-based authorization model for cooperative systems," in *Proceedings of the fifth conference on European Conference on Computer-Supported Cooperative Work*, ser. ECSCW'97. Kluwer Academic Publishers, 1997, pp. 345–360.

[48] Z. Zhang, T. Huang, Q. Wu, and J. Pu, "A cscw-enabling integrated access control model and its application," *Key Engineering Materials*, vol. 460, pp. 96–105, 2011.

[49] M. Lorch, D. B. Adams, D. Kafura, M. S. R. Koneni, A. Rathi, and S. Shah, "The PRIMA System for Privilege Management, Authorization and Enforcement in Grid Environments," in *Proceedings of the 4th International Workshop on Grid Computing*, ser. GRID '03. IEEE Computer Society, 2003.

[50] E. Cohen, R. K. Thomas, W. H. Winsborough, and D. Shands, "Models for coalition-based access control (CBAC)," in *Proceedings of the eighth ACM symposium on Access control models and technologies*, ser. SACMAT'02, 2002, pp. 97–106.

[51] S. Braghin, E. Ferrari, and A. Trombetta, "Combining access control and trust negotiations in an On-line Social Network," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2010 6th International Conference on*, 2010, pp. 1 –10.

[52] R. Sandhu, R. Krishnan, J. Niu, and W. Winsborough, "Group-centric models for secure and agile information sharing," *Computer Network Security*, pp. 55–69, 2010.

[53] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[54] L. Wiskott, J.-M. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," in *Image Processing, 1997. Proceedings., International Conference on*, vol. 1, 1997, pp. 129 –132.

[55] X. Yang and F. Lv, "The design of a face recognition system based on skin color and geometrical characteristics," in *Image Analysis and Signal Processing (IASP), 2011 International Conference on*, 2011, pp. 276 –279.

[56] O. Manyam, N. Kumar, P. Belhumeur, and D. Kriegman, "Two faces are better than one: Face recognition in group photographs," in *International Joint Conference on Biometrics (IJCB)*, 2011.

[57] C. Vasconcelos, V. Jardim, A. Sá, and P. Carvalho, "Photo tagging by collection-aware people recognition," Institute of computing (Brasil), Tech. Rep., 2012.

[58] L. Salgado, J. Menendez, E. Rendon, and N. Garcia, "Automatic car plate detection and recognition through intelligent vision engineering," in *Security Technology, 1999. Proceedings. IEEE 33rd Annual 1999 International Carnahan Conference on*. IEEE, 1999, pp. 71–76.

[59] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and vision computing*, vol. 21, no. 4, pp. 359–381, 2003.

[60] U. Nadimpalli, "Image processing techniques to identify predatory birds in aquacultural settings," Ph.D. dissertation, B.E. Andhra University, 2005.

[61] J. Buhmann and P. Malik, J.and Perona, "Image recognition: Visual grouping, recognition, and learning," *Proceedings of the National Academy of Sciences*, vol. 96, no. 25, pp. 14 203–14 204, 1999.

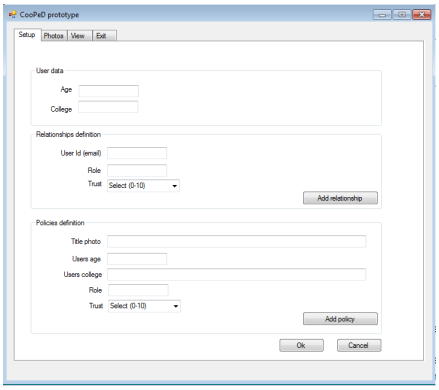
[62] C. BenAbdelkader, R. Cutler, H. Nanda, and L. Davis, "EigenGait: Motion-Based Recognition of People Using Image Self-Similarity," in *Audio- and Video-Based Biometric Person Authentication*, 2001, pp. 284–294.

[63] C. Nakajima, M. Pontil, B. Heisele, and T. Poggio, "People recognition in image sequences by supervised learning," MIT Artificial Intelligence Laboratory, June, Tech. Rep., 2000.

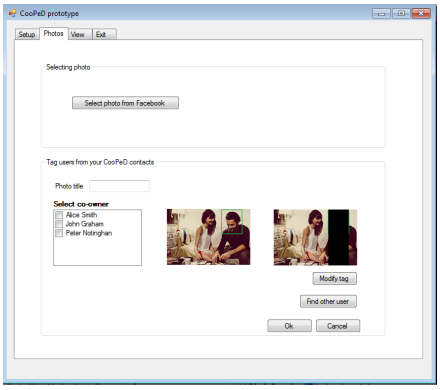
[64] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," *Computer Vision and Image Understanding*, vol. 80, no. 1, pp. 42 – 56, 2000.

[65] D. Ramanan and D. Forsyth, "Using temporal coherence to build models of animals," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003, pp. 338–345 vol.1.

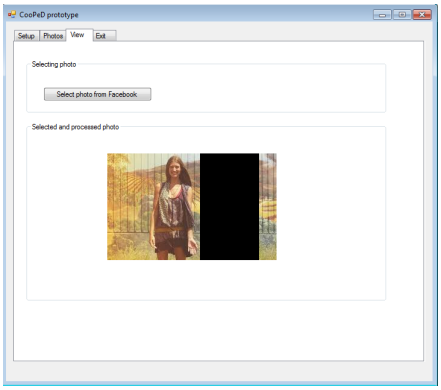
[66] D. Ramanan, D. Forsyth, and K. Barnard, "Building models of animals from video," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 8, pp. 1319–1334, 2006.




(a) Users data, relationships and policies definition




(b) Co-owners specification




(c) Photo request

Figure 9: CooPeD prototype

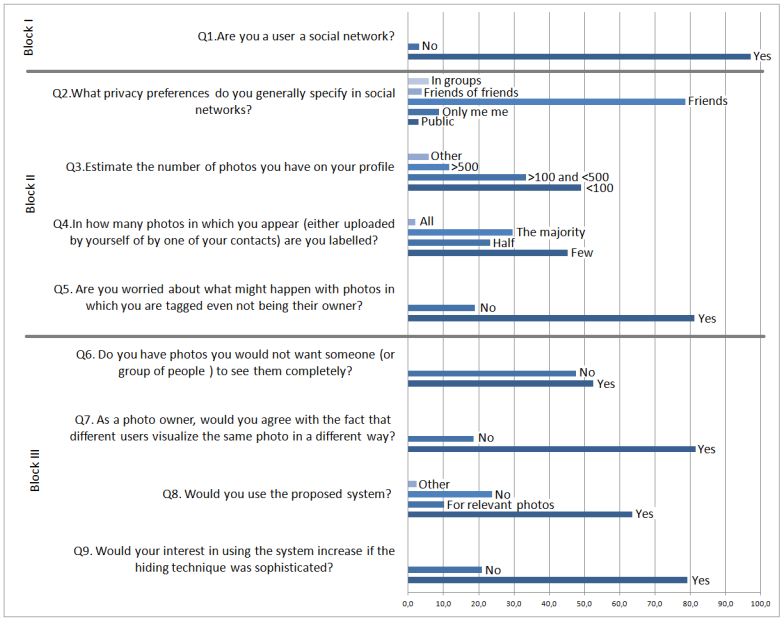



Figure 10: Survey study

## Managing access control in social networks

Thank you for your cooperation in this study conducted by the University Carlos III of Madrid as a part of a doctoral thesis.

It will only take you 5 minutes.

We propose the development of a system to manage access control in social networks. Please read the description of the system and then answer the questions.

* Required

Universidad Carlos III de Madrid
www.uc3m.es

### System description

In social networks there are photos that you are tagged but you do not have control over them. For this reason, we have developed a system to offer to users, who are tagged in photos and do not want to be seen, **the possibility of limiting who can view the part of the photo in which they appear.**

Suppose you are the person in the orange box, who appears in a photo with Alice (a friend) who is in the green box.

Alice You

The **system allows you to specify who can see your part of the photo** and **allows Alice to specify who can see her part of the photo**. A prototype has been implemented in regard to the proposed system, called CooPeD. The prototype consists of a web application that allows co-ownership management of photos of people. It is expected the prototype to be linked to a popular WBSN, Facebook. In particular, this prototype applies Facebook authentication and Facebook photos. In the prototype four functions are available. First, users have to detail some info about themselves and create relationships with other CooPeD users (as in a social network like Facebook), as well as specify privacy preferences to determine who is able to access to their photos (access control policies) (see Figure A). Second, users choose photos of Facebook and tag CooPeD contacts in it (see Figure B).

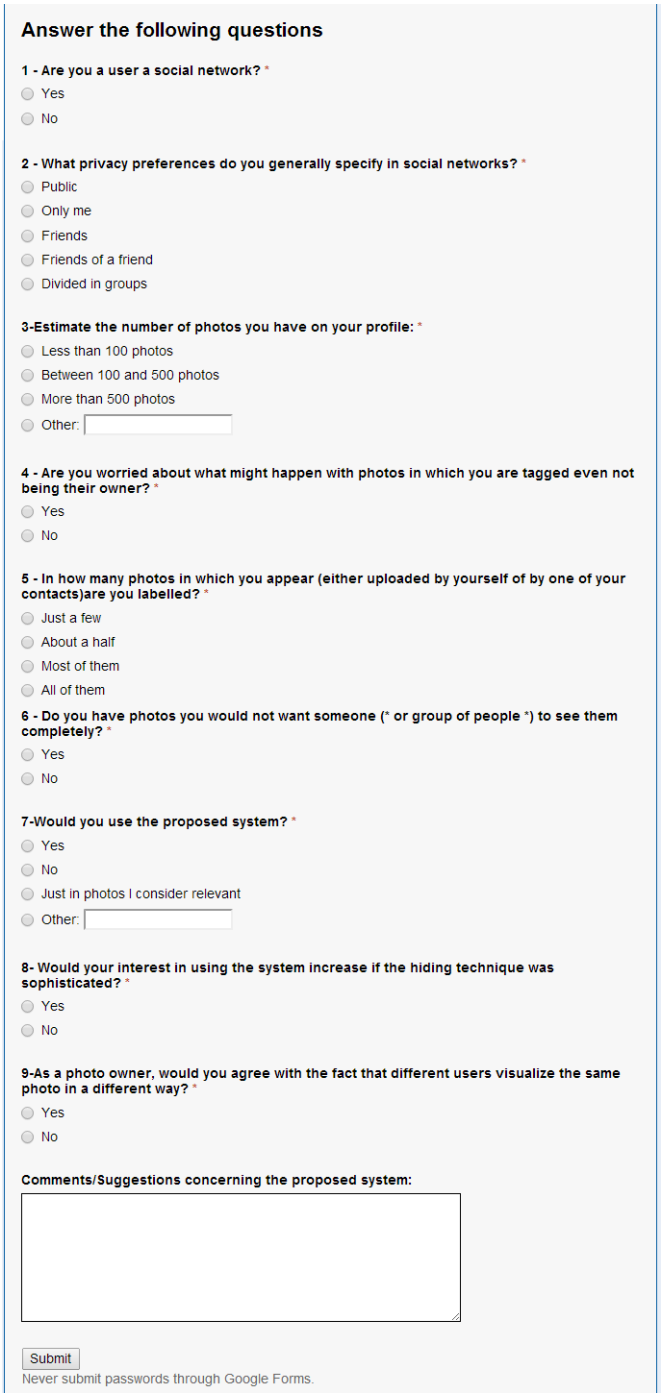
A B

Finally, users can request access to photos. Concerning the first example, if a user enters and both Alice and you let him to see the photo, he will see 1), if you do not want to be seen, he will see 2), if Alice does not want to be seen, he will see 3), and if Alicia and you do not want to be seen, the photo will not be shown.

1 2 3

Note that currently users are hidden applying black rectangles but they can be hiding using more sophisticated techniques:

Figure A.11: Survey - First part



## Answer the following questions

**1 - Are you a user a social network?** *
- Yes
- No

**2 - What privacy preferences do you generally specify in social networks?** *
- Public
- Only me
- Friends
- Friends of a friend
- Divided in groups

**3-Estimate the number of photos you have on your profile:** *
- Less than 100 photos
- Between 100 and 500 photos
- More than 500 photos
- Other:

**4 - Are you worried about what might happen with photos in which you are tagged even not being their owner?** *
- Yes
- No

**5 - In how many photos in which you appear (either uploaded by yourself of by one of your contacts)are you labelled?** *
- Just a few
- About a half
- Most of them
- All of them

**6 - Do you have photos you would not want someone (* or group of people *) to see them completely?** *
- Yes
- No

**7-Would you use the proposed system?** *
- Yes
- No
- Just in photos I consider relevant
- Other:

**8- Would your interest in using the system increase if the hiding technique was sophisticated?** *
- Yes
- No

**9-As a photo owner, would you agree with the fact that different users visualize the same photo in a different way?** *
- Yes
- No

**Comments/Suggestions concerning the proposed system:**

Submit

Never submit passwords through Google Forms.

Figure A.12: Survey - Second part