C3PO: A Network and Application Framework for Spontaneous and Ephemeral Social Networks

Antoine Boutet, Stephane Frenot, Frederique Laforest, Pascale Launay, Nicolas Le Sommer, Yves Mahéo, Damien Reimert

C3PO: A Network and Application Framework for Spontaneous and Ephemeral Social Networks

Antoine Boutet[2] and Stephane Frenot[3] and Frederique Laforest[2] and Pascale Launay[1] and Nicolas Le Sommer[1] and Yves Maheo[1] and Damien Reimert[3]

[1] IRISA, Universite de Bretagne-Sud, first.last@univ-ubs.fr
[2] Laboratoire Hubert Curien, Universite de Saint Etienne, first.last@univ-st-etienne.fr
[3] DICE-INRIA, CITI, Universite de Lyon, first.last@inria.fr

Abstract. The C3PO project promotes the development of new kind of social networks called Spontaneous and Ephemeral Social Networks (SESNs) dedicated to happenings such as cultural or sport events. SESNs rely on both opportunistic networks formed dynamically by the mobile devices of event attendees, and on an event-based communication model. Therefore, user can exchange digital contents with the other members of their SESNs, even without Internet access. This paper presents the framework developed in the C3PO project to provide network and application supports in such challenged networks. This framework exploits the different wireless interfaces of the mobile devices to interconnect them and to disseminate content through the resulting opportunistic network. At the application layer, this framework is composed of plugins that process locally the data stream to offer generic features, or to easily build applications dedicated to specific happenings.

Keywords: Mobile Social Networking, Opportunistic Computing.

1 Introduction

A new era of communication is now underway with the massive use of online social networks and media. Besides the two giants Facebook and Twitter, a multitude of more or less specialized software platforms allow people to interact and share information via the Internet. Although users are faced with a diversity of purposes and types of platforms (business-oriented, dedicated to music, books or sports, designed to share photos, videos or news...), the vast majority of these social networks are build on the same architectural model: a centralized service provider acts as a broker to filter and disseminate the data produced by users. This architecture follows the traditional Web 2.0 model but is not without drawbacks from the user's point of view. First, it induces a dependence on a centralized authority that controls the data management and exchange (more than often, data ownership is even transferred to the service provider). Second, this architecture requires that the user has a permanent access to the Internet to exchange with others. Third, (friendship) links between users are mostly permanent and are used as pipes of communication: information transits among users following these links.

This is but not always the most adapted information spreading principle. For example, during large events like sports events or festivals, the geographic vicinity of users is a more appropriate paradigm: spectators, competitors, organizers and other participants would benefit from communicating contents within the geographic area of the event. Moreover, relationships set up during the event can disappear at the end of the event.

The C3PO project (Collaborative Creation of Contents and Publishing using Opportunistic networks) proposes to investigate multimedia content production and exchange in a new type of social networks that we call Spontaneous and Ephemeral Social Networks (SESNs). SESNs rely on a peer-to-peer distributed architecture formed spontaneously by mobile devices carried by people and, optionally, by fixed devices that can be deployed to support such networks. Devices are interconnected using their wireless interfaces (e.g., Wi-Fi or Bluetooth). Opportunistic communication techniques are devised in order to support the connectivity disruptions resulting from the mobility of users and the short communication range of the radio interfaces. Thus, these techniques allow devices to exchange data even if they are out of the radio range of each others and no end-to-end path exist between them. Due to their spontaneous and ephemeral nature, SESNs are suited to produce multimedia reports on conferences and cultural or sport events, such as a marathon as illustrated in Figure 1. For instance, the spectators of a marathon can take photos of participants, can write comments about the marathon and can publish these contents in the SESN dedicated to this event. SESNs are expected to vanish with the end of the event. See [3] for a general description of the project.

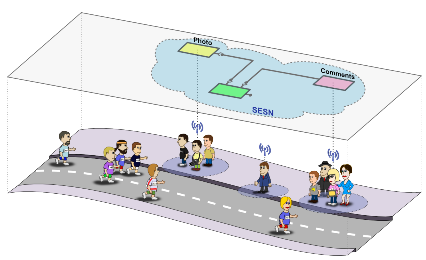


Fig. 1: Cultural and sport events are typical use cases of SESNs

In this paper, we present the framework designed in project C3PO to support opportunistic networking in SESNs. This framework allows the building of a communication middleware that offers a message-oriented API (based on the concepts of topic and named channel) to application programmers so that they can develop C3PO plugins adapted to specific needs (for example a plugin may be dedicated to the emission, reception and presentation of the official results of a race). As a communication framework, the C3PO framework is intended to be specialized to form a communication middleware. This is done by implementing a small number of functions that constitute a set of reactions to standard communication events (reception of a message, appearance of a neighbor...). The programmer either develops original implementations of these functions, or uses the implementations provided in the C3PO toolkit. To date, the toolkit proposes several epidemic-based dissemination strategies.

2 Network Architecture and framework

This section presents the network level concerns of the C3PO project. It considers both the network architecture and the framework that sustains this architecture.

2.1 Network Architecture

A SESN is formed by users carrying off-the-shelf mobile devices (smartphones, tablets) communicating with each others thanks to short range wireless interfaces implementing communication standard such as Bluetooth or IEEE 802.11 (Wi-Fi). In most cases, the users that compose the SESN can move, and are not necessarily all located in the same place at the same time. The topology of the network formed by their devices changes continuously and can be fragmented into communication islands. Thus, as depicted in Figure 2, the network is structured as a collection of what we call *micronets*, grouped together in independent *macronets*.

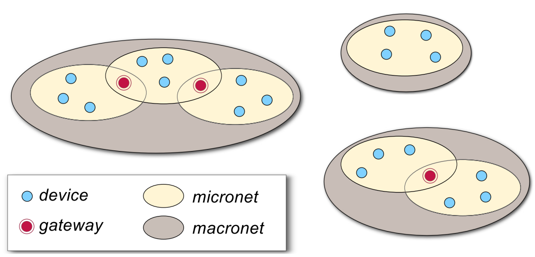


Fig. 2: Network topology

Micronets We define a micronet as a subset of devices connected using a common communication technology, that are able to communicate directly with each others. A micronet is an abstraction of a piconet for Bluetooth, a basic service set (BSS) for Wi-Fi legacy, or a group for Wi-Fi direct. We consider that devices in a micronet communicate *directly* as there exists a network layer that allows any device to address and send messages to any other device in its micronet, even if it is not the case at the MAC layer. Wireless communication technologies available for current off-the-shelf mobile devices allow only a limited number of devices to be interconnected and exchange data directly between them (e.g. at most 8 devices for a Bluetooth piconet). Consequently, micronets have to be interconnected in order to allow SESN users to communicate over the whole network.

Macronets We define a macronet as a group of micronets interconnected through devices that are members of at least two micronets. The resulting communication network is a collection of independent macronets. Macronets can be analogized to Wi-Fi Direct multi-groups as defined in [1] or to Bluetooth scatternets. But the micronets forming a macronet can use different communication technologies.

It cannot be assumed that there is a common addressing scheme in a macronet nor that messages can be transmitted directly between the devices of a macronet, even in the case of a Bluetooth scatternet. The C3PO framework relies on the store-carry-and-forward principle for both intra-macronet and inter-macronet communication. Indeed, temporaneous end-to-end paths in the whole SESN (i.e., between macronets) cannot be established; therefore, a store-carry-and-forward layer must be developed for communication between macronets, and this layer can also be used for exchanging messages inside a macronet. Besides the simplification of the framework, this option is likely to provide a better tolerance against network disruptions.

2.2 C3PO Network Level Framework

The C3PO network level framework is a Java-based framework dedicated to opportunistic networking. It provides application developers with an API offering two distinct application-level communication paradigms, namely a topic-based publish/subscribe paradigm, and a channel-based send/receive paradigm. Its architecture, illustrated in Figure 3, is organized in five main modules.

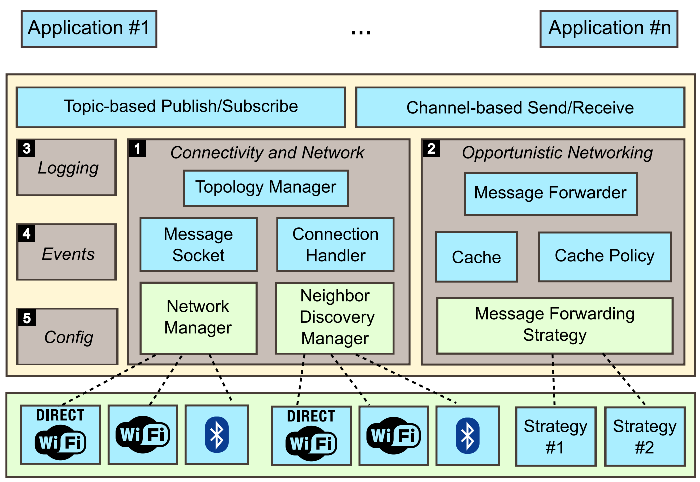


Fig. 3: General architecture of the C3PO network level framework

Modules 1 and 2 are related to the management of micronets and macronets. Module 1 manages wireless interfaces and sets up communications in micronets. It defines abstract neighbor discovery managers and network managers, that are respectively responsible for discovering neighbor devices (i.e., devices in radio range) to build micronets and for managing communications to support data exchanges between devices inside a micronet. Module 2 defines opportunistic networking mechanisms to perform data forwarding inside and between macronets. It includes a cache of messages, a message forwarder and abstract forwarding strategies in order to implement the store-carry-and-forward principle. The three other modules (i.e., modules 3, 4 and 5) composing the framework define functionalities that are respectively dedicated to the configuration of the framework, to the management of the events produced by the framework, and to log the traces that are generated by the framework (e.g., contact and message exchanges traces). Several concrete implementations of the abstract functionalities defined in the C3PO framework are provided in a toolkit. These implementations can be used as they are, or can be extended by developers in order to adapt them to their own needs.

Two distinct application-level communication paradigms are provided by the network level framework: a topic-based publish/subscribe model, and a channel-based send/receive model.

The *topic-based publish/subscribe communication model* makes it possible to develop applications that can publish multimedia contents on specific topics, and subscribe in order to receive the contents related to given topics. The model implemented in the C3PO framework relies on a purely peer-to-peer decentralized approach. The subscription and the publication are local to each device. Thanks to the store-carry-and-forward principle, contents published in a topic by publishers are disseminated opportunistically in the communication network by mobile devices, being either devices hosting subscribers for this topic or ordinary intermediate devices, and are thus delivered to the topic subscribers.

The *point-to-point communication paradigm* using the concept of channel is intended for applications that allow users to communicate with each others by sending messages addressed to specific recipients. In the framework, a channel between two devices is identified by the addresses of the devices and a channel ID. Messages sent through a channel are opportunistically forwarded by intermediate devices towards their destination according to one of the message forwarding strategies implemented in the framework. Thus, two devices can exchange data even if they are not within mutual radio range.

The C3PO framework implements the store-carry-and-forward principle. Different types of messages are currently considered: application-level messages and several types of control messages such as beacon messages that are used to exchange the neighbor lists, gossiping messages, acknowledgements, and drop messages that make it possible to implement network healing protocols in order to reduce the number of messages that are disseminated in the network. The messages that are published or sent by the applications are exchanged by mobile devices according to a given forwarding strategy (i.e., a flooding and a gossip-based message forwarding strategy). These message forwarding strategies are reactive and based on four main events: the emission of a message by a local application, the reception of a message from the network, the discovery and connection of a new neighbor device, and the disappearance and disconnection of a neighbor device.

Message forwarding strategies allow to replicate the messages on all the nodes forming the opportunistic network, provided these nodes have enough space in their local cache to store these messages. Compared to a plain flooding, the gossip-based message forwarding reduces drastically the number of messages, by maintaining on each device a vision of the content of the caches of its neighbors, through the exchange of catalogs of message IDs. More information on the C3PO network level framework can be found in [4].

3 C3PO Application Level Framework

The C3PO application level framework is built on top of the network level framework and uses its API for receiving and sending messages throughout the SESN. This application level framework is designed as an in-browser architecture, so as to comply to any device as soon as it provides a browser compatible with the javascript technologies.

The main objective of the application level framework is to offer a mean to define, adapt and extend the application to the target SESN. As a matter of fact, SESNs should provide generic functionalities that are common to any SESN, but also specific functionalities according to the happening it covers. An example of generic functionality is presenting the flow of contents exchanged by end-users, while a specific functionality can be the identification of numbers in runners pictures during a race. Another challenge in such systems is to help users read first the information that are likely to interest him most. In this section, we present the general architecture of the application level of C3PO, as well as some examples of application functionalities.

3.1 C3PO Application Level Framework Architecture

At the application level, the C3PO framework gets messages as a flow of events, processes them and presents the results to the end-user. This level is also responsible for the creation of new events reflecting some end-user actions. For example, the end-user can create new contents to be exchanged with other end-users; he can also promote a received content to stress his special interest.

Depending on the SESN happening type, the events may carry very different contents and thus event processing requires extensibility. Devices in the same SESN have a set of identical processing units, but some more powerful devices may also have additional specific ones.

The global architecture of the application level part is illustrated on figure 5. Plugins are components responsible for the processing of incoming events and for the interaction with the end-user [2]. Each plugin has a specific event processing unit, dedicated to a kind of event. For each incoming event, the event dispatcher selects in a registry of plugins the corresponding plugins and calls their processing unit with a reference to the event. Plugins can store their processing results in their own context.

When a device connects to a SESN, the SESN Manager registers the corresponding plugins in the registry. A SESN Canvas organizes plugins in the graphical user interface.

A plugin is defined with three elements: (1) the grammar of the events contents it is interested in, (2) the processing modalities of the corresponding events, (3) the management of end-user interactions with the results of its process. For example, an image plugin can declare in its grammar that it requires events containing references to images, process the events by uploading the images, interacting with the end-user by showing the list of images and proposing a "promote" button for each image.

Plugins must conform to the C3PO plugins API, that contains one standardized function invoked by the event dispatcher. This method has one parameter: a reference to an event in the event registry. An event can be processed by many plugins. Each plugin declares regular expressions to be matched on the event content in order to be notified. The plugin registry gathers all regular expressions in a way that optimizes

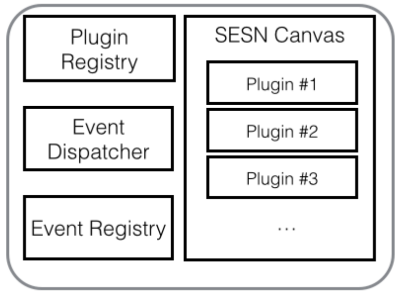


Fig. 4: C3PO Application Level Architecture

plugins selection. The event dispatcher uses this registry to trigger processing units in plugins. Regular expressions are flexible enough to accept any kind of identifier within events. In the current implementation, regular expressions provide a list of hashtags that must be present or absent.

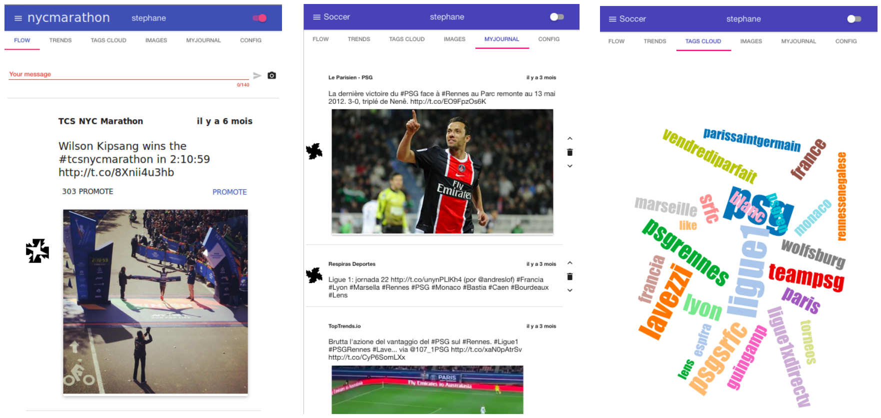
Plugins are developed using the AngularJS framework. User interactions in a plugin are developed as an AngularJS directive. This Directive may also include an event creation. This new event is thrown into the events queue of the network level input.

The graphical user interface is implemented using HTML5, CSS3, Javascript, AngularJS and Design Material. The binding between the user interface and the network level part is done using Apache Cordova. Several Cordova plugins have been developed in order to make accessible the functionalities provided by the framework in Javascript.

3.2 Application Specific Functionalities Plugins

In this section, we describe a subset of the plugins that have already been developed in the C3PO project.

- The "**Flow**" plugin takes all the received and emitted events and displays them as a list ordered with the last received first. The user can browse the list, and can click a "promote" icon that indicates that he believes this event is important. Each promote done by the end-user generates and throws a new Event including a #promote hashtag and the source event id. A screenshot is given on figure **??**.
- The "**Tags Cloud**" plugin takes all the incoming events and builds a tags cloud that reflects the frequency of each hashtag in the events set. Its user interface displays the tags cloud. A screenshot of the Tags Cloud plugin interface is given on figure **??**.
- The "**Image**" plugin selects events containing a reference to an image (url) and displays them as a list ordered with the last received first.
- The "**My Journal**" plugin displays all events promoted by the user. The user can manipulate the displayed list to raise, diminish or remove events. At the SESN termination it provides a summary of the happening. A scrennshot of the MyJournal plugin interface is given on figure **??**.
- The "**Popular**" plugin allows to show the most popular contents, based on the "promote" actions of the SESN members that have been received by the device. The



(a) Flow (b) My Journal (c) Tags Cloud

Fig. 5: C3PO application level is organized through plugins composition

processing of the most popular contents is done independently by each device. So two devices in two isolated macronets will not have the same list of contents in this plugin. Two devices subscribing to the same topic and being in the same micronet have a bigger chance to get the same list of contents in this plugin.

- The "**Vote**" plugin allows to submit a content to a collaborative vote, and to vote about a content. At the user interaction level, vertical swiping is used to vote: down to ignore, up to promote. The Vote plugin is associated to a "VoteResult" plugin that displays the results of the collaborative votes on each submitted content.
- The "**Live Recommendation**" plugin quickly identify the relevant contents over time. To achieve that, the underlying algorithm of this module attributes a score to each content based on both criteria, its popularity and its freshness. While the former exploits the number of promote, the latter captures the delay between the publication of the content in the system and their promote. This plugin uses a sliding window, and refresh the score to each content after a certain time window of size w. The following equation computes the score attached to content c at the current date d:

$$score(c,d) = \sum\nolimits_{P(c,d)\in[d0,d0+w]} \frac{P(c,d)\times w}{\Delta T}$$

where $P(c,d)$ is the number of promote that c has collected so far, $d0$ the starting date of the sliding window, and $\Delta T = d - d_t$ which reflects the freshness of c where d_c is the date when c has been created.
Each element of the sum follows the pattern of $\frac{1}{x}$. As a consequence, each element of the sum (i.e. each promote in the considered sliding window) increases if $\Delta T <$ w. It means when the promote has happened before the considered time window w. In contrast, this score is minored if the promote happens after the time window w. In addition, this score is weighted by the number of promote c has obtained so far. Lastly, the score associated to c is cumulative and sums the score computed to each

promote received in the considered time window *w*. The starting date of the time window is different for each content and is initialized at the reception of its first promote. The size of the time window defines the temporal granularity to follow the event, more the time window is small, more the event is detailed (by default, *w* is set at 10 minutes).
– The "**Config**" plugin helps user to activate or to deactivate each other plugin. When deactivated, the plugin is removed from the user interface but keeps its local context. If the user reactivates a plugin, it starts processing events with the previous saved context.

4 Related Work

Nowadays, it exists a plethora of social networks. Most of them rely on a centralized architecture and require connectivity to the Internet. Over the last years, decentralized social networks have been studied and developed by researchers and non-academic projects, especially to address privacy issues related to online social networks [**?**,**?**]. However, none of them have considered social networks formed spontaneously by a set of mobile devices.

Data sharing in opportunistic networks have been studied different projects. Haggle [**?**] takes advantage of contact opportunities and of device mobility in order to follow the "store, carry and forward" principle in the communication between devices. PodNet [**?**] implements a publish/subscribe model, where users can express their interests via keywords and receive content items accordingly. In Crowd [**?**], multimedia contents are exchanged via an online Web portal through Wi-Fi hotspots. SocialNet mainly aims at detecting the social links between people to improve the opportunistic forwarding of contents. Social links rely on community memberships, history of contacts, recurrent mobility patterns and/or user interests.

In C3PO, people implicitly express their common interests by physically attending happenings, and by being members of the same SESNs. Due to the ephemeral nature of the SESNs, the exploitation of information such as the history of contacts or the recurrent mobility patterns is not relevant. Projects Sarah and Scampi [**?**] both have defined a middleware platform to support communication and service provision in opportunistic networks. They have also implemented Android applications based on these middleware platforms. Sarah allows to discover neighbors, manage a list of contacts, exchange contents in a secure manner. Nevertheless, none of these projects provides incentive tools to support a collaborative production of rich multimedia contents during ephemeral events.

5 Conclusion

In this paper, we have introduced a new type of social networks dedicated to happenings, namely spontaneous and ephemeral social networks (SESNs). A SESN has the characteristic of relying on the opportunistic network formed by the devices of the happening attendees. We have shown how such an opportunistic network can be formed using the Bluetooth, Wi-Fi and Wi-Fi Direct interfaces of standard mobile devices, assembled in a novel network architecture. We have also presented the framework we have developed in project C3PO to support opportunistic communication in SESNs. This framework is able to manage the different wireless interfaces of mobile devices so as to form an opportunistic network dynamically. It is designed to be highly configurable and extensible, namely making it possible to define various message forwarding strategies. Preliminary evaluation results we have obtained for this framework in real conditions confirmed the validity of the approach.

6 Acknowledgments

C3PO project is supported by the French ANR (Agence Nationale de la Recherche) under contract ANR-13-CORD-0005. `http://www.c3po-anr.fr/`

References

1. C. E. Casetti, C.-F. Chiasserini, L. Curto Pelle, C. Del Valle, Y. Duan, and P. Giaccone. Content-centric Routing in Wi-Fi Direct Multi-group Networks. In *16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2015)*. IEEE, June 2015.
2. S. Frénot and S. Grumbach. An in-browser microblog ranking engine. In *Advances in Conceptual Modeling - ER ECDM-NoCoDA workshop*, volume 7518 of *LNCS*, pages 78–88, Florence, Italy, Oct. 2012.
3. F. Laforest, N. Le Sommer, S. Frï¿œnot, F. de Corbiï¿œre, Y. Mahï¿œo, P. Launay, C. Gravier, J. Subercaze, D. Reimert, E. Brodu, I. Daikh, N. Phelippeau, X. Adam, F. Guidec, and S. Grumbach. C3PO: a Spontaneous and Ephemeral Social Networking Framework for a Collaborative Creation and Publishing of Multimedia Contents. In *International Conference on Selected Topics in Mobile and Wireless Networking (MoWNeT'2014)*, pages 214–219, Rome, Italy, Sept. 2014. Elsevier.
4. N. L. Sommer, P. Launay, and Y. Maheo. A framework for opportunistic networking in spontaneous and ephemeral social networks. In *ACM MobiCom Workshop on Challenged Networks - CHANTS*, Paris, France, Sept. 2015. ACM Press.