HELIOS: FOUNDATIONS FOR PEER-TO-PEER SOCIAL MEDIA PLATFORM FOR PRIVACY-AWARE PEOPLE AND PROSUMERS

V. Ollikainen, M. Kylänpää, J. Kuusijärvi

VTT Technical Research Centre of Finland, Finland

ABSTRACT

The paper will present a technical platform, 'HELIOS', a toolset upon which peer-to-peer social media services can be easily tailored. Because of the p2p approach adopted, the social media services are independent of legacy social media platforms, such as Facebook or Twitter.

The paper will provide an overview of the underlying requirement layers and requirements for HELIOS, logical architecture, and the platform components as a whole, as well as provide in-depth description of the toolset available today and discuss about its future plans.

The paper will further summarize the most important features of the toolset, including novel concepts like contextual messaging, information overload control and blockchain-based rewarding for prosumers, not to forget privacy. Open-source repositories and community are supported beyond the end of the project by an interest group established in 2022.

The paper will also summarise, how the original principles of HELIOS evolved, including challenges experienced in mobile p2p.

INTRODUCTION

Social networking platforms are addressing the deep-rooted need for people communicating with each other. Social networking platforms have a substantial impact on society by becoming the primary place for people to share information and experiences. The motivations for participating in social media networks span beyond just being handy means for keeping in touch; sometimes they serve more imperative purposes, especially in areas of crisis.

One could argue that the main challenges of current social media lie in 1) centralization or authority (i.e., rules can be changed whenever convenient) 2) business models based on privacy intrusion and 3) priority in overall engagement (number of clicks), instead of contextual quality (importance of media at present).

Together with 14 partners in eight European countries, project HELIOS (1) was addressing these challenges by developing a modular toolset, 'HELIOS Platform' for peer-to-peer (p2p) social media services. Due to the p2p approach, the social media services are independent of legacy social media platforms, thus enabling independent tools, e.g., for customer relations or prosumer production. Furthermore, privacy-by-design is one of the key

properties in HELIOS, making it easy to create applications that comply with data protection regulations. Due to modularity, applications built upon the HELIOS Platform can be easily tailored. Four partners of the HELIOS Project have established a non-profit Platform Sustainability Initiative as founding members, and five supporting organizations have already joined it.

Prior to HELIOS, there have been a number of approaches for decentralized online social networks (DOSN), distributed systems for social networking with no or limited dependency on any dedicated central infrastructure as described by, e.g., Datta et al 2010 (2). Distribution may take place on server level, giving users freedom to pick a server to sign up; these DOSNs are referred as federated networks. For instance, Diaspora, introduced by Bielenberg et al 2012 (3), has federated servers that enable user communications, and so has MASTODON, presented in a publication by Zulli et al 2020 (4). In contrast to federated networks, P2P operates without servers at all, having nodes communicating directly to each other. HELIOS introduced a toolset to assist developers to create serverless social media services using p2p communications — in a mobile platform.

Being mobile, specific challenges in maintaining p2p services exist, as there cannot be any assumptions on peers being online; mobile devices are randomly switched off, roaming or set to flight mode. While one-to-one communications would only need retransmitting, group services are essentially more complex. The technical issues were resolved to the extent that the project launched unsupervised piloting in 2022. The corresponding toolset with sample apps is available as Open Source, in HELIOS GitHub (5).

The Project followed ‘Android first’ principle, to focus on an operational toolset for at least one major operating system, as supporting low-level differences between iOS and Android would have duplicated the effort during the course of the project. HELIOS Platform is available as Open Source and it currently supports Android versions 9 and above. Having said this, further development for iOS will eventually be a decision by the supporting community.

HELIOS REQUIREMENTS AND ARCHITECTURE

HELIOS has adopted a modular approach, in which platform functionality is allowed to grow by adding new modules to the platform. The lowest level of the architecture comprises basic functionalities upon which a privacy-enabled social media service can be built, functionalities extended by add-on modules. Implementation of most of the functionalities were preceded by academic research; details of the studies and developed algorithms have been published in conferences and scientific journals, as listed in HELIOS Publications (6).

Requirement layers for the platform

The basis for HELIOS architecture was designed to meet technical requirement specifications, based on a) the project's objectives, to meet funding criteria and b) use cases. The requirements were organized in two directions: Vertical layers covered requirements present in *most* of the use cases and objectives, and Horizontal layers covered requirements present in *some* of the use cases and objectives of the project. The layers are presented in Figure *1*.

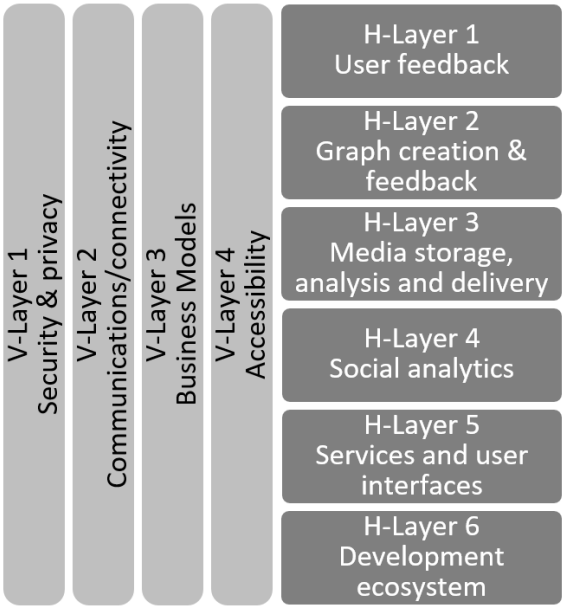

Figure 1. Requirement layers of HELIOS

There are four vertical layers. *Security and privacy* requirements are the most important layer in terms of user's privacy and security. The layer includes requirements for authentication, access control, anonymity, encryption, and integrity. Some of these requirements are also linked to European General Data Protection Regulation 2016 (7).

Communication and connectivity requirements include a requirement for truly distributed communications architecture without any servers, as the main design principle in communication and connectivity of the HELIOS platform is the p2p paradigm.

There are also *Business model* requirements. There should be an incentive model to grant users for their activity and contribution in the network.

The fourth vertical requirement is *Accessibility*. The accessibility requirements are based on the four high-level principles presented in Web Content Accessibility Guidelines (8), stating that user interfaces should be perceivable, operable, understandable, and robust.

The horizontal requirements are more specific requirements typically important only for certain HELIOS modules.

User feedback requirement states that there should be response for the user to their actions, or the actions of their friends, while using the social media platform. This is closely related to *Services and User Interfaces* requirements that contains functional and non-functional requirements for HELIOS modules.

Graph creation and management requirement specifies that social relationships are represented in a social graph utilizing so called Contextual Ego Network (CEN) model. *Social analytics* requirements define analytic services with the aim to analyse the social graph by taking into account the structure of the contextual ego network, including friends and relationships, and the evolution of the social relationships.

As a further requirement, HELIOS must provide features in the domain of Media Services in a consistent manner. *Media storage, analysis, and delivery* requirements contain requirements for video services especially requirement to utilization of the p2p infrastructure.

Development ecosystem requirements comprise requirements concerning the development model requiring scalable, modular, and extensible architecture with interoperability and mobile platform support.

Layered architecture

Providing the core functionalities, *HELIOS Core modules* implement basic functionality like messaging, storage, user context, profile, social graph, and services related to security, privacy, and trust. These modules can be used as building blocks for HELIOS applications, or they can be utilized to build new upper layer modules.

Optional modules are called *HELIOS Extension modules* that may or may not utilize HELIOS Core modules. Examples of these extension modules available in GitHub (5) and summarized in the chapter below. All HELIOS modules provide APIs that are available for HELIOS applications and other modules. In HELIOS samples, third party applications can exchange data with HELIOS using Android intents.

The layered architecture is illustrated in Figure 1: HELIOS Core modules are built utilizing Device Operating System (the first release is Android-based) and underlying hardware features.

HELIOS Extension modules are built using both HELIOS Core modules and features of the Device Operating System.

HELIOS applications are using modules from HELIOS Core and selected HELIOS extension modules, as building blocks supplemented with Device Operating System service.

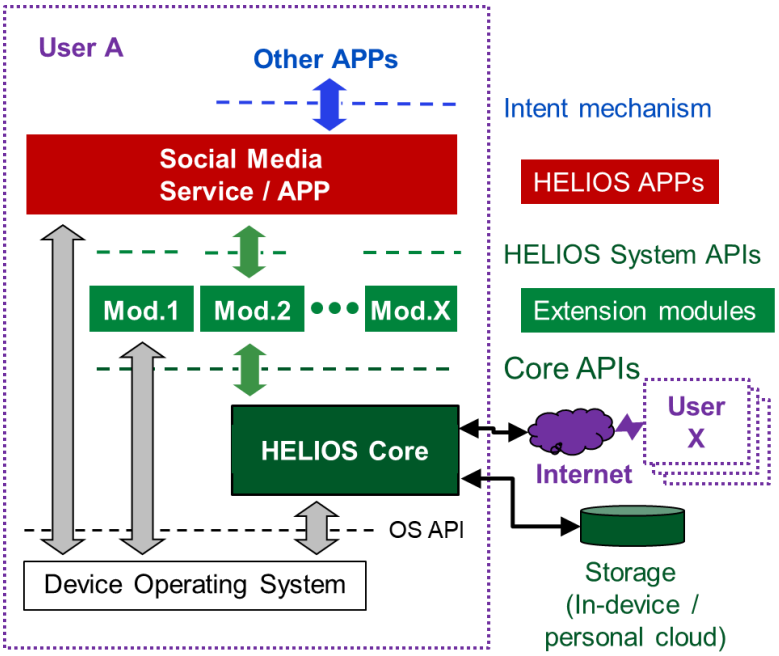

Figure 2. HELIOS logical architecture.

As a notice, the Core provides an abstraction for data storage, making it agnostic, whether the data is physically on the device or e.g. in a personal cloud, supporting several devices the user may have.

Core modules and functionalities

HELIOS Core modules implement basic functionality and provide APIs to other HELIOS modules and applications.

Communication Manager (messaging) module provides HELIOS Messaging API that can be used to communicate with other HELIOS nodes. The implementation includes API calls

that can be used both in publish-subscribe based group communications and in one-to-one direct messaging between the nodes.

Security and Privacy Manager module provides HELIOS Security and Privacy API that provides cryptographic services to HELIOS modules and applications. The module supports symmetric and public key encryption, signing and verification, and key generation functionality. Additionally, there is a basic access control manager available for creating local access rules. Blockchain access control is available as an extension module.

Profile Manager module handles user account creation and creation of associated keys. At least one key pair is needed for signing and verification of electronic signatures and another public/private key pair is needed for encryption and decryption.

Contextual Ego Network Manager module is responsible for the management of social graph related information stored in the local devices.

Context Manager module handles context-related monitoring and reasoning for the user. It provides a framework for implementing contexts and context detection with core implementations. The module also includes information overload control discussed later.

Trust Manager module implements a privacy-preserving trust evaluation model. The module is utilizing the Contextual Ego Network Manager to retrieve relationships and contexts.

Personal Data Storage Manager provides a unified file storage abstraction for HELIOS applications without binding to specific storage location. The module supports local data (mobile device) and HELIOS personal storage elements that allows providing storage in a user controlled personal cloud.

Available extensions modules

There are also multiple HELIOS extension modules, provided as open source in (5). The extension modules provide additional features on top of the basic social media building blocks provided by the core modules. They might not be required in all applications, may also have an impact on the size of the installation package, and can therefore be included if needed. The following presents the currently available HELIOS extension modules in GitHub, with details in (6).

Media Streaming extension modules contain extension modules for file transfer, live video streaming, video call, playing streaming video, and for HTTP Live Steaming (HLS).

Group Communication Services (GCS) consists of multiple extension modules that are built using HELIOS core modules and to expose several communication management functionalities through a Decentralized Group Management framework.

Content Aware Profiling extension module can be used to perform semantic analysis of image collections.

Neurobehavioral classifier extension module can be used to analyse the emotional statements of the users. The module can be used to calculate an emotional score to be used in the trust model.

Rewarding extension module can be used to incentivize HELIOS users in the use of the platform, e.g. by compensating creation of interesting, even prosumer content. The implementation is built upon Ethereum in Alastria blockchain infrastructure.

Blockchain Access Control enables accessing control to resources (i.e., documents, media content) using HELIOS Blockchain and smart contracts, using a HELIOS wallet.

Extension modules for social graph include *Social Graph Mining* and *Social Community Detection* extension modules.

BUILDING A SOCIAL MEDIA PLATFORM

HELIOS software modules are open-source software available in GitHub repositories and mostly licensed with Apache 2.0 license. Each module contains a description that covers main use cases of the module and instructions on how to integrate the module to developer's Android Studio development environment project. The documentation also contains code fragments that describe the usage of the modules. HELIOS APIs are documented using normal Javadoc-style documentation conventions. There are also sample applications and video tutorial describing the functionality of the modules.

HELIOS has developed building blocks that can be used to build new innovative social media applications. It is expected that new developers are able to combine these modules in various ways and build exciting new applications on top of these modules.

Figure *3* illustrates a screenshot of a sample app, having fixed and user-defined 'channels', i.e. contextual discussion forums. Context of each channel can be explicitly defined and automatically detected, for instance by location. Information overload control takes advantage of this information, by optionally limiting number of incoming notifications, if a channel of incoming message is irrelevant to the current context of the user.

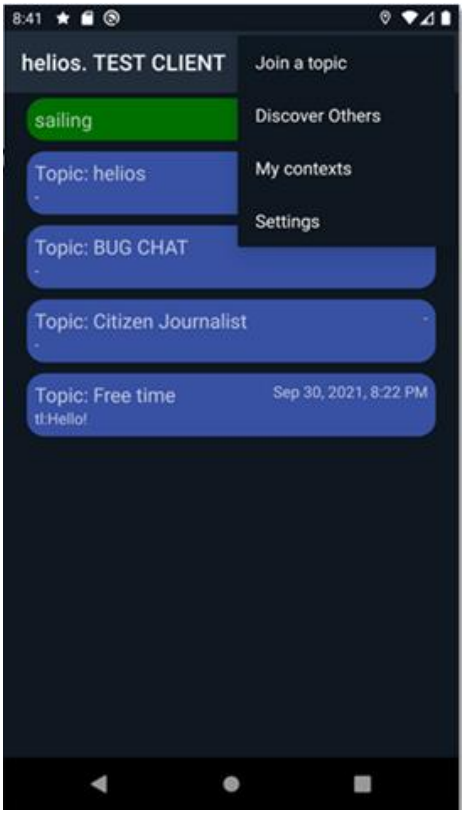

Figure 3. Screen view of the HELIOS Android application utilizing HELIOS Core modules.

EVOLVEMENT OF THE KEY CONCEPTS AND FUTURE DIRECTIONS

In the preparation of the project in late 2017 and early 2018, certain assumptions were made regarding upcoming technology, which were conveyed to project proposal for the European Commission in April 2018 and further on to the project plan (aka Declaration of the Action, 'DoA') later the same year.

Static linking of modules

One of the assumptions was availability of efficient Inter-Process Communications (IPC) on Android platform, as Android is fundamentally based on Linux. The original vision was to operate HELIOS Core as a background service, even having a role as a switchboard for packetized communications between modules, and in some cases providing easy means to install modules as needed. This would also have enabled each device running a single HELIOS core, to which different modules and applications would have made system calls without need for statically linking basic functionalities.

However, new versions of Android applied more stringent privacy policies, making efficient IPC less feasible, and eventually leading to the current architecture of statically linking HELIOS core and extension modules selected by the applications developer. This is how HELIOS became a toolkit, in contrast to Android OS extension in the original concept.

P2P communications on mobile

Several p2p protocols and implementations were examined, such as Bramble (9), webRTC (10), and libp2p (11). Since Libp2p supported several transports, natively support security & privacy solutions, distributed hash tables, had proven usage in InterPlanetary File System (IPFS), most active developer community and permissive licensing, and included implementations in multiple programming languages, especially JavaScript and Go, it was selected for HELIOS p2p transport layer.

In practice, due to active development and feature addition, libp2p also suffered from documentation lagging behind implementation in parts, varying maturity level between components. Like any p2p network, libp2p requires bootstrap node(s) to start/connect to the initial p2p nodes. Also relay nodes are required to overcome situations where p2p nodes cannot connect to each other directly, e.g., due to firewalls or Network Address Translation (NAT).

Based on libp2p, the project provided a simple bootstrap and relay nodes. In the first phase, the nodes were fully accessible via other libp2p nodes as well, as the bootstrap/relay nodes need to be able to be connected. After becoming flooded by other libp2p nodes than HELIOS, causing stability issues, the project implemented support for the available pnet protocol, a pre-shared key private network, which effectively made the HELIOS network only accessible with HELIOS pre-shared key. At the same time, benefits from larger libp2p network was lost.

At the very last moths of the project, Streamr (12) transport layer was successfully implemented to replace basic functionalities provided by libp2p. For reference, in HELIOS GitHub (5), Streamr is implemented in a sample app called 'SpringAPP'.

Unsupervised piloting in 2022

Throughout the project, subsequent waves of COVID-19 with lockdowns narrowed down options to pilot HELIOS with end users. Internally, the project team was testing the platform from October 2020 to the end of the project in 2022.

It was only in 2022 that an opportunity to pilot sample applications in unsupervised conditions (i.e. users were essentially on their own) during a one-month period from mid-January to mid-February. Piloting was organized as part of Vihti Secondary School (Finland) media education, in a class of 20+ pupils introduced to citizen journalism.

The piloting had two applications developed in the project, Citizen Journalism ('CJ') app by project partner Swiss TXT AG, and — connected with intents — the most recent HELIOS social media app called 'WinterAPP'. For stability, only the most mature extension modules were enabled in WinterAPP. WinterAPP was used in planning content production, whereas CJ was used for content acquisition.

It should be noted that libp2p (11) was in use in the pilot, while there was concurrent study of applying Streamr (12) transport layer into HELIOS.

As a result, users did not report any technical difficulties, and the overall evaluation was on a par with Facebook, albeit beyond more popular services like TikTok or Snapchat. This proved the HELIOS platform's functionality.

Future directions

In 2022, the project established the HELIOS Community, with an intent to promote joint research and business development activities in the field of decentralized social media services. Since the Community consists of both industrial and academic partners, it is likely that it will address both academic research (lower technology readiness) and industrial interests.

Regarding industrial interests, adding support for iOS seems essential for commercial applications. The project received questions about the implementation, which may be feasible from the lessons of having one operating system (Android) now operational.

As a further note, distributed and privacy-preserving environments are inherently more challenging, when there is need for recommendations or content discovery. P2P fits well with the recommendation architectures presented by Ollikainen 2019 (13), as a publication of the same project HELIOS.

CONCLUSIONS

HELIOS proved that it is possible to create privacy-aware social media services on at least one mobile platform, using p2p communications instead of federated or centralized servers. The project was able to run an unsupervised one-month pilot with twenty users in early 2022. The Open Source codes used in the pilot, as well as extensions to HELIOS, are available in GitHub (5) for any developer who wants to experiment with the technologies created in the project.

From the experiences from the project, for commercial or mission-critical applications the underlying p2p transport layer should be carefully selected, with roadmaps proactively addressing changes in operating systems. This is probably an issue that should not be left

to open source communities (such as libp2p) that typically are relying on volunteers' interests. During the last months, HELIOS had productive co-operation with Streamr Network, which is an example of an industrial alternative. The resulting open source codes are equally provided in GitHub (5).

REFERENCES

1. HELIOS: A context-aware distributed social networking framework. Project pages of the project HELIOS, funded by the European Commission under Grant Agreement #825585. Accessed in May 2022 at https://helios-social.com/

2. Datta, A., Buchegger, S., Vu, LH., Strufe, T., Rzadca, K., 2010. Decentralized Online Social Networks. In: Furht, B. (eds) Handbook of Social Network Technologies and Applications. Springer, Boston, MA. pp 349 to 378. https://doi.org/10.1007/978-1-4419-7142-5_17

3. Bielenberg, A., Helm, L., Gentilucci, A., Stefanescu, D. and Zhang, H., 2012. The growth of diaspora-a decentralized online social network in the wild. 2012 proceedings IEEE INFOCOM workshops. pp. 13 to 18.

4. Zulli, D., Liu, M., and Gehl, R., 2020. Rethinking the "social" in "social media": Insights into topology, abstraction, and scale on the Mastodon social network. New Media & Society, 22(7). pp 1188 to 1205.

5. HELIOS GitHub. Open source codes and instructions to build HELIOS applications. Available at https://github.com/helios-h2020

6. HELIOS Publications. Accessed in May 2022 at https://helios-h2020.eu/category/publications/

7. EU General Data Protection Regulation (GDPR): Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), OJ 2016 L 119/1. Available at https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679

8. Web Content Accessibility Guidelines. Accessed in May 2022 at https://www.w3.org/WAI/fundamentals/

9. Bramble specifications. Accessed in May 2022 at https://code.briarproject.org/briar/briar-spec/-/tree/master/

10. WebRTC. Real-time communication for the web. Accessed in May 2022 at https://webrtc.org/

11. Libp2p. A modular network stack. Accessed in May 2022 at https://libp2p.io/

12. Streamr. The decentralized real-time data network. Accessed in May 2022 at https://streamr.network/

13. Ollikainen, V., 2019. Networked Collaborative Recommendation Architecture. Paper presented at IBC2019 Conference, Amsterdam, Netherlands. Accessed in May 2022 at https://www.ibc.org/download?ac=10484

ACKNOWLEDGEMENTS

The authors would like to thank the rest of the HELIOS project team for creating the platform. We also would like to thank Matthew Fontana from Streamr Network AG, for fluent co-operation in final stages of HELIOS development.