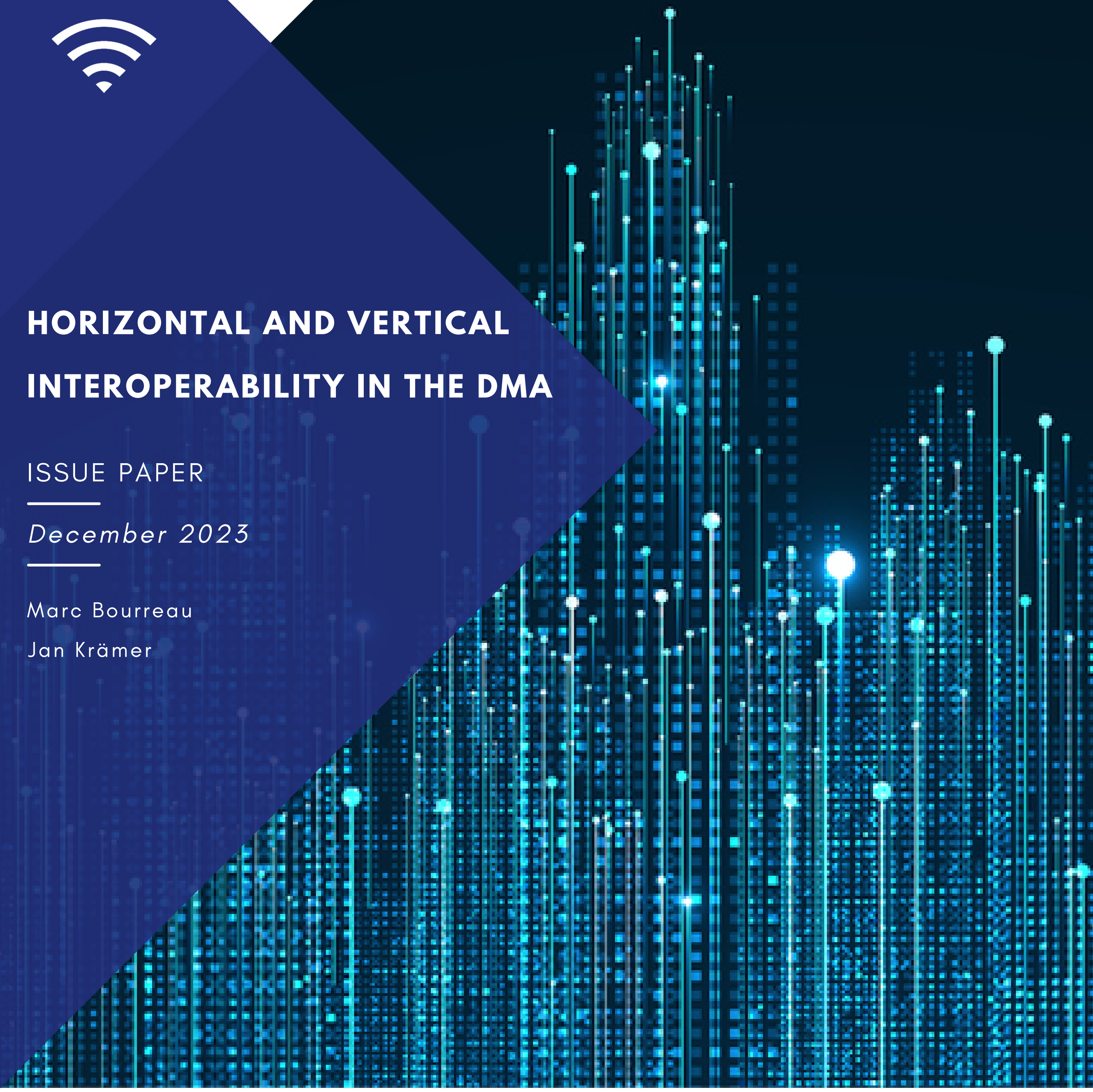
cerre Centre on Regulation in Europe



HORIZONTAL AND VERTICAL INTEROPERABILITY IN THE DMA

ISSUE PAPER

December 2023

Marc Bourreau

Jan Krämer

As provided for in CERRE's bylaws and procedural rules from its “Transparency & Independence Policy”, all CERRE research projects and reports are completed in accordance with the strictest academic independence.

The project, within the framework of which this report has been prepared, received the support and/or input of the following CERRE member organisations: Amazon, Apple, Arcep, Booking.com, DuckDuckGo, Epic Games, Google, Huawei, MFE-MediaForEurope, Meta, Microsoft, Mozilla Corporation, and Qualcomm. However, they bear no responsibility for the contents of this report. The views expressed in this CERRE report are attributable only to the authors in a personal capacity and not to any institution with which they are associated. In addition, they do not necessarily correspond either to those of CERRE, or of any sponsor or of members of CERRE.

© Copyright 2023, Centre on Regulation in Europe (CERRE)
info@cerre.eu – www.cerre.eu

TABLE OF CONTENTS

ABOUT CERRE

Providing top quality studies and dissemination activities, the Centre on Regulation in Europe (CERRE) promotes robust and consistent regulation in Europe's network and digital industries. CERRE's members are regulatory authorities and operators in those industries as well as universities.

CERRE's added value is based on:

- its original, multidisciplinary and cross-sector approach;
- the widely acknowledged academic credentials and policy experience of its team and associated staff members;
- its scientific independence and impartiality;
- the direct relevance and timeliness of its contributions to the policy and regulatory development process applicable to network industries and the markets for their services.

CERRE's activities include contributions to the development of norms, standards and policy recommendations related to the regulation of service providers, to the specification of market rules and to improvements in the management of infrastructure in a changing political, economic, technological and social environment. CERRE's work also aims at clarifying the respective roles of market operators, governments and regulatory authorities, as well as at strengthening the expertise of the latter, since in many Member States, regulators are part of a relatively recent profession.

ABOUT THE AUTHORS

Marc Bourreau is an Academic Co-Director at CERRE and Professor of Economics at Télécom Paris (Institut Polytechnique de Paris). He is affiliated with the Interdisciplinary Institute for Innovation (i3) for his research, which focuses on competition policy and regulation, digital markets, and telecommunications. Marc holds a Ph.D. in Economics from the University of Paris Panthéon Assas.

Jan Krämer is an Academic Co-Director at CERRE and a Professor at the University of Passau, Germany, where he holds the chair of Internet & Telecommunications Business.

Previously, he headed a research group on telecommunications markets at the Karlsruhe Institute of Technology (KIT), where he also obtained a diploma degree in Business and Economics Engineering with a focus on computer science, telematics and operations research, and a Ph.D. in Economics, both with distinction.

His current research focuses on the role of data for competition and innovation in online markets and the regulation of online platforms.

1. GENERAL INTRODUCTION

This issue paper discusses some key issues and trade-offs that may arise in implementing the horizontal and vertical interoperability provisions of the DMA.

It builds on two previous CERRE papers, the 2022 CERRE report “Interoperability in Digital Markets”, which discusses the pros and cons of mandating horizontal and vertical interoperability from an economic perspective, and the 2023 CERRE paper “DMA Horizontal and Vertical Interoperability Obligations”, which addresses several issues of scope and implementation related to interoperability.

In this paper, we first discuss the implementation of the horizontal interoperability provision in the DMA (Article 7). We argue that horizontal interoperability will require proper management of user consent and careful interface design. We also argue that achieving horizontal interoperability poses significant technical challenges to: (i) resolve identities across providers; (ii) establish secure connections; and (iii) deal with malicious users. Solving these technical challenges will raise trade-offs for which there are no easy choices.

Second, we briefly discuss general principles for the implementation of vertical interoperability provisions, i.e., specifically Articles 6(4) and 6(7). Here, we argue for five principles, (i) screening of access requests, (ii) screening of access seekers, (iii) gatekeeper-led definition of interfaces, (iv) equivalence of input, and (v) a neutral choice architecture. We also emphasise that there can be interactions between the four principles so that they must be evaluated in concert, and not in isolation.

2. HORIZONTAL INTEROPERABILITY IN THE DMA

Article 7 of the DMA introduces a horizontal interoperability obligation for gatekeepers providing **number-independent interpersonal communications services** (NI-ICS).

This access obligation covers only a subset of **“basic functionalities**” of the messaging services offered by the gatekeepers. Within six months after the designation decision, interoperability should be available for text messaging and the sharing of images, videos and other files between individual users. In a second step, within two years of the designation decision, group chat should also be interoperable, and within four years, voice and video calls. Access must be provided **upon the request** of an access seeker and be **free of charge**.

The main objective of horizontal interoperability is to **improve the contestability of digital markets**. In the absence of interoperability, incumbent players (the gatekeepers) offering messaging services benefit from strong network effects that limit the contestability of the market. Interoperability is expected to level the playing field between incumbents and new entrants, as network effects are then shared among competitors and constitute a public good. We therefore expect **strengthened competition and reduced barriers to entry** in the market for messaging services. The successful entry of new players via (interoperable) messaging services may also allow them to expand gradually and develop their own ecosystem of complementary products. Therefore, opening up the messaging market to competition could also have a wider impact on digital markets.

At the same time, **horizontal interoperability may reduce multihoming** in messaging apps, which is another important driver of competition in digital markets. Moreover, as interoperability is also possible between gatekeepers, it could even **strengthen their position** vis-à-vis new entrants by making them more central for users. Overall, therefore, the impact of the horizontal interoperability provision on the contestability of digital markets remains uncertain.

3. HORIZONTAL INTEROPERABILITY: CONSENT MANAGEMENT, GROUP CHATS AND DESIGN OF INTERFACES

3.1. Consent for discoverability and interoperable communication

To implement interoperability between messaging services, **user consent** may be required at two different steps: for user discovery and for interoperable communication. We discuss both aspects in turn.

An important first step for the implementation of horizontal interoperability is to define how **user discovery** works, that is, to specify "the process of learning which service(s) a user uses and/or prefers" (Blessing & Anderson 2023).

Consider the **following example**. Alice wants to communicate with Bob, but they use different messaging services. Alice uses the service of a third party, *A*, while Bob uses the service of Gatekeeper, *B*. The only way Alice can reach Bob on *B*'s network is through interoperability. Gatekeeper *B* has published a reference offer (Article 7(4)), *A* has requested interoperability from *B* (Article 7(5)), and interoperability between *A* and *B* is operational. However, Alice must now "discover" that Bob is using *B*'s service in order to communicate with him via interoperability.

Bob could inform Alice of his identity on *B* during a **face-to-face meeting**, for example by using a QR code. By revealing his identity on B to Alice, Bob implicitly consents to being discovered by Alice and to communicating with her. This seems like the simplest solution from a consent perspective. One could also argue that it is sufficient to implement discoverability in this way, since the DMA does not require a specific solution for discoverability (it does not even mention it). However, if users have to share contact information, there is a risk that interoperability will be little used. So, for interoperability to be effective, as required by the DMA, one could argue that user discovery needs to be **automated**, meaning that Alice's application (*A*) should be able to discover that Bob is using *B*.

We now focus on this case where **user discovery is automated**.

A key question is whether Bob should give his **consent** to be discoverable on *B*, that is, opt-in to being discoverable by third-party applications like *A*.[1] Or should Bob instead be discoverable by default, while still having the ability to opt-out? BEREC, for example, considers this an open question (BEREC 2023, p. 25): "With regard to the data sharing and authentication among different interoperable services, the consent of the users to approve the exchange and processing of data to a third-party service needs to be clarified, e.g., if opt-out is possible or opt-in is obligatory." Moreover, Article 7(7) only requires that users "shall remain free to decide whether to make use of the interoperable basic functionalities," without specifying whether this should be done through an opt-in or opt-out regime.

[1] In this section, we focus on the issue of consent for discoverability. In Section 3.1, we also discuss the technical challenges of resolving identities across providers.

Requiring users to **opt in to discovery** imposes a cost on users, who may then prefer to remain undiscoverable (the default). So, there would be a risk that interoperability would be ineffective. In fact, email addresses are openly discoverable, which helps interoperability work seamlessly. Similarly, one of the reasons for WhatsApp's success is that user discovery is achieved automatically by searching users' address books and checking for contacts who use the application (WIK 2022).[2] For these reasons, some argue for an opt-out regime for interoperability. For example, users could be notified that they are discoverable and given clear instructions on how to opt out.

However, we believe that **users should be opted out of user discovery by default**, and thus should give their consent explicitly for discoverability.

The main reason for an opt-out regime is **user privacy**. Since the implementation of interoperability may imply the exchange of personal (meta) data between providers (see Sections 3 and 4 for a detailed discussion), the explicit consent of users may be required for privacy reasons alone. This is mentioned in the DMA, with Article 7(8) stating that the collection and exchange of data for the purposes of interoperability must comply with the GDPR and the ePrivacy Directive. Similarly, Recital 64 states that "interoperability should be without prejudice to the information and choices to be made available to end users of the number-independent interpersonal communication services of the gatekeeper and the requesting provider under this Regulation and other Union law, in particular Regulation (EU) 2016/679 [the GDPR Regulation]."

It is also a matter of **transparency** for users. With an opt-out regime, some users may simply not be aware that they are "discoverable" by users of third-party messaging apps. The opt-in regime ensures that users are fully aware that they can be discovered.

Having established that users should give their consent for discoverability, the next question is **how consent should be given**.

One possibility is to ask users to give **consent for each gatekeeper messaging service** that they use. For example, Bob should give his consent to be discoverable on service *B*. If Bob uses service *C* from another gatekeeper, he should be able to give his consent to be discoverable on *C* separately. Indeed, some users may want to use different messaging apps for different purposes (e.g., one app for work and another for communicating with friends and family), and in some cases may only want to be discoverable "on-net" in a particular app (Blessing & Anderson 2023). Note that in the context of Article 7 DMA, user discovery only concerns gatekeeper messaging services, and thus in principle a few services, so such a solution seems feasible.

Consent could also be more fine-grained. For example, Bob might be perfectly fine with being discovered on *B* and *C* by users of the third-party application *A*, but he might be extremely reluctant

[2] Note that it is possible to send a message to someone who is not a contact (see, e.g., https://www.forbes.com/sites/prakharkhanna/2022/12/22/how-to-send-message-on-whatsapp-without-saving-a-number/?sh=59dca12d5c87https://www.forbes.com/sites/prakharkhanna/2022/12/22/how-to-send-message-on-whatsapp-without-saving-a-number/?sh=59dca12d5c87), though WhatsApp gives users some control over who can contact or call them. So, having a phone number as an ID facilitates discovery, but phone numbers only appear as contacts if they are provided by the user (e.g., by uploading a phone book).

to be discovered by users of another third-party application *D*. In other words, consent could be given for **each pair of messaging services**, involving a gatekeeper's service and a third-party service. However, this approach may be too complex for users.

This also raises the question of **which providers are entitled to discoverability**. For security or privacy reasons, it would make sense to restrict discoverability to "legitimate" third parties. One approach would be to have the gatekeepers define in their reference offers the security and privacy standards that the providers should meet in order to have their interoperability request accepted (under the scrutiny of the European Commission). Alternatively, an industry body could also play this role -- because of the negative externalities that malicious providers could create, the industry as a whole would have an incentive to coordinate and define such standards. On these questions, see also our discussion in Section 4.2.

Alternatively, instead of filtering discoverability by platform, users could opt in to **discoverability by user or user type**. For example, Bob could agree to be discoverable by all of his contacts, regardless of which service they use. However, this solution is unlikely to be practical, unless the same identifier is used across services (see Section 3).

A practical solution should give the user some flexibility without being too complex. **A possible solution** in this regard would be to notify Bob, when he opens his gatekeeper messenger service B, that B is now interoperable with the third-party service A, and to ask him if he wants to take advantage of this interoperability feature. In this way, Bob would "opt in" to interoperability, while being forced to make an informed decision.

In all cases, users must have the **ability to revoke discoverability**. For example, if Bob no longer wants to be discoverable (e.g., because he feels he has received too much spam from third parties), he should be able to do so. Concretely, this would mean that Bob is no longer discoverable on his app *B*, but also that third parties who previously discovered Bob should now "forget" that he uses *B*.

Now, consider that Bob has consented to be discovered on his gatekeeper app *B*. Alice can now communicate with him via interoperability from her third-party app *A*. In this case, should Bob also **consent to an interoperable communication** with Alice? Today, messaging services handle this question differently. For example, Wire and Element require consent to communication, while many other services do not.

One could argue that it is like on a telephone network: if Bob does not want to talk to Alice, he just does not reply. However, there is a difference; Alice's message can show up even if Bob does not want to talk and it could be spam or an abusive message. Of course, Bob could "block" Alice, but this would only be possible after the message has appeared, with all its possible annoyances or risks.

The importance of this second level of consent probably depends on how specific the consent for discoverability is. Let's say Bob has to agree to be discoverable by all users of all third-party platforms. Then, there is probably a role for consent to interoperable communications for Bob to filter incoming communications. Conversely, if Bob has consented to be discoverable specifically by Alice, then there is less need for consent to interoperable communication with her.

3.2. Group chats

So far, we have discussed the impact of discoverability on one-to-one communications. However, within two years of the designation decision, interoperability should also apply to **group chats**. We argue here that group chats present additional challenges.

Consider the **following example** (from Wiewiorra et al., 2022). As shown in Figure 1 below, there are two gatekeepers, *A* and *B*, that are subject to interoperability requirements, and a third party, *C*. We assume that *C* is interoperable with *A* and *B*, but *A* and *B* are not interoperable (because they have chosen not to request interoperability from each other).

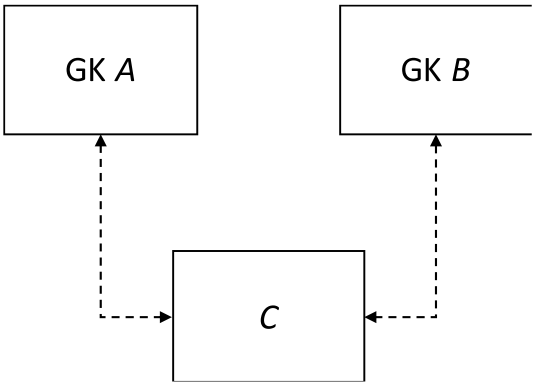


Figure 1: Example 1: third party C is interoperable with gatekeepers A and B, but A and B are not interoperable (example from Wiewiorra et al., 2022).

There is a group of users of *C* who want to chat with a user of *A*. Since C is interoperable with *A*, this group chat can work. However, what happens if they want to invite a user of *B* to join the group chat? *A* users and *B* users may have separately given their consent to be discoverable by *C* users and to communicate with them. However, *A* users have not given their consent to be discoverable by and communicate with *B* users, and vice versa.

Consider now this other scenario, with two gatekeepers (*A* and *B*) and three third parties (*C*, *D* and *E*). *C* is interoperable only with *A*, while *D* is interoperable with both *A* and *B*. Finally, *E* is not interoperable with any gatekeeper. Although there are five providers, four of which are interoperable with some others, in this scenario, group chats cannot occur with more than two providers.

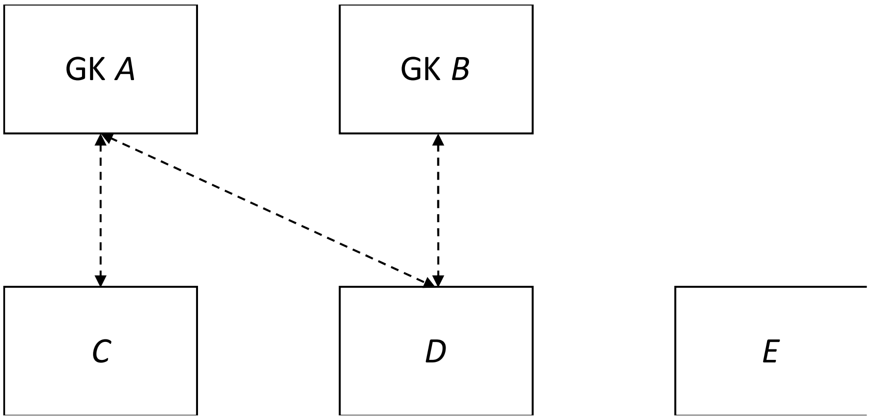


Figure 2: Example 2: third party C is interoperable with gatekeeper A, third party D with A and B, and third party E with no gatekeeper.

These two examples show that there are scenarios where group chats will not work effectively if users opt in to discoverability.

Thus, one could argue for making user discoverability the default (and thus adopting the opt-out regime). However, this does not seem feasible to us for privacy reasons, as explained above, nor is it desirable to maintain transparency for users.

3.3. Design of interfaces

Implementing interoperability also requires new interface design, both for the gatekeepers and for the third parties who will request and use interoperability. And, as Blessing & Anderson (2023) note, "[i]Interface design is critical if messaging interoperability is to enhance, rather than degrade, the user experience."

There is a concern that gatekeepers might choose a bad design to make interoperability ineffective. This is related to the more general issue of choice architecture, which is discussed in more detailed in another CERRE paper.[3]

We discuss here two specific topics that raise design concerns: (i) the choice of communication channel; and (ii) possible alerts to users for interoperable communications.

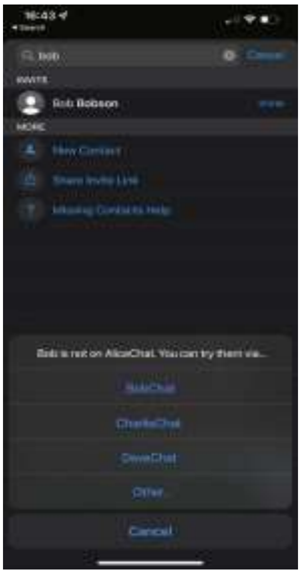
3.3.1. Choice of communication channel

Let's say Alice wants to communicate with Bob. Since they use different messaging services, this communication is done through interoperability. However, Bob uses multiple messaging services, all

[3] See Fletcher, A. (2023), "Choice Architecture for End Users in the DMA," CERRE Issue Paper.

of which are interoperable with Alice. In this case, who should decide which service to use to terminate the communication on Bob's side?

If Alice is the one making the decision, there should be an interface in her application to select which service to use on Bob's side. The figure below shows an example of a possible design from Matrix.



Bob is not on AliceChat. You can try them via...

Figure 3: Alice wants to communicate with Bob. She is using AliceChat (a gatekeeper app). She is prompted to choose a service that Bob uses. Source: https://matrix.org/blog/2022/03/29/how-do-you-implement-interoperability-in-a-dma-world/

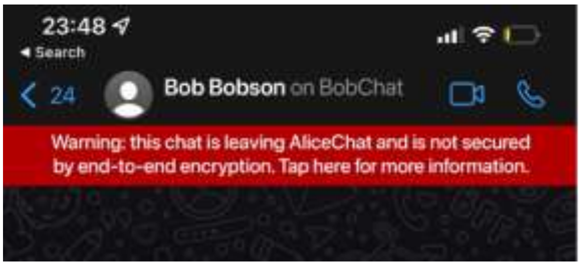
Alternatively, we could argue that it is up to Bob to decide. In this case, there should be an interface where Bob specifies his preferred service to receive interoperable communications. Should it be the same for every contact? Or will Bob be able to fine-tune it by selecting an interoperable channel for each contact?

Finally, the preferences of both the initiator of the communication (Alice) and the destination (Bob) could be taken into account to select the communication channel. For example, Alice and Bob could rank their preferred services and an algorithm could select the best service for them based on a decision rule. However, such a solution could be complicated to implement in practice.

To the extent that Bob has given his consent to communicate with Alice, we would tend to argue that the solution to this problem is not critical. So there could be a default to avoid users having to make this choice for every call, with the possibility for users (Alice and/or Bob) to override the default if they wish.

3.3.2. Alerts to users for interoperable communications

Interoperability may involve privacy or security trade-offs (see our discussion of these trade-offs in Section 4.2). When a user is about to make an interoperable communication, should the user be warned of the possible negative privacy or security consequences? Matthew Hodgson of Matrix argues that "unless everyone speaks the same end-to-end encrypted protocol", the user should be warned that "the conversation is no longer end-to-end encrypted", for reasons of "user experience and transparency" (see the example below).[4]



23:48

◂ Search

‹ 24 Bob Bobson on BobChat

Warning: this chat is leaving AliceChat and is not secured by end-to-end encryption. Tap here for more information.

Figure 4: Alice wants to communicate with Bob. Alice is on AliceChat, but Bob is on BobChat, so she will make an interoperable communication. She is warned that this communication will not necessarily be as secure as on AliceChat. Source: https://matrix.org/blog/2022/03/29/how-do-you-implement-interoperability-in-a-dma-world/

However, such alerts or warnings could be frightening to users and discourage them from interoperable communications. One solution could be to ask users to consent to an interoperable communication with another user when it is about to be established, as we discussed above. The need for such alerts also depends on how strict the screening of interoperable providers is. If only providers that meet certain security standards can become interoperable, security alerts may not be necessary.

In any case, careful design of these interfaces will be necessary for interoperability to be effective.

[4] See the blogpost https://matrix.org/blog/2022/03/29/how-do-you-implement-interoperability-in-a-dma-world/.

4. TECHNICAL CHALLENGES AND TRADE-OFFS IN IMPLEMENTING HORIZONTAL INTEROPERABILITY FOR NI-ICS

Next to implementation challenges with respect to consent management and re-designing the interface of NI-ICS so that interoperability becomes seamless for users, there are also a number of technical challenges and trade-offs that need to be considered and resolved in order to make interoperable end-to-end-encrypted messenger applications a reality.

Indeed, in mandating interoperability between existing NI-ICS, the DMA poses new technical questions for which solutions are not available off-the-shelf. The main reason is that the DMA requires gatekeepers to make messaging apps interoperable that have not been designed with interoperability in mind. While there may exist protocols for a federated, interoperable messaging infrastructure, such as the Matrix protocol,[5] the use of such a protocol requires that every gatekeeper (and every competitor seeking interoperability) updates its current implementation and adopts the standardised federated protocol from here on for off-net communication. Moreover, the Matrix protocol has its own set of security concerns (see, e.g., Albrecht et al., 2023) For on-net communication the proprietary protocol could be maintained (see also Section 4). Federation of messengers works in similar ways as eMail. Users can choose one of many service providers that run independent servers, but all providers need to implement the same federated protocol (e.g., Matrix in case of messengers, or SMTP in case of eMail), so that the different servers can exchange messages. Federation is also the preferred solution to interoperability by many technologists, such as the newly founded workgroup on More Instant Messenger Interoperability (MIMI) by the Internet Engineering Task Force (IETF), the standard setting body for Internet protocols. However, federation and adoption of a common protocol goes well beyond what is demanded by the DMA, which only requires in Article 7(1) that a gatekeeper "shall make the basic functionalities of its number-independent interpersonal communications services interoperable with the number-independent interpersonal communications services of another provider, [] **by providing the necessary technical interfaces or similar solutions that facilitate interoperability."**

The challenge of making secure messenger apps interoperable *ex-post* into existing systems is thus very different from designing a federated secure infrastructure of interoperable messengers *ex-ante* (from scratch). Due to the unique challenge posed by the DMA to open up closed messengers ex-post, technologists have just begun to think about possible solutions and – from a technical perspective – there is a lively debate and no silver bullet solution that would necessarily win the race. In addition, Article 7(4) makes clear that it is upon the gatekeepers and not the Commission or third parties (e.g., firms wishing to request interoperability) to propose a technical solution ("reference offer laying down the technical details and conditions of interoperability"), and gatekeepers may not have the same incentives as the Commission or an independent third party when it comes to implementation options.

[5] https://en.wikipedia.org/wiki/Matrix_(protocol)

As with every technological design, there are many small and large trade-offs that need to be navigated when designing new protocols or interfaces, and non-technical trade-offs (such as governance or transparency issues) can be factored into the design. In effect, this can make a specific interoperability implementation more or less attractive to competitors. Moreover, the effectiveness of the specific interoperability implementation may not even depend on the gatekeeper alone, but also on which competitors precisely seek interoperability and which pre-existing technical designs and business models they pursue. For example, designs that minimise the exchange of metadata (or more generally designs that lean towards user privacy) are likely to lead to less user convenience and are less preferred by firms that seek to run an advertising-based business model. Against this backdrop, **there is no 'gold standard' against which the Commission may judge the gatekeepers' implementation proposals**.

Nevertheless, it is useful to briefly discuss the main technical trade-offs that should be considered when evaluating specific implementation proposals. Len et al., (2023) see three main areas that need to be considered (and agreed on) when designing interoperability ex-post:

1. **Identity interoperability**, i.e., how users can be discovered on other networks;
2. **Protocol interoperability**, i.e., how a secure channel can be established for cross-network communication;
3. **Abuse prevention**, i.e., how networks can and are allowed to deal with malicious actors (e.g., spammers).

We describe each in more detail below. We thereby focus on the simplest case where text messages are to be exchanged between two parties (sender and receiver). This is also the first step that is required by the DMA. In subsequent steps, interoperable group chats and voice communication are required. These present additional challenges and the complexity is likely to rise significantly. This is because communication is n:n in group chats (as opposed to 1:1 communication in two-party exchange) and each sender/receiver may reside on a different network, using a different identity service and protocol. In voice communication, the main additional challenge lies mainly in achieving encryption in a synchronous manner and on-the-fly, which presents additional requirements on hardware and software.

4.1. How to resolve identities across providers

In a centralised non-interoperable system, identity management is a relatively straightforward task, as there is only one central authority that grants user identities. The central authority can make sure that the namespace is well qualified and identities are unique. Users need to trust that their provider verifies identities correctly, so that they are really communicating with whoever they think they are communicating. But users only need to trust their provider. This is not necessarily the case in a decentralised, interoperable system, where different entities can grant identities. Here users need to trust all providers, and there is no guarantee that the namespace is unique and well qualified.

Generally, an identity involves at least two parts: **a common identifier** (e.g., a username) and a **public key** (the cryptographic identity). Different NI-ICS use different types of identifiers. Although NI-ICS are "number independent" (which means they do not rely on the public telephone system), they often use mobile telephone numbers as identifiers. However, other NI-ICS use self-selected usernames, email addresses or random numbers as identifiers. Identifiers can or cannot tie to real world identities, which already presents a trade-off between privacy and security that different providers strike differently.

The public key is one part of a public-private key pair, which is needed to establish a secure connection. Simply put, a sender retrieves the public key belonging to a certain receiver and uses that key to encrypt the message. The message can then only be decrypted using the private (secret) key of the recipient (and the public key of the sender). Therefore, the issue of identity interoperability generally involves two subtasks. First, **identity discovery**, i.e., making identifiers established and authorised by one provider known to the other providers). This also involves learning at which other provider the designated target identity resides. Second, **retrieving the public key** belonging to a specific identity, which is then the prerequisite for initialising a secure connection. While each part bears its own challenges (cp. Len et al., 2023, Blessing & Anderson 2023), we focus on the issue of identity discovery here.

Several different implementation options exist to address the identity discovery problem. According to Rescorla (2022a), they can be roughly categorised in those solutions that strive to achieve a globally unique namespace, which ensures that every identifier exists only once globally, and solutions which allow for non-unique identifiers ("unqualified namespace"). Each approach has advantages and disadvantages.

The advantage of an **unqualified namespace** is that each provider in an interoperable system can continue to use whatever identifiers it is currently using (telephone numbers, random numbers, etc.) irrespective of whether the identifiers is globally unique. In reverse, this approach requires some kind of centralised look up service, which delivers all matches to a given identity search. Users would then pick the appropriate identity from a list (e.g., annotated with some metadata such as location or provider of the user for disambiguation). Such a look up service can pose some risks to privacy (Rescorla 2022b, Len et al., 2023) and no readily available (standardised) solution seems to exist today that is suitable for the specific case of messenger interoperability (Rescorla 2022b), albeit some solutions (like SPIN[6]) have been proposed.

A **globally unique namespace** can be achieved either using a hierarchical approach or a centralised approach. The hierarchical approach is commonly used in federated systems, such as Matrix, eMail or the Domain Name System (DNS). It means that the global namespace is divided into different subspaces, controlled by different entities that ensure that their respective namespace is unique. For example, each eMail address is unique and split into two parts like identifier@server.com. The part behind the @ designates the entity that controls the subnamespace. This must be unique. The part before the @ is the identifier that is guaranteed to be unique only in the given subnamespace. The

[6] See https://www.ietf.org/archive/id/draft-rosenberg-dispatch-spin-00.html

same system can be used for interoperable messaging, where each pre-existing (possibly non-unique) identifier is annotated by a unique identifier for the specific provider. In the hierarchical approach, identity recovery is resolved through the respective server of the subnamespace. This can also have the advantage that no central server exists which has control over all identities, which bears advantages from a privacy point of view and is also more robust to certain types of attacks (e.g., denial of service attacks) than a centralised system. The disadvantage, however, is that identities are provided by several different entities, which need to be trusted. Furthermore, a unique namespace can also be established by relying on another unique namespace, such as mobile telephone numbers. While this may ensure that a number belongs to a certain person, the same mobile number could be registered with several NI-ICS providers. Thus, one would still have to discover at which providers the number is registered, i.e., another type of look up service is required. Finally, one could also establish a centralised database, where each provider is required to register its identities, and which makes sure that the identifiers are unique. This would also require, however, that already existing non-unique identifiers of various providers would have to be resolved somehow, i.e., some users may not be able to keep their existing identifier. It also bears the question who should operate the central database, which would be crucial for the functioning of interoperability across various providers.

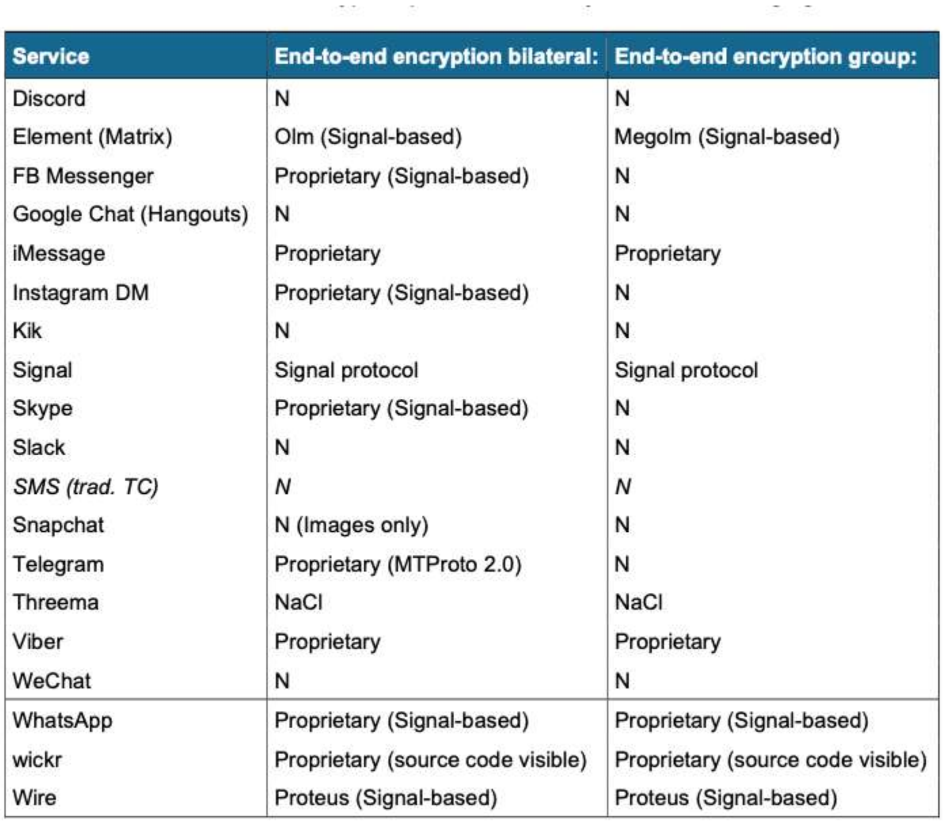
The previous derivations already highlight that the problem of identity discovery is non-trivial when interoperability is imposed ex-post. Most importantly, however, the preceding discussion highlights that **no matter which approach is chosen, some standardisation/agreement between providers is required**. Moreover, either **each provider must be trusted** that it has appropriately verified the identity of the user using some external identity (e.g., by sending a SMS to verify a phone number) or all providers need to trust a central authority to do so. End-to-end encryption is essentially meaningless if the end points of the communication (the identities) are not sufficiently validated (Blessing & Anderson 2023).

4.2. How to establish secure connections

Today many prominent messaging apps employ some kind of end-to-end encryption (E2EE) at least for basic text messages. This means that the communication is secured (to various degrees) between two trusted end points of the communication. End points are typically the hand held devices of participants. Thus, it is important to note that “security” relates only to the communications channel and parties need to trust that the end points are secure and not compromised. Actors having control over the end point (e.g., the operating system, the messaging app itself, or if third parties can access to the smartphone) could theoretically eavesdrop on the communication or establish backdoors without compromising E2EE as such.

Different messaging services typically use **different (incompatible) protocols for E2EE**. Figure 5 shows an overview of the different protocols used for popular messengers presented in Wiewiorra et al (2022). The figure is already a bit dated by now, as some of the messengers have since added support for E2EE (in group chats), or changed the protocol that they use for encryption. However, the main message of this figure is that the protocols implemented in popular messaging apps are diverse and subject to constant evolution. In a more detailed analysis, Rösner & Schwenk (2023) conclude that difference between protocols are “so manifold and diverse that an attempt to provide interoperable messaging by converging the current protocols is pointless.” Albeit several messengers use Signal’s Double Ratchet protocol, and some use derivations of that protocol, the implementations are not directly compatible or interoperable (Blessing & Anderson 2023). It is also important to note that different E2EE protocols imply different security levels (cp. Rösner & Schwenk 2023). In this context, it is noteworthy that some use open source protocols whereas others (including WhatsApp and iMessage) use proprietary protocols. Open source protocols can be verified by independent third parties, as the source code is open; on the contrary this is not the case for proprietary protocols. Thus, the actual level of security is often not known publicly. Further, there is much more to security than just the naked E2EE protocol. For example, some providers may use forward secrecy (a feature that creates temporary keys in order to protect past communication in case an end point has been compromised) whereas others do not. Some providers may rotate the keys more frequently than others, verify user identities more stringently, and so on. Generally, the more secure E2EE is, the more difficult is it to preserve the same level of security when more providers are involved in the communication (Blessing & Anderson 2023).

Also note that **E2EE of group chats is considerably more complex**, and thus less commonly employed in many messengers. Group chats between n parties are often emulated by sending n-1 bilateral messages, which creates significant message overhead. In addition, there need to be protocols to securely and efficiently add and remove group members, which remains an active area of research (Len et al., 2023). A promising candidate E2EE encryption protocol for secure group chat messaging is Messaging Layer Security (MLS), which has in March 2023 been approved by the Internet Engineering



Service	End-to-end encryption bilateral:	End-to-end encryption group:
Discord	N	N
Element (Matrix)	Olm (Signal-based)	Megolm (Signal-based)
FB Messenger	Proprietary (Signal-based)	N
Google Chat (Hangouts)	N	N
iMessage	Proprietary	Proprietary
Instagram DM	Proprietary (Signal-based)	N
Kik	N	N
Signal	Signal protocol	Signal protocol
Skype	Proprietary (Signal-based)	N
Slack	N	N
SMS (trad. TC)	*N*	*N*
Snapchat	N (Images only)	N
Telegram	Proprietary (MTProto 2.0)	N
Threema	NaCl	NaCl
Viber	Proprietary	Proprietary
WeChat	N	N
WhatsApp	Proprietary (Signal-based)	Proprietary (Signal-based)
wickr	Proprietary (source code visible)	Proprietary (source code visible)
Wire	Proteus (Signal-based)	Proteus (Signal-based)

Figure 5: Different end-to-end encryption standards used in different popular messengers according to Wiewiorra et al., (2022). "N" denotes that messages are not end-to-end encrypted. Figure is meant to reflect mere the diversity of protocols used at a given point in time. Figure reflects the state in 2021 and does not provide a complete overview over all messaging services. Changes and updates have occurred since then, reflecting the fast technological progress in the messaging space.

Taskforce (IETF) as a new standard. The standard is backed by some major messaging app providers (e.g., Wire and Google).

In any case, given the myriad of different and incompatible E2EE standards, in order to achieve interoperability there are only two options:

1) The sending provider and the receiving provider would need to agree on a **common encryption protocol** (e.g., one party adopting the protocol of the other). This may also mean that all providers implement all protocols of the other providers, and use whichever protocol is necessary in a given communications scenario.
2) Either the sender or the receiver, or both need to do support multiple protocols and there is some **translation from one protocol** to the other when sending or receiving messages across different protocols.

Interestingly, in Article 7(3) the DMA explicitly demands that E2EE is preserved by interoperability. This seems to rule out scenarios in which there is a server-side translation (so-called "**server-side bridge**"). In this case, the end point of the secure communication would be a central server (the bridge), which decrypts the message coming from the sender, using the sender's encryption protocol, and decrypts the message again using the receiver's encryption protocol. However, this implies that the message is intermittently not encrypted – which breaks the notion of E2EE. If the translation is done at the end point (e.g., a user's smartphone), however, then this would not break E2EE, as translation is done at the end-point (rather than an intermittent server). Therefore, such a **client-side bridge** is an option that has gained some attraction, as E2EE can be preserved, and each provider could largely keep their existing E2EE implementations (Blessing & Anderson 2023). The major flip side of this approach is that with each new provider joining the circle of interoperable providers, all other providers have to update their clients and implement that providers protocol as well. The implementation cost and complexity of this may be insurmountable especially for small provider – who are the intended beneficiaries of interoperability. Moreover, there is additional burden on the end user device, and this makes the end user software more complex (and likely more vulnerable to attacks).

This also relates also to another major design decision when implementing protocol interoperability ex post, i.e., whether a client-to-server framework or server-to-server framework is adopted (see Figure 6).

In a **client-to-server framework** the gatekeeper's server (say receiving a message) allows alternative clients (say sending a message) to connect, possibly in similar ways as the gatekeeper's native clients would connect to the server. Thus, only one server is involved in the end-to-end-communication. Since the receiver's/gatekeeper's provider and the sender's/competitor's provider very likely use different protocols for end-to-end-encryption, the competitor's client would need to implement different protocols for communicating with its own server and that of the competitor and – depending on which server it communicates with – use the appropriate protocol. This approach requires major updates in the client apps of non-gatekeepers and tends to make end user apps more complex, i.e., more error prone and larger in size. This approach seems to be the less favored approach currently by independent experts, such as the IETF MIMI working group.

In a **server-to-server framework**, the gatekeeper's server and the competitor's service exchange the messages directly, and the competitor's clients just communicate with the competitor's server. This means the competitor's app does not have to undergo significant changes. In the server-to-server framework both servers are in involved in the end-to-end-communication. From a top-down

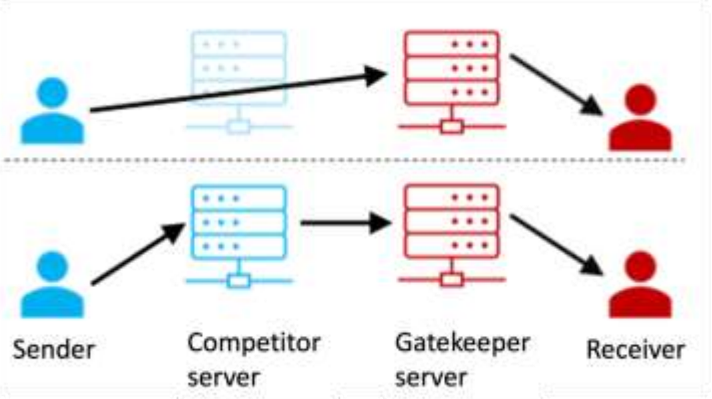


Figure: Interoperability can be implemented in a client-to-server framework (top) or a server-to-server framework (bottom). Figure adapted from Len et al., 2023 perspective, the server-to-server framework seems to be the more attractive choice (Len et al., 2023), as implementation costs are lower for competitors. There may even be some additional advantages with respect to privacy, as each server could act as a privacy relay with respect to its connected users, so that no single server has full knowledge over the social graph (which is an inevitable outcome if there is only one server involved). However, from the perspective of gatekeepers, who are the ones making a proposal on how to implement interoperability, the client-to-server framework could be more attractive (possibly for the same reasons).

4.3. How to take action against malicious users

Abuse prevention is arguably a significant area that contributes to the "**integrity, security and privacy**" of a NI-ICS, and as such it is relevant in the context of Article 7(9), which allows gatekeepers to take strictly necessary and proportionate measures to protect these very values.

Abuse prevention is already a challenge for centralised messaging services. This challenge is significantly amplified in an interoperable system, because there is no central authority that can enforce abuse across all users. A common technique to prevent abuse is content moderation, either through algorithms or through humans, or a combination of both. However, when the message content is end-to-end encrypted – as is common among popular NI-ICS – classic content moderation is not an option. The remaining options are to rely on user reporting (so abusers can be blocked centrally), blacklisting (allowing users individually to block certain users) or to use metadata (e.g., who has communicated with whom, how many messages have been sent, timestamps, length of messages) to detect abusive behavior. All of these measures are typically employed by popular NI-ICS, and in light of Article 7(9) the question arises how they could be preserved in the context of interoperability.

User reporting is considered an effective method for countering abuse at scale (Blessing & Anderson 2023) and is therefore a very popular method employed (Len et al., 2023). When users report abuse, their client typically gathers information about the reported user, such as the user's identity, the reported message, but also a number of previous messages to provide context, and sends it to the central authority. This is also the approach taken in WhatsApp and iMessage (Blessing & Anderson 2023, Len et al., 2023). Providers also employ *message franking* in order to prevent users from providing false abuse reports. Such message franking would also need to be made interoperable, which presents a challenge (Blessing & Anderson 2023).

While user reporting relies on a review by the provider, **blacklisting** is an immediate action that a user can take in order not to receive further messages from an abusive user. It as well is frequently employed in E2EE messengers. However, interoperability requires that users would also be able to blacklist users of other providers, which presents another challenge. In particular, an abusive user may have accounts with several other providers, and in order to be effective across networks, blacklisting would need to be propagated to all those providers at which the abusive user has an account. Also blocked users (either by user reporting or blacklisting) may just make new accounts, either with the same or with new (interoperable) providers. The costs of getting a new identity at a pre-existing provider can be relatively high (e.g., requiring a new telephone number) depending on whether or not the identity is attached to some external identity that is verified by the provider (see Section 3.1).

Instead, obtaining a new identity at a new provider can be of substantially lower costs, e.g., because the new provider does not require an external identity such as a telephone number. This means, with interoperability also those providers at which the abusive users does not yet have an account may need to be notified in order to prevent these so-called sybil attacks.

Both blacklisting and user reporting are retroactive measures. By contrast **abuse detection based on metadata** (e.g., spam filtering) is a proactive measure, which also is frequently employed by messaging providers. Albeit the message content itself is not accessible due to E2EE, metadata is typically not encrypted (and to some extent cannot be encrypted). Metadata is collected and stored to various degrees by different providers, and can involve a user's social graph (which identities have communicated with each other, contact lists, etc.), location data, time stamps of (encrypted) messages sent, filetypes sent, and so on. More privacy affine messengers tend to collect less metadata. But there is a trade-off, as more metadata (e.g., the frequency of messages sent by a given user) also helps to detect abuse. In a centralised system, metadata is collected by one provider, which also facilitates abuse detection. In a decentralised, interoperable system, no single provider likely has all metadata. In fact, (sever-to-server) interoperability can be implemented such that it acts as a privacy relay and prevents the spreading of metadata to other providers (Len et al., 2023). However, this also makes abuse detection and prevention more difficult. Even worse, this may even lead to a rise in abusive behavior, as the moral hazard increases due to the lower detection probability. Further, similar as in the case of blacklisting, effective spam filtering probably requires a shared perception over some metadata, e.g., to impose inter-provider rate limits on forwarded messages in order to prevent the spreading of viral message (Blessing & Anderson 2023).

It is also worth mentioning that the trade-off between privacy and detection probability, as well as the implementation of spam filtering is likely to differ between text messages and voice calls. While text messages could be delayed (for the provider to review, e.g., with respect to metadata), or put in a separate folder (for the user to review), this is not an option for voice calls. This in turn may have an impact on the privacy-detection balance that providers need to navigate, and that they need to find some common ground on when systems are interoperable.

5. TRADE-OFFS

Against the backdrop of the preceding section, we now highlight some of the main trade-offs involved when implementing Article 7 of the DMA more explicitly, and also point to open questions that need to be considered.

We focus on one-to-one messaging, as the interoperability obligation will initially apply only to this context, and only later on extend to group messaging.

5.1. Interoperability implementation trade-offs: APIs vs. standardisation

One main trade-off is between standardisation (which requires all interoperable messengers to implement the same standardised cryptographic API for interoperability) and the use of proprietary APIs (where interoperability is established through the implementation of the various cryptographic APIs of the other providers). In the latter case, we can differentiate between a gatekeeper-side API approach (where the gatekeeper implements the APIs of the competitors seeking access) or the competitor-side API approach (where the competitors implement the APIs of the gatekeeper or gatekeepers).

Rösler and Schwenk (2023) go through the pros and cons of these approaches in some depth and come to the conclusion that only the competitor-side API is realistic. A **standardisation approach** would require all firms to agree on a common standard, which takes time and involves uncertainty. All firms would then have to implement the new standard, which could possibly lower the security standard for some, and increase the security standard for others. For example, some protocols implement forward secrecy, whereas others do not. Importantly, firms could yet keep their proprietary protocols for on-net communication. The standardised protocol would only be needed for off-net communication. Nevertheless, standardisation is a lengthy process, and it is unrealistic that it can be completed in due time. Furthermore, gatekeepers may not have an incentive to conclude that process successfully. However, in the long run, especially if there is more than one gatekeeper[7] to be designated under Article 7, standardisation is arguably the best option from a technological point of view, as it does not create a patchwork of APIs like the other approaches.[8] With a standardised API, all firms would just need to implement one API. With the other approaches, one API per gatekeeper (or competitor) needs to be implemented.

We note that **Article 48** (see also Recital 96) enables the Commission to task a **European standardisation body** to develop an appropriate standard to facilitate interoperability. As we have pointed out above, there are at least three areas in which standardisation may be necessary (discoverability, secure messaging and abuse prevention). On the other hand, the DMA is clear in Article 7(4) that the gatekeeper has to provide a technical reference offer first. By Recital 64, the Commission can consult BEREC (which is not a standardisation body, however) whether the reference

[7] At the time of writing, WhatsApp and Facebook Messenger are designated as core platform services. The designation of iMessage is still under investigation.

[8] Also, the patchwork of APIs seems unworkable for group messaging involving more than two providers.

offer is compliant. It is unclear, however, at which point exactly the Commission can demand a standardisation process according to Article 48.

Gatekeeper-side API would require the competitors to expose an API, which would be implemented by the gatekeeper to send and receive message to and from the respective competitor. This would put the implementation effort mainly on the gatekeeper, which probably has the resources for doing so. However, it would probably be the least preferred option by the gatekeeper, as it has to bear the implementation effort and would be dependent on the APIs provided by competitors.

In reverse, the **competitor-side API** approach requires the competitors to implement the gatekeeper's API; and if there are several NI-ICS core platform services, competitors would need to implement several APIs. The implementation burden would be rather on the side of the competitors. Rösner and Schwenk (2023) differentiate between two instances of the competitor-side API approach. In the first, so-called **competitor-implemented** approach, the gatekeeper would need to specify its cryptographic protocol in sufficient detail, so that it can be implemented by competitors. This poses the biggest implementation effort for competitors, but as a positive side-effect it would de-facto open source the hitherto proprietary cryptographic protocol, which allows for an independent assessment of its security level. The cost of implementation of the competitor-implemented approach can be high, however. The alternative is the **gatekeeper-implemented** approach, where the gatekeeper would provide (closed source) programming libraries to the competitors. The competitors would need to implement those libraries in order to encrypt or decrypt messages to and from the gatekeeper. The cost of implementation for competitors are significantly lower in this case.

The **competitor-side gatekeeper-implemented approach seems to be the most obvious choice** from a gatekeeper perspective. However, it also means that the gatekeeper keeps considerable control over the communication process, as competitors are fully reliant on the gatekeeper's library. In this approach, the level of security cannot be verified by competitors. They need to run executable code of the gatekeeper, and thus also need to trust the gatekeeper's library in that it does what it is supposed to do, and not more.

Importantly, the preceding discussion on standardisation primarily deals with protocol interoperability as presented in Section 3.2. Even if a competitor-side gatekeeper-implemented approach is adopted (which involves no standardisation per se), some standardisation would reasonably be needed to address the issue of identity interoperability (i.e., to publish client identities and to distribute cryptographic keys), as discussed in Section 3.1 (see also Rösler & Schwenk 2023). This is particularly the case when there is more than one gatekeeper service. Additional standardisation is likely needed for interoperable abuse prevention (see Section 3.3). The DMA does not formally require any form of standardisation, however.

From this discussion several questions emerge for the implementation of Article 7:

- Given the benefits of standardisation for an effective implementation, how much can the Commission push for a standardisation approach with respect to i) identity interoperability, ii) protocol interoperability, and iii) abuse prevention? At which point can it invoke Article 48?
- In case a standardisation approach is pursued, should the Commission just relegate the standard setting process to a standard setting body or govern the process more closely in order to ensure that interests of gatekeepers and access seekers are well balanced in the standardisation outcome.[9]
- Under Article 7(6) can additional time be granted before gatekeepers need to be compliant in case they opt for a standardisation process, as it is not realistic to complete the standardisation process within 6 months.
- Under which conditions can the standards be changed (by the gatekeeper or competitors)? Can the Commission invoke Article 48 again to change a standard that has been set previously using Article 48?
- Some gatekeepers may be designated later. In case a standard exists by then, can they be bound to use it? Or can they make a non-standard compliant reference offer?

5.2. Implications of interoperability on security vs. privacy trade-offs

A second major trade-off in the design of any NI-ICS, but especially interoperable NI-ICS, relates to the trade-off between privacy and security. The conflict arises, because E2EE is meaningless if the end points of the communication are not verified for authenticity. As we have discussed in Section 3.1, this often involves verification of user identity through external identifiers (such as phone numbers).

Trade-offs between privacy and security also arise in the context of abuse prevention, as discussed in Section 3.3. Abuse prevention is more effective if metadata is shared among providers, possibly even with providers at which a user does not (yet) have an account in order to prevent sybil attacks.

Article 7(8) demands that not only such personal data is shared as is “strictly necessary to provide effective interoperability”? However, **Article 7(3)** demands that the “level of security” shall be preserved across the interoperable services. These two provisions **may likely be at odds**, as different providers establish different levels of security also by collecting different amounts of metadata that facilitate abuse prevention. As discussed in Section 3.3, there is a need to share metadata for effective abuse prevention, which falls under the umbrella of a system’s “security”.

Len et al., (2023) propose that the sender’s provider should be responsible for abuse detection based on metadata and filter out messages before they are relayed to another provider. This, however, means that providers would have to rely on an external (competing) provider for abuse detection (Blessing & Anderson 2023), which is probably not acceptable for many providers, and also gives rise

[9] Political involvement in standard setting processes is not unusual and was, for example, also the case in the development of the GSM standard for mobile communications.

to issues of moral hazard. This also does not resolve the issue that generally less metadata is available (compared to a central system) on which the detection can be based, which likely lowers the detection rate.

More generally, from our discussion in Section 3 it is **difficult to see how interoperability would not affect the level of security or privacy in one way or another**. We acknowledge that there may be some isolated instances and implementations in which privacy or security is indeed improved through interoperability. For example, because in a server-to-server framework, each server can act as a privacy relay (Len et al., 2023). Or because interoperability requires some clients to adopt more secure protocols (Blessing & Anderson 2023). However, in general interoperability requires to increase the circle of trusted parties, requires to share some (meta-)data across several providers and increases protocol complexity. All of this increases the possible threat vectors and tends to lower the overall level of security (Blessing & Anderson 2023), even if at a cryptographic level the level of security is maintained. In this context, some privacy-focused messengers such as Threema and Signal have already announced publicly that they do not want to seek interoperability under the DMA because of security and privacy reasons. In reverse, Articles 7(3), 7(8) and 7(9) may therefore be powerful defenses for gatekeepers objecting interoperability.

From this discussion further implementation questions arise, such as:

- Is a gatekeeper allowed to reject an interoperability request if the competitor's service does not verify a users' identity based on some external identity? Otherwise, the "level of security" may be lessened.
- How will a gatekeeper verify the level of security of a competitor's service (e.g., using a proprietary protocol)? Will they have to take their word for it? Do they have authority to demand critical information? Can they turn to the Commission to verify the level of security and/or to obtain critical information? For example, if the competitor's service is closed-source, will they have a right to obtain the competitor's source code? Under what conditions can they refuse an interoperability request based on protocol security?
- Can gatekeepers deny interoperability with messengers that do not employ appropriate abuse prevention or cooperate in abuse prevention, e.g., by sharing data about the reported user?
- How much metadata would other services need to share with a gatekeeper, and vice versa, to maintain the same "level of security"? Can gatekeepers refuse an interoperability request if not sufficient metadata is shared?
- Under what conditions can a gatekeeper refuse to trust a third party?

5.3. Interoperability vs. usability trade-offs

Interoperability also involves unique trade-offs for usability and the design of user interfaces. First, there is a trade-off between usability and privacy/security that different providers strike differently. For example, a privacy-affine messenger like Threema does not use phone numbers as identifiers, which arguably has downsides for usability. As discussed in Section 3.1, interoperability requires some identity interoperability which can interfere with that trade-off and likely has a negative impact on usability. Similar trade-offs arise with respect to usability and security. Some messengers change cryptographic keys more frequently than others, but as discussed in Section 3.2, protocol interoperability requires to adopt a scheme that is compatible with the gatekeeper.

Interoperability also has implications for the user interface design. This involves the **discovery process of other users on other messengers**: How many other providers are visible to a user? Can a user choose on which other providers he or she wants to be discoverable? How many "search results" does the discovery service provide? It is evident that such design decisions can have significant implications to which interoperability is perceived useful by end users, and thus to which extent interoperability may facilitate market contestability.

As interoperability is only required in the EU, a question also arises **which users are discoverable across messengers. Only users from the EU, or all users?** Especially if discoverability requires changes in the namespace (see Section 3.1), users outside of the EU are likely to be affected by system-wide changes one way or another.

Interoperability may also require to **distinguish between messages coming from alternative providers**. However, dark patterns could be used to discourage interoperability,.

The interface design also needs to account for the more complex **consent management**, as users can opt out of interoperability by Article 7(7). As the list of interoperable competitors grows, this can have significant implications on usability, especially in the context of group chats. Here, likewise dark patterns may be employed.

5.4. Interoperability vs. innovation

Interoperability can also affect innovation efforts. In Bourreau et al., (2022) we discuss this complex issue in detail and more nuanced, whereas we can only provide a synopsis of the main arguments here: Some have argued that interoperability can spur innovation (Scott-Morton et al., 2021), because interoperability is limited to basic functionalities. Post interoperability of basic functionalities, providers seek to differentiate themselves through new non-interoperable features to attract consumers. However, if this is the case, and consumers indeed see value in those new features, it also undermines the value of interoperability, as important (future) features are not interoperable.

In reverse, interoperability can also undermine innovation efforts when such features are meant to be interoperable. Blessing and Anderson (2023) provide the example of self-exploding messages that are automatically deleted after some time period. If such a feature were to be made interoperable, then first, different providers need to agree on a common form (e.g., acceptable time limits) for those new features, which slows down the innovation process. Second, providers need to rely on and trust other providers that messages are indeed deleted as specified. Users also need to trust that this is indeed the case across providers in order to be able to value this feature.

In case standardised APIs are used to establish interoperability (and to some extent also in the case of a gatekeeper-side competitor-implemented approach), it becomes more difficult to change the standard, as it involves a collective action by all parties involved. This as well can stifle innovation. To be fair, technical standards can allow for some degree of extensibility (such as in the case of XMPP –

the Extensible Messaging and Presence Protocol), which alleviates some of these concerns. However, the argument remains that innovation is more constrained, as it still needs to respect the limits of extensibility and possibly needs to maintain backward compatibility.

In reverse, as Figure 5 shows, competition between messengers (not adhering to a common standard) has led to several innovations and implementations in the cryptographic protocols. Different messengers strike the balance between usability, privacy and security differently. Importantly, as pointed out in Section 4.1, a standardised API may only be necessary for off-net communication, and providers could maintain differentiated protocols for on-net communication. Thus, providers would still be free to innovate with respect to their proprietary protocols (Rösner & Schwenk 2023). However, this as well would over time decrease the benefit of (off-net) interoperability, as the standardised API becomes frozen in time and does not keep up with the innovations that occur for on-net communication. The different innovation trajectory between on-net and off-net communication is amplified by the staggered implementation process of the Article 7 provisions, whereby interoperable functionalities only have to be implemented step-by-step over time. As a consequence, users are likely to perceive on-net communication superior to off-net communication, undermining the value of interoperability.

5.5. Interoperability vs. multihoming

A final trade-off that we want to discuss here involves messengers that facilitate multihoming, as an alternative to messengers that are interoperable. As Len et al., (2023) point out, there already exist a few so-called multi-messengers that integrate several popular messaging apps (like WhatsApp and iMessage) under one combined user interface. These messengers include Beeper, Texts, and Mio. Little seems to be known about their implementation and level of security, but all seem to require that users have proper accounts on all messaging services that they want to communicate with.[10] Further, these messengers seem to rely on client-side bridging (cp. Section 3.2).

As we have pointed out in a previous CERRE report (Bourreau, Krämer & Buiten 2022), interoperability provides a partial substitute to multihoming. A user on a gatekeeper messenger that can communicate (even though only with basic features) with a user on an alternative provider does not need to make a proper user account with the alternative provider anymore. The user has less reasons to try out the alternative provider firsthand, and user experiences with that alternative provider are always mediated through the limited interoperable functionalities. In other words, interoperability lowers multihoming incentives, but multihoming can likewise be a powerful driver for market contestability. Users are thus also less inclined to use multi-messengers. To be clear, the DMA does not require any NI-ICS to take up an interoperability offer. So NI-ICS have an option to rather build on multihoming, or to build on interoperability. However, not all may be fully aware of the trade-offs involved.

[10] Some information about Mio's implementation and security-related aspects can be found in their white paper: https://go.m.io/security-white-paper

6. VERTICAL INTEROPERABILITY IN THE DMA

In this section, we build on our previous reports (Bourreau, Krämer & Buiten 2022, and Bourreau 2022) to discuss **overarching principles** that apply to the **main vertical interoperability obligations** introduced in the DMA:

- Sideloading of applications and app stores (Article 6(4));
- Access to essential hardware and software features of the operating system (Article 6(7)).

We refer the reader to these reports for a more extensive analysis.

It is inherent to vertical interoperability that the gatekeeper controls a bottleneck resource (e.g., the operating system) to which access is being provided. This is not necessarily the case for horizontal interoperability between NI-ICS, albeit – as we have shown above – this can be the outcome of specific implementations (such as the gatekeeper-side API approach).

Whereas the provision on horizontal interoperability is limited to the very specific case of NI-ICS, the provisions on vertical interoperability are potentially open ended and span over a much broader application scope, ranging from alternative app stores and applications to access to the NFC chip in order to enable alternative payment services. Nevertheless, **five overarching principles for implementation** can be highlighted (cf. also Bourreau, Krämer & Buiten 2022, and Bourreau 2022).

6.1. Screening of access requests

Article 6(7) DMA states that gatekeepers must provide “effective interoperability with (…) the same hardware and software features accessed or controlled via the operating system or virtual assistant (…) as are available to services or hardware provided by the gatekeeper.”

Therefore, access to essential hardware and software features is mandated if the gatekeeper uses them for its own products or services, i.e., if it is vertically integrated.

In a previous report (Bourreau et al., 2022), we argued that vertical integration is a necessary but not a sufficient condition for mandating vertical interoperability. Indeed, it is well known that vertical integration also brings several efficiency benefits, such as the avoidance of double marginalisation and hold-up problems. Therefore, mandated vertical interoperability requires a clear theory of harm and justification.

The three-criteria test used in telecommunications regulation could be a possible approach, limiting mandated vertical interoperability to situations where i) there are high and non-transitory barriers to entry, ii) there is no trend towards effective competition, and iii) where competition law is considered insufficient. In particular, it should be examined whether the hardware and software features are indeed "essential", i.e., whether they cannot be replicated by third parties, at least at a reasonable cost.

6.2. Screening of access seekers

Both, Articles 6(4) and 6(7) allow the gatekeeper to take strictly necessary and proportionate measures to protect the integrity and security of the gatekeeper's hardware and software systems. This can provide justification to limit access only to those access seekers that meet certain security or integrity standards. Note also that the screening of access seekers may be a substitute for notifying users of security or integrity risks - see our discussion of this trade-off below.

In addition, "free of charge" access may not send the right signal to access seekers, leading to (excessive) entry of possibly inefficient players. Therefore, the fact that access should be provided "free of charge" makes screening of access seekers particularly important.

One possible approach would be to allow the gatekeeper to grant **access licenses** based on public, explicit and non-discriminatory criteria. Under this access licensing approach, if the access is denied, the access seeker could appeal to the regulator. For access requests under Article 6(4), a fruitful starting point for a catalogue of security and integrity criteria is the "Code of practice for app store operators and app developers" developed by the UK Department for Science, Innovation and Technology.[11]

Another approach would be to confer the administration of the access regime to the regulator or an independent third party. For reasons of timeliness and pragmatism (the gatekeepers know their hardware and software and the associated risks best), we believe it makes sense to start with a gatekeeper-led approach in the beginning, and only turn to other solutions if that fails to achieve the desired goals.

Similarly, the gatekeeper should have the ability **to revoke access licenses**, again based on public, explicit and non-discriminatory criteria, for instance, if the access seeker does not comply *ex-post* with the requested security and integrity standards.

It is also worth pointing out that access conditions (based on security and integrity considerations) are likely to vary significantly depending on the specific functionality that is to be made interoperable. This also means that the access conditions are likely to be different for those cases falling under Article 6(4) and those under Article 6(7).

Specifically, under Article 6(4), if alternative app stores are granted an access license, then these stores should also be responsible for screening the apps that they host. The screening process should comply to the responsibilities conferred under the license, but otherwise be independent of the gatekeeper's screening process.

[11] See https://www.gov.uk/government/publications/code-of-practice-for-app-store-operators-and-app-developers

6.3. Gatekeeper-led definition of interfaces

Effective interoperability, or access to the hardware and software functions controlled by the gatekeeper, requires the definition of relevant hardware and software interfaces. An important question is who should define the interfaces?

We believe that the most appropriate approach is to allow the **gatekeeper to design and manage the interfaces**. From a technical point of view, the gatekeeper is in a better position to design the interface because it has developed the hardware or software technology. In addition, the platform can easily update the interface when technical changes are needed and can also take the necessary measures to ensure integrity and security.

However, there is a potential risk that the gatekeeper may use its dominant position to degrade the quality of the interfaces offered to some third parties. Access to these interfaces must therefore be **non-discriminatory**.

In the event of complaints and concerns about possible non-compliance, the regulator would investigate the technical specifications of the access interface.

An alternative approach would be to develop an open interface standard. The success of the Internet is largely attributed to its versatile open vertical interoperability standards (cp. the Open Systems Interconnection (OSI) model[12] for more details). However, the standardisation of interfaces can take a long time and it can be complex to reach consensus among market participants with different (and sometimes conflicting) incentives.

Note, however, that these two approaches are not necessarily exclusive. Interfaces based on proprietary interfaces could be developed in the short term, while a standardisation process could be initiated with the goal of developing open interfaces in the long term. Further, vertical access provisions under the DMA relate to proprietary platform services, for which it may not always be feasible to provide access through standardised interfaces.

6.4. Equivalence of input

The general guiding principle for access to a particular hardware or software function should be the ‘equivalence of input’; that is, an entrant should have access to the **same function, and on the same terms, as the vertically integrated gatekeeper for its own complementary products and services**.

Note, however, that “equivalence” does not mean “equality”. Access to the hardware or software function may be provided through a specific API that is different from the internal API used by the gatekeeper, as long as the two APIs are “equivalent” in terms of functionality.

The ‘equivalence of input’ principle requires monitoring to verify compliance by the access provider, which can be complex and time-consuming. One possibility would be to have a first level of monitoring, where access providers would submit their process in their annual compliance reports. In

[12] See https://en.wikipedia.org/wiki/OSI_model for more details.

case of complaints from access seekers, more stringent forms of monitoring (e.g., through audits) could be introduced.

The gatekeeper could also gradually provide information on the software and hardware features that are accessible to third parties for access, with details of any restrictions for using them.

An alternative to the 'equivalence of input' principle is an 'equivalence of output' principle. However, we strongly believe that whenever possible, 'equivalence of input' is to be preferred, as an access-seekers 'output' depends on various factors, many of which are not under the control of the access provider.

6.5. Non discrimination in choice architecture

Since vertical interoperability implies that the gatekeeper is forced to open up a bottleneck resource (e.g., operating system) in order to enable alternative downstream providers (e.g., apps), the choice architecture for users for selecting alternative providers will be critical. Dark patterns in choice screens or self-preferencing would limit the ability for users to take advantage of the new alternatives and could therefore constitute a violation of the anti-circumvention clause in Article 13(6) DMA.

Therefore, open questions include what are acceptable choice architectures in the context of alternative distribution channels and what restrictions are absolutely necessary and proportionate for security reasons. The DMA provides some clarifications in Recitals 50-54. However, this remains a complex issue in its own, and it is dealt with in the companion issue paper by Fletcher (2023).

Article 6(4) already provides explicit guidance on the choice architecture in demanding that the "gatekeeper shall, where applicable, not prevent the downloaded third-party software applications or software application stores from prompting end users to decide whether they want to set that downloaded software application or software application store as their default. The gatekeeper shall technically enable end users who decide to set that downloaded software application or software application store as their default to carry out that change easily." The anti-circumvention clause in Article 13(6) DMA implies that the choices offered to end user should not be presented in a non-neutral manner.

In all cases it should be as easy for the consumers to install an alternative provider as it is for them to install the gatekeeper application – without prejudice to the possibility to pre-install applications according to Recital 53 of the DMA. This can also be viewed and rationalised under the lens of **equivalence of input (our fourth principle)**. Further, a neutral choice architecture also means that it is equally easy to change between alternative providers, as well as to change back to the gatekeeper.

It may also involve prompting the user to reconsider their choices in reasonable intervals (see Fletcher (2023) in relation to Article 6(4).

Finally, we wish to point out that there may be interactions between the five principles that should be scrutinised by the Commission under the lens of proportionality. For example, a gatekeeper may justify and employ a strict licensing regime, where it applies a certain security and integrity standard (yet, necessary and proportionate) when screening alternative providers before granting an access license. But in this case – in line with Recital 50 of the DMA – it does not seem “strictly necessary and proportionate” that the gatekeeper additionally presents warning messages to users whenever they seek to engage with one of the pre-vetted alternative providers. In reverse, when the gatekeeper pursues a very lenient access regime, or does no pre-vetting at all, then a warning message to users seems to be proportionate.

REFERENCES

Albrecht, M. R., Celi, S., Dowling, B., & Jones, D. (2023). Practically-exploitable cryptographic vulnerabilities in Matrix. Cryptology ePrint Archive, Paper 2023/485 Available at: https://eprint.iacr.org/2023/485.pdf

Blessing, J., & Anderson, R. (2023). One Protocol to Rule Them All? On Securing Interoperable Messaging. arXiv preprint arXiv:2303.14178. Available at https://arxiv.org/abs/2303.14178https://arxiv.org/abs/2303.14178

Bourreau, M. (2022). DMA Horizontal and Vertical Interoperability Obligations. Centre on Regulation in Europe (CERRE). Issue Paper. 11/2022. Available at: https://cerre.eu/wp-content/uploads/2022/11/DMA_HorizontalandVerticalInteroperability.pdfhttps://cerre.eu/wp-content/uploads/2022/11/DMA_HorizontalandVerticalInteroperability.pdf

Bourreau, M., Krämer, J. & Buiten, M. (2022). Interoperability in Digital Markets. Centre on Regulation in Europe (CERRE) Policy Report, 03/2022. Available at https://cerre.eu/wp-content/uploads/2022/03/220321_CERRE_Report_Interoperability-in-Digital-Markets_FINAL.pdfhttps://cerre.eu/wp-content/uploads/2022/03/220321_CERRE_Report_Interoperability-in-Digital-Markets_FINAL.pdf

Fletcher, A. (2023). Choice Architecture for end users in the DMA. Centre on Regulation in Europe (CERRE) Issue Paper. 09/2023.

Len, J., Ghosh, E., Grubbs, P., & Rösler, P. (2023). Interoperability in End-to-End Encrypted Messaging. Cryptology ePrint Archive, Paper 2023/386. Available at https://eprint.iacr.org/2023/386

Rescorla, E. (2022a). End-to-End Encryption and Messaging Interoperability. Educated Gueswork Blog Post. Available at: https://educatedguesswork.org/posts/messaging-e2e/#identity.

Rescorla, E. (2022b). Discovery Mechanisms for Messaging and Calling Interoperability. Educated Gueswork Blog Post. Available at: https://educatedguesswork.org/posts/messaging-discovery/

Rösler, P., & Schwenk, J. (2023). Interoperability between Messaging Services Secure Implementation of Encryption. Study for the German Federal Network Agency. Available at https://www.roeslpa.de/files/230503_dmaSecureReport.pdfhttps://www.roeslpa.de/files/230503_dmaSecureReport.pdf

Scott Morton, F. M., Crawford, G. S., Crémer, J., Dinielli, D., Fletcher, A., Heidhues, P., & Seim, K. (2021). Equitable Interoperability: the "Super Tool" of Digital Platform Governance. Policy Discussion Paper No. 4, Digital Regulation Project, Yale Tobin Center for Economic Policy. *Available at SSRN* 3923602.

Wiewiorra, L., Steffen, N., Thoste, P., Fourberg, N., Tas, S., Kroon, P., Busch, C., Krämer, J. (2022). Interoperability Regulations for Digital Services. WIK Consult Report. Study for the German

Federal Network Agency. Available at
https://www.bundesnetzagentur.de/DE/Fachthemen/Digitalisierung/Technologien/Onlinekomm/Study_InteroperabilityregulationsDigiServices.pdf?__blob=publicationFile&v=1

cerre Centre on Regulation in Europe

Avenue Louise 475 (box 10)
1050 Brussels, Belgium
+32 2 230 83 60
info@cerre.eu
www.cerre.eu

Centre on Regulation in Europe (CERRE)
CERRE Think Tank