Comparing the openness of archetypical business-to-government information sharing architectures: balancing advantages of openness with the control of risks

van Engelenburg, Sélinde; Janssen, Marijn; Klievink, Bram; Tan, Yao-hua; Rukanova, Boriana

DOI
10.1145/3209281.3209350

Publication date
2018

Document Version
Final published version

Published in
Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age

Citation (APA)
van Engelenburg, S., Janssen, M., Klievink, B., Tan, Y., & Rukanova, B. (2018). Comparing the openness of archetypical business-to-government information sharing architectures: balancing advantages of openness with the control of risks. In *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age* (pp. 1-10). Article 88 (dg.o '18). Association for Computing Machinery (ACM). https://doi.org/10.1145/3209281.3209350

Comparing the Openness of Archetypical Business-to-Government Information Sharing Architectures

Balancing Advantages of Openness with the Control of Risks

Sélinde van Engelenburg
Delft University of Technology
Delft, the Netherlands
S.H.vanEngelenburg@tudelft.nl

Marijn Janssen
Delft University of Technology
Delft, the Netherlands
M.F.W.H.A.Janssen@tudelft.nl

Bram Klievink
Delft University of Technology
Delft, the Netherlands
A.J.Klievink@tudelft.nl

Yao-Hua Tan
Delft University of Technology
Delft, the Netherlands
Y.Tan@tudelft.nl

Boriana Rukanova
Delft University of Technology
Delft, the Netherlands
B.D.Rukanova@tudelft.nl

ABSTRACT

Business-to-government (B2G) information sharing can benefit government organisations, as well as businesses. Yet, businesses are often reluctant to share, as data sharing might not just provide benefits but also entails risks. Therefore, a system supporting B2G information sharing should provide the appropriate level of openness, such that the advantages of openness and possibilities to control risks for businesses are balanced. At the same time, the information obtained by the government should be useful. We identified three architectural layers at which B2G information sharing architectures can have different levels of openness, viz. the Software Layer, the Access Control Layer and the Data Layer. In this work, we compare three archetypical configurations of architectures for B2G information sharing with different levels of openness. Our aim is to provide insight into their impact on the possibilities for obtaining advantages from information sharing and managing risks of opening up data. We found that the relationship between the different levels of openness and the advantages and risks of information sharing is highly complex. We discuss this complexity and find that different levels of openness are appropriate in different situations.

CCS CONCEPTS

• **Applied computing → E-government**; *IT architectures*; *Supply chain management*; • **Information systems** → *Data exchange*; • **Social and professional topics** → *Systems analysis and design*;

KEYWORDS

Business-to-government information sharing, open information sharing, architecture design, open source, supply chain

ACM Reference Format:
Sélinde van Engelenburg, Marijn Janssen, Bram Klievink, Yao-Hua Tan, and Boriana Rukanova. 2018. Comparing the Openness of Archetypical Business-to-Government Information Sharing Architectures: Balancing Advantages of Openness with the Control of Risks. In *Proceedings of 19th Annual International Conference on Digital Government Research (dg.o'18)*, Anneke Zuiderwijk and Charles C. Hinnant (Eds.). ACM, New York, NY, USA, Article 4, 10 pages.

1 INTRODUCTION

To perform some of their functions, government organisations need information from businesses. Therefore, businesses are required to share at least some information with the government; for example tax or customs declarations. Government organisations could benefit from acquiring additional information that businesses do not have to provide according to legislation to cross-validate the accuracy of these declarations. Yet, government organisations often do not want to increase the administrative burden for businesses [5]. Therefore, such additional information sharing would be on a voluntary basis. Because of this voluntariness, the willingness of businesses should be taken into account when designing an information sharing architecture that supports this business-to-government (B2G) information sharing.

To illustrate, in the container shipping domain Customs monitors the goods flow [1, 10]. The information quality in the obligated documents, e.g. customs declarations for import or export of cargo, is typically low compared to the information that the businesses base their own operations on [10, 17, 19]. Customs organisations have use for higher quality information to perform more effective and efficient risk assessment, which also benefits compliant companies [5, 10]; for example customs could use commercial documents such as invoice or container packing list to cross-validate these declarations. Customs is expected to contribute to the competitiveness of their country and thus do not want to obligate the businesses in the supply chain to share the additional high quality information [5]. Hence, for them to obtain the additional information, the businesses need to be willing to share.

There are several ways in which the willingness of businesses can be increased to share additional information with government organisations. Government organisations can incentivise businesses, e.g. by performing less disruptive inspections on businesses that share the additional information that is needed to justify such a

dg.o '18, May 30-June 1, 2018, Delft, Netherlands

ACM ISBN 978-1-4503-6526-0/18/05…$15.00
https://doi.org/10.1145/3209281.3209350

regime [5]. Another way is by reducing the efforts required for businesses to share the information with government organisations. Additional data processing to share data with a separate information sharing system can be avoided when government organisations piggy-back on the data flow between businesses [23]. To make this piggy-backing possible, an information sharing system should extract the B2G information flow from business-to-business (B2B) information flows; e.g. make an invoice or packing list visible for customs via a special dashboard.

Businesses, as well as government organisations, might benefit from a high level of openness in such an information sharing system. A higher level of openness allows government organisations to more easily obtain the data that is useful to them and to freely use this data to improve their effectiveness and efficiency. Furthermore, it allows re-use of data and supports the generation of new services and value, e.g. to the public [16]. The benefits for businesses include the enabling of new value-added services such as improved logistics planning, efficiency gains, and closer collaboration with other parties [6, 8, 22].

However, a higher level of openness may conflict with the need that businesses have for controlling their data [18]. Businesses may view autonomous control over data and sharing arrangements as key to their competitive position [7, 14, 24, 25]. They want control over the information sharing system, as well as use the information sharing system to control access to the data. Opening up data to others, means that businesses have to give up some control and autonomy. They may fear that they will be more vulnerable to misuse of the data or to opportunism by others, and that sensitive information is not kept confidential [6, 9, 15, 26, 29]. For example, businesses might be willing to share their invoices with customs for expedited customs treatment of their cargo at the border, but want to be sure that this data is not visible to the competitors that might be participants of the same sharing system. This negatively impacts the willingness to share data, especially in an open way.

A designer of a system in which government organisations can piggy-back on the B2B information flow will need to balance the opportunities to gain advantages of openness with possibilities for businesses to control risks, in order to increase the willingness of businesses to share. However, the insight into the impact of different levels of openness is restricted to the rough idea that more openness provides easier access to data, but also means loss of control over data. This makes it hard to establish exactly how various degrees of openness can affect the balance required to support information sharing. Yet, in a multi-stakeholder setting, striking such a balance will be unavoidable.

The aim of this investigation is to get insight into the effect of architectural design choices concerning the openness of information sharing systems. To do so, we compare different configurations of information sharing architectures with different levels of openness. Two of the architectures are extremes, viz. completely open and completely closed . In addition, we consider a configuration with an intermediate level of openness as well. We discuss the configurations and their selection in §2.

Openness of data and risk control can be considered by regarding several factors, such as information accessibility and security. To get a more nuanced view (e.g., something can improve security, but harm scalability), we identify these factors and use them in our comparison of the three theoretical configurations. The comparison framework is discussed in §3.

For the actual comparison (presented in §4), we establish what the configurations look like for architectures in the context of information sharing in international container shipping. We presented the configurations and an analysis of their impact during a workshop involving senior level staff at a large business involved in container shipping. We used the results of the discussion with the members of the workshop to establish additional relevant architectural layers and to further work out the analysis. We then discussed the relevance of the results with domain experts.

Finally, in §5, we provide an overview of the comparison and discuss what level of openness would be appropriate in what situation. This overview could support designers of B2G information sharing architectures in finding the appropriate level of openness for their design. This could help them to balance the advantages of openness with the possibility for risk control that is required for businesses to be willing to share additional information via their architecture.

Information sharing in international container shipping is a domain that currently is an active subject for research (see e.g. [12, 21]), and the insights in this paper are thus relevant for this body of work. Furthermore, this work extents the existing work on the design of B2G information sharing architectures (see e.g., [13, 17, 27, 28]).

2 OPENNESS OF THE ARCHITECTURAL LAYERS AND THE CONFIGURATIONS

As discussed in the introduction, we expect that openness will have an effect on the possibility for businesses and government organisations to obtain advantages from information sharing and on the possibilities for businesses to control risks. A way to get insight into these effects is by comparing information sharing architectures with different levels of openness to each other. However, the possible architectural design choices that are concerned with openness are too numerous to cover. We thus have to select the configurations that will provide us with the best insight.

For the selection of architectural configurations to compare, we first identify three layers at which information sharing architectures can have different levels of openness. These architectural layers are 1) the software layer, 2) the access control layer, and 3) the information layer. The architectural layers are shown in figure 1. For each range of openness on the different layers, there are still numerous possibilities. We presume that architectures that have levels of openness that are close to each other, do not have very pronounced differences in advantages and risk control. Comparing two archetypical architectures that are extremes concerning openness, would in that case provide more information on what the impact of openness is. In addition, we include a configuration that is at the centre of the openness scale as well, as the dynamic between openness and advantages and risk control could be different in the centre than in the extremes. In this configuration, openness is context-based.

Openness pertains to enabling parties to access, modify or share something. To mitigate risks, parties want to control the information sharing system itself. The extent to which they can do so

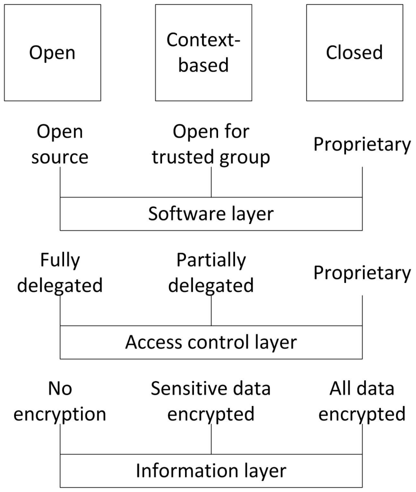

Figure 1: Overview of the architectural layers and different levels of openness on a scale

depends on the extent to which they can access, modify and share the software necessary to use the information sharing system. This thus makes the software layer relevant for our comparison.

For the software layer, on one side of the openness scale a designer could choose to use software that is completely open source, and is freely available to all parties to use, modify or redistribute [2, 11, 30]. On the other end of the openness scale for the software layer, the information sharing system is based on proprietary software. This type of software is developed and owned by a specific party that can sell or license the software to other parties [2]. Other parties cannot modify the code of such software [30].

A context-based level of openness for the software layer is one in which the parties are separated into different groups for which there are different levels of openness of the software. For instance, there could be a group of trusted users that can use and modify the software, while this is not possible for all other parties. Openness then depends on in which group a user is.

Businesses want control over the data itself. The way in which access control is arranged, thus is relevant as well. This is arranged at the access control layer of the architecture of the information sharing system.

When an access control layer of an information sharing system is fully open, access control is fully delegated to other parties. For example, access control could be performed by the party that governs the information sharing system, instead of the party that supplies the data. On the fully closed side of the spectrum, access control is fully proprietary and arranged by parties themselves. For an intermediary level of openness, parties can partially delegate access control to others.

Parties can partially delegate access control by, for instance, delegating the decision making on who should be authorised to access data to another party, while still supplying the business rules on which this decision is based. Another example is when parties themselves decide who is authorised to access data, but they delegate identity management to other parties that they trust.

Table 1: The comparison framework

Category	Comparison factor	Aspects
Obtaining advantages	Information accessibility	Ease of providing access Ease of gaining access
	Scalability	New users New data types
Controlling risks	System reliability	Fault tolerance Risk of failure
	Access control	Level of control Refinedness
	Security	Inference attack risk Unauthorised disclosure risk

The possibilities for actually accessing data, depends on the openness of the information that is shared as well. The last architectural layer that is relevant, thus is the information layer. For the information layer, at one end of the openness scale, all data is shared without encryption. A choice for a completely closed information layer would entail encrypting all the data when it is shared. At an context-based level of openness for in the information layer, parties are allowed to encrypt only parts of the data that are sensitive to them. For instance, they can separate documents that they want to share in data elements and then only encrypt the data elements that are sensitive.

As we cannot compare all possible combinations of different levels of openness for these three layers, we will compare three archetypical configurations. The first one, is the completely open configuration. For this configuration, the software is open source, access control is fully delegated and information shared is not encrypted. In the context-based configuration, the software is only open to a group of users that are trusted. In this configuration access control is partially delegated and only sensitive data is encrypted. For the closed configuration, the software and access control both are proprietary. Furthermore, all data that is shared, is encrypted.

3 THE COMPARISON FRAMEWORK

In this research, we are concerned with balancing the obtaining of advantages from open data sharing and the control of risks. In this section, we discuss some of the factors that make up or impact the obtaining of these advantages and the controlling of risks. These factors are the basis of the comparison framework (see table 1) that we use for comparing the configurations.

We identified the factors based on the empirical findings from the workshop that was conducted for this research. In addition, we took into account empirical findings from the EU-funded CORE project where various implementations of information sharing architectures were developed and tested. In this project, users identified these comparison factors as key issues to be solved.

As discussed in the introduction, both government organisations and businesses can benefit from information sharing. To reap these benefits and to be able to obtain and use data, information accessibility is important. Information accessibility can be perceived from two points of view. The first is from the point of view of the party that wants to obtain access to data. These are the government organisations and the businesses involved in the information sharing.

We can also look at information accessibility from the point of view of the party that wants to provide access to the data. In our case these are the businesses. If it is hard for businesses to provide access, they might not do so. Therefore, it is important to consider data accessibility from their point of view as well.

The accessibility of information shared via the information sharing system is determined by the ease of gaining or providing access to the information. We can view the ease of access as the number of steps that need to be performed by parties to gain or provide access to data. In addition, the effort required to perform those steps is important.

Businesses that are initially not willing to share data via an architecture, might be willing to do so at a later time. When more businesses use the architecture, more information might be accessible to them. This could then cause an influx of more businesses. Furthermore, in time, businesses might want to share new types of information using the information sharing system. However, to allow more businesses to connect and to share additional data, the architecture should be scalable. Scalability is thus another factor related to enabling parties to obtain advantages from a higher level of openness of information sharing. Scalability can be viewed as the ability to change the levels of parameters that capture the performance aspects of a system [20]. The parameters that are interesting for our comparison, are the number of users involved and the different types of data that are shared.

The reliability of the system impacts the possible advantages as well as the risks. If components of the system have a high risk of failing, and this results in the system not being able to perform its functions (i.e., it has a low fault tolerance), then the system cannot be considered reliable. If the system is not reliable, then government organisations as well as businesses might not want their processes to depend on the information sharing via the system. Furthermore, the businesses might not trust that they can actually control risks when using the system.

In the introduction, we discussed some of the risks for businesses associated with opening data. To control these risks, businesses need to be able to determine what parties should and should not get access to their data. The way in which access control is shared by parties, determines the level of access control of a party. Of course, this is directly and clearly impacted by the openness of the access control layer. However, we still include this as a factor, as it can be impacted by openness for other layers as well.

In addition, access control can be more or less refined. Refinedness of access control is determined by the coarseness by which access to data can be controlled. This coarseness refers to the data itself in this case (e.g., full documents or individual data elements). A higher level of refinedness means that businesses can decide to provide different levels of openness to data, based on their need to control risks.

To avoid risks of unauthorised access to information, security is an important factor for comparison as well. There are several ways in which parties can get unauthorised access to information. However, not all of them are related to openness. An inference attack happens when sensitive information can be inferred from non-sensitive information, for instance using datamining [31]. In addition, parties can disclose information to others that are not authorised to have access to this information.

4 COMPARING THE CONFIGURATIONS

In this section, we compare the three configurations specified in section 2. For each of them, we discuss the differences we found between the configurations for each of the factors in the comparison framework and we provide a discussion of how this related to the openness of the configurations on the different architectural layers. These results were derived from the workshop with experts from the container-shipping business, discussion with domain experts and from logical analysis.

In each subsection, we first provide an analysis of the impact of each level of openness for each layer. Then, we summarize these results in a table (see table 2-7). In some cases, we were not able to establish exactly what the impact of openness of an architectural layer was. In those cases, this layer is left out of the table. In the next section, we provide an overview of these findings and determine what configurations are best in different situations.

4.1 Information accessibility

In the completely open configuration, access control is delegated to other parties. This results in less steps for businesses to control access when providing access. For gaining access, delegating access control results in parties having to request access from a different party. We did not find an impact of this on the ease of gaining access to data for government organisations or businesses.

In the open configuration, the architecture has an open information layer. Parties can therefore directly share new data via the system. No prior processing of the data is required to encrypt it, for instance. Once data is shared, parties can immediately use it. A high level of openness for the information layer in the open configuration, is thus associated with a high ease of gaining and providing access to information.

The party supplying the data is themselves responsible for access control in the closed configuration. They should thus fulfil the steps to make a decision and verify the identity of the requesting party. They thus have to perform more steps in order to provide access to data than in the open configuration. We could not find an impact on the ease of gaining access to data, for the access control layer of the closed configuration.

In the closed configuration the ease of providing access is additionally hampered because parties need to encrypt data before they can actually share the data via the system. Furthermore, after the encrypted data has been shared via a system with a closed configuration, several things need to happen before parties can access it. First of all, a party that wants access should obtain the encrypted data. Then, they should request a key from the party that supplied the encrypted data. A closed information layer of the closed configuration is thus associated with a lower ease of gaining access to data.

The level of openness for the access control layer in a context-based open configuration does not seem to impact the efforts required to gain access to information. However, it does affect the

Table 2: Impact of the configurations on data accessibility

	Open	Context-based	Closed
Access control layer	Ease of providing access: High	Ease of providing access: Intermediate	Ease of providing access: Low
Information layer	Ease of providing access: High Ease of gaining access: High	Ease of providing access: Low Ease of gaining access: Depends on sensitivity	Ease of providing access: Low Ease of gaining access: Low

ease of providing access. Businesses in an information sharing system with a context-based open configuration delegate part of access control. This means that they do not need to perform all steps to control access themselves. The ease of providing access to data is therefore between that of an open access control layer and a closed access control layer.

In the context-based open configuration, the data that is to be shared, first needs to be divided into different data elements. Businesses should determine for these data elements whether they are sensitive or not and thus whether they should be encrypted. Then, they need to encrypt that data. This means that they have to perform additional steps, possibly even compared to the configuration with the completely closed information flow. Thus, the ease of providing access to data is lower. After the data is shared via an information sharing system with a context-based configuration, the steps necessary to access it depends on whether data elements are sensitive to the business supplying them. The data elements for which there is no need to keep them confidential are directly accessible, such as in an open configuration. However, to access sensitive, encrypted data elements, parties need to go through the same steps as in a closed configuration.

We could not find a clear relationship between the openness for the software layer for the configurations and information accessibility. In addition, it is important to mention the role of standardisation in information accessibility as well. Standardisation allows for the automation of certain tasks. This automation can lead to the efforts of performing steps becoming nihil after implementing the appropriate systems. For instance, if only certain documents are shared with a standard format, the identification of sensitive data elements and their encryption can be automated. It will require some initial effort to design a system that does this and to make a list of data elements that are considered sensitive. However, after this, the structural efforts required will be lower.

4.2 Scalability

New users of an information sharing system with an open configuration, first of all, need the appropriate software to be able to link to the information sharing system. As the data is open source, it is freely available, and thus easy to obtain. To enable the sharing of new types of data, new data elements need to be added to the interface with the information sharing system, or other modifications need to be made. This can be easier when the information sharing system is based on open source data.

Access control is fully delegated for the information sharing system in the open configuration. When the number of users increases, or the types of data that are shared increases, this can result in additional access control requests. If access control of all businesses are delegated to a single party, then the number of requests that need to be handled by this party might quickly become high, lowering scalability.

However, the delegation of access control allows for federated access control and federated identity management, meaning that a user can use their credentials from one or more service providers to get access to resources of other service providers [3]. In the container-shipping domain, access control can be federated to trusted third parties, such as port community systems. This creates a network of these trusted third parties that are all connected to the information sharing system. If access control is federated, less requests for access might end up at a single party and this can improve scalability. For an open access control layer, scalability thus varies according to whether access control is federated.

At the information layer, information is not separated into individual data elements for the open configuration. This means that it is not necessary to make access control decisions for individual data elements. This can protect scalability.

For the closed configuration, new users need to acquire the software to link to the information sharing system as well. As the software is proprietary, they need to either buy or license it. They cannot modify the source code of the software. Therefore, they cannot adapt it to meet their specific needs. Furthermore, they need to convince the party that owns the software to make the modifications necessary to allow the sharing of additional types of data. Scalability is thus lower due the closed information layer in the closed configuration, than in the open configuration.

Businesses themselves have to arrange access control to their data in a closed configuration. When the number of requests increases, either due to more users or due to sharing of additional types of data, each access request ends up with the business that supplied the data. This means that there is no single point where all the access requests need to be processed. On the other hand, individual businesses might have less capacity to handle access requests than, for instance, the party that controls the information sharing system.

In a completely closed access control layer, access control is not delegated at all. This means that it cannot be federated. If there are many users involved in the information sharing, it might not be feasible for a single business to arrange a way to verify their identity and determine whether they can be trusted. This can severely limit scalability.

Just as for the open configuration, in the closed configuration, the data is not separated into different data elements for which access control decisions need to be made. Therefore, less additional decisions need to be made when the volume of the data that is shared increases due to new users and new data types. This means better scalability.

In the context-based open configuration, parties within a trusted group of users can obtain and modify software. It requires a decision of this group to allow new users and to modify software to

Table 3: Impact of the configurations on scalability

	Open	Context-based	Closed
Software layer	New users: High scalability	New users: Depends on decision making process trusted group	New users: Low scalability
	New data types: High scalability	New data types: Depends on decision making process trusted group	New data types: Low scalability
Access control layer	New users: Depends on federation of access control	New users: Depends on federation of access control	New users: Low scalability
	New data types: Depends on federation of access control	New data types: Depends on federation of access control	New data types: Low scalability
Information layer	New users: High scalability	New users: Low scalability	New users: High scalability
	New data types: High scalability	New data types: Low scalability	New data types: High scalability

accommodate new data types. How easy this is and the number of new users and new data types that the group can make a decision on, depends on their decision-making process.

In a context-based open configuration access control is partially delegated. This means that scalability depends on what parts of access control are delegated and whether the parts that are delegated are federated. For example, a larger number of users could be accommodated when identity management is federated.

For the context-based configuration, the data is divided in data elements. When it has to be decided per data element whether access should be granted, this leads to a higher number of decisions that need to be made when more data is shared because new users or data types are added. This lowers scalability.

4.3 System reliability

For the open configuration, the access control is delegated. When access control for all parties is delegated to a single component of the system, then this component might have to take into account a high variety of users that request access to a variety of data. This makes it highly complex and thus increases the risk of failure. Furthermore, if such a central component fails, fault tolerance is low, as no access control decision can be made anymore, or even wrong decisions might be made on a large scale.

However, if access control is delegated to a federation, there might be several components involved that all serve a lower variety of users and data types. This would result in less complexity and a lower risk of failure for these components. On the other hand, federating access control introduces more failure points, as there are more components that can fail. This increases the risk of failure.

When access control is federated, if the failure in an access control component consist of mistakenly providing access to data of others, than this failure potentially has consequences for a lot of other parties. This reduces fault tolerance. On the other hand, if the failure consists of no longer being able to make access control decisions, access might still be controlled by the other components. This improves fault tolerance. When we look at the access control layer, we can thus conclude that the risk of failure and fault tolerance, depends on the way in which the delegation of access control is arranged in the open configuration.

In the open configuration, information is shared without encryption. This means that there are less components involved in the sharing of the information that could fail. This reduces the risk of failure for the open configuration. We did not find an impact on fault tolerance based on the open information layer in the open configuration.

In the closed configuration, parties arrange access control themselves. As they only need to control access to their own data, access control is less complex than in the case of an open configuration in which access control is not federated. This leads to a low risk of failure. Furthermore, if the systems of one party fails, other parties can still control access to their own data. Fault tolerance is thus high as well, if we look at the access control layer.

All data that is shared in the closed configuration is encrypted. This means that additional components are needed to encrypt the data, decrypt the data and to provide a key. As each of these components could fail, the risk of failure is higher in the closed configuration when we consider its closed information flow.

Access control is partially delegated for the context-based configuration. In the same way as for the open configuration, its fault tolerance and the risk of failure depends on how access control is delegated.

In a context-based configuration, additional components are not only needed to encrypt data, decrypt it and provide keys. In addition, components are needed to distinguish sensitive data elements that should be encrypted from those that are not sensitive. This is thus another component that might fail, leading to a risk of failure that is even higher than that for the closed configuration. Furthermore, the separation of the data in data elements, might make the access control process more complex, increasing the risk of failure even further.

We could not find an impact of the openness of the software layer on the system's reliability for the different configurations. The literature also is not clear on whether open source or proprietary software is more reliable [2].

4.4 Access control

To put in a request for access to information, a party needs to know that the data exists and where to put in a request. For this, they need to be able to link to the information sharing system. Providing parties with the opportunity to obtain the software necessary to make such a link, can be viewed as a first step in providing them access.

Table 4: Impact of the configurations on system reliability

	Open	Context-based	Closed
Access control layer	Fault tolerance: Depends on arrangements for delegation Risk of failure: Depends on arrangements for delegation	Fault tolerance: Depends on arrangements for delegation Risk of failure: Depends on arrangements for delegation	Fault tolerance: High Risk of failure: Low
Information layer	Risk of failure: Low	Risk of failure: Very high	Risk of failure: High

For the open configuration of the information sharing system, the software is open source. This means that the software needed to link to the information sharing system is available to all parties. Every party that wants to link to the information sharing system can obtain the software to do so, in that case. This results in a lower level of access control for the businesses. When we look at the access control layer of the open configuration, we can see that access control is delegated. From this, it follows directly that the level of control of businesses is low in an open configuration.

The data in the open configuration is not encrypted when it is shared. Therefore, businesses do not have the option to control access by key distribution. This means that after the data is stored somewhere else than in their system, they do not control it anymore. An open information flow is thus associated with a lower level of access control. Furthermore, it is associated with a low level of refinedness, since it is not possible to have different levels of control over parts of the data that is shared.

For the closed configuration, parties cannot directly influence who should be able to obtain the software needed to link to the information sharing system. This choice is left to the party that owns the software. This means that they have a lower level of control of access due to the low level of openness at the software layer. Furthermore, as the access control is not delegated at all in the closed configuration, businesses have a high level of control over their data.

When the data is shared, it is encrypted for the closed configuration. This means that storing the data, does not automatically mean having access to the data. This provides businesses with a means to control access to data even after it is shared. This increases their level of control. The refinedness of access control is as low for the closed configuration as for the open configuration. For both, all data is respectively not encrypted or encrypted.

In the context-based open configuration, the trusted group of users together decide who can obtain the software to link to the information sharing system. This means that in this configuration, businesses have a higher level of control over access than in the extreme configurations. In addition, the level of control depends on what parts and how much of the access control is delegated to others in the context-based configuration.

Table 5: Impact of the configurations on access control

	Open	Context-based	Closed
Software layer	Level of control: Low	Level of control: High	Level of control: Low
Access control layer	Level of control: Low	Level of control: Intermediate	Level of control: High
Information layer	Level of control: Low Refinedness: Low	Level of control: High Refinedness: High	Level of control: High Refinedness: Low

For the context-based configuration, businesses have a high level of control over the data that they encrypt. However, they have a low level of control over the data that they do not encrypt, once it is shared. As businesses can choose freely to encrypt or not encrypt data elements, they can choose for different levels of control over the data elements. Therefore, we can still consider a context-based open information layer to be associated with a high level of control over the data when we look at the information layer. Furthermore, the division of data into data elements, allows for a higher level of refinedness of access control in the context-based configuration.

4.5 Security

In the open configuration, businesses delegate access control and identity management to other parties. These other parties could disclose the information to others, even when they are not authorised to access the information. This makes the risk for unauthorised disclosure higher. However, another valid point of view is that it is unclear what unauthorised disclosure means for a completely open configuration. As the businesses have fully delegated the decision to provide access, it could be argued that they are no longer the party that is able to provide an authorisation. As soon as they knowingly share the data via a completely open system, they yield the right to authorise access to other parties.

In an open configuration, the data shared is not encrypted. This means that a party intercepting the data, might get unauthorised access to it. When they have access to the data, they can perform an inference attack as well as disclose it to additional parties.

The proprietary access control in the closed configuration does not have problems with other parties involved in access control that might provide unauthorised disclosure of data. The impact of a low level of openness on the access control layer, thus results in a higher level of data security. However, as parties control access by themselves, it is hard for them to determine what other data from other businesses a party requesting data has. This means that it is hard for them to make access control decisions that reduce the risk of an inference attack.

The closed configuration has better security because of the encryption of the data. Parties that want to access the data illegitimately, need to first intercept the encrypted data, and then they need to obtain a key. If it is harder for parties to intercept data, it is harder for them to perform inference attacks on it and disclose it to others.

In the context-based open configuration, part of access control is delegated. The risk of unauthorised disclosure depends on who access control is delegated to. Furthermore, it is possible to delegate parts of access control in such a way that it could help to reduce inference attacks. If the access control decision for all data is delegated to a central component of the information sharing system, then this component could keep track of who was provided with access to what data. The decision can then be made on basis of business rules provided by the businesses that supply the data. These business rules could express that parties that already have access to certain data elements, should not get access to others. Businesses could choose to not allow combinations of data elements from which sensitive information can be derived.

In an information sharing system with a context-based open information layer, businesses can encrypt the data elements for which it is important to protect security. The risk of interception for this data is the same as for a system with a closed information layer. The data that is not encrypted could be accessed directly when it is intercepted. However, this will be data for which the business is not concerned about its security. This usually will be when the data is not sensitive to them and when it does not have a commercial value. This means that it might not be worth for perpetrators to try to intercept this data, lowering the risks of such interception.

Domain experts indicate that often businesses perceive open source software as less secure and therefore would rather choose proprietary software. However, according to the literature, proprietary software is not inherently more secure than open source software and vice versa [2, 4]. Open source software, or even available source software could improve security by allowing parties to view the source and find security issues [2]. On the other hand, openness provides possible perpetrators with a lot of knowledge as well, that they might misuse [2]. The context-based scenario could provide a nice middle ground in which only trusted parties can view and modify the source code. Further research is required to say something definitive about this and examine whether such a solution would be more secure and would also be perceived as more secure by businesses.

5 SUITABILITY OF THE DIFFERENT LEVELS OF OPENNESS

The comparison of the different configurations of the information sharing system, shows that the impact that the level of openness has on the possible advantages of information sharing and the possibilities for risk control is highly complex. In some cases, the impact of a level of openness of a layer for a configuration might not even be unambiguous (e.g., for impact of a context-based open information layer on access control). In addition, we observe that often there is not a linear relationship between levels of openness and the extent to which advantages can be obtained or risks can be controlled. An example of this is that refinedness of access control is low for architectures that have a configuration with a completely open or completely closed information layer, but not for the context-based configuration.

Table 6: Impact of the configurations on security

	Open	Context-based	Closed
Access control layer	Inference attack risk: Unknown	Inference attack risk: possible to lower by delegating certain parts of access control	Inference attack risk: High
	Unauthorised disclosure risk: Unclear	Unauthorised disclosure risk: Depends on the parties that access control is authorised to	Unauthorised disclosure risk: Low
Information layer	-Inference attack risk: High	Inference attack risk: Low	Inference attack risk: Low
	Unauthorised disclosure risk: High	Unauthorised disclosure risk: Low	Unauthorised disclosure risk: Low

This high complexity also means that it is impossible to conclude that one level of openness or a configuration is better in general than the other. This is very situation specific, as each of them has its advantages and disadvantages. However, we can say that something about what configuration is appropriate in what situation.

Designers should choose a configuration with a high level of openness when they need their system to be highly scalable. They can further improve scalability of such a configuration by federating access control. On the other hand, this choice does result in less opportunities to control risks. An open configuration thus is only an appropriate choice when the consequences associated with losing control over the data for businesses are low. Otherwise, businesses might not want to share most of their data. Furthermore, a fully open configuration can be a good choice when it is important that businesses and government organisations do not have to put in a lot of effort into providing or gaining access to data.

The choice for a closed configuration should be made only when scalability is not an important priority, due to the scalability issues resulting from the use of proprietary software and not delegating access control. It is important to take into consideration that in a closed configuration, when there is a high variety of parties that could request access to data, it might be impossible to verify their identities for the businesses. This makes a completely closed access control layer infeasible in such situations.

A fully closed configuration is a good choice in a situation where losing control over data can have big consequences for businesses. In addition, when an information sharing system with a closed configuration is used, businesses should be willing to put more effort in providing parties with access to their data. Businesses and government organisations that want to obtain access should be willing to go through additional steps as well. The efforts for the party providing data can be reduced by standardising and automating steps (e.g., automating access control decisions).

A context-based open configuration be beneficial when there is a mix between data that is and that is not sensitive. Furthermore,

Table 7: An overview of the configurations and a description of when they are suitable when looking at the different architectural layers

	Open	Context-based	Closed
Information accessibility	High importance	Low importance for party providing access, or possible to automate processes. High importance for party gaining access	Low importance, or possible to automate processes
Scalability	High importance	Low importance	Low importance and limited number of users request access
System reliability	Intermediate importance	Low importance	High importance
Access control	Low importance	Mix of sensitive/non-sensitive data	High importance
Security	Low importance	Mix of sensitive/non-sensitive data	High importance

this configuration is appropriate when the government organisations and businesses need to gain access to data easily as well. The context-based configuration provides high accessibility for the data that is not sensitive and a high level of control for data that is sensitive. These benefits, however, come at the costs of a lower ease of providing access to data, although this can be countered if the processes involved in sharing the data can be automated. Furthermore, reliability of a system with a context-based configuration is lower than for the other options. Reliability thus should not be a high priority. The same is the case for scalability.

6 CONCLUSIONS, LIMITATIONS AND SUGGESTIONS FOR FURTHER RESEARCH

A B2G information sharing system should balance the advantages of open information sharing for government organisations and businesses with the need for risk control by businesses. We identified three architectural layers at which information sharing systems can have different levels of openness, namely the software layer, the access control layer and the information layer. We compared the impact of different levels of openness by comparing open, closed and context-based open configurations of information sharing systems.

We found that the impact of different configurations on the advantages and risk control are highly complex. This means that different configurations are appropriate in different situations. This requires a designer of B2G in formation sharing systems to not just determine how important risk control and gaining advantages is to businesses and government organisations. They need to establish the importance for these parties of different factors (e.g., security, scalability) that impact the advantages and risk control. In that way, they can get a more nuanced view of the situation and they can better determine what configuration is suitable.

Our approach has some limitations. As the impact of openness is highly complex, it is likely that the results are not comprehensive and that there exist other ways in which the possibilities for advantages and risk control are impacted. It would especially be interesting to look at additional configurations that are in between the extremely open and closed ones. Furthermore, additional factors that need to be looked at to determine the impact on advantages and risk control might be identified in further research. Furthermore, future research focusing on studying more practical cases might be used to extend the current insight.

ACKNOWLEDGMENTS

The work in this paper is part of the project JUridical and context-aware Sharing of informaTion for ensuring compliance (JUST). This project is funded by the Netherlands Organisation for Scientific Research (NWO) as part of the ISCOM programme (Innovation in Supply Chain Compliance and Border management) under grant number 438-13-601. Furthermore, the work was supported by the "Governing public-private information infrastructures" research project, which is financed by NWO as Veni grant 451-13-020 and the CORE Project (grant nr. 603993), which is funded by the FP7 Framework Program of the European Commission. Ideas and opinions expressed by the authors do not necessarily represent those of all partners in these projects.

REFERENCES

[1] Nitesh Bharosa, Marijn Janssen, Remco van Wijk, Niels de Winne, Haiko van der Voort, Joris Hulstijn, and Yao-Hua Tan. 2013. Tapping into existing information flows: The transformation to compliance by design in business-to-government information exchange. *Government Information Quarterly* 30 (2013), S9–S18. https://doi.org/10.1016/j.giq.2012.08.006

[2] A. Boulanger. 2005. Open-source versus proprietary software: Is one more reliable and secure than the other? *IBM Systems Journal* 44, 2 (2005), 239–248. https://doi.org/10.1147/sj.442.0239

[3] David W. Chadwick. 2009. Federated identity management. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 5705 LNCS (2009), 96–120. https://doi.org/10.1007/978-3-642-03829-7_3 arXiv:arXiv:1011.1669v3

[4] Crispin Cowan. 2003. Software security for open-source systems. *IEEE Security and Privacy* 1, 1 (2003), 38–45. https://doi.org/10.1109/MSECP.2003.1176994

[5] Customs Administration of the Netherlands. 2014. Pushing boundaries: The Customs Administration of The Netherlands' Point on the Horizon for the Enforcement on Continuously Increasing Flows of Goods. (2014).

[6] Stanley E. Fawcett, Paul Osterhaus, Gregory M. Magnan, James C. Brau, and Matthew W. McCarter. 2007. Information sharing and supply chain performance: the role of connectivity and willingness. *Supply Chain Management: An International Journal* 12, 5 (2007), 358–368. https://doi.org/10.1108/13598540710776935

[7] Annabelle Gawer and Michael a. Cusumano. 2013. Industry Platforms and Ecosystem Innovation. *Journal of Product Innovation Management* 31, 3 (oct 2013). https://doi.org/10.1111/jpim.12105

[8] Irina Harris, Yingli Wang, and Haiyang Wang. 2015. ICT in multimodal transport and technological trends: Unleashing potential for the future. *International Journal of Production Economics* 159 (2015), 88–103. https://doi.org/10.1016/j.ijpe.2014.09.005

[9] Paul Hart and Carol Saunders. 1997. Power and Trust: Critical Factors in the Adoption and Use of Electronic Data Interchange. *Organization Science* 8, 1 (feb 1997), 23–42. https://doi.org/10.1287/orsc.8.1.23

[10] David Hesketh. 2010. Weaknesses in the supply chain: who packed the box. *World Customs Journal* 4, 2 (2010), 3–20.

[11] Dirk Homscheid, Jérôme Kunegis, and Mario Schaarschmidt. 2015. Private-Collective Innovation and Open Source Software: Longitudinal Insights from

Linux Kernel Development Dirk. In *14th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2015 Delft, The Netherlands, October 13-15, 2015*. https://doi.org/10.1007/978-3-319-25013-7_24

[12] Joris Hulstijn, Wout Hofman, Gerwin Zomer, and Yao-Hua Tan. 2016. Towards Trusted Trade-Lanes. *Electronic Government: Proceedings of the 15th IFIP WG 8.5 International Conference, EGOV 2016* Egov (2016), 299–311. https://doi.org/10.1007/978-3-319-44421-5_24

[13] Thomas Jensen and Yao-Hua Tan. 2015. Key Design Properties for Shipping Information Pipeline. In *Open and Big Data Management and Innovation.*, Marijn Janssen, Matti Mäntymäki, Jan Hidders, Bram Klievink, and David Hutchison (Eds.). Springer International Publishing, 491–502. https://doi.org/10.1007/978-3-319-25013-7

[14] H Russell Johnston and Michael R Vitale. 1988. Creating Competitive Advantage With Interorganizational Information Systems. *Mis Quarterly* 12, 2 (1988), 153–165.

[15] Bram Klievink, Marijn Janssen, and Yao-Hua Tan. 2012. A Stakeholder Analysis of Business-to-Government Information Sharing: the Governance of a Public-Private Platform. *International Journal of Electronic Government Research* 8, 4 (2012), 54. https://doi.org/10.4018/jegr.2012100104

[16] Bram Klievink, Alessia Neuroni, Marianne Fraefel, and Anneke Zuiderwijk. 2017. Digital Strategies in Action: A Comparative Analysis of National Data Infrastructure Development. *18th Annual International Conference on Digital Government Research (dg.o 2017)* (2017), 129–138. https://doi.org/10.1145/3085228.3085270

[17] Bram Klievink, Eveline van Stijn, David Hesketh, Huib Aldewereld, Sietse Overbeek, Frank Heijmann, and Yao-Hua Tan. 2012. Enhancing Visibility in International Supply Chains: The Data Pipeline Concept. *International Journal of Electronic Government Research (IJEGR)* 8, 4 (2012), 14–33. https://doi.org/10.4018/jegr.2012100102

[18] M. Lynne Markus and Quang "Neo" Bui. 2012. Going Concerns: The Governance of Interorganizational Coordination Hubs. *Journal of Management Information Systems* 28, 4 (apr 2012), 163–198. https://doi.org/10.2753/MIS0742-1222280407

[19] Sietse Overbeek, Yao-Hua Tan, and Gerwin Zomer. 2011. IT Innovations Enabling Seamless and Secure Supply Chains WITNESS 2011.

[20] Adam M. Ross, Donna H. Rhodes, and Daniel E. Hastings. 2008. Defining changeability: Reconciling flexibility, adaptability, scalability, modifiability, and robustness for maintaining system lifecycle value. *Systems Engineering* 11, 3 (2008), 246–262. https://doi.org/10.1002/sys.20098

[21] Boriana Rukanova, Helle Zinner Henriksen, Stefan Henningsson, and Yao Hua Tan. 2017. The anatomy of digital trade infrastructures. *Lecture Notes in Business Information Processing* 295, January (2017), 184–198. https://doi.org/10.1007/978-3-319-64930-6_14

[22] G. Stefansson. 2002. Business-to-business data sharing: A source for integration of supply chains. *International Journal of Production Economics* 75, 1-2 (2002), 135–146. https://doi.org/10.1016/S0925-5273(01)00187-6

[23] Yao-Hua Tan, Niels Bjørn-Andersen, Stefan Klein, and Boriana Rukanova. 2011. *Accelerating Global Supply Chains with IT-Innovation.* 1–379 pages. https://doi.org/10.1007/978-3-642-15669-4

[24] David Tilson, Kalle Lyytinen, and Carsten Sørensen. 2010. Digital infrastructures: The missing IS research agenda. *Information Systems Research* 21, 4 (2010), 748–759. https://doi.org/10.1287/isre.1100.0318

[25] Amrit Tiwana, Benn Konsynski, and Ashley a. A. Bush. 2010. Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research* 21, 4 (dec 2010), 675–687. https://doi.org/10.1287/isre.1100.0323

[26] Luca Urciuoli, Juha Hintsa, and Juha Ahokas. 2013. Drivers and barriers affecting usage of e-Customs - A global survey with customs administrations using multivariate analysis techniques. *Government Information Quarterly* 30, 4 (oct 2013), 473–485. https://doi.org/10.1016/j.giq.2013.06.001

[27] Sélinde van Engelenburg, Marijn Janssen, and Bram Klievink. 2017. Design of a software architecture supporting business-to-government information sharing to improve public safety and security. *Journal of Intelligent Information Systems* (2017). https://doi.org/10.1007/s10844-017-0478-z

[28] Sélinde van Engelenburg, Marijn Janssen, Bram Klievink, and Yao-hua Tan. 2017. Comparing a Shipping Information Pipeline with a Thick Flow and a Thin Flow. In *Electronic Government, 16th IFIP WG 8.5 International Conference, EGOV2017, St. Petersburg, Russia, September 4-7, 2017, Proceedings*, Vol. 10428. Springer International Publishing AG, 228–239. https://doi.org/10.1007/978-3-319-64677-0

[29] Eveline van Stijn, David Hesketh, Yoa-Hua Tan, Bram Klievink, Sietse Overbeek, Frank Heijmann, Markus Pikart, and Tom Butterly. 2011. The Data Pipeline. In *Global Trade Facilitation Conference 2011*, Vol. 3. 27–32.

[30] Eric von Hippel and Georg von Krogh. 2003. Open Source Software and the "Private-Collective" Innovation Model: Issues for Organization Science. *Organization Science* 14, 2 (2003), 209–223. https://doi.org/10.1287/orsc.14.2.209.14992

[31] C Zhang and S H Li. 2006. Secure information sharing in internet-based supply chain management systems. *Journal of Computer Information Systems* 46, 4 (2006), 18–24.