Access 4 Learning Community

Powered by: SIF

Simple. Secure. Scalable. Standard.

SIF Infrastructure Specification 3.3: SIF Data Protection Enforcer Service

www.A4L.org

Version 3.3, May 2019

1. An Introduction to Privacy Protections

Schools, administration officials, classroom teachers and education authorities are faced with an overwhelming array of applications to assist in a student's learning, school administrative functions and community communication tools. Data may travel from inhouse on premises, highly trusted sources to external applications which may not be as trusted. A paper or digital contract is the only primitive protection for this data exchange.

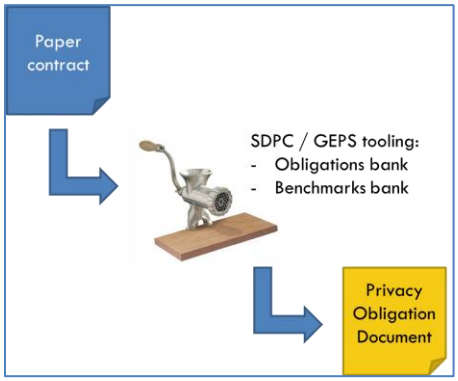
With the release of 3.3 SIF Infrastructure, an approach has been introduced that specifies a machine-readable form of the contract and provides a mechanism that governs the process by which an external application requests data. By referencing the correct version of the contract in each request appropriate filters are able to be applied to the data returned on each request; thereby ensuring the external application receives only the information that it should. By referencing the digital contract, the external application is also acknowledging certain obligations it must adhere to when it comes to handling the returned data. This digital contract is called a Privacy Obligation Document or POD.

The components that make up the privacy protections introduced in the SIF Infrastructure are:

1. POD - Privacy Obligation Document: An artefact derived from a paper contract which contains details of the parties involved, the data which can be transferred from one party to another, details of the technical benchmarks which must be adhered to (e.g. encryption levels) and details of any additional parties which may handle the data.
2. POD Lookup Service – Officially the "Privacy Obligations Registry Utility Service" this provides a means by which external applications request and obtain the current POD that applies to them
3. POD Enforcer – Officially the "Data Protection Enforcer Service" this service:
    a. Checks that any incoming requests from external applications are referencing their correct POD
    b. Uses the rules from the applicable POD to clean the raw data being returned in a request, ensuring that a 'cleansed' data set is returned to the requesting external application.

2. How do PODs work in practice?

1. A standard contract (paper or otherwise) is signed off. This forms the agreement between the provider of data (typically the school or school district) – known as the Data Controller and the downstream consumer of the data known as the Data Processor.
2. A POD is then created from this contract by breaking the contract down into the various contract clauses, linking these to obligations, before finally defining benchmarks which call out the standards a Data Processor is expected to honor when they handle the data. In the U.S. please contact the SDPC[1] for details on contract processing tools, in the AU please contact NSIP[2].



3. Once a POD has been created, it can then be enforced in any environment capable of supporting these. In SIF enabled environments, the POD lookup service and the POD Enforcer are needed to ensure the right POD is being referenced and that the filtering is being applied.

   In the simple example below, a Data Processor (e.g. 3rd party attendance package) is assumed to have already obtained the correct POD, and makes a request of the Data Controller (e.g. School District).

   a. The POD is registered with the POD Lookup Service, with its identifying Data Privacy Marker (DPM). (Using an HTTP PUT request to the Privacy Obligations Registry Service)

   b. The POD is made available to the POD Enforcer, with its identifying Data Privacy Marker (DPM)

[1] *Student Data Privacy Consortium (SDPC): https://privacy.A4L.org*

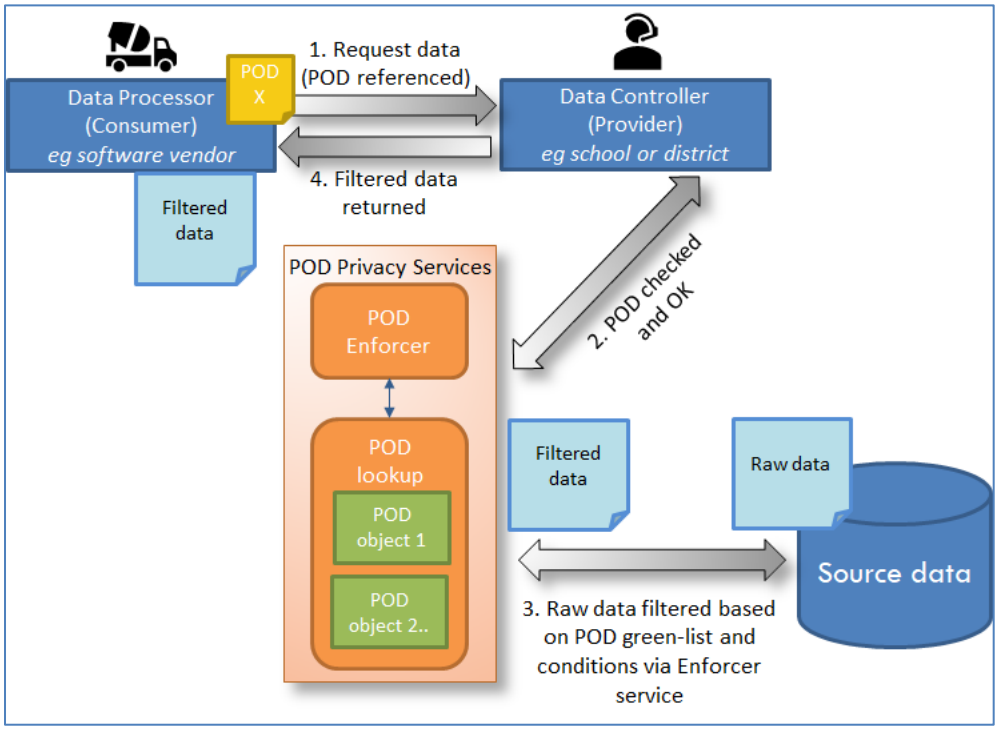
[2] *National Schools Interoperability Program (NSIP): http://www.nsip.edu.au*

c. Data requests from data processors include the applicable DPM and are mediated by the POD Enforcer.

d. For each data request received from a data processor, POD Enforcer ensures:

   i. The POD identified by the DPM provided in the data request, is the most up-to-date and current POD for that data processor.

   ii. Applies the data cleansing rules specified in the applicable POD, to the data received from the data provider, before the cleansed dataset is returned to the data processor



As depicted in the above diagram, a Data Processor (e.g. 3rd party attendance package) is assumed to have already obtained the correct POD and its identifying DPM and makes a request of the Data Controller (e.g. School District).

- Step 1 – A data request is made including a DPM that identifies the applicable POD.
- Step 2 – The POD is checked by making a call to the POD lookup service (using the DPM).
- Step 3 – The data request is forwarded to appropriate backend systems and raw data is extracted. The POD Enforcer applies the data cleansing rules from the applicable POD,

ensuring the data conforms to the fields and other conditions the Data Processor is entitled to receive.

- Step 4 – Filtered data is returned to the Data Processor

3. Data Privacy Services

The SIF 3.3 infrastructure introduces services that provide functionality in regard to data protection legislation. The services are intended to be "support" styled services. That means that SIF cannot enforce nor is it responsible for enforcing data privacy legislation; but it can provide utilities that simplify the implementation of data protection obligations that a particular implementation may be required to adhere to.

The Data Protection Services consist of two (2) core services. They are briefly detailed below:

a. **Privacy Obligation Registry Utility Service**: This service is a standard Utility Service as defined by the SIF 3.x Infrastructure Standard. Its main responsibilities are to provide a **lookup service** where data protection obligations (rules) are stored and maintained in a "machine readable" form. Multiple rules are grouped into a Privacy Obligation Document (POD) which provides a "contract" between vendors, states, districts etc. SIF 3.x Participants (e.g. SIF 3.x Adapters, Brokers etc.) can use the service to enquire and retrieve the latest set of rules (POD) for a particular participant. Administrators are expected to maintain these rules via the POD to reflect any changes to legal terms detailed via traditional paper contracts. The details of the **Privacy Obligation Registry** Utility Service is outlined in the SIF Infrastructure - Utility Services Specification and repeated in this document for reference.

b. **Data Protection Enforcer Service (DPE)** : This service can be used to apply data protection obligations that a SIF 3.x Adapter is bound to. The main functionality (as defined in this document) consists of taking a data set and applying the rules found in the relevant Privacy Obligation Document (POD) to produce a clean, privacy aware data set. The final "sanitized" data set is safer to be released to the requestor of the data. Generally, this functionality consists of filtering elements from a known self-contained data set.

4. Limitations of the Data Protection Enforcer Service

There are limitations to the enforcer service: the enforcer service is intended to act as a filter on the data set returned by a particular query. Therefore, not all rules specified in a POD may be realized by the enforcer service on its own: it may need to work in tandem with the selection of appropriate environments, contexts and zones to ensure the cleanest, most privacy aware data set is transmitted to recipients. For example, restricting a set of student attendance records to only be the senior year (years 7-12) rather than the whole school (year 0-12) cannot be accomplished fully by the Enforcer Service as it requires an external object lookup to determine the subset of filtering required. In this scenario, the attendance objects don't contain a student's year level and therefore the enforcer would need to access other objects (StudentPersonal) to determine the student's year levels. So, whilst the enforcer service cannot do this alone, subset filtering may be achieved in SIF via "Named queries" or "Contexts".

The Data Protection Enforcer Service may filter mandatory elements, which means that the SIF payloads subject to filtering will not necessarily validate against the strongly validating SIF schema (used for GET and CREATE). SIF payloads subject to filtering should instead be validated against the weakly validating SIF schema, which treats all elements as optional (used for UPDATE).

5. Specification Details - Data Protection Enforcer (DPE)

5.1. Data Privacy Marker (DPM)

The idea behind the Data Privacy Marker is that a SIF 3.x Adapter intending to work with a DPE must provide a marker in each call it makes to exchange any data. Given that marker, the DPE can validate whether the SIF Adapter has the latest knowledge about the PODs that applies to the data exchange, and the DPE can also use the marker to identify what PODs must be applied to the SIF Adapter's data. In other words, this marker is a sort of "entry ticket" to be used with the DPE. If the DPM that the Consumer presents has expired, then the Consumer is alerted that their knowledge of current applicable POD is now likely out of date, and they should use the Privacy Obligation Registry Utility service to acquire the current POD.

It is the responsibility of the DPE service to issue these markers (entry tickets). Each SIF Adapter will require a marker in order to interact with a DPE, and these markers will likely change over time as PODs are updated. (PODs will likely change for a variety of reasons including change of privacy contact details.) In practice, each modification to a POD requires a new marker to be issued to SIF Adapters that are bound to the changed POD: the previously issued markers are expired, and any request using an expired DPM will be rejected.

The DPM is an opaque marker, similar to the "Changes Since" opaque marker described in the SIF Infrastructure Service specification. As for "Changes Since" opaque marker the main principles for the "Data Privacy Marker" are:

a. A SIF Adapter must request the latest "Data Privacy Marker" when it joins a SIF Environment. Potentially the SIF Adapter may need to request the latest "Data Privacy Marker" from time to time, to ensure it is up to date (note: A marker for a particular adapter may change as PODs change).

b. A SIF Adapter **MUST NOT** make any assumptions about what the marker means. It shall not infer any information from the value of the marker. Hence the DPM is considered "opaque".

   The DPE service that issues the DPM is fully responsible for maintaining these markers and validating them. It is the DPE service that has the knowledge as to what the marker refers to, how to interpret it and what it encompasses. Markers can expire and are not perpetual. A typical example is when a specific POD is modified then the marker(s) related to this POD must change to indicate a different version of a POD.

5.2. Functions for Data Privacy Marker

The following functions of the DPE Service relate to the "Data Privacy Marker":

- **Get Data Privacy Marker**: This function is called by a SIF Adapter to retrieve the current Data Privacy Marker from the DPE service for use with a particular SIF service (e.g. a student object service, a school service path, a timetable functional service). The retrieved marker is specific to the calling adapter and is shared among all services offered by the adapter. It is expected that the SIF Adapter will use this marker in each subsequent call to the Object Service, Service Path, Functional Service etc.

5.2.1. REST Call Details – Get Data Privacy Marker

URL

https://.../<serviceConnector>/dataprotectionenforcer/dataprivacymarker

Request

Operation: HTTP GET

The HTTP GET allows a number of additional parameters to be provided for the DPE service to return the appropriate marker for a specific SIF Adapter. The parameters are listed in the table below. The table also indicates who may provide these parameters, how they are conveyed and whether they are mandatory or optional.

Parameter Name	Required	Conveyed as	Provided by
applicationId[1)]	Y	Q	P or C
partyId[1)]	Y	Q	P or C
role[1)]	N	Q	P or C
contractId[1)]	N	Q	P or C
podIds	N	Q	P or C
fingerprint	Y	H	P
Cache-Control: no-cache	Y	H	C

Parameters marked with a [1)] refer back to elements in the POD of the Privacy Obligation Registry.

Required:

Y; Yes

N: No (optional)

Conveyed as:
Q: URL Query Parameter
H: HTTP Header
M: Matrix Parameter

Provided by:
P: Provider or Environment Provider on behalf of Consumer
C: Consumer

It is important to note that the caller must provide the “Cache-Control” HTTP header with the value of “no-cache” to ensure that no cached Data Privacy Marker is returned.

Response

Success

The response consists of a standard HTTP Status 204 (No Content) and a HTTP Header called **dataprivacymarker** containing the Data Privacy Marker.

Failure

If the request fails a standard HTTP Error Status will be returned (i.e. 401 if authentication fails). Further a standard SIF Infrastructure Error payload is returned.

Authentication and Access Control

The DPE Service “exists” as a utility service in the “Global Zone” of each adapter environment. The authentication to that service is the same as with all SIF services relating to a given environment. The “QUERY” right must be set to “APPROVED” to enable an adapter access to this operation. All the other rights do not apply.

The following snippet from the adapter’s environment (XML notation) illustrates how the “Get Data Privacy Marker” service will appear in the global zone.

```
<service name="dataprotectionenforcer/dataprivacymarker" type="SERVICE">
  <rights>
    <right type="QUERY">APPROVED</right>
  </rights>
</service>
```

5.3. Functions for Data Cleansing

The function(s) listed in this section form the core DPE functionally. These are the operations that will apply PODs to payloads and return the “cleansed” payload that is safe to be returned to a requestor. As of SIF 3.3 there is one method available called “Apply POD”.

5.3.1. REST Call Details – Apply POD – Data Filtering

This function is responsible for applying POD rules on a payload (data) before the payload is released to the requesting SIF Adapter. The SIF Specification does not mandate that the DPE service, and especially this function, must use the POD Utility Service to fetch the POD rules. It is suggested it makes use of that utility service, but a SIF implementation can choose how or weather it utilizes the POD utility service. The POD Utility Service describes the outcome of Apply POD; it does not prescribe how that outcome is realized.

URL

https://.../<serviceConnector>/dataprotectionenforcer/filterrequest

Request

Operation

A **HTTP PUT** is used.

Payload(s)

The payload on the request is the SIF Object or Object Collection to be "cleansed". This can be in XML or JSON as with all SIF Object, Service Path etc. services. The response payload contains the "cleansed" data. This is the same SIF Object type as on the request but will have certain elements removed (filtered) according to the PODs that apply.

HTTP Headers

There is additional data required for this function to apply the correct POD rules. The minimum and mandatory data element is the "Data Privacy Marker" from the requesting SIF Adapter. The name of this HTTP header is '**dataprivacymarker**'. This HTTP Header is required for two reasons:

- Will determine which POD rules need to be applied to the request data payload
- Is used to ensure that the requestor has the latest valid marker (e.g. Validate marker).

Other HTTP Headers (refer to Base Architecture for the name of the HTTP header) that must be provided to ensure that the correct POD rules are applied include but might not be limited to:

- Zone
- Context
- Requestor Fingerprint
- Mime type (XML, JSON)
- JSON Notation indicator (refer to section 5.3.3 - JSON vs. XML Payloads)

The above values are required, and provided by the provider or environment provider, because some filtering rules may differ depending on the Zone, Context and/or fingerprint of the requestor.

Of course, there are additional standard HTTP Headers supported/required such as Content-Type, Content-Encoding etc.

Response

Success

If the call to this function succeeds, then a standard HTTP Status 200 (Ok) is returned and the response payload contains the "cleansed" data (see also "Payload" in the Request section). However, if all elements are filtered based on the POD then a HTTP Status 204 (No Content) shall be returned. This is also true for the case where a query is issued that returns no data, regardless of data filtering (e.g. get all year 5 students but the consumer is not allowed to see year 5 students).

Optionally the following HTTP Headers can be returned to the consumer:

- podId: The UUID of the particular POD that was applied
- podVersion: The specific version of the POD.

Failure

There are a number of reasons why this function may fail. Typical reasons are authentication failure, MIME type invalid etc. The appropriate HTTP Error Status code will be returned for these types of failures. Further the returned payload won't be the SIF Object from the request, but rather a standard SIF Infrastructure Error payload. This allows additional information to be returned to the caller about the failure reason. Optionally the following HTTP Headers can be returned to the consumer:

- podId: The UUID of the particular POD that was applied
- podVersion: The specific version of the POD.

There is one notable error that needs to be considered. The "Data Privacy Marker" that is passed to this enforcer method could be outdated or invalid. The list below states what HTTP Status code shall be returned for various error cases:

- **HTTP Status 401** (Unauthorized): Invalid or missing credentials.
- **HTTP Status 403** (Forbidden): The provided Data Privacy Marker is not valid (e.g. outdated).
- **HTTP Status 404** (Not Found): Invoke URL is invalid as per HTTP standard.

Authentication and Access Control

The DPE Service is provided as a utility service in the "Global Zone" of each adapter environment. The authentication to that service is the same as with all SIF services relating to a given environment. The "UPDATE" right must be set to "APPROVED" to enable an adapter access to this operation. All the other rights do not apply.

The following snippet from the adapter's environment (XML notation) illustrates how the "Get Data Privacy Marker" service will appear in the global zone.

```
<service name="dataprotectionenforcer/filterrequest" type="SERVICE">
  <rights>
    <right type="UPDATE">APPROVED</right>
  </rights>
</service>
```

5.3.2. Encryption Considerations

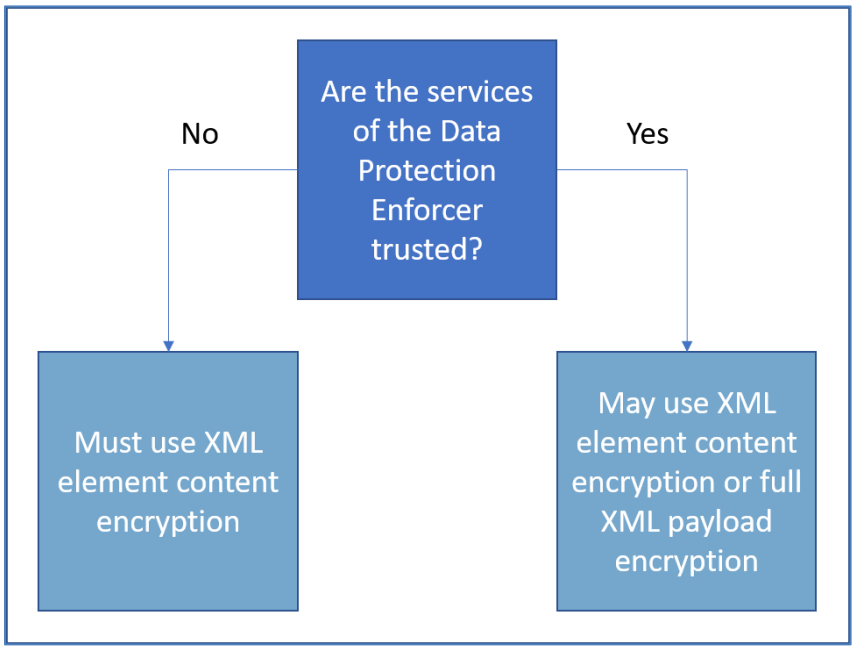
In some implementations payload itself may be encrypted. The encryption in this section refers to actual payload encryption and not to the transport layer security (HTTPS) encryption that is applied to convey the payload on the wire. If a payload is encrypted the DPE cannot apply sanitation to the payload without external help. This section outlines what help is needed to allow the DPE to apply changes to a payload.

5.3.2.1. XML Encryption

The XML standard supports two styles of encryption mechanisms.

- **Payload**: Apply encryption to the entire XML payload
- **Content of element**: Apply encryption to the content of XML elements only.

With regards to "sanitation help" each mechanism has a different implication and requirement for the DPE. Consider if the functions of the Data Protection Enforcer are occurring internally, with a 'trusted' party, or if an external system is providing the DPE service which may not be full trusted to see the payload detail.



Full XML Payload Encryption

To enable the DPE to apply filtering rules as defined in the POD it must be able to open up and read the payload, at minimum the XML element names. Since the payload is fully encrypted the DPE cannot apply filtering rules without decrypting the payload first. To get around this issue, the DPE will require the key to decrypt the payload, then apply the rules and encrypt the payload again before returning it to the caller.

To enable the de- and re-encryption of payloads the DPE would need a registry of keys that would need to be provided by the "producer" of payloads. In addition, if a public/private key system is used, each call to the DPE would need additional information about the "producer" of the payload to determine which key to use to perform de- and re-encryption. In this case it is assumed the "producer" trusts the DPE (and DPE services are provided in-house or by a trusted broker). The DPE service can provide its public key to a client who will encrypt the data. The client provides its public key with the payload encrypted using DPE's public key. Then the payload can be returned to the client safely.

XML Element Content Encryption

This encryption mechanism has no further requirements on the DPE because the element names are still in clear text and therefore filter rules can be applied. Filter rules do apply on elements and are defined in xPath notation in the POD. **This type of XML encryption is the preferred option** as it allows the DPE to work as if the entire payload is in clear text.

Note: if an element is optional, its presence by itself may signal information that should not be made available to third parties. The presence of such an optional element is not concealed through XML Element Content Encryption, even though their values are.

5.3.2.2. JSON Encryption

Unlike XML, JSON doesn't have an encryption standard as part of the JSON standard. However, encryption can still be applied. It is more of an "explicit" process to encrypt/decrypt JSON payloads than it is with XML where the standard supports it natively. As with XML encryption, the producer of the payload has the option to apply blank encryption on the entire payload or encryption can be applied to the "values" of each attribute. Once encryption is applied the same rules and challenges as with XML encryption will apply or arise. Please refer to previous section regarding XML encryption for a discussion about rules and challenges.

5.3.3. JSON vs. XML Payloads

As of SIF Infrastructure 3.3 SIF supports three MIME types:

- XML
- JSON Goessner Notation
- JSON Schema Aware Notation (e.g. the notation also used in PESC)

It is up to the implementation how element filter rules are expressed. It is expected that a binding document, applicable to the specific implementation, will detail what notation is used and how it might be interpreted. Options include but are not limited to xPath (applicable for XML payloads) and JSON Path (applicable for JSON payloads). It is entirely possible that a specific implementation may use an Object Service with a CSV structure, hence the element names of the filter rules simply list the heading of each valid column of the CSV structure. To support filter rules for any of the above MIME types, the POD will support filter rule notations specific for each mime type. This means that there is the potential that each filter rule set in the POD may require up to three sets of rules, one per MIME type. The table below summarizes the applicable filter rule notation for each MIME type. For further details about the POD and specific elements where filter rules are

maintained, refer to the **Privacy Obligation Registry Utility Service** in the Utility Services specification.

The table below lists what notation is most likely used for some typical SIF supported MIME types.

Mime Type	Filter Rule Notation	Comment
XML	xPath	
JSON Goessner	JSON Path	The JSON Path for Goessner will be different to the JSON Path for Schema Aware.
JSON Schema Aware	JSON Path	

The DPE service could inspect the "Content-Type" HTTP header and appropriate HTTP header listed in the Data Model Schema Negotiation Specification, to determine how to interpret the filter rule notation.

Note that the JSON Path for both types can be derived from the XPath, but that schema awareness is required for the transformation, wherever there is a repeating element:

- In JSON Goessner, The XPath fragment /ElementList/Element/, where Element is a repeated element, maps to two corresponding JSON Path fragments: ElementList.Element (for a single Element instance in the object) and ElementList.Element.* (for more than one Element instance in the object). That means that each repeating element in an XPath multiplies the corresponding JSON Paths by two.
- In JSON Schema Aware, the XPath fragment /ElementList/Element/, where Element is a repeated element, maps to ElementList.Element.*: repeating XML elements are always mapped to JSON Lists.

5.3.4. Risks with data queries and query validation

While data cleansing takes care of removing data from payloads (e.g. gender) the fact that data can be inferred by simply issuing an appropriate query may reveal sensitive data. For example, if a consumer is not allowed to see the "Gender" element of a student, and the DPE is set up to remove that element, the gender information can still be retrieved by simply issuing a query to return all "Female" students. While the consumer doesn't retrieve the gender element in the response of that query it still knows that the

returned students are female! This would be considered a data leak and possibly violate the intent of the POD.

Within SIF queries on data are supported through either eXtended Query services or through Query By Example (QBE). The eXtended queries are agreed upon queries and therefore are vetted before they are enabled. Through that vetting process it can be ensured that such "query" related data leaks are reduced. However, QBE style queries are open-ended and can be issued at the consumer's leisure. If a "generic" provider simply processes them without inspecting all the elements first, a data leak may occur. Because QBE uses Data Model payloads to describe the query the DPE provides a "validate query" method. If the query payload holds any elements that are in the list of filtered element, then the query should be rejected by the provider.

Important: The Data Protection Enforcer does not limit the various data queries which may be made by a consumer. Data query controls are handled elsewhere in the SIF specification. The query validation service may assist in reducing the risk but will not eliminate it.

5.3.5. REST Call Details – Query Validator

The query validator allows a QBE to be checked by the data protection enforcer to ensure that the requestor has the permission to run the request (QBE).

URL

https://.../<serviceConnector>/dataprotectionenforcer/queryvalidator

Request

Operation

A **HTTP PUT** is used.

Payload(s)

The payload on the request is the QBE SIF Object to be validated. This can be XML or JSON as with all SIF Object, Service Path etc. services.

HTTP Headers

There is additional data required for this function to validate against the correct POD rules. The minimum and mandatory HTTP Header is the "Data Privacy Marker" from the

requesting SIF Adapter. The name of this HTTP header is '**dataprivacymarker**'. This is required for two reasons:

- Will determine which POD rules need to be applied to the QBE data payload
- Is used to ensure that the requestor has the latest valid marker (e.g. Validate marker).

Other values (refer to Base Architecture for the name of the HTTP header) that must be provided to ensure that the correct POD rules are applied include but might not be limited to:

- Zone
- Context
- Requestor Fingerprint
- Mime type (XML, JSON)
- JSON Notation indicator (refer to section 5.3.3 - JSON vs. XML Payloads)

The above values are required because some filtering rules may differ depending on the Zone, Context and/or fingerprint of the requestor.

Of course, there are additional standard HTTP Headers supported/required such as Content-Type, Content-Encoding etc.

Response

Success

If the call to this function succeeds, meaning the QBE payload is valid, then the standard HTTP Status 200 (Ok) is returned. Optionally the following HTTP Headers can be returned:

- podId: The UUID of the particular POD that was applied
- podVersion: The specific version of the POD.

Failure

Failure means that the QBE payload is invalid and failed the check against the POD. The consumer is not authorized to issue the query. Other failure reasons include authentication failure, mime type invalid, invalid/outdated Data Privacy Marker etc. The appropriate HTTP Error Status code will be returned for these types of failures. Further the returned payload won't be the SIF Object from the request rather a standard SIF Infrastructure Error payload. This allows additional information to be returned to the caller about the failure reason.

- **HTTP Status 401** (Unauthorized): Invalid or missing credentials.
- **HTTP Status 403** (Forbidden): QBE payload failed validation meaning it is not allowed according to the POD. Note that this status is overloaded. Further meaning is that the provided Data Privacy Marker is not valid (e.g. outdated).

- **HTTP Status 404** (Not Found): Invoke URL is invalid as per HTTP standard.
- Optionally the following HTTP Headers can be returned:
    - podId: The UUID of the particular POD that was applied
    - podVersion: The specific version of the POD.

Authentication and Access Control

The DPE Service "exists" as a utility service in the "Global Zone" of each adapter environment. The authentication to that service is the same as with all SIF services relating to a given environment. The "UPDATE" right must be set to "APPROVED" to enable an adapter access to this operation. All the other rights do not apply.

The following snippet from the adapter's environment (XML notation) illustrates how the "Get Data Privacy Marker" service will appear in the global zone.

```
<service name="dataprotectionenforcer/queryvalidator" type="SERVICE">
  <rights>
    <right type="UPDATE">APPROVED</right>
  </rights>
</service>
```

6. DPE Service Usage

This section provides guidelines and workflows to when the DPE Service ought to be utilized. However, these are just guidelines and some implementations may take a slightly different approach in some cases to further improve performance. This will be most obvious in the case of the Event Message pattern that is discussed in this section.

The DPE service will be utilized at different points depending on the message patterns used. The two core patterns in SIF are request/response and eventing.

6.1. Request/Response

The usage of the DPE service is fairly straightforward in the request/response message pattern because the responder knows who the requestor for the data is, and therefore can apply appropriate privacy rules. The responder can do this by means of applying these rules itself or by using the DPE Service to cleanse the payload before sending it to the requestor.

6.2. Events

The invocation of the DPE service in the case of events will most likely occur at a different point than for the request/response message pattern. The main reason is that the publisher of the event doesn't know who will receive the event. The publisher is therefore not expected to call the DPE Service and will most likely publish uncleansed payloads. However, it will become the responsibility of the Queue Provider that the DPE Service is invoked **before (!)** the message is delivered to the caller of the "get next message" request from a queue. At that moment the Queue Provider knows who is requesting the message from the queue and can now apply appropriate privacy rules by means of invoking the DPE Service. Generally, this will be a feature of SIF 3.x Brokers as these are the infrastructure components that commonly provide the Queue endpoints.

If a consumer that reads messages from a queue presents the same Data Privacy Marker for each message read, it may at some stage find it has an invalid/outdated Data Privacy Marker. Since the "queue reading" consists of a request/response process, the same rules as with standard request/responses apply. Specifically, the HTTP error status codes returned are the same as listed in section REST Call Details – Apply POD – Data Filtering

As discussed in section 5.3.2, in situations where the Broker is not fully trusted by the provider of data, where events are to be supported the use of XML element content encryption is mandated (as opposed to implementing full xml payload encryption).

7. POD Registry

The Privacy Obligations Registry contains the set of data protection obligations captured in the Privacy Obligations Documents (PODs) which must be enforced. These obligations are ideally expressed twice each. Once in a human readable way, as it relates to a contract or law. Another time in a machine-readable fashion as part of a data access field list, condition list, deletion requirement or other predefined construct. For SIF integrations the data protection obligations are scoped by Consumer, Zone, and Context. (*Note: Machine readable portions of the POD may be configured when consumed or may be preconfigured and confirmed when consumed. For instance, a Context may be setup ahead of time to satisfy a condition).*

7.1. On the Wire Enforcement

The place where interoperability can play a key role in enforcing privacy is to reduce the data that arrives at a consumer to only what is needed. A POD has two key ways of doing this. First the data access field list may be used to express the fields that are permitted to arrive at a given Consumer. Second the condition lists may define the records that are allowed to be sent to a Consumer often in conjunction with a scoping mechanism such as a Context.

7.2. Off the Wire Enforcement

When it comes to enforcing the privacy rules expressed in a POD, the Consumer has a role to play. Some of the machine-readable portions, such as data deletion requirements, must be handled by the Consumer. Additionally, the human readable sections are most effective when shared with the user or administrator and an interface to view them **may** be provided.

7.3. Service Implementation Strategy

In order to achieve privacy, all components (the Provider, Consumer, and where applicable Broker) need access to the PODs that impact them. The below overview is designed to explain how to get the desired POD and where to impact the data transfer (see 7.4 Further Documentation). These steps assume you will be getting the PODs out of this services interface, however if one of these components also provides this Utility Service that may not be the case.

The Consumer must…

1. Retrieve its Data Privacy Marker (DPM) when it starts and whenever it receives a 403 error.
2. Retrieve the POD by included its DPM in the where clause of a query to this service.
3. Confirm it can function under the conditions of the POD.
4. Affirm it can either automate the requirements of the POD, or notify the administrator of the tasks they must complete in order to conform to the POD.
5. Include the current data Privacy Marker header in all requests it makes.
6. Treat data retrieved or delivered through any mechanism (events, another API, human input, etc.) under the same Benchmarks.

The Provider and/or Broker must…

1. Implement or relay requests to the POD service.
2. Return the most specific POD based on the DPM.
3. Confirm the DPM is current before responding to requests. Otherwise provide the Consumer with a 403/Forbidden HTTP error code.
4. Run every outgoing payload through the Privacy Enforcer Service (wherever it is implemented) before sending it on to the Consumer.
5. Include the Data Privacy Marker header with the DPM used when the payload was processed. Note: Because the POD that was applied may be specifically referenced, PODs must be retrievable by their resource ID like any other service. Note: To ensure the POD(s) attached to a DPM can be retrieved the where clause on the podToken field must be supported by the Utility Service Provider.

7.4. Further Documentation

For full data structure and examples please see the Infrastructure data model documentation.
DataModel:
http://specification.sifassociation.org/Implementation/Infrastructure/3.3/2/infrastructures/PrivacyServices.html#obj:Pod

7.5 Supported POD Operations

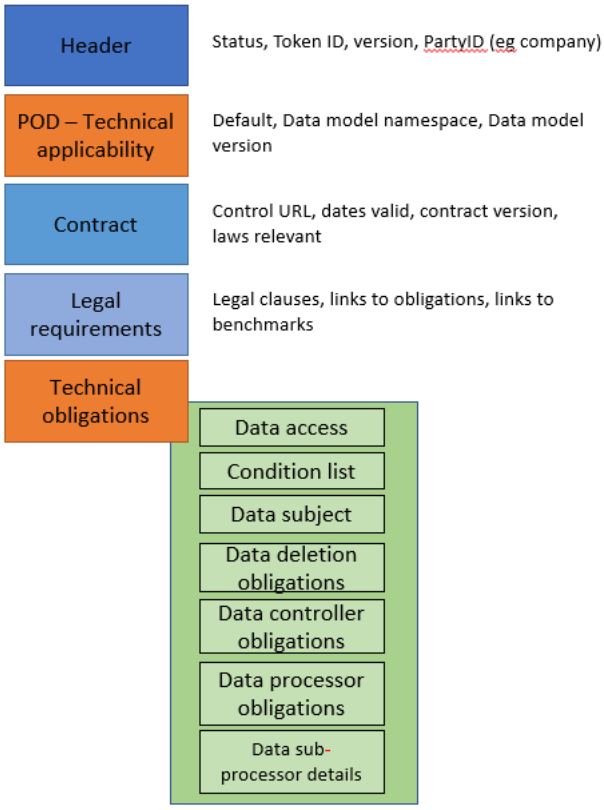
Request	Direct Architecture	Brokered Architecture
Query	M	M
Create	P	P
Update	P	P
Delete	P	P
Events	P	P

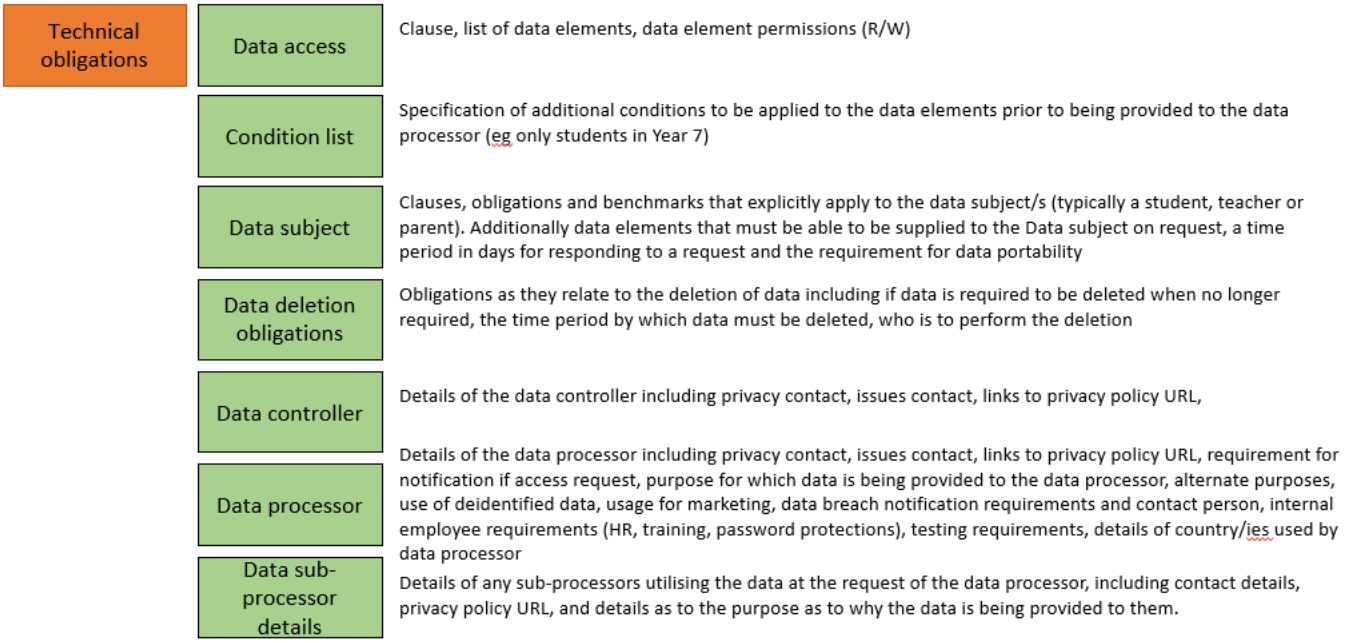
M = Mandatory, O = Optional, P = Prohibited.

8. POD Structure

PODs are a digital representation of a paper contract and also include useful technical details to assist in the application of data set filters, conditions etc.

At a high level, the structure of a POD can be broken down as follows:





Sample POD content:

```
  <xhtml:Example xmlns="" name="pod">
    <pod id="D3E34B39-9D79-4019-8C39-00AA001A1659">
    <podStatus>Live</podStatus>
      <!-- podStatus - The status value for this POD. -->
    <podToken>1</podToken>
      <!-- podToken - The token reference for the POD. -->
    <podVersion>1.0.2</podVersion>
      <!-- podVersion - The POD version number. PODs are expected to change over time. -->
    <partyId>Compass</partyId>
      <!-- partyId - The name of the party (typically the data processor) which will receive the data from the school, district
or state and has entered into an agreement to process the data and respect the privacy obligations of the data -->
    <privacyList>

      <privacy>
        <default>Y</default>
        <dataModelNamespace>http://www.sifinfo.org/infrastructure/2.x</dataModelNamespace>
        <dataModelVersionMin>2.4</dataModelVersionMin>
        <dataModelVersionMax>2.8</dataModelVersionMax>
        <!-- Privacy Obligations - holds information about the contract -->
        <privacyObligationsDocument>
          <contract>
            <contractURI>http://www.vic.priv.contract.edu.au/contract.pdf</contractURI>
            <contractName>Sample contract ABC</contractName>
            <dateValidFrom>2018-03-01</dateValidFrom>
            <dateValidTo>2018-06-01</dateValidTo>
            <contractVersion>0.2</contractVersion>
            <lawList>
              <law>
                <lawName>APP1</lawName>
                <lawDescription>XXXXX</lawDescription>
              </law>
              <law>
                <lawName>APP2</lawName>
                <lawDescription>yyyy</lawDescription>
              </law>
            </lawList>
            <studentDataIPRights>Victorian Department of Education</studentDataIPRights>
          </contract>
          <legalRequirements>
                      <clauseList>
              <clause>
```

```
                <clauseLabel>Data Access</clauseLabel>
                <clauseReference>US-I.3</clauseReference>
                <clauseContent>The agent shall access only data necessary to accomplish the business task for which it is
contracted.</clauseContent>
                <obligationList>
                                <obligation>
                                        <obligationDescription>Restrict access to data fields to match the state
profile.</obligationDescription>

<obligationURL>https://thewave.sde.ok.gov/thewave/Portals/0/2%20x%20Wave%20Requirements_v1%2010.pdf</obligationURL>
                                        <benchmarkList>
                                             <benchmark>
                                                    <benchmarkName>Oklahoma State Profile</benchmarkName>
                                </benchmark>
                                        </benchmarkList>
                                </obligation>
                </obligationList>
              </clause>
            </clauseList>
          </legalRequirements>
          <technicalRequirements>
            <dataAccess>
              <clauseList>
                <clause>
                    <clauseReference>US-I.3</clauseReference>
                </clause>
              </clauseList>
              <fieldList>
                <field>
                  <fieldName>/SchoolInfo/@RefId</fieldName>
                  <controlrights>R</controlrights>
                </field>
                <field>
                  <fieldName>/SchoolInfo/LocalId</fieldName>
                  <controlrights>R</controlrights>
                </field>
                <field>
                  <fieldName>/SchoolInfo/ACARAId</fieldName>
                  <controlrights>RW</controlrights>
                </field>
                <field>
                  <fieldName>/StudentPersonal/@RefId</fieldName>
                  <controlrights>RW</controlrights>
                </field>
```

```
                <field>
                  <fieldName>/StudentPersonal/LocalId</fieldName>
                  <controlrights>RW</controlrights>
                </field>
                <field>
                  <fieldName>/StudentPersonal/FirstAUSchoolEnrollment</fieldName>
                  <controlrights>RW</controlrights>
                </field>
              </fieldList>
            </dataAccess>
             <conditionList>
              <condition>
                <typeOfCondition>XQUERYTEMPLATE</typeOfCondition>
                <conditionName>RetrieveByGrade</conditionName>
                <conditionDescription>Returns a StudentJoin of StudentPersonal and StudentSchoolEnrollment for the GradeLevel
specified as a parameter.</conditionDescription>
               </condition>
               <condition>
                <typeOfCondition>contextId</typeOfCondition>
                <conditionName>CurrentEnrollment</conditionName>
                <conditionDescription>Only works with student data whose records indicate they are currently enrolled according
to the StudentSchoolEnrollment SIF object.</conditionDescription>
                <propertyList>
                  <property>
                    <propertyName>Enrolled After</propertyName>
                    <propertyValue>/StudentSchoolEnrollment/EntryDate&gt;=2018-09-01</propertyValue>
                  </property>
                  <property>
                    <propertyName>Enrolled Before</propertyName>
                    <propertyValue>/StudentSchoolEnrollment/EntryDate&lt;=2019-09-01</propertyValue>
                  </property>
                </propertyList>
              </condition>
            </conditionList>
                              <!-- Who the information is about. Typically a Student and/or a Parent and/or a Teacher. -->
            <dataSubject>
              <clauseList>
                <clause>
                              <clauseReference>US-Subject1</clauseReference>
                              <obligationList>
                                <obligation>
                                      <obligationDescription>FERPA requires records be delivered within 45 days of a
request.</obligationDescription>
```

```
<obligationURL>https://www2.ed.gov/policy/gen/guid/fpco/brochures/parents.html</obligationURL>
                                        <benchmarkList>
                                            <benchmark>
                                              <benchmarkName>Family Educational Rights and Privacy Act
(FERPA)</benchmarkName>

                                            </benchmark>
                                        </benchmarkList>
                                    </obligation>
                  </obligationList>
                </clause>
              </clauseList>

              <fieldList>

                <!-- information that must be supplied to the Data Subject if request as per Clause above. -->
                <field>
                                <fieldName>/StudentPersonal/@RefId</fieldName>
                            </field>
                            <field>
                                <fieldName>/StudentPersonal/LocalId</fieldName>
                            </field>
                            <field>
                                <fieldName>/StudentPersonal/FirstAUSchoolEnrollment</fieldName>
                </field>
              </fieldList>
              <respondInDays>45</respondInDays>
              <requirePortability>Y</requirePortability>
            </dataSubject>
                              <!-- obligations relating to the deletion of data. -->
            <dataDeletion>
              <deleteData>Y</deleteData>
              <dataRetention>30</dataRetention>
              <deleteBy>2019-07-15</deleteBy>
              <partyToDeleteData>
                <organisation>Local LEA</organisation>
                <deletecontactInfo>
                  <name>
                    <familyName>EEe</familyName>
                    <givenName>Agg</givenName>
                    <otherGivenNames>hh</otherGivenNames>
                  </name>
                  <positionTitle>Data Owner</positionTitle>
```

```
                <emailList>
                  <email>DataSec@localLEA.edu.au</email>
                </emailList>
                <phoneNumberList>
                  <phoneNumber>
                    <number>(03) 8888-9999</number>
                  </phoneNumber>
                </phoneNumberList>
              </deletecontactInfo>
            </partyToDeleteData>
          </dataDeletion>
          <!-- Aligned to the SDPC work for consideration. -->
                <securityTechnologyList>
                      <securityTechnology>
                              <clauseList>
                                    <clause>
                                          <clauseReference>US-V.1.c</clauseReference>
                                    </clause>
                              </clauseList>
                              <!-- See Security Protocol clause above. -->
                              <technologyName>TLS</technologyName>
                              <versionMin>1.2</versionMin>
                      </securityTechnology>
                </securityTechnologyList>
        </technicalRequirements>
        <dataController>
          <!-- Those who are in charge of storing and controlling the information on behalf of parents. -->
          <dataControllerName>Victorian Department of Education</dataControllerName>
          <privacyPolicyURL>http://www.vic.priv.policy.edu.au/privacy.pdf
                </privacyPolicyURL>
          <privacyContact>
             <!--                Simplify                these!                                More                like:
http://specification.sifassociation.org/Implementation/NA/3.5/Collections/xStaffs.xhtml -->
              <name>
                <familyName>Smith</familyName>
                <givenName>Christime</givenName>
                <otherGivenNames>Margaret</otherGivenNames>
              </name>
              <positionTitle>Admin</positionTitle>
              <emailList>
                <email>chris@whoville.vic.edu.au</email>
              </emailList>
              <phoneNumberList>
                <phoneNumber>
```

```
            <number>(03) 9600-0102</number>
          </phoneNumber>
        </phoneNumberList>
    </privacyContact>
    <issuesNotificationContact>
        <name>
          <familyName>BBBB</familyName>
          <givenName>AAA</givenName>
          <otherGivenNames>CCC</otherGivenNames>
        </name>
        <positionTitle>Data Security</positionTitle>
        <emailList>
          <email>DataSec@whoville.edu.au</email>
        </emailList>
        <phoneNumberList>
          <phoneNumber>
            <number>(03) 8888-9999</number>
          </phoneNumber>
        </phoneNumberList>
    </issuesNotificationContact>

</dataController>
<dataProcessor>
    <!-- Data processor is the vendor who will be accessing, consuming and/or providing back information. -->
    <dataProcessorName>CEOProcessor2</dataProcessorName>
    <privacyPolicyURL>www.ceo.priv.policy.edu.au</privacyPolicyURL>
    <privacyContact>
        <name>
          <familyName>Woodall</familyName>
          <givenName>Charles</givenName>
          <otherGivenNames>William</otherGivenNames>
        </name>
        <positionTitle>Admin</positionTitle>
        <emailList>
          <email>drseuss@whoville.k12.ceo.edu.au</email>
        </emailList>
        <phoneNumberList>
          <phoneNumber>
            <number>(08) 8555-0102</number>
          </phoneNumber>
        </phoneNumberList>
    </privacyContact>
    <notifyDataControllerOnAccessRequests>Y</notifyDataControllerOnAccessRequests>
    <dataProcessorContactForAccessRequests>
```

```
      <name>
        <familyName>XXX</familyName>
        <givenName>YYY</givenName>
        <otherGivenNames>ZZZ</otherGivenNames>
      </name>
      <positionTitle>Privacy Officer</positionTitle>
      <emailList>
        <email>off@whoville.au</email>
      </emailList>
      <phoneNumberList>
        <phoneNumber>
          <number>(03) 9999-9999</number>
        </phoneNumber>
      </phoneNumberList>
    </dataProcessorContactForAccessRequests>
    <purposeList>
      <purpose>Admin System</purpose>
      <purpose>Australian Schools List</purpose>
    </purposeList>
    <alternatePurposeList>
      <alternatePurpose>Student Portal</alternatePurpose>
    </alternatePurposeList>
    <deidentifiedPurposeList>
      <deidentifiedPurpose>Research</deidentifiedPurpose>
    </deidentifiedPurposeList>
    <dataUsageMarketingAllowed>N</dataUsageMarketingAllowed>
    <personalInformationUpdatedFromSource>10</personalInformationUpdatedFromSource>
    <dataBreachNotification>Y</dataBreachNotification>
    <dataBreachContact>
      <name>
        <familyName>XXX</familyName>
        <givenName>YYY</givenName>
        <otherGivenNames>ZZZ</otherGivenNames>
      </name>
      <positionTitle>Privacy Officer</positionTitle>
      <emailList>
        <email>off@whoville.au</email>
      </emailList>
      <phoneNumberList>
        <phoneNumber>
          <number>(03) 9999-9999</number>
        </phoneNumber>
      </phoneNumberList>
    </dataBreachContact>
```



```
            <employeesMustComplyWithAgreement>Y</employeesMustComplyWithAgreement>
            <employeeConfidentialityAgreement>Y</employeeConfidentialityAgreement>
            <employeeTrainingList>
              <employeeTraining>
                <trainingName/>
                <trainingURL/>
              </employeeTraining>
            </employeeTrainingList>
            <passwordEmployeeAccessStandard>
              <standardName/>
              <standardURL/>
            </passwordEmployeeAccessStandard>
            <securityTestRequiredList>
              <securityTestRequired>
                <testType>Pen Test</testType>
                <testFrequency>Annual</testFrequency>
                <remediationRequiredIn>7 days</remediationRequiredIn>
                <shareResults>Y</shareResults>
              </securityTestRequired>
              <securityTestRequired>
                <testType>Vulnerability Test</testType>
                <testFrequency>Monthly</testFrequency>
                <remediationRequiredIn>24hours</remediationRequiredIn>
                <shareResults>Y</shareResults>
              </securityTestRequired>
            </securityTestRequiredList>
            <countryImpactedList>
              <countryImpacted>
                <country>1101</country>
                <usage>Stored/Processed/Exposed</usage>
              </countryImpacted>
              <countryImpacted>
                <country>1102</country>
                <usage>Exposed</usage>
              </countryImpacted>
            </countryImpactedList>
          </dataProcessor>
          <dataSubProcessorList>
            <!-- List of entities that provide services to the Data Processor.  They also handle the data and must be aware of
privacy obligations. e.g. Google, iCloud, Microsoft Azure, AWS etc. -->
            <dataSubProcessor>
              <subProcessorName>One</subProcessorName>
              <!-- Caps? -->
              <privacyPolicyURL>www.one.edu.au</privacyPolicyURL>
```

```
    <privacyContact>
        <name>
          <familyName>First</familyName>
          <givenName>Charles</givenName>
          <otherGivenNames>William</otherGivenNames>
        </name>
        <positionTitle>Admin</positionTitle>
        <emailList>
          <email>drseuss@whoville.k12.ceo.edu.au</email>
        </emailList>
        <phoneNumberList>
          <phoneNumber>
            <number>(08) 8555-0102</number>
          </phoneNumber>
        </phoneNumberList>
    </privacyContact>
    <purposeList>
      <purpose>Attendance</purpose>
    </purposeList>
  </dataSubProcessor>
  <dataSubProcessor>
    <subProcessorName>Two</subProcessorName>
    <privacyPolicyURL>www.two.edu.au</privacyPolicyURL>
    <privacyContact>
        <name>
          <familyName>Two</familyName>
          <givenName>Charles</givenName>
          <otherGivenNames>William</otherGivenNames>
        </name>
        <positionTitle>Admin</positionTitle>
        <emailList>
          <email>drseuss@whoville.k12.ceo.edu.au</email>
        </emailList>
        <phoneNumberList>
          <phoneNumber>
            <number>(08) 8555-0102</number>
          </phoneNumber>
        </phoneNumberList>
    </privacyContact>
    <purposeList>
      <purpose>TimeTabling</purpose>
    </purposeList>
  </dataSubProcessor>
  <dataSubProcessor>
```

```
            <subProcessorName>Three</subProcessorName>
            <privacyPolicyURL>www.three.edu.au</privacyPolicyURL>
            <privacyContact>
                <name>
                  <familyName>Third</familyName>
                  <givenName>Charles</givenName>
                  <otherGivenNames>William</otherGivenNames>
                </name>
                <positionTitle>Admin</positionTitle>
                <emailList>
                  <email>drseuss@whoville.k12.ceo.edu.au</email>
                </emailList>
                <phoneNumberList>
                  <phoneNumber>
                    <number>(08) 8555-0102</number>
                  </phoneNumber>
                </phoneNumberList>
            </privacyContact>
            <purposeList>
              <purpose>StudentPortal</purpose>
            </purposeList>
          </dataSubProcessor>
      </dataSubProcessorList>
    </privacyObligationsDocument>
  </privacy>
  <privacy>
         <default>N</default>
    <appIDList>
      <appID>TT12</appID>
    </appIDList>
    <adapterFingerprintList>
      <adapterFingerprint>b4ef12ce-7025-11e8-adc0-fa7ae01bbebc</adapterFingerprint>
    </adapterFingerprintList>
    <zoneContextList>
      <zoneContext>
        <zoneId>4001</zoneId>
        <contextId/>
      </zoneContext>
    </zoneContextList>
    <!-- List of rest endpoints -->
    <endpointList/>
    <!-- User Groups? -->
    <!-- Users? -->
    <privacyObligationsDocument>
```

```
                    <technicalRequirements>
                     <dataAccess>
                      <!-- Since a POD represents a contract, only these should very.  As such the replacement logic, should
apply to each immediate child found here.  -->
                      <!-- holds the green list and the permissions or conditions on these elements. -->
                      <fieldList>
                           <field>
                             <fieldName>/SchoolInfo/@RefId</fieldName>
                             <controlrights>R</controlrights>
                           </field>
                           <field>
                             <fieldName>/SchoolInfo/LocalId</fieldName>
                             <controlrights>R</controlrights>
                           </field>
                           <field>
                             <fieldName>/SchoolInfo/ACARAId</fieldName>
                             <controlrights>R</controlrights>
                           </field>
                           <field>
                             <fieldName>/StudentPersonal/@RefId</fieldName>
                             <controlrights>R</controlrights>
                           </field>
                           <field>
                             <fieldName>/StudentPersonal/LocalId</fieldName>
                             <controlrights>R</controlrights>
                           </field>
                           <field>
                             <fieldName>/StudentPersonal/FirstAUSchoolEnrollment</fieldName>
                             <controlrights>R</controlrights>
                           </field>
                           <field>
                             <fieldName>/TimeTable/*</fieldName>
                             <controlrights>RW</controlrights>
                           </field>
                      </fieldList>
                     </dataAccess>
                    </technicalRequirements>
      </privacyObligationsDocument>
    </privacy>
    <privacy>
      <default>N</default>
      <adapterFingerprintList>
                    <adapterFingerprint>b4ef12ce-7025-11e8-adc0-fa7ae01bbebc</adapterFingerprint>
          </adapterFingerprintList>
```

```
            <zoneContextList>
                  <zoneContext>
                        <zoneId>4002</zoneId>
                        <contextId/>
                  </zoneContext>
            </zoneContextList>
        <dataModelNamespace>http://www.sifinfo.org/infrastructure/2.x</dataModelNamespace>
        <dataModelVersionMin>2.3</dataModelVersionMin>
        <dataModelVersionMax>2.9</dataModelVersionMax>
      </privacy>
      <privacy>
        <default>N</default>
            <adapterFingerprintList>
                  <adapterFingerprint>b4ef12ce-7025-11e8-adc0-fa7ae01bbebc</adapterFingerprint>
            </adapterFingerprintList>
            <zoneContextList>
                <zoneContext>
                        <zoneId>4003</zoneId>
                        <contextId/>
                  </zoneContext>
            </zoneContextList>
        <dataModelNamespace>http://www.sifinfo.org/infrastructure/2.x</dataModelNamespace>
        <dataModelVersionMin>2.1</dataModelVersionMin>
        <dataModelVersionMax>2.8</dataModelVersionMax>
        <privacyObligationsDocument>
                  <technicalRequirements>
                        <securityTechnologyList>
                          <securityTechnology>
                              <technologyName>SSL</technologyName>
                              <!-- do we really  need this here? <purpose/>-->
                              <technologyDescription/>
                              <referenceURL/>
                          </securityTechnology>
                        </securityTechnologyList>
                  </technicalRequirements>
        </privacyObligationsDocument>
      </privacy>
    </privacyList>
    </pod>

  </xhtml:Example>
```