Namespace and Versioning

Options for a more flexible PIDX specification

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

Table of Contents

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

1.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

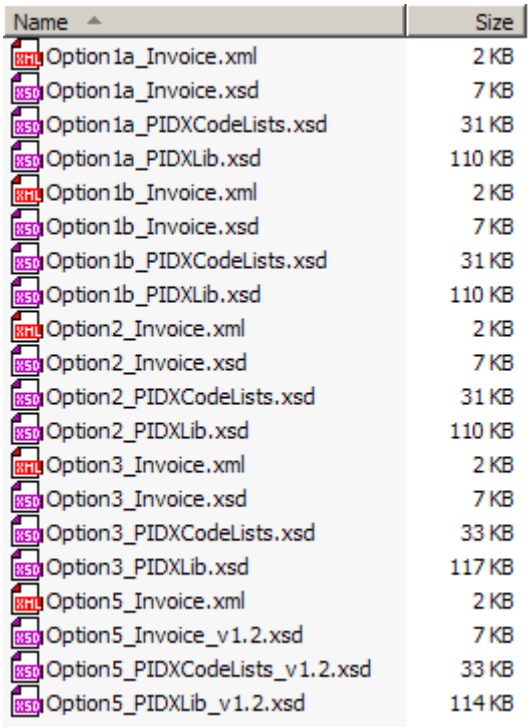
This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2. Introduction

The following options were presented at the PIDX Fall Membership Meeting as options that would support a more flexible method for versioning the PIDX specification as changes are introduced

The document is accompanied by a .zip file (PIDXVersioning.zip) that contains the XML Schema and instance files that reflect the options outlined in this document. The PIDXLib and PIDXCodeLists Schema files have also been included and modified for each option as appropriate. Figure 1 shows the XML files included in the zip file:

Figure 1



This paper recommends that PIDX consider seriously adopting Option 3 ('Use no targetNamespace') for future PIDX specifications. It is, however, critical that PIDX implementers provide feedback since they are the final stakeholders of the PIDX specifications. Absense of a targetNamespace in the common level files will release the PIDX work groups from needlessly impacting Schemas not pertinent to a particular business message.

Note: For brevity, ellipses ('…') are used in certain sections of XML content throughout this document).

Note: The sample XML instance files (included in the PIDXVersioning.zip file) do not contain real-world data.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

Option 1a - Modify the schema's targetNamespace (all cases)

This option is the current mode within which PIDX maintains the 3901 XML specification. For this option, the **v1.2** Invoice schema for would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.PIDX.org/pidXML/v1.2"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:pidx="http://www.PIDX.org/pidXML/v1.2" elementFormDefault="qualified"
attributeFormDefault="unqualified">
    <include schemaLocation=" Option1a_PIDXLib.xsd"/>
    …
</schema>
```

An instance validating against the **v1.2** Invoice schema would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<Invoice xmlns="http://www.PIDX.org/pidXML/v1.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.PIDX.org/pidXML/v1.2 Option1a_Invoice.xsd"
pidx:transactionPurposeIndicator="Add"/>
```

Note that the instance document specifies the target namespace within the root element, so instances are impacted by version changes.

2.1.1

2.1.2 Pros

Schemas must ‘compile’ at design time

Propagated to the application layer

2.1.3 Cons

Minor versions cause big ‘ripple’ effects

Conforming instances are no longer valid

Unchanged schemas are forced to change

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2.2 Option 1b - Modify the schema's targetNamespace (major only)

In this case, PIDX would NOT update the namespace for minor version changes. In the case of major version changes, however, the targetNamespace would need to be modified since backwards compatibility would be compromised and formerly valid instances would no longer validate.

For this option, the Invoice schema header for **v2.0** would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.PIDX.org/pidXML/v2.0"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:pidx="http://www.PIDX.org/pidXML/v2.0" elementFormDefault="qualified"
attributeFormDefault="unqualified">
    <include schemaLocation=" Option1b_PIDXLib.xsd"/>
    …
</schema>
```

An instance validating against the **v2.0** Invoice schema would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<Invoice xmlns="http://www.PIDX.org/pidXML/v2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.PIDX.org/pidXML/v1.2 Option1b_Invoice.xsd"
pidx:transactionPurposeIndicator="Add"/>
```

It's important to note that this option implies that minor version changes would not be reflected in the target namespace. As such, trading partners would have no way of determining which version an instance document conforms to. For example, an instance created to validate against the 3rd rev within the 2.0 version would not be denoted as such unless some mechanism such as the version attribute were to be used (see options 2 and 4). In this case, options 1b and 2 could be combined to create a solution to the versioning problem.

2.2.1 Pros

Previously conforming instances still validate

Schemas must 'compile' at design time

Propagated to the application layer

2.2.2 Cons

Instances need to specify the minor version (not fixed)

Pre-processing of the instance is required

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2.3 Option 2 - Modify the schema version attributed

In this case, PIDX would update the Schema 'version' attribute for both minor and major version changes—instead of updating the target namespace.

For this option, the Invoice Schema header (and the root element) for **v1.2** would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.PIDX.org/pidXML"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:pidx="http://www.PIDX.org/pidXML"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.2">
<include schemaLocation="Option2_PIDXLib.xsd"/>
<element name="Invoice">
    <complexType>
        <sequence>
            <element ref="pidx:InvoiceProperties"/>
            <element ref="pidx:InvoiceDetails"/>
            <element ref="pidx:InvoiceSummary"/>
        </sequence>
        <attribute ref="pidx:transactionPurposeIndicator" use="required"/>
        <attribute ref="pidx:version" use="required"/>
    </complexType>
    …
 </element>
```

Note that the version is captured within the final attribute appearing within the xsd:schema element. Also note that the 'version' attribute on the root element is required, this attribute would be used by processing applications to map instances to the Schemas for which they are valid.

It's important to note that the current PIDX Schemas DO contain these version attributes (both in the schema and the instance). The current schemas, however, denote all versions (major and minor) in the target namespace.

An instance validating against the **v1.2** Invoice schema would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<Invoice xmlns="http://www.PIDX.org/pidXML"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.PIDX.org/pidXML Option3_Invoice.xsd"
transactionPurposeIndicator="Add" version="1.2">
```

This option mandates that all instances specify the Schema version for which they are valid. The instances would specify that by including a 'version' attribute on the root element of the specific business message (e.g., Invoice, OrderCreate, BoL).

If the PIDX were to adopt this option by removing the targetNamespace and relying on the version attributes, special pre-processing by applications would be necessary (see 'Cons' below).

2.3.1 Pros

No impact to instances – Instances valid against a specific version will also validate against other PIDX versions (pending content compatibility).

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

Schema contains version information – The Schema does denote the version, but does so in a ‘loose’ way.

2.3.2 Cons

Validating parsers don’t check this in instances – The instance does not contain any indicator that specifies which Schema version for which it is valid.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2.4 Option 3 - Use no targetNamespace (for common level files only)

With this option, the targetNamespace is removed from the following 2 common level Schema files:

- PIDXLib
- PIDXCodeLists

The targetNamespace is, however, retained for the 'message level' Schema files that make up the specific PIDX business transactions (e.g., Invoice, OrderCreate, BoL). With this design, the common level Schema files are, at runtime, 'coerced' into the namespace as the message level Schemas that use their content.

Absence of a targetNamespace in the common level files will release the PIDX work groups from needlessly impacting Schemas not pertinent to a particular business message.

For this option, the Invoice Schema header for **v1.2** would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.PIDX.org/pidXML/v1.2"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:pidx="http://www.PIDX.org/pidXML/v1.2" elementFormDefault="qualified"
attributeFormDefault="unqualified">
    <include schemaLocation=" Option3_PIDXLib.xsd"/>
    …
</schema>
```

An Invoice instance validating against the **v1.2** Invoice Schema would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<pidx:Invoice xmlns:pidx="http://www.PIDX.org/pidXML/v1.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.PIDX.org/pidXML Option3_Invoice.xsd"
pidx:transactionPurposeIndicator="Add"/>
```

For this option, the PIDXLib Schema header for **v1.2** would appear as follows:

(PIDXCodeLists would also be void of a targetNamespace)

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.2">

…
</schema>
```

Note that the common level Schema files DO include the 'version' attribute at the Schema level. This provides a mechanism to denote clearly the version (major AND minor) of a common level Schema.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2.4.1 Pros

Supports ‘Flattened’ Schemas – With this option, PIDX could create (and publish) flattened Schemas which are a single business message with contents from the common level files (PIDXLib and PIDXCodeLists) contained directly within that file. There are numerous benefits to this approach – from both a PIDX standards development perspective and an implementer’s perspective.

Previously conforming instances still validate

Schemas must ‘compile’ at design time

Propagated to the application layer

Minor versions DO NOT cause big ‘ripple’ effects – Since the namespace within the common level files will not need to change for updates to specific business messages, only the base Schemas affected by a particular change will need to update their namespace.

2.4.2 Cons

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2.5 Option 4 - Create a 'schemaVersion' attribute on the root element

See Option 2 above.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

2.6 Option 5 - Modify the name or location of the schema

This option does not specify the Schema version within the Schema file itself at all. Rather, the Schema name or location would be used to specify the schema.

As such, the **v1.2** Invoice Schema header for this option would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.PIDX.org/pidXML"
xmlns="http://www.w3.org/2001/XMLSchema" xmlns:pidx="http://www.PIDX.org/pidXML"
elementFormDefault="qualified" attributeFormDefault="unqualified">
    <include schemaLocation="Option5_PIDXLib_v1.2.xsd"/>
…
</schema>
```

As such, within PIDX the name of the PIDXLib schema might appear as follows:

- PIDXLib_v1.2.xsd

An Invoice instance validating against the **v1.2** Invoice Schema would appear as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<pidx:Invoice xmlns:pidx="http://www.PIDX.org/pidXML"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.PIDX.org/pidXML PIDXLib_v1.2.xsd"
pidx:transactionPurposeIndicator="Add"/>
```

2.6.1 Pros

2.6.2 Cons

This approach forces instance documents to change, even if the change to the schema would not impact that instance.

All schemas that import the modified schema would need to change since the import statement provides the name and location of the imported schema.

Not propagated to the application layer -- Applications have no indication that Schema content has changed.

The ‘schemaLocation’ attribute in the instance document is optional and is not authoritative even when present. It is a hint to help the processor to locate the schema. As such, exclusive use of this attribute is not good practice.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust

3. Recommendation

Based on the options outlines in this paper, I believe PIDX should consider seriously adoption Option 3 as the method by which the 3901 Schema set is maintained.

If the group decides that 'flattened' schemas would be beneficial, we may want to consider changing the base PIDX messages (e.g., Invoice, CreateOrder, BoL) at some point so that their default namespace is the PIDX namespace ('pidx:') as opposed to the xml schema namespace ('xs:'). This would be required to accommodate a flattened design. PIDX stakeholders should consider the impact this would have on current systems.

© PIDX, Inc. 2011
Use of this copyrighted material is subject to the PIDX End User License Agreement available at www.pidx.org/license.
Each user agrees to such End User License Agreement by making any use of the copyrighted material.

This document was prepared and is maintained in accordance with the PIDX Procedures for Standards Development,
a copy of which is available at www.pidx.org/procedures, and the PIDX Antitrust Guidelines, a copy of which is available at www.pidx.org/antitrust