The Role of XML Data Validation in e-Regulatory Systems

Derek **Millar**

Abstract

Data validation in regulatory environments may start with an XML schema, but it rarely stops there. There are many levels of validation involved. How can this be effectively implemented? What tools, design approaches can assist?

Table of Contents

1. Introduction

XML and SGML are successful data exchange formats. There are enormous amounts of XML/SGML data being exchanged both manually and automatically amongst organizations and systems. This data can be easily processed, but what about the quality of the data? More specifically, is the data valid, not just in an XML context, but in the context of specific business processes?

Defining what constitutes a valid set of data is difficult to do in practice. The formal definition of "valid XML" is clear enough, however this level of validity only provides basic value in a business process context. There are many levels of "validity" that must be considered when implementing systems that rely on producing or receiving "valid" data sets as part of an information supply chain.

One type of data exchange where validation is particularly important and complex is the Regulatory Submission. Regulatory submissions are made by organizations operating in a regulated industry to the regulator of that industry, usually a government agency or department, in order to remain licensed to operate in that industry. The pharmaceutical, pesticide manufacturing, energy, telecommunications and financial industries are examples of regulated industries. Regulatory submissions have traditionally been delivered on (often skids of) paper.

Regulatory submissions are complex for a number of reasons. They are usually comprised of many different types of documents, containing different types of data, often in multiple formats. They often contain proprietary and confidential business information. The submission may have legal and evidentiary status. Finally, the submission must be accepted as valid by the regulatory body in order for it to be reviewed and a decision rendered. In many industries, such as the pharmaceutical industry, delays in the submission review process can translate to substantial loss of revenue. Ensuring that a submission is valid is a critical function of any system that prepares and publishes regulatory submissions.

The movement towards electronic filing of regulatory submissions is well underway, and the industries with forethought and vision are using XML as the exchange format for their electronic submissions. The use of XML schemas to specify the structure and content of a submission provides great opportunity for ensuring validity of submissions, but it is only the beginning. Guidance documents and directives specify the business rules that further constrain what constitutes a valid submission. However, these documents are not easily translated into machine executable tests and they are often incomplete or contain implicit rules or constraints, rather than explicit ones. Once the rules are identified, a decision must be made on the appropriate technology for implementing the rule checking.

These features of a regulatory submission, coupled with the related business processes, imply a number of levels at which a regulatory submission must be "validated". This paper describes these different levels of validation and reviews how standards such as ISO/IEC 19757: Document Schema Definition Languages (DSDL) can help in addressing this challenge.

2. The Regulatory Filing Life-Cycle

From the perspective of the data that is involved in a filing, the regulatory process consists of five stages (Figure 1).

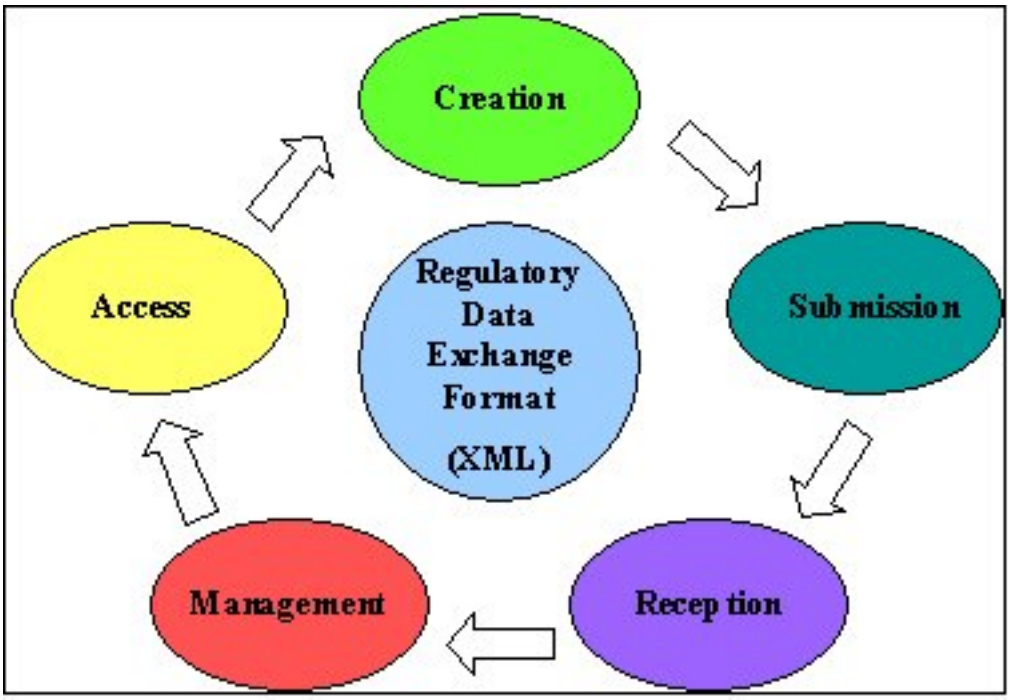

Figure 1. The Regulatory Filing Life-Cycle

Creation involves the processes related to preparing all the necessary information that is required for a regulatory submission. A variety of documents and forms are usually involved.

Submission involves assembling all the information into a package that can be transmitted to the regulatory body. This preparation usually involves a step where an authorized agent signs the data package to attest to its accuracy and completeness.

Reception is a task for the regulatory body and usually involves a screening of the submission to ensure completeness and compliance with the requirements.

Management addresses issues related to storing and cataloguing the submission data in order to support the evaluation process and meet the legal and policy requirements of the regulator.

Finally, **Access** allows the information to be viewed and used in a timely manner in support of a variety of tasks including evaluation, analysis, reporting and responding to requests for information from the public.

2.1. Potential Benefits of Effective Validation in Electronic Regulatory Systems

In each stage of the Regulatory Filing Life-Cycle, there are opportunities to enhance the quality of the data, enable automation, streamline review processes and improve access to the data through the application of validation. A critical requirement for effective validation, of course, is to have the data in a machine readable, open, data format. The use of XML as the Data Exchange Format is currently the best way to meet this requirement.

2.2. Document Types in a Regulatory Submission

A regulatory submission may consist of many documents of many different types, including fill-out forms, covering letters, reports or studies, transcripts, summaries, even raw data sets. The use of XML to capture these document types can also vary greatly. In one industry, a submission may be represented in XML as an electronic index, with the actual documents remaining in a proprietary format such as PDF or MS Word that are referenced from the XML "backbone". In another industry, the content of the "documents" may be entirely captured in XML. In both cases, a submission may consist of multiple XML instances, external entities (e.g. images or documents represented as "blobs"), a meta-data component, most likely also in XML, as well as things like digital signatures or checksums to support data integrity and authenticity.

Validating such a diverse set of information requires a variety of tools and a decomposition of the general problem into a series of steps.

3. Validation Criteria

In a regulatory submission, a data set is prepared and transmitted from industry stakeholder to regulatory authority. Prior to submission, the submitter must validate the data set to ensure that it meets the regulator's criteria for acceptance. Upon reception, the regulator must verify that their criteria are indeed met. Ideally, the systems and processes used to prepare a submission should be able to perform the same validation checks that the receiving systems use to validate a submission on the regulator's end.

The Regulator ultimately defines what constitutes a valid submission. This may be a formal definition, or it may be ad hoc, established during pre-submission consultations or during initial screening. Sometimes the determination criteria are subjective, dependent on the nature of the submission.

In order to automate (even partially) the determination of validity, the acceptance criteria must be clearly and explicitly specified. Ideally, they should consist of:

- An XML schema that defines the structure of documents that comprise a submission.
- A specification of the "data requirements" that describes the business rules and conditions that the content of the XML instance(s) must satisfy in order to be accepted by the regulator.
- A physical packaging specification that describes how the various files that comprise a submission must be organized into directory structures or file archives (e.g. zip files), adhere to file and directory naming conventions and comply with media specifications (e.g. CD-ROM, DVD, etc.) to support transmission.

These criteria can be considered in terms of different levels of validation against which a data set may be evaluated.

4. Levels of Validity

In the case where a submission consists of multiple documents, the validity of the entire submission is a function of the validity of each document that is contained in the submission. Presumably, the Data Requirements Specification and the physical packaging specification identify whether certain documents must be present and how they can be identified. There must be a defined starting point for processing the submission. In decomposing the task of validating a submission, the first step is to determine whether all of the required information is present. Once this is established, each document can be considered individually.

We consider two broad types of validation - *Basic* which covers XML validation, and *Advanced* which forges deeper into the actual content of a submission and increases the complexity of validation.

4.1. Basic Validation

4.1.1. Level 0 - Existence

This is a boot-strapping level. If a file doesn't exist, or is not readable, then there is a fundamental problem. Things such as virus checking or checksum verification could also be included at this level, to ensure that the file(s) comprising the submission are not corrupt. If the data has been digitally signed, then digital signature verification may also fit here. Failure of any of these tests would render the submission invalid and preclude any further validation.

4.1.2. Level 1 - Well-Formedness

This level verifies that the input file is well-formed as per the W3C XML Recommendation. While it is conceivable that an exchange standard may only require well-formedness, this is clearly (I hope!) insufficient on which to base automation of submission and reception processes. One reasonable use for validating to Level 1 is during data creation or data migration.

4.1.3. Level 2 - XML Validity

This level checks that an instance is valid against an XML schema, as per the W3C XML Recommendation. This validation level provides an opportunity to ensure structural and to a certain degree, content characteristics. Depending on what type of schema is used (DTD, W3C Schema, RELAX NG) and the nature of the XML instances ("backbone" or complete documents), this level may provide significant coverage of the acceptance criteria.

4.2. Advanced Validation

4.2.1. Level 3 - Valid Against a Set of Business Rules

These rules specify features of the data that can be checked in the context of XML-based processing. The rules in this level can be further classified:

- **Data typing** - If a W3C Schema or RELAX NG schema is used, this can be addressed during Basic Validation. Examples of this include, date checking, numeric types, ranges, enumerated values, attributes matching specific patterns, etc. Sometimes it may be more effective to leave the data type checking to another level of validation rather than enforce it via the schema.

- **Tagging Guidelines** - These types of rules clarify how the schema is to be applied from a structural and content identification perspective. They are particularly important when the schema is very loose and there are many possible ways to encode information. Insufficient tagging guidelines will result in inconsistently marked up documents, and in the worst case, "tag abuse", which undermines most of the advantages of using XML.

- **Content Rules** - These are the true "business rules", which may be imposed at the "standards level", the "organizational level" and the "document specific" level.

  "Standards level" rules are specified by the regulator or standards body that has published the data exchange format. These rules may constrain the content of specific elements and attributes that are not enforced by the schema, or may specify dependency constraints, such "if the version number is greater than 1, then the revision element must be present and contain at least one revHistory element".

  "Organizational level" rules further constrain the data to meet a specific organization's internal business processes. For example, the organizationId must be the specific organization's identifier, or the productId must match the pattern used by the specific organization.

"Document specific" rules may be required if the schema supports multiple document types, one of which may imply constraints on the structure and content that are specific to that particular document type or subtype.

4.2.2. Level 4 - Valid Against Advanced Business Rules

Advanced, in this context means "beyond XML". These rules are generally not testable using only XML-based technologies. Evaluating these rules may require access to system resources, for example, to verify compliance to packaging specifications, or access to non-XML resources, such as verifying MD5 file signatures or digital signatures. There may be a time-dependent aspect to the rule and require access to external repositories to look up information, such as "is the contact person the one currently identified in the contact management database".

External entity checking is included here. How far do you go in validating externally referenced entities? Is it sufficient to confirm that file exists? Is a file extension check enough? Or is it necessary to invoke, for example, a JPEG or CGM conformance testing utility? What about PDF? This is an area where trade-offs are quickly made. It also raises a shortcoming of regulatory exchange formats that use XML as a "backbone" format. When all of the real content of a submission is in less accessible formats, there is less opportunity to evaluate the quality of the data.

Link validation also fits into this level. Especially if there are references between documents within the same submission, or to documents that have been previously submitted and are part of a "public record".

5. Implementing Basic Validation - Schema Design Considerations

A good way to establish clear acceptance criteria and support effective validation is to adopt a well-designed XML schema.

5.1. Selecting a Schema Language

The most appropriate schema language to use depends on the nature of the data. If the data is document centric, there is little need for data typing and the instances won't need to include other standard schemas (e.g. SVG, MathML, etc.), then a DTD schema may be sufficient. DTDs are simple to write and read. Most document-oriented XML editors (e.g. Epic, XMetaL, FrameMaker) work best with DTDs. Examples of regulatory exchange formats specified by DTDs include: ATA 2200i, the Ontario Energy Board's ERF, ICH's eCTD, EMEA's PIM.

If the information is "data centric", with opportunity to specify explicit data typing of attributes, or with a requirement to include other standard schemas or to use namespaces, then a W3C Schema is more appropriate. W3C Schema also supports an object-oriented approach to data modeling, with abstract classes and user defined data types. Examples include: HL7 CDA and SPL, OECD's Pesticide Registration Template format, etc.

RELAX NG, a part of ISO/IEC 19757 DSDL provides somewhat of the best of both worlds. It uses an XML syntax, so you can use namespaces and include other schemas, and you can use the W3C data typing facilities from a RELAX NG schema. There is currently less support for this schema from tools, but this is quickly changing.

5.2. What Type of Schema is Appropriate?

A schema may be "generic" or "semantic". In a generic model, the schema elements are generic and attribute values identify the type of information (e.g. <field name="City">Toronto</field>). In a "semantic" schema, the elements are named based upon the business information or objects they are intended to capture or represent (e.g. <City>Toronto</City>).

Generic schemas provide almost no opportunity for constraining structure and little support for schema-based validation. On the other hand, a generic schema can easily be extended to accommodate new requirements of the model. Semantic schemas can provide much more structural constraints, but can be more complicated to extend and evolve.

Another factor that influences Basic validation is whether the content models are "loose" or "tight". If all the elements are optional and repeatable, then there is not a lot that can be enforced. The same issue applies to optional vs. required attributes.

Deciding upon a type of schema represents a trade off between flexibility and predictability. Your schema can't be too rigid because the user community is very diverse and business requirements will inevitably change. On the other hand, the whole point of standardizing the data format is so that submissions will be more consistent and predictable in structure and content.

6. Advanced Validation - Encoding and Evaluating Business Rules

In order to test business rules, they must be specified clearly and unambiguously. This is easier said than done, and is easiest done if the language in which they are specified is machine readable. In the context of business rules that can be evaluated through XML processing, a strong candidate for this language is Schematron (ISO/IEC 19757 Part 3: Rule based validation). Schematron provides a standard language for stating assertions about the content of XML documents, which can then be tested against an XML instance by a Schematron processor, which produces a report in XML format, containing the results of the tests.

Schematron is a powerful technology because:

- It provides a way to capture business rules in a machine readable format, that can also be read reasonably well by people
- It provides a the results of validation in XML, making it easy to further process for either generating a formal report or for feeding into a follow-on process
- It is an ISO Standard, so Schematron rules files will be transportable across processor implementations
- It is an ISO Standard, so the specification is open and available to the public
- It reduces the need for encoding business rules in the validation engine - the rules are an input to a process that evaluates the rules.
- The rules are easier to change because they are not encoded in the validation engine

Multiple Schematron rule sets can be implemented to address the different levels of business rule validation. In the context of a regulatory system, the regulator could provide their reference set of rules to industry for use during submission preparation. Organizations could implement their own organization specific rules that could be incorporated into their validation stage in addition to those of the regulator.

For non-XML related validation rules, a validation engine must be able to accommodate some form of "plug-in" rules checkers, either as "dll" or "jar" files that implement specific rules. A validation engine could have an advanced rule interface that these plug-in rules checkers would need to implement.

7. Validation Management

With the problem of validation effectively decomposed into multiple levels with different characteristics, the challenge is to manage the evaluation of all the different levels. Validation Management is within the scope of ISO/IEC 19757 Part 10, which is currently under development.

Pipeline approaches, from simple batch files invoking a chain of processes, to more sophisticated technologies such as Orbeon's XML Pipelining Language (XPL) or Apache's Cocoon framework, can control the flow of data through multiple processes, which could be an XML validation or Schematron processes. A key consideration for pipelining technologies is whether they are linear or asynchronous, i.e. must step B wait until Step A is completed before starting?

Software build tools such as **make** and **Ant** could also provide a framework for validation management. Advantages to these tools include their ability to manage dependencies, while supporting multiple output "targets" which could map to different validation levels.

The ability to integrate custom components and utilities that perform specialized business rule checking is another key consideration for validation management. In the context of real world processes such as a regulatory submission, not all the relevant business rules can be checked through XML processing, so the validation management framework must accommodate this.

Yet another important issue for a validation management framework or engine is how to consolidate the reports that are generated from all of the various processes that may be involved in a complex validation.

Web Services can certainly play an effective role in implementing a validation component of a larger system. In a service oriented world, a regulator could expose their validation engine to their stakeholder community via a web service. However, there may still be a need for a command line interface to a validation engine that can produce a text file containing a report of the results of validation. At a minimum, the regulator could make their business rules, encoded in a standard, open format such as Schematron, available to their stakeholders. A validation management framework should enable both the web service and the command line utility to use the same set of encoded business rules.

8. Conclusion

Validation is a critical component of an Electronic Regulatory System. The use of XML as a regulatory data exchange format enables thorough and extensive automated validation. The ability to implement such validation is dependent on a well designed schema for the regulatory documents and the explicit specification of machine readable validation rules. ISO/IEC 19757, Document Schema Definition Language (DSDL) provides a number of proven, standard techniques for encoding XML validation rules which regulators could use to specify a significant subset of validation criteria for their data sets. Grouping sets of rules into different "levels" of validation can help in planning a validation strategy within the context of larger system. Finally, a validation engine must provide validation management in order to co-ordinate a complex validation process. "Pipelining" technologies, as well as build tools such as **make** and **Ant** provide candidate architectures for implementing validation management.

Biography

Derek **Millar**
Director, Professional Services
Newbook Production Inc. [http://www.newbook.com]
Mississauga
Ontario
Canada

Derek is Director of Professional Services at Newbook Production Inc., an information management and consulting company that develops Technical Publishing Solutions, Electronic Regulatory Solutions and Legal Support Solutions for customers in the public and private sector. He has over ten years of experience providing quality solutions to customers with sophisticated information management and publishing requirements. Industries he has worked with include aerospace, commercial and legal publishing, energy, pharmaceutical and pesticides. He has extensive experience in business analysis, gathering and documenting system requirements and preparing technical specifications.

His current areas of interest include the application of Topic Maps to negotiation and decision support systems and in education, and the use of DSDL standards to address the business requirements related to validating electronic regulatory submissions. Derek is currently the Chair of the Canadian Advisory Committee to ISO/IEC JTC 1 SC34, the ISO subcommittee responsible for SGML and related standards.