

# Suggestions from Data Commons, July 2013

Compilation of notes from Len Armstrong regarding edits/problems with existing schemas

## Technique

- a. I created two global attribute groups. One titled NonLeafElementAttributes and other titled LeafElementAttributes.
- b. Since all non-leaf nodes are based on a defined complex type, I updated the complex types to include the NonLeafElementAttributes group.
- c. New complex type elements that are extensions of the simple types with the attributes added. In general, I used the convention `XyzElement` for extending a simple type named `Xyz`. The simple types remain as they were before but are now effectively "paired" up with a new complex type IF the simple type was used as the base type of an element. In other words, for those previously-defined simple types that were only applied to attributes there is no associated complex type.

A few of the elements BoardActionType, BoardNoticeType, BoardOrderAppealType already have attribute "id" (for a different intent) which is not of type xsd:ID. We have two options here:

- a. Rename the existing "id" attribute to keep the attribute "id" isolated for the common purpose of a document-wide unique element identifier. This, of course, would break the schema for backward compatibility. I'm not sure what the organization's position on this is.
- b. Use an alternate phrase (e.g., "uid") for the new ID attribute.

In doing another deep dive into the document I came up with the following questions and observations:

1. There were some elements typed as `xsd:date` which are now the newly extended type DateElementType. However, since you already had a DateAccuracyType defined with an optional accuracy attribute, can we stay consistent and make all dates types as DateAccuracyType?
2. Should StateOrProvince have a complex type defined in address.xsd and type the StateOrProvince element in members.xsd from the type that could be defined in address.xsd? This would promote a common reuse model.
3. There is a CountryCodeType defined, yet the element named CountryCode is typed as `xsd:integer`. Is this an accident or intended. If intended should we remove the ununder CountryCodeType definition?
4. There is a NationalTelephoneNumberType defined but unused. The schema does contain phone number elements but they are types as NonNullStrings. This may make sense to allow for "+" and "-" or "." in phone numbers. I think NationalTelephoneNumberType may have a base type of `xsd:integer`. So, should we remove NationalTelephoneNumberType?

As a quick test of the new schema I generated a sample document from XMLSpy according to the schema prior to my changes and validated through the new schema. This is not a guarantee I got everything perfect but it should imply I didn't screw up anything too badly. J Let me know if this model is OK and, if so I can submit alternates for address, name and the attachment schema.

Issues to still be discussed with the committee:

1. Consistent use of the "id" attribute. Which should take the term and which should change?
2. Allowing many for some of the zero-or-one or exactly one elements since we now could have multiple variations coming from different sources.