**Remarks on Akoma Ntoso 30**

***A. General remarks***

1. The versioning as used in the file names is confusing, sometimes "1.0", sometimes "30". So it is not clear how reliable the relation between the schema and the documentation is.

2. Some element names (not their models) may be confusing for a newcomer who would never

* look for the title of an act within the element "preface" nor
* expect the signature, an important component of an act, under the element "conclusion"
* nor look for the enacting terms of an act in the element "body"
* nor see "formula" in most cases only as a textual fragment with a particular meaning.
* Opinion (in AN the opinion of a judge, not a document type), but opinions could be document types which even play a role in legislative procedures

If a type is explicitly called "hierarchicalStructure" and another one "collectionStructure", is the latter one not supposed to be hierarchical, although it consists nearly of the same elements ("mainbody" instead of "body")?

3. The annotations in the schema are often in an linguistically not finalised status: typing errors, incomplete, missing.

***B. Specific remarks***

1. The element "foreign" is actually an encapsulation for external specifications. The switch of the name space and the "lax" processing of the contents guarantee a lot of flexibility and openness for the integration of external standards. Without an import of those standards, however, a real control with the external specifications is impossible. Editing tools will not support users by guiding them through the models (e.g. MathML)., This difficulty is far spread in the schema, because the type type "anyOtherType" is extended by plenty other definitions.

2. The arguments for the definition of a single/universal root element are not conclusive. The distinction between document types is just transferred to the next level. By definition, any element defined in an XML schema can be used as root for a valid instance; this is also true for DTDs. Finally, the name of a root element does not say anything about the grammar behind.

3. Elements like "book", "tome", "chapter", "article", "paragraph" etc. all refer to the same type definition "hierarchy" which hides the fact that there might exist a certain hierarchical relation between them, which cannot be inverted, but which is allowed by the type. This approach risks that all business logic is transferred to a tool's level.

4. The annotations concerning hierarchical elements are always the same ('this element is a hierarchical container called "NNN" either explicitly or due to the local tradition' [NNN is the name of the element]), thus not explaining anything.

5. The annotation/comment to the element "application" ("The element efficacy is a metadata element specifying the period of the efficacy modification.") actually refers to the preceding element "efficacy".

6. The document types which are foreseen in the context of case law do not contain orders or conclusions of the advocate general.

7. The elements "listIntroduction" and "listWarpUp" contain text which surrounds the totality of the items of a given list. The phenomenon, however, is wider and as such language dependant. It is possible that the two components form a sentence which the list is a sub-part of. A consequence of this perspective is that the components are not part of the list, but the list is part of the sentence.

What about wrapping components which at the same time introduce a new list?

8. Elements like "b" or "i" are deprecated in HTML and replaced by other mark-up. But they continue to be used in AN with the same meaning as formerly in HTML.

9. AN offers a system of generic containers for cases where documents are structured in a special way. As a consequence other models could be stricter thus strengthening the guidance of users.