1 Составляющие docx-файла

Документация по docx формату содержится в специально разработанном стандарте [1]. Docx-файл —- это zip-архив который физически содержит 2 типа файлов:

- xml файлы с расширениями xml и rels;
- медиа файлы (изображения и т.п.).

Логически файл содержит 3 вида элементов:

- Типы (Content Types) —- список типов медиа файлов (например png) встречающихся в документе и типов частей документов (например документ, верхний колонтитул);
- Части (Parts) — отдельные части документа, то есть непосредственно его содержимое (например document.xml, footer1.xml, header1.xml, comments.xml, endnotes.xml), сюда входят как xml документы, так и медиа файлы;
- Связи (Relationships) идентифицируют части документа для ссылок (например связь между разделом документа и колонтитулом), а также тут определены внешние части (например гиперссылки).

Содержимое для анализа содержится в трёх файлах с расширением xml:

- *document.xml* – содержит текст документа и некоторые свойства параграфов (описание метаданных, ссылки на стили и нумерацию);
- *styles.xml* – содержит описание стилей, используемых в документе;
- *numbering.xml* – содержит информацию о типах нумерованных и маркированных списков, используемых в документе, а так же о стилях списков.

Для разбора этих файлов необходимо знать основные понятия, использующиеся в данных файлах и в документации формата.

Язык xml является языком разметки, то есть содержимое документа размечается специальными конструкциями – тегами. Тег оборачивается в угловые скобки и имеет название. Теги вкладываются друг в друга, образуя иерархию. Помимо названия, теги могут иметь атрибуты – названия полей, которым присвоены строковые значения. В дальнейшем понятия «тег» и «атрибут» будут использоваться в этом смысле.

1.1 Тело документа (body)

Тело документа – это основное текстовое содержимое документа, представляющее из себя последовательность параграфов, таблиц, встроенных графиков и прочих элементов. Тело документа располагается в файле document.xml внутри тега body. Для каждого параграфа по отдельности описаны все его свойства, тело документа выполняет лишь роль перечислителя параграфов в определённом порядке. Как было сказано выше, мы ограничиваемся рассмотрением только параграфов документа, исключая из рассмотрения все остальные элементы, которые могут встретиться в документе.

1.2 Параграф (paragraph)

Параграф – это основная структурная единица документа. Документ делится на параграфы переносом строки, набранным в текстовом редакторе. Параграф соответствует тегу p документа.

У параграфа может быть описан набор свойств, которые описывают отдельный блок текста. Выделяются так называемые «прямые свойства», которые применяются к параграфу напрямую и прописаны в document.xml, а так же «косвенные свойства», описанные в стилях из styles.xml и применяемые в силу того, что в параграфе есть ссылка на определенный стиль. Кроме того, в свойствах параграфа (как в прямых, так и в косвенных) может содержаться информация о том, что параграф является элементом списка. Свойства параграфа соответствуют тегу pPr документации (этот тег вложен в тег p).

Теги, соответствующие некоторым свойствам параграфа:

- ind (indentation) – отступ от края страницы, атрибут left содержит значение отступа от левого края страницы в двадцатых пункта (пункт – это 1/72 дюйма);
- js (justification) – выравнивание, значением атрибута val может быть left (по левому краю), right (по правому краю), center (по центру), both (по обоим краям);
- sz (size) – размер шрифта, значение атрибута val содержит размер шрифта в половинах пункта;
- numPr (numbering properties) – индикатор того, что параграф является элементом нумерованного или маркированного списка. Атрибут numId содержит уникальный идентификатор списка, элементом которого является данный параграф, атрибут ilvl содержит уровень вложенности списка (нумерация с нуля);
- pStyle (paragraph style) – стиль параграфа, значением атрибута val является идентификатор стиля из файла styles.xml.

Параграф, в свою очередь, состоит из списка текстовых элементов с общими свойствами (run). Каждый текстовый элемент содержит свой отдельно описанный набор свойств.

1.3 Текстовый элемент (run)

Текстовый элемент (run) – элемент, составляющий текстовый регион с набором общих свойств (работает на уровне символов). Например, пусть одна часть параграфа написана жирным шрифтом, а другая – обычным. Если остальные свойства текста совпадают, параграф будет состоять из двух текстовых элементов. Помимо описания свойств текста, элемент содержит в себе непосредственно сам текст. Текстовый элемент соответствует тегу r.

У текстового элемента, так же как и у параграфа, описывается набор свойств, как прямых, так и косвенных (то есть может быть ссылка на стиль из styles.xml). Свойства текстового элемента соответствуют тегу rPr документации.

Теги, соответствующие некоторым свойствам текстового элемента:

- b (bold) – жирный шрифт, значение атрибута val может быть 0 (или False, false) – шрифт нежирный, либо 1 (или True, true) – шрифт жирный. Если атрибутов нет, но тег присутствует, то шрифт нежирный, если тега нет, то шрифт нежирный;
- i (italic) – курсив, описание атрибутов то же, что и у bold;
- u (underlined) – подчеркивание, значение атрибута val может быть none – шрифт не подчеркнут, либо указан вид подчеркивания (например, double). Если атрибутов нет, но тег присутствует, то шрифт подчеркнутый, если тега нет, то шрифт не подчеркнутый;
- rStyle (run style) – стиль элемента, значением атрибута val является идентификатор стиля из файла styles.xml;
- sz (size) – размер шрифта (аналогичен тому же свойству у параграфа);
- t (text) – текст элемента, отображающийся в документе (текст находится внутри тега);
- caps – представление текста заглавными буквами, значение атрибута val может быть 0 (или False, false) – не отображать текст заглавными буквами, либо 1 (или True, true) – отображать. Если атрибутов нет, но тег присутствует, то отображать текст заглавными буквами, если тега нет, то не отображать;
- tab (табуляция), br (перенос строки), cr (возврат каретки), sym – специальные символы, которые могут встретиться в тексте элемента, у тега sym значением атрибута char является шеснадцатеричный код символа.

Некоторые свойства могут быть описаны как для параграфов, так и для текстовых элементов. В рамках данной работы, общим свойством является размер шрифта. В таком случае, если свойство не описано в свойствах элемента, рассматривается значение этого свойства для параграфа, иначе свойство элемента перекрывает свойство параграфа.

1.4 Стиль (style)

Стиль – основная структурная единица файла styles.xml, соответствующая тегу style. Стиль содержит в себе описание свойств конкретного элемента в зависимости от типа стиля, то есть тип стиля соответствует типу элемента (таблица, параграф, нумерация, символ). Если стиль описывает свойства параграфа, он также может включать в себя индикатор того, что параграф является элементом списка.

В файле styles.xml описывается набор стилей, примененных к различным элементам документа, и отношения наследования между ними. Если один из стилей является наследником другого, то все не перекрытые свойства стиля-предка становятся свойствами стиля-наследника. Каждый стиль имеет уникальный идентификатор и тип, по этой паре значений можно ссылаться на любой стиль документа. Уникальный идентификатор стиля – это значение атрибута styleId тега style, тип стиля – значение атрибута type. Кроме того, если у стиля есть атрибут default, значение которого равно 1, то данный стиль является стилем по умолчанию среди всех стилей данного типа.

Свойства текста для документа определяются в соотвествии с диаграммой, показанной на рис. 1. Таким образом, к тексту сначала применяются настройки документа по умолчанию, затем свойства параграфа, нумерации и текстового элемента, описанные в стилях, и, наконец, изменяются «прямые свойства», описанные непосредственно в document.xml.

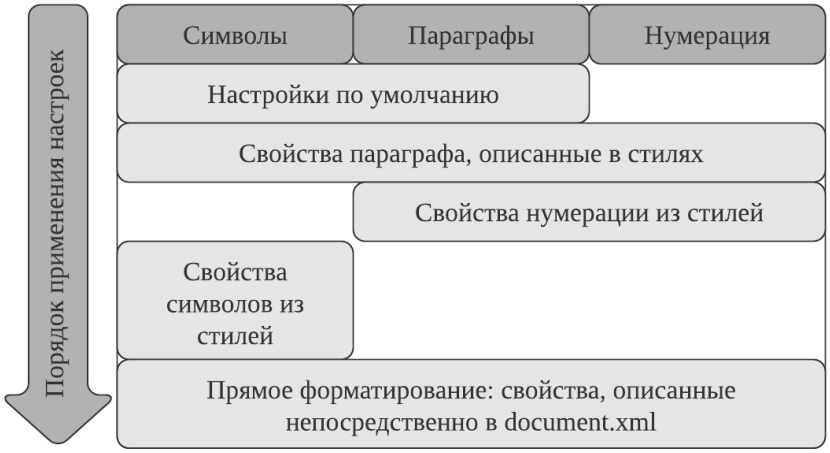

Рис. 1: Порядок примерения настроек свойств

Помимо тегов, описывающих свойства параграфа и текстового элемента (pPr и rPr), в стилях выделяются следующие теги:

- basedOn – наследование стилей (параграфы и символы наследуют свойства параграфов и символов соответственно, нумерация не наследуется), значение атрибута val содержит уникальный идентификатор стиля-родителя;

- docDefaults – настройки стиля по умолчанию (данный тег вложен в тег styles наравне с тегами style и устроен аналогично им).

В стилях параграфов может встречаться индикатор того, что параграф является элементом списка. В таком случае в стиле не будет информации об уровне вложенности списка (нет атрибута ilvl), однако в описании списка в numbering.xml на одном из уровней будет присутствовать ссылка на стиль.

Стили для нумерации содержат лишь ссылку на описание списка в numbering.xml.

1.5 Абстрактная и непосредственная нумерация (abstract numbering and numbering)

В файле numbering.xml содержится информация обо всех типах списков, используемых в документе и их настройках. В документации выделяется два основных понятия: абстрактная нумерация (описание свойств абстрактного списка) и непосредственно нумерация (описание конкретного списка). Абстрактный список описывает свойства списка и может быть общим для нескольких списков, использующих описание свойств.

Абстрактная нумерация (abstract numbering) – описание свойств абстрактного списка (соответствует тегу abstactNum), этому типу присваивается уникальный идентификатор (атрибут abstractNumId), который может использоваться в конкретных реализациях списков. Абстрактная нумерация не может использоваться нигде помимо непосредственной нумерации, наследующей её свойства. Абстрактная нумерация описывает некоторые свойства, общие для всех уровней списка, и свойства, присущие каждому уровню по отдельности. Абстрактные списки могут наследовать свойства других абстрактных списков.

Непосредственная нумерация или нумерация (numbering) – описание конкретных списков документа. Каждый список документа соответствует одному (или нескольким в случае разнородного по стилям списка) списку нумерации (соответствует тегу num). Список нумерации имеет уникальный идентификатор (атрибут numId), с помощью которого в document.xml и styles.xml можно ссылаться на данный список. Кроме того, каждый список нумерации с помощью abstractNumId ссылается на абстрактный список, свойства которого наследуются и могут перегружаться.

Таким образом, файл numbering.xml содержит последовательность из конкретных списков (num), каждый из которых ссылается на абстрактный список (abstractNum) (абстрактные списки так же могут ссылаться на другие абстрактные списки) и может менять некоторые свойства отнаследованного абстрактного списка. В файле document.xml параграфы ссылаются на numId списков напрямую или через стили (непосредственно на списки num, которые наследуются от абстрактных списков abstactNum), тем самым становясь элементами списка. Нумерация списка увеличивается в соответствии с появлением новых элементов одного и того же абстрактного списка, то есть нумерация сохраняется в пределах абстрактного списка до тех пор, пока он не будет прерван другим абстрактным списком (есть исключения, описанные ниже).

Некоторые теги, соответствующие свойствам абстрактного списка (abstractNum):

- numStyleLink – вместо описания свойств абстрактный список может хранить ссылку на другой список, свойства которого будут скопированы в текущий список. Значение атрибута val данного тега равно значению атрибута тега styleLink того списка, свойства которого будут отнаследованы;

- styleLink – индикатор того, что данный абстрактный список описывает свойства, на которые могут ссылаться, атрибут val содержит имя, посредством которого можно ссылаться на эти свойства;

- restartNumberingAfterBreak – это атрибут тега abstractNum для более современных версий приложений, если его значение равно 0, то нумерация не начинается заново, даже если после данного абстрактного списка встречался другой абстрактный список;

- lvl – содержит информацию о свойствах конкретного уровня списка, значение атрибута ilvl равно номеру уровня.

Некоторые теги, соответствующие свойствам конкретного списка (num):

- abstractNumId – значение атрибута val содержит уникальный идентификатор абстрактного списка, свойства которого наследуются;

- lvlOverride – используется для перегрузки свойств некоторых уровней списка. Данный тег содержит список уровней (тегов lvl), каждый из которых перекрывает свойства уровней, описанных в абстрактном списке.

Теги, соответствующие свойствам одного из уровней списка (теги, вложенные в тег lvl):

- lvlText – текстовое представление нумерации списка. В значении атрибута val указана строка специального вида: если в ней встречается символ % с последующей цифрой (цифра означает номер уровня + 1), то в итоговом тексте нумерации вместо % и цифры вставляется текущее значение счётчика нумерации списка для данного уровня;

- numFmt – формат, в котором представляется список на конкретном уровне, значения формата представлены атрибутом val, например, decimal (десятичное число), lowerRoman (римские цифры в нижнем регистре) и т. д.;

- isLgl – если этот тег присутствует, то необходимо игнорировать тег numFmt для всех уровней и использовать нумерацию десятичными цифрами;

- start – начальное значение значение нумерации (для первого элемента списка или для списка, который начался сначала из-за lvlRestart). Значение атрибута val - десятичное число, показывающее номер, с которого начинается отсчет, если тега start нет, то его значение устанавливается в 0;

- lvlRestart – если значение атрибута val равно 0 и предыдущий элемент данного списка находился выше по иерархии, то не начинать нумерацию сначала, иначе нумеровать сначала (в том числе, если тег отсутствует);

- suff – содержимое между текстом нумерации и остальным текстом параграфа, значением атрибута val может быть nothing (пустая строка), space (пробел), tab (табуляция);

- pStyle – в атрибуте val содержит для конкретного уровня уникальный идентификатор стиля из styles.xml (этот стиль содержит индикатор нумерации без указания уровня). Если параграф в document.xml ссылается на этот стиль, то этот параграф считается пронумерованным и уровнем вложенности будет тот уровень, в котором присутствует тег pStyle с указанием данного стиля;

- pPr – свойства пронумерованного параграфа для данного уровня аналогичные свойствам параграфа, описанным выше. Если для пронумерованного параграфа в document.xml прописаны свойства, они перекроют данные свойства нумерации;

- rPr – свойства текстового элемента, которые применяются только к тексту нумерации;

- startOverride – данный тег может присутствовать внутри lvlOverride и отвечает за сбрасывание нумерации того абстрактного списка, наследником которого является список, содержащий этот тег. Значение атрибута val показывает новое начальное значение нумерации.

Список литературы

[1] Office Open XML File Formats — Fundamentals and Markup Language Reference. — 2016.