



Р.Ю. Скорнякова

**Обзор программных средств для
создания HTML-версии журнальной
статьи из исходного материала в
формате Word**

Рекомендуемая форма библиографической ссылки

Скорнякова Р.Ю. Обзор программных средств для создания HTML-версии журнальной статьи из исходного материала в формате Word // Научный сервис в сети Интернет: труды XXV Всероссийской научной конференции (18-21 сентября 2023 г., онлайн). — М.: ИПМ им. М.В.Келдыша, 2023. — С. 332-344.

<https://doi.org/10.20948/abrau-2023-38>

<https://keldysh.ru/abrau/2023/theses/38.pdf>

Видеозапись выступления

Обзор программных средств для создания HTML-версии журнальной статьи из исходного материала в формате Word

Р.Ю. Скорнякова

ИПМ им. М.В. Келдыша РАН

Аннотация. Наряду с традиционной формой электронного представления полных текстов научных статей – форматом PDF – в последние годы получил широкое распространение формат HTML, обладающий для онлайн-публикаций рядом преимуществ за счет имеющихся в нем средств для лучшей структуризации материала, вставки мультимедийного контента и реализации разного рода интерактивных и динамических возможностей. В связи с этим становится весьма актуальной задача получения HTML-версии научной статьи из исходного формата материала, присланного автором. История публикаций полных текстов научных статей в HTML-формате насчитывает около 30 лет, однако единого подхода к подготовке таких публикаций и доступного для всех инструментария за это время не выработано. Технологические цепочки получения HTML-версий статей в разных издательствах могут быть различными. Подход во-многом зависит от кадровых и финансовых возможностей издательства. В работе рассмотрены наиболее популярные из этих подходов и описаны программные средства, применимые в зависимости от того, какой подход выбран. Основное внимание уделено инструментам для работы с исходными текстами в формате Word.

Ключевые слова: HTML-версия журнальной статьи, XML-версия журнальной статьи, стандарт JATS XML, преобразование форматов научных статей

Overview of software tools to create an HTML version of journal article from source material in Word format

R.Y. Skornyakova

Keldysh Institute of Applied Mathematics

Abstract. Along with the traditional form of electronic presentation of full texts of scientific articles - the PDF format - in recent years, the HTML format

has become widespread. HTML has a number of advantages for online publications due to the availability of means for better material structuring, adding multimedia content, and implementing various kinds of interactive and dynamic possibilities. In this regard, the task of obtaining an HTML version of a scientific article from the original format of the material sent by the author becomes very relevant. The history of publishing full texts of scientific articles in HTML format is about 30 years old, but a unified approach to the preparation of such publications and tools accessible to all have not been developed during this time. Workflows of obtaining HTML-versions of articles in different publishers may be different. The approach largely depends on the personnel and financial capabilities of the publishing house. The paper considers the most popular of these approaches and describes the software tools applicable depending on the chosen approach. The main attention is paid to the tools used for source texts in Word format.

Keywords: journal article HTML-version, journal article XML-version, JATS XML, conversion of scholarly article formats

В настоящее время подавляющее большинство научных журналов имеет онлайн-версии и предоставляет полные тексты статей для открытого доступа или на коммерческой основе. Помимо традиционной формы представления полных текстов – формата PDF – многие издательства публикуют полные версии научных статей в HTML-формате. Каждый из этих форматов имеет свои преимущества и недостатки, подробно описанные в работе [1]. Преимущества HTML-формата связаны главным образом с возможностью добавления в него расширенного функционала. Формат PDF более удобен для чтения офлайн. В работе [2] изложены результаты опроса, проведенного в 2013 году издательским домом Elsevier среди читателей журналов на платформе ScienceDirect, о предпочтениях в выборе формата публикации. Большинство опрошенных считает удобным использовать HTML-версию для предварительного просмотра статьи и определения, насколько статья отвечает их интересам, а PDF-версию – для более внимательного чтения. Однако при наличии в HTML-версии динамики и интерактивных возможностей предпочтения могут быть отданы формату HTML. Вывод, сделанный в работах [1, 2], состоит в том, что в ближайшем будущем HTML-формат полностью PDF-формат не заменит, и электронные журналы будут публиковать статьи в обоих форматах. В связи с этим встает вопрос, как организовать процесс получения двух синхронизированных между собой версий научной статьи из исходного материала, присланного автором.

Казалось бы, для материалов в формате Word, это не сложная задача. Редактор Word предоставляет возможность конвертации и в PDF, и в HTML форматы. Однако HTML-версии документов, полученных средствами Word, обладают существенными недостатками. Есть два варианта конвертации: «тяжелый» и «легкий». «Тяжелый» вариант содержит избыточную информацию о форматировании документа,

необходимую только для того, чтобы страница могла быть преобразована обратно в исходный формат. Это делает код сложным и запутанным и затрудняет работу программиста по подключению стилей CSS и добавлению интерактивных и динамических возможностей. Кроме того, созданные файлы имеют очень большой размер, что нежелательно при размещении в Сети. «Легкий» вариант – «фильтрованная» веб-страница – содержит только основную информацию о форматировании, файлы получаются гораздо меньшего размера, однако часть таблиц и все формулы Word преобразует в изображения с низким разрешением – формулы, особенно сложные, на этих изображениях плохо различимы.

Возможность создания HTML-страницы есть и в плагине MathType, часто используемом для ввода математических формул в документ Word. Однако обе опции для формул – преобразовывать их в изображения или конвертировать в формат MathML – не дают желаемых результатов: изображения формул, так же, как и при конвертации средствами Word, имеют низкое разрешение, а конвертация формул в формат MathML не всегда доводится до конца, прерываясь на неопознанной ошибке (из трех документов с формулами нам удалось конвертировать только один).

Можно отметить также, что упомянутые конвертеры не учитывают особенности структуры научной статьи: в результирующем HTML-коде не выделены авторы, аннотация, библиографический список и другие структурные единицы, которые желательно вычленять как для создания удобного для чтения визуального представления, так и для машинной обработки.

Таким образом, встроенные средства Word решают задачу конвертации в HTML научной статьи неудовлетворительно. Для решения задачи получения качественного HTML-кода, который, будучи дополненным кодом JavaScript, создаст HTML-версию статьи, удобную как для предварительного просмотра, так и для внимательного чтения; с интерактивными возможностями; с масштабируемым представлением математических формул, доступным для машинной обработки и поиска, необходимы другие инструменты.

1. XML-представление научной статьи. Подход XML-First

Решение поставленной задачи естественно начинать с создания модели научной статьи. Каждая журнальная статья имеет определенную структуру, как правило, включающую в себя титульную, основную и справочные части, следующие друг за другом. Титульная часть содержит метаданные журнала и метаданные самой статьи, включая аннотацию. Основная часть – это собственно содержание работы, состоящее, как правило, из нескольких разделов и, возможно, подразделов, каждый из которых содержит текст, дополненный рисунками, таблицами, формулами. Справочная часть включает библиографию, благодарности, глоссарий и

т.п. В журнальной статье может присутствовать плавающая часть с отдельными рисунками, таблицами или, например, комментариями рецензентов. Метаданные, библиографический список и другие части научной статьи имеют свою внутреннюю структуру. Например, библиографический список состоит из отдельных библиографических ссылок, в которых могут быть выделены авторы, названия работ, данные журнала, в котором они опубликованы и т.п.

Поскольку большая часть научной статьи представляет собой текст, удобным языком для формального описания подобной структуры является XML, и поэтому вполне естественно, что основным подходом, избранным издательствами при создании HTML-версии научной статьи, стало предварительное создание ее XML-версии. Для получения HTML используют XSLT или какой-либо иной способ автоматического преобразования. Часто XML-версия статьи используется и для автоматического преобразования в PDF, что позволяет получать синхронизированные версии статьи из одного источника. Подход, при котором в основе получения обоих форматов – HTML и PDF – лежит XML, называют **XML-First**.

Предварительное создание XML-версии позволяет представлять все статьи в едином дизайне, который при желании нетрудно изменить. HTML-элементы и атрибуты, полученные при автоматическом преобразовании из XML-элементов, могут быть использованы для организации удобной навигации и реализации интерактивных и динамических возможностей. Например, можно реализовать появление всплывающей подсказки, содержащей полный или частичный текст библиографической ссылки, при наведении курсора мыши на место ссылки внутри статьи. Заголовки разделов и подразделов могут быть использованы для создания меню, обеспечивающего быстрое перемещение между отдельными частями статьи.

Еще одно преимущество такого подхода – в том, что XML-формат отделяет структуру статьи от ее представления и тем самым упрощает хранение и обмен информацией, поиск данных, доступ к ним и управление ими. Современные СУБД предоставляют возможности для хранения данных в XML-формате и быстрого поиска в них. Одну из таких СУБД – MarkLogic Server – использует, например, для хранения статей издательство Nature Publishing Group, выпускающее большое число журналов, в т. ч. журнал Nature.

XML, описывающий структуру научной статьи, можно дополнять элементами и атрибутами, отражающими понятия из конкретных научных областей с использованием определенных словарей и онтологий, продвигаясь тем самым в направлении формального описания научного знания.

2. Стандарт JATS XML

Широкое использование формата XML потребовало выработки единого стандарта для обмена журнальными статьями и хранения их в электронных библиотеках. За основу был взят стандарт NLM DTD, разработанный в Национальном центре биотехнологической информации США (NCBI) Национальной медицинской библиотеки США (NLM) для онлайн-архива биомедицинских статей PubMed. Этот стандарт, выпущенный в 2003 году, стал де-факто стандартом для хранения и обмена открытыми научными публикациями. Модель стандарта NLM DTD базировалась на реальной издательской практике. Были проанализированы XML-схемы более 40 издательств и сотен журналов, которые, как выяснилось, в значительной степени совпадали. Совпадающие у 80% схем элементы и атрибуты включались в модель, независимо от того, считали ли разработчики стандарта, что можно сделать лучше, остальные 20% выбирались на усмотрение разработчиков [3]. Хотя изначально стандарт предназначался для хранения и обмена, многие издательства стали использовать его в процессе подготовки публикаций. В дальнейшем совместно с другими организациями были выработаны три модели научной статьи для разных целей: для обмена и хранения, для подготовки публикации и для создания контента. Совокупность этих моделей, получившая название **Journal Article Tag Suite** (сокращенно JATS), в 2012 году была признана официальным стандартом Национальной организацией по стандартизации информации США (NISO). Текущая официальная версия JATS – 1.3, название NISO стандарта – ANSI/NISO Z39.96-2021 [4].

Модели научной статьи, предназначенные для разных целей, во многом совпадают, но есть и отличия. Модель **Journal Archiving and Interchange Tag Set**, предназначенная для хранения контента журнала и обмена журнальными статьями, содержит наибольшее число элементов, при этом правила их использования являются наименее жесткими. В частности, XML-представление статьи может содержать только метаданные. В модели **Journal Publishing Tag Set**, предназначенной для XML-разметки статьи в издательстве для последующего преобразования в выходной формат, меньше элементов, но среди них больше предписанных, и в большей степени регулируется порядок элементов. В модели **Article Authoring Tag Set**, предназначенной для создания контента научной статьи, меньше всего элементов, но правила для них являются самими жесткими.

Корневым элементом во всех наборах является элемент **article** – статья. Он включает в себя элементы-контейнеры:

front – для заголовка и метаданных;

body – для основного содержания статьи;

back – для дополнительной информации, включающей благодарности, библиографию, приложения, глоссарий и т. п. Для элементов, расположенных отдельно, предусмотрен контейнер **floats group**.

Библиографический список оформляется через элемент **ref-list**, библиографическая ссылка – через элемент **ref**, при этом структура отдельной ссылки может описываться двумя способами: с пунктуацией и пробелами – элемент **mixed-citation** – или без – элемент **element-citation**. Есть отдельные элементы для таблиц, формул, рисунков, списков и элементы-контейнеры, включающие таблицы или рисунки с заголовками и описаниями, и т. д. Таким образом, структура научной статьи может быть описана очень детально. При этом далеко не все элементы являются обязательными – степень гранулированности разметки определяется конкретным издательством.

В настоящее время, по утверждению авторов стандарта, он используется для подготовки научных статей к публикации большинством средних и мелких издательств Северной Америки и Европы. Крупные издательства, в которых накоплено большое число XML-документов и налажен основанный на собственных моделях процесс подготовки публикаций, продолжают использовать свои XML-схемы, хотя и среди них есть прецеденты перехода на JATS. Например, крупное издательство Nature Publishing Group/Palgrave Macmillan перевело на JATS XML издательство новых журналов и запланировало перевести старые.

JATS де факто стал международным стандартом. Он используется более чем в 25 странах, в том числе в России.

3. Виды рабочих процессов с использованием JATS XML

Большинство западных издательств так или иначе использует формат JATS XML в своих рабочих процессах. По этапам, на которых он применяется, эти процессы упрощенно делятся на три основные группы с условными названиями **XML-First**, **XML-Middle** и **XML-Last**.

В процессах вида XML-First предполагается ручной ввод статьи в формате XML и затем автоматическое преобразование в выходные форматы. Процессы вида XML-Middle основаны на использовании программ-конвертеров, преобразующих исходный формат статьи, присланной автором (обычно Word или LaTeX), в формат JATS XML. Процессы вида XML-Last предполагают получение формата JATS XML на конечном этапе после формирования выходных форматов с целью обмена журнальными статьями и хранения их в библиотеках.

По мере расширения использования стандарта JATS XML в процессах подготовки публикаций, появились и программные инструменты для оптимизации этих процессов. Среди них инструменты для валидации JATS XML, инструменты для преобразования JATS XML в

форматы HTML и PDF, XML-редакторы с поддержкой JATS, инструменты для конвертации традиционно используемых авторами форматов в формат JATS XML.

Инструменты для валидации JATS XML нужны для всех видов рабочих процессов; инструменты для преобразования в HTML и PDF необходимы для процессов вида XML-First и XML-Middle; XML-редакторы с поддержкой JATS могут быть использованы в процессах XML-First для ввода контента и в процессах XML-Middle для корректировки результата автоматического преобразования; конвертеры исходных форматов в JATS XML необходимы для процессов вида XML-Middle.

4. Инструменты для преобразования JATS XML в HTML и PDF

Одним из наиболее распространенных способов преобразования XML в другие форматы являются XSL-преобразования. В открытом доступе имеются XSLT-файлы для преобразования JATS XML в HTML и PDF, разработанные в Национальном центре биотехнологической информации США (NCBI) – **JATS Preview Stylesheets** [5]. Они предназначены для предварительного просмотра статей и могут служить отправной точкой для дальнейшей адаптации под требования конкретного издательства. Некоторые журналы, например, PeerJ, публикуют свои XSL-преобразования [6].

Команда разработчиков журнала eLife создала средство для просмотра формата JATS XML в браузерах под названием **Lens** [7]. Это средство реализовано с помощью языка JavaScript и является продуктом с открытым исходным кодом. Окно браузера выюер Lens делит на две части, в одной из которых отображается текст, а в другой дополнительный контент: рисунки, таблицы и т.п. Содержимое правой части синхронизируется с левой при нажатии на соответствующую ссылку.

Среди коммерческих продуктов, позволяющих преобразовывать JATS XML в HTML и PDF, наиболее популярна издательская платформа **Typefi** [8], позволяющая производить рендеринг сложных макетов с использованием динамических шаблонов и дизайнерских методов, задействуя для этого Adobe InDesign.

5. XML-редакторы с поддержкой JATS XML

Поддержка JATS XML имеется в коммерческих XML-редакторах **Oxygen XML Editor** [9], **Fonto** [10], **Xeditor** [11]. Эти редакторы могут быть интегрированы также с редактором формул MathType.

Ведутся разработки по созданию специализированных JATS XML-редакторов с открытым исходным кодом. В 2018 году был выпущен редактор **Texture** [12], реализующий стандарт JATS не полностью и

предназначенный не для ввода контента, а для редактирования результата автоматического преобразования в JATS XML документа Word. Планировалось расширить возможности редактора для того, чтобы он мог использоваться при первоначальном вводе контента, но разработка его была прекращена в 2019 году. На смену редактору Texture должен прийти редактор **Libero** [13], первоначально разрабатываемый в журнале eLife, а затем переданный для дальнейшего развития сообществу Coko Foundation. Работа над этим редактором на данный момент не завершена.

6. Конвертеры документов Word в JATS XML

В издательской среде признано, что среди инструментов, предоставляющих возможность конвертировать документы Word в формат JATS XML, наиболее качественные результаты дают решения компании Inera: **eXtyle JATS** и **eXtyle Custom** [14]. Эти продукты, устанавливаемые как плагины к редактору Word, позволяют автоматизировать трудоемкие аспекты процесса производства XML-документов — вычищение, форматирование, редактирование и собственно преобразование в XML. Преобразование документа Word в формат JATS XML в eXtyle основано на использовании предопределенной палитры пользовательских стилей, которым, как правило, соответствуют элементы JATS XML. Поскольку таких «семантических» стилей много, для ускорения процесса разметки для стилей абзаца в плагине eXtyle предусмотрен специальный диалог (Рис. 1).



Рис. 1. Диалог для выбора стиля в Inera eXtyle

Символьные стили, главным образом используемые для структуризации библиографических ссылок, проставляются автоматически

через вызов функции обработки библиографических ссылок. Для математических формул существуют опции преобразования в MathML, в изображение или в их комбинацию.

Использование Inera eXtyles в комбинации с Turefi является наиболее популярным решением для получения выходных форматов научной статьи из формата Word.

Основываясь на двадцатилетнем опыте развития и эксплуатации eXtyles, компания Inera в 2019 году выпустила новый продукт **eXtyles Arc** [15], который, используя технологии искусственного интеллекта, позволяет получать JATS XML из документа Word без предварительной ручной разметки документа стилями, но документ при этом должен удовлетворять определенным ограничениям.

На технологиях искусственного интеллекта основан и продукт **Intelligent Content for Journals** [16], выпущенный в 2017 году компанией Ictect и представляющий собой сервис, с помощью которого можно загрузить документ Word на специализированный сервер и автоматически получить от него разметку документа тегами JATS. Разметку можно просмотреть и отредактировать с помощью плагина Intelligent Content Tools (icTools), который делит окно документа Word на две части: в левой отображается содержимое документа, в правой – иерархическая структура, соответствующая JATS-разметке (Рис. 2). Панели синхронизированы между собой: когда пользователь помещает курсор на конечный элемент в правой панели, в левой соответствующая часть текста выделяется цветом, и, наоборот, если дважды щелкнуть мышью в левой панели, соответствующий элемент справа выделяется жирным шрифтом.

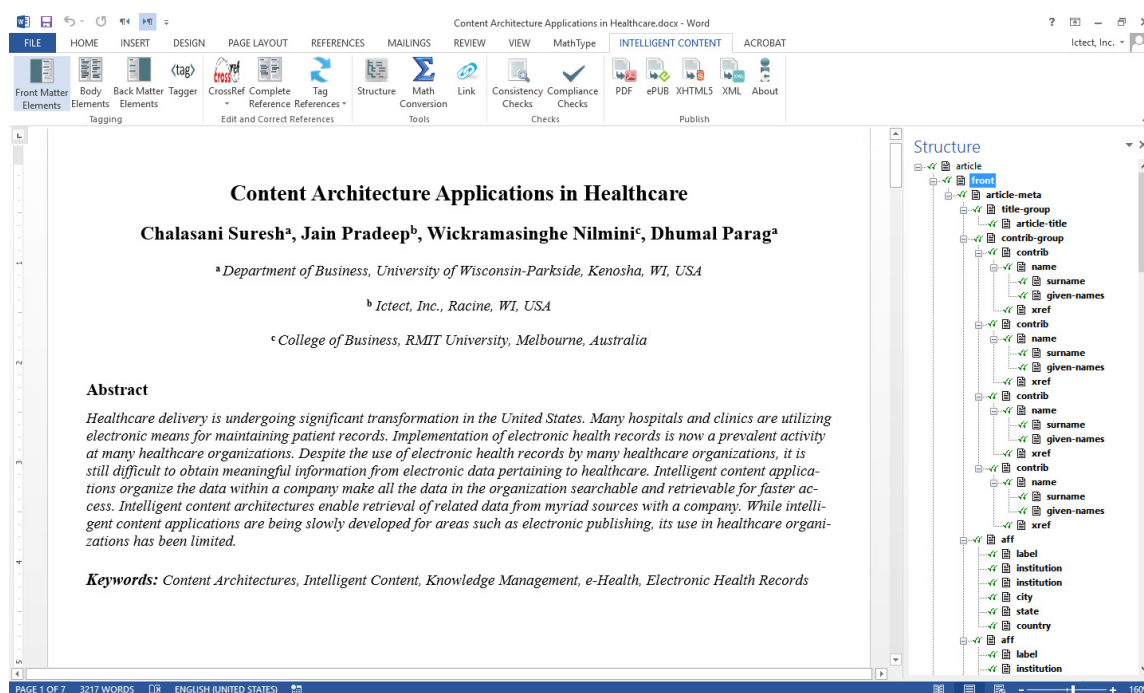


Рис. 2. JATS структура документа Word в плагине icTools

Продукт компании Ictect получил высокую оценку экспертов в области издательских технологий. Тестирование, проведенное крупными издателями, входящими в ассоциацию STM, показало, что этот инструмент создает правильные и детальные структуры JATS из более чем половины исходных рукописей, а более 90% рукописей могут быть усовершенствованы для получения правильного результата менее чем за десять минут обычными редакторами контента, не знакомыми с XML.

Коммерческие продукты компаний Inera и Ictect дают качественные результаты, но они довольно дороги и далеко не все издательства могут себе позволить купить их лицензии. Имеется ряд инструментов с открытым исходным кодом, которые декларируют, что включают возможность конвертации документов Word в формат JATS XML, однако это либо незаконченные разработки, либо конвертация осуществляется с потерями, либо инструмент представляет собой основу, которую надо дополнять пользовательским кодом. Среди этих инструментов можно упомянуть **Pandoc** [17], **Transpect** [18], **meTypeset** [19], **docxToJats** [20].

7. Конвертеры документов Word в HTML

Несмотря на наличие различных программ-конвертеров и накопленный опыт работы с ними, преобразование документов Word в формат JATS XML остается довольно трудоемким и/или финансово затратным. Поэтому рассматриваются и другие подходы. Например, авторы работы [21] предлагают использовать HTML как промежуточный формат при преобразовании формата Word в JATS XML. Семантику, отражающую структуру научной статьи, по их мнению, проще вносить в HTML, чем в документ Word. С этой целью они разработали конвертер с открытым исходным кодом **XSweet** [22], который однако пока не поддерживает изображения и формулы.

Существует множество автоматических конвертеров документов Word в формат HTML, как коммерческих, так и бесплатных, однако это инструменты общего назначения, не учитывающие специфику научных статей. В полученном HTML не будут выделены аннотация, авторы, библиографический список, отдельные элементы библиографического списка и т. п. Результат преобразования надо будет существенно дорабатывать вручную. В особенности это касается статей, содержащих математические формулы. Бесплатные конвертеры (большой, но далеко не полный список таких конвертеров можно найти, например, в обзоре [23]) формулы либо пропускают, либо преобразуют в картинки, что исключает возможность их машинной обработки и существенно ограничивает возможности визуального представления для лучшего восприятия человеком.

Неплохого качества HTML получается при помощи условно бесплатного инструмента **XMLmind Word To XML** [24] французской

компании XMLmind Software, который преобразует формулы в формат svg, дающий достаточно качественное изображение. Коммерческий вариант продукта позволяет управлять результатом преобразования программным путем. Довольно популярен коммерческий инструмент **Doc Converter Pro** [25], позволяющий конвертировать документы Word в различные форматы, в том числе HTML. Пользователь может настраивать преобразование через создание собственных шаблонов, в частности, можно выбрать, в каком виде должны быть представлены формулы: в виде изображений, MathML или текста.

Среди инструментов с открытым исходным кодом, позволяющих преобразовывать документы Word в формат HTML, помимо упомянутого выше XSweet, можно отметить пакеты **Open XML PowerTools** [26] и **Opensagres XDocReport** [27], а также **Mammoth** [28]. На наш взгляд, Mammoth является подходящим инструментом для использования его в качестве основы для написания конвертера научных статей из формата Word в формат HTML. Он проще в использовании, чем Opensagres XDocReport, и, в отличие от Open XML PowerTools, сохраняет только основную информацию о форматировании, а не стремится в точности скопировать внешний вид документа Word. К недостаткам Mammoth можно отнести отсутствие поддержки формул и форматирования таблиц.

Литература

1. Чебуков Д.Е. Об HTML версии полного текста научной статьи // Научный сервис в сети Интернет: труды XXI Всероссийской научной конференции (17–22 сентября 2018 г., г. Новороссийск). — М.: ИПМ им. М.В.Келдыша, 2018. — С. 487–498. — <https://doi.org/10.20948/abrau-2018-16>
2. I.J. Aalbersberg. PDF versus HTML — which do researchers prefer? // Elsevier connect. 9 Jul 2013. — <https://www.elsevier.com/connect/pdf-versus-html-which-do-researchers-prefer>
3. Lapeyre D.A. Introduction to JATS (Journal Article Tag Suite) // XML.com. 12.10.2018. — <https://www.xml.com/articles/2018/10/12/introduction-jats/>
4. Standardized Markup for Journal Articles: Journal Article Tag Suite (JATS) // NISO website, 07.07.2021. — <https://www.niso.org/standards-committees/jats>
5. JATS Preview Stylesheets // GitHub.com — <https://github.com/ncbi/JATSPreviewStylesheets>
6. PeerJ/jats-conversion: Conversion and validation for JATS XML // GitHub.com — <https://github.com/PeerJ/jats-conversion>
7. eLife Lens // GitHub.com — <https://github.com/elifesciences/lens>
8. Typefi: Automated publishing for print, online, and mobile. // Typefi.com — <https://www.typefi.com/products-services/>

9. Oxygen XML Editor. — <https://www.oxygenxml.com/>
10. Finto Editor. — <https://www.fintoxml.com/>
11. XEditor. — <https://www.xpublisher.com/en/xeditor>
12. Texture JATS XML editor // GitHub.com — <https://github.com/substance/texture>
13. Libero Editor // GitHub.com — <https://gitlab.coko.foundation/libero/editor>
14. Inera JATS Solutions. — <https://www.inera.com/jats-solutions/>
15. Inera eXtyle Arc. — <https://www.inera.com/extyle-arc/>
16. Ictect Intelligent Content for Journals. — <https://ictect.com/JATS-XML.php>
17. Pandoc. — <https://pandoc.org/>
18. Transpect. An Open Source framework for converting and checking data. — <https://transpect.github.io/>
19. meTypeset // GitHub.com — <https://github.com/withanage/meTypeset>
20. DocxToJats // GitHub.com — <https://github.com/Vitaliy-1/docxToJats>
21. Piez W. HTML First?: Testing an alternative approach to producing JATS from arbitrary (unconstrained or "wild") .docx (WordML) format Proceedings of Journal Article Tag Suite Conference (JATS-Con) (April 25–26, 2017, Bethesda (MD), USA). — <https://www.ncbi.nlm.nih.gov/books/NBK425546/>
22. XSweet. — <https://xsweet.org/>
23. 13 Best Free Word to HTML Converter Software for Windows. — <https://listoffreeware.com/free-word-to-html-converter-software-windows/>
24. XMLmind Word To XML: Convert DOCX to unstyled, valid, “semantic” XHTML 1.0, 1.1 or 5.0. — https://xmlmind.com/w2x/docx_to_xhtml.html
25. Doc Converter Pro. — <https://docconverter.pro/>
26. Open XML PowerTools. // GitHub.com — <https://github.com/OpenXmlDev/Open-Xml-PowerTools/>
27. Opensagres XDocReport. // GitHub.com — <https://github.com/opensagres/xdocreport>
28. Mammoth. — <https://mike.zwobble.org/projects/mammoth/>

References

1. Chebukov D.E. Ob HTML versii polnogo teksta nauchnoi stati // Nauchnyi servis v seti Internet: trudy XXI Vserossiiskoi nauchnoi konferentsii (17–22 sentiabria 2018 g., g. Novorossiisk). — M.: IPM im. M.V.Keldysa, 2018. — S. 487–498. — <https://doi.org/10.20948/abrau-2018-16>
2. I.J. Aalbersberg. PDF versus HTML — which do researchers prefer? // Elsevier connect. 9 Jul 2013. — <https://www.elsevier.com/connect/pdf-versus-html-which-do-researchers-prefer>
3. Lapeyre D.A. Introduction to JATS (Journal Article Tag Suite) // XML.com. 12.10.2018. — <https://www.xml.com/articles/2018/10/12/introduction-jats/>

4. Standardized Markup for Journal Articles: Journal Article Tag Suite (JATS) // NISO website, 07.07.2021. — <https://www.niso.org/standards-committees/jats>
5. JATS Preview Stylesheets // GitHub.com — <https://github.com/ncbi/JATSPreviewStylesheets>
6. PeerJ/jats-conversion: Conversion and validation for JATS XML // GitHub.com — <https://github.com/PeerJ/jats-conversion>
7. eLife Lens // GitHub.com — <https://github.com/elifesciences/lens>
8. Typefi: Automated publishing for print, online, and mobile. // Typefi.com — <https://www.typefi.com/products-services/>
9. Oxygen XML Editor. — <https://www.oxygenxml.com/>
10. Finto Editor. — <https://www.fintoxml.com/>
11. XEditor. — <https://www.xpublisher.com/en/xeditor>
12. Texture JATS XML editor // GitHub.com — <https://github.com/substance/texture>
13. Libero Editor // GitHub.com — <https://gitlab.coko.foundation/libero/editor>
14. Inera JATS Solutions. — <https://www.inera.com/jats-solutions/>
15. Inera eXtyle Arc. — <https://www.inera.com/extyle-arc/>
16. Ictect Intelligent Content for Journals. — <https://ictect.com/JATS-XML.php>
17. Pandoc. — <https://pandoc.org/>
18. Transpect. An Open Source framework for converting and checking data. — <https://transpect.github.io/>
19. meTypeset // GitHub.com — <https://github.com/withanage/meTypeset>
20. DocxToJats // GitHub.com — <https://github.com/Vitaliy-1/docxToJats>
21. Piez W. HTML First?: Testing an alternative approach to producing JATS from arbitrary (unconstrained or "wild") .docx (WordML) format Proceedings of Journal Article Tag Suite Conference (JATS-Con) (April 25–26, 2017, Bethesda (MD), USA). — <https://www.ncbi.nlm.nih.gov/books/NBK425546/>
22. XSweet. — <https://xsweet.org/>
23. 13 Best Free Word to HTML Converter Software for Windows. — <https://listoffreeware.com/free-word-to-html-converter-software-windows/>
24. XMLmind Word To XML: Convert DOCX to unstyled, valid, “semantic” XHTML 1.0, 1.1 or 5.0. — https://xmlmind.com/w2x/docx_to_xhtml.html
25. Doc Converter Pro. — <https://docconverter.pro/>
26. Open XML PowerTools. // GitHub.com — <https://github.com/OpenXmlDev/Open-Xml-PowerTools/>
27. Opensagres XDocReport. // GitHub.com — <https://github.com/opensagres/xdocreport>
28. Mammoth. — <https://mike.zwobble.org/projects/mammoth/>