

Marcação XML JATS: ferramentas aplicações



Luciano Panepucci
Cofundador e Diretor de Tecnologia

II Encontro Nacional de Portais de Periódicos, Campinas, Brasil, 8 de agosto de 2019

Tópicos

Introdução

XML: Anatomia e sintaxe básica

DTD JATS

Explorando o DTD JATS

Guias de estilo

Ferramentas de validação

Ferramentas para gerar XML

Introdução[1]

O que é XML?

O XML (Extensible Markup Language)[1] é uma linguagem de marcação cujo desenvolvimento começou em meados da década de 1990 pelo consórcio W3C (World Wide Web Consortium)[3] e se tornou uma recomendação oficial em fevereiro de 1998.

As suas principais características são:

- ser **legível por humanos e por máquinas**
- permitir **armazenamento e transporte de dados**
- ser uma **marcação auto-descritiva**

[1] https://www.w3schools.com/xml/xml_what_is.asp

[2] <https://pt.wikipedia.org/wiki/XML>

[3] <https://pt.wikipedia.org/wiki/W3C>

Introdução

O que o XML faz?

Nada!

- armazena dados de forma estruturada,
- utilizando tags e Atributos para definir Elementos.

```
<workshop data="2019-08-08">  
  <titulo>Marcação XML JATS: ferramentas e aplicações</titulo>  
  <autor>Luciano Panepucci</autor>  
</workshop>
```

Introdução

Para que serve?

Arquivos XML servem para armazenar dados de qualquer natureza. No contexto específico da publicação científica, servem para:

- **Armazenar o texto completo** e metadados de artigos em indexadores e arquivos, como: SciELO, PubMed Central.
- **Gerar versões derivadas** para outros suportes como: HTML, ePub, PDF etc.
- **Gerar XML para outros sistemas**: Crossref DOI, ORCID, FundRef, DOAJ etc.
- **Permitir interoperabilidade** entre sistemas: Lattes ↔ Crossref, SciELO ↔ Web of Science etc.
- **Automatizar validações**.

Introdução

O XML ajuda na indexação e internacionalização?

Não!

As revistas são reconhecidas e indexadas quando:

1. **publicam pesquisas de qualidade**, e
2. **praticam boas práticas** editoriais, **adotando recomendações internacionais**.

Introdução

XML é igual a HTML?

Não. Embora ambos usem tags para estruturar documentos:

- o HTML foi desenvolvido para exibir
- no HTML os Elementos são predefinidos
- o XML foi desenvolvido para **armazenar dados**
- no XML os **Elementos são arbitrários** (definidos pelo criador do arquivo)

Introdução

Requisitos para marcar artigos em XML

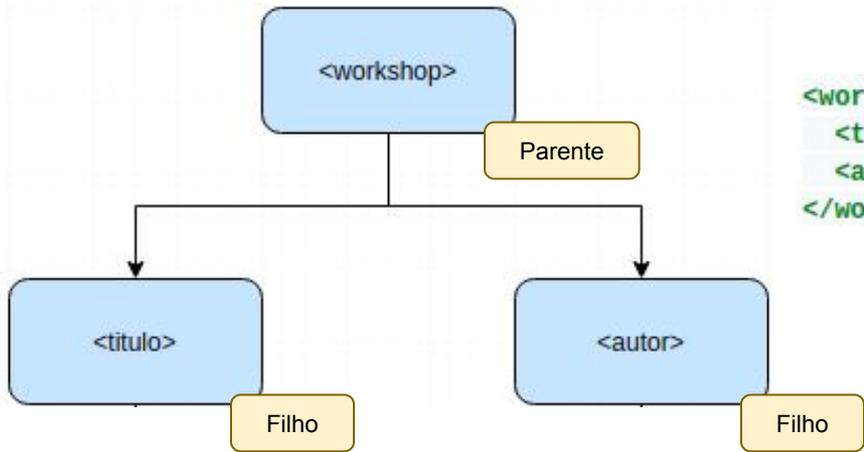
A marcação de artigos científicos usando XML JATS vai exigir que você:

1. Saiba o básico da **anatomia e sintaxe de arquivos XML** genéricos
2. Entenda **o que é um DTD** e saiba navegar pela documentação do DTD JATS
3. Conheça e **siga o guia de estilos** do indexador ou arquivo pretendido (SciELO, PubMed Central, Redalyc)

XML: Anatomia e sintaxe básica

Raiz

- Todo arquivo XML é uma árvore e **exige uma raiz!**
- A raiz é o **parente de todos os outros Elementos.**



```
<workshop data="2019-08-08">  
  <titulo>Marcação XML JATS: ferramentas e aplicações</titulo>  
  <autor>Luciano Panepucci</autor>  
</workshop>
```

XML: Anatomia e sintaxe básica

Prólogo

A linha a seguir é o que se chama de prólogo.

```
<?xml version="1.0" encoding="UTF-8"?>
```

O prólogo é opcional mas, caso exista, deve ser incluído na primeira linha do arquivo XML.

A função do prólogo é indicar:

- a versão (raramente diferente de 1.0), e
- a codificação utilizada para os caracteres.

Para evitar erros é recomendado que você sempre indique a codificação e **grave seus arquivo usando o padrão UTF-8** no seu editor de texto.

XML: Anatomia e sintaxe básica

Comentários

Se precisar documentar seu arquivo XML, por exemplo para investigar problemas, você pode utilizar a sintaxe abaixo para incluir comentários:

- Use `<!--` para iniciar um comentário,
- e `-->` para terminar.

Exemplo:

```
<!-- isto é um comentário -->  
<nome>Pão simples</nome> <!-- isto é outro comentário -->  
<!-- Já isto aqui é um -- comentário inválido -->
```

XML: Anatomia e sintaxe básica

Tag de fechamento

É ilegal omitir uma tag de fechamento.

Todos os elementos devem ter uma tag de fechamento ou serem auto-fechados, assim:

- `<software>Microsoft Office</software>`
- `<versao sistema="windows" edicao="10" /> <!-- isto é uma tag "auto-fechada" -->`

XML: Anatomia e sintaxe básica

Respeite a capitalização

Para o XML `<isto>` não é o mesmo que `<Isto>`.

Você pode usar maiúsculas, minúsculas ou até uma mistura.

O que você **absolutamente não pode fazer** é misturar quando escrever as tags de abertura e fechamento!

Por exemplo:

- `<exemplo>`iniciar um Elemento com uma tag e terminá-lo outra `</Exemplo>`
`<!-- ERRADO -->`

XML: Anatomia e sintaxe básica

Atenção ao aninhamento

Elementos devem estar corretamente aninhados dentro de seus parentes.

- `<peessoa><nome>Ana</nome></peessoa>` `<!-- correto -->`
- `<peessoa><nome>Ana</peessoa></nome>` `<!-- incorreto -->`

XML: Anatomia e sintaxe básica

Caracteres especiais e entidades

Alguns caracteres possuem significado especial no XML.

Por exemplo se você incluir um `<` dentro de um Elemento terá um erro, pois o sistema irá interpretá-lo como início de outro Elemento:

- `<software>Microsoft < Office</software>`

Para evitar esse problema você deve codificar o caractere `<` usando uma Entidade:

- `<software>Microsoft < Office</software>`

Embora apenas “<” e “&” sejam estritamente ilegais, é recomendado como boa prática codificar os 5 caracteres abaixo, quando utilizadas em valores textuais de Elementos ou Atributos:

| | | |
|-------------------------|---|--------------|
| <code>&lt;</code> | < | Menor |
| <code>&gt;</code> | > | Maior |
| <code>&amp;</code> | & | Ê comercial |
| <code>&apos;</code> | ' | Apóstrofo |
| <code>&quot;</code> | " | Aspas duplas |

XML: Anatomia e sintaxe básica

Elementos

Um Elemento é constituído por **tudo que está compreendido entre as tags** (incluindo elas) de abertura e de fechamento:

Exemplos:

- `<software>Markup/software> <!-- Elemento software -->`
- `<peessoa><nome>Ana</nome></peessoa> <!-- Elemento pessoa -->`

Elementos podem conter:

- texto, também chamado de conteúdo tipo CDATA (Character Data),
- Atributos,
- outros Elementos,
- ou uma mistura de tudo isso.

Elementos podem ser vazios, como nos exemplos abaixo:

- `<versao></versao>`
- `<versao />`

XML: Anatomia e sintaxe básica

Um exemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<receita tipo="pão" tempo_de_preparo="5 minutos" tempo_de_cozimento="1 hora">
  <titulo>Pão simples</titulo>
  <ingredientes>
    <ingrediente quantidade="3" unidade="xícaras">Farinha de Trigo</ingrediente>
    <ingrediente quantidade="7" unidade="gramas">Fermento</ingrediente>
    <ingrediente quantidade="1.5" unidade="xícaras"
estado="morna">Água</ingrediente>
    <ingrediente quantidade="1" unidade="colheres de chá">Sal</ingrediente>
  </ingredientes>
  <instrucoes>
    <passo>Misture todos os ingredientes, e dissolva bem</passo>
    <passo>Cubra com um pano e deixe por uma hora em um local morno</passo>
    <passo>Misture novamente, coloque numa bandeja e asse num forno</passo>
  </instrucoes>
</receita>
```

Fonte: <https://pt.wikipedia.org/wiki/XML#Exemplo>

XML: Anatomia e sintaxe básica

Atributos

Os Atributos servem para designar algum valor ou propriedade restrita ao Elemento ao que estão atribuídos.

Atributos devem sempre estar entre aspas simples ou duplas:

- `<receita tipo="pão">`
- `<receita tipo='pão'>`

Caso necessário utilizar uma das aspas no valor do atributo você deve simplesmente usar a outra:

- `<receita tipo="pão d'água">`

Ou então você pode codificar a aspas como uma entidade:

- `<receita tipo='pão d'água'>`

XML: Anatomia e sintaxe básica

Elementos vs Atributos

Como XML tem poucas regras o criador do arquivo pode optar por incluir e estruturar os dados de muitas formas distintas:

- ```
<colaborador tipo="Autor">
 <nome>Luciano Panepucci</nome>
</colaborador>
```
- ```
<colaborador>  
  <tipo>Autor</tipo>  
  <nome>Luciano Panepucci</nome>  
</colaborador>
```

Do ponto de vista das regras de formação do XML, as duas formas acima estão corretas!

XML: Anatomia e sintaxe básica

Exercício 1

O documento XML abaixo está correto?

```
<volume>22</volume>  
<issue>spe 1</issue>  
<fpage>15</fpage>  
<lpage>21</lpage>
```

XML: Anatomia e sintaxe básica

Exercício 1

O documento XML abaixo está correto?

```
<volume>22</volume>  
<issue>spe 1</issue>  
<fpage>15</fpage>  
<lpage>21</lpage>
```

Não!

Todo documento XML deve ter um Elemento Raiz.

XML: Anatomia e sintaxe básica

Exercício 2

O documento XML abaixo está correto?

```
<journal-meta>  
  <journal-id>Autopsy & Case Reports</journal-id>  
  <issn pub-type='ppub'>1414-381X</issn>  
</journal-meta>
```

XML: Anatomia e sintaxe básica

Exercício 2

O documento XML abaixo está correto?

```
<journal-meta>  
  <journal-id>Autopsy & Case Reports</journal-id>  
  <issn pub-type='ppub'>1414-381X</issn>  
</journal-meta>
```

Não!

O caractere & é estritamente proibido. O correto é:

```
<journal-meta>  
  <journal-id>Autopsy &#amp; Case Reports</journal-id>  
  <issn pub-type='ppub'>1414-381X</issn>  
</journal-meta>
```

XML: Anatomia e sintaxe básica

Exercício 3

O documento XML abaixo está correto?

```
<sec sec-type="intro">  
<title>INTRODUCTION</title>  
<p>Climate changes can have impacts on various fields.</sec></p>
```

Não!

O aninhamento está incorreto. Deveria ser:

```
<sec sec-type="intro">  
<title>INTRODUCTION</title>  
<p>Climate changes can have impacts on various fields.</p></sec>
```

XML: Anatomia e sintaxe básica

XML Correto vs Válido

Essas poucas regras que vimos garantem que o arquivo XML seja considerado correto.

Porém, **para ser útil, ele precisa ser Válido!**

Abaixo temos 3 formas corretas! Qual é forma válida?

- `<nome>Luciano Panepucci</nome>`
- `<nome>
 <primeiro_nome>Luciano</primeiro_nome>
 <sobrenome>Panepucci</sobrenome>
</nome>`
- `<colaborador>
 <nome>
 <primeiro_nome>Luciano</primeiro_nome>
 <sobrenome>Panepucci</sobrenome>
 </nome>
</colaborador>`

Para definir a Validade usamos o DTD!

XML: Anatomia e sintaxe básica

DTD

Document Type Definition, ou algo como **Definição do Tipo de Documento**.

O DTD define em detalhes:

- como pode ser a **Estrutura**,
- os **Elementos**, e
- os **Atributos**.

XML: Anatomia e sintaxe básica

Documentos XML válidos

Então, fica assim:

- Um XML com **sintaxe correta** é considerado “**Bem Formado**”, ou “**Well Formed**”.
- Um XML **validado contra um DTD** é “**Bem Formado**” e é “**Válido**”!

Todo documento XML que requer validação deve sempre incluir uma tag especial chamada de **DOCTYPE**, destacada abaixo:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE article PUBLIC "-//NLM//DTD JATS (Z39.96) Journal Publishing DTD v1.0
20120330//EN" "JATS-journalpublishing1.dtd">
<article article-type="research-article">
<front>
<journal-meta>
```

Ah, então JATS é um DTD! Tudo resolvido!

DTD JATS

O que é?

JATS (**Journal Article Tagging Suite**), é exatamente o que o próprio nome diz:

- um **Conjunto de Tags para Marcar Artigos**.

As informações detalhadas estão documentadas em <https://jats.nlm.nih.gov/>, onde estão as 3 bibliotecas (Tag Sets) disponíveis:

- **Journal Archiving and Interchange**:
A versão mais permissiva e com regras relaxadas.
- **Journal Publishing**:
Um conjunto de regras moderadamente restritivas. ← **Esta é a que nos interessa!**
- **Article Authoring**:
A versão mais rígida.

DTD JATS

<https://jats.nlm.nih.gov/>

NCBI



Journal Article Tag Suite

... is an application of NISO Z39.96-2019, which defines a set of XML elements and attributes for tagging journal articles and describes three article models. The content on this site is the supporting documentation for the standard. JATS is a continuation of the NLM Archiving and Interchange DTD work begun in 2002 by NCBI.

About the JATS

[JATS Wiki](#)

[JATS and the NLM DTDs](#)

[FAQ](#)

[NISO z39.96-2019 \(JATS 1.2; Current Standard\)](#)

[Getting the files](#)

JATS-Con

[JATS-Con home](#)

[JATS-Con Proceedings](#)

JATS Extensions

[Book Interchange Tag Suite \(BITS\)](#)

[NISO Standards Tag Suite \(NISO-STTS\)](#)

Tag Sets



[Journal Archiving and Interchange](#)
The most permissive of the Tag Sets.



[Journal Publishing](#)
A moderately prescriptive Tag Set.



[Article Authoring](#)
The most prescriptive of the Tag Sets.

DTD JATS

Motivação

Antes do JATS (ano ~2000):

- Grandes publishers usavam DTD próprio:
 - Elsevier
 - Blackwell
 - Springer
 - Nature
- Dificuldades com **interoperabilidade**
- Dificuldades para **preservação** (arquivamento)
- **XML muito complicado** para pequenos publishers ou revistas

Fonte: Rosenblum B.D. (2017). JATS XML and Related Publishing Standards. *XXV Curso de Editoração Científica*, São Paulo. ABEC disponível em: https://www.abecbrasil.org.br/eventos/xxv_curso/index.asp

DTD JATS

Origem

Início do projeto: 2002

Requisitos:

- PubMed Central
- Arquivo de Harvard (posterior Portico/JSTOR/ITHAKA)
- Financiado pelo NLM (National Library of Medicine) e pela Mellon Foundation.

Fonte: Rosenblum B.D. (2017). JATS XML and Related Publishing Standards. *XXV Curso de Editoração Científica*, São Paulo. ABEC disponível em: https://www.abecbrasil.org.br/eventos/xxv_curso/index.asp

DTD JATS

Objetivos do projeto

Uma biblioteca para marcar o texto completo e os metadados de artigos científicos de qualquer tipo e área:

- Capturar **conteúdo e semântica**, não estilo e visual
- Ser um modelo comum para **arquivamento** e **interoperabilidade**
- Marcação de artigos para:
 - **Metadados** de Artigos, Revistas e Edições
 - **Texto completo**
 - **Referências materiais suplementares**
- Ser **compatível com todas as áreas**

Fonte: Rosenblum B.D. (2017). JATS XML and Related Publishing Standards. *XXV Curso de Editoração Científica*, São Paulo. ABEC disponível em: https://www.abecbrasil.org.br/eventos/xxv_curso/index.asp

DTD JATS

JATS hoje

- Amplamente adotado pela maioria dos grandes publishers
- **Coração de muitos indexadores:**
 - PubMed Central
 - SciELO
 - Redalyc
- Compatível com iniciativas de **boas práticas:**
 - Referências interligadas
 - Indicadores de Acesso e Licença
 - ORCID
 - CRediT - Contributor Roles Taxonomy
 - Identificadores de Instituições e de Financiamento

Explorando o DTD JATS

Acessando a documentação

1. Acesse <https://jats.nlm.nih.gov/> e clique em **Journal Publishing (azul)**
2. Depois, no menu lateral escolha **All Versions** para acessar todas as versões do DTD JATS
 - Para SciELO, o padrão atual exigido é o **NISO JATS Version 1.1**.
 - O PMC aceita outras versões, inclusive mais antigas, para se adequar ao seu processo.

Current Version (Standard):

NISO JATS Version 1.2: <https://jats.nlm.nih.gov/publishing/1.2/>

Previous Versions:

NISO JATS Version 1.2d2 (Draft): <https://jats.nlm.nih.gov/publishing/1.2d2/>

NISO JATS Version 1.2d1 (Draft): <https://jats.nlm.nih.gov/publishing/1.2d1/>

➔ NISO JATS Version 1.1 (2016 Standard): <https://jats.nlm.nih.gov/publishing/1.1/> ➔

NISO JATS Version 1.1d3 (Draft): <https://jats.nlm.nih.gov/publishing/1.1d3/>

NISO JATS Version 1.1d2 (Draft): <https://jats.nlm.nih.gov/publishing/1.1d2/>

NISO JATS Version 1.1d1 (Draft): <https://jats.nlm.nih.gov/publishing/1.1d1/>

NISO JATS Version 1.0 (2012 Standard): <https://jats.nlm.nih.gov/publishing/1.0/>

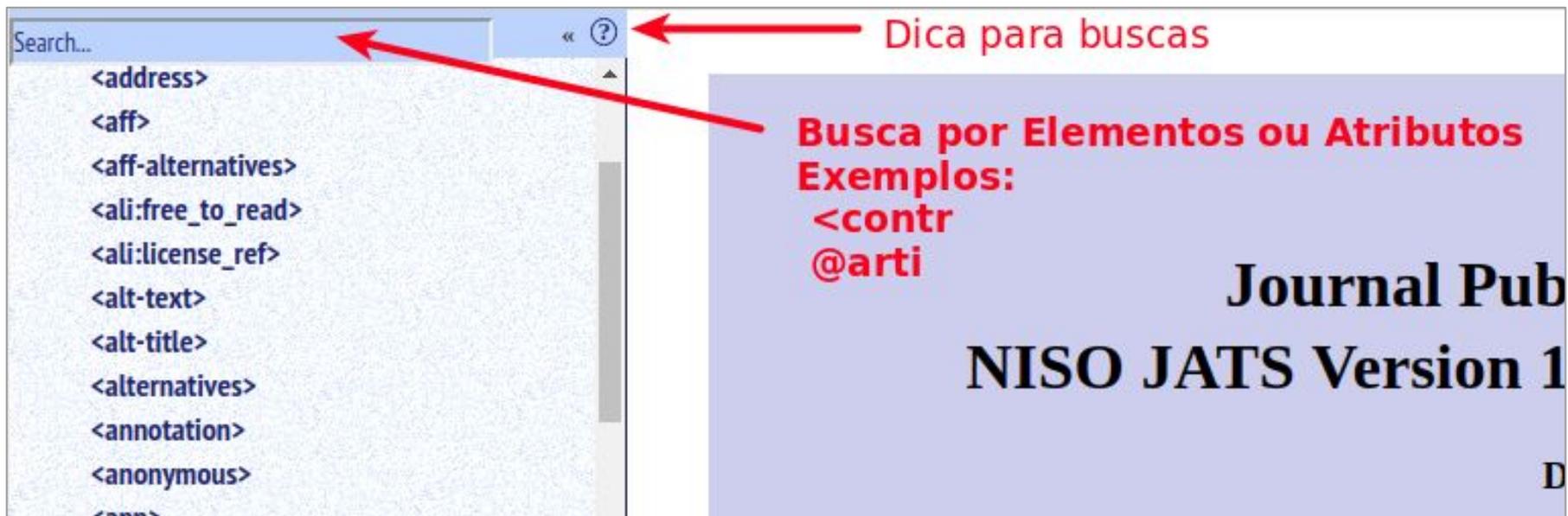
NISO JATS Version 0.4 (Draft): <https://jats.nlm.nih.gov/publishing/0.4/>

Explorando o DTD JATS

Entendendo a documentação do JATS

Os principais conteúdos que você irá usar são:

- Elementos
- Atributos



The image shows a search interface for the JATS DTD. On the left, a search bar is labeled "Search...". Below it, a list of XML elements is displayed, including <address>, <aff>, <aff-alternatives>, <ali:free_to_read>, <ali:license_ref>, <alt-text>, <alt-title>, <alternatives>, <annotation>, and <anonymous>. A red arrow points from the search bar to the text "Busca por Elementos ou Atributos Exemplos: <contr @arti". Another red arrow points from a question mark icon in the top right corner of the search bar to the text "Dica para buscas".

Dica para buscas

Busca por Elementos ou Atributos
Exemplos:
<contr
@arti

Journal Pub
NISO JATS Version 1
D

Explorando o DTD JATS

Elementos

O ponto mais crucial da documentação dos Elementos é entender:

- quais Atributos são possíveis,
- quais Elementos podem ser filhos
- quantas vezes podem aparecer, e
- em qual ordem devem aparecer.

Encontrar essa informação é bem fácil.

Vamos ver alguns exemplos:

- <article>
- <abstract>
- <pub-date>

Guias de estilo

O que são?

São regras sobre como usar as regras.

Os guias de estilo **definem detalhes específicos** que o indexador ou arquivo exige **para um ou mais Elementos**.

Os 2 principais guias são:

- **PubMed Central Tagging Guidelines**

<https://www.ncbi.nlm.nih.gov/pmc/pmcdoc/tagging-guidelines/article/style.html>

- **SciELO Publishing Schema**

https://scielo.readthedocs.io/projects/scielo-publishing-schema/pt_BR/latest/

Guias de estilo

Porque existem?

- O DTD **JATS** permite múltiplas formas de marcar a mesma informação
- **Requisitos específicos**
 - granularidade das afiliações ⇒ estatísticas
 - granularidade nas referências ⇒ bibliometria
 - marcação multi-idioma em legendas de tabelas e figuras
- **Limitações técnicas**
 - falta de flexibilidade para relacionamento Autores x Afiliações
 - falta de flexibilidade na marcação das Referências

No geral SciELO:

- complica algumas marcações para obter estatísticas e métricas, e
- possui menos flexibilidade em outras marcações devido a limitações técnicas, naturais devido a equipe limitada, mas que com o tempo vão sendo vencidas!

Guias de estilo

Exemplos de diferenças

Relação Autores e Afiliações

- **PMC:** Marque as afiliações dentro do <contrib> ou do <contrib-group> dependendo dos relacionamentos. **Aceita 5 formas distintas de marcar!** ([exemplo](#))
- **SciELO:** Marque as afiliações sempre usando <xref>. **Apenas uma forma de marcar.** ([exemplo](#))

Marcação das Referências

- **PMC:** Aceita várias formas usando os 2 estilos possíveis: ([exemplo](#))
 - **<mixed-citation>** preservando toda a pontuação e textos.
 - **<element-citation>** marcando apenas os Elementos importantes.
- **SciELO:** Exige marcar na mesma referência usando os 2 estilos, mas: ([exemplo](#))
 - no **<mixed-citation>** permite apenas a marcação de destaques como **<italic>** etc.
 - **<element-citation>** marcando todos os Elementos.

Guias de estilo

Como se manter atualizado?

PMC:

- Acesse a seção “[Update History](#)” na página do [PubMed Central Tagging Guidelines](#).

SciELO:

- Acesse os tópicos “O que há de novo na SciELO PS...” disponível na página da [SciELO Publishing Schema](#).

Ferramentas de validação

SciELO Style Checker

- **Funções:**
 - Realiza validação (em arquivos individuais) de DTD JATS e de estilo SciELO PS.
 - Gera visualização do HTML para revisão (beta).
- **Plataforma:** Online
- <http://manager.scielo.org/tools/validators/stylechecker/>

Ferramentas de validação

SciELO Package Maker

- **Funções:**
 - Realiza validações (em vários arquivos) de DTD JATS, estilo SciELO PS e validações de conteúdo (prefixo DOI da revista etc).
 - Gera relatório de Erros e Avisos.
 - Gera visualização de tabelas, equações e figuras para revisão.
 - Gera pasta com arquivos finais para entrega.
- **Plataforma:** Windows
- http://docs.scielo.org/projects/scielo-pc-programs/en/latest/pt_installation_configure.html#xml-package-maker

Ferramentas de validação

PMC Style Checker

- **Funções:**
 - Realiza validação (em arquivos individuais) do DTD JATS e de estilo PMC Guidelines.
- **Plataforma:** Online
- <https://www.ncbi.nlm.nih.gov/pmc/tools/stylechecker/>

Ferramentas de validação

PMC Article Previewer

- **Funções:**
 - Realiza validação (em arquivos individuais) do DTD JATS, de estilo PMC Guidelines.
 - Gera visualização HTML do texto completo para revisão.
- **Plataforma:** Online
- <https://www.ncbi.nlm.nih.gov/pmc/tools/article-previewer/>

Ferramentas de validação

Querem ver em ação?

Ferramentas para gerar XML

Markup (SciELO)

Plataforma: Windows

Arquivo de Entrada: Word (.doc) ou (.docx)

Forma de operação: Dentro do Word aplicando tags visíveis diretamente sobre o conteúdo.

Formato de saída:

- arquivo DOC marcado
- arquivo XML

Como obter acesso e ajuda

Informação sobre download e instalação:

- http://docs.scielo.org/projects/scielo-pc-programs/en/latest/pt_installation.html

Tutoriais e material de apoio:

- http://docs.scielo.org/projects/scielo-pc-programs/en/latest/pt_how_to_work_with_xml.html
- https://www.youtube.com/playlist?list=PLQZT93bz3H79NTc-aUFMU_UZgo4VI2iUH
- <https://groups.google.com/forum/#!topic/scielo-discuss/q1JzeVXU3hc> (grupo de apoio)

Ferramentas para gerar XML

Markup (SciELO)

Requisitos para o artigo de entrada

http://docs.scielo.org/projects/scielo-pc-programs/en/latest/pt_how_to_generate_xml-prepara.html

- Pasta com estrutura definida
- Arquivos Word e imagens com nomenclatura específicas
- Arquivo Word com formatação específica

Ferramentas para gerar XML

Marcalyc (Redalyc)

Plataforma: Qualquer (online)

Arquivo de Entrada: HTML

Forma de operação: Tela dividida (formulário e texto)

Formatos de saída:

- HTML,
- PDF Marcalyc
- ePub
- XML JATS4R
- XML SciELO

Como obter acesso e ajuda

Escrever para editores@redalyc.org indicando qual a sua revista indexada e que deseja informações para usar o Marcalyc.

Tutoriais e material de apoio:

- <http://marcalyc.redalyc.org/ayuda/> (os videos são de 2016 mas ilustram bem o uso geral)

Manual em PDF versão 2018:

- http://www.redalyc.org/jatsRepo/materialDidactico/manual_de_usuario_Marcalyc.pdf

Ferramentas para gerar XML

Marcalyc (Redalyc)

Requisitos para o artigo de entrada

<https://youtu.be/LosXmTfAASU?t=206>

Abra o arquivo no Word (ou do InDesing) e **salve como HTML**

- **Sem estilos** e em **uma coluna**
- Em opções web escolha a **codificação UTF-8**
- Salvar com nome **index.html**
- Grave na mesma pasta das imagens
- Compactar a pasta num **arquivo Zip**

Ferramentas para gerar XML

eXtyles (Inera, agora parte da Atypon)

Plataforma: Windows

Arquivo de Entrada: Word (.doc) ou (.docx)

Forma de operação: Dentro do Word **aplicando estilos** de parágrafo e de caractere diretamente sobre o conteúdo.

Formato de saída:

- arquivo DOC estruturado e estilizado
- arquivo XML

Demonstração: <https://vimeo.com/167193412> (minuto 8:30)

Ajuda: <https://www.inera.com/support/documentation>

Contato: <https://www.inera.com/contact>

Ferramentas para gerar XML

Outras

- [PKP OTS – Open Typesetting Stack](#) - pausado
- [typeset.io](#) - interessante
- [texture](#) - interessante
- [Annotum](#) (plugin Wordpress) - descontinuado
- [meTypeset](#) - ??