

Universidade Federal de Minas Gerais  
Instituto de Ciências Exatas  
Departamento de Ciências da Computação

WALISON DIAS DA SILVA

**XML NA LINGUAGEM LOTUSSCRIPT: IMPORTAÇÃO DE DADOS DO LATTES  
PARA O OPUS**

Belo Horizonte  
2012

Universidade Federal de Minas Gerais  
Instituto de Ciências Exatas  
Departamento de Ciências da Computação  
Especialização em Informática: Ênfase: Engenharia de Software

**XML NA LINGUAGEM LOTUSSCRIPT:  
IMPORTAÇÃO DE DADOS DO LATTES PARA O OPUS**

por

WALISON DIAS DA SILVA

Monografia de Final de Curso

Profa. MIRELLA MOURA MORO  
Orientadora

Belo Horizonte  
2012

WALISON DIAS DA SILVA

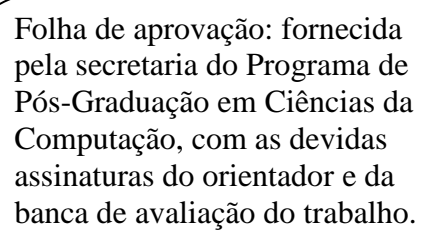
**XML NA LINGUAGEM LOTUSSCRIPT: IMPORTAÇÃO DE DADOS DO LATTES  
PARA O OPUS**

Monografia apresentada ao Curso de Especialização em Informática do Departamento de Ciências Exatas da Universidade Federal de Minas Gerais, como requisito parcial para a obtenção do grau de Especialista em Informática.

Área de concentração: Engenharia de Software

Orientadora: Profa. Mirella Moura Moro

Belo Horizonte  
2012



Folha de aprovação: fornecida pela secretaria do Programa de Pós-Graduação em Ciências da Computação, com as devidas assinaturas do orientador e da banca de avaliação do trabalho.

Ao meu Sr. Jesus Cristo,  
aos incentivos dos meus pais,  
a compreensão da minha esposa e  
a ajuda dos professores,  
dedico este trabalho.

## **AGRADECIMENTOS**

Inicialmente agradeço ao Sr. Jesus Cristo por proporcionar-me condições de vivenciar esse momento.

Agradeço aos meus pais, pelos seus incentivos e esforços na minha educação.

A professora Mirella Moura, pela sua orientação no desenvolvimento deste trabalho.

Aos professores, pelos seus conhecimentos e experiências.

E finalmente, aos colegas por suas ajudas e trocas.

"Uma palavra que não representa uma idéia é uma coisa morta,  
da mesma forma que uma idéia não incorporada em palavras  
não passa de uma sombra."  
Lev Vygotsky

## RESUMO

A LotusScript ou apenas Lotus é uma linguagem de desenvolvimento software de propriedade da IBM no ambiente Domino Notes. Essa linguagem é utilizada pelo Laboratório de Computação Científica da UFMG no desenvolvimento do Sistema Opus e alguns outros aplicativos. O sistema Opus é utilizado para registrar as produções intelectuais da UFMG. Ele permite o lançamento manual por professores, alunos e funcionários de dados sobre uma produção e também o armazenamento da obra integral em formato eletrônico.

Entretanto existem casos em que a produção intelectual é registrada somente no sistema Lattes, que é um sistema de publicação de trabalho acadêmico e técnico externo à UFMG, provido pelo CNPq.

A proposta deste trabalho é utilizar a linguagem XML juntamente com a linguagem de desenvolvimento LotusScript e desenvolver uma importação de dados do sistema Lattes para o Opus, para os casos em que a publicação é feita somente no Lattes, automatizando o processo e diminuindo o tempo do cadastramento de produções.

**Palavras-chave:** XML, LotusScript, Lotus Notes, Lattes, Opus, Importação de Dados.



## ABSTRACT

The LotusScript or simply Lotus is a software development language owned by IBM in Domino Notes environment. This language is used by the Laboratory of Scientific Computing at UFMG in the development of the Opus system and some other applications.

The Opus system is used to record the intellectual productions of UFMG. By using Opus faculty and students can publish relevant information about their works and also store those complete works in electronic format. It permits data about a production to be manually registered by faculties and students and also the storage of complete works in electronic format.

In some cases, however, the work is published only in the Lattes system, another system for publishing academic works, not controlled by UFMG, developed and maintained by CNPq.

The objective of this work is to propose a mechanism for combining the XML and LotusScript languages to importing data from the Lattes system to Opus for the cases in which the work is not published in the latter, automating the process and reducing the time taken to the registration of productions.

**Keywords:** XML, LotusScript, Lotus Notes, Lattes, Opus, Data Import.

## LISTA DE FIGURAS

Figura 1 Produções Bibliográfica Lattes .....	18
Figura 2 Tipos de Produções Bibliograficas do Opus .....	19
Figura 3 Publicação do Tipo Livro no Opus .....	20
Figura 4 Produção Bibliográfica do Lattes.....	21
Figura 5 - Exemplo Documento XML .....	22
Figura 6 Arquivo XML usado no agente LotusScritp .....	23
Figura 7 Dados do Artigo em Periódico Opus .....	25
Figura 8 Dados da Edição do Periódico Opus .....	26
Figura 9 Campos de um artigo completo Lattes.....	27
Figura 10 DTD Curriculo Lattes - Parte Artigos .....	28
Figura 11 Produções XML a serem importadas para o Opus.....	31
Figura 12 Produções importadas no Opus .....	32
Figura 13 Produção importada do Lattes para o Opus .....	33

## **LISTA DE TABELAS**

Tabela 1 Variáveis do artigo do Opus x variáveis do artigo do Lattes.....	31
Tabela 2 Associação variáveis do periódico Opus x variáveis do periódico Lattes.....	32

## LISTA DE SIGLAS

API	Interface de Programação de Aplicativo
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CESAR	Centro de Estudos e Sistemas Avançados do Recife
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
CSS	Cascading Style Sheets
DOI	Digital Object Identifier
DTD	Data Type Definition
ISBN	International Standard Book Number
ISSN	International Standard Serial Number
LMPL	Linguagem de Marcação da Plataforma Lattes
UFBA	Universidade Federal da Bahia
UFMG	Universidade Federal de Minas Gerais
UFRGS	Universidade Federal do Rio Grande do Sul
UFRJ	Universidade Federal do Rio de Janeiro
UFRN	Universidade Federal do Rio Grande do Norte
UFSC	Universidade Federal de Santa Catarina
UNICAMP	Universidade de Campinas
USP	Universidade de São Paulo
UI	User Interface
SAX	API Simples para Xml
SGML	Standard Generalized Markup Language
W3C	World Wide Web Consortium
XML	Extensible Markup Language
XSL	Extensible Stylesheet Language

## SUMÁRIO

1.	INTRODUÇÃO .....	14
1.1.	OBJETIVO .....	15
1.2.	ESTRUTURA DO TRABALHO.....	15
2.	LATTES E OPUS .....	16
2.1.	SISTEMA DE CURRÍCULOS LATTES .....	16
2.2.	SISTEMA OPUS .....	17
2.3.	COMPARAÇÃO ENTRE O LATTES E O OPUS .....	17
3.	XML .....	22
3.1.	UTILIZANDO XML COM LOTUSSCRIP .....	23
4.	ESTUDO DE IMPORTAÇÃO DE DADOS DO LATTES PARA O OPUS.....	25
4.1.	LÓGICA DA IMPORTAÇÃO DOS DADOS.....	28
4.2.	CÓDIGO DE IMPORTAÇÃO DE DADOS DO LATTES PARA O OPUS .....	30
5.	CONCLUSÃO .....	34
	APÊNDICE A .....	36
	APÊNDICE B .....	40
	APÊNDICE C .....	42

## 1. INTRODUÇÃO

Esta monografia tem como assunto núcleo a linguagem de marcação XML que, de acordo com Ricardo José Vaz Tolentino (2004,p.63): “pode ser sintetizada com a representação textual de dados em um conjunto de padrões para o intercâmbio e a publicação de documentos de maneira semi-estruturada.”. A XML é padronizada pelo Consórcio World Wide Web (W3C), as tags dentro do documento são criadas pelo próprio usuário. O documento possui uma estrutura definida pelo uso de regras de aninhamento, conhecida na área da computação científica como hierarquização em árvore.

A linguagem de programação LotusScript é uma linguagem de script que permite desenvolver aplicações dentro dos softwares Lotus Notes da IBM. A maior deficiência do banco de dados Lotus Notes é não ser relacional, o que dificulta bastante o compartilhamento ou relacionamento entre os dados e outros bancos de dados. “Ele usa dois conceitos básicos de dados gravados: os que são vistos graficamente na tela, chamados de UI (User Interface), e os que residem em disco (os documentos e seus itens), recuperados ou não através de uma interface gráfica (os UI Forms)”, (Wikipédia Lotus, 2011). Documentos são os dados armazenados em disco, associados ou não a um Form (virtual ou visível) e os itens são os recipientes que equivalem a campos de um formulário.

Esses itens possuem nomes e são associados a campos com a mesma denominação. “Só é possível ver o conteúdo de um item se houver um formulário (UI) que tenha um campo (Field) de mesmo nome do item”, (Wikipédia Lotus, 2011). Por exemplo, caso exista um campo em um formulário com o nome “valor”, após a gravação dos dados desse formulário, o item será salvo com essa denominação e será associado ao campo “valor” de acordo com o exemplo em questão.

Esse trabalho tem o intuito de utilizar a linguagem LotusScript juntamente com a XML com a finalidade de reaproveitar dados existentes no sistema Lattes, que é a base de dados de currículos de pesquisadores e instituições (Lattes Cnpq, 2011 ), para o Opus. O Opus é um sistema que armazena as produções intelectuais da UFMG, e seus dados são exportados para sistemas internos: sistema INA que faz a avaliação de produção docente, para a Coleta CAPES e para bibliotecas digitais como a Biblioteca de Teses e Dissertações, BDTD, a qual permite o acesso à obra completa. Possui um fluxo de controle dos lançamentos o qual permite que departamentos, cursos, pró-reitorias e outras instâncias possam verificar e corrigir

dados, garantindo a qualidade da informação armazenada (Laboratório de Computação Científica da UFMG, 2011). Atualmente as produções são lançadas manualmente, uma de cada vez, exigindo pessoas e tempo para a execução da tarefa. Ainda podemos citar como problema dessa situação, a falta da informação correta e completa nas produções e a duplicação da mesma obra no sistema.

### **1.1. Objetivo**

A partir do desenvolvimento dessa monografia, pretende-se elaborar uma proposta ou um aplicativo Notes como uma forma de automatizar o lançamento de produções intelectuais dentro do sistema Opus através da importação de dados do sistema de Currículos Lattes. Com isso o processo de lançamento de produções passará a ser mais rápido, com exigência de menos pessoas para o cadastramento, disponibilização de informações mais completas e minimização de chances de duplicação de obras.

### **1.2. Estrutura do Trabalho**

O texto está dividido em duas partes, a primeira é a parte descritiva e conceitual a respeito do assunto abordado, e a segunda, é a proposta de um aplicativo ou simplesmente um documento de estudo de criação do aplicativo.

Faz parte dessa produção aplicativos ligados ao Lotus Domino, como por exemplo, o ambiente de desenvolvimento, Domino Designer, e um banco de documentos para ser utilizados durante os testes. Os testes e o desenvolvimento do aplicativo foram elaborados com a ajuda dos recursos disponibilizados no LCC-UFMG (Laboratório de Computação Científica da UFMG).

## **2. LATTES E OPUS**

### **2.1. Sistema de Currículos Lattes**

De acordo com o CNPq, o Lattes é a base de dados de currículos, instituições e grupos de pesquisa das áreas de Ciência e Tecnologia. Desde os anos 80, já havia interesse por parte dessa instituição de criar um formulário padrão para registrar os currículos dos pesquisadores brasileiros. A partir desse interesse o sistema originou-se com a denominação de Banco de Currículos onde contava com a captação dos dados em papel e em seguida com a digitação no sistema. No início dos anos 90, passa a se chamar BCURR onde a captação dos dados passa a ser em um formulário eletrônico dentro do sistema operacional DOS e em seguida enviado via disquete para ser importado na base de dados. Tempos depois, esse sistema evolui sendo nomeado de Cadastro Nacional de Competências em Ciência e Tecnologia (CNCT) e caracterizado por possuir um formulário eletrônico no ambiente Windows e em seguida os dados eram enviados de forma off line através da Internet. E finalmente, no final dessa mesma década, através dos grupos (CESAR - Centro de Estudos e Sistemas Avançados do Recife - da Universidade Federal de Pernambuco, e o grupo Stela - atual Instituto Stela - da Universidade Federal de Santa Catarina) é desenvolvida uma única versão com a capacidade de integrar as já existentes. Assim, meados do ano de 1999, o CNPq padronizou e lançou o Currículo Lattes para ser o formulário de currículo da esfera do Ministério da Ciência e Tecnologia e CNPq.

A partir de 2001, começou a discussão com relação à abertura e padronização XML com relação ao Lattes. Algumas universidades como UFSC, UNICAMP, UFRJ, USP, UFRGS, UFBA e UFRN solicitaram ao CNPq a abertura tecnológica das informações dessa plataforma, a partir disso, originou a construção da Linguagem de Marcação da Plataforma Lattes (LMPL), sob coordenação da CGINF/CNPQ, sendo os trabalhos de desenvolvimento conduzidos pelo Grupo Stela da UFSC. Mais adiante esse trabalho resultou na formação da Comunidade Virtual LMPL, que definiu o modelo DTD (Data Type Definition) XML do Currículo Lattes. Esse padrão XML foi desenvolvido inicialmente utilizando a linguagem de definição de tipos, DTD (Document Type Definition). Em seguida, com a homologação da linguagem XML Schema pelo Consórcio W3C, a comunidade CONSCIENTIAS-



LMPL construiu uma nova estrutura utilizando a linguagem de esquemas para o mesmo padrão XML de Currículo Vitae.

Com isso, tornou-se viável a partir da versão 1.4 do Lattes a abertura da Plataforma, do ponto de vista de conteúdo dos dados, ficando inalterado o acesso técnico às informações, preservando a segurança dos pesquisadores.

Atualmente o Lattes é um padrão nacional da vida pregressa e atual dos pesquisadores e estudantes. Por sua riqueza de informações e sua crescente confiabilidade e abrangência, tornou-se um elemento essencial e compulsório à análise de mérito e competência das solicitações de financiamentos na área de ciência e tecnologia. E além de ser um sistema estratégico para as atividades de planejamento e gestão é também utilizado na formulação das políticas do Ministério de Ciência e Tecnologia e de outros órgãos governamentais da área de ciência, tecnologia e inovação.

## **2.2. Sistema Opus**

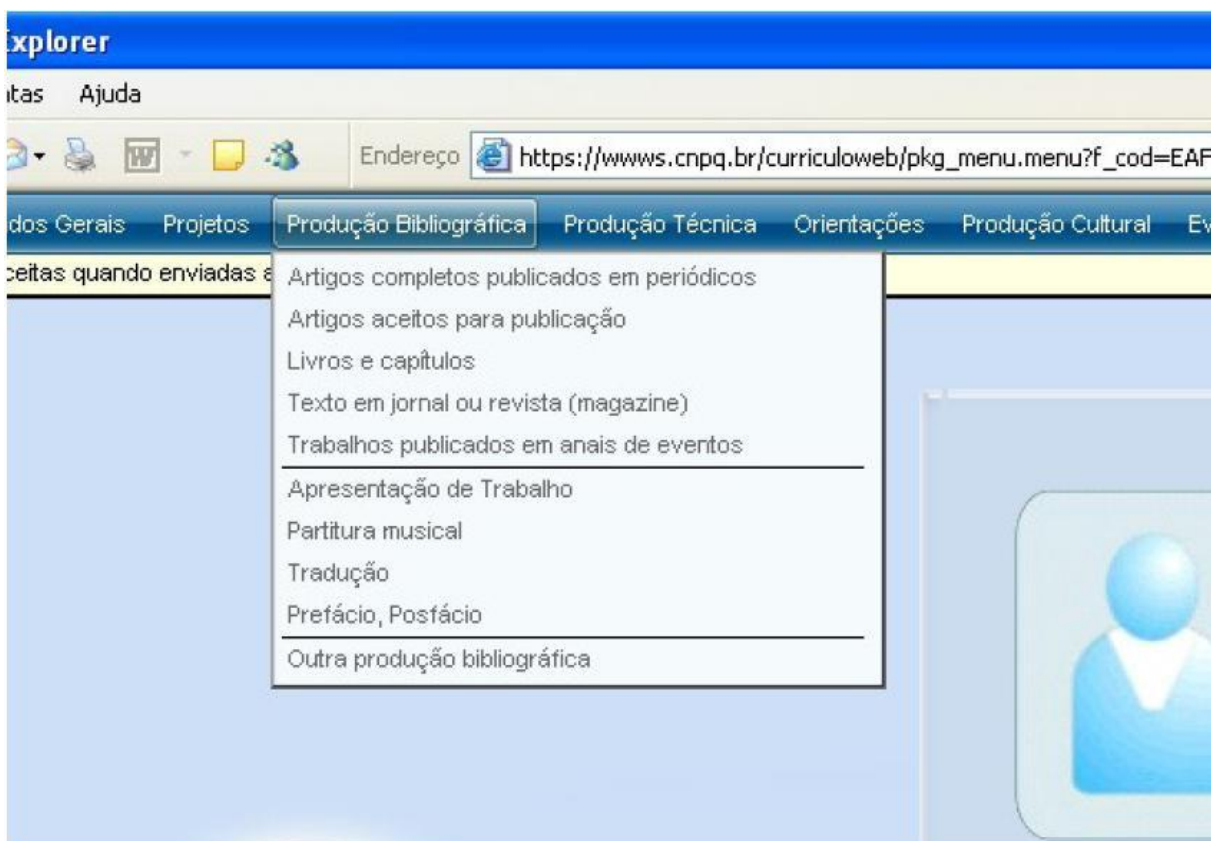
O sistema Opus é responsável pelo cadastro de produção intelectual da UFMG. O processo abrange desde o cadastro de dados de uma produção bibliográfica pelo professor ou departamento, a inserção da obra integral em formato eletrônico, até a aprovação dos dados pelos responsáveis dos diversos órgãos dessa universidade. Em cada tipo de produção desse sistema, existe uma grande quantidade de dados que foram selecionados de forma a permitir a exportação para vários sistemas externos.

O Opus integra e unifica os vários fluxos existentes internamente à UFMG de modo a ceder informações para outras áreas (como o CPPD e PRPq) e exporta dados para outros sistemas como o da Biblioteca Digital, Pergamum e Informações Acadêmicas – INA. E tem como premissa para a segunda fase do seu desenvolvimento a importação e ou exportação de dados para o Lattes.

## **2.3. Comparação entre o Lattes e o Opus**

Tanto o Lattes quanto o Opus possuem formulários de cadastro de produções científicas semelhantes, o que conseqüentemente armazenam as mesmas informações.

A figura 1, retirada do Currículos Lattes, exibe os diferentes tipos de produção bibliográfica existentes nesse sistema:

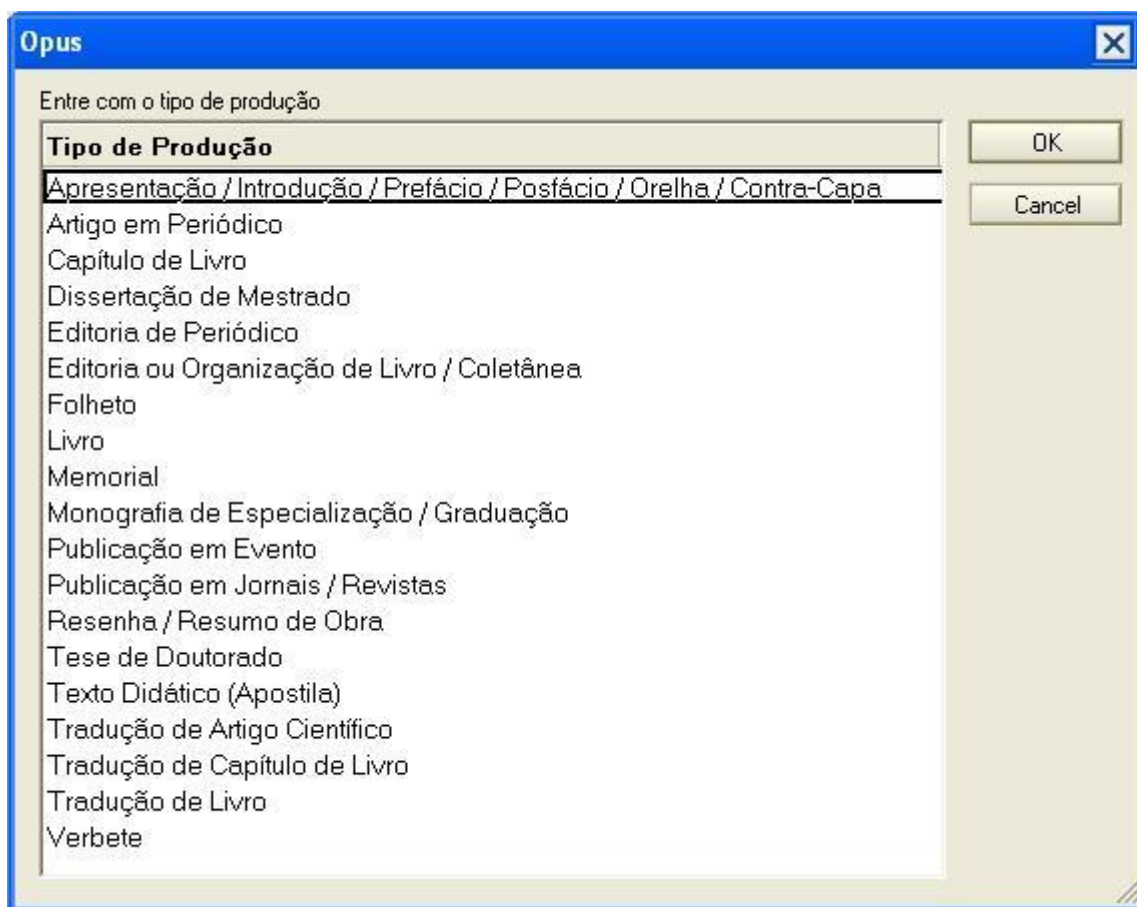


**Figura 1 Produções Bibliográfica Lattes**  
**Fonte: Próprio Autor.**

- Artigos completos publicados em periódicos: área destinada para inclusão de artigos científicos já publicados em revistas indexadas com ISSN.
- Artigos aceitos para publicação; artigos que ainda não foram publicados. Devem ser indexadas com ISSN.
- Livros e capítulos: área para inclusão de informações a respeito de livros ou capítulos de livros produzidos pelo usuário (estas publicações devem ser indexadas com ISBN).
- Texto em jornal ou revista: qualquer publicação escrita que tenha sido publicada em meio jornalístico, como roteiros, ensaios, matérias, reportagens, relatos, depoimentos, entrevistas, resumos, resenhas, crônicas, contos, poemas e afins.
- Trabalhos publicados em anais de eventos: textos publicados em anais de eventos, vinculados a um evento específico (este item possui um vínculo com o item “Eventos”).

- Apresentação de Trabalho: apresentação de trabalho não vinculada a evento (aulas magnas, palestras, trabalhos acadêmicos, etc.)
- Partitura musical: área para cadastro de partituras escritas para canto, coral, orquestra, etc.
- Tradução: destina-se ao cadastramento de artigos, livros ou outras publicações traduzidas pelo usuário.
- Prefácio, posfácio: destina-se ao cadastramento de prefácio, posfácio, introdução ou apresentação de livros.
- Outra produção bibliográfica: qualquer outra produção bibliográfica que não se enquadre nas opções anteriores, inclusive artigos publicados em periódicos sem ISSN.

Agora, no Opus os tipos de produção que podem ser cadastradas são: artigos em periódico, capítulo de livro, dissertação de mestrado, editoria de periódico, publicação em evento, tese de doutorado, tradução de artigo, tradução de livro, tradução de capítulo, entre outros que estão ilustrados na figura 2.



**Figura 2 Tipos de Produções Bibliográficas do Opus**  
**Fonte: Próprio Autor.**

Com a figura 1 (tipo de produções do Lattes) e figura 2 (tipo de produções do Opus), observamos que esses sistemas trabalham os cadastros de produções de maneira específica e personalizada para cada tipo de produção, objetivando tornar o processo o mais amigável e intuitivo possível. Por exemplo, na publicação do tipo livro no sistema Opus temos que inserir os seguintes dados: título, isbn, tipo de edição, número da edição, editora, cidade, data da publicação, número de páginas, instituição, idioma da publicação, formas de divulgação, circulação e autores. A figura 3 ilustra a primeira página do cadastro de uma publicação do tipo livro no sistema Opus.

**Livro**

Publicação

Dados Bibliográficos | Autores | Dados Complementares | Resumo/Texto Completo | Biblioteca

\* campos obrigatórios

Código	DLCC-8R9N2A
* Título	<input type="text"/>
Subtítulo	<input type="text"/>
ISBN	<input type="text"/>
* Tipo de edição	<input type="text"/> ▾
* Número da Edição	<input type="text"/>
* Editora	<input type="text"/>
<b>Local:</b>	
Endereço	<input type="text"/>
* Cidade	<input type="text"/>
Estado/Provincia	<input type="text"/>
* País	<input type="text"/> Brasil ▾
* Data da publicação (sempre incluir o ano)	Ano: <input type="text"/> ▾ Mês: <input type="text"/> ▾ Dia: <input type="text"/> ▾
Número de volumes (não é tiragem)	<input type="text"/>
* Número de páginas	<input type="text"/>
* Idioma da publicação	<input type="text"/> Português ▾
Ilustração	<input type="text"/>
Material que acompanha	<input type="text"/>
Série/coleção	<input type="text"/>
Número da coleção/série	<input type="text"/>
Número do ISBN da coleção/série	<input type="text"/>
* Formas de divulgação	<input type="text"/> Impresso ▾
* Circulação	<input type="text"/> ▾
Palavras-chave (separar com vírgula)	<input type="text"/>
Notas gerais	<input type="text"/>

**Figura 3 Publicação do Tipo Livro no Opus**  
**Fonte: Próprio Autor.**

Já no sistema Lattes os dados a serem adicionados para esse mesmo tipo de publicação são: título do livro, ano, país, idioma, meio de divulgação, número de volumes, número de páginas, isbn, número da edição/revisão, série, cidade da

editora, nome da editora, autores e palavras-chave. A figura 4 exemplifica a tela de cadastro de uma publicação do tipo livro nesse sistema com esses dados.

**Livro publicado / organizado**

Dados gerais

Detalhamento

Autores

Palavras-chave

Áreas

Setores

Outras informações

Traduções para o inglês

**Dados gerais**

Tipo  
 Livro publicado  Organização de obra publicada

Natureza

Título do livro

Ano

País  
Brasil

Idioma  
Português

Meio de divulgação

Home page do trabalho (URL)

É um dos 5 trabalhos mais relevantes de sua produção?  
 sim  não

**Detalhamento**

Número de volumes

Número de páginas

ISBN

Número da edição/revisão

Série

Cidade da editora

Nome da editora

**Autores**

Informe a seqüência de autoria da produção de acordo com a publicação.  
Selecione um item e utilize as setas para alterar a ordem.

Ordem	Autores
+ incluir um novo item	

Palavras-chave

**Figura 4 Produção Bibliográfica do Lattes**  
Fonte: Próprio Autor.

Com a figura 3 e 4, foi possível observar a existência dos mesmos dados cadastrais (título do livro, isbn, data publicação, país, idioma, forma de divulgação, número de volumes, número de páginas, número da edição/revisão, série, cidade da editora, nome da editora, autores e palavras-chave) para a publicação do tipo livro nesses dois sistemas.

### 3. XML

Extensible Markup Language (XML) é uma linguagem de marcação de dados que provê um formato para descrever dados estruturados, permite a definição de um número infinito de tags (marcações) que são utilizadas para criar dados estruturados. Essa linguagem é um subconjunto da SGML (Standard Generalized Markup Language), otimizada para distribuição de dados através da web e é definido pelo World Wide Web Consortium (W3C), o qual assegura que os dados estruturados serão uniformes e independentes de aplicações e fornecedores.

Dentre as características da XML, podemos citar as seguintes:

- Integração de dados de fontes diferentes: permite a distribuição de dados entre banco de dados que não possuem o mesmo schema.
- Computação e manipulação local: os dados XML recebidos por um cliente são analisados, editados e manipulados de acordo com o interesse do usuário e sem a necessidade de reconectar no servidor de origem.
- Múltiplas formas de visualização dos dados: os dados podem ser visualizados de maneiras diferentes, uma vez que a XML define apenas os dados e não o visual. O visual pode ser gerado através dos recursos de CSS e XSL.
- Atualização granular do documento: apenas itens que sofreram modificação ou são adicionados aos já existentes são reenviados para o cliente.

Na figura 5 temos um exemplo de um documento XML sendo exibido em um browser.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
- <curriculo>
  - <dados-pessoa nome="Walison Dias" cpf="10053378698" ID="01">
    <universidade nome="UFMG" ID="0001"/>
    <curso nome="Especialização em Informática" ano="2011" codcurso="0023"/>
  </dados-pessoa>
</curriculo>
```

Figura 5 - Exemplo Documento XML

Fonte: Próprio Autor.

Para manipular documentos XML extensos, como no caso dos documentos XML do Lattes, a forma mais utilizada é através do SAX. O SAX, API simples para XML, é um padrão para manipulação de documentos XML baseado em eventos. Enquanto processa o documento XML, o parser detecta partes do documento (tag

inicial e tag final) e envia esta informação para o programa. O SAX tem como vantagem o menor gasto de memória através do processamento serial do documento. É utilizada em casos em que a leitura de um documento XML de grande porte pode ser feita de forma sequencial. Não há necessidade de se voltar à leitura para algum ponto já lido, mas se isso ocorrer, o SAX reinicia a leitura do documento até chegar ao local desejado, pois a sua especificação estabelece a leitura de um documento XML de forma sequencial.

### 3.1. Utilizando XML com LotusScript

O código, do apêndice A, foi escrito na linguagem LotusScript e tem como objetivo ler e capturar informações de um arquivo XML que encontra-se exemplificado na figura 6, e em seguida, gerar um arquivo texto com as informações desse XML.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
- <curriculo>
  - <dados-pessoa nome="Walison Dias" cpf="10053378698" ID="01">
    <universidade nome="UFMG" ID="0001"/>
    <curso nome="Especialização em Informática" ano="2011" codcurso="0023"/>
  </dados-pessoa>
</curriculo>
```

Figura 6 Arquivo XML usado no agente LotusScript  
Fonte: Próprio Autor.

- Resultado da Leitura do Agente LotusScript:
  1. Início do Documento
  2. Elemento=curriculo
  3. Elemento=dados-pessoa
  4. -- Atributo=ID Valor=01, Tipo =CDATA
  5. -- Atributo=cpf Valor=10053378698, Tipo =CDATA
  6. -- Atributo=nome Valor=Walison Dias, Tipo =CDATA
  7. Elemento=universidade
  8. -- Atributo=ID Valor=0001, Tipo =CDATA
  9. -- Atributo=nome Valor=UFMG, Tipo =CDATA
  10. Fim Elemento=universidade
  11. Elemento=curso

- 12.-- Atributo=codcurso Valor=0023, Tipo =CDATA
- 13.-- Atributo=nome Valor=Especialização em Informática, Tipo =CDATA
- 14.-- Atributo=ano Valor=2011, Tipo =CDATA
- 15.Fim Elemento=curso
- 16.Fim Elemento=dados-pessoa
- 17.Fim Elemento=curriculo
- 18.Fim Documento

Com esse exemplo é possível observar que a linguagem permite acessar e ler os elementos e atributos do arquivo xml através da classe NotesSAXParser. A captura dos elementos e atributos do arquivo xml é feito principalmente pelo evento SAXStartElement. Nesse evento é possível acessar os elementos do arquivo XML através da variável elementname e seus respectivos atributos, caso tenha, através da variável attributes.



#### 4. ESTUDO DE IMPORTAÇÃO DE DADOS DO LATTES PARA O OPUS

Diante dos vários tipos de produção existente no Opus, a complexidade e característica individual dessas produções, escolheu-se um tipo de produção científica, artigo em periódico, para prosseguir no desenvolvimento desse trabalho e em seguida produzir um software que possa importar esses dados do Lattes para o Opus.

O cadastro desse tipo de produção científica no Opus, de acordo com a figura 7, exige obrigatoriamente os seguintes dados: tipo do trabalho, título do artigo, meio de publicação, página inicial, página final, idioma da publicação e ainda, para completar o cadastro exige-se a entrada dos dados do periódico: título do periódico, sede do periódico, volume, número, data da defesa e finalmente os autores.

**Artigo em Periódico**

Publicação

Dados Bibliográficos | Periódico | Autores | Dados Complementares | Resumo/Texto Completo | Biblioteca

\* campos obrigatórios

Código	DLCC-8R9MVX
* Tipo do trabalho	Artigo
* Título do artigo	
Subtítulo	
Título do caderno ou seção	
Número do caderno ou seção	
* Meio de Publicação	Impresso
* Página inicial	
* Página final	
* Idioma da publicação	Português
Palavras-chave (separar com vírgula)	
Notas gerais	

Figura 7 Dados do Artigo em Periódico Opus  
Fonte: Próprio Autor.

SALVAR... FECHAR

### Artigo em Periódico

Publicação

Dados Bibliográficos | Periódico | Autores | Dados Complementares | Resumo/Texto Completo | Biblioteca

#### Dados do Periódico

Este é o periódico que você selecionou. Você pode pesquisar e selecionar outro clicando em Pesquisar Periódico ou você pode pesquisar e incluir os dados da edição em Pesquisar Edição de Periódico.

Código	ECID-5SES7H
* Título do periódico	Revista de Informática Pública
Abreviatura	
ISSN	15160697X
Instituição responsável	
* Sede do periódico	Nacional

Periódico

#### Dados da Edição de Periódico

Este é a edição do periódico que você selecionou. Você pode selecionar ou cadastrar uma nova clicando em Pesquisar Edição de Periódico.

Código	MCMI-79CGNZ
* Volume da publicação	25
* Número ou fascículo	1
* Data da defesa (sempre incluir o ano)	26/8/2006 -26/08/2006

Edição de Periódico

**Figura 8 Dados da Edição do Periódico Opus**  
**Fonte: Próprio Autor.**

As figuras 7 e 8 demonstram campos que são visíveis para o usuário. Porém ainda existem algumas dezenas de outros campos invisíveis, disponível no apêndice B, que funcionam como campos de controle da produção, seja para uma exportação ou importação para outro sistema ou uma marcação do status da produção.

A figura 9 exemplifica um cadastro de um artigo no Lattes, nele precisamos preencher os seguintes campos: DOI, título, ano, idioma, meio de divulgação, título do periódico, ISSN, volume, série, página inicial, página final, palavras-chave e autores.

**Artigo completo publicado em periódico**

D.O.I

Dados gerais

Informe o [Digital Object Identifier \(DOI®\)](#) do artigo e clique no botão para recuperar os dados da produção

Dados gerais

Título  Ano

Idioma  Meio de divulgação

Home page do trabalho (URL)

É um dos 5 trabalhos mais relevantes de sua produção?

sim  não

Detalhamento

Título do periódico/revista em que o artigo foi publicado   ISSN

Volume  Série  Página inicial/ Número artigo eletrônico  Página final

Autores

Informe a sequência de autoria da produção de acordo com a publicação.  
Selecione um item e utilize as setas para alterar a ordem.

**Figura 9 Campos de um artigo completo Lattes**  
**Fonte: Próprio Autor.**

No sistema de currículo Lattes esses dados são acessados através da funcionalidade de exportação de dados, o qual origina um arquivo XML do currículo de uma pessoa. Esse arquivo XML contém informações básicas de caráter pessoal do usuário, as produções bibliográficas e produções técnicas. Para esse trabalho estamos interessados apenas na parte que contém os dados a respeito dos artigos. Na figura 10 temos o DTD do currículo Lattes, porém apenas a parte de artigos científicos.

```

<!ELEMENT ARTIGOS-PUBLICADOS (ARTIGO-PUBLICADO*)>
<!ELEMENT ARTIGO-PUBLICADO (DADOS-BASICOS-DO-ARTIGO?, DETALHAMENTO-
DO-ARTIGO?, AUTORES*, PALAVRAS-CHAVE?, AREAS-DO-CONHECIMENTO?,
SETORES-DE-ATIVIDADE?, INFORMACOES-ADICIONAIS?)>
<!ATTLIST ARTIGO-PUBLICADO
    SEQUENCIA-PRODUCAO CDATA #IMPLIED
>
<!ELEMENT DADOS-BASICOS-DO-ARTIGO EMPTY>
<!ATTLIST DADOS-BASICOS-DO-ARTIGO
    NATUREZA (COMPLETO | RESUMO | NAO_INFORMADO) #IMPLIED
    TITULO-DO-ARTIGO CDATA #IMPLIED
    ANO-DO-ARTIGO CDATA #IMPLIED
    PAIS-DE-PUBLICACAO CDATA #IMPLIED
    IDIOMA CDATA #IMPLIED
    MEIO-DE-DIVULGACAO (IMPRESSO | MEIO_MAGNETICO | MEIO_DIGITAL |
FILME | HIPERTEXTO | OUTRO | VARIOS | NAO_INFORMADO) "NAO_INFORMADO"
    HOME-PAGE-DO-TRABALHO CDATA #IMPLIED
    FLAG-RELEVANCIA (SIM | NAO) "NAO"
    DOI CDATA #IMPLIED
>
<!ELEMENT DETALHAMENTO-DO-ARTIGO EMPTY>
<!ATTLIST DETALHAMENTO-DO-ARTIGO
    TITULO-DO-PERIODICO-OU-REVISTA CDATA #IMPLIED
    ISSN CDATA #IMPLIED
    VOLUME CDATA #IMPLIED
    FASCICULO CDATA #IMPLIED
    SERIE CDATA #IMPLIED
    PAGINA-INICIAL CDATA #IMPLIED
    PAGINA-FINAL CDATA #IMPLIED
    LOCAL-DE-PUBLICACAO CDATA #IMPLIED
>

```

**Figura 10 DTD Currículo Lattes - Parte Artigos**  
**Fonte: Próprio Autor.**

Diante dessas informações temos conhecimento de quais dados precisamos para criar um artigo no Opus e através do DTD do arquivo XML sabemos quais elementos e atributos iremos capturar.

#### 4.1. Lógica da Importação dos Dados

A lógica da importação dos dados do Currículo Lattes para o Opus segue o seguinte algoritmo:

1. Através de uma caixa de diálogo o usuário localiza o arquivo, currículo lattes, em sua máquina.
2. O programa inicia a leitura do arquivo XML através da classe SAX.

3. A leitura do arquivo XML é realizada de forma sequencial. A identificação dos dados dos artigos no currículo lattes é feita através do elemento denominado "ARTIGO-PUBLICADO".

3.1. Ao iniciar um elemento do arquivo XML, este passa pelo evento SAX\_StartElement, onde pode-se acessar o valor desse elemento e seus possíveis atributos.

3.2. Ao terminar a descrição de um elemento, passa-se pelo evento SAX\_EndElement, o qual permite identificar a finalização da descrição do elemento.

4. Para cada artigo encontrado:

4.1. Captura-se os dados: TITULO-DO-ARTIGO, IDIOMA, MEIO-DE-DIVULGACAO, PAGINA-INICIAL, PAGINA-FINAL, TITULO-DO-PERIODICO-OU-REVISTA, ISSN, VOLUME, SERIE, PALAVRA-CHAVE-1, PALAVRA-CHAVE-2, PALAVRA-CHAVE-3, PALAVRA-CHAVE-4, PALAVRA-CHAVE-5, PALAVRA-CHAVE-6.

4.2. Verifica-se a existência do artigo no sistema Opus através do título do artigo e seu issn.

4.3. Caso exista, o programa continua o processo de leitura em busca da próxima publicação.

4.4. Caso não exista, inicia-se o processo de criação do artigo passando do Lattes para o Opus o TITULO-DO-ARTIGO, IDIOMA, MEIO-DE-DIVULGACAO, PAGINA-INICIAL, PAGINA-FINAL. E logo após essa etapa, verifica se o periódico já se encontra no Opus através do título do periódico, issn, volume e série:

4.4.1. Caso exista, acrescenta a esse novo documento os dados do periódico disponível no Lattes: TITULO-DO-PERIODICO-OU-REVISTA, ISSN, VOLUME e SERIE. Caso contrário, o usuário ficará responsável por adicionar os dados do periódico no artigo.

5. O programa, em seguida, salva o novo documento criado com o status de "rascunho" para que o mesmo possa ser completado e ou revisto pelo usuário.

O documento fica disponível no sistema Opus com o status de rascunho até que o usuário responsável entre com os dados dos autores e salve o documento, o que

faz com que o mesmo entre no status “normal”, ficando disponível para consultas e exportação para outros sistemas caso seja necessário.

#### **4.2. Código de Importação de Dados do Lattes para o Opus**

No apêndice C está o código LotusScript que executa o processo de importação de artigos XML para o sistema de cadastro de produções intelectuais da UFMG.

Esse código contém além dos métodos da classe NotesSAXParser mais 3 funções criadas para auxiliar no processo de importação: A função “LocalizaArtigosOpus” que tem como objetivo localizar a existência ou não do artigo capturado no Lattes na base Opus. Em seguida a função “LocalizaEdicaoPeriodicoOpus” que tem como função trazer informações da edição do periódico caso o mesmo já exista inserido na base Opus. Essas funções executam as pesquisas através de campos do tipo texto, o que não é perfeito para esse tipo de pesquisa, pois há o risco de não encontrar um artigo ou periódico, por existir ou não, uma letra no texto procurado. E por último, foi criada a função “CriaArtigo\_Periodico” que executa o processo de criação e gravação do artigo e do periódico na base de dados do Opus.

O código segue os seguintes passos: primeiro captura os dados dos artigos publicados em um arquivo XML exportado do Currículo Lattes, no exemplo disponível na figura 11 há dois artigos pertencentes a uma professora da UFMG que foi exportado do Lattes.

```

<ARTIGOS-PUBLICADOS>
- <ARTIGO-PUBLICADO SEQUENCIA-PRODUCAO="53">
  <DADOS-BASICOS-DO-ARTIGO TITULO-DO-ARTIGO="Type-safe Versioned Object Query Language" ANO-DO-ARTIGO="2006" PAIS-DE-PUBLICACAO="" TITULO-DO-ARTIGO-INGLES="" MEIO-DE-DIVULGACAO="IMPRESSO" HOME-PAGE-DO-TRABALHO="[http://www.jucs.org/jucs_12_7/type_safe_versioned_object]" FLAG-RELEVANCIA="NAO" DOI="10.3217/jucs-012-07-0938" IDIOMA="Inglês" NATUREZA="COMPLETO"/>
  <DETALHAMENTO-DO-ARTIGO TITULO-DO-PERIODICO-OU-REVISTA="Journal of Universal Computer Science" ISSN="098486968" LOCAL-DE-PUBLICACAO="" VOLUME="12" FASCICULO="" SERIE="7" PAGINA-INICIAL="938" PAGINA-FINAL="957"/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Rodrigo Machado" ORDEM-DE-AUTORIA="1" NOME-PARA-CITACAO="MACHADO, R." NRO-ID-CNPQ="0456516366590574"/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Alvaro F Moreira" ORDEM-DE-AUTORIA="2" NOME-PARA-CITACAO="MOREIRA, A. F." NRO-ID-CNPQ="3762855185207010"/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Renata de Matos Galante" ORDEM-DE-AUTORIA="3" NOME-PARA-CITACAO="GALANTE, Renata de Matos" NRO-ID-CNPQ=""/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Mirella Moura Moro" ORDEM-DE-AUTORIA="4" NOME-PARA-CITACAO="MORO, M. M.; Mirella M. Moro; Mirella Moura Moro;Moro, Mirella M."/>
  <PALAVRAS-CHAVE PALAVRA-CHAVE-1="Linguagem de Consultas" PALAVRA-CHAVE-2="Modelo de versões" PALAVRA-CHAVE-3="" PALAVRA-CHAVE-4="" PALAVRA-CHAVE-5="" PALAVRA-CHAVE-6=""/>
  <AREAS-DO-CONHECIMENTO/>
  <INFORMACOES-ADICIONAIS DESCRICAO-INFORMACOES-ADICIONAIS="WebQualis CC 2007: B2. Cited by 2 (Google Scholar, outubro/2010)." DESCRICAO-INFORMACOES-ADICIONAIS-INGLES=""/>
</ARTIGO-PUBLICADO>
- <ARTIGO-PUBLICADO SEQUENCIA-PRODUCAO="54">
  <DADOS-BASICOS-DO-ARTIGO TITULO-DO-ARTIGO="Modelo Temporal de Versões" ANO-DO-ARTIGO="2002" PAIS-DE-PUBLICACAO="" TITULO-DO-ARTIGO-INGLES="" MEIO-DE-DIVULGACAO="IMPRESSO" HOME-PAGE-DO-TRABALHO="[http://www.inf.ufrgs.br/~revista/Numpub.html#Volume%20IX]" FLAG-RELEVANCIA="NAO" DOI="" IDIOMA="Português" NATUREZA="COMPLETO"/>
  <DETALHAMENTO-DO-ARTIGO TITULO-DO-PERIODICO-OU-REVISTA="Revista de Informática Teórica e Aplicada" ISSN="01034308" LOCAL-DE-PUBLICACAO="Porto Alegre" VOLUME="9" FASCICULO="1" SERIE="" PAGINA-INICIAL="37" PAGINA-FINAL="51"/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Nina Edelweiss" ORDEM-DE-AUTORIA="2" NOME-PARA-CITACAO="EDELWEISS, Nina" NRO-ID-CNPQ="2702553270286995"/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Clesio Saraiva dos Santos" ORDEM-DE-AUTORIA="3" NOME-PARA-CITACAO="SANTOS, Clesio Saraiva dos" NRO-ID-CNPQ="0965164091828153"/>
  <AUTORES NOME-COMPLETO-DO-AUTOR="Mirella Moura Moro" ORDEM-DE-AUTORIA="1" NOME-PARA-CITACAO="MORO, M. M.; Mirella M. Moro; Mirella Moura Moro;Moro, Mirella M."/>
  <PALAVRAS-CHAVE PALAVRA-CHAVE-1="Modelo de dados temporal" PALAVRA-CHAVE-2="Modelo de versões" PALAVRA-CHAVE-3="" PALAVRA-CHAVE-4="" PALAVRA-CHAVE-5="" PALAVRA-CHAVE-6=""/>
  <AREAS-DO-CONHECIMENTO/>
  <INFORMACOES-ADICIONAIS DESCRICAO-INFORMACOES-ADICIONAIS="WebQualis CC 2007: B3. Citado por 19 (Google Scholar, março/2010)." DESCRICAO-INFORMACOES-ADICIONAIS-INGLES=""/>
</ARTIGO-PUBLICADO>
</ARTIGOS-PUBLICADOS>

```

Figura 11 Produções XML a serem importadas para o Opus  
Fonte: Próprio Autor.

Para cada artigo capturado, verifica através da primeira função se o arquivo existe no Opus, caso exista, o programa vai para o próximo artigo. Caso contrário, cria-se o artigo no Opus, de acordo com a tabela 1, associando as variáveis do Lattes com as do Opus:

Campo no Opus	Campo no Lattes	Objetivo
TituloPub	TITULO-DO-ARTIGO	Título do artigo
PagInicPub	PAGINA-INICIAL	Página inicial do artigo
PagFimPub	PAGINA-FINAL	Página final do artigo
MeioPub	MEIO-DE-DIVULGACAO	Meio de publicação do artigo
IdiomaPub	IDIOMA	Idioma da publicação

Tabela 1 Variáveis do artigo do Opus x variáveis do artigo do Lattes  
Fonte: Próprio Autor.

Em seguida, verifica-se a existência do periódico na base Opus através da segunda função. Caso não exista, o artigo fica sem associação de um periódico sendo necessária uma associação manual, se ele existir no sistema, continua o processo de associação do novo documento com os dados da edição de periódico localizado. As variáveis destinadas ao periódico no Opus recebem os dados do periódico do Lattes de acordo com a tabela 2.

Campo no Opus	Campo no Lattes	Objetivo
TituloVeiculo	TITULO-DO-PERIODICO-OU-REVISTA	Título do periódico
ISSN	ISSN	Número ISSN do periódico
VolumePub	VOLUME	Número do volume do periódico
FasciculoPub	SERIE	Número série do periódico

**Tabela 2 Associação variáveis do periódico Opus x variáveis do periódico Lattes**

**Fonte: Próprio Autor.**

Após passar pela parte da associação ou não de um artigo com o seu periódico, grava-se o arquivo na base Opus. A figura 12 e 13 ilustra o resultado do processo de importação do artigo identificado pela sequencia 53 na figura 11 dentro do sistema Opus.

The screenshot shows the Opus system interface. On the left, there is a sidebar with the Opus logo and a navigation menu. The main content area is titled 'NOVA PRODUÇÃO' and contains a search bar and a table of publications. The table has three columns: 'Tipo', 'Código', and 'Título'. The first row is expanded to show 'Publicação em Periódico' with code 'DLCC-7YAG4R' and title 'Artigo 11/2009'. Below this, there is a section for 'Publicação sem Veículo' with code 'WDSA-8R3SEN'.

**Figura 12 Produções importadas no Opus**

**Fonte: Próprio Autor.**



**Artigo em Periódico**

Publicação | Avaliação

Dados Bibliográficos | Periódico | Autores | Dados Complementares | Resumo/Texto Completo | Biblioteca | \* campos obrigatórios

Código	WDSA-8RGT4D
* Tipo do trabalho	Artigo
* Título do artigo	Type-safe Versioned Object Query Language
Subtítulo	
Título do caderno ou seção	
Número do caderno ou seção	
* Meio de Publicação	IMPRESSO
* Página inicial	938
* Página final	957
* Idioma da publicação	Inglês
Palavras-chave (separar com vírgula)	
Notas gerais	

**Artigo em Periódico**

Publicação | Avaliação

Dados Bibliográficos | Periódico | Autores | Dados Complementares | Resumo/Texto Completo | Biblioteca |

Dados do Periódico

Código	RVMR-65KP9K
* Título do periódico	Journal of Universal Computer Science
Abreviatura	
ISSN	09848-6968
Instituição responsável	
* Sede do periódico	Estrangeiro

Dados da Edição de Periódico

Código	RVMR-6TBRGE
* Volume da publicação	12
* Número ou fascículo	7
* Data da defesa (sempre incluir o ano)	31/8/2006 -31/08/2006

**Figura 13 Produção importada do Lattes para o Opus**  
**Fonte: Próprio Autor.**

Essa nova produção ainda depende de uma intervenção manual por parte do usuário, pois falta a inserção dos autores que depende de acesso externo por parte do programa a outro banco de dados e ainda, o preenchimento da associação da área de conhecimento que é feita após a seleção dos autores.

## 5. CONCLUSÃO

Esse trabalho utilizou a linguagem LotusScript juntamente com a XML para criar uma solução, em forma de um programa, com a finalidade de reaproveitar dados existentes no sistema Lattes. Para isso, foi necessário estudar os assuntos relativos a XML, a linguagem LotusScript e a implementação da XML com a LotusScript.

Graças à principal característica da XML, distribuição de dados entre banco de dados que não possuem o mesmo schema, associada à classe NotesSaxParse que permite manipular arquivos do tipo XML através da LotusScript, foi possível o desenvolvimento de um programa que importasse dados disponibilizados pelo Lattes para o Opus. Esta interface junto ao sistema Lattes é importante para o Opus neste momento e fundamental em futuras reengenharias que por ventura venham ocorrer, pois soluciona principalmente o problema com relação ao número de pessoas e tempo para o cadastramento das produções. Dessa maneira, tendemos a diminuir o esforço de seus usuários com relação ao gerenciamento de suas produções científicas no Opus.

Como sugestão para futuros trabalhos e melhoria desse programa, propõe-se: a inserção dos autores e associação da área de conhecimento durante o ato de importação, otimização das funções de busca dos artigos e periódicos, testar o ciclo de importação dos artigos criados por esse programa no Opus junto a outros sistemas, como por exemplo, o INA. Estudar as outras produções científicas, tanto no Opus quanto no Lattes, para que as mesmas também possam ser importadas. E claro, a criação da documentação do desenvolvimento desse programa.

Esta monografia faz parte do contínuo processo de melhoria do sistema de produções intelectuais da UFMG e constitui o primeiro passo diante de tantas tarefas ainda a serem desenvolvidas e implantadas para que o mesmo alcance o seu objetivo completo, o da importação de todas as produções científicas do Lattes.

## 6. REFERÊNCIAS

Almeida, M.B. Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares. UFMG - Artigo de Mestrando em ciência da informação – Disponível em: <[www.scielo.br/pdf/ci/v31n2/12903.pdf](http://www.scielo.br/pdf/ci/v31n2/12903.pdf)>. Acesso em: 25 de Maio 2011.

Comunidade LMPL. – Disponível em: < <http://Impl.cnpq.br/Impl/> > Acessado em: 01 de setembro de 2011.

DTD-Curriculum Lattes. – Disponível em:  
<<http://Impl.cnpq.br/Impl/Gramaticas/Curriculo/DTD/Documentacao/DTDCurriculo.pdf>> Acessado em 25 de agosto de 2011.

Domino Designer 6: A Developer's Handbook - Disponível em:  
<<http://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/sg246854.html>>. Acesso em: 25 de Maio 2011.

Elliott Rusty Harold and W. S. Coot Means. XML in a nutshell. 3ed. USA: O' Reilly Media, 2004.

Laboratório de Computação Científica- Sistema Opus - Disponível em:  
<[http://www.lcc.ufmg.br/index.php?option=com\\_content&view=article&id=7&Itemid=8](http://www.lcc.ufmg.br/index.php?option=com_content&view=article&id=7&Itemid=8)> Acessado em: 07 de Julho de 2011.

Ling Liu and M. Tamer Özsu. Encyclopedia of Database Systems. USA, New York: Springer-Verlag, p. 3571-3572. Ago, 2009.

Lattes CNPq. – Disponível em: <<http://lattes.cnpq.br/>> Acessado em: 07 de Julho de 2011.

Lattes Histórico. – Disponível em: <<http://lattes.cnpq.br/conteudo/historico.htm>> Acessado em 25 de agosto de 2011.

Manual Lattes – Disponível em:< <http://www.fisica.ufs.br/npgfi/manual-utili-lattes2009.pdf>>. Acessado em: 08 de julho de 2011.

Opus UFMG – Disponível em: <<http://opus.grude.ufmg.br/>> Acessado em: 07 de julho de 2011.

Tolentino, Ricardo José Vaz. Aplicações Web em XML Estágio Atual e Tendências Futuras. Belo Horizonte: Editora C/Arte, 2004.

Wikipédia LotusScript – Disponível em: <<http://pt.wikipedia.org/wiki/LotusScript>> Acessado em: 08 de julho de 2011.

XML Powered by Domino- How to use XML with Lotus Domino – Disponível em:  
<<http://www.redbooks.ibm.com/redbooks/pdfs/sg246207.pdf>>. Acesso em: 25 de Maio 2011.

## APÊNDICE A

O código a seguir tem como objetivo a leitura de um arquivo XML. Ele representa o primeiro teste de como é processado um arquivo XML através da linguagem LotusScript.

```
'Declaração Global
Dim session As NotesSession 'New NotesSession
Dim db As NotesDatabase
Dim doc As NotesDocument

'Inicio do Processo:
Sub Initialize
    Set session = New NotesSession
    Set db = session.CurrentDatabase
    Set doc = db.CreateDocument()

    Dim saxParser As NotesSAXParser
    Dim xml_in As NotesStream

    filename$ = "C:\Users\walison\Desktop\Ufmg-
T17_notebook_09022012\exemplos XML\arquivo.xml" ' open input file
    Set xml_in=session.CreateStream
    If Not xml_in.Open(filename$) Then
        MessageBox "Cannot open " & filename$,, "XML file error"
        Exit Sub
    End If
    If xml_in.Bytes = 0 Then
        MessageBox filename$ & " is empty",, "XML file error"
        Exit Sub
    End If

    Dim xml_out As NotesStream
```

```

        filename$ = "C:\Users\walison\Desktop\Ufmg-
T17_notebook_09022012\exemplos XML\ExemploArquivo.txt" ' create
output file

        Set xml_out=session.CreateStream

        If Not xml_out.Open(filename$) Then

            Messagebox "Cannot create " & filename$,, "TXT file
error"

            Exit Sub

        End If

        xml_out.Truncate

        Set saxParser=session.CreateSAXParser(xml_in, xml_out)

        On Event SAX_Characters From saxParser Call SAXCharacters
        On Event SAX_EndDocument From saxParser Call SAXEndDocument
        On Event SAX_EndElement From saxParser Call SAXEndElement
        On Event SAX_Error From saxParser Call SAXError
        On Event SAX_FatalError From saxParser Call SAXFatalError
        On Event SAX_IgnorableWhitespace From saxParser _
        Call SAXIgnorableWhitespace
        On Event SAX_NotationDecl From saxParser Call SAXNotationDecl
        On Event SAX_ProcessingInstruction From saxParser _
        Call SAXProcessingInstruction
        On Event SAX_StartDocument From saxParser Call SAXStartDocument
        On Event SAX_StartElement From saxParser Call SAXStartElement
        On Event SAX_UnparsedEntityDecl From saxParser Call
SAXUnparsedEntityDecl

        On Event SAX_Warning From saxParser Call SAXWarning

        saxParser.Process ' initiate parsing

End Sub

```

```

Sub SAXStartDocument (Source As Notessaxparser)

```

```

        Source.Output("Inicio do Documento"+Chr(13)+Chr(13))
End Sub

Sub SAXEndDocument (Source As Notessaxparser)
        Source.Output("Fim Documento" )
End Sub

Sub SAXCharacters (Source As Notessaxparser, Byval Characters As
String, _
Count As Long)

End Sub

Sub SAXEndElement (Source As Notessaxparser, Byval ElementName As
String)
        Source.Output("Fim Elemento=" +elementname+Chr(13)+Chr(13) )
End Sub

Sub SAXError (Source As Notessaxparser, Exception As
NotesSaxException )
        Source.Output ("Error - "+Exception.Message)
End Sub

Sub SAXFatalError (Source As Notessaxparser, Exception As
NotesSaxException)
        Source.Output ("FatalError - "+Exception.Message)
End Sub

Sub SAXIgnorableWhitespace (Source As Notessaxparser, _
Byval characters As String, Count As Long)
        MessageBox "Ignorable Whitespace found", MB_ICONINFORMATION
End Sub

```

```
Sub SAXNotationDecl (Source As Notessaxparser,Byval NotationName As String, Byval publicid As String, Byval systemid As String)
```

```
    MessageBox "Notation Declaration found", MB_ICONINFORMATION
```

```
End Sub
```

```
Sub SAXProcessingInstruction (Source As Notessaxparser, Byval target As String, Byval PIData As String)
```

```
    MessageBox "Processing Instruction found", MB_ICONINFORMATION
```

```
End Sub
```

```
Sub SAXStartElement (Source As Notessaxparser, Byval elementname As String, Attributes As NotesSaxAttributeList)
```

```
    Dim i As Integer
```

```
    'MessageBox "Start reading Element name = "+elementname, MB_ICONINFORMATION
```

```
    Source.Output("Elemento=" +elementname+Chr(13) )
```

```
    If Attributes.Length > 0 Then
```

```
        Dim attrname As String
```

```
        For i = 1 To Attributes.Length
```

```
            attrname = Attributes.GetName(i)
```

```
            Source.Output("        --        Atributo="+attrname+" Valor="+Attributes.GetValue(attrname)+",        Tipo =" +Attributes.GetType(attrname)+ Chr(13))        Next
```

```
        End If
```

```
End Sub
```

```
Sub SAXUnParsedEntityDecl (Source As Notessaxparser, Byval Entityname As String, Byval publicid As String, Byval systemid As String, Byval notationname As String)
```

```
    MessageBox "Unparsed Entity Declaration found", MB_ICONINFORMATION
```

```
End Sub
```

```
Sub SAXWarning (Source As Notessaxparser, Exception As NotesSaxException)
```

```
    Source.Output("Warning - "+Exception.Message)
```

```
End Sub
```

## APÊNDICE B

<b>Tipo Produção</b>	<b>Campo</b>	<b>Obrigatório</b>	<b>Objetivo</b>
Artigo em Periódico	Artigo em Periódico		
Artigo em Periódico	AreasConhecPub		Áreas do conhecimento
Artigo em Periódico	AutorAexAluFunCod		EXPORTAÇÃO
Artigo em Periódico	AutorCaractExterno		EXPORTAÇÃO
Artigo em Periódico	AutorCatGerus		EXPORTAÇÃO
Artigo em Periódico	AutorCtrCod		EXPORTAÇÃO
Artigo em Periódico	AutorDataNasc		EXPORTAÇÃO
Artigo em Periódico	AutorEmail		EXPORTAÇÃO
Artigo em Periódico	AutorNomeCientifico	Sim	Autores da publicação
Artigo em Periódico	AutorNomeCientificoBU		Nome Pessoal
Artigo em Periódico	AutorNomeCientificoBU_semCodigos		Nome Pessoal para visualização
Artigo em Periódico	AutorNomeCompleto		EXPORTAÇÃO
Artigo em Periódico	AutorOrdemPub		EXPORTAÇÃO
Artigo em Periódico	AutorSexo		EXPORTAÇÃO
Artigo em Periódico	AutorTipoPart		EXPORTAÇÃO
Artigo em Periódico	AutorURLPessoal		EXPORTAÇÃO
Artigo em Periódico	dc_CursosAssociados		Curso Vinculado a Autor
Artigo em Periódico	dc_CursosAssociadosCod		Curso Vinculado a Autor
Artigo em Periódico	dc_CursosInseridos		Curso Vinculado a Autor
Artigo em Periódico	dc_CursosPredefinidos		Curso Vinculado a Autor
Artigo em Periódico	dc_CursosPredefinidosCod		Curso Vinculado a Autor
Artigo em Periódico	dc_DeptosAssociados		Depto Vinculado a Autor
Artigo em Periódico	dc_DeptosAssociadosCod		Depto Vinculado a Autor
Artigo em Periódico	do_AutorizacaoAno		Ano autorização
Artigo em Periódico	do_AutorizacaoData		Data autorização
Artigo em Periódico	do_AutorizacaoDia		Dia autorização
Artigo em Periódico	do_AutorizacaoLiberacaoPessoa		Responsável liberação no Opus
Artigo em Periódico	do_AutorizacaoMes		Mês autorização
Artigo em Periódico	do_AutorizacaoRecebida		Autorização recebida
Artigo em Periódico	do_AutorizacaoResponsavel		Quem assinou a autorização
Artigo em Periódico	do_Comentarios		Comentários Obra
Artigo em Periódico	do_IdiomaResumo		Idioma do resumo em língua estrangeira
Artigo em Periódico	do_ListaDeAnexos		Lista de arquivos anexos
Artigo em Periódico	do_ListaDeAnexosWeb		Links web dos anexos
Artigo em Periódico	do_RefDocOriginal		Referência ao documento original
Artigo em Periódico	do_ResumoObra		Resumo da Obra
Artigo em Periódico	do_ResumoObraLinguaEstrangeira		Resumo em língua estrangeira
Artigo em Periódico	do_UnidDocRemovidos		Documentos removidos
Artigo em Periódico	IdiomaPub	Sim	Idioma da publicação
Artigo em Periódico	MeioPub	Sim	Meio de Publicação
Artigo em Periódico	NotasGeraisPub		Notas gerais



Artigo em Periódico	NumCadernoSecaoPub		Número do caderno ou seção
Artigo em Periódico	NumPaginas	Sim	Número de páginas
Artigo em Periódico	OpcoesTipoParticipacao		Tipo de Participação
Artigo em Periódico	PagFimPub	Sim	Página final
Artigo em Periódico	PagInicPub	Sim	Página inicial
Artigo em Periódico	PalavrasChavePub		Palavras-chave (separar com vírgula)
Artigo em Periódico	ProjetoLista		Projetos
Artigo em Periódico	ProjetoSub		Projetos
Artigo em Periódico	RevisaoBU		Revisão BU
Artigo em Periódico	SetoresAplic		Setores de Aplicação
Artigo em Periódico	SubTituloPub		Subtítulo
Artigo em Periódico	TipoTrabalhoPub	Sim	Tipo do trabalho
Artigo em Periódico	TitCadernoSecaoPub		Título do caderno ou seção
Artigo em Periódico	TituloPub	Sim	Título do artigo

## APÊNDICE C

O código desse apêndice tem como objetivo executar o processo de importação dos dados de um arquivo XML, exportado do Lattes, para o sistema Opus. Ele faz a leitura do documento, através da classe NotesSaxParse e captura apenas as informações a respeito dos artigos presente no arquivo XML.

```
'Declaração Global
'sessão desse agente
Dim session As NotesSession

'variavel que armazena os elementos desejados para pesquisa:
Dim arrayelemento(0 To 2) As String

'variavel que armazena os atributos desejados para pesquisa:
Dim arrayatributos(0 To 8) As String

'variavel que armazena os dados de um artigo que podera ser
exportado
Dim arrayArtigo() As String

'Contador de atributos no array
Dim contador As Integer

'Inicio de execução do programa:
Sub Initialize

    Set session = New NotesSession()

    Dim saxParser As NotesSAXParser
    Dim filename As String
    Dim xml_in As NotesStream

    contador=0

    'Elementos desejados no arquivo final a ser gerado
    arrayelemento(0) = "ARTIGO-PUBLICADO" 'Não mudar esse valor de
indice, está sendo usado no Evento EndElement
    arrayelemento(1) = "DADOS-BASICOS-DO-ARTIGO"
    arrayelemento(2) = "DETALHAMENTO-DO-ARTIGO"
    'arrayelemento(3) = "PALAVRAS-CHAVE"

    'Atributos desejados no arquivo final a ser gerado
    'jogando os valores em um array
    arrayatributos(0) ="TITULO-DO-ARTIGO"
    arrayatributos(1) ="IDIOMA"
    arrayatributos(2) ="MEIO-DE-DIVULGACAO"
    arrayatributos(3) ="PAGINA-INICIAL"
    arrayatributos(4) ="PAGINA-FINAL"
    arrayatributos(5) ="TITULO-DO-PERIODICO-OU-REVISTA"
    arrayatributos(6) ="ISSN"
    arrayatributos(7) ="VOLUME"
    arrayatributos(8) ="SERIE"
```

```

        filename$ = "D:\WALISON\ufmg-
T17\Monografia\LattesMirellaMoro.xml" ' open input file
    Set xml_in=session.CreateStream
    If Not xml_in.Open(filename$) Then
        MessageBox "Cannot open " & filename$,, "XML file error"
        Exit Sub
    End If
    If xml_in.Bytes = 0 Then
        MessageBox filename$ & " is empty",, "XML file error"
        Exit Sub
    End If

    Dim xml_out As NotesStream
    filename$ = "D:\WALISON\ufmg-
T17\Monografia\LattesMirellaMoro031.txt" ' create output file
    Set xml_out=session.CreateStream
    If Not xml_out.Open(filename$,"ISO-8859-1") Then
        MessageBox "Cannot create " & filename$,, "TXT file
error"
        Exit Sub
    End If
    xml_out.Truncate

    Set saxParser=session.CreateSAXParser(xml_in, xml_out)

    On Event SAX_Characters From saxParser Call SAXCharacters
    On Event SAX_EndDocument From saxParser Call SAXEndDocument
    On Event SAX_EndElement From saxParser Call SAXEndElement
    On Event SAX_Error From saxParser Call SAXError
    On Event SAX_FatalError From saxParser Call SAXFatalError
    On Event SAX_IgnorableWhitespace From saxParser _
    Call SAXIgnorableWhitespace
    On Event SAX_NotationDecl From saxParser Call SAXNotationDecl
    On Event SAX_ProcessingInstruction From saxParser _
    Call SAXProcessingInstruction
    On Event SAX_StartDocument From saxParser Call SAXStartDocument
    On Event SAX_StartElement From saxParser Call SAXStartElement
    On Event SAX_UnparsedEntityDecl From saxParser Call
SAXUnparsedEntityDecl
    On Event SAX_Warning From saxParser Call SAXWarning

    saxParser.Process ' initiate parsing

End Sub

Sub SAXCharacters (Source As Notessaxparser, Byval Characters As
String, Count As Long)
    '    MessageBox "Characters found", MB_ICONINFORMATION
End Sub

Sub SAXEndElement (Source As Notessaxparser, Byval ElementName As
String)

    Dim resultadopesquisa As String

```

```

'Chegou no final desse elemento e ele foi capturado
Dim var As Variant
var = Arraygetindex( arrayelemento, elementname )
If(var >= 0) Then

    'se for final de dados de um determinado artigo["ARTIGO-
PUBLICADO" ], verifica se esse existe para criar documento no Opus
    If(elementname=arrayelemento(0))Then

        resultadopesquisa = LocalizaArtigoOpus("5ADLNH",
"frmPubPeriodico", arrayArtigo(0) , arrayArtigo(4) )
        If(resultadopesquisa="")Then
            MessageBox "Cria Artigo"

            Call CriaArtigo_Periodico()
        Else
            MessageBox "Artigo Existe"
        End If

        'Zera o contador de atributos
        contador=0
    End If

End If

End Sub

Sub SAXError (Source As Notessaxparser, Exception As
NotesSaxException )
    MessageBox "Error - "+Exception.Message, MB_ICONINFORMATION
    Source.Output ("Error - "+Exception.Message)
End Sub

Sub SAXFatalError (Source As Notessaxparser, Exception As
NotesSaxException)
    MessageBox "FatalError - "+Exception.Message,
MB_ICONINFORMATION
    Source.Output ("FatalError - "+Exception.Message)
End Sub

Sub SAXIgnorableWhitespace (Source As Notessaxparser, Byval
characters As String, Count As Long)
    MessageBox "Ignorable Whitespace found", MB_ICONINFORMATION
End Sub

Sub SAXNotationDecl (Source As Notessaxparser, Byval NotationName As
String, Byval publicid As String, Byval systemid As String)
    MessageBox "Notation Declaration found", MB_ICONINFORMATION
End Sub

Sub SAXProcessingInstruction (Source As Notessaxparser, Byval target
As String, Byval PIData As String)
    MessageBox "Processing Instruction found", MB_ICONINFORMATION
End Sub

```

```
Sub SAXStartElement (Source As Notessaxparser,Byval elementname As String, Attributes As NotesSaxAttributeList)
```

```
Dim i As Integer  
Dim variante As Variant  
variante = Arraygetindex( arrayelemento, elementname )
```

```
'Pegar apenas os Element Name = ARTIGO-PUBLICADO , "DADOS-BASICOS-DO-ARTIGO" , "DETALHAMENTO-DO-ARTIGO"
```

```
'Se encontrar o Elemento do XML na variavel, imprima. Verificar o ENDelement
```

```
If( variante >= 0 )Then
```

```
    If Attributes.Length > 0 Then
```

```
        Dim attrname As String
```

```
        For i = 1 To Attributes.Length
```

```
            attrname = Trim( Attributes.GetName(i) )
```

```
            'identifica se o atributo correte está dentro da lista de atributos desejados para captura
```

```
            variante = Arraygetindex( arrayatributos, attrname )
```

```
            If( variante >= 0 )Then
```

```
                'Source.Output(" - Attribute "+ attrname+" = "+Attributes.GetValue(attrname) + Chr(13))
```

```
                Redim Preserve arrayArtigo(contador)
```

```
                arrayArtigo(contador) =
```

```
Attributes.GetValue(attrname)
```

```
                'controla a quantidade de atributos capturado no artigo
```

```
                contador = contador +1
```

```
            End If
```

```
        Next
```

```
    End If
```

```
End If
```

```
End Sub
```

```
Sub SAXUnParsedEntityDecl (Source As Notessaxparser, Byval Entityname As String, Byval publicid As String, Byval systemid As String, Byval notationname As String)
```

```
    MessageBox "Unparsed Entity Declaration found", MB_ICONINFORMATION
```

```
End Sub
```

```
Sub SAXWarning (Source As Notessaxparser, Exception As NotesSaxException)
```

```
    MessageBox "Warning - "+Exception.Message, MB_ICONINFORMATION
```

```
    Source.Output("Warning - "+Exception.Message)
```

**End Sub**

**Function LocalizaArtigoOpus(TipoPub As String, FormPub As String, TituloArtLattes As String, issnLattes As String) As String**

```
'Objetivo: Localizar uma produção na base de dados do Opus

Dim docPub As NotesDocument
Dim view As NotesView
Dim baseOpus As NotesDatabase
Dim query As String
Dim issnopus As Variant

Set baseOpus = session.GetDatabase("servidorDesenvUFMG", "Opus\opus.nsf")
Set view = baseOpus.GetView("vsProdBiblioTitulo") 'Producoes Bibliograficas Por Titulo

query = "FIELD Form CONTAINS "+FormPub+" AND FIELD Tipo CONTAINS "+TipoPub+" AND FIELD TituloPub CONTAINS "_
+TituloArtLattes+" AND FIELD StatusPub CONTAINS original "

Call view.FTSearch(query)

Set docPub = view.GetFirstDocument()
While Not docPub Is Nothing

'Retirando o traço do ISSN do Opus para comparar com o Lattes
issnopus = Evaluate({"@ReplaceSubstring("}& docPub.issn(0) &{" ;"-"; ""}))

'Se o nome da produção e o issn passado no parametro é igual ao do documento, produção existe:
If(docPub.TituloPub(0) = TituloArtLattes And issnopus(0) = issnLattes)Then
    LocalizaArtigoOpus = "Artigo: " + docPub.CodigoPub(0) + " Issn Opus: "+ docPub.issn(0)+ " Issn Curric.Lattes: "+ issnLattes +". "
    Exit Function
Elseif(docPub.TituloPub(0) = TituloArtLattes And issnopus(0) <> issnLattes)Then
    LocalizaArtigoOpus = "Artigo: " + docPub.CodigoPub(0) + ". Issn do Opus = "+ docPub.issn(0)+ " está diferente do Issn do Currc.Lattes= "+issnLattes+" ."
    Exit Function
Else
    LocalizaArtigoOpus = "Artigo: " + docPub.CodigoPub(0) + ". Issn do Opus ou Lattes está vazio. "
End If

Set docPub = view.GetNextDocument(docPub)

Wend

'Retorno vazio, não encontrou produção.
```

```
LocalizaArtigoOpus=""
```

**End Function**

**Function CriaArtigo\_Periodico()**

```
Dim db As NotesDatabase  
Dim viewRasc As NotesView  
Dim docRasc As NotesDocument  
Dim docxml As NotesDocument  
Dim dadosPeriodico As String
```

```
Set db = session.CurrentDatabase  
Set docxml = db.CreateDocument
```

```
Set viewRasc = db.GetView("vsRascunhos")  
viewRasc.FTSearch("WDSA-8R3SEN")  
Set docRasc = viewRasc.GetFirstDocument()
```

```
Call docRasc.CopyAllItems(docxml)
```

```
'Criar um artigo:
```

```
Call docxml.RemoveItem("CodigoPub") 'Para criar o codigo  
automaticamente após abrir Edição do Rascunho.
```

```
Call docxml.RemoveItem("dsp_URL_dspace") 'Para criar o codigo  
automaticamente após abrir Edição do Rascunho.
```

```
docxml.CadastranteOriginal= "Desenvolvimento LCC UFMG/ADM/UFMG"
```

```
docxml.StatusDoc="rascunho"
```

```
docxml.TituloPub= arrayArtigo(0)
```

```
docxml.PagInicPub= arrayArtigo(7)
```

```
docxml.PagFimPub=arrayArtigo(8)
```

```
docxml.MeioPub= arrayArtigo(2)
```

```
docxml.IdiomaPub=arrayArtigo(1)
```

```
'ESSE DOCUMENTO VAI TER PERÍODICO?
```

```
dadosPeriodico
```

```
LocalizaEdicaoPeriodicoOpus(arrayArtigo(3),arrayArtigo(4),  
arrayArtigo(5), arrayArtigo(6) ) 'TituloPeriodico, ISSN, Volume,  
Fasciculo ou Serie
```

```
If(dadosPeriodico<>"")Then
```

```
Messagebox "Criar Periodico"
```

```
Dim variante As Variant
```

```
variante = Evaluate({"@Explode("& dadosPeriodico &{"  
;";"})})
```

```
'Ordem do Explode é: codveiculo, TituloVeiculo, Issn,  
SedeRegistro/Sede do Periodico,CodigoInstancia, Volume, Serie,  
DataPub
```

```
'Dados do Periodico:
```

```
docxml.CodigoVeiculo = Trim(variante(0))
```

```
docxml.TituloVeiculo = Trim(variante(1))
```

```
docxml.TituloVeiculo_dsp = Trim(variante(1))
```

```
docxml.ISSN = Trim(variante(2))
```

```

docxml.ISSN_dsp = Trim(variante(2))
docxml.SedeRegistro =Trim( variante(3))
docxml.SedeRegistro_dsp =Trim( variante(3))
'Dados da Edição do Periodico
docxml.CodigoInstancia = Trim(variante(4))
docxml.VolumePub = Trim(variante(5))
docxml.VolumePub_dsp = Trim(variante(5))
docxml.FasciculoPub = Trim(variante(6))
docxml.FasciculoPub_dsp = Trim(variante(6))
docxml.DataPub = Trim( Left( Trim(variante(7) ) , 10) )
docxml.DataPub_dsp = Trim( Left( Trim(variante(7) ) ,
10) )
docxml.DataDesc = Trim(variante(8) )
docxml.DataDesc_dsp = Trim(variante(8) )
'Permite deixar visivel os Dados da Edição do Periodico,
de Inicial para Pesquisa
Call
docxml.ReplaceItemValue("StatusPesquisaInstancia","Pesquisa" )
Call
docxml.ReplaceItemValue("StatusPesquisaVeiculo","Pesquisa" )

Else
    MessageBox "Não Criar Periodico"
End If

Call docxml.Save(True,True)

```

#### **End Function**

**Function LocalizaEdicaoPeriodicoOpus(TituloPeriodicoLattes As String, issnLattes As String, volumeLattes As String, serieLattes As String) As String**

```

Dim docEdP As NotesDocument
Dim view As NotesView
Dim baseOpus As NotesDatabase
Dim query As String
Dim issnopus As Variant
Dim inteiro As Integer
Set baseOpus =
session.GetDatabase("servidorDesenvUFMG","Opus\opus.nsf")
Set view = baseOpus.GetView("vsEdicaoPeriodico") 'Visão de
Periódicos

query = "FIELD Form CONTAINS frmEdicaoPeriodico AND FIELD Tipo
CONTAINS 5DFPTA AND FIELD TituloVeiculo CONTAINS "_
+TituloPeriodicoLattes+ " AND FIELD VolumePub CONTAINS
"+volumeLattes+" AND FIELD StatusPub CONTAINS Original "
'Messagebox query
Call view.FTSearch(query)
'inteiro = view.AllEntries.Count
'Messagebox inteiro

Set docEdP = view.GetFirstDocument()
While Not docEdP Is Nothing

```



```

'Retirando o traço do ISSN do Periodico para comparar com
o Lattes
    issnopus = Evaluate({@ReplaceSubstring("& docEdP.issn(0)
&{" ;"-"; ""}})

'Se o nome da produção e o issn passado no parametro é
igual ao do documento, produção existe:
    If(docEdP.TituloVeiculo(0) = TituloPeriodicoLattes And
issnopus(0) = issnLattes And docEdP.VolumePub(0)=volumeLattes And
docEdP.FasciculoPub(0)=serieLattes)Then
        LocalizaEdicaoPeriodicoOpus =
docEdP.CodigoVeiculo(0) + " ; "+ docEdP.TituloVeiculo(0) + " ; "+
docEdP.issn(0) + " ; "+
        docEdP.SedeRegistro(0) + " ; "+
docEdP.CodigoInstancia(0) + " ; "+ docEdP.VolumePub(0) + " ; "+
docEdP.FasciculoPub(0) +
        " ; "+ Str(docEdP.DataPub(0) ) + " ; "+
(docEdP.DataDesc(0) )
        Exit Function
    ElseIf(docEdP.TituloVeiculo(0) <> TituloPeriodicoLattes
And issnopus(0) = issnLattes And docEdP.VolumePub(0)=volumeLattes
And docEdP.FasciculoPub(0)=serieLattes)Then
        LocalizaEdicaoPeriodicoOpus =
docEdP.CodigoVeiculo(0) + " ; "+ docEdP.TituloVeiculo(0) + " ; "+
docEdP.issn(0) + " ; "+
        docEdP.SedeRegistro(0) + " ; "+
docEdP.CodigoInstancia(0) + " ; "+ docEdP.VolumePub(0) + " ; "+
docEdP.FasciculoPub(0) +
        " ; "+ Str(docEdP.DataPub(0) )+ " ; "+
(docEdP.DataDesc(0) )
        Exit Function
    Else
        LocalizaEdicaoPeriodicoOpus = ""
    End If

    Set docEdP = view.GetNextDocument(docEdP)

Wend

'Retorno vazio, não encontrou produção.
LocalizaEdicaoPeriodicoOpus= ""

End Function

```