

## Implementação de uma Ferramenta de Integração de Dados Aplicado a Dados Meteorológicos

Marcelo L. Gross, José E. C. Soares, Alexandre T. Lazzaretti

IFSUL – Instituto Federal Sul-rio-grandense Campus Passo Fundo

Estrada Perimetral Leste, 150 – CEP: 99064-440 Passo Fundo (RS)

{marcelolgross, joseericosoares}@gmail.com,  
alexandre.lazzaretti@passofundo.ifsul.edu.br

**Abstract.** *Some contexts of human society, there is data stored and available in various forms and formats. The use of data is not easy, because they need to be transformed and integrated, to after to be manipulated. The agricultural sector is very important in global economic and social context and is strongly influenced by meteorological variables. Thus, this work aims to show the implementation and design a tool to integrate weather data with the database.*

**Resumo** *Em diversos contextos da sociedade humana, existem dados armazenados e disponíveis em diversas formas e formatos. A utilização desses dados é extremamente trabalhosa, pois precisam ser transformados e integrados para poderem ser manipulados. O setor agrícola possui considerável importância no contexto econômico e social mundial e é influenciado fortemente por variáveis meteorológicas. Nesse sentido, esse trabalho tem por objetivo mostrar a implementação e o projeto uma ferramenta para integração de dados meteorológicos com banco de dados.*

### 1. Introdução

O processo de integração de dados é importante pelo fato de dados oriundos de diferentes fontes poderem ser manipulados de forma única dentro de um determinado domínio de aplicação (ALVARO, 2001).

Na agricultura os modelos de simulação de culturas são capazes de prever o rendimento final de grãos e também representar a simulação da dinâmica do crescimento das culturas através da integração numérica (GRAVES et al., 2002). A utilização de modelos de simulação tenta prever riscos e estimar tendências, e tem se tornado um importante aliado para a tomada de decisão (DONATELLI et al., 2002; SINCLAIR & SELIGMAN, 1996). Para a comunidade científica, tais modelos têm auxiliado na organização do conhecimento e em testes de hipóteses. Os modelos de simulação de culturas necessitam de dados meteorológicos de entrada e após a execução do simulador, geram dados de saída. Dependendo do tipo de simulação executada, faz-se necessário trabalhar com uma grande quantidade de dados em diversos formatos, tanto de entrada quanto de saída. Sem os mecanismos adequados de manipulação e armazenamento, esse trabalho torna-se difícil, ou até mesmo impossível de ser realizado.

No entanto, atualmente poucas ferramentas deste cunho estão disponíveis para realizar este trabalho de integração dos dados e estas normalmente, possuem limitações quanto ao seu uso. Um exemplo é a ferramenta *BPEL Process Manager*, desenvolvida pela empresa Oracle. Essa ferramenta possui integração com as ferramentas da mesma

empresa, e apresenta empecilhos quando aplicada a outros ambientes.

Algumas ferramentas realizam a integração de dados através do mapeamento de esquemas XML (HARA *et. al.*). Também existem ferramentas de integração num contexto ETL (*Extract Transform and Load*). Elas recebem dados de diferentes formatos nas etapas de extração, e realizam mapeamentos semânticos, cálculos de valores, uniões, agregação, validação de dados dentre outras atividades na etapa de transformação. Geralmente os dados de saída na etapa de carga dessas ferramentas é feita por uma aplicação de *data warehouse*.

Dentre as ferramentas de ETL open source se destacam a *Pentaho BI Suite* e a *Talend Data Integration*. Essas ferramentas possuem uma arquitetura semelhante a proposta nesse trabalho principalmente na etapa de transformação. Entretanto, não se adaptam ao contexto agrícola aqui proposto, pois nesse caso pretende-se usar as instruções de regras de mapeamento geradas em processos *batch*, além de gerar diversos formatos de saída, inclusive com conexão com banco de dados.

Assim sendo, este trabalho tem por objetivo apresentar o protótipo de uma ferramenta para integração genérica de dados meteorológicos, apresentando a sua arquitetura de funcionamento, pois já foi parcialmente implementado.

## 2. Materiais e Métodos

### 2.1 Dados Meteorológicos

O DSSAT (*Decision Support System for Agrotechnology Transfer*) é um modelo de simulação do desenvolvimento e crescimento de culturas (DSSAT, 2014), e entre os dados usados como entrada pelos modelos de simulação de culturas está este tipo. As variáveis meteorológicas diárias usadas no arquivo são: radiação solar (SRAD), temperatura máxima do ar (TMAX), temperatura mínima do ar (TMIN) e precipitação pluvial (RAIN).

### 2.2 Tecnologias

Desejava-se criar uma ferramenta Web, por isso foi escolhida a linguagem PHP (*PHP Hypertext Preprocessor*) como linguagem de desenvolvimento, a qual gera um código HTML (*HyperText Markup Language*) para apresentação ao usuário. Também foi usada a linguagem Javascript para interações com o usuário dentro da interface da ferramenta.

Para o tratamento dos metadados foi utilizada a tecnologia XML (*eXtensible Markup Language*). De acordo com o *World Wide Web Consortium* (W3C), XML foi originalmente projetado para vir ao encontro dos desafios da publicação eletrônica em larga escala e tem um importante papel na exportação e troca de dados. Desta forma, num documento XML são armazenados os metadados que correspondem aos arquivos de entrada e saída gerados pela ferramenta implementada.

## 3. Resultados e Discussão

A ferramenta de integração desenvolvida tem o propósito de ser genérica, porém devido ao escopo do projeto, inicialmente realizou-se à integração somente com dados meteorológicos no padrão do sistema DSSAT.

Foi desenvolvido um assistente na linguagem PHP, na qual o usuário apenas comunicaria os detalhes e a aplicação faria a geração automatizada dos metadados em XML, e as validações da interface gráfica seriam feitas com auxílio de Javascript e JQuery. Na interface gráfica da ferramenta inicialmente é feito um pedido ao usuário se ele já possui um arquivo XML com os metadados salvos, e se o usuário não o possuir, ele poderá entrar no modo assistente para fazer a geração.

O assistente gráfico por sua vez ficou dividido em 3 partes: “Dados de Entrada”, “Metadados e Arquivos”, e “Dados de saída”, e as partes são de livre navegação. Para a criação correta dos metadados, o usuário informa corretamente o número de tabelas e o número de campos de cada tabela envolvida no armazenamento dos dados, e é gerada uma janela modal personalizada para o pedido, com todas as tabelas e seus respectivos campos informados, que basta serem preenchidos com informações sobre cada campo. Ao pedir a geração dos XML com os metadados, todos os dados temporários são enviados por meio de um *array* pelo método *POST* para os arquivos do motor XML, que cria o documento XML com todos os metadados para a conversão dos arquivos de origem para a saída desejada.

Para o armazenamento dos metadados (dados sobre as entradas e saídas da ferramenta), foi criado um *template* XML, onde é possível armazenar dados sobre o formato, cabeçalho, escolher o tipo de saída (CSV ou na linguagem SQL) dos dados de entrada.

Na Figura 1 é mostrado um exemplo do documento XML gerado pela ferramenta, com base no arquivo de dados meteorológicos armazenados no arquivo WTH do modelo DSSAT.

```
<root>
  <cabecalho>
    <numeroLinhas>5</numeroLinhas>
    <indicador>@</indicador>
  </cabecalho>
  <arquivo>
    <nomeProfile>dados_meteorologicos</nomeProfile>
    <delimitador></delimitador>
    <nomeSaida>Insert</nomeSaida>
    <formatoSaida>sql</formatoSaida>
  </arquivo>
  <tabelas>
    <tabela sequence="0" generator="Y" name="weather_data_variable" pk="id">
      <campo sequence="0" type="double" value="" notNull="Y">id</campo>
      <campo sequence="1" type="double" value="" notNull="Y">weather_data</campo>
      <campo sequence="2" type="integer" value="" notNull="Y">variable</campo>
      <campo sequence="3" type="char(1)" value="" notNull="Y">data_type</campo>
      <campo sequence="4" type="date" value="" notNull="Y">data</campo>
      <campo sequence="5" type="time" value="" notNull="Y">time</campo>
      <campo sequence="6" type="tp_data" value="" notNull="Y">data_value</campo>
    </tabela>
    <tabela sequence="1" generator="Y" name="weather_data" pk="id">
      <campo sequence="0" type="double" value="" notNull="Y">id</campo>
      <campo sequence="1" type="double" value="" notNull="Y">date_import</campo>
      <campo sequence="2" type="time" value="" notNull="Y">time_import</campo>
      <campo sequence="3" type="integer" value="" notNull="Y">io</campo>
      <campo sequence="4" type="integer" value="" notNull="Y">station</campo>
    </tabela>
  </tabelas>
</root>
```

**Figura 1 - Exemplo de documento XML, gerado pelo framework, com os metadados do arquivo WTH de entrada.**

A *tag* “cabeçalho” armazena informações sobre o cabeçalho do arquivo de entrada, neste caso informando o número de linhas do cabeçalho e qual o caractere identificador do mesmo, pois num mesmo arquivo podem existir diversos cabeçalhos. A *tag* “arquivo” contém informações necessárias para a geração dos arquivos de saída da ferramenta, como o nome do arquivo a ser gerado, o delimitador de dados e o formato de saída dos dados.

A *tag* “tabelas” descreve os metadados referente as tabelas do banco de dados, para onde serão feitas as importações, e pode conter uma ou várias *tag* “tabela”. Cada uma descreve como serão mapeados os campos da tabela do banco de dados para os campos do arquivo de entrada, que é feito pela *tag* campo.

#### 4. Conclusões

Com toda uma base operacional já pronta, e com o domínio de várias tecnologias, foi criada uma interface gráfica no formato de assistente, que salva os metadados escolhidos no XML, para a posterior execução. No entanto, a ferramenta não está pronta. Como trabalhos futuros, falta a geração dos arquivos de saída, fazendo a leitura dos arquivos de dados na memória para a interpretação e geração da saída. Também faltam testes com outros formatos de arquivos.

#### Referências

- Consortium, World Wide Web. XML: Extensible Markup Language. Massachusetts, 2011. Disponível em: < <http://www.w3.org/XML/>>.
- Dssat. Decision Support System for Agrotechnology Transfer. Disponível em: <<http://dssat.net>>. Acesso em 04/09/2014.
- Donatelli, M.; Ittersum, V.; Bindi M.; Porter, J. R.; Modelling cropping systems: Highlights of the symposium and preface to the special issues. Eur. J. Agron., v 18:1-11. 2002.
- Graves, A.R.; Mathheus, H.T.; Stephens, R.B.; Middleton, W.T. Crop simulation models as tools in computer laboratory and classroom-based education. J. Nat. Resour. Life Sci. Educ., 31:48-54. 2002 .
- Hara, Carmem Satie, Ruthes, Eduardo da Rocha, Scopim, Kemmel da Silva, Sunyé, Marcos Sfair. Jintegrator: A Heterogeneous Database Integration Tool. International Conference on Information Systems and Technology Management, CONTECSI. 2006
- Sinclair, T. R.; Seligman, N.G.; Crop modeling: From infancy to maturity. Agron. J. 88:698-704. 1996.