

TEMPTOOL – UMA FERRAMENTA PARA A MODELAGEM TEMPORAL DOS DADOS

**Eugênio de O. Simonetto¹, Cristiano R. Moreira, Duncan D. A. Ruiz²,
Luciano C. Ost¹, Glauco B.V.Santos¹**

RESUMO

O trabalho apresenta uma ferramenta onde o projetista ou o usuário responsável pelas informações temporais de um sistema de informação possa obter todas as estruturas necessárias para a representação do aspecto tempo em seu banco de dados. A ferramenta toma como base a definição do esquema inicial do banco de dados (após a conclusão do projeto de implementação e antes da carga no SGBD). Essa abordagem é baseada no modelo proposto por [Sim98, Rui98]. Existe um projeto (em andamento), para tratar do aperfeiçoamento da ferramenta, para que ela apoie a modelagem temporal dos dados desde o desenvolvimento do modelo conceitual do sistema.

Introdução

Nos últimos anos diversos modelos de dados temporais foram propostos, traduzindo a preocupação da comunidade científica com a representação temporal na modelagem de dados.

Nos sistemas de bancos de dados relacionais convencionais, os dados estão armazenados em tabelas de duas dimensões, onde as linhas são as tuplas e as colunas os atributos [Cod70, Sil99]. Ao introduzir-se o fator tempo associado ao dado estas tabelas podem ser vistas como possuindo mais uma dimensão, o tempo [Tan90].

Para a implementação de bancos de dados temporais em SGBD's relacionais existem duas abordagens. A primeira é estender a semântica do modelo relacional para a incorporação do fator tempo. A outra é implementar o banco de dados temporal sobre o modelo relacional básico com o tempo aparecendo como atributos adicionais. Como exemplo da primeira abordagem temos os modelos relacionais estendidos HRDM [Cli93], TRM [Nav93], HSQL [Sar93] e TSQL2 [Sno95].

Na segunda abordagem o tempo não é suportado completamente pelo SGBD, cabendo ao projetista do sistema a tradução de consultas e atualizações que envolvam tempo para a linguagem de manipulação de dados.

A incorporação do aspecto tempo nos SGBDs relacionais, conforme a segunda abordagem, pode ser feita pela inclusão de atributos adicionais que representem o início e (ou) o fim do tempo de transação ou do tempo de validade [Sno95]. A proposta feita pelo modelo de [Sim98, Rui98] adiciona os atributos de início e fim de tempo de transação, mas as atualizações são feitas pelos usuários convencionais do banco de dados. Também o modelo, não estende a semântica do SGBD para o tratamento da informação temporal; ele utiliza somente as funcionalidades do SGBD para o tratamento de tais informações.

2. O Modelo Temporal de [Sim98, Rui98]

Para a construção do modelo, buscou-se utilizar uma maneira pela qual o usuário do banco de dados não manipulasse diretamente o dado temporal, ou seja, que ele continuasse visualizando

¹ Pontifícia Universidade Católica do Rio Grande do Sul - FACI – Departamento de Informática
Uruguaiana – RS – Brasil - E-mail: [eugenio, lco, glauco]@puccrs.campus2.br

² Pontifícia Universidade Católica do Rio Grande do Sul - FACIN - Porto Alegre-RS-Brasil - E-mail:
duncan@inf.puccrs.br

(e manipulando) o banco de dados como uma base de dados instantânea. Pelas formas possíveis de representação temporal em um banco de dados, concluiu-se que no modelo a ser proposto o aspecto temporal teria de ser representado pelo tempo de transação, pois utilizando-se esta linha de tempo o dado temporal não precisaria ser manipulado diretamente pelo usuário, visto que ele utiliza o tempo do SGBD.

Para a representação dos aspectos temporais dos dados utilizamos o conceito de estados do banco de dados [Sar93], onde cada estado é definido pelo valor de um atributo. Qualquer alteração efetuada no valor do atributo gera um novo estado no banco de dados. Cada estado possui uma determinada duração, que é representada por um intervalo de tempo no qual não é feita nenhuma alteração no valor do atributo. Portanto, cada instância do banco de dados é rotulada com um intervalo temporal.

Neste trabalho, optou-se pela representação temporal na forma de intervalos de tempo. Portanto, para representar o tempo de transação são acrescentados dois atributos adicionais: início do período de transação (I_{TT}) e final do tempo de transação (F_{TT}).

Em bancos de dados temporais, as operações de atualização devem ser tratadas de forma diferente das atualizações em bancos de dados convencionais, pois devem garantir que nenhum dado, que se torne antigo, não venha a ser perdido. Portanto, todas as atualizações devem ser seguidas de operações para manutenção do valor antigo. Para tais atualizações foram desenvolvidas rotinas que atualizam automaticamente o tempo de transação. Estas rotinas ficam totalmente transparentes aos usuários convencionais do banco de dados.

A granularidade de tempo utilizada no modelo pode ser definida pelo projetista do sistema a ser implementado. Para efeitos de validação do trabalho, a granularidade utilizada foi a de *segundos*.

3. TempTool - A Ferramenta Proposta

Em bancos de dados temporais, as operações de atualização devem garantir que nenhum dado, que se torne antigo, venha a ser perdido. Portanto, todas as atualizações devem ser seguidas de operações para manutenção do valor antigo. No modelo proposto [Sim98, Rui98], para tais atualizações, foram desenvolvidas rotinas (utilizando-se o conceito de *triggers* dos SGBDs) que atualizam automaticamente o tempo de transação nas tabelas componentes do modelo. A construção manual das estruturas componentes, propostas neste modelo, é exaustiva e difícil, pois é muito grande a quantidade de linhas de código, entre gatilhos e tabelas, a serem produzidos.

Visto esta dificuldade, estamos desenvolvendo uma ferramenta para apoiar esta etapa do desenvolvimento, em conformidade com o modelo de [Sim98, Rui98]. A utilização da ferramenta se dá após a conclusão do projeto de implementação do banco de dados via uma ferramenta CASE, e antes da carga de especificações no SGBD. O DB-MAIN [DbW] é uma ferramenta CASE que opera desta maneira: ao final do projeto, a geração do esquema do banco de dados é feita em um arquivo texto, chamado de *script* do BD. Posteriormente, o DBA *carrega* o *script* no SGBD. Modificar tal *script*, portanto, é algo que não implica em alterar as rotinas de trabalho das equipes de desenvolvimento de aplicações; acrescenta mais um passo nessas rotinas, o que é inevitável caso se queira a incorporação de aspectos temporais nos BDs das aplicações.

A parte que trata do reconhecimento do *script* em SQL e a geração das novas tabelas e rotinas já estão desenvolvidas, o nosso próximo passo é o desenvolvimento de uma parte gráfica na ferramenta, que proporcionará ao projetista do banco de dados desenvolver seu modelo conceitual diretamente na ferramenta.

3.1 A Especificação da Ferramenta

Para a especificação da ferramenta de apoio ao desenvolvimento do modelo de dados, conforme o modelo temporal de [Sim98, Rui98], procuramos identificar as diferentes etapas que

constituem o processo de desenvolvimento do modelo e, que poderiam vir a ser apoiadas por algum tipo de ferramenta computacional. Foram identificadas as seguintes etapas :

1. *Análise do Esquema Inicial;*
2. *Classificação dos Atributos;*
3. *Criação das Estruturas do Modelo.*

Após a execução destas etapas é obtido o *script* final, conforme o modelo proposto, para ser submetido a um determinado SGBD. Este *script* é composto por todos os gatilhos e tabelas derivados do esquema inicial do banco de dados. De todas as etapas, a única que sofrerá a interferência do usuário é a de classificação dos atributos, onde estes serão classificados como temporais ou estáticos, conforme o ponto de vista (ou a necessidade) do projetista do sistema.

3.1.1 - Análise do Esquema Inicial

A primeira etapa envolve o reconhecimento do banco de dados a ser submetido à ferramenta. Nesta etapa, serão identificadas todas as tabelas componentes do banco de dados e, dentro de cada tabela todos os seus atributos, bem como sua chave primária (*primary key*). Esta etapa tem por finalidade preparar os atributos para sua posterior classificação, a qual irá ocorrer na etapa seguinte a esta.

O esquema do banco de dados a ser reconhecido pela ferramenta, deverá ser fornecido através de um arquivo *script*, gerado por uma ferramenta CASE ou pelo projetista do sistema .

Após esta primeira etapa, onde são feitos os reconhecimentos nas estruturas a serem submetidas a TempTool, é então iniciada a segunda etapa denominada classificação dos atributos.

3.1.2 - Classificação dos Atributos

Nesta etapa, o projetista do sistema deve identificar dentro de cada tabela, a classificação de cada atributo (estático ou temporal) componente desta, conforme o modelo em proposição. Esta é a única etapa que acontece a intervenção humana no funcionamento da ferramenta.

Em uma tabela, além dos atributos componentes da chave primária, poderão ter nenhum ou vários atributos temporais, e nenhum ou vários atributos estáticos.

Na interface para classificação dos atributos (ver figura 3.1), a ferramenta disponibiliza ao projetista do sistema todos os atributos de cada tabela componente do sistema, para estes serem submetidos a classificação. Após a classificação dos atributos, é dada ao usuário a opção **Script**, a qual disponibilizará ao usuário o novo *script* (produto final da ferramenta), em conformidade com o modelo temporal de [Sim98, Rui98], para ser submetido ao SGBD. Nesta etapa ainda, a ferramenta questiona ao usuário para qual SGBD ele deseja gerar o *script*, visto que a sintaxe de definição de *triggers*, varia de um SGBD para outro.

Após a classificação dos atributos por parte do projetista do sistema se dá a parte de criação das estruturas, conforme a descrição do modelo.

3.1.3 - Criação das Estruturas do Modelo

Nesta última etapa de funcionamento da ferramenta, é necessário que sejam construídas as estruturas que garantirão a manutenção e o armazenamento automático do dado temporal do modelo em questão.

Conforme a classificação dos atributos, executada na fase anterior, são determinadas as estruturas componentes da ferramenta, estas novas estruturas implicam na criação de *triggers*, procedimentos e de novas tabelas, na forma de um arquivo *script*.

A primeira estrutura a ser gerada pela ferramenta é a relação definida pelo usuário, que no modelo proposto chama-se *tabela instantânea*. Para cada atributo classificado como **temporal**, é gerada uma *nova tabela* e *rotinas de manutenção do dado temporal* que manipulem este tipo de dado a cada inserção, atualização e remoção do atributo em questão.

Já para os atributos classificados como **estáticos** é definida *somente uma tabela* para todos os atributos classificados desta maneira na relação em análise, bem como suas respectivas *rotinas de manutenção do dado temporal*, no caso de inserção e remoção (pois o atributo estático não varia com o correr do tempo) de algum atributo. O modo como o novo *script* é apresentado pela ferramenta pode ser visualizado na figura 3.2

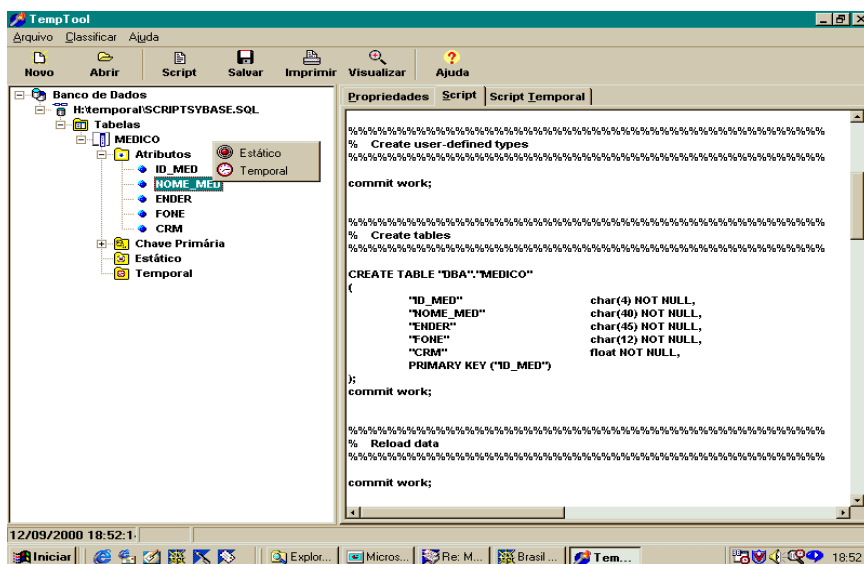


Figura 3.1 – Interface para a Classificação de Atributos

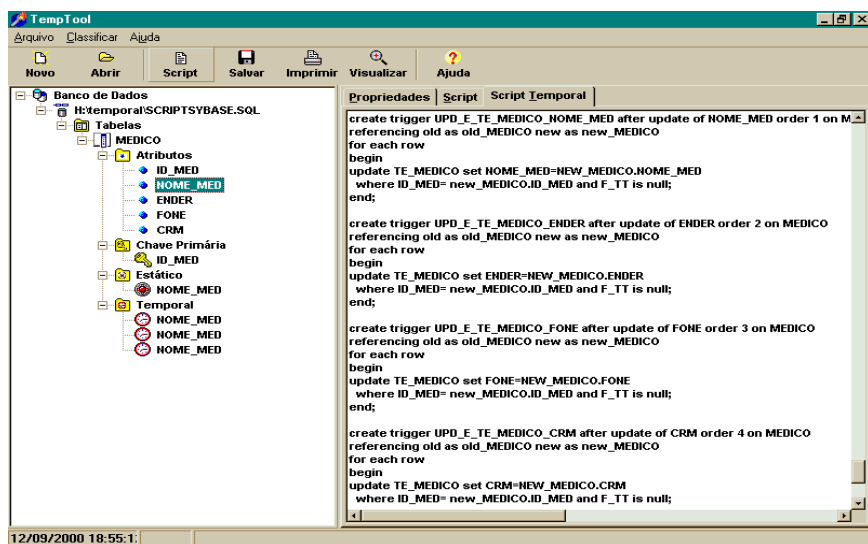


Figura 3.2 - Interface para a Apresentação do Script Temporal

4. Conclusões e Trabalhos Futuros

Neste trabalho foi apresentada uma ferramenta para o desenvolvimento e manutenção de bases de dados temporais, denominada TempTool. A TempTool permite ao projetista de sistemas de informação, após a concepção do projeto de implementação do banco de dados e, antes da carga em algum SGBD de primeira linha, tais como Oracle, Sybase e Ingres, gerar um novo modelo de dados com as estruturas necessárias ao armazenamento e manutenção dos dados que irão variar com o correr do tempo.

Os resultados obtidos utilizando-se a ferramenta foram considerados excelentes, visto que "transformaram" fielmente os *scripts* originais dos modelos de dados dos usuários, em modelos de dados temporais.

Na primeira etapa do desenvolvimento da ferramenta, para fins de validação, implementamos apenas o reconhecimento e a geração do *script* para os SGBDs Sybase, Oracle e Ingres. Como próximo passo no desenvolvimento da ferramenta, iremos desenvolver a geração de *script* para os SGBDs Informix e SQL Server. Após o desenvolvimento para estes SGBDs, iremos desenvolver um método pelo qual a ferramenta disponibilize mecanismos para que o usuário conceba seu diagrama E-R diretamente na TempTool. Assim, os atributos já serão classificados na etapa da modelagem conceitual do sistema, para posteriormente ser gerado o *script* final do banco de dados temporal.

Referências Bibliográficas

- [Cod70] CODD, E.F. A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM*, v.13, n.6, 06/1970.
- [DbW] DB-MAIN Case. **Db-Main Tutorial**. <http://www.info.fundp.ac.be/~dbm>.
- [Jen94] JENSEN, C. et al. **A Consensus Glossary of Temporal Databases Concepts**. Sigmod Record. v.23, n.1, p.53-63, March, 1994.
- [Nav93] NAVATHE, S.; AHMED, R. Temporal Extensions to Relational Model and SQL. In: *Temporal Databases: Theory, Design and Implementation*. Benjamin / Cummings, 1993.
- [Rui98] RUIZ, D.D.; SIMONETTO, E.O. **Uma Proposta para Incorporação de Aspectos Temporais em SGBDs Relacionais**. In :13o.SBBD (1998 : MARINGÁ - PR). **Anais...** Maringá- PR : DIN - UEM, 1998. p.151-165.
- [Sar93] SARDA, N.L. A Historical Query Language. In: *Temporal Databases: Theory, Design and Implementation*. Benjamin/Cummings, 1993.
- [Sil99] SILBERSCHATZ, A.; KORTH, H.; SUDARSHAN, S. *Sistemas de Banco de Dados*. Makron Books, 1999.
- [Sim98] SIMONETTO, E.O.; RUIZ, D.D. **Proposta de um Modelo para Incorporação de Aspectos Temporais em SGBDs Relacionais**. In :IIISIMPÓSIO NACIONAL DE INFORMÁTICA (1998 : SANTA MARIA-RS). **Anais...** Santa Maria-RS : UniFran, 1998. p.108-112.
- [Sno95] SNODGRASS, R.T. **The TSQL2 Temporal Query Language**. Boston: Kluwer Academic Publishers , 1995.
- [Tan90] TANSEL, A.U. Modelling Temporal Data. *Information and Software Technology*. V. 32, n. 8, p.514-520, 1990.