Levantamento de Modelos de Dados em Sistemas Legados

Nuno Palmeiro Ribeiro
Alberto Bigotte de Almeida
DAMAG

Fernando Brito e Abreu
Pedro Sousa
INESC

Resumo

Um conhecimento detalhado do modelo de dados dos sistemas de informação nas organizações, tanto ao nível conceptual como aos níveis lógico e físico, é fundamental para permitir a sua evolução.

A evolução é, obviamente, tanto mais difícil quanto menos conhecimento do modelo de dados se tem. Sem esse conhecimento, pequenas alterações efectuadas poderão ter consequências imprevisíveis, conduzindo ao aumento da dimensão, da redundância e da complexidade em geral do sistema.

A obtenção do modelo conceptual, a partir do catálogo do Sistema de Gestão de Bases de Dados (SGBD´s) é uma funcionalidade que os fabricantes de inúmeras ferramentas dizem ser possível efectuar de uma forma automática. Porém, na prática,

estas são incapazes de lidar com a realidade dos SGBD's legados, que foram normalmente criados sem regras de nomes ou sem declarações de chaves, entre outros.

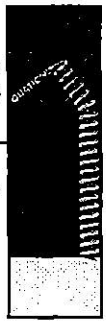
Nesta comunicação, apresentamos um método para a recuperação do modelo de dados, bem como um caso de estudo aplicado ao Sistema Integrado de Informação do Pessoal da Marinha Portuguesa, onde se descrevem as principais dificuldades e resultados obtidos.

Palavras chave: *Reverse Engineering*, Qualidade, Sistemas Legados, Bases de Dados

1. Introdução

É inegável nos dias de hoje, a necessidade de evoluir os sistemas de informação de acordo com os objectivos estratégicos delineados pelas organizações. A qualidade dos sistemas de informação é um requisito fundamental na redução dos custos inerentes à sua evolução.

O conhecimento do modelo de dados é um forte contributo para a qualidade do sistema, não só porque nos permitir avaliar o impacto de futuras alterações necessárias para o cumprimento do planeamento dos sistemas de informação estratégicos da organização, mas também porque, entre outras coisas, nos oferece valiosas indicações sobre dados não normalizados, facilita futuras migrações para outras plataformas tecnológicas e permite detectar incoerências funcionais durante as execuções dos ensaios. O levantamento do modelo de dados conceptual (entidade-associação) a partir



dos catálogos dos SGBD´s relacionais é uma actividade fundamental para a manutenção dos sistemas onde estes não são conhecidos.

O Sistema Integrado de Informação de Pessoal da Marinha Portuguesa, doravante designado por SIIP, foi um sistema criado na década de 80. Inicialmente este sistema utilizava ficheiros VSAM. Com o aparecimento, na Marinha, das bases de dados relacionais, no caso através plataforma SQL/DS, surgiu a oportunidade de mudar a filosofia da gestão de modo a dar resposta às novas necessidades. O sistema começou a ser desenvolvido com base no SSADM, um método de análise de sistemas recomendado pela Administração Pública no Reino Unido para desenvolver grandes sistemas. Inicialmente apenas se considerou a repartição de oficiais, porque estava directamente relacionada com a equipa de desenvolvimento. Mais tarde, este facto acarretou alguns problemas na integração das outras repartições de pessoal (sargentos, praças e civis), uma vez que a gestão do respectivo pessoal era específica a cada repartição.

Aparentemente, o SIIP não foi planeado de acordo com o sentido que actualmente atribuímos ao planeamento de sistemas de informação. Foi desenvolvido com base na sensibilidade dos problemas da gestão dos elementos da equipa de desenvolvimento, e não por influência ou indicações dos utilizadores das diversas repartições. Aliado a este aspecto está o facto de, nesse tempo, a versão disponível do SQL/DS não permitir a definição física e explícita de chaves primárias e estrangeiras, apenas se conseguindo implementar identificadores únicos.

Hoje não existem documentos actualizados e precisos que descrevam os modelos de dados do SIIP. Algumas das causas desta situação são:

- grande parte dos elementos que estiveram directamente envolvidos no desenvolvimento já não se encontram na Marinha, tendo-se perdido muita da informação que levou à criação do sistema tal como ele se apresenta actualmente;

- rotatividade do pessoal, sem existência de uma política de actualização compulsiva dos modelos;

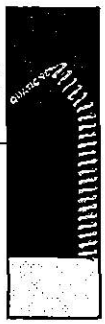
- não existirem mecanismos de verificação da rastreabilidade sincrónica.

Para inverter esta situação, o levantamento do modelo de dados é de extrema importância. É este que permite que os novos elementos possam facilmente entender como o sistema está desenvolvido, sendo um elemento fundamental para qualquer acção de manutenção e/ou evolução do sistema, de forma a alargar o seu tempo de vida útil.

O cenário descrito não é exclusivo da Marinha. Infelizmente, é algo generalizado em muitas organizações [McClure92].

A recuperação do modelo de dados parece, à primeira vista, uma tarefa pouco complexa e, de certa forma, rápida. A existência de várias ferramentas de reengenharia no mercado destinadas a este efeito [Furlan94] contribuem para esta forma de pensar. Porém, a tarefa não é tão simples. As ferramentas existentes baseiam-se, normalmente, apenas no catálogo da base de dados. De facto, assumem entre outras coisas que não existam homónimos (atributos com nomes iguais mas significados diferentes) e sinónimos (atributos com nomes diferentes e significados iguais) e assumem que não existem fragmentações de tabelas. Contudo estas situações existem na prática, mostrando-se essas ferramentas inadequadas no tratamento da maioria dos casos reais.

Para a obtenção do modelo de dados é necessário uma abordagem mais rigorosa e abrangente, na qual é necessário consultar para além do catálogo da base de dados, o código fonte das aplicações que acedem aos dados, os dados e os próprios utilizadores.



2. Metodologia

A metodologia que agora se descreve para a recuperação do modelo de dados fundamenta-se na detecção de chaves primárias[1], chaves estrangeiras[2] das tabelas e na decomposição do problema inicial em problemas mais simples.
Desta forma, a metodologia adoptada seguiu as seguintes fases:

- identificação de chaves primárias;
- agrupamento de tabelas em entidades abstractas;
- especificação de grupos;
- identificação de generalizações e fragmentações;
- detecção das restantes associações.

2.1 Identificação das chaves primárias

Nesta primeira fase pretendemos identificar as chaves primárias de todas as tabelas, bem como conhecer as dependências entre os atributos das várias chaves. Por exemplo, no caso da chave primária de uma tabela ser NRBI e o de outra ser BI, teremos de averiguar se se trata de um atributo com o mesmo conteúdo semântico em ambas as tabelas, apesar dos nomes serem diferentes, isto é, investigar se são sinónimos.

Na ausência de identificação de chaves primárias no catálogo dos SGBD's, recorremos às restrições de índice único, assumindo o atributo ou conjunto de atributos

[1] As chaves primárias são o conjunto de atributos que identificam, de forma exclusiva, cada um dos elementos de uma tabela.
[2] As chaves estrangeiras são os atributos de uma tabela que referem outra tabela, onde são chave primária.

referentes a cada uma das tabelas como sendo a sua chave primária. Nos casos em que não existem índices únicos declarados no catálogo, a chave terá de ser identificada por inspecção aos dados ou ao código fonte, onde se fazem as inserções dos dados nessas tabelas. No caso de existir mais de um índice único declarado no catálogo, surgem duas opções. Na primeira, serão os próprios utilizadores a definir a chave primária, face ao conhecimento que têm do sistema e no caso desta ser infrutífera, a segunda opção será inspeccionar o código de inserções e verificar quais os atributos mais utilizados, adoptando-os como chave primária.

É expectável que esta fase não seja crítica porque as chaves primárias são normalmente conhecidas. De facto, no SIIP, foram identificadas 185 tabelas, das quais 180 tinham restrições de índice único, havendo 3 com mais de um índice. Nas restantes 5 tabelas, a chave primária foi identificada por inspecção ao código, como referido no parágrafo anterior.

2.2 Agrupamento de tabelas em entidades abstractas

Devido à grande dimensão dos sistemas, centenas de tabelas e milhares de atributos, os modelos conceptuais que os representam são também complexos e de grande dimensão. Uma forma bem conhecida de abordar os problemas complexos é dividir o problema em vários problemas mais simples (dividir para conquistar). Assim, nesta fase da metodologia, agrupam-se as tabelas por forma a que se possa fazer o levantamento do modelo de dados das tabelas de cada grupo isoladamente. Muito embora seja necessário uma fase final para agregar os modelos obtidos num único e completar as eventuais dependências entre estes, o problema inicial ficou decomposto numa série de problemas mais tratáveis.

A descrição detalhada dos critérios de agrupamento das tabelas que se fundamentam na composição da chave primária é representada em [Sousa99]. Na figura 1, apresentamos os grupos obtidos processando as 185 tabelas do SIIP. Cada grupo é apresentado como uma entidade abstracta ou como uma associação de um modelo abstracto. Os critérios de agrupamento das tabelas asseguram que:

- o identificador de uma entidade abstracta é a intersecção das chaves primárias das tabelas que a constituem, que é garantidamente não vazia;

- a intersecção das chaves de quaisquer duas tabelas de quaisquer duas entidades abstractas é vazia;

- os grupos representados como associações do modelo abstracto significam que a chave das suas tabelas contêm elementos das chaves das entidades que relacionam.

Apresenta-se um exemplo na figura 1, onde a entidade abstracta "NII" representa o grupo das tabelas que identificam e descrevem as pessoas. A entidade abstracta "UNIDADE" representa o grupo das tabelas que identificam e descrevem as unidades. A associação "PERTENCEM" representa todas as tabelas que relacionam tabelas das pessoas com tabelas das unidades. Este esquema abstracto, para além de reduzir a complexidade do problema do levantamento do modelo de dados é também um precioso auxilio à compreensão do "Universo do Discurso" do SIIP.

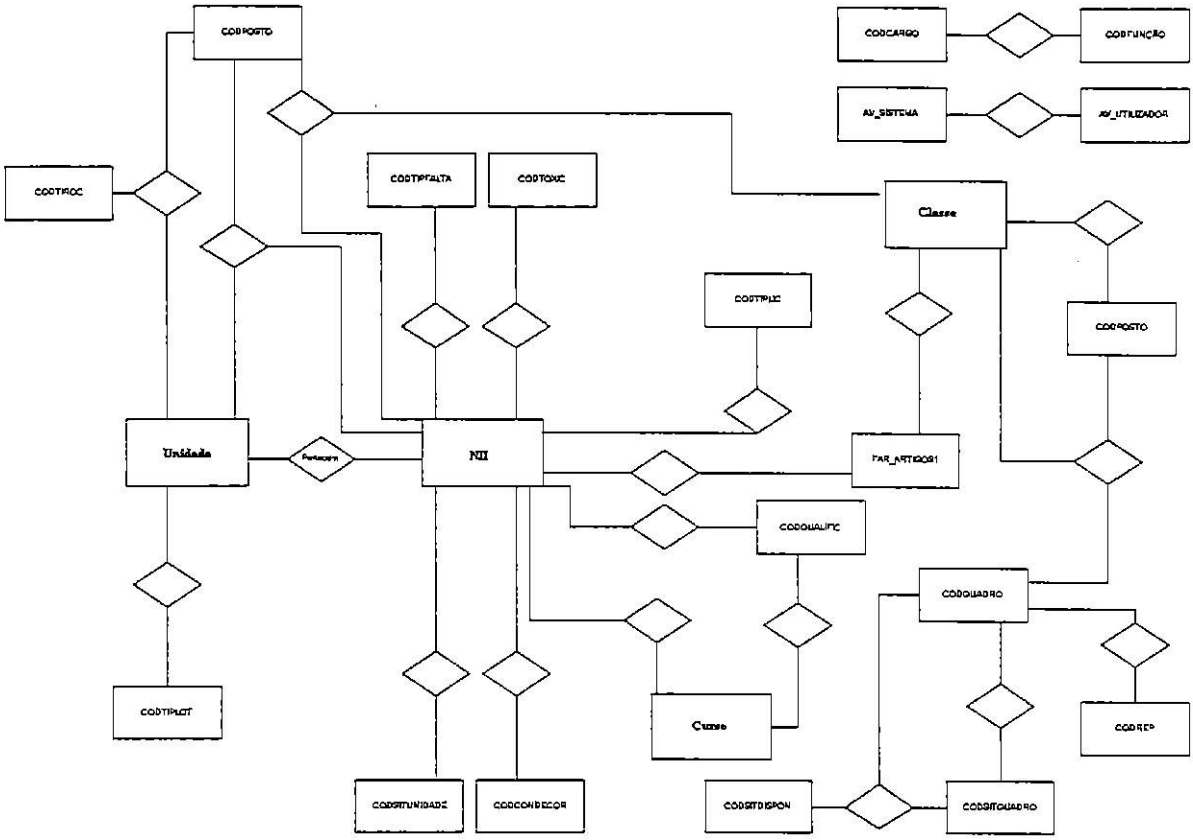
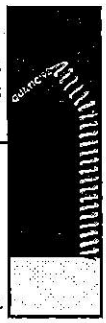


Figura 1- Esquema abstracto do SIIP

2.3. Especificação do esquema abstracto

Nesta fase, pretende-se classificar as diferentes tabelas de cada grupo ou associações abstractas em entidades fortes, fracas e associações de acordo com a semântica do modelo de dados entidade-associação. A classificação tem como base a composição dos atributos que constituem a chave primária da tabela [Batini92, Chiang94, Soutou96].



Deste modo, classificamos como:

- representantes das entidades fortes, as tabelas cuja chave primária não contenha qualquer outra chave primária de outra tabela;

- representantes das entidades fracas, as tabelas cuja chave primária é constituída por uma chave de outra tabela e o(s) restante(s) atributo(s) não contêm a chave de outra tabela;

- associações, caracterizam-se pelas suas chaves primárias serem compostas por chaves de pelo menos duas tabelas.

Foi desenvolvida uma aplicação para produzir automaticamente esta classificação, recorrendo a um conjunto de interrogações à base de dados. Desta forma, foram obtidas 104 tabelas classificadas como fortes, 67 fracas e 14 associações.

A maior dificuldade nesta fase foi entender o significado dos atributos de algumas entidades fracas.

Na figura 2, apresentamos o esquema conceptual obtido pelo levantamento das tabelas que pertencem à entidade abstracta NII.

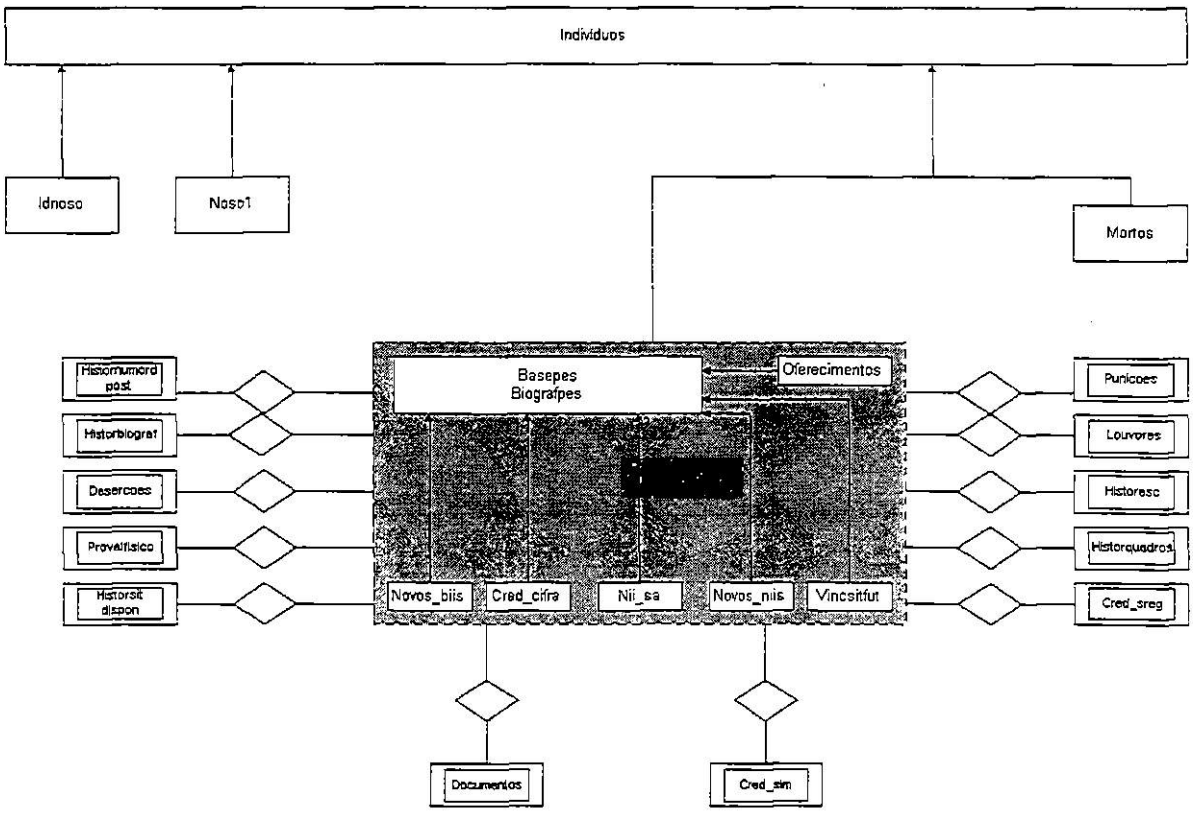


Figura 2 – Representação interna da entidade abstracta NII, nas suas componentes específicas

2.4 Identificação de generalizações e fragmentações

Nesta fase pretende-se identificar as tabelas com as mesmas chaves primárias bem como constatar as dependências de inclusão entre elas.

Quando encontramos duas tabelas com a mesma chave primária interessa conhecer as interdependências das chaves comuns nos registos entre duas tabelas.

Esta informação é obtida através da intersecção dos valores da chave de cada tabela. Desenvolvemos uma aplicação, que gera os ensaios necessários de forma a cobrir todos os cenários possíveis. Assim, podemos ter um dos seguintes quatro resultados:

- sobreposição, no caso de existirem chaves comuns nos registos comuns a ambas as tabelas e simultaneamente cada tabela ter registos exclusivos;

- disjunção, quando a intersecção das chaves comuns nos registos de ambas as tabelas for vazia;

- inclusão, quando todas as chaves comuns nos registos de uma tabela estão incluídas na outra;

- coincidência, no caso das chaves comuns nos registos de ambas as tabelas serem exactamente as mesmas.

No caso do SIIP, foram necessários cerca de 500 ensaios à base de dados para validar as dependências de inclusão das chaves primárias. Como resultados conseguimos identificar duas fragmentações verticais e três generalizações.

A figura 3 ilustra o caso de uma fragmentação das tabelas T_Con_Tipcol e T_Con_Coloca. Trata-se de tabelas de conversão de códigos em que verificamos que além de terem a mesma chave primária, ambas as tabelas tinham o mesmo número de registos e eram exactamente os mesmos (coincidentes). Assim, criando uma entidade abstracta, no caso T_Con, poderemos tratar ambas as tabelas como uma só, a um nível mais elevado em que não se discutem implementações mas conceitos. De seguida verificamos que todos os registos dessas tabelas estavam incluídos numa outra tabela (T_Codunidade).

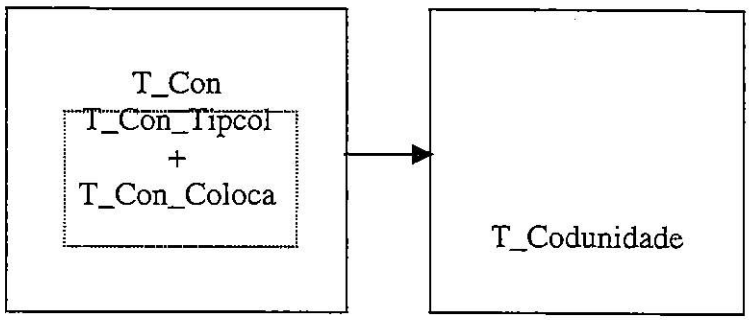


Figura 3 – Fragmentação vertical e subconjunto

A figura 4 ilustra uma generalização. Esta envolve um conjunto de tabelas com a mesma chave primária. Todas as tabelas fazem referências a pessoas. A análise das dependências entre os valores das chaves primárias destas tabelas mostrou que existia um conjunto de tabelas que eram disjuntas da tabela de mortos e simultaneamente estavam incluídos numa entidade representativa dos elementos vivos da Marinha. Note-se que na figura, as entidades VIVOS, INDIVÍDUOS (a sombreado) não estão implementadas fisicamente, são apenas conceitos representados como entidades do modelo conceptual. Entretanto as tabelas Idnoso e Noso1, referentes a registos nosológicos, incluem registos de elementos vivos e mortos. A união destas quatro entidades formam o grupo de todos indivíduos que estiveram ou estão na Marinha.

2.5 Detecção das restantes associações

Nesta fase pretendemos identificar associações através de chaves estrangeiras.

Tal como no ponto 2, a obtenção de chaves estrangeiras segue as seguintes fases:

i) geração de hipóteses;
ii) rejeição das hipóteses falsas com base nos dados;
iii) confirmação das restantes hipóteses.

A geração de hipóteses foi efectuadas segundo duas aproximações distintas, a primeira com base nos nomes e a segunda com base na análise das instruções de SQL (*joins*) do código fonte. Em relação à primeira, fomos procurar sinónimos dos nomes dos atributos que compõem uma chave primária de uma relação e que estão presentes noutra relação. Existe, deste modo, uma suspeita da existência de uma chave estrangeira. Pela segunda via de detecção das chaves estrangeiras é possível, ao analisar o código fonte e, mais especificamente as instruções de SQL, verificar quando há comparações, inserções, actualizações e eliminações de registos entre duas tabelas diferentes. Normalmente isso significa que existe uma relação entre essas tabelas. Daí que, uma vez mais, exista uma forte suspeita de serem possíveis chaves estrangeiras.

Esta análise, para além de permitir gerar um conjunto de possíveis chaves estrangeiras, consegue retirar mais alguma informação no que respeita a homónimos. Como referido anteriormente, estes são muitas vezes difíceis de detectar e, com esta acção poder-se-á inferir algo quanto ao seu verdadeiro significado e verificar se têm a mesma semântica ou, pelo contrário, nada têm em comum à excepção do nome.

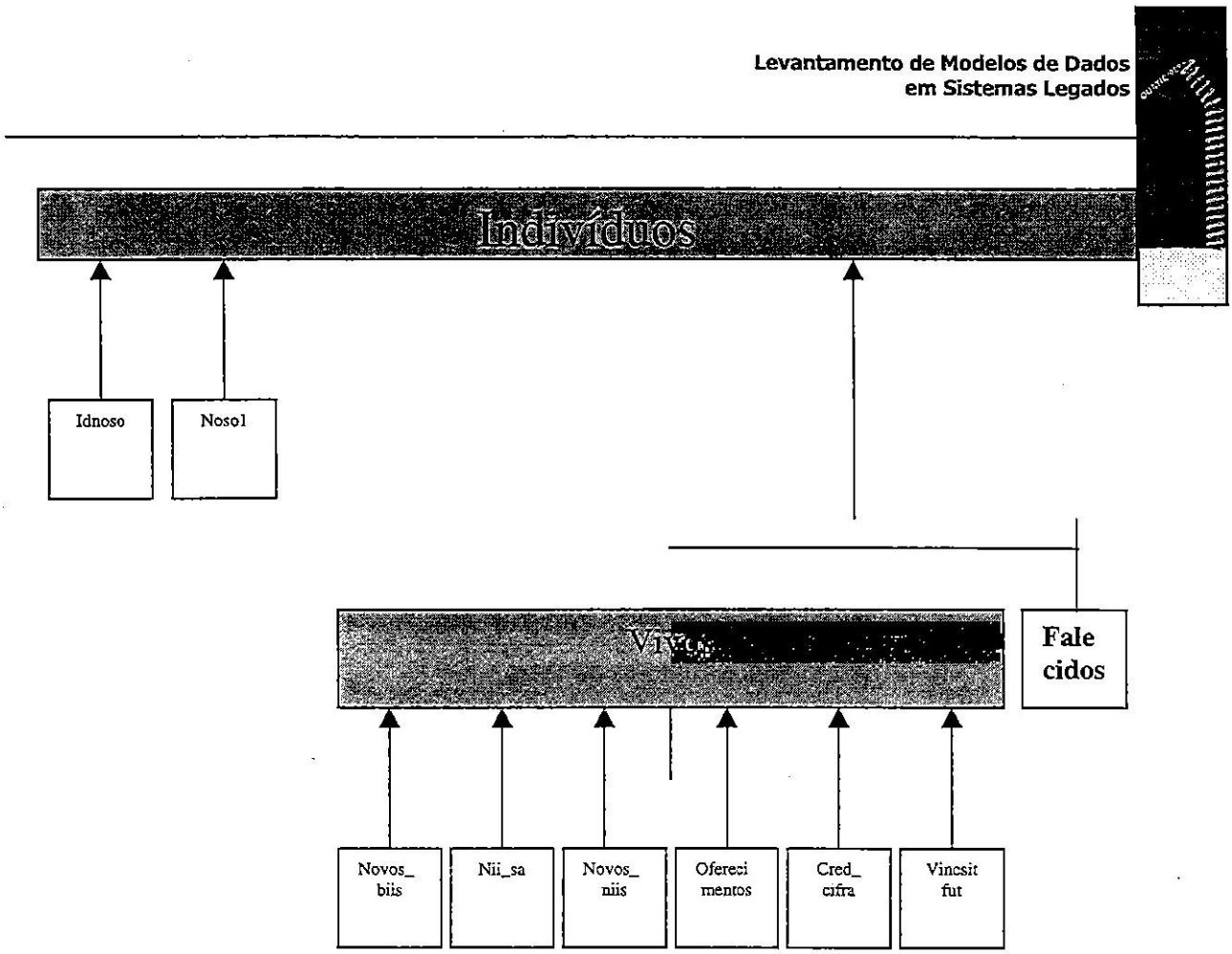
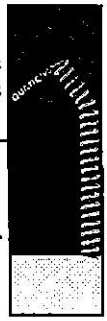


Figura 4 – Generalização de tabelas

Na obtenção das generalizações, encontramos muitos casos de sobreposição de conjuntos. Estes casos de sobreposição só poderão ser modelados como subconjuntos tal como mostra a figura 4.



Naturalmente que estes deverão ser confirmados com os utilizadores do sistema, quer sejam parte da equipa de manutenção ou utilizadores finais.

Foi desenvolvido um conjunto de *queries* que ensaiam se uma tabela referencia outra. Assim, se uma tabela contém todos os registos de um atributo chave de uma outra tabela em que o atributo não pertence à chave, então existem fortes probabilidades de esta última estar a referenciar a primeira.
Estes ensaios vão eliminar algumas das possíveis chaves estrangeiras obtidas anteriormente.

No caso do SIIP, geraram-se cerca de vinte mil ensaios e foram levantadas cerca de quatro mil hipóteses de chaves estrangeiras. Desta hipóteses foram validadas cerca de 150.

2.6 Outros Métodos

Outros métodos de levantamentos de modelos de dados têm sido publicados na literatura científica. Alguns destes são em seguida brevemente descritos.

2.6.1 Utilização de padrões procedimentais na abstracção de esquemas relacionais.

Neste tipo de abordagem proposto em [Signore94], pretende-se desencadear um processo de *reverse engineering* através da identificação do esquema da base de dados, chaves primárias, instruções *SQL*, indicadores procedimentais e heurísticas que conduzem à reconstrução do esquema conceptual.

2.6.2 Utilização de *queries* por forma a melhorar o *reverse engineering* de bases de dados.

Nesta abordagem descrita em [Petit94], pretende-se obter um esquema entidade-associação estendido (EAE) de uma base de dados relacional partindo de dois princípios:

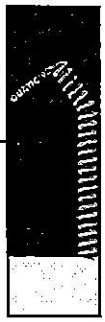
- o método baseia-se em pressupostos verdadeiros (práticos);
- as dependências entre atributos não são conhecidas à priori.

Este método retira informação sobre os nomes das associações, nomes dos atributos, e características destes últimos (valor único, não nulo, etc.), a partir do esquema da base de dados, tal e qual como ele está no dicionário do DBMS (a partir do nível físico e da forma como está implementado). Depois, a extracção semântica é feita por uma análise de *queries*.

2.6.3 *Reverse engineering* de bases de dados relacionais: extracção de um modelo EAE a partir de uma base de dados relacional.

Nesta abordagem referida em [Chiang94], é proposta uma metodologia de *reverse engineering* de bases de dados que consiga obter, a um alto nível de automação, um modelo EAE que corresponda às especificações do desenho de uma qualquer base de dados relacional.

A metodologia inclui a decomposição das tabelas pelo menos para a terceira forma normal, a classificação das tabelas e atributos, a geração de dependências de inclusão e a identificação das estruturas modelares do modelo EAE.



3. Conclusões

Os sistemas legados são uma fonte de preocupação para qualquer organização. Porém, é notória a consciencialização por parte das organizações para a importância do conhecimento do modelo de dados como factor de qualidade dos sistemas de informação, ressaltando o seu contributo para a manutenção, e o desenvolvimento ou melhoria do *software* legado, e/ou a migração dos sistemas para outras plataformas tecnológicas.

Embora o processo de *reverse engineering*, seja aquele em que se obtém um esquema lógico ou conceptual do sistema a partir do nível físico através de ferramentas automatizadas [McClure92], a verdade, é que não é possível automatizar todo o processo. Existem situações em que apenas os utilizadores poderão indicar como o sistema está a funcionar. A título de exemplo considere-se o caso em que existam várias tabelas com a mesma chave primária. Nesta situação surge o problema da definição de associações entre elas. Utilizando ferramentas automatizadas, poderão ser definidas associações erradas. No estado da arte actual, julga-se necessário o apoio de um perito da organização como validação das hipóteses formuladas, no aspecto respeitante à definição das associações.

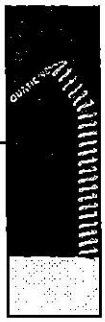
O resultado prático deste projecto foi a obtenção do modelo de dados do SIIP. Este resultado foi alcançado com o suporte de várias ferramentas de apoio à reengenharia, entre as quais se destacam o S-Designer [Designer91] e o DB-Extract. Este último, produzido para o efeito, foi desenvolvido em Visual Basic e produz um ficheiro SQL com informação sobre as tabelas, nomeadamente quanto à sua classificação, agregados e relacionamentos. O S-Designer, foi utilizado como ferramenta gráfica, de modo a permitir a visualização gráfica, do ficheiro SQL gerado pelo DB-Extract.

Ainda é cedo para avaliar o impacto do trabalho efectuado. Este ainda se encontra em fase de conclusão, mas os seguintes resultados serão esperados:

- contributo para a normalização dos dados;
- contributo para a documentação do sistema;
- contributo para a aprendizagem mais rápida dos novos elementos.

Com todos os resultados obtidos nos pontos anteriores pode-se obter a documentação completa e actualizada. Com os actuais meios, qualquer alteração efectuada futuramente pode ser registada na documentação de uma forma bastante prática, permitindo que a documentação seja actualizada à medida que acções de manutenção e desenvolvimento sejam desencadeadas.

A concluir há que referir que, a recuperação do modelo de dados e uma eventual alteração da sua estrutura pode conduzir ao aumento de tempo de vida útil do sistema, tornando-o mais flexível, mais fiável, e contribuindo, por isso, para a melhor qualidade dos sistemas de informação.



4. Trabalho Futuro

A actualização da documentação, referida no ponto anterior, é uma acção incremental e por isso, a alteração dinâmica deverá ser vista como um objectivo futuro.
Todo este trabalho contribuiu para a clarificação da forma como o sistema está a funcionar e pode vir a conduzir a outros desenvolvimentos, nomeadamente à análise dos atributos não chave, à detecção de redundâncias e de atributos derivados e opcionais.
Os resultados obtidos deverão ser um forte suporte ao levantamento de modelo de processos que se espera fazer futuramente.

Referências

[Batini92] Carlo Batini, Stefano Ceri, Shamkant B. Navathe, "Conceptual Database Design - An Entity-Relationship Aproach", Benjamin/Cumings, 1992.

[Chiang94] Roger H.L. Chiang, Terence M. Barron, Veda C. Storey, "Reverse Engineering of relational databases: Extraction of na EER model from a relational database", Data & Knowledge Engineering 12 (1994) 107 - 142, Elsevier Science.

[Designer91] S-DESIGNER, AppModeler for PowerBuilder, Evaluation Version 5.1.0 32-

bit, Sybase Inc., and its subsidiaries, 1991.

[Furlan94] José David Furlan, "Reengenharia da Informação – Do Mito à Realidade", Makron *Books* do Brasil Editora Lda, 1994.

[McClure92] Carma McClure, "The Three Rs of Software Automation : Re-engineering, Repository, Reusability", Prentice Hall, 1992.

[Petit94] J-M. Petit, J. Kouloumdjian, J-F. Boulicaut, F. Toumani, "Using Queries to Improve Database Reverse engineering", in Proc. Of the 13th International Conference on Entity-Relationship Approach, Lecture Notes in Computer Science, Volume 881, pp 369-386, Dec. 1994, Manchester, UK

[Signore94] Oreste Signore, Mario Loffredo, MauroGregori, Marco Cima, "Using Procedural Patterns in Abstracting Relational Schemata", in Proc. Of the 13th International Conference on Entity-Relationship Approach, Lecture Notes in Computer Science, Volume 881, Dec. 1994, Manchester, UK

[Sousa99] Pedro Sousa, Lurdes Pedro-de-Jesus, Fernando Brito e Abreu, "Clustering Relations into Abstract ER Schemas", submetido à CSMR99, Amesterdam, Netherlands.

[Soutou96] Christian Soutou, "Extracting N-ary Relationships Trough Database Reverse Engineering", in Proc. of the 15th International Conference on Conceptual Modeling, 1996, pp. 392-405, Cottbus, Germany.

Palavras chave: *Reverse Engineering*, Qualidade, Sistemas Legados, Bases de Dados