INSTITUTO DE PESQUISAS TECNOLÓGICAS DO ESTADO DE SÃO PAULO

CARLOS SHIDOMI

Modelagem de Processos ETL
Análise e Avaliação

São Paulo

2004

CARLOS SHIDOMI

Modelagem de Processos ETL
Análise e Avaliação

Dissertação apresentada ao Instituto de Pesquisas Tecnológicas do Estado de São Paulo – IPT, para obtenção do título de Mestre em Engenharia de Computação.
Área de concentração: Engenharia de Software

Orientadora: Drª Edit Grassiani Lino de Campos

São Paulo

2004

Shidomi, Carlos
Modelagem de processos ETL análise e avaliação. / Carlos Shidomi. São Paulo, 2004.
104p.

Dissertação (Mestrado em Engenharia de Computação) - Instituto de Pesquisas Tecnológicas do Estado de São Paulo. Área de concentração: Engenharia de Software.

Orientador: Prof. Dra. Edit Grassiani Lino de Campos

1. Processos de ETL 2. Modelagem gráfica 3. Qualidade de dados 4. Tese I. Instituto de Pesquisas Tecnológicas do Estado de São Paulo. Centro de Aperfeiçoamento Tecnológico II. Título

CDU 004.414.23(043)
S555m

Dedicatória

Aos meus pais.

Agradecimentos

Agradeço especialmente a minha esposa Vera Lucia Reiko Yoshida Shidomi pela ajuda, força e companheirismo em todos os momentos. Sem a sua ajuda esta obra não seria finalizada. À minha orientadora Edit pela paciência, compreensão e incentivo. Aos amigos de trabalho, Eloísa e Luis e aos amigos da secretaria do IPT.

Muito Obrigado.

Resumo

As soluções de software conhecidas genericamente por ETL (*Extract, Transform and Load*) são responsáveis pela integração de dados corporativos entre sistemas. Elas consomem em média de 70% a 80% dos recursos utilizados para implementação de um projeto de DW, o que torna o ETL um assunto relevante.

As ferramentas de ETL existentes no mercado são ferramentas de custo muito alto e, devido a isso, poucos programadores as utilizam. Esses programadores precisam ter uma alta produtividade para compensar essa deficiência quantitativa e é neste ponto que a utilização de um modelo gráfico de especificação de processos de ETL pode auxiliar pois, a utilização de um modelo gráfico pode tornar mais fácil e claro o entendimento dos requisitos de um processo de ETL a ser construído.

Este trabalho apresenta um estudo de caso que utiliza um modelo gráfico de especificação de processos de ETL em substituição a um modelo de especificação essencialmente dissertativo e, dessa forma, avalia sobre critérios qualitativos, se houve melhoria com a aplicação do modelo. A partir dessa experiência, os resultados são utilizados para propor sugestões de melhoria de forma a apoiar uma futura evolução do modelo escolhido.

Palavras chaves: Modelagem Gráfica de Processos de ETL, Qualidade de Dados, Modelo Conceitual.

ABSTRACT

Software tools generally commonly refered to as ETL (Extract, Transform and Load) are responsible for the integration of corporate data spreaded among systems. These take an average of 70% to 80% of resources used in the deployment of datawarehouse projects, which makes ETL such a relevant subject.

ETL tools available in the market are very expensive and, as a result, these tools are used by few programmers in the company. Those programmers must have a high productivity to compensate for this fact and this is where a graphical model for the specification of ETL processes can help, turning the understanding of the requirements of an yet to be built ETL process easier and clearer.

This paper presents a case study which uses a graphical conceptual model for the specification of the ETL processes as a replacement to an essentially dissertative specification model and, based on qualitative criteria, evaluates possible improvements of productivity in using such model. The results of this experience are used to suggest modifications on the original conceptual model in order to comply with several requirements of real case situations.

key words: graphic modelling of ETL processes, data quality, conceptual model.

Lista de Ilustrações

Lista de Abreviaturas

Palavra	**Descrição**
CRM	Custom Relation Management
DW	Data Warehouse
ETL	Extract, Transform and Load
BI	Business Intelligence
ETI	Evolutionary Technologies International
XML	eXtensible Markup Language
EBCDIC	Extended Binary Coded Decimal Interchange Code
ASCII	American Standard Code for Information Interchange
OO	Object Oriented
CASE	Computer Aided Software Engenering

Sumário

CAPÍTULO 3

CAPÍTULO 4

CAPÍTULO 5

CAPÍTULO 6

1 Introdução

1.1 Motivação

A tomada de decisões, no atual mundo dos negócios, acessa uma quantidade cada vez maior de dados de fontes distintas. Dessa forma, cada vez mais, é necessário integrar novas fontes de dados para alimentar esse processo decisório. Integrar uma fonte de dados significa, em maior ou menor grau, utilizar processos de ETL (*Extract, Transform and Load*) de forma a disponibilizar dados para as ferramentas de BI (*Business Intelligence*). Integrar uma fonte de dados é uma tarefa que consome uma quantidade de recursos enorme entre levantamento, análise e programação de processos ETL. O consumo de recursos, ainda pode ser agravado por problemas de comunicação entre analistas e programadores. Este trabalho procura, por meio da utilização de um estudo de caso, uma forma de melhorar essa comunicação entre analistas e programadores de forma a minimizar o desperdício de recursos na confecção de processos ETL.

A notoriedade adquirida, nos últimos anos, por ferramentas como o DW (*Data Warehouse*), CRM (*Customer Relationship Management*) e ERP (*Enterprise Resource Planning*) vem das promessas dos benefícios que a implementação dessas ferramentas pode trazer para uma empresa, porém essas soluções utilizam como matéria prima para seu funcionamento dados que pertencem a sistemas específicos espalhados pela estrutura organizacional da empresa. É necessário que esses dados sejam colocados à disposição dessas ferramentas para que essas promessas de benefícios possam ser atendidas.

Colocar um dado à disposição de uma ferramenta de BI significa efetuar a extração do dado do sistema fonte, sua transformação e finalmente sua carga no sistema destino. Essas três atividades são conhecidas como

atividades de ETL e são responsáveis pela integração de dados corporativos entre sistemas.

Vários autores concordam [4, 14, 15] que as atividades de ETL consomem de 70% a 80% dos recursos utilizados para implementação de um projeto de DW, o que torna o ETL um fator primordial na análise da viabilidade de implementação de um projeto deste tipo. Porém, sendo o ETL uma atividade meio para a implementação de uma solução de BI, quase sempre ele é tratado como parte da solução, recebendo um destaque menor do que o merecido.

O ETL pode ser entendido como uma solução de software, e como tal pode ser construído de forma manual (*in-house*) ou por meio de ferramentas de construção. Um recente levantamento [15] apurou que entre 761 profissionais do ramo, 18% constroem seus processos ETL de forma manual, 45% utilizam ferramentas de construção e 37% utilizam ambas as formas de construção em seus processos ETL.

Independente da forma, a construção de um processo de ETL nem sempre é de responsabilidade do indivíduo que faz o levantamento dos requisitos e a especificação dos processos de ETL. Considerando que o indivíduo que especifica o processo de ETL seja diferente daquele que produz esse software, a clareza da documentação passa a ser importante, pois se torna o meio de comunicação entre as partes.

Problemas nessa comunicação podem acarretar dúvidas e interpretações incorretas por parte do indivíduo que configura a ferramenta de construção ou constrói o software de ETL; portanto torna-se necessária à criação de uma linguagem que padronize a comunicação entre as partes e que sirva também para definir e documentar processos de ETL de forma a dirimir ao máximo as dúvidas com relação às especificações definidas para o processo.

Um outro aspecto sobre o ETL é que sendo uma solução de software, ele pode sofrer alterações e evoluções ao longo do tempo. Caso seja necessária a implementação de uma nova regra de transformação ou um novo atributo, é necessário contextualizar esta alteração dentro das funcionalidades do processo já implementado. Isto significa recuperar a "memória do processo", ou seja, recuperar toda a documentação de especificação do processo já implementado e complementar essa especificação com os novos requisitos.

Verifica-se que, para a construção de um processo ETL utilizando um documento de especificação, é aconselhável que este documento de especificação seja construído de forma clara e objetiva, de forma a possibilitar o melhor uso dos recursos disponíveis pela redução dos problemas de interpretação. A utilização de um modelo gráfico de especificação de processo de ETL atende essas necessidades, trazendo uma série de outros benefícios no bojo da sua aplicação.

1.2 Objetivo

Este trabalho utiliza um estudo de caso comparar dois modelos distintos de especificação de processo de ETL. Um modelo de especificação é dissertativo, atualmente utilizado no ambiente de estudo, e o outro modelo é conceitual e gráfico, extraído do artigo *Conceptual Modeling for ETL Process* de autoria de Panos Vassiliadis, Alkis Simitsis e Spiros Skiadopoulos [4], aplicado da forma em que este modelo é apresentado em seu artigo original.

Neste trabalho o modelo gráfico de especificação será chamado de Modelo Conceitual [4] em comparação ao Modelo Dissertativo atualmente em uso no ambiente de estudo.

Os resultados da comparação entre os dois modelos de especificação são avaliados observando os critérios de:

- Representação – Que avaliará qual dos modelos de especificação fornece o melhor ferramental para representação de um processo de ETL.
- Usabilidade – Que avaliará qual dos modelos de especificação possui o ferramental de especificação de processo de ETL de mais fácil utilização.
- Visão Integrada – Que avaliará qual dos modelos de especificação fornece uma visão mais clara do processo de ETL e o seu relacionamento com as entidades e sistemas externos.
- Automatização – Que avaliará qual dos modelos de especificação encontra-se mais próximo de implementação em uma ferramenta CASE.
- Reutilização – Que avaliará qual dos modelos de especificação, quando implementados, permite maior reutilização de código.
- Padronização – Que avaliará qual dos modelos permite uma melhor utilização de padrões para desenvolvimento e especificação de processos de ETL.
- Legibilidade – Que avaliará qual dos modelos representa o processo de ETL de forma mais clara.
- Adaptabilidade – Que avaliará qual dos modelos possui um ferramental mais adaptado a implementação de modificações ao modelo e à notação.

O objetivo da aplicação desse dois modelos de especificação é avaliar diferenças entre eles segundo os critérios estabelecidos, de forma a identificar qual dos modelos é o mais eficiente na representação de processos ETL em suas mais diferentes situações.

1.3 Contribuições Esperadas

Pode-se enumerar uma série de vantagens na adoção de um modelo gráfico de especificação de processo de ETL, porém, a enumeração dessas vantagens não é a contribuição que este trabalho procura dar ao assunto ETL. As principais contribuições deste trabalho para o assunto estão em:

- Avaliar o modelo gráfico utilizado como base deste trabalho em uma situação real de utilização.

  O estudo de caso utilizando o Modelo Conceitual [4], permitirá efetuar a avaliação do modelo em situação real de utilização e dessa forma levantar os problemas e as soluções utilizadas na aplicação do modelo para construção de processos de ETL.

- Verificar as possíveis melhorias no modelo e na metodologia para apoiar todo ciclo de vida do processo de ETL

  O modelo apresentado no artigo que serviu de base para este trabalho leva em consideração a utilização do modelo para a concepção e construção do processo de ETL, porém, assim como o ETL é uma solução de software e como tal sofre modificações durante o seu ciclo de vida, é necessário que o modelo seja capaz de refletir essas modificações abrangendo as demais fases do ciclo de vida do software.

- Avaliar a completude do modelo gráfico

  O modelo gráfico utilizado como base conceitual deste trabalho deve ser capaz de representar todas as necessidades de um processo de ETL. O estudo de caso proposto neste trabalho avaliará o ferramental do modelo, na tarefa de representação das atividades de um processo de ETL,

permitindo identificar se existem pontos de melhoria a serem abordados no modelo.

1.4 Metodologia de Trabalho

A pesquisa realizada para este trabalho baseia-se em diversos documentos sobre o assunto como livros, documentos extraídos de *sites* acadêmicos ou de centros de pesquisa e páginas de empresas do ramo.

Observa-se que o ETL encontra-se mais difundido como uma sub-área da área de *Data Warehouse*. As citações sobre o tema, na maioria dos documentos técnicos existentes, levam em consideração a aplicação do ETL para a implementação de um *Data Warehouse* e sendo este o assunto principal, não é comum que os autores entrem mais profundamente no tema.

Apesar disso, foram levantados alguns documentos exclusivamente sobre processos de ETL e entre esses documentos opta-se por adotar o artigo *Conceptual Modeling for ETL Process* de autoria de Panos Vassiliadis, Alkis Simitsis e Spiros Skiadopoulos [4] como base conceitual deste trabalho, de onde foi extraída a notação básica que é utilizada para definição dos processos de ETL que serão avaliados.

A opção pela adoção deste modelo decorreu da constatação da existência de poucos trabalhos que abordassem de forma clara a especificação de processos de ETL do ponto de vista do atributo do sistema fonte, sua extração, transformação e inserção nas bases de dados do sistema destino. Observar o processo de ETL sob este ponto de vista é importante visto que o objetivo de uma especificação de processo de ETL é mapear as mudanças de estado necessárias para correlacionar um atributo do sistema fonte a um outro atributo no sistema destino.

A partir dessa definição do modelo de especificação a ser avaliado, surgiu a necessidade de efetuar uma avaliação comparativa dos dois modelos de especificação e, a adoção de um estudo de caso utilizando os dois modelos de especificação, é decorrente dessa necessidade.

Um outro trabalho que aborda o processo de ETL sob este ponto de vista é o artigo *A Comprehensive Method for Data Warehouse Design* de Sergio Luján-Mora e Juan Trujillo. Este trabalho aborda o ETL dentro do contexto do desenho de um *Data Warehouse* e dentro deste contexto ele introduz uma notação gráfica para especificação de processo de ETL que possibilita especificar a correlação entre os atributos do sistema fonte e os do sistema destino, assim como as transformações necessárias ao processo porém, o autor não aborda com profundidade este modelo pois o foco principal deste trabalho não é o ETL e a lacuna entre o que foi introduzido e a utilização real da notação proposta neste trabalho torna-se muito grande.

Outros trabalhos estudados foram: - *On the Logical Modeling of ETL Processes* de Panos Vassiliadis, Alkis Simitsis and Spiros Skiadopoulos [5] que procura detalhar o funcionamento do modelo lógico que dá suporte ao artigo que é a base conceitual deste trabalho; *Modeling ETL Activities as Graphs* de Panos Vassiliadis, Alkis Simitsis and Spiros Skiadopoulos [11] que aborda o desenho lógico do cenário de ETL que é o contexto mais amplo em que as atividades de ETL ocorrem e sua redução a um gráfico que os autores chamaram de "Gráfico de Arquitetura"; *A Framework for the Design of ETL Scenarios* de Panos Vassiliadis, Alkis Simitsis, Panos Georgantas e Manolis Terrovitis [22] que apresenta a derivação de um cenário de ETL a partir do "Gráfico de Arquitetura" para uma especificação declarativa do tipo meta-linguagem e que fornece os mecanismos necessários para construção de uma ferramenta gráfica de especificação de processos de ETL a partir de um "Gráfico de Arquitetura", o que é feito pela implementação do programa ARKTOS II.

O estudo de caso objeto deste trabalho envolve uma especificação de processo de ETL que utilize a maior quantidade possível de funcionalidades ETL. Selecionar uma especificação de processo de ETL que por si só, contenha a maior quantidade possível de funcionalidades, torna-se uma tarefa difícil. Por esse motivo, foi selecionada uma especificação de processo de ETL envolvendo um processo de melhoria de dados por meio da utilização de uma fonte de dados de CEP's do Correio e, a essa especificação são adicionadas mais funcionalidades ETL de forma a torná-la mais completa.

Essa especificação no Modelo Dissertativo é então utilizada numa "derivação" para o Modelo Conceitual [4], ou seja, todas as funcionalidades especificadas no Modelo Dissertativo são transcritas para a forma proposta no Modelo Conceitual [4]. Deste procedimento resultarão as duas especificações de processos de ETL que são avaliadas.

O paradigma de desenvolvimento hoje utilizado pelo núcleo é a especificação no Modelo Dissertativo, dessa forma, as comparações com relação ao Modelo Conceitual [4] poderão ser utilizadas no futuro para efetuar uma comparação entre outros modelos que venham a ser sugeridos.

1.5 Organização do Trabalho

O capítulo "Estado da Arte" (Capítulo 2) apresenta as bases teóricas que sustentam o estudo de caso deste trabalho. Este capítulo apresenta uma breve introdução, algumas definições sobre a terminologia utilizada, um breve histórico sobre o ETL e seu desenvolvimento até os dias de hoje, a notação básica retirada do artigo *Conceptual Modeling for ETL Process* de autoria de Panos Vassiliadis, Alkis Simtsis e Spiros Skiadopoulos [4], a metodologia de aplicação do modelo e finalmente uma breve conclusão do capítulo.

O capítulo “Modelo Dissertativo Atual” (Capítulo 3), refere-se ao estudo de caso utilizando uma especificação de processo de ETL no Modelo Dissertativo atualmente utilizado pelo núcleo de desenvolvimento de software da empresa de estudo. Neste capítulo é apresentada uma descrição do documento de especificação no Modelo Dissertativo, a especificação selecionada para o estudo de caso e considerações sobre o estudo de caso propriamente dito.

No capítulo “Modelo Conceitual [4]” (Capítulo 4), é abordado o estudo de caso utilizando uma especificação de processo de ETL baseado no Modelo Conceitual [4] no contexto do núcleo de desenvolvimento de software da empresa de estudo. Neste capítulo são apresentados os documentos resultantes da "derivação" do Modelo Dissertativo para o Modelo Conceitual [4], considerações sobre a “derivação” segundo a metodologia sugerida pelos autores, observações sobre o estudo de caso e avaliações feitas sobre os dois modelos de especificação estudados.

O capítulo “Melhorias ao Modelo Conceitual [4]” (Capítulo 5) apresenta uma avaliação de ambos os modelos de especificação. Nessa avaliação são abordados os pontos fortes e fracos de cada um dos modelos utilizado no estudo de caso. As melhorias no modelo e na metodologia que são sugeridas neste trabalho, são baseadas nesse levantamento de pontos fortes e fracos.

No capítulo “Conclusões” (Capítulo 6), são apresentados um resumo do trabalho, contribuições e sugestões para trabalhos futuros.

2 Estado da Arte

2.1 Introdução

Neste capítulo são apresentadas as bases teóricas utilizadas para o estudo de caso previsto neste trabalho. Inicialmente é feita uma breve conceituação do assunto ETL, incluindo a apresentação do modelo lógico no qual está baseado o Modelo Conceitual [4]. Feita esta conceituação, é apresentado um breve histórico sobre o ETL e sua evolução até os dias de hoje. A seguir o modelo, que é base conceitual deste trabalho, é apresentado. O modelo é composto por uma notação que apresenta o "léxico" utilizado para especificação de processos de ETL e uma metodologia de aplicação do mesmo.

2.2 Conceito

ETL é um acrônimo de *Extract, Transform and Load* (Extração Transformação e Carga). Esses três tipos de atividades são utilizados para movimentar dados de um ou mais sistemas fontes para um ou mais sistemas destino; é considerado como parte integrante de qualquer solução de BI (*Business Inteligence*), a exemplo do DW (*Data Warehouse*) e atua como ponte entre os sistemas transacionais e o *Data Warehouse* [1, 10]. Outra utilização do ETL encontra-se na área de Bancos de Dados Distribuídos, onde a integração de dados é necessária.

A figura 2.1 apresenta de forma resumida o funcionamento de um processo de ETL, onde se observa a atividade de Extração que visa obter do Sistema Fonte os dados que são utilizados para a carga no Sistema Destino. Esses dados são então enviados para uma Área Temporária, utilizada pela função

de Transformação como uma área de trabalho, na qual os dados extraídos sofrerão as transformações previstas no processo de ETL. O papel das atividades de transformação de um processo de ETL é assegurar a compatibilização das diferenças semânticas entre o Sistema Fonte e o Sistema Destino e, quando necessário, consolidar dados em termos de tipos, formatos e conteúdo dos dados. Sem essa atividade seria impossível o acesso à informação de qualidade. Finalizadas as transformações, a função de Carga efetuará a conexão com o Sistema Destino, para o qual são transferidos os dados já processados da Área Temporária. Os dados a serem movimentados no processo de ETL, podem ser do tipo estruturado ou semi-estruturado, de origem interna ou externa à empresa. Entenda-se dado estruturado como sendo aqueles que foram moldados segundo um formato regularmente encontrado [9]. Já os dados semi-estruturados não apresentam uma estrutura rígida, isto é, não possuem um esquema fixo e pré-definido e não são facilmente tipificados. Um exemplo de dados semi-estruturados encontra-se no padrão XML (*eXtensible Markup Language*) atualmente em uso na Internet.

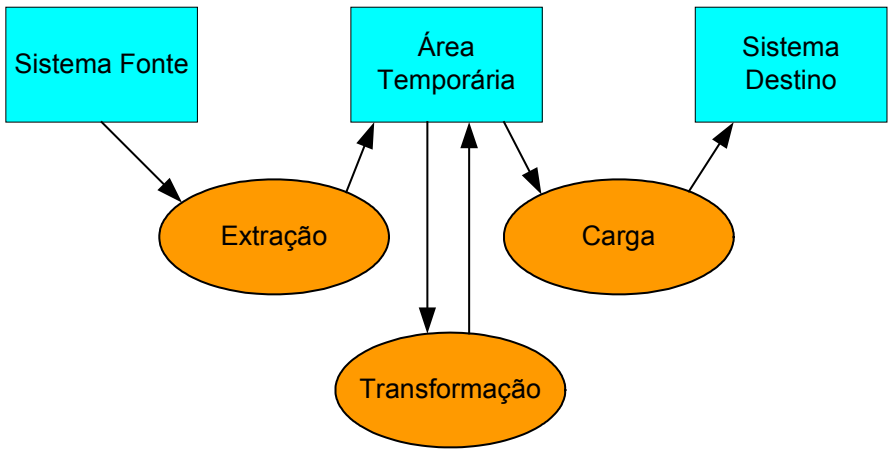


Figura 2. 1.Funcionamento de Processo de ETL [1]

O objetivo de um processo de ETL é disponibilizar o dado para utilização pelo Sistema Destino, ou seja, num processo de ETL, todas as operações são executadas antes do dado ser inserido no Sistema Destino. A compatibilização de domínios necessária para possibilitar a inserção do

dado do sistema fonte no sistema destino exige grandes volumes de recursos para ser efetuada.

Existe uma outra linha de abordagem de ETL [7] que, em prol da redução do tempo e custo no desenvolvimento de processos de ETL (70% a 80% do custo de implementação), propõe que os dados provenientes do sistema origem, sejam carregados no sistema destino diretamente sem transformações.

Nessa abordagem, as funções de limpeza e transformação seriam feitas em tempo de execução por meio de visões do banco de dados. Estas visões podem ser definidas como tabelas virtuais contendo campos de uma ou mais tabelas do banco de dados e podem incluir campos de outras visões. A utilização das visões para efetuar os processos de limpeza e transformação consiste em transferir para o sistema de banco de dados uma grande carga de trabalho o que pode não ser vantajoso.

A abordagem dos processos ETL apresentados neste trabalho levam em consideração que todas as operações de compatibilização são efetuadas antes da carga do sistema destino e, para tanto, são considerados os seguintes passos:

Identificação das estruturas destino

Identificação das estruturas destino dos dados. A partir do modelo de dados do sistema destino, pode-se identificar as entidades que necessitam de carga de dados dos sistemas origem.

Identificação das estruturas origem

Identificação das estruturas origem de dados que serão utilizadas no processo. A partir do conhecimento da necessidade das estruturas destino , é feita uma busca nas possíveis fontes de

dados. Identificada a fonte, é traçada uma estratégia para obtenção do dado a ser inserido na estrutura destino.

Extração do dado origem

Conexão com as estruturas origem de dados e obtenção dos dados nelas armazenados. Existem dois momentos de extração de dados a serem considerados [13]:

1) A atualização imediata do sistema destino

Ocorre quando uma alteração no Sistema Fonte é detectada e imediatamente transferida para o sistema destino mantendo assim os dados do Sistema Destino atualizado.

2) A atualização por demanda do sistema destino

Ocorre periodicamente quando, sob demanda, dados devem ser enviados ao Sistema Destino, devidamente transformados.

Reconstrução histórica do dado

Armazenamento histórico das mudanças de domínio nos dados do sistema fonte. Os sistemas origem estão em constante mudança o que torna necessário à verificação dos dados origem para identificar possíveis mudanças de conteúdo que devem ou não ser refletidas no sistema destino.

Tradução do dado

Tradução para códigos oficiais utilizados no sistema destino. O modelo de dados do sistema destino pode estabelecer um domínio diferente em relação aos atributos correlatos dos

sistemas origem, o que torna necessário uma conversão de códigos entre os dois sistemas.

Re-formatação do dado

Re-formatação para estruturas do banco de dados destino. É a compatibilização com relação aos formatos. Ex.: Datas no sistema origem podem ser representadas como sendo DD/MM/AAAA e no sistema destino pode ser representado como sendo AAAA/MM/DD.

Reestruturação do dado

Reestruturação dos dados para as estruturas destino dos dados segundo seus modelos lógico e físico. É a montagem dos dados para inserção no sistema destino segundo as estruturas especificadas nos projetos lógico e físico.

Sumarização do dado

Sumarização ao nível desejado no sistema destino. O sistema destino pode necessitar de dados agregados onde um registro no sistema destino pode representar muitos registros do sistema origem, esta sumarização indica o nível de granulidade desejada no sistema destino.

Carga do dado

Conexão com o sistema destino e transferência do dado trabalhado para as estruturas destino identificadas

Revisão do dado

Verificações pós-carga para avaliar a qualidade final do dado carregado e identificar as modificações necessárias ao processo.

Para possibilitar a execução dos passos descritos deve-se levar em consideração que um processo de ETL, sendo uma solução de software com a finalidade de atender à complexidade da integração [20], deve executar uma seqüência lógica de atividades de ETL [11], que entre alguns aspectos devem estar preparadas para abranger:

- Mudança de tecnologia, pois normalmente a tecnologia de banco de dados do sistema fonte é diferente da do sistema destino;
- A seleção do dado pode tornar-se complexa visto a necessidade de filtros lógicos e checagens de domínios válidos;
- A re-estruturação das chaves dos registros de entrada normalmente é necessária; na maioria das vezes adiciona-se um elemento de tempo (ex. *timestamp*) à chave para formar um registro histórico.
- A re-formatação dos dados é a compatibilização de formatos entre o sistema fonte e o destino (ex. transformação de datas de formato DD/MM/AAAA no formato AAAA/MM/DD).
- A limpeza do dado implica na aplicação de algoritmos para verificar e corrigir o conteúdo do dado, o que pode exigir desde um simples algoritmo à aplicação de complexas rotinas de inteligência artificial.
- A existência de várias fontes de dados passíveis de utilização, que em momentos diferentes apresentam a condição ideal para extração de dados, implica na utilização de um conjunto de condições que identifica a fonte correta em momento de execução.
- A utilização de várias fontes de dados pode implicar na existência de diferentes chaves de registro, estas devem ser padronizadas para garantir uniformidade ao processo quando da junção de arquivos (ex. geração de uma chave artificial).
- A possível necessidade de geração de diversos níveis de sumarização para os dados resultantes o que faz com que seja necessária a geração de diversas saídas de um único processo de ETL.
- A necessidade de especificação de valores padrões fixos. Em casos em que não existe atributo fonte correspondente ao atributo

destino é necessário estabelecer valores fixos para os atributos destino.

- A eficiência da seleção deve ser perseguida. Em alguns casos, quando ocorre modificação no sistema fonte, acaba por ser necessária a extração de todos os registros em função da impossibilidade de identificação somente dos registros alterados, o que aumenta a carga do processamento.
- A documentação da mudança dos nomes de atributos e das conversões entre os sistemas fonte e destino deve ser construída e mantida.
- Deve possibilitar a conversão entre EBCDIC e ASCII e vice versa.
- Deve possibilitar o processamento em massa de informações.

As funcionalidades descritas estão presentes no modelo gráfico de especificação que é utilizado, modelo este que foi baseado em um modelo conceitual que apresenta os mecanismos para a modelagem gráfica de processos de ETL. Para iniciar a apresentação desse modelo conceitual, é apresentada a figura 2.2 que representa o processo de ETL dentro de um contexto mais amplo que é conhecido como um Cenário de ETL. Neste cenário é apresentado um processo simples de ETL no qual existem duas entidades, uma Fonte de Dados e outra Destino dos Dados. O objetivo é fazer com que os dados da Fonte sejam carregados no Destino.

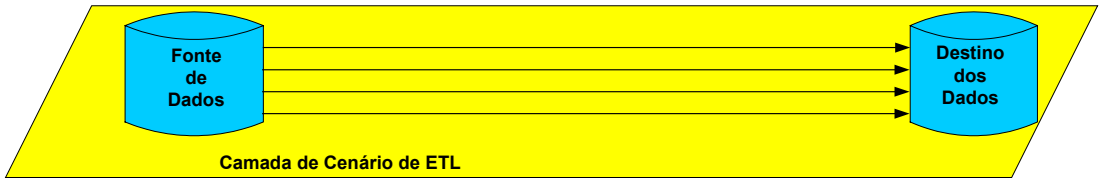


Figura 2. 2 - Cenário Básico de ETL [4]

Considerando que nem sempre pode-se efetuar a carga de um atributo de uma entidade Fonte de Dados para uma entidade Destino de Dados sem efetuar algum tipo de modificação apresenta-se na figura 2.3. o mesmo cenário de ETL no qual estão inseridos alguns itens gráficos que representam transformações, filtros e relacionamentos.

Observa-se que temos neste momento um cenário com Fontes e Destinos de dados relacionados por meio do mapeamento de seus atributos. Observa-se também a necessidade de execução de diversas operações para efetuar a compatibilização dos conteúdos dos sistemas fonte e destino para que os dados do sistema fonte possam ser efetivamente carregados no sistema destino.

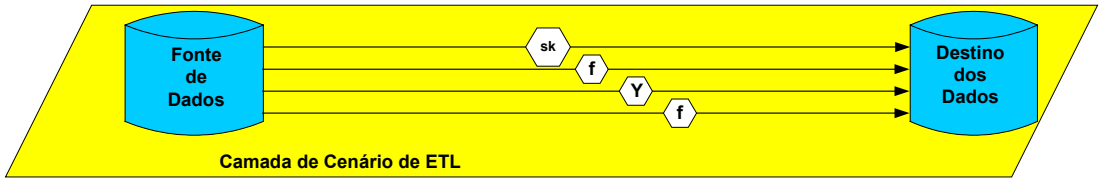


Figura 2. 3 - Cenário e Operações de ETL [4]

A introdução de novos itens gráficos no contexto do cenário levanta a questão de qual é o conjunto de itens gráficos necessários para especificar um processo de ETL. O modelo conceitual, no qual o modelo gráfico de especificação de processos de ETL se baseia, apresenta o conjunto de itens gráficos disponíveis para compor um cenário de ETL. Esse conjunto de itens gráficos encontra-se na Camada de Modelos. A figurar 2.4. apresenta a Camada de Modelos que passa a funcionar como uma biblioteca contendo todos os itens possíveis de utilização para compor um cenário de ETL.

O modelo conceitual, portanto, identifica quatro tipos de itens gráficos a serem utilizados para a especificação de um processo de ETL, são estes: 1- Entidade; 2- Restrições de ETL; 3- Transformações e 4- Relacionamentos. O modelo gráfico de especificação, como é visto mais adiante, utiliza esses quatro tipos de itens gráficos para composição de um processo de ETL [4].

Visto que o ETL deve estar preparado para abranger toda a complexidade da integração utilizando para isso somente quatro conjuntos de itens gráficos para sua representação, torna-se necessário efetuar extensões dessas representações para que todas as funcionalidades ETL sejam consideradas.

Essas extensões enriquecem a Camada de Modelos da figura 2.4 onde são incorporadas as novas funcionalidades ETL.

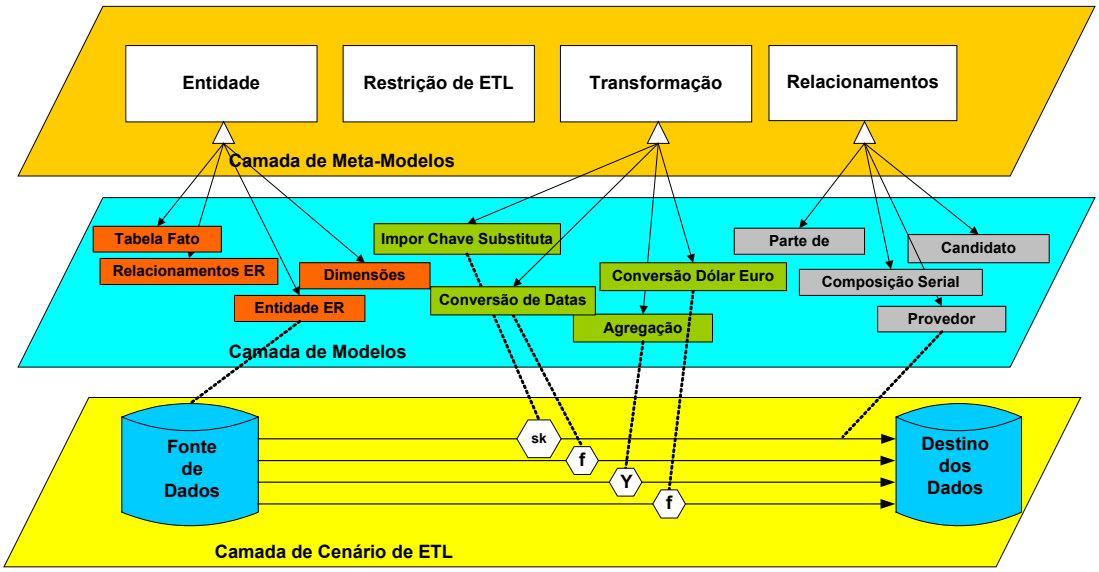


Figura 2. 4 - Modelo Conceitual base para construção do Modelo gráfico de Especificação de ETL [4]

Uma das funcionalidades mais interessantes e, que cada vez mais tem se tornado presente nas novas ferramentas ETL disponíveis no mercado, são as funcionalidades relacionadas à limpeza de dados.

A limpeza de dados é uma funcionalidade de ETL complexa e a sua utilização em no Modelo Conceitual [4] necessita de fundamentação sobre os processos de qualidade de dados envolvidos.

De uma forma geral, existem alguns estágios ou passos para a obtenção da qualidade em dados. A figura 2.5 apresenta os estágios necessários para a obtenção da qualidade em dados.

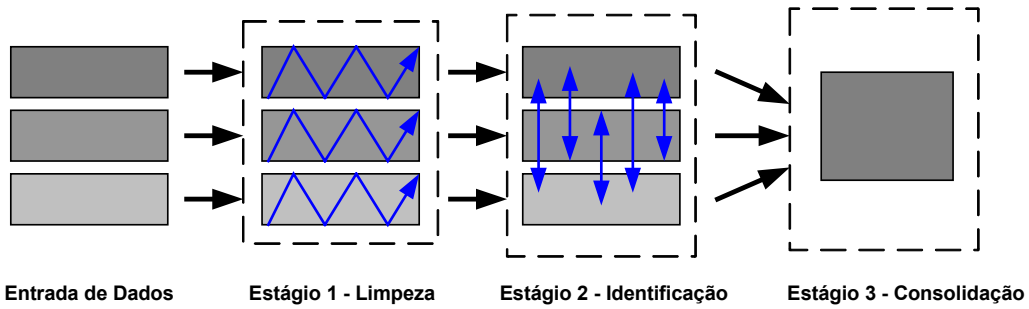


Figura 2. 5- Estágios da Qualidade de Dados [28]

As atividades envolvidas no Estágio um são:

- Divisão em partes menores

  Nessa atividade os dados são divididos em partes menores para facilitar a correção e identificação dos mesmos. Ex.: Um nome pode ser dividido em primeiro nome, nome do meio e último nome

- Correção dos Dados

  A atividade de correção de dados é utilizada para corrigir um conteúdo incorreto. Ex: Um campo Cidade cujo conteúdo seja “SP”, dependendo da unidade federativa ao qual a cidade pertença, deve ter seu conteúdo corrigido para “São Paulo”.

- Padronização dos Dados

  A padronização dos dados permite uniformizar alguns conteúdos de dados. Ex: Pode ser estabelecido que a abreviatura do título de Doutor seja Dr e de Doutora Dra . Dessa forma se o conteúdo de um campo <Título> passa pela atividade de Padronização com o conteúdo Doutora, esse conteúdo será alterado para Dra. automaticamente.

- Enriquecimento dos Dados

  Uma outra atividade do Estágio um é o enriquecimento da informação com base em outras fontes de dados. Ex:: Pode-se utilizar um arquivo dos correios para fornecer o endereço completo a partir da utilização do CEP, dessa forma os dados resultantes são “enriquecidos” com informações novas.

O Estágio dois é a atividade de Identificação que consiste em aplicar processos para identificar informações duplicadas, mesmo que alguns dos

dados não sejam exatamente iguais. Essa atividade possibilita identificar as informações de um indivíduo em fontes de dados distintas, mesmo que uma chave comum não tenha o mesmo conteúdo, utilizando para isso o conteúdo de outros campos como critérios de desempate. Por exemplo, no caso de uma comparação de logradouros entre as fontes de dados (A) e (B), ela identifica similaridades de forma a localizar, com certo grau de certeza, que um determinado logradouro da fonte de dados (A) seja igual a um determinado logradouro da fonte de dados (B) utilizando para isso alguns algorítimos de identificação, entre esses algorítimos temos:

1. Identificação utilizando uma chave (*Key-Code Matching*)

   Efetua comparações utilizando os primeiros caracteres de um ou mais campos. É um método primitivo e pouco utilizado por incorrer em erros na identificação.

2. Identificação utilizando fonética (*Soundexing*)

   Efetua comparações por fonemas. Este método incorre em erros pois os algorítimos, em sua maioria, efetuam comparações com fonemas em inglês. Além disso, a comparação por fonemas pode incorrer em erro pois muitas palavras tem grafia diferente e sons semelhantes.

3. Identificação utilizando similaridade (*Similarity Matching* ou *Fuzzy Matching*)

   Identifica duplicidades computando um grau de confiança entre dois componentes. É considerado o melhor método de identificação, principalmente, quando os dados não podem ser padronizados. Ex. Sobrenome.

4. Identificação utilizando pesos (*Weighted Matching*)

   Utilizado em conjunto com a identificação fonética ou por similiaridade, permite indicar a importância relativa dos campos atribuindo pesos aos mais importantes.

O Estágio três, a partir da identificação efetuada no Estágio 2, consolida as informações gerando uma base de dados capaz de fornecer informações de melhor qualidade. A consolidação não ocorre para sumarizar valores ou efetuar contagens e análises estatísticas, mas para identificar e selecionar a melhor informação para cada um dos campos do registro de saída. Dessa forma, num processo de consolidação para qualidade de dados, a geração de dois registros de clientes diferentes a partir de três fontes de dados distintas pode fazer com que o nome do primeiro cliente tenha como origem à fonte de dados um e o do segundo cliente a fonte de dados três, isso a partir de critérios de seleção.

Observa-se que nem sempre um processo de ETL utiliza todas as funcionalidades de qualidade de dados e, portanto, é necessário conhecer todas as atividades individualmente para possibilitar sua utilização quando necessário. Dessa forma, os processos de qualidade de dados podem ser considerados, de uma forma ou de outra, como parte do processo de ETL visto que podem ser classificados como processos de transformação necessários à transferência dos dados de uma entidade fonte para um destino de dados.

2.3 Histórico

Segundo W.H. Inmon [2], por volta dos anos 60 e 70 iniciou-se a discussão do conceito de integração de dados corporativos. Nessa época poucas aplicações tinham esta integração como um dos objetivos.

Nos anos 80, as corporações passaram a tentar fazer com que seus sistemas aplicativos, que já haviam sido construídos e implementados, parecessem integrados, inserindo funcionalidades que deveriam ter sido planejadas e implementadas quando da concepção de cada sistema aplicativo. Tentavam também, fazer o processamento analítico da informação diretamente em seus aplicativos, mesmo que suas aplicações tivessem sido construídas para processamento transacional, o que causava uma frustração com relação ao desempenho dos aplicativos e resultados obtidos, pois muitas vezes não havia como cruzar os dados da empresa num processo de pesquisa adequado, que retornasse os resultados desejados, pois os dados encontravam-se em sistemas diferentes e em plataformas distintas. Nessa década surgiu também a re-engenharia, que sugeria que todos os problemas de integração seriam resolvidos se a modelagem tivesse sido construída corretamente e portanto, seria necessário re-escrever todos os sistemas, o que seria inviável. As aplicações já existentes simplesmente não iriam mudar.

Nos anos 90 surgiram várias correntes de integração de dados entre elas o DW e a tecnologia ERP. Cada uma delas, à sua própria maneira, tentou solucionar o problema dos dados não integrados. O DW tem a intenção de solucionar o problema da integração mas não soluciona o problema do processamento transacional. O apelo da tecnologia do DW não está na solução do problema do processamento das transações, mas sim no processamento analítico da informação. Já a tecnologia do ERP tem sua visão focada na integração das transações de uma empresa. Muitas empresas têm descoberto que depois da implementação de um ERP, ainda necessitam da implementação de um DW para o processamento analítico da informação. A integração dos dados exigida pela implementação dessas ferramentas leva em consideração a utilização de processos ETL, em sua maioria, construídos à partir da utilização de algum tipo de ferramenta de ETL.

Atualmente observa-se que o mercado para ferramentas ETL tem se expandido, o que pode ser constatado pelo surgimento de diversas ferramentas ETL nos últimos anos, trazendo com elas muitos melhoramentos tecnológicos. Observa-se também que diversos fornecedores de pacotes de banco de dados implementaram funcionalidades ETL em seus produtos para torná-los mais atraentes. Todavia o que podemos chamar de tendência atual das ferramentas ETL está nas chamadas Plataformas de Integração [21]. Esta emergente geração de ferramentas ETL provê grande desempenho; alta capacidade e escalabilidade para grandes volumes de dados a altas velocidades; permite uma redução no tempo de processamento em lote; efetua o processo de carga do sistema destino de forma mais rápida e confiável; utiliza técnicas de captura de dados alterados para melhoria da extração; permanece em operação continuamente tendo uma disponibilidade de vinte e quatro horas, sete dias por semana; melhora a execução do processo e implementa novas funcionalidades como melhoria da qualidade do dado e administração do processo.

As plataformas de integração funcionam como roteadores, conectando vários bancos de dados, sistemas, aplicações e outros roteadores de integração. Eles capturam os dados de processamentos em lote ou em tempo real por meio de arquiteturas de distribuição como *hub-and-spoke* ou *peer-to-peer* e direcionam esses dados de forma a efetuar a sua carga nos sistemas destino; nesse caminho, já efetuam a manutenção automática de metadados de forma a possibilitar a identificação imediata de quaisquer mudanças nos domínios dos campos.

A evolução das ferramentas trouxe um aumento no custo da aquisição, implementação e treinamento das pessoas para utilização de ferramentas ETL e, em conseqüência, acaba agravando a situação na qual existem poucos programadores muito especializados e muitos analistas gerando definições de processos de ETL que em sua maioria não seguem um padrão

de construção, dificultando o entendimento da especificação por parte dos programadores.

É necessário, portanto, um meio de comunicação eficiente para maximizar a utilização dos recursos de programação e que, ao mesmo tempo, permita a evolução da ferramenta de construção ou mesmo a troca desta por outra, sem a perda das definições anteriormente já efetuadas. Isto pode ser obtido pela utilização de um modelo gráfico de especificação de processo de ETL como o a seguir descrito.

2.4 Modelo Conceitual para ETL

Segundo Jeff Johnson e Austin Henderson [3], um modelo conceitual é uma descrição de alto nível de como um sistema é organizado e operado; especifica e descreve os seguintes itens:

- Representações gráficas de metáforas e analogias para serem empregadas no projeto;
- Os conceitos do sistema que o modelo deve representar, seu funcionamento, atributos e operações;
- Os relacionamentos entre todos esses conceitos;
- O mapeamento entre esses conceitos e o funcionamento do sistema

Seguindo as definições acima, a figura 2.6 apresenta as representações gráficas adotadas pelo Modelo Conceitual [4], modelo este adotado como base conceitual para o estudo de caso deste trabalho. Mais à adiante é esclarecido seu funcionamento.

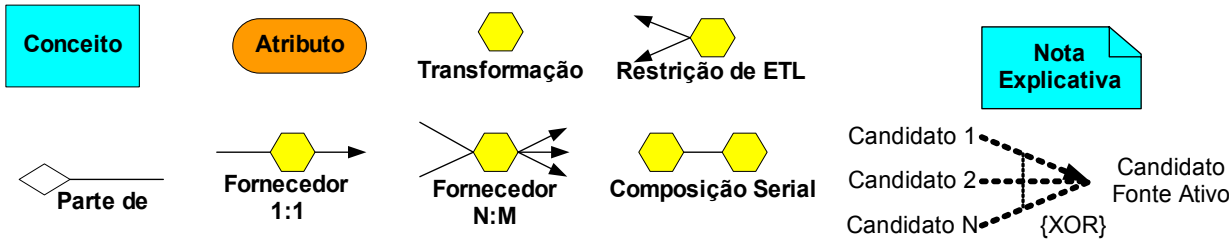


Figura 2. 6Itens gráficos utilizados para representação de processos de ETL [4]

As representações gráficas do modelo não foram baseadas na tecnologia UML, devido à preocupação com a importância do atributo para as operações de ETL. Na tecnologia UML o atributo é colocado em segundo plano, sendo representando dentro de uma Classe. No modelo conceitual sugerido, o atributo é considerado "cidadão de primeira classe" sendo apresentado em destaque.

São apresentadas uma notação e uma metodologia. Na notação são apresentadas as representações gráficas da figura que formam o léxico gráfico a ser utilizado para definição de um processo de ETL. Por léxico entenda-se o conjunto de palavras de que dispõe um idioma [5]. Cada uma das representações é conceituada para explicitar o seu funcionamento. Na metodologia são apresentados os passos que devem ser seguidos para construção de uma especificação de processo de ETL utilizando a metodologia proposta pelos autores.

2.4.1 Notações do Modelo

2.4.1.1 Entidade [4]

A figura 2.7. apresenta os itens gráficos utilizados para definir completamente uma Entidade. A Entidade é a representação de uma fonte

ou destino de dados e é formalmente definida por um conceito e um conjunto finito de atributos.

Figura 2. 7. Itens gráficos utilizados para representação de uma Entidade [4]

A figura 2.8 apresenta a Entidade CLIENTE definida segundo o modelo onde observa-se a Entidade como sendo composta de vários atributos unidos ao conceito por uma relação de dependência na qual os atributos são parte do conceito.

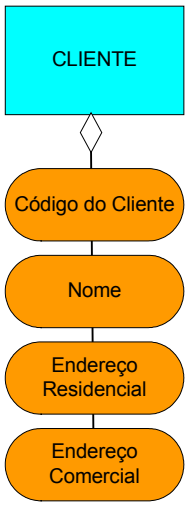
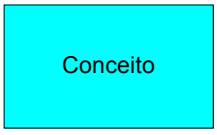


Figura 2. 8. Entidade CLIENTE representada segundo o Modelo [4]

Individualmente, cada um dos itens gráficos utilizados para definir uma entidade possui características próprias que são descritas porém, o relacionamento “Parte de”, pertence a um outro grupo de itens gráficos chamado de Relacionamentos e, por isso é descrito no tópico apropriado.

Conceito [4]

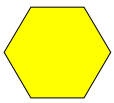


Um conceito pode ser definido como uma representação de um objeto por meio de suas características gerais como qualidade, abstração, idéia e significação. No contexto de uma entidade, representa a idéia que unifica o conjunto de atributos.

Atributo [4]

Pode ser definido como sendo um módulo de informação atômica.

2.4.1.2 Transformação [4]



É uma abstração que representa parte ou módulo de código executando uma tarefa de transformação única. É definida por :

- Um conjunto finito de atributos de entrada que recebem os argumentos necessários para efetuar a transformação (parâmetros de entrada do módulo).
- Um conjunto finito de atributos de saída que são os resultados das transformações solicitadas (parâmetros de saída do módulo).
- Um símbolo que representa graficamente a natureza de uma transformação independente de qual seja.

As transformações podem ser divididas em duas categorias básicas:

- Operação de Filtragem e Limpeza, que efetua a verificação do dado quanto à consistência de seu conteúdo, ou sua rejeição em caso de inconsistência;

- Operação de Transformação, que é definida como sendo qualquer operação em que se mapeia uma configuração em outra [6], ou seja, aquela pela qual ocorre uma mudança em um objeto, devido à alteração de sua configuração pela aplicação de um algoritmo de transformação.

A figura 2.9 contextualiza a utilização de uma transformação em um processo no qual mapeia-se uma entidade fonte em uma outra chamada entidade destino, com a ressalva de que a transformação desejada seja a re-formatação do atributo Data de Nascimento proveniente da entidade origem para o formato YYYY/MM/DD. Assim sendo a transformação em questão possui como parâmetro de entrada o atributo Data de Nascimento da entidade origem e como parâmetro de saída a Data de Nascimento da entidade destino, e efetua como transformação uma inversão de formato de data.

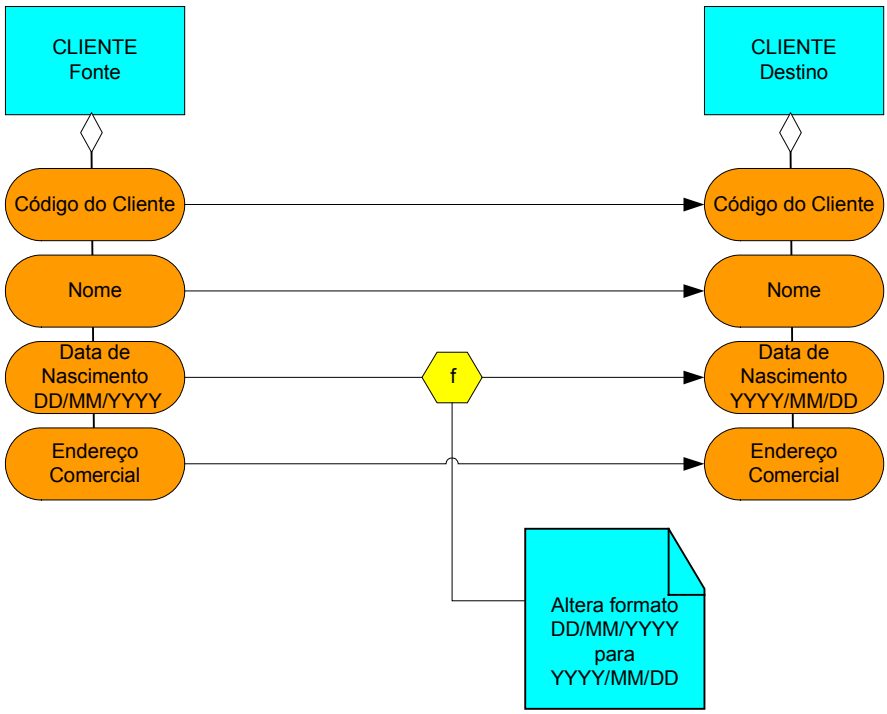


Figura 2. 9. Contextualização de uma transformação em um processo de ETL [4]

As transformações previstas no modelo são divididas em cinco grupos básicos a saber:

- **Filtros** são restrições que podem ser aplicadas para selecionar ou garantir a qualidade da informação que está sendo manipulada;

    - Imposição (I) – Força, por meio de uma transformação, um valor em um conjunto de atributos.
    - Seleção (S) – Efetua a seleção de registros baseado em argumentos de pesquisa
    - Não Nulo (NN) – É uma restrição imposta para garantir valor não nulo no preenchimento do atributo.
    - Violação de Chave Primária (PK) – Efetua verificação se o registro selecionado causa violação de chave primária na entidade destino.
    - Violação de Chave Estrangeira (FK) – Efetua verificação se o registro selecionado causa violação de chave estrangeira em alguma das entidades relacionadas a entidade destino dos dados.
    - Violação de Valor Único (UN) – Efetua verificação se o registro selecionado viola uma condição de valor único imposta a um campo na entidade destino dos dados.
    - Domínio Não Previsto (DM) – Efetua a verificação se o registro selecionado não viola o conjunto previsto de domínios relacionados aos campos da entidade destino dos dados.

- **Operações Unárias** são operações utilizadas para efetuar a transferência de dados de uma entidade fonte para uma entidade destino de dados.

    - Propagação (P) – Efetua uma propagação automática de valores da entidade origem para a entidade destino baseada nos nomes dos atributos.

- Agregação (A) – Executa operações de sumarização para geração de valores totalizados a serem carregados na entidade destino dos dados.
- Projeção (Pr)
- Aplicação de Função (f) – Efetua uma chamada de um módulo ou porção de software com função própria para efetuar um tratamento ou alteração para compatibilizar as informações vidas de uma entidade fonte de dados para outra entidade destino de dados.
- Geração de Chave Substituta (SK) – Gera valores para utilização como chaves de acesso em substituição às chaves originais dos sistemas fonte de dados.
- Normalização de Tupla (N) – Efetua a junção de informações de uma entidade fonte de dados de forma a gerar um único registro na entidade destino de dados.
- Desnormalização de Tupla (DN) – Efetua a separação de informações de uma entidade fonte de dados de forma a gerar vários registros na entidade destino de dados.

- **Operações Binárias** são operações utilizadas para efetuar cargas numa entidade destino de dados a partir de um relacionamento entre uma ou mais entidades fontes de dados;

  - União (U) – Efetua a união de duas ou mais fontes de dados para a carga de uma entidade destino dos dados.
  - Junção (J) – Efetua a junção de duas ou mais fontes de dados, de acordo com critérios pré-definidos de forma a gerar um registro único contendo dados das duas entidades fontes de dados para ser carregado na entidade destino dos dados.
  - Diferença (Dif) – Identifica a diferença entre duas fontes de dados, a partir de critérios pré-definidos, para localizar um subconjunto de uma das entidades fontes que poderá ser

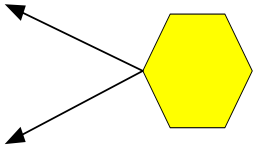
utilizado no processo de carga das entidades destino dos dados.

- Detecção de Atualização (DifUPD) – Identifica se ocorreu alteração de algum dado da entidade fonte de dados utilizando para isso mecanismos apropriados de comparação de forma a identificar e segregar somente os registros que foram alterados para sua utilização na atualização dos registros da entidade destino dos dados.

- **Operações de Arquivo** são aquelas que dão suporte operacional ao processo de ETL;
  - Conversões EBCDIC para ASCII (EB2AS) – Utiliza-se esses tipos de conversões para compatibilizar dados entre plataformas heterogêneas.
  - Classificação de Arquivo (Sort) – Possibilita a classificação dos dados das entidades fonte de dados.

- **Operações de Transferência** que são operações que dão suporte de comunicação ao processo de ETL.
  - FTP (FTP) – Possibilita a utilização de processos de transferência de dados entre plataformas heterogêneas utilizando para isso o Protocolo de Transferência de Arquivos.
  - Comprimir / Descomprimir (Z/dZ) – Efetua a compressão ou expansão dos dados na entidade destino de dados de forma a racionalizar o espaço em disco utilizado pela entidade.
  - Criptografar / Descriptografar (Cr/dCr) – Institui processos de criptografia para garantir a segurança das informações na entidade destino dos dados.

2.4.1.3 Restrição [4]



Uma restrição é uma seleção ou imposição de valor aplicado a um atributo ou conjunto de atributos. A figura 2.10 apresenta um exemplo de sua utilização onde é aplicado um valor nulo a vários atributos na entidade “CLIENTE Destino”.

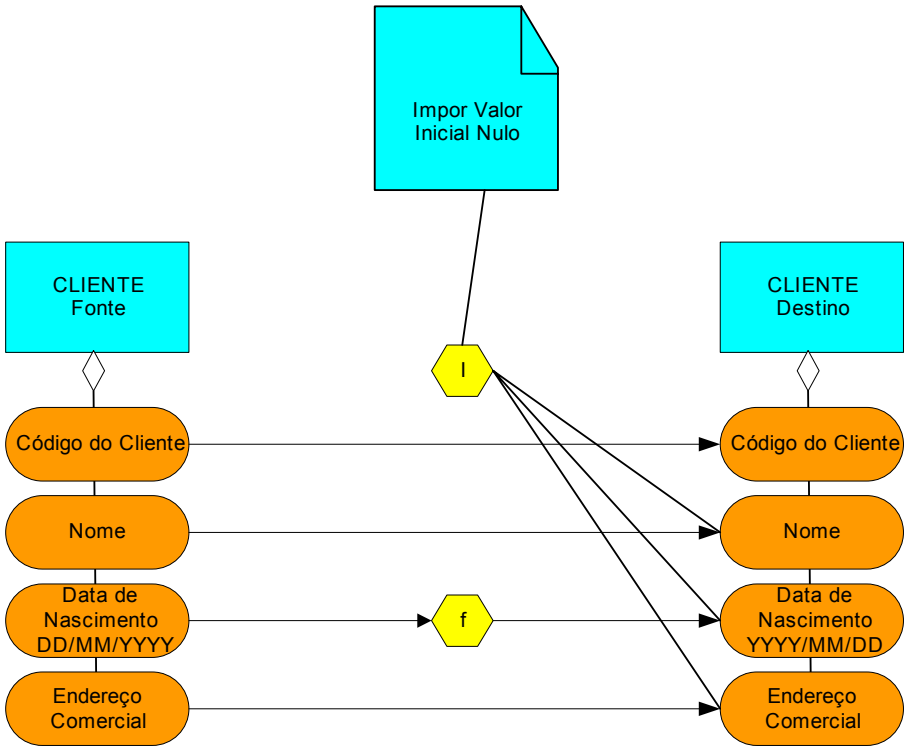


Figura 2. 10. Exemplo de utilização de uma restrição [4]

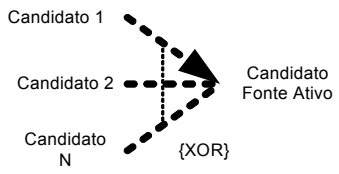
2.4.1.4 Relacionamentos

No caso do modelo apresentado os relacionamentos indicam, com exceção dos Relacionamentos Candidatos, as formas pelas quais uma transformação

relaciona-se com os demais itens gráficos do modelo, assim sendo o foco de cada relacionamento não é a transformação que é inerente à representação gráfica do relacionamento, mas sim o relacionamento dessa transformação com os demais itens do modelo.

Os tipos de relacionamento são : - Relacionamento Candidatos; Relacionamentos Fornecedores; Composição Serial e Relacionamentos “Parte de”. A forma como esses relacionamentos podem ser utilizados é apresentada a seguir.

Relacionamentos Candidatos [4]



Relacionamentos candidatos são aqueles onde existe mais de uma entidade fonte de dados para a carga da mesma tabela no sistema destino. São capturados pela representação acima que denota o fato de que uma tabela no sistema destino pode ser carregada por mais de uma fonte de dados dependendo de condições específicas. A figura 2.11. apresenta a utilização de um relacionamento candidato. A eleição de qual das entidades efetuará o povoamento da entidade destino é uma condição ambiental que é verificada no momento em que a execução do processo for iniciada. Essa condição ambiental é formalmente documentada por meio de uma nota explicativa. A seta na representação do relacionamento indica que somente uma das entidades fontes candidatas é eleita para efetuar o povoamento da entidade destino. Essa entidade fonte candidata é conhecida como Candidato Fonte Ativo.

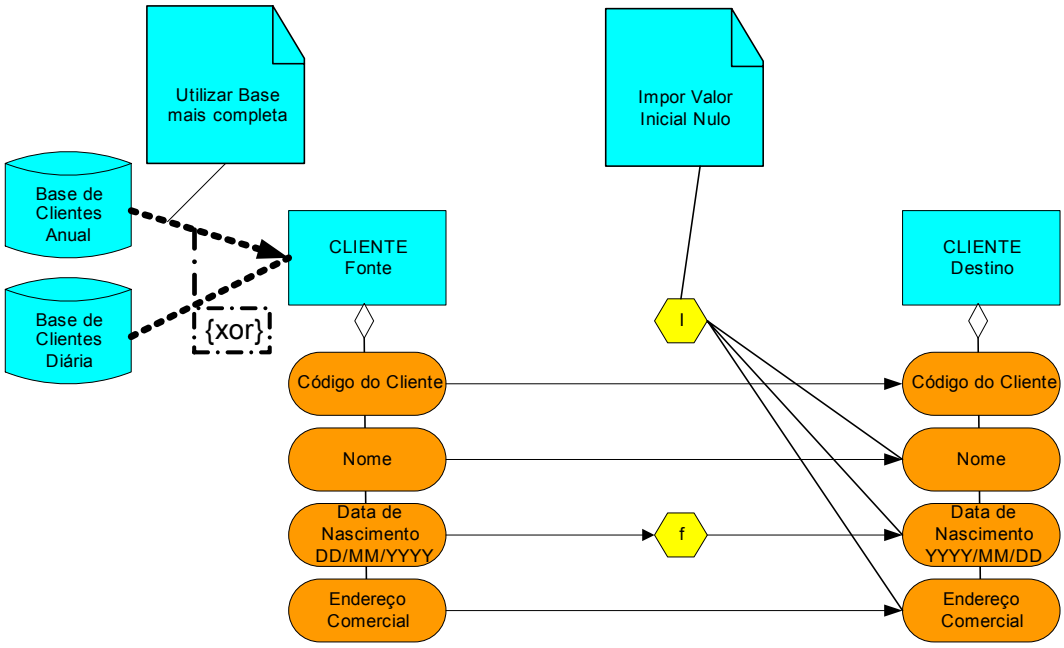
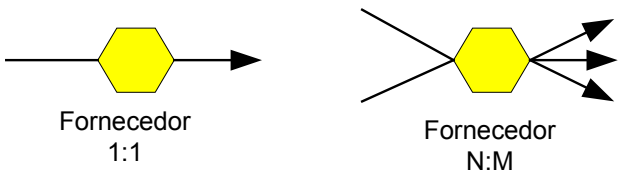


Figura 2. 11. Exemplo de utilização de um relacionamento candidato [4]

Dessa forma considera-se que um relacionamento candidato deva ser definido por:

- Entidades fonte candidatas
- Candidato Fonte Ativo
- Condição de seleção

Relacionamentos Fornecedores [4]



Esses relacionamentos mostram como uma transformação se relaciona com os demais itens do modelo. Têm a função de mapear um conjunto de atributos de entrada para um conjunto de atributos de saída.

A figura 2.9 (Item 2.4.1.2) apresenta um relacionamento fornecedor do tipo 1:1 entre as entidades "CLIENTE Fonte" e "CLIENTE Destino" para o atributo Data de Nascimento.

A figura 2.12 apresenta um relacionamento fornecedor do tipo N:M entre as entidades "ESTOQUE Fonte Detalhado" e "ESTOQUE Destino Agregado" onde uma função de agregação é utilizada para povoar uma entidade destino agregada pela data de validade, contendo os totais de produtos que estão para perder a validade na data e qual o valor total do estoque que estará perdendo a validade.

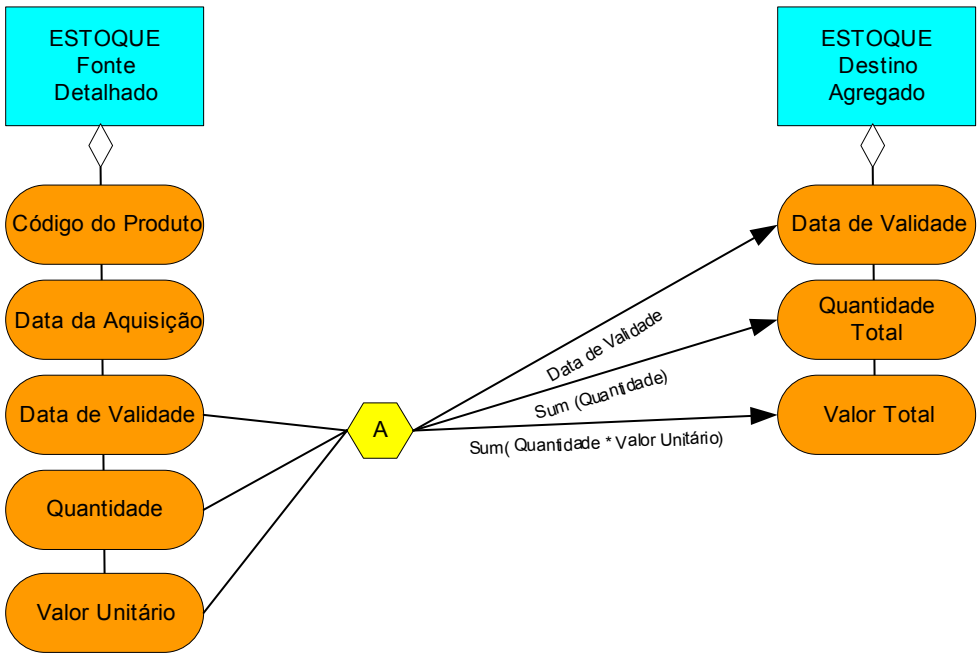
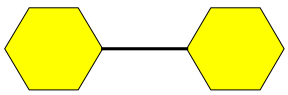


Figura 2. 12 - Exemplo de relacionamento fornecedor do tipo N:M [4]

Composição Serial [4]



Em alguns casos é possível que seja necessário combinar várias transformações para descrever um relacionamento entre o atributo de uma entidade fonte e o atributo de uma entidade destino; nestes casos é utilizada

uma composição serial de relacionamentos para evidenciar a seqüência de transformações necessárias.

A figura 2.13 apresenta uma composição serial para o atributo data de validade abrangendo duas transformações básicas necessárias. A primeira das transformações é uma inversão de datas e a segunda é uma função de agregação, que agrega valores pela data de validade no formato YYYY/MM/DD. Observa-se a utilização de uma composição serial unindo as duas transformações necessárias e a utilização de um relacionamento fornecedor do tipo N:M unido os demais atributos.

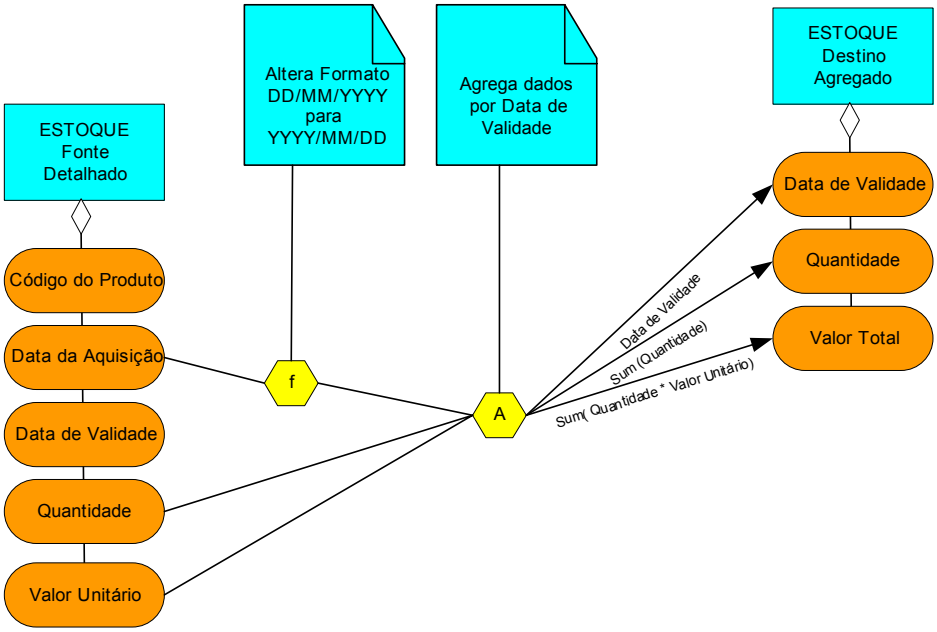
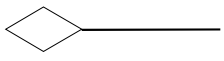


Figura 2. 13. Exemplo de relacionamento de composição serial [4]

Relacionamento Parte de [4]



É utilizado para caracterizar uma entidade como sendo um agregado de atributos relacionados a um conceito. Na figura 2.14 é utilizado um relacionamento Parte de, no contexto de uma entidade, relacionando os atributos ao conceito CLIENTE.

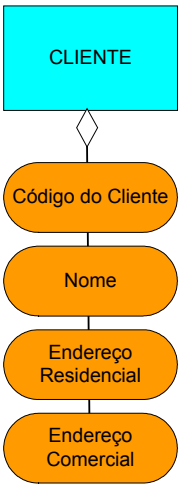
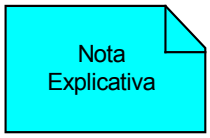


Figura 2. 14. Exemplo de relacionamento de Parte de [4]

2.4.1.5 Notas Explicativas [4]



São rótulos informais que denotam comentários extras e são utilizados para:

- Comentários e explicações simples;
- Explicações sobre a semântica das transformações;
- Informação sobre as restrições de ETL e seus efeitos sobre os diferentes aspectos do processo de ETL..

2.4.2 Metodologia de Aplicação

Os itens gráficos componentes do modelo necessitam de um maior detalhamento para sua utilização em situação real, já que a simples citação do ferramental para descrição de um processo de ETL não indica a forma correta de sua utilização. É necessário, portanto, uma metodologia de aplicação para orientar a construção correta dos modelos. A aplicação dessa metodologia é apresentada a seguir.

2.4.2.1 Passo 1 - Identificação das Fontes Apropriadas

O primeiro passo na metodologia do Modelo Conceitual [4] é identificar as estruturas de dados mais apropriadas para serem utilizadas como fontes de dados para o processo de ETL. Identificadas as estruturas é efetuado um desenho do cenário de execução do processo de ETL.

A figura 2.15. apresenta um cenário planejado para efetuar a carga de uma base de Vendas Totais num Data Warehouse. Neste cenário observa-se um processo de Junção onde as fontes de dados relativas a Valores Vendidos pelas Filiais e Custos das Vendas das Filiais (PS1 e PS2) gerando uma nova estrutura de dados chamada de Vendas de Filiais. Esta estrutura de dados, por sua vez, é unida com a fonte de dados de Vendas da Matriz resultando na carga da base de Vendas Totais.

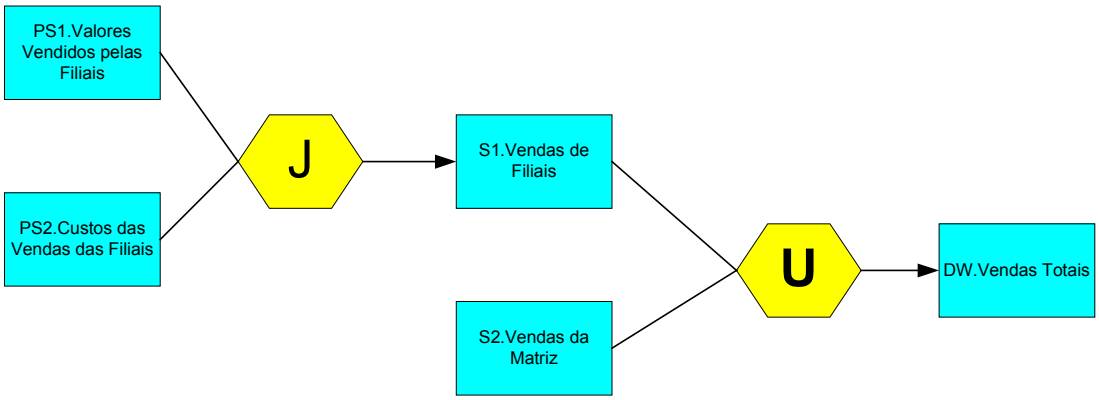


Figura 2. 15. Cenário para geração de Vendas Totais no DW [4]

2.4.2.2 Passo 2 - Identificação das Fontes Candidatas

O segundo passo na metodologia é a identificação das fontes candidatas que poderão ser utilizadas no processo. Fontes candidatas são aquelas que dependendo de variáveis ambientais podem ou não ser eleitas como sendo as fontes de dados ideais para efetuar a carga das estruturas destino dos dados.

A figura 2.16 apresenta o cenário de carga para a base de dados de Vendas Totais onde foram incluídas as informações de fontes candidatas. Nesta figura observa-se a Nota Explicativa onde é descrito que : - No caso de fechamento anual, a fonte mais indicada para a carga da entidade Vendas da Matriz é a entidade Vendas Anuais da Matriz. Caso contrário, a entidade mais adequada é a Vendas da Matriz até o Mês Corrente.

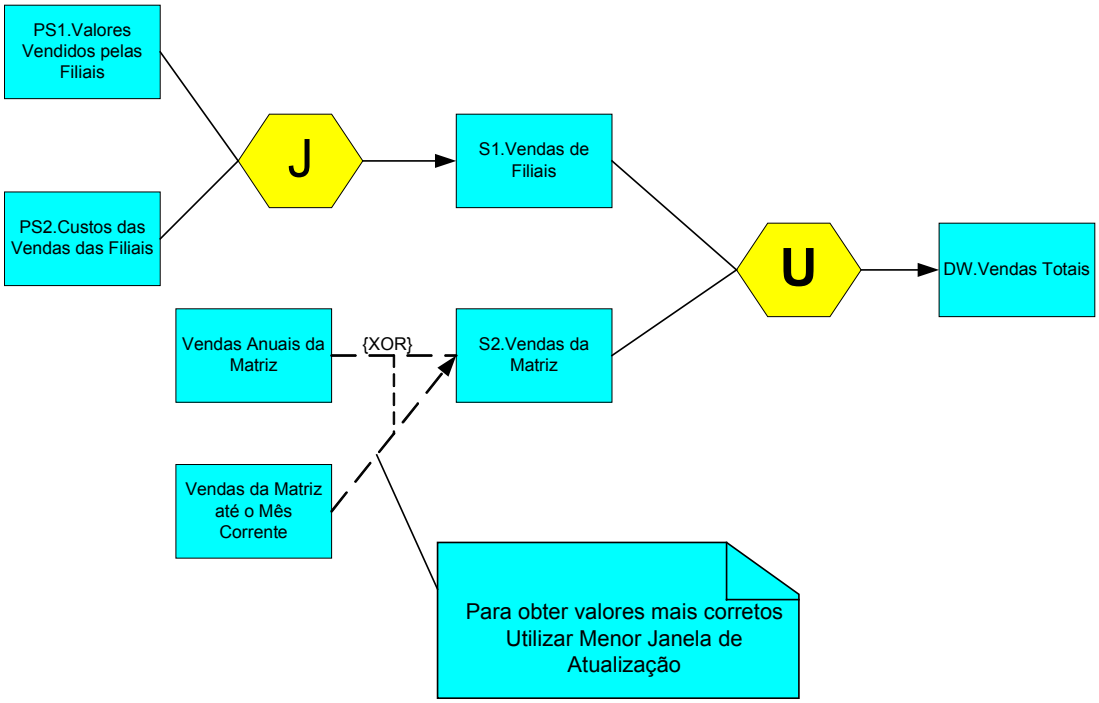


Figura 2. 16. Cenário de ETL incluindo Fontes Candidatas

2.4.2.3 Passo 3 - Mapeamento dos Atributos entre Fontes e Destinos

O terceiro passo da metodologia é efetuar o mapeamento dos atributos das estruturas fonte de dados para as estruturas destino dos dados.

A figura 2.17. apresenta o detalhamento do mapeamento do cenário montado nos passos 1 e 2 da metodologia. Observa-se as fontes PS1 e PS2 relativos às vendas das filiais e o processo de Junção efetuado para geração da base de dados S1 de Vendas de Filiais. Observa-se também que a base de dados de Vendas da Matriz não possui o mesmo detalhamento da base

de Vendas de Filiais e que é necessário efetuar uma Agregação para garantir a mesma granulidade na base de dados de DW.Vendas Totais; além disso observa-se que é necessário efetuar uma substituição da chave Nro. Nota Fiscal por uma chave substituta gerada pela transformação identificada como SK e que garante que na carga não ocorrerá problemas de chaves duplicadas em função do número da nota fiscal. A restrição de PK (Chave primária) efetuará a verificação de chave primária para o conjunto de atributos Chave + Código do Produto + Data. A transformação identificada por NN efetuará a verificação do conteúdo do campo Custo da Venda proveniente da fonte S1 permitindo somente a carga de valores Não Nulos na base de dados destino. As transformações identificadas pela letra f são aplicações de funções que realizam operações específicas com o objetivo de efetuar a padronização dos dados garantindo a uniformidade de padrões na base de dados destino.

2.4.2.4 Passo 4 - Anotação das Restrições de Tempo de Execução

O quarto passo da metodologia é indicar as restrições de tempo para execução do cenário planejado. Os processos de ETL que efetuam o processamento em lote devem ter as restrições de tempo sob controle pois a quantidade de registros que devem ser processados é muito grande e esse processamento consome muitos recursos.

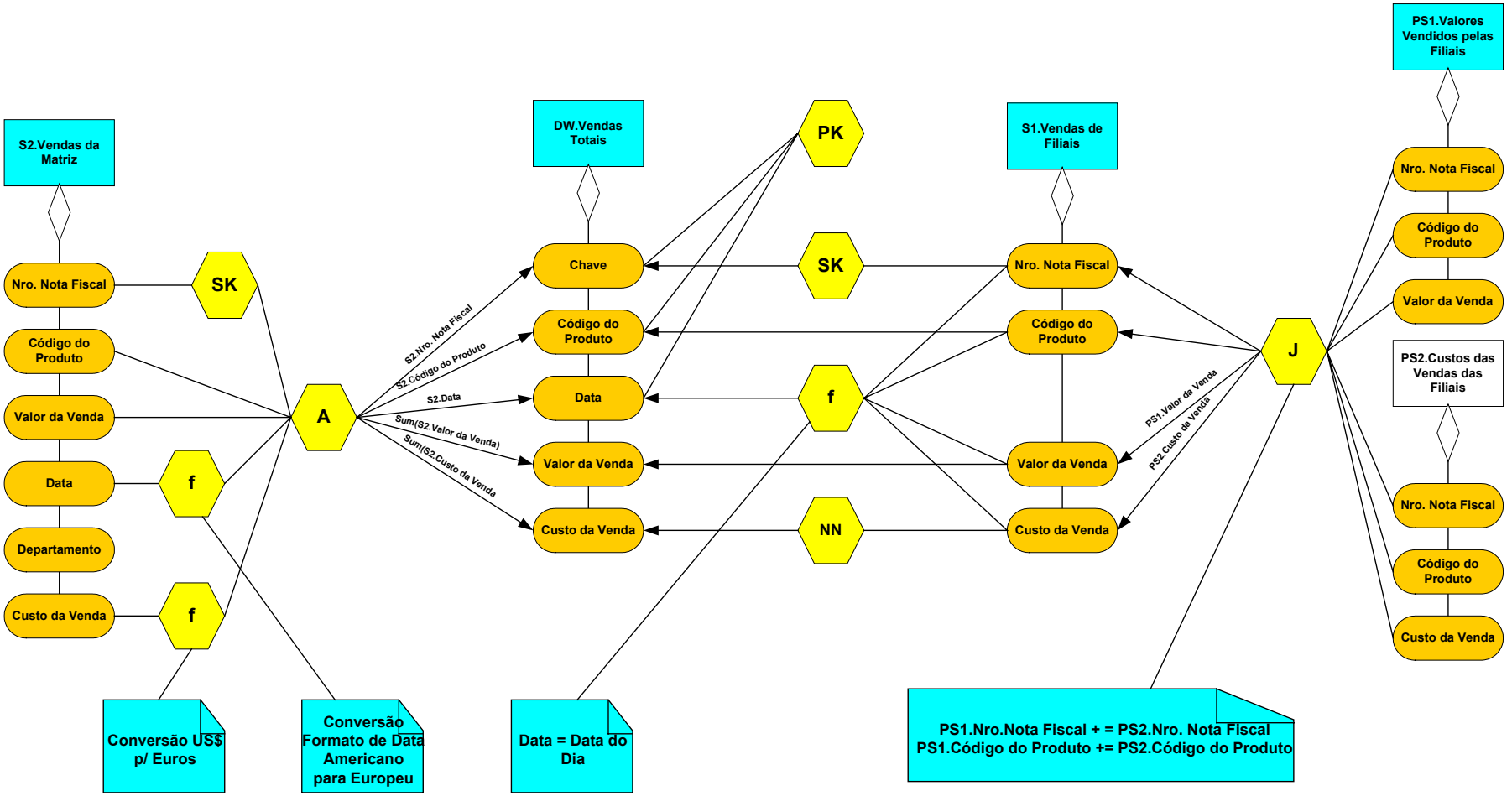


Figura 2. 17 - Detalhamento do Mapeamento

A figura 2.18. apresenta o cenário planejado onde a restrição de tempo foi descrita em nota explicativa.

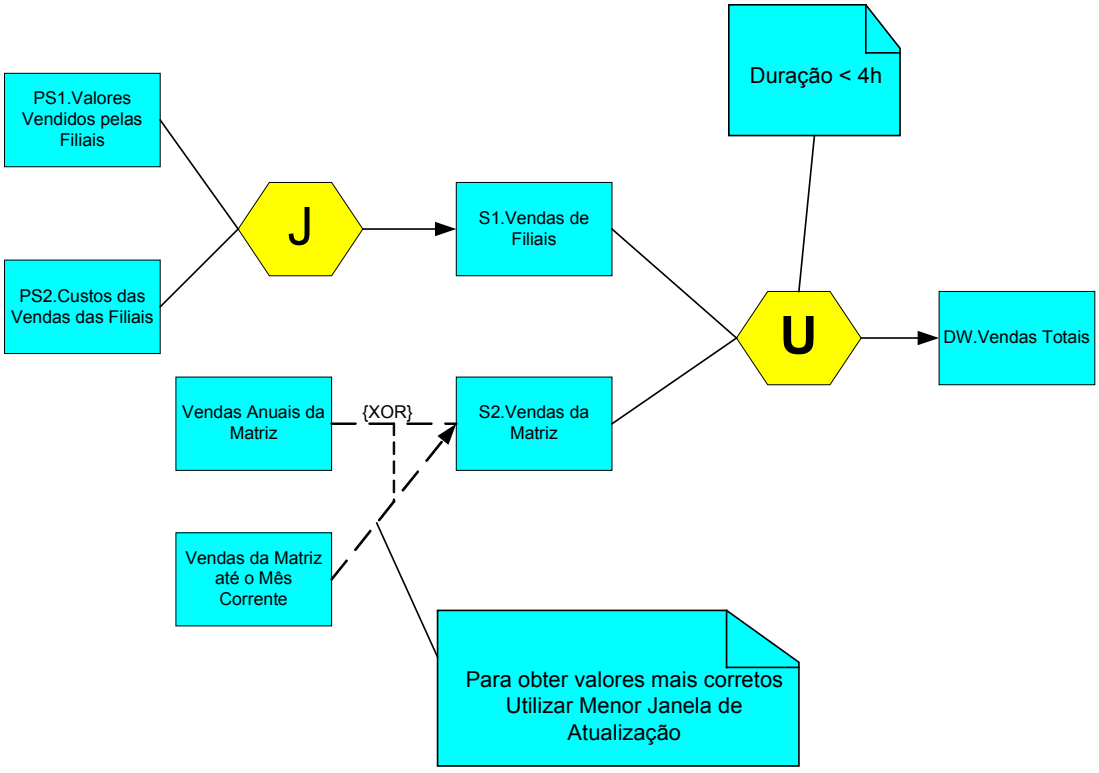


Figura 2. 18. Cenário de execução do processo com Informação de restrição de tempo de execução

2.4.2.5 Passo 5 - Montagem do Diagrama final

O último passo da metodologia é a montagem do diagrama final. Neste passo o mapeamento dos atributos, efetuado no passo 3, é complementado com informações do cenário planejado.

A figura 2.19. apresenta o diagrama final completo onde pode-se observar a inserção de informações do cenário planejado como a informação de que a base de Vendas Totais é resultado da União das bases de Vendas das Filiais e da Matriz, assim como as informações relativas às fontes candidatas existentes no processo.

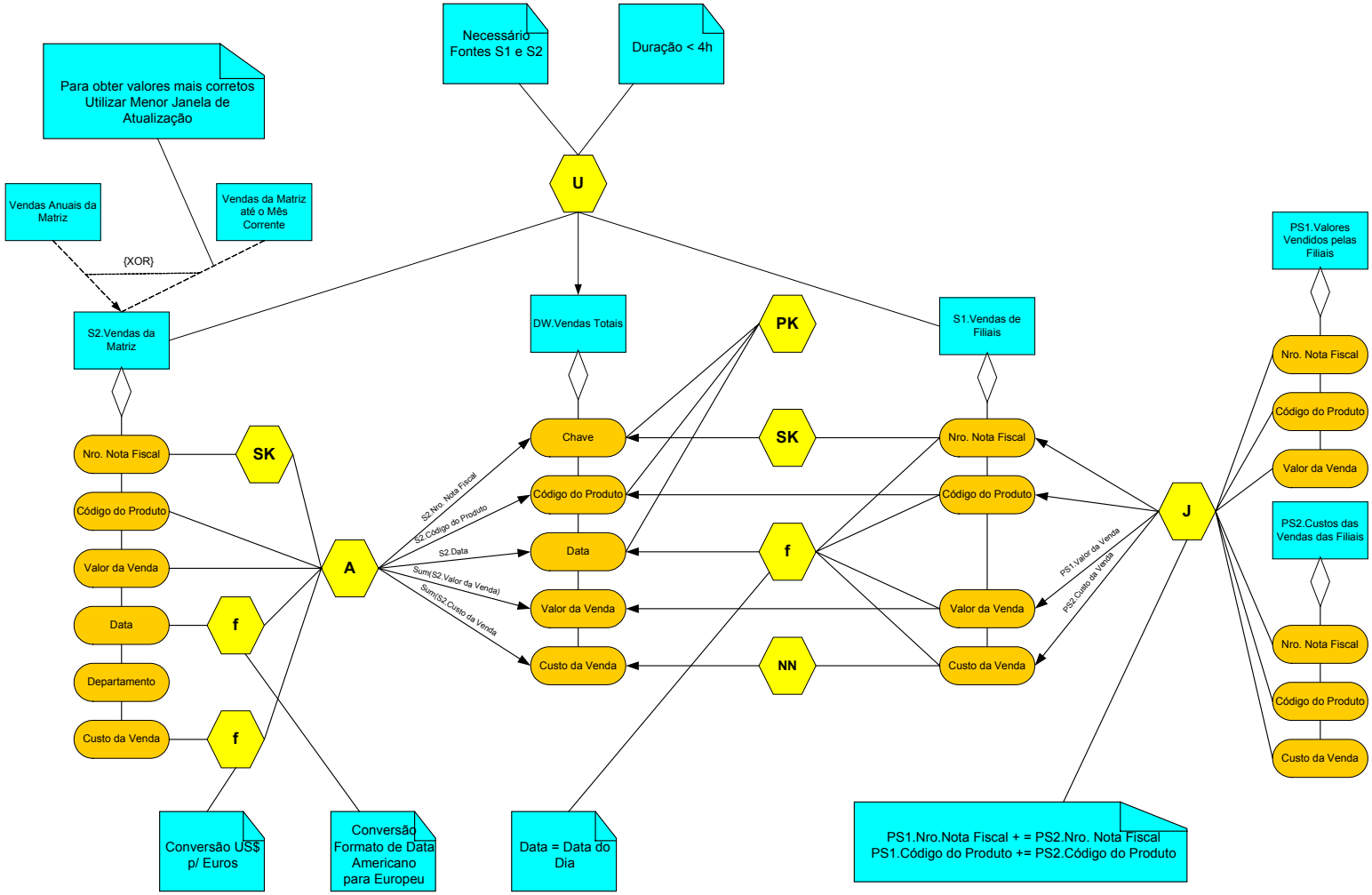


Figura 2. 19 - Diagrama Final

2.5 Conclusão

A metodologia e a notação utilizada no Modelo Conceitual [4] possibilitam a especificação de processos de ETL. O diagrama final apresentado no último passo da metodologia fornece dados suficientes para possibilitar a construção de um processo de ETL a partir desse diagrama, porém, é necessário avaliar o modelo. Para tanto, é apresentado um estudo de caso utilizando os modelos Dissertativo e Conceitual [4] o que permitirá uma avaliação mais realista do Modelo Conceitual [4].

3 Modelo Dissertativo Atual

3.1 Introdução

Este capítulo, aborda um estudo de caso envolvendo uma definição de processo de ETL, utilizando o Modelo Dissertativo que é um modelo de especificação no qual todas as etapas e transformações necessárias ao processo são detalhadamente descritas pelo analista por meio de um texto explicativo.

O Modelo Dissertativo é o modelo de especificação atualmente utilizado no ambiente de estudo e sua utilização no estudo de caso é necessária para obter meios de comparação com o Modelo Conceitual [4] que é aplicado no capítulo seguinte.

O Modelo Dissertativo é composto de várias partes que são apresentadas e comentadas de forma a possibilitar o entendimento das necessidades de uma especificação de processo de ETL neste modelo.

O ambiente deste estudo de caso é o núcleo de desenvolvimento de software de uma das maiores empresas do ramo financeiro do mercado brasileiro. A empresa utiliza os processos de ETL para alimentar bases analíticas que tem o objetivo de possibilitar o acompanhamento da vida financeira de clientes de forma a ofertar melhores produtos e serviços.

Para o estudo de caso, uma especificação de processo de ETL no Modelo Dissertativo é selecionada observando o critério da maior abrangência possível de funções e transformações passíveis de utilização dentro de um processo de ETL. Este critério foi escolhido porque permite a verificação da capacidade do modelo em suportar as mais diversas situações exigidas.

3.2 Modelo Dissertativo

O Modelo Dissertativo, hoje em uso na empresa, busca atender as necessidades de comunicação entre os analistas de sistemas que criam as especificações de processos de ETL e programadores que constroem esses processos a partir das especificações. O Modelo é um documento composto pelos seguintes itens:

- **Objetivo/Função**

  É feita uma breve descrição dos objetivos principais do processo de ETL.

- **Termo de Aditamento**

  Documenta todas as alterações efetuadas numa especificação de um processo de ETL no decorrer do tempo. Essas alterações podem ser a inclusão ou retirada de um atributo ou mesmo uma alteração de configuração no software.

  Através de um identificador (símbolo gráfico) o programador pode localizar os pontos onde foram inseridas e/ou retiradas partes da especificação, de forma a identificar todas as funcionalidades que foram implementadas ou retiradas em cada uma das alterações.

- **Arquivos de Entrada e Saída**

  Fornece ao programador as informações sobre quais são as entidades fontes de dados e quais são as entidades destino dos dados.

- **Tratamento de Arquivos**

  Especifica os procedimentos a serem seguidos em caso de arquivo vazio ou chaves duplicadas detectadas nas entidades fontes de dados.

- **Descrição da Integração**

  Especifica como as fontes devem se integrar para, ao final, obter as informações necessárias e efetuar a carga das entidades destino dos dados.

- **Funções Utilizadas**

  Especifica quais as funções ou módulos de código que devem ser utilizados pelo processo de ETL para obtenção e/ou tratamento de dados. Ex: Funções de Conversão de Moedas, Funções de Tratamento de Datas, etc. Essas funções ou módulos podem residir em bibliotecas de funções que também devem ser indicadas neste tópico.

- **Regras de Transformação**

  A transferência de dados entre sistemas heterogêneos é possível quando o dado do sistema origem tem o mesmo domínio de valores do sistema destino. Nos casos em que isso não acontece, é necessária uma série de verificações e conversões para compatibilizar os domínios de valores do sistema fonte com os domínios de valores do sistema destino. As regras de transformação indicam como deve ser essa compatibilização, descrevendo claramente os passos necessários para transformar o dado de forma a ser carregado no sistema destino sem prejuízo de seu significado original.

3.3 Especificação de Processo de ETL

Observa-se que nas situações do dia a dia não são utilizadas muitas das funcionalidades disponíveis em um processo de ETL, dessa forma, selecionar uma especificação de processo de ETL que contenha a maior quantidade possível de funcionalidades para sua utilização no estudo de caso, torna-se uma tarefa difícil. Por esse motivo, foi selecionada uma

especificação envolvendo um processo de melhoria da qualidade de dados utilizando para isso uma outra fonte de dados de CEP's do Correio e, a partir dessa especificação, são incluídas outras funcionalidades ETL.

Para melhor entendimento do problema envolvendo a melhoria de dados a partir da utilização de uma fonte de dados de CEP's do Correio considera-se uma Fonte de Dados (A) que possui um campo CEP cujo conteúdo era originalmente composto por um número de cinco posições numéricas. Com a mudança do CEP para oito posições numéricas, as três últimas posições do CEP (Complemento do CEP) foram preenchidas com Zeros. A partir da adequação do sistema, novos registros e novos CEP's foram sendo adicionados a Fonte de Dados (A).

O resultado é que a Fonte de Dados (A), com relação ao seu campo CEP, é um conjunto heterogêneo onde co-existem dados de CEP's corretos e incorretos (CEP's cujo complemento encontra-se preenchido com Zeros).

Com o procedimento de melhoria de qualidade utilizando-se a base dos Correios, que é considerada como sendo a Fonte de Dados (B), pretende-se corrigir o conteúdo do campo CEP da Fonte de Dados (A).

O procedimento para a melhoria da qualidade do campo CEP da Fonte de Dados (A) segue os seguintes passos:

- Divisão do campo Endereço da Fonte de Dados (A) para Tipo de Logradouro e Logradouro
- Divisão do campo CEP da Fonte de Dados (A) para CEP e Complemento do CEP
- Divisão do campo Endereço da Fonte de Dados (B) para Tipo de Logradouro e Logradouro
- Divisão do campo CEP da Fonte de Dados (B) para CEP e
- Complemento de CEP
- Padronização do Logradouro das fontes de dados (A) e (B). A Padronização é um processo de qualidade de dados que, por meio de

um algorítimo, converte partes menores de um conteúdo para denominadores comuns, dessa forma torna-se mais fácil identificar conteúdos semelhantes. Ex: Considerando dois campos de Logradouro cujos conteúdos sejam "Rua Dr. Arnaldo" e "R. Doutor Arnaldo", aplicando-se a regra de padronização de logradouro nesses dois campos, pode-se obter como resultado "R. Dr. Arnaldo" e, dessa forma, é possível efetuar uma comparação exata do logradouro das fontes de dados (A) e (B). Essa regra é possível pois pode-se montar um algorítimo para converter "Rua" para "R." e "Doutor" para "Dr.".

- Identificação entre as Fontes de Dados (A) e (B) utilizando uma comparação exata de CEP e aproximada de Logradouro. A identificação são processos de qualidade de dados que possuem o objetivo identificar, por meio de algorítimos estatísticos, informações iguais a partir de dados não estruturados. A Padronização nem sempre consegue "corrigir" todo o conteúdo de um campo e por isso é feita uma comparação aproximada no campo Logradouro onde é selecionada o Logradouro de (B) que mais se aproxima do Logradouro de (A).
- Localizado o registro da Fonte de Dados (B) (Correios) que possui um endereço corresponde ao endereço da Fonte de Dados (A), se obtém o Complemento de CEP correto para o melhoria do dado CEP da Fonte de Dados (A), o que possibilita a correção do seu conteúdo.

Selecionada a especificação envolvendo processos de qualidade de dados são incluídas as seguintes funcionalidades:

- Filtragem de dados – São selecionados somente os registros de Pessoas Físicas.
- Re-formatação de Dados – Campos do tipo Data são re-formatados para o formato YYYYMMDD.
- Re-estruturação das Chaves – É gerada uma nova chave para identificar os grupos de moradores de uma mesma residência.

- Especificação de Valores Fixos – Um dos campos é preenchido com a data do dia.
- Identificação de Registros Alterados – O resultado de todo o processo é comparado com o resultado obtido no processamento anterior e as diferenças são carregadas para uma entidade de histórico.

Dessa forma obtém-se uma especificação de processo de ETL que abrange as mais importantes funcionalidades e que, portanto, pode ser utilizada no estudo de caso.

Selecionada a especificação a ser utilizada no estudo de caso, é necessário transcrevê-la para o Modelo Dissertativo. A figura 3.1 apresenta a especificação do processo de ETL selecionada para o estudo de caso, transcrita para o Modelo Dissertativo.

Objetivo / Função
Geração de arquivo de clientes, com CEP corrigido, que agrupa moradores de uma mesma residência possibilitando sua utilização na redução de custos em processos de envio de mala postal da empresa

Termo de Aditamento		
Data	**Descrição das Alterações**	**Responsável**
30/08/2004	Versão Inicial	José

Arquivos do Processo			
Tipo	**Nome**	**Layout**	**Descrição**
Entrada	E1	L1	Arquivo de Clientes
Entrada	E2	L2	Correios
Entrada	E3	L6	Cadastro de Clientes (Anterior)
Trabaho	T1	L1	Clientes Pessoa Física
Trabaho	T2	L3	Clientes Pessoa Física Dividido
Trabalho	T3	L4	Correios Dividido
Trabalho	T4	L3	Clientes Pessoa Física Padronizado
Trabalho	T5	L4	Correios Padronizado
Trabaho	T6	L5	Clientes Pessoa Física Enriquecido
Trabalho	T7	L5	Clientes Pessoa Física (Atualizações)
Trabalho	T8	L5	Clientes Pessoa Física (Inclusões)
Trabalho	T9	L5	Cadastro de Clientes (Atualizado)
Trabalho	T10	L5	Novos Clientes Pessoa Física
Saída	S1	L6	Cadastro de Clientes (Novo)

Figura 3. 1- Processo Selecionado no Modelo Dissertativo

Tratamento de Arquivos
Todos os arquivos são do tipo seqüenciais sem chaves de acesso.

Descrição da Integração	
Passo	**Descrição**
1	A partir do arquivo E1, gerar o arquivo T1 selecionando somente os registros de pessoas físicas, re-formatando a data de nascimento do cliente e movendo a data do sistema para a referência do arquivo.
2	A partir do arquivo T1, gerar o arquivo T2 dividindo o campo endereço para Tipo de logradouro e Logradouro; dividir o campo CEP para CEP e Complemento de CEP
3	A partir do arquivo E2, gerar o arquivo T3 dividindo o campo CEP em CEP e Complemento de CEP.
4	A partir do arquivo T2, gerar o arquivo T4 padronizando os campos de Tipo de Logradouro utilizando para isso a função P1, Logradouro utilizando para isso a função P2.
5	A partir do arquivo T3, gerar o arquivo T5 padronizando os campos de Tipo de Logradouro utilizando para isso a função P1, Logradouro utilizando para isso a função P2.
6	Para cada registro de T4 localizar os registros de T5 com o mesmo CEP e para cada CEP localizado em T5 efetuar a comparação aproximada do Logradouro padronizado de T4 com T5 utilizando para isso a função Q1 de forma a localizar o registro de T5 correto para obter o Complemento de CEP. Identificado o registro correto de T5, gerar o arquivo T6 com o Complemento de CEP corrigido.
7	O arquivo resultante T6 será comparado com o arquivo resultante do processamento anterior E3, para obtermos um arquivo das atualizações T7.
8	O arquivo resultante T6 será comparado com o arquivo resultante do processamento anterior E3, para obtermos um arquivo das inclusões T8.
9	A partir do arquivo T7, efetuar a atualização dos dados de E3, gerando o arquivo T9.
10	A partir do arquivo T8, gerar o arquivo T10 e criar, para cada registro de T10, o campo ID_CHAVE que será definido como sendo uma chave identificadora de um Cliente. É um número seqüência crescente a partir do último cliente do Cadastro de Clientes.
11	Incluir os registros de T9 e T10 no arquivo S1,

Funções Utilizadas
D1 – Função de divisão de Endereço para Tipo de Logradouro e Logradouro D2 – Função de divisão de CEP em CEP e Complemento de CEP P1 – Função de padronização de Tipo de Logradouro P2 – Função de padronização de Logradouro Q1 – Função de identificação para Logradouro

Figura 3. 1 - Processo Selecionado no Modelo Dissertativo (Continuação)

Regras de Transformação – PASSO 1		
Obs: Selecionar os registros de Pessoas Físicas		
Origem - E1	**Transformação**	**Destino - T1**
CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
CGCCPF	Se CGCCPF_TIPO igual a 'F' (Pessoa Física) Cópia Simples Senão Rejeitar Registro	CGCCPF
NOME	Cópia Simples	NOME
ENDERECO	Cópia Simples	ENDERECO
NUMERO	Cópia Simples	NUMERO
COMPLEMENTO	Cópia Simples	COMPLEMENTO
CEP	Cópia Simples	CEP
NASCIMENTO	Re-formatar a data de DDMMYYYY para YYYYMMDD	NASCIMENTO
	Mover Data do Sistema	REFERENCIA

Regras de Transformação – PASSO 2		
Obs:		
Origem – T1	**Transformação**	**Destino – T2**
CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
CGCCPF	Cópia Simples	CGCCPF
NOME	Cópia Simples	NOME
ENDERECO	Separar o Tipo de Logradouro do Endereço e mover para Saída utilizando a função D1	LOGRADOURO_TIPO
ENDERECO	Separar o Logradouro do Endereço e mover para Saída utilizando a função D1	LOGRADOURO
NUMERO	Cópia Simples	NUMERO
COMPLEMENTO	Cópia Simples	COMPLEMENTO
CEP	Separar o CEP (Cinco primeiras posições) e mover para Saída utilizando a função D2	CEP
CEP	Separar o Complemento do CEP (Três últimas posições) e mover para Saída utilizando a função D2	CEP_COMPLEMENTO
NASCIMENTO	Cópia Simples	NASCIMENTO
REFERENCIA	Cópia Simples	REFERENCIA

Regras de Transformação – PASSO 3		
Obs:		
Origem – E2	**Transformação**	**Destino – T3**
CEP	Separar o CEP (Primeiras 5 posições) e mover para Saída utilizando a função D2	CEP
CEP	Separar o Complemento de CEP (Últimas três posições) e mover para Saída utilizando a função D2	CEP_COMPLEMENTO
LOGRADOURO_TIPO	Cópia Simples	LOGRADOURO_TIPO
LOGRADOURO	Cópia Simples	LOGRADOURO
CIDADE	Cópia Simples	CIDADE
ESTADO	Cópia Simples	ESTADO

Figura 3. 1 - Processo Selecionado no Modelo Dissertativo (Continuação)

Regras de Transfomação – PASSO 4		
Obs:		
Origem – T2	**Transformação**	**Destino – T4**
CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
CGCCPF	Cópia Simples	CGCCPF
NOME	Cópia Simples	NOME
LOGRADOURO_TIPO	Utilizar Padronização de Tipo de Logradouro P1	LOGRADOURO_TIPO
LOGRADOURO	Utilizar Padronização de Logradouro P2	LOGRADOURO
NUMERO	Cópia Simples	NUMERO
COMPLEMENTO	Cópia Simples	COMPLEMENTO
CEP	Cópia Simples	CEP
CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
NASCIMENTO	Cópia Simples	NASCIMENTO
REFERENCIA	Cópia Simples	REFERENCIA

Regras de Transformação – PASSO 5		
Obs:		
Origem – T3	**Transformação**	**Destino – T5**
CEP	Cópia Simples	CEP
CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
LOGRADOURO_TIPO	Utilizar Regra de Padronização de Tipo de Logradouro P1	LOGRADOURO_TIPO
LOGRADOURO	Utilizar Regra de Padronização de Logradouro P2	LOGRADOURO
CIDADE	Cópia Simples	CIDADE
ESTADO	Cópia Simples	ESTADO

Regras de Transformação – PASSO 6		
Obs: Efetuar a leitura do arquivo T4 e, para cada registro de T4 efetuar pesquisa no arquivo T5 utilizando para isso o campo CEP. Localizado os registros de T5 com o mesmo CEP, pesquisar qual dos registros de T5 possuem o mesmo Logradouro de T4 utilizando a comparação aproximada pela função Q1. Ao final gravar T6 conforme descrito.		
Origem – T4 + T5	**Transformação**	**Destino – T6**
T4.CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
T4.CGCCPF	Cópia Simples	CGCCPF
T4.NOME	Cópia Simples	NOME
T5.LOGRADOURO_TIPO	Cópia Simples	LOGRADOURO_TIPO
T5.LOGRADOURO	Cópia Simples	LOGRADOURO
T4.NUMERO	Cópia Simples	NUMERO
T4.COMPLEMENTO	Cópia Simples	COMPLEMENTO
T4.CEP	Cópia Simples	CEP
T5.CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
T4.NASCIMENTO	Cópia Simples	NASCIMENTO
T4.REFERENCIA	Cópia Simples	REFERENCIA

Figura 3.1 - Processo Selecionado no Modelo Dissertativo (Continuação)

Regras de Transformação – PASSO 7		
Obs: Para cada registro de T6, pesquisar E3 utilizando para isso a chave CGCCPF. Para cada registro localizado, gravar arquivo T7 conforme descrito a seguir.		
Origem – T6 X E3	**Transformação**	**Destino – T7**
T6.CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
T6.CGCCPF	Cópia Simples	CGCCPF
T6.NOME	Cópia Simples	NOME
T6.LOGRADOURO_TIPO	Cópia Simples	LOGRADOURO_TIPO
T6.LOGRADOURO	Cópia Simples	LOGRADOURO
T6.NUMERO	Cópia Simples	NUMERO
T6.COMPLEMENTO	Cópia Simples	COMPLEMENTO
T6.CEP	Cópia Simples	CEP
T6.CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
T6.NASCIMENTO	Cópia Simples	NASCIMENTO
T6.REFERENCIA	Cópia Simples	REFERENCIA

Regras de Transformação – PASSO 8		
Obs: Para cada registro de T6, pesquisar E3 utilizando para isso a chave CGCCPF. Caso não seja localizado registro correspondente no arquivo E3, gravar arquivo T8 conforme a seguir.		
Origem – T6 X E3	**Transformação**	**Destino – T8**
T6.CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
T6.CGCCPF	Cópia Simples	CGCCPF
T6.NOME	Cópia Simples	NOME
T6.LOGRADOURO_TIPO	Cópia Simples	LOGRADOURO_TIPO
T6.LOGRADOURO	Cópia Simples	LOGRADOURO
T6.NUMERO	Cópia Simples	NUMERO
T6.COMPLEMENTO	Cópia Simples	COMPLEMENTO
T6.CEP	Cópia Simples	CEP
T6.CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
T6.NASCIMENTO	Cópia Simples	NASCIMENTO
T6.REFERENCIA	Cópia Simples	REFERENCIA

Regras de Transformação – PASSO 9		
Obs: Para cada registro de E3 verificar se existe registro correspondente no arquivo T7 utilizando para isso a chave CGCCPF. Caso seja localizado, gravar S1 com dados de T7. Caso contrário, gravar S1 com dados de E3. O campo de identificação ID_CHAVE, sempre deve ser movido a partir de E3.		
Origem – T7 X E3	**Transformação**	**Destino – T9**
E3.CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
E3.CGCCPF	Cópia Simples	CGCCPF
T6.NOME E3.NOME	Se T6.CGCCPF = E3.CGCCPF Mover T6.NOME Senão Mover E3.NOME	NOME
T6.LOGRADOURO_TIPO E3.LOGRADOURO_TIPO	Se T6.CGCCPF = E3.CGCCPF Mover T6.LOGRADOURO_TIPO Senão Mover E3.LOGRADOURO_TIPO	LOGRADOURO_TIPO

Figura 3. 1 - Processo Selecionado no Modelo Dissertativo (Continuação)

T6.LOGRADOURO E3.LOGRADOURO	Se T6.CGCCPF = E3.CGCCPF Mover T6.LOGRADOURO Senão Mover E3.LOGRADOURO	LOGRADOURO
T6.NUMERO E3.NUMERO	Se T6.CGCCPF = E3.CGCCPF Mover T6.NUMERO Senão Mover E3.NUMERO	NUMERO
T6.COMPLEMENTO E3.COMPLEMENTO	Se T6.CGCCPF = E3.CGCCPF Mover T6.COMPLEMENTO Senão Mover E3.COMPLEMENTO	COMPLEMENTO
T6.CEP E3.CEP	Se T6.CGCCPF = E3.CGCCPF Mover T6.CEP Senão Mover E3.CEP	CEP
T6.CEP_COMPLEMENTO E3.CEP_COMPLEMENTO	Se T6.CGCCPF = E3.CGCCPF Mover T6.CEP_COMPLEMENTO Senão Mover E3.CEP_COMPLEMENTO	CEP_COMPLEMENTO
T6.NASCIMENTO E3.NASCIMENTO	Se T6.CGCCPF = E3.CGCCPF Mover T6.NASCIMENTO Senão Mover E3.NASCIMENTO	NASCIMENTO
T6.REFERENCIA E3.REFERENCIA	Se T6.CGCCPF = E3.CGCCPF Mover T6.REFERENCIA Senão Mover E3.REFERENCIA	REFERENCIA
E3.ID_CHAVE	Cópia Simples	ID_CHAVE

Regras de Transformação – PASSO 10

Obs:
Acrescentar aos registros de T8 o campo ID_CHAVE de S1 que é a chave substituta do registro. Ela é um número seqüencial crescente a partir do último cliente do Cadastro de Clientes

Origem – T8	Transformação	Destino – T10
CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
CGCCPF	Cópia Simples	CGCCPF
NOME	Cópia Simples	NOME
LOGRADOURO_TIPO	Cópia Simples	LOGRADOURO_TIPO
LOGRADOURO	Cópia Simples	LOGRADOURO
NUMERO	Cópia Simples	NUMERO
COMPLEMENTO	Cópia Simples	COMPLEMENTO
CEP	Cópia Simples	CEP
CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
NASCIMENTO	Cópia Simples	NASCIMENTO
REFERENCIA	Cópia Simples	REFERENCIA
	Mover valor máximo de ID_CHAVE em S1 + 1	ID_CHAVE

Figura 3. 1 - Processo Selecionado no Modelo Dissertativo (Continuação)

Regras de Transformação – PASSO 11		
Obs: Efetuar uma união entre os arquivos T9 e T10 para carga da saída S1.		
Origem – T9 e T10	**Transformação**	**Destino – S1**
CGCCPF_TIPO	Cópia Simples	CGCCPF_TIPO
CGCCPF	Cópia Simples	CGCCPF
NOME	Cópia Simples	NOME
LOGRADOURO_TIPO	Cópia Simples	LOGRADOURO_TIPO
LOGRADOURO	Cópia Simples	LOGRADOURO
NUMERO	Cópia Simples	NUMERO
COMPLEMENTO	Cópia Simples	COMPLEMENTO
CEP	Cópia Simples	CEP
CEP_COMPLEMENTO	Cópia Simples	CEP_COMPLEMENTO
NASCIMENTO	Cópia Simples	NASCIMENTO
REFERENCIA	Cópia Simples	REFERENCIA
ID_CHAVE	Cópia Simples	ID_CHAVE

Figura 3. 1 - Processo Selecionado no Modelo Dissertativo (Continuação)

Observa-se no documento de especificação no Modelo Dissertativo as seguintes funcionalidades previstas para o estudo de caso:

Qualidade de Dados

O documento de especificação no Modelo Dissertativo cita a utilização de processos de qualidade de dados na construção do processo de ETL.

Quando é solicitada a divisão de campos (Regras de Transformação – Passo 2 e 3, descritos na tabela 3.1 das páginas 52 e 53) ou a padronização de conteúdos (Regras de Transformação - Passos 4 e 5, descritos na tabela 3.1 da página 53), são utilizadas funcionalidades de qualidade de dados inseridas no processo de ETL. As funções D1 e D2 que são descritas no quadro de “Funções Utilizadas” da tabela 3.1 (Pág. 46) e são funcionalidade preparadas especialmente para as atividades de divisão dos campos CEP e Endereço respectivamente.

Observa-se que a divisão do campo CEP não é uma funcionalidade extremamente complexa pois o CEP é um dado relativamente estruturado; já o Endereço é um dado essencialmente não estruturado. Um campo

Endereço, sendo preenchido por intervenção humana, comporta quaisquer formatos e, a função D2 deve, portanto, possuir um algorítimo que leve em consideração a maior quantidade possível de possibilidades de preenchimento do campo Endereço possibilitando a identificação das partes componentes de um endereço de forma a possibilitar sua separação.

O mesmo observa-se com relação às funções de padronização P1 e P2 que também são descritas no quadro de “Funções Utilizadas” da tabela 3.1 (Pág. 46) . Essas funções, por meio de algorítimos que procuram prever a maior quantidade de possibilidades de preenchimento de cada uma das partes separadas, permite efetuar uma “correção” dos conteúdos para um conteúdo padrão.

Filtragem de Dados

Observa-se que a utilização de filtros neste estudo de caso abrange somente a filtragem de registros de Pessoas Físicas (Passo 1 descrito na tabela 3.1 da página 52). Esse tipo de filtro é extremamente simples e normalmente é implementado de forma fixa no código do programa.

De uma forma geral filtrar dados significa aplicar um teste condicional aos dados de forma a identificar se estes se encontram ou não em conformidade com requisitos estabelecidos. Isto possibilita estabelecer atitudes a serem tomadas nos casos de conformidade ou não com esses requisitos. No ambiente de estudo são utilizadas três formas de filtragem:

- Condição de teste fixa no software para a verificação de um pequeno conjunto de valores ou faixas de valores. Esse tipo de teste é chamado como Condição de Teste Simples e é o tipo de teste que é aplicado no Passo 1 deste estudo de caso.
- Condição de teste na qual é efetuada uma pesquisa em uma Tabela de Verificação que contém os valores de domínio válidos. Esse tipo de teste é chamado como Verificação de Valores de Domínio.

- Condição de teste na qual é efetuada uma conversão dos valores dos domínios do sistema origem para o sistema destino por meio da utilização de uma tabela de conversão contendo os valores de domínio válidos para o sistema origem e sua correspondência para o sistema destino. Esse tipo de teste é chamado como Conversão de Valores de Domínio.

Re-estruturação das Chaves dos Registros de Entrada

A re-estruturação das chaves dos registros de entrada pode ser necessária para desvincular o registro de suas chaves originais, para isso, pode ser necessária a criação de uma chave substituta, o que pode ser obtido pela utilização de elementos de tempo (*Timestamp*), seqüenciadores de registro ou outras informações.

Uma outra utilização para chaves substitutas é a simplificação do acesso às tabelas de um sistema. Em sistemas com chaves de acesso compostas por vários argumentos de pesquisa, a utilização de chaves substitutas pode simplificar o acesso a essas tabelas.

No Modelo Dissertativo, a utilização da re-estruturação das chaves é especificada na forma de texto descritivo, no qual todos os passos para a geração da chave substituta são detalhados.

O exemplo aplicado no quadro “Regras de Transformação - Passo 10” da tabela 3.1 (Pág 50) do estudo de caso utiliza a re-estruturação das chaves para geração de um número seqüencial identificador de um cliente. Para cada novo registro de cliente é gerado um número seqüencial crescente que o identifica dentro do sistema.

Re-formatação de Dados

A re-formatação leva em consideração a necessidade de compatibilizar formatos de dados entre sistemas. Um exemplo de re-formatação de dados pode ser encontrado quando é necessário alterar o formato de uma data do

sistema origem de DD.MM.YYYY para YYYY.MM.DD de forma a compatibilizá-la com o formato do sistema destino. Observa-se, a partir do exemplo, que existe a possibilidade de criar funções específicas para alguns tipos de re-formatação de dados, porém, existem outras aplicações para a re-formatação, além de aplicações para datas e, estabelecer uma lista de aplicações é impraticável pois a utilização da re-formatação de dados depende da utilização do dado no contexto do sistema destino.

No Modelo Dissertativo, pelo fato de não haver processo único e genérico para efetuar as re-formatações, o analista descreve em detalhes os passos necessários para a re-formatação dos dados e o programador implementa essas especificações no processo de ETL.

No quadro “Regras de Transformação - Passo 1” da tabela 3.1 (Pág. 46) do estudo de caso é apresentado como a re-formatação é utilizada no Modelo Dissertativo. Neste exemplo ocorre uma re-formatação da Data de Nascimento. Essa é uma re-formatação simples, porém, demonstra como a funcionalidade é implementada no modelo.

Especificação de Valores Fixos

Nos casos em que não existe correlação entre atributos dos sistemas origem e destino deve ser possível a especificação de valores fixos para carga das entidades destino de dados.

No quadro “Regras de Transformação - Passo 1” da tabela 3.1 (Página 46) do estudo de caso observa-se a utilização de uma especificação de valores fixos quando informamos que a Data do Sistema deve ser movida para o campo Referência.

Identificação de Registros Alterados

Buscando eficiência de processamento, pode-se aplicar algoritmos de identificação de alterações em registros. Isso possibilita identificar dados

novos a serem carregados e dessa forma efetuar o processamento somente desses dados.

No ambiente de estudo existe a preocupação de efetuar o processamento somente dos dados que sofreram alteração, de forma a melhorar a eficiência do processamento, porém, a identificação das alterações nos dados é efetuada por programas construídos especialmente para cada uma das situações.

No Modelo Dissertativo a identificação de alterações é especificada de forma detalhada; cada procedimento necessário para identificação das alterações nos registros é descrito em detalhes para permitir a sua implementação por parte dos programadores.

No quadro “Regras de Transformação - Passo 9” da tabela 3.1 (Pág. 49) do estudo de caso, observa-se a utilização de uma identificação de registros alterados onde, por uma comparação entre o Arquivo de Clientes Anterior (E3) e o Arquivo de Clientes Atualizações (T7), descritos no quadro de “Arquivos do Processo” da tabela 3.1 (Pág. 45), é feita a atualização dos dados do Arquivo de Cliente Anterior com os dados do Arquivo de Clientes Atualizações.

3.4 Análise dos Resultados Obtidos

Observa-se que a utilização de uma especificação de processo de ETL no Modelo Dissertativo apresenta uma série de pontos fortes e pontos fracos entre eles:

Pontos Fortes

- Permite o estabelecimento do nível de detalhes na especificação do processo pois a utilização de texto explicativo permite ao analista

descrever as necessidades do processo de ETL com maior ou menor detalhe dependendo da necessidade.

- Modelo já conhecido pelo núcleo de desenvolvimento de software o que permite a aplicação do modelo sem a necessidade de treinamento prévio.
- O documento resultante da especificação de processo de ETL no Modelo Dissertativo agrega informações sobre a própria documentação garantindo um certo controle das alterações.
- Utilização de formato tabular para preenchimento da especificação permite que o analista não tenha que se preocupar com as dimensões da especificação já que a disposição do texto sobre o papel passa a ser automaticamente gerenciada pelo editor de textos.

Pontos Fracos

- Visão de conjunto prejudicada pois, a descrição de um processo de ETL não permite uma visão imediata do cenário de ETL desejado e, com isso, ocorre um consumo maior de tempo no entendimento do processo por parte dos programadores.
- Não existe padrão para a especificação já que a utilização de texto descritivo faz com que, para a descrição de um mesmo processo de ETL, analistas de sistemas diferentes criem documentos totalmente diferentes. Isso é uma desvantagem porque os programadores passam a consumir mais tempo no entendimento dos processos de ETL porque não conseguem identificar nas descrições, padrões que facilitariam a compreensão e a construção do processo especificado.
- A clareza das especificações depende única e exclusivamente da capacidade do analista em se expressar na linguagem escrita. Em geral, a capacidade de expressão de um indivíduo não é algo mensurável e depende muitas vezes de fatores sócio-culturais e em função disso, nem sempre a especificação criada torna-se clara.
- Qualquer alteração no processo de ETL exige que o documento de especificação do processo de ETL seja revisado para que a

implementação da alteração possa ser feita sem perda da coerência na documentação. Isso consome muito tempo e esforço por parte do analista de sistemas.

- A utilização de texto não estruturado não permite a criação de ferramentas para auxilio ao desenvolvimento e manutenção de especificações nesse modelo.

3.5 Conclusões

Neste capítulo é abordado um estudo de caso utilizando o Modelo Dissertativo, onde ele é detalhadamente descrito e a documentação do processo de ETL utilizada no estudo de caso é apresentada. Na apresentação da documentação são feitas considerações sobre quais funcionalidades ETL devem ser consideradas no estudo de caso para validá-lo na tarefa de avaliar a eficácia e eficiência na utilização do modelo. A documentação de especificação de processo de ETL selecionada é então apresentada para ilustrar a forma como o Modelo Dissertativo é utilizado. Essa documentação é então comentada para facilitar a identificação das funcionalidades selecionadas na especificação produzida e, ao final, considerações sobre a aplicação encerram o estudo de caso.

Sendo que o Modelo Dissertativo de especificação é utilizando diariamente no ambiente de estudo, pode-se aceitá-lo como sendo um modelo de especificação eficaz, porém, sua eficiência deverá ser avaliada a partir de um outro modelo de especificação. Essa avaliação será efetuada no próximo capítulo onde o estudo de caso utilizará a especificação no Modelo Dissertativo para efetuar uma “derivação” para o Modelo Conceitual [4].

4 Modelo Conceitual

4.1 Introdução

Neste capítulo a especificação no modelo dissertativo será utilizada para auxiliar na "derivação" para o modelo gráfico. Nessa “derivação” os passos da metodologia de utilização do Modelo Conceitual [4] são seguidos e ao final efetua-se a comparação entre os modelos Dissertativo e Conceitual [4] utilizados no estudo de caso.

4.2 Derivação para o Modelo Conceitual

A construção do documento de especificação utilizando no Modelo Conceitual [4] que será utilizado neste capítulo, leva em consideração o documento de especificação resultante do capítulo anterior feita no Modelo Dissertativo. Todas as funcionalidades utilizadas no estudo de caso do capítulo anterior serão transcritas para o novo modelo de especificação, considerando a sua notação e metodologia de aplicação.

4.2.1 Passo 1 - Identificação das Fontes Apropriadas

A figura 4.1 apresenta a aplicação do primeiro passo da metodologia, que é a Identificação das Fontes Apropriadas. Nesta figura apresenta-se o fluxo total do processo. Neste fluxo identifica-se os mesmos passos descritos no estudo de caso para o Modelo Dissertativo (Capítulo 3). Nele observa-se os passos:

- Geração do arquivo T1 a partir da seleção dos registros de pessoas físicas do arquivo E1 (Passo um);
- Divisão das informações de Endereço e CEP do arquivo T1 para o arquivo T2 (indicados em nota explicativa) (Passo dois);
- Padronização do Endereço do arquivo T2 para o arquivo T4(Passo três);
- Divisão das informações de Endereço e CEP do arquivo E2 para o arquivo T3 (Passo quatro);
- Padronização do Endereço do arquivo T3 para T5 (Passo cinco);
- A junção dos arquivos T4 e T5 para geração do arquivo T6 enriquecido (Passo seis);
- Geração dos arquivos T7 e T8 com as atualizações e inclusões de clientes com base na comparação entre o arquivo T6 com o Cadastro de Clientes do arquivo E3 (Passos sete e oito);
- Atualização dos dados de E3 a partir do arquivo T7, gerando o arquivo T9 (Passo nove);
- Geração de uma chave substituta no arquivo T8, gerando o arquivo T10 (Passo dez) ;
- Junção dos dois cadastros T9 e T10 gerando a saída S1 (Cadastro de Clientes Novo) (Passo onze).

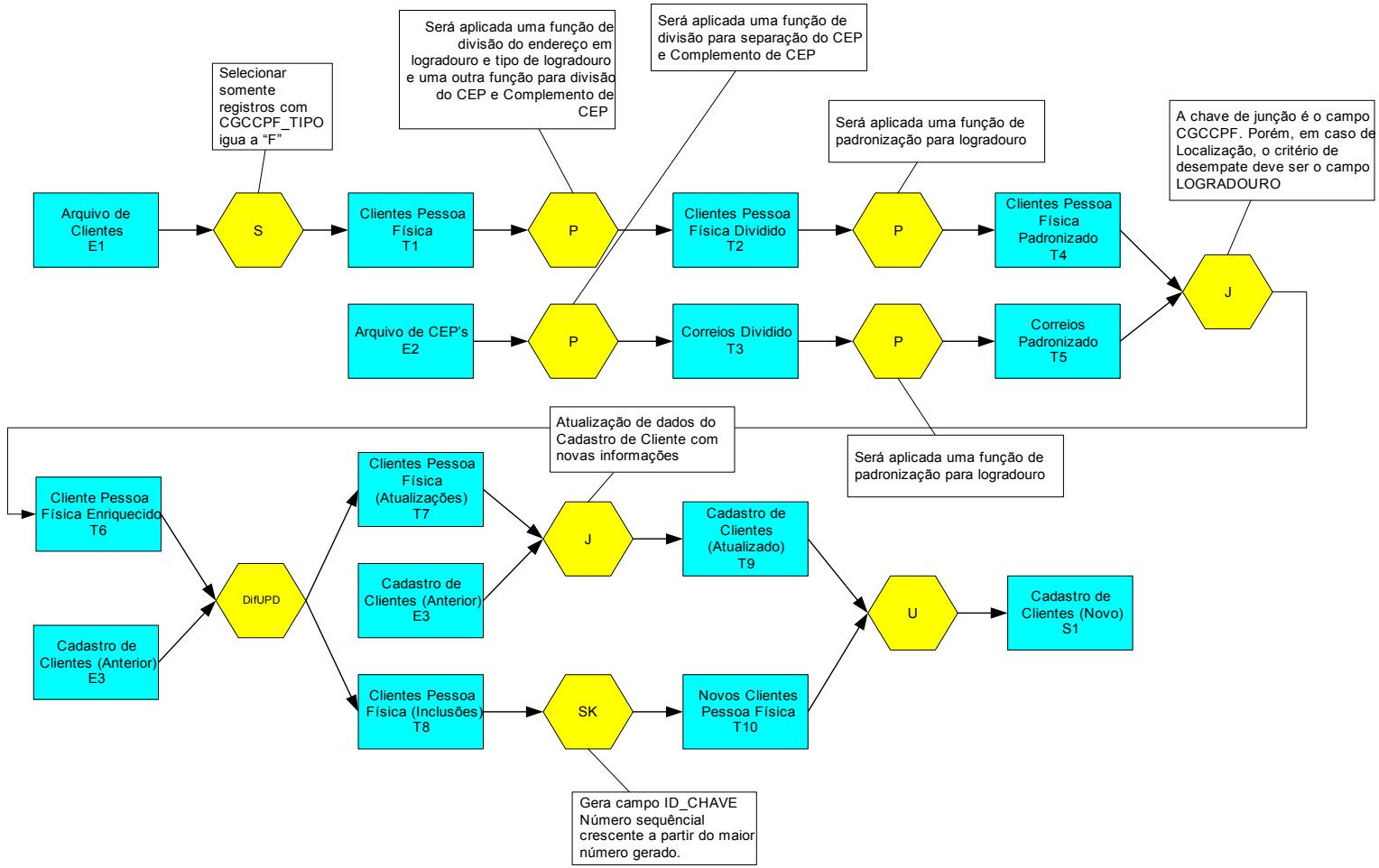


Figura 4. 1 – Cenário de Execução (Passo 1 da Metodologia Aplicada ao Estudo de Caso)

4.2.2 Passo 2 - Identificação das Fontes Candidatas

Para a amostra selecionada para o estudo de caso não foi identificada nenhuma situação de fontes candidatas; dessa forma, o diagrama da figura 4.1 da página 65 não se altera.

4.2.3 Passo 3 - Mapeamento dos Atributos entre Fontes e Destinos

O passo 1 da metodologia, já abordado anteriormente, apresenta um cenário para processo de ETL que envolve diversas entidades. A apresentação visual de todas essas entidades em seu nível mais detalhado e em um único gráfico, tornaria o cenário irreconhecível, por isso a especificação é "quebrada" em partes permitindo uma melhor compreensão da inter-relação entre essas entidades.

As figuras 4.2, 4.3, 4.4, 4.5, 4.6 e 4.7, apresentam o detalhamento do fluxo do processo necessário para geração do Cadastro de Clientes e são detalhadas a seguir.

Na figura 4.2 observa-se:

- A passagem das informações de E1 para T1 onde são filtrados os registros com CGCCPF_TIPO igual a "F" (Pessoa Física).
- A propagação dos conteúdos da entidade T1 para as entidades T2 e T4, carregando a maior parte das informações de E1 até T4 sem maiores modificações.

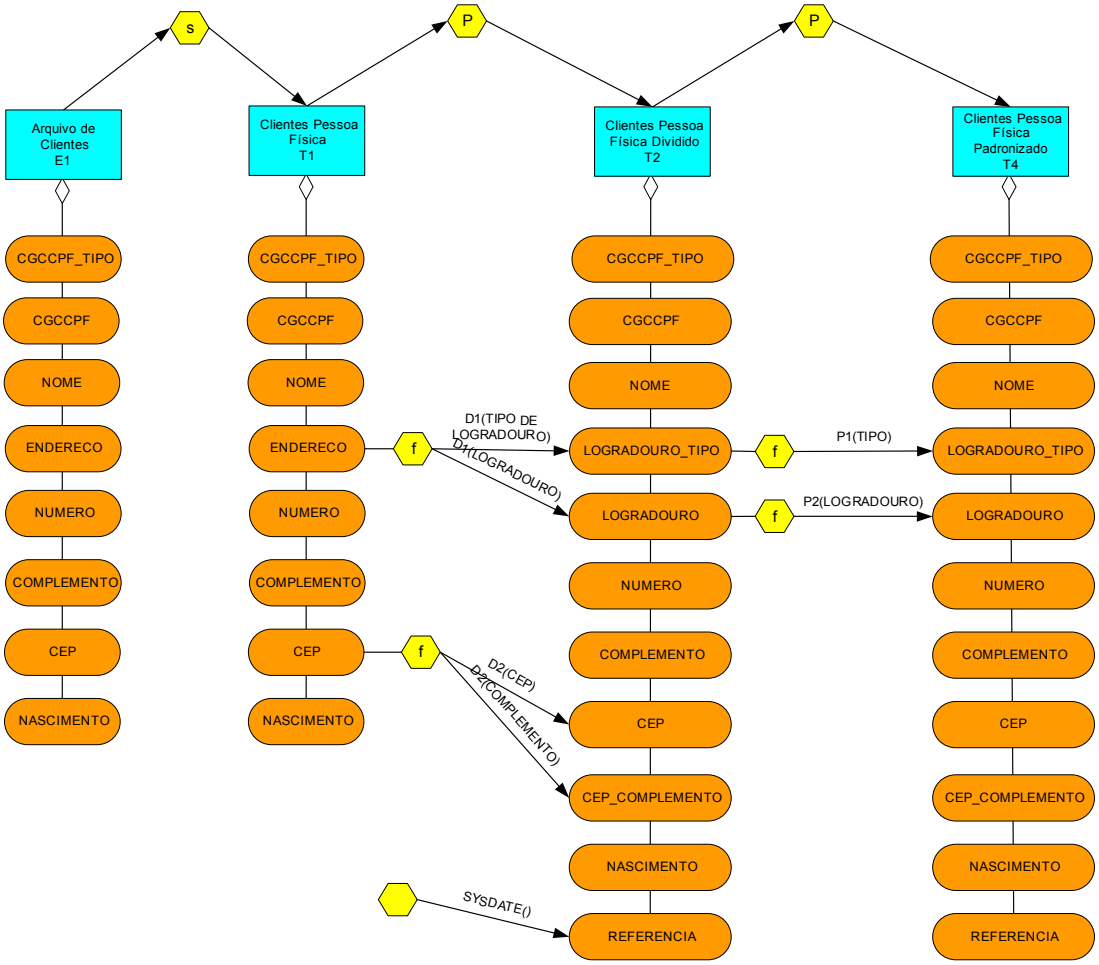


Figura 4. 2 – Transferência de Informações de E1 para T4

- A aplicação das funções D1 e D2 na propagação de valores de T1 para T2, são utilizadas para separar o Endereço em Tipo de Logradouro e Logradouro (D1) e para separar o CEP em CEP e Complemento de CEP (D2). Observa-se que as regras de separação dessas duas funções são completamente diferentes em função das regras de formação dos conteúdos dos campos. A regra D1, para um Endereço “Av. Paulista” retorna como Tipo de Logradouro “Av.” e Logradouro “Paulista”. A regra D2, para um CEP 05027020 retorna o CEP 05027 e o Complemento de CEP 020.

- Utilização das funções P1 e P2 para efetuar a padronização dos campos Tipo de Logradouro e Logradouro. As funções de padronização são utilizadas para tornar uniforme o conteúdo desses campos de forma a

reduzir as diferenças de grafia e abreviações. Ex: Um campo Tipo de Logradouro pode conter “Rua” ou “R” ou “R.”, etc. a padronização desse campo poderia alterar todos esses conteúdos para “Rua”, o que tornaria seu conteúdo mais uniforme.

A figura 4.3 apresenta a propagação das informações de E2 para T5. Nessa passagem, assim com apresentado na figura 4.2, observa-se a utilização da função D2 para a divisão do CEP em CEP e Complemento de CEP e, das funções P1 e P2 para o tratamento de padronização das informações Tipo de Logradouro e Logradouro.

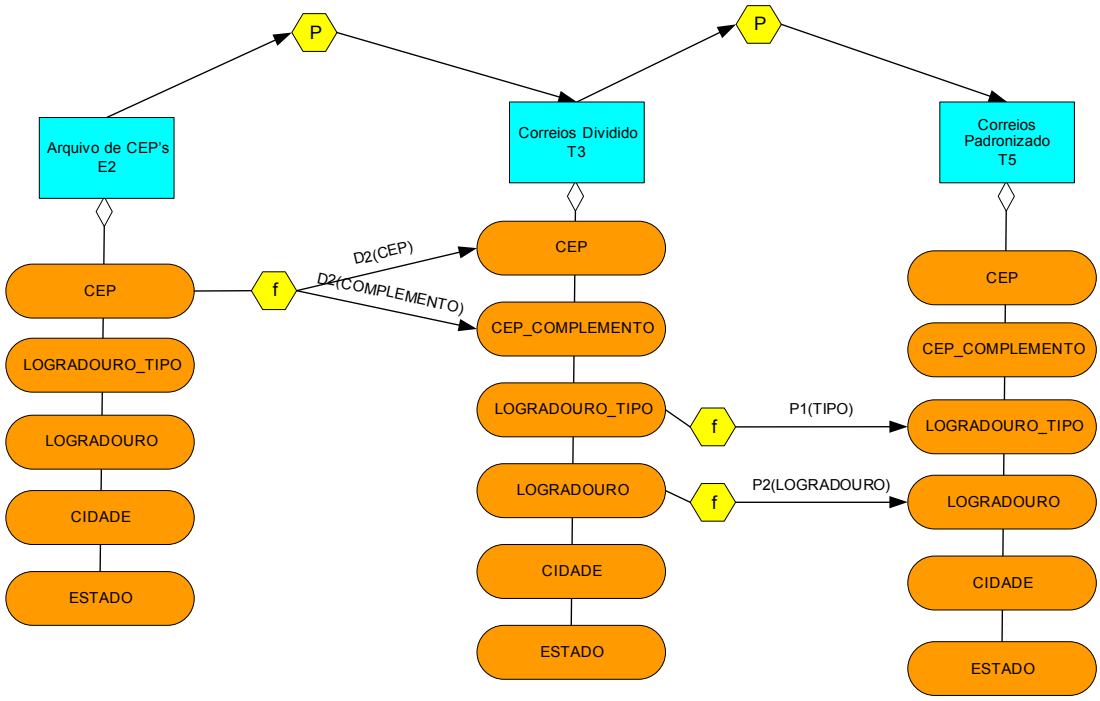


Figura 4. 3 –Transferência de Informações de E2 para T5

A figura 4.4 apresenta um processo de Junção entre os arquivos T4 e T5 para a carga do arquivo T6. Observa-se que, para carga do arquivo T6, deve ocorrer a localização dos registros de T5 com o mesmo CEP de T4. Isso pode retornar vários registros de T5 para um único registro de T4, visto que podem ocorrer vários complementos de CEP para um único CEP. Desses registros retornados deve ser selecionado aquele cujo logradouro mais se

aproximar do logradouro de T4 e desse registro de T5 deverá ser obtido o Complemento de CEP para a carga de T6.

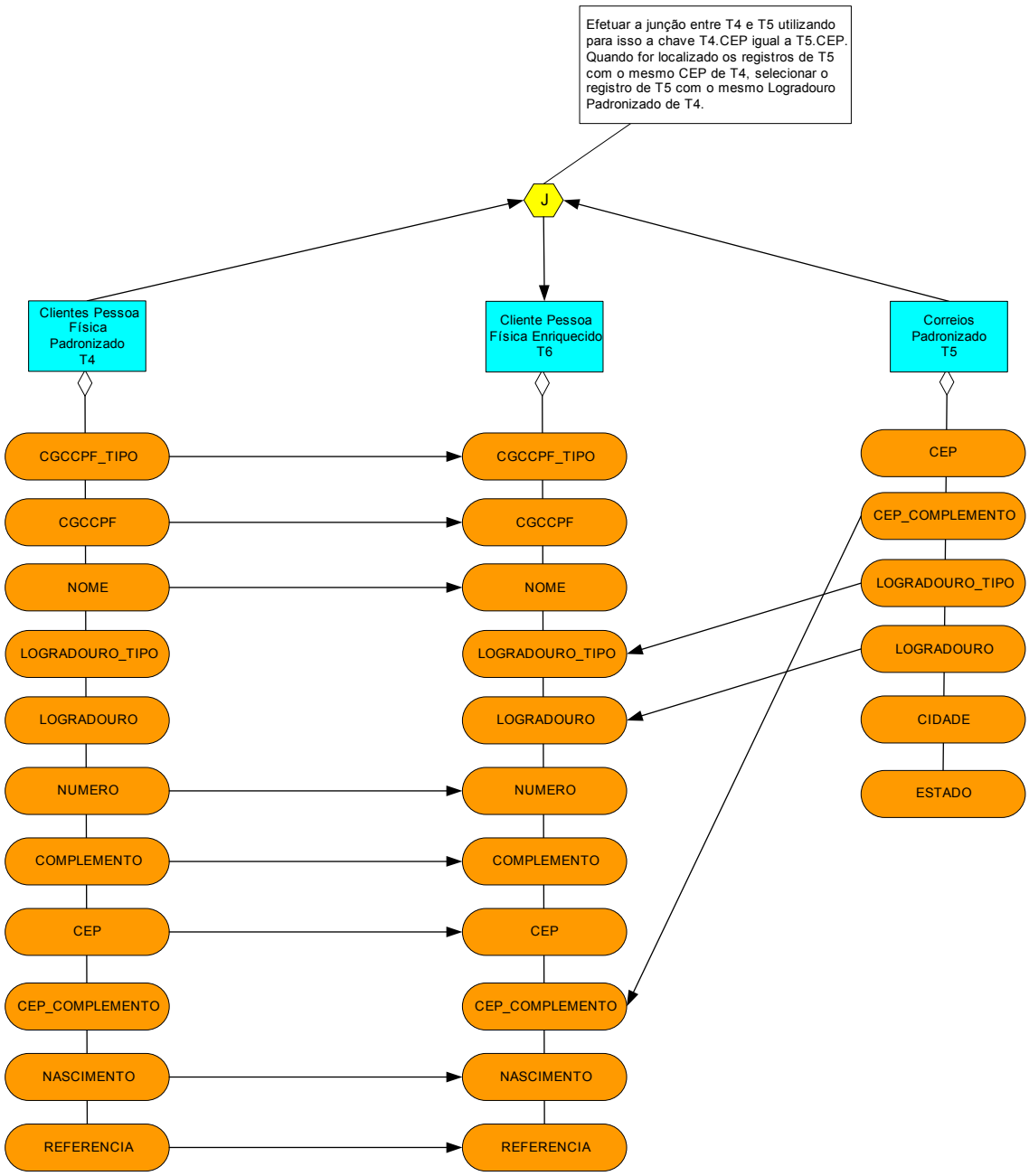


Figura 4. 4 - Junção de T4 com T5 para geração de T6

Conforme conceituado, pode-se utilizar diversos algorítimos de comparação para se chegar a conclusão de que o logradouro padronizado de T4 seja equivalente ao logradouro padronizado de T5 por meio de processos de identificação. Atenta-se para esse fato pois o processo de padronização, como dito anteriormente, procura uniformizar o conteúdo dos campos. Nem sempre isso é possível e nem sempre o resultado da padronização é totalmente uniforme ao ponto de, numa comparação simples e direta, identificar que o conteúdo de um logradouro de T4 seja ou não igual ao de outro logradouro de T5.

Em conseqüência dos problemas de padronização, deve-se localizar, com certo grau de certeza, que um determinado logradouro de T4 seja igual a um determinado logradouro de T5. Para isso podem ser utilizados alguns algorítimos de identificação.

A representação da funcionalidade identificação no exemplo da figura 4.4., pode ser implementada pela utilização do símbolo de Junção (J), porém não deve ser entendido como sendo uma simples junção de arquivos, mas sim como sendo a representação da utilização de uma poderosa ferramenta de identificação.

A figura 4.5 apresenta outra parte do processo, onde é utilizada uma funcionalidade de detecção de atualizações. Com base no arquivo E3, é efetuada uma comparação com o arquivo T6, utilizando como chave de pesquisa o campo CGCCPF (Número do CPF do Cliente). Por meio de comparações entre as chaves, identifica-se os registros a serem incluídos no cadastro (Clientes Novos) e os registros que sofreram atualizações cadastrais (Clientes Atualizados), gerando respectivamente os arquivos T8 e T7.

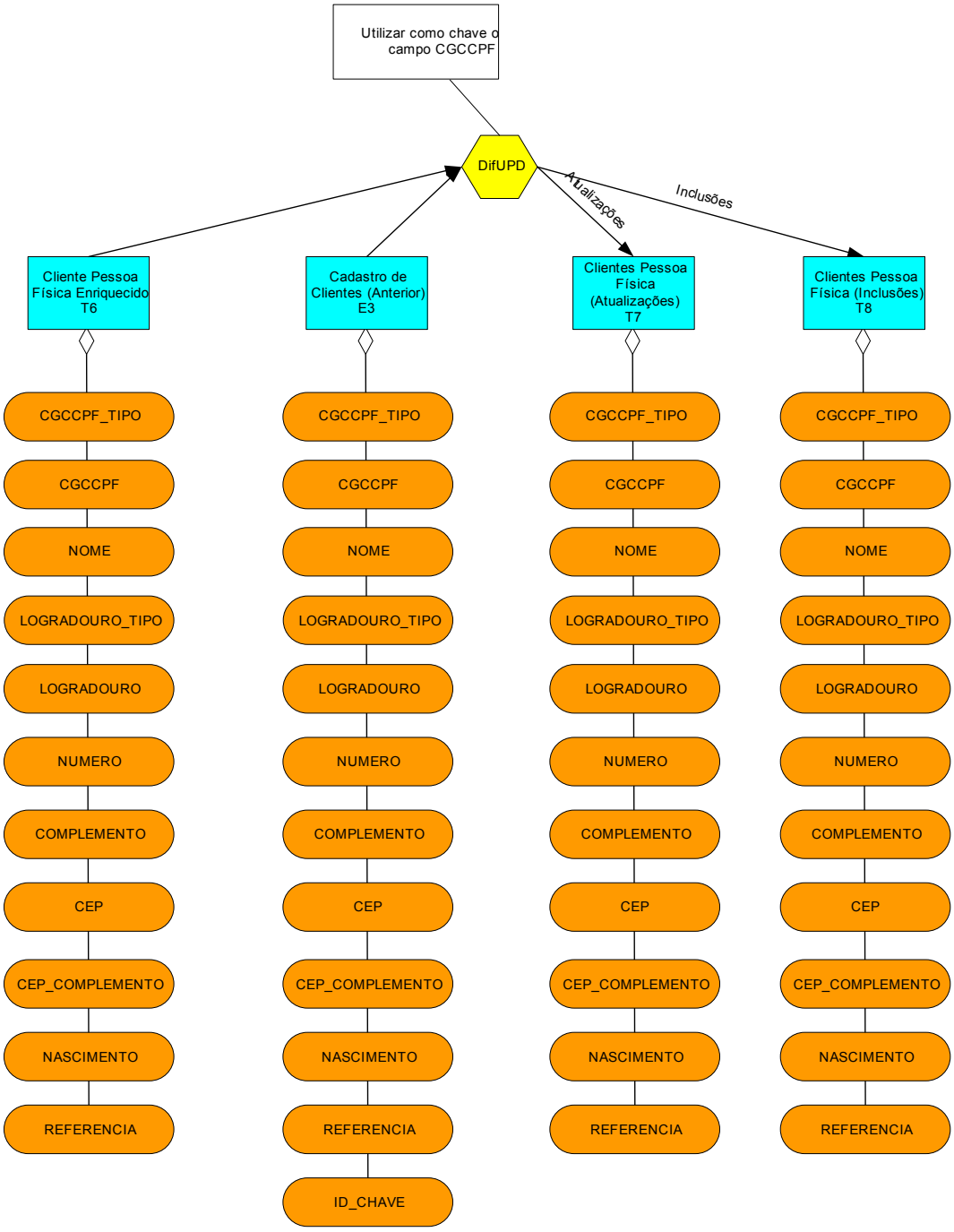


Figura 4. 5 - Identificação de Atualizações

A figura 4.6. apresenta a atualização do Cadastro de Clientes Anterior (E3) com os dados de Clientes Pessoa Física (Atualizações). É feita uma junção entre os arquivo E3 e T7 para efetuar a carga do arquivo T9. Observa-se que os dados utilizados para a carga de T9 são provenientes do arquivo T7, com exceção do campo ID_CHAVE que é uma chave identificadora gerada

num processo mais adiante e que não se encontra presente no arquivo T7. Nos caso em que um registro de E3 não seja localizado em T7, efetua-se a cópia do registro de E3 para T9 sem modificações, conforme indicado em nota explicativa.

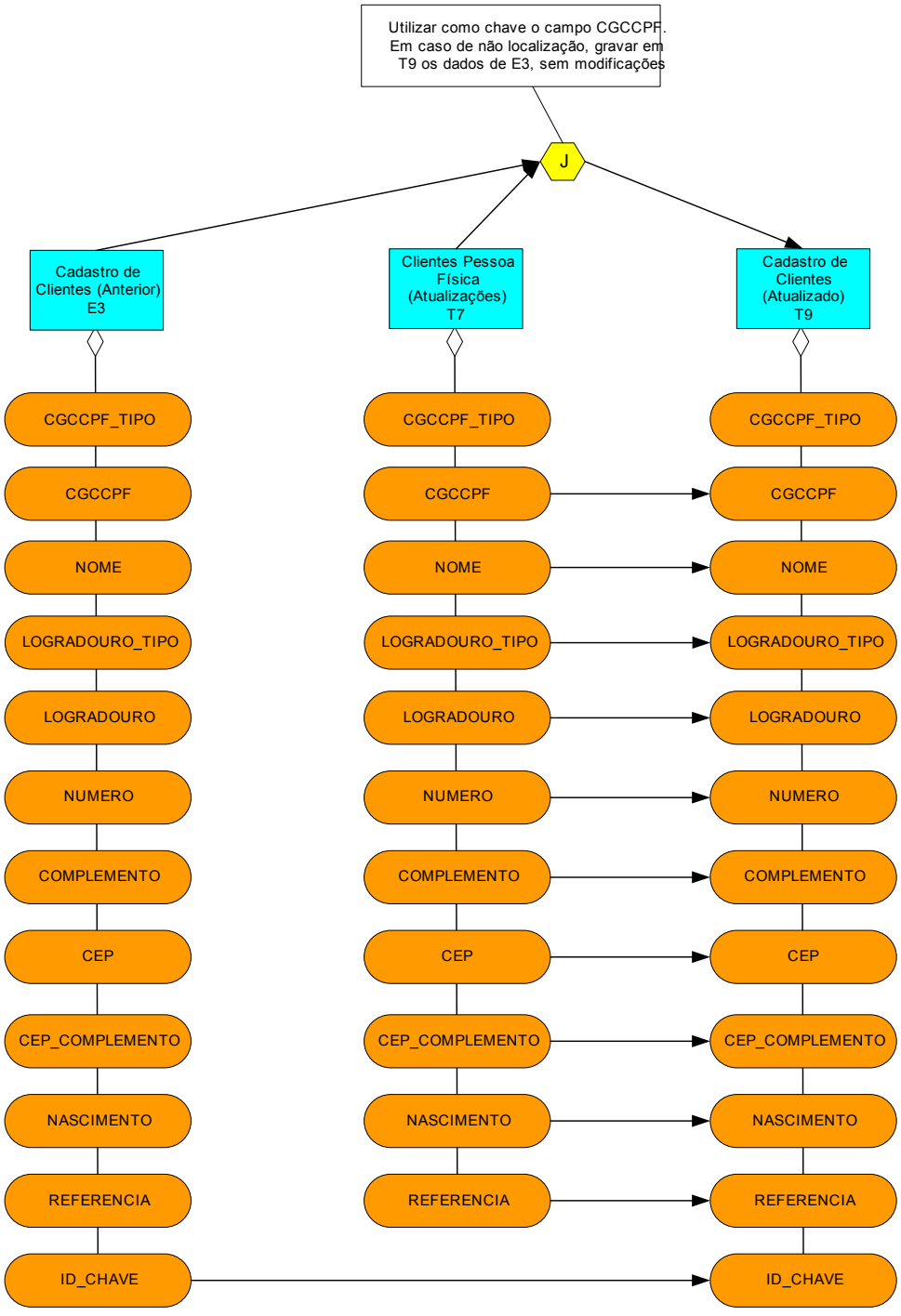


Figura 4. 6 - Atualização do Cadastro de Clientes

A figura 4.7 apresenta a geração do arquivo T10. Neste passo é feita uma propagação dos campos do arquivo T8 para o arquivo T10, com exceção do campo ID_CHAVE. Esse campo é uma chave substituta que deve ser gerada como sendo um número seqüencial crescente e representa a identificação única do Cliente no Cadastro de Clientes.

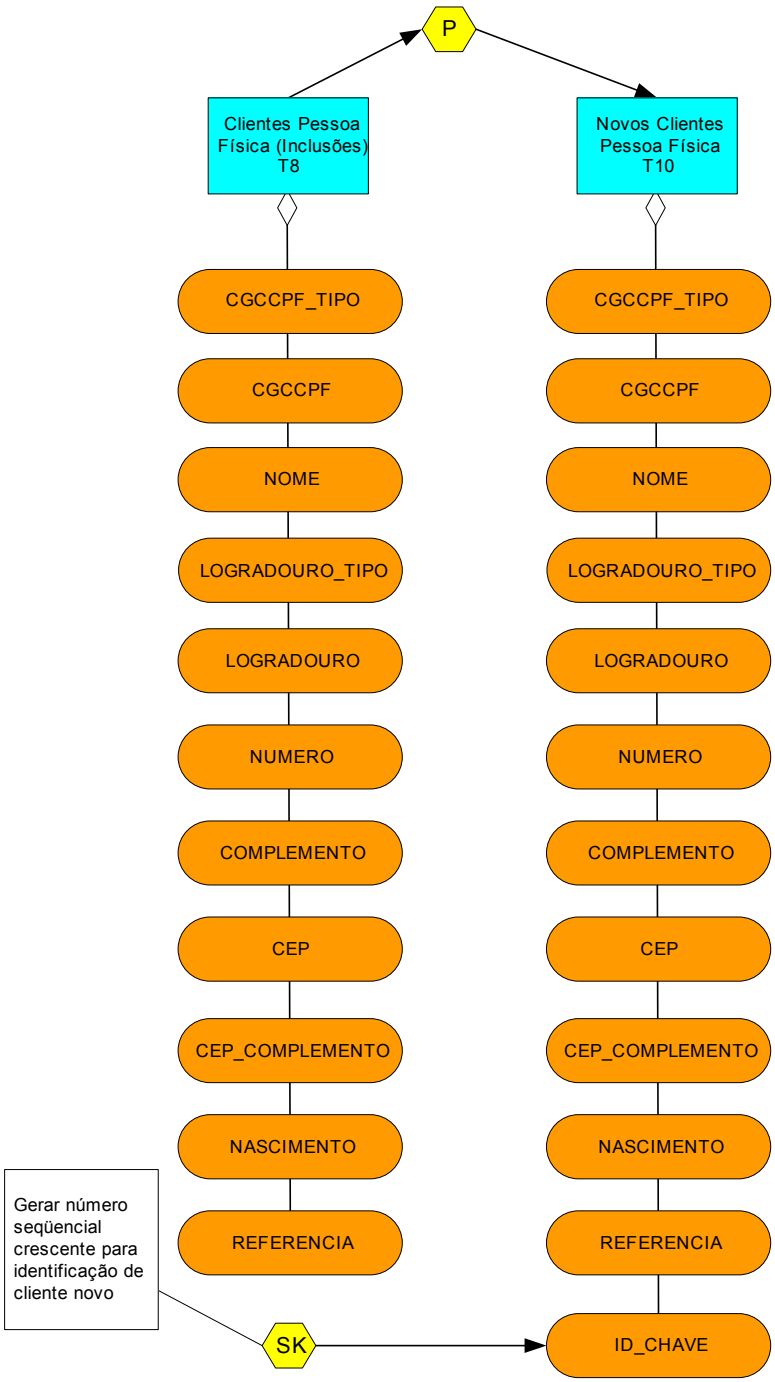


Figura 4. 7 - Geração do arquivo T10 com geração do campo ID_CHAVE

A figura 4.8 apresenta a união dos arquivos T9 e T10 para efetuar a carga do arquivo S1, que passa a conter o Cadastro de Clientes Atualizado.

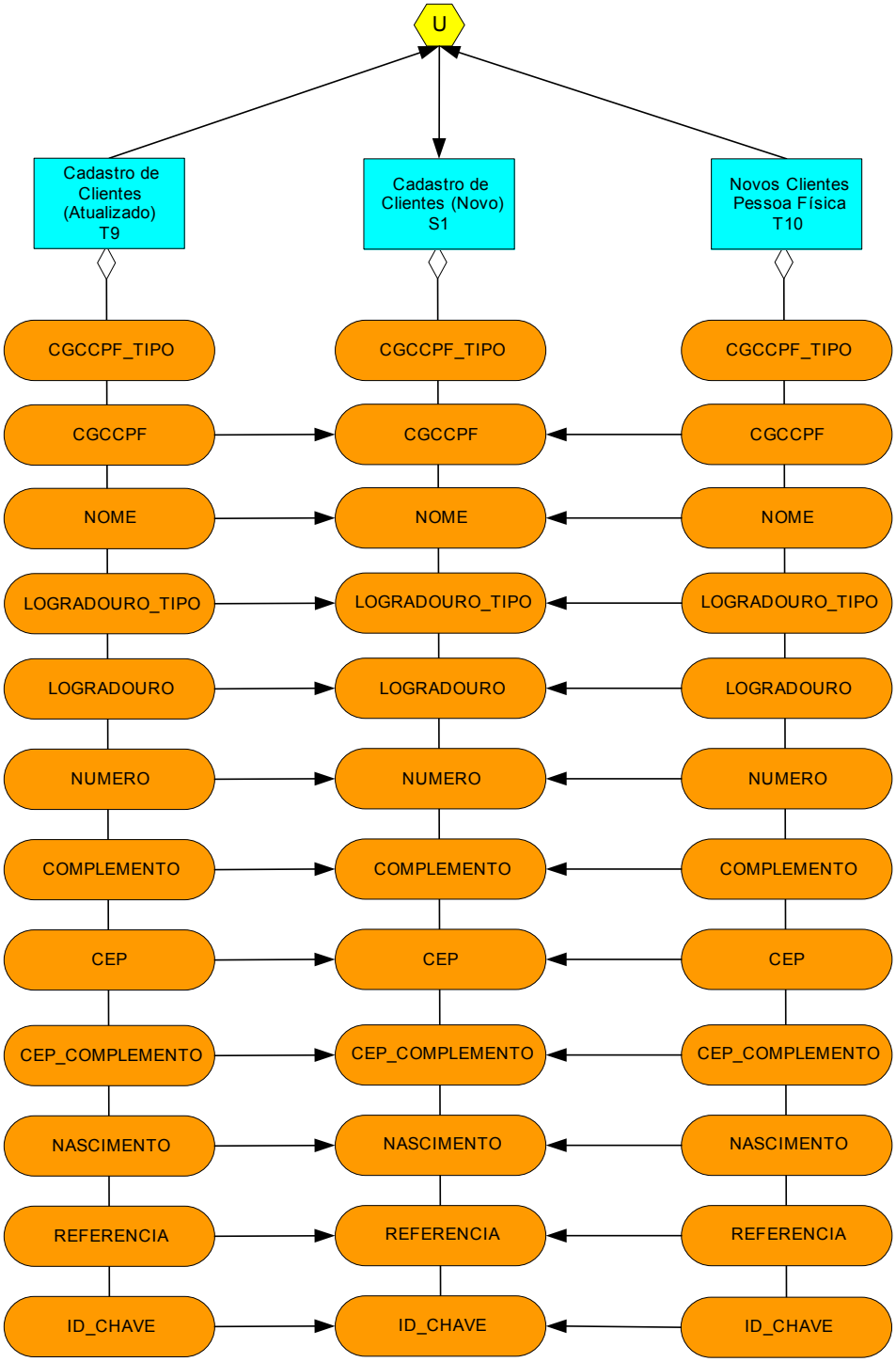


Figura 4. 8 - Geração final do arquivo S1

4.2.4 Passo 4 - Anotação das Restrições de Tempo de Execução

Para a amostra selecionada para o estudo de caso não foi identificada nenhuma situação em que fosse necessário o registro de restrições de tempo para processamento, portanto, o diagrama apresentado na figura 4.1 não se altera.

4.2.5 Passo 5 – Montagem do Diagrama final

Sem a necessidade de representação das restrições de tempo ou outras anotações, o diagrama final utilizando a metodologia proposta pelos autores é como apresentado nas figuras 4.2, 4.3, 4.4, 4.5, 4.6, 4.7 e 4.8.

Nesses diagramas podemos observar a utilização das mesmas funcionalidades utilizadas no Modelo Dissertativo agora aplicadas ao Modelo Conceitual [4].

4.3 Análise dos Resultados Obtidos

Observa-se que a aplicação de uma especificação de processo de ETL no Modelo Conceitual [4] apresenta uma série de pontos fortes e pontos fracos entre eles:

Pontos Fortes

- O Modelo Conceitual [4], no Passo 1 da metodologia, como observado na figura 4.1, possibilita uma visão de contexto clara sobre todas as operações envolvidas no processo de ETL a ser construído.
- Os pontos principais de uma definição (pontos mais complexos) ficam evidentes. A atividades simples são tratadas de modo geral por meio

de funções como a Propagação que movimenta todos os campos de mesmo nome da origem para o destino sem especificá-los individualmente.

- Cada item gráfico que o compõe o Modelo Conceitual [4], concentra uma série de conceitos que faz com que uma especificação de processo de ETL que utilize essas notações, seja claramente representada por meio de uma quantidade pequena de itens gráficos relacionados.
- Os itens gráficos do Modelo Conceitual [4] podem ser utilizados em várias especificações de processos de ETL porque podem ser re-combinados conforme a necessidade.
- As especificações de processos de ETL, construídas com base no Modelo Conceitual [4], são montadas com base em regras de utilização que podem ser aplicadas na construção de ferramentas gráficas para confecção de diagramas utilizando o modelo.
- Sendo composto por itens que representam funcionalidades fechadas, uma especificação de processo de ETL no Modelo Conceitual [4] pode ser facilmente modificada pela inclusão ou retirada de uma ou mais funcionalidades, pois as demais funcionalidades da especificação não seriam afetadas.

Ponto Fraco

- Consome muito tempo na especificação, pois a manipulação de itens gráficos na diagramação de um processo de ETL torna o processo de construção da documentação mais lento e cansativo.

4.4 Modelo Conceitual X Modelo Dissertativo

A utilização do Modelo Conceitual [4], por meio do estudo de caso apresentado neste capítulo, em relação ao Modelo Dissertativo, fornece meios para avaliação dos dois modelos de especificação. Observando-se os

critérios de comparação estipulados no item 1.2 deste trabalho, pode-se construir o quadro comparativo da figura 4.9 a seguir, que apresenta o resultado dessa avaliação no estudo de caso com os modelos envolvidos.

Quadro Comparativo Modelo Conceitual [4] X Modelo Dissertativo		
Critério	**Modelo Conceitual [4]**	**Modelo Dissertativo**
Representação Compacta	Utiliza notação gráfica e visual	Utiliza texto não estruturado
Usabilidade	Necessita esforço de diagramação	Utiliza formato tabular de preenchimento
Visão Integrada	Apresenta visão de contexto em formato gráfico	Apresenta visão de contexto em texto não estruturado
Automatização	Notação pode ser utilizada em ferramentas gráficas	Texto não estruturado não adaptável a ferramentas gráficas.
Reutilização	É intrínseco ao modelo, pois é gráfico e visual.	Não é intrínseco ao modelo, pois utiliza texto não estruturado.
Padronização	Ocorre de forma natural	A desejo do analista
Legibilidade	Fácil percepção dos objetivos e funcionalidades	Necessita leitura e interpretação de texto
Adaptabilidade	Modificações podem ser facilmente implementadas	Modificações consomem tempo e esforço.

Figura 4. 9 - Quadro Comparativo Modelo Conceitual [4] X Modelo Dissertativo

O quadro apresenta pontos fortes e fracos dos dois modelos segundo os critérios de avaliação considerados no estudo de caso. A figura 4.10, por sua vez, apresenta um quadro que, a partir da análise dos pontos fortes e fracos dois modelos de especificação de processos de ETL, feitos com base nos resultados apresentados no quadro anterior, sintetiza a avaliação comparativa desses dois modelos.

Resultado da Avaliação Comparativa do Modelo Conceitual [4]	
Critério	**Resultado**
Representação Compacta	Dado que o Modelo Conceitual [4] é composto por uma notação que sintetiza uma série de conceitos em um único item gráfico, ele pode ser considerado melhor que o Modelo Dissertativo neste critério de avaliação.
Usabilidade	Dado que o Modelo Conceitual [4] consome maior esforço na diagramação do que o Modelo Dissertativo, ele pode ser considerado pior do que o Modelo Dissertativo neste critério de avaliação.
Visão Integrada	Dado que o Modelo Conceitual [4] possui uma visão de contexto em formato gráfico, ele pode ser considerado melhor do que o Modelo Dissertativo neste critério de avaliação.
Automatização	Dado que as notações do Modelo Conceitual [4], por ser gráfico e conceitual, podem ser implementadas em uma ferramenta gráfica de especificação, ele pode ser considerado melhor que o Modelo Dissertativo neste critério de avaliação.
Reutilização	Dado que a reutilização é intrínseca ao Modelo Conceitual [4] já que cada item gráfico é um conjunto de conceitos fechados que podem ser reutilizados, ele pode ser considerado melhor que o Modelo Dissertativo neste critério de avaliação.
Padronização	Dado que os itens gráficos do Modelo Conceitual [4] são conceitos fechados que são utilizados como padrões para o desenvolvimento de especificações de processos de ETL, O Modelo Conceitual [4] pode ser considerado melhor que o Modelo Dissertativo neste critério de avaliação.
Legibilidade	Dado que o Modelo Conceitual [4] é um modelo gráfico e, portanto, de fácil percepção, ele pode ser considerado melhor que o Modelo Dissertativo neste critério de avaliação.
Adaptabilidade	Dado que no Modelo Conceitual [4] uma inserção ou retirada de uma funcionalidade é uma tarefa mais fácil do que no Modelo Dissertativo, ele pode ser considerado melhor do que o Modelo Dissertativo neste critério de avaliação.

Figura 4. 10 - Resultado da Avaliação do Modelo Conceitual [4]

A avaliação do quadro da figura 4.10 indica que o Modelo Conceitual [4] é melhor do que o Modelo Dissertativo quando observados os critérios de

Representação Compacta, Visão integrada, Automatização, Reutilização, Padronização, Legibilidade e Adaptabilidade e é pior do que o Modelo Dissertativo quando observado o critério de Usabilidade.

Além da avaliação o estudo de caso utilizando os dois modelos de especificação permite observar que na utilização do Modelo Conceitual [4], assim como no Modelo Dissertativo, existe uma certa dificuldade de representação de todas as transformações quando o cenário de ETL é composto por mais de uma entidade em etapas de processamento diferentes.

No Modelo Conceitual [4] existe a mesma tendência do Modelo Dissertativo de representar as transformações por etapas, de forma a restringir a duas ou três entidades diretamente relacionadas para que o fluxo do processo se torne mais claro para o programador.

Uma outra observação que também poderia ser feita sobre diferenças entre os dois modelos seria a possibilidade de extensão das notações. No Modelo Dissertativo, por não haver regras rígidas para a descrição de processos ETL, analistas de sistemas diferentes criaram especificações diferentes para um mesmo processo de ETL. Isso já ocorre em menor grau no Modelo Conceitual [4] porque existe um conjunto fechado de conceitos quando utilizamos uma notação gráfica. Isso significa também que existem regras mais rígidas para a especificação de processos de ETL e, em conseqüência, quando existe a necessidade de representação de novas funcionalidades no Modelo Conceitual [4] é necessário que uma extensão da notação seja criada para abarcar essa nova funcionalidade.

A utilização do Modelo Conceitual [4] evidencia um aumento no tempo de especificação, em detrimento de uma melhoria na clareza da especificação. Isso significa transferir parte do ônus do entendimento de um processo de ETL do programador para o analista de sistemas, pois a utilização do Modelo Conceitual [4] por parte do analista, acaba por obrigá-lo a construir especificações mais claras para o programador.

4.5 Conclusão

Neste capítulo foi apresentado o estudo de caso utilizando o Modelo Conceitual [4] de especificação de processo de ETL. Foi apresentada a documentação resultante da “derivação” do Modelo Dissertativo para o Modelo Conceitual [4] onde todos os procedimentos utilizados foram descritos. Pontos fortes e fracos do modelo foram abordados e foi montado um quadro comparativo entre os dois modelos de forma a evidenciar as vantagens e desvantagens na utilização de um Modelo Conceitual [4] em substituição a um Modelo Dissertativo.

Conclui-se que a utilização de um Modelo Conceitual [4] em substituição a um Modelo Dissertativo traz benefícios, pois torna mais fácil a interpretação das necessidades de um processo de ETL reduzindo o tempo e o esforço no entendimento de uma especificação.

Observa-se, porém, que todas as particularidades do processo acabaram, de alguma forma, se tornando notas explicativas das especificações. No próximo capítulo são sugeridas melhorias à notação e a metodologia do Modelo Conceitual [4] de forma a melhor abrigar essas particularidades.

5 Melhorias ao Modelo Conceitual

5.1 Introdução

Neste capítulo são apresentadas diversas sugestões de melhoria a serem implementadas ao Modelo Conceitual [4], assim como os motivos pelos quais essas melhorias facilitariam a manipulação do modelo em situação real de utilização.

5.2 Melhorias na Notação e Metodologia Originais

Pelo estudo de caso utilizando os modelos Conceitual [4] e Dissertativo, observa-se que existe a necessidade de melhoria das notações e da metodologia para um melhor aproveitamento das melhores características do modelo.

Com base no estudo de caso do capítulo 4, apresenta-se as seguintes sugestões de melhoria na notação e na metodologia de aplicação do Modelo Conceitual [4].

5.2.1. Níveis de Abstração

O diagrama do Passo 1 da metodologia de aplicação do Modelo Conceitual [4] (Figura 4.1 da Pág. 65) , não permite uma visão clara das entradas e saídas do processo de ETL, devido a isso a figura 5.1 apresenta uma sugestão de melhoria pela adoção de um diagrama de mais alto nível, onde podem ser identificadas as entradas e saídas do processo. Neste diagrama,

as entras e saídas do processo encontram-se identificadas por cores diferentes o que facilita a sua visualização.

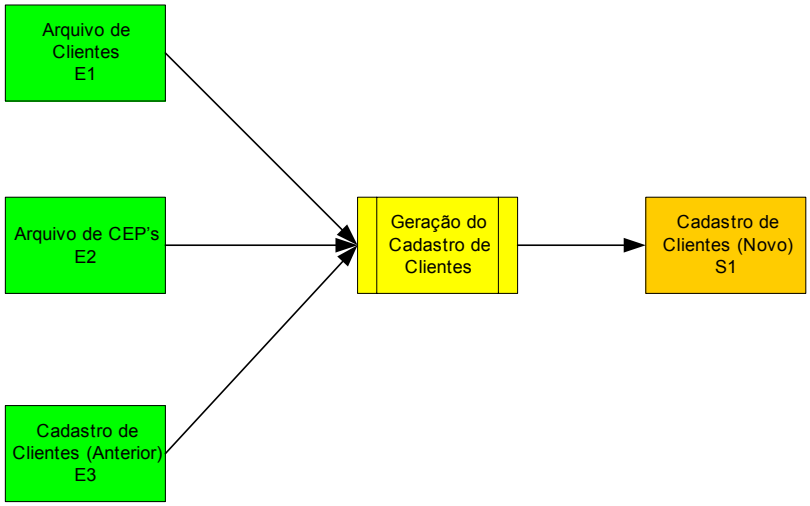


Figura 5. 1 - Sugestão para Diagrama de Nível Zero

A utilização de cores para diferenciação das entradas e saídas do processo pode ser utilizada no diagrama do passo 1 da metodologia (Fig. 4.1 da Pág. 65). Dessa forma uma característica positiva do Modelo Conceitual [4] que é uma visão mais clara do processo pode ser melhorada.

5.2.2. Controle de Configurações e Versões

Identifica-se que o Modelo Conceitual [4] possui uma visão centrada na especificação de um processo de ETL novo, porém, não possui uma estrutura adequada para permitir um controle das alterações da documentação, permitindo um acompanhamento da evolução do processo no decorrer do tempo.

No Modelo Dissertativo, esse problema é resolvido pela adoção de um formulário com uma estrutura que permite o registro das alterações que o documento de especificação recebe ao longo do tempo.

No item “Aditamentos” do formulário de especificação no Modelo Dissertativo (Tabela 3.1 da Pág. 50), são registradas as alterações que o documento de especificação recebe ao longo do tempo. Todavia, o registro de todas as alterações de especificação em um único documento dificulta o acompanhamento da evolução do processo no decorrer do tempo e dificulta a manutenção da especificação, pois, é necessário contextualizar a alteração a ser implementada na documentação já existente.

A adoção do controle de configuração e versão no Modelo Conceitual [4] permitirá a implementação de meios para acompanhar a evolução do processo no decorrer do tempo e pode ser feita a partir de procedimentos simples. Pode-se adotar uma estrutura de diretórios para isolar os componentes do modelo e utilizar o diagrama do Cenário de Execução (Passo 1 da Metodologia) como um índice para acessar os demais documentos relacionados.

Para exemplificar essa modificação é apresentada uma solução baseada na adoção do software Microsoft Visio utilizado para criar e manter os diagramas utilizados, porém, a solução proposta independe da ferramenta utilizada. Para confecção dos diagramas utilizados no Modelo Conceitual [4] podem ser utilizadas diversas ferramentas de diagramação, assim como editores de texto disponíveis no mercado e, as soluções propostas, podem ser implementadas na maioria dessas ferramentas.

Para a implementação dessa melhoria utilizando a ferramenta proposta, será necessário efetuar uma modificação no diagrama de Cenário de Execução. Observa-se que o seu funcionamento depende de quatro tipos de itens gráficos diferentes, a saber: - Conceito, Transformação, Nota Explicativa e Relacionamentos.

A figura 5.2 apresenta uma parte do Cenário de Execução utilizado no estudo de caso do Capítulo 4, onde o Arquivo de Clientes (E1) sofre uma seleção para geração do arquivo de Clientes Pessoa Física (T1).

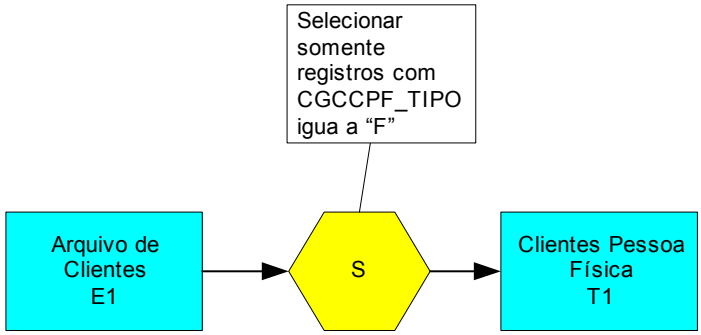


Figura 5. 2 - Detalhe do Cenário de Execução do Estudo de Caso

Observa-se que em não existe referência para controle de versão na notação acima apresentada. Para corrigir esse problema é proposta uma modificação na notação para a implementação de informações sobre a versão atual de cada item gráfico. A figura 5.3 apresenta a proposta para registro das informações de versão a ser implementada na notação do Modelo Conceitual [4].

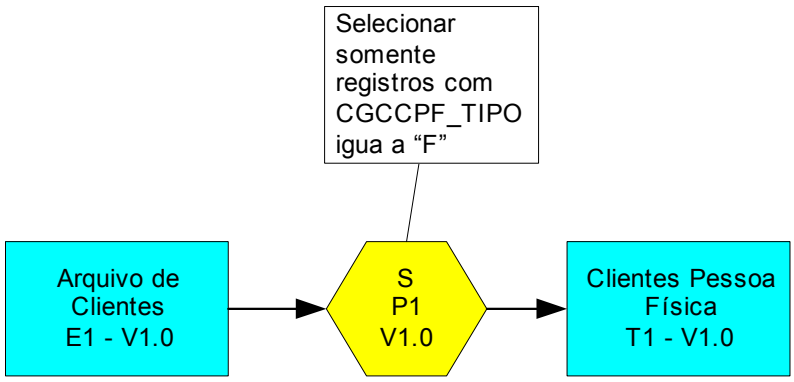


Figura 5. 3 - Proposta para registro de Informações de Versão para Modelo Conceitual [4]

Observa-se que a implementação da versão atual (V1.0) na notação dos itens gráficos de Conceito e Transformação, aplicados ao Cenário de

Execução permite saber qual é a versão de cada arquivo ou processo utilizado no cenário.

Observa-se, também, que foi necessária uma modificação na notação utilizada para a transformação, onde foi implementada, além da informação da versão, a notação P1. Essa notação utiliza-se de uma característica do Modelo Conceitual [4], também presente no Modelo Dissertativo, que é a necessidade do modelo de representar as etapas de processamento envolvendo poucas entidades relacionadas à transformação.

Dessa forma, a notação central do item de transformação no Cenário de Execução passa a representar a etapa de processamento na qual a transformação será utilizada.Por meio de uma funcionalidade da ferramenta de diagramação, também presente na maioria das ferramentas de diagramação e editores de texto comercializados no mercado, cria-se um *hiperlink* entre cada figura e seu arquivo original. Antes da criação do *hiperlink* é necessário estabelecer a estrutura de diretórios para armazenamento da documentação relativa à especificação de um processo de ETL utilizando o Modelo Conceitual [4]. A figura 5.4 apresenta uma proposta de estrutura de diretórios que pode ser utilizada para esse fim.

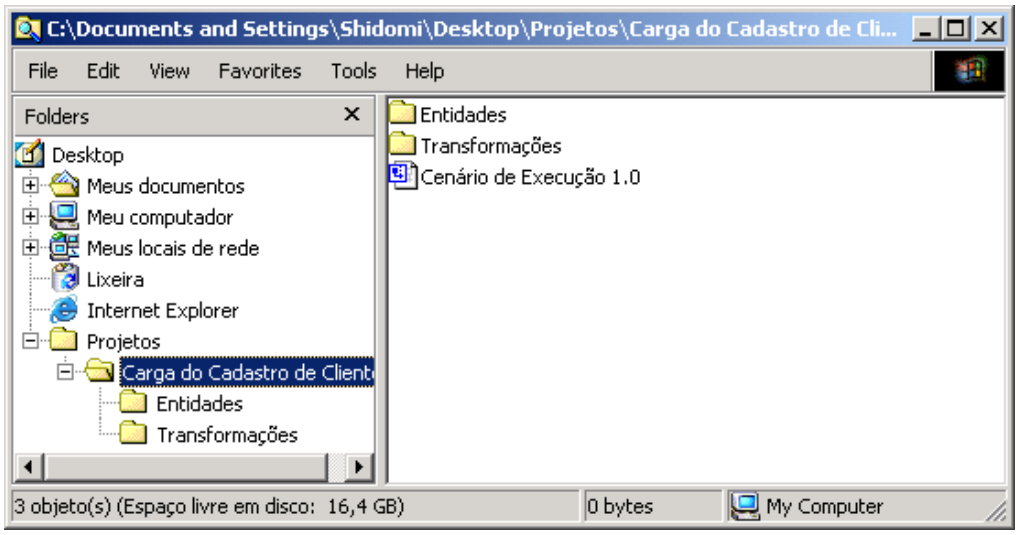


Figura 5. 4 - Estrutura de Diretórios para Controle de Configuração e Versão

Observa-se a divisão do projeto Carga do Cadastro de Clientes em dois diretórios distintos, Entidades e Transformações, ainda pode-se observar o Cenário de execução do processo de ETL no diretório raiz.

O diretório Entidades, como apresentado na figura 5.5, armazena o detalhamento de cada uma das entidades envolvidas no processo de ETL.

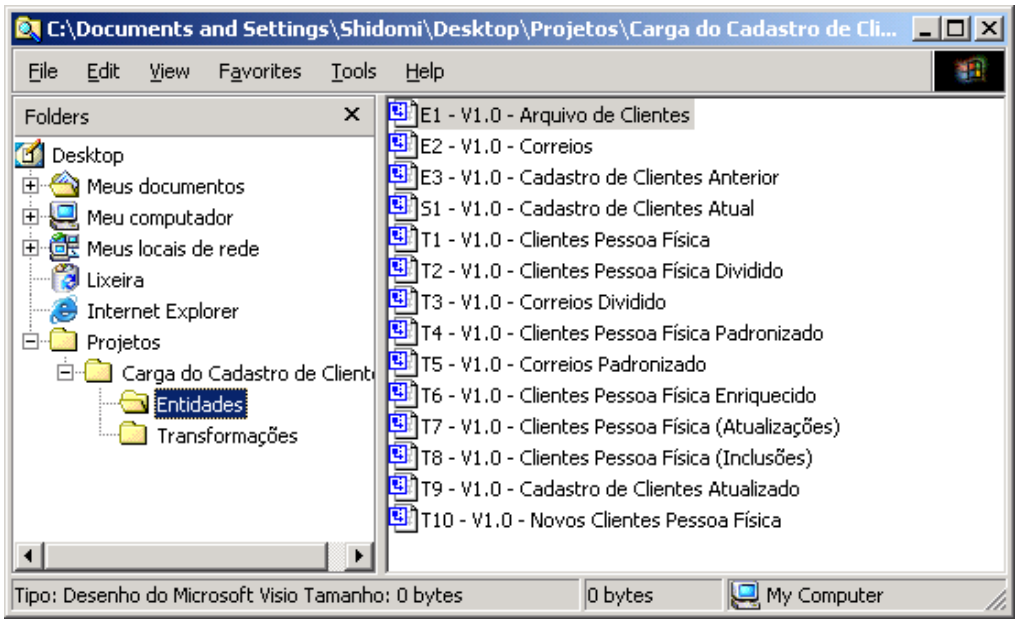


Figura 5. 5 - Diretório de Entidades

Cada arquivo do diretório Entidades, armazena as definições de uma entidade utilizada no Cenário de Execução e o conteúdo de cada arquivo deste diretório é exemplificado pela figura 5.6.

A figura 5.7, por sua vez apresenta o detalhamento do diretório Transformações que armazena o detalhamento de cada etapa de processamento utilizado no Cenário de Execução do processo de ETL.

A figura 5.8 exemplifica o conteúdo da etapa P1 do Cenário de Execução dentro do controle de versões. Observa-se que a descrição da etapa de processamento feito nesta figura utiliza a entidade, relacionada a um processo de seleção com o intuito de geração de um arquivo temporário T1.

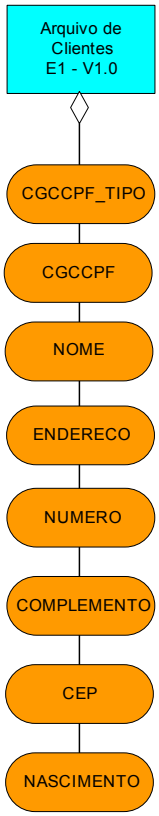


Figura 5. 6 - Conteúdo do Arquivo E1 do Diretório Entidades

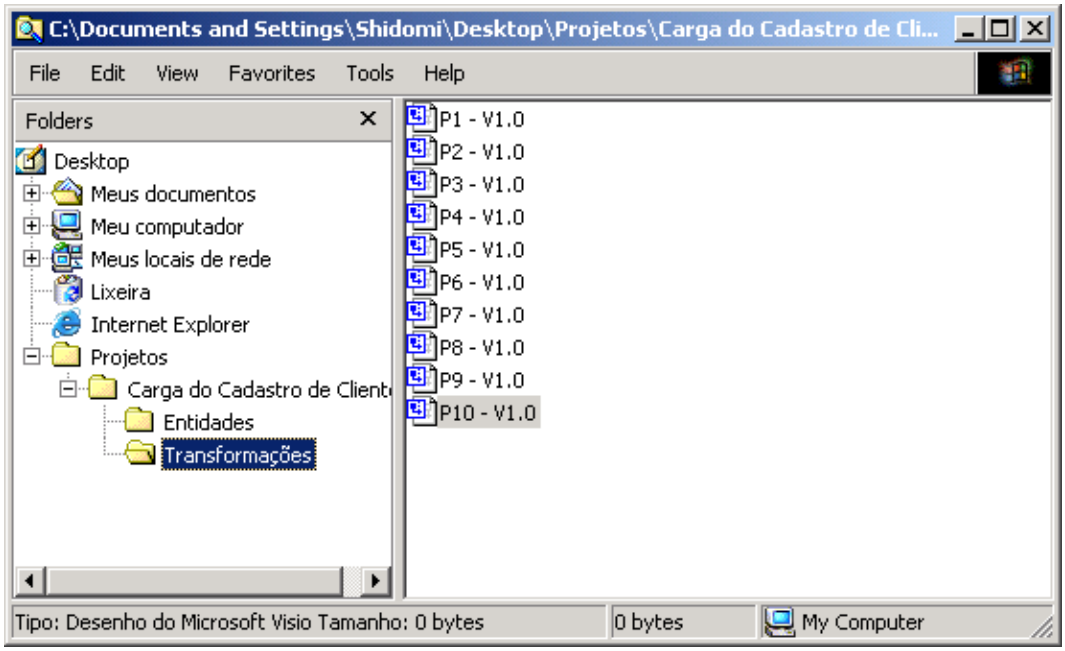


Figura 5. 7 - Diretório Transformações

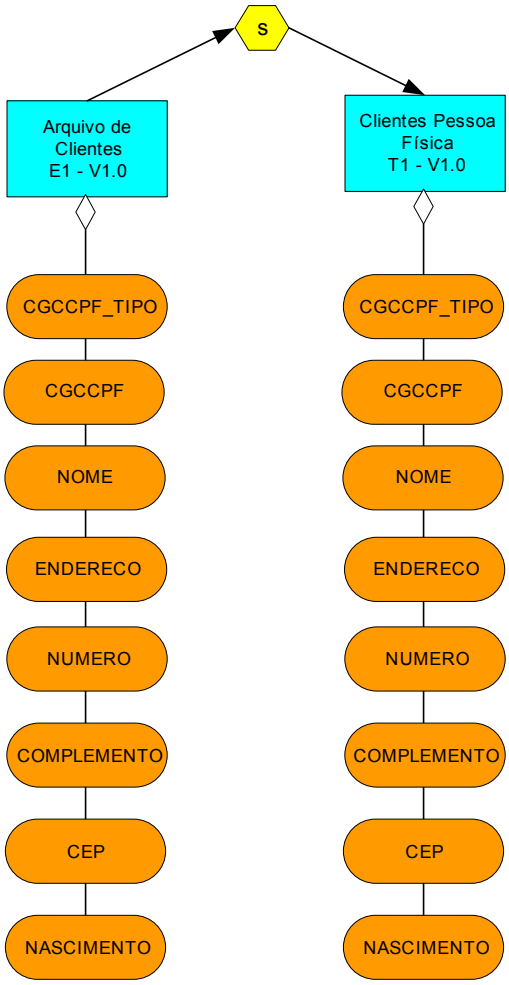


Figura 5. 8 – Conteúdo da Etapa P1 do Diretório de Transformações

As documentações, dispostas em seus devidos diretórios, podem então ser "ligadas" por meio da utilização de *hiperlinks* de forma a possibilitar, por meio do diagrama do Cenário de Execução, acessar todas as documentações do processo de ETL como se o diagrama fosse um índice do processo, utilizando para isso os *hiperlinks* gerados.

Para cada modificação efetuada no processo, seria gerada uma nova versão do índice (Novo Cenário de Execução) que ligaria outras versões das documentações dos diretórios de Entidades e Transformações à nova documentação de cenário.

Dessa forma, pela verificação da evolução dos documentos de cenário será possível identificar e acompanhar a evolução do processo de ETL no decorrer do tempo.

A estrutura da documentação, como proposta, possibilitará, também, implementar novas funcionalidades ou, retirar funcionalidades do cenário sem afetar as demais funcionalidades não envolvidas na alteração.

A implementação do controle de configuração e versão proposta possibilitará um aproveitamento maior das melhores características do Modelo Conceitual [4] sem perda do controle da documentação envolvida.

5.2.3. Item Gráfico para Filtragem e Conversão

Para um modelo que se propõe a facilitar a especificação de processos de ETL, as funções de filtragem (muito utilizadas na construção de processos ETL) não possuem representação própria quando da verificação ou conversão de valores de domínio (Tópico de Filtragem de Dados do item 3.3). Isso faz com que, no caso de haver uma verificação ou conversão de valores de domínio, devam ser representadas as tabelas de domínio que não tem relevância dentro do fluxo do processo.

As figuras 5.9 e 5.10 apresentam exemplos de Verificação de Domínio e Conversão de Domínio como devem ser representados no Modelo Conceitual [4]. Nesses exemplos observa-se que as tabelas de Verificação e Conversão de Domínio não são relevantes ao processo e acabam por dificultar uma visão mais clara do mesmo.

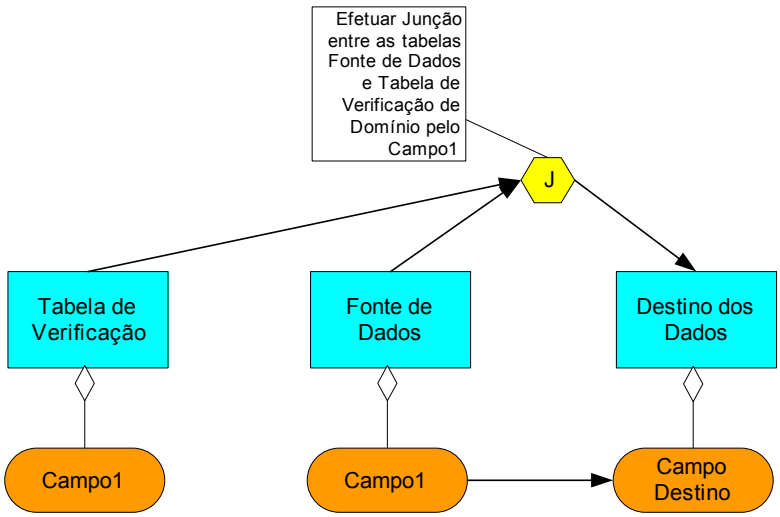


Figura 5. 9 - Verificação de Domínio no Modelo Conceitual [4]

Na notação utilizada no Modelo Conceitual [4], para a implementação de uma Verificação de Valores de Domínio, é utilizada uma junção com a tabela de referência (Tabela de Verificação de Domínio) de forma a “alimentar” o Campo Destino com os valores que possuem correspondência com essa tabela.

Análogo a Verificação de Valores de Domínio, a Conversão de Valores de Domínio utiliza uma junção para identificação do valor de correspondência na Tabela de Conversão, valor este utilizado na carga do Campo Destino. No exemplo da figura 5.7 a junção é efetuada utilizando-se o conteúdo do Campo1 como chave de pesquisa e o valor do Campo2 obtido é então utilizado na carga do Campo Destino.

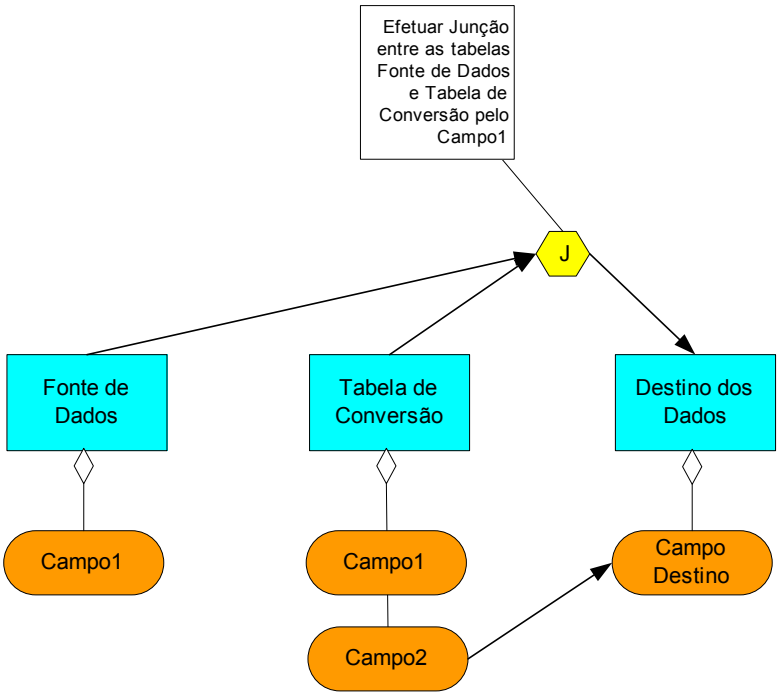


Figura 5. 10 - Conversão de Domínio no Modelo Conceitual [4]

A adoção de notação própria para representação da verificação e da conversão de domínio poderia facilitar a implementação dessas funcionalidades durante a diagramação. A figura 5.11 apresenta as sugestões para a implementação de novas notações para a verificação e a conversão de valores de domínio.

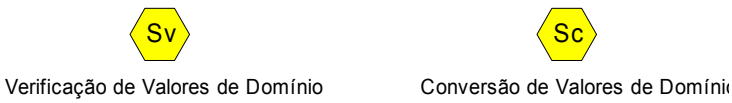


Figura 5. 11 - Sugestão para Verificação e Conversão de Domínio

As figuras 5.12 e 5.13 apresentam a aplicação das notações sugeridas de forma a ilustrar sua implementação no Modelo Conceitual [4].

Observa-se na figura 5.12 a eliminação da junção entre a Fonte de Dados e a Tabela de Verificação de Domínio. Neste caso os valores da Fonte de Dados são transferidos para o Campo Destino por meio da utilização de uma

funcionalidade específica, que efetua a verificação da existência do valor do Campo1 na Tabela de Verificação antes de movimentá-lo para o Campo Destino.

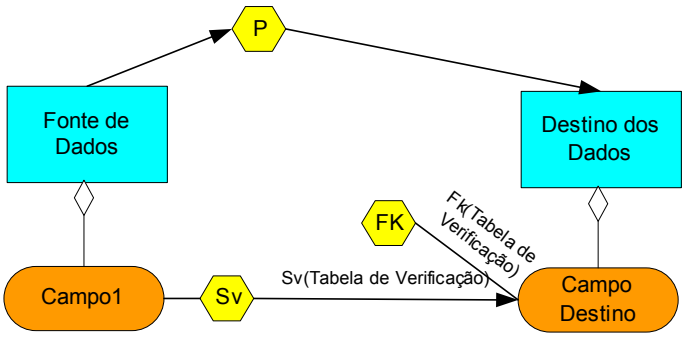


Figura 5. 12 - Sugestão de Melhoria para Verificação de Domínio

Observa-se que a criação de uma notação específica para identificar uma Verificação de Valores de Domínio auxilia a interpretação do diagrama e identifica melhor a necessidade da implementação da funcionalidade.

O mesmo ocorre na figura 5.13 que apresenta a sugestão de melhoria para a Conversão de Valores de Domínio.

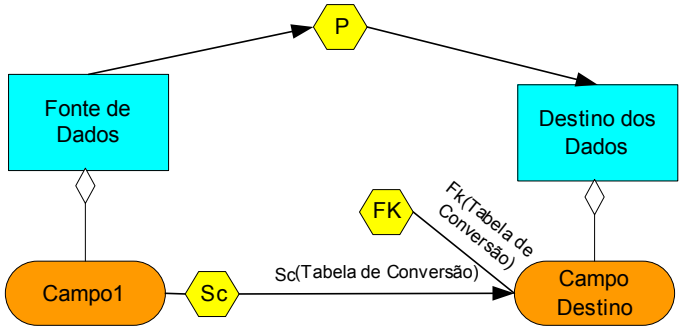


Figura 5. 13 - Sugestão de Melhoria para Conversão de Domínio

Uma Conversão de Valores de Domínio exige um processo de transformação onde é necessário o acesso a uma base de dados para obtenção de valores de correspondência. Isso deve ser feito antes da carga dos valores no destino dos dados.

Observa-se que a notação sugerida indica a necessidade da implementação de uma Conversão de Domínio no diagrama indicando a Tabela de Conversão que será utilizada sem a necessidade de incluir a Tabela de Conversão no diagrama, o que melhora a visualização do diagrama.

5.2.4. Conectores

A metodologia não prevê situações em que é necessário manipular muitos campos, o que tornaria o diagrama resultante incompreensível, por isso é necessário prover a notação com conectores adequados para possibilitar a continuação da especificação em páginas diferentes.

Na figura 5.14 são apresentados os conectores para páginas e para campos. A utilização desses dois tipos de conectores permitira uma melhor distribuição dos gráficos no papel.

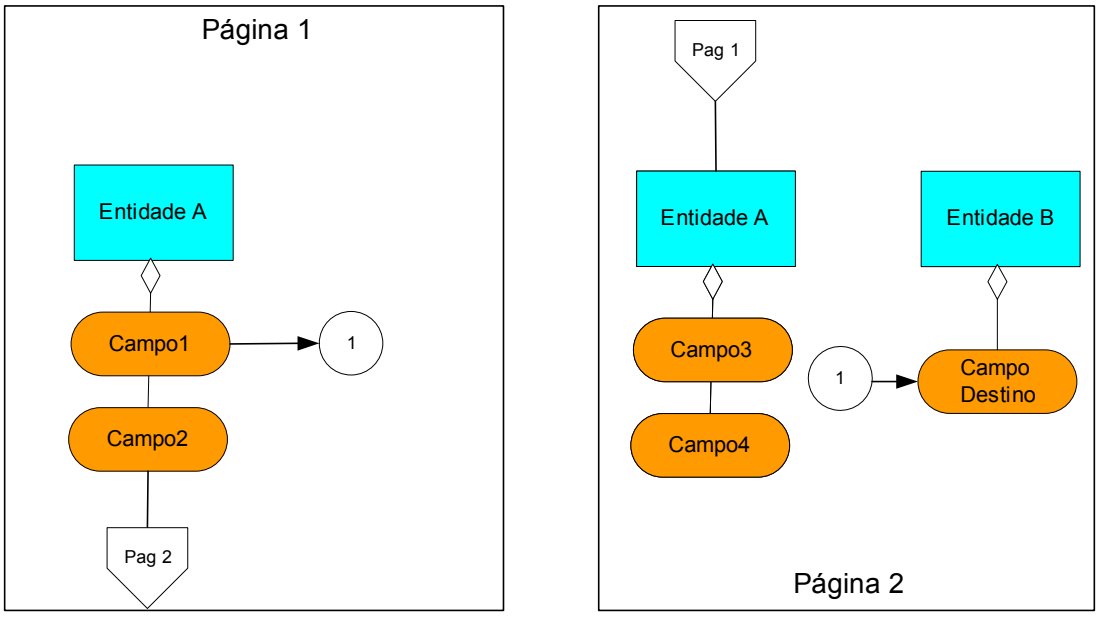


Figura 5. 14 - Sugestão para Conectores

5.2.5. Representação Compacta

Em geral num processo de ETL, a maioria das operações executadas, são relativas ao transporte de informações de um atributo para outro e portanto não envolvem transformações. Nesses casos a permanência desses atributos no diagrama prejudica a visualização pois acaba por desviar a atenção do leitor sobre os pontos mais importantes da especificação, além de consumir um esforço desnecessário na diagramação do gráfico. Todavia, retirar da especificação os atributos que não sofrem algum tipo de transformação deixaria a especificação incompleta.

Uma solução para o problema encontra-se na figura 5.15. Ela apresenta uma sugestão de modificação para o modelo onde a notação para atributo, no caso de campos que não sofrem transformação, seria modificada para comportar o conjunto de atributos que seria transportado sem modificação ao final de cada entidade. Essa modificação facilitaria a construção do gráfico porque reduziria a quantidade de itens gráficos manipulados e tornaria o diagrama mais claro.

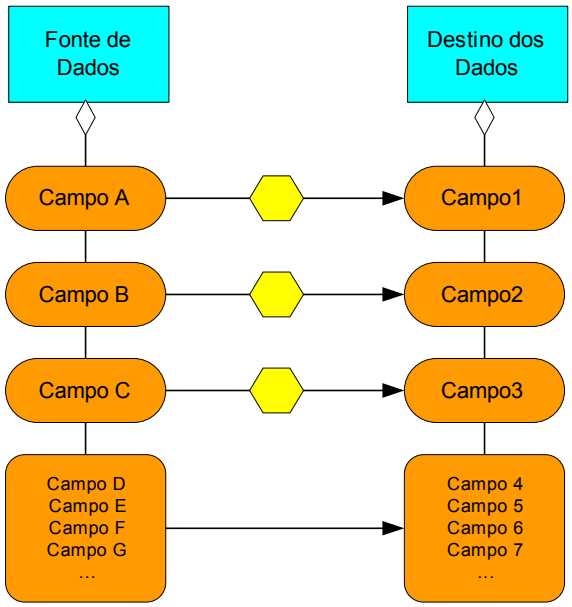


Figura 5. 15 - Sugestão para transporte de campos sem modificação

5.2.6. Funções Especiais

As notações do Modelo Conceitual [4] utilizadas no estudo de caso do capítulo 4 representam funcionalidades ETL genéricas e que podem ser utilizadas em várias situações na montagem dos diagramas. O modelo, sendo gráfico e conceitual, pode utilizar as representações genéricas para expressar a utilização de funcionalidades semelhantes mas muito mais complexas na especificação dos processos. Um caso desse tipo é a aplicação das funcionalidades na especificação de processos envolvendo qualidade de dados.

A funcionalidade de *Matching* (Item 2.2 Pág. 20) é uma extensão da funcionalidade de Junção do Modelo Conceitual [4] utilizada nos processos de qualidade de dados.

A funcionalidade de Junção, no Modelo Conceitual [4], efetua uma junção entre entidades com base em argumentos de pesquisa. Os processos de qualidade de dados que utilizam funções de *Matching* executam o mesmo trabalho, porém, levam em consideração complexos algorítimos estatísticos que fornecem o subsídio necessário para efetuar essa junção.

A figura 5.16 apresenta uma sugestão de melhoria para estender a notação de junção do Modelo Conceitual [4] de forma comportar funções de identificação. No exemplo é demonstrada uma identificação entre duas fontes de dados distintas com o objetivo de eleger a informação que será carregada no destino dos dados.

Na nota explicativa da figura é apresentada as regras de identificação dos registros selecionados para a carga onde identifica-se uma regra geral para efetuar a junção das duas fontes de dados com o objetivo de efetuar a carga do destino dos dados e um regra alternativa que identifica por meio de um sistema de pesos (*matching*) qual é o fluxo de execução a ser tomado.

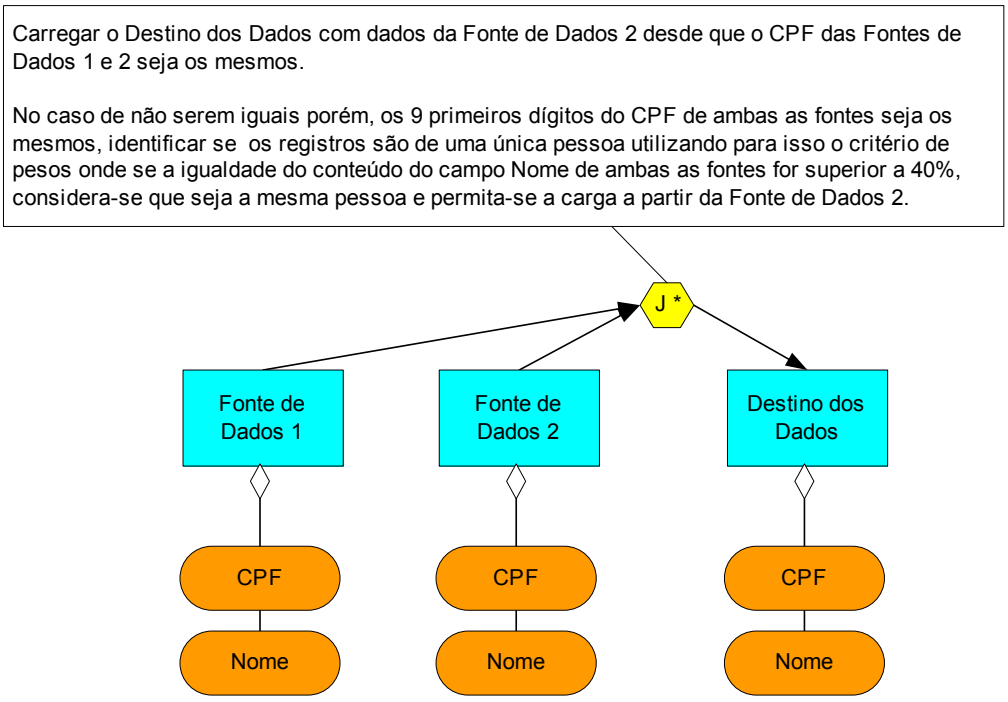


Figura 5. 16 - Sugestão para Identificação

5.3 Conclusão

Este capítulo apresentou diversas melhorias ao Modelo Conceitual [4] de especificação de processo de ETL, a saber: - Níveis de abstração mais elevados para observação de um processo de ETL; Controle de configurações e versões para os documentos de um processo de ETL; Novos itens gráficos para filtragem e conversão de valores de domínio; Conectores para indicações de continuidade do diagrama em outras páginas; Representações compactas para atributos que facilitam a diagramação e Funções especiais que estendem funções do próprio modelo para abarcar funcionalidades complexas.

6 Conclusões

6.1 Resumo

Este trabalho apresentou um estudo de caso utilizando formas de especificação de processos de ETL distintas. Uma delas, já em uso no ambiente de estudo, é conhecido como Modelo Dissertativo. Neste modelo as transformações são descritas textualmente de forma detalhada. O outro modelo de especificação utilizado no estudo de caso foi o Modelo Conceitual [4] que, por meio de uma notação gráfica e de uma metodologia de aplicação.

Fez-se um estudo de caso utilizando os dois modelos de especificação, onde ambos os modelos foram avaliados segundo os critérios de Representação Compacta; Usabilidade; Visão Integrada; Automatização; Reutilização; Padronização; Legibilidade e Adaptabilidade.

O Modelo Conceitual [4] mostrou-se melhor do que o Modelo Dissertativo em Representação Compacta, Visão integrada, Automatização, Reutilização, Padronização, Legibilidade e Adaptabilidade.

Ainda assim, o Modelo Conceitual [4] apresentou pontos que poderiam ser melhorados. Para tanto foram propostas melhorias como Níveis de Abstração, Controle de Configurações e Versões, Novos Itens Gráficos para Filtragem e Conversão de Domínios, Conectores, Representações Compactas para Atributos e Funções Especiais, tendo em vista completar estes aspectos que não são abordados na proposta original, aspectos estes que facilitariam a utilização das notações e metodologia na confecção de especificações para processos de ETL utilizando o modelo.

6.2 Contribuições

Este trabalho abordou um estudo de caso utilizando os modelos Dissertativo e Conceitual [4] de especificação. A avaliação dos dois modelos trouxe como contribuição não só as melhorias sugeridas à notação e a metodologia de aplicação do modelo gráfico, que por si só, representam uma contribuição importante.

Os resultados da avaliação feita pela utilização dos dois modelos de especificação, também fornecem uma contribuição importante na medida que podem ser utilizados como referência de avaliação de outros modelos.

A idéia da avaliação de um modelo pela utilização em um estudo de caso comparativo é, também, uma contribuição importante na medida que essa experiência pode ser repetida na avaliação de outros modelos de especificação.

A comparação entre os dois modelos de especificação, utilizados no estudo de caso deste trabalho, também contribuiu para identificar as vantagens e desvantagens que a utilização de um modelo gráfico e conceitual de especificação de processo de ETL, utilizado em substituição a um modelo essencialmente dissertativo, pode trazer.

A utilização do modelo estendido, como base para construção de uma nova ferramenta gráfica de especificação de processo de ETL, também pode ser considerada como uma contribuição na medida que indica um caminho de evolução para o modelo.

6.3 Trabalhos Futuros

Este trabalho abordou a utilização de um modelo gráfico para confecção de especificações de processos de ETL num estudo de caso onde foi

identificado que algumas melhorias poderiam ser implementadas a esse modelo.

Uma das melhorias que poderiam ser implementadas seria o Controle da configuração e versionamento para o modelo, porém, esse item merece um trabalho à parte para construção de um banco de dados de configuração e versão, a exemplo do que é sugerido pelas áreas de engenharia de software e banco de dados.

Outra atividade para abordagem futura seria a construção de ferramentas gráficas e editores para automatizar o ambiente de desenvolvimento de processos ETL e a própria documentação envolvida.

Aplicar o Modelo Conceitual [4] em outros ambientes com características diferentes ainda na tentativa de avaliar melhor a adequação da utilização do modelo e criação novos itens gráficos seria, também, um item para abordagem futura.

Referências Bibliográficas

[1] BLOOR. White Papers. In: **ETL - Extract, Transform, Load**
Disponível em:
http://www.bloor-research.com/research_library.php?pid=146
[26/09/2004 11:00]

[2] INMON, W.H.: **A Brief History of Integration** In: Articles
Disponível em : http://www.inmoncif.com//library/articles/histint.asp
[26/09/2004 23:00]

[3] JOHNSON, JEFF & HENDERSON, AUSTIN - **Conceptual Models: Begin by Designing What to Design** In: Articles
Disponível em : http://doi.acm.org/10.1145/503355.503366
[24/07/2003 19:00]

[4] VASSILIADIS, P. & SIMTSIS, A. & SKIADOPOULOS, S. - **Conceptual Modeling for ETL Processes** In: College of Information Science & Tecnology
Disponível em :
http://www.cis.drexel.edu/faculty/song/dolap02/paper/p14-vassiliadis.pdf
[26/09/2004 23:00]

[5] VASSILIADIS, P. & SIMTSIS, A. & SKIADOPOULOS, S. - **On the Logical Modeling of ETL Processes**
Disponível em :
http://citeseer.ist.psu.edu/rd/13200248%2C556795%2C1%2C0.25%2CDownload/http://citeseer.ist.psu.edu/cache/papers/cs/26935/http:zSzzSzwww.mm.di.uoa.grzSz%7ErouvaszSzssizSzcaise2002zSz23480782.pdf/vassiliadis02logical.pdf
[26/09/2004 23:00]

[6] NOVO AURÉLIO - **Dicionário da Lingua Portuguesa** In: UOL
Disponível em:
http://www.uol.com.br/aurelio/index_result.html?stype=k&verbete=transforma%E7%E3o&x=0&y=0 [26/11/2003 23:00]

[7] MILLET, IDO & PARETE, DIANE H. & FIZEL, JOHN L. - **Data Warehouse Design for Ease of Data Transformation** In:
Disponível em :
http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-58/millet.pdf [26/11/2003 23:00]

[8] HERNANDEZ, MICHAEL J. - **What Are Views?**
Disponível em :
http://msdn.microsoft.com/library/default.asp?url=/library/em-us/dnacbk02/html/odc_DBs4Mortals.asp [26/11/2003 23:00]

[9] INMON, W. H. - **Como Construir o Data Warehouse 2a. Edição**. Pages. 255 –285

[10] STONEBRAKER, MICHAEL - **Too Much Middleware**
Disponível em:
http://www.acm.org/sigmod/record/issues/0203/industry-ms.pdf [24/07/2003 19:00]

[11] VASSILIADIS, P. & SIMTSIS, A. & SKIADOPOULOS, S. - **Modeling ETL Activities as Graphs**
Disponível em: http://www.cs.uoi.gr/~pvassil/publications/dmdw02.pdf [26/11/2003 23:00]

[12] INMON, W. H. & WELCH, J. D. & GLASSEY, KATHERINE L. - **Managing the Data Warehouse**

Pages: 161 – 181

[13] HUMPHRIES, HAWKINS, & DY - **Data Warehouse Architecture and Implementation**
Page: 198

[14] INMON, W.H. - I**ntegration and Transformation**
Disponível em:
http:// www.inmoncif.com /library/articles/introit.asp
[26/09/2004 23:00]

[15] ECKERSON, WAYNE & WHITE, COLIN - **Evaluating ETL and Data Integration Plataform**
Disponível em:
http://download.101com.com/tdwi/research_report/2003ETLReport.pdf [26/09/2004 23:00]

[16] NISSEN, GARY - **Is Hand-Coded ETL the Way to Go ?**
Disponível em :
http://www.intelligententerprise.com/030531/609warehouse1_1.shtml
[26/09/2004 23:00]

[17] GONÇALVES, MARCIO - **Extração de Dados para Data Warehouse**
Página: 51

[18] BRACKETT, MICHAEL H. - **The Data Warehouse Challenge**

[19] DEVLIN, BARRY - **Information Integration - A Step Beyond the Data Warehouse**
Disponível em:

http://www.dw-institute.com/research/display.asp?id=6665
[26/11/2003 23:00]

[20] INMON, W. H. **- Integration/Transformation Complexity**
Disponível em:
http://www.billinmon.com//library/articles/itcompl.asp
[26/11/2003 23:00]

[21] ECKERSON, WAYNE & WHITE, COLIN - **Evaluating ETL and Data Integration Plataform**
Disponível em:
http://mimage.hummingbird.com/alt_content/binary/pdf/collateral/mc/etlreport.pdf [26/11/2003 23:00]

[22] VASSILIADIS, P. & SIMITSIS, A. & GEORGANTAS, P. & TERROVITIS, M. - **A Framework for the Design of ETL Scenarios**
Disponível em:
http://www.cs.uoi.gr/~pvassil/publications/caise03_long.pdf
[26/11/2003 23:00]

[23] HOLTEN, ROLAND - **Conceptual Models as Basis for Integrated Information Warehouse Development**
Disponível em:
http://www.wi.uni-muenster.de/inst/arbber/ab81.pdf [26/11/2003 23:00]

[24] LUJÁN-MORA, SERGIO & TRUJILLO, JUAN - **A comprehensive Method for Data Warehouse Design**
Disponível em:
http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-77/01_Lujan.ps [26/11/2003 23:00]

[25] ENGLISH, L. **10 years of Information Quality Advances: What Next ?** In: DM Review, Fevereiro 2001.
Disponível em
http://www.dmreview.com/master.cfm?NavID=55&EdID=3009
[01/10/2002 23:35]

[26] MASSACHUSETTTS INSTITUTE OF TECHNOLOGY. **The MIT Total Quality Management Program**.
Disponível em:
http://web.mit.edu/tdqm/www/about.shtml [01/10/2002 22:10]

[27] ENGLISH, L. **Improving Data Warehouse and Business Information Quality**. New York: Wiley, 1999.

[28] FIRSTLOGIC. **Customer Data Quality – Building the Foundation for a One-to-one Customer Relationship. A White Paper**.
Disponível em: http://www.firstlogic.com/pdfs/db_oldWhitepaper.pdf
[08/10/2002 21:12]