Licenciatura em Gestão de Sistemas e Tecnologias da Informação

Título do Trabalho

Migração de bases de dados, metodologias e ferramentas

Elaborado por João Paulo Cabrita Alves Pires

Aluno nº 20000560

Orientador: Professor João Valentim

Barcarena

Novembro de 2008

Universidade Atlântica

Licenciatura em Gestão de Sistemas e Tecnologias da Informação

Título do Trabalho

Migração de bases de dados, metodologias e ferramentas

Elaborado por João Paulo Cabrita Alves Pires

Aluno nº 20000560

Orientador: Professor João Valentim

Barcarena

Novembro de 2008

DECLARAÇÃO

Nome : João Paulo Cabrita Alves Pires

Endereço electrónico: joaopires@netvisao.pt Telefone: 965774246

Número do Bilhete de Identidade: 8443946

Título do Trabalho : Migração de bases de dados, metodologias e ferramentas

Orientador(es): Professor João Valentim

Declaro que concedo à Universidade Atlântica uma licença não-exclusiva para arquivar e tornar acessível, o presente trabalho, no todo ou em parte.

Retenho todos os direitos de autor relativos ao presente trabalho, e o direito de o usar futuramente

Assinatura

Universidade Atlântica, Barcarena 05/11/2008

Só é possível a produção de conhecimento porque se alicerça nas fundações dos que nos precederam.

O autor é o único responsável pelas ideias expressas neste relatório

Agradecimentos

Ao Professor João Valentim pelo constante apoio e acompanhamento na realização do presente trabalho.

Aos familiares, pela ajuda e pela ausência.

A todos os outros, muitos, a quem devo o suficiente para ter conseguido efectuar o presente.

Resumo

Migração de bases de dados, metodologias e ferramentas

Este trabalho versa na temática da migração das bases de dados com enfoque particular na distinção da granularidade que pode conter. Assim, face à complexidade são propostas diversas abordagens necessárias para a resolução dos problemas característicos, em causa. São propostas diversas metodologias de migração, é abordada a sua evolução histórica e são apresentadas ferramentas de suporte à migração com um estudo de caso comparativo de três delas.

Base de dados, ETL, Metodologias de migração.

Abstract

Database migration, methodologies and tools

The present work approaches database migration with particular focus on the distinction of the granularity that it can contain. Thus, given the complexity, several approaches are necessary for the resolution of the characteristically problems concerned. Several migration methodologies are proposed, it's addressed his historical development and are presented tools that support and automate migration with a comparative case study of three of these.

Database, ETL, Data Migration methodology

Índice

Agradecimentos iii

Resumo iv

Abstract iv

Índice vi

Índice de figuras ix

Índice de tabelas x

Lista de abreviaturas e siglas xi

Introdução 1

1. Migração de bases de dados 2

1.1. O projecto 9

1.2. Aspectos específicos de bases de dados 13

1.2.1. Modelação dos dados 13

1.2.2. O esquema 16

1.2.3. Segurança 20

1.3. Aspectos específicos do processo de migração 21

1.3.1. Avaliação da qualidade dos dados 21

1.3.2. Reengenharia dos dados 22

1.3.3. Testes 23

1.3.4. Implementação e verificação final 27

1.4. Casos particulares de migração 29

1.4.1. Migração de sistemas legados 29

1.4.2. Migração de tipos de bases de dados diferentes 31

2. Cenários e metodologias de Migração 34

2.1. Migração ligeira 35
2.1.1. Actividades chave no processo 36
2.2. Migração Média 38
2.2.1. Actividades chave no processo 39
2.2.2. Outras actividades relevantes 40
2.2.2.1. Gestão de meta dados 41
2.2.2.2. ETL, desenho e desenvolvimento 43
2.3. Migração complexa 44
2.3.1. Avaliação e planeamento da migração complexa 48
2.3.1.1. Esquema de avaliação de negócio e definição estratégica 50
2.3.1.2. Esquema de avaliação e opção tecnológica 52
2.3.1.3. Roadmap e actividades fundamentais 54
3. O Modelo OAIS e a preservação digital 56
3.1. O modelo OAIS 56
3.2. Repositórios digitais 58
3.3. A preservação digital 59
3.3.1. Objecto digital 60
3.3.2. Estratégias para a preservação digital 61
3.3.2.1. Preservação de Tecnologia 62
3.3.2.2. Refrescamento 62
3.3.2.3. Emulação 62
3.3.2.4. Migração 63
3.3.2.5. Normalização 63
3.3.2.6. Encapsulamento 64

4. Ferramentas ETL 65

4.1. Business Intelligence e Data Warehouse 66

4.2. Mercado das ferramentas ETL 70

4.2.1. Limitações das ferramentas ETL Open Source 73

4.3. Metodologia de avaliação de ferramentas de ETL 74

4.4. Comparação de ferramentas ETL na execução de uma migração 79

Conclusão 91

Bibliografia 93

Índice de figuras

Figura 1 - Diagrama Entidade relacionamento 14

Figura 2 - Modelo lógico dos dados 15

Figura 3 - Modelo físico dos dados 15

Figura 4 - Metodologia de investigação da qualidade dos dados 22

Figura 5 - Visão geral do processo de testes 24

Figura 6 - Visão geral dos tipos de testes 26

Figura 8 - Cenário de migração média 39

Figura 9 - Requisitos necessários ao desenvolvimento de um modelo de gestão de meta dados 41

Figura 10 - Enquadramento do desenho conceptual no ciclo de vida do desenvolvimento de “software” 44

Figura 11 - Cenário de migração complexa 46

Figura 12 - Componentes chave do centro integrado e operacional de armazenamento de dados 47

Figura 13 - Modelo de portfolio aplicacional 51

Figura 14 - Modelo de gestão da informação, tipos e conteudos 52

Figura 15 - Níveis de abstracção presentes num objecto digital. 61

Figura 16 - O ambiente de BI abordado como uma refinaria. 67

Figura 17 - Sistema de integração de informação em ambiente DB2. 69

Figura 18 - Estrutura básica de um sistema de BI. 70

Figura 19 - Ilustração do processo de Data Flow realizado no Integration Services (1) 82

Figura 20 - Ilustração do processo de Data Flow realizado no Integration Services (2) 83

Figura 21 - Ilustração do processo de Data Flow realizado no Integration Services (3) 84

Figura 22 - Ilustração do processo de Data Flow realizado no Talend (1) 85

Figura 23 - Ilustração do processo de Data Flow realizado no Talend (2) 86

Figura 24 - Ilustração do processo de Data Flow realizado no Talend (3) 87

Índice de tabelas

Tabela 1 - Mapeamento dos grupos de processo com as diferentes áreas de conhecimento 11

Tabela 2 - Recursos necessários à migração 35

Tabela 3 - Escala quantitativa 77

Tabela 4 - Matriz de avaliação 77

Tabela 5 - Peso relativo das valências 78

Tabela 6 - Peso relativo dos critérios de avaliação 79

Tabela 7 – Escala quantitativa da migração 88

Tabela 8 – Atribuição dos pesos relativos dos critérios de avaliação no processo de migração 89

Tabela 9 – Valor final da avaliação das ferramentas de ETL 90

Lista de abreviaturas e siglas

ANSI: American National Standards Institute

BCNF: Boyce-Codd Normal Form

BD: Base de Dados

BI: Business Intelligence

DF: Data Flow

DSS: Decision Support System

DW: Data Warehouse

EAI: Enterprise Application Integration

ERP: Enterprise Resource Planning

ELT: Extract Load Transform

ETL: Extract, Transform, Load

IODS: Integrated and Operational Data Center

IS: Integration Services

SGBD: Sistema de Gestão de Bases de Dados

SGBDOO: Sistema de gestão de Bases de Dados Orientadas a Objectos

SOA: Service Oriented Architecture

SPARC: Standards Planning and Requirements Committee

TL: Talend

XML: Extensible Markup Language

Introdução

Desde tempos imemoráveis que se procede ao armazenamento e migração de dados. O suporte físico do primeiro será o cérebro, referindo-me às memórias, e o suporte do segundo poderá ser a voz, na transmissão oral. Ainda hoje, e apesar dos progressos, é facilmente perceptível o insucesso das tentativas de replicação deste sistema humano, com o qual continuamos a aprender.

Ainda assim, e também porque têm muito para evoluir, os sistemas computacionais renovam-se periodicamente originando a implementação de novas funcionalidades aos níveis de hardware e software com melhor desempenho.

A este ciclo associa-se um outro, que é o do software que suportam onde se incluem as bases de dados. Estas, também sujeitas às marés, evoluem, e poderá emergir a necessidade de as adoptar. O evolucionismo, no entanto, não será o único responsável pelas migrações. Muitos outros motivos poderão estar na sua origem como veremos adiante.

Abordar-se-á esta temática, numa perspectiva histórica, estudando diversas abordagens realizadas por vários autores. Dividir-se-á para conquistar, no que respeita a passos a seguir em migrações mais complexas. Planificar-se-á para sabermos para onde nos dirigir. E proceder-se-á à validação de tudo o que foi feito.

Apresentar-se-ão, também, ferramentas que permitem automatizar os processos de migração, ou pelo menos parte deles, se a sua complexidade for muito grande. Far-se-á uma análise comparativa das mesmas e proceder-se-á à elaboração de uma matriz de comparação entre elas.

Por fim, executa-se um processo de migração em que se pré estabelece a utilização de três destas ferramentas, e com base na matriz de comparação apresentada anteriormente efectuar-se-á uma análise das mesmas.

1. Migração de bases de dados

Com o aumento do uso dos computadores e respectivas aplicações informáticas das últimas décadas, foram criadas um grande número de diferentes tecnologias de gestão de dados, que vão desde sistemas com arquivos de texto até sistemas de gestão de bases de dados relacionais ou orientados a objectos. Independentemente do tipo de armazenamento de dados existente, o que à época era a melhor solução, hoje poderá estar desactualizado, ou em vias de entrar em desuso. Nestes casos, é tomada a decisão de migrar os dados de um sistema para outro.

A prescrição da tecnologia, se assim se pode designar, não é a única razão para a migração de dados. A diminuição do custo com o licenciamento, a procura de soluções com incremento de desempenho, estratégias de consolidação de várias tecnologias, a junção de organizações ou a extensão do acesso aos dados a um número maior de utilizadores serão também motivos suficientes para que se proceda a uma migração de dados.

As razões extrínsecas conducentes a um projecto deste tipo podem radicar nos próprios sistemas de suporte às bases de dados, pois não existem, elas próprias como um fim, mas sempre como um meio. A migração de um sistema legado para um novo sistema pode ser uma actividade simples ou complexa dependendo dos motivos da mesma, podendo assumir a seguinte natureza:

- Uma migração de um sistema simples para outro sistema.
- A promoção da melhoria de um sistema, vulgo “upgrading”, para uma nova versão com a alteração dos dados subjacentes.
- A convergência de múltiplos sistemas num único.
- A migração complexa de um sistema para outro com implementação prolongada no tempo, ou seja, com “roll out” faseado.

- A migração de sistemas múltiplos, a correr em paralelo, para um único, num esforço de consolidação. Esta tipologia costuma designar-se como "IT Transformation".

Já em 1974, o estudo de Housel para a IBM é um dos exemplos em que alguns dos aspectos relevantes da migração de dados são identificados a par com a sugestão de áreas futuras de pesquisa. As causas principais de migrações de dados então identificadas, continuam tão actuais e presentes como nos nossos dias, pelo menos a nível geral.

Considerava então as seguintes:

- Alteração do "hardware" do sistema
- Migração de sistema de suporte.
- Modificações na estrutura ou programa em resultado de alterações dos requisitos da aplicação.
- A agregação de uma nova aplicação a uma base de dados já existente.

No entanto, apesar de no geral os motivos continuarem a ser essencialmente os mesmos, relevo os factores custo e complexidade das soluções adoptadas. O primeiro, pelo facto de não ser considerado da mesma forma, de 1974 até à data. O segundo, pela crescente panóplia de soluções possíveis para a resolução do mesmo problema, acarretando problemas diferentes a nível de decisão.

Nos anos 70 o foco das várias pesquisas em curso aproximava-se da tentativa de definir linguagens comuns que permitissem a definição de dados, armazenamento e processos de mapeamento. Housel et al. (1974) referiram-se a esta abordagem, ou tentativa, como causa provável de problemas.

Esta definição por parte dos utilizadores incluiria a escolha das tarefas mais importantes de um projecto de migração, como a determinação das aplicações, transformação dos dados, alteração de programas e desenvolvimento. Estes serão necessários a

determinado nível de intervenção humana obrigatória em projectos de grande complexidade, mas também propõem a automatização de determinadas tarefas como a transformação dos dados que pode ser conseguida através da utilização de linguagens de alto nível.

O papel destes últimos não incluirá a escolha de uma linguagem específica, mas sim a definição dos requisitos destas incluindo o número de categorias no mapeamento da origem para o destino bem como a verificação da integridade dos dados da migração, uma vez finalizado este processo.

Esta formalização processual foi ganhando consistência e numa época de grandes transformações marcada por inúmeros movimentos de consolidação e concentração empresarial à escala global como foram os finais dos anos 80 e toda a década de 90 tornou-se objecto de mais e maiores estudos. Não é portanto de estranhar a dinâmica que esta matéria ganhou durante aquele período a ponto de ainda hoje ser dessa época o maior número de referências sobre a matéria.

A meados dos anos 80, o nível de preocupação e as bases de dados em causa já eram semelhantes aos actuais e continuam a ser válidos, senão, veja-se que Elmasri et al. (1984, 1986) abordam a importância da integração do esquema. Dizem que no carregamento de dados há tabelas cujo carregamento é prioritário, pois são tabelas de apoio com valores de referência, das quais dependerá a validação das restrições de integridade. O caso tornar-se-á ainda mais crítico se o destino for um sistema distribuído composto por várias aplicações.

Vai entretanto emergindo a procura de uma metodologia transversal ao processo de migração. A identificação cada vez mais detalhada das suas actividades é também, em primeira análise, a procura da solução para este reflexo da maior complexidade da realidade que lá vive espelhada.

No início dos anos noventa, Youn and Ku (1992) dão-nos um resumo bastante completo de todas as questões essenciais no processo de migração de dados, entre duas bases de dados, origem e destino, ou mesmo, entre várias de origem e outras tantas no destino.

Dizem que o que pode, e de facto dificulta todo este processo, sendo portanto objecto de análise, é o facto de a origem e o destino serem frequentemente diferentes no seu desenho e especificações. Assim, todo o processo de migração é acompanhado permanentemente de decisões pontuais sobre quais as estruturas e que dados serão necessários no destino, se necessitarão de algum tipo de transformação e que tipo de alterações haverá a efectuar. Alguns campos poderão desaparecer, outros poderão ser concatenados num único. As regras de negócios são analisadas, reavaliadas e actualizadas.

Quando a origem e o destino são estruturalmente diferentes ou quando os dados são inconsistentes nas suas diversas origens, terá que haver uma série de decisões quanto à forma mais fiável de migração dos mesmos não deixando de observar ou mesmo eliminar a hipótese de erros na sua transmissão entre origem e destino.

A abordagem destes autores fragmenta o procedimento de migração em extracção e carregamento, seguido de transformação e integração dos dados. Como parte do processo inicial de planeamento releva a necessidade de desenvolvimento de um modelo conceptual do sistema de origem que poderá e deverá ser usado a posteriori para o desenvolvimento do sistema de destino.

Aqui, assumirá também especial importância a comparação dos respectivos domínios, identificando a sua semelhança ou diferença. Ainda, o facto de poderem existir potenciais inconsistências de valores e a identificação das tabelas de mapeamento que se podem utilizar para ajudar à transformação de campos para estes casos em que existe inconsistência de dados é assumido como bastante importante.

Outro estudo realizado por Moriarty e Hellwege (1998) tem o seu enfoque num aspecto particular de migração de dados, caso das “Data Warehouse” (DW), em que o fluxo migratório é constante e com grande volume de dados. A migração dá-se quase, ou mesmo, em tempo real, em alguns casos, e os dados provêem com frequência de muitas aplicações. Aqui, o aumento do fluxo migratório obriga ao aumento proporcional de relatórios de erros e de alteração dos intervalos de tolerância a falhas no que respeita à qualidade dos dados.

Já no final dos anos 90, Hudicka (1999), estabelece uma sequência ordenada de fases, não sobrepostas, no processo de migração de bases de dados.

- Na fase pré-estratégica o gestor de projecto deverá identificar a quantidade de sistemas legados e contar as suas estruturas de dados. Os "interfaces" deverão também, se possível, ser identificados aqui.

- Na fase seguinte, estratégica, os utilizadores deverão quantificar o volume de dados em causa através da contagem de linhas e colunas e demais estatísticas relativas aos dados de origem em causa.

- Após esta, na fase de pré-análise deverão ser atribuídas as tarefas a realizar. Será também uma boa oportunidade para criar um ambiente de testes. Terá a vantagem adicional de permitir aos utilizadores um conhecimento prévio do sistema final, facilitando desta forma a aprendizagem ao dilatar no tempo a curva de aprendizagem.

- A fase de análise deverá consistir de, uma ou mais "check-list" de dados de origem que serão alvo de migração, e ainda de sessões com os utilizadores para incluir requisitos específicos a incorporar no novo sistema. É ainda nesta que se deverá efectuar o mapeamento de restrições de integridade chave e o mapeamento dos dados do nível lógico para o nível físico.

- O domínio da fase de testes subsequente deverá ser a correcção dos erros sintácticos de ambos os níveis, lógico e físico. E uma vez migrados os dados de teste deverão ser feitas as seguintes perguntas: Quantos registos deveriam ter sido criados? Quantos foram criados? Os dados migraram para os campos correctos? O formato dos dados foi o correcto?

Outras questões relevantes não abordadas por Hudicka seriam também:

Os dados de origem contêm valores nulos? Em caso afirmativo, a respectiva migração foi bem sucedida? A precisão dos valores numéricos migrou correctamente? A ter havido erros na migração das restrições de integridade é possível determinar quais os

valores que os originaram? Caso as BD de origem não tenham restrições de integridade, como assegurá-las na BD de destino?

É ainda nesta fase, de pré-teste, teste, implementação, que Hudicka defende que os utilizadores devem partir desde logo para o uso das estruturas de dados de destino pois, estando mais familiarizados com os dados e com as variações existentes no seu relacionamento intrínseco, poderão mais facilmente proceder à sua validação. As últimas fases serão as de revisão e manutenção.

Para finalizar, advoga a utilização de ferramentas de transformação dos dados durante a migração, desde que o projecto seja suficientemente grande para justificar a despesa.

Vai-se assistindo a um evoluir na abordagem. Torna-se mais holística, mais integradora, com a contemplação de matérias que até à data não eram consideradas. Aqui, os testes passam a assumir maior visibilidade, e sabemos hoje a real importância que têm em qualquer projecto de desenvolvimento e migração.

Já Kelly e Nelms, num passado muito recente (2003) abordam no seu artigo metodologias de auditoria dos dados de forma a assegurar a migração correcta dos mesmos. Assim, dizem que este processo de validação pode ocorrer das seguintes formas:

- Depois da migração dos dados.
- Durante o processo de migração.
- Através da revisão da abordagem metodológica da gestão de todo o processo.

A primeira opção obrigará à utilização de tempo adicional o que poderá ser um obstáculo em determinados ambientes de trabalho. A terceira opção pressupõe que a gestão do projecto segue uma metodologia determinada. A segunda, será por isso a mais aconselhada, ou mesmo, a mais utilizada. Assim, com a utilização desta última, a sugestão dos autores passa pela utilização de uma ferramenta como o Excel para a validação dos dados entre as BD origem e destino, tendo presente a sua limitação no

que concerne à utilização em BD de grande dimensão, apenas utilizável em situações específicas.

Relevam ainda situações determinantes no sucesso da migração dos dados como a utilização simultânea dos dois sistemas para a validação dos dados, a consequente determinação das diferenças verificadas entre ambos, a possibilidade de haver alteração nos dados durante o processo que terá que se propagar simultaneamente em ambos os sistemas e a fiabilidade e precisão indispensáveis na determinação dos dados a incluir na migração.

A finalizar concluem ainda que a migração efectiva dos dados para a nova BD deve ser efectuada no espaço de tempo mais curto para a entrada em funcionamento do novo sistema, ressalvando a existência de mecanismos de concorrência transaccional, de actualizações e de cópias de segurança bem como de restauro de sistema.

Todos estes estudos se complementam de alguma forma e abordam a temática da migração de bases de dados ou mesmo dos sistemas que as acolhem, dos quais são indissociáveis, com um vector constante de evolução quanto à solução a adoptar.

Por outro lado, e como podemos estar a considerar realidades bem diversas, será bastante fácil perceber que a migração de uma base de dados de um SGBD para outro diferente não poderá de forma alguma ser abordada da mesma forma que uma migração de um sistema empresarial para outro novo com reengenharia deste último e redesenho da base de dados com a inevitável transformação dos campos que a compõem. Também, paralelamente, não será possível realizar uma análise deste tipo, sem deixar de considerar o âmbito em que se realiza, de evidente mudança.

Existem determinados casos e características específicas da migração de BD, transversais a vários cenários de migração que pelas suas particularidades merecem um enfoque particular. A apresentação de outros tópicos relativos ao processo de migração segue também uma lógica sequencial.

1.1. O projecto

A migração de bases de dados necessitará de ter um plano. A definição do caminho a percorrer entre a origem, situação actual, e o destino, situação futura, desejável. A este plano chama-se de projecto.

O conceito de projecto está ligado a um conjunto de características específicas que hoje são genericamente aceites e que devem ser tidas em consideração na decisão de classificar um trabalho que tem que ser realizado como tal:

- Complexidade. A complexidade advém da dificuldade e da dimensão que, embora com pesos relativos que variam de projecto para projecto, estão sempre presentes. Os riscos envolvidos no projecto e o grau de inovação ou de utilização de tecnologias recentes ou não dominadas pela Organização, são factores que contribuem para um aumento da complexidade. Da mesma forma a estrutura organizacional e a dimensão dos clientes e fornecedores, bem como a natureza das obrigações contratuais, podem contribuir para a complexidade do projecto.

- Criador de mudança. Os projectos devem estar directamente relacionados com a implementação das estratégias da Organização, sendo os vectores fundamentais para a sua concretização. A mudança a que estão associados os projectos é, por natureza, uma mudança drástica, e pode ser, fundamentalmente, de dois tipos:

- Mudança do negócio (por exemplo, concepção e desenvolvimento de novos produtos ou novos serviços, alargamento do negócio para novos mercados);

- · Mudança da organização (por exemplo, reengenharia dos processos de negócio, modificação da concepção organizacional, implementação de novos métodos de trabalho).

- Objectivos específicos, com limitações de qualidade, custo e tempo. Os projectos são definidos com base num conjunto de objectivos (de âmbito, qualidade, prazos e custos) que são previamente definidos e que constituem a base de avaliação do sucesso do projecto.

- Singularidade. Os resultados a obter em cada projecto, bem como a forma dos os concretizar, são novos para a Organização, nunca anteriormente foram realizados e não voltarão a repetir-se.

- Envolvimento multifuncional. A diversidade de competências que contribuem para a realização de um projecto obriga à participação consertada de diferentes funções dentro da Organização ou mesmo ao recurso a capacidades exteriores, sempre que essas não existirem internamente.

- Transitoriedade das equipas de projecto. Os recursos que constituem as equipas de projecto são agregados em função dos objectivos do projecto e da natureza do trabalho a realizar, sendo a equipa desmembrada logo que o projecto é concluído, ou mesmo à medida que os objectivos vão sendo atingidos e os recursos podem ser dispensados. As características definidoras dos projectos, que acima foram apresentadas, são diferenciadoras das operações de rotina que são levadas a cabo pelas Organizações e que se caracterizam, fundamentalmente, pela repetição, não limitação temporal, introdução de mudanças evolutivas, equilíbrio e estabilidade dos recursos.

A gestão de projectos é actualmente entendida como um conjunto de processos de gestão que devem ser executados para que o projecto atinja a sua finalidade. O PMI (Project Management Institute), que estabelece as normas ANSI (American National Standards Institute) no domínio da gestão de projectos, considera um conjunto de nove áreas de conhecimento, que vão incorporar os diferentes processos de gestão. Estas áreas de conhecimento correspondem, em parte, aos objectivos do projecto, acima referidos. No quadro abaixo apresenta-se a matriz dos processos de gestão de projectos, mapeando os grupos de processos com as diferentes áreas de conhecimento.

Knowledge Areas	Project Management Process Groups				
	Initiating	Planning	Executing	Monitoring & Controlling	Closing
Project Management Integration	Develop Project Charter	Develop Project Management Plan	Direct and Manage Project Execution	Monitor and Control Project Work	Close Project
	Develop Prelliminary Project Scope Statement			Integrated Chenge Control	
Project Scope Management		Scope Planning		Scope Verification	
		Scope Definition		Scope Control	
		Create WBS			
Project Time Management		Activity Definition		Schedule Control	
		Activity Sequencing			
		Activity Resource Estimating			
		Activity Duration Estimating			
		Schedule Development			
Project Cost Management		Cost Estimating		Cost Control	
		Cost Budgeting			
Project Quality Management		Quality Planning	Perform Quality Assurance	Perform Quality Control	
Project Human Resource Management		Human Resource Planning	Acquire Project team	Manage Project Team	
			Develop Project Team		
Project Communications Management		Communications Planning	Information Distribution	Performance Reporting	
				Manage Stakeholders	
Project Risk Management		Risk Management Planning		Risk Monitoring and Control	
		Risk Identification			
		Qualitative Risk Analysis			
		Quantitative Risk Analysis			
		Risk Response Planning			
Project Procurement Management		Plan Purchases and Acquisitions	Request Seller Responses	Contract Administration	Contract Closure
		Plan Contracting	Select Sellers		

Tabela 1 - Mapeamento dos grupos de processo com as diferentes áreas de conhecimento

Fonte: PMBOK 2004

Na gestão dum projecto são considerados três níveis de execução dos processos de gestão:

Nível de Integração (ou de negócio) – é neste nível que é feita a integração do projecto nos objectivos de negócio da Organização. Neste nível são executadas as seguintes tarefas de planeamento:

- Definição da Finalidade do projecto
- Identificação das grandes áreas de trabalho (também designadas por fases do projecto)
- Identificação das categorias de recursos (apenas os tipos de recursos, sem identificação nominal desses recursos)
- Identificação dos constrangimentos do projecto (tempo, custos e qualidade)

- Avaliação dos riscos do projecto e dos pressupostos em que se baseou a análise dos riscos

O output deste nível é constituído pelos seguintes documentos:

- Documento de Definição do Projecto
- Especificação de Requisitos

Nível Estratégico (ou de Gestão) – é neste nível que o gestor de projecto exerce o controlo do projecto. As fases são desagregadas em pacotes de trabalho e são identificados os "milestones" do projecto, ou objectivos intermédios. São atribuídos os papéis e responsabilidades na execução do projecto. Neste nível são executadas as seguintes tarefas de planeamento:

- Identificação de pacotes de trabalho e "milestones"
- Atribuição de responsabilidades organizacionais na execução dos trabalhos

O output deste nível é constituído pelos seguintes documentos:

- Matrizes de responsabilidades (pacotes de trabalho)
- Especificações funcionais

Nível Táctico (ou operacional) – é neste nível que é planeado os componentes operacionais do projecto, isto é, os componentes em que vai ser realizado trabalho directo de criação de produtos ou de prestação de serviços. Neste nível são executadas as seguintes tarefas de planeamento:

- Definição das actividades componentes de cada pacote de trabalho
- Afectação de recursos (nominalmente) à execução das actividades

O output deste nível é constituído por:

- Planos de actividades

- Matrizes de responsabilidades (actividades)
- Concepção detalhada
- Execução dos trabalhos

Costuma dizer-se que a pior alternativa ao mau planeamento é a ausência do mesmo. Também aqui se aplica, pelo que se entendeu incluir esta alusão, se bem que ligeira. Podemos considerar que no que respeita às bases de dados uma das tarefas iniciais será certamente a construção da, ou das, bases de dados de destino.

1.2. Aspectos específicos de bases de dados

1.2.1. Modelação dos dados

O processo de modelação de dados vai ser utilizado para a construção das bases de dados de destino em ambientes de teste e de produção. Após a definição de requisitos, se estivermos no âmbito de um sistema relacional será efectuado o diagrama de entidades e relacionamentos, cuja notação foi proposta originalmente por Chen (1976) e é composta por entidades (rectângulos), relacionamentos (losangos), atributos (círculos) e linhas de conexão (linhas) que indicam a cardinalidade de uma entidade no relacionamento. Chen propõe também símbolos para entidades fracas e para entidades associativas. Estamos na fase conceptual.

Apresenta-se um destes diagramas como exemplo:

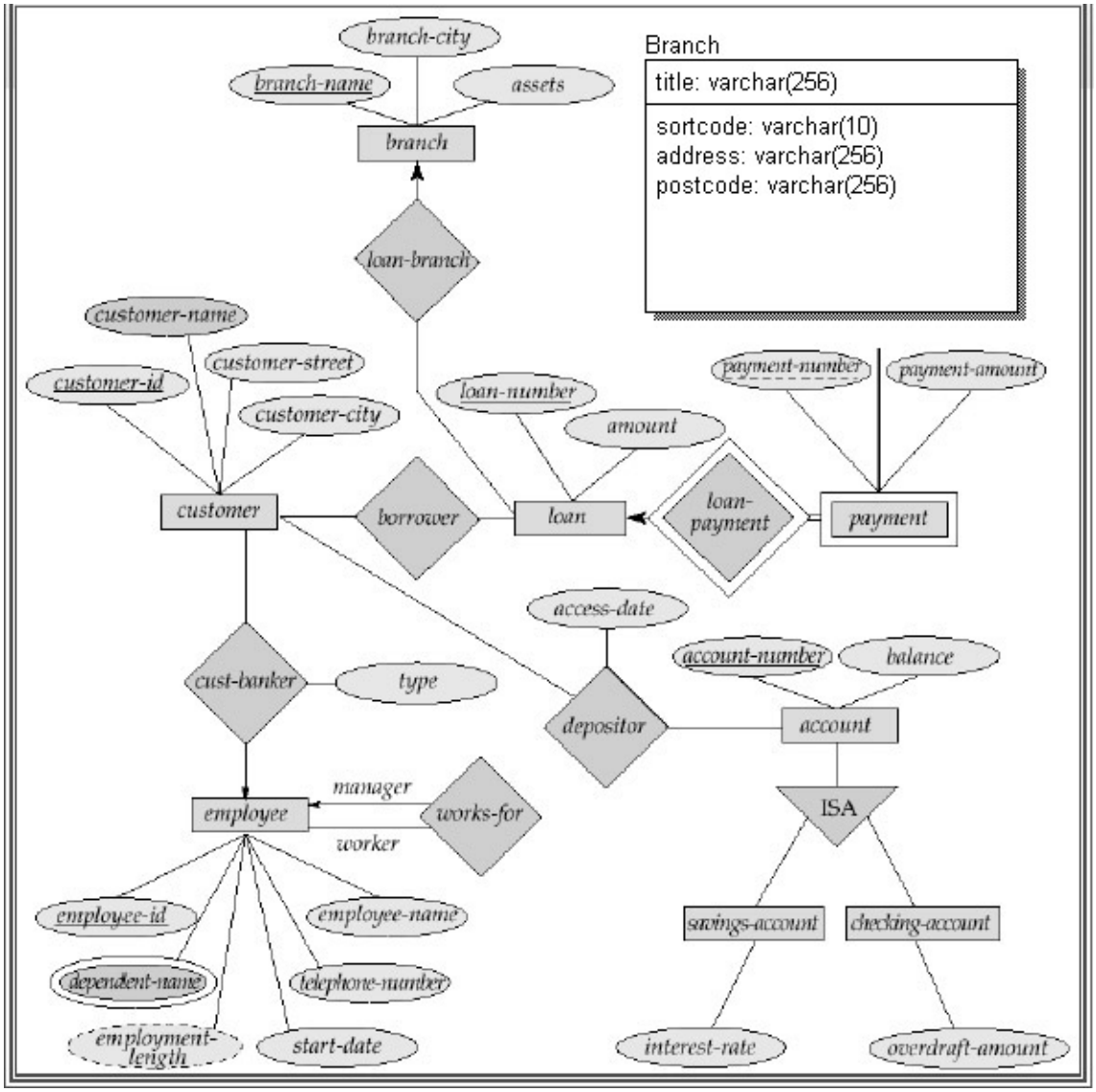

Figura 1 - Diagrama Entidade relacionamento

Fonte: http://www.15seconds.com/graphics/issue/030917_17.gif

Este diagrama é de seguida transposto para o modelo lógico da base de dados. Aqui, assegura-se que os atributos estão todos definidos, definem-se as chaves primárias e estrangeiras, são estabelecidas as cardinalidades dos relacionamentos e dá-se a normalização de todo o modelo. Esta modelação lógica tem outro tipo de representação como se pode observar na próxima página:

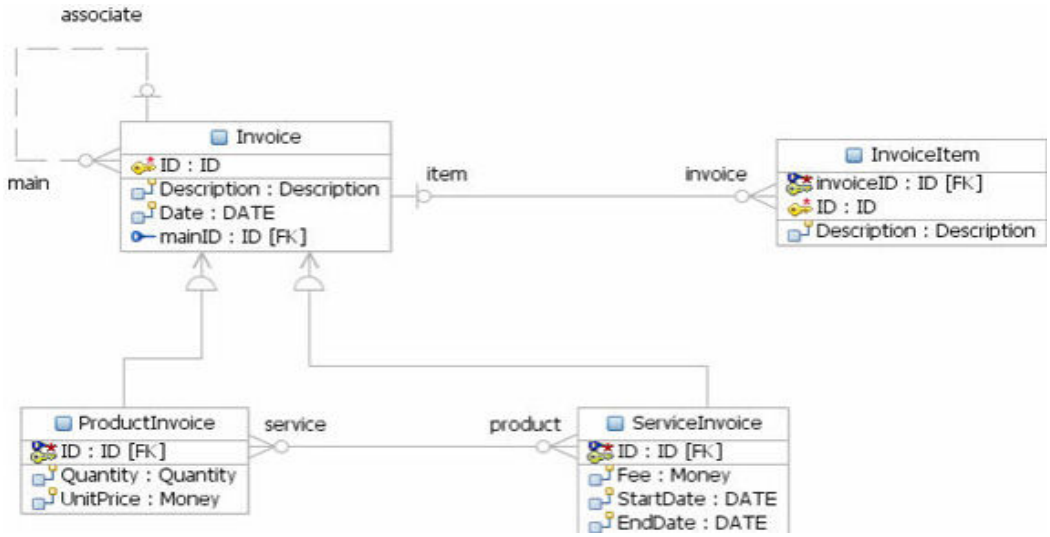

Figura 2 - Modelo lógico dos dados

Fonte: http://www.ibm.com/developerworks/db2/library/techarticle/dm-0708chang/

Após este último passamos, ao desenho do modelo físico dos dados. Neste atribuímos o tipo de dados a cada um dos atributos e preocupamo-nos com questões como o volume total dos dados, cardinalidades, particionamento e com os sistema de indexação, de "backup" e de recuperação em caso de falhas. Mostra-se um exemplo abaixo:

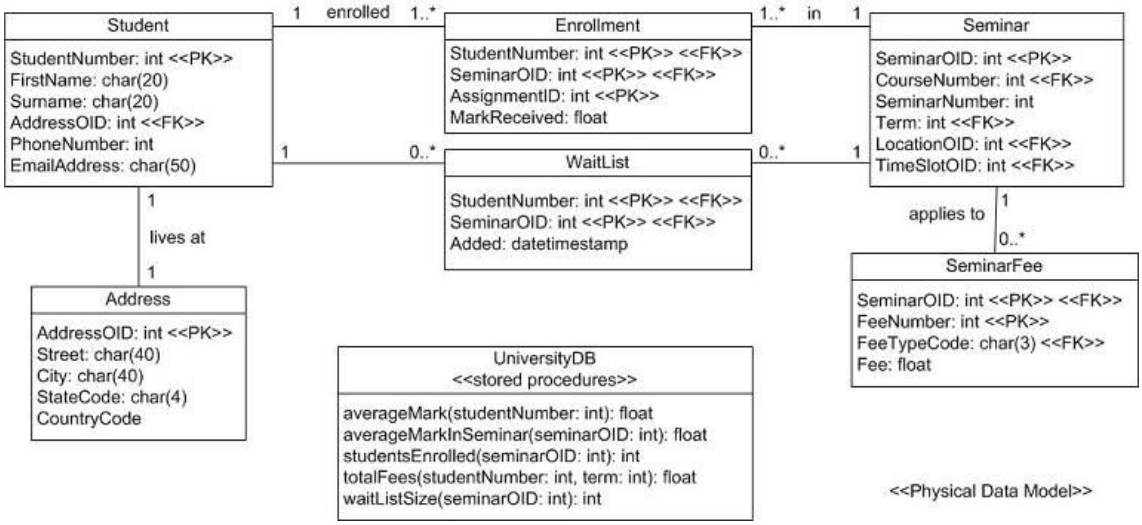

Figura 3 - Modelo físico dos dados

Fonte: http://www.agilemodeling.com/artifacts/physicalDataModel.htm

Por vezes é criado um centro de dados que congrega as extracções de dados efectuadas sobre múltiplos produtores antes de os carregar no ambiente de testes. Este centro, uma vez que representa um modelo comum e integrado dos dados, será o ponto de partida óptimo para proceder a tarefas de análise de sistema e de reengenharia.

De facto, esta tarefa pode ser referida como uma das inicias, mas quanto mais complexo for o sistema maior é a iteração das suas tarefas que se vão concluindo de forma transversal e gradativa. Estes dados são representados e acedidos através dos metadados da BD, visíveis através do seu esquema.

1.2.2. O esquema

No âmbito de um SGBD, será da sua responsabilidade estabelecer o "interface" entre os três níveis da sua arquitectura normalizada ANSI/SPARC[1], nível externo, nível conceptual e nível interno, fazendo os mapeamentos necessários. Para o efeito existe uma entidade designada por dicionário de dados que armazena a informação existente em cada nível e os mapeamentos entre as suas estruturas.

O nível interno refere-se ao armazenamento físico dos dados. Organização de ficheiros, métodos de acesso, etc. Deverá ser organizado de forma a beneficiar as operações que ocorram com maior frequência em caso de necessidade, ou, pelo contrário, proporcionar de forma transversal o melhor desempenho possível de todas as operações em geral.

O nível externo tem a ver com a visibilidade oferecida a cada utilizador da base de dados em causa, trabalhando apenas com uma parte da base dados em causa, com a parte do esquema onde estão os dados que lhe dizem respeito

Quanto ao nível conceptual diz-nos Pereira (1998: p.35) que: "Também conhecido por esquema conceptual. Neste nível representa-se o modelo conceptual de dados, independentemente de qualquer utilizador ou aplicação particular, constituindo o chamado esquema ou estrutura da base de dados. Esta é a camada que permite esconder

[1] Proposta apresentada numa tentativa de estabelecer um padrão para toda a indústria. Cada um dos três níveis descreve a base de dados numa perspectiva diferente de abstracção.

do nível aplicacional os detalhes de implementação física dos ficheiros que armazenam os dados".

O esquema é o conjunto das descrições que definem os tipos de dados e sua organização no SGBD. Poderão existir muitos ou poucos tipos de dados, dependendo da complexidade da base de dados. Mesmo nos SGBD mais simples há diferença entre dados numéricos e de texto, reflectida também na alocação de recursos ao seu armazenamento. Em bases de dados mais complexos, o esquema diferencia níveis de dados numéricos, além de identificar outras características dos dados.

Da leitura de Stern, compreende-se melhor a importância do esquema no âmbito de um projecto de migração de bases de dados. Na migração de uma base de dados, o esquema pode ser um grande desafio. É necessário mapear os tipos de dados do produtor para os tipos de dados do novo sistema. Não necessitam de ter o mesmo nome, mas devem ter a mesma função. Por exemplo, se não se proceder ao mapeamento dos tipos de dados correctamente, poderão ocorrer resultados imprevisíveis nas pesquisas posteriores. Pior ainda, os dados podem-se danificar ou mesmo perder.

Também compete ao esquema assegurar determinadas relações entre tabelas, conhecidas por integridade referencial. Esta última pode ser uma área problemática, essencialmente se permitir o acesso a utilizadores que não conheçam as implicações de não respeitar as restrições de integridade impostas.

Ainda hoje se efectuam migrações de sistemas antigos para novos sistemas em que na origem era permitido o uso simultâneo de vários tipos de dados definidos pelos utilizadores. Actualmente, esta utilização obriga sempre ao cumprimento de restrições e regras específicas. O mapeamento deste tipo de dados com aplicações antigas na origem exigem que o novo sistema ofereça compatibilidade binária.

A compatibilidade binária é a visão de que um registo é composto por uma série de "bytes", e a este nível é necessário garantir que os dados migrados sejam compatíveis com a aplicação. Assim, ao migrá-los, além de ser necessário verificar que os seus tipos são compatíveis em funcionalidade e capacidade, é também imperativo que tenham o mesmo formato.

Normalmente, o esquema criado no novo SGBD será idêntico ou muito parecido ao anterior. No entanto, este momento deve ser aproveitado para analisar se a reestruturação do esquema pode proporcionar o melhor desempenho ou quaisquer outros benefícios. Poderá incluir-se o uso de novos tipos de dados ou a normalização da base de dados, se o SGBD for relacional.

Na sequência do aparecimento do modelo relacional, e devido à necessidade de organizar os dados de forma a que possam ser tratados relacionalmente, surgiu um processo designado por normalização cujo objectivo é encontrar um esquema de base de dados relacional capaz de suportar adequadamente os dados relevantes a um determinado universo.

A este propósito Pereira (1998: pp 178-179) diz-nos que:" A normalização que, inicialmente, e decorrendo da própria definição do modelo relacional, se ficava pela desagregação dos dados em domínios atómicos(1ª forma normal), foi mais tarde completada por Codd (2ª e 3ª formas normais) quando este reconheceu a existência de algumas anomalias que poderiam surgir no decorrer da utilização de uma base de dados relacional "pouco normalizada".

Mais tarde, com base noutros contributos, surgiram outras formas normais, mais refinadas (*Boyce-Codd Normal Form*, 4ª e 5ª formas normais), resolvendo problemas mais específicos e menos frequentes, aperfeiçoando ainda mais o esquema relacional resultante."

A 1ª forma normal visa eliminar os grupos de valores repetidos que, eventualmente, possam existir em estruturas não normalizadas. Uma relação na 2ª forma normal é aquela em que, além de estar na 1ª forma normal, todos os atributos não pertencentes a qualquer chave candidata devem depender da totalidade da chave e não apenas de parte dela. A 3ª forma normal está assegurada quando, além de estar na 2ª formal normal, não existem dependências funcionais entre os atributos não chave, ou seja, cada atributo deve depender apenas da chave primária da relação. Quanto às BCNF, a 4ª forma normal está assegurada quando, além de estarmos na 3ª forma normal, não existirem

dependências multi-valor e estamos perante a 5ª forma normal quando não se pode decompor mais a relação sem haver perda de informação.

A situação ideal será a existência do modelo de dados do sistema de origem que se vai migrar, mas tal nem sempre acontece, podendo ainda acontecer que esteja desactualizado, ou mal documentado, e aí todo o trabalho estaria a ser efectuado com base num falso pressuposto com a inevitável consequência de um trabalho mal feito.

Houve investigadores que abordaram esta temática e desenvolveram metodologias para a extracção da estrutura lógica de uma BD a partir do respectivo esquema, do modelo hierárquico, em rede ou relacional para o modelo ER, com a utilização de vários passos intermédios, ou mesmo com mapeamento directo. É neste contexto que surge uma abordagem que advoga o uso de um formulário intermédio em que se procede ao mapeamento através da utilização de um modelo semântico.

Abu-Hamdeh et al. (1994), têm por base este estudo e propõem a automatização deste processo através da utilização prévia de um passo intermédio em que é criada uma lista de factos do primeiro esquema que é depois traduzida para o segundo esquema. Propõem a utilização do modelo ER como ponto intercalar, uma vez que este pode ser visto como um conjunto contendo os três modelos de dados tradicionais.

Ainda de acordo com os autores, não é neste mapeamento que reside a dificuldade, mas na inexistência de DDL para a definição de esquemas no modelo ER. Sugerem então que se utilize um sistema de tradução de fonte para fonte chamado de TXL e desenvolvido por Cordy, Halpern-Hamu, e Promislow (1991) para a implementação destas transformações. A sua função seria a de analisar o esquema de origem, transformá-lo e criar o esquema de destino.

Há actualmente, no entanto, ferramentas disponíveis, comerciais ou “open-source”, que efectuam esta tradução. O seu custo é variável chegando até às dezenas de milhar de euro, pelo que a natureza e complexidade de cada projecto ditará o recurso ao tipo de ferramenta escolhida para efeito de migração.

No início dos anos 90 é discutida a integração do esquema da BD nos seus aspectos de consolidação e manipulação estrutural com a abordagem de métodos mais eficientes de desenho do esquema como um todo. Este conceito de integração do esquema foi previamente estudado por Batini (1986) que sugeriu poder ser dividido em três fases: comparação de esquema, equivalência de esquema e fusão do mesmo.

A dificuldade na integração de esquemas releva do facto de dois esquemas diferentes modelarem a mesma realidade com diferenças a nível de terminologia, estruturas e enfoque. Pode dar-se o caso de dois objectos de nome diferente se referirem ao mesmo conceito, sendo sinónimos, ou ainda, dois objectos com o mesmo nome referirem-se a conceitos diferentes, sendo homónimos. Como exemplo poder-se-á referir também o caso da entidade pessoa, que por um lado pode ser caracterizada pelos atributos físicos, e por outro, pelos atributos sociais.

Dadas a complexidade e o período de tempo envolvidos será aconselhável que antes de tudo o mais seja efectuado um planeamento aos níveis de negócio e tecnológico. Este é mesmo imprescindível. Vai determinar a linha de orientação a seguir, objectivos, intermédios e finais, estimativas para a realização dos mesmos, e face a estes, permite assegurar que as tecnologias existentes são suficientes bem como em qualquer altura medir o afastamento do que foi inicialmente proposto.

1.2.3. Segurança

Outro dos factores críticos de sucesso na implementação de qualquer SGBD é a sua política de segurança. Poderá acontecer que no decurso do processo de migração as alterações estruturais resultantes impliquem modificações a nível do paradigma de segurança associado, havendo alterações no que respeita grupos, utilizadores ou mesmo de permissões.

Assumirão a maior relevância os perfis associados aos utilizadores e aos grupos, administração e autenticação de utilizadores, funções ou funcionalidades atribuídas aos utilizadores e ainda a administração da BD destino, integrada ou não no sistema que a suporta.

No que respeita a BD's acedidas via "web" poderá ser implementado um servidor com as regras de segurança que contém funcionalidades adicionais face às que são tradicionalmente implementadas numa BD. Com esta abordagem, o processo de autenticação não é efectuado directamente na BD; passa antes por este servidor, que procede às validações necessárias, e só depois é encaminhado para a BD.

Desta forma, é possível gerir de forma mais eficiente quebras de segurança como "denial of service" e o servidor passa a funcionar como gestor de restrições de acesso gradativo, o que significa que são atribuídos níveis de acesso aos objectos da BD, que são acedidos pelos utilizadores consoante o seu grau de autorização. Mecanismos de "user time-out", ou gestão de sessão, bem como a prevenção de "denial of service" através da monitorização de um número máximo pré definido de carga de transacções e consequente bloqueio do serviço, são também conseguidos através de acções sobre o servidor "web".

Se existirem, não um, mas vários servidores que implementam as regras de segurança, será também necessário assegurar, se não redefinir, a propagação das restrições de acesso por todos eles. Poderá ser o caso se no decorrer de um processo de migração estivermos a partir de um sistema produtor para vários sistemas consumidores. De seguida abordar-se-ão aspectos relativos ao processo de migração de bases de dados.

1.3. Aspectos específicos do processo de migração

Há a salientar que consoante o grau de complexidade da migração a realizar todas estas tarefas se poderão realizar ou não, chegando mesmo ao nível de terem que ser realizadas no decorrer de todo o projecto de forma iterativa à medida que se concluem determinados "milestones".

1.3.1. Avaliação da qualidade dos dados

Aqui, o enfoque estará na avaliação efectuada ao estado actual dos dados e das respectivas estruturas. Será medida a sua integridade, consistência, completude e validade. Como parte do processo, são identificados problemas ao nível dos atributos, das tabelas, e entre estas. È importante que esta avaliação comece a ser efectuada na

fase inicial do projecto de migração para reduzir o risco de falha. Decorrerá ainda até ao final do mesmo procedendo à monitorização constante da qualidade dos dados.

A avaliação da qualidade dos dados vai produzir resultados fundamentais para o processo de investigação decorrente do projecto em causa ao fornecer de forma detalhada o perfil dos dados envolvidos bem como os padrões que vão permitir a definição de regras de normalização.

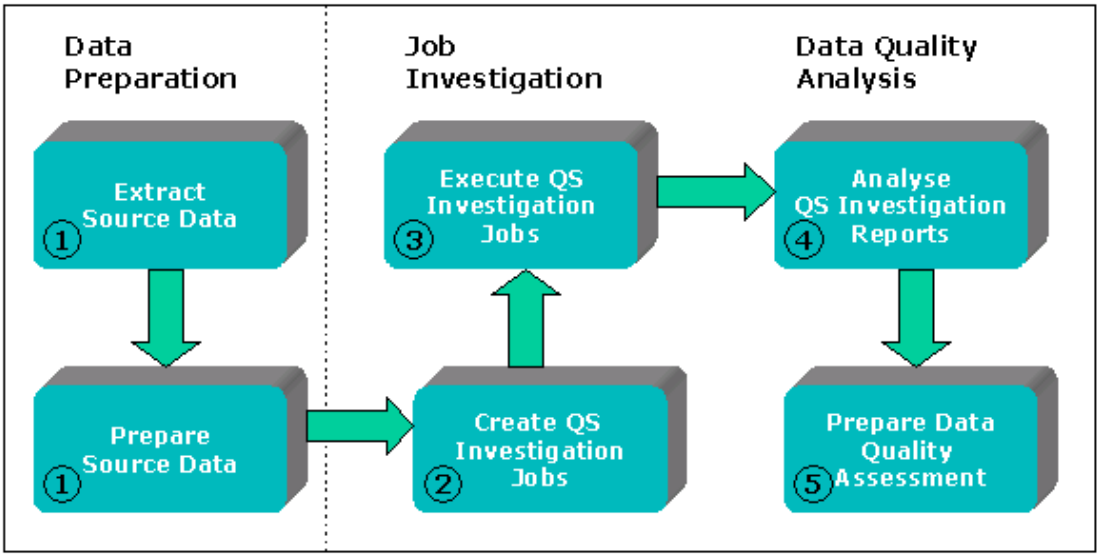

Figura 4 - Metodologia de investigação da qualidade dos dados

Fonte: http://mike2.openmethodology.org/wiki/Prepare_for_assessment_Deliverable_Template

A figura anterior representa uma metodologia de investigação da qualidade de dados utilizada em projectos de reengenharia de dados, que seguidamente se descreve.

1.3.2. Reengenharia dos dados

A reengenharia dos dados é efectuada para normalizar, corrigir, fazer corresponder, eliminar duplicações e melhorar os dados no sistema consumidor. Numa grande parte dos projectos de migração este processo é utilizado de alguma forma e pode mesmo significar grande parte do trabalhado realizado num cenário complexo.

O processo em si segue o princípio de Pareto, ou regra dos 80/20[2], com a iteração nos ciclos de desenvolvimento de software até que os dados atinjam o nível em que consigam proporcionar o maior valor para o negócio. A este respeito existe uma metodologia desenvolvida por Larry English em 1993, através da compilação de estudos e princípios previamente realizados por W. Edwards Deming's ("14 points of quality"), Masaki Imai ("Kaizen"), Joseph Juran ("Quality Planning"), Philip Crosby ("Quality is Free") e do Quality Function Deployment Institute[3] chamada de TIQM (Total Information Quality Management)4.

Todas estas alterações e redefinições são validadas, testadas, para aferir o seu comportamento face ao pretendido. É essa a função dos testes, que se passa a abordar.

1.3.3. Testes

Este processo poderá implicar o carregamento faseado de dados no novo sistema para testar as novas funcionalidades. Um caso de teste é constituído por um conjunto de dados de entrada, condições de execução de uma ou mais operações e resultados esperados ou dados de saída, desenvolvidos com um objectivo particular. O desenho dos casos de teste e a preparação dos dados de teste constituem actividades fundamentais do planeamento dos testes realizados.

---

[2] O princípio de Pareto foi criado no Séc. XIX por um economista italiano chamado Alfredo Pareto que, ao analisar a sociedade concluiu que grande parte da riqueza se encontrava nas mãos de um número demasiado reduzido de pessoas. Após concluir que este princípio estava válido em muitas áreas da vida quotidiana, estabeleceu o designado método de análise de Pareto, também conhecido como dos 20-80% e que significa que um pequeno número de causas (geralmente 20%) é responsável pela maioria dos problemas (geralmente 80%). Actualmente, serve de base aos Diagramas de Pareto, uma importante ferramenta de controlo da qualidade desenvolvida por Joseph Juran.

[3] http://www.qfdi.org/

[4] TIQM foi inicialmente chamado de TQdM (Total Quality data Management). Larry English estabeleceu a designação em 1993 quando começou a aplicar o trabalhão de Deming, Juran, Imai e de outros gurus da relacionados com a qualidade da informação. Em 2001, aproximadamente, apercebeu-se que os executivos preferiam a terminologia informação por associarem o termo dados a aspectos mais técnicos e alterou a designação para TIQM Quality System. Tem um site na internet onde disponibiliza serviços empresariais e recursos através de registo prévio em: http://www.information-quality.com/tiqmmethodology.cfm

Para o efeito são realizados vários tipos de testes: Funcionais, ou "Black box" e estruturais, ou "White box". Silva e Correia (2004) dizem-nos que de acordo com Sommerville (2000), "…a abordagem funcional por exemplo, é melhor aplicada sob componentes de sistema e realizada por uma equipa de testes, enquanto a abordagem estrutural é melhor aplicada a componentes individuais ou a colecções de componentes dependentes e realizada pela equipa de desenvolvimento."

Mostra-se abaixo um esquema com a representação de uma visão geral do processo de testes adaptado por Silva e Correia (2004) de Sommerville (2000).

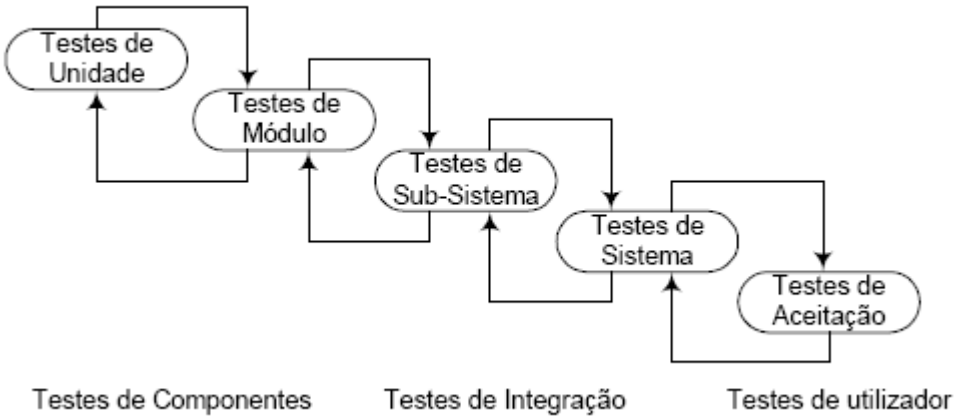

Figura 5 - Visão geral do processo de testes

Fonte: Silva e Correia (2004)

Através da abordagem de testes funcionais, os casos de testes derivam da especificação do sistema ou componente a ser testado. O sistema é visto como uma caixa fechada e o seu comportamento apenas pode ser derivado através do estudo dos possíveis valores de entrada e dos valores de saída relacionados, de acordo com as regras especificadas.

Os testes funcionais são aqueles em que se pretende determinar se a aplicação responde de acordo com as regras definidas nos requisitos de negócio. Neste cenário de migração é normalmente onde se concentra o maior número de casos de teste e o seu enfoque recai na validação das áreas apresentadas na página seguinte:

- Regras de negócio.
- Mapeamento dos dados.
- Chaves e restrições de integridade.
- Dados.
- Filtros.
- Reengenharia dos dados.
- Cálculos efectuados durante o processo.

Estes testes poderão ser realizados de forma automática ou manual. Dizem-nos Silva e Correia (2004) que: "A qualidade de um caso de teste é descrita através de quatro atributos. O primeiro consiste na capacidade de encontrar defeitos. O segundo refere a capacidade em exercitar mais de um aspecto reduzindo assim a quantidade de casos de teste requeridos. O terceiro e quarto fazem considerações de custo. O terceiro é inferido baseado no custo necessário para a realização do caso de teste incluindo o esforço de desenho, execução e análise dos resultados de teste. O quarto atributo refere o esforço de manutenção necessário sobre o caso de teste a cada alteração do sistema. Estes quatro atributos devem ser balanceados de forma a se ter casos de teste de boa qualidade."

Assim, a automatização de um caso de teste refere-se apenas ao factor económico em causa que por sua vez implica o esforço de manutenção necessário. O esforço de construção e de manutenção requerido para um teste automático é normalmente maior do que para um teste manual equivalente, mas uma vez construído, tende a ser mais económico que o teste manual. O esforço de execução e de verificação de resultados será uma pequena fracção do esforço de construção.

Visto que o custo original da implementação e o custo da manutenção de um caso de teste automático será diluído a cada execução que seja necessária, os testes de regressão são fortes candidatos a serem automatizados.

O teste de regressão consiste na aplicação de testes à versão mais recente do software para garantir que não surgiram novos defeitos em componentes já testados. Se ao juntar o novo componente, ou as suas alterações, aos restantes componentes do sistema, surgirem novos defeitos em componentes inalterados, então considera-se que o sistema regrediu.

Assim, os testes realizados, dependerão do tipo de suporte, da sua granularidade e da sua abordagem. Quanto ao suporte poderão ser manuais ou automáticos com recurso a "scripts", quanto à granularidade, de unidade, de módulo, de integração, de sistema ou de aceitação, e quanto à abordagem poderão der de caixa branca ou preta. Apresenta-se um esquema que demonstra de forma clara esta realidade:

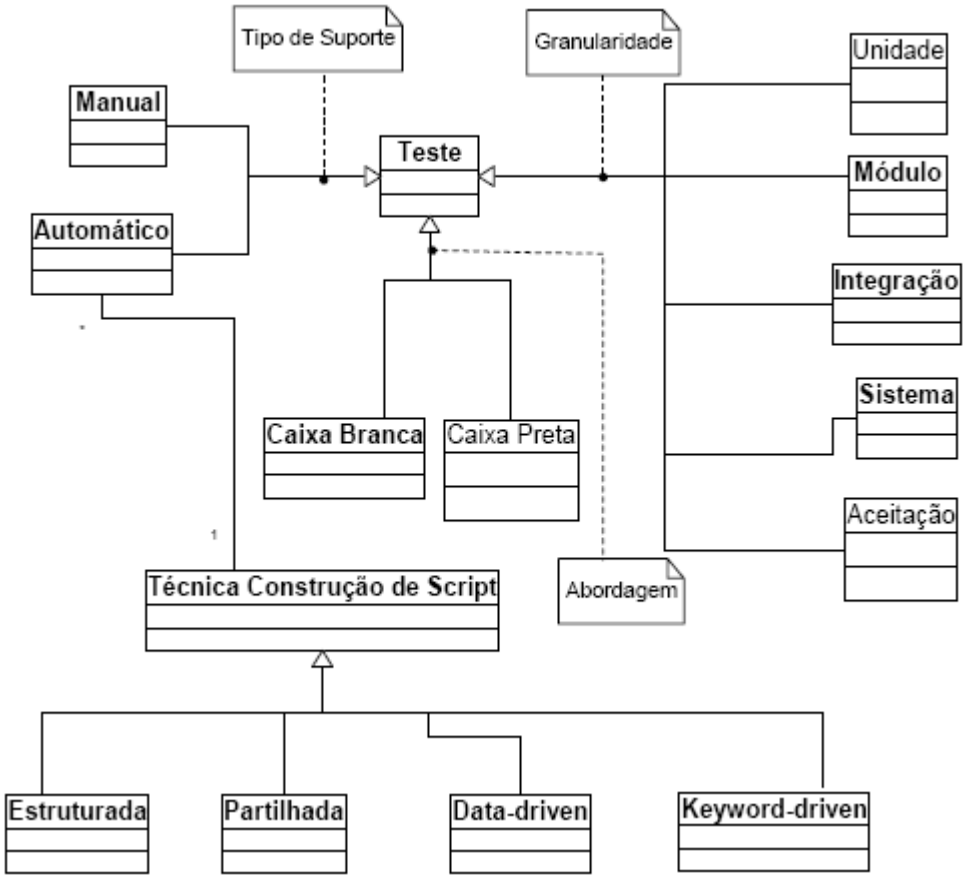

Figura 6 - Visão geral dos tipos de testes

Fonte: Silva e Correia (2004)

Após a fase de testes, os dados são carregados em produção no sistema de destino e poderá haver nova abordagem a processos de análise e melhoria da qualidade dos dados. Deverão ser testadas aqui as funcionalidades específicas de ambiente uma vez que só agora será possível realizá-las na íntegra. Uma vez efectuados todos os testes poder-se-á activar definitivamente o sistema, o que poderá implicar a carga faseada do mesmo.

Por fim, na conclusão do processo seguir-se-á a implementação e verificação final. Em cenários mais complexos de migração, faseados, a implementação acompanha todo o processo.

1.3.4. Implementação e verificação final

A implementação de "software" é, segundo Dearle (2007), o conjunto de processos que medeiam a aquisição e a execução do software. O implementador, de acordo com OMG(2003), Objects Management Group, executa o processo através da aquisição, preparação do mesmo para a execução, e possível execução do mesmo.

Diz-nos ainda Dearle (2007) que a implementação pode ser considerado como : "...a process consisting of a number of inter-related activities including the release of software at the end of the development cycle; the configuration of the software, the installation of software into the execution environment, and the activation of the software (Carzaniga et Al.). It also includes post installation activities including the monitoring, deactivation, updating, reconfiguration, adaptation, redeploying and undeploying of the software."

Acerca do citado, considera a "release", ou versão como o "interface" entre a equipe de desenvolvimento e os restantes actores envolvidos no ciclo de vida do software. A instalação como o processo de transferência do software para o cliente e envolve o processo de configuração, ou parametrização, para estar disponível para activação. A activação, diz-nos, é o processo de iniciar a execução do "software" através da colocação de "triggers" que o iniciam no momento pretendido, podendo ser de natureza distinta, como o recurso a "interfaces" gráficos, elementos de "script", ou através da utilização de processos "daemon".

Quanto aos processos que se verificam na fase de pós instalação, refere a desactivação como a oposição natural ao processo de activação e acontece com frequência o recurso à sua utilização, antes de se proceder à adaptação ou reconfiguração, nas ocasiões em que determinada parte do "software" tem que ficar inerte e renderizado sem invocação. O processo de actualização corresponde à alteração de uma parte do "software" instalado, normalmente despoletado pelo lançamento de uma nova versão pela equipe de desenvolvimento. É um caso especial de instalação e pode implicar a desactivação prévia do "software" instalado seguida de reactivação e reconfiguração. Ao contrário da actualização e reconfiguração, a adaptação é o processo de modificação o "software" instalado de forma a reagir às alterações no ambiente em que está instalado. Para finalizar, diz-nos que o "undeployment" é o processo de remoção do software implementado da máquina onde está. Também é conhecido como desinstalação.

Para a execução do processo de implementação são necessárias as seguintes especificações:

- Empacotamento do "software" e metadados associados para a transferência entre o produtor e o implementador.
- Recepção e configuração do "software" no ambiente do implementador antes de serem tomadas decisões de implementação.
- Descrição das características do sistema final de execução do "software".
- Planeamento do processo de implementação na infra-estrutura distribuída de destino.
- Preparação da aplicação para execução, ou seja, cópia dos elementos de "software" para o local físico de execução.
- Iniciar, monitorizar e concluir a execução do "software".

E com esta visão do processo de implementação, conclui-se a abordagem sequencial do processo de migração, em traços latos. De seguida particularizar-se-á o tema da migração atendendo à especificidade dos casos analisados.

1.4. Casos particulares de migração

Os casos a apresentar merecem particular destaque pois, por um lado iremos acompanhar uma das situações mais difíceis e complexas de realizar, e por outro, uma situação cada vez mais recorrente. O primeiro, é o da migração de sistemas legados e o segundo o da migração de bases de dados de tipologia diferente.

1.4.1. Migração de sistemas legados

Dos casos que apresentam maior grau de dificuldade será sem dúvida o da migração de bases de dados a funcionar no âmbito de sistemas legados. Ainda há muitos em funcionamento. Normalmente grandes e de extrema complexidade, a sua migração pode, e é normalmente, uma empreitada bastante dispendiosa no que respeita ao capital financeiro, humano e temporal necessários à mesma, pelo que as organizações têm sistematicamente procurado simplificar e reduzir os seus custos.

Nestes sistemas, o risco envolvido na opção de migrar é paradoxal; em caso afirmativo pode haver falhas no funcionamento do sistema que obriguem à sua paragem, mau funcionamento do mesmo com a ocorrência de resultados inesperados, perda de dados, ou mesmo a corrupção dos mesmos, entre outras. A hipótese contrária pode conduzir à desactualização do sistema, que é um processo permanente, ao crescente peso relativo do mesmo, perdendo performance, tornando-o ainda ingovernável.

No entanto, não se poderá abordar a migração de um sistema legado nos mesmos termos da migração de um SGBD. As camadas de apresentação, de lógica de negócio e de acesso a dados, que nas arquitecturas dos sistemas actuais estão logicamente separadas, nos sistemas legados existem misturadas numa única camada.

De acordo com Hasselbring et al. (2004), os sistemas legados não podem, pura e simplesmente, ser substituídos. Têm necessariamente que ser parte integrante do processo de migração. As suas justificações são:

- Os sistemas legados representam um investimento substancial que não pode ser facilmente descartado ou ignorado.

- É corrente a necessidade de manter o negócio durante todo o processo de migração. Desligar o sistema não é opção, e deve ser raro o caso em que os recursos do mesmo permitam uma grande sobrecarga.

- O "software" destes sistemas legados é frequentemente o único local em que há documentação de determinadas regras de negócio. Assim, é aqui onde os arquitectos e programadores do novo sistema recorrem para proceder à reengenharia do novo sistema.

Em alternativa à real migração do sistema existem soluções como o "wraping", também denominado de "screen scraping" no caso dos sistemas legados e consiste no desenvolvimento de um "interface" gráfico que melhora a usabilidade do sistema ao utilizador.

Em 1995 Brodie and Stonebraker propõem um método a aplicar na migração de sistemas legados a que chamam "Chicken Little Methodology". É uma estratégia de onze etapas, com recurso à implementação de uma série de "gateways" que estabelecem a comunicação entre os sistemas origem e destino, que correm em paralelo durante todo o processo. A implementação do sistema de destino é faseada. Começa por constituir uma pequena parte do de origem e aumenta progressivamente até o representar na totalidade.

Passados dois anos, Wu et al. (1997) questionam a utilização desta metodologia, nomeadamente no que respeita à necessidade de haver dois sistema a correr em paralelo, e apresentam uma alternativa a que chamam "Butterfly Methodology". A sua proposta é uma solução constituída por cinco fases:

- Determinação da semântica do sistema legado e desenvolvimento do esquema de destino.

- Construção de um sistema de dados para teste no sistema de destino baseado em dados representativos de lógica diversa na origem.

- Migração de todos os componentes, à excepção dos dados, para o sistema de destino.

- Migração gradual dos dados, e formação dos utilizadores sobre o novo sistema.

- Fecho do sistema legado e início da utilização do novo sistema.

Em suma, no que respeita à migração de sistemas legados, relevam-se aspectos como o planeamento cuidado, a utilização faseada de "gateways" ou adaptadores, correr aplicações em paralelo e desenvolver "interfaces" gráficas ao utilizador, acima referidas.

Estes sistemas têm também evoluído e arquitecturalmente são actualmente bastante mais complexos que no passado. Em alguns casos, como é por exemplo o da IBM, os seus "mainframe" mais recentes, da série "z" podem processar em simultâneo aplicações legadas e novas aplicações, como se pode verificar:

"O z10 EC também suporta uma ampla gama de aplicações e ambientes. Para além de Linux, XML, Java, Websphere e aplicações relacionadas com Service Oriented Architecture (SOA), a IBM está a trabalhar com a Sun Microsystems e a Sine Nomine Associates para criar um piloto do sistema operativo Open Solaris em System z, demonstrando o espírito de abertura e flexibilidade do mainframe." (IBM Notícias)

Técnicas avançadas de virtualização, segurança e a existência de cinco tipos de processadores diferentes destinados especificamente a determinado tipo de programas, como o ZIIP para DB2, ZAAP para Java, e outros são ainda algumas das características que fazem com que actualmente exista um mercado intermédio de grandes clientes que podem optar por soluções arquitecturais distribuídas ou pela consolidação em "novas" soluções centralizadas.

1.4.2. Migração de tipos de bases de dados diferentes

É exactamente neste cenário de constante evolução que surgem as migrações de bases de dados de tipologia diferente. Também a nível lógico, a substituição de uma abordagem estruturada a nível de programação para uma outra, essencial, mais próxima

de qualquer realidade objectiva mais complexa, orientada a objectos, criou a necessidade de tipologias diferentes de implementação lógica dos dados.

Surgiram também, consequentemente, SGBD "Object Oriented", doravante designados por SGBDOO. A justificação para tal tem que ver com a equivalência resultante entre as estruturas de dados a nível da aplicação que se pretendem armazenar de forma permanente e aquela que de facto é armazenada. É assim possível armazenar dados de complexidade bastante superior e com ganhos de performance.

Como consequência, passou a a haver casos concretos em que a substituição de um modelo por outro implicava ganhos imprescindíveis. De acordo com Behm et. Al.(1997), existem três formas diferentes de abordar esta questão:

- A construção de um "interface" orientado a objectos que se liga ao SGBD relacional.
- A migração para um sistema de base de dados Objecto-Relacional, Stonebraker(1995).
- A conversão do esquema relacional e respectivos dados para um modelo orientado a objectos.

A primeira abordagem é a mais fácil, e já existem inclusivamente bibliotecas de classes para este efeito. O sistema não é parado para efeitos de migração e não há perda de informação semântica. Por outro lado, existe uma perda de performance devida à necessidade de conversão dos dados sempre que há comunicação entre a aplicação e o SGBD.

Na segunda hipótese, os dados têm que ser efectivamente migrados para um SGBD que suporte a implementação de características e recursos orientados a objectos através da extensão ou modificação do esquema da BD em causa. Por fim, no terceiro caso, procede-se à migração integral da base de dados e respectivo conteúdo para um SGBDOO com a criação de um novo esquema.

Qualquer das escolhas implicará sempre uma situação de compromisso que deverá ser adequada ao problema particular, em causa, a resolver, e enquadram-se inevitavelmente neste cenário de migrações complexas. De acordo com Behm et al.(1997), "Each of the three possibilities has advantages and disadvantages. Especially in the third one, an underlying method and tool support are critical for successful migration".

De seguida apresenta-se uma abordagem, em que são pré-definidos três cenários possíveis de migração, inserida num projecto "open source" de maior amplitude com o objectivo de aperfeiçoar técnicas de gestão de informação empresarial denominado de MIKE2.0 e acedido em MIKE2.0.

2. Cenários e metodologias de Migração

Na maioria das grandes organizações a migração de aplicações empresariais é bastante complexa. Torna-se prioritário determinar bem a abrangência ou raio de acção do problema, o seu âmbito e delimitá-lo. Depois, sobre este, é necessário medir a complexidade da migração em causa e quais as actividades necessárias para a sua realização. Aqui, diz-nos Steve Callan (2006) que "Here is something else to consider: break a dependency chain before it breaks you and the migration process."

Independentemente da solução efectiva a adoptar, quer a nível de arquitectura quer de tecnologia são propostos três cenários; uma migração leve ou ligeira, uma migração média, e uma outra, pesada ou complexa.

Para que se possa compreender melhor o que as diferencia apresenta-se um quadro com as tarefas necessárias a cada uma delas.

Capability Required	**Lite Scenario**	**Medium Scenario**	**Heavy Scenario**
Data Profiling	Direct Copy of Sources	Key Integrity validated	Referential Integrity required
Data Replication	None	None	Multiple Targets
Data Transfer	To Target	To Target	Target / Downstream
Data Synchronisation	None	None	For Interfaces
Data Transformation	Modest	Significant to similar structures	Major Activity
Data Mapping	Minimal	SME supported	Major Activity
Data Standardisation	None	Key Attributes	All attributes
Pattern Analysis and Parsing	None	None	Yes
Record Matching	None	Based on similar IDs	IDs and pattern matching
Record De-Duping	None	None	Yes
Out of Box Business Rules	As Appropriate	As Appropriate	As Appropriate
Configure Complex Rules	None	Application	Application/Infrastructure
Out of the Box Interfaces	As appropriate	As appropriate	As appropriate
Configure Custom Interfaces	None	Application	Application/Infrastructure

Data Governance Process Model	Documented in high level form	Key or Lynchpin Processes modeled	End to End Models
DB Independent Functions	As Existed at Source	Few Custom APIs	Infrastructure Services
Data Management Reporting	Data Move Metrics only	DQ and DM metrics	Reporting as a Service
Active Metadata Repository	Specific and 'physical'	Multiple Passive dictionaries	Initial implementation

Tabela 2 - Recursos necessários à migração

http://mike2.openmethodology.org/wiki/Data_Migration_Solution_Offering

2.1. Migração ligeira

A migração ligeira é comparativamente fácil e rapidamente se alcança grande objectividade. Implica o carregamento de dados de uma fonte única para um destino único. Assumindo que os seus dados estão "limpos", são necessárias poucas melhorias a nível da qualidade dos dados. O mapeamento é relativamente simples bem como as funcionalidades aplicacionais implementadas. A integração dos dados estará na base do sistema, no seu "back-end" e a migração será efectuada de uma só vez, numa técnica apelidada de "big bang".

Este tipo de migração apenas obriga à cópia dos dados e a alterações menores a nível dos processos. As estruturas de dados no produtor e consumidor são semelhantes e apenas são necessárias transformações simples. A extracção dos dados pode ser melhorada significativamente com a utilização de ferramentas de extracção, transformação e carregamento de dados, designadas de ETL, já existentes, comerciais, ou não. É efectuada ainda uma estimativa sobre a qualidade dos dados para efeitos de decisão. A quantidade de planeamento táctico e estratégico, com maior enfoque neste último, é em geral bastante diminuto numa migração deste tipo apesar de ser possível que o trabalho em particular esteja contido num projecto de maior envergadura.

Mostra-se de seguida a Fig.7 com o esquema gráfico de uma destas migrações. Procede-se à extracção dos dados do estádio original (1), carregam-se os dados no ambiente de destino (2), executam-se acções eventualmente necessárias de "limpeza" de dados, como o tratamento de valores nulos, a eliminação de registos duplicados ou a normalização de dados transversais ao sistema, por exemplo, num ambiente de pré-

produção, ou de testes (3), procede-se à passagem do sistema para o ambiente de produção (4) e activa-se o sistema após a verificação da total integridade do mesmo (5).

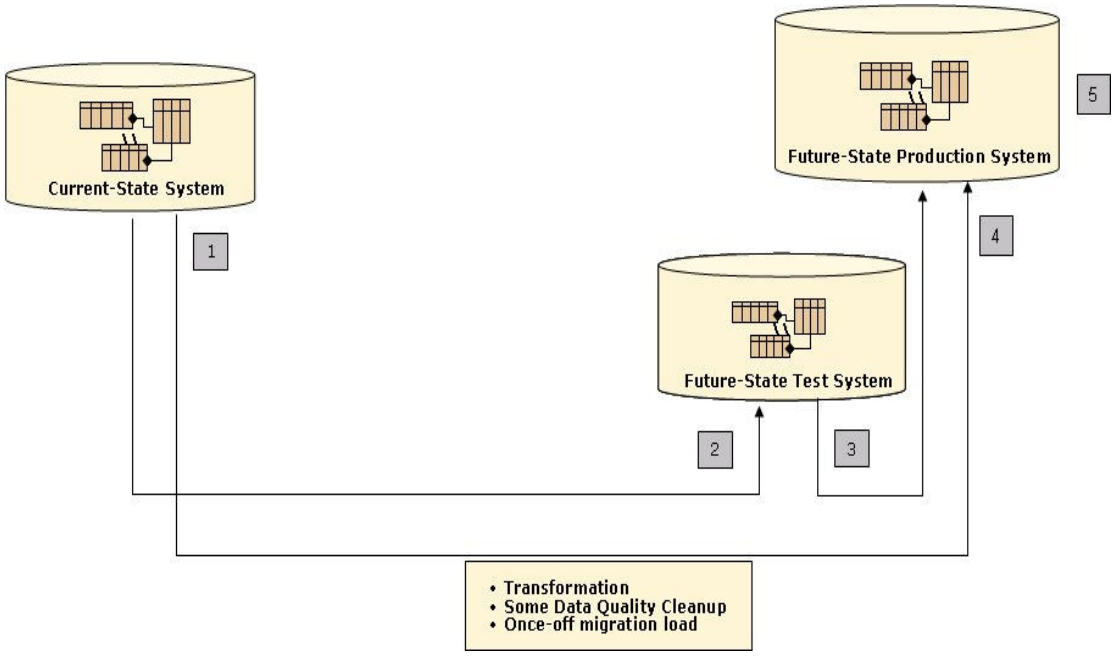

Figura 7 - Cenário de migração ligeira

http://mike2.openmethodology.org/wiki/Data_Migration_Solution_Offering

Uma das actividades primárias deste tipo de migração é a definição detalhada dos requisitos de negócio. Do ponto de vista do desenvolvimento do sistema de informação, traduz-se no enfoque que a abordagem tem na relação entre os requisitos de negócio anteriores e actuais no que respeita aos dados que têm que ser extraídos do produtor.

2.1.1. Actividades chave no processo

As actividades principais deste processo compreendem:

- A extracção dos dados da fonte para uma área de teste do sistema, que pode ser uma base de dados ou um sistema de ficheiros. Apesar de haver apenas uma fonte na origem dos dados, poderá haver a adição de mais dados em caso de necessidade por parte dos requisitos funcionais do novo sistema. Apesar de simples, a técnica de "Data Profiling", que consiste na análise dos dados e

respectivas estruturas, actuais e futuros, poderá revelar-se da maior importância na determinação da qualidade dos dados com impacto implícito no desempenho da aplicação. Esta, acompanha todo o projecto.

- O ambiente de testes de destino implementa as mesmas funcionalidades do sistema final de produção. Neste, de testes, são efectuadas as mesmas operações que de futuro serão realizadas quando da passagem ao sistema de produção. O desenho lógico a nível de ETL, o desenho físico e a construção da base de dados são relativamente simples neste tipo de migração.

- Os testes são realizados no respectivo ambiente. Este processo pode implicar o carregamento faseado dos dados no novo sistema para testar novas funcionalidades aplicacionais, entretanto desenvolvidas. Testes funcionais, "End-to-End Testing" e testes de utilizador final ("User acceptance testing - UAT") são actividades chave a desenvolver; Os testes de integração de sistema e de carga e desempenho, o "Stress and Volume Testing (SVT)", este ultimo de uma perspectiva de migração, não são de todo necessários neste cenário, na assumpção de baixo volume de dados.

- A passagem a produção pode ter a sua origem no ambiente de produção actual ou no ambiente de testes, mas normalmente é mais fácil que ocorra com base neste último, i.e., após efectuados todos os testes de verificação e validado o funcionamento expectável o sistema cumpre os requisitos que modelaram, logo, está pronto a ser copiado para o ambiente de produção. Em alternativa, seriam novamente aplicadas as transformações desejadas ao sistema de origem e copiava-se directamente para produção. Uma vez que estas migrações são bastante mais simples, será mais adequada uma passagem única para o novo ambiente de produção.

- Os dados são carregados no novo sistema de produção e nesta fase poderão ainda ocorrer mais verificações quanto à qualidade dos dados. Finalmente temos os testes de verificação de ambiente de produção, normalmente realizados pela equipe de desenvolvimento e deverá incluir testes funcionais de características

específicas do ambiente. Após a realização destes, o sistema é activado e fica a funcionar.

Para este tipo de iniciativas é frequente que a arquitectura do sistema de informação em causa forneça os vectores de alinhamento dos conceitos técnico e desenho conceptual, e será ideal se for complementado por um planeamento pré definido de actividades sequenciais aos níveis de implementação do negócio e da tecnologia pretendidas.

A solução arquitectural abrange o desenho conceptual de todos os componentes importantes no sistema, inclusive a infra-estrutura e o processo do ciclo de vida do desenvolvimento do "software". Só a correcta tradução dos requisitos de negócio ao nível aplicacional para a solução tecnológica irá possibilitar que se verifique a implementação bem sucedida dos requisitos de dados na nova aplicação.

2.2. Migração Média

O cenário de migração média implica a consolidação de um número não muito grande de produtores numa estrutura de consumidores. Será bem entendida se não houver um grande número de racionalização e estandardização de atributos. As extracções do sistema fonte podem ser directas ou para uma imagem para efeito de avaliação da qualidade dos dados.

As migrações médias podem ser efectuadas através da movimentação única de todos os dados. As estratégias de coexistência aplicacional entre os sistema de origem e destino raramente são necessários apesar de eventualmente poder vir a ser necessário vários fluxos de dados para proceder à migração total de todos os dados. Face a esta complexidade, são pré-estabelecidas versões face aos objectivos intermédios pré-determinados. O trabalho relativo à qualidade dos dados poderá ser efectuado após cada um destes fluxos.

O mapeamento dos dados do produtor para o consumidor poderá ser complexo; Deverão existir especialistas nesta matéria para acompanhar o mapeamento e a criação de casos de teste. Os sistemas poderão ser descontinuados de forma iterativa à medida que os processos são migrados para o sistema de destino.

O recurso a modelos para a representação lógica das bases de dados é essencial. É através destes que se consegue compreender a complexidade dos sistemas em causa. Os arquitectos e programadores envolvidos numa migração de BD's necessitam de trabalhar com entidades e relacionamentos, modelos lógicos e físicos, esquemas de origem e de destino.

2.2.1. Actividades chave no processo

Mostra-se de seguida um esquema deste tipo de migração:

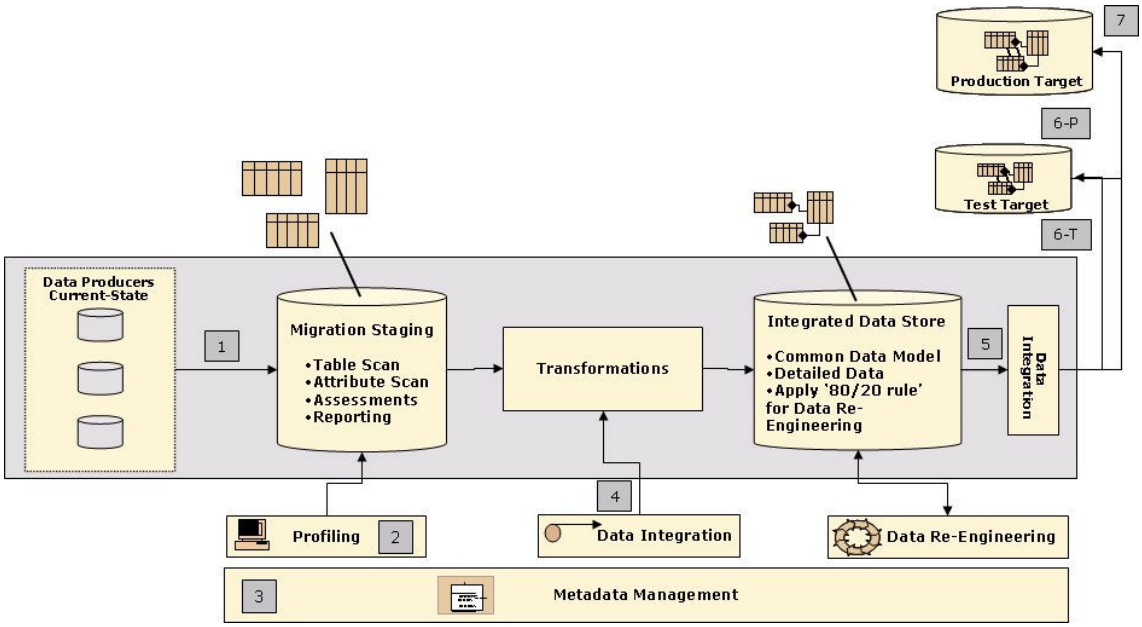

Figura 8 - Cenário de migração média

http://mike2.openmethodology.org/wiki/Data_Migration_Solution_Offering

As actividades principais deste processo compreendem:

- A extracção dos dados dos produtores para a área de testes.
- Os dados existentes na área de testes serão analisados para calcular colunas ao longo das linhas e entre tabelas, numa primeira abordagem ao nível de qualidade dos dados. O resultado vai ser utilizado para determinar quais as regras de negócio e as transformações necessárias na fase inicial do projecto.

- Nesta fase começam a estabelecer-se as regras de mapeamento sobre o metadados. A normalização será estabelecida e utilizada na preparação da extracção dos dados. Todos os atributos na origem serão mapeados para os do destino no ambiente de gestão dos metadados.

- São implementadas as transformações necessárias e a normalização definida, necessárias à extracção dos dados para a área de testes para que estes sejam realizados. Os dados são movidos para a área de dados integrada de testes.

- É efectuada nova análise à qualidade dos dados e é novamente medido o hiato entre o pretendido e o existente. São efectuadas novas acções de normalização dos dados para que estejam consistentes com os objectivos estabelecidos. Após esta, as regras para os registos que não podem ou não devem ser movidos são aplicadas. Neste passo em particular deve realizar-se uma análise exaustiva.

- Este passo refere-se à real extracção, transformação e carregamento de dados, num dos dois ambientes finais, ou o de testes, ou o de produção. Os processos e os recursos utilizados devem ser os mesmos, independentemente do ambiente utilizado.

- Os dados são carregados no ambiente de produção onde poderá ser necessária, análise e ajustes adicionais à qualidade dos mesmos. Os testes de verificação de produção são realizados, onde se incluem também testes funcionais específicos de ambiente. Após a conclusão dos testes o sistema é activado em sistema de produção a funcionar em tempo real.

O armazenamento da informação num repositório de metadados durante o processo, possibilitará a existência de recursos que permitirão a integração e gestão dos dados em situações futuras.

2.2.2. Outras actividades relevantes

As actividades apresentadas até à data não são as únicas a ocorrer durante o processo. Em simultâneo, antes, ou depois, as supracitadas são complementadas, quer pelas que se

abordaram no primeiro capítulo, quer pelas que passo a enunciar, utilizadas sempre que o grau de complexidade o justifique.

2.2.2.1. Gestão de meta dados

Dadas a complexidade da migração e a quantidade de acções realizadas, haverá muito provavelmente um grande número de transformações efectuadas sobre os meta dados, em consequência da definição de dados, da aplicação de regras de negócio, implementação de regras de transformação dos campos, e do esforço para melhorar a qualidade dos dados.

Acontece que toda esta informação deverá ser armazenada num repositório de meta dados, prevendo-se a futura utilização da mesma, que eventualmente acabará por acontecer; os sistemas evoluem. A criação e utilização deste repositório devem ser efectuados numa fase inicial do projecto.

Para o efeito são apresentados os requisitos necessários ao desenvolvimento de um modelo de gestão de meta dados, visível na seguinte figura:

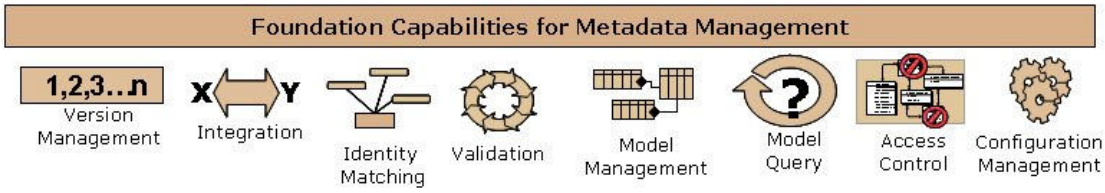

Figura 9 - Requisitos necessários ao desenvolvimento de um modelo de gestão de meta dados

Fonte: http://mike2.openmethodology.org/wiki/Metadata_Management_Foundation_Capabilities_Component

O "Model management", gestor do modelo, é o recurso que possibilita a gestão das estruturas e processos utilizados para descrever os meta dados no sistema. É da sua competência a definição do domínio e profundidade dos dados existentes neste modelo. E isto porque nem todos os metadados são passíveis de serem armazenados e geridos em qualquer altura, dada a enorme e constante produção informação de algumas organizações de forma continuada.

A “Metadata integration”, integração de metadados, permite que existam relações entre estes. Isto é, são produzidos e consumidos por uma série de componentes da arquitectura de informação. Para atingir a necessárias consistência, qualidade e reutilização, é necessária a integração dos registos na fonte dos mesmos. È esta integração que possibilita uma gestão activa dos meta dados.

O “Identity matching”, identificador de equivalências, é um recurso simultaneamente estratégico e técnico que assegura a consistência, reutilização, validação e catalogação de versão, possibilitando assim a identificação de forma inequívoca dos metadados.

A “Validation”, validação, é responsável por assegurar que o fluxo de metadados tem a qualidade e a consistência necessárias para representar de forma fiável e completa a respectiva estrutura de dados.

A parte de “Versioning”, controle de versões, assegura o histórico de anteriores versões de metadados. A granularidade das versões irá depender das necessidades da organização, mas o sistema deverá providenciar este recurso básico.

O “Configuration management”, gestor de configuração, é um processo fundamental para o desenvolvimento de metadados. À semelhança de outros activos, também estes têm um ciclo de vida de desenvolvimento. Competirá a este o controle e gestão das operações efectuadas neste ambiente de gestão de meta dados, assegurando para o efeito a disponibilização e afectação dos recursos humanos e tecnológicos necessários ao seu bom funcionamento actual contemplando ainda a agilização do mesmo de forma a estar preparado para qualquer evolução futura.

É através do “Model Query”, modelo de pesquisas, que se acedem aos metadados e é também este que permite o desenvolvimento de aplicações orientadas a serviços. Enquanto que a integração de metadados proporciona serviços de programação de baixo nível que permitem a integração de registos de meta dados, este implementa uma camada de abstracção de alto nível que possibilita o acesso facilitado aos mesmos. É responsável por exemplo pelos relatórios produzidos.

Por fim, o "Access control", controle de acesso, é um recurso que estabelece de forma transversal uma camada de controle sobre os modelos de metadados. É um recurso fundamental para proteger dados restritos a determinados utilizadores e está integrado no mecanismo de acesso aos metadados através do modelo de pesquisa, interfaces de integração, serviços de metadados e conjuntos de relatórios. Serve ainda de complemento às restrições de segurança implementadas a nível do sistema e de acesso a dados.

2.2.2.2. ETL, desenho e desenvolvimento

O processo de integração dos dados neste tipo de cenário de migração média costuma ser suficientemente complexo para obrigar a um processo de desenho que cubra os três níveis, conceptual, lógico e físico.

Nesta fase, vai-se proceder à extracção, transformação, e carregamento dos dados de forma integrada no destino. Esta, pode ser efectuada de forma mais ou menos automática, e também de forma mais ou menos faseada consoante a sua complexidade. Poderão, e até é natural que se criem, tabelas específicas de apoio à migração, eliminadas após concretização desta.

Remeto este tema para uma fase posterior do presente trabalho, em que o irei abordar, tanto do ponto de vista teórico como prático, com a apresentação de um caso de migração de uma base de dados de uma origem para um destino em que haverá um processo de redesenho da mesma.

Fica de seguida um esquema em que se mostra o enquadramento do desenho conceptual no ciclo de vida do desenvolvimento do "software". Como se pode observar corre em paralelo com outras actividades, e é bom que assim seja, pois deve procurar-se a atomicidade no desenvolvimento e a interligação dos módulos via "interfaces".

Este desenho conceptual precede o lógico que por sua vez irá ser determinante na construção do físico.

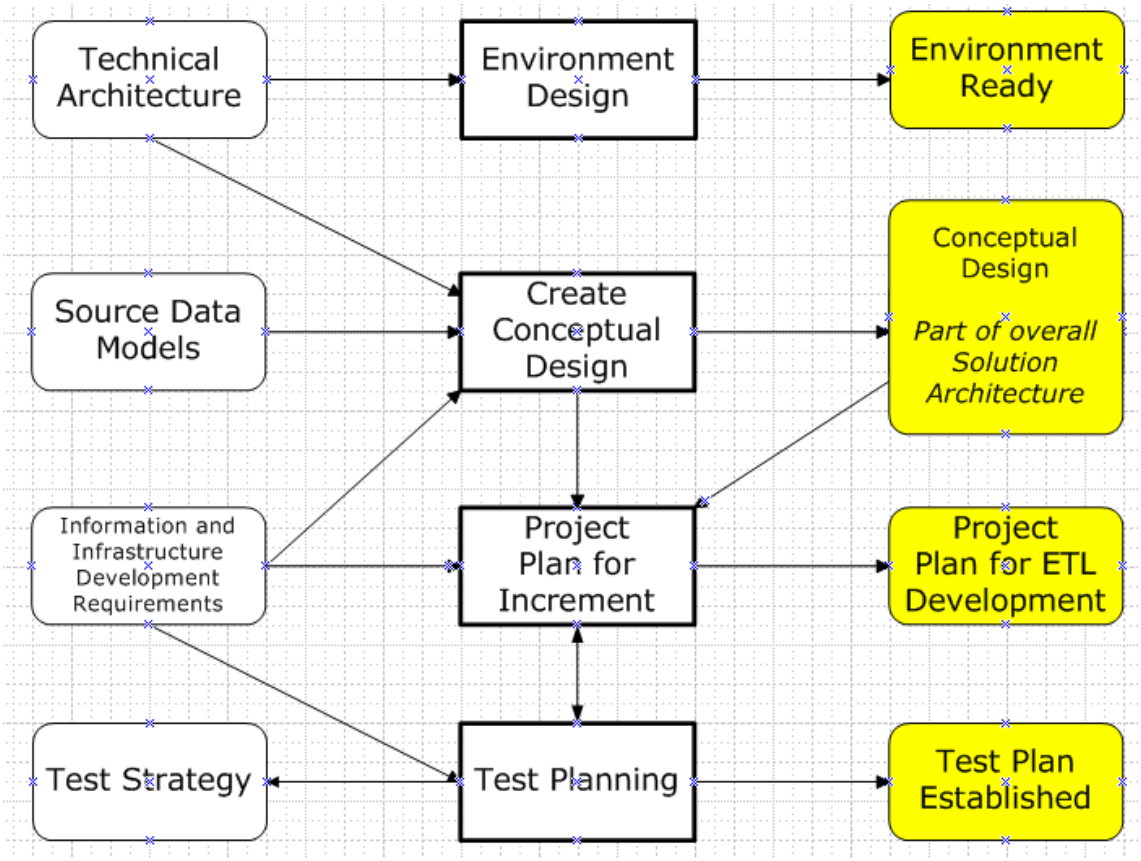

Figura 10 - Enquadramento do desenho conceptual no ciclo de vida do desenvolvimento de "software"

Fonte: http://mike2.openmethodology.org/wiki/ETL_Conceptual_Design_Deliverable_Template

2.3. Migração complexa

Na perspectiva de Stern, uma migração complexa, a que chamam de nível três, é bastante diferente das anteriores, na medida em que, se nas outras o objectivo da migração era alterar o sistema de armazenamento de dados, numa migração deste tipo, o sistema ou as aplicações que acedem aos dados também se alteram. Na sua óptica existem dois objectivos subjacentes, o primeiro em que é alterado a estrutura dos dados, o segundo, a alteração da camada de acesso aos dados. É também referida a necessidade de maior investimento e o acréscimo de risco envolvido.

Ora, se mesmo numa arquitectura relativamente simples o processo decorrente deste tipo de cenário é alvo de um aumento exponencial da complexidade envolvida, se nos focarmos no domínio de arquitecturas mais complexas e sofisticadas é imediata a

percepção da necessidade de um bom planeamento. Em MIKE2.0, é exactamente este o domínio das aplicações complexas, e a sua abordagem foca o tipo de preocupações e de necessidades características deste cenário.

De acordo com os autores, a migração complexa carece de uma solução para a co-existência aplicacional que permita a execução paralela de vários sistemas em simultâneo, através da implementação de uma estrutura de integração. Atendendo a que o seu prazo de execução é o médio, de três a cinco anos, e exigem um grande esforço de integração de dados, é da maior utilidade a construção de um sistema paralelo de análise que proceda à extracção de uma visão vertical da informação.

Vai ser necessária uma estratégia integrada do projecto que inclua as pessoas, o processo, a organização e a tecnologia. Envolverá múltiplas alterações tecnológicas e um grande número de "stakeholders"[5].

[5] O termo "Stakeholder" foi utilizado pela primeira vez por R. Edward Freeman no livro: "Strategic Management: A Stakeholder Approach", (Pitman, 1984), para referir-se àqueles que possam afectar ou são afectados pelas atividades de uma empresa. Estes grupos ou indivíduos são o público interessado ("stakeholders"), que segundo Freeman deve ser considerado como um elemento essencial na planificação estratégica de negócios.

Mostra-se um esquema descritivo deste tipo de migração:

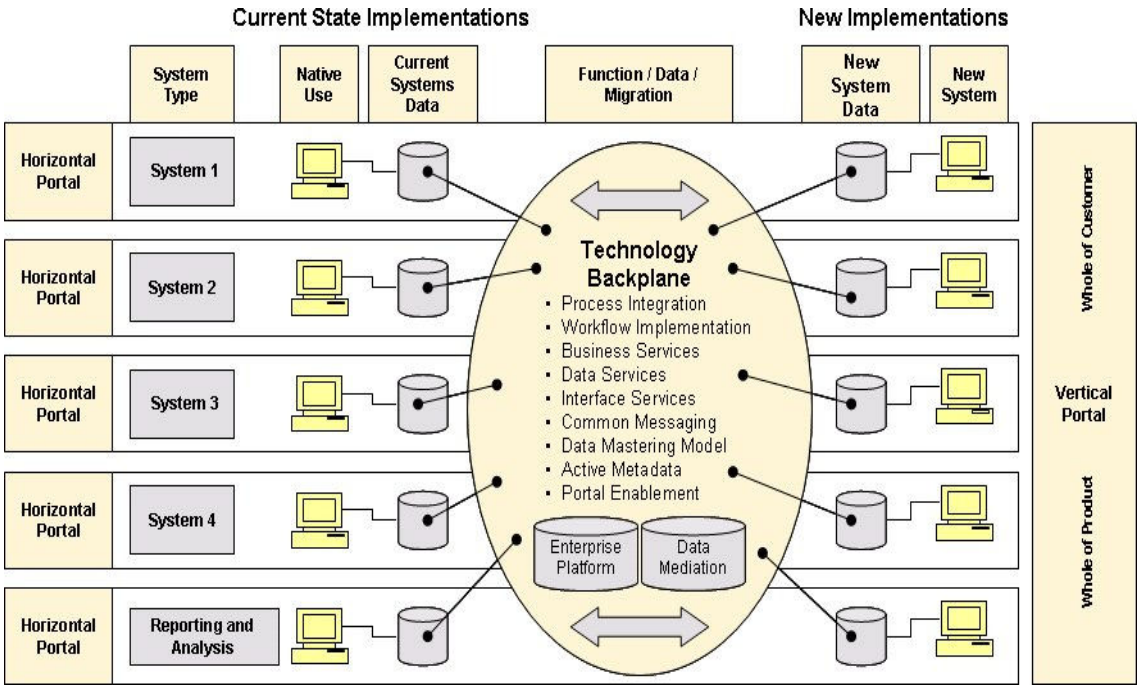

Figura 11 - Cenário de migração complexa

Fonte: http://www.openmethodology.org/wiki/Data_Migration_Solution_Offering

A arquitectura escolhida para este cenário terá que assegurar que os sistemas de origem continuam a agir sobre os dados diariamente na fase de co-existência aplicacional. A melhoria da qualidade dos dados inicia-se através de um processamento "batch"[6] para carregamento inicial, e é aumentado progressivamente à medida que se vai procedendo à integração dos recursos analisados. Com esta, possibilita-se a análise integrada dos dados e procede-se à sua racionalização, como é o caso, por exemplo, da eliminação de registos duplicados, natural nesta estratégia. Inicia-se ainda o processo de mapeamento dos dados entre os dois sistemas.

Dado o enorme esforço associado à construção deste ambiente integrado, poderá aproveitar-se a oportunidade para desenvolver outras iniciativas. É-nos proposto o desenvolvimento e implementação de um centro integrado e operacional de

[6] O processamento Batch é a execução de uma série de programas, vulgarmente chamados de Jobs, num computador, sem a interacção humana, em que todos os inputs são carregados através da parametrização de linha de comandos ou através de scripts.

armazenamento de dados que irá permitir o acesso vertical, ou transversal se preferirmos, a todos os dados, doravante designado por IODS.

Este IODS poderá ser utilizado como plataforma intermediária para o centro de armazenamento de dados, "Warehouse", analítico, bem como para funcionar como um ponto de ligação, "hub", entre aplicações co-existentes, proporcionando uma saída paralela, ou "output", de novas funcionalidade de negócio na parte analítica, por um lado, e de novas funcionalidade operacionais, por outro.

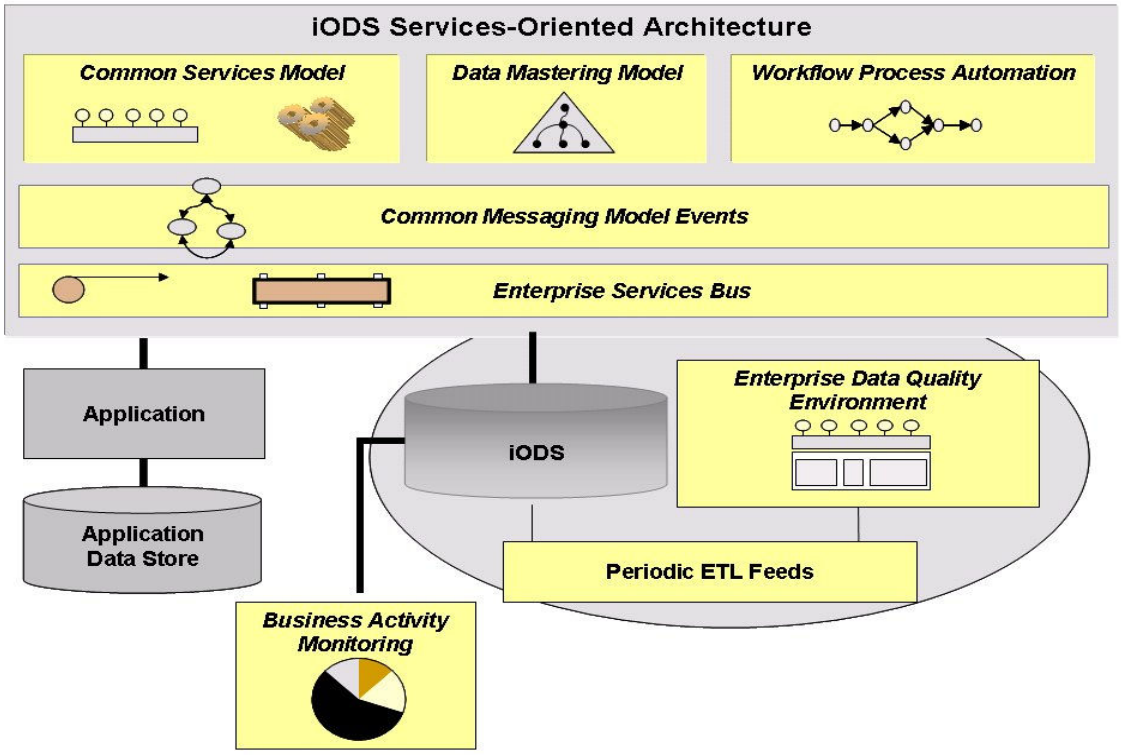

Figura 12 - Componentes chave do centro integrado e operacional de armazenamento de dados

Fonte: http://www.openmethodology.org/wiki/Integrated_Operational_Data_Store

Este IODS é construído através de um padrão "composite"[7] de componentes arquitecturais através do recurso a uma arquitectura SOA [8]. Desta forma, o processo de

[7] Composite é um padrão de projecto de software utilizado para representar um objeto que é constituído pela composição de objetos que lhe sejam similares.

8 Service-oriented architecture (SOA), pode-se traduzir como arquitectura orientada a serviços, e é um estilo de arquitetura de software cujo princípio fundamental preconiza que as funcionalidades implementadas pelas aplicações devem ser disponibilizadas na forma de serviços.

integração também pode derivar de uma perspectiva assente nos processos e na lógica de negócio e não apenas na tradicional integração dos dados existentes nos vários sistemas. Os serviços de "interface" existentes nesta plataforma de dados são semelhantes aos existentes na camada aplicacional proporcionando acesso aos dados através de eventos de negócio granulares.

Esta proposta utiliza os dois tipos de processamento, tanto o "batch", como o despoletado pelo utilizador, em tempo real. A maioria dos requisitos de negócio não obriga ao primeiro tipo, mas, no entanto, as limitações impostas pelos sistemas legados obrigam a que seja contemplado. As melhorias verificadas nas tecnologias de integração permitem que esta se realize de forma mais simples ao mesmo tempo que se sente uma pressão para fornecer mais serviços de integração de dados em tempo real, face ao aumento do nível de exigência dos processos de tomada de decisão no que respeita ao tempo de serviço requerido, e face ao aumento do nível de integração entre sistemas distribuídos.

2.3.1. Avaliação e planeamento da migração complexa

Neste tipo de cenário, existe um envolvimento vertical de todos os actores pertencentes aos níveis constituintes da organização. Esta, encerra determinada finalidade, um propósito ou razão de existir que se traduz na sua missão. Para o efeito estabelece objectivos e define estratégias para os alcançar. Os gestores são os seus elementos constituintes encarregados de atingir com sucesso as metas propostas e para o efeito executam tarefas de planeamento, organização, direcção e controle.

Conseguem identificar-se subsistemas, níveis constituintes da organização com tarefas distintas dentro da mesma. São estes o estratégico, táctico e operacional. Varajão (1998 p.25) diz-nos que "A representação piramidal inicialmente proposta por Anthony (Anthony 1965), indicando o número de pessoas que se encontram tradicionalmente a cada nível, é a mais utilizada para classificar as actividades que têm lugar no seio de uma organização (Ribas 1989). Esta visão, que constitui pacificamente em todo o mundo um verdadeiro paradigma para a conceptualização e a praxis da gestão (Oliveira

1994), considera a organização estruturada internamente em três níveis ou subsistemas: estratégico, táctico e operacional."

O subsistema estratégico é o que se relaciona com o exterior, e compete-lhe o acompanhamento das alterações verificadas reagindo ou pro agindo internamente face às mesmas para que se posicione correctamente em cada contexto temporal. Para o efeito existe um modelo que identifica a relação entre as ameaças, as oportunidades, as forças e as fraquezas da organização, o modelo SWOT ("Strengths, Weaknesses, Opportunities, Threats")[9].

O subsistema táctico, de acordo com Varajão (1998), cuida da articulação entre os outros níveis, adequando as decisões do nível estratégico à operacionalização dos níveis inferiores através da transformação das estratégias elaboradas em programas de acção. Cabe a este nível a administração e controlo das diversas áreas organizacionais e a afectação dos recursos necessários de forma eficiente para cumprir os objectivos particulares de cada área e para a organização como um todo.

Ainda o mesmo autor, diz-nos, a propósito do nível operacional que "A distinção chave entre as actividades de gestão operacional e táctica reside no facto do controlo operacional ser centrado na tarefa, enquanto que a gestão táctica é centrada na decisão. (...) A sua principal responsabilidade é implementar ou executar os planos definidos nos níveis superiores, sendo o horizonte temporal muito reduzido." Varajão (1998 p.31).

Se nas migrações simples e média, não há necessidade, na maior parte dos casos, de grande envolvimento do nível estratégico, aqui, pelo contrário, assume um papel determinante e acompanha o decorrer de todo o processo. A importância das tecnologias da informação é enorme em qualquer organização, e apesar de ser um centro de custos, é através dela que flui toda a informação, bem primário e essencial de qualquer actividade. A capacidade de qualquer sistema de informação de gerar com qualidade a informação pretendida pelos diversos níveis de gestão é essencial para o sucesso organizacional.

[9] http://empreendedor.ifdep.pt/index.php?option=com_content&task=blogcategory&id=74&Itemid=83

2.3.1.1. Esquema de avaliação de negócio e definição estratégica

A proposta de MIKE2.0 define uma abordagem baseada num esquema de avaliação do negócio e de definição da estratégia. Neste, o enfoque será o desenvolvimento de uma estratégia de desenvolvimento da informação e da infra-estrutura que correspondam aos requisitos de negócio, extraída em parte do fluxo aplicacional existente. Para esta análise é concebida uma actividade utilizada para a avaliação do estado actual e futuro do sistema, que na migração complexa inclui a elaboração de um portfólio aplicacional, que documenta os sistemas principais e as suas funcionalidades de alto nível, e a avaliação da gestão e administração da maturidade da informação no que respeita aos dados que a compõem. Isto é, relaciona os recursos de negócio que as organizações pretendem com os dados disponíveis que os sustentam. Após a definição estratégica a nível do negócio é-nos proposta uma arquitectura estratégica para a escolha da solução tecnológica, "Strategic Architecture for the Federated Enterprise".

O esquema de avaliação do negócio e de definição da estratégia é considerado o componente mais importante do projecto de implementação da migração. É aqui que se procede ao mapeamento dos objectivos estratégicos pretendidos com as soluções tecnológicas disponíveis. Esta fase compreende a melhoria da percepção dos conceitos de gestão de informação, a definição dos requisitos estratégicos de negócio, a avaliação do sistema actual e perspectivação do ambiente futuro, o desenho da arquitectura conceptual e a definição do plano geral do projecto.

A avaliação do estado actual do negócio é efectuada, iniciando-se com a definição do estado actual do ambiente, determinante no que será necessário realizar. Esta informação é de alto nível e este processo é contínuo, pois o sistema estará em permanente evolução. Após esta definição inicial, confronta-se o estado actual com o futuro, pretendido, efectuando-se o levantamento das lacunas ("gap") entre os dois, e estudando-se ainda os potenciais ganhos na migração, através do recurso a ferramentas de análise. O portfólio aplicacional é elaborado, de acordo com o modelo de exemplo apresentado na página seguinte, e é efectuada a avaliação da maturidade da informação.

System Name: ClassEd	
Description	Classified & Editorial system including text and image processing. (the organisation also utilises this system for advertising booking, deal pricing and pagination)
Platform	VMS and proprietary database; clients run on Windows
Level 1 & 2 Functions	Content Creation – Text – Normal Content Creation – Text – Wire Content Creation – Photos – Normal Content Creation – Photos – Wire Content Creation – Graphics Production – Backbench Production – Sub-editing
Application Complexity	High
Divisions	Serves all divisions and at the group level
Inter-Divisional Complexity	High
Issues & Limitations	Software has been superseded by the vendor and will be end-of-life in the medium term The business cannot achieve its goals with the existing version of software.
Application Life Expectancy	Considering replacing the system within the next 4 –5 years
System Owner	Business Owner contact Technology contact Operational contact
Comment	

Figura 13 - Modelo de portfolio aplicacional

Fonte: http://www.openmethodology.org/wiki/Application_Portfolio

A definição da arquitectura estratégica para a escolha da solução tecnológica é a fase seguinte. É-nos proposto um modelo conceptual, transversal às aplicações, aos dados, e à infra-estrutura, concebido para responder de forma eficaz às complexidades implícitas de grandes organizações ou grupos de empresas.

Esta arquitectura aborda uma série de recursos, desde os fundamentais até aos mais avançados, emergentes no meio da gestão da informação, constantes do esquema da próxima página, e descreve a forma como o conteúdo e os dados devem ser geridos na organização.

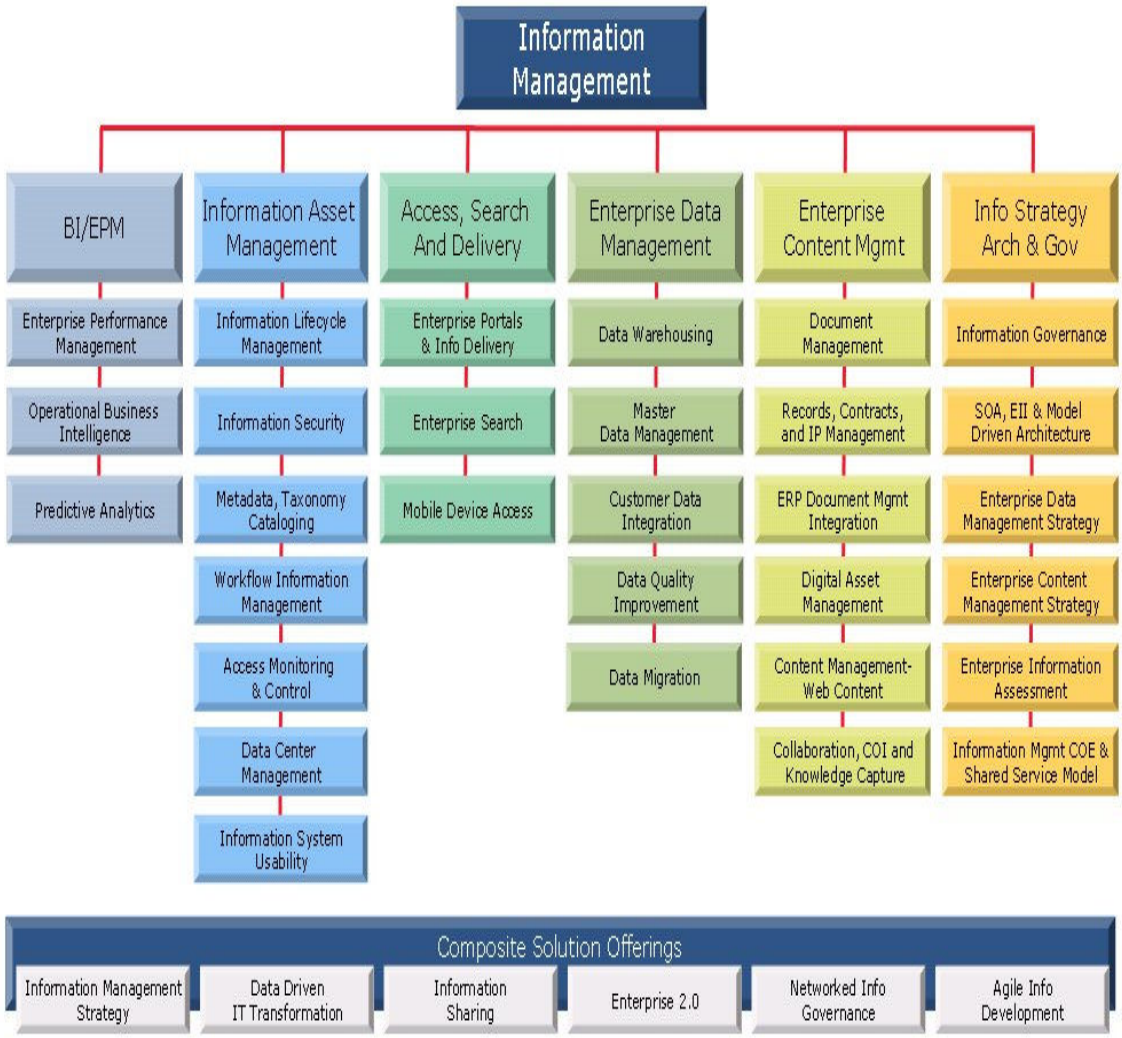

Figura 14 - Modelo de gestão da informação, tipos e conteudos

Fonte: http://www.openmethodology.org/wiki/Enterprise_Information_Management_Concept

2.3.1.2. Esquema de avaliação e opção tecnológica

Este esquema foca-se nos aspectos técnicos do esquema anterior. Estabelece a ligação entre a estratégia de negócio e a arquitectura conceptual definidas, com uma arquitectura aos níveis lógico e físico, a definir.

Os requisitos gerais para a gestão da informação, "business intelligence"[10] e integração dos dados, são refinados, e a sua granularidade aumentada. A infra-estrutura técnica é definida bem como um conjunto "standards" que vão suportar o processo de desenvolvimento. É definido o plano geral de entregas do projecto que serve de ponto de partida à fase de implementação.

A infra-estrutura definida pode incluir uma série de tecnologias como por exemplo as seguintes:

- Investigação dos dados
- Reengenharia dos dados
- Integração dos dados (ETL e EAI[11])
- Gestão de metadados
- Business Intelligence
- Segurança
- Gestão de conteúdo empresarial
- Colaboração
- Ambiente de Hardware e de sistema operativo

10 A Business intelligence (BI), é o conjunto de tecnologias, aplicações e uso de boas práticas para a recolha, integração, análise, e apresentação de informação de gestão ou de outra considerada relevante. Como o seu objectivo é o de proporcionar decisões mais fundamentadas também é apresentado como um sistema de suporte à decisão (DSS).

11 A EAI, Enterprise Application Integration é a estrutura arquitectural lógica que permite a integração de diversas estruturas aplicacionais.

É ainda nesta fase que as actividades de "Data Governance"[12] passam da fase inicial de definição ou criação, para a fase de como irão funcionar. São então estabelecidos os "standards", políticas e procedimentos estratégicos para o desenvolvimento da informação organizacional.

2.3.1.3. Roadmap e actividades fundamentais

Nesta fase são-nos propostas duas actividades complementares que abordam, por um lado, o refinamento do âmbito do projecto de "software", e por outro, a eliminação de riscos em áreas chave que possam existir no mesmo: O "Roadmap"[13] e as actividades fundamentais.

Este "Roadmap" vai fornecer de forma detalhada os requisitos e as soluções a aplicar nas três fases do processo de implementação da migração. Pode ser descrito como a fase de entrega do plano de implementação, contendo os requisitos e a definição da solução. Os requisitos de negócio do sistema são analisados em detalhe, o sistema é descrito através da identificação dos seus componentes principais e da sua forma de funcionamento de forma integrada, tanto entre si próprios, como no seu relacionamento com a solução global.

São identificados todos os objectos ou programas e dados necessários para a modelação do domínio em que a aplicação, ou as aplicações, vão funcionar. São estabelecidos "standards" e são desenvolvidas soluções para problemas comuns, sempre que estas tenham impacto no desenho ou na implementação parcial ou total do sistema. Definem-

[12] A Data governance é uma prática de controlo de qualidade para o acesso, gestão, utilização, melhoria, monitorização, manutenção e protecção da informação organizacional. È um sistema de atribuição de decisões e responsabilidades aos processos relacionados com a informação.

Data Governance Intitute: http://www.datagovernance.com (Acedido a 31.08.2008).

[13] De acordo com Garcia e Bray (1997 p.12), o Roadmap é: "…a needs-driven technology planning process to help identify, select, and develop technology alternatives to satisfy a set of product needs. It brings together a team of experts to develop a framework for organizing and presenting the critical technology-planning information to make the appropriate technology investment decisions and to leverage those investments."

se os ambientes de desenvolvimento e de entrega e é detalhado o plano para cada ciclo de implementação.

O processo das actividades fundamentais revela-se essencial na finalização do esquema de "Roadmap" e estabelece a base para a construção das soluções tecnológica e de negócios. Uma dos seus processos tem o enfoque na análise integrada dos vários fluxos simultâneos de desenvolvimento no que respeita às suas dependências e consequente atribuição de prioridades. É também aqui que se procede à identificação, análise e resolução de possíveis problemas que possam vir a surgir no plano tecnológico.

Torna-se evidente a ausência de maior detalhe nas actividades a realizar abordadas, mas a preocupação fundamental relativa a este tema da migração complexa é a complementaridade ao exposto quando da apresentação das migrações anteriores. O planeamento, a estratégia, mesmo a táctica, assumem aqui um destaque particular e são factores críticos de sucesso que tinham que ser considerados como prioritários neste tema.

Como consequência de qualquer processo de migração é natural a pretensão de preservação da informação anterior. Aliás, nestes sistemas complexos, é também comum o purgo diário de algumas tabelas e o arquivamento dos respectivos dados. Os suportes de armazenamento evoluem, pelo que convém delinear estratégias que permitam o acesso aos dados numa óptica de longo prazo. É o que veremos no capítulo seguinte.

3. O Modelo OAIS e a preservação digital

O rápido crescimento no mundo da informática e das comunicações levou a um aumento nas transacções digitais entre as organizações. A informação segue agora cada vez mais caminhos e meios digitais em detrimento dos meios mais tradicionais como o papel. As próprias organizações entraram ou foram levadas a entrar neste novo paradigma de funcionamento sem prestarem atenção ao problema que estavam a criar.

Possivelmente muitas ou pelo menos algumas das organizações nunca pensaram na possibilidade de enfrentarem a questão de preservação. O modelo OAIS vem definir um conjunto de requisitos e recomendações que as organizações devem seguir para amenizar ou combater esta problemática da preservação da informação digital no longo prazo ou de forma permanente.

Em Janeiro de 2002 o Management Council of the Consultative Committee for Space Data Systems doravante designado por CCSDS aprovou um documento que representa o acordo levado a cabo pelos participantes do CCSDS Member Agencies. Consiste numa série de recomendações técnicas com o objectivo de estabelecer um consenso globalizado no que respeita à preservação digital de informação de forma permanente, ou no longo prazo, como se preferir CCSDS (2002).

Pode dizer-se que se procedeu à definição dos requisitos necessários de forma a obter um "Open Archival Information System", doravante designado por OAIS, e comummente designado por modelo de referência OAIS, pois exactamente disso que se trata: um modelo de referência, onde se estabelecem uma série de recomendações, estratégias a seguir, linhas mestras a ter em conta e onde também são identificados os componentes e agentes necessários para atingir, num determinado arquivo, a preservação da informação digital.

3.1. O modelo OAIS

Em traços gerais, este modelo, assenta em três operações fundamentais: a ingestão da informação, a administração do arquivo e a disseminação da informação. Todas as

características assim como todas as recomendações apontam no sentido de uma preservação digital da informação, mantendo presente a obrigatoriedade desta permanecer acessível e inteligível por longos períodos de tempo. Pretende ainda definir um standard ISO[14] (International Organization for Standardization).

O OAIS não refere nem especifica plataformas computacionais, linguagens de desenvolvimento, sistemas gestores de bases de dados ou interfaces. Enfim, não condiciona o desenvolvimento do sistema ao nível da tecnologia envolvida. Servirá então como um guia para quem pretender desenvolver um arquivo digital.

As implicações da preservação a longo prazo serão a ocorrência de mudanças ao nível das tecnologias envolvidas, dos formatos dos documentos ou até da comunidade de utilizadores (interesse). O modelo em si passa por um conjunto ou organização de pessoas e sistemas que aceitam a responsabilidade de manter a informação digital "sempre" disponível para a sua comunidade de interesse.

Diz-se aberto, (Open A. I. S.) pois as recomendações para o seu desenvolvimento devem ser mantidas em fóruns abertos e disponíveis a todos, oferecendo para o efeito os conceitos necessários para que todas as organizações possam participar de forma activa neste processo no presente, e conceitos para uso futuro no que diz respeito à comparação dos sistemas e das suas arquitecturas. Questões relacionadas com estratégias, técnicas de preservação digital e alterações que os modelos adoptados possam sofrer ao longo do tempo, também são abordadas por este modelo de referência.

É ainda disponibilizada uma base para tratar questões relacionadas com a preservação de informação que não se encontre em formatos digitas. O modelo refere alguns consensos relacionados com todo o processo da preservação que possibilitarão que o mercado aumente a oferta deste tipo de produtos.

As questões inerentes à preservação digital residem no processo de ingestão de informação no sistema, no armazenamento, na sua administração, na preservação, e no acesso e disseminação da informação. A questão relacionada com a troca de informação

14 http://www.iso.org/

entre arquivos também é abordada, assim como o papel que o software assume na problemática da preservação digital.

O OAIS, pode, por definição, ser aplicado a qualquer tipo de arquivo, mas é, no entanto, mais direccionado para organizações que necessitem de ver a sua informação preservada por longos períodos de tempo. Também será do maior interesse para quem pretenda extrair informação de arquivos orientados segundo o mesmo modelo.

Sabe-se que a informação digital pode muito facilmente perder-se ou ser corrompida. No momento da produção da mesma temos um acesso privilegiado à forma como essa mesma informação é produzida, os seus metadados. Quem constrói e desenha os sistemas deve ter em atenção a extrema importância em documentar todo tipo de informação gerada. Contudo facilmente nos apercebemos que muitas vezes se dá a colisão com objectivos de mercado de uma rápida produção e disseminação de produtos para os seus consumidores apesar dessa percepção por parte da área tecnológica da organização.

Revela-se fundamental o acesso à informação pela respectiva comunidade de interesse; um sistema OAIS assenta de forma sólida sobre esta premissa. O sistema pode ser actualizado com informação de forma regular ou não e poderá também ter que dar respostas mais ou menos complexas dependendo dos casos. É um dos objectivos, não de um sistema OAIS em si mas do modelo de referência OAIS. O de proporcionar termos e conceitos que ajudem as organizações a lidar com a preservação.

3.2. Repositórios digitais

Antes de avançar mais no que diz respeito à preservação de bases de dados é importante referir os repositórios digitais. Genericamente, são entendidos como o local onde os conteúdos digitais podem ser armazenados para uso futuro. Esse uso passará pela pesquisa e consulta dos conteúdos. Estes repositórios deverão possuir mecanismos para importar, exportar, armazenar e preservar os conteúdos ou a informação a eles associada.

No âmbito dos repositórios digitais verifica-se que podem existir por um lado as bibliotecas digitais e por outro os arquivos digitais. No caso das primeiras, podemos começar por falar nas bibliotecas tradicionais, cuja informação se encontra guardada ou armazenada em livros, num suporte físico que é o papel. Nas bibliotecas digitais, o paradigma passa do suporte físico para um suporte digital. As colecções estão armazenadas em formato digital e passam a ser acedidas por um computador, local, ou remotamente, através da internet, como todo o tipo de conteúdo digital.

Ramalho e Ferreira (2008), dizem-nos que os arquivos digitais são diferentes das bibliotecas digitais, devido, essencialmente, a três aspectos: as fontes de informação, a organização dos conteúdos e a unicidade dos conteúdos. As fontes da informação nos arquivos digitais são primárias – cartas, processos judiciais, registos paroquiais, etc. No caso das fontes primárias a informação provém directamente de quem a produz, por contraponto às fontes secundárias encontradas nas bibliotecas, nos seus livros. No que diz respeito à organização dos conteúdos, nas bibliotecas estes encontram-se catalogados individualmente, ao passo que nos arquivos os conteúdos são organizados em grupos; podendo ser agrupados pela sua proveniência e pela ordem que detinham originalmente. Os arquivos também se demarcam das bibliotecas devido à unicidade dos conteúdos.

3.3. A preservação digital

Em termos técnicos, um arquivo digital será um repositório de objectos digitais que além da sua representação física ("stream" binária) têm associada meta informação descritiva e meta informação de preservação. Uma das preocupações consistirá em garantir o acesso continuado e a longo prazo à informação contida nas bases de dados relacionais armazenadas. O acesso a esta não significa um simples acesso aos bits que constituem o objecto digital, mas sim, um acesso à informação contida.

A informação tem de fazer sentido para quem a procura. Apesar de a informação digital poder ser preservada de forma exactamente igual, recorrendo apenas a uma simples cópia dos bits que a constituem, não significa que mais tarde seja possível percepcionar o sentido da mesma. Verificamos, como foi já referido, que a evolução na área das tecnologias digitais é enorme, o que constitui um obstáculo na inteligibilidade futura.

Urge realçar que normalmente as plataformas informáticas perdem a sua capacidade de auto preservação num prazo de sensivelmente 5 anos, e o que permite que os bits do objecto digital sejam transformados em algo inteligível ao ser humano são exactamente as plataformas, que até à data têm assegurado a retro compatibilidade.

3.3.1. Objecto digital

Podemos fazer uma distinção entre os objectos ditos nado digitais, os que nasceram já num contexto digital, e os objectos digitalizados, oriundos de um processo de digitalização. Na sua forma mais abrangente e que engloba ambos os casos, podemos considerar como objecto digital todo aquele que é possível representar através de um "bitstream", ou seja, todo aquele que pode ser representado através de uma sequência de dígitos binários, zeros e uns, de acordo com ferreira (2006).

Este mesmo objecto, pode então ser caracterizado a diversos níveis; físico, lógico e conceptual. Em termos lógicos, é representado por dígitos binários, mas teremos contudo que discutir o suporte físico sobre o qual o lógico assenta. Actualmente os suportes físicos mais comuns são o disco rígido e o CD / DVD, apesar de também existem as memorias "flash"e as fitas magnéticas, entre outros. Pegando então no exemplo do disco rígido e dos CD / DVD, inferimos que estes suportes físicos divergem no que concerne à tecnologia usada para guardar os códigos binários.

Verifica-se que a base ou estrutura física do objecto digital é fundamental até para se pensar, desde já, em possíveis estratégias de preservação. Todavia, como camada seguinte, existe a estrutura lógica ou objecto lógico; o qual corresponde à cadeia de dígitos binários. Estes detêm uma determinada disposição que irá definir o formato do objecto, dependendo do software que os irá interpretar. A interpretação por parte do software do objecto lógico irá corresponder ao aparecimento do objecto conceptual, aquele que o ser humano é capaz de entender e interpretar, podendo experimentá-lo. Mostra-se de seguida um esquema correspondente aos níveis de abstracção existentes num objecto digital:

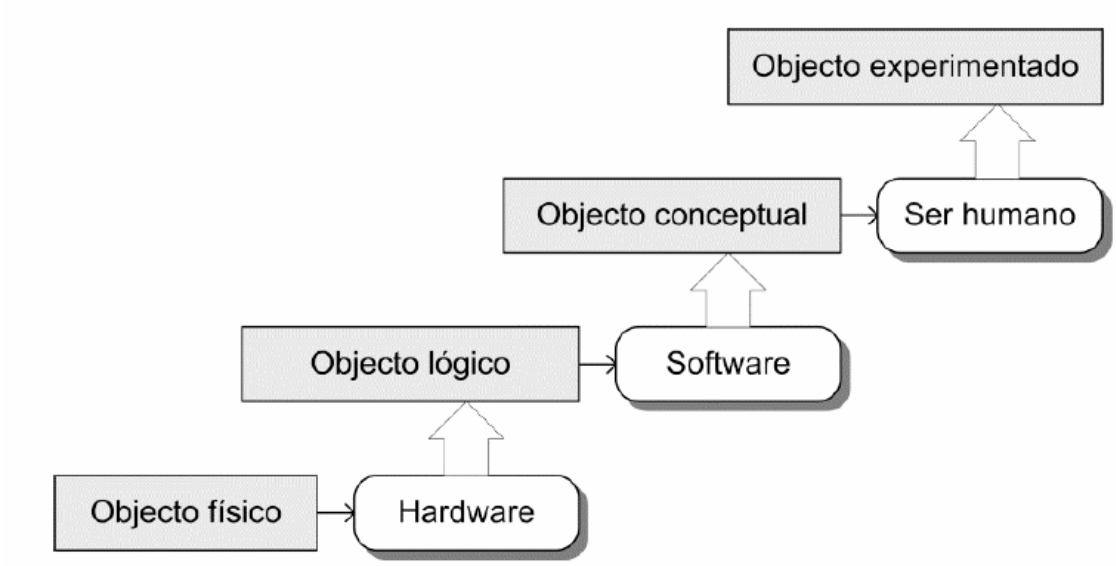

Figura 15 - Níveis de abstracção presentes num objecto digital.

Fonte: Ferreira (2006 p.23).

Através da observação de toda esta cadeia de interpretações e níveis de abstracção, constata-se que a preservação digital ou a estratégia para a preservação irá de certa forma definir qual o estado do objecto a ser preservado.

Os vários tipos de estratégias serão analisados de seguida. A propósito dos vários níveis de abstracção do objecto digital, sabemos agora que o relacionamento que estes estabelecem entre si e a própria existência destes é fundamental para a preservação. Na análise de Ferreira (2006), quando acontecer uma quebra ou falha nessa cadeia, o objecto digital muito certamente deixará de ser inteligível, o que se pode traduzir no risco de se perder para sempre.

3.3.2. Estratégias para a preservação digital

As estratégias para a preservação digital consistem nas várias abordagens reiteradas por vários investigadores, a fim de garantir que o objecto ou objectos digitais se mantenham interpretáveis ao longo dos tempos. Como já foi dito, existem preocupações diferentes no que diz respeito ao que é essencial preservar. Uns dirão que será fundamental garantir a autenticidade e então defendem que a única estratégia possível será a preservação de tecnologia, de forma a manter o objecto na sua forma original. Outros

advogam que bastará preservar o objecto conceptual, aquilo que no fundo interage com o mundo real, ou seja, aquilo que o ser humano irá experimentar. Pelo meio existem ainda outros tipos de abordagens, pelo que se deve assinalar desde já que esta não é uma problemática consensual.

3.3.2.1. Preservação de Tecnologia

Esta abordagem procura preservar o ambiente tecnológico, tal como existe ou existiria. Esta estratégia passa pela conservação e manutenção de todo o hardware e software que caracterizavam e constituíam os objectos digitais originais.

Este tipo de estratégia dá especial ênfase à preservação do objecto físico e lógico, tendo os defensores desta abordagem usado como justificação o facto de esta forma ser a única que poderá representar e recriar fielmente o objecto digital original. Acontece que esta estratégia se revela muito complicada de implementar, pelo menos para todos os tipo de objectos digitais. Teriam que se que encontrar locais físicos e com uma determinada localização geográfica para armazenar todo o legado tecnológico, hardware e software. Diz-nos Ferreira (2006), que no fundo, acabaríamos por construir museus de tecnologia. Os custos, aos mais diversos níveis, seriam incomportáveis.

3.3.2.2. Refrescamento

Esta técnica consiste na passagem da informação existente num determinado suporte físico, que eventualmente possa estar a tornar-se obsoleto, para um mais actual. Como exemplo desta prática temos as cópias de informação de suportes que se estão a tornar obsoletos, como as disquetes, para suportes mais actuais, como os CD. Podemos garantir desta forma pelo menos a informação continua acessível do ponto de vista do hardware. Claro está que será necessário conjugar esta abordagem com outras estratégias, de forma a atingir uma preservação digital efectiva

3.3.2.3. Emulação

O que iremos ter sobretudo será um “software” capaz de emular outros. Podemos constatar que este “software”, denominado de emulador, poderá e virá certamente a sofrer ele próprio de obsolescência, o que é à partida uma desvantagem. Este emulador

tenta recriar as condições tecnológicas para que uma determinada aplicação possa correr sobre ele, assemelhando-se ao ambiente original. Esta estratégia assume principal relevo quando falamos em aplicações executáveis com aspectos dinâmicos e interactivos como é o caso dos jogos, assumindo especiais vantagens, uma vez que é capaz de atingir altos níveis de preservação no que diz respeitos às propriedades e características do objecto digital original.

3.3.2.4. Migração

Esta estratégia consiste na conversão da informação existente num determinado formato ou numa determinada plataforma "software/hardware" para formatos mais actuais. Objectivamente falando, a ideia é conseguir que a informação se mantenha sempre num estado considerado como actual e interpretável pelas tecnologias vigentes. A migração é uma das estratégias para a preservação digital mais usadas actualmente e com mais provas dadas no mundo real. Não resolverá no entanto a questão em definitivo. Periodicamente terá que haver novo processo de migração.

A actualização de versões é uma estratégia de Migração muito utilizada senão a mais comum. Consiste essencialmente em importar os objectos digitais para versões mais recentes, visando manter a actualidade dos mesmos. No entanto, encerrar em si alguns problemas, se estivermos dependentes da empresa que detém determinado produto. Sabemos que muitas vezes quando se passa de uma versão para outra mais actual nem sempre se garante que todos os aspectos e atributos de um determinado documento são incorporados na nesta última, e perante tais descontinuidades é necessário por vezes transportar os documentos para formatos concorrentes que não são dependentes de qualquer fabricante.

3.3.2.5. Normalização

A Normalização pretende encontrar formatos que sejam amplamente conhecidos e com normas internacionais abertas. Deste modo pode-se migrar os objectos para esses formatos, a fim de que as estratégias de preservação se preocupem com um reduzido número de tipos de formatos a preservar. Se ao ingerir num repositório, todos os objectos forem convertidos para um formato único, reduziremos em muito o custo de

preservação, visto que em vez de ser necessário preservar um vasto número de formatos, a preservação irá centrar-se apenas num determinado formato. Por tudo isto, verificamos que a escolha do formato é muito importante, e convém que agrade à comunidade de interesse comportando ainda todos os aspectos relevantes de um determinado tipo de artefacto digital.

3.3.2.6. Encapsulamento

O encapsulamento surge para enfrentar os problemas relacionados com a preservação em que um objecto digital é guardado durante um período de tempo elevado sem que ninguém nem nada o altere. Aqui, pode surgir uma situação em que a migração tenha um custo muito elevado ou até se torne completamente inviável. Esta estratégia consiste em guardar os objectos inalterados, assim como a sua meta informação. Desta forma, quando os artefactos forem solicitados no futuro, a meta informação irá possibilitar a construção dos conversores necessários.

Passo de seguida a abordar a comparação de várias ferramentas ETL[15], de suporte a várias fases do processo de migração de bases de dados.

[15] ETL é acrónimo de Extract, Transform, Load, que à letra significa extracção, transformação e carregamento, e designa o processo de migração de dados, de uma ou mais origens, para um ou mais destinos, com transformação intermédia de determinados valores.

4. Ferramentas ETL

Estas ferramentas, já referidas quando da abordagem ao processo de migração, podem ser indispensáveis em alguns dos casos atrás referidos. Surgiram em inícios dos anos 90 como resposta ao problema da construção de sistemas de informação de apoio ou suporte à decisão, chamados de "Decision Support Systems", doravante designados por DSS.

De acordo com um estudo de Eckerson (2003a) para o "The Data Warehousing Institute"[16], doravante designado por TDWI, as ferramentas de ETL têm a árdua e enorme tarefa de integrar conjuntos de dados heterogéneos, provenientes de sistemas diferentes, através, primeiro, da extracção dos dados, segundo, transformação num formato normalizado, e por último o carregamento dos mesmos para um repositório de dados designado por "Data Warehouse"[17], doravante designada por DW, ou para um sistema de bases de dados diferente do de origem, para que a organização consiga aceder a um dos seus bens essenciais, a informação.

No caso das DW, servem para a construção de repositórios de dados com actualizações periódicas destinado a tarefas subsequentes que conduzem à inferência de determinada lógica que lhes está associada ou à produção de relatórios.

Ainda a propósito das ferramentas de ETL, e no que respeita à forma como implementam o processo de migração, há actualmente uma tendência predominante de efectuar este processo de forma diferente daquela descrita acima. Isto é, esta orientação vai no sentido de se efectuar, primeiro, a extracção, depois o carregamento e só por fim a transformação. A esta abordagem designa-se de ELT e Chandras (2008) descreve-a desta forma: "…and, yes, that's extract-load-transform (also called "pushdown") not conventional extract-transform-load (ETL). There's no doubt that ELT is now a

[16] http://www.tdwi.org/ (Acedido a 03.09.2008)

[17] Uma Data Warehouse é um repositório que contém dados armezenados na sua forma binária. São construídas para facilitar a geração dos processos de análise e de relatórios. Temos como componentes essenciais as ferramentas utilizadas na consulta e análise dos dados, na extracção, transformação e carregamento dos dados e na gestão do dicionários de dados.

mainstream capability, and (…) inclusion of pushdown optimization in the recently released (…) brings ELT the legitimacy it deserves."

Continua o seu artigo relevando o facto deste processo acelerar todo o processo de migração de dados, e termina da seguinte forma: "…pushdown seems like the best thing that has happened to ETL since data quality and EII, and I can see mixed-mode options, such as that offered by (…), giving ETL solution architects and designers a needed boost in designing sophisticated ETL solutions and containing data load jobs within processing Windows.".

4.1. Business Intelligence e Data Warehouse

Noutro dos seus artigos, Eckerson (2003b), faz a analogia entre o processo de "Business Intelligence", doravante designado por BI, e uma refinaria, em que se procede à transformação de uma matéria prima, dados, numa multiplicidade de produtos relativos à informação. O BI evoluiu, e actualmente é considerado um ramo das tecnologias da informação que aborda o desafio de transformar meros dados em informação útil. Engloba temas como relatórios, análise, apresentação e integração de dados.

Estas tecnologias destinam-se a obter uma compreensão melhor dos dados para agilizar e acelerar o processo de tomada de decisão. A sua missão é, de acordo com Eckerson (2003b), a melhoria da eficácia e eficiência organizacional. Na página seguinte mostra-se um esquema, extraído do mesmo artigo que nos mostra a analogia com uma refinaria.

Figura 16 - O ambiente de BI abordado como uma refinaria.

Fonte: http://www.tdwi.org/research/display.aspx?ID=6838

Nesta proposta, pega-se na matéria-prima, os dados, que são alvo de um processo de refinação conducente à sua transformação em sabedoria. Em si mesmos, os dados podem significar muito pouco se não comparados entre si, com outros, ou contextualizados em determinada situação. Assim, por via das ferramentas de análise estes dados são transformados em conhecimento. Através da aplicação de regras e de modelos a este conhecimento podem gerar-se planos a ser traduzidas em acções por via da experiência. Após a revisão, refinação e mensuração poderá haver sabedoria.

Os vários passos envolvidos na criação de um sistema de BI são, a extracção de dados de vários sistemas operacionais, também chamados de transaccionais, por parte da DW, transformando-os em informação. Depois, através de ferramentas de análise, como é o caso das consultas, relatórios, "Online Analytical Processing"[18], doravante designado

[18]OLAP, ou, Online Analytical Processing, é uma abordagem que permite responder rapidamente a pesquisas multidimensionais a uma DW. O seu domínio de aplicabilidade típico é a criação de relatórios de negócio para as área de vendas, marketing, relatórios de gestão, gestão de processos de negócio, orçamentação e previsão, relatórios financeiros e áreas similares.

por OLAP e através de ferramentas de "Data Mining"[19], os utilizadores investigam a informação e identificam tendências, padrões e excepções, transformando a informação em conhecimento. A partir deste conhecimento e com o recurso a modelos mais ou menos complexos procede-se ao processo de inferir regras, válidas, que possam servir de base ao planeamento e a acções futuras. Mais recentemente, tem-se promovido o desenvolvimento de sistemas de BI a funcionar cada vez mais em tempo real, chamados de quase tempo real.[20]

A estrutura base deste sistema, o "Information Integrator"[21], é a última versão de um "Middleware" que fornece os recursos chave necessários às aplicações de integração de dados e de BI, acedendo e fornecendo, entradas ou saídas, independentemente da localização física ou lógica dos mesmos, conforme nos é ilustrado na figura 17, na página seguinte, em que se consegue visualizar um núcleo composto de três partes. Instrumentos de pesquisa, o motor de integração, "Federation Engine" e uma camada de conectividade a diversas interfaces.

[19] Data mining é o processo de percorrer grandes quantidades de dados procedendo à recolha de informação relevante. Pode descrever-se como a extracção não trivial de informação, previamente desconhecida e útil a partir de dados conhecidos. No contexto dos ERP é a análise lógica e estatística de grandes volumes de dados transaccionais procurando identificar modelos que possam ajudar no processo de decisão.

[20] http://www.redbooks.ibm.com/abstracts/SG246071.html (Acedido a 04.09.2008).

[21] http://www-306.ibm.com/fcgi-bin/common/ssi/ssialias?infotype=an&subtype=ca&appname=Demonstration&htmlfid=897/ENUS204-202 (Acedido a 04.09.2008).

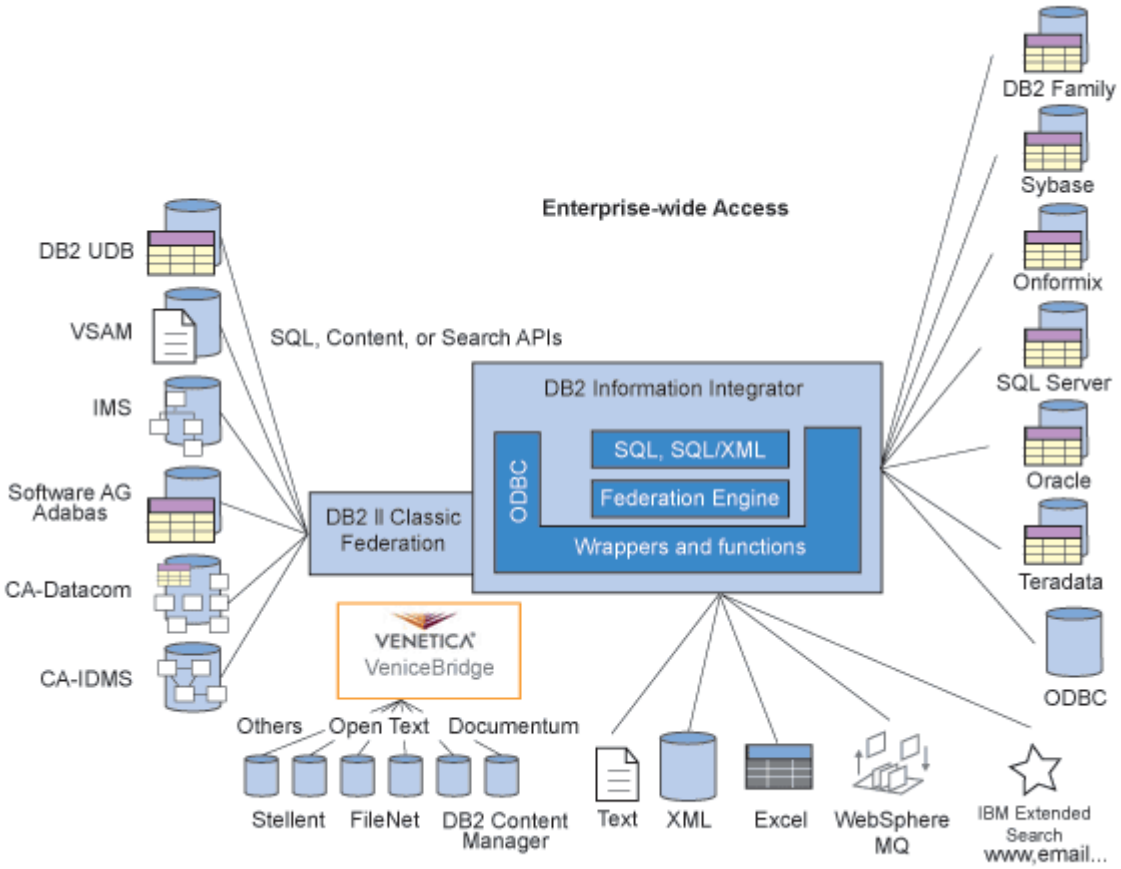

Figura 17 - Sistema de integração de informação em ambiente DB2.

Fonte: http://www.ibm.com/developerworks/db2/library/techarticle/dm-0411simchuk/

De seguida mostra-se, de forma básica, a estrutura de um sistema BI, em que, grosso modo, apenas se observam dois componentes, o ambiente de alimentação da DW e o ambiente analítico. Existe, por um lado, uma equipe técnica que constrói a DW através da construção de dimensões de dados em que estes são representados em função de outros.

Por exemplo no caso da dimensão tempo podem não me interessar todos os movimentos diários, mas a agregação dos movimentos mensais face ao volume de negócio. Para o efeito os dados são extraídos, limpos, modelados, transformados, transferidos, e

carregados. Este processo repete-se de forma automatizada com maior ou menor frequência e ficamos com o ambiente da DW funcional.

Por outro lado existe um ambiente analítico construído para os utilizadores de negócio através do qual realizam pesquisa à DW que podem ser a pesquisa, a geração de relatórios, a análise, inferir informação, a visualização ou a realização de acções com base na informação disponível.

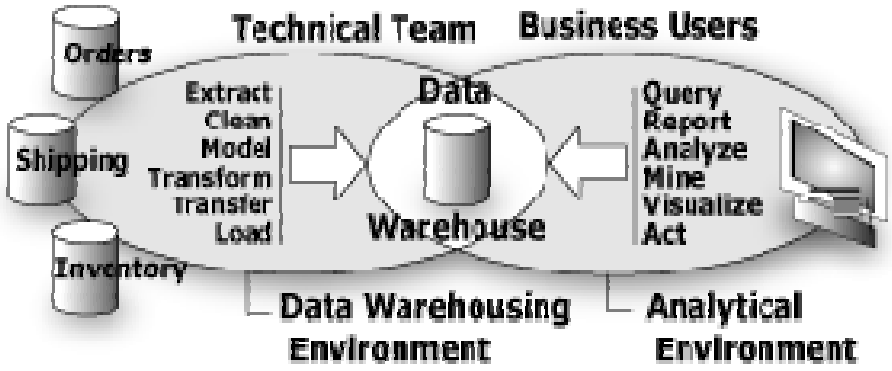

Figura 18 - Estrutura básica de um sistema de BI.

Fonte: http://www.tdwi.org/research/display.aspx?ID=6838

Foi abordado o contexto em que as ferramentas ETL existem e a importância que estas detêm no âmbito do mesmo. O seu domínio é a extracção, transformação e integração periódica de dados heterogéneos, cuja origem e destino podem ser inúmeros alvos, destinadas a sistemas de BI ou à simples migração de dados. Passo agora a abordar o mercado dessas mesmas ferramentas. Quais as existentes? As melhores? Comerciais ou "Open Source"? Qual o seu custo?

4.2. Mercado das ferramentas ETL

De acordo com o Meta Group (2004), citado por Scott et Al. (2005), o mercado de ferramentas de ETL terá crescido cerca de 10 a 20% ao ano nestes últimos quatro anos, devido à crescente implementação de projectos de integração de dados e de

consolidação em DW's. Evidentemente que muitas das soluções anteriores, feitas à medida através de desenvolvimento específico para o efeito, foram, e cada vez mais continuarão a ser, substituídas por este tipo de ferramenta.

De entre estas, lê-se no estudo de Forrester (2007) que as mais conhecidas são as de nível empresarial, como a Ab Initio[22], a Business Objects[23], a IBM[24], a Informatica[25] e a SAS[26]. Além destas dever-se-á considerar também a Microsoft, única apenas executável em plataformas Windows enquanto que todas as restantes detêm a particularidade de serem executadas em várias plataformas. Todas elas asseguram a conectividade com uma série de fontes e destinos diferentes, excelentes níveis de desempenho e de escalabilidade, mas também um custo bastante elevado, podendo ir até aos 100 000 USD.

É no entanto de referir que, actualmente, após a escolha e implementação de um sistema de BI com uma destas ferramentas, é muito difícil e oneroso reverter a decisão. Isto é, é complicado inflectir a decisão e passar para outra ferramenta. Nada do que se faz com cada uma destas ferramentas é passível de portabilidade para qualquer outra. A consequência é a manutenção dos preços altos do lado dos fabricantes.

Forrester (2007) prossegue a sua análise e transporta-nos para um tipo de mercado diferente, em que o suporte a ambientes heterogéneos é limitado. Temos como exemplo o Microsoft's SQL Server Integration Services (SSIS)[27], integrado nas versões 2005 e

[22] http://www.abinitio.com/abinitio/ab.nsf/index-flash(Acedido a 03.09.2008)

[23] http://www.businessobjects.com/ (Acedido a 03.09.2008)

[24] http://www-01.ibm.com/software/data/businessintelligence/(Acedido a 03.09.2008) No endereço http://www.ibm.com/developerworks/db2/library/techarticle/dm-0411simchuk/, (Acedido a 03.09.2008), o autor apresenta-nos um artigo interessante sobre a implementação para UNIX®, Linux™, e Windows® de um processo de extracção, transformação e carregamento de dados numa base de dados IBM®DB2® Universal Database™ (UDB) através da utilização de produtos da IBM. Informa ainda sobre outros produtos disponíveis no mercado.

[25] http://www.informatica.com/ (Acedido a 03.09.2008)

[26] http://www.sas.com/ (Acedido a 03.09.2008)

[27] http://msdn.microsoft.com/en-us/library/ms141026.aspx, SQL Server 2008, (Acedido a 04.09.2008).

2008[28] do SQL Server, em que neste último é parte de um sistema de BI e o Oracle Warehouse Builder (OWB)[29], integrado nos SGBD Oracle. Enquanto que o SSIS apenas corre em plataformas Windows, mas podendo ligar-se a outros SGBD, o OWB é essencialmente destinado a uma utilização sobre RDBMS ("Remote Database Management Systems") Oracle.

O mesmo autor apresenta-nos ainda um outro mercado, "Open Source", em que existem alguns projectos que conseguem disponibilizar um bom conjunto de recursos, dos quais destaca o Clover.ETL[30], o Ketl[31], o Kettle[32] da Pentaho, e o Talend[33].

Os projectos Pentaho e ainda o Jaspersoft[34] são projectos "Open Source" de BI.

Estas ferramentas, ETL, apresentam algumas limitações face às comerciais, pois não dispõem de todos os recursos necessários à realização de projectos de migração, integração e análise de dados, em ambientes organizacionais complexos, não deixando no entanto de ser, actualmente, e cada vez mais, excelentes alternativas para a execução de projectos mais simples. A medida desta simplicidade, ou complexidade, e os recursos disponíveis em cada uma destas ferramentas, em constante evolução, poderão ditar o sucesso da sua utilização.

[28] http://download.microsoft.com/download/C/8/4/C8470F54-D6D2-423D-8E5B-95CA4A90149A/SQLServer2008_BI_Datasheet.pdf , (Acedido a 04.09.2008).

[29] http://www.oracle.com/technology/products/warehouse/index.html , (Acedido a 04.09.2008).

[30] http://www.cloveretl.org/ , (Acedido a 04.09.2008).

[31] http://sourceforge.net/projects/ketl , (Acedido a 04.09.2008).

[32] http://kettle.pentaho.org/ , (Acedido a 04.09.2008).

[33] http://www.talend.com/index.php , (Acedido a 04.09.2008).

[34] http://www.jaspersoft.com/ , (Acedido a 04.09.2008).

4.2.1. Limitações das ferramentas ETL Open Source

O conhecimento dos limites deste tipo de ferramentas é decisivo na sua utilização. O sucesso de determinado projecto vai com certeza derivar de uma boa avaliação prévia da capacidade dos recursos existentes.

Podemos começar por apontar um problema relacionado: a conectividade com a informação crítica armazenada em sistemas não relacionais, como é o caso dos mainframe, das aplicações legadas, das "message queues"35 (TIBCO, JMS, WebSphere MQ), e o suporte a formatos de ficheiro "standard" (HIPAA, SWIFT, ACCORD), diz-nos Forrester (2007).

São ainda abordados dois aspectos essenciais: o desenvolvimento simultâneo pelos elementos das equipas envolvidos no projecto e os possíveis requisitos de transformação complexa de campos. No primeiro caso, se o projecto não exigir mais que dois ou três intervenientes, então não haverá grande problema na utilização deste tipo de ferramentas, mas se o seu âmbito for de maior complexidade, onde seja exigida uma grande colaboração entre várias equipas compostas por vários indivíduos, será aconselhável a escolha de uma ferramenta comercial. No que respeita ao segundo caso, verifica-se que a implementação de regras de transformação complexas nas ferramentas não comerciais é frequentemente conseguida à custa de "scripts"[36], enquanto que nas outras, comerciais, existem "interfaces" de fácil utilização que facilitam imenso o trabalho, principalmente se a aplicação destas for em grande número.

Após esta introdução às ferramentas de ETL passo a abordar uma metodologia para a avaliação das mesmas. Alguns estudos sobre a matéria referem aproximadamente 60 aspectos a considerar. Outros, ainda, cerca de 80. Vou apresentar a abordagem efectuada por Henry (2005), que contempla cerca de 40 aspectos, contidos em várias categorias.

[35] A message queue é um componente de engenharia de software utilizado para a comunicação entre processos diferentes ou dentro do mesmo.

[36] Script é uma descrição geral de qualquer programa escrito em linguagem interpretada, ou seja, não compilada.

4.3. Metodologia de avaliação de ferramentas de ETL.

O processo de apreciação considera as seguintes como valências essenciais de qualquer ferramenta.

- O seu custo
- A facilidade de utilização
- A flexibilidade
- A robustez
- A escalabilidade
- A rapidez.

Por custo entende-se o somatório dos valores envolvidos na aquisição, formação, implementação, manutenção, "upgrades" futuros e eventual suporte do vendedor. A complexidade e a forma da curva de aprendizagem também são importantes, pois caso a sua utilização fosse muito complicada e o seu ensino difícil, seria considerado inútil.

A flexibilidade determina a capacidade de personalização do produto, isto é, o grau de parametrização que cada indivíduo consegue impor ao produto para que fique ao seu gosto e ainda o grau da capacidade e amplitude das funcionalidades da ferramenta.

A robustez mede a relação de dependência do produto com o seu meio e a medida em que as alterações deste último têm maior ou menor impacto no primeiro. A robustez implicará, por exemplo, a capacidade de lidar com falhas de rede, falhas ou quebras do servidor, ou ainda a falta de espaço de armazenamento.

A escalabilidade determina e mede a capacidade do produto de operar projectos simples ou complexos com sucesso, bem como a quantidade, maior ou menor, de dados envolvidos.

A rapidez, por último, é a medida temporal do processo, directamente dependente da complexidade e quantidade de dados envolvidos.

Relativamente às características essenciais das valências, apresenta-nos ainda oito aspectos fundamentais que podem estar presentes em cada uma delas, a saber:

- A arquitectura do produto
- O suporte a dados
- A extracção dos dados
- A transformação dos dados
- O carregamento dos dados
- A verificação da igualdade e fusão dos dados
- A gestão de meta dados
- O ambiente de desenvolvimento

Para a avaliação da arquitectura são considerados o processo de instalação, o suporte a múltiplas plataformas no que respeita a sistemas operativos e bases de dados, a recuperação e o reinício em caso de falha, o armazenamento intermédio, a encenação do processo, o processamento parcial, o suporte a actividades paralelas independentemente da sua fonte no que respeita a dados, a capacidade de execução paralela de vários "jobs" ou módulos, e a documentação.

O suporte a dados tem a ver com os formatos, tipos e índice de actualidade, como é o caso de dados em tempo real, que podem intervir no processo.

A avaliação da extracção e do carregamento terá que ver com a capacidade de a realizar de um número heterogéneo de fontes, ou destinos, enquanto que a verificação da igualdade e fusão dos dados é auto-explicativa.

Na transformação dos dados será avaliada a capacidade de definição destas regras, a quantidade de regras e funções pré estabelecidas e ainda a capacidade de detectar e corrigir ou remover dados corruptos ou errados de determinado conjunto, tabela ou BD

numa política de qualidade de dados. Deverá ainda ser possível o recurso a linguagens de programação para a definição de regras complexas.

A gestão dos metadados é essencial na construção de um DW e são indispensáveis boas funcionalidades que ajudem os utilizadores nesta tarefa. As características essenciais devem ser a extensibilidade, porquanto pode haver necessidade de expandir o repositório inicial. Este deve também estar num formato aberto, bem documentado e de fácil acesso. Os metadados devem ainda poder ser partilhados com outras aplicações, nomeadamente na integração em produtos de BI. Para finalizar, a ferramenta deverá gerar relatórios do conteúdo deste repositório.

Por último, o ambiente de desenvolvimento. É neste que os utilizadores vão passar a maior parte do tempo e as questões mais relevantes são: A existência de uma interface gráfica de utilização, o suporte a linha de comando, um ambiente de ferramentas integrado, existência de processamento sequencial, suporte à depuração do programa, geração de relatórios de ETL, administração centralizada e a possibilidade de efectuar o agendamento de tarefas.

Após a apresentação destes aspectos essenciais aborda-se a temática dos cenários de teste, as medidas quantitativas, escalas a definir, pesos relativos e é-nos apresentada uma matriz de avaliação para cada um dos aspectos referidos acima, em que são relacionados com os critérios de avaliação que os suportam.

Aqui, iremos atribuir valores ao desempenho verificado em cada uma das características específicas de cada critério e repetiremos o procedimento para todas as outras. Estes valores podem ser os representados na tabela da página seguinte:

Definição	Valor numérico
1	Não corresponde às expectativas
2	Ligeiramente abaixo das expectativas
3	Corresponde às expectativas
4	Ligeiramente acima das expectativas
5	Excede as expectativas

Tabela 3 - Escala quantitativa

Fonte: Henry (2005)

No que respeita à arquitectura do produto teríamos como exemplo a representação da tabela seguinte:

	Facilidade de utilização	Flexibilidade	Robustez	Escalabilidade	Velocidade
Arquitectura do produto					
Processo de instalação					
Suporte a plataformas					
Recuperação a falhas					

Tabela 4 - Matriz de avaliação

Fonte: Henry (2005)

Neste caso atribuímos valores a cada um dos espaços da tabela e dividimos pelo seu total, oito. Isto se todos se considerassem com o mesmo peso relativo. Estabelecem-se também os pesos relativos das valências face ao total, que representa a avaliação atribuída à ferramenta, de acordo com a tabela da página seguinte:

Valência	Peso relativo
Custo	0.10
Facilidade de utilização	0.20
Flexibilidade	0.20
Robustez	0.20
Escalabilidade	0.20
Velocidade	0.10

Tabela 5 - Peso relativo das valências

Fonte: Henry (2005)

De seguida estabelecem-se os pesos relativos de cada um dos critérios, que poderá ser como se ilustra:

Valência / Critério	
Facilidade de utilização	**Peso relativo**
Arquitectura do produto	0.10
Extracção de dados	0.15
Transformação de dados	0.15
Carregamento de dados	0.15
Verificação da igualdade e fusão de dados	0.05
Gestão de meta dados	0.15
Ambiente de desenvolvimento	0.25
Flexibilidade	**Peso relativo**
Arquitectura do produto	0.05
Suporte a dados	0.15
Extracção de dados	0.15
Transformação de dados	0.15
Carregamento de dados	0.15
Verificação da igualdade e fusão de dados	0.05
Gestão de meta dados	0.20
Ambiente de desenvolvimento	0.10
Velocidade	**Peso relativo**
Arquitectura do produto	0.05
Extracção de dados	0.30
Transformação de dados	0.30
Carregamento de dados	0.30

Verificação da igualdade e fusão de dados	0.05
Robustez	**Peso relativo**
Arquitectura do produto	0.20
Suporte a dados	0.05
Extracção de dados	0.20
Transformação de dados	0.20
Carregamento de dados	0.15
Verificação da igualdade e fusão de dados	0.05
Gestão de meta dados	0.10
Ambiente de desenvolvimento	0.05
Escalabilidade	**Peso relativo**
Arquitectura do produto	0.10
Extracção de dados	0.20
Transformação de dados	0.20
Carregamento de dados	0.20
Verificação da igualdade e fusão de dados	0.05
Gestão de meta dados	0.20
Ambiente de desenvolvimento	0.05

Tabela 6 - Peso relativo dos critérios de avaliação

Fonte: Henry (2005)

Assim, com o preenchimento destas tabelas e com o apuramento do valor final, atinge-se determinado resultado abstracto que servirá de comparação com outras ferramentas. Os valores dos pesos relativos poderão ser alterados, não invalidando a metodologia. No entanto esta conclusão assenta no pressuposto de que o ambiente de teste é o mesmo, caso contrário não terá qualquer validade.

De seguida proceder-se-á comparação de três ferramentas ETL a utilizar na execução de um processo de migração.

4.4. Comparação de ferramentas ETL na execução de uma migração

O caso que vai ser apresentado de seguida baseia-se num caso real de migração executado numa instituição de crédito deste País. Nesse, optou-se pelo carregamento manual dos dados a nível de catálogo, que é o domínio do nosso estudo. A experiência anterior assim o ditou.

Posto perante a possibilidade do preenchimento de aproximadamente 5000 folhas de Excel de forma manual, procedi à importação das tabelas do sistema que continham os dados para o mesmo, e utilizei o modelo que pretendiam preenchido para lhe adicionar as "combo box" necessárias para a selecção dos produtos.

Depois de o ter feito desta forma, ainda sugeri que fosse utilizada uma base de dados, mais fácil de actualizar, e desenvolvi programaticamente um "interface" que lhe acedia.

A migração em causa é a dos atributos em catálogo dos produtos que são abrangidos pela aplicação de liquidações, ou seja, daquela que gera movimentações a crédito ou a débito nas contas dos clientes. Esta, trabalha a três níveis, catálogo, onde estão os produtos, contas, onde estão condições particulares, e clientes, onde também estão condições particulares.

Por outro lado, cada produto terá as suas especificidades na liquidação, e existem aplicações diferentes de liquidação consoante o tipo de produtos. Assim, o nosso foco, o domínio do presente trabalho prático, vai ser o da migração dos atributos da BD de catálogo, que são variáveis na aplicação de liquidações dos produtos de contas à ordem.

Para o efeito teve que haver um trabalho prévio de "limpeza" dos dados, tanto na realidade como na execução desta apresentação. No decurso da presente, teve que haver adaptações, até porque, uma vez que uma das BD que se utiliza é o Oracle 10g Express, dá-se a impossibilidade de importar uma tabela com aproximadamente 300 colunas, que por sinal também não cabe no Excel. Por outro lado, o cenário de consolidação recente do sistema bancário Português originou que à data ainda houvesse um atributo em sistema que identificava o banco de origem, e por essa via todos os valores em causa podiam ser diferentes.

A existência destas tabelas, enormes, que contrariam em absoluto o processo de normalização de Boyce-Cood têm a sua existência justificada pelo factor desempenho. Em tempo real torna-se bastante mais rápida a leitura de apenas uma tabela ao invés de percorrer várias tabelas na procura de determinado registo. Ainda assim, podem-nos ser apresentadas de duas formas. Podem existir de forma estática em sistema, ou ser geradas dinamicamente num pré processamento anterior ao processo que as irá utilizar.

Estas soluções arquitecturais estão directamente relacionadas com o custo de processamento que é responsável, nos grandes sistemas, pela maior fatia do orçamento.

As ferramentas escolhidas inicialmente para a realização do processo de migração foram o Microsoft Integration Services existente no SQLServer 2008 Enterprise Edition, o Talend, e o MapForce 2008. A base de dados de origem foi o Oracle 10g Express Edition e a base de dados de destino foi o SQLServer 2008.

Se inicialmente se pensava utilizar o MapForce 2008, quando da percepção de que o mesmo não dispunha de mecanismos de "Data Flow" (DF), foi imediatamente descartado pois a sua utilização implicaria a criação de ficheiros ou tabelas temporárias de armazenamento de dados que no caso concreto atrasaria bastante a realização do processo de migração, sem necessidade. A ferramenta tem um custo aproximado de € 1000 na sua versão mais simples e o Talend, ferramenta "Open Source" de utilização gratuita, realiza a tarefa com recurso a mecanismos de DF.

Assim foi efectuado um processo de migração de dados com base em 9 tabelas de origem e uma de destino com a transformação dos valores dos campos envolvidos. Utilizou-se na prática o Integration Services (IS) e o Talend (TL).

As imagens seguintes retratam o processo de DF no ambiente IS.

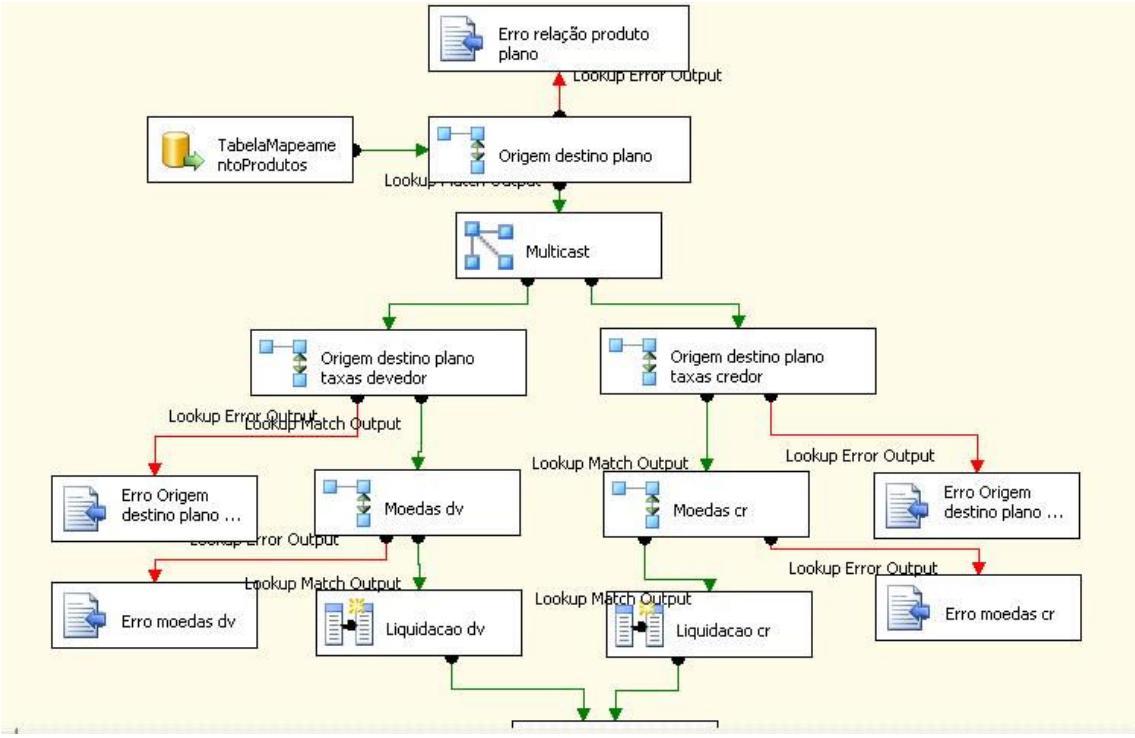

Figura 19 - Ilustração do processo de Data Flow realizado no Integration Services (1)

Para que se perceba melhor o processo há que dizer que os produtos têm associados planos de remuneração. É nestes planos de remuneração que se encontram as taxas a pagar. A sua natureza pode ser credora, devedora ou de liquidação antecipada.

O processo de migração passa pelo seguinte:

1. Identificar na tabela de origem dos produtos quais os que são alvo de migração. A forma de o conseguir foi o "Lookup" na tabela de mapeamento de produtos, onde existem todos os produtos a migrar e respectiva conversão de código. Este "Lookup" não é mais que um "Inner Join" entre duas tabelas e é utilizado de forma recorrente ao longo do DF. Em 752 linhas migraram 382 produtos.

2. De seguida procedeu-se à duplicação da tabela resultante, face à existência de taxas credoras e devedoras, para proceder a um novo "Lookup" à tabela que contém as taxas efectivas a aplicar.

3. Os identificativos códigos de moeda, como os que designam o Euro, EUR, ou o escudo, PTE, tiveram que ser convertidos, aqui, com novo recurso a um "Lookup".

4. Após esta, foi adicionado um novo campo, identificador do tipo de liquidação, credora, ou devedora, no sistema de destino e foi efectuado uma união das duas tabelas dado que todas as características específicas das taxas já tinham sido trabalhadas. Aqui ficou-se com 764 linhas.

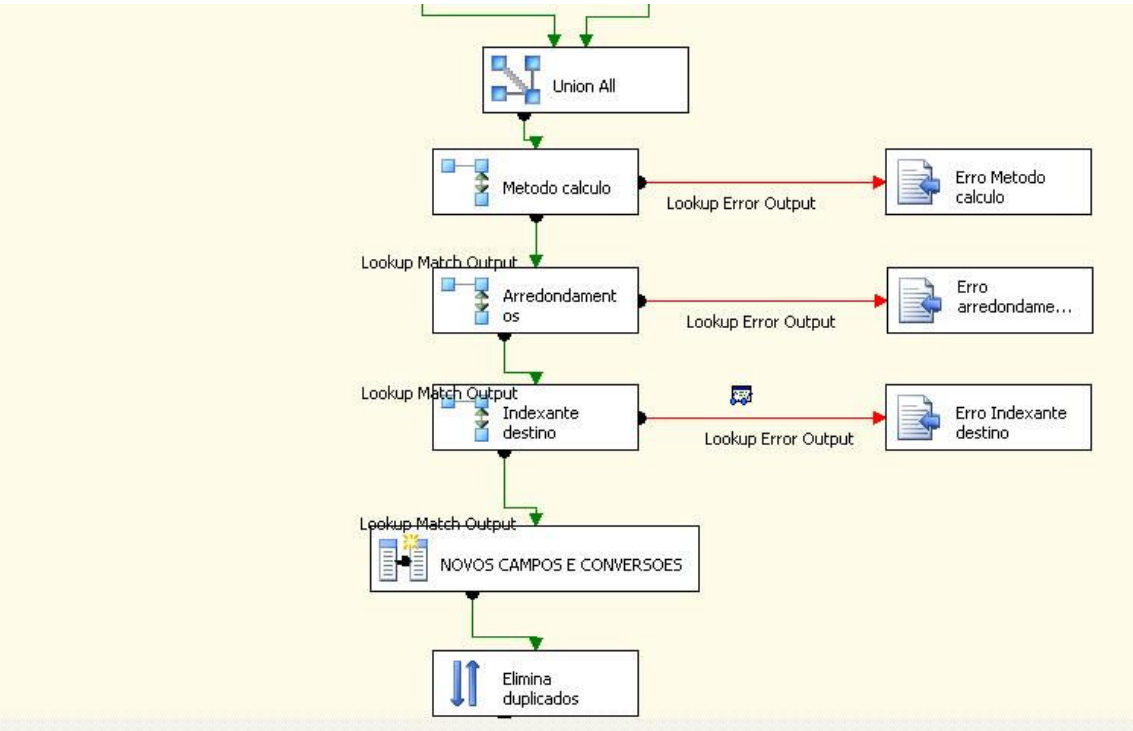

Figura 20 - Ilustração do processo de Data Flow realizado no Integration Services (2)

5. Seguidamente procedeu-se ao mapeamento do tipo de cálculo de juros efectuado quando do processo de liquidação. A título exemplificativo poder-se-á perceber a diferença entre um cálculo que contemple os valores existentes na conta dia após dia e um cálculo efectuado sobre a média dos saldos existentes na mesma conta. Mais uma vez foi utilizado o "Lookup".

6. Depois mapeou-se os códigos de arredondamento de valores. Estes, poderão existir até à oitava de ponto percentual, e recorreu-se a novo "Lookup". Procedeu-se de forma semelhante no que respeita aos indexantes, que são taxas que servem de referência ao cálculo de juros, como por exemplo a Euribor.

7. A transformação chamada novos campos e conversores é responsável pela adição de novos atributos à base de dados de destino bem como à conversão do tipo de dados de alguns dos atributos que têm vindo a ser transformados. De seguida são eliminados valores duplicados e é feita a replicação das 764 linhas resultantes em três novas tabelas temporárias cada uma delas com um escalão de montante e respectiva taxa associada, como se observa abaixo.

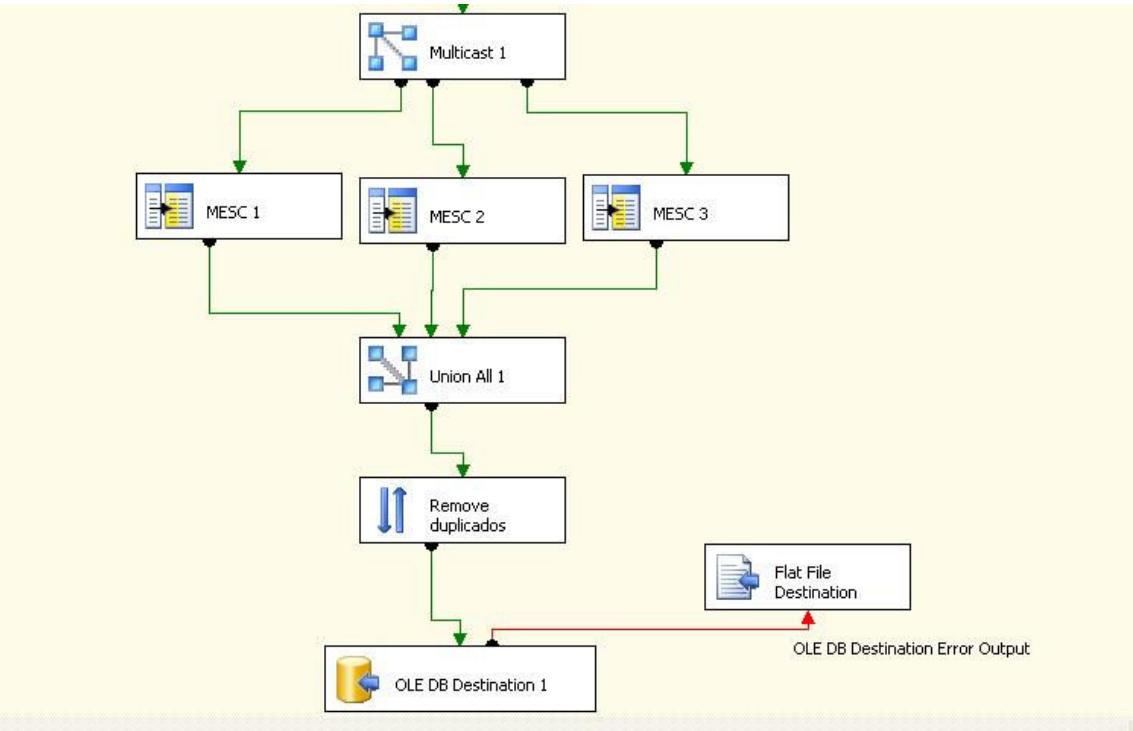

Figura 21 - Ilustração do processo de Data Flow realizado no Integration Services (3)

8. Seguidamente procedeu-se a união dos "data set" e o resultado, 2292 linhas foi alvo da eliminação de valores duplicados tendo originado um total de 1644 linhas inseridas na BD de destino. O tempo de migração foi de 4,861 segundos e não houve qualquer registo rejeitado.

Paralelamente utilizou-se o Talend para realizar o mesmo processo, tendo sido utilizada uma abordagem diferente ao nível do DF.

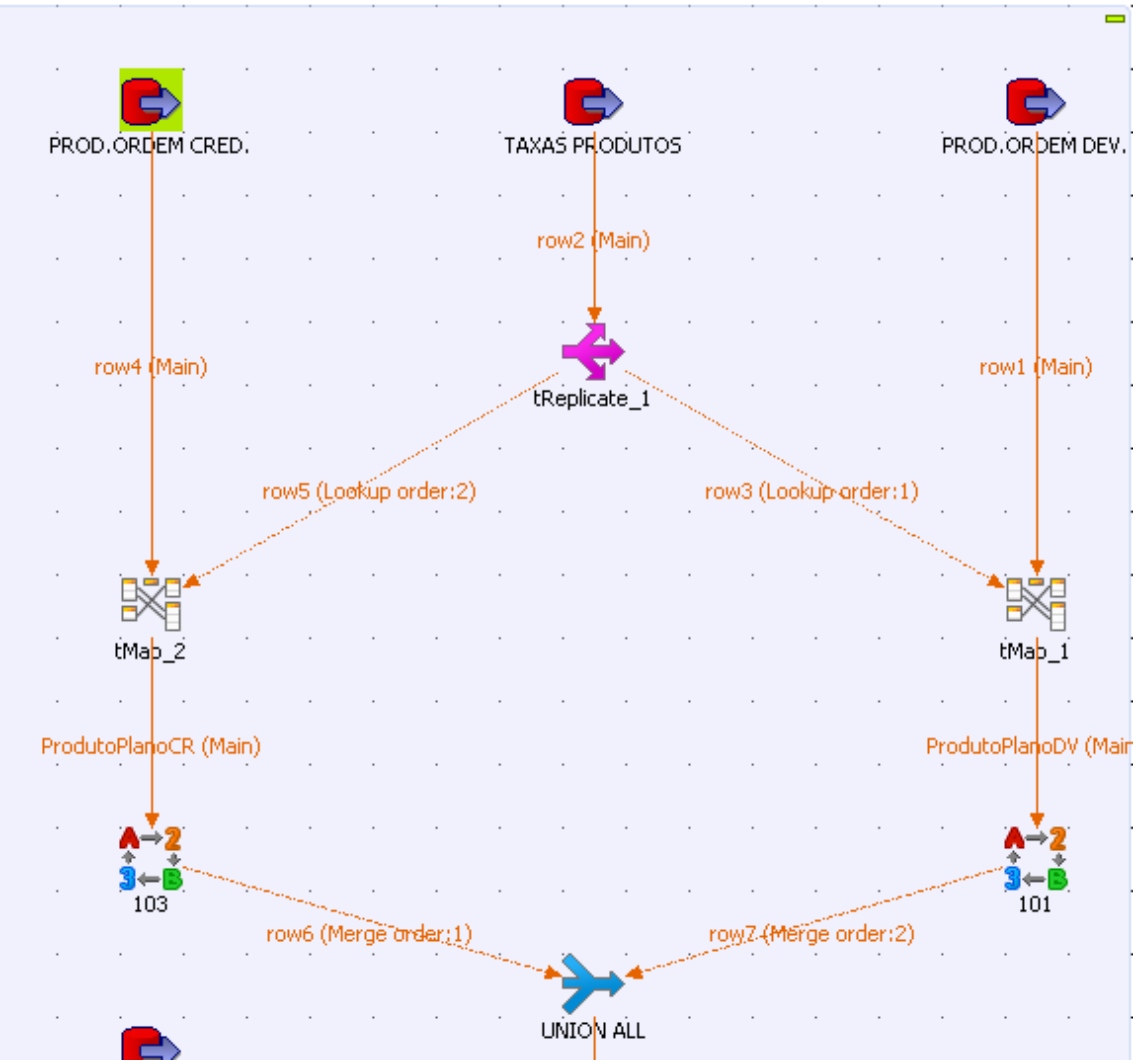

Figura 22 - Ilustração do processo de Data Flow realizado no Talend (1)

Passar-se-ão a explicar os procedimentos adoptados:

1. Neste caso começou por se efectuar a leitura de duas tabelas da BD, uma delas com dois acessos simultâneos. A que contém os produtos de origem, com 434 registos. Procedeu-se à replicação da outra tabela, que contém as taxas, contendo 1389 registos e realizou-se o "join" de ambas com recurso à ferramenta tMap do TL, tendo resultado 434 linhas.

2. De seguida adicionou-se um novo campo com o valor identificador das taxas credoras, ou devedoras e procedeu-se à união das duas tabelas em memória. O resultado foi de 868 linhas.

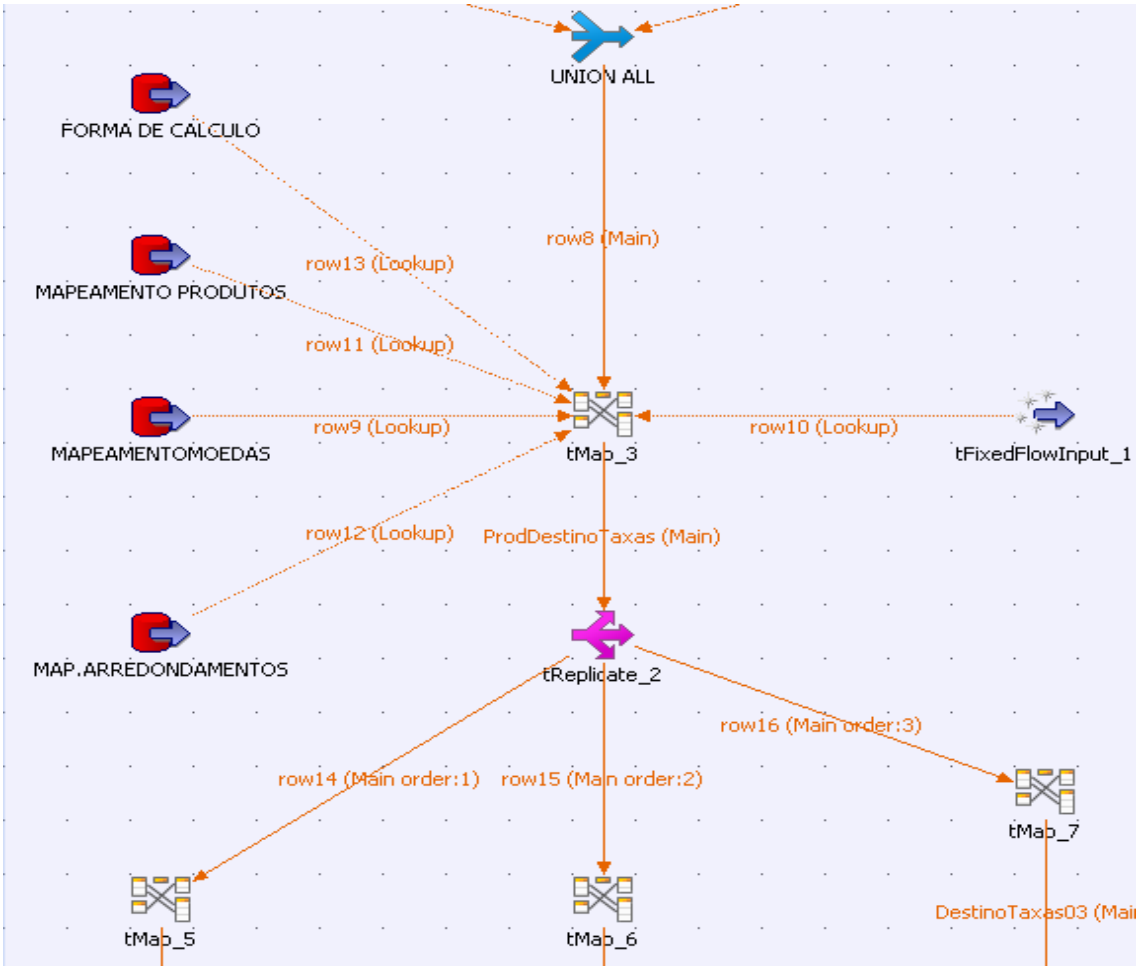

Figura 23 - Ilustração do processo de Data Flow realizado no Talend (2)

3. Após esta união, recorreu-se novamente à ferramenta tMap para efectuar o join de quatro tabelas lidas da BD e a uma tabela criada com novos campos a utilizar no decurso da migração. As tabelas são iguais às utilizadas anteriormente com o IS. Neste caso, porém temos a possibilidade inexistente no IS de efectuar o "join" e aplicar transformações a mais que duas tabelas. O resultado foi de 746 linhas, as mesmas que tínhamos no IS.

4. Procedeu-se à replicação do "data set" em três novas tabelas, cada uma com as taxas a aplicar por montante e finalmente procedeu-se à inserção dos registos na BD destino tendo sido inseridas 1644 linhas, com um tempo de execução de 2,406 segundos.

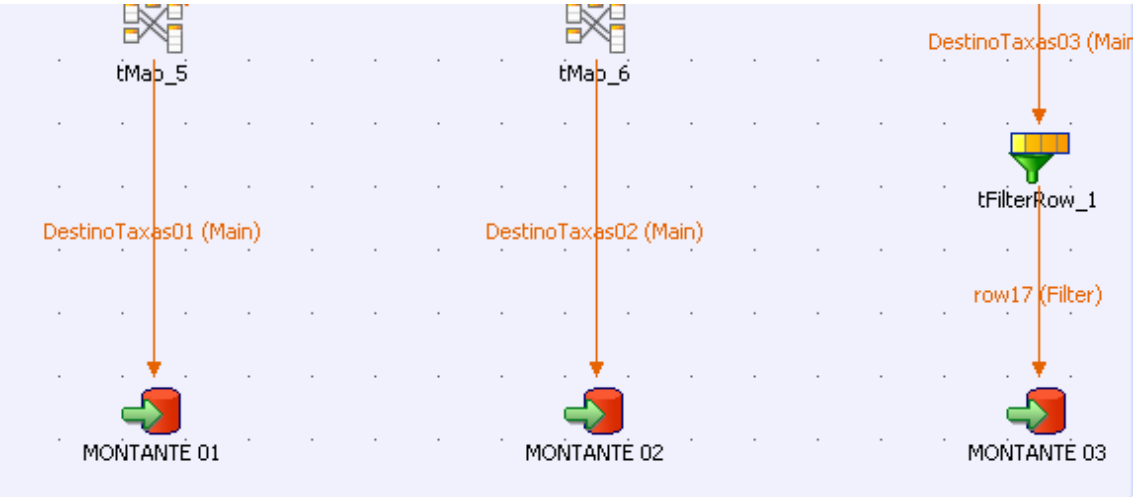

Figura 24 - Ilustração do processo de Data Flow realizado no Talend (3)

De seguida ir-se-á proceder à comparação das duas ferramentas, bastante agradáveis de utilizar, à luz da matriz apresentada anteriormente. A escala de valoração é a que se apresenta, em que 1 é o valor mais baixo e 5 o mais alto.

Definição	Valor numérico
1	Não corresponde às expectativas
2	Ligeiramente abaixo das expectativas
3	Corresponde às expectativas
4	Ligeiramente acima das expectativas
5	Excede as expectativas

Tabela 7 – Escala quantitativa da migração

Tem que ser referido que o teste de migração não foi suficientemente complexo para poder encontrar qualquer tipo de problema na utilização de qualquer das duas ferramentas, nem a experiência na utilização das mesmas é a bastante para que se considere a extrapolação desta análise para além do contexto em que foi realizada. Assim, passa-se à avaliação das mesmas.

Produto	**Int.Services**	**Talend**	
Valência / Critério			
Facilidade de utilização			**Peso relativo**
Arquitectura do produto	4	4	0.10
Extracção de dados	4	4	0.15
Transformação de dados	3	4	0.15
Carregamento de dados	4	4	0.15
Verificação da igualdade e fusão de dados	4	4	0.05
Gestão de meta dados	4	4	0.15
Ambiente de desenvolvimento	4	4	0.25
Total	**3.85**	**4**	
	Int.Services	**Talend**	
Flexibilidade			**Peso relativo**
Arquitectura do produto	3	4	0.05
Suporte a dados	4	4	0.15
Extracção de dados	4	4	0.15
Transformação de dados	4	4	0.15
Carregamento de dados	4	4	0.15
Verificação da igualdade e fusão de dados	4	4	0.05
Gestão de meta dados	4	4	0.20
Ambiente de desenvolvimento	4	4	0.10
Total	**3.95**	**4**	
	Int.Services	**Talend**	
Velocidade			**Peso relativo**
Arquitectura do produto	4	4	0.05
Extracção de dados	3	4	0.30
Transformação de dados	3	4	0.30

Carregamento de dados	3	4	0.30
Verificação da igualdade e fusão de dados	4	4	0.05
Total	**3.1**	**4**	
	Int.Services	**Talend**	
Robustez			**Peso relativo**
Arquitectura do produto	4	4	0.20
Suporte a dados	4	4	0.05
Extracção de dados	4	4	0.20
Transformação de dados	4	4	0.20
Carregamento de dados	4	4	0.15
Verificação da igualdade e fusão de dados	4	4	0.05
Gestão de meta dados	4	4	0.10
Ambiente de desenvolvimento	4	4	0.05
Total	**4**	**4**	
	Int.Services	**Talend**	
Escalabilidade			**Peso relativo**
Arquitectura do produto	3	3	0.10
Extracção de dados	4	4	0.20
Transformação de dados	4	4	0.20
Carregamento de dados	4	4	0.20
Verificação da igualdade e fusão de dados	4	4	0.05
Gestão de meta dados	4	4	0.20
Ambiente de desenvolvimento	4	4	0.05
Total	**3.9**	**3.9**	

Tabela 8 – Atribuição dos pesos relativos dos critérios de avaliação no processo de migração

Em suma chegou-se à avaliação seguinte:

	Int.Services	**Talend**	
Valência			**Peso relativo**
Custo	**1**	**5**	0.10
Facilidade de utilização	**3.85**	**4**	0.20
Flexibilidade	**3.95**	**4**	0.20
Robustez	**4**	**4**	0.20
Escalabilidade	**3.9**	**3.9**	0.20
Velocidade	**3.1**	**4**	0.10
Valor final	**3.55**	**4.08**	

Tabela 9 – Valor final da avaliação das ferramentas de ETL

É de referir que este valor final apresentado tem a sua maior dependência da variável custo. Também a velocidade, no caso da ferramenta Talend em que a migração foi efectuada em Java, poderá apresentar outros valores se executada sobre um "script" Perl, funcionalidade existente. Apesar disto verificaram-se valores melhores que com o IS, o que pode ser o reflexo da forma como foi executado o processo.

Assim, no caso concreto, esta migração é perfeitamente exequível com o recurso a uma ferramenta de código fonte livre e de utilização gratuita em ambiente monoutilizador.

Conclusão

Os processos de migração de bases de dados podem ser extremamente morosos e complicados. No entanto, existe solução: O bom planeamento, e a utilização de metodologias e ferramentas adequadas, reduzem o prazo e a complexidade. Devem promover canais de comunicação, a fragmentação de tarefas complexas e o encadeamento correcto das actividades a desenvolver.

A quantidade de migrações bem sucedidas, dentro do prazo e orçamento, tem aumentado, com as contribuições de vários autores que estudaram o tema e da aplicação das suas metodologias. Assim, consoante o grau de complexidade concluímos poder ser de maior utilidade determinada abordagem.

A preservação digital dos dados, problema recente, assume grande importância numa óptica de longo prazo e deve ser contemplada. A criação de repositórios baseados em normas internacionais que permitam o acesso continuado à informação deve ser visto como acto primordial de gestão e a sua criação realizada.

Abordou-se também o processo de automatização das migrações com recurso a ferramentas próprias para o efeito e verificaram-se os benefícios colhidos. A sua utilização poderá ser parcial ou integral no decorrer do processo de migração, agilizando-o e reduzindo o seu prazo e falhas.

Por fim, procedeu-se à comparação de três destas ferramentas na execução de uma migração concreta. Uma foi eliminada à cabeça por falta de requisitos e as outras mostraram-se bastante agradáveis de utilizar, semelhantes no uso, e total capacidade de execução do processo, diferindo essencialmente no custo da sua utilização.

Conclui-se assim que, se em tempos imemoráveis se podia considerar a migração de dados como tal, houve de lá para cá grandes alterações quanto ao conteúdo e não menos quanto à forma. Verifica-se actualmente que os dados estão bastante melhor preservados, são mais controláveis, mais migráveis e com boas perspectivas para os tempos futuros. As ferramentas e metodologias de que dispomos assim o ditam.

Bibliografia

Abu-Hamdeh, R., Cordy, J. & Martin, P. (1994). Schema translation using structural transformation. *Proceedings of the 1994 conference of the Centre for Advanced Studies on Collaborative research*, (pp. 123-43), IBM Press.

Anthony, R.N.(1965). Planning and Control Systems: A Framework for Analysis, Cambridge, Harvard University Press,

A. Carzaniga, A. Fuggetta, R. S. Hall, D. Heimbigner, A. v. d. Hoek, and A. L. Wolf (1998). "A Characterization Framework for Software Deployment Technologies", Technical Report Department of Computer Science, Colorado, University of Colorado.

Dearle, Alan (2007).Software Deployment, Past, Present and Future, Future of Software Engineering(FOSE'07), IEEE

Batini, C., Lenzerini, M. & Navathe, S. B. (1986). A Comparative Analysis of Methodologies for Database Schema Integration. ACM Computing Surveys, 18, 4, 23-64.

Behm, Andreas & Geppert, Andreas & Dittrich, Klaus R. (1997), On the migration of relational schemas and data to object-oriented database systems, *In Proc. Of the 5th International Conference on Re-Technologies in Information Systems,* Klagenfurt, Austris, p. 2.

Brodie, M. & Stonebraker, M. (1995). Migrating Legacy Systems: Gateways, Interfaces, and the Incremental Approach ,San Francisco, Morgan Kaufmann.

Chandras (2008), Disponível online em: http://www.intelligententerprise.com/blog/archives/2008/01/elt_vs_etl_much.html , Último acesso a 06.09.2008.

Chen, P. (1976) – "The Entity-Relationship Model – Towards a Unified View of Data", *Association for Computer Machinery Transaction on Database Systems*, 1, 1, pp 9-36.

Consultative Committee for Space Data Systems (2002). "Reference Model for an Open Archival Information System (OAIS) – Blue Book", Washington, National Aeronautics and Space Administration.

Eckerson, Wayne (2003a). ETL Industry Expanding as Data Integration Platforms Emerge. Disponível online em: http://www.tdwi.org/display.aspx?id=6617 , Último acesso a 03.09.2008.

Eckerson, Wayne (2003b). Understanding Business Intelligence. Disponível online em: http://www.tdwi.org/research/display.aspx?ID=6838 , Último acesso a 03.09.2008.

Elmasri, R & Navathe, S. B. (1984). Object Integration in Database Design. *Proceedings of* IEEE *Conference on Data Engineering*. Los Angeles.

Forrester (2007), www.forrester.com, Market Overview: Open Source ETL Tools - An Attractive Alternative to Custom Code. Disponível online em: http://www.tdwi.org/Marketplace/Whitepaper.aspx?pID=569 , Último acesso a 03.09.2008.

Garcia, Marie L. e Bray, Olin H. (1997), Fundamentals of Technology Roadmapping, Albuquerque, Sandia National Laboratories.

Gonsalves (2008), Disponível online em: http://www.intelligententerprise.com/showArticle.jhtml?articleID=206904832, Último acesso a 04.09.2008.

Hasselbring, W., Reussner, R., Schlegelmilch, J., Teschke, T., & Krieghoff, S. (2004). The Dublo Architecture Pattern for Smooth Migration of Business Information Systems: An Experience Report. *Proceedings of the 26th International Conference on Software Engineering* (ICSE '04), 117-26.

Housel, Barron C., Lum, V. & Shu, N. (1974). Architecture to an Interactive Migration System. *Proceedings of the 1974 ACM SIGFIDET (now SIGMOD) workshop on Data description, access and control,* pp. 157-169, New York, ACM Press.

Hudicka, J. R. (1999). The Complete Data Migration Methodology, Disponível online em: http://www.dulcian.com , Último acesso a 13.06.2008.

IBM Notícias, "*IBM lança Mainframe 'System z10 E'"*, Disponível online em: http://www.ibm.com/news/pt/pt/2008/02/26.html , Último acesso a 24.06.2008

Kelly, C. & Nelms, C. (2003).Roadmap to checking data migration. *Computers & Security,* 22, 6, 506-510.

Macrosoft, 2007, Disponível online em: http://www.macrosoftinc.com/mainframe_dev.html , Último acesso a 23.06.2008.

Miguel Ferreira (2006), Introdução à preservação digital - Conceitos, estratégias e actuais consensos, Guimarães, Escola de Engenharia da Universidade do Minho.

MIKE2.0, Disponível online em: http://www.openmethodology.org/wiki/MIKE2.0_Methodology , Último acesso a 31.08.2008.

Moriarty, T. & Hellwege, S. (1998). Data migration, *Database Programming & Design*, pp 11-14.

Navathe, S. B., Elmasri, R. & Larson, J. (1986). Integrating User Views in Database Design., *Computer,* 19, 1, pp 50-62.

Oliveira, A (1994). O valor da informação, *Sistemas de Informação: Revista da Associação Portuguesa de Sistemas de Informação*, 2, pp 39-56.

OMG (2003). Specification for Deployment and Configuration of Component-based Distributed Applications, Disponível online em: http://www.omg.org/docs/mars/03-05-08.pdf , Último acesso a 03.09.2008.

Pereira, José Luis(1998). Tecnologia de bases de dados (3ª ed.). Lisboa: FCA – Editora de informática.

PMBOK(2004).. Guide to the Project management body of knowledge(3ª ed. Pennsylvania – Project Management Institute, Inc.

J. Ramalho e M. Ferreira (2008), *Sistemas de suporte à Governação Electrónica*, Amarante, Seminário de Governação Digital.

Ribas,F.G-P. (1989). Estruturas Organizativas e Informação na Empresa, Porto, Editorial Domingos Barreira.

Scott, Henry. Hoon, Sherlynn. Hwang, Meeky. Diane, Lee. D. DeVore, Michael (2005), Engineering Trade Study: Extract, Transform, Load Tools for Data Migration, University of Virginia, Charlottesville, Disponível online em: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1497124 , Último acesso a 03.09.2008.

SearchDataManagement (2007), Disponível online em: http://searchdatamanagement.techtarget.com/news/article/0,289142,sid91_gci1241492,00.html, Último acesso a 04.09.2008.

Silva e Correia (2004), Técnicas para Construção de Testes Funcionais Automáticos. Disponível online em: http://isg.inesc-id.pt/alb/static/papers/2004/n12-sc-Quatic2004.pdf , Último acesso a 28.08.2008.

Sommerville, Ian (2000). Software Engineering. Boston, Addison-Wesley(6th. Edition).

Stern, Disponível online em: http://www.stern.com.br/suporte/documentos/MigracaodeBancosdeDados.pdf , Último acesso a 23.07.2008.

Stonebraker, Michael, Moore, Dorothy (1995). Object-relational DBMS - The Next Great Wave, San Francisco, Morgan Kaufmann Publishers Inc

Varajão, João E. Q. (1998), A Arquitectura da Gestão de Sistemas de Informação (3ª ed.). Lisboa: FCA – Editora de informática.

WinRunner information: Disponível online em: http://www.mercury.com/us/products/qualitycenter/functionaltesting/winrunner/ , Último acesso a 07.09.2008.

Youn, Cheong & Ku, Cyril S. (1992). Database Migration. IEEE

Zurek, Bob (2008), Open Source ETL Conversion Tool ?, Disponível online em: http://www.ibm.com/developerworks/blogs/page/BobZurek?entry=open_source_etl_conversion_tool , Último acesso a 03.09.2008.