

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ERNANDO SILVA

**Um ambiente de Business Intelligence para
apoio a decisões sobre o ciclo de vida de
serviços**

Dissertação apresentada como requisito parcial
para a obtenção do grau de Mestre em Ciência
da Computação

Prof^a. Dr. Renata Galante
Orientadora

Prof^a. Dr. Karin Becker
Co-orientadora

Porto Alegre, junho de 2013.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Silva, Ernando

Um ambiente de Business Intelligence para apoio a decisões sobre o ciclo de vida de serviços [manuscrito] / Ernando Silva. – 2013.

77 f.: il.

Orientador: Renata Galante; Co-orientador: Karin Becker.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2013.

1.Evolução de serviços web. 2.Impacto da mudança 3.Business Intelligence. I. Galante, Renata. II. Becker, Karin. III. Um ambiente de Business Intelligence para apoio a decisões sobre o ciclo de vida de serviços.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Pós-Graduação: Prof. Vladimir Pinheiro do Nascimento

Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb

Coordenador do PPGC: Prof. Luigi Carro

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

“Only the ideas that we really live have any value.”
— HERMANN HESSE

AGRADECIMENTOS

Agradeço inicialmente a Deus, pela minha vida, pela minha saúde, pela proteção e pelas pessoas que coloca no meu caminho.

Meus agradecimentos se estendem aos meus pais, Eduardo e Jozenilda, e aos meus irmãos, Edvan e Erika, que sempre foram constante fonte de apoio, inspiração e compreensão em todas as etapas da minha vida.

Também deixo os meus sinceros agradecimentos às minhas orientadoras, Renata e Karin, que me auxiliaram imensamente durante o período de mestrado, sempre pacientes para orientar e tirar dúvidas, o que foi primordial para o êxito deste trabalho. Os ensinamentos que obtive vão além do resultado deste trabalho, me auxiliando também na evolução como pesquisador e como pessoa.

Adicionalmente, também agradeço ao Programa de Pós-Graduação em Computação do Instituto de Informática da Universidade Federal do Rio Grande do Sul, que proporcionou a oportunidade de realização do mestrado e forneceu toda a estrutura para o desenvolvimento do mesmo. Sem dúvida foi uma grande experiência para mim. Também agradeço ao CNPq pela bolsa e apoio financeiro.

Não posso deixar de agradecer também aos amigos de laboratório: Bruno Vollino, Bruno Rezende, Edimar, Guilherme, Gustavo, Isabel, Marcelo, Maurício, Miriam, Renato, Solange e Tiago. Todos me acolheram muito bem desde os primeiros dias de mestrado, e isso ajudou bastante na minha adaptação à rotina do mestrado e à vida em Porto Alegre. Em relação à adaptação, também agradeço à Daruichi, grande amiga que me apoiou nesse período de adaptação em terras gaúchas.

Também agradeço aos professores com os quais tive contato desde a educação básica, passando pelo ensino médio, técnico e período de graduação. De alguma forma, todos contribuíram para o caminho que me levou a concluir esta etapa da minha vida. Meu agradecimento especial ao Prof. Luciano Digiampietri, meu orientador de TCC no meu período de graduação e grande incentivador do meu projeto de seguir para o mestrado.

Por fim, dirijo os meus agradecimentos aos amigos de longa data que, mesmo com a distância, sempre me deram bastante apoio e incentivo. Se eu citasse cada nome, provavelmente acabaria esquecendo alguns importantes pra mim. Portanto, deixo o meu muito obrigado a todos que, direta e indiretamente, contribuíram em momentos importantes da minha vida. Obrigado por tudo.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	9
LISTA DE FIGURAS	11
LISTA DE TABELAS	13
RESUMO	15
ABSTRACT	17
1 INTRODUÇÃO	19
1.1 Cenário Motivador	21
1.2 Contextualização.....	22
1.3 Objetivo	23
1.4 Organização do Texto	24
2 FUNDAMENTAÇÃO TEÓRICA	25
2.1 Serviços Web	25
2.2 Versionamento e Compatibilidade.....	26
2.3 Tipos de Mudança em Serviços Web	27
2.4 Business Intelligence (BI).....	28
2.5 Data Warehousing	30
2.5.1 Data Warehouse.....	30
2.5.2 Extração, Transformação e Carga de Dados (ETL)	31
2.5.3 Camada de Apresentação.....	31
2.6 Considerações	32
3 TRABALHOS RELACIONADOS	35
3.1 Gestão da Evolução de Serviços	35
3.2 Stakeholders	37
3.2.1 Stakeholders na abordagem de Treiber et al. (2008).....	37
3.2.2 Stakeholders em abordagens SOA	38
3.2.3 Consolidação dos Stakeholders	39
3.3 Governança	40
3.4 Necessidades Relacionadas à Tomada de Decisão	41

3.5	Considerações Finais	43
4	O GERENCIADOR DE APOIO À DECISÃO	45
4.1	Visão Geral.....	45
4.2	Integração com os componentes do framework WS-EVOLV.....	47
4.2.1	Integração com o Gerenciador de Versões	47
4.2.2	Integração com o Gerenciador de Perfis.....	47
4.3	O Ciclo de Vida de BI.....	48
4.4	Considerações Finais	48
5	MODELAGEM DIMENSIONAL DO GERENCIADOR DE APOIO À	
DECISÃO		51
5.1	Visão Geral.....	51
5.2	Identificação de KPIs	51
5.2.1	KPIs de Uso	52
5.2.2	KPIs Financeiros	53
5.3	Identificação da Granularidade dos Dados.....	53
5.3.1	Dados de Uso.....	53
5.3.2	Dados Financeiros	54
5.4	Modelagem Dimensional.....	56
5.5	Considerações Finais	60
6	ARQUITETURA ETL.....	61
6.1	Visão Geral.....	61
6.2	Identificação de Fontes de Dados.....	61
6.3	Utilizando Wrappers para a Extração dos Dados	63
6.4	Carga das Dimensões e Fatos	63
6.4.1	Dimensões	64
6.4.2	Tabelas Fato.....	64
6.5	Considerações Finais	65
7	ESTUDO DE CASO	67
7.1	Cenário AWS	67
7.2	Dados Sintéticos	68
7.3	Implementação.....	69
7.4	Recursos Analíticos	69
7.5	Cenário 1: Análise do Impacto de Mudanças Incompatíveis	70
7.6	Cenário 2: Análise da Rentabilidade de um Serviço	71
7.7	Considerações Finais	72
8	CONCLUSÕES E TRABALHOS FUTUROS	73
	REFERÊNCIAS.....	75

LISTA DE ABREVIATURAS E SIGLAS

SLA	Service-level Agreement
SaaS	Service as a Service
SOA	Service Oriented Architecture
AWS	Amazon Web Services
QoS	Quality of Service
BI	Business Intelligence
DW	Data Warehouse
WS-EVOLV	Web Service - Evolution
SOAP	Simple Object Access Protocol
WSDL	Web Service Description Language
VIDs	Version Identifiers
XML	eXtensible Markup Language
KPI	Key Performance Indicator
ETL	Extract, Transform and Load
MTFS	Multiple Fact Table Schema
OLAP	OnLine Analytical Processing
UDDI	Universal Description, Discovery and Integration
ERP	Enterprise Resource Planning
CRM	Customer Relationship Management

LISTA DE FIGURAS

Figura 1.1: Exemplo de um portfólio de serviços baseado no cenário da AWS.	19
Figura 1.2: Visão geral do framework WS-EVOLV.	23
Figura 2.1: Estrutura organizacional para a execução de operações.	29
Figura 2.2: Fluxo das informações em um DW até a camada de apresentação.	32
Figura 3.1: O ciclo de mudanças na abordagem de Treiber et al. (2008).	38
Figura 4.1: O Gerenciador de Apoio à Decisão no <i>framework</i> WS-EVOLV.	45
Figura 4.2: A arquitetura de BI do Gerenciador de Apoio à Decisão.	46
Figura 4.3: O ciclo de BI adaptado para a gestão da evolução de serviços.	48
Figura 5.1: Etapas para a elaboração do modelo dimensional.	51
Figura 5.2: O modelo MTFS proposto para o DW.	57
Figura 5.3: A dimensão CLIENTE relacionada às duas tabelas fato.	58
Figura 5.4: A dimensão STATUS_DO_SERVICO relacionada às duas tabelas fato. ...	58
Figura 5.5: A dimensão SERVICO_POR_VERSAO relacionada à FATO_FINANCEIRA e a dimensão SERVICO_POR_OPERACAO relacionada à fato de uso.	59
Figura 5.6: A dimensão TEMPO relacionada às duas tabelas fato.	60
Figura 5.7: A dimensão TIPO_DE_COBRANCA relacionada apenas a uma fato.	60
Figura 6.1: Visão geral da arquitetura ETL.	61
Figura 6.2: A área de fontes de dados da arquitetura ETL.	62
Figura 6.3: Os <i>wrappers</i> e a área de <i>staging</i> da arquitetura ETL.	63
Figura 6.4: A área de carga das dimensões e tabelas fato da arquitetura ETL.	64
Figura 7.1: <i>Dashboard</i> relacionando indicadores financeiros e de uso.	70
Figura 7.2: Tabela <i>pivot</i> relacionando as versões que apresentam as piores relações entre consumo e lucratividade.	71

LISTA DE TABELAS

Tabela 3.1: Relação entre os stakeholders definidos pelos trabalhos relacionados.....	39
Tabela 3.2: Relação entre as fases da metodologia para ciclo de vida orientado a mudança, governança de serviços e os <i>stakeholders</i> em serviços <i>web</i>	42
Tabela 5.1: KPIs considerando a perspectiva de uso e financeira dos serviços.	52
Tabela 5.2: Métodos de cobrança utilizados em alguns serviços do portfólio AWS.	54
Tabela 5.3: Escopo e granularidade das métricas financeiras e de uso dos serviços.	56
Tabela 5.4: Dimensões no contexto de serviços.....	57
Tabela 7.1: Os serviços AWS selecionados para o estudo de caso.	67
Tabela 7.2: Requisições Diretas e Indiretas de cada Grupo de Cliente	68

RESUMO

Com o crescimento da demanda por aplicações orientadas a serviços, a complexidade da gestão de mudanças em serviços aumenta cada vez mais. Nesse contexto, provedores de portfólios de serviços têm como objetivo a minimização do impacto de tais mudanças para os seus clientes. Os trabalhos existentes abordam essencialmente a mudança sob o viés técnico, lidando com técnicas de versionamento e determinação de compatibilidade. No entanto, os provedores possuem dilemas de decisão que vão além dos aspectos técnicos, abrangendo o impacto na perspectiva de negócios, considerando clientes afetados, receitas, custos, violações de SLA, entre outros.

Este trabalho faz parte de um *framework* voltado para a análise da evolução de serviços considerando a perspectiva de uso dos mesmos. O presente trabalho propõe o uso de técnicas de *Business Intelligence* e *Data Warehousing* para apoiar a tomada de decisões orientadas a negócios na gestão do ciclo de vida de serviços. Em tal contexto, considera-se o cenário de portfólio de serviços e mudanças profundas, abrangendo cadeias de serviços e clientes diretos/indiretos. Este trabalho identifica e analisa requisitos de decisão, juntamente com indicadores financeiros e de uso relacionados à provisão de serviços. Um *data warehouse* é modelado para prover uma visão unificada e integrada de tais indicadores, de acordo com diferentes perspectivas de análise. Uma arquitetura para a extração, transformação e carga dos dados também é apresentada com a proposição de alternativas para lidar com a integração e heterogeneidade das fontes de dados. Através de um estudo de caso inspirado em um cenário real, demonstra-se como a análise do impacto é analisada através da abordagem proposta.

Palavras-Chave: Evolução de serviços web, impacto da mudança, business intelligence.

A Business Intelligence environment to support decisions on the service lifecycle

ABSTRACT

With the growing demand for service-oriented applications, the complexity of service change management is increasing. In this context, service providers aim at the minimal impact of changes to its clients. Existing work essentially addresses change decisions from a technical perspective, with versioning techniques and compatibility assessments. However, providers have decisional dilemmas that go beyond the technical aspects, addressing the business impact in terms of clients affected, revenues, costs, SLA penalties, among others.

This work integrates a framework addressing the service evolution analysis based on service usage perspective. We propose the use of Business Intelligence and Data Warehousing techniques to support business-oriented decisions on the service life-cycle management. Our approach covers service portfolios and the deep change context, i.e. services consumed in large scale and direct/indirect clients. This work is centered on the identification and analysis of decisional requirements and financial and usage indicators related to the service provision business. A data warehouse is modeled to provide a unified and integrated view of these indicators, according to different analysis perspectives. An architecture to extract, transform and loading data also is presented, proposing alternatives to lead with the integration and the heterogeneity of the data sources. Through a case study inspired by a real world scenario, we demonstrate how change impact is analyzed according our approach.

Keywords: web service evolution, change impact, business intelligence.

1 INTRODUÇÃO

Um serviço *web* é um *software* desenvolvido para apoiar a comunicação entre diferentes sistemas através da *web*, recebendo requisições e retornando respostas segundo uma interface bem definida, processável por máquina e independente de plataforma (W3C, 2011). Serviços *web*, referenciados a partir deste ponto apenas como serviços, constituem uma das principais tecnologias para a implantação de aplicações baseadas em SOA (*Service-oriented Architecture*). A utilização de aplicações SOA fornece diversas vantagens, como o reuso e o baixo acoplamento. Essas características ajudam a reduzir custos na criação de processos de negócios e na integração entre aplicações distintas (BENATALLAH *et al.*, 2006). Nos últimos anos, a demanda por aplicações SOA tem aumentado consideravelmente e estima-se um crescimento ainda maior em um curto prazo. Atualmente, muitas organizações (*e.g.* Amazon, Microsoft, Salesforce e AT&T) possuem importantes segmentos de negócios focados em prover soluções baseadas no paradigma de SaaS (*Software as a Service*). Tipicamente, o modelo de negócio baseado em SaaS envolve um portfólio de serviços inter-relacionados e, conseqüentemente, abrange uma grande cadeia de clientes, os quais podem consumir distintos serviços e integrar os mesmos em suas próprias aplicações.

A Figura 1.1 exemplifica um portfólio de serviços baseado no portfólio da AWS¹. As setas indicam relações de consumo, tanto em relação a grupos de clientes que consomem serviços específicos, quanto a serviços que dependem de operações de outros serviços. Cada grupo representa um conjunto de milhares de clientes distintos, os quais podem consumir os serviços diretamente (*i.e.*, fazendo requisições diretas para um serviço) ou indiretamente (*i.e.*, fazendo requisições para um serviço que vai consumir outro serviço, que então retornará a resposta para o cliente).

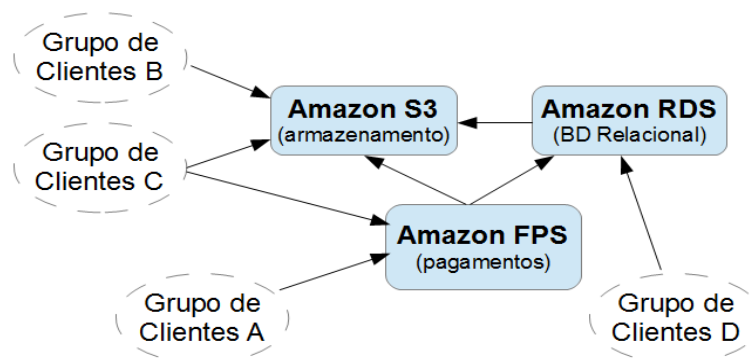


Figura 1.1: Exemplo de um portfólio de serviços baseado no cenário da AWS.

¹ <http://aws.amazon.com/>

Assim como os softwares tradicionais, os serviços *web* tendem a sofrer mudanças durante o seu ciclo de vida. O termo evolução de serviços remete ao processo de mudança para adicionar novas funcionalidades, atender a requisitos de mudanças ou melhorar aspectos de QoS (LEITNER *et al.*, 2008). O processo de evolução leva a mudanças estruturais, semânticas ou não-funcionais em um serviço. Essas mudanças podem ser incompatíveis para os clientes existentes, o que significa que a aplicação cliente baseada na versão antiga deixa de funcionar após a implantação da mudança. Mudanças incompatíveis consequentemente podem afetar clientes, tanto diretamente (i.e., impactando clientes que fazem requisições diretas para o serviço alterado), quanto indiretamente (i.e. para clientes que fazem requisições indiretas para o serviço alterado, consumindo diretamente um serviço que depende do serviço alterado). As mudanças limitadas a um serviço e seus clientes diretos são conhecidas como *mudanças superficiais*, ao passo que mudanças que abrangem uma cadeia de serviços e seu conjunto de clientes diretos e indiretos são conhecidas como *mudanças profundas* (PAPAZOGLU; ANDRIKOPOULOS; BENBERNOU, 2011). Mudanças profundas são particularmente críticas, pois podem causar severas consequências financeiras e de reputação para provedores de serviços em larga escala, justamente pelo risco do efeito cascata da mudança através da cadeia de clientes e serviços inter-relacionados.

Para evitar os efeitos de mudanças incompatíveis, provedores adotam o versionamento, que representa a criação de diferentes versões de um mesmo serviço. O versionamento, assim como a compatibilidade, o uso de *proxies* para redirecionamento, e outras abordagens voltadas para mudanças superficiais, são aspectos técnicos de mudanças em serviços que são tratados pela maioria dos trabalhos existentes (FANG *et al.*, 2007), (FRANK *et al.*, 2008), (LEITNER *et al.*, 2008), (BECKER *et al.*, 2008), (ANDRIKOPOULOS; PAPAZOGLU; BENBERNOU, 2012).

Por outro lado, manter diversas versões de um serviço implica em maiores custos operacionais, o que onera o provedor de serviços. Lidar com mudanças, na perspectiva de um provedor de serviços, significa estimar o impacto das decisões para optar pela alteração do serviço, o que vai além da perspectiva técnica e envolve o alinhamento com as estratégias de negócios. Sob esse ponto de vista, apenas saber sobre incompatibilidades entre versões de serviço não é suficiente: é necessário compreender o que essas mudanças representam para os clientes em termos de impacto direto e indireto, e como isso pode afetar aspectos como o lucro, a reputação e a posição de mercado do provedor. Encontrar as respostas que permitam esse entendimento, especialmente no contexto de mudanças profundas, não é uma tarefa trivial.

Papazoglou, Andrikopoulos e Benbernou (2011) apresentam uma metodologia para o ciclo de vida de serviços orientado a mudanças profundas. Tal ciclo é composto por fases e atividades que visam compreender a necessidade, o impacto e a implantação da mudança em um portfólio de serviços. Também na perspectiva de atividades relacionadas ao processo de mudança, Treiber *et al.* (2009) relacionam os tipos de mudança, atividades e *stakeholders* em serviços, indicando os efeitos da mudança para cada um. Considerando a linha de gestão de serviços, o conceito de governança de serviços é abordado em alguns trabalhos. Governança abrange a criação, a gestão e a aplicação de políticas e padrões para os processos relacionados à gestão do ciclo de vida dos serviços, destacando áreas para as quais as políticas e padrões devem existir para apoiar a tomada de decisão (WOOLF, 2006). Apesar de esses trabalhos destacarem necessidades de decisão e atividades que estão relacionadas à evolução de serviços, há uma carência de abordagens que tratem o apoio à tomada de decisão inerente ao

processo de mudança, especialmente considerando a perspectiva estratégica de um provedor de portfólio de serviços em grande escala.

Business Intelligence (BI) tem sido aplicada para apoiar a tomada de decisão em diversos segmentos (LÖNNQVIST; PIRTTIMÄKI, 2006), mas os trabalhos existentes não têm explorado esse potencial para apoiar as decisões inerentes à evolução de serviços. BI remete ao uso de dados organizacionais internos e externos para o apoio à tomada de decisão, transformando dados em informação, e informação em conhecimento (GOLFARELLI *et al.*, 2004). Uma abordagem comum para BI, adotada por grande parte da indústria, envolve o fornecimento de recursos analíticos por meio de uma base de dados centralizada, confiável, integrada, histórica e orientada a assunto (CHEN; CHIANG; STOREY, 2012) (KIMBALL; ROSS, 2002). Tal base de dados é conhecida como *Data Warehouse* (DW). Alguns dos desafios inerentes ao uso de BI para apoiar as decisões relacionadas ao ciclo de vida de serviços incluem: a) caracterização das necessidades de decisão em tal cenário; b) identificação de dados relevantes sobre a provisão e o consumo de serviços (e seus respectivos clientes), organizando estas informações em um modelo dimensional apropriado, que seja capaz de apoiar a análise em diferentes perspectivas; e c) proposição de uma arquitetura flexível capaz de lidar com a heterogeneidade das fontes de dados, que podem variar de acordo com o ambiente do provedor de serviços e com as suas práticas de negócios.

1.1 Cenário Motivador

Como cenário motivador, ilustra-se um ambiente fictício inspirado no portfólio AWS apresentado na Figura 1.1, que abrange serviços de *e-commerce* (*Amazon FPS*), bases de dados relacionais (*Amazon RDS*) e armazenamento (*Amazon S3*). Os clientes podem consumir tais serviços através de suas próprias aplicações ou utilizando esses serviços para compor seus próprios serviços. A partir desse ambiente é possível detectar casos que mostram as necessidades de um provedor de serviços.

Por exemplo, suponha que o provedor deseje compreender os efeitos de mudanças incompatíveis em operações de uma versão do serviço FPS. Analisando apenas informações sobre o volume de requisições, o provedor descobre que todas as operações recebem um alto número de requisições dos grupos de clientes A e C da Figura 1.1. Isso pode induzir o provedor a pensar que a mudança não deva ser realizada, pois o impacto seria proporcional ao consumo. Entretanto, caso o provedor conhecesse os padrões de uso de seus clientes, ele poderia detectar, por exemplo, que apenas o grupo de clientes A seria mais afetado, enquanto o grupo C não sofreria grandes impactos, pelo fato deste último consumir operações do serviço FPS que não seriam afetadas pela mudança. Cruzando esse tipo de informação com informações financeiras, como a receita gerada por grupos de clientes estrategicamente importantes para a organização, o provedor teria valiosos indícios para decidir sobre a mudança, alinhando ao mesmo tempo perspectivas estratégicas e de negócios.

Suponha agora outra situação, na qual o provedor nota uma queda no seu lucro e ele precisa reverter essa situação. O provedor considera duas alternativas: aumentar o preço cobrado pelo fornecimento dos serviços ou reduzir os custos de provisão por meio da desativação de versões antigas e/ou pouco consumidas. Enquanto a primeira alternativa pode representar uma desvantagem competitiva e a perda de clientes para os concorrentes que não aumentaram preços, a segunda alternativa representa uma possibilidade de realocar recursos de infraestrutura de maneira mais eficiente,

desonerando o provedor de serviços com a desativação de versões em desuso. Mas determinar qual versão deve ser desativada, considerando o menor impacto possível, não é uma tarefa trivial. O provedor de serviços necessita de informações para avaliar a melhor opção e decidir, considerando também os clientes indiretos que podem afetados e o impacto que isso pode gerar.

Os casos supracitados reforçam a necessidade de mecanismos que possibilitem que provedores de serviços façam análises correlacionando informações de diferentes naturezas, oriundas do contexto financeiro e de uso dos serviços, por exemplo, para apoiar decisões estratégicas e determinar ações frente a cenários de mudanças profundas.

1.2 Contextualização

O presente trabalho faz parte de um *framework* voltado para a análise da evolução de serviços considerando a perspectiva de uso. O *framework* WS-EVOLV (*Web Service Evolution*) surgiu com o objetivo de ser uma base para a gestão de mudanças na evolução de serviços. Ao invés de considerar o cenário tradicional de pior caso de impacto da mudança, o WS-EVOLV baseia-se no impacto real da mudança, i.e., considerando que mudanças podem afetar grupos de aplicações diferentemente, dependendo do uso que é feito do serviço.

A Figura 1.2 ilustra o *framework* e os seus componentes presentes no ambiente do provedor de serviços. A partir das requisições de clientes, a captura de dados relacionados às requisições serve como entrada para algoritmos que identificam padrões de uso e de compatibilidade a partir dos tipos de uso. Os componentes estão relacionados e são classificados como:

- Gerenciador de Versões (*Version Manager*), que realiza a avaliação automática de mudanças entre versões de um serviço, determinando a compatibilidade em nível de *feature* (i.e., serviço, operação ou tipo) (YAMASHITA et al., 2012) (YAMASHITA; BECKER; GALANTE, 2012);
- Gerenciador de Perfis (*Profile Manager*), que utiliza técnicas de mineração de dados para agrupar clientes de acordo com os padrões de uso dos serviços (VOLLINO; BECKER, 2013);
- Gerenciador de Uso (*Usage Manager*), que é voltado para a análise do impacto das mudanças em relação aos perfis de uso (VOLLINO; BECKER, 2013);
- Gerenciador de Apoio à Decisão (*Decision Support Manager*), que abrange o apoio à tomada de decisão na gestão da evolução de serviços e que é proposto no presente trabalho.

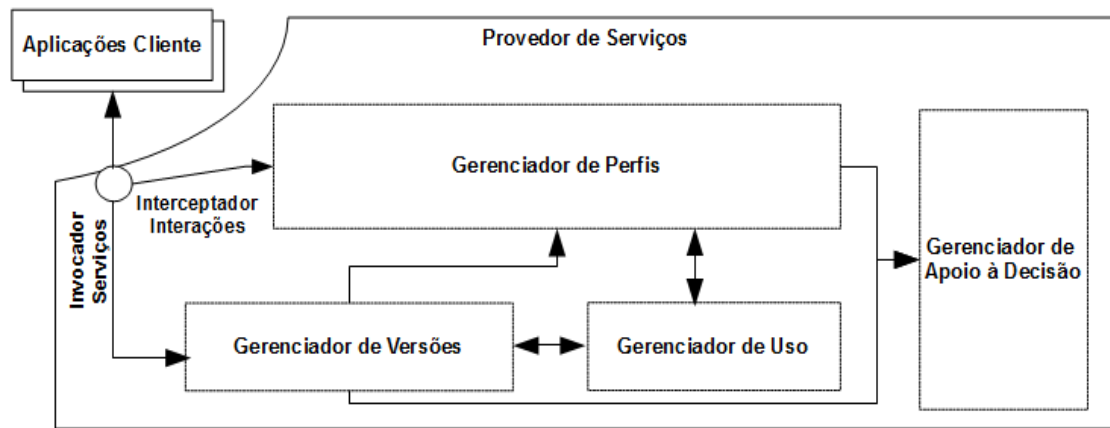


Figura 1.2: Visão geral do framework WS-EVOLV.

O Gerenciador de Apoio à Decisão representa uma importante extensão ao WS-EVOLV, relacionando as informações geradas pelos demais módulos do *framework* com as necessidades de decisão em uma perspectiva de negócios.

1.3 Objetivo

O objetivo do presente trabalho é propor uma abordagem de BI para apoiar a tomada de decisão relacionada ao impacto das mudanças durante o ciclo de vida dos serviços, com a proposta de ir além da análise baseada no pior caso, e considerar o impacto real por meio da análise de indicadores de uso. Com isso, pretende-se demonstrar que, através da análise de indicadores financeiros e de uso dos serviços, as técnicas de BI podem ser empregadas para mensurar o impacto das mudanças durante o ciclo de vida dos serviços. A abordagem de BI apresentada neste trabalho engloba:

- a identificação de métricas financeiras e de uso que podem ser utilizadas para mensurar o impacto da mudança;
- a integração das métricas supracitadas em uma modelagem multidimensional que abrange as perspectivas de análise relacionadas à provisão de serviços, com o objetivo de prover uma visão integrada e unificada.

Uma arquitetura de *Data Warehousing* foi desenvolvida para lidar com a falta de padrões e com a heterogeneidade das fontes de dados, cenário recorrente durante a integração e consolidação dos dados no contexto de provisão de serviços.

Também foi elaborado um estudo de caso utilizando dados sintéticos, construídos a partir da análise de cenários reais. Esse estudo de caso é utilizado para demonstrar a aplicabilidade da solução proposta.

A principal contribuição deste trabalho abrange a proposição de uma abordagem capaz de representar métricas para atender aos diferentes requisitos de decisão na gestão da evolução de serviços, o que aprimora o apoio ao processo de tomada de decisão do provedor de serviços em uma perspectiva estratégica e de negócios. Resultados preliminares foram descritos em (SILVA et al., 2012) e (SILVA; BECKER; GALANTE, 2013).

1.4 Organização do Texto

Esta dissertação está organizada em oito capítulos. No Capítulo 2 são apresentados os conceitos básicos relacionados a serviços *web* e BI, indicando como esses conceitos são utilizados no presente trabalho. O Capítulo 3 descreve os trabalhos relacionados, analisando as principais abordagens que tratam a evolução de serviços, os *stakeholders* no contexto de serviços e a governança de serviços. No Capítulo 4 é apresentada uma visão geral da solução proposta neste trabalho. O Capítulo 5 detalha o modelo do DW que faz parte da solução proposta, apresentando os indicadores identificados, a forma como os mesmo são representados e a modelagem dimensional. O Capítulo 6 especifica os componentes da arquitetura desenvolvida para o processo de *Data Warehousing*. O Capítulo 7 apresenta um estudo de caso que ilustra as possibilidades analíticas da abordagem proposta neste trabalho. Por fim, o Capítulo 8 apresenta as conclusões e perspectivas para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os principais conceitos que fundamentam este trabalho. Os conceitos abrangem a fundamentação básica sobre serviços *web*, a evolução de serviços e as classificações quanto aos tipos de mudanças. Também são apresentados os principais conceitos de BI e DW, os quais são usados como parte da abordagem proposta neste trabalho.

2.1 Serviços Web

Fang *et al.* (2007) definem serviço como uma função autônoma de negócios, a qual recebe requisições e retorna respostas segundo uma interface bem definida. Um serviço *web* é a implementação deste conceito no contexto *web*, que é o meio para a troca de requisições/respostas. Serviços possuem como principais características a independência de plataforma e o baixo acoplamento. De acordo com Endrei *et al.* (2006), esta última característica é o objetivo chave de arquiteturas orientadas a serviços, pois possibilita alterar aspectos comportamentais de um componente sem afetar os demais.

Um serviço *web* é disponibilizado através de uma interface que descreve as operações, os tipos de dados de entrada/saída, os parâmetros necessários para a utilização e outras informações. Para descrever tais informações de maneira compreensível e padronizada, em geral são utilizadas linguagens como a WSDL. Protocolos como o SOAP são empregados na comunicação entre aplicações e serviços. Os aspectos de implementação (métodos e construções internas) do serviço são transparentes para as aplicações que consomem o serviço, pois apenas a interface do serviço fica exposta externamente para as aplicações consumidoras (PAPAZOGLU, 2003).

O princípio de um serviço *web* é prover alguma funcionalidade em nome de uma pessoa ou organização, o *provedor* do serviço. O provedor é o responsável por oferecer o serviço e pela sua manutenção. A pessoa/organização que utiliza as funcionalidades providas pelo serviço é o *cliente* do serviço (W3C, 2011). Provedores e clientes constituem os *stakeholders* básicos dentro da arquitetura de serviços. Outros *stakeholders* (por exemplo, desenvolvedores e integradores de serviços) também estão relacionados ao contexto de serviços (TREIBER *et al.*, 2008).

É comum que essa relação entre provedores e clientes seja apoiada por contratos de nível de serviços (SLAs). Os SLAs têm como objetivo formalizar os requisitos que o provedor deve atender ao disponibilizar o serviço, determinando penalidades em casos de quebra do serviço ou quando aspectos de QoS não são atingidos.

Atualmente diversas empresas têm desenvolvido portfólios de produtos baseados no paradigma de SaaS. Essa abordagem baseia-se na disponibilização de ecossistemas de serviços *web* sob demanda, os quais visam fornecer serviços como terceirização de pagamentos, mecanismos de autenticação, computação em nuvem, armazenamento de dados, etc. Nesse cenário, vários serviços podem ser usados em conjunto. Por exemplo, uma loja que realiza vendas pela *web* pode utilizar um serviço que permita a realização de pagamentos e um serviço que armazene os dados sobre cada venda realizada.

Neste trabalho, a noção de portfólio de serviços em grande escala e as relações entre provedor e cliente em termos de fornecimento dos serviços são as bases para compreender o cenário e o problema tratados pela solução proposta.

2.2 Versionamento e Compatibilidade

Os serviços, assim como os softwares tradicionais, tendem a sofrer mudanças que fazem parte do processo contínuo de melhora do serviço. As mudanças levam à evolução e, dentro do paradigma de orientação a serviços, isso representa a criação e a desativação de versões de um mesmo serviço. Tal processo é conhecido como versionamento. No contexto de serviços, o versionamento possui duas dimensões: versionamento de implementação e versionamento de interface (ANDRIKOPOULOS; BENBERNOU; PAPAZOGLU, 2012).

O versionamento de implementação está diretamente relacionado com os aspectos de desenvolvimento do serviço (código, documentação e recursos, por exemplo). Tipicamente, esse tipo de versionamento envolve ferramentas para controle de versão, assim como no contexto de desenvolvimento de software tradicional. Os aspectos de implementação e o seu versionamento são transparentes para os clientes, i.e., não envolvem mudanças na interface do serviço (ANDRIKOPOULOS; BENBERNOU; PAPAZOGLU, 2012).

Já o versionamento de interface envolve a descrição do serviço, o que afeta a forma como o cliente interage com o serviço. Na literatura há diversas propostas (não mutuamente-exclusivas) para o versionamento de interface em diversos aspectos (ANDRIKOPOULOS; BENBERNOU; PAPAZOGLU, 2012). A maioria dessas propostas abrange:

- a) identificação de versões de serviço - envolve formas de nomear e distinguir as versões. Existem técnicas de criação de versões com identificação do tipo X.Z, onde X remete a um algarismo que representa uma atualização maior e o Z identifica uma atualização menor da versão X. Outras técnicas envolvem a representação de versões utilizando uma data ao invés de um identificador sequencial;
- b) métodos para versionamento do serviço - há métodos que envolvem a utilização de novos *namespaces* XML para cada nova versão principal, enquanto outros usam a técnica de VIDs (*Version Identifiers*) para identificar dar unicidade às diferentes versões. Os VIDs podem ser aplicados como atributo no elemento raiz da descrição do serviço, ou para cada parte do documento que representa o serviço ou ainda no endereço do serviço. As técnicas de novos *namespaces* XML e VIDs também podem ser utilizadas em conjunto;

- c) estratégias para versionamento - abrangem a manutenção de múltiplas versões do serviço, para fins de compatibilidade, o que implica em estratégias para a desativação de versões antigas e o agrupamento de versões semelhantes;
- d) modelo de versionamento - os modelos são classificados como *cliente* (quando se deixa o cliente descobrir por conta própria a nova versão implantada), *notificação* (o cliente é alertado que uma nova versão existe e, geralmente, tem de tomar alguma ação dentro de um período pré-estabelecido) ou *transparente* (a nova versão não precisa ser informada, pois não afeta o cliente).

Relacionado ao versionamento transparente, o conceito de compatibilidade envolve a garantia de que uma nova versão de um serviço pode ser implantada sem afetar os clientes existentes.

Em termos gerais, o conceito de retrocompatibilidade se refere a sistemas de hardware ou software que podem usar interfaces de versões anteriores de um sistema. Em relação a serviços, a retrocompatibilidade remete a como as mudanças de interface em um serviço afetam os clientes do mesmo (ENDREI et al., 2006). Uma mudança é considerada retrocompatível quando não afeta o cliente e envolve a inclusão de novas operações ou de novos tipos de dados (BROWN; ELLIS, 2004). De acordo com Andrikopoulos, Benbernou e Papazoglou (2012), essas mudanças caracterizam-se quando as alterações em um serviço não implicam em alteração (*upgrade*) do cliente para utilizar as novas funcionalidades. Quando a mudança afeta o cliente então ela é considerada uma mudança retroincompatível (BROWN; ELLIS, 2004). A retroincompatibilidade é gerada quando ocorre a remoção e/ou a troca de nome de uma ou várias operações, quando os parâmetros de uma operação são alterados e quando são modificados tipos de dados de entrada e/ou saída (FANG et al., 2007), (BECKER et al., 2008).

O conceito de versionamento é abordado neste trabalho por ser o principal mecanismo utilizado pelos provedores de serviço durante a alteração de serviços, garantindo a compatibilidade com os clientes existentes. A análise de compatibilidade é estrategicamente relevante para o provedor, pois existem consequências quando se decide continuar provendo uma versão de um serviço: enquanto manter versões de um serviço mantém os clientes ilesos de impactos e satisfeitos, essa manutenção também implica em custos não-lineares de provisão.

2.3 Tipos de Mudança em Serviços Web

Um provedor pode ter diversas motivações para realizar uma alteração que implique em mudanças na interface do serviço. Alterações de objetivos de negócios do cliente (p.ex., novas requisições organizacionais) ou do próprio provedor (p.ex., atender novos perfis de consumidores), alterações de implementação e/ou semântica, integração de novas aplicações, adequações para atender questões ligadas a QoS e a realização de manutenções e correções são alguns exemplos de cenários que podem resultar em mudanças na interface do serviço. As necessidades de adaptação são naturais e fazem parte do processo de evolução de todo software (LEITNER et al., 2008).

Andrikopoulos, Benbernou e Papazoglou (2012) classificam a natureza de uma mudança em:

- estrutural - envolve a interface do serviço (tipos de dados e operações, por exemplo);
- comportamental - afeta os protocolos de negócios e as regras de troca de mensagens;
- não-funcional - tange os aspectos ligados a fatores como regras de negócio e QoS.

As mudanças podem ser classificadas como mudanças superficiais (*shallow changes*) ou como mudanças profundas (*deep changes*). Mudanças superficiais são aquelas que afetam apenas um serviço específico e os consumidores ligados ao mesmo. Uma possível abordagem para lidar com essa categoria de mudança envolve a utilização de versões compatíveis. As mudanças profundas, por outro lado, são aquelas capazes de afetar uma cadeia ou mesmo um portfólio de serviços, causando um efeito cascata para clientes/serviços que podem não ter relação direta com a mudança implantada. Esse tipo de mudança demanda uma metodologia apropriada para a gestão do ciclo de vida orientado a serviços (PAPAZOGLU; ANDRIKOPOULOS; BENBERNOU, 2011).

Neste trabalho, o cenário de mudanças profundas é abordado com mais ênfase por envolver necessidades de decisão críticas, envolvendo clientes diretos e indiretos. Não existem abordagens que ofereçam apoio para a tomada de decisão nesse tipo de cenário.

2.4 Business Intelligence (BI)

BI se tornou popular nas áreas de negócios e TI nos anos 90. Chen, Chiang e Storey (2012) dividem BI em três segmentos:

- BI 1.0: baseia-se na integração e análise de dados que na maioria das vezes são estruturados, oriundos de sistemas legados, e armazenados em bases de dados. As aplicações e tecnologias adotadas atualmente na indústria são consideradas BI 1.0. O processo de *Data Warehousing* é considerado a base de BI 1.0 e é detalhado na Seção 2.4.1;
- BI 2.0: tem como mote a *Web 2.0*, explorando a análise de dados semi-estruturados gerados por consumidores, sistemas de redes sociais e de massificação. Esse tipo de dado é organizado e visualizado através de técnicas de mineração de textos e de mineração de dados *web*.
- BI 3.0: uma área emergente e que ainda deve amadurecer, a qual baseia-se na massificação de dispositivos móveis e sensores. Esses dispositivos geram novos tipos de dados com reconhecimento de localização, centrados em pessoas e com relevância de contexto.

O presente trabalho se fundamenta nos conceitos de BI 1.0, aqui referenciado simplesmente como BI, pois tal segmento mostra-se o mais maduro para o cenário de análise de integração de dados organizacionais, oferecendo os mecanismos mais adequados.

BI remete ao uso de informações organizacionais internas e externas para a tomada de decisões de negócios. Esse processo representa a transformação de dados em

informação e de informação em conhecimento útil para a tomada de decisão organizacional (KIMBALL; ROSS, 2002). A Figura 2.1 ilustra a pirâmide que representa a estrutura hierárquica de uma organização na execução de operações.



Figura 2.1: Estrutura organizacional para a execução de operações (adaptada a partir de (GOLFARELLI; RIZZI; CELLA, 2004)).

O conceito de BI foi desenvolvido para atender às necessidades de tomada de decisão no nível estratégico de uma organização, que envolve tipicamente a direção e/ou os executivos da organização. No entanto, também se aplica às decisões de nível tático (gerência da organização). Alguns autores expandem a utilização de mecanismos de BI também para o nível operacional e para as necessidades de decisão inerentes à execução de tarefas (GOLFARELLI; RIZZI; CELLA, 2004). Um sistema de BI é direcionado a um tipo de usuário diferente daquele que geralmente utiliza os softwares operacionais tradicionais, pois o responsável pela tomada de decisão deve observar o rumo da organização e fazer perguntas que auxiliarão o seu processo de tomada de decisão (KIMBALL; ROSS, 2002).

Na literatura existem diversos modelos de processos de BI, os quais se diferenciam especialmente pelo número de fases, estrutura, fontes de informação e métodos de armazenamento (LÖNNQVIST; PIRTTIMÄKI, 2006). De acordo com Moss e Atre (2003), o ciclo de vida de um projeto de BI possui as seguintes etapas:

- detecção de oportunidade/necessidade de negócio;
- estratégia de apoio à decisão;
- planejamento, levantamento de requisitos estratégicos e análise do negócio;
- projeto da solução;
- desenvolvimento, testes e implementação;
- avaliação e tomada de decisão.

Neste trabalho, BI é o meio para possibilitar que provedores possam compreender o impacto da mudança em seu portfólio de serviços em uma perspectiva de negócios. Um ambiente de BI pode apoiar o provedor com possibilidades de análise úteis para avaliar o impacto das alternativas de mudança em termos de estratégias de negócios.

2.5 Data Warehousing

O processo de *data warehousing* contempla a união e o gerenciamento de dados das mais diversas fontes, com o objetivo de obter uma visão consolidada e detalhada de uma porção de negócios (GARDNER; 1998). Os principais componentes de *data warehousing* são apresentados na sequência.

2.5.1 Data Warehouse

O *Data Warehouse* (DW) é uma das principais bases para uma solução de *Data warehousing*. DW pode ser definido como uma base de dados centralizada, integrada, histórica e orientada a assunto, voltada para a tomada de decisão. Um DW pode ser composto por DWs menores referentes a assuntos específicos. Tais porções são conhecidas como *Data Marts* (e.g. *Data Mart* de vendas, *Data Mart* de marketing, *Data Mart* de recursos humanos, etc) Os dados em um DW são representados através de um modelo multidimensional baseado em tabelas fato e tabelas dimensão (KIMBALL; ROSS, 2002).

Tabelas fato armazenam métricas relacionadas a um contexto de negócios. As métricas representam tanto mensurações de um determinado quanto KPIs (*Key Performance Indicators*). KPIs representam uma mistura de diferentes tipos de indicadores, como KRIs (*Key Result Indicators*) e PIs (*Performance Indicators*). (PARMENTER, 2010). Neste trabalho, KPIs e métricas são usados como sinônimos. Pelo fato de cada métrica representar um aspecto da estratégia de negócios da organização, a sua análise influencia diretamente a tomada de decisão. A maioria das métricas são numéricas e aditivas, i.e., podem ser somadas quando são sumarizadas. Porém, algumas métricas podem ser semiaditivas, quando só podem ser somadas quando sumarizadas com dimensões específicas, e não-aditivas, quando não podem ser somadas e só podem ser sumarizadas por meio de operações de contagem ou média (KIMBALL; ROSS, 2002).

As dimensões complementam as informações contidas nas tabelas fatos, detalhando as perspectivas de análise em relação ao fato. É comum que as tabelas que representam dimensões contenham diversas colunas para representar os atributos da dimensão (por exemplo, uma dimensão que representa produtos pode conter dados sobre o nome do produto, a categoria, o tipo, etc). Os atributos também podem ser organizados hierarquicamente (por exemplo, uma dimensão que representa regiões pode ter uma hierarquia composta por país, estado e município). O detalhamento de cada dimensão em relação a uma tabela fato determina o nível de granularidade do fato (KIMBALL; ROSS, 2002).

Por exemplo, uma tabela fato pode conter dados sobre a venda de um produto. Analisando a tabela que contém as métricas relacionadas à quantidade de produtos vendidos, é possível detalhar os dados por algumas dimensões, como data (e.g. o dia em que o produto foi vendido), local (e.g. a cidade em que o produto foi vendido) e produto (e.g. o nome do produto vendido). Portanto, a granularidade da tabela fato é dia/cidade/produto. Porém, essas informações podem ser agregadas de acordo com as informações disponíveis em cada dimensão (por exemplo, os dados podem ser agregados por ano/país/categoria do produto).

O modelo estrela (*star schema*) é um dos mais utilizados na modelagem de um DW. O nome estrela é baseado na representação da tabela fato como o “centro” da estrela e das dimensões como as “pontas” da estrela e relacionadas à tabela fato.

Um problema ocorre quando as métricas não possuem exatamente a mesma granularidade, o que impede que elas possam ser armazenadas em uma única tabela fato. Quando essas métricas se relacionam com dimensões conformadas (i.e., dimensões que têm o mesmo contexto para cada fato), propõe-se o modelo constelação de fatos (*fact constellation schema*), ou simplesmente MTFS (*Multiple Fact Table Schema*). O MTFS permite a representação de múltiplos fatos com granularidades distintas, que mesmo não tendo relação direta, podem ser relacionados por meio de dimensões conformadas. Isso garante melhor aproveitamento das dimensões e mais expressividade para o modelo (POE; KLAUER; BROBST, 1998) (BECKER; RUIZ; SANTOS, 2002).

Pelo fato de o MTFS expressar melhor cenários com múltiplas tabelas fato em um mesmo espaço, esse modelo foi o escolhido para a modelagem utilizada neste trabalho.

2.5.2 Extração, Transformação e Carga de Dados (ETL)

O processo de Extração, Transformação e Carga dos Dados, representado pela sigla ETL (*Extract-Transformation-Load*), realiza as ações de transformação e integração dos dados de diferentes fontes de dados durante a consolidação no DW.

As fontes operacionais da organização armazenam os dados que precisam ser integrados no DW. Tais fontes podem ser bancos de dados operacionais, planilhas, arquivos texto, serviços, etc. Conseqüentemente, elas podem ter diversos formatos sob os quais se possui pouco controle (KIMBALL; ROSS, 2002).

A partir das diferentes fontes operacionais ocorre a extração das informações, as quais são armazenadas em uma área intermediária entre as fontes operacionais e a camada de apresentação, a área de *staging*. Nessa área são carregados os dados operacionais e são realizadas as operações de transformação dos dados (limpeza, combinação de dados de múltiplas fontes, atribuição de chaves de DW, etc), os quais são inseridos no DW e ficam disponíveis para a consulta na camada de apresentação. A área de *staging* pode ser composta por uma base de dados normalizada ou mesmo por um sistema de arquivos *flat* (KIMBALL; ROSS, 2002).

Para extrair os dados das fontes de dados, que geralmente possuem características distintas e demandam métodos específicos, a arquitetura baseada em *wrappers* se torna uma alternativa. *Wrappers* são artefatos de *software* que usam uma interface única para encapsular uma ou mais aplicações. Eles são necessários para integrar dados brutos espalhados em diferentes fontes de dados e também para realizar derivações conforme a necessidade (SILVEIRA; BECKER; RUIZ, 2010).

Para o presente trabalho, a abordagem de utilização de *wrappers* foi adotada, pois a mesma se aplica a cenários que demandam flexibilidade e adaptabilidade em relação a múltiplas e diversas fontes de dados.

2.5.3 Camada de Apresentação

Os dados armazenados em um DW podem ser consultados através de ferramentas analíticas utilizadas pelo responsável pela tomada de decisão. A Figura 2.2 sintetiza o fluxo de dados desde a coleta e integração de diferentes fontes de dados no DW até as diferentes ferramentas de consulta que os usuários de negócios podem utilizar para explorar o DW. Dependendo do tipo de análise que cada usuário pretende fazer, a ferramenta de visualização mais indicada pode envolver:

- cubos OLAP - são gerados a partir dos dados dimensionais disponíveis no DW e podem ser explorados por meio de tabelas *pivot* (*cross tabs*). Cubos de dados permitem que o usuário execute operações de consulta como *slicing* e *dicing* (fatiamento dos dados), *drill down/up* (navegação detalhada/sumarizada através das dimensões do cubo) e *roll-up* (agregações de dados através dos níveis de cada dimensão);
- *dashboards* - representam a sumarização de diversos KPIs em um único painel, facilitando a visão macro da situação dos indicadores organizacionais;
- relatórios - contêm gráficos e/ou tabelas, dependendo do foco (operacional ou analítico), para apresentar maiores detalhes sobre um contexto;
- alertas – disparadores de avisos para os responsáveis pela tomada de decisão. As condições para o disparo geralmente dependem de uma condição de fundamental importância para a estratégia organizacional e que deve ser monitorada constantemente (e.g. momento em que o indicador que representa a quantidade de vendas atinge um valor abaixo do mínimo esperado).

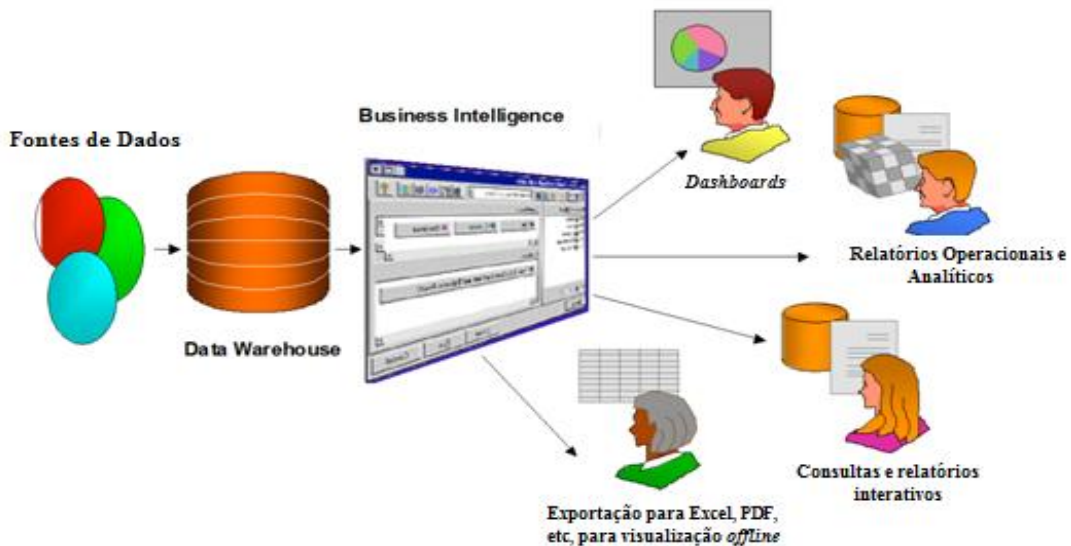


Figura 2.2: Fluxo das informações em um DW até a camada de apresentação.

Neste trabalho, utiliza-se o conceito de DW para implementar o ambiente de BI proposto como solução para o problema tratado. Essa abordagem é aquela que se mostra mais madura e adequada frente à necessidade de integração e análise de dados internos do provedor de serviços.

2.6 Considerações

Neste capítulo foram apresentados os principais conceitos relacionados a serviços *web*, abrangendo especialmente conceitos como compatibilidade, versionamento e tipos de mudança. Também foram abordados os princípios básicos de *Business Intelligence*, explorando o processo de *Data Warehousing* e suas etapas.

Com a fundamentação sobre serviços, versionamento e compatibilidade, é possível compreender o cenário de evolução de serviços, enquanto o entendimento sobre BI e DW esclarece como um ambiente de suporte à tomada de decisão pode ser implementado e como esse ambiente pode ser útil para o responsável pela tomada de decisão.

Esses conceitos são inerentes à proposta central deste trabalho, uma abordagem que relaciona BI e necessidades de decisão na gestão do ciclo de vida de serviços.

3 TRABALHOS RELACIONADOS

Este capítulo aborda os principais trabalhos que estão relacionados à gestão da evolução de serviços. Os trabalhos são analisados sob três vieses: quanto ao gerenciamento da mudança em serviços, em especial considerando o contexto de gestão de mudanças profundas; quanto aos papéis dos *stakeholders* no contexto de serviços; e quanto ao conceito de governança de serviços.

Por fim, apresenta-se uma análise comparativa entre os trabalhos estudados, com o objetivo de estabelecer uma relação entre *stakeholders*, políticas de governança e atividades/fases para a gestão de mudanças em serviços.

3.1 Gestão da Evolução de Serviços

Na literatura relacionada a serviços *web*, há poucos trabalhos que abordam os desafios inerentes à implantação de mudanças profundas e os problemas relacionados à gestão da evolução de serviços sob a perspectiva de negócios. A maioria dos trabalhos trata a gestão da mudança considerando a perspectiva técnica, apresentando alternativas que envolvem técnicas de versionamento, mecanismos para a determinação de compatibilidade e a utilização de *proxies* para redirecionamento (FANG et al., 2007), (FRANK et al., 2008), (LEITNER et al., 2008), (BECKER et al., 2008), (PAPAZOGLU; ANDRIKOPOULOS; BENBERNOU, 2011).

Essas abordagens, apesar de serem importantes para o entendimento das mudanças superficiais e do impacto durante o *design* do serviço, consideram o cenário de pior caso de impacto da mudança. Não se leva em consideração que os serviços podem ser usados de maneiras distintas pelos diversos clientes e que, portanto, podem ser afetados pelas mudanças de modos também distintos (YAMASHITA; BECKER; GALANTE, 2011), (YAMASHITA et al., 2012). Esse tipo de análise abre novas perspectivas para o provedor de serviços, especialmente considerando o cenário de mudanças profundas, o qual demanda um tipo de apoio à decisão mais preciso e que vai além do aspecto técnico, abrangendo a análise do impacto real ao invés do pior caso.

Papazoglou, Andrikopoulos e Benbernou (2011) abordam a gestão de mudanças profundas no contexto de portfólio de serviços. Os autores propõem uma metodologia para o ciclo de vida orientado a mudanças, com o intuito fornecer uma base sólida para tratar as mudanças profundas. Essa metodologia engloba métodos e procedimentos padronizados para manipular mudanças de forma eficiente, minimizando os impactos e garantindo a execução do serviço dentro do comportamento esperado. A metodologia também objetiva dar a capacidade de antecipar condições não previstas ou não especificadas (por exemplo, questões de QoS) e abordá-las de maneira apropriada. Essa

abordagem possibilita que os serviços afetados sejam reconfigurados, alinhados e controlados.

A proposta de Papazoglou, Andrikopoulos e Benbernou divide-se em três fases inter-relacionadas, compostas por atividades que definem todo o processo de mudança:

1. Necessidade de Evolução - esta fase envolve a identificação das causas que levam à necessidade de mudança em serviços dentro de um portfólio e o entendimento sobre o escopo da mudança. A coleta de KPIs apresenta-se como uma das atividades previstas nesta fase, possibilitando a análise do impacto nas próximas fases;
2. Análise do impacto das mudanças - esta fase foca na análise, no replanejamento e na melhora dos serviços existentes, provendo entendimento profundo sobre o escopo, a funcionalidade e o reuso dos serviços que devem ser alterados. A análise do impacto da mudança é realizada através dos modelos “*as-is*”, para entender o portfólio atual de serviços, e “*to-be*”, para detalhar os serviços com as novas funcionalidades requisitadas e visualizar o nível de desempenho com a aplicação das mudanças. Os autores propõem a utilização de modelos de análise de *gap* para relacionar os modelos “*as-is*” e “*to-be*”, o que permite a análise comparativa entre o serviço atual e a versão futura, em termos de funcionalidade, níveis de desempenho (tendo KPIs como referência) e capacidade (em relação ao alcance das especificações, concordância com o SLA e requisitos).

Ainda nesta fase, é preciso definir as abordagens para problemas que podem surgir com a implantação das mudanças, como:

- o fluxo do serviço, com o sequenciamento e a duplicação de atividades;
- o controle do serviço, que envolve as diretrizes sobre o funcionamento do serviço;
- a sobreposição/redundância e o conflito na funcionalidade dos serviços, que estão relacionados com a estratégia de negócios e/ou com as políticas para serviços;
- a entrada/saída, o tratamento para possíveis reflexos em QoS devido à latência de I/O.

A estimativa de custos para o projeto de implementação das mudanças é outra atividade importante nesta segunda fase, pois ajuda a determinar se os custos estão dentro do esperado ou não, se é viável realizar o projeto ou mesmo se a organização deve considerar a hipótese de terceirização ou alternativas distintas.

3. Alinhar, refinar e definir - esta fase ocorre após a decisão por realizar a mudança e trata da implementação das novas funcionalidades, desde o alinhamento do projeto final e o refinamento até os testes e a implantação. Os desenvolvedores do serviço criam um Modelo de Integração do Serviço para implantar a estratégia de integração do serviço. O modelo também possibilita o monitoramento dos serviços em produção, considerando as

necessidades de orquestração e de mensagens. A identificação de tais necessidades pode gerar novas demandas e reiniciar o ciclo de mudanças.

Essa metodologia pode ser vista como uma abordagem inovadora no contexto de mudanças em serviços, especialmente pelo enfoque em mudanças profundas, mas também por ser aplicável a situações em que ocorrem mudanças superficiais. No entanto, a metodologia para o ciclo de vida orientado a mudanças assume que dependências, limitações e componentes são previamente conhecidos, mas não especifica como estes dados podem ser obtidos. Além disso, os autores não exploram as possibilidades analíticas com mais detalhes e nem o tipo de suporte que pode ser provido para a tomada de decisão.

3.2 Stakeholders

O termo *stakeholder* é utilizado para designar um indivíduo ou um grupo que pode afetar ou ser afetado pelo desempenho da organização em relação ao alcance dos objetivos estabelecidos para a mesma (GU; PARKIN; LAGO, 2011). Alguns trabalhos trazem classificações distintas em relação aos *stakeholders* do ambiente de serviços e específicos do contexto de SOA. Em especial, nas subseções seguintes são analisados os trabalhos de Treiber et al. (2008), Gu, Parkin e Lago (2011) e Kajko-Mattson, Lewis e Smith (2007). Esses trabalhos são os mais recentes que analisam *stakeholders* e seus papéis no contexto de SOA. Também é realizada uma comparação entre os *stakeholders* identificados nas três abordagens.

3.2.1 Stakeholders na abordagem de Treiber et al. (2008)

Treiber et al. (2008) analisam as interdependências das mudanças em serviços e os efeitos gerados pelas mesmas, considerando os distintos *stakeholders* relacionados.

A metodologia proposta para classificar os fatores de influência abrange os seguintes passos:

- identificação dos *stakeholders*, através da análise do processo de desenvolvimento do serviço e do fluxo da comunicação;
- identificação de tarefas e atribuição de responsabilidades, com a definição do papel de cada *stakeholder*;
- coleta de dados e identificação da origem dos mesmos, onde os dados podem ser estáticos (e.g. informações obtidas manualmente, por meio de *feedback* do cliente ou através de inserção em um framework específico) ou em tempo de execução (e.g. QoS, estatísticas de uso ou latência).

Os *stakeholders* são classificados como Provedor (*Service provider*), Desenvolvedor (*Service developer*), Cliente (*User*) ou Integrador de Serviços (*Service Integrator*).

Uma das grandes contribuições do trabalho de Treiber *et al.* envolve o posicionamento das mudanças em serviços frente ao papel de cada *stakeholder* dentro de um ecossistema de serviços. As tarefas estão relacionadas a cada *stakeholder* da seguinte maneira:

- Provedor:
 - planeja e concebe o serviço (define requisitos, negocia o SLA, determina os preços);
 - gerencia as mudanças.
- Desenvolvedor:
 - implementa o serviço;
 - cuida das alterações na interface do serviço;
 - gerencia as diferentes versões implementadas;
 - identifica as mudanças de requisitos para mudanças de implementação/interface.
- Cliente:
 - solicita mudanças ligadas a QoS;
 - modifica os requisitos dos serviços.
- Integrador de serviços:
 - interessa-se por mudanças de interface;
 - modifica os requisitos dos serviços.

A Figura 3.1 ilustra como a atuação de um *stakeholder* e o ciclo de mudanças que leva ao processo de evolução do serviço. Os tipos de alteração estão inter-relacionados e implicam em uma cadeia de modificações. Uma alteração de requisitos, por exemplo, pode ser disparada pelo Provedor, pelo Desenvolvedor ou pelo Integrador de Serviços. Esta alteração gera mudanças de implementação, aspectos de QoS, uso e SLA. Os demais tipos de mudança também podem disparar uma cadeia de mudanças.

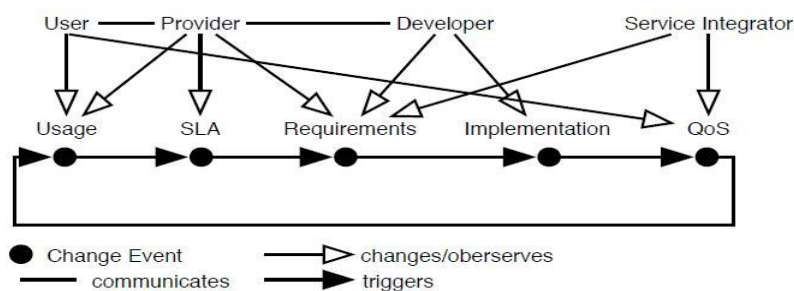


Figura 3.1: O ciclo de mudanças na abordagem de Treiber et al. (2008).

Pode-se observar que o papel de Integrador de Serviços é semelhante ao papel de Cliente, pois o Integrador atua como um tipo de cliente que também se preocupa com as alterações na interface de um serviço.

3.2.2 Stakeholders em abordagens SOA

Outros trabalhos apresentam classificações de *stakeholders* considerando o contexto de SOA em geral.

Gu, Parkin e Lago (2011) propõem uma taxonomia dos *stakeholders* na engenharia de serviços, abrangendo não apenas serviços *web*, mas também o contexto de SOA. Os

autores destacam cinco papéis que representam o foco de atuação de diferentes grupos de *stakeholders*:

- *Service provider* - proprietário do serviço, o qual concebe e disponibiliza serviços e também é responsável pela implementação e manutenção dos mesmos;
- *Service composer* - disponibiliza serviços compostos para a utilização interna ou externa à organização;
- *Application builder* - semelhante ao *Service composer*, mas a composição ocorre com aplicações SOA compostas;
- *Application client* - o usuário final, que utiliza o serviço para atingir objetivos específicos;
- *Supporting role* - especialista que conhece o ciclo de vida do serviço e atua dando assistência para questões técnicas e/ou de negócios ou gerenciando o projeto de implementação/mudança do serviço.

Outro trabalho, desenvolvido por Kajko-Mattsson, Lewis e Smith (2007), tem como objetivo apresentar uma ideia para a organização dos papéis desempenhados na realização de atividades relacionadas a SOA. Os autores propõem um framework para a classificação de *stakeholders* como atores desempenhando papéis. Os papéis são segmentados em diferentes grupos (desenvolvimento, suporte, evolução e manutenção de aplicações SOA), e os atores são classificados como:

- *Infrastructure developer* - responsável pela infraestrutura necessária para manter o serviço e pela definição de ferramentas e aplicações;
- *Application developer* - realiza a integração dos serviços com as aplicações, ficando responsável pela descoberta, composição e invocação dos serviços;
- *Service provider* - foca no desenvolvimento do serviço, entendendo os requisitos dos potenciais clientes, desenvolvendo e publicando os serviços.

3.2.3 Consolidação dos Stakeholders

A Tabela 3.1 demonstra a relação entre os *stakeholders* descritos por Treiber et al. (2008), Gu, Parkin, Lago (2011) e Kajko-Mattsson, Lewis e Smith (2007).

Tabela 3.1: Relação entre os stakeholders definidos pelos trabalhos relacionados.

Treiber et al. (2008)	Gu, Parkin, Lago (2011)	Kajko-Mattsson, Lewis e Smith (2007)
Desenvolvedor	<i>Service provider</i>	<i>Service provider</i>
Provedor		
Integrador de serviços	<i>Service Composer</i>	<i>Application developer</i>
	<i>Application Builder</i>	
Cliente	<i>Application Client</i>	
	<i>Supporting role</i>	
		<i>Infrastructure developer</i>

As diferenças entre as classificações de *stakeholders* presentes nas abordagens supracitadas podem ser explicadas pelas perspectivas distintas em cada trabalho, pois enquanto Treiber et al. (2008) focam nos *stakeholders* em relação às mudanças em serviços, as outras abordagens compreendem ambientes SOA em geral.

O presente trabalho adota como base a abordagem de Treiber et al. (2008) para a definição dos *stakeholders* que atuam na gestão do ciclo de vida de serviços. A abordagem de Treiber et al. (2008) foi escolhida porque consegue representar a maioria dos *stakeholders* descritos em outras abordagens, o que possibilita que papéis, como o do Integrador de Serviços, por exemplo, também possam ser representados como Cliente. Assim, este trabalho considera que os *stakeholders* em SOA são representados por três papéis principais: o Provedor, o Desenvolvedor e o Cliente.

3.3 Governança

Governar implica dirigir, criar e impor regras e normas, garantindo que as mesmas sejam seguidas pelos *stakeholders* do processo. O conceito de governança já existe há algum tempo no universo de TI e a sua aplicação no âmbito de SOA exige bastante coordenação, especialmente por envolver sistemas distribuídos (WOOLF, 2006).

Niemann et al. (2010) argumentam que, apesar de a governança ser um elemento crucial no projeto de SOA, não há um consenso na literatura quanto à definição e à estrutura de governança nesse contexto. Após a análise comparativa de onze abordagens diferentes de governança, Niemann et al. concluem que nenhuma das abordagens compreende todos os aspectos de governança em SOA.

Considerando a abordagem de Woolf (2006), governança em SOA pode ser definida como um mecanismo que engloba todas as tarefas e processos para a especificação e o gerenciamento dos serviços. A governança é composta por três fundamentos, que envolvem a definição sobre:

- Quem é o responsável pela tomada de decisões;
- Quais decisões precisam ser tomadas;
- Quais as políticas para a tomada de decisão consistente.

Mais especificamente, Woolf descreve que a governança especifica:

- Cadeias de responsabilidades para a capacitação de pessoas;
- Medição para avaliação de efetividade;
- Políticas que orientem a organização a alcançar as metas estabelecidas;
- Mecanismos de controle para assegurar o cumprimento das diretrizes;
- Estruturas de comunicação para manter todos os *stakeholders* informados.

Há uma diferença primordial entre governança e gerenciamento: a governança define políticas, enquanto o gerenciamento segue as diretrizes definidas. O gerenciamento envolve o processo de tomada de decisão propriamente dito e a consequente implementação.

Woolf (2006) afirma que a governança implica em “vários conjuntos de práticas coordenadas juntamente”, e classifica tais práticas como segue:

- definição do serviço - visa garantir que as tarefas de criação dos serviços sejam realizadas, coordenando as equipes de criação e requisição de serviços e orientando quanto ao escopo e limites do serviço;
- implantação do ciclo de vida do serviço - define o ciclo de vida do serviço, que tipicamente é dividido em cinco fases:
 - planejado - o serviço está em *status* pré-implantação;
 - em teste - o serviço foi implementado e está na fase de testes;
 - ativo - o serviço está completamente disponível para uso;
 - *deprecated* - o serviço ainda está ativo, mas sem manutenção e com uma data de desativação anunciada;
 - *sunsetting* - o estágio final do serviço, quando ele passa a não ser mais provido. De acordo com Woolf, este é um estágio inevitável.
- versionamento do serviço - preocupações quanto ao gerenciamento de versões (criação, gerenciamento e descontinuação) e à compatibilidade;
- migração do serviço - compreende a elaboração de estratégias de migração periódica dos clientes para novas versões do serviço;
- registro do serviço – publicação das descrições sobre os serviços oferecidos, em um repositório (UDDI, por exemplo), para que possíveis clientes possam localizar serviços que atendam suas necessidades e negociar a sua utilização com o provedor;
- modelo de mensagens do serviço - envolve a definição de um modelo de dados canônico para a troca de mensagens, compartilhada por todos os serviços, evitando o “caos” de formatos para a troca de mensagens de acordo com as diferentes equipes que criam os serviços;
- monitoramento do serviço - para constatar se os serviços funcionam de acordo com o que está estabelecido pelos SLAs e para a detecção e prevenção de problemas que podem ocorrer na utilização dos serviços;
- definição de responsáveis pelo serviço - abrange a cooperação entre os diferentes *stakeholders* envolvidos com o desenvolvimento dos serviços;
- aplicação de testes ao serviço - execução de testes, em nível que deve ser definido entre provedor e consumidor (através de SLA), para garantir que o serviço funciona adequadamente, de acordo com as regras do SLA;
- segurança do serviço - abrange a definição do tipo de segurança a ser implementado.

3.4 Necessidades Relacionadas à Tomada de Decisão

Com o objetivo de identificar os requisitos de decisão no contexto de evolução de serviços, foram relacionadas as etapas do ciclo de vida orientado a mudanças definido por Papazoglou, Andrikopoulos e Benbernou (2011), os papéis e as atividades que cada *stakeholder* assume no modelo proposto por Treiber et al. (2008), e as políticas de governança SOA, seguindo a abordagem de Woolf (2006). A Tabela 3.2 sintetiza como

essas abordagens se relacionam. Essa análise foi publicada originalmente em (SILVA et al., 2012), e cabe ressaltar que ela é uma das contribuições desta dissertação.

Tabela 3.2: Relação entre as fases da metodologia para ciclo de vida orientado a mudança, governança de serviços e os *stakeholders* em serviços *web*.

Fases (PAPAZOGLU; ANDRIKOPOULOUS; BENBERNOU, 2011)	Atividades		Aspectos de Governança (WOOLF, 2006)	Stakeholders (TREIBER <i>et al.</i> , 2008)
	(PAPAZOGLU; ANDRIKOPOULOUS; BENBERNOU, 2011)	(TREIBER <i>et al.</i> , 2008)		
Identificar a necessidade e o escopo da mudança	Identificação de requisitos que levam à necessidade de mudança	Entendimento dos pré-requisitos e consequências da mudança	<ul style="list-style-type: none"> Definição do serviço 	Provedor
	Coleta de KPIs			
Avaliar o impacto da mudança e decidir	Análise do impacto da mudança	Entender as consequências das mudanças	<ul style="list-style-type: none"> Definição do serviço Implantação do ciclo de vida Segurança Versionamento Migração 	Provedor, Desenvolvedor
	Modelos “ <i>as-is</i> ” e “ <i>to-be</i> ” para a análise de <i>gap</i>			
	Estimativa de custos	Planejamento e concepção do serviço (requisitos, SLA, preços, etc)		
	Abordagem para problemas de controle, fluxo, conflito e I/O			
Realizar e implantar as mudanças		Implementação do serviço <i>web</i>	<ul style="list-style-type: none"> Modelo de mensagens Versionamento 	Desenvolvedor
		Realizar mudanças na interface do serviço		
		Gerenciar as versões		
		Controlar as mudanças de serviços		
	Refinamento, testes e implantação dos serviços	Gestão da mudança	<ul style="list-style-type: none"> Registro do serviço Monitoramento Testes 	Provedor
	Monitoramento de desempenho			
	Mensurar o alinhamento com a estratégia			

As atividades, *stakeholders* e aspectos de governança estão agrupados por fases, seguindo o modelo definido por Papazoglou, Andrikopoulos e Benbernou (2011). As fases são caracterizadas por três momentos distintos do processo de evolução de um serviço: (i) solicitação da mudança; (ii) análise do impacto e decisões quanto aos serviços envolvidos; e (iii) implementação e implantação da mudança.

Na fase “Identificar a necessidade e escopo da mudança” busca-se determinar se as mudanças são necessárias e quais são as outras mudanças que podem surgir como consequência. Nessa fase, estão agrupadas as atividades que Papazoglou, Andrikopoulos e Benbernou (2011) descrevem como inerentes ao momento em que uma necessidade de mudança é detectada. Esse ponto leva à identificação do escopo das mudanças (serviços diretamente envolvidos) e à coleta de dados para apoiar a análise do impacto da mudança, como a coleta de KPIs. Treiber *et al.* (2008) também relacionam atividades semelhantes, atribuídas ao Provedor, que tem a responsabilidade de entender os pré-requisitos que levam às mudanças e as consequências das mesmas. Nesse contexto, as práticas de governança classificadas como “Definição do Serviço” por

Woolf (2006) podem ser relacionadas às atividades desta fase, pois tais práticas de governança abrangem justamente o escopo e os limites do serviço.

A fase seguinte, “Avaliar o impacto da mudança e decidir”, contempla as atividades que, em sua maioria, estão relacionadas com o Provedor, como, por exemplo, a estimativa do impacto das mudanças e a análise de custos e políticas para o tratamento de problemas, como Papazoglou, Andrikopoulos e Benbernou (2011) descrevem na sua metodologia. As tarefas mais técnicas, como a definição de abordagens para o tratamento de possíveis problemas pós-mudança, como a sobreposição de serviços e problemas de I/O, estão relacionadas ao Desenvolvedor. Outras atividades também são agrupadas nesta fase e envolvem o que Treiber *et al.* (2008) determinam como atividades adicionais do Provedor: o planejamento/concepção do serviço e o entendimento das consequências das mudanças. Ainda nesta fase, os aspectos de governança englobam as definições quanto à implantação do ciclo de vida do serviço (que define as fases do ciclo de vida do serviço) e sobre segurança, versionamento e estratégia de migração, pois são definições que têm impacto na forma como a mudança é conduzida.

Por fim, a fase “Realizar e implantar as mudanças” abrange as atividades após a decisão pela realização da mudança. Essas atividades estão relacionadas a questões de implementação e implantação do serviço. Quanto à implementação, Treiber *et al.* (2008) atribuem as tarefas técnicas de implementação ao *stakeholder* Desenvolvedor. Esse tipo de tarefa é orientado pelas políticas de governança, as quais envolvem a padronização do modelo para a troca de mensagens que os serviços utilizam. Quanto às atividades ligadas à implantação, Papazoglou, Andrikopoulos e Benbernou (2011) descrevem que tais atividades envolvem o refinamento das implementações realizadas, o monitoramento de aspectos de QoS e o alinhamento com a estratégia da organização. Essas atividades podem ser relacionadas à tarefa definida por Treiber *et al.* (2008) como gerenciamento das mudanças, responsabilidade do Provedor. Políticas relacionadas principalmente ao registro, ao monitoramento e ao teste do serviço norteiam as atividades desenvolvidas na implantação do serviço.

A identificação das atividades relacionadas aos *stakeholders* mostra que as decisões são inerentes a todas as fases do ciclo de vida orientado a mudanças. As tarefas inerentes ao processo de decisão ajudam a identificar os pré-requisitos de análise, em direção a uma abordagem capaz de fornecer os subsídios necessários para apoiar a tomada de decisão.

3.5 Considerações Finais

Neste capítulo foram apresentados os trabalhos que tratam os aspectos fundamentais para a evolução de serviços, considerando como a gestão da mudança superficial/profunda é abordada, quem são os *stakeholders*, quais atividades realizam no contexto de SOA, e o que representa a governança de serviços.

Também foi apresentada uma relação entre as abordagens descritas em (PAPAZOGLU; ANDRIKOPOULOS; BENBERNOU, 2011), (TREIBER *et al.*, 2008) e (WOOLF, 2006). Essa relação demonstrou como as atividades, os requisitos de decisão e as políticas de governança estão relacionados aos *stakeholders* e como se encaixam dentro das etapas do ciclo de vida para a gestão de mudanças profundas. Os requisitos identificados são considerados para orientar a elaboração da abordagem de BI proposta neste trabalho.

4 O GERENCIADOR DE APOIO À DECISÃO

Este capítulo apresenta uma visão geral do Gerenciador de Apoio à Decisão, proposto neste trabalho para a gestão da evolução de serviços. Discute-se a arquitetura geral do Gerenciador de Apoio à Decisão, como o mesmo se integra aos demais módulos do WS-EVOLV, um projeto maior no qual esse trabalho está inserido, e como o ciclo de vida de BI é adaptado ao apoio à gestão do ciclo de vida de serviços.

4.1 Visão Geral

O Gerenciador de Apoio à Decisão integra o *framework* WS-EVOLV, conforme ilustra a Figura 4.1, abrangendo a identificação e a utilização de dados de uso integrada com indicadores relacionados ao ambiente de provisão de serviços. A integração com o Gerenciador de Versões e o Gerenciador de Perfis é discutida na Seção 4.2.

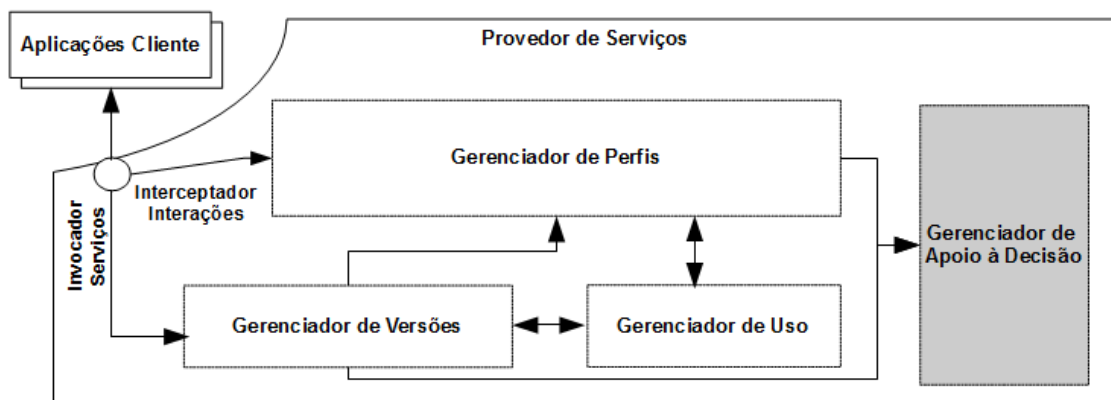


Figura 4.1: O Gerenciador de Apoio à Decisão no *framework* WS-EVOLV.

O objetivo central do Gerenciador de Apoio à Decisão é fornecer um ambiente que possibilite a representação de informações e indicadores para apoiar a tomada de decisão na gestão do ciclo de vida de serviços. Como mecanismos de BI utilizando *Data Warehousing* são consolidados na indústria e se mostram maduros para integrar, consolidar e apresentar informações analíticas, a abordagem proposta nesta dissertação envolve o uso de *Data Warehousing* na integração de dados relacionados ao ambiente de provimento de serviços, para a integração de indicadores na perspectiva financeira e de uso de serviços.

A Figura 4.2 ilustra a arquitetura geral da abordagem de BI proposta neste trabalho. A arquitetura é dividida em três camadas: a Camada de Fontes de Dados, a Camada de Transformação e Integração de Dados, e a Camada Analítica.

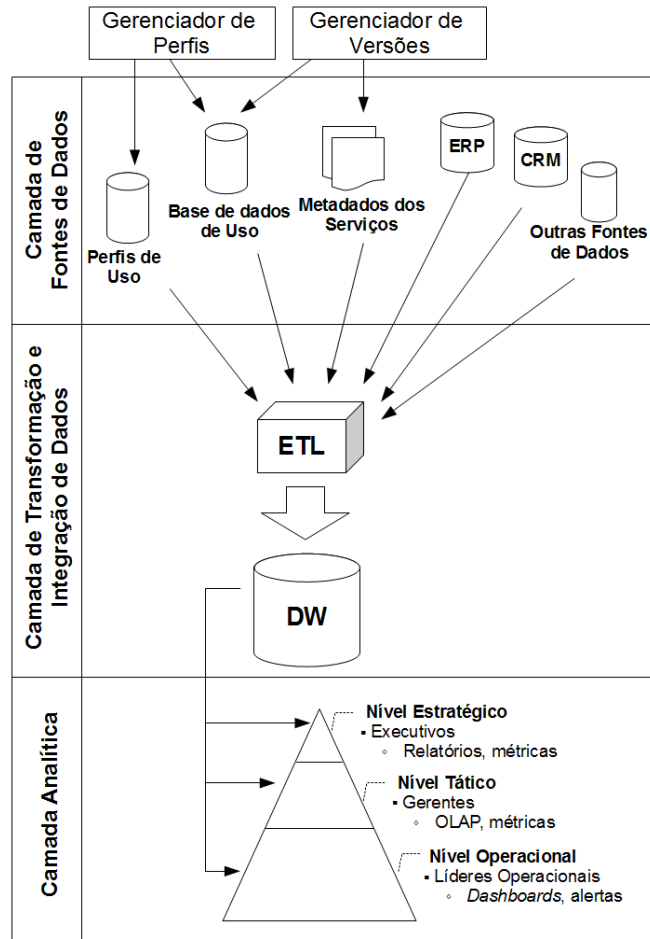


Figura 4.2: A arquitetura de BI do Gerenciador de Apoio à Decisão.

A Camada de Fontes de Dados representa a seleção de diferentes fontes de dados organizacionais, incluindo aquelas oriundas do Gerenciador de Perfis e do Gerenciador de Versões. Maiores detalhes sobre as fontes de dados no contexto de serviços são apresentados no Capítulo 6.

A Camada de Transformação e Integração dos Dados consolida os dados extraídos das fontes operacionais por meio do processo de ETL. A modelagem do DW é realizada de acordo com os objetivos de análise, a disponibilidade de informação, e os indicadores a serem analisados. A identificação de indicadores e a modelagem do DW são detalhadas no Capítulo 5. O processo de ETL, responsável por carregar o DW, é descrito no Capítulo 6.

A Camada Analítica contempla a análise dos dados consolidados no DW. Essa informação é analisada no apoio à tomada de decisão em diferentes níveis organizacionais (estratégico, tático/gerencial e organizacional), os quais apresentam diferentes necessidades e exploram as informações por meio de ferramentas analíticas específicas (relatórios, *dashboards*, cubos OLAP, etc). Os recursos analíticos para a tomada de decisão são explorados no estudo de caso apresentado no Capítulo 7.

4.2 Integração com os componentes do framework WS-EVOLV

No contexto do WS-EVOLV, o Gerenciador de Apoio à Decisão utiliza informações oriundas do Gerenciador de Versões e do Gerenciador de Perfis, conforme ilustrado na Figura 4.1.

4.2.1 Integração com o Gerenciador de Versões

O Gerenciador de Versões tem como objetivo prover uma representação mais detalhada para localizar e avaliar a compatibilidade de mudanças nas descrições dos serviços. O Gerenciador de Versões possui um modelo para determinar a compatibilidade no nível de *feature*, que representa um serviço, uma operação de um serviço ou um tipo de dado de uma operação. O modelo de compatibilidade baseia-se em um modelo de versionamento diferenciado que, ao invés de versionar todo o serviço, procura versionar apenas as *features* alteradas ou aquelas detectadas como afetadas pela mudança (YAMASHITA et al., 2012) (YAMASHITA; BECKER; GALANTE, 2012).

Uma das saídas produzidas pelo Gerenciador de Versões abrange um repositório de versões, o qual mantém as descrições de serviços versionadas e é utilizado por subcomponentes responsáveis pela análise de compatibilidade. Esse repositório de versões contém informações relevantes para o ambiente de análise proposto no Gerenciador de Apoio à Decisão, pois possibilita a obtenção de informações sobre serviços com maior nível de granularidade, abrangendo versões e operações. O provedor pode analisar o uso de um serviço detalhado por operação ou agregando essa informação por versão/serviço.

4.2.2 Integração com o Gerenciador de Perfis

O objetivo principal do Gerenciador de Perfis é detectar, através de técnicas de mineração de dados, perfis que representem padrões de uso dos serviços. O Gerenciador de Perfis como fonte de dados as enormes bases de dados geradas pelo monitoramento e registro das interações diretas e indiretas entre clientes e operações dos serviços em um portfólio. Em (YAMASHITA et al., 2012) e (SILVA et al., 2012), propõe-se um processo de KDD (*Knowledge Discovery in Databases*) utilizando *Clustering* para agrupar aplicações de acordo com os padrões de uso do serviço, gerando os perfis de uso. Os perfis de uso estão relacionados às aplicações cliente para as quais os padrões de uso foram identificados e aos serviços, versões e operações que cada aplicação utiliza.

Além dos perfis detectados, uma base de dados de uso também é gerada a partir dos dados de uso analisados. Essa base contém os dados brutos de uso, os quais são utilizados no processo de KDD (VOLLINO; BECKER, 2013).

A base de dados de uso gerada pelo Gerenciador de Perfis fornece uma fonte alternativa para obtenção dos dados sobre o uso dos serviços. A grande vantagem é que todos os dados capturados a partir da interceptação de *logs* são consolidados nessa base, o que facilita a extração durante o processo de ETL para carregar o DW, evitando a manipulação de inúmeros arquivos de *log* de requisições dos serviços,.

A integração entre o Gerenciador de Perfis e o Gerenciador Apoio à Decisão também possibilita a obtenção de informações sobre grupos de clientes de acordo com o uso do serviço, o que enriquece as possibilidades analíticas no ambiente de apoio à decisão. Com isso, ao invés de analisar cada cliente individualmente, o provedor pode fundamentar a sua análise nos grupos de clientes, detalhando apenas os clientes que fazem parte de grupos específicos relacionados à mudança.

4.3 O Ciclo de Vida de BI

Considerando o ciclo de vida de BI, a abordagem proposta neste trabalho relaciona-se às etapas que compõem o ciclo de vida de um projeto de BI. Essas etapas são adequadas às especificidades do domínio de provisão de serviços. A Figura 4.3 representa a visão geral do ciclo de vida de BI adaptado para o contexto de apoio à decisão na evolução de serviços.

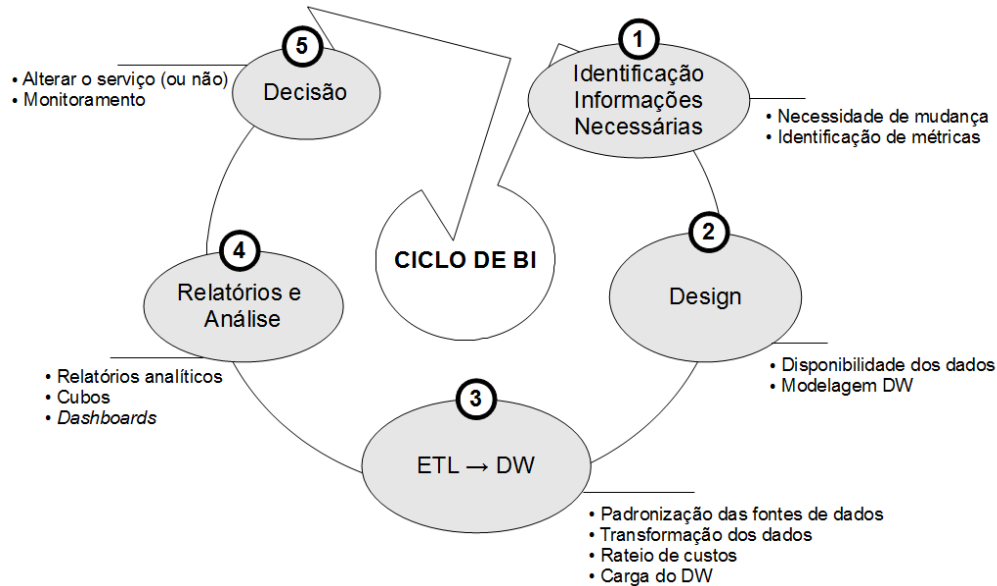


Figura 4.3: O ciclo de BI adaptado para a gestão da evolução de serviços.

Etapa 1 representa o início do ciclo após a detecção de uma necessidade de mudança, envolvendo a identificação de métricas relevantes para atender às necessidades de análise do provedor de serviços. Essa etapa é detalhada no Capítulo 5.

A Etapa 2 envolve a identificação dos dados disponíveis para análise e o projeto da modelagem dimensional, de acordo com as informações disponíveis e os requisitos de decisão. Essa etapa também é detalhada no Capítulo 5.

A Etapa 3 engloba a carga dos dados no DW. Essa etapa abrange uma série de atividades que visam padronizar as fontes de dados, além de agregar, compor e consolidar as informações no DW. Tais atividades são descritas no Capítulo 6.

Por fim, as Etapas 5 e 6 remetem à utilização de ferramentas analíticas para analisar as informações do DW, apoiando a tomada de decisão. As possibilidades analíticas são ilustradas no estudo de caso detalhado no Capítulo 7.

4.4 Considerações Finais

Este capítulo apresentou uma visão geral do presente trabalho, indicando como as informações oriundas dos demais componentes do WS-EVOLV são integradas ao Gerenciador de Apoio à Decisão, a arquitetura geral do mesmo, e a relação com o ciclo de BI no contexto de evolução de serviços. Considerando as perspectivas orientadas à análise do uso dos serviços, este trabalho apresenta um ambiente de apoio à tomada de decisão no contexto de evolução de serviços, considerando dados de uso e financeiros para compor indicadores e perspectivas analíticas.

A proposta inicial deste trabalho foi publicada em (SILVA et al., 2012), enquanto a consolidação da abordagem completa será publicada em (SILVA; BECKER; GALANTE, 2013).

5 MODELAGEM DIMENSIONAL DO GERENCIADOR DE APOIO À DECISÃO

Este capítulo detalha as duas primeiras etapas do ciclo de BI que compõe a abordagem proposta neste trabalho. A identificação da natureza das informações existentes no contexto financeiro e de uso dos serviços corresponde à etapa 1, enquanto a identificação e a representação de perspectivas e métricas de análise para a tomada de decisão, com a proposição de uma modelagem dimensional, correspondem à etapa 2 do ciclo de BI.

5.1 Visão Geral

A Figura 5.1 esquematiza o processo de elaboração da modelagem dimensional para apoiar a tomada de decisão na evolução de serviço. A construção do modelo dimensional para o contexto de mudanças em serviços envolve (i) a identificação de KPIs úteis para a análise, (ii) a identificação da granularidade dos dados, e (iii) a representação de indicadores e perspectivas de análise no modelo dimensional. As próximas seções detalham essas etapas.

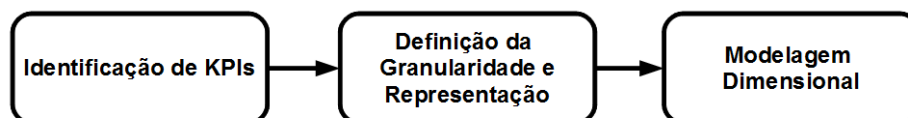


Figura 5.1: Etapas para a elaboração do modelo dimensional.

5.2 Identificação de KPIs

Este trabalho combina e analisa KPIs financeiros e de uso em um ambiente de apoio à tomada de decisão na gestão da evolução de serviços. A escolha de indicadores financeiros e de uso se deve pela adequação dos mesmos no auxílio aos requisitos de decisão mais típicos no contexto de serviços. Esses requisitos foram identificados a partir da análise apresentada na Seção 3.4, atendendo prioritariamente aos seguintes requisitos de decisão:

- analisar o impacto da mudança (quais aplicações são afetadas pela mudança);
- estimar os custos relacionados à mudança;
- planejar e conceber os serviços (requisitos, SLAs, preços);
- mensurar o alinhamento com a estratégia organizacional.

Com o uso de KPIs financeiros e de uso, torna-se possível demonstrar as possibilidades analíticas no contexto de evolução de serviços. A análise desses indicadores ajudará o provedor de serviços a tomar decisões levando em consideração os requisitos supracitados. Além disso, apesar de a abordagem proposta ser restrita a indicadores financeiros e de uso, ela também pode ser estendida para indicadores de naturezas e perspectivas distintas, de acordo com diferentes requisitos de decisão relevantes para os provedores de serviços. O processo de inclusão de novos indicadores e perspectivas de análise segue o mesmo processo aqui descrito.

Os KPIs de uso e financeiros propostos neste trabalho são apresentados na Tabela 5.1 e detalhados nas subseções seguintes.

Tabela 5.1: KPIs considerando a perspectiva de uso e financeira dos serviços.

Perspectiva	KPIs simples	KPIs derivados
Uso	<ul style="list-style-type: none"> • Quantidade de requisições diretas • Quantidade de requisições indiretas • Número de aplicações cliente 	<ul style="list-style-type: none"> • Total de requisições
Financeira	<ul style="list-style-type: none"> • Receita • Estimativa de gastos de infraestrutura • Custos com penalidades 	<ul style="list-style-type: none"> • Lucro

5.2.1 KPIs de Uso

Considerando o contexto de uso do serviço, o provedor tem a possibilidade de mensurar a quantidade de requisições para cada serviço, detalhando as requisições por versão ou mesmo para cada operação do serviço. É possível distinguir as requisições compondo indicadores que representem as requisições diretas e indiretas. O primeiro tipo representa o consumo direto de uma operação de um serviço por um cliente, enquanto o segundo tipo abrange o consumo indireto de uma operação de um serviço específico. Analisar indicadores de requisições indiretas permite que o provedor estime o impacto da mudança de acordo com o tipo de uso que se faz do serviço, expandindo os limites da análise baseada no cenário de pior caso do impacto da mudança e orientando estratégias para a evolução do serviço.

Também é possível derivar um indicador que represente a quantidade total de requisições, como segue:

$$TotalRequisições = RequisiçõesDiretas + RequisiçõesIndiretas$$

O provedor pode basear a sua análise tanto na quantidade geral de requisições para um serviço quanto no número de requisições diretas e indiretas. Essas informações auxiliam o provedor a decidir por desativar ou não um serviço, por exemplo.

Outra métrica, baseada no uso e que também auxilia a tomada de decisão, abrange a determinação do número de aplicações clientes que realizam requisições. Essa métrica é útil quando se analisa as requisições agregadas, i.e., por grupo de cliente.

5.2.2 KPIs Financeiros

Indicadores financeiros referem-se a mensurações de aspectos financeiros de uma organização. Esses indicadores oferecem uma percepção precisa sobre a saúde dos negócios e tipicamente são alvos de análise pelos responsáveis pelas decisões estratégicas de uma organização. No contexto de serviços, um dos KPIs primordiais se refere à receita, que representa o quanto foi recebido pelo provimento do serviço. O valor que cada provedor recebe sofre influência de diversos fatores, como as características de cada serviço, as quais demandam um tipo de cobrança específico para cada serviço, ou fatores estratégicos, como a adequação aos preços de concorrentes.

Já a estimativa de gastos com infraestrutura consolida os gastos gerados pela provisão dos serviços. Tal estimativa é importante para determinar o quanto é vantajoso manter uma determinada versão de serviço disponível. Analisar esse tipo de indicador é essencial em um cenário no qual o provedor de serviços possui diversas versões disponíveis ao mesmo tempo. Se por um lado essa prática é necessária para evitar incompatibilidades com clientes usando versões antigas, por outro lado, a criação de versões implica em mais gastos com manutenção e se torna uma desvantagem para o provedor no médio/longo prazo.

Propõe-se também um indicador para determinar os custos com penalidades devido ao não cumprimento de cláusulas de SLA (disponibilidade do serviço, por exemplo). Esse indicador é importante para identificar serviços críticos e possivelmente candidatos a sofrerem mudanças.

Para determinar o lucro obtido com os serviços, uma métrica indicando o lucro do serviço é derivada a partir dos KPIs financeiros supracitados, como segue:

$$\text{Lucro} = \text{Receita} - (\text{GastosInfraestrutura} + \text{CustoPenalidades})$$

O provedor estima o lucro obtido com um serviço/versão considerando a diferença entre a receita obtida *versus* as despesas com infraestrutura e as penalidades causadas pelo serviço/versão.

Esses indicadores foram escolhidos por serem os mais representativos para o contexto de serviços, mensurando variáveis que são relevantes na tomada de decisão durante a gestão da evolução.

5.3 Identificação da Granularidade dos Dados

A modelagem de uma visão integrada e unificada de métricas e perspectivas de negócios demanda:

- a identificação do menor nível de granularidade dos dados disponíveis;
- a determinação das vantagens e desvantagens em representar as informações em um determinado nível de agregação/detalhe, o que varia de acordo com as necessidades de tomada de decisão.

Nas subseções seguintes, discute-se a disponibilidade dos dados na sua menor granularidade, o que influencia a modelagem do DW.

5.3.1 Dados de Uso

As informações sobre o uso dos serviços representam as interações entre clientes e as operações de uma versão específica de um serviço. Embora esse tipo de informação

seja extremamente importante para enriquecer a análise, coletar tais dados é uma tarefa complicada devido à natureza distribuída dos serviços. Algumas técnicas, como o monitoramento, a interceptação e o registro (*logging*) das requisições de clientes, são aplicáveis à tarefa de coleta de dados de uso (CHUVAKIN; PETERSON, 2009).

A escolha da abordagem utilizada para monitorar as interações serviço-cliente influencia o escopo de dados extraídos (desde versões até operações específicas do serviço), os custos e a capacidade de monitoramento. Conseqüentemente, esses fatores afetam o nível de detalhe da informação disponível para análise, assim como também afetam as métricas que são representadas/combinadas para a tomada de decisão. Por exemplo, a coleta de dados de uso no nível de versão do serviço, sem o detalhamento pelas operações requisitadas, não permite que se tenha a percepção sobre as principais operações a partir da perspectiva do cliente. Nesse caso, assume-se um cenário impreciso, no qual um serviço dispara requisições para outros em uma cadeia de serviços. Por outro lado, capturar e representar informações no nível de operação do serviço possibilita entender exatamente quais operações os clientes utilizam mais, o que auxilia diversas decisões, tais como orientar as políticas de reformulação dos serviços, derivar métricas relacionadas a mudanças profundas, determinar a compatibilidade orientada ao uso, e assim sucessivamente.

Neste trabalho, considera-se que os dados sobre requisições diretas e indiretas são obtidos a partir do módulo Gerenciador de Perfis (WS-EVOLV), conforme discutido no Capítulo 4.

Com a modelagem proposta neste trabalho, a análise de dados de uso deve ser detalhada por operação de versão de serviço e por cliente, o que possibilita representar a quantidade de requisições diretas e indiretas de algum cliente em um momento específico para uma operação de uma versão de um serviço. Essa informação pode ser agregada de diferentes maneiras, por serviço/versão e/ou por grupo de clientes.

5.3.2 Dados Financeiros

A receita quantifica o quanto o provedor obtém por meio da cobrança pelo consumo do serviço. Os tipos de cobrança variam de acordo com diferentes fatores. A Tabela 5.2 exemplifica as formas de cobrança utilizadas pelos principais serviços do portfólio AWS.

Tabela 5.2: Métodos de cobrança utilizados em alguns serviços do portfólio AWS.

Serviço	Foco	Método de Cobrança
Amazon FPS	Terceirização de pagamentos	<ul style="list-style-type: none"> • Percentual sobre cada transação
Amazon S3	Armazenamento de dados	<ul style="list-style-type: none"> • Período de uso • Quantidade de dados armazenada
Amazon RDS	Bases de dados relacionais	<ul style="list-style-type: none"> • Período de Uso • Quantidade de dados armazenada • Quantidade de dados transferidos
Amazon EC2	Elasticidade computacional	<ul style="list-style-type: none"> • Consumo (por hora e por tráfego)

Com toda essa diversidade de métodos de cobrança, a análise dos mesmos de maneira unificada e integrada só é possível quando os dados sobre a receita são transformados e levados para uma unidade comum. Essa unidade comum deve ser capaz de abstrair os distintos métodos de cobrança, mas deve também permitir que ainda seja possível identificar cada método de cobrança associado à receita. A grande vantagem disso é que se torna possível comparar e analisar métricas financeiras independentemente do método de cobrança.

A receita é detalhada por versão do serviço, por tipo de cobrança e por cliente, podendo ser agregada por serviço e/ou por grupo de cliente. Importante notar que, diferentemente das métricas de uso, que possuem granularidade no nível de operação do serviço, a receita tem granularidade no nível de versão do serviço. Isso ocorre porque é bastante incomum que provedores cobrem os seus clientes por meio de preços diferenciados para cada operação do serviço: o preço é aplicado à versão do serviço.

Quando se considera as métricas de custos, as quais, no contexto de serviços, consolidam gastos com a provisão do portfólio de serviços, estas são divididas em gastos com penalidades causadas por descumprimentos de SLAs e gastos com infraestrutura.

As penalidades geradas pela violação de termos estabelecidos pelos SLAs são relacionadas a clientes específicos e aos seus respectivos contratos de uso. Essas penalidades tipicamente representam recompensações para os clientes, ou em forma de ressarcimentos financeiros com o oferecimento de algum tipo de bônus (por exemplo, uso sem cobrança pelo mesmo período em que o serviço ficou indisponível), e consequentemente representam prejuízos financeiros para o provedor, os quais precisam ser mensurados e analisados no processo de mudança dos serviços.

Os custos com a provisão do serviço são mais complexos para representar, pois tais custos são uma composição de custos fixos (p. ex., o mesmo custo é atribuído para todos os clientes) ou variáveis (p. ex., o custo é distribuído para cada cliente considerando algum fator que diferencie o valor atribuído), o que varia de acordo com as características do serviço ou seguindo as políticas organizacionais. A determinação de custos de provisão no contexto de portfólio de serviços não é uma tarefa trivial, uma vez que ela implica na definição de matrizes de dependência e vários coeficientes que são difíceis de estabelecer. Frequentemente, simplificações são realizadas nessa estrutura de custos, o que implica em um rateio igualitário de custos entre grupos de clientes ou no balanceamento de custos de acordo com o volume de requisições de cada cliente, por exemplo. Essas simplificações tornam o processo de análise mais simples e, geralmente, as possíveis discrepâncias causadas por tais simplificações são cobertas pela margem de lucro. Portanto, mecanismos de rateio de custos são aplicados para prover a integração dos diferentes tipos de custos com as métricas financeiras discutidas anteriormente. Além disso, é essencial ser capaz de combinar custos e receita para derivar os indicadores de lucratividade dos serviços.

Propõe-se neste trabalho a utilização de uma função de rateio apropriada para o contexto de provisão de serviços, o que possibilita o detalhamento das métricas de custo e lucratividade por versão do serviço, por tipo de cobrança e por cliente. A agregação pode ocorrer por serviço e/ou por grupo de cliente.

5.4 Modelagem Dimensional

Uma modelagem dimensional foi elaborada para proporcionar uma visão integrada e unificada dos indicadores financeiros e de uso, preservando as diferenças importantes para a análise.

São propostos os seguintes indicadores de análise: (i) Receita, Custos com violações de SLA, Custos com Infraestrutura e Lucro, na perspectiva financeira, e (ii) Requisições Diretas, Requisições Indiretas e Total de Requisições e Contagem de Clientes, na perspectiva de uso.

Considerando as análises apresentadas na Seção 5.3, este trabalho propõe as seguintes perspectivas de análise para as métricas de uso e financeiras: Cliente, Serviço, *Status*, Tipo de Cobrança e Tempo. A Tabela 5.3 sumariza a relação entre estas dimensões e as métricas propostas.

Tabela 5.3: Escopo e granularidade das métricas financeiras e de uso dos serviços.

Dimensão	Requisições Diretas e Indiretas		Receita, Custos e Lucro	
	Dimensão de análise	Grão	Dimensão de análise	Grão
Tempo	Sim	Mês	Sim	Mês
Cliente	Sim	Cliente	Sim	Cliente
Serviço	Sim	Operação	Sim	Versão
<i>Status</i>	Sim	<i>Status</i>	Sim	<i>Status</i>
Tipo de cobrança	Não	-	Sim	Tipo Cobrança

A partir da Tabela 5.3, é possível verificar que:

- as métricas financeiras e de uso possuem várias dimensões de análise em comum;
- algumas dimensões de análise de análise são específicas em relação a algumas métricas, como é o caso da dimensão Tipo de Cobrança em relação às métricas financeiras;
- algumas dimensões, apesar de serem comuns, quando associadas às métricas devem ser consideradas em granularidade diferente, como é o caso da dimensão serviço, que para as métricas de uso tem granularidade no nível de operação, mas para as métricas financeiras a granularidade só vai até o nível de versão.

Com essas constatações, verifica-se que o modelo MFTS, o qual é o mais tradicionalmente usado para modelar temas múltiplos e inter-relacionados, é o mais adequado para o cenário de gestão do ciclo de vida de serviços.

Este trabalho propõe duas tabelas fato principais: FATO_FINANCEIRA e FATO_USO. Tais tabelas agrupam KPIs similares quanto à natureza (conforme descrito na Tabela 5.1), e são relacionadas com as suas dimensões correspondentes, descritas na Tabela 5.4.

Tabela 5.4: Dimensões no contexto de serviços.

	Dimensão Conformada	Descrição
Tempo	Sim	Descreve o momento em que o fato ocorreu, ou seja, quando a requisição a uma operação foi realizada ou o período referente à receita/despesa.
Cliente	Sim	Descreve os clientes do portfólio de serviços associados ao fato (que fizeram uma requisição ou geraram uma receita/despesa).
Serviço_por_Versão, Serviço_por_Operação	Parcialmente	Descreve os serviços que compõem o portfólio, com granularidade até o nível de versão e operação, respectivamente.
Status_do_Serviço	Sim	Descreve o status inerente a cada versão/operação de um serviço no momento em que o fato ocorreu.
Tipo_de_cobrança	Não	Descreve os diferentes tipos de cobrança inerentes a uma versão de um serviço.

A representação em um modelo dimensional possibilita combinar informações financeiras e de uso por meio de dimensões comuns, preservando diferenças como as distintas granularidades em relação ao serviço/operação, a habilidade de analisar as informações sobre uso no nível de operação, ou a perspectiva de diferentes tipos de cobrança com relação aos dados financeiros. A visão geral do modelo de DW é representada pela Figura 5.2 e é detalhada ao longo desta seção.

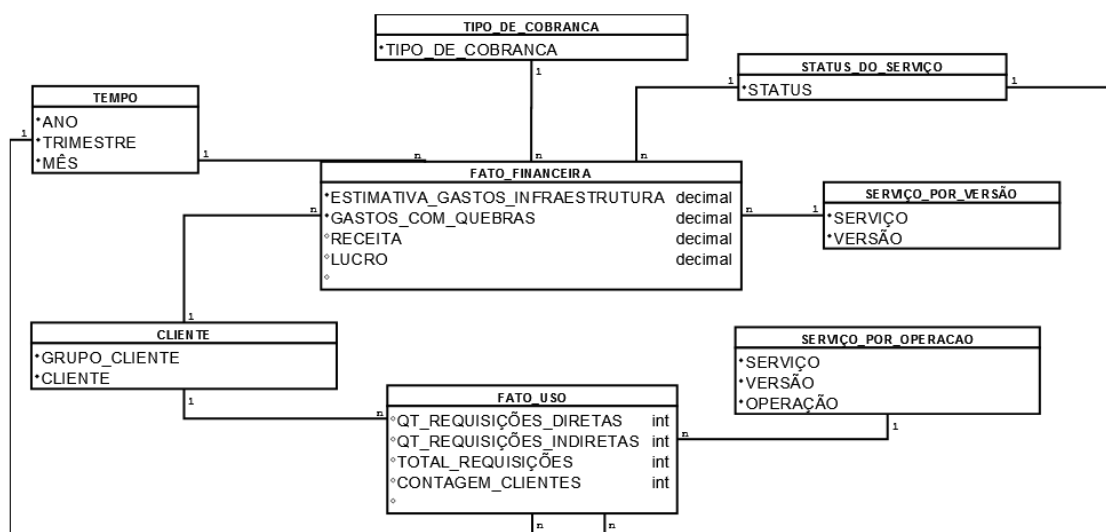


Figura 5.2: O modelo MTFS proposto para o DW.

A dimensão Cliente representa aqueles que fazem as requisições para os serviços e que são uma fonte de receitas e/ou despesas para o provedor. Conforme ilustrado na Figura 5.3, essa dimensão se relaciona às duas tabelas fato e é composta por uma

hierarquia de dois níveis: cliente e grupo de clientes. A determinação de grupos de clientes baseia-se na ideia de agrupar clientes de acordo com algum critério de similaridade. Diversos critérios são usados para realizar o agrupamento (*e.g.* região geográfica ou importância estratégica). Outras dimensões poderiam ser utilizadas para caracterizar os clientes, como a localização geográfica ou o segmento de negócios. Nesse caso, essas dimensões são adicionadas ao modelo e relacionadas às tabelas fato.

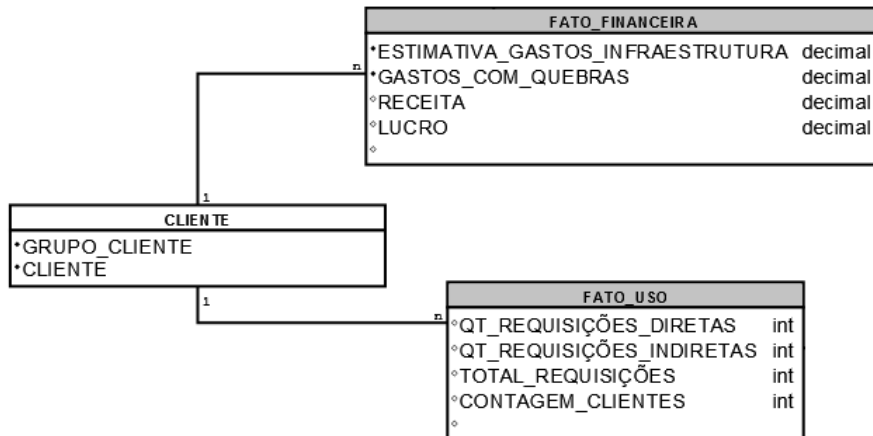


Figura 5.3: A dimensão CLIENTE relacionada às duas tabelas fato.

A dimensão *Status* do Serviço é utilizada para indicar o estágio do serviço no ciclo de vida de provisão (*e.g.* serviço ativo, obsoleto ou desativado). Conforme representado na Figura 5.4, essa dimensão tem um único nível, *Status*, e se relaciona com as duas tabelas fato. As informações oriundas dessa dimensão são usadas em uma perspectiva de análise relacionada ao uso (*e.g.* quantos clientes estão consumindo uma versão obsoleta) ou em relação a uma análise financeira. Nesse último caso, ter esse tipo de informação é especialmente interessante para analisar o lucro obtido com uma versão obsoleta ou para alinhar a estratégia organizacional com os resultados obtidos com um serviço que ficar ativo.

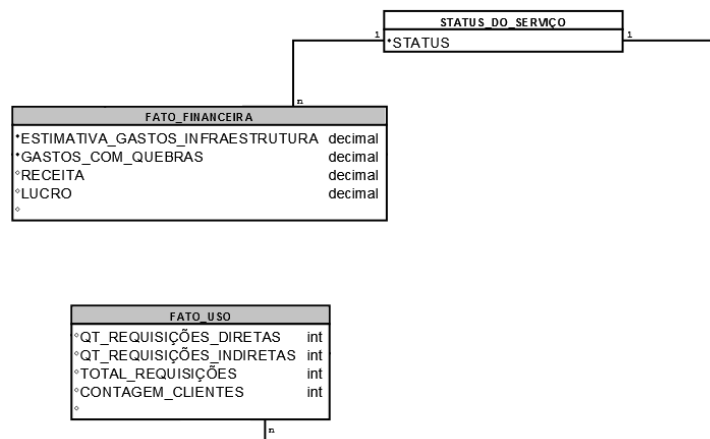


Figura 5.4: A dimensão STATUS_DO_SERVICO relacionada às duas tabelas fato.

Outra dimensão comum deveria ser a dimensão de Serviço. No entanto, conforme discutido anteriormente, as métricas financeiras e de uso dos serviços têm diferentes granularidades de análise, por versão e por operação, respectivamente (Tabela 5.3). Dessa maneira, para analisar o uso, a hierarquia da dimensão serviço é composta por operação, versão e serviço. No entanto, para analisar as métricas financeiras, a dimensão

Serviço tem versão como nível mais detalhado. Para consolidar essas informações, cada tabela fato foi relacionada a uma dimensão de serviço com uma granularidade específica: `SERVIÇO_POR_VERSÃO` e `SERVIÇO_POR_OPERAÇÃO`, conforme ilustra a Figura 5.5. Assim, essas dimensões são similares quando a análise de uso ocorre no nível agregado (por versão ou por serviço) e, ao mesmo tempo, preserva-se a possibilidade de realizar a análise dos indicadores de uso no nível de operação.

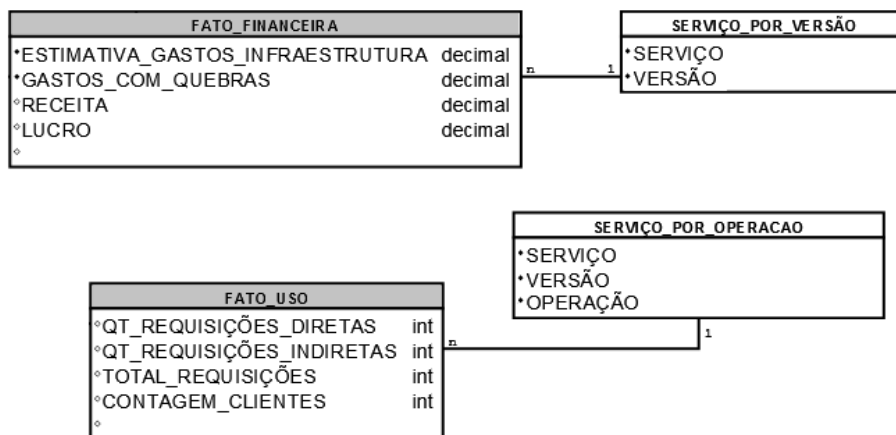


Figura 5.5: A dimensão `SERVIÇO_POR_VERSÃO` relacionada à `FATO_FINANCEIRA` e a dimensão `SERVIÇO_POR_OPERAÇÃO` relacionada à fato de uso.

A dimensão Tempo é comum para as duas fatos, conforme ilustra a Figura 5.6. A dimensão Tempo indica o período de tempo que agrupa requisições diretas e indiretas ou a receita, o lucro e as despesas que foram geradas no período. Sugere-se o ano como maior grão da dimensão, podendo detalhar-se as informações por trimestre e por mês, menor grão da dimensão. Neste trabalho, a escolha pelo grão mês se deve pela conformidade com a natureza dos dados financeiros, já que geralmente a cobrança por um serviço é realizada mensalmente, o que indica que a sumarização dos dados por mês é a alternativa mais apropriada. No entanto, a granularidade poderia ser detalhada ainda mais (*e.g.* por quinzena, semana ou dia), de acordo com a necessidade de análise do provedor. Mesmo a análise em uma granularidade ainda menor de tempo poderia ser considerada devido à natureza das informações de uso (*e.g.* requisições por hora), mas novamente chega-se à constatação de que essa granularidade dificilmente faria sentido para a análise unificada de dados de uso e financeiros. Por fim, se realmente essa situação se tornasse uma necessidade, deveria ser adotada a mesma alternativa usada para a dimensão Serviço, como já discutido.

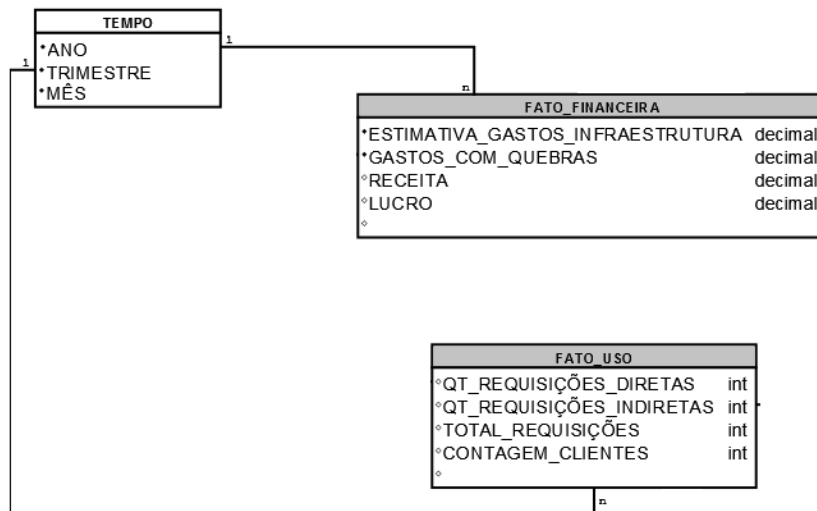


Figura 5.6: A dimensão TEMPO relacionada às duas tabelas fato.

Por fim, há dimensões que são específicas apenas para o contexto financeiro, como a dimensão Tipo de Cobrança. Essa dimensão possui apenas um nível e é restrita à tabela FATO_FINANCEIRA (Figura 5.7). Com o uso dessa dimensão, é possível adequar o modelo aos diversos tipos de cobrança que existem, tornando possível a análise desse tipo de dado em uma unidade de análise comum.

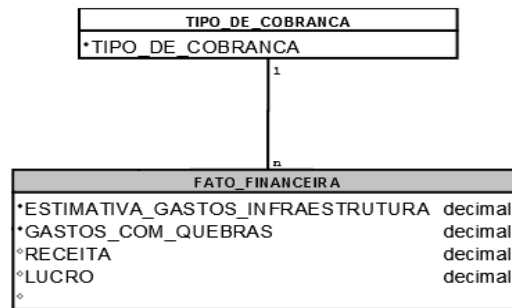


Figura 5.7: A dimensão TIPO_DE_COBRANCA relacionada apenas a uma fato.

5.5 Considerações Finais

Este capítulo apresentou a modelagem do DW para o ambiente de BI proposto neste trabalho, o qual considera a combinação de métricas financeiras e de uso no apoio à tomada de decisões na gestão de evolução de serviços em larga escala. A modelagem objetiva a representação das perspectivas financeira e de uso por meio de uma modelagem dimensional adequada às características dessas informações.

O modelo apresentado pode ser enriquecido com outras tabelas fato, com a adição de novos indicadores de impacto e de novas dimensões, caso mais detalhes de análise sejam necessários para domínios específicos. Nesse caso, tudo com relação à abordagem proposta aqui é aplicável, desde a identificação de KPIs, a disponibilidade e a granularidade dos dados, até o detalhamento/agregação da análise no modelo dimensional.

6 ARQUITETURA ETL

Este capítulo descreve a arquitetura de ETL desenvolvida para lidar com as características da integração de informações no contexto de serviços, discutindo os tipos de fontes de dados que são inerentes ao contexto de portfólio de serviços e as alternativas para lidar com a heterogeneidade de tipos e fontes de dados.

6.1 Visão Geral

A visão geral da arquitetura de ETL proposta neste trabalho é representada na Figura 6.1 e é composta por três áreas: Área de Fontes de Dados, Área *Staging* e Área de Carga das Dimensões e Fatos. Essas áreas são discutidas no restante deste capítulo.

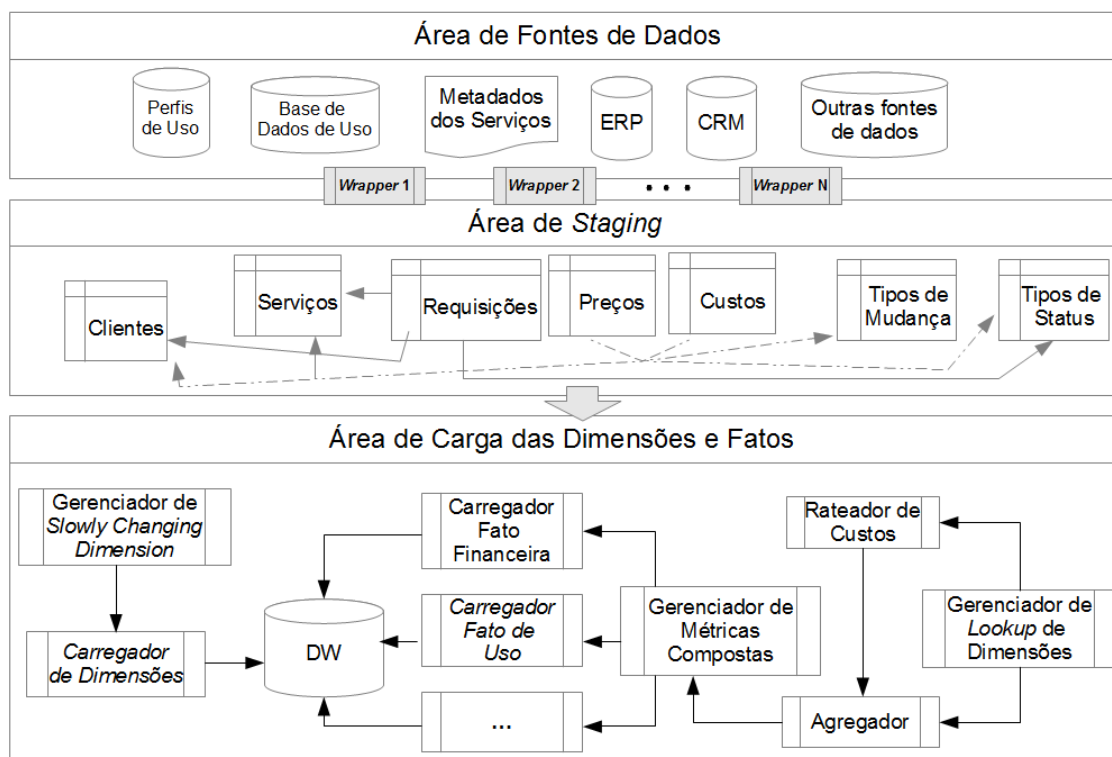


Figura 6.1: Visão geral da arquitetura ETL.

6.2 Identificação de Fontes de Dados

Esta etapa abrange a identificação das fontes de dados disponíveis na organização, com o objetivo de identificar quais devem ser utilizadas para extrair os dados necessários para a construção do DW.

No contexto de portfólio de serviços, uma das principais características é a falta de padrões e processos bem estabelecidos para o armazenamento de dados operacionais. Essa falta de padronização se deve especialmente por se tratar de uma área de negócios relativamente recente, e que demanda aplicações operacionais diferentes, as quais possuem seus próprios repositórios com modelos de dados específicos. A natureza distribuída dos serviços faz com que os dados sobre os serviços acabem ficando também distribuídos por diferentes fontes de dados.

A área de Fontes de Dados, ilustrada pela Figura 6.2, elenca os principais repositórios de dados utilizados pelos provedores de serviços para armazenar os dados relacionados aos negócios e aos serviços, tais como:

- bases de dados de mineração de uso e de perfis de uso;
 - repositório com os perfis gerados por processos que determinam padrões de consumo, por exemplo.
- metadados de serviços;
 - saídas de algoritmos que analisam as especificações WSDL dos serviços, determinando compatibilidade entre versões, por exemplo.
- sistemas ERP;
 - dados financeiros sobre as receitas e as despesas obtidas com os serviços, por exemplo.
- sistemas de CRM;
 - informações sobre clientes que contratam e consomem os serviços, por exemplo.
- outras fontes de dados:
 - aplicações operacionais internas;
 - personalizadas para atender às necessidades específicas do provedor de serviços, como a composição de modelos de cobrança para os serviços ou um ambiente para a orquestração de serviços, por exemplo.
 - SLAs;
 - especificações das cláusulas estabelecidas entre provedor e cliente, por exemplo.

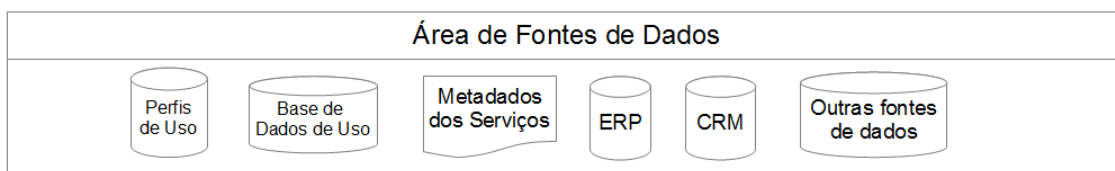


Figura 6.2: A área de fontes de dados da arquitetura ETL.

A arquitetura ETL foi projetada para interagir com essas e com outras fontes de dados que possam existir no ambiente do provedor de serviços. Essa adaptabilidade a diferentes fontes é possível com a utilização de *wrappers* específicos para cada fonte de dados, como detalhado na próxima seção.

6.3 Utilizando Wrappers para a Extração dos Dados

Não existe uma padronização quanto à escolha das aplicações ou em relação ao modo como os provedores armazenam os dados. Além disso, alguns tipos de dados podem nem existir, o que leva à necessidade de derivar tais dados a partir dos dados brutos nas fontes de dados existentes.

Para integrar as diferentes fontes de dados na área de *staging*, a arquitetura proposta neste trabalho baseia-se em mecanismos de extração utilizando *wrappers*. Isso significa que, para cada fonte de dados identificada, constrói-se um *wrapper* capaz de extrair os dados de tal fonte, de acordo com o formato dos dados e as especificidades de conexão com a fonte de dados. Um *wrapper* pode ser aplicável a uma ou mais fontes de dados, de acordo com a semelhança entre as mesmas.

Uma rotina de extração de dados deve coordenar a execução dos *wrappers*, viabilizando o sincronismo de extração e a consolidação dos dados na área de *staging*. Com a abordagem baseada em *wrappers*, o processo se torna mais adaptável para a adição de novas e diversas fontes de dados, sem afetar o modelo normalizado na área de *staging*, tornando a arquitetura flexível e possibilitando que a mesma possa ser adotada em distintos ambientes de provisão de serviços.

A área de *staging* é construída de acordo com as dimensões e fatos previstos para o DW. Isso facilita o processo de carga, evitando que inclusões/alterações de fontes de dados implique em mudanças estruturais na área de *staging*. Neste trabalho, a área de *staging* é criada com as tabelas *Clientes*, *Serviços*, *Tipos de mudança*, *Tipos de status*, *Requisições*, *Preços* e *Custos*, espelhando arquitetura projetada para o DW.

A Figura 6.3 demonstra o fluxo de dados a partir da área de Fontes de Dados, que contém as fontes de dados identificadas, passando pela criação de *wrappers* para a extração dos dados (a quantidade de *wrappers* varia em função da quantidade e variedade de fontes de dados), os quais carregam as informações na área de *staging*.

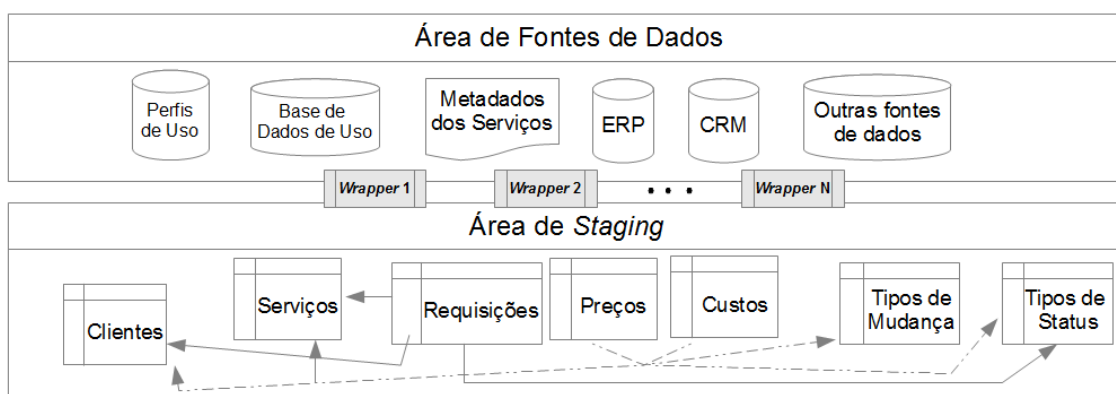


Figura 6.3: Os *wrappers* e a área de *staging* da arquitetura ETL.

6.4 Carga das Dimensões e Fatos

Os dados armazenados na área de *staging* têm a vantagem de possibilitar a abstração da heterogeneidade e discrepâncias dos dados brutos originais. Na Área de Carga das Dimensões e Fatos, que tem como base a área de *staging*, ocorre a limpeza e a transformação dos dados, os quais são convertidos para um modelo multidimensional e carregados no DW.

Os processos de transformação e carga dividem-se em dois fluxos distintos: um responsável pela carga das dimensões do DW e outro responsável pela carga das tabelas fato. Os dois fluxos são ilustrados na Figura 6.4, com os componentes de carga das dimensões à esquerda do DW e os componentes de carga das tabelas fato à direita do DW. Os componentes de cada fluxo são detalhados nas próximas subsecções.

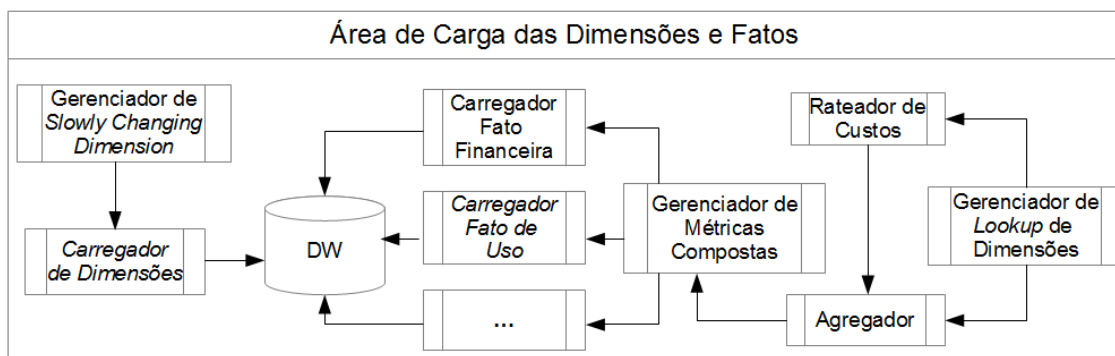


Figura 6.4: A área de carga das dimensões e tabelas fato da arquitetura ETL.

6.4.1 Dimensões

O processo de carga das dimensões é composto por dois componentes: o gerenciador de *Slowly Changing Dimension* e o Carregador de Dimensões.

O gerenciador de *Slowly Changing Dimension* tem por responsabilidade:

- validar os dados quanto à integridade e conformidade dos mesmos;
- aplicar estratégias para a alteração estrutural da dimensão.

Ter uma estratégia para cenários de alterações estruturais em dimensões é útil para permitir que o responsável pela decisão tenha a possibilidade de visualizar o que mudou na estrutura de uma dimensão. No domínio de serviços, o componente de *slowly changing dimension* trata dimensões que sofrem alterações estruturais com alguma frequência, como as dimensões *SERVICO_POR_VERSAO/OPERACAO*, que mudam quando versões e operações de serviços são criadas/descontinuadas, e a dimensão *CLIENTE*, que muda quando grupos de clientes são criados/alterados/removidos, de acordo com os resultados gerados pelos processos de detecção de padrões de consumo.

Na arquitetura proposta nesta dissertação, as estratégias devem ser parametrizadas pelo provedor de serviços, que pode optar por (i) sobrescrever os dados antigos, tornando as alterações estruturais transparentes para o usuário final; ou (ii) manter o histórico de alterações estruturais em campos específicos em cada dimensão, os quais podem ser explorados pelas ferramentas analíticas. A escolha da estratégia deve atender às necessidades analíticas acordo a alternativa que for a mais adequada para visualizar as alterações estruturais.

O componente Carregador de Dimensões tem por responsabilidade realizar a carga de cada dimensão, após a limpeza e transformação dos dados.

6.4.2 Tabelas Fato

O fluxo de carga das tabelas fato possui um conjunto de componentes para transformar os dados mensuráveis e realizar a carga das fatos.

O componente gerenciador de *Lookup* de Dimensões tem como principais finalidades:

- a) garantir a consistência do DW, verificando a integridade dos dados a serem carregados nas tabelas fato com os dados nas dimensões existentes;
- b) garantir que somente entradas válidas sejam inseridas no DW (i.e., nunca deve existir um fato que faça referência a um membro que não existe na dimensão referenciada).

Após o *lookup* de Dimensões, os KPIs de custos devem ser tratados pelo componente de *Rateador de Custos*. Esse componente precisa ser utilizado quando o provedor não consegue determinar os custos de infraestrutura por cliente. O Rateador de custos aplica funções para realizar o rateio de custos entre um conjunto de clientes, tais como:

- a) rateio fixo, que distribui os custos homogeneamente entre os clientes;
- b) rateio variável, que utiliza o volume de requisições de cada cliente para distribuir os custos proporcionalmente.

Na arquitetura proposta nesta dissertação, deve-se configurar, via parâmetros, se o rateio de custos deve ser aplicado. Se o rateio for utilizado, deve-se indicar qual função de rateio deve ser aplicada.

O componente *Agregador* se aplica a todos os KPIs (oriundos do componente Rateador de Custos ou não) e tem como função agrupar os mesmos de acordo com os níveis de análise inerentes a cada fato. O componente é aplicado sobre os dados brutos disponíveis no grão mais detalhado do que o menor nível da tabela fato correspondente. Por exemplo, requisições diretas/indiretas devem ser agrupadas por mês, o menor nível da dimensão Tempo, mesmo que os dados brutos estejam disponíveis no nível dia.

O componente *Gerenciador de Métricas Compostas* é utilizado para a criação de métricas derivadas a partir das métricas simples existentes. Este trabalho propõe a criação de duas métricas compostas:

- Lucro (Receita – Gastos com Quebras – Gastos com Infraestrutura), na tabela fato financeira;
- Total de Requisições (Requisições Diretas + Requisições Indiretas), na tabela fato de uso.

A arquitetura proposta conta com componentes de carga específicos para cada fato. O último passo na área de carga de dimensões e fatos envolve a utilização dos componentes *Carregador Fato Financeira* e *Carregador Fato de Uso* para carregar as respectivas tabelas fato. Outros componentes devem ser criados caso novas tabelas fato sejam inseridas no modelo, mantendo o mesmo fluxo de transformação (*lookup*, rateio de custos/agregação e composição de métricas).

6.5 Considerações Finais

Este capítulo apresentou a arquitetura de ETL elaborada para lidar com a idiosincrasia das fontes de dados no contexto de serviços. A arquitetura é composta por *wrappers* que realizam a normalização dos dados em uma área de *staging* e por

componentes responsáveis pela transformação e consolidação dos dados nas dimensões e fatos do DW.

A arquitetura apresenta como grandes vantagens a flexibilidade e a adaptabilidade, pelo fato de ser aplicável a diferentes cenários mediante a adição de componentes específicos para manipular fontes de dados distintas. Essas características são fundamentais no imprevisível ambiente de um provedor de serviços, o qual pode possuir distintas aplicações e sistemas para armazenamento dos dados.

7 ESTUDO DE CASO

Este capítulo apresenta um estudo de caso que tem como objetivo demonstrar como a abordagem proposta neste trabalho é capaz de apoiar diferentes necessidades de tomada de decisão durante a evolução de serviços. Inicialmente, a partir da ilustração apresentada na Seção 1.1, detalha-se como esse cenário é composto e, na sequência, são discutidos alguns dilemas de decisão inerentes ao cenário apresentado.

7.1 Cenário AWS

Uma base com dados sintéticos inspirados no ambiente de provimento de serviços Amazon AWS foi construída para compor o cenário utilizado no presente estudo de caso. O portfólio AWS foi escolhido por incluir serviços de diferentes características, com formas de cobrança distintas, e com níveis distintos de complexidade, i.e., com poucas ou várias operações, características que se adequavam ao estudo de caso.

Foram escolhidos cinco serviços para representar o portfólio AWS: *Amazon Flexible Payments Service* (FPS), *Amazon Simple Storage Service* (S3), *Amazon Elastic Compute Cloud* (EC2), *Amazon SimpleDB*, e *Amazon Relational Database Service* (RDS). A Tabela 7.1 apresenta as características de cada serviço.

Tabela 7.1: Os serviços AWS selecionados para o estudo de caso.

Serviço	Segmento	# Versões ²	# Operações ³	Depende de	Cobrança por
FPS	Pagamentos	3	25	SimpleDB RDS	Transação
S3	Armazenamento	45	16	EC2	Armazenamento
EC2	Computação na nuvem	41	137		Consumo
SimpleDB	Base de Dados	20	10	S3	Armazenamento; Consumo
RDS	Base de Dados	31	28	S3 EC2	Consumo; Transf. dados; Armazenamento

² <http://aws.amazon.com/releases/notes/>

³ <http://aws.amazon.com/products-solutions/>

7.2 Dados Sintéticos

Considerando o conjunto de serviços apresentado, foram criados dados sintéticos para simular dados relacionados ao histórico de requisições e resultados financeiros obtidos com os serviços em um período específico. A utilização de dados sintéticos se deve pela dificuldade em conseguir dados reais de natureza estratégica. Os dados sintéticos também possibilitam a simulação de todas as possibilidades analíticas que se deseja demonstrar, pois não há restrições quanto à divulgação de resultados que possam revelar dados organizacionais sigilosos.

A primeira etapa para a simulação dos dados envolveu a análise da política de manutenção dos serviços da AWS, a partir de um estudo realizado sobre o histórico de atualizações da AWS⁴. Com a análise da especificação WSDL dos serviços AWS, foi possível determinar as versões e operações de cada serviço. Assumindo que serviços e clientes dependentes não são obrigados a migrar imediatamente para a versão mais recente do serviço, tem-se um cenário em que inúmeras versões de um mesmo serviço podem existir concorrentemente, com seus próprios clientes e custos de provisão.

Na etapa seguinte, foi criada uma lista de clientes fictícia para representar o conjunto de clientes dos serviços analisados. Esses clientes foram agrupados em perfis que representam padrões de uso dos serviços. A relação cliente/serviço foi elaborada conforme demonstra a Tabela 7.2.

Tabela 7.2: Requisições Diretas e Indiretas de cada Grupo de Cliente

Serviço	Grupos que realizam requisições diretas	Grupos que realizam requisições indiretas
FPS	A, C	
S3	B, C	A, D
EC2	A, B, C, D	
SimpleDB	B, D	C
RDS	D	A, C

Considerando os grupos/clientes e serviços/versões/operações do portfólio, foram criados os seguintes arquivos de dados:

- lista de serviços, versões e operações;
- lista de grupos de clientes e clientes;
- lista com valores aleatórios para preço original, preço cobrado e estimativa de gastos relacionados com a infraestrutura necessária para manter cada serviço e gastos com violações de SLAs, relacionados a cada versão de serviço, no período de janeiro a dezembro de 2012, considerando o status do serviço (ativo, depreciado e desativado) e os tipos de cobrança;
- lista com a quantidade de requisições diretas/indiretas por operação de cada versão de serviço, no período de janeiro a dezembro de 2012, também considerando o status do serviço (ativo, depreciado e desativado).

⁴ <http://aws.amazon.com/releases/>

Esses dados são integrados em um DW, o qual serve como base para apoiar as necessidades de tomada de decisão em cenários de abrangem mudanças em serviços da AWS.

7.3 Implementação

Para implementar a abordagem de BI proposta neste trabalho, foram utilizadas as seguintes ferramentas:

- Microsoft Integration Services 2010, para realizar o processo de ETL;
- Microsoft SQL Server 2005, como base de dados para armazenar o DW;
- IBM Cognos 8.3, para a construção do ambiente de análise, incluindo:
 - Cognos Framework Manager, para a modelagem dos metadados e criação de camadas de negócios e análise;
 - Cognos Transformer, para a criação do cubo OLAP para análise;
 - Cognos Report Studio, para a criação dos relatórios analíticos.

O ambiente foi construído em uma máquina rodando Windows 7, com processador Intel Core 2 Quad de 2,8 GHz, com memória de 3 GB. Parte do ambiente (banco de dados e ambiente de análise) foi integrado em uma máquina virtual rodando Windows XP.

7.4 Recursos Analíticos

Com a utilização de recursos analíticos tradicionais, torna-se possível visualizar as informações da maneira mais adequada à tomada de decisão por meio de diversas ferramentas que se adequam às diferentes necessidades de decisão. A suíte Cognos, por exemplo, possui recursos para a construção (*Cognos Transformer*) e a visualização de cubos (*Cognos PowerPlay* e *Cognos Analysis Studio*), além de ferramentas para construção de relatórios e *dashboards* (*Cognos Metrics Studio* e *Cognos Report Studio*).

Com a utilização recursos para a exploração de cubos, por exemplo, torna-se possível o fatiamento e a navegação pela hierarquia de informações representadas em um cubo de dados, o que aumenta a rapidez para detalhar/sumarizar uma informação. Por exemplo, o provedor pode analisar a quantidade de requisições por serviço, podendo detalhar a sua análise pelos níveis de versão e operação.

Criar relatórios com análises específicas possibilita a visualização de um cenário útil que o responsável pela tomada de decisão pode acessar facilmente. Um *dashboard*, composto por diversos cenários distintos em uma visão única, também representa um instrumento importante para apoiar decisões mais complexas e que dependem da análise de diferentes variáveis de negócios.

Nas próximas duas seções, os problemas de decisão discutidos no Capítulo 1 são retomados e analisados considerando o uso de alguns dos recursos analíticos supracitados.

7.5 Cenário 1: Análise do Impacto de Mudanças Incompatíveis

Tipicamente, provedores necessitam determinar os possíveis impactos resultantes de mudanças em versões de serviços disponíveis. Esse primeiro caso abrange um cenário em que, para tomar a decisão sobre a mudança, o provedor precisa analisar as seguintes questões:

- Qual o impacto de uma mudança incompatível em um determinado serviço do portfólio (por exemplo, o serviço S3)?
- Quais grupos de clientes são estrategicamente mais importantes para organização, os quais devem ser priorizados e minimamente afetados pela mudança?

Como já foi mencionado, assume-se o uso de ferramentas analíticas disponíveis em ambientes de BI, tais como *dashboards*, relatórios e tabelas *pivot*. O provedor pode utilizar um *dashboard*, por exemplo, para compor diferentes perspectivas de análise. A Figura 7.1 ilustra tal *dashboard*, o qual:

- identifica como os diferentes grupos de clientes utilizam os serviços, conforme ilustrado no gráfico de pizzas na parte superior da Figura 7.1;
- analisa a relação entre o lucro, as despesas, e o volume de requisições gerados pelos distintos grupos de clientes que consomem o serviço S3, conforme demonstra o gráfico de bolhas na parte inferior da Figura 7.1.

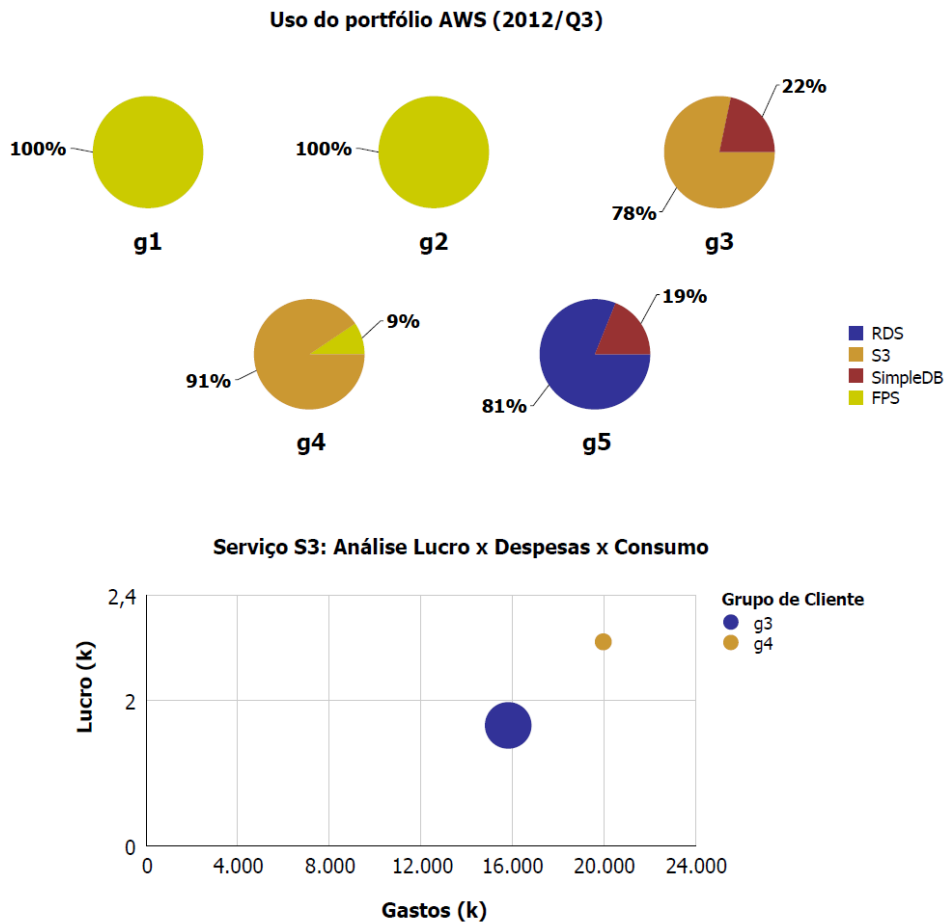


Figura 7.1: *Dashboard* relacionando indicadores financeiros e de uso.

Essa perspectiva consolidada permite que o provedor observe que dois grupos de clientes distintos utilizam o serviço S3: g3 e g4. No gráfico de bolhas, com a análise centrada no consumo do serviço S3 pelos grupos g3 e g4, o provedor pode observar que, apesar de os clientes que compõem o grupo g3 fazerem mais requisições para o serviço S3, essas requisições geram menos lucro do que as requisições realizadas pelos clientes do grupo g4. Portanto, considerando a perspectiva de melhor retorno financeiro para a organização, os clientes do grupo g4 devem ser priorizados na realização da mudança.

Essas perspectivas de análise ilustram a importância do cruzamento de indicadores de diferentes naturezas para apoiar a tomada de decisão estratégica, indo além da análise de uso baseada no pior caso e aproximando-se mais das necessidades de decisão na perspectiva de negócios.

7.6 Cenário 2: Análise da Rentabilidade de um Serviço

Determinar quando uma versão de um serviço deve ser desativada não é uma tarefa trivial para o provedor de serviços. Porém, essa é uma situação inevitável em diversas ocasiões, especialmente quando se deseja reduzir custos. Nesse contexto, ter mecanismos que indiquem qual versão pode ser desativada (seja pela subutilização ou pela falta de lucratividade obtida com o provimento da versão) possibilita uma tomada de decisão mais embasada e alinhada com as estratégias de negócios estabelecidas pelo provedor de serviços. Portanto, a necessidade de decisão passa pela seguinte questão:

- Quais versões de serviços do portfólio são as melhores candidatas a serem desativadas?

A Figura 7.2 ilustra uma tabela *pivot* que relaciona as três versões de cada serviço que possuem as piores relações entre volume de requisições e lucro. Os valores em vermelho ilustram versões de serviços que geram prejuízos para o provedor de serviços.

Service	Version	Quantity of Requests	Profit (\$)
EC2	v17	2.225.545	-5.681,59
	v30	2.225.087	117,11
	v40	2.235.583	1.758,72
FPS	v1	184.115	730,70
	v2	189.468	-4.422,02
	v3	187.973	-931,63
RDS	v26	184.741	-1.826,81
	v5	184.896	-375,61
	v7	184.857	-1.079,30
S3	v31	148.134	-3.578,12
	v4	148.907	-2.457,41
	v41	148.393	-1.819,67
SimpleDB	v14	86.692	-1.139,18
	v16	87.392	-1.330,77
	v17	91.947	-186,42

Figura 7.2: Tabela *pivot* relacionando as versões que apresentam as piores relações entre consumo e lucratividade.

Para o serviço EC2, por exemplo, é fácil notar que a versão v17 é a melhor candidata a ser desativada, por apresentar o pior resultado financeiro. No entanto, se o provedor analisasse apenas o uso, essa determinação seria mais complicada, dado que as versões analisadas possuem uso semelhante. Além disso, outras versões podem apresentar peculiaridades que levem o provedor a incluir novas perspectivas de análise para tomar a decisão, como a análise por status do serviço ou pelo tipo de cobrança.

7.7 Considerações Finais

Os cenários ilustrados neste estudo de caso dão uma breve demonstração de como os provedores de serviços podem utilizar a abordagem proposta neste trabalho. O grau de detalhamento dependerá da necessidade analítica relacionada com a decisão que se pretende tomar no ciclo de vida de serviços.

Vários outros tipos de análise, como análises detalhadas por operações de serviços ou a relação entre uso/receita para as últimas versões lançadas, podem ser realizados com as diversas ferramentas analíticas que exploram o DW proposto nesta abordagem.

8 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho apresentou uma abordagem de BI para apoiar a tomada de decisão de negócios na gestão da evolução de serviços, considerando a perspectiva do provedor de serviços. O presente trabalho integra o framework WS-EVOLV, o qual é voltado para a análise da evolução de serviços considerando a perspectiva de uso. Essa integração representa uma importante extensão ao WS-EVOLV, relacionando informações que abrangem o uso dos serviços, geradas pelos demais módulos do framework, com as necessidades de decisão em uma perspectiva de negócios. A principal contribuição deste trabalho está na proposição de uma abordagem capaz de representar métricas para atender aos diferentes requisitos de decisão, o que aprimora o apoio ao processo de tomada de decisão do provedor de serviços em uma perspectiva estratégica e de negócios.

A abordagem proposta possibilita avaliar o impacto da mudança por meio da análise de indicadores financeiros e de uso, os quais provêm percepções sobre as consequências das mudanças nos negócios. Para a modelagem do DW foi utilizado o modelo constelação de fatos (MTFS), o qual possibilitou a representação de métricas e dimensões de maneira conformada, preservando a habilidade de analisar os fatos de acordo com os seus aspectos comuns e também analisar os mesmos de acordo com dimensões específicas. A arquitetura de ETL foi projetada para lidar com fontes heterogêneas, agregações e mecanismos de rateio de custos. Essa arquitetura foi construída para adequar-se à falta de padrões e práticas de negócios típicas no cenário de organizações que provêm serviços em grande escala. Um estudo de caso demonstrou as possibilidades analíticas com a combinação de diferentes tipos de indicadores, melhorando a visão que o tomador de decisão tem do seu cenário atual e dando subsídios para a tomada de decisão na evolução de serviços em uma perspectiva de negócios.

Uma das limitações encontradas durante o trabalho tangeu a disponibilização de dados reais para a construção do estudo de caso. No entanto, apesar de os dados utilizados serem sintéticos, eles são inspirados em propriedades que são típicas de cenários reais. O uso de indicadores financeiros e de uso apoia dilemas de decisão que os provedores tipicamente enfrentam, mas a abordagem vai além desses indicadores específicos. Outras classes de indicadores relevantes e de domínio específico podem ser adotadas de acordo com as necessidades de decisão do provedor. Para a inclusão de novos indicadores, a mesma abordagem de análise de granularidade, modelagem multidimensional e inserção em uma estrutura ETL são aplicáveis.

Ao longo deste trabalho, foram realizadas algumas publicações com os seguintes resultados parciais alcançados:

1. SILVA, E. et al. A business intelligence approach to support decision making in service evolution management. In: Services Computing (SCC), 2012 IEEE Ninth International Conference on. IEEE, 2012. p. 41-48.
2. SILVA, E.; GALANTE, R.; BECKER, K. Uma abordagem utilizando Business Intelligence para apoiar o processo de tomada de decisão na gestão da evolução de serviços web. In: WORKSHOP DE TESES E DISSERTAÇÕES EM BANCO DE DADOS, WTDBD, 11, SIMPÓSIO BRASILEIRO DE BANCO DE DADOS, SBBD, 27, Anais... São Paulo: SBC, 2012. p. 69-74.
3. SILVA, E.; BECKER, K.; GALANTE, R. Supporting Strategic Decision Making on Service Evolution Context Using Business Intelligence. In: Services Computing (SCC), 2013 IEEE 10th International Conference on. IEEE, 2013. No prelo 2013.

O tratamento de necessidades de decisão de negócios, as quais vão além das necessidades de decisões técnicas tradicionais, torna-se cada vez mais importante com os crescentes investimentos em aplicações baseadas em serviços em larga escala. Como trabalhos futuros, pretende-se explorar a análise de perspectivas relacionadas à compatibilidade e ao versionamento, derivando métricas que representam aspectos técnicos como indicadores de impacto para o provedor de serviços. Também se considera a possibilidade de explorar a aplicação de Data Warehouses multiversão para a representação de cenários *what-if*, com a aplicação de análise de sensibilidade sobre o impacto das mudanças de acordo com hipóteses, o que vai de encontro às novas tendências de BI. Por fim, também se pretende avaliar o desempenho considerando cenários de big data, com a aplicação da abordagem proposta neste trabalho em cenários reais de provedores de serviços de larga escala.

REFERÊNCIAS

ANDRIKOPOULOS, V.; BENBERNOU, S.; PAPAZOGLU, M. P. On the evolution of services. *Software Engineering, IEEE Transactions on*, v. 38, n. 3, p. 609-628, 2012.

BECKER, K.; RUIZ, D. D.; SANTOS, K. MF-Retarget: Aggregate Awareness in Multiple Fact Table Schema Data Warehouses. *ADBIS Research Communications*, p. 41-51, 2002.

BECKER, K. et al. Automatically determining compatibility of evolving services. In: *Web Services, 2008. ICWS'08. IEEE International Conference on*. IEEE, 2008. p. 161-168.

BENATALLAH, B. et al. Service mosaic: A model-driven framework for web services life-cycle management. *Internet Computing, IEEE*, v. 10, n. 4, p. 55-63, 2006.

BROWN, K.; ELLIS, M. Best Practices for Web services Versioning. Disponível em: <http://www.ibm.com/developerworks/webservices/library/ws-version/>, 2004. Acesso em janeiro de 2013.

CHEN, H.; CHIANG, R.H.L.; STOREY, V.C. Business intelligence and analytics: from big data to big impact. *MIS Quarterly*, v. 36, n. 4, p. 1165-1188, 2012.

CHUVAKIN, A.; PETERSON, G. Logging in the age of web services. *Security & Privacy, IEEE*, v. 7, n. 3, p. 82-85, 2009.

ENDREI, M. et al. Moving forward with web services backward compatibility. Disponível em: <http://www.ibm.com/developerworks/java/library/ws-soa-backcomp>. 2006. Acesso em março de 2013.

FANG, R. et al. A version-aware approach for web service directory. In: *Web Services, 2007. ICWS 2007. IEEE International Conference on*. IEEE, 2007. p. 406-413.

FRANK, D. et al. Using an interface proxy to host versioned web services. In: *Services Computing, 2008. SCC'08. IEEE International Conference on*. IEEE, 2008. p. 325-332.

GARDNER, S. R. Building the Data Warehouse. *Communications of the ACM*, v. 41, n. 9, 52-60, 1998.

GOLFARELLI, M.; RIZZI, S.; CELLA, I. Beyond data warehousing: what's next in business intelligence?. In: *Proceedings of the 7th ACM international workshop on Data warehousing and OLAP*. ACM, 2004. p. 1-6.

GU, Q.; PARKIN, M.; LAGO, P. A taxonomy of service engineering stakeholder types. In: *Towards a Service-Based Internet*. Springer Berlin Heidelberg, 2011. p. 206-219.

KAJKO-MATTSSON, M.; LEWIS, G. A.; SMITH, D. B. A framework for roles for development, evolution and maintenance of soa-based systems. In: *Systems Development in SOA Environments, 2007. SDSOA'07: ICSE Workshops 2007. International Workshop on. IEEE, 2007. p. 7-7.*

KIMBALL, R.; ROSS, M. *The data warehouse toolkit - 2nd.* New York, John Willey & Sons, 2002.

LEITNER, P. et al. End-to-end versioning support for web services. In: *Services Computing, 2008. SCC'08. IEEE International Conference on. IEEE, 2008. p. 59-66.*

LÖNNQVIST, A.; PIRTTIMÄKI, V. The measurement of business intelligence. *Information Systems Management*, v. 23, n. 1, p. 32-40, 2006.

MOSS, L. T.; ATRE, S. *Business intelligence roadmap: the complete project lifecycle for decision-support applications.* Addison-Wesley Professional, 2003.

NIEMANN, M. et al. Structuring SOA Governance. *International Journal of IT/Business Alignment and Governance (IJITBAG)*, v. 1, n. 1, p. 58-75, 2010.

PAPAZOGLU, M. P. Service-oriented computing: Concepts, characteristics and directions. In: *Web Information Systems Engineering, 2003. WISE 2003. Proceedings of the Fourth International Conference on. IEEE, 2003. p. 3-12.*

PAPAZOGLU, M. P. The challenges of service evolution. In: *Advanced Information Systems Engineering.* Springer Berlin Heidelberg, 2008. p. 1-15.

PAPAZOGLU, M. P.; ANDRIKOPOULOS, V.; BENBERNOU, S. Managing evolving services. *Software, IEEE*, v. 28, n. 3, p. 49-55, 2011.

PARMENTER, D. *Key performance indicators (KPI): developing, implementing, and using winning KPIs.* Wiley, 2010.

POE, V.; KLAUER, P.; BROBST, S. *Building a Data Warehouse for Decision Support* 2nd edition. Prentice Hall, 1998.

SILVA, E. et al. A business intelligence approach to support decision making in service evolution management. In: *Services Computing (SCC), 2012 IEEE Ninth International Conference on. IEEE, 2012. p. 41-48.*

SILVA, E.; GALANTE, R.; BECKER, K. Uma abordagem utilizando Business Intelligence para apoiar o processo de tomada de decisão na gestão da evolução de serviços web. In: *WORKSHOP DE TESES E DISSERTAÇÕES EM BANCO DE DADOS, WTDBD, 11, SIMPÓSIO BRASILEIRO DE BANCO DE DADOS, SBBD, 27, Anais... São Paulo: SBC, 2012. p. 69-74.*

SILVA, E.; BECKER, K.; GALANTE, R. Supporting Strategic Decision Making on Service Evolution Context Using Business Intelligence. In: *Services Computing (SCC), 2013 IEEE 10th International Conference on. IEEE. No prelo 2013.*

SILVEIRA, P. S.; BECKER, K.; RUIZ, D. D. SPDW+: a seamless approach for capturing quality metrics in software development environments. *Software Quality Journal*, v. 18, n. 2, p. 227-268, 2010.

TREIBER, M.; TRUONG, H.; DUSTDAR, S. On analyzing evolutionary changes of web services. In: *Service-Oriented Computing–ICSOC 2008 Workshops*. Springer Berlin Heidelberg, 2009. p. 284-297.

VOLLINO, B.; BECKER, K. A Framework for Web Service Usage Profiles Discovery. In: *Web Services (ICWS), 2013 IEEE 20th International Conference on*. IEEE. No prelo 2013.

W3C Working Group. *Web Services Architecture*. Disponível em: <http://www.w3.org/TR/ws-arch/>. 2011. Acesso em outubro de 2011.

WOOLF, B. Introduction to SOA governance–Governance: The official IBM definition, and why you need it. IBM. Disponível em: <http://www.ibm.com/developerworks/library/ar-servgov/>, 2006. Acesso em março de 2013.

YAMASHITA, M.; BECKER, K.; GALANTE, R. Service Evolution Management Based on Usage Profile. In: *Web Services (ICWS), 2011 IEEE 18th International Conference on*. IEEE, 2011. p.746-747.

YAMASHITA, M. et al. Measuring change impact based on usage profiles. In: *Web Services (ICWS), 2012 IEEE 19th International Conference on*. IEEE, 2012. p. 226-233.

YAMASHITA, M.; BECKER, K.; GALANTE, R. A Feature-based Versioning Approach for Assessing Service Compatibility. *JIDM*, [S.l.], v.3, n.2, p.120–131, 2012.