



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
CIÊNCIAS DA COMPUTAÇÃO

Enzo Gomes Sônego

Desenvolvimento de um Software de Business Intelligence que modela o processo
de desenvolvimento de uma empresa baseado no padrão de modelagem conceitual
de Conta/Transação

Florianópolis – SC

2026

Enzo Gomes Sônego

Desenvolvimento de um Software de Business Intelligence que modela processos de
negocio, baseado no padrão de modelagem conceitual de Conta/Transação

Trabalho de Conclusão de Curso submetido ao curso de Ciências da Computação da Universidade Federal de Santa Catarina para a obtenção do título de Bacharel em Ciências da Computação.

Orientador(a): Prof. Raul Sidnei Wazlawick, Dr.

Universidade Federal de Santa Catarina

Centro Tecnológico - CTC

Departamento de Informática e Estatística

Ciências da Computação

Orientador: Prof. Raul Sidnei Wazlawick, Dr.

Florianópolis – SC

2026

Sônego, Enzo Gomes

Desenvolvimento de um Software de Business Intelligence
que modela processos de negocio, baseado no padrão de
modelagem conceitual de Conta/Transação / Enzo Gomes
Sônego ; orientador, Raul Sidnei Wazlawick, 2026.

95 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro
Tecnológico, Graduação em Ciências da Computação,
Florianópolis, 2026.

Inclui referências.

1. Ciências da Computação. 2. Gargalo. 3. Business
Intelligence. 4. Conta/Transação. I. Wazlawick, Raul
Sidnei. II. Universidade Federal de Santa Catarina.
Graduação em Ciências da Computação. III. Título.

Enzo Gomes Sônego

Desenvolvimento de um Software de Business Intelligence que modela processos de
negocio, baseado no padrão de modelagem conceitual de Conta/Transação

Este Trabalho de Conclusão de Curso foi julgado aprovado para a obtenção do
Título de Bacharel em Ciências da Computação, e aprovado em sua forma final pelo
Curso de Ciências da Computação da Universidade Federal de Santa Catarina.

Local Florianópolis, 15 de Junho de 2026.

Coordenação do Curso

Banca examinadora

Prof. Raul Sidnei Wazlawick, Dr.

Orientador

Prof.(a) Carina Friedrich Dorneles, Dr.(a)

Avaliadora

Universidade Federal de Santa Catarina – UFSC

Prof. Renato Fileto, Dr.

Avaliador

Universidade Federal de Santa Catarina – UFSC

Florianopolis, 2024.

in memoriam

*Enio Sônego, Wilde Carlos Gomes, Roger Albino,
vocês continuam vivos em minhas memórias e coração*

AGRADECIMENTOS

Agradeço primeiramente ao Prof. Raul, por ter dado a ideia durante a aula no ano de 2020 enquanto introduzia o padrão, a aqueles que aguentaram minhas campanhas e personagens de D&D, e aos meus familiares e meus amigos que mantiveram a minha sanidade durante a jornada acadêmica.

Declaro que durante a elaboração deste trabalho acadêmico utilizei ferramentas de inteligência artificial com o objetivo de realizar revisão gramatical, ortográfica e de estilo de escrita (ChatGPT, Claude), assim como sugestões de código (Cursor). Todo o conteúdo gerado pela ferramenta foi submetido a uma rigorosa curadoria, verificação de fatos e revisão de autoria própria, assumo total responsabilidade pelas informações e conclusões apresentadas.

RESUMO

Um gargalo em um processo empresarial representa uma limitação que pode afetar a eficiência de uma empresa, resultando em atrasos, menor produtividade e aumento de custos. Gargalos não mitigados podem causar o enfileiramento de tarefas, a ociosidade de determinado recurso, diminuição da qualidade do produto e impactos negativos na reputação da empresa. Ferramentas de Business Intelligence (BI) podem ajudar a identificar e resolver estes gargalos, permitindo a análise de dados e a tomada de decisão baseada em fatos.

Este trabalho se propõe a criar um protótipo de um software de BI para a análise das vantagens do padrão conta/transação, e documentar o processo de modelagem e desenvolvimento, analisando a aplicabilidade do padrão conta/transação de modelar processos empresariais de variadas áreas.

Palavras-chave: Gargalo; Business Intelligence; Conta/Transação.

ABSTRACT

A bottleneck in a business process represents a limitation that can affect a company's efficiency, resulting in delays, reduced productivity, and increased costs. Unmitigated bottlenecks can create task queues, resource idleness, decreased product quality, and negative impacts on a company's reputation. Business Intelligence (BI) tools can help identify and mitigate these bottlenecks by enabling data analysis and evidence-based decision-making.

This project aims to create a BI software prototype to demonstrate the advantages of the Account/Transaction pattern, to document the modeling and development process, and to examine the applicability of the Account/Transaction model for representing business processes across various domains.

Keywords: Bottleneck; Business Intelligence; Account/Transaction.

LISTA DE FIGURAS

Figura 1 – Classes Conta/Transação	25
Figura 2 – Estado Inicial Exemplo	26
Figura 3 – Pedido de Compra Criado Exemplo.....	27
Figura 4 – Modelagem Completa Exemplo	28
Figura 5 – Modelagem (JAMALADDIN AZIZOV, 2021)	35
Figura 6 – Casos de Uso Software	40
Figura 7 – Diagrama de Classes Software	41
Figura 8 – Representacao de Conta.....	43
Figura 9 – Transação conectando Contas.....	44
Figura 10 – Editor de Conta.....	45
Figura 11 – Exemplo Produtor – Consumidor.....	46
Figura 12 – UI de Simulação.....	50
Figura 13 – UI Análise de Simulação.....	51
Figura 14 – Exemplo Padaria.....	54

LISTA DE TABELAS

Tabela 1. Objetivos de BI em setores variados	22
Tabela 2. Comparação entre os trabalhos relacionados e a proposta deste.....	38
trabalho	
Tabela 3 - <i>Comparação entre Conta/Transação e o Modelo Relacional.....</i>	52
<i>Tradicional</i>	

LISTA DE ABREVIATURAS E SIGLAS

BI – Business Intelligence

UML – Unified Modeling Language

DSS – Decision Support Systems

API – Application Programming Interface

CEO – Chief Executive Officer

KPI – Key Performance Indicators

JSON – Javascript Object Notation

SUMÁRIO

1 – Introdução e Objetivos	16
1.1 – Introdução	16
1.2 – Justificativa	17
1.3 – Objetivos Gerais	18
1.4 – Objetivos Específicos	18
1.5 – Estrutura do Documento	19
2 – Fundamentação Teórica	20
2.1 – Business Intelligence	20
2.1.1 – Definições e Relevância	20
2.1.2 – Exemplos de Aplicações	21
2.2 – Conta/Transação	24
2.3 – Conclusão do Capítulo	29
3 – Trabalhos Relacionados	31
3.1 – Produções Acadêmicas	31
3.2 – Softwares no Mercado	35
3.3 – Conclusão do Capítulo	38
4 – Desenvolvimento	40
4.1 – Prova de Conceito	40
4.2 – Implementação	41
4.2.1 – Criando Contas	42
4.2.2 – Conectando Contas (Criando Transações)	43
4.2.3 – Selecionando Contas	44
4.2.4 – Executando Transações	45
4.2.5 – Persistencia do Modelo	47
4.2.6 – Simulacao do Modelo (Detecção de Gargalos)	47
4.2.7 – Análise da Simulação (Sugestão de Correção)	50
4.2.8 – Exemplo Padaria	53
4.3 – Discussão dos Resultados	55
4.3.1 – Viabilidade da Modelagem Conta/Transação	53
4.3.2 – Comportamento da Sim. e Detecção de Gargalos	53
4.3.3 – Ameaças à Validade	58
4.3.4 – Limitações do Protótipo	59
4.3.5 – Possibilidades de Evolução	60
4.3.6 – Problema das Subclasses	60
5 – Conclusão e Trabalhos Futuros	62
Referências	64

APÊNDICE A – Código fonte da aplicação desenvolvida no projeto	66
APÊNDICE B - Artigo no formato SBC	67

1 INTRODUÇÃO E OBJETIVOS

1.1 INTRODUÇÃO

Em uma empresa, a eficiência organizacional é frequentemente limitada pela presença de gargalos em seus processos. Um gargalo é aquele recurso cuja capacidade é igual ou menor que a demanda colocada nele (GOLDRATT; COX, 1984), causando o enfileiramento de tarefas, ociosidade de recursos e atrasos na entrega. Além de aumentar os custos operacionais com horas extras e prejudicar a qualidade do produto final, essas ineficiências impactam negativamente a experiência do cliente e a motivação dos colaboradores.

Para mitigar esses problemas, gestores dependem de dados precisos para prever prazos e tomar decisões. É nesse contexto que o Business Intelligence (BI) se destaca. Segundo Reinschmidt e Francoise (2000), o BI torna o processo decisório mais rápido e fundamentado em fatos. Ferramentas de BI permitem não apenas a análise do estado atual do negócio, mas também a simulação de cenários futuros para testar estratégias de mitigação e realizar *benchmarks*.

No entanto, o grande desafio para a implementação de um sistema de BI eficiente é a complexidade e a heterogeneidade dos dados gerados por diferentes setores de uma empresa. Para que o BI consiga identificar gargalos com precisão, os dados dos processos precisam estar estruturados de forma padronizada. É aqui que entra o padrão de modelagem conceitual de Conta/Transação, definido por Wazlawick (2015).

Poderia o padrão Conta/Transação atuar como um tradutor geral para os processos de negócio? A resposta que este trabalho propõe se baseia em quatro propriedades do padrão que o tornam especialmente adequado como fundamento

para sistemas de BI. Primeiro, ao representar qualquer processo como movimentações de saldos entre contas, o padrão impõe uma estrutura de dados uniforme a domínios heterogêneos, eliminando a necessidade de transformações para compatibilizar fontes distintas. Segundo, as métricas fundamentais para detecção de gargalos, sendo elas: saldo; taxa de execução; e taxa de bloqueio de transações, são intrínsecas ao próprio padrão, dispensando a definição manual de KPIs para cada domínio. Terceiro, a possibilidade de simular o modelo antes da coleta de dados históricos habilita uma forma de BI preditivo inacessível às ferramentas tradicionais. Quarto, a criação de novas contas e transações em tempo de execução, (ou no caso do protótipo produzido, na mesma sessão de trabalho) permite que o modelo seja adaptado a novos processos sem alterações estruturais no sistema.

Ao modelar qualquer atividade empresarial, seja a movimentação de estoque, alocação de horas de funcionários ou fluxo de pedidos, estritamente como transferências de valores entre "contas", os dados perdem sua complexidade isolada e ganham uma estrutura uniforme. Essa padronização viabiliza a criação de um motor de simulação genérico e flexível.

A meta deste trabalho é, portanto, a criação de um protótipo de software de BI que utilize a ampla aplicabilidade do padrão Conta/Transação como base arquitetural. O objetivo é demonstrar como essa modelagem alimenta o software e potencializa a identificação e resolução de gargalos organizacionais.

1.2 JUSTIFICATIVA

O desenvolvimento de soluções de BI exige fontes de dados consistentes para gerar análises confiáveis. Muitas empresas enfrentam dificuldades para integrar

dados de diferentes naturezas ao tentar mapear, diagnosticar e resolver gargalos operacionais em seus fluxos de trabalho.

A principal justificativa para este projeto reside na proposta de unir a capacidade analítica do BI com a simplificação e padronização estrutural oferecida pelo padrão Conta/Transação (WAZLAWICK, 2015). Enquanto grande parte das ferramentas de BI disponíveis no mercado foca predominantemente na visualização de dados em dashboards após o processamento, este trabalho propõe que a modelagem transacional dos processos ocorra na base do sistema, servindo como o motor da análise.

Ao implementar um protótipo de software fundamentado nessa premissa, o trabalho busca exemplificar, na prática, a ampla aplicabilidade do padrão Conta/Transação para processos administrativos. A documentação deste processo de modelagem e desenvolvimento servirá para validar a viabilidade técnica de usar o padrão como um alicerce de dados robusto, estabelecendo uma base sólida para a criação de futuras ferramentas de BI mais ágeis e adaptáveis a diferentes realidades empresariais.

1.3 OBJETIVOS GERAIS

Desenvolvimento de um Software de Business Intelligence que modela processos de negocio, baseado no padrão de modelagem conceitual de Conta/Transação

1.4 OBJETIVOS ESPECÍFICOS

- 1) Avaliar a aplicabilidade do padrão Conta/Transação como fundamento para sistemas de BI
- 2) Projetar e implementar um protótipo que integre modelagem Conta/Transação, simulação de eventos discretos e análise heurística de gargalos.
- 3) Verificar a viabilidade técnica da abordagem, discutindo os limites observados e os requisitos para estendê-la a outros domínios de processo.

1.5 ESTRUTURA DO DOCUMENTO

Nesta seção inicial, o trabalho é introduzido, assim como são abordadas a justificativa e os objetivos. Na seção 2 são apresentados conceitos relevantes, e as tecnologias que serão utilizadas. A seção 3 é dedicada a apresentar trabalhos relacionados, e como este trabalho se diferencia de outros similares. Na seção 4 é apresentado o desenvolvimento do software e suas funcionalidades, são discutidos os resultados obtidos e as limitações encontradas. Por fim, a seção 5 mostra as conclusões e apresenta propostas para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 BUSINESS INTELLIGENCE

2.1.1 Definições e relevância

Segundo Sezões, Oliveira e Baptista (2006), Business Intelligence engloba um vasto conjunto de aplicações de apoio à tomada de decisão que possibilitam acesso rápido, compartilhado e interativo às informações, bem como sua análise e manipulação.

Como mencionado por Reinschmidt e Francoise (2000), a consolidação e organização de dados com o objetivo de melhorar a tomada de decisão pode gerar vantagem competitiva. O conhecimento sobre como identificar essas vantagens e utilizá-las constitui a base do Business Intelligence. Mais do que a combinação de dados e tecnologia, o BI ajuda pessoas a criar conhecimento a partir de grandes volumes de informação, desde que se obtenham os dados corretos, se extraia deles valor e esse valor seja compartilhado. Dessa forma, BI transforma informação em conhecimento ao colocar os dados certos nas mãos das pessoas certas, no momento adequado, oferecendo suporte efetivo ao processo decisório.

Ainda segundo Sezões, Oliveira e Baptista (2006), a área de BI surge nas empresas acompanhando tendências como:

- Grande intensidade concorrencial na maioria dos mercados;
- Exigência crescente por padrões elevados de qualidade em produtos e serviços;
- Necessidade de compreender mercados e clientes cada vez mais segmentados e específicos;

- Demanda por racionalização de processos internos e redução de custos operacionais;
- Imperativo de avaliar, em tempo real, o desempenho organizacional, garantindo decisões em tempo útil;
- Necessidade de conhecer, controlar e mitigar riscos de negócio associados a cada atividade.

Além dessas tendências, observa-se o movimento de horizontalização das empresas, que se tornam menos hierarquizadas, adotam downsizing e concedem maior autonomia aos níveis operacionais. Esses processos exigem que informações críticas sejam disseminadas para um número maior de colaboradores, especialmente gestores.

Todos esses fatores contribuem para que as empresas busquem identificar o que é essencial em seu negócio por meio de diagnóstico, análise, acessibilidade, compartilhamento e elaboração de relatórios, transformando o volume de dados coletados em mudanças que geram resultados — isto é, em conhecimento útil de forma eficaz.

Na subseção seguinte, serão discutidas as origens dos sistemas de suporte à decisão, bem como apresentados exemplos de aplicações que os utilizam em diferentes setores do mercado.

2.1.2 Exemplos de Aplicações

Alguns dos setores que mais usam sistemas de BI incluem, corretoras de ações, seguradoras, bancos e empresas do setor financeiro, telecomunicações, e empresas de manufatura (OLSZAK, C. M., & ZIEMBA, E., 2004), a tabela a seguir,

traduzida de (OLSZAK, C. M., & ZIEMBA, E., 2004) e baseada em (REINSCHMIDT, & FRANCOISE, 2000), lista os objetivos das aplicações de BI utilizadas em diversos setores.

Tabela 1. Objetivos de BI em setores variados

Setor	Objetivos
Varejo	• Previsão de Demanda. Usa dados provenientes da digitalização para prever a demanda e com base nesta previsão, definir requisitos de estoque • Pedidos e Reabastecimento. Usa informações para tomar decisões mais rapidamente sobre itens a serem pedidos e otimizar quantidades • Marketing. Fornece análise das transações de clientes (o que está vendendo e quem está comprando) • Distribuição e Logística. Ajuda os centros de distribuição a gerenciar volumes maiores e pode usar informações antecipadas de remessa para programar e consolidar entrada e saída de produtos • Gestão de transportes. Ajuda no desenvolvimento de planos ideias de consolidação de carga e cronogramas de rotas a serem tomadas. • Planejamento de Estoque. Auxilia na identificação do nível de estoque necessário garantindo um grau de serviço
Seguradoras	• Análise de Sinistros e Prêmios. Providência a capacidade de analisar reivindicações detalhadas e históricos máximos por produto, política, tipo de revindicação e outras especificações • Análise de Clientes. Permite analisar as necessidades do cliente e os padrões de uso do produto, ajuda a

	desenvolver programas de marketing com base nas características do cliente, analisar os riscos e melhorar o atendimento ao cliente • Análise de Risco. Ajuda a Identificar segmentos de mercado de alto risco e oportunidades em segmentos específicos, relacionando segmentos de mercado, e reduzindo a frequência de sinistros
Financeiro	• Análise de Rentabilidade do Cliente. Auxilia a determinar a lucratividade geral de clientes, atual e de longo prazo, fornecendo a base para vendas de lucro alto e relacionamento com o banco, maximizando as vendas para clientes de alto valor, reduzindo custos para clientes de baixo valor, fornecendo os meios para maximizar a lucratividade de novos produtos e serviços • Administração de Crédito. Apoia o estabelecimento de padrões de progressão de problemas de crédito por classe e tipo de cliente, alertando os clientes para evitar problemas de crédito, ajudando na gestão de limites de crédito, na avaliação da carteira de crédito do banco e na redução de perdas de crédito • Vendas de Filiais. Melhora o atendimento ao cliente e a venda de contas e facilita a venda cruzada, melhorando o suporte ao cliente e fortalecendo a fidelidade a empresa
Telecomunicações	• Categorização e Segmentação de Clientes. Contribui para a determinação de perfis de produtos e segmentos de clientes que geram mais lucro, para o fornecimento de perfis de clientes detalhados e integrados, para o desenvolvimento de programas individualizados de chamadas frequentes e a determinação de futuras necessidades de clientes

	• Previsão da Demanda do Cliente. Facilita a previsão das necessidades futuras de produtos ou serviços, fornecendo base para a análise do controle de rotatividade para melhorar a retenção de clientes
Manufatura	• Vendas. Permite analisar dados de transações específicas de clientes • Previsão. Ajuda na previsão da demanda e requisitos de estoque • Pedidos e Reabastecimento. Permite encomendar quantidades ideais de itens • Compra. Ajuda os centros de distribuição a gerenciar volumes maiores. • Distribuição e Logística. Pode usar informações antecipadas de remessa para programar e consolidar frete de entrada e saída • Gestão de Transportes. Auxilia a desenvolver planos ideais de consolidação de carga e cronogramas de roteamento • Planejamento de Estoque. Permite visualizar o nível de estoque atual e necessário para garantir um determinado grau de serviço

2.2 CONTA/TRANSAÇÃO

Como descrito em (WAZLAWICK, 2015), o padrão conta/transação é fortemente relacionado à área de negócios, mas tem variadas aplicações em múltiplas áreas. O exemplo do livro se refere a uma empresa que vende livros, e são citadas certas movimentações que são feitas acima dos produtos, estes livros podem ser catalogados, pedidos, recebidos, enviados, descartados etc. Estas movimentações, tanto quanto as transações financeiras associadas, dão origem a

conceitos como Pedido, Envio, Descarte, Devolução etc., cada um com determinados atributos e associações.

O mesmo padrão pode ser aplicado a domínios distintos. Em um sistema de controle de horas de trabalho, por exemplo, as contas representariam departamentos e colaboradores, enquanto as transações modelariam alocações de tempo. Na modelagem de um sistema logístico, por outro lado, as contas seriam centros de distribuição, e as transações, transferências de mercadorias entre eles. Esta aplicabilidade estrutural é o que fundamenta a proposta deste trabalho.

Um modelo de negócio que tem conceitos similares a estes, tem em seu núcleo, dois artefatos conceituais que podem ser explorados, um deles seria a *Conta*, que pode ser simplificada para algo que contem quantidades de algo (itens, produtos ou dinheiro por exemplo), esta quantidade é representada pelo *Saldo* da conta, que consiste da soma de todos os depósitos e saques de quantidades variadas da “coisa” representada pelo saldo.

A partir do momento que se identifica a existência de saques e depósitos pode se concluir que estes são apenas movimentações de uma conta para outra, portanto uma *Transação* consiste de duas movimentações: um depósito em uma conta e uma retirada do mesmo valor de outra conta.

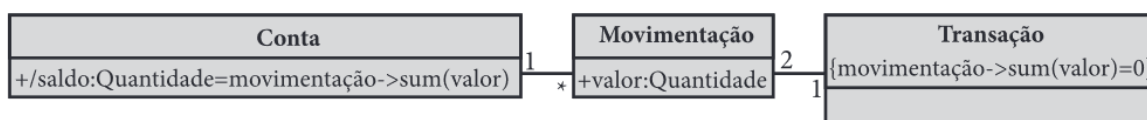


Figura 1 – Classes Conta/Transação

Para exemplificar as capacidades do padrão (WAZLAWICK, 2015) descreve a modelagem de uma empresa hipotética da seguinte maneira:

- Inicialmente são criadas cinco contas, que representam fornecedores, pedidos pendentes, produtos em estoque, produtos vendidos e produtos enviados, todas elas com saldo zerado

<u>editora:Conta</u>
saldo = 0

<u>pedidosPendentes:Conta</u>
saldo = 0

<u>estoque:Conta</u>
saldo = 0

<u>vendidos:Conta</u>
saldo = 0

<u>enviados:Conta</u>
saldo = 0

Figura 2 – Estado Inicial Exemplo

- Então, uma transação que modela um pedido de compra é criada, esta transação movimenta 50 itens da editora para os pedidos pendentes

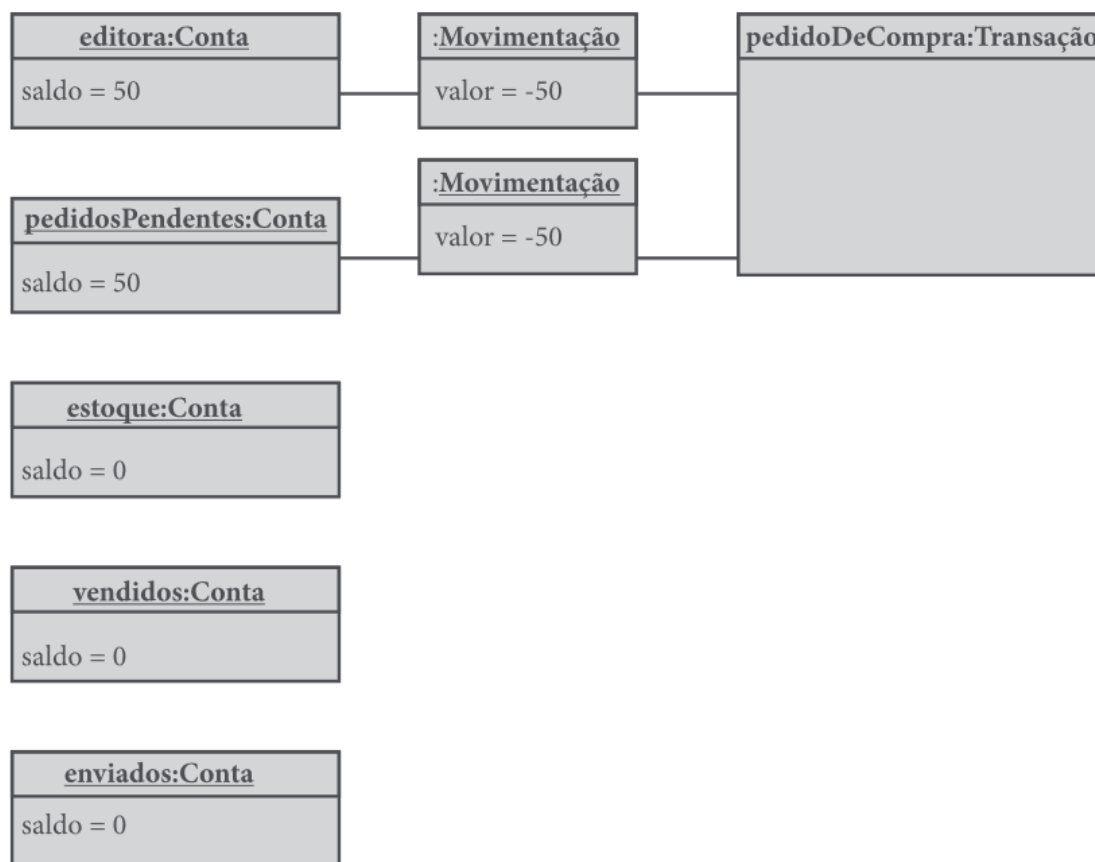


Figura 3 – Pedido de Compra Criado Exemplo

- Outras transações são adicionadas para modelar uma sequência de eventos, no caso, 40 itens sendo entregues, 25 destes sendo vendidos e então enviados para o cliente

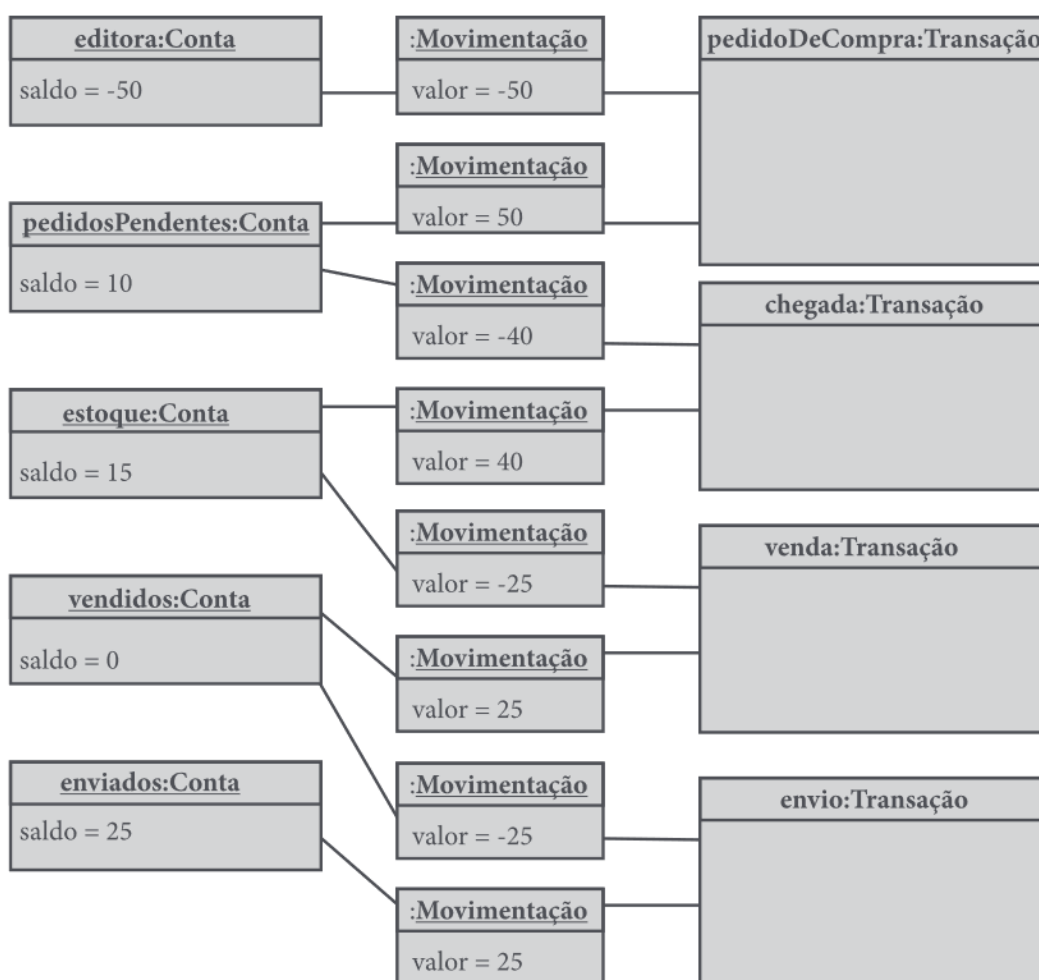


Figura 4 – Modelagem Completa Exemplo

Uma variação de movimentação dentro deste padrão que se mostra relevante e deve ser levada em consideração neste trabalho são as *movimentações mnemônicas* que como descritas em (WAZLAWICK, 2015) são movimentações que não necessariamente modelam transações entre duas contas, mas sim guardam alguma informação relevante, o exemplo canônico sendo uma transação monetária entre duas contas tendo que guardar uma parcela desta para modelar uma tributação hipotética que será paga em um momento futuro.

Estas movimentações mnemônicas se diferenciam de movimentações normais pelo fato de que uma transação necessariamente pode ter no máximo 2

movimentações normais (uma da conta fonte e a outra para a conta destino) porém esta mesma transação pode ter uma quantidade grande de movimentações mnemônicas por se tratar de uma “anotação” e não uma operação que de fato altera saldos de contas.

2.3 CONCLUSÃO DO CAPÍTULO

Estabelecidos os fundamentos teóricos de Business Intelligence e do padrão Conta/Transação, cabe agora articular explicitamente como esses três elementos se relacionam na proposta deste trabalho.

A principal dificuldade enfrentada por sistemas de BI reside na heterogeneidade dos dados gerados por processos empresariais distintos. Cada área da empresa, seja ela financeira, logística, recursos humanos ou de produção, gera dados em formatos e estruturas diferentes, exigindo processos de integração complexos antes que qualquer análise seja possível. O padrão Conta/Transação atua como um elemento normalizador ao nível conceitual: ao forçar que qualquer processo seja descrito como transferências de quantidades entre contas, ele impõe uma estrutura única independentemente do domínio. O motor de análise do sistema de BI, ao operar sobre essa estrutura uniforme, não precisa ser adaptado a cada novo processo, ele sempre encontrará contas com saldos e transações com frequências de execução.

Essa uniformidade estrutural tem uma consequência direta para o BI, as métricas necessárias para identificar gargalos operacionais são intrínsecas ao padrão. O saldo crescente de uma conta indica acúmulo de recursos naquela etapa do processo; a alta taxa de bloqueio de uma transação indica que ela não consegue executar por insuficiência de recursos na conta de origem. Esses indicadores

emergem naturalmente de qualquer modelo Conta/Transação, sem que o analista precise defini-los manualmente para cada domínio.

3 TRABALHOS RELACIONADOS

3.1 PRODUÇÕES ACADÊMICAS

Neste capítulo são apresentados trabalhos correlatos cujo tema e relacionado com este, para simultaneamente auxiliar no desenvolvimento, auxiliando no desvio de possíveis erros evitáveis, tanto como justificar a existência deste por não abordar o tema da mesma maneira. Também serão incluídos softwares que estão atualmente no mercado para levantar o estado da arte e justificar a presente proposta de trabalho.

3.1.1 Business Intelligence: the implementation of a BI solution to a case study (VALENTI, 2022)

O trabalho de Valenti, desenvolvido no Politecnico di Torino, implementa uma solução de BI para um estudo de caso empresarial específico. Sua contribuição mais relevante para o este trabalho reside no detalhamento da arquitetura clássica de sistemas de BI: o modelo composto por fontes de dados heterogêneas, processos ETL (Extract, Transform, Load), Data Warehouse e ferramentas de visualização por meio de dashboards interativos. Essa descrição permite compreender como os sistemas de BI tradicionais estruturam o fluxo de dados desde sua origem até a geração de relatórios para tomada de decisão.

Entretanto, essa mesma arquitetura evidencia uma limitação central: o processo ETL pressupõe que os dados dos processos de negócio já estão disponíveis em fontes estruturadas como bancos de dados operacionais ou planilhas. A etapa de transformação do ETL tem por objetivo padronizar dados que já existem, mas que foram gerados por sistemas heterogêneos. Isso significa que o sistema de BI é, reativo: ele analisa dados produzidos por processos que já

ocorreram, sem participar do processo de concepção destes processos.

Este trabalho se diferencia, antes de fazer qualquer análise, se modela diretamente no software utilizando o padrão Conta/Transação, e a própria modelagem gera os dados que alimentarão a simulação e a análise de gargalos, em vez de depender de fontes de dados preexistentes. Essa diferença arquitetural posiciona o protótipo proposto não como um substituto dos sistemas de BI tradicionais, mas como um complemento voltado a cenários em que o processo ainda não está suficientemente estruturado para alimentar um pipeline ETL convencional.

3.1.2 Implementation of a business intelligence solution: a case study of a workforce and staffing solutions company (FERREIRA, 2022)

O trabalho de Ferreira (2022) descreve o desenvolvimento de uma solução de BI para uma empresa do setor de recursos humanos, motivado pela ausência de um mecanismo ágil de acesso a informações gerenciais para o Chief Executive Officer (CEO) da organização. A solução implementada organiza as informações em seis categorias de dashboards: visão geral da empresa, situação financeira, análise de clientes, análise de colaboradores, KPIs e previsões. Cada categoria agrega métricas como número de novos clientes, taxa de retenção de colaboradores e receita por cliente, e as apresenta de forma consolidada para apoio à tomada de decisão.

A principal contribuição analítica deste trabalho, para este TCC, é demonstrar quais tipos de métricas são relevantes para a gestão de um processo empresarial do ponto de vista do BI. Métricas como número de colaboradores ativos, número de reuniões e taxa de novos colaboradores por mês têm correspondência direta com o

padrão Conta/Transação, por exemplo, colaboradores ativos podem ser modelados como o saldo de uma conta, e contratações ou desligamentos como transações que alteram esse saldo ao longo do tempo. Essa correspondência sugere que processos de RH são modeláveis pela estrutura proposta neste trabalho, o que amplia o escopo de aplicação do protótipo.

Entretanto, a abordagem de Ferreira (2022) parte do pressuposto de que os dados já estão disponíveis nos sistemas internos da empresa e precisam apenas ser extraídos, agregados e visualizados, similar ao trabalho de Valenti introduzido anteriormente. A solução não propõe nenhum mecanismo de padronização conceitual dos processos antes da coleta dos dados. Isso limita sua aplicabilidade a quaisquer empresas que já possuam uma estrutura de processos definida, uma condição que muitas organizações de pequeno e médio porte não satisfazem. O presente trabalho busca preencher essa lacuna ao propor que a modelagem transacional dos processos precede e orienta a coleta de dados, tornando o sistema aplicável em contextos onde os processos ainda não foram formalmente ou conceitualmente estruturados.

3.1.3 Expensetracker – Personal Finance Manager (AZIZOV, 2021)

O trabalho de Azizov (2021) propõe um gerenciador de finanças pessoais cujo modelo de domínio apresenta semelhança estrutural relevante com o padrão Conta/Transação. Conforme ilustrado na Figura 5, o modelo define as entidades centrais Account e Transaction, mantendo entre elas uma relação que espelha o núcleo do padrão de Wazlawick (2015): contas contêm saldos e transações movimentam valores entre elas.

A análise do diagrama de classes revela, entretanto, diferenças significativas decorrentes da especialização do domínio financeiro. O modelo de Azizov introduz entidades como Budget, BudgetHistory, Category e Currency, além de uma enumeração de tipos de conta (Cash, CreditCard, CurrentAccount, Savings). Cada uma dessas adições resolve um problema específico do domínio financeiro pessoal: o orçamento (Budget) impõe restrições ao gasto; a categoria (Category) classifica as transações para análise posterior; a moeda (Currency) permite operações multimoeda. Esses elementos são necessários no contexto financeiro, mas introduzem dependências de domínio que reduzem a generalidade do modelo.

É exatamente nesse ponto que reside a diferença arquitetural central em relação a este trabalho. O padrão Conta/Transação de Wazlawick (2015) opera em um nível de abstração mais alto: contas representam qualquer entidade que contenha um recurso quantificável, e transações representam qualquer movimentação desse recurso entre contas, independentemente do domínio. Ao abrir mão de especializações como Budget e Category, o padrão ganha em aplicabilidade o que perde em especificidades do domínio. O protótipo desenvolvido neste trabalho herda essa filosofia: as variáveis de cada conta podem ser configuradas pelo usuário, permitindo que o mesmo mecanismo modele pães num forno, tickets num sistema de suporte ou horas alocadas ao desenvolvimento de um projeto.

A contribuição analítica do trabalho de Azizov (2021) para o presente TCC está, portanto, não no que os dois modelos compartilham, mas no que eles diferem, demonstrando como a especialização de domínio, ao ser introduzida no nível estrutural do modelo, limita sua reutilização em outros contextos, argumento que reforça a escolha do padrão Conta/Transação como base de um sistema de BI de propósito mais geral.

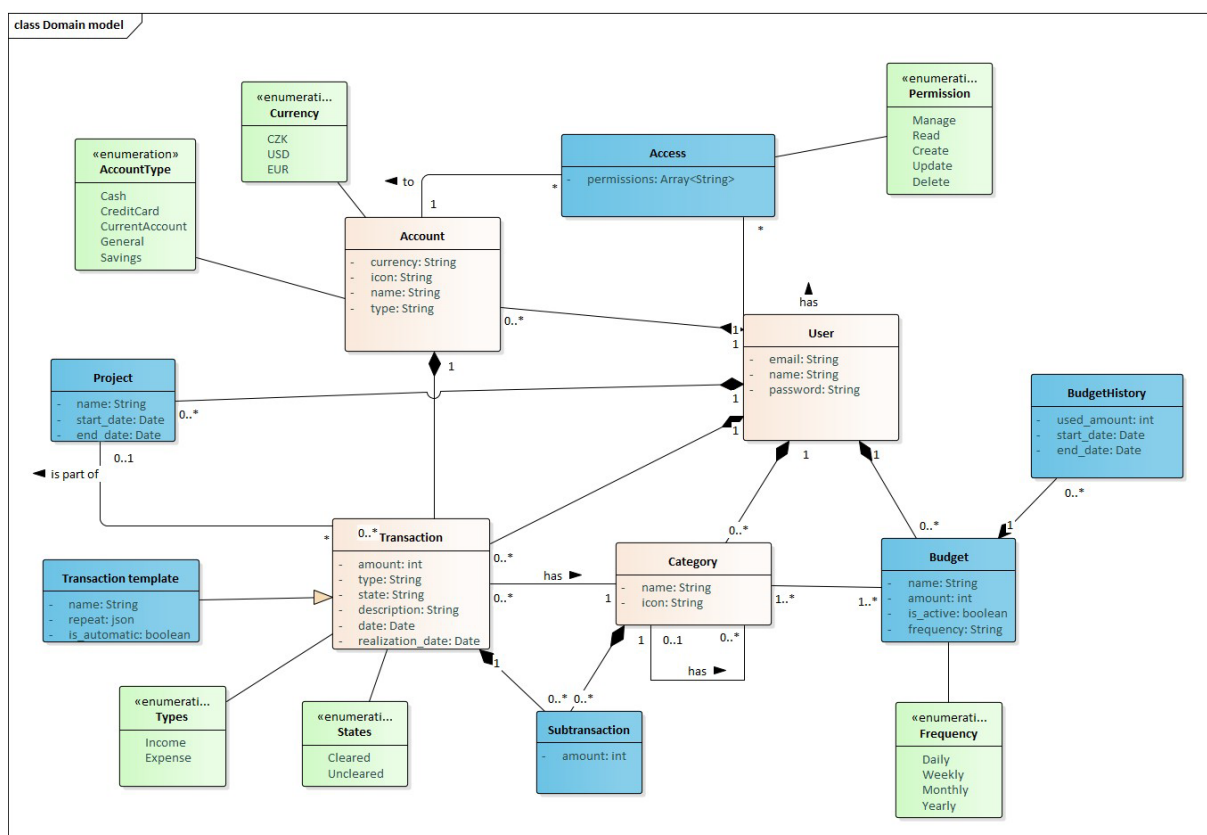


Figura 5 – Modelagem (JAMALADDIN AZIZOV, 2021)

3.2 Softwares no Mercado

Para posicionar a proposta deste trabalho em relação às ferramentas comerciais disponíveis, foram analisados dois dos principais softwares de BI do mercado: IBM Cognos Analytics e Microsoft Power BI. A análise foi conduzida com base em três critérios relevantes para o escopo deste TCC:

- 1 – a capacidade de modelar processos de negócio internamente, sem depender de fontes de dados preexistentes;
- 2 – o suporte a simulação de comportamento dinâmico ao longo do tempo;
- 3 – a capacidade de analisar processos de diferentes áreas.

3.2.1 IBM Cognos Analytics

O IBM Cognos Analytics é uma plataforma consolidada de BI voltada à exploração de dados, criação de dashboards interativos e geração de relatórios corporativos (IBM, 2024). Suas principais funcionalidades incluem modelagem semântica de dados, conectividade com múltiplas fontes, análise preditiva por séries temporais e recursos de inteligência artificial para sugestão automática de visualizações.

Em relação ao critério 2, o Cognos possui “elementos de simulação” por oferecer previsão estatística e árvores de decisão. É necessário, entretanto, distinguir esses dois mecanismos, previsão estatística (forecasting) consiste em extrapolar tendências de dados históricos para estimar valores futuros, enquanto simulação de eventos discretos modela o comportamento dinâmico de um sistema ao longo do tempo, com eventos que alteram o estado da simulação conforme regras definidas pelo usuário. O Cognos realiza o primeiro, mas não o segundo. Em relação ao critério (1), a plataforma pressupõe que os dados dos processos já existem em fontes estruturadas, não há mecanismo nativo para que o usuário modele um processo antes de coletá-lo.

3.2.2 Microsoft Power BI

O Microsoft Power BI é atualmente a ferramenta de BI mais amplamente adotada em ambientes corporativos, com destaque para sua interoperabilidade com o ecossistema Microsoft (Excel, SharePoint, Azure) e sua curva de aprendizagem relativamente acessível para criação de dashboards (MICROSOFT, 2024). O produto oferece modelagem de dados via DAX (Data Analysis Expressions), conectividade com centenas de fontes de dados e recursos de análise colaborativa em nuvem.

Um aspecto que merece atenção especial é a funcionalidade Process Mining, integrada ao Microsoft Power Automate desde 2022. Esse recurso permite que a plataforma importe logs de eventos de sistemas corporativos e gere automaticamente mapas de processos, identificando variações, gargalos e desvios de fluxo. A funcionalidade comercial mais próxima do objetivo deste TCC.

Entretanto, o Process Mining do Power Automate opera sobre dados históricos já registrados nos sistemas da empresa, em organizações que ainda não possuem sistemas capazes de gerar esses logs estruturados, a ferramenta é dificilmente aplicável.

3.2.3 Comparação entre os Softwares do Mercado e o Protótipo Proposto

Nenhuma das ferramentas analisadas satisfaz simultaneamente os três critérios estabelecidos. Ambas operam sobre dados preexistentes, não suportam simulação de eventos discretos baseada em modelos definidos pelo usuário, e não oferecem um mecanismo de modelagem conceitual independente de domínio. O protótipo proposto neste trabalho ocupa uma lacuna específica, a da capacidade de modelar um processo de negócio antes da existência de dados históricos, simulá-lo, e identificar possíveis gargalos com base em um padrão conceitual genérico. Nesse sentido, o software desenvolvido é complementar às ferramentas de mercado, podendo até servir de inspiração para futuros plugins ou funcionalidades adicionais.

3.3 Conclusão do Capítulo

Tabela 2. Comparação entre os trabalhos relacionados e a proposta deste trabalho

Trabalho	Domínio	Independência de dados pré-existent	Simulação comportamental dinâmica	Modelagem conceitual independente de domínio	Foco em detecção de gargalos
Valenti (2022)	Estudo de caso empresarial	Não	Não	Não	Não
Ferreira (2022)	Recursos Humanos	Não	Não	Não	Não
Azizov (2021)	Finanças pessoais	Sim	Não	Não	Não
IBM Cognos Analytics	BI corporativo genérico	Não	Não *	Não	Não
Microsoft Power BI (Process Mining)	BI corporativo + mineração de processos	Não	Não **	Não	Sim
Protótipo proposto (este trabalho)	Genérico por design	Sim	Sim	Sim	Sim

*Cognos oferece previsão estatística (forecasting), não simulação de eventos discretos.

****Power BI Process Mining analisa logs históricos já registrados; não simula cenários hipotéticos.**

A análise dos trabalhos relacionados permite posicionar com maior precisão a contribuição do presente trabalho. Os estudos acadêmicos de Valenti (2022) e Ferreira (2022) demonstram como sistemas de BI são implementados na prática, mas compartilham uma premissa comum: a existência prévia de dados estruturados nos sistemas da organização. O trabalho de Azizov (2021), por sua vez, ilustra como o padrão transacional pode ser especializado para um domínio específico e evidencia por contraste a importância da abstração proposta por Wazlawick (2015).

As ferramentas comerciais, o IBM Cognos Analytics e o Microsoft Power BI oferecem soluções maduras para análise e visualização de dados históricos. O recurso de Process Mining do Power Automate é o mecanismo comercial mais próximo do objetivo deste TCC, mas igualmente dependente de logs de eventos preexistentes. Nenhuma das ferramentas analisadas suporta a modelagem conceitual de processos por meio de um padrão transacional genérico, nem a simulação preditiva baseada nessa modelagem.

Essa lacuna justifica a proposta deste trabalho e define seu escopo de contribuição. O protótipo desenvolvido não pretende substituir as plataformas de BI existentes, mas oferecer um mecanismo complementar voltado a contextos em que o processo ainda não está suficientemente estruturado para alimentar as ferramentas tradicionais.

4 DESENVOLVIMENTO

4.1 PROVA DE CONCEITO

Além do estudo teórico feito até então, da parte prática do trabalho, foram feitos diagramas UML para modelar estruturalmente e comportamentalmente o software a ser finalizado, foram identificados 3 componentes principais do software, a modelagem, a simulação, e a análise, com seus correspondentes casos de uso.

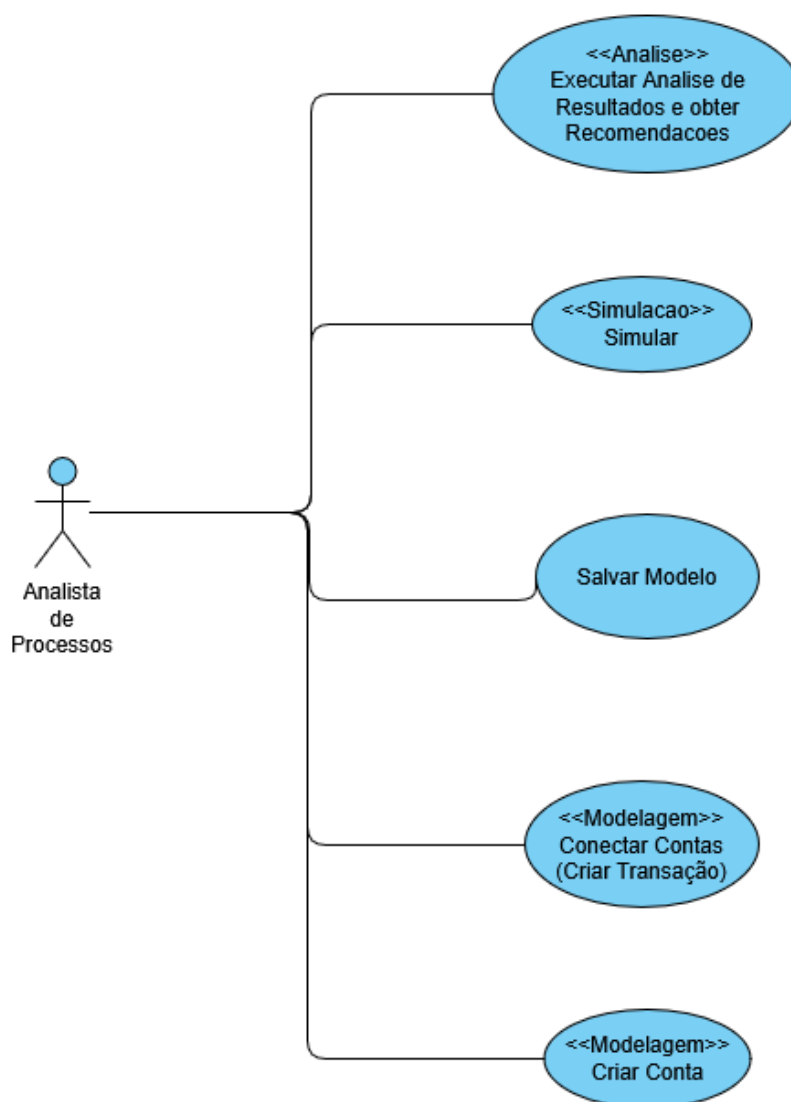


Figura 6 – Casos de Uso Software

Também para garantir a adesão ao padrão conta/transação, a modelagem do programa, desde início, foi deliberadamente feita para implementar o funcionamento transacional, tal qual um sistema financeiro.

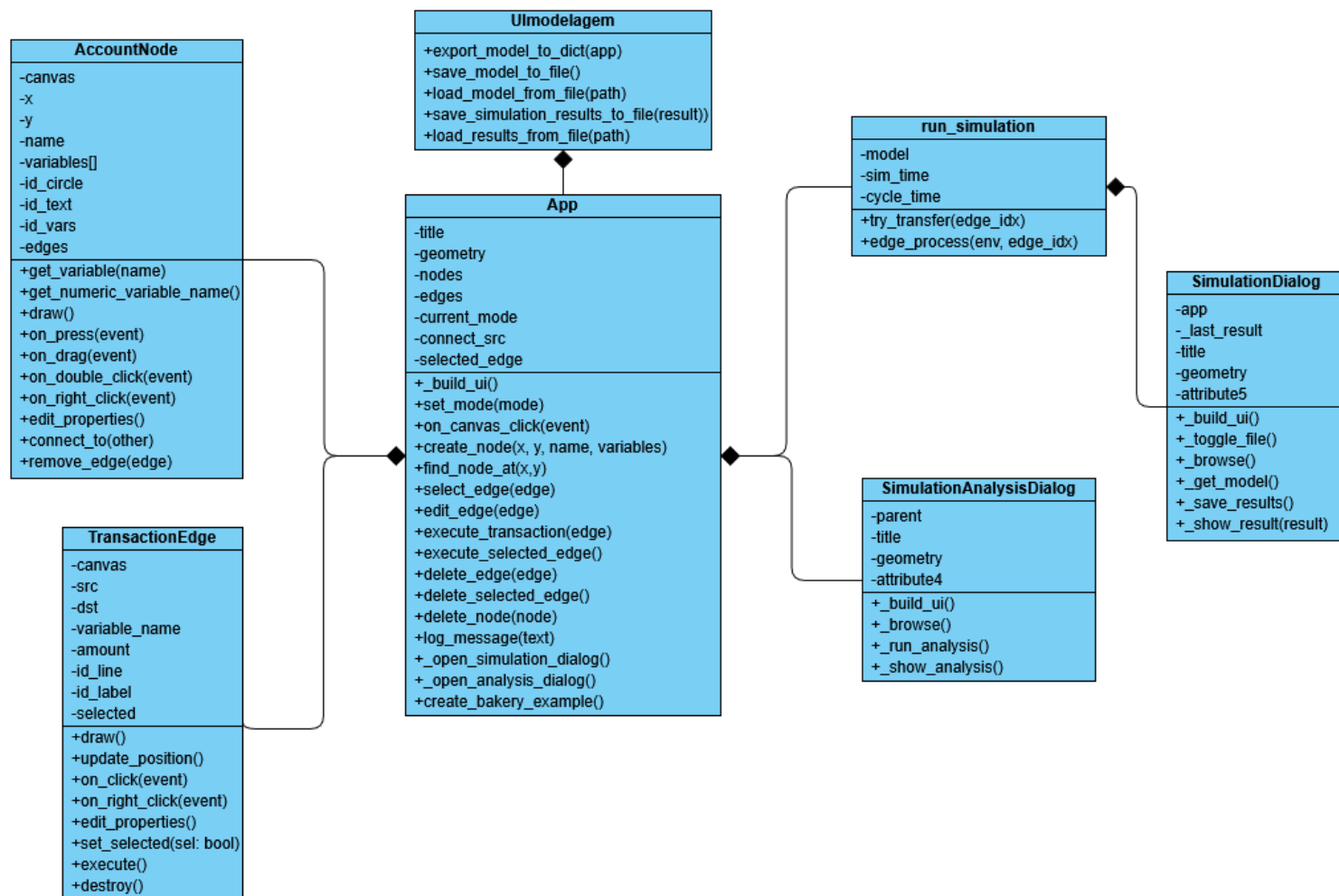


Figura 7 – Diagrama de Classes Software

4.2 IMPLEMENTACAO

O código da implementação se encontra no [repositorio do github](#)

As tecnologias e bibliotecas utilizadas incluem o simpy, tkinter, json e threading.

Nas seções seguintes serão apresentadas as regras de negócio e o uso do software para realizá-las

4.2.1 Criando Contas

Uma conta representa a principal unidade de modelagem do sistema. Conforme definido pelo padrão Conta/Transação, uma conta corresponde a uma entidade que contém determinado recurso, cuja quantidade é representada por um saldo. Dependendo do domínio modelado, esse recurso pode representar produtos em estoque, horas de trabalho disponíveis, pedidos pendentes, tickets de suporte, ou qualquer outro elemento passível de movimentação entre etapas de um processo.

Na implementação desenvolvida, uma conta é representada visualmente por um nodo no ambiente de modelagem, similar a um estado de uma máquina de estados finita. Em sua forma mais simples, ela possui apenas um nome e um saldo. Entretanto, para permitir a representação de cenários mais complexos, uma conta pode armazenar múltiplas variáveis, cada uma com seu próprio tipo e conjunto de restrições.

As variáveis atualmente suportadas incluem números inteiros, números de ponto flutuante e valores booleanos. Essa flexibilidade permite representar tanto recursos quantitativos quanto estados lógicos relevantes para determinados processos de negócio onde podem ocorrer gargalos.

Durante a construção de um modelo, o Analista de Processos deve inicialmente identificar quais elementos do processo podem ser representados como contas. Após ativar o modo de criação por meio do botão 'Criar Conta', novos nodos podem ser inseridos na área de modelagem utilizando o botão esquerdo do mouse.



Figura 8 – Representacao de Conta

4.2.2 Conectando Contas (Criando Transações)

Embora as contas representem os estados do processo modelado, são as transações que definem a dinâmica do sistema. Uma transação representa a movimentação de um recurso entre duas contas, caracterizando uma mudança de estado dentro do processo analisado.

Na interface do software, as transações são representadas visualmente por setas direcionadas. O ponto de origem corresponde à conta da qual o recurso será removido, enquanto o ponto de destino identifica a conta que receberá esse recurso.

Além das contas de origem e destino, cada transação possui informações adicionais responsáveis por definir qual variável será movimentada e qual quantidade será transferida a cada execução.

A criação de transações está sujeita a restrições de consistência. Como uma movimentação pressupõe compatibilidade entre os recursos envolvidos, apenas contas que possuam variáveis compatíveis podem ser conectadas.

Para criar uma transação, o Analista deve ativar o modo de conexão através do botão 'Modo Conectar'. Em seguida, seleciona-se a conta de origem e posteriormente a conta de destino, ambas utilizando o botão esquerdo do mouse.



Figura 9 – Transação conectando Contas

4.2.3 Selecionando Contas

Após a criação das contas, é necessário definir suas características específicas. Para atender essa necessidade, o software disponibiliza um editor de propriedades responsável pela configuração e manutenção dos elementos que compõem cada conta, por meio dessa funcionalidade, o Analista de Processos pode alterar o nome da conta, modificar variáveis existentes ou criar variáveis necessárias para representar aspectos particulares adicionais do processo modelado.

Além da definição do nome, cada variável possui propriedades próprias, incluindo seu tipo de dado, valor inicial e restrições de domínio, como a permissão ou proibição de valores negativos.

A possibilidade de personalizar variáveis permite que o mesmo mecanismo de modelagem seja utilizado em diferentes cenários empresariais, reforçando a proposta de generalização do padrão Conta/Transação.

O acesso ao editor é realizado através do modo de seleção. Após ativar o botão 'Modo Seleção', o usuário pode clicar com o botão direito sobre uma conta e selecionar a opção 'Editar' para abrir a interface de configuração.

Editar Conta

Nome da conta: Conta

Variáveis:

Nome	Tipo	Valor	Pode ser negativo	Ação
Saldo	float	0.00	☒	Remover
Var2	int	0	☐	Remover
Var3	bool	True	☐	Remover

+ Adicionar variável

OK Cancel

Figura 10 – Editor de Conta

4.2.4 Executando Transações

A execução de transações constitui um dos mecanismos centrais de funcionamento do protótipo. É por meio dela que o estado do modelo evolui ao longo do tempo, executando o comportamento do processo real que está sendo representado.

Quando uma transação é executada, o sistema realiza a movimentação da quantidade especificada da variável associada, removendo esse valor da conta de origem e adicionando-o à conta de destino. Dessa forma, a execução das transações altera continuamente os saldos das contas e produz o fluxo de recursos que caracteriza o processo modelado.

Do ponto de vista conceitual, a execução de uma transação corresponde à ocorrência de um evento dentro do processo de negócio. Em um ambiente industrial, por exemplo, ela pode representar a transferência de matéria-prima entre setores. Em um ambiente de suporte técnico, pode representar a solução, transferência ou inatividade de um ticket.

Uma particularidade importante da implementação é o suporte a transações cuja origem e destino são a mesma conta. Embora aparentemente redundantes, essas transações possibilitam representar mecanismos de geração ou consumo contínuo de recursos.

Esse recurso se mostra especialmente útil na modelagem de cenários inspirados no problema produtor-consumidor, nos quais determinadas contas atuam como fontes geradoras de recursos enquanto outras representam elementos consumidores desses mesmos recursos.

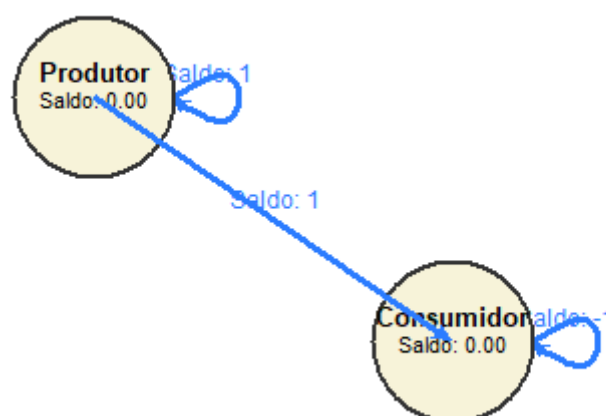


Figura 11 – Exemplo Produtor - Consumidor

4.2.5 Persistencia do Modelo

Uma característica importante do software é a capacidade de persistir os modelos criados pelo usuário. Essa funcionalidade permite interromper e retomar sessões de modelagem sem a necessidade de reconstruir manualmente toda a estrutura do processo.

Para implementar esse mecanismo, o sistema percorre todas as contas e transações presentes no modelo e converte suas informações para o formato JSON. A escolha desse formato deve-se à sua simplicidade, ampla compatibilidade e facilidade de integração com outras ferramentas.

Além de preservar a modelagem realizada pelo usuário, o arquivo gerado também funciona como uma representação intermediária utilizada pelos demais módulos do sistema. Dessa forma, tanto a simulação quanto futuras funcionalidades podem operar diretamente sobre o modelo serializado.

A funcionalidade é disponibilizada através dos botões ‘Salvar Modelo’ e ‘Carregar Modelo’, responsáveis respectivamente pela exportação e importação dos dados da modelagem.

4.2.6 Simulacao do Modelo (Detecção de Gargalos)

Uma das funcionalidades centrais do protótipo é a capacidade de simular o comportamento de um processo modelado utilizando o padrão Conta/Transação. Essa funcionalidade permite observar a evolução dinâmica do sistema ao longo do tempo e identificar possíveis gargalos operacionais antes que estes ocorram em um ambiente real.

Para realizar essa simulação, o software utiliza a biblioteca SimPy, um framework de simulação de eventos discretos desenvolvido para a linguagem

Python. Nessa abordagem, as transações definidas pelo usuário são tratadas como eventos que tentam executar periodicamente, movimentando recursos entre contas de acordo com as regras estabelecidas na modelagem.

Durante a execução da simulação, cada transação verifica continuamente se existem condições suficientes para sua realização. Caso os recursos necessários estejam disponíveis na conta de origem, a movimentação é efetuada e os saldos das contas envolvidas são atualizados. Caso contrário, a transação permanece temporariamente bloqueada até que os recursos necessários estejam disponíveis novamente.

Essa dinâmica permite reproduzir comportamentos observados em processos reais. Quando uma determinada etapa do processo recebe recursos em uma velocidade superior à sua capacidade de processamento ou transferência, ocorre um acúmulo progressivo de saldo em uma ou mais contas. Esse comportamento caracteriza a existência de um possível gargalo operacional.

No contexto do modelo proposto, a identificação de gargalos é realizada através da observação do comportamento das contas e transações ao longo da simulação. Contas que apresentam crescimento contínuo de saldo podem indicar etapas congestionadas do processo, enquanto transações que executam com baixa frequência podem indicar restrições operacionais ou insuficiência de recursos. Essas informações são registradas durante a execução da simulação para processamento posterior pelo módulo de análise.

Uma vantagem da utilização do padrão Conta/Transação é que a detecção de gargalos independe do domínio específico do processo modelado. Como todos os recursos são representados por contas e todas as operações são representadas

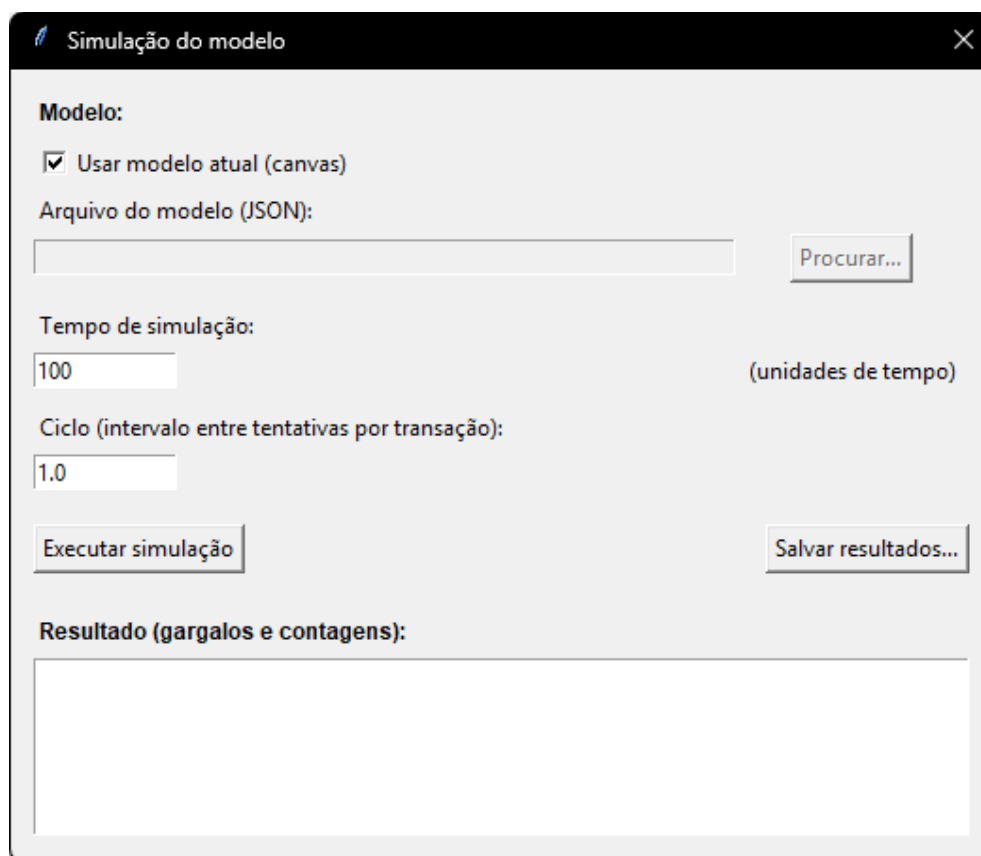
por transações, o mecanismo de simulação pode utilizar a mesma lógica para analisar processos produtivos, logísticos, administrativos ou financeiros.

Ao término da execução, o sistema gera um conjunto de métricas que descrevem o comportamento do modelo durante o período simulado. Essas informações incluem dados como a quantidade de Os resultados podem então ser exportados para um arquivo JSON, servindo como entrada para o módulo de análise apresentado na seção seguinte.

Para executar uma simulação, o usuário deve possuir um modelo carregado no canvas ou um arquivo de modelagem previamente salvo. A interface de transacoes saidas de determinada conta, e quantas foram bloqueadas por falta de saldo, por exemplo. A simulação pode ser acessada através do botão “Simulação”, onde são definidos parâmetros como o tempo total de simulação e o intervalo entre tentativas de execução das transações.

Após o início da execução, os eventos simulados são processados pelo SimPy e os resultados são apresentados na parte inferior da interface. Ao término do processo, o usuário pode salvar os resultados gerados por meio do botão “Salvar Resultados”, produzindo um arquivo JSON que poderá ser posteriormente utilizado pelo módulo de análise de gargalos.

Diferentemente de ferramentas tradicionais de Business Intelligence, que normalmente analisam apenas dados históricos já produzidos pela organização, a abordagem proposta permite simular cenários hipotéticos antes de sua implementação. Dessa forma, gestores podem avaliar o impacto de alterações em processos, capacidades produtivas ou fluxos de recursos sem a necessidade de modificar inicialmente o ambiente real.



Simulação do modelo

Modelo:

☒ Usar modelo atual (canvas)

Arquivo do modelo (JSON):

Procurar...

Tempo de simulação:

(unidades de tempo)

Ciclo (intervalo entre tentativas por transação):

Executar simulação Salvar resultados...

Resultado (gargalos e contagens):

Figura 12 – UI de Simulação

4.2.7 Análise da Simulação

A etapa de análise da simulação representa o componente de Business Intelligence do protótipo. Após a execução da simulação, o sistema utiliza os dados produzidos para identificar padrões que possam indicar a existência de gargalos operacionais, acúmulo excessivo de recursos em determinadas contas ou desequilíbrios entre produção e consumo.

O processo de análise é realizado sobre um arquivo de resultados previamente gerado pelo simulador, com base nessas informações, o sistema gera recomendações destinadas ao Analista de Processos. Essas recomendações não alteram automaticamente a modelagem, mas servem como suporte à tomada de

decisão, permitindo que o usuário refine o modelo e execute novas simulações até obter um comportamento considerado satisfatório.

O acesso à interface de análise é realizado por meio do botão 'Análise da Simulação'. Nessa interface, o usuário seleciona um arquivo de resultados previamente salvo e aciona o comando 'Analisar Resultados'. Após o processamento das informações, as recomendações identificadas são apresentadas na tela para avaliação do analista.

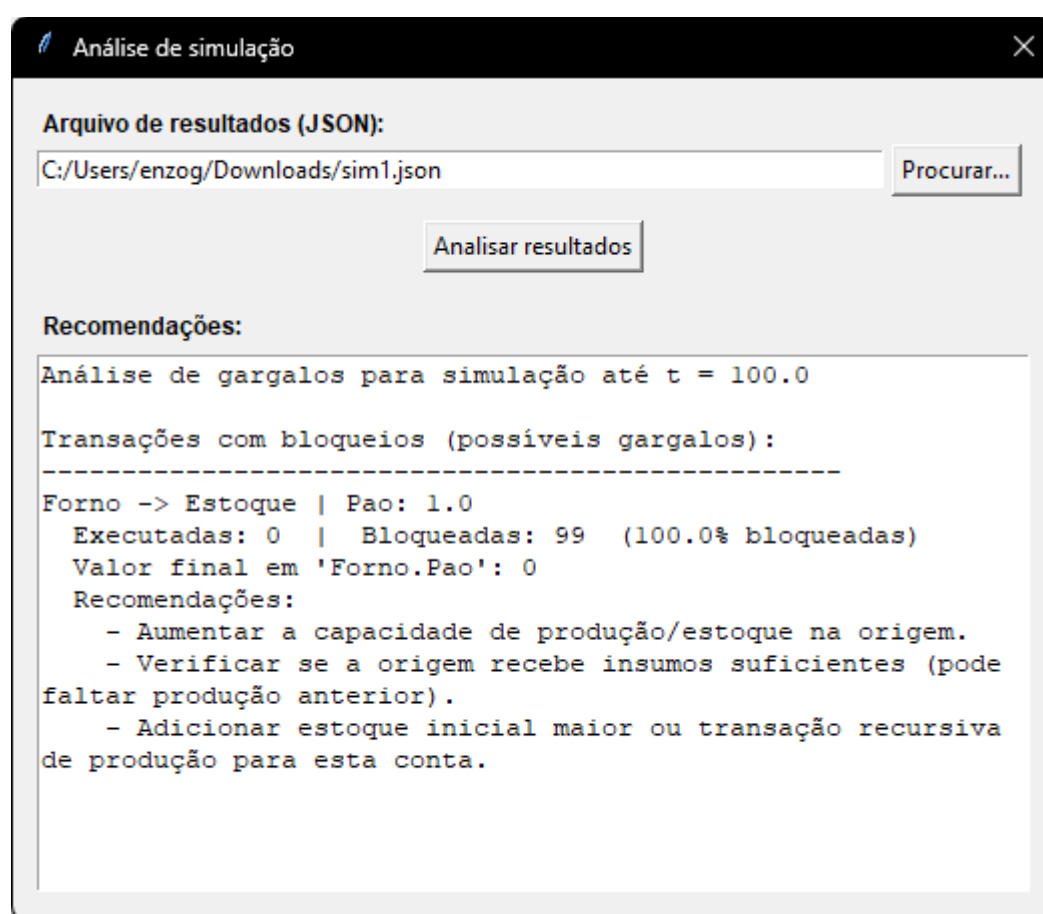


Figura 13 – UI Análise de Simulação

Como já introduzido, um gargalo é aquele recurso cuja capacidade é igual ou menor que a demanda colocada nele (GOLDRATT; COX, 1984). Na análise realizada na imagem, do exemplo da padaria que será descrito posteriormente, temos um gargalo evidente no elemento “Forno”. Ele possui a menor capacidade do

sistema, e todos os pães produzidos precisam obrigatoriamente passar por ele antes de irem ao balcão, onde serão retirados do sistema. Portanto, neste exemplo, conforme detectado pelo software, o forno é um gargalo a ser corrigido. Isso poderia ser feito, por exemplo, ao aumentar a capacidade de produção, como sugerido, o que se traduziria na prática como “adicionar mais fornos” para paralelizar o processo e não depender de apenas uma unidade.

4.2.7.1 Heurísticas de Avaliação

Tabela 3 - Métricas utilizadas e recomendações sugeridas pelo software

Métrica observada	Interpretação	Recomendação sugerida
Bloqueios > Execuções	Produção insuficiente ou restrição de capacidade	Aumentar capacidade de produção ou estoque da origem
Valor transferido elevado	Fluxo excessivamente concentrado	Reduzir quantidade transferida ou aumentar frequência das operações
Execuções = 0	Ausência de recursos disponíveis na origem	Verificar etapas produtoras anteriores
Saldo inicial/final igual a zero durante toda a simulação	Processo sem recursos para iniciar	Aumentar estoque inicial ou criar produção recursiva
Nenhuma condição anterior	Possível competição por recursos compartilhados	Avaliar priorização entre transações

Esta abordagem não utiliza técnicas de aprendizado de máquina ou métodos estatísticos para identificação de gargalos. As recomendações são produzidas por meio de heurísticas determinísticas baseadas em regras, construída a partir da interpretação das métricas coletadas durante a simulação. Embora essa estratégia tenha se mostrado adequada para a prova de conceito desenvolvida neste trabalho, processos empresariais mais complexos poderão exigir métricas adicionais e algoritmos de análise mais sofisticados em trabalhos futuros.

4.2.8 Exemplo Padaria

Para demonstrar a utilização do protótipo, foi desenvolvida uma modelagem simplificada de uma padaria. O exemplo busca representar um processo produtivo relativamente simples, mas suficiente para ilustrar a identificação de gargalos utilizando o padrão Conta/Transação.

Nesse modelo, o Padeiro atua como produtor de pães, gerando continuamente unidades do produto. Os pães produzidos são encaminhados ao Forno, responsável pelo processamento necessário antes da disponibilização para venda. Após essa etapa, os produtos são transferidos para o Estoque, que atua como um buffer intermediário, e finalmente chegam ao Balcão, onde ocorre o consumo da produção.

A presença do Forno e do Estoque permite reproduzir situações típicas de gargalos operacionais. Caso a capacidade do Forno seja inferior à taxa de produção do Padeiro, por exemplo, ocorrerá o acúmulo progressivo de produtos na etapa anterior do processo. Da mesma forma, limitações nas movimentações envolvendo o Estoque podem provocar atrasos no abastecimento do Balcão. Durante a simulação, essas situações podem ser identificadas pelo aumento contínuo dos saldos de

determinadas contas ou pela baixa frequência de execução de transações específicas.

Embora simplificado, o cenário da padaria demonstra a capacidade do software de representar processos produtivos utilizando o padrão Conta/Transação e de utilizar essa representação como base para a identificação de gargalos. Dessa forma, o exemplo serve como uma validação inicial da proposta de utilizar uma modelagem transacional genérica como fundamento para análises de Business Intelligence.

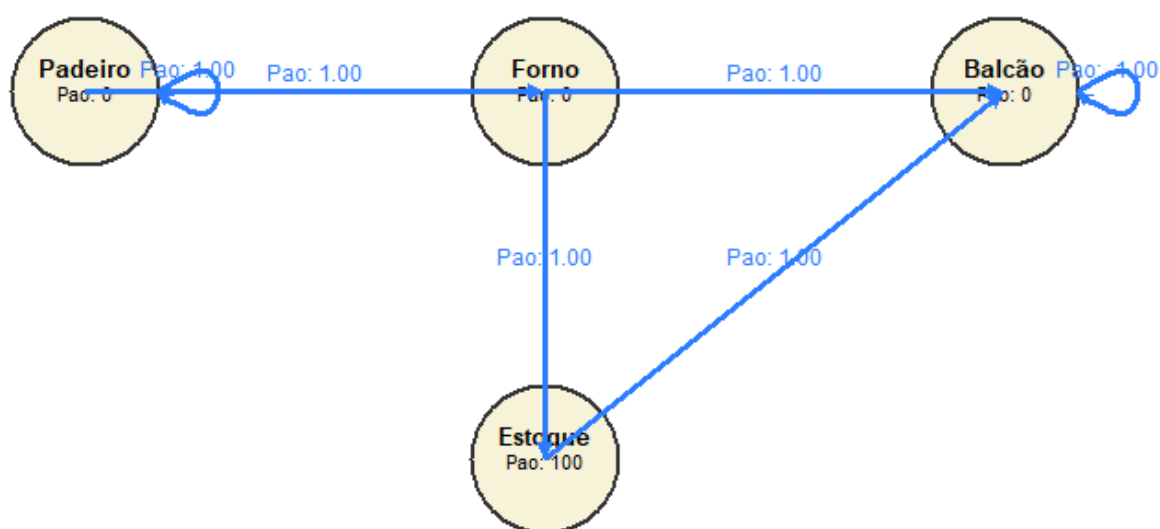


Figura 14 – Exemplo Padaria

4.3 DISCUSSÃO DOS RESULTADOS

Tabela 3 – Comparação entre Conta/Transação e o Modelo Relacional Tradicional

Conta/Transação	Modelo Relacional Tradicional
Contas representam estados	Diversas tabelas específicas
Transações representam fluxo	Chaves estrangeiras
Motor único	Regras por domínio
Mesmo algoritmo	Consultas específicas

No modelo desenvolvido neste trabalho, o processo da padaria é representado por quatro contas (Padeiro, Forno, Estoque e Balcão) conectadas por transações que movimentam recursos entre essas etapas. O estado do sistema é obtido diretamente pelos saldos das contas, enquanto o comportamento é determinado pela execução das transações durante a simulação.

Em uma modelagem relacional tradicional, por outro lado, o mesmo processo seria representado por entidades específicas do domínio, como Pao, Lote, Forno, Estoque e Venda, além de relacionamentos que descrevem a etapa atual de cada item. A determinação de quantos produtos estão em cada fase dependeria de várias consultas ao banco de dados, frequentemente envolvendo múltiplas tabelas e regras específicas daquele processo.

A principal vantagem do padrão Conta/Transação, no contexto deste trabalho, não é substituir a modelagem relacional para armazenamento de dados, mas fornecer uma representação conceitual uniforme sobre a qual o mesmo mecanismo

de simulação e análise pode operar. Dessa forma, o algoritmo responsável por detectar gargalos permanece inalterado mesmo quando o domínio modelado muda, bastando que novos processos sejam descritos em termos de contas e transações.

A implementação do protótipo, portanto, permitiu avaliar, de forma prática, a aplicação do padrão Conta/Transação como base para um sistema de Business Intelligence orientado à identificação de gargalos. Os cenários modelados demonstraram que processos com fluxo claro de recursos entre etapas podem ser representados de maneira consistente por meio de contas e transações, reduzindo a necessidade de estruturas específicas para cada domínio de negócio.

Observou-se que a separação entre modelagem, simulação e análise facilitou o processo iterativo de refinamento do modelo. O usuário pode construir o processo, executar a simulação, analisar os resultados e ajustar a modelagem sucessivamente, aproximando-se do ciclo típico de melhoria contínua encontrado em ambientes de gestão operacional.

4.3.1 Viabilidade da Modelagem Conta/Transação

O desenvolvimento indicou que o padrão Conta/Transação mostra-se adequado para representar fluxos de produção, armazenamento e consumo de recursos. Como ilustrado pelo exemplo da padaria com a produção, processamento, armazenamento e venda modelados utilizando apenas contas e movimentações entre elas.

Essa uniformidade estrutural simplifica a construção do motor de simulação, já que diferentes tipos de processo compartilham a mesma representação básica. Em vez de criar estruturas específicas para cada área da empresa, o sistema opera sobre um conjunto reduzido de conceitos fundamentais.

Entretanto, observou-se que processos que dependem fortemente de lógica condicional, regras temporais complexas ou múltiplos recursos simultâneos podem exigir extensões adicionais ao modelo atual.

4.3.2 Comportamento da Simulação e Detecção de Gargalos

A utilização do SimPy permitiu executar o modelo ao longo do tempo e observar o comportamento dinâmico das transações. Nos cenários avaliados, os gargalos manifestaram-se principalmente como acúmulo progressivo de saldo em determinadas contas ou redução significativa na frequência de execução de transações específicas.

No exemplo da padaria, quando a capacidade do Forno era configurada abaixo da taxa de produção do Padeiro, o sistema passou a acumular recursos nas etapas anteriores do processo, indicando a formação de um gargalo operacional. \

Esse comportamento está alinhado ao conceito de gargalo conforme definido pela Teoria das Restrições (GOLDRATT; COX, 1984), na qual a etapa de menor capacidade de um sistema determina o *throughput* global do processo. A modelagem Conta/Transação, ao representar todos os recursos como saldos mensuráveis, viabiliza a identificação automática dessa condição sem que o analista precise inspecionar manualmente cada etapa.

A análise posterior dos resultados permitiu identificar essas situações e apresentar recomendações ao usuário, demonstrando a viabilidade da integração entre modelagem transacional, simulação e análise de BI.

4.3.3 Ameaças à Validade

A principal ameaça à validade externa deste trabalho está relacionada ao ambiente de avaliação adotado. A exemplificação do protótipo foi realizada utilizando um cenário simplificado de uma padaria, modelado especificamente para demonstrar o funcionamento da abordagem proposta. Embora esse exemplo seja suficiente para verificar a correta integração entre a modelagem, a simulação e o mecanismo de análise, ele não permite afirmar que o mesmo desempenho será observado em processos empresariais mais complexos, envolvendo múltiplas restrições, regras condicionais ou grande quantidade de recursos simultâneos.

Outra limitação decorre da utilização de dados sintéticos produzidos durante a própria execução da simulação. Diferentemente de sistemas corporativos de Business Intelligence, que operam sobre registros históricos provenientes de bancos de dados organizacionais, o protótipo trabalha com eventos gerados pelo modelo construído pelo usuário. Dessa forma, aspectos comuns em bases reais, como dados incompletos, inconsistências, registros duplicados, falhas de integração entre sistemas ou informações ausentes, não foram considerados durante a avaliação. Consequentemente, ainda não é possível afirmar como o mecanismo de modelagem e análise se comportaria diante dessas condições.

Também deve ser considerada a escalabilidade da solução. O protótipo foi concebido como uma prova de conceito e avaliado em modelos de pequena dimensão. Embora a representação baseada no padrão Conta/Transação seja independente do domínio de aplicação, não foram realizados experimentos para avaliar seu desempenho em cenários contendo centenas de contas, milhares de transações ou longos períodos de simulação. Assim, questões relacionadas ao

consumo de memória, tempo de processamento e capacidade de análise permanecem como tópicos para investigação futura.

Outra ameaça refere-se ao mecanismo de identificação de gargalos. As recomendações produzidas pelo sistema são baseadas em heurísticas determinísticas simples construídas a partir de métricas como número de execuções, quantidade de bloqueios e estado das contas ao término da simulação. Embora essas métricas estejam alinhadas ao conceito de restrição apresentado pela Teoria das Restrições (GOLDRATT; COX, 1984), não foi realizada uma comparação sistemática entre os diagnósticos produzidos pelo protótipo e avaliações realizadas por especialistas em processos ou por ferramentas comerciais de mineração de processos. Dessa forma, os resultados obtidos devem ser interpretados como uma validação inicial da abordagem proposta, e não como uma comprovação definitiva da eficácia da heurística.

Por fim, a hipótese de que o padrão Conta/Transação possa servir como uma representação conceitual aplicável a diferentes domínios também deve ser interpretada com cautela. O trabalho demonstrou que essa abordagem é adequada para representar um processo produtivo simplificado, porém sua aplicação a áreas como logística, recursos humanos, saúde ou processos financeiros ainda depende de validações adicionais. Assim, os resultados apresentados indicam o potencial de generalização da proposta, mas não permitem afirmar uma aplicabilidade ampla.

4.3.4 Limitações do Protótipo

Apesar dos resultados positivos, o protótipo possui limitações importantes. A interface gráfica foi desenvolvida como prova de conceito e ainda não oferece mecanismos avançados de edição, validação ou visualização de grandes modelos.

Além disso, o armazenamento em JSON é adequado para experimentação, mas não substitui uma solução de persistência mais robusta para ambientes reais.

Outra limitação relevante é que o mecanismo de análise atualmente utiliza heurísticas simples para identificar gargalos. Embora suficientes para os cenários testados, essas heurísticas podem não capturar comportamentos mais complexos presentes em processos empresariais de grande escala.

4.3.5 Possibilidades de Evolução

Os resultados obtidos sugerem várias possibilidades de evolução. Entre elas, destacam-se a integração com bancos de dados relacionais, a importação automática de dados de sistemas corporativos, e o uso de algoritmos de otimização para ajuste de parâmetros.

Também seria possível expandir o modelo para incluir restrições temporais, prioridades entre transações e recursos compartilhados, aproximando o protótipo de cenários industriais mais complexos.

4.3.6 Criação em Tempo de Execução e o Problema das Subclasses

Uma das propriedades mais relevantes do padrão Conta/Transação, é a possibilidade de criar contas e transações em tempo de execução, sem necessidade de alterações no código ou na estrutura do sistema. No contexto do protótipo desenvolvido, isso se manifesta na capacidade do Analista de Processos de adicionar novas contas ao modelo e conectá-las com transações durante a própria sessão de trabalho, sem intervenção técnica.

Essa característica levanta, no entanto, uma questão de design relevante. Se cada domínio de negócio, seja ele financeiro, logístico, ou de recursos humanos, possui contas com comportamentos específicos, seria necessário criar subclasses de Conta

para cada tipo? Uma conta financeira precisa lidar com moedas e câmbio; uma conta de estoque precisa tratar unidades de medida distintas; uma conta de RH precisa registrar colaboradores individualmente. Se cada especificidade gerar uma subclasse, o modelo perde sua aplicabilidade generalizada e retorna à fragmentação que o padrão se propõe a evitar.

O protótipo desenvolvido neste trabalho soluciona essa questão quando, ao em vez de especializar Conta por herança, o sistema utiliza variáveis configuráveis em tempo de execução. Cada conta pode receber qualquer número de variáveis com nome, tipo de dado, valor inicial e restrições definidas pelo próprio usuário. Uma conta financeira pode ter uma variável saldo do tipo *float* com permissão de valores negativos; uma conta de estoque pode ter uma variável quantidade do tipo *int* sem permissão de negativos; uma conta de RH pode ter variáveis *headcount* e disponível do tipo *int* e *bool*, respectivamente. O mecanismo de movimentação e detecção de gargalos opera sobre qualquer uma dessas variáveis sem distinguir seu significado de domínio.

A consequência prática é que o sistema permanece extensível sem que seu núcleo precise ser modificado: adicionar suporte a um novo tipo de processo não requer conhecimento de programação, requer apenas que o analista configure novas variáveis nas contas existentes ou crie novas contas configuradas para o domínio desejado.

Reconhece-se, entretanto, que essa solução tem limites. Processos que demandam lógica condicional complexa entre diversas variáveis, como regras de negócio que dependem de múltiplas variáveis simultaneamente, podem exigir extensões ao modelo de variáveis atual.

5 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho propôs o desenvolvimento de um protótipo de software de *Business Intelligence* fundamentado no padrão de modelagem conceitual de Conta/Transação. A motivação central foi a dificuldade encontrada por gestores em identificar gargalos operacionais devido à falta de padronização nos dados de processos complexos.

Ao longo do desenvolvimento, demonstrou-se que a aplicação do padrão Conta/Transação (WAZLAWICK, 2015) funciona como um elemento integrador. Os resultados obtidos sugerem que o padrão Conta/Transação tem potencial de atuar como estrutura integradora de dados em sistemas de BI. Para validar de forma mais ampla a aplicabilidade do padrão, trabalhos futuros deverão testar o modelo em domínios distintos, como logística, recursos humanos e finanças.

O protótipo desenvolvido em Python validou a viabilidade de criar uma interface visual capaz de mapear esses processos de forma intuitiva. A integração entre a modelagem de sistemas e as ferramentas de análise de dados mostrou-se eficaz para a detecção de pontos de congestionamento, uma vez que o padrão transacional fornece, intrinsecamente, as métricas de entrada, saída e acúmulo necessárias para o cálculo de gargalos.

A união entre BI e o padrão Conta/Transação, portanto, além de simplificar a arquitetura do software, demonstra potencial para ser aplicada em diferentes áreas. Essa abordagem propõe um caminho para mitigar a desconexão apontada inicialmente entre a modelagem de sistemas de informação e as necessidades estratégicas de análise de negócio.

Como sugestões para trabalhos futuros, aponta-se a necessidade de testar o protótipo em cenários de alta volumetria de dados e a implementação de algoritmos de inteligência artificial para a previsão preditiva de gargalos baseada no histórico de transações acumulado, além da expansão dos domínios de teste, como já mencionado.

REFERÊNCIAS

- WAZLAWICK, Raul Sidnei. **Análise e design orientados a objetos para sistemas de informação**: modelagem com UML, OCL e IFML. 3. ed. Rio de Janeiro: Elsevier: Campus, c2015. 462 p. ISBN 9788535279849.
- SEZÕES, C., OLIVEIRA, J., & BAPTISTA, M; **Business Intelligence**. SPI Sociedade Portuguesa de Inovação, 2006.
- OLSZAK, C. M., & ZIEMBA, E. (2004). **Business intelligence systems as a new generation of decision support systems**. Proceedings of PISTA 2004, International Conference on Politics and Information Systems: Technologies and Applications.
- REINSCHMIDT, J., & FRANCOISE, A. (2000). **Business Intelligence Certification Guide**. IBM.
- SILVA, Ricardo Pereira e. **UML 2**: modelagem orientada a objetos. Florianópolis: Visual Books, 2007. 232 p. ISBN 8587393588.
- GRETA VALENTI, **Business Intelligence: the implementation of a BI solution to a case study**. Rel. Luca Mastrogiacomo. Politecnico di Torino, Corso di laurea magistrale in Ingegneria Gestionale (Engineering And Management), 2022
- JOSÉ EDUARDO DE SOUSA FREITAS FERREIRA **Implementation of a business intelligence solution: a case study of a workforce and staffing solutions company** disponível em: <https://hdl.handle.net/10216/144340>, 2022
- JAMALADDIN AZIZOV. **ExpenseTracker - Personal Finance Manager. Bachelor's Thesis**. České vysoké učení technické v Praze, 2021
- GOLDRATT, Eliyahu M.; COX, Jeff. *A Meta: um processo de melhoria contínua*. São Paulo: Nobel, 1984.

IBM. **Cognos Analytics**. 2024. Disponível em:

<https://www.ibm.com/products/cognos-analytics>. Acesso em: 8 de Julho de 2024.

MICROSOFT. **Power BI**, 2024. Disponível em:

https://www.microsoft.com/pt-br/power-platform/products/power-bi#tabs-pill-bar-ocb9d418_tab1. Acesso em: 8 de Julho de 2024.

TEAM SIMPY. SimPy: Discrete-event simulation for Python. Versão 4.1.2. Disponível em: <https://simpy.readthedocs.io>. Acesso em: 15 de Junho de 2026.

APÊNDICE A - CÓDIGOS FONTES DAS APLICAÇÕES DESENVOLVIDAS NO PROJETO

O código fonte para o software desenvolvido se encontra em:

<https://codigos.ufsc.br/enzo.sonego/TCC>

APÊNDICE B – ARTIGO NO FORMATO SBC

Desenvolvimento de um Software de Business Intelligence que modela processos de negócio, baseado no padrão de modelagem conceitual de Conta/Transação

Enzo Gomes Sônego¹

¹Departamento de Informática e Estatística – Centro Tecnológico (CTC)
Universidade Federal de Santa Catarina (UFSC)
Florianópolis – SC – Brasil

enzo.gomes.sonego@grad.ufsc.br

Abstract. *A bottleneck in a business process represents a limitation that can affect a company's efficiency, resulting in delays, reduced productivity, and increased costs. Unmitigated bottlenecks can create task queues, resource idleness, decreased product quality, and negative impacts on a company's reputation. Business Intelligence (BI) tools can help identify and mitigate these bottlenecks by enabling data analysis and evidence-based decision-making. This project aims to create a BI software prototype to demonstrate the advantages of the Account/Transaction pattern, to document the modeling and development process, and to examine the applicability of the Account/Transaction model for representing business processes across various domains.*

Keywords: *Bottleneck, Business Intelligence, Account/Transaction.*

Resumo. *Um gargalo em um processo empresarial representa uma limitação que pode afetar a eficiência de uma empresa, resultando em atrasos, menor produtividade e aumento de custos. Gargalos não mitigados podem causar o enfileiramento de tarefas, a ociosidade de determinado recurso, diminuição da qualidade do produto e impactos negativos na reputação da empresa. Ferramentas de Business Intelligence (BI) podem ajudar a identificar e resolver estes gargalos, permitindo a análise de dados e a tomada de decisão baseada em fatos. Este trabalho se propõe a criar um protótipo de um software de BI para a análise das vantagens do padrão conta/transação, e documentar o processo de modelagem e desenvolvimento, analisando a aplicabilidade do padrão conta/transação de modelar processos empresariais de variadas áreas.*

Palavras-chave: *Gargalo, Business Intelligence, Conta/Transação.*

1. Introdução e Objetivos

1.1. Introdução

Em uma empresa, a eficiência organizacional é frequentemente limitada pela presença de gargalos em seus processos. Um gargalo é aquele recurso cuja capacidade é igual ou menor que a demanda colocada nele [Goldratt and Cox 1984], causando o enfileiramento de tarefas, ociosidade de recursos e atrasos na entrega. Além de aumentar os custos operacionais com horas extras e prejudicar a qualidade do produto final, essas ineficiências impactam negativamente a experiência do cliente e a motivação dos colaboradores.

Para mitigar esses problemas, gestores dependem de dados precisos para prever prazos e tomar decisões. É nesse contexto que o Business Intelligence (BI) se destaca. Segundo [Reinschmidt and Francoise 2000], o BI torna o processo decisório mais rápido e fundamentado em fatos. Ferramentas de BI permitem não apenas a análise do estado atual do negócio, mas também a simulação de cenários futuros para testar estratégias de mitigação e realizar *benchmarks*.

No entanto, o grande desafio para a implementação de um sistema de BI eficiente é a complexidade e a heterogeneidade dos dados gerados por diferentes setores de uma empresa. Para que o BI consiga identificar gargalos com precisão, os dados dos processos precisam estar estruturados de forma padronizada. É aqui que entra o padrão de modelagem conceitual de Conta/Transação, definido por [Wazlawick 2015].

Poderia o padrão Conta/Transação atuar como um tradutor geral para os processos de negócio? A resposta que este trabalho propõe se baseia em quatro propriedades do padrão que o tornam especialmente adequado como fundamento para sistemas de BI. Primeiro, ao representar qualquer processo como movimentações de saldos entre contas, o padrão impõe uma estrutura de dados uniforme a domínios heterogêneos, eliminando a necessidade de transformações para compatibilizar fontes distintas. Segundo, as métricas fundamentais para detecção de gargalos, sendo elas: saldo, taxa de execução e taxa de bloqueio de transações, são intrínsecas ao próprio padrão, dispensando a definição manual de KPIs para cada domínio. Terceiro, a possibilidade de simular o modelo antes da coleta de dados históricos habilita uma forma de BI preditivo inacessível às ferramentas tradicionais. Quarto, a criação de novas contas e transações em tempo de execução (ou, no caso do protótipo produzido, na mesma sessão de trabalho) permite que o modelo seja adaptado a novos processos sem alterações estruturais no sistema.

Ao modelar qualquer atividade empresarial, seja a movimentação de estoque, alocação de horas de funcionários ou fluxo de pedidos, estritamente como transferências de valores entre “contas”, os dados perdem sua complexidade isolada e ganham uma estrutura uniforme. Essa padronização viabiliza a criação de um motor de simulação genérico e flexível.

A meta deste trabalho é, portanto, a criação de um protótipo de software de BI que utilize a ampla aplicabilidade do padrão Conta/Transação como base arquitetural. O objetivo é demonstrar como essa modelagem alimenta o software e potencializa a identificação e resolução de gargalos organizacionais.

1.2. Justificativa

O desenvolvimento de soluções de BI exige fontes de dados consistentes para gerar análises confiáveis. Muitas empresas enfrentam dificuldades para integrar dados de diferentes naturezas ao tentar mapear, diagnosticar e resolver gargalos operacionais em seus fluxos de trabalho.

A principal justificativa para este projeto reside na proposta de unir a capacidade analítica do BI com a simplificação e padronização estrutural oferecida pelo padrão Conta/Transação [Wazlawick 2015]. Enquanto grande parte das ferramentas de BI disponíveis no mercado foca predominantemente na visualização de dados em *dashboards* após o processamento, este trabalho propõe que a modelagem transaccional dos processos ocorra na base do sistema, servindo como o motor da análise.

Ao implementar um protótipo de software fundamentado nessa premissa, o trabalho busca exemplificar, na prática, a ampla aplicabilidade do padrão Conta/Transação para processos administrativos. A documentação deste processo de modelagem e desenvolvimento servirá para validar a viabilidade técnica de usar o padrão como um alicerce de dados robusto, estabelecendo uma base sólida para a criação de futuras ferramentas de BI mais ágeis e adaptáveis a diferentes realidades empresariais.

1.3. Objetivos Gerais

Desenvolvimento de um Software de Business Intelligence que modela processos de negócio, baseado no padrão de modelagem conceitual de Conta/Transação.

1.4. Objetivos Específicos

1. Avaliar a aplicabilidade do padrão Conta/Transação como fundamento para sistemas de BI.
2. Projetar e implementar um protótipo que integre modelagem Conta/Transação, simulação de eventos discretos e análise heurística de gargalos.
3. Verificar a viabilidade técnica da abordagem, discutindo os limites observados e os requisitos para estendê-la a outros domínios de processo.

1.5. Estrutura do Documento

Nesta seção inicial, o trabalho é introduzido, assim como são abordadas a justificativa e os objetivos. Na Seção 2 são apresentados conceitos relevantes e as tecnologias que serão utilizadas. A Seção 3 é dedicada a apresentar trabalhos relacionados e como este trabalho se diferencia de outros similares. Na Seção 4 é apresentado o desenvolvimento do software e suas funcionalidades, são discutidos os resultados obtidos e as limitações encontradas. Por fim, a Seção 5 mostra as conclusões e apresenta propostas para trabalhos futuros.

2. Fundamentação Teórica

2.1. Business Intelligence

2.1.1. Definições e relevância

Segundo [Sezões et al. 2006], Business Intelligence engloba um vasto conjunto de aplicações de apoio à tomada de decisão que possibilitam acesso rápido, compartilhado e interativo às informações, bem como sua análise e manipulação.

Como mencionado por [Reinschmidt and Francoise 2000], a consolidação e organização de dados com o objetivo de melhorar a tomada de decisão pode gerar vantagem competitiva. O conhecimento sobre como identificar essas vantagens e utilizá-las constitui a base do Business Intelligence. Mais do que a combinação de dados e tecnologia, o BI ajuda pessoas a criar conhecimento a partir de grandes volumes de informação, desde que se obtenham os dados corretos, se extraia deles valor e esse valor seja compartilhado. Dessa forma, BI transforma informação em conhecimento ao colocar os dados certos nas mãos das pessoas certas, no momento adequado, oferecendo suporte efetivo ao processo decisório.

Ainda segundo [Sezões et al. 2006], a área de BI surge nas empresas acompanhando tendências como:

- Grande intensidade concorrencial na maioria dos mercados;
- Exigência crescente por padrões elevados de qualidade em produtos e serviços;
- Necessidade de compreender mercados e clientes cada vez mais segmentados e específicos;
- Demanda por racionalização de processos internos e redução de custos operacionais;
- Imperativo de avaliar, em tempo real, o desempenho organizacional, garantindo decisões em tempo útil;
- Necessidade de conhecer, controlar e mitigar riscos de negócio associados a cada atividade.

Além dessas tendências, observa-se o movimento de horizontalização das empresas, que se tornam menos hierarquizadas, adotam *downsizing* e concedem maior autonomia aos níveis operacionais. Esses processos exigem que informações críticas sejam disseminadas para um número maior de colaboradores, especialmente gestores.

Todos esses fatores contribuem para que as empresas busquem identificar o que é essencial em seu negócio por meio de diagnóstico, análise, acessibilidade, compartilhamento e elaboração de relatórios, transformando o volume de dados coletados em mudanças que geram resultados — isto é, em conhecimento útil de forma eficaz.

Na subseção seguinte, serão discutidas as origens dos sistemas de suporte à decisão, bem como apresentados exemplos de aplicações que os utilizam em diferentes setores do mercado.

2.1.2. Exemplos de Aplicações

Alguns dos setores que mais usam sistemas de BI incluem corretoras de ações, seguradoras, bancos e empresas do setor financeiro, telecomunicações e empresas de manufatura [Olszak and Ziemba 2004]. A Tabela 1, traduzida de [Olszak and Ziemba 2004] e baseada em [Reinschmidt and Francoise 2000], lista os objetivos das aplicações de BI utilizadas em diversos setores.

Tabela 1. Objetivos de BI em setores variados

Setor	Objetivos
Varejo	• Previsão de Demanda. Usa dados provenientes da digitalização para prever a demanda e, com base nesta previsão, definir requisitos de estoque • Pedidos e Reabastecimento. Usa informações para tomar decisões mais rapidamente sobre itens a serem pedidos e otimizar quantidades • Marketing. Fornece análise das transações de clientes (o que está vendendo e quem está comprando) • Distribuição e Logística. Ajuda os centros de distribuição a gerenciar volumes maiores e pode usar informações antecipadas de remessa para programar e consolidar entrada e saída de produtos • Gestão de Transportes. Ajuda no desenvolvimento de planos ideais de consolidação de carga e cronogramas de rotas a serem tomadas • Planejamento de Estoque. Auxilia na identificação do nível de estoque necessário garantindo um grau de serviço
Seguradoras	• Análise de Sinistros e Prêmios. Providencia a capacidade de analisar reivindicações detalhadas e históricos máximos por produto, política, tipo de reivindicação e outras especificações • Análise de Clientes. Permite analisar as necessidades do cliente e os padrões de uso do produto, ajuda a desenvolver programas de marketing com base nas características do cliente, analisar os riscos e melhorar o atendimento ao cliente • Análise de Risco. Ajuda a identificar segmentos de mercado de alto risco e oportunidades em segmentos específicos, relacionando segmentos de mercado e reduzindo a frequência de sinistros
Financeiro	• Análise de Rentabilidade do Cliente. Auxilia a determinar a lucratividade geral de clientes, atual e de longo prazo, fornecendo a base para vendas de lucro alto e relacionamento com o banco, maximizando as vendas para clientes de alto valor, reduzindo custos para clientes de baixo valor, fornecendo os meios para maximizar a lucratividade de novos produtos e serviços • Administração de Crédito. Apoia o estabelecimento de padrões de progressão de problemas de crédito por classe e tipo de cliente, alertando os clientes para evitar problemas de crédito, ajudando na gestão de limites de crédito, na avaliação da carteira de crédito do banco e na redução de perdas de crédito • Vendas de Filiais. Melhora o atendimento ao cliente e a venda de contas e facilita a venda cruzada, melhorando o suporte ao cliente e fortalecendo a fidelidade à empresa
Telecomunicações	• Categorização e Segmentação de Clientes. Contribui para a determinação de perfis de produtos e segmentos de clientes que geram mais lucro, para o fornecimento de perfis de clientes detalhados e integrados, para o desenvolvimento de programas individualizados de chamadas frequentes e a determinação de futuras necessidades de clientes • Previsão da Demanda do Cliente. Facilita a previsão das necessidades futuras de produtos ou serviços, fornecendo base para a análise do controle de rotatividade para melhorar a retenção de clientes

continua na próxima página

Tabela 1 – continuação da página anterior

Setor	Objetivos
Manufatura	• Vendas. Permite analisar dados de transações específicas de clientes • Previsão. Ajuda na previsão da demanda e requisitos de estoque • Pedidos e Reabastecimento. Permite encomendar quantidades ideais de itens • Compra. Ajuda os centros de distribuição a gerenciar volumes maiores • Distribuição e Logística. Pode usar informações antecipadas de remessa para programar e consolidar frete de entrada e saída • Gestão de Transportes. Auxilia a desenvolver planos ideais de consolidação de carga e cronogramas de roteamento • Planejamento de Estoque. Permite visualizar o nível de estoque atual e necessário para garantir um determinado grau de serviço

2.2. Conta/Transação

Como descrito em [Wazlawick 2015], o padrão conta/transação é fortemente relacionado à área de negócios, mas tem variadas aplicações em múltiplas áreas. O exemplo do livro se refere a uma empresa que vende livros, e são citadas certas movimentações que são feitas acima dos produtos: estes livros podem ser catalogados, pedidos, recebidos, enviados, descartados etc. Estas movimentações, tanto quanto as transações financeiras associadas, dão origem a conceitos como Pedido, Envio, Descarte, Devolução etc., cada um com determinados atributos e associações.

O mesmo padrão pode ser aplicado a domínios distintos. Em um sistema de controle de horas de trabalho, por exemplo, as contas representariam departamentos e colaboradores, enquanto as transações modelariam alocações de tempo. Na modelagem de um sistema logístico, por outro lado, as contas seriam centros de distribuição, e as transações, transferências de mercadorias entre eles. Esta aplicabilidade estrutural é o que fundamenta a proposta deste trabalho.

Um modelo de negócio que tem conceitos similares a estes tem em seu núcleo dois artefatos conceituais que podem ser explorados. Um deles seria a *Conta*, que pode ser simplificada para algo que contém quantidades de algo (itens, produtos ou dinheiro, por exemplo); esta quantidade é representada pelo *Saldo* da conta, que consiste da soma de todos os depósitos e saques de quantidades variadas da “coisa” representada pelo saldo.

A partir do momento que se identifica a existência de saques e depósitos, pode-se concluir que estes são apenas movimentações de uma conta para outra, portanto uma *Transação* consiste de duas movimentações: um depósito em uma conta e uma retirada do mesmo valor de outra conta.

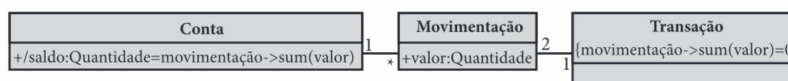


Figura 1. Classes Conta/Transação

Para exemplificar as capacidades do padrão, [Wazlawick 2015] descreve a modelagem de uma empresa hipotética da seguinte maneira:

- Inicialmente são criadas cinco contas, que representam fornecedores, pedidos pendentes, produtos em estoque, produtos vendidos e produtos enviados, todas elas com saldo zerado.

Figura 2. Estado Inicial Exemplo

- Então, uma transação que modela um pedido de compra é criada; esta transação movimenta 50 itens da editora para os pedidos pendentes.

Figura 3. Pedido de Compra Criado Exemplo

- Outras transações são adicionadas para modelar uma sequência de eventos, no caso, 40 itens sendo entregues, 25 destes sendo vendidos e então enviados para o cliente.

Figura 4. Modelagem Completa Exemplo

Uma variação de movimentação dentro deste padrão que se mostra relevante e deve ser levada em consideração neste trabalho são as *movimentações mnemônicas*, que, como descritas em [Wazlawick 2015], são movimentações que não necessariamente modelam transações entre duas contas, mas sim guardam alguma informação relevante, o exemplo canônico sendo uma transação monetária entre duas contas tendo que guardar uma parcela desta para modelar uma tributação hipotética que será paga em um momento futuro.

Estas movimentações mnemônicas se diferenciam de movimentações normais pelo fato de que uma transação necessariamente pode ter no máximo 2 movimentações normais (uma da conta fonte e a outra para a conta destino), porém esta mesma transação pode ter uma quantidade grande de movimentações mnemônicas, por se tratar de uma “anotação” e não uma operação que de fato altera saldos de contas.

2.3. Conclusão do Capítulo

Estabelecidos os fundamentos teóricos de Business Intelligence e do padrão Conta/Transação, cabe agora articular explicitamente como esses elementos se relacionam na proposta deste trabalho.

A principal dificuldade enfrentada por sistemas de BI reside na heterogeneidade dos dados gerados por processos empresariais distintos. Cada área da empresa, seja ela financeira, logística, recursos humanos ou de produção, gera dados em formatos e estruturas diferentes, exigindo processos de integração complexos antes que qualquer análise seja possível. O padrão Conta/Transação atua como um elemento normalizador ao nível conceitual: ao forçar que qualquer processo seja descrito como transferências de quantidades entre contas, ele impõe uma estrutura única independentemente do domínio. O motor de análise do sistema de BI, ao operar sobre essa estrutura uniforme, não precisa ser adaptado a cada novo processo, ele sempre encontrará contas com saldos e transações com frequências de execução.

Essa uniformidade estrutural tem uma consequência direta para o BI: as métricas necessárias para identificar gargalos operacionais são intrínsecas ao padrão. O saldo crescente de uma conta indica acúmulo de recursos naquela etapa do processo; a alta taxa de bloqueio de uma transação indica que ela não consegue executar por insuficiência de recursos na conta de origem. Esses indicadores emergem naturalmente de qualquer modelo Conta/Transação, sem que o analista precise defini-los manualmente para cada domínio.

3. Trabalhos Relacionados

3.1. Produções Acadêmicas

Neste capítulo são apresentados trabalhos correlatos cujo tema é relacionado com este, para simultaneamente auxiliar no desenvolvimento, auxiliando no desvio de possíveis erros evitáveis, tanto como justificar a existência deste por não abordar o tema da mesma maneira. Também serão incluídos softwares que estão atualmente no mercado para levantar o estado da arte e justificar a presente proposta de trabalho.

3.1.1. Business Intelligence: the implementation of a BI solution to a case study (Valenti, 2022)

O trabalho de [Valenti 2022], desenvolvido no Politecnico di Torino, implementa uma solução de BI para um estudo de caso empresarial específico. Sua contribuição mais relevante para este trabalho reside no detalhamento da arquitetura clássica de sistemas de BI: o modelo composto por fontes de dados heterogêneas, processos ETL (*Extract, Transform, Load*), *Data Warehouse* e ferramentas de visualização por meio de *dashboards* interativos. Essa descrição permite compreender como os sistemas de BI tradicionais estruturam o fluxo de dados desde sua origem até a geração de relatórios para tomada de decisão.

Entretanto, essa mesma arquitetura evidencia uma limitação central: o processo ETL pressupõe que os dados dos processos de negócio já estão disponíveis em fontes estruturadas como bancos de dados operacionais ou planilhas. A etapa de transformação do ETL tem por objetivo padronizar dados que já existem, mas que foram gerados por sistemas heterogêneos. Isso significa que o sistema de BI é reativo: ele analisa dados produzidos por processos que já ocorreram, sem participar do processo de concepção destes processos.

Este trabalho se diferencia: antes de fazer qualquer análise, modela-se diretamente no software utilizando o padrão Conta/Transação, e a própria modelagem gera os dados que alimentarão a simulação e a análise de gargalos, em vez de depender de fontes de dados preexistentes. Essa diferença arquitetural posiciona o protótipo proposto não como um substituto dos sistemas de BI tradicionais, mas como um complemento voltado a cenários em que o processo ainda não está suficientemente estruturado para alimentar um *pipeline* ETL convencional.

3.1.2. Implementation of a business intelligence solution: a case study of a workforce and staffing solutions company (Ferreira, 2022)

O trabalho de [Ferreira 2022] descreve o desenvolvimento de uma solução de BI para uma empresa do setor de recursos humanos, motivado pela ausência de um mecanismo ágil de acesso a informações gerenciais para o *Chief Executive Officer* (CEO) da organização. A solução implementada organiza as informações em seis categorias de *dashboards*: visão geral da empresa, situação financeira, análise de clientes, análise de colaboradores, KPIs e previsões. Cada categoria agrega métricas como número de novos clientes, taxa de retenção de colaboradores e receita por cliente, e as apresenta de forma consolidada para apoio à tomada de decisão.

A principal contribuição analítica deste trabalho, para este TCC, é demonstrar quais tipos de métricas são relevantes para a gestão de um processo empresarial do ponto de vista do BI. Métricas como número de colaboradores ativos, número de reuniões e taxa de novos colaboradores por mês têm correspondência direta com o padrão Conta/Transação, por exemplo, colaboradores ativos podem ser modelados como o saldo de uma conta, e contratações ou desligamentos como transações que alteram esse saldo ao longo do tempo. Essa correspondência sugere que processos de RH são modeláveis pela estrutura proposta neste trabalho, o que amplia o escopo de aplicação do protótipo.

Entretanto, a abordagem de [Ferreira 2022] parte do pressuposto de que os dados já estão disponíveis nos sistemas internos da empresa e precisam apenas ser extraídos, agregados e visualizados, similar ao trabalho de Valenti introduzido anteriormente. A solução não propõe nenhum mecanismo de padronização conceitual dos processos antes da coleta dos dados. Isso limita sua aplicabilidade a quaisquer empresas que já possuam uma estrutura de processos definida, uma condição que muitas organizações de pequeno e médio porte não satisfazem. O presente trabalho busca preencher essa lacuna ao propor que a modelagem transacional dos processos precede e orienta a coleta de dados, tornando o sistema aplicável em contextos onde os processos ainda não foram formalmente ou conceitualmente estruturados.

3.1.3. Expensetracker – Personal Finance Manager (Azizov, 2021)

O trabalho de [Azizov 2021] propõe um gerenciador de finanças pessoais cujo modelo de domínio apresenta semelhança estrutural relevante com o padrão Conta/Transação. Conforme ilustrado na Figura 5, o modelo define as entidades centrais *Account* e *Transaction*, mantendo entre elas uma relação que espelha o núcleo do padrão de [Wazlawick 2015]: contas contêm saldos e transações movimentam valores entre elas.

A análise do diagrama de classes revela, entretanto, diferenças significativas decorrentes da especialização do domínio financeiro. O modelo de Azizov introduz entidades como *Budget*, *BudgetHistory*, *Category* e *Currency*, além de uma enumeração de tipos de conta (*Cash*, *CreditCard*, *CurrentAccount*, *Savings*). Cada uma dessas adições resolve um problema específico do domínio financeiro pessoal: o orçamento (*Budget*) impõe restrições ao gasto; a categoria (*Category*) classifica as transações para análise posterior; a moeda (*Currency*) permite operações multimooeda. Esses elementos são necessários no contexto financeiro, mas introduzem dependências de domínio que reduzem

a generalidade do modelo.

É exatamente nesse ponto que reside a diferença arquitetural central em relação a este trabalho. O padrão Conta/Transação de [Wazlawick 2015] opera em um nível de abstração mais alto: contas representam qualquer entidade que contenha um recurso quantificável, e transações representam qualquer movimentação desse recurso entre contas, independentemente do domínio. Ao abrir mão de especializações como *Budget* e *Category*, o padrão ganha em aplicabilidade o que perde em especificidades do domínio. O protótipo desenvolvido neste trabalho herda essa filosofia: as variáveis de cada conta podem ser configuradas pelo usuário, permitindo que o mesmo mecanismo modele pães num forno, tickets num sistema de suporte ou horas alocadas ao desenvolvimento de um projeto.

A contribuição analítica do trabalho de [Azizov 2021] para o presente TCC está, portanto, não no que os dois modelos compartilham, mas no que eles diferem, demonstrando como a especialização de domínio, ao ser introduzida no nível estrutural do modelo, limita sua reutilização em outros contextos, argumento que reforça a escolha do padrão Conta/Transação como base de um sistema de BI de propósito mais geral.

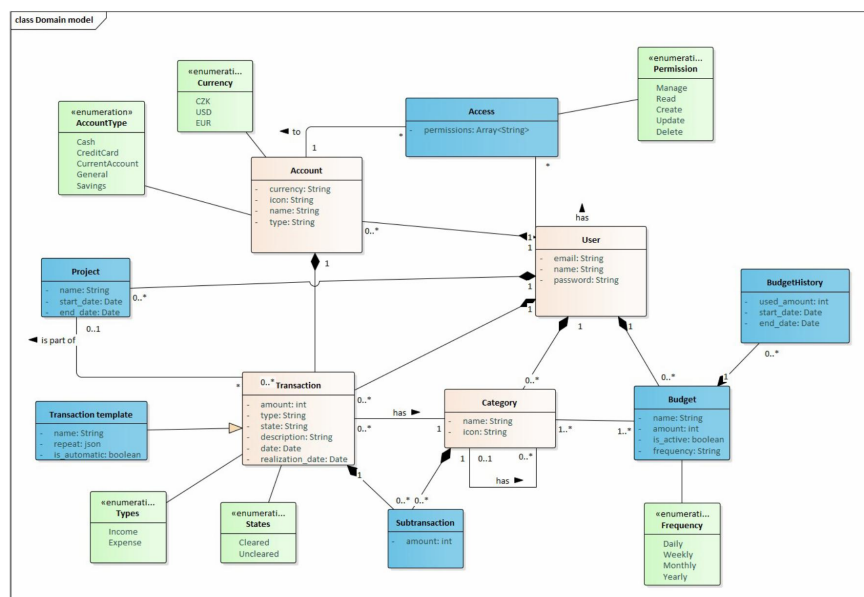


Figura 5. Modelagem [Azizov 2021]

3.2. Softwares no Mercado

Para posicionar a proposta deste trabalho em relação às ferramentas comerciais disponíveis, foram analisados dois dos principais softwares de BI do mercado: IBM Cognos Analytics e Microsoft Power BI. A análise foi conduzida com base em três critérios relevantes para o escopo deste TCC:

1. a capacidade de modelar processos de negócio internamente, sem depender de fontes de dados preexistentes;
2. o suporte a simulação de comportamento dinâmico ao longo do tempo;
3. a capacidade de analisar processos de diferentes áreas.

3.2.1. IBM Cognos Analytics

O IBM Cognos Analytics é uma plataforma consolidada de BI voltada à exploração de dados, criação de *dashboards* interativos e geração de relatórios corporativos [IBM 2024]. Suas principais funcionalidades incluem modelagem semântica de dados, conectividade com múltiplas fontes, análise preditiva por séries temporais e recursos de inteligência artificial para sugestão automática de visualizações.

Em relação ao critério 2, o Cognos possui “elementos de simulação” por oferecer previsão estatística e árvores de decisão. É necessário, entretanto, distinguir esses dois mecanismos: previsão estatística (*forecasting*) consiste em extrapolar tendências de dados históricos para estimar valores futuros, enquanto simulação de eventos discretos modela o comportamento dinâmico de um sistema ao longo do tempo, com eventos que alteram o estado da simulação conforme regras definidas pelo usuário. O Cognos realiza o primeiro, mas não o segundo. Em relação ao critério 1, a plataforma pressupõe que os dados dos processos já existem em fontes estruturadas, não há mecanismo nativo para que o usuário modele um processo antes de coletá-lo.

3.2.2. Microsoft Power BI

O Microsoft Power BI é atualmente a ferramenta de BI mais amplamente adotada em ambientes corporativos, com destaque para sua interoperabilidade com o ecossistema Microsoft (Excel, SharePoint, Azure) e sua curva de aprendizagem relativamente acessível para criação de *dashboards* [Microsoft 2024]. O produto oferece modelagem de dados via DAX (*Data Analysis Expressions*), conectividade com centenas de fontes de dados e recursos de análise colaborativa em nuvem.

Um aspecto que merece atenção especial é a funcionalidade *Process Mining*, integrada ao Microsoft Power Automate desde 2022. Esse recurso permite que a plataforma importe *logs* de eventos de sistemas corporativos e gere automaticamente mapas de processos, identificando variações, gargalos e desvios de fluxo. A funcionalidade comercial mais próxima do objetivo deste TCC.

Entretanto, o *Process Mining* do Power Automate opera sobre dados históricos já registrados nos sistemas da empresa, em organizações que ainda não possuem sistemas capazes de gerar esses *logs* estruturados, a ferramenta é dificilmente aplicável.

3.2.3. Comparação entre os Softwares do Mercado e o Protótipo Proposto

Nenhuma das ferramentas analisadas satisfaz simultaneamente os três critérios estabelecidos. Ambas operam sobre dados preexistentes, não suportam simulação de eventos discretos baseada em modelos definidos pelo usuário, e não oferecem um mecanismo de

modelagem conceitual independente de domínio. O protótipo proposto neste trabalho ocupa uma lacuna específica, a da capacidade de modelar um processo de negócio antes da existência de dados históricos, simulá-lo, e identificar possíveis gargalos com base em um padrão conceitual genérico. Nesse sentido, o software desenvolvido é complementar às ferramentas de mercado, podendo até servir de inspiração para futuros *plugins* ou funcionalidades adicionais.

3.3. Conclusão do Capítulo

A Tabela 2 sintetiza a comparação entre os trabalhos relacionados, as ferramentas comerciais analisadas e a proposta deste trabalho.

Tabela 2. Comparação entre os trabalhos relacionados e a proposta deste trabalho

Trabalho	Domínio	Independência de dados pré-existentes	Simulação com portamento dinâmico	Modelagem conceitual independente de domínio	Foco em detecção de gargalos
Valenti (2022)	Estudo de caso empresarial	Não	Não	Não	Não
Ferreira (2022)	Recursos Humanos	Não	Não	Não	Não
Azizov (2021)	Finanças pessoais	Sim	Não	Não	Não
IBM Cognos Analytics	BI corporativo genérico	Não	Não*	Não	Não
Microsoft Power BI (Process Mining)	BI corporativo + mineração de processos	Não	Não**	Não	Sim
Protótipo proposto (este trabalho)	Genérico por design	Sim	Sim	Sim	Sim

*Cognos oferece previsão estatística (*forecasting*), não simulação de eventos discretos.

**Power BI Process Mining analisa *logs* históricos já registrados; não simula cenários hipotéticos.

A análise dos trabalhos relacionados permite posicionar com maior precisão a contribuição do presente trabalho. Os estudos acadêmicos de [Valenti 2022] e [Ferreira 2022] demonstram como sistemas de BI são implementados na prática, mas compartilham uma premissa comum: a existência prévia de dados estruturados nos sistemas da organização. O trabalho de [Azizov 2021], por sua vez, ilustra como o padrão transacional pode ser especializado para um domínio específico e evidencia por contraste a importância da abstração proposta por [Wazlawick 2015].

As ferramentas comerciais, o IBM Cognos Analytics e o Microsoft Power BI, oferecem soluções maduras para análise e visualização de dados históricos. O recurso de *Process Mining* do Power Automate é o mecanismo comercial mais próximo do objetivo deste TCC, mas igualmente dependente de *logs* de eventos preexistentes. Nenhuma das ferramentas analisadas suporta a modelagem conceitual de processos por meio de um padrão transacional genérico, nem a simulação preditiva baseada nessa modelagem.

Essa lacuna justifica a proposta deste trabalho e define seu escopo de contribuição. O protótipo desenvolvido não pretende substituir as plataformas de BI existentes, mas

oferecer um mecanismo complementar voltado a contextos em que o processo ainda não está suficientemente estruturado para alimentar as ferramentas tradicionais.

4. Desenvolvimento

4.1. Prova de Conceito

Além do estudo teórico feito até então, da parte prática do trabalho, foram feitos diagramas UML para modelar estruturalmente e comportamentalmente o software a ser finalizado. Foram identificados 3 componentes principais do software: a modelagem, a simulação e a análise, com seus correspondentes casos de uso.

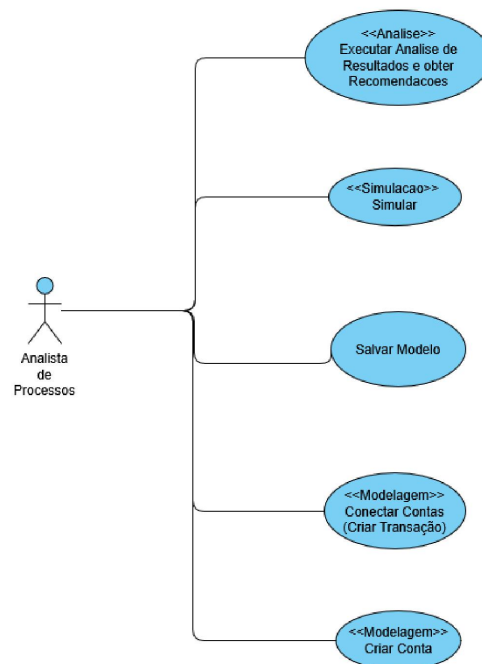


Figura 6. Casos de Uso Software

Também para garantir a adesão ao padrão conta/transação, a modelagem do programa, desde o início, foi deliberadamente feita para implementar o funcionamento transacional, tal qual um sistema financeiro.

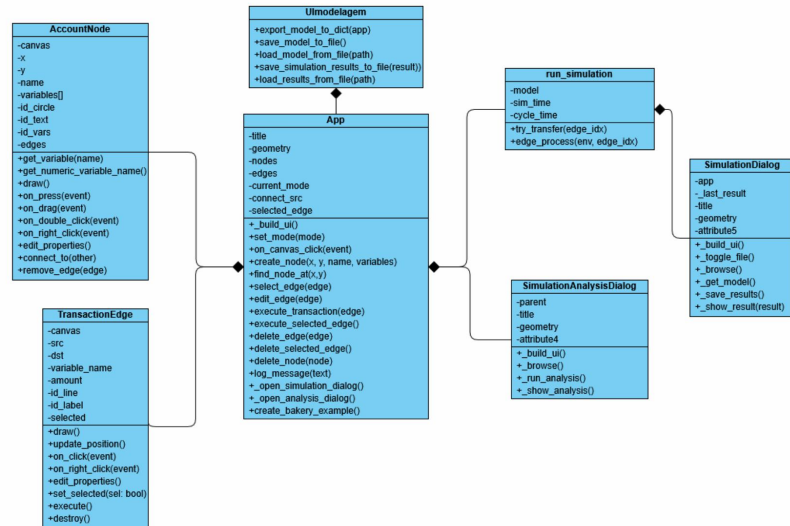


Figura 7. Diagrama de Classes Software

4.2. Implementação

O código da implementação se encontra no repositório: <https://codigos.ufsc.br/enzo.sonego/TCC>. As tecnologias e bibliotecas utilizadas incluem o SimPy [Team SimPy 2024], tkinter, json e threading.

Nas seções seguintes serão apresentadas as regras de negócio e o uso do software para realizá-las.

4.2.1. Criando Contas

Uma conta representa a principal unidade de modelagem do sistema. Conforme definido pelo padrão Conta/Transação, uma conta corresponde a uma entidade que contém determinado recurso, cuja quantidade é representada por um saldo. Dependendo do domínio modelado, esse recurso pode representar produtos em estoque, horas de trabalho disponíveis, pedidos pendentes, tickets de suporte, ou qualquer outro elemento passível de movimentação entre etapas de um processo.

Na implementação desenvolvida, uma conta é representada visualmente por um nodo no ambiente de modelagem, similar a um estado de uma máquina de estados finita. Em sua forma mais simples, ela possui apenas um nome e um saldo. Entretanto, para permitir a representação de cenários mais complexos, uma conta pode armazenar múltiplas variáveis, cada uma com seu próprio tipo e conjunto de restrições.

As variáveis atualmente suportadas incluem números inteiros, números de ponto flutuante e valores booleanos. Essa flexibilidade permite representar tanto recursos quan-

titativos quanto estados lógicos relevantes para determinados processos de negócio onde podem ocorrer gargalos.

Durante a construção de um modelo, o Analista de Processos deve inicialmente identificar quais elementos do processo podem ser representados como contas. Após ativar o modo de criação por meio do botão 'Criar Conta', novos nodos podem ser inseridos na área de modelagem utilizando o botão esquerdo do mouse.



Figura 8. Representação de Conta

4.2.2. Conectando Contas (Criando Transações)

Embora as contas representem os estados do processo modelado, são as transações que definem a dinâmica do sistema. Uma transação representa a movimentação de um recurso entre duas contas, caracterizando uma mudança de estado dentro do processo analisado.

Na interface do software, as transações são representadas visualmente por setas direcionadas. O ponto de origem corresponde à conta da qual o recurso será removido, enquanto o ponto de destino identifica a conta que receberá esse recurso.

Além das contas de origem e destino, cada transação possui informações adicionais responsáveis por definir qual variável será movimentada e qual quantidade será transferida a cada execução.

A criação de transações está sujeita a restrições de consistência. Como uma movimentação pressupõe compatibilidade entre os recursos envolvidos, apenas contas que possuam variáveis compatíveis podem ser conectadas.

Para criar uma transação, o Analista deve ativar o modo de conexão através do botão 'Modo Conectar'. Em seguida, seleciona-se a conta de origem e posteriormente a conta de destino, ambas utilizando o botão esquerdo do mouse.

Figura 9. Transação conectando Contas

4.2.3. Selecionando Contas

Após a criação das contas, é necessário definir suas características específicas. Para atender essa necessidade, o software disponibiliza um editor de propriedades responsável pela configuração e manutenção dos elementos que compõem cada conta. Por meio dessa funcionalidade, o Analista de Processos pode alterar o nome da conta, modificar variáveis existentes ou criar variáveis necessárias para representar aspectos particulares adicionais do processo modelado.

Além da definição do nome, cada variável possui propriedades próprias, incluindo seu tipo de dado, valor inicial e restrições de domínio, como a permissão ou proibição de valores negativos.

A possibilidade de personalizar variáveis permite que o mesmo mecanismo de modelagem seja utilizado em diferentes cenários empresariais, reforçando a proposta de generalização do padrão Conta/Transação.

O acesso ao editor é realizado através do modo de seleção. Após ativar o botão 'Modo Seleção', o usuário pode clicar com o botão direito sobre uma conta e selecionar a opção 'Editar' para abrir a interface de configuração.

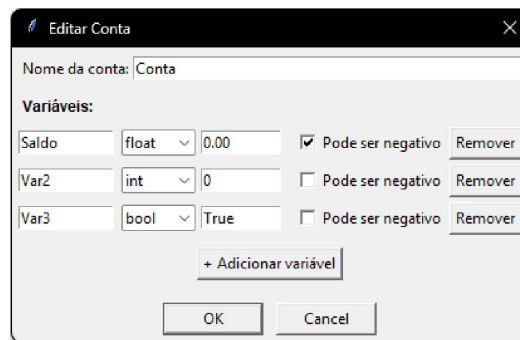


Figura 10. Editor de Conta

4.2.4. Executando Transações

A execução de transações constitui um dos mecanismos centrais de funcionamento do protótipo. É por meio dela que o estado do modelo evolui ao longo do tempo, executando o comportamento do processo real que está sendo representado.

Quando uma transação é executada, o sistema realiza a movimentação da quantidade especificada da variável associada, removendo esse valor da conta de origem e adicionando-o à conta de destino. Dessa forma, a execução das transações altera continuamente os saldos das contas e produz o fluxo de recursos que caracteriza o processo modelado.

Do ponto de vista conceitual, a execução de uma transação corresponde à ocorrência de um evento dentro do processo de negócio. Em um ambiente industrial, por exemplo, ela pode representar a transferência de matéria-prima entre setores. Em um ambiente de suporte técnico, pode representar a solução, transferência ou inatividade de um ticket.

Uma particularidade importante da implementação é o suporte a transações cuja origem e destino são a mesma conta. Embora aparentemente redundantes, essas transações possibilitam representar mecanismos de geração ou consumo contínuo de recursos.

Esse recurso se mostra especialmente útil na modelagem de cenários inspirados

no problema produtor-consumidor, nos quais determinadas contas atuam como fontes geradoras de recursos enquanto outras representam elementos consumidores desses mesmos recursos.

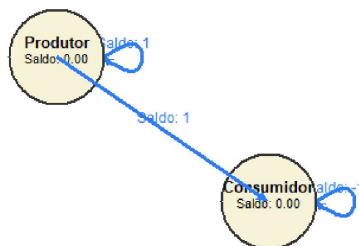


Figura 11. Exemplo Produtor – Consumidor

4.2.5. Persistência do Modelo

Uma característica importante do software é a capacidade de persistir os modelos criados pelo usuário. Essa funcionalidade permite interromper e retomar sessões de modelagem sem a necessidade de reconstruir manualmente toda a estrutura do processo.

Para implementar esse mecanismo, o sistema percorre todas as contas e transações presentes no modelo e converte suas informações para o formato JSON. A escolha desse formato deve-se à sua simplicidade, ampla compatibilidade e facilidade de integração com outras ferramentas.

Além de preservar a modelagem realizada pelo usuário, o arquivo gerado também funciona como uma representação intermediária utilizada pelos demais módulos do sistema. Dessa forma, tanto a simulação quanto futuras funcionalidades podem operar diretamente sobre o modelo serializado.

A funcionalidade é disponibilizada através dos botões ‘Salvar Modelo’ e ‘Carregar Modelo’, responsáveis respectivamente pela exportação e importação dos dados da modelagem.

4.2.6. Simulação do Modelo (Detecção de Gargalos)

Uma das funcionalidades centrais do protótipo é a capacidade de simular o comportamento de um processo modelado utilizando o padrão Conta/Transação. Essa funcionalidade permite observar a evolução dinâmica do sistema ao longo do tempo e identificar possíveis gargalos operacionais antes que estes ocorram em um ambiente real.

Para realizar essa simulação, o software utiliza a biblioteca SimPy, um *framework* de simulação de eventos discretos desenvolvido para a linguagem Python. Nessa abordagem, as transações definidas pelo usuário são tratadas como eventos que tentam executar

periodicamente, movimentando recursos entre contas de acordo com as regras estabelecidas na modelagem.

Durante a execução da simulação, cada transação verifica continuamente se existem condições suficientes para sua realização. Caso os recursos necessários estejam disponíveis na conta de origem, a movimentação é efetuada e os saldos das contas envolvidas são atualizados. Caso contrário, a transação permanece temporariamente bloqueada até que os recursos necessários estejam disponíveis novamente.

Essa dinâmica permite reproduzir comportamentos observados em processos reais. Quando uma determinada etapa do processo recebe recursos em uma velocidade superior à sua capacidade de processamento ou transferência, ocorre um acúmulo progressivo de saldo em uma ou mais contas. Esse comportamento caracteriza a existência de um possível gargalo operacional.

No contexto do modelo proposto, a identificação de gargalos é realizada através da observação do comportamento das contas e transações ao longo da simulação. Contas que apresentam crescimento contínuo de saldo podem indicar etapas congestionadas do processo, enquanto transações que executam com baixa frequência podem indicar restrições operacionais ou insuficiência de recursos. Essas informações são registradas durante a execução da simulação para processamento posterior pelo módulo de análise.

Uma vantagem da utilização do padrão Conta/Transação é que a detecção de gargalos independe do domínio específico do processo modelado. Como todos os recursos são representados por contas e todas as operações são representadas por transações, o mecanismo de simulação pode utilizar a mesma lógica para analisar processos produtivos, logísticos, administrativos ou financeiros.

Ao término da execução, o sistema gera um conjunto de métricas que descrevem o comportamento do modelo durante o período simulado. Essas informações incluem dados como a quantidade de transações saídas de determinada conta, e quantas foram bloqueadas por falta de saldo, por exemplo. Os resultados podem então ser exportados para um arquivo JSON, servindo como entrada para o módulo de análise apresentado na seção seguinte.

Para executar uma simulação, o usuário deve possuir um modelo carregado no *canvas* ou um arquivo de modelagem previamente salvo. A interface de simulação pode ser acessada através do botão “Simulação”, onde são definidos parâmetros como o tempo total de simulação e o intervalo entre tentativas de execução das transações.

Após o início da execução, os eventos simulados são processados pelo SimPy e os resultados são apresentados na parte inferior da interface. Ao término do processo, o usuário pode salvar os resultados gerados por meio do botão “Salvar Resultados”, produzindo um arquivo JSON que poderá ser posteriormente utilizado pelo módulo de análise de gargalos.

Diferentemente de ferramentas tradicionais de Business Intelligence, que normalmente analisam apenas dados históricos já produzidos pela organização, a abordagem proposta permite simular cenários hipotéticos antes de sua implementação. Dessa forma, gestores podem avaliar o impacto de alterações em processos, capacidades produtivas ou fluxos de recursos sem a necessidade de modificar inicialmente o ambiente real.

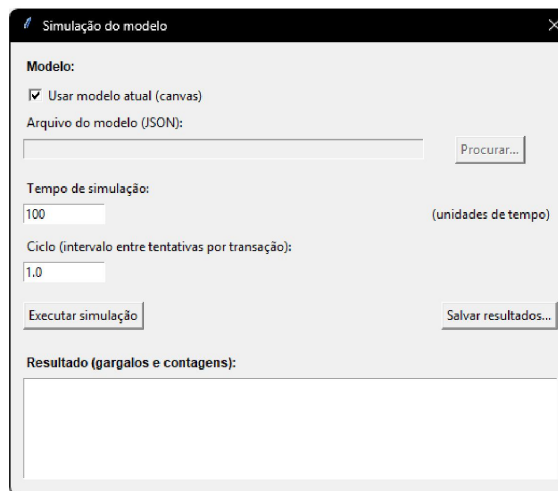


Figura 12. UI de Simulação

4.2.7. Análise da Simulação

A etapa de análise da simulação representa o componente de Business Intelligence do protótipo. Após a execução da simulação, o sistema utiliza os dados produzidos para identificar padrões que possam indicar a existência de gargalos operacionais, acúmulo excessivo de recursos em determinadas contas ou desequilíbrios entre produção e consumo.

O processo de análise é realizado sobre um arquivo de resultados previamente gerado pelo simulador. Com base nessas informações, o sistema gera recomendações destinadas ao Analista de Processos. Essas recomendações não alteram automaticamente a modelagem, mas servem como suporte à tomada de decisão, permitindo que o usuário refine o modelo e execute novas simulações até obter um comportamento considerado satisfatório.

O acesso à interface de análise é realizado por meio do botão ‘Análise da Simulação’. Nessa interface, o usuário seleciona um arquivo de resultados previamente salvo e aciona o comando ‘Analisar Resultados’. Após o processamento das informações, as recomendações identificadas são apresentadas na tela para avaliação do analista.

Figura 13. UI Análise de Simulação

Como já introduzido, um gargalo é aquele recurso cuja capacidade é igual ou menor que a demanda colocada nele [Goldratt and Cox 1984]. Na análise realizada na imagem, do exemplo da padaria que será descrito posteriormente, temos um gargalo evidente no elemento “Forno”. Ele possui a menor capacidade do sistema, e todos os pães produzidos precisam obrigatoriamente passar por ele antes de irem ao balcão, onde serão retirados do sistema. Portanto, neste exemplo, conforme detectado pelo software, o forno

é um gargalo a ser corrigido. Isso poderia ser feito, por exemplo, ao aumentar a capacidade de produção, como sugerido, o que se traduziria na prática como “adicionar mais fornos” para paralelizar o processo e não depender de apenas uma unidade.

Heurísticas de Avaliação. A Tabela 3 apresenta as métricas utilizadas e as recomendações sugeridas pelo software.

Tabela 3. Métricas utilizadas e recomendações sugeridas pelo software

Métrica observada	Interpretação	Recomendação sugerida
Bloqueios > Execuções	Produção insuficiente ou restrição de capacidade	Aumentar capacidade de produção ou estoque da origem
Valor transferido elevado	Fluxo excessivamente concentrado	Reduzir quantidade transferida ou aumentar frequência das operações
Execuções = 0	Ausência de recursos disponíveis na origem	Verificar etapas produtoras anteriores
Saldo inicial/final igual a zero durante toda a simulação	Processo sem recursos para iniciar	Aumentar estoque inicial ou criar produção recursiva
Nenhuma condição anterior	Possível competição por recursos compartilhados	Avaliar priorização entre transações

Esta abordagem não utiliza técnicas de aprendizado de máquina ou métodos estatísticos para identificação de gargalos. As recomendações são produzidas por meio de heurísticas determinísticas baseadas em regras, construídas a partir da interpretação das métricas coletadas durante a simulação. Embora essa estratégia tenha se mostrado adequada para a prova de conceito desenvolvida neste trabalho, processos empresariais mais complexos poderão exigir métricas adicionais e algoritmos de análise mais sofisticados em trabalhos futuros.

4.2.8. Exemplo Padaria

Para demonstrar a utilização do protótipo, foi desenvolvida uma modelagem simplificada de uma padaria. O exemplo busca representar um processo produtivo relativamente simples, mas suficiente para ilustrar a identificação de gargalos utilizando o padrão Conta/Transação.

Nesse modelo, o Padeiro atua como produtor de pães, gerando continuamente unidades do produto. Os pães produzidos são encaminhados ao Forno, responsável pelo processamento necessário antes da disponibilização para venda. Após essa etapa, os produtos são transferidos para o Estoque, que atua como um *buffer* intermediário, e finalmente chegam ao Balcão, onde ocorre o consumo da produção.

A presença do Forno e do Estoque permite reproduzir situações típicas de gargalos operacionais. Caso a capacidade do Forno seja inferior à taxa de produção do Padeiro, por exemplo, ocorrerá o acúmulo progressivo de produtos na etapa anterior do processo. Da mesma forma, limitações nas movimentações envolvendo o Estoque podem provocar

atrasos no abastecimento do Balcão. Durante a simulação, essas situações podem ser identificadas pelo aumento contínuo dos saldos de determinadas contas ou pela baixa frequência de execução de transações específicas.

Embora simplificado, o cenário da padaria demonstra a capacidade do software de representar processos produtivos utilizando o padrão Conta/Transação e de utilizar essa representação como base para a identificação de gargalos. Dessa forma, o exemplo serve como uma validação inicial da proposta de utilizar uma modelagem transacional genérica como fundamento para análises de Business Intelligence.

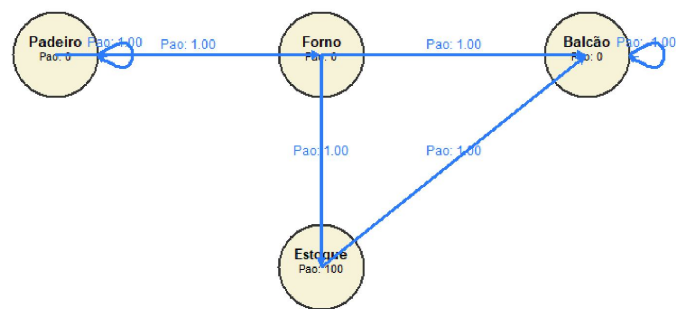


Figura 14. Exemplo Padaria

4.3. Discussão dos Resultados

A Tabela 4 sintetiza a comparação entre o padrão Conta/Transação e o modelo relacional tradicional.

Tabela 4. Comparação entre Conta/Transação e o Modelo Relacional Tradicional

Conta/Transação	Modelo Relacional Tradicional
Contas representam estados	Diversas tabelas específicas
Transações representam fluxo	Chaves estrangeiras
Motor único	Regras por domínio
Mesmo algoritmo	Consultas específicas

No modelo desenvolvido neste trabalho, o processo da padaria é representado por quatro contas (Padeiro, Forno, Estoque e Balcão) conectadas por transações que movimentam recursos entre essas etapas. O estado do sistema é obtido diretamente pelos saldos das contas, enquanto o comportamento é determinado pela execução das transações durante a simulação.

Em uma modelagem relacional tradicional, por outro lado, o mesmo processo seria representado por entidades específicas do domínio, como Pao, Lote, Forno, Estoque e Venda, além de relacionamentos que descrevem a etapa atual de cada item. A determinação de quantos produtos estão em cada fase dependeria de várias consultas ao

banco de dados, frequentemente envolvendo múltiplas tabelas e regras específicas daquele processo.

A principal vantagem do padrão Conta/Transação, no contexto deste trabalho, não é substituir a modelagem relacional para armazenamento de dados, mas fornecer uma representação conceitual uniforme sobre a qual o mesmo mecanismo de simulação e análise pode operar. Dessa forma, o algoritmo responsável por detectar gargalos permanece inalterado mesmo quando o domínio modelado muda, bastando que novos processos sejam descritos em termos de contas e transações.

A implementação do protótipo, portanto, permitiu avaliar, de forma prática, a aplicação do padrão Conta/Transação como base para um sistema de Business Intelligence orientado à identificação de gargalos. Os cenários modelados demonstraram que processos com fluxo claro de recursos entre etapas podem ser representados de maneira consistente por meio de contas e transações, reduzindo a necessidade de estruturas específicas para cada domínio de negócio.

Observou-se que a separação entre modelagem, simulação e análise facilitou o processo iterativo de refinamento do modelo. O usuário pode construir o processo, executar a simulação, analisar os resultados e ajustar a modelagem sucessivamente, aproximando-se do ciclo típico de melhoria contínua encontrado em ambientes de gestão operacional.

4.3.1. Viabilidade da Modelagem Conta/Transação

O desenvolvimento indicou que o padrão Conta/Transação mostra-se adequado para representar fluxos de produção, armazenamento e consumo de recursos, como ilustrado pelo exemplo da padaria, com a produção, processamento, armazenamento e venda modelados utilizando apenas contas e movimentações entre elas.

Essa uniformidade estrutural simplifica a construção do motor de simulação, já que diferentes tipos de processo compartilham a mesma representação básica. Em vez de criar estruturas específicas para cada área da empresa, o sistema opera sobre um conjunto reduzido de conceitos fundamentais.

Entretanto, observou-se que processos que dependem fortemente de lógica condicional, regras temporais complexas ou múltiplos recursos simultâneos podem exigir extensões adicionais ao modelo atual.

4.3.2. Comportamento da Simulação e Detecção de Gargalos

A utilização do SimPy permitiu executar o modelo ao longo do tempo e observar o comportamento dinâmico das transações. Nos cenários avaliados, os gargalos manifestaram-se principalmente como acúmulo progressivo de saldo em determinadas contas ou redução significativa na frequência de execução de transações específicas.

No exemplo da padaria, quando a capacidade do Forno era configurada abaixo da taxa de produção do Padeiro, o sistema passou a acumular recursos nas etapas anteriores do processo, indicando a formação de um gargalo operacional.

Esse comportamento está alinhado ao conceito de gargalo conforme definido pela Teoria das Restrições [Goldratt and Cox 1984], na qual a etapa de menor capacidade de um sistema determina o *throughput* global do processo. A modelagem Conta/Transação, ao representar todos os recursos como saldos mensuráveis, viabiliza a identificação automática dessa condição sem que o analista precise inspecionar manualmente cada etapa.

A análise posterior dos resultados permitiu identificar essas situações e apresentar recomendações ao usuário, demonstrando a viabilidade da integração entre modelagem transacional, simulação e análise de BI.

4.3.3. Ameaças à Validade

A principal ameaça à validade externa deste trabalho está relacionada ao ambiente de avaliação adotado. A exemplificação do protótipo foi realizada utilizando um cenário simplificado de uma padaria, modelado especificamente para demonstrar o funcionamento da abordagem proposta. Embora esse exemplo seja suficiente para verificar a correta integração entre a modelagem, a simulação e o mecanismo de análise, ele não permite afirmar que o mesmo desempenho será observado em processos empresariais mais complexos, envolvendo múltiplas restrições, regras condicionais ou grande quantidade de recursos simultâneos.

Outra limitação decorre da utilização de dados sintéticos produzidos durante a própria execução da simulação. Diferentemente de sistemas corporativos de Business Intelligence, que operam sobre registros históricos provenientes de bancos de dados organizacionais, o protótipo trabalha com eventos gerados pelo modelo construído pelo usuário. Dessa forma, aspectos comuns em bases reais, como dados incompletos, inconsistências, registros duplicados, falhas de integração entre sistemas ou informações ausentes, não foram considerados durante a avaliação. Consequentemente, ainda não é possível afirmar como o mecanismo de modelagem e análise se comportaria diante dessas condições.

Também deve ser considerada a escalabilidade da solução. O protótipo foi concebido como uma prova de conceito e avaliado em modelos de pequena dimensão. Embora a representação baseada no padrão Conta/Transação seja independente do domínio de aplicação, não foram realizados experimentos para avaliar seu desempenho em cenários contendo centenas de contas, milhares de transações ou longos períodos de simulação. Assim, questões relacionadas ao consumo de memória, tempo de processamento e capacidade de análise permanecem como tópicos para investigação futura.

Outra ameaça refere-se ao mecanismo de identificação de gargalos. As recomendações produzidas pelo sistema são baseadas em heurísticas determinísticas simples construídas a partir de métricas como número de execuções, quantidade de bloqueios e estado das contas ao término da simulação. Embora essas métricas estejam alinhadas ao conceito de restrição apresentado pela Teoria das Restrições [Goldratt and Cox 1984], não foi realizada uma comparação sistemática entre os diagnósticos produzidos pelo protótipo e avaliações realizadas por especialistas em processos ou por ferramentas comerciais de mineração de processos. Dessa forma, os resultados obtidos devem ser interpretados como uma validação inicial da abordagem proposta, e não como uma comprovação definitiva da eficácia da heurística.

Por fim, a hipótese de que o padrão Conta/Transação possa servir como uma representação conceitual aplicável a diferentes domínios também deve ser interpretada com cautela. O trabalho demonstrou que essa abordagem é adequada para representar um processo produtivo simplificado, porém sua aplicação a áreas como logística, recursos humanos, saúde ou processos financeiros ainda depende de validações adicionais. Assim, os resultados apresentados indicam o potencial de generalização da proposta, mas não permitem afirmar uma aplicabilidade ampla.

4.3.4. Limitações do Protótipo

Apesar dos resultados positivos, o protótipo possui limitações importantes. A interface gráfica foi desenvolvida como prova de conceito e ainda não oferece mecanismos avançados de edição, validação ou visualização de grandes modelos. Além disso, o armazenamento em JSON é adequado para experimentação, mas não substitui uma solução de persistência mais robusta para ambientes reais.

Outra limitação relevante é que o mecanismo de análise atualmente utiliza heurísticas simples para identificar gargalos. Embora suficientes para os cenários testados, essas heurísticas podem não capturar comportamentos mais complexos presentes em processos empresariais de grande escala.

4.3.5. Possibilidades de Evolução

Os resultados obtidos sugerem várias possibilidades de evolução. Entre elas, destacam-se a integração com bancos de dados relacionais, a importação automática de dados de sistemas corporativos, e o uso de algoritmos de otimização para ajuste de parâmetros.

Também seria possível expandir o modelo para incluir restrições temporais, prioridades entre transações e recursos compartilhados, aproximando o protótipo de cenários industriais mais complexos.

4.3.6. Criação em Tempo de Execução e o Problema das Subclasses

Uma das propriedades mais relevantes do padrão Conta/Transação é a possibilidade de criar contas e transações em tempo de execução, sem necessidade de alterações no código ou na estrutura do sistema. No contexto do protótipo desenvolvido, isso se manifesta na capacidade do Analista de Processos de adicionar novas contas ao modelo e conectá-las com transações durante a própria sessão de trabalho, sem intervenção técnica.

Essa característica levanta, no entanto, uma questão de design relevante. Se cada domínio de negócio, seja ele financeiro, logístico ou de recursos humanos, possui contas com comportamentos específicos, seria necessário criar subclasses de Conta para cada tipo? Uma conta financeira precisa lidar com moedas e câmbio; uma conta de estoque precisa tratar unidades de medida distintas; uma conta de RH precisa registrar colaboradores individualmente. Se cada especificidade gerar uma subclasse, o modelo perde sua aplicabilidade generalizada e retorna à fragmentação que o padrão se propõe a evitar.

O protótipo desenvolvido neste trabalho soluciona essa questão quando, em vez de especializar Conta por herança, o sistema utiliza variáveis configuráveis em tempo de execução. Cada conta pode receber qualquer número de variáveis com nome, tipo de dado, valor inicial e restrições definidas pelo próprio usuário. Uma conta financeira pode ter uma variável `saldo` do tipo `float` com permissão de valores negativos; uma conta de estoque pode ter uma variável `quantidade` do tipo `int` sem permissão de negativos; uma conta de RH pode ter variáveis `headcount` e `disponível` do tipo `int` e `bool`, respectivamente. O mecanismo de movimentação e detecção de gargalos opera sobre qualquer uma dessas variáveis sem distinguir seu significado de domínio.

A consequência prática é que o sistema permanece extensível sem que seu núcleo precise ser modificado: adicionar suporte a um novo tipo de processo não requer conhecimento de programação, requer apenas que o analista configure novas variáveis nas contas existentes ou crie novas contas configuradas para o domínio desejado.

Reconhece-se, entretanto, que essa solução tem limites. Processos que demandam lógica condicional complexa entre diversas variáveis, como regras de negócio que dependem de múltiplas variáveis simultaneamente, podem exigir extensões ao modelo de variáveis atual.

5. Conclusão e Trabalhos Futuros

Este trabalho propôs o desenvolvimento de um protótipo de software de Business Intelligence fundamentado no padrão de modelagem conceitual de Conta/Transação. A motivação central foi a dificuldade encontrada por gestores em identificar gargalos operacionais devido à falta de padronização nos dados de processos complexos.

Ao longo do desenvolvimento, demonstrou-se que a aplicação do padrão Conta/Transação [Wazlawick 2015] funciona como um elemento integrador. Os resultados obtidos sugerem que o padrão Conta/Transação tem potencial de atuar como estrutura integradora de dados em sistemas de BI. Para validar de forma mais ampla a aplicabilidade do padrão, trabalhos futuros deverão testar o modelo em domínios distintos, como logística, recursos humanos e finanças.

O protótipo desenvolvido em Python validou a viabilidade de criar uma interface visual capaz de mapear esses processos de forma intuitiva. A integração entre a modelagem de sistemas e as ferramentas de análise de dados mostrou-se eficaz para a detecção de pontos de congestionamento, uma vez que o padrão transacional fornece, intrinsecamente, as métricas de entrada, saída e acúmulo necessárias para o cálculo de gargalos.

A união entre BI e o padrão Conta/Transação, portanto, além de simplificar a arquitetura do software, demonstra potencial para ser aplicada em diferentes áreas. Essa abordagem propõe um caminho para mitigar a desconexão apontada inicialmente entre a modelagem de sistemas de informação e as necessidades estratégicas de análise de negócio.

Como sugestões para trabalhos futuros, aponta-se a necessidade de testar o protótipo em cenários de alta volumetria de dados e a implementação de algoritmos de inteligência artificial para a previsão preditiva de gargalos baseada no histórico de transações acumulado, além da expansão dos domínios de teste, como já mencionado.

Referências

- Azizov, J. (2021). Expensetracker – personal finance manager. Bachelor's thesis, České vysoké učení technické v Praze.
- Ferreira, J. E. d. S. F. (2022). Implementation of a business intelligence solution: a case study of a workforce and staffing solutions company. Master's thesis, Universidade do Porto. Disponível em: <https://hdl.handle.net/10216/144340>.
- Goldratt, E. M. and Cox, J. (1984). *A Meta: um processo de melhoria contínua*. Nobel, São Paulo.
- IBM (2024). Cognos analytics. Disponível em: <https://www.ibm.com/products/cognos-analytics>. Acesso em: 8 de julho de 2024.
- Microsoft (2024). Power BI. Disponível em: <https://www.microsoft.com/pt-br/power-platform/products/power-bi>. Acesso em: 8 de julho de 2024.
- Olszak, C. M. and Ziemba, E. (2004). Business intelligence systems as a new generation of decision support systems. In *Proceedings of PISTA 2004, International Conference on Politics and Information Systems: Technologies and Applications*.
- Reinschmidt, J. and Francoise, A. (2000). Business intelligence certification guide. Technical report, IBM, International Technical Support Organization.
- Sezões, C., Oliveira, J., and Baptista, M. (2006). *Business Intelligence*. SPI – Sociedade Portuguesa de Inovação, Porto.
- Silva, R. P. e. (2007). *UML 2: modelagem orientada a objetos*. Visual Books, Florianópolis. ISBN 8587393588.
- Team SimPy (2024). Simpy: Discrete-event simulation for Python. Versão 4.1.2. Disponível em: <https://simpy.readthedocs.io>. Acesso em: 15 de junho de 2026.
- Valenti, G. (2022). Business intelligence: the implementation of a BI solution to a case study. Master's thesis, Politecnico di Torino, Torino. Corso di laurea magistrale in Ingegneria Gestionale (Engineering and Management). Relatore: Luca Mastrogiacomio.
- Wazlawick, R. S. (2015). *Análise e design orientados a objetos para sistemas de informação: modelagem com UML, OCL e IFML*. Elsevier, Rio de Janeiro, 3 edition. ISBN 9788535279849.