

# Multivocal Literature Mapping on DSLs and Tools for Entity-Relationship Modeling

Jonnathan Riquelmo<sup>1</sup>, Maicon Bernardino<sup>1</sup>, Fábio Basso<sup>1</sup>, Elder Rodrigues<sup>1</sup>

<sup>1</sup>Laboratory of Empirical Studies in Software Engineering (LESSE),  
Software Engineering, Federal University of Pampa (UNIPAMPA)

{jonnathan.riquelmo, fabiopbasso, eldermr}@gmail.com, bernardino@acm.org

**Abstract.** *The variety of database system technologies that became available in recent years makes difficult the selection of tools for modeling entity-relationship (ER). In this sense, we contribute with an overview of the recent innovations through a systematic literature mapping complemented by research in the gray literature. This paper scopes ten (10) primary studies focused on Domain-Specific Languages (DSL) and identifies fifty-five (55) tools already applied in industry and academia for ER modeling at the conceptual, logical, and physical level. Hence, as a significant increment to existing mapping studies, this presents the state of the art and practice for ER modeling, including its characterization and research gaps.*

**Resumo.** *A variedade de tecnologias de sistemas de banco de dados que se tornou disponível nos últimos anos dificulta a seleção de ferramentas para modelagem de entidade-relacionamento (ER). Nesse sentido, contribui-se com um levantamento de inovações recentes por meio de um mapeamento sistemático complementado por uma pesquisa na literatura cinza. Este trabalho abrange um conjunto final de 10 estudos primários focados em Linguagens Específicas de Domínio (DSL) e identifica 55 ferramentas já aplicadas na indústria e academia para modelagem ER em nível conceitual, lógico e físico. Assim, este estudo apresenta o estado da arte e da prática para modelagem ER, incluindo sua caracterização e lacunas de pesquisa.*

## 1. Introdução

No processo de desenvolvimento de software, os modelos de dados podem sofrer diferentes transformações. A abordagem de modelagem de três níveis para projeto de Banco de Dados (BDs) [ANSI 1975] sugere separar modelos de dados em diferentes graus de abstração, conhecidos como modelos conceituais, lógicos e físicos. A modelagem conceitual de sistemas de bancos de dados é a que requer o mais alto nível de abstração, sendo que a técnica mais difundida para sua elaboração é a proposta de modelagem conceitual de dados concebida por [Chen 1976]. Essa proposta, influenciada pela abordagem de três níveis, foi tão bem aceita que passou a ser considerada uma referência definitiva para modelagem de BD [Cougo 2013].

A modelagem de BD é uma parte importante do ciclo de vida de desenvolvimento de software, na qual geralmente é suportado por alguma DSL (*Domain-Specific Language*) [Voelter 2009]. De acordo com [Van Deursen et al. 2000], uma DSL é uma linguagem de programação ou linguagem de especificação executável que

oferece, por meio de notações e abstrações apropriadas, poder expressivo focado e geralmente restrito a um domínio de problema específico. Como outras linguagens, uma DSL deve apresentar um conjunto de sentenças bem definidas por uma sintaxe e semântica em si. Para [Fowler 2010] uma DSL é definida como uma linguagem de programação de computadores com expressividade limitada e focada em um domínio particular. Além das abordagens de *code-first*, a adoção de DSLs em processos de software promove um aumento significativo na produtividade, qualidade, facilidade de uso e flexibilidade [Mernik et al. 2005, Vara et al. 2014]. A maior vantagem no uso de DSLs é a abstração, na qual o conhecimento necessário para o desenvolvimento é elevado a um nível menos complexo. Desta forma, especialistas em domínio podem entender, validar e modificar o código por meio de transformações e refinamentos de modelos [Van Deursen et al. 2000].

De acordo com [Mernik et al. 2005], as DSLs podem ser classificadas em três dimensões: origem (interna ou externa), aparência (textual, gráfica, tabular ou simbólica) e implementação (modo de execução). Em geral, a principal consideração para a criação de uma DSL deve ser sua origem, uma vez que cada abordagem tem vantagens e desvantagens específicas [Fowler 2010]. Uma DSL interna é projetada a partir das regras gramaticais de uma linguagem existente, que pode ser uma linguagem de propósito geral (GPL) ou outra DSL. Uma DSL externa é uma linguagem com sintaxe distinta e depende de sua própria infraestrutura para análise léxica, sintática, semântica, interpretação, compilação, otimização e geração de código [Mernik et al. 2005]. Quanto a execução, conceitualmente, essa dimensão engloba desde a especificação de notações gráficas e textuais de linguagens aplicadas sobre domínios específicos até soluções implementadas de fato.

O objetivo deste trabalho é mapear DSLs existentes para modelagem de BD em qualquer um dos três níveis de modelo. Para atender a essa finalidade, um Mapeamento Multivocal da Literatura (*Multivocal Literature Mapping - MLM*) foi realizado para encontrar estudos primários e ferramentas que apresentem abordagens, métodos ou técnicas para a criação de modelos de BD.

Este artigo está organizado da seguinte maneira. A Seção 2 apresenta a discussão dos trabalhos relacionados. O protocolo usado é detalhado na Seção 3. A sua execução é descrita na Seção 4, enquanto que os resultados obtidos são analisados e discutidos na Seção 5. A Seção 6 descreve as ameaças à validade do estudo. Por fim, a Seção 7 apresenta a conclusão e trabalhos futuro.

## 2. Trabalhos Relacionados

Antes de realizar esta pesquisa foi executada uma busca *ad-hoc*<sup>1</sup> na literatura por trabalhos que propusessem pesquisas relacionadas a abordagens para modelagem entidade-relacionamento (ER) em BD, com foco em pesquisas e mapeamentos.

[Gregersen and Jensen 1999] analisaram as propriedades de projetos usadas para especificar atributos temporais em abordagens de modelagem ER. O objetivo do estudo é apresentar os novos conceitos usados para facilitar a modelagem temporal

---

<sup>1</sup>Pesquisa *Ad-Hoc* é uma pesquisa personalizada, usando um método feito sob medida, e estruturada para atender necessidades específicas sem necessariamente um alto grau de rigor científico.

em 19 diferentes propostas. Estas propostas baseiam-se em modelos, nomeadamente: ER, EER, OO e ERC+. Além disso uma avaliação de dez (10) modelos contra várias propriedades de projeto para verificar sua completude foi realizada. Os autores concluem que, embora os modelos coletivamente tenham uma cobertura de projeto adequada, nenhum deles satisfaz todos os aspectos da modelagem avaliada.

[Siau 2012] realizou uma revisão da literatura sobre abordagens relacionais e semânticas de modelagem ER. Ele mostra que a maioria dos estudos aponta que os modelos semânticos são melhores que os modelos relacionais e que, em menor grau, outros estudos fazem exatamente o contrário. Além disso, discute alguns trabalhos que afirmam não haver diferenças significativas entre as duas abordagens estudadas.

Pode-se mencionar outros três importantes mapeamentos, não sistemáticos, de abordagens ER. A revisão bibliográfica não estruturada apresentada em [Ma and Yan 2010] pesquisa contribuições para informações incertas na modelagem conceitual, abrangendo objetos complexos e dados imprecisos. Em [Lorenz et al. 2016], os autores apresentam um mapeamento de literatura não estruturada incluindo as melhores práticas industriais e acadêmicas em projetos ER. Em [Torres et al. 2017], os autores descrevem um estudo de mapeamento para caracterizar nove (9) plataformas que suportam ORMS (*Object-Relational Mapping Solutions*), incluindo exemplificações nas notações UML e Barker. Estes são estudos complementares e não se sobrepõem às questões de pesquisa propostas neste MLM.

### 3. Mapeamento Multivocal de Literatura

Uma *Multivocal Literature Mapping - MLM* é uma forma de Mapeamento Sistemático de Literatura (*Systematic Literature Mapping - SLM*) que inclui a literatura cinza. Os MLMs são úteis para pesquisadores e para profissionais pois fornecem visões abrangentes sobre o estado da arte e da prática em uma determinada área. Neste estudo conduziu-se um MLM<sup>2</sup> utilizando o processo de SLM definido por [Petersen et al. 2008] e as diretrizes propostas por *et al.* [Garousi, V. and Felderer, M. and Mäntylä 2019] para pesquisa na literatura cinza.

Para tanto, foram definidas as seguintes Questões de Pesquisa (QPs): **QP1.** Qual o estado da arte do desenvolvimento de DSLs para transformação de modelos ER? **QP1.1.** Quais são as metodologias, técnicas e propostas de refinamento (desenho automatizado) baseado em modelos de dados? **QP1.2.** Qual é o ferramental usado como apoio ao desenvolvimento dessas DSLs? **QP2** Quais os métodos de avaliação utilizados nos estudos primários? **QP2.1.** Quais os pontos positivos e negativos observados na execução dos estudos? **QP2.2.** Quais desafios são apontados após execução dos estudos empíricos? **QP3.** Quais são as ferramentas para modelagem conceitual de bancos de dados? **QP3.1.** Quais são as notações ou linguagens que estas ferramentas usam? **QP3.2.** Quais são os níveis de modelagem (Conceitual, Lógico, Físico) que estas ferramentas suportam?

Para o SLM foi necessário realizar acesso em bibliotecas digitais (*Digital Library - DL*) para obter estudos científicos. Para a pesquisa conduzida as DLs consultadas deveriam contemplar três requisitos obrigatórios: possuir um motor de

---

<sup>2</sup>Dados do pacote experimental disponíveis no repositório Zenodo DOI: 10.5281/zenodo.5724135

busca baseado na Web; ter um mecanismo de buscar por palavras-chave, e; abranger estudos primários da área da Ciência da Computação. Sendo assim, realizou-se a busca em cinco DLs: (i) ACM DL; (ii) IEEE Xplore; (iii) ScienceDirect; (iv) Scopus; (v) SpringerLink. Após, foram estabelecidos os termos e sinônimos mais expressivos para o objeto de pesquisa. A combinação gerou a *string* genérica da Figura 1.

(DSL OR Domain Specific Language OR Domain-Specific Language OR Domain-Specific-Language OR DSML OR Domain Specific Modeling Language OR Domain-Specific Modeling Language OR Domain-Specific-Modeling-Language OR Query Language) AND (ER OR Entity-Relationship OR Enhanced Entity-Relationship OR Extended Entity-Relationship OR Database)

**Figura 1. Search string.**

Foram elaborados critérios de seleção para os estudos primários, com o intuito de se incluir os trabalhos mais relevantes. Os critérios foram divididos em: (i) *Crítérios de Inclusão (CI)*: (**CI1.**) Estudo que propõe alguma técnica, método, abordagem ou ferramenta para a representação e transformação de modelos de BD utilizando DSL. (ii) *Crítérios de Exclusão (CE)*: (**CE1.**) Estudo com menos de quatro páginas; (**CE2.**) Estudo que não esteja escrito em inglês; (**CE3.**) Estudo duplicado; (**CE4.**) Estudo que não fornece acesso completo ao seu conteúdo; (**CE5.**) Estudo que não atende o CI1.

Além disso, foi estabelecido um conjunto de sete (7) de critérios para a avaliação de qualidade (CQs). O objetivo da avaliação de qualidade foi qualificar a relevância dos estudos e auxiliar na interpretação dos resultados, bem como limitar o viés do SLM [Dybå and Dingsøyrr 2008]. Complementarmente foi atribuído uma pontuação aos CQs: **T:Total**, considerando de forma integral o CQ avaliado; **P:Parcial**, considerando parcialmente o CQ avaliado e conforme seu peso; **N:Negativo**, não contempla de forma alguma o CQ avaliado. A pontuação máxima possível é dez (10) e a mínima zero (0). Cada CQ teve um peso específico (1 → 1.5 → 2) dependendo da sua importância. A Tabela 1 apresenta a lista dos CQs e seus respectivos pesos.

**Tabela 1. Critérios de Avaliação de Qualidade**

CQs	Descrição	Peso
CQ1.	O estudo apresenta alguma contribuição para a área de modelagem de BD?	1.5
CQ2.	O estudo apresenta técnicas ou propostas de refinamento baseado em modelos de dados?	1.5
CQ3.	O estudo apresenta alguma forma de avaliação empírica?	1.5
CQ4.	O estudo apresenta as características do processo de criação da DSL?	1.5
CQ5.	O estudo caracteriza as atividades de transformação do modelo para diferentes tecnologias de BDs?	2.0
CQ6.	O estudo apresenta pontos positivos e negativos observados em sua execução?	1.0
CQ7.	O estudo aponta desafios decorrentes da sua execução?	1.0

Uma vez realizada a avaliação de qualidade é necessário executar a etapa de extração dos dados para a interpretação dos resultados. Desta forma, foi criado um formulário de extração de dados composto por uma série de itens, como a origem da solução, o ano da publicação, o objetivo, a avaliação da proposta, os pontos positivos e negativos observados, entre outros dados.

Utilizou-se o motor de busca da Google para a pesquisa por ferramentas de modelagem de bancos de dados na literatura cinza. Para isto, combinações de *keywords* foram definidas a partir da *string* de busca e adequando-as para o contexto

da busca na *Web*: (i) ERD Modelling Tool; (ii) ERD Design Tool; (iii) Conceptual Design of Database; (iv) Conceptual Modelling of Database; (v) Conceptual Modelling; (vi) Database Modelling Tool; (vii) Database Design Tool. A verificação dos resultados foram limitados a 10 (dez) páginas no motor de busca para cada *keyword*.

A aceitação das ferramentas no MLM estava sujeita a um processo de seleção baseado em alguns requisitos. Primeiramente, a ferramenta deveria oferecer acesso para alguma forma de uso (*e.g.* versão *trial*), necessitaria dar suporte a algum nível de modelagem de BDs e precisaria ter *interface* em inglês ou português. Após, as ferramentas incluídas deveriam ter as notações e níveis de modelagem extraídos, juntamente com outras informações relevantes, e então categorizadas.

#### 4. Condução da MLM

Para a execução do SLM foi necessário adaptar a sintaxe da *string* genérica para gerar outras versões, buscando assim adequar as peculiaridades de parametrização das diferentes bases utilizadas. Em seguida, foi realizada a busca dos estudos nas bases de dados. O número de estudos primários retornados por biblioteca digital obteve-se a seguinte distribuição: (i) IEEE Xplore: 513; (ii) Scopus: 465; (iii) ACM: 1240; (iv) Springer Link: 683; (v) ScienceDirect: 826.

Com o conjunto inicial de 3727 estudos primários identificados, foram definidos cinco (5) ciclos de seleção. Nestas iterações houve a exclusão de estudos duplicados (restando 3513), a seleção de estudos baseado em título e *abstract* (restando 34), seleção baseada em texto completo (restando 18) e a seleção baseada na avaliação de qualidade (restando 10). Cada iteração tinha o objetivo de eliminar estudos que estavam fora do escopo da pesquisa ou considerados não relevantes. Na última iteração, 18 estudos tiveram sua qualidade analisada. Foi estabelecido que apenas estudos com pontuação acima de 5 (cinco) seriam aceitos. Assim, após a aplicação dos CQs foram excluídos 8 (oito) trabalhos. O conjunto final de dez (10) estudos aprovados na SLM seguiu para a etapa de extração de dados. Com os trabalhos acadêmicos a menor pontuação foi de 5.3, enquanto o maior alcançou 10. A Tabela 2 resume os resultados obtidos na avaliação da qualidade.

**Tabela 2. Resultados da Avaliação de Qualidade**

Estudos Referência	Critérios de Qualidade							Pontuação
	CQ1	CQ2	CQ3	CQ4	CQ5	CQ6	CQ7	Total
Ayadi et al. 2016	P	P	T	P	N	T	P	5.7
Celikovic et al. 2014	T	T	N	T	P	P	T	7.0
Dimitrieski et al. 2015	T	T	T	T	T	T	T	10.0
Hammer and Mc Leod 1981	T	T	N	T	N	P	P	5.5
Jagannathan et al. 1988	T	T	N	T	P	N	N	5.5
Kersten et al. 2011	P	T	T	P	N	N	P	5.3
Litwin et al. 1989	P	T	N	T	N	P	T	5.4
Mazairac and Beetz 2013	T	T	P	T	N	N	P	5.9
Shipman 1981	T	T	N	T	P	N	P	6.0
Tian et al. 2006	T	P	N	T	P	P	T	6.4

A execução do protocolo de busca na literatura cinza retornou um total de 132 ferramentas. Após sumarização foi realizado a exclusão de duplicatas (restando 67). Durante esse processo houve doze (12) ferramentas que não puderam ser avaliadas *e.g.* impossibilidade de instalação ou incompatibilidade com o ambiente utilizado.

Ao fim, foi possível executar testes de uso em 55 ferramentas, e conseqüentemente, a extração dos dados relevantes ao estudo. Durante o uso das ferramentas foram realizados modelos simples de bancos de dados, em que se procurou observar o suporte aos níveis de modelagem e as notações e linguagens usadas.

## 5. Resultados do MLM

Em relação ao estado da arte (**QP1**) do desenvolvimento de DSLs aplicadas na modelagem de bancos de dados, identificou-se um estudo que apresenta a *System Modeling Tool (MIST)* [Dimitrieski et al. 2015], o qual utiliza uma DSL bidirecional chamada EERDSL, para modelagem conceitual utilizando a abordagem EER (*Extended Entity-Relationship*). Quanto às metodologias, técnicas ou propostas de refinamento (**QP1.1**) com base em modelos de dados, apenas os dois estudos mencionados aplicam conceitos para refinamento, sendo que tais conceitos são apoiados na normalização de BD para auxiliar os desenvolvedores que utilizam sua solução.

No que se refere às tecnologias usadas como suporte para o desenvolvimento de DSL (**QP1.2**), registram-se: Xtext, Xtend, Sirius e Eugenia [Celikovic et al. 2014, Dimitrieski et al. 2015], StarUML [Ayadi et al. 2016], IfcDoc Tool e ViewEdit Tool [Mazairac and Beetz 2013], MonetDB [Kersten et al. 2011], Java, JFlex e JCup [Tian et al. 2006]. No entanto, estudos mais antigos eram geralmente especificações de DSL, não apresentando qualquer forma de implementação ou ferramenta usada [Jagannathan et al. 1988, Litwin et al. 1989, Shipman 1981].

Sobre os métodos usados para avaliar as DSL (**QP2**) existe apenas um estudo preliminar que apresenta a validação da proposta [Dimitrieski et al. 2015] usando 16 participantes, sendo 2 especialistas em IHC, 3 especialistas em modelagem de sistemas e 11 estudantes (6 mestrados na área de bancos de dados e 5 doutorandos com experiência em modelagem). Em geral, os outros estudos indicam a falta de uma avaliação de suas proposições como um possível trabalho futuro.

Entre os aspectos positivos e negativos observados (**QP2.1**), destaca-se como aspectos positivos ser de fácil compreensão e com modelagem intuitiva, além da independência de plataforma [Mazairac and Beetz 2013, Tian et al. 2006]. Já os pontos negativos foram a falta de geração automática de código SQL [Ayadi et al. 2016, Dimitrieski et al. 2015] e a falta de implementação real de DSLs, havendo apenas especificações entre os trabalhos [Ayadi et al. 2016, Hammer and McLeod 1981, Jagannathan et al. 1988, Kersten et al. 2011, Tian et al. 2006].

Os principais desafios identificados pelos estudos (**QP2.2**), em geral, são as avaliações das abordagens, bem como a evolução e/ou simplificação das propostas. Na Tabela 3 as DSLs são apresentadas em relação ao seu tipo. No entanto, é importante observar que os estudos que marcam a coluna **bidirecional** [Celikovic et al. 2014, Dimitrieski et al. 2015] são versões diferentes da mesma implementação de DSL, enquanto o [Hammer and McLeod 1981, Jagannathan et al. 1988] são uma especificação e implementação de DSL baseada nesta especificação, respectivamente.

Quanto ao estado da prática (**QP3**) das ferramentas utilizadas na modelagem de BDs, foram mapeadas 55 ferramentas. Assim, houve a classificação quanto ao seu tipo, sendo 29 exclusivas de modelagem BDs (*Data Modeling*), 13 de modelagem

Figura 2. Ferramentas para Modelagem de Bancos de Dados.

#	Ferramenta	Tipo				Modelos				Notações Suportadas							Ambiente			Licença	
		DM	FIDE	DG	EM	C	L	F	CF	IDE/Flx	CN	MN	UML	BN	AN	ON	D	W	C	G	
1	AnalyseSI	✓				✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
2	Aqua Data Studio ER Modeler		✓			✓	✓	✓	✓	✓	✓					✓	✓	✓	✓	✓	
3	Astah			✓		✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
4	brModelo			✓		✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
5	Creately			✓		✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
6	Database Deployment Manager		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
7	Database Workbench		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
8	DB Designer					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
9	DB-Main		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
10	DBDesigner 4		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
11	DBDesigner.net		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
12	dbdiagram.io					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
13	dbDiff					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
14	dbForge Studio for MySQL		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
15	DBSchema					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
16	DBVisualizer					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
17	DbWrench					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
18	DeZign for Databases					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
19	Dia					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
20	dModelAid					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
21	Enterprise Architect					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
22	ER-Assistant					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
23	ER/Builder					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
24	ER/Studio Data Architect					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
25	ERD Concepts					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
26	ERDesigner NG					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
27	ERDPPlus					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
28	Erwin Data Modeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
29	GenMyModel RDS					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
30	InfoSphere Data Architect					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
31	Jeddict					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
32	ModelRight					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
33	MySQL Workbench		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
34	Navicat Data Modeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
35	Navicat Data Modeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
36	Open ModelSphere					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
37	Oracle SQL Developer Data Modeler		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
38	pgModeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
39	PowerDesigner					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
40	QuickDBD					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
41	RISE					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
42	Software Ideas Modeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
43	SQL Database Modeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
44	SQL Maestro					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
45	SQL Power Architect		✓			✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
46	SQL Server Management Studio					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
47	SQLDBM					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
48	SQLyog					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
49	Toad Data Modeler					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
50	Valentina Studio					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
51	Vertabelo					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
52	Visual Paradigm					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
53	Win Ak&D					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
54	WWW SQL Designer					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	
55	xCase					✓	✓	✓	✓	✓						✓	✓	✓	✓	✓	

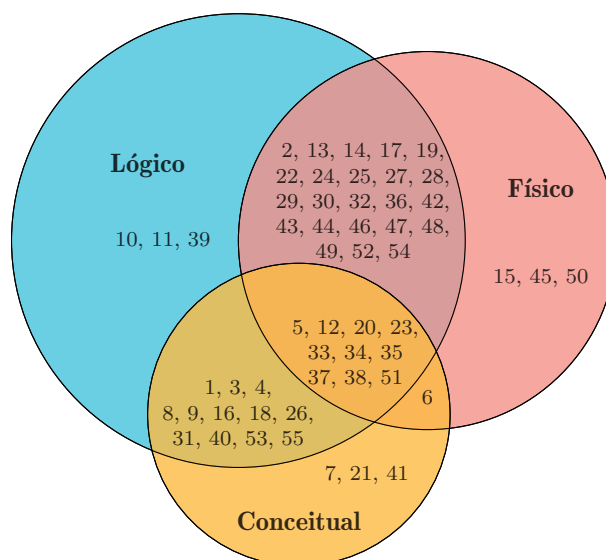
Legenda: DM (Data Modeling) FIDE (Full IDE) DG (Diagramming) EM (Enterprise Modeling) EM (Enterprise Modeling) | C (Conceptual) L (Lógico) F (Físico) | CF (Crow's Foot) CN (Chen's Notation) MN (Merise Notation) BN (Barker's Notation) AN (Arrow Notation) ON (Other Notation) | D (Desktop) W (Web) C (Comercial) G (Gratuita)

**Tabela 3. Categorização das DSLs Propostas**

Referência	Estudos Primários		Tipo de DSL		
	DSL	Textual	Gráfica	Bidirecional	
Ayadi et al. 2016	Ayadi's Notation		✓		
Celikovic et al. 2014	EERDSL v.1				✓
Dimitrieski et al. 2015	EERDSL v.2				✓
Hammer and McLeod 1981	SDM	✓			
Jagannathan et al. 1988	SDM	✓			
Kersten et al. 2011	SciQL	✓			
Litwin et al. 1989	MSQL	✓			
Mazairac and Beetz 2013	BIMQL	✓			
Shipman 1981	DAPLEX	✓			
Tian et al. 2006	NeuroQL	✓			

com conexão ao BDs integrado e execução de consultas (*Full IDE*), 10 com suporte a diagramação de diversos tipos de modelos (*Diagramming*) e 3 ferramentas projetadas para grandes empresas (*Enterprise Modeling*).

Em relação às notações usadas nas ferramentas (**QP3.1**) foram identificados 18 (dezoito) notações, com destaque a notação *Crow's Foot* com 35 ocorrências e a notação IDEF1X com 23 registros. E, finalmente, no que diz respeito aos modelos suportados pelas ferramentas (**QP3.2**) foi constatado que, individualmente, 26 ferramentas oferecem suporte à modelagem conceitual, 48 ferramentas à modelagem lógica e 37 à modelagem física. O conjunto representando as intersecções do suporte aos modelos quanto às ferramentas é apresentado no diagrama de Venn da Figura 3.



**Figura 3. Diagrama de Venn dos modelos suportados nas ferramentas.**

## 6. Ameaças à Validade

Ameaças ao resultado do estudo foram identificadas no MLM realizado, e então avaliadas de acordo com [Cook and Campbell 1979]. A possibilidade de que as QPs ou os termos de pesquisa que estruturam a *string* de busca sejam incompletos é uma ameaça. Para mitigá-la, pesquisadores da área de DSL e modelagem de bancos de dados foram consultados. Além disso, foi realizada uma pesquisa piloto para avaliar



a consistência da *string* de pesquisa. Vale destacar que outra ameaça é a qualidade do material publicado que foi coletado na literatura cinza. Outra ameaça é o uso de métodos incorretos de busca, o que pode levar a exclusão de estudos relevantes, a ocorrência de vieses na seleção ou no conteúdo dos estudos primários. Na tentativa de mitigar esses riscos, um protocolo foi definido com base em modelos de referência já bem estabelecidos na literatura. Uma possível ameaça é o viés na extração de dados, o que leva a erros de conclusão. Para atenuar esse problema foi realizado uma leitura criteriosa e, assim como os testes de uso das ferramentas, houve a síntese de dados em planilha eletrônica para uma melhor análise entre os pesquisadores.

## 7. Considerações Finais

Este estudo forneceu algumas evidências de que, a cada ano, um número significativo de trabalhos apresentando diferentes tipos de notações é publicado. Isso é de certa forma surpreendente, devido ao fato de que as notações ER usadas hoje pela indústria e pela academia, como as de Peter Chen e Barker, não são propostas recentes. Portanto, conclui-se que a modelagem ER continua um amplo campo de pesquisa com algumas lacunas, *e.g.* a perda de ferramentas que suportam o projeto em muitos ciclos de vida por meio de técnicas de projeto automatizadas.

## Referências

- ANSI, A. N. S. I. (1975). Interim Report: ANSI/X3/SPARC Study Group on DBMSs 75-02-08. *ACM SIG on Management of Data*.
- Ayadi, M. G., Bouslimi, R., and Akaichi, J. (2016). A framework for medical and health care databases and data warehouses conceptual modeling support. *Network Modeling Analysis in Health Informatics and Bioinformatics*.
- Celikovic, M., Dimitrieski, V., Aleksic, S., Ristic, S., and Lukovic, I. (2014). A DSL for EER Data Model Specification. In *23rd Int. Conf. on Information Systems Development*, pages 290–297.
- Chen, P. P.-S. (1976). The Entity-relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems*, 1(1):9–36.
- Cook, T. and Campbell, D. (1979). *Quasi-Experimentation: Design and Analysis Issues for Field Settings*. Houghton Mifflin.
- Cougo, P. (2013). *Modelagem conceitual e projeto de banco de dados*. Elsevier.
- Dimitrieski, V., Čeliković, M., Aleksić, S., Ristić, S., Alargt, A., and Luković, I. (2015). Concepts and Evaluation of the Extended Entity-relationship Approach to Database Design in a Multi-paradigm Information System Modeling Tool. *Computer Languages, Systems & Structures*, 44(Part C):299–318.
- Dybå, T. and Dingsøyr, T. (2008). Strength of evidence in systematic reviews in software engineering. In *2nd ACM-IEEE Int. Symp. on Empirical Software Engineering and Measurement*, pages 178–187. ACM.
- Fowler, M. (2010). *Domain Specific Languages*. Addison-Wesley, 1st edition.
- Garousi, V. and Felderer, M. and Mäntylä, M. V. (2019). Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. *Information & Software Technology*, 106:101–121.

- Gregersen, H. and Jensen, C. S. (1999). Temporal entity-relationship models—a survey. *IEEE Transactions on Knowledge and Data Engineering*, 11(3):464–497.
- Hammer, M. and McLeod, D. (1981). Database Description with SDM: A Semantic Database Model. *ACM Transactions on Database Systems*, 6(3):351–386.
- Jagannathan, D., Fritchman, B. L., Guck, R. L., Thompson, J. P., and Tolbert, D. M. (1988). SIM: A Database System Based on the Semantic Data Model. *ACM SIGMOD Record*, 17(3):46–55.
- Kersten, M., Zhang, Y., Ivanova, M., and Nes, N. (2011). SciQL, a Query Language for Science Applications. In *Work. on Array Databases*, pages 1–12. ACM.
- Litwin, W., Abdellatif, A., Zeroual, A., Nicolas, B., and Vigier, P. (1989). MSQL: A Multidatabase Language. *Information Sciences*, 49:59–101.
- Lorenz, M., Hesse, G., and Rudolph, J.-P. (2016). Object-relational Mapping Revisited - A Guideline Review and Consolidation. In *11th International Joint Conference on Software Technologies*, pages 157–168.
- Ma, Z. and Yan, L. (2010). A Literature Overview of Fuzzy Conceptual Data Modeling. *Information Science and Engineering*, 26:427–441.
- Mazairac, W. and Beetz, J. (2013). BIMQL - an open query language for building information models. *Advanced Engineering Informatics*, 27(4):444–456.
- Mernik, M., Heering, J., and Sloane, A. M. (2005). When and How to Develop Domain-specific Languages. *ACM Computer Survey*, 37(4):316–344.
- Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering. In *12th Int. Conf. on Evaluation and Assessment in Software Engineering*, pages 68–77. BCS Learning & Dev. Ltd.
- Shipman, D. W. (1981). The Functional Data Model and the Data Languages DAPLEX. *ACM Transactions Database Systems*, 6(1):140–173.
- Siau, K. (2012). *Innovations in Database Design, Web Applications, and Information Systems Management*. Information Science Reference.
- Tian, H., Sunderraman, R., Calin-Jageman, R., Yang, H., Zhu, Y., and Katz, P. (2006). NeuroQL: A Domain-specific Query Language for Neuroscience Data. In *Int. Conf. on Current Trends in Database Technology*, pages 613–624. Springer.
- Torres, A., Galante, R., Pimenta, M. S., and Martins, A. J. B. (2017). Twenty years of object-relational mapping: A survey on patterns, solutions, and their implications on application design. *Information and Software Technology*.
- Van Deursen, A., Klint, P., and Visser, J. (2000). Domain-specific languages: An annotated bibliography. *ACM SIGPLAN Notices*, 35(6):26–36.
- Vara, J., Bollati, V., Jiménez, A., and Marcos, E. (2014). Dealing with Traceability in the MDD of Model Transformations. *Trans. on Soft. Eng.*, 40(6):555–583.
- Voelter, M. (2009). Best Practices for DSLs and Model-Driven Development. *Journal of Object Technology*, 8(6):79–102.