Table of Contents

Preface xi

1. Graph Thinking 1
Why Now? Putting Database Technologies in Context 2
1960s–1980s: Hierarchical Data 3
1980s–2000s: Entity-Relationship 4
2000s–2020s: NoSQL 5
2020s–?: Graph 7
What Is Graph Thinking? 9
Complex Problems and Complex Systems 10
Complex Problems in Business 10
Making Technology Decisions to Solve Complex Problems 12
So You Have Graph Data. What's Next? 15
Seeing the Bigger Picture 19
Getting Started on Your Journey with Graph Thinking 20

2. Evolving from Relational to Graph Thinking 21
Chapter Preview: Translating Relational Concepts to Graph Terminology 21
Relational Versus Graph: What's the Difference? 22
Data for Our Running Example 23
Relational Data Modeling 25
Entities and Attributes 26
Building Up to an ERD 27
Concepts in Graph Data 28
Fundamental Elements of a Graph 28
Adjacency 29
Neighborhoods 30
Distance 30

Degree 31
The Graph Schema Language 33
Vertex Labels and Edge Labels 33
Properties 34
Edge Direction 35
Self-Referencing Edge Labels 38
Multiplicity of Your Graph 38
Full Example Graph Model 41
Relational Versus Graph: Decisions to Consider 43
Data Modeling 43
Understanding Graph Data 43
Mixing Database Design with Application Purpose 44
Summary 44

3. Getting Started: A Simple Customer 360 47
Chapter Preview: Relational Versus Graph 48
The Foundational Use Case for Graph Data: C360 48
Why Do Businesses Care About C360? 50
Implementing a C360 Application in a Relational System 51
Data Models 51
Relational Implementation 54
Example C360 Queries 58
Implementing a C360 Application in a Graph System 61
Data Models 62
Graph Implementation 63
Example C360 Queries 70
Relational Versus Graph: How to Choose? 75
Relational Versus Graph: Data Modeling 75
Relational Versus Graph: Representing Relationships 76
Relational Versus Graph: Query Languages 76
Relational Versus Graph: Main Points 77
Summary 78
Why Not Relational? 79
Making a Technology Choice for Your C360 Application 79

4. Exploring Neighborhoods in Development 81
Chapter Preview: Building a More Realistic Customer 360 81
Graph Data Modeling 101 82
Should This Be a Vertex or an Edge? 83
Lost Yet? Let Us Walk You Through Direction 86
A Graph Has No Name: Common Mistakes in Naming 89
Our Full Development Graph Model 91

Before We Start Building 93
Our Thoughts on the Importance of Data, Queries, and the End User 94
Implementation Details for Exploring Neighborhoods in Development 95
Generating More Data for Our Expanded Example 97
Basic Gremlin Navigation 97
Advanced Gremlin: Shaping Your Query Results 106
Shaping Query Results with the project(), fold(), and unfold() Steps 107
Removing Data from the Results with the where(neq()) Pattern 110
Planning for Robust Result Payloads with the coalesce() Step 112
Moving from Development into Production 115

5. Exploring Neighborhoods in Production..... 117
Chapter Preview: Understanding Distributed Graph Data in Apache Cassandra 119
Working with Graph Data in Apache Cassandra 120
The Most Important Topic to Understand About Data Modeling: Primary Keys 120
Partition Keys and Data Locality in a Distributed Environment 121
Understanding Edges, Part 1: Edges in Adjacency Lists 126
Understanding Edges, Part 2: Clustering Columns 128
Understanding Edges, Part 3: Materialized Views for Traversals 132
Graph Data Modeling 201 136
Finding Indexes with an Intelligent Index Recommendation System 140
Production Implementation Details 142
Materialized Views and Adding Time onto Edges 142
Our Final C360 Production Schema 144
Bulk Loading Graph Data 146
Updating Our Gremlin Queries to Use Time on Edges 149
Moving On to More Complex, Distributed Graph Problems 152
Our First 10 Tips to Get from Development to Production 152

6. Using Trees in Development..... 155
Chapter Preview: Navigating Trees, Hierarchical Data, and Cycles 155
Seeing Hierarchies and Nested Data: Three Examples 156
Hierarchical Data in a Bill of Materials 156
Hierarchical Data in Version Control Systems 157
Hierarchical Data in Self-Organizing Networks 157
Why Graph Technology for Hierarchical Data? 158
Finding Your Way Through a Forest of Terminology 159
Trees, Roots, and Leaves 159
Depth in Walks, Paths, and Cycles 160
Understanding Hierarchies with Our Sensor Data 162

Understand the Data 163
Conceptual Model Using the GSL Notation 170
Implement Schema 171
Before We Build Our Queries 174
Querying from Leaves to Roots in Development 174
Where Has This Sensor Sent Information To? 175
From This Sensor, What Was Its Path to Any Tower? 178
From Bottom Up to Top Down 184
Querying from Roots to Leaves in Development 184
Setup Query: Which Tower Has the Most Sensor Connections So That We Could Explore It for Our Example? 185
Which Sensors Have Connected Directly to Georgetown? 186
Find All Sensors That Connected to Georgetown 187
Depth Limiting in Recursion 189
Going Back in Time 190

7. Using Trees in Production..... 191
Chapter Preview: Understanding Branching Factor, Depth, and Time on Edges 191
Understanding Time in the Sensor Data 192
Final Thoughts on Time Series Data in Graphs 200
Understanding Branching Factor in Our Example 200
What Is Branching Factor? 201
How Do We Get Around Branching Factor? 202
Production Schema for Our Sensor Data 203
Querying from Leaves to Roots in Production 205
Where Has This Sensor Sent Information to, and at What Time? 205
From This Sensor, Find All Trees up to a Tower by Time 206
From This Sensor, Find a Valid Tree 209
Advanced Gremlin: Understanding the where().by() Pattern 211
Querying from Roots to Leaves in Production 213
Which Sensors Have Connected to Georgetown Directly, by Time? 214
What Valid Paths Can We Find from Georgetown Down to All Sensors? 215
Applying Your Queries to Tower Failure Scenarios 218
Applying the Final Results of Our Complex Problem 223
Seeing the Forest for the Trees 223

8. Finding Paths in Development..... 225
Chapter Preview: Quantifying Trust in Networks 226
Thinking About Trust: Three Examples 226
How Much Do You Trust That Open Invitation? 226
How Defensible Is an Investigator's Story? 227

How Do Companies Model Package Delivery? 228
Fundamental Concepts About Paths 229
Shortest Paths 230
Depth-First Search and Breadth-First Search 232
Learning to See Application Features as Different Path Problems 233
Finding Paths in a Trust Network 234
Source Data 234
A Brief Primer on Bitcoin Terminology 236
Creating Our Development Schema 236
Loading Data 237
Exploring Communities of Trust 238
Understanding Traversals with Our Bitcoin Trust Network 240
Which Addresses Are in the First Neighborhood? 240
Which Addresses Are in the Second Neighborhood? 241
Which Addresses Are in the Second Neighborhood, but Not the First? 242
Evaluation Strategies with the Gremlin Query Language 244
Pick a Random Address to Use for Our Example 245
Shortest Path Queries 246
Finding Paths of a Fixed Length 247
Finding Paths of Any Length 250
Augmenting Our Paths with the Trust Scores 253
Do You Trust This Person? 259

9. Finding Paths in Production. 261
Chapter Preview: Understanding Weights, Distance, and Pruning 262
Weighted Paths and Search Algorithms 262
Shortest Weighted Path Problem Definition 263
Shortest Weighted Path Search Optimizations 264
Normalization of Edge Weights for Shortest Path Problems 267
Normalizing the Edge Weights 267
Updating Our Graph 272
Exploring the Normalized Edge Weights 273
Some Thoughts Before Moving On to Shortest Weighted Path Queries 277
Shortest Weighted Path Queries 277
Building a Shortest Weighted Path Query for Production 278
Weighted Paths and Trust in Production 288

10. Recommendations in Development. 291
Chapter Preview: Collaborative Filtering for Movie Recommendations 292
Recommendation System Examples 292
How We Give Recommendations in Healthcare 292
How We Experience Recommendations in Social Media 293

How We Use Deeply Connected Data for Recommendations in Ecommerce 294
An Introduction to Collaborative Filtering 295
Understanding the Problem and Domain 295
Collaborative Filtering with Graph Data 297
Recommendations via Item-Based Collaborative Filtering with Graph Data 298
Three Different Models for Ranking Recommendations 299
Movie Data: Schema, Loading, and Query Review 303
Data Model for Movie Recommendations 303
Schema Code for Movie Recommendations 305
Loading the Movie Data 307
Neighborhood Queries in the Movie Data 311
Tree Queries in the Movie Data 314
Path Queries in the Movie Data 316
Item-Based Collaborative Filtering in Gremlin 318
Model 1: Counting Paths in the Recommendation Set 318
Model 2: NPS-Inspired 319
Model 3: Normalized NPS 322
Choosing Your Own Adventure: Movies and Graph Problems Edition 324

11. Simple Entity Resolution in Graphs..... 325
Chapter Preview: Merging Multiple Datasets into One Graph 325
Defining a Different Complex Problem: Entity Resolution 326
Seeing the Complex Problem 328
Analyzing the Two Movie Datasets 329
MovieLens Dataset 329
Kaggle Dataset 336
Development Schema 339
Matching and Merging the Movie Data 340
Our Matching Process 340
Resolving False Positives 343
False Positives Found in the MovieLens Dataset 343
Additional Errors Discovered in the Entity Resolution Process 344
Final Analysis of the Merging Process 346
The Role of Graph Structure in Merging Movie Data 347

12. Recommendations in Production..... 349
Chapter Preview: Understanding Shortcut Edges, Precomputation, and Advanced Pruning Techniques 350
Shortcut Edges for Recommendations in Real Time 350
Where Our Development Process Doesn't Scale 351
How We Fix Scaling Issues: Shortcut Edges 352
Seeing What We Designed to Deliver in Production 353

Pruning: Different Ways to Precompute Shortcut Edges 354
Considerations for Updating Your Recommendations 356
Calculating Shortcut Edges for Our Movie Data 357
Breaking Down the Complex Problem of Precalculating Shortcut Edges 357
Addressing the Elephant in the Room: Batch Computation 362
Production Schema and Data Loading for Movie Recommendations 363
Production Schema for Movie Recommendations 364
Production Data Loading for Movie Recommendations 365
Recommendation Queries with Shortcut Edges 366
Confirming Our Edges Loaded Correctly 367
Production Recommendations for Our User 368
Understanding Response Time in Production by Counting Edge Partitions 372
Final Thoughts on Reasoning About Distributed Graph Query Performance 375

13. Epilogue..... 377
Where to Go from Here? 378
Graph Algorithms 378
Distributed Graphs 379
Graph Theory 380
Network Theory 380
Stay in Touch 382

Index..... 383