

List of Figures

1.1	Massive Data Growth in past 10 years	4
1.2	SQL vs. NoSQL Database Usage	4
1.3	Single Database vs. Multi-Database Use	5
1.4	Multi-Database Combinations	5
1.5	Overview of Provenance	9
1.6	Annotation Propagation	15
2.1	Temporal Database as time cube with Snapshot (Conventional) Database as time slice in cube	42
2.2	Generalized Architecture of ZILD	44
2.3	Example Query 3 Result	46
3.1	Example : Why and How Provenance	57
3.2	Multi-layer Provenance Example	58
3.3	Multi-layer Provenance Graph of Example in Figure 3.2	59
3.4	ZILRDB Schema Design Flowchart	63
3.5	Proposed DPHQ Framework	69
3.6	DPHQ : Querying and Provenance Generation for Example Query 4	78
3.7	Provenance Graph of Example Query 4	82
3.8	Storage Requirement for Provenance Data	89
3.9	Average execution time of queries on provenance : Traversal Depth 1	90
3.10	Average execution time of queries on provenance : Traversal Depth 2	91
3.11	Average execution time of queries on provenance : Traversal Depth 3	92
3.12	Average execution time of queries on provenance : Traversal Depth 4	92

3.13	Average execution time of queries on provenance : Traversal Depth 5	93
3.14	Average execution time of queries on provenance : Traversal Depth 6	93
4.1	Twitter Social Network Graph	101
4.2	ZILGDB Architecture	105
4.3	Snapshot of Zero-Information Loss Graph Database	107
4.4	Twitter Data Model	110
4.5	Timeline with Tweet	111
4.6	Neo4j Schema for Twitter Data Set	112
4.7	Provenance Graph Construction Flow Diagram	113
4.8	Snapshot of Result of Example Query 2	115
4.9	Partial Provenance Graph of Example Query 2	116
4.10	Provenance of Result Tuple t1 of Example Query 2	116
4.11	Partial Provenance Graph of Example Query 3	117
4.12	Snapshot of Result of Example Query 4	117
4.13	Snapshot of Provenance of Example Query 4	118
4.14	Result of Provenance Query PQ1	119
4.15	Query Performance with and without timeline	124
4.16	Query Performance without Provenance Capture	125
4.17	Query Performance with Provenance Capture	125
4.18	Querying Provenance for Justifying Result Tuples	128
4.19	Querying Provenance for Historical Data	128
4.20	A Cyclic Process Model Based on Provenance Framework	129
4.21	Snapshot of Result of Example Query 5	130
4.22	Partial Provenance Graph of Example Query 5	131
4.23	Partial Provenance Graph	132
4.24	All Tweets Posted by Identified User 'Demise_ _ _ _'	132
5.1	Cassandra Column, Row & Column-Family Structure	137
5.2	ZILKVD Architecture	140
5.3	A Snapshot of "update_provenance" Column Family	144
5.4	Twitter Data Streaming	145

5.5	Open Authentication Process for Twitter	146
5.6	Cassandra Data Model	149
5.7	Example Query 3 Result	153
5.8	Example Query 4 Result	155
5.9	Example Query 5 Result	156
5.10	Provenance Storage	158
5.11	A Snapshot of "select_provenance" Column Family	159
5.12	Performance of Select Queries without and with Provenance	165
5.13	Performance of Aggregate Queries without and with Provenance	166
5.14	Performance of Update Queries without and with Provenance	167
5.15	Overall Query Performance without and with Provenance	168
5.16	Provenance Overhead for Different Query Sets	168
5.17	Provenance Querying	169
6.1	Provenance Query Engine	175
6.2	Relational Database Schema	180
6.3	Graph Database Schema	181
6.4	Key-Value Pair Database Schema	182
6.5	Provenance Graph in RDBMS	184
6.6	Provenance Graph in GDBMS	184
6.7	Provenance Graph in KVPDB	185

List of Tables

1.1	Why-Provenance and How-Provenance for query in Figure 1.6	17
1.2	Qualitative analysis of different provenance solutions for Relational Database with proposed DPHQ Framework	33
1.3	Qualitative analysis of different provenance solutions for Graph Database with proposed SDP Framework	35
1.4	Qualitative analysis of different provenance solutions for Key-Value Pair Database with proposed BSDP Framework	37
2.1	A Time Stamped Temporal Relation	45
2.2	Example Query 1 Result (Now)	46
2.3	Example Query 1 Result (01/01/2020)	46
2.4	Example Operation Sequence	51
2.5	Example Query Table	52
2.6	Result of Query Q4	53
2.7	Result of Query Q10	53
3.1	Provenance Relation Algebra (PRA)	71
3.2	Sample Part Table	71
3.3	Sample Partsupp Table	72
3.4	Sample Region Table	72
3.5	Sample Supplier Table	72
3.6	Sample Nation Table	72
3.7	Example Query Q1 Result	73
3.8	Example Query Q2 Result	74

3.9	Example Query Q3 Result	74
3.10	Example Query Q4 Result	75
3.11	Example Query Q5 Result	76
3.12	Example Query Q6 Result	77
3.13	Sample Query Table (querytabletpch) in Relational Database	81
3.14	Sample Provenance Table (provtbl1) in Relational Database	81
3.15	Sample Data Queries for Provenance Capture	87
3.16	Sample Queries on Provenance	89
4.1	Sample Data Retrieval Queries with usefulness	123
4.2	Sample Data Update Queries	126
4.3	Sample Queries on Provenance	127
5.1	Example Provenance Query PQ2 Result	162
5.2	Example Provenance Query PQ3 Result	163
5.3	Example Provenance Query PQ4 Result	163
5.4	Example Provenance Query PQ5 Result	163
5.5	Sample Select Queries	165
5.6	Sample Aggregate Queries	166
5.7	Sample Update Queries	167
5.8	Sample Provenance Queries	168
6.1	Keywords with Color Code	178
6.2	Example Provenance Queries in English	179
6.3	Provenance Query Templates	199

List of Abbreviations/Symbols

Notation	Definition
$agg(T_{id})$	Aggregating all T_{id} 's using '+' in Project Queries in Relational Database
$agg(P_i)$	Aggregating all P_i 's using '+' in Join Queries in Relational Database
$agg(T_{id})$	Aggregating all T_{id} 's using ' \otimes ' in Aggregate Queries in Relational Database
G_Q	Query Graph
G_P	Provenance Graph
F_{rl}	Twitter User's Friends List
F_{ol}	Twitter User's Followers List
F_{rld}	Twitter User's Friend Detail
F_{old}	Twitter User's Follower Detail
A_t	Access Token for Streaming Real-Life Twitter Data
A_{ts}	Access Token Secret for Streaming Real-Life Twitter Data
C_k	Consumer Key for Streaming Real-Life Twitter Data
C_{sk}	Consumer Secret Key for Streaming Real-Life Twitter Data
CV_o	Old Column Value before Update in Key-Value pair Database
CV_{owt}	Writetime of Old Column Value before Update in Key-Value pair Database
p_i	Provenance Path Expression of Non Key Column C_i in result tuple r of Select Query in Key-Value Pair Database
P	Comma Separated list of all p_i of C_i 's in result tuple r of Select Query in Key-Value Pair Database
$pv[i]$	pv is a vector of Provenance Path Expressions of all C_i 's in result tuple r of Aggregate Query where $pv[i]$ is Comma Separated List of all Provenance Path Expressions of C_i in Key-Value Pair Database