Transactional Information Systems:
Theory, Algorithms, and the Practice of
Concurrency Control and Recovery

Gerhard Weikum and Gottfried Vossen

© 2002 Morgan Kaufmann
ISBN 1-55860-508-8

“Teamwork is essential. It allows you to blame someone else.” (Anonymous)
Part I: Background and Motivation

• 1 What Is It All About?
• 2 Computational Models
Chapter 2: Computational Models

2.2 Ingredients
- 2.3 Page Model
- 2.4 Object Model
- 2.5 Roadmap
- 2.6 Lessons Learned

“Between theory and practice, some talk as they were two. Between theory and practice, both can be gained.”
(Bhagavad-gita 5:4)
Reminder: Database System Layers

Clients

Database Server

- Language & Interface Layer
- Query Decomposition & Optimization Layer
- Query Execution Layer
- Access Layer
- Storage Layer

Requests

Request Execution Threads

Data Accesses

Database
Ingredients

• Elementary operations
• Transactions (i.e., transaction program executions)
• Histories and schedules
• Characterization of correct schedules
• Protocols (i.e., rules for online algorithms)
Chapter 2: Computational Models

- 2.2 Ingredients
- 2.3 Page Model
- 2.4 Object Model
- 2.5 Roadmap
- 2.6 Lessons Learned
Definition 2.2 (Page Model Transaction):
A transaction \( t \) is a partial order of steps (actions) of the form \( r(x) \) or \( w(x) \), where \( x \in D \) and reads and writes as well as multiple writes applied to the same object are ordered. We write \( t = (op, <) \) for transaction \( t \) with step set \( op \) and partial order \( < \).

**Example:** \( r(s) \) \( w(s) \) \( r(t) \) \( w(t) \)
Page Model

“Syntax”: 

**Definition 2.2 (Page Model Transaction):**
A **transaction** \( t \) is a partial order of steps (actions) of the form \( r(x) \) or \( w(x) \), where \( x \in D \) and reads and writes as well as multiple writes applied to the same object are ordered. We write \( t = (\text{op}, <) \) for transaction \( t \) with step set \( \text{op} \) and partial order \(<\).

**Example:** \( r(s) \) \( w(s) \) \( r(t) \) \( w(t) \)

“Semantics”: 

Interpretation of \( j^{th} \) step, \( p_j \), of \( t \):
If \( p_j = r(x) \), then interpretation is assignment \( v_j := x \) to local variable \( v_j \)
If \( p_j = w(x) \) then interpretation is assignment \( x := f_j(v_{j_1}, ..., v_{j_k}) \).
with unknown function \( f_j \) and \( j_1, ..., j_k \) denoting \( t \)‘s prior read steps.
Chapter 2: Computational Models

- 2.2 Ingredients
- 2.3 Page Model
- **2.4 Object Model**
- 2.5 Roadmap
- 2.6 Lessons Learned
Definition 2.3 (Object Model Transaction):
A transaction $t$ is a (finite) tree of labeled nodes with
• the transaction identifier as the label of the root node,
• the names and parameters of invoked operations as labels of inner nodes, and
• page-model read/write operations as labels of leaf nodes, along with a partial order $<$ on the leaf nodes such that for all leaf-node operations $p$ and $q$ with $p$ of the form $w(x)$ and $q$ of the form $r(x)$ or $w(x)$ or vice versa, we have $p < q \lor q < p$.
Object Model

Definition 2.3 (Object Model Transaction):
A transaction \( t \) is a (finite) tree of labeled nodes with
• the transaction identifier as the label of the root node,
• the names and parameters of invoked operations as labels of inner nodes, and
• page-model read/write operations as labels of leaf nodes, along with a partial order \( < \) on the leaf nodes such that for all leaf-node operations \( p \) and \( q \) with \( p \) of the form \( w(x) \) and \( q \) of the form \( r(x) \) or \( w(x) \) or vice versa, we have \( p < q \lor q < p \)

Special case: layered transactions
(all leaves have same distance from root)

Derived inner-node ordering: \( a < b \) if all leaf-node descendants of \( a \) precede all leaf-node descendants of \( b \)
Example: DBS Internal Layers

Search ("Austin")

Fetch(x)

Fetch(y)

Store(z)

r (r)

r (l)

r (p)

r (q)

r (f)

r (p)

w (p)

r (r)

r (l)

w (l)
Example: Business Objects

Withdraw (x, 1000)

Append (h, ...)

Search (...)

Modify (x)

Fetch (x)

Modify (a)

Fetch (a)

Store (e)

Modify (d)

Modify (a)

Deposit (y, 1000)

Search (...)

Fetch (y)

Modify (y)

r (r)

r (l)

r (p)

r (p)

w (p)

r (s)

r (t)

r (t)

w (t)

r (t)

w (t)

r (s)

w (s)

r (r)

r (l)

r (q)

r (q)

w (q)
Chapter 2: Computational Models

- 2.2 Ingredients
- 2.3 Page Model
- 2.4 Object Model

**2.5 Roadmap**

- 2.6 Lessons Learned
## Roadmap

### Part II: Concurrency Control
- 3 Notions of Correctness PM
- 4 CC Algorithms
- 5 Multiversion CC
- 6 Notions of Correctness OM
- 7 CC Algorithms on Objects
- 8 CC on Relational DB
- 9 CC on Search Structures
- 10 Impl. & Pragmatic Issues

### Part III: Recovery
- 11 Transaction Recovery
- 12 Crash Recovery Correctness
- 13 CR Algorithms PM
- 14 CR Algorithms OM
- 15 Special Issues of Recovery
- 16 Media Recovery
- 17 Application Recovery

### Part IV: Coordination of Distributed Transactions
- 18 Distributed CC
- 19 Distributed Transaction Recovery
Chapter 2: Computational Models

- 2.2 Ingredients
- 2.3 Page Model
- 2.4 Object Model
- 2.5 Roadmap
- 2.6 Lessons Learned
Lessons Learned

“Nothing is as practical as a good theory.” (Albert Einstein)