Multi-level locking using B-trees
(or other hierarchies)

table
| page
| tuple

lock hierarchically
<table>
<thead>
<tr>
<th>Requested Lock</th>
<th>Existing Lock</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS</td>
<td>IS</td>
</tr>
<tr>
<td>IX</td>
<td>✓</td>
</tr>
<tr>
<td>S</td>
<td>✓</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
index on R(a)

select count(*)
from R
where a = 5;

insert into R(a)
values (5);

insert into R(a)
values (20);
LOG

log sequence no

LSN < OP

PREV LSN

100 Ti update
PX A B

\[ w_1(x) \]

\[ w_2(y) \]

commit \( T_1 \)
\[ w_3(z) \]
\[ w_3(w) \]

\[ w_2(x) \]

abort \( T_2 \)
MEMORY

LOG

101 Ti update P_x

DATA PAGE

P_x 101

DATA PAGES

P_x 101

DISK (non-volatile)

(Volatile)
write a page \( Rx \) to disk

\{ flush the log \}

\{ write the page \( Rx \) to disk \}
DISK

101 T1 update
  PX A B

102 T1 commit

DATA

PX 101