Dualities in Programming Languages

Martin Hirzel and Priya Nagpurkar
IBM Watson Research Center
PLDI-FIT 2010
Row-Based Layout

Values of different fields but same object contiguous.

Good locality if few hot objects.

Field access dereferences objectAddress + fieldOffset.
Column-Based Layout

Values of different objects but same field contiguous.

Good locality if few hot fields.

Field access dereferences fieldAddress + objectOffset.
Row-Based $\cong$ Column-Based Layout

Values of different fields but same object field contiguous.

Good locality if few hot objects.

Field access dereferences $\text{objectAddress} + \text{fieldOffset}.$

Object 0
Object 1
Object 2
Object 3
Observations

• Duality =
  pair of concepts
  + terminology substitution.

• Not always perfect
  (e.g., memory management easier for row-based layout.)
Observations

• Duality = pair of concepts + terminology substitution.

• Not always perfect (e.g., memory management easier for row-based layout.)

• Good excuse for fancy formatting.
Incremental \(\equiv\) Demand-driven Computation

Evaluate expression only when input becomes available. output is requested.

Trigger next expression when pushing output from previous expression.

Do less work per input per output than from-scratch computation.
Other Dualities in Programming Languages

<table>
<thead>
<tr>
<th>Garbage collection (GC)</th>
<th>=</th>
<th>Transactional memory (TM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static analysis</td>
<td>=</td>
<td>Dynamic analysis</td>
</tr>
<tr>
<td>Reachability-based GC</td>
<td>=</td>
<td>Reference-counting GC</td>
</tr>
<tr>
<td>Change propagation</td>
<td>=</td>
<td>Memoization</td>
</tr>
<tr>
<td>Language runtime system</td>
<td>=</td>
<td>Operating system</td>
</tr>
<tr>
<td>Method-based JIT</td>
<td>=</td>
<td>Trace-based JIT</td>
</tr>
<tr>
<td>Subject/Observer</td>
<td>=</td>
<td>Iterator</td>
</tr>
</tbody>
</table>

And finally, the most practical duality, which you can use right away …
Problem $\equiv$ Solution

Research often starts from problem solution and seeks a solution problem.

In other words, the problem solution becomes a meta-problem solution.

Can do this by looking at dual problem solution and adopting its solution problem.

Happy dualities hunting!