

DDD IN A DISTRIBUTED WORLD

Gojko Adzic

<http://gojko.net>
gojko@gojko.net
@gojkoadzic

Aggregates...

Object clusters defining strong relationships...

...objects outside of an aggregate should only reference the aggregate root and not anything inside it.

Aggregates are not just about
pointers

Aggregates are not just about
pointers

Think of **business units of
consistency**

Aggregates are not just about
pointers

Think of **business units of
consistency**

“a meaningful whole”

Technical impacts

- A natural unit of distribution

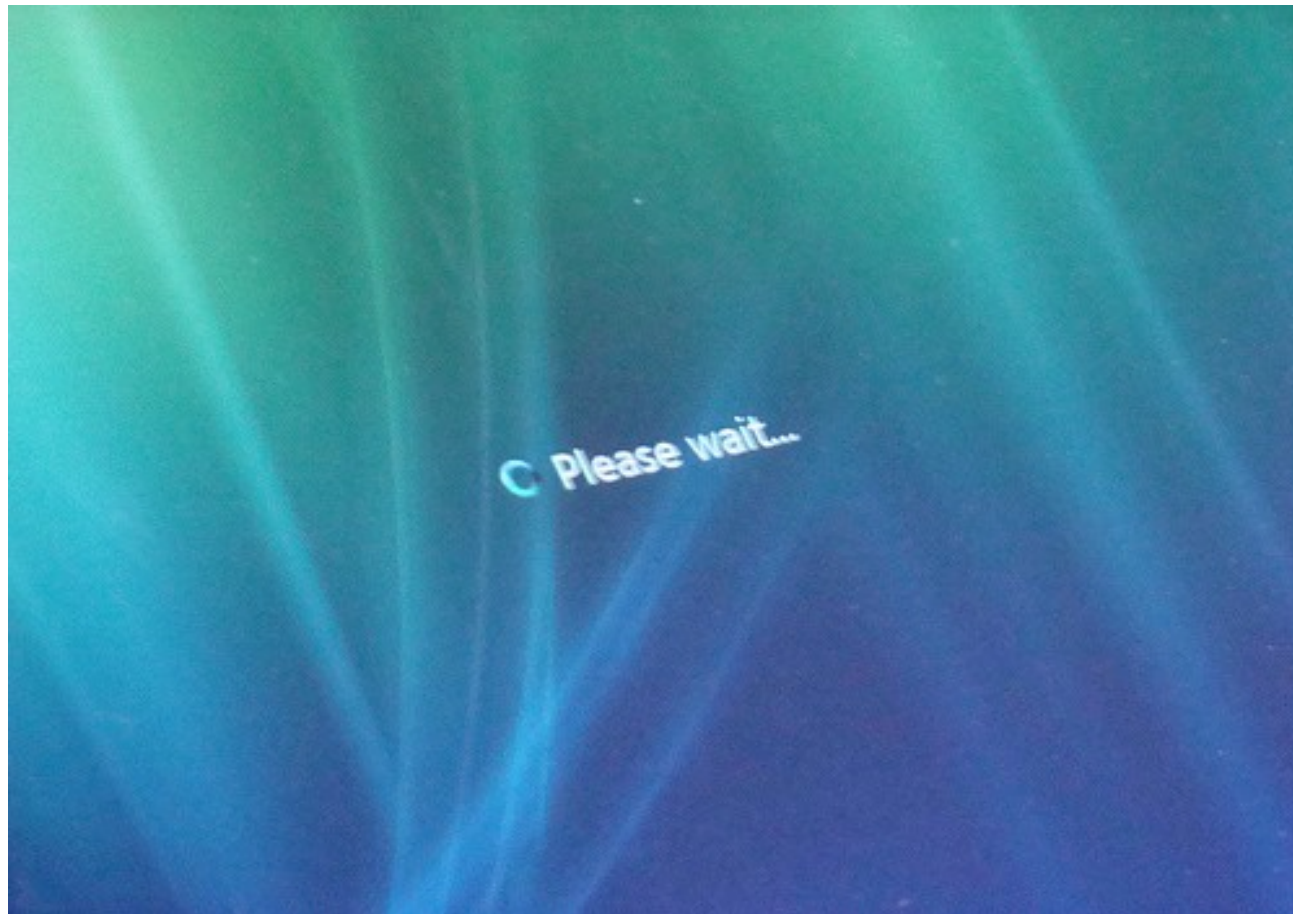
Technical impacts

- A natural unit of distribution
- A natural unit for transactions

Technical impacts

- A natural unit of distribution
- A natural unit for transactions
- A natural unit of synchronous work

Hint #1: If you don't aggregate enough, even the fastest computers will spin their wheels in vain













Latency

Aggregate is a meaningful whole...

So whoever needs a part of it
probably needs it whole

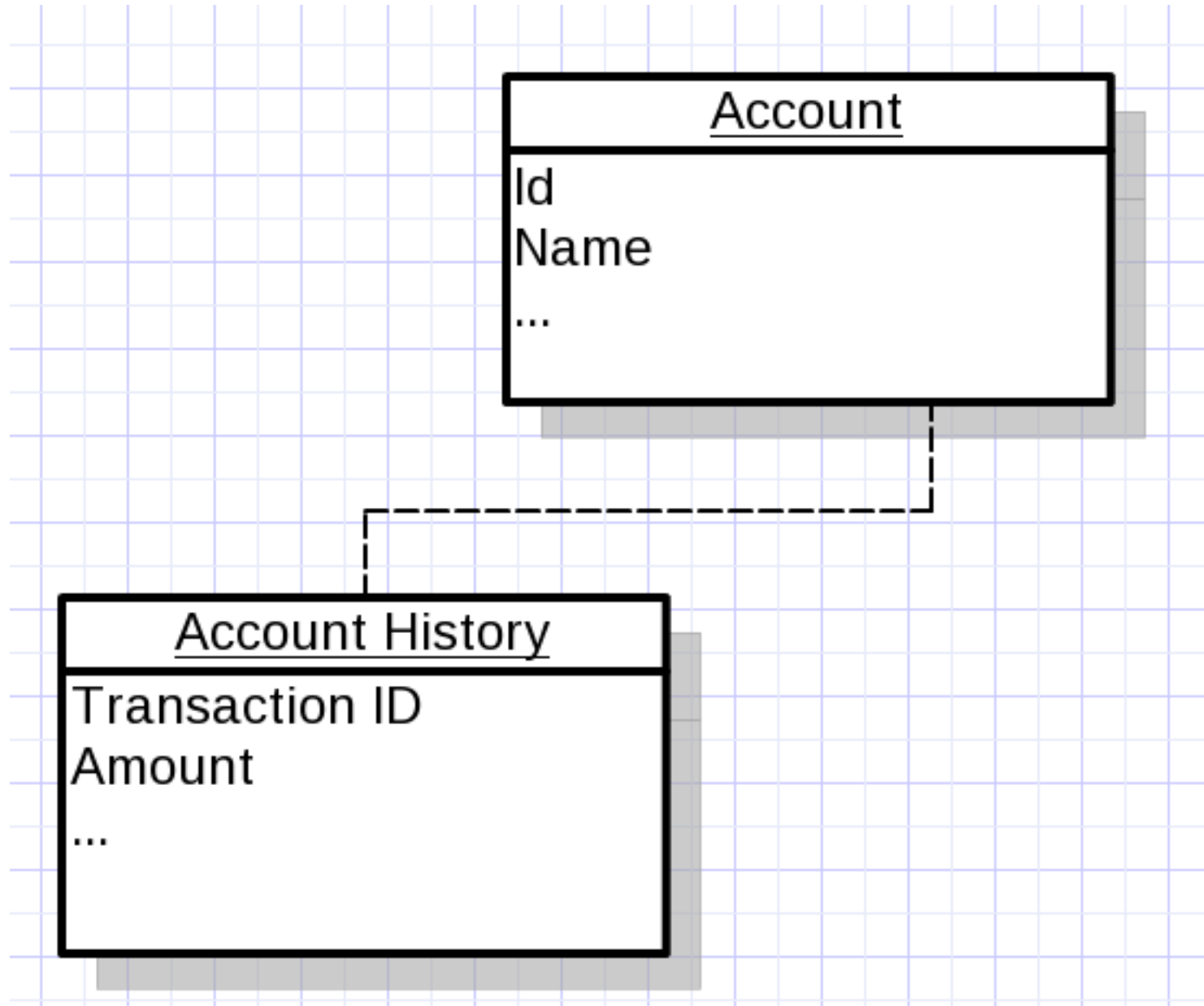
Ship entire aggregates straight
away to avoid latency

Hint #2: If you aggregate too much,
baggage becomes a problem



<http://www.flickr.com/photos/12496504@N06>

What is the boundary here?



If you don't need it all, it is probably
not an aggregate

Hint #3: If you don't aggregate enough, things are going to be broken just when you need them



<http://www.flickr.com/photos/nowhere77/>

Aggregates are units of consistency...

so aggregate updates should obey the
samurai principle:

Either return successfully or not at all

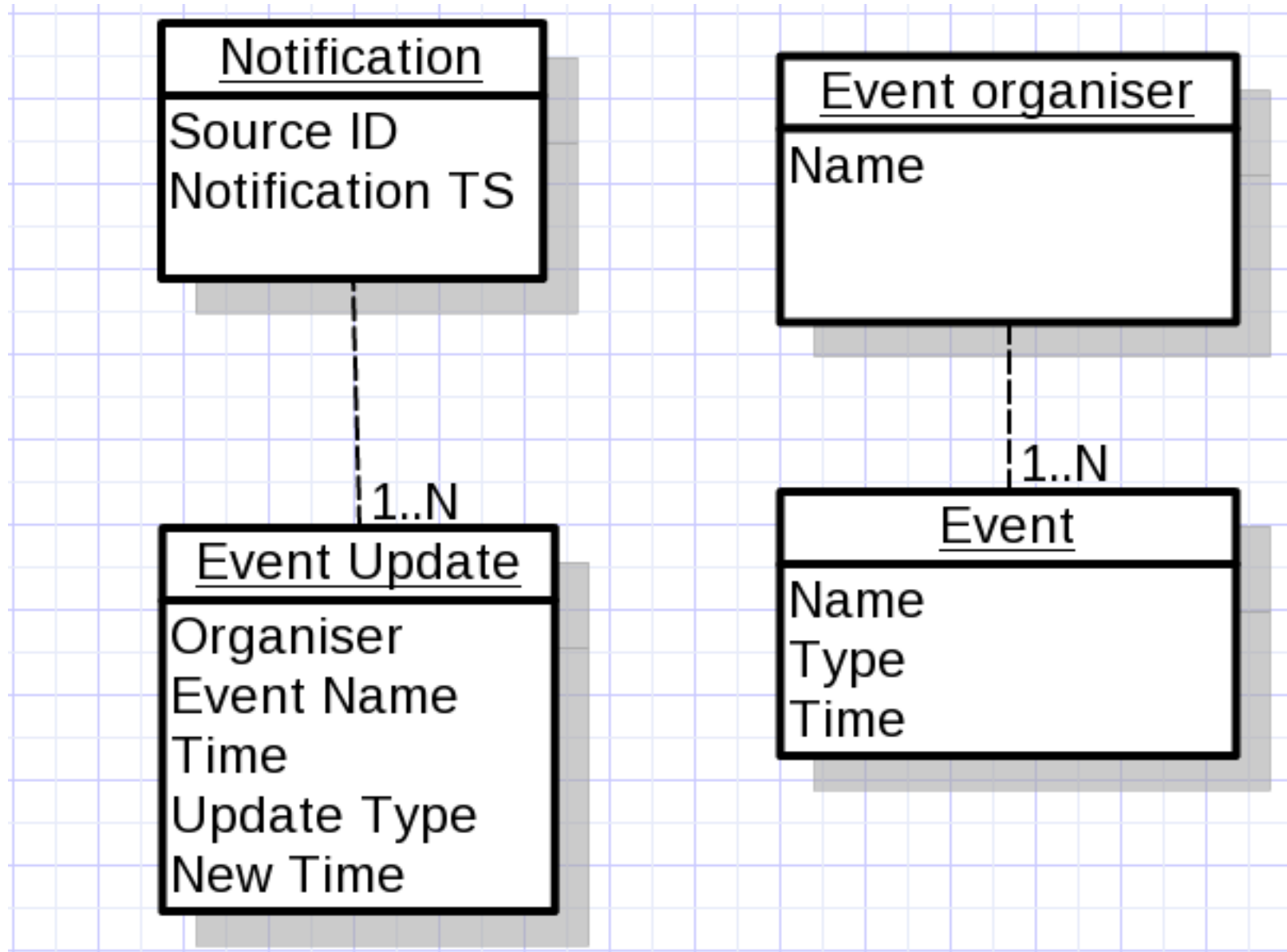
An aggregate should be processed
synchronously within a transaction

Hint #4: If you aggregate wrong things, deadlocks galore



<http://www.flickr.com/photos/meckleychina/>

What should we lock here?



When we lock, we should really lock
aggregates

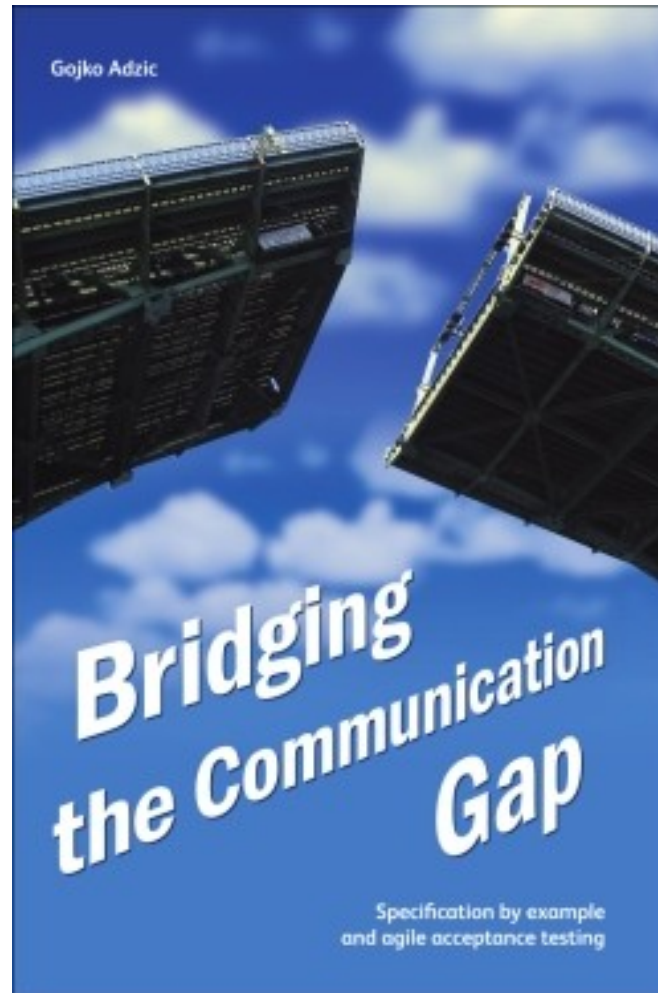
The technical and the business side
of aggregates confirm and reinforce
each-other

Spotting conflicts between them
points to problems straight away!

Unless these are resolved, the
system is just not going to work

Summary

- Aggregates are a business concept – a unit of consistency and a meaningful whole
- Technically they are units of distribution, locking and synchronicity
- A conflict between the two sides points to design issues
- Getting them just right is a key to making a nicely designed and performant system



gojko@gojko.com
<http://gojko.net>
@gojkoadzic