

coding
{the}
architecture



How much architecture is enough?

The way we **want to** and **have to** deliver software is constantly in **conflict**

In the **Waterfall**, we often
strive to be **more agile**

In **agile**, we often want
more architecture

We need to design for now,
but we can't ignore the
future

Scott Ambler tells us to...

“think about the
future, just wait
to act”

How much
thinking do we need to do?

What **level of detail**
is just enough?

How should we **share the**
architecture?

When is **the last**
responsible
moment to act?

How much architecture is
enough?

coding {the} architecture



Website

<http://www.codingthearchitecture.com>

London User Group

Monthly mix of presentation and discussion

Google Group

<http://groups.google.com/codingthearchitecture>