

Decay of Code

The quality of a piece of code doesn't stay fixed

It changes as new features are added

A structure that seemed good at the start

Soon becomes overly stressed as system expands

Quality of code quickly decays as it evolves

You can only bend it so much, before it breaks...





A38

e

Tamar Bridge Cycle Path

Tamar Bridge Cycle Path

Refactoring

It is OK to adapt the design of your code over time
You can rewrite, rearrange and reorganise things
This is often called "refactoring"

Refactoring takes much effort (for no functional gain)
The trick is to do it only when it pays off
(When your code is looking ugly !)

Try not to wait too long !
Large-scale refactoring is difficult and painful

Hints and Tips for Refactoring

When a method gets too long or complex...
...split into sub methods

If you see a lot of duplication...
...factor code into versatile method & parameterise
(Remember the DRY code example from last time ?)

When a class gets too long...
...split it into two coherent classes

If you have a cyclic dependency...
...redistribute methods and responsibility