

Model, View, Controller

Lyrics and music by James Dempsey.

Model View, Model View, Model View Controller
MVC's the paradigm for factoring your code,
into functional segments so your brain does not explode.
To achieve reusability you gotta keep those boundaries clean,
Model on the one side, View on the other, the Controller's in between.

Model View ? It's got three layers like Oreos do.
Model View creamy Controller

Model objects represent your applications raison d' 꺄 e.
Custom classes that contain data logic and et cetra.
You create custom classes in your app's problem domain,
then you can choose to reuse them with all the views,
but the model objects stay the same.

You can model a throttle in a manifold,
Model level two year old.
Model a bottle of fine Chardonnay.
Model all the twaddle stuff people say.
Model the coddle in a boiling eggs.
Model the waddle in Hexley's legs.

One, two, three, four.
Model View ? You can model all the models that pose for GQ.
Model View Controller

View objects tend to be controls that view and edit,
Cocoa's got a lot of those, well written to its credit.
Take an UITextView, hand it any old Unicode string,
the user interacts with it, it can hold most anything.
But the view don't knows about the Model:
That string could be a phone number or the words of Aristotle.
Keep the coupling loose and so achieve a massive level of reuse.

Model View ? All rendered very nicely in Aqua blue
Model View Controller

You're probably wondering now.
You're probably wondering how,
the data flows between Model and View.
The Controller has to mediate,
between each layer's changing state,

to synchronize the data of the two.
It pulls and pushes every changed value.
Yeah.

Model View ? mad props to the smalltalk crew!
for Model View Controller

Model View ? it's pronounced Oh Oh not Uh Uh
Model View Controller

There's a bit more on this story,
a few more miles upon this road,
well nobody seems to get much glory
writing controller code.
Well the model is mission critical
and gorgeous is the view,
But I'm not being lazy, but sometimes it's just crazy
how much code i write is just glue.
And it wouldn't be so tragic,
but the code ain't doing magic:
it's just moving values through.
And I wish I had a dime
for every single time
I set a TextField's stringValue.

Model View ? how we're gonna deep?six all that glue
Model View Controller

Controller's know the Model and View very
uuhh ? intimately
They often are hardcoding
which is very verboten for reusability.
But now you can connect any value you select
to any view property.
And I think you'll start binding,
then you'll be finding less code in your source tree.
Yeah I know I was astounded,
that's not even a rhyme.

But I think it bares repeating
all the code you won't be needing,
when you hook it up in IB.

Model View ? it even handles multiple selections too
Model View Controller

Model View ? hope I get my G5 before you

Model View Controller

Yeah, yeah, yeah. Yeah.

中译（蔡学镛）：

模型，视图，控制器

词曲：James Dempsey

MVC 是一种范型

它构造代码成为功能段，免得你脑袋淤阻

为达到复用，你必须让边界干净

这边是模型，那边是视图，控制器在中间

模型 视图，和夹心饼干一样有三层

模型 视图 控制器

模型 视图 模型 视图 模型 视图 控制器

模型对象正是你的应用系统存在的理由

你设计的对象，包含了数据、逻辑，之类的

在你的应用问题域，你创建定制类

你可以选择复用所有视图

但模型对象无须改变

你可以建模一部机器，随便什么机器。

建模一个三岁小孩

建模一瓶白葡萄酒

建模人们的窃窃私语

建模一些水煮蛋

建模 Hexley 的蹒跚步履

模型 视图

你可以建模 GQ 时尚杂志中的模特儿。

模型 视图 控制器

视图对象通常是控件，用来显示和编辑

Cocoa 也是这样，设计良好，赢得好评

Java 也是

把任何老的 Unicode 字符串交给 NSString

用户可以和它交互，它几乎可以包含任何东西

但视图不知道模型

字符串可以是一个电话号码，或者亚里士多德的文学作品

保持松耦合

达到最高的复用。

模型 视图，一切都是水波荡漾的蓝色

模型 视图 控制器

你可能怀疑现在做吗？

你可能怀疑怎么做到？
模型和视图之间的数据流动
是由控制器居间协调进行
两者之间状态的变化，数据的同步
都是由控制器控制的。

控制器负责将每个改变的状态送进送出
模型 视图，对于 Smalltalk 的人来说
这是最大的支柱
模型 视图 控制器

模型 视图，读做「噢噢」，不是「呜呜」
模型 视图 控制器

旅程尚未结束
前方还有道路
编写控制器的人
似乎没有得到掌声

模型的使命很重要
视图的外观很美妙
我或许很懒，但有时却是疯了
我写了多少代码，只是为了黏着两者
其实并不是惨痛
说穿了代码并没有神奇之处
只是用来搬动值罢了

我无意出言恐吓
但这是有声誉的
对于控制器你照做就是了

我真希望能得到一个铜板的奖赏
每次将字符串
送给 TextField 时

模型 视图
我们要如何丢弃黏结
模型 视图 控制器

控制器相当熟悉模型和视图
所以常常硬编码会妨碍复用
你可以将模型的键连结到任何视图的属性

一旦开始绑定
你会发现源码变少了。

是的，这一切自动又免费，让我感到洋洋得意

我知道一旦你使用接口设计工具
许多代码都可以自动产生
省下许多功夫

模型 视图，好处多多
模型 视图 控制器

模型 视图,但是我的应用已经交付
来不及采用 MVC 了
模型 视图 控制器

译注 1 Cocoa 是 MacOS X 的面向对象 API。

译注 2 这是 Cocoa 中的一个 Class。

译注 3 这是 MacOS X 的用户界面 Aqua 预定的颜色。