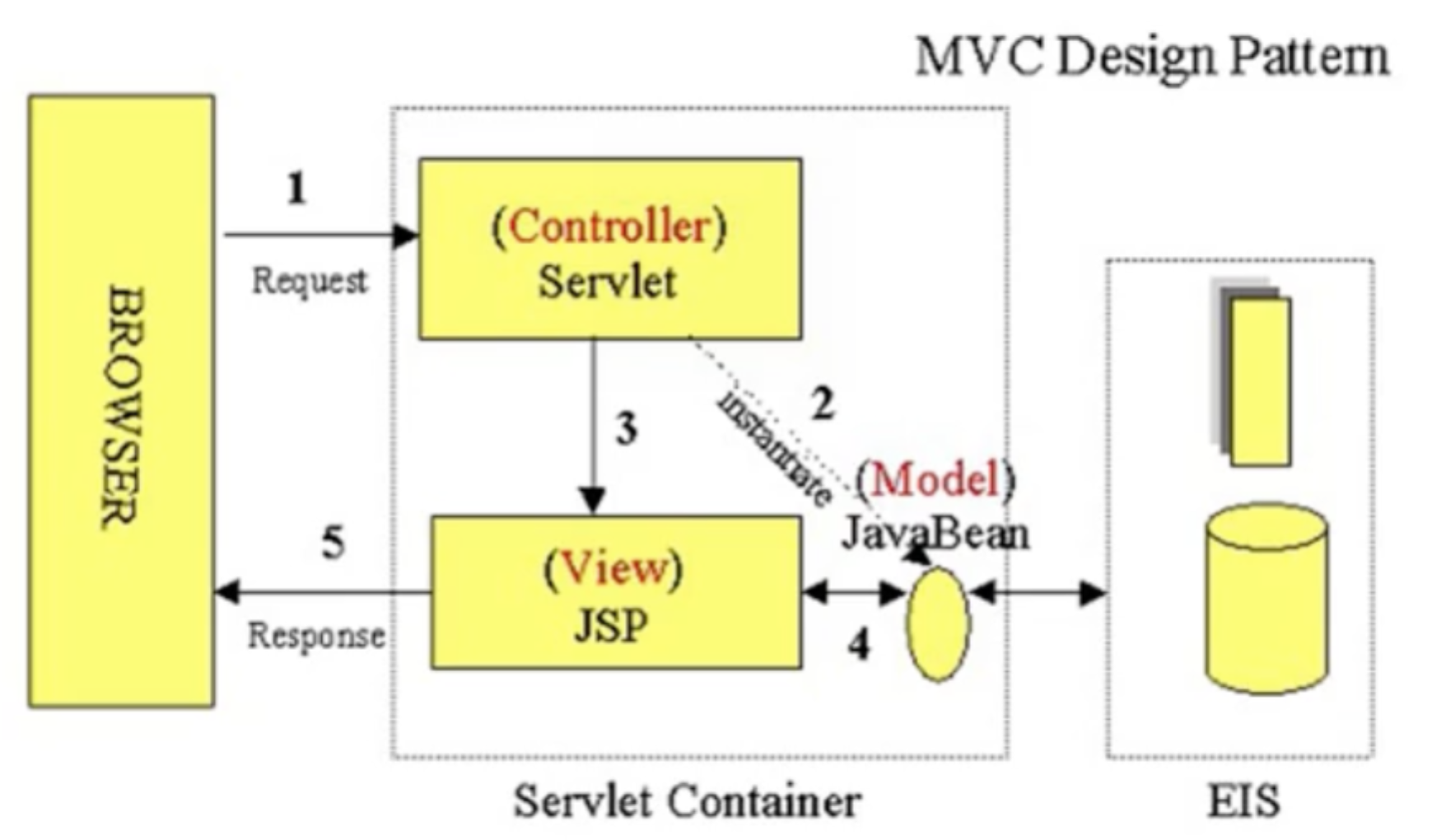
MVC模式是WEB开发或者说软件开发中一种非常流行的设计思想和软件架构模式。旨在将软件系统分为三个基本部分(或者说分为三层)：模型(Model), 视图(View)和控制器(Controller). MVC模式实现了逻辑(后台)和视图(前台)的分离，方便了以后的维护，提高开发速度。



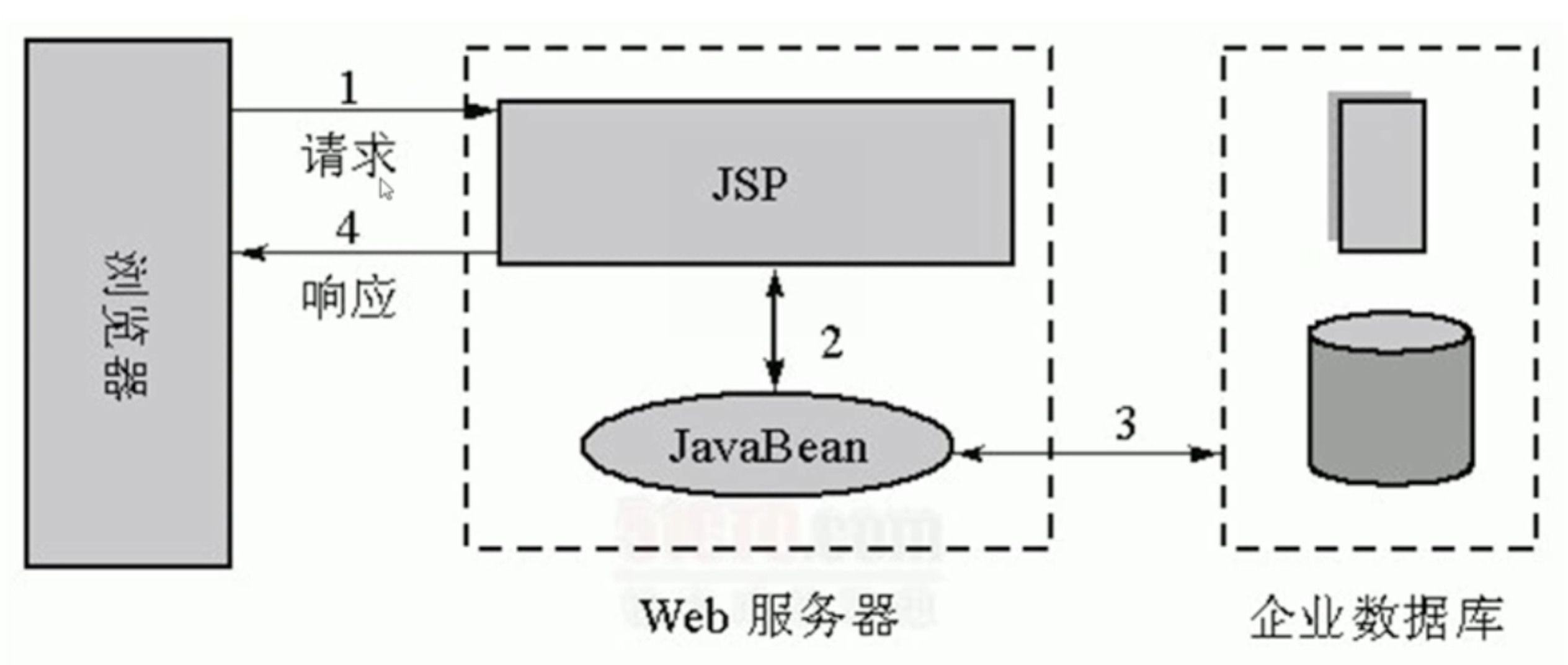
上图是MVC的一个结构图，以WEB开发为例。浏览器发出一个请求，充当Controller控制器角色的是Servlet负责接收请求；然后由Servlet实例化模型Model层的对象，一般由JavaBeans充当该角色，由JavaBeans再去调用最右的数据库层, Enterprise Information System企业信息系统，得到的数据可以返回给控制器controller, 也可以通过视图View层来访问这些数据；充当View角色的是JSP页面。控制层Servlet会根据反馈回来的结果来给用户呈现不同的响应页面，也即跳转到不同的JSP页面呈现不同的视图View.

注意：模型Model层JavaBeans对象访问数据库的过程，需要依靠JDBC技术。

Model1介绍 ( JSP + JavaBeans )

Model1是Java WEB开发中较早出现的一个开发模式，在model1模型出现前，整个WEB应用的情况：几乎全部由JSP页面组成，JSP页面接受处理客户端请求，对请求处理后直接做出响应。弊端：在界面层充斥着大量的业务逻辑的代码和数据访问层的代码，Web程序的可扩展性和可维护性非常差。

当JavaBeans出现后，使得JSP页面中使用JavaBeans封装的数据或者调用JavaBeans的业务逻辑功能代码，这样大大提升了程序的可维护性，诞生了Model1模式。如下图所示：

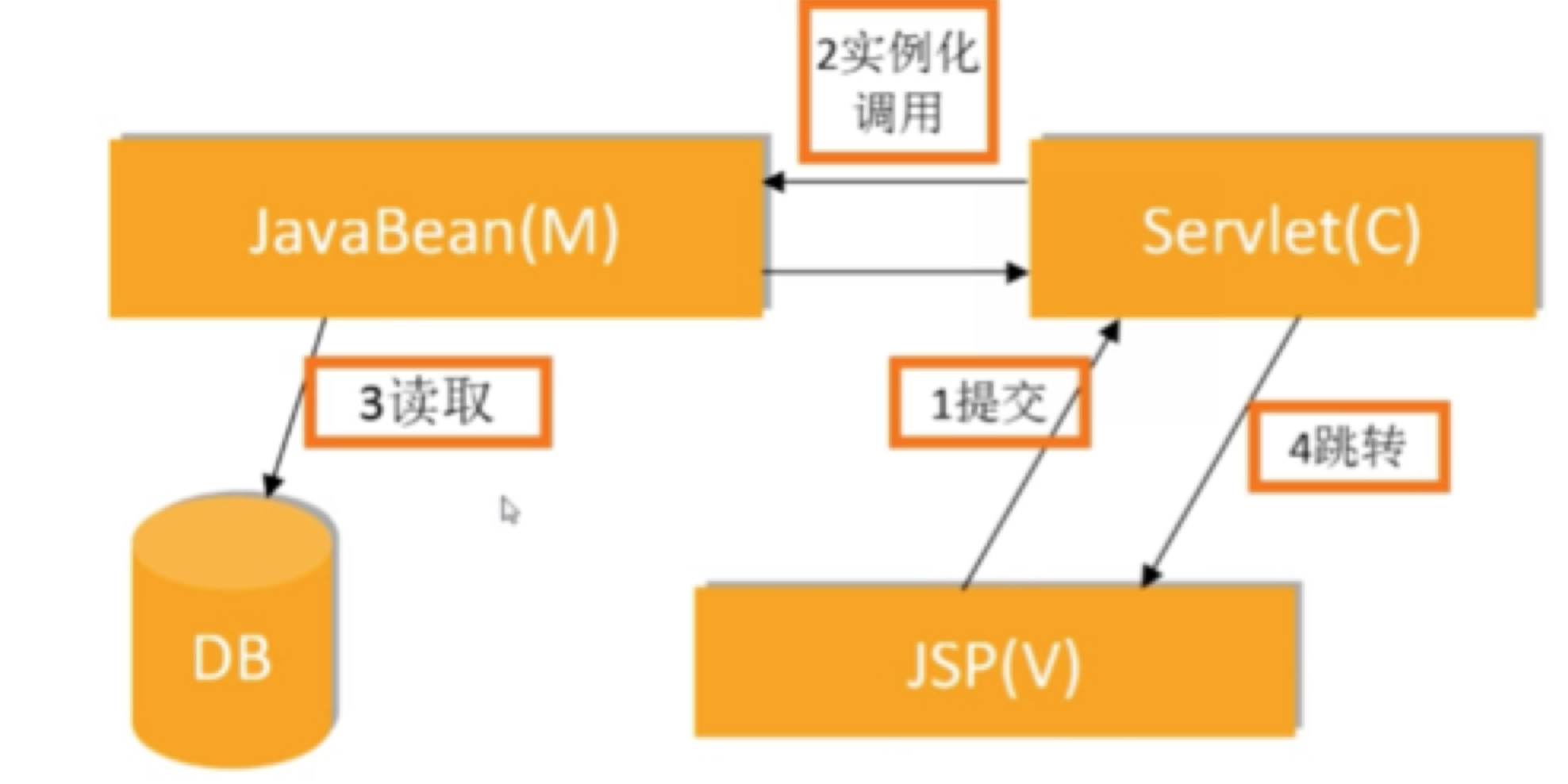


JavaBeans是符合某种特定规范的Java类，是Java中一种特殊的类。使用JavaBeans的好处是解决代码重复编写，减少代码冗余，功能区分明确，提高了代码的维护性。JavaBeans既可以封装数据，也可以封装业务逻辑功能。

Model1模式已经初步具备了分层的思想。

Model2介绍

Java Web开发的Model2模型就是MVC模式的进一步体现。



首先由视图层JSP页面提交一个请求给控制层的Servlet, 再由Servlet去实例化调用模型层的JavaBeans对象，然后由模型层去访问数据库层读取数据，再反馈给控制层，控制层Servlet会根据反馈回来的结果来给用户呈现不同的响应页面，也即跳转到不同的JSP页面。