一般情况下DataType通过creatingType属性与一个POJO建立映射关系。其中DataType下的每一个PropertyDef都与POJO中的一个属性对应。

有时候我们希望可以添加一个PropertyDef，而这个PropertyDef并不与对应POJO中任何一个属性相关。这样好比为了开发的需要我们希望创建一个VO或DTO对象。

在DataType中新建虚拟属性，只要在内部添加一个PropertyDef就可以。

由于我们不会实际在对应的POJO中添加这个虚拟属性，那么通过Java代码完成虚拟属性的存取就有必要通过动态代理技术。在Dorado7中提供了专门的工具类协助我们虚拟属性的设定：

Product targetProduct = EntityUtils.toEntity(product);

EntityUtils.setValue(targetProduct, "prop1",

RandomStringUtils.randomAlphanumeric(8));

首先通过EntityUtils的toEntity方法可以将product对象转换为一个动态代理对象targetProduct，并通过EntityUtils工具完成对动态代理对象的虚拟属性“prop1”的赋值。

虚拟属性的值的获取：

String prop1 = EntityUtils.getString(targetProduct, "prop1");