1. Error

一种是纯粹的error定义，常用语程序运行时发生的错误（程序运行错误才触发error）。一般定义在define/error

另外一种是包含了rule和error，常用语对输入值进行判断（即违反了rule才触发error）。一般定义在define/validateRule。

Mongodb的validate的rule和error和第二种类似（因为要存入db的数据大部分来自input value），所以不单独定义一个文件，而是和第二种复用，直接写一个新的error rc和msg即可。

1. Error/msg

除了validateRule的msg是由validate自动生成的，其他的msg都和rc定义在一起

3. input rule

inputRule是基于mongo structure手工产生的，只记录client端会修改并保存到db的field（所以需要定义rule来检查），定义完成后，用作

a. 用作server端的对client的输入进行检查

b. 用作产生mongo对应的field的validator（有些field是node产生，默认正确，不用validator）

c. 用作产生客户端的rule和inputAttr

mongo自动产生的ObjectId，无需加入

server自动生成的字段，例如salt,cDate,uDate，无需加入~~，但是无需检测因为不会调用checkInput进行检查。通过generateInternal判断。~~

~~无需检测的字段，可能需要返回给client，通过showInClient判断，是否需要加入到inputAttr。ClientRule还是通过generateInternal判断。~~

从server返回的数据，直接加入recorder，而不是inputAttr。

inputAttr：只存储在client可以进行修改的字段（即generateInternal=false的字段）

clientInputRule：同上，只存储cleint可以修改字段的rule

1. Model

对于create/update，Model只负责最基本的db操作。Populate的操作放在controller中完成（因为CU只对一个doc进行操作，所以可以通过函数判断此doc是否需要populate）

对于readAlL操作，populate直接放在model，因为此时可能有多个doc，无法单独判断单个doc是否需要populate，所以直接populate

5 update

如果需要删除一个外键，设为null，会通过函数miscFunc.constructUpdateCriteria将设为null的字段放入{$unset:{}}中，以便通知mongodb执行删除操作

1. CRUD说明

Create：需要通过函数来判断是否需要populate（因为只生成一个doc，可以单独进行判断）

Read：总是populate，因为可能有多个doc，无法逐个判断是否需要populate。同时，需要通过{dDate:{$exists:false}}来排除被删除的文档

Update：同create，需要通过函数判断是否需要populate

Delete：更新dDate字段（而不是真正的删除）