**函数耗时监测：**

开关：

可通过dictator后门或者gm指令进行开关，非生产环境默认开启，每隔30分钟会输出一份报告，相关开关：

ctrl\_measure

dump\_measure

报告：

输出到lua\_measure.out．PROFILE报告按耗时和次数排序输出，WASTE报告输出耗时超出200ms的单次函数按时间戳排序

注意：

１．函数监测本质上是给每个执行函数加上hook，对效率影响颇大，暂时决定只在非生产环境开启

２．服务器启动后频繁关闭和开启，会导致开启和关闭时间段内部分报告不够精准或者丢失

3．由于每个函数首尾加上hook之后大概会增加0.5-2ms不等的耗时，在开启measure模块的情况下测出的耗时会有所偏差（尤其是对于某些含有大量函数的函数体，偏差的耗时会随着函数数目线性增长）．所以，建议将measure的输出报告作为项目热点分析的参考，而不是完全相信输出耗时，在确定热点之后，可以在关闭measure的前提下，使用start/stop来对相关代码段进行测时．

待处理：

１．输出格式优化

２．无用信息筛选

３．尾递归/报错行为监测

代码段手动测时：

可使用measure.start和measure.stop接口，兼容中间协程挂起情况：

