一、从lambda说起，编译器如何完成仿函数

二、再说coroutine，编译器可能如何完成协程类

三、resume function的两种形式：yield+await

四、编译器实现await的细节

五、一个await的范例

六、一个mysql+callback的范例

七、通用的绿化callback

八、具体绿化mysql+callback

九、一个message box的范例

十、asio+callback的传统应用

十一、resume function如何与asio系统工作

十二、总结resume function要解决的痛点

十三、对比三种协程的优缺点

十四、librf的协程调度

十五、librf如何处理异常

十六、librf如何处理多线程

十七、librf之sleep()

十八、librf之mutex

十九、librf之channel

<https://github.com/tearshark/resumef>

演讲议题形式  
演讲时长：40 分钟，包括提问环节  
演讲内容标准  
主题：申请者需要有明确的主题观点，内容新颖，有深度，能够帮助听众在知识、实操等方面有所启发。  
经历：申请者在相关领域的成就以及从业经历，我们也会作为评审的参考标准。  
感受：我们非常重视听众在参加演讲分享过后的收获，这同时也是举办技术会议的重要意义。

有什么特色吗  
你可以深入讲一下，对比一下  
做这个的动机

Resumef+asio

Resumef+messageBox

Resumef+net应答

Resumef+异步阻塞操作

Resumef实现概要

Resumef+提前准备好该怎么做

Resumef+协程

优化协程调度

asio 服务器，使用lambda完成do\_something\_1/2/3，  
asio 服务器，使用rf完成do\_something\_1/2/3  
cocos2d-x 客户端，使用rf完成msgbox  
多线程下的rf  
librf解决什么  
librf如何处理成协程  
librf如何处理异常  
librf如何处理多线程  
librf如何处理先完成/后调用的情况《？》  
librf如何实现mutex

对现有callback代码的保护

跟第三方异步代码的集成难易度