# 感知与分析思路

## 监控数据分析

目前的研究，主要是进行分层，分别对基础设施（CPU和内存资源）及应用（服务之间的调用以及外部需求）的数据进行监控。

我们目前无法想到还要去监控什么数据，现在的论文已经涉及到了需求了，当外部需求改变了，对系统做自适应。底层资源的监控也就是那些。我们在这个新的地方，想不到新的思路。

## 监控数据联系

目前很多自适应框架都是分层处理的，对于数据之间的联系是几乎没有的。所以我们考虑能不能按照原先的自适应框架，通过底层事件去预测上层事件产生的概率。

## 定制化感知

这个地方，可以设置权重进行处理。通过建立相关的映射，去判断哪个监控数据对于系统本身更重要。从而实现定制化感知。

## 现状撰写问题

我们今天尝试了写了一下现状，但是发现存在了几个问题，让我们难以自洽。

首先是目前的自适应应用于微服务的论文，描写的数据监控都是很简单的。要么是直接分层监控，要么就是去中心化和中心化的结合。我们参考了原先自然基金写现状分析的写法。

杨启亮团队结合可视化模型和形式化建模方法，基于UML建立自适应软件行为模型并将其转换为时间自动机网络，通过模型检验工具对软件行为进行有效性评估，主要对系统资源约束、软件行为有效性、自适应策略的正确性以及响应时间进行有效性评估。

可以看见基本要阐述一篇论文所用的方法，但是目前我们所看的论文对于感知都是简单的阐述，直接说监控业务数据和底层资源，对其感知的方法几乎没有阐述。这样我们写的文档几乎都没有**实际的方法，显得十分空洞**。

我们初步想考虑的写法如下：

Adalberto团队在每一个微服中添加了监控协议栈，通过构建模型中间层来监控微服务资源使用、日志及消息。Peini团队通过分层的自适应框架，监控基础设施控制层的CPU和内存资源及应用控制层的微服务调用关系。

由于这篇论文仅仅对感知阐述了一点，就马上过渡到了后面的阶段（大部分论文也是采取的这种方式）。我们很难在现状研究中总结出一个**具体的方法**来写。而且感知是与整个自适应系统框架紧密结合起来的，我们如果阐述这部分框架的内容，感觉**会与系统建模重叠**。

目前的问题就是感觉无法具体，写出来的现状没有具体的方法。（因为这些论文本身也没有阐述。）