

Monitor 网络智能运维系统

设计文档

所在赛道和赛项：*B-EPI*

目录

一、意义价值与目标问题	1
1.1 作品意义价值	1
1.2 解决问题	2
1.3 实现的目标与基本功能	3
1.4 作品应用领域	4
二、设计思路与方案	4
2.1 设计思路与技术路线	4
2.1.1应用架构设计	5
2.1.2技术架构设计	5
2.2 详细的设计方案	6
2.2.1可视化界面设计	6
2.2.2告警设计	8
三、方案实现	8
3.1 作品实施方案	8
3.1.1动态阈值算法	8
3.1.2基于 Flask 的数据获取	10
3.2 具体功能	11
四、运行结果	12
4.1 整体界面展示	12
4.2 主要功能展示	13
4.2.1数据计算与显示模块	13
4.2.2网络结构与报表模块	14
五、创新与特色	16
5.1 创新点	16
5.2 特色点	16

一、意义价值与目标问题

1.1 作品意义价值

随着经济社会的发展，社会信息化建设也取得了长足的进步。社会信息化就是计算机信息处理和传输的广泛应用为标记的新技术革命，改造社会生活方式的过程。在我国，根据最新公布的国家信息化发展战略，信息化是充分利用信息技术，开发利用信息资源，促进信息交流和信息共享，提高经济增长质量，推送经济社会发展转型的历史过程。

信息化建设离不开信息系统的建设，而信息系统正常运行离不开运维管理。信息系统运维管理是信息化建设的重要组成部分，是指采用专业的技术和方法，对信息系统的软硬件环境、计算机网络、应用系统等进行综合管理；目的是保证系统和网络的正常运行，进一步提高运行效率和服务质量。

很多企业中，网络运维还是在用非常原始的方法，通过 EXCEL 记录维护资产信息，没有预见性，等到设备或者网络出现故障后才进行事后处理补救，通过多个工具查看或者获取数据，数据单一分散没法很好的汇总和高价值的展现等。运维人员不仅工作内容单调重复、压力大，运维人员也随着 IT 系统的不断增加而膨胀，无法有效控制，并且运维一直处理补漏的状态。据有关估计，运维工作量占了整个信息服务部门整体工作量的 80 % 左右。如图 1 所示：

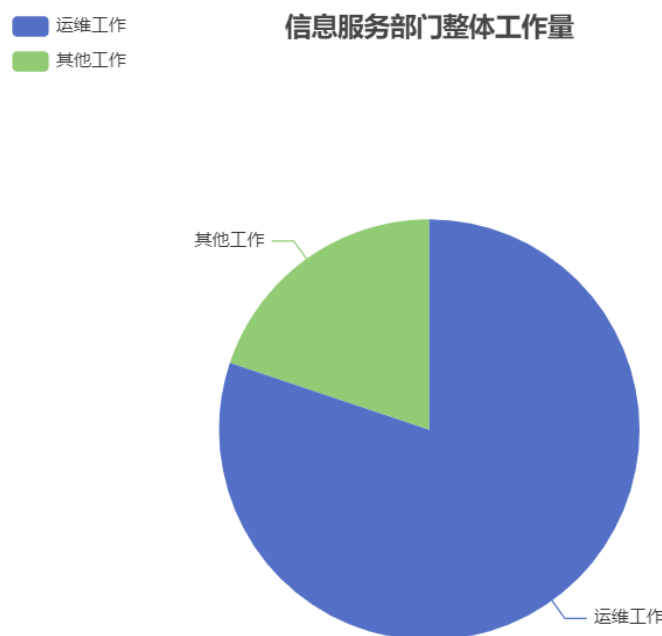


图 1 信息服务部门整体工作量

有效的网络运维系统具有性能稳定、界面友好、及时监控和数据有效汇总等特点，

可极大的简化系统的运维管理，主动及时获取计算机设备及其它网络设备的运行状态，发现业务系统的隐患，归并整合分析数据，展现易于理解和更符合场景的数据报表供决策分析。特别是运维系统可视化，是网络运维重要的发展方向。

运维可视化，核心是将所运维的服务、资源、设备的状态和正在发生的事件通过可视化的手段呈现出来，指导运维人员或者产品研发人员做出正确的运维决策。某种程度上，运维与可视化相辅相成，可视化程度越高，运维就越简单，运维效率也就越高。

1.2 解决问题

本系统解决主要有如下的问题：

(1) 用户体验差：运维系统往往偏向于专业人员使用，因此已往的系统设计中非常强调功能性，在用户体验方面欠缺考虑；

(2) 表现手段单一，效果普通：传统的报表依然是主要的展现方式，维度不丰富，受到报表维度的限制，有些数据没法有效的结合起来展现，无法充分体现出的数据的价值；作为外部业务部门受专业所限难以理解，使得维护成了运维部门自己的事情；

(3) 维护没有预见性：运维基本上的 IT 网络运维都是被动补救式的，只有当问题发生时才能进行处理。直接导致运维人员工作很忙碌却一直从事重复性的劳动，导致工作枯燥运维人员流动性较高，这些传统式被动、孤立、半自动式的 IT 运维管理模式经常让 IT 部门疲惫不堪；

(4) 运维数据提炼不够：随着 IT 系统越来越多，运维也变得越来越复杂，往往一个企业内部都存在多个运维系统，各自处理对应的领域，没有很好的交集，获取的数据较为分散，不同的系统获取各自的数据，没有将数据很好的汇总综合起来。

(5) 运维系统自动化程度不够：很多检查和设置操作都需要手工进行，而这些工作基本上每天都需要做，当网络规模大设备多的情况下工作量就很大。

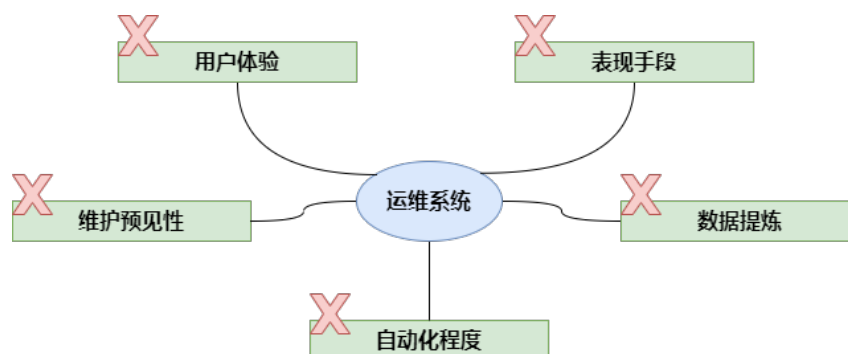


图 2 运维系统存在问题

1.3 实现的目标与基本功能

针对上述提到的问题，我们探究了信息系统运维中可视化方案 and 技术的实现，以及基于信息设备历史数据自动计算预警值技术的研究和功能的实现；这两个领域都是当前运维技术发展的主要方向之一，也和客户的需求密切相关，在实际的运维管理中能起动非常重要的作用，具有很高的社会效益和经济效益。总结起来，实现的目标与基本功能主要包括以下内容：

(1) 运维可视的展现效果设计和实现。运维可视的关键点就在于可视化的展现效果和展现能力，展现能力是指是否可以将各种运维数据进行有机融合展现，而不仅仅是各自通过一个个单独的页面进行展现。界面的大屏投影展现直接体现的可视效果的好坏，好的展现效果可以给整个运维系统添色；

(2) 展现效果自定义方案设计和实现。信息系统的运维可视和用户的实际场景是密切相关的，比如模拟用户机房的实际场景等，因为不存在设计一个界面可以满足所有用户的可能，必须提供简单、高效的自定义功能，让用户可以按照实际情况自己设计展现界面及效果；

(3) 可视化数据的处理机制的研究和实现。企业及政府单位的信息系统运维管理往往存在以下情况，不同的领域由不同的厂家的运维系统支撑，包括告警等也各自发送处理，数据分散，好的可视化功能需要能够对其他系统的运维数据也有效，并且要有能力进行有机的融合展现；

(4) 自动计算预警阈值的算法的研究。通过对信息设备运行的历史数据进行分析，获取运行规则，研究算法由此得出设备未来的运行阈值，该阈值是动态的，不同的时间点阈值不同，为后面的自动预警规则提供基础；

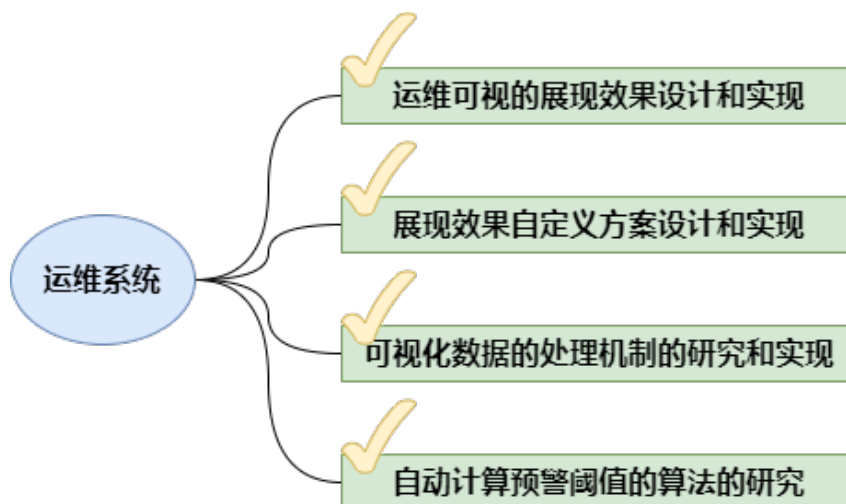


图3 本系统解决的主要问题

1.4 作品应用领域

IT(Information Technology) 运维和信息系统的建设几乎是同时产生，相辅相成的。早期的 IT 运维很简单，就是登记信息、保证网络及设备的正常运行。工作原理就是对网络设备进行监控，获取运行数据，当网络中设备出现故障时，及时发现问题并定位故障设备，方便运维人员对故障进行处理。

随着 IT 系统的不断发展成熟，企业对 IT 运维的要求也在提高，开始关注整个业务系统的运维，需要运维系统可以落实保障业务系统正常运作的各个环节。为了让 IT 管理更加规范和有效，CCTA (英国国家计算机和电信局) 将英国各个行业在 IT 管理方面的最佳实践归纳起来变成规范，这个规范就是 ITIL。ITIL 为企业的 IT 服务管理实践提供了一个客观、严谨、可量化的标准和规范，很快在世界范围内各个行业都得到了广泛应用。结合 ITIL 规范，IT 运维系统满足了对 IT 资源的统一管理、降低运行成本、提高服务质量和运行效率，从而保障业务系统正常运行，体现了 IT 运维的价值。

运维可视化和自动化是 IT 运维的最新发展方向，越来越复杂的运维工作必须通过可视化来表达，同时自动化的实现又离不开可视化，才能保证所有人理解一致。海量运维数据的价值体现离不开良好的数据展现。国外做 IT 运维的主要是 IBM、HP、BMC、CA 等，主要占据高端的金融、电信等行业。比如 IBM 提出的“下一代运维”具有前瞻性、整合性、灵活性和高度友好性等特点，使用者操作非常方便。

我国真正意义上的 IT 运维管理与 ITIL 概念一起在 2000 年左右进入中国。之后逐步走向模块化，开始与国外的产品同步发展。国内市场早期基本由国外企业主导，目前已经开始和国外企业抗衡，在本地化上占有明显优势。国内的 IT 运维也融合了 ITIL 的理念，在 IT 运维上实现了制度化，提高了运维的稳定性，实现了标准化。由于起步及整体信息化水平低等原因，国内运维与国外整体上还有差距，主要体现在简单的自动化没有启动应有的作用，不够重视可视化建设，被动补救式的运维没有形成主动的监控管理，目前 IT 治理手段欠缺和系统管理粗放是国内企业在 IT 运维管理领域面临的主要问题。合理利用 IT 资源、有效控制 IT 风险，成为了近几年 IT 运维的主要话题。

二、 设计思路与方案

2.1 设计思路与技术路线

本系统的设计分成两个层次，应用架构设计和技术架构设计。其中，应用架构关注系统功能实现，包括组件设计、功能集成等；技术架构为业务功能的实现提供支持，包含了各种开发技术。

2.1.1 应用架构设计

在应用架构的设计上，可以把系统分别在横向和纵向上划分模块，横向可以分为可视化模块和数据报告模块，这两个模块分别提供了图形化和数据化的显示，可视化模块能够让用户更加直观地观察数据变化，而数据报告模块则提供了精确的数据。纵向可以分为数据获取模块、业务功能模块、显示模块，各个模块之间通过接口来实现通信，提高了各个模块的独立性和可扩展性。应用架构设计如图 4 所示。

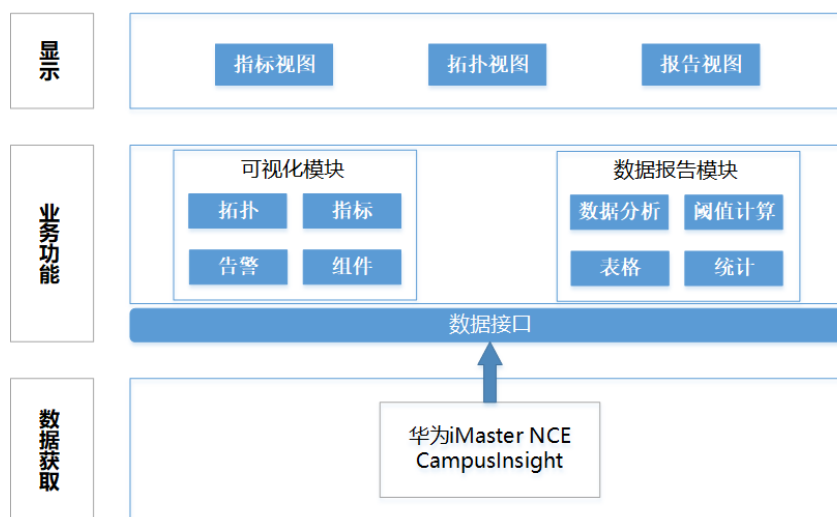


图 4 应用架构图

上图中纵向模块的内容为：

（1）数据获取模块：这一模块使用华为 iMaster NCE CampusInsight 提供的 API 接口获取园区网络的最近数据，并向上提供给业务功能模块。

（2）业务功能模块：业务功能是系统的核心部分，这一模块可以横向分为可视化和数据报告两大模块。其中，可视化模块包含拓扑图、网络性能指标、告警信息和组件的展现；数据报告模块负责对获取的数据进行分析处理，计算网络指标的安全阈值，统计网络数据超过安全范围的次数，最终以表格的形式呈现结果。

（3）显示模块：在视图对两大功能模块的业务功能进行展现，将各种数据汇总在 web 界面中，进行有序的管理。

2.1.2 技术架构设计

系统的开发用到的主要技术有 Python、Jquery、Restful、Ajax、Flask 等，开发技术架构如图 5 所示。

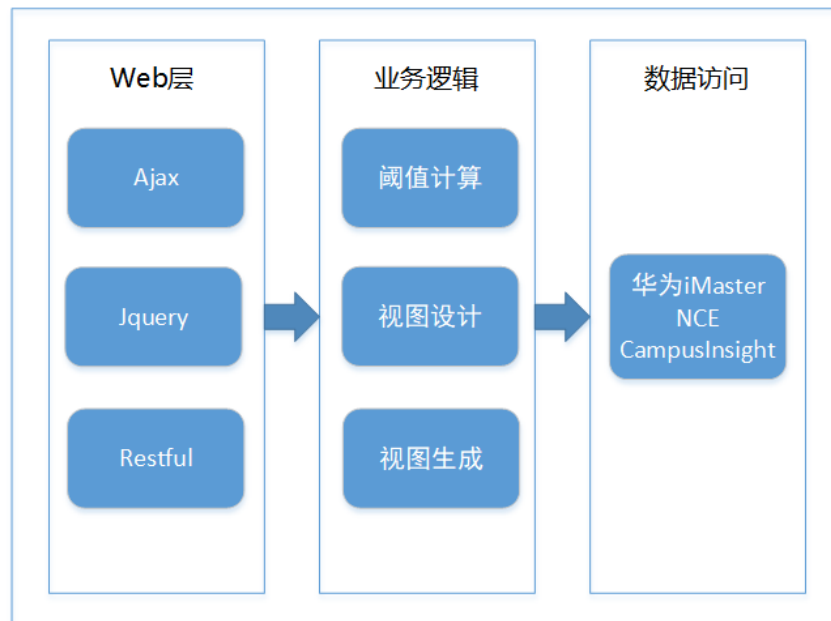


图 5 开发技术架构图

主要的开发技术为 Python，前端界面使用了 Jquery、Ajax 等技术，每个模块之间通过接口实现数据的传递。技术架构这样设计有以下好处：

- （1）可扩展性好，各个模块使用接口来实现互相通信，彼此之间独立性比较高，保证了各功能之间的低耦合性。
- （2）系统分层明确，便于后续的维护工作。
- （3）兼容性好，综合使用了传统的和新出现的开发技术，保证了本系统能较好地兼容现有的其他系统。

2.2 详细的设计方案

2.2.1 可视化界面设计

（1）性能指标视图设计：视图设计的一个很重要的因素就是要美观且清晰，本系统不同性能指标的视图设计遵循了界面设计的一致性原则，左边为菜单栏，主体部分为当前指标的数据，以折线图的方式展示，清晰直观地反映了数据的变化情况。右侧为当前指标突破安全阈值的次数，让人可以非常方便地了解当前网络指标的总体健康度。同时，将数据放到统计图中，选中的节点或线段亮度会发生改变，这种微交互可以提高使用的体验感。在页面的上部有一些常用指标的具体数值，给用户提供信息的同时也增加了美观度。设计的原型效果如图 6 所示。



图 6 性能指标界面原型

（2）拓扑图设计：拓扑图的主要作用就是展示设备之间的连接情况，如果手动绘制拓扑图将会花费大量的时间，本系统实现了根据终端和 AP 位置来自动绘制拓扑图的功能。具体的实现方法是：根据终端和 AP 的位置信息来绘制其在平面图中的位置，然后根据 AP 与终端的关联信息来绘制两者之间的连线。同时，用户将鼠标移动到终端和 AP 的位置上时，将会展示该终端或 AP 的基本信息，这样的设计让用户可以根据直观位置来查询某一设备的基本信息。拓扑图界面的原型设计如图 7 所示。



图 7 拓扑图界面原型

（3）渲染效果设计：渲染效果设计对可视化界面来说至关重要，因为渲染效果直接影响了界面的美观程度。我们在整体界面的设计中，利用了光晕、亮度、阴影、动画效果等来丰富界面的渲染效果。

2.2.2 告警设计

告警在网络运维中有着很重要的作用，本系统的告警以统计设备数据超过安全阈值的次数来实现。系统根据设备的历史数据来获取运行的规律，根据特定的算法来计算出每一个时间点对应的安全阈值，当设备的性能数据突破阈值时，会计入统计中，最终通过可视化平台反馈给用户，告警设计的示意图如下：

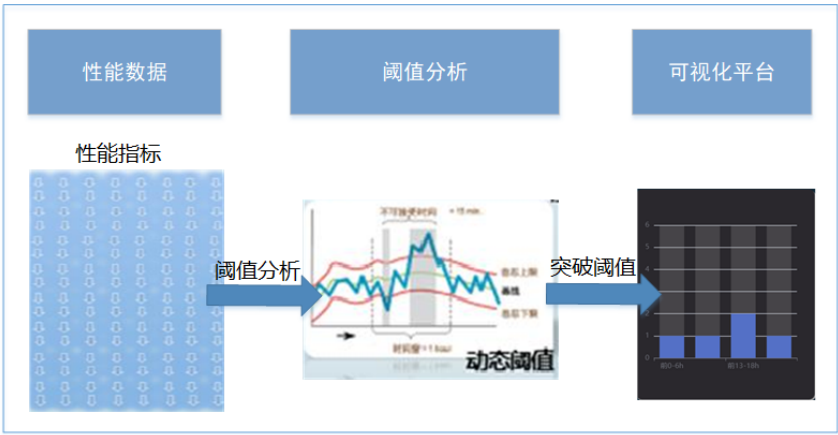


图 8 告警设计示意图

其中，性能数据通过华为 iMaster NCE CampusInsight API 来进行获取，然后，使用历史数据来动态进行阈值分析，从示意图中可以看到，随着数据的变化，阈值曲线也是不断发生变化的。而数据中超过阈值曲线的点将会被认为是不健康的数据，以条形图的形式在可视化平台中展示。

三、 方案实现

3.1 作品实施方案

3.1.1 动态阈值算法

在数据报告模块，我们需要提供网络质量的指标，核心功能就是动态阈值的算法。目前运用在这个方面的阈值计算方式有很多种，每一种方式有各自的优缺点和适用范围，但有一个共同点就是根据历史数据来进行阈值的计算，主要的计算方式有：

- （1）线性回归：根据历史数据来调整参数，这种方法计算比较简单方便，但是如果数据的波动大则变得不那么准确。
- （2）方差：对数据原始值的要求较高，一些超参数的选取会对结果产生显著影响。
- （3）正态分布：通过计算均值和标准差来预测上下阈值，这种方法需要数据近似地满足正态分布。

考虑到设备的使用频率在时间分布上是不均匀的，使用静态的阈值来进行判别具有很大的局限性，分析园区网络的使用特点，我们可以提出以下假设：

- (1) 不存在一些网络设备长期不使用，历史数据可靠。
- (2) 设备运行的指标数据具有周期性，且大致以天为变化周期。
- (3) 设备运行产生的数据是数字，而不是文字的形式。

根据以上假设，我们可以根据设备产生的历史数据来估计每一个时间点的阈值，认为在阈值以内的数据是健康的，而数据突破阈值的次数越多，表明网络更容易出现不稳定的现象。用于动态阈值计算的方法有很多种，本系统的研究以公司网络为例，每天的上班时间都是网络使用的高峰期，而半夜、凌晨则是网络使用的低谷期，可以近似地认为设备运行数据符合正态分布。根据以上特点，我们选用了正态分布的方法来计算动态阈值。

正态分布的计算公式如下：

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

正态分布有两个参数，即均值 μ 和标准差 σ ，其中， μ 是正态分布的位置参数，它描述了正态分布的集中位置； σ 是正态分布的形状参数，它描述了正态分布数据分布的离散程度， σ 越小，数据分布越集中， σ 越大，数据分布越分散。正态分布的曲线图如图 6 所示。

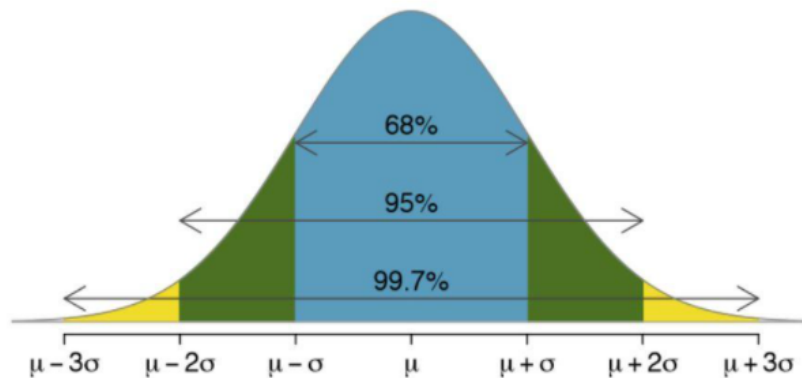


图 9 正态分布曲线

由图可见，均值 μ 增加和减少 3 倍标准差 σ 的范围已经几乎覆盖了所有正常数据的范围。按照正态分布的理论，我们可以将均值 μ 增加和减少三倍标准差 σ 来分别作为正常数据的上下阈值。

对于当前小时的均值和标准差预测，根据能够获取到的数据，我们采用过去 6 天、3 天和 1 天内同一小时的平均值再进行加权平均的方式来计算。具体的计算公式如下：

$$\mu = w_1Ds + w_2Dt + w_3Do$$

其中, w_1 、 w_2 、 w_3 分别为 6 天、3 天、1 天数据所占的权值, 而 D_s 、 D_t 、 D_o 分别为 6 天、3 天、1 天同一小时的平均数据。每个部分所占权值设定的主要依据是历史数据的数量, 时间最长的 6 天均值数据权值为 0.4, 其他两个权值都为 0.3。

得到均值 μ 的预测值之后, 可以计算标准差, 标准差是是离均差平方的算术平均数的算术平方根, 具体计算公式如下:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n}}$$

计算出均值 μ 和标准差 σ 之后, 取 $\mu - 3\sigma$ 为下阈值, $\mu + 3\sigma$ 为上阈值。

3.1.2 基于 Flask 的数据获取

前端界面的绘图需要使用到 API 提供的数据, 我们使用了 Flask 的路由来实现数据的传递。具体的实现步骤如下:

(1) 使用 Python 完成后端的代码逻辑编写, 函数返回 json 格式的数据, 利用 Flask 框架的路由将函数内容放到指定 url 中, 获取健康度趋势的代码如下:

```
@app.route("/health_trend")
def health_trend():
    url = "https://117.78.31.209:26335/rest/campuswlanqualityservice/v1/expmonitor/rate/basictable"
    token = get_token()
    headers = {
        "Content-Type": "application/json",
        "X-Auth-Token": token,
        "Accept": "application/json"
    }
    begintime = int(time.time() * 1000) - 24 * 1000 * 60 * 60 # 获取1天内的数据
    endtime = int(time.time() * 1000) - 1000
    print(endtime)
    data = {
        "id": "857b706e-67d9-49c0-b3cd-4bd1e6963c07", # 深圳站点
        "level": 3,
        "regionType": "site",
        "startTime": str(begintime),
        "endTime": str(endtime),
    }
    url = url + "?param="+parse.quote(json.dumps(data))
    res = requests.get(url, headers=headers, verify=False)
    return res.json()
```

(2) 前端部分使用 Ajax 技术请求指定 url 中的数据, 相当于调用了目标函数, 获取了 json 格式的返回值, 然后进行后续的处理, 数据获取并使用 echarts 绘图的示例代码如下:

```
$.ajax({
  type: "get",
  url: "http://127.0.0.1:8899/single_dimension_rate/" + str,
  dataType: "json",
  success: function(data){

    var axis = [];
    for(var i=0;i<24;i++){
      content[i] = data.data.baseline[i].value;
      axis[i] = i;
    }

    var coverage_now = document.getElementById("coverage_now");
    var accessSuccessRate_now = document.getElementById("accessSuccessRate_now");
    var accessConsume_now = document.getElementById("accessConsume_now");
    var capacity_now = document.getElementById("capacity_now");

    var chartDom = document.getElementById("u28_div");
    var myChart = echarts.init(chartDom);
    var option;

    option = {
      xAxis: {
        type: 'category',
        data: axis
      },
      yAxis: {
        type: 'value'
      },
      series: [{
        data: content,
        type: 'line'
      }]
    };
    option && myChart.setOption(option);
  }
})
```

需要注意的是，打开前端页面之前，后端的 python 程序必须先运行，否则会发生找不到 url，获取数据失败的情况。

3.2 具体功能

(1) 自动生成多指标的网络信息可视化展示，数据为最近 24 小时内收集的最新数据。

(2) 根据历史数据动态计算网络指标安全阈值，并统计最近 24 小时内收集的数据中超过安全范围的次数，以可视化形式展示。

(3) 自动绘制终端与 AP 拓扑图，鼠标悬浮可以查看指定终端或 AP 的基本信息。

(4) 生成数据报告，将各项网络指标汇总整理成表格形成数据报告。

四、运行结果

4.1 整体界面展示

将 4.1 可视化界面设计中的原型用 HTML、JS 等技术实现并进一步美化渲染后得到产品的最终界面显示。如图 10、11、12 所示。

其中容量健康度、吞吐达标率、信号与干扰、接入耗时数以及接入成功率的主要指标功能界面框架基本与图 10 相同，图 11 为生成拓扑图界面，图 12 为生成数据报告界面。



图 10 主要指标界面



图 11 楼层拓扑图界面



图 12 网络数据报告界面

4.2 主要功能展示

Monitor 网络智能运维系统主要分为两个功能模块：一是将各项数据以及对应阈值计算结果使用图表进行显示；二是对网络的整体结构拓扑显示和整体数据报告表格显示。系统功能图如图 14 所示。

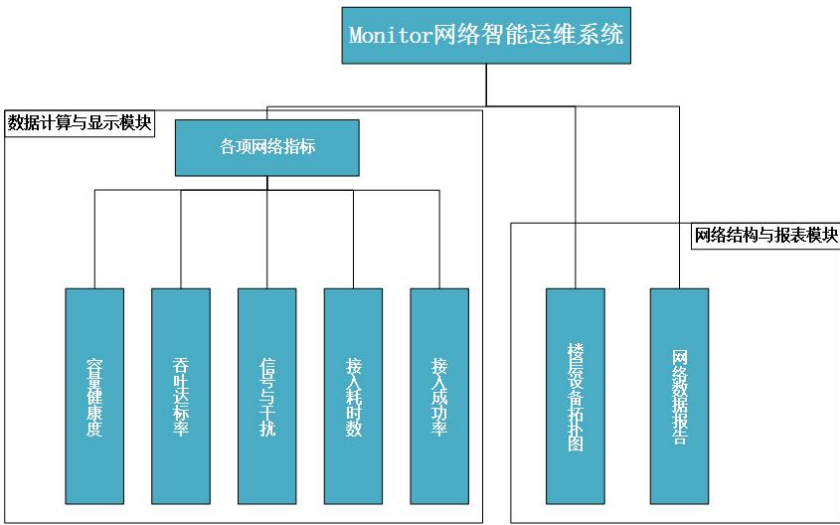


图 13 Monitor 网络智能运维系统功能图

4.2.1 数据计算与显示模块

1. 各项网络指标的数值显示



图 14 各项网络指标显示

2. 各项网络指标数据以及对应阈值突破次数图表显示

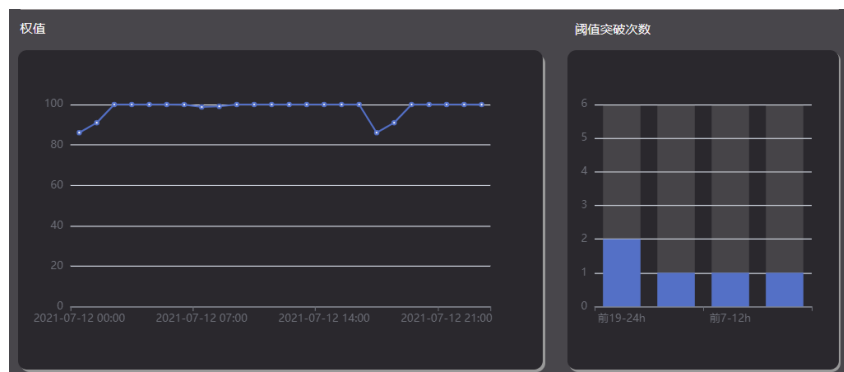


图 15 网络指标数据以及对应阈值突破次数图表显示 (各项数据显示形式相同)

这一部分使用了折线图对数据获取当天的数据权值变化进行了显示，并采用对应的阈值计算方法 (参考 3.1.1) 将数据阈值计算并对比出当天的阈值突破次数，采用柱状图的形式进行显示，这部分的数据包括图 13 中对应模块中的：容量健康度、吞吐达标率、信号与干扰、接入耗时数与接入成功率。

4.2.2 网络结构与报表模块

1. 楼层设备拓扑图

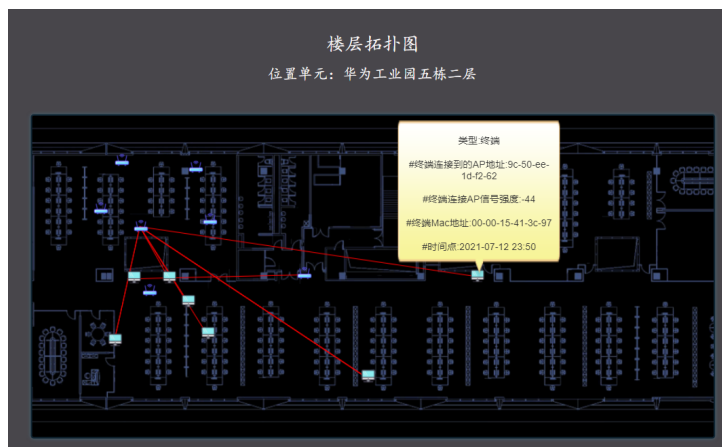


图 16 楼层设备拓扑图 (终端详细信息展示)



图 17 楼层设备拓扑图（AP 详细信息展示）

这一部分首先在顶部显示响应数据对应的站点楼层；然后根据获取到的终端以及 AP 终端的位置信息，在底层楼层平面图中对应位置显示对应的终端图标和 AP 终端图标，并在鼠标移动到对应终端或 AP 终端的点位时根据终端数据信息进行显示。如图 16、图 17 所示，显示信息分为两类，一是设备类型为终端的设备，显示信息包括：类型、终端连接到的 AP Mac 地址、终端连接到的 AP 信号强度、终端 Mac 地址以及时间点；二是设备类型为 AP 终端的设备，显示信息包括：类型、AP Mac 地址、AP 设备的名称。

同时，系统会根据终端信息中的连接到 AP 设备的信息，用连线的方式将终端以及对应连接到的 AP 终端进行连接成一个网络拓扑的形式，展示网络结构信息。

2. 网络数据报告



图 18 详细网络数据报告

这个部分是将所有获取到的网络指标数据以及阈值计算结果以图标的形式进行二维展示，如图 18 所示，横向是显示单项指标在不同时间段的均值，以及过去一天内的峰值和突破阈值的次数；纵向使显示某一时间段或对应数据项的所有网络指标的数值显示。系统也会依据这些数据进行计算，对网络状态进行评估。

五、创新与特色

5.1 创新点

信息系统运维的可视化代表了该领域的最新发展方向之一，结合工作中的实际场景，本系统可以有效的改善 IT 运维的效率，具有一定的创新点，具体有以下几点：

（1）将数据与实际场景结合展现。很多时候客户关注整体的网络运行情况，图表有助于形象且快捷地展现，特别是对非 IT 运维领域的人来说，能够一眼看懂。

（2）展现场景具有动态效果，支持多种方式展现数据。

（3）提高数据价值，对沉淀的历史数据进行利用。

（4）提高系统自动化程度，减少人力手工操作，增加阈值计算的准确性。

（5）采用动态阈值计算方法，比传统静态阈值更能根据实际场景科学地评估网络质量。

5.2 特色点

同时，本系统在设计风格等方面也颇具特色，具体有以下几点：

（1）网页风格采用深色主色调，以减少用户眼睛疲劳，让界面更容易阅读。

（2）采用多种数据呈现方式，如折线图，拓扑图等，使得用户更为直观地阅读数据。

（3）良好的网页色彩搭配，内容简洁明了。