长江大保护时空大数据云平台建设方案研究

* 报告直接引用本报告样式，严禁使用或引用其他报告样式或模板
* 报告章节编号请严格遵照我院质量管理文件要求：



报告中单位统一用英文（m km m2 m3 hm2等），出现表、图的地方，正文中应写见表1.1-1、图1.1-1等



二零二零年九月

# 绪论（张钟海）

## 研究背景

## 研究意义

## 研究现状

### 长江流域信息化平台研究现状

#### 感知体系

##### xx

（1）xx

### 时空大数据云平台研究现状

### 研究现状总结

## 研究目标、内容及技术路线

### 研究目标

### 研究内容

### 技术路线

## 考核指标与完成情况

xx图

xx表

# 长江大保护时空大数据云平台建设需求分析

## 引言

## 长江流域管理涉及主要单位部门及其职责

## 长江流域信息化建设现状及存在的问题

## 长江大保护时空大数据云平台服务对象及范围

## 长江大保护时空大数据云平台业务需求

## 长江大保护时空大数据云平台功能需求

## 长江大保护时空大数据云平台数据需求

## 长江大保护时空大数据云平台安全需求

## 长江大保护时空大数据云平台性能需求

## 长江大保护时空大数据云平台非功能需求

## 长江大保护时空大数据云平台建设目标及原则

# 长江大保护时空大数据云平台标准体系构建

## 时空大数据云平台标准体系（匡欢）

### 总体研究流程

构建标准体系是运用系统论指导标准化的一种方法。构建标准体系主要体现为编制标准体系结构图和标准明细表，提供标准统计表、编写标准体系编制说明，是开展标准体系建设的基础和前提工作，也是编制标准、修改规划和计划的依据。标准体系表是一定范围内包含现有、应有和预计制定标准的蓝图，是一种标准体系模型。总体研究流程如下：

##### 确定标准化方针目标

1. 了解标准化所支撑的业务战略;
2. 明确标准体系建设的愿景、近期拟达到的目标;
3. 确定实现标准化目标的标准化方针或策略、指导思想、基本原则；
4. 确定标准体系的范围和边界。

##### 调查研究

开展标准体系的调查研究，通常包括：

1. 标准体系建设的国内外情况；
2. 现有的标准化基础，包括已制定的标准和已开展的相关标准化研究项目和工作项目；
3. 存在的标准化相关问题；
4. 对标准体系的建设需求;

##### 分析整理

根据标准体系建设的方针、目标以及具体的标准化需求，借鉴国内外现有的标准体系的结构框架，从标准的类型、专业领域、级别、功能、业务的生命周期等若干不同标准化对象的角度，对标准体系进行分析，从而确定标准体系的结构关系。

##### 编制标准体系表

编制标准体系表，通常报告：

1. 确定标准体系结构图

根据不同维度的标准分析的结果，选择恰当的维度作为标准体系框架的主要维度，编制标准体系结构图，编写标准体系结构的各级子体系、标准体系模块的内容说明。

1. 编制标准明细表

收集整理拟彩的国际标准、国家标准等外部标准和本领域已有的内部标准，提出近期和将来规划拟制定的标准列表，编制标准明细表。

1. 编写标准体系编制说明

标准体系表编制说明的内容一般包括：

1. 标准体系建设的背景；
2. 标准体系的建设目标、构建依据及实话原则；
3. 国内外相关标准化情况综述；
4. 各级子体系划分原则和依据；
5. 各级子体系的说明，包括主要内容、适用范围等；
6. 与其他体系交叉情况和处理意见；
7. 需要其他体系协调配套的意见；
8. 结合统计表，分析现有的标准与国际、国外的差距和薄弱环节，明确今后的主攻方向；
9. 标准制修订规划建议；

##### 动态维护更新

标准体系是一个动态的系统，在使用过程中应不断优化完善，并随着业务的需求、技术发展的不断变化进行维护更新。

### 初步分析

长江流域信息化的研究和建设取得了较大进展，沿长江流域的水利外网、水利专网、防汛骨干网、电子政务网已基本建成。各类雨量站、水位站、水情、旱情、水质站、取水口与排污口视频监测站也逐步完善，水情监测系统、防汛抗旱系统、水资源监控能力建设等核心业务应用日趋丰富。但各应用系统多局限于某个区域、部门或者业务，且存在信息资源共享困难、缺少业务协同、信息化管理机制体制不健全等问题。在长江大保护的背景下，长江流域已有信息化系统难以满足长江流域科学管理的需求，迫切需要一个统一管理平台来支撑长江大保护背景下的综合管理。

当前长江大保护时空大数据相关的系统存在以下问题：

1. 数据分散

当前长江大保护系统均采用物理硬件独立部署，增加了管理的复杂度，导致各系统成为信息孤岛，不能即时共享信息。同时由于历史原因，应用系统使用的数据库包括ORACLE、MYSQL、PROGRESQL、SQLSERVER等，并且相同的数据库使用的版本不同，数据无法集中管理，综合利用率不高，维护人员工作负担重。

1. 硬件设备老化

长江大保护应用系统设备投入使用年限过长，高比例的服务器存储设备在线运行超过5年，大部分已运行3年，设备老化的问题，越来越影响到业务系统的处理能力，无法满足现有的需求。

1. 资源利用率低

由于各应用系统独立部署，每个应用按照峰值进行硬件资源配置，导致大量的硬件资源并没有合理高效经济的利用。并且由于系统独立，分布在各单位，各部门，导致数据不能集中，与数据的挖掘设置了屏障。

1. 运维成本高

由于每个单位或部门建立了自己的应用系统，采用了专用的服务器、网络、操作系统、数据库、存储等软硬件系统，导致了场地投入、机柜投入、制冷设备投入、硬件投入、运维人员投入的重复投入，导致运维成本高。

1. 安全配置不足

当前的安全防御系统主要实现在物理服务器的场景，无法满足业务系统虚拟化的场景，无法实现应用层安全防御。

1. 运行风险高

为了满足政策法规、部门规章对于应用系统的安全性与合规性要求，运维人员进行了大量的工作以保证要求，但由于数量和业务众多，导致这一目标在实际当中难以实现。并由于运维人员的技术水平、工作效率上的差异，硬件条件的区别，导致实施结果存在很大的差异，从而带来各种风险。

1. 部流程多、上线周期长

随着经济的发展，系统需求的不断演进，需要购置新的服务器，导致需要计划部门、采购部门、维护部门等参与，由于各部门的进度和流程不一致，经常导致部署流程多、上线周期长。

因此制定一套长江大保护时空大数据云平台标准体系很有必要性、迫切性、指导性、规范性作用。

结合需求、当前长江大保护信息化发展水平、未来发展需要，长江大保护时空大数据云平台标准体系包括政策法规、基础标准、专用标准大三模块。

### 编制方法

1. 制定工作计划
2. 任务分析与理解

理清编制目标和任务要求是做好标准制订的前提和基础。标准制订团队在负责人组织下共同参与学习讨论，对所要编制的标准总体目标和任务进行充分的讨论和深入理解，讨论过程都发表意见，统一了认识，形成了一个全面的共识。

1. 工作任务分解

对工作任务进行细化和拆分，分解到相对独立的子项任务。分解不能出现漏项，也不能包含不在任务范围之内的任何技术和要求。工作任务分解编制成了树形的电子文档。

1. 编制工作计划

工作计划内容包括：标准制度的总体目标和分项工作目标及阶段工作目标;任务的描述和质量要求;完成任务的支持条件;时间节点;责任人和配合人;形成了正式的文件。

1. 资料和信息的收集
2. 确定收集范围

依据标准编制的计划要求确定完成工作任务需要的相关资料和信息的范围、调研的对象、调研的方式、预设调研的题目，列出题纲。对调研的内容要依据编制原则的要求分裂出具体的调研点。

1. 调研方法

在调研阶段，为了标准制订的规范、全面、科学，收集信息采用现场调研、专家调研和文献查阅相结合的方式。

1. 信息处理

每一次的调研都编写了调研报告，完成调研后，将所有的资料和信息进行了汇总，分门别类的进行归纳总结，形成了最后的调研报告。

1. 信息分析

信息分析是为了全面的了解现状，包括国内外、行业内外的情况。在此过程中，一方面是找寻长处和不足，及存在的共性问题，另一方面通过对长江大保护时空大数据相关资料的学习，清楚明确对于时空、大数据、云平台三者相互之间的关系及相关技术的要求;再一方面对收集国内外、行业内外的相关标准特点进行分析，总结优劣，并与现状进行对比，确定需要改进的地方。对于信息分析过程中发现的潜在问题和不完善的地方，进行补充针对性的调研或信息的收集。

1. 标准的编写
2. 标准编写采取分工编写共同讨论的方法。团队中每个人按各自熟悉的领域和专业特长，起草标准时充分发挥了组员的特长。共同讨论是为了多角度看问题，力争表述准确，用词规范，逻辑严谨。
3. 标准编写的内容全面的遵守了标准编写的原则，充分体现行业特点。
4. 对于长江大保护云平台现状存在的不足进行改善时，充分借鉴国内外、行业内外的先进经验，借鉴时十分清楚其规定的核心思想和要义。对相关内容完全适用的进行引用，与实际情况不同的不可直接引用，但借鉴他们的思想进行部分的吸收或转换表达方式。
5. 意见征询

标准起草后，为避免编写团队囿于经验、知识和能力方面的不足等客观存在的因素，造成标准编写存在缺陷、漏洞或错误，在行业内进行意见的征询。

1. 征询意见广泛

意见的征询做到较广泛，行业内相关单位，行业外信息化水平高的公司或单位，行业相关的高校，及相关领域的专家。

1. 反馈意见认真对待

反馈结束后，对反馈的意见进行了学习和讨论，采纳好的建议对标准进行改进和完善。

### 规范设计

1）标准体系编码规则

依据编制原则和编制需求，结合长江大保护时空大数据云平台建设特点，制定长江大保护时空大数据云平台标准体系表编号规则，对列入技术标准体系中的每一项技术标准均赋予唯一的标准编号。编号规则采用结构编码+标准序列号的组合方式：

（1）结构编码分为：类别编号、子类别编号，每级结构编号由两位数字组成，当类别结构或专业结构需要扩充，可在已有结构编号基础上进行编号递增扩充，最多可扩充至99，当层级空缺时用“00”表示；

（2）标准序列号由三位数字组成，数字递增编码，当标准序列号需要递增，可在已有标准序列号基础上进行编号递增扩充，每级结构中的标准号最多可扩充至999。

2）标准体系结构表及说明

|  |  |  |  |
| --- | --- | --- | --- |
| 编码 | 类别 | 子类别 | 说明 |
| 01 | 政策法规 |  |  |
| 01.01 |  | 国家法律法规 |  |
| 01.02 |  | 地方条例法规 |  |
| 02 | 基础标准 |  |  |
| 02.01 |  | 顶层设计 |  |
| 02.01 |  | 术语 |  |
| 03 | 专用标准 |  |  |
| 03.01 |  | 物联感知 |  |
| 03.02 |  | 网络传输 |  |
| 03.03 |  | 基础设施 |  |
| 03.04 |  | 数据资源 |  |
| 03.05 |  | 基础支撑 |  |
| 03.06 |  | 业务应用 |  |
| 03.07 |  | 信息化管理 |  |
| 03.08 |  | 信息安全 |  |
| 03.09 |  | 运维管理 |  |

长江大保护时空大数据云平台标准体系分类说明

### 成果内容

标准体系主要成果包括：标准体系结构图、标准明细表、标准统计表、标准体系编制说明。

#### 标准体系结构图

长江大保护标准体系结构图，如下图所示，分为:

1. 政策法规，包括：国家法律法规和地方条例法规，用于指导长江大保护云平台的工作的开展，确保系统建设在合规合法的前提下有序推进。
2. 基本标准，包括：顶层设计和术语，用于总体指导云平台工作开展。
3. 专用标准，包括物联感知、网络传输、基础设施、数据资源、基础支撑、业务应用、信息化管理、信息安全、运维管理。其中数据资源包括：数据分类与编码、数据存储模块、数据管理，用于规范统一相关模块的工作开展。

标准体系结构图

#### 标准明细表

#### 标准统计表

## 时空大数据云平台顶层设计

### 总体研究流程

### 初步分析

### 编制方法

### 规范设计

### 成果内容

## 时空大数据云平台术语

### 总体研究流程

### 初步分析

### 编制方法

### 规范设计

### 成果内容

## 时空大数据云平台共享交换数据服务目录

### 总体研究流程

### 初步分析

### 编制方法

### 规范设计

### 成果内容

# 长江大保护时空大数据云平台数据资源建设

## 资源汇聚

### 主要时空数据资源

#### 基础时空数据

#### 公共专题数据

#### 物联网实时感知数据

#### 互联网在线抓取数据

### 资源汇聚方式

### 数据库设计

#### 总体设计

#### 逻辑设计

#### 概念设计

#### 物理设计

## 空间数据处理

### 处理原则

### 处理流程

### 处理方法

## 数据引擎

### 引擎架构

研究矢量数据、影像数据在云平台中的组织和索引方式，提出一个私有云的地理数据存储体系，帮助用户在线调用现成的时空大数据中的数据

### 属性数据存储及索引

### 矢量数据存储及索引

### 影像数据存储及索引

### 数据安全设计

## 分布式管理系统开发

### 设计原则

### 系统架构

### 系统划分

### 模块设计

#### 动态数据获取

#### 数据管理

#### 分析量测

#### 模拟推演

#### 大数据挖掘

#### 大数据管理

### 接口设计

#### 外部接口

#### 内部接口

# 长江大保护时空大数据云平台运行环境保障

## 时空大数据运行服务设计

### 设计原则

### 总体设计

### 数据资源管理

### 资源可视化管理

### 服务引擎设计

### 云端部署

## 云平台运行服务设计（匡欢）

### 设计原则

长江大保护云平台结合当前长江大保护信息化存储的问题和未来发展需要，在设计当中，遵守以下原则：

1. 先进性

广泛采用虚拟化、自动化调配等先进技术或模式，确保先进技术与应用模式的有效与适用。

1. 可扩展性

云平台支持资源应能根据业务应用工作负荷需求进行伸缩，这样性能及与服务水平的符合性保持适当。在系统进行容量扩展时，只需要增加相应数量的硬件设备，并在其上部署相应的资源调度管理软件和业务应用软件，即可实现系统扩展。

1. 成熟性

采用成熟的技术手段实现各种功能，满足相关部门的业务要求。

1. 开放性与兼容性

开放性本身有两个含义：源代码开放和标准开放。源代码开放，可以拥有完全的掌控，可以修改或增加新地功能满足自身的需求;标准开放意味着可以通过各种符合标准的产品构成自己的云平台方案。

采用业界通用的服务器，能够兼容主流的操作系统，虚拟化软件，及应用程序，降低使用、管理、运维等成本。

1. 可靠性

充分考虑系统的高可用，系统在出现高并发达到瓶颈、出故障等问题时，能够较自动化进行伸缩、重启、切换、迁移，使得用户无感知，保证业务的稳定高效的运维，提高用户体验。

1. 安全性

必须高度的防范网络入侵攻击、病毒感染，也需要对内部人员权限的管理，确保信息安全和私密性。

1. 统一管理与自动化

实现系统的按需运营，多种服务的开通，这依赖于对计算、存储、网络资源的调度和分配，同时提供用户管理、组织管理、工作流管理等。从IT资源的申请、审批到分配部署的智能化。管理系统不仅要实现对传统物理资源和新的虚拟资源进行管理，还要从全局而非割裂地管理资源。

1. 标准规范化原则

在云平台的建设、运维过程中，将依据国际、国内相关标准，避免彩私有的协议与标准，而导致互通困难。同时通过遵循统一的标准，实现资源共享、业务协同、安全可靠运转奠定坚实的基础。

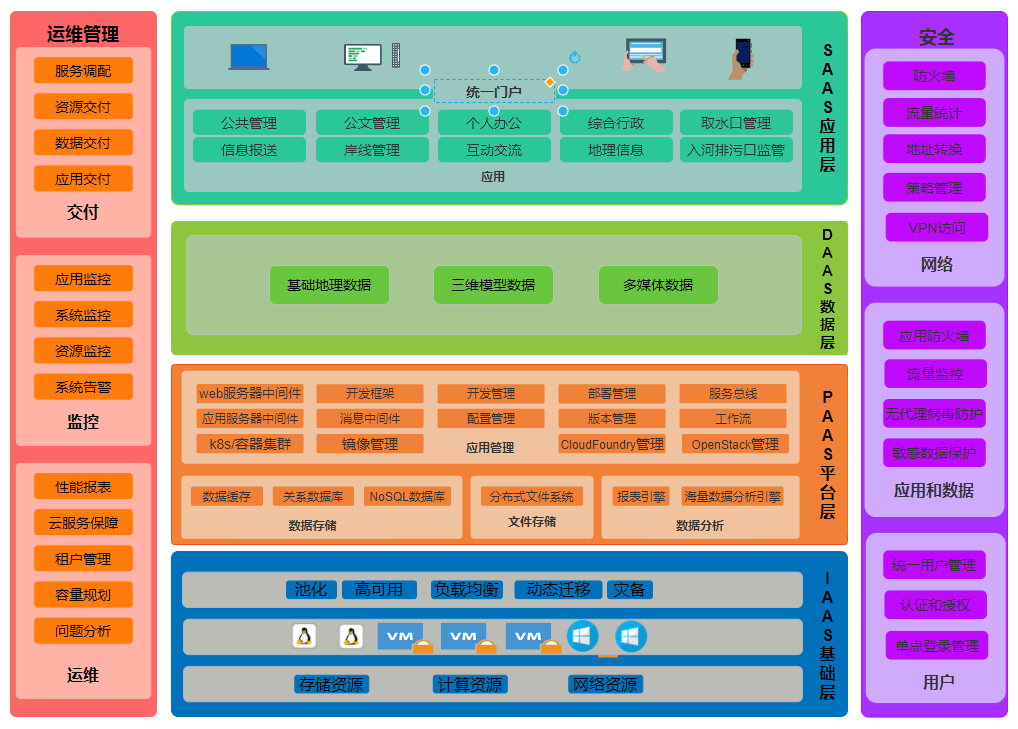
1. 开放接口

为保证服务器、存储、网络等资源能够被虚拟化运营平台良好的调度与管理，提供开放的API接口，能够通过API接口、命令行脚本实现对设备的配备与策略的下发联动。同时云平台提供开放的API接口，未来可以在这些接口的基础上进行再开发，实现面向虚拟化的数据中心管理平台。

### 总体设计

长江大保护时空大数据云平台使用四层架构设计，分别是SaaS应用层、DaaS数据层、PaaS平台层以及IaaS基础设施层。如图5-2.1所示。

IaaS层基于OpenStack、KVM虚拟化以及Ceph分布式存储技术，对各个功能组件经过深度优化开发，为业务系统提供基础承载平台。实现了存储资源、计算资源、网络资源的虚拟化。通过统一的接口，对这些虚拟资源进行集中的调度和管理。



总体结构

PaaS层在Iaas之上，使用混合的PaaS架构，实现高可用架构的统一的云管理平台、运维体系、流程和工具、安全体系、数据存储、数据分析、文件存储等。

在安全方面，按物理安全、网络安全、主机安全、应用安全、数据安全进行设计。实现统一登录认证、认证和授权、单点登录的用户管理，在网络方面实现防火墙、流量统计、地址转换、VPN访问。应用和数据安全方面实现应用防火墙、流量监控、无代理病毒防护、敏感数据保护、全链接追踪、日志监控、性能监控等。

运维管理方面，包括监控、交付、运维等功能。

DaaS层，提供基础地理数据、三维模型数据、多媒体数据等数据服务功能。

SaaS应用层，用户通过统一门户可以使用公共管理、公文管理、个人办公、综合行政、取水口管理等应用。

云平台使用3+N的拓扑结构，即包含了3台控制节点， N台计算存储节点（N>=3）。该结构既可以为用户提供可靠的分布式存储，也能保证OpenStack服务的高可用性，也增强云平台的可扩展性。

### IaaS层设计

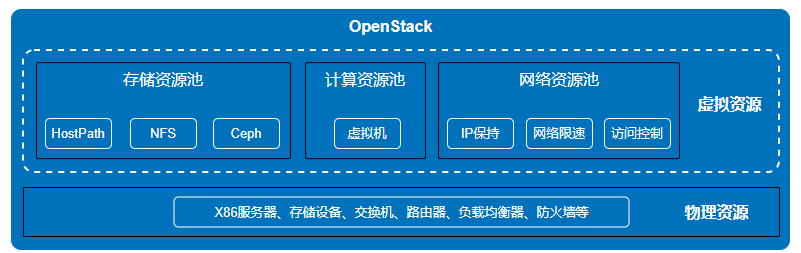
基础架构即服务 (IaaS，Infrastructure as a Service) 提供托管的 IT 基础架构，供用户调配处理能力、存储、网络和其他基础计算资源。IaaS 提供运行并管理此基础架构，用户可以在此基础架构上运行选择的操作系统和应用程序软件。

IaaS层建立了统一标准的信息化基础设施。通过资源池化技术和虚拟化技术。把硬件资源如服务器、存储设备和网络资源通过用户访问和需求动态计算的方式向辖区内各接入单位提供灵活多变的应用资源交付。用户几乎不用担心服务器端硬件资源的维护和拓扑布局，可以实时自由访问所需要的资源。IaaS层提供了整个架构的物理基础稳定的同时，亦提供了可靠的可访问性和可维护性。

资源层将物理资源与PaaS应用业务解耦，屏蔽了节点之间的配置差异。资源层主要分为两部分：物理资源与虚拟资源。如图5.2-2所示。

物理资源层为平台提供了最基础的物理资源，包括主机、存储、网络及其他硬件在内的硬件设备，例如X86服务器集群、存储设备、交换机、路由器、物理资源负载均衡器以及防火墙等为云平台提供虚拟资源池的物理设备，是容器化云平台的基础和硬件条件。

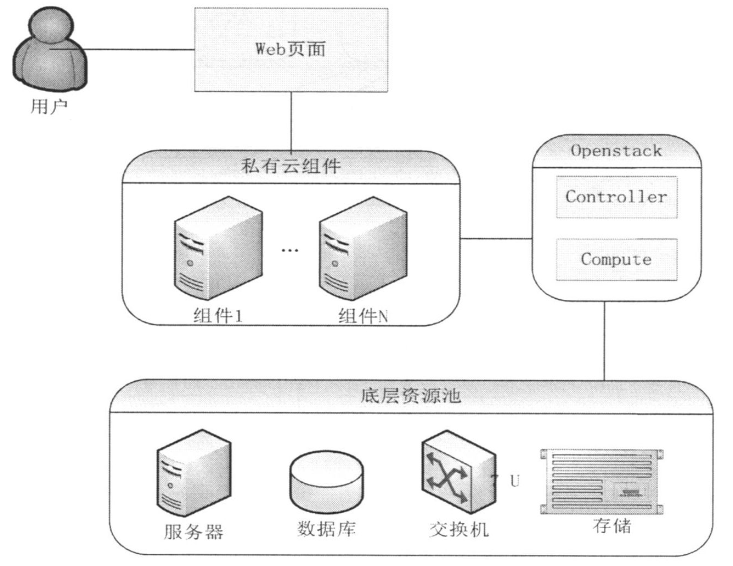
虚拟资源层隐藏了物理资源的复杂性，实现了多种资源的虚拟化服务。通过对于多种资源的虚拟化整合技术，完成了集群资源的池化管理。虚拟资源层主要提供三种资源池：计算资源池、存储资源池和网络资源池。计算资源池中包括了各种不同配置的容器；存储资源池支持HostPath、NFS和Ceph这三种主流的持久化存储卷类型；网络资源池提供了相关的网络处理能力，如IP保持、网络限速及访问控制等。



IaaS结构图

基于OpenStack平台，结合KVM虚拟化技术以及Ceph分布式存储技术搭建云平台。实现虚拟计算服务、网络服务、对象存储服务、镜像管理服务、块存储服务及控制面板服务等服务。通过浏览器来对计算、存储和网络等资源的访问控制，如创建、启动、停止虚拟机实例、创建路由子网、设置安全组、虚拟机管理、认证管理、镜像管理以及计算资源、网络资源和存储资源的管理和调度等操作。

云平台的物理架构如图5.2-3所示，主要分为用户界面、私有云组件、底层资源池模块。

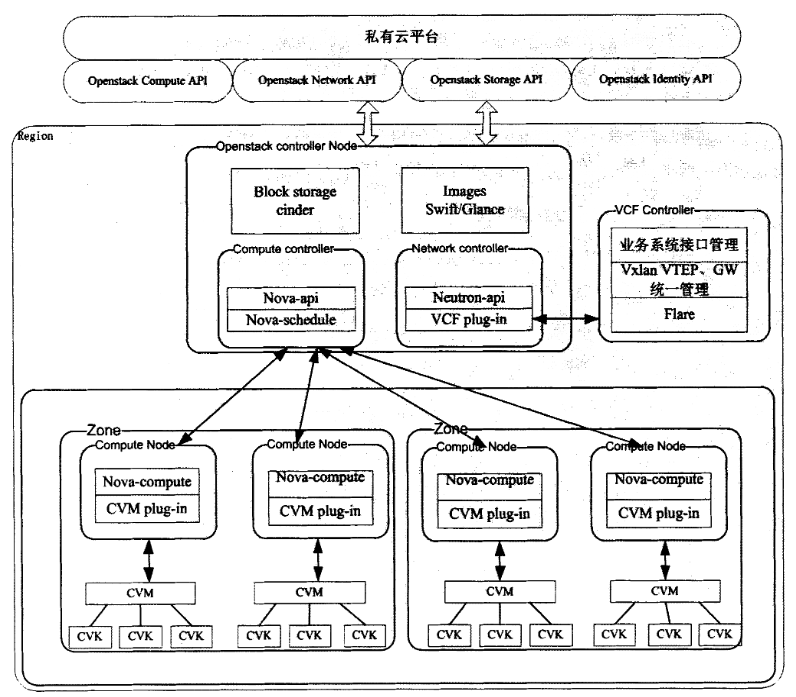


云平台物理架构

云平台用户界面模块主要实现用户与平台的交互，如资源的申请、删除、查找，用户的管理，租户的管理，镜像的管理等操作。

私有云组件模块主要包括Openstack的组件，如计算节点、存储节点、控制节点和网络节点。

底层资源也部署了大量的高性能服务器，利用KVM虚拟化技术实现资源整合，实现云服务器、云存储的融合交付;部署数据中心级的交换机和防火墙等多业务安全网关，构建overlay虚拟网络，实现云安全、云网络服务的交付。

LaaS逻辑架构图

为了实现大规模资源管理，同时支持业界的标准，对OpenStack的Nova(计算)、Cinder(块存储)、Neutron(网络)、Glance(镜像)和Swift(对象存储)进行定制，提供面向接口与不同厂商的Hypervisor和基础设施对接，实现对LaaS层资源的管理。在Keystone的基础上对租户管理进行开发，满足复杂的组织结构关系。

### DaaS层设计

数据即服务（DaaS，Data as a Service）是指与数据相关的任何服务都能够发生在一个集中化的位置，如数据的聚合、质量管理、清洗等，然后再将数据提供给不同的系统和用户，而不用再考虑这些数据从哪些数据源来的。DaaS层实现对基础地理数据、三维模型数据及多媒体数据的集中管理，并开发服务及API对外使用。

### PaaS层设计

平台即服务 (PaaS，Platform as a Service) 提供在集成式云环境中开发、测试、运行和管理 SaaS 应用程序所需的基础架构和计算资源。

PaaS 平台是基于底层不同云基础资源之上的一个开放、可运营、可动态扩展、易运维的云服务平台。要求实现对不同云基础资源统一化管理、第三方工具快速集成管理、应用产品快速部署管理、统一用户和应用管理、统一公告管理、平台运行状态实时监控等。为保证平台通用性，不受IaaS层限制，通过打造 PaaS 平台，实现对所有IaaS层基础服务统一封装和协议转换，适配不同云服务提供商的开发接口，为上下层平台提供一个标准化开发运行环境。

PaaS 平台层构建在 IaaS 基础层上，把软件开发、测试和部署环境以服务方式对外进行提供。PaaS 平台层为各业务提供包括消息中间件、关系数据库、大数据框架、Window/Linux操作系统、开发环境(python、C、C++、Java、Go、nodejs)等在内的软件，允许进行应用远程开发、配置、部署。如图5.2-3所示。

PaaS服务层，采用混合型PaaS架构设计。其核心思路是基于统一的PaaS框架，提供适合不同业务需求的PaaS开发环境。针对无法改造的业务，提供VM模板和应用模板两种PaaS服务，让传统业务可以平滑迁移到云上。针对传统数据库和大数据两种数据类型的功能，提供Oralce、My SQL、Postgresql、MongoDB、Hadoop、Spark、Flink、Storm等技术。针对传统应用架构但可以优化改造的系统，采用CloudFoundry的PaaS服务。针对新开发的应用，采用基于Kubernetes的分布式容器服务PaaS服务，作为后续演进的统一的PaaS架构。

大数据分析引擎作为 PaaS 平台基础服务能力，处于 PaaS 平台核心应用层 ，可与 PaaS 平台消息引擎、用户管理、基础服务能力管理、云管理中间件无缝结合，为 SaaS 层业务系统提供统一的数据处理能力，大幅提升SaaS层业务系统数据处理效率。利用OpenStack中的Sahara组件实现在OpenStack云环境中简单、快速地部署并管理Hadoop集群，集成Hadoop可以向上提供和扩展理高效的大数据存储、大数据安全备份、大数据处理分析等服务。

开发支持环境为用户提供了一站式软件开发部署平台，除了Devops(Development和Operations的组合词，是一组过程、方法与系统的统称，用于促进开发、技术运营和质量保障部门之间的沟通、协作与整合)工作流管理功能，还集成了丰富的开发工具，力求解决应用服务开发、测试、交付和运维一体化的问题。



Paas服务

PaaS层资源调度包括自定义的调度规则、实时调度器与资源调整，并将容器组作为调度的基本单位。调度规则可以帮助调度器更加精确、合理、公平地为容器组，应用和租户之间分配资源。实时调度器可辅助容器组选择合适的主机。资源调整模块可在资源分配不合理的情况下通过改变部分容器组的运行节点来改善集群整体的资源分配状况。

Paas层提供了应用管理、镜像管理、容器管理、应用市场管理、资源管理、审计管理、多租户管理和监控管理等多种管理和维护功能。如图5.2-4所示。



功能分解

1. 应用管理

应用管理为两类用户提供服务：PaaS运维管理人员和应用开发者。此模块为前者提供了必要的监控告警和日志管理等，提高了管理效率以及为应用的高稳定、高可靠性等提供了平台级的保障。应用管理为应用开发者提供了丰富类型的组件和应用运行环境，开发者可以更加专注于业务逻辑开发，进一步加快开发与发布应用的效率。

负载均衡管理：由于容器生命周期短暂，随时可能被删除和重建，是不稳定的对象，为了对外部提供更稳定的访问入口，让用户感知不到容器状态的变化，设计了负载均衡器模块。用户在访问容器服务的时候，访问的是负载均衡器的地址，由负载均衡器将流量转发到后端对应的一组容器当中去。

应用拓扑管理：为了让用户更好地了解应用内服务之间的依赖关系，每当应用发布后，会自动生成一个应用拓扑关系图。用户从应用拓扑管理中可以直观地查看应用中服务的关联关系。

服务和应用监控：PaaS平台需要随时了解应用和服务的实时健康状态，包括系统的性能指标以及应用指标。当应用异常时，会立即触发告警并帮助运维人员快速定位错误发生的位置及原因。

服务扩容：当服务在运行过程中需要增加容器实例时，服务扩容模块可以支持用户手动增加容器数量。容器关联的负载均衡器通过事件监听发现容器数量变化后，自动修改并重新加载配置，动态调整流量的分发策略。弹性伸缩：当业务需求变化后，弹性伸缩模块可以自动为用户调整物理计算资源量，在提高系统资源利用率的同时保证业务能够正常平稳地运行。

应用编排：应用编排是将一组关联应用组件自动整合为一个完整应用的过程，其目的就是将这些应用组件统一管理，统一监控。模板化的应用编排为应用开发者重用解决方案、快速发布应用提供了可靠的技术支持。

灰度升级：将正在运行的服务相关的容器逐个进行升级；针对不同的应用设置不同的滚动周期，逐一替换；升级过程中如果发生异常，可以对已升级的容器做回滚操作，确保服务的可用性。当需要对正在运行的应用进行升级时，为了保证服务不中断，需要进行灰度升级。

应用和服务的生命周期管理：提供了完整的应用和服务的生命周期管理机制，可监控应用和服务的创建、部署、启动、回滚，扩容缩容和停止下线等过程的状态。

(2)资源管理

PaaS平台的稳定运行离不开基础设施的支持，底层资源既可以是IaaS平台提供的虚拟化资源，也可以是数据中心直接提供的物理资源。资源管理模块的主要功能就是对底层资源进行整合，并建立其与容器的映射关系，解决运行容器应用所依赖的物理机计算、存储、网络等，简化了创建和维护容器化应用的难度。

主机管理：负责对接多种不同类型的主机，包括各种云平台的虚拟机、自有的物理机等。支持查询集群中管理的主机列表，每台主机的状态、操作系统、使用的Docker版本、主机上运行的容器列表等。

存储管理：支持为PaaS平台添加多种不同类型的存储池。容器需要使用共享存储卷持久化数据时，从存储池里创建存储卷，挂载给容器使用。同时还提供文件管理器，容器中的应用可以直接对存储卷中的数据做操作，比如上传文件、下载文件、编辑文件、移动文件等等。网络管理：支持为PaaS平台对接多种不同类型的网络插件，如Calico、Flannel以及作者设计的Xanet插件等，这些插件将在以后的章节详细介绍。

1. 镜像管理

镜像是容器技术的核心之一。镜像管理模块应用Docker容器技术把应用及其所依赖的软件包、操作系统文件等封装在镜像文件中，使应用在开发、测试和发布过程中具有相同的运行环境。管理镜像仓库和其中的镜像也是PaaS平台需要提供的功能。

镜像仓库：包含所有平台中提供的镜像。无论发布镜像，还是下载镜像，都要通过镜像仓库来完成。镜像仓库和租户体系相结合，可对仓库中的各种操作鉴权，使用角色限制用户对镜像的操作权限。

镜像构建：支持用户在PaaS平台利用DockerBuild命令在线构建新的镜像。把Dockerfile及其依赖的安装包、可执行文件等压缩成tar包，提交到Docker服务中构建镜像，最后把构建好的镜像上传到本地的镜像仓库中。

镜像管理：提供Docker镜像的管理功能，例如镜像权限管理，多租户间镜像隔离，镜像版本查询等。除此之外，还可以统计topN镜像、配置备份仓库中的镜像做定时同步、导入外部镜像到本地仓库中等。

镜像仓库管理：与镜像管理类似，不过管理的对象由单个镜像变为整个镜像仓库。系统默认为每个租户创建一个镜像仓库，镜像仓库中的镜像默认是私有权限，只对本租户下的用户可见。管理员可以修改镜像属性为公开，提供镜像给租户外部的用户使用。

1. 应用市场管理

PaaS平台提供了功能强大的应用市场支持，把常用的Mencache集群、MySql集群、Redis集群、Tomcat等封装成模板，普通用户可以自由选择这些独立的组件模板构建应用；同时也提供了大量经过验证的微服务集群应用，为用户获取和分享业务组件提供了技术手段。

应用市场管理包含大量经过验证的微服务集群应用软件，支持用户快速获取和分享业务组件。容器管理主要负责容器运行过程中的展示、监控、迁移以及容器中数据的持久化。多租户管理可有效隔离多个具有不同权限租户之间的应用状态和数据。监控管理对平台中容器、主机、应用的重要指标进行监控，并汇总监控结果生成报表，辅助系统快速找出故障原因及位置。

1. 容器管理

在PaaS平台当中，应用、运行环境软件、系统文件等被封装在容器的镜像中。启动镜像后，容器成为一个镜像的实体，业务就运行在其中。所有的容器都必须属于某一个服务，不允许对容器进行单独的申请和特殊操作。PaaS平台中的容器管理模块主要负责管理容器的运行情况，以及容器和服务之间的依赖关系。

容器控制台：在门户提供控制台功能，普通用户可以在控制台中执行命令，相当于用户直接登录到容器的操作系统中执行命令。容器控制台功能使用websocket实现，使用websocket建立门户和后台服务的双工通信，把用户的操作封装后发送到后台服务，把容器的响应封装后推送到门户中做展示。

容器监控：支持容器级别的性能监控，主要监控容器的CPU、内存、存储和网络信息。容器监控通过cAdvisor实现，进行原始的数据采集，监控服务在每个采集周期内调用cAdvisor接口查询原始数据，并对原始数据做处理和汇聚，处理后的数据提供给Grafana做展示。容器列表：支持多维度的容器列表展示，包括应用下的容器列表、服务下的容器列表和主机下的容器列表。

容器网络：容器网络使用Docker的none模式，容器创建时只创建一个空的网络命令空间，由网络插件设置容器的网络命令空间参数，对容器分配IP，保证容器到主机、容器到容器之间的网络通信。

容器迁移：容器迁移的目标是使容器在不停止业务运行的前提下，从一台主机迁移到另一台主机。由于容器本身不支持迁移操作，作者选择了一种折中的实现方案，当容器要迁移时，先在目标主机上创建一个相同的容器，然后等容器正常工作后更新负载均衡器，把新创建的容器接入到负载均衡器，把旧的容器从负载均衡器上删除。最后删除旧的容器。

容器持久化：容器持久化利用共享存储实现。容器启动时候挂载共享存储卷到容器中使用，需要持久化的数据写入共享存储卷，保证容器停掉或迁移后数据不会丢失。

1. 多租户管理

对于一个企业级的PaaS平台来说，多租户架构不仅能够降低服务的开发成本与运维成本，还可以让企业内不同部门的授权用户独立创建并管理自己的应用。当应用服务运行时，租户之间的状态和数据都是隔离开的，对于租户来说，自己是独享服务的，租户是可以持有资源的最大单位。

租户管理：负责创建新租户，将租户与机构绑定或解绑，访问查询租户信息等操作。管理的租户信息包括基本信息，成员信息，权限等级，所属机构等。

用户管理：负责创建新用户，将用户与机构绑定或解绑，设置用户角色，访问查询租户信息等操作。管理用户的基本信息，例如登录名，密码及其它个人信息等。用户和租户使用角色关联，每个用户登录后可以看到所属租户下的资源信息，可以对所属租户下的资源进行操作。

组织机构管理：分级管理全部租户的应用权限，包括普通用户、普通管理员、以及超级管理员。普通用户可以对所属租户下的资源进行操作。普通管理员除了管理当前租户的资源外，也同时管理本租户下的用户信息，修改用户的权限信息。超级管理员管理集群中的所有资源，同时管理所有的租户和用户信息。

用户权限管理：设置并管理租户，权限和角色的映射关系。根据用户和租户的权限内容授予其不同的资源操作权限。

1. 监控管理

监控管理主要负责PaaS平台中各种资源的监控管理，并支持查看资源和应用的各种指标，例如负载指标、服务指标等。同时可以自定义指标的汇总方式，设置相关告警，帮助运维人员快速定位故障及解决故障。容器监控：与容器管理中的容器监控相同。

服务器监控：定时从服务器拉取服务器的CPU、内存、存储、网络等指标数据。运行在主机上的node-exporter进行原始的数据采集，监控服务在每个采集周期调用接口获取指标数据，数据处理和汇聚后提供给Gafana做数据展示。

应用监控：使用特定的应用监控exporter实现。每种exporter采集特定应用的指标，如mysql-exporter负责采集MySql数据库的连接数、每秒事务量等；Redis-exporter负责采集Redis的内存使用率、延迟时间等。部署应用时，如果用户选择需要采集应用指标，PaaS平台除了部署应用本身外，也会部署应用指标采集的exporter。监控服务通过定时调度，定时通过特定的exporter采集指标数据，数据处理和汇聚后提供给Grafana做展示。

指标管理：负责接收性能指标数据，如容器、服务器、节点、CPU和内存的利用率、硬盘存储量、磁盘读写和网络速率，系统IP数，吞吐量以及集群整体的各项指标，并提供查询和展示的功能。

监控汇总分析：支持用户在平台中自定义如何处理原始指标数据，比如采集周期的长短，多条应用指标以哪种方式汇聚，使用原始指标生成复杂的报表指标等。处理后的指标依然和原始指标一样可以被查询和使用。

监控告警：通过配置告警规则，当如存储、CPU使用量、网络、内存等超过告警规则临界值时，触发告警，通过webhook方式，通知相应工作人员。

### SaaS层设计

软件即服务 (SaaS，Software as a Service) 使用户可以通过 Internet 访问软件应用程序。用户不必购买并在自己的计算机或设备上安装、更新和管理这些资源，而可以通过Web 浏览器访问并使用它们。SaaS 提供商在云中为用户管理软件、处理能力和存储。大多数 SaaS 解决方案在公共云中运行，并以订阅或免费服务的形式提供

SaaS应用层包括两方面内容 ：一是应用服务产品体系 ，应用服务须支持多租户、可配置、可扩展、网络化 ；二是服务运营管理体系，服务须安全稳定，支持产品管理、终端管理、用户管理等功能。在 SaaS 应用层提供智能汇聚、融合生产、内容库管理、内容发布、全媒体业务协同指挥等模块。这一层是通过网络浏览器来接入 ，在远程服务器上的任一应用都可以通过网络来运行。通过这种模式 ，不需要顾虑类似安装等琐事 ，能够方便用户使用 ，并提供一定的可定制性以满足用户的特殊需求。

平台门户主要分为三个部分：运营门户、运维门户和开发者门户，分别对应PaaS平台不同的使用对象，为他们提供了更直观的系统信息内容整合。平台门户帮助使用者快速获取平台、集群资源和应用服务的使用情况和运行状况。

### 运行环境要求

运行环境有以下要求：

1. 部署服务器要求

CPU：支持基于Intel VT /AMD-V以上X86指令集架构的处理器；

内存：单条内存8G，16G，32G，选用DDR4接口标准为最佳；

机械硬盘：需要支持SATA，SAS硬盘接口；

SSD硬盘：对IO要求高的应用需要配置SSD盘，支持SATA或PCI-E接口；

网络：每台服务器至少三块网卡，网卡需要支持1GbE/10GbE/infiniband，根据不同需求选择不同网络。

1. 推荐配置

服务器：6台, CPU E5-2620V3\*2

内存：DDR4 16G\*16

万兆网卡：双口万兆网卡\*2

千兆网卡：双口千兆网卡\*1

系统硬盘：SAS 300G\*2

硬盘：SATA或SSD 4T\*12

交换机：

管理网: 有带外管理口48口千兆交换机

计算网：48口万兆交换机（支持堆叠）

1. 组件列表

云平台所需组件如下表5.2-1所示：

|  |  |
| --- | --- |
| 组件 | 说明 |
| OpenStack版本 | Victoria |
| 操作系统 | CentOS7.4及以上 |
| 虚拟软件 | KVM |
| 数据库 | MySQL |
| 消息队列 | RabbitMQ |
| 高可用 | HAProxy、Keepalive |
| 应用容器引擎 | Docker |
| 容器集群管理 | Kubernetes |
| 镜像仓库 | Harbor |
| 自动化运维工具 | Ansible |
| 日志监控 | Elasticsearch、Kibana、Logstash、Kafka |
| 性能监控 | Promethues、Grafana |
| 网络监控 | Skydive |
| 网络隔离 | VxLAN |
| 网络服务 | Neutron |
| 分布式存储 | Ceph |

组件列表

### 安全保障要求

平台在建设中，需充分考虑安全机制和保密机制。安全地让用户能进行数据交互。规划拥有有效的、完整的、全面的安全方案，是设计平台之初最重要原则。在规划和开发系统时，最先要保障的就是平台的数据、功能、资源的安全性。在开发过程中，需要一直遵守设计最初制定的安全方案。在设计平台网络方面，设计结构应为三层，以保障全面的、安全的数据库数交互；在服务器选择方面，抉择良好的数据库和平台，保障达到计算机安全级别C2及以上安全性的平台；在管理方面，应当严格加密交互数据；在权限方面，应当实现分机权限机制，保障平台的安全性，设计出完善合理的角色权限机制。

一个对外运行的平台，必须要能提供一年三百六十五天二十四小时不休假的持续服务工作，所以必须要能有稳定和可靠的特点。依靠质量的保障、流程业务的管理和平台整体设计的优化，来确保平台的可靠性。会对平台使用方造成损失，如果平台出现故障，这就是平台高稳定性的作用。而，对外运行开放的平台，通常拥有很庞大数量的数据的访问和处理，这就是平台高可靠性的作用。

# 长江大保护时空大数据云平台高效管理及智慧应用

## 长江大保护信息管理与服务支撑平台高效管理要求

### 数据资源建设要求

### 模型建设与管理要求

### 资源统一管理要求

## 长江大保护时空大数据云平台高效管理要求（匡欢）

### 信息共享与高效分发要求

### 空间数据管理要求

### 多层级访问要求

### 风险管控要求

### 应急管理要求

## 镜像长江建设

### 设计原则

### 系统架构

### 系统划分

### 模块设计

### 接口设计

## 智慧应用

### 供水监管

### 排水监管

### 岸线监管

### 防洪减灾管理

### 工程建设管理

### 水环境治理管理

### 应急处置管理

### 综合决策支持（匡欢）

# 关键技术

## 长江大保护时空大数据统一的描述方法

## 长江大保护背景下的时空大数据云平台建设方案

## 长江大保护数据、资源高效管理和平台性能保障

# 总结

## 结论

## 创新点

## 研究展望