

团 体 标 准

T/UNP XXXX-XXXX

计算机视觉开发推理系统技术规范

Technical specification of inference system for computer vision
development

(征求意见稿)

20XX-XX-XX 发布

20XX-XX-XX 实施

中国联合国采购促进会

发布

目 次

前 言	II
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 基本要求	2
5 性能要求	4
6 系统架构	5
7 功能要求	8
8 安全要求	9
9 测试与运行维护	10

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由山东极视角科技股份有限公司提出。

本文件由中国联合国采购促进会归口。

本文件起草单位：山东极视角科技股份有限公司、.....。

本文件主要起草人：XXX、XXX、XXX.....。

计算机视觉开发推理系统技术规范

1 范围

本文件规定了计算机视觉开发推理系统的术语和定义、基本要求、性能要求、系统架构、功能要求、安全要求、测试与运行维护。

本文件适用于计算机视觉开发推理系统的设计、开发、测试、部署与维护。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 8567 计算机软件文档编制规范
GB/T 16680 系统与软件工程 用户文档的管理者要求
GB/T 20270 信息安全技术 网络基础安全技术要求
GB/T 20988 信息安全技术 信息系统灾难恢复规范
GB/T 22239 信息安全技术 网络安全等级保护基本要求
GB/T 22240 信息安全技术 网络安全等级保护定级指南
GB/T 28035 软件系统验收规范
GA/T 708 信息安全技术 信息系统安全等级保护体系框架

3 术语和定义

下列术语和定义适用于本文件。

3.1

计算机视觉开发推理系统 computer vision development inference system

一种集成数据处理、模型训练、算法开发与推理应用功能，旨在实现计算机视觉任务从数据到模型部署全流程自动化与智能化的软件系统。

3.2

数据标注 data annotation

对计算机视觉数据（如图像、视频）中的目标对象进行标记、分类或绘制边界框等操作，为模型训练提供有监督学习的样本标签，标注方式包括人工标注、基于深度学习模型的智能标注以及两者结合的半自动化标注。

3.3

异构算法开发 heterogeneous algorithm development

在计算机视觉开发推理系统中，针对不同硬件架构（如 CPU、GPU、FPGA 等）和软件环境进行算法适配与开发的过程。

3.4

模型优化智能调参 model optimization intelligent parameter tuning

利用自动化算法或工具，在计算机视觉模型训练过程中对模型的超参数（如学习率、批大小、迭代次数等）进行调整与优化，以提高模型的性能指标（如准确率、召回率、F1 值等）。

4 基本要求

4.1 一般要求

4.1.1 系统建设应遵循整体设计、统筹建设，对接审批、优化服务，统一标准、安全可靠的原则。应易于操作、界面美观、方便用户进行浏览、搜索和交互。

4.1.2 系统中涉及涉密数据时，应符合国家和行业保密管理的规定。

4.1.3 系统运行环境应符合国家信息安全保密管理的规定，系统应对用户实行统一身份认证，实现分权 分域管理。

4.1.4 系统的密码使用和管理应符合国家密码管理的规定。

4.1.5 系统应通过验收，系统验收应符合本文件和 GB/T 28035 的有关规定。

4.1.6 系统应及时进行日常管理维护、软件维护、数据维护、运行环境维护等。

4.1.7 系统文档内容和编排应满足 GB/T 8567 的要求，系统文档管理应满足 GB/T 16680 的要求。

4.2 运行环境要求

4.2.1 硬件环境

4.2.1.1 服务器配置

4.2.1.1.1 符合国家现行标准配置，应具有可靠性、安全性、开放性、便于扩充等特性。

4.2.1.1.2 服务器的 CPU 配置应支持多核心处理器，如 Intel Xeon 系列或 AMD EPYC 系列，核心数根据系统并发处理需求配置，宜不少于 16 核，应满足数据处理、模型训练与推理的计算需求。

4.2.1.1.3 适配 NVIDIA GPU 系列（如 A10、A100 等）或其他兼容的高性能 GPU。对于分布式训练或大规模推理场景，可配置多 GPU 卡，支持单机多卡或多机多卡架构。

4.2.1.1.4 系统内存容量应不少于 64 GB，且具备高速数据读写能力，如 DDR4 或更高规格内存，在数据加载、模型训练与多任务并发执行时数据应快速存储与访问。

4.2.1.1.5 配备高速系统盘用于安装操作系统与系统软件，数据盘应采用大容量存储设备（如 HDD 或 SSD 阵列），总容量根据数据存储需求规划，应满足海量图像、视频数据及模型文件的长期存储与管理要求，支持数据的快速读写与随机访问。

4.2.1.2 网络设备

4.2.1.2.1 网络基础设施应包含高性能交换机与路由器，内部网络带宽应支持数据密集型任务（如大规模数据集传输、分布式训练数据同步）的高效进行，对外网络连接应具备足够带宽以满足系统与外部设备（如摄像头、客户端）的数据交互需求。

4.2.1.2.2 应严格按照国家有关保密政策的要求配置，应具有可靠性、安全性、开放性、便于扩充等特性。

4.2.1.2.3 涉密的数据只能在涉密网中运行，非涉密的数据才可在非涉密网中运行。

4.2.1.2.4 根据 GB/T 22239 及数据安全等级设定相应网络安全设备，建立相一致的网络环境安全管理制度。

4.2.1.2.5 建立较为完备的网络日常管理维护制度，对网络系统进行日常维护；宜配置网络管理软件，实现对网络资源进行管理维护，实现故障管理、配置管理、安全管理等方面的功能。

4.2.2 软件环境

4.2.2.1 操作系统

支持 Linux 操作系统发行版（如 Ubuntu、CentOS 等），利用其稳定的性能、多用户支持与丰富的开源软件生态，为系统运行提供基础软件平台。操作系统应进行安全配置与性能优化，包括定期更新系统补丁、调整内核参数（如网络参数、内存管理参数）。

4.2.2.2 容器化技术

采用 Docker 或 Kubernetes 等容器化技术进行系统部署与资源管理。容器引擎应保持最新版本。通过容器化技术实现系统组件的解耦与独立部署。

4.2.2.3 深度学习框架

集成主流深度学习框架，如 TensorFlow、PyTorch 等，并应支持框架的持续更新与版本兼容。深度学习框架应充分利用 GPU 计算资源，提供高效的模型训练与推理接口，支持常见计算机视觉模型（如 CNN、R-CNN 系列等）的快速开发与部署，具备丰富的模型库与工具集，同时应提供可视化工具辅助模型训练过程监控与分析。

4.2.2.4 数据库系统

宜选用 MySQL、PostgreSQL 等关系型数据库或 MongoDB 等非关系型数据库存储系统元数据、用户信息、任务配置、模型参数等结构化与非结构化数据。数据库应具备高可用性与可扩展性，支持数据的高效读写、查询优化与事务处理，数据应一致、完整，利用数据备份与恢复机制保障数据安全，应满足系统在大规模用户与复杂任务场景下的数据管理需求。

4.3 数据管理

4.3.1 基本要求

- 4.3.1.1 各类数据及元数据应按本文件的规定，建立数据库，且应满足系统高效运行和查询检索的需要。
- 4.3.1.2 数据处理完成后，应进行成果质量检查，通过后方可提交系统入库。
- 4.3.1.3 数据均应及时更新，并应保证其准确性与有效性；数据更新前应做历史数据的备份工作。
- 4.3.1.4 数据的安全保密应符合 GA/T 708 第二级基本要求的规定。

4.3.2 数据接入

- 4.3.2.1 应支持多种格式的图像（如 JPEG、PNG、TIFF 等）、视频（如 AVI、MP4、MOV 等）数据接入，同时兼容常见文本格式（如 TXT、CSV 等）与音频格式（如 WAV、MP3 等）。支持本地文件系统、FTP、OSS 等多种数据来源接入方式，应提供灵活的数据上传接口与批量导入功能。
- 4.3.2.2 应具备数据接入校验机制，对上传数据进行格式验证、完整性检查与数据质量评估，接入数据应符合系统处理要求与任务规范。对于不符合要求的数据，应及时反馈错误信息与提示，引导用户进行数据修正或重新上传。避免因数据问题导致系统运行错误或模型训练偏差。

4.3.3 数据预处理

- 4.3.3.1 针对图像数据，应提供去噪、灰度化、归一化、图像增强（如旋转、缩放、裁剪、翻转、亮度/对比度调整、添加噪声等）等预处理操作。

- 4.3.3.2 对于视频数据，应支持抽帧、帧率转换、关键帧提取等处理功能。
- 4.3.3.3 文本数据应支持进行词法分析、去除停用词、文本编码转换等预处理步骤。
- 4.3.3.4 音频数据应支持降噪、音量归一化等操作。
- 4.3.3.5 应支持用户根据具体任务需求灵活组合与配置预处理流程。
- 4.3.3.6 应支持自定义预处理算法与插件扩展机制，允许用户基于系统提供的开发接口编写特定的数据预处理逻辑，适应不同领域与特殊任务的数据预处理要求。

4.3.4 数据集管理

- 4.3.4.1 应提供数据集创建、编辑、删除、查询功能，支持数据集版本管理，记录数据集的更新历史与变更信息。用户应对数据集进行分类、标注、添加描述信息等操作。
- 4.3.4.2 应具备数据集分割功能，支持按照一定比例或规则将数据集划分为训练集、验证集、测试集，支持随机抽样、分层抽样等分割方式。

4.3.5 数据标注

- 4.3.5.1 应提供多种标注工具，包括但不限于矩形框标注、多边形标注、点标注、语义分割标注、实例分割标注等，符合不同计算机视觉任务（如目标检测、图像分类、语义分割、实例分割等）的标注需求。标注工具应具备友好的用户界面与便捷的操作方式，支持鼠标、键盘快捷键操作。
- 4.3.5.2 应支持多人协作标注，具备任务分配、进度监控、冲突解决功能，支持实时跟踪标注人员的工作进度与标注质量，对标注结果进行交叉验证与审核。同时支持智能标注功能，利用预训练的深度学习模型对数据进行预标注，辅助人工标注人员提高标注效率。

5 性能要求

5.1 系统容量

- 5.1.1 支持至少 100 个注册用户，应满足大规模团队或多用户场景下的并发使用需求。系统应具备良好的用户管理与权限控制机制。
- 5.1.2 可管理的结构化数据量应不少于 500 GB，非结构化数据量（如图像、视频数据）不低于 4 TB，适应不同规模数据存储与处理需求。系统应具备高效的数据存储架构与索引机制，随着数据量增长应实现线性或近似线性的性能扩展，系统在海量数据环境下应持续稳定运行。

5.2 响应时间

- 5.2.1 系统首页访问平均响应时间在业务正常情况下应小于 3 s，业务高峰时应小于 5 s；用户登录平均响应时间业务正常时应小于 3 s，高峰时应小于 5 s；页面打开及刷新平均响应时间业务正常时应小于 3 s，高峰时应小于 5 s；基本提交操作响应时间业务正常时应小于 2 s，高峰时应小于 3 s。
- 5.2.2 数据查询操作响应时间根据查询复杂度区分，简单查询（如基于单一条件的精确查询）响应时间不超过 5 s，复杂查询（如多条件组合查询、模糊查询）响应时间不超过 10 s；基本统计报表响应时间不超过 5 s，复杂统计报表响应时间不超过 10 s；图形统计报表响应时间不超过 15 s；模型加载响应时间不超过 30 s。

5.3 吞吐量

系统应具备根据硬件资源与网络条件自适应调整的能力，在不同业务负载下保持稳定的吞吐量。

5.4 可靠性

- 5.4.1.1 应运行稳定，准确完成多源异构数据采集、存储、管理任务，并具有容错能力。
- 5.4.1.2 系统软件及硬件的升级不应影响服务系统的运行及服务的提供。
- 5.4.1.3 数据库服务与应用服务器应采用集群部署，系统的业务场景应在线处理，系统应通过业务与数据库事务实现数据的完整一致性，系统应通过记录系统日志与系统管理工具查询系统状态。

5.5 易用性

应提供联机帮助，软件中各子系统用户界面风格应一致，应搭建统一风格的接口软件，应易学易用。

5.6 集成性

应具有开放式体系结构，提供多种数据接口，与其他通用应用软件及专业应用软件之间应实现集成。

5.7 可扩展性

结构应具有可扩展性，系统应有统一的中控系统，可实现对各个应用模块的数据维护管理。并支持通过增加服务器或换用处理能力更强的服务器的方式对系统进行扩展。组成系统的每个逻辑单元都可独立于其他单元进行升级。

6 系统架构

6.1 功能架构

系统底层通过 Gitlab、Harbor、Kubernetes 等技术实现系统集群管理，包括数据处理、算法开发、推理应用与管理后台四个功能模块，如图1所示。

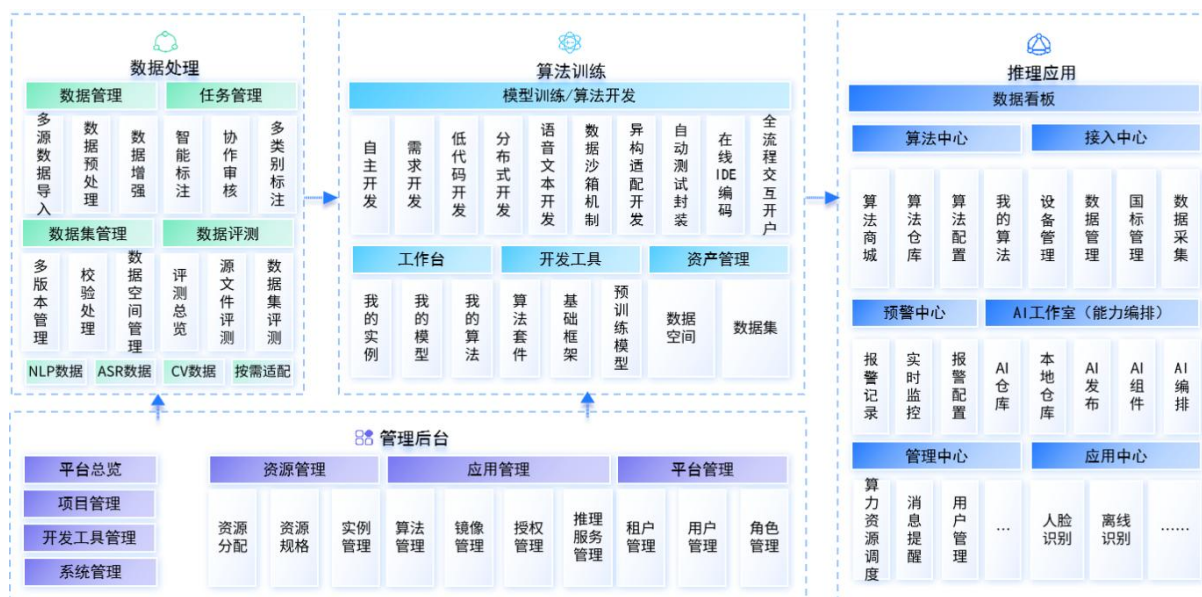


图1 功能架构图

6.2 技术架构

6.2.1 底层技术

6.2.1.1 采用分布式文件系统（如 Ceph、HDFS 等）存储大规模图像、视频数据，实现数据的分布式存储、冗余备份与高可用性，支持海量数据的高效管理与快速读写。

6.2.1.2 基于 Kubernetes 等容器编排引擎实现系统组件的容器化部署与集群管理，对系统中的数据处理、模型训练、推理服务等模块进行资源调度与生命周期管理。通过容器化技术实现组件的隔离与独立升级，在不同硬件环境下快速部署与弹性扩展系统。

6.2.2 中间层技术

6.2.2.1 设计微服务架构，将系统功能拆分为多个独立的微服务模块，如数据管理服务、模型训练服务、推理服务、用户管理服务、监控服务等，各微服务通过轻量级通信协议（如 RESTful API、gRPC 等）进行交互与协作。

6.2.2.2 引入消息队列（如 RabbitMQ、Kafka 等）实现系统组件间的异步通信与事件驱动架构，用于处理任务调度、数据更新通知、日志收集等异步任务。消息队列缓冲消息，解耦发送方与接收方。

6.2.3 上层技术

6.2.3.1 提供可视化开发环境与用户界面，支持用户通过拖拽、配置等方式进行模型开发、训练任务配置与数据处理流程设计。可视化界面应提供丰富的图表、报表展示系统运行状态、数据统计信息与模型性能指标。

6.2.3.2 集成自动化测试与持续集成 / 持续部署（CI/CD）工具链，在代码提交、模型更新时自动触发测试流程，包括单元测试、集成测试、性能测试等，确保系统代码质量与功能稳定性。CI/CD 流程实现系统的自动化构建、测试与部署，将经过测试的代码与模型快速部署到生产环境。

6.2.3.3 技术架构如图 2 所示。



图 2 技术架构图

7 功能要求

7.1 数据处理功能

7.1.1 实现数据的全流程处理，涵盖数据接入、清洗、标注、增强、转换等环节，形成完整的数据处理流水线。各环节应具备高度的可配置性与扩展性，用户可根据任务需求灵活调整处理流程与参数，支持不同类型数据与任务的定制化处理，为后续模型训练提供高质量的数据支持。

7.1.2 支持数据的实时处理与批量处理模式，对于实时视频流数据应进行在线处理与分析，及时提取视觉信息并反馈结果；同时对于大规模离线数据集，支持进行高效的批量处理操作，利用系统资源提高数据处理速度，应满足不同应用场景下的数据处理需求，如实时监控系统与离线数据分析任务。

7.2 算法开发功能

7.2.1 应提供多种主流算法框架（如 TensorFlow、PyTorch、MXNet 等）的支持，允许开发者在系统中选择合适的框架进行模型开发。

7.2.2 系统应提供框架的版本管理与环境配置功能，不同框架间应具有兼容性与独立运行，开发者应根据项目需求与自身技术栈进行灵活选择与切换。

7.2.3 具备模型训练、调试、评估、优化功能，支持分布式训练架构（如数据并行、模型并行），支持利用多 GPU、多节点资源加速模型训练过程。

7.2.4 提供可视化工具用于监控训练过程中的损失函数、准确率、召回率等指标变化，同时提供模型评估指标计算与分析功能，对训练后的模型进行全面性能评估，指导模型优化方向。

7.3 推理应用功能

7.3.1 支持将训练好的模型快速部署到推理服务器或边缘设备，提供模型的在线推理服务。

7.3.2 推理服务应具备高并发处理能力，支持同时处理多个推理请求，推理响应时间满足实时性要求，如在视频监控场景下及时对视频流中的每一帧进行目标检测与识别，输出准确的结果。

7.3.3 提供推理结果的后处理功能，如对目标检测结果进行非极大值抑制、类别筛选、结果可视化等操作，将原始推理输出转换为用户可理解与应用的形式。

7.3.4 支持与外部系统（如业务应用系统、数据库系统）的集成，通过标准接口（如 RESTful API、WebSocket 等）将推理结果推送或查询至其他系统，支持计算机视觉功能在实际业务流程中的嵌入与应用。

7.4 系统管理功能

7.4.1 实现用户管理、权限管理、资源管理功能：

- a) 用户管理包括用户注册、登录、信息管理、角色分配等操作；
- b) 权限管理应基于角色的访问控制（RBAC）模型，对不同用户角色授予相应的系统功能访问权限与数据操作权限；
- c) 资源管理对系统硬件资源（如 CPU、GPU、内存、存储）和软件资源（如算法库、模型库、数据集）进行监控、分配与调度。

7.4.2 具备系统监控与日志管理功能，对系统运行状态（如 CPU 使用率、GPU 负载、内存占用、网络流量等）进行实时监控与可视化展示，及时发现系统故障与性能瓶颈，并通过告警机制通知管理员进行处理。

7.4.3 日志管理记录系统操作、任务执行、错误信息等日志数据。

8 安全要求

8.1 信息安全

应符合GB/T 20270、GB/T 20988的相关要求。

8.2 系统安全

8.2.1 采用身份认证、授权管理、数据加密、访问控制等安全技术，采用多因素认证（如密码、短信验证码、指纹识别等），并对用户操作进行审计和追溯，防止非法操作和数据泄露。

8.2.2 基于角色的访问控制（RBAC）应精确到系统功能模块与数据资源级别，用户只能在其授权范围内操作，定期审查与更新用户权限。

8.2.3 访问控制应满足以下要求：

- a) 应支持基于调度员、监控员、运维人员、审计管理员、系统管理员等角色的访问控制功能；
- b) 应支持角色与权限的绑定，不同角色人员应按照工作范围、职责分工分配相应的访问控制权限；
- c) 应支持角色互斥功能，禁止配置同时具有控制和维护修改权限的角色；
- d) 应依据安全策略控制用户对监控信息等文件或数据库表等客体的访问。

8.3 网络安全

8.3.1 部署防火墙、入侵检测/防御系统（IDS/IPS）、防病毒软件等网络安全设备，对网络流量进行访问控制与安全监测。

8.3.2 防火墙应根据安全策略规则允许或拒绝网络数据包的进出，阻止外部非法访问与内部敏感信息泄露；IDS/IPS 实时监测网络流量，识别并防范端口扫描、恶意攻击（如 DDoS、SQL 注入、XSS 攻击等），及时发出警报并采取相应防护措施。

8.3.3 采用虚拟专用网络（VPN）技术保障远程访问的安全，对传输数据进行加密封装，数据在公网传输过程中应具有保密性与完整性。VPN 应支持多种认证方式（如用户名/密码、数字证书等），加强用户身份验证，防止非法用户接入系统网络。

8.4 算法安全

8.4.1 对开发的计算机视觉算法进行安全审计与验证，算法应正确、稳定与安全。审计过程应涵盖算法逻辑、数据处理流程、模型训练机制等方面，评估算法在不同输入情况下的行为表现，算法输出应符合预期且无安全隐患。

8.4.2 在算法开发与部署过程中，防范对抗样本攻击，采用对抗训练、输入数据预处理等技术增强算法的鲁棒性，使模型能抵御恶意构造的对抗样本干扰，在复杂环境与恶意攻击场景下算法应具有准确性与稳定性。

8.5 数据安全及备份恢复

8.5.1 数据保密性

对公民身份信息、活动信息等敏感数据，应进行加密存储。

8.5.2 数据安全

采用数据加密技术对敏感数据（如用户信息、标注数据、模型参数等）进行存储加密与传输加密，数据在静态存储与网络传输过程中应保密。加密算法应选用符合行业要求的高强度加密算法（如 AES、RSA 等），密钥管理应遵循安全规范，定期更新密钥，防止数据泄露与恶意攻击。

8.5.3 数据备份恢复

数据备份恢复应包括：

- a) 应有数据备份机制，定期对系统数据进行全量或增量备份，备份数据存储在异地或独立的存储介质中；
- b) 数据备份与恢复过程应进行完整性与一致性校验，在使用恢复的数据前应校验其可用性、完整性；
- c) 日志数据、采集数据、基础数据、主题数据、业务数据和知识库数据采用每日全量备份的策略备份；
- d) 采用反向代理技术实现 WEB 集群服务的负载均衡，支撑数据存储、处理、运算及应急处置的系统应保证硬件冗余，避免关键节点存在单点故障。

9 测试与运行维护

9.1 软件测试

按GB/T 22239、GB/T 22240规定的安全级别确定检查和测试的项目。

9.2 测试方法

9.2.1 制定全面的测试计划，包括单元测试、集成测试、系统测试、性能测试、安全测试等环节：

- a) 单元测试针对系统的最小功能单元（如函数、类）进行测试；
- b) 集成测试验证系统组件间的交互与集成是否正常；
- c) 系统测试从用户视角对系统整体功能进行测试；
- d) 性能测试评估系统在不同负载下的响应时间、吞吐量等性能指标；
- e) 安全测试检查系统在数据安全、网络安全、系统安全、算法安全等方面的防护能力。

9.2.2 采用自动化测试工具与手动测试相结合的方式，对于重复性测试任务（如回归测试）利用自动化测试脚本提高测试效率与覆盖率，如使用 Selenium 进行 UI 自动化测试、JUnit 进行单元测试自动化等。

9.2.3 对于复杂业务逻辑与安全场景，结合手动测试进行深入分析与验证。

9.3 运行维护

9.3.1 基本要求

9.3.1.1 系统应具有运行维护能力，主要包括运行维护能力、运维准备、运维执行、运维验收、运维改进和运维过程管理。

9.3.1.2 运行维护的过程管理应至少包含服务级别管理、报告管理、事件管理、问题管理、配置管理、变更管理、信息安全管理等内容。

9.3.2 日常维护

建立系统维护团队，负责系统的日常维护工作，包括服务器维护、数据库维护、应用程序维护、网络维护等。应做好日常的监控、检查和维护工作，每月进行项目文档的归档、每天监控项目运行日志，并分析可能发生的异常情况。

9.3.3 程序代码可维护

代码编写格式系统统一规范，重要代码需注释，提高程序的可读性，便于维护。采用代码版本控制软件对代码版本进行控制。

9.3.4 运行故障应急处理

对于系统运行故障，需要做好应急处理预案，确保小故障1 h内恢复，一般故障6 h内恢复，灾难性故障1 d内恢复，并详细排查故障原因，做好总结完善工作。
