

应用系统工程 应用软件架构设计规范

Application system engineering-Application software architecture
design specification

(征求意见稿)

(本草案完成时间: 2025-6-17)

在提交反馈意见时, 请将您知道的相关专利连同支持性文件一并附上。

XXXX - XX - XX 发布

XXXX - XX - XX 实施

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 需求分配	2
5.1 需求分配原则	2
5.2 需求分配过程	3
5.3 需求分配结果	3
6 应用软件系统架构设计要求	3
6.1 一般要求	3
6.2 用户特征	4
6.3 约束条件	4
6.4 已有信息资源利用	5
6.5 采购限制	5
7 设计方法与设计图	5
7.1 设计方法	5
7.2 极限状态	5
7.3 应用软件系统架构设计图	6
8 应用软件系统指标体系	7
8.1 数据生产能力	7
8.2 极限状态	8
8.3 软件与数据质量指标	8
8.4 系统运行状态与模式	8
8.5 界面与交互指标	8
8.6 综合技术经济指标	8
9 功能规模与应用软件系统规格	8
9.1 功能规模	8
9.2 应用软件系统规格说明要求	8
9.3 非功能系统规格说明	9

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。

本文件由辽宁省工业和信息化厅提出并归口。

本文件起草单位：大连昆腾数据科技有限公司、大连市大数据中心、大连市友谊医院、大连市财政事务服务中心、国家计算机网络与信息安全管理中心辽宁分中心、中国（辽宁）自由贸易试验区营口片区管委会数字经济发展局、营口自贸区建设发展有限公司、荣科科技股份有限公司、大连正润科技有限公司、大连华屹工程咨询有限公司、辽宁雨溪项目管理咨询有限公司、大连新闻传媒集团、中电云计算技术有限公司、大连市甘井子区大数据中心、大连市工程咨询协会、大连软件行业协会、大连竹方工程咨询有限公司、大连亿丰科技有限公司。

本文件主要起草人：于锋、贾瑄、段刚、谭跃、王璐、王江、赵世宏、娄巍、赵鹤、吉庆、葛文跃、荣宪波、王启章、张文革、马双翼、徐慧、刘俊江、王佳驹、宋悦、吉长军、刘斌、冯艳爽、朱明、时凤燕、邢宇、裴鑫、王鹏、田野、杨光、陈春声、智靖淼、曹秀坤、孙建峰、陈晓云、赵志明、朱善彬、宋惠妍、武毅。

本文件发布实施后，任何单位和个人如有问题和意见建议，均可以通过来电和来函等方式进行反馈，我们将及时答复并认真处理，根据实际情况依法进行评估及复审。

本文件归口单位通讯地址：沈阳市北陵大街45-2号，联系电话：024-86913384

本文件起草单位通讯地址：大连市沙河口区星河二街25号1单元38层6-1,5-2号，联系电话：0411-84119516

应用系统工程 应用软件架构设计规范

1 范围

本文件规定了应用软件架构设计的主要内容、过程及编制深度要求，包括：需求分配、应用软件架构设计、应用系统架构设计、指标体系等方面内容。

本文件适用于新建、扩建和改建的应用系统工程。

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 25000.12 系统与软件工程 系统与软件质量要求和评价（SQuaRE）第12部分：数据质量模型
- GB/T 25000.22 系统与软件工程 系统与软件质量要求和评价（SQuaRE）第22部分：使用质量测量
- GB/T 25000.23 系统与软件工程 系统与软件质量要求和评价（SQuaRE）第23部分：系统与软件产品质量测量
- DB21/T 3756-2023 应用系统工程 应用软件工程量清单规范
- DB21/T 3757-2023 应用系统工程 工程造价咨询规范
- DB21/T 3758-2023 应用系统工程 初步设计文件规范
- DB21/T 3759-2023 应用系统工程 关键性评价指标
- DB21/T XXXX-2024 应用系统工程 工程架构设计规范

3 术语和定义

DB21/T 3758-2023、DB21/T 3759-2023和GB/T 25000系列界定的以及下列术语和定义适用于本文件。

3.1

应用软件架构 application software architecture

应用软件的基本概念和属性以及实现和演化原则。

3.2

规格说明 specification

实体的特定品质、技术细节的详细描述。根据上下文，有时也可以简写为规格。

3.3

业务需求规格说明 business requirements specification

结构化地收集业务或任务要求（业务或任务问题、机会定义、概念和解决方案所需条件）及其与外部环境的关系。

3.4

利益相关方需求规格说明 stakeholder requirements specification

利结构化地收集利益相关方的要求（需求、特征、上下文、概念、限制和优先级）及其与外部环境的关系。

3.5

系统需求规格说明 system requirements specification

结构化地收集系统及其操作环境和外部接口的要求（功能、性能、设计约束和属性）。

3.6

软件需求规格说明 software requirements specification

结构化集合，包含软件及其外部接口的基本需求（功能、性能、设计约束和属性）。

3.7

NESMA 方法 NESMA method

一种功能点分析方法，包括三种类型：预估功能点分析、估算功能点分析、详细功能点分析。

参见：《GB/T 42588-2023 系统与软件工程 功能规模测量NESMA方法》。

3.8

基线需求 baseline requirements

需求的正式批准版本。

4 基本要求

应用软件架构设计应符合以下基本要求：

- a) 已经完成了需求调查，需求已经被验证和确认。业务需求规格说明、利益相关方需求规格说明、系统需求规格说明和软件功能规格说明应符合 IS029148 要求；
- b) 应用软件架构设计一般过程包括：需求分配、应用软件架构设计、应用系统架构设计、应用系统指标体系设计；
- c) 使用 NESMA 方法进行估算应用软件功能规模；
- d) 须根据应用软件系统数据分级分类结果，将核心数据、重要数据的处理划分为独立的应用软件系统；
- e) 应明确应用系统架构设计方法；
- f) 应用系统的网络安全等级保护为 3 级或以上时，应单独提出密码应用保护方案；
- g) 应用系统运行模式至少应包括常规、降级和紧急等运行模式；
- h) 应提出业务运营政策和规则调整的处理方法，降低业务流程变动带来的应用软件系统变更风险；
- i) 应提出应用软件系统的交付方式要求；
- j) 设计结果不应包含任何限制供应商选择的描述。

5 需求分配

5.1 需求分配原则

需求分配应遵循以下原则：

- a) 完整性。所有列入需求基线中的需求均应被分配，不应有遗漏。原型系统需求作为项目建设过程需求，不列入需求基线，但也不应被遗漏；

- b) 一致性。所有需求在分配过程中应保持一致，如需求进行了修改，应通过唯一标识保持利益相关方对需求理解的一致性。如需求被删除，则相应的唯一标识不应被删除，保持利益相关方对这一事实的理解保持一致；
- c) 可追溯性。应通过维护需求基线表，保证需求分解、分配过程中的可追溯性。

5.2 需求分配过程

需求分配的一般过程包括：

- a) 对需求规格进行检查，每一项需求都应具有以下特征：
 - 1) 必要。该需求定义了一个基本功能/能力、特性、限制或质量因素；
 - 2) 合适。需求与它涉及的实体相匹配；
 - 3) 明确无误。需求的描述不应有“待确定”、“待解决”、“待分配”等字样；
 - 4) 完整。该需求充分描述了实体所需的必要能力、特性、约束或质量等因素，无须其他信息即可理解该需求；
 - 5) 单一。需求应陈述一个单一的能力、特性、限制或质量因素，不能同其他需求重复；
 - 6) 可行。该需求可以在系统限制内实现（如：成本、时间、技术、法律、监管等），风险可接受；
 - 7) 可验证。需求可以被证明为客户满意或符合客户要求。当该需求可测量时，可验证性得到增强。
- b) 对需求项进行分解和分类。需求分类包括：功能、性能、接口、过程要求、非功能性等；
- c) 对需求项进行排序或加权，以表示优先级、时间顺序或相对重要性；
- d) 将需求划分为需求集。根据业务场景、用户类型等因素，将需求划分为不同需求集，使利益相关方能够更好的达成一致理解，为需求验证、设计验证和实施提供更好的描述；
- e) 以需求集为单位，将需求分配到应用软件系统架构模型中并将需求项分配给子系统和基础设施元素；
- f) 迭代和递归。在需求分配、应用软件系统架构设计和工程架构设计过程中进行适当的迭代和递归，必要时进行性能测试，以获得最优的设计方案；
- g) 验证。通过评审会等形式，验证可以在限制范围内实现这些需求。

5.3 需求分配结果

以图表的形式说明每个需求集明细及分配结果。

需求项可能被多次分配，但不应重复计算功能规模。

6 应用软件系统架构设计要求

6.1 一般要求

应按业务场景、用户类型等因素，将需求集划分为不同的应用软件系统：

- a) 以图表的形式给出应用软件系统架构模型图和数据实体模型图，明确主要作业、业务量及用户数量等，定义作业系统/子系统、非功能系统及其对应的数据实体；
- b) 应将数据管理需求划分为单独的数据管理系统。如建设单位已有数据管理系统，则应将这部分需求视为对原数据管理系统的增加需求；
- c) 业务数据中如果包含了核心数据和重要数据，则应将这部分需求划分为单独的应用软件系统。

注1：核心数据、重要数据是指符合《数据安全法》规定的的数据分类分级保护制度要求的数据。

应用软件系统通常由若干子系统构成。每个子系统应体现作业/任务的主要特征。应从以下几个方面描述应用软件系统：

- a) 主要功能，并按 NESMA 方法估算功能规模；
- b) 明确运行环境与系统运行限制条件：
 - 1) 系统接口\硬件接口\通信接口；
 - 2) 与其他软件或服务的接口（与 SaaS 或云服务的交互）；
 - 3) 内存与存储容量要求。
 - 4) 其他要求，如：不低于 B 级数据中心、不低于 R2 级互联网数据中心级别等；
- c) 数据生产能力等性能效率指标要求；
- d) 指定运行模式下的检查点、控制方式、可用功能、效率等。；
- e) 确定应用系统恢复和重启功能的有效性、效率和满意度标准；
- f) 质量要求，包括：产品质量、数据质量和使用质量测度指标总体要求。

6.2 用户特征

6.2.1 用户识别与用户界面实例

识别应用软件系统用户类型，包括：使用者、操作员、维护人员（按功能、位置、设备类型）等，并对用户进行分组、划分用户群体：

- a) 描述每个用户组的人数以及他们使用系统的性质，对用户一般特征进行描述，如：教育水平、经验、技术专长和生理能力等可能影响可用性的特征；
- b) 提出用户组的功能权限、身份鉴别、操作模式等内容，描述用户组的使用情境、适用条件和约束；
- c) 识别用户特征并定义用户群体，如：技能、母语、生理特征等；
- d) 定义用户界面实例：根据用户群体特征，描述用户界面对应的用户群体及信息呈现设备、输入设备等；
- e) 提出不同用户群体的使用情境、适用条件和约束。

6.2.2 用户界面与交互设计要求

通常，应用软件系统无须额外的设计就可以满足大多数用户交互需要。用户界面与交互设计的目的是为了适应用户群体的复杂性，提供更多的使用情境和交互功能，提升应用系统的用户体验和总体满意度。

在架构设计阶段，应对实施阶段的用户界面与交互验证与确认提出要求：

- a) 一致性要求，包括外部一致性和内容一致性；
- b) 功能性要求；
- c) 可用性要求，如：有效性、效率、满意度；
- d) 可访问性要求，如：周境完备性、使用周境灵活性等；
- e) 用户界面与交互设计验证原型要求，如：可用性、功能完整性、逻辑流畅性、反馈准确性。

注2：更多信息请参阅GB/T 19878.11-2023。

6.3 约束条件

指定由外部标准、监管要求或项目限制对系统设计施加的约束：

- a) 法律依据及主要条文的约束；
- b) 监管需要或约束；

- c) 应遵循的标准规范名称、条文的约束；
- d) 其他正在执行的标准规范；
- e) 已有的概念模型/实体模型/数据模型/数据架构等，对系统的限制或要求；
- f) 对数据资产/主数据/参考数据的可用性和可访问性的限制；
- g) 描述在成本和时间范围内执行项目的限制；
- h) 其他限制，如：技术限制、资源限制、人员限制等。

6.4 已有信息资源利用

描述应用软件系统对已有信息资源的利用情况，包括：

- a) 应利用的数据资产、主数据/参考数据资源、已有资料等；
- b) 应复用的开发资源，如：构件/组件库、数据库、接口等。

6.5 采购限制

提供任何将限制供应商选择的其他项目的总体描述，包括：

- a) 资源限制，如：特定的设备配置、具有唯一来源的支撑软件等。
- b) 与其他应用程序的接口。
- c) 质量要求，如：可靠性、兼容性等特定质量检测要求。
- d) 来自其他系统的限制，如：通过接口从受控系统获取信息的实时要求。

7 设计方法与设计图

7.1 设计方法

应使用极限状态设计方法或概率极限状态设计方法。必要时，也可考虑使用限额设计方法、指标设计方法等。

可根据数据生产能力、性能效率、操作限制与运行模式、采购限制等实际需求，提出不同的极限状态测度指标，以实现不同的架构设计。

当采用限额设计方法、指标设计方法时，应对非功能需求进行剪裁以实现架构设计。

应用系统架构设计时，须对需求项进行分解和确认，因此应同需求调研、需求分配、应用软件架构设计等过程进行迭代和递归，以获得最佳的应用系统架构设计方案。

7.2 极限状态

应用系统极限状态应以主要作业运行资源要求为基础，提出不同业务量、用户数量情况下的资源配置要求并进行验证。验证时，可以参考设备供应商、支撑软件供应商、类似应用软件产品供应商等供应商的技术资料，也可以利用测试环境或运行环境对设备、类似应用软件产品进行测试。

当市场无类似应用软件或采用新设备时，可以采取以下方法验证设备配置要求：

- a) 调整运行环境建设工期并对应用软件原型及主要功能性能进行性能测试；
- b) 对新设备进行性能与质量测试；
- c) 可以提出实验环境设计方案并开展测试。

7.2.1 正常运行极限状态资源计算

正常运行极限状态资源计算要求如下：

- a) 计算应用软件系统的数据生产能力指标，包括：数据生产能力、最大数据生产能力、最小连续生产时间、最大在线用户数、最大并发用户数；
- b) 计算最大数据生产能力指标下的 CPU/GPU、内存、显存、容量、网络带宽等资源配置参数；
- c) 描述数据生产能力、资源配置参数的测度指标要求：
 - 1) 最大数据生产能力、最大在线用户数和最大并发用户数；
 - 2) 主要作业用户最长可忍耐响应时间和最长可忍耐作业完成时间。最长可忍耐响应时间一般不应超过 15 秒。最长可忍耐作业完成时间应根据应用系统数据生产能力进行计算；
 - 3) 主要作业的平均周转时间、平均吞吐量等指标；
 - 4) 访问控制性、数据加密正确性、数据完整性、内部数据抗讹误性、缓冲区溢出防止率、数字签名使用率、系统日志保留满足度等信息安全性测度要求；
 - 5) 设备可访问性、数据更新的时间延迟、数据值可跟踪性、数据可用性比率、数据可用概率、数据可恢复性比率等依赖应用系统的数据质量测度要求。

7.2.2 承载能力极限状态

一般情况下，承载能力极限状态指标应比正常运行极限状态指标提高30%。可不考虑平均周转时间、吞吐量等指标。

承载能力极限状态指标应符合《DB21/T XXXX-2024 应用系统工程 工程架构设计规范》第4.3.2节要求。

7.3 应用软件系统架构设计图

应明确设计的技术依据，简要描述架构设计的特点、优势以及满足需求的情况。

应用系统架构设计图应包括：架构设计系统图、密码应用方案、界面与交互设备设施、基础设施设计内容说明。

7.3.1 系统图/拓扑图

根据应用软件系统架构模型图，给出系统图或拓扑图，清晰、完整体现应用软件系统的分配及相互关系。

应按照《DB21/T 3758-20231 应用系统工程 初步设计文件规范》6.4节、6.5节、6.7节要求绘制应用软件系统架构图纸。

此外，还应遵循：

GB/T 1526-1989 信息处理 数据流程图、程序流程图、系统流程图、程序网络图和系统资源图的文件编制符号及约定。

7.3.2 密码应用方案

密码应用保护级别在3级或以上时，须单独描述密码应用方案。在2级或以下时，须在应用软件系统的安全性中体现密码应用设计：

- a) 明确应用系统密码保护级别及要求：
 - 1) 明确物理和环境安全、网络和通信安全、设备和计算安全等基础设施要求；
 - 2) 明确应用和数据安全细节：身份鉴别、访问控制信息完整性、重要信息资源安全标记完整性、重要数据传输机密性完整性、重要数据存储机密性完整性、不可否认性、密码服务、密码产品等内容；
 - 3) 明确管理制度、人员管理、建设运行、应急处置等要求；

- b) 识别并定义应用系统的身份鉴别信息、密钥数据、重要数据等机密性保护对象；
- c) 识别并定义应用系统的身份鉴别信息、密钥数据、日志记录、访问控制信息、重要信息资源安全标记、重要可执行程序、视频监控记录、门禁记录、传输的重要数据、应用系统存储的重要数据等完整性保护对象；
- d) 识别并定义数据原发行为和数据接收行为等不可否认性保护对象；
- e) 识别并定义真实性技术要求场景：
 - 1) 进入重要物理区域人员的身份鉴别；
 - 2) 通信双方的身份鉴别；
 - 3) 网络设备接入时的身份鉴别；
 - 4) 重要可执行程序的来源真实性保证；
 - 5) 登录操作系统和数据库管理系统的身份鉴别；
 - 6) 应用系统用户身份鉴别。
- f) 确定应用系统的机密性、完整性保护对象和真实性场景所需要的密码服务或密码产品要求；
- g) 估算密码应用相关的功能规模，调整应用软件系统功能规模，提出密码服务、密码产品的规格要求。

7.3.3 界面与交互设备设施

根据第6.2节要求，提出须配置的信息呈现、信息控制等输入输出设备：

- a) 信息控制设备，即输入设备，如：键盘、鼠标、定位器、操纵杆、轨迹球、触控板、数位板和覆盖层、触控屏、触控笔和光笔，以及语音和手势控制设备、体感设备等；
- b) 信息呈现设备，即输出设备，如：视觉显示设备、听觉输出设备、触觉反馈设备等。

7.3.4 基础设施设计内容说明

应按照《应用系统工程 工程架构设计规范》（立项编号2024013），说明基础设计设计内容。

8 应用软件系统指标体系

通过量化指标全面体现应用系统的设计与施工质量，称之为应用系统的指标体系。指标体系包括：

- a) 数据生产能力；
- b) 极限状态；
- c) 软件与数据质量指标；
- d) 系统运行状态与模式；
- e) 界面与交互指标；
- f) 综合技术经济指标。

应用系统指标应该在完成系统性能优化、设备配置调整后进行测度，以反应应用系统最优运行状态。

8.1 数据生产能力

数据生产能力类指标包括：年数据生产能力、日数据生产能力、数据生产能力、最大数据生产能力、最小连续生产时间、最大在线用户数、最大并发用户数等指标构成：

- a) 应根据《DB21/T 3759-2023 应用系统工程 关键性评价指标》，提出每个应用系统的指标要求；
- b) 应提出应用系统的年度数据资产指标和设计使用年限内总数据资产指标；
- c) 宜提出主数据、参考数据相关的指标，如：年度主数据数量规模、参考数据贡献规模等。

8.2 极限状态

按照第7.2节要求确定应用系统极限状态指标。

8.3 软件与数据质量指标

按照GB/T 25000系列标准确定测度与指标值，包括：产品质量、数据质量和使用质量。

8.4 系统运行状态与模式

定义应用软件系统支持的运行模式：

- a) 描述对操作时间的限制；
- b) 可能受到的安全限制；
- c) 对用户操作人数的限制；
- d) 对计算资源的限制；
- e) 对运行地点或基础设施的限制；
- f) 运行模式对应的用户类别、功能、数据处理权限。

8.5 界面与交互指标

描述设计时应提出的应用系统界面与交互指标，包括：

- a) 有效性测度，包括：任务完成率、目标实现率；
- b) 按用户群体提出效率测度，包括：任务用时与时间效率指标、成本效率；
- c) 按用户群体提出满意度测度，包括：总体满意度、特征满意度、自主使用率、特征利用率、用户愉悦性、身体舒适性等。

此外，宜提出运行阶段的用户界面与交互指标，包括：

- a) 有效性测度，包括：任务中的差错数、出错任务率、任务差错密度；
- b) 按用户群体提出效率测度，包括：生产性时间比、非必要动作率、疲劳影响；
- c) 按用户群体提出满意度测度，包括：用户投诉率、具体特征用户投诉率、用户可信性等。

8.6 综合技术经济指标

定义应用系统的综合技术经济指标，包括：按功能规模计算的功能点综合单价指标、按年数据生产能力计算的造价指标、应用软件投资占比、基础设施投资占比等。

给出综合技术经济指标计算公式、计量要求。

9 功能规模与应用软件系统规格

9.1 功能规模

汇总密码应用方案、界面与交互实例和信息呈现与输入设备提出的功能规模，对应用软件系统规格说明进行调整，确保应用软件系统规格完整、唯一、不重复，给出NESMA功能规模估算过程和估算结果。

以附表的形式提出应用软件系统功能规格说明和功能规模估算结果。

9.2 应用软件系统规格说明要求

应用软件系统规格说明应包括以下内容：

- a) 功能规格说明。给出软件功能基本操作要求，包括：

- 1) 主要功能集功能规模。按照 NESMA 功能规模估算方法要求,描述内部逻辑文件外部逻辑文件、数据元素类型数量、信息记录类型数量等内容;
- 2) 输入的有效性检查;
- 3) 对异常情况的响应,包括:内存溢出、通信设施故障、硬件故障和失效、错误处理和恢复等内容;
- b) 非功能规格说明。提出非功能需求的分解要求;
- c) 数据管理规格说明。提出数据架构需求的分解要求;
- d) 外部接口规格说明;
- e) 运行环境要求;
- f) 技术支持说明。

这些要求作为应用系统在系统规格设计阶段的调研内容,应明确调研结果要求,包括:

- a) 准确描述每个软件功能对应的内部逻辑文件、外部逻辑文件以及详尽的数据元素类型、信息记录类型;
- b) 每项软件功能对应的输入、输出、外部查询以及数据元素类型、信息记录元素类型等内容;
- c) 以图表的形式描述应用软件系统规格需求调研结果要求并以附件的形式提供。

9.3 非功能系统规格说明

为保证业务功能质量,通常应构建以下非功能子系统:作业控制系统、数据管理系统、用户管理系统、日志与审计系统、系统运行管理系统、状态监测系统等。

a) 作业控制系统

按照业务政策和规则,对作业系统的操作顺序和操作规则进行调整,使得作业系统能够更好的适应业务流程和业务运营环境。应说明以下内容:

- 1) 数据模型;
- 2) 作业流程控制;
- 3) 作业操作规程调整;
- 4) 作业优先级。

b) 数据管理系统

通常应构建独立的逻辑数据库以存储和管理主数据/参考数据、年度数据集、日志数据等数据资产,避免在执行大数据分析、AI数据加工、数据资产加工等过程中影响作业系统性能。应说明以下内容:

- 1) 数据模型;
- 2) 主数据/参考数据访问;
- 3) 数据资产生产;
- 4) 数据分类分级;
- 5) 大数据分析;
- 6) AI 数据加工;
- 7) 日志管理。

c) 用户管理系统

构建独立的用户管理系统有助于实现应用系统内的单点登录和更好的用户体验。如果组织已有用户管理系统时,应考虑复用。应基于用户界面实例说明以下内容:

- 1) 数据模型;
- 2) 用户群体划分结果;
- 3) 复用已有组件或开发可复用组件,如:用户信息管理、登录与注销、身份鉴别、个人隐私与权利保护、数据权限等。

d) 日志与审计系统

完善的日志系统对提高应用软件系统质量有至关重要的作用，对提升应用软件系统的经济效益也有巨大帮助。应用系统应构建完整的日志体系并对日志进行管理和分析：

- 1) 数据模型；
- 2) 日志规则；
- 3) 日志完整性约束；
- 4) 日志保留要求；
- 5) 安全访问权限；
- 6) 日志分析；
- 7) 日志监测；
- 8) 审计功能。
- 9) 日志存储位置与资源占用；
- 10) 可复用的日志接口等。

e) 系统运行管理系统

为应对突发业务状况、保证主要作业系统性能和系统状态，应构建不同的系统运行模式，以保证不同优先级的作业能够更好地运行。应说明以下内容：

- 1) 数据模型；
- 2) 运行模式与切换；
- 3) 系统状态切换，如：运营准备、营业、日终、年终等；
- 4) 操作时间限制。

f) 状态监测系统

为保证系统可靠、稳定运行，应构建独立的系统运行状态监测系统，对监测指标进行监测：

- 1) 数据模型；
 - 2) 数据生产能力指标监测；
 - 3) 响应时间监测；
 - 4) 平均吞吐量指标监测；
 - 5) 信息安全性指标监测，如：缓冲区溢出监测、关键变量数据完整性监测等；
 - 6) 数据质量指标监测，如：设备可访问性、数据更新延迟等；
 - 7) 承载状态指标监测，如：CPU 平均利用率、内存平均利用率、可利用存储容量等。
-