

ICS 35. xxx
CCS Lxx

团 体 标 准

T/ISC XXX—XXXX

数字孪生低代码平台技术要求

Technical requirements for digital twin low code platform

(征求意见稿)

2024-09-03

XXXX - XX - XX 发布

XXXX - XX - XX 实施

中国 互 联 网 协 会 发 布

目 次

前 言	III
引 言	V
数字孪生低代码平台技术要求	1
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
3.1 数字孪生 digital twin	1
3.2 低代码 low code	1
3.3 零代码 no code	1
3.4 端渲染 client-side rendering	1
3.5 云渲染 server-side rendering	1
3.6 场景编辑 scene editing	1
3.7 场景服务 scene service	2
3.8 应用编辑 application editing	2
3.9 应用服务 application service	2
4 符号和缩略语	2
5 总体要求	2
5.1 技术框架	2
5.2 平台稳定性要求	3
5.3 用户界面和交互设计要求	3
5.4 安全性要求	4
5.5 扩展性和灵活性要求	4
6 技术要求	4
6.1 场景构建	4
6.2 应用开发	6
6.3 性能要求	8
参 考 文 献	10

前 言

本文件按照GB/T 1.1-2020给出的规则起草。

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这些专利的责任。

本文件由中国互联网协会提出并归口。

本文件主要起草单位：

本文件主要起草人：

引 言

数字孪生低代码平台是采用图形化、拖拉拽方式，支持用少量代码扩展实现数字孪生场景搭建和可视化应用开发的数字技术工具平台。为提升数字孪生低代码平台产品供给质量，促进数字孪生应用高质量发展，有必要制定《数字孪生低代码平台技术要求》团体标准。

数字孪生低代码平台技术要求

1 范围

本标准规定了数字孪生可视化低代码软件平台的技术要求。

本标准适用于数字孪生低代码平台产品的设计、研发、测试、评估和验收，可用于各应用领域低代码平台产品的能力测试和选型评估。

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 43441.1-2023 信息技术 数字孪生 第1部分：通用要求

T/ISC 0022-2023 数字孪生城市平台技术要求

3 术语和定义

下列术语和定义适用于本文件。

3.1 数字孪生 digital twin

现实世界特定目标实体的数字化表达。

[来源:GB/T 43441.1-2023,3.4,有修改]

3.2 低代码 low code

用少量代码即可快速开发系统，并可以实现快速配置和部署的技术和工具。

3.3 零代码 no code

无需编写代码即可创建应用程序的开发方式，通常通过图形化界面和预定义的组件实现。

3.4 端渲染 client-side rendering

在客户端设备上进行处理和渲染的技术。此方式依赖于客户端设备的处理能力和性能。适合于孪生场景范围适中（如园区、港口码头）且有多终端并发需求的应用场景。

3.5 云渲染 server-side rendering

在服务器端进行处理和渲染，并将渲染结果传输到客户端显示的技术。此方式通常能够提供更高的性能和一致的用户体验。适合于孪生场景范围较大（如城市）且有高渲染效果需求的应用场景。

3.6 场景编辑 scene editing

使用场景编辑器对三维模型和虚拟环境进行创建、修改和配置的过程。

3.7 场景服务 scene service

提供三维场景数据存储、管理和实时渲染的服务器端服务。

3.8 应用编辑 application editing

使用应用编辑器对业务应用和数据展示进行创建、修改和配置的过程，支持多种图表和数据展示形式。

3.9 应用服务 application service

提供业务应用数据存储、管理和运行的服务器端服务。

4 符号和缩略语

下列缩略语适用于本文件。

API: 应用程序编程接口 (Application Programming Interface)

CGCS2000: 2000国家大地坐标系 (China Geodetic Coordinate System 2000)

GIS: 地理信息系统 (Geographic Information System)

HTTP: 超文本传输协议 (Hypertext Transfer Protocol, HTTP)

OGC: 开放地理空间信息联盟 (Open Geospatial Consortium)

OSGB: 开放场景图形二进制格式 (Open Scene Graph Binary)

OSGeo: 开源地理空间基金会 (The Open Source Geospatial Foundation)

PBR: 基于物理的渲染 (Physically Based Rendering)

TMS: OSGeo提出的切片地图服务 (Tile map Service)

WGS84: 世界大地测量系统1984版本 (World Geodetic System 1984)

WMTS: OGC提出的Web 地图瓦片服务 (Web Map Tile Service)

5 总体要求

5.1 技术框架

数字孪生低代码平台的设计应满足高稳定性、高安全性、良好的扩展性和灵活性等要求，支持广泛的应用场景。技术框架如图1，包括场景构建部分和应用开发部分，均为低代码平台的必要组成部分，确保平台的整体性能和用户体验。



图1 数字孪生低代码平台技术框架

5.1.1 场景构建

场景构建部分负责数字孪生三维场景的创建、管理、发布，包括场景编辑、场景服务两部分。

（根据技术选型的不同，目前主流渲染架构分为端渲染和云渲染两种，对场景构建的要求会略有差异，后面功能单独针对云渲染的会标出。）

- 场景编辑:提供图形化界面，支持用户创建、编辑和配置数字孪生三维场景。场景编辑器应支持多种三维模型格式的导入和处理，具备灵活的场景配置功能。
- 场景服务:用于存储和管理数字孪生三维场景数据，支持高效的数据检索、更新和实时渲染。场景服务应具备高效的三维渲染能力，确保三维场景与实际物理状态的实时同步。

5.1.2 应用开发

应用开发部分负责业务数据的管理和展示，根据用户使用场景不同，分为零代码应用开发和低代码应用开发两部分。

- 零代码应用开发无需编程操作，通过托拉拽的方式实现业务数据的管理和展示。其组件包括：应用编辑、应用服务。
- 低代码应用开发采用少量代码，通过平台提供的统一开发 API 实现业务数据的管理和展示，提升平台的开放性和可扩展性。其组件包括：统一开发 API、API 调试。

5.2 平台稳定性要求

系统可靠性：平台应具备高可靠性，能够在长时间运行和复杂环境下保持稳定。应支持自动故障检测和恢复机制，确保系统的持续可用性。

容错能力：平台应具备良好的容错机制，能够在部分组件或功能故障时保持核心功能的可用性。

持久性：平台应确保数据的持久性和一致性，防止数据丢失或损坏。

5.3 用户界面和交互设计要求

用户友好性：平台界面设计应符合用户习惯和直觉，操作简单明了，降低用户学习曲线。

操作效率：平台应支持快速导航、查找和操作，减少用户操作步骤和时间消耗。

5.4 安全性要求

数据保护：平台应采取必要的加密和访问控制措施，保护数据的机密性和完整性。

身份验证：平台应提供安全的用户身份验证机制，防止未经授权的访问。

安全审计：平台应记录和监控系统的安全事件和操作，及时响应和处理安全威胁。

5.5 扩展性和灵活性要求

模块化设计：平台应采用模块化架构，支持灵活的组件替换和扩展，满足不同业务场景的需求。

开放接口：平台应提供标准化的接口和API，便于与外部系统集成和扩展功能。

定制化支持：平台应支持用户定制化需求，允许根据具体场景进行配置和调整。

6 技术要求

6.1 场景构建

数字孪生低代码平台应具备场景构建功能，支持一个数字孪生可视化场景的完整搭建，包括：

6.1.1 场景编辑

场景编辑包含但不限于模型导入、模型对象、GIS 地图、倾斜摄影、画刷对象、光照编辑、特效编辑、模型编辑、材质编辑、关节编辑、动画编辑、粒子编辑、状态编辑、坐标编辑、场景发布、模型资产库、场景案例库。

- a) 模型导入：宜支持导入常用 3D 模型文件格式，准确读取模型层级结构、纹理材质等属性参数；
- b) 模型对象：可支持定义模型层级结构，编辑模型外观、内部结构、部件模型，支持结构剖分；
- c) GIS 地图：应支持卫片、高程、道路、水系、兴趣点等栅格、矢量数据接入，支持文件型接入或 OGC、OSGeo 规范服务接入；
- d) 倾斜摄影：应支持加载倾斜摄影数据，兼容 3DTiles、OSGB 等数据标准，支持对加载精细度进行参数调整；
- e) 画刷对象：应支持绘制大量模型的画刷对象，支持点画刷、线画刷不同模式，支持配置绘制对象集合，支持等距排列、等距平铺、单对象移动、多对象移动等灵活功能组合；
- f) 光照编辑：应具备光影效果配置能力，支持主光源，环境光、辅助光源属性配置，支持对光源颜色、角度、强度进行编辑；
- g) 材质编辑：宜支持 PBR（物理材质渲染）材质编辑、包括基本颜色、光泽度、金属度、粗糙度、半透明、自发光、环境反射、玻璃特性、水特性、图层特性、材质绘制深度变换、贴图坐标动画等多层纹理复合渲染；
- h) 关节编辑：宜支持为对象集合定义及绑定关节，支持数值型、布尔型、枚举型多种绑定数据类型，提供节点关节、材质关节、粒子关节多种类型对象关节控制；
- i) 动画编辑：宜支持定义模型复杂脚本动画，通过定义各关节数值变化范围、起始时间、持续时长等参数，生成关节时间线数值序列，形成模型动画；
- j) 时间编辑：宜支持场景时间设置，支持四季变化和一天 24 小时的日月运行和光照变化，用于模拟数字孪生场景的不同时间状态，支持与真实时间数据联动；
- k) 气象编辑：宜支持晴空、多云、小/中/大雨、小/中/大雪、沙尘暴等多种气象环境效果，用于模拟数字孪生场景不同的天气条件，支持与真实气象数据联动；
- l) 特效编辑：宜支持屏幕控件反射、全局投影、胶片颗粒、景深、模糊、锐化、色差、辉光、屏幕色彩空间变换、色彩调节等场景渲染滤镜，支持特效组合叠加；

- m) 粒子编辑：宜支持建立粒子系统，通过粒子发射频率、速度、角度、生命周期等参数，设定粒子效果，并支持粒子组合；
- n) 状态编辑：宜支持将场景的光照、特效、对象显隐、摄像机位置，以及不同模型的节点、关节、动画属性配置，组合保存为多个独立的场景状态，实现一键属性切换；
- o) 坐标编辑：应支持场景匹配真实世界地图位置、范围、朝向定位，为二次开发阶段精确叠加显示具备经纬度信息的业务数据提供支撑；
- p) 场景发布：应支持一键发布景服务，供用户直接浏览或二次开发调用。支持自定义发布场景渲染质量，用以匹配不同硬件能力的浏览设备；
- q) 模型资产库：应支持预制多种高质量模型资产，可直接拖拽到场景中使用，快速构建场景。模型资产库应包含通用建筑、交通设施、城市部件等类别，并通过资产管理功能对资产进行增删改查操作。
- r) 场景案例库：宜支持内置各种行业数字孪生常见业务场景，一键导入私有工作空间，并可查看和编辑场景内容，通过修改参数配置形成和输出新的应用场景。

6.1.2 场景服务

场景服务包含但不限于通道管理、场景管理、发布管理、场景浏览服务、场景开发调用服务、场景导入导出、场景预热驻留、项目管理、版本管理、多机集群、系统工具、日志管理。

- a) 通道管理（适用于云渲染架构）：可支持对云渲染服务器上的显卡硬件资源以渲染通道的形式，进行统一管理，支持多显卡识别、多显卡管理、支持设置通道分辨率，并设置每块显卡的指定规格通道数量，支持实时查看显卡和通道运行状态；
- b) 场景管理：应支持发布场景管理，发布场景关联项目的具体版本，支持对发布渲染场景的渲染分辨率、视频流编码质量、渲染增强系数、预热模式、许可令牌、浏览时间限制等属性进行配置；
- c) 发布管理（适用于云渲染架构）：可支持将数字孪生场景发布为公有云服务，允许用户通过 http 协议直接进行场景浏览或二次开发访问，支持对发布的多个场景进行管理，可查看场景发布状态、浏览次数，暂停、取消、更新发布场景，支持限制场景访问时间，允许单独用户独享一个场景或多个用户共享一个场景服务；
- d) 场景浏览服务（适用于云渲染架构）：可提供数字孪生场景浏览展示能力，通过浏览器中访问场景服务浏览地址加载场景，并进行场景浏览、场景漫游、状态切换、信息查看等操作；
- e) 场景开发调用服务：应支持统一 API 对场景服务的调用提供服务；
- f) 场景导入导出：宜支持场景服务数据包导入、导出，支持在公有云、私有云之间，不同用户不同私有云之间，进行场景服务的共享和迁移；
- g) 场景预热驻留（适用于云渲染架构）：可支持云渲染场景预加载，并持续预热驻留，在用户访问该场景时，可免除场景读取、启动时间，立刻显示渲染结果。通过后台管理，支持查看所有通道预热占用状态，支持设置场景预热/非预热通道数量；
- h) 项目管理：应支持项目管理，针对不同场景文件，建立不同的发布项目，每个项目支持建立多个版本，可查看版本历史记录、版本文件大小、创建时间、版本描述等信息；
- i) 版本管理：应支持每个项目建立多个发布版本，有效管理历史版本记录、版本发布状态、版本大小、存储位置等重要信息；
- j) 多机集群（适用于云渲染架构）：可支持构建云渲染场景服务器集群，支持集群管理，支持动态增加集群渲染节点，支持集群硬件资源自动统一调度分配；
- k) 系统工具：宜提供网络带宽测试工具，网络延迟测试工具，以及硬件性能检测工具，帮助用户便捷掌握服务器渲染硬件等级评分、网络太宽状态；

- 1) 日志管理：应支持记录场景服务访问日志、操作日志、错误日志，方便用户详细了解场景服务运行态势，排查定位场景服务故障，详细分析定位错误。

6.2 应用开发

数字孪生低代码平台应具备web应用的低代码和零代码开发功能，包括：

6.2.1 零代码应用编辑

零代码应用编辑包含但不限于工程管理、应用页面管理、页面编辑、场景服务集成、页面控件编辑、场景图层编辑、交互逻辑编辑、组件编辑、数据集成、参数编辑、数据接口集成、多设备页面编辑、应用预览发布。

- a) 工程管理：应支持工程新建、删除、编辑、移动、复制、改名、预览、传送等功能，支持建立文件夹对工程分类进行管理；
- b) 应用页面管理：应支持页面新建、删除、编辑、移动、复制、改名、预览等功能，并支持建立文件夹对页面分类进行管理；
- c) 页面编辑：应支持在应用页面中集成场景服务，添加和编辑数据图表、数据仪表盘、页面交互控件、场景数据图层、应用组件等多种对象，支持页面对象管理列表，支持页面对象任意叠加、调整位置布局、对象打组/解组操作，支持页面对象配置交互逻辑指令，支持设置页面分辨率、自适应浏览器方式，允许同一应用的不同页面由不同账户编辑，多人协同同步开发；
- d) 场景服务集成：应支持在页面中集成已发布的场景服务，支持页面对通过交互指令，深度控制场景服务的行为和包含的数字孪生对象，支持在页面中切换不同的场景服务、场景状态；
- e) 页面控件编辑：应内置数据图表、页面交互控件，全拖拉拽自有调整控件在页面中的大小和位置布局，可配置丰富的控件样式、数据源绑定、交互逻辑属性，支持静态数据、数据库数据、数据接口数据三种数据绑定方式；
- f) 场景图层编辑：应支持在集成的场景服务中，添加多种类型数据图层，支持地标图、柱图、气泡图、热力图、关系图、区域图、轨迹图等类型的数据图层，支持添加点、路线、区域、场景特效等不同类型的图层对象，可配置丰富图层样式、数据源绑定、交互逻辑属性；
- g) 交互逻辑编辑：应支持页面对象、图标对象、图层对象、控件对象相互间定义交互逻辑关系，根据交互触发条件执行相关交互指令。内置交互指令可深度控制数据图层、图层中对象、界面控件、数据图表的各种行为，可灵活配置对象间交互关系，实现图层联动、图表联动、表表联动、表图联动、关联选中、页面切换、数据筛选、模型剖分等复杂功能，支持扩展脚本编程；
- h) 组件编辑：应支持将基础图表和控件进行组合，提供独立的页面组件编辑器，可配置组件的分辨率、组件样式，组件包含的图表和控件，以及组件布局，支持页面调用时传递参数，支持页面通过交互指令对组件进行控制；
- i) 数据集成：宜支持集成常用关系数据库等数据源，支持配置数据表中每一个数据字段的数据类型；
- j) 参数编辑：应支持跨数据源数据联动筛选的参数编辑，可定义类别、空间、时间、ID 等不同类别参数，以及定义改参数具备的特定枚举值集合；
- k) 数据接口集成：应支持数据图表对象、数据图层对象、场景模型对象绑定数据接口，通过访问数据接口实时获取自身数据，支持自定义数据结构转换逻辑和接口二次编排，支持通过数据接口实时改变数据图表、数据图层、场景模型的显示数据和自身状态；
- l) 多设备页面编辑：宜支持多设备页面编辑，根据访问应用的设备不同，动态匹配显示不同设备页面，可为同一个应用页面，编辑不同设备下的设备页面，设计不同的控件组合、页面布局、

最佳分辨率和自适应方式；

- m) 多用户编辑：应支持不同应用、不同页面可有多个不同用户协同编辑，可采用加锁机制保证结果正确；
- n) 应用预览发布：应支持实时预览应用运行效果，支持一键发布，自动上传到应用服务器，并发布为云端零代码应用服务。

6.2.2 零代码应用服务

零代码应用服务包含但不限于应用打包、应用管理、版本管理、发布管理、应用访问服务、应用导入导出。

- a) 应用打包：自动将应用编辑器工程文件和相关资源进行打包，并将应用包发布托管至服务器，生成可通过浏览器的访问链接；
- b) 应用管理：应支持发布应用管理，提供一站式托管服务，支持发布应用建立文件夹分类管理，管理应用服务资源；
- c) 版本管理：应支持应用的多版本机制，各个版本间编辑、发布均不影响；
- d) 资产中心：应支持应用构建的成果复用，包括业务组件、页面、效果、数据接口、模型等；
- e) 发布管理：应支持发布的应用服务进行管理，可查看应用发布状态，暂停或取消应用发布，查阅应用访问日志，查看应用浏览次数；
- f) 应用访问服务：宜提供零代码应用服务能力，支持通过浏览器中访问零代码应用服务的浏览地址加载应用，并进行应用场景服务调用、应用图表/图层/控件数据解析，应用交互响应，应用数据访问等操作；
- g) 应用导入导出：宜支持应用服务数据包导入、导出，支持用户在公有云不同租户之间、公有云私有云之间、不同用户在不同私有云之间，进行应用服务的共享和迁移。

6.2.3 低代码统一开发 API

低代码统一开发API包括但不限于场景服务管理、场景状态控制、图层对象绘制、图层对象控制、模型对象绘制、模型对象控制、特效对象绘制、特效对象控制、GIS对象控制、倾斜摄影对象控制、模型对象控制、镜头视野控制、交互模式控制、时间控制、气象控制、多坐标系支持、大数据聚簇图层、

- a) 场景服务管理：应支持获取场景服务信息、初始化场景服务、预加载场景服务、销毁场景服务、切换场景服务等丰富开发接口；
- b) 场景状态控制：应支持场景服务状态查询、场景服务状态切换、场景服务状态恢复等开发接口，控制场景状态切换逻辑，接受和处理场景状态各种触发事件；
- c) 图层对象绘制：应支持地标点、热力图、柱图、气泡图、地标图、轨迹图、三维柱图、区域图、栅格图、关系图、以及数据聚合图等样式的数据图层绘制开发接口，通过传递结构简单的 Json 数据参数，在数字孪生场景内以特定数据图层形式呈现业务数据；
- d) 图层对象控制：应支持图层显隐、删除、改变样式、数据查询、数据更新、添加悬浮窗口等图层控制开发结构，支持对图层以及图层的子对象的聚焦、框选、点击、悬停等交互事件的截取，以及添加自定义处理逻辑；
- e) 模型对象绘制：应支持场景模型对象、模型地标图、模型轨迹图等多种模型对象绘制开发接口；
- f) 模型对象控制：应支持场景模型对象显隐、删除、改变姿态、改变样式、关节设置、动画控制、数据查询、数据更新、添加悬浮窗口等模型对象控制开发接口；
- g) 特效对象绘制：可支持雷达扫描、高亮地标、火焰、爆炸等场景特效对象的绘制开发接口；

- h) 特效对象控制：可支持特效对象显隐、删除、改变样式、数据查询、数据更新、添加悬浮窗口等特效对象控制开发接口；
- i) GIS 对象控制：应支持加载符合 OGC、OSGeo 规范的 GIS 数据，如 TMS、WMTS 瓦片地图数据、数字高程地形数据等；
- j) 倾斜摄影对象控制：应支持加载倾斜摄影数据开发接口，兼容 3DTiles、OSGB 等数据标准，直接连接数据服务地址，可支持加载精细度进行参数调整；
- k) 模型对象控制：宜支持对场景中的模型对象进行信息查询、拆解、展开、剖分、定位模型、定位部件等模型对象控制开发接口；
- l) 镜头视野控制：应支持镜头聚焦、多镜头视野控制开发接口；
- m) 交互模式控制：应支持第三人称沙盘模式、第一人称行走模式、第一人称飞行模式等场景交互模式设置开发接口；
- n) 时间控制：宜支持场景的时间设置，季节设置开发接口，支持四季变化和 24 小时的日月运行光影变化，模拟数字孪生场景不同时间状态，支持与真实世界时间数据绑定；
- o) 气象控制：宜支持场景气象控制开发接口，支持清空、多云、小/中/大雨、小/中/大雪、沙尘暴等多种气象环境效果，用于模拟数字孪生场景不同天气条件，支持与真实世界气象数据绑定；
- p) 多坐标支持：支持地理坐标系和投影坐标系，支持 WGS84、CGCS2000 坐标系，支持经纬度展点上图；
- q) 大数据聚簇图层：应支持读取大数据模型，制定多种类型可视化图层，以聚簇分级显示，根据观察距离逐级分解显示数据集合和细节数据。

6.2.4 低代码 API 调试

低代码API调试包括但不限于调试工程管理、API范例单体测试、API综合测试、运行事件监测、辅助工具、数据标绘导出、辅助代码生成。

- a) 调试工程管理：应支持工程新建、删除、编辑、移动、复制、改名等功能，支持建立文件夹对调试工程分类进行管理；
- b) API 范例单体测试：应支持 API 接口范例在场景中运行演示，针对每个单项 API 接口，均可浏览包含详尽注释的范例调用代码，支持在线修改调用代码，支持系统运行消息显示，实时追踪理解代码运行、消息处理过程；
- c) API 综合测试：宜支持将多个单体测试进行组合测试，实时修改代码、一键运行；
- d) 运行事件监测：应内置事件监听功能，直观测试运行代码，输出监测结果；
- e) 辅助工具：宜支持获取场景点位经纬度数据，获取任意视角参数信息，支持数据导出；
- f) 数据标绘导出：宜支持在场景中标绘点/线/面对象数据，自动生成 API 调用代码，支持自动模拟生成辅助数据，支持运行效果预览、代码导出。

6.3 性能要求

6.3.1 功能性

- a) 服务器端应支持数据缓存能力，实现数据快速调用，提升效率；
- b) 应支持后台数据服务订阅；
- c) 页面响应时间宜在 3 至 6 秒内；
- d) 应支持大规模（大于 500 万个三角面渲染）场景三维建模要求；
- e) 标准评测用模型（模型大小、面数、材质特效既定），在测评方的标准显卡配置条件下，渲染

帧率应不小于 24;

- f) 编辑端应支持场景数据缓存能力，实现页面刷新或网络中断下能恢复场景配置的能力。

6.3.2 安全性

- a) 系统应保证管理面域用户面隔离;
- b) 所有能对系统进行管理的人机接口及跨信任网络的机机接口应具备接入认证机制(标准协议定义中无认证机制的额除外)，相关接入认证机制应缺省启用;
- c) 系统对每一个需要授权访问的请求都应该核实用户会话表示是否合法、用户是否被授权执行操作;
- d) 对用户的最终认证处理过程应放到服务端进行;
- e) 用户登录认证通过后应更换会话标识;
- f) 禁止存在绕过正常认证机制直接进入系统隐蔽通道;
- g) 禁止使用私有密码算法实现加密，产品应使用通过认证的或业界开源公认的密码算法库;
- h) 对敏感数据的访问，根据产品业务场景和风险，应采取适当安全机制(如认证、授权或加密等)。

6.3.3 易用性

- a) 系统操作上下文应支持在线帮助文档;
- b) 所有用户操作界面应直观、易用、贴近实际业务;
- c) 应支持用户通过可用的接入设备来接入系统。

6.3.4 兼容性

- a) 宜支持跨平台操作系统，包括 windows10 或更高版本，国产操作系统或 Linux 操作系统等。
- b) 云渲染客户端应支持主流浏览器中的一种或多种，如 Chrome、Edge、Firefox、或国产浏览器。

6.3.5 可靠性

- a) 应能够自动备份关键数据，如数据库数据、应用程序运行过程中产生的所有数据、操作系统软件、中间件软件、数据库软件等;
- b) 宜支持 HA 部署架构，在单点故障下，主备服务器进行任何切换时，均能为用户提供在用户客户端和应用程序、数据库之间无缝的、不间断地、连续的连接;
- c) 应支持容灾恢复机制。

6.3.6 可伸缩性

- a) 应支持横向扩容提高性能，如增加服务器;
- b) 应支持纵向扩容提高系统性能，如增加单台服务器内存和处理能力。

6.3.7 可维护性

- a) 应支持服务启停控制，支持通过页面操作，启动、关闭或重启服务进程;
- b) 应支持日志记录，记录内容覆盖系统运行、操作、配置、维护各方面。系统配置变更、系统运行记录、告警上报等;
- c) 宜支持日志压缩存放，日志空间满后自动删除最早的压缩日志，且日志空间大小要求至少记录

90 天日志信息，保存天数可配置；

- d) 应日志文件应具备定期归档及清理机制；
- e) 不允许提供任何删除或修改日志的界面操作功能；
- f) 应支持告警机制，当出现异常情况（如网络故障、资源占用过高、接口调用失败、数据访问超时等），自动产生告警，支持在页面上查看详细信息；
- g) 告警级别应分类清晰，所有告警均能够在界面查询并上报；
- h) 应支持一键收集日志、告警、配置、版本等信息，系统具备查询版本号功能，以上操作可通过界面执行，所有日志收集操作不应影响业务；
- i) 应提供巡查程序或工具，支持巡查配置信息，自动生成巡检报告等；
- j) 应支持版本平滑升级，升级过程业务不中断或中断时间小于 2 小时；
- k) 应提供标准错误反馈码，如网络不通、鉴权错误、解码失败等。

参 考 文 献

- [1] GBT 43441.1-2023 信息技术 数字孪生 第 1 部分：通用要求
-