达州市城市体检信息平台项目

需求书

目 录

1. 项目概述

1.1 项目名称

达州市城市体检信息平台

1.2 建设背景

当前，我国城镇化率已超过 60%，在推动城市高质量发展与治理能力现代化背景下，城市建设的重点正转入对存量的提质增效阶段，城市发展已进入城市更新的重要时期。城市体检作为统筹城市规划建设管理和促进城市开发建设方式转型的重要抓手，能精准查找城市建设和发展中的短板与不足，及时采取有针对性的措施加以解决，建设没有“城市病”的城市。

2015 年，习近平总书记在中央城市工作会议上提出“城市工作要把创造优良的人居环境作为中心目标，努力把城市建设成为人与人，人与自然和谐共处的美丽家园”。

2017 年，习近平总书记考察北京城市规划建设管理工作时提出要“建立‘城市体检’评估机制”，探索建立“一年一体检、五年一评估”的城市体检工作制度。

2018 年，住房和城乡建设部会同北京市政府率先开展了城市体检工作。

2019 年，住房和城乡建设部在沈阳、南京、厦门等 11 个城市展开了城市体检试点工作。

2020 年，在新冠疫情背景下，按照党中央、国务院关于做好“六稳”工作、落实“六保”任务，把防风险、打基础、惠民生、利长远的改革有机统一起来，住房和城乡建设部在全国范围内选取 36 个样本城市以“防疫情、补短板、扩内需”为主题开展城市体检工作，全面查找城市发展和城市规划建设管理存在的问题和短板。

全国住房和城乡建设工作会议提出，2021 年要“加快建设城市体检评估信息平台，加强城市体检数据管理、综合评价和监测预警”。2021 年 8 月，四川省住房和城乡建设厅总结成都市和遂宁市作为国家城市体检样本城市的经验，印发了《四川省城市体检工作方案》《四川省城市体检指标体系》《关于组织申报全

省 2021 年城市体检试点工作的通知》等文件，围绕“生态宜居、健康舒适、安全韧性、交通便捷、风貌特色、整洁有序、多元包容、创新活力”八个方面，为城市“问诊把脉”。在我市以及自贡、南充、德阳、内江、乐山、绵阳、广元、泸州 9 个地级市开展试点。

当前信息技术创新日益加快，以云计算、物联网、大数据和人工智能、5G 为代表的新一代信息技术蓬勃发展，高速互联、智能感知、边缘计算等技术创新层出不穷。伴随城市体检工作，运用新-代信息化技术，整合数字化、网络化、智能化产生的城市海量数据，建立城市体检评估信息平台成为当务之急。以期加快城市建设管理的技术创新，提高城市发展建设质量和治理水平。

为深入贯彻习近平总书记关于城市体检工作重要指示精神，根据四川省住房和城乡建设厅《关于组织申报全省 2021 年城市体检试点工作的通知》（川建景园发〔2021〕216 号）布置安排，按照《达州市人民政府办公室关于印发<达州市 2021 年城市体检工作实施方案>的通知》（达市府办函〔2021〕147 号）要求，开展 2021 年达州市城市自体检工作，采用新一代信息技术，不断完善与优化反映城市运行状态的数字化底板，实现信息跨部门共享、指标管理与智能计算、体征感知与动态监测等功能，形成城市体检评估及建设数字化平台；查找达州市城市建设中存在的问题和短板，同时针对问题提出整改建议，编制城市自体检报告，为各部门制定整治行动方案提供参考，提升城市建设工作的整体性和系统性。

1.3 建设目标

1）有效治理“城市病”

随着城市化进程的不断加快，“城市病”愈演愈烈，城市精细化治理逐渐成为治理现代化重要且紧迫的落脚点。推动建设没有“城市病”的城市，促进城市人居环境高质量发展，深入查找“城市病”根源，提出“治疗方案”，开展城市体检工作，既是我国城市治理体系和治理能力建设的迫切需要，又是推动我市人居环境高质量发展的重要举措。建设城市体检可以及时找出城市发展中的弱项、短板，针对存在的“城市病”提出“诊疗”方案，对容易产生的“城市病”提出预防措施，为政府科学决策提供政策建议。

2）综合决策

对空间对象的分析与挖掘能力，在数字空间中构建城市的信息化模型，通过空间技术对信息的分析、挖掘，能够在数字空间中对现实世界的状态发展、变化趋势、管理响应等进行判断与预测，从而能够准确地把握城市发展，指导制定城市运维管理政策与手段，使未来城市更安全、更节能、更高效、更智慧，减少现实中的试错成本，实现将数字成果映射到现实世界，再将数字成果反哺现实世界的双向驱动，达到城市数字驱动发展的战略目标。

贯彻落实住建部，建立国家、省、市三级城市体检评估信息平台要求，建设市级城市体检评估信息平台，实现同城市体检级、国家级城市体检评估信息平台的对接，实现城市体检数据管理、综合评价和监测预警。围绕城市体检工作要求和对城市指标数据库、信息系统的建设要求，充分利用政府信息共享平台以及相关委办局平台数据资源等搭建城市体检级城市体检评估信息平台，主要实现以下建设目标：

1）建立城市体检指标数据库。依据相关标准及建库规范，分年份构建覆盖市区的空间数据和非空间数据的城市体检指标数据库。围绕各城市发展质量的整体评价和问题把握，形成“动态监测、定期评估、问题反馈、决策调整、持续改进”的建设管理闭环，在对各体检城市工作进行管理中，形成包括数据收集与分析、监测预警、定期评估、整改进度管理等功能，为城市体检数据采集与处理技术研究、利用遥感影像对样本城市有关指标进行核实和分析评价等工作提供支撑。

2）根据试点要求，采用地理信息系统（GIS）技术，对接 CIM 基础平台，建立城市体检评估信息平台。以城市体检指标数据库为基础，结合各部门共享的业务数据、城市管理数据，实现城市体检相关的综合查询、统计分析、评估预警等功能，为精准把脉“城市病”、治理“城市病”提供有效辅助决策，为城市规划建设管理和城市治理提供支撑服务。

3）通过建设集“数据采集、校核更新、模型分析、评估预警”于一体的城市体检评估信息平台，保持城市体检指标数据库的现势性，保障“城市体检——问题反馈——决策调整——持续改进”长效机制的有效运行，提高城市治理智能化、标准化、精细化水平，为数字城市、智慧社会建设提供基础支撑。

4）结合达州市实际，建立由基础指标与特色指标组成的城市体检指标体系，以官方统计数据为主要依据，对城市体检各项指标测算分析，对标各类指标标准

值，查找短板及突出问题，提出对策建议，形成年度城市自体检报告。结合第三方评估和社会满意度调查结果，因地制宜提出年度达州市城市治理措施建议及项目计划。

1.4 编制依据

- 《住房和城乡建设部关于开展 2021 年城市体检工作的通知》(建科函〔2021〕44 号)
- 《关于支持开展 2020 年城市体检工作函》（建科函〔2020〕92 号）
- 《2020 年城市体检工作方案》
- 《城市体检技术指南 2020 年试行版》
- 《四川城市体检城市体检工作方案》
- 《四川城市体检城市体检指标体系》
- 《城市体检级/市级城市体检评估信息平台建设指南（试行）》
- 《达州市人民政府办公室关于印发<达州市 2021 年城市体检工作实施方案>的通知》（达市府办函〔2021〕147 号）

2. 需求分析

2.1 功能需求分析

2.1.1 数据底座

2.1.1.1 城市体检数据库建设

根据前台信息平台的需求及问题诊断与识别的要求,设计城市体检数据库结构，建设可存储大数据及空间数据的数据中心。

2.1.1.2 数据加工处理

- 研究政府数据的特性，根据指标需求对数据进行清洗、筛选；
- 根据指标计算需求对数据进行标准化处理，形成中间结果数据集；
- 协助空间落图及坐标转换，完成数据的空间化；
- 根据评价指标的需求，利用人工智能、大数据、空间分析等新技术手段，构建计算分析模型，形成最终结果数据集。

2.1.1.3 多场景数据访问接口

- 实现政府自体检指标数据的上报接口；
- 综合报告指标数据下载相关接口；
- 城市城市大脑等其他数据输入接口。

2.1.1.4 商业数据

手机信令、百度签到、消费数据、企业数据等。

2.1.2 指标填报

2.1.2.1 指标信息录入

实现对城市体检指标及拆解后的子指标数据填报功能。

2.1.2.2 指标录入情况查看

指标信息录入随录随存,可随时在指标信息界面查看指标是否录入成功以及所录入的指标数值。

2.1.2.3 指标信息编辑维护及查看

对于已经录入的指标，可进行指标信息及拆分指标信息的查看，并对需要变

更的信息，进行编辑维护。

2.1.2.4 指标信息提交

对已经录入完的指标信息，并且确保信息准确性的填报用户，可以进行数据提交。提交完成后，用户可以查看提交的状态。

2.1.2.5 数据管理

根据委办局及年份查询所有相关的填报数据，并对填报的信息及情况进行查看下载，做到对指标信息搜集工作的监督及跟踪。

2.1.3 体征感知与监测

2.1.3.1 人口要素感知与监测

在多源大数据的基础上，实现人口数据的感知与监测，同时根据指标体系的标准范围， 实现城市体征的预警与趋势预测。

2.1.3.2 文化旅游要素感知与监测

在多源大数据的基础上，实现文化旅游数据的感知与监测，同时根据指标体系的标准范围，实现城市体征的预警与趋势预测。

2.1.3.3 交通要素感知与监测

在多源大数据的基础上，实现出行、车 速、机动车等相关数据的感知与监测，同时根据指标体系的标准范围，实现城市体征的预警与趋势预测。

2.1.4 智能评估与辅助决策

2.1.4.1 社区概览

以图表及地图形式展示全市建成区内社区情况。

2.1.4.2 社区画像

页面通过列表展示社区 5 大类指标项的评分、指标值、指标范围，社区的总评分，并以图表的形式展示社区总评分的中位数情况。

2.1.4.3 社区评估

以公服设施完善、商服设施遍历、交通便捷、安全健康、品质生活 5 大类指标项为依据，制定评分标准，对城区所有居住社区进行评估，以空间分布、图表、列表的形式展示社区的评估情况，对社区在五大类指标体系下的现状及问题

进行分析展示。

2.1.5 指标管理与智能计算

2.1.5.1 指标管理

提供指标录入、提交、查询、删除功能。

2.1.5.2 智能计算

提供指标计算能力，包括缓冲区的动态计算，矢量数据的坐标转换，空间数据的几何计算。

2.1.5.3 一键生成专题报表

报表内容包括城市体检 65 项指标及对应的指标值，根据勾选的指标项，生成相关专题报表。

2.1.6 社会满意度调查分析

采用词云、柱状图、饼状图、折线图的展示方式，对社会满意度调查情况及分析结果进行可视化、矢量化展示。

2.1.7 案例库（知识库）

收集、整理与城市高品质发展、城市建设管理相关的国内外优秀案例。针对政策机制、城市品质、宜居社区、老旧小区改造、智慧社区等多个方面，实现在平台上可检索、可学习的优秀案例知识库。

2.1.8 城市体检可视化展示

2.1.8.1 指标概览

对八方面指标进行整体概括展示，主要通过表格和图表的方式展示。

2.1.8.2 城市级指标对比

获取不同城市指标数据进行对比，对比结果进行可视化展示。

2.1.8.3 细化指标可视化展示-基础指标

“生态宜居、健康舒适、安全韧性、交通便捷、风貌特色、整洁有序、多元包容、创新活力”八方面指标分八个模块展示，每个模块主要通过地图和图表实现对指标的细化拆解展示。

2.1.8.4 细化指标可视化展示-特色指标

分析城市特色指标，对特色指标的进行可视化展示，主要以地图和图表的形式动态展示。

2.1.9 系统管理

2.1.9.1 用户管理

实现平台用户信息的创建、更新、删除、查询的功能。

2.1.9.2 角色权限管理

定义用户所属角色及该角色的基本信息、访问权限。功能包含角色信息的创建、更新、删除、查询及角色权限管理。

2.1.9.3 权重设置

实现指标体系权重的调整功能。权重调整后，后台重新计算社区相关的评价值并获得最新结果集。

2.2 非功能需求分析

系统应兼顾实用性和可发展性。实用性即满足现有系统的运行要求；可发展性即满足随后 3-5 年的系统可扩展性要求，并且硬件升级时，系统不需要做大的调整变动。

2.2.1 性能需求

本项目的性能需求如下：

（1）保证基础平台 7×24 小时的运行；提供高稳定性，保证在数据量或应用连接数高峰运行时的系统运行正常，保障持久化的系统运行。

（2）内部应用系统应具备 2000 人在线访问的规模，社会化服务（社区满意度）应具备 1 万人在线访问的规模，保证稳定、流畅运行。

- 简单事务处理（包含各类信息录入、修改、查询业务、主要页面平均响应时间等）≤3s；
- 信息录入、修改型简单事务：平均响应时间≤2s；
- 复杂事务处理≤60s；

- 各类固定统计报表形成时间：≤2 分钟。

（3）要有通畅的网络保证用户终端与服务器的互访。

（4）要有足够计算能力和足够的网络带宽以保证数据的高效并发访问。

可维护与可管理性

系统应该具备较强的可管理性和易操作性，便于系统管理人员尽快熟练掌握系统的操作和管理技能，保证系统安全可靠运行。

成熟性和可扩展性

系统所选用的技术支撑平台和软件产品，都必须是当今世界上具备主导和领先地位的成熟产品；同时具有较好的互操作性，便于系统集成。除此之外，在整体设计思想上，也具备较好的超前性，在整体技术上达到领先水平。

系统所选用的技术支撑平台和软件产品的选型及配置，要充分考虑到整体系统的可扩展性，适应业务不断发展而增加用户以及业务流程数量的要求。

界面需求

系统运行需要考虑其交互体验，设计出美观、简洁、大方的系统用户界面，做到易用、美观、方便、快捷；使用菜单、工具按钮、快捷键合理布局；支持操作标准界面。

2.2.2 基础设施环境需求分析

本项目的应用系统优先考虑部署到电子政务云平台，需根据项目建设规模设计云平台计算资源规模，保障项目稳定、安全、有效运行。

2.2.3 基础软件平台需求

本项目操作系统、数据库管理系统、应用服务器软件、其他软件基础中间件和应用支撑平台软件拟采用国产或开源软件，介绍详见后续章节。

系统网络需求

本项目依托城市体检电子政务外网部署，通过城市体检政务外网的统一出口与互联网连接。面向各级业务人员、监管部门、领导部门和社会公众，需保证基础网络环境的安全。

系统涉及大量页面处理、数据交互，以每个市 200 个操作员，区县(含街道社区)15 个系统操作员计。单笔业务(含影像)约 750KB，并发量按 10%计算，带宽需求约 500MB。

在投入正式运行的初始阶段，可按实施上线的规模逐步按需扩展和升级。

2.2.4 计算存储资源需求

按照社会化服务在线人数 1 万人，内部应用系统同时在线人数约 2000 人计，每年的数据增长量预计在 4-5TB，包含结构化和非结构化数据。

根据以上访问量、业务和数据增长量，估算所需要的计算资源如下：

环境	用途	硬件资源	规格	数量
生产环境	数据库服务器	RDS for MySQL	2 核 CPU，16G 内存，50G 数据盘	15
	数据库服务器	RDS for MySQL	4 核 CPU，16G 内存，500G 数据盘	2
	数据库服务器	RDS for MySQL	4 核 CPU，16G 内存，100G 数据盘	1
	数据库服务器	RDS for MySQL	4 核 CPU，16G 内存，1000G 数据盘	1
	数据库服务器	RDS for MySQL	8 核 CPU，32G 内存，2000G 数据盘	1
	应用服务器	云主机	8 核 CPU，32G 内存，160G 系统盘	7
	web 服务器	云主机	2 核 CPU，8G 内存，80G 系统盘	1
	消息服务器	云主机	2 核 CPU，4G 内存，40G 系统盘	3
	mongo 服务器	云主机	2 核 CPU，4G 内存，80G 系统盘，1T 数据盘	3
	日志服务器	云主机	4 核 CPU，8G 内存，80G 系统盘，500G 数据盘	1
	日志服务器	云主机	2 核 CPU，8G 内存，40G 系统盘	1
	数据库 Redis	redis 云服务	16G 集群版	1
	对象存储 OSS	OSS 云服务		1

由于云平台具备“弹性计算，按需扩展”的特点，项目可根据实际实施进度和访问量逐步按需增加计算存储资源（后续以测试结果和实际需要为准）。

2.3 城市体检报告编制需求

为全面推动四川省开展城市体检工作，2021 年 9 月四川省住房和城乡建设厅印发了《关于组织申报全省 2021 年城市体检试点工作的通知》，通过申报遴选，率先在自贡、南充、德阳、内江、达州、乐山、绵阳、广元、泸州等 9 个地级市开展省级试点。其中，城市自体检由试点城市组织开展，围绕生态宜居、健康舒适、安全韧性、交通便捷、风貌特色、整洁有序、多元包容、创新活力等 8 方面，以官方统计数据为主要依据，对城市体检各项指标测算分析，查找城市人居环境质量存在的问题，提出对策建议，形成年度城市自体检报告。

针对指标数据进行分析，统筹考虑法律法规、标准规范、政策文件和国家、省、市建设发展与管理要求，构建“汇交—采集—诊断”的问题诊断机制，按照定性与定量、主观与客观相结合的原则分析论证，综合评价城市人居环境质量，及时发现城市发展特色优势与“短板”“弱项”，针对城市建设问题提出整改建议，形成年度城市自体检报告。

2.4 信息安全需求

鉴于本平台中涉及到城市运维信息数据。一旦业务信息遭到非法入侵、修改、增加、删除等不明侵害（形式可以包括丢失、破坏、损坏等），会对公民、法人、事业单位和社会团体的合法权益造成影响和损害。

根据《关于落实国家信息安全规定推进信息系统等级保护工作的通知》，结合市金融风险监测预警和监管信息平台的实际情况，初步确定本项目需达到信息安全保护等级 3 级标准。

根据确定的定级对象及定级建议，参考《信息系统安全等级保护基本要求》中对各子系统提出的安全建设要求，分析出达州市城市体检信息平台的技术防护需求包括物理安全、网络安全、主机安全、应用安全以及数据与备份安全方面，安全建设需求包括：

2.4.1 物理安全

主要针对的是重要信息系统运行的机房建设，提供良好的运行环境，用以支撑重要应用系统的运行，防止电磁信息的泄露、防止设备被盗被破坏，主要是设置冗余或并行的电力电缆线路。

2.4.2 网络安全

主要关注的方面包括：网络结构、网络边界以及网络设备自身安全等，具体的控制点包括：结构安全、通信传输、边界防护、入侵防范、访问控制、安全审计等六个控制点，通过网络安全的防护，为用户信息系统的安全运行提供一个安全的环境。

2.4.3 主机系统安全

主机系统是构成信息系统的主要部分，其上承载着各种应用。因此，主机系统安全是保护信息系统安全的中坚力量。具体的控制点包括：身份鉴别、访问控制、安全审计、入侵防范、恶意代码防范、资源控制等六个控制点。

通过网络、主机系统的安全防护，最终应用安全成为信息系统整体防御的最后一道防线。具体的控制点包括：身份鉴别、访问控制、安全审计、软件容错、资源控制、数据完整性、备份与恢复、剩余信息保护、个人信息保护等九个控制点。

3. 总体建设方案

3.1 建设内容

建设内容包括“一套规范、一个数据中心、一个平台”：

“一套规范”即数据标准规范，根据城市管理要求，基于国家、城市体检、市相关技术标准规范编制一套适应城市管理建设要求和实际情况的标准规范体系；

“一个数据中心”即城市体检数据中心，结合各部门共享数据、城市管理数据形、城市体检数据，形成城市体检数据中心。

“一个平台”即城市体检评估信息平台。包括指标管理、指标数据采集、体检评估、信息共享等功能，为城市体检工作开展提供平台支撑。

3.2 设计原则

（1）统筹和整合原则

城市体检评估信息平台全面整合现有各业务部门平台建设资源，在现有全市信息化资源成果的基础上进行整合扩充，提高信息资源共享水平，提高互联互通程度，提高统筹协调效率，最大程度地避免浪费与重复建设。

（2）统一和分布原则

城市体检评估信息平台的建设必须在统一组织领导下，统一确立各阶段重点，明确分工，突出重点，先急后缓，先易后难，分步实施，注重实效，避免系统陷入相互不兼容或者前期投资浪费的情况。

（3）先进和实用原则

城市体检评估信息平台的建设要尽可能采用先进的技术、方法、软件、硬件和网络平台，确保系统的先进性，同时兼顾成熟性，使系统成熟而且可靠。系统在满足全局性与整体性要求的同时，能够适应未来技术发展和需求的变化，使系统能够可持续发展。应从用户需求出发，在详细用户需求分析的基础上确保数据的完善性，以保证数据信息和功能模块能满足用户的实际需求。

（4）共享和兼容原则

在设计和建设过程中，必须充分考虑系统集成和数据共享。数据共享包括本部门的数据共享给外单位和本部门业务需要外单位共享的数据两种情况。系统在设计的时候就应该选定适当的开发平台和技术框架，尽量做到技术上保持一致性。同时要考虑系统应该具有良好的兼容性，可以兼容现有的业务系统。

（5）开放和规范原则

城市体检评估信息平台必须是开放性的才能够兼容和不断发展，才能保证前期投资持续发展。平台在运行环境的软、硬件平台选择上要符合云计算、大数据、GIS、IT 等行业标准，从设计到验收均执行相应的国际、行标及省市的有关标准和规定。

（6）扩展与维护原则

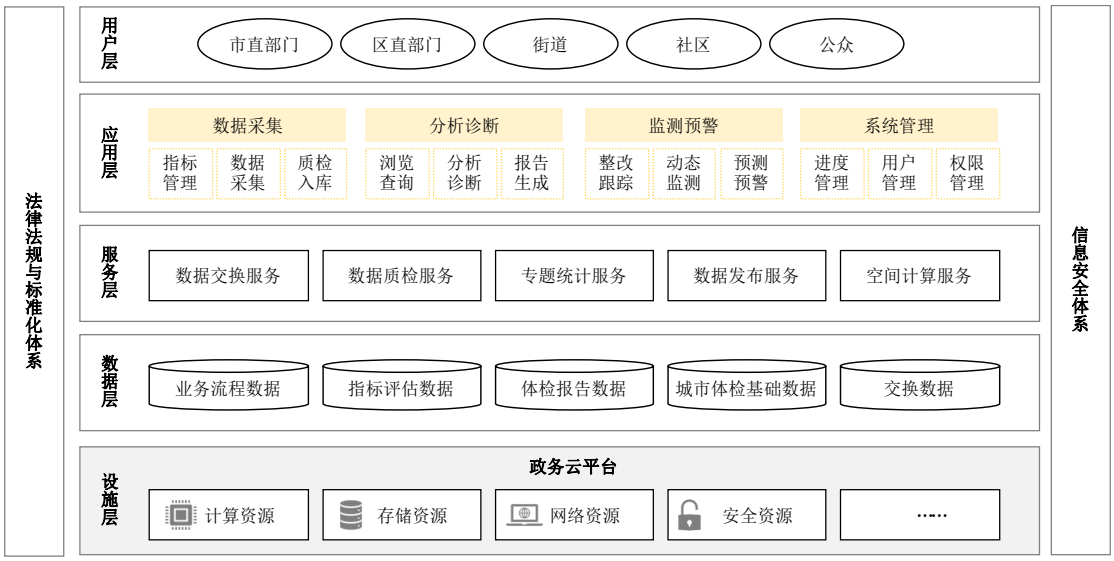
城市体检评估信息平台应考虑到未来的发展，机构、业务的变化，系统需采用灵活的设计方法，在相关数据、文档和资料格式变化时，能够快速进行转换、导入、导出和扩充等，既保证了动态条件下业务流程的正确性，又保留了足够的业务可扩充性。

（7）安全和保密原则

城市体检评估信息平台应建立全面严谨的数据安全保护机制，应从硬件、软件、制度各个层面采用必要的安全保护技术并建立相应的管理办法制度，从而保证数据的真实性、完整性、安全性和保密性。

3.3 总体架构

系统总体架构主要包括设施层、数据层、服务层、应用层、政策法规、安全标准等。



系统总体架构

设施层：基于政务云平台，提供支撑平台运行的各类软硬件基础设施资源，形成可动态扩展的高性能计算环境、大容量存储环境，满足海量数据存储、多类型用户并发应用和信息公开共享查询，以及各级业务系统接入信息平台的需要。

数据层：平台运行的数据基础，为平台的各类应用服务提供数据资源，是平台运行的动力和源泉。 数据层由普查调查数据、体检指标数据、各部门专题业务数据、基础地信息数据服务、 网络感知数据等组成，负责数据的统一组织、存储和管理，对应用层的体检成果信息交换、共享和查询、信息挖掘分析提供数据支撑。

服务层：提供城市体检成果信息的查询、分析、交换、共享服务。支撑平台各类业务应用的基础服务，涉及省-市平台之间的对接，以及第三方城市体检平台、城市自建体检平台的对接，是城市体检信息平台运行的基本结构。

应用层：高度吻合用户的业务需求，包括工作进度管理、数据填报、数据分析、体检报告管理、预警跟踪和数据管理等定制化功能，为用户提供良好的业务应用。

用户层：面向各类用户，包括体检成果报送机构、体检成果管理机构和体检成果共享部门，分为市级用户、区级用户和镇街用户。根据不同用户业务需求，设置对应的应用层系统访问权限，为不同用户提供应用服务。

3.4 标准化体系设计

标准化工作是组织、协调项目顺利发展的重要手段，也是系统的重要组成部分。通过制定和贯彻执行各类技术标准，就能从技术上、组织管理上把各方面有机的联系起来，形成一个统一的整体，保证项目有条不紊的进行。国内外信息化的实践证明，信息化建设必须有标准化的支持，尤其要发挥标准化的导向作用，以确保其技术上的协调一致和整体效能的实现。因此，标准体系建设在系统实施过程中具有非常重要的意义，是系统设计和工程建设的重要基石。为保证标准体系建设的顺利进行，制定以下总体目标：

- 系统标准化建设将与国家信息化建设标准保持一致性，建立并不断完善系统标准体系；
- 制定系统关键基础标准，为系统互联互通、信息共享、信息安全打好基础；
- 建立系统标准贯彻实施机制为标准的实施提供有效服务。
- 结合实际情况，统一制定并发布城市体检技术标准、数据标准等。

3.5 数据采集与建库

能够综合运用遥感信息提取、全自动化制图等技术，结合城市规划与设计、生态、计算地学的多学科研究方法，实现城市有关地理信息和建设情况的数据获取，包括但不限于地表覆盖分类、城市建成区边界提取、城市建筑物高度估算、城市交通路网信息提取等。

3.5.1 体检指标数据库

体检指标数据库围绕“生态宜居、健康舒适、安全韧性、交通便捷、风貌特色、整洁有序、多元包容、创新活力”八个方面，建立全方位、多维度的城市体检指标、指标拆解、指标标准等相关的数据库。

3.5.2 行为感知数据库

行为感知数据库囊括城市体检汇聚的城市感知数据，是实时、动态数据的聚集中心，数据包括生态环境、土地、人口、建筑、设施、经济、产业等来自政府

的数据，城市大脑的 API 接口是主要的获取数据的途径；来自物联网、互联网的手机信令、LBS（签到服务）、导航、工商企业等动态感知的大数据；问卷调查、移动端采集的社会参与数据。

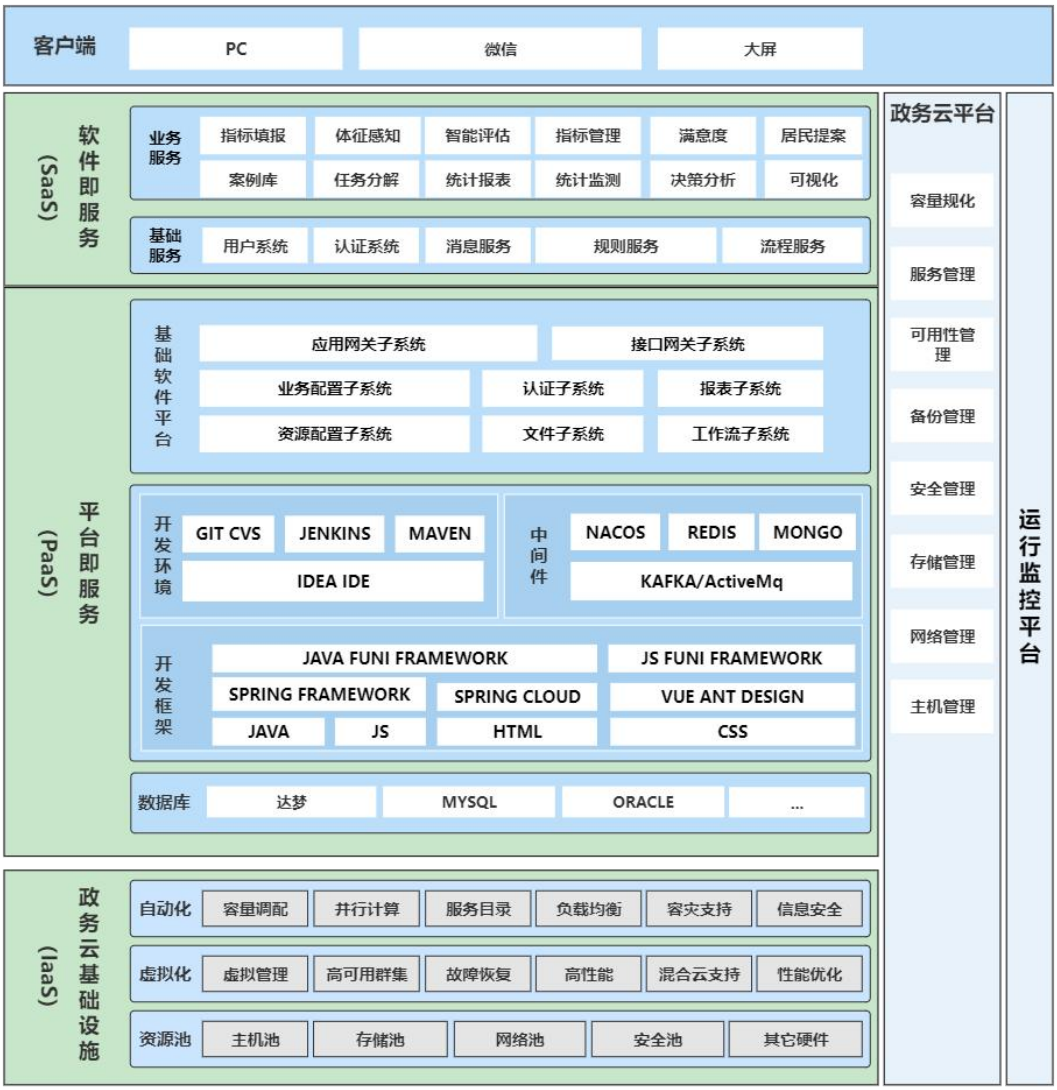
3.5.3 空间要素数据库

空间是城市中人们居住、生活、工作、游憩、交通的载体，城市体检指标的计算需要空间矢量数据的支持，空间要素数据库存储城市体检评估信息平台需要展示、计算与分析的所有空间数据，空间数据建议采用“2000 国家大地坐标系（CGCS2000）。空间要素数据库采用分层的方法进行组织管理，包括图层名称、几何特征及属性表。基于 2021 年城市体检指标体系。

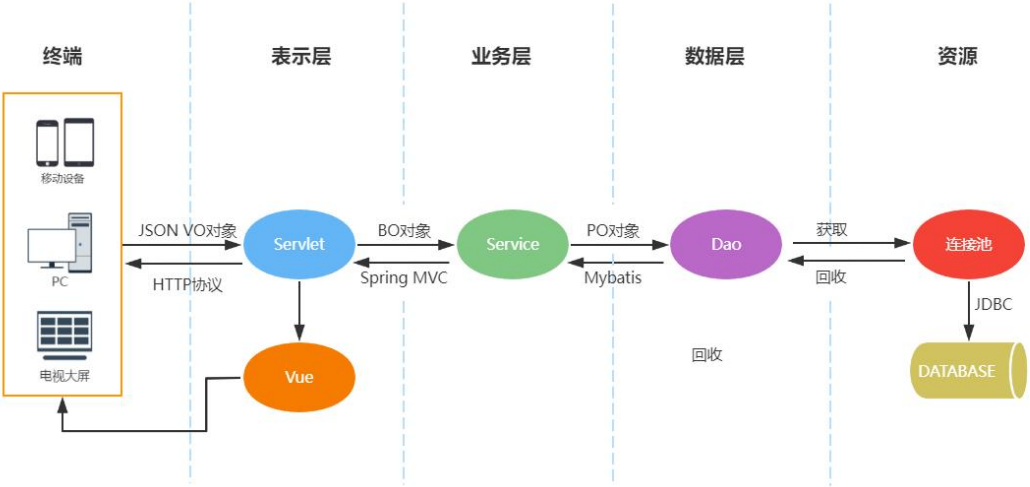
3.6 技术体系

3.6.1 总体设计思路

达州市城市体检信息平台的总体架构，基于 B/S 架构实现 SASS 化设计，要从传统烟囱架构模式走向面向服务（SOA）大平台的架构，并且除了满足正常业务办公外，重点加强对各类数据的管理、分析和应用，既满足现势需求，又兼顾未来发展需要。在设计上完全面向微服务的分布式架构，并引入 DDD 进行领域服务拆分设计，保证应用服务设计的合理性，具体如下：



1）采用 B/S 多层体系结构：采用 B/S 多层体系结构实现。包括表示层、业务层、数据层；



如上图所示三层架构自左而右将系统分为表示层、业务逻辑层、数据访问层。

表示层由处理用户交互的客户端组件及其容器所组成，采用 J2EE 规范中的 Servlet 协议实现，前端使用 Vue 组件，整个协议框架基于 SpringMVC 封装实现。

业务层由解决业务问题的组件组成，通过事务包裹保证业务读写的准确性、一致性。

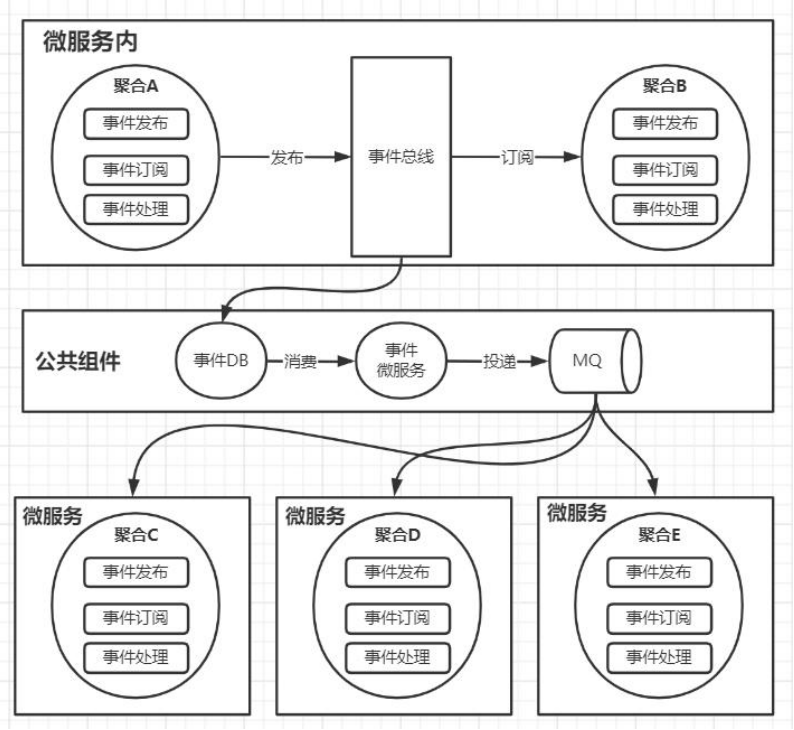
数据层由一个或多个数据库组成，并可包含存储过程，采用 J2EE 规范中的 JDBC 协议实现，使用 Mybatis 框架封装实现 DAO，底层资源管控采用 Druid 实现数据库连接池，通过连接重用提升程序对数据库的读写能力。

这种三层架构，在处理客户端的请求时，使客户端不用进行复杂的数据库处理；透明地为客户端执行许多工作，如查询数据库、执行业务规则和连接现有的应用程序；并且能够帮助开发人员创建适用于企业的大型分布式应用程序。

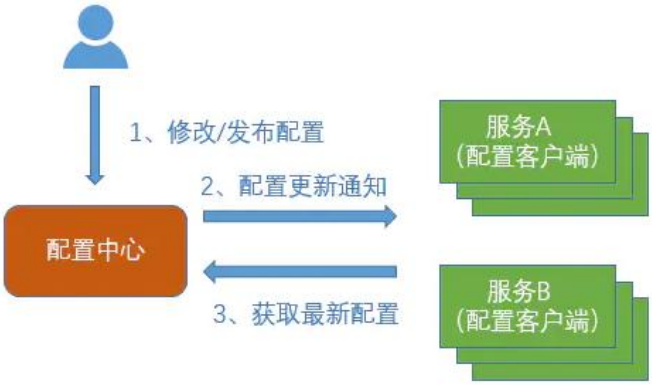
达州市城市体检信息平台的三层体系结构设计，使应用系统具备了良好的灵活性和可拓展性，在应对于今后在房产管理工作不断深入和细化的要求方面，将无需重新调整体系结构，完全可以随着软硬件网络环境的扩充而支持更多的应用。

2) 基于 J2EE 体系：为了保证系统的兼容性，高可用性、高可靠性和可扩展性，系统必须沿用前期项目的技术路线，要选择支持强大的企业级计算的成熟的 J2EE 企业标准；

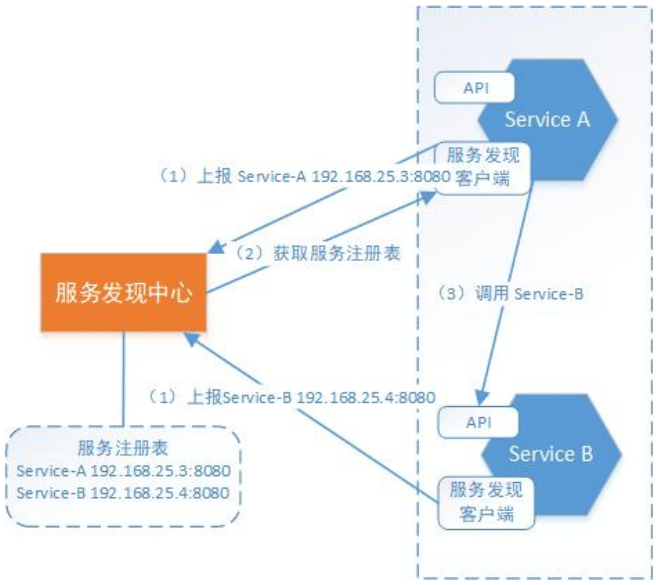
3) 完全面向微服务的分布式架构，并引入 DDD 进行领域服务拆分设计，保证应用服务设计的合理性；通过领域服务拆分，各个应用服务实现了松耦合，然后通过配置中心、注册发现、服务网关、事件总线等组件完成一体化建设，以适应不断变化的业务管理要求，提升平台的管理能力。



4）配置中心：传统模式，一个应用一个配置文件都打包放在项目中，如需修改配置须先改代码再打包，最后重启，而微服务架构下，应用的所有配置都存储在配置中心，配置中心直接修改，而后重启应用即可，部分标识符配置还支持修改后立即生效无需重启，提升平台运维效率；

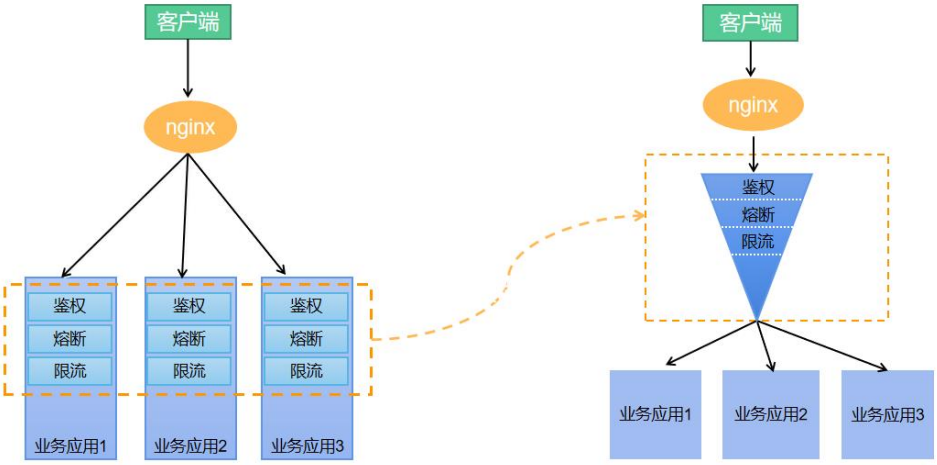


5） 注册发现：传统 RPC 调用，是必须在客户端配置相应的远程 IP 和端口，而微服务架构引入注册发现组件，服务 B 在启动时候会自动把 IP 和端口信息注册到注册发现中心，服务 A 请求前从注册发现中心得到 B 的 IP 和端口，最后直接请求 B；



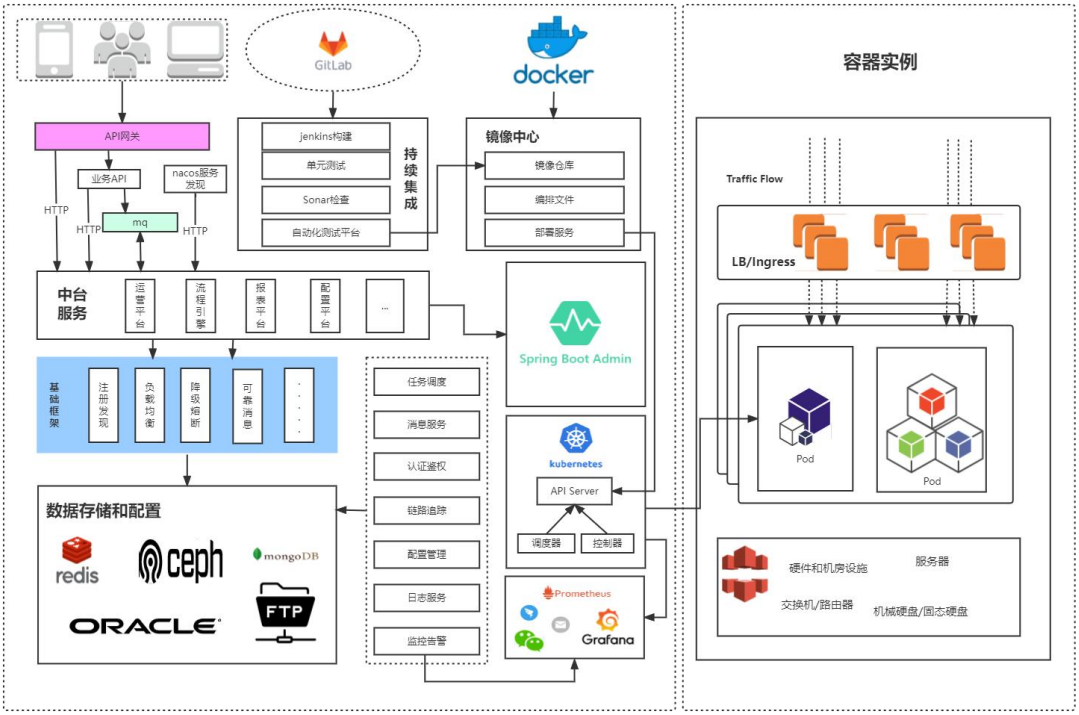
6）RPC 框架：平台通过整合 Spring Cloud Ribbon 与 Spring Http Invoker，完成 RPC 组件设计，结合 OOP 原则客户端直接使用本地接口方法便完成微服务间业务协同。

7）服务网关：网关是一种特殊的应用服务，它将业务应用中共通的能力进行剥离，拆分为独立应用，实现权限校验、熔断、流控、负载均衡等能力。

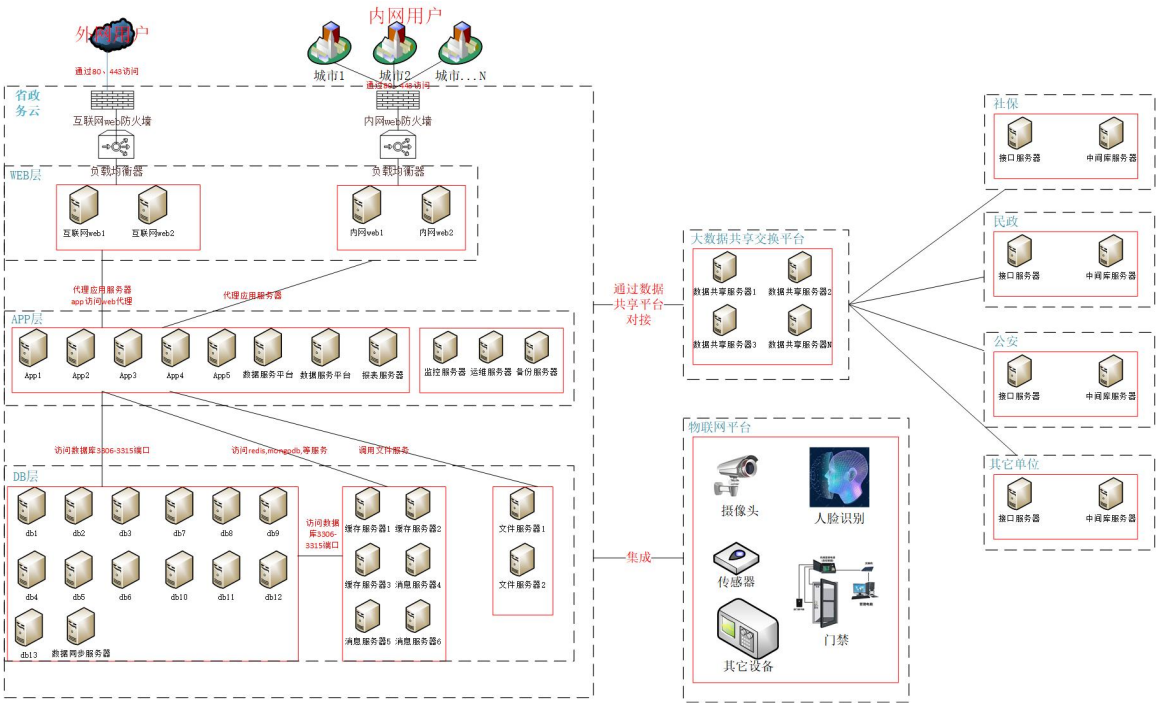


8）去中间件部署：传统应用依赖 WEB 容器对外提供服务，WEB 容器在接收到请求后完成预处理和封装在传递给容器内的项目，而微服务内嵌 HTTP 组件，可直接监听端口不需要安装 WEB 容器；

9）支持容器云部署：为提升服务器资源的利用率，平台不仅支持基于操作系统的原生部署，亦支持容器云部署，即所有微服务均通过容器（docker）进行部署，然后通过 Kubernetes 进行容器编排。该部署技术不仅可实现金丝雀部署、蓝绿部署，还可以根据流量大小进行在线扩缩容。由于所有计算资源以及存储资源都是通过 Kubernetes 统一调配，因此一定比例的硬件故障并不会影响整个应用的服务水平，从而实现 ServerLess。



3.6.2 系统部署架构



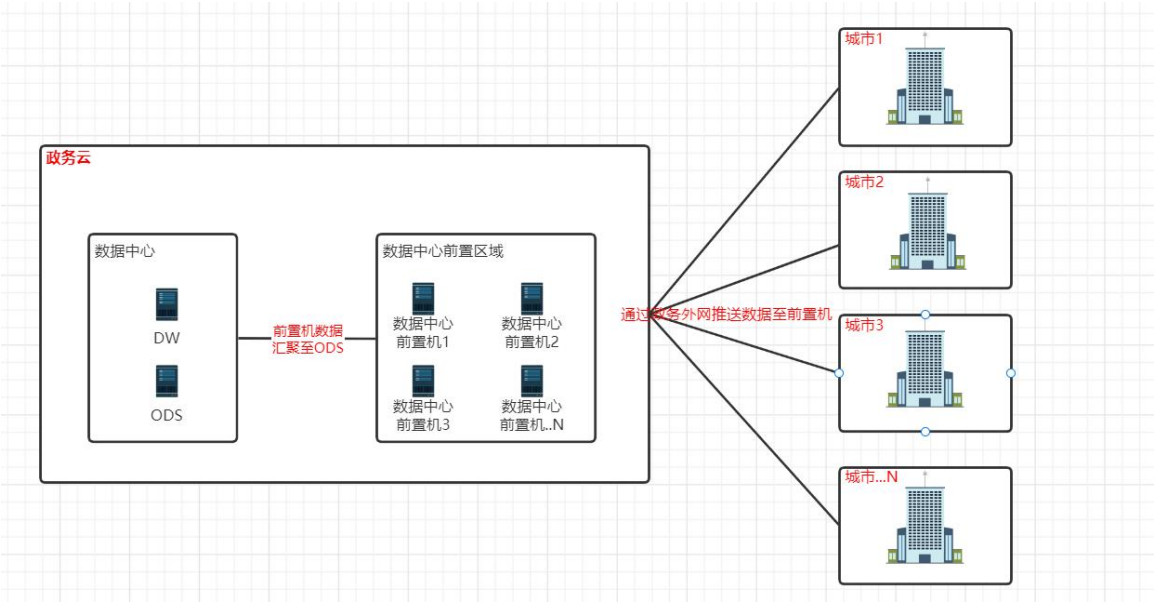
内网用于通过内网地址访问系统，外网用户通过域名访问系统。均需经过 WEB 防火墙，负载均衡器代理 WEB 服务器。

内网 WEB 服务器、外网 WEB 服务器放置于 WEB 层，WEB 服务器代理应用服务器。

应用服务器、监控服务器、运维服务器、备份服务器、数据服务平台等放置于 APP 层，应用服务器需要访问数据库、中间件、文件服务器等。监控服务器需要访问所有服务器。

业务数据库、前置机中间库、DW 库、ODS 库、中间件服务器、文件服务器等放置于 DB 层。

与社保、民政、公安等通过大数据共享交换平台进行对接。



数据网络拓扑图

3.6.3 国产化设计

3.6.3.1 服务器端 CPU 架构

平台架构设计选用 JAVA 生态，兼容 X86 以及 ARM 架构 CPU 芯片，实现同统信软件国产化适配认证，兼容鲲鹏 916、鲲鹏 920、飞腾 FT2000+/64、飞腾腾云 S2500 等芯片，并与龙芯中科科技股份有限公司完成 3B3000、3B4000 处理器适配认证，支持兆芯、海光、申威等。

3.6.3.2 服务器端操作系统

平台通过 JVM 实现跨平台兼容，原则上支持市面上所有 Linux 内核以及 Windows 内核服务器操作系统，包括中标麒麟、银河麒麟、统信 UOS、中科方德、欧拉操作系统等，并已完成统信 UOS 认证。

3.6.3.3 服务器端数据库系统

平台采用 J2EE 标准下的 JDBC 规范作为数据库连接标准，应用程序采用标准 SQL 语言开发，对于非标的查询设计则采用自定义函数以及 SQL 方言等技术完成，目前已完成国产数据库达梦、神舟通用、人大金仓、高斯、南大通用，开源数据

库 MYSQL，商业数据库 ORACLE 等关系型数据库适配。

1、 JDBC 接口：平台研发语言为 JAVA，其作为成熟的应用系统开发语言已得到几乎所有的数据库厂商支持，通过 JAVA 提供的标准 JDBC 接口，我们几乎兼容市面上所有的关系性数据库；

2、标准 SQL： 平台内部单表查询均通过对象查询实现，完成百分百标准 SQL 语句封装；

3、自定义函数： 平台自建了一套通用的函数表，通过标准自定义函数完成不同数据库厂商的内置函数适配；

4、SQL 方言路由：在完成以上标准化设计后，几乎可以涵盖业务系统中百分之九十的 SQL 使用场景，剩下百分之十的业务 SQL 因其复杂性可能需要保留数据库厂商的自定义语法，平台通过封装 SQL 方言路由的方式实现同一条业务 SQL 允许不同的 DB 查询语言存在，请求发生时 ORM 组件通过当前数据库连接自动选择相应厂商的 SQL 方言，完成业务执行；

3.6.3.4 服务器端应用中间件

平台采用微服务架构设计，默认情况下采用嵌入式中间件实现 J2EE 规范，包括 Servlet、JDBC、JTA、协议等，同时支持采用 WAR 方式部署，兼容东方通、中创、金蝶天燕、宝兰德等 J2EE 中间件。

3.6.3.5 客户端浏览器

平台采用 B/S 架构，后台提供标准的 REST 风格接口，前端利用客户端浏览器完成界面渲染、以及功能交互支持，兼容火狐浏览器、360 可信浏览器、IE 浏览器、CHROME 浏览器等。

3.6.3.6 客户端操作系统

平台在与外部设备进行通信交互时存在客户端操作系统适配要求，如调用高拍仪等操作均是通过本地安装的小助手完成，目前小助手已完成部分国产化操作系统的适配，支持宝德电脑等。

3.6.4 关键技术及选型

3.6.4.1 前端技术

3.6.4.1.1 开发语言

（1）JavaScript

JavaScript 一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为 JavaScript 引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在 HTML（标准通用标记语言下的一个应用）网页上使用，用来给 HTML 网页增加动态功能。

（2）HTML5

HTML5 万维网的核心语言、标准通用标记语言下的一个应用超文本标记语言（HTML）的第五次重大修改。HTML5 的设计目的是为了在移动设备上支持多媒体。新的语法特征被引进以支持这一点，如 video、audio 和 canvas 标记。HTML5 还引进了新的功能，可以真正改变用户与文档的交互方式。

（3）Less

Less 是一门 CSS 预处理语言，它扩充了 CSS 语言，增加了诸如变量、混合（mixin）、函数等功能，让 CSS 更易维护、方便制作主题、扩充。Less 可以运行在 Node 或浏览器端。LESS 以 CSS 语法为基础，又借用了很多熟知编程式语言的特性，这对于开发人员来讲学习成本几乎可以忽略，它在保留 CSS 语法的基础上扩展了更多实用的功能，LESS 为开发者提供了一种新的编写样式表的方法，可以根据我们的项目特性选择使用 LESS 的部分特性，开发者只需用很少的成本就可以换来很大的回报，一句话，Less is more，借助 LESS 可以更便捷的进行 Web 开发。

3.6.4.1.2 脚本框架

（1）Vue

Vue 是一套用于构建用户界面的渐进式框架。与其它大型框架不同的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支

持类库结合使用时，Vue 也完全能够为复杂的单页应用提供驱动。Vue 具有易用、灵活、性能高效等特点。本次项目尤其在涉及到表单等方面采用此框架。

（2）Jquery

jQuery 是一个快速、简洁的 JavaScript 框架，是继 Prototype 之后又一个优秀的 JavaScript 代码库（或 JavaScript 框架）。jQuery 设计的宗旨是“write Less，Do More”，即倡导写更少的代码，做更多的事情。它封装 JavaScript 常用的功能代码，提供一种简便的 JavaScript 设计模式，优化 HTML 文档操作、事件处理、动画设计和 Ajax 交互。

jQuery 的核心特性可以总结为：具有独特的链式语法和短小清晰的多功能接口；具有高效灵活的 CSS 选择器，并且可对 CSS 选择器进行扩展；拥有便捷的插件扩展机制和丰富的插件。jQuery 兼容各种主流浏览器。

3.6.4.1.3 开发工具

WebStorm 是 jetbrains 公司旗下一款 JavaScript 开发工具。目前已经被广大中国 JS 开发者誉为“Web 前端开发神器”、“最强大的 HTML5 编辑器”、“最智能的 JavaScript IDE”等。与 IntelliJ IDEA 同源，继承了 IntelliJ IDEA 强大的 JS 部分的功能。

3.6.4.1.4 软件构建（Build）工具

（1）Gulp

Gulp 是基于 node.js 的一个前端自动化构建工具，开发者可以使用它构建自动化工作流程（前端集成开发环境）。使用 gulp 可以简化工作量，让开发把重点放在功能的开发上，从而提高开发效率和工作质量。

（2）webpack

webpack 是一个模块打包器。webpack 的主要目标是将 JavaScript 文件打包在一起，打包后的文件用于在浏览器中使用，但它也能够胜任转换（transform）、打包（bundle）或打包（package）任何资源（resource or asset）。

3.6.4.2 后端技术

3.6.4.2.1 开发语言 Java

平台采用 JAVA 语言编写，基于 JDK1.8 版本，Java 是一门面向对象编程语言，不仅吸收了 C++语言的各种优点，还摒弃了 C++里难以理解的多继承、指针等概念，因此 Java 语言具有功能强大和简单易用两个特征。

Java 语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。

3.6.4.2.1 Java 具有简单性、面向对象、分布式、健壮性、安全性、平台独立与可移植性、多线程、动态性等特点。可以编写 Web 应用程序、分布式系统等。REST 服务框架 SpringMVC

Spring MVC 属于 SpringFrameWork 的后续产品，已经融合在 Spring Web Flow 里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用 Spring 进行 WEB 开发时，可以选择使用 Spring 的 SpringMVC 框架或集成其他 MVC 开发框架。

3.6.4.2.2 IOC 与 AOP 框架 Spring

Spring 是一个开源框架，它由 Rod Johnson 创建。它是为了解决企业应用开发的复杂性而创建的。Spring 使用基本的 JavaBean 来完成以前只可能由 EJB 完成的事情。然而，Spring 的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何 Java 应用都可以从 Spring 中受益。

Spring 是一个轻量级的控制反转(IoC)和面向切面(AOP)的容器框架。

轻量——从大小与开销两方面而言 Spring 都是轻量的。完整的 Spring 框架可以在一个大小只有 1MB 多的 JAR 文件里发布。并且 Spring 所需的处理开销也是微不足道的。此外，Spring 是非侵入式的，如典型地 Spring 应用中的对象不依赖于 Spring 的特定类。

控制反转——Spring 通过一种称作控制反转（IoC）的技术促进了松耦合。当应用了 IoC，一个对象依赖的其它对象会通过被动的方式传递进来，而不是这个对象自己创建或者查找依赖对象。你可以认为 IoC 与 JNDI 相反——不是对象从容器中查找依赖，而是容器在对象初始化时不等对象请求就主动将依赖传递给它。

面向切面——Spring 提供了面向切面编程的丰富支持，允许通过分离应用

的业务逻辑与系统级服务（例如审计（auditing）和事务（transaction）管理）进行内聚性的开发。应用对象只实现它们应该做的——完成业务逻辑——仅此而已。它们并不负责（甚至是意识）其它的系统级关注点，例如日志或事务支持。

容器——Spring 包含并管理应用对象的配置和生命周期，在这个意义上它是一种容器，你可以配置你的每个 bean 如何被创建——基于一个可配置原型（prototype），你的 bean 可以创建一个单独的实例或者每次需要时都生成一个新的实例——以及它们是如何相互关联的。

框架——Spring 可以将简单的组件配置、组合成为复杂的应用。在 Spring 中，应用对象被声明式地组合，典型地是在一个 XML 文件里。Spring 也提供了很多基础功能（事务管理、持久化框架集成等等），将应用逻辑的开发留给了你。

所有 Spring 的这些特征使你能够编写更干净、更可管理、并且更易于测试的代码。它们也为 Spring 中的各种模块提供了基础支持。

3.6.4.2.3 ORM 框架 MyBatis

MyBatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。

MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，将接口和 Java 的 POJOs(PlainOrdinaryJavaObject,普通的 Java 对象)映射成数据库中的记录。

具备以下优点：

1 简单易学：本身就很小且简单。没有任何第三方依赖，最简单安装只要两个 jar 文件+配置几个 sql 映射文件易于学习，易于使用，通过文档和源代码，可以比较完全的掌握它的设计思路和实现；

2 灵活：mybatis 不会对应用程序或者数据库的现有设计强加任何影响。sql 写在 xml 里，便于统一管理和优化。通过 sql 语句可以满足操作数据库的所有需求；

3 解除 sql 与程序代码的耦合：通过提供 DAO 层，将业务逻辑和数据访问逻辑分离，使系统的设计更清晰，更易维护，更易单元测试。sql 和代码的分离，提高了可维护性；

4 提供映射标签，支持对象与数据库的 orm 字段关系映射；

5 提供对象关系映射标签，支持对象关系组建维护；

6 提供 xml 标签，支持编写动态 sql；

3.6.4.2.4 数据源组件

（1）Druid 数据库连接池

数据库连接是一种关键的、有限的、昂贵的资源。传统的模式（如传统的 java web 项目中，servlet 的 beans 中建立数据库连接），每次连接都需要验证用户，消耗了大量的时间和资源。而数据库连接池在系统初始化的时候，将数据库连接作为对象存储在内存中，当用户需要访问数据库时，并非建立一个新的连接，而是从连接池中取出一个已经建立的空闲连接对象。使用完毕后，用户不关闭连接，而是将数据库连接对象放回连接池中。数据库连接池管理数据连接的建立、断开，同时监视数据库连接数量和使用情况。使用数据库连接池会显著提高整个应用程序的伸缩性（大大提高了连接数量）和健壮性（能够应对大量用户频繁连接数据库，减少系统资源的消耗），提高应用程序的性能指标。平台采用 Druid 作为底层数据库连接池管理工具，Druid 阿里巴巴开发的号称为监控而生的数据库连接池(可以监控访问数据库的性能)，Druid 是目前最好的数据库连接池。在功能、性能、扩展性方面，都超过其他数据库连接池。

3.6.4.2.5 日志组件 Logback

Logback 是一个开源日志组件。logback 当前分成三个模块：logback-core,logback- classic 和 logback-access。logback-core 是其它两个模块的基础模块。logback-classic 是 log4j 的一个 改良版本。此外 logback-classic 完整实现 SLF4J API 使你可以很方便地更换成其它日志系统如 log4j 或 JDK14 Logging。logback-access 访问模块与 Servlet 容器集成提供通过 Http 来访问日志的功能。。

3.6.4.2.6 缓存组件

（1）Spring Cache

Spring Cache 是 Spring 3.1 引入的基于注释（annotation）的缓存（cache）技术，它本质上不是一个具体的缓存实现方案（例如 EHCache 或者 OSCache），而是一个对缓存使用的抽象，通过在既有代码中添加少量它定义的各种 annotation，即能够达到缓存方法的返回对象的效果。

Spring Cache 的缓存技术还具备相当的灵活性，不仅能够使用 SpEL（Spring Expression Language）来定义缓存的 key 和各种 condition，还提供开箱即用的缓存临时存储方案，也支持和主流的专业缓存例如 EHCache 集成。

其特点总结如下：

- 通过少量的配置 annotation 注释即可使得既有代码支持缓存；
- 支持开箱即用 Out-Of-The-Box，即不用安装和部署额外第三方组件即可使用缓存；
- 支持 Spring Express Language，能使用对象的任何属性或者方法来定义缓存的 key 和 condition；
- 支持 AspectJ，并通过其实现任何方法的缓存支持；
- 支持自定义 key 和自定义缓存管理者，具有相当的灵活性和扩展性。

（2）Redis

Redis 是一个 key-value 存储系统。和 Memcached 类似，它支持存储的 value 类型相对更多，包括 string(字符串)、list(链表)、set(集合)、zset(sorted set --有序集合)和 hash（哈希类型）。这些数据类型都支持 push/pop、add/remove 及取交集并集和差集及更丰富的操作，而且这些操作都是原子性的。在此基础上，redis 支持各种不同方式的排序。与 memcached 一样，为了保证效率，数据都是缓存在内存中。区别的是 redis 会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了 master-slave(主从)同步。

Redis 是一个高性能的 key-value 数据库。redis 的出现，很大程度补偿了 memcached 这类 key/value 存储的不足，在部 分场合可以对关系数据库起到很好的补充作用。它提供了 Java，C/C++，C#，PHP，JavaScript，Perl，Object-C，Python，Ruby，Erlang 等客户端，使用很方便。

Redis 支持主从同步。数据可以从主服务器向任意数量的从服务器上同步，从服务器可以是关联其他从服务器的主服务器。这使得 Redis 可执行单层树复制。存盘可以有意无意的对数据进行写操作。由于完全实现了发布/订阅机制，使得从数据库在任何地方同步树时，可订阅一个频道并接收主服务器完整的消息发布记录。同步对读取操作的可扩展性和数据冗余很有帮助。

3.6.4.2.7 服务管控 Nacos

Nacos 是阿里巴巴开源的一款支持服务注册与发现，配置管理以及微服务管理的组件。用来取代以前常用的注册中心（zookeeper ， eureka 等等），以及配置中心（spring cloud config 等等）。Nacos 是集成了注册中心和配置中心的功能，做到了二合一，提升运维效率。

3.6.4.2.8 开发工具 IDEA

IDEA 全称 IntelliJ IDEA，是用于 java 语言开发的集成环境（也可用于其他语言），IntelliJ 在业界被公认为最好的 java 开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE 支持、Ant、JUnit、CVS 整合、代码审查、创新的 GUI 设计等方面的功能可以说是超常的。IDEA 是 JetBrains 公司的产品，这家公司总部位于捷克共和国的首都布拉格，开发人员以严谨著称的东欧程序员为主。

3.6.4.2.9 构建工具

（1）Gradle

Gradle 是一个基于 Apache Ant 和 Apache Maven 概念的项目自动化构建开源工具。它使用一种基于 Groovy 的特定领域语言(DSL)来声明项目设置，抛弃了基于 XML 的各种繁琐配置。

Gradle 是一个基于 JVM 的构建工具，是一款通用灵活的构建工具，支持 maven，Ivy 仓库，支持传递性依赖管理，而不需要远程仓库或者是 pom.xml 和 ivy.xml 配置文件，基于 Groovy，build 脚本使用 Groovy 编写。

（2）Maven

Maven 是一个项目管理工具，它包含了一个项目对象模型 (Project Object Model)，一组标准集合，一个项目生命周期(Project Lifecycle)，一个依赖管理系统(Dependency Management System)，和用来运行定义在生命周期阶段(phase)中插件(plugin)目标(goal)的逻辑。当你使用 Maven 的时候，你用一个明确定义的项目对象模型来描述你的项目，然后 Maven 可以应用横切的逻辑，这些逻辑来自一组共享的（或者自定义的）插件。

Maven 有一个生命周期，当你运行 mvn install 的时候被调用。这条命令告诉 Maven 执行一系列的有序的步骤，直到到达你指定的生命周期。遍历生命

周期旅途中的一个影响就是，Maven 运行了许多默认的插件目标，这些目标完成了像编译和创建一个 JAR 文件这样的工作。

此外，Maven 能够很方便的帮你管理项目报告，生成站点，管理 JAR 文件等等。

3.6.4.3 数据库选型

3.6.4.3.1 关系型数据库

平台采用关系型数据库技术存储业务数据，关系型数据库，是指采用了关系模型来组织数据的数据库，其以行和列的形式存储数据，以便于用户理解，关系型数据库这一系列的行和列被称为表，一组表组成了数据库。

用户通过查询来检索数据库中的数据，而查询是一个用于限定数据库中某些区域的执行代码。关系模型可以简单理解为二维表格模型，而一个关系型数据库就是由二维表及其之间的关系组成的一个数据组织。

其具备以下优秀特点：

1.存储方式：传统的关系型数据库采用表格的储存方式，数据以行和列的方式进行存储，要读取和查询都十分方便；

2.存储结构：关系型数据库按照结构化的方法存储数据，每个数据表都必须对各个字段定义好（也就是先定义好表的结构），再根据表的结构存入数据，这样做的好处就是由于数据的形式和内容在存入数据之前就已经定义好了，所以整个数据表的可靠性和稳定性都比较高；

3.存储规范：关系型数据库为了避免重复、规范化数据以及充分利用好存储空间，把数据按照最小关系表的形式进行存储，这样数据管理的就可以变得很清晰、一目了然；

4.查询方式：关系型数据库采用结构化查询语言（即 SQL）来对数据库进行查询，SQL 早已获得了各个数据库厂商的支持，成为数据库行业的标准，它能够支持数据库的 CRUD（增加，查询，更新，删除）操作，具有非常强大的功能，SQL 可以采用类似索引的方法来加快查询操作；

5.规范化：在数据库的设计开发过程中开发人员通常会面对同时需要对一个或者多个数据实体（包括数组、列表和嵌套数据）进行操作，这样在关系型数据

库中，一个数据实体一般首先要分割成多个部分，然后再对分割的部分进行规范化，规范化以后再分别存入到多张关系型数据表中，这是一个复杂的过程。好消息是随着软件技术的发展，相当多的软件开发平台都提供一些简单的解决方法，例如，可以利用 ORM 层（也就是对象关系映射）来将数据库中对象模型映射到基于 SQL 的关系型数据库中去以及进行不同类型系统的数据之间的转换；

6.事务性：关系型数据库强调 ACID 规则（原子性（Atomicity）、一致性（Consistency）、隔离性（Isolation）、持久性（Durability）），可以满足对事务性要求较高或者需要进行复杂数据查询的数据操作，而且可以充分满足数据库操作的高性能和操作稳定性的要求。并且关系型数据库十分强调数据的强一致性，对于事务的操作有很好的支持。关系型数据库可以控制事务原子性细粒度，并且一旦操作有误或者有需要，可以马上回滚事务；

借助 J2EE 规范和 SQL 规范的优势，平台在数据库方面兼容性很好，原则上支持市面上所有的关系型数据库产品，如 MYSQL/大梦/Oracle 等。

3.6.4.3.2 非关系数据库技术

平台采用非关系数据库技术存储非业务数据，其中平台元数据采用文档数据库 MONGODB 存储，MongoDB 是非关系数据库当中功能最丰富，最像关系数据库的。它支持的数据结构非常松散，是类似 json 的 bson 格式，因此可以存储比较复杂的数据类型。

Mongo 最大的特点是它支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立索引。

缓存数据库则采用 REDIS 存储，redis 是一个 key-value 存储系统。和 Memcached 类似，它支持存储的 value 类型相对更多，包括 string(字符串)、list(链表)、set(集合)、zset(sortedset--有序集合)和 hash（哈希类型）。这些数据类型都支持 push/pop、add/remove 及取交集并集和差集及更丰富的操作，而且这些操作都是原子性的。

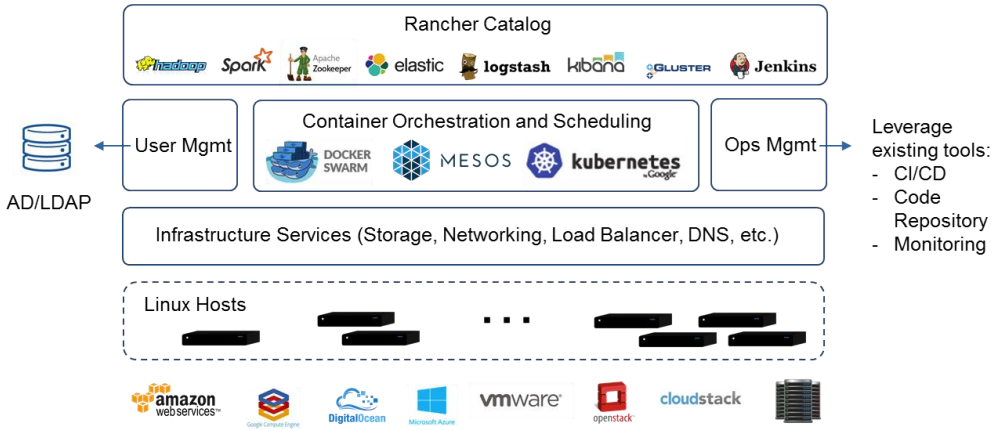
在此基础上，redis 支持各种不同方式的排序，与 memcached 一样，为了保证效率，数据都是缓存在内存中。区别的是 redis 会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了

master-slave(主从)同步。

3.6.4.4 主要中间件

3.6.4.4.1 容器管理平台 Rancher

Rancher 是一个开源的企业级容器管理平台。通过 Rancher，企业再也不必自己使用一系列的开源软件去从头搭建容器服务平台。Rancher 提供了在生产环境中使用的管理 Docker 和 Kubernetes 的全栈化容器部署与管理平台，包括基础设施编排、容器编排与调度、应用商店、企业级权限管理十大能力。



3.6.4.4.2 业务流程管理平台（BPM）

城市体检服务平台核心业务需要利用主流的业务流程管理（BPM）平台实现业务流程的全生命周期管理，包括业务流程的定义、设计、执行、评估、优化等。BPM 主要支持：流程和表单可视化建模（拖拽模式）；流程服务的编排；多流程场景应用模式，包括：同步和异步子流程、自由跳转、动态任务委派、批量会签和动态加减签、任务代理、催办等；流程表单一体化集成；流程多版本管理；流程 KPI 绩效管理；流程仿真模拟、调试，并支持开发、测试和生产三套环境的应用模式等，平台通过自研工作流平台 BUS 实现 BPM 功能。

3.6.4.4.3 商业智能平台（BI）

城市体检业务涉及到各种结构化、非结构化和空间数据，需要对房源数据、保障对象以及租售补贴等数据及其历史数据开展专题分析，构建面向问题或某一主题的业务分析和数据挖掘模型，为城市体检决策提供辅助支撑，提供领导驾驶舱服务。这些都需要商业智能平台(BI)的支持。

本项目建设将采购主流的商业智能平台（BI）用于支撑数据决策等系统应用开发。

3.6.4.4.4 企业级 Web 报表工具

城市体检业务管理过程中，涉及不同层级、不同类型大量专业报表定制工作，并且需要根据业务“随需应变”，都需要有企业级报表工具支撑，为所有的报表应用提供统一的报表建模、报表查阅、输出及打印提供服务。

报表工具要求采用标准纯 html 方式展现，支持各类浏览器，提供了高效的报表设计方案、强大的报表展现能力、灵活的部署机制，支持强关联语义模型，并且具备强有力的填报功能，能够与数据分析与商业智能进行对接，实现了高性能、高效率的报表定制和输出，平台提供积木报表工具，实现打印报表、数据报表绘制。

3.6.4.4.5 数据仓库工具（ETL）

城市体检要实现决策支持应用的前提是建立统一的数据仓库，建设仓库的过程中需要使用数据仓库工具对城市体检原始数据实行深加工、深处理，通过抽取-转换-加载，浓缩其价值密度，构建多维数据集市等。根据应用的需要，对数据仓库中的某些数据实行深加工、深处理，通过抽取-转换-加载，浓缩其价值密度，构建多维数据集市。针对加工处理后的高密度价值数据，应建立起相应的城市体检数据仓库或数据集市，实现更加有效的管理和利用。

为了降低采购成本，本项目建议采用企业级 ETL 工具作为数据仓库工具。

3.6.4.4.6 地理信息基础平台（GIS）

在城市体检业务中，房源数据管理涉及到空间数据，同时需要接入其他委办局的基础地理数据服务，为后续智能应用提供位置服务。

和非空间数据不同，空间数据的生产、存储、坐标参考、空间索引、空间查询、拓扑分析、图形展示等，都有其特殊性。地理信息基础平台提供了一整套技术方案来解决上述问题：在数据生产方面，提供业务数据和 GIS 无缝衔接的技术；在数据存储方面，通过扩展关系型数据库，实现空间数据和非空间数据的一体化存储；在数据应用方面，基于空间数据标准，提供一整套数据查询、分析接口；在数据展示方面，提供丰富的符号化方法，以及适应多平台的空间数据展示接口

等。

无特殊要求本项目采用互联网地图企业（高德、百度）或天地图。

3.6.4.4.7 分布式文件系统（OSS）

公共住房管理中有大量非结构化数据，比如交易合同档案扫描件、监控视频等需要进行管理，满足检索要求。对于那些不方便用二维关系表来组织的数据，数据结构不规则或不完整的数据，没有预定义数据模型的数据，都称之为非结构化数据。常见的非结构化数据包括：所有格式的办公文档、文本、图片，通用标记语言子集 XML、HTML，各类报表、图像、音频、视频数据等。在所有数据中，非结构化数据占比可达 80%甚至更多。

随着 IT 技术的发展和业务需求的深化，人们更加重视对非结构化数据的价值挖掘，非结构化数据管理与数据分析、业务分析洞察的结合愈加紧密。

采用 OSS 来统一管理非结构化数据，并满足全文检索的需求。其主要特点如下：

- 高可靠性：支持一个对象多个副本，保证数据的高可靠性。当一个节点失效后，自动隔离，保证系统的高可用性。当某个数据丢失时，通过副本，可自动恢复。
- 实现高扩展性：整个数据是大规模分布式存储的，不是集中存放在单个物理存储设备上，可以水平扩展。后端服务器的动态扩展并不影响前端应用系统的正常运行。
- 数据和元数据分离：两者分别采用集群部署，支持动态增删节点，避免单点故障和性能瓶颈。
- 数据读取高性能：文件分片，采用条带存储。通过带缓存的数据读写，满足大数据量并发访问的高性能要求。
- 支持分布式部署，并且支持全文检索。

3.6.4.4.8 服务监管平台（APIM）

在平台实现面向微服务架构时，会将导致很多应用资产的服务化。城市体检业务体系庞大，所产生的服务种类数量众多，这些服务除了需要注册、发布、应用、注销等全生命周期管理外，还需要对这些服务的运行情况、访问授权情况实

行管理与监控，保证每类服务都在正常运行。

服务监管中间件（API Monitor，APIM）对服务的一种表现形式，即应用程序开发接口（API）实行管理和监控。其主要特性如下：

- 端到端的 API 服务集成：通过 API 服务的创建、服务化和集成，可降低应用程序（APP）开发成本，简化应用开发过程。
- 自动化 API 创建与发现：把遗留业务系统中的可复用功能转化为 API 服务，有助于原有应用的改进，或者形成新的应用场景。
- 自助式 API 访问：可加快开发人员对 API 的发现，为 API 提供运营安全与治理机制，以提高应用开发效率。
- 支持多种开发语言和开发框架，减少对具体开发技术的依赖。

使用 API,可强化基础平台的灵活性，降低平台运行的总体拥有成本等。本项目拟采购成熟的服务监管中间件（APIM）来满足需求。

3.6.4.5 二次开发支持

平台支持二次开发扩展，主要通过 URL 界面集成和接口集成。URL 集成主要通过界面集成，简单的网址通过平台管理端管理起来，具体的 URL 由第三方开发完成。接口集成主要包括：REST API 和 JAVA SDK 方式进行集成。具体的技术如下：

3.6.4.5.1 REST API

REST 即表述性状态传递（英文：Representational State Transfer，简称 REST）是 Roy Fielding 博士在 2000 年他的博士论文中提出来的一种软件架构风格。它是一种针对网络应用的设计和开发方式，可以降低开发的复杂性，提高系统的可伸缩性。

目前在三种主流的 Web 服务实现方案中，因为 REST 模式的 Web 服务与复杂的 SOAP 和 XML-RPC 对比来讲明显的更加简洁,越来越多的 web 服务开始采用 REST 风格设计和实现。

表述性状态转移是一组架构约束条件和原则。满足这些约束条件和原则的应用程序或设计就是 RESTful。需要注意的是，REST 是设计风格而不是标准。REST 通常基于使用 HTTP，URI，和 XML（标准通用标记语言下的一个子集）以及 HTML

（标准通用标记语言下的一个应用）这些现有的广泛流行的协议和标准。

REST 定义了一组体系架构原则，可以根据这些原则设计以系统资源为中心的 Web 服务，包括使用不同语言编写的客户端如何通过 HTTP 处理和传输资源状态。如何考虑使用它的 Web 服务的数量,REST 近年来已经成为最主要的 Web 服务设计模式。事实上，REST 对 Web 的影响非常大，由于其使用相当方便，已经普遍地取代了基于 SOAP 和 WSDL 的接口设计。

3.6.4.5.2 Java SDK

Java 是一门面向对象编程语言，不仅吸收了 C++语言的各种优点，还摒弃了 C++里难以理解的多继承、指针等概念，因此 Java 语言具有功能强大和简单易用两个特征。Java 语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。

Java 具有简单性、面向对象、分布式、健壮性、安全性、平台独立与可移植性、多线程、动态性等特点。Java 可以编写桌面应用程序、Web 应用程序、分布式系统和嵌入式系统应用程序等。

（1）简单性

Java 看起来设计得很像 C++，但是为了使语言小和容易熟悉，设计者们把 C++语言中许多可用的特征去掉了，这些特征是一般程序员很少使用的。例如，Java 不支持 go to 语句,代之以提供 break 和 continue 语句以及异常处理。Java 还剔除了 C++的操作符过载（overload）和多继承特征，并且不使用主文件，免去了预处理程序。因为 Java 没有结构，数组和串都是对象，所以不需要指针。Java 能够自动处理对象的引用和间接引用，实现自动的无用单元收集，使用户不必为存储管理问题烦恼，能更多的时间和精力花在研发上。

（2）面向对象

Java 是一个面向对象的语言。对程序员来说，这意味着要注意应中的数据和操纵数据的方法（method），而不是严格地用过程来思考。在一个面向对象的系统中，类（class）是数据和操作数据的方法的集合。数据和方法一起描述对象（object）的状态和行为。每一对象是其状态和行为的封装。类是按一定体系和层次安排的,使得子类可以从超类继承行为。在这个类层次体系中有一个根类，它是具有一般行为的类。Java 程序是用类来组织的。

Java 还包括一个类的扩展集合，分别组成各种程序包（Package），用户可以在自己的程序中使用。例如，Java 提供产生图形用户接口部件的类（java.awt 包），这里 awt 是抽象窗口工具集（abstract windowing toolkit）的缩写，处理输入输出的类（java.io 包）和支持网络功能的类（java.net 包）。

（3）分布性

Java 设计成支持在网络上应用，它是分布式语言。Java 既支持各种层次的网络连接，又以 Socket 类支持可靠的流（stream）网络连接，所以用户可以产生分布式的客户机和服务器。

网络变成软件应用的分布运载工具。Java 程序只要编写一次，就可到处运行。

（4）编译和解释性

Java 编译程序生成字节码（byte-code），而不是通常的机器码。Java 字节码提供对体系结构中性的目标文件格式，代码设计成可有效地传送程序到多个平台。Java 程序可以在任何实现了 Java 解释程序和运行系统（run-time system）的系统上运行。

在一个解释性的环境中，程序开发的标准“链接”阶段大大消失了。如果说 Java 还有一个链接阶段，它只是把新类装进环境的过程，它是增量式的、轻量级的过程。因此，Java 支持快速原型和容易试验，它将导致快速程序开发。这是一个与传统的、耗时的“编译、链接和测试”形成鲜明对比的精巧的开发过程。

（5）稳健性

Java 是一个强类型语言，它允许扩展编译时检查潜在类型不匹配问题的功能。Java 要求显式的方法声明，它不支持 C 风格的隐式声明。这些严格的要求保证编译程序能捕捉调用错误，这就导致更可靠的程序。

可靠性方面最重要的增强之一是 Java 的存储模型。Java 不支持指针，它消除重写存储和讹误数据的可能性。类似地，Java 自动的“无用单元收集”预防存储漏泄和其它有关动态存储分配和解除分配的有害错误。Java 解释程序也执行许多运行时的检查，诸如验证所有数组和串访问是否在界限之内。

异常处理是 Java 中使得程序更稳健的另一个特征。异常是某种类似于错误的异常条件出现的信号。使用 try/catch/finally 语句，程序员可以找到出错的处理代码，这就简化了出错处理和恢复的任务。

（6）安全性

Java 的存储分配模型是它防御恶意代码的主要方法之一。Java 没有指针，所以程序员不能得到隐蔽起来的内幕和伪造指针去指向存储器。更重要的是，Java 编译程序不处理存储安排决策，所以程序员不能通过查看声明去猜测类的实际存储安排。编译的 Java 代码中的存储引用在运行时由 Java 解释程序决定实际存储地址。

Java 运行系统使用字节码验证过程来保证装载到网络上的代码不违背任何 Java 语言限制。这个安全机制部分包括类如何从网上装载。例如，装载的类是放在分开的名字空间而不是局部类，预防恶意的小应用程序用它自己的版本来代替标准 Java 类。

（7）可移植性

Java 使得语言声明不依赖于实现的方面。例如，Java 显式说明每个基本数据类型的大小和它的运算行为（这些数据类型由 Java 语法描述）。

Java 环境本身对新的硬件平台和操作系统是可移植的。Java 编译程序也用 Java 编写，而 Java 运行系统用 ANSIC 语言编写。

（8）高性能

Java 是一种先编译后解释的语言，所以它不如全编译性语言快。但是有些情况下性能是很要紧的，为了支持这些情况，Java 设计者制作了“及时”编译程序，它能在运行时把 Java 字节码翻译成特定 CPU（中央处理器）的机器代码，也就是实现全编译了。

Java 字节码格式设计时考虑到这些“及时”编译程序的需要，所以生成机器代码的过程相当简单，它能产生相当好的代码。

（9）多线程性

Java 是多线程语言，它提供支持多线程的执行（也称为轻便过程），能处理不同任务，使具有线程的程序设计很容易。Java 的 lang 包提供一个 Thread 类，它支持开始线程、运行线程、停止线程和检查线程状态的方法。

Java 的线程支持也包括一组同步原语。这些原语是基于监督程序和条件变量风范，由 C.A.R.Haore 开发的广泛使用的同步化方案。用关键词 synchronized，程序员可以说明某些方法在一个类中不能并发地运行。这些方法在监督程序控制之下，确保变量维持在一个一致的状态。

（10）动态性

Java 语言设计成适应于变化的环境，它是一个动态的语言。例如，Java 中的类是根据需要载入的，甚至有些是通过网络获取的。

3.6.4.6 其他关键技术

3.6.4.6.1 统一身份认证

平台内部按照领域对账户体系部分进行了划分，包括认证子系统和业务配置管理子系统，认证子系统职责是实现账户认证以及扩展认证，而配置管理子系统则实现账户新建、授权等功能操作，整个微服务平台采用基于统一架构实现统一认证，支持单点登录依权操作模式，一旦用户通过认证领域认证，即可无缝使用其他子系统，无需重新登录，平台整体采用 RBAC 权限模型，将权限资源细分到菜单、按钮、功能、数据，四个粒度，权限可捆绑到角色、用户对象上，用户认证后仅能拉去自己权限范围内的菜单、按钮、功能、数据权限，平台用户可细分为三类，包括运营、管理主体、从业主体类。

运营类权限最高，运营类账号可以为管理主体类账号创建管理员账号，管理主体管理员账号可以给自己所属管理主体创建业务账号，业务账号登录后可以办理业务。

在登录认证时，认证子系统默认集成了用户名密码(密码前端加密)、CA 硬件 KEY 两种认证模式，亦支持以账户唯一性为扩展点的第三方认证方式，包括电子身份证认证等，提升账户安全性，保障系统的安全。

3.6.4.6.2 面向微服务架构

微服务是一种服务间松耦合的、每个服务之间高度自治并且使用轻量级协议进行通信的可持续集成部署的分布式架构体系，相较与传统单体架构，微服务有以下几个突出特点：

松耦合：每个微服务内部都可以使用 DDD（领域驱动设计）的思想进行设计领域模型，服务间尽量减少同步的调用，多使用消息的方式让服务间的领域事件来进行解耦，最终形成高度内聚、高度可复用的领域组件。

轻量级协议：使用 Restful 风格的 API，成熟轻量级协议可以很好地支持跨语言开发的服务，可能有的微服务用 Java 语言实现，有的用 Go 语言，有的用

C++,但所有的语言都可以支持 Http 协议通信,所有的开发人员都能理解 Restful 风格 API 的含义。

高度自治和持续集成：微服务可以很好得和容器技术结合，容器技术比微服务出现得晚，但是容器技术的出现让微服务的实施更加简便，目前 Docker 已经成为很多微服务实践的基础容器。因为容器的特色，开发者只需要构建好镜像就可以在多个平台运行,且一台机器上可以部署几十个、几百个不同的微服务实例，随着系统吞吐的变化运营人员还可根据负载压力配置相应应用服务容器的自动上线、下线。从而实现应用的弹性扩展和一体化监控管理。同时，因为 Docker 的容器编排社区日渐成熟，类似 Mesos、Kubernetes 及 Docker 官方提供的 Swarm 都可以作为持续集成部署的技术选择。

基于以上特性，微服务非常适合构建平台化产品，将业务功能作为服务资产进行输出，提升每个微服务的内聚性、敏捷性，缩短建设周期，减少维护成本，提升稳定性。

3.6.4.6.3 数据分析和展现技术

城市体检领域历史积累了大量数据，未来的数据量将更大。如何更好地利用这些数据？这带来的不仅是机遇，同时也是挑战。传统的数据分析统计手段已经无法满足领导快速分析、全面决策的需求，需要采用数据分析技术，来应对大数据量下的实时数据分析的需要。涉及的技术主要包括五大类，其中：

基础架构支持：主要包括为支撑各类数据处理的基础架构级数据中心管理、存储设备及技术、网络技术、资源监控等技术。需要帮助数据管理人员，有效、实时地管理和运维这些数据。

数据采集技术：数据采集技术是数据处理的必备条件。首先需要有数据采集手段，把数据收集上来，才能应用上层的数据处理技术。数据采集除了各类传感设备等硬件、软件设施之外，主要涉及到的是数据 ETL（采集-转换-加载）过程，能对数据进行清洗、过滤、校验、转换等各种预处理，将有效的数据转换成适合的格式和类型。为了支持多源异构的数据采集和存储访问，还需设计数据总线，方便各个应用和服务之间的数据交换和共享。

数据存储技术：数据经过采集和转换之后，需要进行合理的存储归档。针对大批量的数据，一般可以采用文件式存储和数据库存储的混合方式，对低价值/

高价值数据，定期/实时分析数据，多维/简单分析数据，进行不同方式的存储方式管理。

数据计算：把与数据查询、统计、分析、预测、挖掘、图谱处理、业务智能（BI）等各项相关技术统称为数据计算技术。数据计算技术涵盖数据处理的方方面面，是大数据分析技术的核心。

数据展现与交互：数据展现与交互在数据分析技术中也至关重要，因为数据最终需要为人们所使用，为生产、运营、规划提供决策支持。选择恰当的、生动直观的展示方式，能够帮助用户更好地理解数据内涵及其关联关系，也能够更有效地解释和运用数据，发挥其价值。在展现方式上，除了传统的报表、图形之外，还可以结合可视化工具及人机交互手段，甚至未来可以基于如 Google 眼镜等增强现实手段，来实现数据与现实的无缝接口。

3.6.4.6.4 HTML5 技术

城市体检业务本身具有很强的复杂性，其信息化应用涉及大量的图、文、表、档信息。虽然业务流程相对固定，但在实际处理过程中，由于涉及多个业务部门，涉及多层级别领导，对业务信息的精确性要求极高，如果系统操作复杂，步骤繁多，信息散乱，不仅运行效率低下，而且极有可能给业务办理带来麻烦，降低工作效率，甚至出现差错。

传统的 Web 应用程序对复杂的业务应用支持较弱，用户与系统交互时，体验性差。通过采用 HTML5 技术，充分利用其两大技术特点——强化的 Web 网页表现性能，以及支持本地数据库的 Web 应用功能——增强前端的 Web 展现，以改善系统的用户交互体验。其主要优势包括：较好的可用性，非常友好的用户交互体验，更好的多媒体页面元素支持，更好的可移植性，更好的移动应用支持等。目前，HTML5 已逐步成为前端页面开发的主流技术，将成为未来 5-10 年内，移动、互联网领域的主宰者。

3.6.4.6.5 BIM 技术

BIM(Building Information Modeling)技术是一种应用于工程设计、建造、管理的数据化工具，通过对建筑的数据化、信息化模型整合，在项目策划、运行和维护的全生命周期过程中进行共享和传递，使工程技术人员对各种建筑信息作

出正确理解和高效应对，为设计团队以及包括建筑、运营单位在内的各方建设主体提供协同工作的基础，在提高生产效率、节约成本和缩短工期方面发挥重要作用。在本项目中，实现 BIM 模型与 GIS 数据的集成浏览查看。

3.6.4.6.6 移动互联技术

在政府信息化走向企业级架构的今天，移动互联无疑起到了积极的推动作用。移动互联提供用户全天候接入、信息及时获取、有针对性的应用体验，已经成为最被看好的信息系统接入和办公方式之一。

其典型的应用场景可分为：一是满足内部办公需要，例如领导可以随时随地办公，及时处理案件或事务；二是满足内部信息服务需要，业务人员可随时获取工作相关资料、为参加会议了解相关项目情况、周期性地了解决策数据和相关报表，以及了解项目的空间布局情况等；三是满足公共服务需要，通过信息公开、公众参与、咨询、在线查询等移动 APP 服务，可加强政府与社会公众之间的联系和互动。

随着 4G、5G 时代的到来，移动办公已经成为大势所趋。移动互联技术与移动安全技术也日渐成熟，为移动应用创造了良好的技术条件。

3.6.4.6.7 业务规则

业务规则管理为几乎为所有行业的组织机构提供了关键规则管理技术，是业务的监测监管并实现其自动化的强大解决方案。规则平台的最佳应用是与业务流程管理平台（BPM）的组合应用，应用时可以将流程中经常变化的部分设计为规则服务，这样可实现业务流程与业务规则的剥离，让流程更加简洁、敏捷，大大简化业务流程建模和业务流程版本的发布，让规则管理集中对外提供统一的规则服务。同时用户可随时根据业务需要启用或关停某项规则服务。启用规则后，前端在办理业务时会立即检测业务是否满足规则要求。

在具体的城市体检业务中的资格审查规则、监测监管规则管理及其应用上将充分利用规则技术。具体的架构如下：

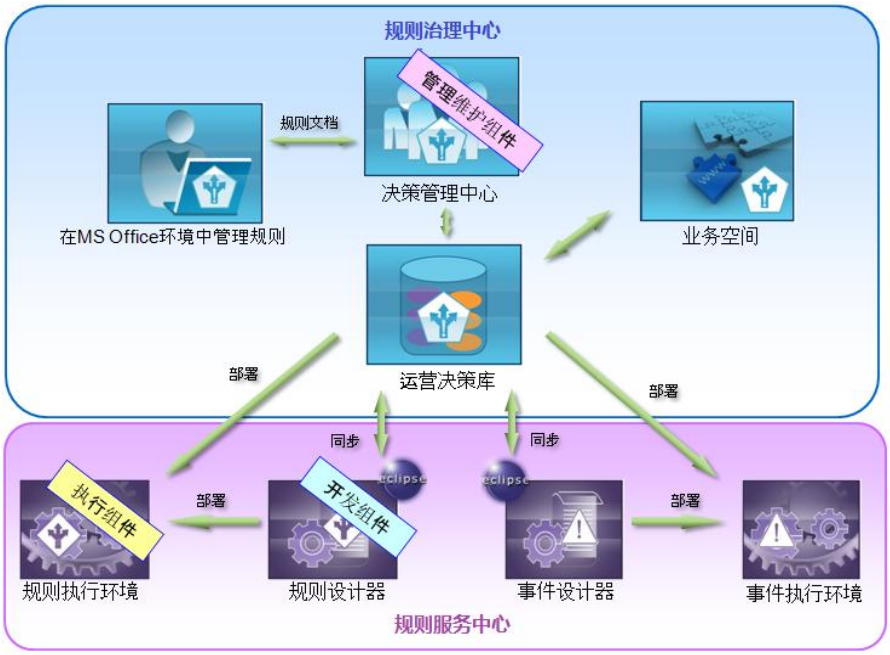


图 9- 8 业务规则管理组件架构

3.6.4.6.8 电子签章技术

为了满足全程网办关键需求，除了采用人工智能如人脸识别等新技术外，也需采用电子签章技术集成应用。电子签名主要应用包括签名和印章等。签名和印章是一份文件生效的象征或标识，不仅具有权威性，还是一种防伪体现。在业务办公系统中通常应用电子签章技术来提高电子文件的法律效力和可信任度，例如在业务审批、公文管理、公文交换模块中的电子签名或电子印章（公章）功能。电子签章是利用图像处理技术将印章或手写签名转化为与纸质文件盖章操作相同的可视效果，同时利用电子签名技术保障电子信息的真实性和完整性以及签名人的不可否认性。电子签章技术的应用形式包含电子印章和电子签名两种形式，在电子文件中加盖印章，与传统的手写签名、盖章文件具有同等的法律效应。

电子印章其实是将电子文书内容的数字签名通过数字水印、加密等技术，和电子印章图像进行有效的绑定（如利用隐藏技术将数据隐藏在电子印章的图像中等），无论是加盖印章或者手写签名的形式，都需要进行签章验证，只有通过电子签名验证技术证明与电子印章相关联的电子文书是真实的，电子印章的图像才被承认是有效的，否则就只能是一张简单的图片。

3.6.4.6.9 AR/VR 技术

虚拟现实/增强现实（Virtual Reality/Augmented Reality，VR/AR）是仿

真技术的一类应用，是仿真技术与计算机图形学、人机接口技术、多媒体技术、传感技术、网络技术等多种技术的集合。

虚拟现实技术（VR）主要包括模拟环境、感知、自然技能和传感设备等方面。也有人称为VR艺术，即以虚拟现实（VR）、增强现实（AR）等人工智能技术作为媒介手段而加以运用的艺术形式。虚拟现实（VR）应用往往需要佩戴一定设备，让人沉浸在虚拟的世界中。而增强现实（AR）是虚拟世界与现实世界的信息集成，并且具有互动特性。在“达州市城市体检信息平台”的建设中，考虑采用VR技术实现场景看房，采用AR技术实现场景选房。

3.6.4.6.10 服务化、云化

总体架构设计在结合新需求，引入新技术方面，充分利用信息中心的云环境资源，重点实现服务化与云化两个特性。

（1）服务化

服务化是微服务架构的基础，是云化的前提。

传统架构模式往往会产生各应用系统的相互割裂，并最终形成应用“烟囱”，已有的信息资产难以得到充分复用。服务化正是解决这些问题的最佳实践。服务化过程中，通常需要将信息资产实行服务化改造。对于使用效果良好、使用频率较高的历史应用系统，可以将其内部资产包装为服务组件，注册到企业服务总线或服务管理平台上，可供其他应用系统使用，提高了复用率。在完成服务化后，对服务资产需要提供全生命周期管理，包括服务开发、注册、组合等。此时，可以使用微服务中间件来完成服务编排组合、服务管控等需求。

（2）云化

云计算主要包括三种基本模式：私有云、公有云和混合云模式。在架构上，通常分为三层:基础设施即服务(IaaS)、平台即服务(PaaS)和软件即服务(SaaS)。SaaS解决的是应用层的服务化，用户无需采购软件，只需使用网络租用软件服务即可；PaaS提供平台级服务能力，包括开发接口（API）调配、服务引擎、流程服务等能力，并提供这些服务能力的监控；IaaS是一种资源共享服务模式，即将物理资源包括内存、硬盘、CPU、网络等虚拟化、池化，以提高物理资源的利用率、可扩展性、可靠性等，并且大大提高业务应用的部署速度，具有按需自动扩容、故障自动转移、负载均衡等特性。

通过 IT 应用资产的服务化和云化，将极大地提高“达州市城市体检信息平台”对业务应用系统的支撑力度，加强信息资源共享，提高平台应变能力，打造一个随需应变的信息化政务环境。

3.6.4.6.11 基础设施集群

为了保证平台的可用性、高性能、高稳定性、高并发等特性，基础设施的集群是必需的。集群主要有主备集群、互备集群和并行集群三种模式。

（1）主备集群：主机工作，备机处于监控准备状态。当主机宕机时，备机接管主机的一切工作；待主机恢复正常后，按使用者的设定以自动或手动方式将服务切换到主机上运行。数据的一致性通过共享存储系统来解决。

（2）互备集群：两台主机同时运行各自的服务且相互监测情况，当任一台主机宕机时，另一台主机立即接管它的一切工作，保证正常工作。应用服务系统的关键数据存放在共享存储系统中。

（3）并行集群：多台主机一起工作，各自运行一个或几个服务，各为服务定义一个或多个备用主机，当某个主机故障时，运行在其上的服务就可以被其它主机接管。

目前，采用并行集群方式较多，在该模式下，集群有多台服务器构成，可以实现多台服务器之间的负载均衡，提供高访问量的应用需求，如 Web 访问及数据库等应用。服务器并行集群方式一般由应用系统自身（如主从模式 、集群模式、分布式模式等）或外部专用服务器负载均衡设备实现。“达州市城市体检信息平台”将采用并行集群模式。

3.7 安全体系

根据 GB/T 25070-2010 信息安全技术信息系统等级保护安全设计技术要求，对“达州市城市体检信息平台”进行定级。在信息系统安全定级基础上，从管理体系、技术体系上，实行安全系统同步设计规划与建设，形成多个层面的安全保障体系，包括基础设施层安全、数据层安全、应用层安全和安全管理等。如下图所示。

图 11- 1 安全保障体系结构图

在安全保障体系中，基础设施层安全主要考虑物理安全、网络安全和主机安全，数据层安全主要考虑数据传输安全、数据不可抵赖性设计、数据分级管理、数据备份、数据恢复和灾难恢复，应用层安全主要从身份认证、授权管理、安全审计和软件容错四方面考虑，安全管理主要从安全管理制度、安全管理机构、人员安全管理和系统建设管理四方面考虑。

本项目按照三级等保要求进行规划，主要包括云平台和业务系统的安全保障两部分。其中：云平台的安全保障在技术方面按照分层、纵深防御的思想，基于安全域的划分，从物理基础设施、虚拟化、网络、系统、应用、数据等层面进行综合防护；在管理方面，应对云平台、云服务、云数据的整个生命周期、安全事件、运行维护和监测、度量和评价进行管理；业务系统的安全保障采取统一身份认证、分级授权管理、第三方软件测评等措施，保障业务系统的运行安全。

3.7.1 信息安全总体框架

本项目依托于政务云平台部署，由大数据局提供统一的云平台安全保障，按照满足安全等保三级要求，信息安全总体框架如下图所示。

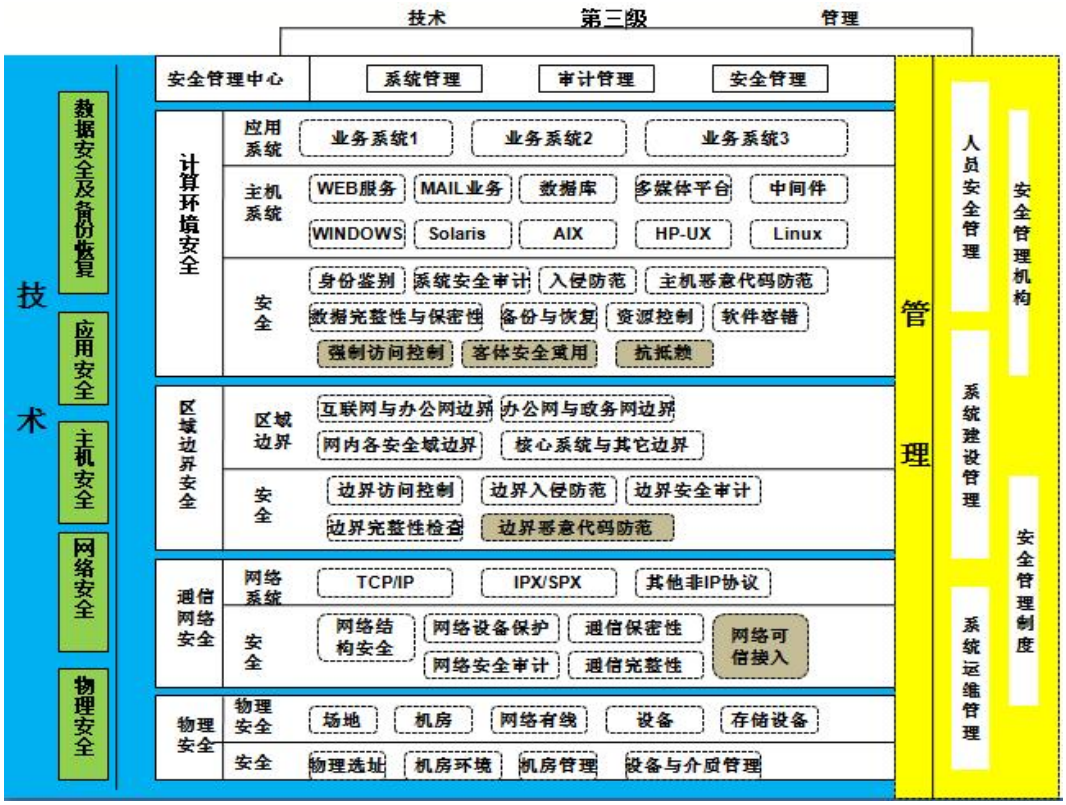


图 11- 2 信息安全总体框架

3.7.2 基础设施层安全

《计算机信息系统安全保护等级划分准则》明确要求物理安全包括环境安全、设备安全、记录介质安全。因此，物理安全主要在环境安全、设备安全、主机安全三个方面采取保护措施，如门控系统、监控报警系统和区域保护措施等。

3.7.2.1 环境安全

（1） 机房与设施安全

要保证信息系统的安全、可靠，必须保证系统实体有一个安全的环境条件。这个安全环境就是指机房及其设施，它是保证系统正常工作的基本环境。包括机房环境条件、机房安全等级、机房场地的环境选择、机房的建造、机房的装修和计算机的安全防护等。对系统所在环境的安全保护，如区域保护和灾难保护等在GB50174－93《电子计算机机房设计规范》、GA/T390-2002《计算机信息系统安全等级保护通用技术要求》、GB2887－2000《电子计算机场地通用规范》、GB9361－88《计算站场地安全要求》等标准中有详细的描述。

（2） 环境与人员安全

环境与人员安全通常是指防火、防水、防震、防振动冲击、防电源掉电、防温度湿度冲击、防盗以及防物理、化学和生物灾害等，是针对环境的物理灾害和人为蓄意破环而采取的安全措施和对策。

（3） 其他自然灾害防范

其他自然灾害主要包括湿度、洁净度、腐蚀、虫害、振动与冲击、噪音、电气干扰、地震、雷击等，需要充分考虑各类灾害对系统运行环境的威胁。

3.7.2.2 设备安全

设备安全主要包括计算机设备的防盗、防毁、防电磁泄漏发射、抗电磁干扰及电源保护等。

（1） 防盗与防毁

当计算机系统或设备被盗、被毁时，除了设备本身丢失或毁损带来的损失外，更多的损失则是失去了有价值的程序和数据。因此，防盗、防毁是计算机防护的一个重要内容。通常采取的防盗、防毁措施主要有：

设置报警器：在机房周围空间放置侵入报警器。侵入报警的形式主要有：光电、微波、红外线和超声波。

锁定装置：在计算机设备中，特别是在个人计算机中，设置锁定装置，以防犯罪盗窃。

计算机保险：在计算机系统受到侵犯后，可以得到损失的经济补偿，但是无法补偿失去的程序和数据，为此应设置一定的保险装置。

列出清单或绘出位置图：最基本的防盗安全措施是列出设备的详细清单，并绘出其位置图。

（2） 防止电磁泄露发射

抑制计算机中信息泄漏的技术途径有两种：一是电子隐蔽技术，二是物理抑制技术。电子隐蔽技术主要是用干扰、调频等技术来掩饰计算机的工作状态和保护信息；物理抑制技术则是抑制一切有用信息的外泄。物理抑制技术可分为包容法和抑源法。包容法主要是对辐射源进行屏蔽，以阻止电磁波的外泄传播。抑源法就是从线路和元器件入手，从根本上阻止计算机系统向外辐射电磁波，消除产生较强电磁波的根源。

（3） 防电磁干扰

“电磁干扰”是指当电子设备辐射出的能量超过一定程度时，就会干扰设备本身以及周围其他电子设备的现象。计算机与各种电子设备，广播、电视、雷达等无线设备，以及电子仪器等都会发出电磁干扰信号，计算机要在这样复杂的电磁干扰环境中工作，其可靠性、稳定性和安全性将受到严重影响。因此，实际使用中，需要了解和考虑计算机的抗电磁干扰问题，即电磁兼容性问题。

（4） 物理隔离

需要将不同安全等级要求的系统进行隔离，包括网络、主机系统等。

3.7.2.3 云平台安全

本项目主要基于云平台部署，需要特别关注云平台的安全。目前，国家质监总局和国家标准委联合有关部门，正在制定新的等保管理办法。从征求意见稿来看，将对云计算安全作出专项要求。

- **云计算安全责任划分**

云计算以服务为本质，包括云服务方和云租户两大责任主体，并且包括不同的服务模式和部署模式。其中，主要的部署模式包括公共云、专有云、社区云和混合云。主要的云计算服务模式包括基础设施即服务（IaaS）、平台即服务（PaaS）和软件即服务（SaaS）等。云计算环境的安全由云服务方和云租户共同保障。针对不同服务模式，云服务方和云租户的安全责任边界不同。对云服务组件的管理和控制权限范围，由二者决定各自的安全责任边界。具体如下图所示。

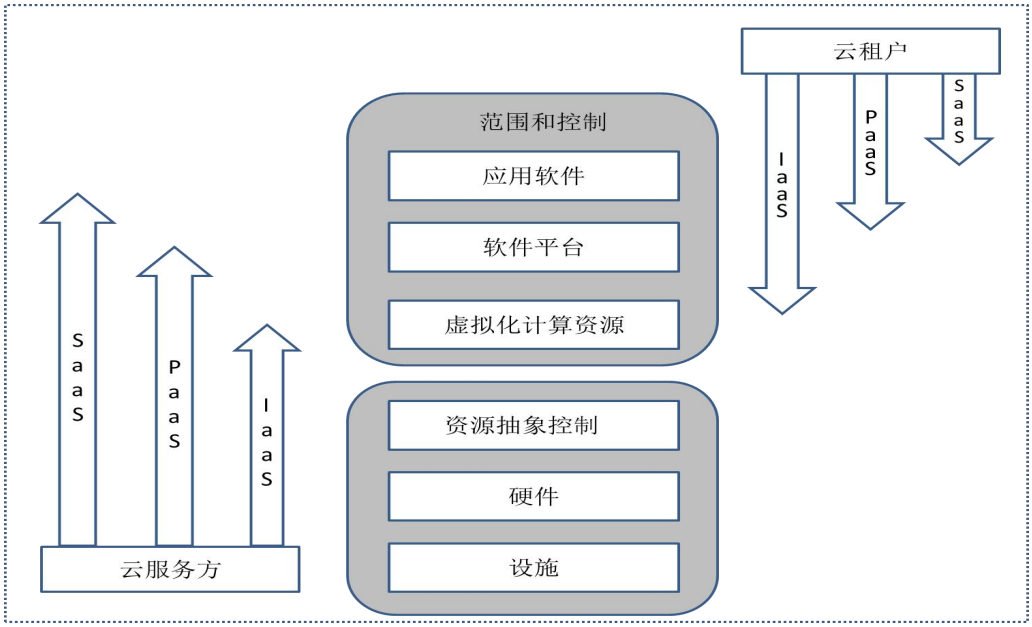


图 11- 3 云服务模式与资源控制范围的关系

云计算保护环境是云服务方的云计算平台，及云租户在云计算平台之上部署的软件及相关组件的集合。其中，云计算平台的等级保护定级和按照等级的保护工作由云服务方负责。对于大型云计算平台，可以将云计算基础设施平台及辅助支撑系统划分为不同的等级对象，各自独立定级。如果云租户在云计算平台上部署的软件及相关组件可以构成等级保护定级对象的，则一般称为云租户信息系统，针对其具体定级和按等级开展的保护工作，由云租户负责。

云服务方的云计算平台可以承载多个不同等级的云租户信息系统，云计算平台的安全保护等级应不低于其承载云租户信息系统的最高安全保护等级。

- **云平台安全设计**

（1） 身份鉴别

身份鉴别通用安全设计要求包括以下方面：

1）应支持用户标识和用户鉴别；

2）在每一个用户注册到云计算平台、服务或云上租户业务系统时，应分别采用用户名和用户标识符标识用户身份，并确保在云计算服务或云上租户业务系统整个生存周期用户标识的唯一性；

3）在每次用户登录云计算平台、服务或云上租户业务系统时，应采用受控的口令或具有相应安全强度的其他机制进行用户身份鉴别；

4）应对鉴别信息进行保密性和完整性保护；

5）用户身份鉴别信息应有复杂度要求，口令应定期更换；

6）具有登录失败处理功能，应配置并启用结束会话、限制非法登录次数和当登录连接超时自动退出等相关措施。

服务层身份鉴别安全设计要求包括以下方面：

应支持注册到云计算服务的云租户建立主子账号，并采用用户名和用户标识符标识主子账号用户身份，并确保在云计算服务整个生存周期用户标识的唯一性。

（2） 访问控制

访问控制通用安全设计要求包括以下方面：

1）应对登录的用户分配账号和权限，仅授予用户使用或管理所需的最小权限；

2）应重命名默认帐户或修改这些帐户的默认口令；

3）应及时删除或停用多余的、过期的帐户，避免共享帐户的存在；

4）应根据管理用户的角色建立不同账户并分配权限，仅授予管理用户所需的最小权限，实现管理用户的权限分离；

5）应由授权主体配置访问控制策略，访问控制策略规定主体对客体的访问规则；

6）访问控制策略主体的粒度为用户级，客体的粒度为文件或数据库表级；访问操作包括对客体的创建、读、写、修改和删除等。

服务层访问控制安全设计要求包括以下方面：

应支持建立云租户账号体系，实现主体与对虚拟机、云数据库、云网络、云存储等客体的访问授权。

（3） 安全审计

安全审计通用安全设计要求包括以下方面：

1）应提供安全审计功能，审计覆盖到每个用户，对重要的用户行为和重要安全事件进行审计；

2）审计记录应包括事件的日期和时间、用户、事件类型、事件是否成功及其他与审计相关的信息；

3）应提供审计记录查询、分类、分析和存储保护。

服务层安全审计安全设计要求包括以下方面：

1）应支持租户收集和查看与本租户资源相关的审计信息；

2）应保证云服务方对云租户系统和数据的访问操作可被云租户审计。

（4） 数据完整性保护

资源层数据完整性保护安全设计要求包括以下方面：

应确保虚拟机迁移过程中，重要数据的完整性，并在检测到完整性受到破坏时采取必要的恢复措施。

服务层数据完整性保护安全设计要求包括以下方面：

应采用常规校验机制，检验存储的用户数据的完整性，以发现其完整性是否被破坏；

应采用校验码技术保证重要数据在传输过程中的完整性。

（5） 数据备份与恢复

数据备份与恢复通用安全设计要求包括以下方面：

1）应提供重要数据处理系统的热冗余，保证系统的高可用性；

2）应支持异地备份功能，利用通信网络将重要数据定时批量传送至备用场地。

资源层数据备份与恢复安全设计要求包括以下方面：

云平台应采取冗余架构或分布式架构设计，保证高可用性。

服务层数据备份与恢复安全设计要求包括以下方面：

1）云租户应在本地保存其业务数据的备份；

2）应提供查询云租户数据及备份存储位置的方式；

3）应提供数据多副本存储方式，并保证至少存放在不同机架上；

4）应提供通用的接口确保云租户可以将业务系统及数据迁移到其他云计算平台和本地系统，保证可移植性。

（6） 虚拟化安全

资源层虚拟化安全设计要求包括以下方面：

1）应实现虚拟机之间的 CPU、内存和存储空间安全隔离，确保某个虚拟机发生异常后不影响其他虚拟机；

2）应确保分配给虚拟机的 CPU、内存和存储空间不能被其他虚拟机和物理宿主机访问，并能在检测到资源隔离失效后进行告警；

3）应在虚拟机监控器的控制下实现物理资源和虚拟资源的管理调度和资源分配；

4）应限制虚拟机对宿主机物理资源的直接访问，应能检测到虚拟机对宿主机物理资源的异常访问；

5）应保证虚拟机只能接收到目的地址包括自己地址的报文。

服务层虚拟化安全设计要求包括以下方面：

应限制云租户不超过范围使用服务资源，云租户只能访问和操作为其分配的服务资源。

（7） 入侵防范

入侵防范通用安全设计要求包括以下方面：

1）应遵循最小安装的原则，仅安装需要的组件和应用程序；

2）应关闭不需要的系统服务、默认共享和高危端口；

3）应通过设定终端接入方式或网络地址范围对通过网络进行管理的管理终

端进行限制；

4）应能发现可能存在的漏洞，并在经过充分测试评估后，及时修补漏洞；

5）应能够检测到对重要节点进行入侵的行为，并在发生严重入侵事件时提供报警。

资源层入侵防范安全设计要求包括以下方面：

应能够对虚拟化管理平台存在的漏洞进行检测，及时检测发现并修复虚拟化共享技术带来的技术漏洞。

服务层入侵防范安全设计要求包括以下方面：

1）应支持对暴力破解、撞库攻击等鉴别信息窃取行为进行检测和告警；

2）应提供对 Web 漏洞的安全检测和告警。

（8） 恶意代码防范

资源层恶意代码防范安全设计要求包括以下方面：

物理机和宿主机应安装经过安全加固的操作系统或进行主机恶意代码防范。

服务层恶意代码防范安全设计要求包括以下方面：

1）虚拟机应安装经过安全加固的操作系统或进行主机恶意代码防范，防止恶意代码在虚拟机间蔓延；

2）应支持对 Web 应用恶意代码检测和防护的能力。

（9） 软件容错

服务层软件容错安全设计要求包括以下方面：

1）应用系统及相关软件组件应提供数据有效性检验功能，保证通过人机接口输入或通过通信接口输入的内容符合云平台系统设定要求；

2）应提供应用系统及相关软件的故障保护功能，当发生故障时，应能够保护当前所有状态，保证系统能够进行恢复。

（10） 客体安全重用

客体安全重用通用安全设计要求包括以下方面：

应采用技术措施保证内存、磁盘等客体资源重新分配前，对其原使用者的鉴别信息进行清除，以确保信息不被泄露。

服务层客体安全重用安全设计要求包括以下方面：

应保证鉴别信息所在的存储空间被释放或重新分配前得到完全清除。

（11） 接口安全

服务层接口安全设计要求包括以下方面：

1）对外提供服务的 Web 接口应采用安全的数据传输协议来提高传输数据的安全性；

2）对外提供服务的 API 接口应在调用前进行用户鉴别和鉴权，应确保接口访问控制的有效性。

（12） 镜像和快照安全

服务层镜像和快照安全设计要求包括以下方面：

1）应提供对虚拟机镜像和快照文件的完整性保护；

2）采取加密或其他技术手段防止虚拟机镜像、快照中可能存在的敏感资源被非授权访问；

3）针对重要业务系统提供加固的操作系统镜像或支持对操作系统镜像进行自加固。

（13） 个人信息保护

服务层个人信息保护安全设计要求包括以下方面：

1）应仅采集和保存业务需要的用户个人信息；

2）应禁止未授权访问和使用用户个人信息。

3.7.2.4 网络安全

本项目部署在全市统一的云平台上，依据《信息安全技术信息系统等级保护安全建设技术方案设计要求》，《GB/T 22239-2008 信息安全技术信息系统安全等级保护基本要求》，本项目通信网络安全需求如下。

通信网络安全设备需求

序号	要求项	具体要求	安全需求
1	网络和通信安全	网络架构	高性能核心交换机
2		通信传输	边界防火墙、入侵防御、应用防火墙、审计设备、VPN 设备、漏洞扫描系统
3		边界防护	
4		访问控制	
5		入侵防范	

6		安全审计	

- **边界防火墙**

在构建安全的网络环境的过程中，防火墙作为第一道安全防线，正受到越来越多用户的关注。通常在购买网络安全设备时，总是把防火墙放在首位。目前，防火墙已经成为世界上用得最多的网络安全产品之一。

防火墙是一种将内部网和公众网如 Internet 分开的方法。它能限制被保护的网络与互联网络之间，或者与其他网络之间进行的信息存取、传递操作。防火墙可以作为不同网络或网络安全域之间信息的出入口，能根据组织的安全策略控制出入网络的信息流，且本身具有较强的抗攻击能力。它是提供信息安全服务，实现网络和信息安全的基础设施。在逻辑上，防火墙是一个分离器，一个限制器，也是一个分析器，有效地监控了内部网和 Internet 之间的任何活动，保证了内部网络的安全。防火墙的安全技术包括包过滤技术、网络地址转换——NAT 技术、代理技术等。

包过滤技术（Packet Filter）是防火墙为系统提供安全保障的主要技术，它通过设备对进出网络的数据流进行有选择的控制与操作。包过滤操作通常在选择路由的同时对数据包进行过滤（通常是对从互联网络到内部网络的包进行过滤）。用户可以设定一系列的规则，指定允许哪些类型的数据包可以流入或流出内部网络；哪些类型的数据包的传输应该被拦截。包过滤规则以 IP 包信息为基础，对 IP 包的源地址、IP 包的目的地址、封装协议（TCP/UDP/ICMP/IP Tunnel）、端口号等进行筛选。包过滤这个操作可以在路由器上进行，也可以在网桥，甚至在一个单独的主机上进行。

传统的包过滤只是与规则表进行匹配。防火墙的 IP 包过滤，主要是根据一个有固定排序的规则链过滤，其中的每个规则都包含着 IP 地址、端口、传输方向、分包、协议等多项内容。同时，一般防火墙的包过滤的过滤规则是在启动时配置好的，只有系统管理员才可以修改，是静态存在的，称为静态规则。也可以采用基于连接状态的检查，将属于同一连接的所有包作为一个整体的数据流看待，通过规则表与连接状态表的共同配合进行检查。

网络地址转换是一种用于把内部 IP 地址转换成临时的、外部的、注册的 IP 地址的标准。它允许具有私有 IP 地址的内部网络访问因特网。它还意味着用户

不需要为其网络中每台机器取得注册的 IP 地址。

在内部网络通过安全网卡访问外部网络时，将产生一个映射记录。系统将外出的源地址和源端口映射为一个伪装的地址和端口，让这个伪装的地址和端口通过非安全网卡与外部网络连接，这样对外就隐藏了真实的内部网络地址。在外部网络通过非安全网卡访问内部网络时，它并不知道内部网络的连接情况，而只是通过一个开放的 IP 地址和端口来请求访问。OLM 防火墙根据预先定义好的映射规则来判断这个访问是否安全。当符合规则时，防火墙认为访问是安全的，可以接受访问请求，也可以将连接请求映射到不同的内部计算机中。当不符合规则时，防火墙认为该访问是不安全的，不能被接受，防火墙将屏蔽外部的连接请求。网络地址转换的过程对于用户来说是透明的，不需要用户进行设置，用户只要进行常规操作即可。

应用代理或代理服务器（Application Level Proxy or Proxy Server）是代理内部网络用户与外部网络服务器进行信息交换的程序。它将内部用户的请求确认后送达外部服务器，同时将外部服务器的响应再回送给用户。这种技术被用于在 Web 服务器上高速缓存信息，并且扮演 Web 客户和 Web 服务器之间的中介角色。它主要保存因特网上那些最常用和最近访问过的内容，为用户提供更快的访问速度，并且提高网络安全性。这项技术对 ISP 很常见，特别是如果它到因特网的连接速度很慢的话。在 Web 上，代理首先试图在本地寻找数据，如果没有，再到远程服务器上去查找。也可以通过建立代理服务器来允许在防火墙后面直接访问因特网。代理在服务器上打开一个套接字，并允许通过这个套接字与因特网通信。

如果防火墙系统本身被攻击者突破或迂回，对内部系统来说它就毫无意义。因此，保障防火墙自身的安全是实现系统安全的前提。一个防火墙要抵御黑客的攻击必须具有严密的体系结构和安全的网络结构。

- **网络入侵检测**

网络入侵是威胁计算机或网络的安全机制（包括机密性、完整性、可用性）的行为。入侵可能是来自互联网的攻击者对系统的非法访问，也可能是系统的授权用户对未授权的内容进行的非法访问。入侵检测就是对发生在计算机系统或者网络上的事件进行监视、分析是否出现入侵的过程。入侵检测系统（英文称 IDS：Intrusion Detection System）是自动进行入侵检测的监视和分析过程的硬件或

软件产品。入侵监测系统处于防火墙之后对网络活动进行实时监测。许多情况下，由于可以记录和禁止网络活动，所以入侵监测系统是防火墙的延续。防火墙看起来好像可以满足系统管理员的一切需求。然而，随着基于内部人员的攻击行为和产品自身问题的增多，IDS 由于能够在防火墙内部监测非法的活动正变得越来越必要。新的技术同样给防火墙带来了严重的威胁，这些破坏行为也是防火墙无法抵御的。IDS 已经成为网络安全防护系统的三大重要组成部分之一。

- **漏洞扫描**

安全扫描技术是一类重要的网络安全技术。安全扫描技术与防火墙、入侵检测系统互相配合，能够有效提高网络的安全性。通过对网络的扫描，网络管理员可以了解网络的安全配置和运行的应用服务，及时发现安全漏洞，客观评估网络风险等级。网络管理员可以根据扫描的结果更正网络安全漏洞和系统中的错误配置，在黑客攻击前进行防范。如果说防火墙和网络监控系统是被动的防御手段，那么安全扫描就是一种主动的防范措施，可以有效避免黑客攻击行为，做到防患于未然。

安全扫描技术主要分为两类：主机安全扫描技术和网络安全扫描技术。网络安全扫描技术主要针对系统中不合适的设置脆弱的口令，以及针对其它同安全规则抵触的对象进行检查等；而主机安全扫描技术则是通过执行一些脚本文件模拟对系统进行攻击的行为并记录系统的反应，从而发现其中的漏洞。

- **传输链路安全**

由于移动端应用将是目前信息技术发展的趋势，为保障传输链路安全，可通过 SSL VPN 解决传输通道安全。

3.7.3 应用层安全

应用系统的安全和其自身的设计和实现技术密切相关，其存在的漏洞也会给系统的安全带来严重的隐患，因此通过应用安全技术和应用系统相结合是防护应用层安全的重要手段。

3.7.3.1 身份认证

（1） 用户管理

为系统提供统一管理用户的界面。用户管理集中统一后，每个用户账号只申

请一次，这样可以减少用户身份的副本，增加安全性，用户数据只维护一次即可到处使用。

用户管理除了提供单个录入的方式外，还提供方便的批量导入的方式，批量导入的数据经校验后的直接进入到系统中。

为了避免密码泄露，系统在进行密码存储和传输时，一律采用不可逆加密的方式。为了防止对简单密码的猜测，初始密码随机生成，随机密码为大写字母、数字和小写字母的随机组合。

（2） 统一身份认证

身份认证采用中央认证服务的方式来完成，每个系统不再需要自己的身份认证，实际的身份认证都自动转发到中央认证服务，由中央认证服务来完成。

（3） 单点登录

用户经统一身份认证之后，如果需要进入其它系统，不需要再次登录认证，从而为用户提供多应用系统方便的单点登录功能，实现“一点登录、多点漫游”的功能。

3.7.3.2 分级管理

通过建立统一用户授权管理系统，为系统的各应用子系统提供通用的、支撑性的用户管理，实现可靠访问控制，提供用户管理的高效性，降低后台管理人员的维护工作量，并通过共享的用户信息服务，将各应用系统有机的整合在一起，实现互联互通，消除“信息孤岛“。

统一用户授权管理采用基于角色的访问控制（RBAC）授权管理模型，通过角色信息与应用系统内部权限信息的映射，形成“用户—角色—权限“三元对应关系，对各类用户进行严格的访问控制，以确保应用系统不被非法或越权访问，防止信息泄漏。

3.7.3.3 日志审计

对系统的操作记录提供事后审计和日志统计，保证系统操作的可追溯性和安全性。

系统内提供了详细的日志统计功能，对所有用户角色在各功能模块的操作都

进行了记录，形成详细的日志信息，一旦出现任何问题，可通过日志查找根源。

3.7.3.4 软件容错

软件容错的主要目的是提供足够的冗余信息和算法程序，使系统在实际运行时能够及时发现程序设计错误，采取补救措施，以提高软件可靠性，保证整个计算机系统的正常运行。系统应具有较强的容错性，对于用户的错误操作，应给予友好的提示；对于系统出现的异常，应向用户解释原因，提示用户如何处理；对于已经发生错误或异常，系统应尽可能恢复到原来操作状态。在系统软件设计时充分考虑软件容错设计，包括：提供数据有效性检验功能，保证通过人机接口输入或通过通信接口输入的数据格式或长度符合系统设定要求；具备自保护功能，在故障发生时，应用系统应能够自动保存当前所有状态，确保系统能够进行恢复。

3.7.4 数据层安全

3.7.4.1 数据库审计

数据库审计与风险控制系统主要用于监视并记录对数据库服务器的各类操作行为，通过对网络数据的分析，实时地、智能地解析对数据库服务器的各种操作，并记入审计数据库中以便日后进行查询、分析、过滤，实现对目标数据库系统的用户操作的监控和审计。

数据库审计与风险控制系统可以监控和审计用户对数据库中的数据库表 、视图、序列、包、存储过程、函数、库、索引、同义词、快照、触发器等的创建、修改和删除等，分析的内容可以精确到 SQL 操作语句一级。还可以根据设置的规则，智能的判断出违规操作数据库的行为，并对违规行为进行记录、报警。由于数据库安全审计系统是以网络旁路的方式工作于数据库主机所在的网络，因此它可以在根本不改变数据库系统的任何设置的情况下对数据库的操作实现跟踪记录、定位，实现数据库的在线监控，在不影响数据库系统自身性能的前提下，实现对数据库的在线监控和保护，及时地发现网络上针对数据库的违规操作行为并进行记录、报警和实时阻断，有效地弥补现有应用业务系统在数据库安全使用上的不足，为数据库系统的安全运行提供了有力保障。

（1） 数据库审计与风险控制系统主要功能：

- 实时监测并智能地分析、还原各种数据库操作过程；
- 根据规则设定及时阻断违规操作，保护重要的数据库表和视图；
- 实现对数据库系统漏洞、登录账号、登录工具和数据操作过程的跟踪，发现对数据库系统的异常使用；
- 支持对登录用户、数据库表名、字段名及关键字等内容进行多种条件组合的规则设定，形成灵活的审计策略；
- 提供包括记录、报警、中断和向网管系统报警等多种响应措施；
- 具备强大的查询统计功能，可生成专业化的报表。

（2） 数据库审计与风险控制系统主要特点

- 采用旁路技术，不影响被保护数据库的性能；
- 使用简单，不需要对被保护数据库进行任何设置；
- 支持 SQL-92 标准，适用面广，可以支持 Oracle、MS SQL Server、Sybase、Informix 等多类数据库；
- 审计精细度高，可审计并还原 SQL 操作语句；
- 采用分布式监控与集中式管理的结构，易于扩展；
- 完备的"三权分立"管理体系，适应对敏感内容审计的管理要求。

3.7.4.2 数据备份

数据是系统的血液，任何情况下，保障数据的安全对于系统保持健康运行都具有决定性的意义。完善的数据备份机制，是保障数据安全的重要手段之一。

本项目的数据主要存储于电子政务云，数据的远程灾备由政务云进行保障。

数据的日常备份方式有完全备份、增量备份、差量备份、完全备份和增量备份组合以及完全备份和差量备份组合等几种。

备份机制：考虑到住建局日常业务的特点，每天产生变化的数据量不会很大，在需要数据恢复时，要求恢复时间尽可能短，因此，建议系统采用完全备份和增量备份组合的机制。每周一个备份循环。周六或周日进行完全备份，其它工作日采用增量备份。具体规划如下表所示：

备份规划

周六	周日	周一	周二	周三	周四	周五

完全备份	增量备份	增量备份	增量备份	增量备份	增量备份

这种备份机制，轮巡方式简单明了，易于实施管理。系统可将自动备份时间设定在每日晚 10 点，通常此时开始备份已经不会影响正常工作。需要数据恢复时，只需要完全备份部分加上周一至周 X（X = 恢复日星期数 — 1）的增量备份即可。

为防止诸如地震、火灾、水灾及战争等不可抗拒的外来因素对数据备份介质的永久性损坏而带来的数据损失，备份数据的硬拷贝介质应该进行周期性的复制并异地存放，以最大限度地保障数据的安全性。

（1）数据恢复

无论是采用手工方式，还是通过计算机软件对数据库中的数据进行修改，都有可能导致数据错误的发生。当发生数据错误时，系统应该能够恢复。这就要求数据库管理系统具有如下功能：

- **自动恢复：**在数据出错时可把数据修复到修改前状态；
- **自动备份：**数据库修改后，原有的数据应作备份；
- **历史数据：**当数据库中的数据被修改后，原有的数据要保留入历史库中，以备数据回溯和查询使用。

（2）灾难恢复

灾难恢复措施在整个备份制度中占有相当重要的地位。因为它关系到系统在经历灾难后能否迅速恢复。灾难恢复措施包括：灾难预防制度、灾难演习制度及灾难恢复。

1） 灾难预防制度

为了预防灾难的发生，需要做灾难恢复备份。关于灾难预防制度，通常应该考虑：

- 灾难恢复备份应是完全备份。
- 在系统发生重大变化后，建议重新生成灾难恢复盘，并进行灾难恢复备份。如安装了新的数据库系统，或安装了新硬件等。

2） 灾难演习制度

要能够保证灾难恢复的可靠性，光进行备份是不够的，还要进行灾难演练。每过一段时间，应进行一次灾难演习。可以利用淘汰的机器或多余的硬盘进行灾难模拟，以熟练灾难恢复的操作过程，并检验所生成的灾难恢复软盘和灾难恢复

备份是否可靠。

3） 灾难恢复

灾难恢复的步骤非常简单：准备好最近一次的灾难恢复备份磁带，连接好磁带机，装入磁带，打开计算机电源，灾难恢复过程就开始了。根据系统提示进行下去，就可以将系统恢复到进行灾难恢复备份时的状态。再利用其他备份数据，就可以将服务器和其他计算机恢复到最近的状态。

3.7.5 安全管理

“达州市城市体检信息平台”安全管理须遵循《信息系统安全规范》。系统上线前，须通过软件漏洞安全扫描，达到相应安全等级要求。

3.7.5.1 安全管理制度

在信息安全中，最活跃的因素是人，对人的管理包括法律、法规与政策的约束、安全指南的帮助、安全意识的提高、安全技能的培训、人力资源管理措施，这些功能的实现都是以完备的安全管理政策和制度为前提。这里所说的安全管理制度包括信息安全工作的总体方针、策略、规范各种安全管理活动的管理制度以及管理人员或操作人员日常操作的操作规程。

安全管理制度主要包括：管理制度、制定和发布、评审和修订。要求形成信息安全管理制度体系，对管理制度的制定要求和发布过程进一步严格和规范。对安全制度的评审和修订要求领导小组负责。

3.7.5.2 安全管理机构

平台安全管理需要建立一个健全、务实、有效、统一指挥、统一步调的完善的安全管理机构，明确机构成员的安全职责，这是信息安全管理得以实施、推广的基础。在单位的内部结构上必须建立一整套从单位最高管理层到执行管理层以及业务运营层的管理结构来约束和保证各项安全管理措施的执行。其主要工作内容包括对机构内重要的信息安全工作进行授权和审批、内部相关业务部门和安全管理部门之间的沟通协调以及与机构外部各类单位的合作、定期对系统的安全措施落实情况进行检查，以发现问题进行改进。

安全管理机构主要包括：岗位设置、人员配备、授权和审批、沟通和合作以及审核和检查等。对于岗位设置，不仅要求设置信息安全的职能部门，而且机构上层应有一定的领导小组全面负责机构的信息安全全局工作。授权审批方面加强了授权流程控制以及阶段性审查。沟通与合作方面加强了与外部组织的沟通和合作，并聘用安全顾问。同时对审核和检查工作进一步规范。

3.7.5.3 人员安全管理

很多重要的信息系统安全问题都涉及到用户、设计人员、实施人员以及管理人员。如果这些与人员有关的安全问题没有得到很好的解决，任何一个信息系统都不可能达到真正的安全。只有对人员进行了正确完善的管理，才有可能降低人为错误、盗窃、诈骗和误用设备的风险，从而减小了信息系统遭受人员错误造成损失的概率。

对人员安全的管理，主要涉及两方面：对内部人员的安全管理和对外部人员的安全管理。具体包括：人员录用、人员离岗、人员考核、安全意识教育和培训、外部人员访问管理等。增强对关键岗位人员的录用、离岗和考核要求，对人员的培训教育更具有针对性，外部人员访问要求更具体。

3.7.5.4 系统建设管理

根据基本要求制定系统建设管理制度，包括：系统定级、安全方案设计、产品采购和使用、自行软件开发、外包软件开发、工程实施、测试验收、系统交付、系统备案、等级评测、安全服务商选择等方面。从工程实施的前、中、后三个方面，从初始定级设计到验收评测完整的工程周期角度进行系统建设管理。

3.7.5.5 安全管理措施实现

制定安全检查制度，明确检查的内容、方式、要求等，检查各项制度、措施的落实情况，并不断完善。定期对信息系统安全状况进行自查，第三级信息系统每年自查一次。经自查，信息系统安全状况未达到安全保护等级要求的，应当进一步开展整改。安全管理具体实现如下：

3.8 管理运维体系

要充分发挥信息化建设成果的效用，需要一套完整的管理运维体系。结合实际的业务发展规划与 IT 系统现状，搭建以服务为导向的综合监控管理平台，确保 IT 部门能够根据业务部门提出的业务要求，协调 IT 资源，交付 IT 服务，使繁杂的 IT 管理变得标准有序，从被动式服务转向主动式服务，有效提升 IT 资产运作效率，确保 IT 建设能够充分适应组织战略发展的能力要求。

管理运维体系包括一致的决策体系、统一的监控平台、基于最佳实践的运维流程建设，采用整合的服务台、整合的事件管理、自助服务、变更配置和发布、持续的服务改进、产品组合与财务管理等流程，使得各个部门承担相应的角色职能和责任，从而让 IT 运维服务实现从由事后处理向预防性维护的根本转变，变事后被动维修向主动临测，减少维护工作对业务开展的影响，实现预期成效，做出明智决策，平衡成本和服务质量，提高员工工作效率并节省开支，实现端到端的服务交付，最大限度地降低服务中断，遵从法规需求和认证要求。

为确保本项目在试运行及正式运行阶段能够稳定、安全、有效的运行，必须设计项目的运行维护方式。由技术开发服务单位提供 7×24 小时的技术支持服务，对所有技术支持、服务请求、故障报修、技术咨询的工作分专业由专人负责，同时对所有问题实行记录、分派、跟踪、管理、分析和报告。

3.8.1 管理运维参考

基础信息平台的建设成果将集成多种硬件和网络设备、不同用途的软件系统、多种来源的数据库和多种 IT 技术、标准规范等，是一个规模庞大，由异构硬件、异构网络、异构数据、异构应用等多框架、多体系形成的复杂系统。

对于这样一种庞杂系统的运营管理，已经不是单纯的资金、技术和人力投入问题，它需要有一整套完整的 IT 服务管理思想和管理体系来支撑。

ITIL 即 IT 基础架构库(Information Technology Infrastructure Library，ITIL，信息技术基础架构库)，主要适用于 IT 服务管理（ITSM)。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。ITIL 已经在全球 IT 服务管理领域得到了广泛的认同和支持，无论各公司的理念和解决方案有多大差异，但目标都是一致的：把 IT 与业务相结合，以业务为核心搭建和管理

IT 系统。

基于 ITIL 的成功实践，本项目的 IT 运营维护框架可参考 ITIL 标准来制定，整个 IT 运维框架如下图所示：

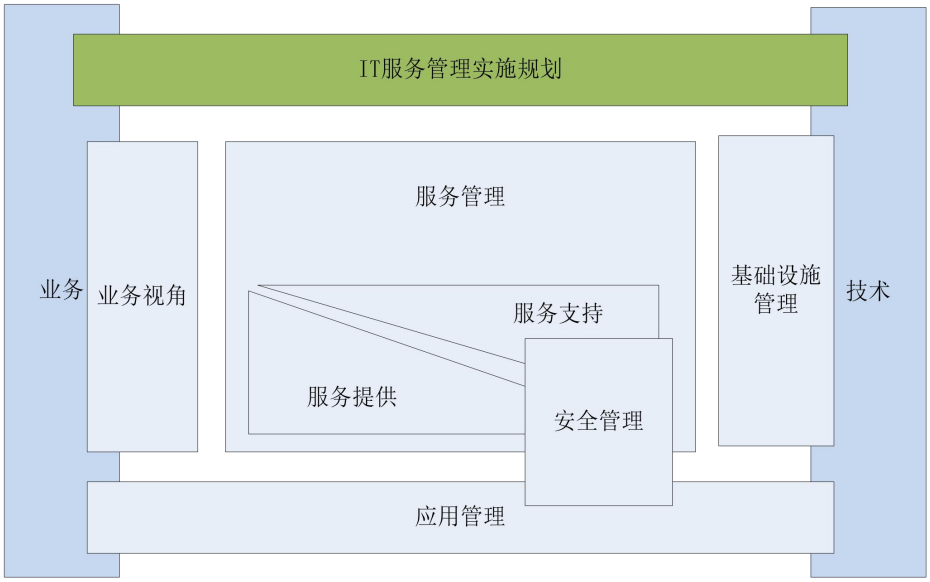


图 12- 1 ITIL 整体框架

3.8.1.1 服务提供

服务提供由服务级别管理、IT 服务财务管理、IT 服务持续性管理、可用性管理和能力管理 5 个流程组成。

（1）服务级别管理

服务级别管理流程主要定义、协商、订约、检测和评审信息中心提供给各业务处室的服务质量水准。这些服务的质量水准记录在服务级别协议中，该协议规定了服务双方各自的责任、权利和义务，是 IT 服务成功运营的前提。

服务级别管理流程的任务是确保服务级别协议是根据业务人员的业务需求而不是服务提供者的技术能力确定的，才能保证服务级别协议得到有效执行，并在服务双方出现争议时提供有效的证据和解决争议的指导规则。

（2）服务财务管理

服务财务管理是指负责预算和核算信息中心提供 IT 服务所需的成本。该流程包括 IT 投资预算、IT 服务成本核算和服务计费 3 个子流程，其目标是通过量化服务成本，减少成本超支的风险，减少不必要的浪费，合理引导业务人员的行为，保证所提供的服务符合成果效益的原则。

服务财务管理流程产生的预算和核算信息为服务级别管理、能力管理、服务持续性管理和变更管理等管理流程提供了决策依据。

（3）服务持续性管理

服务持续性管理是指确保发生灾难后有足够的技术、财务和管理资源来确保IT服务持续性的管理流程。它关注的焦点是：在发生服务故障后，仍然能够提供预定级别的IT服务，从而支持住建业务持续运营的能力。

（4）可用性管理

可用性管理是通过分析业务人员的可用性需求，据以优化和设计各类应用系统的可用性，确保以合理的成本满足不断增长的可用性需求。可用性管理是一个前瞻性的管理流程，它通过对业务可用性需求的定位，使得IT服务的设计建立在真实需求的基础上，避免采用过度的可用性级别，节约IT服务的运营成本。

（5）能力管理

能力管理是指在成本和业务需求的双重约束下，通过配置合理的服务能力，使IT资源发挥最大效能。

能力管理包括业务能力管理、服务能力管理和资源能力管理三个子流程。其中业务能力管理子流程主要关注当前及未来的业务需求，服务能力管理子流程主要关注当前IT服务的绩效是否能够支持正常的业务运营，而资源能力管理子流程主要关注IT服务的技术基础，确保各应用系统能发挥最大的效能。

3.8.1.2 服务支持

服务支持由事故管理、问题管理、配置管理、变更管理和发布管理五个流程及服务台组成。

（1）服务台（Service Desk）

服务台是一项管理职能而不是一个管理流程。它作为IT服务提供方（即信息中心）与IT服务客户（即局业务人员）之间的统一联系点。当业务人员提出服务请求、报告事故或问题时，负责记录这些请求、事故和问题，尽量解决它们。在不能解决时，转交给相应的应用开发商，并负责协调开发商和业务人员之间的交互。另一方面，作为应用系统开发商和业务人员之间的桥梁，了解应用系统的使用情况，把开发商的处理进展及时通报给业务人员。此外，服务台还为其他管

理流程如变更管理、配置管理、发布管理、服务级别管理及 IT 服务持续性管理提供了接口。

（2）事故管理

在出现事故时，应尽可能快地恢复系统的正常运转，避免造成业务中断，这是事故管理的主要目的。为了实现这个目的，事故管理流程必须利用各种信息资源支持业务的顺畅运转，维护有效的事故记录，形成统一的事故报告机制和报告方法。

（3）问题管理

问题是指导致事故的潜在原因。

问题管理就是尽量减少应用系统、数据库、人为错误和外部事件等的缺陷或过失对业务造成的影响，防止它们重复发生。问题管理与事故管理的区别是：后者是尽可能快地恢复服务，而前者的主要目的是找出事故产生的根本原因。如有必要，问题管理可能要求中断服务。

如果找到事故产生的原因，应立即将其明确标示出来，提出变更请求（RFC），以消除事故隐患。

（4）配置管理

配置管理是识别和确认应用系统的配置项，记录和报告配置项状态和变更请求，检验配置项的正确性和完整性等活动。其主要目的是支持其他服务管理流程，特别是变更管理和发布管理。为此，配置管理需要计量所有 IT 资产，为其他流程提供准确的信息，为事故管理、问题管理、变更管理和发布管理提供基础。

（5）变更管理

变更是对已批准构建或实施的、已在维护的或作为基准的硬件、网络、软件、应用、环境、系统及相关文档所做的增加、修改或删除。变更管理的目的是使用标准方法和规程来快速有效地处理所有变更，以减少任何有关事故对服务的影响。

与前面提到的服务台、事故管理和问题相比，变更管理追求的是“标本兼治”，它不仅要找到解决事故或问题的根本方法，更要变更 IT 基础架构，以防止此类事故和问题的再次发生。

（6）发布管理

发布（版本）是指一组经过测试后导入实际运营环境的配置项（工件）。发布管理的目的是为了保证发布的成功，主要应用于大型的或关键硬件、主要软件、

数据库的变更。

3.8.1.3 基础设施管理

基础设施是提供 IT 服务的物质前提，也是 IT 服务管理的对象和基础。ICT 基础设施管理模块覆盖了识别业务需求、实施、部署以及支持和维护各类 IT 资产的过程。其目标是确保提供一个稳定可靠的信息化基础设施，以支撑各类业务的正常运转。

3.8.1.4 业务视角

ITIL 所强调的思想是应该从业务而不是技术的角度理解 IT 服务需求。业务视角模块用于帮助业务人员深入了解 IT 资产支持业务流程的能力，以及 IT 服务管理在提供端到端服务过程中的作用，协调好业务人员与服务提供方（即信息中心）之间的关系。

3.8.1.5 IT 服务管理实施规划

作为服务提供方，信息中心如何根据自己的需求和实际情况，实施 ITIL 中的某一个或多个流程，或者某个流程中的一部分，是一件富有挑战性的工作。IT 服务管理实施规划即用于解决这个问题。它为信息中心确立服务目标，分析服务现状，确定合理的服务质量并进行差距分析，判断服务活动的优先级，以及服务流程的评审等，提供了全面指导。

3.8.1.6 应用管理

IT 服务管理包括对应用系统的支持、维护和运营。

从业务人员的角度说，他们关注的不是应用系统的开发、测试和部署过程，而是这个过程最终的结果，即它可以实现哪些功能，实现的功能是否满足业务需求以及是否可靠。

因此，IT 服务管理的职能应该合理地延伸，介入信息化的过程体系，即应用系统的开发、测试和部署过程。应用管理就是协调信息化的服务体系和过程体系，以使它们一致地服务于业务应用。

3.8.1.7 安全管理

安全管理的目标是保护 IT 基础设施，使其避免未经授权的使用。安全管理模块为如何确定安全需求、制定安全政策和策略，以及处理安全事故提供了全面的指导。

3.8.2 管理运维制度

为了保障基础信息平台的稳定、安全运行，需根据需要制定对应的管理制度规范。

管理运维流程：针对需求变更、新需求提出、版本发布更新等需制定相应的执行流程。例如：针对需求变更以及新需求需要业务组确认，并通过管理组认可后方可进入分析、设计及开发阶段；版本只有通过严格测试并通过安全扫描后方可执行版本更新，并且更新维护等都需有严格的日志记录。

管理运维审计及其他管理办法：管理组可定期进行平台的日志审核，以期发现异常操作，避免影响基础信息平台的稳定运行以及数据的准确性，发现问题及时向领导组汇报，并采取管理措施加以纠正等，同时做好备案；同时也可以把具体的审计结果作为业务组和开发组的绩效考核具体参考指标之一。其他的管理办法包括权限管理办法、运营维护手册、数据管理使用办法、信息安全管理办法等制度类文本。

3.8.2.1 管理运维流程

平台的管理运维流程主要是针对需求变更、新需求提出及版本更新发布等为主的流程。

需求变更、新需求提出：各一线办公人员提出的需求，首先由业务组进行登记并初步判定是否合理，并给出接收或变更或不接收的原因，并由业务组汇报管理组，由管理组最终确定，业务组根据最终确定结果负责答复需求提出者，针对确定要修改的需求，交由开发组，并列出计划开发执行，最后成果由业务组验证通过后，正式发布版本。

版本更新发布：开发完成的版本，需首先部署到测试环境中，并由测试工具专业测试，包括功能测试、压测、性能测试、安全扫描测试等等，并出具测试报

告，只有通过测试的版本才允许正式发布到生产环境中。发布版本需保留 1-2 天的缓冲时间，例如在周五晚上发布版本，以便遇到紧急情况时周末可以挽救。同时，正式环境的版本，在更新前务必备份好历史版本，同时数据业务做好相应的备份工作。

3.8.2.2 管理运维审计

定期不定期由监管和业务小组审查日志，并记录存在的可疑日志，针对可疑日志需结合真实业务数据逐一核实，并汇报管理小组。最终确定后，需记录并提出修改措施，以协助操作人员避免类似可疑操作。

在审计的同时，也需针对平台的稳定运行日志做审计，及早查出可能存在的问题隐患，例如性能瓶颈分析、硬盘空间是否足够、是否有安全漏洞隐患等等。一旦发现隐患，需及时汇报，并采取措施加以预防。

3.8.2.3 配套管理办法

其他的管理办法包括权限管理办法、运营维护手册、数据管理使用办法、信息安全管理办法等制度类文本。

例如：运营维护手册，需明确各服务器的启动、停止操作顺序等，以避免由于低级错误而导致平台崩溃。数据管理使用办法需明确哪类数据可以共享使用，哪类数据必需经领导小组许可后方可共享等。安全管理办法需明确数据的备份、系统灾备、权限授权等级、数据加密等等安全措施。

3.8.2.3.1 数据管理使用办法

通过数据管理使用办法强化“谁生产、谁负责”的数据责任原则，对数据的管理、汇交、更新、利用等环节明确职责、岗位、审查、追踪、考核机制，制定清晰规范的操作流程，设计行政手段加技术手段方式进行数据管理使用办法的落地。

3.8.2.3.2 信息安全管理办法

信息安全是指信息系统（包括硬件、软件、数据、人、物理环境及其基础设施）受到保护，不受偶然的或者恶意的原因而遭到破坏、更改、泄露，系统连续

可靠正常地运行，信息服务不中断，最终实现业务连续性。通过建立信息安全管理办法，确保局信息的保密性、真实性、完整性、授权拷贝。信息安全管理办法应在国家、省及市有关安全管理的要求下优先制定信息安全保障体系框架，从技术层面、管理制度层面进行安全管理内容的制定。

3.8.3 管理运维工具

在大量的管理运维工作中，除了人工手动管理运维外，有些工作需借助管理运维工具，以提高管理运维的水平、效率和准确性。

行业内应用比较多的，如网络管理及监控软件、服务器监控软件、应用性能管理软件、资产管理软件、客户服务支持管理软件、密码管理工具、日志分析审计工具、数据备份工具等等。有了工具的支持，可大大提高管理运维的效率。

3.8.4 管理运维内容

3.8.4.1 政务云基础设施运维

政务云基础设施运维由大数据局负责，对机房基础设施、支撑网络、云平台按照标准建设与运营，使得政务云平台达到高可用性、稳定性、安全性的目标：

- 电力配电系统，可用率在 99.99％以上；
- 基础支撑网络，可用率在 99.99％以上；
- 政务云系统，可用率在 99.99%以上。

政务云平台及机房基础设施维护体系按照统一监控、统一调度、分层运维的模式组建，由各运维支撑部门负责监控调度、基础维护、网络维护、系统维护。

（1） 监控调度

负责基础设施、支撑网络、政务云平台 7×24 小时实时监控，对各系统运行状态实时关注，对于异常情况进行判断处理或者将故障升级至相应维护支撑部门；负责客户使用问题投诉及咨询受理，并根据客户需求回复反馈信息。

（2） 基础维护

负责基础设施运行维护、业务生产、机房管理（人员及设备进出、施工、参观等），预防和排除各类隐患与风险，保障系统的正常运行。

（3） 网络维护

负责机房基础支撑网络的建设与运行维护，业务生产工作。

（4） 系统维护

负责云平台建设与部署，日常运行维护、资源配置与管理。

（5） 运维人员备案制度

政务云服务器机房运维人员的登记、离职或离岗时应及时备案。

机房内巡检包括以下内容：设备硬件灯告警情况，风机运转情况，机房环境卫生，机柜内异常情况，精密空调的显示屏显示情况，机房内其他情况。

云平台巡检包括以下内容：物理机和 VM 运行状况，数据库运行状况，存储运行状况等。

3.8.4.2 基础中间件运维

在系统上线后，为了保障业务系统基础环境的正常运转，需要确保基础环境运维工作的正常进行——主要针对数据库、中间件层面的运维。通过运维工作，确保数据库、中间件软件正常使用，对基础环境运行中的风险和异常情况，提前进行处理，防范于未然。

中间件涉及到移动应用中间件、信息门户中间件、业务流程管理（BPM）中间件、业务规则和事件管理中间件、移动管理中间件、LDAP 基础中间件、Web 应用服务器中间件、数据库中间件和报表应用中间件等，每个中间件需要运维的内容各自不同。

以下以数据库管理系统基础运维为例，说明运维工作内容。其工作内容如下：

（1） 检查分区工作，每周定时检查数据库分区表、分区索引建立情况，确保分区建立成功，并且对于未成功的分区、索引等进行处理，以确保系统稳定正常。

（2） 检查表空间工作，每周定时检查数据库表空间的使用情况，确保数据库磁盘空间及扩展正常。对于数据库表空间不足时，增加数据文件扩展表空间，确保数据安全。对于磁盘不足的情况，及时通知并驱动扩充磁盘空间。

（3） 日常备份检查工作，每日检查数据库备份情况，确保数据有效性和安全性。

（4） 日常数据库结构更新，每日定时处理流程审核通过后的数据库更新项，

满足最新业务的需求。

（5） 定期数据库健康巡检工作，定期对数据库进行警告日志及跟踪日志文件的检查，确保数据库的稳定运行。

（6） 检查重建索引重建工作，每月定期检查索引重建情况，确保所有索引重建成功。

（7） 数据库初始化安装工作，数据库初始化安装工作包括搭建数据库、调整数据库、清除测试数据投入到正式生产中。

（8） 初始化数据库性能调优总数，初始化的数据库性能优化工作，保证系统正常运行。

最常见的运维工作是随着系统使用量的增加，需要增加资源，进行系统扩容，并完成数据库和应用系统调优。

3.8.4.3 应用系统运维

为了进一步提高信息服务水平和质量，本项目建成的应用系统需要设立配套的运维管理流程。为了在项目建成后能够达到运维的效果，还需要统筹各方资源建立运维队伍，在业务和技术技能来保障各个应用系统的高效、可靠运行。

结合本次应用系统建设内容，运维方法主要从两个方面入手。即通过计划性运维和日常运维。

计划性运维，周期性的安排运维工作。把业务背后的支撑的体系做好完整的质量把关，逐步提高 IT 的弹性服务。工作内容覆盖系统、数据、版本、工具等方面。

日常运维是以应用系统的使用方在使用过程中对系统提出的疑问所需要的技术支持、系统问题的跟踪以及处理。通过日常运维，逐步提升对城市体检业务的服务水平和能力。进一步把握业务反映到应用系统中的需求变化，作为对需求变更的落脚点。

（1） 计划性运维

本项目建设主要所覆盖的用户主要分为局内、局外信息管理、业务管理和监督、数据更新管理。根据用户性质和系统的优先特性，通过设定固定周期的巡检时间计划，确定系统运维管理要点，能够起到以点带面的作用。运维管理要点可

以在具体项目建设时进一步优化。有针对性的对关键点实行轮询检查，定期跟进，及早发现、处理和改进系统问题。可以较大程度上杜绝系统的隐患。

（2） 日常运维

日常运维往往是紧急情况下提出的问题，需要在很短时间内予以解决。为了能够从源头解决问题，需要从需求层面入手，持续优化业务和系统功能，而不是被动应对逐渐沦为救火队伍。

由于各个系统复杂处理问题有时需要多人协作才能完成，需要规范日常运维管理要求。

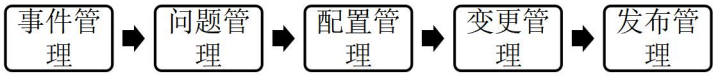


图 12- 3 日常运维管理内容

事件管理的目标是：在不影响业务的情况下，尽可能快速的恢复服务，从而保证最佳的效率和服务的可持续性。事件管理流程的建立包括事件分类，确定事件的优先级和建立事件的升级机制。

问题管理的目标是：调查系统运行的所有可用信息，包括事件数据库，来确定引起事件发生的真正潜在原因，一起提供的服务中可能存在的故障。

配置管理的目标是：针对问题处理的前后配置信息予以管理，使之能够恢复到某个状态下随时跟踪或者重现问题。对于过程中产生的文档变化、设计变化、参数调整等，都应当予以管理。

变更管理的目标是：以受控的方式，对需求、设计进行管理。确保所有变更得到评估、批准和实施，能够良好的支撑后续的持续运维。

发布管理的目标是：在实际运行环境中的更新版本或者成果，都予以过程管理。通过发布流程能够有效对正式环境下一个或者多个版本的管理。

在实际运维工作中，大多存在以下几类工作：

- **技术咨询**：业务人员使用系统，提供有针对性的疑问。需要运维队伍提供专业的使用意见，通过运维工作起到纠偏作用。
- **常见运维工作**：如处室、人员、权限等发生变化，需要尽快修改该授权让系统能够适应最新的实际情况。对于可以通过配置而调整的应用功能，在运维时应根据要求及时配置调整。以上运维工作一般可以通过系统工具配置完成。
- **业务细化引起的系统的修改**：随着业务精细化管理的要求，对于流程规

则、表单输入控制、报表打印格式、监控规则、管控要求等提出新的变化要求。本项目采用开放式 SOA 架构设计，通过流程可视化组合搭建、服务编排等多种方式，可以方便系统内部逻辑修改，使得系统资产能够在持续运维中发挥积极作用。但是，这些修改都需要在先提出修改方案，经过相关领批准后实施。通过规范的运维流程，从源头对需求、系统设计、实施做好持续把关。

以上运维工作，在每个步骤和阶段都需要遵循运维管理的要求，在规范下作业。能够做到对每个问题可以进行追溯，不出现问题的反复、需求的反复等管理效率低下的问题。

3.9 体检报告编制

3.9.1.1 准确采集数据

结合达州市职能部门统计数据、网络大数据、抽样调查数据等三方面信息，多途径、多渠道、全方位采集指标数据，形成“多源数据互为支撑验证、互为补充校核”的基础数据，建立“三管齐下、三位一体、相互校核”的数据采集机制，确保城市体检基础数据科学、真实、准确。

3.9.1.2 科学分析评估

统筹考虑法律法规、标准规范、政策文件和国家、省、市建设发展与管理要求，构建“汇交—采集—诊断”的问题诊断机制，及时发现城市发展特色优势与“短板”“弱项”，打通城市体检的反馈渠道，为治理城市病打下坚实基础。

3.9.1.3 形成体检报告

结合达州市实际，建立由基础指标与特色指标组成的城市体检指标体系，以官方统计数据为主要依据，对城市体检各项指标测算分析，对标各类指标标准值，查找短板及突出问题，提出对策建议，形成年度城市自体检报告。结合第三方评估和社会满意度调查结果，因地制宜提出下一年达州市城市治理措施建议及项目计划。

4. 项目组织机构和人员培训

4.1 领导和管理机构

达州市城市体检信息平台基于以下原则建设：

（1）“统筹共建”原则。坚持按照“共用统建、集约发展”的总体要求，共同推动项目集约化、标准化、规范化建设。重大事项提交领导小组会议研究决定。

（2）“集约建设”原则。项目建设充分考虑利用我市已建成的政务云和云灾备平台等基础性设施，同时做好现有信息系统的利旧和升级改造，全面整合至基础信息平台，避免重复投资、低效建设。

（3）“共享建设”原则。坚持与政务云系统一体化对接，做好与政务信息资源共享平台等综合性应用系统的衔接与数据共享工作。

4.2 工作机制

（1）成立领导小组

为积极推进项目建设，保障各项工作顺利开展，成立“达州市城市体检信息平台建设工作领导小组”，领导小组组长由住房和城乡建设局指定负责人担任，负责平台建设的领导统筹和重大问题决策工作。

平台建设工作领导小组下设综合协调组、开发建设组、政策和技术指导组、政策监督组，负责具体建设工作开展。

（2）职责分工

综合协调组：负责项目建设立项、招标、实施和验收的全过程领导、监督和协调工作，以及项目初验、终验的组织审查与批复。

开发建设组：负责项目的立项申报、总体需求分析、开发建设、系统测试运行和项目验收等具体性工作。

政策和技术指导组：负责对项目建设方案、项目建设全过程的政策把控、技术指导，立项审批前的前置审查，与政务云基础平台、政务信息资源共享平台、政务服务平台、“互联网+监管”平台的对接工作。

实施监督组：负责项目整体实施情况、开展进度和总体成效的监督工作，负责项目资金使用监督管理工作。

4.3 人员培训方案

4.3.1 培训原则

（1）理论联系实际的原则

对系统使用人员进行培训时要结合实际统计工作对系统操作进行培训，叙述要简单明了，通用易理解，不要过于术语化，使工作人员能够在短时间内熟练操作系统，提高工作效率。

（2）逐级负责、齐抓共管的原则

针对相关业务管理机构工作人员的素质状况和技术人员和业务主管人员的不同培训需求，分别制定不同的培训目标、培训内容和培训要求。

4.3.2 培训方式

（1）集中培训

集中培训是以培训班形式开展的一种培训方式，受训者将会接受相关内容的正规培训，并会参与交互式的、系统的、循序渐进的学习，最终提高自身的技术水平。针对本项目，根据受训者职责不同合理安排培训内容。

（2）现场培训

单纯的讲解是枯燥无味的，受训者也不容易接受。在培训过程中，安排实际操作培训，增加受训者感性认识，提高培训质量。

（3）操作辅导与答疑

由于个体差异，以及培训的效果不同，不是每个受训者都能通过一次培训把讲师所讲的内容全部消化吸收，对培训内容有疑问是培训过程中必然存在的。如何消除受训者的疑问是提高培训质量的重要环节，是后续培训课程顺利进行的前提。

针对上述情况，提出操作辅导和答疑培训服务。根据受训者在操作过程中提出问题、分析问题、解决问题：

（4）集中培训答疑

这部分的答疑主要在受训者集中培训的过程中进行的，可以在课后，可以在操作现场。

（5）实施过程中答疑

受训者在集中培训之后，在实施过程中可能还有疑问，提供远程答疑。受训者和讲师可以通过通讯工具交流问题，如果不能解决的，派相关技术人员到现场指导。

4.3.3 培训组织管理

（1）每次培训组织编写培训计划（包括培训内容、培训方式、培训效果评价）。

（2）每次培训的组织不能够影响到业务单位日常管理业务。

（3）每次培训必须配发相关的培训资料。

（4）每次培训必须进行相关的考核，以检验培训的效果，对于培训效果不合格的，进行再次培训。

4.3.4 培训队伍要求

“达州市城市体检信息平台”的建设，是一个涉及多部门、投资金额巨大的信息化工程，必须组建一支分工明确、人员固定的技术培训队伍。而这支实施队伍的组建工作应该在信息化建设项目启动前完成。

对每一阶段的试运行前，都要针对涉及系统功能的相关员工做培训，并达到系统使用人员能够熟练操作系统，从而使新系统在试用过程中，不影响住建厅日常管理业务。

4.3.5 培训对象

本次“达州市城市体检信息平台”培训对象主要涉及各区市县相单业务单位相关工作人员，以及部分房地产从业主体企业相关负责人、工作人员等，并针对可能改变的业务流程，向办事群众提供相关的业务办理变更情况说明。

4.3.6 培训内容及要求

培训内容以 “达州市城市体检信息平台”系统的日常管理知识培训为主，

根据不同的培训对象分批次进行集中式培训。培训内容及时间安排如下所示。

序号	培训课程	培训内容	学时	培训对象
1	“达州市城市体检信息平台”系统日常管理、使用培训	系统基本介绍、演示 系统后台管理、操作介绍 系统联合应用介绍、演示 系统后台管理、操作上机练习 现场答疑	5 天	市住建局工作人员
2	“达州市城市体检信息平台”行业应用日常使用培训	应用基本介绍、演示 应用操作介绍 现场答疑	4 天	部分相关从业主体人员

5. 项目风险分析

5.1 项目风险与风险对策

5.1.1 编制依据

项目风险分析和评估的依据包含相关法律、法规、规章、规范性文件以及其他政策性文件，国家出台的区域经济社会发展规划、国务院及有关部门批准的相关规划等。

5.1.2 风险调查

项目风险范围包含本次的建设单位、使用单位以及运维保障单位。风险调查的方式有全面调查、抽样调查、个案调查和典型调查，调查的方法有观察法、访谈法、文献法、问卷法、实验法等。

根据本次城市体检系统的特点及城市体检和房屋管理信息化建设的实际情况，选择适用的方式方法进行调查。实际工作中可采取公告公示、实地踏勘、召开座谈会以及舆情分析等多种方式和方法，以达到广泛调查、充分收集各方意见和诉求的目的。

5.1.3 风险识别

风险识别是项目风险管理活动中的重要环节。城市体检升级建设项目由于受到政府项目监管及财政管理体系、领导个人意志、层层审批决策机制以及实施方

对政府业务特点把握能力等多种客观因素的影响，风险种类更多，如果不能很好地进行管理，会对整个项目的进展造成严重影响，甚至导致项目失败。

结合本次城市体检信息系统升级建设项目的特点和前期的需求调研准备工作，目前已识别的项目风险包含以下内容：

5.1.3.1 领导决策风险

由于政府是层层决策机制，很难在一开始就得到高层领导的指示，而每一级领导通常都会有自己的看法，经常出现项目实施已接近完成，却被主管领导一票否决的情况。

出现概率：低

影响程度：大

5.1.3.2 制度风险

系统设计时未充分考虑外部因素，实施过程中受到其它强力政府部门以不符合某方面规划等理由（例如安全、保密）对系统提出较大幅度的更改要求。

出现概率：较高

影响程度：中

5.1.3.3 项目协同配合风险

需要多部门配合的系统可能会涉及到某个或某几个部门的内部规则（如行业或者部门管理模式），或者由于这些部门信息化建设制约（如各种理由导致系统无法接口），在建设和推广过程中受到阻挠。

出现概率：高

影响程度：大

5.1.3.4 技术风险

项目应用创新及系统软件的开发技术风险。由于本项目采用的均是较为成熟的信息化技术，此风险较低。

出现概率：低

影响程度：大

5.1.3.5 运行风险

很多情况下，新的信息系统建成后，由于旧传统旧模式的惯性，在运行过程中会存在业务部门对新系统不习惯、不接受、不认可的风险。同时，业务部门对新系统往往有这样那样的意见，如果这些意见得不到及时解决，他们对新系统的应用就可能不配合，甚至有抵触情绪，这对新系统的成功运行会带来相当的风险。

出现概率：中

影响程度：中

5.1.3.6 安全风险

项目建设过程中涉及到大量敏感数据信息，系统安全必须保障。本项目针对系统建设中最可能出现的安全问题提出了相应的手段，但仍然存在一定的安全风险。

出现概率：中

影响程度：中

5.1.4 外部风险分析及防范

我们针对以上已识别的外部风险深入分析，提出了以下项目风险规避及防范措施，保证项目工作正常运行：

5.1.4.1 领导决策风险对策

高层领导考虑的多是战略层面的问题，基层领导考虑的多是细节层面的问题，通常难以统一，想让需求一次性确定基本是不可能的。因此在做系统的时候要尽量使架构灵活，可扩充性强。软件开发尽可能采用构件或模块方式，增强重用性，最大限度适应需求频繁变更。在正式实施前多通过静态原型等手段汇报沟通，充分了解各级领导的偏好后再确定方案。

另外正式实施前要多请示，阶段工作要常汇报，在让上级领导决策前要尽量

说明前期已完成工作，并预先指出哪些变更会对项目产生颠覆性的影响，以免领导在未做详细了解的情况下主观表态。

5.1.4.2 制度风险对策

建设前期尽量与各主管部门及所有可能涉及到的业务部门加强沟通，全面征求意见，事先取得支持，同时在技术实现上尽可能采用开放标准和可扩展的架构。

5.1.4.3 项目协同配合风险对策

一方面要与相关部门的主管领导沟通争取得到支持，另一方面要想办法尽量减低对各部门既得利益的损伤，并与所有的既得利益者商议如何通过新系统获得新的利益均衡，必要时需要请监察局等部门进行协调或报请更高层的领导批示。

5.1.4.4 战略改变风险对策

通常只有大的人事变动或者大的政策变化才会影响到一个部门的整体战略，因此要经常与主管部门沟通，保持与各相关部门尤其是规划部门的密切联系，确保项目目标与部门战略的长期一致性，如果无法避免战略变更，应通过多层面沟通，争取在制定新战略时优先考虑加入与项目相关的战略内容。

5.1.4.5 技术风险对策

降低技术风险的主要对策包括：

(1) 采用成熟、主流技术，并进行风险评估及论证；

(2) 参照省内外类似项目建设的成功案例和建设经验；

(3) 关键产品的选择进行选型前的测试；

(4) 选好项目实施的合作伙伴；

(5) 聘请知名专家组成专家咨询委员会，对工程建设的规划、设计和实施进行技术指导和把关；

(6) 工程建设逐步展开、稳妥推进。

(7) 规范操作，严格按照规程进行操作，并及时督促相关单位做好维护维修。

5.1.4.6 运行风险的对策和管理

首先，系统建设要尽最大努力给用户带来方便的体验。要给用户供最合适的IT 产品和服务。要尽量避免让用户当测试方，尽量在这方面少占用业务部门的时间和精力;在无法避免时，要尽量缩短用户测试期。要坚决避免为了图自己的方便，将还没完成测试的产品交付用户，让业务部门当义务测试员，增加业务部门的工作量，这会给用户带来非常不好的试用体验，造成用户对新系统的反感。

5.1.4.7 安全风险对策

在项目详细设计阶段，再次论证系统的安全风险问题，提出更为完全的安全保障方案。预先筹备安全事项，采取防火墙、认证技术、加密技术等确保系统使用安全。

5.1.5 风险对策和管理

根据规定的里程碑监督风险、制定风险决策与风险减轻策略，采取如下措施。

1) 如果一项风险减轻策略无效，则改变这一策略。

2) 实施计划好的应急活动。

3) 当风险不存在时，将其从可能风险列表中消除。

在没有可使用的应急计划时，使用权变措施——对风险事件未计划的应对措施。

5.1.6 风险评估综述

正是因为平台建设中存在技术、管理、运行等多方面的风险，在实施项目中应积极采取措施和手段来合理规避项目建设过程中的软、硬件、场地技术性风险，项目管理中的人员、管理手段等风险后，最大限度的确保本项目按照项目质量、进度、投资要求顺利完成项目建设，并确保项目投入运行。