软件工程遗留系统维护中的常见问题与解决措施

龚晨宇
青岛恒星科技学院大数据与智能信息工程学院，山东 青岛 266000
DOI: 10.61369/TACS.2026080004

摘　　要： 遗留系统在企业信息化运行中仍承担着数据沉淀、业务衔接和流程支撑功能，但受技术架构老化、代码规范不足、数据治理薄弱、运维机制不健全等因素影响，维护难度持续增加。文章从软件工程维护类型与基本原则出发，分析遗留系统在技术、数据、管理人员、业务成本等方面的常见问题，并提出架构分步迭代、代码治理、数据清洗、安全加固、流程规范、人员交接、故障分级和风险管控等措施，以期提升遗留系统维护的稳定性、经济性与可持续性。
关 键 词： 软件工程；遗留系统；系统维护；数据治理；运维管理

Common Problems and Solutions in the Maintenance of Legacy Systems in Software Engineering

Gong Chenyu
School of Big Data and Intelligent Information Engineering, Qingdao Hengxing University of Science and Technology, Qingdao, Shandong 266000

Abstract: Legacy systems continue to play vital roles in enterprise informatization, including data accumulation, business integration, and process support. However, their maintenance difficulty is continuously increasing due to aging technical architectures, inadequate coding standards, weak data governance, and imperfect operation and maintenance (O&M) mechanisms. Based on the types and fundamental principles of software engineering maintenance, this paper analyzes common problems in legacy systems concerning technology, data, personnel, and business costs. Furthermore, it proposes measures such as step-by-step architecture iteration, code governance, data cleansing, security hardening, process standardization, personnel handover, fault grading, and risk management. These measures aim to enhance the stability, cost-effectiveness, and sustainability of legacy system maintenance.
Keywords: software engineering; legacy system; system maintenance; data governance; O&M management

引言

在企业数字化转型不断推进的背景下，许多早期建设的信息系统并未退出运行，而是继续承担订单处理、客户管理、财务结算、生产调度等基础业务。此类遗留系统往往与企业历史流程、核心数据和岗位习惯深度绑定，直接替换成本高、风险大。与此同时，运行环境变化、业务需求扩展和安全要求提升，又使其维护压力日益突出。如何在不影响业务连续性的前提下完成维护优化，已成为软件工程实践中必须面对的现实课题。

一、软件工程遗留系统维护概述

遗留系统维护首先要明确其工作边界。只有辨清维护类型、把握维护原则，后续问题分析才不会停留在零散故障层面。对企业而言，遗留系统既是技术对象，也是业务载体，维护工作需要在稳定运行、有限成本和持续迭代之间寻找平衡。

（一）遗留系统维护的主要类型

遗留系统维护是围绕系统持续运行、业务延续和技术适配展开的综合性工程活动。按照软件工程运维规范，其维护类型通常可划分为纠错性维护、适应性维护、完善性维护和预防性维护。纠错性维护主要针对系统运行中暴露出的程序缺陷、接口异常、数据处理错误等问题，目标是尽快恢复系统稳定状态[1]。适应性维护则多发生在软硬件环境、数据库版本、操作系统或外部接口发生变化之后，重点解决系统与新环境之间的不匹配问题。完善性维护侧重于根据业务部门提出的新需求，对原有功能、操作流程或报表输出方式进行优化。预防性维护更强调风险前置，通过代

龚晨宇（2006—），男，汉族，福建南平人，本科学生。

Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.

码重构、性能巡检、安全加固和文档补全，减少后续故障发生概率。四类维护相互关联，共同支撑遗留系统的持续可用。

（二）遗留系统维护的基本原则

遗留系统往往承载着企业长期积累的业务规则和历史数据，维护工作不能脱离实际运行场景，更不能以单纯追求技术先进性为目标。实践中应坚持业务优先原则，先判断系统问题对核心业务、用户服务和数据流转的影响程度，再确定维护顺序。最小改动原则要求维护人员在充分理解原有逻辑的基础上控制修改范围，避免局部调整引发连锁故障。安全可控原则强调任何补丁、接口调整和数据迁移都要经过测试、备份和回退设计[2]。成本适配原则要求在系统剩余价值、维护投入和替代成本之间作出合理权衡，避免低价值系统长期占用资源。兼容迭代原则则强调新旧系统之间的平稳衔接，通过接口封装、模块替换、分阶段升级等方式，为后续维护和转型留下空间。

二、软件工程遗留系统维护中的常见问题

遗留系统维护难度是技术老化、数据积累、管理松散和业务变化共同作用的结果。若只处理表层故障，系统风险仍会反复出现。因此，有必要从多个维度梳理常见问题，找到维护压力持续加重的真实来源。

（一）技术层面存在的问题

遗留系统最直接的维护压力通常来自技术底座老化。部分系统仍依赖早期开发语言、过时框架和低版本数据库，运行环境一旦升级，原有程序便可能出现接口失效、驱动不兼容、性能下降等问题。更棘手的是，许多遗留系统在建设初期缺少统一编码规范，后续又经历多轮补丁式修改，导致业务逻辑被分散嵌入不同模块，冗余代码、无效代码和重复判断较多。维护人员在定位故障时，往往需要反复追踪调用链，稍有疏漏便可能误改核心逻辑。同时，需求文档、设计文档和运维记录更新不足，使系统结构、数据流向、异常处理方式难以被准确还原，维护工作容易陷入“凭经验判断、靠试探修改”的状态。

（二）数据层面存在的问题

遗留系统长期运行后，数据问题往往比程序问题更加隐蔽。由于早期数据标准不统一，不同模块可能采用不同字段命名、存储格式和校验规则，造成同一业务对象在多个表中存在重复记录或含义不一致的情况。部分历史数据在迁移、补录和人工修改过程中缺少校验机制，容易出现数据缺失、错位、乱码和关联关系断裂等现象。一旦这些数据继续参与统计、结算或业务审批，就会影响系统输出结果的可信度。与此同时，一些遗留系统安全防护设计薄弱，数据加密、权限分级、备份恢复机制不健全，面对误操作、硬件故障或外部攻击时缺少有效应对能力。若数据又无法与新系统顺畅互通，企业便难以形成完整的数据资产。

（三）管理与人员层面存在的问题

遗留系统维护并非单纯的技术工作，管理机制薄弱会进一步放大系统风险。部分单位缺少稳定的运维流程，对故障登记、原因分析、版本发布、权限调整和回退处理没有形成标准要求，导致同类问题反复出现却难以沉淀经验。版本管控不严时，不同人员可能在不同时间对同一模块进行修改，最终出现代码覆盖、配置混乱或测试环境与生产环境不一致等问题。人员层面的断层同样突出，熟悉早期架构和业务规则的技术骨干离岗后，新维护人员只能通过零散代码和旧资料理解系统，接手周期长，误判概率高。若缺少故障分级机制，日常小问题还会占用大量维护时间，真正影响核心业务的隐患反而得不到及时处理。

（四）业务与成本层面存在的问题

遗留系统之所以难以简单替换，往往是因为其仍承担着关键业务，但功能形态已难以匹配新的业务节奏。随着企业流程调整、线上服务扩展和数据共享需求增加，原有系统固定化的功能结构逐渐暴露出局限，新增需求只能通过临时字段、外部表格或人工补录方式勉强支撑，业务效率随之下降。与此同时，系统老化带来的故障频率上升，使维护人员不得不投入更多时间处理性能卡顿、接口异常和数据修复问题，软硬件兼容、备份设备、老版本授权等隐性成本也会持续增加。更为突出的是，遗留系统中许多功能存在强耦合关系，局部调整可能牵动多个业务环节，一次看似简单的修改也可能引发连锁故障，影响系统连续运行。

三、软件工程遗留系统维护中常见问题的解决措施

（一）技术层面优化措施

遗留系统技术维护应避免“一推倒就重建”的简单思路，更适合采取分层评估、局部替换、持续验证的渐进式策略。维护团队可先对系统架构、运行环境、核心模块、外部接口进行技术盘点，明确哪些部分必须保留，哪些部分可以封装，哪些部分具备替换条件。对于仍能稳定支撑业务的模块，不宜频繁改动；对于存在明显性能瓶颈、接口风险和安全隐患的模块，则可通过轻量化重构逐步优化[3]。具体实施中，可先搭建独立测试环境和灰度验证环境，将老旧框架、数据库驱动、运行组件的升级控制在可回退范围内，防止一次性升级造成生产系统中断。代码治理也应同步推进，维护人员可按照业务模块建立代码清单，清理长期不用的冗余代码、重复逻辑和失效配置，对核心方法、接口调用、异常处理补充必要注释，并形成统一命名、提交、审查和测试规范。对于无法直接改造的老旧系统，可通过中间件、网关接口、适配器模式等方式完成外部兼容，使其能够与新系统、新设备和第三方平台保持数据与业务交互。技术文档补全不能作为事后整理，而要嵌入每一次维护过程，凡是涉及架构调整、参数变更、故障修复和版本发布，都应同步更新系统说明、接口说明、配置说明和应急处置记录。这样既能降低维护对个别人员经验的依赖，也能为后续系统迁移和更新提供可靠依据。

（二）数据层面优化措施

数据层面的维护应坚持“先摸底、再清洗、后治理”的顺序，不能在数据关系尚未厘清的情况下直接迁移或批量修正。维护团队可先建立数据资产目录，梳理遗留系统中核心数据表、字段含义、主外键关系、历史数据来源及其对应业务流程，重点识别重复字段、异常值、空值比例较高字段和长期无人使用的数据

 Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.

项。在此基础上制定清洗规则，对格式不统一、编码不一致、重复记录、无效数据和逻辑冲突数据进行分批处理。涉及客户信息、订单记录、财务数据、审批数据等关键内容时，应先备份原始数据，再通过校验脚本、小批量试运行和人工抽检验证处理结果，避免清洗过程造成新的数据损坏。数据存储标准也要同步建立，明确字段命名、数据类型、长度限制、校验规则和更新责任，防止新数据继续沿用旧问题。安全防护方面，应根据数据敏感程度划分访问权限，对重要数据采取加密存储、传输加密、日志留痕和定期备份措施，关键岗位操作还应设置审批和复核环节。对于遗留系统与新系统之间的数据孤岛问题，可通过数据接口、消息队列、数据中台或定时同步任务实现互通，但接口设计必须明确同步频率、失败重试、冲突处理和异常告警机制。只有把数据清洗、数据安全和数据共享放在同一治理框架中推进，遗留系统中的历史数据才能从维护负担转化为可继续使用的业务资源。

（三）管理与人员层面优化措施

遗留系统维护要真正稳定下来，必须把个人经验式处理转向制度化、流程化管理。企业可围绕日常巡检、故障受理、问题定位、代码修改、测试验证、版本发布、权限调整和应急回退建立完整流程，每一类操作都要明确责任人、审批节点、记录方式和完成标准。日常维护中，可设置统一的问题登记台账，对故障时间、影响范围、处理过程、根本原因和后续预防措施进行记录，避免同类故障反复出现。版本管理应使用统一工具和分支策略，未经评审和测试的代码不得直接进入生产环境，配置文件、数据库脚本和接口参数也应纳入版本管控范围。对于人员断层问题，应建立“系统知识库 + 交接机制 + 岗位梯队”的组合策略。熟悉旧系统的技术人员要将关键业务逻辑、特殊处理规则、历史故障案例和运维注意事项沉淀为文档，新接手人员则通过代码走读、故障复盘、模拟演练逐步熟悉系统[4]。企业还可安排新老人员共同处理一段时间的真实维护任务，让技术经验在具体问题中完成传递。故障处理方面，应按照业务影响范围、数据风险程度、用户受影响数量和恢复时限建立分级机制。重大故障优先保障核心业务恢复，一般问题进入排期处理，低风险优化纳入周期性迭代。通过制度、人员和优先级三方面协同，维护资源才能从被动救火转向有序配置。

（四）业务与成本层面优化措施

遗留系统维护不能只围绕技术缺陷展开，还要回到业务价值和成本承受能力上进行判断。企业可先组织业务部门、技术部门和管理层共同完成系统价值评估，明确哪些功能仍支撑核心流程，哪些功能只是历史遗留，哪些功能已经被新平台替代。对于仍具备使用价值的功能，应以“小步调整、贴近流程”为原则开展优化，例如调整表单字段、简化审批路径、优化查询报表、增加必要的数据校验，使系统能够更好服务现有业务；对于长期闲置或重复建设的功能，则应逐步停用、归档或合并，减少无效维护范围。成本管控方面，应建立年度维护清单，将硬件设备、软件授权、故障处理、人力投入、安全加固和备份资源等成本纳入统一核算，避免维护支出长期处于模糊状态。对故障频发模块，可通过巡检、性能监测、容量预警和日志分析提前发现风险，减少临时抢修带来的高成本投入。涉及系统升级和功能迭代时，必须建立风险评估机制，提前判断改动范围、关联模块、数据影响和回退路径。重要修改应经过需求确认、方案评审、测试验证、备份留存、灰度上线和运行观察等环节，不能因业务催促而跳过必要控制。对于确实难以继续支撑业务的系统，也应制定替代路线图，明确过渡周期、数据迁移方式和新旧系统并行策略[5]。这样既能控制维护成本持续攀升，又能降低系统更新对正常业务造成的冲击。

四、结语

遗留系统维护是围绕业务连续、数据安全和技术更新展开的系统性治理工作。面对架构老化、数据混乱、人员断层和成本攀升等问题，企业应避免盲目重构或临时性修复，转而建立分步迭代、标准运维、数据治理和风险控制相结合的维护机制。只有将技术优化与业务需求、人员储备、成本管理同步考虑，才能降低系统运行风险，延长有效使用周期，并为后续系统升级或平稳替换奠定基础。

参考文献

[1] 孙玥，刘军，蒋林江，等 . 核设施滞留量多点位在线测量系统软件开发 [J]. 核电子学与探测技术 ,2025,45(11):1705-1713.

[2] 杨晨，池瑞楠，李晨来 . 软件工程技术专业人才培养模式与策略研究 [J]. 深圳职业技术大学学报 ,2025,24(5):99-104.

[3] 陈徐毅 . 大模型演进与软件工程实践的融合探索 [J]. 通信世界 ,2025,(16):35-36.

[4] 朱溯，胡竹青 . 业务中台与成熟套装软件集成协作模式研究 [J]. 自动化应用 ,2025,66 (12):91-93.

[5] 高生华，何济武，胡素臻，等 . 面向遗留系统的数据采集方法 [J]. 网络安全与数据治理 ,2023,42 (S2):156-160.

Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.