THE FORELAND OF
TRADING TECHNOLOGY

内部资料 免费交流
《准印证》编号沪（K）0671

交易技术前沿

2023年 第一期 总第52期

本期主题

信息系统安全

No.1

内部资料 2023 年第一期（总第 52 期）
准印证号：沪（K）0671

NO.1

主管：上海证券交易所
主办：上海证券交易所
总编：邱勇、蔡建春
副总编：王泊
执行总编：唐忆
责任编辑：徐广斌、徐丹、陆伟、王昕、黄淦

上海市杨高南路 388 号
邮编：200127
电话：021-68607129，021-68602496
传真：021-68813188
投稿邮箱：ftt.editor@sse.com.cn

篇首语

习近平总书记指出：“安全是发展的前提，发展是安全的保障，安全和发展要同步推进。”证券期货从业机构的信息系统是否安全，直接影响资本市场的功能发挥和广大投资者利益，进而影响国家安全、政治安全和经济安全。当前，金融业正处于数字化转型和信息技术应用创新深入推进的关键时点，必须做到发展“第一要务”与安全“头等大事”并驾齐驱，实现稳中求进高质量发展。本期《交易技术前沿》以“信息系统安全”为主题，收录行业关于运行保障、网络与信息安全等方面的优秀文章。

《AIOPS 语义级日志异常检测在证券行业的探索与实践》针对传统的日志解析、关键字匹配等故障检测方式存在的缺陷，提出了一种融合语义理解、模式识别等多种技术手段的日志异常检测方式，并在生产环境下检验其有效性。

《数字驱动下的网络安全生态建设》结合证券公司信息安全体系建设实践，分享了检测防护、安全运营等多方面的技术探索成果。

《初探证券业 IPv4/IPv6 过渡阶段的安全防护》基于 IPv6 技术现状及面临的挑战，围绕网络协议、系统改造与升级、共存网络、人才供给等方面分享共同提升安全防护水平的案例经验。

《证券公司关于账号权限风险管理与稽核应用创新的探索和实践》聚焦系统权限的统一管理和自动化稽核，研发数据自动化对接、OA 系统自动授权、权限异动告警通知等功能，实现第三方业务系统权限模型与权限管理系统的快速对接，提升了权限管理的准确性和时效性。

《证券行业网站业务系统 IPv6 网络安全风险及防护技术探索》重点围绕过渡阶段、隐私泄露、网络安全、管理策略等关键环节，阐述防范化解广泛部署 IPv6 协议所面临的风险及挑战。

《交易技术前沿》编辑部

2023 年 7 月 13 日

目录 Contents

本期热点

实践探索

前沿技术应用

信息资讯采撷

本期热点

1 数字驱动下的网络安全生态建设

2 大数据流处理技术在金融资讯领域应用研究

数字驱动下的网络安全生态建设

陈凌云、李宁、刘兆友、李珂、殷俊 / 德邦证券股份有限公司 上海 200010
E-mail：yinjun@tebon.com.cn

金融行业数字化进程的高速发展，使得网络安全问题的影响不断凸显。随着我国网络安全法规制度不断完善，以及边界安全、数据安全、安全服务等重点领域的不断发展，德邦证券也在积极开展信息安全体系建设，通过检测防护、安全运营等十个方面的技术探索，为证券业务发展提供安全保障。

关键词：数字化转型；网络安全；安全体系

1 引言

金融行业数字化进程的高速发展，使得网络安全问题的影响不断凸显。以勒索、挖矿、APT等为代表的新型网络攻击活动的持续活跃，引发了全球每年数以千计的重大网络安全事件，网站被黑、业务停摆的新闻屡见不鲜[1]。数字化变革正面临着网络安全的重重危机。

2022年是“十四五”规划的第二年，我国网络法治建设在各个领域取得显著成果。网络安全法规体系基本构建，相继出台数据安全法、个人信息保护法、关键信息基础设施安全保护条例等夯实了网络法治的制度基础。数据治理立法体系全面展开，在国家总体安全观的指引下，统筹安全与发展，规范数据交易、跨境数据流动等数据处理活动，加强移动应用程序等领域个人信息权益保护。随着远程办公常态化、业务模式全球化、万物实现互联，社会走向全面信息化的过程中，也面临着前所未有的安全威胁与风险，网络安全行业因此正处在一个机遇与挑战并存的阶段。

2 安全现状

自2017年《中华人民共和国网络安全法》正式实施以来，我国网络安全法规制度不断完善。

2019 年《信息安全等级保护条例》2.0 的出台，倡导安全防护从过去的被动防御逐步向主动治理转移，标志着我国正式迈入等保 2.0 时代。而在 2021 年至 2022 年期间，《数据安全法》、《个人信息保护法》、《关键信息基础设施安全保护条例》、《网络数据安全管理条例（征求意见稿）》、《数据出境安全评估办法》等政策相继出台，在“十四五”规划中首次将数据作为生产要素，意味着我国从过去的信息安全全面走向数据安全。2022 年 9 月，国家互联网信息办公室发布了关于修改《中华人民共和国网络安全法》的决定（征求意见稿），处罚力度大幅提高表明了国家对网络安全作为国家战略的决心。

现阶段，网络边界安全仍然是网络安全行业最主要的赛道。然而随着内网攻击事件频发，数据接入方式和终端的不断多样化，云化业务场景的普及，网络安全边界逐渐模糊，越来越多的迹象显示，传统的防火墙、VPN、IPS、上网行为管理、邮件网关等基于边界的安全防护模式已难以应对日趋复杂的攻击环境。Gartner 在 2022 年八大安全和风险管理趋势中提出，要建立以身份为优先的安全机制，将身份作为新的安全边界。以零信任为代表的基于身份授权来实现访问控制的产品，成为近年行业的新增长点。

安全服务方面，安全咨询、安全托管服务（MSS）、安全教育与培训是全球网络安全的主流市场。然而在我国，安全市场仍以软硬件产品作主导，安全服务占据的规模仍然较小，产业成熟度也与发达国家水平存在一定差距。造成这种结构化差异的原因可能是因为安全服务在我国起步较晚，用户还没有对安全服务养成付费习惯。然而随着企业对于体系化安全架构需求的激增，SOAR 等运维产品的不断成熟，安全托管服务的需求及市场在近年持续增长，网络安全行业格局也将逐步由软硬件产品导向转向安全服务导向。

数据安全方面，随着物联网的深度普及，车联网、智能家居等物联网系统在实际运用过程中，涉及了大量的数据采集、存储与分析，数据逐渐成为企业最主要的资产。用户个人信息泄露、竞争企业数据窃取与篡改等事件频频出现，2021 年 7 月“滴滴严重泄露用户信息”事件引发了热烈的社会讨论，数据隐私问题受到了空前的关注。在“十四五”规划中，“数据”首次被列为生产要素，数据安全也成为了发展风口。

安全有效性验证方面，目前国内主要的投入基本集中在事前和事中阶段，对于具体偏事后的安全溯源、取证分析、安全验证上偏弱。有效的安全验证才能形成完整的闭环，安全验证简单来看可以划分为如下三个层次：第一层，点验证，主要验证各个防御设备的检测能力，是不是攻击能够全部检测；第二层，线验证，主要确保安全设备、siem、soar、告警平台之间的链路是否存在异常；第三层，面验证，通过红蓝对抗的方式，全面检测企业的安全防御能力和安全运营效果。目前业内安全有效性验证的解决方案及产品刚起步，但已展露出强大的潜力和市场需求，这会是未来几年内安全专业可深挖的一个方向。

此外，云安全、移动互联、物联网、工业互联网、人工智能等新型技术的深度发展，为网络安全带来了爆发式的需求，成为了 2022 年乃至以后网络安全行业发展的重点领域 [2]。

3 德邦证券安全举措

面对数字化转型下的安全现状，德邦证券也在积极开展信息安全体系建设，为业务发展提供保障。

3.1 网络边界防护

网络边界是公司与互联网之间的第一道防线，确保网络边界有切实有效的防护措施是保障公司网络安全的基础。目前公司实施区域化隔离措施，公司全网划分核心区交易区、分支接入区、外部单位接入区、DMZ 互联网区等等，各个区域

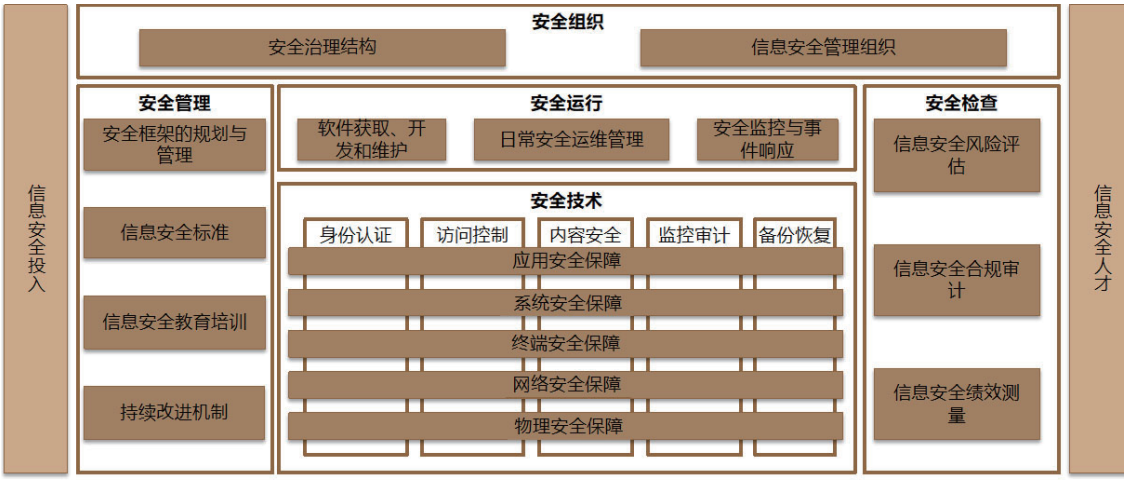


图 1：信息安全体系建设框架

之间通过防火墙进行安全隔离。生产网、办公网、IDC 均部署了 WAF 应用防火墙和 IPS 入侵防御系统，从实际情况来看，能够有效阻挡应用层恶意攻击，和防火墙网络层的保护形成互补。邮件系统部署了邮件安全网关，提高邮件安全防护能力。网站使用 SSL 安全加密机制，提高网站安全性。

3.2 内网防护

攻击者一旦突破边界防护，内网防护将会变得非常薄弱，因此我司在内网区域部署了蜜罐系统，通过在攻击者入侵的关键路径上部署诱饵和陷阱，诱导攻击者转移攻击目标，进入与真实网络隔离的蜜网，让攻击者在蜜网中攻击“假”目标，获取虚假数据，从而拖延攻击时间，间接保护真实资产。在此过程中，蜜罐能完整记录攻击者行为，捕获高级未知攻击（比如基于 0day 的 APT），并且可以对攻击者做身份溯源，为我司提供先人一步的主动防御手段。

3.3 终端安全

终端部署了企业版杀毒软件、主机入侵检测系统、终端安全软件，杀毒软件能够对木马病毒、恶意文件实时查杀；主机入侵检测系统能检测 webshell、暴力破解、恶意代码执行、本地提权等恶意行为，资产清点功能很快检测服务器的组件安装情况，大大提高了排查速度，为漏洞修复

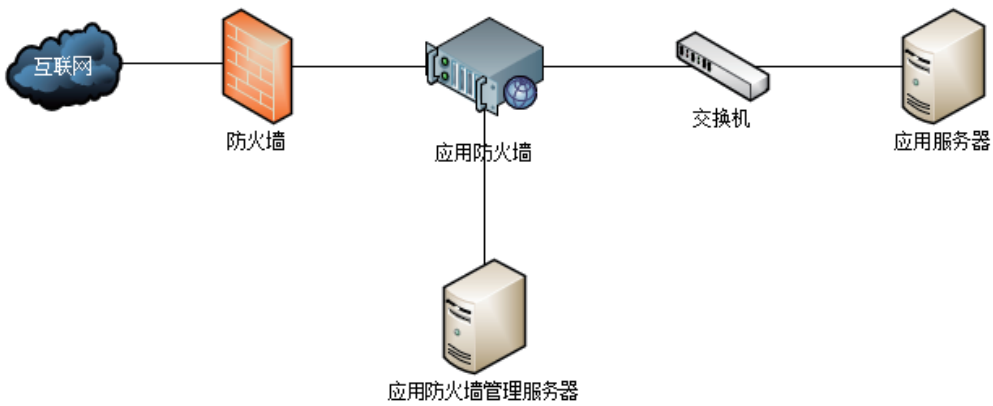


图 2：网络架构图

提供了宝贵时间，日常监控能捕获到攻击者的整个攻击过程，包括执行记录、启动记录、权限变更记录、登录记录等，并精准定位攻击源，避免了攻击者的进一步攻击，合规基线功能能用于多种不同操作系统、Web 服务、容器服务、数据库等应用的合规检测，检测出不合规配置；终端安全软件能对办公 PC 进行软件、基线检查、USB 等统一管控，有效加强终端安全。

3.4 安全检测机制

我司实行不间断的安全检测机制，每周开展信息安全日常巡检、每季度进行系统和 WEB 扫描、不定期的渗透测试，要求高危漏洞和中危漏洞要及时完成修复，形成了良好的漏洞检测修复闭环机制。

3.5 SOC 运营中心建设

进行态势感知平台的建设，通过流量探针、日志探针、分析平台等套件存储全量原始数据和日志，通过关联分析模块整合告警、过滤掉可信度低、无用的告警，降低人力成本的同时提高安全响应水平。经市场调研，对多家态势感知产品进行了测试和横向比较，选择综合能力最好的产品进行采购，为 SOC 运营中心建设奠定基础。

3.6 红蓝对抗演练

红蓝对抗是模拟真实攻击的实战演习，以获取内网核心系统、数据、权限为目标，采用互联网系统入侵、社工、内网隐蔽移动等手段，挖掘深层次多角度的安全问题。每年我司会安排 1-2 次红蓝对抗演练，以此来检验我司面对网络攻击时的应急处置能力以及配合协同能力。

3.7 网络安全风险排查

我司定期开展网络安全风险排查，包括弱口令排查、双网卡排查、访问策略收紧、互联网出口合并、三无账号清理、域名清理、离职换岗人员的账号权限回收、机房门禁权限梳理等等，对于存在问题的要求相关人员完成整改，对于安全加固取得了不错的效果。

3.8 红军建设

培养专业攻击型安全人才，我司每年积极参加社会组织的各种网络安全竞赛，比如网鼎杯、观安杯等，通过比赛来锻炼安全人员的渗透能力，让安全人员站在攻击者的角度来审视我司的防护

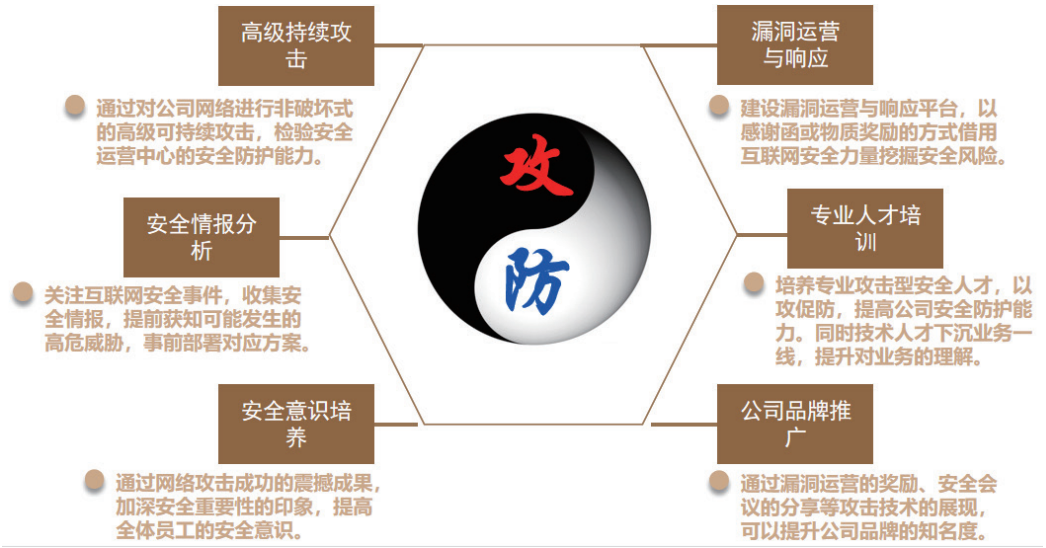


图 3：红军建设

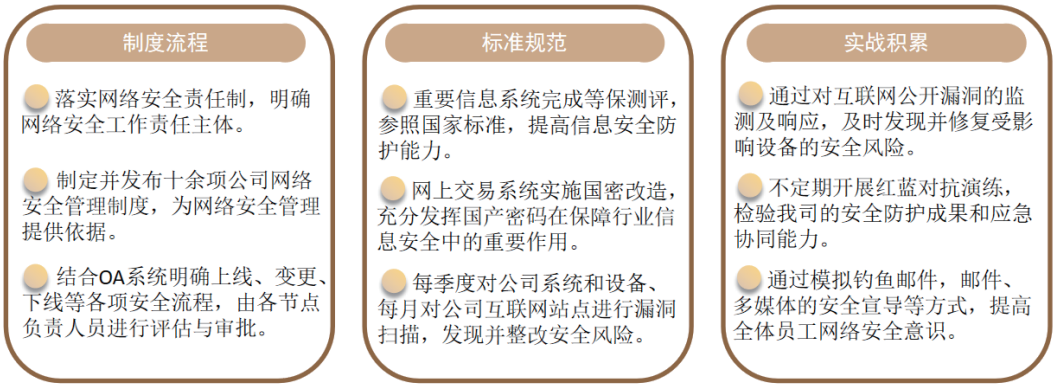


图 4：制度规范

体系，以攻促防，同时技术人才下沉业务一线，提升对业务的理解，以此提高公司安全防护能力。

3.9 制度规范

我司成立了以公司主要负责人为组长的信息安全工作小组，形成自上而下的网络安全组织架构，发布了一系列落实责任制、系统上线变更安全检测机制、网络安全应急处置、安全培训等十余项制度规范，为网络安全管理提供依据。

3.10 安全意识宣贯

我司定期开展网络安全培训，通过线上课程、钓鱼邮件专项测试、举办网络安全宣传周活动、安全知识考试等多种形式，来为全司员工普及安全知识，让大家知道哪些可为、哪些不可为，保护员工自身权益的同时也是在保护公司，通过几年来不懈的努力，员工的安全意识提升明显。

4 安全展望

如今的互联网不仅是我们日常生活的一部分，也是各个行业数字化转型的重要载体，新技术势必带来新的安全问题，有些问题已经出现，有些问题尚在萌芽之中 [3]，在此背景下，任何细小的漏洞都可能导致我们的经济资产遭受损失，我们应密切关注了解新技术发展动态，并为可能出现的风险做好准备。

4.1 基于物联网的网络攻击

物联网经过多年的发展，引领信息产业革命的新一轮浪潮。随着物联网的深入应用，其对人们的日常生活以及社会经济发展已产生了极大的影响，物联网设备作为物联网的载体，使用率不断提高也为黑客利用敞开了大门。网络犯罪分子可以通过攻击物联网设备来实现窃取敏感数据、身份冒充、信令拥塞、恶意程序植入、僵尸网络感染等等，危害风险巨大。

4.2 人工智能不是网络安全的“救世主”

在有关新一代网络安全技术的讨论中，AI 无疑是一个高频出现的词汇，总是被反复提及。当前的网络攻击形势愈发严峻，安全人员对于利用 AI 技术保护数字资产表现出了极大热情。他们认为，人为错误是不可避免的，而 AI 技术可以利用大数据和机器学习持续性观察和监控威胁发展趋势，从而优化提升某些防御行为。但事物总是具有双面性，随着对 AI 技术应用效果的观察，很多用户和安全专家发现，目前的 AI 技术还并不能成为网络安全领域的“救世主”，至少并非像很多安全厂商宣传的那样。人们往往只关

注 AI 的好处，却有意或无意地忽略了它的应用缺陷，包括 AI 过度依赖大数据、AI 应用技能不足、AI 没有创造力和自发性等等，以上缺陷都会成为黑客利用的焦点。

4.3 软件供应链安全

软件供应链是各类关键信息基础设施平稳运行的基础，其关键组件的设计、开发、部署、监控和持续运营等生命周期核心环节的安全可控，成为网络安全的关键考量因素。从近几年的软件供应链攻击事件来看，利用开源社区、公共开源存储仓库等开源软件生态入侵事件较为严重。因此需要从监管层面加强供应链产品安全认证管理。作为企业，需要构建自身产品的软件成分清单来梳理软件供应链信息，软件成分清单依据识别成分的粒度，可以分为不透明、微透明、半透明和透明几个阶段。透明程度高的软件成分清单，能显著提升软件供应链安全评估的准确性，同时需要完善供应链资产管理和安全检查，理清企业供应链依赖关系，从而在监测到预警时能够从容应对。

除了以上提到的三点之外，主动免疫可信计算，隐私计算，新技术新应用安全，关键信息基础设施安全保护，网络安全保险，数字货币安全，网络安全教育、技术与产业融合都是未来安全需着重需要关注的安全风口。

未来 5 年将是网络安全发展的关键时期，数字化改革势必带动一系列新法规、新技术、新产品的变革和诞生，对于安全是一种挑战也是一种机遇。

参考文献：

[1] 2022 年网络安全趋势报告 [ER/OL]. https://zhuanlan.zhihu.com/p/523184830. 2022,6.

[2] 云计算、大数据、物联网、工业互联网、人工智能等新技术新应用 [ER/OL]. https://zhuanlan.zhihu.com/p/137693382. 2022,4.

[3] 警惕 6 种或将愈演愈烈的互联网安全威胁 [ER/OL]. https://mp.weixin.qq.com/s/Xd73mCheI-epVqbYAjg09DA. 2022,11.

大数据流处理技术在金融资讯领域应用研究

李云涛、柴森、蔡跃 / 中信建投证券股份有限公司 信息技术部 北京 100010
E-mail：caiyue@csc.com.cn

随着金融资讯领域信息技术应用的发展，用户对资讯的稳定性、准确性、实时性提出了更高的需求。本文以中信建投证券股票 F10 资讯数据服务为案例，分析当前资讯数据处理过程中所面临的时效性差、稳定性低、质量难以保证等问题，梳理大数据处理技术发展趋势，介绍资讯数据处理架构的演进过程，论证基于 Flink 流式计算引擎搭建的实时处理框架在资讯数据服务场景下应用的有效性。

关键词：金融资讯；大数据；实时计算

1 概述

1.1 背景及意义

随着企业数字化的开展，数据的重要性已经越来越被业内重视 [1]。数据作为券商开展业务的前提，其重要性不言而喻。金融市场数据主要包括股票、债券、基金、衍生品、指数、文本（新闻、公告、研报）等各类资讯，数据来源渠道广，且以非结构化数据居多，导致数据处理工作量很大。为了满足业务对于金融资讯数据的需求，券商往往采取采购、自研等多种形式来获取数据。

大型券商往往采购多源的金融资讯数据，从而避免系统对单一数据商依赖过重。同时，为减少数据在使用时前置的计算和衍生时间，有些券商会打造自己的金融资讯数据结构，并通过数据抽取、转换、加载（ETL）等形式，将数据商的数据进行处理和入库。原有的数据链路是基于 T+1 的场景进行设计，这种技术由于成熟稳定，是券商技术选型优先考虑的方案。但随着信息技术的发展，用户对资讯的准确性和实时性提出了更高的需求，原有的工具链在数据应用场景支持方面，已经显得捉襟见肘。

金融公司的资讯数据，除了在公司内部对运营决策提供数据支持[2]，还在股票 F10 等对外版块使用，为客户提供优质的资讯服务。由于系统所使用的部分原始数据，是由专业资讯数据商分类、清洗、落库，再分发到券商，其时效性会随之受到影响。为提高数据的时效性，提升用户体验，可以采用流处理技术降低数据在落地到应用间的时间损耗。本文以股票 F10 资讯数据从源头到最终接口输出的服务为例，分析数据处理架构的演进方式和流处理技术在金融资讯领域的应用成效，以此帮助券商打造自身的核心竞争力。

1.2 存在的问题

为了减少 API 在对原资讯商的数据调取时进行衍生计算的时间损耗，需要将资讯商的数据按照股票 F10 的业务需求进行数据转换。由于采购的股票数据源，属于实时更新数据源，原有架构在设计之初，流处理相关技术并不成熟，业内股票 F10 的数据仍以 T+1 更新为主，因此此时的技术选型为批处理 ETL 工具——Kettle。

作为传统 ETL 成熟工具，Kettle 基于 Java 语言编写，很好的继承了 Java 的跨平台性，而且免费开源，是理想的技术选型工具。但是由于采用了 Kettle 工具，且数据量较大，为保证数据链路的稳定性，我们调整了每个任务的 ETL 调度周期，通过 Kettle 的批处理方式，定时保持数据的同步与更新。

股票 F10 分为资金、新闻、简况和公告等模块，我们借助 Kettle 工具将源数据进行加工处理时，发现资金和简况等模块对应的原始数据，更新频率较低，以资金模块的五日两融数据为例，每个交易日早上由上海证券交易所和深圳证券交易所官网披露上一个交易日的两融数据，针对两融数据进行 ETL 操作时，可以保证每天开市之前，客户能看到最新的两融数据。而对于新闻类数据，每天个股新闻量较大，更新时间不固定，且新闻本身具有很高的时效性价值，有价值的新闻可以很好的帮助客户做理财决策。另外，公告数据的更新具有很强的时间集中性，每个交易日闭市之后，是上市公司密集发布公告的时间。通过 Kettle 的批处理方式，会造成在更新时段的服务资源紧缺，特别是当数据量大且集中更新时，kettle 批处理的任务往往在一个调度周期内无法完成全部的数据更新，这样不仅影响数据传输效率，还面临调度任务无法正常运行的风险。

金融公司的资讯数据，在面向外部客户提供资讯服务时，APP 调用的数据来自于互联网环境。在 Kettle 将数据进行加工处理后，需要将处理之后的数据实时同步到互联网环境。异构关系型数据库之间的数据同步，原方案是基于触发器的设计，首先需要基于所有表单，创建基于表单增加、删除和修改的触发器，用以捕获数据变化，将数据变化记录到表中，再通过专业工具对变化表进行轮询，找出本时间周期变化的数据，同步到外网数据库。该同步方式虽然可以满足数据同步的需求，但由于数据变化会激发触发器的执行，消耗数据库大量性能，对源库的正常运行产生很大的影响，且当出现大批量数据更新时，记录数据变化的表会变得很大，专业工具在对该表进行轮询时，极易造成数据库性能下降，导致工具的崩溃。该同步方式基于触发器实现，在时效性上有保障，但是变化的数据最终需要记录到表单中，工具本身的同步是基于表单中数据的轮询，本质还是批处理方式，无法做到真正的数据实时同步。

综上，现阶段的技术选型工具，在数据 ETL 和实时同步时，不但有很大的运行隐患，而且做不到数据真正的实时处理和传输，无法为客户提供优质的资讯服务，成为了券商提升自身竞争力的桎梏。

2 大数据技术发展

“大数据”随着信息技术的发展逐渐走近并深入日常生活中，其含义主要为两方面。一是指

信息社会产生的海量业务数据，二是指用于处理海量数据的技术。两者相辅相成，数据量的增长促进相关技术的发展，技术的发展使得数据得到更有效的应用。随着近些年的发展，业内逐步沉淀出较多完善、优质的大数据架构 [3]。

相较于互联网科技领域，金融行业大数据技术发展较为缓慢，鲜有成型且特有的技术架构。行业内应用较为广泛的技术来源主要为开源社区、互联网领域成熟商业产品等。虽然金融领域数据结构与其他领域存在明显差异，但通用的大数据技术在金融数据上仍可表现出一定的优势。

2.1 变更数据捕获

广义上来说，日志、网络报文、数据库等数据的变更捕获相关技术都可以称为 Change Data Capture（CDC）[4]。本文关注的主要是数据库层面的数据变更捕获技术，其主要应用在数据采集、数据同步、分发等场景。目前主流的技术实现可大致分为基于 SQL 查询和基于日志两种方案。

其中，基于 SQL 查询的 CDC 工具实现机制为通过数据库语句直接查询数据表，依据主键、时间戳进行扫描，捕获数据表中变化情况。针对不同类型的数据库，只需包含主键和时间戳即可实施，但捕捉到的内容并不是数据表中发生的全部变化，而是自上次扫描后所有发生插入、更新的数据行。该实现方式有一定的局限性，如某数据行一段时间内发生多次更新时，只能捕获到最新的数据情况，变化细节无法获取，数据行删除操作也无法被获取。针对上述的不足之处，一种基于触发器的实现机制被提出。该方案在第一章中有所涉及，是我们前期使用的技术方案，该方案通过在拟捕获变更的数据表上建立触发器，当数据表发生增删改时，执行触发动作，将对应的操作记录到一张变更记录表中，再由扫描程序通过主键、时间戳从该记录表中获取增量的变更数据，从而最终实现对数据表的完整捕获。整体来看，基于 SQL 查询的 CDC 工具实现简单，相关方案可适用于多种类型数据库。同时，如上文所述，该方案存在数据时效性不高、消耗数据库运算资源、无法保障数据一致性等缺点。

基于日志的 CDC 工具在一定程度上解决了基于 SQL 查询机制上存在的问题，采集程序利用数据库日志，如通过 MySQL binlog，获取数据变更情况，避免使用触发器和查询语句造成数据库性能的消耗。同时，通过数据库日志可完整获取到增删改操作内容。但实际设计过程中，因各种类型数据库记录日志的实现方式有一定差别，所以需要单独设计针对不同格式日志的抓取、解析方式。

通过对现有成熟的 CDC 工具进行梳理，如图 2-1 所示，经过对比可以发现各种工具在数据采集、数据同步、分发等场景上的功能支持程度。其中，Kettle 为公司之前主要应用的数据同步迁移工具。该工具发展时间较长，具有插件丰富、图形化开发、开源社区与商业化均有支持等优势。在实际应用过程中，做数据全量同步任务的同时可应对复杂的数据处理需求。但随着资讯数据量的增长，其架构在处理数据量、实时性上出现瓶颈。DataX 相较于 Kettle 对数据库压力比较小，更善于做数据同步，但过程中的数据清洗、转换能力较弱，且其数据同步方案为离线调度，实时性要求比较高的场景很难胜任。Oracle Golden gate 和 Canal 是基于日志的 CDC 机制的具体实现，分别为 MYSQL 和 oracle 提供侵入性低、时效性高的数据抓取方式。同时，由于 Canal 部署轻量，在轻度集成场景下得到广泛应用。与其他工具相比，Flink CDC 和 Attunity Replicate（AR）的部署及使用显得更复杂。Flink CDC 需要依托 Flink 实时计算引擎。AR 作为商业数据同步软件的佼佼者，功能全面的同时，自然也需要较为复杂的配置流程。但用户大规模的应用，往往更关注功能上稳定与全面。因此，有开源社区大力支持的 Flink CDC 和成熟的商业产品 AR 受到了更多技术框架变革者的关注。

	Flink CDC	DataX	Canal	Attunity Replicate	Kettle	Oracle Goldengate
CDC机制	日志	查询	日志	日志	查询	日志
增量同步	√	×	√	√	×	√
断点续传	√	×	√	√	×	√
全量同步	√	√	×	√	√	√
全量+增量	√	×	×	√	×	√
架构	分布式	单机	单机	单机	分布式	分布式
支持类型	▲▲	▲▲▲	▲	▲▲▲▲	▲▲	▲

图 2-1：CDC 工具对比

2.2 大数据处理

实际应用过程中，某些场景仅通过 CDC 工具的同步、分发是不能满足的，需要在 CDC 的基础上具有进一步的数据处理能力。上节介绍的 DataX、AR、Kettle 等都具备一定的数据清洗、处理能力。其中，Kettle 的处理能力最为强大。公司最初的架构也是以其为核心输出数据处理服务能力。但随着数据量的增长以及业务逻辑复杂性的提高，Kettle 架构的时效性低、灵活性差，大数据量情况下处理效率不足等缺点日趋明显。相比之下，近些年发展迅速的大数据计算引擎，如 Spark、Flink 等，具有分布式架构、开发 API 丰富、批流一体、与 CDC 工具解耦等特点，使得在大数据量场景下数据处理的低延迟、高效率、便捷开发成为可能。

Spark 和 Flink 同为大数据计算引擎经常被拿来比较 [5]。流处理能力上，两者最重要的区别是 Spark Streaming 的 Micro Batching 模式和 Flink 的 Native Streaming 模式。Micro Batching 模式如图 2-2 所示，在该模式下流处理为批处理的特例，即流计算的本质也是批处理计算，数据处理延迟取决于批的划分是否足够小。与之相对，Native Streaming 模式如图 2-3 所示，该模式认为批处理是流处理的特例，该定义更加贴近流的概念，数据进入即处理，故该模式占据了流式计算领域“低延迟”的核心 [6]。两者均有强大的开源社区做支持，相关的生态系统也越来越完善。现阶段来看，Flink 在流式计算上表现更优，Spark 更善于批处理任务。随着用户对批流一体强烈的需求，两者均向着具有完备的批处理和流处理能力发展。

综上，CDC 工具和大数据处理引擎的应用场景广泛，且可供选择的框架也视应用场景各有

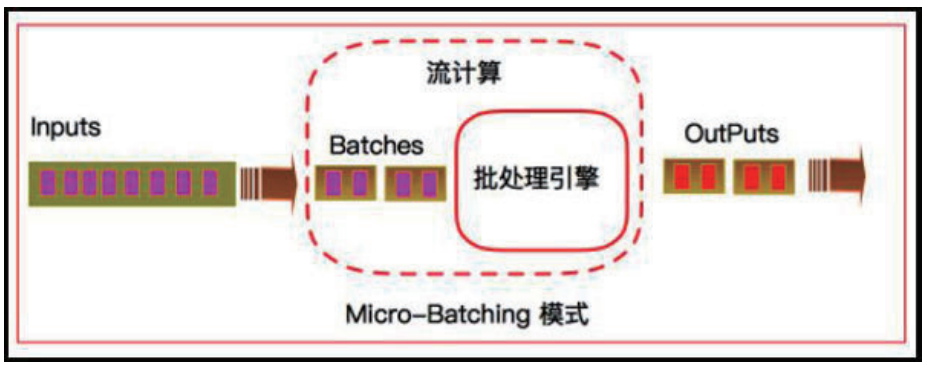


图 2-2：Spark Streaming 核心

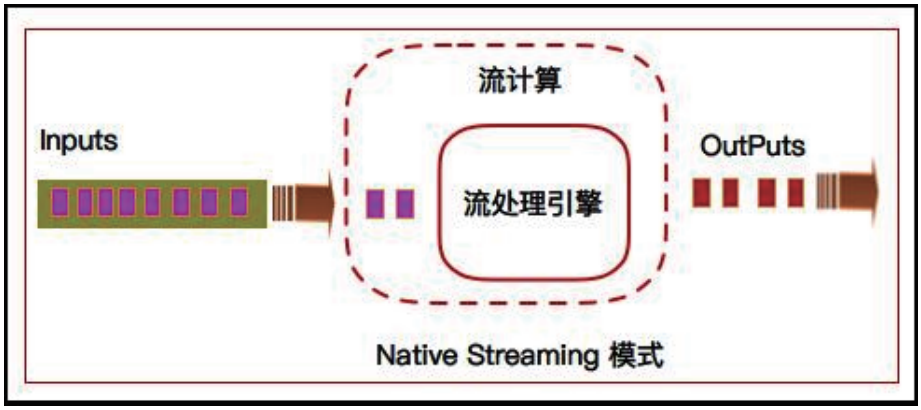


图 2-3：Flink 核心

侧重。我们的技术选型也紧贴通用、成熟、稳定的技术架构，针对金融资讯数据处理特点构建数据捕获和处理的框架。

3 资讯数据处理架构演进

随着金融行业的发展，证券市场逐渐活跃，越来越多的融资者与投资者参与到市场交易中，行情瞬息万变，伴随着市场中的活动所产生的资讯数据也日益增多。资讯数据中所蕴含的价值对市场中各参与方都显得尤为重要，每一步决策都离不开数据的支持。面对日渐庞杂的资讯数据，仅仅靠人工处理的方式已无法应付，如何及时获取并从中挖掘出高质量的资讯内容成了各参与方角力的新赛道。公司在做好资讯管理工作，维护资讯数据稳定运行的同时，关于如何做好资讯数据整合并向业务人员和客户实时且高质量的输出市场信息，是我们亟需进一步攻克的问题。为此，需要在原有数据处理架构的基础上，结合实际应用过程进行有针对性的调整，从而提供给用户更优质的资讯服务。

3.1 早期架构

早期系统架构如图 3-1 所示，资讯中心库是

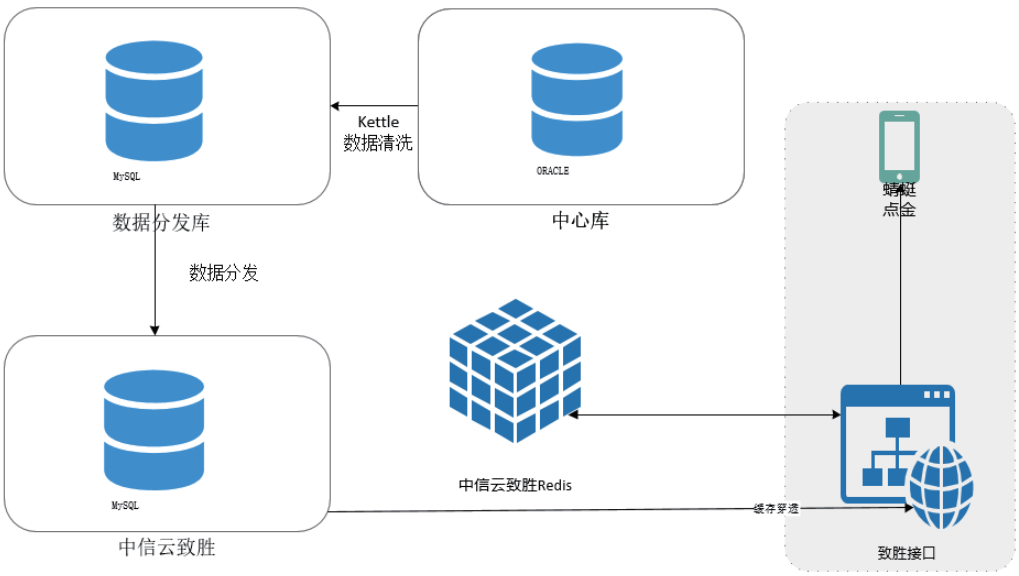


图 3-1：早期架构图

上游资讯数据的整合，为多源数据设计了统一的金融模型，使数据更加方便使用。当用户提出业务需求时，使用 Kettle 开发相关业务逻辑，并通过周期调度任务将清洗好的数据导入到内网 IDC 数据库，进而使用数据分发工具 DPS（一种基于触发器和表扫描的数据同步工具）将数据同步到云端的数据库，以便下游使用。资讯数据通常以接口提供服务，在用户请求第一次到达时，读取云端数据库数据，并将结果缓存在 Redis 中，Redis 作为纯内存存储，可提高接口的读取性能。Redis 中的数据会设置一定的过期时间，到期后数据自动清除。若请求再次到达，会穿透缓存直接读取数据库，从而获取到最新版本数据。

此方案采用了被动缓存的方式，无法实时读取最近更新的数据。该架构存在两个较为突出的问题。一是基于触发器和表扫描的数据同步工具消耗较多数据库资源，数据量不大时尚可，但在现阶段资讯大数据背景下，数据库资源弥足珍贵，大量的触发器和同步扫描语句严重影响了数据库性能。二是资讯数据所蕴含的信息量随时间减少，由被动缓存的方式造成的信息量损失不能被用户所接受。

3.2 半实时化架构

为解决图 3-1 架构存在的性能损失和时效性问题，我们有针对性的对链路中关键节点做了优化如图 3-2。首先是由被动写缓存改为主动写缓存。为了及时获取数据库变化情况，起初从众多 CDC 工具中选取部署轻量、集成方便、无需其他组件的 Canal 来实施，通过捕获云端数据库的变更，开发人员编写 java 程序实时读取订阅 Canal 的数据，将数据解析组合为业务所需的数据格式，然后写入 Redis 中，下游用户在使用相关接口时即可获取最新的资讯数据，无需再等待数据被动过期。方案运行一段时间之后，我们发现主动写缓存过程逐渐成为资讯接口服务中最重要的一环，但 Canal 作为开源工具，功能还在不断的完善过程中，如程序监控、告警等需要单独开发实现，稳定性和高可用方面也略显不足。因此，我们将视野投向了更为成熟稳定的商业化软件

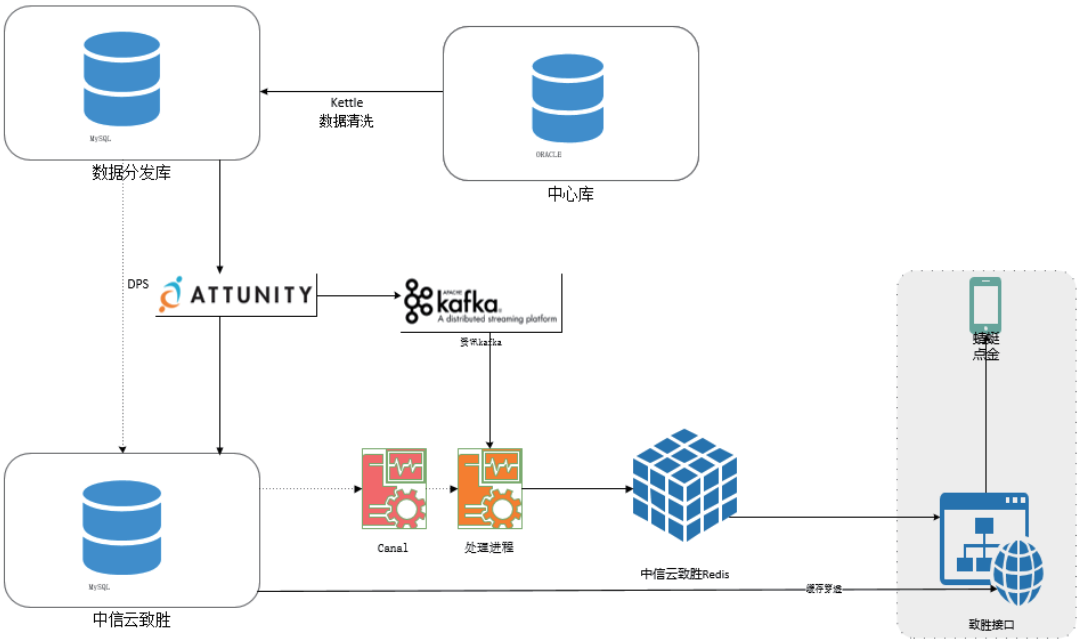


图 3-2：半实时化架构

AR，AR 的 CDC 原理与 Canal 相同，且具有完备的异常监控和恢复能力，图形化的操作页面减轻了运维压力。AR 将捕获到的数据库变更数据缓冲到 Kafka，避免数据堆积产生影响。下游的主动写缓存程序直接消费 Kafka 数据，基本解决了主动写缓存问题。同时，AR 还具有数据分发能力，用其替换掉原有的基于触发器和表扫描方式的 DPS 工具，很大程度上降低了数据库性能消耗，使得数据库在应付大量资讯数据需求时能有更好的表现。

至此，资讯数据的服务能力在时效性和质量上得到很大提升，整个架构的瓶颈点也由数据捕获和数据分发转移到数据处理能力上。虽然 AR 具备一定的数据处理能力，但相较于原架构中的 Kettle，其处理能力显得尤为薄弱，故将 Kettle 作为整个架构中唯一的数据处理能力输出工具。正如本文第二章所论述的，面对日渐增长的资讯数据量，Kettle 陈旧的架构略显乏力。同时，Kettle 作为数据处理工具并不具备输出数据到 Redis 的能力，一部分数据处理逻辑分散到 java 开发的处理进程中，导致相同的数据生成逻辑在 Kettle 和处理进程中各做一遍，维护不方便且存在潜在的云数据库与 Redis 数据不一致的风险。

3.3 全实时化架构

针对图 3-2 架构中的数据处理问题，我们经过调研和实践，最终确认了如下图 3-3 所示的资讯数据处理架构，重点在数据链路和数据处理环节做了调整。数据处理方面，选用开源成熟的分布式流处理框架 Flink 作为整个架构的核心，在有高时效性要求的资讯数据处理场景完全替代 Kettle。Flink 作为流式计算引擎，天然的与 Kafka 集成，除了可以处理 AR 在资讯中心库中捕获的增量数据，还可以无缝对接具有直接输出消息到 Kafka 能力的系统，如新闻标签系统。数据处理方面，原有的 Kettle 和写缓存程序中的相同业务逻辑统一由 Flink SQL 实现，使整个资讯数据处理过程做到了集中管理。Flink 下游则是将处理后的结果，输出到 Redis 和云数据库中，省去原有 IDC 库环节，缩短了链路，在一定程度上节约了数据传输时间。同时，输出端支持多目标库写入，保证了数据一致性。整体来看，分布式、高可用架构贯穿了上中下游，使得资讯服务能力可以更稳定、高效的输出。资讯数据定义宽泛，资讯来源也不统一，Flink 丰富的连接器也可支持数据源和目的端的进一步扩展。当业务场景越来越多时，可依托当前

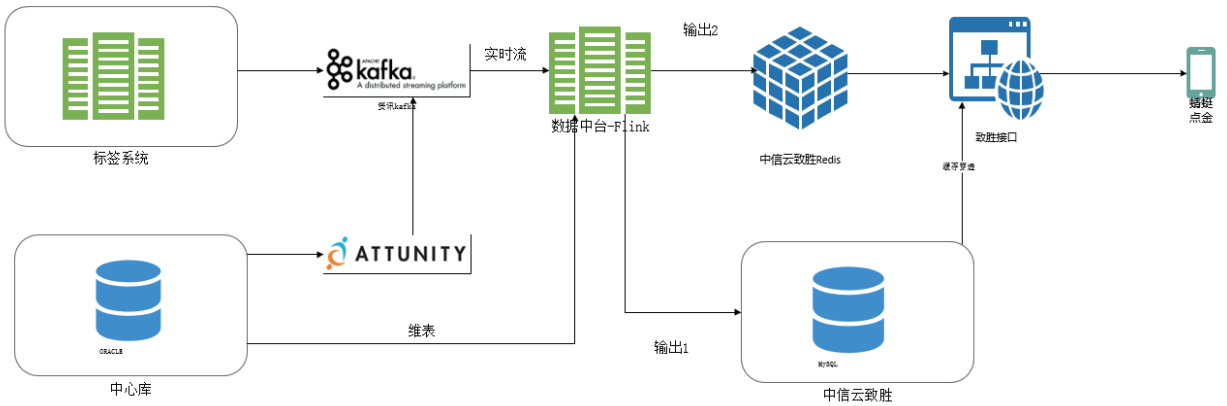


图 3-3：全实时化架构

架构构建集中的管理平台。

纵观资讯处理架构变更的几个阶段，可以看到大数据理念在资讯场景中的逐步使用和落地。实践过程中，对大数据产品有了更深入的体验和对比。从中可以发现，随着数据量的增加，功能复杂的产品往往更容易碰到瓶颈，将框架中所需的能力拆分成数据变更捕获、数据处理、数据分发等子模块，选择成熟、稳定、先进的专注于某一模块的技术，分而治之，最终形成的全局框架具备承载更大业务量的能力。

4 总结与展望

在大数据流处理技术与金融资讯领域结合方式探索的过程中，经过多次的尝试和架构演进，逐步形成了金融资讯实时处理框架。该框架为公司业务人员和客户提供了实时性更高、质量更强的资讯内容。资讯实时性的提高意味着用户可以第一时间掌握金融市场动态，及时做出决策，而资讯质量的提高则辅助用户做出更优的决策。数据与技术发展相辅相成，同样，系统架构也与业务发展有着千丝万缕的联系，系统架构升级的驱动力也是源于业务对资讯全方位处理能力要求的提高，业务发展也因技术服务水平的提高而有了拓展更多场景的可能。

当然，现有架构仍存有待优化的问题。数据处理方面，目前主要专注于实时资讯处理，而离线计算能力较为薄弱。随着批流一体的概念被提出，如何依托现有框架打造批流一体化平台成为下一阶段待解决的问题。同时，资讯内容质量方面，虽然架构在演进过程中数据处理和表达能力得到提高，但其中仍欠缺资讯数据质量管控模块，如何利用交叉检验等技术实现资讯数据自动化质量监测及预警也是下一步的重点研究目标。

参考文献：

[1] 韩维蜜 . 中国证券业信息化发展之路 [J]. 金融电子化 ,2019(10):84-85.

[2] 于鹏 , 刘绍晖 , 乐剑平 . 金融科技助力证券公司资讯服务智能化 [A]. 中国证券业协会 . 创新与发展：中国证券业 2017 年论文集 [C].: 中国证券业协会 ,2018:9.

[3] 陈永坚 . 大数据技术与金融行业的深度融合研究 [J]. 中国商论 ,2020(04):68-69.

[4] Andreakis A , Papapanagiotou I . DBLog: A Watermark Based Change-Data-Capture Framework[J].2020.

[5] Chintapalli S , Dagit D , Evans B , et al. Benchmarking Streaming Computation Engines: Storm, Flink and Spark Streaming[C]// IEEE International Parallel & Distributed Processing Symposium Workshops. IEEE, 2016.

[6] Carbone P , Katsifodimos A , Kth, et al. Apache flink : Stream and batch processing in a single engine. 2015.

实践探索

AIOPS语义级日志异常检测在证券行业的探索与实践

李进武[1]、刘博[1]、杨兵[1]、王东[1]、罗秋清[1]、郑铁樵[2]、胡小荣[2]/
[1] 海通证券股份有限公司 上海 200120
[2] 云智慧（北京）科技有限公司 北京 100089
E-mail：ljw11803@haitong.com

在 IT 运维领域，程序运行日志历来是生产故障检测和定位的重要抓手之一，但是由于程序日志具有较大的灵活性和差异性，传统的日志解析、关键字匹配等监控方式存在较大的局限性。针对这一痛点，本文通过语义理解、模式识别等多种技术手段相结合的方式提出了一种智能日志异常检测方案，并且在实际生产运行环境中对该算法的落地效果进行了探索和检验。

关键字：智能运维；日志异常检测；语义分析；智能预警

1 概述

随着证券行业信息化水平的逐步提升，安全生产保障能力已成为各金融机构科技能力建设工作的重中之重。经过行业多年的探索与实践，以数据为基础、场景为导向、算法为支撑的智能运维技术已逐步成为业务运维的新方向，在众多场景中，又以提升安全生产保障能力的智能预警技术尤为受到关注，其中日志作为公认的运维基本观测项（指标、日志、调用链）中信息含量最大的一种观测项（图 1），已经成为智能预警场景下最具挑战和最具研究价值的基本研究问题之一。

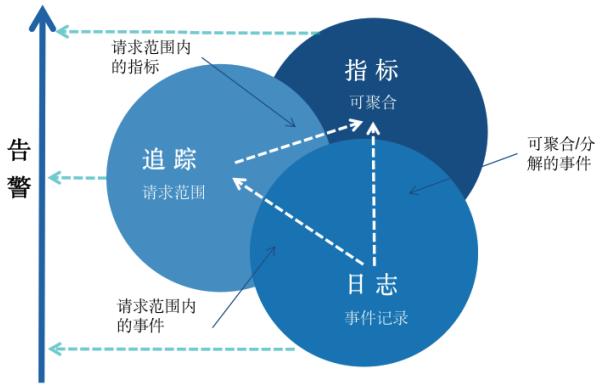


图 1：运维基本观测项

本文针对传统日志监控中存在的依赖人工经验、修复成本高等痛点，探索了一种面向证券行

业基于 AIOPS 的语义级日志异常检测方案，使用语义理解、模式识别、异常检测等多种技术手段，主动识别生产安全隐患，提高故障诊断的效率和准确性，提升证券行业生产安全保障能力。

2 日志异常检测的现状和痛点

日志记录了应用软件运行时的详细信息，蕴含着丰富的系统信息，在 IT 运维中扮演着极其重要的角色。目前针对日志的异常检测方法可分为两类：基于有监督的异常检测和基于无监督的异常检测。基于有监督的异常检测方法存在以下两个问题：

1）依赖人工经验，需要人工对训练结果进行标注、反馈等，工作量比较大，且对于首次发生的问题难以有效检测。

2）出现程序质量类问题之后，经过修复一般会得以解决，但事后花费大量精力分析并添加告警规则的效果不佳。

从海通证券的历史事件统计来看，目前的主要痛点是由程序质量问题产生且首次发生的生产事件。早期证券行业的 IT 投入较少，系统大多以外购为主，这导致一些遗留的应用系统无法对日志进行结构化修改，程序一旦产生问题，需要大量人力和时间去排查和处理。另外，证券行业应用系统的日志也具有鲜明的行业特点，如夜间期间跑批、节假日休市等等。

3 证券行业日志异常检测的方案

该方案总体基于 AIOPS 的语义级日志异常检测并结合传统日志监控普适痛点与证券行业特点。首先基于语义理解的技术对原始日志信息进行富集，扩展模型对日志信息的感知，增强异常检测模型对异常结果进行分析的能力。然后基于 Drain 类算法进行改进，对日志模板信息进行自动解析，识别出日志的模板，在保证准确性的同时，保证解析的效率。在模式解析基础上，利用基于指标的异常检测方法对日志的模式分布变化进行监控。整体方案如图 2 所示。

方案中异常检测为本次研究方案的核心，包括基于自然语言处理的语义理解技术、基于无监督算法的模式解析技术与时序数据异常检测技术，异常检测流程为：日志输入 -》语义识别 —》模式解析 -》异常检测 -》结果输出，下面对方案中的核心进行详细介绍。

3.1 语义识别

语义识别在自然语言处理领域已经成为较为成熟的技术，如在英文最具权威的 SQuAD2.0 数

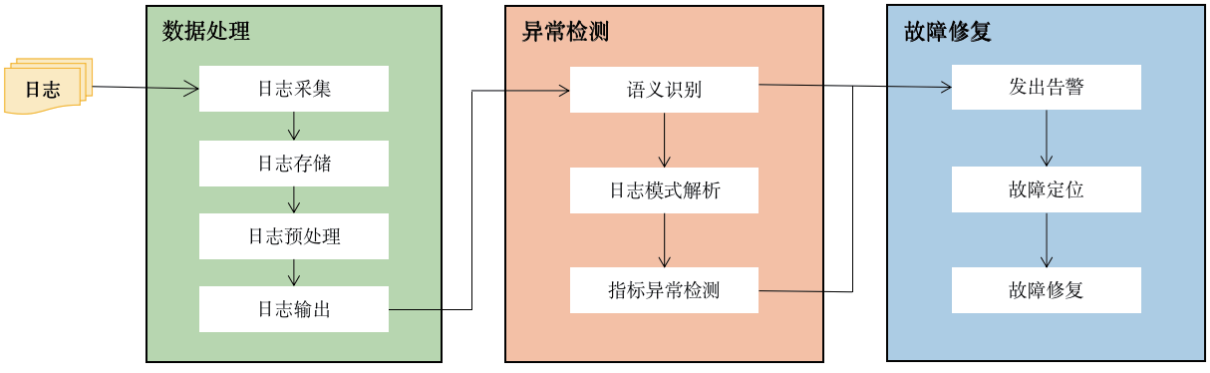


图 2：证券行业日志异常检测整体方案

据集中：EM 和 F1 两个指标上，人类的表现分类为 86.831 和 89.452，而目前学术界所研发的模型已取得了 EM 值 88.592，F1 值 90.859 的表现（均已超过人类水平），但是将语义识别技术应用于日志尚属首次。

本文将日志的语义信息理解和识别抽象为两阶段的日志语义异常检测问题，第一阶段检测日志语义中是否包含异常信息，第二阶段检测日志中包含的异常类型。针对第一阶段日志语义中是否包含异常信息的问题，采用机器学习与深度学习中的二分类算法（如 SVM、Random Forest、Bert 等），训练模型，将日志分为正常日志与错误日志。针对第二阶段日志异常语义中包含哪种类型异常的问题，依据智能运维业务场景中语义类别进行梳理，将日志中包含的异常类型分为网络异常、数据库异常等类型。然后采用机器学习与深度学习中的多分类算法（如 Random Forest、BERT 等），训练模型，判断日志语义中包含的异常类型。同时，利用日志来源检测模型对日志、异常的来源进行理解，扩展异常检测模型对日志的感知能力。日志语义异常检测整体流程如图 3 所示：

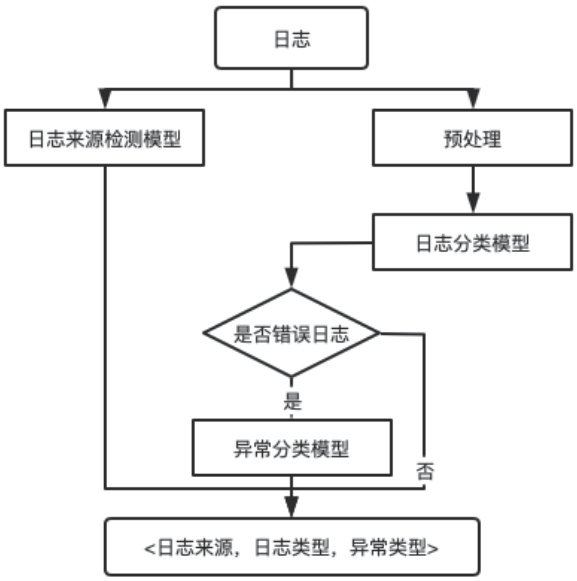


图 3：日志语义异常检测整体方案

3.1.1 日志来源检测

越成熟的系统 / 中间件 / 服务，其日志格式与描述越统一。因此，可针对不同来源的日志进行分析，总结其日志格式，并提取正则表达式，为每一个来源的日志设置一个日志格式，根据格式匹配检测日志来源。针对日志语义异常数据集中不同开源的日志进行格式分析，得到日志来源匹配规则，部分结果如表 1 所示：

表 1：日志来源检测结果样例

来源	日志样例	日志格式
clickhouse	2021.03.15 11:50:18.947333 [13036] {} <Error> Application: DB::Exception: Effective user of the process (root) does not match the owner of the data (test). Run under 'sudo -u test'.	Timestamp [pid] {} <log_level> log_msg
hadoop	2015-10-17 15:38:08,364 INFO [main] org.apache.hadoop.mapred.MapTask: Processing split: hdfs://msra-sa-41:9000/pageinput2.txt:1073741824+134217728	Timestamp log_level [process] log_msg
kafka	[2020-03-11 10:11:08,029] INFO Socket connection established to tsb-hw-prod-ckkafka2/10.20.3.197:2181, initiating session (org.apache.zookeeper.ClientCnxn)	[timestamp] log_level [kafka_str] log_msg (process)

3.1.2 预处理

证券领域日志中包含特殊的命名实体如时间 戳、URL、IP、File、Path、Number、Email等，通过对日志语义异常检测数据集中不同实体的标注分析，构建通用正则表达式进行日志领域的命名实体识别。在命名实体识别的基础上，对日志数据进行分词与停用词过滤，中文分词采用 Jieba 等分词工具，英文分词利用空格等符号进行切分。与其他 NLP 任务类似，基于语义异常的日志分析方法需要首先对日志进行向量化表示，考虑到日志中的特殊表示与命名实体会导致利用通用词向量库进行日志向量化过程中产生 OOV 问题，因此本文采用通用语料、运维领域通用语料、系统 / 中间件日志数据集与证券领域业务日志数据构造语料集，利用 Word2Vector 训练运维领域词向量库，对日志数据进行向量化表示。

3.1.3 日志分类模型

通常日志中包含日志等级字段，如 debug,info,warning,error 等。通常可以利用日志等级字段对日志进行分类，但是实际中这种日志分类方式通常会存在两个问题：

1）有些系统 / 业务日志中并不包含日志等级字段，如 linux、mac、ntpd、proxifier、redis 等，无法用日志等级字段进行分类；

2）有些系统 / 业务日志中，日志等级字段标注不准确，或者将异常情况发生时的相关状态或情况标注为“error”等，实际上这类日志语义上并不包含错误信息。

因此，首先对错误日志与异常日志进行区分：异常日志是指发生异常时打印的日志，可能仅为异常发生时的某个状态或情况说明，本身并不包含错误信息；错误日志是指语义中包含错误信息的日志。本方案中，将日志分为正常日志与错误日志，即根据日志的语义信息将日志进行分类，然后分别利用传统机器学习二分类算法支持向量机（SVM）、集成学习算法随即森林（Random Forest）、深度学习算法 BERT 在标准日志标准数据集进行实验。

支持向量机（SVM）是一种二分类模型，它的基本模型是定义在特征空间上的间隔最大的线性分类器，SVM 学习的基本思想是求解能够正确划分训练数据集并且几何间隔最大的分离超平面。如图 4 所示，w · x+b=0 即为分离超平面，对于线性可分的数据集来说，这样的超平面有无穷多个（即感知机），但是几何间隔最大的分离超平面却是唯一的。SVM 的求解可以使用二次凸优化问题的数值方法，例如内点法和序列最小优化算法，在拥有充足学习样本时也可使用随机梯度下降。

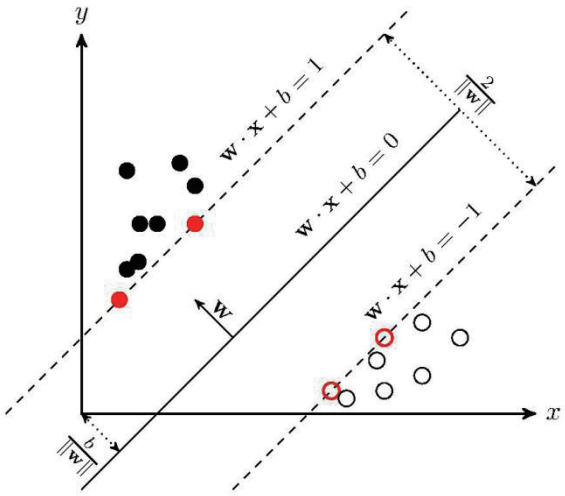


图 4：SVM 算法

随机森林是一种集成学习算法，集成学习的基本思想就是将多个分类器组合，从而实现一个预测效果更好的集成分类器。随机森林采用集成学习中 Bagging 的思想，即 a. 每次有放回地从训练集中取出 n 个训练样本，组成新的训练集；b. 利用新的训练集，训练得到 M 个子模型；c. 对于分类问题，采用投票的方法，得票最多子模型的分类类别为最终的类别；对于回归问题，采用简单的平均方法得到预测值。随机森林以决策树为基本单元，通过集成大量的决策树，就构成了随机森林。

BERT 是一个预训练的语言表征模型。它强

调了不再像以往一样采用传统的单向语言模型或者把两个单向语言模型进行浅层拼接的方法进行预训练，而是采用新的 masked language model（MLM），以致能生成深度的双向语言表征。以往的预训练模型的结构会受到单向语言模型（从左到右或者从右到左）的限制，因而也限制了模型的表征能力，使其只能获取单方向的上下文信息。而 BERT 利用 MLM 进行预训练并且采用深层的双向 Transformer 组件来构建整个模型，因此最终生成能融合左右上下文信息的深层双向语言表征。BERT 整体结构如图 5 所示。

利用分类算法经典评价指标准确率（Accuracy）、精确率 (Precision)、召回率 (Recall)、F1 值作为日志分类模型的评价指标。各指标计算方式如下：

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 = \frac{2*Precision*Recall}{Precision + Recall} \quad (4)$$

其中，TP 表示真阳性，即算法预测为正例 (P)，实际也为正例 (P) 的个数，即算法预测正确 (True)；TN 表示真阴性，即算法预测为负例 (N)，实际上也是负例 (N) 的个数，即算法预测正确 (True)；FP 表示假阳性，即算法预测为正例 (P)，实际为负例 (N) 的个数，即算法预测错误 (False)；FN 表示假阴性，即算法预测为负例 (F)，实际为正例 (P)

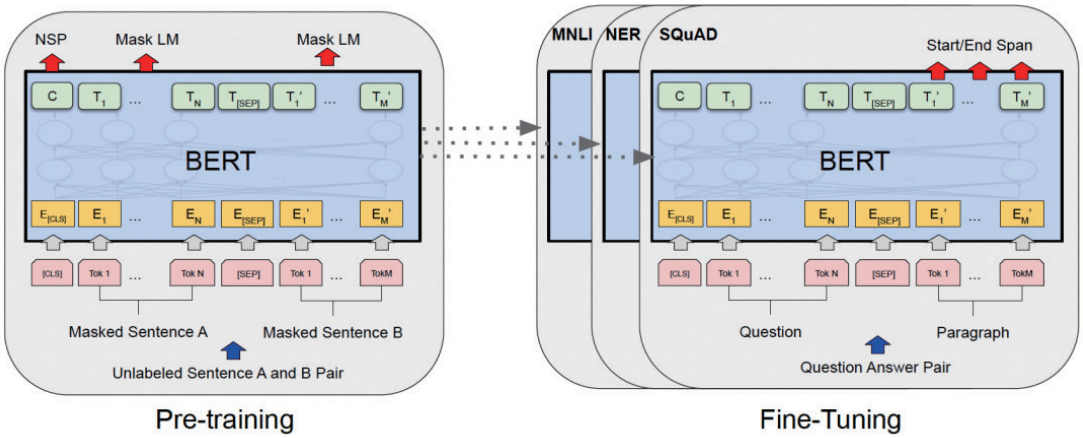


图 5：BERT 模型结构

表 2：日志分类算法实验结果

方法	accuracy	precision	recall	f1
支持向量机	0.962	0.964	0.983	0.973
随机森林	0.963	0.969	0.945	0.956
BERT	0.987	0.989	0.992	0.99

的个数，即算法预测错误(False)。

实验结果如表 2 所示。整体而言，三类算法的实验效果均达到人类水平，BERT 整体效果最佳，随机森林与支持向量机相比准确率与精确率效果更好，而支持向量机召回率更高。在实际应用中，BERT 模型训练对服务器硬件要求更高，且性能较差；而支持向量机泛化能力较随机森林相比较差，因此本方案最终采用随机森林算法进行日志分类。

3.1.4 异常分类模型

首先，对日志中包含的异常类型进行分析与总结，本方案将日志中包含的异常类型分为：文件 / 文件夹操作异常、网络异常、数据库异常、系统异常和其他异常等 5 类。与日志分类模型相同，选择随机森林（Random Forest）算法在日志标准数据集中进行多分类实验，结果如表 3 所示，可以看出该模型在各类异常检测中均取得较高准确率。

3.2 日志模式解析

经典日志模式解析算法有三种类型：基于频繁模式挖掘、基于机器学习和启发式的方法。基于频繁模式挖掘这一类日志解析算法需要对原始日志中的高频词汇进行统计，以此挖掘高频词的共现信息，生成模式，典型的算法以 SLCT(Simple Logfile Clustering Tool) 为代表。基于

表 3：异常分类算法实验结果

异常类型	类别	accuracy	precision	recall
文件/文件夹操作异常	1	1.00	0.67	0.80
网络异常	2	0.99	0.97	0.98
数据库异常	3	0.98	0.97	0.97
系统异常	4	0.99	0.84	0.91
其他异常	5	0.88	0.99	0.93

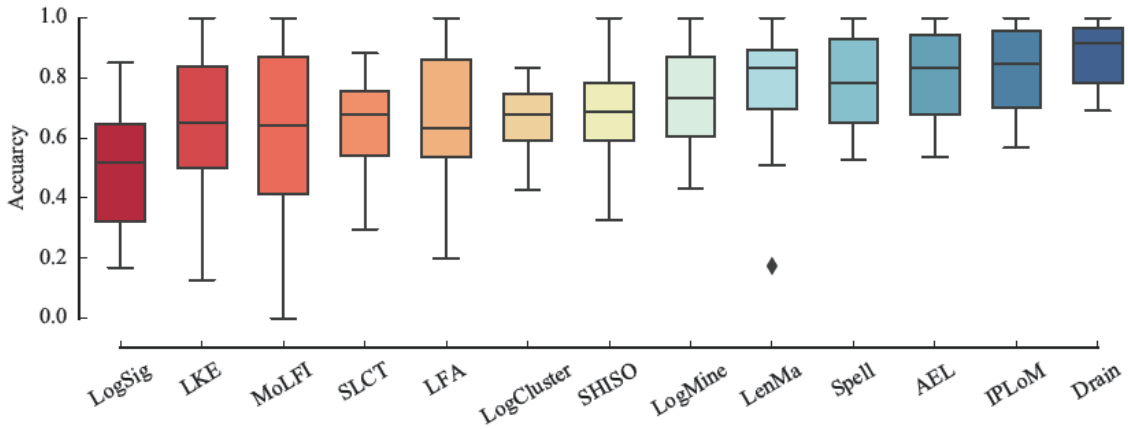


图 6：各日志解析算法准确性对比

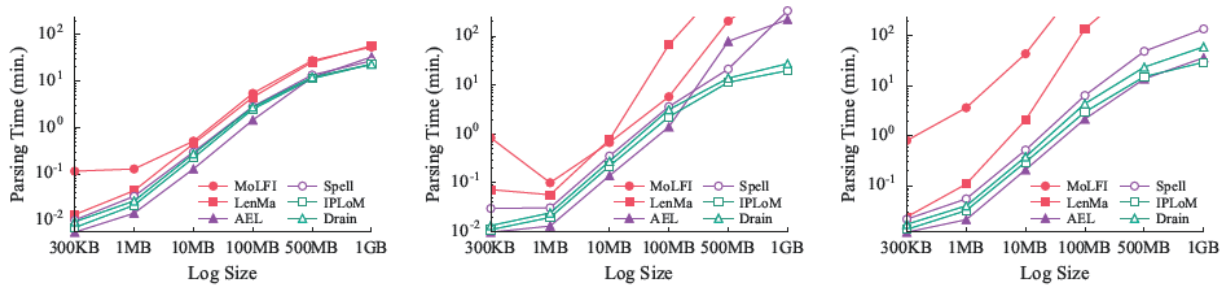


图 7：各日志解析算法效率对比

机器学习的方法，利用聚类的日志信息进行模板提取，典型的以 LKE 为代表，其利用字符编辑距离为日志距离的度量指标对日志进行聚类。启发式的日志解析算法以 Drain 为代表，充分利用日志的特点，对日志进行划分和组合。各日志解析算法准确性（图 6）与效率（图 7）对比如下图。其中 Drain 算法的准确性中位数在所有算法中最高。

Drain 算法是目前国内外工业界使用最为广泛的模式解析算法，该算法的基本假设为：具有相同日志模式的原始日志消息具有相同的长度；具有相同日志模式的原始日志消息的前几个单词是相同的。基于以上的假设，利用树形数据结构，实现对日志的快速分组聚类，从聚类的结果中提取模板。该算法的优势是快速高效，同时对于分词良好的原始日志信息具有很好的适应性。其原理如图 8 所示。

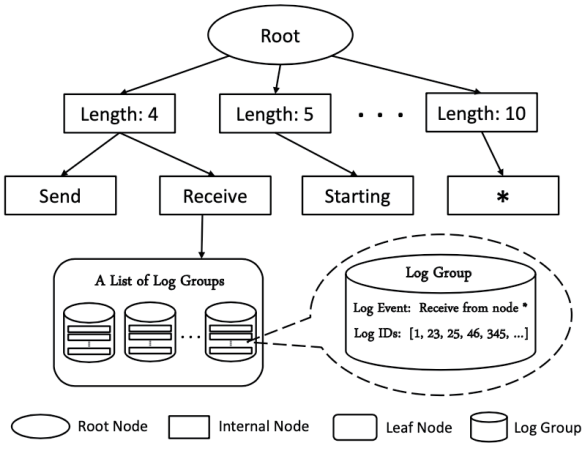


图 8：Drain 算法原理

然而针对证券领域的复杂日志该算法仍然存在以下缺陷：

1）算法结果和分词强相关，不当的分词可能会让相同模式的算法有不同的长度，从而被分到不同的模式中；

2）对长度强依赖，只有长度相同的日志才会被分到一起，而某些模式的日志长度不一定相同；

3）对树深长度参数敏感，树的深度反映的是前缀词的长度，所以若选定树的深度为 k，那么就会使得日志的前 k 个词（前缀词）为常量，且只要前缀词不同，则模式必然不同，而相同模式日志的前几个单词不一定相同。

针对上述中 Drain 算法在证券领域日志模式解析中存在的问题，本方案引入对不同长度的日志模式进行二次聚合，缓解 Drain 算法在不定长日志模式中的错误。有以下三处改进：

1）深度固定。从搜索到日志所属的模式角度来看，其深度为 2，搜索的值分别是日志长度及 split token，对应 Length Layer 与 Token Layer。

2）自适应相似度阈值。通过搜索得到对应的模式后，需要与其中的所有日志进行相似度比较，选择一个相似度最高且超过相似度阈值的日志，从而对比更新模式。相似度阈值随着模式的更新而更新。

3）merge 机制。merge 允许不同长度的模式进行合并，得到新的模式。达到相似度阈值的模式会进行合并，相似度算法为 LCS。

根据上述优化，本文提出了 LogSlaw 日志模

式解析算法，其流程图 9 所示：

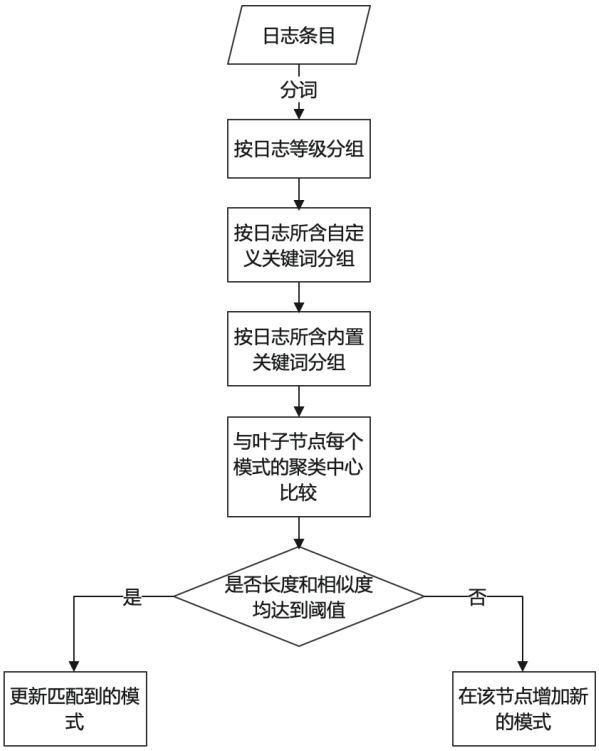


图 9：LogSlaw 算法原理

利用 LogSlaw 算法与 Drain 算法在标准数据集中进行测试，实验结果如表 4 所示。

可以看到，经过改进之后的 LogSlaw，可以取得不弱于 Drain 算法的效果，且不再强依赖于分词，对于不同长度的日志，也可以更为灵活的处理。且 LogSlaw 配置自定义关键词简单，加入自定义关键词配置的 LogSlaw，其效果更是要远远优于 Drain 算法。

3.3 时序异常检测

通过机器学习的方法识别出日志的模式后，将其转为实时时序信息，包括不限于日志模式计数时序、日志中所包含的参数分布等数据，采用相应的时序异常检测算法对日志模式进行监控与异常发现。可以对新模式进行检测，即日志出现了与历史上未曾出现的新模式时进行告警，可能反映了一些系统变更情况；同时基于历史上同一模式的数量分布情况进行对比，对于属于平稳性的模式数量分布，可以检测相应的突增与突降，对于周期性的业务模式数量分布可以发现器周期性的变化情况，及时发出告警信息提醒运维人员对异常时刻进行关注；还可以对不同模式之间的分布随时间变化差异情况进行监控，如某一时刻，每一日志系统其 top50 的日志模式的分布占比与历史 7 天内的平均分布占比有显著的差异，则发出告警。

表 4：Drain、LogSlaw 算法结果对比

数据源	原始日志数	标注模式数	LogSlaw			Drain		
			耗时 /s	F1	PA	耗时 /s	F1	PA
apache	52004	33	1.8941	0.9995	0.9689	1.212	1	1
arangodb	28146	12	1.9072	1	0.0027	0.9939	0.9999	0.0027
clickhouse	111710	74	11.6607	0.9779	0.6307	3.4448	0.9823	0.8381
hadoop	170223	297	32.4266	0.9909	0.8355	6.579	0.9899	0.8742
kafka	33940	117	3.014	1	0.9909	1.0193	1	0.995
linux	336990	387	21.4908	0.6531	0.3985	3.7171	0.5428	0.3037
mac	102609	647	32.2322	0.9588	0.6877	3.483	0.9722	0.6638
openstack	207631	51	18.9309	0.8022	0.3821	11.6883	0.5714	0.3324
proxifier	21320	8	0.7415	0.9945	0.0252	0.5251	0.7475	0
redis	20390	15	0.5837	1	1	0.4321	1	1
ssh	655147	51	35.9585	0.95	0.5776	16.3284	0.9497	0.5795
tengine	128293	7	3.3923	1	0.9999	2.9722	1	0.9999
zookeeper	948859	109	61.2852	1	0.9988	24.2453	0.9999	0.9831

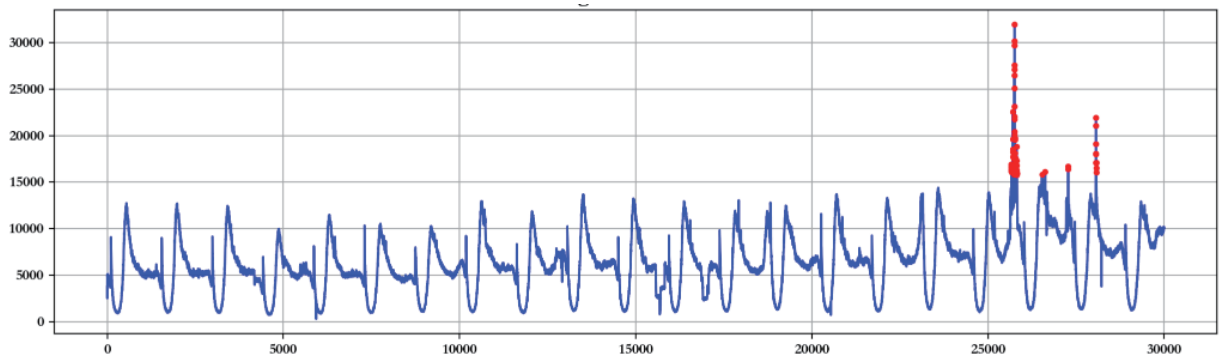


图 10：传统时序异常检测算法结果样例

时序异常检测算法中，较为通用的算法包括固定阈值、n-sigma 自动阈值等方式，如图 10 所示。

然而通用时序异常检测算法如固定阈值、自动阈值等误报率较高，同时券商行业的日志也具有鲜明的行业特点，如夜间期间跑批、节假日休市等，而通用时序异常检测算法并未考虑到这些行业特点。如图 11 为证券领域模式时序数据样例，该模式工作日与节假日模式并不相同，即不能利用全部历史数据计算其上下界，而利用固定阈值或 n-sigma 等算法则无法处理该问题。

因此，针对证券领域的日志数据特点，本方案提出了一种融合滑动窗口与层次权重模式的多场景日志异常检测方法（DW_LoADs）。首先经过同比日期选择器，输入模式参数，获得待检测点的同比日期序列；然后经过滑动窗口选择器，自定义窗口大小获得待检测点的同比时间点矩阵与对应的取值矩阵；再利用层次权重模式对待检测点的同比取值序列进行加权平均，得到最终的历史同比取值序列；最后利用 n-sigma 进行异常检测，整体流程如图 12 所示。

该算法的优势在于：

1）增加日历管理功能。含节假日和非节假日。节假日与非节假日的数据模式存在明显差异，因此将节假日的数据与节假日的数据对比，非节假日的数据与非节假日的数据进行对比，可以提高异常检测的准确率，减少误报。

2）丰富的场景异常识别。在时序异常检测算法的基础上融合了在场景领域的丰富的故障检测经验，包含了历史新增、时段新增、时段突增、时段突降等特别的异常检测算法。

3）异常自适应功能。某些异常在第一次发生时会给予特别的关注，但随着时间的推移，自

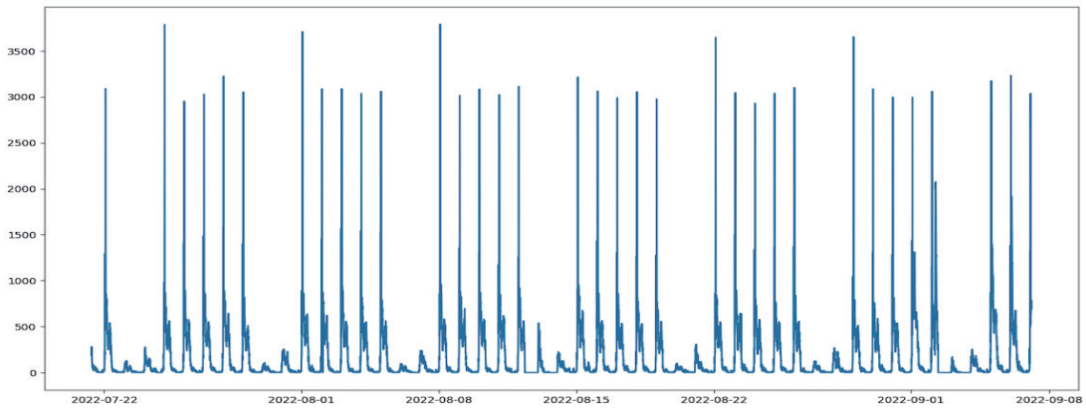


图 11：券商行业日志时序样例

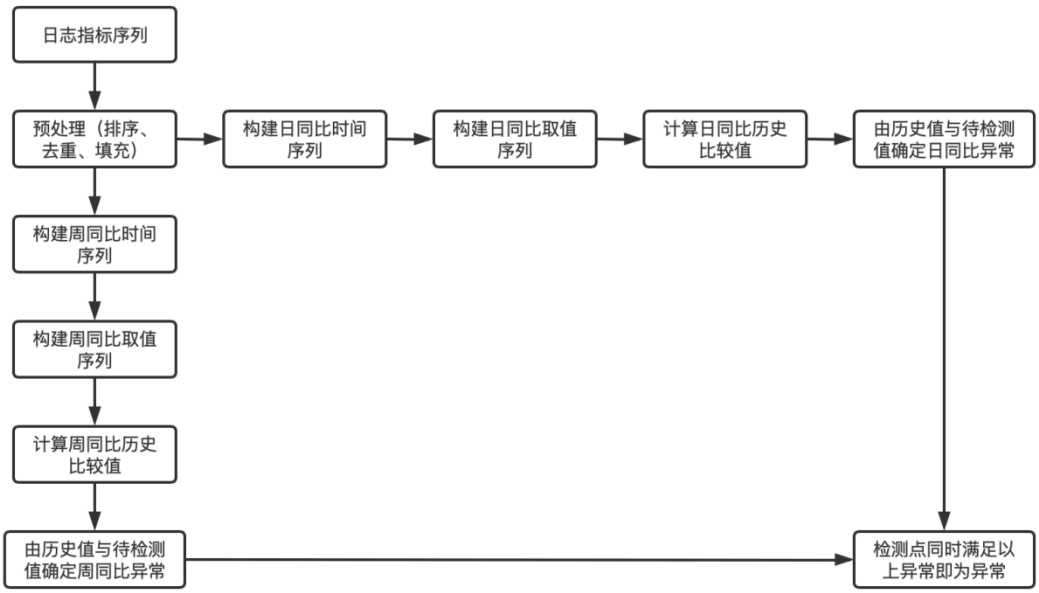


图 12：时序日志异常检测算法整体流程

适应的方法可以对重复出现的异常进行抑制。采用动态权重的方法对历史的数值进行修正，包括：a. 增加滑动窗口，当前点的数据对和历史数据点前后 5 分钟数据进行计算（交易日对比交易日，非交易日对比非交易日），并且根据距离对应时间点的远近，设置动态的权重，以减轻单一时间点异常造成的影响。b. 加权平均计算同时间段的历史数据，距离当前时间点越近的日期赋予更高的权重，该策略会使相同模式的日志随着训练时长增加，异常点逐渐消失，增加了异常检测算法的学习能力。

4 日志异常检测的实践

该算法落地的系统架构如图 13 所示，通过在实践结果看，每天可实时分析 10TB+ 日志，故障告警的精确率和召回率达到 90%。同时可帮助技术人员第一时间感知系统异常波动，并通过分析结果快速定位到问题，减少业务中断几率、缩短业务中断时间，提升系统可用性。本方案业务实践价值主要体现在以下两个方面：

4.1 提升故障发现能力，覆盖传统手段难以检测的问题

传统日志监控场景普遍使用关键字规则进行监控，例如 shutdown，error 等。但是实际情况中，并非所有的异常情况均有类似错误关键字的输出，导致关键字监控方案失效。而本方案利用基于指标的异常检测方法对日志的模式分布变化进行分析，检测历史新增、时段新增、时段突增、时段突降等异常情况，不仅能够对传统的错误信息进行识别，还能够对业务的突发性变化进行有效的提示，极大的拓展了问题发现的能力。

4.2 告警压缩，极大降低运维人工作量，提高运维效率

传统模式下，日志监控利用“error”、“shutdown”等关键字在海量原始日志中进行搜索匹配，往往会产生大量的告警信息，导致运维人员难以逐一进行分析。而在本方案中，通过模式匹配的方式，通过智能算法对日志进行初步的模式分类，能够极大的压缩、屏蔽此类海量的告警信息。如下图，对比传统的关键字和规则匹配，

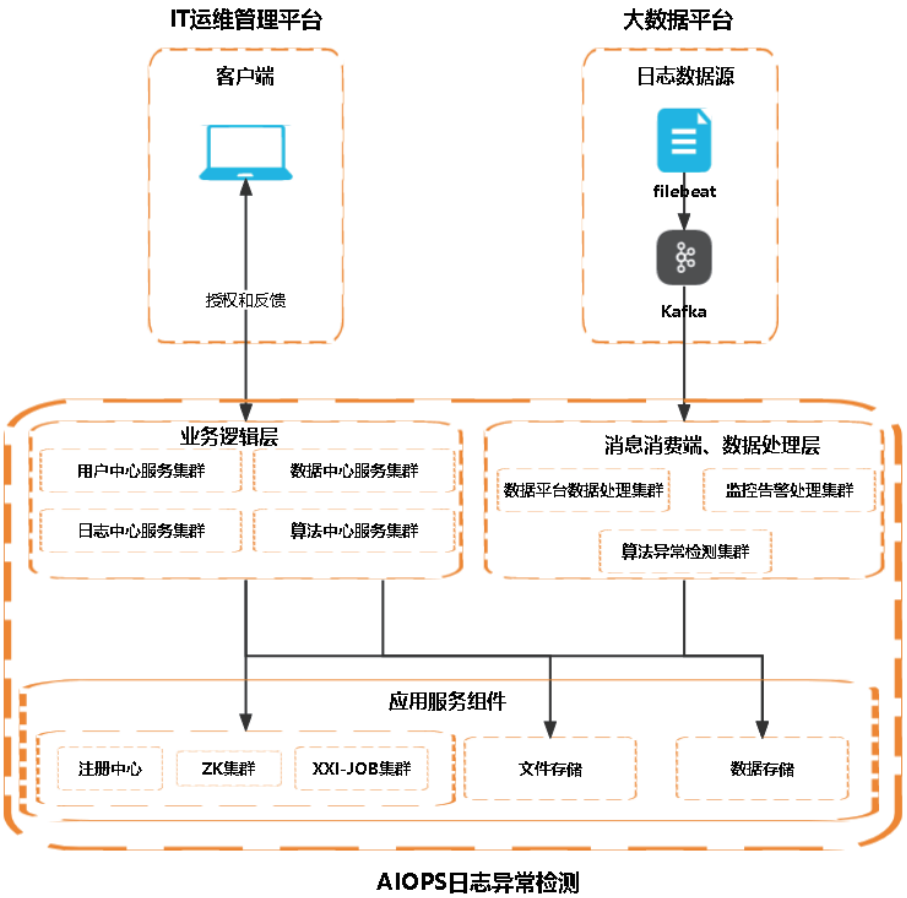


图 13：算法落地系统架构

可以看到同一系统同一时间段，关键字监控异常数量高达百万级（图 15），而时序日志异常检测输出仅为 17 条（图 16）。

通过选取 4 套业务系统，不同业务系统的日志数量、日志格式、日志结构复杂度均不相同，其模式解析结果如下（表 5）。

以业务系统 4 为例，原始 1848w 的日志数据解析为 73 个模式（图 17）。

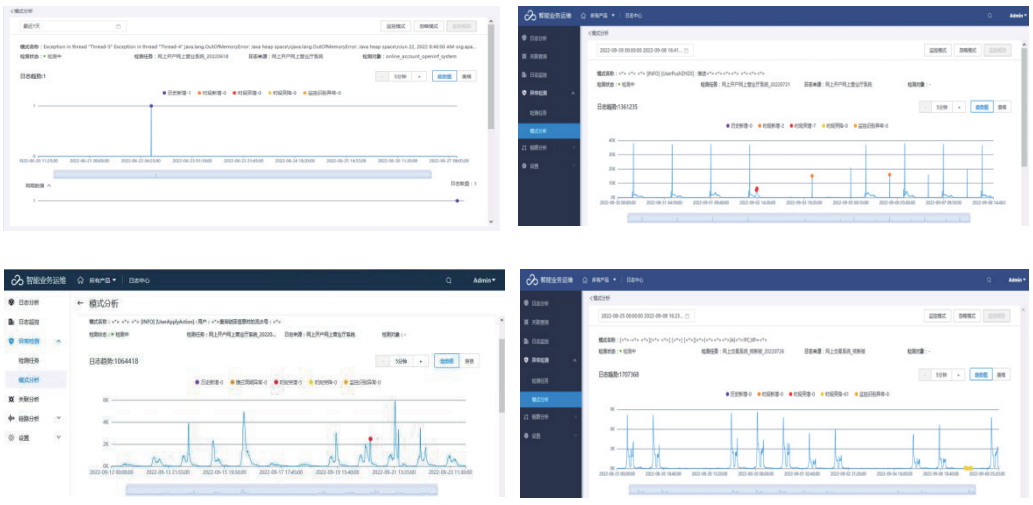


图 14：日志异常样例

图 15：传统关键字异常检测结果样例

图 16：时序日志异常检测结果样例

表 5：不同业务系统模式解析结果

应用系统	日志特点	日志数量(万条)/天	模式数量（7天）
业务系统 1	日志呈周期性、格式较为统一、日志描述字段复杂情况较少	200-300	44
业务系统 2	日志呈周期性、格式不统一、日志描述字段复杂情况较少	200-300	108
业务系统 3	日志呈周期性、格式较为统一、日志描述字段复杂情况较少	200-300	239
业务系统 4	日志呈周期性、格式较为统一、日志描述字段复杂情况较多	1500-2000	1090

5 总结展望

针对日志异常检测方案方法存在的问题，本文提出了一种语义级日志异常检测方案。该方案采用了基于机器学习的智能日志异常检测算法，自动实时对日志的模式进行解析和异常检测。通过在具有代表性的应用系统日志进行了实践，对传统有监督异常检测方法的准确性和效率做了对

图 17：日志模式解析结果样例

比。从实践结果看，该方案提高了故障诊断的效率和准确性，为业务定位问题提供了支撑，具有可推广性以及一定的参考价值，可普遍适用于证券行业应用系统的日志异常检测。

证券公司关于账号权限风险管理与稽核应用创新的探索和实践

吴哲锐[1]、杨怀宇[1]、韩啸[2]、高伟[1]/[1]民生证券股份有限公司 信息技术中心 上海 200120
[2]奇点浩翰数据技术（北京）有限公司 产品部 北京 100010
E-mail：yanghuaiyu@mszq.com

信息化建设在为证券公司带来巨大经济效益的同时，也对证券公司的管理、协调、运营和可持续发展提出了巨大的挑战。作为信息化管理的众多内容之一，信息系统用户业务授权及合规性控制管理是其中非常重要的一个管控环节，用户业务授权问题可能会造成高昂的损失，甚至会影响到企业的正常经营。用户业务授权管理关注的核心包括很多，如：用户在系统内所授予的授权是否合理（授权最小化、以岗定权、不相容业务操作是否分离）；系统中关键、敏感的操作或数据是否只能够被限定的人员所访问等。此外信息系统中用户业务授权相关缺陷或问题也是内外部审计机构关注的重点，因此如何确认用户业务授权的合规性，加强用户业务授权的管理，已是当前证券公司面临的重要管理课题与挑战。

关键词：权限管理；数据安全；风险防范

1 引言

随着我国数字化转型的加速推进，数据安全、个人信息保护的重要性愈发凸显。2021 年国家相继出台《数据安全法》、《个人信息保护法》、《关键信息基础设施安全保护条例》等法律法规。在证券行业，2021 年中国证监会发布《证券期货业“十四五”科技发展规划》，明确指出要“加强数据安全防护和审计要求”、“加强个人信息安全管理”、“设置分类分级数据访问权限”。而规范的信息系统账号权限管理正是做好数据安全和个人信息保护的关键环节。此外，2018 年证监会发布《证券基金经营机构信息技术管理办法》（第 152 号令），第三十二条明确提出关于账号管理“最少功能、最小权限分配原则”和“对信息系统权限的定期检查与核对机制，确保用户权限与其工作职责相匹配，防止出现授权不当的情形”的监管要求。2021 年证券行业违规案例中，有至少

3 起涉及信息系统账号权限管理问题遭受监管处罚。

基于国家加强数据安全、个人信息保护的指导思想，落实行业监管要求，结合公司信息系统的特点和技术调研分析，民生证券从 2021 年开始在应用系统权限的统一管理和自动化稽核进行了落地应用实践，截止至 2022 年 6 月，完成了全量业务系统权限数据的统一台账管理，建设了数据自动化对接、OA 系统自动授权、权限异动告警通知、时间维度和基线维度的权限自动化稽核、权限数据补充导入等功能，打破传统 API 接口对接模式，让任何第三方业务系统权限模型均能够快速实现与权限管理系统对接，提升了权限管理的准确性和时效性，有效的落实了权限内控管理要求。本文重点从需求痛点、建设思路、实践历程和后续规划等角度进行分享。

2 权限管理面临的问题

2.1 系统多、权限杂

随着业务的发展和科技的进步，民生证券拥有第三方厂商采购和自行研发的投资管理系统、量化评级系统、风控系统、RPA 系统、数据中台、投行业务系统、手机 APP、集中交易系统等各类应用系统百余套，但各系统的账号权限信息仅能在本系统中查询，并且每个业务系统的开发语言、系统架构、业务架构各不相同。系统的产品权限、附加权限、功能权限、菜单权限等，甚至下沉到业务本身，针对相同员工在不同系统内的角色划分、业务责任也不尽相同。在对员工权限管理、稽核等场景中，只能分别登录各自系统查询，无法形成集中的权限管理和权限稽核，这都无疑对权限稽核工作增加了难度。

2.2 权限无集中的台帐

因各个业务系统上线时间节点、负责部门、系统架构等因素的不一致，每个业务系统的权限情况各自独立，没有有效的数据对接和联动机制；从而导致当公司内部发生人事调动、任免等情况时，各业务系统的账号更新只能通过管理流程辅助手动操作进行联动，往往会出现“系统中账号权限情况不准确，权限变更结果无反馈”等问题。以上问题归因是未建立统一权限管理台帐，无法利用技术手段实现对公司权限数据的及时、准确稽核。

2.3 稽核工作人工占比高

当前大多数证券公司，乃至银行、基金、保险、期货、信托等其他金融类公司，基本都采用全人工方式对各业务系统的账号进行权限数据比对，不仅工作效率低、速度慢，而且往往无法保证数据的准确性，容易出错，进而导致在外部审计单位对系统权限问题进行审计时，常常作为观察项（或整改项）内容，严重时遭受处罚，对公司造成名誉和财产损失。

2.4 权限审计无抓手

依据监管要求，各机构定期开展信息系统用户权限检查与核对的审计工作，需要各系统管理员导出用户权限，同时结合公司人事信息对 IT 系统用户权限进行审计比对和抽检，通过人工进行；内外审计周期长、多部门人员参与配合审计工作，不同信息系统的权限管理模式不统一，未形成统一的管理规范；缺少对权限的动态管理工作机制，权限更新不及时；监督检查不方便，容易出现监控盲点；权限变更工作未留痕，发现风险隐患时难以追查责任；传统人工审计缺乏自动化工作抓手，难以及时发现权限风险，且工作效率低下。

2.5 被动式管理

稽核“已经发生的权限变更情况”是目前权限稽核工作绕不开的话题，权限管理应是前置于业务并且伴随员工账号全生命周期，但是目前稽

核到权限问题后，还需要花费大量时间对各应用系统权限进行修改配置，如何将 OA、人力系统与业务系统权限进行统一和权限管理联动是解决被动式管理的关键问题。

3 系统建设思路

权限稽核与管理是指信息技术安全管理与审计针对应用系统的用户权限、角色权限、菜单 / 产品权限、附加 / 其它权限进行稽核。主要功能需要包含基线管理、权限监控、审计报告与权限跟踪日志，以便提供未来作为审核、分析与管理之用。

3.1 构建统一台账

通过对权限管理监管要求和稽核工作模式的探索，做好权限稽核管理体系，首先要构建一套能够实现自动化、智能化、流程化的账号权限稽核、基线设立 / 比对、历史轨迹查询等功能的权限稽核系统。要完成权限稽核系统建设首先必须构建统一的账号权限管理台帐，通过技术手段对公司业务系统所有账号建立索引和关联关系，针对公司人事变化、权限匹配等问题，实现智能化的稽核比对。建立一套权限台账通用模型把分散在几十套系统的账户进行了账号映射和关联匹配，为准确有效的进行权限稽核提供了一整套的数据模型。

3.2 权限自动对接

每个业务系统的权限模型设计均不相同，系统的功能权限、菜单权限等，甚至下沉到业务本身，针对相同员工在不同系统内的角色划分、业务责任也不尽相同。例如简单通用系统仅通过账号、角色、菜单权限进行赋权管理，复杂权限模型的系统又要包括产品组合维度、时间管理维度的权限数据，权限稽核系统提供了用户自主配置权限输入参数可自动生成数据转换代码的功能，快速的完成权限数据的对接，充分提升了系统的自主可控能力。

3.3 权限异动告警

为了能让 IT 和业务部门及时发现用户权限风险（例如休假的员工登陆了业务系统，管理员要第一时间排查该员工的登录原因），需要有及时、准确的手段通知到管理员。告警需要支持通过邮件发送，并且需要对告警规则进行配置，确保告警信息能够及时准确的通知到管理员。

3.4 审计自动化

账号权限稽核系统的核心之一是权限自动化稽核比对，及时发现用户账号在各业务系统的权限差异及变化情况，并结合员工信息对人员调岗、离职等事实事件，在业务系统账号权限中有所匹配，避免出现权限过大、功能过多等权限不匹配情形。另一方面，该系统应支持自定义审计报表的自动化生成时间（按日期 / 间隔日期），以满足定期内外审计的要求。

3.5 从稽核面向管理、从事后走向前置

将权限稽核系统与 OA、人力系统打通联动，精确的将权限审批人员和权限申请人员对应起来，赋予权限稽核系统权限申请及变更管理职能，结合 RPA 技术，实现权限申请、权限审批、权限授予、权限审计、权限回收的账号权限全生命周期管理，完成权限事后审计到权限前置管理的过程。从以往的被动处理向主动管理进行转变，将事后审计和事前监管相结合，更加完善了权限的梳理和监管。使我公司更快、更准、更方便管理权限问题，极大提升了公司的风控能力。

用户授权新建、变更、冻结、解冻、复核操作与合规性控制管理模块紧密结合，利用自动授权模块中定义的职责分离控制和敏感访问规则，在用户申请授权阶段对用户所需要的权限集合进行分析，在第一时间为相关人员提供风险预警。

同时，在授权申请、冻结 / 解冻申请过程中，对于用户跨系统、跨模块的操作进行提醒，防止误操作的发生，提高授权类运维管理的准确性。通过针对用户业务授权管理需求进行分析，提供用户便捷、自主、快速的权限选择方案，并在权限稽核系统中予以实现，简化并规范各应用系统权限管理，提升运维权限管理效率，实现用户快速、准确、自动授权管理。

4 实践历程

建立账号权限稽核系统，除了需要考虑系统并发量及稳定性、安全性可达到公司要求外，还要求系统功能简单易用，可方便对接其他系统数据，以便于面向业务人员进行推广，快速实现业务需求。

4.1 系统功能架构

权限稽核系统基于现有的 IT 业务系统创建统一的账号管理台账，将该系统的用户信息与各业务系统的账号信息进行对应关联；以实现基于账号管理台账的账号权限跟踪实时管理，保证账号权限可按照既定流程进行实时、准确更新，同时支持不同时间维度的数据对比、权限审计等功能。

权限稽核系统在日常跑批过程中，会将员工基准数据、业务系统账号数据、业务系统权限数据作为当日时间点的权限快照数据进行分别存储。在数据导入结束后，通过账号台账、权限基线等基准数据，将员工数据、账号数据、权限数据形成下层的权限稽核数据仓库，供上层应用功能进行调用。

上层应用可根据对数据使用的维度，将不同业务系统的相关数据进行提取，形成对应功能的告警、报表、差异数据，供前端用户查询使用。

4.2 建立应用系统的账号映射和关联匹配，形成公司级统一权限数据模型

按照系统分类或者组织架构视角实现不同业务系统账号和员工的关联匹配，通过唯一的库表键值实现多系统账号与员工进行关联，保证账号信息的准确。另外建设智能匹配功能，可快速检索出系统中有冲突的账号（例如同名、同 ID 等情况），辅助系统管理人员对账号进行甄别和手

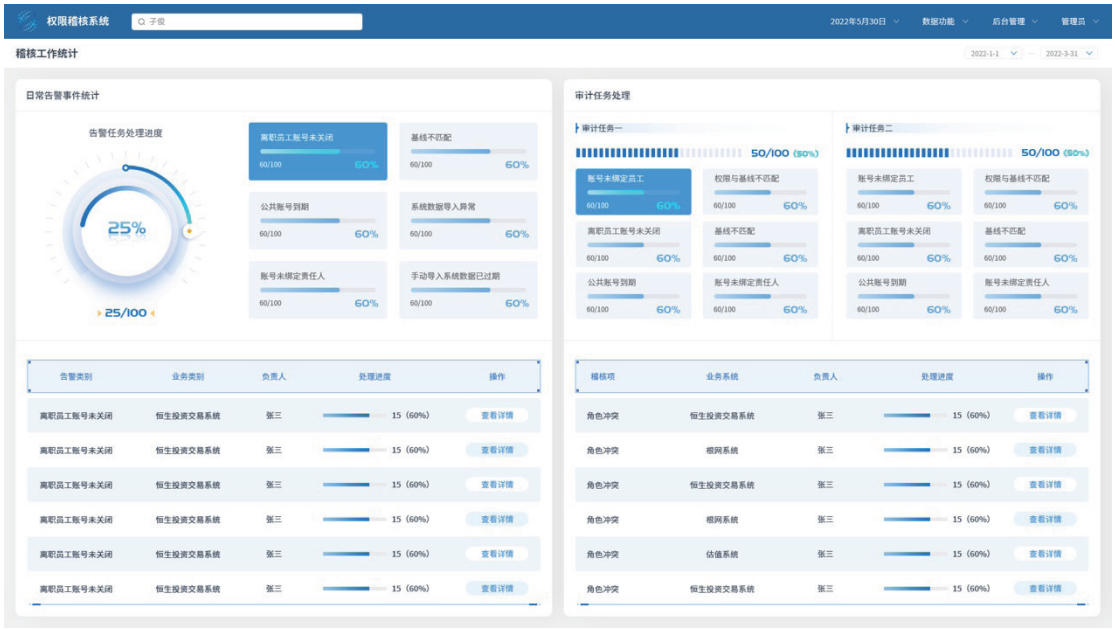


图 1：权限管理驾驶舱展示图

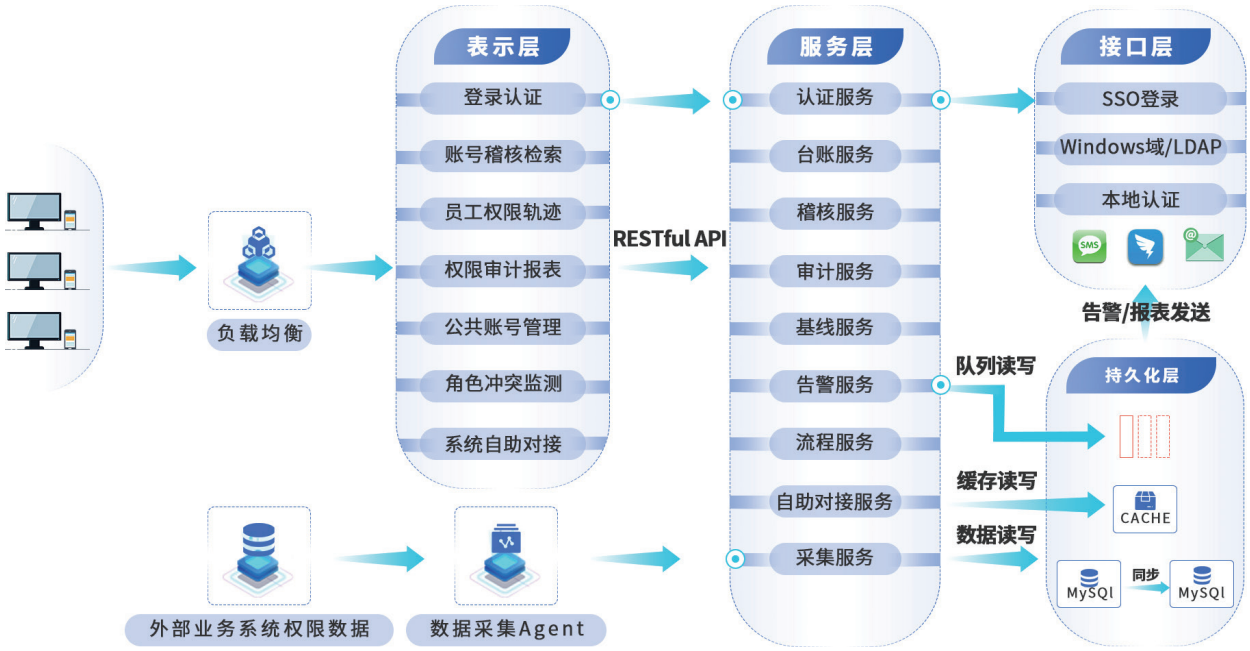


图 2：系统功能架构示意图

动关联，最终实现账号与员工匹配的一致性。将公司各内部运营管理需要的业务系统账号实现关联唯一。

基于权限台帐的建立，按照“部门视角”和“系统视角”两个视角对账号权限进行查询检索，同时可对某单一指定账号进行管束定位用户账号或系统账号信息。可查看该账号所在部门和业务系统中账号状态情况及业务系统账号权限情况。从而实现通过一个系统对账号权限进行查询，快速对权限比对结果进行查询定位。

4.3 明确权限基线，建立权限申请管理标准

在权限稽核系统建设完成后，建立了统一的员工权限管理台账。在权限申请、变更流程中，结合权限稽核系统的相关数据与接口，在权限审批通过后，由权限稽核系统汇总员工权限变化，生成权限变更清单，结合 RPA 系统，实现权限的自动变更，在权限变更完成后，由权限稽核系统核查权限变更情况，并根据变更结果发出相应提示。

4.4 实现权限对比自动化

发现用户账号在各业务系统的权限差异、变化，并结合员工信息对人员调岗、离职等事实事件，在业务系统账号中有所匹配，避免出现权限过大、功能过多等权限不匹配情形。另一方面，该系统应通过权限自动化比对，满足定期内外审计的要求。

权限自动比对是账号权限稽核系统的核心功能之一，比对维度可以按照“时间维度”、“基线维度”和“系统维度”等不同维度进行权限的自动比对。

基于“时间维度”的权限稽核比对是按照不同时间维度对一个用户（或所有用户）在一个业务系统（或所有业务系统）中的权限稽核比对，并可以自动化的定期输出比对稽核结果。当然仍需支持手动方式的稽核，可随时的对稽核结果进行导出。

基于“基线维度”的权限稽核比对是需要依托于对权限基线的建立，可对任何一个时间维度的权限数据与权限基线进行比对，实现监管审计

要求的“最小功能，最小权限”。

基于“系统维度”（即不同系统间）的权限稽核比对，可将同一用户账号在不同系统中权限信息进行比对，并将比对结果进行导出，如在业务系统与管理系统中的权限差异比对。

4.5 账号权限全生命周期展现

基于权限台帐的建立，结合对各业务系统用户账号信息的对接，权限稽核系统应自动按照数据内容生成每一个账号的历史权限变更轨迹，便于权限的内、外部审计和追踪。并且基于与 OA、人力系统的打通联动，实现了从员工入职、账号申请、权限审批、权限稽核、权限回收的全生命周期管理，满足监管及行业自律要求。

4.6 临时授权跟踪管理，封堵权限漏洞

建立临时授权管理功能模块，针对如交易员休假或临时外出，在临时授权其他同事交易员权限、对其功能权限进行操作时，系统自动对该授权合规性进行监测检查，查看是否满足授权要求。同时对其授权时间进行规定，及时触发回收临时授权权限流程，当发现不属于合规授权时，可及时发出告警通知给相关人员。

4.7 日志留痕满足追溯

账号权限稽核系统应提供系统内用户登录 / 退出时间、用户登录异常、用户操作记录、登录系统的客户端 IP 地址 / 浏览器等内容的操作日志的审计功能，同时可按照不同字段进行日志查询检索。

4.8 权限自动对接，实现权限稽核工作的自主可控

权限稽核系统完成了可自主配置的第三方业务系统权限数据对接功能。基于 Web 界面实现业务系统权限结构模型配置、数据源配置、权限数据导入（SQL/ 自动导入 /Excel/ 数据上传）等功能。

该功能适用于任何第三方业务系统权限模型，可通过自主配置实现第三方业务系统与权限稽核系统数据的快速对接。后续我公司所有系统上线后均可快速接入权限稽核系统进行权限管理。

5 未来建设规划

5.1 通过审计任务的方式实现自动化审计

根据自查工作模板中的稽核要求，与系统内可选择的稽核规则（如检查是否与基线匹配、临时权限过期等）生成权限审计模板。审计模板可根据稽核规则、部门范围和业务系统的组合选择事件处理人、审核人，模板配置完成后，根据所配置的时间条件，定期生成审计任务。审计任务开始后，会生成各处理人的待办事项。审计任务处理完成或任务期限截止后，会邮件通知审核人所有事项的核查结果，进一步提高审计效率。

5.2 基于业务视角的岗位基线

实现可跨系统设置的岗位基线，可定义基线在跨业务系统中应当拥有的角色、权限信息，并可维护员工与岗位基线的关系。当岗位基线设置完毕，新增人员或人员调岗后可以针对岗位一键赋予新权限并及时撤回旧权限；当员工申请岗位基线外的权限，管理员在权限审批时也会提示额外权限申请，并审视其合规性是否符合最小权限原则。

5.3 告警自动化，告警复核流程

异常告警发出后，系统中记录告警记录，并发送给相关系统管理员。系统管理员登录后，需要对告警内容进行确认，可设置告警的自动忽略到期时间。在到期时间内，告警不再重复发送通知。

管理员可针对公共账号设置默认生效时间，账号授权超时则产生日常告警。

针对各系统中角色的互斥关系设置（基于岗位基线），如果各系统账号设置的角色信息，匹配冲突规则，将发出告警。

6 总结及思考

加强业务稽核，防范化解金融风险，是金融公司完善治理结构的内在需求。实际工作中，管理成本高、效率低、稽核过程繁琐是权限管理传统模式中的主要瓶颈。对信息系统权限进行稽核，保证权限合理、准确、遵循最少功能以及最小权限等原则分配信息系统权限，减少潜在风险，将逐渐成为公司日常管理工作中重要的一个环节。

权限稽核系统实现了可从时间维度、角色维度、系统维度的数据比对和权限审计。通过信息系统按照设定时间（如：每天、每周、每月等等）或即时查看权限比对信息和历史变化轨迹，从原有的每季度通过人工审计方式比对，转变成了系统设定自动比对模式。既解决了工作量和时效性的问题，同时也让比对结果精准无误。

基于 Web 界面的业务系统权限结构模型配置、数据源配置以及权限数据导入（SQL/ 自动导入 /Excel/ 数据上传）等功能打破了传统 API 接口对接模式，让任何第三方业务系统权限模型均能够快速实现与权限稽核系统的对接。

公司在 OA 系统中，设计了标准的业务权限变更工作流程，员工通过该流程可以实现对所有系统权限变更申请的审批。业务权限管理信息系统提供了用户信息系统权限查询、过期权限查询、审批流水查询、业务系统权限互斥规则库等功能，实现了对权限分配信息进行集中动态管理，并实现了与 OA 系统数据的无缝对接。OA 权限变更申请流程兼顾了员工的办公习惯，员工使用 OA 系统可以自助发起权限申请，系统可以提供权限变更引导，权限设置完成后，员工的权限变更信息会自动同步到业务权限管理系统中。

根据外部监管、审计机构对于用户授权及合规性的要求及职责互斥的要求，结合我公司信息化应用系统现有的业务流程，梳理了各项业务活动间职责互斥可能带来的风险事件，最终设计了公司统一的一套职责互斥规则库。对于业务流程中相关业务活动存在职责互斥带来的风险不同，在职责互斥矩阵确定了存在风险的属性，即高、中和低三种类型；同时，由于几个低风险的职责互斥权限授予同一个用户时，可能导致高风险的舞弊事件发生的可能性增大。

职责互斥规则库是识别业务流程中是否存在风险的依据，是信息化应用系统业务操作规范性的标准，同时也是信息化应用系统用户权限合规性治理的基础。根据内外部监管、审计要求及企业管理水平提升，职责互斥规则库也需要进行相应调整。

用户授权及合规性管理是解决企业信息化实施风险防范及管理水平提升的必由之路。权限稽核系统最终实现了用户业务授权及合规性管理模式的成功转型，系统建立以统一的职责互斥规则库管控准则、标准化角色技术控制规范、定制化业务角色体系为基础，并完成与企业信息化系统间的无缝接口，提升企业合规性管控水平、风险防范水平，为企业的高质量发展保驾护航。

基于上证云信创基础设施的应用系统容器化改造探索与实践

王利鹏[1]、裘岱[1]、张晓军[1]、倪智[1]、李俊勇[2]/[1]上证所信息网络有限公司 基础架构部 上海 200120
[2]上证所信息网络有限公司 技术研发部 上海 200120
E-mail：lpwang@sse.com.cn

为满足多样化应用系统的运行需求，基础架构不断推陈出新，从计算机系统架构分层、虚拟化到云计算，IaaS、PaaS、SaaS 等概念也已成为现实，从资源到架构全面弹性的云计算不仅可以降低运营成本，同时更加契合变化的业务需求，云时代已经到来；与此同时，应用架构也发生了从单体架构、分布式架构、SOA 架构及微服务架构的技术演进。随着容器、容器编排等技术发展，应用也逐步向云原生应用模式转变，以容器为代表的云原生技术，正成为加速企业数字化转型的利器。同时，随着中国的不断发展，我国明确了“数字中国”建设战略，提出了信息技术应用创新发展新要求。遵循国家相关部门监管政策，基于技术领先、稳定可靠、安全合规的金融级行业云信创基础设施建设证券行业信息服务应用系统具有重要意义。本文主要介绍了数字化转型下基于上证云信创基础设施的应用容器化改造的探索与实践经验。

关键词：上证云；信创；容器化

1 背景

随着中国的不断发展，信息化进入加快数字化发展、建设数字中国的新阶段，我国明确了“数字中国”建设战略，面对复杂多变的国际形势同时提出了信息技术应用创新发展的新要求。

党中央、国务院高度重视信息化工作，近年来，我国大力推进 5G、物联网、云计算、大数据、人工智能、区块链等新技术新应用，坚持创新赋能、激发数字经济新活力，数字生态建设取得积

极成效。为响应国家号召，各方有力促进各类要素在生产、分配、流通、消费各环节的有机衔接，实现了产业链、供应链、价值链优化升级和融合贯通，为建设网络强国和数字中国奠定重要基础。遵循国家相关部门监管政策，上海证券交易所基于技术领先、稳定可靠、安全合规的云计算技术建设了金融级上证云基础设施 -- 面向证券、基金、监管机构、核心机构等金融机构推出的云计算服务。

深处百年未有之大变局之中，面对复杂多变的国际形势，尤其是俄乌冲突以来俄罗斯计算机产业被欧美全面制裁，传统的欧美 IT 巨头如英特尔、AMD、苹果、谷歌、微软等均宣布全面停止向俄罗斯供货，导致了俄罗斯 IT 产业进入了极为艰难的境地。中国知名企业华为被美国无理打压，也充分暴露了国与国之间高科技产业竞争的激烈。打造以 CPU 和操作系统为重点的国产化信息技术体系，逐步取代欧美产品，实现信息技术领域的自主可控，降低信息安全的风险，是目前的一项重要国家战略，也是当今形势下国家经济发展的新动能。

为顺应金融科技发展和全球治理体系变革的时代要求，上海证券交易所致力于加快打造“数字智能型交易所”，增强技术基础设施运维和安全保障能力，持续推进数字化安全运营。作为证券行业信息技术应用创新的排头兵，为推动行业数字化生态建设、发挥科技创新支撑作用，上海证券交易所陆续投入了大量的人力、物力等资源开展相关研究，不断完善数据中心布局与云服务，并打造了信创云基础设施。

作为打造“数字智能型交易所”的重要一环，根据上海证券交易所“十四五”科技战略规划及交易所集团化的战略路线，上证所信息网络有限公司（后续简称信息公司）作为上交所控股子公司，陆续开展信息服务系统的上云及信息技术应用创新改造，以支持交易所业务拓展，为市场服务提供数据支持。

2 云基础设施与云原生发展

2.1 基础架构演进

在应用系统开发建设时，预先设定了应用今后运行所依赖的环境 -- 基础架构。基础架构涉及到应用运行所需的系统资源，解决了应用运行的一些通用性问题，往往针对不同的基础架构，应用的部署、运行等也不尽相同。

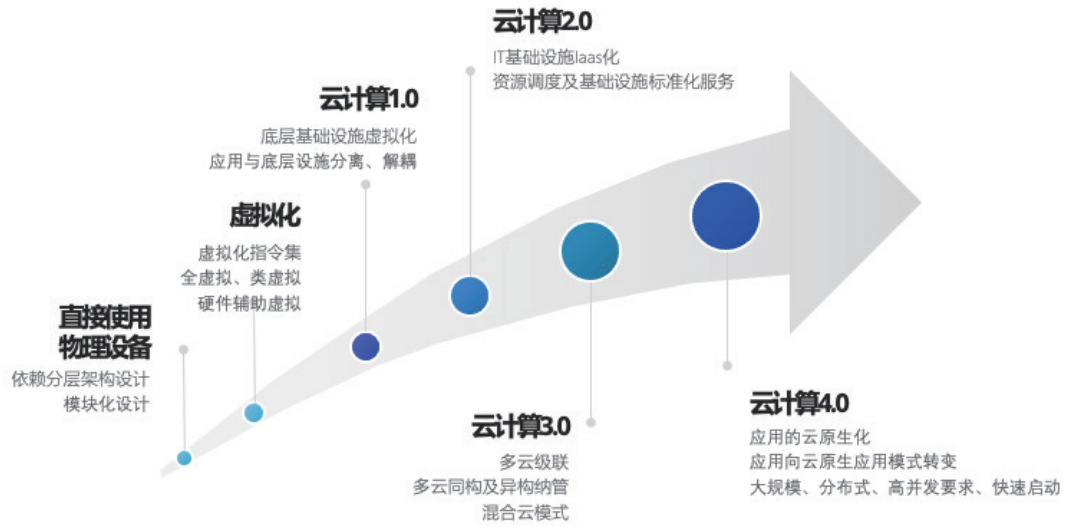


图 1：基础架构演进

在计算机系统的架构上，分层思想得以贯彻并取得卓著成效，分层大大降低了系统设计的复杂性，提高了软件的可移植性。借助全虚拟化、类虚拟化、硬件辅助虚拟化等方式，虚拟化技术得以蓬勃发展并解决了部分资源利用率的问题，通过虚拟机镜像模式封装相关技术栈（包含应用服务及所有依赖）大大简化了应用部署及交付。

虚拟化软件虽解决了硬件资源利用率问题，但缺乏时间、空间、操作的灵活性，如要创建虚拟机需要预先人工指定虚拟机放在哪台物理服务器上。为实现资源到架构的全面弹性，云计算应运而生并突飞猛进，使得用户可以通过网络，以按需、易扩展的方式获取所需资源。随着 IaaS、PaaS、SaaS 等相关产品蓬勃发展，云计算不断演进，从 IT 基础设施虚拟化到 IT 基础设施 IaaS 化，再到多云级联，云计算把一台台服务器连接起来构成一个庞大的资源池，提供标准化的、云化的资源给上层应用（如虚拟机、虚拟存储、虚拟网络、应用运行时环境及依赖等），让应用无须担心下层的资源分配调度。基于云计算技术的云基础设施使得高效能并行计算走进普通用户，其所提供的创新性和灵活性的资源使用，不仅可大大降低运营成本，且更加契合变化的业务需求。

2.2 应用架构演进

在基础架构演进的同时，应用架构也在不断迭代和完善。应用架构的重要性在于实现应用的非功能性需求，而非功能性需求往往能够决定一个应用的运行时质量（如可扩展性和可靠性等），通常也能决定一个应用开发时的质量（如可维护性、可测试性与可部署性等）。

应用架构的终极目标是用最少的人力成本来实现构建和维护应用的需求。从应用架构模式的演进路线来看，大体上可以分为单体架构、基于组件的架构、分布式与 SOA 架构以及近来较火的微服务架构等。但应用架构不是脱离基础架构独立发展而来，只有在基础架构的底层“系统资源”支撑下，配套的、先进的应用架构才能够得以实现并被广泛使用。

2.3 容器、容器编排及云原生

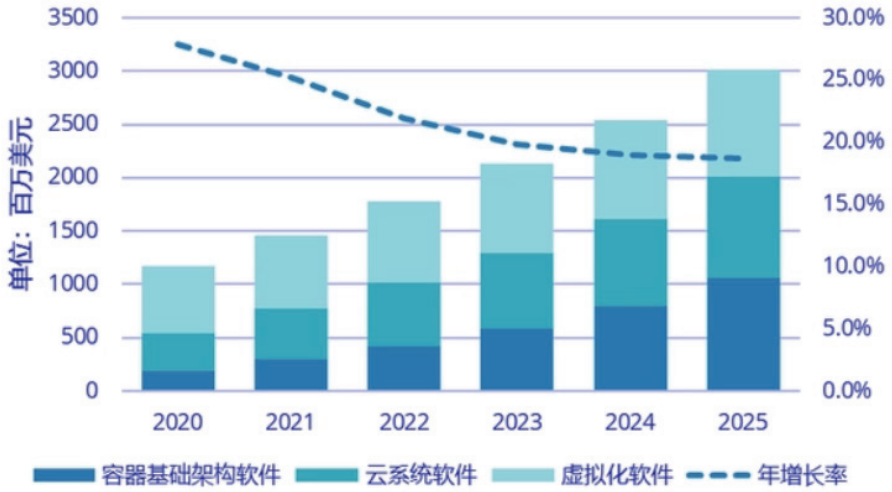
虚机机虽然解决了部分硬件资源利用率的问题，但将应用打包成虚拟机镜像进行部署交付的这种模式下，每个应用仍然占用整台虚拟机的开销（包含其操作系统），资源利用率并未达到最大化，且由于虚拟机镜像较大导致其部署速度相对较慢，如涉及 OS 和运行时打补丁，还要重新制作镜像。

随着虚拟化技术的发展，轻量级的容器技术诞生。不同于虚拟机对底层硬件设备的抽象处理，容器对操作系统进行抽象处理；相比虚拟机，容器化的应用只是宿主机上的普通进程，带来了更小的性能损耗。以 docker 为代表的容器技术将应用及其所依赖的框架、中间件等运行环境打包至分层的镜像中，在单机上可以为应用运行提供相互隔离、轻量化、简便化的运行环境，让应用以一种自包含且一致的形式部署在任何环境中。

容器技术也有其局限性，无法支持跨主机的分布式应用的打包部署，随着应用规模的逐步扩大对散落在不同节点上的容器的管理成本非线性上升，自动化管理和协调容器的容器编排技术得以发展。随着 k8s 成为现今最主流的容器编排技术与行业事实标准，其创新性实现了容器的生命周期管理，凭借其强大的编排和调度能力也逐步被认为是支撑分布式应用的云平台上的分布式操作系统。

2013 年 Pivotal 公司的 Matt Stine 提出了云原生概念，应用因云而生，即云原生，应用原生被设计为在云上以最佳的方式运行。随着云计算、微服务架构、容器等相关技术的发展，云原生技术使得最大化利用云平台的能力，发挥“云”的价值并进一步避免资源浪费和效率低下成为可能。

以容器为代表的云原生技术，正成为加速企



来源：IDC中国，2021

图 2：中国软件定义计算软件市场预测

业数字化转型的利器。根据 IDC 预测，容器软件市场在近几年呈爆发式增长，到 2025 年，容器基础架构软件市场收入将与虚拟化软件市场、云系统软件市场齐平，成为近几年促使软件定义计算市场增长的新动力。

3 顶层规划 谋定后动

为打造“数字智能型交易所”，实现上交所数字化转型，“十四五”以来，上交所进一步加强顶层设计，在数字化专业委员会统筹下，梳理完善技术总体布局，将技术系统按板块进行划分管理，优化数据中心布局，统筹基础设施板块统一治理，确立了《上海证券交易所“十四五”科技战略规划》。

根据上交所科技战略规划及交易所集团化的战略路线，信息公司也投入了大量资源加快公司数字化转型，整体规划了公司信息技术系统建设路线。

3.1 组织保障

为推进数字化转型，指导和协调公司科技发展与安全运行工作，切实提升公司数字化和安全运行水平，公司进行了相应的组织架构等变革。

1）成立数字技术委员会

由数字技术委员会组织制定和实施公司信息技术发展规划，组织标准化体系的建设，制定制度标准并组织实施等工作。

2）组建数字化转型人才队伍

抽调业务、开发、测试、运维等符合数字化转型需求的核心人才，组建相应队伍，负责基础设施、软硬件、系统架构、应用架构、技术路线选型以及业务系统规划建设，完成创新实践活动的落地落实。

3）开展讲座、培训与沙龙等活动

定期举办技术研发实战系列讲座、创新业务系统讲座，参加证券基金行业信息技术应用创新联盟活动，开展上云改造经验交流会，组织内部技术沙龙等活动，营造数字化转型学习氛围。

3.2 明确方向

根据公司信息技术系统建设路线，围绕“5+3+1”技术总体框架：优化整合基础设施，构建安全运行底座，基于行业云基础设施，赋能信息服务系统；统筹高质量数字化转型，打造低碳高效服务新模式；提升自主可控能力，加强安全

图 3：信息技术系统建设方向

与数据治理；探索应用治理新模式，提升数字化治理水平。

1）应云尽云

信息公司的信息服务系统大体上都是面向互联网用户的应用系统，在技术路线上与互联网公司有一定的相似度。之前，信息公司的信息服务系统依赖的基础设施为租赁的同城灾备传统 IDC 机房以及负责业务处理的自有机房。为优化基础设施和降低成本，本次互联网系统转型目标基础设施全面转向上交所的金融级上证云基础设施，应云尽云，逐步推进，待时机成熟后优化公司整体数据中心布局。

2）低碳高效

结合国家“双碳”目标，充分平衡好绿色发展，以新技术为抓手，深入开展精细化运维、自动化运维，加强节能技术研究，优化数据中心功能布局，实现资源利用率提升，提升能效比，降低整体运营成本。

3）自主可控

面对复杂多变的国际形势，将自主可控与公司信息技术系统建设紧密结合。基于底层信创基础设施，优化供应链，建立软硬件白名单机制，强化开源治理，上云改造与信创改造相结合，统筹推动等保测评、商密密评和国密应用适配，实现规模部署 IPv6。

4）提升数字化治理水平

以信创上云改造为契机，完善信息技术规则制度体系，提升 IT 管理水平，基于云原生、分布式或微服务架构等技术完善应用全生命周期管理，组建人才队伍，夯实技术运维服务管理，健全风险管理，提升技术服务持续交付效能。

3.3 确立目标

根据公司信息技术系统建设路线规划及上云改造总体指导方向，确立基于上证云信创基础设施进行试点应用系统的容器化改造，探索并逐步完善相应的技术运维服务体系及配套保障措施。

经充分评估后，首批试点应用系统选择具有代表性的公司公共服务系统，该系统作为公司信息服务系统的底层支撑系统，承载了包括短信邮件服务、身份识别服务、应用网关、支付接口等核心功能。

4 总体设计 纵观全局

4.1 以上证云信创基础设施为骨

考虑公司的公共服务系统包含支付、身份识别等核心功能，本次上云改造在部署底座上选择上证云信创基础设施。该基础设施底层服务器均为 X86 架构的海光 CPU 服务器以及 ARM 架构的鲲鹏 CPU 服务器，以国产芯片实现信息技术应用创新。云底座为基于国内云服务商的信创云技术建设的证券行业技术领先、稳定可靠、安全合规的金融级云，在可用区（AZ）下具备 2 个数据中心（DC），依托于良好的架构设计可以有效实现应用系统容灾。

在云服务器操作系统选型上，本次改造是基于麒麟服务器操作系统 V10。该系统针对企业级关键业务，支持云原生应用，可有效满足数据中心及下一代的虚拟化（含 Docker 容器）、大数据、云服务的需求。经过前期试点测试验证，该操作系统可有效满足公司应用系统的支撑及相关运维要求。

另外，在数据库的选型上，则优先采用了云平台提供的海量数据库（Vastbase）云服务。基于平台层提供的基础服务支撑，在应用系统的改造中可更加关注于上层架构设计与业务逻辑实现，而不用过于关心底层基础服务的健壮性，可有效缩短大规模系统上云改造的整体时间。同时，海量数据库作为基于 openGauss 内核开发的企业级关系型数据库，在前期的应用系统适配、性能测试和迁移改造难度等方面的试点测试中表现不俗。

4.2 以容器镜像交付为筋

在传统的服务器架构中，通过连接到服务器，升级或回滚软件包、调整配置文件以及部署新代码等操作，服务器状态会不断地被更新和修改，这种可变基础设施通常会导致：灾难发生时，难以快速重新构建服务；在服务运行中，持续修改服务器会引入中间状态，从而导致不可预知的问题。

本次上云改造结合容器技术，采用不可变基础设施的交付理念（不可变基础设施是指任何基础设施的实例，如服务器、容器等各种软硬件，一旦创建之后便成为一种只读状态，不可对其进行任何更改），避免手动配置造成不可控制化修改，大大提升发布效率。同时，基于容器镜像交

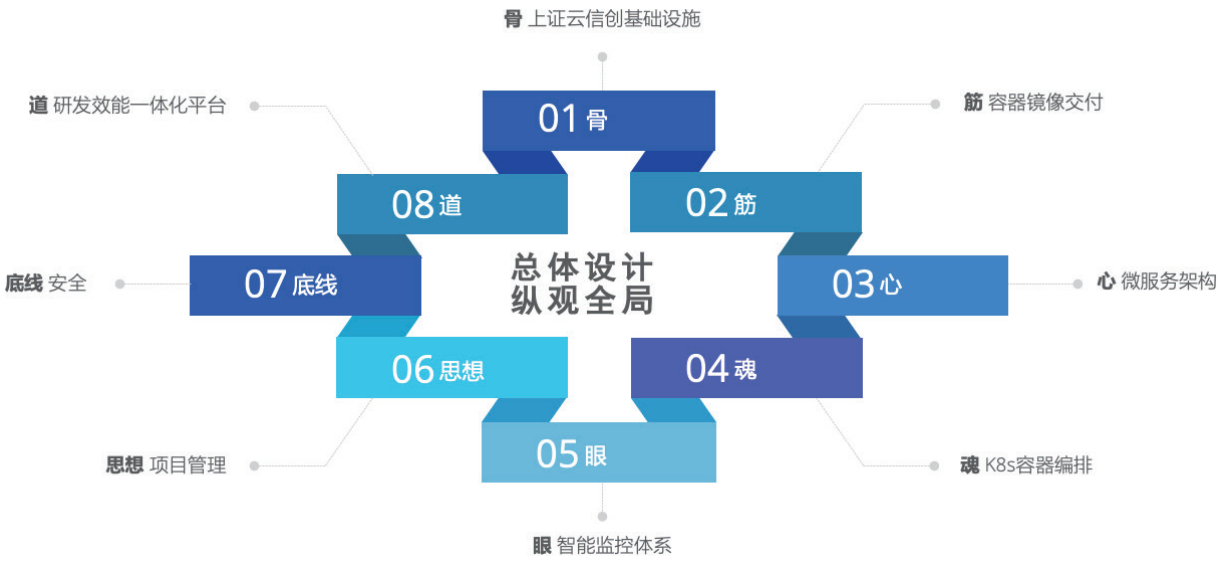


图 4：系统改造总体设计

付，可解决环境重现性问题，减少因环境差异导致的生产问题；通过升级镜像用新组件替换旧组件也可保持良好的生产运行状态。基于不可变基础设施的一致性，可以实现大规模下水平扩展性，回滚和恢复也会更加方便。

4.3 以微服务架构为心

基于云原生的应用部署的哲学基础和根本原则就是让那些高度分布式的应用程序可以在不断变化的环境中运行。应用架构演进的关键之一在于确定组件的架构单元以及单元间的耦合性。微服务架构为增量变更设计，基于微服务定义了部署时的边界，封装了服务所依赖的组件。

本次上云改造，应用系统采用微服务架构设计，围绕领域概念形成限界上下文，使得变更只影响微服务所在的上下文，同时每个微服务有着明确边界，实现解耦，确保设计上高内聚、低耦合。

基于良好微服务架构设计的应用颗粒度更小且可独立部署，本次改造的网关应用系统将涉及邮件、短信、支付、身份识别、路由网关等等服务进行了细分，每个细分的微服务依托于容器进行资源分配与调度，可以实现轻量级资源占用，间接较大提升了硬件资源利用率，实现节能减排效果。

4.4 以 k8s 容器编排为魂

容器实现了单节点上的应用打包、发布及运行等功能，有了分层的容器镜像后虽利于应用交付部署，但随着微服务等架构发展、应用规模的逐渐扩大，对于散落在不同节点的容器管理成本突增，对于技术服务运维体系也颇具挑战。基于容器编排平台统一管理分布式的容器节点成为趋势，通过其自动化管理和协调容器能力也可实现容器的生命周期管理。

随着 k8s 成为现今最主流的容器编排技术，凭借其先进的架构理念、运行架构被称为云平台上的分布式操作系统。针对本次上云改造，技术上引入 k8s 集群作为微服务化容器的运行平台，而上证云信创基础设施本身所提供的容器集群云服务凭借其技术优势使得 k8s 集群的生命周期管理变得简单高效，在应用服务的管理维护方面其所提供的 web 界面也相当人性化、交互性极好（相较于 k8s 的命令行使得运维门槛大大降低）。此外，微服务化的容器应用，有了容器编排加持，使得容器资源调度更加高效可靠同时进一步提升了资源利用率，相较于传统的服务器架构也更加符合低碳高效、绿色发展的理念。

4.5 以智能监控体系为眼

信息技术系统的安全运行除了依赖基础设施、应用架构的高可用及应用本身的优良设计之外，还需要安全防护以及监控体系的辅助，没有监控的技术系统就是在裸奔。不论是数字化转型，还是信息技术应用创新，安全都是基本原则，“安全运行”更是交易所的生命线。

作为本次上云改造的目标之一 -- 探索并逐步完善相应的技术运维服务体系及配套保障措施，在改造应用系统的同时建立配套的智能监控体系也是本次改造的重中之重。尤其是公司公共服务系统所承载的是其他业务应用系统的基础服务支撑功能，对于其可靠、高可用等方面的非功能性要求也相当之高。

本次的上云改造，技术上采用上证云信创基础设施本身所提供的智能监控体系，其除了涵盖传统的监控指标之外，还纳入了容器监控及日志归集等功能，无须特殊配置，开箱即用。同时在云上监控体系主用的基础上，通过指标接入并进一步完善公司本身的统一运维监控平台，以完善精细化运维、自动化运维，在降低整体运营成本的基础上提升技术系统的整体可靠性。

4.6 以项目管理为思想

本次试点上云改造的应用系统作为后续批量

应用系统改造的先行者，对于基础设施、应用架构、系统架构等都需要进行充分的验证，相应的架构设计和技术运维服务体系及配套保障措施将在未来的一段时间内承载公司所有涉互联网的业务系统。针对试点改造的工作推进、目标完成都有着时间后墙；改造以上证云信创基础设施为支撑，在组织协调上外部涉及子公司间、服务提供商等的协作，在公司内部也涉及领导、技术条线、业务条线等等人员沟通协作，相关干系人众多；同时，此次改造在相关软硬件等方面，也涉及到多项采购工作。

本次上云改造除了统一规划、人才队伍保障、多方协作等外，将项目管理的思想贯彻始终。作为打造“数字智能型交易所”、实现公司数字化转型的重要一环，在本次试点系统改造项目立项前，就由公司项目管理高级人员及外部专家开展项目管理系列培训及项目管理实践经验分享；在项目目标确定后就指定了资深项目经理，并抽调各部门骨干精英和具备项目管理经验的管理人才共同组建项目团队。

在项目的整体推进中，通过制定的项目章程，逐步求精完善项目管理计划，借鉴项目管理的十大知识领域和 PDCA 循环理论对项目进行管理和推进，并根据具体情况适时调整计划或督办相应事项。例如，考虑到干系人众多，在沟通管理方面，选择了公司通讯软件、公司邮件作为日常沟通主要工具，会议安排除了小组不定期通气会、项目团队内部例行周会外，月度开展信息技术应用创新专题会、板块协调沟通会，开展采购进度专题会等，会议纪要通过邮件等方式及时通知到相关干系人；在沟通方面也根据具体沟通情况和相关需求进行适时调整，如考虑疫情影响安排线上与线下会议相结合。

4.7 以安全为底线

“安全运行”是生命线。不论是应对复杂多变的国际形势，还是打造“数字智能型交易所”，抑或是公司数字化转型，安全都是高压线和红线。不论是运行安全、信息安全、业务安全、网络安全，还是加强自主可控都是信息技术系统建设所要遵守的底线。

本次上云改造，立足坚守安全底线，在系统

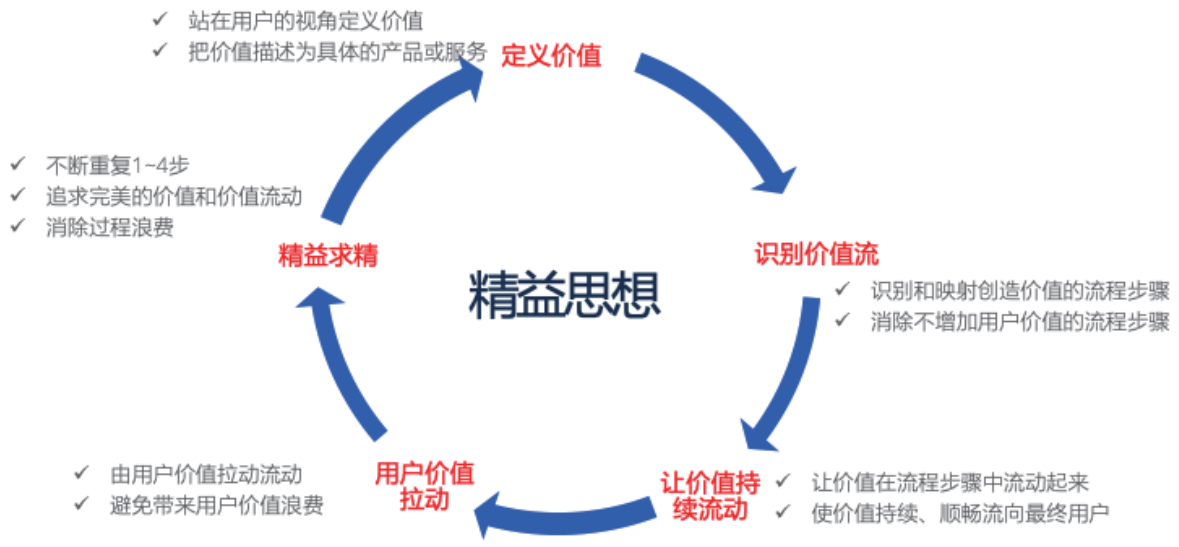


图 5：精益思想

建设过程中贯彻安全理念，在业务系统建设中融入安全防护体系，在团队建设中宣贯安全思想。

4.8 以研发效能一体化平台为道

传统的应用开发模式往往只注重开发，不注重交付。在 IDC 中，为满足业务需求，后续大量的运维和运营工作占了操作频率的 90% 以上。传统软件部署方式中，应用软件包开发完成后交给运维人员，运维人员再准备各类资源部署应用程序和相关依赖，这种交付部署方式极其繁琐且不够灵活。

随着 DevOps 理念和持续交付概念的发展，越来越多的企业将精益思想贯彻到技术服务交付的整个价值流过程，从产品设计、开发、质量保证到运维，通过更低的成本保障技术服务交付的高效性、高质量、可靠性和稳定性。

本次上云改造，在完成应用系统本身的建设之外，同时要改善整个技术服务交付价值流的流动速度。将过往分散的项目管理、需求管理、开发管理、测试管理、制品管理等工具链条打通，打造研发效能一体化平台，贯彻 DevOps 和持续交付理念，建设 CI/CD 流水线，通过电子流程实现各个环节的联通（同时做好用户及权限管理和审批控制），最终目标是通过现代的软件工程模式实现 IT 价值快速流动。

5 细心打磨 步步为营

即便有了全局规划，在具体应用的上云改造过程中，由于其独特性往往还存在不少的细节问题需要攻克，针对本次试点改造公共服务系统所遇到的部分细节在此与大家分享。

5.1 微服务

5.1.1 框架选型

本次改造在架构选型上并没有使用开源社区微服务通用框架全家桶，而是根据公共服务系统的应用架构做了组件裁剪，优先选择了 Spring cloud gateway、Spring cloud sentinel、Ribbon、Nacos，中间件则选择了 RabbitMQ 和 Redis，安全框架则使用了 Springsecurity。

5.1.2 设计原则

考虑到公共服务系统作为其他应用的基础支撑服务，在架构设计上通过统一的 API 网关对外提供服务，各服务间通过 REST 协议通信，同时每个微服务有自己独立的数据存储；在 API 设计方面则采用 API 组合模式与 CQRS（Command and Query Responsibility Segregation）模式相结合。

同时，本次试点改造不仅仅是为了搬迁应用到云平台，而是充分借鉴云原生相关理念，使得设计可以支撑应用不断变化、高度分布式的要求，在此结合本次改造经验简单介绍下所使用到的云原生模式：

1）**坚持应用程序冗余。**区分有状态和无状态服务，针对无状态服务设计上满足水平伸缩；针对有状态服务，解耦状态并进行持久化，如引入统一配置中心；注意 http 会话和黏性会话处理。

2）**关注应用程序配置。**区分可变与不可变配置，针对不可变配置参照不可变基础设施方式结合容器镜像进行固化，避免此类配置不受控修改引入运行风险；针对可变配置，统一配置注入应用程序的方式（可通过容器环境变量注入等方

表 1：微服务框架选型

	服务注册发现	服务调用方式	服务网关	断路器	配置中心	负载均衡	消息中间件	缓存中间件
组件选型	Nacos	REST/HTTP	Springcloudgateway	Springcloudsentinel	Nacos	Ribbon	RabbitMQ	Redis

式，也可借助统一配置中心如 Nacos 进行管理，应用方面实现热加载功能）。

3）考虑如何访问应用程序。需要综合考虑部署单元的服务注册、服务发现、服务路由以及服务负载均衡（客户端负载均衡及服务端负载均衡）等，如实施中借助 Nacos 实现服务注册发现，借助 Ribbon 实现客户端调用负载均衡。

4）配套重试和控制循环等交互冗余。云原生应用服务间的通信需要充分考虑网络的不可靠性以实现应用层面的交互冗余，综合考虑配套重试机制与兜底回退逻辑，同时避免重试风暴、合理利用循环控制。

5）附加熔断与限流功能，考虑降级服务。与电路系统的保险丝一样，云原生应用也需要考虑“负载足够大时电线过热引发火灾”的情况，应用的设计不能寄希望于客户端处于永久的“温和调用”，如借助于 sentinel 实现熔断限流；同时，考虑极端情况下为保障应用系统的核心功能可用性，非核心功能需支持降级开关功能。

6）聚焦应用程序生命周期。随着应用的微服务化拆分，部署单元的骤增将会带来应用治理的难题，在应用系统建设中要全程聚焦生命周期管理。如本次改造中基于 k8s 的所有部署单元都设计了健康端点和探测机制，保证部署单元的全程可见性。

5.1.3 拆分颗粒度

对应用进行微服务拆分是一项挑战，必须明确服务之间的依赖关系。在单体架构中，分层关注点在技术层面，包括应用的持久化存储、UI、业务规则等。仅关注技术维度分层，往往没有清晰的业务领域概念，当业务领域需求变更时，涉及到必须修改每一层（如展现层、业务逻辑层、持久层等），而大部分应用场景变更是围绕领域概念进行的，业务领域成了微服务应用功能架构的核心。

本次上云改造，每个微服务都尽量围绕业务领域概念来定义，同时兼顾技术分层和前后端分离，将实现架构及其依赖的其他组件封装在微服务中，从而满足高度解耦和灵活性，以避免大的部署单元难以演进，而过小的单元则不利于业务领域建模。综合考虑，本次公共服务系统一期将其服务分解为 gateway-vue、gateway-mid、gateway-manager、verify、ocr、email、captcha、platform-vue、platform-manager 9 个部署单元。

5.2 容器及容器编排

5.2.1 合理控制进入容器的东西

通过容器镜像这种不可变基础设施理念，可

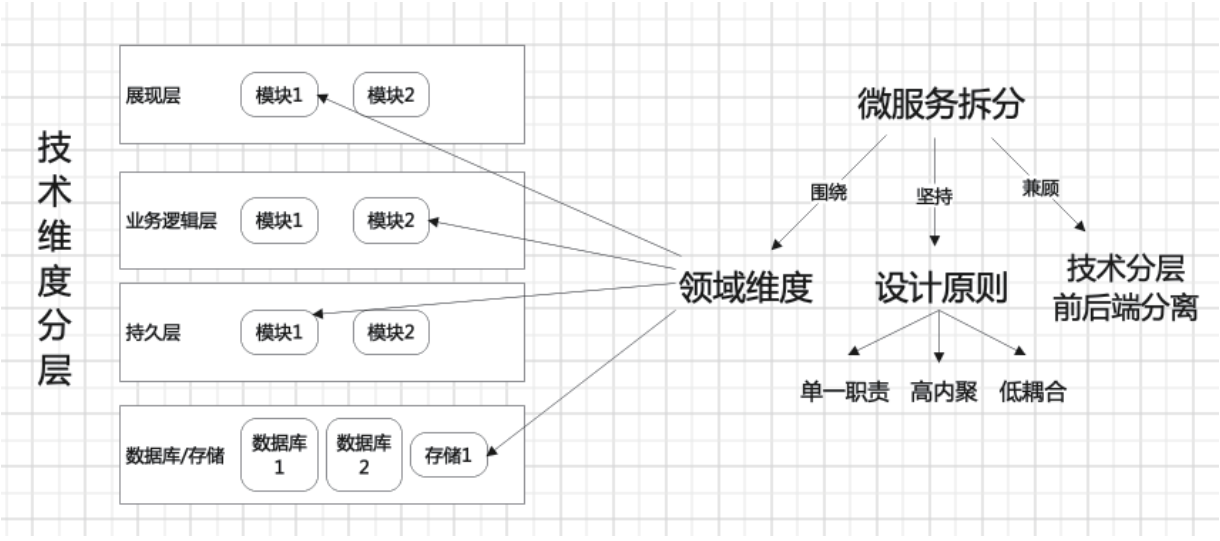


图 6：微服务拆分

以有效控制运行时环境，实现可重复性及所需的安全、合规保障。作为承载应用服务的“基础镜像”将在与应用结合后以“应用镜像”进行大规模批量化部署，由于任何软件都可能存在安全漏洞，除了应用本身外也需要合理控制进入容器的东西。

1）最佳实践保证最小的基础镜像。如本次的容器基础镜像采用 Kylin V10 底层镜像，并采取安全基线控制（如限制 ssh 访问容器等）；针对应用运行环境进行分类，不同运行环境采用不同镜像并最小化安装，如 java 语言的安装统一 JRE 环境，如遇特殊版本要求则创建不同的镜像。

2）纳管所有基础镜像和应用镜像。本次改造利用云平台所提供的容器镜像仓库服务纳管所有镜像，未通过安全测评和合规要求的基础镜像禁止被引入应用镜像构建。所有生产的部署单元均需从云上镜像仓库获取，严禁其他通道获取或“在野镜像”运行。在团队分工上，应用程序团队负责交付应用程序，平台支撑及运维团队负责满足企业安全和合规要求；应用程序团队只负责提供应用程序，而平台支撑及运维团队负责提供其他的一切。

3）严格管理应用镜像的构建和流转机制。通过研发效能一体化平台，打造了集成代码扫描的自动化应用镜像构建通道，通过制品库和审批控制电子流程审核所有推送生产的应用镜像，所有生产镜像严格纳入云上仓库管理，如遇外部产品此类直接交付的应用镜像也需提交相应的安全和合规报告后方可纳入云上仓库管理。

5.2.2 合理规划 k8s 集群

本次上云改造所采用的云上容器集群底层采用了 k8s 实现容器编排，与开源 k8s 自建集群类似，云上容器集群也需要进行合理规划，在此简述部分本次改造的实践举措与经验：

1）控制集群规模。信息公司的互联网类系统根据技术可用性等级、依据上交所相关技术系统规范归集为 13 个，按业务切分到 4 个业务部门。通过集群可隔离业务系统，此次在集群规模设计上按业务规划了 4 个 VPC，每个 VPC 下再行按系统划分集群；集群规模上基于 worker 节点标准配置（32C/64G）和部署单元平均资源限额并充分考虑公司系统情况，规划了单节点容纳 32 个 Pod，单集群承载 30 节点（合计单集群可承载 30*32 约 1000 个 Pod）；同时基于云上资源配额，一期预留了 64 个标准集群，必要时可合并标准集群为更大集群规模。

2）选择容器网络插件。上证云信创平台容器集群提供的高性能容器网络插件，支持容器隧

对比维度	容器隧道网络	VPC网络
数据面依赖	OVS	IPVlan，VPC路由
适用集群	CCE集群 虚机集群	CCE集群 虚机集群
是否支持网络策略（networkpolicy）	是	否
是否支持ENI	否	是，容器网络与VPC网络深度整合，Pod内直接使用VPC弹性网卡。
IP地址管理	IP地址可迁移	• 每个节点分配一个小子网。 • 在VPC Router上添加静态路由，下一跳为节点IP。
网络性能	基于vxlan隧道封装，有性能损耗。	• 无隧道封装，性能好，媲美主机网络。 • 跨节点通过VPC Router转发。
组网规模	最大可支持2000节点	受限VPC路由表能力。
外部依赖	无	依赖VPC Router静态路由表能力。
适用场景	• 一般容器业务场景。 • 对网络时延、带宽要求不是特别高的场景。	• 对网络时延、带宽要求高。 • 容器与虚机IP互通，使用了微服务注册框架的，如Dubbo、CSE等。

图 7：网络插件对比

道网络和 VPC 网络 2 种模型，但集群创建后网络模型不可更改。基于公共服务系统对网络延时较高，且自身使用了微服务框架，本次对其改造通过评估选用了 VPC 网络模型，此模型下无隧道封装、性能好，且容器网络可与 VPC 网络深度整合。但具体应用系统改造仍需根据具体需求合理选择网络插件，例如 VPC 网络下不支持 networkpolicy 相关策略。

5.3 资源配置及容灾

k8s 的最小部署单元为 pod，云上容器集群可以根据需要设置的不同节点池，也可以结合 label 与 taints、亲和与反亲和性等实现 pod 的资源调度和编排。同时，通过 requests 和 limits 可以实现 pod 资源申请和上限控制，如果没有合理设置 requests 和 limits 则可能出现 pod 被迫重启或者由于资源竞争导致原有 pod 被驱逐。

本次上云改造，在 pod 资源配置上通过主动声明 requests 和 limits 以保障 pod 的 QoS，防止由于资源竞争导致 pod 服务被迫中断，这对于公共服务系统尤为重要。

另外，上证云信创平台所提供的容器集群在底层具备两个数据中心，除了通过上层应用架构和 k8s 实现应用系统的高可用、高可靠之外，还可以借助基础设施的双数据中心设计容灾。通过调度同一组分布式微服务的 pod 实例分布到两个数据中心的虚机资源，同时合理设置集群水位，可确保极端情况单数据中心故障下应用系统的健壮性。

5.4 监控

5.4.1 端点及探测

基于 pod 的声明式配置 liveness Probe 与 readiness Probe，并通过暴露相应的应用监控端点，可实现 pod 服务的存活探测与健康检查；同时，通过配合设置合理的 restart Policy 策略可以实现容器的自重启治愈。

5.4.2 日志

基于传统架构的虚机部署，运维人员需要首先判断运行服务所在的机器，然后登录到相应机器实现日志搜索及分析，规模化运维成本较高。依托于上证云信创平台所提供的日志云服务，可实现容器集群日志的自动采集，并可以实现在线日志搜索功能（无需命令行登录容器集群）。

除了标准输出日志自动采集之外，可以基于云平台的弹性文件服务在线持久化应用日志，还可以根据需求弹性扩容存储空间。如果需要满足日志存储规范，还可以借助 ELK 或其他日志平台实现日志转储，同时也可以接入日志监控系统实现日志监控。本次上云试点改造，一是基于日志云服务实现了日志归集及在线搜索（默认存储 7 天），方便运维人员日常在线分析；二是基于规范化、持久化的应用日志对接自研统一监控平台实现准实时日志监控及告警；三

名称	状态	所属节点池	规格	可分配资源	IP地址	可用区
dev-cluster-1-77628-az1dc1	可用 可调度	DefaultPool	32 核 | 64 GB | kc1.8xlarge.2	CPU: 0.28 Core 内存: 2.04 GiB	10.2...	az1.dc1
dev-cluster-1-48313-az1dc1	可用 可调度	DefaultPool	32 核 | 64 GB | kc1.8xlarge.2	CPU: 31.65 Core 内存: 57.31 GiB	10.2...	az1.dc1
dev-cluster-1-38941-az1dc2	可用 可调度	DefaultPool	32 核 | 64 GB | kc1.8xlarge.2	CPU: 8.23 Core 内存: 11.08 GiB	10.2...	az1.dc2
dev-cluster-1-90446-az1dc2	可用 可调度	DefaultPool	32 核 | 64 GB | kc1.8xlarge.2	CPU: 31.65 Core 内存: 57.31 GiB	10.2...	az1.dc2

图 8：双数据中心容灾部署

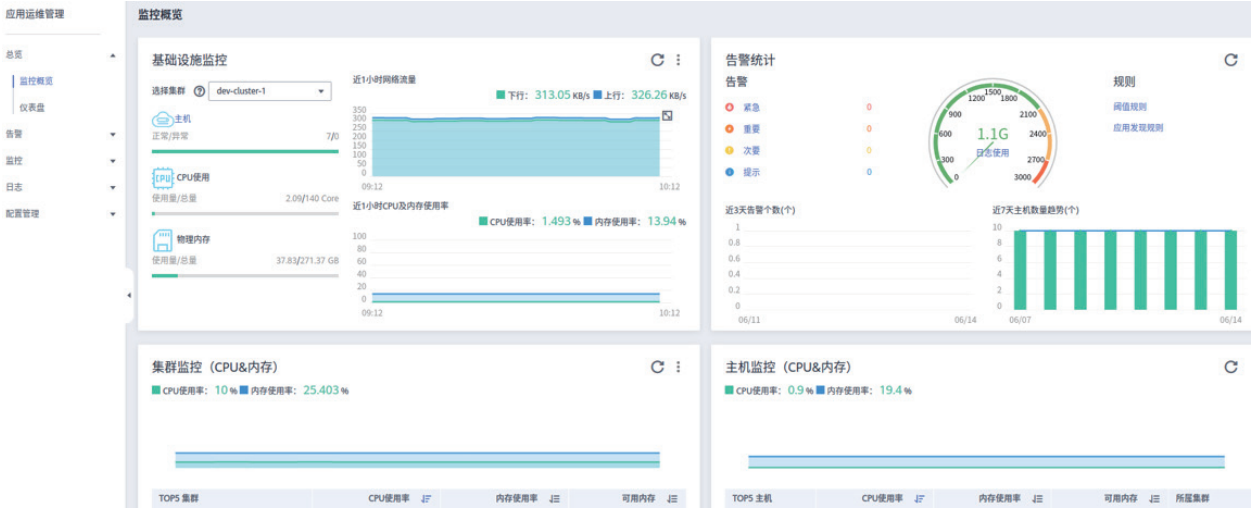


图 9：AOM 监控示例

是近线备份持久化应用日志以满足技术数据存储要求。

5.4.3 常规监控与容器监控

基于云平台的 AOM 功能，可实现容器集群、节点的常规监控和事件告警；基于容器集群所提供 prometheus 插件，可实现单容器集群的 k8s 组件及集群资源如 pod、service 等监控及数据展示，接入微服务暴露的 metrics 端点后还可以实现自定义指标监控。除此之外，为方便多容器集群统一监控，本次改造配套建设了集中式容器监控体系，单集群 prometheus 监控仅存储 7 天指标，集中式容器监控则对接多容器集群，实现指标按需冷热存储，并基于热数据实现容器监控告警。

5.5 中间件

本次上云信创试点改造，除了底层芯片（服务器）、操作系统、数据库等选型之外，在中间件上也做了适配与改造。

公共服务系统作为本次试点系统，在开发方面主要延续了原 java 语言，在开发框架上则选择了 spring boot，在中间件选型方面需要满足可内嵌。通过 POC 测试及商务采购，最终一期是基于东方通 tongweb 产品进行了适配，但公共服务系统作为基础服务支撑，需要考虑异构容灾部署，在完成一期基础上将推进宝兰德 bes 等适配，后续再进一步开展性能分析。

5.6 安全建设

当前信息公司的互联网类系统主要还是基于传统 IDC 进行部署，在安全方面主要是基于硬件安全设备如 IPS、WAF、流量分析、主机 HIDS，东西向防护如 iptables、区域防火墙，以及堡垒机、数据库审计、网页防篡改等产品构筑安全体系。

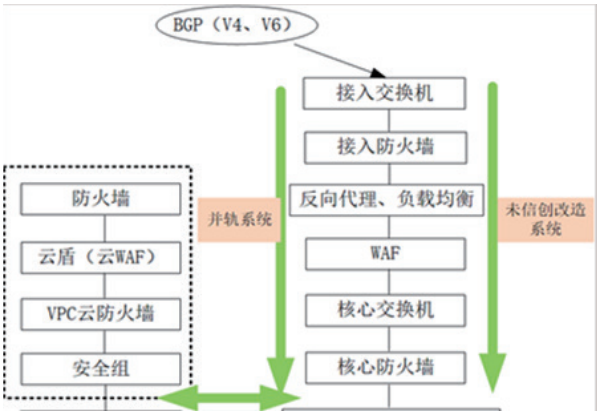


图 10：入口层双规并行

在上云信创改造路线上，则是云相关安全产品作为配套试点，在流量入口层面暂行双规并行策略，待云上相应安全体系建设完成且配套运行机制建立并完善后再实现单轨运行，随着应用系统的不断迁移改造再逐步下线传统 IDC 资源。

6 破茧而出 成果初现

经过各方的持续努力和共同协作，基于上证云信创基础设施进行的公共服务系统试点容器化改造如期完成，实现了传统云下服务与信创云上服务并轨运行，后续随着各业务系统的调用切换，预计 2023 年底将逐步实现公共服务系统信创云上服务的单轨运行。

6.1 上云、信创及容器化改造可行

本次试点改造，通过公共服务系统的先试先行，逐步建立了基于上证云信创基础设施进行应用容器化改造的相关指导原则，积累了相关经验，探索并初步建立了相应的技术运维服务体系及配套保障措施。

具有代表性的公共服务系统改造成功，验证了公司互联网类系统整体上云改造的方案具备可行性，也确认了未来几年公司基于上证云信创基础设施实现数字化转型的正确性，下一步待业务系统逐步实现信创单轨运行后，可进一步整合原公司租赁的数据中心，优化上交所数据中心整体功能布局。

6.2 资源利用率显著提高

本次试点改造后，基于 pod 级资源配额及 k8s 容器编排，可实现 pod 级别资源的按需调度，相比于传统虚机部署，整体资源利用率上获得显著提高。

通过进一步观察 pod 资源真实使用量及配置云平台提供的弹性伸缩 HPA/CustomedHPA 功能，针对 common-platform-ocr、common-platform-verify、common-platform-captcha、common-platform-email 及 api-gateway-mid 微服务配置了基于 metrics-server 采集指标（CPU 使用率）

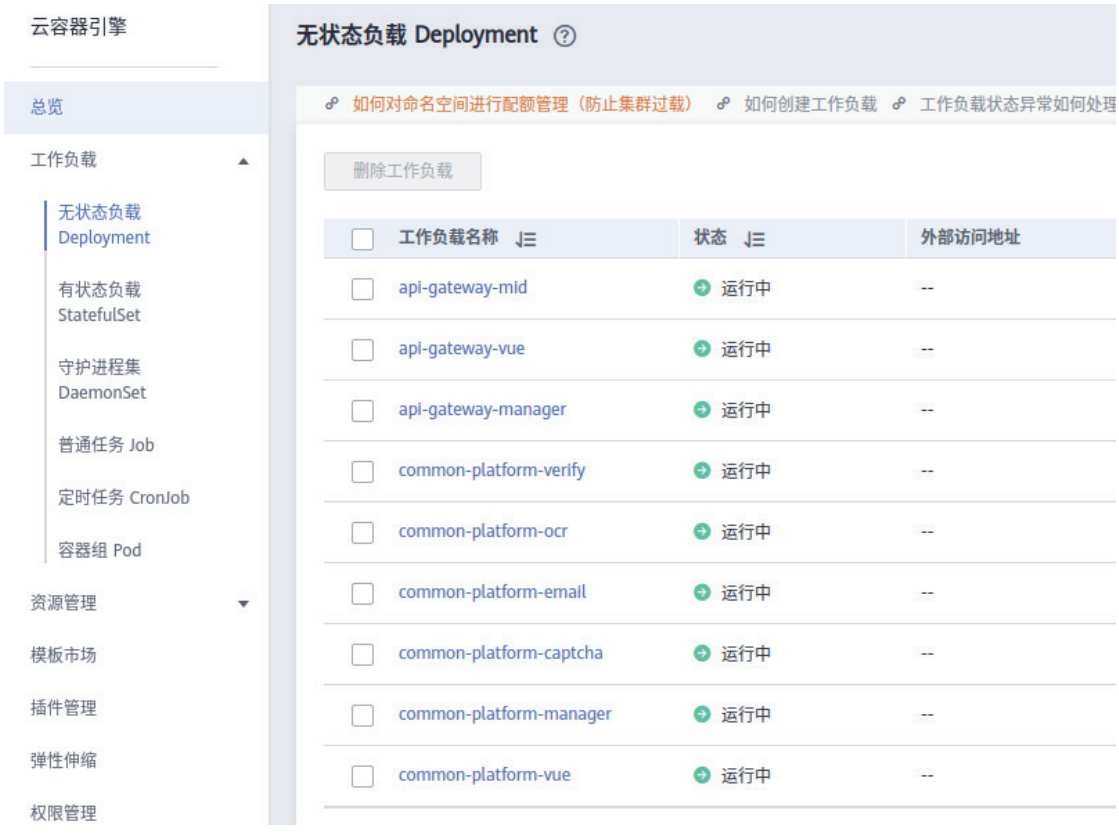


图 11：工作负载示例

表 2: 机器套餐配置及使用情况

机器配置			使用情况	
套餐名	CPU	内存	虚机部署方式	容器部署方式
机器套餐一	4C	8G	25	0
机器套餐二	8C	16G	3	0
机器套餐三	32C	64G	0	2

表 3：资源占用情况对比

部署方式	资源占用情况		
	机器总台数	总 CPU	总内存
传统虚机部署	28	124C	248G
云上容器部署	2	52C	104G

表 4：微服务资源配额

微服务部署单元				单元 CPU 需求	单元内存需求
名称	CPU	内存	实例数		
common-platform-manager	4C	8G	1	4C	8G
common-platform-vue	2C	4G	1	2C	4G
common-platform-captcha	4C	8G	2	8C	16G
common-platform-email	4C	8G	2	8C	16G
common-platform-ocr	4C	8G	2	8C	16G
common-platform-verify	4C	8G	2	8C	16G
api-gateway-vue	2C	4G	1	2C	4G
api-gateway-mid	4C	8G	2	8C	16G
api-gateway-manager	4C	8G	1	4C	8G
合计	-	-	14	52C	104G

的弹性伸缩策略。

在 CPU 使用率较低的情况下将 pod 实例都降到 1 个，在负载较高的情况，又可以通过 HPA 实现快速自动扩容，从而实现进一步提升资源利用率，降低资源占用。

6.3 自主可控有所突破

针对信息技术应用创新对芯片、操作系统、

图 12：ocr 微服务设置 HPA 策略示例

数据库及中间件的相关要求，本次改造基于上证云信创基础设施满足了对芯片、操作系统、数据库的相关要求，中间件方面则通过引入国产中间件厂商东方通、宝兰德等，通过商务招标购买了相关产品实现了针对原 websphere 等产品的替代。因此，本次试点改造在信创方面实现了芯片、操作系统、数据库及中间件的全面国产化替换。

在对外服务方面，通过国密网关设备接入，实现系统的国密浏览器访问支持；基于云的统一出口，实现了系统的 IPv6 服务支持。而在安全防护层面，则启用了云相关安全产品作为配套试点，如信创服务侧开启云 WAF 防护。

面对复杂的国际形势，通过本次试点系统改造真正做到了自主可控方面有所突破。

6.4 技术运维服务体系及配套保障有所产出

本次改造采用上证云信创平台所提供的智能监控体系，实现了传统的指标监控之外还纳入了容器监控及日志归集等功能，通过其交互性友好的界面大大方便了日常使用，减轻了运维人员的运维成本。

在容器镜像管理方面，通过平台所提供的容器镜像云服务实现了生产基础镜像和应用镜像的统一管控，大大提高了镜像管理和生产运行环境的安全性。

在制度规范管理方面，形成了《docker 容器安全配置指引》、《容器基础镜像制作及管理指引》和《应用镜像数据及日志持久化规范》等相关文档。

另外，在云上智能监控体系主用的基础上，通过指标接入公司本身的统一运维监控平台等方式，进一步完善和拓展了监控体系，真正做到监控更早发现、处置更加及时，有助于牢牢守住“安全运行”生命线。

6.5 人才队伍有所成长

本次试点改造中，前期组建的数字化转型人才队伍在负责基础设施、软硬件、系统架构、应用架构、技术路线选型以及业务系统规划建设等方面身体力行，在实践活动中都各有成长。

除了数字化转型人才队伍成员外，在系列开展的证券基金行业信息技术应用创新联盟培训、公司研发实战系列讲座、上云信创改造经验分享会等等活动中，聚集了大量的人才参与（尤其是新员工等的踊跃参与），形成了优秀的后备力量，可以有效支撑未来批量业务系统的改造开展。此外，在金融业信创人才培训认证及厂商国产化产品人才认证等方面也取得了进步，认证了一批优秀的信创专家。

不仅数字化技术人才方面有所建树，在项目管理方面，通过本次改造也锻炼了一批项目管理人才，同时在采购方面也培养了采购管理专业人才。

图 13：自研统一运维监控平台

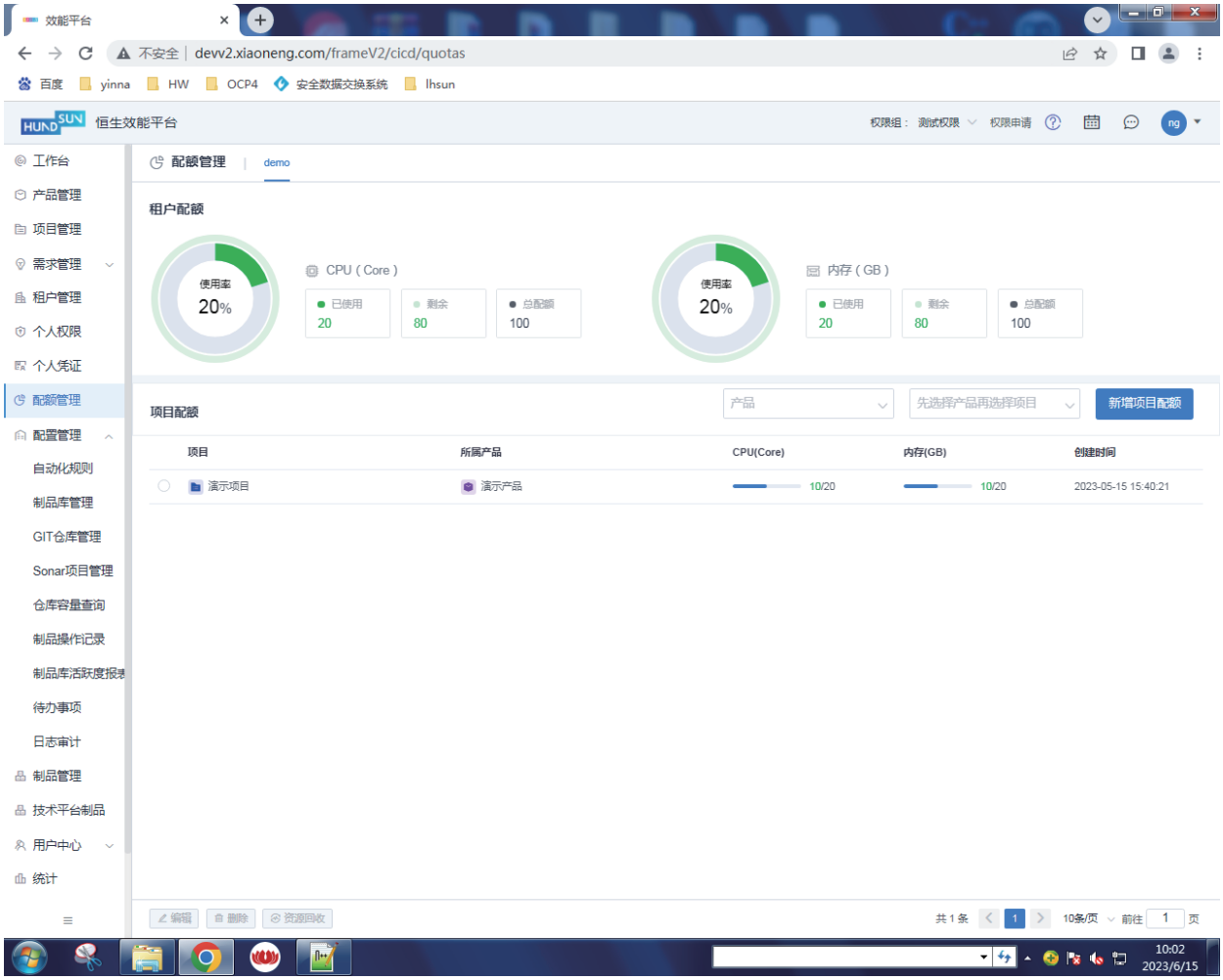


图 14：研发效能一体化平台

6.6 数字化治理水平有所提高

本次改造过程和党建、主题教育相结合，通过党建促进学习并完善信息技术规则制度体系，宣贯安全运行理念，开展安全生产大检查和大讨论，以实践不断提升 IT 管理水平。基于云原生、分布式或微服务架构、容器、容器编排等技术完善了应用的全生命周期管理，通过监控体系的完善夯实了技术运维服务管理。

在提升技术服务持续交付能力方面，通过打造的研发效能一体化平台，贯彻 DevOps 和持续交付理念，建设 CI/CD 流水线，通过电子流程实现了各个环节的联通（同时做好用户及权限管理和审批控制），有效提升了整体交付效能。

7 总结

本文主要介绍了基于时代背景下信息公司采用上证云信创基础设施试点开展公共服务系统容器化改造的探索与实践经验。信息公司规划了公司的信息技术系统建设路线，旨在加速数字化转型，不断完善技术总体布局，统筹基础设施板块统一治理，优化数据中心布局，紧跟《上海证券交易所“十四五”科技战略规划》，为打造“数字智能型交易所”贡献力量。本次试点系统的成功，为后续业务系统的批量改造奠定了坚实的基础，信息技术系统的建设也将更好地为信息服务赋能 -- 有助于充分利用证券市场相关数据及信

息，为多层次资本市场参与者提供专业、可靠、优质的产品及服务。

本次试点上云改造借鉴了云原生理念，采用了许多其关键技术，但也有未尝试的如动态赋能（服务网格）技术，在应用性能分析、调用链路追踪等方面还需要进一步探索。在持续运营方面，本次也还未完全实现，虽打造了研发效能一体化平台，但在运维工作中仍需要部分手工操作，这也将是下一步的努力方向。此外，随着 AI 兴起，GPU 等硬件资源的需求也成为部分系统的必选项，如何针对此类技术系统完成改造也是下一步需要不断探索的。

参考文献：

[1] 贺阮，史冰迪 . 云原生架构 – 从技术演进到最佳实践 [M]. 北京：电子工业出版社，2021.10.

[2] (美)CorneliaDavis，张若飞，宋净超 . 云原生模式 – 设计拥抱变化的软件 [M]. 北京：电子工业出版社，2020.8.

[3] 夏学平，邹潇湘，贾朔维，徐艳飞 .《“十四五”国家信息化规划》专家谈：加强数字化发展治理推进数字中国建设 [ER/OL]. http://www.cac.gov.cn/2022−02/16/c_1646636851356568.htm. 2022−02−16.

[4] 云技术之家 . 中国容器市场份额 TOP5：华为、阿里、IBM、腾讯、博云 [ER/OL]. https://baijiahao.baidu.com/s?id=1703240647530850948&wfr=spider&for=pc. 2021−06−22.

[5] 赞龙科技 . 从俄乌战争看中国信创产业的重要性 [ER/OL].https://baijiahao.baidu.com/s?id=1731795653938155539&wfr=spider&for=pc. 2022−05−03.

[6] 界面新闻 . 上交所总经理蔡建春：将加快打造数字智能型交易所，推进数字化安全运营 [ER/OL]. https://baijiahao.baidu.com/s?id=1748285955440063510&wfr=spider&for=pc. 2022−11−01.

分布式交易系统的监控设计与实践

蔡文豪、王伟、周尤珠、李鹤晨、王东 / 海通证券股份有限公司 上海 201201
E-mail：cwh8632@haitong.com

海通新一代分布式交易系统具有多节点多组件的特点，相比集中式交易系统的监控，面临多节点跨广域网、节点内多组件跨物理服务器、各组件监控指标各异、应用日志总量大等问题。本文以分布式交易系统的统一监控为目标，首先通过提取监控指标数据模型，解决各组件监控指标各异问题。然后本着对被监控对象侵入性最小的原则，设计监控数据流，解决监控数据量大和多节点跨广域网统一监控问题。进而为简化新增节点引发的监控数据流消费者部署、减少因组件指标调整或监控规则改变引发的频繁升级问题，我们自研了监控消费端组件，重点介绍其依赖的核心技术和应用层关键设计。最后给出性能数据和优化方向，总结分布式交易系统监控实践和探索的感悟。

关键词：分布式交易系统；统一监控；指标数据模型；librdkafka 网络库

1 引言

海通证券新一代交易系统采用分布式总线架构，每个系统实例（即单节点，下文简称交易节点或节点）由若干组应用组件构成。生产环境部署多套交易节点，存在不同节点跨广域网（如异地机房）的情况。为了能对各节点各组件的运行情况统一观测，在满足预设条件时及时预警，我们设计实现了统一监控系统，图 1 为运行图例。本文对面临的问题和解决方案做一个设计回顾，总结实践后的经验和思考。

1.1 监控目标

监控系统的设计目标是统一、实时、量化，借此保证分布式交易系统在生产环境的平稳运行。统一是指对所有节点、所有组件通过一个统一的入口监控。实时是指能在秒级延时内观测到被监控对象的监控指标。量化是指监控指标的数

图 1：统一监控系统运行图例

字化。

同时，在监控系统实时可观测、可统计的数据支持下，促进分布式交易系统在高可用、低延时等方面的迭代演进。

1.2 监控对象

本系统的监控对象是交易节点内各组件进程的内部技术和业务指标。

对于公司统建监控（e 海智维）已经聚合的应用系统无差别指标，如服务器操作系统磁盘、网卡、内存、CPU 等监控，不做重复监控。

2 监控指标

分布式交易系统面临各组件监控指标各异、应用日志总量大、多节点跨广域网如何统一监控等问题，本章通过对监控对象分析，设计监控指标的数据模型，解决指标各异问题。然后本着对被监控对象侵入性最小的原则，对多节点、跨广域网的分布式交易系统，设计监控数据流，解决统一监控问题。

2.1 数据模型

以一个交易节点为例，假设节点由 n 类组件构成，可视为 m 个进程。实际监控对象是这 m 个进程。

每类组件有自己的技术和业务指标，通过一个二元组 (group_id, idx_key) 及其 device_id 可表示节点内某个唯一的指标，其中 group_id 表示哪一类组件，idx_key 是表示指标名。比如 (25,pid) 表示交易核心的进程号指标。device_id 表示组成员编号，是指标的实例所以不必在本质的二元组内。

在实践中，我们发现有的组件情况稍复杂。比如报盘，group_id 为 53，假设二元组 (53,1001) 表示报盘的委托确认数，(53,1002) 表示报盘的委托成交数。由于报盘应用对渠道（按对接的交易所网关类型分类）、席位、交易所后端平台等参数可配置，我们希望被监控的指标能进一步细分，体现出不同报盘通道下的委托确认和成交数，比如我们想知道同一个报盘应用内上交所竞价、上交所综合、深交所竞价平台各自的委托确认数。所以唯一指标进一步调整为 (group_id，idx_

key, idx_subkey) 三元组，为了兼容用不到 idx_subkey 的指标，idx_subkey 被设计为数组的秩。用不到 idx_subkey 的指标，其值为 0。对于报盘组件，idx_subkey 就是报盘通道对应的通道数组下标。如，(53,1001,1) 表示通道 1 的委托确认数，(53,1003,1) 表示通道 1 的交易所接口类型，假设这里取值表示上交所竞价接口，(53,2001,1) 表示通道 1 的席位信息 S1，即上交所竞价的席位 S1，他们都是通道 1 的细化指标，细到可以满足监控需求。

确定数据模型后，各类组件就可以按这个三元组的形式列出指标清单了。

2.2 数据流

本着对被监控对象侵入性最小的原则，交易节点内各进程通过日志文件[1]输出 2.1 节设计的指标。然而单组件应用日志总量大（几十 GB）且不可控，如果监控指标追加到应用日志中，显然浪费传输带宽和解析算力。因此我们设计了指标数据通过独立的监控日志文件数据流发布，由监控消费端组件聚合订阅消费，是典型的发布 / 订阅数据流模型。按单组件所有指标打印一次需 1024 字节、每 5 秒打印一次计算，独立监控日志每日的固定输出量为 17.7MB。发布者除了从监控日志采集，还可从数据库等持久化介质采集。

分布式交易系统在生产环境，存在多个实例（即多个交易节点）跨异地机房部署。若只部署一个监控程序订阅所有机房的数据，存在数据量较大的跨广域网传输，在交易时间占用宝贵的网络带宽资源。为解决跨广域网节点的监控成本问题，同时考虑对所有节点统一监控入口的目标，我们设计了同一局域网内，采用发布 / 订阅数据流；跨广域网，把各局域网内监控消费者的处理结果，同步给主监控数据库，如图 2 所示。按每个指标物理表记录 248 字节计算，97 个指标数据量预估约 25KB/ 每 5 秒（因为源端每 5 秒输出一次指标快照）。同步处理结果的数据量可控，较直接消费原始指标的数据量大幅减少。

3 架构设计

在明确监控数据流后，为简化新增节点引发

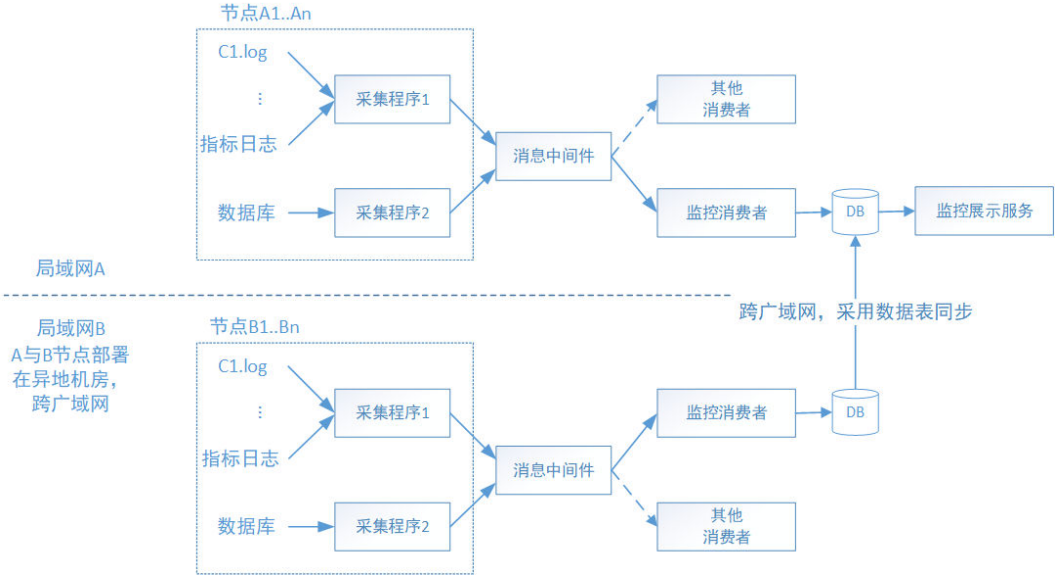


图 2：监控处理数据流

[1] https://www.cnblogs.com/timlong/p/6933616.html，就像武剑锋在 EzOES 高层设计理念分享里提到的 EzOES 监控接口 ASHR_status.txt

的监控部署问题，本章提出并重点介绍同一局域网内的监控发布 / 订阅环节数据流的系统架构，以及订阅端组件依赖的关键技术。

3.1 系统架构

基于 2.2 节的订阅 / 发布数据流，在局域网内监控发布 / 订阅环节架构如图 3 中，日志采集程序部署在每台被监控组件所在的服务器上。我们通过对采集程序的配置，增量扫描所需文件内容并发布到 Kafka 集群。

Apache Kafka[2] 是一款分布式消息总线，最初被设计用来解决 LinkedIn 公司内部的数据流问题。在海通证券，Kafka 作为基础软件设施，应用在实时盯盘系统 [3] 等多个场景。

订阅者采用我们自研的监控消费端组件。该应用基于 C++ 编写，使用了两项关键技术栈，librdkafka 网络库和海通自研的内存库。

3.2 librdkafka

监控消费端组件作为主题的消费者，需要和 Kafka 集群通讯并处理各种网络和来自 Kafka 集群的异常。为了降低开发难度，我们选用了 librdkafka。

librdkafka[4] 是一款 Kafka 的客户端库，底层使用 C 实现 Apache Kafka 协议，对应用层提供生产者、消费者和管理者的 API 接口，由 Magnus Edenhill 设计实现并作为主要维护者。

从消费客户端角度，librdkafka 内部设计实现了与 Kafka broker 之间的 TCP 通讯及应用层缓冲区管理、Apache Kafka 协议，提供了便于应用开发的配置、主题订阅、消息回调、Rebalance 事件回调、日志回调、统计回调等 API。

作为消费者应用，使用 librdkafka 主要关注 3 件事。

第一件是设置 bootstrap broker（启动代理服务器），也就是给谁发 metadata（元数据）请求。Metadata 请求由客户端发起，用来获取客户端关心的主题，应该从哪些 broker 请求。在 metadata 应答里，包括这些主题有哪些分区，每个分区有哪些副本，主副本在哪些 broker 等信息。任何一

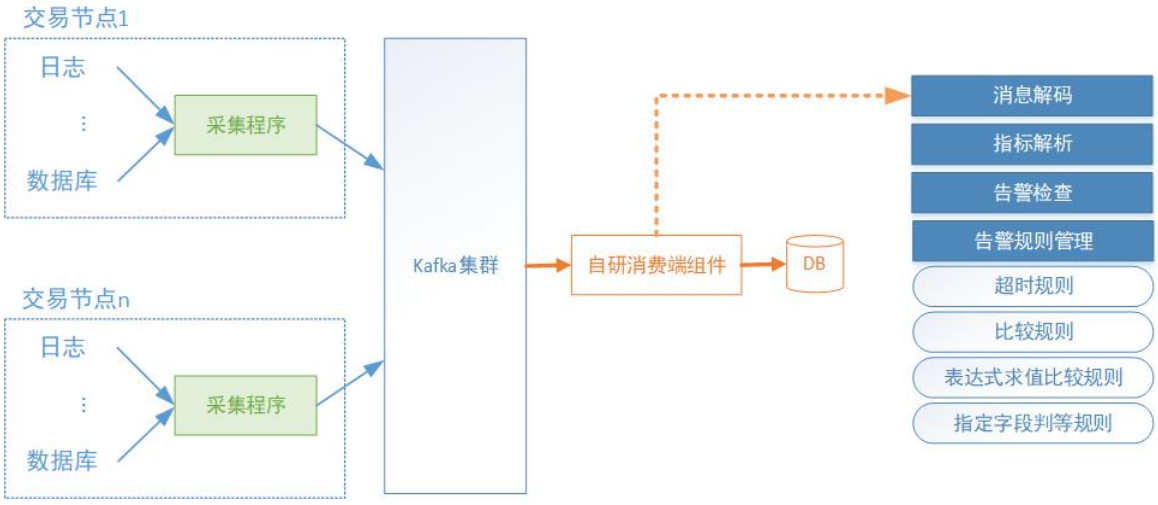


图 3：监控发布 / 订阅环节的系统架构图

[2] https://engineering.linkedin.com/distributed-systems/log-what-every-software-engineer-should-know-about-real-time-datas-unifying

[3] https://istock.ssetech.com.cn/wiki/doku.php?id=service:techmag:201803_030:08

[4] https://github.com/edenhill/librdkafka

台 broker 都有这些 metadata 信息。在自研应用设置 bootstrap broker 时，一般设置 1 个以上，作为备选。

第二件事是订阅主题。在 Kafka 集群中，每个主题是分区的，每个分区可以有多个副本，其中只有一个主副本。每个副本存储在不同的 broker。所有的主题消费和生产请求，都是给到主副本所在的 broker。从副本只负责从主副本同步。在 librdkafka 收到 metadata 应答时，就知道哪些主题从哪些 broker 获取，并且创建与该 broker 通讯的线程、定时器、线程间通讯队列等资源。

第三件事是获取订阅消息。librdkafka 使用 FetchRequest 请求获取订阅的消息，并通过 rd_kafka_consumer_poll 方法通知应用层。librdkafka 通过重定向队列，将各个 broker 通讯线程收到的消息重定向到应用层消息队列，简化了应用层的处理。

另外 librdkafka 还考虑了 Rebalance（消费端负载均衡）场景的处理。自研应用由于设计时对消息容量的控制，暂没必要考虑多个实例并行消费同一个主题。

通过使用 librdkafka，阅读其源码，我们对如何使用 Kafka 有了更深的理解。

3.3 内存库

监控消费端组件使用海通自研的内存库处理 Kakfa 消息，以及处理结果的持久化。内存库的使用极大地提高了消费端的处理能力，保证了监控的实时性。

这套内存库基于共享内存，支持将从物理库中加载关系表到内存，并按需构建内存表索引。索引支持哈希和红黑树两种，可分别用于匹配查询和范围查询。内存库支持事务语义，并提供跨线程异步持久化的能力。

在该组件设计中，对容量可预估的数据，如指标结果、告警规则等，采用内存库快速查询、更新。对告警通知等容量不可预估的数据，不采用内存表，而通过跨线程队列持久化。

4 应用设计

监控应用需要根据各种规则预警，为减少因组件指标调整或监控规则改变引发的频繁升级问题，我们希望未来不管新增多少组件，只要按照约定的指标数据模型，配置或者快速修改极少量的代码就能发挥监控作用。本章介绍在自研消费端程序的应用层关键设计。

4.1 消息解码器设计

消息解码器是对 JSON（JavaScript Object Notation）格式消息的解析。由于我们采用 JSON 作为 Kafka 消息的格式，按不同采集源根据约定的 JSON 格式解析提取消息。比如，日志采集消息解码器等。

对于约定的消息格式，这些（编）解码类，存在复用到其他项目的可能。

4.2 指标解析器设计

为了尽可能收敛 n 个组件 m 个指标带来的测试工作量，我们设计了指标解析器和告警检查器类。

指标解析器是对指标数据模型的解析，解析结果是生成 2.1 节数据模型下的 KV（Key，Value）键值对。

之所以设计这个类是因为我们注意到对于每一类组件，指标解析后的目的是将一个个 (K,V) 对的指标，做告警检查。从消息解码器获取的指标原始串，在应用层的处理代码可简化为类似迭代器的方式处理。

为此，我们设计了 CIParser 基类，定义 Next 方法作为接口。子类采用接口继承，不同组件的解析器根据自己的指标格式特点提供逐个指标。解析特点不同的组件，只要继承 CIParser，实现

独有的指标解析就可以了。

对于指标解析的测试，也收敛为对 CIParser 继承类 Next 方法的单元测试。

4.3 告警规则引擎设计

告警规则引擎是指对解析出来的指标，根据告警规则，做检查决定是否告警。

告警规则采用内存表存储计算，支持盘中修改后实时加载到内存表，目前支持的告警规则有，

1. 无规则。

2. 比较规则。可用于数值型和字符型比较。

3. 超时规则。用于指标多久没更新。

4. 与指定表指定字段的判等规则。如用于版本号不匹配的告警。

5. 表达式求值比较规则。即根据表达式求值，再使用比较规则。如用于内存表容量监控。

告警检查类的设计是无状态的，提供规则检查接口。为支持连续 n 次告警，将持久化告警信息接口分离出来单独提供。

回归测试只要在充足的指标和规则案例支持下，对检查接口执行单元测试就可以满足我们最初希望的测试收敛。

5 性能

本节我们针对监控应用，分别测算了 librdkafka 压力测试和自研消费端组件的业务处理性能。结果如下，

1. librdkafka 性能

压力测试选取了两类典型的组件，在没有对 librdkafka 做任何参数调优的情况下获得数据[5]。从结果分析，librdkafka（1.5.0）本身的吞吐量和延时足以满足监控需要。

2. 业务处理性能

表 2 测试环境业务处理性能数据

该数据为自研消费端程序在收到 librdkafka 消息后的业务处理耗时，是基于内存表单线程处理的数据。

注 1：以上测试使用 3 节点集群 Kakfa，3 副本。

注 2：测试服务器配置：CPU Skylake 2.6GHz * 8，内存 16GB，HDD 200GB。

注 3：组件 B 的指标比组件 D 少，但平均耗

表 1：测试环境 librdkafka 性能数据

消息字节数	消息数	librdkafka 消费耗时，吞吐量
1000	250000	786ms，317.76MB/s
120	1500000	2.20s，81.65MB/s

表 2：测试环境业务处理性能数据

组件名	指标个数	监控日志条数	平均处理耗时（us）
组件 A	2	2048566	12.023
组件 B	26	8804	963.487
组件 C	31	31911	209.667
组件 D	61	13215	548.485

[5] 未特别关注精确延时，没有计算分位值。如需可使用 librdkafka 提供的 hdrhistogram。

时长，因为组件 B 的指标格式解析更耗时。

6 优化

结合设计和性能测算的观察，自研消费端组件的性能有以下优化方向，

一是日志库的优化[6]。测试发现同步日志模式，在高吞吐量消息时会阻塞正常的执行流程，日志库需采用异步模式。该优化已完成，日志性能为 10 个线程每个线程写 20 万条记录，共耗时 0.7 秒。二是支持指标多线程并行处理。三是内存库锁机制优化，目前内存库的锁是进程级别的，多线程的优势被稀释，实现好内存库锁机制能极大提高并发处理能力。四是要设计好的监控业务模型降低业务复杂度。比如每个系统节点的相同组件指标告警规则可能不同，在没有设置规则时采用兜底设置，兜底还分好几个业务优先级。再比如，有些指标在告警事件处理后，后续的告警就算是从头开始了，而有些则不能一笔勾销。随着业务场景的需求增加，需要合理设计数据结构和查找算法，来提高监控处理性能。五是 librdkafka 参数调优，配置项多达 100+。

7 总结

应该说，基于 Kafka 设计监控系统在业界并不是新的思想，但不同于使用 Java 技术栈的 Storm，Kafka Stream 等架构，我们首先对被监控对象做了数据模型分析，把“大”数据控制在有用的“小”数据来消费，而这些“小”数据通过轻量级的消费者消费更合适。选择 C++ 技术栈的内存库，开源的 librdkafka，使得多节点交易系统的监控仅需部署若干个自研消费端组件就可以做到。

在本次实践中，我们使用了 librdkafka，Kafka 等开源代码和软件。在了解原理后，也给了我们很多新的灵感，比如日志采集程序的实现原理是否可以用在证券静态行情导入，Kafka+librdkafka+ 数据验证的组合是否能够用来解决系统间数据交互的痛点。同时我们也深刻意识到，用好开源技术关键是不断学习实践，拥有与之匹配的设计实现能力。

最后回到监控的初心，分布式交易系统监控的目的是能够第一时间发现异常，而这些异常也促使我们思考如何设计出能够自动从异常恢复的分布式交易系统。

参考文献：

[1] Neha Narkhede,Gwen Shapira,Todd Palino. Kafka The Definitive Guide. OReilly. 2017

[2] William P.Bejeck Jr.Kafka Stream In Action. Manning. 2018

[3] 邓俊辉 . 数据结构 (C++ 语言版). 清华大学出版社 . 2013

[6] 日志库优化已完成。性能提升为 10 个线程，每个线程写 20 万条消息，每条消息内容 100 字节，总耗时 0.7s。

兴业证券移动应用组件化技术探索和实践

刘洋、石良生、苏昌骏 / 兴业证券股份有限公司 金融科技部 福建 福州 350001
E-mail：hedongyang@xyzq.com.cn

兴业证券移动应用项目经过多年开发，是一个聚合大量业务复杂系统。在大团队并行开发、系统可维护性、系统复用性遇到了挑战。通过纵向按业务、横向按复用性进行组件化重构来应对挑战，在实际中我们取的如下成果：提供一系列高可复用的基础组件、改变原有开发模式团队可拆解为不同专职业务团队并行开发、从机制上面解决业务规模和系统复杂性成线性关系、有效提升兴业证券移动应用可维护性可靠性可复用可持续性。

关键词：组件化；开发模式；业务规模和系统复杂性

1 背景

兴业证券移动应用项目经过多年的开发，目前已经聚合了大量的业务，是一个复杂的系统，虽然有划分出多个模块，但存在如下不足：模块划分的不彻底，模块之间还是有耦合，模块代码没有物理隔离，随着业务持续增长，系统越来越复杂，可维护性越来越差。基于成熟移动互联网组件化思想的架构，成为实现高内聚、低耦合，解决业务模块依赖的主要方案。

基于上述分析，我们希望在移动应用中进行组件化改造，达成下述几个目的：改变移动应用开发模式，从机制上面解决业务规模和系统复杂性成线性关系问题；有效提升移动应用可维护、可靠性、可复用、可持续性，并提升团队成员的设计及技术能力；如需新建其它移动应用，可直接使用现有移动应用的基座，按需挑选SDK、基础组件、框架、业务组件进行复用，并从一开始在架构层面约束避免系统复杂性线性增长问题。

可预见面临挑战：需要兼顾产品迭代需求交付，产品迭代与框架优化平滑进行，遇到需求和组件化工程改造冲突情况。

2 目标

对移动应用进行组件化改造：重新梳理代码结构，提升项目工程可靠性、可维护性、可复用性达到提能增效；在这过程中培养团队成员设计能力并打造高水平技术开发团队、探讨出移动端组件化技术标准和规范、同时沉淀出可供公司、集团其它项目复用的 SDK、基础组件和框架，以及可供其它移动应用项目参考的组件化改造最佳实践。

3 技术方案

3.1 设计思路

以分而治之为指导思想，在面向对象设计原则约束下。通过制定统一的相关规范，各个模块按相应规范进行组件化改造。推进方式按先打通组件化全流程，再逐步补齐组件数量的思路进行。

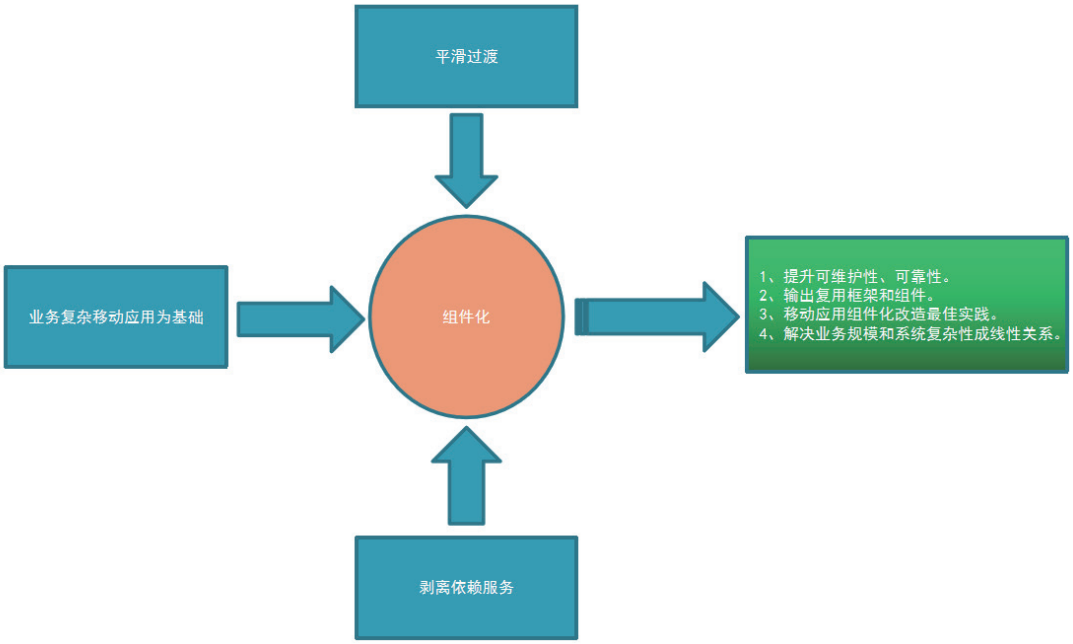


图 1：组件化目标示意图

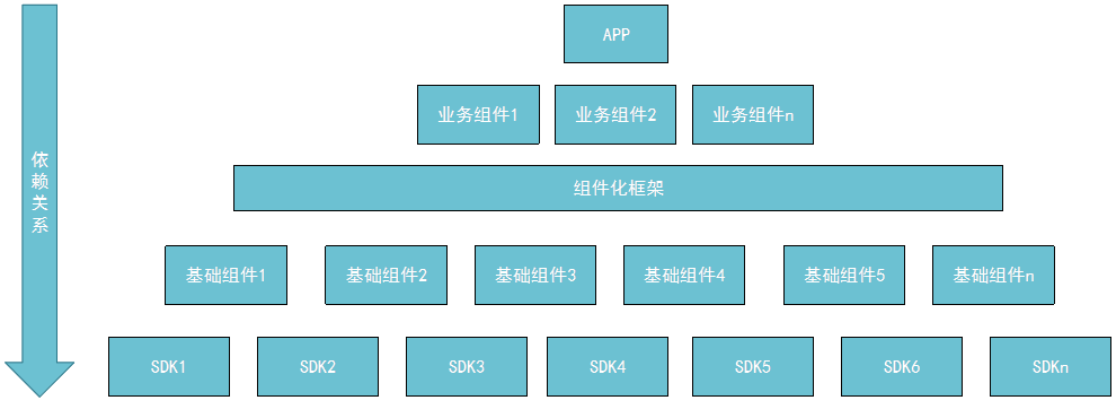


图 2：组件切分示意图

3.2 整体架构设计

在上述思路下进行整体架构设计，对移动应用架构设计如图 3。

系统实现的概念模型如图 4。

第一期进行必要的最小化改造，主要是红色背景部分，至于剩下其它视情况再决定是否要继续推进。

3.3 标准化

通过对不同类型的组件进行定义并制定标准。

图 3：整体架构设计图

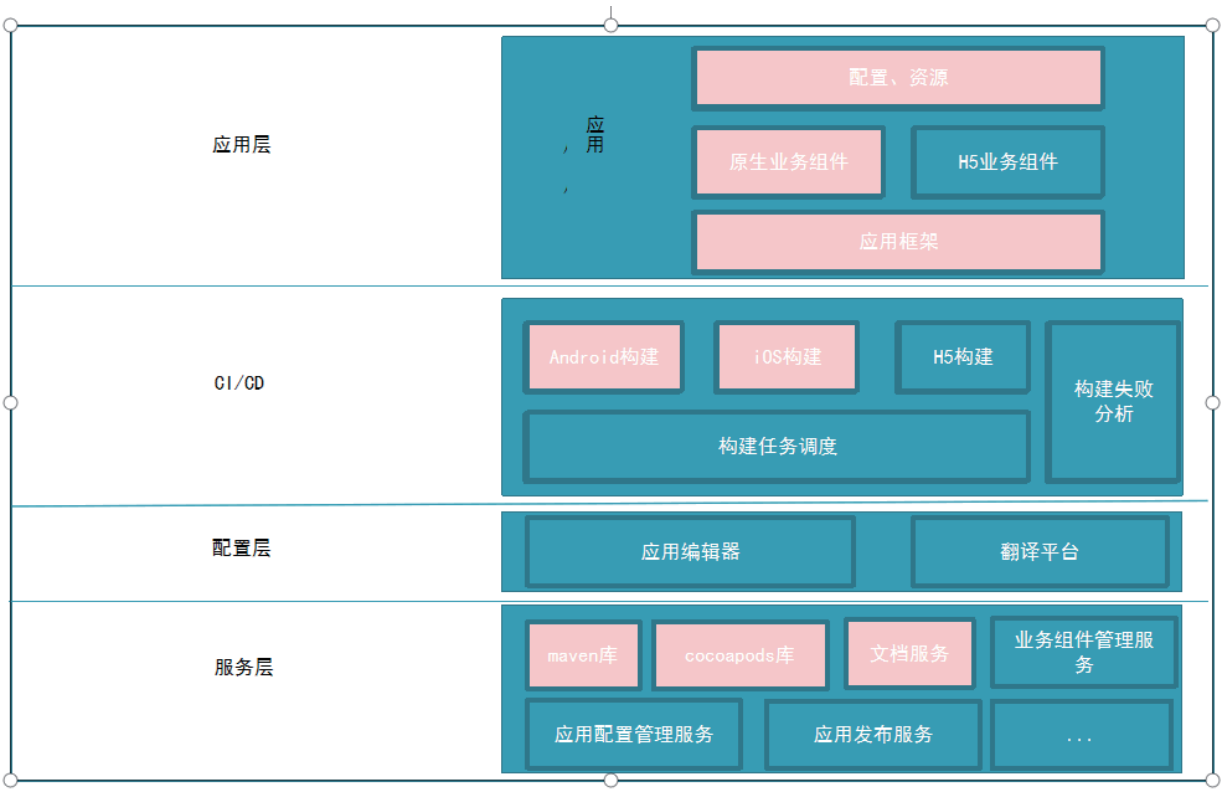


图 4：系统概念模型图

表 1：组件类型定义

组件类型	定义
SDK 组件	1、功能独立，能单独供外部复用。 2、只依赖系统。 3、现成第三方 SDK（如 OKHttp）。 4、无具体复杂业务。
基础组件	1、功能独立，能单独供外部复用。 2、无服务（大部分没有）。 3、可以有多个依赖。 4、无具体复杂业务。 5、可以有 UI。
业务组件	1、功能独立，有条件的供外部复用。 2、有 UI（大部分有）。 3、可依赖后台服务。 4、业务比较内聚(如登录业务组件)。

3.3.1 页面规范

通过规范页面代码结构，达到组件内代码结构一致有利于团队交流，从结构层面提升代码质量。Android 页面代码结构要求是 MVP，IOS 页面代码结构要求是 MVVM。

3.3.2 SDK 组件规范

SDK 组件定义：

功能独立，能单独供外部复用；无依赖网络服务；只依赖系统相关方法；现成第三方 SDK（如 OkHttp）无具体复杂业务；无横向依赖其它 SDK 组件。

SDK 组件约束：

1. 要说明 SDK 能力范围（领域）；

2. 能独立被依赖（如安卓 aar 发布到 maven，IOS 发布到 cocoapods）；

3. 要有 sample，使用教程，版本履历和总入口 readme.md（教程可以复用部分 API 文档，注意控制访问范围）；

4. 要有设计文档。（符合设计原则，根据实际情况简单写 or 完整写）；

5. 优先提供直接静态工具类方法（入口类）final，次之单例；

6. 对外提供简单同步方法；

7. 非入口类尽量不对外提供访问能力；

8. 依赖外部能力支持初始化方式反向注入；

9. 日志支持开关能力

9.1 可直接使用系统，建议每个 SDK 定义自己封装一个统一调用；

9.2 开关在具体实现里面统一控制；

9.3 安卓类名称建议：具体实现为 SelfSdk**LogUtil.java 类；

10. 资源都要添加模块前缀，避免冲突；

11. 版本号要遵循语义化版本；

12. 尽可能进行单元测试；

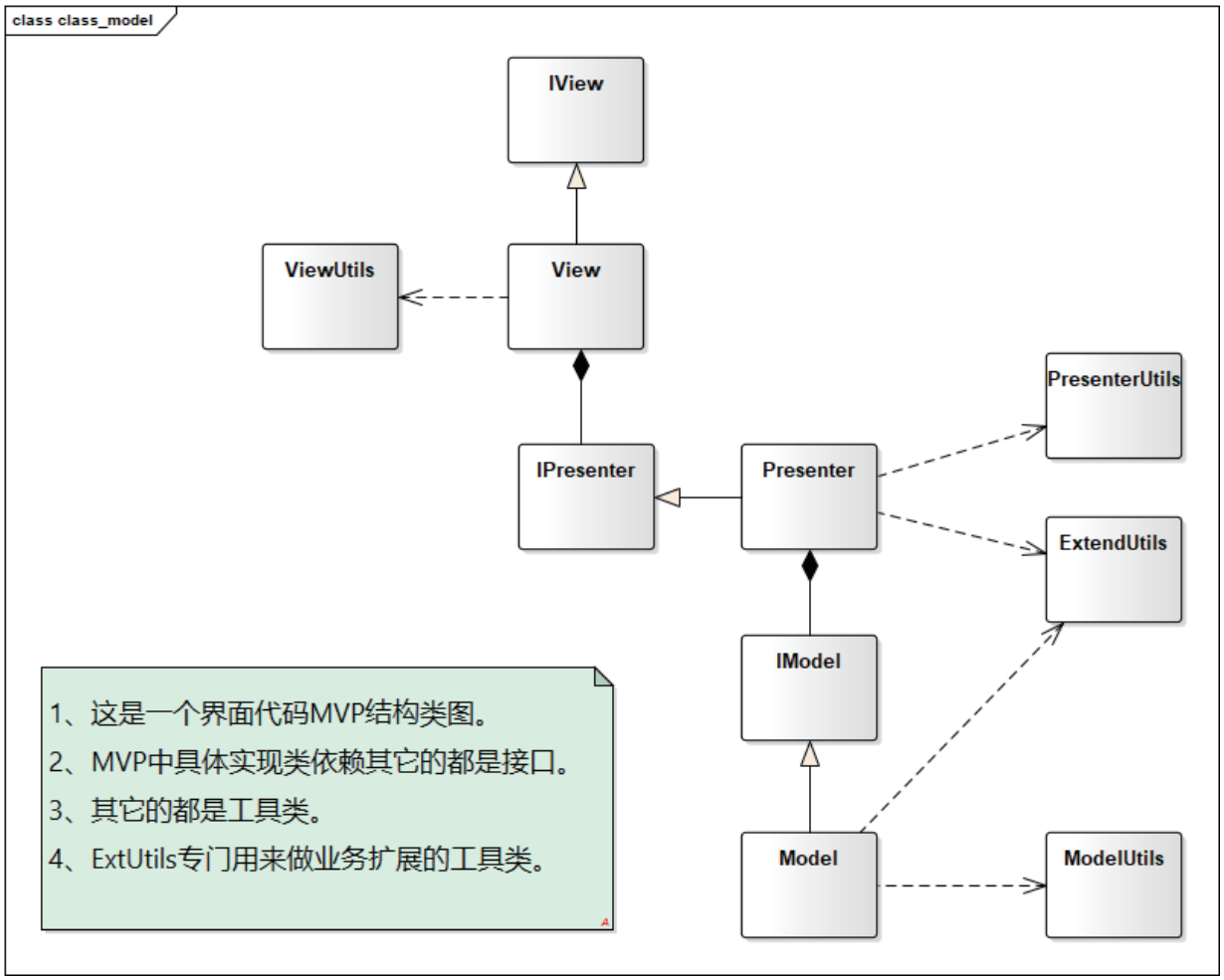


图 5：界面结构示意图

13. 对参数对象，如是 JavaBean 如必需参数，在构造函数中申明，

参数对象提供 toString eaual hashCode 方法，要有完整的注释；

14. 对外 API 每个版本要做兼容，如实在无法兼容要协调所有使用方一起修改；

15. 对外接口参数如超过 7 个，使用接口，否则直接传参；

16. 遵循一致的命名规范。

3.3.3 基础组件规范

基础组件定义：它和 SDK 组件的区别是它可以依赖 SDK 组件。

约束：在遵循 SDK 组件约束基础上多一个依赖 SDK，数量建议不超过 5 个，如超过需要进行团队内讨论其必要性。

3.3.4 业务组件规范

业务组件定义：功能独立，有条件的供外部复用；有 UI（大部分有）；业务比较内聚；可依赖后台服务；可依赖基础组件和 SDK 组件；无横向代码依赖其它业务组件，如有业务依赖通过框架解耦。

约束：在遵循基础组件约束基础上，可依赖基础组件和 SDK 组件；

如业务上面有依赖其它业务组件，要通过框架解耦，禁止直接代码依赖其它业务组件；服务端能力要抽象出来，具体实现通过接口反向注入形式提供；支持业务路由、事件、跨组件访问能力（通过框架）。

不同业务组件之间业务依赖通过描述符描述，并通过框架解耦；统一使用 APP 日志。

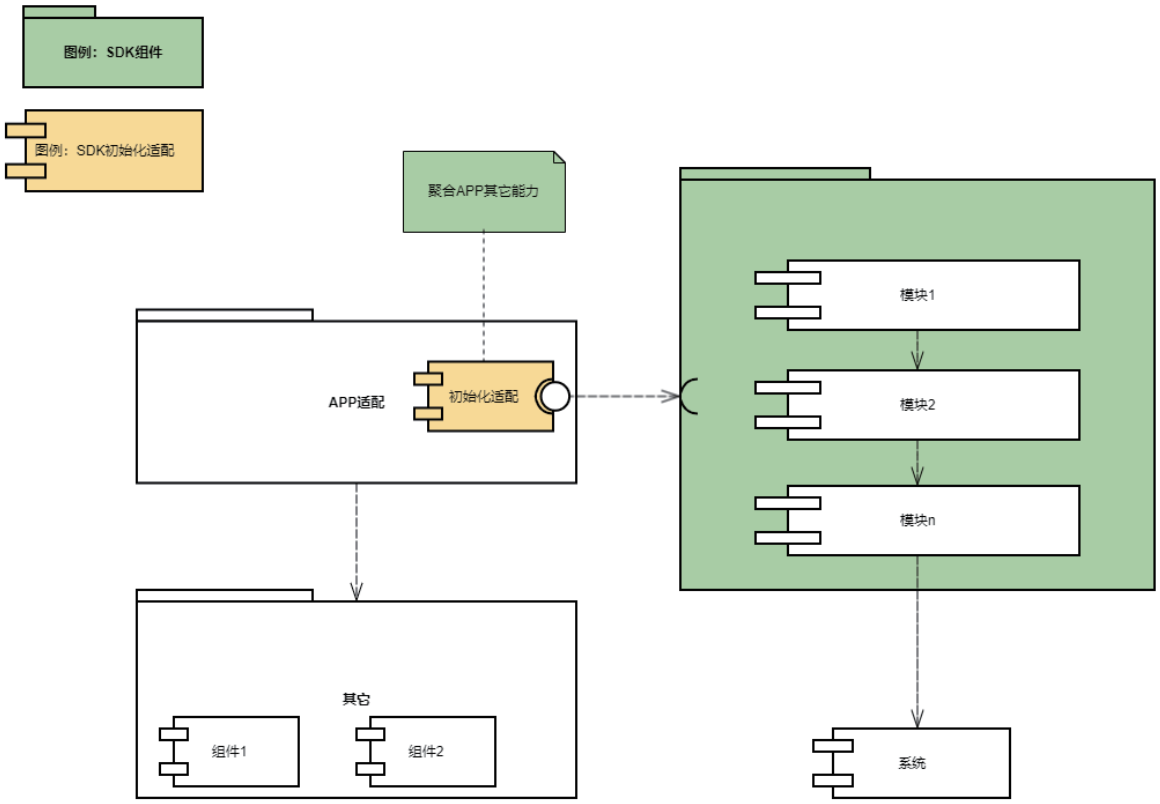


图 6：SDK 架构图

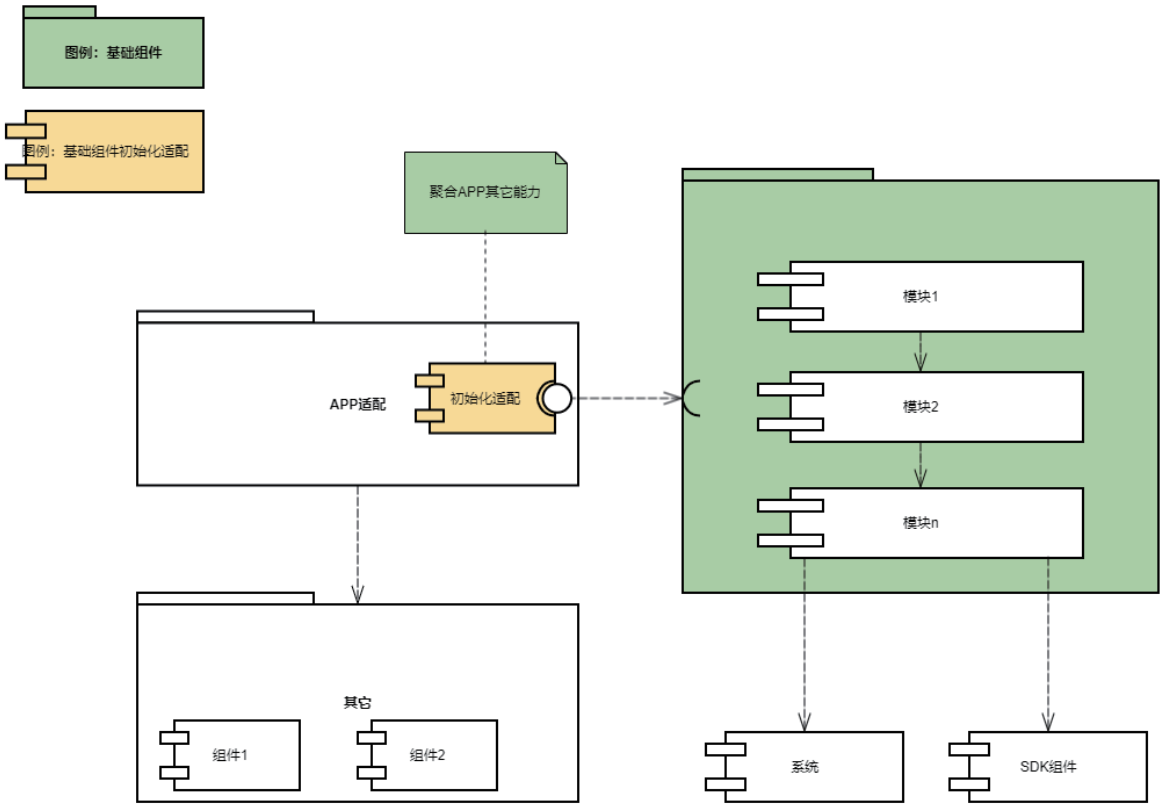


图 7：基础组件架构图

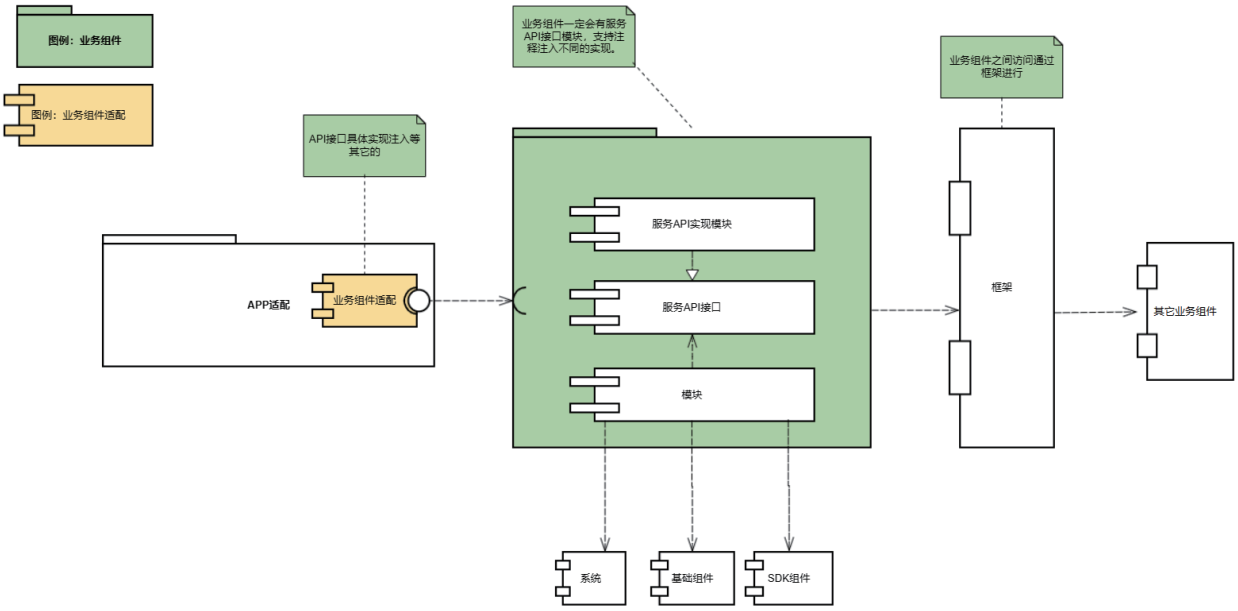


图 8：业务组件架构图

3.4 改造流程标准化

通过制定组件开发规范，防止组件代码腐烂，提升可维护性和代码质量。

4 成果及应用

目前，针对移动应用组件化改造分三个里程碑推进，分别是：“打通全流程”、“完善组件能力”、“改造深水区”。当前进展处于第二个里程碑的尾巴。提供两端可复用移动应用基座能力、大量 SDK& 基础组件、以及部分业务组件。在公司另外一个新的移动应用建设中采用上述基座能力和组件，有效减少重复建设成本和交付时间。

5 总结与展望

本文主要介绍了兴业证券在移动端应用组件化技术探索与实践，我们技术包含了各种组件的标准和改造流程，同时沉淀出一批可供公司、集团其它项目复用的基础组件。根据目前移动应用组件化改造情况来看主要是实现高内聚低耦合，如何做到高内聚低耦合可复用的组件一直围绕整个进程。我们总体上采用分而治之指导思想，并采用面向对象设计原则进行具体设计，在这大半年改造过程中证明上述方案是正确的。我们的解决方案目前仍然存在一些不足，我们计划在以下几个方面继续优化：

1、对于频繁变动业务组件，组件化后频繁打包导致开发效率低，我们计划通过开发相应的工具方式解决。

2、对于各个组件依赖关系治理，计划通过在 CI 增加检查工具 + 白名单形式进行控制，避免无意中提升依赖的第三方库版本。

3、在组件集成到移动应用时，由于版本迭代修改代码导致冲突，通过制定相应代码提交流程和组件集成自测规范来避免无意中引入 bug。

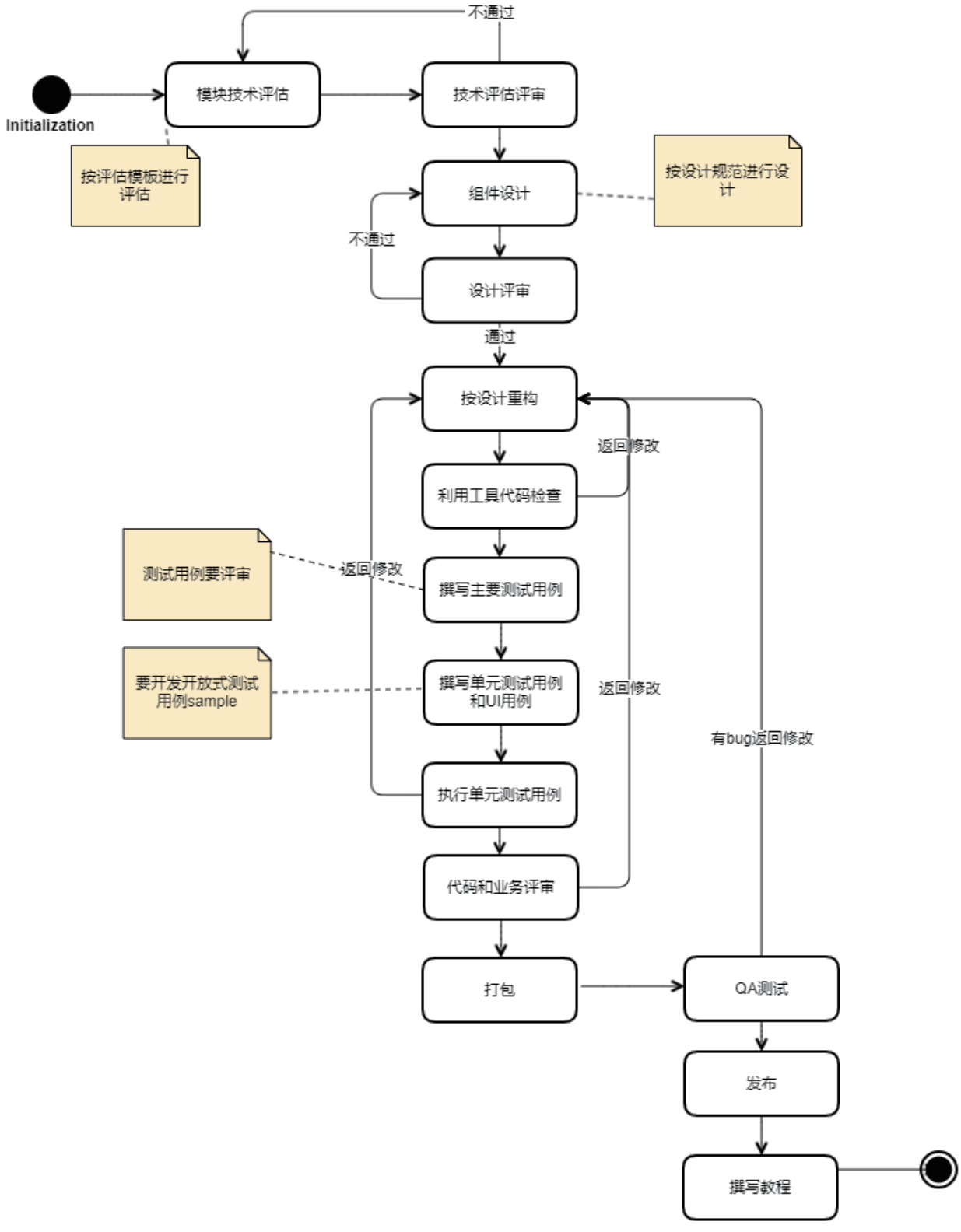

不通过
模块技术评估
技术评估评审
Initialization
按评估模板进行评估
组件设计
按设计规范进行设计
不通过
设计评审
通过
按设计重构
返回修改
利用工具代码检查
测试用例要评审
返回修改
撰写主要测试用例
要开发开放式测试用例sample
撰写单元测试用例和UI用例
返回修改
有bug返回修改
执行单元测试用例
代码和业务评审
打包
QA测试
发布
撰写教程

图 9：开发流程示意图

前沿技术应用

初探证券业IPv4/IPv6过渡阶段的安全防护

沙明、樊芳、陈治先 / 上交所技术有限责任公司 网络安全部 上海 200120
E-mail：msha@sse.com.cn

近年来，随着互联网技术的快速发展和应用，越来越多的网络设备接入互联网络，IPv4 地址面临地址空间耗尽、路由表急剧膨胀等严峻问题，且 IPv4 协议本身不提供任何安全机制、移动性差，严重制约了互联网的应用和发展。而下一版本互联网协议 IPv6，不仅能解决网络地址资源数量的问题，同时也解决了多种设备接入互联网的障碍，是 IPv4 有效的代替协议。本文将根据 IPv6 技术现状及面临的挑战，探讨证券行业 IPv6 安全防护思路，总结分享实践案例及经验。

1 概述

1.1 背景

全面构建 IPv6 网络基础设施的大幕已经拉开，金融行业是其中的重要一环，银行、证券等均是重点部署对象。2019 年 1 月 10 日，中国人民银行发布关于金融行业贯彻《推进互联网协议第六版 (IPv6) 规模部署行动计划》的实施意见（后文简称《实施意见》），对金融行业推进 IPv6 规模部署的原则、目的、范围、进度提出具体的要求。2020 年 3 月，工业和信息化部颁发了《关于开展 2020 年 IPv6 端到端贯通能力提升专项行动的通

知》，细化了部署 IPv6 的各项具体要求。2021 年 7 月 23 日，中央网络安全和信息化委员会办公室、国家发展和改革委员会、工业和信息化部此前印发的《关于加快推进互联网协议第六版（IPv6）规模部署和应用工作的通知》对外发布，明确了 IPv6 规模部署的加速化进程。

1.2 意义

在国家政策的推动下，加速推进 IPv6 的规模部署已成为证券行业的一项重要工作。自 2012 年开始，已有部分大型科技公司及互联网企业将 IPv6 投入商业用途，但由于我国金融证券行业对系统的安全稳定运行有着极高的要求，且证券公司业务系统与基础设施规模庞大且复杂，涉及范围广，升级改造过程面临较大挑战。面对上述情形，业界亟需一套完整有效的 IPv6 治理方案，以供参考借鉴，这对于推进金融证券业的 IPv6 规模部署有着重要意义。

2 IPv6 技术现状和挑战

2.1 IPv6 技术现状

IPv6 协议是国际上用以解决下一代 IP 问题而制定的协议标准，自本世纪初期开始投入测试使用，已有较多成熟案例。相较 IPv4 协议，IPv6 具有以下优势：IPv6 协议提供超大规模 IP 地址数量（约 3.4 × 1038 个），可以确保互联网在未来几百年里不必由于地址匮乏而中断发展；IPv6 协议加强了对扩展报头和选项部分的支持，一些不重要的扩展字段都被放在了 IPv6 头部之后的扩展头部中，使得 IPv6 的头部信息更加简单清晰，不仅能使路由器更高效地处理，而且可以按照不同协议要求增加扩展报头种类，对网络加载新的应用提供了充分的支持；IPv6 技术可以在路由表中用 1 条记录表示一个巨大的自治域网络和大量的子网，实现网络扁平化的基础上进行地址分层规划，取消多层 NAT 转换，显著提升网络效率；IPv6 协议还可以支持用于网络层认证与加密，保证了网络层端到端通信的完整性和机密性。

在当前的网络环境中，IPv6 协议并不能立即取代 IPv4 协议，在未来很长一段时间中，二者将共存在同一网络环境中。在这段过渡时期，主要有三种技术可用于二者的兼容：一是双协议栈技术，通过保有一个 IPv4 协议栈以及一个 IPv6 协议栈，实现并轨运行；二是隧道技术，将局部 IPv6 网络的 IPv6 数据包作为数据封装到 IPv4 数据包，使得 IPv6 数据包可以在 IPv4 网络中传输；三是网络地址转换（Network Address Translator，NAT）技术，通过 NAT 实现 IPv4 和 IPv6 主机的互通。

2.2 IPv6 带来的挑战

IPv6 协议采取了许多全新的技术，同时也带来不同于 IPv4 时代的挑战：一是 IPv6 协议本身存在的安全风险，二是将 IPv4 迁移到 IPv6 时采取双栈运行的安全问题，三是 IPv6 改造对网络设备有更高的安全要求。

IPv6 与 IPv4 都是网络层协议，网络层的许多脆弱性并不会因为 IP 协议版本升级而改变，所以很多 IPv4 网络层攻击方式在 IPv6 网络里继续可用；同时由于 IPv6 协议的新特性，也会产生新的安全威胁。

在协议并行期间采用的过渡技术也会带来一定的安全挑战。双协议并行情况下，NAT 转换也将继续存在，并不能显著提高互联网转发效率；双栈网络环境下同时使用 IPv4 和 IPv6 这两个通道，所有设备都要配置双栈协议，增加了系统的暴露性，且防火墙、防入侵设备等也必须配置双栈，原有 IPv4 网络攻击风险仍然存在，同时增加了 IPv6 网络层面暴露风险，也增加了网路管理工作复杂程度。

同时，DNS 服务器需要更多的安全投入。在 IPv6 网络中，DNS 解析请求主要来自于用户设备的 IPv6 地址，因此 DNS 系统日志将保存大量用

户 IPv6 源地址信息，一旦 DNS 系统日志被入侵，可能造成大量用户的 IPv6 真实地址数据泄露。

3 探索证券行业 IPv6 安全防护

证券行业对网络安全的机密性、可用性、完整性有着更高的要求。为贯彻落实《实施意见》要求，加快推进基于 IPv6 的下一代互联网在金融系统规模部署，促进互联网演进升级与金融领域的融合创新，证券行业 IPv6 安全防护应当具有以下特点：

3.1 合理采用网络协议优势预防风险

加强网络与有关应用安全质量评估，加强网络检测，持续开展攻击与预防演练；检测与填充网络协议漏洞，合理优化网络协议机制，调整报文头构成部分；充分利用协议安全加密机制，保障网络层面的安全 [1]。

3.2 应用系统改造与软硬件基础设施升级相结合

协同推进应用系统和软硬件基础设施 IPv6 改造工作，以应用系统支持 IPv6 连接访问为主攻方向，同步加快软硬件基础设施支持 IPv6 协议升级改造步伐。统筹考虑应用系统和软硬件基础设施的改造实施计划，防范集中式升级改造引发的安全生产风险。

部署支持 IPv6 网络的安全设备。一是在网络边界部署支持 IPv6 特性的较新的设备并对不必要的 IPv6 数据包进行过滤，这可以抵御大部分 IPv6 攻击，如利用 ICMPv6、多播协议、扩展数据包头进行的攻击。二是部署支持 IPv6 安全特性的 IPS 入侵检测设备对所有必要流量进行分析和监控，并在确认异常流量的情况下对其进行阻断。三是对于新建 IPv6 网络设备和安全设备进行功能性、应用标识指标、应用质量指标、高可用性、网络质量指标等的测试。

系统及时进行安全加固。目前来说，双栈运行模式是 IPv4 迁移到 IPv6 的主流过渡阶段，针对 IPv4 和 IPv6 提供同级别的安全防护。传统的安全手段对 IPv6 的防护并没有 IPv4 成熟，加固操作系统可以帮助抵御大部分 IPv6 安全威胁，如及时更新系统补丁软件、操作系统配置 ICMPv6 数据包过滤、关闭非必需开放的端口等。

3.3 加强共存网络与其他网络的综合防护

在对网络进行防护时，除应配置 IPv4 和 IPv6 的专用防护外，还应加强对过渡期间 IPv4 与 IPv6 共存和混合网络的 IPv4 和 IPv6 之间交叉威胁的防护 [2]。

实施物理隔离保障运行安全。一是选择 IPv6/IPv4 双栈互联网出口的实施方案，在新规划的网络区域新建互联网专线、负载均衡、防火墙、IPS、DNS 等基础设施。二是新建 IPv6/IPv4 网络区域与原有 IPv4 网络区域并行运行，两个区域互不干扰，隔离了技术系统变更对原有 IPv4 的互联网应用有可能带来的影响，避免了对关键设备升级带来的安全运行风险。

3.4 加强人才队伍建设

培训相关技术人才，全面加强 IPv6 网络技术与安全培训，借此提高相关工作人员的素质，加快 IPv6 专项人才培养。制定 IPv6 改造计划和实施方案，明确任务分工，相互协同，对可能出现的问题提前分析与研究，遇到问题时能有应对方案，及时发现并有效解决。

4 实践案例及经验

根据《推进互联网协议第六版（IPv6）规模部署行动计划》要求，上交所技术有限责任公司（后文简称技术公司）于 2021 年 11 月完成互联网信息系统 IPv6 试点改造工作。按照“分阶段、分网络域”的原则，对涉互联网信息系统采取以

下措施开展 IPv6 规模部署工作：一是利用 IPv4 地址和 IPv6 地址之间的相互转换来建立 IPv4 网络和 IPv6 网络之间的通信，实现相关网站和应用系统对外提供 IPv6 连接服务。二是新建 DMZ 区 nginx 服务的 IPv4/IPv6 双栈服务，对互联网用户提供 IPv4/IPv6 双栈服务。三是不断完善 IPv6 基础设施，建立健全 IPv6 运行状态、事件告警等指标的网络安全监测。四是建立完善 IPv6 建设维护相关的安全制度。

考虑到现阶段技术公司网站、应用开发均基于 IPv4 网络，将业务直接由 IPv4 部署到 IPv6 中，可能会出现网站、应用无法访问或出现错误的情况。在这过渡时期，为确保技术公司网站、应用系统安全、稳定、高效运行，采用一种 IPv4 主机和 IPv6 主机互联网通信的技术是很有必要的，这种机制不需要双栈协议支持也不需要隧道支持，只需要借助一个 转换网关，转换网关用于实现 IPv4 和 IPv6 报头的转换和地址的转换，同时根据不同的协议对分组进行不同的处理。以 NAT64 转换实现技术公司关应用系统对外提供 IPv6 的连接服务。

为同时向互联网用户提供 IPv6/IPv4 双栈服务，技术公司重点对 DMZ 区 nginx 服务的 IPv4/IPv6 双栈服务进行了改造和测试，同时保持 DMZ 服务器到内部中间件、中间件到内部数据库的 IPv4 访问网络不变，这样即可在不对应用系统架构进行重大调整的情况下，实现应用系统对外提供 IPv4/IPv6 双栈连接服务。

技术公司从应用层面、网络层面、安全层面对技术系统 IPv4/IPv6 双栈分别进行了测试和验证，发现存在一些技术上的限制。例如部分防火墙、IDS、IPS 不支持 IPv6，运营商不支持在现有的互联接入专线上增加 IPv6 地址等。为此，下一步，技术公司将采取分阶段改造的策略，先改造互联网接入网 IPv4/IPv6 双栈线路，再改造内部网络 IPv4/IPv6 双栈线路，确保 IPv4/IPv6 双栈线路和原有 IPv4 网络区域并行运行，两个区域互不干扰，隔离了技术系统变更对原有 IPv4 网络可能带来的影响，避免了对关键设备升级带来的安全运行风险 [3]。同时，加紧提升 IPv6 技术系统安全运维监控能力。对 IPv6 相关网络、系统和应用信息建立监控运维体系，部署日志系统，实时监控，有序推动 IPv6 安全运维功能的实施，最终形成与 IPv4 同等的安全运维管控能力。

建立完善 IPv6 建设维护相关的安全制度。制定较为完备的业务连续性方案，定期对相关的预案进行评估修订，并针对 IPv6 改造实施制定的应急预案，组织相关的应急演练，保障业务连续性。

IPv6 改造是一个相对长期的过程，在实施 IPv6 改造前需要提前准备、尽早规划和全面布局。制定有效的保障措施，对现有网络情况进行排查评估，选择合适的 IPv6 互联网接入方案，对应用系统进行充分验证，是保障 IPv6 改造工作顺利进行的必要条件。

5 总结

IPv6 是全球公认的下一代互联网商业解决方案，IPv6 取代 IPv4 已是大势所趋，推进 IPv6 规模化部署是一项长期的工作。作为新一代网络协议，IPv6 规模化部署、应用过程中势必存在一定安全隐患，要实现所有应用的迁移，并确保所有迁移应用能够安全稳定的运行，需要系统性的开展建设工作。本文从证券行业的实际情况出发分析了部署存在的挑战，并给出相应的应对措施。首要的是需要构建证券行业 IPv6 安全防护体系，推动加强运维支撑平台对 IPv6 的支持能力，其次不断完善 IPv6 基础设施，确保应用系统在完成 IPv6 迁移后能够安全稳定运行，最后持续推动应用系统 IPv6 改造，加快完成 IPv6 规模部署的工作任务。

IPv6 规模部署要坚持发展与安全并举，网络

安全系统同步规划、同步建设、同步运行，是构建安全可信 IPv6 网络的方法论。一个安全可靠、自主可控和高健壮性的 IPv6 网络，才是网络强国的真正的坚实基础。

参考文献：

[1] 翟昱，买尔旦·肉孜．关于 IPv6 网络带来的安全风险与应对措施 [J]. 数字通信世界 ,2021(08):153−154.

[2] 张连成，郭毅 . IPv6 网络安全威胁分析 [J]. 信息通信技术 ,2019,13(06):7−14.

[3] 黄锐，居宏伟 . 推进 IPv6 规模部署 保障金融网络安全 [J]. 金融电子化 ,2020(02):20−21.

证券行业网站业务系统IPv6网络安全风险及防护技术探索

周蒙、裘岱、宋良夏、王志玮、董小宇 / 上证所信息网络有限公司 基础架构部 上海 200120
E-mail：mengzhou@sse.com.cn

随着 IPv4 网络地址的消耗殆尽，互联网技术的发展需要新一代的互联网协议的支撑。近年来证券期货行业 IPv6 网络建设快速发展，核心机构、会员单位围绕网站业务系统的 IPv6 部署、IPv6 应用资源建设、双栈协议转换等方面开展了一系列工作。IPv6 网络下的安全问题也成为了一个亟需解决的热点问题。本文分析了广泛使用 IPv6 协议所面临的风险挑战，并探索实践了有效的安全防护策略。

关键词：网络安全、IPv6、安全防护

1 概述

随着移动互联网、物联网、工业 4.0 等新兴产业迅速发展，现今互联网协议第四版（IPv4）已远远不能满足万物互联、人工智能的发展需求了，而下一代互联网协议（IPv6）拥有 128 位的地址长度，广阔的网络地址空间甚至可以为地球上每一粒沙子分配一个 IP 地址，完全满足发展需要。且其网络数据报文基本由 40 个字节（Byte）的报头与扩展报头组成，相较于 IPv4 的报文结构更加简单。经过多年的发展，目前证券期货行业的 IPv6 的覆盖面已非常广泛，其技术应用完全成熟，已经成为行业中网站业务系统的通用基础，IPv6 技术本身具备较高的机密性和完整性。

2 证券行业网站业务 IPv6 发展现状

2019 年 1 月，中国人民银行发布关于金融行业贯彻《推进互联网协议第六版（IPv6）规模部署行动计划》的实施意见（后文简称《实施意见》），对金融行业推进 IPv6 规模部署的原则、目的、范围、进度提出具体的要求，以渐进式推进与增量式推进相结合的方式，完成面向公众服务的、支持 IPv6 连接访问的互联网应用系统的建设升级。在国家政策的推动下，学习银行和保

险等成熟的 IPv6 治理方案，证券期货行业加速推进 IPv6 的部署规模。

当前行业内各核心机构已基本完成 IPv6 部署，全面支持 IPv4 和 IPv6 网络运行。办公部分区域试点部署了纯 IPv6 网络环境，全面支持 IPv6 用户终端接入，同时各机构的门户网站和重要互联网信息系统均支持 IPv6 访问。但在当前的网络环境中，IPv6 协议并不能立即取代 IPv4 协议 [1]，在未来很长一段时间中，二者将共存在同一网络环境中。在这段过渡时期，主要有三种技术可用于二者的兼容：一是双协议栈技术，通过保有一个 IPv4 协议栈以及一个 IPv6 协议栈，实现并轨运行；二是隧道技术，将局部 IPv6 网络的 IPv6 数据包作为数据封装到 IPv4 数据包，使得 IPv6 数据包可以在 IPv4 网络中传输；三是网络地址转换（Network Address Translator，NAT）技术，通过 NAT 实现 IPv4 和 IPv6 主机的互通。

3 IPv6 网络面临的安全风险

随着 IPv6 技术快速普及和大规模的使用，给证券期货行业的网络安全工作带来了新的挑战，具体表现以下几个方面：

3.1 过渡方案的安全风险

目前各机构的 IPv4/IPv6 处于共存时期，不论是采用双栈、隧道、还是网络地址转换，都会引发新的安全挑战。在双栈环境下，各机构普遍运行情况同时并行着 IPv4/IPv6 两个逻辑通道，因为其中一个协议的漏洞引发的攻击，可能会通过支持双栈的网络节点，在逻辑上的两张网络中相互传播，进而影响整个网站业务系统核心网和办公网的正常运行。如图 1 所示，攻击者可利用 IPv6 协议栈漏洞，针对双栈网关发起 DDoS 攻击，导致 IPv4、IPv6 网络均受影响。此外，双栈意味着网站业务系统核心网络出口、数据中心防火墙、流控等防护设备要配置双栈策略，网络管理更加复杂，被攻击的概率也会相应增加 [2]。

通过网络地址转换（NAT64/IVI）实现 IPv4 与 IPv6 网络的互通，也会面临常见的 DDoS 攻击的风险。如图 2 所示，通过伪造大量 IPv6 地址发起地址转换请求，攻击将导致映射 IPv4 地址池快速被消耗，转换节点无法为网站业务的正常访问用户分配转换地址，进而影响网站对外的服务。

3.2 地址资源池隐私风险

面对庞大的 IPv6 地址资源，任何物理节点

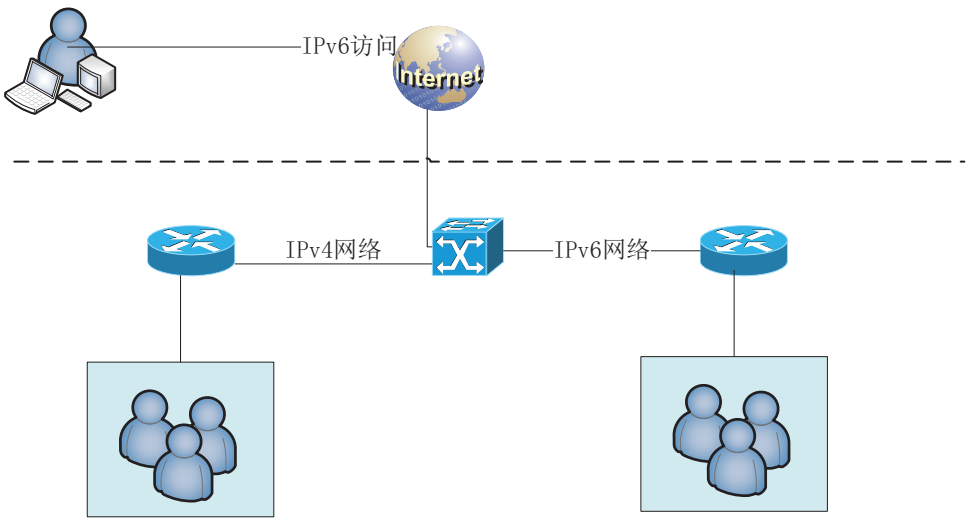


图 1：双栈环境风险

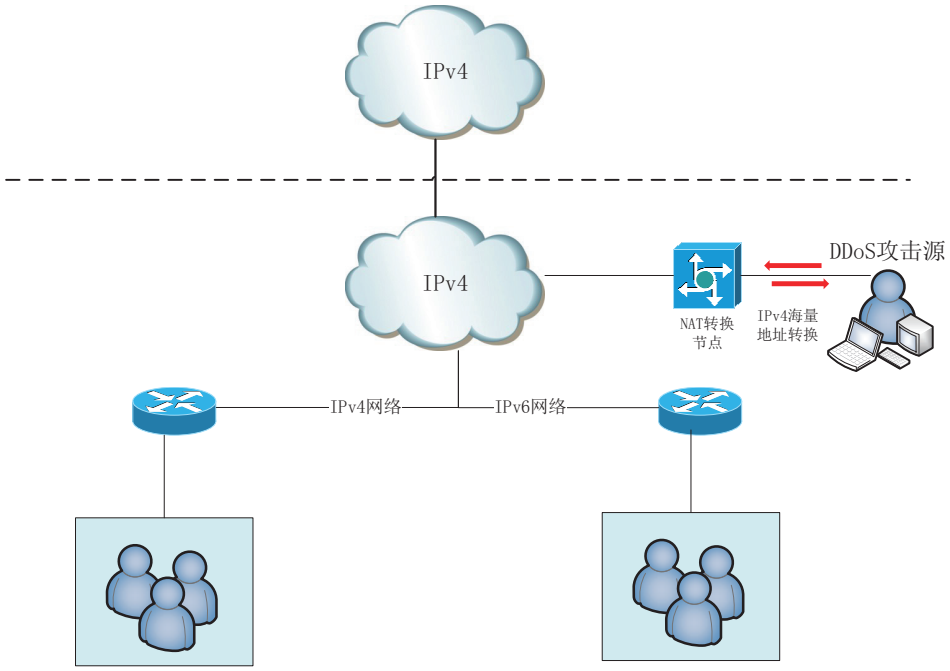


图 2：NAT 环境风险

都可以轻松获得 Internet IP 地址，从而促进点对点访问。如果所有网络设备都可以通过互联网访问，那设备上的个人用户就很容易通过网络被追踪。这个问题很早就被弗吉尼亚理工大学的研究团队发现了，在前期的一份报告中分析道：如果标准系统接收到 IPv6 地址，在自动配置地址（DHCP）的情况下，第三方将可以通过简单的指令（诸如 ping、traceroute 等）在全球范围追踪和监视目标用户。同时用户发送数据的接受方信息，也会被第三方监视者获取 [3]。

3.3 网络安全运维风险

一方面由于 IPv6 网络协议自身的限制 [4]，其不能严格禁止分片组装的特点，导致网络攻击者可以绕过防火墙。分片方式的差异，使得 IPS 需要重新组装报文，会不在其检查和审核范围内。IPv6 不能像 IPv4 一样随意丢弃报文中的字节，所以存在一定的安全隐患。另一方面 IPv6 发展对应着带宽时代发展，网络攻击成本相对较低，攻击容量较大，也便于获取，对网络安全架构设计得要求具有较大挑战。

3.4 管理安全风险

由于 IPv6 的地址数量众多，使得对于地址的统一分配和管理较为困难，证券期货行业内部对于地址的分配和管理尚未形成有效的管理规范。另外由于 IPv6 网络协议需要使用公私钥进行身份的认证，目前国内各行业对于加密算法的要求和规范未形成最佳实践和标准规范。同时，各机构忙于 IPv6 在使用和部署方面的技术问题，而忽视了相关网络安全知识的培训，部分终端使用者的网络安全意识缺乏，为 IPv6 网络的使用埋下了安全隐患。

4 IPv6 网络安全风险的防护策略

4.1 系统保障过渡阶段

结合各机构过渡阶段的方案，网络安全设备需同步支持纯 IPv6 环境下、过渡期间 IPv4/IPv6 双栈、NAT 转换及隧道等场景，充分考虑 IPv4 和 IPv6 两个逻辑通道的安全需求。日常网站业务所部署的网络环境中，防火墙设备需要支持 IPv4/IPv6 双栈协议及过渡时期的常用隧道技

术，同时其应用网关要支持 IPv6 解析，病毒检测、入侵防御（IPS）等功能所需的规则库均需要升级，以支持 IPv6 或 IPv4/IPv6 双栈场景。系统思维前提下，在日常运维中需要对各设备配置的 IPv4/IPv6 安全策略进行一致性校验。

4.2 注重隐私泄露防范

一方面，网络安全设备采用 Privacy Enhancing Technologies（隐私增强技术，PET）也对 IPv6 网络下的安全防护起到了重要作用。在启动 IP 协议头之后，可通过添加新的扩展协议头轻松扩展 IPv6。在网络层，隐藏客户端的 MAC 地址，IP 消息中的 MAC 地址可以采取部分加密机制。在应用层，操作系统的隐私技术可以对新操作系统中的机器隐藏某些 MAC 地址。另一方面，弗吉尼亚理工大学的研究团队发现采用 Moving Target IPv6 Defense (MT6D) 实现动态变化地址可以保护机构用户的隐私，使得通信双方实现匿名和安全的通信。MT6D 类似于跳频技术，当两台主机在 IPv6 网络中通信时，攻击者拦截到的是多个独立主机地址的配对，无法判别哪个地址配对才是真正的通信双方，继而无法简单的对某个地址进行攻击。

4.3 强化网络安全管理

基于 IPv6 流量展开多样化的安全监测，科学的应用真实源地址验证技术，提升整体网络安全威胁感知能力。对各网络安全设备进行升级改造，使得其能全面满足 IPv6 网络下的安全防护要求，同时加快推进软硬件系统以及相关应用系统能全面升级改造，做好网站业务系统和网络运行状态的集中监控。

4.4 动态完善管理策略

在今后 IPv6 带宽需求井喷式发展中，一是在基础环境建设规划中提前考量网络安全设备选型与网络部署，预留充足的网络性能提升空间。二是，技术人员要动态的加强 IPv6 网络安全体系的研究分析和应用，全面落实国密、商密的要求形成行业密码算法的最佳实践和行业标准。三是，技术人员要参与到专业化安全培训中，掌握更多 IPv6 知识，提升网络安全意识。同时在强化网络安全宣传基础上，普及更多 IPv6 知识，提高网络安全教育成效。

5 结语

随着 IPv6 在证券行业网站业务系统群的全面推广应用，各机构在梳理 IPv6 带来的安全风险的同时，结合过渡阶段的网络与系统的建设需求，遵循安全先行的原则，尽可能地在网站业务系统立项、设计、建设和日常的管理中完善 IPv6 安全防护。学习 IPv4 网络下安全态势感知和业务安全运行的保障能力和实践经验，收集和共享行业内外 IPv6 攻击态势、IPv6 攻击分布、IPv6 威胁预警等威胁情报数据，着力提升证券期货行业整体的 IPv6 主动防御水平。

参考文献：

[1] 高秋燕 . 基于高校的 IPv6 网络安全研究与实现 [J] . 信息系统工程，2021, (2)：55-56.

[2] 朱慧 . 高校 IPv6 网络安全风险及其应对策略研究 [J] . 网络安全技术与应用 , 2021, (07)：98-100.

[3] 张连成，郭毅 . IPv6 网络安全威胁分析 [J]. 信息通信技术，2019，13(06)：7-14.

[4] 戴仁杰 . IPv6 环境下的网络安全风险及防御措施 [J]. 中国高新科技，2020(15)：143-144.

[5] 黄锐，居宏伟 . 推进 IPv6 规模部署 保障金融网络安全 [J]. 金融电子化 ,2020(02):20-21.

信息资讯采撷

监管科技全球追踪

监管科技全球追踪

3 月 26 日，美国联邦存款保险公司（FDIC）发布公告称，第一公民银行股份公司与 FDIC 签订了对硅谷银行的收购协议，将承接硅谷银行所有存款和贷款。

3 月 28 日，为识别金融机构不当行为对消费者造成的损害，欧洲银行管理局（EBA）首次发布了一系列新指标。指标覆盖了 EBA 消费者保护范围内的银行产品，包括抵押贷款、消费信贷、存款、支付账户和支付服务，可以衡量消费者在购买金融产品或服务时面临的零售风险。

3 月 30 日，彭博（Bloomberg）推出为金融界打造的大型语言模型 BloombergGPT。该大语言模型专门针对各类金融数据进行训练，以全方位支持金融领域的自然语言处理任务。

据彭博微信公众号消息，该模型将帮助彭博改进现有的金融 NLP 任务，如市场情绪分析、命名实体识别、新闻分类和问题回答等。此外，BloombergGPT 还将释放更多新机遇，调动彭博终端上的海量数据，将人工智能的潜力带到金融领域。

4 月 4 日，加拿大隐私专员办公室（the Office of the Privacy Commissioner of Canada，OPC）宣布对聊天机器人 ChatGPT 开发公司 OpenAI 展开调查，该调查涉及“OpenAI 未经同意收集、使用和披露个人信息”的指控。

4 月 5 日，新加坡金融管理局组织亚太地区金融监管机构和云服务供应商举办首届金融行业云服务论坛，共有包括印尼、韩国等国家级金融监管机构在内的 20 余家机构参与，受邀各方就云风险管理实践交换意见。

近来，国内科技巨头纷纷布局大模型人工智能。3 月 16 日，百度发布 AI 产品“文心一言”；3 月 30 日，腾讯表示正在研发类 ChatGPT 聊天机器人；4 月 7 日，阿里云自研大模型“通义千问”邀请用户测试体验；4 月 8 日，京东集团表示将在今年发布新一代大模型“ChatJD”。

4 月 17 日，英国联合监管监督委员会（JROC）公布了其下一阶段对本国开放银行业务的建议：一是平衡可用性和性能；二是降低金融犯罪风险；三是确保有效的消费者保护；四是改善流向第三方提供商（TPP））和最终用户的信息流；五是将非全面可变经常性支付（VRP）作为推广附加服务的试点。

4 月 21 日，东京证券交易所宣布，计划 2024 年 11 月 5 日起将股票市场交易时间延长 30 分钟至当地时间 15 时 30 分，每天交易时长将达 5.5 小时（09:00-11:30，12:30-15:30）。

4 月 24 日，英国政府宣布成立 AI 特别工作组，并提供 1 亿英镑初始资金用于开发医疗和教育等领域使用的基础模型，包括 ChatGPT 等聊天机器人使用的人工智能。

4 月 23 日，为加快推进 IPv6 技术演进和应用创新发展，推进数字中国建设，工信部、中央网信办、发改委、教育部、交通运输部、人民银行、国资委和能源局联合发布关于推进 IPv6 技术演进和应用创新发展的实施意见。

4 月 28 日，中共中央政治局召开会议分析研究当前经济形势和经济工作，中共中央总书记习近平主持会议。会议指出，要夯实科技自立自强根基，培育壮大新动能。要重视通用人工智能发展，营造创新生态，重视防范风险。

5 月 9 日，新加坡金融管理局（MAS）宣布将从 2024 年下半年起分阶段推出“洗黑钱和恐怖主义融资资料分享平台”（Collaborative Sharing of ML/TF Information & Cases，COSMIC），帮助金融机构共享可疑非法交易相关信息，合作打击金融犯罪，包括星展银行在内的 6 家银行将率先使用该平台。

5 月 10 日，国家区块链技术创新中心正式投入运行。该中心位于中关村国家自主创新示范区，由北京微芯区块链与边缘计算研究院牵头建设。

5 月 18 日，国家金融监督管理总局正式挂牌，中共中央政治局委员、国务院副总理何立峰出席仪式并揭牌。国家金融监督管理总局领导班子成员和干部职工代表，中央金融委员会办公室以及人民银行、证监会、国家外汇管理局等有关部门负责同志参加仪式。

5 月 23 日，国际证券监督机构（IOSCO）公布了首个监管加密资产和数字市场的全球标准。拟议的标准涵盖处理利益冲突、市场操纵、跨境监管合作、加密资产托管、运营风险和零售客户的处理。

5 月 24 日，国务院总理李强签署国务院令，公布修订后的《商用密码管理条例》（以下简称《条例》），自 2023 年 7 月 1 日起施行。

5 月 31 日，新加坡金融管理局（MAS）和 Google Cloud 签署了一份谅解备忘录（MoU），就基于负责任的生成式人工智能（AI）解决方案进行合作。

5 月 30 日，为了指导和帮助个人信息处理者规范、有序备案个人信息出境标准合同，中央网信办编制了《个人信息出境标准合同备案指南（第一版）》，对个人信息出境标准合同备案方式、备案流程、备案材料等具体要求作出了说明。

6 月 7 日，环球银行金融电信协会（SWIFT）宣布其正与十余家金融机构和市场基础设施合作，测试信息传递网络的使用，以指导一系列公共和私有区块链网络上的代币化价值转移。

6 月 13 日，美国网络安全和基础设施安全局（CISA）发布第 23-02 号指令，要求所有联邦文职机构在 14 天内从公网上移除特定的管理接口或实施有强制访问控制的零信任架构。

6 月 28 日，欧盟委员会通过了两项提案，确保公民和企业能够在整个欧元区继续使用欧元纸币和硬币进行支付，并为欧洲央行未来可能发行的数字欧元制定法律框架。

6 月 29 日，网信办与香港特区政府创新科技及工业局签署《关于促进粤港澳大湾区数据跨境流动的合作备忘录》，在国家数据跨境安全管理制度框架下，建立粤港澳大湾区数据跨境流动安全规则，促进粤港澳大湾区数据跨境安全有序流动，推动粤港澳大湾区高质量发展。

7 月 5 日，巴哈马保险委员会正式成为国际保险监管机构（IAIS）多边谅解备忘录（MMoU）成员之一，该备忘录是一项国际监管合作和信息交流的协议。

2023年二季度《交易技术前沿》征稿启事

《交易技术前沿》由上海证券交易所主管、主办，以季度为单位发刊，主要面向全国证券、期货等相关金融行业的信息技术管理、开发、运维以及科研人员。2023 年二季度征稿主题如下：

一、云计算

（一）云计算架构

主要包含但不限于：云架构剖析探索，云平台建设经验分享，云计算性能优化研究。

（二）云计算应用

主要包含但不限于：云行业格局与市场发展趋势分析，国内外云应用热点探析，金融行业云应用场景与实践案例。

（三）云计算安全

主要包含但不限于：云系统下的用户隐私、数据安全探索，云安全防护规划、云安全实践，云标准的建设、思考与研究。

二、大模型技术

（一）应用技术研究

主要包含但不限于：大语言模型 /AIGC 的数据处理和治理、可解释的大语言模型、用于大语言模型 /AIGC 的神经网络架构、训练和推理算法、多模态大语言模型等。

（二）应用场景研究

主要包含但不限于：基于大语言模型的智能客服、语音数据挖掘、柜员业务辅助等。

主要包含但不限于：金融预测、反欺诈、授信、辅助决策、金融产品定价、智能投资顾问等。

主要包含但不限于：金融知识库、风险控制等。

主要包含但不限于：机房巡检机器人、金融网点服务机器人等。

三、数据中心

（一）数据中心的迁移

主要包含但不限于：展示数据中心的接入模式和网络规划方案；评估数据中心技术合规性认证的必要性；分析数据中心迁移过程中的影响和业务连续性；探讨数据中心迁移的实施策略和步骤。

（二）数据中心的运营

主要包含但不限于:注重服务，实行垂直拓展模式;注重客户流量，实行水平整合模式;探寻数据中心运营过程中降低成本和提高服务质量的途径。

四、分布式账本技术（DLT）

（一）主流分布式账本技术的对比

主要包含但不限于：技术架构、数据架构、应用架构和业务架构等。

（二）技术实现方式

主要包含但不限于：云计算 + 分布式账本技术、大数据 + 分布式账本技术、人工智能 + 分布式账本技术、物联网 + 分布式账本技术等。

（三）应用场景和案例

主要包含但不限于：结算区块链、信用证区块链、票据区块链等。

（四）安全要求和性能提升

主要探索国密码算法在分布式账本中的应用，以及定制化的硬件对分布式账本技术性能提升的作用等。

五、信息安全与 IT 治理

（一）网络安全

主要包括但不限于：网络边界安全的防护、APT 攻击的检测防护、云安全生态的构建、云平台的架构及网络安全管理等。

（二）移动安全

主要包括但不限于：移动安全管理、移动互联网接入的安全风险、防护措施等。

（三）数据安全

主要包括但不限于：数据的分类分级建议、敏感数据的管控、数据共享的风险把控、数据访问授权的思考等。

（四）IT 治理与风险管理

主要包括但不限于：安全技术联动机制、自主的风险管理体系、贯穿开发全生命周期的安全管控、安全审计的流程优化等。

六、交易与结算相关

（一）交易和结算机制

主要包含但不限于：交易公平机制、交易撮合机制、量化交易、高频交易、高效结算、国外典型交易机制等。

（二）交易和结算系统

主要包含但不限于：撮合交易算法、内存撮合、双活系统、内存状态机、系统架构、基于新技术的结算系统等。

投稿说明

1、本刊采用电子投稿方式，投稿采用 word 文件格式（格式详见附件），请通过投稿信箱 ftt.editor@sse.com.cn 进行投稿，收到稿件后我们将邮箱回复确认函。

2、稿件字数以 4000-6000 字左右为宜，务求论点明确、数据可靠、图表标注清晰。

3、本期投稿截止日期：2023 年 8 月 31 日。

4、投稿联系方式 021-68607129，021-68602496 欢迎金融行业的监管人员、科研人员及技术工作者投稿。稿件一经录用发表，将酌致稿酬。

《交易技术前沿》编辑部

证券信息技术研究发展中心（上海）

附件：投稿格式（可通过电子邮件索要电子模板）

标题（黑体 二号 加粗）

作者信息（姓名、工作单位、邮箱）（仿宋 GB2312 小四）

摘要：（仿宋 GB2312 小三 加粗）

关键字：（仿宋 GB2312 小三 加粗）

一、概述（仿宋 GB2312 小三 加粗）

二、一级标题（仿宋 GB2312 小三 加粗）

（一）二级标题（仿宋 GB2312 四号 加粗）

1、三级标题（仿宋 GB2312 小四 加粗）

（1）四级标题（仿宋 GB2312 小四）

正文内容（仿宋 GB2312 小四）

图：（标注图 X. 仿宋 GB2312 小四）

正文内容（仿宋 GB2312 小四）

表：（标注表 X. 仿宋 GB2312 小四）

正文内容（仿宋 GB2312 小四）

三、结论 / 总结（仿宋 GB2312 小三 加粗）

四、参考文献（仿宋 GB2312 小四）

电子平台

欢迎访问我们的电子平台 http://www.sse.com.cn/services/tradingtech/transaction/ 。我们的电子平台不仅同步更新当期的文章，同时还提供往期所有历史发表文章的浏览与查阅，欢迎关注！

联系电话：021-68607129
021-68602496
投稿邮箱：ftt.editor@sse.com.cn

ITRDC
证券信息技术研究发展中心（上海）
上海证券交易所 SHANGHAI STOCK EXCHANGE
中国上海市杨高南路388号
邮编：200127
公众咨询服务热线：4008888400
网址：http://www.sse.com.cn