



MicroD

基于执行轨迹追踪的微服务故障诊断系统

王 焘、王子勇、石钟浩、张文博

wangtao@otcaix.iscas.ac.cn

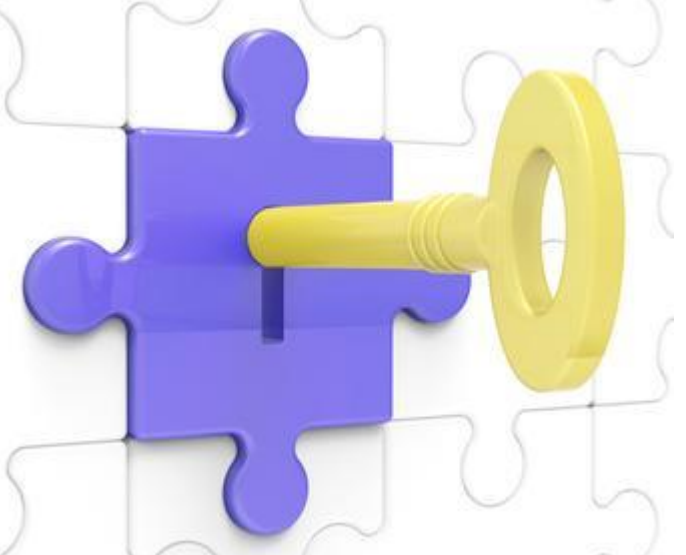


● 隐性故障

“微服务突然失效，什么原因导致的？”

“微服务系统性能衰减，如何有针对性的优化？”

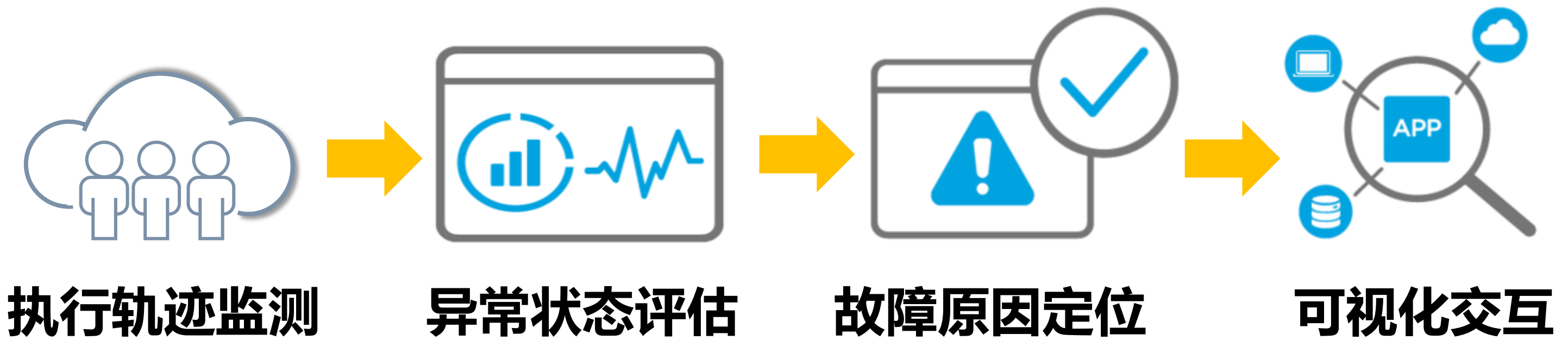
“微服务系统未报错但不能正常工作，哪个环节出了问题？”



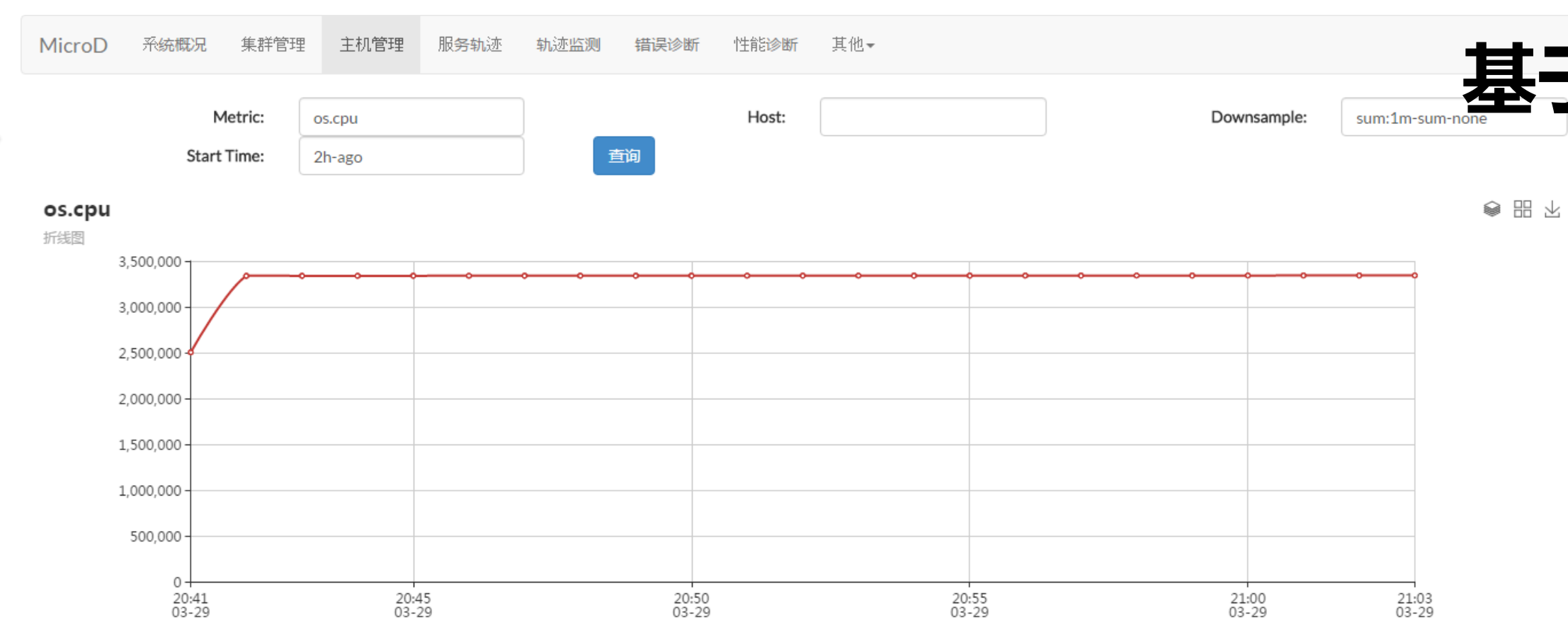
● MicroD

监测请求处理的执行轨迹，评估服务组件的异常程度，定位故障相关的执行逻辑，辅助分析性能瓶颈，实现了微服务系统的细粒度故障诊断，并提供可视化交互界面。

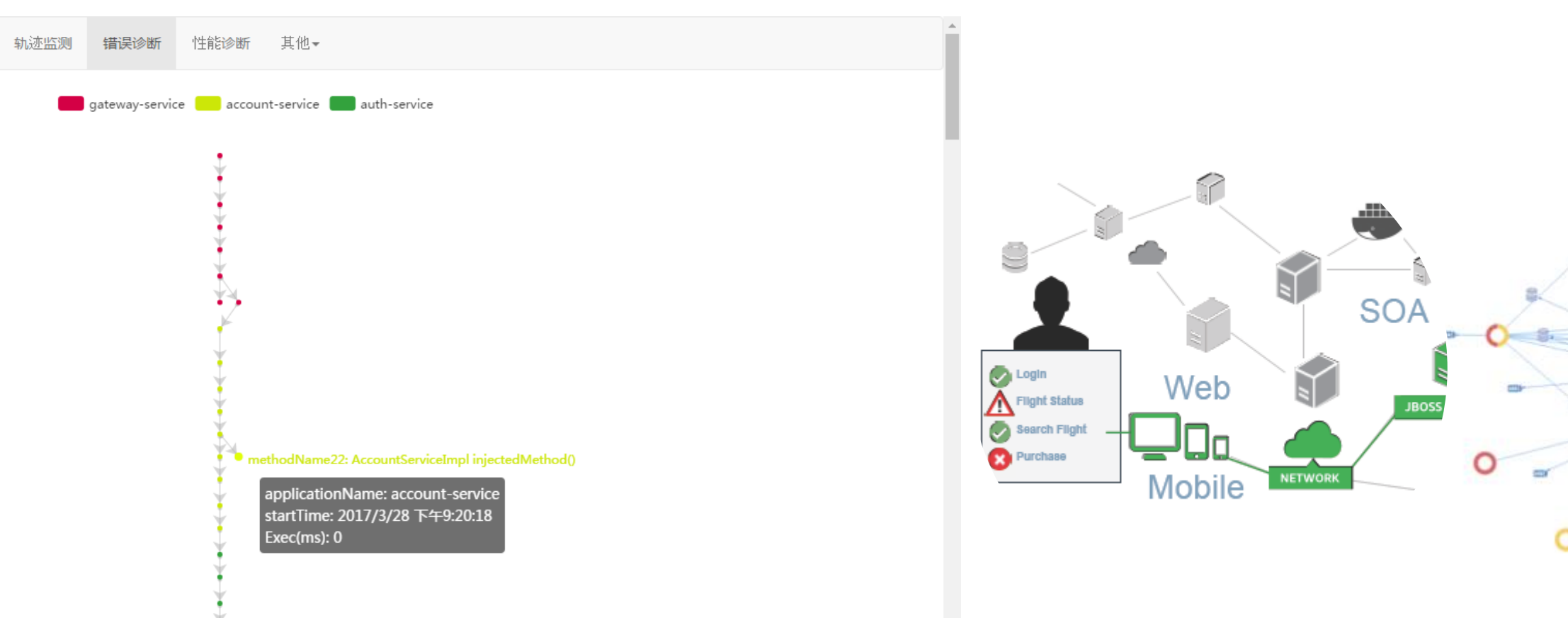
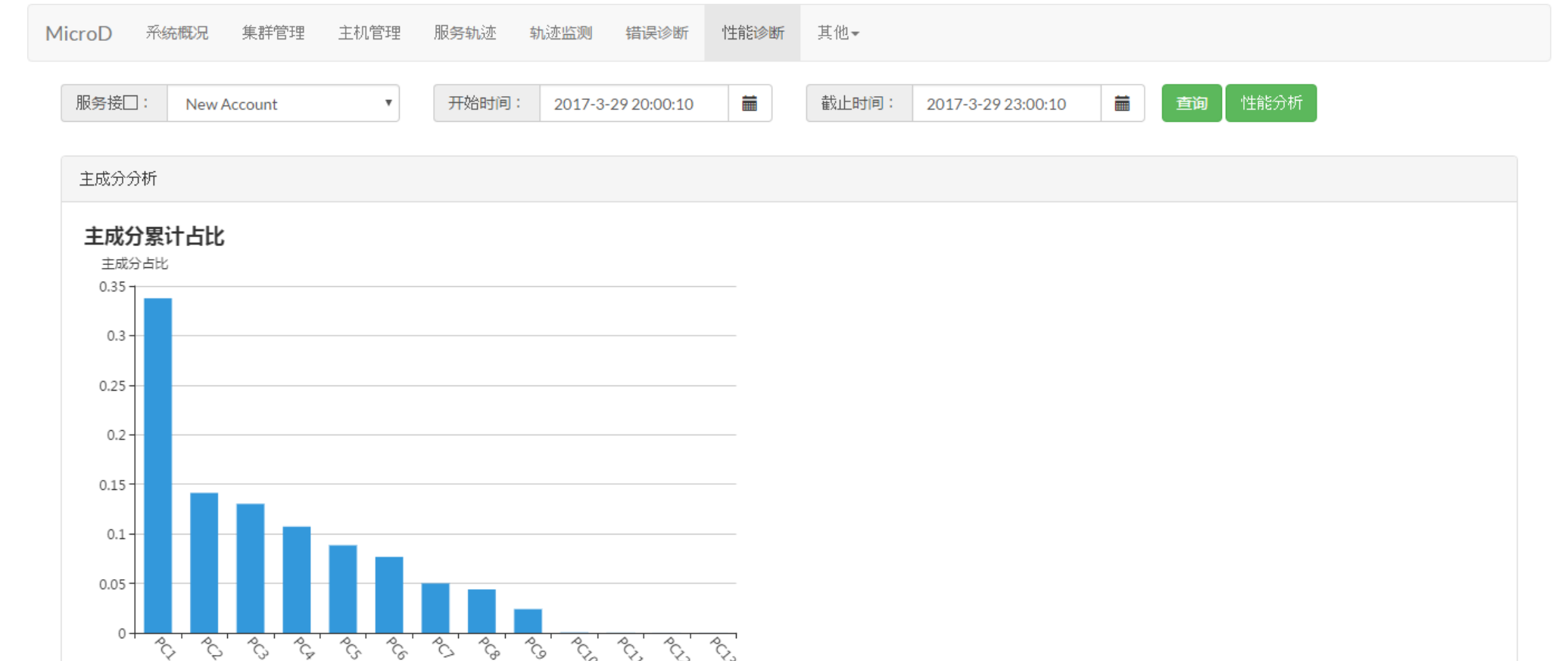
服务、组件、方法等细粒度、可视化、智能化隐性故障原因定位，有效保障微服务系统的可靠性。



关键技术



基于大数据的运行状态监测



执行轨迹监测与构建



可视化展现

#	Method	Service	PC1	PC2	PC3	PC4
0	TomcatServlet.process()	API Gateway	0.0032305588431254466	-0.11500270625861385	0.0590014423254226	-0.16118374568766902
1	StandardHostValve.invoke(Request request, Response response)	API Gateway	0.0983683316542855	0.7500676520247467	0.254942200808982	-0.1057853269790954
2	FrameworkServlet.doPost(HttpServletRequest request, HttpServletResponse response)	API Gateway	0.08647620494747761	-0.4866749378643234	0.57879905200268	-0.0170397146820747
3	HybridCommandQueue.doPost()	API Gateway	0.05089385383493093	0.06029007478941025	0.061526632860610555	-0.25109145254364616
4	HybridCommandQueue.doPost()	API Gateway	0.15805698753354172	0.27422862397841735	0.43965093984046477	0.6879625480101889
5	HttpServlet.doPost(HttpServlet request, HttpServletResponse response)	API Gateway	0.0882749623508883	0.257075087245121	-0.07073781469913695	-0.4164845191781013
6	HttpRequestHandler.execute(HttpRequest request, HttpServletResponse response)	API Gateway	0.189194305742049837	-0.07804881407856552	0.5667984934616406	-0.426977493299112
7	TomcatServlet.process()	Account Service	0.6949268591030014	-0.049830896340278	0.1409205027273136	0.1534867342513487
8	StandardHostValve.invoke(Request request, Response response)	Account Service	0.00706811012544681	-0.0099841569709330038	-0.000107735978176468	-0.01643221094945308
9	FrameworkServlet.doPost(HttpServlet request, HttpServletResponse response)	Account Service	0.00173080492070981	0.000586019385384817	-0.00345719711105218	-0.0061470296961274
10	AccountController.createNewAccount(User user)	Account Service	0.004434397273660175	0.003702419249493313	0.00557927133452726	-0.00312889571726811
11	AccountServiceImpl.create(User user)	Account Service	-0.005008113368444364	-0.005200607068881583	-0.00642192253318396	0.01620238351247494
12	HybridCommandQueue.doPost()	Account Service	-0.39702879620115145	0.11144241004953169	0.13987468405265321	-0.1304569458948125
13	AsynchronousInvocation	Account Service	-0.0027959774380362794	0.0007848057045741668	0.0009850329862862905	-0.0009187321449963471
14	HybridCommandQueue.doPost()	Account Service	0	0	0	0
15	HttpRequestHandler.execute(HttpRequest request, HttpServletResponse response)	Auth Service	-0.38304890903097074	0.10751830152666084	0.1349495191212218	-0.1036630386449965
16	TomcatServlet.process()	Auth Service	-0.011183099752145118	0.00313922818286667	0.003940311945145162	-0.0036749285799853883
17	StandardHostValve.invoke(Request request, Response response)	Auth Service	0	0	0	0
18	FrameworkServlet.doPost(HttpServlet request, HttpServletResponse response)	Auth Service	-0.0027959774380362794	0.0007848057045741668	0.0009850329862862905	-0.0009187321449963471
19	UserController.createUser(User user)	Auth Service	-0.0055915946072559	0.0015696114091483335	0.00197006597257281	-0.00183746289992642
20	UserServiceImpl.createUser(User user)	Auth Service	-0.0027959774380362794	0.0007848057045741668	0.0009850329862862905	-0.0009187321449963471
21	AccountServiceImpl.invoke(Request request, Response response)	Account Service	-0.3522931571932713	0.09888518776364495	0.12411413427207259	-0.11576025026949393

异常状态评估与故障原因定位

1 基于大数据的运行状态监测

- ✓ 微服务集群管理，可视化配置监测度量。
- ✓ 基于大数据平台实时收集微服务系统资源与性能并高效分析。

2 执行轨迹监测及构建

- ✓ 基于动态插桩注入监测代码，细粒度追踪请求处理的执行轨迹。
- ✓ 基于在线增量式聚类挖掘执行轨迹模式以构建基准状态集合。

3 自动化的故障诊断

- ✓ 基于树编辑距离动态评估执行轨迹异常程度。
- ✓ 基于路径搜索自动定位故障服务、组件和方法。
- ✓ 基于机器学习分析引发性能故障的敏感调用以定位性能瓶颈。

4 友好的可视化界面

- ✓ 实时查看系统资源与性能，以多种图表展现历史数据变化趋势。
- ✓ 以调用树形式展现执行轨迹，提供图表形象展现故障诊断结果。

系统架构

● 基础设施

利用HBase、OpenTSDB、Spark等大数据技术以满足海量数据的监测、存储与分析。

● 系统监测及故障诊断

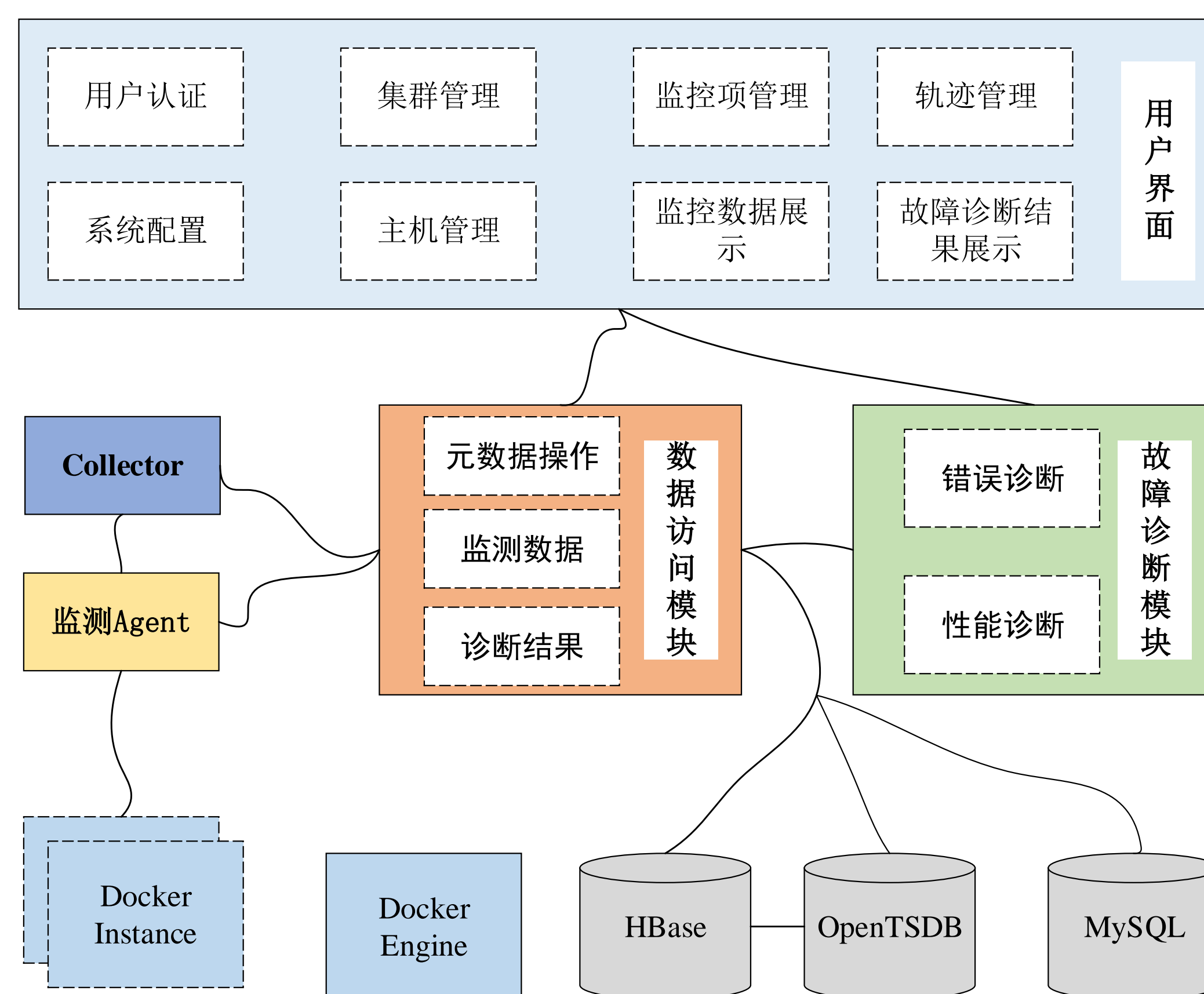
利用ASM基于动态插桩细粒度监测执行信息；
利用Spark Mlib基于机器学习实现智能故障诊断。

● 可视化展示

利用AngularJS、Echarts等前端框架提供多视角监测数据与分析结果的展现形式。

● 实例研究

基于典型微服务架构验证应用PiggyMetrics进行验证评估。



PiggyMetrics

