

CA Application Performance Management

for SOA 实施指南
版本 9.5



本文档包括内嵌帮助系统和以电子形式分发的材料（以下简称“文档”），其仅供参考，CA随时可对其进行更改或撤销。

未经 CA 事先书面同意，不得擅自复制、转让、翻印、透露、修改或转录本文档的全部或部分内容。本文档属于 CA 的机密和专有信息，不得擅自透露，或除以下协议中所允许的用途，不得用于其他任何用途：(i) 您与 CA 之间关于使用与本文档相关的 CA 软件的单独协议；或者 (ii) 您与 CA 之间单独的保密协议。

尽管有上述规定，但如果您为本文档中所指的软件产品的授权用户，则您可打印或提供合理数量的本文档副本，供您及您的雇员内部用于与该软件相关的用途，前提是所有 CA 版权声明和标识必须附在每一份副本上。

打印或提供本文档副本的权利仅限于此类软件所适用的许可协议的有效期内。如果该许可因任何原因而终止，您应负责向 CA 书面证明已将本文档的所有副本和部分副本已退还给 CA 或被销毁。

在所适用的法律允许的范围内，CA 按照“现状”提供本文档，不附带任何保证，包括但不限于商品适销性、适用于特定目的或不侵权的默示保证。CA 在任何情况下对您或其他第三方由于使用本文档所造成的直接或间接的损失或损害都不负任何责任，包括但不限于利润损失、投资受损、业务中断、信誉损失或数据丢失，即使 CA 已经被提前明确告知这种损失或损害的可能性。

本文档中涉及的任何软件产品的使用均应遵照有关许可协议的规定且根据本声明中的条款不得以任何方式修改此许可协议。

本文档由 CA 制作。

仅提供“有限权利”。美国政府使用、复制或透露本系统受 FAR Sections 12.212、52.227-14 和 52.227-19(c)(1) - (2) 以及 DFARS Section 252.227-7014(b)(3) 的相关条款或其后续条款的限制。

版权所有 © 2013 CA。保留所有权利。此处涉及的所有商标、商品名称、服务标识和徽标均归其各自公司所有。

CA Technologies 产品引用

本文档涉及以下 CA Technologies 产品和功能：

- CA Application Performance Management (CA APM)
- CA Application Performance Management ChangeDetector (CA APM ChangeDetector)
- CA Application Performance Management ErrorDetector (CA APM ErrorDetector)
- CA Application Performance Management for CA Database Performance (CA APM for CA Database Performance)
- CA Application Performance Management for CA SiteMinder® (CA APM for CA SiteMinder®)
- CA Application Performance Management for CA SiteMinder® Application Server Agents (CA APM for CA SiteMinder® ASA)
- CA Application Performance Management for IBM CICS Transaction Gateway (CA APM for IBM CICS Transaction Gateway)
- CA Application Performance Management for IBM WebSphere Application Server (CA APM for IBM WebSphere Application Server)
- CA Application Performance Management for IBM WebSphere Distributed Environments (CA APM for IBM WebSphere Distributed Environments)
- CA Application Performance Management for IBM WebSphere MQ (CA APM for IBM WebSphere MQ)
- CA Application Performance Management for IBM WebSphere Portal (CA APM for IBM WebSphere Portal)
- CA Application Performance Management for IBM WebSphere Process Server (CA APM for IBM WebSphere Process Server)
- CA Application Performance Management for IBM z/OS® (CA APM for IBM z/OS®)
- CA Application Performance Management for Microsoft SharePoint (CA APM for Microsoft SharePoint)
- CA Application Performance Management for Oracle Databases (CA APM for Oracle Databases)
- CA Application Performance Management for Oracle Service Bus (CA APM for Oracle Service Bus)
- CA Application Performance Management for Oracle WebLogic Portal (CA APM for Oracle WebLogic Portal)

- CA Application Performance Management for Oracle WebLogic Server (CA APM for Oracle WebLogic Server)
- CA Application Performance Management for SOA (CA APM for SOA)
- CA Application Performance Management for TIBCO BusinessWorks (CA APM for TIBCO BusinessWorks)
- CA Application Performance Management for TIBCO Enterprise Message Service (CA APM for TIBCO Enterprise Message Service)
- CA Application Performance Management for Web Servers (CA APM for Web Servers)
- CA Application Performance Management for webMethods Broker (CA APM for webMethods Broker)
- CA Application Performance Management for webMethods Integration Server (CA APM for webMethods Integration Server)
- CA Application Performance Management Integration for CA CMDB (CA APM Integration for CA CMDB)
- CA Application Performance Management Integration for CA NSM (CA APM Integration for CA NSM)
- CA Application Performance Management LeakHunter (CA APM LeakHunter)
- CA Application Performance Management Transaction Generator (CA APM TG)
- CA Cross-Enterprise Application Performance Management
- CA Customer Experience Manager (CA CEM)
- CA Embedded Entitlements Manager (CA EEM)
- CA eHealth® Performance Manager (CA eHealth)
- CA Insight™ Database Performance Monitor for DB2 for z/OS®
- CA Introscope®
- CA SiteMinder®
- CA Spectrum® Infrastructure Manager (CA Spectrum)
- CA SYSVIEW® Performance Management (CA SYSVIEW)

联系技术支持

要获取在线技术帮助以及办公地址、主要服务时间和电话号码的完整列表，请联系技术支持：<http://www.ca.com/worldwide>。

目录

第 1 章： 关于 CA Application Performance Management for SOA	15
什么是面向服务的体系结构?	15
SOA 基础架构的常见组件	16
如何使用 CA Introscope® 监控 SOA 性能.....	17
监控 Web 服务客户端和服务器	18
使用特定于 SOA 的显示板进行主动管理	19
查看依赖关系和使用度量标准进行分类和诊断	19
跟踪包含 SOA 组件的事务	19
跨多个平台和传输进行监控	20
了解用于监控 SOA 的体系结构	20
其他 SOA 平台支持	22
第 2 章： 安装和配置 CA APM for SOA	25
安装和配置.....	25
CA Introscope® 组件和版本	26
代理的基本系统要求.....	26
代理的 SOAP 和应用程序服务器要求	26
了解目录和文件命名约定.....	27
从先前版本升级之前进行备份	28
如何将 CA APM for SOA 添加到代理中.....	29
在独立代理安装程序中选择 CA APM for SOA.....	29
使用静默模式为代理启用 CA APM for SOA.....	30
将 CA APM for SOA 手工添加到代理.....	31
在安装后配置代理属性.....	33
如何启用 SOA 平台扩展	33
在企业管理器上启用扩展	35
验证 CA APM for SOA 部署	38
删除 CA APM for SOA	38
第 3 章： 监控面向服务的体系结构	41
在 CA Introscope® 中查看特定于 SOA 的信息	41
关于客户端和服务器 Web 服务和操作	42
关于服务命名空间和操作名称	42
关于无法识别的服务和操作	43
关于虚拟代理	43
使用 SOA 性能显示板	44
SOA 性能“概览”显示板	47

SOA 性能“客户端运行状况”显示板	47
SOA 性能“服务器运行状况”显示板	48
SOA 性能“最关键操作”显示板	48
SOA 性能“最依赖操作”显示板	49
SOA 性能“最忙操作”显示板	49
使用调查器查看 SOA 性能度量标准	49
可用度量标准.....	50
查看“概览”选项卡上的汇总度量标准.....	51
查看“依赖关系”选项卡上的依赖关系度量标准	52
查看“偏差”选项卡上的偏差度量标准	56
在“最关键”选项卡上查看关键操作度量标准	57
在“最依赖”选项卡上查看依赖操作	57
在“错误”选项卡上查看 SOAP 故障和错误度量标准	57
在调查器中查看 Boundary Blame	58
查看默认 SOA 特定度量标准组	59
查看默认 CA APM for SOA 报警	59

第 4 章： 使用 SOA 依赖关系地图 63

了解 SOA 依赖关系地图的用法	63
了解 SOA 环境的挑战	63
了解 SOA 依赖关系地图所提供的内容	64
了解依赖关系和 SOA 术语	64
了解使用 SOA 依赖关系地图可执行的任务	65
SOA 依赖关系视图获取数据的方式	65
关于 SOA 依赖关系视图的永久性数据	66
使用新服务和操作更新依赖关系地图	67
过时地图节点的老化和删除	68
如果依赖关系视图显示不完整数据应如何处理	68
关于逻辑等效规则	69
了解 SOA 依赖关系地图中的上下文	70
了解内容类型如何影响查看的内容	70
了解调查器节点如何影响所示内容	71
显示 SOA 依赖关系视图	71
关于调查器树节点和地图节点	72
关于独立地图节点和依赖地图节点	72
关于操作的地图节点	72
选择物理或逻辑视图	73
选择内容类型	74
为依存关系映射节点设置主要度量标准	74
为依赖关系地图节点选择工具提示度量标准	75
更改显示的依赖关系级别	76
检查地图节点的其他依赖关系	76

隐藏地图节点的依赖关系.....	77
检查图中所有项目的其他依赖关系.....	77
隐藏图中所有项目的依赖关系.....	78
在 SOA 依赖关系地图内导航	78
平移、缩放和适合 SOA 依赖关系地图	78
从地图节点跳转到关联的调查器树节点	79
显示服务的所有操作.....	80
显示代理的所有服务.....	80
保存 SOA 依赖关系地图图像	80
SOA 依赖关系视图共享	81
将“SOA 依赖关系地图”另存为快照	81
从视图节点启动事务跟踪	83
群集的 SOA 依赖关系视图	83
第 5 章： 在 SOA 环境中使用事务跟踪	85
关于跨进程事务跟踪.....	85
了解如何链接事务段.....	86
了解跨进程事务跟踪的上下文.....	86
使用事务跟踪解决问题.....	87
启动并查看事务跟踪.....	87
使用摘要视图.....	88
使用跟踪视图.....	89
使用树视图.....	89
使用顺序视图.....	90
为事务跟踪设置筛选.....	93
添加并保存筛选.....	94
删除筛选.....	95
使用复杂筛选.....	95
查询事件数据库以查找 SOA 服务	95
查看错误事件信息.....	96
查看相关联的事件信息.....	96
第 6 章： 配置特定于 SOA 的属性	99
配置 Web 服务的显示名称	99
配置关联跟踪.....	101
配置客户端和服务器的属性.....	102
禁用相关联的跟踪.....	102
配置 SOAP 处理程序的插入点.....	103
用于插入 SOAP 处理程序的客户端和服务器属性	103
更改 SOAP 处理程序的默认顺序	104
配置 SOA 依赖关系地图的限制	104

第 7 章：监控 Oracle Service Bus

107

关于 Oracle Service Bus (OSB)	107
如何启用对 Oracle Service Bus 的监控	109
启用用于监控 Oracle Service Bus 的代理	110
启用 Oracle Service Bus 企业管理器扩展	112
使用显示板监控 Oracle Service Bus	113
了解并查看 Oracle Service Bus 的度量标准	116
业务服务的度量标准	117
管道的度量标准	118
代理服务的度量标准	118
传输的度量标准	119
UDDI 的度量标准	120
XQuery 的度量标准	120
查看默认的 Oracle Service Bus 度量标准组	120
查看默认的 Oracle Service Bus 报警	121
查看 Oracle Service Bus 依赖关系	122
跟踪 Oracle Service Bus 的事务	123
了解跨进程事务跟踪的值	123
启动并查看示例事务跟踪	123

第 8 章：监控 TIBCO BusinessWorks

125

关于 TIBCO BusinessWorks	125
如何启用对 TIBCO BusinessWorks 的监控	127
启用代理以监控 TIBCO BusinessWorks	127
配置 TIBCO BusinessWorks 以使用代理	130
配置自动代理命名	131
启用企业管理器扩展	132
使用显示板监控 TIBCO BusinessWorks	133
了解并查看 TIBCO BusinessWorks 的度量标准	135
活动的度量标准	136
组操作的度量标准	137
Hawk 的度量标准	137
作业和作业池的度量标准	144
流程的度量标准	146
传输的度量标准	148
Web 服务的度量标准	150
查看默认的 TIBCO BusinessWorks 度量标准组	151
查看默认的 TIBCO BusinessWorks 报警	152
查看 TIBCO BusinessWorks 依赖关系	153
跟踪 TIBCO BusinessWorks 的事务	154
如何在 Tibco 中使用事务跟踪数据	155
了解跨进程事务跟踪的值	155

启动并查看示例事务跟踪.....	156
关于 BusinessWorks 的前端和后端.....	157
了解对生成的流程的限制.....	158
禁用多个后端与一个前端的关联.....	158
添加自定义 TIBCO BusinessWorks 后端.....	158
自定义度量标准老化和删除.....	159
配置 TIBCO BusinessWorks 的关联跟踪.....	160

第 9 章：监控 TIBCO Enterprise Message Service

161

关于 TIBCO Enterprise Message Service	161
如何安装 SOA Extension for TIBCO EMS	162
运行独立代理安装程序	162
使用响应文件进行静默安装.....	163
手工提取安装存档.....	164
准备 TIBCO EMS 服务器以进行监控	164
配置 TIBCO EMSMonitor 代理.....	165
配置基本连接属性.....	166
配置服务器的轮询时间间隔.....	167
配置筛选以进行选择性监控.....	168
为队列和主题配置筛选.....	168
定义要包含的高级组件.....	169
定义高级组件的正则表达式筛选.....	169
EMS 名称中特殊字符的替换.....	170
定义监控级别.....	170
使用默认监控级别定义.....	171
修改默认监控级别定义.....	172
创建加密密码.....	174
使用 SSL 连接到 TIBCO EMS 服务器实例	175
启动 EMSMonitor 代理	176
使用启动脚本启动 EMSMonitor	177
以 Windows 服务形式运行 EMSMonitor 代理.....	177
启用企业管理器扩展	178
使用显示板监控 TIBCO EMS	179
了解并查看 TIBCO EMS 度量标准	180
上次检查的度量标准.....	183
队列的度量标准.....	184
服务器的度量标准.....	188
主题的度量标准.....	196
路由的度量标准.....	199
通道的度量标准.....	201
网桥的度量标准.....	202
查看默认的 TIBCO EMS 度量标准组	202

查看默认的 TIBCO EMS 报警	203
代理配置属性汇总	204

第 10 章：监控 webMethods Broker 211

关于 webMethods Broker	211
如何安装 SOA Extension for webMethods Broker	212
验证必要条件	213
运行独立代理安装程序	213
准备 WebMethods Broker 以进行监控	214
为 webMethods Broker 配置代理	215
启用 webMethods Broker 的企业管理器扩展	220
使用显示板监控 webMethods Broker	221
了解并查看 Broker 的度量标准	223
Broker 的度量标准	225
客户端组的度量标准	226
客户端的度量标准	227
文档类型的度量标准	228
重试排队的度量标准	229
区域统计信息的度量标准	230
跟踪队列的度量标准	231
使用率的度量标准	231
查看默认的 Broker 度量标准组	232
查看默认 Broker 报警	232

第 11 章：监控 webMethods Integration Server 235

关于 webMethods Integration Server	235
如何启用对 webMethods Integration Server 的监控	238
手工启用代理以监控 webMethods Integration Server	239
关于 webMethods Integration Server 的指令文件	240
启用企业管理器扩展	241
使用显示板监控 webMethods	241
筛选监控和显示的服务	246
关于默认配置文件	246
使用正则表达式包含和排除服务	247
为配置文件指定其他位置	247
查看 webMethods 的度量标准并进行导航	248
适配器的度量标准	249
授权的度量标准	251
业务流程的度量标准	251
流服务的度量标准	254
Java 服务的度量标准	254
JDBC 池的度量标准	255

线程池的度量标准.....	255
Trading Networks 的度量标准.....	256
触发器的度量标准.....	259
Web 服务的度量标准.....	259
XSLT 服务的度量标准.....	261
查看默认的 webMethods 度量标准组.....	261
查看默认 webMethods 报警.....	262
查看 webMethods 依赖关系.....	262
webMethods 的跟踪事务	263
关于配置跨进程事务跟踪.....	264
了解跨进程事务跟踪的值.....	264
启动并查看示例事务跟踪.....	264
对 webMethods 流程使用顺序视图.....	265
第 12 章： 监控 WebSphere Process Server 和 WESB	267
关于 WebSphere Process Server 和 WESB	267
监控 WebSphere Process Server 组件	268
监控 WebSphere Enterprise Service Bus（独立）	271
如何启用对 WebSphere Process Server 或 WESB 的监控.....	271
启用代理以监控 WPS 或 WESB.....	272
启用 WPS 或 WESB 的企业管理器扩展	276
使用显示板监控 WPS 或 WESB	278
关于 WebSphere Process Server 显示板	279
关于 WESB 显示板	280
查看 WebSphere Process Server 或 WESB 显示板	281
查看 WPS/WESB 的度量标准并进行导航.....	282
业务对象映射的度量标准.....	283
业务流程的度量标准.....	283
业务规则的度量标准.....	283
业务状态计算机的度量标准.....	283
数据绑定的度量标准.....	284
人工任务的度量标准.....	284
接口映射的度量标准.....	284
J2CA 适配器的度量标准	285
Java 组件的度量标准	285
中介流和中介元素的度量标准.....	285
关联关系的度量标准.....	289
选择器的度量标准.....	289
服务集成总线通信的度量标准.....	289
WebSphere Process Server 错误的度量标准	289
查看 WPS 和 WESB 默认度量标准组	289
查看 WPS 和 WESB 默认报警	290

查看 WPS 或 WESB 依赖关系	291
跟踪 WPS 或 WESB 的事务	292
了解跨进程事务跟踪的值.....	292
启动并查看示例事务跟踪.....	292
为 WebSphere 流程使用顺序视图.....	293
配置跟踪以包括业务流程活动/业务状态机活动.....	294

附录 A：特定于 SOA 的代理配置属性 295

了解代理属性的位置.....	295
配置代理属性.....	295
关于特定于 SOA 的代理属性	296
agent.httpheaderinsertion.enabled.....	297
agent.httpheaderread.enabled.....	298
agent.soapheaderinsertion.enabled.....	299
agent.soapheaderread.enabled.....	300
agent.soa.metricNameFormatting	300
agent.transactiontrace.boundaryTracing.cacheFlushFrequency	301
agent.transactiontrace.boundaryTracing.enable.....	302
soa.client.prependhandler	302
soa.server.appendhandler	303

附录 B：特定于 SOA 的企业管理器配置属性 305

配置企业管理器属性.....	305
关于特定于 SOA 的企业管理器属性	306
soa.dependencymap.aging.refresh.interval	308
soa.dependencymap.aging.expire.interval	309
soa.dependencymap.heuristics.clientside.enable	309
soa.dependencymap.heuristics.namematch.enable	310
soa.dependencymap.heuristics.serverside.enable	311
soa.dependencymap.log.suppress.....	312
soa.dependencymap.max.edge.ratio.....	313
soa.dependencymap.max.vertices.....	314
soa.deviation.enable	315
soa.deviation.art.enable	316
soa.deviation.dependencymetric.enable.....	316
soa.deviation.count.per.metric.....	317
soa.deviation.dependency.refreshrate.....	318
soa.deviation.errors.enable	319
soa.deviation.max.metric.count	319
soa.deviation.mean.days	320
soa.deviation.metric.expressionlist	320
soa.deviation.metric.calledbackends.....	321
soa.deviation.usage.enable	322

soa.dashboard.typeviewer.average.enable	323
soa.dashboard.typeviewer.errors.enable	324
soa.dashboard.typeviewer.response.enable	325
soa.dashboard.typeviewer.mostcritical.enable	326
soa.dashboard.typeviewer.mostcritical.count	327
soa.dashboard.typeviewer.mostdependent.enable	328
soa.dashboard.typeviewer.mostdependent.count	329

附录 C：特定于 SOA 的 Workstation 配置属性

331

配置 Workstation 属性	332
关于特定于 SOA 的 Workstation 属性	333
soa.dependencymap.ui.view.nodecount	334
com.wily.introscope.soa.dependencymap.ui.view.edgecount	335
soa.dashboard.typeviewer.average.enable	336
soa.dashboard.typeviewer.errors.enable	337
soa.dashboard.typeviewer.response.enable	338
soa.dashboard.typeviewer.mostcritical.enable	339
soa.dashboard.typeviewer.mostcritical.count	340
soa.dashboard.typeviewer.mostdependent.enable	341
soa.dashboard.typeviewer.mostdependent.count	342
workstation.soa.dependencymap.fetchmetrics	343
workstation.traceview.crossprocess.duration.full	344
workstation.traceview.crossprocess.duration.net	345

附录 D：特定于 SOA 的 WebView 配置属性

347

配置 WebView 属性	347
关于特定于 SOA 的 WebView 属性	348
soa.dependencymap.ui.view.nodecount	348
com.wily.introscope.soa.dependencymap.ui.view.edgecount	349

第 1 章：关于 CA Application Performance Management for SOA

面向服务的体系结构 (SOA) 是一种应用程序平台，组织可以通过该平台共享和重用松散耦合的服务，以完成业务目标。

使用面向服务的体系结构来部署应用程序有多种优点，但在监控 SOA 环境方面同样面临特殊的挑战。本部分重点介绍了可以使用 CA Application Performance Management for SOA (CA APM for SOA) 以应对这些挑战的主要方式。

此部分包含以下主题：

- [什么是面向服务的体系结构？ \(p. 15\)](#)
- [SOA 基础架构的常见组件 \(p. 16\)](#)
- [如何使用 CA Introscope® 监控 SOA 性能 \(p. 17\)](#)
- [了解用于监控 SOA 的体系结构 \(p. 20\)](#)
- [其他 SOA 平台支持 \(p. 22\)](#)

什么是面向服务的体系结构？

面向服务的体系结构 (SOA) 是一种应用程序平台，它依赖于标准化通信协议，通过实现松散耦合的服务来完成业务目标。

使用 SOA 的优点包括提高业务灵活性和弹性、改进客户服务和效率，以及降低开发成本。但是，与管理传统的客户端-服务器环境相比，监控和管理复杂的 SOA 环境要困难得多。

在传统的客户端-服务器环境中，客户端和数量有限的服务器之间可以直接进行通信。发生问题时，找到故障的原因通常很简单，因为任何单个业务事务只涉及少数几个系统。您可以直接调查事务所直接涉及的特定系统来隔离问题的根源。

当 Web 应用程序服务器充当中心点来跨多个客户端-服务器系统分配对应用程序的访问时，确定问题的根源变得更加困难。性能下降、错误或操作故障几乎可由参与 Web 服务器所连接基础架构的任何组件或计算机造成。

面向服务的体系结构引入了附加的复杂层，用于监控应用程序的性能和可用性。通过 SOA，松散耦合的服务依赖标准化通信来集成和扩展不同平台上运行的应用程序。由于这些服务将业务逻辑与下层的操作系统或平台分离开来，因此在应对市场变化或产品动态时，组织可以更加灵活并进行快速响应。可以将单个服务设计为处理复杂或多步业务流程的特定部分，这样便在异构环境创建了依赖关系链。

使用面向服务的体系结构时，由于可以将服务作为独立组件进行重用、修改或替换，因此组织可以以更具成本效益的方式更快速地开发和部署应用程序。但是，这种针对应用程序体系结构的高效模块化方法也为应用程序管理提出了其自己的挑战。

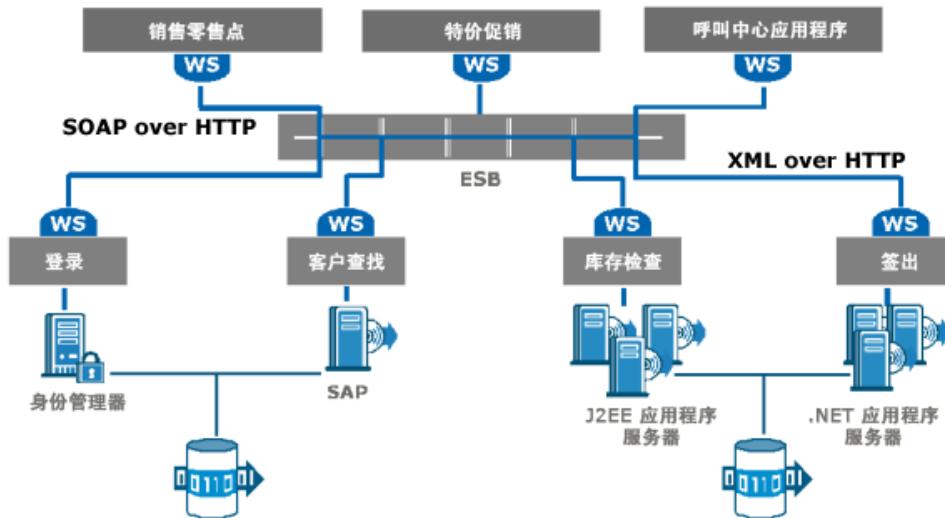
SOA 基础架构的常见组件

虽然使用 SOA 简化了业务流程的实施和集成，但是下层的 SOA 基础架构通常依赖于多个组件的复杂交互。例如，完成简单的业务事务通常涉及在不同组件上运行多个单独的服务，这些组件需要使用不同的协议彼此交换消息。监控事务的关键组件需要了解以下信息：服务如何通信，如何将请求和响应从一个服务路由到另一个服务，哪些服务依赖于其他服务，以及出现严重瓶颈的位置。

虽然具体的监控要求取决于 SOA 实施，但典型的 SOA 基础架构包括：

- 多个服务，这些服务使用标准接口来使客户端组件与服务器端组件互相交换信息
- 消息处理系统，它使请求和响应能够从一个组件路由、转换并传送到其他组件
- 适配器，使得外部系统或旧系统能够连接并使用部署的服务
- 注册表，它记录和发布有关可用服务的信息

下图说明了一种简化的 SOA 基础架构，它使用简单对象访问协议 (SOAP) 和可扩展标记语言 (XML)，使得 Web 服务 (WS) 可以通过企业服务总线 (ESB)（控制消息发送到适当目的地）相互通信。



因为 SOA 基础架构依赖于可重用和适应性组件的模块化传送，所以这些环境中的事务通常涉及更多的组件。随着事务涉及的组件数目的增加，跟踪事务流变得更加困难。并且，当更多消息从某个流程传递到其他流程，或从一个平台传递到其他平台（通常通过不同的传输协议）时，会存在更多潜在的故障点。此外，一旦组织决定在 SOA 环境中部署应用程序，他们往往会把越来越多的关键任务服务移到该环境中，使得对该基础架构的监控对于业务的正常运行变得愈发重要。

CA APM for SOA 通过提供 SOA 基础架构运行状况和参与 SOA 事务的组件的实时性能的可见性，解决了这些独特的挑战。

如何使用 CA Introscope® 监控 SOA 性能

CA APM for SOA 为 SOA 环境提供了应用程序性能的实时视图和历史视图。- 它扩展了核心 CA Introscope® 功能，使您能够从用户请求服务的前端系统到用于执行请求的后端系统监控并跟踪 Web 服务事务。通过将 CA Introscope® 与 CA APM for SOA 结合使用，您可以主动监控 SOA 客户端和服务器性能、分类突发事件以及深入分析服务相关问题，而无需访问应用程序源代码或咨询 SOA 架构师。

通过提供下列各项，将 CA APM for SOA 添加到 CA Introscope® 可帮助您监控 SOA 环境：

- 显示 SOA 客户端和服务器运行状况及性能的显示板，以对环境进行主动管理
- 概述和详细度量标准，帮助您将问题隔离到 Web 服务、应用程序或后端
- 代理、服务和操作之间依赖关系的直观表示，以帮助您评估一个组件的问题可能对其他组件的影响
- 跨平台、传输协议和应用程序服务器的相关事务跟踪
- Web 服务组件产生的应用程序错误或 SOAP 错误的可见性和上下文

除了这些用于 SOA 客户端和服务器的监控和应用程序管理的默认功能之外，您还可以自定义 CA Introscope® 和 CA APM for SOA 以满足您的需求。例如，您可以：

- 通过验证 Web 服务调用的收到和完成来构建 Web 服务审核路径
- 创建 Web 服务的应用程序组，并了解组是否超过响应时间、事务量或错误阈值
- 设置虚拟代理以收集 Web 服务客户端和服务的聚合度量标准

监控 Web 服务客户端和服务器

大多数情况下，Web 服务事务包括请求 Web 服务的客户端及响应该请求的服务器。例如，可以将用户或程序用来请求股票报价的 Java 类视为 Web 服务客户端。提供价格以响应请求的股票报价 Web 服务则被视为 Web 服务服务器。

代理分别跟踪并显示客户端度量标准和服务器度量标准，从而使您可以分别监控和设置客户端性能和服务器端性能的阈值。将客户端与服务器端分开还允许您在适当的上下文中查看依赖关系。

大多数情况下，Web 服务的客户端和服务器在单独的 Java 虚拟机 (JVM) 或 Microsoft 公共语言运行时 (CLR) 环境中运行。因此，客户端操作度量标准和服务器端操作度量标准通常由两个或更多代理报告。如果您要查看聚合在一起的客户端和服务器度量标准，可以通过配置[虚拟代理 \(p. 43\)](#)来实现。

使用特定于 SOA 的显示板进行主动管理

通过使用 CA APM for SOA，您可以在生产负载下全天实时监控关键服务。特定于 SOA 的显示板汇总关键性能指标并提供默认阈值和报警指示器，以便在响应时间增加或出现异常活动（例如负荷增加或吞吐量减少）时，及早通知您。使用显示板和报警指示符可以快速并主动地确定潜在问题，然后详细分析问题以确定问题的性质和谁负责解决该问题。

通过使用 CA Introscope® 和 CA APM for SOA 显示板，您可以持续监控 Web 服务客户端和服务器性能，从而在潜在问题处于发展状态时轻松检测出该问题。在许多情况下，采取这种积极的方法有助于您在故障组件影响客户或最终用户前确定并修复该组件。

查看依赖关系和使用度量标准进行分类和诊断

当复杂的 SOA 环境中出现问题时，经常很难确定从何处开始寻找问题的根源。由于应用程序体系结构更为模块化，因此每个事务有更多的组件参与，存在更多潜在的故障点。

显示板和报警指示符可以帮助及早发现潜在问题。SOA 依赖关系视图可以帮助了解模块化组件之间如何相互关联以及这些关联关系如何影响性能。通过依赖关系，您可以分析潜在瓶颈或关键操作的位置，并比较彼此具有依赖关系的服务的平均响应时间或其他度量标准。

高级视图概述总体应用程序性能，针对潜在问题进行报警，并说明组件之间的依赖关系。具有 CA APM for SOA 的 CA Introscope® 还提供高级摘要和详细的操作级度量标准，组织的架构师和开发人员可以使用这些摘要和度量标准查看、隔离和解决问题。有关使用度量标准评估 Web 服务性能和操作的详细信息，请参阅[使用调查器查看 SOA 性能度量标准 \(p. 49\)](#)。

跟踪包含 SOA 组件的事务

在 SOA 环境中，使用多个协议在松散耦合的组件之间传递消息很常见，这使得跟踪事务流变得更加困难。事务跟踪提供有关涉及 Web 服务的单个事务、Web 服务故障的数量和性质以及组件交互的详细信息。例如，事务跟踪确定每个事务段的发生位置以及每个段的实时表现情况。然后应用程序专家可以使用此信息来确定错误或性能下降的根本原因。

当突发事件发生时（例如，特定 Web 服务的响应时间变慢），在违反服务级别协议 (SLA) 或最终用户受影响之前，会发出报警以通知相应的利益干系人。然后应用程序支持人员可以监控单个事务以便以直观方式确定事务中花费时间最多的部分以及用来完成每个捕获的事务的调用顺序（同步或异步），从而收集有关问题性质的更多详细信息。

跨多个平台和传输进行监控

通过使用具有 CA APM for SOA 的 CA Introscope®，您可以监控并管理整个基础架构，包括跨多个操作系统、应用程序平台或传输协议的事务。在典型的 SOA 环境中，在事务中使用 SOAP、XML、HTTP、HTTPS 和 JMS 传输协议的组合并包含多个应用程序服务器环境中的处理是很常见的。

通过使用 CA APM for SOA，您可以跨支持的 J2EE 或 .NET 服务器的任何组合跟踪事务。例如，您可以跟踪以下事务的完整路径：该事务包括在 WebLogic 上运行的服务，该服务向在 SAP NetWeaver 或 .NET 应用程序服务器上运行的服务发送请求。根据启用的其他 SOA 扩展，您还可以跨 Oracle Service Bus、IBM WebSphere Process Server、IBM WebSphere Enterprise Service Bus、TIBCO BusinessWorks、webMethods Integration Server 或 IBM WebSphere MQ 组件的任何组合监控事务。

另外，在 SOA 环境中同时包含通过适配器连接到 SOA 基础架构的复合应用程序以及使用标准化协议（例如 SOAP 和 XML）的新应用程序也是很常见的。由于环境中既包含旧应用程序又包含新应用程序，因此，您可能需要监控使用各种传输和负载类型的应用程序。例如，通过使用 CA APM for SOA，您可以监控使用 SOAP over HTTP、SOAP over JMS、XML over HTTP 或其他消息传输和协议的应用程序。

了解用于监控 SOA 的体系结构

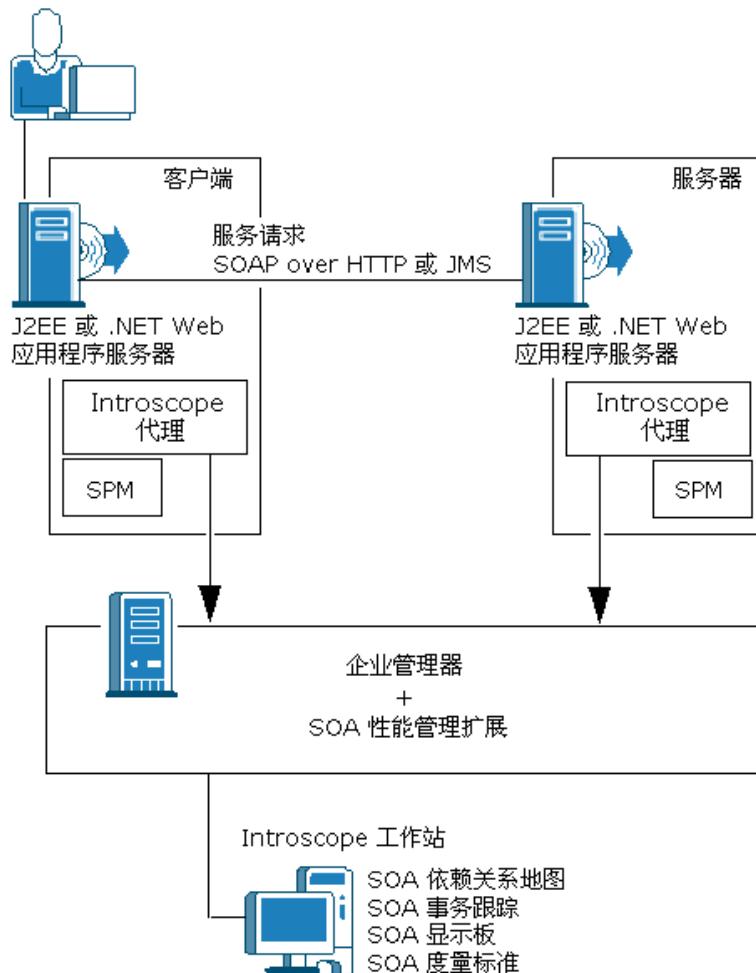
CA APM for SOA 为 CA Introscope® 提供了下列轻量级组件：

- 针对 Java 或 .NET 代理的特定于 SOA 的扩展
- 针对企业管理器的特定于 SOA 的管理模块

代理扩展使代理能够在支持的应用程序服务器上监控基于 Java 和 .NET 的 Web 服务，并向企业管理器发送有关这些 Web 服务的性能的信息。

借助于针对企业管理器的特定于 SOA 的扩展，由代理收集的特定于 SOA 的信息可以使用 CA Introscope® Workstation 显示在特定于 SOA 的依赖关系视图、显示板、调查器节点和选项卡中。

下图说明了包含两个应用程序服务器的代理和 CA APM 的基本体系结构。这些应用程序服务器发送和接收 Web 服务请求，并将 Web 服务数据报告给启用了 CA APM for SOA 的单个企业管理器。

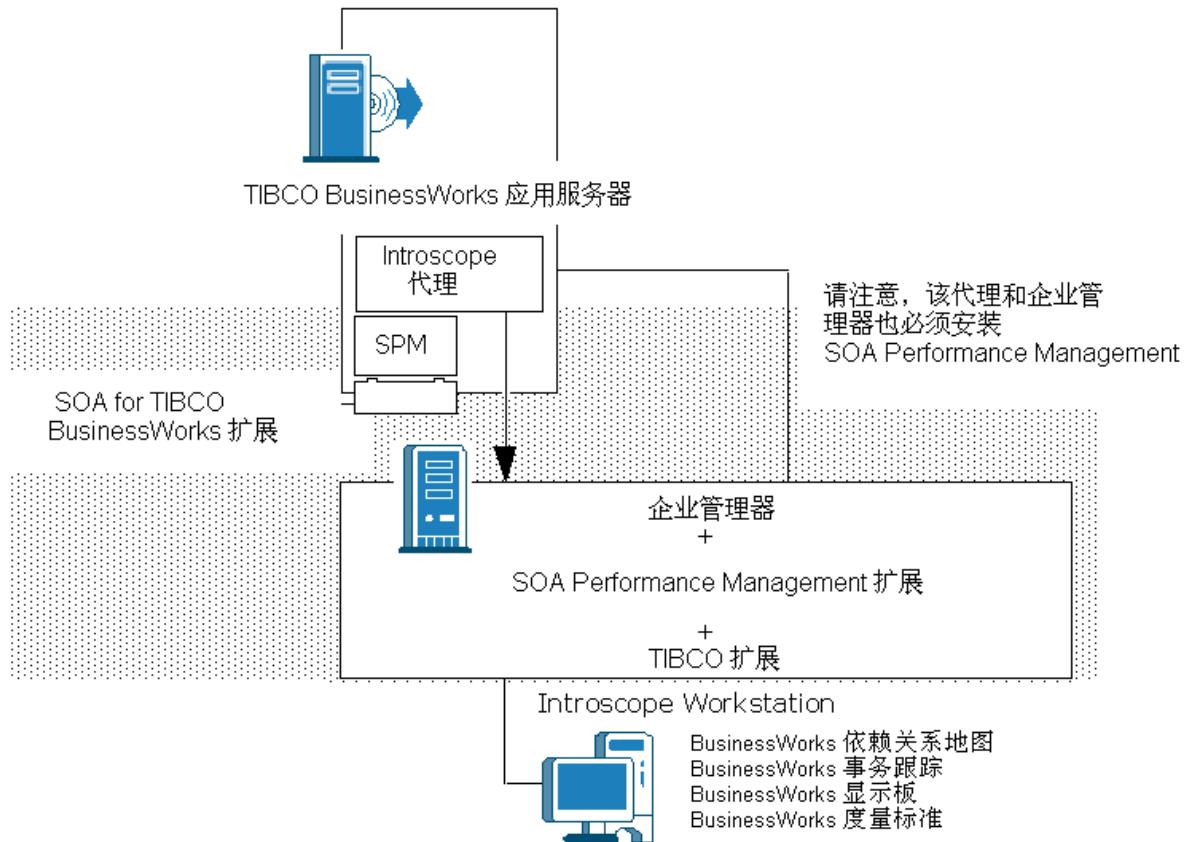


其他 SOA 平台支持

除由 CA APM for SOA 提供的功能之外，还存在其他几个您可以监控的 SOA 平台。根据环境的不同，您还可以监控下列其他 SOA 平台：

- Oracle Service Bus (OSB) 是为 Oracle WebLogic 处理消息分发和转换的企业服务总线。
- TIBCO BusinessWorks 是业务服务开发和处理引擎。
- 通过使用队列、主题和高级组件(例如网桥、多播通道和路由)，TIBCO Enterprise Message Service 使得在整个企业的多个系统之间可以执行同步和异步通信。
- webMethods Broker 提供异步处理和消息处理服务以发布和传送文档。
- webMethods Integration Server 提供基础架构组件，以便组织可以创建、管理和集成业务流程和 Web 服务。
- 具有 IBM WebSphere Enterprise Service Bus (WESB) 的 IBM WebSphere Process Server (WPS) 是一种集成平台，它包括通过 WebSphere Enterprise Service Bus 执行的消息处理。
- IBM WebSphere Enterprise Service Bus (WESB) 作为独立的产品。WebSphere Enterprise Service Bus 借助中介流和中介元素管理服务和客户端之间的消息流、数据转换和路由。

大部分 SOA 平台扩展都向代理和企业管理器中添加文件。例如，下图说明将 SOA Extension for TIBCO BusinessWorks 添加到应用程序服务器和企业管理器中：



对于某些 SOA 扩展，无法使用代理。例如，SOA Extension for TIBCO Enterprise Message Service 使用独立代理。有关任何 SOA 扩展的详细信息，请参阅介绍该扩展的相应章节。

第 2 章： 安装和配置 CA APM for SOA

此部分包含以下主题：

- [安装和配置 \(p. 25\)](#)
- [CA Introscope® 组件和版本 \(p. 26\)](#)
- [如何将 CA APM for SOA 添加到代理中 \(p. 29\)](#)
- [在企业管理器上启用扩展 \(p. 35\)](#)
- [验证 CA APM for SOA 部署 \(p. 38\)](#)
- [删除 CA APM for SOA \(p. 38\)](#)

安装和配置

可以使用以下方法在代理和企业管理器上安装和配置 CA APM for SOA：

是否已安装代理？	是否已启用 CA APM for SOA？	后续步骤
是	是	在企业管理器上启用扩展 (p. 35) 。
是	否	手工将 CA APM for SOA 添加到代理 (p. 31) 。
否	否	检查 CA APM for SOA 先决条件 (p. 26) 。 将 CA APM for SOA 添加到代理 (p. 29) 。 注意： 使用此信息与《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》中的信息来完成安装。

CA Introscope® 组件和版本

CA APM for SOA 为 CA Introscope® 提供扩展，用于监控面向服务的体系结构和 Web 服务。您可以在安装核心 CA Introscope® 组件的同时安装 CA APM for SOA，也可以在安装核心 CA Introscope® 组件后单独安装。无论与 CA Introscope® 一起安装 CA APM for SOA 还是单独安装 CA APM for SOA，都必须安装下列 CA Introscope® 组件和版本：

- Java 代理或 .NET 代理必须与 CA Introscope® 的版本相同。
- 企业管理器的版本必须与代理的版本相同或更高。

代理的基本系统要求

CA APM for SOA 向企业管理器和代理中添加文件。由于 CA APM for SOA 扩展代理，因此对处理、内存和可用磁盘空间的最低系统要求与代理的要求相同。

有关 CA APM 系统要求的具体信息，请参阅以下指南：

- 有关基本的内存和磁盘空间要求，请参阅《CA APM 安装和升级指南》。
- 有关调整处理负载及管理系统资源（如 CPU）的指导信息，请参阅《CA APM 规模调整和性能指南》。
- 有关 JVM 和 JRE 版本以及相关 JVM 内存的最低要求，请参阅《CA APM Java 代理实施指南》。

除基本代理要求外，启用了 CA APM for SOA 的代理还要求受支持的 SOAP 引擎在本地操作环境中可用。有关受支持的 SOAP 引擎的信息，请参阅《CA APM 兼容性指南》。您使用的 SOAP 引擎决定 SOAP 栈实现以及代理可支持的 Web 服务消息交换类型。

代理的 SOAP 和应用程序服务器要求

您使用的操作系统和应用程序服务器决定可用于处理 Web 服务消息的 SOAP 引擎和 SOAP 栈实现。例如，使用 Apache Axis 2.0 的计算机上的应用程序可以使用 JAX-RPC 或 JAX-WS 处理 SOAP 消息。

Java 代理的本地 SOAP 引擎支持

如果将 CA APM for SOA 添加到 Java 代理，代理将可以支持各种 SOAP 引擎和应用程序编程接口 (API)。根据应用程序服务器和版本的不同，代理可以支持遵守本地、JAX_RPC 或 JAX-WS 标准的 SOAP 栈实现。

注意：有关代理在当前版本中支持的应用程序服务器和 SOAP 堆栈实现的完整列表，请参阅《兼容性指南》中的“CA APM for SOA”部分。

.NET 代理的本地 SOAP 引擎支持

默认情况下，.NET 代理支持监控 ASP.NET Web 服务。如果向 .NET 代理中添加 CA APM for SOA，该代理会为您使用的 SOAP 通信服务提供标准的 Web 服务命名度量标准和其他依赖关系相关度量标准以及基于依赖关系的映射。要支持使用 .NET Framework 的 .NET 代理上 Web 服务的监控，可使用以下 Web 服务：

- 标准 ASP.NET Web 服务
- Windows Communication Foundation (WCF) Web 服务

除 .NET Framework 之外，要监控 ASP.NET Web 服务，请使用以下属性：

- 服务器端 Web 服务必须已应用 WebMethod 属性。
- 客户端 Web 服务必须已应用 SOAPDocumentMethod、HttpMethod 或 SoapRpcMethod 属性。

注意：有关 SOAP 支持的信息，请参阅《兼容性指南》中的“SOA 性能管理”部分。有关这些属性或如何将它们应用于方法或类的信息，请参阅 Microsoft Developer Network (MSDN) 库。

了解目录和文件命名约定

本指南对于文件名和目录路径使用以下约定：

位置	含义
<code><Agent_Home></code>	安装代理的顶级目录。此目录通常命名为 <code>wily</code> 。
<code><EM_Home></code>	安装企业管理器的顶级目录。

位置	含义
<version>	包含在文件名中或显示在用户界面中的版本特定标识符。 例如，以下文件名： <i>com.wily.introscope.soa.dependencymap_<version>.jar</i> 表示版本特定文件名，如： <i>com.wily.introscope.soa.dependencymap_v9.1.0.0.jar</i>
正斜杠 (/) 路径分隔符	使用的平台上目录名称中使用的路径分隔符。 在本指南的示例以及 UNIX 平台上使用的均是正斜杠 (/)，但您应该使用与您使用的平台相对应的分隔符。
美元符号 (\$) 环境变量	在您的平台上使用的环境变量符号。 在本指南的示例中，我们使用了 UNIX 平台的美元符号 (\$)，而您则应使用自己所用平台的相应字符。

从先前版本升级之前进行备份

如果要从先前版本的 Web Services Manager 或 CA APM for SOA 升级，在添加 CA APM for SOA 代理或企业管理器文件之前，应先备份当前环境并将现有文件保存在备份目录中。

如果从早期版本进行升级，请执行以下操作：

- 将现有的 <Agent_Home>/IntroscopeAgent.profile 文件复制到备份目录。
- 将现有的 <Agent_Home>/webservices.pbd 文件复制到备份目录。
- 将现有的 <EM_Home>/modules/SPM_ManagementModule.jar 文件复制到备份目录。

如果从早期版本进行升级且代理为 .NET 代理，请执行以下操作：

- 确保在安装 .NET 代理的新版本之前已卸载 .NET 代理的较旧版本。
- 确认要升级的 .NET 代理与 CA APM for SOA 文件处于相同的版本级别。版本信息不一致可能导致错误，并阻止对应用程序的监控。

如何将 CA APM for SOA 添加到代理中

以交互方式安装代理或使用响应文件以无提示方式安装代理时，可以自动将 CA APM for SOA 和与 SOA 相关的扩展添加到代理中。您也可以在安装代理后手工将 CA APM for SOA 和其他与 SOA 相关的扩展添加到代理中。

根据您选择的安装类型，执行相应的任务：

- [在独立代理安装程序中选择 CA APM for SOA \(p. 29\)](#)
- [在静默安装期间添加 CA APM for SOA \(p. 30\)](#)
- [将 CA APM for SOA 手工添加到代理 \(p. 31\)](#)

安装或更新代理以包含 CA APM for SOA 后，请参阅[安装后配置代理属性 \(p. 33\)](#)以获取有关配置应用环境的信息。如果要监控其他 SOA 平台（例如 Oracle Service Bus 或 TIBCO BusinessWorks），也可以通过交互方式、使用响应文件安装这些文件，或者在安装代理并选择 CA APM for SOA 之后进行手工安装。有关详细信息，请参阅[如何启用 SOA 平台扩展 \(p. 33\)](#)。

在独立代理安装程序中选择 CA APM for SOA

如果以交互方式安装代理，您可以在安装过程中选择添加 CA APM for SOA 和其他监控扩展。在独立代理安装程序中显示的具体选项取决于您选择的代理和应用程序服务器。例如：

- 如果要启用 CA APM for SOA，请选择“默认”、“JBoss”、“Tomcat”、“WebLogic”或“WebSphere”作为应用程序服务器。
- 如果要启用 CA APM for Oracle Service Bus，请选择“WebLogic”作为应用程序服务器。
- 如果要启用 CA APM for TIBCO BusinessWorks 或 CA APM for webMethods Integration Server，请选择“默认”作为应用程序服务器。
- 如果要启用 CA APM for WebSphere Process Server（带有 WESB）或 CA APM for WebSphere Enterprise Service Bus (WESB)（作为独立产品），请选择“WebSphere”作为应用程序服务器。

例如，如果在 Windows 上以交互方式运行独立代理安装程序并选择“默认”作为应用程序服务器，您可以选择 CA APM for SOA 和 CA APM for TIBCO BusinessWorks 或 CA APM for webMethods Integration Server 作为监控选项。

如果在独立代理安装程序中选择 CA APM for SOA 选项，代理的相关文件会自动复制到代理主目录的相应目录中，并且代理的配置文件会自动进行更新以包含用于监控 SOA 环境的相应 ProbeBuilder 指令文件。例如，会自动将 *appmap-soa.pbd* 和 *webservices.pbd* 添加到 *introscope.autoprobe.directivesFile* 属性。

如果使用独立代理安装程序安装 CA APM for SOA，请执行[配置 \(p. 33\)](#) Java 和 .NET 代理的后续步骤。

使用静默模式为代理启用 CA APM for SOA

如果以静默模式安装代理，可以使用响应文件中的设置。使用这些设置可以指定是否要为代理启用 CA APM for SOA。通过使用响应文件，您可以远程安装和配置代理属性而无需用户交互。借助该方法，可以使用脚本或其他软件交付选项实现过程自动化。

注意：有关以静默模式安装代理的详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

遵循这些步骤：

1. 打开 *SampleResponseFile.Agent.txt* 文件，该文件与独立代理安装程序位于同一目录中。
2. 编辑 *SampleResponseFile.Agent.txt* 文件，使之符合您的首选安装设置。
3. 将 *shouldEnableSPM* 属性设置为 *true* 以将 CA APM for SOA 添加到代理。例如：
`shouldEnableSPM=true`
如果将该属性设置为 *false*，SPM 特定的文件会自动复制到 *<Agent_Home>/examples* 目录中，但不会更新代理配置文件。
您也可以使用响应文件中的属性为特定 SOA 平台启用监控。如果要启用对其他 SOA 平台的监控，必须将 *shouldEnableSPM* 属性设置为 *true*。
4. 保存 *SampleResponseFile.Agent.txt* 文件。
5. 在命令行中输入相应的命令以调用安装程序。

如果使用响应文件以无提示模式安装 CA APM for SOA，接下来的步骤是[在安装之后配置代理属性 \(p. 33\)](#)。

详细信息：

[CA Introscope® 组件和版本 \(p. 26\)](#)

[如何启用 SOA 平台扩展 \(p. 33\)](#)

将 CA APM for SOA 手工添加到代理

如果您未在独立代理安装程序中或用于静默安装的响应文件中启用 CA APM for SOA，可以在安装后手工更新代理。手工更新代理以包含 CA APM for SOA 需要修改代理配置文件。根据您需要手工更新的代理类型，请参阅相应部分：

- [手工更新 Java 代理 \(p. 31\)](#)
- [手工更新 .NET 代理 \(p. 32\)](#)

手工更新 Java 代理以使用 CA APM for SOA

您可以手工更新 Java 代理以在安装之后使用 CA APM for SOA。将文件从 `<Agent_Home>/examples` 目录移动到 `<Agent_Home>/core/ext` 目录。然后编辑代理配置文件以包含 CA APM for SOA 的相应 ProbeBuilder 指令文件。

遵循这些步骤：

1. 导航到 `<Agent_Home>/examples/SOAPerformanceManagement/ext` 目录。
2. 将文件从该目录复制并粘贴到 `<Agent_Home>/core/ext` 目录。
3. 停止应用程序服务器。
4. 在文本编辑器中打开 `<代理主目录>/core/config/IntroscopeAgent.profile` 文件。
5. 将 `spm.pbl` 值添加到 `IntroscopeAgent.profile` 文件中的 `introscope.autoprobe.directivesFile` 属性中。

例如：

`introscope.autoprobe.directivesFile=websphere-full.pbl,hotdeploy,spm.pbl`

6. 如果需要更改默认设置，请在 `IntroscopeAgent.profile` 文件中修改其他任何代理属性。保存文件并退出文本编辑器。

7. 重新启动应用程序服务器。

应用程序服务器会重新启动代理并检测应用程序以启用 SOA 性能监控。

在手工配置 Java 代理配置文件之后，您将在[安装后配置代理属性 \(p. 33\)](#)。

手工更新 .NET 代理以使用 CA APM for SOA

您可以手工更新 .NET 代理以在安装之后使用 CA APM for SOA。将文件从 `<Agent_Home>/examples` 目录移动到 `<Agent_Home>/ext` 目录，并编辑默认 ProbeBuilder 列表以包含 CA APM for SOA 文件。在更新代理配置文件之后，您将在[安装后配置代理属性 \(p. 33\)](#)。

遵循这些步骤：

1. 导航到 `<Agent_Home>\examples\SOAPerformanceManagement` 文件夹。
2. 将文件从 `<Agent_Home>\examples\SOAPerformanceManagement\ext` 文件夹复制到 `<Agent_Home>\ext` 文件夹。
3. 将 `wily.WCFServicesAgent.ext.dll` 文件从 `<Agent_Home>\examples\SOAPerformanceManagement\wcf\ext` 文件夹复制到 `<Agent_Home>\ext` 文件夹。
4. 停止 IIS 应用程序服务器。
5. 在文本编辑器中打开 `<Agent_Home>\IntroscopeAgent.profile` 文件，并验证 `introscope.autoprobe.directivesFile` 属性设置。

例如：

```
introscope.autoprobe.directivesFile=default-full.pbl,hotdeploy
```

6. 保存更改并关闭 `IntroscopeAgent.profile` 文件。
7. 在文本编辑器中打开为 `introscope.autoprobe.directivesFile` 属性指定的 `default-full.pbl` 或 `default-typical.pbl`。
8. 保存并关闭 `default*.pbl` 文件。
9. 在完成配置之后重新启动 IIS 服务器。

将更新 .NET 代理以使用 CA APM for SOA。

在安装后配置代理属性

完成安装后，您可以根据需要配置其他代理属性并更新应用程序服务器。例如，您可以自定义代理配置文件来调整受监控的组件或关联信息的处理。根据您的应用程序环境，您可能还需要配置应用程序服务器中的设置或修改服务器启动脚本。

注意：如果您已安装 Java 代理，请参阅《CA APM Java 代理实施指南》以获取有关配置应用环境的详细信息。有关安装和配置 .NET 代理以及设置 .NET 环境属性的详细信息，请参阅《CA APM .NET 代理实施指南》。Java 代理或 .NET 代理与 CA APM for SOA 一起工作无需任何特定的其他更改。

如何启用 SOA 平台扩展

如果有一个启用了 CA APM for SOA 的代理，您可以将监控扩展到以下 SOA 平台的组件：

- AquaLogic Service Bus (ALSB) 和 Oracle Service Bus (OSB)
- IBM WebSphere Process Server (WPS) 和 WebSphere Enterprise Service Bus (WESB) 或作为独立产品的 WebSphere Enterprise Service Bus (WESB)
- TIBCO BusinessWorks 或 TIBCO ActiveMatrix BusinessWorks (BW)
- Software AG webMethods Integration Server (IS)

根据您的应用程序服务器和代理环境，您可以在以交互方式运行独立代理安装程序时，使用响应文件以无提示方式安装代理时，为这些 SOA 平台启用 CA APM，或者可以在安装代理后手工为这些 SOA 平台启用 CA APM。

除了依赖于代理和 CA APM for SOA 的与 SOA 相关的扩展之外，还有独立的扩展用于监控以下 SOA 平台：

- TIBCO Enterprise Message Service (EMS)
- webMethods Broker

这些独立扩展的安装和配置均与代理分开。

注意：有关安装和配置独立扩展的信息，请参阅相应部分。

以交互方式添加与 SOA 相关的扩展

如果以交互方式运行独立代理安装程序，则可以选择要在安装程序中启用的与 SOA 相关的扩展。可选择的特定监控选项取决于所选择的应用程序服务器。例如，如果选择“WebLogic”作为应用程序服务器，您可以选择 CA APM for SOA 和 CA APM for Oracle Service Bus。如果选择“默认”作为应用程序服务器，您可以选择 CA APM for SOA 和 CA APM for TIBCO BusinessWorks 或 CA APM for webMethods Integration Server。

基于在安装程序中所做的选择，相应的文件会自动复制到代理主目录的相应目录中，且相应的 ProbeBuilder 指令会自动添加到代理的配置文件中。

使用静默安装响应文件添加与 SOA 相关的扩展

如果在静默模式下安装代理，则可使用响应文件中的设置指定是否要为代理启用 SOA 扩展。

在静默模式下为代理启用 SOA 平台扩展

1. 打开 *SampleResponseFile.Agent.txt* 文件，该文件与独立代理安装程序位于同一目录中。
2. 编辑 *SampleResponseFile.Agent.txt* 文件，使之符合您的首选安装设置。
3. 将 *shouldEnableSPM* 属性设置为 *true* 以将 CA APM for SOA 添加到代理。例如：

```
shouldEnableSPM=true
```

如果要启用 CA APM for Oracle Service Bus (OSB)、CA APM for IBM WebSphere Process Server (WPS)、CA APM for WebSphere Enterprise Service Bus (WESB)、CA APM for TIBCO BusinessWorks (BW) 或 CA APM for webMethods Integration Server，必须将此属性设置为 *true*。

4. 将相应的 *shouldEnable** 属性设置为 *true* 可将 SOA 扩展添加到代理。例如：

- *shouldEnableSOAExtForOSB=true* 会启用 CA APM for Oracle Service Bus (OSB)
- *shouldEnableSOAExtForWPSandWESB=true* 会启用 CA APM for IBM WebSphere Process Server (WPS) (带有 WebSphere Enterprise Service Bus (WESB)) 或 CA APM for WebSphere Enterprise Service Bus (作为独立产品)
- *shouldEnableSOAExtForTibcoBW=true* 会启用 CA APM for TIBCO BusinessWorks
- *shouldEnableSOAExtForWebMethodsIS=true* 会启用 CA APM for webMethods Integration Server

如果将属性设置为 *false*，系统会将特定于扩展的文件复制到 *<Agent_Home>/examples* 目录中，但不会更新代理配置文件。有关在安装代理之后启用扩展的信息，请参阅[安装代理后手动添加 SOA 扩展 \(p. 35\)](#)。

5. 保存 *SampleResponseFile.Agent.txt* 文件。

6. 在命令行中输入相应的命令以调用安装程序。

安装代理后手工添加与 SOA 相关的扩展

如果没有在独立代理安装程序或响应文件中启用与 SOA 相关的扩展，您可以在完成安装后手工更新代理。手工更新代理通常包括下列步骤：

- 将文件从相应的 *<Agent_Home>/examples* 目录复制到 *<Agent_Home>/core/ext* 目录。
- 配置代理配置文件以包含平台特定代理扩展的相应 ProbeBuilder 指令 (.pbd 或 .pbl)。
- 配置服务器以调用代理或重新启动服务器。

但是，根据 SOA 平台，手工更新代理时可能需要执行其他平台特定步骤。

注意：有关平台特定步骤的信息，请参阅要启用 SOA 扩展的相应部分。

在企业管理器上启用扩展

安装企业管理器时，管理模块以及有关 CA APM for SOA 和与 SOA 相关的扩展的其他文件默认安装在 *<EM_Home>/examples* 目录中。例如，有关 CA APM for SOA 的文件默认安装在 *<EM_Home>/examples/SOAPerformanceManagement* 目录中。

要启用任何扩展，请将文件从扩展的 `<EM_Home>/examples` 子目录移到相应的 `<EM_Home>` 目录，并部署扩展的管理模块。

将 CA APM for SOA 添加到企业管理器

1. 停止企业管理器。
2. 导航到 `<EM_Home>/examples/SOAPerformanceManagement` 目录。
3. 将
`<EM_Home>/examples/SOAPerformanceManagement/config/modules` 目录的内容复制并粘贴到 `<EM_Home>/config/modules` 目录中。
`<EM_Home>/examples/SOAPerformanceManagement/config/modules` 目录包含 CA APM for SOA 管理模块 (`SPM_ManagementModule.jar`)。群集环境中有一个以上的企业管理器时，请仅将此文件复制到要用作 MOM 计算机的企业管理器上的 `<EM_Home>/config/modules` 目录中。
4. 将 `<EM_Home>/examples/SOAPerformanceManagement/ext` 目录的内容复制并粘贴到 `<EM_Home>/ext` 目录中。
`<EM_Home>/examples/SOAPerformanceManagement/ext/xmltv` 目录包含使 Workstation 可以显示 WSOA 概览、偏差和依赖关系度量标准的 CA APM for SOA 选项卡定义。
5. 将 `<EM_Home>/examples/SOAPerformanceManagement/product/enterprisemanager/plugins` 目录复制并粘贴到 `<EM_Home>/product/enterprisemanager/plugins` 目录中。
`<EM_Home>/examples/SOAPerformanceManagement/product/enterprisemanager/plugins` 目录包含支持事务跟踪、偏差度量标准和 SOA 依赖关系视图的文件。如果已经将企业管理器部署为群集，请将 `<企业管理器主目录>/examples/SOAPerformanceManagement/product/enterprisemanager/plugins` 中的文件复制到群集中所有企业管理器上的 `<企业管理器主目录>/product/enterprisemanager/plugins` 中。如果未在收集器企业管理器上找到这些文件，依赖关系视图将无法正确显示。
6. 将 `<EM_Home>/examples/SOAPerformanceManagement/ws-plugins` 目录复制并粘贴到 `<EM_Home>/ws-plugins` 目录中。
`<EM_Home>/examples/SOAPerformanceManagement/ws-plugins` 目录包含支持 SOA 依赖关系视图、依赖关系度量标准和事务跟踪的其他文件。
7. 重新启动企业管理器。

默认情况下，CA APM for SOA 包含控制某些操作（如依赖关系视图中关系刷新频率或报告哪些偏差度量标准）的企业管理器属性 (p. 295)。对于大多数组织，默认设置是合适的，您不需要更改这些设置；但是，您可以修改属性。

如果要对其他 SOA 平台启用监控，必须首先在企业管理器上启用 CA APM for SOA。

将平台特定 SOA 扩展添加到企业管理器中

1. 停止企业管理器。
2. 导航到相应的 `<EM_Home>/examples` 目录。例如，转到下列目录之一：

- `SOAExtensionForOSB`
 - `SOAExtensionForTibcoBW`
 - `SOAExtensionForTibcoEMS`
 - `SOAExtensionForWebMethodsBroker`
 - `SOAExtensionForWebMethodsIS`
 - `SOAExtensionForWPSandWESB`
3. 将 `<企业管理器主目录>/examples/<扩展>/config/modules` 目录的内容复制并粘贴到 `<企业管理器主目录>/config/modules` 目录中。
`<企业管理器主目录>/examples/<扩展>/config/modules` 目录包含 SOA 扩展的管理模块。在群集环境中有一个以上的企业管理器时，仅将管理模块复制到将用作 MOM 计算机的企业管理器上的 `modules` 目录中。
或者，如果企业管理器正在运行，则可以将管理模块文件复制到 `<企业管理器主目录>/deploy` 目录。

4. 将 `<企业管理器主目录>/examples/<扩展>/ext/xmltv` 目录的内容复制并粘贴到 `<企业管理器主目录>/ext/xmltv` 目录中。
`<企业管理器主目录>/examples/<扩展>/ext/xmltv` 目录包含使 Workstation 显示特定于扩展的度量标准的选项卡定义。
5. 将任何其他目录的内容复制并粘贴到 `<企业管理器主目录>` 目录下的相应目录中。

例如，将 `<企业管理器主目录>/examples/<扩展>/product/enterprisemanager/plugins` 目录的内容复制到 `<企业管理器主目录>/product/enterprisemanager/plugins` 目录中。

注意：如果已经将企业管理器部署为群集，请将 `<企业管理器主目录>/examples/SOAPerformanceManagement/product/enterprisemanager/plugins` 中的文件复制到群集中 MOM 和所有企业管理器上的 `<企业管理器主目录>/product/enterprisemanager/plugins` 中。

其他目录取决于启用的 SOA 扩展。

6. 重新启动企业管理器。

注意：有关配置和使用特定 SOA 平台的详细信息，请参阅相应平台的相应部分。

验证 CA APM for SOA 部署

通过调用包括 Web 服务或其他 SOA 组件的应用程序并确认“Web 服务”节点存在于调查器树中，可以验证 CA APM for SOA 和与 SOA 相关的扩展的部署。

遵循这些步骤：

1. 启动企业管理器（如果它尚未运行）。
2. 启动应用程序服务器（如果尚未运行）。
重新启动应用程序服务器会启动代理并检测应用程序以启用 SOA 性能监控。
3. 启动使用 Web 服务的任一应用程序并执行一个或多个事务。
4. 启动 Workstation 并打开调查器。
5. 展开安装 CA APM for SOA 的代理。
6. 验证是否可访问代理下的“Web 服务”>“客户端”或“Web 服务”>“服务器”节点。

如果事务包括同一代理上的客户端操作和服务器端操作，则应该可以在代理中看到“Web 服务”>“客户端”节点和“Web 服务”>“服务器”节点。

如果使用已发现 Web 服务来执行事务，则应能展开服务“命名空间”节点来查看与事务的客户端或服务器端关联的度量标准。

在某些情况下，示例应用程序只是通过模拟事务来演示 Web 服务操作，可能不生成任何度量标准。例如，WebLogic 测试客户端针对 CA APM for SOA 检测不到的 WebLogic 托管的服务运行。

7. 选择代理节点，然后单击“概览”选项卡，验证企业管理器 `ws.overview.tv.xml` 文件是否正常加载。

删除 CA APM for SOA

如果您决定从环境中删除 CA APM for SOA 和与 SOA 相关的扩展，请执行下列操作：

- 删除安装和配置的所有代理扩展。
- 删除已部署的企业管理器扩展。

从 Java 代理中删除 CA APM for SOA

1. 停止应用程序服务器。
2. 导航到 `<Agent_Home>/core/config` 目录，并在文本编辑器中打开 `IntroscopeAgent.profile` 文件。
3. 从 `introscope.autoprobe.directivesFile` 属性中删除 `appmap-soa.pbd` 和 `webservices.pbd` 值。保存并关闭文件。
4. 从 `<Agent_Home>` 目录中删除 `webservices.pbd`。
5. 转到 `<Agent_Home>/core/ext` 目录，删除以下文件：
`webServicesAgent.jar`
`BoundaryOnlyTrace.jar`
6. 启动应用程序服务器。

CA APM for SOA 将从 Java 代理中删除。

从 .NET 代理中删除 CA APM for SOA

1. 停止 IIS 应用程序服务器。
2. 打开命令提示符并转到 `<Agent_Home>/ext` 目录。删除以下文件：
 - `wily.WebServicesAgent.ext.dll`
 - `wilyHttpCorrelationTracers.ext.dll`
 - `wilyBoundaryOnlyTrace.ext.dll`
3. 启动 IIS 应用程序服务器。

CA APM for SOA 将从 .NET 代理中删除。

从企业管理器中删除 CA APM for SOA

1. 停止企业管理器。
2. 转到 `<EM_Home>/config/modules` 目录，删除以下文件：
`SPM_ManagementModule.jar`
3. 转到 `<EM_Home>/ext/xmltv` 目录，删除以下文件：
`ws.overview.tv.xml`
`ws.wsdependency.tv.xml`
`ws.wsdeviation.tv.xml`
4. 转到 `<EM_Home>/product/enterprisemanager/plugins` 目录，删除以下文件：
`com.wily.introscope.soa.tracefilters.common_<version>.jar`
`com.wily.introscope.soa.tracefilters.em_<version>.jar`
`com.wily.introscope.soa.dependencymap.common_<version>.jar`
`com.wily.introscope.soa.dependencymap_<version>.jar`
`com.wily.introscope.soa.deviationmetrics_<version>.jar`

5. 转到 *<EM_Home>/ws-plugins* 目录，删除以下文件：

com.wily.introscope.soa.crossprocessviewer.workstation_<version>.jar
com.wily.introscope.soa.dependencymap.common_<version>.jar
com.wily.introscope.soa.dependencymap.ui_<version>.jar
com.wily.introscope.soa.dependencymetrics.typeviewer_<version>.jar
com.wily.introscope.soa.tracefilters.common_<version>.jar
com.wily.introscope.soa.tracefilters.workstation_<version>.jar
com.wily.introscope.ui.tomsawyer_<version>.jar

6. 启动企业管理器。

CA APM for SOA 将从企业管理器中删除。

第3章：监控面向服务的体系结构

CA APM for SOA 使您可以通过查看关于 Web 服务客户端和提供商及其基本操作的信息（包括流程依赖关系、关键操作、流程流和 SOAP 故障）来监控 SOA 基础架构。

本部分介绍用于监控 SOA 基础架构的默认显示板和度量标准，以及如何访问客户端和服务器操作的 SOA 特定信息。

此部分包含以下主题：

[在 CA Introscope® 中查看特定于 SOA 的信息 \(p. 41\)](#)

[使用 SOA 性能显示板 \(p. 44\)](#)

[使用调查器查看 SOA 性能度量标准 \(p. 49\)](#)

[在调查器中查看 Boundary Blame \(p. 58\)](#)

[查看默认 SOA 特定度量标准组 \(p. 59\)](#)

[查看默认 CA APM for SOA 报警 \(p. 59\)](#)

在 CA Introscope® 中查看特定于 SOA 的信息

安装 CA APM for SOA 之后，通过使用 Workstation 和 SOA 特定显示板、选项卡和度量标准，查看关于 Web 服务客户端和 Web 服务提供商的信息，可以监控 SOA 基础架构。

以下列表汇总了该产品向 Workstation 添加的专用于监控 SOA 环境的内容：

- 使用 SOA 性能显示板可以一目了然地监控部署代理中 SOA 环境的总体运行状况。
- SOA 依赖关系视图直观地显示 SOA 组件之间的依赖关系，帮助了解 Web 服务和操作彼此间的关系。
- 使用“依赖关系”选项卡可以查看依赖关系度量标准，如特定服务所依赖的直接或间接关键操作的数目。
- 使用“偏差”选项卡可以查看偏差度量标准，如特定命名空间或操作的平均响应时间偏差。
- 使用“最关键”选项卡可以查看对有最多依赖操作的操作。
- 使用“最依赖”选项卡可以查看对其他操作有最多依赖关系的操作。
- “错误”选项卡提供关于 Web 服务操作生成的 SOAP 故障错误的其他信息。

另外，您可以通过使用调查器深入查看标准度量标准和 SOA 特定度量标准来查看关于 SOA 特定组件的详细信息，并使用事务跟踪查看器来监控跨 JVM 或 CLR 边界的 SOA 相关事务。

关于客户端和服务器 Web 服务和操作

安装 CA APM for SOA 之后重新启动应用程序服务器时，代理将根据其指令文件列表中定义的指令对其找到的应用程序进行检测。在此过程中，代理将从远程过程调用或用于调用操作的 SOAP 消息中识别等同于方法调用的 Web 服务操作。每种 Web 服务中的各个操作都被识别为客户端（或出站）请求或服务器响应，这些操作被置于代理应用程序服务器的“Web 服务|客户端”节点或“Web 服务|服务器”节点下。

大多数情况下，Web 服务的客户端和服务器在单独的 Java 虚拟机 (JVM) 或 Microsoft 公共语言运行时 (CLR) 环境中运行。因此，“客户端”节点和“服务器”节点常列在不同代理下。如果同一 JVM 或 CLR 既处理请求又处理响应，则“客户端”节点和“服务器”节点都列在该代理的“Web 服务”节点下。例如，如果客户端和服务器使用同一 JVM，则调查器树在 `<hostname>|<process_name>|<agent_name>| “Web 服务”` 节点下同时显示“客户端”节点和“服务器”节点。

关于服务命名空间和操作名称

在调查器中展开“客户端”节点和“服务器”节点时，各个 Web 服务将按 Web 服务命名空间显示。Web 服务命名空间是修改版本的 Web 服务统一资源标识符 (URI)。有两种类型的 URI：

- 统一资源定位器 (URL)
- 统一资源名称 (URN)

调查器中显示的命名空间经过修改后可替换具有保留意义的字符，如 URL 中的冒号 (:)，但其他字符与 Web 服务 URL 或 URN 相同。例如，典型的命名空间可能类似如下：

`http://CreditManagement.demobank.ca.com`

可展开任何 Web 服务命名空间以查看其操作。Web 服务中的操作名称等同于方法名称，通过它可在分析性能比较差时深入查看应用程序中的最低级别。例如，典型的操作名称可能描述要执行的具体活动，如 `getSmallBusinessAcctBalance` 或 `isAcctNumberValid`。

关于无法识别的服务和操作

在某些情况下，CA APM for SOA 代理扩展可能无法识别 Web 服务命名空间或操作。无法识别 Web 服务名称或操作可能会影响报告的度量标准和数据的聚合。例如，您可能会看到调查器树中显示的是 `UnknownService` 而不是命名空间，或者显示的是 `UnknownOperationName` 而不是操作名称。

如果针对 Web 服务客户端或服务器看到 `UnknownService` 或 `UnknownOperationName`，请查看以下内容：

- 针对正在使用的协议、API 和应用程序服务器版本的 Web 服务实施。
- 传入和传出消息的构造。

关于虚拟代理

默认情况下，针对每个代理独立显示度量标准。如果要查看多个代理或客户端和服务器命名空间中聚合的 Web 服务度量标准，必须定义虚拟代理。例如，如果 Web 服务客户端和 Web 服务服务器使用同一 JVM，则可能需要定义一个虚拟代理，查看包括客户端和服务器的聚合数据形式的 Web 服务度量标准。

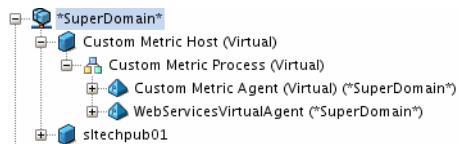
通过修改代理要报告的企业管理器的

`<EM_Home>/config/agentclusters.xml` 文件可以定义虚拟代理。在 `agentclusters.xml` 文件中，指定要聚合的代理和度量标准。例如，可配置一个虚拟代理来收集和聚合所有服务器端服务中平均响应时间的数据，或组合服务器端服务和客户端 Web 服务的平均响应时间度量标准。

下列示例说明如何定义名为 `WebServicesVirtualAgent` 的聚合所有非自定义代理中服务器度量标准的虚拟代理：

```
<agent-cluster name="WebServicesVirtualAgent" domain="SuperDomain" >
    <agent-specifier>(.*)\|^Custom.*\|(.*)\|(.*)</agent-specifier>
    <metric-specifier>WebServices\|Server\|(.*) : Average Response Time
        \((ms)\)</metric-specifier>
    <metric-specifier>WebServices\|Server\|(.*) : Concurrent
        Invocations</metric-specifier>
    <metric-specifier>WebServices\|Server\|(.*) : Errors Per
        Interval</metric-specifier>
    <metric-specifier>WebServices\|Server\|(.*) : Responses Per
        Interval</metric-specifier>
    <metric-specifier>WebServices\|Server\|(.*) : Stall Count</metric-specifier>
    <metric-specifier>WebServices\|Server\|(.*) : SOAP Faults Per
        Interval</metric-specifier>
</agent-cluster>
```

然后可展开 WebServicesVirtualAgent 节点来查看聚合的服务器度量标准。下图说明在调查器树中显示为节点的 agentclusters.xml 文件中定义的虚拟代理：



注意: 有关配置虚拟代理的详细信息,请参阅《CA APM 配置和管理指南》。

使用 SOA 性能显示板

CA APM for SOA 包括可用于监控 SOA 环境总体运行状况的预配置显示板。显示板通过聚合代理中的数据来汇总性能和可用性信息并提供对潜在问题的高度直观可视化。-

通常, 将显示板用作监控环境的起点, 因为使用显示板可以:

- 一目了然地监控 SOA 基础架构关键组件的总体运行状况、性能、可用性以及状态
- 在低级别度量标准发出已超过警告或危险阈值的信号时, 及早获取生产应用环境中存在潜在问题的通知
- 深入查看性能信息, 识别遇到延迟或产生错误的 Web 服务客户端、提供商或操作并进行隔离

CA APM for SOA 的企业管理器扩展包括 SPM_ManagementModule.jar 文件中打包的特定于 SOA 的预配置显示板。

遵循这些步骤:

1. 启动企业管理器 (如果目前未运行)。
2. 启动 Workstation 并登录到安装有 SOA 扩展的企业管理器。
3. 单击 “Workstation” > “新建控制台”。

4. 从“显示板”下拉列表中选择一种 SOA 性能显示板。

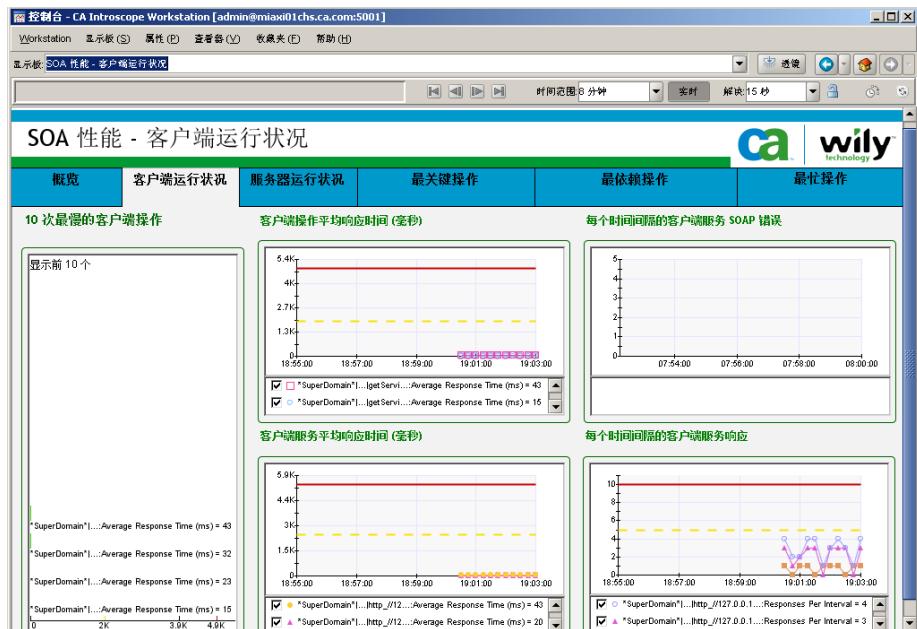
例如，选择 SOA 性能“概览”显示板作为监控 Web 服务的起点。SOA 性能“概览”显示板显示关于已部署服务总体运行状况的信息。



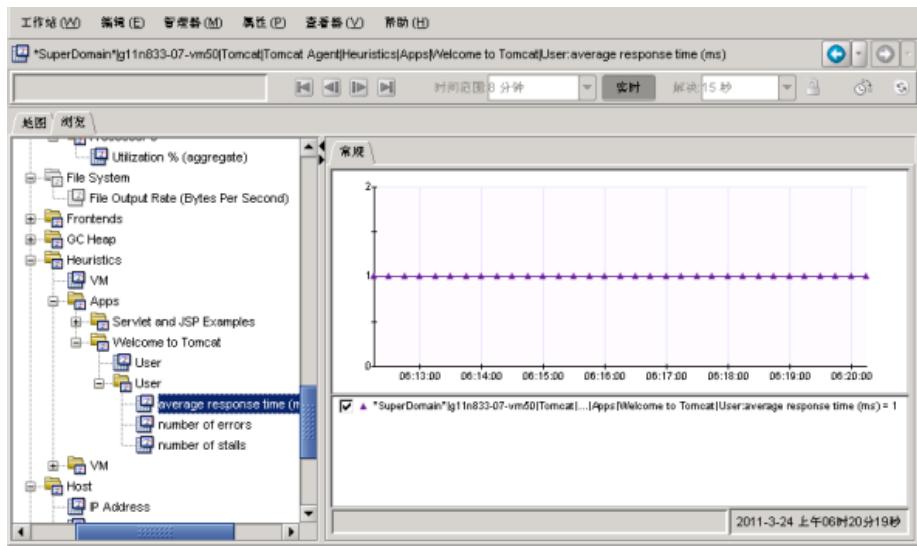
从 SOA 性能“概览”显示板中，您可以导航以下显示板，以便查看有关客户端或服务器操作的更多详细信息：

- 客户端运行状况
- 服务器运行状况
- 最关键操作
- 最依赖操作
- 最忙操作

例如，如果问题看似与客户端平均响应时间有关，您可以双击“客户端运行状况”选项卡，查看显示板中关于最慢客户端操作、平均响应时间、每个时间间隔内的响应数以及每个时间间隔内的 SOAP 故障数的详细信息： -



5. 您可以从任何显示板中，双击某个服务名或操作名，打开 Workstation 调查器，以进一步进行分析。例如，如果在“客户端运行状况”显示板中选择一个速度很慢的服务，将打开调查器，显示该客户端服务的相关度量标准：



注意：有关启动和使用 Workstation、访问显示板或打开和导航调查器的信息，请参阅《CA APM Workstation 用户指南》。

详细信息：

[使用调查器查看 SOA 性能度量标准 \(p. 49\)](#)

SOA 性能“概览”显示板

SOA 性能—“概览”显示板提供受监控 Web 服务的概览，Web 服务分为客户端服务操作和服务器端服务操作。

“概览”显示板包括以下内容：

- 最慢的客户端服务和服务器服务的列表
- 客户端服务和服务器服务的平均响应时间的图表
- 客户端服务和服务器服务的响应时间、SOAP 故障、错误和停顿的报警指示器

您可以使用此显示板查看要监控 Web 服务的汇总信息。

SOA 性能“客户端运行状况”显示板

SOA 性能—“客户端运行状况”显示板从客户端角度提供有关 Web 服务操作的详细信息。

“客户端运行状况”显示板包括以下内容：

- 十个速度最慢的客户端操作的列表
- 每个客户端操作的聚合平均响应时间
- 每个客户端 Web 服务中所有操作的聚合平均响应时间
- 时间间隔内每个客户端 Web 服务的聚合 SOAP 故障总数
- 时间间隔内每个客户端 Web 服务的聚合响应总数

当得到关于正监控的客户端 Web 服务存在问题的报警时，可使用此显示板查看关于客户端操作的更多详细信息。

SOA 性能“服务器运行状况”显示板

SOA 性能—“服务器运行状况”显示板从服务器角度提供有关 Web 服务操作的详细信息。

“服务器运行状况”显示板包括以下内容：

- 十个速度最慢的 Web 服务操作的列表
- 每个服务器端操作的平均响应时间
- 每个服务器端 Web 服务中所有方法或操作的聚合平均响应时间
- 时间间隔内每个服务器端 Web 服务的聚合 SOAP 故障总数
- 时间间隔内每个服务器端 Web 服务的聚合响应总数

当得到关于正监控的服务器端 Web 服务存在问题的报警时，可使用此显示板查看关于服务器端操作的更多详细信息。

SOA 性能“最关键操作”显示板

SOA 性能—“最关键操作”显示板提供具有最高数目依赖操作的操作的偏差度量标准。由于关键操作最可能影响其他操作的成功与否，因此应密切监控这些操作。

“最关键操作”显示板包括具有最高数目依赖操作的操作列表以及以下内容的图表：

- 平均响应时间偏差
- 每个时间间隔的错误偏差
- 每个时间间隔的响应偏差

高偏差值可能表明关键操作未可靠地执行或下游操作执行不正确。

如果启用对其他 SOA 平台（如 CA APM for TIBCO BusinessWorks）的监控，则此显示板还可包括关于该扩展的最关键流程的信息。

SOA 性能“最依赖操作”显示板

SOA 性能—“最依赖操作”显示板提供具有最高数目对其他操作的依赖关系的操作的偏差度量标准。

“最依赖操作”显示板包括对其他操作的依赖关系最多的操作列表以及以下内容的图表：

- 平均响应时间偏差
- 每个时间间隔的错误偏差
- 每个时间间隔的响应偏差

高偏差值可能表明上游操作未可靠地执行。

如果启用对其他 SOA 平台（如 CA APM for TIBCO BusinessWorks）的监控，则此显示板还可包括关于该扩展的最依赖流程的信息。

SOA 性能“最忙操作”显示板

SOA 性能—“最忙操作”显示板提供关于客户端和服务器端所经历的每时间间隔内 Web 服务最高调用率的详细信息。

“最忙操作”显示板包括以下内容：

- 最繁忙客户端操作和服务器操作的列表
- 所有客户端操作每时间间隔内的总响应数
- 所有服务器端操作每时间间隔内的总响应数

您可以使用此显示板查看关于正监控的客户端 Web 服务和服务器 Web 服务的当前吞吐量的更详细信息。

使用调查器查看 SOA 性能度量标准

显示板为您提供客户端和服务器运行状况以及潜在故障点（如最关键或最忙操作）的概览，在这种情况下，通过调查器便可深入查看关于 SOA 环境的具体详细信息以帮助确定问题的根源。

通过调查器，可从顶级应用程序的聚合度量标准开始分析，向下一直到对各个服务中单个操作的具体度量标准进行分析，找到出现性能低或错误活动的根源。

可用度量标准

除每个受监控操作或服务命名空间的标准 CA Introscope® 度量标准之外，CA APM for SOA 还提供若干个特定于 SOA 的度量标准。- 单个操作的度量标准会聚合为单个 Web 服务的度量标准，可用于评估应用程序的客户端和服务器端性能。

下列标准度量标准及 SOA 特定度量标准通常可用于作为 Web 服务实施的应用程序：

- **平均响应时间 (毫秒)**—在 15 秒时间段内完成调用方法、操作或服务的平均时间（毫秒）。
- **平均响应时间偏差**—每个时间间隔的平均响应时间的平均偏差。
- **并发调用**—15 秒时间段结束时正在进行但尚未完成的并发请求数。
- **每个时间间隔的错误**—15 秒时间段内生成的异常、HTTP 错误或 SOAP 故障的数目。
- **每个时间间隔偏差的错误**—每个时间间隔内异常、HTTP 错误或 SOAP 故障数与平均数的偏差。
- **每个时间间隔的响应**—15 秒时间段内完成的操作请求或服务请求数。
- **每个时间间隔偏差的响应**—每个时间间隔内完成调用数与平均数的偏差。
- **停顿计数**—30 秒时间段结束时未完成的请求数。
- **每个时间间隔的 SOAP 故障**—15 秒时间段内 SOAP 引擎生成或消耗的 SOAP 消息错误数。
- **关键直接数**—直接依赖当前成功执行操作的下游操作数。
- **关键间接数**—直接或间接依赖当前成功执行操作的下游操作数。
- **依赖直接数**—当前操作直接依赖的上游操作的数目。
- **依赖间接数**—当前操作直接或间接依赖的上游操作的数目。

查看“概览”选项卡上的汇总度量标准

通常，使用调查器开始查找问题的根本原因的最佳位置是“概览”选项卡。“概览”选项卡上显示的信息取决于所选择的调查器节点。例如，选择代理节点时，“概览”选项卡将显示代理中已检测的所有应用程序的高级运行状况指示器。**Workstation** 默认情况下提供该“概览”选项卡。**CA APM for SOA** 还为以下节点提供了更多“概览”选项卡：

- “Web 服务”节点—选择“Web 服务”节点时，“概览”选项卡将显示与选定代理应用程序关联的所有客户端或服务器 Web 服务命名空间。如果代理中既有客户端 Web 服务，又有服务器 Web 服务，则“概览”选项卡将在两个独立表中显示客户端命名空间和服务器命名空间。
- “客户端”节点—选择“客户端”节点时，“概览”选项卡将列出与选定代理应用程序上的客户端操作关联的 Web 服务命名空间。
- “服务器”节点—选择“服务器”节点时，“概览”选项卡将列出与选定代理应用程序上的服务器端操作关联的 Web 服务命名空间。
- “*<web_service_namespace>*”节点—选择单个 Web 服务命名空间时，“概览”选项卡将列出一些操作，这些操作属于由选定的 *<web_service_namespace>* 标识的选定客户端或服务器 Web 服务。
- “*<operation_name>*”节点—选择单个操作时，“概览”选项卡中显示的图表会显示使用标准 CA Introscope® 度量标准的活动的当前级别，以及选定 *<operation_name>* 所标识操作的“每个时间间隔的 SOAP 故障”。

遵循这些步骤：

1. 打开调查器，选择某个代理节点，然后在“查看器”窗格中单击“概览”选项卡。

“概览”选项卡会显示选定代理中已检测的每个 Web 服务的报警指示器。

2. 展开代理节点，在调查器树中选择“Web 服务”。单击“概览”选项卡。

将显示分别包含客户端和服务器 Web 服务度量标准的两个表。您具有以下选项：

- 在“客户端”或“服务器”表中选择某一特定的服务命名空间。此时将显示该服务的“平均响应时间”图表。
- 单击要查看的度量标准的度量标准列标题可更改显示的图表。例如，要显示“每个时间间隔的错误”的图表，请单击“每个时间间隔的错误”度量标准列标题。

3. 展开“Web 服务”节点，在调查器树中选择“客户端”或“服务器”节点。单击“概览”选项卡。
仅显示客户端命名空间列表或服务器命名空间列表。
4. 展开“客户端”或“服务器”节点，在调查器树中选择特定的 `<web_service_namespace>`。单击“概览”选项卡。
此时将显示选定 Web 服务命名空间的操作和相关度量标准。
5. 展开“`<web_service_namespace>`”节点，在调查器树中选择 `<operation_name>`。单击“概览”选项卡。
此时将显示该操作的度量标准的图表。

查看“依赖关系”选项卡上的依赖关系度量标准

除标准 CA Introscope® 度量标准之外，CA APM for SOA 还提供用于确定 Web 服务事务中最关键和最依赖操作的度量标准。这些度量标准基于每个操作的直接和间接依赖关系的数目。

要在“依赖关系”选项卡上查看这些基于依赖关系的度量标准，请选择相应的调查器节点。-

在“依赖关系”选项卡上显示的信息取决于所选择的调查器节点。例如，当您选择：

- “Web 服务”节点，“依赖关系”选项卡会显示与选定代理应用程序关联的客户端和服务器 Web 服务命名空间、操作和依赖关系度量标准。
- “客户端”节点，“依赖关系”选项卡会显示与选定代理应用程序关联的客户端和服务器 Web 服务命名空间、操作和依赖关系度量标准。
- “服务器”节点，“依赖关系”选项卡将显示与选定代理应用程序上的服务器端操作关联的 Web 服务命名空间、操作和依赖关系度量标准。
- 单个 `<web_service_namespace>` 节点，“依赖关系”选项卡会显示与选定 `<web_service_namespace>` 标识的选定客户端或服务器 Web 服务关联的操作和依赖关系度量标准。

关于直接和间接操作依赖关系

在 Web 服务事务中，调用其他操作的任何操作都会创建一个依赖关系链，其中被调用的操作依赖于调用操作。来自被调用操作的其他调用或回复会创建其他依赖关系。操作之间的依赖关系分直接或间接两种，具体取决于操作在依赖关系链中的位置。

例如，如果操作 A 调用操作 B，操作 B 调用操作 C，则事务如下所示：

A → B → C。

在该事务中：

- 操作 A 和 B 之间存在直接依赖关系
- 操作 B 和 C 具有直接依赖关系
- 操作 A 和 C 之间存在间接依赖关系

根据操作在依赖关系链中的位置，也可将其标识为发生在另一操作的上游或下游。在包含多个操作的事务中，存在直接依赖关系的操作为直接上游或下游操作。在上面的示例中，操作 A 位于操作 B 和 C 的上游，但仅位于操作 B 的直接上游。

同样，间接的依赖关系可发生在未直接调用的操作的上游或下游。例如，如果操作 A 调用操作 B，然后操作 B 调用操作 C 和 D，则 C 和 D 与操作 A 之间都存在下游间接依赖关系。

关于关键操作和度量标准

Web 服务中的操作可视为关键或（和）依赖，具体取决于与其他操作之间的交互。

关键操作是下游操作所依赖的操作。下游操作可能与关键操作之间存在直接或间接依赖关系，但若要执行下游操作，必须成功执行关键操作。

与关键操作关联的度量标准有两种：

- “关键直接数”度量标准仅跟踪直接位于某一操作上游的操作的数目。如果某一操作不依赖于任何上游操作，也就是说该操作不位于任何其他操作的下游，则其“关键直接数”度量标准为零。
- “关键间接数”度量标准跟踪直接和间接位于某一操作上游的操作的数目。

例如，如果 `getAccountNum` 操作的“关键间接数”度量标准值是 5，则表明有五个直接或间接操作需要在执行 `getAccountNum` 操作后执行。

关于依赖操作和度量标准

依赖操作是依赖上游操作的执行的操作。依赖操作可能与上游操作之间存在直接或间接依赖关系，但若要执行依赖操作，必须成功执行上游操作。

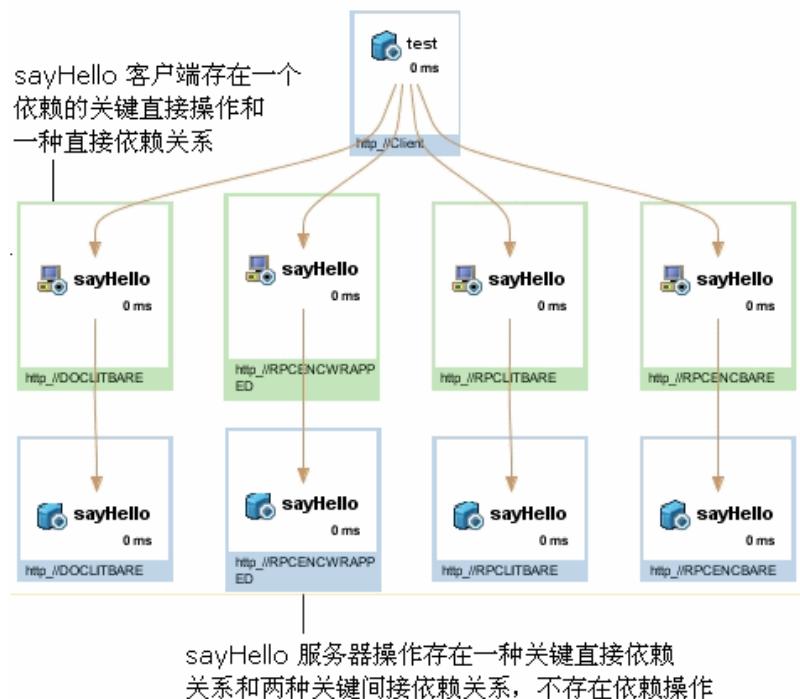
与关键操作关联的度量标准有两种：

- “依赖关系直接数”度量标准跟踪直接位于某一操作上游的操作的数目。如果某一操作没有任何依赖操作，也就是说该操作不位于任何其他操作的上游，则其“依赖关系直接数”度量标准为零。
- “依赖关系间接数”度量标准跟踪操作的直接和间接下游操作的数目。

例如，如果 *sendConfirmation* 操作的“依赖关系间接数”度量标准值是 8，则表明在执行 *sendConfirmation* 操作前必须执行八个直接或间接操作。

进一步了解关键和依赖关系度量标准

为了更好地了解“关键直接数/间接数”和“依赖关系直接数/间接数”度量标准的作用原理，可参照下面的示例，其中测试服务器操作调用四个 *sayHello* 客户端操作。每个客户端操作又调用一个下游 *sayHello* 服务器操作，如下列示例依赖关系地图中所示：



该示例中应用了以下关键和依赖关系度量标准值：

测试服务器

由于不存在上游操作，因此关键操作的度量标准值为：关键直接数 = 0，关键间接数 = 0。

由于存在 4 个直接下游操作和 4 个其他间接下游操作，因此依赖操作的度量标准值为：依赖关系直接数 = 4，依赖关系间接数 = 8

sayHello 客户端

由于存在直接上游操作，因此关键操作的度量标准值为：关键直接数 = 1，关键间接数 = 1。

由于存在直接下游操作，因此依赖操作的度量标准值为：依赖关系直接数 = 1，依赖关系间接数 = 1。

sayHello 服务器

由于在 sayHello 上存在直接上游依赖关系，在测试服务器上存在间接依赖关系，因此关键操作的度量标准值为：关键直接数 = 1，关键间接数 = 2。

由于不存在直接或间接下游操作，因此依赖操作的度量标准值为：依赖关系直接数 = 0，依赖关系间接数 = 0。

查看“依赖关系”选项卡上的依赖关系度量标准：

1. 展开代理节点，在调查器树中选择“Web 服务”，然后在“查看器”窗格中单击“依赖关系”选项卡以查看分别包含客户端和服务器依赖关系度量标准的两个表格。

对于每个操作，“依赖关系”选项卡都会显示“关键直接数”、“关键间接数”、“依赖关系直接数”、“依赖关系间接数”度量标准值。

2. 展开“Web 服务”节点，并在调查器树中选择“客户端”或“服务器”节点，然后在“查看器”窗格中单击“依赖关系”选项卡即可仅显示客户端命名空间或仅显示服务器命名空间和关联的依赖关系度量标准。
3. 展开“客户端”或“服务器”节点，在调查器树中选择某一特定的 `<web_service_namespace>`，然后在“查看器”窗格中单击“依赖关系”选项卡即可显示选定 Web 服务命名空间的操作和相关依赖关系度量标准的列表。

查看“偏差”选项卡上的偏差度量标准

CA APM for SOA 提供了用于监控“平均响应时间”、“每个时间间隔的错误”和“每个时间间隔的响应”值与平均值的偏差的度量标准。低偏差值表示度量标准数据（如“平均响应时间”）聚集在平均值附近。高偏差值表示度量标准数据分布范围广，且高于或低于平均值。

注意：如果在某一间隔中未调用某操作，则该操作的“平均响应时间”度量标准的值为零。平均响应时间计算不包含零值度量标准。因此，有时“偏差”选项卡看起来会停止监控。这种表面上的暂停是正常的。

要查看偏差度量标准，请选择某一调查器节点，然后单击“偏差”选项卡。例如：

- “Web 服务”节点会显示与选定代理应用程序关联的客户端和服务端 Web 服务命名空间、操作和偏差度量标准。
- “客户端”节点会显示与该代理关联的 Web 服务命名空间、操作和偏差度量标准。
- “服务器”节点会显示与该服务器关联的 Web 服务命名空间、操作和偏差度量标准。
- 单个 <web 服务命名空间> 节点显示与选定客户端或服务器 Web 服务关联的操作和偏差度量标准。

注意：默认情况下，只为命名空间级别计算偏差。要为 Web 服务、客户端和服务器节点启用偏差，请配置

[com.wily.introscope.soa.deviation.metric.expressionlist](#) (p. 320) 属性以计算其他度量标准。

偏差度量标准计算

偏差度量标准需要 CA APM for SOA 来计算平均值。该产品根据滚动的“n”天内的操作度量标准确定平均值。计算滚动平均值的默认天数是 7。要调整天数，请设置 `com.wily.introscope.soa.deviation.mean.days` 属性值。

为了最大程度地降低对企业管理器的性能影响，CA APM for SOA 会限制可为其计算偏差度量标准的操作数目。默认情况下，为每一类型的依存关系度量标准的前 25 次操作计算偏差度量标准。要更改计算偏差的操作的最大数目，可在企业管理器属性文件中修改 `com.wily.introscope.soa.deviation.count.per.metric` 属性。

要减少生成偏差度量标准的性能开销，还可有选择地启用或禁用特定类型的偏差度量标准。例如，可将配置属性设置为仅生成“平均响应时间偏差”度量标准。或将 `com.wily.introscope.soa.deviation.enable` 属性设置为 `false`，从而不计算偏差度量标准。

详细信息:

[配置企业管理器属性 \(p. 305\)](#)

在“最关键”选项卡上查看关键操作度量标准

SOA 环境中最关键的操作是受最多数量操作依赖的操作。这些操作是最重要的潜在故障点，应通过调查器中的“最关键操作”显示板和“最关键”选项卡对其进行密切监控。在调查器中选择“Web 服务”节点时，可在“最关键”选项卡上查看最关键操作及其偏差度量标准列表。

遵循这些步骤:

1. 展开代理节点，在调查器树中选择 Web 服务，然后在“查看器”窗格中单击“最关键”选项卡。
2. 在关键操作列表中选择单个操作可查看该操作的偏差度量标准的图表。

在“最依赖”选项卡上查看依赖操作

SOA 环境中最依赖的操作是存在最多数量依赖关系的操作。在调查器中选择“Web 服务”节点时，可在“最依赖”选项卡上查看最依赖操作的列表及其偏差度量标准。

遵循这些步骤:

1. 展开代理节点，在调查器树中选择 Web 服务，然后在“查看器”窗格中单击“最依赖”选项卡。
2. 选择依赖操作列表中的单个操作，以查看该操作的偏差度量标准的图表。

在“错误”选项卡上查看 SOAP 故障和错误度量标准

调查器默认情况下提供“错误”选项卡，其中显示在调查器树中所选择节点的错误消息和详细信息。如果安装了 CA APM for SOA 扩展，则“错误”选项卡还会显示有关客户端和服务器端 SOAP 故障的信息。

“每个时间间隔的错误”度量标准是代理生成的标准 CA Introscope® 度量标准。“每个时间间隔的 SOAP 故障”度量标准是 CA APM for SOA 代理扩展生成的 SOA 特定度量标准。- 这些度量标准之间不存在直接关联。

SOAP 故障是 Web 服务返回错误响应而非正常响应时 SOAP 引擎生成或消耗的消息。SOAP 故障可以通过响应某一条件而自动生成或在应用程序逻辑中进行明确编码。捕获和处理故障的方式取决于所使用的 SOAP 引擎。

“每个时间间隔的错误”度量标准跟踪抛出的异常和 HTTP 错误。在“每个时间间隔的错误”度量标准中可能存在 SOAP 故障的对应错误，具体取决于 Web 服务的业务逻辑和应用程序服务器的行为。

如果应用程序服务器遇到问题并发出 SOAP 错误，则客户端和服务器“每个时间间隔的 SOAP 错误”度量标准都会记录该错误。如果应用程序服务器抛出与 SOAP 错误相关的异常，则其“每个时间间隔的错误”也会更新。如果客户端上不存在将 SOAP 故障捕获为异常的代码，将不会更新客户端的“每个时间间隔的错误”度量标准。

由于“每个时间间隔的错误”度量标准仅跟踪已经过 CA APM for SOA 检测的应用程序服务器上或 SOAP 堆栈方法中的已捕获异常和错误代码，因此该度量标准可能不包含 SOAP 故障。

遵循这些步骤：

1. 展开代理节点，在调查器树中选择“Web 服务”或展开“Web 服务”节点，并选择“客户端”或“服务器”节点。
2. 在“查看器”窗格中右键单击“错误”选项卡。
“错误”选项卡的顶部窗格会列出每个错误的时间、说明和错误消息。在错误消息的开始，SOAP 引擎生成的错误消息会显示 SOAP 错误。
3. 选择单个错误可显示该错误的详细信息。
“堆栈视图”选项卡中会显示 SOAP 故障的详细信息，其中消息文本用红色突出显示。

[在调查器中查看 Boundary Blame](#)

Boundary Blame 将度量标准隔离为仅包括调查器树中的前端和后端组件，以帮助您确定响应时间问题是否是应用程序服务器的内部问题，例如，慢 Servlet 的结果或对后端系统（如数据库服务器）外部调用的结果。CA APM for SOA 通过自动检测 Web 服务对后端系统的调用将该功能扩展到 SOA 组件。

默认情况下，用于确定前端和后端组件度量标准的这一扩展功能处于启用状态。

注意：有关应用程序的配置、禁用或 Boundary Blame 自定义的信息，请参阅《CA APM Java 代理实施指南》、《CA APM .NET 代理实施指南》和《CA APM Workstation 用户指南》。

查看默认 SOA 特定量标准组

CA APM for SOA 包括用于定义默认显示板和报警的默认度量标准组。在自定义显示板或报警中也可以使用默认度量标准组。默认度量标准组包含在 CA APM for SOA 的企业管理器扩展中，打包在 **SPM_ManagementModule.jar** 文件中。可使用 **Workstation** 管理模块编辑器查看默认度量标准组。

遵循这些步骤：

1. 在调查器中，单击“Workstation” > “新建管理模块编辑器”。
2. 展开“*超级域*” > “管理模块” > “SOA 性能管理 (*超级域*)”。
3. 展开“度量标准组”节点可查看为 SOA 性能管理模块定义的所有度量标准组。
4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

可根据需要修改任何度量标准组的默认设置或创建自己的自定义度量标准组。

注意：有关创建和使用度量标准组的详细信息，请参阅《CA APM 配置和管理指南》。

查看默认 CA APM for SOA 报警

CA APM for SOA 包括预配置显示板中使用的默认报警定义。默认报警定义包含在 CA APM for SOA 的企业管理器扩展中，并打包在 **SPM_ManagementModule.jar** 文件中。

可以使用 **Workstation** 管理模块编辑器查看默认报警定义。还可以将 CA APM for SOA 管理模块扩展为包括自定义报警定义和通知类型，或在自定义显示板中使用默认报警定义。

遵循这些步骤:

1. 在调查器中，单击“Workstation” > “新建管理模块编辑器”。
2. 展开“*超级域*” > “管理模块” > “SOA 性能管理 (*超级域*)”。
3. 展开“报警”节点，查看为 CA APM for SOA 定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

尤其是，应检查“警告”和“危险”阈值的默认设置并在必要时调整值。您可能还希望添加通知或修正操作。

下表概述了与预配置 CA APM for SOA 报警定义关联的默认警告和危险阈值。

报警	警告	危险
客户端操作平均响应时间	2000 毫秒	5000 毫秒
服务器操作平均响应时间		
服务器服务平均响应时间		
客户端服务平均响应时间	2500 毫秒	5500 毫秒
每个时间间隔的客户端操作错误	10 个错误	15 个错误
每个时间间隔的客户端服务错误		
每个时间间隔的服务器操作错误		
每个时间间隔的服务器服务错误		
每个时间间隔的客户端操作响应	5 次响应	10 次响应
每个时间间隔的客户端服务响应		
每个时间间隔的服务器操作响应		
每个时间间隔的服务器服务响应		
每个时间间隔的客户端操作 SOAP 错误	10 个错误	15 个错误
每个时间间隔的客户端服务 SOAP 错误		
每个时间间隔的服务器操作 SOAP 错误		
每个时间间隔的服务器服务 SOAP 错误		
客户端操作停顿计数	1 个停顿	2 个停顿
客户端服务停顿计数		
服务器操作停顿计数		
服务器服务停顿计数		

使用 CA APM for SOA 为代理创建自定义报警

1. 在管理模块编辑器中，单击“元素”>“新建报警”。
2. 在“名称”字段中，输入报警的名称。
3. 从“管理模块”下拉列表中选择“SOA 性能管理”，然后单击“确定”。-

注意：有关创建报警和自定义报警通知的详细信息，请参阅《CA APM 配置和管理指南》。

第 4 章： 使用 SOA 依赖关系地图

SOA 依赖关系地图会直观地显示已在 SOA 环境中部署的 Web 服务或业务流程工作流。可使用该地图查看和导航组件之间的实时依赖关系、监控相关组件之间的关键度量标准以及启动事务跟踪或深入查看问题服务的依赖关系。

此部分包含以下主题：

- [了解 SOA 依赖关系地图的用法 \(p. 63\)](#)
- [SOA 依赖关系视图获取数据的方式 \(p. 65\)](#)
- [了解 SOA 依赖关系地图中的上下文 \(p. 70\)](#)
- [显示 SOA 依赖关系视图 \(p. 71\)](#)
- [选择物理或逻辑视图 \(p. 73\)](#)
- [选择内容类型 \(p. 74\)](#)
- [为依存关系映射节点设置主要度量标准 \(p. 74\)](#)
- [为依赖关系地图节点选择工具提示度量标准 \(p. 75\)](#)
- [更改显示的依赖关系级别 \(p. 76\)](#)
- [在 SOA 依赖关系地图内导航 \(p. 78\)](#)
- [保存 SOA 依赖关系地图图像 \(p. 80\)](#)
- [从视图节点启动事务跟踪 \(p. 83\)](#)
- [群集的 SOA 依赖关系视图 \(p. 83\)](#)

了解 SOA 依赖关系地图的用法

在典型的 SOA 环境中，在可用的 Web 服务或业务流程之间存在复杂的关系。但由于那些服务是松散耦合的，因此组件之间的关系通常难以跟踪或理解。SOA 依赖关系地图会直观地显示已部署的服务，有助于监控和了解多个组件之间的关系。

了解 SOA 环境的挑战

在大多数组织中，开发和部署服务会涉及多个组，且通常不存在概述如何使用服务的单个结构图。在某些情况下，可通过注册表记录已部署服务，但该列表不一定是最新的。即使在服务注册表维护良好的环境中，这种注册表也不能随时供适当人员使用，或准确描述服务之间的复杂关系。

例如，对于应用程序来说，嵌入由不同组开发的多层服务是很常见的。这些其他服务层并非十分明显，应用开发人员可能对其一无所知，但这些服务层的性能可以成为依赖于它们的应用程序的关键任务问题。公开这些“隐藏”服务会为查找、确定和解决问题提供更高级别的可见性。

了解 SOA 依赖关系地图所提供的内容

要应对 SOA 环境的挑战，需要以自动和及时的方式了解 SOA 环境中的所有服务。还需要能够导航组件之间的关系，以了解一种服务被其他服务使用的位置和方式或存在潜在瓶颈的位置。如果要实时在生产环境中监控所有服务的运行状况和可用性，该信息十分重要。

通过 SOA 依赖关系地图可了解这种信息。使用依赖关系地图，可一目了然并实时了解以下内容：

- 实际部署了哪些服务
- 何时部署新服务或何时更改依赖关系
- 服务之间的关系如何
- 客户端请求和服务器端响应的执行方式

了解依赖关系和 SOA 术语

在 SOA 依赖关系地图的上下文中，通常使用术语 **服务** 表示一系列设计用于完成特定结果的操作。在不同的环境中，有关服务的组件的详细信息会有所不同或定义不同。例如，选择依赖关系地图的“服务”视图可显示有关 Web 服务、业务流程、适配器或消息服务的信息。

某个操作调用后续操作时即存在依赖关系。后续操作可能属于完全独立的服务，且可能在不同的应用程序服务器上运行。例如，输入旅行网站的用户可以启动预定机票事务。旅行网站的预订服务可能调用机票预订服务来预定座位，然后调用信用卡处理服务来订票。对于该示例，启动事务的旅行网站上的服务依赖于机票预订服务和信用卡处理服务来完成该事务。

依赖关系通常是低级服务组合以形成高级服务的结果。例如，旅行网站的预订服务依赖于机票预订服务和信用卡处理服务的成功处理。这些服务中任何出现故障或性能问题都会影响旅行网站的预订服务的行为。

根据您选择的上下文，SOA 依赖关系地图可显示已部署的所有顶级服务或具有依赖关系的各个服务的详细结构以及依赖操作的性能信息。由于范围和显示的详细信息可以更改，SOA 依赖关系地图为分析 SOA 环境中的问题提供了很高的灵活性。通过使用 SOA 依赖关系地图，可在流程流中跟踪服务和操作之间的依赖关系，以帮助您确定响应时间慢、SOAP 错误或其他问题的根本原因。

了解使用 SOA 依赖关系地图可执行的任务

使用 SOA 依赖关系地图可完成以下重要的任务：

- 查看服务当前准确的部署信息，而不是初始设计信息。
- 通过查看服务依赖关系和单击图形节点深入查看有关调查器树中组件的详细信息，来研究并解决服务问题。
- 通过提供应用程序依赖关系的物理和逻辑视图与 IT 设计人员一起计划和执行企业范围内的 SOA 项目。
- 创建图像的快照映像，以与组织中的其他利益相关者共享有关服务依赖关系和服务问题的信息。
- 在没有应用程序专家帮助的情况下，了解监控代理、服务和操作的依赖关系以及这些依赖关系如何影响服务性能。
- 直接从 SOA 依赖关系地图启动问题服务上的“事务跟踪”，可在不需要设置手动“事务跟踪”的情况下开始诊断问题。

SOA 依赖关系视图获取数据的方式

首次安装 CA APM for SOA 并启动应用程序服务器时，代理会自动发现找到的所有 Web 服务或业务流程，并将该信息发送到企业管理器。

调用和执行这些服务时，代理会跨进程跟踪调用序列，并将该信息传递至企业管理器。企业管理器使用调用先后顺序信息来识别服务之间的依赖关系。企业管理器将发现的依赖关系保存为文件，并将图形数据发送到 Workstation 进行显示。

初始发现后，代理会定期检查新的应用程序以及对以前发现的依赖关系所做的更改。如果检测到新服务，代理会将有关服务调用的详细信息发送到企业管理器，企业管理器将新的图形数据发送到 Workstation，以更新显示任何以前的未知服务或任何已知服务的新发现依赖关系的 SOA 依赖关系视图。

如果发现新的 SOA 依赖关系，企业管理器会检查是否启用了逻辑对等规则，如果是，则会确定新的依赖关系是否在逻辑上对等于现有依赖关系。如果新的依赖关系是唯一的，则将其添加至企业管理器上所存储的永久性依赖关系数据中。Workstation 每 15 秒检查一次数据存储，以查看 SOA 依赖关系视图的数据是否已更改，然后在需要时更新显示的 SOA 依赖关系视图。

关于 SOA 依赖关系视图的永久性数据

企业管理器将发现的 SOA 依赖关系保存在本地数据存储中，这样不需要重新发现即可提供信息。SOA 依赖关系的数据存储包括 `<EM_Home>/data/dependencymap` 目录中的下列文件：

- `dependencymap.sav` 包含最近保存的依赖关系数据。
- `dependencymap.bak` 提供了以前保存的 SOA 依赖关系的备份副本。

企业管理器会每小时以及在每次手动关闭企业管理器时将所有发现的依赖关系保存在 `dependencymap.sav` 文件。企业管理器还会在 `dependencymap.sav` 文件保存的间隔期间将当前发现的 SOA 依赖关系视图中的依赖关系保存在内存中。

企业管理器启动时会从 `dependencymap.sav` 文件加载依赖关系，这样，CA APM for SOA 不必重新发现以前发现的依赖关系。如果企业管理器在没有保存 `dependencymap.sav` 文件的新副本的情况下突然停止，`dependencymap.sav` 文件中数据的上次保存时间可多达一个小时之前。如果找不到 `dependencymap.sav` 文件，企业管理器将等待代理重新发现服务依赖关系，然后重新创建文件。

强制企业管理器重新发现依赖关系

在大多数情况下，从 `dependencymap.sav` 文件加载永久性数据可在无需重新发现的情况下提供了最新信息。但也存在这样的情况，您可能希望企业管理器在启动时忽略以前保存的依赖关系并强制重新发现所有依赖关系。例如：

- 如果要更改逻辑对等启发的启用或禁用状态，还可能希望强制企业管理器重新发现所有依赖关系并将新规则应用于依赖关系。有关逻辑等效启发的详细信息，请参阅“[关于逻辑等效规则](#)” (p. 69)。有关设置逻辑对等启发的属性的详细信息，请参阅[配置企业管理器属性](#) (p. 295)。
- 如果在企业管理器停止时断开与代理的连接，该代理的地图节点和依赖关系可能继续显示在地图中直到过期，默认情况下，保留 60 天。您可能希望强制企业管理器重新发现依赖关系，以在中断连接的代理的数据过期之前将其从依赖关系地图中删除。

忽略或删除保存的依赖关系数据

如果希望企业管理器在启动时忽略保存的依赖关系数据并重新发现依赖关系，则必须删除、移动或重命名 *dependencymap.sav* 和 *dependencymap.bak* 文件。

删除、移动或重命名保存的 SOA 依赖关系地图文件：

1. 停止企业管理器。
2. 导航到 *<EM_Home>/data/dependencymap* 目录。
3. 删 除、移动或重命名 *dependencymap.sav* 文件。例如，将文件重命名为 *old_dependencymap.savedcopy*，以阻止企业管理器使用文件中数据来显示依赖关系地图。
4. 删 除、移动或重命名 *dependencymap.bak* 文件。例如，将文件重命名为 *old_dependencymap.backupcopy*，以阻止企业管理器使用文件来显示依赖关系地图。
5. 启动企业管理器可重新发现当前的依赖关系并创建新的依赖关系地图和 *dependencymap.sav* 文件。

断开连接并重新安装代理

如果在企业管理器正在运行时从企业管理器断开与某个代理的连接，则该代理的地图节点和依赖关系变为非活动状态。如果从工作站卸载代理，卸载事件会导致该代理的所有地图节点和依赖关系从地图中删除，包括以删除服务起始或结束的所有依赖关系。

重新安装代理时，企业管理器会重新发现其所有的依赖关系并将其添加回地图中。

使用新服务和操作更新依赖关系地图

代理只要发现新服务或操作，即会将更新信息发送到企业管理器。企业管理器更新其依赖关系地图存储并通知工作站新数据可在 SOA 依赖关系地图中显示。工作站收到该通知后会显示“可用数据已经更改”消息。然后可重新加载依赖关系地图，以工作站中显示新服务或操作。

重新加载 SOA 依赖关系地图：

1. 检查“可用数据已经更改”消息是否在“SOA 依赖关系地图”工具栏之下显示。
2. 单击此处以重新加载依赖关系，并查看新服务或操作以及相关度量标准。例如，重新加载后，不再显示“可用数据已经更改”消息，且地图中会包括新发现的服务和依赖关系。

重新加载依赖关系地图后，显示的视图是与最后单击的调查器树节点关联的默认视图。例如，如果选择了“Web 服务”节点并重新加载地图，地图会显示在服务的物理视图中。

如果更改内容类型，请查看是否显示“可用数据已经更改”消息并重新加载地图，地图将重新加载服务的物理视图，因为这是 Web 服务树节点的默认视图。如果在调查器树中单击其他节点（如代理节点），然后重新加载地图，则地图会显示代理的物理视图。

过时地图节点的老化和删除

企业管理器跟踪每个发现的依赖关系的期限，并定期重新发现依赖关系来确保它们仍然存在。如果在一段时间内未重新发现某一特定代理、服务或操作的地图节点，则说明该项目已过期并已从依赖关系地图中将其删除。如果稍后已删除的代理、服务或操作再次被调用，企业管理器会再次将该项目添加到依赖关系地图中。

默认情况下，企业管理器会每 6 个小时检查一次依赖关系的最近发现清苦康，并删除 60 天未发现的依赖关系。

通过在 *IntroscopeEnterpriseManager.properties* 文件中设置属性值，可配置企业管理器检查发现的依赖关系期限的频率，以及依赖关系未被重新发现后在过期前应保留在地图中的时间。有关配置企业管理器的属性的详细信息，请参阅[配置企业管理器属性 \(p. 295\)](#)。

如果依赖关系视图显示不完整数据应如何处理

在大多数情况下，服务和依赖关系的初始发现需要一段时间才能完成，因为代理必须先收集后才能将信息传送至企业管理器。初始发现后，不会始终立即显示对服务和依赖关系所做的更改。通常，如果运行的是独立企业管理器，则仅需几分钟即可显示更改。如果运行的是群集中的企业管理器，则所需时间可能会稍长些。

但是，您还应记住，代理要求执行服务或积极运行服务，才能收集所有节点和依赖关系的数据。未被调用的服务和操作在被调用之前不会显示在视图上。

如果依赖关系视图显示任何数据或显示不完全数据，则可能是因为：

- 您正在监控的应用程序服务器或服务没有积极运行，或尚未被调用。确认服务器正在运行，且服务可以调用。
- 非群集（无 MOM）环境中正在运行的代理已经断开与企业管理器的连接。- 刷新代理缓存，强制代理重新发现先前发现的依赖关系，从而将信息发送给新的或重新启动的企业管理器。
- 代理缓存刷新已删除所有的服务和依赖关系数据。代理缓存刷新之后，需要一些时间来收集信息并将信息发送给企业管理器。
- 代理配置文件中已禁用依赖关系视图所需的视图跟踪器。确认没有将 *IntroscopeAgent.profile* 文件中的 `com.wily.introscope.agent.transactiontrace.boundaryTracing.enable` 属性设置为 `False`。
- 更改了代理的名称，导致没有关联视图节点的数据。代理需要一些时间来收集为新代理视图节点显示的数据。
- 缺少的服务在代理未监控的应用程序服务器上运行。
- 缺少的 Web 服务在不受支持的平台或不受支持的 SOAP 引擎上实施。

关于逻辑等效规则

CA APM for SOA 为确定服务和操作的逻辑等效提供三个启发式。可以根据环境需要启用或禁用这些启发式。默认情况下，CA APM for SOA 使用以下规则来确定客户端和服务端等效性：

- 对于某个客户端服务，如果该服务依赖于多个服务器端服务，则所有这些服务器端服务都必须是逻辑等效的。
- 对于某个服务器端服务，如果该服务依赖于多个客户端 Web 服务，则所有这些客户端 Web 服务都必须是逻辑等效的。
- 对于名称匹配，当删除代理说明符后两个物理服务操作具有相同的度量标准路径时，这些操作被视为逻辑等效。

关于用于配置逻辑等效启发式的属性的信息，请参阅[配置企业管理器属性](#) (p. 295)。

了解 SOA 依赖关系地图中的上下文

您在 SOA 依赖关系地图中看到的内容取决于如下顺序的两个因素：

- 您选择的 SOA 依赖关系地图内容类型。
- 您要查看其信息的 调查器树节点。

内容类型和调查器树节点确定了 SOA 依赖关系地图上下文。选择不同数据时，这个上下文会发生更改。

了解内容类型如何影响查看的内容

确定 SOA 依赖关系地图所显示内容的第一个因素是 SOA 依赖关系地图内容类型。您可以选择的内容类型是：

- 代理
- 服务
- 操作

例如，选择“代理”，您可以查看代理之间的高级别依赖关系；选择“操作”，您可以查看操作之间的低级别依赖关系。通过选择适当的内容类型，您可以深入查看依赖关系的特定子集，同时仍可知晓目前在整体 SOA 拓扑结构中的位置。

更改内容类型而不更改选定的节点

如果您选择“服务”，然后在调查器树中单击代理节点，SOA 依赖关系地图将显示与代理相关的所有服务和服务级依赖关系。

如果您保留在同一调查器树节点上并选择另一个内容类型（例如“代理”），SOA 依赖关系地图将显示与选定代理节点相关的所有代理和依赖关系。例如，如果代理 Tomcat01 上的服务依赖于代理 WebLogic02 上正在运行的服务，则 SOA 依赖关系地图显示 Tomcat01 代理图标，该图标有一个指向 WebLogic02 代理图标的依赖关系箭头，表示代理级依赖关系。-

将内容类型更改应用于地图中的所有对象

如果正在显示的内容类型是“代理”，且依赖关系地图显示两个代理（例如 Tomcat01 代理和 WebLogic02 代理）之间的依赖关系，将内容类型更改为“服务”时将显示 Tomcat01 和 WebLogic02 代理上的所有服务级依赖关系。只要更改内容类型，更改都会应用于 SOA 依赖关系地图上当前显示的一切内容。在这种情况下，从“代理”更改为“服务”时会设置上下文，显示地图中当前显示的代理上的所有服务。

了解调查器节点如何影响所示内容

确定 SOA 依赖关系地图显示什么内容的第二个因素是选择的调查器树节点。尽管内容类型决定了所显示依赖关系的类型，但决定地图中所含信息范围的是调查器节点。顶层地图节点和其下的依赖关系总是基于当前选定的调查器树节点。如果选择不同的调查器树节点，则关联的 SOA 依赖关系地图的显示可能完全不同。

然而，还可以使用为当前选定调查器树节点显示的地图节点来导航至新的调查器树节点。例如，可以使用 SOA 依赖关系地图来扩展广泛的服务网络，然后单击地图节点来跳转到新的调查器树节点，并从该节点重绘地图。这允许通过一次单击，将焦点置于一个特定的感兴趣的调查器或 SOA 依赖关系地图节点，消除不感兴趣的网络分支带来的混乱。

显示 SOA 依赖关系视图

通过选择调查器树节点，然后单击“SOA 依赖关系视图”选项卡，可以显示 SOA 依赖关系视图。如果已为代理启用 CA APM for SOA，则可以在调查器中选择代理或任何“Web 服务”节点以显示 SOA 依赖关系视图。如果监控其他任何 SOA 平台，如 CA APM for Oracle Service Bus 或 TIBCO BusinessWorks，在选择与扩展相关的调查器树节点时，也可能能够显示 SOA 依赖关系视图。例如，如果正在监控 Oracle Service Bus，则可以在调查器树中选择代理服务节点或特定代理服务名称，显示所有代理服务或特定代理服务的依赖关系视图。

显示 SOA 依赖关系视图：

- 在调查器树中选择适当的节点。
- 在“查看器”窗格中单击“SOA 依存关系映射”选项卡。

SOA 依赖关系视图显示一个基于选择的调查器节点的默认视图。例如：

- 对于代理节点，SOA 依赖关系视图显示代理级依赖关系的物理视图。
- 对于“Web 服务”、“客户端”或“服务器”节点，SOA 依赖关系视图显示服务级依赖关系的物理视图。
- 对于 *<web_service_namespace>* 节点，SOA 依赖关系视图显示服务级依赖关系的物理视图。
- 对于 *<operation_name>* 节点，SOA 依赖关系视图显示操作级依赖关系的物理视图。

例如，如果在调查器树中选择“Web 服务”>“服务器”，然后单击“SOA 依赖关系视图”选项卡，视图将显示所有服务器端服务和其他服务上一个下游级别的依赖关系。

关于调查器树节点和地图节点

在调查器树中选择一个节点后，该节点是 SOA 依赖关系地图的起点。SOA 依赖关系地图上仅显示起源于选定节点的依赖关系。

SOA 依赖关系地图上的每一项都被认为是一个地图节点。显示的地图节点取决于选择的内容类型。例如，有表示代理、客户端或服务器端服务或单个操作的地图节点。

关于独立地图节点和依赖地图节点

如果代理、服务或操作没有任何依赖关系，它显示为一个独立地图节点。根据设置的上下文，SOA 依赖关系地图可能只显示独立地图节点、独立地图节点和依赖地图节点的组合或仅显示依赖地图节点。例如，如果选择“服务”作为内容类型，且在调查器树中选择“客户端”或“服务器”节点，则可能看到在 SOA 依赖关系地图中显示与选定节点相关的独立服务和依赖服务。

如果在未更改调查器节点的情况下将内容类型更改为“代理”，如果没有代理级依赖关系，则只能看到单个独立代理地图节点。

然而，如果将内容类型更改为“操作”，地图将显示与选定调查器树节点相关联的所有依赖操作中一个级别的依赖关系。依赖操作可能包括客户端操作或服务器端操作。-

关于操作的地图节点

如果将内容类型设置为“操作”，或右键单击一个服务并选择“显示所有操作”，SOA 依赖关系地图将使用下列约定显示操作级依赖关系：

- 客户端服务命名空间显示为绿色框中的地图节点。
- 服务器端服务命名空间显示为蓝色框中的地图节点。
- 代理名称显示为灰色框中的地图节点。

这些颜色框的约定用于帮助识别特定操作所属的服务和代理。

选择物理或逻辑视图

SOA 依赖关系视图提供在您的环境中显示 SOA 组件的两种方式：

- 物理视图是基于服务和操作的物理位置的一种表示，由代理来检测。
- 逻辑视图是基于类似命名服务和操作的分组的一种表示，不考虑其物理位置。

要说明物理视图和逻辑视图之间的差异，以及它们之间的转换如何帮助解决问题，请考虑以下示例。

在分析服务问题时，您将某一特定服务识别为根本原因，但是发现识别的服务是一个分布式服务。负载平衡器将服务请求分发给五个单独的应用程序服务器。

查找该情景中问题的源时需要了解：

- 逻辑实例之间的关系，以确定一个服务调用另一个服务的时间。
- 物理实例之间的关系，以确定实际被调用的物理实例（服务 A 的哪个实例正在调用服务 B 的哪个实例）。
- 物理实例之间的性能如何变化。
- 在其实例之间如何分发服务流量。

例如，如果一个逻辑服务正在经历每分钟 100 次调用，但是这个逻辑服务由五个实例组成，那么对您来说，关键是了解这 100 次调用在每个实例上如何中止。如果发现 100 次调用中有 90 次都转至某一个实例，您可以将问题隔离到该实例，解决问题，然后进一步地分析该实例上负载增加的原因。

SOA 依赖关系视图允许使用“显示”下拉列表在物理视图和逻辑视图之间切换。这使您能够区分逻辑服务实例和物理服务实例，以确定特定物理实例中的问题如何影响逻辑服务的整体性能。

要选择 SOA 依赖关系视图物理或逻辑视图：

- 在 SOA 依赖关系视图工具栏中单击“显示”下拉列表，然后选择“物理”或“逻辑”。

SOA 依赖关系视图基于选择的视图刷新和显示 SOA 环境。例如，如果您一直在物理视图中查看操作，然后选择逻辑视图，则 SOA 依赖关系视图将基于 SOA 依赖关系视图的逻辑表示来刷新和显示操作。

默认情况下，SOA 依赖关系视图显示物理视图，除非您在调查器树中选择了虚拟代理节点。对于虚拟代理节点及其子节点，默认情况下显示逻辑视图。

选择内容类型

由于内容类型会影响您可查看的内容，因此可以针对“操作”、“服务”或“代理”显示具有依赖关系的视图。默认内容类型取决于在调查器树中选择的节点。您可以更改内容类型，以便从不同角度深入查看依赖关系。您可以在视图之间移动以帮助诊断问题服务。

遵循这些步骤：

1. 在 SOA 依赖关系视图工具栏中单击内容类型的下拉列表。
2. 选择“代理”、“服务”或“操作”。

注意: 如果显示的是“逻辑”视图，或者已选择“自定义虚拟代理”，则只能选择“操作”或“服务”。

将刷新 SOA 依赖关系视图以显示内容类型的依赖关系。

详细信息：

[了解内容类型如何影响查看的内容 \(p. 70\)](#)

为依存关系映射节点设置主要度量标准

您可以在 SOA 依存关系映射上查看标准 CA Introscope® 和 SOA 特定度量标准。在每个 SOA 依赖关系视图服务和操作节点下，主要度量标准显示您最感兴趣的标准 CA Introscope® 度量标准。此外，将鼠标悬停在任何 SOA 依赖关系视图代理、服务或操作节点上时，工具提示将显示选择查看的度量标准。

SOA 依赖关系视图针对代理和服务显示的度量标准是该代理或服务下所有操作的聚合值，不仅仅是 SOA 依赖关系视图上当前选择或显示的视图节点。例如，如果将鼠标悬停在服务器端服务视图节点上，工具提示将显示选择查看的度量标准。然而，视图节点的度量标准可以包括依赖客户端节点和非受监控客户端的客户端调用，或 SOA 依赖关系视图中没有显示的客户端（如 C++ 客户端）。所有客户端调用都对服务器端度量标准计算起一定作用。

遵循这些步骤:

1. 在 SOA 依赖关系视图工具栏中单击“主要度量标准”下拉列表。
2. 选择要在每个依赖关系视图节点上显示的标准 CA Introscope® 度量标准之一:
 - 平均响应时间
 - 每个时间间隔的响应数
 - 每个时间间隔的错误
 - 并发调用
 - 停顿计数

SOA 依赖关系视图的默认主要度量标准是平均响应时间。

设置主要度量标准后，依存关系映射中的所有映射节点都显示该度量标准的当前度量标准值。

如果将鼠标悬停在视图节点上，工具提示将显示附加信息，如收集数据点数目的最小值、最大值和当前计数。

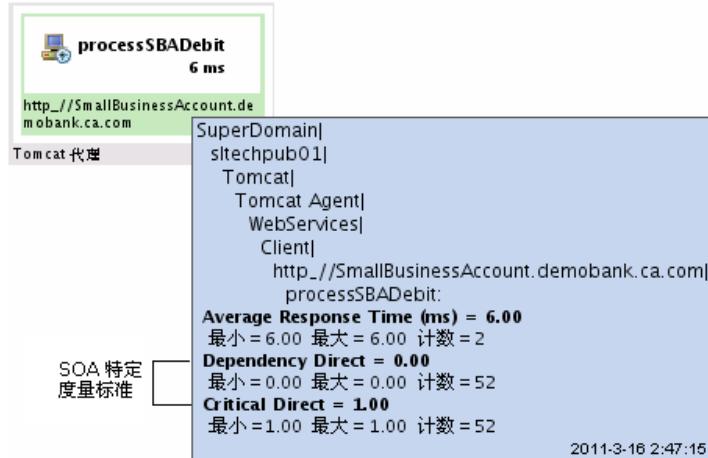
为依赖关系地图节点选择工具提示度量标准

将鼠标悬停在地图节点时 SOA 依赖关系地图会显示工具提示。默认情况下，工具提示显示主要度量标准数据。还可以在工具提示中显示 SOA 特定度量标准。例如，您可以选择在工具提示中显示直接和间接度量标准或偏差度量标准。

要设置 SOA 依赖关系地图工具提示度量标准:

1. 在 SOA 依赖关系地图工具栏中单击“工具提示”按钮以显示“选择工具提示度量标准”对话框。
2. 选择要显示为 SOA 依赖关系地图工具提示的度量标准。例如，您可以选择 SOA 特定度量标准，如依赖关系直接数和关键直接数度量标准。
3. 单击“确定”。

基于选择的调查器节点，将鼠标悬停在 SOA 依赖关系地图节点上时，SOA 依赖关系地图将根据企业管理器报告的关于该调查器树节点的度量标准来显示所有或某些选定的度量标准。例如，如果您设置 SOA 依赖关系地图显示关联的依赖关系直接数工具提示度量标准，且将鼠标悬停在操作节点上，您可以在工具提示中看到此度量标准的数据：



然而，如果将鼠标悬停在服务节点上，将不显示依赖关系直接数度量标准，因为服务不会生成数据来计算依赖关系直接数度量标准。

更改显示的依赖关系级别

在调查器树中选择一个节点并单击“SOA 依赖关系地图”选项卡时，SOA 依赖关系地图会显示选定节点的依赖关系以及一个其他级别的依赖关系（默认情况下）。您可以修改以下列几种方式显示的依赖关系：

- 您可以针对地图中的单项检查其他依赖关系。
- 您可以隐藏图中单项的低级依赖关系。
- 您可以针对地图中的每一项展开或打开另一个全级依赖关系。
- 您可以针对地图中的每一项隐藏或卷起最低级别的依赖关系。

检查地图节点的其他依赖关系

默认情况下，SOA 依赖关系地图中只显示一个级别的依赖关系。如果您对特定地图节点感兴趣，您可以在没有为其他地图节点扩展低级别的依赖关系的情况下，检查该代理、服务或操作的其他低级别的依赖关系。

针对地图的一项检查其他依赖关系时，您一次可打开一个依赖关系级别。每次您检查其他依赖关系时，下一个最低级别的依赖关系都会添加到地图中。如果您检查下一个级别且没有找到其他依赖关系，则地图显示消息：未找到其他依赖关系。

要检查特定地图节点上的其他依赖关系:

- 右键单击地图的单个项，然后从菜单中选择“显示下一依赖关系”。

SOA 依赖关系地图仅针对选定地图节点打开下一个级别的依赖关系。例如，正在展开服务器端操作时，打开的下一级别的依赖关系是关联的客户端操作。

隐藏地图节点的依赖关系

如果已针对特定代理、服务或操作打开多个依赖关系级别，您可能要隐藏该项的一些依赖关系级别。针对地图中的一项隐藏依赖关系时，您一次可卷起一个依赖关系级别。每次您隐藏依赖关系时，将从地图中删除下一个最低级别的依赖关系。

要删除特定地图节点的依赖关系:

- 右键单击图中的单个项目，然后从菜单中选择“隐藏依赖关系”。

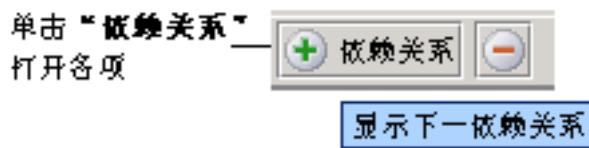
SOA 依赖关系地图针对选定地图节点卷起所有依赖关系级别。

检查图中所有项目的其他依赖关系

默认情况下，SOA 依赖关系地图中只显示一个级别的依赖关系。要检查当前地图中所有项的其他低级别依赖关系，您可以针对整个地图一次打开一个依赖关系级别。每次您检查其他依赖关系时，下一个最低级别的依赖关系都会添加到地图中。如果您检查下一个级别且没有找到其他依赖关系，则地图显示消息：未找到其他依赖关系。

针对地图中的每一项打开依赖关系:

- 单击 SOA 依赖关系地图工具栏上的“显示下一依赖关系”按钮。例如：



SOA 依赖关系地图会针对地图中的每个最低级别项打开依赖关系。因此，如果初始地图节点显示三个依赖关系，单击此按钮时可针对三项中的每一项添加新级别的依赖关系。

隐藏图中所有项目的依赖关系

如果已针对地图中的所有代理、服务或操作打开多个依赖关系，您可能要从整个地图中删除一些依赖关系级别。隐藏图中所有项目的依赖关系时，一次可卷起一个依赖关系级别。每次您隐藏依赖关系时，将从地图中删除下一个最低级别的依赖关系。

隐藏图中每个项目的依赖关系：

- 单击“SOA 依赖关系地图”工具栏上的“隐藏上个依赖关系”按钮。例如：



SOA 依赖关系地图会卷起地图中最低级别的依赖关系。

在 SOA 依赖关系地图内导航

如果您展开 SOA 依赖关系地图以显示多个依赖关系级别，或者您的组织具有特别复杂的 SOA 基础架构，您可能发现 SOA 依赖关系地图变得难以导航或很难找到要查找的特定信息。SOA 依赖关系地图包括几个工具，可帮助您在地图上导航，在地图和调查器之间移动，以及快速显示或隐藏信息。

平移、缩放和适合 SOA 依赖关系地图

如果您正在查看的依赖关系地图太大，您可能要导航到特定部分，或重新调整某些部分以提供更好的可视性。SOA 依赖关系地图提供下列工具在其选项卡之内移动、放大和重新适合 SOA 依赖关系地图：

- 平移
- 放大选择项
- 放大/缩小
- 适合选项卡

要在选项卡视图上移动 SOA 依赖关系地图:

1. 单击“SOA 依赖关系地图”工具栏上的“平移”工具按钮。“平移”工具按钮显示一个手形图标。
2. 在 SOA 依赖关系地图内单击。
3. 在选项卡内拖动地图以集中检查您感兴趣的区域。

要放大 SOA 依赖关系地图的某一区域:

1. 单击“SOA 依赖关系地图”工具栏上的“放大选择项”工具按钮。“放大选择项”按钮在选定区域内显示一个放大镜。
2. 在 SOA 依赖关系地图内单击。
3. 拖动以选择要放大的矩形区域。

缩放 SOA 依赖关系地图:

1. 单击“SOA 依赖关系地图”工具栏上的“放大和缩小”工具按钮。“放大和缩小”按钮显示一个带有向上和向下箭头的放大镜。
2. 在 SOA 依赖关系地图中单击并按住鼠标按钮。
3. 在 SOA 依赖关系地图上，向前滚动鼠标可放大，向后滚动鼠标可缩小。

要调整 SOA 依赖关系地图的大小以适合选项卡:

1. 单击依赖关系地图工具栏上的“适合选项卡”工具按钮。“适合选项卡”按钮会显示一个有网格线的放大镜。
SOA 依赖关系地图会通过调整大小来适合选项卡。
2. 使用“放大选择项”或“放大和缩小”工具可自定义地图的可读性。

从地图节点跳转到关联的调查器树节点

如果要查看物理视图的 SOA 依赖关系地图节点，您可以单击任何地图节点以转至调查器树中的相应组件。跳转到新的调查器树节点时，新的调查器树节点会用作起点来重绘 SOA 依存关系映射。这使您能够基于选择的地图节点位置重新确定自己的位置。

要从地图节点跳转到新的调查器树节点并重绘地图:

1. 验证 SOA 依赖关系地图以物理视图显示。
2. 在 SOA 依赖关系地图中选择地图节点。
3. 右键单击，然后选择“此处在树中跳转”。

新调查器树节点已被选中并成为 SOA 依存关系映射的新焦点。

显示服务的所有操作

如果选择“服务”内容类型，您可以展开 SOA 依赖关系地图来显示特定服务的所有操作。

显示服务的所有操作：

1. 在调查器树的“客户端”或“服务器”节点下选择任何单个服务操作。

2. 单击“SOA 依存关系映射”选项卡。

SOA 依赖关系地图显示相应的客户端或服务器操作及其表示相应服务和代理的框中的第一级依赖关系。

3. 选择服务地图节点，然后右键单击，选择“显示所有操作”。

“SOA 依存关系映射”将显示该客户端或服务器端服务的所有操作。

显示代理的所有服务

如果已选择显示“代理”内容类型，您可以展开 SOA 依赖关系地图以显示特定代理的所有服务。

显示代理的所有服务：

1. 在调查器树中，选择“客户端”或“服务器”节点下的任意单个服务。

2. 单击“SOA 依存关系映射”选项卡。

SOA 依赖关系地图显示相应的客户端或服务器服务及其表示相应代理的一个或多个框中的第一级依赖关系。

3. 在 SOA 依赖关系地图中选择关联的代理，然后右键单击，选择“显示所有服务”。

“SOA 依存关系映射”将显示该代理的所有服务。

保存 SOA 依赖关系地图图像

您可以各种格式保存全部或部分 SOA 依赖关系地图，使您能够：

- 与同事共享信息以帮助您解决问题。
- 将当前地图另存为环境快照，这样您可以稍后返回。

SOA 依赖关系视图共享

解决服务问题时经常需要您与不熟悉 CA Introscope® 或 SOA 环境的同事合作。例如，如果您要响应一个报警，您可能需要与负责不同服务的开发者合作才能解决应用程序问题。将依赖关系视图另存为图像，使您能够与这些同事共享关于服务与当前度量标准之间关系的有价值信息。

您也可以选择保存显示代理级、服务级或操作级依赖关系以及具有依赖关系的物理或逻辑视图的依赖关系视图。通过提供服务依赖关系的图形视图，有助于进行沟通并帮助组织更有效地解决问题。

将“SOA 依赖关系地图”另存为快照

SOA 依赖关系地图总是反映受监控服务之间最近发现的依赖关系。可以定期检查和刷新数据，但并不会因历史目的而保存依存关系信息。

在大多数情况下，SOA 依赖关系地图中的依赖关系是完全静态的，因为多数组织只是偶尔在生产环境中部署、修改或删除应用程序。然而，在某些情况下，您可能发现将实时环境图像另存为依赖关系数据的“快照”是有用的，因为您稍后可以返回来查看这些快照。

要将全部或部分 SOA 依赖关系地图另存为图像：

1. 单击 SOA 依赖关系地图工具栏上的“另存为图像”按钮以显示“导出图像”对话框。“另存为图像”按钮显示一个磁盘图标。
2. 为输出文件选择以下设置。

类型

选择要用于图像的文件格式。您可以将依赖关系地图另存为 JPEG、GIF、PDF、PNG 或 SVG 文件。

文件名

键入输出图像文件的目标路径和文件名。

图像内容

选择一个选项，在图像文件中保存特定部分的 SOA 依赖关系地图。

可视窗口

此选项仅在您选择“当前缩放级别”大小选项时才可用。只有要保存当前窗口中可见的地图部分时才选择此选项。例如，如果 SOA 依赖关系地图包括 100 个节点，但是已经通过放大只查看 10 个节点，则此选项会创建只有这 10 个节点的图像。

绘制网格

未启用该选项。

仅选定的对象

此选项仅在您已选择地图节点后才可用。只要求保存在地图中已选择的对象时才选择选项。例如，如果您选择了客户端 Web 服务地图节点，此选项仅创建该客户端 Web 服务的图像。

图像特征

滑动滑块或在 1 和 100 之间挑选一个数来指定图像质量和文件大小，此处 100 表示最佳图像输出质量。图像质量越高，保存图像的文件大小越大。

大小

选择下列选项之一可控制图像大小：

当前缩放级别

如果您已放大地图的某一区域，则选择此选项。只有要求图像仅显示放大的区域时此选项才与“仅可视窗口”一起使用。

实际尺寸

选择此选项可按实际尺寸保存整个地图。此选项不会缩小地图。

适合画布

选择该选项以保存与窗口大小相符的整个映射。此选项会缩小地图以适合窗口大小（如果需要）。

自定义

选择此选项可以像素为单位选择图像宽度和高度设置。

3. 单击“确定”创建文件。

该文件保存在指定的位置。

从视图节点启动事务跟踪

您可以直接从视图节点启动事务跟踪会话。这允许您从某个服务中关系的高级视图快速导航至正通过该服务的实际事务的详细视图。

例如，假定您在看到“SOA 性能一概览”显示板上的红色报警后试图解决特定服务中的问题。由于您要调查的服务是由关键业务事务使用的，因此对您来说最重要的是尽可能快地查看与该服务相关的事务。

不必手动启动事务跟踪会话和输入筛选条件，您可以通过直接从视图节点启动新的事务跟踪会话来节省 SOA 依赖关系视图上服务的时间。

从视图节点启动事务跟踪会话

1. 右键单击任何视图节点并选择“启动事务跟踪”可启动事务跟踪会话。

从依赖关系视图节点启动事务跟踪可以基于自动选择的视图节点打开具有默认筛选的“新建事务跟踪会话”对话框。

2. 单击“添加”将默认筛选添加到筛选列表中或根据需要修改筛选条件，然后单击“添加”。
3. 单击“设置筛选”，然后单击“确定”可打开“事务跟踪查看器”。“事务跟踪查看器”会收集和显示与涉及选定视图节点的事务相关的所有跟踪。

注意：有关在 SOA 环境中使用跨进程事务跟踪以及使用筛选的详细信息，请参阅“[在 SOA 环境中使用事务跟踪 \(p. 85\)](#)”。有关使用新事务跟踪会话和事务跟踪查看器的详细信息，请参阅《CA APM Workstation 用户指南》。

群集的 SOA 依赖关系视图

如果您有群集的 CA Introscope® 环境，SOA 依赖关系视图将显示服务和向 MOM 报告的收集器的已发现依赖关系。例如，如果具有配备三个收集器企业管理器的环境，您可以查看在各个收集器上运行的代理或服务之间的依赖关系。依存关系的显示方式与应用程序服务器的相应显示方式相同，都是基于内容类型和调查器树节点。

SOA 依赖关系视图数据仅保存在收集器上。MOM 从收集器中接收其数据。MOM 不保存任何 [SOA 依赖关系视图数据 \(p. 65\)](#)。

显示 SOA 依赖关系视图要求在 MOM 企业管理器和群集中的每个收集器
企业管理器上安装下列文件：

`com.wily.introscope.soa.dependencymap.common_<version>.jar`
`com.wily.introscope.soa.dependencymap_<version>.jar`

这些文件安装在群集中所有企业管理器上的 <企业管理器主目
录>/product/enterprisemanager/plugins 目录中。如果找不到这些文件，
SOA 依赖关系视图将无法正确显示。

第 5 章： 在 SOA 环境中使用事务跟踪

通过收集关于性能、操作和服务相关组件总体运行状况的度量标准，CA APM for SOA 使您能够监控服务和业务流程。显示板或特定度量标准建议可能存在服务问题时，您可以使用事务跟踪来查看实际业务事务的详细信息，帮助找到原因和解决问题。

此部分包含以下主题：

- [关于跨进程事务跟踪 \(p. 85\)](#)
- [使用事务跟踪解决问题 \(p. 87\)](#)
- [启动并查看事务跟踪 \(p. 87\)](#)
- [为事务跟踪设置筛选 \(p. 93\)](#)
- [查询事件数据库以查找 SOA 服务 \(p. 95\)](#)

关于跨进程事务跟踪

在 SOA 环境中，事务在多个 Java 虚拟机 (JVM) 或公共语言运行时 (CLR) 实例上运行是很常见的。处理从一个服务传递给另一个服务。因此，要查看事务的完整路径，需要能够跟踪 JVM 和 CLR 边界上的同步和异步调用。还需要能够跟踪运行 Java 或 .NET 代理的多个平台上的事务。

启用 CA APM for SOA 后，您可以跟踪使用 HTTP、HTTPS 或 JMS 传输协议且在支持平台上运行的任何业务事务的完整路径。完整事务可能包括在许多不同平台上处理的区段。例如，您可以跟踪一个使用服务的事务，该服务在 WebLogic 上运行，然后被传递给在 SAP NetWeaver 或 .NET 服务器上运行的服务。根据您启用的其他 SOA 扩展，一个事务也可以包括涉及以下任一组件的区段：

- Oracle Service Bus
- WebSphere Process Server
- WebSphere Enterprise Service Bus
- TIBCO BusinessWorks
- webMethods Integration Server
- WebSphere MQ
- Spring Web Service。

代理中的 Spring Web Service 支持可向客户端和服务器提供 Web 服务内部度量标准和 Web 服务传输度量标准。

只要为需要跟踪的每个节点启用 CA APM for SOA，业务事务即可包括平台的任何组合。通过为代理启用 CA APM for SOA，您可以深入查看在不同节点上运行的多个 JVM 或 CLR 实例上调用服务的事务的详细信息。

启用 CA APM for SOA 后，不必再进行其他任何配置。

注意：有关支持的信息，请参阅《兼容性指南》中的“SOA 性能管理”一节。

了解如何链接事务段

通常，事务由一系列调用和响应组成，这些调用和响应从一个流程传递到另一个流程。通常不同的事务段在不同的逻辑或物理服务器上运行，或可分发到不同的组件或后端系统。

因此，装配完整事务需要识别哪个处理段属于同一事务，以及事务中的一个流程何时调用另一个流程。要跟踪调用流程(可在不同 JVM 或 CLR 实例上运行)的事务的完整路径，代理将关联标识符添加到事务中。可以将关联标识符从一个流程传递给另一个流程以识别属于同一事务的段。

除识别属于同一事务的组件之外，还有提供排序信息以跟踪调用事务不同部分的顺序的关联标识符。排序信息使您能够查看调用事务段的顺序。对于同步事务，该顺序可帮助您识别事务段之间调用与被调用关系。对于异步事务，该顺序可帮助您针对复杂客户端和服务器事务段识别跨多个流程的处理工作流。

关联标识符数据集自动由代理管理并发送到企业管理器。然后，该信息用于构造“事务跟踪查看器”中显示的选定事务的图形表示。

了解跨进程事务跟踪的上下文

您可以基于服务命名空间或操作名称启动事务跟踪会话。在大多数情况下，命名空间提供适当信息来帮助您识别有问题的服务。如果您已确定某个特定操作可能是问题源，您可能要针对该操作启动事务跟踪。

然而，如果您使用操作名称，您应该记住，服务命名空间会提供操作的上下文。例如，操作名称 *makeReservation* 可用于航空公司预定机票、餐厅预定座位或医院预约时间的应用程序中。如果启动针对操作名称 *makeReservation* 的事务跟踪，您需要了解使用该操作的命名空间。

选择某个特定跟踪时，操作的命名空间信息显示在“跟踪视图”选项卡的“组件详细信息”中。例如，如果选择“跟踪视图”选项卡上的 *makeReservation* 操作，可通过检查“组件详细信息”来检查与该操作关联的命名空间，以了解该操作属于旅行社 Web 服务还是酒店预订服务。

使用事务跟踪解决问题

要了解跨进程事务跟踪如何帮助您快速有效地识别和评估问题，请考虑以下示例。运行事务跟踪会话后，应用程序支持找到一个执行时间为 6 秒（6000 毫秒）的事务。

在事务的“跟踪视图”中可以明显看出，事务包括从客户端 Web 服务 *dataservice.yourcompany.net/Invoke* 到服务器端 Web 服务 *cics.mycompany.net/Invoke* 的调用，并且服务器端 Web 服务正在向 CICS 主机进行大量的调用。--

虽然 CICS 处理时间不会显式显示在跟踪中（由于该部分事务未检测），但是“跟踪视图”显示了 CICS 后端正在快速连续地处理重复请求。- 在该事务跟踪中，应用程序支持专家会发现，重复的调用大多情况下是由服务器端服务中的程序设计逻辑（如嵌套循环）而引起的，并且 *调用* 操作花费的时间占事务总执行时间的大部分。了解此信息后，应用程序支持专家可以直接联系服务器端 Web 服务的所有者或开发者，请求对调用 CICS 后端的应用程序逻辑进行深入调查。

启动并查看事务跟踪

业务事务中包含对本地或远程计算机上不同 JVM 或 CLR 实例的调用时，可以使用跨进程事务跟踪来识别和解决 Web 服务的问题。但是，当企业管理器不是群集的一部分时，参与事务的 JVM 或 CLR 上的代理必须将信息报告给同一企业管理器，否则您将无法看到关联跟踪。如果代理将信息报告给同一企业管理器，则您可以启动一个事务跟踪会话来捕获有关整个业务流程的信息。在 Workstation 连接到 MOM 的群集环境中，参与跨进程事务的代理可以连接到任何收集器以及相关跟踪。

您可以以下述方式启动事务跟踪会话：

- 直接通过 SOA 依赖关系视图中的视图节点。
- 通过单击“Workstation”>“新建事务跟踪会话”，从 Workstation 手工启动。

遵循这些步骤：

1. 右键单击“SOA 依赖关系视图”中的视图节点或单击“Workstation”>“新建事务跟踪会话”以显示“新建事务跟踪会话”对话框。
2. 单击第四个复选框以激活与 SOA 相关的筛选选项。

以下示例说明了激活 SOA 筛选要选择的复选框和命名空间筛选定义：



3. 从与 SOA 相关的筛选选项列表中选择要使用的事务跟踪筛选类型，然后单击“添加”。

CA APM for SOA 包括特定于 SOA 的事务跟踪筛选，通过这些筛选，可以基于服务命名空间或某个特定操作名称来选择事务。系统会自动设置默认的筛选和值，具体取决于启动新事务跟踪之前选定的节点。

可以选择命名空间或操作名称筛选以跟踪包含与您指定的条件相匹配的任何 Web 服务或 Web 服务操作的事务。此条件可以包括 SOA 平台提供的 Web 服务，如 TIBCO BusinessWorks 或 webMethods Integration Server。

4. 单击“设置筛选”。
5. 在“运行会话时间”字段中，输入运行事务跟踪会话的分钟数。
6. 在“跟踪代理”部分，选择要需进行事务跟踪的所有代理或特定代理。
7. 单击“确定”。

此时会启动事务跟踪会话。

注意：有关从 Workstation 启动事务跟踪的详细信息，请参阅《CA APM Workstation 用户指南》。

详细信息：

[从视图节点启动事务跟踪 \(p. 83\)](#)

使用摘要视图

启动事务跟踪会话后，事务跟踪结果将显示在事务跟踪查看器的顶部。该列表显示您正在跟踪的命名空间或操作的所有事务。您可以单击“持续时间”列标题，按持续时间对事务进行排序。

在查看器顶部选择一个事务，以便在“摘要视图”选项卡上显示构成此事务的所有调用，包括对其他 JVM 或 CLR 的调用。

选择某个事务后，“摘要视图”将列出所选事务中调用的服务和操作，包括调用次数、调用时长（以毫秒为单位）以及最短、平均及最长调用时间。

注意：有关使用事务跟踪的摘要视图的详细信息，请参阅《CA APM Workstation 用户指南》。

使用跟踪视图

要以图形形式查看事务的详细信息，请单击“跟踪视图”选项卡。

事务跟踪视图将事务中的每个组件显示为一个条，并且显示每个组件在事务执行总时间中所占的时长。对于同步事务，“跟踪视图”选项卡还指出正在查看的事务段涉及的各节点上的组件之间的调用关联关系。

如果事务包括处理多个 JVM，则每个 JVM 的跟踪将显示在单独的行中。对于 SOA 事务，处理多个 JVM 很常见。

在图表中选择事务组件后，有关组件的其他详细信息将显示在图下方的“组件详细信息”中。

注意：有关在事务跟踪中使用“跟踪视图”或“组件详细信息”所显示信息的详细信息，请参阅《CA APM Workstation 用户指南》。

使用树视图

要查看选定事务的组件的层级视图，请单击“树视图”选项卡。

“树视图”以层次结构显示事务组件，并根据组件对事务总体执行时间的贡献时间显示红色、黄色或绿色指示器。

您可以选择并展开组件，以查看更多详细信息。如果选择任一组件，则选项卡将在“组件详细信息”窗格中显示其他信息。

注意：有关在事务跟踪中使用“树视图”或“组件详细信息”所显示信息的详细信息，请参阅《CA APM Workstation 用户指南》。

使用顺序视图

由于 SOA 环境中分布式应用程序的复杂性，对于单个用户事务来说，跨越在单独代理 JVM 或 CLR 上运行的多个线程和同时包含同步及异步调用是很普遍的。需要计入这些单个事务段，以提供事务的完整路径作为逻辑单元。“顺序视图”选项卡显示事务段之间的调用方与被调用方关联关系，以便直观显示调用的先后顺序。

“顺序视图”对于以下事务特别有用：

- 包括异步调用
- 包括对互相时间不同步的代理上运行的流程的调用
- 包括跨多个 JVM 或 CLR 的复杂同步调用
- 包括对下游代理自动启动的跟踪，这些代理是从其他代理上的事务跟踪会话中调用的。

要显示选定事务中流程的调用顺序，请单击“顺序视图”选项卡。在该选项卡中，各箭头指向调用的流程，条形图按持续时间排序列出了各流程。如果存在确定为事务一部分的跟踪，但无法确定其正确位置，则用虚线将其添加到顺序中。如果在图表或条形图中选择某个流程，该流程的其他详细信息将显示在右侧面板中。

最慢流程可通过其在条形图中的排名和慢流程图标来识别。如果流程执行中存在任何错误，将为该流程显示错误图标。

选择标记流程节点的方法

“顺序视图”中列出的每个流程都表示一个针对完成某项操作所涉及流程的组件跟踪。如果跟踪来自同一服务器或代理，则可能很难或无法识别针对流程显示的标签。

默认情况下，将使用包含操作名称的组件、包含 EJB 字符串的组件或者前置有代理名称的第一个唯一组件资源名称来自动选择流程标签。如果不希望使用默认选择，则可以更改流程标签，以便使用第一个组件名、最后一个组件名、线程名称或说明。

如果选择最后一个组件名作为流程标签，则所用名称是调用堆栈中最后一个明确名称。根据调用顺序以及是否存在同步或异步调用，该名称可能是流程的最后一个组件，也可能不是。

要选择不同的流程标记方案，请执行以下操作：

- 选择“顺序视图”选项卡上的“流程标签”选项。

选择计算持续时间的方法

计算“顺序视图”选项卡上跟踪的持续时间有两种方法：

- **完全持续时间**利用流程的开始和结束时间进行计算。该方法类似于在“跟踪视图”选项卡上显示持续时间的方式。使用该方法时，根跟踪始终是同步事务中最慢的。这是默认计算。
- 通过从总体持续时间中减去跟踪的异步子项的持续时间，可计算出**净持续时间**。该方法的主要优势在于同步事务的最慢流程是显而易见的。从完全持续时间切换到单纯持续时间将改变按已分级持续时间列出的流程的顺序，并且持续时间将根据计算执行时间的方式发生变化。

要更改计算持续时间的方式，请执行以下操作：

- 在“顺序视图”选项卡上选择“完全”或“单纯”作为“持续时间类型”。

在“顺序视图”和“跟踪视图”之间导航

多数情况下，您会发现在“顺序视图”和“跟踪视图”之间导航非常方便。通过右击菜单选项，您可以在这些视图间快速移动，而不会丢失特定流程或组件的上下文。

要从“顺序视图”中的某个流程跳转到“跟踪视图”，请执行以下操作：

1. 在图表中选择某个流程。
2. 右键单击，然后在“跟踪视图”中单击“选择线程”。

从跟踪视图中的组件跳到顺序视图：

1. 选择图表中的某个组件。
2. 右键单击，然后在顺序视图中单击“选择线程”。

查看事务中流程的详细信息

如果在图表或条形图中选择某个流程，可以在右侧面板中查看有关该流程的详细信息。根据选定的流程，可以获取以下部分或全部信息：

线程名称

调用堆栈中选定线程的名称。

线程组名称

线程组的名称。

应用程序名称

用于应用程序的显示名称。

类

已检测的类。

方法

已检测的方法。

方法描述符

已检测的方法参数或签名。

资源名称

应用程序资源的类型。例如，资源类型可以识别 servlet 或 EJB。

URL

用于标识应用程序前端的完整的统一资源定位器 (URL)。

规范化 URL

在代理配置文件中定义为标识应用程序前端的方法的统一资源定位器 (URL) 的规范化部分。

参考 URL

参考页的统一资源定位器 (URL)。

服务器名称

执行流程的服务器的名称。

服务器端口

执行流程的端口号。

方案

使用的统一资源标识符 (URI) 方案的类型。

URI 方案是统一资源标识符命名结构的顶层，由方案名称加冒号 (:) 构成。

例如，使用超文本传输协议访问的 Web 服务的 URI 方案可能是 http。

HTTP 方法

使用的 HTTP 方法类型。例如，最常用的 HTTP 方法是 POST 和 GET。

CrossProcessData

用于标识属于同一事务的流程的唯一关联标识符。

跟踪类型

流程的事务跟踪类型。跟踪类型指出该流程是事务跟踪会话的一部分，还是在抽样事务中收集的。

为事务跟踪设置筛选

启用“CA APM for SOA”会将命名空间筛选和操作名称筛选添加到 Workstation 中，以帮助快速分离您所关心的事务。例如，如果关键业务事务中使用了某个特定服务，则命名空间筛选将允许您筛选所有传入事务，以查找具有该特定命名空间的事务。确定发生问题的命名空间之后，您可以使用操作名称筛选来筛选事务以查找一个或多个特定操作，以帮助分类该问题。

您还可以将命名空间和操作名称筛选与其他筛选结合使用，以加强对返回的事务结果的控制。例如，您可以将命名空间筛选与“用户 ID”或“错误”筛选结合使用，以便仅捕获同时满足这两个条件的事务。

遵循这些步骤：

1. 单击“Workstation”>“新建事务跟踪会话”。
可用的具体筛选取决于您已安装的组件。
2. 单击复选框以激活某个特定筛选。
例如，单击“持续时间长于”复选框，以查找持续时间长于指定时间的事务。您可以使用特定参数筛选（例如用户 ID 或会话 ID）将事务限制在您指定的条件内。
3. 单击复选框，以激活 CA APM for SOA 和其他 SOA 平台的筛选。
4. 选择筛选类型，筛选条件，键入值，然后选择是否区分大小写。单击“添加”将该筛选添加到可用筛选列表中。

命名空间和操作名称选项是 Web 服务的标准筛选。这些筛选还适用于通过 SOA 平台（如 Oracle Service Bus、TIBCO BusinessWorks 或 webMethods Integration Server）公开的 Web 服务。

但是，如果您正在监控其他 SOA 平台，则可使用以下的附加筛选类型来捕获事务：

适配器节点

WebSphere Process Server 出站适配器

业务流程

TIBCO BusinessWorks 流程实例

webMethods Integration Server 业务服务

WebSphere Process Server 业务流程

业务服务

Oracle Service Bus 业务服务

业务状态机

WebSphere Process Server 业务状态计算机

中介流

WebSphere Enterprise Service Bus 中介流

中介流操作

WebSphere Enterprise Service Bus 中介流操作

消息服务器

IBM WebSphere MQ 消息服务器

代理服务

Oracle Service Bus 代理服务

如果手工键入命名空间或操作名称，请使用名称的正确拼写，以使筛选找到匹配的事务。

5. 如有需要，请向筛选列表中添加其他筛选，并且组合这些筛选以[构建复杂的筛选 \(p. 95\)](#)。例如，您可以为命名空间添加一个筛选，然后为操作名称添加另一个筛选。在定义的筛选列表中选择全部两个筛选，然后单击“AND”同时在您指定的命名空间和操作名称上筛选事务。
6. 选择您希望用于该会话的筛选，然后单击“设置筛选”。
先前设置的所有筛选都将被替换。每次仅一个筛选有效。
7. 在“运行会话时间”字段中，输入运行事务跟踪会话的分钟数。
8. 在“跟踪代理”部分，选择要需进行事务跟踪的所有代理或特定代理。
9. 单击“确定”。

此时会启动事务跟踪会话。

添加并保存筛选

通过定义筛选并单击“添加”，您可以添加所需任意数量的不同筛选。单击“添加”时，将把新筛选添加到可用筛选的列表中。筛选不允许重复。

要指定希望在任何事务跟踪会话中使用的筛选，必须在可用筛选列表中选择筛选定义，然后单击“设置筛选”。

单击“确定”以启动事务跟踪会话时，将存储已定义的筛选，以供下次启动新事务跟踪会话时使用。单击“确定”将自动按用户存储筛选。下次登录工作站时，您可以使用自己先前定义的一套筛选来选择筛选以进行跟踪。

删除筛选

通过在可用筛选列表中选择某个筛选并单击“删除”，可将其删除。单击“删除”将从“新建事务跟踪会话”对话框的可用筛选列表中删除选定的筛选。必须单击“确定”才能永久删除选定的筛选。如果不单击“确定”，则删除的筛选将在下次启动“新建事务跟踪会话”时还原到可用筛选列表。

仅当某个筛选未用作其他筛选的组件时，才可将其删除。例如，如果将 *namespace filter1* 和 *namespace filter2* 组合起来创建筛选 *namespace filter1 AND namespace filter2*，则在删除 AND 筛选之前无法删除 *namespace filter1*。

使用复杂筛选

通过使用 AND 或 OR 运算符组合筛选列表中的一个或多个筛选，可以创建复合筛选。已定义的筛选列表中必须已经存在您组合的筛选。用 AND 或 OR 运算符将两个简单的筛选语句连起来，就形成一个新筛选，并且该筛选随后可以单独用作筛选，或者作为创建其他筛选的构建块。

使用 AND 运算符意味着跟踪必须通过筛选的所有组件。例如，筛选 *namespace contains demobank AND operationname starts with process* 要求传入跟踪需同时满足这两个筛选条件。

使用 OR 运算符意味着跟踪必须通过至少一个筛选组件。例如，筛选 *namespace contains demobank OR namespace contains creditcheck* 要求传入跟踪需满足第一个筛选条件和/或第二个筛选条件。

查询事件数据库以查找 SOA 服务

事务跟踪查看器主要用来查看事务跟踪会话中的实时数据或最新数据。事务跟踪会话的结果将自动存储在事务事件数据库中，以供使用历史查询查看器进行查看。如果稍后需访问该信息，您可以查询事务事件数据库，以便显示与 SOA 相关服务有关的所有业务事务或错误方面的信息。

要在数据库中查询与 Web 服务有关的所有事务，请执行以下操作：

1. 打开工作站后，选择“工作站”>“查询历史事件”。
2. 在“查询”字段中，输入查询信息以检索适当信息。例如：
 - 要查询包含 Web 服务的任何事件，请输入 *webservices*
 - 要查询与特定 Web 服务端点有关的任何事件，请输入所需 URL
3. 如有必要，请选择查询的时间范围。
4. 单击“执行”以显示包含 Web 服务信息的所有业务事务或错误。
5. 单击表中的任意行，在底部窗格中查看其详细信息。

例如，如果选择的行是事务跟踪 (T)，则可以像在事务跟踪查看器中一样，单击“摘要视图”、“跟踪视图”、“树视图”和“顺序视图”选项卡来查看其他信息。

查看错误事件信息

如果历史事件是一个错误 (E)，则底部窗格将显示“堆栈视图”选项卡。如果事件是一个事务，则底部窗格将显示“摘要视图”、“跟踪视图”、“树视图”和“顺序视图”选项卡。您可以浏览显示的信息，以确定引起问题是 Web 服务中的哪个组件。

查询数据库中与 Web 服务相关的错误：

1. 选择“工作站”>“查询历史事件”。
2. 在“查询”字段中，输入：
type:errorsnapshot AND webservices
3. 单击表中的某一行，在“堆栈视图”选项卡中查看错误详细信息。
 - 对于服务器端的错误，可在红色错误消息中看到错误的根本原因。
 - 在客户端和服务器端上，可看到根本原因和 SOAP 错误异常。
 - SOAP 错误的错误消息在红色错误消息的开头有 SOAP 错误前缀。

查看相关联的事件信息

如果您在查看有关事务的详细信息，则可检查属于同一事务、但在事务期间调用的其他流程或节点上运行的事件。

要查看事务跟踪的相关事件，请执行以下操作：

1. 从事务跟踪查看器或历史查询查看器内显示的列表中选择一个事务跟踪，然后单击“跟踪视图”选项卡。
2. 选择“跟踪”>“相关的事件”，以显示属于选定事务跟踪的其他流程或节点上的跟踪列表。

随后可选择任一相关跟踪，以显示该跟踪的“摘要视图”、“跟踪视图”、“树视图”或“顺序视图”。

第6章：配置特定于 SOA 的属性

本节介绍了使用配置属性自定义 CA APM for SOA 的方法。

此部分包含以下主题：

[配置 Web 服务的显示名称 \(p. 99\)](#)

[配置关联跟踪 \(p. 101\)](#)

[配置 SOAP 处理程序的插入点 \(p. 103\)](#)

[配置 SOA 依赖关系地图的限制 \(p. 104\)](#)

配置 Web 服务的显示名称

默认情况下，使用 Web 服务命名空间对 Web 服务节点进行命名。Web 服务命名空间类似于其 URL。例如，如果某个 Web 服务使用以下 URL：

`http://ClearingHouse.demobank.ca.com`

默认情况下，其节点如下所示：

`http://ClearingHouse.demobank.ca.com`

您也可以将代理配置为使用 Web 服务端点作为节点名称。Web 服务端点包含其他信息，例如服务的服务器名称和端口号。例如，如果选择显示 `ClearingHouse.demobank.ca.com` Web 服务的 Web 服务端点，则其节点在调查器中如下所示：

`http_localhost_8383_demobank_services_ClearingHouseService`

通过编辑代理的 `webservices.pbd` 文件并指定希望使用 `{namespace}` 还是 `{servicename}`，可以更改显示的名称。大多数情况下，命名空间是调查器和 SOA 依赖关系地图中显示的最易识别的名称。但如果编辑 `webservices.pbd` 文件和 `IntroscopeAgent.profile` 文件，则可使用服务端点。

要在调查器和依赖关系地图中将服务端点用作节点名称，请执行以下操作：

1. 打开 `<Agent_Home>` 目录中的 `webservices.pbd` 文件。
2. 搜索所有 `{namespace}` 实例并用 `{servicename}` 将其替换。

因为它作为完全限定的度量标准名称的一部分包含在跟踪器中，因此在文件中多次出现字符串 `{namespace}`。

3. 保存 `webservices.pbd` 文件。
4. 打开 `<Agent_Home>` 目录中的 `appmap-soa.pbd` 文件。
5. 搜索所有 `{namespace}` 实例并用 `{servicename}` 将其替换。

6. 保存 *appmap-soa.pbd* 文件。
7. 重新启动应用程序服务器。

重新启动应用程序服务器后，调查器树和 SOA 依赖关系地图将显示 Web 服务端点名称，而不显示命名空间。

如果您使用的是服务端点名称，而不是命名空间，则您可能还希望更改“客户端和服务器概览”选项卡上显示的“命名空间”标签。

要更改“客户端和服务器概览”选项卡中使用的标签，请执行以下操作：

1. 打开 *<EM_Home>/ext/xmltv/* 目录中的 *ws.overview.tv.xml* 文件。
2. 搜索并使用服务替换所有命名空间实例。
3. 保存 *ws.overview.tv.xml* 文件。
4. 重新启动企业管理器。

重新启动企业管理器后，在调查器树中选择“Web 服务”、“客户端”或“服务器”节点时，“概览”选项卡将显示“服务”标签。

要使用 **Web Services Manager 7.0.x** 的度量标准命名约定，请执行以下操作：

1. 打开 *<Agent_Home>* 目录中的 *IntroscopeAgent.profile* 文件。
 2. 将 *com.wily.introscope.agent.soa.metricNameFormatting* 属性添加到文件中。
 3. 按如下设置 *com.wily.introscope.agent.soa.metricNameFormatting* 属性：
com.wily.introscope.agent.soa.metricNameFormatting=/
- 在度量标准名称中，该设置用下划线（_）替换正斜杠（/）和冒号（:）。
进行该设置后，<http://CheckingAccount/demobank.ca.com> 将显示为 http_CheckingAccount_demobank.ca.com。
4. 保存 *IntroscopeAgent.profile* 文件。
 5. 重新启动应用程序服务器。

重新启动应用程序服务器后，调查器树和 SOA 依赖关系地图将显示 Web 服务端点名称，而不显示命名空间。

配置关联跟踪

跨进程事务跟踪需要代理插入一个关联标识符，该标识符可从一个进程传递到另一个进程。代理可将该关联标识符插入 SOAP 或 HTTP 头。

由于大多数服务使用 SOAP 消息，因此默认情况下，从 SOAP 头插入和读取该关联标识符。但在某些情况下，您可能更愿意使用 HTTP 头传递关联标识符。例如，在少数情况下，向 SOAP 头添加关联标识符可导致消息被拒绝（由于关联 ID 更改了 SOAP 负载）。

根据应用程序处理 SOAP 消息和您实施安全性的方式，您可以通过修改 *IntroscopeAgent.profile* 文件来选择在 HTTP 头还是 SOAP 头中传递关联 ID。

有四个代理配置属性可以控制对客户端和服务器上的关联标识符的处理。您可以通过这些属性来指定标识符应当插入 SOAP 头、HTTP 头、这两种头协议，或者根本不插入。这些属性以标准代理前缀 *com.wily.introscope.agent* 开头：

soapheaderinsertion.enabled

允许客户端在 SOAP 头中插入关联标识符。

- 将属性设为 `true` 以允许客户端使用 SOAP 头。
- 如果希望阻止客户端在 SOAP 头中插入关联标识符，则将属性设为 `false`。

默认情况下，该属性设置为 `true`。

httpheaderinsertion.enabled

允许客户端在 HTTP 头中插入关联标识符。

- 将该属性设置为 `true` 以允许客户端使用 HTTP 头。
- 如果要防止客户端在 HTTP 头中插入关联标识符，可将该属性设置为 `false`。

默认情况下，该属性设置为 `false`。

soapheaderread.enabled

允许服务器从 SOAP 头中读取关联标识符。

- 将该属性设置为 true 以允许服务器使用 SOAP 头。
- 如果要防止服务器读取 SOAP 头，可将该属性设置为 false。

默认情况下，该属性设置为 true。

httpheaderread.enabled

允许服务器从 HTTP 头中读取关联标识符。

- 将该属性设置为 true 以允许服务器使用 HTTP 头。
- 如果要防止服务器读取 HTTP 头，可将该属性设置为 false。

默认情况下，该属性设置为 false。

配置客户端和服务器的属性

您可以逐个地对代理设置这些属性，以启用或禁用特定行为。例如，如果某个服务器处理基于 SOAP 和 HTTP 的服务中的消息，则您可能希望配置该服务器上的代理，以读取任一类型中的关联 ID，但阻止代理在 SOAP 头中插入标识符：

```
com.wily.introscope.agent.soapheaderread.enabled=true  
com.wily.introscope.agent.httpheaderread.enabled=true  
com.wily.introscope.agent.soapheaderinsertion.enabled=false  
com.wily.introscope.agent.httpheaderinsertion.enabled=true
```

通过进行这些设置，本地计算机可以读取 SOAP 头中由其他代理发送来的关联标识符，但是只能在 HTTP 头中插入关联标识符。但是，要启用跟踪，应记住 Web 服务客户端和服务器必须从相同类型的头插入和读取关联标识符。例如，如果将客户端配置为在 HTTP 头中插入关联标识符，则必须将服务器配置为从 HTTP 头读取标识符。

禁用相关联的跟踪

如果希望同时禁用 SOAP 和 HTTP 的关联标识符插入并关闭跨进程事务跟踪，则可将这四个属性都设为 false。例如：

```
com.wily.introscope.agent.soapheaderread.enabled=false  
com.wily.introscope.agent.httpheaderread.enabled=false  
com.wily.introscope.agent.soapheaderinsertion.enabled=false  
com.wily.introscope.agent.httpheaderinsertion.enabled=false
```

修改这些属性前，应记住 SOA 依存关系视图和跨进程事务跟踪需要关联标识符从一个流程传递到另一个流程。

如果将这些属性设为 `false` 以禁用相关跟踪，则 SOA 依赖关系地图将无法正常显示依赖关系，并且无法准确收集依赖关系度量标准，事务跟踪也可能不完整。

配置 SOAP 处理程序的插入点

您可以使用 `IntroscopeAgent.profile` 文件中的配置属性来控制 SOAP 处理程序在 SOAP 处理程序链中的插入位置。这些属性为包含加密或需要针对原始 SOAP 消息的签名验证的应用程序提供了灵活性，因此插入 SOAP 头不会影响应用程序对 SOAP 消息的验证。

用于插入 SOAP 处理程序的客户端和服务器属性

对于使用 SOAP 处理程序的应用程序服务器，您可以使用以下配置属性来控制 CA APM for SOA SOAP 头：

soa.client.prependhandler

使 SOAP 头在客户端上由 SOAP 处理程序链的第一个或最后一个处理程序插入。

- 将属性设为 `true`，以便由 SOAP 处理程序链的第一个处理程序插入 SOAP 头。
- 如果希望 SOAP 标头后面带 SOAP 处理程序链的最后一个处理程序，则将属性设为 `false`。

默认情况下，该属性设置为 `true`。

soa.server.appendhandler

使 SOAP 头在服务器上由 SOAP 处理程序链中的第一个或最后一个处理程序读取。

- 将该属性设置为 `true`，以便该 SOAP 头由 SOAP 处理程序链中的最后一个处理程序读取。
- 如果希望 SOAP 头由 SOAP 处理程序链中的第一个处理程序读取，然后将其删除，将该属性设为 `false`。

默认情况下，该属性设置为 `true`。

更改 SOAP 处理程序的默认顺序

默认情况下，客户端上处理程序链的第一个处理程序插入 SOAP 头，服务器上处理程序链的最后一个处理程序读取头。对于某些应用程序，可能需要修改默认行为以确认正确传递 CA APM for SOA SOAP 头和原始 SOAP 消息。例如，如果应用程序必须验证 SOAP 消息签名，则可能需要修改默认行为。

为说明为何要更改默认属性设置，请考虑由处理程序链中第一个处理程序插入 SOAP 头的应用程序。SOAP 消息随后在第二个处理程序中签名。在服务器上收到消息时，第一个处理程序会尝试验证签名。由于 CA APM for SOA SOAP 头已插入但尚未读取和删除，签名失败并且拒绝该消息。

通过配置属性，您可以更改默认行为，以便链的第一个处理程序读取 CA APM for SOA SOAP 头，然后删除该头。该设置允许链中下一个处理程序针对原始 SOAP 消息验证签名。例如：

```
com.wily.introscope.agent.soa.client.prependhandler=true  
com.wily.introscope.agent.soa.server.appendhandler=false
```

虽然默认属性设置在多数环境中是适当的，但您可通过这些配置属性灵活地使 CA APM for SOA 适应不同的应用程序方案。

配置 SOA 依赖关系地图的限制

存储在企业管理器中的依赖关系地图数据通常表示整个发现的应用程序中的所有依赖关系，提供了已部署的面向服务的体系结构的完整模型。但是，在非常大或复杂的 SOA 环境中，完整表示所有 SOA 组件及其依赖关系可能会影响企业管理器自身的性能和操作。

为了防止依赖关系地图影响企业管理器的性能，可以用配置属性来限制地图的大小和复杂性。默认情况下，这些配置属性将依赖关系地图限制为最多 5000 个节点和 25000 个依赖关系（每节点 5 个依赖关系的比率）。

您可以在 *IntroscopeEnterpriseManager.properties* 文件中设置以下企业管理器配置属性，以控制这些限制：

dependencymap.max.vertices

设置依赖关系数据可以存储在独立或收集器企业管理器上的最大节点数。

默认情况下，该属性设置为 5000。

dependencymap.max.edge.ratio

设置依赖关系与节点的最大比率，以控制可以存储在独立或收集器企业管理器上的依赖关系数据的总体复杂性。

默认情况下，该属性设置为 5。

对于大多数组织，SOA 网络需要的组件少于 5000 个，典型的依赖关系比率是每个组件对应 1 个或 2 个依赖关系。因此，相对于大多数组织的需要，默认设置允许的大小和复杂性更大。但是在少数情况下，您可能希望修改默认值。例如，您可能希望修改属性：

- 以便增加依赖关系地图的大小和复杂性（如果默认值不适用于您的 SOA 环境）。您应考虑更改可能对企业管理器性能的影响。
- 以便有意限制依赖关系地图的大小和复杂性（如果您的 SOA 环境不需要默认值允许的节点和依赖关系数）。您应考虑是否有任何更改会导致 SOA 依赖关系地图模型变得非必要的不完整。

例如，将属性修改成为更少的节点但每节点更多的依赖关系存储数据：

```
com.wily.introscope.soa.dependencymap.max.vertices=2000  
com.wily.introscope.soa.dependencymap.max.edge.ratio=8
```


第 7 章：监控 Oracle Service Bus

Oracle Service Bus (OSB) 提供了松散耦合服务使用者和服务提供商之间的消息处理和通信。通过 SOA Extension for Oracle Service Bus，您可以通过关键组件（如管道、代理服务和传输协议）来监控 Oracle Service Bus 操作。

本节介绍了特定于 OSB 的显示板、度量标准以及报警，可使用它们来监控和分析 Oracle Service Bus 环境的性能、可用性和总体运行状况。

此部分包含以下主题：

- [关于 Oracle Service Bus \(OSB\) \(p. 107\)](#)
- [如何启用对 Oracle Service Bus 的监控 \(p. 109\)](#)
- [使用显示板监控 Oracle Service Bus \(p. 113\)](#)
- [了解并查看 Oracle Service Bus 的度量标准 \(p. 116\)](#)
- [查看默认的 Oracle Service Bus 度量标准组 \(p. 120\)](#)
- [查看默认的 Oracle Service Bus 报警 \(p. 121\)](#)
- [查看 Oracle Service Bus 依赖关系 \(p. 122\)](#)
- [跟踪 Oracle Service Bus 的事务 \(p. 123\)](#)

关于 Oracle Service Bus (OSB)

在面向服务的体系结构中，企业服务总线通常在服务使用者和服务提供商之间提供一个消息层。企业服务总线支持跨分布式异构环境的数据和消息的验证、转换、路由以及安全性。

Oracle Service Bus 是企业服务总线的一个示例，它在代理接口的轻量型中间层中提供了消息代理、中介以及服务生命周期管理。

如果已使用 Oracle Service Bus 定义了处理消息的规则，则可用以下度量标准监控 Oracle Service Bus 的操作：

业务服务

业务服务是对于 Oracle Service Bus 是客户端的外部服务的定义。

外部服务由外部系统实施和托管。要使用外部服务，Oracle Service Bus 必须知道要调用的接口、调用方式以及调用的预期结果。OSB 中的业务服务定义外部接口模型，以便总线可以调用外部系统并与其交互。在 OSB 内，业务服务配置包括其接口、传输设置和安全设置。

使用 SOA Extension for OSB，您可以监控业务服务与外部系统交互的方式并收集有关其总体运行状况的数据（在“OSB”>“业务服务”节点下）。

管道

管道是处理“请求”、“响应”或“错误”消息流的特定步骤的命名顺序。

要实施代理服务的处理逻辑，需要在管道对节点中成对使用请求和响应管道。这些管道对节点可与其他节点组合到单根的树结构中，以控制整体流。错误管道处理消息流中的阶段和节点错误，也处理消息流或业务服务中的错误。

使用 SOA Extension for OSB，您可以监控请求和响应管道的性能并收集有关其总体运行状况的数据（在“OSB”>“管道”节点下）。

代理服务

代理服务是服务总线在本地实施和托管的中间 Web 服务的定义。

Oracle Service Bus 使用代理服务在业务服务与服务客户端（如演示应用程序或其他业务服务）之间路由消息。

代理服务配置包括其接口、传输设置、安全设置和消息流定义。消息流定义了消息流过代理服务时确定处理消息的方式的逻辑。

使用 SOA Extension for OSB，您可以监控代理服务的性能并收集有关其总体运行状况的数据（在“OSB”>“ProxyServices”节点下）。

传输

传输定义了发送和递交消息的机制，可包括 Oracle Service Bus 支持的任何传输协议。

传输提供程序管理传输端点的生命周期和运行时行为。目标端点是消息的源和目标资源。通过 OSB 的本地提供程序，可以配置需要这些传输协议的代理和业务服务。您还可以创建或安装自定义的传输提供程序。

使用 SOA Extension for OSB，您可以监控所有支持的传输协议并收集入站和出站端点的度量标准（在“OSB”>“传输”节点下）。

UDDI

通用描述、发现和集成 (UDDI) 是一个独立于平台、基于 XML 的注册中心，全球企业可使用该中心在 Internet 上列出它们的服务。

UDDI 是一个开放行业倡议，企业可通过其发布服务列表、搜索其他企业的服务列表并定义服务或软件应用程序在 Internet 上互交的方式。

通过符合 UDDI 版本 3.0 的 Oracle Service Bus 和 UDDI 注册中心，您可以：

- 将任何代理服务方面的信息发布到注册中心。注册中心可能基于 Web 服务描述语言 (WSDL)、SOAP 或 XML。
- 查询注册中心中的特定服务或列出所有可用服务。您可以根据业务实体和/或服务名称模式进行搜索。
- 从注册中心导入业务服务。

使用 SOA Extension for OSB，您可以监控 UDDI 注册中心并收集发布、导入和查询操作的度量标准（在“OSB”>“UDDI”节点下）。

XQuery

Oracle Service Bus 支持 XQuery，可使用 XQuery 引擎的 Oracle Data Services Platform 实施进行数据转换。

Oracle XQuery 引擎使用转换地图来描述数据类型之间的映射，并且 Oracle Service Bus 支持使用 XQuery 的数据映射。您可以创建、解析和执行转换。

使用 SOA Extension for OSB，您可以监控 XQuery 转换以创建、解析和执行 XQuery 并收集这些操作的度量标准（在“OSB”>“XQuery”节点下）。

如何启用对 Oracle Service Bus 的监控

启用对 Oracle Service Bus 的监控包括以下高级步骤：-

1. 确认已安装支持的 Oracle Service Bus 版本。
注意：有关系统要求，请参阅《兼容性指南》。
2. 确认已安装并启用代理和 CA APM for SOA。
3. 通过将代理配置文件配置为包含用于监控 Oracle Service Bus 的适当指令，[启用用于监控 Oracle Service Bus 的代理](#) (p. 110)。如果已使用独立代理安装程序或响应文件启用代理，则可跳过该步骤。
4. 通过将文件从 `<EM_Home>/examples/SOAEExtensionForOSB` 目录移动到适当的企业管理器目录，[启用企业管理器扩展](#) (p. 112)。

启用用于监控 Oracle Service Bus 的代理

安装代理并选择“WebLogic”作为应用程序服务器时，可以添加和启用 CA APM for Oracle Service Bus。还可以在安装代理后手工启用 CA APM for Oracle Service Bus。

如果安装代理时已启用 CA APM for Oracle Service Bus，则代理配置文件将自动配置成默认设置，无需其他步骤即可启用代理扩展。

如果安装代理时尚未启用 CA APM for Oracle Service Bus，则需要手工配置代理配置文件以启用监控。

手工启用 CA APM for Oracle Service Bus:

1. 确认已安装并启用代理和 CA APM for SOA。
2. 确认 CA APM for Oracle Service Bus 目录 *SOAExtensionForOSB* 位于 *<Agent_Home>/examples* 目录中，并将文件从 *<Agent_Home>/examples/SOAExtensionForOSB/ext* 目录复制到 *<Agent_Home>/core/ext* 目录。
3. 在文本编辑器中打开 *IntroscopeAgent.profile* 文件。
4. 将适合的 *OSB-full.pbl* 或 *OSB-typical.pbl* 文件添加到 *IntroscopeAgent.profile* 文件的 *introscope.autoprobe.directivesFile* 属性中。

有关选择适合的 ProbeBuilder 指令的详细信息，请参阅[关于 Oracle Service Bus 的指令文件](#) (p. 111)。

5. 如果希望更改默认设置，则在 *IntroscopeAgent.profile* 文件中修改其他特定于 SOA 的代理配置属性。

应注意，有些应用程序可能需要对默认设置进行更改，具体取决于实施安全的方式。

例如，如果为 Web 服务加密通信，则应修改代理的默认配置，以使代理能使用 HTTP 关联并禁用 SOAP 关联。默认情况下，Oracle Service Bus 的 SOAP 和 HTTP 关联都是启用的。有关设置 HTTP 或 SOAP 关联的配置属性的详细信息，请参阅[配置关联跟踪](#) (p. 101)。

6. 将所做更改保存到 *IntroscopeAgent.profile* 文件并关闭文本编辑器。

关于 Oracle Service Bus 的指令文件

在 *IntroscopeAgent.profile* 中设置 *introscope.autoprobe.directivesFile* 属性时，您可以选择默认的典型或完整检测来控制适合部署代理的环境的监控级别、度量标准可见性和性能开销。随后，通过打开或关闭对特定跟踪器组的跟踪，您可以使用典型或完整的切换文件来微调对特定组件的监控。

OSB-full.pbl

用于深入检视所有 Oracle Service Bus 组件的操作的完整检测。

完整检测提供了详细报告，但需要更多的开销。建议用于测试或开发环境，但生产环境通常不使用。

默认情况下，OSB-full.pbl 文件列表中列出了以下指令文件：

- OSB-toggles-full.pbd
- OSB.pbd

OSB-typical.pbl

对开销比较珍贵的生产环境中重要 Oracle Service Bus 组件的典型监控。

典型检测提供较少的详细信息但可减少开销，建议用于生产环境。

默认情况下，OSB-typical.pbl 文件列表中列出了以下指令文件：

- OSB-toggles-typical.pbd
- OSB.pbd

OSB-toggles-full.pbd

使用完整检测时，用于打开或关闭对 Oracle Service Bus 组件的监控的切换文件。

该文件为其他指令文件中提供的跟踪提供指令。默认情况下，该文件中的大多数跟踪器组已打开。

OSB-toggles-typical.pbd

在使用典型检测时，交替启用或关闭对 Oracle Service Bus 组件的监控。该文件为其他指令文件中提供的跟踪提供指令。默认情况下，仅会启用跟踪器组的一小部分。

应注意，向同一企业管理器报告数据的所有代理应当使用相同的检测级别，并且应将企业管理器配置对所有典型或完整检测代理执行操作。例如，如果使用典型检测配置代理，则应部署企业管理器文件的典型版本。

启用 Oracle Service Bus 企业管理器扩展

安装企业管理器时，您可以添加和启用 CA APM for Oracle Service Bus。还可以在安装企业管理器后手工启用 CA APM for Oracle Service Bus。

安装企业管理器时，CA APM for Oracle Service Bus 文件默认安装在 `<EM_Home>/examples` 目录中。要启用 CA APM for Oracle Service Bus，请将企业管理器文件从 `<EM_Home>/examples` 目录复制或移动到企业管理器主目录的适当位置。

注意：只有在企业管理器上启用 CA APM for SOA 后，才可以使用 CA APM for Oracle Service Bus。

遵循这些步骤：

1. 确认 CA APM for Oracle Service Bus 目录 SOAExtensionForOSB 位于 `<EM_Home>/examples` 目录中。
2. 将典型或完整检测的相应文件从 `<EM_Home>/examples/SOAExtensionForOSB` 目录复制到企业管理器目录结构中的对应位置。例如，如果您使用的是典型检测，则将 `com.wily.powerpacks.osb.emext.calculator_typical.jar` 文件从 `<EM_Home>/examples/SOAExtensionForOSB/ext` 目录复制到 `<EM_Home>/ext` 目录。
3. 如果企业管理器是群集环境中的收集器，则将 Oracle Service Bus 管理模块 `OSB_ManagementModule_typical.jar` 或 `OSB_ManagementModule_full.jar` 从 `<EM_Home>/config/modules` 目录中删除。

仅将管理模块复制到用作 MOM 计算机的企业管理器上的 `<EM_Home>/config/modules` 目录中。其他所有文件和脚本应当同时安装在收集器企业管理器和 MOM 企业管理器中。

4. 如果您正在从 SOA Extension for Oracle Service Bus 的之前版本升级，则删除企业管理器文件的旧版本。

- 如果企业管理器上已安装 CA APM for Oracle Service Bus 的早期版本，则必须先手工删除旧版本的企业管理器文件，然后再开始使用新版本的 CA APM for Oracle Service Bus。
- 如果从 CA APM for Oracle Service Bus 的早期版本升级，则从企业管理器主目录中删除以下文件：

```
<EM_home>/config/modules/OSB_ManagementModulev<version>.jar  
<EM_home>/ext/xmltv/OSB_overview.tv.xml  
<EM_home>/product/enterprisemanager/plugins/  
com.wily.powerpacks.osb.emext.calculator_<version>.jar
```

例如，如果要从版本 8.1.1 升级，请删除以下文件：

```
OSB_ManagementModulev8.1.1.0.jar  
OSB_overview.tv.xml  
com.wily.powerpacks.osb.emext.calculator_8.1.1.0.jar
```

5. 重新启动 Workstation。

将加载特定于 CA APM for Oracle Service Bus 的显示板和“概览”选项卡。

详细信息：

[在企业管理器上启用扩展 \(p. 35\)](#)

使用显示板监控 Oracle Service Bus

SOA Extension for Oracle Service Bus 包含多个预配置的显示板，您可以使用这些显示板监控应用程序环境的总体运行状况。显示板汇集已部署代理中的数据，以汇总性能信息并帮助您快速诊断和解决问题。

通常，将显示板用作监控环境的起点，因为使用显示板可以：

- 一目了然地监控 Oracle Service Bus 的关键组件的总体运行状况、性能、可用性和当前状态。
- 低级别度量标准发出已超过警告或危险阈值的信号时，及早获取生产应用环境中存在潜在问题的通知。
- 深入查看性能信息，以隔离和识别正遇到延迟或产生错误的 Oracle Service Bus 组件、传输协议、端点或操作。

预配置的 Oracle Service Bus 显示板在 Oracle Service Bus 的企业管理器扩展中打包，作为已部署的 OSB 管理模块（*OSB_ManagementModule_typical.jar* 或 *OSB_ManagementModule_full.jar*）的一部分。

OSB 管理模块为 Oracle Service Bus 提供下列预配置的显示板：

OSB 主目录

贯穿 Oracle Service Bus 关键组件的工作流的顶层视图，包括所有传输、代理服务、管道和业务服务总体运行状况的报警指示器。

OSB 代理服务—概览

所有代理服务的汇总状态，包括“平均响应时间”、“每个时间间隔的错误”和“停顿计数”的报警和图表，以及十个最慢代理服务的列表。

OSB 代理服务—详细信息

代理服务中请求和响应操作的独立汇总度量标准，包括“平均响应时间”、“每个时间间隔的错误”和“停顿计数”的报警和图表。

OSB 业务服务—概览

所有业务服务的汇总状态，包括“平均响应时间”和“每个时间间隔的错误”的报警和图表，以及十个最慢业务服务的列表。

OSB 管道—概览

所有管道的汇总状态，包括“平均响应时间”、“每个时间间隔的错误”和“停顿计数”的报警和图表，以及十个最慢的管道。

OSB 传输 I—概览

常用传输类型（HTTP/HTTPS、JMS、EJB、FTP、电子邮件和 SFTP）的汇总运行状况，包括所有传输类型中的“平均响应时间”、“每个时间间隔的错误”和“停顿计数”的报警，以及 HTTP/HTTPS、JMS、EJB、FTP、电子邮件和 SFTP 传输类型的单独“平均响应时间”图表。

OSB 传输 II—概览

其他传输类型（Tuxedo、MQ、WS、JPD、SB、DSP、文件和本地）的单独“平均响应时间”图表。

OSB 传输端点—概览

所有传输类型中的入站服务和出站端点的汇总状态，包括所有入站服务和出站端点的“平均响应时间”图表、“每个时间间隔的错误”和“停顿计数”报警，十个最慢入站服务的列表以及十个最慢出站端点的列表。

OSB XQuery一概览

所有 XQuery 执行尝试的汇总状态，包括“执行”操作的“平均响应时间”、“每个时间间隔的错误”及“停顿计数”的图表和报警，以及“解析”操作的“平均响应时间”和“每个时间间隔的错误”的图表。

OSB UDDI一概览

所有 UDDI 注册中心操作的汇总状态，包括 UDDI 注册中心“导入”和“发布”操作的“平均响应时间”和“每个时间间隔的错误”的图表和报警。

您可以使用 **Workstation** 控制台查看预配置的显示板。您还可以扩展 OSB 管理模块以包含自定义显示板，或者修改默认显示板定义以包含自定义度量标准或报警。

注意：有关创建和修改显示板的详细信息，请参阅《CA APM 配置和管理指南》。

遵循这些步骤：

1. 启动企业管理器（如果目前未运行）。
2. 启动 **Workstation** 并登录到已安装 SOA Extension for Oracle Service Bus 的企业管理器。
3. 单击“**Workstation**” > “新建控制台”。
4. 从“显示板”下拉列表中选择某个 OSB 显示板。

例如，选择“OSB 主目录”显示板可查看 Oracle Service Bus 的体系结构概览，其中具有 OSB 工作流所有关键组件（例如，事务、代理服务、管道和业务服务）的报警指示器。

5. 双击显示板中的某个报警，以打开该组件的显示板。

例如，双击“代理服务响应时间”报警，转到“OSB 代理服务 - 概览”显示板，以查看最慢代理服务的列表和所有代理服务的响应时间、停顿计数及错误的顶层报警指示器。

6. 双击显示板中的特定代理服务、管道或传输，以打开调查器进行进一步分析。

例如，在“代理服务概览”显示板中，您可以双击“十个最慢代理服务”列表中的某个特定代理服务，以便在调查器中查看其详细信息，并通过汇集的度量标准继续细化到所选管道的各个组件，以发现显示板报警指示的延迟或错误阈值的根本原因。

注意：有关在调查器中查看特定于 OSB 的信息的详细信息，请参阅[了解并查看 Oracle Service Bus 的度量标准 \(p. 116\)](#)。有关在显示板之间或在显示板和 Workstation 调查器之间导航的详细信息，请参阅《*CA APM Workstation 用户指南*》。

了解并查看 Oracle Service Bus 的度量标准

在调查器树中导航时，您可以查看大多数 Oracle Service Bus 组件和操作的标准 CA Introscope® 度量标准。标准度量标准数据被收集和聚合到特定于 OSB 的度量标准类别中，这些类别在调查器树中显示为节点和子节点。显示的特定度量标准类别和节点名称取决于您的应用程序使用的 Oracle Service Bus 资源。

在调查器树中浏览时，您可以选择查看针对各个操作的低级度量标准或查看聚合的度量标准（具体取决于您选择的节点），从而监控各种 OSB 组件的总体运行状况。

有关 CA CPM for SOA 提供的标准度量标准和其他度量标准的概览，请参阅[可用度量标准 \(p. 50\)](#)。有关 OSB 组件度量标准的其它信息，请参阅以下各节：

- [业务服务的度量标准 \(p. 117\)](#)
- [管道的度量标准 \(p. 118\)](#)
- [代理服务的度量标准 \(p. 118\)](#)
- [传输的度量标准 \(p. 119\)](#)
- [UDDI 的度量标准 \(p. 120\)](#)
- [XQuery 的度量标准 \(p. 120\)](#)

在调查器中查看和浏览 Oracle Service Bus 度量标准:

1. 展开代理节点，然后选择“OSB”以显示 Oracle Service Bus 组件的“概览”选项卡。“概览”选项卡列出了所有已定义供 Oracle Service Bus 使用的代理服务和业务服务的摘要信息。
2. 双击“概览”选项卡上的代理服务或业务服务，以便采用图形形式查看该代理服务或业务服务的所有聚合度量标准。
3. 展开 OSB 节点，以显示顶层 Oracle Service Bus 度量标准类别的子节点：业务服务、管道、代理服务、传输、UDDI 以及 XQuery。
4. 单击某个子节点，以显示具有该度量标准类别的摘要信息的“概览”选项卡。例如，单击“OSB”>“业务服务”以查看“概览”选项卡中列出的所有业务服务。
5. 展开某个子节点，以查看各个服务或操作，以及与各个服务或操作关联的度量标准。

例如，展开“业务服务”节点，以查看已在 Oracle Service Bus 中配置的所有业务服务。随后，可以展开单个业务服务名称，以查看该特定服务的标准度量标准。

业务服务的度量标准

对于 Oracle Service Bus，业务服务被定义为外部系统托管的外部企业服务。仅下列标准度量标准可用于“OSB”>“业务服务”节点下的 Oracle Service Bus 业务服务：

- 平均响应时间 (毫秒)
- 每个时间间隔的错误
- 每个时间间隔的响应数

在 Oracle Service Bus 平台上，业务服务是 Oracle Service Bus 可与其交换信息的外部服务。由于业务服务的消息表示对外部服务的调用，因此它不包括请求和响应事务。因此，计算业务服务的“平均响应时间”度量标准依赖于确定业务服务执行的总时间。

CA Introscope® 基于 Oracle Service Bus 传输时间、实际的服务执行时间和网络失效时间的总和计算业务服务的执行时间。例如，如果 OSB 传输时间是 10 毫秒，实际的服务执行时间是 18 毫秒，网络失效时间是 5 毫秒，则业务服务执行时间是 33 毫秒。

管道的度量标准

管道是处理“请求”、“响应”或“错误”消息流的特定步骤的命名顺序。如果您正使用 OSB 完整检测，所有标准 CA Introscope® 度量标准都可用于“OSB”>“管道”节点下的 Oracle Service Bus 管道。

如果您正在使用 OSB 典型检测，则不会在调查器中显示“管道”节点。如果您想在调查器中显示管道的度量标准，请修改
OSB-toggles-typical.pbd 文件，以启用管道跟踪。

代理服务的度量标准

代理服务定义了 Oracle Service Bus 应如何在业务服务之间路由消息。来自外部的网站服务或数据库的消息可能被传递到服务客户端（如演示应用程序或其他业务服务）。所有标准 CA Introscope® 度量标准都可用于“OSB”>“代理服务”节点下的 Oracle Service Bus 代理服务。

代理服务包括请求与响应事务。各个代理服务名称下的所有度量标准都是基于“请求”和“响应”子节点下所列度量标准的聚合度量标准。

例如，mortgageBroker1 到 loanGateway1 代理服务的并发调用由将 loanGateway1 请求添加到并发 loanGateway1 响应决定。

每个事务的响应时间定义为总往返时间。当收到来自某一调用方的请求时，事务开始。当响应返回调用方时，事务结束。

通过总事务处理时间除以事务数量计算平均响应时间。

传输的度量标准

传输定义了发送和递交消息的机制。对于 Oracle Service Bus，您可以为最常见的传输协议收集度量标准，包括默认情况下在 Oracle Service Bus 中支持的下列传输类型：

- HTTP/HTTPS
- JMS
- EJB
- FTP
- 电子邮件
- SFTP
- Tuxedo
- MQ
- WS
- JPD
- SB
- DSP
- 文件
- 本地

如果代理服务使用特定传输类型，则会在调查器树中显示该传输类型及其度量标准。如果一种传输类型未被任何服务使用，则不会在调查器树中显示该传输类型。

如果代理服务使用自定义传输类型，则会在调查器中显示为该传输类型定义的 Java 包名称。例如，如果 OSB 控制台使用自定义传输类型 com.bea.wli.sb.test.service，调查器会将 *com.bea.wli.sb.test.service* 显示为传输类型。

对于您正在使用的特定传输协议，所有标准 CA Introscope® 度量标准均在“OSB”>“传输”节点下提供。对于每个传输协议，显示的节点取决于您配置的是典型检测还是完整检测。

例如，如果您正在使用典型检测，则会在 `send` 和 `receive` 节点下汇总度量标准。如果您正在使用完整检测，然后展开一种传输类型，则显示入站和出站子节点，并且为该传输类型列出单个操作（如 `getRequestPayload`、`send` 或 `sendMessageAsync`）的度量标准。

UDDI 的度量标准

您必须将 AquaLogic 或 Oracle Service Bus 注册中心配置为与 UDDI 3.0 相容注册中心相互作用，然后才能收集和查看 UDDI 度量标准。如果已将 Oracle Service Bus 配置为使用 UDDI，则可以收集和查看导入、查询和发布操作的聚合度量标准。

对于导入、查询和发布操作，所有标准 CA Introscope® 度量标准均在“OSB”>“UDDI”子节点下提供。-

XQuery 的度量标准

如果已将 Oracle Service Bus 配置为使用 XQuery，则可以收集和查看创建、解析和执行操作的聚合度量标准。

对于创建、解析和执行操作，所有标准 CA Introscope® 度量标准均在“OSB”>“XQuery”子节点下提供。-

对于每个 XQuery，创建和解析方式的度量标准只填入一次。每次执行 XQuery 表达式或条件时都会填入执行方式的度量标准。

查看默认的 Oracle Service Bus 度量标准组

SOA Extension For Oracle Service Bus 包括用于定义默认显示板和报警的默认度量标准组。您还可以在自定义显示板和报警中使用这些默认度量标准组。

默认的度量标准组在 Oracle Service Bus 的企业管理器扩展中打包，作为已部署的 OSB 管理模块（*OSB_ManagementModule_typical.jar* 或 *OSB_ManagementModule_full.jar*）的一部分。

可使用 Workstation 管理模块编辑器查看默认度量标准组。您还可以扩展 OSB 管理模块以便包括自定义的度量标准组，或在自定义显示板或报警中使用默认的度量标准组。

遵循这些步骤：

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开您已部署的“*超级域*”>“管理模块”>“OSB_ManagementModule (*超级域*)”。

例如，如果您已经部署典型设置，则展开 *OSB_ManagementModule Typical* (*超级域*)。

3. 展开“度量标准组”节点，查看为 Oracle Service Bus 管理模块定义的所有度量标准组。
4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

您可以修改任意度量标准组的默认设置或自行创建自定义度量标准组。

注意：有关创建或修改度量标准组的详细信息，请参阅《CA APM Workstation 用户指南》。

查看默认的 Oracle Service Bus 报警

SOA Extension For Oracle Service Bus 包括在预配置的显示板中使用的默认报警定义。您也可以在自定义的显示板中使用这些默认报警。大部分默认报警预配置有默认的“警告”和“危险”阈值，如果超过阈值或重要级别增加，则会向控制台发送通知。

默认报警定义作为已部署的 OSB 管理模块（*OSB_ManagementModule_typical.jar* 或 *OSB_ManagementModule_full.jar*）的一部分，打包在 Oracle Service Bus 的企业管理器扩展中。

可以使用 Workstation 管理模块编辑器查看默认报警定义。还可以将 OSB 管理模块扩展为包括自定义报警定义和通知类型，或在自定义显示板中使用默认报警定义。

遵循这些步骤：

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开已部署的“*超级域*”>“管理模块”>“OSB_ManagementModule (*超级域*)”。
例如，如果已部署典型设置，则请展开“典型 OSB_ManagementModule (*超级域*)”。
3. 展开“报警”节点，查看为 Oracle Service Bus 管理模块定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

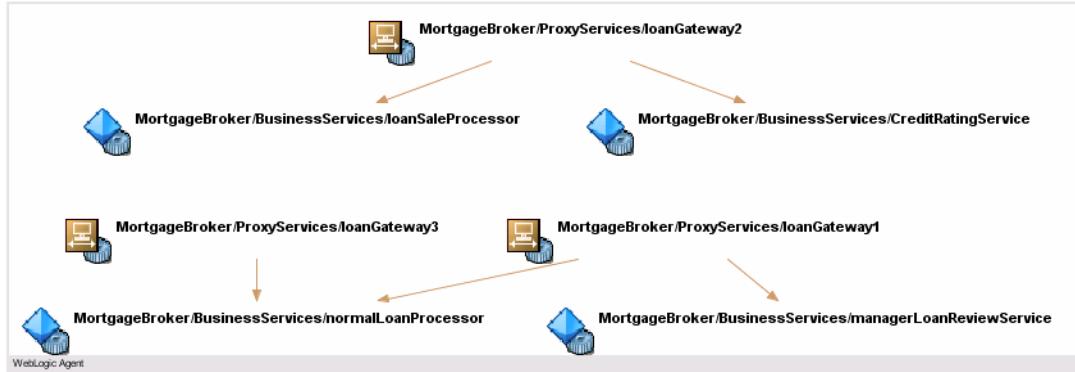
您应当注意查看“警告”和“危险”阈值的默认设置并根据需要调整这些值，并添加通知或纠正措施。

您可以修改任何报警的默认设置，也可创建您自己的自定义报警。

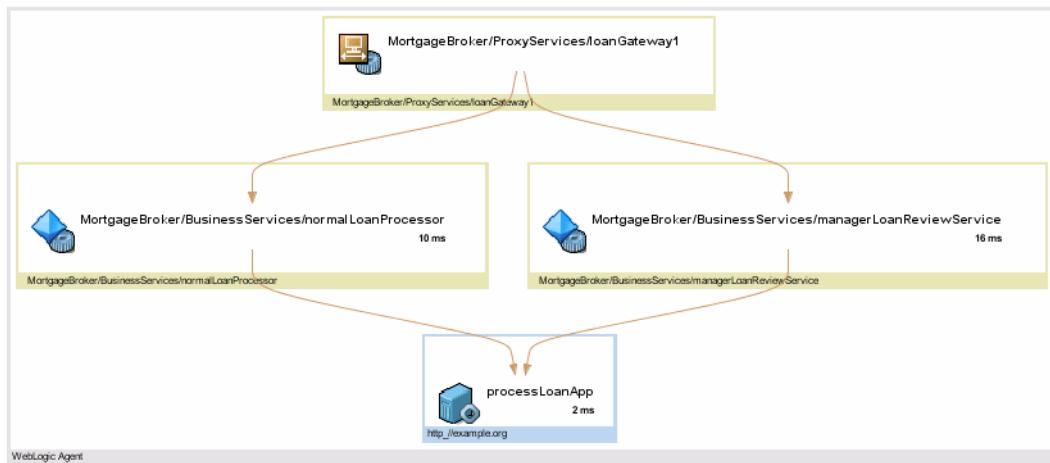
注意：有关创建或修改报警的详细信息，请参阅《CA APM Workstation 用户指南》。

查看 Oracle Service Bus 依赖关系

您可以通过在调查器树中选择“代理服务”或“业务服务”节点或单个代理服务或业务服务，然后单击“SOA 依赖关系地图”选项卡，来查看 Oracle Service Bus 代理服务和业务服务的依赖关系。例如，对于代理上所有代理服务的依赖关系的高级视图，您可以选择调查器中的“代理服务”节点，然后单击“SOA 依赖关系地图”选项卡：



您选择的节点决定了依赖关系地图中显示的上下文。然后您可以将您查看的上下文和详细信息级别卷起以进行折叠或放开以进行展开。例如，对于特定代理服务中依赖关系的高级视图，您可以在调查器中选择代理服务的名称，然后单击“SOA 依赖关系地图”选项卡。以下示例说明了调查器树中选定的用于处理贷款请求的代理服务，依赖关系地图已展开，以显示其他依赖关系：



您可以继续向地图中添加依赖关系级别，以便查看业务流程的整个工作流，或根据需要在地图的特定节点上进行放大。有关浏览依赖关系地图的更多详细信息，请参阅[使用 SOA 依赖关系地图 \(p. 63\)](#)。

跟踪 Oracle Service Bus 的事务

事务跟踪捕获完成事务所涉及的特定步骤，包括在 SOAP 消息、HTTP 或 HTTPS 标头，Java Message Service 调用中传递的操作，以及通过执行消息、数据或协议转换的 Oracle Service Bus 组件传递的操作。

如果事务包括通过 Oracle Service Bus 代理服务或其他 Oracle Service Bus 组件路由的任何消息，您可以在跨进程事务跟踪中查看有关执行的操作以及完成每个操作所用时间的信息。只要在每个被跟踪的节点上启用 CA APM for SOA 和 CA APM for Oracle Service Bus，您就可以跨任何平台组合跟踪业务事务。

了解跨进程事务跟踪的值

跨进程事务跟踪提供有关面向服务的体系结构中的松散耦合服务所执行操作的有价值信息。您可以使用跨进程事务跟踪来确定：

- 消息通过 Oracle Service Bus 组件进行转换和路由的方式。
- 事务期间调用的业务服务和代理服务。
- 事务期间执行调用的顺序。
- 处理请求或回复最慢的位置。

启动并查看示例事务跟踪

您可以以下述方式启动事务跟踪会话：

- 直接通过 SOA 依赖关系视图中的视图节点。
- 通过单击“Workstation”>“新建事务跟踪会话”，从 Workstation 手工启动。

如果从依赖关系视图启动事务跟踪，则视图节点类型自动设置默认筛选。如果手工启动新的事务跟踪会话，可以选择下列 Oracle Service Bus 筛选类型之一：

- 业务服务
- 代理服务
- 命名空间
- 操作

例如，要筛选事务中的特定 Oracle Service Bus 代理服务，您可以选择代理服务筛选，然后键入代理服务的全称或部分名称。

启动事务跟踪会话之后，您可以使用事务跟踪查看器来查看有关选定事务各个段的摘要和详细信息。因为涉及 Oracle Service Bus 的事务通常包括异步调用，所以对于在事务过程中异步执行的流程，您可能会发现单击“顺序视图”查看事务工作流非常有用。“顺序视图”显示流程执行的顺序（以该顺序可确定为限度）。对于 Oracle Service Bus 事务，该顺序不一定表示传统的调用方-被调用方关联关系，也可能说明某一流程触发执行其他流程的时间。

注意：有关事务跟踪的详细信息，请参阅[在 SOA 环境中使用事务跟踪](#) (p. 85)。有关为代理配置跟踪的更多详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。有关使用事务跟踪视图和历史数据的详细信息，请参阅《CA APM Workstation 用户指南》。

第8章：监控 TIBCO BusinessWorks

TIBCO BusinessWorks 是一个包括多个基础架构组件和多种功能的 SOA 平台。通过 CA APM for TIBCO BusinessWorks，可以监控 TIBCO BusinessWorks 处理引擎的许多关键元素，包括流程启动器、作业实例和传输协议。

本节介绍了特定于 TIBCO BusinessWorks 的显示板、度量标准以及报警，可使用它们来监控和分析 TIBCO BusinessWorks 环境的性能、可用性和总体运行状况。

此部分包含以下主题：

- [关于 TIBCO BusinessWorks \(p. 125\)](#)
- [如何启用对 TIBCO BusinessWorks 的监控 \(p. 127\)](#)
- [使用显示板监控 TIBCO BusinessWorks \(p. 133\)](#)
- [了解并查看 TIBCO BusinessWorks 的度量标准 \(p. 135\)](#)
- [查看默认的 TIBCO BusinessWorks 度量标准组 \(p. 151\)](#)
- [查看默认的 TIBCO BusinessWorks 报警 \(p. 152\)](#)
- [查看 TIBCO BusinessWorks 依赖关系 \(p. 153\)](#)
- [跟踪 TIBCO BusinessWorks 的事务 \(p. 154\)](#)
- [关于 BusinessWorks 的前端和后端 \(p. 157\)](#)
- [自定义度量标准老化和删除 \(p. 159\)](#)
- [配置 TIBCO BusinessWorks 的关联跟踪 \(p. 160\)](#)

关于 TIBCO BusinessWorks

利用 TIBCO BusinessWorks，公司能够公开和集成现有服务和旧系统、设计和测试新服务并将松散耦合服务编排和装入到应用程序中。

作为集成平台，BusinessWorks 包含多个分布式子系统。例如，TIBCO BusinessWorks 支持多种消息服务和传输协议。

您可以通过下列高级组件的度量标准来监控 TIBCO BusinessWorks 的运行：

活动

活动是单个工作单元，用于执行 TIBCO 业务流程定义中的操作。

通过 TIBCO BusinessWorks 扩展，您可以在“Tibco”>“活动”节点下监控业务流程中活动的性能和总体运行状况。

组操作

组操作确定一组相关活动参与的操作类型。

通过 TIBCO BusinessWorks 扩展，您可以监控“Tibco”>“组操作”节点下业务流程中的活动性能和总体运行状况。

Hawk 度量标准

TIBCO Hawk 微代理向本地代理提供监控主机操作系统的方式。

通过 TIBCO BusinessWorks 扩展，您可以在“Tibco”>“Hawk 度量标准”节点下监控 TIBCO Hawk 微代理收集的流程引擎统计信息。

作业

作业表示流程实例的执行，这些流程实例在内存中创建，并由 TIBCO BusinessWorks 引擎在作业池内执行。

通过 TIBCO BusinessWorks 扩展，您可以在“Tibco”>“作业”节点下监控流程启动器和作业池的性能和总体运行状况。

流程

流程是旨在完成特定任务的业务工作流。在 TIBCO Designer 中定义的流程可以包括子流程，这些子流程作为父流程的一部分运行，或与父流程并行运行。单个业务流程是 TIBCO BusinessWorks 流程定义的一个运行时实例。

通过 TIBCO BusinessWorks 扩展，您可以在“Tibco”>“流程”节点下监控业务流程的性能和总体运行状况。

传输

传输定义了发送和递交消息的机制。

通过 TIBCO BusinessWorks 扩展，您可以在“Tibco”>“传输”节点下监控 HTTP、SOAP 和 FTP 传输协议的性能和总体运行状况，还可通过“Tibco”>“传输”>“RV”节点下的度量标准监控 Rendezvous 的性能和总体运行状况。

WebServices

Web 服务度量标准表示客户端和服务器业务服务端点和每个服务内的关联操作。

通过 TIBCO BusinessWorks 扩展，您可以在“Web 服务”节点下监控客户端和服务器 Web 服务端点的性能和总体运行状况。

如何启用对 TIBCO BusinessWorks 的监控

要为 TIBCO BusinessWorks 启用监控，请执行以下高级步骤：-

1. 验证您是否已安装受支持的 TIBCO BusinessWorks。

注意：有关 TIBCO BusinessWorks 要求的完整列表，请参阅《兼容性指南》的“SOA 性能管理”部分。

2. 确认已安装并启用代理和 CA APM for SOA。
3. [配置代理配置文件以监控 TIBCO BusinessWorks \(p. 127\)](#)，使代理可以使用 CA APM for TIBCO BusinessWorks。如果已使用独立代理安装程序或响应文件在代理中启用了 CA APM for TIBCO BusinessWorks，则可以跳过该步骤。
4. [配置 TIBCO BusinessWorks 以使用代理 \(p. 130\)](#)。
5. [启用适用于 TIBCO BusinessWorks 的企业管理器扩展 \(p. 132\)](#)。

启用代理以监控 TIBCO BusinessWorks

可以在安装代理时选择“默认”作为应用程序服务器来启用对 TIBCO BusinessWorks 的监控，也可在安装代理后手工启用该监控。

如果在安装代理时选择 CA APM for TIBCO BusinessWorks，则代理配置文件将自动配置成默认设置，无需执行其他步骤。

如果未在安装代理时选择 CA APM for TIBCO BusinessWorks，则需要手工配置代理配置文件以启用监控。

手工启用 SOA Extension for TIBCO BusinessWorks:

1. 确认已安装并启用代理和 CA APM for SOA。
2. 确认 CA APM for TIBCO BusinessWorks 目录 *SOAExtensionForTibcoBW* 位于 *<Agent_Home>/examples* 目录中。
3. 将文件从 *<Agent_Home>/examples/SOAExtensionForTibcoBW/ext* 目录复制到 *<Agent_Home>/core/ext* 目录。
4. 在文本编辑器中打开 *IntroscopeAgent.profile* 文件。
5. 将典型或完整检测的适当指令文件添加到 *IntroscopeAgent.profile* 文件的 *introscope.autoprobe.directivesFile* 属性中。
有关选择典型或完整检测的详细信息,请参阅[关于典型检测和完整检测](#) (p. 128)。有关通过 ProbeBuilder 指令文件自定义跟踪的详细信息,请参阅[关于 TIBCO BusinessWorks 的指令文件](#) (p. 129)。
6. 将所做更改保存到 *IntroscopeAgent.profile* 文件并关闭文本编辑器。

关于典型检测和完整检测

在 *IntroscopeAgent.profile* 中设置 *introscope.autoprobe.directivesFile* 属性时,您可以选择默认的典型或完整检测来控制适合部署代理的环境的监控级别、度量标准可见性和性能开销。随后,通过打开或关闭对特定跟踪器组的跟踪,您可以使用典型或完整的切换文件来微调对特定组件的监控。

tibcobw-full.pbl

提供完整检测,以便深入检视所有 TIBCO BusinessWorks 组件的操作。

完整检测提供了详细报告,但需要更多的开销。建议用于测试或开发环境,但生产环境通常不使用。

默认情况下, *tibcobw-full.pbl* 文件列表中列出了以下指令文件:

- *tibcobw-toggles-full.pbd*
- *tibcobw-webservices.pbd*
- *tibcobw-processes.pbd*
- *tibcobw-tasks.pbd*
- *tibcobw-RV.pbd*
- *tibcobw-activities.pbd*
- *tibcobw-jobs.pbd*
- *tibcobw-correlation.pbd*
- *tibcobw-transports.pbd*
- *tibcobw-hawk.pbd*

tibcobw-typical.pbl

定义对开销比较珍贵的生产环境中重要 TIBCO BusinessWorks 组件的典型监控。

典型检测提供较少的详细信息但可减少开销，建议用于生产环境。

默认情况下，tibcobw-typical.pbl 文件列表中列出了以下指令文件：

- tibcobw-toggles-typical.pbd
- tibcobw-webservices.pbd
- tibcobw-processes.pbd
- tibcobw-tasks.pbd
- tibcobw-RV.pbd
- tibcobw-activities.pbd
- tibcobw-correlation.pbd

tibcobw-toggles-full.pbd

在使用完全检测时，启用或关闭对 TIBCO BusinessWorks 组件的监控的切换文件。

该文件为其他指令文件中提供的跟踪提供指令。默认情况下，该文件中的大多数跟踪器组已打开。

tibcobw-toggles-typical.pbd

在使用典型检测时，启用或关闭对 TIBCO BusinessWorks 组件的监控的切换文件。该文件为其他指令文件中提供的跟踪提供指令。默认情况下，仅会启用跟踪器组的一小部分。

有关指令文件的详细信息，请参阅[关于 TIBCO BusinessWorks 的指令文件](#)(p. 129)。

关于 TIBCO BusinessWorks 的指令文件

通过 *tibcobw-toggles-full.pbd* 和 *tibcobw-toggles-typical.pbd* 文件，可以控制 ProbeBuilder 指令 (.pbd) 文件集中提供的默认跟踪。您也可以视需要修改下列文件中的设置来手动自定义提供的跟踪：

tibcobw-activities.pbd

提供针对业务流程内的活动和分组活动操作的监控规则。

tibcobw-correlation.pbd

启用针对 TIBCO BusinessWorks 组件的跨进程事务跟踪

tibcobw-hawk.pbd

提供针对 TIBCO BusinessWorks 内的 Hawk 度量标准的监控规则。

tibcobw-jobs.pbd

提供针对 TIBCO BusinessWorks 上运行的作业和作业池的监控规则。

tibcobw-processes.pbd

提供针对 TIBCO BusinessWorks 业务流程定义（包括流程启动器以及派生和非派生子流程）的监控规则。

tibcobw-RV.pbd

提供针对 BusinessWorks 内的 Rendezvous 消息服务的监控规则。

tibcobw-tasks.pbd

提供针对 TIBCO BusinessWorks 业务流程内的任务的监控规则。

tibcobw-transports.pbd

提供针对单个传输协议（如 FTP、HTTP 和 SOAP）的监控规则。

tibcobw-webservices.pbd

提供针对 BusinessWorks Web 服务端点的监控规则。

注意：有关配置跟踪的详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

配置 TIBCO BusinessWorks 以使用代理

使用安装程序启用代理或安装后手工启用代理之后，必须将 TIBCO BusinessWorks 配置为运行代理来检测应用程序。配置步骤取决于您在环境中部署存档的方式。

在 TIBCO Designer 中创建流程定义时，可将这些定义作为项目存档 (EAR) 导出。每个项目存档随后可包含一个或多个流程存档 (PAR)。

在典型的 TIBCO BusinessWorks 部署中，每个流程存档在其自己的 JVM 内运行。如果环境中的每个流程存档都在其自己的 JVM 内运行，您可以使用下列步骤将 TIBCO BusinessWorks 配置为使用代理。

在没有服务容器的情况下配置 TIBCO BusinessWorks

1. 打开 TIBCO BusinessWorks 中的 *bwengine.tra* 文件或已部署应用程序的 *<application>.tra* 文件，将以下条目添加到 *java.extended.properties* 属性：

```
java.extended.properties=-javaagent:<Agent_Home>/Agent.jar -Dcom.wily.introscope.agentProfile=<Agent_Home>/core/config/IntroscopeAgent.profile
```

注意：对于 UNIX 上的 TIBCO BusinessWorks，必须为 *java.extended.properties* 中的冒号 (:) 和等号 (=) 进行转义。

如果您已经部署了应用程序，请在更改 *bwengine.tra* 文件之后取消部署这些应用程序，然后再对其进行重新部署。或者，也可以将该属性添加到各个应用程序的 *.tra* 文件。例如，您可将该属性添加到 *<TIBCO_DOMAIN_HOME>/<DOMAIN_NAME>/application/<application_name>* 目录下的 *<application_name>.tra* 文件。

2. 在已安装 SOA Extension For Tibco BusinessWorks 的计算机上，停止然后重新启动应用程序的任何正在运行的服务实例。

作为在每个流程存档自己的 JVM 中运行流程存档的替代方法，您可以决定部署一个 TIBCO BusinessWorks 服务容器。您可通过服务容器在单个 JVM 中运行多个流程存档。如果使用服务容器来部署流程存档，您可以使用下列步骤将 TIBCO BusinessWorks 配置为使用代理。

在使用服务容器时配置 TIBCO BusinessWorks

1. 打开 *<TIBCO_BW_HOME>/bin* 目录中的 *bwcontainer.tra* 文件，将以下条目添加到 *java.extended.properties* 属性：

```
java.extended.properties=-javaagent:<Agent_Home>/Agent.jar  
-Dcom.wily.introscope.agentProfile=<Agent_Home>/core/config/IntroscopeAgent.profile -Dcom.wily.introscope.agent.agentName=<Agent_Name>
```

注意：对于 UNIX 上的 TIBCO BusinessWorks，必须为 *java.extended.properties* 中的冒号 (:) 和等号 (=) 进行转义。

2. 确认已部署应用程序的 *bwengine.tra* 或 *<application>.tra* 文件未配置代理。
3. 停止并重新启动服务容器实例。

配置自动代理命名

大多数 TIBCO BusinessWorks 域具有部署在多个流程存档 (PAR) 中的多个业务流程。由于每个流程存档都在其自己的 JVM 实例中运行，因此每个流程存档也会启动其自己的代理实例。

为避免使用每个应用程序的 *.tra* 文件中的 *com.wily.introscope.agent.agentName* 属性手动命名每个代理实例，SOA Extension For TIBCO BusinessWorks 提供了配置文件和属性，以便能够使用 TIBCO BusinessWorks 域名前缀后跟下划线 (_) 和应用程序名称来对代理进行动态命名。例如，如果 TIBCO 域名为 *caDomain*，应用程序名称为 *soap_over_http-Process_Archive*，在启用自动代理命名时，会动态构成如下代理名称：

caDomain_soap_over_http-Process_Archive

如果找不到流程存档的名称属性，则使用部署名称自动命名代理。例如：
caDomain_soap_over_http

配置自动代理命名:

1. 停止所有运行的 TIBCO BusinessWorks 应用程序。
2. 验证 *TibcoBWNaming.jar* 文件是否位于 *<Agent_Home>/core/ext* 目录中。

如果您没有看到该文件, 请将其从
<Agent_Home>/examples/SOAExtensionForTibcoBW/ext 目录复制到
<Agent_Home>/core/ext 目录。

3. 在文本编辑器中打开 *<代理主目录>/core/config/IntroscopeAgent.profile* 文件。
4. 找到 *introscope.agent.agentAutoNamingEnabled* 属性并将其设置为 *true*。例如:
`introscope.agent.agentAutoNamingEnabled=true`
5. 设置连接延迟属性, 以定义尝试自动命名时代理连接到企业管理器应当延迟的时间。例如, 要延迟连接 30 秒, 请按如下所示设置属性:
`introscope.agent.agentAutoNamingMaximumConnectionDelayInSeconds=30`
6. 设置重命名时间间隔属性, 以定义启用自动命名时代理应检查动态构成的名称的频率。例如, 要每一分钟检查一次自动生成的代理名称, 请按如下所示设置属性:
`introscope.agent.agentAutoRenamingIntervalInMinutes=1`
7. 保存对 *IntroscopeAgent.profile* 文件所做的更改。
8. 重新启动 TIBCO BusinessWorks 应用程序。

您只应为在单独流程存档中部署的 TIBCO BusinessWorks 应用程序使用自动代理命名。如果使用 BusinessWorks 服务容器来部署流程存档, 应将 *TibcoBWNaming.jar* 文件从 *<Agent_Home>/ext* 目录中删除。

启用企业管理器扩展

安装企业管理器时, CA APM for TIBCO BusinessWorks 文件默认安装在 *<EM_Home>/examples* 目录中。要启用 CA APM for TIBCO BusinessWorks, 需要将企业管理器文件从 *<EM_Home>/examples* 目录复制或移动到企业管理器主目录中的适当位置。

注意: 在企业管理器上启用 CA APM for SOA 后, 才可使用 SOA Extension for TIBCO BusinessWorks。有关启用 CA APM for SOA 企业管理器扩展的信息, 请参阅[在企业管理器上启用扩展 \(p. 35\)](#)。

遵循这些步骤：

1. 确认 CA APM for TIBCO BusinessWorks 目录 *SOAExtensionForTibcoBW* 位于 *<EM_Home>/examples* 目录中，然后将文件从 *<EM_Home>/examples/SOAExtensionForTibcoBW* 目录复制到企业管理器目录结构中的对应位置。例如，将文件从 *<EM_Home>/examples/SOAExtensionForTibcoBW/ext* 目录复制到 *<EM_Home>/ext* 目录。
2. 如果企业管理器是群集环境中的收集器，则将 CA APM for TIBCO BusinessWorks 管理模块 *TibcoBWManagementModule.jar* 从 *<EM_Home>/config/modules* 目录中删除。
应仅将管理模块复制到用作 MOM 计算机的企业管理器上的 *<企业管理器主目录>/config/modules* 目录中。其他所有文件和脚本应当同时安装在收集器企业管理器和 MOM 企业管理器中。
3. 重新启动 Workstation，以加载特定于 CA APM for TIBCO BusinessWorks 的显示板和“概览”选项卡。

使用显示板监控 TIBCO BusinessWorks

SOA Extension for TIBCO BusinessWorks 包括多个预配置的显示板，您可以使用这些显示板监控应用程序环境的总体运行状况。显示板汇集已部署代理中的数据，以汇总性能信息并帮助您快速诊断和解决问题。

通常，将显示板用作监控环境的起点，因为使用显示板可以：

- 一目了然地监控 TIBCO BusinessWorks 的关键组件的总体运行状况、性能、可用性和当前状态；
- 在低级别度量标准发出已超过警告或危险阈值的信号时，及早获取生产应用环境中存在潜在问题的通知；
- 深入查看性能信息，以隔离和识别正遇到延迟或产生错误的 TIBCO BusinessWorks 流程、传输协议或 Web 服务。

预配置的 TIBCO BusinessWorks 显示板在 TIBCO BusinessWorks 的企业管理器扩展中打包，作为 SOA Extension for TIBCO BusinessWorks 管理模块 (*TibcoBWManagementModule.jar*) 的一部分。

TIBCO BusinessWorks 管理模块为 TIBCO BusinessWorks 提供下列预配置的显示板：

Tibco BW—概览

TIBCO BusinessWorks 关键活动的顶级概览，包括所有 Web 服务的总响应时间、错误、停顿和 SOAP 错误的报警指示器，以及业务流程的总响应时间、错误和停顿的报警指示器。

Tibco BW—业务流程

所有业务流程的汇总状态，包括业务流程的响应时间、错误计数和停顿计数的报警指示器和图表，以及最慢业务流程的列表。

Tibco BW—活动

所有活动的汇总状态，包括活动的响应时间、错误计数和停顿计数的报警指示器和图表，以及最慢活动的列表。

Tibco BW—作业

所有作业和作业池的汇总状态，以及已创建作业数、正运行作业数、每小时创建作业数和已完成作业数图表。

该显示板还包括指出受监控的流程启动器数和受监控作业的流限制设置的图表。

Tibco BW—传输

所有传输类型的汇总状态，以及每种传输类型的所有操作的平均响应时间图表。例如，如果您正在监控使用 SOAP 的 TIBCO BusinessWorks 流程，该显示板将显示 SOAP 操作（如 `writeSoapEnvelope` 和 `sendMessage`）的平均响应时间。

您可以使用 Workstation 控制台查看预配置的显示板。还可以将 TIBCO BusinessWorks 管理模块扩展为包括自定义显示板，或者将默认显示板定义修改为包括自定义度量标准或报警。

注意：有关创建和修改显示板的详细信息，请参阅《CA APM 配置和管理指南》。

遵循这些步骤：

1. 启动企业管理器（如果目前未运行）。
2. 启动 Workstation 并登录到安装有 SOA Extension for TIBCO 的企业管理器。
3. 单击“Workstation”>“新建控制台”。
4. 从“显示板”下拉列表中选择 TIBCO BusinessWorks 显示板之一。
例如，选择“Tibco BW—概览”显示板，以查看 TIBCO BusinessWorks 前端、后端、Web 服务以及业务流程的顶级报警指示器。

5. 双击其他选项卡或报警打开相关显示板，以查看更详细的信息。

例如，双击“业务流程”选项卡或“业务流程响应时间”报警，以查看有关最慢的业务流程以及所有业务流程的平均响应时间、每个时间间隔的错误和停顿计数的更详细信息。

6. 双击显示板中的特定业务流程、活动或作业名称，打开调查器以进行进一步分析。例如，双击“Tibco—业务流程”显示板中最慢的流程，可打开显示该流程的调查器。

注意：有关在调查器中查看特定于 TIBCO 的信息的详细信息，请参阅[了解并查看 TIBCO BusinessWorks 的度量标准 \(p. 135\)](#)。有关启动和使用 Workstation、访问显示板或打开和导航调查器的信息，请参阅《CA APM Workstation 用户指南》。

了解并查看 TIBCO BusinessWorks 的度量标准

在调查器树中导航时，您可以查看大多数 TIBCO BusinessWorks 基础架构组件的标准 CA Introscope® 度量标准。收集标准度量标准的数据并将其聚合到在调查器树中显示为节点和子节点且特定于 TIBCO 的度量标准类别中。显示的特定度量标准类别和节点名称取决于已在环境中部署的流程、服务和资源。

在调查器树中导航时，可根据所选节点查看各操作的低级度量标准或聚合度量标准，以便监控各种 TIBCO BusinessWorks 组件的总体运行状况。

有关 CA APM for SOA 提供的标准度量标准和其他度量标准的概览，请参阅[了解可用的度量标准 \(p. 50\)](#)。有关特别用于监控 TIBCO BusinessWorks 或用于特定的 specificTIBCO BusinessWorks 组件的度量标准的其他信息，请参阅以下度量标准类别。

- [活动的度量标准 \(p. 136\)](#)
- [组操作的度量标准 \(p. 137\)](#)
- [Hawk 的度量标准 \(p. 137\)](#)
- [作业和作业池的度量标准 \(p. 144\)](#)
- [流程的度量标准 \(p. 146\)](#)
- [传输的度量标准 \(p. 148\)](#)
- [Web 服务的度量标准 \(p. 150\)](#)

在调查器中查看和浏览 TIBCO BusinessWorks 度量标准:

1. 展开代理节点，然后单击“Tibco”节点以显示“概览”选项卡，该选卡列出所有正在监控的 TIBCO BusinessWorks 流程的摘要信息。
2. 双击某个流程，以图形格式查看该流程的所有度量标准。
根据检测的级别，您可能也会看到与选定流程关联的任务列表。
3. 展开“Tibco”节点，以显示顶级 TIBCO BusinessWorks 度量标准类别的子节点。
4. 单击或展开子节点，以显示带有该度量标准类别相关摘要信息的“概览”选项卡。例如，单击“作业”节点以在“概览”选项卡中显示作业池度量标准。
要看传输的概览信息，请展开“传输”节点，然后选择“FTP”、“HTTP”或“SOAP”节点以显示“概览”选项卡。对于 Rendezvous 度量标准，展开“传输”>“RV”，然后选择某个子节点以显示“概览”选项卡。
5. 展开任意子节点以查看有关单个组件的更详细信息，如子流程、活动或内存中作业以及其中每一项的关联度量标准。

活动的度量标准

活动是 TIBCO 业务流程定义中的单个工作单元。活动通常是连接外部系统的操作，但活动也可用于内部处理。在 TIBCO Designer 中，活动按照类型分组，并在各种面板中提供。例如，您可以使用“常规活动”面板进行普通操作（如调用流程，设置计时器，或写入日志文件），或使用“文件”面板定义文件处理活动。

在调查器树中，活动也会按要执行的活动类型分组，节点名称反映某种活动类型的特定操作。例如，调查器的“文件”节点可包括 FileCreateActivity、FileReadActivity 和 FileWriteActivity 的度量标准。一些活动的节点名称可能与 TIBCO Designer 中使用的面板名称类似，但并非总是如此。例如，“常规活动”面板中的活动可能在调查器树的“CallProcess”、“错误处理”或“流”节点下列出。

所有标准 CA Introscope® 度量标准都可用于“Tibco”>“活动”节点下的 TIBCO BusinessWorks 活动。

组操作的度量标准

组用来创建成组的活动，这些活动一起来完成某种特定类型的操作。例如，可以使用组来定义一组重复执行直到某个条件为真（有条件执行）的活动，或用其来定义一组参与同一事务并一起提交和回滚的活动。组操作确定活动参与的操作类型，如执行 if-then 条件的一组活动的 If 组，或按顺序对每一项进行重复的一组活动的 Iterative Loop 组。

“组操作”的度量标准监控执行操作（如提交或回滚事务，或通过一系列步骤迭代）的组的性能。度量标准并非基于组内各个活动的聚合数据。

所有标准 CA Introscope® 度量标准均可用于“Tibco”>“组操作”节点下的 TIBCO BusinessWorks 组。

Hawk 的度量标准

TIBCO Hawk 微代理提供针对 BusinessWorks 处理引擎、流程定义、活动度量标准和内存使用的监控。

您可使用下列位于“Tibco”>“Hawk 度量标准”节点下的度量标准类别查看 Hawk 微代理度量标准：

- [ExecInfo \(p. 138\)](#) 度量标准提供有关流程引擎的执行操作的信息，如系统运行时间和执行的线程数。
- [MemoryUsage \(p. 138\)](#) 度量标准描述可用内存和已用内存。
- [ProcessDefs \(p. 138\)](#) 度量标准提供了有关单个流程定义和为每个流程定义执行的活动的详细信息。
- [Processes \(p. 141\)](#) 度量标准提供有关活动流程实例的摘要信息。（这些度量标准仅适用于 Hawk 4.8.1。）
- [ServicesInfo \(p. 142\)](#) 度量标准提供有关所有已部署服务的信息。
- [Status \(p. 143\)](#) 度量标准提供有关流程引擎状态的常规信息。

执行度量标准

以下度量标准可用于 “Tibco” > “Hawk 度量标准” > “ExecInfo” 节点下的 TIBCO BusinessWorks。

Status

执行引擎的当前状态。该度量标准值指出引擎在受监控的服务器上是“活动”、“已挂起”、“待命”还是“正在停止”。

Threads

引擎中的工作线程数目。

Uptime

流程引擎启动以来经过的总时间。

Version

受监控服务器的配置版本信息。

内存使用情况度量标准

以下度量标准可用于 “Tibco” > “Hawk 度量标准” > “MemoryUsage” 节点下的 TIBCO BusinessWorks。

FreeBytes

可用字节的总数。

PercentUsed

当前使用的总字节数的百分比。

TotalBytes

已分配给引擎流程的总字节数。

UsedBytes

当前使用的总字节数。

流程定义和活动度量标准

对于单个流程子节点，下列度量标准可用于 TIBCO BusinessWorks（在 “Tibco” > “Hawk 度量标准” > “ProcessDefs” 节点下）：

Aborted

选定流程定义已中止的次数。

AverageElapsed

使用选定的流程定义完成的所有流程平均经过的时间(以毫秒表示)。

AverageExecution

使用所选流程定义完成的所有流程的平均执行时间 (毫秒)

Checkpointed

所选流程定义充当检查点的次数。

Completed

所选流程定义已完成的次数。

CountSinceReset

自上次重置以来完成的流程数。

Created

为选定的流程定义创建的流程数。

MaxElapsed

使用所选流程定义完成的所有流程的最长处理时间（毫秒）。

MaxExecution

使用所选流程定义完成的所有流程的最长执行时间（毫秒）。

MinElapsed

使用所选流程定义完成的所有流程的最短处理时间（毫秒）。

MinExecution

使用所选流程定义完成的所有流程的最短执行时间（毫秒）。

MostRecentElapsedTime

最近经过的时间（以毫秒表示）。

MostRecentExecutionTime

最新执行时间（毫秒）。

Name

流程定义的名称。

Queued

所选流程定义被放入队列中的次数。

Starter

选定流程定义的流程启动器活动的名称。

Suspended

使用选定流程定义的流程挂起的次数。

Swapped

所选流程定义被交换的次数。

TimeSinceLastUpdate

更新最新的值以来的毫秒数。

TotalElapsed

使用所选流程定义完成的所有流程的总处理时间（毫秒）。

TotalExecution

使用所选流程定义完成的所有流程的总执行时间（毫秒）。

“Tibco” > “Hawk 度量标准” > “ProcessDefs” > “*process_name*” 节点下单个活动名称的下列度量标准可用于 TIBCO BusinessWorks：

ActivityClass

实施选定活动的类的名称。

CalledProcessDefs

选定活动调用的流程定义的名称。

ElapsedTime

选定活动的所有调用使用的总运行时间（毫秒），包括等待睡眠、调用流程和等待活动花费的所有时间。

ErrorCount

选定活动返回错误的次数。

ExecutionCount

TIBCO BusinessWorks 引擎实例执行选定活动的次数。

ExecutionCountSinceReset

自上次重置以来执行选定活动的次数。

ExecutionTime

选定活动的所有调用使用的总执行时间（毫秒）。

该度量标准不包括睡眠、调用流程和等待活动的等待时间。

LastReturnCode

由最近执行的选定活动返回的状态。该度量标准的有效返回代码是 OK、ERROR、DEAD 或 DEBUG。

MaxElapsedTime

选定活动的最大运行时间（毫秒）。

MaxExecutionTime

所选活动的最长执行时间（毫秒）。

MinElapsedTime

所选活动的最短处理时间（毫秒）。

MinExecutionTime

所选活动的最短执行时间（毫秒）。

MostRecentElapsedTime

最近经过的时间（以毫秒表示）。

MostRecentExecutionTime

最新执行时间（毫秒）。

Name

活动的名称。

TimeSinceLastUpdate

更新最新的值以来的毫秒数。

Tracing

表示是否对选定活动启用跟踪。

流程度量标准

适用于 Hawk 4.8.1:

如果使用的是 4.8.1 版本的 Hawk，“Tibco”>“Hawk 度量标准”>“流程”下的下列度量标准可用于 TIBCO BusinessWorks：

CurrentActivityName

当前由选定流程执行的活动的名称。

CustomId

选定流程的自定义标识符（如果存在）。

Duration

自选定流程启动以来，时钟运行时间（毫秒）。

MainProcessName

主流程定义的名称。

Name

选定流程的流程名称。

StartTime

流程的启动时间（毫秒）。

StarterName

启动选定流程实例的流程启动器的名称。

Status

所选流程的当前状态。例如，该度量标准表示作业当前是否处于活动状态。

SubProcessName

选定流程的子流程定义的名称（如果适用）。

TrackingId

跟踪所选流程的标识符（如果存在）。

服务度量标准

以下度量标准可用于“Tibco”>“Hawk 度量标准”>“ServicesInfo”节点下的 TIBCO BusinessWorks。

Description

服务说明。

EndpointURL

服务的端点 URL 地址。

Name

服务的名称。

PortName

服务的端口名称。

PortTypeName

服务的端口类型说明。

TransportType

服务的传输类型说明。

WSDL_Namespace

服务的具体 Web 服务描述语言 (WSDL) 命名空间。

WSDL_URL

服务的 Web 服务描述语言 (WSDL) URL 地址。

bindingName

服务的绑定名称。

状态度量标准

以下度量标准可用于“Tibco”>“Hawk 度量标准”>“状态”节点下的 TIBCO BusinessWorks。

Adapter Name

应用程序的名称。

Host

运行 TIBCO BusinessWorks 引擎流程的主机的名称。

Instance ID

引擎实例的实例标识符。

New Errors

自上次调用该度量标准以来出现的新错误的数目。

Process ID

TIBCO BusinessWorks 引擎流程的流程标识符。

Total Errors

自启动以来的错误总数。

Uptime

自启动以来的秒数。

启用 Hawk 度量标准

收集 Hawk 度量标准是性能密集型选项，默认情况下不启用。启用该选项之前应考虑性能开销。您要使用 Hawk 微代理启用 Hawk 度量标准的收集：

遵循这些步骤：

1. 在文本编辑器中打开 *IntroscopeAgent.profile* 文件。
2. 设置 *introscope.autoprobe.directivesFile* 属性以使用完全检测监控级别。例如：
`introscope.autoprobe.directivesFile=tibcobw-full.pbl`
3. 取消注释并将下列 Hawk 属性设置为 true：

```
com.wily.soaextension.tibcobw.hawkmonitor.enabled=true
```

将该属性设置为 true 可以启用对 TIBCO Hawk 的监控。

4. 通过取消注释下列属性可以控制从 Hawk 微代理收集度量标准的轮询时间间隔。例如：
`com.wily.soaextension.tibcobw.hawkmonitor.frequency`
注意：默认情况下，收集度量标准的轮询时间间隔为 30 秒（30000 毫秒）。
5. 重新启动应用程序服务器。
这些更改将生效。

作业和作业池的度量标准

当流程定义的流程启动器接收数据以开始执行时，将创建作业。作业代表流程实例中任务的执行。大多数情况下，作业是在内存中创建的，并由 TIBCO BusinessWorks 引擎在作业池中执行。在转到下一个作业之前，作业池中的每个作业可以执行有限数目的任务。作业池中可执行的最大任务数是由 StepCount 属性控制的。BusinessWorks 引擎可并发执行的最大作业数是由 ThreadCount 属性定义的。

“作业”和“作业池”类别中的度量标准有助于您监控创建的、保存在内存中的和每个作业池中执行的作业数。作业和作业池的度量标准每 30 秒由代理刷新一次。

“Tibco”>“作业”节点下的下列度量标准可用于 TIBCO BusinessWorks 作业和作业池：

作业池中的活动作业

选定作业池中活动和非内存页流程的总数。

作业池中的作业错误计数

自启动以来选定作业池中所有作业的错误总数。

作业池中的运行中作业

所选作业池中正在运行的流程和已排队流程的总数。

线程计数

为 BusinessWorks 引擎配置的最大工作线程数。

完成作业数

自启动以来已执行完成的流程数。

创建作业数

自启动以来在 BusinessWorks 引擎内创建的流程数。

流限制

“TIBCO BusinessWorks 流限制”设置的值。

每小时创建的作业数

每小时在 TIBCO BusinessWorks 引擎内创建的流程数。

进程启动器状态

表示作业的进程启动器的当前状态。

运行中作业数

TIBCO BusinessWorks 引擎中运行的业务流程数。

可以展开任何单个作业池名称，以显示已为该流程实例创建的作业的子节点。

启用或禁用作业和作业池的度量标准

当使用默认轮询时间间隔为 30 秒的完全检测时，默认情况下作业和作业池监控处于开启状态。可以使用以下属性来启用或禁用对作业和作业池的监控，以及控制轮询时间间隔频率：

```
com.wily.soaextension.tibcobw.jobmonitor.enabled  
com.wily.soaextension.tibcobw.jobmonitor.frequency
```

例如，如果使用的是典型检测，可以将

`com.wily.soaextension.tibcobw.jobmonitor.enabled` 属性设置为 `true` 以启用监控；如果使用的是完全检测，可以将该属性设置为 `false` 以禁用监控：
`com.wily.soaextension.tibcobw.jobmonitor.enabled=true`

然后可以使用 `com.wily.soaextension.tibcobw.jobmonitor.frequency` 属性控制收集度量标准的轮询时间间隔。例如：

```
com.wily.soaextension.tibcobw.jobmonitor.frequency=30000
```

如果使用的是典型检测并且将

`com.wily.soaextension.tibcobw.jobmonitor.enabled` 属性设置为 `true`，则还需要在 `tibcobw-toggles-typical.pbd` 文件中取消下列行的注释以启用作业和作业池监控：

```
#TurnOn: JobTracing  
#TurnOn: JobPoolTracing
```

例如，将 `com.wily.soaextension.tibcobw.jobmonitor.enabled` 属性设置为 `true`，并修改 `tibcobw-toggles-typical.pbd` 文件，以在使用典型检测时启用作业和作业池监控跟踪器：

```
TurnOn: JobTracing  
TurnOn: JobPoolTracing
```

如果更改任何一个作业监控属性，应重新启动应用程序服务器来使更改生效。

流程的度量标准

TIBCO BusinessWorks 的主要组件是其流程引擎。流程引擎包括设计、部署和管理业务流程的工具。在 TIBCO BusinessWorks 中，业务流程是 TIBCO 流程定义的运行时实例。流程定义可以包括活动、子流程和任务，可以通过监控业务流程的这些元素来监控流程引擎。

所有标准 CA Introscope® 度量标准均可用于“Tibco”>“流程”节点下的业务流程、生成的子流程以及任务。对于非生成的子流程，只能使用“平均响应时间”、“并发调用”、“每个时间间隔的错误”以及“每个时间间隔的响应”。-

除了标准度量标准之外，下列度量标准也可用于流程和生成的子流程：

运行平均响应时间

业务流程在 TIBCO BusinessWorks 引擎中主动执行所花费的平均时间（毫秒），不包括等待输入所花费的时间。

正在运行的并发调用数

正在 TIBCO BusinessWorks 引擎上运行的业务流程数。

每个时间间隔正在运行的响应数

每个时间间隔内在 TIBCO BusinessWorks 引擎中成功运行的业务流程的数目。

此度量标准并不表示业务流程在时间间隔内已完成。例如，度量标准可以包括挂起执行的业务流程。

例如，展开“Tibco”>“流程”以显示在 TIBCO BusinessWorks 的 TIBCO Designer 中定义的业务流程名称。“流程”节点可能包括与父业务流程和单个任务节点处于同一级别的生成或非生成的子流程：



也可以为 TIBCO BusinessWorks 流程收集依赖关系和偏差度量标准。有关标准依赖关系和偏差度量标准的信息，请参阅[使用调查器查看 SOA 性能度量标准 \(p. 49\)](#)。

关于业务流程的完成和主动执行

由于业务流程定义可以包括计时器或等待用户输入的挂起执行，因此“流程”度量标准类别提供度量下列内容的度量标准：

- 业务流程完成：这些度量标准说明业务流程的端对端处理运行状况，不管流程处于挂起还是活动状态。
- 业务流程主动执行：只有业务流程在 TIBCO BusinessWorks 引擎中主动执行时，这些度量标准才会提供业务流程的处理运行状况。

业务流程的“**平均响应时间**”度量标准度量从开始到完成的平均响应时间，包括挂起流程等待输入或用户响应的任何时间。例如，如果业务流程在 BusinessWorks 引擎中启动并且花费 15 毫秒主动执行时间，等待用户输入花费 30 毫秒，然后花费 10 毫秒完成执行操作，则从开始到完成的总响应时间是 15 毫秒、30 毫秒和 10 毫秒的总和，即总共 55 毫秒。例如：

$$\text{平均响应时间 (毫秒)} = \text{执行时间} + \text{挂起时间}$$

由于“平均响应时间”度量标准包括处理挂起的时间，因此不反映 BusinessWorks 引擎自身的性能。单独的“**运行平均响应时间**”度量标准跟踪业务流程在 BusinessWorks 引擎中主动执行的时间。

“运行平均响应时间”度量标准仅度量业务流程的主动执行时间。它不包括等待输入花费的所有时间。例如，如果业务流程在 BusinessWorks 引擎中启动并执行 15 毫秒，停止执行等待用户输入，然后在 BusinessWorks 引擎中执行 10 毫秒完成业务流程，则首次执行操作的“运行平均响应时间”是 15 毫秒，第二次执行操作的“运行平均响应时间”是 10 毫秒。不包括挂起流程等待用户输入的时间。例如：

$$\text{运行平均响应时间 (毫秒)} = \text{总执行时间} - \text{挂起时间}$$

同样，标准“**并发调用**”和“**每个时间间隔的响应**”度量标准将所有流程都计算在内，不管是主动运行的还是挂起的。“**正在运行的并发调用数**”和“**每个时间间隔正在运行的响应数**”度量标准仅反映 BusinessWorks 引擎中发生的处理活动。

“运行平均响应时间”、“正在运行的并发调用数”和“每个时间间隔正在运行的响应数”度量标准都仅适用于业务流程和生成的流程，而不适用于非生成的子流程或单个任务。

关于业务流程的停顿计数

对于业务流程，“**停顿计数**”度量标准仅考虑 BusinessWorks 引擎中主动运行的流程。挂起等待用户输入的流程不包括在停顿计数中。

关于子流程中每个时间间隔的错误

如果执行非生成的子流程时出现错误，则将仅对该特定子流程显示“**每个时间间隔的错误**”度量标准。如果运行非生成的子流程时出现错误，子流程及其父流程的“**每个时间间隔的错误**”计数都将加一。

对于父流程和生成的子流程，默认情况下总是显示“**每个时间间隔的错误**”度量标准。如果运行生成的子流程时出现错误，则会将错误记录在该子流程的“**每个时间间隔的错误**”度量标准中。但是，不会将生成的子流程的错误添加到父流程的“**每个时间间隔的错误**”计数中。

传输的度量标准

传输定义了发送和递交消息的机制。对于 TIBCO BusinessWorks，可以收集一系列传输协议的度量标准，包括可用于在 TIBCO BusinessWorks 中负责消息路由和处理的企业消息系统 TIBCO Rendezvous，以及 HTTP、SOAP、JMS、FTP 和 XML 等标准传输协议。

通过 TIBCO BusinessWorks 扩展，您可以在“Tibco”>“传输”节点下监控 HTTP、SOAP 和 FTP 传输协议的性能和总体运行状况，还可通过“Tibco”>“传输”>“RV”节点下的度量标准监控 Rendezvous 的性能和总体运行状况。

还可以通过打开标准 *j2ee.pbd* 文件中的 JMS 和 XML 跟踪，来监控 JMS 和 XML 的性能和总体运行状况。默认情况下，*j2ee.pbd* 文件随代理一起提供，但是可与 TIBCO BusinessWorks 扩展功能一起使用来提供其他监控。

对于您正在使用的特定传输协议（如 SOAP 和 HTTP），所有标准 CA Introscope® 度量标准均可用于“传输”节点下的各种操作。然后可以展开传输协议节点以显示传输类（如驱动程序、处理程序或请求程序类），随后展开子节点以深入查看单个操作（如 *sendMessage* 或 *writeSoapEnvelope*）的度量标准。

如果要使用代理的 *j2ee.pbd* 文件监控 JMS 或 XML，可以查看代理节点下的 JMS 和 XML 传输协议节点，而不是“Tibco”>“传输”节点。

Rendezvous (RV) 的度量标准

Rendezvous 是企业消息系统，通过该系统，分布的应用程序能够在网络中交换数据。作为集成平台，TIBCO BusinessWorks 将 Rendezvous 用作其支持的消息服务之一来可靠地处理发送与接收的大量消息。

Rendezvous 的度量标准提供关于 Rendezvous 消息传送、队列和传输的详细信息。在 TIBCO BusinessWorks 中，可以使用“Tibco”>“传输”>“RV”节点下以下类别的度量标准来监控 Rendezvous：

Message Listeners

消息侦听程序是侦听入站 Rendezvous 消息的对象。在新的入站消息到达时，侦听程序会调用“消息处理器”对象来处理消息。

Message Processors

消息处理器是处理消息侦听程序所发现的消息的回调方式。

Queue Groups

通过队列组，可以对事件的分发顺序进行细粒度控制。每个队列在组中的相对优先级决定着分发流程调用队列以分发其事件的顺序。

此类别的度量标准监控分发方式的响应时间、吞吐量和错误。

Queues

队列用于按照异步事件（如入站消息或计时器流程）的接收顺序存储这些事件，直至可以由消息处理器处理这些事件为止。事件发生时，将置于事件队列中，以等待分发例程将其发送至处理器进行处理。

此类别的度量标准监控分发方式的响应时间、吞吐量和错误。

Transport

传输定义了发送和递交消息的机制。对于 TIBCO BusinessWorks 的 Rendezvous，可以收集“可靠传输”（标准 Rendezvous 传输）和“认证传输”（RVCM）传输类型的度量标准。

消息侦听程序、消息处理器和传输度量标准

对于消息侦听程序、消息处理器以及可靠和认证传输协议，所有标准 CA Introscope® 度量标准均可用于“Tibco”>“传输”>“RV”子节点下的 Rendezvous。

队列组度量标准

“Tibco”>“传输”>“RV”>“队列组”节点下提供了下列分发相关度量标准以用于与单个队列组关联的分发方法：

- 平均响应时间 (毫秒)
- 每个时间间隔的错误
- 每个时间间隔的响应数

队列度量标准

“队列”节点下的下列分发相关度量标准可用于与单个队列关联的分发方法：

- 平均响应时间 (毫秒)
- 每个时间间隔的错误
- 每个时间间隔的响应数

启用或禁用 Rendezvous 的度量标准

当使用默认轮询时间间隔为 30 秒的典型或完全检测时，默认情况下将启用对 Rendezvous 的监控。如果要禁用对 Rendezvous 的监控，请注释掉 *tibcobw-toggles-typical.pbd* 或 *tibcobw-toggles-full.pbd* 文件中的下列行：

```
TurnOn: TibrvTransport  
TurnOn: TibrvDispatchable  
TurnOn: TibcoRVMessageProcessorTracing  
TurnOn: TibcoRVMessageListenerTracing
```

例如，如果使用的是典型检测，请在文本编辑器中打开 *tibcobw-toggles-typical.pbd* 文件，并按照如下方法修改相应行来禁用跟踪：

```
#TurnOn: TibrvTransport  
#TurnOn: TibrvDispatchable  
#TurnOn: TibcoRVMessageProcessorTracing  
#TurnOn: TibcoRVMessageListenerTracing
```

Web 服务的度量标准

所有标准 CA Introscope® 度量标准均可用于“Web 服务”节点下的客户端和服务器 Web 服务端点。此外，“每个时间间隔的 SOAP 错误”是在 SOA 平台中监控 Web 服务的所有扩展功能的标准度量标准。有关“每个时间间隔的错误”和“每个时间间隔的 SOAP 错误”之间的差异的信息，请参阅“查看‘错误’选项卡上的 SOAP 错误和错误度量标准”。

您还可以收集 TIBCO BusinessWorks Web 服务操作的依赖关系和偏差度量标准。有关标准依赖关系和偏差度量标准的信息，请参阅[使用调查器查看 SOA 性能度量标准 \(p. 49\)](#)。

查看调查器中的 Web 服务度量标准并进行导航：

1. 展开代理节点，然后展开“Web 服务”以显示 Web 服务端点的“客户端”和“服务器”节点。
2. 展开“客户端”或“服务器”节点，以便显示要查看其度量标准的单个 Web 服务端点。

3. 展开特定的 Web 服务端点 <命名空间> 或 <服务名称>, 以显示操作或选择该 Web 服务的度量标准。
4. 展开 Web 服务中的特定操作, 以显示该操作的度量标准。

关于服务器端点的平均响应时间

对于服务器端操作, “平均响应时间” 度量标准表示 TIBCO BusinessWorks 接收和解析传入的 SOAP 消息所需要的时间。大多数情况下, 该处理只是更复杂的事务的一部分。例如, 典型的 BusinessWorks 事务包括 Web 服务服务器端的下列阶段:

- 接收和解析传入 SOAP 消息所花费的时间
- BusinessWorks 引擎执行由 Web 服务公开的流程定义以及准备发送回复所花费的时间
- 发送回复所花费的时间

由于每个阶段的事务可以在不同的线程上运行, 因此“平均响应时间”度量标准只能度量每次操作的第一部分事务, 然后聚合 Web 服务端点的数据。由于度量标准无法计算其他线程上的处理时间, 因此度量标准不表示整个 Web 服务事务的总响应时间。

关于客户端和服务器 SOAP 错误度量标准

对于服务器端操作 (“Web 服务” > “服务器” > “命名空间”), “每个时间间隔的 SOAP 错误” 度量标准表示 BusinessWorks 响应请求时向 Web 服务客户端发送的 SOAP 错误的数目。

对于客户端操作 (“Web 服务” > “客户端” > “命名空间”), “每个时间间隔的 SOAP 错误” 度量标准表示客户端中 SOAPSendReceiveActivity 接收的 SOAP 错误的数目。

查看默认的 TIBCO BusinessWorks 度量标准组

SOA Extension for TIBCO BusinessWorks 包括定义默认显示板和报警用的默认度量标准组。您还可以在自定义显示板和报警中使用这些默认度量标准组。

默认度量标准组打包在 SOA Extension for TIBCO BusinessWorks Management Module (*TibcoBWManagementModule.jar*) 的组成部分 Enterprise Manager Extension for TIBCO BusinessWorks 中。

遵循这些步骤:

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>Introscope SOA Extension for TibcoBW <version> (*超级域*)。
3. 展开“度量标准组”节点，查看为 TIBCO 管理模块定义的所有度量标准组。
4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

您可以修改任意度量标准组的默认设置或自行创建自定义度量标准组。

注意: 有关创建或修改度量标准组的详细信息，请参阅《CA APM Workstation 用户指南》。

查看默认的 TIBCO BusinessWorks 报警

SOA Extension for TIBCO BusinessWorks 包括预配置的显示板中使用的默认报警定义。您也可以在自定义的显示板中使用这些默认报警。大部分默认报警预配置有默认的“警告”和“危险”阈值，如果超过阈值或重要级别增加，则会向控制台发送通知。

默认报警定义作为 TIBCO BusinessWorks 管理模块 (*TibcoBWManagementModule.jar*) SOA 扩展的一部分，打包在 TIBCO BusinessWorks 的企业管理器扩展中。

查看 TIBCO BusinessWorks 代理的默认报警定义:

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>“TibcoBW <version> 的 Introscope SOA 扩展 (*超级域*)”。
3. 展开“报警”节点，查看为 TIBCO BusinessWorks 管理模块定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

您应当注意查看“警告”和“危险”阈值的默认设置并根据需要调整这些值，并添加通知或纠正措施。

您可以修改任何报警的默认设置，也可创建您自己的自定义报警。

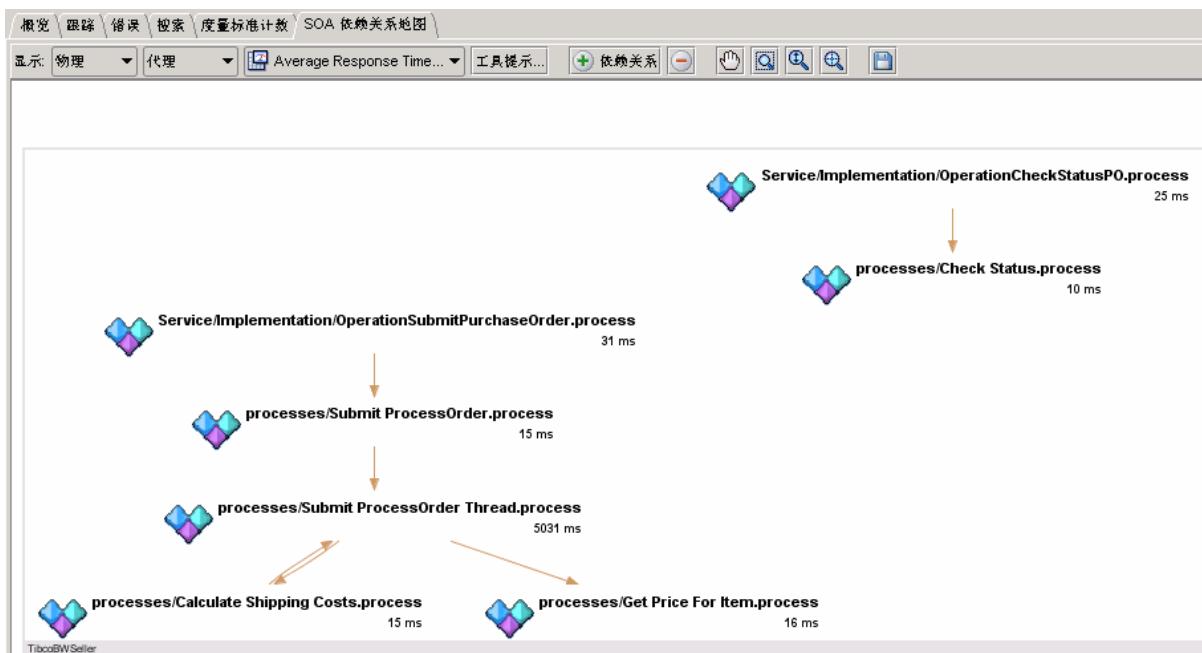
注意: 有关创建或修改报警的详细信息，请参阅《CA APM Workstation 用户指南》。

查看 TIBCO BusinessWorks 依赖关系

通过在调查器树中选择 TIBCO BusinessWorks “流程” 或 “Web 服务” 节点，然后单击“SOA 依赖关系地图”选项卡，可以查看 TIBCO BusinessWorks Web 服务、服务操作和流程定义的依赖关系。

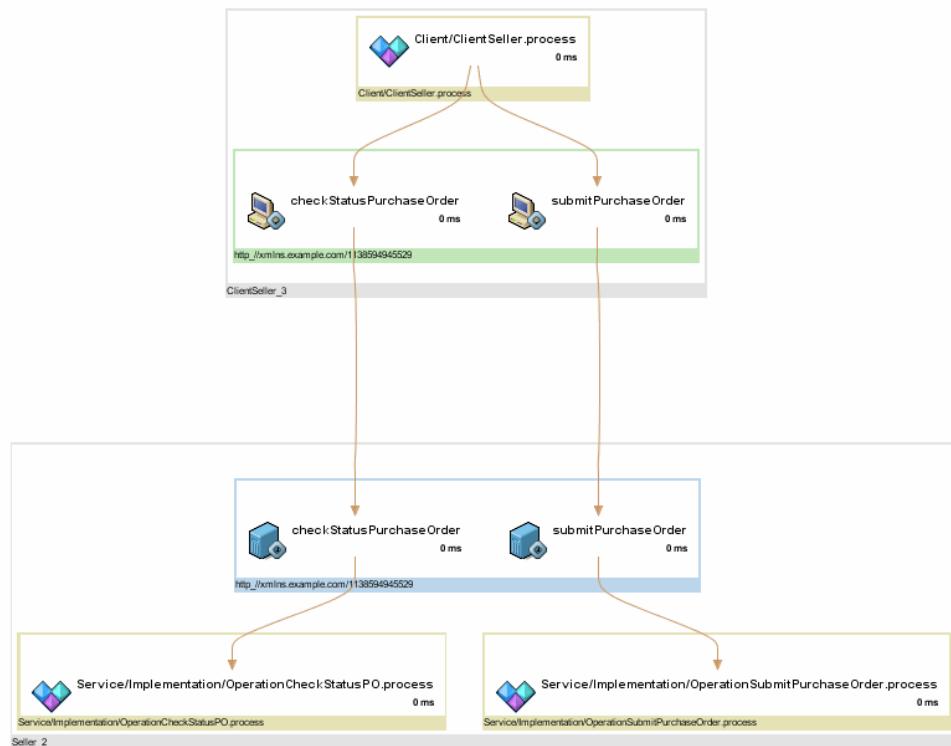
您选择的节点决定了依赖关系地图中显示的上下文。然后您可以将您查看的上下文和详细信息级别卷起以进行折叠或放开以进行展开。

例如，对于代理上所有业务流程中的依赖关系的高级视图，您可以在调查器中选择“流程”节点，然后单击“SOA 依赖关系地图”选项卡：



要查看代理中特定业务流程依赖关系的高级视图，可以在调查器中选择业务流程名称，然后单击“SOA 依赖关系地图”选项卡。请注意，更改内容类型或调查器树节点时，也将更改依赖关系地图的上下文。例如，如果将内容类型从“服务”更改为“操作”，地图会显示更改，但是地图中包含的详细信息可能相同，也可能不同，具体取决于在调查器树中选择的节点以及流程定义。

以下示例说明在调查器树中选择的用于提交采购订单申请的业务流程，该业务流程已展开以显示其他依赖关系。



您可以继续向地图中添加依赖关系级别，以便查看业务流程的整个工作流，或根据需要在地图的特定节点上进行放大。有关导航依赖关系地图的更多详细信息，包括显示和隐藏依赖关系级别，请参阅[使用 SOA 依赖关系地图](#) (p. 63)。

跟踪 TIBCO BusinessWorks 的事务

事务跟踪提供完成业务事务所涉及的特定步骤的详细或摘要视图。对于 TIBCO BusinessWorks 业务流程或 Web 服务，可以跟踪包括通过下列协议路由的操作的事务：

- 简单对象访问协议 (SOAP)
- 超文本传输协议 (HTTP)
- 安全的超文本传输协议 (HTTPS)
- Java Message Service (JMS)

涉及 TIBCO BusinessWorks 业务流程或 Web 服务的事务可以包括使用不同线程并行运行的同步和异步调用和活动。要对跨多个线程或流程的事务启用事务跟踪，将在每个组件和操作中插入关联标识符并将其用作事务步骤。然后可以查看所执行的特定操作的相关详细信息以及完成每个操作所花费的时间。

只要在每个被跟踪的节点上启用 CA APM for SOA 和 CA APM for TIBCO BusinessWorks，您还可以跨任何平台组合跟踪业务事务。这样您就能够查看关于事务的详细信息，即使该事务跨多个 JVM 或 CLR 也是如此。

如果来自 TIBCO BusinessWorks 的调用是由不受监控的服务器上的流程执行的，事务跟踪将提供调用不受监控流程的业务流程的执行时间，以便可以确定接收回复所花费的时间，即使未记录不受监控的流程本身执行时间也是如此。事务跟踪还记录是否发生了任何错误。

如何在 Tibco 中使用事务跟踪数据

在 Tibco 业务流程中出现非常高的平均响应时间时（例如 x），您无法使用时间筛选来筛选大于 x 的事务。业务流程通常由多个事务组成，因此，请执行以下步骤来诊断问题区域。

1. 转到“π”窗口，并为该特定业务流程启用筛选。
2. 调用应用程序。

会出现对该业务流程的无数事务跟踪。跟踪是业务流程的执行单元。

3. 找出所有较长的执行单元。深入查看该特定跟踪以获得更多信息。
会显示执行跟踪之间的时差。此时间指示何时挂起业务流程。使用该信息，查明挂起业务流程的频率和时长。
HTTP 请求超时会导致相应的活动显示错误。

了解跨进程事务跟踪的值

跨进程事务跟踪提供有关面向服务的体系结构中的松散耦合服务所执行操作的有价值信息。您可以使用跨进程事务跟踪来确定：

- 业务流程是如何通过组件路由的。
- 在事务期间调用和执行哪些业务流程。
- 事务处理期间所做调用的顺序。
- 处理请求或回复最慢的位置。

启动并查看示例事务跟踪

如果业务流程事务包括对本地或远程计算机上不同 JVM 或 CLR 的调用，并且调用的 JVM 或 CLR 上的代理向同一企业管理器报告调用，则可以运行事务跟踪来查看所有作为业务流程的一部分执行的调用。

您可以以下述方式启动事务跟踪会话：

- 直接通过 SOA 依赖关系视图中的视图节点。
- 通过单击“Workstation”>“新建事务跟踪会话”，从 Workstation 手工启动。

可以配置筛选以控制在事务跟踪会话中捕获的事务，并指定会话的持续时间。如果从依赖关系视图启动事务跟踪，则视图节点类型自动设置默认筛选。如果手工启动新的事务跟踪会话，您可以选择以下 TIBCO BusinessWorks 筛选类型之一：

- 业务流程
- 命名空间
- 操作

例如，要筛选特定 TIBCO BusinessWorks 流程定义的事务，可以选择业务流程筛选并键入完整或部分流程定义名称。

配置筛选并启动事务跟踪会话后，将显示事务跟踪查看器。在事务跟踪查看器中，可以选择某个跟踪来查看事务中所执行的调用的其他相关信息。例如，选择某个跟踪，然后执行下列操作：

- 单击“摘要视图”选项卡查看事务中涉及的组件
- 单击“跟踪视图”选项卡显示事务中所执行的调用以及事务中涉及的每个调用所花费的时间
- 单击“顺序视图”显示调用顺序或事务的流程工作流以及在事务跟踪中花费时间最多的线程。

注意：有关事务跟踪的详细信息，请参阅[在 SOA 环境中使用事务跟踪](#) (p. 85)。有关配置跟踪的信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。有关使用事务跟踪视图和历史数据的详细信息，请参阅《CA APM Workstation 用户指南》。

关于 BusinessWorks 的前端和后端

SOA 中的事务通常涉及对多个系统的同步和异步调用的复杂处理。例如，单个事务可能包括对外部 Web 服务、TIBCO Enterprise Message Service 服务器实例以及数据库的调用。对于 TIBCO BusinessWorks，后端调用可能都是使用不同线程进行处理的，从而导致更难将这些调用识别为属于同一事务。

为了使多个后端调用与事务同一起点能够关联，SOA Extension for TIBCO BusinessWorks 包括下列配置属性：

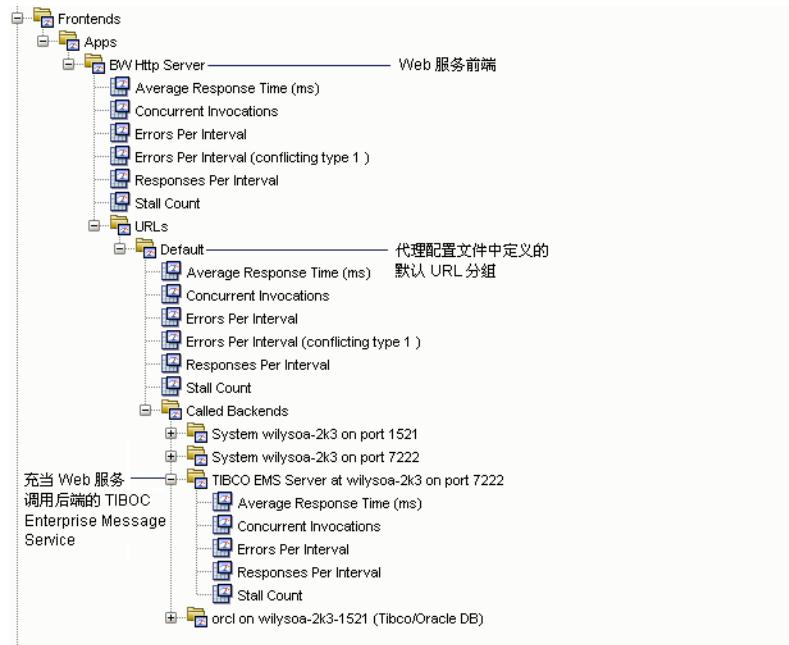
`com.wily.soaextension.tibcwbw.mbb.enabled`

默认情况下，该属性设置为 `true`，以使参与同一事务的后端调用线程包含在“后端”或“调用的后端”节点下，并与适当的前端关联。

前端标识事务的起点。对于 TIBCO BusinessWorks，默认前端是业务流程的流程启动器以及 Web 服务的 TIBCO BusinessWorks HTTP Servlet。

前端度量标准按照 TIBCO BusinessWorks 应用程序名称收集和发布在“前端”节点下。由事务的同一前端启动的所有后端调用均收集在该前端的“调用的后端”节点下。例如，如果流程启动器发出对 Web 服务、数据库和 TIBCO Enterprise Message Service 的调用，则可以在“前端”节点下查看与该流程启动器关联的调用的度量标准。业务流程后端生成的所有错误都将传播到关联的前端节点。

例如，要查看调用 TIBCO Enterprise Message Service 服务器实例的 Web 服务事务的度量标准，可以在调查器中展开“BusinessWorks HTTP 服务器”前端，然后从中展开“调用的后端”节点：



了解对生成的流程的限制

如果业务流程创建了生成的子流程，原始流程通常不会等待生成的子流程运行完成。由于生成子流程的业务流程不等待生成的子流程完成，因此生成的子流程产生的任何度量标准或错误都可能不包括在调用的后端的度量标准或业务流程前端的度量标准中。

如果生成的流程在主营业务流程运行时执行并运行完成，则生成的流程的度量标准和错误可以包括在业务流程的前端中。

禁用多个后端与一个前端的关联

默认情况下，*com.wily.soaextension.tibcobw.mbbs.enabled* 属性设置为 **true**，以使对运行在不同线程上的多个后端系统的调用与启动事务的前端关联。如果要禁用此功能，可以将 *com.wily.soaextension.tibcobw.mbbs.enabled* 属性设置为 **false**。

如果将该属性设置为 **false**，则仅在“调用的后端”节点下收集使用同一线程作为前端执行的后端调用。由于 TIBCO BusinessWorks 使用多个线程进行处理很常见，因此仅当不需要将性能度量标准和错误与其相应的前端关联时，才应将此属性设置为 **false**。

添加自定义 TIBCO BusinessWorks 后端

如果要添加自定义后端，并将自定义后端与相关的前端关联，必须通过定义自定义后端跟踪器来标记自定义后端，然后定义代理的自定义调用的后端跟踪器和调用的后端错误跟踪器。例如，如果定义下列后端跟踪器：

```
SetTracerClassMapping: <CustomBackendTracer>
com.wily.introscope.agent.trace.BackendTracer
com.wily.introscope.probebuilder.validate.ResourceNameValidator
SetTracerParameter: <CustomBackendTracer> nameformatter <CustomNameFormatter>

TraceOneMethodWithParametersIfFlagged: <CustomTracingFlag> <method>
<CustomBackendTracer> "Metric Path"
```

然后可以按照如下所示内容定义 `CalledBackendTracer` 和 `CalledBackendErrorTracer`:

```
SetTracerClassMapping: <customCalledBackendTracer>
com.wily.soaextension.tibcobw.multithread.blame.CalledBackendTracer
com.wily.introscope.probebuilder.validate.ResourceNameValidator
SetTracerParameter: <CustomCalledBackendTracer> nameformatter
<CustomNameFormatter>

SetTracerClassMapping: <customCalledBackendErrorTracer>
com.wily.soaextension.tibcobw.multithread.blame.CalledBackendErrorTracer
com.wily.introscope.probebuilder.validate.MetricNameValidator
SetTracerParameter: <CustomCalledBackendErrorTracer> nameformatter
<CustomNameFormatter>

TraceOneMethodWithParametersIfFlagged: <customTracingFlag> <method>
CustomCalledBackendTracer "Metric Path"
TraceOneMethodWithParametersIfFlagged: <customTracingFlag> <method>
CustomCalledBackendErrorTracer "Metric Path:Errors Per Interval"
```

注意：有关配置自定义跟踪器、定义名称格式化程序以及标记自定义前端和后端的详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

自定义度量标准老化和删除

度量标准老化将定期从代理的内存缓存中删除死度量标准。死度量标准是指在给定时间内没有报告任何新数据的度量标准。这样有助于代理提高性能并避免收集过多度量标准，导致系统无法处理。

但是，如果有长期运行的事务或业务流程，则可能需要避免某些度量标准的老化。例如，如果有运行超过 12 个小时的业务流程，可以使用类似下列内容的配置属性避免业务流程的“并发调用”度量标准老化。

```
introscope.agent.metricAging.metricExclude.ignore.2=Tibco\|Processes\|*.*:Concurrent_Invocations
```

还可以使用其他代理属性来自定义度量标准老化。例如，可以设置属性以自定义度量标准变成删除候选所需的时间，或每个时间间隔中要检查的度量标准数。

注意：有关度量标准老化和配置代理属性的详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

从度量标准老化中排除 TIBCO BusinessWorks 度量标准

1. 导航到 `<Agent_Home>/core/config` 目录
2. 在文本编辑器中打开 `IntroscopeAgent.profile` 文件。
3. 找到“代理度量标准老化”部分并取消注释以下属性：
`introscope.agent.metricAging.metricExclude.ignore.2=Tibco\|Processes\|*.*:C`
`oncurrent Invocations`

如果有运行超过 12 个小时的业务流程，建议从度量标准老化中排除“并发调用”。

配置 TIBCO BusinessWorks 的关联跟踪

跨进程事务跟踪需要代理插入一个关联标识符，该标识符可从一个进程传递到另一个进程。代理可将该关联标识符插入 SOAP 或 HTTP 头。默认情况下，SOA Extension for TIBCO BusinessWorks 配置为支持 SOAP 和 HTTP 头中的关联信息的传送。有关如何使用关联信息启用跨进程事务跟踪的更多信息，请参阅“[了解如何链接事务段 \(p. 86\)](#)”。有关配置关联跟踪的属性的更多信息，请参阅[配置关联跟踪 \(p. 101\)](#)。

第9章：监控 TIBCO Enterprise Message Service

通过启用整个企业中系统间的同步和异步通信，TIBCO Enterprise Message Service 提供 SOA 基础架构的关键组件。通过 SOA Extension for TIBCO Enterprise Message Service (EMS)，可以监控 TIBCO EMS 服务器的许多关键元素，包括队列、主题和高级组件（如网桥、多播通道和路由）。

本节介绍可用于监控和分析 TIBCO EMS 环境的运行状况和操作的显示板和度量标准。

此部分包含以下主题：

- [关于 TIBCO Enterprise Message Service \(p. 161\)](#)
- [如何安装 SOA Extension for TIBCO EMS \(p. 162\)](#)
- [运行独立代理安装程序 \(p. 162\)](#)
- [准备 TIBCO EMS 服务器以进行监控 \(p. 164\)](#)
- [配置 TIBCO EMSMonitor 代理 \(p. 165\)](#)
- [配置筛选以进行选择性监控 \(p. 168\)](#)
- [定义监控级别 \(p. 170\)](#)
- [创建加密密码 \(p. 174\)](#)
- [使用 SSL 连接到 TIBCO EMS 服务器实例 \(p. 175\)](#)
- [启动 EMSMonitor 代理 \(p. 176\)](#)
- [启用企业管理器扩展 \(p. 178\)](#)
- [使用显示板监控 TIBCO EMS \(p. 179\)](#)
- [了解并查看 TIBCO EMS 度量标准 \(p. 180\)](#)
- [查看默认的 TIBCO EMS 度量标准组 \(p. 202\)](#)
- [查看默认的 TIBCO EMS 报警 \(p. 203\)](#)
- [代理配置属性汇总 \(p. 204\)](#)

关于 TIBCO Enterprise Message Service

TIBCO Enterprise Message Service 是基于标准的消息层，通过提供跨多种平台和应用程序技术的兼容 JMS 的通信，可以充当面向服务的体系结构的主干。-

由于安全可靠地发送消息是提供业务服务和完成业务事务的关键部分，因此 SOA Extension for TIBCO EMS 提供了用于监控 TIBCO Enterprise Message Service 的关键组件和性能度量标准的单独代理。

用于监控 TIBCO Enterprise Message Service 的代理是独立代理，而不是核心 Java 或 .NET 代理的扩展功能。代理 *EMSMonitor* 不检测 Java 类，而是使用 TIBCO EMS Admin API 收集 TIBCO EMS 组件的性能信息，然后将这些信息报告给企业管理器。

由于 *EMSMonitor* 代理是独立代理，因此它是作为单独的软件包分发的，需要自己的 *IntroscopeAgent.profile* 和独有的配置步骤。例如，必须配置要监控的 EMS 服务器的连接信息，并添加筛选以自定义要收集其度量标准的组件。

配置相应的代理属性后，可以使用 *EMSMonitor* 代理监控本地和远程的 TIBCO EMS 组件。

如何安装 SOA Extension for TIBCO EMS

EMSMonitor 代理不依赖于核心代理。可以在运行 TIBCO Enterprise Message Service 的一台或多台计算机上安装并配置独立于其他任何 CA Introscope® 组件的 *EMSMonitor* 代理。

添加 SOA Extension for TIBCO EMS 包括下列高级步骤： -

1. 确认已安装受支持的 TIBCO Enterprise Message Service 版本。
注意：有关系统要求，请参阅《兼容性指南》。
2. 确认已在您的环境中安装企业管理器和 Workstation。确认具有 *EMSMonitor* 代理将向其发送数据的企业管理器的连接信息。
3. [运行独立代理安装程序 \(p. 162\)](#)或使用响应文件将所需代理文件添加到环境中。
4. [为监控 EMS 服务器做准备。 \(p. 164\)](#)
5. [配置 TIBCO EMSMonitor 代理 \(p. 165\)](#)配置文件以定义连接和监控属性。
6. [启用适用于 TIBCO BusinessWorks 的企业管理器扩展 \(p. 178\)](#)。

运行独立代理安装程序

使用独立代理安装程序可以安装不依赖于核心 Java 或 .NET 代理的独立代理。如果要启用对 TIBCO Enterprise Message Service 的监控，您可以使用独立代理安装程序将与代理相关的文件添加到环境中，并配置代理与企业管理器的连接。独立代理安装程序会提取所需的文件并将其置于适当的位置，以供您根据需要进行修改以完成代理配置。

运行独立代理安装程序以监控 TIBCO EMS

1. 针对您的操作环境启动适当的独立代理安装程序。
2. 在“简介”页面上，单击“下一步”。
3. 选择要安装的监控软件包，然后单击“下一步”。例如，选择“SOA Extension For TIBCO Enterprise Message Service”选项以启用对 TIBCO EMS 的监控。
4. 对于安装目录，单击“下一步”接受默认位置，或单击“浏览”指定其他位置。
5. 对于企业管理器连接设置，指定要向其发送数据的代理的企业管理器主机名和端口号，然后单击“下一步”。
6. 检查您的设置摘要，然后单击“安装”开始进行安装。
7. 安装完成后，单击“完成”关闭独立代理安装程序。

使用响应文件进行静默安装

如果不希望以交互方式运行独立代理安装程序，可以编辑独立示例响应文件来安装代理文件。使用该方法可以在静默模式下启用 SOA Extension for TIBCO EMS。

遵循这些步骤：

1. 打开 SampleResponseFile.StandaloneAgentPP.txt 文件，该文件与独立代理安装程序位于同一目录中。
2. 编辑 SampleResponseFile.StandaloneAgentPP.txt 文件以将 shouldInstallTibcoEMS 属性设置为 true，从而向代理添加 SOA Extension for TIBCO Enterprise Message Service。例如：
`shouldInstallTibcoEMS=true`
3. 保存 SampleResponseFile.StandaloneAgentPP.txt 文件。
4. 在命令行中输入相应的命令以调用安装程序。

注意：有关以静默模式安装代理的详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

手工提取安装存档

如果没有访问独立代理安装程序或独立响应文件的权限，您也许能够下载适用于您的操作环境的独立安装存档。您可以从 [CA Support](#) 上的 CA APM 软件下载区域下载 CA APM 产品。下载完毕后，可以使用适用于操作环境的命令从存档中手工提取文件。例如，使用 UNIX 计算机上的 `tar` 命令：

```
tar -xvf IntroscopeStandaloneAgentPPInstaller<版本>unix.tar  
r
```

准备 TIBCO EMS 服务器以进行监控

必须先准备 `EMSMonitor` 代理要使用的用户帐户，并验证监控所需的库在本地是否可用，然后才能监控 TIBCO Enterprise Message Service。

遵循这些步骤：

1. 使用具有 `change-admin-acl` 和 `change-user` 权限的帐户登录 EMS 服务器。例如，使用 `admin` 帐户登录。
2. 标识或创建具有 `EMSMonitor` 代理的 `view-all` 权限的用户帐户。例如，要创建新用户以运行 `EMSMonitor` 代理，请在“EMS 管理工具”中运行以下命令：
创建用户 `wilyemsuser`
向管理员 `wilyemsuser` 授予 `view-all` 权限
3. 验证 `<TIBCO_EMSS_HOME>/lib` 目录中是否存在以下 TIBCO EMS 库：

```
jms.jar  
tibjms.jar  
tibjmsadmin.jar
```

如果要监控远程服务器实例，运行 `EMSMonitor` 代理的计算机应配置为 EMS 客户端。通常，这意味着，`jms.jar`、`tibjms.jar` 和 `tibjmsadmin.jar` 文件可用且已添加到 `EMSMonitor` 代理所在计算机上的类路径中。

注意：有关设置 EMS 客户端应用程序并将这些文件添加到类路径的信息，请参阅《*TIBCO Enterprise Message Service 用户指南*》。

适用于 EMS 5.x 或更高版本：如果必须使用安全套接字层 (SSL) 连接将代理连接到 EMS 服务器，则本地计算机上的以下库必须可供 EMS 5.x 或更高版本使用：

```
tibcrypt.jar  
slf4j-api-1.4.2.jar  
slf4j-simple-1.4.2.jar
```

适用于 EMS 服务器 4.4.x: 只有 *tibcrypt.jar* 库是必需的。

注意: 有关配置适用于 EMS 的 SSL 通信的详细信息, 请参阅《*TIBCO Enterprise Message Service 用户指南*》。

适用于 EMS 服务器 4.4.x 和 5.x: 如果要监控包含 EMS 4.4.x 和 5.x 的环境, 请确保本地计算机上所需的所有 .jar 文件都是最新可用的 .jar 文件。例如, 如果环境中包含 EMS 4.4.x 和 5.x, 请确保 EMSMonitor 代理使用 5.x .jar 文件。

配置 TIBCO EMSMonitor 代理

运行独立代理安装程序后, 应该会看到 *TibcoEMSMonitor* 目录。

TibcoEMSMonitor 目录提供了以下用于监控 TIBCO Enterprise Message Service 的文件和目录:

ext 目录

包含 *Supportability-Agent.jar* 文件。*Supportability-Agent.jar* 文件是一个用于在日志文件中记录系统属性、配置文件和其他信息的扩展, 如果需要 EMSMonitor 代理支持, 可以将这些日志文件提供给 CA Support。

lib 目录

包含打包为 .jar 文件 (*Agent.jar* 和 *jline-0.9.9.jar*) 的必需的 EMSMonitor 代理库。

properties 目录

包含 *IntroscopeAgent.profile*、*TibcoEMSMonitor.properties* 和 *MonitoringLevel.xml* 文件。

- 您可以使用 *IntroscopeAgent.profile* 来配置到企业管理器的连接。
- 您可以使用 *TibcoEMSMonitor.properties* 文件来配置用于连接到本地和远程 TIBCO EMS 服务器实例的参数, 并自定义监控级别和要收集其度量标准的组件。
- 您可以使用 *MonitoringLevel.xml* 文件来自定义组成最低、建议和完全监控级别集的度量标准。

EMSMonitor.jar 文件

提供代理用于监控 EMS 服务器实例的类。

emsPwdEncryptor 文件

提供密码加密脚本 (*emsPwdEncryptor.bat* 或 *emsPwdEncryptor.sh*), 可使您能够加密连接到 EMS 服务器所用的 EMS 用户密码和 SSL 客户端证书密码。

EMSMonitor 文件

在您准备开始监控 EMS 服务器实例时，提供启动脚本（EMSMonitor.bat 或 EMSMonitor.sh）以运行 EMSMonitor 代理。

在 UNIX 平台上，您可以使用 EMSMonitor.sh 脚本来启动、停止、重新启动或检查 EMSMonitor 代理的状态。在 Windows 上，您可以使用 EMSMonitor.bat 或 Windows 服务来启动代理并停止 EMSMonitor 代理。

通过 *TibcoEMSMonitor* 目录中的文件，您可以配置 *EMSMonitor* 代理连接属性、监控属性和安全通信属性。

配置基本连接属性

EMSMonitor 代理将来自 TIBCO EMS 服务器的数据报告给企业管理器。要启用对度量标准的收集和报告，必须指定 *EMSMonitor* 代理向其报告的企业管理器的连接信息，并标识代理从中收集数据的服务器列表。

要配置 *EMSMonitor* 代理连接属性，请执行下列操作：

1. 在文本编辑器中打开 *IntroscopeAgent.profile* 文件以指定企业管理器连接参数。例如：

```
introscope.agent.enterprisemanager.transport.tcp.host.DEFAULT=mercury  
introscope.agent.enterprisemanager.transport.tcp.port.DEFAULT=5001
```

2. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件，并将 *ems.server.list* 属性设置为要监控的 EMS 服务器实例的名称。例如：

```
ems.server.list = tibco_ems_srv01, tibco_ems_srv02
```

您可以使用在该列表中选择的所有名称。不要求与要监控的服务器实例的实际名称匹配。但是，列表中的名称必须与用于定义附加属性的服务器实例名称匹配。例如，如果有两个名为 *NewYork* 的服务器实例，可以在 *ems.server.list* 属性中使用别名 *newyork1* 和 *newyork2*，然后使用别名定义这两个服务器实例的其余属性。例如：

```
ems.server.list = newyork1, newyork2  
newyork1.host = catsmcn01  
newyork1.port = 6001
```

然后，每个实例的 URL 会附加到调查器树中显示的服务器实例名称中，使您能够监控并唯一标识这两个实例。

3. 配置用于连接到 *ems.server.list* 属性中指定的每个服务器实例的主机名和端口。例如：

```
tibco_ems_srv01.host=localhost  
tibco_ems_srv01.port=7222
```

```
tibco_ems_srv02.host=vespa  
tibco_ems_srv02.port=7222
```

4. 运行 *emsPwdEncryptor* 实用工具，以生成要用于连接到 EMS 服务器的用户帐号的加密密码。

有关创建或选择用于连接到 EMS 服务器的用户帐号的信息，请参阅[为监控 EMS 服务器做准备 \(p. 164\)](#)。有关使用 *emsPwdEncryptor* 实用工具的信息，请参阅[创建加密密码 \(p. 174\)](#)。

配置服务器的轮询时间间隔

EMSMonitor 代理会定期轮询每个服务器实例，以便检索最新状态和配置信息。您可以使用配置属性控制这些查询的频率。

要配置 *EMSMonitor* 代理查询服务器的频率，请执行下列操作：

1. 配置轮询每个服务器实例以收集与状态相关的度量标准的时间间隔。

您指定的时间间隔控制查询 EMS 服务器实例获取状态信息的频率。例如，要每隔 60 秒查询一次 *tibco_ems_srv01* 实例，每隔 120 秒查询一次 *tibco_ems_srv02* 实例，将需要设置以下属性：

```
tibco_ems_srv01.delaytime=60  
tibco_ems_srv02.delaytime=120
```

如果不指定时间间隔，默认情况下会每隔 60 秒刷新一次度量标准。

2. 配置刷新每个服务器实例的配置相关静态度量标准的时间间隔。

由于配置属性不会经常更改，因此 *EMSMonitor* 代理在启动时会收集一次值，然后在值变为 *<ServerInstance>.report.static.freq* 属性乘以 *<ServerInstance>.delaytime* 属性后进行轮询以获取配置度量标准。如果 *<ServerInstance>.delaytime* 属性设置为 60 秒，*<ServerInstance>.report.static.freq* 属性设置为 20，则系统每 1200 秒（20 分钟）检查一次静态度量标准。

例如：

```
tibco_ems_srv01.report.static.freq=20  
tibco_ems_srv02.report.static.freq=20
```

如果将该属性设置为 0，则 *EMSMonitor* 代理会在启动时报告所有配置度量标准，但随后不会报告对其所做的任何更改。

配置筛选以进行选择性监控

默认情况下，**EMSMonitor** 代理会收集 *ems.server.list* 属性中指定的服务器实例上的所有队列和主题的度量标准。如果要选择性地监控队列和主题，可以配置筛选以指定您感兴趣的队列和主题。还可以配置筛选以启用对高级组件（如网桥、多播通道和路由）的监控，并使用正则表达式来确定要收集其度量标准的高级组件的子集。

为队列和主题配置筛选

EMS 服务器既支持静态队列和主题，也支持动态队列和主题。静态队列和主题由 **EMS** 管理员创建和管理，将保留在服务器上，即使未被使用也是如此。动态队列和主题（包括临时动态队列）由 **EMS** 客户端根据需要创建和销毁，仅在服务器上保留较短时间。

默认情况下，**EMSMonitor** 代理仅监控静态队列和主题以防止不必要的开销。但是，如果要监控动态队列或动态主题，可以定义筛选以将其包含在内。也可以使用正则表达式配置筛选，以便识别要收集其度量标准的特定队列和主题。

要启用对动态队列或主题的监控，请执行下列操作：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 **EMS** 服务器实例的名称。
3. 为要监控其动态对象的每个服务器实例配置动态队列或动态主题的包含筛选。例如，启用对 **EMS** 服务器实例 *tibco_ems_srv01* 上的动态队列和动态主题的监控：

```
tibco_ems_srv01.queue.filter.include.dynamic=true  
tibco_ems_srv01.topic.filter.include.dynamic=true
```

要使用正则表达式监控特定的队列或主题，请执行下列操作：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 **EMS** 服务器实例的名称。
3. 如果需要，使用正则表达式为每个服务器实例配置队列和主题筛选。例如：

```
tibco_ems_srv01.queue.filter.includeonly.regex=sample.*  
tibco_ems_srv01.topic.filter.includeonly.regex=sample.*  
  
tibco_ems_srv02.queue.filter.includeonly.regex=test.*  
tibco_ems_srv02.topic.filter.includeonly.regex=test.*
```

如果不指定筛选，则会收集所有队列和主题（包括动态和临时队列和主题，如果属性已配置为将其包含在内）的度量标准。如果针对要包含的队列指定正则表达式，则只会监控与该正则表达式匹配的队列。如果为要包括的主题指定正则表达式，则只监控与该正则表达式匹配的主题。

定义要包含的高级组件

默认情况下，**EMSMonitor** 代理不会报告网桥、通道或路由的度量标准。要启用对这些高级组件的监控，需要在 *TibcoEMSMonitor.properties* 文件中指定包含筛选。启用对要包含的高级组件的监控后，可以使用正则表达式筛选来进一步细化要收集其度量标准的高级组件列表。

要启用对网桥、通道和路由的监控，请执行下列操作：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 EMS 服务器实例的名称。
3. 为要监控的每个服务器实例上的每个高级组件配置包含筛选。例如，如果要监控服务器实例 *tibco_ems_srv1* 上的网桥、通道和路由，请向该文件添加以下包含筛选：

```
tibco_ems_srv1.monitor.bridges=true  
tibco_ems_srv1.monitor.channels=true  
tibco_ems_srv1.monitor.routes=true
```

定义高级组件的正则表达式筛选

如果已经定义包含筛选来启用对网桥、通道或路由的监控，还可以配置正则表达式筛选，以便识别要收集其度量标准的特定网桥、通道或路由。

如果定义组件的包含筛选而没有定义正则表达式筛选，则会监控该类型的所有组件。例如，如果启用对网桥的监控，但没有添加网桥的正则表达式筛选，则会收集所有网桥的度量标准。如果未定义组件的包含筛选，则会忽略该组件的任何正则表达式筛选。

要定义网桥、通道或路由的正则表达式筛选，请执行下列操作：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 EMS 服务器实例的名称。
3. 为要监控的每个服务器实例上的每个高级组件配置正则表达式筛选。例如，如果已经启用对服务器实例 *tibco_ems_srv1* 上的网桥、通道和路由的监控，请向该文件添加以下包含筛选：

```
<ServerInstance>.bridge.filter.includeonly.regex  
<ServerInstance>.channel.filter.includeonly.regex  
<ServerInstance>.route.filter.includeonly.regex
```

对于每个筛选，使用正则表达式来识别要监控的组件的子集。例如：

```
tibco_ems_srv1.bridge.filter.includeonly.regex=new.*  
tibco_ems_srv1.channel.filter.includeonly.regex=.*bulletin  
tibco_ems_srv1.route.filter.includeonly.regex=.*
```

正则表达式筛选仅适用于已将包含筛选配置设置为 *true* 的高级组件。

EMS 名称中特殊字符的替换

TIBCO Enterprise Message Service 允许您在所有组件名称中包含特殊字符，如冒号 (:) 或管道 (|) 字符。但是，冒号 (:) 和管道 (|) 字符在 CA Introscope® 中是保留字符，在组件名称中会自动替换为下划线字符 (_)。例如，如果您的队列名称为 *Queue:WorkInProgress*，则 EMS 代理会将冒号字符替换为下划线，并在名为 *Queue_WorkInProgress* 的节点下报告此队列的度量标准。

如果您具有名称类似的队列，如 *Queue:WorkInProgress* 和 *Queue /WorkInProgress*，替换特殊字符会导致名称重复。由于替换冒号和管道字符后队列具有相同的名称，因此调查器仅会显示名为 *Queue_WorkInProgress* 的队列的一组度量标准。

该特殊字符限制适用于所有 EMS 组件。如果要定义筛选以配置监控，请考虑替换冒号或管道字符对组件名称的影响。

定义监控级别

EMSMonitor 代理可以收集每个 TIBCO EMS 服务器实例的大量度量标准。要控制收集并报告给企业管理器的度量标准数，可以定义每个 TIBCO EMS 服务器实例的监控级别。对于每个服务器实例，可以分别定义服务器、队列和主题度量标准的监控级别。

使用默认监控级别定义

为了控制各个 EMS 服务器及其组件的度量标准数, *EMSMonitor* 代理支持服务器度量标准、队列度量标准和主题度量标准的以下监控级别:

最低要求

仅收集监控服务器、队列或主题所必需的度量标准。默认情况下, 为最低监控级别定义的度量标准需要在预配置的显示板和“概览”选项卡中显示信息。

如果要尽可能减小 EMS 服务器实例上服务器、队列或主题度量标准的开销, 可以使用该监控级别。

推荐

收集所有的最低级别度量标准以及其他度量标准, 如服务器的容错度量标准、队列的入站和出站消息统计信息或主题的多播相关度量标准。

如果您可以接受开销方面的小幅增加以更好地了解服务器、队列或主题的性能和操作, 可以使用该监控级别。

完全

收集并报告服务器、队列或主题的所有可用度量标准, 包括最低度量标准和建议度量标准。

如果要深入了解服务器、队列或主题的性能和操作, 或者如果您不介意性能开销, 可以使用该监控级别。

监控级别指定要为每个服务器实例上的服务器、队列或主题收集的度量标准集, 可以为每个服务器实例上的每个组件单独设置。

例如, 要收集服务器实例 *tibco_ems_server01* 上的所有服务器度量标准、最低队列度量标准和建议主题度量标准, 可以使用以下属性设置这些组件的监控级别:

```
tibco_ems_server01.monitoring.level = full  
tibco_ems_server01.queue.monitoring.level = minimum  
tibco_ems_server01.topic.monitoring.level = recommended
```

然后, 您可以为其他服务器实例上的服务器、队列和主题监控级别配置这些属性。例如:

```
tibco_ems_server02.monitoring.level = minimum  
tibco_ems_server02.queue.monitoring.level = recommended  
tibco_ems_server02.topic.monitoring.level = recommended
```

要设置 TIBCO EMS 服务器实例的监控级别，请执行下列操作：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 EMS 服务器实例的名称。
3. 配置每个服务器实例的监控级别属性
<ServerInstance>.monitoring.level。例如，对于 EMS 服务器实例 *tibco_ems_srv01*，将监控级别设置为 “minimum”，对于 EMS 服务器实例 *tibco_ems_srv02*，将监控级别设置为 “full”：
tibco_ems_srv01.monitoring.level=minimum
tibco_ems_srv02.monitoring.level=full

设置 TIBCO EMS 队列的监控级别：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 EMS 服务器实例的名称。
3. 为每个服务器实例配置队列的监控级别属性
<ServerInstance>.queue.monitoring.level。例如，对于 EMS 服务器实例 *tibco_ems_srv01* 上的队列，将监控级别设置为 “minimum”，对于 EMS 服务器实例 *tibco_ems_srv02* 上的队列，将监控级别设置为 “recommended”：
tibco_ems_srv01.queue.monitoring.level=minimum
tibco_ems_srv02.queue.monitoring.level=recommended

您设置的监控级别适用于指定的服务器实例上的所有静态和动态队列。

设置 TIBCO EMS 主题的监控级别：

1. 在文本编辑器中打开 *TibcoEMSMonitor.properties* 文件。
2. 复查要监控的 EMS 服务器实例的名称。
3. 为每个服务器实例配置主题的监控级别属性
<ServerInstance>.topic.monitoring.level。例如，对于 EMS 服务器实例 *tibco_ems_srv01* 上的主题，将监控级别设置为 *full*，对于 EMS 服务器实例 *tibco_ems_srv02* 上的主题，将监控级别设置为 *recommended*：
tibco_ems_srv01.topic.monitoring.level=full
tibco_ems_srv02.topic.monitoring.level=recommended

所设监控级别将应用到指定服务器实例上的所有静态和动态主题。

修改默认监控级别定义

如果要更改与 “minimum”、“recommended” 或 “full” 监控级别关联的度量标准，可以通过编辑 *MonitoringLevel.xml* 文件来完成此操作。

MonitoringLevel.xml 文件列出了所有可用于 EMS 服务器实例、队列和主题的度量标准。该文件还定义了每个度量标准的监控级别。您可以在该文件中更改监控级别，以便更改在 *TibcoEMSMonitor.properties* 文件中将服务器、队列和主题的监控级别设置为“minimum”、“recommended”或“full”时要收集的度量标准。

您无法使用该文件配置与网桥、通道或路由关联的度量标准的监控级别。可以使用筛选自定义对网桥、通道或路由的监控。有关使用筛选自定义对网桥、通道和路由的监控的更多信息，请参阅[配置筛选以进行选择性监控 \(p. 168\)](#)。

要更改单个度量标准的监控级别，请执行下列操作：

1. 在文本编辑器中打开 *MonitoringLevel.xml* 文件。
2. 找到要修改其度量标准设置的组件的 *<MetricGroup>* 部分。例如，如果要更改服务器度量标准的监控级别，请找到 *<MetricGroup name="Server">* 部分。

所有可用于监控 EMS 服务器实例的度量标准以及定义为每个度量标准名称属性的度量标准的监控级别将一并列出。例如，将监控级别设置为“minimum”时，默认情况下会收集“队列计数”和“主题计数”度量标准：

```
<Metric level="Minimum">队列计数</Metric>
<Metric level="Minimum">主题计数</Metric>
```

3. 找到要修改其监控级别的度量标准，并将该度量标准的 *level* 属性修改为新的监控级别。

例如，将“队列计数”和“主题计数”的监控级别从“minimum”更改为“recommended”：

```
<Metric level="Recommended">队列计数</Metric>
<Metric level="Recommended">主题计数</Metric>
```

如果将任何度量标准的监控级别从“minimum”更改为“recommended”或“full”，*EMSMonitor* 代理会在其日志文件中记录警告并显示警告消息。该代理不会对其他类型的更改发出警告。

您还应该记住，更改最小度量标准集可能会阻止某些数据显示或包含在图表和显示板中。在将度量标准从“minimum”更改为“recommended”或“full”之前，应先检查该度量标准是用于任何预配置或自定义视图还是显示板。

4. 使用启动脚本重新启动 *EMSMonitor* 代理，以便使对监控级别所做的更改生效。

做出此更改并重新启动代理后，将仅为监控级别设置为“recommended”的服务器实例收集“队列计数”和“主题计数”度量标准。

<*MetricGroup name*> 和 <*Metric level*> 属性不区分大小写。但是，度量标准名称在 *MonitoringLevel.xml* 文件中区分大小写。

代理使用 *MonitoringLevel.xml* 文件来确定每个监控级别的度量标准。如果某个度量标准未列出或已从该文件中删除，代理将无法收集或报告该度量标准的信息。

如果 *MonitoringLevel.xml* 文件已损坏或 *EMSMonitor* 代理无法读取该文件，代理将记录错误消息并报告启动时发现的度量标准。

创建加密密码

要使 *EMSMonitor* 代理连接到您指定的 EMS 服务器，该代理必须能够提供有效的用户凭据以进行身份验证。如果在客户端和 EMS 服务器之间使用安全套接字层 (SSL) 连接，且代理连接到已配置为验证客户端安全证书的任何 EMS 服务器，则该代理还必须能够提供签名证书以进行验证。

要加密和存储用户密码或客户端证书路径和密码，请执行下列操作：

1. 在文本编辑器中打开 *emsPwdEncryptor* 脚本，并将 *JAVA_HOME* 环境变量设置为相应的目录。例如：
`set JAVA_HOME=C:\Java\jdk1.5.0_10`
2. 运行 *emsPwdEncryptor* 脚本。

对于已为 *TibcoEMSMonitor.properties* 文件中的 *ems.server.list* 属性列出的每个 EMS 服务器实例，系统会提示您指定用户名和密码。

- 要创建服务器实例的加密密码，请输入 *y*，然后输入 EMS 用户名和密码。
- 要跳过服务器实例加密密码的创建，请输入 *n*。

完成用于连接到每个实例的用户名和密码的输入后，用户名和加密版本的密码会写入到 *TibcoEMSMonitor.properties* 文件中。EMS 服务器必须能够验证指定的用户名和密码以使连接成功。

3. 当系统提示您配置安全套接字层 (SSL) 属性时，输入 *y* 或 *n*。

如果使用的是 SSL 连接，且任何 EMS 服务器已配置为验证客户端的安全证书，则输入 `y`。然后，可以指定客户端证书路径和用于读取客户端证书的密码。

完成信息输入后，`client.identity` 和加密版本的 `ssl.password` 属性会写入到 `EMSMonitor` 代理的 `TibcoEMSMonitor.properties` 文件中：

- `client.identity` 属性指定了 EMS 服务器可用于验证 `EMSMonitor` 代理身份的证书的路径。例如：
`client.identity=C:/Tibco/TibcoEMSMonitor/certs/client_identity.p12`
- `ssl.password` 属性指定了客户端安全证书的加密密码。

使用 SSL 连接到 TIBCO EMS 服务器实例

如果 TIBCO EMS 配置为将安全套接字层 (SSL) 协议用于网络连接，则 `EMSMonitor` 代理必须配置为使用 SSL 连接到 TIBCO EMS 服务器。将 `EMSMonitor` 代理配置为使用 SSL 包括设置 `TibcoEMSMonitor.properties` 文件中的多个属性。某些属性必须单独对每个 EMS 服务器实例进行设置。其他属性可以一次性对所有 EMS 服务器实例进行设置。

要定义 TIBCO EMS 服务器实例的 SSL 连接信息，请执行下列操作：

1. 在文本编辑器中打开 `TibcoEMSMonitor.properties` 文件。
2. 复查要监控的 EMS 服务器实例的名称。
3. 配置 `<ServerInstance>.ssl.connection` 属性以启用或禁用每个服务器实例的加密通信。该属性需要您指定服务器实例的名称。
例如，将代理配置为使用 SSL 连接到 EMS 服务器实例
`tibco_ems_srv01:`
`tibco_ems_srv01.ssl.connection=enable`
4. 配置 `<ServerInstance>.verify.host` 属性以指定 `EMSMonitor` 代理是否必须验证 EMS 服务器的证书。

该属性需要您指定服务器实例的名称。例如，代理连接到 EMS 服务器实例 `tibco_ems_srv01` 时需要验证：

```
tibco_ems_srv01.verify.host=true
```

如果将该属性设置为 `true`，代理将根据为 `trusted.certificates` 属性定义的列表验证 EMS 服务器的安全证书。

5. 配置 `trusted.certificates` 属性以指定 `EMSMonitor` 代理用于验证服务器证书的受信任证书逗号分隔列表。例如：
`trusted.certificates=C:/Tibco/wily/TibcoEMSMonitor/certs/tbx_root.cert.pem`

如果 `verify.host` 属性设置为 `true`，且适用于所有使用 SSL 的 EMS 服务器实例，则该属性是必需的。

6. 配置 `<ServerInstance>.verify.hostname` 属性以指定代理是否应验证服务器证书的通用名 (CN) 字段。

`tibco_ems_srv01.verify.hostname=true`

如果将该属性设置为 `true`, 代理会将连接主机的名称或在 `<ServerInstance>.expected.name` 属性中指定的名称与服务器证书中的通用名 (CN) 字段的值进行比较。如果名称不匹配, 代理将拒绝连接。

如果将该属性设置为 `false`, 代理会建立与服务器的 SSL 连接, 但不会验证服务器的名称。

7. 配置 `<ServerInstance>.expected.hostname` 属性以指定 EMSMonitor 代理用于与服务器证书中通用名 (CN) 字段的值进行比较的名称。

`tibco_ems_srv01.expected.hostname=tbxserver`

8. 配置 `cipher.suites` 属性以指定密码套件逗号分隔列表, EMSMonitor 代理可使用该列表加密与启用 SSL 的 EMS 服务器之间的通信。-

`cipher.suites` 属性是可选属性。EMSMonitor 代理可以使用要监控的 EMS 服务器所支持的任何加密软件包。如果设置该属性, 它将适用于所有启用 SSL 的 EMS 服务器实例。

您可以将标准算法名称或 OpenSSL 名称用于密码套件。例如, 如果要使用 OpenSSL 名称, 可以将 `cipher.suites` 属性设置为 `RC4-MD5`; 如果要使用标准名称来指代同一密码套件, 可以将该属性设置为 `SSL_RSA_WITH_RC4_128_MD`。

例如:

`cipher.suites=RC4-MD5,RC4-SHA`

9. 如果已将 EMS 服务器配置为验证客户端的安全证书, 则运行 `emsPwdEncryptor` 实用工具以设置客户端证书的路径和加密密码。

有关使用 `emsPwdEncryptor` 实用工具的信息, 请参阅[创建加密密码 \(p. 174\)](#)。

如果不需 EMS 服务器验证客户端证书, 则可以跳过该步骤。

启动 EMSMonitor 代理

配置完连接属性后, 可以配置其他属性或启动代理以开始监控 EMS 服务器实例。如果要使用默认的监控属性, 您可以:

- 从启动脚本启动代理
- 将代理配置为以 Windows 服务形式运行

使用启动脚本启动 EMSMonitor

在 UNIX 计算机上，可以使用 `EMSMonitor.sh` 脚本来启动和停止 EMSMonitor 代理。在 Windows 上，可以使用 `EMSMonitor.bat` 脚本或配置 EMSMonitor 代理，作为 Windows 服务运行。

遵循这些步骤：

1. 在文本编辑器中打开 EMSMonitor 启动脚本。
2. Set the TIBCO_EMS_HOME environment variable to the TIBCO Enterprise Message Service installation directory. 例如：

```
set TIBCO_EMS_HOME=C:\tibco\ems\5.1
```
3. 将 JAVA_HOME 环境变量设置为适当的目录。例如：

```
set JAVA_HOME=C:\Java\jdk1.5.0_10
```

适用于 JRE 1.5 或更高版本： 监控 TIBCO Enterprise Message Service 需要 JRE 1.5 或更高版本。
4. 运行 EMSMonitor 脚本。如果在 UNIX 计算机上运行 EMSMonitor，该脚本将支持以下命令行参数：

```
EMSMonitor.sh [start|stop|restart|status]
```

要启动代理的新实例，请使用 `start` 选项。例如，在 UNIX 计算机上：
`./EMSMonitor.sh start`

以 Windows 服务形式运行 EMSMonitor 代理

如果在 Windows 上安装 *EMSMonitor* 代理，可以选择将代理作为 Windows 服务运行。将代理作为 Windows 服务运行具有以下优点：

- 代理服务可以在主机计算机启动或关闭时自动启动或停止。
- 代理可以作为后台进程运行，而不是在控制台中运行，从而使其不易被篡改和进行未经授权的访问。
- 即使用户注销了当前会话，代理也可以继续运行。

以 Windows 服务形式运行 EMSMonitor 代理：

1. 安装并配置 EMS 服务器、*IntroscopeAgent.profile* 和 *TibcoEMSMonitor.properties* 文件，如前几节中所述。
2. 确认 `JAVA_HOME` 环境变量已设置为适当的 JVM，且已配置为系统变量，而不是用户变量。
3. 在文本编辑器中打开
`TibcoEMSMonitor\Windows Service\jsw-3.2.3\conf\wrapper.conf` 文件。

4. Set the *TIBCO_EMS_HOME* environment variable to the TIBCO Enterprise Message Service installation directory. 例如:
`set TIBCO_EMS_HOME=C:\tibco\ems\5.1`
5. 保存 *wrapper.conf* 文件。
6. 打开命令窗口，运行 *TibcoEMSMonitor\Windows Service\RegisterEMSMonitorAgentService.bat* 文件以将 EMSMonitor 代理注册为 Windows 服务。
如果之后要删除作为已注册服务的 EMSMonitor 代理，可以运行 *RegisterEMSMonitorAgentService.bat* 文件。

启用企业管理器扩展

安装企业管理器时，CA APM for TIBCO Enterprise Message Service 文件默认安装在 *<EM_Home>/examples* 目录中。要启用 CA APM for TIBCO Enterprise Message Service，需要将企业管理器文件从 *<EM_Home>/examples* 目录复制或移动到企业管理器主目录中的适当位置。

注意：只有在企业管理器上启用 CA APM for SOA 后，才能使用 CA APM for TIBCO Enterprise Message Service。有关启用 CA APM for SOA 企业管理器扩展的信息，请参阅[在企业管理器上启用扩展 \(p. 35\)](#)。

遵循这些步骤：

1. 确认 CA APM for TIBCO Enterprise Message Service 目录 *SOAExtensionForTibcoEMS* 位于 *<EM_Home>/examples* 目录中，然后将文件从 *<EM_Home>/examples/SOAExtensionForTibcoEMS* 目录复制到企业管理器目录结构中的对应位置。例如，将文件从 *<EM_Home>/examples/SOAExtensionForTibcoEMS/ext* 目录复制到 *<EM_Home>/ext* 目录。
2. 如果企业管理器是群集环境中的收集器，则从 *<EM_Home>/config/modules* 目录中删除 CA APM for TIBCO Enterprise Message Service 管理模块 *TibcoEMSMgmtModule.jar*。
应仅将管理模块复制到用作 MOM 计算机的企业管理器上的 *<企业管理器主目录>/config/modules* 目录中。其他所有文件和脚本应当同时安装在收集器企业管理器和 MOM 企业管理器中。
3. 重新启动 Workstation，以加载特定于 SOA Extension for TIBCO Enterprise Message Service 的显示板和“概览”选项卡。

使用显示板监控 TIBCO EMS

SOA Extension for TIBCO Enterprise Message Service 包括可用于监控 EMS 环境的总体运行状况的多个预配置显示板。显示板汇集已部署代理中的数据，以汇总性能信息并帮助您快速诊断和解决问题。

通常，将显示板用作监控环境的起点，因为使用显示板可以：

- 一目了然地监控 Enterprise Message Service 的关键组件的总体运行状况、性能、可用性和当前状态。
- 低级别度量标准发出已超过警告或危险阈值的信号时，及早获取生产应用环境中存在潜在问题的通知。
- 深入查看性能信息以隔离和识别出现延迟或生成错误的 Enterprise Message Service 组件。

预配置的 TIBCO Enterprise Message Service 显示板打包在 TIBCO Enterprise Message Service 管理模块 (*TibcoEMSMgmtModule.jar*) 的组成部分 Enterprise Manager Extension for TIBCO Enterprise Message Service 中。

TIBCO Enterprise Message Service 管理模块为 TIBCO Enterprise Message Service 提供了以下预配置的显示板：

Tibco EMS—概览

Enterprise Message Service 的关键活动的顶级概述，包括 EMS 服务器和代理之间的连接状态、EMS 服务器的总体运行状况、最大队列深度、最大主题深度、路由和通道的总体运行状况以及使用中连接的百分比的报警指示器。

Tibco EMS—服务器

所有 EMS 服务器的汇总状态，包括 EMS 服务器和备份服务器的状态以及使用中连接的百分比的报警指示器。

显示板还显示汇总了连接计数、使用方计数、生成方计数、入站和出站消息总发送率和待发送消息计数的图表。

Tibco EMS—队列

所有 EMS 队列的汇总状态，包括队列的入站和出站消息发送率的报警指示器和图表、最大队列深度的报警指示器以及队列的待发送消息计数和使用方计数的图表。

Tibco EMS—主题

所有 EMS 主题的汇总状态，包括主题的入站和出站消息发送率的报警和图表、最大主题深度的报警指示器和待发送消息计数的图表。

显示板还包括订户总计数以及活动和持久订户的数目。

Tibco EMS—路由

所有 EMS 路由的汇总状态，包括路由的入站和出站消息发送率的报警和图表、路由状态的报警指示器以及积压消息计数和积压消息大小的图表（如果适用）。

Tibco EMS—通道

所有 EMS 通道的汇总状态，包括通道总体状态、通道消息发送率和积压消息计数的报警指示器以及消息发送率、字节率、积压消息计数和积压消息大小的图表。

遵循这些步骤：

1. 启动企业管理器（如果目前未运行）。
2. 启动 Workstation 并登录到已安装 SOA Extension for TIBCO EMS 的企业管理器。
3. 单击“Workstation”>“新建控制台”。
4. 从“显示板”下拉列表中选择一个 TIBCO Enterprise Message Service 显示板。

例如，选择“Tibco EMS—概览”显示板以查看有关 TIBCO Enterprise Message Service 环境概述的总体运行状况的信息，包括连接状态、使用的连接百分比，以及最大队列和主题深度。

5. 双击组件选项卡打开该组件的显示板，并查看有关服务器、队列、主题、路由或通道的更多详细信息。例如，双击“服务器”选项卡可显示“Tibco EMS—服务器”显示板。-

了解并查看 TIBCO EMS 度量标准

EMSMonitor 代理可以监控本地和远程 EMS 服务器实例，并在“Tibco EMS 服务器”>“<主机名>”>“<EMS 服务器实例名>”节点下提供有关这些实例的总体运行状况的数据。

可以使用以下类别的度量标准监控 TIBCO EMS 组件的性能和运行状况：

网桥

通过网桥您可以在目标之间路由消息，以便发送给一个目标的消息也会传送给所有桥接目标。可以创建从一个目标到一个或多个相同或不同类型的目标的网桥。例如，可以创建从主题到队列或从队列到主题的网桥，或创建一个目标和多个目标之间的网桥。

在“网桥”节点下，可以查找与已定义的各个网桥关联的单个目标的度量标准。

通道

多播消息将消息一次广播给许多使用方，而不是将消息的副本分别发送给每个订阅使用方。借助于多播消息，服务器通过多播通道将消息发送到启用多播的主题。通道决定了服务器将消息发送到的多播端口和多播组地址。

在“通道”节点下，可以查找各个多播通道的度量标准。

上次检查

在“上次检查”节点下，可以查找有关 EMS 服务器实例和代理之间的连接的度量标准。

队列

队列是临时存储对象，用于存储等待转发给客户端或 Enterprise Message Service 网络中其他队列的消息。

在“队列”节点下，可以查找各个队列的度量标准，如已传送或传送中的消息数以及每个队列的入站或出站率。

路由

路由用于将两个 TIBCO Enterprise Message Service 服务器作为服务器对连接。对中的每个服务器都会将消息沿路由转发给另一个服务器上的相应目标。路由只转发在两个服务器上拥有相同名称的全局主题的消息，或具有同一队列所有者的路由队列的消息。

在“路由”节点下，可以查找在服务器之间定义的各个路由的度量标准，如路由积压消息中的消息数或每个路由的入站和出站率。

服务器

“服务器”类别提供 Enterprise Message Service 的主要运行时流程的度量标准。EMS 服务器流程创建并管理组成消息事务的其他组件。

在“服务器”节点下，可以查找服务器流程的度量标准，如连接数、可用内存以及服务器所管理的队列和主题数。

主题

主题表示逻辑主题，发布者可以写与其有关的消息，订户可以接收与其有关的发布消息。与队列不同的是，队列中只存储一个消息副本，且只有一个接收方可以接收该消息副本，而主题由 EMS 服务器维护，可将单个消息副本发布给多个感兴趣的订户。

在“主题”节点下，可以查找各个主题的度量标准，如每个主题的入站、出站以及待发送消息数。

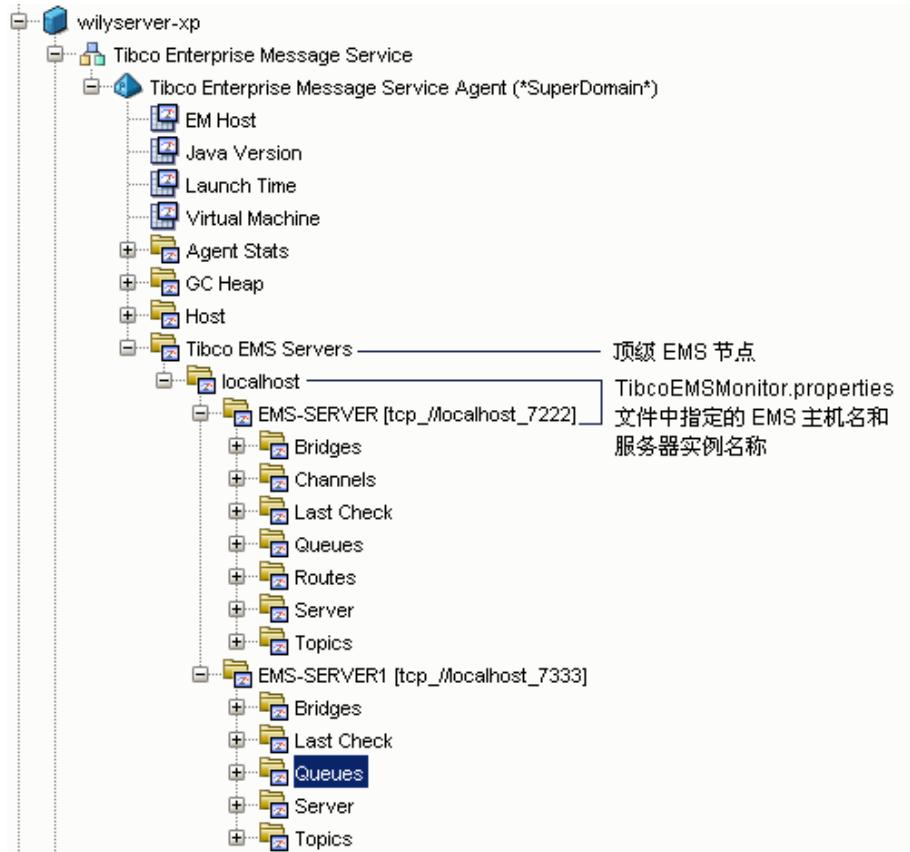
查看调查器中的 TIBCO EMS 度量标准摘要并进行导航：

1. 展开代理节点和 TIBCO Enterprise Message Service 节点，然后单击“Tibco EMS 服务器”以显示“概览”选项卡，该选项卡将列出所有监控的服务器实例及其当前状态。
2. 在“概览”选项卡上选择一个服务器实例以图形格式显示该服务器实例的最关键状态度量标准。
3. 单击“配置”选项卡显示选定服务器实例的配置度量标准。
4. 在“查看器”窗格中的选项卡上，展开服务器实例并单击任一子节点以显示有关该度量标准类别的摘要或配置信息。例如，选择“队列”节点，然后单击“临时队列”选项卡以显示所有临时队列的度量标准摘要，或单击“动态队列”选项卡以显示所有动态队列的度量标准摘要。
5. 展开任一子节点或选择单个组件（如特定队列或主题），查看有关这些单个组件以及与每个组件关联的度量标准的更多详细信息。

查看调查器中的 TIBCO EMS 度量标准节点并进行导航：

1. 展开代理，然后展开“Tibco EMS 服务器”节点以显示正在监控的 TIBCO EMS 服务器实例的主机名。
2. 展开对应于您在 *TibcoEMSMonitor.properties* 文件中定义的主机名的单个 *server_name* 节点。

3. 展开对应于您在 *TibcoEMSMonitor.properties* 文件中指定的服务器实例的单个 TIBCO EMS 服务器 *instance_name* 节点，以显示顶级 TIBCO EMS 度量标准类别的子节点。-- 例如：



4. 展开一个子节点以显示有关该度量标准类别的信息。例如，单击“队列” > “静态队列” > “<queue_name>”以查看选定队列的配置或状态度量标准。
5. 进一步展开子节点以查看选定队列的配置或状态度量标准。例如，展开“未发送的消息队列” > “<queue_name>” > “状态”节点以查看选定队列的状态度量标准。

上次检查的度量标准

除有关 TIBCO EMS 服务器及其组件的性能和操作的度量标准之外，*EMSMonitor* 代理还收集有关各个 EMS 服务器和代理之间的连接的度量标准。“上次检查”节点下的度量标准每 30 秒收集一次。这些度量标准的频率不受延迟时间或静态频率配置属性的控制，将收集所有监控级别的度量标准。

以下度量标准可用于评估每个 EMS 服务器和 *EMSMonitor* 代理之间的连接状态：

代理—EMS 连接状态

表示 *EMSMonitor* 代理当前是否已连接到 TIBCO EMS 服务器的文本字符串。

度量标准值“已连接”表示该连接存在。值“未连接”表示代理和 EMS 服务器已断开连接。

代理—EMS 连接状态值

一个数值，表示 *EMSMonitor* 代理当前是否已连接到 TIBCO EMS 服务器。

度量标准值零 (0) 表示代理已连接到 EMS 服务器。度量标准值为一 (1) 表示代理未连接到 EMS 服务器。

EMS 服务器名称

所监控的 EMS 服务器的名称。

时间戳

最近一次收集“上次检查”度量标准的日期和时间。此度量标准的格式为：

yyyy-MM-dd HH:mm:ss

队列的度量标准

以下配置度量标准可用于 TIBCO EMS 中的动态和静态队列：

桥接队列

在文本字符串中指出队列中是否存在任何桥接目标：

- 度量标准值 `true` 表示队列中存在桥接目标。
- 值 `false` 表示队列中不存在桥接目标。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

独占

在文本字符串中指出是否为队列设置独占属性：

- 度量标准值 `true` 表示队列已配置为独占队列。
- 值 `false` 表示队列未配置为独占队列。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

有效期限（秒）

表示选定队列上的消息过期之前的最大时长。如果此度量标准值为零 (0)，消息将不会过期。

如果已为队列设置服务器有效期限属性，则此属性的值将覆盖消息生成方设置的 JMSExpiration 值。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

故障防护

在文本字符串中指出是否已将队列配置为故障防护目标：

- 度量标准值 `true` 表示队列已配置为故障防护目标。
- 值 `false` 表示未将队列配置为故障防护目标。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

注意：此度量标准仅适用于 EMS 4.4。

全局

在文本字符串中指出是否已将队列配置为全局队列，以便将消息从一个服务器路由到另一个服务器：

- 度量标准值 `true` 表示队列是全局队列。
- 值 `false` 表示队列不是全局队列。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

路由是否已连接

表示选定队列的路由当前是否已连接。仅当选定队列为路由的队列时才显示此度量标准。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

流控制最大字节数

表示对此队列启用流控制的最大字节数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

最大消息数

表示服务器可以存储为此队列的待发送消息的最大消息数。

默认情况下，仅当监控级别设为 `minimum` 时才收集该度量标准。

最大重新传送次数

表示服务器可以尝试将给定消息从选定队列传送给队列接收方的最大次数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

溢出策略

表示队列超过最大消息大小或消息数时要用于此队列的溢出策略。以下溢出策略值为有效值：

- 0—表示默认溢出策略。如果超过最大字节或消息数，服务器将拒绝新消息并将一个错误返回给消息生成方。
- 1—表示 `discardOld` 溢出策略。如果队列上的消息超过最大字节数或最大消息数，将从队列中丢弃最早的消息并将一个错误返回给消息生成方。
- 2—表示 `rejectIncoming` 溢出策略。如果超过最大字节或消息数，服务器将拒绝新消息并将一个错误返回给消息生成方。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

预取计数

表示消息使用方可以从 EMS 服务器中提取的最大消息数。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

路由名称

标识与队列相关的路由名称。仅当选定队列为路由的队列时才显示此度量标准。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

路由的队列

在文本字符串中指出选定队列是否为路由的队列：

- 度量标准值 `true` 表示队列是路由的队列。
- 值 `false` 表示队列不是路由的队列。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

安全

在文本字符串中指出队列是否已配置为对传入连接进行身份验证：

- 度量标准值 `true` 表示传入连接已通过身份验证。
- 值 `false` 表示传入连接未通过身份验证。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

发送方名称

在文本字符串中指出是否已为队列配置 `sender_name` 属性：

- 度量标准值 `true` 表示已设置 `sender_name` 属性。
- 值 `false` 表示未设置 `sender_name` 属性。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

强制使用发件人姓名

在文本字符串中指出是否已为队列配置 `sender_name_enforced` 属性：

- 度量标准值为 `true` 表示已强制 `sender_name` 属性。
- 值为 `false` 表示未强制 `sender_name` 属性。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

存储名称

标识存储永久性消息的存储名称。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

注意：此度量标准适用于 EMS 5.x 或更高版本。

以下状态度量标准可用于 TIBCO EMS 中的动态和静态队列：

已发送的消息计数

表示已发送并确认的消息总数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

入站字节率

表示 EMS 客户端和路由的服务器每秒发送给选定队列的入站字节数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

传送中消息计数

表示已传送给队列所有者但尚未经过确认的消息总数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

入站消息发送率

表示 EMS 客户端或路由的服务器每秒发送到选定队列的入站消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

出站字节率

表示每秒发送给选定队列上的使用方或路由到其他服务器的出站字节数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

出站消息发送率

表示每秒发送给选定队列上的使用方或路由到其他服务器的出站消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

待发送消息计数

表示选定队列中当前待发送消息的总数。此度量标准相当于监控队列深度。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

待发送消息大小

表示选定队列中所有待发送消息的总大小 (KB)。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

接收方计数

表示选定队列的接收方数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

服务器的度量标准

以下配置度量标准可用于 TIBCO EMS 服务器流程：

已启用授权

表示是否对服务器启用了授权。

如果此度量标准为 True，则主动连接到该服务器的其他服务器必须通过名称和密码对自己进行身份验证。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

备份服务器名称

表示服务器实例的名称以及配置为选定服务器实例的备份服务器的计算机的主机名。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

客户端心跳服务器时间间隔（秒）

表示从客户端发送到服务器的心跳消息之间的秒数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

客户端超时服务器连接（秒）

表示客户端在终止与服务器的连接之前等待来自该服务器的心跳的秒数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

容错激活时间（秒）

表示备份服务器在确定活动服务器已失败之前等待心跳消息的秒数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

容错重新读取

表示故障转移后备份服务器是否重新读取配置文件(除主文件之外)。

度量标准值为 `true` 表示已重新读取文件。值 `false` 表示未重新读取文件。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

容错心跳时间间隔（秒）

表示从活动服务器发送到备份服务器的心跳消息之间的秒数。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

容错重新连接超时（秒）

表示新活动服务器在故障切换之后等待客户端重新连接的秒数。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

容错服务器 URL

表示备份服务器的 URL。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

已启用流控制

表示是否对消息生成方和服务器上的目标启用了流控制。

度量标准值为 `true` 表示已启用流控制。值 `false` 表示未启用流控制。

默认情况下，将监控级别设置为“`minimum`”时收集此度量标准。

最大日志文件大小

表示最大日志文件大小 (KB)。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

最大连接数

表示 EMS 服务器的最大连接数。

如果此度量标准为零 (0)，则允许的连接数没有限制。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

最大消息内存

表示可用于存储消息的最大内存量（字节）。

如果此度量标准为零 (0)，则可用于消息的内存大小没有限制。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

最大统计信息内存

表示可分配用于收集详细统计信息的最大内存量（字节）。

默认情况下，仅当将监控级别设置为“`full`”时才收集此度量标准。

异步存储文件最小大小

表示服务器的异步存储文件的最小大小。

此度量标准值的单位可以是 MB 或 GB，具体取决于存储文件的配置。

默认情况下，仅当将监控级别设置为“`full`”时才收集此度量标准。

注意：此度量标准仅适用于 EMS 4.4。

最小大小的存储文件

表示服务器的存储文件的最小大小。

此度量标准值的单位可以是 MB 或 GB，具体取决于存储文件的配置。

默认情况下，仅当将监控级别设置为“`full`”时才收集此度量标准。

注意：此度量标准仅适用于 EMS 4.4。

最小大小的同步存储文件

表示服务器的同步存储文件的最小大小。

此度量标准值的单位可以是 MB 或 GB，具体取决于存储文件的配置。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

注意：此度量标准仅适用于 EMS 4.4。

已启用多播

表示是否对服务器启用了多播消息。

度量标准值为 true 表示已启用多播。值为 false 表示未启用多播。

由于 EMS 版本 4.4.x 不支持多播消息，因此该度量标准对 EMS 4.4.x 服务器实例始终为 false。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

保留内存

表示保留内存的大小 (KB)。

默认情况下，该度量标准是在您将监控级别设置为 recommended 时收集的。

已启用路由

表示是否已为服务器启用路由。

度量标准值为 true 表示已启用路由。值为 false 表示未启用路由。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

服务器心跳客户端时间间隔 (秒)

表示从服务器发送到客户端以确认连接的心跳之间的秒数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

服务器心跳服务器时间间隔 (秒)

表示此服务器发送到其他服务器的心跳之间的秒数。两个服务器可通过路由或作为容错对连接。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

客户端连接的服务器超时 (秒)

表示服务器在终止客户端连接之前等待该客户端心跳的秒数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

服务器连接的服务器超时 (秒)

表示服务器在终止与其他服务器的连接之前等待来自该服务器的心跳的秒数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

服务器开始时间

表示服务器启动时间（格式为 yyyy MM-dd HH:mm:ss）。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

统计数据清除时间间隔（秒）

表示统计信息清除时间间隔（秒）。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

已启用统计信息

表示是否启用了统计信息收集。

度量标准值为 `true` 表示已收集统计信息。值 `false` 表示未收集统计信息。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

已启用存储截断

表示服务器是否应在需要时尝试截断存储文件。

度量标准值为 `true` 表示已启用存储截断。值为 `false` 表示未启用截断。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

注意：此度量标准仅适用于 EMS 4.4。

已启用交换

表示是否启用了消息交换以允许服务器将消息从流程内存交换到磁盘。

度量标准值为 `true` 表示已启用交换。值为 `false` 表示未启用交换。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

已启用 Tibco RV 传输

表示是否已为服务器启用 TIBCO Rendezvous 消息传送。

度量标准值 `true` 表示启用了桥接到 `tibrv` 和 `tibrvcn` 传输以及从 `tibrv` 和 `tibrvcn` 传输桥接。值为 `false` 表示未启用 Rendezvous 传输。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

已启用 Tibco SmartSockets 传输

表示是否已为服务器启用 TIBCO SmartSockets 传输协议。

度量标准值为 `true` 表示已启用与 SmartSockets 传输之间的往返桥接。值为 `false` 表示未启用 SmartSockets 传输。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

URL

表示服务器实例 URL。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

版本

表示 EMS 服务器实例的版本号。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

以下状态度量标准可用于 TIBCO EMS 服务器流程：

异步数据库大小

表示异步存储文件的当前大小 (KB)。

默认情况下，该度量标准是在您将监控级别设置为 recommended 时收集的。

备份服务器状态

表示将服务器的当前状态表示为“正在运行”或“已停止”的文本字符串。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

备份服务器状态值

表示服务器当前状态的数值。

值零 (0) 表示备份服务器正在运行。值一 (1) 表示备份服务器已停止。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

连接计数

表示服务器的活动连接数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

使用方计数

表示服务器上的消息使用方总数。此度量标准包括所有订户和队列接收方。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

持久订户计数

表示服务器上的持久订户数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

入站字节率

表示服务器接收入站消息的速率，单位为字节/秒 (Bps)。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

入站消息发送率

表示服务器每秒接收的入站消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

日志文件大小

表示日志文件的当前大小 (KB)。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

已使用消息内存

表示服务器上当前用于存储消息的内存（字节）。

默认情况下，该度量标准是在您将监控级别设置为 recommended 时收集的。

出站字节率

表示服务器发送出站消息的速率，单位为字节/秒 (Bps)。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

出站消息发送率

表示服务器每秒的出站消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

待发送消息计数

表示服务器的待发送消息总数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

待发送消息大小

表示服务器的待发送消息的总大小 (KB)。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

生成方计数

表示服务器上的消息生成方总数。

此度量标准包括所有主题发布者和队列发送方。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

队列计数

表示服务器上队列的总数，包括静态、动态和临时队列。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

路由恢复计数

表示可存储在服务器上的路由恢复消息总数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

路由恢复时间间隔（秒）

表示检查服务器上的路由恢复消息的时间间隔（秒）。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

会话计数

表示客户端应用程序在服务器上创建的会话总数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

服务器状态

表示将服务器的当前状态表示为“活动”或“待机”的文本字符串。

如果代理无法连接到 EMS 服务器，服务器状态将显示为“未知”。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

服务器状态值

表示服务器当前状态的数值。

值零 (0) 表示服务器处于“活动”状态。值为一 (1) 表示服务器处于待机状态。值二 (2) 表示“未知”状态。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

同步 DB 大小

表示同步存储文件的当前大小 (KB)。

默认情况下，该度量标准是在您将监控级别设置为 recommended 时收集的。

主题计数

表示服务器上主题的总数，包括静态、动态和临时主题。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

详细信息：

[关于监控主服务器和备份服务器 \(p. 195\)](#)

关于监控主服务器和备份服务器

如果要监控主服务器和备份服务器对，主服务器和备份服务器必须配置为使用相同的用户名和密码设置。要在主服务器脱机时允许 *EMSMonitor* 代理连接到备份服务器，需要使用相同的用户名和密码。

如果主服务器脱机并且是与备份服务器一起配置的，服务器状态和备份服务器状态度量标准将继续报告主服务器和备份服务器的状态。但是，仅为主服务器报告备份服务器名称度量标准。然而，根据成对服务器的状态，服务器状态度量标准可从“活动”切换到“备用”，一次只可以为一个服务器报告备份服务器名称。

主题的度量标准

以下配置度量标准可用于 TIBCO EMS 动态和静态主题：

桥接主题

在文本字符串中指出是否存在针对主题的任何桥接目标：

- 度量标准值 `true` 表示存在针对主题的桥接目标。
- 值为 `false` 表示不存在主题的桥接目标。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

有效期限（秒）

表示选定主题上的消息过期前的最大时长。如果此度量标准值为零(0)，消息将不会过期。

如果已为队列设置服务器有效期限属性，则该属性的值将覆盖消息生成方设置的 `JMSExpiration` 值。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

故障防护

在文本字符串中指出是否已将主题配置为故障防护目标：

- 度量标准值 `true` 表示主题已配置为故障防护目标。
- 值 `false` 表示主题未配置为故障防护目标。

注意：此度量标准仅适用于 EMS 4.4。

全局

在文本字符串中指出是否已将主题配置为全局主题，以及是否可用于将消息从一个服务器路由到另一个服务器：

- 度量标准值 `true` 表示主题是全局主题。
- 值 `false` 表示主题不是全局主题。

默认情况下，仅当监控级别设为 `recommended` 时才收集该度量标准。

流控制最大字节数

表示对此主题启用流控制的最大字节数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

最大消息数

表示服务器可以存储为选定主题的待发送消息的最大消息数。

默认情况下，仅当监控级别设为 `minimum` 时才收集该度量标准。

多播通道名称

在为主题启用多播时，识别与主题相关的多播通道名称。

默认情况下，仅当监控级别设为 `recommended` 时才收集该度量标准。

已启用多播

表示是否对主题启用了多播：

- 度量标准值 `true` 表示已为主题启用多播。
- 值 `false` 表示未启用多播。

注意：由于 EMS 4.4.x 不支持多播消息，因此此度量标准对 EMS 4.4.x 服务器实例上的主题始终为 `false`。

默认情况下，仅当监控级别设为 `recommended` 时才收集该度量标准。

溢出策略

表示主题超过最大消息数或最大消息大小（字节）时要用于该主题的溢出策略。以下溢出策略值为有效值：

- 0—表示默认溢出策略。如果订户超过最大字节数或最大消息数，该订户将不会接收到消息。不会将任何错误返回给消息生成方。
- 1—表示 `discardOld` 溢出策略。如果订户超过最大字节数或最大消息数，将新消息传送给该订户之前，会丢弃最早的消息。
- 2—表示 `rejectIncoming` 溢出策略。如果订户超过最大字节数或最大消息数，将拒绝所有新消息并将一个错误返回给生成方。

默认情况下，仅当将监控级别设置为“`full`”时才收集此度量标准。

预取计数

表示消息使用方可以从主题提取的最大消息数。

默认情况下，该度量标准是在您将监控级别设置为 `recommended` 时收集的。

安全

在文本字符串中指出主题是否已配置为对传入连接进行身份验证：

- 度量标准值 `true` 表示传入连接已通过身份验证。
- 值 `false` 表示传入连接未通过身份验证。

默认情况下，将监控级别设置为“`minimum`”时收集此度量标准。

发送方名称

在文本字符串中指出是否已为主题配置 `sender_name` 属性：

- 度量标准值 `true` 表示已设置 `sender_name` 属性。
- 值 `false` 表示未设置 `sender_name` 属性。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

强制使用发件人姓名

在文本字符串中指出是否已为主题配置 `sender_name_enforced` 属性：

- 度量标准值为 `true` 表示已强制 `sender_name` 属性。
- 值为 `false` 表示未强制 `sender_name` 属性。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

存储名称

标识存储永久性消息的存储名称。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

注意：此度量标准仅适用于 EMS 5.0 或更高版本。

以下状态度量标准可用于 TIBCO EMS 动态和静态主题：

活动的持久订户计数

表示选定主题的当前活动的持久订户数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

持久订户计数

表示选定主题的持久订户总数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

入站字节率

表示 EMS 客户端和路由的服务器每秒发送到选定主题的入站字节数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

入站消息发送率

表示 EMS 客户端或路由的服务器每秒发送到选定主题的入站消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

出站字节率

表示每秒发送给选定主题上的使用方或路由到其他服务器的出站字节数。

默认情况下，仅当将监控级别设置为“full”时才收集此度量标准。

出站消息发送率

表示每秒发送给选定主题上的使用方或路由到其他服务器的出站消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

待发送消息计数

表示选定主题的待发送消息数。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

待发送消息大小

表示选定主题的所有待发送消息的总大小 (KB)。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

订户计数

表示选定主题的订户总数。

此度量标准包括持久用户和非持久订户。

默认情况下，将监控级别设置为“minimum”时收集此度量标准。

路由的度量标准

默认情况下不监控 TIBCO EMS 路由，但可以通过将 `<ServerInstance>.monitor.routes` 属性添加到 `TibcoEMSMonitor.properties` 文件来启用对路由的监控。如果启用对路由的监控，以下配置度量标准可用于 TIBCO EMS 路由：

远程服务器 URL

路由所连接到的远程服务器的 URL。

路由类型

表示路由是主动路由还是被动路由。

度量标准值表示路由为“主动”还是“被动”。

区域名称

选定路由所属的路由区域的名称。

区域类型

表示路由所属的路由区域是单跳 (1hop) 区域类型还是多跳 (mhop) 区域类型。

如果无法确定区域类型，度量标准值将为 UNKNOWN。

如果启用对路由的监控，则以下**状态**度量标准可用于 TIBCO EMS 路由：

积压消息计数

路由的积压消息中的消息数。

此度量标准仅适用于 EMS 5.0 或更高版本。

积压消息大小

路由的积压消息中所有消息的大小 (KB)。

此度量标准仅适用于 EMS 5.0 或更高版本。

入站字节率

以每秒字节数 (Bps) 衡量的路由入站消息的速率。

入站消息发送率

每秒接收的入站消息数。

已连接

表示路由当前是否已连接的文本字符串。

度量标准值为 true 表示已连接路由。值为 false 表示未连接路由。

已连接值

表示路由是否已连接的数值。

度量标准值为零 (0) 表示已连接路由。任何非零值表示路由已断开连接。

出站字节率

以每秒字节数 (Bps) 衡量的路由出站消息的速率。

出站消息发送率

每秒发送的出站消息数。

停顿目标

表示路由是否具有任何停顿目标。

度量标准值为 true 表示停顿的目标。值 false 表示没有任何停顿目标。

通道的度量标准

默认情况下不监控 TIBCO EMS 多播通道，但可以通过将 `<ServerInstance>.monitor.channels` 属性添加到 `TibcoEMSServer.properties` 文件来启用对通道的监控。如果启用了对通道的监控，以下配置度量标准可用于 TIBCO EMS 多播通道操作：

通道接口

用于发送多播数据的、支持多播的 IP 地址。

最大时间

服务器保留已发送消息以便进行重新传输的最大时间长度(以秒为单位)。

最大速率

传输多播数据的最大传输速率，单位为字节/秒 (Bps)。

多播组地址

消息发送到的多播组地址和端口。

多播生存时间

消息可在服务器与多播后台驻留程序之间产生的最大网络跳数。

Priority

带宽分配过程中为此通道指定的优先级。

最高优先级为 -5，最低优先级为 5。

如果启用对通道的监控，则以下状态度量标准可用于 TIBCO EMS 多播通道操作：

积压消息计数

缓冲区中等待通过通道发送的消息数。

积压消息大小

缓冲区中等待通过通道发送的消息大小（以字节为单位）。

字节速率

通道处理入站消息和出站消息的速率，单位为字节/秒 (Bps)。

通道状态

表示通道是处于活动状态还是已在服务器配置中定义但未处于活动状态的文本字符串。

度量标准为“活动”表示通道处于活动状态，“非活动”表示通道已在服务器配置中定义但未处于活动状态。

通道状态值

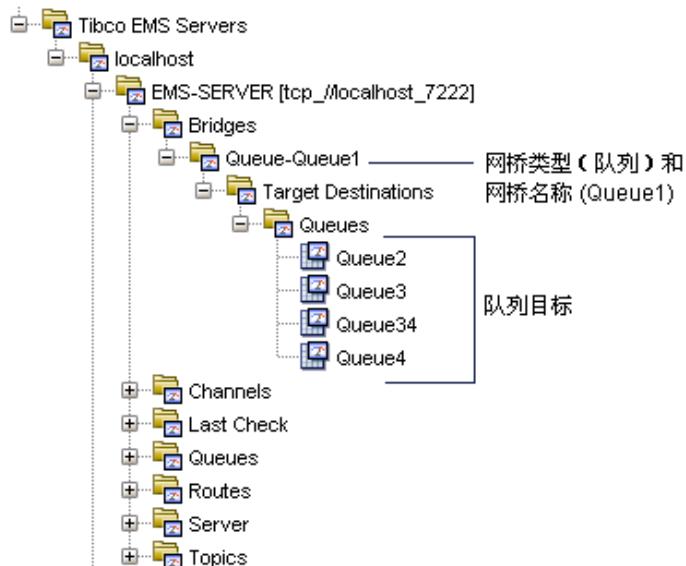
表示通道是处于活动状态 (0) 还是已在服务器配置中定义但未处于活动状态（任何非零值）的数值。

消息速率

通道每秒处理的消息数。此度量标准包括入站消息和出站消息。

网桥的度量标准

默认情况下不监控 TIBCO EMS 桥，但通过将 `<ServerInstance>.monitor.bridges` 属性添加到 `TibcoEMSMonitor.properties` 文件，可启用对桥的监控。如果启用了对网桥的监控，节点名称将标识网桥类型和网桥源名称，其子节点按目标类型来标识目标。- 目标和用于筛选消息的消息选择器（如果定义了选择器）是调查器树中报告的仅有的度量标准。例如：



查看默认的 TIBCO EMS 度量标准组

SOA Extension for TIBCO Enterprise Message Service 包括定义默认显示板和报警用的默认度量标准组。您还可以在自定义显示板和报警中使用这些默认度量标准组。

默认度量标准组作为 TIBCO Enterprise Message Service 管理模块 (`TibcoEMSMgmtModule.jar`) 的一部分包含在 TIBCO Enterprise Message Service 的企业管理器扩展中。

查看 TIBCO Enterprise Message Service 代理的默认度量标准组

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>Introscope SOA Extension for TibcoEMS <version> (*超级域*)。
3. 展开“度量标准组”节点，查看为 TIBCO Enterprise Message Service 管理模块定义的所有度量标准组。
4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

您可以修改任意度量标准组的默认设置或自行创建自定义度量标准组。

注意：有关创建或修改度量标准组的详细信息，请参阅《CA APM Workstation 用户指南》。

查看默认的 TIBCO EMS 报警

SOA Extension for TIBCO Enterprise Message Service 包括预配置的显示板中使用的默认报警定义。您也可以在自定义的显示板中使用这些默认报警。大部分默认报警预配置有默认的“警告”和“危险”阈值，如果超过阈值或重要级别增加，则会向控制台发送通知。

默认报警定义作为 TIBCO Enterprise Message Service 管理模块 (*TibcoEMSMgmtModule.jar*) 的一部分，打包在 TIBCO Enterprise Message Service 的企业管理器扩展中。

查看 TIBCO Enterprise Message Service 代理的默认报警定义

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>Introscope SOA Extension for TibcoEMS <version> (*超级域*)。
3. 展开“报警”节点，查看为 TIBCO Enterprise Message Service 管理模块定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

尤其是，应该检查您认为最重要的报警的“警告”和“危险”阈值默认设置，并根据需要调整这些值。您可能还需要为一些报警添加通知或指定更正操作。

您可以修改任何报警的默认设置，也可创建您自己的自定义报警。

注意：有关创建或修改报警的详细信息，请参阅《CA APM Workstation 用户指南》。

代理配置属性汇总

EMSMonitor 代理依赖于 *TibcoEMSMonitor.properties* 文件中的大量配置属性来定义所连接到的 EMS 服务器实例及其监控的组件。下面列出了这些属性的汇总，并描述了每个属性可接受的值的类型（作为设置有效值的快速参考）。

通过几个配置属性，您可以为模式匹配指定正则表达式。有关正则表达式语法和有效构造的摘要，请参阅 Java 文档中有关[模式匹配](#)的信息。

ems.server.list

包含想要监控的 EMS 服务器实例名称的逗号分隔列表。

所指定的名称不需要与正在监控的服务器实例的实际名称匹配。例如，如果您有两个具有相同名称但不同端口的服务器实例，可以在 *ems.server.list* 属性中使用别名对其进行区分。

例如：

```
ems.server.list=mercury01,mercury02,jupiter03
```

<ServerInstance>.host

承载指定服务器实例的 EMS 服务器的主机名或 IP 地址。默认值为 *localhost*。

例如：

```
mercury01.host=winsrvT400
```

<ServerInstance>.port

承载指定服务器实例的 EMS 服务器的端口号。默认值为 7222。

例如：

```
mercury01.port=7200
```

<ServerInstance>.username

具有管理权限的用户名，可用于建立到 EMS 服务器的连接。通常在使用 *emsPwdEncryptor* 实用工具对帐户密码进行加密时设置此属性。

默认值为 *admin*。

例如：

```
mercury01.username=jgarcia
```

<ServerInstance>.password

为 *<ServerInstance>.username* 属性指定的用户帐户的密码。通常在使用 *emsPwdEncryptor* 实用工具对帐户密码进行加密时设置此属性。

默认值为空字符串（无密码）。

例如：

```
mercury01.password=YCLhqcwQfpC=
```

<ServerInstance>.delaytime

在对 EMD 服务器进行查询以刷新正在监控的 EMS 服务器组件的度量标准之前的时间间隔（以秒为单位）。

EMSMonitor 代理将按此时间间隔收集有关服务器实例及其组件的状态的度量标准。

默认值为 60 秒。

例如：

```
mercury01.delaytime=90
```

<ServerInstance>.report.static.freq

在静态度量标准查询之间执行的 EMS 服务器状态查询数。例如，如果将此属性设置为 20，则 *EMSMonitor* 代理每收集 20 次度量标准，与静态配置相关的度量标准才更新一次。

默认值为 20。

例如：

```
mercury01.report.static.freq=50
```

<ServerInstance>.queue.filter.includeonly.regex

指定一个正则表达式来监控名称与该表达式匹配的队列。可以使用任何有效的正则表达式。某些特殊字符需要转义序列。

默认情况下，如果不使用此属性指定筛选，则 *EMSMonitor* 代理会收集所有队列的度量标准。

例如：

```
mercury01.queue.filter.includeonly.regex=[A-H]
```

<ServerInstance>.topic.filter.includeonly.regex

指定正则表达式以监控其名称与表达式匹配的主题。可以使用任何有效的正则表达式。某些特殊字符需要转义序列。

默认情况下，如果不使用该属性指定筛选，*EMSMonitor* 代理将收集所有主题的度量标准。

例如：

```
mercury01.topic.filter.includeonly.regex=[a-hA-H]
```

<ServerInstance>.queue.filter.include.dynamic

指定是否要监控此 EMS 服务器实例的动态队列。如果要包括动态队列的度量标准，请将此属性设置为 *true*。

默认情况下，*EMSMonitor* 代理仅监控静态队列。

例如：

```
mercury01.queue.filter.include.dynamic=true
```

<ServerInstance>.topic.filter.include.dynamic

指定是否要监控该 EMS 服务器实例的动态主题。如果要包括动态主题的度量标准，可将该属性设置为 *true*。

默认情况下，*EMSMonitor* 代理仅监控静态主题。

例如：

```
mercury01.topic.filter.include.dynamic=true
```

<ServerInstance>.monitor.bridges

指定是否要监控该 EMS 服务器实例的桥。如果要包括桥的度量标准，可将该属性设置为 *true*。

默认情况下，*EMSMonitor* 代理不监控网桥。

例如：

```
mercury01.monitor.bridges=true
```

<ServerInstance>.monitor.channels

指定是否要监控该 EMS 服务器实例的多播通道。如果要包括通道的度量标准，可将该属性设置为 *true*。

默认情况下，*EMSMonitor* 代理不监控通道。

例如：

```
mercury01.monitor.channels=true
```

<ServerInstance>.monitor.routes

指定是否要监控该 EMS 服务器实例的路由。如果要包括路由的度量标准，可将该属性设置为 *true*。

默认情况下，*EMSMonitor* 代理不监控路由。

例如：

```
mercury01.monitor.routes=true
```

<ServerInstance>.bridge.filter.includeonly.regex

指定一个正则表达式来监控名称与该表达式匹配的网桥。可以使用任何有效的正则表达式。某些特殊字符需要转义序列。

默认情况下，如果将 *<ServerInstance>.monitor.bridges* 属性设置为 *true* 且未使用此属性指定筛选，则 *EMSMonitor* 代理会收集所有网桥的度量标准。

例如：

```
mercury01.bridge.filter.includeonly.regex=test.*
```

<ServerInstance>.channel.filter.includeonly.regex

指定正则表达式以监控其名称与表达式匹配的通道。可以使用任何有效的正则表达式。某些特殊字符需要转义序列。

默认情况下，如果已将 `<ServerInstance>.monitor.channels` 属性设置为 `true`，且未使用该属性指定筛选，*EMSMonitor* 代理将收集所有通道的度量标准。

例如：

```
mercury01.channel.filter.includeonly.regex=test.*
```

<ServerInstance>.route.filter.includeonly.regex

指定正则表达式以监控其名称与表达式匹配的路由。可以使用任何有效的正则表达式。某些特殊字符需要转义序列。

默认情况下，如果已将 `<ServerInstance>.monitor.routes` 属性设置为 `true`，且未使用该属性指定筛选，*EMSMonitor* 代理将收集所有路由的度量标准。

例如：

```
mercury01.route.filter.includeonly.regex=test.*
```

<ServerInstance>.monitoring.level

定义 EMS 服务器实例的监控级别。此属性的有效设置有 `minimum`、`recommended` 和 `full`。

默认服务器监控级别为 `recommended`。

例如：

```
mercury01.monitoring.level=minimum
```

<ServerInstance>.queue.monitoring.level

为 EMS 服务器实例上的队列定义监控级别。此属性的有效设置有 `minimum`、`recommended` 和 `full`。

默认队列监控级别为 `recommended`。

例如：

```
mercury01.queue.monitoring.level=recommended
```

<ServerInstance>.topic.monitoring.level

为 EMS 服务器实例上的主题定义监控级别。此属性的有效设置有 `minimum`、`recommended` 和 `full`。

默认主题监控级别为 `recommended`。

例如：

```
mercury01.topic.monitoring.level=full
```

client.identity

指定可供 EMS 服务器用于验证 *EMSMonitor* 代理身份的证书的路径。大多数情况下，在运行 *emsPwdEncryptor* 程序时设置此属性。

例如：

```
client.identity=C:/TibcoEMSMonitor/certs/client.p12
```

ssl.password

为客户端安全证书指定加密密码。

大多数情况下，在运行 *emsPwdEncryptor* 程序时设置此属性。

<ServerInstance>.ssl.connection

指定是否使用安全套接字层 (SSL) 协议连接到服务器实例。

如果要使用 SSL 连接到服务器实例，请将此属性设置为 *enable*。如果想要允许不安全的通信，请将此属性设置为 *disable*。

与安全性相关的属性没有默认值。

<ServerInstance>.verify.host

指定 *EMSMonitor* 代理是否必须要验证 EMS 服务器的证书。将此属性设置为 *true* 将要求代理根据为 *trusted.certificates* 属性定义的列表来验证 EMS 服务器的安全证书。

与安全性相关的属性没有默认值。

例如，要求在代理连接到 EMS 服务器实例 *mercury01* 时进行验证：

```
mercury01.verify.host=true
```

trusted.certificates

指定 *EMSMonitor* 代理用于验证服务器证书的可信证书的列表（以逗号分隔）。如果 *verify.host* 属性设置为 *true*，且适用于所有使用 SSL 的 EMS 服务器实例，则该属性是必需的。

与安全性相关的属性没有默认值。

<ServerInstance>.verify.hostname

指定代理是否应验证服务器证书的“公用名称 (CN)”字段。

如果希望代理将已连接主机的名称或

<ServerInstance>.expected.name 属性中指定的名称与服务器证书中的“公用名称 (CN)”字段进行比较，请将此属性设置为 *true*。如果设为 *true* 且名称不匹配，代理将拒绝连接。

与安全性相关的属性没有默认值。

<ServerInstance>.expected.hostname

指定 *EMSMonitor* 代理预期在服务器证书中的“公用名称 (CN)”字段中找到的名称。

与安全性相关的属性没有默认值。

cipher.suites

指定一个以逗号分隔的密码组列表，*EMSMonitor* 代理可使用该列表来加密与启用了 SSL 的 EMS 服务器的通信。- *EMSMonitor* 代理可以使用要监控的 EMS 服务器所支持的任何加密软件包。如果设置该属性，它将适用于所有启用 SSL 的 EMS 服务器实例。

与安全性相关的属性没有默认值。

第 10 章： 监控 webMethods Broker

Software AG WebMethods Broker 提供了异步处理和消息处理服务，用于单独从 WebMethods Integration Server 发布文档。通过 webMethods Broker，您可以发布文档且保证送达，以确认这些文档能够到达其目的地。如果无法将保证送达的文档送达其目的地，Broker 会确认永不发布这些文档。通过 SOA Extension for webMethods Broker，您可以监控关键的 webMethods Broker 组件，包括 Broker 服务器、各个 Broker 实例、Broker 客户端以及文档处理事件。

本节介绍了可用于监控和分析 webMethods Broker 环境的性能、可用性以及总体运行状况的、特定于 Broker 的显示板、度量标准以及报警。

此部分包含以下主题：

- [关于 webMethods Broker \(p. 211\)](#)
- [如何安装 SOA Extension for webMethods Broker \(p. 212\)](#)
- [使用显示板监控 webMethods Broker \(p. 221\)](#)
- [了解并查看 Broker 的度量标准 \(p. 223\)](#)
- [查看默认的 Broker 度量标准组 \(p. 232\)](#)
- [查看默认 Broker 报警 \(p. 232\)](#)

关于 webMethods Broker

webMethods Broker 管理应用程序之间的文档路由，并作为组织中业务流程、企业系统和传统系统、数据库和其他后端系统、内部工作流与外部工作流以及 Web 服务之间的链接。在面向服务的体系结构中，webMethods Broker 负责处理消息路由、排队、存储以及筛选，以适应各种平台和协议中的大量消息传递。-

webMethods Broker 协调客户端程序之间的文档交换。它提供以下服务：

- 对 Broker 客户端发布的文档的所有文档事件进行排队。
- 将文档发送到已订阅并准备好接收事件的客户端。
- 确认可靠地送达所接收到的文档。
- 提供筛选服务，通过此服务 Broker 客户端可根据所接收文档的内容对文档有选择地进行筛选。
- 维护有关以下各项的信息：文档类型、客户端组、发布权限、订阅权限、网络访问控制、每个组中客户端的队列特征以及所创建的每个 Broker 客户端的状态信息。

由于安全可靠地送达消息是送达业务服务和完成业务事务的关键部分，因此，SOA Extension for webMethods Broker 提供了一个用于监控 webMethods Broker 的操作和性能度量标准的独立代理。

用于监控 webMethods Broker 的独立代理不是核心 Java 或 .NET 代理的扩展。代理 *WilyWMBrokerMonitor* 不检测 Java 类，而是使用 WmBrokerClient API 收集 webMethods Broker 组件和操作的性能信息，然后将此信息报告给企业管理器。

由于 *WilyWMBrokerMonitor* 代理是独立代理，因此它作为单独的软件包进行分发，并需要完成自己特有的配置步骤。例如，必须配置要监控的 Broker 服务器的连接信息，并添加筛选以自定义要为其收集度量标准的组件。

在配置适当的代理属性之后，可以使用 *WilyWMBrokerMonitor* 代理开始监控 webMethods Broker 服务器和客户端操作。

如何安装 SOA Extension for webMethods Broker

由于 *WilyWMBrokerMonitor* 代理不依赖于核心代理，因此可以独立于其他任何 CA Introscope® 组件来安装和配置 *WilyWMBrokerMonitor* 代理。

要添加 SOA Extension for webMethods Broker，请执行以下高级步骤：-

1. [确保实施满足添加 SOA Extension for webMethods Broker 的](#) (p. 213) 先决条件。
2. [运行独立代理安装程序](#) (p. 213) 或[使用响应文件](#) (p. 214) 将所需代理文件添加到环境中。
3. [准备 webMethods Broker 服务器以进行监控](#) (p. 214)。
4. [配置 WilyWMBrokerMonitor 代理配置文件](#) (p. 215) 以定义连接和监控属性。
5. [启用 webMethods Broker 的企业管理器扩展](#) (p. 220)。

验证必要条件

在您添加 SOA extension for webMethods Broker 之前，您的实施必须符合特定要求。

遵循这些步骤:

1. 确认已安装 WebMethods Broker。
注意: 有关 webMethods Broker 要求，请参阅《兼容性指南》。
2. 确认已在您的环境中安装企业管理器和 Workstation。
3. 确认具有 WilyWMBrokerMonitor 代理将向其发送数据的企业管理器的连接信息。

运行独立代理安装程序

使用独立代理安装程序可以安装不依赖于核心 Java 或 .NET 代理的独立代理。通过独立代理安装程序，可以监控 webMethods Broker。安装程序会将代理相关的文件添加到环境中，并配置与企业管理器的代理连接。独立代理安装程序会提取所需的文件并将其置于适当的位置，以供您修改以完成代理配置。

遵循这些步骤:

1. 针对您的操作环境启动适当的独立代理安装程序。
2. 在“简介”页面上，单击“下一步”。
3. 选择要安装的监控软件包，然后单击“下一步”。例如，选择“SOA Extension For webMethods Broker”。
此选项将启用对 webMethods Broker 的监控。
4. 对于安装目录，单击“下一步”接受默认位置，或单击“浏览”指定其他位置。
5. 对于企业管理器连接设置，指定要向其发送数据的代理的企业管理器主机名和端口号，然后单击“下一步”。
6. 检查设置摘要，然后单击“安装”。
此时将开始安装。
7. 安装完成之后，单击“完成”。
此时将关闭独立代理安装程序。

使用响应文件进行静默安装

如果不希望以交互方式运行独立代理安装程序，可以编辑独立示例响应文件来安装代理文件。使用该方法可以在静默模式下启用 SOA extension for webMethods Broker。

遵循这些步骤:

1. 打开 SampleResponseFile.StandaloneAgentPP.txt 文件，该文件与独立代理安装程序位于同一目录中。
2. 编辑 SampleResponseFile.StandaloneAgentPP.txt 文件，将 shouldInstallWMBroker 属性设置为 true，以向代理添加 SOA Extension for webMethods Broker。例如：
`shouldInstallWMBroker=true`
3. 保存 SampleResponseFile.StandaloneAgentPP.txt 文件。
4. 在命令行中输入相应的命令以调用安装程序。

注意: 有关以静默模式安装代理的详细信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

手工提取安装存档

如果没有访问独立代理安装程序或独立响应文件的权限，您可以下载适用于您的操作环境的独立安装存档。

遵循这些步骤:

1. 从 [CA Support](#) 上的 CA APM 软件下载区域跳转。
2. 下载适用于您的操作环境的独立安装存档。
3. 使用适用于操作环境的命令从存档中手工提取文件。例如，使用 UNIX 计算机上的 tar 命令：

```
tar -xvf IntroscopeStandaloneAgentPPInstaller<版本>unix.tar
```

准备 WebMethods Broker 以进行监控

首先确定一个具有管理权限的客户端组，并验证本地是否存在需要监控的库，然后才能监控 WebMethods Broker。

遵循这些步骤:

1. 验证您是否拥有 admin 客户端组，可用于创建供 WmBrokerAgent 使用的 Broker 客户端。

2. 确认是否已提供下列 Broker 库：

对于 Webmethods 7.x:

`wmbrokerclient.jar`
`g11nutils.jar`

对于 Webmethods 8.x:

`wm-brokerclient.jar`
`wm-g11nutils.jar`

这些文件通常位于以下位置：

- `<Broker_Home>/common/lib`—适用于 webMethods Broker 6.5。
- `<Broker_Home>/lib`—适用于其他受支持的 webMethods Broker 版本。

注意：有关 webMethods Broker 要求，请参阅《兼容性指南》。

为 webMethods Broker 配置代理

运行安装程序以将代理文件添加到环境中之后，会存在 `WilyWMBrokerMonitor` 目录。您可以使用该目录中的文件来为 webMethods Broker 配置代理。

遵循这些步骤：

1. 将以下 Broker 库从 `<Broker_Home>/lib` 目录或 `<Broker_Home>/common/lib` 复制到 `WilyWMBrokerMonitor/lib` 目录：
 - `wm-brokerclient.jar`
 - `wm-g11nutils.jar`
2. 使用 `WilyWMBrokerMonitor` 目录和子目录中的文件配置 `WmBrokerAgent` 连接和监控属性。

lib 目录

包含所需的已打包成 .jar 文件的库 (`jline-0.9.9.jar`)。

config 目录

包含 `WilyWMBrokerMonitor.properties` 文件，可使用该文件配置用于连接到 webMethods Broker 客户端和服务器的参数。

“Windows 服务” 目录

包含用于将 `WmBrokerAgent` 流程作为 Windows 服务进行注册和取消注册的文件。

Agent.jar 文件

提供代理用于收集度量标准并将这些标准转发给企业管理器的类。

IntroscopeAgent.profile 文件

提供使您能够配置与企业管理器之间的连接的属性。

WmBrokerAgent.jar 文件

提供代理用于监控 webMethods Broker 服务器和客户端的类。

wmBrokerPwdEncryptor 文件

提供密码加密脚本（wmBrokerPwdEncryptor.bat 或
wmBrokerPwdEncryptor.sh）。使用该脚本可以加密用于连接到
webMethods Broker 服务器的用户密码和 SSL 客户端证书密码。

WmBrokerAgent 文件

提供启动脚本（WmBrokerAgent.bat 或 WmBrokerAgent.sh），以
在您准备开始监控 WebMethods Broker 时运行 WmBrokerAgent。

配置基本连接属性

WmBrokerAgent 可将 WebMethods Broker 服务器中的数据报告给企
业管理器。可启用度量标准收集和报告功能，如下所示：

1. 指定接收 WmBrokerAgent 报告的企业管理器的连接信息。
2. 识别代理从中收集数据的服务器的列表。

遵循这些步骤：

1. 在文本编辑器中打开 IntroscopeAgent.profile 文件，验证企业
管理器连接参数。例如：

```
introscope.agent.enterprisemanager.transport.tcp.host.DEFAULT=mercury  
introscope.agent.enterprisemanager.transport.tcp.port.DEFAULT=5001
```

如果使用独立代理安装程序安装了代理文件，请使用您在安装过程中
输入的值。如果手工提取了代理文件，您可能需要修改默认设置才能
连接到正确的企业管理器。

2. 在文本编辑器中打开 WmBrokerAgent 启动脚本。
3. 将 JAVA_HOME 环境变量设置为适当的目录。例如：

```
set JAVA_HOME=C:\webMethods7\jvm\win150\jre
```

适用于 JRE 1.5 或更高版本： 监控 webMethods Broker 需要 JRE 1.5 或
更高版本。

4. 在文本编辑器中打开 WilyWMBrokerMonitor.properties 文件，并设置
主机属性以标识要监控的 WebMethods Broker 服务器。例如：

```
wily.webmethods.broker.server.host=vepsa09
```

5. 设置端口属性，以指定用于连接到要监控的 WebMethods Broker 服务器的端口号。例如：

```
wily.webmethods.broker.server.port=6890
```

6. 设置客户端组属性，以指定应在其中创建 BrokerAdminClient 的客户端组。例如：

```
wily.webmethods.broker.server.clientgroup=admin
```

配置收集度量标准的轮询时间间隔

WmBrokerAgent 会定期轮询每个 Broker 服务器，以检索最新状态和配置信息。- 您可以使用配置属性控制这些查询的频率。

要配置 WmBrokerAgent 查询服务器的频率，请设置时间间隔属性以配置用于收集 Broker 度量标准的轮询时间间隔。例如，每隔 14000 毫秒收集并报告一次度量标准：

```
wily.webmethods.broker.interval=14000
```

配置要监控的 Broker 客户端组

可以使用 wily.webmethods.broker.clientstat.clientGroups 配置属性控制要在 Broker 上监控的特定客户端组。

要配置要监控的 Broker 客户端组，请指定要监控的 Broker 客户端组列表。默认情况下，会报告指定客户端组中所有客户端的度量标准。例如，要收集并报告 IntegrationServer 和 IS-Backup 客户端组中所有 Broker 客户端的度量标准，应按如下所示设置属性：

```
wily.webmethods.broker.clientstat.clientGroups=IntegrationServer,IS-Backup
```

为 WebMethods Broker 配置 SSL 连接

如果您在客户端组与 WebMethods Broker 服务器之间使用安全套接字层 (SSL) 连接，则代理必须提供待确认的签名证书。要启用对 SSL 连接的支持，请为 SSL 证书创建加密密码，并在 WilyWMBrokerMonitor.properties 文件中设置某些附加属性。

注意：将此过程用于支持的 webMethods Broker 服务器版本，其中 6.5 版除外。有关支持版本的更多信息，请参阅《兼容性指南》。

可以使用 wmBrokerPwdEncryptor 脚本指定要使用的相应证书文件的路径以及创建 SSL 连接的加密密码。

遵循这些步骤:

1. 在文本编辑器中打开 `wmBrokerPwdEncryptor` 脚本，并将 `JAVA_HOME` 环境变量设置为相应的目录。例如：

```
set JAVA_HOME=C:\webMethods7\jvm\win150\jre
```

2. 运行 `wmBrokerPwdEncryptor` 脚本并输入 `y` 以配置 SSL 连接属性。

- a. 系统提示您输入 `keystore` 文件的路径时，请输入 Broker 的 PKCS12 密钥库证书文件的路径。

- b. 系统提示您输入 `truststore` 文件的路径时，请输入 Broker 的 Java 密钥库 (JKS) 信任库证书文件的路径。

- c. 系统提示您输入 `keystore/证书文件的密码` 时，请输入用于验证该文件的密码。

- d. 按 `Enter` 键完成配置。

`wmBrokerPwdEncryptor` 脚本执行以下操作：

- 加密 `SSL keystore` 或证书密码。
- 在 `WilyWMBrokerMonitor.properties` 文件中，更新 Broker 的相应属性。

3. 在文本编辑器中打开 `WilyWMBrokerMonitor.properties` 文件。

- a. 将 `wily.webmethods.broker.connection.ssl_encrypted` 属性设置为 `true`，以启用对 SSL 连接的加密。例如：

```
wily.webmethods.broker.connection.ssl_encrypted=true
```

- b. 将 `wily.webmethods.broker.ssl.distinguished_name` 属性设置为用于连接到 Broker 服务器的完全可分辨名称。例如：

```
wily.webmethods.broker.ssl.distinguished_name=cn=vespa09,dc=test,dc=org
```

- c. 保存并关闭文件。

详细信息:

[配置适用于 webMethods Broker 6.5 的 SSL 连接 \(p. 219\)](#)

配置适用于 webMethods Broker 6.5 的 SSL 连接

如果您在客户端组与 WebMethods Broker 服务器之间使用安全套接字层 (SSL) 连接，则代理必须提供待确认的签名证书。要启用对 SSL 连接的支持，请为 SSL 证书创建加密密码，并在 WilyWMBrokerMonitor.properties 文件中设置某些附加属性。

注意：您设置的具体属性由您正在使用的 webMethods Broker 版本决定。对 webMethods Broker 6.5 请执行以下步骤。有关支持的 webMethods Broker 版本，请参阅《兼容性指南》。

可以使用 `wmBrokerPwdEncryptor` 脚本指定要使用的相应证书文件的路径以及创建 SSL 连接的加密密码。

遵循这些步骤：

1. 在文本编辑器中打开 `wmBrokerPwdEncryptor` 脚本，并将 `JAVA_HOME` 环境变量设置为相应的目录。例如：

```
set JAVA_HOME=C:\webMethods<version_number>\jvm\win150\jre
```

2. 运行 `wmBrokerPwdEncryptor` 脚本并输入 `y` 以配置 SSL 连接属性。
 - a. 系统提示您输入 `keystore` 文件的路径时，请按 `Enter` 键将此属性留空。
 - b. 系统提示您输入 `truststore` 文件的路径时，请按 `Enter` 键将此属性留空。
 - c. 系统提示您输入证书文件的路径时，请输入 Broker 的 SSL 证书文件的路径。

Broker 的证书通常采用 PKCS10 格式。可以使用 WebMethods Broker 证书管理器 (`awcert`) 生成该文件。

- d. 系统提示您输入 `keystore`/证书文件的密码时，请输入用于验证该文件的密码。
- e. 按 `Enter` 键完成配置。

`wmBrokerPwdEncryptor` 脚本执行以下操作：

- 加密 SSL `keystore` 或证书密码。
- 在 `WilyWMBrokerMonitor.properties` 文件中，更新 Broker 的相应属性。

3. 在文本编辑器中打开 `WilyWMBrokerMonitor.properties` 文件。
 - a. 将 `wily.webmethods.broker.connection.ssl_encrypted` 属性设置为 `true`，以启用对 SSL 连接的加密。例如：

```
wily.webmethods.broker.connection.ssl_encrypted=true
```

- b. 将 wily.webmethods.broker.ssl.distinguished_name 属性设置为用于连接到 Broker 服务器的完全可分辨名称。例如：

```
wily.webmethods.broker.ssl.distinguished_name=cn=vespa09,dc=test,dc=org
```

- c. 保存并关闭文件。

4. 执行与您正在使用的操作环境相对应的操作：

- 在 Windows 中，将文件 awssl65jn.dll 从 <webMethods_Home>\common\lib 目录复制到 WilyWMBrokerMonitor\lib 目录。
- 在 UNIX 中，将文件 libawssl65jn.so 从 <webMethods_Home>/common/bin 目录复制到 WilyWMBrokerMonitor/lib 目录。

详细信息：

[为 WebMethods Broker 配置 SSL 连接 \(p. 217\)](#)

启动代理

在为代理配置连接和监控属性之后，您可以启动代理。可使用代理启动脚本来启动代理并开始监控 Broker 服务器和客户端。

要启动代理以监控 webMethods Broker，请运行 WmBrokerAgent 启动脚本。

此时将开始对 webMethods Broker 服务器进行监控。

启用 webMethods Broker 的企业管理器扩展

安装企业管理器时，CA APM for webMethods Broker 文件默认安装在 <EM_Home>/examples 目录中。要启用 CA APM for webMethods Broker，需要将文件从 examples 目录复制或移动到企业管理器主目录中。

遵循这些步骤：

1. 确认 SOAExtensionForWebMethodsBroker 目录位于 <EM_Home>/examples 目录中。
2. 将文件从 <EM_Home>/examples/SOAExtensionForWebMethodsBroker 目录复制到企业管理器目录结构中的对应位置。例如，将文件从 <EM_Home>/examples/SOAExtensionForWebMethodsBroker/ext 目录复制到 <EM_Home>/ext 目录。

3. 如果企业管理器在群集环境中用作收集器，请从 `<EM_Home>/config/modules` 目录中删除 webMethods Broker 管理模块 `WebMethodsBrokerManagementModule.jar`。

注意：应仅将管理模块复制到用作 MOM 计算机的企业管理器上的 `<企业管理器主目录>/config/modules` 目录中。其他所有文件和脚本应同时安装在收集器企业管理器和 MOM 企业管理器中。

4. 重新启动 Workstation，以加载特定于 SOA Extension for webMethods Broker 的显示板和“概览”选项卡。

使用显示板监控 webMethods Broker

SOA Extension for webMethods Broker 包括多个预配置的显示板，您可以使用这些显示板监控应用程序环境的总体运行状况。显示板汇集已部署代理中的数据，以汇总性能信息并帮助您快速诊断和解决问题。

通常，将显示板用作监控环境的起点，因为使用显示板可以：

- 一目了然地监控 webMethods Broker 服务器的关键组件的总体运行状况、性能、可用性和当前状态。
- 低级别度量标准发出已超过警告或危险阈值的信号时，及早获取生产应用环境中存在潜在问题的通知。
- 深入分析性能信息，以隔离并确定哪些 Broker、客户端组、客户端或文档类型遇到交付或发布延迟或最为繁忙。

预配置的 WebMethods Broker 显示板在 WebMethods 的企业管理器扩展中打包成 WebMethods Broker 管理模块 `WebMethodsBrokerManagementModule.jar` 的一部分。

WebMethods Broker 管理模块为 WebMethods Broker 提供以下预配置的显示板：

WebMethods Broker—概览

WebMethods Broker 服务器的关键活动和存储统计信息的顶级概览，其中包括：

- 已发布文档和已排队文档的图表和报警指示器。
- 客户端队列长度和已检索文档的图表。
- 队列长度、队列大小以及未确认文档数的报警指示器。

WebMethods Broker—Broker

概述了所有 Broker 实例的状态，其中包括具有以下特点的 Broker 列表：

- 具有最多客户端。
- 包含最多文档类型。
- 具有最多已发布文档。
- 最多的已排队文档。
- 重试排队中包含最新文档。
- 重试排队中包含最大数量的尝试交付事件。
- “重试”队列中最多的已发布文档。

WebMethods Broker—客户端

概述了所有 Broker 客户端组和客户端的状态，其中包括以下列表：

- 具有最多已交付文档的客户端组。
- 发布文档最多的客户端组。
- 发送文档最多的客户端。
- 发布文档最多的客户端。
- 已排队文档最多的客户端。
- 检索事件最多的客户端。

在此显示板中，您可以双击“客户端详细信息”链接以显示 **WebMethods Broker—客户端详细信息** 显示板，其中列出了具有最多已排队文档的客户端的队列大小、队列长度、扫描次数、最大队列长度以及未确认的最大文档数。

WebMethods Broker—文档

概述了所有 Broker 文档类型的状态，其中包括接收和发送最频繁的文档类型列表，交付、发布和接收最频繁且具有最多客户端订阅的文档类型列表。

WebMethods Broker—区域

所有 Broker 区域的汇总状态，包括以下列表：

- 已排队文档最多的区域。
- 具有最多已置入队列、已转发和已接收文档类型的区域。
- 具有最大队列大小、最大当前队列长度以及最大队列长度的区域。

遵循这些步骤：

1. 启动企业管理器（如果目前未运行）。
2. 启动 Workstation 并登录到已安装 SOA Extension for webMethods 的企业管理器。
3. 单击“Workstation”>“新建控制台”。
4. 从“显示板”下拉列表中选择某个 webMethods Broker 显示板。
例如，选择“WebMethods Broker—概览”显示板以查看 WebMethods Broker 度量标准概览。
5. 双击其他选项卡或报警打开相关显示板，以查看更详细的信息。
例如，双击“客户端”选项卡以查看与 WebMethods Broker 客户端组和客户端有关的更多详细信息。
6. 双击显示板中的特定 Broker、客户端、文档类型或区域度量标准，以打开调查器进行进一步分析。
例如，在“WebMethods Broker—客户端详细信息”显示板中，可以双击队列中排在最前面的客户端，以打开已选择了该客户端的“事件”>“已排队”度量标准的调查程序。

了解并查看 Broker 的度量标准

WmBrokerAgent 可以监控 Broker 服务器和客户端，以便在 <端口号> 节点的 <主机名> 上的 Broker 服务器下提供与其总体运行状况有关的数据。

可以使用以下度量标准类别的度量标准监控 WebMethods Broker 服务器和客户端的性能和运行状况：

Broker

在“Broker”节点下，可以查找单个 Broker 实例的度量标准，例如客户端数、订阅数以及已发布和已交付文档数。

客户端组

在“客户端组”节点下，可以查找已在 WilyWMBrokerMonitor.properties 文件中确定要监控的客户端组及其关联客户端的度量标准（例如，已发布和已交付文档数）。

文档类型

在“文档类型”节点下，可以查找已定义的各种文档类型的度量标准。文档类型包含一组字段，定义用于在合作伙伴或程序之间交换信息的文档结构和内容。

重试排队

在“重试排队”节点下，可以查找内部重试排队的度量标准。

区域统计信息

在“区域统计信息”节点下，可以查找各个区域的度量标准。

跟踪

在“跟踪”节点下，可以查找内部跟踪队列的度量标准。

使用率

在“使用率”节点下，可以查找用于描述 Broker 配置和数据存储统计信息的度量标准。

注意：有关每个度量标准类别中提供的度量标准的信息，请参阅该类别的相应部分。

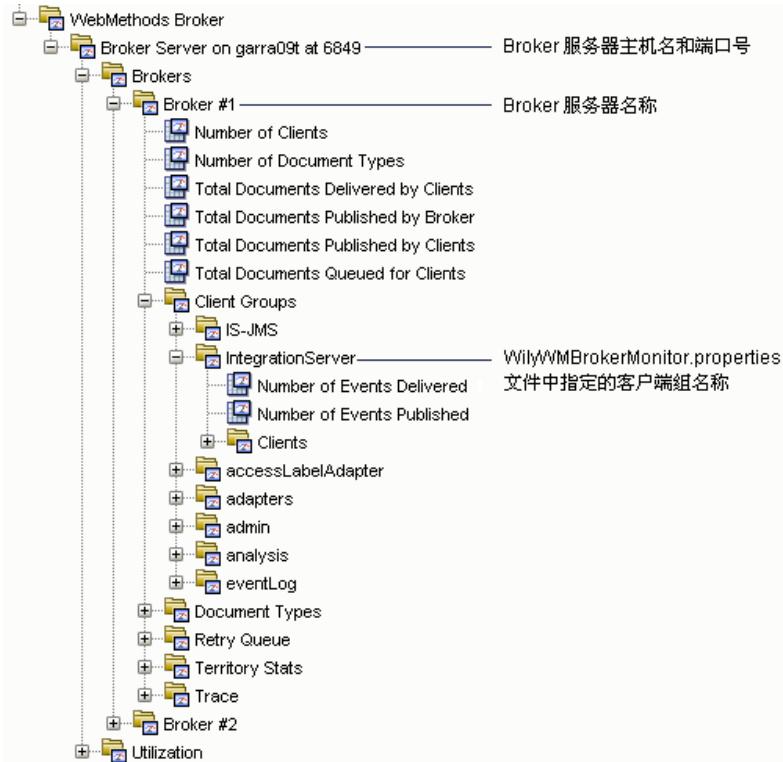
查看调查器中的 webMethods Broker 度量标准摘要并进行导航：

1. 展开代理节点以及 *<port_number>* 节点的 *<host_name>* 上的 Broker 服务器，然后单击“Broker”以显示“概览”选项卡，其中列出了监控的所有 Broker 服务器实例以及与已交付、已发布和已排队的文档数有关的摘要信息。
2. 选择特定 Broker 名称，以在“概览”选项卡上以图形格式显示与该实例的已交付和已发布文档有关的摘要信息。例如，如果选择了单个 Broker 名称，“概览”选项卡将以图形格式显示以下内容：
 - 交付的文档总数 (按客户端)
 - 发布的文档总数 (按客户端)
 - 发布的文档总数 (按 Broker)
 - 客户端排队等候的文档总数
 - 客户端数
 - 文档类型数
3. 展开单个 Broker 名称，然后单击任何子节点以显示与该度量标准类别有关的摘要信息。例如，选择“客户端组”节点，以在“概览”选项卡上显示 Broker 的客户端组列表以及每个客户端组的已交付和已发布文档摘要。

查看调查器中的 webMethods Broker 度量标准节点并进行导航：

1. 展开代理以及 *<port_number>* 节点的 *<host_name>* 上的 Broker 服务器，然后展开“Broker”节点以显示正在监控的 Broker 服务器名称。
2. 展开单个 *Broker_server_name* 节点，以显示该 Broker 服务器实例的度量标准。

3. 展开“客户端组”节点，以查看在 *WilyWMBrokerMonitor.properties* 文件中指定要监控的客户端组，然后展开特定的 *<client_group_name>*，以显示该客户端组的度量标准。例如：



4. 展开“客户端”节点，然后展开单个 *<client_name>* 子节点，以显示该客户端的度量标准。 -
5. 展开“文档类型”节点，然后展开单个 *<document_type_name>* 子节点，以显示选定 Broker 的文档类型度量标准。 -
6. 展开“重试排队”、“区域统计信息”或“跟踪”子节点，以显示选定 Broker 的重试排队、区域或跟踪度量标准。
7. 展开“使用率”>“存储统计信息”，然后展开“CONFIG”或“DATA”子节点，以显示与选定 Broker 服务器的配置或数据存储有关的信息。 -

Broker 的度量标准

WebMethods Broker 服务器是一台主计算机，用于管理客户端、Broker 和各种应用程序之间的文档流。客户端程序以文档形式发布和订阅信息，Broker 服务器会自动对文档进行路由、排队和筛选。每个 Broker 服务器都有一个或多个单个 Broker 驻留在上面。

每个单个 Broker 都可以接收客户端连接并存储与自身文档类型、客户端队列和订阅有关的信息。客户端发布文档时，Broker 会确定哪些 Broker 客户端已订阅接收该类型的文档，并将该文档置于相应的 Broker 客户端队列中。

以下度量标准适用于单个 Broker 实例名称的“Broker”子节点下要监控的 Broker 实例：

客户端数

选定 Broker 服务器的客户端总数。

文档类型数

为选定 Broker 服务器定义的文档类型总数。

交付的文档总数 (按客户端)

由客户端交付到选定 Broker 服务器的文档总数。

发布的文档总数 (按 Broker)

由选定 Broker 服务器发布到其客户端的文档总数。

发布的文档总数 (按客户端)

由客户端发布到选定 Broker 服务器的文档总数。

客户端排队等候的文档总数

由选定 Broker 服务器排队以交付到其客户端的文档总数。

客户端组的度量标准

在 WebMethods Broker 中，客户端组允许您同时设置多个 Broker 客户端的属性。以下度量标准适用于单个客户端组名称的“客户端组”子节点下要监控的客户端组：

已发送事件数

选定客户端组中所有客户端的文档交付事件总数。

已发布事件数

选定客户端组中所有客户端的文档发布事件的总数。

客户端的度量标准

Broker 客户端是由客户端程序创建并用于连接到特定 Broker 的 WebMethods 对象。客户端程序创建一个或多个 Broker 客户端，并根据需要连接到一个或多个 Broker。对于 WebMethods Broker 客户端，会将文档处理任务记录为事件。例如，如果成功将文档交付到客户端，则该操作将记录为文档交付事件。

可以使用以下度量标准类别中的度量标准（位于“客户端”>“*<client_name>*”节点下）监控 Broker 客户端：

Events

事件会记录文档处理任务。例如，如果成功将文档交付到客户端，则该操作将记录为文档交付事件。

队列

队列用于按接收顺序存储异步事件，直至可以处理这些事件。

Session

会话会向 Broker 展示客户端连接。

事件度量标准

以下度量标准在选定 Broker 客户端的各个客户端的“事件”子节点下可用：-

已发送

由选定客户端记录的文档交付事件总数。

已发布

由选定客户端记录的文档发布事件的总数。

已排队

选定客户端已排队的文档总数。

已检索

选定客户端已从队列中检索的文档总数。

未确认

客户端程序已从客户端队列中检索但尚未向 Broker 确认的文档总数。

Broker 在收到确认后会从客户端队列中删除文档。

队列度量标准

在选定 Broker 客户端的单个客户端的“队列”子节点下提供了以下度量标准： -

字节大小

队列大小（以字节为单位）。

最大长度

记录的队列长度的最大值。

长度

客户端队列中当前具有的文档总数，其中包括客户端程序尚未检索的文档以及客户端程序已检索但尚未确认的文档。

会话度量标准

在选定 Broker 客户端的单个客户端的“会话”子节点下提供了以下度量标准： -

上次连接

最新创建的客户端会话的会话标识符。

上次断开连接

最近与 Broker 断开连接的客户端会话的会话标识符。

上次检索

最近从此客户端队列中检索到文档的客户端会话的会话标识符。

文档类型的度量标准

文档是通过网络从发布者经 Broker 传递给订户的消息。每个单独的文档是文档类型的一个实例。每种文档类型都具有唯一名称以及与其关联的属性，例如文档文件夹名称、创建时间、Broker 客户端已发布和检索的次数以及订阅数。

以下度量标准适用于各种文档类型的“文档类型”子节点下要监控的 Broker 实例：

客户端订阅数

选定 Broker 上客户端对此文档类型的订阅总数。

已发送事件数

选定 Broker 交付的选定文档类型的文档总数。

已发布事件数

发布到选定 Broker 的选定文档类型的文档总数。

收到的转发次数

从同一区域的其他 Broker 中转发的选定文档类型的文档总数。

可执行发布的组数

可发布选定文档类型的客户端组总数。

可以订阅的组数

可以订阅选定文档类型的客户端组的总数。

重试排队的度量标准

WebMethods Broker 重试排队显示 Broker 用来存储其必须重试的请求的内部队列统计信息。这些统计信息通常由技术支持人员用于故障排除。

以下度量标准适用于在“重试排队”子节点下要监控的 Broker 实例：

当前事件

重试排队中当前交付的文档总数。

当前发布数

“重试”队列中当前要发布的文档数。

最大事件数

要交付且可存放在重试排队中的最大文档数。

最大发布数

可保留在“重试”队列中的要发布的最大文档数。

下一个操作序号

标识下一个要重试的请求的序列号。

尝试次数

Broker 已尝试交付“重试排队”中的文档次数。

保留的保证事件

在重试排队中为其保留空间且标记为保证交付的文档数。

保留的总发布

在重试排队中为其保留空间且要发布的文档总数。

保留的可变事件

在重试排队中为其保留空间且要交付的可变文档数。

保留的可变发布

在“重试”队列中为其保留空间的要发布的可变文档数。

区域统计信息的度量标准

WebMethods Broker 区域统计信息按区域提供与事件和队列有关的信息。对于每个区域，这些统计信息都描述了已置入队列、已转发和已接收的事件数以及队列大小和长度。

以下度量标准适用于“区域统计信息”>“*<territory_name>*”>“事件”子节点下要监控的 Broker 实例：

已入列数

已置入区域队列中的文档交付事件数。

已转发次数

为区域转发的文档发送事件数。

已接收次数

为区域接收的文档发送事件数。

以下度量标准可用于您在“区域统计信息”>*<territory_name>*>“队列”子节点下监控的 Broker 实例：

字节大小

选定区域的队列总大小（字节）。

最大长度

为区域记录的队列长度的最大值。

长度

区域队列中当前的文档总数，包括客户端程序尚未检索的文档和客户端程序已检索但尚未确认的文档。

跟踪队列的度量标准

WebMethods Broker 跟踪队列是存储跟踪事件的临时存储区域。系统会提出跟踪事件以对 Integration Server 上的发布或确认触发器做出响应。由于它是内部队列活动事件和跟踪事件，因此跟踪队列的统计信息通常由技术支持人员用于故障排除。以下度量标准可用于您在“跟踪”子节点下监控的 Broker 实例：

已排队的事件数

选定 Broker 的跟踪队列中的文档总数。

队列字节大小

选定 Broker 的“跟踪”队列的总大小（字节）。

队列长度

选定 Broker 的“跟踪”队列中的项目数。

使用率的度量标准

在“使用率”>“存储统计信息”子节点下提供可用于 Broker 服务器的配置 (CONFIG) 和运行时 (DATA) 数据存储的下列度量标准：

当前保留的 KB 数

为选定存储文件保留的大小 (KB)。

当前使用的 KB 数

选定存储文件的当前大小 (KB)。

可用的最大 KB 数

允许选定存储文件增长的最大大小 (KB)。

最大事务 (KB)

Broker 服务器的最大事务大小 (KB)。

会话 URL

存储设备配置的位置和类型。

将“使用率”>“存储统计信息”下的度量标准分成 CONFIG 和 DATA 存储。CONFIG 度量标准描述了需要定义 Broker 服务器、区域、Broker、文档类型和统计信息的数据存储。DATA 度量标准描述了客户端队列、文档和日志所需的数据存储。类似的度量标准也适用于“文件统计信息”>“*<file_name>*”子节点下的 Broker 服务器的单个配置和数据存储文件名称。-

查看默认的 Broker 度量标准组

SOA Extension for webMethods 包括定义默认显示板和报警用的默认度量标准组。您还可以在自定义显示板和报警中使用这些默认度量标准组。

在 WebMethods Broker 的企业管理器扩展中将默认度量标准组打包成 WebMethods Broker 管理模块 *WebMethodsBrokerManagementModule.jar* 的一部分。

可使用 CA Introscope® Workstation 管理模块编辑器查看默认度量标准组。还可以将 webMethods Broker 管理模块扩展为包括自定义度量标准组，或在自定义显示板或报警中使用默认度量标准组。

查看 webMethods Broker 代理的默认度量标准组

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>“WebMethodsBrokerManagementModule (*超级域*)”。
3. 展开“度量标准组”节点，查看为 webMethods 管理模块定义的所有度量标准组。
4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

查看默认 Broker 报警

SOA Extension for webMethods 包括预配置的显示板中使用的默认报警定义。您也可以在自定义的显示板中使用这些默认报警。大部分默认报警预配置有默认的“警告”和“危险”阈值，如果超过阈值或重要级别增加，则会向控制台发送通知。

默认报警定义作为 webMethods Broker 管理模块 (*WebMethodsBrokerManagementModule.jar*) 的一部分，打包在 webMethods Broker 的企业管理器扩展中。

可以使用 CA Introscope® Workstation 管理模块编辑器查看默认报警定义。还可以将 webMethods Broker 管理模块扩展为包括自定义报警定义和通知类型，或在自定义显示板中使用默认报警定义。

查看 webMethods Broker 代理的默认报警定义

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>“WebMethodsBrokerManagementModule (*超级域*)”。

3. 展开“报警”节点，查看为 webMethods Broker 管理模块定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

您应当注意查看“警告”和“危险”阈值的默认设置并根据需要调整这些值，并添加通知或纠正措施。

第 11 章： 监控 webMethods Integration Server

Software AG WebMethods 产品套件是一个由多个基础架构组件和功能组成的 SOA 平台，使组织能够创建、安排和集成业务流程和 Web 服务。通过 SOA Extension for webMethods，您能够监控 WebMethods 基础架构的许多主要元素。

本节介绍了 WebMethods 服务以及可用来监控和分析 webMethods Integration Server 的运行状况和操作的相关显示板和度量标准。

此部分包含以下主题：

- [关于 webMethods Integration Server \(p. 235\)](#)
- [如何启用对 webMethods Integration Server 的监控 \(p. 238\)](#)
- [使用显示板监控 webMethods \(p. 241\)](#)
- [筛选监控和显示的服务 \(p. 246\)](#)
- [查看 webMethods 的度量标准并进行导航 \(p. 248\)](#)
- [查看默认的 webMethods 度量标准组 \(p. 261\)](#)
- [查看默认 webMethods 报警 \(p. 262\)](#)
- [查看 webMethods 依赖关系 \(p. 262\)](#)
- [webMethods 的跟踪事务 \(p. 263\)](#)

关于 webMethods Integration Server

利用 WebMethods Integration Server，公司能够公开和集成新业务服务和现有业务服务。WebMethods Integration Server 中包含一些工具，利用这些工具，您能够设计、测试和部署新服务，并且能够使松散耦合服务和旧系统自动化并将其安排、整合到经过改进的业务流程中。

WebMethods Integration Server 为运行和分发服务提供了一个集中式平台。它可以接收和解释客户端请求，标识和调用请求的服务，以预期格式将数据传递到正在运行的服务，接收服务产生的输出，以及将输出返回到客户端。

作为集成平台，WebMethods 主要用于协调应用程序服务器、数据库和自定义应用程序之间的操作，以及使公司或贸易伙伴能够交换电子文档。

可使用以下顶级组件的度量标准监控 webMethods Integration Server 的操作：

适配器

适配器使外部应用程序能够通过服务接口的通用适配器框架与 WebMethods 集成。适配器由以下各项组成：

适配器连接, 使 Integration Server 能够在运行时连接到外部资源或系统。-

适配器服务, 在外部资源上启动 Integration Server 时运行。

适配器通知, 可以监控外部资源, 并在发生 Integration Server 未启动的事件时通知 Integration Server。

通过 SOA Extension for webMethods, 您可以使用“适配器连接池”、“适配器服务”和“适配器通知”节点下的度量标准, 监控已为 WebMethods Integration Server 部署的所有适配器的性能和总体运行状况。

业务流程

业务流程是一系列相关联的业务任务, 这些任务根据一组特定的业务规则按特定顺序执行。大多数业务流程还需要与多个系统和具有不同角色的多个用户进行交互。

例如, 您可能具有完成以下任务所需的各种业务流程: 准备迎接新员工、处理订单或交付发票。每个业务流程可能包括多项业务任务, 例如, 为新员工分配办公空间, 将员工添加到人力资源 (HR) 系统, 以及订购办公设备和办公用品。

通过 SOA Extension for webMethods, 您可以使用“WebMethods”>“业务流程”节点下的度量标准监控已定义的业务流程的性能和总体运行状况。

流服务

流服务是以 WebMethods 流语言编写并部署在 WebMethods Integration Server 上的服务。流服务可以调用在 WebMethods 服务器上运行的任何服务, 包括其他流服务、用户定义的服务、内置服务以及其他提供商提供的服务(例如 WebMethods 适配器或 .NET 插件)。-

通过 SOA Extension for webMethods, 您可以监控所有流服务的性能和总体运行状况, 或筛选流服务以排除不需要监控的流服务。单个流的度量标准可应用到“WebMethods”>“流服务”节点下列出的流服务。

Java 服务

Java 服务是以 Java 语言或其他语言编写的内置服务和用户定义的服务，使用 Java 类进行包装，并且可以显示为 WebMethods Integration Server 上的服务。

使用 SOA Extension for webMethods，您可以监控所有 Java 服务的性能和总体运行状况，或者筛选 Java 服务以排除您不想监控的 Java 服务。每个 Java 类中的 Java 方法的度量标准均在“WebMethods”>“Java 服务”节点下列出。

JDBC 连接池

WebMethods Integration Server 使用 Java 数据库连接通过网络交流和传输信息。

通过 SOA Extension for webMethods，您可以使用“WebMethods”>“JDBC 连接池”节点下的度量标准监控 JDBC 连接的可用性。

线程池

WebMethods Integration Server 使用线程执行服务、从 WebMethods Broker 中检索文档以及执行触发器。

通过 SOA Extension for webMethods，您可以使用“WebMethods”>“线程池”节点下的度量标准监控线程的可用性。

Trading Networks

利用 Trading Networks，组织能够交换文档，以建立并丰富企业之间的关系。

通过 SOA Extension for webMethods，您可以使用“WebMethods”>“Trading Networks”节点下的度量标准监控文档识别和处理。

触发器

触发器可建立对可发布文档类型的订阅，并指定如何处理这些文档的实例。

Broker 触发器或本地触发器是一种具有以下功能的触发器，即可订阅和处理在 Integration Server 上本地发布的文档或交付到 Broker 的文档。Broker 触发器通常与异步适配器通知相关联。

JMS 触发器是一种具有以下功能的触发器，即可从 JMS 提供程序的某个目标（队列或主题）接收消息，然后对这些消息进行处理。

通过 SOA Extension for webMethods，您可以使用“WebMethods”>“触发器”节点下的度量标准监控触发器。

WebServices

Web 服务度量标准表示客户端和服务器业务服务端点和每个服务内的关联操作。

通过 SOA Extension for webMethods，您可以监控“WebMethods”>“Web 服务”节点下的客户端和服务器 Web 服务端点的性能和总体运行状况。

XSLT 服务

在 WebMethods 中，您可以使用 XSLT 样式表将 XML 数据转换为其他格式，并包括其他服务的转换。

通过 SOA Extension for webMethods，您可以监控“WebMethods”>“XSLT 服务”节点下的 XSL 转换的性能和总体运行状况。

如何启用对 webMethods Integration Server 的监控

作为管理员，请通过执行以下高级步骤为 webMethods Integration Server 启用监控功能：

1. 确认已安装 webMethods Integration Server。

注意：有关 webMethods Integration Server 要求，请参阅《兼容性指南》。

2. 确认您是否有支持的 webMethods WmPRT.jar 文件版本。在 Web 浏览器中打开以下 URL。确认已列出该版本级别。

`http://<Integration_Server_Hostname>:<port_number>/wmRoot/Updates.dsp`

3. 确认已安装并启用代理和 CA APM for SOA。

4. 通过配置代理配置文件，使代理能够使用 CA APM for webMethods Integration Server。

重要信息！如果已使用独立代理安装程序或响应文件在代理中启用了 CA APM for webMethods Integration Server，请跳过该步骤。

5. [启用企业管理器扩展 \(p. 241\)](#)。

详细信息：

[手工启用代理以监控 webMethods Integration Server \(p. 239\)](#)

手工启用代理以监控 webMethods Integration Server

您可以使用以下方法之一启用对 webMethods Integration 的监控：

- 安装代理时选择“CA APMfor webMethods Integration Server”。代理配置文件将自动配置为默认设置。无需执行其他步骤。
- 安装代理时不选择“CA APM for webMethods Integration Server”。使用下列步骤，手工配置代理配置文件。

注意：有关 webMethods Integration Server 要求，请参阅《兼容性指南》。

遵循这些步骤：

1. 确认已安装并启用默认代理和 CA APM for SOA。
2. 确认 CA APM for WebMethods Integration Server 目录位于<代理主目录>/examples 目录中。
3. 将文件从<代理主目录>/examples/SOAExtensionForWebMethodsIS 目录复制到相应的<代理主目录>目录中。
例如，将<代理主目录>/examples/SOAExtensionForWebMethodsIS/ext 中的文件复制到<代理主目录>/core/ext 目录中。
4. 在文本编辑器中打开<代理主目录>/core/config/IntroscopeAgent.profile 文件。
 - a. 将 webmethods.pbl（默认代理 pbl）和 spm pbl 添加到 IntroscopeAgent.profile 文件的 introscope.autoprobe.directivesFile 属性中。例如：

```
introscope.autoprobe.directivesFile=default-typical.pbl,hotdeploy,spm.pbl,webmethods.pbl
```


注意：您可以使用 webMethods Integration Server 的 [ProbeBuilder 指令文件](#) (p. 240) 自定义跟踪。
 - b. 将 introscope.agent.agentName 属性值设置为 webMethods Agent。例如：

```
introscope.agent.agentName=webMethods Agent
```


此步骤可确保在显示板中仅显示 webMethods 数据。
 - c. 如果需要更改默认设置，请在 IntroscopeAgent.profile 文件中修改其他任何属性。

- d. 保存并关闭 IntroscopeAgent.profile 文件。
5. 仅对 webMethods Integration Server 7.x 有效，在 <代理主目录>/core/config/webmethods-toggles.pbd 文件中执行以下编辑：
 - a. 在“业务流程”部分取消注释 BProcAndStepTracing71 标志。
打开 BProcAndStepTracing71 标志。
关闭 BProcAndStepTracing80 标志。
关闭 BProcAndStepTracing82 标志。
 - b. 在“Web 服务”部分中取消注释下列选项：
关闭 WM8xWebServicesTracing 标志。
打开 WM7xWebServicesTracing 标志。
关闭 webMethods8xwsClientTracing 标志。
打开 webMethods7xwsClientTracing 标志。
关闭 webMethods65wsClientTracing 标志。
 - c. 保存并关闭 webmethods-toggles.pbd 文件。
6. 重新启动 webMethods Integration Server 进程。

关于 webMethods Integration Server 的指令文件

在 IntroscopeAgent.profile 中设置 introscope.autoprobe.directivesFile 属性时，可为 webMethods Integration Server 启用默认检测。如果要修改此默认监控功能，则可在 webmethods.pbd 或 webmethods-toggles.pbd 中自定义 ProbeBuilder 指令。例如，可以使用切换文件通过打开或关闭对特定跟踪器组的跟踪，来微调针对特定组件的监控。

webmethods.pbd

监控 webMethods Integration Server 上的主要服务，包括流服务、Java 服务、Trading Network、Web 服务、XSLT 服务以及业务流程。

webmethods-toggles.pbd

在打开或关闭对 webMethods Integration Server 组件的监控之间来回切换。

webmethods.pbl

提供用于监控 webMethods Integration Server 的 ProbeBuilder 指令文件的默认列表：

- webmethods.pbd
- webmethods-toggles.pbd

启用企业管理器扩展

安装企业管理器时，CA APM for webMethods Integration Server 文件默认安装在 <企业管理器主目录>/examples/SOAExtensionForWebMethodsIS 目录中。要启用 CA APM for webMethods Integration Server，需要将 webMethods Integration Server 的企业管理器文件从 <企业管理器主目录>/examples 目录复制或移动到企业管理器主目录中的适当位置。

注意：只有[在企业管理器上启用](#) (p. 35) CA APM for SOA 后，才能使用 CA APM for webMethods Integration Server。

遵循这些步骤：

1. 确认 CA APM for webMethods Integration Server 目录 SOAExtensionForWebMethodsIS 位于 <企业管理器主目录>/examples 目录中，然后将文件从 <企业管理器主目录>/examples/SOAExtensionForWebMethodsIS 目录复制到企业管理器目录结构中的对应位置。例如，将文件从 <企业管理器主目录>/examples/SOAExtensionForWebMethodsIS/ext 目录复制到 <企业管理器主目录>/ext 目录。
2. 如果企业管理器是群集环境中的收集器，则从 <企业管理器主目录>/config/modules 目录删除 webMethods Integration Server 管理模块 WebMethodsISManagementModule.jar。
应仅将管理模块复制到用作 MOM 计算机的企业管理器上的 <企业管理器主目录>/config/modules 目录中。其他所有文件和脚本应当同时安装在收集器企业管理器和 MOM 企业管理器中。
3. 重新启动 Workstation，以加载特定于 CA APM for webMethods Integration Server 的显示板和“概览”选项卡。

使用显示板监控 webMethods

SOA Extension for webMethods Integration Server 包括多个预配置的显示板，您可以使用这些显示板监控应用程序环境的总体运行状况。显示板汇集已部署代理中的数据，以汇总性能信息并帮助您快速诊断和解决问题。

通常，将显示板用作监控环境的起点，因为使用显示板可以：

- 一目了然地监控 webMethods Integration Server 的关键组件的总体运行状况、性能、可用性和当前状态。
- 低级别度量标准发出已超过警告或危险阈值的信号时，及早获取生产应用环境中存在潜在问题的通知。
- 深入了解性能信息，以便隔离并标识出现延迟或产生错误的 webMethods Integration Server 业务流程、服务或连接池。

预配置的 webMethods 显示板作为 webMethods Integration Server 管理模块 (*WebMethodsISManagementModule.jar*) 的一部分，打包在 webMethods 的企业管理器扩展中。

webMethods Management Module 为 webMethods Integration Server 提供了以下预配置的显示板：

WebMethods—主页

是 WebMethods Integration Server 及其主要组件的顶级体系结构概览，其中包括针对所有服务、业务流程、Trading Network 组件和外部后端系统的总体运行状况的报警指示器。

WebMethods—IS 服务概览

概述了流服务、Java 服务、XSLT 服务以及触发器的状态，其中包括“平均响应时间”的图表、针对错误和停顿的报警指示器等。

从该显示板中，可以双击“IS 最慢的服务”链接或任何“总体运行状况”标签，以显示“WebMethods—IS 最慢的服务”显示板，其中列出了最慢的流服务、Java 服务、XSLT 服务以及触发器列表。

WebMethods—业务流程

概述了所有业务流程的状态，其中包括同时处于过程级别和步骤级别的“平均响应时间”和“每个时间间隔的错误”的图表。

该显示板包括：在流程级别上执行了取消、重新启动、挂起和继续操作的报警指示器，在步骤级别上用于响应时间、停顿、并发调用和错误的报警指示器，以及最慢的业务流程的列表。

WebMethods—Web 服务概览

概述了所有客户端和服务器端 Web 服务的状态，其中包括突出显示客户端和服务器端性能的“平均响应时间”图表，最慢的客户端和服务器端服务的列表，以及用于客户端和服务器服务的响应时间、SOAP 故障、错误以及停顿的报警指示器。

从该显示板中，可以双击“Web 服务操作”链接或任何“总体运行状况”标签或图表，以显示“WebMethods—Web 服务操作”显示板，其中包括针对客户端和服务器端操作的“平均响应时间”图表，针对 SOAP 故障和停顿的报警指示器，以及最慢客户端和服务器端 Web 服务操作的列表。

WebMethods—适配器

概述了所有 webMethods 适配器的状态，其中包括针对所有适配器服务、适配器通知以及外部后端系统的“平均响应时间”图表。

对于适配器服务，该显示板还显示：

- “每个时间间隔的错误”和“停顿计数”图表
- 针对并发调用、错误以及停顿的报警指示器
- 最慢的适配器服务的列表

对于适配器通知和外部后端，该显示板还显示针对错误和停顿的报警指示器。

WebMethods—Trading Networks

概述了由 Trading Networks 处理的所有文档处理的状态，其中包括所有文档类型和处理规则的“平均响应时间”和“每个时间间隔的错误”图表，针对错误和停顿的报警指示器，以及所处理的最慢文档的列表。

WebMethods—连接和线程池

汇总了适配器连接池、JDBC 连接池和线程池的状态，其中包括适配器连接和 JDBC 连接的可用连接数及当前池大小的图表，以及线程池中正在使用的线程数和最大线程数的图表。

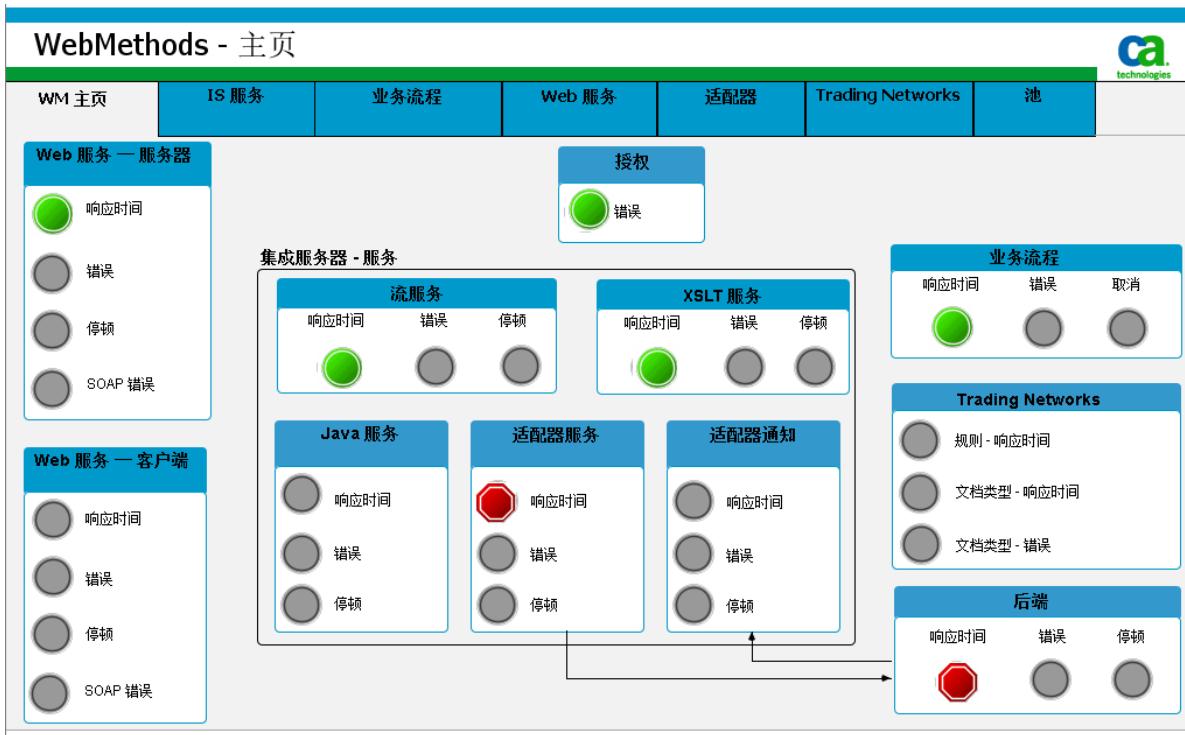
您可以使用 Workstation 控制台查看预配置的显示板。还可以将 webMethods Integration Server 管理模块扩展为包括自定义显示板，或者将默认显示板定义修改为包括自定义度量标准或报警。

注意：有关创建和修改显示板的详细信息，请参阅《CA APM 配置和管理指南》。

遵循这些步骤：

1. 启动企业管理器（如果目前未运行）。
2. 启动 Workstation 并登录到已安装 SOA Extension for webMethods 的企业管理器。
3. 单击“Workstation”>“新建控制台”。
4. 从“显示板”下拉列表中选择某个 webMethods 显示板。

例如，选择“WebMethods Home”显示板可查看 webMethods 关键组件和内部工作流的概览。

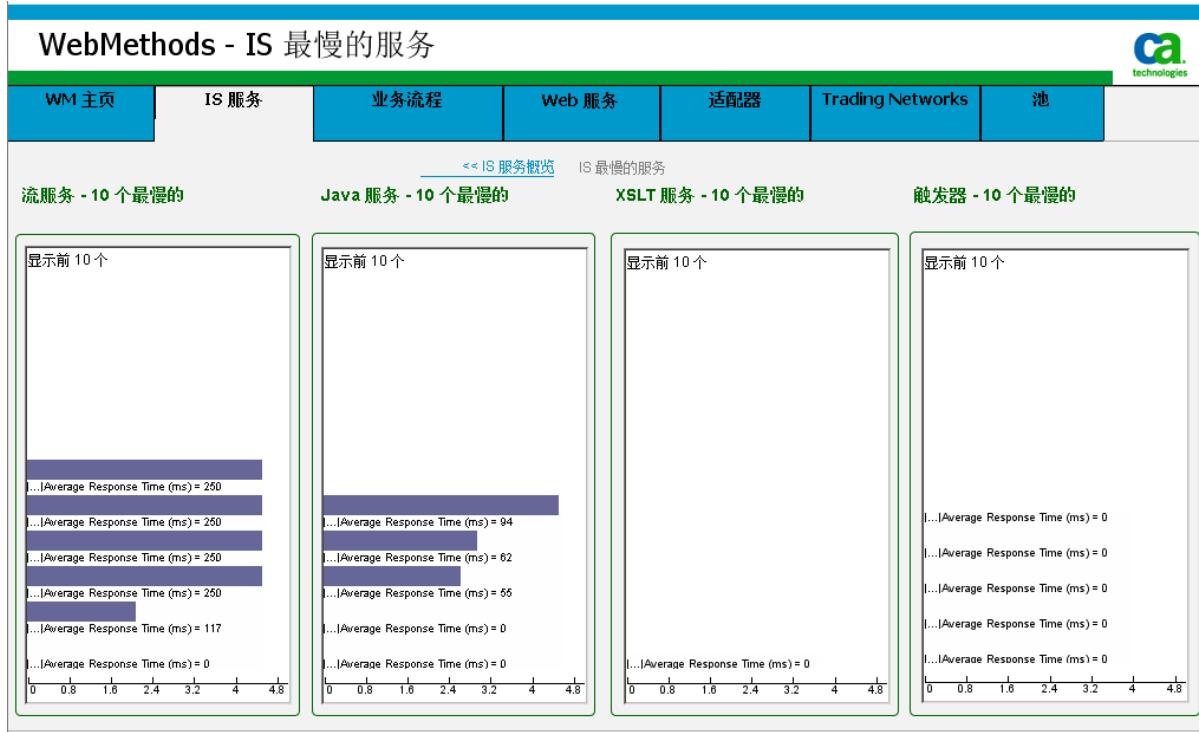


5. 双击其他选项卡或显示板中的报警，打开相关显示板以查看更多详细信息。

例如，双击“Java 服务”报警，可在“WebMethods—IS 服务概览”显示板中查看有关 webMethods Integration Server 流服务、Java 服务和 XSLT 服务的总体运行状况的更多详细信息。



6. 从“WebMethods—IS 服务概览”显示板中，可双击“IS 最慢的服务”以显示最慢的单个流服务、Java 服务、XSLT 服务以及触发器服务的列表。例如：



7. 双击显示板中的某个特定服务、业务流程或文档度量标准，以打开调查器进行进一步分析。

例如，从“WebMethods—IS 最慢的服务”显示板中，可双击最慢的流服务以打开该服务的“平均响应时间”处于选中状态的调查器。

筛选监控和显示的服务

对于 webMethods Integration Server，可以通过在配置文件中指定筛选来控制在调查器中监控并包括哪些流服务或 Java 服务。默认配置文件为 `wmExtension.config`。通过编辑此文件，可以使用用于确定要包括在监控范围内或从监控范围排除的流服务的正则表达式来创建筛选。

关于默认配置文件

默认配置文件 `wmExtension.config` 位于 <代理主目录>/common 目录下，并配置了默认的包含和排除筛选。例如，为排除 webMethods 内置服务，定义了以下默认筛选：

```
com.wily.wm.service.filter.exclude=wm.* , pub.*
```

为包含匹配的 `webMethods` 服务，配置文件还提供了以下默认筛选：

```
com.wily.wm.service.include=wm.tn:receive,wm.tn.route:routeBizdoc,pub.prt.tn:  
handleBizDoc
```

您可以修改默认配置文件，使其包含或排除其他服务（如有需要）或删除默认筛选（如果希望监控包括内置服务在内的所有 `webMethods` 服务）。如果 `wmExtension.config` 文件中没有定义任何筛选，则所有服务（包括内置服务）都会显示在调查器树中。

使用正则表达式包含和排除服务

在 `wmExtension.config` 配置文件内，可以使用 `com.wily.wm.service.filter.include` 和 `com.wily.wm.service.filter.exclude` 属性来确定可包含或排除的服务。例如，可以在 `com.wily.wm.service.filter.exclude` 属性中设置一个正则表达式来定义希望从监控范围内排除的服务。包含和排除属性可以包含任何适用于全限定服务名称的表达式。例如，要排除以字符串 `webservice` 结尾的所有流服务，请按如下方式设置排除属性：

```
com.wily.wm.service.filter.exclude=.*webservice
```

要定义多个筛选，请使用逗号分隔各个正则表达式。例如，要排除分别以 `wm.server` 和 `wm.tomcat` 开头的所有流服务和 Java 服务，可按如下方式设置 `com.wily.wm.service.filter.exclude` 属性：

```
com.wily.wm.service.filter.exclude=wm.server.*,wm.tomcat.*
```

与该正则表达式匹配的所有流服务和 Java 服务均会被排除，调查器中仅显示剩余的服务。但是，如果指定的正则表达式无效，则不会从监控范围内排除任何服务，而是显示包括内置服务在内的所有流服务和 Java 服务。

为配置文件指定其他位置

虽然默认的 `wmExtension.config` 文件位于 `<Agent_Home>/common` 目录中，但是您可以通过在服务器的启动脚本中指定其他位置来指定其他目录或其他配置文件。例如，如果备用 `wmExtension.config` 文件位于目录 `C:\CA-Introscope` 下，可以将 `com.wily.wm.service.filter.fileloc` 属性添加到服务器的启动脚本中的 Java 参数：

```
设置 JAVA_OPTS=%JAVA_OPTS%  
-Dcom.wily.wm.service.filter.fileloc=C:\CA-Introscope
```

如果指定的路径中存在名为 `wmExtension.config` 的文件，代理会检查该配置文件中的包含和排除属性，以确定要监控的流服务和 Java 服务。

如果在服务器的 Java 参数中指定了配置文件的路径，代理会使用该文件中指定的筛选来确定要排除的流服务和 Java 服务。如果未指定配置文件的路径，代理会使用 <代理主目录>/common/wmExtension.config 文件中定义的筛选。如果找不到 <代理主目录>/common/wmExtension.config 或备用 wmExtension.config 文件，则表明未执行任何筛选，将显示包括内置服务在内的所有流服务和 Java 服务。

查看 webMethods 的度量标准并进行导航

在调查器树中导航时，您可以查看 webMethods Integration Server 基础架构的大多数组件的标准 CA Introscope® 度量标准。收集标准度量标准的数据并将其聚合到在调查器树中显示为节点和子节点且特定于 webMethods 的度量标准类别中。- 显示的特定度量标准类别和节点名称取决于已在环境中部署的流程、服务和资源。

当浏览调查器树时，可以选择查看各个操作的低层度量标准或汇总度量标准（具体取决于所选的节点），这样将可以监控已通过 webMethods Integration Server 部署的各个服务的总体运行状况。

查看调查器中的 webMethods 度量标准并进行导航

1. 展开代理节点，然后单击“WebMethods”节点显示“概览”选项卡，其中列出了有关监控的所有 webMethods 流服务和业务流程的标准 CA Introscope® 度量标准的摘要信息。
2. 在列表中选择一个流服务或业务流程，以图表形式查看该服务或流程的所有标准度量标准。
3. 展开“WebMethods”节点，可显示 webMethods Integration Server 的顶层度量标准类别的子节点。
4. 单击或展开子节点，以显示带有该度量标准类别相关摘要信息的“概览”选项卡。例如，单击“Java 服务”节点，可在“概览”选项卡上显示 Java 服务的度量标准摘要。
5. 要查看有关各个业务流程、流服务、Java 服务或连接池以及与这些项相关联的度量标准的更详细信息，请展开相关子节点。

例如，可以展开“流服务”节点，随后展开具体的流服务名称和子文件夹，以显示单个流服务的度量标准。

适配器的度量标准

默认 webMethods 适配器可无缝连接到帮助组织实现业务流程实时执行的信息资源和企业应用程序。通过适配器，组织可使用 webMethods 连接到打包的应用程序（如 SAP 和 Oracle 应用程序包）或数据库（如 Microsoft SQL Server 和 Oracle RDBMS），并最大程度地减少自定义编程或耗时的集成开发。可用于部署的具体适配器取决于您使用的 webMethods Integration Server 版本。

适配器的度量标准提供了有关已部署的具体适配器的详细信息，通过这些度量标准可监控与这些适配器关联的连接、操作和事件。

您可以使用以下度量标准类别（位于“WebMethods”节点下）的度量标准来监控适配器：

适配器连接池

通过适配器连接，Integration Server 可在运行时连接到外部应用程序或信息存储。

适配器通知

通过适配器通知，您可以在发生由资源启动的事件时监控外部资源并通知 Integration Server。当事件发生时，适配器通知会向 webMethods Broker 发布一个文档。

适配器服务

通过适配器服务，您可以使用适配器与外部资源的连接从 Integration Server 启动针对资源的操作。

例如，webMethods 通过一个 JDBC 适配器提供了一套用户界面、服务和模板，使您可以使用 JDBC 驱动程序与数据库进行集成。可以使用“适配器连接池”类别中的度量标准监控与数据库的连接。

通过 JDBC 适配器服务，Integration Server 可以对数据库启动并执行数据库操作（例如插入、更新或删除数据）。通过适配器通知，您可以监控数据库并在特定数据库表发生更新、插入或删除操作时通知 Integration Server。

例如，适配器服务使交易伙伴可以查询您的库存数据库，以确定某种特定物品当前是否有现货。并且，在对库存数据库表执行更新时，适配器通知会通知 Integration Server。

查看调查器中与适配器相关的度量标准并进行导航:

1. 展开代理节点，然后展开“WebMethods”，可显示适配器度量标准类别。
2. 展开“WebMethods”>“适配器连接池”，然后展开具体的适配器名称，可查看该适配器的连接信息。
3. 展开“WebMethods”>“适配器通知”，然后展开具体的适配器名称，可查看该适配器的通知信息。如果没有为适配器配置任何通知，则不会显示此度量标准类别。
4. 展开“WebMethods”>“适配器服务”，然后展开具体的适配器类型，可查看活动连接的列表。
5. 展开活动的适配器连接，可查看 Integration Server 上运行的适配器服务的列表。
6. 展开单个服务可查看该服务的度量标准。

适配器连接池的度量标准

以下度量标准适用于各个适配器名称的“WebMethods”>“适配器连接池”子节点下配置的适配器：

可用连接

可用于适配器的空闲连接数。

当前活动连接数减去繁忙连接数，便可得到可用连接数。随着服务器多次尝试连接到资源，可用连接数会越来越少。

当前大小

当前活动连接总数。

最大大小

选定适配器所允许的最大连接数。

最小大小

为所选适配器配置的最小连接数。

适配器通知的度量标准

如果配置了适配器通知，则轮询进程或侦听程序进程会监控外部资源中的更改（例如对数据库表的插入、更新或删除操作），以便相应流服务或 Java 服务可以对更改做出响应。

例如，如果您部署了适配器服务从而使交易伙伴能够查询库存数据库，您可以配置适配器通知，以便在每次对数据库表执行更新时都通知 Integration Server。要处理与通知关联的文档（例如，发送基于适配器通知的发票），可以配置 Integration Server 触发器。

对于各个适配器服务名称，所有标准 CA Introscope® 度量标准均可用于“WebMethods”>“适配器通知”子节点下的适配器通知。

适配器服务的度量标准

对于各个适配器连接服务名称，所有标准 CA Introscope® 度量标准均可用于“WebMethods”>“适配器服务”子节点下的适配器服务。

“每个时间间隔的错误”度量标准仅包括执行适配器服务时发生的错误，而不包括执行服务之前发生的访问异常或任何其他类型的错误。

授权的度量标准

通常，webMethods Integration Server 包括多个具有特定访问控制列表(ACL)的用户组，访问控制列表用于定义授予不同用户的权限。例如，指定给组的默认权限包括 Administrator、Developer、Monitor User 和 Replicator 等。您也可以根据需要创建自己的用户组和权限。如果用户尝试执行无权访问的 Integration Server 服务，系统会拒绝其访问并记录一个错误。

通过选择“WebMethods”>“授权”节点下的“每个时间间隔的错误”度量标准，可查看有关授权失败的信息。

因为拒绝用户访问时没有调用服务，所以未将错误记录在服务的“每个时间间隔的错误”度量标准中。但是，也可以通过单击调查器中的“错误”选项卡来看“访问异常”错误。在列表中选择错误，可在“错误快照”中显示有关该错误的详细信息。

业务流程的度量标准

业务流程包括一系列用于完成业务事件的步骤，例如下订单或添加新员工。

只有下列标准 CA Introscope® 度量标准可用于“WebMethods”>“业务流程”>“*<business_process_name>*”节点下的 webMethods 业务流程和业务流程步骤：

- 平均响应时间(毫秒)
- 每个时间间隔的错误
- 每个时间间隔的响应数

在业务流程级别，这些度量标准会跟踪成功完成流程所用的时间、成功完成的流程数以及生成错误和失败的流程数。对于分布式流程，仅针对完成了业务流程的代理显示流程级别上的“平均响应时间”和“每个时间间隔的响应”。

除标准度量标准之外，还为业务流程提供了以下度量标准：

每个时间间隔的取消

在时间间隔中取消的流程数量。

每个时间间隔的挂起

时间间隔内挂起的流程数。

每个时间间隔的重新启动

时间间隔内重新启动的流程数。

每个时间间隔的继续

时间间隔内继续的流程数。

您也可以收集 webMethods Integration Server 业务流程的依赖关系和偏差度量标准。有关标准依赖关系和偏差度量标准的信息，请参阅[使用调查器查看 SOA 性能度量标准 \(p. 49\)](#)。

查看业务流程的步骤级度量标准

默认情况下，在监控 webMethods Integration Server 时，所有标准 CA Introscope® 度量标准也可用于业务流程步骤，如下所示：

WebMethods > BusinessProcesses > *<business_process_name>* > *<step_identifier>* 节点

适用于 webMethods Integration Server 6.5.2 到 6.5.3： 步骤级别的度量标准仅针对单级别步骤显示。默认情况下，不会汇总其他步骤级别的度量标准（例如，在内部作为多个步骤执行的人工任务步骤的度量标准，以及引用流程、子流程或流服务步骤的度量标准等）。

要在监控 webMethods Integration Server 6.5.2 到 6.5.3 版时获取业务流程的完整步骤级别的度量标准，请在 webmethods-toggles.pbd 文件中取消注释以下行： -

```
TurnOn: BProc<version_number>FlowStepMarkTracing  
TurnOn: BProc<version_number>StepFlowTracing
```

注意：有关 webMethods Integration Server 要求，请参阅《兼容性指南》。

遵循这些步骤:

1. 展开代理节点，然后展开“WebMethods”>“业务流程”。

将显示您已经在 webMethods Designer 中定义并在环境中作为软件包部署的业务流程。

为区分正在运行的业务流程的不同版本，在业务流程名称后面附加了下划线（_）和版本号。

2. 展开任一流程名称。

此时将显示为该流程定义的步骤。

3. 展开任一步骤标识符 (StepID)。

将显示在 Integration Server 上表示步骤的步骤级别度量标准和流服务度量标准。

关于已完成流程的“平均响应时间”

webMethods 业务流程在执行期间，可能会被 My WebMethods Server 取消或挂起。在许多情况下，还可以手动从 My WebMethods Server 继续执行使用 My WebMethods Server 取消或挂起的流程。因为这些流程可以从 My WebMethods Server 重新提交并运行直至完成，所以，流程级别的“平均响应时间”度量标准仅表示成功完成的流程。

如果流程失败或挂起，然后在 My WebMethods Server 中重新提交，则“平均响应时间”度量标准会反映从初次调用流程到重新提交后成功完成的时间。如果流程失败但未使用 My WebMethods Server 重新提交，则该流程不会包括在“平均响应时间”度量标准中。但是，如果将流程作为其流程流的一部分取消（例如，使用“终止”步骤），则该流程会包括在“平均响应时间”度量标准中。

如果业务流程中涉及多个 Integration Server，则仅针对完成了业务流程的 Integration Server 报告“平均响应时间”度量标准。

关于已完成流程的“每个时间间隔的响应”

“每个时间间隔的响应”度量标准反映成功完成的流程数。如果业务流程中涉及多个 Integration Server，则仅报告其中的业务流程已完成的 Integration Server 的“每个时间间隔的响应”度量标准。

关于失败流程的“每个时间间隔的错误”

“每个时间间隔的错误”度量标准仅在流程无法完成运行时显示。

关于流程中步骤的并发调用

尽管某些类型的步骤（例如“子流程”、“引用流程”和“人工任务”步骤）在 **webMethods Designer** 中显示为单个步骤，但在内部是作为多个步骤来执行。例如，“人工任务”步骤在内部作为两个步骤执行，即 *PRE_StepID* 和 *POST_StepID*。“并发调用”度量标准会将这些内部步骤计为单独的调用，而不是汇总为一个步骤。

流服务的度量标准

流服务是以 **webMethods** 流语言编写的服务。一个流服务包括一系列流步骤，每个步骤都具有明确定义的输入和输出。每个流步骤是一个基本的工作单元，**webMethods** 在运行时会解释和执行这些工作单元。特定服务中的所有流步骤是使用同一线程执行的，从而使该流中一个步骤的输出可用作下一个步骤的输入。

所有标准 **CA Introscope®** 度量标准均可用于“WebMethods”>“流服务”节点下的各个 **webMethods** 流服务。将收集各个流步骤的数据并将其汇总到一个或多个流服务的度量标准中，从而监控 **Integration Server** 上所有流服务的总体运行状况。

显示的节点名称是已选择要监控且未使用筛选器排除的服务的完全限定名称。有关筛选希望监控的流服务的信息，请参阅[筛选监控和显示的服务](#) (p. 246)。

调用其他流服务的流服务

在大多数情况下，所有标准 **CA Introscope®** 度量标准及其相应的聚合度量标准值应用于 **webMethods** 流服务的方式与其应用于其他应用程序组件的方式相同。

如果一个流服务调用另一个流服务，则这两个流服务的度量标准都在调查器树中的“WebMethods”>“流服务”节点下显示为独立节点。它们不会相互嵌套（即一个节点不会位于另一个节点下）。

要以调用顺序查看在同一 **webMethods Integration Server** 上执行的所有流步骤，可以启动事务跟踪会话。

Java 服务的度量标准

Java 服务可以是任何用户定义的服务或内部的内置 **webMethods** 服务，这些服务可以使用 **Java** 语言实现或以其他语言编写，使用 **Java** 类包装，还可以显示为 **webMethods Integration Server** 上的服务。例如，**webMethods Integration Server** 提供的所有内置服务均为 **Java** 服务。

所有标准 CA Introscope® 度量标准都可用于“WebMethods”>“Java 服务”节点下已部署的各个 webMethods Java 服务。

在 webMethods 中，位于同一文件夹中的 Java 服务是同一个类的方法。例如，完全限定名称为 *recording.user.accounts:createAccount* 的 Java 服务由 Java 软件包 (*recording.user*)、Java 类 (*accounts*) 和 Java 方法 (*createAccount*) 组成。调查器中显示的节点名称是已选择要监控且没有使用筛选器排除的服务的完全限定名称。

有关筛选需监控的 Java 服务的信息，请参阅[筛选监控和显示的服务 \(p. 246\)](#)。

JDBC 池的度量标准

webMethods Integration Server 通过 JDBC 连接点连接至数据库，来收集和存储关于用户、文档、内部服务器函数、审核和错误日志的信息以及其他信息。JDBC 池中允许的最大 JDBC 连接数可以通过 Integration Server 管理员进行配置。

以下度量标准可用于监控“WebMethods”>“JDBC 池”子节点下各个线程池名称的 JDBC 使用情况：

可用连接

JDBC 池的可用连接数。

当前活动连接数减去繁忙连接数，便可得到可用连接数。

当前大小

当前处于活动状态的 JDBC 连接的总数。

最大大小

为所选池配置的最大 JDBC 连接数。

最小大小

为选定池配置的最小 JDBC 连接数。

线程池的度量标准

WebMethods Integration Server 使用线程执行服务、从 WebMethods Broker 中检索文档以及执行触发器。服务器启动时，线程池最初包含的是最小线程数。服务器将根据需要向该池添加线程，直到达到所允许的最大值。如果使用的线程数量已达到最大值，则服务器在进程完成之前会一直等待，然后在开始更多进程之前将线程返回到池中。

各个池名称的 WebMethods > “线程池”子节点下提供了以下用于监控线程使用情况的度量标准：

最大大小

为所选线程池配置的最大线程数。

最小大小

为选定线程池配置的最小线程数。

使用的线程

所选线程池当前使用的线程数。

Trading Networks 的度量标准

利用 Trading Networks，组织能够交换文档，以建立并丰富企业之间的关系。例如，可以将购买者、供应商、战略合作伙伴或其他组织视为可与其交换文档的交易合作伙伴。通过交换文档，可简化跨组织边界的业务流程。通过 webMethods Integration Server，Trading Networks 可充当交易伙伴之间的网关。

下列标准度量标准可用于 “WebMethods” > “Trading Networks” 节点下的 webMethods Trading Networks：

- 每个时间间隔的错误

您还可以使用下列度量标准类别（位于 “WebMethods” > “Trading Networks” 节点下）监控 XML 文档识别的 Trading Networks 以及处理规则：

文档类型

文档类型确定了要在不同合作伙伴关系中交换的数据的结构和类型。

处理规则

处理规则介绍了如何通过 Integration Server 将文档路由到其目的地。

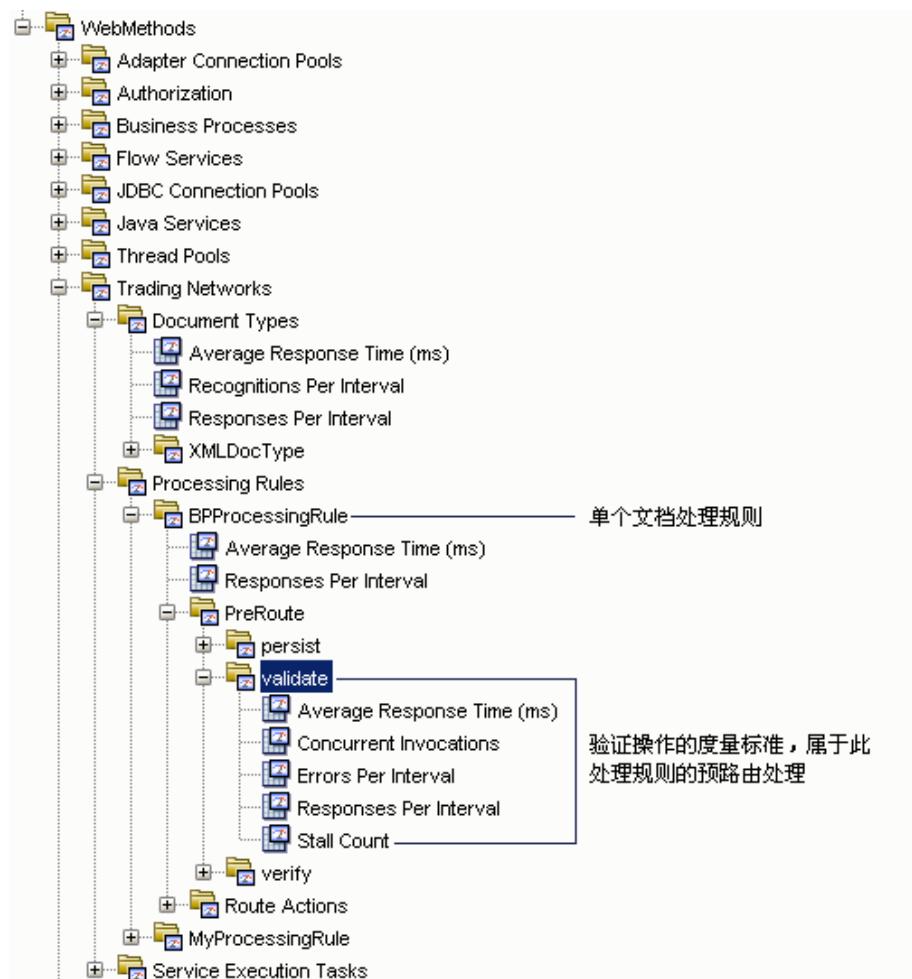
服务执行任务

当处理规则以异步方式执行具有重试限制的服务时，会创建服务执行任务。

服务线程

创建服务线程以处理未受重试限制而路由的文档的异步处理。

例如，可以展开“WebMethods”>“Trading Networks”>“文档类型”子节点，以查看特定文档类型的度量标准，包括单个文档类型的文档识别和接受度量标准。同样，可以展开“WebMethods”>“Trading Networks”>“处理规则”子节点，以查看特定处理规则的度量标准，包括各个预路由和路由操作的度量标准。-



文档类型的度量标准

只有下列标准 CA Introscope® 度量标准可用于“WebMethods”>“Trading Networks”>“文档类型”节点下的 webMethods Trading Networks:

- 平均响应时间 (毫秒)
- 每个时间间隔的响应数

“文档类型”的“平均响应时间”和“每个时间间隔的响应”度量标准在处理文档时会聚合。仅当执行处理规则时才报告这些度量标准。如果提交文档但该文档未经任何处理规则处理，则不会报告“平均响应时间”和“每个时间间隔的响应”度量标准。

除标准度量标准之外，下列度量标准在“WebMethods”>“TradingNetworks”>“文档类型”节点下也可用（如果提交的文档有效）：

每个时间间隔的识别

在 15-秒时间间隔结束时识别为有效交易合作伙伴文档的文档数。

对于识别的各个文档类型，所有标准 CA Introscope® 度量标准在“WebMethods”>“Trading Networks”>“文档类型”>“*<document_type_name>*”节点下均可用。

处理规则的度量标准

对于各个处理规则，只有下列标准 CA Introscope® 度量标准可用于“WebMethods”>“Trading Networks”>“处理规则”>“*<processing_rule_name>*”子节点下的 webMethods Trading Networks:

- 平均响应时间 (毫秒)
- 每个时间间隔的响应数

对于处理规则中的特定操作，所有标准 CA Introscope® 度量标准均可用于“WebMethods”>“Trading Networks”>“处理规则”>“*<processing_rule_name>*”>“PreRoute”或“路由操作”子节点下的 webMethods 各个处理规则操作。

关于处理规则的平均响应时间

处理规则的“平均响应时间”度量标准聚合以同步方式路由文档所花费的平均响应时间。对于处理规则中的同步调用，该度量标准与聚合的“文档类型平均响应时间”相同。

服务执行任务的度量标准

下列度量标准可用于“WebMethods”>“Trading Networks”>“服务执行任务”子节点下的 webMethods Trading Networks:

- 每个时间间隔的新增任务
- 每个时间间隔的任务故障
- 每个时间间隔的已完成任务

此外，下列标准 CA Introscope® 度量标准也可用于“WebMethods”>“Trading Networks”>“服务执行任务”>“调用”子节点下的各个调用操作： -

- 平均响应时间 (毫秒)
- 每个时间间隔的响应数

服务线程的度量标准

下列标准 CA Introscope® 度量标准可用于“WebMethods”>“Trading Networks”>“服务线程”子节点下的 webMethods Trading Networks： -

- 平均响应时间 (毫秒)
- 每个时间间隔的响应数

触发器的度量标准

您可以将 Integration Server 触发器配置为使用 webMethods Broker 或 Java Message Service (JMS) 来处理文档。webMethods Broker 触发器是订阅和处理本地发布或传送给 webMethods Broker 的文档的触发器。JMS 触发器是一种可从 JMS 提供商的某个目标（如队列或主题）接收消息，然后对这些消息进行处理的触发器。

对于各个触发器名称，所有标准 CA Introscope® 度量标准都可用于“WebMethods”>“触发器”子节点下的触发器。如果您尚未配置任何 Broker、本地或 JMS 触发器，则不显示该度量标准类别。

Web 服务的度量标准

Web 服务是一些打包成单元并发布到网络中供用户或软件程序使用的构造块。

- 适用于受支持的 webMethods Integration Server 版本，Web 服务连接器定义了 Integration Server 是充当 Web 服务使用方（客户端）还是充当 Web 服务提供方（服务器）。例如，可以暴露您在外部部署充当 Web 服务的任何流服务或基于 Java 的服务。可使用提供方 Web 服务描述符将服务信息发布到 UDDI 注册表。webMethods Integration Server 还可以通过使用方 Web 服务描述符调用外部应用程序服务器上的 Web 服务，以便作为客户端请求服务。
- 适用于 webMethods Integration Server 6.5.2 到 6.5.3，可从 WSDL 文档生成 Web 服务连接器，以便调用远程 Web 服务。Web 服务连接器是一种具有输入和输出签名的流服务。该签名与来自创建此流服务的 WSDL 文档中的输入和输出消息相对应。通过 webMethods Integration Server，您还可以通过 Integration Server 和 Developer 将 Integration Server 软件包中的任何现有服务转化为 Web 服务。

注意：有关 webMethods Integration Server 要求，请参阅《CA APM 兼容性指南》。

所有标准 CA Introscope® 度量标准都可用于 webMethods Integration Server 上的客户端和服务器 Web 服务及操作。此外，“每个时间间隔的 SOAP 错误”是在 SOA 平台中监控 Web 服务的所有扩展功能的标准度量标准。

详细信息：

[在“错误”选项卡上查看 SOAP 故障和错误度量标准 \(p. 57\)](#)

关于服务器端度量标准

作为 Web 服务提供方，webMethods 具有多个 SOAP 处理器，可用来处理 Web 服务请求：

- 默认 Web 服务 SOAP 处理器
- 用于接收和处理 SOAP 远程过程调用的 SOAP RPC 处理器
- 用于在没有定义或忽略进程指令时处理消息的默认 SOAP 消息处理程序

服务器端 Web 服务度量标准包括有关所有这三种 SOAP 处理器的信息。但自定义 SOAP 处理器不包括在这些度量标准中。

关于客户端度量标准

客户端度量标准表示执行外部应用程序服务器上运行的 Web 服务请求。

当 webMethods Integration Server 充当 Web 服务使用方时，它会自动为每个操作生成 Web 服务连接器。随后客户端会通过该连接器直接绑定到 Web 服务的端点。当执行 Web 服务连接器时，调用 Web 服务的请求会直接转至 Web 服务实施。在内部，Web 服务连接器为流服务，可使用“流服务”度量标准进行监控。

"WebServices" > “客户端”度量标准表示执行调用外部 Web 服务（而非客户端直接请求的操作）的连接器。

服务器端度量标准表示执行 webMethods Integration Server 上运行的 Web 服务。

XSLT 服务的度量标准

可扩展样式表语言 (XSL) 和 XSL 转换 (XSLT) 提供基于 XML 的语言，用于将源 XML 文档转换为其他文档。例如，原始 XML 文档可用于创建新的 XML 文档，转化为 HTML 以便显示为网页，以及以纯文本发布。

在调用 XSLT 服务时，Integration Server 会检索相关文档（样式表）中的指令，然后应用这些指令以按照样式表定义的格式将源 XML 文档转换为新的文档。

所有标准 CA Introscope® 度量标准都可用于监控
“WebMethods” > “XSLT 服务”节点下的各个 webMethods XSLT 服务。

查看默认的 webMethods 度量标准组

SOA Extension for webMethods Integration Server 包括定义默认显示板和报警用的默认度量标准组。您还可以在自定义显示板和报警中使用这些默认度量标准组。

在 webMethods Integration Server 的企业管理器扩展中，默认度量组会作为 webMethods Integration Server 管理模块 (*WebMethodsISManagementModule.jar*) 的一部分进行打包。

查看 webMethods 代理的默认度量标准组

1. 在调查器中，单击“Workstation” > “新建管理模块编辑器”。
2. 展开“*超级域*” > “管理模块” > “WebMethods IS (*超级域*)”。

3. 展开“度量标准组”节点，查看为 webMethods 管理模块定义的所有度量标准组。

4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

您可以修改任意度量标准组的默认设置或自行创建自定义度量标准组。

注意：有关创建或修改度量标准组的详细信息，请参阅《CA APM Workstation 用户指南》。

查看默认 webMethods 报警

SOA Extension for webMethods 包括预配置的显示板中使用的默认报警定义。您也可以在自定义的显示板中使用这些默认报警。大部分默认报警预配置有默认的“警告”和“危险”阈值，如果超过阈值或重要级别增加，则会向控制台发送通知。

默认报警定义作为 webMethods Integration Server 管理模块 (*WebMethodsISManagementModule.jar*) 的一部分，打包在 webMethods Integration Server 的企业管理器扩展中。

查看 webMethods 代理的默认报警定义：

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 展开“*超级域*”>“管理模块”>“WebMethods IS (*超级域*)”。
3. 展开“报警”节点，查看为 webMethods Integration Server 管理模块定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

特别是，您应检查默认“警告”和“危险”阈值以及针对严重报警的预定义操作，并对其进行调整使其适用于您的环境。例如，您可能希望调整阈值（如有需要）、添加通知或定义正确的操作。

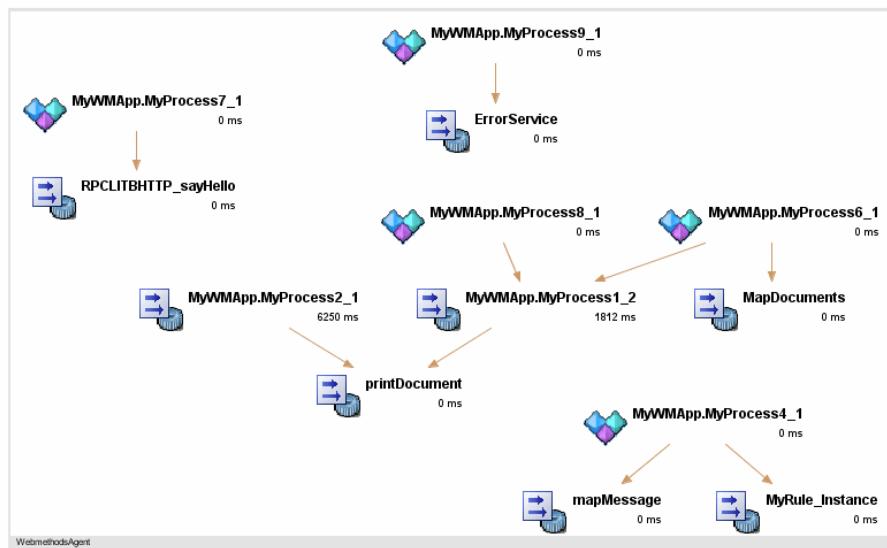
您可以修改任何报警的任意默认设置或创建自己的自定义报警。

注意：有关创建或修改报警的详细信息，请参阅《CA APM Workstation 用户指南》。

查看 webMethods 依赖关系

通过在“调查器”树中选择一个相应 webMethods 节点并单击“SOA 依赖关系地图”选项卡，可查看 webMethods 流服务、java 服务、适配器服务、Web 服务以及 webMethods 业务流程的依赖关系。

您选择的节点决定了依赖关系地图中显示的上下文。然后您可以将您查看的上下文和详细信息级别卷起以进行折叠或放开以进行展开。例如，对于所有业务流程的顶级依赖关系视图，您可以在调查器中选择“业务流程”节点，然后单击“SOA 依赖关系地图”选项卡：



要查看某个特定业务流程的高级依赖关系视图，您可以在调查器中选择相应业务流程名称，并继续将依赖关系级别添加到该图中，以便查看该业务流程的整个工作流程或根据需要放大图中的特定节点。有关浏览依赖关系地图的更多详细信息，请参阅[使用 SOA 依赖关系地图](#) (p. 63)。

webMethods 的跟踪事务

事务跟踪提供完成业务事务所涉及的特定步骤的详细或摘要视图。对于 webMethods Integration Server 业务流程或应用程序服务，您可以跟踪包括通过下列协议路由的操作的事务：

- 简单对象访问协议 (SOAP)
- 超文本传输协议 (HTTP)
- 安全的超文本传输协议 (HTTPS)
- Java Message Service (JMS)
- webMethods Broker 消息服务

涉及 webMethods 服务的事务可包括同步和异步调用，以及使用不同线程以并行方式运行的活动。要对跨多个线程或流程的事务启用事务跟踪，将在每个组件和操作中插入关联标识符并将其用作事务步骤。然后可以查看所执行的特定操作的相关详细信息以及完成每个操作所花费的时间。

只要在每个被跟踪的节点上启用 CA APM for SOA 和 CA APM for webMethods Integration Server，您还可以跨任何平台组合跟踪业务事务。这样您就能够查看关于事务的详细信息，即使该事务跨多个 JVM 或 CLR 也是如此。

关于配置跨进程事务跟踪

通常，通过将关联信息从一个进程传递到另一个进程，可配置跨进程事务跟踪功能。使用 SOAP 或 HTTP 头传递信息。适用于 webMethods Integration Server 6.5.2 到 6.5.3 版，当使用 RPC 样式 Web 服务客户端（连接器）时，不能使用 SOAP 头传递关联标识符。

要查看涉及 webMethods Integration Server RPC Web 服务连接器的业务事务的事务跟踪，请[配置代理关联跟踪 \(p. 101\)](#)以使用 HTTP 头传递关联标识符。

了解跨进程事务跟踪的值

跨进程事务跟踪提供有关面向服务的体系结构中的松散耦合服务所执行操作的有价值信息。您可以使用跨进程事务跟踪来确定：

- 如何通过流或 Java 服务路由业务流程。
- 在事务期间调用和执行哪些服务。
- 事务处理期间所做调用的顺序。
- 处理请求或回复最慢的位置。

启动并查看示例事务跟踪

您可以使用以下方式启动事务跟踪会话：

- 直接通过 SOA 依赖关系视图中的视图节点。
- 通过单击“Workstation”>“新建事务跟踪会话”，从 Workstation 手工启动。

如果从依赖关系视图启动事务跟踪，则视图节点类型自动设置默认筛选。如果手工启动新的事务跟踪会话，您可以选择以下 webMethods 筛选类型之一：

- 业务流程
- 命名空间
- 操作名称

配置筛选并启动事务跟踪会话后，将显示事务跟踪查看器。您可以选择跟踪来查看有关在事务中执行的调用的其他详细信息。这些详细信息包括在 webMethods Integration Server 上执行的所有触发器、流服务操作或业务流程步骤。

对 webMethods 流程使用顺序视图

与 webMethods Integration Server 相关的事务通常包括异步调用，因此您可能会发现，单击“顺序视图”即可查看在事务处理过程中异步执行的进程的事务工作流。“顺序视图”显示流程执行的顺序（以该顺序可确定为限度）。对于 webMethods Integration Server 事务来说，顺序不一定表示传统的调用方-被调用方关联关系，而是表示某一进程在何时会触发其他进程的执行。

但是，应当注意是 webMethods Integration Server 流程的处理时间是通过从流程启动到完成的完整持续时间计算得到的，其中包括与其调用的流程相关的处理时间。webMethods Integration Server 流程不支持单纯持续时间，即从调用流程的持续时间中减去非分块同步和异步流程的处理时间。

有关事务跟踪的详细信息，请参阅[在 SOA 环境中使用事务跟踪 \(p. 85\)](#)。

注意：有关配置跟踪的信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。有关使用事务跟踪视图和历史数据的详细信息，请参阅《CA APM Workstation 用户指南》。

第 12 章：监控 WebSphere Process Server 和 WESB

WebSphere Process Server (WPS) 是一个综合的面向服务的体系结构 (SOA) 集成平台，它提供消息和数据管理服务以帮助您使用服务组件体系结构 (SCA) 集成现有的应用程序，并开发和部署新应用程序。通过 SOA Extension for WebSphere Process Server (WPS)，您可以监控 WebSphere Process Server 的关键组件，如 WebSphere Enterprise Service Bus (WESB)、业务服务组件和辅助服务。

本节介绍了特定于 WPS 和 WESB 的显示板、度量标准以及报警，可使用它们来监控和分析 WebSphere Process Server 或 WebSphere Enterprise Service Bus 环境的性能、可用性和总体运行状况。

此部分包含以下主题：

- [关于 WebSphere Process Server 和 WESB \(p. 267\)](#)
- [如何启用对 WebSphere Process Server 或 WESB 的监控 \(p. 271\)](#)
- [使用显示板监控 WPS 或 WESB \(p. 278\)](#)
- [查看 WPS/WESB 的度量标准并进行导航 \(p. 282\)](#)
- [查看 WPS 和 WESB 默认度量标准组 \(p. 289\)](#)
- [查看 WPS 和 WESB 默认报警 \(p. 290\)](#)
- [查看 WPS 或 WESB 依赖关系 \(p. 291\)](#)
- [跟踪 WPS 或 WESB 的事务 \(p. 292\)](#)

关于 WebSphere Process Server 和 WESB

WebSphere Process Server (WPS) 是一个综合的面向服务的体系结构 (SOA) 集成平台，可用于集成新旧应用程序，并部署松散耦合服务以实现业务目标。

WebSphere Process Server 体系结构包括：

- 服务组件
- 支持服务
- WebSphere Enterprise Service Bus 消息传递

WebSphere Process Server 的服务组件包括业务流程、业务状态机、业务规则以及人工任务。WebSphere Process Server 的支持服务包括中介流、接口映射、业务对象映射、Java 组件、服务集成总线 (SIB) 通信、业务流程、业务状态机、关系、选择器以及适配器。WebSphere Enterprise Service Bus (WESB) 管理通过 WebSphere Process Server 的消息流，并处理服务和客户端之间所需的任何数据转换或路由。

通过 SOA Extension for WebSphere Process Server，可以监控包括 WebSphere Enterprise Service Bus 在内的完整 WebSphere Process Server 体系结构。或者，可以仅监控 WebSphere Enterprise Service Bus。可以将 WebSphere Enterprise Service Bus 作为独立的产品进行安装。

如果监控 WebSphere Process Server，显示板和度量标准将显示 WPS 和 WESB 的相关信息。如果将 WESB 作为独立产品进行监控，显示板和度量标准将仅包含 WESB 的相关信息。

监控 WebSphere Process Server 组件

如果使用 WebSphere Process Server，则可以通过以下内容的度量标准来监控 WebSphere Process Server 环境的操作：

业务对象映射

业务对象映射基于源业务对象服务组件中的值将值分配给目标业务对象服务组件。一个业务对象成为源，另一个成为目标。业务对象映射可对源和目标进行映射。

通过 SOA Extension for WPS，您可以在“WProcServer”>“BO 映射”节点下监控业务对象映射的性能和总体运行状况。

业务流程

业务流程由按特定顺序执行以达到特定目标的单个任务组成。业务流程至少由一个接收消息以响应请求所涉及到的任务组成。

通过 SOA Extension for WPS，您可以在“WProcServer”>“业务流程”节点下监控业务流程中活动的性能和总体运行状况。

业务状态机

业务状态机是一个事件驱动的应用程序。应用程序将等待事件发生，并根据事件选择要执行的适当操作。执行操作后，状态机会进入其他状态或者仅等待其他事件发生。

通过 SOA Extension for WPS，您可以在“WProcServer”>“业务状态机”节点下监控事件驱动的流程的性能和总体运行状况。

业务规则

业务规则捕获并实施业务策略和实践。业务规则可强制实施业务策略、做出决定或从现有数据推断新数据。

通过 SOA Extension for WPS，您可以在“WProcServer”>“业务规则”节点下监控已定义业务规则的性能和总体运行状况。

数据绑定

数据绑定用于在入站和出站处理期间将业务对象转换为数据流。

通过 SOA Extension for WPS，您可以在“WProcServer”>“数据绑定”节点下监控数据绑定的性能和总体运行状况。

人工任务

人工任务表示由人员执行但在某种方式下也会与 WPS 中的流程或服务进行交互的任务。

通过 SOA Extension for WPS，您可以在“WProcServer”>“人工任务”节点下监控需要人工交互的任务的性能和总体运行状况。

接口映射

接口映射使用转换和其他基本操作协调具有不同接口的组件间的差异，从而使具有不同接口的组件可以进行通讯。

通过 SOA Extension for WPS，您可以监控“WProcServer”>“接口映射”节点下接口映射的性能和总体运行状况。

J2CA（适配器）

适配器是使 WebSphere Process Server 可以与外部企业信息系统 (EIS) 进行通讯的应用程序。当 WebSphere Process Server 调用特定于 EIS 的操作时，会发生出站通讯。当应用程序侦听特定 EIS 事件时，会发生入站通讯。

通过 SOA Extension for WPS，您可以在“WProcServer”>“J2CA”节点下监控适配器的性能和总体运行状况。

Java 组件

通过 Java 组件，您可以在 WPS 或 WESB 中自定义 Java 实施。

通过 SOA Extension for WPS，您可以监控“WProcServer”>“Java 组件”节点下 Java 组件的性能和总体运行状况。

中介流

中介流包括处理请求的请求流、处理响应的响应流、处理事件的事件流以及处理错误的错误流。针对每个源操作开始请求流，针对每个目标操作开始响应流。

通过 SOA Extension for WPS，您可以在“WProcServer”>“WESB”>“中介流”节点下监控中介流的性能和总体运行状况。

中介元素

中介元素是实施中介流组件的功能的构造块。每个中介流包含使您可以转换数据的中介元素。

通过 SOA Extension for WPS，您可以监控“WProcServer”>“WESB”>“中介元素”节点下中介元素的性能和总体运行状况。

关系

关联关系在两种或两种以上数据类型的数据之间建立关联。例如，可使用关联关系定义客户和客户 ID 之间的关联。

通过 SOA Extension for WPS，您可以监控“WProcServer”>“关系”节点下关联关系定义的性能和总体运行状况。

选择器

运行时，使用选择器可以对服务组件进行灵活的处理。选择器接受来自客户端应用程序的一个调用，并允许基于选择条件在运行时调用不同目标。

通过 SOA Extension for WPS，您可以监控“WProcServer”>“选择器”节点下已定义的选择器的性能和总体运行状况。

SIB 通信

显示来自 Process Server Controller 组件的服务集成总线 (SIB) 度量标准，它用于控制业务流程和业务状态机的执行。可以在“WProcServer”>“SIB 通信”节点下监控这些度量标准。

监控 WebSphere Enterprise Service Bus（独立）

如果使用 WebSphere Enterprise Service Bus（作为独立产品），则可以通过以下内容的度量标准来监控 WebSphere Enterprise Service Bus 环境的操作：

业务对象映射

对于独立的 WESB 环境，您可以在“WESB”>“BO 映射”节点下监控业务对象映射的性能和总体运行状况。

数据绑定

对于独立的 WESB 环境，您可以在“WESB”>“数据绑定”节点下监控数据绑定的性能和总体运行状况。

J2CA（适配器）

对于独立的 WESB 环境，您可以在“WESB”>“J2CA”节点下监控适配器的性能和总体运行状况。

Java 组件

对于独立 WESB 环境，您可以监控 WESB > “Java 组件”节点下 Java 组件的性能和总体运行状况。

中介流

对于独立 WESB 环境，您可以监控 WESB > “中介流”节点下中介流的性能和总体运行状况。

中介元素

对于独立 WESB 环境，您可以监控 WESB > “中介元素”节点下中介元素的性能和总体运行状况。

关系

对于独立 WESB 环境，您可以监控 WESB > “关联关系”节点下关联关系定义的性能和总体运行状况。

如何启用对 WebSphere Process Server 或 WESB 的监控

可以通过执行以下步骤启用对 WebSphere Process Server 或 WebSphere Enterprise Service Bus 的监控：

1. 确认已安装了受支持的 WebSphere Process Server 或 WebSphere Enterprise Service Bus 版本。

注意：有关系统要求，请参阅《兼容性指南》。

2. 确认已安装并启用代理和 CA APM for SOA。

3. 通过[配置代理配置文件](#) (p. 285)，使代理能够使用 CA APM for WebSphere Process Server 或 WebSphere Enterprise Service Bus。应包含用于监控 WebSphere Process Server 或 WebSphere Enterprise Service Bus 的相应 ProbeBuilder 指令文件。
4. 为 WebSphere Process Server 或 WebSphere Enterprise Service Bus [启用企业管理器扩展](#) (p. 276)。将文件从 `<EM_Home>/examples/SOAExtensionForWPSandWESB` 目录移动到相应的企业管理器目录中。

启用代理以监控 WPS 或 WESB

可以在安装代理时选择“WebSphere”作为应用程序服务器来启用对 WebSphere Process Server 的监控（其中包括针对 WebSphere Process Server 的监控功能），也可在安装代理后手工启用该监控。

如果在安装代理时选择了 CA APM for WebSphere Process Server 和 WESB，则在 `<Agent_Home>` 目录中提供用于监控 WebSphere Process Server 和 WebSphere Enterprise Service Bus 的文件。

安装代理之后，必须通过配置代理配置文件手工启用 CA APM for WebSphere Process Server 或 CA APM for WebSphere Enterprise Service Bus。

手工启用 CA APM for WPS 或 CA APM for WESB:

1. 确认已安装并启用代理和 CA APM for SOA。
2. 确认 CA APM for WebSphere Process Server 目录 `SOAExtensionForWPSandWESB` 位于 `<Agent_Home>/examples` 目录中，然后将文件从 `<Agent_Home>/examples/SOAExtensionForWPSandWESB/ext` 目录复制到 `<Agent_Home>/ext` 目录。
3. 在文本编辑器中打开 `<代理主目录>/core/config/IntroscopeAgent.profile` 文件。

4. 将相应的指令文件添加到 *IntroscopeAgent.profile* 文件中的 *introscope.autoprobe.directivesFile* 属性中。
 - 添加 *wps.pbd* 以监控所有 WebSphere Process Server 组件，包括 WebSphere Enterprise Service Bus。
 - 添加 *wesb.pbd* 以仅监控独立 WESB 环境中的 WebSphere Enterprise Service Bus 组件。

如果要修改默认监控，可在 *wps.pbd* 或 *wesb.pbd* 文件中自定义 ProbeBuilder 指令。例如，您可以随后微调对特定组件的监控，只需打开或关闭对特定跟踪器组的跟踪即可。

5. 保存并关闭 *IntroscopeAgent.profile* 文件。

Business Process Manager 8.0 及 8.1 所需配置

对于 Business Process Manager 8.0 及 8.1，配置 *wps.pbd* 以启用跟踪。

遵循这些步骤：

1. 如果 Business Process Manager 正在运行，请停止运行。
2. 转到 <代理主目录>/core/config 目录并打开“*wps.pbd*”文件。

注意：如果您正在使用传统模式，使用同一步骤修改文件“*wps-legacy.pbd*”。

3. 从 *pbd* 文件中注释下列跟踪器映射和跟踪器指令语句：

```
#SetTracerClassMapping: ThreadCreationTracer
com.wily.powerpack.webspheredprocserver.tracer.hc2.ThreadCreationTracer
com.wily.introscope.probebuilder.validate.ResourceNameValidator

#TraceOneMethodWithParametersOfClass:
com.ibm.bpe.framework.navigation.Navigationworker dowork
ThreadCreationTracer "Navigation-work"
```

4. 从 *pbd* 文件中取消注释下列跟踪器映射和跟踪器指令：

```
SetTracerClassMapping: ThreadCreationTracer80
com.wily.powerpack.webspheredprocserver.tracer.hc2.ThreadCreationTracer80
com.wily.introscope.probebuilder.validate.ResourceNameValidator

TraceOneMethodWithParametersOfClass:
com.ibm.bpe.framework.navigation.Navigationworker dowork
ThreadCreationTracer80 "Navigation-work"
```

5. 保存并关闭 *wps.pbd*。

当您完成修改默认配置时，重新启动 Business Process Manager。

为业务流程和业务状态机的多个版本收集度量标准

您可以配置 `wps.pbd`, 以便为业务流程和业务状态机的不同版本提供单独的度量标准。如果您仅维护每个业务流程和业务状态机的一个版本，则可以跳过以下配置步骤。

遵循这些步骤:

1. 转到 `<Agent_Home>/core/config` 目录并打开 `wps.pbd` 文件。
2. 注释掉 `ProcessCustomTracer` 和 `ProcessFaultTracer` 参数。例如:

```
##SetTracerParameter: ProcessCustomTracer nameformatter  
com.wily.powerpack.webspheredprocserver.nameformatter.ProcessContextFormat  
ter  
#SetTracerParameter: ProcessFaultTracer nameformatter  
com.wily.powerpack.webspheredprocserver.nameformatter.ProcessContextFormat  
ter  
#SetTracerParameter: BPMMapTracer nameformatter  
com.wily.powerpack.webspheredprocserver.nameformatter.ProcessContextFormat  
ter  
#SetTracerParameter: BusinessProcessTracer nameformatter  
com.wily.powerpack.webspheredprocserver.nameformatter.BusinessProcessNameA  
ndTimeFormatter  
#SetTracerParameter: BusinessProcessFaultTracer nameformatter  
com.wily.powerpack.webspheredprocserver.nameformatter.BusinessProcessNameA  
ndTimeFormatter  
#SetTracerParameter: BPMMap7Tracer nameformatter  
com.wily.powerpack.webspheredprocserver.nameformatter.BusinessProcessNameA  
ndTimeFormatter
```

3. 取消注释 ProcessCustomTracer 和 ProcessFaultTracer 参数，以获取时间属性。例如：

```
SetTracerParameter: ProcessCustomTracer nameformatter
com.wily.powerpack.webspheredprocserver.nameformatter.ProcessAndTimeNameFo
rmatter
SetTracerParameter: ProcessFaultTracer nameformatter
com.wily.powerpack.webspheredprocserver.nameformatter.ProcessAndTimeNameFo
rmatter
SetTracerParameter: BPMMapTracer nameformatter
com.wily.powerpack.webspheredprocserver.nameformatter.ProcessAndTimeNameFo
rmatter
SetTracerParameter: BusinessProcessTracer nameformatter
com.wily.powerpack.webspheredprocserver.nameformatter.BusinessProcessNameF
ormatter
SetTracerParameter: BusinessProcessFaultTracer nameformatter
com.wily.powerpack.webspheredprocserver.nameformatter.BusinessProcessNameF
ormatter
SetTracerParameter: BPMMap7Tracer nameformatter
com.wily.powerpack.webspheredprocserver.nameformatter.BusinessProcessNameF
ormatter
```

注意：如果跳过此步骤，“业务流程”和“业务状态机”节点将附加开发文件时使用的时间格式。

4. 保存并关闭 wps.pbd。

完成修改默认配置之后，就可以重新启动 WebSphere Process Server 或 WESB 服务器。

收集性能监控基础架构 (PMI) 度量标准

如果在分布式环境中监控 WebSphere 应用程序服务器，则可以选择收集 WebSphere 性能监控基础架构 (PMI) 度量标准，并在调查器中查看 WebSphere Process Server 组件的这些度量标准。

为此，必须首先按照《*CA APM for IBM WebSphere Distributed Environments 指南*》中的说明配置监控。然后可以在应用程序服务器上修改其他代理属性，以启用对 WebSphere Process Server 组件的监控。

遵循这些步骤：

1. 打开 <Agent_Home>/core/config 中的 *IntroscopeAgent.profile* 文件。

2. 搜索以下属性：

```
introscope.agent.pmi.enable.WBISstats.RootGroup
introscope.agent.pmi.enable.SCASTats.RootGroup
```

3. 将这两个属性均设置为 true，以启用 PMI 度量标准报告。例如：

```
introscope.agent.pmi.enable.WBISstats.RootGroup=true
introscope.agent.pmi.enable.SCASTats.RootGroup=true
```

如果当前未定义这些属性，请手工将它们添加到 *IntroscopeAgent.profile* 中，然后将其设置为 true。

4. 保存并关闭 *IntroscopeAgent.profile* 文件。

如果已完成修改默认配置，则可以重新启动 WebSphere Process Server 或 WESB 服务器。

5. 使用 WebSphere 管理控制台启用 WebSphere Process Server 和 WebSphere Enterprise Service Bus 模块的性能监控基础架构。

注意：有关使用 WebSphere 管理控制台启用模块的信息，请参阅 IBM WebSphere 文档。

启用 WPS 或 WESB 的企业管理器扩展

安装企业管理器时，CA APM for WebSphere Process Server 和 WebSphere Enterprise Service Bus 文件默认安装在 *<EM_Home>/examples* 目录中。要启用 CA APM for WebSphere Process Server 或 WebSphere Enterprise Service Bus，请将企业管理器文件从 *<EM_Home>/examples* 目录复制或移动到企业管理器主目录中的适当位置。

注意：只有[在企业管理器上启用](#) (p. 35) CA APM for SOA 后，才可以使用 CA APM for WebSphere Process Server 或 WebSphere Enterprise Service Bus。

启用 WebSphere Process Server 的企业管理器扩展

1. 确认 CA APM for WebSphere Process Server 目录

SOAExtensionForWPSandWESB 位于 *<EM_Home>/examples* 目录中。将文件从 *<EM_Home>/examples/SOAExtensionForWPSandWESB* 目录复制到企业管理器目录结构中的对应位置。例如，将文件从 *<EM_Home>/examples/SOAExtensionForWPSandWESB/ext* 目录复制到 *<EM_Home>/ext* 目录。

2. 从

<EM_HOME>/examples/SOAExtensionForWPSandWESB/config/modules 目录中复制 *WPS_Management_Module.jar* 文件，并将其粘贴到 *<EM_Home>/config/modules* 目录中。

<EM_HOME>/examples/SOAExtensionForWPSandWESB/config/module s 目录包含 WebSphere Process Server 和 WebSphere Enterprise Service Bus（独立）的管理模块。如果监控 WebSphere Process Server（含 WebSphere Enterprise Service Bus），请仅复制 *WPS_Management_Module.jar* 文件。

如果群集环境中多个企业管理器，请仅将此文件复制到要用作 MOM 计算机的企业管理器上的 *<EM_Home>/config/modules* 目录中。其他所有文件和脚本应当同时安装在收集器企业管理器和 MOM 企业管理器中。

3. 如果要从 CA APM for WebSphere Process Server 的早期版本进行升级，请删除旧版本的企业管理器文件。

如果企业管理器上已安装某个早期版本，则必须先手工删除旧版本的企业管理器文件，然后才可以开始使用新版本的 CA APM for WebSphere Process Server。

如果从某个早期版本升级，请从企业管理器主目录中删除以下文件：

- `<EM_home>/config/modules/WPS_Management_ModuleV<version>.jar`
- `<EM_home>/product/enterprisemanager/plugins/com.wily.powerpack.webspherereprocserver.em.ext_<version>.jar`

例如，如果要从版本 8.1 升级，请删除以下文件：

- `WPS_Management_ModuleV8.1.0.0.jar`
- `com.wily.powerpack.webspherereprocserver.em.ext_8.1.0.jar`

4. 重新启动 Workstation。

将加载特定于 SOA Extension for WebSphere Process Server 或 WebSphere Enterprise Service Bus 的显示板和“概览”选项卡。

启用独立 WESB 的企业管理器扩展

1. 确认 CA APM for WebSphere Process Server 目录

`SOAExtensionForWPSandWESB` 位于 `<EM_Home>/examples` 目录中。将文件从 `<EM_Home>/examples/SOAExtensionForWPSandWESB` 目录复制到企业管理器目录结构中的对应位置。例如，将文件从 `<EM_Home>/examples/SOAExtensionForWPSandWESB/ext` 目录复制到 `<EM_Home>/ext` 目录。

2. 从

`<EM_HOME>/examples/SOAExtensionForWPSandWESB/config/modules` 目录中复制 `WESB_Management_Module.jar` 文件，并将其粘贴到 `<EM_Home>/config/modules` 目录中。

`<EM_HOME>/examples/SOAExtensionForWPSandWESB/config/module`s 目录包含 WebSphere Process Server 和 WebSphere Enterprise Service Bus（独立）的管理模块。如果监控 WebSphere Enterprise Service Bus（独立），请仅复制 `WESB_Management_Module.jar` 文件。

如果群集环境中多个企业管理器，请仅将此文件复制到要用作 MOM 计算机的企业管理器上的 `<EM_Home>/config/modules` 目录中。其他所有文件和脚本应当同时安装在收集器企业管理器和 MOM 企业管理器中。

3. 如果要从 CA APM for WebSphere Enterprise Service Bus 的早期版本进行升级，请删除旧版本的企业管理器文件。

如果企业管理器上已安装某个早期版本，则必须先手工删除旧版本的企业管理器文件，然后才可以开始使用新版本的 CA APM for WESB。如果从某个早期版本升级，请从企业管理器主目录中删除以下文件：

- `<EM_home>/config/modules/WESB_Management_ModuleV<version>.jar`
- `<EM_home>/product/enterprisemanager/plugins/com.wily.powerpack.webspheredprocserver.em.ext_<version>.jar`

例如，如果要从版本 8.1 升级，请删除以下文件：

- `WESB_Management_ModuleV8.1.0.0.jar`
- `com.wily.powerpack.webspheredprocserver.em.ext_8.1.0.jar`

4. 重新启动 Workstation。

将加载特定于 CA APM for WebSphere Process Server 或 WebSphere Enterprise Service Bus 的显示板和“概览”选项卡。

使用显示板监控 WPS 或 WESB

SOA Extension for WebSphere Process Server 包括多个预配置的显示板，您可以使用这些显示板监控应用程序环境的总体运行状况。显示板汇集已部署代理中的数据，以汇总性能信息并帮助您快速诊断和解决问题。

通常，将显示板用作监控环境的起点，因为使用显示板可以：

- 一目了然地监控 WebSphere Process Server 的关键组件的总体运行状况、性能、可用性和当前状态；
- 在低级别度量标准发出已超过警告或危险阈值的信号时，及早获取生产应用环境中存在潜在问题的通知；
- 深入查看性能信息，以分离和识别正遇到延迟或产生错误的 WebSphere 业务流程、业务规则、中介流或其他组件。

预配置的 WebSphere Process Server 和 WebSphere Enterprise Service Bus 显示板在 WebSphere 的企业管理器扩展中打包，作为 SOA Extension for WebSphere Process Server 管理模块 (`WPS_Management_Module.jar`) 或 SOA Extension for WebSphere Enterprise Service Bus 管理模块 (`WESB_Management_Module.jar`) 的一部分。

关于 WebSphere Process Server 显示板

WebSphere Process Server 管理模块为 WebSphere Process Server 提供下列预配置的显示板：

WPS - 概览

WebSphere Process Server 运行状况的顶级概览，包括 WebSphere Process Server 体系结构中所有主要组件的总响应时间、错误和停顿的报警指示器。

WPS—业务流程和状态机

所有业务流程和业务状态机的汇总状态，包括所有业务流程和状态机的“平均响应时间”图表、错误和停顿的报警指示器，以及最慢业务流程和状态机的列表。

WPS—业务规则和人工任务

所有业务规则和人工任务的汇总状态，包括所有业务规则和人工任务的“平均响应时间”图表、错误和停顿的报警指示器，以及最慢业务规则和人工任务的列表。

WPS—接口映射、BO 映射和关联关系

所有接口映射、业务对象映射和关联关系的汇总状态，包括接口映射、业务对象映射和关联关系的“平均响应时间”图表，错误和停顿的报警指示器，以及最慢接口映射、业务对象映射和关联关系的列表。

WPS—选择器和 Java 组件

所有选择器和 Java 组件的汇总状态，包括所有选择器和 Java 组件的“平均响应时间”图表、错误和停顿的报警指示器，以及最慢选择器和 Java 组件的列表。

WPS—数据绑定

所有数据绑定的汇总状态，包括“平均响应时间”、“每个时间间隔的错误”和“停顿计数”的图表和报警指示器，以及最慢数据绑定的列表。

WPS—适配器

所有入站和出站适配器的汇总状态，包括所有入站和出站适配器的“平均响应时间”图表、错误和停顿的报警指示器，以及最慢入站和出站适配器的列表。

WESB—中介流

中介流总体运行状况的汇总状态，以及请求流、响应流和错误流的单独图表和报警指示器。

由于中介流包括请求流、响应流和错误流，因此该显示板将显示中介流、请求流、响应流和错误流的“平均响应时间”图表，中介流、请求流、响应流和错误流的错误和停顿的报警指示器，以及最慢中介流的列表。

WESB—中介元素

中介元素总体运行状况的汇总状态，包括所有中介元素的响应时间和错误的报警指示器，以及请求流、响应流和错误流的响应时间、错误和停顿的单独报警指示器。

关于 WESB 显示板

如果监控 WebSphere Enterprise Service Bus（作为独立产品），则只有以下显示板可用：

WESB - 概览

WebSphere Enterprise Service Bus 运行状况的顶级概览，包括 WebSphere 体系结构的 WebSphere Enterprise Service Bus 层中所有主要组件的总响应时间、错误和停顿的报警指示器。

WESB—中介流

中介流总体运行状况的汇总状态，以及 WebSphere Enterprise Service Bus 的请求流、响应流和错误流的单独图表和报警指示器。

WESB—中介元素

概括了中介元素总体运行状况的状态，包括所有中介元素的响应时间和错误的报警指示器，以及 WebSphere Enterprise Service Bus 的请求流、响应流和错误流的响应时间、错误和停顿的单独报警指示器。

WESB—适配器

概括了所有入站和出站适配器的状态，包括所有入站和出站适配器的平均响应时间的图表，错误和停顿的报警指示器以及 WebSphere Enterprise Service Bus 的最慢入站和出站适配器的列表。

WESB—BO 映射和关联关系

所有业务对象映射和关联关系的汇总状态，包括所有业务对象映射和关联关系的“平均响应时间”图表、错误和停顿的报警指示器，以及 WebSphere Enterprise Service Bus 最慢业务对象映射和关联关系的列表。

WESB—数据绑定

概括了所有数据绑定的状态，包括平均响应时间、每个时间间隔的错误和停顿计数的图表和报警指示器，以及 WebSphere Enterprise Service Bus 的最慢数据绑定的列表。

WESB—Java 组件

所有 Java 组件的汇总状态，包括所有 Java 组件的“平均响应时间”、“每个时间间隔的错误”和“停顿计数”的图表和报警指示器，以及 WebSphere Enterprise Service Bus 最慢 Java 组件的列表。

WebSphere Enterprise Service Bus 的显示板类似于为 WebSphere Process Server 提供的显示板，但仅提供专用于 WebSphere Enterprise Service Bus 组件的报警指示器和度量标准。

查看 WebSphere Process Server 或 WESB 显示板

WebSphere Process Server 或 WebSphere Enterprise Service Bus 的显示板提供报警指示器和度量标准，以帮助您一目了然地评估系统运行状况。

遵循这些步骤：

1. 启动企业管理器（如果目前未运行）。
2. 启动 Workstation，并登录到已安装 SOA Extension for WebSphere Process Server 或 SOA Extension for WESB 的企业管理器。
3. 单击“Workstation”>“新建控制台”。
4. 从“显示板”下拉列表中选择某个 WPS 或 WESB 显示板。

例如，选择“**WPS—概览**”显示板以查看 WebSphere Process Server 运行状况的概览，包括所有服务组件和辅助服务的报警指示器。

在“WPS—概览”或“WESB—概览”显示板中，通过双击带有要查看的显示板名称的选项卡可导航至其他显示板。

5. 双击其他选项卡或显示板中的报警，打开相关显示板以查看更多详细信息。

例如，双击业务流程的响应时间报警，可在“**WPS—业务流程和状态机**”显示板上查看有关花费时间最多的特定业务流程的更详细信息。

“WPS—业务流程和状态机”显示板显示最慢业务流程和状态机的列表、“平均响应时间”图表，以及错误和停顿计数的报警指示器。

6. 在“WPS—业务流程和状态机”显示板中，可在“最慢的业务流程”列表中双击特定业务流程名称以打开调查器，其中选择了该业务流程的响应时间以进行进一步分析。

注意：有关启动和使用 Workstation、访问显示板或打开和导航调查器的信息，请参阅《CA APM Workstation 用户指南》。

查看 WPS/WESB 的度量标准并进行导航

在调查器树中导航时，您可以查看 WebSphere Process Server 或 WebSphere Enterprise Service Bus 基础架构大多数组件的标准 CA Introscope® 度量标准。标准度量标准数据被收集和聚合到特定于 WebSphere Process Server 或 WESB 的度量标准类别中，这些类别在调查器树中显示为节点和子-节点。所显示的特定度量标准类别和节点名称取决于您在环境中已部署和访问的组件、服务和资源。

在调查器树中导航时，可根据所选节点查看各操作的低级度量标准或聚合度量标准，从而监控各种 Process Server 或 WESB 组件的总体运行状况。如果根本未使用某个组件，例如，没有为任何流程定义人工任务，则调查器树中不显示该组件的度量标准类别。

查看调查器中的度量标准摘要并进行导航

1. 展开代理节点，然后单击“WProcServer”或“WESB”节点以显示“概览”选项卡来获取摘要信息。

例如，如果选择“WProcServer”节点，“概览”将列出 WebSphere Process Server 正使用的所有业务流程、业务状态机和中介流的摘要信息。

2. 双击业务流程、业务状态机或中介流，以图形格式查看所有关联的度量标准。
3. 展开“WProcServer”或“WESB”节点，以显示顶级 WebSphere Process Server 或 WebSphere Enterprise Service Bus 度量标准类别的子节点。
4. 单击或展开子节点，以显示带有该度量标准类别相关摘要信息的“概览”选项卡。例如，单击“业务规则”节点，在“概览”选项卡上显示业务规则列表。
5. 展开任意子节点，以查看有关单个组件的更详细信息。例如，选择特定业务流程的名称，以图形格式查看该业务流程的度量标准。
6. 展开任意单个对象，以查看该对象的度量标准。例如，展开任意单个 Java 组件的名称，然后展开一个操作以查看该操作的度量标准。

业务对象映射的度量标准

业务对象映射基于源业务对象服务组件中的值将值分配给目标业务对象服务组件。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“BO 映射”或“WESB”>“BO 映射”节点下的目标命名空间和单个业务对象映射。

业务流程的度量标准

业务流程包括按特定顺序执行以达到特定业务目标的各个任务。该度量标准类别仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该度量标准类别不适用。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“业务流程”>“*<business_process_name>*”节点下的单个 WebSphere 业务流程和业务流程步骤。

如果已配置跟踪来显示同一业务流程不同版本的单独度量标准（如[为业务流程的多个版本收集度量标准 \(p. 274\)](#)中所述），则“业务流程”节点下的业务流程名称会添加下划线 [_] 和时间戳来标识同一流程的不同版本。

业务规则的度量标准

业务规则捕获并实施业务策略和实践。该度量标准类别仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该度量标准类别不适用。

所有标准 CA Introscope® 度量标准都可用于在“WProcServer”>“业务规则”>“*<rule_name>*”节点下部署的各个业务规则。

业务状态计算机的度量标准

业务状态机是执行操作以响应事件的事件驱动的应用程序。该度量标准类别仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该类别不适用。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“业务状态机”>“*<business_state_machine_name>*”节点下的各个业务状态机。

如果已配置跟踪 (p. 274) 来为同一业务状态机的各个版本显示单独的度量标准，那么“业务状态机”节点下的名称将显示如下：

- 名称带有下划线 []。
- 时间戳标识相同流程的不同版本。

数据绑定的度量标准

数据绑定用于在入站和出站处理期间将业务对象转换为数据流。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“数据绑定”>“*<data_binding_name>*”或“WESB”>“数据绑定”>“*<data_binding_name>*”节点下的各个数据绑定。

人工任务的度量标准

人工任务表示由人员执行但也与 WebSphere Process Server 中的流程或服务进行交互的任务。该度量标准类别仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该度量标准类别不适用。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“人工任务”节点下的各个人工任务。

接口映射的度量标准

接口映射协调组件之间的差异，从而使具有不同接口的组件可以进行通讯。该度量标准类别仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该度量标准类别不适用。

所有标准 CA Introscope® 度量标准都可用于已在“WProcServer”>“接口映射”节点下定义的接口映射。

对于同步调用，单个接口映射的度量标准表示接口映射组件所花费的总时间。对于异步调用，度量标准分别表示输入和输出转换。

J2CA 适配器的度量标准

适配器是允许 WebSphere Process Server 与外部企业信息系统进行通信的应用程序。当 WebSphere Process Server 调用特定于 EIS 的操作时，会发生出站通信。当应用程序侦听特定 EIS 事件时，会发生入站通讯。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“J2CA”或“WESB”>“J2CA”节点下的 WebSphere Server 和外部系统之间的入站和出站交互。

Java 组件的度量标准

使用 Java 组件可以在 WPS 或 WESB 中自定义 Java 实现。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“Java 组件”或“WESB”>“Java 组件”节点下的各个 Java 组件和操作。

中介流和中介元素的度量标准

中介流包括处理请求的请求流、处理响应的响应流、处理事件的事件流以及处理错误的错误流。针对每个源操作开始请求流，针对每个目标操作开始响应流。中介元素是实现中介流功能的生成块。每个中介流根据需要包含使您可以转换数据的中介元素。

所有标准 CA Introscope® 度量标准都可用于

“WProcServer” > “WESB” > “中介流” 或 “WESB” > “中介流” 节点下的中介流和中介元素。“中介流” 节点列出了具有中介流组件的模块。然后，您可以展开中介流组件，以查看该中介流的聚合度量标准，及其操作、流和中介元素。例如：



如果响应流、事件流或错误流已作为异步调用进行调用，则节点名称在调查器中显示为 `<flow_type>_Asynch`，其中 `<flow_type>` 将流标识为 ResponseFlow、EventFlow 或 FaultFlow。例如，如果使用回调异步调用响应流，则显示在调查器中的节点名称为 ResponseFlow_Asynch。

在中介流度量标准的上下文中，异步调用是使用回调函数异步调用的操作。

关于同步和异步响应时间度量标准

从流名称下操作的所有“平均响应时间”度量标准中聚合中介流组件的度量标准。聚合中介流操作的度量标准的方式有所不同，具体取决于操作使用的是同步调用还是异步调用：

- 如果操作进行同步调用，则从操作的请求流中聚合操作级别的度量标准，因为请求流间接包含同步调用的响应流持续时间。
- 如果操作进行异步调用，则从操作名称下现有的请求流 (**RequestFlow**)、异步响应流 (**ResponseFlow_Asynch**)、异步错误流 (**FaultFlow_Asynch**) 和异步事件流 (**EventFlow_Asynch**) 中聚合操作级别的度量标准。

例如，如果请求流每个时间间隔的平均响应时间为 10 毫秒，异步请求流为 12 毫秒，异步错误流为 8 毫秒，异步事件流为 10 毫秒，请求流的计数值为 1，响应流为 1，错误流为 2，事件流为 3，则该操作的平均响应时间计算为：

$$((10 \times 1) + (12 \times 1) + (8 \times 2) + (10 \times 3))/3$$

关于同步和异步并行调用

中介流组件的度量标准由流名称下各操作的所有“并发调用”度量标准聚合而得。聚合中介流操作的度量标准的方式有所不同，具体取决于操作使用的是同步调用还是异步调用：

- 如果操作进行同步调用，则操作级别的并行调用度量标准表示该操作请求流的并行调用，因为请求流间接包含同步调用的响应流持续时间。
- 如果操作进行异步调用，则从操作名称下的现有请求流 (**RequestFlow**) 以及异步响应流 (**ResponseFlow_Asynch**)、异步错误流 (**FaultFlow_Asynch**) 和异步事件流 (**EventFlow_Asynch**) 的并发调用中聚集操作级别的度量标准。

关于同步和异步错误度量标准

中介流组件的度量标准由流名称下各操作的所有“每个时间间隔的错误”度量标准聚合而得。计算中介流操作的“每个时间间隔的错误”度量标准的方式有所不同，具体取决于操作使用的是同步调用还是异步调用：

- 如果操作进行同步调用，则通过添加请求流的“每个时间间隔的错误”度量标准聚合操作级别的度量标准。
- 如果操作进行异步调用，则通过将现有请求流 (RequestFlow)、异步响应流 (ResponseFlow_Asynch) 和异步事件流 (EventFlow_Asynch) 的现有“每个时间间隔的错误”度量标准添加到操作名称下异步错误流 (FaultFlow_Asynch) 的“每个时间间隔的响应”度量标准中，来聚合操作级别的度量标准。

由于错误发生时应产生错误流，因此在操作级别将异步错误流的“每个时间间隔的响应”度量标准值添加到“每个时间间隔的错误”度量标准值中，以表示在该操作的调用期间已发生错误。

关于同步和异步响应度量标准

中介流组件的度量标准由流名称下各操作的所有“每个时间间隔的响应”度量标准聚合而得。中介流操作的“每个时间间隔的响应”度量标准的计算因操作使用同步还是异步调用而异：

- 如果操作进行同步调用，则会通过添加请求流的“每个时间间隔的响应”来聚集操作级别的度量标准。
- 如果操作进行异步调用，则“每个时间间隔的响应”度量标准由操作名称下的请求流 (RequestFlow) 的最大“每个时间间隔的响应”、异步响应流 (ResponseFlow_Asynch)、异步错误流 (FaultFlow_Asynch) 和异步事件流 (EventFlow_Asynch) 确定。

关于同步和异步停顿计数度量标准

中介流组件的度量标准由流名称下各操作的所有“停顿计数”度量标准聚合而得。中介流操作的“停顿计数”度量标准的计算因操作使用同步还是异步调用而异：

- 如果操作进行同步调用，则通过将为请求流记录的停顿数相加来计算停顿计数。
- 如果操作进行异步调用，则通过将为操作名称下的请求流和异步响应流 (ResponseFlow_Asynch)、异步错误流 (FaultFlow_Asynch) 和异步事件流 (EventFlow_Asynch) 记录的停顿数相加来计算停顿计数。

关联关系的度量标准

关联关系标识语义等效的业务对象。

所有标准 CA Introscope® 度量标准都可用于“WProcServer”>“关联关系”或“WESB”>“关联关系”节点下的各个关联关系。

选择器的度量标准

运行时，使用选择器可以对服务组件进行灵活的处理。- 该度量标准类别仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该度量标准类别不适用。

所有标准 CA Introscope® 度量标准都在“WProcServer”>“选择器”节点下提供。

服务集成总线通信的度量标准

服务集成总线 (SIB) 支持使用基于消息及面向服务的体系结构的应用程序。可以在“WProcServer”>“SIB 通信”节点下查看 SIB 通信度量标准。

WebSphere Process Server 错误的度量标准

当 WPS 层中发生错误时，如果未处理错误，则会抛出异常。“每个时间间隔的 WPS 错误”度量标准显示了在 15 秒的时间片中出现的异常数。该度量标准仅适用于 WebSphere Process Server。如果监控 WebSphere Enterprise Service Bus（作为独立产品），则该度量标准类别不适用。

查看 WPS 和 WESB 默认度量标准组

SOA Extension for WebSphere Process Server and WESB 包括用于定义默认显示板和报警的默认度量标准组。您还可以在自定义显示板和报警中使用这些默认度量标准组。

默认度量标准组在 WebSphere 的企业管理器扩展中打包，作为 SOA Extension for WebSphere Process Server 管理模块 (*WPS_Management_Module.jar*) 或 SOA Extension for WebSphere Enterprise Service Bus 管理模块 (*WESB_Management_Module.jar*) 的一部分。

查看默认的 WebSphere Process Server 或 WESB 度量标准组

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 如果监控所有 WebSphere Process Server 组件，请展开“*超级域*”>“管理模块”>“WPS_ManagementModule (*超级域*)”，或者，如果监控独立的 WESB 服务器，请展开“WESB_ManagementModule (*超级域*)”。
3. 如果监控所有 WebSphere Process Server 组件，请展开“度量标准组”节点以查看 WebSphere Process Server 的所有默认度量标准组，或者，如果监控独立的 WESB 服务器，可查看 WebSphere Enterprise Service Bus 的默认度量标准组。
4. 单击某一特定度量标准组可在“查看器”窗格中查看其定义。

您可以修改任何度量标准组的任意默认设置或自行创建自定义度量标准组。

注意：有关创建或修改度量标准组的详细信息，请参阅《CA APM Workstation 用户指南》。

查看 WPS 和 WESB 默认报警

SOA extension for WebSphere Process Server and WESB 包括在预配置的显示板中使用的默认报警定义。您也可以在自定义的显示板中使用这些默认报警。大部分默认报警预配置有默认的“警告”和“危险”阈值，如果超过阈值或重要级别增加，则会向控制台发送通知。

默认报警定义作为 SOA Extension for WebSphere Process Server 管理模块 (*WPS_Management_Module.jar*) 或 SOA Extension for WebSphere Enterprise Service Bus 管理模块 (*WESB_Management_Module.jar*) 的一部分，打包在 WebSphere 的企业管理器扩展中。

要查看 WebSphere Process Server 或 WESB 代理的默认报警定义，请执行以下操作：

1. 在调查器中，单击“Workstation”>“新建管理模块编辑器”。
2. 如果监控的是所有 WebSphere Process Server 组件，请展开“*超级域*”>“管理模块”>“WPS_ManagementModule (*超级域*)”，或者，如果监控的是独立 WESB 服务器，则请展开“WESB_ManagementModule (*超级域*)”。

3. 展开“报警”节点，查看为 WebSphere Process Server 管理模块定义的所有报警。
4. 在“查看器”窗格中单击特定报警以查看其定义。

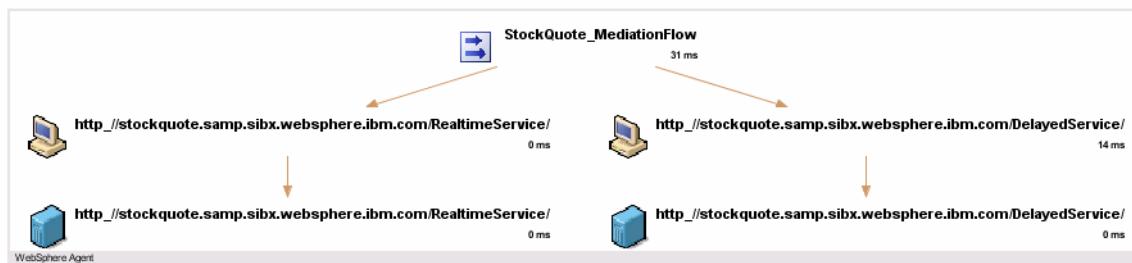
特别是，您应检查默认“警告”和“危险”阈值以及针对严重报警的预定义操作，并对其进行调整使其适用于您的环境。例如，您可能希望调整阈值（如有需要）、添加通知或定义正确的操作。

查看 WPS 或 WESB 依赖关系

通过使用调查器树中的节点，在 WebSphere Process Server 中选择“业务流程”、“业务状态机”或“中介流”或在 WebSphere Enterprise Service Bus 中选择“中介流”，然后单击“SOA 依赖关系地图”选项卡，可以查看业务流程、业务状态机、中介流和适配器出站组件的依赖关系。如果选择调用 EIS 操作的组件，还可以在依赖关系地图中查看适配器出站依赖关系。例如，如果中介流使用适配器从 JDBC 表检索行，则可以通过选择该中介流并显示其依赖关系，在依赖关系地图中显示该中介流和出站适配器之间的依赖关系。

您选择的节点决定了依赖关系地图中显示的上下文。然后您可以将您查看的上下文和详细信息级别卷起以进行折叠或放开以进行展开。例如，对于业务流程中依赖关系的高级视图，您可在调查器中选择业务流程名称，然后单击“SOA 依赖关系地图”选项卡。

以下示例说明了调查器树中选定的股票报价业务服务的中介流，依赖关系地图已展开，以显示其他依赖关系：



您可以继续向地图中添加依赖关系级别，以便查看业务流程的整个工作流，或根据需要在地图的特定节点上进行放大。有关浏览依赖关系地图的更多详细信息，请参阅[使用 SOA 依赖关系地图 \(p. 63\)](#)。

跟踪 WPS 或 WESB 的事务

事务跟踪提供完成业务事务所涉及的特定步骤的详细或摘要视图。对于包括 WebSphere Process Server 或 WebSphere Enterprise Service Bus 组件或 Web 服务的事务，您可以跟踪包括通过以下协议路由的操作的事务：

- 简单对象访问协议 (SOAP)
- 超文本传输协议 (HTTP)
- 安全的超文本传输协议 (HTTPS)
- Java Message Service (JMS)

在 WebSphere Process Server 或 WebSphere Enterprise Service Bus 组件外启动的事务可包括在 WebSphere Process Server 或 WebSphere Enterprise Service Bus 环境中启动的异步调用。要在参加事务的所有组件间跟踪这些事务，在每个组件和操作中插入关联标识符，并将其用作事务步骤。然后可深入查看事务的详细信息，以了解事务中涉及的 WebSphere Process Server 或 WebSphere Enterprise Service Bus 组件，包括所执行的操作和完成每个操作所用的时间。

只要在每个被跟踪的节点上启用 CA APM for SOA 和 CA APM for WebSphere Process Server，您还可以跨任何平台组合跟踪业务事务。这样您就能够查看关于事务的详细信息，即使该事务跨多个 JVM 或 CLR 也是如此。

了解跨进程事务跟踪的值

跨进程事务跟踪提供有关面向服务的体系结构中的松散耦合服务所执行操作的有价值信息。您可以使用跨进程事务跟踪来确定：

- 通过 WebSphere Enterprise Service Bus 组件路由消息的方式。
- 在事务期间调用和执行哪些组件和活动。
- 事务处理期间所做调用的顺序。
- 处理请求或回复最慢的位置。

启动并查看示例事务跟踪

您可以以下述方式启动事务跟踪会话：

- 直接通过 SOA 依赖关系视图中的视图节点。
- 通过单击“Workstation”>“新建事务跟踪会话”，从 Workstation 手工启动。

如果从依赖关系视图启动事务跟踪，则视图节点类型自动设置默认筛选。如果手动启动新的事务跟踪会话，您可以选择以下 WebSphere Process Server 筛选类型之一：

- 适配器节点
- 业务流程
- 业务状态机
- 中介流
- 中介流操作

例如，要筛选事务的特定中介流，可选择中介流筛选并键入中介流的全称或部分名称。

配置筛选并启动事务跟踪会话后，将显示事务跟踪查看器。您可以选择跟踪来查看有关在事务中执行的调用的其他详细信息，包括使用 WebSphere Process Server 执行的任何业务流程、状态更改或 Java 组件。

为 WebSphere 流程使用顺序视图

与 WebSphere Process Server 相关的事务通常包括异步调用，因此您可能会发现，单击“顺序视图”即可查看在事务处理过程中异步执行的流程的事务工作流。“顺序视图”显示流程执行的顺序（以该顺序可确定为限度）。对于 WebSphere Process Server 事务，顺序不一定表示传统的调用方-被调用方关联关系，而是表示某一流程在何时会触发其他流程的执行。

不过，您应该注意的是，WebSphere Process Server 和 WESB 流程的处理时间是使用流程从开始到完成的整个持续时间计算而得，包括与所调用流程相关联的处理时间。WebSphere Process Server 和 WESB 流程不支持净持续时间，该持续时间是从调用流程的持续时间中减去非阻塞同步流程和异步流程的处理时间而得。

有关事务跟踪的详细信息，请参阅[在 SOA 环境中使用事务跟踪 \(p. 85\)](#)。

注意：有关配置跟踪的信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。有关使用事务跟踪视图和历史数据的详细信息，请参阅《CA APM Workstation 用户指南》。

配置跟踪以包括业务流程活动/业务状态机活动

默认情况下，WebSphere Process Server 业务流程/业务状态机的事务跟踪包括活动级别的性能信息。这些跟踪不包括有关执行活动本身的其他详细信息。由于检索活动名称会存在较大的性能开销，因此默认情况下会关闭其他活动级别的跟踪。

可以手工打开活动跟踪来查看有关在业务流程/业务状态机中执行的活动的详细信息。但是，请仅在有限时间内打开活动跟踪。

遵循这些步骤：

1. 停止 WebSphere Process Server 或 WESB 应用程序服务器。
2. 转到 *<Agent_Home>* 目录，然后在文本编辑器中打开 *wps.pbd* 文件。
3. 取消注释 *ActivityTracing* 和 *ActivityResponseTracing* 标志：

```
TurnOn: ActivityTracing  
TurnOn: ActivityResponseTracing
```

将启用其他活动级别的跟踪。

注意：如果已[启用参数](#) (p. 274)以获取时间属性，请取消注释下面的 *ActivityTracer*。然后，“活动”节点将显示在同一个“业务流程” / “业务状态机”节点下，而不会显示在另外的“业务流程” / “业务状态机”节点中。

```
#SetTracerParameter: ActivityTracer nameformatter  
com.wily.powerpack.webspherereprocserver.nameformatter.ActivityTimeNameform  
atter
```

4. 保存并关闭 *wps.pbd*。
5. 启动 WebSphere Process Server 或 WESB 服务器。
代理会使用新的配置重新启动。

附录 A：特定于 SOA 的代理配置属性

本节介绍了专用于配置 CA APM for SOA 的代理属性。

此部分包含以下主题：

[了解代理属性的位置 \(p. 295\)](#)

[配置代理属性 \(p. 295\)](#)

[关于特定于 SOA 的代理属性 \(p. 296\)](#)

了解代理属性的位置

CA APM for SOA 代理属性存储在代理配置文件 *IntroscopeAgent.profile* 中。该配置文件是属性名称和属性值的文本文件。代理按以下方式搜索代理配置文件：

- 在由 *com.wily.introscope.agentProfile* 系统属性指定的位置中。
- 如果未设置 *com.wily.introscope.agentProfile* 属性，则代理将使用 *IntroscopeAgent.profile* 文件。

在大多数情况下，必须先重新启动代理监控的应用程序服务器，对代理属性进行的更改才能生效。

配置代理属性

通过代理属性，您可以控制代理的行为和操作，并自定义适合您的环境的设置。CA APM for SOA 包括多种特定于 SOA 且带有默认设置的代理属性。请先将属性名称添加到 *IntroscopeAgent.profile* 文件并为属性设置有效值，然后才能使用这些属性。

遵循这些步骤：

1. 在文本编辑器中打开 *IntroscopeAgent.profile* 文件。
2. 标识要从[特定于 SOA 的代理属性 \(p. 296\)](#)列表中添加到 *IntroscopeAgent.profile* 文件的属性。
3. 在 *IntroscopeAgent.profile* 文件中特定于 SOA 的部分键入属性的全名，包括 *com.wily.introscope* 前缀。例如：

```
#####
# SOA Performance Management Agent Settings
# =====
com.wily.introscope.agent.httpheaderread.enabled
```

4. 根据需要，设置属性值。例如：

```
com.wicom.wily.introscope.agent.httpheaderread.enabled=true
```

5. 保存并关闭 *IntroscopeAgent.profile* 文件。

将属性添加到 *IntroscopeAgent.profile* 文件后，您可以根据需要更改属性值。

关于特定于 SOA 的代理属性

您可以在 *IntroscopeAgent.profile* 中设置以下特定于 SOA 的属性。所有属性均以 com.wily.introscope 前缀开头，但是为了便于阅读，不列出属性的前缀。

注意：有关其他代理属性的信息，请参阅《CA APM Java 代理实施指南》或《CA APM .NET 代理实施指南》。

agent.httpheaderinsertion.enabled (p. 297)

启用或禁用在 HTTP 头中插入客户端相关性信息。

agent.httpheaderread.enabled (p. 298)

启用或禁用从 HTTP 头中进行相关性信息的服务器端检索。

agent.soapheaderinsertion.enabled (p. 299)

启用或禁用在 SOAP 头中插入客户端相关性信息。

agent.soapheaderread.enabled (p. 300)

启用或禁用从 SOAP 头中进行相关性信息的服务器端检索。

agent.soa.metricNameFormatting (p. 300)

修改 CA APM for SOA 度量标准名称，以使用下划线 [_] 替换任何指定的一个或多个字符。

agent.transactiontrace.boundaryTracing.cacheFlushFrequency (p. 301)

指定刷新缓存前依赖关系数据可在代理内存中保留的天数。

agent.transactiontrace.boundaryTracing.enable (p. 302)

启用或禁用事务的边界跟踪。

soa.client.prependhandler (p. 302)

控制 SOAP 头在客户端的插入点。

soa.server.appendhandler (p. 303)

控制服务器端 SOAP 头的检索和删除。

在大多数情况下，更改代理属性后需要重新启动应用程序服务器才能使属性更改生效。

agent.httpheaderinsertion.enabled

启用或禁用在 HTTP 头中插入客户端关联标识符。

SAP NetWeaver 不支持该属性。

属性设置

可将属性设置为 true 或 false:

- True—将客户端相关性信息和依赖关系信息添加到 HTTP 头。
- False—不允许更改 HTTP 头中的客户端相关性信息和依赖关系信息。

默认

对于 CA APM for SOA、CA APM for WebSphere Process Server 和 CA APM for WebSphere Enterprise Service Bus 为 False。

对于 CA APM for Oracle Service Bus、CA APM for TIBCO BusinessWorks 和 CA APM for webMethods Integration Server 为 True。

示例

```
com.wily.introscope.agent.httpheaderinsertion.enabled=true
```

注释

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

agent.httpheaderread.enabled

启用或禁用从 HTTP 头中进行关联标识符的服务器端检索。

SAP NetWeaver 不支持该属性。

属性设置

可将属性设置为 true 或 false:

- True—从 HTTP 头读取服务器端相关性信息和依赖关系信息。
- False—不检查 HTTP 头中是否存在相关性信息和依赖关系信息。

默认

对于 CA APM for SOA、CA APM for WebSphere Process Server 和 CA APM for WebSphere Enterprise Service Bus 为 False。

对于 CA APM for Oracle Service Bus、CA APM for TIBCO BusinessWorks 和 CA APM for webMethods Integration Server 为 True。

示例

```
com.wily.introscope.agent.httpheaderread.enabled=true
```

注释

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

agent.soapheaderinsertion.enabled

启用或禁用在 SOAP 头中插入客户端关联标识符。

SOA 依存关系视图和跨进程事务跟踪需要关联标识符从一个流程传递到另一个流程。代理可将该关联标识符插入 SOAP 或 HTTP 头。

如果由于意外头条目导致基于 SOAP 的应用程序不能正常运行，应将该属性设置为 `false`。如果将该属性设置为 `false`，且不启用在 HTTP 标头中插入关联标识符，则无法在 SOA 依存关系视图或跨进程事务跟踪中显示相关信息。

属性设置

可将属性设置为 `true` 或 `false`:

- `True`—将客户端相关性信息和依赖关系信息添加到 SOAP 头。
- `False`—不允许更改 SOAP 头中的客户端相关性信息和依赖关系信息。

默认

`True`

示例

```
com.wily.introscope.agent.soapheaderinsertion.enabled=true
```

注释

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

agent.soapheaderread.enabled

启用或禁用从 SOAP 头中进行关联标识符的服务器端检索。

属性设置

可将属性设置为 true 或 false:

- True—从 SOAP 头读取服务器端相关性信息和依赖关系信息。
- False—不检查 SOAP 头中是否存在相关性信息和依赖关系信息。

默认

True

示例

com.wily.introscope.agent.soapheaderread.enabled=true

注释

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

agent.soa.metricNameFormatting

使用下划线字符 (_) 替换 CA APM for SOA 度量标准名称中的任何指定的一个或多个字符。

例如，设置属性 com.wily.introscope.agent.soa.metricNameFormatting 以替换正斜杠 (/) 会导致 URL http://CheckingAccount/demobank.ca.com 显示为：

http_CheckingAccount_demobank.ca.com

如果希望在度量标准名称中使用下划线 (_) 替换指定字符，请设置该属性。

您可能希望执行此替换，以便采用 Web Services Manager 中使用的格式显示度量标准名称。在这种情况下，您还必须在 webservices.pbd 文件中将命名空间替换为服务名称。这些更改将导致显示在调查器树或 SOA 依赖关系视图中的 CA APM for SOA 度量标准名称使用服务名称，而不是命名空间。

属性设置

可以将该属性设置为要在度量标准名称中使用下划线 (_) 字符替换的一个或多个字符。

如果未定义属性或属性没有值，则不会执行字符替换。

默认

无

示例

```
com.wily.introscope.agent.soa.metricNameFormatting=/:
```

注释

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

agent.transactiontrace.boundaryTracing.cacheFlushFrequency

指定刷新缓存前依赖关系数据可在代理内存中保留的天数。可以设置该属性以定期刷新空闲线程和清除代理缓存。在指定时间段结束时，刷新缓存会导致在代理中删除 SOA 依赖关系视图的依赖关系数据。接着，代理会重新发现依赖关系数据，并将数据转发到企业管理器，以刷新 SOA 依赖关系视图中显示的信息。

默认情况下，以前发现的依赖关系会在代理缓存中保留 30 天，以防止代理不必要地将有关已知依赖关系的信息发送至企业管理器。在大多数情况下，30 天是可接受的时间段，因为依赖关系数据相当固定，将以前发现的依赖关系存储在代理缓存中将减少代理必须在常规时间间隔内传输给企业管理器的数据。

如果运行中的代理与独立企业管理器中断连接，或故障切换到非群集（无 MOM）环境中的新企业管理器，则可以将该属性设置为 1，以便每天刷新代理缓存。将该属性设置为 1 会从缓存中删除已知依赖关系，并强制代理重新发现这些依赖关系，以便将数据发送到新企业管理器。

属性设置

可将属性设置为任何大于零的整数。

默认

30 天

示例

```
com.wily.introscope.agent.transactiontrace.boundaryTracing.cacheFlushFrequency=1
```

注释

对该属性所做的更改将即时生效，无需重新启动托管应用程序服务器。

agent.transactiontrace.boundaryTracing.enable

启用或禁用事务跟踪边界跟踪。如果要在 SOA 依赖关系地图中显示信息，必须启用边界跟踪。如果未使用 SOA 依赖关系地图，则可以禁用该属性以减少开销。

会从代理缓存中以特定大小发送边界跟踪，以避免出现开销问题。

属性设置

可将属性设置为 true 或 false:

- True—收集 SOA 依赖关系地图所需的事务跟踪边界跟踪信息，并将其发送到企业管理器。
- False—不收集 SOA 依赖关系地图所需的事务跟踪边界跟踪信息。

默认

True

示例

```
com.wily.introscope.agent.transactiontrace.boundaryTracing.enable=true
```

注释

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

soa.client.prependhandler

控制 CA APM for SOA SOAP 头在客户端的插入点。默认情况下，SOAP 处理程序链中的第一个处理程序插入 CA APM for SOA SOAP 头。您可以使用该属性使 SOAP 头由 SOAP 处理程序链中的第一个或最后一个处理程序插入，具体取决于应用程序处理 SOAP 消息的方式。

该属性仅适用于使用 SOAP 处理程序的应用程序服务器，并取决于所使用的 SOAP 引擎和 API。例如，当应用程序服务器运行旧版本的 WebLogic 或 WebSphere 或者当应用程序服务器运行 SAP NetWeaver 时，可使用该属性。如果使用 Apache Axis、Apache CXF、native JBoss、WebLogic 10.x (JAX-WS)、WebSphere 7.0 (JAX-WS)、.NET 或 Spring Web 服务，则该属性不适用。

属性设置

您可以将此属性设置为 True 或 False:

- True—使用 SOAP 处理程序链中的第一个处理程序插入 SOAP 头。
- False—使用 SOAP 处理程序链中的最后一个处理程序添加 SOAP 头。

默认**True****示例**`com.wily.introscope.soa.client.prependhandler=true`**注释**

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

[soa.server.appendhandler](#)

控制服务器端 CA APM for SOA SOAP 头的检索和删除。默认情况下，SOAP 处理程序链中的最后一个处理程序读取 CA APM for SOA SOAP 头。根据应用程序处理 SOAP 消息的方式，您可以使用该属性在 SOAP 处理程序链中首先或最后读取或删除 SOAP 头。

该属性仅适用于使用 SOAP 处理程序的应用程序服务器，并取决于所使用的 SOAP 引擎和 API。例如，当应用程序服务器运行旧版本的 WebLogic 或 WebSphere 或者当应用程序服务器运行 SAP NetWeaver 时，可使用该属性。如果使用 Apache Axis、Apache CXF、native JBoss、WebLogic 10.x (JAX-WS)、WebSphere 7.0 (JAX-WS)、.NET 或 Spring Web 服务，则该属性不适用。

属性设置

您可以将此属性设置为 True 或 False:

- **True**—使用 SOAP 处理程序链中的最后一个处理程序读取 CA APM for SOA SOAP 头，然后将其删除。
- **False**—使用 SOAP 处理程序链中的第一个处理程序读取 CA APM for SOA SOAP 头，然后将其删除。

默认**True****示例**`com.wily.introscope.soa.server.appendhandler=false`**注释**

您必须重新启动托管应用程序，对此属性所做的更改才能生效。

附录 B：特定于 SOA 的企业管理器配置属性

本节介绍了专用于配置 CA APM for SOA 的企业管理器属性。

此部分包含以下主题：

[配置企业管理器属性 \(p. 305\)](#)

[关于特定于 SOA 的企业管理器属性 \(p. 306\)](#)

配置企业管理器属性

通过企业管理器配置属性，您可以控制企业管理器的行为和操作，并自定义适合您的环境的设置。CA APM for SOA 包括许多特定于 SOA 且带有默认设置的企业管理器属性。但是，必须先将属性名称添加到 *IntroscopeEnterpriseManager.properties* 文件并更新其属性值，才可以使用这些属性。

将配置属性添加到 *IntroscopeEnterpriseManager.properties*:

1. 在文本编辑器中打开 *<EM_Home>/IntroscopeEnterpriseManager.properties* 文件。
2. 标识要从“关于特定于 SOA 的企业管理器属性”中的属性列表添加到 *IntroscopeEnterpriseManager.properties* 文件的属性。
3. 在 *IntroscopeEnterpriseManager.properties* 文件特定于 SOA 的部分键入属性的完整名称，包括 *com.wily.introscope* 前缀。例如：

```
#####
# SOA 性能管理 EM 设置
# -----
com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count
```
4. 根据需要，设置属性值。例如，使用该属性的默认设置：
com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count=10
5. 保存并关闭 *IntroscopeEnterpriseManager.properties* 文件。

将企业管理器扩展属性添加到 *IntroscopeEnterpriseManager.properties* 文件后，可以根据需要更改属性值。

更改企业管理器配置属性值:

1. 在文本编辑器中打开
`<EM_Home>/IntroscopeEnterpriseManager.properties` 文件。
2. 找到要更改的属性，并根据环境需要设置新值。例如：
`com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count=15`
以上示例仅是说明，并非 CA APM for SOA 设置建议。
3. 保存并关闭 `IntroscopeEnterpriseManager.properties` 文件。
4. 重新启动企业管理器。

关于特定于 SOA 的企业管理器属性

您可以在 `IntroscopeEnterpriseManager.properties` 文件中设置以下特定于 SOA 的属性。所有属性均以 `com.wily.introscope` 前缀开头，但是为了便于阅读，不列出属性的前缀。

注意：有关设置其他企业管理器属性的信息，请参阅《CA APM 配置和管理指南》。

soa.dependencymap.aging.refresh.interval (p. 308)

指定检查以前发现的依赖关系期限的时间间隔。

soa.dependencymap.aging.expire.interval (p. 309)

指定依赖关系过期前可在 SOA 依赖关系视图中保留的最大天数。

soa.dependencymap.heuristics.clientside.enable (p. 309)

启用或禁用客户端逻辑等效启发式规则。

soa.dependencymap.heuristics.namematch.enable (p. 310)

启用或禁用逻辑等效启发式名称匹配规则。

soa.dependencymap.heuristics.serverside.enable (p. 311)

启用或禁用服务器端逻辑等效启发式规则。

soa.dependencymap.log.suppress (p. 312)

指定相同错误或警告消息可写入日志文件的最大次数。

soa.dependencymap.max.edge.ratio (p. 313)

指定视图中依赖关系与节点的最大比率。

soa.dependencymap.max.vertices (p. 314)

指定独立或收集器企业管理器中的依赖关系视图可存储的最大节点数。

soa.deviation.enable (p. 315)

启用或禁用生成偏差度量标准所需的计算。

soa.deviation.art.enable (p. 316)

启用或禁用“平均响应时间”偏差度量标准。

soa.deviation.dependencymetric.enable (p. 316)

启用或禁用生成依赖关系度量标准所需的计算。

soa.deviation.count.per.metric (p. 317)

指定每个偏差度量标准的最大操作数。

soa.deviation.dependency.refreshrate (p. 318)

指定缓存的 SOA 依赖关系视图依赖关系数据的刷新频率（以小时为单位）。

soa.deviation.errors.enable (p. 319)

启用或禁用“每个时间间隔的错误”偏差度量标准。

soa.deviation.max.metric.count (p. 319)

指定要报告的偏差度量标准的最大数目。

soa.deviation.metric.expressionlist (p. 320)

定义偏差度量标准表达式的名称列表。

soa.deviation.metric.calledbackends (p. 321)

使用在 deviation.metric.expressionlist 属性中定义的名称构建新属性，以便创建偏差度量标准。

soa.deviation.usage.enable (p. 322)

启用或禁用“每个时间间隔的响应偏差”度量标准。

soa.dashboard.typeviewer.average.enable (p. 323)

启用或禁用所选操作的“平均响应时间偏差”图表的显示。

soa.dashboard.typeviewer.response.enable (p. 325)

启用或禁用选定操作的“每个时间间隔的响应偏差”图表的显示。

soa.dashboard.typeviewer.errors.enable (p. 324)

启用或禁用选定操作的“每个时间间隔的错误偏差”图表的显示。

soa.dashboard.typeviewer.mostcritical.enable (p. 326)

启用或禁用“最关键操作”显示板和“最关键”选项卡的显示。

soa.dashboard.typeviewer.mostcritical.count (p. 327)

指定报告的关键操作的最大数目。

[soa.dashboard.typeviewer.mostdependent.enable \(p. 328\)](#)

启用或禁用“最依赖操作”显示板和“最依赖”选项卡的显示。

[soa.dashboard.typeviewer.mostdependent.count \(p. 329\)](#)

指定所报告的最大依赖操作数。

在大多数情况下，必须重新启动企业管理器，属性更改才能生效。

[soa.dependencymap.aging.refresh.interval](#)

指定企业管理器执行下一个 SOA 依赖关系地图依赖关系老化检查前要经历的小时数。

企业管理器跟踪每个发现的依赖关系的期限，并定期重新发现依赖关系来确保它们仍然存在。依赖关系的期限基于其最近的发现。

在 SOA 依赖关系地图依赖关系老化检查期间，企业管理器可确定 SOA 依赖关系地图中每个依赖关系的期限。

默认情况下，SOA 依赖关系地图的刷新时间间隔定义为 1 小时。因此，六个 SOA 依赖关系地图的刷新时间间隔将持续 6 小时。

属性设置

可将属性设置为任何大于零的整数。

默认

6 小时

示例

`com.wily.introscope.soa.dependencymap.aging.refresh.interval=6`

注释

- 该属性与 `com.wily.introscope.soa.dependencymap.aging.expire.interval` 属性配合使用，以确定是否应删除依赖关系。
- 对该属性所做的更改将即时生效，无需重新启动企业管理器。

soa.dependencymap.aging.expire.interval

SOA 依赖关系地图中的依赖关系的最大期限（以天为单位）。在依赖关系老化检查期间，企业管理器从 SOA 依赖关系地图中删除所有期限超过指定值的依赖关系。

例如，如果 *com.wily.introscope.soa.dependencymap.aging.expire.interval* 的值为 60 天，则在过去 60 天内未被重新发现的依赖关系为过期的依赖关系。- 如果某个依赖关系在此期间未被重新发现，则会从依赖关系地图中将其删除。

属性设置

可将属性设置为任何大于零的整数。

默认

60 天

示例

```
com.wily.introscope.soa.dependencymap.aging.expire.interval=90
```

注释

- 该属性可与 *com.wily.introscope.soa.dependencymap.aging.refresh.interval* 属性配合使用。
- 该属性确定删除依赖关系的条件，随后将在 SOA 依赖关系地图依赖关系老化检查过程中应用该条件。
- 对该属性所做的更改将即时生效，无需重新启动企业管理器。

soa.dependencymap.heuristics.clientside.enable

启用或禁用 CA APM for SOA 客户端逻辑等效启发式规则。

客户端逻辑等效启发式规则表明了以下内容：

当客户端类型的两种物理服务操作依赖于服务器类型的同一个物理服务操作时，将这两种操作视为在逻辑上等效。

当两个不同的应用程序执行相同的客户端服务操作调用时，会出现此设置。当两种操作同时包含或不包含相同的度量标准路径（不包括代理说明符）时，该规则将检测此逻辑等效。

属性设置

您可以将此属性设置为 True 或 False:

- True—应用 CA APM for SOA 客户端逻辑等效启发式规则。
- False—禁用 CA APM for SOA 客户端逻辑等效启发式规则。

默认

True

示例

`com.wily.introscope.soa.dependencymap.heuristics.clientside.enable=true`

注释

如果该规则更改，不会对以前发现的依赖关系应用更改。因此，除非停止企业管理器并删除以前保存的 SOA 依赖关系视图文件类型，否则更改规则值可能会导致不可预测的结果。

对此属性所做的更改会立即生效，不需要重新启动企业管理器。

详细信息:

[忽略或删除保存的依赖关系数据 \(p. 67\)](#)

soa.dependencymap.heuristics.namematch.enable

启用或禁用 CA APM for SOA 逻辑等效启发式名称匹配规则。

当两种操作具有相同的度量标准路径（不包括代理说明符）时，逻辑等效启发式名称匹配规则将检测逻辑等效。

CA APM for SOA 逻辑等效启发式名称匹配规则表明，删除代理说明符后，具有相同度量标准路径的两个物理 Web 服务操作被视为在逻辑上等效。

属性设置

您可以将此属性设置为 True 或 False:

- True—启用逻辑等效启发式名称匹配规则。
- False—禁用逻辑等效启发式名称匹配规则。

默认**False****示例**`com.wily.introscope.soa.dependencymap.heuristics.namematch.enable=true`**注释**

如果该规则更改，不会对以前发现的依赖关系应用更改。因此，除非停止企业管理器并删除以前保存的 SOA 依赖关系视图文件类型，否则更改规则值可能会导致不可预测的结果。

如果在 CA APM for SOA 中创建虚拟代理，请启用此属性。

对此属性所做的更改会立即生效，不需要重新启动企业管理器。

详细信息：

[忽略或删除保存的依赖关系数据 \(p. 67\)](#)

soa.dependencymap.heuristics.serverside.enable

启用或禁用 CA APM for SOA 服务器端逻辑等效启发式规则。

服务器端逻辑等效启发式规则表明，当且仅当存在类型客户端的某物理服务器操作依赖于类型服务器的两种物理服务操作时，将这两种操作视为在逻辑上等效。

通常，仅当 SOA 合并某种负载平衡机制，以将客户端 Web 服务操作调用分发到其中一个可用的服务器端 Web 服务操作实施中时，才会出现这种情况。 -

属性设置

您可以将此属性设置为 True 或 False:

- True—启用 CA APM for SOA 服务器端逻辑等效启发式规则。
- False—禁用 CA APM for SOA 服务器端逻辑等效启发式规则。

默认**True****示例**`com.wily.introscope.soa.dependencymap.heuristics.serverside.enable=true`

注释

如果该规则更改，不会对以前发现的依赖关系应用更改。因此，除非停止企业管理器并删除以前保存的 SOA 依赖关系视图文件类型，否则更改规则值可能会导致不可预测的结果。对此属性所做的更改会立即生效，不需要重新启动企业管理器。

详细信息：

[忽略或删除保存的依赖关系数据 \(p. 67\)](#)

soa.dependencymap.log.suppress

指定在阻止重复消息之前，相同错误或警告消息可写入日志文件的最大次数。

使用该属性，可阻止每次出现重复的错误或警告消息时都将其发送到企业管理器日志文件。例如，如果您的依赖关系地图超过所允许的最大节点数或最大依赖关系数，则每次达到最大值时，都会在日志文件中记录警告消息。使用该属性，当错误或警告发送到日志文件的次数已达到指定次数后，会将指示消息已被阻止的文本字符串添加到最后一次错误或警告，而不会在日志文件中记录以后的错误或警告，直到重新启动企业管理器或修改该属性为止。

属性设置

可将属性设置为任何大于零的整数。

默认

5 个日志条目

示例

`com.wily.introscope.soa.dependencymap.log.suppress=5`

注释

该属性不需要您重新启动企业管理器。

但是，如果更改属性值，阻止错误和警告消息的计数将重新开始。例如，如果特定错误消息出现了 5 次，默认会阻止该消息。如果之后将属性值更改为 6，将不再阻止该错误消息，且该错误必须出现 6 次以上才会被阻止。

soa.dependencymap.max.edge.ratio

指定视图中依赖关系与节点的最大比率。使用该属性，您可以控制存储在大型或复杂的 SOA 环境中的企业管理器上的依赖关系地图的整体复杂性。

存储在企业管理器中的依赖关系地图数据通常表示整个发现的应用程序中的所有依赖关系，提供了已部署的面向服务的体系结构的完整模型。但是，在非常大或复杂的 SOA 环境中，完整表示所有 SOA 组件及其依赖关系可能会影响企业管理器自身的性能和操作。使用该属性，您可以指定节点与依赖关系之间的比率，以控制允许的复杂性级别。达到该限制后，将不再保存其他依赖关系，并在企业管理器日志文件中写入警告。

您还可以将该属性与 com.wily.introscope.soa.dependencymap.max.vertices 属性配合使用，以限制依赖关系地图中存储的节点总数。

在群集环境中，该属性仅适用于收集器企业管理器。您可以使用 MOM 存储收集器间组合的 SOA 环境的完整表示。

属性设置

可将属性设置为任何大于零的整数。

默认

每个节点 5 个依赖关系

示例

```
com.wily.introscope.soa.dependencymap.max.edge.ratio=5
```

注释

使用 com.wily.introscope.soa.dependencymap.max.vertices 和 com.wily.introscope.soa.dependencymap.max.edge.ratio 属性的默认值，可将依赖关系地图限制为 5000 个节点和最多 25000 个依赖关系 (5×5000)。

大多数 SOA 网络的组件少于 5000 个，且依赖关系比率为每个组件 1 个或 2 个依赖关系。因此，默认设置所允许的复杂性要超出大多数组织所需要的复杂性。可以使用这些属性有目的性地限制依赖关系地图的大小和复杂性，但这样做可能会导致 SOA 依赖关系地图模型出现不必要的不完整。

soa.dependencymap.max.vertices

指定独立或收集器企业管理器中的依赖关系视图可存储的最大节点数。您可以使用该属性来控制存储在大型或复杂 SOA 环境中的企业管理器上的依赖关系地图的最大大小。

存储在企业管理器中的依赖关系地图数据通常表示整个发现的应用程序中的所有依赖关系，提供了已部署的面向服务的体系结构的完整模型。但是，在非常大或复杂的 SOA 环境中，完整表示所有 SOA 组件及其依赖关系可能会影响企业管理器自身的性能和操作。使用该属性，您可以通过指定要包括的最大节点数来限制 SOA 模型的大小。达到该限制时，不会再保存任何其他信息，但警告则会写入企业管理器日志文件。

您还可以将该属性与 com.wily.introscope.soa.dependencymap.max.edge.ratio 属性结合使用，以限制存储在依赖关系地图中的依赖关系的总数。

在群集环境中，该属性仅适用于收集器企业管理器。您可以使用 MOM 存储收集器间组合的 SOA 环境的完整表示。

属性设置

可将属性设置为任何大于零的整数。

默认

5000 个节点

示例

```
com.wily.introscope.soa.dependencymap.max.vertices=5000
```

soa.deviation.enable

启用或禁用生成以下偏差度量标准所需的计算：

- 平均响应时间偏差
- 每个时间间隔的错误偏差
- 每个时间间隔的响应偏差

如果将该属性设置为 true，则会收集并可报告偏差度量标准。随后可以选择地单独启用或禁用每个偏差度量标准。

属性设置

可将属性设置为 true 或 false：

- True—执行计算以提供偏差度量标准数据。
- False—不报告任何偏差度量标准。

默认

True

示例

```
com.wily.introscope.soa.deviation.enable=true
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.art.enable

启用或禁用“平均响应时间”偏差度量标准。如果您计算偏差度量标准，该属性可确定是否报告平均响应时间偏差度量标准。

属性设置

可将属性设置为 true 或 false:

- True—执行计算并报告“平均响应时间偏差”度量标准数据。
- False—不报告“平均响应时间偏差”度量标准。

默认

True

示例

```
com.wily.introscope.soa.deviation.art.enable=true
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.dependencymetric.enable

该属性启用或禁用生成以下依赖关系度量标准所需的计算:

- 关键直接数
- 关键间接数
- 依赖关系直接数
- 依赖关系间接数

该属性用于确定是否报告所有 CA APM for SOA 依赖关系度量标准。

属性设置

您可以将此属性设置为 True 或 False:

- True—执行计算以提供依赖关系度量标准数据。
- False—不执行依赖关系度量标准计算。

默认

True

示例

```
com.wily.introscope.soa.deviation.dependencymetric.enable=true
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.count.per.metric

此属性指定在计算偏差度量标准的 Web 服务节点下操作的最大数目。在报告操作数目超过此数目时，使用每个服务器和客户端节点命名空间下的最关键操作。

属性设置

可将属性设置为任何大于零的整数。

默认

每个依存关系度量标准 25 个操作

示例

```
com.wily.introscope.soa.deviation.count.per.metric=25
```

注释

要使此属性生效，请重新启动企业管理器。

soa.deviation.dependency.refreshrate

指定缓存的 SOA 依赖关系视图依赖关系数据的刷新频率（以小时为单位）。

SOA 依赖关系地图数据由偏差度量标准服务缓存。每 15 秒报告一次这些服务和操作的关键、依赖和偏差度量标准。

根据该属性定期刷新缓存的 SOA 依赖关系地图数据。

该地图很少更改，通常仅当部署更改时才会更改。

属性设置

可将属性设置为任何大于零的整数。

默认

1 小时

示例

```
com.wily.introscope.soa.deviation.dependency.refreshrate=1
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.errors.enable

启用或禁用错误偏差度量标准。该属性用于确定在您计算偏差度量标准时是否报告错误计数偏差度量标准。

属性设置

可将属性设置为 true 或 false:

- True—执行计算并报告“每个时间间隔的错误偏差”度量标准数据。
- False—不报告“每个时间间隔的错误偏差”度量标准。

默认

True

示例

```
com.wily.introscope.soa.deviation.errors.enable=true
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.max.metric.count

指定要报告的偏差度量标准的最大数目。

CA Technologies 不建议更改此属性的默认值。默认值报告的度量标准足以提供有用信息，并且不会影响企业管理器性能。如果更改默认值，企业管理器可能会遇到性能问题。

属性设置

可将属性设置为任何大于零的整数。

默认

总共 1000 个偏差度量标准

示例

```
com.wily.introscope.soa.deviation.metric.count=1000
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.mean.days

此属性指定累积操作以计算平均值的天数。在指定天数后自动计算平均值。

属性设置

可将属性设置为任何大于零的整数。

默认

7 天。

示例

```
com.wily.introscope.soa.deviation.mean.days=3
```

注释

要使此属性生效，请重新启动企业管理器。

soa.deviation.metric.expressionlist

此属性定义偏差度量标准表达式的名称列表。将该属性设置为由逗号分隔的、要用于创建偏差度量标准的名称的列表。每个名称均用作单独的度量标准表达式。

deviation.metric.calledbackends 属性引用此属性以创建用户定义的偏差度量标准。

属性设置

可将该属性设置为任何用户定义的名称列表。

默认

```
calledbackends
```

示例

```
deviation.metric.expressionlist=alpha1, beta2, gama3
```

计算其他偏差

默认情况下，只为单个操作计算偏差度量标准。为使偏差在 Web 服务、客户端或服务器命名空间级别可用，请将以下语句添加到 *IntroscopeEnterpriseManager.properties* 文件。

```
com.wily.introscope.soa.deviation.metric.expressionlist=test,test1  
com.wily.introscope.soa.deviation.metric.test=webServices\\|client\\|.*  
com.wily.introscope.soa.deviation.metric.test1=webServices\\|Server\\|.*
```

重要信息！如果以这种方式在多个级别计算偏差，需要额外处理开销。

注释

要使此属性的更改生效，请重新启动企业管理器。

soa.deviation.metric.calledbackends

使用在 *deviation.metric.expressionlist* 属性中定义的名称构建新属性，以便创建偏差度量标准。

新属性名称的格式如下：

```
com.wily.introscope.soa.deviation.metric.<user-defined_name> = <user-defined  
regular expression>
```

分配给 *deviation.metric.calledbackends* 属性的值为度量标准表达式。例如：

```
com.wily.introscope.soa.deviation.metric.alpha1=Frontends/Called Backends  
com.wily.introscope.soa.deviation.metric.beta2=Frontends/Called Backends
```

默认

calledbackends

示例

```
com.wily.introscope.soa.deviation.metric.calledbackends=Frontends/Frontends/F  
BApp$Frontend/Called Backends/FBApp$Backend
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.deviation.usage.enable

启用或禁用“每个时间间隔的响应偏差”度量标准。该属性用于确定在您计算偏差度量标准时是否报告响应计数偏差度量标准。

属性设置

可将属性设置为 true 或 false:

- True—执行计算并报告“每个时间间隔的响应偏差”度量标准数据。
- False—不报告“每个时间间隔的响应偏差”度量标准。

默认

True

示例

```
com.wily.introscope.soa.deviation.usage.enable=true
```

注释

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.average.enable

使针对所选操作收集的偏差数据显示在以下位置的“平均响应时间偏差”图表中：

- SOA 性能—“最关键操作”显示板
- SOA 性能—“最依赖操作”显示板
- “最关键”选项卡
- “最依赖”选项卡

仅当偏差数据是由代理报告到企业管理器，且允许在工作站中显示时，才可在显示板和选项卡中使用偏差数据。有关报告偏差数据的详细信息，请参阅[查看“偏差”选项卡中的偏差度量标准 \(p. 56\)](#)。

属性设置

可将属性设置为 true 或 false：

- True—使所选操作的数据显示在“平均响应时间偏差”图表中。
- False—禁止在“平均响应时间偏差”图表中显示所选操作的数据。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.average.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果同时在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中定义该属性，将使用 *IntroscopeWorkstation.properties* 属性设置。

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.errors.enable

使收集的选定操作的偏差数据显示在以下位置的“每个时间间隔的错误偏差”图表中：

- SOA 性能—“最关键操作”显示板
- SOA 性能—“最依赖操作”显示板
- “最关键”选项卡
- “最依赖”选项卡

仅当偏差数据是由代理报告到企业管理器，且允许在工作站中显示时，才可在显示板和选项卡中使用偏差数据。有关报告偏差数据的详细信息，请参阅[查看“偏差”选项卡中的偏差度量标准 \(p. 56\)](#)。

属性设置

可将属性设置为 true 或 false:

- True—使选定操作的数据显示在“每个时间间隔的错误偏差”图表中。
- False—禁止在“每个时间间隔的错误”图表中显示所选操作的数据。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.response.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.response.enable

使收集的选定操作的偏差数据显示在以下位置的“每个时间间隔的响应偏差”图表中：

- SOA 性能—“最关键操作”显示板
- SOA 性能—“最依赖操作”显示板
- “最关键”选项卡
- “最依赖”选项卡

仅当偏差数据是由代理报告到企业管理器，且允许在工作站中显示时，才可在显示板和选项卡中使用偏差数据。有关报告偏差数据的详细信息，请参阅[查看“偏差”选项卡中的偏差度量标准 \(p. 56\)](#)。

属性设置

可将属性设置为 true 或 false:

- True—使选定操作的数据显示在“每个时间间隔的响应偏差”图表中。
- False—禁止在“每个时间间隔的响应偏差”图表中显示选定操作的数据。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.response.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.mostcritical.enable

使“最关键操作”显示板显示在控制台中，“最关键”选项卡显示在调查器中。

属性设置

可将属性设置为 true 或 false:

- True—使“SOA 性能—最关键操作”显示板选项显示在控制台显示板菜单中，“最关键”选项卡显示在调查器中。
- False—禁止访问“SOA 性能—最关键操作”显示板和“最关键”选项卡。

如果将该属性设置为 false，则调查器中的所有节点均不显示“最关键”选项卡。“最关键操作”显示板仍会列在控制台显示板的下拉列表中，但在选中时不会显示。选择时不会显示“最关键操作”显示板，而是显示一条错误消息，表明该显示板不可用。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.mostcritical.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.mostcritical.count

确定“SOA 性能—最关键操作”显示板和“最关键”选项卡中由所有代理报告的关键操作的总数。

属性设置

最小值为 0。

最大值应小于或等于 *com.wily.introscope.soa.deviation.count.per.metric* 属性（其默认值为 25）的值。

默认

10 个操作

示例

```
com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count=5
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeEnterpriseManager.properties* 文件和 *IntroscopeWorkstation.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.mostdependent.enable

使“最依赖操作”显示板显示在控制台中以及“最依赖”选项卡显示在调查器中。

属性设置

可将属性设置为 true 或 false:

- **True**—使“SOA 性能—最依赖操作”显示板选项显示在控制台显示板菜单中，以及“最依赖”选项卡显示在调查器中。
- **False**—禁止访问“SOA 性能—最依赖操作”显示板和“最依赖”选项卡。

如果将该属性设置为 **false**，则调查器中不显示“最依赖”选项卡。“最依赖操作”显示板仍会列在控制台显示板的下拉列表中，但在选中时不会显示。不会显示“最依赖操作”显示板，但会出现一条错误消息，指示该显示板不可用。

默认

True

示例

`com.wily.introscope.soa.dashboard.typeviewer.mostdependent.enable=true`

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeEnterpriseManager.properties* 文件和 *IntroscopeWorkstation.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.mostdependent.count

确定“SOA 性能—最依赖操作”显示板和“最依赖”选项卡中由所有代理报告的最依赖操作的总数。

属性设置

最小值为 0。

最大值应小于或等于 com.wily.introscope.soa.deviation.count.per.metric 属性（其默认值为 25）的值。

默认

10 个操作

示例

```
com.wily.introscope.soa.dashboard.typeviewer.mostdependent.count=5
```

注释

可在 IntroScopeEnterpriseManager.properties 文件或 IntroScopeWorkstation.properties 文件中定义该属性。如果在 IntroScopeEnterpriseManager.properties 文件和 IntroScopeWorkstation.properties 文件中均定义了该属性，则使用 IntroScopeWorkstation.properties 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

附录 C: 特定于 SOA 的 Workstation 配置属性

通过 Workstation 配置属性，您可以控制 Workstation 的行为和操作。这些属性还允许您自定义最适合您的环境的 Workstation。使用 Workstation 属性可以专门配置 CA APM for SOA。

此部分包含以下主题：

[配置 Workstation 属性 \(p. 332\)](#)

[关于特定于 SOA 的 Workstation 属性 \(p. 333\)](#)

配置 Workstation 属性

CA APM for SOA 包括多种特定于 SOA 且带有默认设置的 Workstation 属性。但是，必须先将所有属性名称添加到 *IntroscopeWorkstation.properties* 文件，才可以配置这些属性。

将 Workstation 属性添加到 *IntroscopeWorkstation.properties*:

1. 打开 *<EM_Home>/config* 目录中的 *IntroscopeWorkstation.properties* 文件。
2. 在 [关于特定于 SOA 的 Workstation 属性](#) (p. 333)列出的属性中找到要添加到 *IntroscopeWorkstation.properties* 文件的属性。
3. 将属性的全名（包括 *com.wily.introscope* 前缀）添加到 *IntroscopeWorkstation.properties* 文件。例如：
`com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count`
4. 根据需要，设置属性值。例如，使用该属性的默认设置：
`com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count=10`
5. 保存并关闭 *IntroscopeEnterpriseManager.properties* 文件。

将 Workstation 属性添加到 *IntroscopeWorkstation.properties* 文件后，您可以根据需要配置属性值。

更改 Workstation 配置属性值:

1. 打开 *<EM_Home>/config* 目录中的 *IntroscopeWorkstation.properties* 文件。
2. 找到要更改的属性，并根据环境需要设置新值。例如：
`com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count=15`
该设置仅作为一个示例，并不是 CA APM for SOA 建议设置。
3. 保存并关闭 *IntroscopeWorkstation.properties* 文件。
4. 重新启动 Workstation，使属性更改生效。

关于特定于 SOA 的 Workstation 属性

您可以在 *IntroscopeWorkstation.properties* 文件中设置以下特定于 SOA 的属性。所有属性都以 *com.wily.introscope* 开头，但出于可读性原因，没有列出属性的该前缀。

注意：有关设置其他 Workstation 属性的信息，请参阅《CA APM 配置和管理指南》。

soa.dependencymap.ui.view.nodecount (p. 334)

指定 SOA 依赖关系视图中显示的最大视图节点数。

com.wily.introscope.soa.dependencymap.ui.view.edgecount (p. 335)

指定 SOA 依存关系视图中显示的最大视图边数。

soa.dashboard.typeviewer.average.enable (p. 323)

启用或禁用所选操作的“平均响应时间偏差”图表的显示。

soa.dashboard.typeviewer.response.enable (p. 325)

启用或禁用选定操作的“每个时间间隔的响应偏差”图表的显示。

soa.dashboard.typeviewer.errors.enable (p. 324)

启用或禁用选定操作的“每个时间间隔的错误偏差”图表的显示。

soa.dashboard.typeviewer.mostcritical.enable (p. 326)

启用或禁用“最关键操作”显示板和“最关键”选项卡的显示。

soa.dashboard.typeviewer.mostcritical.count (p. 327)

指定报告的关键操作的最大数目。

soa.dashboard.typeviewer.mostdependent.enable (p. 328)

启用或禁用“最依赖操作”显示板和“最依赖”选项卡的显示。

soa.dashboard.typeviewer.mostdependent.count (p. 329)

指定所报告的最大依赖操作数。

workstation.soa.dependencymap.fetchmetrics (p. 343)

控制度量标准在 SOA 依赖关系视图中的显示。

workstation.traceview.crossprocess.duration.full (p. 344)

确定是否为事务跟踪使用完整持续时间。

workstation.traceview.crossprocess.duration.net (p. 345)

确定是否对事务跟踪使用净持续时间。

多数情况下，必须重新启动 Workstation 才能使属性更改生效。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了某属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

soa.dependencymap.ui.view.nodecount

此属性指定显示在 Workstation、SOA 依存关系视图上的最大视图节点数。

如果选择了一个调查器节点，且 SOA 依赖关系视图中的节点数超过 `com.wily.introscope.soa.dependencymap.ui.view.nodecount` 值，将显示一条错误消息。而不会显示 SOA 依赖关系视图。

如果选择了新的上下文，且 SOA 依赖关系视图中的节点数超过 `com.wily.introscope.soa.dependencymap.ui.view.nodecount` 值，将显示一条错误消息，且 SOA 依赖关系视图将返回到前一个视图。例如，如果从“代理”的“物理”视图切换到“服务”视图的“物理”视图，且节点数超过了限制，SOA 依赖关系视图将显示一个错误消息，并返回到“代理”的“物理”视图。

如果使用 `ShowAllOperations` 或 `ShowAllServices` 展开已显示的信息，且节点数超过了计数值，SOA 依赖关系视图将显示一个错误消息，并显示最近添加的 SOA 依赖关系视图节点。

属性设置

可将属性设置为任何大于零的整数。

默认

200 个视图节点

示例

`com.wily.introscope.soa.dependencymap.ui.view.nodecount=200`

注释

您必须重新启动 Workstation，对该属性所做的更改才能生效。

com.wily.introscope.soa.dependencymap.ui.view.edgecount

此属性指定显示在 Workstation、SOA 依存关系视图上的最大视图边数。

如果选择了一个调查器节点，且 SOA 依存关系视图中的边数超过 com.wily.introscope.soa.dependencymap.ui.view.edgecount 属性值，将显示一条错误消息。而不会显示 SOA 依赖关系视图。

如果选择了新的上下文，且 SOA 依存关系视图中的边数超过 com.wily.introscope.soa.dependencymap.ui.view.edgecount 属性值，将显示一条错误消息，且 SOA 依存关系视图将返回到前一个视图，而不显示任何 SOA 依存关系视图。例如，如果从“代理”的“物理”视图切换到“服务”视图的“物理”视图，且节点边数超过了限制，SOA 依存关系视图将显示一个错误消息，并返回到“代理”的“物理”视图，不显示任何视图。

如果使用 ShowAllOperations 或 ShowAllServices 展开已显示的信息，且边数超过了计数值，SOA 依存关系视图将显示一个错误消息，并显示最近添加的 SOA 依存关系视图边。

要防止 SOA 依存关系视图影响应用程序的性能，您可以把默认值从 1000 减少到 400，来限制视图的大小和复杂性。

属性设置

可将属性设置为任何大于零的整数。

默认

1000

建议值

400

示例

com.wily.introscope.soa.dependencymap.ui.view.edgecount=250

注释

您必须重新启动 Workstation 应用程序，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.average.enable

使针对所选操作收集的偏差数据显示在以下位置的“平均响应时间偏差”图表中：

- SOA 性能—“最关键操作”显示板
- SOA 性能—“最依赖操作”显示板
- “最关键”选项卡
- “最依赖”选项卡

仅当偏差数据是由代理报告到企业管理器，且允许在工作站中显示时，才可在显示板和选项卡中使用偏差数据。有关报告偏差数据的详细信息，请参阅[查看“偏差”选项卡中的偏差度量标准 \(p. 56\)](#)。

属性设置

可将属性设置为 true 或 false：

- True—使所选操作的数据显示在“平均响应时间偏差”图表中。
- False—禁止在“平均响应时间偏差”图表中显示所选操作的数据。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.average.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果同时在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中定义该属性，将使用 *IntroscopeWorkstation.properties* 属性设置。

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.errors.enable

使收集的选定操作的偏差数据显示在以下位置的“每个时间间隔的错误偏差”图表中：

- SOA 性能—“最关键操作”显示板
- SOA 性能—“最依赖操作”显示板
- “最关键”选项卡
- “最依赖”选项卡

仅当偏差数据是由代理报告到企业管理器，且允许在工作站中显示时，才可在显示板和选项卡中使用偏差数据。有关报告偏差数据的详细信息，请参阅[查看“偏差”选项卡中的偏差度量标准 \(p. 56\)](#)。

属性设置

可将属性设置为 true 或 false:

- True—使选定操作的数据显示在“每个时间间隔的错误偏差”图表中。
- False—禁止在“每个时间间隔的错误”图表中显示所选操作的数据。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.response.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.response.enable

使收集的选定操作的偏差数据显示在以下位置的“每个时间间隔的响应偏差”图表中：

- SOA 性能—“最关键操作”显示板
- SOA 性能—“最依赖操作”显示板
- “最关键”选项卡
- “最依赖”选项卡

仅当偏差数据是由代理报告到企业管理器，且允许在工作站中显示时，才可在显示板和选项卡中使用偏差数据。有关报告偏差数据的详细信息，请参阅[查看“偏差”选项卡中的偏差度量标准 \(p. 56\)](#)。

属性设置

可将属性设置为 true 或 false:

- True—使选定操作的数据显示在“每个时间间隔的响应偏差”图表中。
- False—禁止在“每个时间间隔的响应偏差”图表中显示选定操作的数据。

默认

True

示例

`com.wily.introscope.soa.dashboard.typeviewer.response.enable=true`

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

对该属性所做的更改生效之前，您必须重新启动企业管理器。

soa.dashboard.typeviewer.mostcritical.enable

使“最关键操作”显示板显示在控制台中，“最关键”选项卡显示在调查器中。

属性设置

可将属性设置为 true 或 false:

- True—使“SOA 性能—最关键操作”显示板选项显示在控制台显示板菜单中，“最关键”选项卡显示在调查器中。
- False—禁止访问“SOA 性能—最关键操作”显示板和“最关键”选项卡。

如果将该属性设置为 false，则调查器中的所有节点均不显示“最关键”选项卡。“最关键操作”显示板仍会列在控制台显示板的下拉列表中，但在选中时不会显示。选择时不会显示“最关键操作”显示板，而是显示一条错误消息，表明该显示板不可用。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.mostcritical.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeWorkstation.properties* 文件和 *IntroscopeEnterpriseManager.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.mostcritical.count

确定“SOA 性能—最关键操作”显示板和“最关键”选项卡中由所有代理报告的关键操作的总数。

属性设置

最小值为 0。

最大值应小于或等于 *com.wily.introscope.soa.deviation.count.per.metric* 属性（其默认值为 25）的值。

默认

10 个操作

示例

`com.wily.introscope.soa.dashboard.typeviewer.mostcritical.count=5`

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeEnterpriseManager.properties* 文件和 *IntroscopeWorkstation.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.mostdependent.enable

使“最依赖操作”显示板显示在控制台中以及“最依赖”选项卡显示在调查器中。

属性设置

可将属性设置为 true 或 false:

- True—使“SOA 性能—最依赖操作”显示板选项显示在控制台显示板菜单中，以及“最依赖”选项卡显示在调查器中。
- False—禁止访问“SOA 性能—最依赖操作”显示板和“最依赖”选项卡。

如果将该属性设置为 false，则调查器中不显示“最依赖”选项卡。“最依赖操作”显示板仍会列在控制台显示板的下拉列表中，但在选中时不会显示。不会显示“最依赖操作”显示板，但会出现一条错误消息，指示该显示板不可用。

默认

True

示例

```
com.wily.introscope.soa.dashboard.typeviewer.mostdependent.enable=true
```

注释

可在 *IntroscopeEnterpriseManager.properties* 文件或 *IntroscopeWorkstation.properties* 文件中定义该属性。如果在 *IntroscopeEnterpriseManager.properties* 文件和 *IntroscopeWorkstation.properties* 文件中均定义了该属性，则使用 *IntroscopeWorkstation.properties* 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

soa.dashboard.typeviewer.mostdependent.count

确定“SOA 性能—最依赖操作”显示板和“最依赖”选项卡中由所有代理报告的最依赖操作的总数。

属性设置

最小值为 0。

最大值应小于或等于 com.wily.introscope.soa.deviation.count.per.metric 属性（其默认值为 25）的值。

默认

10 个操作

示例

```
com.wily.introscope.soa.dashboard.typeviewer.mostdependent.count=5
```

注释

可在 IntroScopeEnterpriseManager.properties 文件或 IntroScopeWorkstation.properties 文件中定义该属性。如果在 IntroScopeEnterpriseManager.properties 文件和 IntroScopeWorkstation.properties 文件中均定义了该属性，则使用 IntroScopeWorkstation.properties 属性设置。

您必须重新启动企业管理器和 Workstation，对该属性所做的更改才能生效。

workstation.soa.dependencymap.fetchmetrics

控制度量标准在 SOA 依赖关系视图中的显示。

属性设置

可将属性设置为 true 或 false:

- True—请求从企业管理器中选择的主要度量标准和“工具提示”度量标准。如果将该属性设置为 true, SOA 依赖关系地图将在图中的每个节点上显示主要度量标准。当悬停在图中的某个节点上时，“工具提示”中会显示其他度量标准。
- False—不在 SOA 依赖关系地图节点中显示主要度量标准，也不在“工具提示”窗口中显示其他度量标准。

默认

True

示例

```
com.wily.introscope.workstation.soa.dependencymap.fetchmetrics=true
```

注释

您必须重新启动 Workstation，对该属性所做的更改才能生效。

workstation.traceview.crossprocess.duration.full

确定是否在事务跟踪查看器的“顺序视图”选项卡中显示事务跟踪的完整持续时间，并将其用作跟踪的默认值。

完整持续时间是根据流程的起止时间来计算的，流程的起止时间包括该流程等候其他流程执行所花费的时间。使用该方法时，根跟踪始终是同步事务中最慢的。通过从总体持续时间中减去跟踪的异步子项的持续时间，可计算出净持续时间。

如果不更改工作站属性文件，则“顺序视图”选项卡允许您选择完整持续时间或净持续时间（默认情况下显示完整持续时间）。

属性设置

如果仅将 *introscope.workstation.traceview.crossprocess.duration.full* 属性添加到工作站属性文件中，“顺序视图”选项卡中将仅显示完整持续时间。

如果将 *introscope.workstation.traceview.crossprocess.duration.full* 和 *introscope.workstation.traceview.crossprocess.duration.net* 添加到工作站属性文件中，“顺序视图”选项卡中将显示完整持续时间和净持续时间，默认值由具有最低值的属性决定。

例如，如果要允许选择完整持续时间或净持续时间，但希望在默认情况下显示净持续时间，可在工作站属性文件中按如下方式设置属性：

```
com.wily.introscope.workstation.traceview.crossprocess.duration.full=2  
com.wily.introscope.workstation.traceview.crossprocess.duration.net=1
```

默认

1 (显示完整持续时间)

示例

```
com.wily.introscope.workstation.traceview.crossprocess.duration.full=1
```

注释

您必须重新启动 Workstation，对该属性所做的更改才能生效。

workstation.traceview.crossprocess.duration.net

确定是否将事务跟踪的净持续时间显示并用作事务跟踪查看器中“顺序视图”选项卡上跟踪的默认值。

通过从总体持续时间中减去跟踪的异步子项的持续时间，可计算出净持续时间。该方法的主要优势在于同步事务的最慢线程是显而易见的。

如果不更改工作站属性文件，则“顺序视图”选项卡允许您选择完整持续时间或净持续时间（默认情况下显示完整持续时间）。

属性设置

如果仅将 *introscope.workstation.traceview.crossprocess.duration.net* 属性添加到工作站属性文件中，则“顺序视图”选项卡上仅显示净持续时间。

如果将 *introscope.workstation.traceview.crossprocess.duration.full* 和 *introscope.workstation.traceview.crossprocess.duration.net* 添加到工作站属性文件中，“顺序视图”选项卡中将显示完整持续时间和净持续时间，默认值由具有最低值的属性决定。

例如，如果要允许选择完整持续时间或净持续时间，但希望在默认情况下显示净持续时间，可在工作站属性文件中按如下方式设置属性：

```
com.wily.introscope.workstation.traceview.crossprocess.duration.full=2  
com.wily.introscope.workstation.traceview.crossprocess.duration.net=1
```

默认

2（净持续时间不作为默认值显示）

示例

```
com.wily.introscope.workstation.traceview.crossprocess.duration.net=1
```

注释

您必须重新启动 Workstation，对该属性所做的更改才能生效。

附录 D：特定于 SOA 的 WebView 配置属性

通过 WebView 配置属性，您可以控制 WebView 应用程序的行为和操作。这些属性还允许您自定义最适合您的环境的 WebView。使用 WebView 属性可以专门配置 CA APM for SOA。

此部分包含以下主题：

[配置 WebView 属性 \(p. 347\)](#)

[关于特定于 SOA 的 WebView 属性 \(p. 348\)](#)

配置 WebView 属性

CA APM for SOA 包括特定于 SOA 且带有默认设置的 WebView 属性。在您开始配置之前，将属性名添加到 *IntroscopeWebView.properties* 文件中。

将 WebView 属性添加到 *IntroscopeWebView.properties*:

1. 打开 *<EM_Home>/config* 目录中的 *IntroscopeWebView.properties* 文件。
2. 查找想要添加到 *IntroscopeWebView.properties* 文件的 [SOA 特定 WebView 属性 \(p. 348\)](#)。
3. 将属性的全名（包括 *com.wily.introscope* 前缀）添加到 *IntroscopeWebView.properties* 文件。例如：
`com.wily.introscope.soa.dependencymap.ui.view.edgecount`
4. 根据需要，设置属性值。例如，使用该属性的默认设置：
`com.wily.introscope.soa.dependencymap.ui.view.edgecount=1000`
5. 保存并关闭 *IntroscopeWebView.properties* 文件。

将 WebView 属性添加到 *IntroscopeWebView.properties* 文件后，您可以根据需要配置属性值。

更改 WebView 配置属性值:

1. 打开 *<EM_Home>/config* 目录中的 *IntroscopeWebView.properties* 文件。
2. 找到要更改的属性，并根据环境需要设置新值。例如：
`com.wily.introscope.soa.dependencymap.ui.view.edgecount=250`
该设置仅作为一个示例，并不是 CA APM for SOA 建议设置。
3. 保存并关闭 *IntroscopeWebView.properties* 文件。
4. 重新启动 WebView 应用程序，使属性更改生效。

关于特定于 SOA 的 WebView 属性

您可以在 `IntroscopeWebView.properties` 文件中设置以下特定于 SOA 的属性。所有属性都以 `com.wily.introscope` 开头，但出于可读性原因，没有列出属性的该前缀。

注意：有关设置其他 WebView 属性的信息，请参阅《CA APM 配置和管理指南》。

[soa.dependencymap.ui.view.nodecount \(p. 348\)](#)

指定显示在 WebView、SOA 依存关系视图上的最大视图节点数。

[com.wily.introscope.soa.dependencymap.ui.view.edgecount \(p. 349\)](#)

指定显示在 WebView、SOA 依存关系视图上的最大视图边数。

[soa.dependencymap.ui.view.nodecount](#)

此属性指定显示在 WebView、SOA 依存关系视图上的最大视图节点数。

如果选择了一个调查器节点，且 SOA 依存关系视图中的节点数超过 `com.wily.introscope.soa.dependencymap.ui.view.nodecount` 属性值，将显示一条错误消息。而不会显示 SOA 依赖关系视图。

如果选择了新的上下文，且 SOA 依存关系视图中的节点数超过 `com.wily.introscope.soa.dependencymap.ui.view.nodecount` 属性值，将显示一条错误消息，且不显示 SOA 依存关系视图。例如，如果从“代理”的“物理”视图切换到“服务”视图的“物理”视图，且节点数超过了限制，SOA 依存关系视图将显示一个错误消息，且不显示任何视图。

如果使用 `ShowAllOperations` 或 `ShowAllServices` 展开已显示的信息，且节点数超过了计数值，SOA 依赖关系视图将显示一个错误消息，并显示最近添加的 SOA 依赖关系视图节点。

要防止 SOA 依存关系视图影响应用程序的性能，您可以把默认属性值从 200 减少为更低值，从而限制视图的大小和复杂性。

属性设置

可将属性设置为任何大于零的整数。

默认

200 个视图节点

建议值

200 个视图节点

示例

```
com.wily.introscope.soa.dependencymap.ui.view.nodecount=200
```

注释

您必须重新启动 WebView 应用程序，对该属性所做的更改才能生效。

com.wily.introscope.soa.dependencymap.ui.view.edgecount

此属性指定显示在 WebView、SOA 依存关系视图上的最大视图边数。

如果选择了一个调查器节点，且 SOA 依存关系视图中的边数超过 com.wily.introscope.soa.dependencymap.ui.view.edgecount 值，将显示一条错误消息。而不会显示 SOA 依赖关系视图。

如果选择了新的上下文，且 SOA 依存关系视图中的边数超过 com.wily.introscope.soa.dependencymap.ui.view.edgecount 值，将显示一条错误消息，且不显示 SOA 依存关系视图。例如，如果从“代理”的“物理”视图切换到“服务”视图的“物理”视图，且边数超过了限制，SOA 依存关系视图将显示一个错误消息，且不显示任何视图。

如果使用 ShowAllOperations 或 ShowAllServices 展开已显示的信息，且边数超过了计数值，SOA 依存关系视图将显示一个错误消息，并显示最近添加的 SOA 依存关系视图边。

要防止 SOA 依存关系视图影响应用程序的性能，您可以把默认值从 1000 减少到 250，来限制视图的大小和复杂性。

属性设置

可将属性设置为任何大于零的整数。

默认

1000 个视图边

建议值

250 个视图边

示例

```
com.wily.introscope.soa.dependencymap.ui.view.edgecount=250
```

注释

您必须重新启动 WebView 应用程序，对该属性所做的更改才能生效。

