

CA Application Performance Management

for Oracle WebLogic Server 안내서
릴리스 9.5



포함된 도움말 시스템 및 전자적으로 배포된 매체를 포함하는 이 문서(이하 "문서")는 정보 제공의 목적으로만 제공되며 CA 에 의해 언제든지 변경 또는 취소될 수 있습니다.

CA 의 사전 서면 동의 없이 본건 문서의 전체 혹은 일부를 복사, 전송, 재생, 공개, 수정 또는 복제할 수 없습니다. 이 문서는 CA 의 기밀 및 독점 정보이며, 귀하는 이 문서를 공개하거나 다음에 의해 허용된 경우를 제외한 다른 용도로 사용할 수 없습니다: (i) 귀하가 이 문서와 관련된 CA 소프트웨어를 사용함에 있어 귀하와 CA 사이에 별도 동의가 있는 경우, 또는 (ii) 귀하와 CA 사이에 별도 기밀 유지 동의가 있는 경우.

상기 사항에도 불구하고, 본건 문서에 기술된 라이선스가 있는 사용자는 귀하 및 귀하 직원들의 해당 소프트웨어와 관련된 내부적인 사용을 위해 합당한 수의 문서 복사본을 인쇄 또는 제작할 수 있습니다. 단, 이 경우 각 복사본에는 전체 CA 저작권 정보와 범례가 첨부되어야 합니다.

본건 문서의 사본 인쇄 또는 제작 권한은 해당 소프트웨어의 라이선스가 전체 효력을 가지고 유효한 상태를 유지하는 기간으로 제한됩니다. 어떤 사유로 인해 라이선스가 종료되는 경우, 귀하는 서면으로 문서의 전체 또는 일부 복사본이 CA 에 반환되거나 파괴되었음을 입증할 책임이 있습니다.

CA 는 관련법의 허용 범위 내에서, 상품성에 대한 묵시적 보증, 특정 목적에 대한 적합성 또는 권리 위반 보호를 비롯하여(이에 제한되지 않음) 어떤 종류의 보증 없이 본 문서를 "있는 그대로" 제공합니다. CA 는 본 시스템의 사용으로 인해 발생하는 직, 간접 손실이나 손해(수익의 손실, 사업 중단, 영업권 또는 데이터 손실 포함)에 대해서는 (상기 손실이나 손해에 대해 사전에 명시적으로 통지를 받은 경우라 하더라도) 귀하나 제 3 자에게 책임을 지지 않습니다.

본건 문서에 언급된 모든 소프트웨어 제품의 사용 조건은 해당 라이선스 계약을 따르며 어떠한 경우에도 이 문서에서 언급된 조건에 의해 라이선스 계약이 수정되지 않습니다.

본 문서는 CA 에서 제작되었습니다.

본 시스템은 "제한적 권리"와 함께 제공됩니다. 미합중국 정부에 의한 사용, 복제 또는 공개는 연방조달규정(FAR) 제 12.212 조, 제 52.227-14 조, 제 52.227-19(c)(1)호 - 제(2)호 및 국방연방구매규정(DFARS) 제 252.227-7014(b)(3)호 또는 해당하는 경우 후속 조항에 명시된 제한 사항을 따릅니다.

Copyright © 2013 CA. All rights reserved. 본 시스템에서 언급된 모든 상표, 상호, 서비스 표시 및 로고는 각 해당 회사의 소유입니다.

CA Technologies 제품 참조

이 문서에서는 다음과 같은 CA Technologies 제품과 기능을 참조합니다.

- CA Application Performance Management(CA APM)
- CA Application Performance Management ChangeDetector(CA APM ChangeDetector)
- CA Application Performance Management ErrorDetector([assign the value for wed in your book])
- 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)

CA 에 문의

기술 지원팀에 문의

온라인 기술 지원 및 지사 목록, 기본 서비스 시간, 전화 번호에 대해서는 <http://www.ca.com/worldwide>에서 기술 지원팀에 문의하십시오.

목차

제 1 장: 소개 9

소개.....	9
안내서 사용.....	9

제 2 장: 확장 설치 및 구성 11

시작하기 전에 유의할 사항.....	11
CA APM for Oracle WebLogic Server 설치 방법.....	12
설치 관리자 실행.....	12
에이전트 설치 관리자 실행.....	13
관리 모듈 설정.....	14
Workstation 확장 설정.....	15
JMX 메트릭 수집을 사용하도록 설정.....	15
WebLogic Server 에서 시작 클래스 구성.....	16
메트릭 만료 처리에서 증가 카운터 제외.....	17
선택 사항: CA APM ChangeDetector 통합.....	19
확장 제거.....	21

제 3 장: 확장 사용 23

Workstation 에서 WebLogic Server 메트릭 보기.....	23
특수 요소 보기.....	24
WebLogic Server 대시보드 보기.....	25
WebLogic Application Server - 개요.....	25
WebLogic - JDBC 연결 풀.....	26
WebLogic - HTTP 세션.....	26
WebLogic - EJB 하위 시스템.....	27
WebLogic - JMS 하위 시스템.....	27
WebLogic - JTA 하위 시스템.....	28
WebLogic - 보안.....	28

부록 A: WebLogic Server 메트릭 31

EJB 하위 시스템.....	31
상대 저장 메트릭.....	32
NRU 캐시 메트릭.....	32

LRU 캐시 메트릭	33
상태 비저장 메트릭	33
엔터티 캐시 메트릭	34
Message-Driven 메트릭	34
엔터티 메트릭	34
서블릿 하위 시스템 메트릭	35
HTTP 세션	35
JMS 하위 시스템	36
분산 대상 메트릭	37
Security Subsystem(보안 하위 시스템)	38
Clustering(클러스터링)	39
XML Subsystem(XML 하위 시스템)	40
JMX 메트릭	41
JMX 집계 메트릭	45

제 1 장: 소개

이 섹션은 다음 항목을 포함하고 있습니다.

[소개](#) (페이지 9)

[안내서 사용](#) (페이지 9)

소개

CA APM for Oracle WebLogic Server 는 Oracle Corporation 과 협력하여 개발한 CA APM 확장입니다. 이 확장은 프로덕션 WebLogic Server 환경을 위한 고급 성능 관리 기능을 제공합니다.

참고: 이 안내서에서 CA APM for Oracle WebLogic Server 를 단순히 "확장"이라고 하는 경우도 있습니다. 이전에는 이 제품을 "PowerPack"이라고 했습니다.

CA APM for Oracle WebLogic Server 를 사용하면 중요 WebLogic 리소스를 모니터링할 수 있으며 응용 프로그램 병목 지점을 찾고 WebLogic Server 및 응용 프로그램의 가용성을 향상시킬 수 있습니다.

안내서 사용

이 안내서에는 CA APM for Oracle WebLogic Server 설정 및 사용 관련 지침이 나와 있습니다.

참고: 시스템 요구 사항에 대해서는 *Compatibility Guide*(호환성 안내서)를 참조하십시오.

이 안내서에서는 다음과 같은 상위 수준의 정보를 제공합니다.

- [CA APM for Oracle WebLogic Server 설치 및 구성](#) (페이지 11)
- [CA APM for Oracle WebLogic Server 사용](#) (페이지 23)
- [WebLogic Server 메트릭](#) (페이지 31)

제 2 장: 확장 설치 및 구성

이 장에서는 CA APM 배포에서 CA APM for Oracle WebLogic Server 를 설치 및 구성하는 방법을 설명합니다.

이 섹션은 다음 항목을 포함하고 있습니다.

[시작하기 전에 유의할 사항](#) (페이지 11)

[CA APM for Oracle WebLogic Server 설치 방법](#) (페이지 12)

[확장 제거](#) (페이지 21)

시작하기 전에 유의할 사항

CA APM for Oracle WebLogic Server 를 설치하기 전에 다음 필수 단계를 수행하십시오.

다음 단계를 수행하십시오.

1. 지원되는 버전의 WebLogic Server 가 환경에 설치되어 있는지 확인하십시오.

참고: 지원되는 버전의 전체 목록은 *CA APM 호환성 안내서*를 참조하십시오. *CA APM 호환성 안내서*에는 CA APM 구성 요소의 소프트웨어 및 하드웨어 호환성이 나와 있습니다. 이 안내서는 CA Support 에서 사용할 수 있습니다.

2. CA Introscope® 환경에서 다음 디렉터리 위치를 확인합니다.
 - **응용 프로그램 서버 홈 디렉터리** - 응용 프로그램 서버의 홈 디렉터리로, 지금부터 <AppServer_Home>으로 표시합니다.
 - **APM 홈 디렉터리** - Enterprise Manager 서버에서 APM 의 설치 디렉터리로, 지금부터 <APM_Home>으로 표시합니다.
 - **APM 에이전트 디렉터리** - 응용 프로그램 서버에서 APM 에이전트의 설치 디렉터리로, 지금부터 <Agent_Home>으로 표시합니다.

3. 지원되는 버전의 JVM(Java 가상 컴퓨터)이 WebLogic Server 환경에 설치되어 있는지 확인하십시오.

참고: CPU 사용량을 줄려면 캐시 연결 속성을 다음과 같이 설정하십시오.

```
introscope.agent.sqlagent.cacheConnectionsURLs=true
```

4. CA APM for Oracle WebLogic Server 를 설치하려는 응용 프로그램 서버 인스턴스를 종료합니다.

중요! 설치 지침에서는 JVM AutoProbe 를 사용하여 응용 프로그램을 계측한다고 가정합니다. JVM AutoProbe 사용에 대한 자세한 내용은 *CA APM Java Agent 구현 안내서* 또는 *CA APM .NET 에이전트 구현 안내서*를 참조하십시오.

CA APM for Oracle WebLogic Server 설치 방법

CA APM for Oracle WebLogic Server 를 설치 및 구성하기 위한 단계는 다음과 같습니다.

1. [설치 관리자를 실행합니다](#) (페이지 12).
2. [에이전트 설치 관리자를 실행합니다](#) (페이지 13).
3. [관리 모듈을 설정합니다](#) (페이지 14).
4. [Workstation 확장을 설정합니다](#) (페이지 15).
5. [JMX 메트릭 수집을 사용하도록 설정합니다](#) (페이지 15).
6. [WebLogic Server 에서 시작 클래스를 구성합니다](#) (페이지 16).
7. [메트릭 만료 처리에서 증가 카운터를 제외합니다](#) (페이지 17).
8. [\(선택 사항\) ChangeDetector 를 통합합니다](#) (페이지 19).

설치 관리자 실행

CA APM 설치 관리자를 실행합니다.

CA APM 설치 관리자가 `<APM_Home>\examples\<Extension_Home>` 디렉터리에 파일을 넣습니다. `\examples` 디렉터리에 구성된 방식에 따라 파일을 올바른 위치로 복사합니다.

CA APM 설치 관리자 실행에 대한 자세한 내용은 *CA APM 설치 및 업그레이드 안내서*를 참조하십시오.

참고: 다음 목록에서는 Microsoft Windows 디렉터리 구문을 사용합니다. UNIX 등 다른 설치 플랫폼의 경우 "\" 문자를 "/" 문자로 대체하십시오.

설치 관리자가 <APM_Home>\examples\<Extension_Home>\ 디렉터리에 다음 파일을 넣습니다.

.\config\modules\PPWebLogicManagementModule.jar

WebLogic 관련 메트릭을 볼 수 있는 대시보드 및 경고가 미리 구성되어 있는 관리 모듈입니다.

.\ext\PPWebLogicExtensionPlugins.jar

JMX 집계 메트릭을 사용되도록 설정하는 Enterprise Manager 확장입니다.

.\ext\ddtv\PPWebLogicTypeview.xml

Investigator 에서 AppServer 뷰를 사용되도록 설정하는 Workstation 확장입니다.

에이전트 설치 관리자 실행

확장에 필요한 에이전트 정보를 설치 및 최초 구성하려면 에이전트 설치 관리자를 실행하십시오.

자세한 내용은 *CA APM Java Agent 구현 안내서* 또는 *CA APM .NET 에이전트 구현 안내서*를 참조하십시오.를 참조하십시오.

참고: 다음 목록에서는 Microsoft Windows 디렉터리 구문을 사용합니다. UNIX 등 다른 설치 플랫폼의 경우 "\" 문자를 "/" 문자로 대체하십시오.

에이전트 설치 관리자가 응용 프로그램 서버 컴퓨터의 <Agent_Home>\wily 디렉터리에 다음 파일을 넣습니다.

ppweblogic.pbd

WebLogic Server 용 응용 프로그램을 계측하는 데 필요한 추적 프로그램이 들어 있는 ProbeBuilder 지시문 파일입니다. 이 파일은 <Agent_Home>/wily/core/config 디렉터리에 있습니다.

PPWebLogicJMXFilterString.txt

JMX 필터 문자열이 들어 있는 텍스트 파일입니다. 이 파일은 `<Agent_Home>/common` 디렉터리에 있습니다.

ChangeDetector-config-WebLogic.xml

ChangeDetector 구성 파일입니다. 이 파일은 `<Agent_Home>/common` 디렉터리에 있습니다.

설치 관리자가 에이전트 및 확장 에이전트 파일에서 Enterprise Manager 호스트 및 포트 정보를 구성합니다.

에이전트 설치 중에 확장을 사용되도록 설정한 경우:

- 설치 관리자가 `ppweblogic.pbd` 파일을 `<Agent_Home>/wily/core/config` 또는 `JVM/wily` 디렉터리에 설치합니다. 설치 위치는 Java 응용 프로그램을 AutoProbe 기반 웹 응용 프로그램 서버를 사용하여 계측하는지 아니면 JVM 을 사용하여 계측하는지에 따라 달라집니다.
- 설치 관리자가 `IntroscopeAgent.profile` 파일(WebLogic Server 호스트의 `<Agent_Home>/wily/core/config` 디렉터리에 있음)에서 `introscope.autoprobe.directivesFile` 속성을 수정하여 새 지시문 파일 `ppweblogic.pbd` 를 목록에 추가합니다. 예:
`introscope.autoprobe.directivesFile=weblogic-full.pbd,ppweblogic.pbd`

관리 모듈 설정

CA APM for Oracle WebLogic Server 대시보드를 표시하도록 관리 모듈을 설정해야 합니다.

다음 단계를 수행하십시오.

1. 관리 모듈 파일 `PPWebLogicManagementModule.jar` 를 `<APM_Home>/config/modules` 디렉터리로 복사합니다.
2. Enterprise Manager 를 다시 시작합니다.

CA APM 은 CA APM for Oracle WebLogic Server 대시보드를 표시할 수 있습니다.

Workstation 확장 설정

Workstation 에서 CA APM for Oracle WebLogic Server 데이터를 표시하도록 Workstation 확장을 설정합니다.

다음 단계를 수행하십시오.

1. PPWebLogicTypeview.xml 을 <APM_Home>/ext/ddtv 디렉터리로 복사합니다.
2. PPWebLogicExtensionPlugins.jar 를 <APM_Home>/ext 디렉터리로 복사합니다.
3. Enterprise Manager 를 다시 시작합니다.

JMX 메트릭 수집을 사용하도록 설정

JMX 메트릭 수집이 사용되도록 확장을 구성할 수 있습니다.

다음 단계를 수행하십시오.

1. IntroscopeAgent.profile 파일을 열고 JMX Configuration 머리글 아래에서 다음 속성을 찾습니다.
 - a. introscope.agent.jmx.enable -- 이 속성의 주석 처리를 제거하고 값을 true 로 설정합니다.
 - b. introscope.agent.jmx.name.primarykeys -- 이 속성을 주석 처리합니다.
 - c. introscope.agent.jmx.name.filter -- 이 속성의 주석 처리를 제거합니다.
2. 기본값에 지정한 것보다 더 많은 메트릭을 수집하려면 PPWebLogicJMXFilterString.txt 파일에 지정되어 있는 문자열을 추가합니다. 이 파일은 <Agent_Home>\wily\common 디렉터리에 있습니다(introscope.agent.jmx.name.filter 속성에 추가).

참고: 권장 필터 문자열 값을 사용할 경우 WebLogic Server 에 대한 주요 JMX 메트릭이 선택됩니다. 다른 JMX 메트릭을 수집하려면 메트릭을 권장 필터 값에 쉼표로 구분된 형식으로 추가하십시오.

중요! 문자열은 대/소문자를 구분합니다.

3. *IntroscopeAgent.profile* 파일을 저장하고 응용 프로그램을 다시 시작합니다.

참고: JMX 메트릭은 WebLogic Server 관리 콘솔의 WLDF 콘솔 확장에 있는 메트릭과 일치합니다.

WebLogic Server 에서 시작 클래스 구성

JMX 데이터를 가져오려면 WebLogic Server 에 CA Introscope®에 대한 시작 클래스가 구성되어 있어야 합니다. 이미 WebLogic Server 에서 APM 시작 클래스를 구성한 경우 이 단계를 건너뛸 수 있습니다.

다음 단계를 수행하십시오.

1. "WebLogic Administration Console"(WebLogic 관리 콘솔)에서 "Lock & Edit"(잠금 및 편집) 단추를 클릭합니다.
2. 구성하려는 서버 아래에서 "Environment"(환경) 노드를 확장하고 "Startup & Shutdown"(시작 및 종료) 링크를 클릭합니다.
기존의 시작 및 종료 클래스가 있는 테이블이 표시됩니다.
3. 테이블에서 "새로 만들기" 단추를 클릭합니다.
"Configure a New Startup or Shutdown Class"(새 시작 또는 종료 클래스 구성) 대화 상자가 나타납니다.
4. "Class Type"(클래스 유형) 아래에서 "Startup Class"(시작 클래스) 옵션을 클릭하고 "다음"을 클릭합니다.
5. "Name"(이름) 필드에 다음을 입력합니다.
Introscope Startup Class
6. "ClassName"(클래스 이름) 필드에 다음을 입력합니다.
com.wily.introscope.api.weblogic.IntroscopeStartupClass
7. "다음"을 클릭합니다.
8. "Select Targets"(대상 선택) 아래에서 다음을 수행하십시오.
 - 시작 클래스를 사용할 서버에 대한 상자를 선택합니다.
 - 응용 프로그램 서버 인스턴스가 클러스터된 경우 클러스터에 대한 상자를 선택합니다.

9. "마침"을 클릭하십시오.
10. "Activate Changes"(변경 내용 활성화)를 클릭하고 응용 프로그램 서버를 다시 시작합니다.

WebAppSupport.jar 에서 시작 클래스가 구성되고 구현됩니다.

중요! 응용 프로그램 시작 클래스 경로에 WebAppSupport.jar 의 위치를 추가했는지 확인하십시오.

메트릭 만료 처리에서 증가 카운터 제외

에이전트 연결이 끊기면 메트릭이 회색으로 나타나거나 흐리게 나타날 수 있습니다. 증가 및 감소 카운터에 대한 메트릭은 메트릭 만료 처리에서 고려되지 않아야 합니다.

예: 메트릭 만료 처리

초기 세션 수가 5 라고 가정합니다. 이 메트릭이 메트릭 만료 처리에 포함되어 있고 지정된 기간 동안 데이터를 수신하지 못하면 메트릭이 해제, 즉 사용하지 않도록 설정됩니다. 카운터는 0 으로 다시 초기화됩니다. 일정 시간이 지난 후 이 메트릭이 데이터를 수신하면 카운터가 0 부터 시작하게 되어 메트릭이 잘못된 데이터를 표시하게 됩니다.

따라서 증가 및 감소 카운터를 포함하는 메트릭이 잘못된 정보를 제공할 수 있으므로 메트릭 만료 처리에 포함되지 않아야 합니다.

다음 단계를 수행하십시오.

1. <AgentHome>\wily\core\config 로 이동하여 IntroscopeAgent.profile 파일을 엽니다.
2. introscope.agent.metricAging.metricExclude.ignore 속성의 주석 처리를 제거하여 증가 및 감소 카운터와 관련된 메트릭을 만료 처리에서 제외합니다.

```
introscope.agent.metricAging.metricExclude.ignore.<X+1>=WebLogic|Servlet
Subsystem:Error Response Count
```

```
introscope.agent.metricAging.metricExclude.ignore.<X+2>=WebLogic|HTTP
Sessions Subsystem|All Sessions:Session Count
```

```
introscope.agent.metricAging.metricExclude.ignore.<X+3>=WebLogic|HTTP
Sessions Subsystem|Cookie Sessions:Session Count
```

```

introscope.agent.metricAging.metricExclude.ignore.<X+4>=WebLogic|HTTP
Sessions Subsystem|File Sessions:Session Count
introscope.agent.metricAging.metricExclude.ignore.<X+5>=WebLogic|HTTP
Sessions Subsystem|JDBC Sessions:Session Count
introscope.agent.metricAging.metricExclude.ignore.<X+6>=WebLogic|HTTP
Sessions Subsystem|Memory Sessions:Session Count
introscope.agent.metricAging.metricExclude.ignore.<X+7>=WebLogic|HTTP
Sessions Subsystem|Replicated Sessions:Session Count
introscope.agent.metricAging.metricExclude.ignore.<X+8>=WebLogic|Clustering|C
hange Event:Count
introscope.agent.metricAging.metricExclude.ignore.<X+9>=WebLogic|Clustering|F
ull State Dump:Count
introscope.agent.metricAging.metricExclude.ignore.<X+10>=WebLogic|Clustering|
Announce:Count
introscope.agent.metricAging.metricExclude.ignore.<X+11>=WebLogic|Clustering|
Peer Gone Listeners:Count
introscope.agent.metricAging.metricExclude.ignore.<X+12>=WebLogic|Clustering|
RJVM Remote Call:Error Count
introscope.agent.metricAging.metricExclude.ignore.<X13>=WebLogic|XML
Subsystem|SAX Parsers:Creation Count
introscope.agent.metricAging.metricExclude.ignore.<X+14>=WebLogic|XML
Subsystem|Document Builder:Creation Count
introscope.agent.metricAging.metricExclude.ignore.<X+15>=WebLogic|XML
Subsystem|SAX Transformer:Creation Count

```

참고: 이 목록을 사용하여 IntroscopeAgent.profile 파일의 메트릭 만료 처리 제외 목록에 메트릭을 추가할 수 있습니다. 그렇게 할 때는 줄 바꿈을 제거하십시오. 줄 바꿈은 단지 가독성을 위해 포함된 것입니다.

이 목록에서 X 는 IntroscopeAgent.profile 파일의

introscope.agent.metricAging.metricExclude.ignore 속성에 대한 기준 시퀀스 번호입니다. 예를 들어, IntroscopeAgent.profile 파일에서 introscope.agent.metricAging.metricExclude.ignore 속성에 대한 마지막 시퀀스 번호가

introscope.agent.metricAging.metricExclude.ignore.15=Thread a 인 경우 X=15 입니다.

선택 사항: CA APM ChangeDetector 통합

CA APM ChangeDetector 를 CA APM for Oracle WebLogic Server 와 통합하려면 다음 단계를 수행하십시오.

ChangeDetector 구성 파일을 설치한 후 CA APM ChangeDetector 를 사용하여 WebLogic Server 의 구성된 폴더에서 발생하는 변경 내용을 모니터링하십시오.

참고: WebLogic Server 를 시작하면 ChangeDetector 가 감지하는 모든 파일이 초기 버전으로 표시됩니다. 따라서 표시되는 초기 변경 내용은 실제 변경 내용이 아니라 모니터링되는 폴더에 있는 모든 파일의 초기 버전이 추가된 것을 나타냅니다. 이는 설계된 동작입니다.

다음 단계를 수행하십시오.

1. <Agent_Home>/wily/common 디렉터리로 이동합니다.
2. Java 응용 프로그램에 대한 메트릭이 필요한지 아니면 JVM 에 대한 메트릭이 필요한지에 따라 ChangeDetector-config-WebLogic.xml 을 <AppServer_Home>/wily 또는 JVM/wily 디렉터리에 복사합니다.
3. WebLogic Server 컴퓨터에서 IntroscopeAgent.profile 파일을 열고 다음 속성 값을 설정합니다. ChangeDetector 구성 파일의 위치를 지정합니다.

```
introscope.changeDetector.profile=<path to
AppServerHome/wily>\ChangeDetector-config-WebLogic.xml
```

Java 환경의 예:

```
introscope.changeDetector.profile=C:\\bea\\wlserver_10.3\\wily\\ChangeDetector-config-WebLogic.xml.
```

UNIX 환경의 예:

```
introscope.changeDetector.profile=/usr/bin/bea/wlserver_10.3/wily/ChangeDetector-config-WebLogic.xml
```

참고: 도메인이 여러 개 있고 모든 도메인에 ChangeDetector 를 구성하려는 경우 IntroscopeAgent.profile 에서 다음 속성이 사용되도록 설정하는 것이 좋습니다.

```
introscope.changeDetector.agentID=<SampleApplicationName>
```

이 속성이 사용되도록 설정하지 않으면 Investigator 에서 ChangeDetector 에이전트 ID 가 충돌하여 에이전트 로그 파일에 오류가 기록될 수 있습니다.

4. IntroscopeAgent.profile 파일을 저장합니다.

5. 다음 단계 중 하나를 수행하여 ChangeDetector에 맞게 환경을 구성합니다.
 - [시작 스크립트를 수정합니다](#) (페이지 20)(권장 사항).
 - [에이전트 프로필을 수정합니다](#) (페이지 20).
- 이제 ChangeDetector가 성공적으로 통합되었습니다.

시작 스크립트 수정

시작 스크립트를 수정하여 ChangeDetector에 맞게 환경을 구성하십시오.

중요! 여러 에이전트에 동일한 에이전트 프로필을 사용하려면 에이전트 프로필 대신 시작 스크립트를 수정하십시오. 이렇게 하지 않으면 동일한 에이전트 프로필을 사용하는 도메인 간에 충돌이 발생할 수 있습니다.

다음 단계를 수행하십시오.

1. CA APM for Oracle WebLogic Server로 모니터링되는 WebLogic 응용 프로그램을 시작하는 .cmd 또는 .bat 파일에 다음 행을 추가합니다.

```
-DDOMAIN_HOME=<filepath1> -DAPPLICATION_HOME=<filepath2>
```

여기서 <filepath1>과 <filepath2>는 각각 도메인 또는 응용 프로그램 홈의 파일 경로입니다.

2. .cmd 또는 .bat 파일을 저장합니다.
3. 응용 프로그램을 다시 시작합니다.

참고: DOMAIN_HOME 및 APPLICATION_HOME 속성은 CA APM for Oracle WebLogic Server와 함께 제공되는 ChangeDetector 구성 파일에 미리 설정되어 있습니다. 이와 같은 미리 설정된 속성 대신 다른 속성을 사용하려면 필요한 속성이 사용되도록 ChangeDetector XML 구성 파일을 수정하십시오. ChangeDetector 및 ChangeDetector XML 구성 파일을 사용하는 방법에 대한 자세한 내용은 *CA APM ChangeDetector 사용자 안내서*를 참조하십시오.

에이전트 프로필 수정

에이전트 프로필을 수정하여 ChangeDetector에 맞게 환경을 구성하십시오.

중요! 여러 에이전트에 동일한 에이전트 프로필을 사용하려면 에이전트 프로필 대신 시작 스크립트를 수정하십시오. 이렇게 하지 않으면 동일한 에이전트 프로필을 사용하는 도메인 간에 충돌이 발생할 수 있습니다.

다음 단계를 수행하십시오.

1. WebLogic Server 에서 IntroscopeAgent.profile 파일을 열고 파일 끝에 다음 속성을 추가합니다.

```
DOMAIN_HOME=<filepath1>
```

```
APPLICATION_HOME=<filepath2>
```

여기서 <filepath1>과 <filepath2>는 각각 도메인 또는 응용 프로그램 홈의 파일 경로입니다.

참고: DOMAIN_HOME 및 APPLICATION_HOME 속성은 CA APM for Oracle WebLogic Server 와 함께 제공되는 ChangeDetector 구성 파일에 미리 설정되어 있습니다. 이와 같은 미리 설정된 속성 대신 다른 속성을 사용하려면 필요한 속성이 사용되도록 ChangeDetector XML 구성 파일을 수정하십시오. ChangeDetector 및 ChangeDetector XML 구성 파일을 사용하는 방법에 대한 자세한 내용은 *CA APM ChangeDetector 사용자 안내서*를 참조하십시오.

2. IntroscopeAgent.profile 파일을 저장합니다.
3. 응용 프로그램을 다시 시작합니다.

확장 제거

확장을 제거하려면 PBD 파일과 Enterprise Manager 및 에이전트의 파일을 수동으로 제거해야 합니다.

다음 단계를 수행하십시오.

1. <APM_Home>에서 다음 파일을 제거합니다.
 - .\config\modules\PPWebLogicManagementModule.jar
 - .\ext\ddtv\PPWebLogicTypeview.xml
 - .\ext\PPWebLogicExtensionPlugins.jar

2. 응용 프로그램 서버 컴퓨터의 *wily* 디렉터리에서 다음 파일을 제거합니다.
 - PPWebLogicJMXFilterString.txt
 - ChangeDetector-config-WebLogic.xml
3. IntroscopeAgent.profile 의 introscope.autoprobe.directives 파일에서 ppweblogic.pbd 를 제거합니다.
4. 에이전트를 다시 시작합니다.

제 3 장: 확장 사용

이 장에서는 확장에 포함되어 있는 대시보드와 경고에 대해 설명합니다.

이 섹션은 다음 항목을 포함하고 있습니다.

[Workstation 에서 WebLogic Server 메트릭 보기](#) (페이지 23)

[특수 요소 보기](#) (페이지 24)

[WebLogic Server 대시보드 보기](#) (페이지 25)

Workstation 에서 WebLogic Server 메트릭 보기

Workstation 에서 WebLogic Server 메트릭을 볼 수 있습니다.

다음 단계를 수행하십시오.

1. 응용 프로그램을 시작합니다.
2. Enterprise Manager 를 시작합니다.
3. Workstation 을 시작하고 로그인합니다.
4. Investigator 트리를 열어 메트릭을 표시합니다.

모든 WebLogic 관련 메트릭이 Investigator 트리의 WebLogic 노드 아래에 나타납니다. 응용 프로그램에서 사용하는 WebLogic 리소스에 따라 Workstation 에 메트릭이 표시됩니다.

5. 탭을 선택하여 Investigator 에 표시되는 정보를 보는 방식을 변경할 수 있습니다. 사용할 수 있는 탭 보기는 Investigator 트리에서 현재 선택되어 있는 리소스 또는 메트릭에 따라 달라집니다.

참고: 표준 탭 보기에 대한 자세한 내용은 *CA APM Workstation 사용자 안내서*를 참조하십시오.

추가 정보:

[WebLogic Server 메트릭](#) (페이지 31)

특수 요소 보기

AppServer 탭 보기에는 확장된 특수 요소가 포함되어 있습니다.

이러한 요소를 보려면 Investigator 트리에서 에이전트 또는 WebLogic 노드를 선택하십시오.

AppServer 탭 보기에는 다음과 같은 메트릭이 표시됩니다.

라이브 HTTP 세션 수

쿠키, 파일, 메모리 및 JDBC 세션의 성능에 대한 정보를 제공합니다. WebLogic Server의 경우 라이브 HTTP 세션 수는 다음 메트릭에서 결정됩니다.

`WebLogic\|HTTP Sessions Subsystem(HTTP 세션 하위 시스템)\|All Sessions(모든 세션):Session Count(세션 수)`

스레드 풀 대기 요청 수

실행 가능한 스레드의 가용성에 대한 정보를 제공합니다. WebLogic Server의 경우 스레드 가용성은 다음 메트릭에서 결정됩니다.

`JMX\|(.*)\|(Type=Ex|Ex)ecuteQueueRuntime(.*):PendingRequestCurrentCount`

JDBC 연결 풀 대기 스레드 수

풀의 여유 연결 수에 대한 정보를 제공합니다. WebLogic Server의 경우 연결 가용성은 다음 메트릭에서 결정됩니다.

`JMX\|(.*)\|((Type=JDBC|JDBC)DataSourceRuntime|(Type=JDBC|JDBC)ConnectionPoolRuntime)(.*):WaitingForConnectionCurrentCount`

EJB 풀 대기 스레드 수

사용 가능한 EJB 인스턴스 수에 대한 정보를 제공합니다. WebLogic Server의 경우 EJB 가용성은 다음 메트릭에서 결정됩니다.

`JMX\|(.*)\|(Type=EJB|EJB)PoolRuntime(.*):Waiter(Total|Current)Count`

WebLogic Server 대시보드 보기

CA APM for Oracle WebLogic Server 에는 WebLogic Server 성능 메트릭을 표시하는 대시보드가 미리 구성되어 있습니다. 이 대시보드 이름은 *WebLogic* 으로 시작하므로 다른 설치된 대시보드나 표준 대시보드와 구분됩니다.

Workstation 대시보드에 표시되는 경고는 많은 성능 메트릭에 정의되는 임계값에 따라 달라집니다. 이러한 경고 및 임계값을 환경에 맞게 사용자 지정하는 방법에 대한 자세한 내용은 *CA APM Workstation 사용자 안내서*를 참조하십시오.

참고: Investigator 탭 보기의 경고에 대한 임계값은 사용자 지정할 수 없습니다.

다음 단계를 수행하십시오.

1. Workstation 을 시작합니다.
2. "Workstation" > "New Console"(새 콘솔)로 이동합니다.
새 콘솔 창이 열립니다.
3. 새 콘솔 창의 맨 위에 있는 드롭다운 목록을 사용하여 원하는 WebLogic Server 대시보드로 이동합니다.

참고: 대시보드의 설계, 사용 및 탐색에 대한 자세한 내용은 *CA APM Workstation 사용자 안내서*를 참조하십시오.

WebLogic Application Server - 개요

"WebLogic Application Server - 개요" 대시보드는 WebLogic Application Server 의 전체 건전성을 보여 주는 상위 수준 대시보드입니다. 이 개요 대시보드는 WebLogic 리소스를 "서버 실행 스레드 가용성", "JDBC 연결 요청 대기 수", "라이브 HTTP 세션 수" 및 "EJB 풀 가용성"의 네 가지 범주에서 모니터링합니다. 경고 표시기에는 각 리소스의 상태가 한눈에 파악할 수 있는 형태로 표시됩니다.

"개요" 대시보드에는 경고 표시기의 의미에 대한 설명과 권장 수정 조치가 있는 작업 메모가 포함되어 있습니다.

보조 대시보드 중 하나를 사용하여 자세한 정보를 보려는 특정 하위 시스템으로 이동할 수 있습니다. "WebLogic Application Server - 개요" 대시보드에서 경고 표시기 중 하나를 클릭하면 관련 대시보드가 표시됩니다.

WebLogic - JDBC 연결 풀

"WebLogic - JDBC 연결 풀" 대시보드에는 JDBC 연결의 성능을 보여 주는 그래프가 포함되어 있습니다.

"WebLogic - JDBC 연결 풀" 대시보드는 그래프 동작에 대해 설명하고 권장 수정 조치를 제공하는 작업 메모와 함께 다음과 같은 정보를 제공합니다.

- 연결 요청 대기 수
- 활성 연결 수
- 누수된 연결 수
- 총 연결 수

WebLogic - HTTP 세션

"WebLogic - HTTP 세션" 대시보드에는 쿠키, 파일, 메모리 및 JDBC 세션에 대한 HTTP 세션 정보를 표시하는 그래프가 포함되어 있습니다. 이 대시보드는 열려 있는 세션 수와 세션을 생성하는 데 걸리는 평균 시간에 대한 정보를 제공합니다.

"WebLogic - HTTP 세션" 대시보드는 그래프 동작에 대해 설명하고 권장 수정 조치를 제공하는 작업 메모와 함께 다음과 같은 정보를 제공합니다.

- 라이브 HTTP, 쿠키, 메모리 세션
 - 라이브 HTTP 세션 수
 - 쿠키 세션 수
 - 메모리 세션 수

- 파일, JDBC, 복제된 세션:
 - 파일 세션 수
 - JDBC 세션 수
 - 복제된 세션 수

WebLogic - EJB 하위 시스템

"WebLogic - EJB 하위 시스템" 대시보드에는 EJB 하위 시스템의 성능을 보여주는 그래프가 포함되어 있습니다.

"WebLogic - EJB 하위 시스템" 대시보드는 그래프 동작에 대해 설명하고 권장 수정 조치를 제공하는 작업 메모와 함께 다음과 같은 정보를 제공합니다.

- EJB 풀/캐시/기능 평균 응답 시간
 - 엔터티 Bean 풀
 - 상태 비저장 Bean 풀
 - 메시지 구동 Bean 풀
 - 상태 저장 NRU 캐시
 - 상태 저장 LRU 캐시
 - 상태 저장 Bean 읽기/저장/복제 시간

WebLogic - JMS 하위 시스템

"WebLogic - JMS 하위 시스템" 대시보드에는 JMS 메시징 시스템의 성능을 보여주는 그래프가 포함되어 있습니다.

"WebLogic - JMS 하위 시스템" 대시보드는 다음과 같은 정보를 제공합니다.

- JMS 동작
 - JMS 보내기/받기 메시지 시간
 - JMS 보내기/받기 메시지 비율/초
 - JMS 읽기/쓰기 현재 바이트 수
 - JMS 읽기/쓰기 누적 바이트 수

- JMS 큐, JMS 항목
 - JMS 큐/항목 현재 소비자 수
 - JMS 큐/항목 현재 메시지 수
 - JMS 큐/항목 현재 바이트 수
- JMS 큐/항목 누적 메시지 수

WebLogic - JTA 하위 시스템

"WebLogic - JTA 하위 시스템" 대시보드에는 트랜잭션 관리자의 JTA(Java Transaction API) 성능을 보여 주는 그래프가 들어 있습니다.

"WebLogic - JTA 하위 시스템" 대시보드는 다음과 같은 정보를 제공합니다.

- 트랜잭션 수 합계
- 커밋 수
- 중단 횟수
- 트랜잭션 롤백 및 추론
 - 롤백 총 수
 - 롤백 시간 만료 수
 - 총 추론 수

WebLogic - 보안

"WebLogic - 보안" 대시보드에는 응용 프로그램의 보안 기능에 대한 정보를 표시하는 그래프가 포함되어 있습니다.

"WebLogic - 보안" 대시보드는 다음과 같은 정보를 제공합니다.

- 사용자 인증 기능의 평균 응답 시간:
 - 기본 사용자 양식 인증
 - 사용자 인증
 - 사용자로 실행

- 다른 보안 기능의 평균 응답 시간:
 - 액세스 검사
 - 역할 가져오기
 - 결과 심사
 - 주체 확인

부록 A: WebLogic Server 메트릭

이 부록에서는 CA APM for Oracle WebLogic Server 메트릭에 대해 설명합니다.

이 섹션은 다음 항목을 포함하고 있습니다.

[EJB 하위 시스템](#) (페이지 31)

[서블릿 하위 시스템 메트릭](#) (페이지 35)

[HTTP 세션](#) (페이지 35)

[JMS 하위 시스템](#) (페이지 36)

[분산 대상 메트릭](#) (페이지 37)

[Security Subsystem\(보안 하위 시스템\)](#) (페이지 38)

[Clustering\(클러스터링\)](#) (페이지 39)

[XML Subsystem\(XML 하위 시스템\)](#) (페이지 40)

[JMX 메트릭](#) (페이지 41)

[JMX 집계 메트릭](#) (페이지 45)

EJB 하위 시스템

Workstation 은 다음과 같은 하위 노드 아래에 WebLogic EJB 하위 시스템에 대한 메트릭을 표시합니다.

- 상태 저장 메트릭
- NRU 캐시 메트릭
- LRU 캐시 메트릭
- 상태 비저장 메트릭
- 엔터티 캐시
- 메시지 구동
- 엔터티

상태 저장 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "상태 저장" 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

File Serialization(파일 직렬화):Average Response Time (ms)(평균 응답 시간(ms))

파일을 직렬화하는 데 걸린 평균 시간입니다.

File Deserialization(파일 직렬화 취소): Average Response Time (ms)(평균 응답 시간(ms))

파일 직렬화를 취소하는 데 걸린 평균 시간입니다.

Replicate Bean(Bean 복제):Average Response Time (ms)(평균 응답 시간(ms))

Bean 을 복제하는 데 걸린 평균 시간입니다.

NRU 캐시 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "NRU 캐시" 노드 아래에서 최근에 사용되지 않은 캐시에 대한 다음과 같은 메트릭을 사용할 수 있습니다.

Get Beans from Pool(풀에서 Bean 가져오기):Average Response Time (ms)(평균 응답 시간(ms))

NRU 캐시 풀에서 Bean 을 가져오는 데 걸린 평균 시간입니다.

Get Beans from Pool(풀에서 Bean 가져오기):Average Responses Per Second(초당 평균 응답 수)

NRU 캐시 풀에서 Bean 을 가져오는 빈도입니다.

Return Beans to Pool(풀에 Bean 반환):Average Response Time (ms)(평균 응답 시간(ms))

NRU 캐시 풀의 Bean 을 반환하는 데 걸린 평균 시간입니다.

Return Beans to Pool(풀에 Bean 반환): Average Responses Per Second(초당 평균 응답 수)

NRU 캐시 풀의 Bean 을 반환하는 빈도입니다.

LRU 캐시 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "LRU 캐시" 노드 아래에서 최근에 가장 적게 사용된 캐시에 대한 다음과 같은 메트릭을 사용할 수 있습니다.

Get Beans from Pool(풀에서 Bean 가져오기):Average Response Time (ms)(평균 응답 시간(ms))

LRU 캐시 풀에서 Bean 을 가져오는 데 걸린 평균 시간입니다.

Get Beans from Pool(풀에서 Bean 가져오기):Average Responses Per Second(초당 평균 응답 수)

LRU 캐시 풀에서 Bean 을 가져오는 빈도입니다.

Return Beans to Pool(풀에 Bean 반환):Average Response Time (ms)(평균 응답 시간(ms))

LRU 캐시 풀의 Bean 을 반환하는 데 걸린 평균 시간입니다.

Return Beans to Pool(풀에 Bean 반환):Average Responses Per Second(초당 평균 응답 수)

LRU 캐시 풀의 Bean 을 반환하는 빈도입니다.

상태 비저장 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "상태 비저장" 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Get Bean(Bea 가져오기):Average Response Time (ms)(평균 응답 시간(ms))

상태 비저장 Bean 을 가져오는 데 걸린 평균 시간입니다.

Get Bean(Bea 가져오기):Average Responses Per Second(초당 평균 응답 수)

상태 비저장 Bean 을 가져오는 빈도입니다.

Return Bean(Bea 반환):Average Response Time (ms)(평균 응답 시간(ms))

상태 비저장 Bean 을 반환하는 데 걸린 평균 시간입니다.

Return Bean(Bea 반환):Average Responses Per Second(초당 평균 응답 수)

상태 비저장 Bean 을 반환하는 빈도입니다.

엔터티 캐시 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "엔터티 캐시" 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Get Bean(Bean 가져오기):Average Response Time (ms)(평균 응답 시간(ms))

엔터티 LRU 캐시에서 Bean 을 가져오는 데 걸린 평균 시간입니다.

Get Bean(Bean 가져오기):Average Responses Per Second(초당 평균 응답 수)

엔터티 LRU 캐시에서 Bean 을 가져오는 빈도입니다.

Return Bean(Bean 반환):Average Response Time (ms)(평균 응답 시간(ms))

엔터티 LRU 캐시의 Bean 을 반환하는 데 걸린 평균 시간입니다.

Return Bean(Bean 반환):Average Responses Per Second(초당 평균 응답 수)

엔터티 LRU 캐시의 Bean 을 반환하는 빈도입니다.

Message-Driven 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "메시지 구동" 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Get Bean(Bean 가져오기):Average Response Time (ms)(평균 응답 시간(ms))

메시지 구동 Bean 을 가져오는 데 걸린 평균 시간입니다.

Get Bean(Bean 가져오기):Average Responses Per Second(초당 평균 응답 수)

메시지 구동 Bean 을 가져오는 빈도입니다.

엔터티 메트릭

Investigator 트리의 "WebLogic" > "EJB 하위 시스템" > "엔터티" 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Get Bean(Bean 가져오기):Average Response Time (ms)(평균 응답 시간(ms))

엔터티 Bean 을 가져오는 데 걸린 평균 시간입니다.

Get Bean(Bean 가져오기):Average Responses Per Second(초당 평균 응답 수)

엔터티 Bean 을 가져오는 빈도입니다.

서블릿 하위 시스템 메트릭

Investigator 트리의 "WebLogic" > "Servlet Subsystem"(서블릿 하위 시스템) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Error Response Count(오류 응답 수)

누적 오류 수입니다.

프록시 서비스:Average Response Time (ms)(평균 응답 시간(ms))

프록시 서비스의 평균 시간입니다.

HTTP 세션

Investigator 트리의 "WebLogic" > "HTTP Sessions Subsystem"(HTTP 세션 하위 시스템) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다. 메트릭은 "Cookie Sessions"(쿠키 세션), "File Sessions"(파일 세션), "JDBC Sessions"(JDBC 세션), "Memory Sessions"(메모리 세션) 및 "Replicated Sessions"(복제된 세션)에 대한 하위 노드 아래에 구성됩니다.

All Sessions(모든 세션):Session Count(세션 수)

열려 있는 HTTP 세션의 합계입니다.

다음은 쿠키 세션입니다.

세션 수

열려 있는 쿠키 세션의 수입니다.

Create Session(세션 생성):Average Response Time (ms)(평균 응답 시간(ms))

세션을 생성하는 데 걸린 평균 시간입니다.

다음은 파일 세션입니다.

세션 수

열려 있는 파일 세션의 수입니다.

Create Session(세션 생성):Average Response Time (ms)(평균 응답 시간(ms))

세션을 생성하는 데 걸린 평균 시간입니다.

다음은 JDBC 세션입니다.

세션 수

열려 있는 JDBC 세션의 수입니다.

Create Session(세션 생성):Average Response Time (ms)(평균 응답 시간(ms))

세션을 생성하는 데 걸린 평균 시간입니다.

다음은 메모리 세션입니다.

세션 수

열려 있는 메모리 세션의 수입니다.

Create Session(세션 생성):Average Response Time (ms)(평균 응답 시간(ms))

세션을 생성하는 데 걸린 평균 시간입니다.

다음은 복제된 세션입니다.

세션 수

열려 있는 복제된 세션의 수입니다.

Create Session(세션 생성):Average Response Time (ms)(평균 응답 시간(ms))

세션을 생성하는 데 걸린 평균 시간입니다.

JMS 하위 시스템

Investigator 트리의 "WebLogic" > "JMS Subsystem"(XML 하위 시스템) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Send Message(메시지 보내기):Average Response Time (ms)(평균 응답 시간(ms))

JMS 메시지를 보내는 데 걸린 평균 시간입니다.

Send Message(메시지 보내기):Average Responses Per Second(초당 평균 응답 수)

JMS 메시지를 보내는 속도입니다.

Receive Message:(메시지 받기):Average Response Time (ms)(평균 응답 시간(ms))

JMS 메시지를 받는 데 걸린 평균 시간입니다.

Receive Message:(메시지 받기):Average Responses Per Second(초당 평균 응답 수)

JMS 메시지를 받는 속도입니다.

Producer Send Response(생산자 응답 보내기):Average Response Time (ms)(평균 응답 시간(ms))

메시지 생산자가 응답을 보내는 데 걸린 평균 시간입니다.

Producer Send Response(생산자 응답 보내기):Average Responses Per Second(초당 평균 응답 수)

메시지 생산자가 응답을 보내는 속도입니다.

분산 대상 메트릭

Investigator 트리의 "WebLogic" > "JMS 하위 시스템" > "Distributed Destination"(분산 대상) 하위 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Add Rate response(속도 응답 추가):Average Response Time (ms)(평균 응답 시간(ms))

분산 대상이 응답을 추가하는 데 걸린 평균 시간입니다.

Add Rate response(속도 응답 추가):Average Responses Per Second(초당 평균 응답 수)

분산 대상이 속도 응답을 추가하는 속도입니다.

Remove Rate Response(속도 응답 제거):Average Response Time (ms)(평균 응답 시간(ms))

분산 대상이 응답을 제거하는 데 걸린 평균 시간입니다.

Remove Rate Response(속도 응답 제거):Average Responses Per Second(초당 평균 응답 수)

분산 대상이 속도 응답을 제거하는 속도입니다.

Security Subsystem(보안 하위 시스템)

Investigator 트리의 "WebLogic" > "Security Subsystem"(보안 하위 시스템) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Basic User Form Authentication(기본 사용자 양식 인증): Average Response Time (ms)(평균 응답 시간(ms))

기본 사용자 양식 인증을 확인하는 데 걸린 평균 시간입니다.

사용자 인증(Authenticate Users): Average Response Time (ms)(평균 응답 시간(ms))

사용자를 인증하는 데 걸린 평균 시간입니다.

Access Checks(액세스 검사): Average Response Time (ms)(평균 응답 시간(ms))

액세스를 검사하는 데 걸린 평균 시간입니다.

Get Roles(역할 가져오기): Average Response Time (ms)(평균 응답 시간(ms))

리소스와 제목이 지정된 역할을 가져오는 데 걸린 평균 시간입니다.

Adjudicate Results(결과 심사): Average Response Time (ms)(평균 응답 시간(ms))

결과를 심사하는 데 걸린 평균 시간입니다.

Validate Principal(주체 확인): Average Response Time (ms)(평균 응답 시간(ms))

주체를 확인하는 데 걸린 평균 시간입니다.

Run as Users(사용자로 실행): Average Response Time (ms)(평균 응답 시간(ms))

특정 사용자로 실행하는 요청에 대한 평균 시간입니다.

Clustering(클러스터링)

Investigator 트리의 "WebLogic" > "Clustering(클러스터링) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

Change Event(변경 이벤트):Count(수)

모든 수신기에 대해 클러스터 변경 이벤트가 실행된 횟수입니다.

Change Event(변경 이벤트):Average Responses Per Second(초당 평균 응답 수)

모든 수신기에 대해 클러스터 변경 이벤트가 실행된 속도입니다.

Announce(알림):Count(수)

원격 클러스터로부터 수신한 알림의 수입니다.

Announce(알림):Average Response Time (ms)(평균 응답 시간(ms))

원격 클러스터의 알림을 처리하는 데 걸린 평균 시간입니다.

Announce(알림):Average Responses Per Second(초당 평균 응답 수)

원격 클러스터로부터 알림을 수신하는 속도입니다.

Full State Dump(전체 상태 덤프):Count(수)

모든 원격 클러스터로 전송된 전체 상태 덤프의 수입니다.

Full State Dump(전체 상태 덤프):Average Responses Per Second(초당 평균 응답 수)

모든 원격 클러스터로 전송된 전체 상태 덤프의 속도입니다.

NAK Processing(NAK 처리):Average Response Time (ms)(평균 응답 시간(ms))

원격 클러스터의 NAK 를 처리하는 데 걸린 평균 시간입니다.

NAK Processing(NAK 처리):Average Responses Per Second(초당 평균 응답 수)

원격 클러스터로부터 NAK 를 수신하는 속도입니다.

Peer Gone Listeners(피어가 사라진 수신기):Count(수)

RJVM 에 대한 피어가 사라진 수신기의 수입니다.

Peer Gone Listeners(피어가 사라진 수신기)|Add(추가):Average Response Time (ms)(평균 응답 시간(ms))

피어가 사라진 수신기를 추가하는 데 걸린 평균 시간입니다.

Peer Gone Listeners(피어가 사라진 수신기)|Add(추가):Average Responses Per Second(초당 평균 응답 수)

피어가 사라진 수신기를 추가한 비율입니다.

RJVM Remote Call(RJVM 원격 호출):Error Count(오류 수)

RJVM 원격 호출에 대한 오류 수입니다.

XML Subsystem(XML 하위 시스템)

Investigator 트리의 "WebLogic" > "XML Subsystem"(XML 하위 시스템) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

SAX Parsers(SAX 파서):Creation Count(생성 수)

생성된 SAX 파서의 수입니다.

Document Builder(문서 작성기):Creation Count(생성 수)

생성된 문서 작성기 파서의 수입니다.

SAX Transformer(SAX 변환기):Creation Count(생성 수)

생성된 SAX 변환기의 수입니다.

JMX 메트릭

Investigator 트리의 JMX 노드 아래에서 다음과 같은 메트릭을 `JMX\|(.*)\|(Type=)?<MBean Name>:<Attribute Name>` 형식으로 사용할 수 있습니다.

ServerRuntime: OpenSocketsCurrentCount

서버에서 현재 열려 있는 소켓의 개수입니다.

ServerRuntime: SocketsOpenedTotalCount

서버에서 열린 적이 있는 소켓의 합계입니다.

ExecuteQueueRuntime: ExecuteThreadCurrentIdleCount

큐에 할당된 유휴 스레드의 수입니다.

ExecuteQueueRuntime: PendingRequestCurrentCount

큐에서 대기 중인 요청의 수입니다.

ExecuteQueueRuntime: ServicedRequestTotalCount

이 큐가 처리하는 요청의 수입니다.

JDBCDataSourceRuntime: ActiveConnectionsCurrentCount

이 데이터 원본에서 현재 사용되고 있는 JDBC 연결의 수입니다.

JDBCDataSourceRuntime: ConnectionsTotalCount

이 데이터 원본이 배포된 시점부터 데이터 원본에서 생성된 JDBC 연결의 합계입니다.

JDBCDataSourceRuntime: LeakedConnectionCount

누수가 발생한 JDBC 연결의 수입니다.

JDBCDataSourceRuntime: WaitingForConnectionCurrentCount

JDBC 연결을 대기 중인 연결 요청의 수입니다.

JDBCDataSourceRuntime: NumAvailable

이 데이터 원본에서 현재 사용할 수 있는 JDBC 연결의 수입니다.

EJBCacheRuntime: ActivationCount

이 EJB 홈에서 활성화된 Bean 의 합계입니다.

EJBCacheRuntime: CacheAccessCount

이 캐시에서 Bean 에 액세스하려고 시도한 횟수의 합계입니다.

EJBCacheRuntime: CachedBeansCurrentCount

이 EJB 홈에서 EJB 캐시에 있는 Bean 의 현재 수입입니다.

EJBCacheRuntime: CacheHitCount

성공한 캐시 액세스 시도의 수입입니다.

EJBLockingRuntime: TimeoutTotalCount

Bean 에 대한 잠금을 대기하다 만료된 스레드의 합계입니다.

EJBPoolRuntime: BeansInUseCount

이 풀에서 현재 사용되고 있는 Bean 인스턴스의 수입입니다.

EJBPoolRuntime: IdleBeansCount

이 풀에서 현재 사용 가능한 Bean 인스턴스의 수입입니다.

EJBPoolRuntime: TimeoutTotalCount

이 풀에서 사용 가능한 Bean 을 대기하다 만료된 스레드의 합계입니다.

EJBPoolRuntime: WaiterCurrentCount

현재 여유 풀에서 사용 가능한 Bean 인스턴스를 대기 중인 스레드의 수를 제공합니다.

EJBTransactionRuntime: TransactionCommittedTotalCount

이 EJB 에 대해 커밋된 트랜잭션의 합계입니다.

JMSRuntime: ConnectionsTotalCount

마지막으로 다시 설정한 후 이 WebLogic Server 에서 수행된 JMS 연결의 합계입니다.

JMSRuntime: JMSServersCurrentCount

이 WebLogic Server 인스턴스에 배포된 JMS 서버의 합계입니다.

JMSPooledConnectionRuntime: NumAvailable

풀에서 현재 사용되고 있지 않은 사용 가능한 JMS 세션의 수입입니다.

JMSDestinationRuntime: BytesReceivedCount

마지막으로 다시 설정한 후 이 대상에서 수신한 바이트 수입입니다.

JMSDestinationRuntime: ConsumersTotalCount

마지막으로 다시 설정한 후 이 대상을 액세스한 소비자의 합계입니다.

JMSDestinationRuntime: MessagesReceivedCount

마지막으로 다시 설정한 후 이 대상에서 수신한 메시지 수입입니다.

JMSDestinationRuntime: BytesCurrentCount

현재 대상에 저장된 바이트 수입입니다.

JMSDestinationRuntime: ConsumersCurrentCount

현재 이 대상에 액세스하는 소비자 수입입니다.

JMSDestinationRuntime: MessagesCurrentCount

대상의 현재 메시지 수입입니다.

JMSDurableSubscriberRuntime: MessagesReceivedCount

마지막으로 다시 설정한 후 이 지속 구독자가 수신하는 메시지 수입입니다.

JMSDurableSubscriberRuntime: BytesCurrentCount

이 지속 구독자가 수신하는 바이트 수입입니다.

JMSDurableSubscriberRuntime: MessagesCurrentCount

이 지속 구독자가 여전히 사용할 수 있는 메시지 수입입니다.

JMSServerRuntime: BytesReceivedCount

마지막으로 다시 설정한 후 이 JMS 서버가 수신하는 바이트의 합계입니다.

JMSServerRuntime: BytesCurrentCount

현재 이 JMS 서버에 저장된 바이트 수입입니다.

JMSConnectionRuntime: SessionsCurrentCount

이 연결의 현재 세션 수입입니다.

TransactionNameRuntime: TransactionAbandonedTotalCount

마지막으로 다시 설정한 후 중단된 트랜잭션의 합계입니다.

TransactionNameRuntime: TransactionCommittedTotalCount

마지막으로 다시 설정한 후 커밋된 트랜잭션의 합계입니다.

TransactionNameRuntime: TransactionHeuristicsTotalCount

마지막으로 다시 설정한 후 추론 상태로 완료된 트랜잭션의 합계입니다.

TransactionNameRuntime: TransactionRolledBackTotalCount

마지막으로 다시 설정한 후 롤백된 트랜잭션의 합계입니다.

TransactionNameRuntime: TransactionRolledBackTimeoutTotalCount

마지막으로 다시 설정한 후 시간 만료로 인해 롤백된 트랜잭션의 합계입니다.

TransactionNameRuntime: TransactionTotalCount

마지막으로 다시 설정한 후 처리된(커밋/롤백/추론) 트랜잭션의 합계입니다.

TransactionResourceRuntime: TransactionCommittedTotalCount

마지막으로 다시 설정한 후 커밋된 트랜잭션의 합계입니다.

TransactionResourceRuntime: TransactionHeuristicsTotalCount

마지막으로 다시 설정한 후 추론 상태로 완료된 트랜잭션의 합계입니다.

TransactionResourceRuntime: TransactionRolledBackTotalCount

마지막으로 다시 설정한 후 롤백된 트랜잭션의 합계입니다.

TransactionResourceRuntime: TransactionRolledBackTimeoutTotalCount

마지막으로 다시 설정한 후 시간 만료로 인해 롤백된 트랜잭션의 합계입니다.

TransactionResourceRuntime: TransactionTotalCount

마지막으로 다시 설정한 후 처리된(커밋/롤백/추론) 트랜잭션의 합계입니다.

JTARuntime: TransactionAbandonedTotalCount

마지막으로 다시 설정한 후 중단된 트랜잭션의 합계입니다.

JTARuntime: TransactionCommittedTotalCount

마지막으로 다시 설정한 후 커밋된 트랜잭션의 합계입니다.

JTARuntime: TransactionHeuristicsTotalCount

마지막으로 다시 설정한 후 추론 상태로 완료된 트랜잭션의 합계입니다.

JTARuntime: TransactionRolledBackTotalCount

마지막으로 다시 설정한 후 롤백된 트랜잭션의 합계입니다.

JTARuntime: TransactionRolledBackTimeoutTotalCount

마지막으로 다시 설정한 후 시간 만료로 인해 롤백된 트랜잭션의 합계입니다.

JTARuntime: TransactionTotalCount

마지막으로 다시 설정한 후 처리된(커밋/롤백/추론) 트랜잭션의 합계입니다.

Server: IdleConnectionTimeout

HTTP 세션 시간 만료의 현재 값입니다.

JMX 집계 메트릭

Investigator 트리의 "WebLogic" > "JMX Aggregate"(JMX 집계) 노드 아래에서 다음과 같은 메트릭을 사용할 수 있습니다.

스레드 풀: 대기 요청 수

현재 여유 풀에서 사용 가능한 인스턴스를 요청 중인 스레드의 합계를 제공합니다.

JDBC 연결 풀: 대기 스레드 수

현재 여유 풀에서 사용 가능한 연결 인스턴스를 대기 중인 스레드의 합계를 제공합니다.

EJB 풀: 대기 스레드 수

현재 여유 풀에서 사용 가능한 Bean 인스턴스를 대기 중인 스레드의 합계를 제공합니다.