

App Server Agents Supported Environments

On this page:

- [Java Agent Supported Platforms](#)
- [.NET Agent Support](#)
- [Node.js Agent Support](#)
- [PHP Agent Support](#)
- [Python Agent Support](#)
- [Apache Server Agent Support](#)
- [C/C++ Agent Supported Platforms](#)
- [Go Agent Support](#)
- [IIB Agent Support](#)

This page provides an aggregated view of the supported environments for the app server agents.

Java Agent Supported Platforms

In the following tables, note that:

- A dash ("-") in a table cell indicates that this column is not relevant or not supported for that particular environment.
- In cases where no version is provided, assume that all versions are supported. Contact AppDynamics Sales for confirmation.
- For environments that require additional configuration, a separate table describing or linking to configuration information follows the support matrix.
- For environments supported by AppDynamics End User Monitoring, see [Supported Environments and Versions - Web EUM](#).
- For environments supported by AppDynamics Server Visibility, [Standalone Machine Agent Requirements and Supported Environments](#).

JVM Support

The AppDynamics Java Agent uses the standard JVM Tool Interface (JVMTI) mechanism allowing it to instrument any software running on a JVM supporting this mechanism.

AppDynamics certifies the successful operation of the basic mechanisms of instrumentation used by the agent on the following Java runtimes. These capabilities are supported on both JRE or full JDK installations.

Where the agent supports the following advanced memory monitoring features, they are listed for the JVM: Object Instance Tracking (OIT), Automatic Leak Detection (ALD), Content Inspection (CI), and Access Tracking (AT).

JVM	OS	Memory Monitoring Features
Amazon Corretto 8, 11	Linux, Windows	
Azul Zing 15.x.	Linux x64	OIT, ALD

Azul Zulu 1.6, 1.7, 1.8, 9, 10, 11 JDK11 is supported from 4.5.6 onwards	Linux x64	OIT, ALD
HP OpenVMS		
IBM JVM 1.6.x, 1.7.x, 1.8.x		ALD, CI Object instance tracking, automatic leak detection, and custom memory structure monitoring are not supported with the AppDynamics IBM Java Agent. IBM JVMs can be instrumented with the AppDynamics Sun Java Agent to work around this limitation, however this only enables automatic leak detection and custom memory structure monitoring. Object instance tracking is not available. Working around this limitation can result in negative performance impact and is not recommended. In such cases, the IBM JVM needs to be restarted to enable custom memory structure monitoring.
Oracle Rockit JVM 28.1+	Linux Intel 64, Windows	
Oracle/BEA JRockit 1.6		
Oracle/Sun JVM 1.6, 1.7, 1.8, 9, 10, 11 JDK11 is supported from 4.5.6 onwards	Solaris Sparc 64, Windows, Linux	OIT, ALD, CI, AT Content Inspection and Access Tracking require a JVM restart.
OpenJDK 1.6, 1.7, 1.8, 1.9, 10, 11 JDK11 is supported from 4.5.6 onwards		OIT, ALD, CI, AT Content Inspection and Access Tracking require a JVM restart.
SAP JDK 6+	Windows, Solaris, Linux, HP-UX, i5/OS, AIX	

JVM Application Server and Framework Support

AppDynamics supports the use of the Java Agent to instrument any application component running on a supported JVM, irrespective of how that component is built. The power of the AppDynamics platform is that it can automatically discover the topology and behavior of complex enterprise applications without requiring deep technical knowledge of the application's underlying code.

Frequently, Java-based systems employ standard framework code to implement business logic. Automatic instrumentation of framework code relies on knowledge of the business logic and programming patterns employed by the framework. AppDynamics instrumentation targets processing hand-offs between components, called [entry points](#) and [exit points](#), either within the JVM or between JVMs. This includes hand-offs between frameworks in cases where multiple frameworks are being used together. This section covers the capabilities for frameworks for which AppDynamics provides automatic detection rules.

Monitoring application components built using frameworks not listed here may require custom configuration. The custom configuration

may involve, for example, custom [POJO entry](#) or [exit points](#). If you understand how the application behaves internally, you can easily configure this type of instrumentation. For more complex configuration tasks, contact your account representative to discuss how to engage the AppDynamics customer success organization.

JVM Language Frameworks Support

No additional configuration is required for these frameworks.

Vendor	JVM Language Framework	Version	Correlation/Entry Points	Exit Points	Transports	Notes
Open Source	Akka Actor	2.1 – 2.4.x	Yes	Yes	Netty	4.3.1 required for 2.4.x 2.4x support includes Persistence Remoting exit/entry supported
Open Source	Akka HTTP	Akka Actor 2.4.x Akka HTTP upto 10.1.5 Scala 2.11, 2.12	Yes	Yes	HTTP	EUM is supported
Open Source	Groovy	-	Yes	Yes		
Open Source	Play for Scala Play for Java	2.1 – 2.6 Scala 2.11, 2.12	Yes	-	HTTP over Netty server Akka HTTP server	Includes framework specific entry and exit points Play EUM-APM correlation supported
Open Source	Scala	2.11.6				
Open Source	Spray toolkit (Spray.io)	1.1.x 1.1.3	Yes	Yes	HTTP	Entry points are detected and configurable as servlet entry point and exit points as HTTP exits
Pivotal	Grails	-	-	-	-	

Java Frameworks Support

The Java Agent supports these Java frameworks. Some require additional configuration as indicated in the Configuration Notes column.

Vendor	Framework	Version	SOA protocol (WebServices)	Auto Naming	Entry Points	Exit Points	Detection	Configuration Notes
Adobe	BlazeDS	-	HTTP and JMS adaptor	-	Yes		-	Example Message Queue Backend Configuration
Adobe	ColdFusion	8.x, 9.x	-	-	Yes	-	Configuration required for transaction discovery	Configuration is required for transaction discovery. See: <ul style="list-style-type: none"> Java Business Transaction Detection Servlet Entry Points
Apache	Cassandra with Thrift framework	-	-	-	Yes	Yes	Apache Thrift Entry and Exit points are detected	

Apache	Struts	1.x, 2.x	-	-	Yes		Struts Actions are detected as entry points; struts invocation handler is instrumented	Struts Entry Points
Apache	Tapestry	5	-	-	Yes	-	Not by default	See: <ul style="list-style-type: none"> • Java Business Transaction Detection • Servlet Entry Points
Apache	Wicket	-	-	No	Yes	-	Not by default	See: <ul style="list-style-type: none"> • Java Business Transaction Detection • Servlet Entry Points
Apple	WebObjects	5.4.3	HTTP	Yes	Yes	-	Yes	Apple WebObjects Startup Settings
axonframework.org	Axon	2.x, 3.x	-	-	Commands on the Command Bus continue existing Business Transactions	Correlation for Distributed Command Bus on JGroups and for Spring Cloud Connector transport as an exit		
Open Source	CometD	2.6	HTTP	Yes	Yes	-	-	See also "HTTP Exit Points" on Java Backend Detection.
Eclipse	RCP (Rich Client Platform)	-	-	-	-	-	-	
Google	Google Web Toolkit (GWT)	2.5.1	HTTP	Yes	Yes	-	-	
JBoss	JBossWS Native Stack	4.x, 5.x	Native Stack	-	-	-	-	
IBM	IBM-BPM	8.5.7, 8.6	-	Yes	Yes	Yes	Yes	IBM-BPM Support
Open Source	Direct Web Remoting (DWR)	-	-	-	-	-	-	
Open Source	Eclipse Vert.x Core	3.3.3-3.5.4	HTTP	Yes	Yes	Yes	Yes	EUM Correlation is supported
Open Source	Enterprise Java Beans (EJB)	2.x, 3.x	-	-	Yes	-	-	EJB Entry Points
Open Source	Grails	-	-	-	Yes	-	Not by default	
Open Source	Hibernate JMS Listeners	1.x	-	-	-	-	-	
Open Source	Java Abstract Windowing Toolkit (AWT)	-	-	-	-	-	-	
Open Source	Java Server Faces (JSF)	1.x, 2.x	-	Yes	Yes	-	-	Java Business Transaction Detection and Servlet Entry Points
Open Source	Java Server Pages	2.x	-	Yes	-	-	Yes	Servlet Entry Points
Open Source	Java Servlet API	2.x, 3.0	-	-	-	-	-	

Open Source	Jersey	1.x, 2.x	REST, JAX-RS	Yes	Yes	No	Not by default	<p>JAX-RS Support and node properties:</p> <ul style="list-style-type: none"> rest-num-segments rest-transaction rest-uri-segment-scheme <p>See App Agent Node Properties Reference for information on the properties.</p>
Open Source	JRuby HTTP	-	-	-	Yes	-	Not by default	<p>See:</p> <ul style="list-style-type: none"> Java Business Transaction Detection Servlet Entry Points
Open Source	Netty	3.x	HTTP	Yes	Yes	Yes	-	<ul style="list-style-type: none"> Node property to disable Netty Instrumentation: netty-enabled, by default it is true
Open Source	Spring Annotated Web Services	2.x+	HTTP	Yes	Yes	No	-	
Open Source	Spring WebFlux	5.0, 5.1	HTTP	Yes	Spring Boot (Netty, Jetty, Tomcat)	WebClient (Reactor Netty, Reactive Jetty)	By default	-
Open Source	WebSocket	1.0 (Java EE 7, JSR-356)	-	Yes, BT Naming not configurable	Yes, correlation not supported	Yes	Detection is automatic	Node property: websocket-entry-calls-enabled
Oracle	Coherence with Spring Beans	2.x, 3.x	-	-	-	-	-	
Oracle	Swing (GUI)	-	-	-	-	-	-	
Oracle	WebCenter	10.0.2, 10.3.0	-	-	-	-	-	
Spring	Spring MVC	-	-	-	Yes	-	Not by default	See App Agent Node Properties Reference .

Application Servers

The Java Agent supports the following application servers. Some require additional configuration. Click the link on the server or OSGi Runtime for information about additional requirements or related configuration topics. The agent usually discovers application servers as an entry point.

Vendor	Application Server / OSGi Runtime	Version	SOA Protocol	RMI Supported	JMX	Entry Points	Configuration Notes
Adobe	Cold Fusion	8.x, 9.x	-	No	-	Yes	Requires configuration for transaction discovery; see Servlet Entry Points
	Equinox	-	-	-	-	Yes	OSGi Infrastructure Configuration
Apache	Felix	-	-	-	-	Yes	OSGi Infrastructure Configuration
Apache	Sling	-	-	-	-	Yes	OSGi Infrastructure Configuration
Apache	Tomcat	5.x, 6.x, 7.x, 8.x, 9	-	-	Yes		Apache Tomcat Startup Settings
Apache	Resin	1.x - 4.x	-	-	-	-	Resin Startup Settings
Eclipse	Jetty	6.x, 7.x, 8.x, 9x	-	-	-	-	Jetty Startup Settings
IBM	InfoSphere	8.x	-	-	-	Yes	IBM WebSphere and InfoSphere Startup Settings
IBM	WebSphere	6.1, 7.x, 8.x, 9.x	JAX-WS	Yes, detect and correlate	Yes for WebSphere PMI	Yes	IBM WebSphere and InfoSphere Startup Settings
Open Source	Liferay Portal	-	-	-	-	-	

Open Source	JBoss Wildfly (formerly JBoss Application Server) Including versions shipped with JBoss EAP under JBoss Wildfly	4.x to 14.x		Yes		Yes	JBoss and Wildfly Startup Settings
Sun/Oracle	GlassFish Enterprise Server	2.x	-	-	Yes	Yes	GlassFish Startup Settings
Oracle	GlassFish Server and GlassFish Server Open Source Edition	3.x, 4.x	-	-	Yes for AMX	Yes	GlassFish Startup Settings
Oracle and BEA	WebLogic Server	9.x+	JAX-WS	Yes, detect and correlate for 10.x	Yes	Yes	Oracle WebLogic Startup Settings
Software AG	webMethods	9.5, 9.6	-	-	-	Yes	webMethods Startup Settings
Tibco	ActiveMatrix BusinessWorks Service Engine	5.x, 6.x	-	-	-	Yes	Tibco ActiveMatrix BusinessWorks Service Engine Settings
	Application Server (OC4J)	-	-	Yes, detect and correlate for 10.x	-	Yes	
-	Grails, with Tomcat 7.x, Glassfish v3, Weblogic 12.1.1 (12c)	-	-	-	-		

Servlet 3.x detection is not supported.

PaaS Providers

PaaS Provider	Buildpack
Pivotal Cloud Foundry	Java Buildpack 3.4 and higher See Using AppDynamics with Java Applications on Pivotal Cloud for more information.
Red Hat OpenShift 3	JBoss EAP 6.4 and WildFly 8.1 Docker images For documentation and download information, see the AppDynamics Java APM Agent page on the Red Hat Customer Portal.

Message Oriented Middleware Support

The Java Agent supports the following message oriented middleware environments. Some require additional configuration as indicated in the Configuration Notes column. Message oriented middleware servers are usually found by the Java Agent as an entry point.

Vendor	Messaging Server	Version	Protocol	Correlation/Entry Points	Exit Points	JMX	Configuration Notes
Amazon	Simple Queue Service (SQS)	-	-	Yes (correlation only)	Yes	-	See "Amazon Simple Queue Service Backends" on Java Backend Detection
Amazon	Simple Notification Service (SNS)	-	-	No	Yes	-	See "Amazon Simple Notification Service Backends" on Java Backend Detection
Apache	ActiveMQ	5.x+	JMS 1.x	Yes	Yes	Yes	
Apache	ActiveMQ	5.x+	STOMP	No	-	Yes	
Apache	ActiveMQ	5.8.x+	AMQP 1.0	No	-	Yes	Example Message Queue Backend Configuration
Apache	Axis	1.x, 2.x	JAX-WS	Yes	Yes	-	Default exclude rules exist for Apache Axis, Axis2, and Axis Admin Servlets. See also "Web Service Entry Points" on Java Backend Detection .
Apache	Apache CXF	2.1	JAX-WS	Yes	Yes	-	To enable correlation, set node property <code>enable-soap-header-correlation=true</code> .
Apache	Kafka	0.9.0.0 to 1.1.1	-	Yes	Yes	Yes	Kafka consumer entry points are disabled by default. Correlation is supported. See Apache Kafka Consumer Backends .
Apache	Synapse	2.1	HTTP	Yes	Yes	-	To enable correlation, set node property <code>enable-soap-header-correlation=true</code>

Fiorano	Fiorano MQ		-	-	-	-	
IBM	IBM Web Application Server (WAS)	6.1+, 7.x	Embedded JMS	-	Yes	-	Example Message Queue Backend Configuration
IBM	IBM MQ (formerly IBM WebSphere MQ)	6+	JMS	Yes	Yes	-	Example Message Queue Backend Configuration
Mulesoft	Mule ESB	3.4, 3.6, 3.7, 3.8, 3.9, 4.1x	HTTP, JMS	Yes	Yes	-	Mule ESB Startup Settings
Open Source	Eclipse Vert.x verticles	3.3.x, 3.4.x, 3.5.0, 3.6.0	-	Yes (correlation only)	Yes	-	The Java Agent detects messaging exit calls between verticals.
Open Source	Open MQ	-	-	-	-	-	
Oracle	Java Message Service	2.0	JMS	Correlation of the listener is disabled by default	Yes		
Oracle	Oracle AQ	-	JMS	-	Yes	-	
Oracle	OSB deployed on WebLogic	12.2.1	HTTP, JMS	Yes	Yes		OSB Support
Oracle / BEA	WebLogic	9.x+	JMS 1.1	Yes	Yes	Yes	Oracle WebLogic Startup Settings
Progress	SonicMQ	-	-	-	-	-	
Pivotal	RabbitMQ	-	HTTP	-	Yes	-	See "RabbitMQ Backends" on Java Backend Detection
Rabbit	RabbitMQ Spring Client	-	-	Yes	Yes	-	See "RabbitMQ Backends" on Java Backend Detection
Red Hat	HornetQ (formerly JBoss Messaging and JBoss MQ)	-				Yes	
Red Hat	JBoss A-MQ	4.x+	-	-	-	Yes	
Spring	Spring Integration	2.2.0+, 4.0+	JMS	Yes	Yes	Yes	Spring Integration Support See also "Java Message Service Backends" on Java Backend Detection
WSO2	ESB	4.7.0	-	Yes	Yes	-	EUM Correlation is not supported

JDBC Drivers and Database Servers Support

The Java Agent supports these JDBC driver and database server environments. AppDynamics can follow transactions using these drivers to the designated database.

JDBC Vendor	Driver Version	Driver Type	Database Server	Database Version
Apache	10.9.1.0	Embedded or client	Derby	-
Apache	-	-	Cassandra	-
Progress	DataDirect	data connectivity for ODBC and JDBC driver access, data integration, and SaaS and cloud computing solutions	-	-
IBM	JDBC 3.0 version 3.57.82 or JDBC 4.0 version 4.7.85	DB2 Universal JDBC driver	DB2	9.x
IBM	JDBC 3.0 version 3.66.46 or JDBC 4.0 version 4.16.53	DB2 Universal JDBC driver	DB2	10.1
IBM	-	Type IV	Informix	-
Microsoft	4	Type II	MS SQL Server	2012
Oracle MySQL, MySQL Community	5.x	Type II, Type IV	MySQL	5.x
Oracle	RAC			

Oracle	9.x	Type II, Type IV	Oracle Database	8i+
Open Source PostgreSQL	42.2.5	Type IV	Postgres	8.x, 9.x, 11x
Sybase	jConnect	Type IV	Sybase	-
Teradata			Teradata	-

Notes:

- Type II is a C or OCI driver
- Type IV is a thin database client and is a pure Java driver

NoSQL/Data Grids/Cache Servers Support

The Java Agent supports these NoSQL, data grids and cache server environments. Some require additional configuration. Click the link on the database, data grid or cache name in the following support matrix for information about additional configuration required or related configuration topics.

Vendor	Database/Data Grid/Cache	Version	Correlation/Entry Points	JMX	Configuration Notes
Amazon	DynamoDB	-	Exit Points	-	See "Amazon Web Services" on Java Backend Detection .
Amazon	Simple Storage Service (S3)	-	-	-	"Amazon Simple Storage Service Backends" on Java Backend Detection .
Apache	Cassandra <ul style="list-style-type: none"> • DataStax drivers • Thrift drivers 	1.x, 2.x	Correlation for Thrift drivers only	Yes	<ul style="list-style-type: none"> • "Cassandra Backends" on Java Backend Detection. • For Cassandra server-side support, see Apache Cassandra Startup Settings
Apache	Lucene - Apache Solr	1.4.1	Entry Points	Yes	Apache Solr Startup Settings
JBoss	Cache TreeCache	-	-	-	JBoss Startup Settings
JBoss	Infinispan	5.3.0+	Correlation	-	-
Open Source	Memcached	-	-	-	Memcached Exit Points
Open Source	MongoDB	3.1	-	-	See "MongoDB Backends" on Java Backend Detection
Oracle	Coherence	3.7.1	Custom-Exit	Yes	Coherence Startup Settings
Red Hat	JBoss DataGrid	-	-	-	JBoss Startup Settings
	JBoss Cache TreeCache	-	-	-	
	JBoss Infinispan	5.3.0+	Correlation	-	
Terracotta	EhCache	-	-	-	EhCache Exit Points

RPC/Web Services API/HTTP Client Support

The Java Agent supports these RPC, web services or API framework types. Some require additional configuration as indicated in the Configuration Notes column.

Vendor	RPC/Web Services API Framework	Version	SOA Protocol-WebServices	Auto Naming	Correlation/Entry Points	Exit Points	Configurable BT Naming Properties	Detection	Configuration Notes
Apache	Apache CXF	2.1	JAX-WS	Yes	Yes	Yes	Yes	Yes	

Apache	Apache HTTP Client	-	HTTPClient (now in Apache HTTP Components)	Yes	Yes (correlation only)	Yes	-	Yes	See "HTTP Backends" on Java Backend Detection
Apache	Apache Async HTTP Client	4.1.x	-	-	-	-	-	-	-
Apache	Ribbon HTTP Client	2.1.0	HTTP Client	Yes	Yes (correlation) Entry - NA	Yes	NA	Yes	
Apache	Apache Thrift	-	-	Yes	Yes	Yes	Yes	Yes	Binary Remoting Entry Points for Apache Thrift
Eclipse	Jetty	8.x, 9.x	HTTP Client	Yes	Yes (correlation only)	Yes (ART supported)	-	Yes	See "HTTP Backends" on Java Backend Detection
IBM	WebSphere	6.x, 7.x, 8.x	JAX-RPC	-	-	-	-	-	IBM WebSphere and InfoSphere Startup Settings; also see Default configuration excludes WebSphere classes
IBM	Websphere	7.x, 8.x	IIOp	-	-	-	-	-	IBM WebSphere and InfoSphere Startup Settings; also see Default configuration excludes WebSphere classes
Open Source	java.net.Http	-	HTTP	Yes	-	Yes	Yes	Yes	See "HTTP Backends" on Java Backend Detection.
Open Source	HTTPClient	0.3-3	Oracle SOA (and potentially others that embed this library)	-	Correlation: Yes; Entry: No	Yes	-	Yes	Oracle WebLogic Startup Settings; also see Default configuration excludes WebLogic classes
Open Source	Grizzly	Grizzly Async HTTP Client (com.ning.http-client 1.8.x, 1.9.x, grizzly-http-client 1.1x) <ul style="list-style-type: none"> NingAsyncClient v1 with NettyProvider, GrizzlyProvider NingAsyncClient v2 with NettyProvider 	HTTP	-	Correlation: Yes; Entry: No	Yes	-	Yes	
Oracle	GlassFish Metro	-	JAX-WS	-	-	-	-	-	
Oracle	GlassFish Metro with Grails	-	JAX-WS	-	Yes	-	-	Not by Default	
Oracle	Oracle Application Server	ORMI	-	no	-	-	-	-	
Oracle	WebLogic	10.x	T3, IIOp	Yes	Correlation: Yes; Entry: No	Yes	-	Yes	
Oracle	WebLogic	9.x, 10.x	JAX-RPC	-	-	-	-	-	
Oracle/Sun	Sun RMI	-	IIOp	-	Not by Default	-	-	-	
Oracle/Sun	Sun RMI	-	JRMP	-	No	Yes	host/port	Yes	

Red Hat	JBoss A-MQ	4.x+	RMI	Yes	Yes	Yes	Yes	Yes	JBoss and Wildfly Startup Settings
Square	OkHttp								
Square	OkHttp	-	HTTP	Yes	Correlation: Yes Entry: No	Yes	-	Synchronous only	
-	Web Services	-	SOAP over HTTP	-	Yes	Yes	-	-	Create Match Rules for Web Services "Web Service Entry Points" on Java Backend Detection
jersey.github.io	Reactive JAX-RS client API	2.25+	HTTP Client	Yes	Yes (correlation) Entry – NA	Yes	NA	Yes	"Web Service Entry Points" on Java Backend Detection

Business Transaction Error Detection

The Java Agent supports the following logging frameworks for business transaction error detection:

- Apache Log4j and Log4j 2
- java.util.logging
- Simple Logging Facade for Java (SLF4J)
Support for the following method has been added: `public void error(String format, Object... argArray)`
- Logback

To instrument other types of loggers, see [Error Detection](#).

.NET Agent Support

Supported Runtime Environments

This section lists the environments where the .NET Agent does some automatic discovery after little or no configuration.

OS Versions

- Microsoft Windows Server 2008 (32-bit and 64-bit)
- Microsoft Windows Server 2008 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2016
- Microsoft Windows 7, 8, 8.1, 10

Microsoft .NET Frameworks

Microsoft .NET Framework versions 2.0, 3.0, 3.5, 4.0, 4.5, 4.5.2, 4.6, 4.7 on the following runtime environments:

- Microsoft IIS versions 6.0, 7.0, 7.5, 8.0, 8.5, 10
- Managed Windows Services
- Managed Standalone Applications
- Microsoft SharePoint 2010, 2013 as services running inside IIS
- Microsoft .NET Core 2.0/2.1 for Windows
- Microsoft .NET Core 2.2 for Windows is supported for .NET Agent versions 4.5.7 and later

Microsoft Windows Azure

- Azure App Services for .NET 4.6 environments in the Azure Portal
 - Web Apps
 - Web Jobs
 - API Apps
 - Container Services

For Azure App Services, the .NET Machine Agent disables certain .NET Machine Agent infrastructure monitoring features: CLR crash reporting, machine snapshots, and Windows performance counter monitoring.

- Azure Cloud Services
 - Web Roles
 - Worker Roles

Unsupported Frameworks

- Microsoft .NET versions 1.0, 1.1
- Unmanaged native code

Automatically Discovered Business Transactions

The .NET Agent discovers business transactions for the following frameworks by default. The agent enables detection without additional configuration.

Type	Custom Configuration Options?	Downstream Correlation?
ASP.NET*	Yes	Yes
ASP.NET MVC 2 ASP.NET MVC 3 ASP.NET MVC 4 ASP.NET MVC 5	Yes	Yes
ASP.NET Core on the full framework	Yes	Yes
Open Web Interface for .NET (OWIN) web API	Yes	Yes
.NET Remoting	No	See Enable Correlation for .NET Remoting .
Windows Communication Foundation (WCF)	No	Yes
Web Services including SOAP	No	Yes
Message Queues		
Apache ActiveMQ NMS framework and related MQs	No	Yes
IBM WebSphere MQ	No	Yes
Microsoft Message Queuing (MSMQ)	No	Yes
Microsoft Service Bus / Windows Azure Service Bus	No	Yes
NServiceBus over MSMQ or RabbitMQ transport	No	Yes
RabbitMQ	Yes	Yes
TIBCO Enterprise Message Service	No	Yes
TIBCO Rendezvous	No	Yes

Windows Azure Queue	No	No
---------------------	----	----

* The .NET Agent automatically discovers entry points for ASP.NET web forms with the Async property set to "true" in the [Page directive](#).

Supported Loggers for the .NET Agent

- Log4Net
- NLog
- System Trace
- Windows Event Log

If you are using a different logger, see [Error Detection](#).

Remote Service Detection

The .NET Agent automatically detects the following remote service types. The agent enables detection by default. You do not need to perform extra configuration.

Type	Custom Configuration Options?	Async Detection?*	Downstream Correlation?
Directory Services, including LDAP	No	No	N/A
HTTP	Yes	See Asynchronous Exit Points for .NET .	Yes
MongoDB: C# and .NET MongoDB Driver version 1.10, 2.0	No	See Asynchronous Exit Points for .NET .	N/A
.NET Remoting	Yes	No	See Enable Correlation for .NET Remoting .
WCF	Yes	See Asynchronous Exit Points for .NET .	Yes
WCF Data Services	Yes	No	No
Web Services, including SOAP	Yes	See Asynchronous Exit Points for .NET .	Yes
Azure Service Fabric Remoting v1 and v2—for the .NET Microservices Agent	-	-	-
Data Integration			
Microsoft BizTalk Server 2010, 2013	No	Yes	See Correlation Over Microsoft BizTalk .
Message Queues			
Apache ActiveMQ NMS framework and related MQs	Yes	No	Yes
IBM WebSphere MQ (IBM XMS)	Yes	No	Yes
Microsoft Message Queuing (MSMQ)	Yes	See MSMQ Backends for .NET	See MSMQ Backends for .NET
Microsoft Service Bus / Windows Azure Service Bus	No	Async exit points only	Yes
NServiceBus over MSMQ or RabbitMQ transport	No	See NServiceBus Backends for .NET	Yes
RabbitMQ	See RabbitMQ Backends for .NET	No	Yes
TIBCO Enterprise Message Service	Yes	No	Yes
TIBCO Rendezvous	Yes	No	Yes

Windows Azure Queue	No	No	No
---------------------	----	----	----

* The agent discovers asynchronous transactions for the Microsoft .NET 4.5 framework. See [Asynchronous Exit Points for .NET](#)

Supported Windows Azure Remote Services

Type	Customizable Configuration?	Downstream Correlation?
Azure Blob	No	No
Azure Queue	No	No
Microsoft Service Bus	No	Yes

Cache Clients

Type	Customizable Configuration?	Async Detection?*	AppD for Databases?
StackExchange.Redis	No	Yes	No

Data Storage Detection

The .NET Agent automatically detects the following data storage types. The agent enables detection by default. You do not need to perform extra configuration.

Type	Customizable Configuration?	Async Detection?*	AppD for Databases?
ADO.NET (see supported clients below)	Yes	Yes	No
Windows Azure Blob Storage	No	Yes	No
Windows Azure File Storage	No	Yes	No
Windows Azure Table Storage	No	Yes	No

* The agent discovers asynchronous transactions for the Microsoft .NET 4.5 framework. See [Asynchronous Exit Points for .NET](#)

Supported ADO.NET Clients

AppDynamics can monitor any ADO.NET client version and type. Clients we've tested include the following:

Database Name	Database Version	Client Type
Oracle	10, 11, 12	ODP.NET
Oracle	10, 11, 12	Microsoft Provider for Oracle
MySQL	5.x	Connector/Net and ADO.NET
Microsoft SQL Server*	2005, 2008, 2012	ADO.NET

* *Microsoft*, *SQL Server*, and *Windows* are registered trademarks of Microsoft Corporation in the United States and other countries.

Node.js Agent Support

Node.js Versions

- The 4.5.11 Node.js agent supports Node.js v0.8, v0.10, v0.12, v4, v5, v6, v7, v8, v9, v10, and v11.
- The 4.5.12+ Node.js agent supports Node.js v6, v7, v8, v9, v10, and v11. (v8, v9, v10, v11 on Mac).

For agent versions 4.5.12 and on the npm install will stop and print a message for the two scenarios listed:

- nodejs version less than 6
 - This version of AppDynamics agent supports Node.js versions 6.0 and above.
For older versions of Node.js use the the AppDynamics agent 4.5.11 by installing with 'npm install appdynamics@4.5.11'
- nodejs less than 8 on mac
 - This version of AppDynamics agent on Mac OS supports Node.js versions 8.0 and above.
For older versions of Node.js use the the AppDynamics agent 4.5.11 by installing with 'npm install appdynamics@4.5.11'

Transaction Naming

Entry Type	Default Transaction Naming
Node.js Web	URI

HTTP Exit Points

Supported HTTP Exit Points
Node.js HTTP client library

See <http://nodejs.org/api/http.html> for information about the Node.js HTTP client library.

Database Exit Points

- MongoDB
- MySQL
- PostgreSQL
- Riak - will be displayed as HTTP backends
- Couchbase
- DynamoDB using the official AWS SDK driver
- Cassandra

Cache Exit points

- Memcached
- Redis

PHP Agent Support

PHP Versions

The PHP agent supports PHP 5.6, 7.0, 7.1, 7.2, and 7.3.

The PHP agent *does not*:

- Monitor PHP applications in Zend Thread Safety (ZTS) mode. If you are using ZTS, AppDynamics suggests that you review your dependencies on ZTS to confirm that you actually need it, and if you do not, to switch to non-ZTS mode
- Support Zend Monitor
- Officially support plugins that encrypt and/or obfuscate PHP code, such as Zend Guard or ionCUBE Loader

PHP Web Servers

- Apache 2.2 and 2.4 in the following modes:
 - `prefork` mode using `mod_php`
 - worker MPM mode using `mod_fastcgi` with `php-fpm` or `mod_fcgid` with `php-cgi`
- Any web server compatible with `php-fpm`.

Operating Systems

- Any Linux distribution based on glibc 2.5+.
- Mac OS X 10.9+

PHP Frameworks and Protocols

Framework/Protocol	Version	Entry Point Type
Drupal	7, 8	Drupal
WordPress	3.4+, 4.x, 5.x	Wordpress
Zend	1, 2	PHP MVC
CodeIgniter	2.x, 3.x	PHP MVC
FuelPHP	1.5x, 1.6x	PHP MVC
Magento	1.5, 1.6, 1.7	PHP MVC
Symfony	1, 2	PHP MVC
CakePHP	2.x, 3.x	PHP MVC
Laravel	5.7	PHP MVC
HTTP		PHP Web
CLI		PHP CLI

If your PHP framework is not listed here, the agent detects your entry points as PHP Web and names the business transactions based on the first two segments of the URI — the default naming convention for PHP Web transactions. So it is still possible to monitor applications on *unsupported* frameworks. Laravel BTs are detected as symfony, as laravel itself is built on top of symfony.

Transaction Naming

Framework/Environment	Default Transaction Naming
Drupal	page callback name
Wordpress	template name
PHP MVC Frameworks	controller:action
PHP Modular MVC Frameworks	module:controller:action
PHP Web	URI
PHP Web Service	service name.operation name
PHP CLI	last two segments of the script's directory path plus the name of the script

Virtual host prefixing is available for all supported entry point types except PHP CLI.

PaaS Providers

PaaS Provider	Buildpack
Pivotal Cloud Foundry	https://github.com/Appdynamics/php-buildpack See http://docs.pivotal.io/appdynamics/index.html for information about integration with PCF.

Exit Points

Supported HTTP Exit Points
<code>curl/curl-multi*</code>
<code>drupal_http_request()</code>
<code>fopen(), file_get_contents()</code>
<code>Zend_HTTP_Client::request()</code>

*The total time reported for a `curl/multi_curl` request in the Controller is the same as reported by the function `curl_getinfo`. Also, we report the the following execution metrics in the exit call details for the `curl/multi_curl` request which are included in the total time:

- `namelookup_time`
- `connect_time`
- `pretransfer_time`
- `redirect_time`

Supported Database Exit Points
MySQL old native driver (removed for PHP 7)
MySQLi Extension*
OCI8
PDO
PostgreSQL accessed via PDO and pgsq extensions

*`mysqli_multi_query` is not supported.

Supported Cache Exit Points
Memcache
Memcached
Predis 0.8.5 and 1.1.1, on PHP versions 5.6 and higher
Phredis 4.1

Although Predis is a full PHP client library, the PHP Agent supports Predis as an exit point only, not as an entry point.

Supported Web Service Exit Points
PHP SOAPClient
NuSOAP 0.9.5

Supported Message Queue Exit Points
RabbitMQ

RabbitMQ support requires the [amqp extension](#).

Opcode Cache Compatibility

Alternative PHP Cache (APC)	<input type="checkbox"/>
-----------------------------	--------------------------

Python Agent Support

Python Versions

- The Python agent supports CPython 2.6, 2.7, 3.4, 3.5, 3.6, and 3.7.

Operating Systems

- Any Linux distribution based on glibc 2.5+
- Mac OS X 10.8+

Python Frameworks and Protocols

Framework/Protocol	Version	Entry Point Type	Default Transaction Naming
WSGI	1.0	Python Web	First two segments of URI
Tornado	3.2 - 4.5	Python Web	First two segments of URI

AppDynamics has tested the Python Agent on Tornado, Django, Flask, CherryPy, Bottle, and Pyramid.

You can configure the agent to instrument any WSGI-based application or framework as Python Web, including but not limited to those listed below.

At present, the Python agent fully supports exception detection in Django, Flask, CherryPy, Bottle, Pyramid, and Tornado frameworks.

Other WSGI frameworks and custom WSGI applications may install exception handlers that effectively hide some exceptions from the agent. In such cases, the agent will only detect exceptions during exit calls, uncaught exceptions which are propagated to the WSGI server, and exceptions reported via the custom business transaction API.

WSGI-Based Frameworks	Notes
Bottle	
CherryPy	
Django	
Flask	
PasteDeploy	
Pyramid	

Database Exit Points

Supported Database Exit Points	Version
cx_Oracle	5.1.x
MongoDB	3.1+
MySQL-Python	
mysqlclient	
MySQL Connector/Python	
Psycopg 2	
PyMySql	
TorMySql	

HTTP Exit Points

Supported HTTP Exit Points
httplib*
httplib2
requests
urllib
urllib2
urllib3
tornado.httpclient

* The agent detects calls to any external library built on top of httplib. Therefore, backend calls to such services, such as boto, dropbox, python-twitter, and so on are detected and displayed as HTTP exit calls.

Cache Exit Points

Supported Cache Exit Points

Memcache

Redis-py

Apache Server Agent Support

Apache Web Servers

Supported Apache Web Server Version

- Apache HTTP Server 2.2.x
- Apache HTTP Server 2.4.x
- IBM HTTP Server 7.0 +
- Oracle HTTP Server 11g+

Operating Systems

- Any Linux distribution based on glibc 2.5+

Architectures

- 32-bit
- 64-bit

Automatically Discovered Business Transactions

The Apache Agent automatically discovers the following business transactions:

Type	Custom Configuration Options	Downstream Correlation
Web (HTTP)	Yes	Yes

By default the agent excludes requests for the following static file types:

bmp
cab
class
conf
css
doc
gif
ico
jar
jpeg
jpg
js
mov
mp3
mp4
pdf
png

pps
properties
swf
tif
txt
zip

Remote Service Detection

Apache Modules

The Apache Agent automatically detects loaded Apache modules as remote services. The agent excludes a list of common modules from detection.

▼ [Show the list of excluded modules...](#)

- core.c
- http_core.c
- mod_access_compat.c
- mod_actions.c
- mod_alias.c
- mod_allowmethods.c
- mod_appdynamics.cpp
- mod_auth_basic.c
- mod_auth_digest.c
- mod_authn_alias.c
- mod_authn_anon.c
- mod_authn_core.c
- mod_authn_dbd.c
- mod_authn_dbm.c
- mod_authn_default.c
- mod_authn_file.c
- mod_authn_socache.c
- mod_authnz_ldap.c
- mod_authz_core.c
- mod_authz_dbd.c
- mod_authz_dbm.c
- mod_authz_default.c
- mod_authz_groupfile.c
- mod_authz_host.c
- mod_authz_owner.c
- mod_authz_user.c
- mod_autoindex.c
- mod_cache.c

mod_cache_disk.c
mod_cgi.c

mod_data.c

mod_dbd.c

mod_deflate.c

mod_dir.c

mod_disk_cache.c

mod_dumpio.c

mod_echo.c

mod_env.c

mod_expires.c

mod_ext_filter.c

mod_file_cache.c

mod_filter.c

mod_headers.c

mod_include.c

mod_info.c

mod_lbmethod_bybusyness.c

mod_lbmethod_byrequests.c

mod_lbmethod_bytraffic.c

mod_lbmethod_heartbeat.c

mod_log_config.c

mod_logio.c

mod_lua.c

mod_mem_cache.c

mod_mime.c

mod_mime_magic.c
mod_negotiation.c

mod_perl.c

mod_python.c

mod_remoteip.c

mod_reqtimeout.c

mod_rewrite.c

mod_setenvif.c

mod_slotmem_plain.c

mod_slotmem_shm.c

mod_so.c

mod_socache_dbm.c

mod_socache_memcache.c

mod_socache_shmcb.c
mod_speling.c
mod_ssl.c
mod_status.c
mod_substitute.c
mod_suexec.c
mod_systemd.c
mod_unique_id.c
mod_unixd.c
mod_userdir.c
mod_usertrack.c
mod_version.c
mod_vhost_alias.c
prefork.c
util_ldap.c

For End User Monitoring, the Apache Agent does not support automatic injection of the Javascript `adrum` header and footer to instrument web pages.

C/C++ Agent Supported Platforms

Operating Systems

- Any Linux distribution based on glibc 2.5+
- Windows Server 2008 R2 and above
 - Visual Studio 2015, or later

Go Agent Support

Operating Systems

- Any Linux distribution based on glibc 2.5+.

IIB Agent Support

IIB Versions

The IIB agent supports WMB v8, IIB v9, v10.

Operating Systems

The IIB agent supports the following operating systems:

- Linux x86-64 bit

IIB Node Types

The agent can continue business transactions detected upstream at the following node types:

- SOAPInput
- HTTPInput
- JMSInput
- MQInput

The agent can detect and tag exit calls for downstream correlation at the following node types:

- SOAPRequest
- HTTPRequest
- JMSOutput, JMSReply
- MQOutput, MQReply

For MQ, we use the MQRFH2 message header to provide correlation. Any applications consuming MQ messages from IIB with IIB agent must support the MQRFH2 header.

The agent can detect database backend calls for the following node types:

- DatabaseRetrieve
- DatabaseRoute

All nodes are represented within AppDynamics Business Transactions as Threads. The per node timings can be seen in the tree view of the Business Transaction dashboard, and in transaction snapshots.