



Deprecated API

This appendix lists the APIs, fields, and methods that have been deprecated. A deprecated API is not recommended for use, generally due to improvements, and a replacement API is usually given. Deprecated APIs may be removed in future implementations.

- [Deprecated Interfaces, on page 1](#)
- [Deprecated Fields, on page 1](#)
- [Deprecated Methods, on page 2](#)

Deprecated Interfaces

`com.cisco.jtapi.extensions.CiscoRouteAddress`

This interface has not been implemented.

Deprecated Fields

Deprecated fields

`com.cisco.jtapi.extensions.CiscoAddress.APPLICATION_CONTROLLED_RECORDING`

`com.cisco.jtapi.extensions.CiscoAddress.DEVICE_CONTROLLED_RECORDING`

These constants are deprecated. Applications that upgrade to Release 9.0 or later releases should use the new `SELECTIVE_RECORDING` constant and not the deprecated `APPLICATION_CONTROLLED_RECORDING` and `DEVICE_CONTROLLED_RECORDING` constants. In Release 9.0 or later releases Unified CM and JTAPI never return the `DEVICE_CONTROLLED_RECORDING` constant.

`com.cisco.jtapi.extensions.CiscoProviderCapabilityChangedEv.MODIFY_CGPN`

This constant is not returned by any interface, should not be used by application.

`com.cisco.jtapi.extensions.CiscoProviderCapabilityChangedEv.MONITOR_PARKDN`

This constant is not returned by any interface, should not be used by application.

`com.cisco.jtapi.extensions.CiscoProvCallParkEv.REASON_CALLPARKREMAINDER`

This interface is deprecated due to a spelling error. Use the new interface `REASON_CALLPARKREMINDER`.

Deprecated fields

`com.cisco.jtapi.extensions.CiscoFeatureReason.REASON_PARKREMAINDER`

Use `REASON_PARKREMINDER`.

`com.cisco.jtapi.extensions.CiscoProviderCapabilityChangedEv.SUPERPROVIDER`

This constant is not returned by any interface, should not be used by application.

Deprecated Methods

Deprecated methods

`com.cisco.jtapi.extensions.CiscoTermDataEv.getData()`

Use `byte[] getTermData`

`com.cisco.jtapi.extensions.CiscoJtapiException.getErrorDescription(int)`

Use `String getErrorDescription ()`; instead.

`com.cisco.jtapi.extensions.CiscoJtapiException.getErrorName(int)`

Use `String getErrorName ()`; instead.

`com.cisco.jtapi.extensions.CiscoConsultCallActiveEv.getHeldTerminalConnection()`

Replaced by `CiscoConsultCall.getConsultingTerminalConnection()`

`com.cisco.jtapi.extensions.CiscoCall.getLastRedirectingPartyInfo()`

- use `getLastRedirectedPartyInfo()`;

`com.cisco.jtapi.extensions.CiscoAddress.getRegistrationState()`

This method has been replaced by the `getState()` method.

`com.cisco.jtapi.extensions.CiscoTerminal.getRegistrationState()`

This method has been replaced by the `getState()` method.

`com.cisco.jtapi.extensions.CiscoMediaTerminal.register(InetAddress, int)`

`com.cisco.jtapi.extensions.CiscoTerminal.sendData(String)`

`com.cisco.jtapi.extensions.CiscoJtapiProperties.setSecurityPropertyForInstance(String, String, String, String, String, String, String, String, boolean)`

This method is replaced by overloaded method `setSecurityPropertyForInstance` which takes an extra parameter `certStorePassphrase`, a passphrase for java key store. This method might have some security vulnerability.

`com.cisco.services.tracing.TraceManager.setSubFacilities(String[])`

and replaced with `TraceManager.addSubFacilities` method

Deprecated methods

`com.cisco.services.tracing.implementation.TraceManagerImpl.setSubFacilities(String[])`

replaced by `addSubFacilities(String[])`

`com.cisco.services.tracing.TraceManager.setSubFacility(String)`

and replaced with `TraceManager.addSubFacility` method

`com.cisco.services.tracing.implementation.TraceManagerImpl.setSubFacility(String)`

replaced by `addSubFacility(String)`

`com.cisco.jtapi.extensions.CiscoJtapiProperties.updateCertificate(String, String, String, String, String, String, String, String)`

This method is replace by overloaded method `updateCertificate` which takes an extra parameter `certStorePassphrase`, a passphrase for java key store. This method might have some security vulnerability.

`com.cisco.jtapi.extensions.CiscoJtapiProperties.updateServerCertificate(String, String, String, String, String)`

This method is replace by overloaded method `updateServerCertificate` which takes an extra parameter `certStorePassphrase`, a passphrase for java key store. This method might have some security vulnerability.

