



Jabra Integration Service v1.3



Table of Contents

- 1 Introduction 4
- 2 Requirements..... 5
 - 2.1 Client platform support..... 5
 - 2.2 Client installer 5
 - 2.3 Lync/Skype for Business 6
 - 2.4 Jabra device SDK..... 6
- 3 Reporting data..... 7
 - 3.1 Data format 7
 - 3.2 Data example 9
- 4 Call status events..... 12
 - 4.1 Aggregated call data..... 12
- 5 Firmware update and settings configuration..... 13
 - 5.1 API documentation..... 13
 - 5.1.1 SettingsDefinition-Jabra.xml 14
 - 5.1.2 Mapping.xml..... 14
 - 5.1.3 Region and language selection..... 15
 - 5.1.4 Configuration of device events 15
- 6 Deployment and backend configuration..... 16
 - 6.1 Deployment for default backend 16
 - 6.2 Deployment in a PowerSuite environment..... 16
 - 6.3 Deployment in an OVOC environment..... 17
- 7 Log files for troubleshooting 18
- 8 Jabra resources..... 19
 - 8.1 Retrieving resources..... 20
 - 8.2 Resource-mapping.xml..... 20
 - 8.2.1 Variants 20
 - 8.3 Firmware file download 21
 - 8.3.1 ClientId 21
 - 8.4 Product image download 21
 - 8.5 Using the resources..... 21
- 9 FAQ..... 22
- 10 Supported devices..... 24



1 Introduction

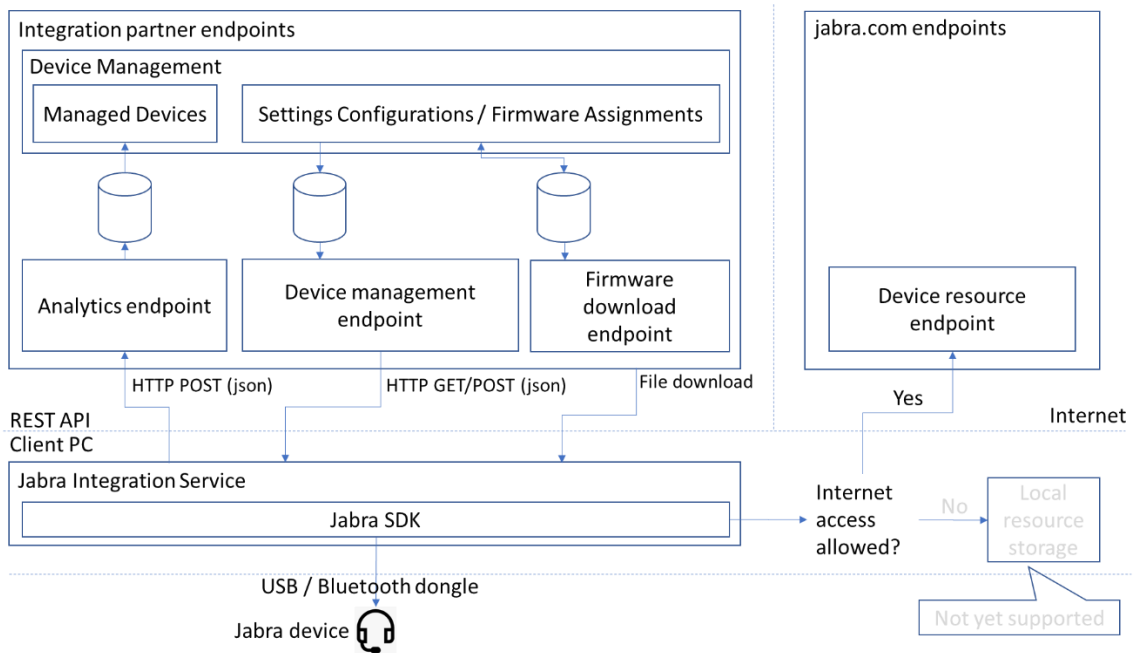
The purpose of this document is to provide an overview of the Jabra Integration Service solution.

The target audience for this document is Jabra integration partners and IT administrators responsible for Jabra device management.

The Jabra Integration Service is a Windows Desktop .NET application (the client) that provides the following high-level functionality.

- Jabra device and system information event logging
- Jabra device firmware update
- Jabra device settings configuration

The integration partner should provide the device management backend system and provide the necessary JSON documents (configurations) for the client. The client connects to an analytics endpoint where it posts Jabra device and system information. Furthermore it connects to the device management endpoint where it gets configurations for the managed devices connected to the client PC. Both endpoint URLs are partner specific and shall be configured during installation of the client application (using MSI properties). The endpoints shall provide a RESTful API as described in the following sections.





2 Requirements

2.1 Client platform support

- 32 bit Windows 7 or newer
- 64 bit Windows 7 or newer

2.2 Client installer

- Windows Installer (.MSI file based)
- Non-UI-installer
- Customized with MSI Properties

With these requirements implemented, this software component can be mass deployed/upgraded/uninstalled using Microsoft SCCM or similar tools.

The following can be customized:

- Analytics backend endpoint (a URL) – default is empty meaning that analytics is disabled. Can be either <http://> or <https://>. Installer property: ANALYTICSENDPOINT
- Device management backend endpoint (a URL) – default is empty meaning that device management is disabled. Can be either <http://> or <https://>. Installer property: DEVICEMANAGEMENTENDPOINT
- Show/Hide tray icon – default is show (YES). Installer property: TRAYICON
- Check for updates interval in minutes – default is 10080 (7 days). Installer property: CHECKFORUPDATESINTERVALMINUTES
- Partner specific backend type. Default is (Default). Installer property: BACKENDTYPE
- Partner specific backend configuration options. Default is empty. Installer property: BACKENDCONFIGURATION

Example. Hide trayicon + set the analytics and device management endpoints to <https://myserver/api/myendpoint> + set the check for updates interval to 1 hour

```
msiexec.exe /i JabraIntegrationServiceX64Setup.msi TRAYICON="NO"  
ANALYTICSENDPOINT="https://myserver/api/myendpoint"  
DEVICEMANAGEMENTENDPOINT="https://myserver/api/myendpoint"  
CHECKFORUPDATESINTERVALMINUTES="60"
```

Pre-requisites:

- .NET 4.5.1 or newer – due to the Jabra SDK. If this is not fulfilled the installation will fail with an error message describing this.

Installation directory and executable:

- On a 64 bit machine the client is installed in the "C:\Program Files (x86)\Jabra\Jabra Integration Service" directory



- The client executable is called `JabraLoggingClient.exe`

2.3 Lync/Skype for Business

This software integrates with Lync/Skype for Business using the Microsoft Lync 2013 SDK¹. This SDK support Lync 2013 or newer (including Skype for Business). The software can run even if Skype for Business is not installed/running.

2.4 Jabra device SDK

This client uses the Jabra SDK² as a library for Jabra device detection, firmware update and device settings configuration. Currently the Jabra SDK requires internet access to retrieve device meta data and device resources.

¹ <https://www.microsoft.com/en-us/download/details.aspx?id=36824>

² <https://developer.jabra.com/site/global/sdks/windows/index.gsp>



3 Reporting data

One of the purposes of this component is to send data (http or https post) to a backend endpoint after certain events.

The data is:

- Posted when a device is attached (or the client is started)
 - Event = "DeviceAdded"
- Posted when a device is removed
 - Event = "DeviceRemoved"
- Posted when a call is started and again when the call is ended
 - Event = "CallStatus"
- Posted when a Lync/SfB conversation is started
 - Event = "ConversationStarted"

The data is formatted as JSON. The data is a HTTP POST.

Example:

https://requestb.in
POST /tkq90jtk

<> application/json
📎 477 bytes

8m ago
From 213.198.83.4,
162.158.90.73

FORM/POST PARAMETERS	HEADERS
None	Connect-Time: 1 Via: 1.1 vegur X-Request-Id: aaf0aeaa-066f-4550-a08b-20e4699e9f49 Host: requestb.in Total-Route-Time: 0 Content-Type: application/json User-Agent: JabraLogging Cf-Visitor: {"scheme":"https"} Cf-Ray: 3cd131abd80126ba-FRA Accept-Encoding: gzip Cf-Connecting-Ip: 213.198.83.4 Connection: close Content-Length: 477 Cf-Ipcountry: DK

RAW BODY

```
{"Event": "DeviceAdded", "UTCDateTime": "12/14/2017 12:33", "MachineName": "CPHMFREDERIKSE3", "UserName": "mfederiksen", "UserDomainName": "C"}
```

3.1 Data format

The Microsoft Lync 2013 SDK provides these data:

- Conversation Id

The Jabra SDK provides these data:

- A list of attached Jabra devices:
 - Device name



- Firmware version
- Electronic serial number
- For a wireless device the following is reported:
 - Battery level (0-100)
 - Battery low (true/false)
 - Battery charging (true/false)
- Basic call status (for all supported Jabra devices)
 - Call started
 - Call ended
- Advanced call status with aggregated call data (Jabra Engage 50)
 - Average audio exposure during call (measured in dB)
 - Maximum audio exposure during call (measured in dB)
 - Average background noise during call (measured in dB)
 - Maximum audio exposure during call (measured in dB)
 - Number of times boom arm was badly positioned during call

The client provides these data:

- Computer name³
- Domain/User⁴⁵
- Windows details⁶
- OS version⁷
- List of PC IP addresses⁸
- Jabra Integration Service version number

³ [https://msdn.microsoft.com/en-us/library/system.environment.machinename\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.environment.machinename(v=vs.110).aspx)

⁴ [https://msdn.microsoft.com/en-us/library/system.environment.username\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.environment.username(v=vs.110).aspx)

⁵ [https://msdn.microsoft.com/en-us/library/system.environment.userdomainname\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.environment.userdomainname(v=vs.110).aspx)

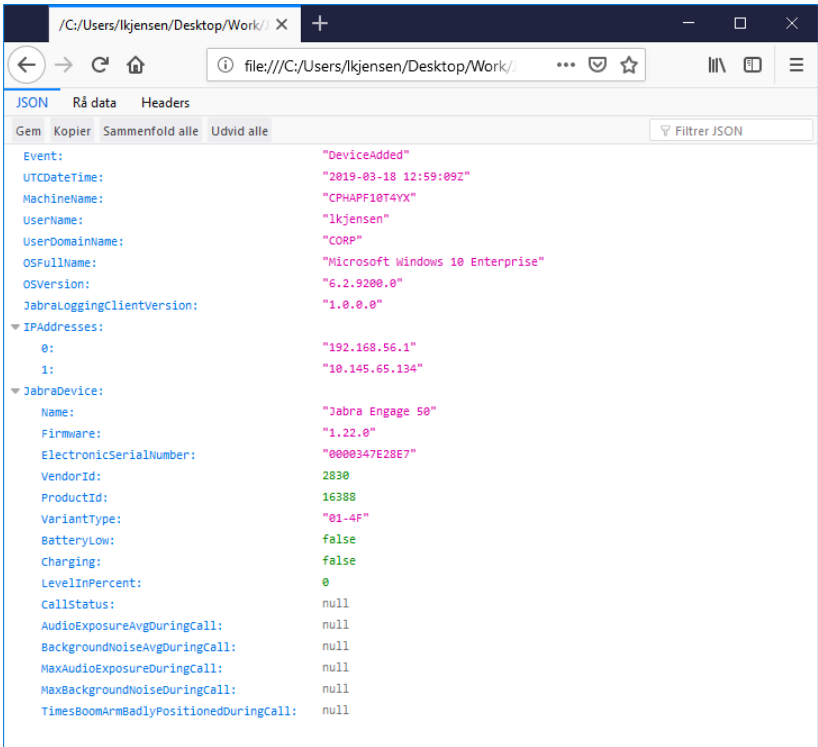
⁶ <https://stackoverflow.com/a/39888998/600559>

⁷ [https://msdn.microsoft.com/en-us/library/system.environment.osversion\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.environment.osversion(v=vs.110).aspx)

⁸ <https://stackoverflow.com/a/21155473/600559>

3.2 Data example

DeviceAdded

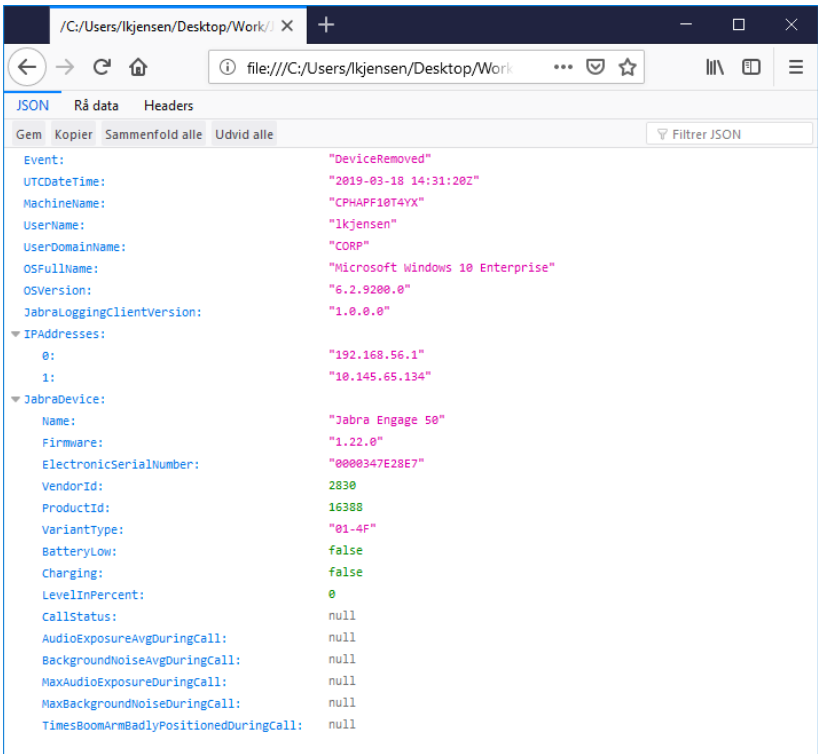


```

{
  "Event": "DeviceAdded",
  "UTCDateTime": "2019-03-18 12:59:09Z",
  "MachineName": "CPHAPF10T4YX",
  "UserName": "lkjensen",
  "UserDomainName": "CORP",
  "OSFullName": "Microsoft Windows 10 Enterprise",
  "OSVersion": "6.2.9200.0",
  "JabraLoggingClientVersion": "1.0.0.0",
  "IPAddresses": [
    {
      "0": "192.168.56.1",
      "1": "10.145.65.134"
    }
  ],
  "JabraDevice": {
    "Name": "Jabra Engage 50",
    "Firmware": "1.22.0",
    "ElectronicSerialNumber": "0000347E28E7",
    "VendorId": 2830,
    "ProductId": 16388,
    "VariantType": "01-4F",
    "BatteryLow": false,
    "Charging": false,
    "LevelInPercent": 0,
    "CallStatus": null,
    "AudioExposureAvgDuringCall": null,
    "BackgroundNoiseAvgDuringCall": null,
    "MaxAudioExposureDuringCall": null,
    "MaxBackgroundNoiseDuringCall": null,
    "TimesBoomArmBadlyPositionedDuringCall": null
  }
}

```

DeviceRemoved



```

{
  "Event": "DeviceRemoved",
  "UTCDateTime": "2019-03-18 14:31:20Z",
  "MachineName": "CPHAPF10T4YX",
  "UserName": "lkjensen",
  "UserDomainName": "CORP",
  "OSFullName": "Microsoft Windows 10 Enterprise",
  "OSVersion": "6.2.9200.0",
  "JabraLoggingClientVersion": "1.0.0.0",
  "IPAddresses": [
    {
      "0": "192.168.56.1",
      "1": "10.145.65.134"
    }
  ],
  "JabraDevice": {
    "Name": "Jabra Engage 50",
    "Firmware": "1.22.0",
    "ElectronicSerialNumber": "0000347E28E7",
    "VendorId": 2830,
    "ProductId": 16388,
    "VariantType": "01-4F",
    "BatteryLow": false,
    "Charging": false,
    "LevelInPercent": 0,
    "CallStatus": null,
    "AudioExposureAvgDuringCall": null,
    "BackgroundNoiseAvgDuringCall": null,
    "MaxAudioExposureDuringCall": null,
    "MaxBackgroundNoiseDuringCall": null,
    "TimesBoomArmBadlyPositionedDuringCall": null
  }
}

```



ConversationStarted

```
file:///C:/Users/lkjensen/Desktop/Work/JabraIn
JSON Rå data Headers
Gem Kopier Sammenfold alle Udvid alle Filter JSON
Event: "ConversationStarted"
UTCDateTime: "2019-03-18 12:49:26Z"
MachineName: "CPHAPP10T4YX"
UserName: "lkjensen"
UserDomainName: "CORP"
OSFullName: "Microsoft Windows 10 Enterprise"
OSVersion: "6.2.9200.0"
JabraLoggingClientVersion: "1.0.0.0"
IPAddresses:
  0: "192.168.56.1"
  1: "10.145.65.134"
Conversation:
  Id: "AdrdiQSe4u4PUISS3+ShhFlowCjuQ=="
  ActiveAudioDevice: "Speaker (2- Jabra Engage 50)"
  IsActiveAudioDeviceCertified: true
AudioDevices:
  0: "Speaker (2- Jabra Engage 50)"
  1: "Speakers (Jabra Link 370)"
  2: "Speaker/HP (Realtek High Definition Audio)/Microphone Array (Realtek High Definition Audio)"
```



CallStatus (CallStarted)

```
file:///C:/Users/lkjensen/Desktop/Work/...
JSON Rå data Headers
Gem Kopier Sammenfold alle Udvid alle Filtre JSON

Event: "Callstatus"
UTCDateTime: "2019-03-18 13:01:52Z"
MachineName: "CPHAPF10T4YX"
UserName: "lkjensen"
UserDomainName: "CORP"
OSFullName: "Microsoft Windows 10 Enterprise"
OSVersion: "6.2.9200.0"
JabraLoggingClientVersion: "1.0.0.0"
IPAddresses:
  0: "192.168.56.1"
  1: "10.145.65.134"
JabraDevice:
  Name: "Jabra Engage 50"
  Firmware: "1.22.0"
  ElectronicSerialNumber: "0000347E28E7"
  VendorId: 2830
  ProductId: 16388
  VariantType: "01-4F"
  BatteryLow: false
  Charging: false
  LevelInPercent: 0
  Callstatus: "CallStarted"
  AudioExposureAvgDuringCall: null
  BackgroundNoiseAvgDuringCall: null
  MaxAudioExposureDuringCall: null
  MaxBackgroundNoiseDuringCall: null
  TimesBoomArmBadlyPositionedDuringCall: null
```

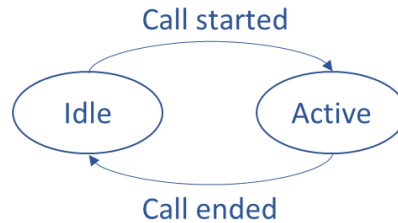
CallStatus (CallEnded)

```
file:///C:/Users/lkjensen/Desktop/Work/...
JSON Rå data Headers
Gem Kopier Sammenfold alle Udvid alle Filtre JSON

Event: "Callstatus"
UTCDateTime: "2019-03-18 13:02:20Z"
MachineName: "CPHAPF10T4YX"
UserName: "lkjensen"
UserDomainName: "CORP"
OSFullName: "Microsoft Windows 10 Enterprise"
OSVersion: "6.2.9200.0"
JabraLoggingClientVersion: "1.0.0.0"
IPAddresses:
  0: "192.168.56.1"
  1: "10.145.65.134"
JabraDevice:
  Name: "Jabra Engage 50"
  Firmware: "1.22.0"
  ElectronicSerialNumber: "0000347E28E7"
  VendorId: 2830
  ProductId: 16388
  VariantType: "01-4F"
  BatteryLow: false
  Charging: false
  LevelInPercent: 0
  Callstatus: "CallEnded"
  AudioExposureAvgDuringCall: 57
  BackgroundNoiseAvgDuringCall: 24
  MaxAudioExposureDuringCall: 71
  MaxBackgroundNoiseDuringCall: 46
  TimesBoomArmBadlyPositionedDuringCall: 2
```

4 Call status events

As described in the previous section the Jabra Integration Service posts call status events to the analytics endpoint when a call is started and again when the call is ended. The call status event reflects a state change in the Jabra device between an idle state and an active call state as illustrated below.



The state change can be triggered by a mobile phone call for Bluetooth enabled devices such as Jabra Engage 75 or Jabra Evolve 75 connected to the PC via a Jabra Link 370 adapter. It can also be triggered by a desk phone call or by any softphone that integrates with the Jabra device either through Jabra Direct or via standard HID telephony signaling as it is the case for Skype for Business.

It is important to note that the call status events are triggered by the device and that the payload included in an event only contains device specific information and general system information such Windows version, user name and IP addresses i.e. there is no information about the source of the call. Correlating call status events from a Jabra device to an actual softphone call, as an example, is a server side responsibility.

4.1 Aggregated call data

For Jabra Engage 50 it is possible to include aggregated call data at the end of each call. The Jabra Engage 50 is capable of sending a lot of events (background noise, audio exposure and badly positioned boom arm) during a call but in order to reduce network traffic and backend storage the Jabra Integration Service aggregates the data during the call and only sends one event at the end of the call. The format of the data is described in the previous section.



5 Firmware update and settings configuration

Once installed the client polls the device management endpoint for configuration updates. This is done whenever a Jabra device (headset) is connected or whenever the check for updates interval elapses.

The backend identifies the client's GET/POST request for the device management endpoint by the User-Agent string. An example is shown here:

```
"JabraClient/0.1 (Vendor=0B0E; Product=2465; Variant=01-3D; ESN=501AA5D6337F; Firmware=2.0.0)"
```

The individual devices are identified by their electronic serial number (ESN).

In response to the GET/POST request the endpoint returns a configuration (JSON document) containing the managed firmware version, a download URL for that firmware and a list of managed settings.

```
{
  "firmware": {
    "version": "2.0.0",
    "downloadUrl": "Firmware/Jabra-Evolve-75-2.0.0.zip"
  },
  "settings": {
    "AUDIO_PROTECTION": "5",
    "WIRELESS_RANGE": "3"
  }
}
```

The download URL can be relative as shown above or an absolute URL starting with either [http:](http://), [https:](https://) or [ftp:](ftp://).

Assuming that the device management endpoint is configured to `https://myserver/api/myendpoint` then the complete download URL in above example will be `https://myserver/api/myendpoint/Firmware/Jabra-Evolve-75-2.0.0.zip`

The "firmware" and "settings" elements are mandatory but they can be left empty indicating that there is only firmware or only settings to manage.

The firmware file is downloaded to the PC before the firmware update procedure is initiated. The device settings are not persisted on the PC as they are applied immediately.

5.1 API documentation

As part of the client release two documents provide the details of supported devices and supported settings.



5.1.1 SettingsDefinition-Jabra.xml

The SettingsDefinition-Jabra.xml file contains the complete list of device settings configuration options in the given release. An example entry is shown below.

```
<SettingDefinition Target="Jabra" Name="WIRELESS_RANGE"
UserFriendlyName="Wireless signal range" ValueType="String"
ValueConstraintType="Enumeration" DefaultValue="1" RequiresReboot="true"
AllowsNull="false">
  <Description>Select the wireless range between the headset and base. If
you experience signal interference from other wireless devices, select Medium or
Short.</Description>
  <UserFriendlyDescription>Select the wireless range between the headset
and base. If you experience signal interference from other wireless devices,
select Medium or Short.</UserFriendlyDescription>
  <SupportedValues>
    <SupportedValue Value="1" UserFriendlyValue="Long" />
    <SupportedValue Value="2" UserFriendlyValue="Medium" />
    <SupportedValue Value="3" UserFriendlyValue="Short" />
  </SupportedValues>
</SettingDefinition>
```

The "Name" of the setting is the unique identifier for a setting. The identifier along with the desired "Value" should be provided in the "settings" list of the configuration JSON as shown below.

```
"settings": {
  "WIRELESS_RANGE": "2"
}
```

The user friendly names are intended to be used by the integration partner device management system when presenting the configuration options in a UI. When creating the device management configurations it is important to provide the actual "Value" values and not the "UserFriendlyValue" values.

5.1.2 Mapping.xml

The mapping.xml file contains the complete list of devices supported by the given client release. The settings entry indicates which settings are supported by a given device. An example is shown below.

```
<device name="Jabra Evolve 30 II Stereo">
  <variants>
    <variant vid="2830" pid="42249" variantType="01-40" />
    <variant vid="2830" pid="786" variantType="01-40" />
    <variant vid="2830" pid="787" variantType="01-40" />
  </variants>
  <settings>
    <setting name="AUDIO_PROTECTION_1" />
    <setting name="RINGTONE_IN_HEADSET_SOFTPHONE" />
  </settings>
</device>
```

The Jabra USB vendor id (vid) is 2830 (0x0B0E), the device USB product id (pid) and the variant (variantType) are model specific. Variants of the same Jabra



product are listed under variants. When creating a device management UI it is not necessary to present any of the variants separately as they all share the same firmware and settings.

5.1.3 Region and language selection

Some settings are used to select the desired display language pack, voice announcements language pack or simply the desired language as these are (for some devices) binary files used during the firmware update process. For these settings to take any effect it is required to also provide a firmware file URL as part of the configuration JSON i.e. the firmware element of the JSON configuration cannot be empty. The relevant settings are listed below.

[SCREEN_REGION](#)

[SCREEN_REGION_1](#)

[VOICE_ANNOUNCEMENTS_LANGUAGE_1](#)

[VOICE_ANNOUNCEMENTS_LANGUAGE_2](#)

[VOICE_ANNOUNCEMENTS_REGION](#)

5.1.4 Configuration of device events

By default the CallStatus event is enabled for all devices i.e. every time a call is started and ended with a Jabra device, a CallStatus event is posted to the analytics endpoint. The CallStatus event can be disabled for individual devices by adding below setting to the settings JSON element.

```
"settings": {  
  "EVENT_CALL_STATUS": "0"  
}
```



6 Deployment and backend configuration

The Jabra Integration Service can be mass deployed/upgraded/uninstalled using Microsoft SCCM or similar tools. The default out-of-the-box deployment configures the client to run with the default behavior as described in the previous sections. For selected Jabra partners, however, the Jabra Integration Service provides specific customizations that slightly change the behavior of the client to accommodate easy integration with existing device management systems.

6.1 Deployment for default backend

The most basic usage of the Jabra Integration Service MSI package is to enable the analytics part by only providing an analytics endpoint as shown below.

```
msiexec.exe /i JabraIntegrationServiceX64Setup.msi TRAYICON="NO"  
ANALYTICSENDPOINT="https://myserver/api/myendpoint"
```

In this example the device management features (firmware update and device settings configuration) are completely disabled. This basic deployment method is useful for getting an overview of all Jabra devices and installed firmware versions within the company.

Adding a device management endpoint as shown below enables the company to push out the latest device firmware and to configure the devices as desired.

```
msiexec.exe /i JabraIntegrationServiceX64Setup.msi TRAYICON="NO"  
ANALYTICSENDPOINT="https://myserver/api/myendpoint"  
DEVICEMANAGEMENTENDPOINT="https://myserver/api/myendpoint"
```

6.2 Deployment in a PowerSuite environment

PowerSuite⁹ by Unify Square offers Jabra device management features such as device overview and firmware and device configuration management.

Deployment of Jabra Integration Service in a PowerSuite environment requires at least configuration of the analytics endpoint. This will enable the device management overview to be updated with known Jabra devices. To enable firmware update and device configuration the device management endpoint also needs to be configured as shown below.

```
msiexec.exe /i JabraIntegrationServiceX64Setup.msi TRAYICON="NO"  
ANALYTICSENDPOINT="https://myserver/DeviceMgmt"  
DEVICEMANAGEMENTENDPOINT="https://myserver/DeviceMgmt"
```

The actual endpoints are typically one of the **Listen Addresses** configured in PowerSuite under Configuration -> General -> Device Management

The MSI property `BACKENDTYPE` can be set to `PowerSuite` but this is not required as the PowerSuite behavior is identical to the default behavior.

⁹ <https://www.unifysquare.com/unified-communications-solutions>



6.3 Deployment in an OVOC environment

One Voice Operations Center (OVOC¹⁰) by AudioCodes offers Jabra device management features such as device overview and firmware and device configuration management.

Deployment of Jabra Integration Service in an OVOC environment only requires configuration of the device management endpoint. This will enable the Monitor -> Devices Status view to be updated with known Jabra devices as well as enable firmware update and device configuration. An example is shown below.

```
msiexec.exe /i JabraIntegrationServiceX64Setup.msi TRAYICON="NO"  
DEVICEMANAGEMENTENDPOINT="http://OVOC_IP/" BACKENDTYPE="Ovoc"  
CHECKFORUPDATESINTERVALMINUTES="10"
```

NOTE: The MSI property `BACKENDTYPE` must be configured to `Ovoc`.

¹⁰ <https://www.audiocodes.com/solutions-products/products/management-products-solutions/one-voice-operations-center>



7 Log files for troubleshooting

Log-files for the client are created in this folder:

```
C:\Users\\AppData\Roaming\Jabra\JabraIntegrationService
```

Log-files for the firmware updater are created in this folder:

```
C:\Users\\AppData\Local\Jabra\JabraCmdlineFwUpdater
```

When requesting Jabra support those log files are essential.



8 Jabra resources

The purpose of this section is to explain how to retrieve relevant Jabra resources when adding support for Jabra devices in a device management system.

The target audience for this section is Jabra integration partners, however it can also be useful for IT administrator.

The Jabra resources in scope

- Jabra firmware files
 - Type: .zip files containing an info.xml and relevant binary files
- Jabra firmware release notes
 - Type: string
- Jabra product images
 - Type: .png
- Jabra device identification information
 - Device name
 - Type: string
 - Vendor id (vid)
 - Type: integer
 - USB vendor id (as defined by usb.org)
 - 2830 (0x0B0E) for Jabra devices
 - Product id (pid)
 - Type: integer
 - USB product id (as defined by usb.org)
 - The value is specific to the device model, but can be shared between device variants of the same model. As an example an MS variant would have a dedicated pid
 - Variant type
 - Type: string
 - Jabra proprietary variant type
 - The first part of the string represents a device type. The relevant types are as follows
 - "00" = base (like Jabra Pro 9450 base)
 - "01" = headset (like Jabra Evolve 75)
 - "04" = dongle (Bluetooth adapter like Jabra Link 370)
 - "07" = USB (USB adapter like Jabra Link 860)
 - "08" = speaker phone (like Jabra Speak 710)
 - The second part represents a specific Jabra model variant within the scope of the device type e.g.
 - "01-03" is the mono variant of Jabra BIZ 2400 headset
 - "08-03" is a Jabra Speak 710 speaker phone

NOTE: vid, pid and variant type are all required in order to uniquely identify the specific model type/variant of a Jabra device.

8.1 Retrieving resources

In the following sections it is described how to retrieve resources from the Jabra backend. The recommendation and intended use is to download the relevant resources and then add/integrate those resources into the relevant device management system as depicted in the Jabra Integration Service system overview in section 1. For the time being this is recommended to be a manual step that should be considered whenever a new Jabra Integration Service version is released.

8.2 Resource-mapping.xml

In order to download the appropriate firmware file for a given Jabra device model, it is important to consult the **resource-mapping.xml** file (part of the Jabra Integration Service release package). This file lists all the supported Jabra devices and variants as well as relevant firmware download URLs and firmware release notes.

An example entry from the resource-mapping.xml is shown below.

```
<device name="Jabra Engage 50 Stereo"
imageUrl="https://devicecapabilities.jabra.com/v4/product/PID_4006/image?type=01-4E"
thumbnailUrl="https://devicecapabilities.jabra.com/v4/product/PID_4006/thumbnail80?type=01-4E">
  <variants>
    <variant vid="2830" pid="16385" variantType="01-4E" />
    <variant vid="2830" pid="16386" variantType="01-4E" />
    <variant vid="2830" pid="16390" variantType="01-4E" />
  </variants>
  <fwVersions>
    <fwVersion version="1.22.0"
firmwareFile="https://firmware.jabra.com/v3/download/4006/1.22.0">
      <releaseNote />
    </fwVersion>
    <fwVersion version="1.24.0"
firmwareFile="https://firmware.jabra.com/v3/download/4006/1.24.0">
      <releaseNote>• New feature: Stereo in conversation mode*
• New feature: Balanced voice
• Updated: Busylight function can be customized using Jabra SDK
• Updated: Sidetone adjustments
• Performance and stability improvements
*Dependent on softphone support
      </releaseNote>
    </fwVersion>
  </fwVersions>
</device>
```

8.2.1 Variants

The combination of Vendor Id (vid), Product Id (pid) and variant type uniquely identifies the type of device. In the example above the pids are different between variants. In other cases the variantType changes but the pids are the same.

All variants within a device element share the same resources, that is, the same name, the same images (above example is a stereo/duo headset) and the same firmware and firmware release notes.



8.3 Firmware file download

Jabra device firmware files can be downloaded through a Web API. There is a Swagger UI for testing purposes at this site

<https://firmware.jabra.com/swagger/index.html#!/Firmware/V4DownloadByProductIdByVersionGet>

However, the Swagger UI should NOT be used for actual file download. It is strongly recommended to use PowerShell, cURL or wget to download the firmware files. Below is an example on how to download a firmware file using PowerShell.

Step 1. download the firmware file to the `$download` variable. The highlighted URL below is obtained from the resource-mapping.xml.

```
PS C:\Temp>$download = Invoke-WebRequest -Uri
'https://firmware.jabra.com/v3/download/4006/1.24.0' -Headers
@{"accept"="application/octet-
stream";"ClientId"="C6raG/RknoKJ8PnFw6SLTc/AYv6yqzobKiWpq3yn6JQ="}
```

Step 2. write variable content to a file with the appropriate filename

```
PS C:\Temp>[System.IO.File]::WriteAllBytes((Get-Location -PSProvider
FileSystem).ProviderPath+"\">$download.headers["Content-
Disposition"].Split("=")[1],$download.content)
```

8.3.1 ClientId

The ClientId in the above PowerShell example is the Jabra SDK test ClientId also available from the Swagger site. It is possible, although not required, to request a dedicated Jabra SDK ClientId by registering at <https://developer.jabra.com/>

8.4 Product image download

Jabra product images are downloaded slightly different from firmware files and there is no need for the ClientId. However, the download URL is still obtained from the resource-mapping.xml file.

Step 1. download an image to a file with the corresponding product name

```
PS C:\Temp>Invoke-WebRequest
'https://devicecapabilities.jabra.com/v4/product/PID_4006/image?type=01-4E' -OutFile
Jabra_Engage_50_Stereo.png
```

8.5 Using the resources

The device management backend can obtain Jabra device identification info from the User-Agent string transmitted in the GET/POST requests. An example is shown here:

```
"JabraClient/0.1 (Vendor=0B0E; Product=2465; Variant=01-3D; ESN=501AA5D6337F;
Firmware=2.0.0)"
```

Based on the mapping.xml, the resource-mapping.xml and the User-Agent string the backend can provide the correct Jabra resources for the relevant device.



9 FAQ

Q: What is the software and what does it do?

A: With the software installed you are able to manage (change settings and firmware) Jabra devices. Moreover, the software provides data relevant for tracking call quality issues but also to give an overview of the USB devices that you have in your environment

Q: Does this have to be for all users or can the software be deployed on demand?

A: Deployment is controlled by your deployment software like SCCM

Q: Is this something which can be committed to long-term? E.g. no product withdrawals/changes planned

A: Yes, Jabra guarantees that the software will be maintained in a long-term perspective

Q: Who writes the software? How reliable/secure is it? How many updates? What happens if there is a bug?

A: Jabra develops the software that is installed locally on the PC and supports this like any other Jabra developed software. Bugs etc. are either reported to the integration partner or directly to Jabra Support. Currently we expect to release a new client with new product support 2-3 times a year. It is only necessary to update the client if the new products need to be managed

Q: Can it be easily packaged (low cost) for mass deployment?

A: Yes, it is a standard MSI package

Q: Can we remotely change the configuration?

A: Yes, for device firmware update and settings configuration this can be changed remotely. Configuration of the client software itself happens during the installation using MSI properties.

Q: How much disk space, CPU load, RAM is required?

A: ~10MB on disk, ~20MB RAM and ~0% CPU load when idle and very little load when a device is connected/disconnected or updated

Q: Is there any visible pop-up or notification to the users?



A: Yes, there is a UI pop-up but only before a firmware update is initiated as it is a critical operation requiring the attention of the end user – change of device settings happens instantly with no user interaction

Q: Why is there no ConversationEnded event for Skype for Business?

A: The Skype for Business desktop client SDK does not provide the ConversationId when a conversation is ended. The recommendation is to monitor the CallStatus device event instead and correlate that event to a Skype for Business call (or any other call) on the server side. The Skype for Business ConversationAdded event is still supported for backwards compatibility reasons.

10 Supported devices

Jabra BIZ 1100 Mono
Jabra BIZ 1100 Stereo
Jabra BIZ 1500 USB Duo
Jabra BIZ 1500 USB Mono
Jabra BIZ 2300 USB Duo
Jabra BIZ 2300 USB Mono
Jabra BIZ 2400 II CC USB Mono
Jabra BIZ 2400 II CC USB Stereo
Jabra BIZ 2400 II Duo
Jabra BIZ 2400 II Mono
Jabra BIZ 2400 II USB Duo BT
Jabra BIZ 2400 II USB Mono BT
Jabra BIZ 2400 II USB Mono CC
Jabra BIZ 2400 USB
Jabra BIZ 2400 USB Duo
Jabra BIZ 2400 USB Mono
Jabra Engage 50 Mono
Jabra Engage 50 Stereo
Jabra Engage 65 Convertible
Jabra Engage 65 Mono
Jabra Engage 65 Stereo
Jabra Engage 75 Convertible
Jabra Engage 75 Mono
Jabra Engage 75 Stereo
Jabra Evolve 20 Mono
Jabra Evolve 20 Stereo
Jabra Evolve 20SE Mono
Jabra Evolve 20SE Stereo
Jabra Evolve 30 II Mono
Jabra Evolve 30 II Stereo
Jabra Evolve 30 Mono
Jabra Evolve 30 Stereo
Jabra Evolve 40/80
Jabra Evolve 65 Mono
Jabra Evolve 65 Stereo
Jabra Evolve 65t
Jabra Evolve 75
Jabra Evolve 75e
Jabra LINK 230
Jabra LINK 260
Jabra LINK 265
Jabra LINK 280
Jabra LINK 30
Jabra LINK 32
Jabra LINK 33
Jabra LINK 360
Jabra LINK 370
Jabra LINK 43



Jabra LINK 44
Jabra LINK 850
Jabra LINK 860
Jabra MOTION UC
Jabra Pro 920
Jabra Pro 930
Jabra Pro 9450
Jabra Pro 9460
Jabra Pro 9465
Jabra Pro 9465/9470
Jabra Pro 9470
Jabra SPEAK 410
Jabra SPEAK 510
Jabra SPEAK 710
Jabra SPEAK 810
Jabra STEALTH UC
Jabra SUPREME UC
Jabra UC VOICE 150a Duo
Jabra UC VOICE 150a Mono
Jabra UC VOICE 150a MS Duo
Jabra UC VOICE 150a MS Mono
Jabra UC VOICE 250a
Jabra UC VOICE 250a MS
Jabra UC VOICE 550a Duo
Jabra UC VOICE 550a Mono
Jabra UC VOICE 550a MS Duo
Jabra UC VOICE 550a MS Mono
Jabra UC VOICE 750a Duo
Jabra UC VOICE 750a Mono
Jabra UC VOICE 750a MS Duo
Jabra UC VOICE 750a MS Mono