Overview

= Filter

Guides

UI Guide

Reference

Samples

Design & Quality

War das hilfreich?

山 切

Enhance your apps with 5G Build client-server applications

with gRPC

- Transferring data without draining the battery
- Reduce network battery drain
- Transfer data using Sync Adapters
- Bluetooth
- ▼ Bluetooth Low Energy

Overview

Find BLE devices

Connect to a GATT server

Transfer BLE data

- ▶ NFC
- Telecom
- Wi-Fi
- USB
- **UWB**
- VPN
- overview Open Mobile API reader support
- Renderscript

Session initiation protocol

Android Developers > Docs > Guides

Bluetooth Low Energy --

The basics

Auf dieser Seite

Key terms and concepts Roles and responsibilities

Android provides built-in platform support for Bluetooth Low Energy (BLE) in the central role and provides APIs that

Common use cases include the following:

Transferring small amounts of data between nearby devices.

Interacting with proximity sensors to give users a customized experience based on their current location.

apps can use to discover devices, query for services, and transmit information.

In contrast to classic Bluetooth, BLE is designed for significantly lower power consumption. This allows apps to

monitors, and fitness devices.

communicate with BLE devices that have stricter power requirements, such as proximity sensors, heart rate



devices is accessible to all apps on the user's device. For this reason, if your app captures sensitive data, you should implement app-layer security to protect the privacy of

Caution: When a user pairs their device with another device using BLE, the data that's communicated between the two

For BLE-enabled devices to transmit data between each other, they must first form a channel of communication. Use of the Bluetooth LE APIs requires you to declare several permissions in your manifest file. Once your app has

The basics

that data.

permission to use Bluetooth, your app needs to access the BluetoothAdapter and determine if Bluetooth is available on the device. If Bluetooth is available, the device will scan for nearby BLE devices. Once a device is found, the capabilities of the BLE device are discovered by connecting to the GATT server on the BLE device. Once a connection is made, data can be transferred with the connected device based on the available services and characteristics.

• Generic Attribute Profile (GATT)

Key terms and concepts

The GATT profile is a general specification for sending and receiving short pieces of data known as

The following is a summary of key BLE terms and concepts:

"attributes" over a BLE link. All current BLE application profiles are based on GATT. Review the Android BluetoothLeGatt sample on GitHub to learn more.

Profiles The **Bluetooth SIG** defines many profiles for BLE devices. A profile is a specification for how a device works in a particular application. Note that a device can implement more than one profile. For example, a

device could contain a heart rate monitor and a battery level detector.

characteristic can be thought of as a type, analogous to a class.

GATT is built on top of the Attribute Protocol (ATT). This is also referred to as GATT/ATT. ATT is optimized

Attribute Protocol (ATT)

a Universally Unique Identifier (UUID), which is a standardized 128-bit format for a string ID used to uniquely identify information. The attributes transported by ATT are formatted as characteristics and services. Characteristic

A characteristic contains a single value and 0-n descriptors that describe the characteristic's value. A

to run on BLE devices. To this end, it uses as few bytes as possible. Each attribute is uniquely identified by

Descriptor

specify a human-readable description, an acceptable range for a characteristic's value, or a unit of measure that is specific to a characteristic's value. Service

A service is a collection of characteristics. For example, you could have a service called "Heart Rate

Monitor" that includes characteristics such as "heart rate measurement." You can find a list of existing

Descriptors are defined attributes that describe a characteristic value. For example, a descriptor might

GATT-based profiles and services on bluetooth.org .

Roles and responsibilities

• Central versus peripheral. This applies to the BLE connection itself. The device in the central role scans, looking for advertisement, and the device in the peripheral role makes the advertisement.

The following roles and responsibilities apply when a device interacts with a BLE device:

- GATT server versus GATT client. This determines how two devices talk to each other once they've established the connection. To understand the distinction, imagine that you have an Android phone and an activity tracker
- that is a BLE device. The phone supports the central role; the activity tracker supports the peripheral role. To establish a BLE connection you need one of each—two things that only support peripheral couldn't talk to each other, nor could two things that only support central. Once the phone and the activity tracker have established a connection, they start transferring GATT metadata to one another. Depending on the kind of data they transfer, one or the other might act as the server. For example, if the

In the example used in this topic, the app (running on an Android device) is the GATT client. The app gets data from the GATT server, which is a BLE heart rate monitor that supports the Heart Rate Profile . You could alternatively design your app to play the GATT server role. See **BluetoothGattServer** for more information.

War das hilfreich?

凸 切

activity tracker wants to report sensor data to the phone, it might make sense for the activity tracker to act as the

server. If the activity tracker wants to receive updates from the phone, then it might make sense for the phone to act

Content and code samples on this page are subject to the licenses described in the Content License. Java and OpenJDK are trademarks or registered trademarks of Oracle and/or its affiliates.

MORE ANDROID

Android

Android for

Enterprise

Security

Source

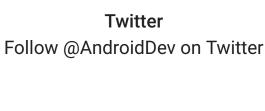
News

Blog

Podcasts

Last updated 2021-10-27 UTC.

as the server.



Connect with the Android Developers community on LinkedIn

LinkedIn

YouTube Check out Android Developers on YouTube

DOCUMENTATI...

DOWNLOADS

Android Studio

Developers guides

AND

guide

DISCOVER

Machine Learning

Gaming

Privacy

5G

ANDROID DEVICES

Wear OS

Android for cars

Android Things

Chrome OS devices

Large screens

Android TV

Pie Oreo

Nougat

Lollipop

KitKat

RELEASES

Android 11

Android 10

Marshmallow

Download Studio

Android NDK

API reference

Report documentation bug Google Play support Join research

studies

bug

SUPPORT

Report platform

Google Developers Android

Privacy | License | Brand guidelines

Chrome Firebase

Google Cloud Platform

All products

Get news and tips by email

Subscribe

Language ▼