Content Providers

Learning Objective

After studying this module learner should be able to:

- Define Content Provider
- List Android Native Content Provider
- Access the Content Provider
- Understand the parts of Content URI
- List methods of Content Provider
- Use Contacts Content Provider

Introduction

A content provider is a component. It supplies data from one application to others on request. A content provider stores its data in different ways. This data can be stored in a database, in files, or even over a network.

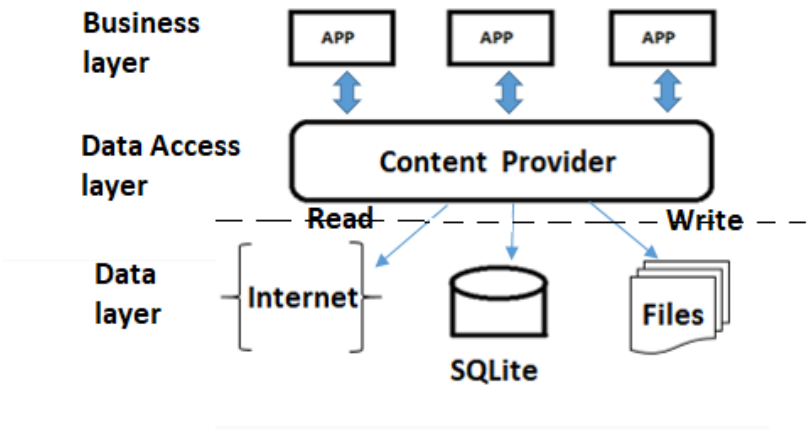


Figure-1 Content Provider

Content providers becomes very useful for sharing data across applications,.

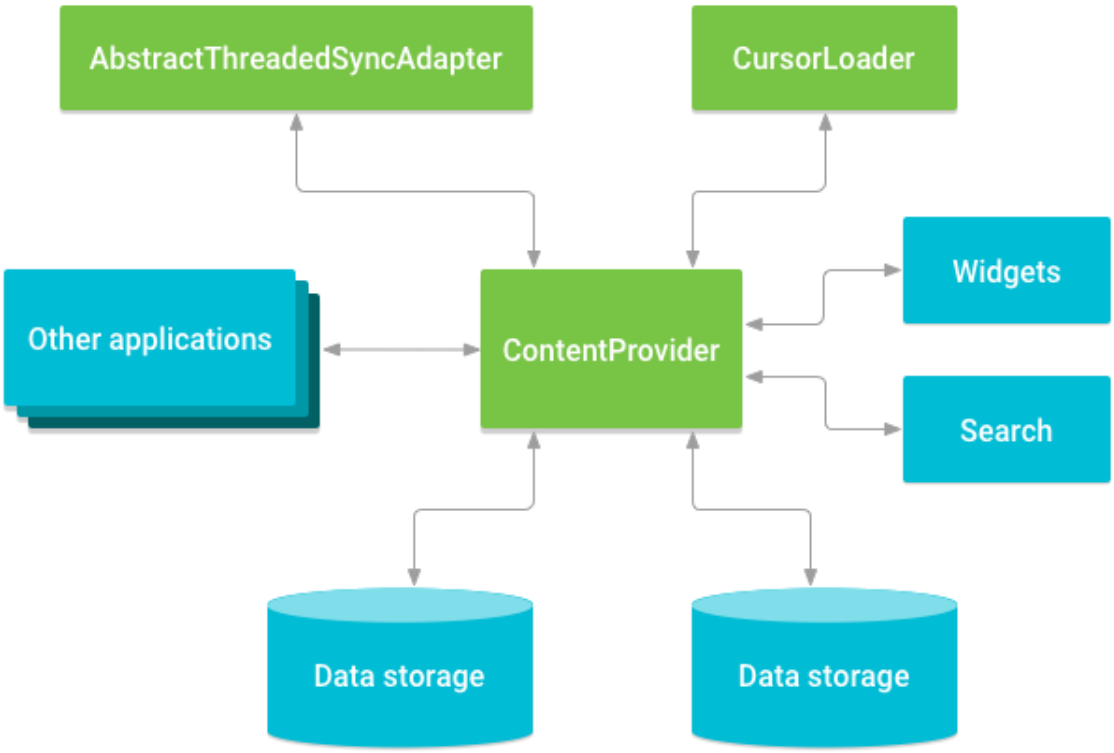
Content providers works as a central content in one place and have many different applications access it as needed. A content provider behaves very much like a database.

You work with content providers when:

- One may want to implement code to access an existing content provider in another application.
- One may want to create a new content provider in your application to share data with other applications.

Content Provider

A content provider:



- Can share of access to your application data from any other applications
- Can send data to a widget or application.
- Can return custom search suggestions for your application through the search framework using SearchRecentSuggestionsProvider.
- Can synchronize application data with your server using an implementation of AbstractThreadedSyncAdapter.
- Can load data in your User Interface using a CursorLoader.

Android Native ContentProvider

- **Browser** - Read or modify bookmarks, browser history, or web searches.
- **CallLog** - View or update the call history.
- **Contacts** - Retrieve, modify, or store the personal contacts. Three-tier data model of tables under a ContactsContract object:
    - **ContactsContract.Data** - Contains all kinds of personal data.
    - **ContactsContract.RawContacts** - Contains a set of Data objects associated with a single account or person.
    - **ContactsContract.Contacts** - Contains an aggregate of one or more RawContacts, presumably describing the same person.
- **MediaStore** - Access audio, video, and images.
- **Setting** - View and retrieve Bluetooth settings, ring tones, and other device preferences.

Android defines CONTENT_URI **constants** for all the providers that come with the platform.

ContactsContract.CommonDataKinds.Phone.CONTENT_URI

Browser.BOOKMARKS_URI

MediaStore.Video.Media.EXTERNAL_CONTENT_URI

ContactsContract.Contacts.CONTENT_URI

Content URI

To query a content provider, you specify the query string in the form of a URI which has format content://authority/path/id

URI Part	Desription
content:	The string content:// is always present, and identifies this as a content URI.
Authority	This specifies the name of the content provider, for example contacts, browser etc.

Path	Zero or more segments, separated by a forward slash (/), that identify some subset of the provider's data.
Id	A unique numeric identifier for a single row in the subset of data identified by the preceding path part.

There can be an instance identifier that refers to a specific data instance.

content://media/internal/images :
return the list of all internal images on the device.

content://media/external/images :
return the list of all the images on external storage (e.g., SD card) on the device.

content://call_log/calls :
return a list of all the calls registered in the call log.

content://browser/bookmarks :
return a list of bookmarks stored in the browser.

content://contacts/people/45 :
return the single result row, the contact with ID=45.

Accessing Content Provider

- If you want to access data using content provider, use the ContentResolver object in your application's Context to communicate with the provider as a client.
- The ContentResolver object communicates with the provider object. This object receives data requests from clients, performs the requested action, and returns the results.
- The ContentResolver methods provide the basic "CRUD" (create, retrieve, update, and delete) operations of a storage.
- To access ContentProvider from your UI easiest way is to use a CursorLoader to run an asynchronous query in the background. The Activity or Fragment is used as UI that calls a CursorLoader to the query, This gets the ContentProvider using the ContentResolver.

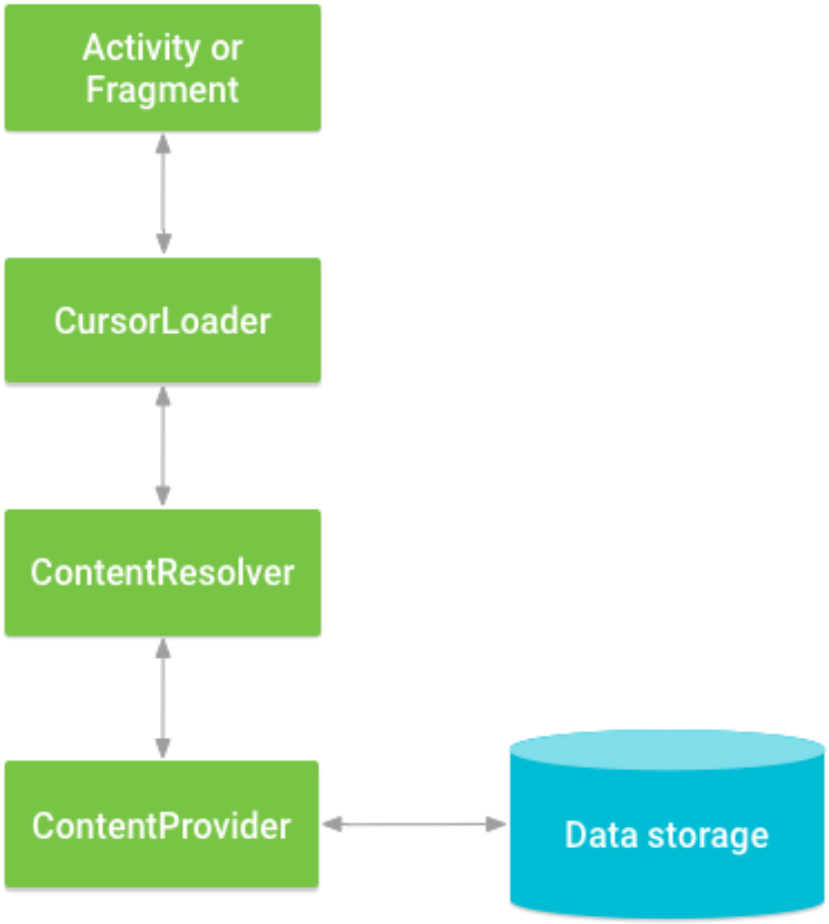


Figure-2 Accessing Content Provider

A content provider is implemented as a subclass of ContentProvider class.

```
public class My Application extends ContentProvider {

}
```

Methods

Methods of ContentProvider:

Method	Description
query()	Retrieve data from your provider. Use the arguments to select the table to query, the rows and columns to return, and the sort order of the result. Return the data as a Cursor object.

insert()	Insert a new row into your provider. Use the arguments to select the destination table and to get the column values to use. Return a content URI for the newly-inserted row.
update()	Update existing rows in your provider. Use the arguments to select the table and rows to update and to get the updated column values. Return the number of rows updated.
delete()	Delete rows from your provider. Use the arguments to select the table and the rows to delete. Return the number of rows deleted.
getType()	Return the MIME type corresponding to a content URI. This method is described in more detail in the section Implementing content provider MIME types.
onCreate()	Initialize your provider. The Android system calls this method immediately after it creates your provider.

Steps to Create Custom Content Provider

1. Create a Content Provider class that extends the ContentProviderbaseclass.
2. Define your content provider URI address which will be used to access the content.
3. Create your own database to keep the content.
4. Usually, Android uses SQLite database and framework needs to override onCreate() method which will use SQLite Open Helper method to create or open the provider's database. When your application is launched, the onCreate() handler of each of its Content Providers is called on the main application thread.
5. Implement Content Provider queries to perform different database specific operations.
6. Register your Content Provider in your activity file using <provider> tag.

Let us Sum Up

Content Providers: Content provider is a component that supplies data from one application to others on request.

For accessing Content Providers: Use ContentResolver and CursorLoader.

To query a content provider, use the query string in the form of a URI.

Further Reading

Recommended links:

http://developer.android.com/

Recommended Books:

1. Reto Meier, “Professional Android 2 Application Development”, Wiley India Pvt Ltd (2011)
2. Teach.Yourself.Android.Application.Development.in.24. Hours. 2nd.Edition.
3. Learning Android-Book by Marko Gargenta (2011)

Assignments

1. Create an Android app to add name and age and then retrieve the student record by using content provider.
2. Work of ContentResolver object?
3. What is use of CursorLoader?
4. Specify the query string in the form of a URI to query a content provider?
5. Mention URI parts of Content URI?

Acknowledgement: “The content in this module is modifications based on work created and shared by the Android Open-Source Project and used according to terms described in the Creative Commons 2.5 Attribution License.”