Programming Handheld Systems
ContentProvider
ContentProvider

Represents a repository of structured data
Encapsulates data sets
Enforces data access permissions
**ContentProvider**

**Intended for inter-application data sharing**

**Clients access ContentProviders through a ContentResolver**
**ContentResolver**

Presents a database-style interface for reading & writing data

query, insert, update, delete, etc.

Provides additional services such as change notification
**ContentResolver**

Get reference to **ContentResolver** by calling `Context.getContentResolver()`
**ContentProvider & ContentResolver**

Together these classes let code running in one process access data managed by another process.
Android ContentProviders

Browser – bookmarks, history
Call log – telephone usage
Contacts – contact data
Media – media database
UserDictionary – database for predictive spelling
Many more
## ContentProvider Data Model

Data represented logically as database tables

<table>
<thead>
<tr>
<th>_ID</th>
<th>artist</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Lady Gaga</td>
</tr>
<tr>
<td>44</td>
<td>Frank Sinatra</td>
</tr>
<tr>
<td>45</td>
<td>Elvis Presley</td>
</tr>
<tr>
<td>53</td>
<td>Barbara Streisand</td>
</tr>
</tbody>
</table>
Content providers referenced by URIs

The format of the URI identifies specific data sets managed by specific ContentProviders.
Format

```
content://authority/path/id
```

*content* — scheme indicating data that is managed by a content provider

*authority* — id for the content provider

*path* — 0 or more segments indicating the type of data to be accessed

*id* — a specific record being requested
Example: Contacts URI

ContactsContract.Contacts.CONTENT_URI =
"content://com.android.contacts/contacts/"
ContentResolver.query()

Cursor query (  
   Uri uri,          // ContentProvider Uri
   String[] projection  // Columns to retrieve
   String selection   // SQL selection pattern
   String[] selectionArgs  // SQL pattern args
   String sortOrder    // Sort order
  )

Returns a Cursor for iterating over the set of results
ContentProviderWithCursorAdapter

Extracts Contact information from the Android Contacts ContentProvider
Displays each contact’s name and photo, if available
CursorLoader

Conducting intensive operations on the main thread can affect application responsiveness.

CursorLoader uses an AsyncTask to perform queries on a background thread.
Using a CursorLoader

Implement LoaderManager's LoaderCallbacks interface
Create and initialize a cursor loader
initLoader()

**Initialize and activate a Loader**

Loader<D> initLoader(
    int id,
    Bundle args,
    LoaderCallbacks<D> callback)
Called to instantiate and return a new Loader for the specified ID

Loader<D> onCreateLoader (int id, Bundle args)
**Loader Callbacks**

Called when a previously created loader has finished loading

```java
void onLoadFinished(
    Loader<D> loader, D data)
```
LoaderCallbacks

Called when a previously created loader is reset

void onLoaderReset (Loader<D> loader)
ContentProviderWithCursorLoader

Extracts Contact information from the android Contacts ContentProvider
Displays each contact's name and photo, if available
But it uses a CursorLoader when querying the Content Provider
ContentResolver.delete()

int delete (  
    Uri url,       // content Uri
    String where,  // SQL selection pattern
    String[] selectionArgs  // SQL pattern args
)

Returns the number of rows deleted
ContentResolver.insert()

Uri insert (  
    Uri url,       // content Uri  
    ContentValues values       // values for new row  
)

Returns the Uri of the inserted row
ContentResolver.update()

Uri update(
    Uri url,                   // content Uri
    ContentValues values       // new field values
    String where,             // SQL selection pattern
    String[] selectionArgs    // SQL pattern args
)

Returns the number of rows updated
ContentProviderInsertContacts

Application reads contact information from the Android Contacts ContentProvider.
Inserts several new contacts into Contacts ContentProvider.
Displays old and new contacts.
Deletes these new contacts on exit.
Creating a ContentProvider

Implement a storage system for the data
Define a Contract Class to support users of your ContentProvider
Implement a ContentProvider subclass
Declare and configure content provider in AndroidManifest.xml
ContentProviderCustom

Application defines a ContentProvider for ID/string pairs
ContentProviderCustomUser

Reads ID/String pairs from the ContentProvider we just examined
Displays the data in a ListView
Next Time

Service