

Identifying Content Provider Content



Jim Wilson

MOBILE SOLUTIONS DEVELOPER & ARCHITECT

@hedgehogjim blog.jwhh.com



What to Expect from This Module



Content Provider Organization

Content Provider Contract Class

Specifying Table URIs

Matching Table URIs

URI Request Handling

Adding a Table That Abstracts DB Joins

Content Provider Organization

Content providers expose tables

- Normally expose multiple tables
- Often correspond to database tables
- May also provide an abstraction

Need a way to describe content provider

- Identify available tables
- Identify table columns



Content Provider Contract Class

Presents the public appearance

- Avoid exposing details of structure
- Focus is on data access
- Includes base URI for content provider

`content://com.jwhh.jim.notekeeper.provider`



Content Provider Contract Class

Table information is in nested classes

- Separate class for each table
- Content provider's public representation

Each class describes how to access a table

- Names of available columns
- Table URI



Content Provider Contract Class

Table URIs

- Starts with content scheme
- Includes provider authority
- Table is added as a path
- Use Uri.withAppendedPath static method

`content://com.jwhh.jim.notekeeper.provider/courses`



Content Provider Contract Class

Each table class describes table access

- Table URI
- Names of available columns

Defining column name constants

- Could add constants directly to classes



Content Provider Contract Class

Columns often duplicated across tables

- Same name and meaning
- Could lead to redundant constants
- Better to manage column constants separately from table classes



Content Provider Contract Class

Column constants managed in interfaces

- Interfaces group related columns
- Simplifies organization and maintenance

Interfaces nested within contract class

- Not for use outside of provider
- Usually marked protected

Associating constants with table classes

- Classes implement appropriate interfaces



Matching Content URIs

Content provider URI handling

- Often need to support many URIs
- Interpreting URIs can be challenging
- URI patterns can become complex



Matching Content URIs

UriMatcher

- Handles details of interpreting URIs
- Translates a URI to an integer constant



Matching Content URIs

Preparing UriMatcher in a content provider

- Usually have a static member field
- Add valid URIs in a static initializer
- Use addURI method
- Map each URI to an integer constant

Handling requests with UriMatcher

- Use match method
- Translate URI to an integer constant
- Take action based on constant



Table Abstraction

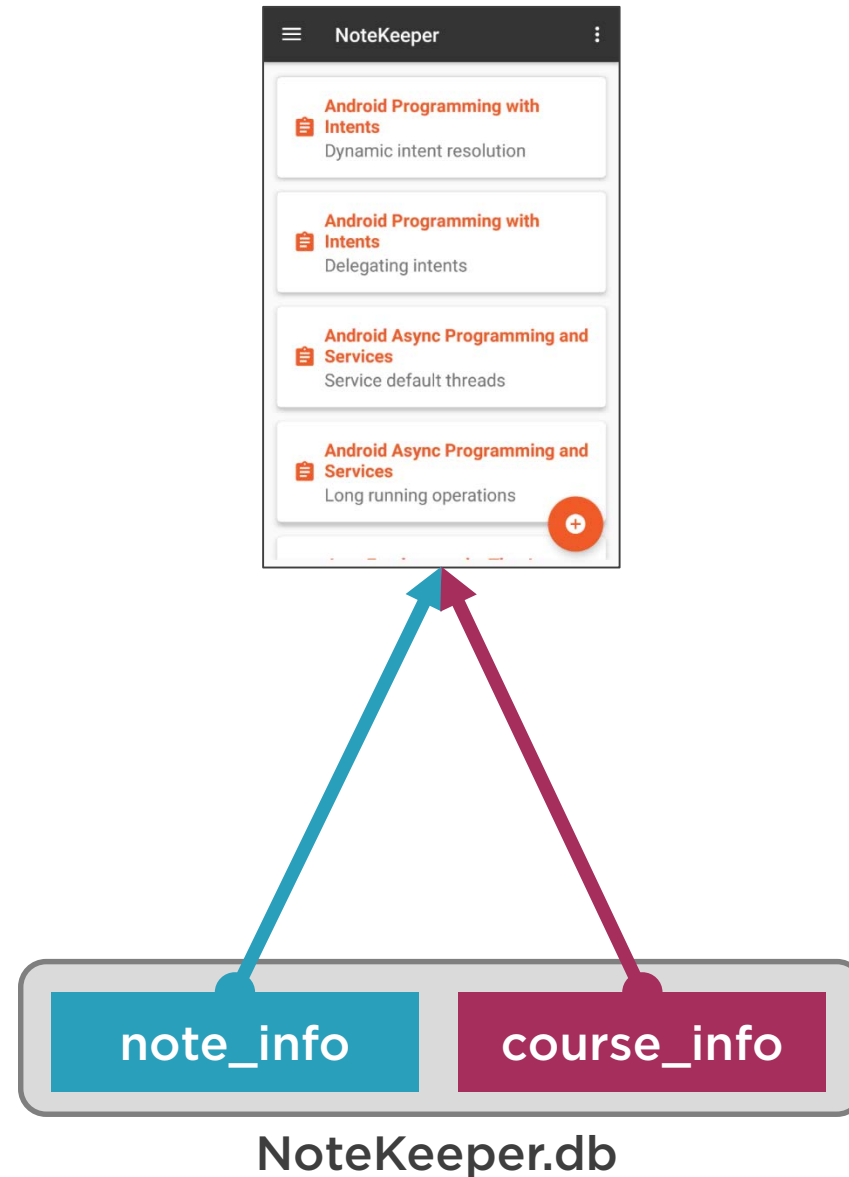


Table Abstraction

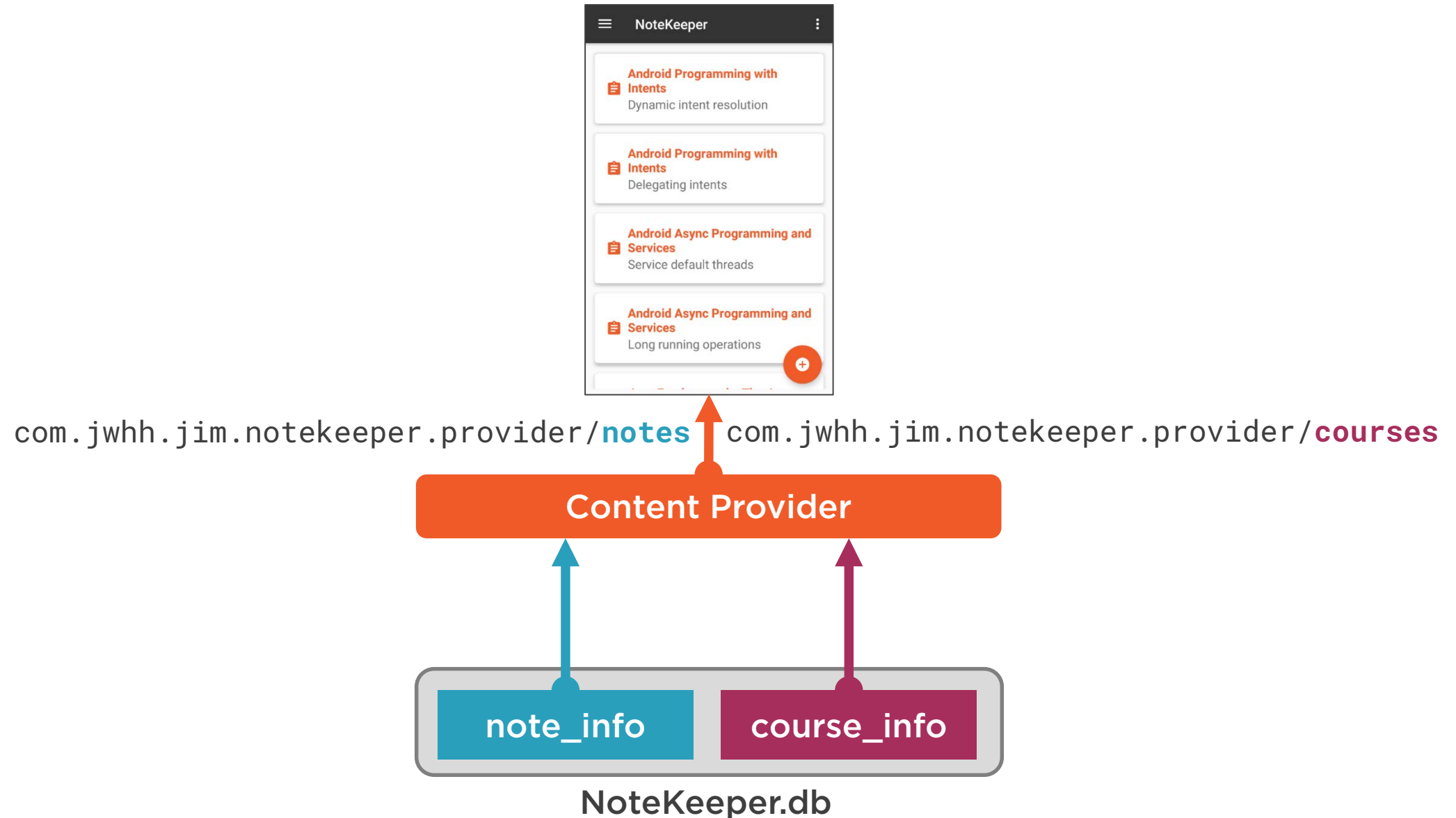
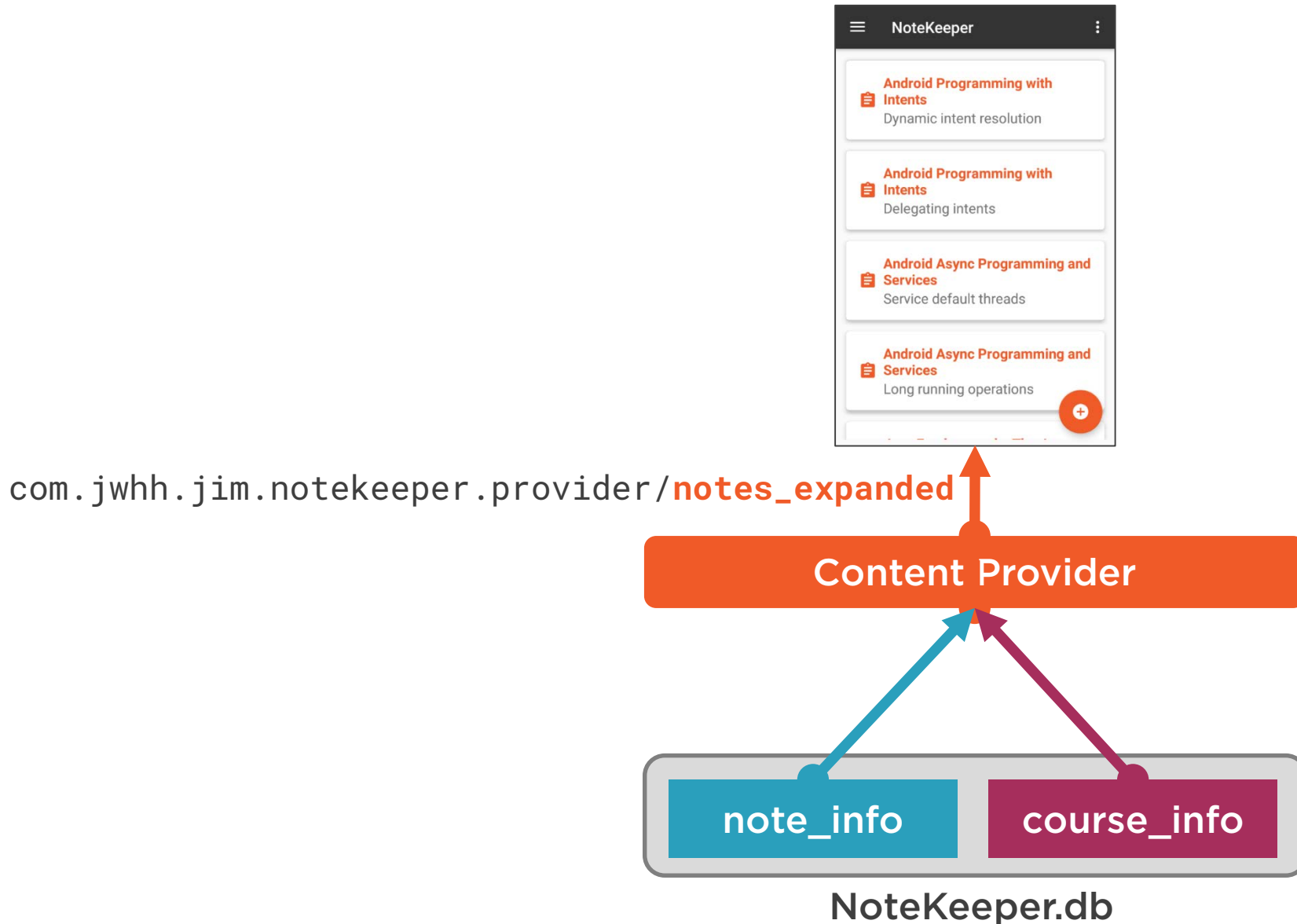


Table Abstraction



Summary



Content providers expose tables

- Normally expose multiple tables
- Often correspond to database tables
- May also provide an abstraction
- Each table has an individual URI

Contract class

- Presents public appearance
- Includes classes for contained tables

Summary



Table classes

- Classes nested in contract class
- Table URI
- Names of available columns

Column name constants

- Often defined in interfaces
- Interfaces nested in contract class
- Provides central management
- Tables implement appropriate interfaces

Summary



Table URIs

- Starts with content scheme
- Includes provider authority
- Table is added as a path

Handling table URIs

- Use UriMatcher class
- Translates URI to an integer constant