EXAMPLE 6-1. Example book data mappings

```
book_get_info : isbn -> {title, author, publisher, price, cover picture}
book_get_reviews: isbn -> set(review_ids)
bookuser_get_reviews: books_user_id -> set(review_ids)
review_get_info: review_id -> {isbn, books_user_id, rating, commentary}
```

All of these are ultimately implemented as something very similar to simple sets and fetches from an indexed data table. Any such book site worth its salt would likely implement other functions that are not so simple, such as the simple "search" in Example 6-2.

EXAMPLE 6-2. A simple search mapping

```
search_title_string: title_string -> set({isbn, relevance score})
```

Each key in the domain of these functions generally would justify at least one web page on *http://fettermansbooks.com*—a unique set of logic surrounding the range data, rendered through a unique display path. For instance, to see a selection of reviewer X's submissions, a *http://fettermansbooks.com* user would likely be directed to visit a page like *fettermansbooks.com/reviews.php?books_user_id=X*, or to see all info about a particular book with ISBN *Y* (including hops to individual review pages), he would visit *http://fettermansbooks.com/book.php?isbn=Y*.

A notable property of sites such as *http://fettermansbooks.com* is that nearly every piece of data is available to every user. It generates all the content in, say, the `book_get_info` mapping to aid users in discovering as much information about a book as possible. This may be optimal in the case of a site trying to sell books, but visibility restrictions govern much of the architectural considerations of the data access layer in the following example using social data.

Some Facebook Core Data

With the rise in popularity of the network of technologies called Web 2.0, the centrality of data within systems has only grown more obvious. The central themes of Web 2.0 presences are that they are data-driven, and that users themselves provide the majority of that data. Facebook, like *http://fettermansbooks.com*, primarily comprises a set of core data mappings that motivate the feel and functionality of its website. An extremely stripped-down set of these Facebook mappings could look like the set in Example 6-3.

EXAMPLE 6-3. Example social data mappings

```
user_get_friends: uid -> set(uids)
user_get_info: uid -> {name, pic, books, current_location,...}
can_see: {uid_viewer, uid_viewee, table_name, field_name} -> 0 or 1
```

`uid` here refers to a (numeric) Facebook user identifier, and the info returned from `user_get_info` refers to a user's profile content (see *users.getInfo* in Facebook's developer documentation), including perhaps titles of the user's favorite books as they are entered on