

- Creating a data-driven markup language to both integrate application display back into Facebook and enable use of otherwise inaccessible data (Facebook FBML)

And once we've evolved the architecture of an application significantly from a separate stack:

- Building technologies that bridge the gap between the Facebook experience and the external application's experience

For the data platform consumer, this chapter illustrates the design decisions made and the rationale behind them. Notions such as user sessions and authentication, web services, and the various ways of handling the application's logic will constantly appear as themes in these types of platforms all over the Web. Understanding the thought behind them provides a great exercise in data architecture, and proves relevant when thinking about the kinds of features and forms these platform producers will likely create in the future.

The data platform producer is encouraged to keep in mind his own data set, and learn from the ways Facebook has opened its data model. Some of the design choices and reconciliations remain specific to Facebook, or at least to handling social data guarded by privacy, and may not be wholly applicable to a given data set. Nonetheless, at each step we present a product problem, a data-driven solution, and the solution's high-level implementation. With each new solution, we are essentially creating a new product or platform, so at all points we must reconcile this new product with the expectations of users. In turn, we create new technologies to accompany each step of the evolution, and sometimes change the web architecture surrounding the application itself.

An open source version of the Facebook Platform is available at <http://developers.facebook.com/>. Like much of that release, the code in this chapter is written in PHP. Feel free to follow along, noting that the samples here are abbreviated for clarity.

We start with the motivation for these types of integrations with an example of "external" application logic and data (a book store), Facebook's social data (user information and "friend" relationships), and the case for integrating the two.

## Some Application Core Data

Web applications, even those that do not produce or consume a data platform of any sort, are still motivated largely by their internal data. As an example, take <http://fettermansbooks.com>, a hypothetical website that provides information on books (and likely, the ability to purchase these titles if the mood struck). The site's features may include a searchable index of inventory, basic information about each of the products, and even user-contributed reviews about each title. Access to this specific information forms the core of the application and motivates the rest of the architecture. The site may employ Flash and AJAX, be accessible through mobile devices, and provide an award-winning user interface. However, <http://fettermansbooks.com> exists essentially to provide visitors some kind access to core mappings like those in Example 6-1.