http://facebook.com. This system differs little from *http://fettermansbooks.com* at its core, except that the epicenter of the data, and hence site functionality, revolves around users' connection to other users ("friends"), users' content ("profile information"), and visibility rules for that content ("can_see").

This `can_see` data set is special. Facebook has a very central notion of privacy around user-generated data—business rules for user X's view of user Y's information. Never directly viewable by itself, this data motivates very important considerations that will emerge again and again when we see examples of integrating external logic and data with Facebook's. By itself, Facebook's pervasive use of this data set differentiates it from a website like *http://fettermansbooks.com*.

The Facebook Platform and other social platforms are a recognition that these types of social mappings are useful, both within a site like *http://facebook.com*, and when *integrated* with the functionality of a site such as *http://fettermansbooks.com*.

Facebook's Application Platform

For a user of both *http://fettermansbooks.com* and *http://facebook.com*, the picture of Internet applications at this point looks something like Figure 6-1.

In the usual n-tier architecture, an application maps input (for the Web, the union of GET, POST, and cookie information) to requests for raw data likely residing in a database. These are translated to in-memory data and passed to some *business logic* for intelligent processing. The output module translates these data objects for display, into HTML, JavaScript, CSS, and so on. Here, on the top of the figure, is an application's n-tier stack running on its infrastructure. Before the advent of applications in its Platform, Facebook operated wholly under the same architecture. Importantly, in both architectures, the business logic (including Facebook's privacy) is effectively executed according to rules established in some data component of the system.

More voluminous and relevant data means the business logic may deliver more personally tailored content, so the experience of browsing books to review, read, or purchase on *http://fettermansbooks.com* (or any such application) would be powerfully *augmented* by a user's social data from Facebook. Specifically, showing friends' book reviews, wish lists, and purchases could aid a user in making her own purchasing decisions, discovering new titles, or strengthening connections to other users. If Facebook's internal mapping `user_get_friends` were accessible to other external applications such as *http://fettermansbooks.com*, this would add powerful social context to these otherwise separate applications, and eliminate the application's need to create its own social network. Applications of all sorts could do well to integrate this data, since developers can apply these core Facebook mappings to countless other web presences where users produce *or* consume content.