Clearly, not all container sites implementing their own platforms need these modifications, but FBJS demonstrates the kinds of solutions needed to work around a new web architecture like this. We present only the solutions as general ideas here; much of FBJS involves incremental improvements incorporated into FBML and extensive proprietary JavaScript libraries.

First, JavaScript generally has access to the entire Document Object Model (DOM) tree of the document that contains it. Yet in a platform canvas page, Facebook includes many of its own elements, which developers are not allowed to change. The solution? Prefix developer-provided HTML elements and JavaScript symbols with the ID of the app itself (e.g., `app1234567`). In this way, attempting to call this disallowed `alert()` in developer JavaScript will call the undefined function `app1234567_alert`, and only portions of the document's HTML that the developer provided himself can be accessed by something such as JavaScript's `document.getElementById`.

As an example of the kinds of transforms FBJS needs to make on provided FBML (including `<script>` elements), we create a simple FBML page implementing AJAX functionality in Example 6-30.

EXAMPLE 6-30. An FBML page using FBJS

```
These links demonstrate the Ajax object:
<br /><a href="#" onclick="do_ajax(Ajax.RAW); return false;">AJAX Time!</a><br />
<div>
<span id="ajax1"></span>
</div>

<script>
function do_ajax(type) {
  var ajax = new Ajax(); // FBJS Ajax library.
  ajax.responseType = type;
  switch (type) {
  <!-- note FBJS's Ajax object also implements AJAX.JSON and AJAX.FBML, omitted
  for brevity -->
    case Ajax.RAW: ajax.ondone = function(data) {
      document.getElementById('ajax1').setTextValue(data);
    };
    break;
  };

 ajax.post('http://www.fettermansbooks.com/testajax.php?t='+type);

}
</script>
```

FBML with our FBJS modifications transforms this input to the HTML in Example 6-31. The `NOTE` comments in this example refer to each kind of transform required, and are not part of the actual output.