Drag and drop a folder onto Chrome now available

By Eiji Kitamura

Developer Advocate in Tokyo

As web apps evolve, you might have found it handy to let users drag and drop files from the desktop onto the browser to edit, upload, share, etc. But unfortunately, we've been unable to drag and drop folders onto web pages. Luckily, beginning with Chrome 21, this issue will be addressed (already available in the Chrome dev channel).

Passing multiple files with drag and drop

Let's look at a code sample of existing drag and drop.

```
<div id="dropzone"></div>
```

```
var dropzone = document.getElementById('dropzone');
dropzone.ondrop = function(e) {
  var length = e.dataTransfer.files.length;
  for (var i = 0; i < length; i++) {
    var file = e.dataTransfer.files[i];
    ... // do whatever you want
  }
};
```

In this example, you can actually drag and drop a file or files from the desktop to your browser, but when you try to pass a folder, notice that it will be either rejected or treated as a `File` object resulting in a failure.

How to handle dropped folders

Chrome 21 allows you to drop a folder or multiple folders into the browser window. To handle these, you need to change the way you handle dropped objects.

```
<div id="dropzone"></div>
```

```
var dropzone = document.getElementById('dropzone');
dropzone.ondrop = function(e) {
```

```
  var length = e.dataTransfer.items.length;
  for (var i = 0; i < length; i++) {
    var entry = e.dataTransfer.items[i].webkitGetAsEntry();
    if (entry.isFile) {
      ... // do whatever you want
    } else if (entry.isDirectory) {
      ... // do whatever you want
    }
  }
};
```

Notice that a big difference here is that you can treat a dropped object as `Entry` (`FileEntry` or `DirectoryEntry`) by using new function called `getAsEntry` (`webkitGetAsEntry`). After obtaining access to the `Entry` object, you can use standard file handling methods that were introduced in the FileSystem API specification. For example, this example shows how you can detect if a dropped object is a file or a directory by examining the `.isFile` (or the `.isDirectory`) field.

For further information regarding the FileSystem API, read <u>Exploring the FileSystem APIs</u> [↗], regarding new drag and drop capability, read <u>this document</u>. Of course, these features are open standards or are waiting to become one soon.

Last updated July 2, 2018.