Loading Remote Modules

Jeff Valore
CodingWithSpike.wordpress.com
@CodingWithSpike

pluralsight
hardcore dev and IT training
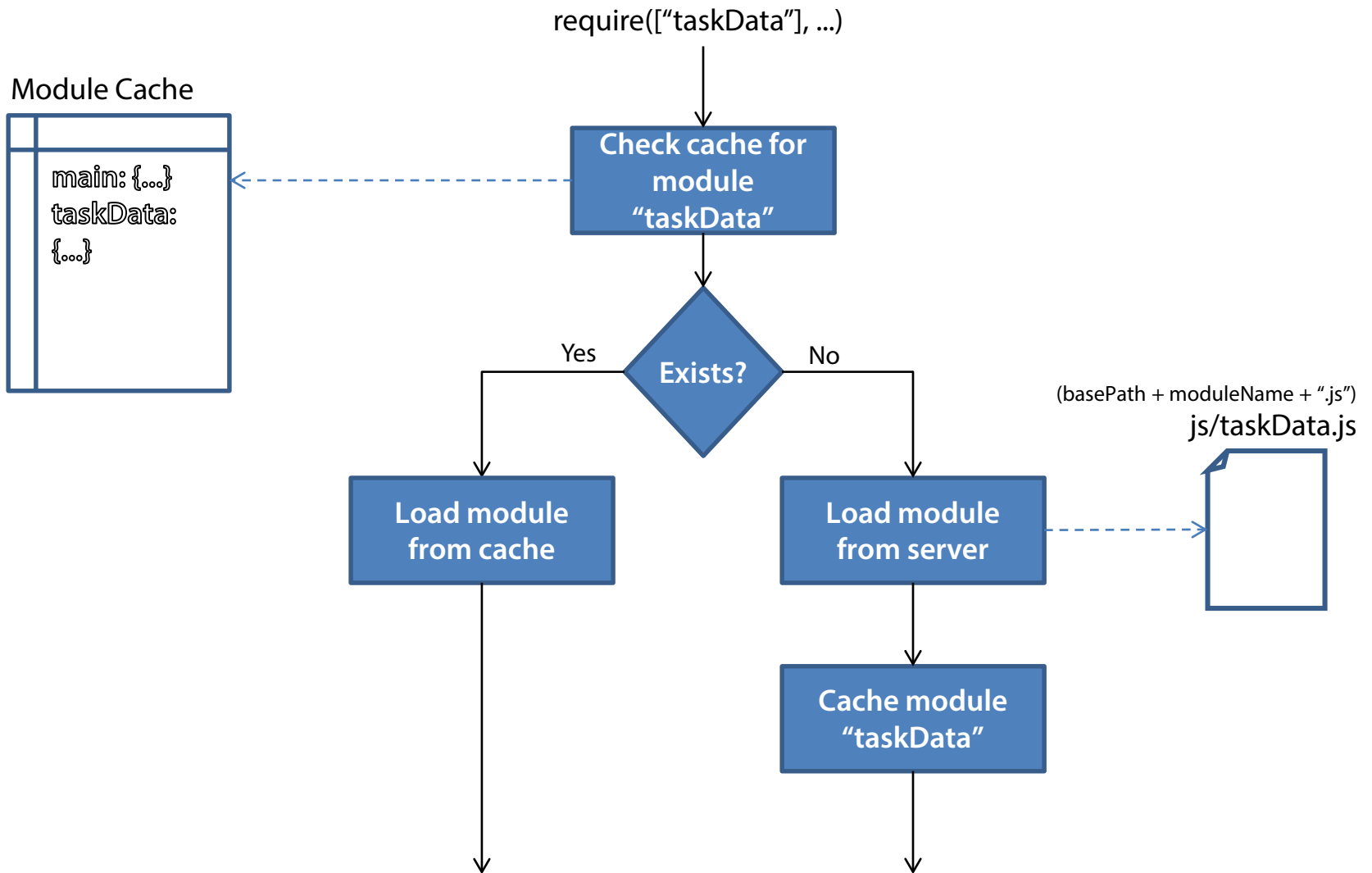
tasks.js

main.js

taskData.js

main.js

taskRenderer.js

require(["taskData"], ...)

Module Cache

main: {...}
taskData:
{...}

Check cache for module "taskData"

Exists?

Yes

No

(basePath + moduleName + ".js")
js/taskData.js

Load module from cache

Load module from server

Cache module "taskData"

define() Function Parameters

- **When defining a module inline with other code:**

  define(*{moduleNameString}*, *{dependencyArray}*, *{callbackFunction}*);

- **When defining a module in a separate file:**

  define(*{dependencyArray}*, *{callbackFunction}*);

- **Simplified CommonJS Wrapper:**

  define(*{moduleNameString}*, function(require, exports, module) {
      var *{module}* = require(*{moduleName}*);
      exports. *{publicMember}* = *{publicValue}*;

Summary

- **RequireJS will load AMD modules from remote files relative to the "base path."**

- **The "base path" is set to the location of the "main" JavaScript file.**

- **The define() function no longer specifies a module name when using a remote file.**
  - define(*{dependencyArray}*, *{callbackFunction}*);

- **Using subdirectories helps to organize code at development time, but causes a performance hit at runtime, unless the RequireJS Optimizer is used.**