A common challenge with combining [chunkhash] and Code Splitting is that the entry chunk includes the webpack runtime and with it the chunkhash mappings. This means it's always updated and the [chunkhash] is pretty useless because this chunk won't be cached.

A very simple solution to this problem is to create another chunk that contains only the webpack runtime (including chunkhash map). This can be achieved with optimization.runtimeChunk options. To avoid the additional request for another chunk, this pretty small chunk can be inlined into the HTML page.

The configuration required for this is:

- use [chunkhash] in output.filename (Note that this example doesn't do this because of the example generator infrastructure, but you should)
- use [chunkhash] in output.chunkFilename (Note that this example doesn't do this because of the example generator infrastructure, but you should)

example.js

```
// some module
import("./async1");
import("./async2");
```

webpack.config.js

```
var path = require("path");
module.exports = {
    // mode: "development // "production",
    entry: {
        main: "./example"
    },
    optimization: {
        runtimeChunk: true
    },
    output: {
        path: path.join(__dirname, "dist"),
        filename: "[name].[chunkhash].js",
        chunkFilename: "[name].[chunkhash].js"
    }
};
```

index.html

```
<html>
    <head> </head>
    <body>
        <!-- inlined minimized file "runtime~main.[chunkhash].js" -->
            (()=>{"use strict"; var e,r,t,o,n={},a={};function i(e) {var r=a[e];if(void 0!==r)
        </script>
        <script src="dist/main.[chunkhash].js"></script>
</html>
dist/runtime~main.[chunkhash].js
/*****/ (() => { // webpackBootstrap
```

```
/*****/
           "use strict";
/*****/
           var __webpack_modules__ = ({});
/* webpack runtime code */
/*****/
           // The module cache
/*****/
           var __webpack_module_cache__ = {};
/*****/
/*****/
          // The require function
/*****/
           function __webpack_require__(moduleId) {
/*****/
              // Check if module is in cache
/*****/
              var cachedModule = __webpack_module_cache__[moduleId];
/*****/
              if (cachedModule !== undefined) {
/*****/
                  return cachedModule.exports;
/*****/
              }
/*****/
              // Create a new module (and put it into the cache)
/*****/
              var module = __webpack_module_cache__[moduleId] = {
/*****/
                  // no module.id needed
/*****/
                  // no module.loaded needed
/*****/
                  exports: {}
/*****/
              };
/*****/
              // Execute the module function
/*****/
/*****/
              __webpack_modules__[moduleId] (module, module.exports, __webpack_require__);
/*****/
/*****/
              // Return the exports of the module
/*****/
              return module.exports;
/*****/
           }
```

```
/*****/
/*****/
           // expose the modules object (__webpack_modules__)
/*****/
            __webpack_require__.m = __webpack_modules__;
/*****/
/*****/
            /* webpack/runtime/chunk loaded */
/*****/
            (() => {
/*****/
               var deferred = [];
               __webpack_require__.0 = (result, chunkIds, fn, priority) => {
/*****/
/*****/
                   if(chunkIds) {
/*****/
                       priority = priority || 0;
/*****/
                       for(var i = deferred.length; i > 0 && deferred[i - 1][2] > priority
/*****/
                       deferred[i] = [chunkIds, fn, priority];
/*****/
                       return;
/*****/
                   }
/*****/
                   var notFulfilled = Infinity;
/*****/
                   for (var i = 0; i < deferred.length; i++) {</pre>
/*****/
                       var [chunkIds, fn, priority] = deferred[i];
/*****/
                       var fulfilled = true;
/*****/
                       for (var j = 0; j < chunkIds.length; j++) {</pre>
/*****/
                           if ((priority & 1 === 0 || notFulfilled >= priority) && Object.
/*****/
                               chunkIds.splice(j--, 1);
/*****/
                           } else {
/*****/
                               fulfilled = false;
/*****/
                               if(priority < notFulfilled) notFulfilled = priority;</pre>
/*****/
                           }
/*****/
                       }
/*****/
                       if(fulfilled) {
/*****/
                           deferred.splice(i--, 1)
/*****/
                           var r = fn();
/*****/
                           if (r !== undefined) result = r;
/*****/
                       }
/*****/
                   }
/*****/
                   return result;
/*****/
               };
/*****/
           })();
/*****/
/*****/
            /* webpack/runtime/create fake namespace object */
/*****/
            (() => \{
/*****/
               var getProto = Object.getPrototypeOf ? (obj) => (Object.getPrototypeOf(obj)
/*****/
               var leafPrototypes;
/*****/
               // create a fake namespace object
/*****/
               // mode & 1: value is a module id, require it
/*****/
               // mode & 2: merge all properties of value into the ns
/*****/
               // mode & 4: return value when already ns object
/*****/
               // mode & 16: return value when it's Promise-like
```

```
/*****/
                // mode & 8/1: behave like require
/*****/
                __webpack_require__.t = function(value, mode) {
/*****/
                    if(mode & 1) value = this(value);
/*****/
                    if(mode & 8) return value;
/*****/
                    if(typeof value === 'object' && value) {
/*****/
                        if((mode & 4) && value.__esModule) return value;
/*****/
                        if((mode & 16) && typeof value.then === 'function') return value;
/*****/
/*****/
                    var ns = Object.create(null);
/*****/
                    __webpack_require__.r(ns);
/*****/
                    var def = {};
/*****/
                    leafPrototypes = leafPrototypes || [null, getProto({}), getProto([]), g
/*****/
                    for(var current = mode & 2 && value; typeof current == 'object' && !~le
/*****/
                        Object.getOwnPropertyNames(current).forEach((key) => (def[key] = ()
/*****/
                    }
/*****/
                    def['default'] = () => (value);
/*****/
                    __webpack_require__.d(ns, def);
/*****/
                    return ns;
/*****/
                };
            })();
/*****/
/*****/
/*****/
            /* webpack/runtime/define property getters */
/*****/
            (() => {
/*****/
                // define getter functions for harmony exports
/*****/
                __webpack_require__.d = (exports, definition) => {
/*****/
                    for(var key in definition) {
/*****/
                        if(__webpack_require__.o(definition, key) && !__webpack_require__.o
/*****/
                            Object.defineProperty(exports, key, { enumerable: true, get: de
/*****/
                        }
/*****/
                    }
/*****/
                };
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/ensure chunk */
/*****/
            (() => {
/*****/
                __webpack_require__.f = {};
/*****/
                // This file contains only the entry chunk.
/*****/
                // The chunk loading function for additional chunks
/*****/
                __webpack_require__.e = (chunkId) => {
/*****/
                    return Promise.all(Object.keys(__webpack_require__.f).reduce((promises,
/*****/
                        __webpack_require__.f[key](chunkId, promises);
/*****/
                        return promises;
/*****/
                    }, []));
/*****/
                };
/*****/
            })();
/*****/
```

```
/*****/
            /* webpack/runtime/get javascript chunk filename */
/*****/
            (() => {
/*****/
               // This function allow to reference async chunks
/*****/
                __webpack_require__.u = (chunkId) => {
/*****/
                    // return url for filenames based on template
/*****/
                    return "" + chunkId + ".[chunkhash].js";
/*****/
                };
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/hasOwnProperty shorthand */
/*****/
/*****/
                __webpack_require__.o = (obj, prop) => (Object.prototype.hasOwnProperty.cal
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/load script */
/*****/
            (() => \{
/*****/
               var inProgress = {};
/*****/
                // data-webpack is not used as build has no uniqueName
/*****/
                // loadScript function to load a script via script tag
/*****/
                __webpack_require__.l = (url, done, key, chunkId) => {
/*****/
                    if(inProgress[url]) { inProgress[url].push(done); return; }
/*****/
                    var script, needAttach;
/*****/
                    if(key !== undefined) {
/*****/
                        var scripts = document.getElementsByTagName("script");
/*****/
                        for(var i = 0; i < scripts.length; i++) {</pre>
/*****/
                            var s = scripts[i];
/*****/
                            if(s.getAttribute("src") == url) { script = s; break; }
/*****/
                        }
/*****/
                    }
/*****/
                    if(!script) {
/*****/
                        needAttach = true;
/*****/
                        script = document.createElement('script');
/*****/
/*****/
                        script.charset = 'utf-8';
/*****/
                        script.timeout = 120;
/*****/
                        if (_webpack_require__.nc) {
/*****/
                            script.setAttribute("nonce", __webpack_require__.nc);
/*****/
                        }
/*****/
/*****/
                        script.src = url;
/*****/
/*****/
                    inProgress[url] = [done];
/*****/
                    var onScriptComplete = (prev, event) => {
/*****/
                        // avoid mem leaks in IE.
/*****/
                        script.onerror = script.onload = null;
/*****/
                        clearTimeout(timeout);
```

```
/*****/
                        var doneFns = inProgress[url];
/*****/
                        delete inProgress[url];
/*****/
                        script.parentNode && script.parentNode.removeChild(script);
/*****/
                        doneFns && doneFns.forEach((fn) => (fn(event)));
/*****/
                        if(prev) return prev(event);
/*****/
                    }
/*****/
/*****/
                    var timeout = setTimeout(onScriptComplete.bind(null, undefined, { type:
/*****/
                    script.onerror = onScriptComplete.bind(null, script.onerror);
/*****/
                    script.onload = onScriptComplete.bind(null, script.onload);
/*****/
                    needAttach && document.head.appendChild(script);
/*****/
                };
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/make namespace object */
/*****/
            (() => \{
/*****/
                // define __esModule on exports
/*****/
                __webpack_require__.r = (exports) => {
/*****/
                    if(typeof Symbol !== 'undefined' && Symbol.toStringTag) {
/*****/
                        Object.defineProperty(exports, Symbol.toStringTag, { value: 'Module
/*****/
/*****/
                    Object.defineProperty(exports, '__esModule', { value: true });
/*****/
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/publicPath */
/*****/
            (() => \{
/*****/
                __webpack_require__.p = "dist/";
/*****/
            })();
/*****/
/*****/
            /* webpack/runtime/jsonp chunk loading */
/*****/
            (() => {
/*****/
                // no baseURI
/*****/
/*****/
                // object to store loaded and loading chunks
/*****/
                // undefined = chunk not loaded, null = chunk preloaded/prefetched
/*****/
                // [resolve, reject, Promise] = chunk loading, 0 = chunk loaded
/*****/
                var installedChunks = {
/*****/
                    1: 0
/*****/
                };
/*****/
/*****/
                __webpack_require__.f.j = (chunkId, promises) => {
/*****/
                        // JSONP chunk loading for javascript
/*****/
                        var installedChunkData = __webpack_require__.o(installedChunks, chu
/*****/
                        if(installedChunkData !== 0) { // O means "already installed".
/*****/
```

```
/*****/
                            // a Promise means "currently loading".
/*****/
                            if(installedChunkData) {
/*****/
                                promises.push(installedChunkData[2]);
/*****/
                            } else {
/*****/
                                if(1 != chunkId) {
/*****/
                                    // setup Promise in chunk cache
/*****/
                                    var promise = new Promise((resolve, reject) => (install
/*****/
                                    promises.push(installedChunkData[2] = promise);
/*****/
/*****/
                                    // start chunk loading
/*****/
                                    var url = __webpack_require__.p + __webpack_require__.u
/*****/
                                    // create error before stack unwound to get useful stac
/*****/
                                    var error = new Error();
/*****/
                                    var loadingEnded = (event) => {
/*****/
                                        if(__webpack_require__.o(installedChunks, chunkId))
/*****/
                                            installedChunkData = installedChunks[chunkId];
/*****/
                                            if(installedChunkData !== 0) installedChunks[ch
/*****/
                                            if(installedChunkData) {
/*****/
                                                var errorType = event && (event.type === '1
/*****/
                                                var realSrc = event && event.target && even
/*****/
                                                error.message = 'Loading chunk ' + chunkId
/*****/
                                                error.name = 'ChunkLoadError';
/*****/
                                                error.type = errorType;
/*****/
                                                error.request = realSrc;
/*****/
                                                installedChunkData[1](error);
/*****/
/*****/
                                        }
/*****/
                                    };
/*****/
                                    __webpack_require__.l(url, loadingEnded, "chunk-" + chu
/*****/
                                } else installedChunks[chunkId] = 0;
/*****/
                            }
/*****/
                        }
/*****/
                };
/*****/
/*****/
                // no prefetching
/*****/
/*****/
                // no preloaded
/*****/
/*****/
                // no HMR
/*****/
/*****/
                // no HMR manifest
/*****/
/*****/
                __webpack_require__.0.j = (chunkId) => (installedChunks[chunkId] === 0);
/*****/
/*****/
                // install a JSONP callback for chunk loading
/*****/
                var webpackJsonpCallback = (parentChunkLoadingFunction, data) => {
```

```
/*****/
                   var [chunkIds, moreModules, runtime] = data;
/*****/
                   // add "moreModules" to the modules object,
/*****/
                   // then flag all "chunkIds" as loaded and fire callback
/*****/
                   var moduleId, chunkId, i = 0;
/*****/
                   if(chunkIds.some((id) => (installedChunks[id] !== 0))) {
/*****/
                       for(moduleId in moreModules) {
/*****/
                           if(__webpack_require__.o(moreModules, moduleId)) {
/*****/
                               __webpack_require__.m[moduleId] = moreModules[moduleId];
/*****/
/*****/
                       }
/*****/
                       if(runtime) var result = runtime(__webpack_require__);
/*****/
                   }
/*****/
                   if(parentChunkLoadingFunction) parentChunkLoadingFunction(data);
/*****/
                   for(;i < chunkIds.length; i++) {</pre>
/*****/
                       chunkId = chunkIds[i];
/*****/
                       \verb|if(\__webpack_require\__.o(installedChunks, chunkId)| \&\& installedChunks| \\
/*****/
                           installedChunks[chunkId][0]();
/*****/
/*****/
                       installedChunks[chunkIds[i]] = 0;
/*****/
/*****/
                   return __webpack_require__.0(result);
/*****/
               }
/*****/
/*****/
               var chunkLoadingGlobal = self["webpackChunk"] = self["webpackChunk"] || [];
/*****/
               chunkLoadingGlobal.forEach(webpackJsonpCallback.bind(null, 0));
/*****/
               chunkLoadingGlobal.push = webpackJsonpCallback.bind(null, chunkLoadingGloba
/*****/
           })();
/*****/
           *********************
/*****/
/*****/
/*****/ })()
dist/main.[chunkhash].js
(self["webpackChunk"] = self["webpackChunk"] || []).push([[0],[
/* 0 */
/*!*******************
  !*** ./example.js ***!
```

/*! runtime requirements: __webpack_require__.e, __webpack_require__.t, __webpack_require__
/***/ ((__unused_webpack_module, __unused_webpack_exports, __webpack_require__) => {

/*! unknown exports (runtime-defined) */

```
// some module
__webpack_require__.e(/*! import() */ 2).then(_webpack_require__.t.bind(_webpack_require__
__webpack_require__.e(/*! import() */ 3).then(_webpack_require__.t.bind(_webpack_require__

/***/ })
],

/******/ __webpack_require__ => { // webpackRuntimeModules

/******/ var __webpack_exec__ = (moduleId) => (_webpack_require__(_webpack_require__.s = /******/ var __webpack_exports__ = (_webpack_exec__(0));

/******/ }
]);
```

Info

Unoptimized

```
asset runtime~main.[chunkhash].js 12.2 KiB [emitted] (name: runtime~main)
asset main.[chunkhash].js 873 bytes [emitted] (name: main)
asset 2.[chunkhash].js 285 bytes [emitted]
asset 3.[chunkhash].js 279 bytes [emitted]
Entrypoint main 13 KiB = runtime~main.[chunkhash].js 12.2 KiB main.[chunkhash].js 873 bytes
chunk (runtime: runtime~main) main.[chunkhash].js (main) 55 bytes [initial] [rendered]
  > ./example main
  ./example.js 55 bytes [built] [code generated]
    [used exports unknown]
    entry ./example main
chunk (runtime: runtime~main) runtime~main.[chunkhash].js (runtime~main) 7.6 KiB [entry] [re
  > ./example main
 runtime modules 7.6 KiB 10 modules
chunk (runtime: runtime~main) 2.[chunkhash].js 28 bytes [rendered]
  > ./async1 ./example.js 2:0-18
  ./async1.js 28 bytes [built] [code generated]
    [used exports unknown]
    import() ./async1 ./example.js 2:0-18
chunk (runtime: runtime~main) 3.[chunkhash].js 28 bytes [rendered]
  > ./async2 ./example.js 3:0-18
  ./async2.js 28 bytes [built] [code generated]
    [used exports unknown]
    import() ./async2 ./example.js 3:0-18
webpack 5.51.1 compiled successfully
```

Production mode

```
asset runtime~main.[chunkhash].js 2.73 KiB [emitted] [minimized] (name: runtime~main)
asset main.[chunkhash].js 157 bytes [emitted] [minimized] (name: main)
asset 114. [chunkhash].js 69 bytes [emitted] [minimized]
asset 172. [chunkhash].js 69 bytes [emitted] [minimized]
Entrypoint main 2.89 KiB = runtime~main.[chunkhash].js 2.73 KiB main.[chunkhash].js 157 byte
chunk (runtime: runtime~main) 114.[chunkhash].js 28 bytes [rendered]
  > ./async1 ./example.js 2:0-18
  ./async1.js 28 bytes [built] [code generated]
    [used exports unknown]
    import() ./async1 ./example.js 2:0-18
chunk (runtime: runtime~main) 172.[chunkhash].js 28 bytes [rendered]
  > ./async2 ./example.js 3:0-18
  ./async2.js 28 bytes [built] [code generated]
    [used exports unknown]
    import() ./async2 ./example.js 3:0-18
chunk (runtime: runtime-main) main.[chunkhash].js (main) 55 bytes [initial] [rendered]
  > ./example main
  ./example.js 55 bytes [built] [code generated]
    [no exports used]
    entry ./example main
chunk (runtime: runtime~main) runtime~main.[chunkhash].js (runtime~main) 7.6 KiB [entry] [re
  > ./example main
 runtime modules 7.6 KiB 10 modules
webpack 5.51.1 compiled successfully
```