

Manifest settings

The HERE Core Runtime requires each application or platform to provide a JSON formatted manifest file with configuration information. This manifest file defines the visual properties of the app, its dependencies, and other deployment settings. The developer is in control of the app development process and how this file is hosted and configured. Both the [HERE Core CLI tool](#) and HERE Core RVM use a manifest file to launch applications as well as to make use of other configuration data.

Starting with HERE Core RVM version 15, the RVM retrieves the application manifest over a secure connection (HTTPS with a valid server certificate). This approach reduces the odds of a malicious actor substituting a modified manifest file. If the secure fetch fails, the RVM logs the failure and the reason. If security settings are not explicitly defined (see [Enable HTTPS security](#)), then RVM retries fetching the manifest file with the default settings.

Many of the properties that can be used in a manifest are defined as types in the HERE Core API. Follow links to the API reference for complete details.

The application manifest is read by the RVM when an application starts up and so it needs to be available in order for settings to take effect. If the manifest cannot be read, the RVM uses default settings where applicable, including those affect reporting about application errors. The RVM can cache manifest settings and use the cached values if the actual manifest fails to load (for example, if the file server containing it is unavailable). In such a circumstance, the last cached version of the manifest is used.

Required properties: Properties marked with a plus-sign (+) are required to be defined for their enclosing object.

Top-level properties

The following properties can be declared at the top level of a manifest file. Complex properties in this table link to later sections that detail their members.

Property	Type	Description
<code>appAssets</code>	array of objects	Defines assets that can be used by the application, as described in appAssets properties .
<code>autoShow</code>	boolean	Whether to automatically show the application's window or windows on launch.
<code>dialogSettings</code>	object	Object containing properties to customize the appearance of the RVM progress dialog, described in dialogSettings properties .
<code>deleteRemovedAssets</code>	boolean	If true: If an asset was declared in a previous version of an application manifest, but is now no longer present in the version about to be launched, then the asset is presumed unused and is removed.
<code>devtools_port</code>	integer	Not available in v25; available in <=v24 and >=v26. A port number to use to access the Chromium development tools by navigating to the selected port.
<code>extensions</code> (v38.126.83.73+)	array of objects	Definitions of Chrome extensions; refer to Microsoft Single Sign On extension support
<code>licenseKey</code>	string	A licensing identifier for each customer contract. The license indicates which HERE features and support level a customer is covered by. If you are

Property	Type	Description
		interested in a Community or Enterprise license, please contact us for one . If you're launching your app via the .NET Adapter, specify the license key in RuntimeOptions.LicenseKey . If you're using the Java Adapter, specify the license key in RuntimeConfiguration.setLicenseKey .
<code>offlineAccess</code>	boolean	Whether the application can be run without a network connection. Applies to Workspace and Notification Center as well as non-Workspace applications.
<code>overrideConfigUrl</code>	string	An override for the location of your application manifest. The RVM renames the location of the manifest and uses the new one.
<code>platform</code>	object	Defines properties for a HERE Core platform, described in Platform properties .
<code>runtime</code>	object	Configures the HERE Core Runtime that should be used with this application. Refer to Runtime properties .
<code>shortcut</code>	object	Defines a desktop shortcut for the application; refer to Shortcut properties
<code>services</code>	array of objects	Defines services that are available to this application. Each service object must have a <code>name</code> property and may

Property	Type	Description
		have a <code>manifestUrl</code> property. Refer to Services for more information.
<code>snapshot</code>	object	Defines a snapshot that can be loaded in a platform. It has one member, <code>"windows"</code> , which is an array of unnamed window options objects
<code>splashScreenForceDownload</code>	boolean	Whether the RVM re-downloads the splash screen on every launch.
<code>splashScreenImage</code>	string	File path or URL for an image to show while the application is starting up.
<code>startup_app</code>	object	Defines properties for a "classic" (standalone) app, described in startup_app properties
<code>supportInformation</code>	object	Defines properties to customize error messages, described supportInformation properties

Example top-level properties

```
{
  "splashScreenImage": "http://SERVER/IMAGE_FILE.png",
  "offlineAccess": true,
  "licenseKey": "LICENSE_KEY",
  "runtime": {
    "version": "XX.xx.xx.xx"
  },
  "startup_app": {
    "name": "APP_NAME",
    "uuid": "APP_UUID",
    "url": "APP_SERVER_NAME/START_PAGE.html",
```

```

    "autoShow": true,
    "defaultWidth": 500,
    "defaultHeight": 500
  }
}

```

appAssets properties

Property	Type	Description	Default
<code>alias</code>	string	Name of the asset. The name is used in <code>launchExternalProcess()</code> to launch the asset.	""
<code>args</code>	string	The default command line arguments for the <code>target</code> .	""
<code>mandatory</code>	boolean	When set to true, the app fails to load if the asset cannot be downloaded. When set to false, the app continues to load if the asset cannot be downloaded.	true
<code>path</code>	string	local file path to a file containing assets	""
<code>src</code>	string	URL to a zip file containing the package files (executables, DLLs, etc).	""
<code>target</code>	string	Specifies default executable to launch. This option can be overridden in <code>launchExternalProcess</code> .	""
<code>version</code>	string	Version of the package. To force new updates, increment the version.	""

Example appAssets object properties

```
"appAssets": [{
  "src": "http://SERVER_NAME/PATH/APP_NAME.zip",
  "alias": "APP_ALIAS",
  "version": "3.2.50.71",
  "target": "APP_NAME.exe",
  "args": "--flag=3",
  "mandatory": true
},
{
  "src": "http://SERVER_NAME/PATH/ANOTHER_APP.zip",
  "alias": "ANOTHER_APP_ALIAS",
  "version": "0.0.0.72"
}]
```

dialogSettings properties

Property	Type	Description	Default
<code>bgColor</code>	integer	The color of the dialog. Color value is a decimal representation of a 32 bit number (A,R,G,B). For instance : FF00FF00 or 4278255360 is fully opaque green.	4163183909
<code>logo</code>	string	The logo or image in the progress dialog. The logo must be a semi-transparent PNG, 100×25 pixels.	OpenFin

Property	Type	Description	Default
<code>progressBarBgColor</code>	integer	The background color of the area where the progress bar is displayed.	4278190080
<code>progressBarFillColor</code>	integer	The color of the progress bar.	4281513430
<code>progressBarBorderColor</code>	integer	The border color of the progress bar.	4282137660
<code>textColor</code>	integer	The color of the text displayed above the progress bar location.	4293980400

Example dialogSettings object properties

```

"dialogSettings": {
  "logo": "http://SERVER_NAME/LOGO_FILE.png",
  "bgColor": 4163183909,
  "textColor": 4293980400,
  "progressBarBgColor": 4278190080,
  "progressBarFillColor": 4281513430,
  "progressBarBorderColor": 4282137660
}

```

domainSettings properties

A `domainSettings` object may be a property of either a [platform object](#) or a [startup_app object](#). It contains a `rules` array and (starting in version 37), a `default` object that defines default values for certain properties. The `rules` property is an array of [match pattern](#) strings, and option definitions for rules that apply to domains that match the match patterns.

Refer to [Domain-based permissions](#) for details.

In Runtime v34 and earlier, the `domainSettings` object, and the corresponding `DomainSettings` and `DomainSettingsRule` interfaces, were known as `defaultDomainSettings`, `DefaultDomainSettings`, and `DefaultDomainSettingsRule`, respectively.

Platform properties

Property	Type	Description	Default
<code>allowLaunchIntoPlatform</code>	Boolean	Whether launching content manifests into the platform is allowed	(v36 & <code>false</code>)
<code>defaultWindowOptions</code>	object	Options that apply by default to all platform windows. These can include any window options , plus <code>stylesheetUrl</code> , a string containing the path of a custom CSS file to be injected into all windows that use default options.	<code>null</code>
<code>defaultViewOptions</code>	object	View options that apply by default to all views created by the platform.	<code>null</code>
<code>disableErrorReporting</code>	Boolean	(RVM v10+) Whether to show the Report Error button in RVM error reporting dialogs for the application.	false

Property	Type	Description	Default
		This property can also be set in Desktop owner settings , which takes precedence over application manifest settings.	
<code>domainSettings</code>	object	Contains a <code>rules</code> property that controls access to file downloads and HERE Core APIs, based on match rules for domains; described in domainSettings properties . (Was <code>defaultDomainSettings</code> in v34 and earlier)	null
<code>enableAppLogging</code>	boolean	If <code>true</code> , a log file is generated and console logs from apps running within the platform can be included in the platform's application log , subject to logging levels defined in window options, view options, or domain-based settings . If <code>false</code> , no log file is created for any console logs. (v43+)	<code>true</code>
<code>reportErrorUrl</code>	string	(RVM 10+) The URL of a server to send error reports to. For information on hosting a custom error reporting	<code>"http://dl.op.servi.error"</code>

Property	Type	Description	Default
		service, contact support@openfin.co .	
<code>securityPermissions</code> <code>MoreInfoUrl</code>	string	URL to a page containing information about API security permissions, used in the Review Security Permissions dialog box. (RVM v6.9+)	<code>"http://localhost:8080/platform/provider.html"</code>
<code>viewProcessAffinityStrategy</code>	string	Possible values are "same" and "different", indicating whether windows and views (with a given origin) should be created in the same process or different ones.	<code>"same"</code>

Example platform object properties

```

"platform": {
  "uuid": "register-with-store",
  "icon": "http://localhost:8080/favicon.ico",
  "autoShow": false,
  "providerUrl": "http://localhost:8080/platform/provider.html",
  "preventQuitOnLastWindowClosed": true,
  "permissions": {
    "System": {
      "launchExternalProcess": true,
      "downloadAsset": true,
      "openUrlWithBrowser": {
        "enabled": true,
        "protocols": ["mailto"]
      }
    }
  },
  "defaultWindowOptions": {

```

```

    "permissions": {
      "System": {
        "openUrlWithBrowser": {
          "enabled": true,
          "protocols": ["mailto"]
        }
      }
    },
    "workspacePlatform": {
      // Configure page tab width
      "pageTabDimensions": {
        "minWidth": "30px",
        "maxWidth": "400px"
      },
      // Configure view tab width
      "viewTabDimensions": {
        "minWidth": "50px",
        "maxWidth": "400px"
      }
    }
  }
}

```

A platform object inherits, and therefore can accept, properties from the following:

- [Application options](#)
- [Window creation options](#)

The following Application options are *usually* specified for a platform:

Property	Type	Description	Default
<code>uuid+</code>	string	Required The <i>unique universal identifier</i> of the application; it must be unique within the set of all applications running in the HERE Core Runtime.	""
<code>name</code>	string	The name of the application. This name is used in error messages, API security dialog boxes, and the directory name in <code>%localappdata%/</code>	""

Property	Type	Description	Default
		<code>OpenFin/apps/<name><hash></code> .	
<code>url</code>	string	The URL of the application's main window.	<code>"about-blank"</code>

A platform object can also contain a `permissions` object, which declares the secured APIs used by the application. Refer to [Declaring APIs in an application manifest](#) for details.

Runtime properties

Property	Type	Description	Default
<code>version+</code>	string	Specifies the version of the Runtime the app should use. The value can be a version or a release channel. For the available values, see the versions page .	""
<code>fallbackVersion</code>	string	The RVM will fall back to this version if it fails to retrieve the desired Runtime version, assuming this version is already installed. (RVM 2.8+)	""
<code>futureVersion</code>	string	The RVM preloads this newer HERE Core version to prepare for an upgrade. The download occurs silently and does not interrupt the startup of your app in the current version. (RVM 3.3+) The desktop is checked to make sure an upgrade to the specified	""

Property	Type	Description	Default
		<p>version can succeed. The results are reported to the application owner by calling the <code>getInstalledRuntimes</code> method,</p> <p>In RVM v10 and later, the string is validated and sent to the HERE licensing server so that HERE deployment support can track upgrade progress.</p>	
<code>arguments</code>	string	<p>Command line arguments to set when launching the runtime. The HERE Core Runtime supports Chromium command line switches e.g.: <code>--disable-accelerated-compositing</code> <code>--enable-threaded-compositing</code> or <code>--user-gesture-required</code> (to prevent media autoplay). For more information, refer to Current Chromium Switches.</p>	""
<code>forceLatest</code>	boolean	<p>If using runtime channels, setting this to true forces the RVM to always get the latest version before launching the application (prevents background installs).</p>	false
<code>enableCacheMigration</code>	boolean	<p>If true, then when the RVM detects that an app has upgraded its Runtime version (e.g. from v8.1.0.0 -> 8.2.0.0), it will migrate Local Storage, IndexedDB, and</p>	true

Property	Type	Description	Default
		winposCache data to the cache of the new version	
<code>policies</code>	object	Contains a <code>CloudAPAuthEnabled</code> string property (Windows only). If set to <code>"enabled"</code> and <code>--security-realm</code> is set either on the command line or in the <code>arguments</code> property, then the Chromium CloudAPAuthEnabled policy is enabled. This property is ignored if the <code>CloudAPAuthEnabled</code> policy is set as a Group Policy . (v43+)	""

Example runtime object properties

```
"runtime": {
  "arguments": "",
  "version": "15.80.48.12",
  "fallbackVersion": "14.78.48.16",
  "futureVersion": "16.83.53.26"
}
```

Shortcut properties

Property	Type	Description	Default
<code>company</code> +	string	Company name for the application shortcut.	""

Property	Type	Description	Default
<code>description</code>	string	A short description of the application shortcut. It is shown when hovering over the shortcut icon.	""
<code>diagnostics-shortcut</code>	boolean	Creates a shortcut menu-item to run your application in Diagnostics Mode. Diagnostics Mode provides you and your end-users an opportunity to recreate previously reported issues within your application and remotely deliver the Runtime debug logs to HERE. It can also be manually triggered by adding "--diagnostics" to the command line for your application.	true in RVM v5.2+
<code>force</code>	boolean	If set to true, a desktop icon is always created on application start-up (even when user has deleted it). If set to false, a desktop icon is created on initial application launch but not created on subsequent application launches.	false
<code>icon+</code>	string	Location for the icon image to be used when installing the application shortcut. Note: This property must point to an .ico file.	""

Property	Type	Description	Default
<code>name</code>	string	Name of the application to display with the shortcut.	""
<code>startMenuRootFolder</code>	string	Set this value with a folder path (e.g., <code>foo</code>) and the RVM creates the start menu shortcut under <code>foo/Company/App</code> .	""
<code>target</code>	string	Locations for where the application shortcut is added on the desktop owner's machine. Available options are "desktop", "start-menu" and "automatic-start-up".	["start-menu", "desktop"]
<code>uninstall-shortcut</code>	boolean	If true, creates an uninstall shortcut for the app	true

Example shortcut object properties

```

"shortcut": {
  "company": "COMPANY_NAME",
  "description": "APP_DESCRIPTION",
  "icon": "http://SERVER_NAME/PATH/ICON_FILE.ico",
  "name": "APP_NAME",
  "target": ["desktop", "start-menu", "automatic-start-up"],
  "startMenuRootFolder": "FOLDER_NAME"
}

```

startup_app properties

A startup_app object can accept properties from any of the following:

- [Application options](#)
- [Window creation options](#)

A `startup_app` object can also contain:

- A `permissions` object, which declares the secured APIs used by the application. Refer to [Declaring APIs in an application manifest](#) for details.
- A `domainSettings` object to control file downloads and access to the HERE Core API, through domain-based rules.

The following Application options are *usually* specified for a classic app:

- `uuid`+: The *unique universal identifier* of the application; it must be unique within the set of all applications running in the HERE Core Runtime.
- `name`: The name of the application. This name is used in error message and permissions dialog boxes and in the directory name in `%localappdata%/OpenFin/apps/<name><hash>`. If there is a [Platform properties](#) section of the manifest, then this value is ignored in favor of the `name` value provided in the platform section.
- `url`: The URL of the application's main window; defaults to `"about-blank"`.

The following option can be specified to customize the dialog box that appears when a user is prompted to allow or deny security permissions to an app:

- `securityPermissionsMoreInfoUrl`: A string containing a URL to be linked from "more information" in the dialog box.

Example `startup_app` options

```
"startup_app": {
  "name": "APP_NAME",
  "icon": "http://SERVER_NAME/PATH/ICON_FILE.ico",
  "description": "APP_NAME",
  "url": "http://SERVER_NAME/PATH/START_PATH.html",
  "uuid": "UUID",
```

```

"backgroundColor": "#ffffff",
"contextMenu": true,
"contentNavigation": {
  "whitelist": ["https://ALLOWED_DOMAIN/*"]
},
"autoShow": true,
"defaultWidth": 480,
"maxWidth": 480,
"minWidth": 480,
"maxHeight": 50,
"defaultHeight": 50,
"minHeight": 50,
"defaultTop": 50,
"defaultLeft": 10,
"resizable": false,
"maximizable": false,
"frame": false,
"cornerRounding": {
  "width": 5,
  "height": 5,
},
"taskbarIconGroup": "APP_GROUP",
"loadErrorMessage": "There was an error loading the
application."
}

```

supportInformation properties

The following properties are used to customize the dialog box that appears when an application fails to load, that is, the *app load error dialog*. Note that if the application's manifest itself cannot be reached, the app load error dialog uses default values for these settings.

Property	Type	Description	Default
<code>company</code> +	string	Displays the company name in the title bar for the app load error dialog.	""

Property	Type	Description	Default
<code>email+</code>	string	<p>The email address displayed in any RVM error dialogs for the application. If the user clicks Report Error in the dialog and <code>forwardErrorReports</code> is true, this address is sent to the error reporting service along with the error report; the default service then sends a copy of the error report to this address. (A custom error reporting service might behave differently.)</p>	"support@openfir"
<code>enableErrorReporting</code>	boolean	<p>A legacy property to control the possibility for an end user to use the Report Error button in the RVM error dialog. If set to false, in RVM 10 and later, the button is hidden; in RVM 9 and earlier, the button is disabled. The default HERE Core error reporting service can send a copy of the error report to the email address set in the <code>email</code> property. In RVM 10 and later, the preferred way to control this button is with the</p>	true

Property	Type	Description	Default
		<p><code>disableErrorReporting</code> option under the <code>platform</code> or <code>startup_app</code> properties; the new property can also be set in Desktop owner settings and therefore can affect all HERE Core applications on the system.</p>	
<code>forwardErrorReports</code>	boolean	<p>If set to true, the email address specified in the <code>email</code> property is included in the information sent to the error reporting service. The HERE error reporting service sends a copy of the error report to this address. (A custom error reporting service might behave differently.)</p>	false
<code>moreInfoUrl</code>	string	<p>URL that is linked from "troubleshooting steps" in the RVM error dialog. See the Sample troubleshooting guide for example content of such a page</p>	<p>If no value is specified, the line containing the link suppressed: "For help, visit troubleshooting steps."</p>

Property	Type	Description	Default
<code>product</code> +	string	Displays the product name or application name in the RVM error dialog.	""

Example supportInformation object properties

```
"supportInformation" : {  
  "company": "COMPANY_NAME",  
  "product": "PRODUCT_NAME",  
  "email": "EMAIL_ADDRESS",  
  "forwardErrorReports": true,  
  "moreInfoUrl": "TROUBLESHOOTING_INFO_URL"  
}
```

Have questions? Get in touch with us at support@here.io.