# Manifest

Component for reading [phar.io](https://phar.io/) manifest information from a [PHP Archive (PHAR)](http://php.net/phar).

## Installation

You can add this library as a local, per-project dependency to your project using [Composer](https://getcomposer.org/):

composer require phar-io/manifest

If you only need this library during development, for instance to run your project's test suite, then you should add it as a development-time dependency:

composer require --dev phar-io/manifest

## Usage Examples

### Read from manifest.xml

use PharIo\Manifest\ManifestLoader;

use PharIo\Manifest\ManifestSerializer;

$manifest = ManifestLoader::fromFile('manifest.xml');

var\_dump($manifest);

echo (new ManifestSerializer)->serializeToString($manifest);

Output object(PharIo\Manifest\Manifest)#14 (6) {

["name":"PharIo\Manifest\Manifest":private]=>

object(PharIo\Manifest\ApplicationName)#10 (1) {

["name":"PharIo\Manifest\ApplicationName":private]=>

string(12) "some/library"

}

["version":"PharIo\Manifest\Manifest":private]=>

object(PharIo\Version\Version)#12 (5) {

["originalVersionString":"PharIo\Version\Version":private]=>

string(5) "1.0.0"

["major":"PharIo\Version\Version":private]=>

object(PharIo\Version\VersionNumber)#13 (1) {

["value":"PharIo\Version\VersionNumber":private]=>

int(1)

}

["minor":"PharIo\Version\Version":private]=>

object(PharIo\Version\VersionNumber)#23 (1) {

["value":"PharIo\Version\VersionNumber":private]=>

int(0)

}

["patch":"PharIo\Version\Version":private]=>

object(PharIo\Version\VersionNumber)#22 (1) {

["value":"PharIo\Version\VersionNumber":private]=>

int(0)

}

["preReleaseSuffix":"PharIo\Version\Version":private]=>

NULL

}

["type":"PharIo\Manifest\Manifest":private]=>

object(PharIo\Manifest\Library)#6 (0) {

}

["copyrightInformation":"PharIo\Manifest\Manifest":private]=>

object(PharIo\Manifest\CopyrightInformation)#19 (2) {

["authors":"PharIo\Manifest\CopyrightInformation":private]=>

object(PharIo\Manifest\AuthorCollection)#9 (1) {

["authors":"PharIo\Manifest\AuthorCollection":private]=>

array(1) {

[0]=>

object(PharIo\Manifest\Author)#15 (2) {

["name":"PharIo\Manifest\Author":private]=>

string(13) "Reiner Zufall"

["email":"PharIo\Manifest\Author":private]=>

object(PharIo\Manifest\Email)#16 (1) {

["email":"PharIo\Manifest\Email":private]=>

string(16) "reiner@zufall.de"

}

}

}

}

["license":"PharIo\Manifest\CopyrightInformation":private]=>

object(PharIo\Manifest\License)#11 (2) {

["name":"PharIo\Manifest\License":private]=>

string(12) "BSD-3-Clause"

["url":"PharIo\Manifest\License":private]=>

object(PharIo\Manifest\Url)#18 (1) {

["url":"PharIo\Manifest\Url":private]=>

string(26) "https://domain.tld/LICENSE"

}

}

}

["requirements":"PharIo\Manifest\Manifest":private]=>

object(PharIo\Manifest\RequirementCollection)#17 (1) {

["requirements":"PharIo\Manifest\RequirementCollection":private]=>

array(1) {

[0]=>

object(PharIo\Manifest\PhpVersionRequirement)#20 (1) {

["versionConstraint":"PharIo\Manifest\PhpVersionRequirement":private]=>

object(PharIo\Version\SpecificMajorAndMinorVersionConstraint)#24 (3) {

["originalValue":"PharIo\Version\AbstractVersionConstraint":private]=>

string(3) "7.0"

["major":"PharIo\Version\SpecificMajorAndMinorVersionConstraint":private]=>

int(7)

["minor":"PharIo\Version\SpecificMajorAndMinorVersionConstraint":private]=>

int(0)

}

}

}

}

["bundledComponents":"PharIo\Manifest\Manifest":private]=>

object(PharIo\Manifest\BundledComponentCollection)#8 (1) {

["bundledComponents":"PharIo\Manifest\BundledComponentCollection":private]=>

array(0) {

}

}

}

<?xml version="1.0" encoding="UTF-8"?>

<phar xmlns="https://phar.io/xml/manifest/1.0">

<contains name="some/library" version="1.0.0" type="library"/>

<copyright>

<author name="Reiner Zufall" email="reiner@zufall.de"/>

<license type="BSD-3-Clause" url="https://domain.tld/LICENSE"/>

</copyright>

<requires>

<php version="7.0"/>

</requires>

</phar>

### Create via API

$bundled = new \PharIo\Manifest\BundledComponentCollection();

$bundled->add(

new \PharIo\Manifest\BundledComponent('vendor/packageA', new \PharIo\Version\Version('1.2.3-dev')

)

);

$manifest = new PharIo\Manifest\Manifest(

new \PharIo\Manifest\ApplicationName('vendor/package'),

new \PharIo\Version\Version('1.0.0'),

new \PharIo\Manifest\Library(),

new \PharIo\Manifest\CopyrightInformation(

new \PharIo\Manifest\AuthorCollection(),

new \PharIo\Manifest\License(

'BSD-3-Clause',

new \PharIo\Manifest\Url('https://spdx.org/licenses/BSD-3-Clause.html')

)

),

new \PharIo\Manifest\RequirementCollection(),

$bundled

);

echo (new ManifestSerializer)->serializeToString($manifest);

Output <?xml version="1.0" encoding="UTF-8"?>

<phar xmlns="https://phar.io/xml/manifest/1.0">

<contains name="vendor/package" version="1.0.0" type="library"/>

<copyright>

<license type="BSD-3-Clause" url="https://spdx.org/licenses/BSD-3-Clause.html"/>

</copyright>

<requires>

<php version="\*"/>

</requires>

<bundles>

<component name="vendor/packageA" version="1.2.3-dev"/>

</bundles>

</phar>