

Concept\Config & Storage Documentation

Overview

The `Concept\Config` component is a flexible and extensible configuration system built around the `DotArray` and plugin architecture. It supports loading configurations from multiple sources (PHP, JSON), dynamic merging, contextual resolution, and custom plugin processing.

Core Components

1. ConfigInterface / Config: main API for reading and merging config data.
2. StorageInterface / Storage: internal array storage (DotArray-based).
3. Context: represents the runtime/environment context.
4. Adapters (PhpAdapter, JsonAdapter): load data from files.
5. Plugins: extend configuration (env, include, reference, context).
6. Resolver: resolves placeholders like \${ENV_VAR} or @include().

Usage Example: Storage (DotArray)

```
use Concept\Config\Storage;
```

```
$storage = new Storage([  
    'db' => [  
        'host' => 'localhost',  
        'port' => 3306,  
        'credentials' => [  
            'user' => 'root',  
            'pass' => 'secret'  
        ]  
    ]  
]);
```

```
$host = $storage->get('db.host'); // 'localhost'  
$user = $storage->get('db.credentials.user');  
$storage->set('app.debug', true);  
$storage->toggle('app.debug');
```

Usage Example: Config Loading

```
use Concept\Config\Config;

use Concept\Config\Resource;

$config = new Config();
$config->load(new Resource('config/app.php'));
$debug = $config->get('app.debug');
```

Deep DotArray Access

```
$array = new DotArray([
    'users' => [
        'admin' => [
            'email' => 'admin@example.com',
            'settings' => [
                'notifications' => true
            ]
        ]
    ]
]);

$email = $array->get('users.admin.email');
$notifications = $array->get('users.admin.settings.notifications');
```

Plugins Overview

- EnvPlugin: replaces \${ENV_VAR} with environment values
- IncludePlugin: includes other config files
- ReferencePlugin: resolve @ref() expressions
- ContextPlugin: inject values from the context
- InterpolatorPlugin: inline value replacement

Advanced Example: Config Factory

```
use Concept\Config\Factory;

$config = (new Factory())->fromFiles([
```

```
'config/app.php',  
'config/db.json'  
]);
```

```
$config->get('app.name');
```

```
$config->get('db.connections.mysql.host');
```