Service Providers

Overview

Service providers are the central place of application bootstrapping. All of a hub's core services are bootstrapped via service providers.

Every application or "client" type, such as "administrator" or "api", has their own list of services. These are all of the service provider classes that will be loaded for your application. Providers are laze loaded, meaning they will not be loaded on every request, but only when the services they provide are actually needed.

Standard Provider

Service providers must extend the HubzeroBaseServiceProvider class and are required to define at least one method: register(). Aside from the register method, a boot method may also be defined, which allows for a little setup or processing that may need to occur after all services have been registered.

The Register Method

```
<?php

namespace AppProviders;

use HubzeroBaseServiceProvider;

class FooServiceProvider extends ServiceProvider
{
    /**
     * Register services in the container.
     *
     * @return void
     */
    public function register()
    {
        $this->app['foo'] = function ($app)
        {
            return new Foo();
        });
    }
}
```

The Boot Method

The boot method is called after all other service providers have been registered, giving it access to all other services that have been registered by the framework.

```
<?php

namespace AppProviders;

use HubzeroBaseServiceProvider;

class FooServiceProvider extends ServiceProvider
{
    /**
     * Perform post-registration booting of services.
     *
     * @return void
     */
    public function boot()
    {
       $this->app['foo']->bar();
    }
}
```

Middleware Provider

A Middleware provider is an extended service provider with a handle method. Rather than extending HubzeroBaseServiceProvider, these providers extend HubzeroBaseMiddleware. While they can register services, they are not required to do so. Instead, they handle (i.e., modify) the incoming request and outgoing response.

The Handle Method

The handle method is called after the application has been booted and accepts a HubzeroHttpRequest object as the only argument.

```
<?php

namespace AppProviders;

use HubzeroBaseServiceProvider;
```

SERVICE PROVIDERS

```
class FooServiceProvider extends ServiceProvider
{
    /**
     * Perform post-registration booting of services.
     *
     * @return void
     */
    public function handle(Request $request)
    {
       // Forcefully set the 'foo' var to 'bar'
       $request->setVar('foo', 'bar');

       return $this->next($request);
    }
}
```