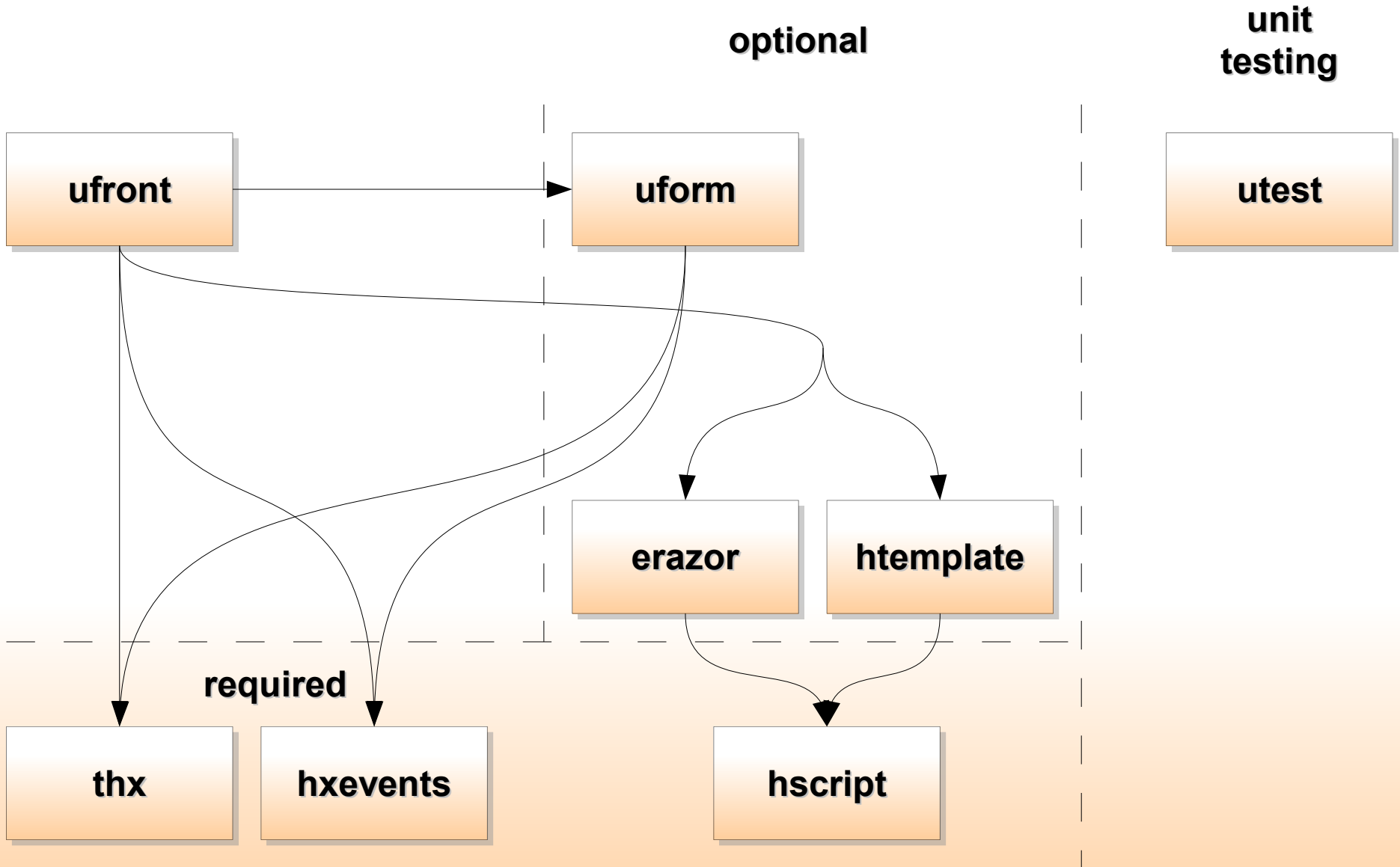




ufront / uform:
web development made easy
by
Franco Ponticelli



ufront relations





utest

Cross Platform Unit Testing Library

```
import utest.Runner;
import utest.ui.Report;

class Main
{
    static function main()
    {
        var runner = new Runner();

        runner.addCase(new Test());

        Report.create(runner);
        runner.run();
    }
}
```

```
import utest.Assert;

class Test
{
    public function testInt()
    {
        Assert.isTrue(1 > 0);
    }

    public function new();
}
```



hxevents

Tiny Event Framework

```
import hxevents.Dispatcher;

class Main
{
    public var value(default, setValue) : Int;
    public var onChange(default, null) : Dispatcher<Int>;
    public function new()
    {
        onChange = new Dispatcher();
        value = 0;
    }

    function setValue(v : Int)
    {
        onChange.dispatch(value = v);
        return value;
    }

    static function main()
    {
        var main = new Main();
        main.onChange.add(function(v) trace("value changed to: " + v));
        main.value = 10;
        main.value = 7;
    }
}
```



thx

General Purpose Library

- I18n (culture/language/translation)
- format functions
- html parser/writer
- xml writer
- csv
- ini
- json
- validation
- sys (typedefs for neko/php)
- many helpers for basic types: String, Arrays, Ints, Floats ...
- colors ... and ???



erazor

Template Engine

```
var template = new Template(template);  
trace(template.execute({ title : "contacts", list : people }));
```

```
<h1>@title</h1>  
@if(list.length > 0)  
{  
    <div class="warning">empty list</div>  
} else {  
    <ul>  
        @for(person in list)  
        {  
            <li><a href="mailto:@person.email">@person.name (@person.age())</a></li>  
        }  
    </ul>  
}
```

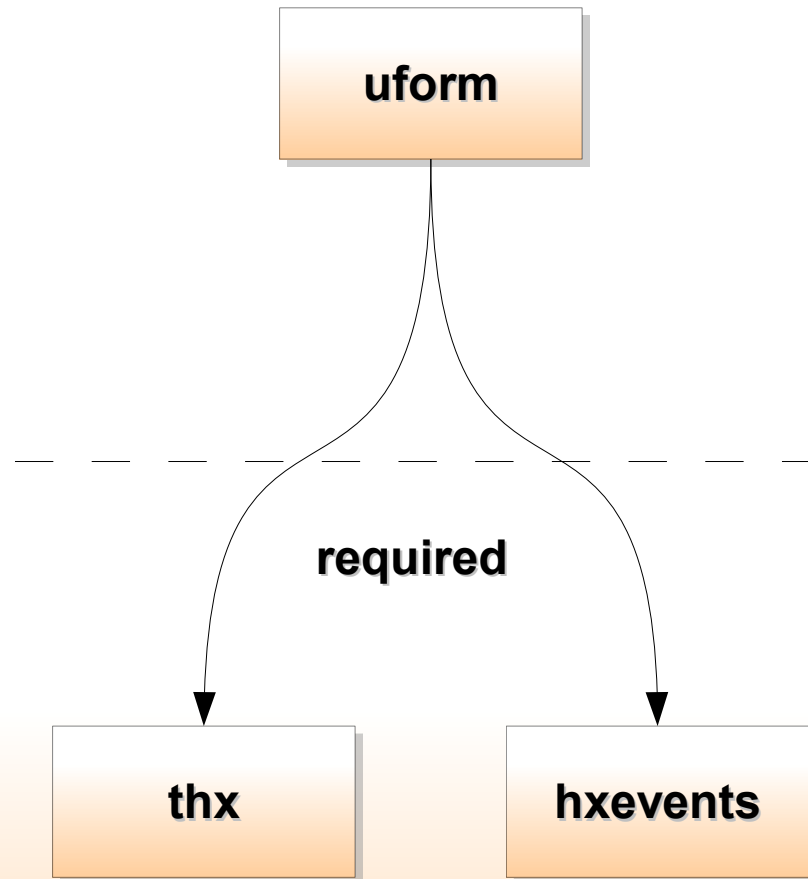


uform

Form Management Library



uform relations





uform

Validation

```
import uform.model.core.Schema;
import uform.model.value.Value;

class DemoSchema
{
    public static function main()
    {
        var schema = new Schema()
            .add("name", Value.string(3))
            .addOpt("age", Value.int(10));

        var ob = { name : "Franco", age : null };

        trace(schema.validate(ob));

        ob.age = 3;
        trace(schema.validate(ob));

        ob.age = 39;
        trace(schema.validate(ob));
    }
}
```



uform

Logic Components

```
import uform.ui.comp.NumberControl;

class DemoControl
{
    public static function main()
    {
        var control = new NumberControl("mycontrol");

        control.min.set(1);
        control.max.set(10);

        trace(control.value.isNull());

        control.value.set(20); // constrained to 100
        trace(control.value.get());

        control.value.set(5);
        trace(control.value.get());
    }
}
```



ufront

MVC Framework for Server-Side Web Development

HttpRequest

```
query : Hash<String>
post  : Hash<String>
cookies : Hash<String>
clientHeaders : Hash<String>
httpMethod : String
url : String
setUploadHandler(handler)
...
```

HttpResponse

```
write(s)
writeBytes(b)
setHeader(name, value)
setCookie(cookie)
redirect(url)
...
```

Session

```
dispose()
clear()
get(name) : Dynamic
set(name, value)
exists(name) : Bool
remove(name)
id() : String;
```

simplified upload handler: IHttpUploadHandler

```
uploadStart(name, filename)
uploadProgress(bytes, pos, len)
uploadEnd()
```



ufront

HttpApplication

AsyncDispatcher

`onBeginRequest`
`onResolveRequestCache`
`onPostResolveRequestCache`
`onMapRequestHandler`
`onPostMapRequestHandler`
`onAcquireRequestState`
`onPostAcquireRequestState`
`onPreRequestHandlerExecute`
`onPostRequestHandlerExecute`
`onReleaseRequestState`
`onPostReleaseRequestState`
`onUpdateRequestCache`
`onPostUpdateRequestCache`
`onLogRequest`
`onPostLogRequest`
`onApplicationError`

Dispatcher

`onEndRequest`



ufront

Modules

- `UrlRoutingModule`
- `ErrorModule`
- `TraceModule`



ufront

Routing

```
var routes = new RouteCollection()  
  .addRoute("/", { controller : "home", action : "index" } )  
  .addRoute("/edit/{id}", { controller : "crud", action : "edit" } )  
  .addRoute("/list/{?page}", { controller : "crud", action : "edit", page : 1 } )  
  .addRoute("/path/{*path}", { controller : "color", action : "listJson" } )
```

responds to/generates
/

responds to/generates
/edit/7 or /edit/franco

responds to/generates
/edit/my/long/path



ufront

Url Filters

```
IUrlFilter
```

```
function filterIn(url : PartialUrl, request : HttpRequest)
```

```
function filterOut(url : VirtualUrl, request : HttpRequest)
```

- DirectoryUrlFilter
- PathInfoUrlFilter
- QueryStringUrlFilter
- SegmentToParamUrlFilter



ufront

Controller Output

```
class Home extends Controller {  
  public function index() {  
    return new ViewResult( {  
      title : "ufront and friends ..."  
    } );  
  }  
}
```

- String
- ViewResult
- JsonResult
- ContentResult
- FileContentResult
- FilePathResult
- RedirectResult
- ForwardResult



ufront

Controller Input

```
.addRoute("/", { controller : "home", action : "index" } )
```

```
public function index()
```

```
.addRoute("/edit/{id}", { controller : "crud", action : "edit" } )
```

```
public function edit(id : Int)
```

or

```
public function edit(id : String)
```

```
.addRoute("/list/{?page}", { controller : "crud", action : "edit", page : 1 } )
```

```
public function list(?page : Int)
```

```
.addRoute("/list/{?page}", { controller : "crud", action : "edit" } )
```

```
public function list(page = 1)
```



ufront

Attributes / Controller Filters

```
@authorize({roles:["admin"]})  
class AdminController extends Controller  
{  
  ...  
}
```



ufront

View Helpers

- FormatHelper
- HtmlHelper
- TemplateHelper
- TranslationHelper
- UFormHelper
- UrlHelper



D3



what next?

ufront

- more attributes for: Content-Types, Output Filter ((X)HTML beautification) ...
- support for haXe remoting
- integration with client side
- remove dependency on Rtti in Controller
- support NodeJS?
- support C++?
- model?



what next?

uform

- better integration with ufront
- connect to custom classes
- JS Linker
- linkers to other UI
- more controls

utest

- clean-up Html generation
- test server?

thx

- complete and improve D3.js port
- integrate event system?
- integrate template engine?
- integrate utest?

erazor

- support generation of haXe classes
- improve syntax (few glitches and removal of brackets under certain circumstances)